ระบบแม่ข่ายโทรสาร

Fax Server System

โดย

นาย วิศิษฏ์ มูลทรัพย์ รหัส 35103247

นาย สุรพงษ์ ทองพลับ รหัส 35103256

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิศวกรรมคอมพิวเตอร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2537

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบแม่ข่ายโทรสาร

Fax Server System

โดย

นาย วิศิษฎ์ มูลทรัพย์ รหัส 35103247

นาย สุรพงษ์ ทองพลับ รหัส 35103256

อาจารย์ที่ปรึกษา

อ.วัชระ ฉัตรวิริยะ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิศวกรรมคอมพิวเตอร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2537

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2537

ภาควิชา วิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบแม่ข่ายโทรสาร (Fax Server System)

ผู้จัดทำ

1. นายวิศิษฏ์ มูลทรัพย์ รหัสประจำตัว 35103247
2. นายสุรพงษ์ ทองพลับ รหัสประจำตัว 35103256

อาจารย์ที่ปรึกษา

(อาจารย์วัชระ ฉัตรวิริยะ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบแม่ข่ายส่งโทรสาร

วิศิษฎ์ มูลทรัพย์
สุรพงษ์ ทองพลับ
อ. วัชระ ฉัตรวิริยะ อาจารย์ที่ปรึกษา
ปีการศึกษา 2537

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้ได้เสนอระบบส่งโทรสาร โดยเครื่องหนึ่งให้บริการกับผู้ใช้หลายคนหรือหลายงานในเครือข่ายคอมพิวเตอร์ ผู้ใช้ต้องสร้างไฟล์ข้อมูลแบบข้อความภาษาอังกฤษและภาษาไทยแบบ สมอ. ซึ่งสามารถเลือกทำตัวอักษรพิเศษ เช่น ตัวหนา, ตัวใหญ่, ตัวขีดเส้นใต้ได้ด้วย ระบบจะอ่านไฟล์ข้อความดังกล่าวมาแปลงเป็นรูปแบบของโทรสาร แล้วส่งไปยังปลายทาง ระบบแม่ข่ายจะรับงานจากเครื่องลูกข่ายมาทำการจัดเรียงลำดับตามระดับความเร่งด่วนของงาน และตามเวลาที่ผู้ใช้กำหนดไว้ในโปรแกรม แล้วทำการส่งโทรสารให้ต่อไป

ปริญญานิพนธ์ฉบับนี้ได้ใช้รูปแบบสัญญาณในกลุ่ม3 ตามมาตราฐาน CCITT และคำสั่งที่ใช้โทรสารในกลุ่มที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Fax Server System

Wisit Mulsup

Surapong Tongplub

Watchara Chatwiriya Advisor

1994

Abstract

This thesis presents the Facimile system (Only sending). One facimile can serve simutinuously multiuser or multitasking in network. User have to create text file and TIS THAI code that also can choose bold type,enlarge type,underline character type. The software will read that text file and convert to facimile format then send to target. The Server system receive the job from the client and sort by priority order and time specified by user in program before send to the remote facimile.

This project used signal format group 3rd by CCITT standard and facimile controlling group 2nd.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทที่ 1 บทนำ	1
บทที่ 2 ทฤษฎีและหลักการ	2
2.1 เครื่องโทรสาร	2
2.2 เครื่องโทรสารโมเด็ม	5
2.3 การเข้ารหัส	11
2.4 การบริหารจัดลำดับงาน	15
บทที่ 3 การออกแบบการทำงาน	21
3.1 ซอฟต์แวร์ที่ทำงานบนเครื่องลูกข่าย	21
3.2 ซอฟต์แวร์ที่ทำงานบนเครื่องแม่ข่าย	27
บทที่ 4 ขั้นตอนการทำงาน	30
บทที่ 5 สรุปและวิจารณ์	36
ภาคผนวก	
เอกสารอ้างอิง	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

ระบบแม่ข่าย(Server system) เป็นระบบที่ได้รับความนิยมในปัจจุบัน เนื่องจากระบบแม่ข่ายจะช่วยลดการซ้ำซ้อนในการทำงานนั้นๆ ลงเช่นระบบแม่ข่ายแฟ้มข้อมูลช่วยลดการจัดการของแฟ้มข้อมูล รวมทั้งสะดวกในการบริหารและค้นหา หรือในระบบแม่ข่ายงานพิมพ์ก็สามารถใช้เครื่องพิมพ์ร่วมกันได้อย่างมีประสิทธิภาพ

ดังนั้นในปริญญาณนิพนธ์นี้จึงได้นำหลักการของระบบแม่ข่ายมาประยุกต์ใช้ในการส่งโทรสาร โดยให้ระบบแม่ข่ายโทรสารรับงานที่ได้รับจากระบบลูกข่าย ซึ่งเป็นแฟ้มข้อมูลที่ได้ทำการแปลงให้อยู่ในรูปแบบของแฟกซ์ มาทำการจัดเรียงลำดับตามความต้องการของผู้ใช้ และทำการส่งโทรสารตามลำดับงานที่เข้ามาและ ตามเวลาที่ผู้ใช้กำหนด

โดยระบบสามารถที่จะส่งโทรสารที่เป็นแฟ้มข้อมูลแบบตัวอักษรทั้งภาษรอังกฤษและภาษาไทย ที่เป็นรหัส สมอ.ได้ พร้อมทั้งเครื่องลูกข่ายบนเครือข่ายคอมพิวเตอร์สามารถส่งโทรสารได้พร้อมๆ กัน ซึ่งทำให้งานเกี่ยวกับการส่งโทรสารมีความคล่องตัวเพิ่มขึ้น

จะเห็นว่าในการทำงานของระบบแม่ข่ายโทรสารจะเอื้ออำนวยความสะดวกในการส่งโทรสาร ลดความยุ่งยากในการจัดการเกี่ยวกับ เอกสารโทรสาร และ สามารถใช้ทรัพยากรที่มีอยู่ร่วมกันอย่างเต็มประสิทธิภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและหลักการ

2.1 เครื่องโทรสาร

เครื่องโทรสารในปัจจุบันได้เป็นออกเป็นกลุ่มตามมาตราฐานของ CCITT ได้เป็นกลุ่มต่างๆ 4 กลุ่มคือ กลุ่ม1 ถึง กลุ่ม4 โดยเครื่องโทรสารในกลุ่ม1 และ กลุ่ม2 ใช้การส่งข้อมูลโดยสัญญาณที่ใช้เป็นสัญญาณอนาล็อกล้วนๆ ส่วนในกลุ่ม3 สัญญาณที่ใช้จะเป็นสัญญาณดิจิตอล และในกลุ่ม4 เป็นมาตราฐานใหม่ที่เสริมความสามารถให้แก่กลุ่ม3 โดยเฉพาะคุณสมบัติทางด้านเน็ตเวิร์ก และเทเลเท็กซ์ ซึ่งกำลังอยู่ระหว่างการศึกษาเครื่องโทรสารส่วนมากในปัจจุบันเป็นเครื่องโทรสารในกลุ่ม3 แทบทั้งสิ้น และโครงงานนี้ก็ได้ใช้เครื่องโทรสารในกลุ่ม3 จึงขอกล่าวถึงเฉาพะในกลุ่ม3 เท่านั้น

เครื่องโทรสารกลุ่ม3

ในปี 1980 ทาง CCITT ได้ประกาศมาตราฐานระบบสื่อสารแบบ T.4 ขึ้นมา และได้มีการปรับปรุงต่อมา จนในปี 1990 ก็ได้มีการประกาศมาตรฐานโทรสารกลุ่มใหม่ในกลุ่ม G3 โดยมีข้อกำหนดของ CCITT ที่เกี่ยวข้องกับ 3 อยู่หลายตัว เช่น V.17 V.29 และ V.33 เป็นต้น คุณสมบัติที่เพิ่มขึ้นมานอกเหนือจากความละเอียดและความเร็วแล้ว ยังมีความสามารถในการส่งข้อมูลที่อยู่ในรูปของไบนารีด้วย โทรสารในกลุ่ม 3 มีความละเอียดสูงขึ้นเป็น 200 บรรทัดต่อนิ้ว (ในโหมด Fine) และที่ความละเอียดปกติ 100 จุดต่อนิ้ว (ในโหมด Standard) และมีการแทนค่าของข้อมูลด้วยระดับสีเพียงสองระดับ คือสีขาวและสีดำ การสร้างระดับสีอื่นๆ ในโทรสาร G3 นี้จะใช้วิธีทาง Image Processing เช่น Dither เข้ามาช่วย (ในโหมด Halftone)

นอกจากนี้ยังสามารถเพิ่มประสิทธิภาพของการสื่อสารข้อมูลด้วยโมเด็มอย่าง เช่น การบีบขนาดข้อมูล(Data Comprtssion) หรือการควบคุมข้อผิดพลาดของข้อมูล(Error Control) เข้ามาช่วยทำให้ความเร็วในการรับส่งข้อมูลทำได้เร็วขึ้น ด้วยเหตุที่โทรสารในกลุ่ม G3 มีความคล้ายคลึงกับการสื่อสารด้วยโมเด็มทั่วไป ทำให้การกำหนดมาตรฐานของโทรสารในกลุ่ม G3 เปลี่ยนไปใช้ข้อกำหนดในกลุ่ม V เช่นเดียวกับโมเด็ม โดยมีข้อกำหนดที่สำคัญ คือ V.27ter สำหรับการรับส่งข้อมูลที่ความเร็ว 2400 bps หรือที่ 4800 bps, V.29 ใช้ในการรับส่งข้อมูลที่ความเร็ว 7200 bps และที่ 9600 bps นอกจากนี้ยังกำหนดให้โทรสารในกลุ่ม G3 สามารถที่จะเลือกสื่อสารกันแบบ Half Duplex ที่ความเร็ว 7200, 9600, 12000 และ 14400 bps โดยใช้ข้อกำหนด V.17 หรือที่นิยมเรียกกันว่า V.FAX หรือเลือกใช้การสื่อสารแบบ Full Duplex ที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความเร็ว 1200 และ 14400 bps โดยใช้ข้อกำหนด V.33 และโทรสารในกลุ่มนี้สามารถส่งเอกสารขนาด A4 ได้ในเวลาเพียง 30 วินาที ถึง 1 นาที

2.1.1 ขั้นตอนการรับส่งข้อมูลของเครื่องโทรสารกลุ่ม 3

ตามข้อกำหนดของ CCITT มาตรฐาน T.30 ได้แบ่งขั้นตอนในการรับส่งข้อมูลของเครื่องโทรสารกลุ่มที่ 3 ออกเป็น 5 เฟส คือ

Phase A: Call establishment

Phase B : Pre-message procedure

Phase C : Message transmission

Phase D : Post-message procedure

Phase E : Call release

Phase A

เป็นการเริ่มต้นการเรียก เป็นขั้นตอนที่จะเชื่อมโยงเครื่องโทรสารต้นทางเข้ากับเครื่องโทรสารปลายทาง โดยผ่านข่ายทางสายโทรศัพท์ อาจเป็นแบบควบคุมโดยโอเปอเรเตอร์ หรืออาจเป็นแบบอัตโนมัติในขั้นแรกเครื่องโทรสารทางด้านผู้เรียกจะทำการเรียกไปยังเครื่องโทรสารปลายทางที่ต้องการติดต่อด้วยสัญญาณ Dial tone และส่งสัญญาณ CNG (calling tone)ไป เพื่อเริ่มการติดต่อ ส่วนทางด้านถูกเรียกจะตอบสนองต่อการเรียกนั้นโดยส่งสัญญาณ CED (called station identification) ซึ่งเป็นสัญญาณความถี่ 2,100 Hz กลับไปยังเครื่องโทรสารด้านผู้เรียกทุกๆ 3 วินาที

Phase B

เป็นขั้นตอนที่เครื่องโทรสารต้นทางเชื่อมโยงกับเครื่องโทรสารปลายทางแล้วต้องทำการตรวจสอบคุณสมบัติและความสามารถในการรับและส่งให้ตรงกัน ก่อนที่จะเริ่มส่งข่าวสาร มี 2 ขั้นตอนคือ

1>. ส่วนการตรวจสอบสัญญาณ ทำการส่งสัญญาณเพื่อบอกให้ทราบว่าเป็นเครื่องโทรสารในกลุ่มใด (group identification), พร้อมที่จะรับข้อมูลหรือไม่ (confirmation for reception), หมายเลขโทรศัพท์ของเครื่องโทรสารต้นทาง (subscriber identification) รวมถึงความสามารถอื่นๆที่ไม่ใช้มาตรฐานของ CCITT (nonstandard facilities identification) เป็นทางเลือก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2>. ส่งสัญญาณเพื่อตรวจสอบระบบสื่อสาร คำสั่งเพื่อที่จะเป็นผู้รับหรือผู้ส่ง (group command), ตรวจสอบระบบสื่อสัญญาณ (phasing training) และการซิงโครไนซ์ (synchronization)

โดยเครื่องโทรสารทางด้านถูกเรียกจะส่งสัญญาณ DIS (digital identification signal) ไปยังด้านผู้เรียกเพื่อแสดงถึงคุณสมบัติและความสามารถต่างๆของเครื่อง ซึ่งเมื่อทางด้านผู้เรียกได้รับสัญญาณนี้แล้วก็จะส่งสัญญาณ DCS (digital command dignal) มายังด้านถูกเรียกเพื่อแจ้งให้ทราบว่าได้เลือกใช้คุณสมบัติและความสามารถใดของเครื่องที่ถูกเรียกเป็นข้อกำหนดในการรับส่งข้อมูลครั้งนี้ ซึ่งเครื่องโทรสารด้านถูกเรียกก็จะต้องเลือกโหมดของตนเองให้เป็นไปตามข้อกำหนดนั้น จากนั้นเครื่องโทรสารด้านผู้เรียกจะส่งสัญญาณความเร็วสูงที่เรียก training เพื่อตรวจสอบระบบสื่อสาร ซึ่งหลังจากเครื่องโทรสารด้านถูกเรียกได้รับสัญญาณ training และตรวจสอบเรียบร้อยแล้ว ก็จะส่งสัญญาณ CFR (comfirmation to receive) กลับไปยังเครื่องโทรสารต้นทางเพื่อยืนยันความพร้อมในการรับข้อมูล

Phase C

เป็นขั้นตอนการส่งข้อมูลประกอบไปด้วย

เฟส C1 ส่งสัญญาณควบคุมข่าวสาร เป็นขั้นตอนส่งสัญญาณควบคุมเพื่อการซิงโครไนซ์ การตรวจสอบข้อผิดพลาด การแก้ไขข้อผิดพลาด และตรวจสอบสถานะของระบบสื่อสารในขณะที่กำลังส่งข่าวสาร

เฟส C2 เป็นขั้นตอนการส่งข่าวสาร โดยมีรูปแบบตามแต่ชนิดของเครื่องโทรสารแต่ละกลุ่ม

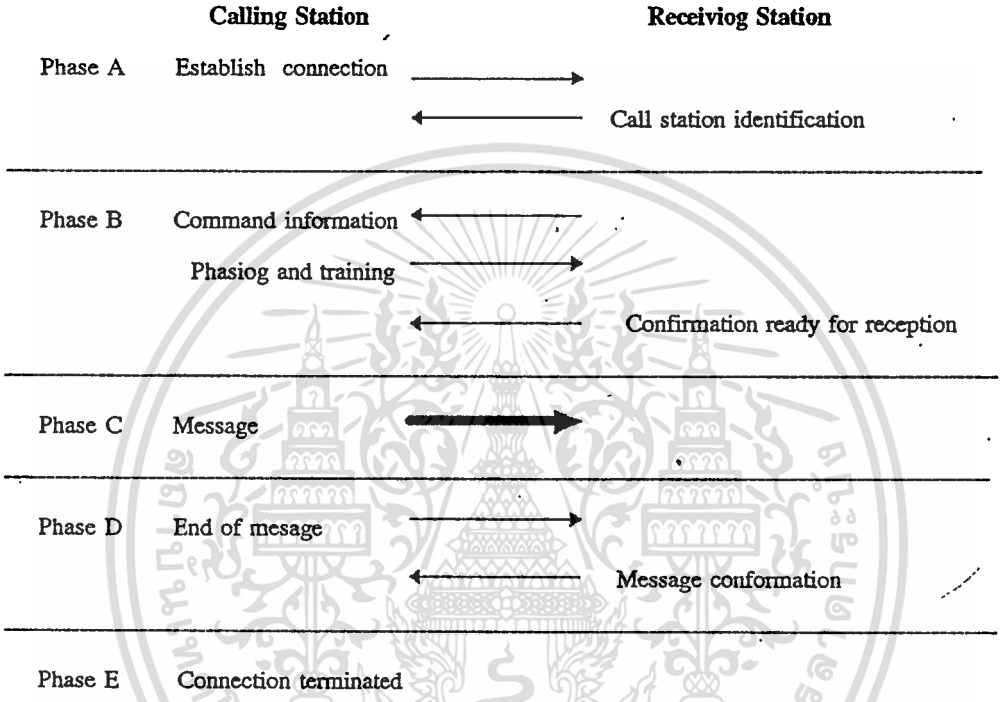
Phase D

เป็นขั้นตอนหลังการส่งข่าวสาร โดยเครื่องโทรสารด้านผู้เรียกจะส่งสัญญาณ RTC (return to control) ไปยังด้านถูกเรียกเพื่อทำการปรับโมเด็มให้กลับไปอยู่ที่ 300 บิท/วินาที จากนั้นจึงจะส่งสัญญาณ EOP (end of procedure) ซึ่งทางด้านถูกเรียกก็จะส่งสัญญาณ MCF (message confirmation) กลับมาให้ทางด้านต้นทางเพื่อยืนยันการรับข้อมูลที่ถูกต้องเรียบร้อย ในกรณีที่มีการส่งข้อมูลหลายๆ หน้า เมื่อจบหน้าก็จะมีการส่งสัญญาณเพื่อแสดงว่ามีข้อมูลที่จะส่งต่อไปอีก แทนการส่งสัญญาณ EOP เช่น EOM (end of message) หรือ MPS (multi page signal)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Phase E

เป็นขั้นตอนการเลิกการติดต่อระหว่างเครื่องโทรสารต้นทาง และเครื่องโทรสารปลายทาง อาจเป็นแบบ manual หรือ automatic ก็ได้ โดยเครื่องโทรสารด้านผู้เรียกจะส่งสัญญาณ DCN (disconnect) ไปยังด้านปลายทางเพื่อยกเลิกการติดต่อ



2.2 เครื่อง โทรสาร Modems

ถ้าจะมองทางด้านรูปลักษณ์ภายนอกที่มองเห็น เราไม่สามารถแยกความแตกต่างระหว่างโทรสารโมเด็มและดาต้าโมเด็มออกจากกันได้ และวิธีการมอดดูเลทหรือวิธีการตรวจสอบความผิดพลาดที่ใช้ในดาต้าโมเด็มก็คล้ายๆกับที่ใช้ในโทรสารโมเด็ม ข้อแตกต่างที่เด่นชัดระหว่างโมเด็มที่ใช้ส่งข้อมูลทั่วไปกับโทรสารโมเด็ม คือโปรโตคอล (protocol) ซึ่งควบคุมการติดต่อสื่อสารโทรสารโมเด็มจะต้องสนับสนุนโปรโตคอลที่กำหนดไว้ใน T.4, T.6 และ T.30 ของ CCITT กล่าวคือโทรสารโมเด็มจะต้องจัดหาคำสั่งใหม่ๆ และตอบสนองต่อการเริ่มต้นที่นำไปสู่การส่งโทรสาร

โมเด็มที่ใช้ในเครื่องโทรสารกลุ่ม 3 ได้กำหนดไว้ตามมาตราฐานของ ดังตารางต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Bits per Second	Baud Rate	Bits per Sample	Type	Carrier Frequency	Bandwidth in He
14400	2400	6	V.17	1800	550-3050
12000	2400	5	V.17	1800	550-3050
9600	2400	4	V.29	1700	450-2950
7200	2400	3	V.29	1700	450-2950
4800	1600	3	V.27 ter	1800	950-2650
2400	1200	2	V.27 ter	1800	1150-2450

