# โครงข่าย X.25

# X.25 NETWORK


โดย

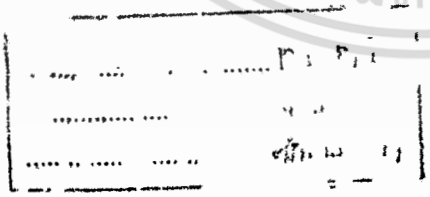
นายพิษณุพงษ์ วงษ์ขวัญ 35103240

นายไพบูลย์ รุ่งเจริญพร 35103242

นายรณรงค์ ชนากร 35103243

อาจารย์ที่ปรึกษา

ดร. วรวัฒน์ ลิ้มโภคา

วิทยานิพนธ์สำหรับปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมคอมพิวเตอร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2537


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้ง



035034

ปริญญานิพนธ์ปีการศึกษา 2537

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง โครงข่าย X.25 ( X.25 NETWORK )


ผู้จัดทำ

1. นายพิษณุพงษ์ วงษ์ขวัญ เลขประจำตัว 35103240
2. นายไพบูลย์ รุ่งเจริญพร เลขประจำตัว 35103242
3. นายรณรงค์ ชนากร เลขประจำตัว 35103243

..... อาจารย์ที่ปรึกษา

( ดร. วรวัฒน์ ลิ้มโภคา )



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# โครงข่าย X.25

นายพิษณุพงษ์ วงษ์ขวัญ
นายไพบูลย์ รุ่งเจริญพร
นายรณรงค์ ชนากร
ดร.วรวัฒน์ ลิ้มโภคา อาจารย์ที่ปรึกษา
ปีการศึกษา 2537

## บทคัดย่อ

X.25 เป็นโปรโตคอล (protocol) สำหรับการเชื่อมต่อโครงข่ายที่มีลักษณะของการรับส่งข้อมูลเป็นแบบแพคเกท ใน Public Switching Data Network (PSDN) X.25 จะทำการเชื่อมโยงระหว่าง Data Terminal Equipment (DTE) กับ Data Circuit Terminating Equipment (DCE) โดยครอบคลุมทั้ง 3 Layer ของ Open System Interconnection (OSI)

มีหลายหน่วยงานที่นำโปรโตคอล X.25 มาใช้งานไม่ว่าจะเป็นหน่วยงานจากภาครัฐและภาคเอกชน เพราะมีข้อดีหลายอย่างทั้งในด้าน Speed Conversion, Flow Control, Fast Select, PVCs, ความปลอดภัย, ความถูกต้องของข้อมูล, การรองรับกับโปรโตคอลอื่นๆ

ปริญญานิพนธ์นี้ จะกล่าวถึงหัวข้อต่างๆ คือ มาตรฐาน OSI , ทฤษฎี X.25 รวมถึงการเปรียบ เทียบ Layer ทั้ง 3 ของ OSI , Packet Assembly/Disassembly (PAD) รวมถึงมาตรฐาน X.3,X.28 และ X.29 นอกจากนี้จะเป็นผลการทดลองการเขียนโปรแกรมสื่อสารผ่าน Modem ซึ่งอาจจะเป็นประโยชน์ในการพัฒนาการสื่อสารต่อไปในอนาคต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# X.25 NETWORK

Phaiboon Rungjareanporn
Phisanupong Wongkwan
Ronnarong Chanakorn
Dr. Worawat Limpoka Advisor
1994

## Abstract

X.25 is protocol for connect network which characteristic of data communicate with packet in Public Switching Data Network (PSDN). It due to connect between Data Terminal Equipment (DTE) and Data Circuit Terminating Equipment (DCE). By to cover 3 layer of Open System Interconnection (OSI)

More company used protocol X.25 either government and unanimity because it has benefit of speed conversion, flow control, fast select, PVCs, safety, connect of data and support with another protocal

This thesis to told standard of OSI model, X.25 theory, and compare 3 layer of OSI model, Packet Assembly/Disassembly (PAD) and standard of X.3, X.28 and X.29. In addition to result of test which communicate of program with modern which maybe useful in to development communicate to furture