ตารางมาตราฐานของโมเด็มที่ใช้ในเครื่องโทรสารกลุ่ม 3 ตามข้อกำหนดของ CCITT

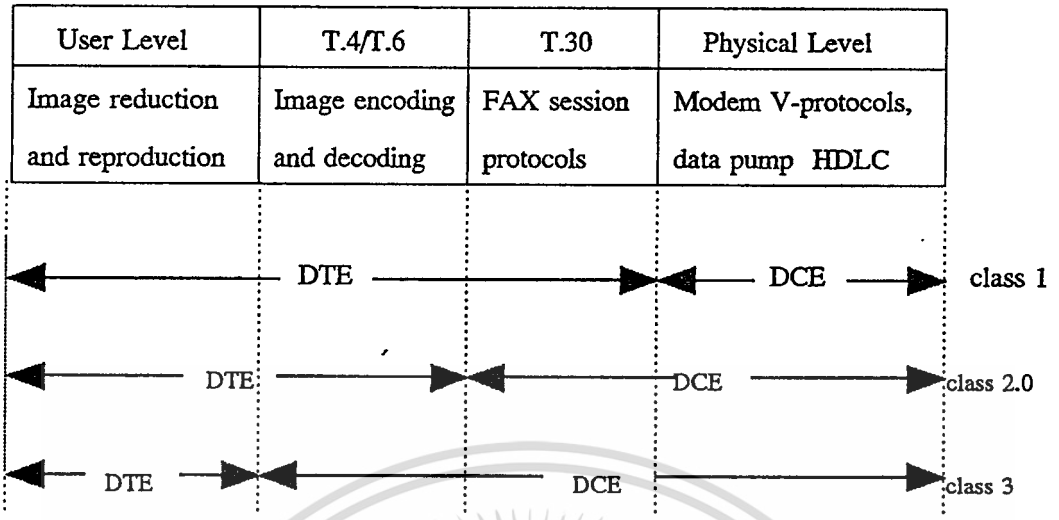
โมเด็มที่ใช้ในเครื่องโทรสารกลุ่ม 3 จะตรวจสอบการติดต่อระหว่างเครื่องโทรสารทั้ง 2 ด้านก่อนการส่งข้อมูล และโมเด็มจะพยายามส่งข้อมูลด้วยความเร็วสูงสุดเท่าที่จะเป็นไปได้ ส่วนใหญ่โมเด็มในเครื่องโทรสารได้ถูกกำหนดไว้ตามมาตรฐาน V.27ter และ V.29 แต่จะมีเครื่องโทรสารรุ่นใหม่ๆ บางเครื่องที่จะใช้มาตรฐาน V.33 โดย V.27ter เป็นมาตรฐานเกี่ยวกับโมเด็มที่ส่งข้อมูลด้วยความเร็ว 4800/2400 บิท/วินาที ผ่านข่ายสายโทรศัพท์ โดยการมอดดูเลทแบบเลื่อนเฟส โดยมีความถี่พาห์ที่ 1800+1 เฮิรตซ์ ส่วน V.29 เป็นข้อแนะนำในการส่งแบบ 9600 บิท/นาที โดยความถี่พาห์เป็น 1700+1 เฮิรตซ์ และ V.33 จะส่งข้อมูลด้วยความเร็วสูงสุด 14400 บิท/วินาที มอดูเลทแบบ QAM (TCM)

โดยส่วนใหญ่ในการส่งข้อมูลของเครื่องโทรสารกลุ่ม 3 นั้น จะเริ่มต้นที่ความเร็ว 9600 บิท/วินาที แต่ถ้าหากไม่สามารถส่งข้อมูลด้วยความเร็วนี้ได้ โมเด็มจะลดความเร็วในการส่งลงเป็น 7200, 4800 หรือ 2400 บิท/วินาที แทนตามความเหมาะสม

<u>การแบ่งกลุ่มโมเด็มโดย EIA</u>

ความสามารถของเครื่องโทรสารที่เพิ่มขึ้นจากโมเด็มปกติ ทำให้จำเป็นต้องมีคำสั่งใหม่ๆ ในการติดต่อ แต่ไม่มีบริษัทใดๆ ที่เป็นผู้กำหนดมาตรฐานนี้ ดังนั้นโทรสารโมเด็มจึงถูกพัฒนาโดยบริษัทต่างๆ แตกต่างกันออกไป ทำให้ไม่สามารถติดต่อเชื่อมโยงกันได้ เพื่อแก้ปัญหาดังกล่าว EIA จึงได้พัฒนามาตรฐานซึ่งกำหนดขั้นตอนและคำสั่งที่ใช้ระหว่าง DTE (คอมพิวเตอร์) และ DCE (โมเด็ม)ขึ้น โดยแบ่งโมเด็มออกเป็น 3 กลุ่มตามความสามารถของโมเด็มในการนำไปสู่การส่งโทรสารดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปแสดงการแบ่งกลุ่มโมเด็มโดย EIA

<u>โทรสารโมเด็มกลุ่มที่ 1</u>

EIA ได้กำหนดโทรสารโมเด็มกลุ่มที่ 1 ไว้ใน EIA/TIA-578 ซึ่งโมเด็มกลุ่มนี้จะหมายถึงโทรสารโมเด็มที่มีบริการเท่าที่จำเป็นต่อการรองรับขั้นตอนของโทรสารกลุ่ม 3 เท่านั้น คอมพิวเตอร์จะตอบสนองต่อการเข้ารหัสของภาพโดย T.4 และการจัดการในการส่งข้อมูลโดย T.30 โทรสารโมเด็มกลุ่มที่ 1 จะมีบริการต่างๆ ต่อไปนี้

- GSTN interface
- หมุนหมายเลขอัตโนมัติ
- V series signal conversion (modulation)
- การรับส่งข้อมูล
- การใช้รูปแบบ HDLC เฟรม, transparency และการตรวจสอบความผิดพลาด
- คำสั่งควบคุมและการตอบสนอง

การส่งโทรสารที่โมเด็มกลุ่ม 1 จะต้องใช้ภายใต้การควบคุมของซอฟต์แวร์ ต่างจากการส่งแบบใช้โมเด็มข้อมูล (data modem) ซึ่งการกำหนดเวลา, การเข้ารหัสและการลำดับ การส่งต้องอาศัย T.30 ทำให้ไม่สามารถควบคุมการส่งโทรสารโดยใช้คำสั่งจากผู้ควบคุมเครื่องได้

<u>โทรสารโมเด็มกลุ่มที่ 2.0</u>

โทรสารโมเด็มกลุ่มนี้มีความสามารถสูงกว่าโมเด็มกลุ่มแรก การแบ่งส่วนต่างๆก็จะเปลี่ยนไป ซึ่งขั้นตอนที่ยุ่งยากในการส่งโทรสารโดยใช้ T.30 จะถูกตัดออกจากคอมพิวเตอร์ไปเป็นหน้าที่ของโมเด็มแทน ซึ่งโมเด็มกลุ่มที่ 2.0 นี้มีบริการต่างๆต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- GSTN interface
- การหมุนหมายเลขอัตโนมัติ
- V series signal conversion
- T.30 protocol implemention
- การรายงานสภาวะการทำงาน
- การส่งข้อมูลเฟส C
- padding สำหรับเวลาในการสแกนที่น้อยที่สุด
- การตรวจสอบข้อมูลที่ได้รับ
- โปรโตคอลแบบแพคเก็ต (packet) สำหรับ DTE/DCE interface

โมเด็มกลุ่มที่ 2.0 จะเริ่มต้นการเรียก, จัดการขบวนการติดต่อสื่อสาร, ส่งข้อมูลภาพ และหน้าที่อื่นๆระหว่างรูปแบบของ T.4 (เครื่องโทรสารกลุ่ม 3) และ T.6 (เครื่องโทรสารกลุ่ม 4) เครื่องคอมพิวเตอร์จะทำหน้าที่ในการจัดเตรียมและลดขนาดข้อมูลภาพสำหรับการสื่อสารและตีความข้อมูลที่ถูกลดขนาดมาแล้วในทางด้านรับ

โทรสารโมเด็มกลุ่มที่ 3

การกำหนดมาตราฐานของโทรสารโมเด็มกลุ่มที่ 3 ยังอยู่ในระหว่างการศึกษา แต่มีแนวโน้มในการโอนหน้าที่ต่างๆ ในขบวนการส่งโทรสารจากเครื่องคอมพิวเตอร์ไปให้กับโมเด็ม จากมาตรฐาน T.30 และหน้าที่ทั่วไปของโมเด็มในการส่งข้อมูล โมเด็มกลุ่มที่ 3 จะเปลี่ยนข้อมูลภาพให้เป็นภาพที่ถูกลดขนาดตามมาตรฐาน T.4 และ T.6 ซึ่งโมเด็มอาจจะเป็นตัวขยายภาพทางด้านรับด้วยก็ได้

กลุ่มคำสั่งที่ใช้ควบคุมโมเด็ม

คำสั่งสำหรับควบคุมโทรสารโมเด็มกลุ่มที่ 1 และ 2.0 โดย EIA ถูกออกแบบเพิ่มเติมจากกลุ่มคำสั่ง AT ซึ่งสายของอักขระที่ถูกส่งไปยังโมเด็มจะเรียกว่า แถวคำสั่ง (command line) และจะต้องขึ้นต้นด้วยอักษร AT หรือ at แถวคำสั่งประกอบด้วยสัญลักษณ์ใน ASCII และลงท้ายด้วยเครื่องหมายแสดงการสิ้นสุดคำสั่งที่เรียกว่า carriage return

มีเพียง low-order 7 บิท ของแต่ละอักษรเท่านั้นที่จะถูกตรวจสอบในขณะที่โมเด็มแปลความหมายของคำสั่ง คำสั่งที่เขียนด้วยอักษรตัวใหญ่หรือตัวเล็กจะมีความหมายเหมือนกัน ช่องว่างและเครื่องหมายควบคุมอื่นๆที่นอกเหนือไปจาก ASCII 13 (carriage return) และ ASCII 8 (ช่องว่าง) ที่ปรากฏในคำสั่งจะถูกละเลยไป และจะถูกแสดงค่าด้วยศูนย์ ด้วยเหตุนี้โทรสารโมเด็มทุกเครื่องจึงต้องมี XON/XOFF flow control และอาจมี flow control ชนิดอื่นๆอีกก็

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ได้ รูปแบบตามข้อกำหนดของ EIA แต่ละคำสั่งจะขึ้นต้นคำสั่งด้วย +F เสมอ รูปแบบของคำสั่งมี 3 แบบ

-คำสั่งชุดที่ 1 คือ คำสั่งตามการใช้งาน ได้แก่ ความสามารถ (capabilities), สถานะ แบบดังนี้

+F *command* = ?

command จะแสดงคำสั่งโทร ซึ่ง สารที่ใช้งานได้ โมเด็มจะตอบสนองโดยการเลือกค่าหรือช่วงของค่าที่มันสามารถรับได้ เช่นโมเด็มอาจอาจจะตอบสนองต่อแถวคำสั่งรูปแบบ AT+FCLASS=? (เป็นการบอกว่าการติดต่อสื่อสารในครั้งนี้ใช้กลุ่มใด) ได้ค่า 0, 1, 2 ผลตอบสนองนี้แสดงให้เห็นว่าโมเด็มสามารถถูกกำหนดให้เป็นกลุ่มที่ 1 และที่ 2 ได้

-คำสั่งชุดที่ 2 คือ คำสั่งเกี่ยวกับสถานะ จะเป็นการกำหนดค่าขององค์ประกอบหรือทางเลือกในโครงสร้าง

คำสั่งนี้มีรูปแบบดังนี้คือ

+F*command*?

ตัวอย่างการกำหนดรูปแบบการทำงานเช่น กำหนดคำสั่งว่า AT+FCLASS? โมเด็มซึ่งทำงานเหมือนเป็นโทรสารโมเด็มกลุ่มที่ 2 จะตอบสนองต่อคำสั่งด้วย 2

- คำสั่งชุดที่ 3 ใช้สำหรับกำหนดค่าขององค์ประกอบหรือผ่านค่าองค์ประกอบที่ใช้ควบคุมการทำงานของโมเด็ม มีรูปแบบบดังนี้

+F*command=val*

ซึ่ง *command* จะแสดงถึงองค์ประกอบที่ต้องการกำหนดและ *val* แสดงถึงค่าขององค์ประกอบที่ต้องการ ทั้งนี้ขึ้นอยู่กับแต่ละคำสั่ง *val* อาจจะเป็นตัวเลขหรือตัวอักษรก็ได้ ซึ่งการกำหนดเป็นตัวเลขทั้งในคำสั่งมาตรฐาน AT data modem และในคำสั่งโทรสารกลุ่ม 1 จะแสดงด้วยเลขฐานสิบ

สรุปคำสั่งที่ใช้สำหรับโมเด็ม

คำสั่งที่ใช้ควบคุมโมเด็มในที่นี้จะขอกล่าวถึงเฉพาะคำสั่งสำหรับโทรสารโมเด็มกลุ่มที่2 เพราะโครงงานนี้ได้ใช้คำสั่งในกลุ่มที่2 ซึ่งสามารถดูรูปแบบคำสั่งต่างๆ ที่ใช้ได้จากภาคผนวก ก

สำหรับ กลุ่มที่ 2 คำสั่งที่ใช้สำหรับกำหนดค่าขององค์ประกอบหรือผ่านองค์ประกอบที่ใช้ในการควบคุมการทำงานของโมเด็มอาจจะใช้ได้ค่ากับค่าเดียวหรือหลายค่าก็ได้ ค่าคงที่ในกลุ่มที่ 2.0 จะต้องเป็นเลขฐานสิบหก ค่าคงที่ต่างๆนี้ สร้างขึ้นมาจาก 0 ถึง 9 (ใน ASCII คือ 30h ถึง 39h) และ A ถึง F (ใน ASCII คือ 41h ถึง 46h)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

035029

สำหรับการกำหนดค่าที่มีหลายๆค่า แต่ละค่าจะถูกแยกด้วยเครื่องหมาย " ," เช่น ผลตอบสนองคือ (0,2,4,8) ส่วนการกำหนดค่าเป็นช่วงจะแสดงด้วยเครื่องหมาย " -" เช่น กำหนดค่าอยู่ในช่วง 0-255 (ฐานสิบ) สามารถเขียนได้ว่า 0-FF และคำสั่งในกลุ่มที่ 2 ยังยอมรับการกำหนดค่าหลายแบบผสมกัน ประกอบด้วย ค่าที่เป็นลำดับซึ่งแต่ละค่าจะอยู่ในวงเล็บและแบ่งด้วย ", " โดยจะไม่สนใจช่องว่างระหว่างค่าเช่น (0, 1, 2,), (0), (0,-3)

คำสั่งกลุ่ม 2 จะถูกทำจากซ้ายไปขวาและแต่ละคำสั่งจะถูกแยกจากกัน โดยไม่สนใจสิ่งที่ตามหลังแถวคำสั่ง ถ้าทุกคำสั่งสามารถทำได้ถูกต้อง จะแสดงด้วยรหัสที่บอกสถานะของคำสั่งสุดท้ายในบรรทัด แต่ถ้าเกิดความผิดพลาดขึ้นหรือพบคำสั่งที่ไม่ถูกต้องตามข้อกำหนด การประมวลผลคำสั่งจะหยุดลงและคำสั่งอื่นๆ ที่ยังไม่ได้ทำจะยกเลิกไป รหัสแสดงผลของโทรสารโมเด็มกลุ่มที่ 2 มีดังนี้คือ

0 : OK

1 : CONNECT

2 : RING

3 : NO CARRIER

4 : ERROR

5 : NO DIALTONE

7 : BUSY

8 : NO ANSWER

การพัฒนาข้อกำหนดของโทรสารโมเด็มกลุ่ม 2.0 ถูกนำโดยอนุกรรมการ 29.2 ในการเชื่อมต่อของเครื่องโทรสารระบบดิจิตอล ในเดือนสิงหาคม 1990 อนุกรรมการดังกล่าวได้ออก -2388- โดยร่างชุดแรกเกี่ยวกับการนำไปสู่มาตราฐานโทรสารกลุ่ม 2.0 แต่ต่อมาได้มีการออก มาแทน ซึ่งมีบางส่วนที่ยังคงยึดถือตามมาตราฐานโมเด็มกลุ่มที่ 2 ที่ออกในเดือนสิงหาคม 1990 โดยคำสั่งเกี่ยวกับการจัดกลุ่มจะใช้ดังต่อไปนี้

0 : data modem

1 : EIA/TIA - 578

2 : SP-2388-A

2.0 : TIA/EIA-592

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 การเข้ารหัส

วิธีการเข้ารหัสของสัญญาณโดยใช้การเข้ารหัสแบบ Huffman ที่จะส่งไปยังเครื่องโทรสารปลายทางทำได้ 2 วิธีคือ

One-dimensional codiog scheme

การเข้ารหัสแบบ One-dimensional ใช้หลักการเข้ารหัสแบบ MH (Modified Huffman) ข้อมูลที่มีค่าความน่าจะเป็นสูง(เกิดขึ้นบ่อย) จะถูกแทนด้วยรหัสที่มีความยาวน้อย ส่วนข้อมูลที่มีค่าความน่าจะเป็นต่ำจะถูกแทนด้วยรหัสที่มีความยาวมาก ซึ่งมาตรฐาน CCITT ได้กำหนดรหัสที่ใช้เข้ารหัสไว้แล้ว

ในการเข้ารหัสจะเริ่มต้นด้วยจุดขาวก่อนเสมอ กรณีที่จุดแรกเป็นจุดดำ รหัสของจุดขาวที่มีความยาวเท่ากับ 0 จะถูกส่งออกไป ข้อมูลจุดขาวหรือจุดดำสามารถมีความยาวได้ถึง 1728 พิคเซล ซึ่งเป็นความยาวสูงสุดสำหรับหนึ่งเส้นสแกนมาตรฐาน รหัสที่ใช้แทนข้อมูนี้มี 2 ชนิด คือ

1. เทอร์มิเนตติ้งโค้ด (Terminating Code) ใช้แทนค่าพิคเซลที่มีความยาวตั้งแต่ 0 ถึง 63
2. เมคอัพโค้ด (Make-up Code) ใช้แทนค่าพิคเซลที่มีความยาวเป็นจำนวนเท่าของ 64 ถึง 1728

กรณีของพิคเซลซึ่งมีความยาวตั้งแต่ 64 - 1728 จะแทนด้วยเมคอัพโค้ดซึ่งมีค่าเท่ากับหรือน้อยกว่าความยาวของพิคเซลก่อน โดยที่ความยาวของพิคเซลแตกต่างจากค่าที่แทนโดยเมคอัพโค้ดไม่เกิน 63 จากนั้นก็แทนค่าที่เหลือด้วยเทอร์มิเนตติ้งโค้ด

ตัวอย่างเช่น ถ้าพิคเซลซึ่งเป็นจุดขาวมีความยาวเท่ากับ 500 จะประกอบไปด้วยเมคุอัพโค้ดความยาว 448 (0110 0100) และตามด้วยเทอร์มิเนตติ้งโค้ดความยาว 52 (0101 0101) และในกรณีที่จำนวนพิคเซลมีมากกว่า 1728 จุด ซึ่งใช้เมื่อขนาดของกระดาษใหญ่กว่า A4 สามารถแทนได้ด้วยรหัสในตาราง

ตัวอย่างการเข้ารหัสแบบโมดิฟายด์ ฮัฟฟ์แมน แสดงดังรูปต่อไปนี้

1 3 102 656 951 15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Color	Run Length	Makeup Codeword	Terminating Codeword
white	1	-	00 0111
black	1	-	10
while	102	1 1011	0001 0111
black	656	0 0000 0100 1010	00 0000 0111
while	951	0 1101 0011	0101 1000
black	15	-	0 0001 1000

รูป แสดงการเข้ารหัสแบบ Modified Huffman

จุดสิ้นสุดของเส้นสแกน (End-of-line:EOL)

เมื่อสิ้นสุดการสแกนข้อมูลครบ 1 เส้น ข้อมูลจะต้องตามด้วยรหัส EOL เพื่อแสดงจุดสิ้นสุดของเส้นสแกน นอกจากนี้รหัส EOL ยังใช้นำหน้าเส้นสแกนเส้นแรกของแต่ละหน้าด้วย

Format : 000000000001

บิทเต็ม (Fill)

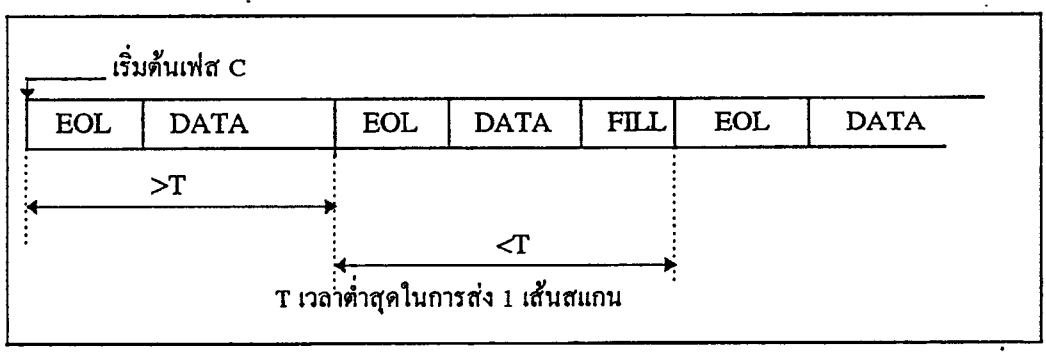
Fill นี้จะอยู่ระหว่างข้อมูลกับ EOL ในแต่ละ line, Fill จะถูกรวมเป็นเวลาในการส่งข้อมูล โดย Fill และ EOL จะต้องไม่น้อยกว่าเวลา ที่น้อยที่สุดในการส่งของ code รวมตลอดทั้งline

Format : variable length string of 0s.

กลับคืนสู่การควบคุม (Return to control, RTC)

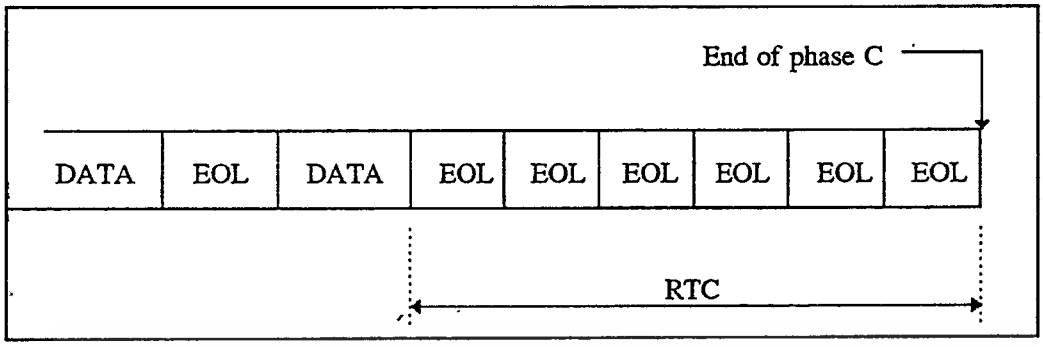
การจบของการส่งข้อมูลแต่ละแผ่นจะต้องตามด้วย RTC เพื่อเป็นการกลับเข้าสู่โหมดการควบคุม โดยการส่ง EOLs ตามๆ กันมาหกครั้ง

Format : 000000000001.....000000000001 (total of 6 times)



รูปแสดงเส้นสแกนหลายเส้นที่จุดเริ่มต้นของหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปแสดง Return to control

Two-Dimension coding schema

นอกจากการเข้ารหัส แบบ one-dimension แล้ว T.4 ยังได้มีการอธิบายถึงการเข้ารหัสแบบ Two-Dimension ซึ่งมีวิธีการเข้ารหัสแบบ 2 dimension จะเหมือนกับ one-dimension แต่ในแต่ละ line จะมีการเข้ารหัสให้สัมพันธ์กันด้วย โดยใน scan line จะเก็บเฉพาะส่วนที่แตกต่างจาก line ก่อนหน้า ซึ่งจะทำให้การเข้ารหัสมีประสิทธิภาพมากขึ้น

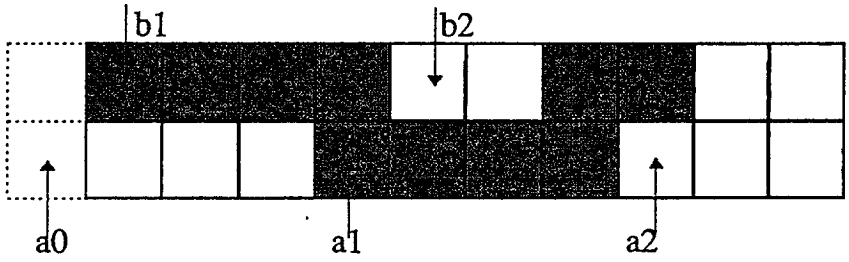
แน่นอนการเข้ารหัสแบบ Two-dimension นี้ อาจจะมี error เพิ่มมากขึ้นได้เพราะ error จากเพียง 1 scan line อาจจะทำให้ scan line อื่นๆเกิดการผิดพลาดได้ ดังนั้นการจำกัดขอบเขตของ error ในแต่ละ line จึงถูกกำหนดขึ้น(K) หรือพูดอีกนัยหนึ่งคือ หลังจาก ส่งการเข้ารหัสแบบ one dimension แล้ว ในที่สุด ก็จะส่ง การเข้ารหัสแบบ two-dimenson ซึ่งค่า K ที่ แนะนำให้ใช้มากที่สุด คือ 2 สำหรับ standard vertical resolution(3.85 lines per millimeter) และ 4 สำหรับความละเอียดสูง (7.7 lines per millimeter)

ขั้นตอนการเข้ารหัส Two-dimension

1. ใน line แรกให้เข้ารหัสแบบ one-dimension โดย line นี้จะใช้เป็น line อ้างอิงในการเข้ารหัส

2.ทำการส่ง next line (enconding lines) โดยที่ encoding line จะถูกเปรียบเทียบกับบรรทัดอ้างอิง ทุกๆ จุด (pixel) เพื่อแยกความแตกต่างของการเปลี่ยนแปลงของ pixcel ในบรรทัด (transition piel)โดยการเข้ารหัสของทุก line จะถูกสมมุติให้ ตำแหน่งของ pixel จะเป็นสีขาวนำหน้า pixel อันจริง ดังรูป Definition of transition pixels for two-dimension encoding

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



a0 = Anchor pixel on the encoding line

a1 = First pixel on the encoding line after a0 of opposite color to a0

a2 = First pixel on the encoding line after a1 of opposite color to a1

b1 = First pixel on the reference line after a0 of opposite color to a0

b2 = First pixel on the reference line after b2 of opposite color to b2

รูป Definition of transition pixels for two-dimension encoding

a0 เป็น pixel หลักในการเข้ารหัส โดย a0 จะถูกกำหนดค่าเริ่มต้นเป็น pixel สีขาวที่จุดเริ่มต้นของ line ที่จะเข้ารหัสซึ่งตำแหน่ง a0 จะขึ้นอยู่กับ algorithm ของการเข้ารหัส

a1 เป็นจุดเปลี่ยนแปลงแรกทางขวาของ a0 บน encoding line

a2 เป็นจุดเปลี่ยนแปลงแรกทางขวาของ a1 บน encoding line

b1 เป็นจุดเปลี่ยนแปลงของ pixel บน referrence line โดยจะต้องอยู่ทางขวาของ a0และมีสีตรงข้าม

b2 เป็นจุดเปลี่ยนแปลงแรกของ pixel บน reference line ซึ่งอยู่ทางขวาของ b1

3. ทำการกำหนด mode ของการเข้ารหัสซึ่งจะมีความสัมพันธ์ กับตำแหน่งของการเปลี่ยนแปลงของ pixel ที่มีการเปลี่ยนแปลง ซึ่งจะมีอยู่ 3 mode ด้วยกันคือ

- Pass mode
- Vertical mode
- Horizontal mode

Pass mode

ถ้า b2 ปรากฏอยู่ด้านซ้ายของ a1 แล้วก็ให้ส่งcard word 0001 ไปทางฝ่ายรับซึ่งจะแปลงคำสั่งโดยจะทำการ fill ข้อมูลโดยมีสีเหมือนกับ a0 โดยเริ่ม fill ตั้งแต่ตำแหน่งของ a0 จนถึงตำแหน่งก่อน b2 แล้วทำการเปลี่ยนตำแหน่งของ a0 ไปใต้ของ b2 แล้วก็ทำตาม algorithm ต่อไป ดังรูป pass mode

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Vertical mode

เมื่อตรวจสอบแล้วปรากฏว่าไม่ใช่ pass mode แล้ว ถ้าตำแหน่งของ a1 อยู่ภายใน 3 pixcel ทางด้านซ้าย หรือ ขวาของ b1 แล้วแสดงว่าการเข้ารหัสอยู่ใน vertical mode ซึ่งแสดงดังรูป vertical mode โดยความสัมพันธ์ระหว่างระยะห่างของ a1 กับ b1 สามารถ แสดงได้ดัง table เมื่อได้ค่า code word แล้ว ก็จะทำการส่งไปที่ฝ่ายรับ และทำการเปลี่ยนจุด a0 ไปที่ a1 โดยที่ฝ่ายรับ สีของ pixel ของ a0 จะถูกเขียนตั้งแต่ตำแหน่งของ a0 จนถึงตำแหน่งที่ให้ค่าโดย code word

Horzontal mode

เมื่อตรวจสอบแล้วปรากฏว่าไม่ใช่ pass mode แล้ว ถ้าตำแหน่งของ a1 มีระยะห่างจาก b1 มากกว่า 3 pixcel การเข้ารหัสจะอยู่ใน horizontal mode โดยอธิบายได้ดังรูป horizontal mode ซึ่งเมื่ออยู่ใน horizontal mode แล้วฝ่ายส่งจะส่ง horizontal mode codeword (0001) + one-dimension codeword สำหรับ a0-a1 และ a1-a2 และทำการเลื่อนตำแหน่งของ a0 ไปที่ตำแหน่งของ a2 เพื่อเตรียมทำงานต่อไป ที่ฝ่ายรับก็จะทำการพิมพ์ สี และ ความยาที่เหมาะสตามที่ได้เข้ารหัสไว้

2.4 การบริการจัดลำดับงาน(Queue Service)

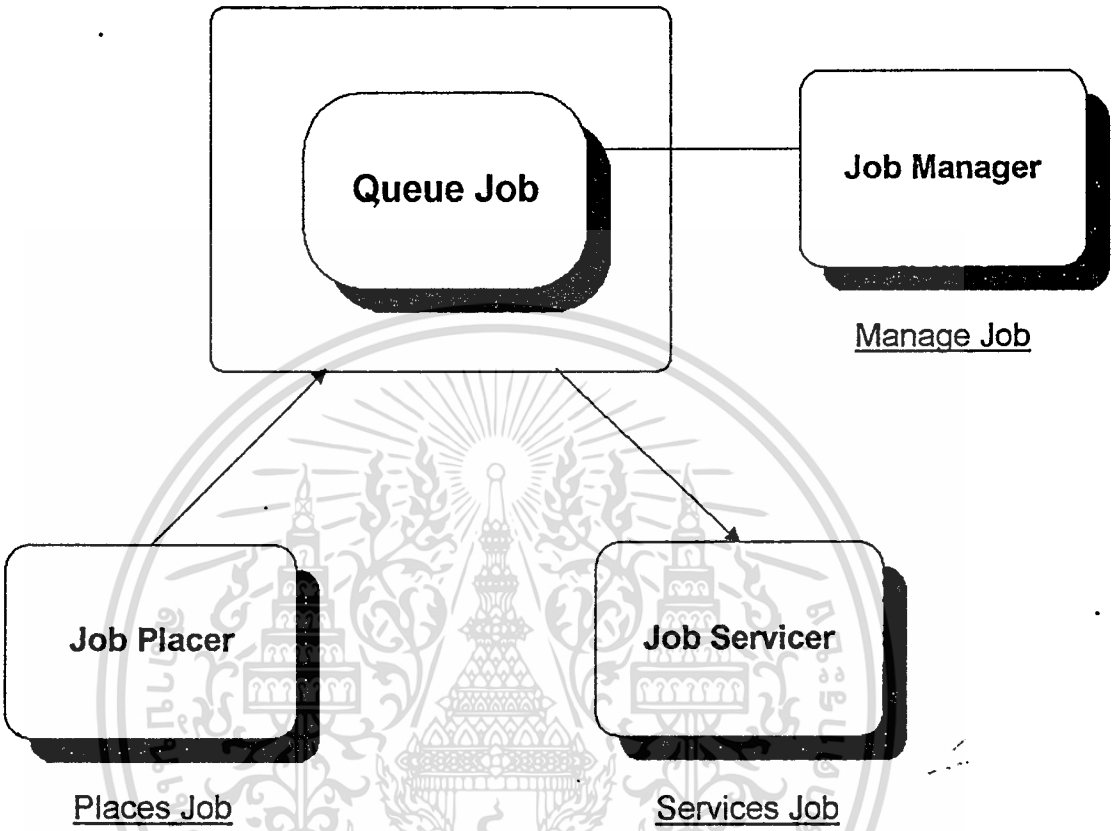
Queue Management System (QMS) เป็นระบบหนึ่งซึ่งเป็นประโยชน์มากในการใช้งาน QMS เป็นอยู่บน Netware 2.1x เพิ่มเติมจาก Netware 2.0 และ Version ที่ต่ำกว่า ซึ่ง QMS จะถูกติดตั้งรวมอยู่บน เครื่องที่ให้บริการ(Server)โดยจะจัดการ เรื่อง บริการด้านงานพิมพ์ (Service print job) จะเห็นว่า QMS มีความเป็นทั่วไป(Generalize) ทำให้ programmer สามารถใช้ประโยชน์ของ QMS ที่มีอยู่ได้อย่างเต็มที่

การบริการนี้จะยอมให้ ผู้พัฒนาสามารถใช้ศักยภาพของ Netware 2.1x ในการกระจายงานการประมวลผล (processing) โดย workstation ที่มีสมรรถนะสูง สามารถที่จะกลายหรือเปลี่ยนมาเป็น ตัวให้บริการงาน(Job server) คอยการทำงาน ต่างๆเช่น batch processing, Compiling, การคัดลอกแฟ้มข้อมูล, งานพิมพ์ ในขณะเดียวกัน workstation เครื่องอื่นก็สามารถทำงานอื่นได้ ซึ่ง Application ดังกล่าวยอมให้ทำ ทรัพยากรของเครือข่าย (netware resource) ได้เกือบทั้งหมด

สำหรับหลักการทำงานของ คิวเซอร์วิส จะมีองค์ประกอบที่สำคัญอยู่ 3 อย่างคือ

1. Workstation ที่ทำการใส่ งาน ลงไปใน คิว หรือ Workstation ที่จะทำงานนั้นๆ นั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป แสดงส่วนต่างๆ ที่เกี่ยวข้องกับ Queue Management

2.ส่วนจะอำนวยความสะดวกในบริหารการทำงานของคิว (ปรกติแม่ข่ายแฟ้มข้อมูลจะเป็นตัวที่ทำงาน ในส่วนนี้ แต่ workstation สามารถที่จะทำการ แก้ไขคุณสมบัติของ คิว ได้ ตย.ลำดับของ Job)

3.ส่วนบริการงานของ คิว จะเป็นส่วนที่จะนำ งาน ออกจาก คิว แล้ว Process งาน นั้นๆ

องค์ประกอบต่างๆ (ยกเว้นแม่ข่ายแฟ้มข้อมูล) จะต้องถูกพัฒนาโดย Program ซึ่งจะอยู่ในรูปของ Routine ต่างๆ เช่น Routine ในการใส่ งาน ลงใน คิว Routine ในการแก้ไข คิว และ Routine ที่จะ Remove และ Sevice

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คิวเปรียบเหมือนกับการเชื่อมโยงออปเจ็ค(Queue as Bindery Object)

คิว เปรียบเสมือนกับการเชื่อมโยง ออปเจ็ค เนื่องจาก คิว จะเป็นตัวเชื่อม Object ต่างๆ ด้วยตัวเอง ซึ่งจะกล่าวถึงเรื่องข้อกำหนดในการอ่านหรือเขียนของ Objectดังนั้น คิวออปเจ็ค จะมีคุณสมบัติทั่วไปในการกำหนดว่า ใครสามารถใช้ คิว ได้บ้าง (Who can use the Queue)ใครจะเป็นคนบริการ คิว(Who can service the Queue)และกำหนดใครสามารถทำอะไรได้บ้างได้อย่างแน่นอน แต่สิ่งสำคัญที่จะลืมไม่ได้คือ ขณะที่หน้าที่การทำงานของ คิว จะใช้ประโยชน์ของ Bindery Object แล้ว ผู้พัฒนาจะต้องระวังเรื่อง standard ของ Novell netware ในเรื่องการ Bindery Object(Novell's standards for Bindery Object) ดังจากเรื่อง Bindery จะมีการเตรียมมาตรฐานของ Bindery จนถึง 0x8000 เป็น Bindery Type ที่ได้มีการกำหนดไว้แล้ว ดังนั้นจึงเป็นเรื่องที่ผู้ที่จะพัฒนาได้มีการ create Bindery Object Type 9 ก็จะหมายถึง Arehive เซอร์ฟเวอร์ ซึ่ง Archive จะต้องมีคุณสมบัติและ protocal ที่สอดคล้องกับ Novell's Archive เซอร์ฟเวอร์ และเหมือนกับการใช้ คิว เพราะฉะนั้น Print Queue ก็จะมี type เป็น 3 และ General Job Queue ก็ควรจะเป็น type 10 แต่ถ้าผู้พัฒนาทำการ Create คิว ที่ไม่ใช้ predefine type คือใช้ type number มากกว่า 0x8000 (ถ้าเกิดต้องการใช้เป็นการค้าขาย Netware ก็สามารถ จะกำหนดค่าให้ได้

การเข้าใช้งาน คิวออปเจ็ค จะให้ทุกๆคนสามารถอ่าน และ Supervisor เขียน กล่าวคือ read Security นั้นจะยอมให้ทุกคนสามารถทำการตรวจ Bindery สำหรับ ชนิดของคิว ปรกติ ขณะที่ write Security จะป้องกัน คิว จากของทุกๆคนนอกจาก Supervisor หรือคนที่มีสิทธิเทียบ โดยที่แต่ละ คิว จะทำการกำหนดเหมือนกับ การเชื่อมโยง ออปเจ็ค(Bindery Object) ประกอบด้วย ชื่อเจ้าของ, ชนิด,IDnumber, ระดับของความปลอดภัย และคุณสมบัติ(properties) โดยที่คุณสมบัติจะช่วยในการกำหนดว่า คิว จะทำงานที่ไหน ใครสามารถจะใช้มันได้ ใครสามารถที่จะเปลี่ยนค่ามัน รวมทั้งใครที่สามารถ Service คิว ได้ ถึงแม้ว่าคุณสมบัติอื่น ผู้พัฒนาสามารถสร้างขึ้นมา เพื่อใช้งานได้เฉพาะเพียงแต่ คุณสมบัติมาตรฐาน (Standard properties) จะถูกใช้โดย QMS คือ Directory file ที่จะใส่ลงไปหรือใครสามารถใช้งาน, ควบคุมงาน, และทำการให้บริการงานใน คิว

คิวพร็อพเพอร์ตี้ (Queue Propertie)

โดย หลักๆที่ใช้โดย QMS คือ

Q_DIRECTORY พร็อพเพอร์ตี้

คิว จะต้องมี Directory ที่จะใช้เก็บแฟ้มระบบของ คิว และแฟ้มงานที่จะทำการ ประมวลผล Directory โดย SYS:SYSTEM มักจะถูกเลือกใช้เป็นส่วนใหญ่ แต่Directory อื่นๆก็สามารถใช้งานได้ ซึ่งชื่อ directory เต็มๆสามารถหาได้จากการต่อกันระหว่าง Path name ที่กำหนดกับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Queue's ID number ตัวอย่างเช่น Programmer กำหนด Path name Directory เป็น SYS:SYSTEM และ Object ID
ของ คิว คือ 04F3A เพราะฉะนั้น Directory ที่จะถูกใช้โดย QMS คือ "SYS:SYSTEM\4F3A" โดยที่จำนวนสูงสุดของ ตัวอักษรที่เป็นไปได้สำหรับคุณสมบัติอันนี้คือ 718ตัวอักษรรวมทั้ง Null terminator ด้วย สำหรับคุณสมบัติเป็นตัวโดดๆ จะมี 1 อันต่อ 1คิว ออปเจ็ค

Q_USERS พร็อพเพอร์ตี้

Q_USER เป็น พร็อพเพอร์ตี้ ที่สามารถกำหนดได้ ซึ่ง Properties นี้ จะถูกบรรจุด้วยรายการทั้งหมดของ Object ID ของ USER รวมทั้งกลุ่มที่สามารถใส่ งาน ลงไปใน คิว ได้ ซึ่ง QMS จะใช้ พร็อพเพอร์ตี้ นี้เป็นส่วนหนึ่งในกลไกรักษาความปลอดภัย ซึ่งจะแน่ใจได้ว่า USER มีสิทธิในการที่จะใส่ งาน ลงไปใน คิว ได้ ทุกๆ user ที่มีสิทธิเหมือนกับ user หรือกลุ่มใน list ของ พร็อพเพอร์ตี้ นี้ก็สามารถที่จะใส่ งาน ได้ ดังนั้นสมาชิกของกลุ่มที่อยู่ใน list ก็สามารถ access คิว ได้

ใน Novell, supervisor จะต้องอยู่ใน Properties นี้ด้วย ซึ่งจะมีคุณสมบัติเหมือนกับ user อื่นๆ แต่ supervisor สามารถที่จะเขียนข้อมูลลงไปใน คิว ได้ นอกจากนั้นกลุ่ม EVERYONE จะถูกเพิ่มเข้าไปเพื่อให้ทุกๆ user สามารถ place งาน ลงไปใน คิว ได้ เพราะฉะนั้นเมื่อ user ถูก remove ออกจาก logged-in objectก็จะถูก remove ออกจาก property ที่สามารถใส่งาน ลงไปใน คิว ได้ สำหรับ property นี้เป็นวิธีทางหนึ่งซึ่งมีความยืดหยุ่นมาก ในการนี้ผู้พัฒนาสามารถ จะใช้อำนวยความสะดวกในการรักษาความปลอดภัย

Q_SERVERS พร็อพเพอร์ตี้

ในกรณีที่ property Q_users กำหนดว่าใครสามารถส่ง งาน ลงไปใน คิว ได้ Q_SERVERS ก็จะเป็น property ที่บรรจุ object ID ของการเชื่อมโยงว่า Objectไหนสามารถที่จะเชื่อมต่อกับ คิว และทำการ Service job นั้นๆ