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ


| | หน้า |
|---|---|
| บทคัดย่อ | I |
| Abstract | II |
| สารบัญ | III |
| สารบัญรูปภาพ | VI |
| สารบัญตาราง | VIII |
| **บทที่ 1 The Open System Interconnection (OSI)** | **1** |
| 1.1 The Physical Layer | 8 |
| 1.2 The Data Link Layer | 11 |
| 1.3 The Network Layer | 13 |
| 1.4 The Transport Layer | 17 |
| 1.5 The Session Layer | 20 |
| 1.6 The Presentation Layer | 23 |
| 1.7 The Application Layer | 25 |
| **บทที่ 2 ลักษณะทั่วไปของ X.25** | **29** |
| 2.1 Packet Switching | 30 |
| 2.2 Packets | 31 |
| 2.3 การเชื่อมโยงระหว่าง DTE กับ DCE | 33 |
| 2.4 Virtual Circuits | 34 |
| 2.5 Logical Channels | 35 |
| 2.6 Datagrams | 36 |
| 2.7 วิธีการของ Datagram และ Virtual Circuit | 37 |
| **บทที่ 3 X.25 Level 1 - Physical Layer** | **39** |
| 3.1 บทนำ | 39 |
| 3.2 Interchange Circuits | 40 |
| 3.3 คุณลักษณะทางไฟฟ้า | 42 |


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ (ต่อ)


| | หน้า |
|---|---|
| **บทที่ 4 X.25 Level 2 - Data Link Layer** | **44** |
| 4.1 บทนำ | 44 |
| 4.2 ข้อสังเกตที่เกี่ยวกับตาราง HDLC | 44 |
| 4.3 วิธีการปฏิบัติที่เกี่ยวกับ Addressing | 44 |
| 4.4 Poll/Final bit | 44 |
| 4.5 ขั้นตอนต่างๆ สำหรับ Link set-up | 45 |
| 4.6 ขั้นตอนต่างๆ สำหรับการส่งถ่ายข่าวสาร | 46 |
| 4.7 Procedure for resetting | 49 |
| 4.8 System Parameters | 50 |
| **บทที่ 5 X.25 Level 3 - Network Layer** | **51** |
| 5.1 บทนำ | 51 |
| 5.2 Ready Phase | 52 |
| 5.3 Virtual Procedure | 54 |
| 5.4 Permanent virtual circuit service | 70 |
| 5.5 Datagram service | 72 |
| 5.6 Optional suser facilitie | 75 |
| **บทที่ 6 มาตรฐาน PAD,X.3,X.28 และ X.29** | **85** |
| 6.1 บทนำ | 85 |
| 6.2 X.3 Parameter | 86 |
| 6.3 X.28 Command and response | 94 |
| 6.4 X.29 and Remote PAD Configuration | 95 |
| **บทที่ 7 โปรแกรมด้านการสื่อสาร** | **97** |
| 7.1 ความรู้ด้านโปรแกรมสื่อสาร | 97 |
| 7.2 ลำดับขั้นตอนการทำงาน | 102 |
| 7.3 การพัฒนาโปรแกรม | 104 |


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ (ต่อ)

หน้า

บทที่ 8 บทวิจารณ์และบทสรุป..... 113

ภาคผนวก Frames , Packets และ Status

กิตติกรรมประกาศ

หนังสืออ้างอิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญรูปภาพ


| | | หน้า |
|---|---|---|
| รูปที่ 1 | OSI Reference Model Layers | 1 |
| รูปที่ 2 | ทางเดินระหว่าง User Machines | 13 |
| รูปที่ 3 | การเติมส่วน Headers และ Trailers | 28 |
| รูปที่ 4 | การแยกส่วน Headers และ Trailers | 28 |
| รูปที่ 5 | แสดง Level ทั้ง 3 ของ X.25 กับ PSE (Packet Switching Exchange) | 29 |
| รูปที่ 6 | แสดง Packet Switching Network (PSN) | 30 |
| รูปที่ 7 | แสดงลักษณะของแพคเกทสวิทชิ่งใน PSDN | 32 |
| รูปที่ 8 | แสดงการเชื่อมต่อระหว่าง DTE / DCE | 33 |
| รูปที่ 9 | Virtual Circuit | 34 |
| รูปที่ 10 | Permanent Virtual Circuit (PVC) | 35 |
| รูปที่ 11 | (ก) Datagram | 37 |
| รูปที่ 11 | (ข) Virtual Circuit | 38 |
| รูปที่ 12 | X.21 Interchange Circuit | 40 |
| รูปที่ 13 | Connector Pin Numbering | 42 |
| รูปที่ 14 | โครงสร้างโดยทั่วไปของ Network Layer | 52 |
| รูปที่ 15 | Ready Phase | 54 |
| รูปที่ 16 | Call Set-Up Procedure | 55 |
| รูปที่ 17 | DTE Addressing in X.25 | 57 |
| รูปที่ 18 | DTE and DCE data Packets | 59 |
| รูปที่ 19 | Packet Sent sequence numbers P(S) | 61 |
| รูปที่ 20 | State Diagram | 67 |
| รูปที่ 21 | DTE and DCE DATAGRAM | 72 |
| รูปที่ 22 | DATAGRAM SERVICE SIGNAL | 73 |
| รูปที่ 23 | CLOSED USER GROUP | 79 |
| รูปที่ 24 | ความสัมพันธ์ของ X.3, X.28 และ X.29 | 85 |