Q_SERVER property จะถูกอ่านเข้ามาเมื่อ Application เรียกใช้ Routine "Attach Server to Queue" เพื่อทำการเชื่อมโยง job service เข้ากับ คิว ซึ่งถ้าหาก Object ที่เรียกใช้เป็นสมาชิกของ (หรือ สิทธิเท่าเทียมกับสมาชิกใน property)
ก็จะเชื่อมโยงสำเร็จ แต่ถ้าไม่ใช่สมาชิก หรือไม่มีสิทธิเทียบเท่าสมาชิกใน property การเชื่อมโยงก็จะไม่สำเร็จ และจะส่งค่าความผิดพลาดกับออกมาเป็นโค้ด

โดยทั่วไปสมาชิกใน Q_SERVERS จะเป็น object ID ของผู้ใช้ที่สามารถเริ่ม งาน service Application ได้ โดยที่ Novell แนะนำว่าบางเหตุการณ์ มันสามารถช่วยในการสร้างแยกกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระหว่าง login entity กับสิทธิในการเริ่มการให้บริการงาน ซึ่งตรง object type นี้ สามารถที่จะช่วยได้มาก ถ้าหาก Application จะทำการตรวจหา Bindery object ทุกๆ เซอร์ฟเวอร์ บน Network หรือผู้ใช้สามารถที่จะกำหนด เซอร์ฟเวอร์
ที่แน่ใจว่าจะ service job ที่ place ลงไปใน คิว ได้หรือจะกำหนดให้ เซอร์ฟเวอร์ใดๆ
สามารถ service job ก็ได้

Q_OPERATORS พร็อพเพอร์ตี้

จะเหมือนกับ พร็อพเพอร์ตี้ ที่กล่าวมาแล้วคือ Q_OPERATORS จะเป็นการกำหนด คุณสมบัติของ คิว โดย คุณสมบัติ (Properties) นี้จะเก็บ รหัสประจำตัวของ ออปเจ็ค ที่สามารถดำเนินงาน(Operations) บนคิวได้ เช่นการเปลี่ยนแปลง file ในโครงสร้างของงาน (Job Entry Structure) หรือ เรียงลำดับงานใหม่บน คิว เป็นต้น เหมือนกับ รหัสประจำตัว ออปเจ็คของผู้ใช้งาน หรือ รหัสประจำตัวออปเจ็คของ กลุ่มสมาชิกที่สามารถเพิ่มเข้าไปใน Q_USERS พร็อพเพอร์ตี้ แม้แต่ Supervisor ก็ไม่ได้รับสิทธิพิเศษ (Privileges) ในการเป็น operator ของ คิว ดังนั้น รหัสประจำตัว ออปเจ็คของ Supervisor จึงต้องเพิ่มเติมเข้าไปใน Q_OPERATORS properties ด้วย โดยที่ properties ทั้งหมดนั้นจะถูกสร้างขึ้นตอนที่สร้าง คิว แต่ Q_DIRECTORY พร็อพเพอร์ตี้ จะถูกกำหนดค่าเริ่มต้นไว้เลย ในขณะที่ properties ที่เหลือสามารถทำการแก้ไข ผ่านทาง Bindery Service API และผู้พัฒนาจะต้องระวังว่าในการที่จะทำการสร้าง คิว และเปลี่ยนแปลง ผู้ใช้งาน ที่อยู่ใน List โปรแกรมประยุกต์จะต้องใช้สิทธิเป็น Supervisor หรือ ผู้ที่มีสิทธิเทียบเท่า

ลักษณะโครงสร้างของงาน (Job Entry Structure)

งานทุกงานที่ส่งไปยัง คิว จะต้องมีข้อมูลที่เกี่ยวกับ คิว ตรงกับลักษณะโครงสร้างของงาน (Job Entry Structure) ซึ่งในบาง Information ที่อยู่ในโครงสร้างจะ ถูกเตรียมให้บาง Information ที่อยู่ในโครงสร้างจะถูกเตรียมให้ โดย QMS (Queue Management System) ในขณะที่บาง information จะต้องถูกเตรียมขึ้นเองโดย Application ซึ่งในโครงสร้างที่ จะส่งเข้าไป จะไม่ถูกกำหนดตายตัว คือ ผู้ใช้งานอาจจะเป็นคนส่ง หรือ ผู้ใช้งาน บางคนที่เป็นสมาชิกใน Q_OPERATORS สามารถเปลี่ยนแปลง ลักษณะ งาน ที่ส่งเข้ามาได้ ซึ่งส่วนประกอบของงานที่จะส่งเข้าไปใน คิว จะเป็นดังนี้คือ

ClientStation	BYTE
ClientTaskNumber	BYTE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ClientIDNumber	LONG	hi - low
TargetServerIDNumber	LONG	hi - low
TargetExecutionTime	BYTE[6]	
JobEntryTime	BYTE[6]	
JobNumber	WORD	ho - low
JobType	WORD	hi - low
JobPosition	BYTE	
JobControlFlags	BYTE	
JobFileName	BYTE[14]	
JobFileHandel	BYTE[6]	
ServerStation	BYTE	
ServerTaskNumber	LONG	hi - low
TestJobDescription	BYTE[50]	
ClientRecordArea	BYTE[152]	

ข้อดีของ Queue Management System

1.จะได้ประโยชน์จากลักษณะของ Data Structure ของ คิว ในการกระจายการทำงาน ของงาน เพื่อลดปัญหาการใช้เวลาในการประมวลผล (Processing Problem) เนื่องจาก คิว เป็น Data Structure ที่เหมาะสมในการที่จะ สนับสนุนงานที่มีขนาดใหญ่ๆ จำนวนมากรวมทั้งงานที่ต้องการความเร็วสูงแต่ไม่ต้องการผลตอบสนอง แต่ Queue Mamagement จะไม่เหมาะสมกับงานที่ประเภท Real Time เนื่องจากงานในลักษณะนี้จะต้องการ การโต้ตอบแบบทันทีซึ่งในระบบ คิว ไม่สามารถทำได้ และ วิธีที่ดีกว่าและเหมาะสมกว่าคือไปใช้ แบบ Peer to Peer

2.สามารถที่จะควบคุมขนาดหรือขอบเขตของ คิว ได้ซึ่งในระบบ Peer to Peer ไม่ สามารถทำได้ เช่น ผู้ที่จะมาทำการ Service job สามารถที่จะเปลี่ยนลำดับของ คิว หรือ เปลี่ยนแปลง Priority ฯลฯ

3.เปิดโอกาสให้ผู้เขียน Program มีความยืดหยุ่นสูง กล่าวคือ สามารถที่จะมีกลวิธีในการควบคุมและใช้ คิว เองได้เพราะว่าส่วนหัวของ คิว งาน จะมีข้อมูลที่ตรงกันกับ ข้อมูลภายใน งาน นอกจากนี้ใน งาน ยังมีส่วนที่เตรียมไว้เพื่อความสะดวกอื่นอีกคือ ClientRecordArea อีก 152 byte ด้วย

นอกจากนี้แล้ว Queue Management System ยังสามารถที่จะทำ Security ในการใช้งานได้อีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

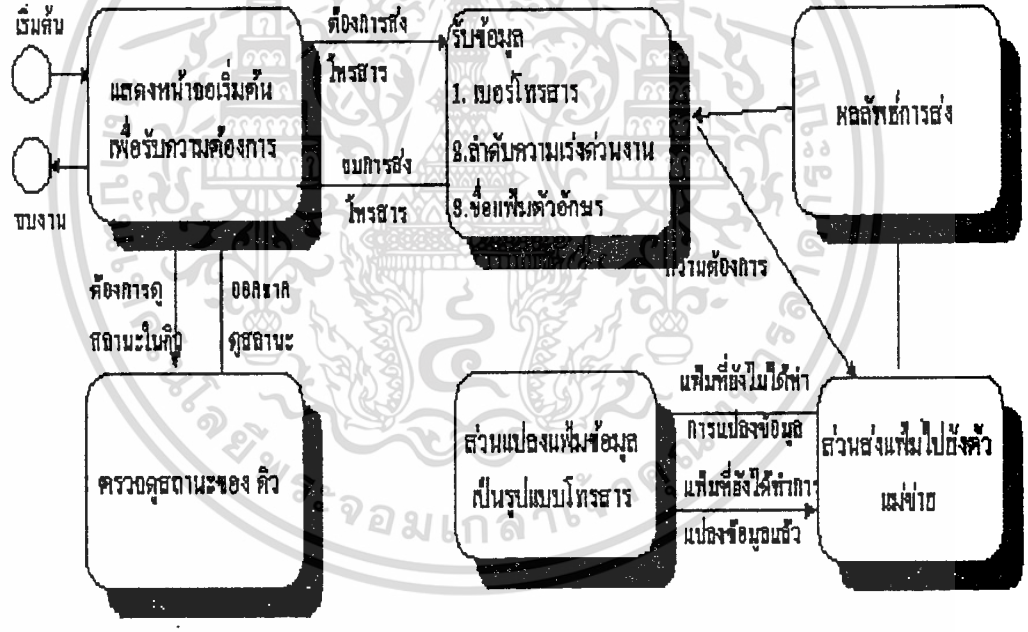
การออกแบบการทำงานของซอฟต์แวร์

ในการทำโครงงานนี้ได้แบ่งตัวซอฟต์แวร์ออกเป็น 2 ส่วนใหญ่ ๆ ด้วยกัน คือ ซอฟต์แวร์ที่อยู่ที่ทำงานเรื่องลูกข่ายกับซอฟต์แวร์ที่อยู่บนตัวแม่ข่าย ซึ่งอธิบายได้ดังนี้

3.1 ซอฟต์แวร์ที่ทำงานบนเครื่องของลูกข่าย ประกอบด้วย

3.1.1 ส่วนรับความต้องการของผู้ใช้

เป็นส่วนที่ใช้ติดต่อกับผู้ใช้ ซึ่งจะรับข้อมูลที่จะใช้ในระบบเข้ามาเพื่อส่งให้แบบแม่ข่าย โดยการทำงานจะอยู่บน โหมดตัวอักษร ดังแสดงขั้นตอนการทำงานได้ดังนี้



หลักการคือ เมื่อผู้ใชเรียกใช้โปรแกรมในส่วนนี้ก็จะมีการรับข้อมูลว่าต้องการส่งโทรสารหรือดูสถานะในคิว แต่ถ้าไม่ต้องการก็จะจบออกจากโปรแกรมในส่วนนี้

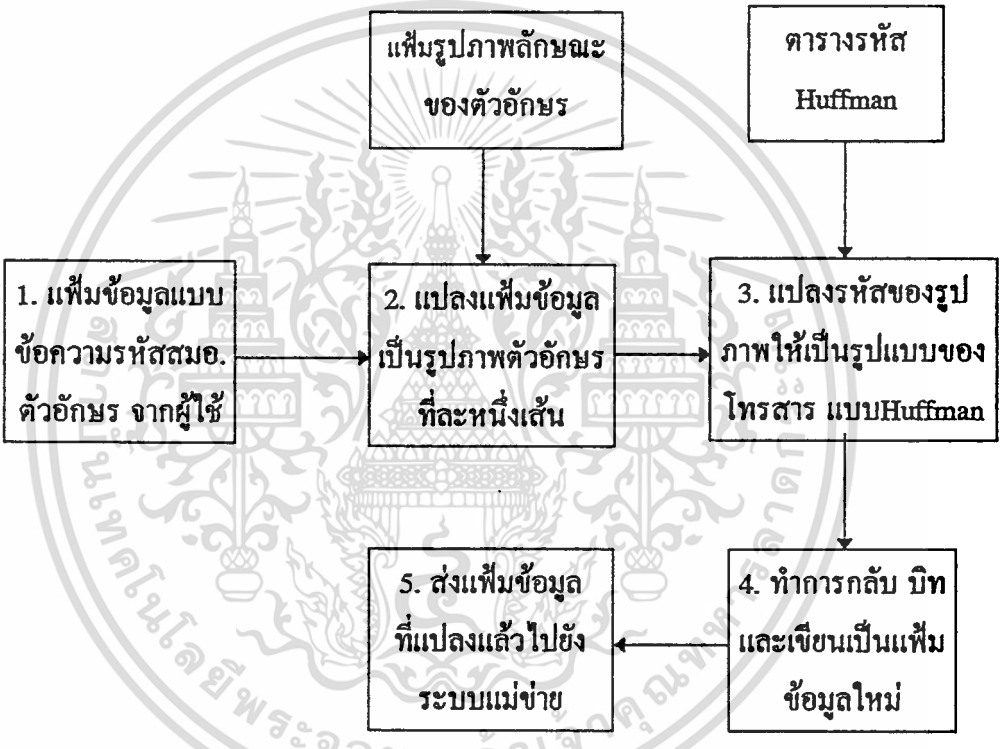
- ถ้าต้องการส่งโทรสารก็จะมีหน้าจอในการรับเบอร์โทรสาร ชื่อแฟ้มข้อมูล และระดับความเร่งด่วน จากนั้นก็จะเข้าไปยังส่วนที่ส่งแฟ้มไปยังแม่ข่าย เพื่อทำการแปลงรูปแบบแฟ้มข้อมูล และรับผลลัพธ์การส่งไปยังแม่ข่ายมาแสดง และจะส่งได้เรื่อยจนกว่าจะออกจากหน้าจอในการส่งโทรสาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ถ้าต้องการดูสถานะของคิว โปรแกรมก็จะไปทำการตรวจหาสถานะของคิวแล้วนำมาแสดงเมื่อจะจบการดูก็จะกลับมาอยู่ที่หน้าจอเริ่มต้น

3.1.2 ส่วนแปลงแฟ้มข้อมูลและการสร้างฟอนต์

การแปลงแฟ้มข้อมูลตัวอักษร แฟ้มข้อมูลภาษาไทยรหัส สมอ. ที่ได้รับจากผู้ใช้จะต้องมีการแปลง ตัวอักษรเหล่านั้นให้อยู่ในรูปแบบของโทรสาร



1. รับแฟ้มข้อมูลจากผู้ใช้

2. อ่านตัวอักษรจากแฟ้มข้อมูลตัวอักษรมาทำการเปิดแฟ้มรูปภาพตัวอักษรนั้นๆ แล้วอ่านค่าของรูปภาพที่ละหนึ่งแถวตามแนวนอนของตัวอักษรแต่ละตัวตลอดทั้งบรรทัด

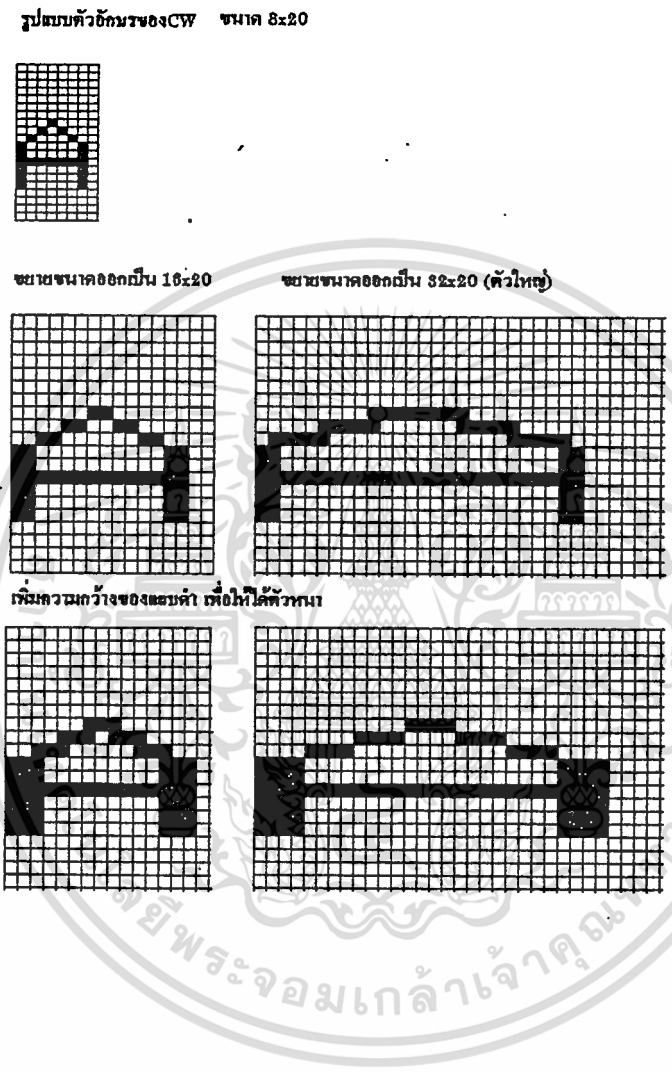
3. นับจำนวนจุดที่เป็นจุดขาวกี่จุด จุดดำกี่จุด เพื่อทำการบีบอัดข้อมูล โดยการเข้ารหัสของ Huffman แล้วจัดเรียงให้เป็นบิทที่ต่อๆ กันไป

4. กลับบิทของข้อมูลที่เข้ารหัสของ Huffman แล้ว เขียนลงแฟ้มข้อมูลใหม่

5. ส่งแฟ้มข้อมูลที่แปลงแล้วไปยังระบบแม่ข่ายโทรสาร

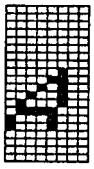
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การสร้างรูปแบบตัวอักษร นำรูปแบบตัวอักษรจากแฟ้มข้อมูลรูปภาพมาทำการขยายออกให้ได้กับขนาดตามที่ต้องการ ดังรูป

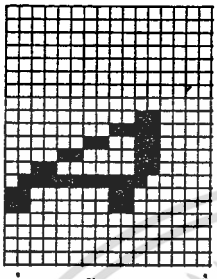


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

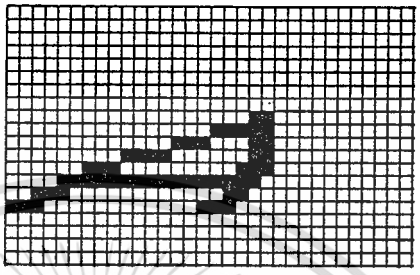
รูปแบบตัวอักษรของ CW ขนาด 8x20



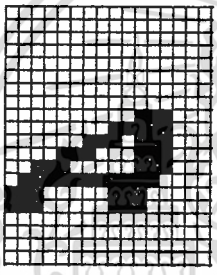
ขยายขนาดตัวอักษรเป็น 16x20

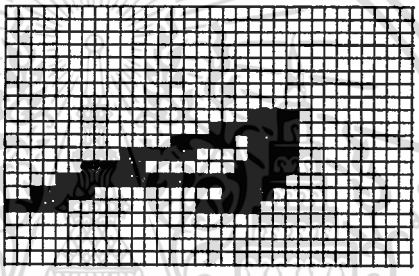


ขยายขนาดตัวอักษรเป็น 32x20 (ตัวเอียง)

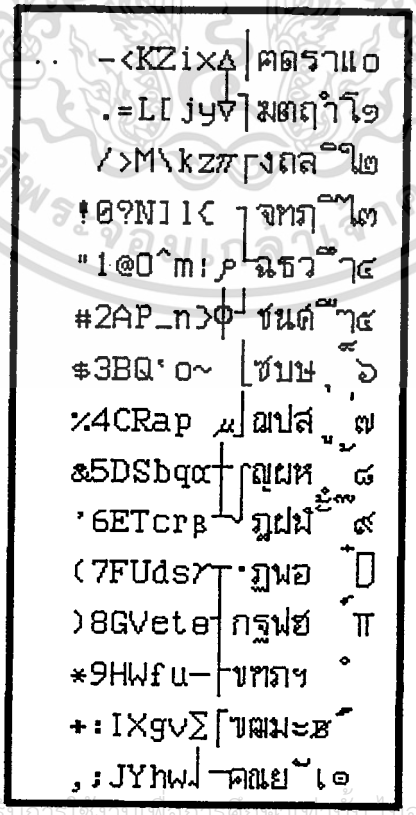


เพิ่มความกว้างของแบบตัวอักษรเพื่อให้เป็นตัวหนา





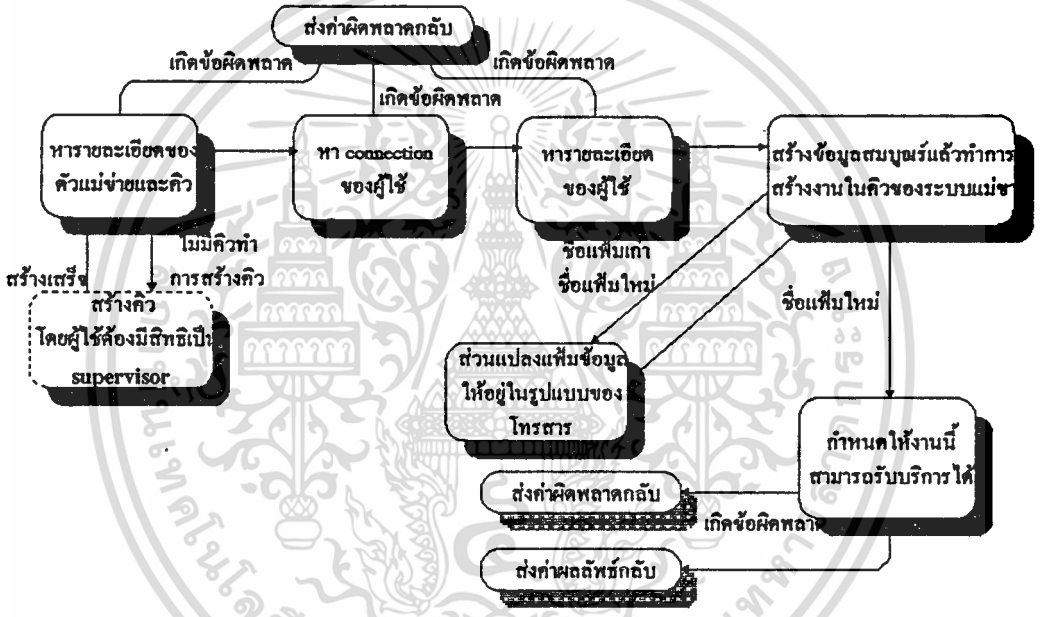
ตัวอักษรต่างๆ ที่มีให้ใช้ในแฟ้มข้อมูลตัวอักษร



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.3 ส่วนส่งงาน ผู้ใช้ไปยังตัวแม่ข่ายโทรสาร

โดยตอนแรกได้ออกแบบโดยใช้หลักการเขียนบน วิธีการ(protocal) IPX เลยแต่เมื่อได้ทดลองตามความต้องการของระบบ เห็นว่าไม่จำเป็นต้องใช้วิธีนี้ เพราะมีความซับซ้อนเกินไปในการทำ ดังนั้นจึงหันมาใช้ระบบการบริการคิวของโปรแกรม Netware โดยในที่นี้จะกล่าวเฉพาะที่ออกแบบบนการขอใช้บริการคิวของ Netware อย่างเดียว และบริการเชื่อมโยง ซึ่งสามารหาอ่านเพิ่มได้ในหนังสืออ้างอิงของโครงงานนี้



หลักการทำงานในส่วนนี้คือ : เมื่อรับข้อมูลจากส่วนรับข้อมูลแล้วส่วนการส่งงานผู้ใช้ไปยังตัวแม่ข่ายโทรสารจะดำเนินงานดังนี้

- จะทำการหารายละเอียดของตัวแม่ข่ายโทรสารและคิว โดยในปริญญานิพนธ์นี้ ได้กำหนด ชื่อคิว (Queue name) และชนิดของคิว (Queue Type) ตายตัวไปเลย เพื่อขอบเขตที่แน่ชัด โดยที่ในการให้บริการด้านคิวของ Netware และการบริการเชื่อมโยงออปเจ็คของ Netware เปิดโอกาสให้สามารถได้หลาย ๆ คิว โดยกำหนดชื่อคิวว่า FAX_Q และชนิดของคิวเป็น 0x8001 ซึ่งเป็นชนิดของออปเจ็คที่ว่างไว้สำหรับให้ผู้พัฒนาสามารถเขียนโปรแกรมประยุกต์ใช้งานได้ โดยได้กล่าวไว้ในเรื่อง Bindery Object แล้ว ซึ่งการหารายละเลียดของออปเจ็คแม่ข่ายนี้ จะเรียกรูทีนในการตรวจหาออปเจ็ค (Search Bindery Object) ที่เป็นบริการในการเชื่อมโยงออปต์เจ็ค (Bindery Service) ถ้าเกิดข้อผิดพลาดก็จะส่งผลลัพธ์กลับไปแสดง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ถ้าตรวจพบว่าไม่มี ออปต์เจ็คคิวของระบบแม่ข่ายแล้ว ถ้าหากผู้ที่ใช้เป็นผู้บริหารระบบหรือมีสิทธิเท่าเทียมก็จะสามารถสร้างออปต์เจ็คคิวของระบบแม่ข่ายได้ เนื่องจากในโครงงานนี้จะสร้างคิวใน "SYS:System เลขประจำตัวถ้าไม่มีสิทธิก็จะมีข้อผิดพลาดส่งกลับแสดง

- เมื่อตรวจว่ามีคิวแล้วก็จะได้เลขประจำคิวมา จากนั้นก็ต้องหารายละเอียดของผู้ใช้เพื่อใช้ในการแสดงผลและบันทึกเก็บไว้เป็นการใช้งาน ซึ่งจะเรียกใช้ ฟังก์ชั่น Netware เพื่อตรวจหา การเชื่อมต่อกับ Netware ของ User เป็นลำดับที่เท่าไร และก็ทำการนำข้อมูลมาใส่ในส่วนคำสั่งเพิ่มเติมในการสร้างคิว (Client Record Area) ซึ่งในไคล์เอนต์เรคอดแอเรียที่จะนำมาใช้ในโครงงานนี้สามารถแสดได้ดังนี้

โครงสร้างของ Client Record Area ที่ใช้ในโครงงานนี้

BYTE	ลำดับความเร่งด่วน (Priority)
BYTE	เบอร์โทรสารหลัก (Faxnumber) [20]
BYTE	ชื่อผู้ใช้ (User-names) [16]
BYTE	ขนาดของแฟ้มข้อมูล (file Size)
BYTE	จำนวนของการคัดลอก (Carbon Copy)
BYTE	ชื่อแฟ้มข้อมูลของผู้ใช้ (Source-filename) [14]
BYTE	ชื่อเส้นทางที่เก็บชื่อแฟ้มข้อมูล (Source-Directory) [[50]
BYTE	เบอร์โทรสารการคัดลอก 1 (Carbon Fax No.1) [20]
BYTE	เบอร์โทรสารการคัดลอก 2 (Carbon Fax No.2) [20]
BYTE	สงวนไว้ภายหน้า [8]

- เมื่อได้ข้อมูลทั้งหมดในการสร้างงานแล้ว ก็จะนำไปใส่ในส่วนร้องขอการสร้างงาน (Request Packet of Create Job Queue) แล้วทำการสร้างคิวโดยการเรียกใช้บริการคิวในการสร้างงานลงไปในคิว ถ้าสร้างงานเสร็จสมบูรณ์ก็จะทำการแปลงแฟ้มของผู้ใช้ให้เป็นรูปแบบของโทรสาร แล้วนำไปเก็บไว้ที่เส้นทาง (Directory Path) "SYS:MAIL\ *UserID*" โดยชื่อแฟ้มข้อมูลจะมีชื่อเหมือนกับแฟ้มที่สร้างในคิว+กับfx ตย. เช่นถ้าแฟ้มที่ถูกสร้างในคิว คือ 005C0001.Q แฟ้มนี้สร้างโดยโปรแกรมจะสร้งชื่อแฟ้มที่ถูกแปลงแล้วชื่อ 005C001.QFX เป็นต้น

- เมื่อเสร็จสิ้นส่วนการสร้างงานแล้วถ้าไม่เกิดข้อผิดพลาด ตัวโปรแกรมก็จะส่งผลลัพธ์ว่าทำงานเสร็จสิ้นไปให้ส่วนไปให้ส่วนรับความต้องการของผู้ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ซอฟต์แวร์ที่ทำงานอยู่บนเครื่องแม่ข่าย ประกอบด้วย

3.2.1 ส่วนบริหารงานคิวแม่ข่ายโทรสาร

จะเป็นส่วนที่ทำงานที่อยู่ในคิวของแม่ข่ายโทรสาร ออกมาส่งโทรสารตามความต้องการของผู้ใช้ โดยกลไกในการตรวจสอบว่าจะทำงานไหนออกมาส่งโดยใช้ Algorithm ดังนี้

```
Current-Priority = Maxpriority
While (1)
{
    For (index; index<Jobcount; index++)
    Job = ReadQueue[Index];
        Current_Time = Read Time FromServer
        if (Current_Time <= JobExecutime)
        {
          if (current_Priority == Job Priority)
          do {
              รอ 1 นาที
              ทำการส่งโทรสารโดยเรียกส่วนโทรสาร
              if (ส่งไม่สำเร็จ และส่งครบ 3 ครั้งแล้ว)
                  ผลลัพธ์ = มีข้อผิดพลาด
              else
                {
                  ลบแฟ้มโทรสารจาก SYS:MAIL
                  นำงานออกจากคิวของระบบแม่ข่ายโทรสาร
                }
            }
          while ( จะส่งจนสำเร็จหรือ ส่งครบ 3ครั้งแล้ว)
          if(ผลลัพธ์ == ส่งไม่สำเร็จ )
          {
            เปลี่ยนตำแหน่งของงานไปอยู่ลำดับสุดท้ายของคิว
            เปลี่ยนสิทธิไปเป็นผู้ใช้
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

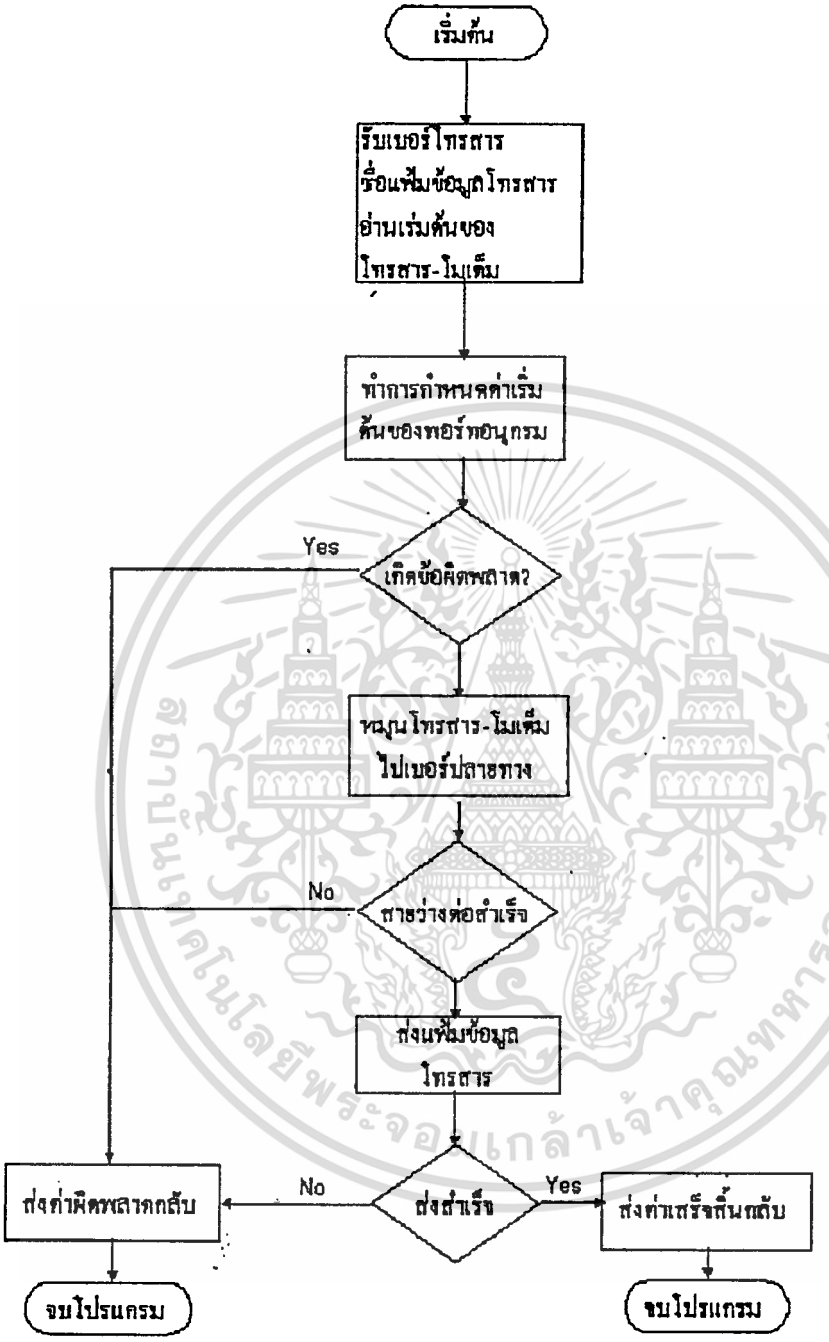
```
                    Job Priority = LowPriority
                    เปลี่ยนสิทธิกับมาเป็นผู้บริการ
                    ส่งเอกสาร(Message) ไปบอกผู้ใช้ว่าไม่สามารถส่งโทรสารได้
                    }
                  break;
                 }
              }
            }