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญรูปภาพ (ต่อ)

หน้า

รูปที่ 25 X.3 และ X.28 95
รูปที่ 26 X.25 และ X.29 95

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญตาราง


| | | หน้า |
|---|---|---|
| ตารางที่ 1 | CCITT OSI Model X-200 Series Recommendations | 2 |
| ตารางที่ 2 | Principles of Layering | 5 |
| ตารางที่ 3 | The OSI Physical layer | 10 |
| ตารางที่ 4 | The OSI Data Link Layer | 11 |
| ตารางที่ 5 | The OSI Network Layer | 15 |
| ตารางที่ 6 | The OSI Transport Layer | 19 |
| ตารางที่ 7 | The OSI Session Layer | 21 |
| ตารางที่ 8 | The OSI Presentation Layer | 24 |
| ตารางที่ 9 | The OSI Application Layer | 26 |
| ตารางที่ 10 | HDLC TABLE | 45 |
| ตารางที่ 11 | Packet types and their use in various services | 70 |
| ตารางที่ 12 | Diagnostic Code and DTE/DCE Interface States | 71 |
| ตารางที่ 13 | X.3 PAD parameter | 86 |
| ตารางที่ 14 | PAD Command Signal | 94 |
| ตารางที่ 15 | PAD Service Signals | 96 |


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 1

# THE OPEN SYSTEM INTERCONNECTION ( OSI )

ในปัจจุบันนี้ได้มีการผลิตเครื่องจักรกลขึ้นมาอย่างมากมาย สิ่งหนึ่งที่สำคัญมากเกี่ยวกับลักษณะการดำเนินการที่มีต่ออนาคตของการประมวลผลข้อมูล คือการกำหนดมาตรฐานที่จะให้เครื่องจักรกลของผู้ผลิตที่ต่างกัน และต่างประเทศกันสามารถติดต่อกันได้ ในขณะที่เริ่มต้นในการกำหนดให้เป็นมาตรฐานให้เป็นที่แน่นอนลงไปนั้น International Standard organization (ISO) ที่กรุงเจนิวา ประเทศสวิตเซอร์แลนด์ ได้มีการกำหนดโครงสร้าง (Architecture) ไว้เป็น 7-layer ด้วยกัน 7-ISO layers กำหนดลักษณะโครงสร้างโดยทั่วไป เรียกว่า Reference Model ในเรื่องของ Open System Interconnection (OSI Reference Model หรือ OSI Model) OSI Reference Model ได้แสดงไว้ในรูปที่ 1 ISO International Standard 7498 คือ แหล่งกำเนิดต้นฉบับที่เกี่ยวกับเอกสารของ OSI Reference Model ในปัจจุบัน OSI Reference Model เป็นเอกสารสนับสนุนที่สอดคล้องกับข้อเสนอแนะใน CCITT X.200 Series ด้วยเหมือนกัน (ดูตารางที่ 1) ISO และ CCITT Version ที่เกี่ยวกับ OSI Model เป็นพื้นฐานอย่างเดียวกันวัตถุประสงค์ประการแรกของ OSI Model คือกำหนดหลักเกณฑ์ให้เป็นระดับเดียวกัน ในการพัฒนาข้อกำหนดที่เกี่ยวกับการเชื่อมต่อระหว่างซิสเต็มต่างๆ โดยการใช้อุปกรณ์การสื่อสารข้อมูลให้มีความคล่องตัวขึ้นแผนงานของ CCITT เช่นเอกสารที่มีอยู่ใน X.Series ที่เกี่ยวกับข้อเสนอแนะข้อกำหนดและโปรโตคอล อย่างกว้าง สำหรับแต่ละชั้นของ Seven OSI Model Layers ซึ่งมีเอกสารสนับสนุนจำนวนมากพร้อมแล้วใน CCITT Red Book