          if( current_Priority == 0)
                current_Priority = Maxpriority;
          if( ไม่มีงานใดสามารถส่ง โทรสารได้เลย)
                ลด  current_Priority ลง 1
    }
```

โดยโปรแกรมจะวนรอ เพื่อตรวจงานทีเข้ามา และงานทีจะส่งโทรสารจะรอ 1 นาทีก่อนที่จะส่งซึ่งถ้างานนั้นไปสามารถส่งไปได้ ภายใน 3 ครั้ง งานนั้นก็จะถูกนำไปไว้ท้ายคิวพร้อมลดลำดับความสำคัญลงพร้อมทั้งจะส่ง เอกสาร(Message) ไปบอกผู้ใช้ด้วย

3.2.2 ส่วนจัดส่งโทรสาร

เป็นส่วนที่จะส่ง โทรสารไปยังปลายทาง ตามเบอร์และ แฟ้มข้อมูลโทรสารที่ได้รับจากส่วนบริหารงาน คิวของแม่ข่ายโทรสาร ซึ่งถ้าหากเกิดข้อผิดพลาดระหว่างการส่งข้อผิดพลาดกลับไปยังส่วนบริหารงานคิวแม่ข่ายโทรสารเพื่อดำเนินงานต่อไป ส่วนจัดส่งโทรสารเพื่อดำเนินต่อไป ส่วนจัดโทรสาร มีหลักการดังนี้คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
ขั้นตอนในการทำงาน

1. สร้างโปรแกรมเพื่อส่งโทรสารที่เครื่องลูกข่ายโดยตรง

ในส่วนของการทำโปรแกรมส่งโทรสารทางโทรสารโมเด็มนี้โดยใช้คำสั่ง"AT Command" เหมือนกับโมเด็ม ยี่ห้อ Hayes ซึ่งมีความเป็นที่นิยมใช้มากที่สุด รูปแสดงการติดต่อกันระหว่างฝั่งส่งกับฝั่งรับ เป็นขั้นตอนดังนี้

PC (DTE)	Local Modem	Remote Fax Terminal
AT+FCLASS=2.0 ->		
	<- "OK"	
ATDs ->	dial number	
		answer
	CNG ->	
	<- CED	
	<- "+FCO"	<- HDLC flags
	<- "+FNF:nsf"	<- NSF frame
	<- "+FCI:csi"	<- CSI frame
	<- "+FIS:dis"	<- DIS frame
	<- "OK"	
AT+FDT ->	TSI frame ->	
	DCS frame ->	
	TCF frame ->	
	<- "+FCS:cfr"	<- CFR frame
	<- "CONNECT"	
<image data> ->	<image data> ->	
<DLE><2Eh> ->	EOP frame ->	
	DCN frame ->	<- MCF frame
	<- "OK"	
ATHO ->	hang up	hang up

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมที่สร้างขึ้นมีลำดับขั้นตอนในการทำงาน คือ

1.1 รับค่าของ พอร์ทอนุกรม

1.2 กำหนดค่าเริ่มต้นของ พอร์ทอนุกรมนั้นตามค่าที่กำหนด

1.3 กำหนดกลุ่มของคำสั่งเป็นกลุ่มที่2 ตั้งค่าของโมเด็ม

1.4 ติดต่อไปยังเครื่องโทรสารปลายทาง ด้วยคำสั่ง ATDs คือคำสั่งในการหมุนโทรศัพท์ และรอการตอบรับของสัญญาณ จากนั้นโมเด็มและโทรสารปลายทางจะมีการแลกเปลี่ยนค่าต่างๆ ระหว่างกัน เพื่อกำหนดความถี่และรูปแบบให้ตรงกัน

1.5 เมื่อได้รับสัญญาณตอบรับ'OK' ก็จะส่งคำสั่ง 'AT+FDT' เพื่อเชื่อมต่อกันในการส่งข้อมูล แล้วรอจนกว่าเครื่องโทรสารปลายทางจะพร้อมที่จะรับข้อมูลก็จะตอบกลับมาด้วยคำว่า 'CONNECT'

1.6 เมื่อเครื่องโทรสารปลายทางพร้อมรับข้อมูลแล้ว ก็ส่งข้อมูลไปยังเครื่องโทรสารปลายทาง จนจบข้อมูล

1.7 ถ้าเกิด ข้อผิดพลาดที่ขั้นตอนใดก็แสดงค่าผิดพลาด และจบ Program

ปัญหาของการส่งและการแก้ไข

ปัญหา เมื่อส่งแฟ้มข้อมูลที่สร้างตามการเข้ารหัสแบบ Huffman ไปยังโทรสารปลายทาง จะเกิดข้อผิดพลาดในการส่ง เครื่องโทรสารไม่มีการพิมพ์ข้อความใดๆ ออกมา

แก้ไข จะต้องมีการกลับบิทของข้อมูลที่เข้ารหัส Huffman ก่อนที่จะทำการส่งออกโมเด็ม ทุกไบท์เพราะที่โมเด็มจะทำการส่งข้อมูลไปยังโทรสารปลายทาง โดยจะส่งครั้งละ 1ไบท์ เริ่มจากบิทต่ำก่อน ดังนั้นถ้ารหัสที่สร้างตามแบบรหัสของ Huffman ต้องมีการกลับบิทก่อน คือบิทสูงเป็นบิทต่ำ เช่น 1011 0101 จะกลับได้เป็น 1010 1101

ปัญหา เมื่อทำการส่งข้อมูลให้โทรสารปลายทางได้แล้ว แต่ตัวอักษรที่โทรสารพิมพ์ออกมา ยังมีในบางแถวของตัวอักษร และบางบรรทัด ตัวอักษรที่พิมพ์ออกมาไม่ชัดเจน ผิดเพี้ยนไป

แก้ไข เมื่อพบข้อมูลที่ที่มีค่าเป็น 10h จะต้องมีการส่งค่าข้อมูล 10h ซ้ำไปอีกหนึ่งไบท์ เพราะค่าข้อมูล 10h ไปตรงกับรหัสการขึ้นหน้าใหม่ของงาน คือ 10h 2Eh เมื่อ โทรสารปลายทางได้รับค่า 10h จะไม่ทำการถอดรหัสแต่จะทำการรอไบท์ต่อไป ว่าเป็นค่า 2Eh หรือไม่ ถ้าไม่ใช่ก็จะทำการถอดรหัสต่อไป ดังนั้นจึงต้องส่ง 10h ไป อีกหนึ่งครั้งเพื่อ ให้โทรสารปลายทางถอดรหัสได้ถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. สร้างรูปแบบตัวอักษร และแปลงแฟ้มข้อมูลตัวอักษรไปเป็นรูปแบบโทรสาร

การสร้างรูปแบบตัวอักษร

การจัดทำรูปแบบตัวอักษร ที่จะใช้ในการแปลงไฟล์ตัวอักษร ให้ไปเป็นรหัสที่เครื่องแฟกซ์จึงได้ทำการออกแบบตัวอักษรที่จะใช้ในการแปลงไฟล์ตัวอักษร ซึ่งรูปแบบตัวอักษรที่ใช้ได้ได้นำมาจากรูปแบบตัวอักษรภาษาไทยรหัส สมอ. คือไฟล์ชื่อ "Normal.fon"และ "Italic.fon"ของ CW ที่ใช้ในการแสดงผลที่หน้าจอ ซึ่งมีขนาดของตัวอักษรแต่ละตัวเท่ากับ 8x20 จุด ถ้านำมาใช้เลยพบว่าเมื่อพิมพ์ออกมาแล้วมีขนาดเล็กเกินไปจึงได้ทำการขยายขนาดตัวอักษรออกเป็นขนาด 16x20 โดยจะทำการเช็คทีละบิทของไฟล์รูปแบบตัวอักษรของCW ถ้าบิทนั้นเป็น 0 รูปแบบตัวอักษรใหม่ก็จะเป็นบิท 0 สองบิท เช่น

0001 0110 ก็จะได้เป็น 0000 0011 0011 1100

สำหรับรูปแบบตัวอักษรที่เป็นตัวใหญ่ ก็ได้จากการขยายขนาดรูปแบบตัวอักษรปรกติ คือจาก 16x20 เป็น 32x20 รูปแบบตัวอักษรที่ได้จากการขยายครั้งนี้จะเป็นแฟ้มข้อมูลตัวใหญ่หนา ส่วนตัวหนาของตัวอักษรปรกตินั้น ได้จากการขยายจุดดำ หรือบิทที่เป็นหนึ่ง คือเช็คดูทีละบิทถ้าสถานะของบิทเปลี่ยนจาก 1 เป็น 0 ก็จะกำหนดให้ค่าบิทที่เป็น 0 นั้นเป็น 1 เป็นการเพิ่มความยาวของจุดดำหรือบิท 1

เมื่อทำการขยายรูปแบบตัวอักษรแล้วพบว่า รูปร่างของตัวอักษรไม่สวยงามเส้นไม่ต่อเนื่องจึงเขียนโปรแกรมที่สามารถเรียกตัวอักษรแต่ละตัวขึ้นมาดู และทำการแก้ไขรูปแบบตัวอักษรนั้นได้ด้วย คือ จะรับค่าตัวอักษรจากคีย์บอร์ด แล้วมาคำนวนหาตำแหน่งของตัวอักษรตัวนั้นจากไฟล์รูปแบบตัวอักษร โดยการคูณกับขนาดจำนวนไบท์ของตัวอักษรแต่ละตัว แล้วเช็คบิทแต่ละบิท บิทที่เป็น 1 จะระบายสีในช่องของตำแหน่งบิทนั้นๆ

การแปลงไฟล์ตัวอักษร

การแปลง แฟ้มข้อมูลตัวอักษร

การแปลงไฟล์ตัวอักษร(text file)ให้ เป็นไฟล์ในรูปแบบของโทรสาร โดยการเข้ารหัสจะใช้ หลักการของ Huffman Code โปรแกรมที่ใช้ในการแปลงไฟล์ตัวอักษรมีหลักการดังนี้

1. ทำการอ่านไฟล์รูปแบบของตัวอักษร เข้ามาเก็บในอะเรย์ โดยตัวอักษรแต่ละตัวจะใช้พื้นที่ในการเก็บตัวอักษรแต่ละตัวเท่ากับ 40 ไบท์ รูปแบบของตัวอักษรก็จะถูกจัดเก็บเรียงตามลำดับกันไปในอะเรย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ทำการอ่านตัวอักษรจากไฟล์ตัวอักษร 1 บรรทัด เข้ามาเก็บในอะเรย์ ซึ่งจะมีขนาด 80 ไบท์ ดังนั้นถ้าใน 1 บรรทัดของไฟล์ตัวอักษร มีจำนวนตัวอักษรมากเกินกว่า 80 ตัวอักษร ก็จะถูกแบ่งออกเป็น 2 บรรทัดเมื่อโทรสารพิมพ์ออกมา

3. นำค่า 2 ไบท์แรกของตัวอักษร(1 scan line) ที่อ่านได้จากไฟล์ตัวอักษรจากไฟล์รูปแบบของตัวอักษร เก็บในอะเลย์ เรียงต่อๆกันไปจนครบบรรทัด หรือ 80 ตัวอักษร

4. นำค่าในอะเลย์มาทำการเช็คทีละบิท เพื่อทำการนับจำนวนบิทที่เป็น '0' และบิทที่เป็น '1' โดยเมื่อนับจำนวนบิทที่เป็น '0' ที่อยู่ติดกัน เมื่อพบบิทที่เป็น '1' ก็จะทำการนำจำนวนบิทนั้นไปเปิดตารางว่า บิท'0' ที่มีจำนวนบิทเท่านี้มีโค้ดเท่าไหร่ จากตาราง ค่า Modefine Huffman แล้วนำมาเรียงจัดไบท์

5. ทำการกลับบิทในแต่ละไบท์ คือ บิทสูงไปเป็นบิทต่ำ บิทต่ำเป็นบิทสูง เช่น 53h เป็น CAh แล้วเขียนลงไฟล์

6. อ่านค่า 2 ไบท์ถัดไป(scan line ต่อไป) ของตัวอักษรในบรรทัดเดิมมาทำเช่นเดิม จนครบทั้ง 40 ไบท์ หรือครบ 20 scanline แล้วจึงทำการอ่านตัวอักษรในบรรทัดต่อไป และนำมาทำในลักษณะเดิมจนจบโปรแกรม

การแปลงแฟ้มข้อมูลของ CW.

สำหรับการแปลงไฟล์ของ CW นั้น CW ได้กำหนดให้มีตัวอักษร 2 ขนาดคือ ตัวใหญ่และตัวปรกติ และยังสามารถแสดง ตัวหนาและตัวเอียงได้อีก ดังนั้นจึงได้สร้างรูปแบบของตัวอักษรขึ้นมา 8 รูปแบบ 8แฟ้มข้อมูลคือแฟ้มข้อมูลตัวปรกติ, ตัวใหญ่, ตัวหนา, ตัวใหญ่หนา, ตัวเอียง, ตัวเอียงหนา, ตัวใหญ่เอียง และ ตัวใหญ่เอียงหนา

รหัสของ CW ที่ใช้บอกลักษณะของตัวอักษรเป็นดังนี้

02h = ตัวหนา

05h = ตัวใหญ่

12h = ขีดเส้นใต้ 2 เส้น

13h = ขีดเส้นใต้ 1 เส้น

17h = ตัวเอียง

ลักษณะการเก็บรหัสของ CW จะมีรหัสบอกลักษณะของตัวอักษรนี้ ทั้งก่อนและ หลังตัวอักษรเหล่านั้น ถ้าเป็นตัวอักษรปรกติจะไม่มีรหัสบอกลักษระตัวอักษรเหล่านี้อยู่ และลักษณะการจัดเรียงลำดับของรหัสบอกลักษณะตัวอักษรจะเป็นดังนี้ 13h, 17h, 02h, 12h, 05h

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมที่เขียนขึ้นนั้นจะทำการแปลงในลักษณะเดียวกับ แฟ้มข้อมูลตัวอักษร เพียงแต่จะต้องมีการเช็คค่ารหัสบอกลักษณะตัวอักษร เพื่อที่จะได้รู้ว่าจะต้องใช้ไฟล์รูปแบบตัวอักษรแบบใหนเมื่อรู้ว่าจะต้องรูปแบบตัวอักษรแบบใหน ก็จะทำการอ่านไฟล์นั้นเข้ามาเก็บในอะเลย์โดยจะแยกตัวอักษณตัวปรกติกับตัวใหญ่ เมื่อรูปแบบตัวอักษรที่ต้องการเปลี่ยนไปก็จะต้องอ่านไฟล์รูปแบบตัวอักษรเข้ามาเก็บใหม่ทุกครั้ง

3. ส่วนการส่งไฟล์จากลูกข่ายไปยังแม่ข่ายโทรสาร

ในส่วนนี้จะเป็นส่วนที่นำผลลัพธ์จาก 2 ขั้นตอนแรกมาทำงานต่อคือ

3.1 สร้างโปรแกรมชื่อ Svcjob.c เป็นตัวแม่ข่ายในการทดลอง เพื่อใช้ดูว่ามีงานเข้ามาในคิวหรือไม่ โดยในโปรแกรมส่วนนี้ได้ใช้ Function บริการด้านคิวของ Netware

3.2 สร้างโปรแกรมส่วนส่งไฟล์จากลูกข่ายไปยังแม่ข่ายโทรสาร ในการส่งความต้องการของงานไปที่คิว ของตัวแม่ข่ายโทรสารมีขั้นตอนในการทำงานดังนี้

3.2.1 ตรวจหา ออปต์เจ็ค ของตัวให้บริการโทรสาร

3.2.2 ส่งงานลงไปในคิวในขั้นตอนนี้จะมีการแก้ไขในเขตข้อมูล (field data) ของ Netware บางส่วนเพื่อให้เป็นไปตามลักษณะงาน ที่ต้องการ

3.2.3 เรียกโปรแกรมแปลงแฟ้มข้อมูล เพื่อแปลงแฟ้มข้อมูลให้อยู่ในรูปแบบของโทรสาร

3.2.3 กำหนดให้ งานนั้นพร้อมที่จะรับบริการแล้ว

3.2.4 หากเกิด ข้อผิดพลาดขั้นตอนใด ๆก็จะแสดงข้อผิดพลาดพร้อมรหัสข้อมูลที่ผิดพลาดด้วย

3.3 ตรวจหาข้อผิดพลาดใด ๆที่เกิดขึ้นในระหว่างขั้นตอนการทำงานและ พยายามแก้ไขให้สมบูรณ์

4. ส่วนตัวแม่ข่ายโทรสาร

เป็นส่วนที่จะนำงานที่อยู่ในคิวแม่ข่ายโทรสารมาส่ง โทรสาร ไปยังเครื่องโทรสารปลายทางซึ่งมีขั้นตอนในการทำงานดังนี้

4.1 เรียกใช้โปรแกรมชื่อ Placejob.c เพื่อทำการส่งแฟ้มข้อมูลจากเครื่องลูกข่ายลงในคิวของแม่ข่าย โดยใส่ลักษณะงานให้มีความหลากหลายของข้อมูล โดยจะต้องมีส่วน ข้อมูลสำคัญ ๆ 3 อย่างที่ต้องส่งไป คือเบอร์โทรสารปลายทาง, ชื่อแฟ้มข้อมูลที่จะส่ง,วันเวลาที่ต้องการจะส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 แก้ไขโปรแกรมในการให้บริการที่ตัวแม่ข่ายโทรสาร เพื่อทำการตรวจหา งานที่เลยเวลาหรือ ถึงเวลาส่งแล้ว จากนั้นก็ตรวจสอบลำดับความสำคัญของงานสูงสุดก่อน แล้วเลือกงานมาส่งโทรสารจนหมดลำดับความสำคัญ แล้วจึงไปทำการอ่านใหม่ ถ้าส่งแล้วไม่เกิดข้อผิดพลาด ก็จะนำงานนั้นออกจากคิว แล้วส่งข้อความไปบอกผู้ใช้ว่า สามารถส่งโทรสารไปยังปลายทาง ได้เรียบร้อยแล้ว แต่ถ้าเกิดข้อผิดพลาดในการส่ง ก็จะพยายามในการส่งใหม่อีก 2 ครั้ง ถ้าหากยังส่งไม่ได้อีก งานนั้นก็จะถูกลดลำดับความสำคัญลง แล้วย้าย ตำแหน่งของงานไปไว้ที่ลำดับสุดท้าย จากนั้นก็จะส่งข้อความ ไปยังผู้ใช้ที่ส่งโทรสาร ว่ายังไม่สามารถส่ง โทรสารได้

4.3 เขียน โปรแกรมในส่วนที่จะลบงานที่ค้างอยู่ใน คิวเนื่องจากงานนั้นไม่ สามารถ ทำการส่งโทรสารได้สำเร็จ

5. ส่วนของโปรแกรมที่จะส่งโทรสารจากแม่ข่ายโทรสารไปยังโทรสารปลายทาง

นำโปรแกรมในขั้นตอนแรกมาแก้ไข และ ลองสมมุติค่าที่รับมาจาก โปรแกรมที่ใช้ในการติดตั้ง แม่ข่ายโทรสารและ โทรสารโมเด็ม

6. ส่วนที่เป็นตัวติดต่อกับผู้ใช้(User Interface)

ในส่วนนี้จะเป็นการสร้างส่วนที่ติดต่อกับผู้ใช้เพื่อรับความต้องการของผู้ใช้ นอกจากนี้ยังเพิ่มเติมในส่วนการตรวจดูสถานะของ คิว และลบงานของตัวเองออกจาก คิวได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

บทสรุปและวิจารณ์

ระบบแม่ข่ายโทรสาร จะเอื้ออำนวยความสะดวกต่อองค์กรที่มีการใช้โทรสารในการดำเนินงานร่วมกับระบบเครือข่ายคอมพิวเตอร์เป็นอันมาก ซึ่งในโครงงานนี้ได้ทำการพัฒนาโปรแกรม ในการส่งโทรสาร จากเครื่องลูกข่ายพร้อมๆกันได้ ทั้งนี้สามารถส่งแฟ้มข้อมูลที่เป็นภาษาอังกฤษ และ ภาษาไทย รหัส สมอ. ไปยังโทรสารปลายทางได้ ผ่านทางเครื่องแม่ข่ายโทรสาร โดยเครื่องแม่ข่ายจะเป็นผู้เลือกงานในคิวไปทำการจัดส่งเอง

จากโปรแกรมที่เขียนขึ้นจะเห็นว่ายังขาดส่วนของการ รับแฟ้มข้อมูล ที่เป็นรูปภาพ และแฟ้มข้อมูลรหัสอื่นได้ พร้อมทั้ง โปรแกรมในส่วนการเลือกงานอาจจะมี กรรมวิธี ที่ดีกว่าในการเลือกงานขึ้นมาส่งโทรสาร นอกจากนี้โปรแกรมในส่วนติดต่อกับผู้ใช้ยังไม่สวยงาม ทางคณะผู้จัดทำหวังว่า หากได้มีการพัฒนาโปรแกรมในส่วนนี้ให้ทำงานบน โปรแกรม วินโดน์ แล้วสามารถส่ง แฟ้มข้อมูลบนโปรแกรม วินโดน์ได้ จะเป็นโปรแกรมที่น่าใช้งานโปรแกรมหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

ชุดคำสั่งที่ใช้ควบคุมเครื่อง โทรสาร-โมเด็ม

+FAA (Select Fax Auto Answer Mode : โหมดตอบรับอัตโนมัติ)

รูปแบบ : AT+FAA = n

+FAA=0ตอบรับเมื่อโทรสารโมเด็มเลือกใช้เฉพาะโหมดที่กำหนดโดยคำสั่ง +FCLASS

+FAA=1ตรวจจับข้อมูลและการเรียกอัตโนมัติและตอบรับตามความเหมาะสม

ค่าเริ่มต้น : +FAA = 0

ในโหมดตอบรับอัตโนมัติ โมเด็มจะปรับตัวเอง โดยอัตโนมัติและตอบรับทันที

+FAXERR (Fax T.30 Error Response : ผลแสดงความผิดพลาดตามมาตรฐานโทรสาร)

รูปแบบ : AT+FAXERR = n

n คือ รหัสแสดงสถานะการวางสาย

แสดงถึง สาเหตุของการวางสาย และถูกกำหนดโดยโมเด็มที่อยู่ในระหว่างการสิ้นสุดในแต่ละขั้นตอน โมเด็มจะปรับให้พารามิเตอร์นี้เป็น 0 เมื่อเริ่มต้นแต่ละขั้นตอน ค่าของ n แสดงในตารางผลตอบสนอง +FHNG

+FBOR (Select Data Bit Order : การเลือกลำดับบิทของข้อมูล)

รูปแบบ : AT +FBOR = n

+FBOR = 0 ลำดับของบิทโดยตรงสำหรับเฟส B, C, D

+FBOR = 1 กลับลำดับของบิทเป็นตรงกันข้ามสำหรับเฟส C แต่ลำดับของบิทโดยตรง สำหรับเฟส B และ D

+FBOR = 2 ลำดับของบิทโดยตรงสำหรับเฟส C กลับลำดับของบิทสำหรับเฟส B และ D

+FBOR = 3 กลับลำดับของบิทสำหรับเฟส B, C, D

ค่าเริ่มต้น : +FBOR = 0

คำสั่งนี้จะเลือกลำดับของการส่งข้อมูลระหว่าง DTE กับโมเด็มท้องถิ่น (local modem) และการส่งข้อมูลออกไปยังปลายทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Direct-bit : บิทแรกในแต่ละไฟล์ที่ส่งจาก DTE ไปยังโมเด็มจะเป็นบิทแรกที่ส่งท้ายในการส่งผ่านการสื่อสารไปยังปลายทาง

Reverse-bit : บิทแรกในแต่ละไฟล์ที่ส่งจาก DTE ไปยังโมเด็มจะเป็นบิทสุดท้ายในการส่งผ่านการสื่อสาร

การกำหนดพารามิเตอร์นี้ ไม่ใช้กับลำดับบิทของการควบคุมและรูปแบบเฟรมที่สร้างโดยโมเด็ม

+FBUF (Query Buffer Size : การหาขนาดของบัฟเฟอร์)

รูปแบบ : AT+FBUF?

คำสั่งที่อ่านได้อย่างเดียวนี้ เป็นการถามขนาดของบัฟเฟอร์ของโมเด็ม โดยโมเด็มจะตอบ เป็นรูปของ bs, xoft, xont. bc โดยที่

bs : แสดงขนาดของบัฟเฟอร์

xoft : แสดงจำนวนไบท์ที่ต้องการใช้ในการกระตุ้น (trigger) XOFF

xont : แสดงจำนวนไบท์ที่จะใช้เป็น XON หลังจาก XOFF

bc : แสดงถึงลำดับไบท์ของบัฟเฟอร์

+FCFR (Confirmation to Receive Response : การยืนยันการรับผลตอบสนอง)

รูปแบบ : AT+FCFR

โมเด็มท้องถิ่น (local modem) จะส่งผลตอบสนองนี้ไปยัง DTE หลังจากได้รับสัญญาณ DCS และ TFC ที่ถูกต้องจากเครื่องโทรสารที่อยู่ไกลออกไป

+FCLASS (Service Class Identification and Control : แสดงลักษณะและการควบคุม)

รูปแบบ : AT+FCLASS? : เป็นการถามถึงกลุ่มของคำสั่งที่ใช้ในขณะนี้

AT+FCLASS=? : เป็นการถามถึงความสามารถของโมเด็มในการใช้คำสั่งในกลุ่มต่างๆ

AT+FCLASS=n : เป็นการกำหนดกลุ่มของคำสั่งที่ใช้ในโมเด็ม

คำสั่งนี้ใช้สำหรับการกำหนดและถามโหมดการทำงานของโทรสารโมเด็มและความสามารถ ซึ่งโมเด็มจะตอบกลับดังนี้

1 : EIA/TIA-578 กลุ่ม 1

2 : กำหนดโดยโรงงาน (SP-2388-A กลุ่ม 2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.0 : TIA/EIA-592 กลุ่ม 2.0

+FDCC (Set DCE Fax Capabilities : การกำหนดความสามารถของ DTE)

รูปแบบ : AT+FDCC = s

s คือรหัสของพารามิเตอร์ย่อยในขบวนการที่กำหนดโดย T.30

คำสั่งนี้จะเป็นการกำหนดความสามารถของโมเด็มท้องถิ่นให้เป็นไปตามมาตรฐาน CCITT T.30 เมื่อ DTE เปลี่ยนแปลง +FDCC โมเด็มจะบรรทึกคำสั่งนี้ลงในคำสั่ง +FDIS ด้วย รหัสของพารามิเตอร์ย่อยแสดงในตารางต่อไปนี้

Label	Function	Values	Description
VR	Vartical resolution	0	Normal, 98 lpi
		1	Fine, 196 lpi
BR	Data transfer speed	0	2400 bps, V.27ter
		1	4800 bps, V.27ter
		2	7200 bps, V.29 or V.17
		3	9600 bps, V.29 or V.17
		4	12000 bps, V33 or V.17
		5	14400 bps, V.33 or V.17
WD	Page width	0	1728 pixels in 215 mm
		1	2048 pixels in 255 mm
		2	2435 pixels in 303 mm
		3	1216 pixels in 151 mm
		4	864 pixels in 107 mm
LN	Page length	0	A4, 297 mm
		1	B4, 364 mm
		2	Unlimited length
DF	Data compression format	0	1-dimensional, modified Huffman
		1	1-dimensional, modified Read
		2	2-demensional, uncompressed
		3	2-demensional, modified Read

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Label	Function	Values	Description
EC	Error correction	0	Disable ECM
		1	Enable error correction mode, 64 bytes/frame
		2	Enable error correction mode,256 bytes/frame
BF	Binary file transfer	0	Disable binary file transfer
		1	Enable binary file transfer
ST	Scan time per line	0	0 ms
		1	5ms
		2	10 ms normal, 5 ms fine
		3	10 ms
		4	20 ms normal, 10 ms fine
		5	20 ms
		6	40 ms normal, 20 ma fine
		7	40 ms

ตาราง แสดงรหัสของพารามิเตอร์ย่อย (ต่อ)

+FCON (Fax Commection Response : ผลตอบสนองการติดต่อของโทรสาร)

รูปแบบ : +FCON

โมเด็มจะแสดงผลนี้เพื่อบอกให้รู้ว่า การติดต่อระหว่างเครื่องโทรสารประสบความสำเร็จโดยจะถูกส่งโดยโมเด็มท้องถิ่นไปยัง DTE หลังจากที่ได้รับแฟลกซ์ (flags) แรกของ HDLC

+FCR (Set Capcbility to Receive : การกำหนดความสามารถในการรับ)

รูปแบบ : AT+FCR = n

+FCR =0 โมเด็มจะไม่รับข้อมูลและไม่สามารถโพล (Poll) กับเครื่องโทรสารที่อยู่ไกลออกไปได้ ซึ่งจะใช้เมื่อโมเด็มมีบัฟเฟอร์ไม่พอ

+FCR=1 โมเด็มจะสามารถรับข้อมูลข่าวสารได้

พารามิเตอร์นี้จะถูกพิจารณาในระหว่างเฟส A และ D

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

+FCSI (Report Receive CSI Response : รายงานการรับสัญญาณตอบสนอง CSI)

รูปแบบ : +FCSI : "s"

s คือ ID ของสถานีที่ถูกเรียก

ผลตอบสนองนี้แสดงถึงการได้รับสัญญาณ CSI จากสถานีด้านถูกเรียก

+FDCS (Set Current Session Parameters : การกำหนดพารามิเตอร์ที่ใช้ในขบวนการ)

รูปแบบ : AT+FDCS = s

s คือรหัสของพารามิเตอร์ย่อยในขบวนการ T.30

ค่าเริ่มต้น : รหัสของพารามิเตอร์ทุกตัวจะถูกตั้งเป็น 0

คำสั่งนี้ใช้ในการกำหนดค่า DCS ที่จะส่งให้เครื่องโทรสารที่อยู่ไกลออกไป โดยเลือกจากสัญญาณ DIS ที่เครื่องโทรสารอยู่ไกลออกไปส่งมา ค่า s เป็นไปตามตารางค่า +FDCC

+FDCS (Report Current Session Parameters Resonse : การรายงานผลตอบสนองพารามิเตอร์ที่ใช้ในขบวนการ)

รูปแบบ : AT+FDCS : s

s คือ รหัสของพารามิเตอร์ย่อยในขบวนการ T.30

โมเด็มจะส่งสัญญาณตอบสนองนี้เพื่อแสดงลักษณะของการติดต่อ ตามที่เครื่องโทรสารที่อยู่ไกลออกไปส่งมา ค่าของ เป็นไปตามตารางค่า

+FDIS (Set Current Session Negotiation Parameter : การกำหนดพารามิเตอร์ตามการตกลงกันในขบวนการ)

รูปแบบ : AT+FDIS = s

s คือรหัสของพารามิเตอร์ย่อยในขบวนการ T.30

เป็นการกำหนดค่าพารามิเตอร์ลงใน DIS เฟรมโดยตรง

+FDIS (Transmit Phase C Data : ผลตอบสนองแสดงความสามารถของเครื่องโทรสารที่อยู่ไกล)

รูปแบบ : AT+FDIS : s

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

s คือรหัสของพารามิเตอร์ย่อยในขบวนการ T.30

สัญญาณแสดงถึงความสามารถของเครื่องโทรสารที่อยู่ไกล ตามที่ได้รับสัญญาณ DCS ของเครื่องโทรสารที่อยู่ไกล

+FDT (Transmit Phase C Data : การส่งข้อมูลในเฟส C)

รูปแบบ : AT+FDT[=]n

n = พารามิเตอร์ย่อย DF, VR, WD, LN

ในเฟส B : คำสั่งนี้จะบอกให้โมเด็มทำการติดต่อกับเครื่องโทรสารที่อยู่ห่างออกไป และส่ง DCS ไปด้วย

ในเฟส C : จะเป็นการเริ่มต้นการส่งข้อมูลต่อไปค่าของพารามิเตอร์ย่อยแสดงในตารางค่า +FDCC

+FET (End the Page or Document : สิ้นสุดหน้าหรือข้อมูล)

รูปแบบ : AT+FET = n

n = 0-15

เป็นคำสั่งแสดงการสิ้นสุดหน้าหรือข้อมูลที่ทำการส่ง ถ้าอยู่ในโหมดการไม่มีการควบคุมความผิดพลาด (non-error-control operation) โมเด็มจะส่ง RTC ไปกับเครื่องโทรสารที่อยู่ไกล และเข้าสู่เฟส D โดยการส่งรหัส post-page ตามข้อกำหนดใน T.30

+FET (Post-Page Message Response : ผลตอบสนองข้อความแสดง post-page)

รูปแบบ : +FET : n

n แสดงดังตารางต่อไปนี้

รหัส PPM	สัญลักษณ์ย่อโดย~	รายละเอียด
0	[PPS-]MPS	หน้าต่อไปในข้อมูลเดียวกัน
1	[PPS-]EOM	ข้อมูลถัดไป
2	[PPS-]EOP	ไม่มีข้อมูลที่จะส่งแล้ว
3	PPS-NULL	ส่วนอื่นๆของหน้าถัดไป
4	[PPS-]PRI-MPS	หน้าต่อไปและมีการขัดจังหวะ
5	[PPS-]PRI-EOM	ข้อมูลอื่นและมีการขัดจังหวะ
6	[PPS-]PRI-EOP	ทำทั้งหมดและมีการขัดจังหวะ

ตารางแสดงค่าของ n

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลตอบสนองนี้สร้างโดยโมเด็มท้องถิ่น โดยสร้างตามข่าวสาร post-page ที่รับมาจากเครื่องโทรสารที่อยู่ห่างออกไป และคำสั่งนี้จะทำหลังจากที่ทำคำสั่ง +FDR เรียบร้อยแล้ว

+FHNG (Call Termination Status response : ผลตอบสนองสถานะการเรียกต่อต่อ)

รูปแบบ : +FHNG : n

n คือรหัสในการวางสาย แสดงดังตารางต่อไปนี้

n	Class 1 Definition
	Call Placement and Termination
0	Normal and proper end of commection
1	Ring detect without successful handshake
2	Call aborted from +FK or <CAN>
3	Np loop current
	Transmit Phase A and Miscellaneous Errors
10	Unspecified phase A error
11	Np answer (T.30 T1 timeout)
	Transmit Phase B Hangup Codes
20	Unspecified phase B error
21	Remote cannot receive or send
n	Class 1 Definition
22	Command received error in transmit phase B
23	Command received invalid command received
24	Respomse received error
25	DCS sent three times without response
26	DCS/DTC received three times; DCS not response
27	Failure to train
28	Response received invalid response
	Transmit Phase C Hangup Codes
40	Unapecified phase C error

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

43	DTE to DCE data underflow
	Transmit Phase D Hangup Codes
50	Unspecified phase D error
51	Response received error
52	No response to MPS repeated three times
53	Invalid response to MPS
54	No response to EOP repeated three times
55	Invaoid response to EOP
56	No response to EOM repeated three times
57	Invalid response to EOM
58	Unspecified receive phase B error
	Receive Phase B Hangup Codes
70	Unspecified receive phase B error
71	Response receive phase B error
72	Command received error
73	T.30 T2 timeout, expected page not received
74	T.30 T1 timeout after EOM received
	Receive Phase C Hangup Codes
90	Unspecified receive phase C error
n	**Class 1 definition**
91	Missing EOL after 5 seconds
92	Unused code
93	DCE to DTE buffer overflow
94	Bad CRC or frame (ECM or BFT modes)
	Rescive Phase D Hangup Codes
00	Unspecifide receive phase D error
01	Response received invalid response received
02	Command received invalid response received
03	Unable to comtinue after PIN or PIP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่งนี้จะถูกโมเด็มส่งให้กับ DTE เพื่อแสดงถึงสาเหตุในการวางสาย และจะเก็บไว้ในพารามิเตอร์ +FAXERR ด้วย

+FK (Terminate Fax Session : สิ้นสุดขบวนการ)

รูปแบบ : AT+FK

คำสั่งนี้จะบอกให้โมเด็มหยุดขบวนการติดต่อในทันทีที่เป็นไปได้ โมเด็มจะส่งข่าวสาร DCN ในลำดับต่อไปแล้ววางสาย

+FLID (Set Local ID String : กำหนดค่า local ID string)

รูปแบบ : AT+FLID="s"

s คือรหัส ASCII 20 ตัว

เป็นการกำหนดค่า local ID ที่จะใช้ใน TSI/CSI เฟรม

+FPHCTO (Set Phase C Response Timeout : กำหนดเวลาออกจากเฟส C)

รูปแบบ : AT+PHCTO=n

n มีค่า 0-255 (*100 milliseconds)

ค่าเริ่มต้น : 30

คำสั่งนี้ใช้กำหนดเวลาที่โมเด็มจะคอยคำสั่ง จากเครื่องโทรสารที่อยู่ห่างออกไปหลังจากส่งข้อมูลเฟส C หมดแล้ว หรือเวลาที่จะรอข้อมูลหลังจากคำสั่ง +FDT ถ้าหมดเวลาที่ตั้งเอาไว้แล้วโมเด็มท้องถิ่นจะส่ง EOP ไปให้เครื่องโทรสารที่อยู่ไกล

+FPTS (Set Page Transfer Status : กำหนดสถานะการส่งข้อมูล)

รูปแบบ : AT+FPTS = ppr

ppr มีค่า 1-5

ค่าเริ่มต้น : 1

คำสั่งนี้จะกำหนดค่า post-page ซึ่งต้องดูจาก copy-checking หรือ สัญญาณแสดงคุณภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่า	สัญลักษณ์ย่อ	รายละเอียด
1	MCF	page good
2	RIN	page bad, retrain requested
3	RTP	page good, retrain requested
4	PIN	page bad, interrupt requested
5	PIP	page good, interrupt requested

ตารางแสดงค่าของ ppr

+FPTS (Receive Page Transfer Status Response : การรับผลตอบสนองสถานะการส่งข้อมูล)

รูปแบบ : +FPTS : *prr, lc[, blc, cblc[, lbc]]*

ppr : ค่าของข่าวสาร post-page ซึ่งกำหนดเพื่อให้ +FPTS ใช้ในการกำหนดคำสั่งแสดงสถานะการส่งข้อมูล

lc : ตำแหน่งของบรรทัด

blc : ตำแหน่งของบรรทัดที่เสีย

cblc : เรียงลำดับบรรทัดที่เสีย

lbc : ตำแหน่งของไบท์ที่เสียหาย

ผลตอบสนองนี้จะถูกส่งจากโทรสารโมเด็มไปยัง DCE เมื่อสิ้นสุดเฟส C เพื่อรายงานสถานะการส่งข้อมูล

+FPTS (Transmit Page Transfer Status Response : การส่งผลตอบสนองสถานะการส่งข้อมูล)

รูปแบบ : +FPTS : ppr

ppr : ค่าของข่าวสาร post-page ซึ่งกำหนดเพื่อให้ +FPTS ใช้ในการกำหนดคำสั่งแสดงสถานะการส่งข้อมูล

โมเด็มจะส่งผลตอบสนองนี้ไปยัง DTE เพื่อแสดงคุณภาพและข่าวสาร post-page จากเครื่องโทรสารที่อยู่ห่างออกไป

+FTSI (Report Received TSI Response : การรายงานผลตอบสนองการรับ TSI)

รูปแบบ : +FTSI : "s"

s คือ ID ของสถานีส่ง

ผลตอบแทนนี้แสดงถึง TSI ที่ได้รับจากสถานีผู้เรียก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่งพื้นฐาน AT

มีคำสั่ง AT อยู่ 2 ประเภท ประเภทแรกของคำสั่งกระทำการทันที เช่น การหมุนหมายเลขโทรศัพท์ ATD การตัดการติดต่อ ATH คำสั่งประเภทที่ 2 เป็นคำสั่งการเปลี่ยนค่าที่ตั้งไว้ ตัวอย่างเช่น คำสั่ง ATS7=90 ตั้งค่ารีจิสเตอร์ S7 เป็นการบอกโมเด็มว่าให้รอการตอบรับหลังจากหมุนหมายเลขโทรศัพท์เป็นเวลา 90 นาที

ON-LINE และ OFF-LINE

ก่อนที่เราจะออกคำสั่งต่อโมเด็ม โมเด็มจะต้องพร้อมรับคำสั่งเสียก่อนสถานะนี้เรียกว่า "คอมมานด์สเตท" คือทุกสิ่งที่ส่งไปยังโมเด็มจะถูกแปลความเป็นคำสั่ง โมเด็มปกติจะอยู่ใน คอมมานด์สเตท ในขณะที่มันไม่ได้ต่อเข้ากับโมเด็มตัวใดๆอยู่ ถ้าโมเด็มอยู่ในสภาวะ "ออนไลน์"คือเชื่อมต่ออยู่กับโมเด็มปลายทาง ทุกอย่างที่เราส่งให้โมเด็มจะถูกส่งไปยังโมเด็มปลายทาง ถ้าต้องการป้อนคำสั่งสู่โมเด็ม โดยทำให้โมเด็มกลับมาอยู่ในคอมมานด์สเตท โดยส่งส่งคำสั่งที่อยู่ในรูปของชุดลำดับอักขระหลีก ชุดลำดับอักขระหลีกของโมเด็มที่คอมแพทิเบิลกับ Hayes คือเครื่องหมายบวกซ้อนกันสามตัว + + + หลังจากนั้น โมเด็มจะกลับมาอยู่ในคอมมานด์สเตท

การตั้งค่ารีจิสเตอร์ S

รีจิสเตอร์ S เก็บค่าที่ตั้งไว้สำหรับควบคุมการปฏิบัติการของโมเด็ม คุณสามารถตั้งค่าและตรวจสอบรีจีสเตอร์ S ได้ด้วยคำสั่ง ATS ในการตั้งค่ารีจิสเตอร์ให้ป้อนหมายเลขรีจิสเตอร์และค่าที่ต้องการตั้ง ตัวอย่างเช่น ในการตั้งค่ารีจิสเตอร์ 0 ให้มีค่า 9 เราป้อนคำสั่ง

ATS0=9?

ในการตรวจสอบค่าของรีจิสเตอร์ S ให้ป้อนคำสั่ง ATS ตามด้วยหมายเลขรีจิสเตอร์และเครื่องหมายคำถาม โมเด็มจะตอบสนองกลับมาด้วยค่าและพร้อมท์ OK ตามที่แสดงดังนี้

ATS11?

095

OK

ในตัวอย่างนี้แสดงว่า S11 มีค่า 95

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่งเพิ่มเติม

คำสั่งเดิมที่ใช้กับโมเด็ม Hayes Smartmodem เริ่มต้นใช้อักษร A และจบด้วย Z นี่ทำให้เกิดคำสั่ง 26 คำสั่ง เมื่อโมเด็มเริ่มซับซ้อนเพิ่มมากขึ้น มีความจำเป็นต้องใช้คำสั่งมากกว่า 26 คำสั่งแล้ว เพื่อแก้ไขเรื่องนี้ Hayes ได้เสนอให้ใช้คำสั่งเพิ่มเติมขึ้นอีกชุด มีรูปแบบคือ

AT&X

โดยเครื่องหมาย & บอกให้รู้ว่า X เป็นชุดของคำสั่งที่เพิ่มเติม

หลายๆคำสั่ง

เราสามารถป้อนคำสั่งหลายๆคำสั่งเข้าสู่โมเด็มโดยให้อยู่บรรทัดเดียวกันได้ บรรทัดคำสั่งสามารถรับอักขระได้ถึง 40 อักขระ รวมทั้ง AT ด้วย ถ้าต้องการแยกคำสั่งก็สามารถใช้ช่องว่างแยกระหว่างคำสั่งแต่ละคำสั่งได้ ดังนี้

AT S7=90 V1 X4 DT 1-305-555-1234

การตั้งค่าโปรไฟล์ (Profile)

โมเด็มของ Hates จะจัดการกับการตั้งค่า 4 ชุด ชุดแรกคือ แอ็คทีฟคอนฟีกิวเรชัน (active comfiguration) คือกลุ่มของค่าที่ตั้งไว้แก่โมเด็ม โดยค่าเหล่านี้มีการใช้กันอย่างปกติเมื่อทำการต่อโทรศัพท์หรือตอบรับคอลล์เมื่อป้อนคำสั่ง AT หรือเปลี่ยนค่ารีจิสเตอร์ S โมเด็มจะเปลี่ยนแอ็คทีฟคอนฟีกิวเรชั่นใหม่ โมเด็มยังมีคอนฟีกิวเรชั่นอีกสามชุด ชุดหนึ่งไว้ในหน่วยความจำรอมของโมเด็ม และอีกสองชุดไว้ในหน่วยแรม ชุดที่จัดเก็บไว้ในรอมเรียกว่า factory configuration เป็นกลุ่มของค่าที่ตั้งไว้ล่วงหน้าโดยทางโรงงานเป็นผู้ตั้งแล้วจัดเก็บไว้ในรอม ซึ่งสามารถใช้คำสั่ง AT&F เพื่อฟื้นคือค่า factory configurationได้