| APPLICATION LAYER |
|---|
| TRANSPORT LAYER |
| PRESENTATION LAYER |
| SESSION LAYER |
| NETWORK LAYER |
| DATA LINK LAYER |
| PHYSICAL LAYER |

**รูปที่ 1 OSI Reference Model Layers**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **Recommendation X.200** : Reference Model of Open Systems Interconnection for CCITT Applications
- **Recommendation X.210** : Open System Interconnection (OSI) Layer Service Definition Conventions
- **Recommendation X.213** : Network Service Definition for Open Systems Interconnection (OSI) for CCITT Applications
- **Recommendation X.214** : Transport Service Definition for Open Systems Interconnection (OSI) for CCITT Applications
- **Recommendation X.215** : Session Service Definition for Open Systems Interconnection (OSI) for CCITT Applications
- **Recommendation X.224** : Transport Protocol Specification for Open Systems Interconnection (OSI) for CCITT Applications
- **Recommendation X.225** : Session Protocol Specification for Open Systems Interconnection (OSI) for CCITT Applications
- **Recommendation X.244** : Procedure for the Exchange of Protocol Identification during Virtual Call Establishment on Packet Switched Public Data Networks
- **Recommendation X.280** : Tormal Description Techniques for Data Communication Protocols and Services

**ตารางที่ 1 CCITT OSI Model X-200 Series Recommendations**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## CORPORATION FOR OPEN SYSTEMS

ผู้ผลิตเครื่องคอมพิวเตอร์จำนวนมาก บริษัทซอฟต์แวร์ต่างๆ และผู้แทนจำหน่ายด้านโทรคมนาคมทั้งหลายได้ดำเนินกิจกรรม รวมถึงในการพัฒนาที่เกี่ยวกับข้อกำหนดต่างๆ ที่สนับสนุน OSI อย่างต่อเนื่อง เพื่อที่จะได้ให้อุปกรณ์และระบบต่างๆ ต่อร่วมกันได้อย่างมีประสิทธิภาพดังนั้นจึงต้องทำให้สอดคล้องกับ OSI Model ในปี พ.ศ. 2528 ได้มีการแนะนำให้รู้จักกับ Corporation for Open System (COS) ซึ่งเป็นระบบหนึ่งในสหรัฐอเมริกาภายใต้การอุปถัมป์ของ Computer and Communi cation Industry Association องค์การต่างๆ หลายองค์การ รวมทั้ง ไอ บี เอ็ม และผู้ผลิตเครื่องคอมพิวเตอร์อื่นๆ จำนวนมากได้สมัครเข้าเป็นสมาชิกของ OSI และดำเนินการที่เกี่ยวข้องทั้งหมดกับข้อกำหนดต่างๆ ในพื้นที่ที่ได้มีการพัฒนาเกี่ยวกับ Integrated Services Digital Network (ISDN) ด้วยเช่นกัน ISDN Standards คือการรวบรวมประเภทต่างๆของทราฟฟิคทั้งหมดที่ประกอบด้วยเรื่องของเสียง , ข้อมูลและรูปภาพ ให้สามารถส่งผ่านอุปกรณ์สื่อสารอันเดียวกันได้