คอนฟีกิวเรชั่นอีกสองตัวที่เหลือเรียกว่า ยูสเซอร์-โปรไฟล์ เนื่องจากมันถูกจัดเก็บไว้ใน NVRAM มันจะไม่หายไปเมื่อโมเด็มถูกปิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่ง AT ที่มักใช้กันประจำ

คำสั่งตอบรับ ATA

คำสั่งนี้สั่งให้โมเด็มตอบรับคอลย์มันถูกใช้เฉพาะเมื่อตอบรับด้วยมือเพราะโมเด็มของยังมีโหมดตอบรับอย่างอัตโนมัติ เมื่อมีการออกคำสั่ง ATA โมเด็มจะ off-hook ในโหมดตอบรับและรอรับสัญญาณแคเรียะจากโมเด็มต้นทางที่เรียกเข้ามา โมเด็มจะรอเป็นเวลาตามที่ค่าตั้งไว้ใน S7 หน่วยเป็นวินาที และจะวางหูถ้าไม่มีแคเรียะมา ดูรายละเอียดรีจิสเตอร์ S0 และ S7 ด้วย

คำสั่งหมุนหมายเลขโทรศัพท์ ATD ตามด้วยหมายเลขโทรศัพท์

คำสั่ง D สั่งให้โมเด็ม off-hook และหมุนหมายเลขโทรศัพท์ที่ระบุเป็นตัวเลขตามหลังคำสั่งโดยค่าดีฟอลต์ โมเด็มจะรอไดแอ็ลโทนก่อนที่จะเริ่มต้นหมุนหมายเลขโทรศัพท์ ถ้าไม่มีไดแอ็ลโทนมาภายใน S6 วินาที (S6 เป็นรีจิสเตอร์ S หมายเลข 6 ที่เก็บค่าตั้งไว้) โมเด็มจะส่งรหัสผลลัพธ์ออกมาเป็น "No Dial Tone" คำสั่ง X อนุญาตให้คุณเขียนทับเวลาที่รอสำหรับไดแอ็ลโทน

หลังจากที่หมุนโทรศัพท์แล้วโมเด็มจะรอเป็นเวลา S7 วินาทีสำหรับการตอบกลับถ้าโมเด็มสามารถสร้างการเชื่อมต่อได้ โมเด็มจะเข้าสู่โหมดออนไลน์ ถ้าการเรียกเข้าล้มเหลวโมเด็มจะกลับคือสู่คอมมานด์สเตท

คำสั่ง D มีหลายออฟชันที่อนุญาตให้ควบคุมการหมุนหมายเลขโทรศัพท์ ตัวดัดแปรสามารถใส่ร่วมกับอะไรก็ได้หลังคำสั่ง D ตารางต่อไปนี้แสดงตัวดัดแปรของไดแอ็ล

ตัวดัดแปร	จุดมุ่งหมาย
!	กดสวิตช์ฮุก (hook) ชั่วขณะแล้วปล่อย (flash)
,	หยุดชั่วขณะ 2 วินาที ในขณะหมุนโทรศัพท์
;	กลับคืนสู่คอมมานด์สเตทหลังการหมุนโทรศัพท์
P	หมุนโทรศัพท์ด้วยระบบโทรศัพท์แบบหมุน ค่านี้เป็นค่าดีฟอลต์
R	ต่อโทรศัพท์อยู่ในโหมด "รีเวอร์ส" (reverse) ซึ่งเป็นปกติของการหมุนโทรศัพท์ แต่โมเด็มอยู่ในโหมด "ตอบรับ"
S=x	หมายเลขโทรศัพท์ถูกจัดเก็บไว้ในตำแหน่ง x
T	หมุนโทรศัพท์ด้วยระบบโทรศัพท์แบบกดปุ่ม
W	รอไดแอ็ลโทน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวดัดแปร ! ที่แสดงในตาราง ใช้กดสวิตซ์ฮุคให้ปิดแล้วเปิดที่เรียกกันว่า "แฟล็ช" การกดสวิตซ์ตัวนี้จะเกิดการขัดจังหวะขึ้นในวงจรโทรศัพท์ ปกตินาน 600 มิลลิวินาที ในโทรศัพท์สมัยก่อนจะแฟล็ชให้โอเปอเรเตอร์รู้โดยการกดแล้วปล่อยสวิตซ์ฮุคบนโทรศัพท์ยุคเก่าที่แขวนอยู่ตามผนัง ระบบ PBX ของสำนักงานบางแห่งใช้การแฟล็ชสวิตซ์ฮุกเพื่อควบคุมออฟชันบางอย่างในระบบ PBX

ตัวดัดแปร W และ , มีประโยชน์ตอนที่โมเด็มต่ออยู่กับ PBX หรือระบบโทรศัพท์อื่นอยู่ ตัวดัดแปร Wบอกโมเด็มให้รอโดแอ็ลโทนเป็นเวลาหนึ่งวินาทีก่อนที่จะหมุนตัวเลขที่เหลือของโทรศัพท์ต่อ

ตัวดัดแปร T และ P บอกโมเด็มถึงวิธีที่ใช้หมุนโทรศัพท์คือ T เป็นการระบุให้หมุนในลักษณะโทรศัพท์แบบกดปุ่ม และ P เป็นการระบุโทรศัพท์ในลักษณะโทรศัพท์แบบหมุน

ตัวดัดแปร R สั่งให้โมเด็มหมุนหมายเลขแต่ทำงานในลักษณะของโมเด็มโหมดตอบรับ สามารถใช้วิธีนี้เพื่อต่อโทรศัพท์เข้าไปยังโมเด็มที่ตั้งไว้เป็นออลิจิเนต

คำสั่งเปิดและปิดการสะท้อนกลับ (Echo) ATE

โดยปกติคำสั่งที่ป้อนเข้าไปในโมเด็มจะสะท้อนกลับสู่พีซี ถ้าเราป้อนคำสั่ง ATZ ด้วยมือให้แก่โมเด็มเราจะเห็นอักขระ ATZ ในจอภาพ ถ้าใช้คำสั่ง ATE0 จะปิดทางคำสั่งสะท้อนกลับ เมื่อป้อนคำสั่งด้วยมือไปที่โมเด็มจะไม่เห็นสิ่งที่ป้อนเข้าไปเลย ถ้าใช้คำสั่ง ATE1 จะเป็นการเปิดทางสะท้อนกลับของคำสั่งที่ส่งไป โดยทั่วไปจะตั้งค่านี้เป็น E1

คำสั่งวางหูโทรศัพท์ ATH

คำสั่งนี้บอกให้โมเด็มวางหูโทรศัพท์และทำการตัดการสื่อสารโดยปกติจะตามหลัง +++

คำสั่งกำหนดชื่อให้โมเด็ม ATI

คำสั่ง ATI กำหนดชื่อประเภทและรุ่นของรอมของโมเด็มรุ่นนั้นๆ มีรายงานอยู่ 3 อย่างที่เตรียมไว้ คือ

ATI0 ส่งคืนรหัสผลิตภัณฑ์ของโมเด็ม

ATI1 ส่งคืนค่าค่าเช็กซัม (checksum) ของรอม

ATI2 ตรวจสอบค่าเช็กซัมของรอม และส่งข่าวสารว่า หรือ

ผู้ผลิตโมเด็มบางรายจัดเตรียมออฟชันของคำสั่ง เพิ่มเติมก็มี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตั้งขนาดของเสียงจากลำโพง ATL

คำสั่งนี้ช่วยให้ควบคุมขนาดของเสียงจากลำโพงได้ ATL0 เป็นการตั้งลำโพงให้ส่งเสียงเบาที่สุด และ ATL3 ส่งเสียงดังที่สุด ค่าดีฟอลต์ที่ตั้งไว้คือ ATL2 ในการปิดเสียงของลำโพงให้ใช้คำสั่ง ATM

ควบคุมลำโพงด้วย ATM

คำสั่ง M ใช้ควบคุมลำโพงภายในของโมเด็ม มันมีออฟชั่นต่างๆ ดังนี้

ATM0 ลำโพงถูกปิดอยู่เสมอ

ATM1 ลำโพงเปิดในขณะต่อโทรศัพท์ และปิดหลังจากที่การเชื่อมต่อได้เกิดขึ้นแล้ว

ATM2 ลำโพงถูกเปิดอยู่เสมอ

ATM3 ลำโพงถูกปิดในขณะต่อโทรศัพท์ และเปิดอยู่จนกระทั่งตรวจจับสัญญาณ carrial ได้

คำสั่งออฟฮุก (Go off Hook) ATO

คำสั่งนี้บอกโมเด็มออกจากคอมมานด์สเตทกลับสู่ออนไลน์สเตท มันถูกใช้เพื่อฟื้นคืนการเชื่อมต่อที่เคยต่อไว้แล้ว คำสั่ง ATO อนุญาตให้ป้อนชุดลำดับอักขระหลีก ป้อนคำสั่ง AT อื่นด้วยมือ และจากนั้นกลับคืนสู่ออนไลด์สเตท

คำสั่งโหมดเงียบ (Quiet Mode) ATQ

โดยปกติโมเด็มส่งคืนรหัสผลลัพธ์ ภายหลังที่คำสั่งหรือกลุ่มของคำสั่งเสร็จสมบูรณ์แล้ว คำสั่ง ATQ อนุญาตให้เปิดหรือปิดรหัสผลลัพธ์ ค่าดีฟอลต์ที่ตั้งไว้เป็น ATQ0 คืออนุญาตให้โมเด็มส่งรหัสผลลัพธ์การตั้งค่าเป็นโหมดเงียบ ใช้คำสั่ง ATQ1 เพื่อปิดทางรหัสผลลัพธ์ การตั้งค่าแบบที่สามคือ ATQ2 เป็นการเปิดทางรหัสผลลัพธ์ในขณะต่อโทรศัพท์และปิดทางเมื่อมีการตอบรับการเรียก

คำสั่งตั้งค่ารีจิสเตอร์ S ของโมเด็ม ATS

คำสั่งตั้งค่า ATSใช้เรียกดูและทำการเปลี่ยนเนื้อหาของรีจิสเตอร์ ของโมเด็ม ในการตั้งค่ารีจิสเตอร์ S ให้ใช้คำสั่ง

ATSx=y

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่ x คือหมายเลขของรีจิสเตอร์ และ y คือค่าที่มอบหมายให้แก่รีจิสเตอร์ เช่น ATS11=50 คือการตั้งค่ารีจีสเตอร์ S11 ให้มีค่าเป็น 50

ในการดูค่าของรีจิสเตอร์ให้ใช้คำสั่ง

ATSx?

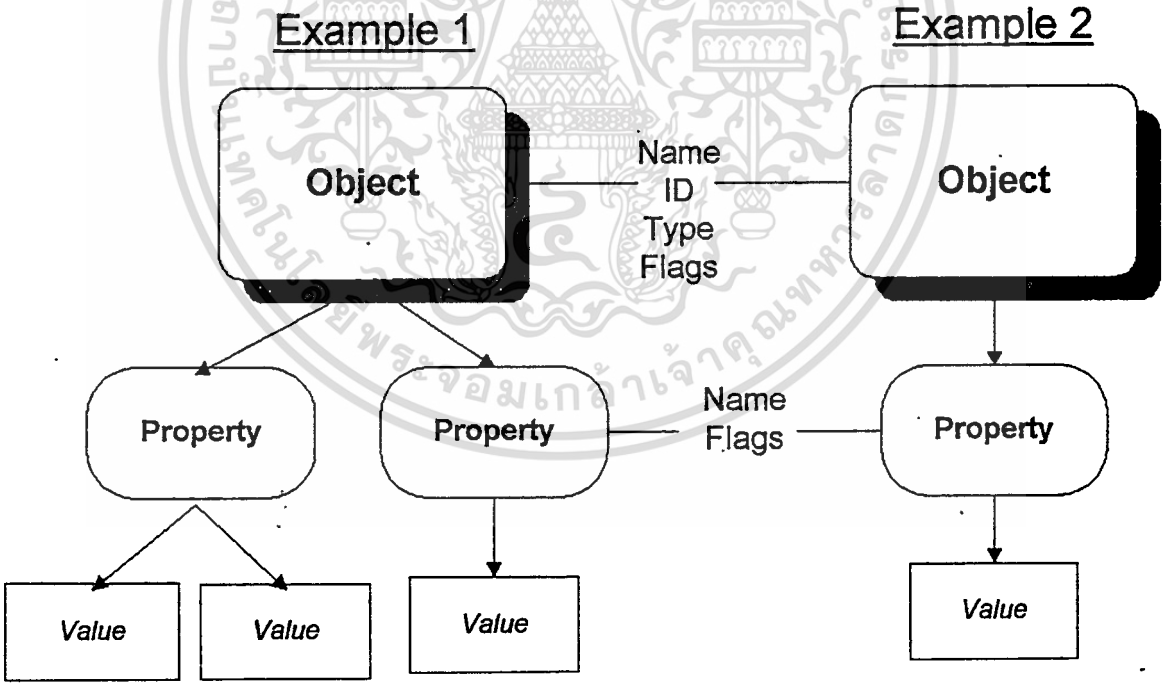
โดยที่ x คือ หมายเลขรีจิสเตอร์ที่ต้องการตรวจสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

หลักการให้บริการในการเชื่อมโยง (Bindery Service)

ภายใน Netwareแม่ข่ายแฟ้มข้อมูล จะเป็นการเชื่อมโยง (Bindery) ฐานข้อมูล (Database) สำคัญ เช่น user group คิว และบริการอื่นๆ การเชื่อมโยงได้ถูกเตรียมไว้เป็นรากฐานของ Netware ตั้งแต่เริ่มแรก คือ Naming Service และการ Design เปิดโอกาสให้มีการติดต่อที่แน่นอนของข้อมูลบนทรัพยากรที่ว่างอยู่บนแม่ข่ายแฟ้มข้อมูล การเชื่อมโยง(Bidery) จะจัดการเชื่อมโยงออปเจ็ค เป็นแบบเปลี่ยนแปลงได้ (Dynamic) เช่นแม่ข่ายแฟ้มข้อมูล สามารถที่จะมองเห็น หรือมองไม่เห็นบน Network และก็จะเหมือนกันทุก ออปเจ็ค อย่างเช่น User



รูป ลักษณะการเชื่อมต่อออปต์เจ็ค, พร็อพเพอร์ตี้ และ ค่าของพร็อพเพอร์ตี้ใน Netware

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยภาพที่มองเห็นในความคิด การเชื่อมโยงนี้จะแบ่งออกเป็น ฐานข้อมูล 3 ชั้น ประกอบด้วย ออปเจ็ค,พร็อพเพอร์ตี้ ที่มีความสัมพันธ์กับ ออปเจ็ค นั้นและค่าของแต่ละ properties ซึ่งทั้ง 3 ส่วนนี้มีข้อกำหนดง่ายๆแต่เป็นกลไกที่มีประสิทธิภาพในการบริหารทรัพยากร ดังรูป

การเชื่อมโยงออปเจ็ค (Bindery Object)

ส่วนประกอบพื้นฐานของการเชื่อมโยงก็คือ ออปเจ็ค ซึ่ง ออปเจ็ค ก็จะแยกโดย ชื่อ และชนิดของ ออปเจ็ค ซึ่งอาจจะเป็น user,group,ไฟล์เซอร์ฟเวอร์ ,Print Queue หรือสิ่งอื่นๆ แล้วแต่ผู้พัฒนาจะทำขึ้น ส่วน พร็อพเพอร์ตี้ คือ คุณสมบัติที่มีความสัมพันธ์กับ ออปเจ็ค นั้น ยกตัวอย่าง เช่น ข้อมูลที่มีความสัมพันธ์ กับ User คือ กลุ่มที่สังกัด ข้อมูลในการเข้าใช้ระบบ และรหัสผ่าน จะมี พร็อพเพอร์ตี้ ที่ตรงกันคือ GROUP_I'M_IN , LOGIN_CONTROL และ PASSWORD ซึ่ง พร็อพเพอร์ตี้ สามารถมีได้ 2 แบบ คือ

1. เซ็ท ซึ่งจะเป็นรายการของ เลขประจำออปเจ็ค ที่ใช้แยก ออปเจ็ค Number มีขนาด 4 ตัวอักษร

2. รายการ(items) จะเก็บข้อมูลที่เฉพาะตัวสำหรับ เขตข้อมูลเฉพาะเช่น พร็อพเพอร์ตี้ ของรหัสผ่าน จะเก็บเป็น สายของตัวอักษร

โดยที่แต่ละ พร็อพเพอร์ตี้ สามารถที่จะมีค่าได้ เช่น พร็อพเพอร์ตี้ รหัสผ่าน จะเก็บเป็นสายของตัวอักษร ซึ่งมีการเข้ารหัสไว้สำหรับแตละ ผู้ใช้

การรักษาความปลอดภัย ออปเจ็ค (Object Security)

เรื่องสำคัญอีกเรื่องในการเชื่อมโยง ก็คือ การรักษาความปลอดภัย ซึ่งจะมีผู้บริหารระบบเท่านั้นที่สามารถ กำหนด ระดับการรักษาความปลอดภัยของการเชื่อมต่อได้ โดยจะมีระดับในการเข้าใช้ดังนี้ (Bindery Access Levers) คือ

0 Anyone หมายถึง ทุกๆออปเจ็คอื่นสามารถเข้าใช้โดยที่ไม่ต้องเข้าใช้แม่ข่ายแฟ้มข้อมูลก่อน

1 Logged หมายถึง การที่จะเข้าใช้ได้ออปเจ็ค ต้องทำการเข้าสู่แม่ข่ายแฟ้มข้อมูล ก่อน

2 Object หมายถึง การที่จะเข้าใช้ได้ออปเจ็ค จะต้องเข้าสู่ไฟล์เซอร์ฟเวอร์ ด้วยชื่อออปเจ็คและรหัสผ่าน

3 Supervisor หมายถึง จะให้สิทธิในการเข้าใช้ออปเจ็คเฉพาะผู้ที่เป็นผู้บริหารระบบ หรือ มีสิทธิเท่า ผู้บริหารระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4 Netware หมายถึง จะให้สิทธิในการเข้าใช้ออปเจ็คเฉพาะ Netware เพื่อการรักษาความปลอดภัย

ซึ่งการกำหนด การรักษาความปลอดภัยของ ออปเจ็ค และ คุณสมบัติได้ 2 แบบ คือ สิทธิในการอ่านและการเขียน ค่ารักษาความปลอดภัยจะเก็บในรูป 1 ตัวอักษร โดย 4 bitบนจะหมายถึง สิทธิในการ เขียน และ 4 bit ล่างจะหมายถึงสิทธิในการอ่าน

ออปเจ็ค (Object)

ออปเจ็ค เป็นส่วนประกอบพื้นฐานของการเชื่อมโยง ซึ่งการเชื่อมโยงของออปเจ็คจะกำหนดโดย ชื่อ และ ชนิดของออปเจ็ค (ดังนั้น จึงทำให้มี 2 ออปเจ็คสามารถมีชื่อเดียวกัน แต่ทำหน้าที่ต่างกันได้) นอกจากนั้นยังมี เขตข้อมูลอื่นๆอีกที่สัมพันธ์ กับ ออปเจ็ค เช่น แฟล็ก(flag) และ เขตข้อมูลรักษาความปลอดภัย

ชื่อออปเจ็คในการเชื่อมโยง (Bindery Object Name)

การกำหนดชื่อของออปเจ็ค สามารถตั้งได้ยาวถึง 48 ตัวอักษรและที่อักษรตัวสุดท้ายจะต้องเป็นอักษรบอกจบ(Null) ทั้งนี้ต้องไม่มี อักษรควบคุม ช่องว่าง สแลส แบ็คสแลส โคลอน เซมิโคลอน ลูกน้ำ ดอกจัน และ ? ในชื่อของออปเจ็ค

เลขประจำออปเจ็คในการเชื่อมโยง (Bindery Identification Number)

เมื่อมีการสร้างออปเจ็คขึ้น ออปเจ็คนั้นก็จะมีถูกกำหนดรหัสประจำ ออปเจ็คอยู่ในรูปของเลขจำนวน 4 ตัวอักษร และจะเป็นชุดเดียวเท่านั้น ภายในแม่ข่ายแฟ้มข้อมูล

ชนิดของออปเจ็คในการเชื่อมโยง (Bindery Object Type)

จะมีค่าจำนวน 2 ตัวอักษร สำหรับทุกๆ ออปเจ็ค ซึ่งค่านี้จะใช้ในการตรวจหาออปเจ็คซึ่งถ้าหากไม่รู้ว่าชนิดของออปเจ็คเป็นแบบไหน จะมีค่าเป็น 0 โดยออปเจ็คที่มีค่าตั้งแต่ 1-0x8000 จะถูกสงวนไว้สำหรับ Netware และถัดจากนั้นจนถึง 0xFFFE จะมีไว้สำหรับผู้ที่จะพัฒนาใช้ทำโปรแกรมประยุกต์ และ และค่า 0xFFFF ใช้สำหรับในการ ตรวจหาออปเจ็คที่ไม่ได้มีการกำหนดชนิด โดยชนิดของออปเจ็คแสดงได้ดังนี้

ชนิด	ค่าฐาน16
Unknow	0h

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

User	1h
User Group	2h
Print Queue	3h
File Server	4h
Job Server	5h
GateWay	6h
Print Server	7h
Archive Queue	8h
Archive Server	9h
Job Queue	Ah
Administation	Bh
Remote Bride Serever	24h
Advertisting Bridge Server	47h
Reserved up to	8000h
Available for develop	8001-FFFEh
Wild	FFFF(-1)

แฟล็กการเชื่อมโยงออปเจ็ค

จะเป็นเลขจำนวน 1 ตัวอักษรที่แยกว่า ออปเจ็คนี้เป็นแบบ เปลี่ยนแปลงได้(Dynamic) มีค่า(1) หรือ ไม่เปลี่ยนแลง(Static) มีค่าเป็น(0) โดยที่ถ้าออปเจ็คเป็นแบบ สแตติค ออปเจ็ค นั้นจะอยู่จนกว่าจะมีการ ลบออปเจ็คนั้น ในขณะที่ ถ้าเป็นแบบ ไดนามิค ออปเจ็ค จะถูกลบออกไปโดยอัตโนมัติเมื่อแม่ข่ายแฟ้มข้อมูล ถูกปิด

แฟล็กรักษาความปลอดภัยการเชื่อมโยงออปเจ็ค

จะประกอบด้วย 1 ตัวอักษร ซึ่งจะกำหนดว่าสามารถอ่านหรือเขียนลงไปในออปเจ็คได้ ดังที่กล่าวแล้วในเรื่อง การรักษาความปลอดภัยออปเจ็ค (Object Security)

คุณสมบัติของออปเจ็ค (Properties)

ใน Netware คำว่า พร็อพเพอร์ตี้ คือตัวข้อมูลซึ่งสัมพันธ์กับออปเจ็ค โดยที่พร็อพเพอร์ตี้จะไม่ได้เก็บตัวข้อมูลไว้เอง เช่น พร็อพเพอร์ตี้ IDENTIFICATION จะไม่ได้เก็บรายชื่อของผู้ใช้ไว้แต่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าข้อมูล(Value) จะเป็นตัวที่เก็บข้อมูลจริงๆของพร็อพเพอร์ตี้ไว้ในที่นี้ใน ค่าข้อมูล จะเก็บ ชื่อผู้ใช้ไว้ นอกจากนี้ พร็อพเพอร์ตี้ยังมีอีกหลายเขตข้อมูลเหมือนกันกับในออปเจ็ค คือชื่อของ พร็อพเพอร์ตี้ แฟล็กพร็อพเพอร์ตี้ และ แฟล็กรักษาความปลอดภัยของพร็อพเพอร์ตี้

ชื่อพร็อพเพอร์ตี้ (Properties name)

จะเก็บชื่อของ พร็อพเพอร์ตี้ไว้มีขนาดความยาว 15 ตัวอักษร ไม่รวม อักษรควบคุม ช่องว่าง สแลส แบ็คสแลส โคลอน เซมิโคลอน ลูกน้ำ คอกจัน และ ? ในชื่อของออปเจ็ค

แฟล็กรักษาความปลอดภัยของพร็อพเพอร์ตี้(Properties Security flag)

จะเหมือนกันกับในส่วน แฟล็กรักษาความปลอดภัยการเชื่อมโยงออปเจ็ค

พร็อพเพอร์ตี้แฟล็กของการเชื่อมโยง(Binder Properties flag)

จะเหมือนกันกับในส่วน ออปเจ็คแฟล็ก ยกเว้นกันตรงที่ bit สูงสุดของแฟล็กจะใช้เป็นตัวบอกว่าเป็น พร็อพเพอร์ตี้แบบไหน แบบ เซ็ท(1) หรือ รายการ(0)

ค่าข้อมูล(Value)

ค่าข้อมูลเป็นค่าของข้อมูลของ พร็อพเพอร์ตี้นั้นจริงๆซึ่งค่าข้อมูลจะเป็น รายการ หรือ เซ็ท ก็ขึ้นกับbit บนสุดของพร็อพเพอร์ตี้แฟล็กของการเชื่อมโยงว่าเป็น 1 หรือ 0 โดยที่ ค่าข้อมูลสามารถที่จะมีขนาดใหญ่กว่า 128 ตัวอักษรของขนาดผลลัพธ์(Reply packet) ซึ่งขนาดของผลลัพธ์ยังสามารถเชื่อมโยงกันได้ถึง 128 รีพายเพ็คเกท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

Boogie Bug, "เทคโนโลยีของแฟกซ์", วารสารชอปส์ปิ้งคอมพิวเตอร์ , ฉบับที่ 46, 2535, หน้าที่ 159-165.

Borland International, "Turbo C Reference Guid" , Borland International, 1987.

Borland International, "Borland C++ version 3.1 Usere Guid" , Borland International, 1992.

Charles G. Rose, "Programmer's Guid to NetWare", McGraw-Hill ,1990.

Robert L. Hummel, "Programmer 's Technical Reference: Data and Fax Communications", Ziff-Davis Press, 1993.

Schwaderer W. David, " C Programmer 's Guide to NetBios, IPX, and SPX ", SAMS Publishing, 1992.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้