## INTERCONNECTION OF SYSTEMS

OSI Model เกี่ยวข้องกับการเชื่อมต่อระหว่างระบบต่างๆเข้าด้วยกัน เพื่อเป็นเส้นทางของการแลกเปลี่ยนที่ไม่ใช่ข่าวสารข้อมูลหน้าที่ต่างๆภายในที่ดำเนินการนั้นจะถูกกำหนดโดยระบบ ใน OSI Terminology ระบบ หมายถึง *"เครื่องคอมพิวเตอร์จำนวนหนึ่งหรือมากกว่าที่ประกอบด้วย ซอฟต์แวร์,อุปกรณ์นำเข้า/ส่งออก,เทอร์มินอล,ผู้ปฏิบัติงาน,กระบวนการทางฟิสิคอล,วิธีการส่งถ่ายและอื่นๆ ดังนั้น วิธีการต่างๆ สามารถดำเนินการทั้งหมดเองได้โดยอิสระในเรื่องของการประมวลผลข้อสนเทศ และ/หรือ การส่งถ่ายข่าวสารข้อมูล "*

OSI Model ได้ให้ข้อคิดเห็นโดยทั่วๆไปที่เกี่ยวกับโครงสร้างที่แบ่งออกเป็นชั้นๆ โดยให้คำจำกัดความไว้อย่างกว้างที่ได้กำหนดไว้สำหรับระบบโครงสร้างนี้สามารถนำมาประยุกต์ใช้กับระบบได้โดยตรงเช่นการเชื่อมต่อเทอร์มินอลเข้ากับคอมพิวเตอร์ที่มีการแบ่งประเภทที่มีความซับซ้อนมากเช่น การต่อระหว่างโครงข่ายคอมพิวเตอร์สองโครงข่ายเข้าด้วยกัน สามารถนำเอารูปแบบจำลองมาใช้สำหรับโครงสร้างที่เกี่ยวกับโครงข่ายได้เช่นเดียวกัน การพัฒนาที่เกี่ยวกับ OSI Model ยังคงดำเนินการต่อไป สำหรับบางพื้นที่ข้อกำหนดทางมาตรฐานยังคงต้องมีการพัฒนาอีก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีแนวความคิดพื้นฐานบางอย่าง ที่เป็นฐานของโครงสร้างที่แบ่งออกเป็นชั้น ๆ ของ OSI Model สิ่งแรกคือเกี่ยวกับ Entity ซึ่งเป็นอุปกรณ์ที่ดำเนินการอยู่ภายในชั้น Entity 2 ชุดที่อยู่ภายในชั้นเดียวกันเรียกว่า Peer Entities Entity ต่างๆ ที่อยู่ในชั้นหนึ่งจะให้บริการต่างๆ กับ Entity ต่างๆที่อยู่ในชั้นที่เหนือขึ้นไปและในทางกลับกันก็จะได้รับบริการต่างๆ จากชั้นที่ต่ำกว่า เช่น Entity ต่างๆใน Presentation Layer ให้บริการต่างๆ กับ Application Layer และจะได้รับบริการต่างๆ จาก Session Layer ปัจจัยที่สำคัญของคำจำกัดความของชั้น คือ กำหนดการบริการต่างๆที่จะให้ นอกจากนั้นแต่ละชั้นต้องดำเนินการกำหนดหน้าที่ต่างๆที่จะให้การบริการกับชั้นของตัวเองและให้บริการกับชั้นที่สูงกว่า

## SERVICE-DATA-UNITS

ยูนิตต่างๆ ของข้อมูลที่ได้จัดขึ้นในชั้นต่างๆ นั้นได้ถูกกำหนดโดย OSI Model เช่น Service-data-units ชื่อของ Service-data-unit แต่ละยูนิตเริ่มต้นด้วยชื่อของชั้นนั้นๆที่มียูนิตต่างๆสมทบอยู่ Service-data-units ที่จัดขึ้นโดย Physical Layer เรียกว่า Physical-service-data-units Service-data-unit ที่จัดขึ้นโดย Data-link-layer เรียกว่า Data-link-service-data-units และต่อไปเรื่อยๆ ยูนิตข้อมูลต่างๆ ที่จัดขึ้นโดย Lower-level Layers มักมีชื่อเรียกไม่เป็นไปตามที่กำหนดในหลายๆ กรณีด้วยกันดังที่ได้อ้างถึงในเรื่องที่เขียนเกี่ยวกับเรื่องใดเรื่องหนึ่งโดยเฉพาะ เช่น Data-link-service-data-units มักเรียกว่า เฟรม บ่อยๆ และ Network-service-data-units มักเรียกว่า packet แทน

## DEFINING THE LAYERS

CCITT Study Group มีหน้าที่รับผิดชอบเกี่ยวกับการกำหนด OSI Reference Model เริ่มต้นโดยการกำหนดหลักการต่างๆ เพื่อเป็นแนวทางการพัฒนาเกี่ยวกับโครงสร้าง หลักการต่างๆ เหล่านี้ได้แสดงไว้ในตารางที่ 2 ถึงแม้ว่า มันจะไม่สามารถพิสูจน์ได้ว่า 7 ชั้น ที่ถูกเลือกขึ้นมาเป็นตัวอย่างนั้นจะแก้ปัญหาได้ดีที่สุดที่เป็นไปได้ หลักการต่างๆโดยทั่วไปที่ได้แสดงในตารางที่ 2 เป็นแนวทางที่ CCITT Study Group ได้กำหนดขึ้นในการตอบคำถามในเรื่องของการแบ่งชั้นว่า ควรจะแบ่งออกเป็นกี่ชั้น ซึ่งได้กำหนดไว้เป็น 10 หลักการด้วยกัน สิ่งที่จะกล่าวต่อไปนี้เป็นพื้นฐานแห่งเหตุผลที่ได้ใช้ในการกำหนดของชั้นแต่ละชั้นของ Seven OSI Layers :

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| |
|---|
| **Principle 1.** Do no create so many layers as to make the system engineering task of describing and integrating the layers more difficult than necessary. |
| **Principle 2.** Create a boundary at a point where the description for services con be small and the number of interactions across the boundary are minimized. |
| **Principle 3.** Create separate layers to handle functions that are manitestly different in the process performed or the involved technology. |
| **Principle 4.** Collect similar functions in the same layer. |
| **Principle 5.** Select boundaries at a point that past experience has demonstrated to be successful. |
| **Principle 6.** Create a layer of easily localized functions so that the layer could be totally redesigned and its protocols changed in a major way to take advantage of new advances in architectural, hardware, or software technology without changing the expected services from and provided to the adjacent layers. |
| **Principle 7.** Create a boundary where it may be useful at some point in time to have the corresponding interface standardized. |
| **Principle 8.** Create a layer where there is a need for a different level of abstraction in the handling of data. |
| **Principle 9.** Allow changes of functions or protocols to be made withing a layer without affecting other layers. |
| **Principle 10.** Create for each layer boundaries with its upper and lower layers only. |

**ตารางที่ 2 Principles of layering**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Physical Layer

ได้ถูกกำหนดขึ้นโดย OSI Model ซึ่งได้มีการวางแผนไว้ล่วงหน้าเกี่ยวกับประเภทของเครื่องมือทางด้านฟิสิคอลไว้อย่างกว้างๆ เพื่อสนับสนุนการต่อร่วมที่ใช้ในการควบคุมให้เหมาะสม โดยการนำหลักการที่ 3,5 และ 8 มาประยุกต์ใช้เพื่อเป็นแนวทางการวินิจฉัยในเรื่องของ Physical Layer ซึ่งเป็นชั้นต่ำสุดในโครงสร้าง

- Data Link layer

มีบ้างที่เครื่องมือทางด้านฟิสิคอลที่ต้องการเทคนิคบางอย่างเพื่อไม่ให้เกิดข้อผิดพลาดสำหรับการส่งถ่ายข้อมูลที่ผ่านจุดเชื่อมต่อที่มีอัตราการผิดพลาดค่อนข้างสูง Data Link Protocols สำหรับควบคุมการส่ง เพื่อไม่ให้เกิดข้อผิดพลาดนั้นมีอยู่แล้ว และจะต้องกำหนดเกี่ยวกับการใช้โปรโตคอลให้เหมาะสมกับโครงสร้างนี้ โดยเฉพาะจะต้องยอมรับว่าโครงสร้างควรจะปรับตัวให้เข้ากับเครื่องมือสื่อสารทางฟิสิคอลใหม่ๆ ซึ่งอาจจะต้องการขั้นตอนการควบคุมใหม่ๆด้วย โดยการนำเอาหลักการที่ 3,5 และ 8 มาประยุกต์ใช้เพื่อให้เป็นแนวทางการวินิจฉัยในเรื่องของ Data Link Layer ซึ่งเป็นชั้นที่อยู่บนของ Physical Layer ในโครงสร้าง

- Network Layer

เป็นที่ยอมรับกันแล้วว่า ระบบต่างๆบางระบบจะกระทำหน้าที่เช่นเดียวกับจุดหมายปลายทางของข้อมูล ในขณะเดียวกันก็ทำหน้าที่คล้ายกับ Intermediat Nodes เพื่อส่งผ่านข้อมูลไปยังจุดหมายปลายทางของตัวเอง โดยการนำเอาหลักการที่ 3,5 และ 7 มาใช้เพื่อเป็นแนวทางการวินิจฉัยในเรื่องของ Network Layer ที่อยู่บนของ Data Link Layer. Network Oriented Protocols เช่น ตัวอย่างการกำหนดเส้นทางก็เป็นชุดใน Network Layer. Network Layer จะกำหนดการเชื่อมต่อเส้นทางระหว่างระบบต่างๆ รวมถึงกรณีซึ่งมี Intermediat Nodes รวมอยู่ด้วย

- Transport Layer

องค์ประกอบของ Transport Layer เป็นชั้นสูงสุดที่เกี่ยวกับการให้บริการการขนส่งข้อมูลที่กำหนดโครงสร้าง Transport Layer ทำหน้าที่แยก Higher-level Entities ออกจากส่วนที่เกี่ยวข้องกับการขนส่งข้อมูลจากระบบหนึ่งไปยังอีกระบบหนึ่ง และสร้างโครงข่ายโดยตรงให้กับผู้ใช้ต่างๆของโครงข่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**- Session Layer**

ระบบของ Controlling และ Synchronizing สำหรับการโต้ตอบที่เกิดขึ้นระหว่างระบบต่างๆและการดำเนินการที่เกี่ยวกับการแลกเปลี่ยนข้อมูลนั้นเป็นสิ่งจำเป็นซึ่งเป็นที่ยอมรับ โดยการนำเอาหลักการที่ 3 และ 4 มาใช้เพื่อเป็นแนวทางการวินิจฉัยในเรื่องของ Session Layer ที่อยู่ชั้นเหนือกว่า Transport Layer

**-Presentation Layer**

หน้าที่ต่างๆ ที่ยังมีอยู่อีกเว้นแต่ว่าเป็นประเภททั่วไปและไม่ได้กำหนดการใช้งานโดยเฉพาะเพราะเกี่ยวข้องกับการเป็นตัวแทนและการเปลี่ยนแปลงให้เหมาะสมกับข้อมูลที่สร้างขึ้นสำหรับการนำโปรแกรมไปใช้งานให้เป็นประโยชน์ โดยการนำเอาหลักการที่ 3 และ 4 มาใช้เพื่อเป็นแนวทางการวินิจฉัยในเรื่องของ Presentation Layer ที่อยู่ชั้นเหนือกว่า Session Layer

**- Application Layer**

ขั้นตอนต่างๆ ที่นำมาประยุกต์ใช้เป็นประโยชน์กับการบริการต่างๆ ที่เกี่ยวกับชั้นทั้ง 6 ที่อยู่ต่ำลงไปและโปรโตคอล ซึ่งใช้สื่อสารนั้นได้สร้างขึ้นที่ Application Layer ซึ่งเป็นชั้นสูงสุดของโครงสร้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 1.1 THE PHYSICAL LAYER

Physical layer มีหน้าที่เกี่ยวกับการส่งผ่านของ Physical sevice data units (จำนวนบิทที่ใช้จริง) โดยเฉพาะที่ผ่านอุปกรณ์การส่งทางฟิสิคอลที่ต่ออยู่กับ Data Link Entities สองหรือมากกว่า รวมถึงการเชื่อมต่อระหว่างเครื่องคอมพิวเตอร์ 2 เครื่อง ที่มีการแลกเปลี่ยนสัญญาณไฟฟ้าระหว่างกัน ตัวอย่าง ฮารด์แวร์ ประกอบด้วยเคเบิล,ตัวต่อที่จัดไว้โดยเฉพาะ,อุปกรณ์การสื่อสาร 2 ชุดซึ่งสามารถใช้ได้ทั้งเป็นตัวกำเนิดและตรวจจับแรงดันไฟฟ้าที่มีเคเบิลเชื่อมต่ออยู่ด้วย ซอฟต์แวร์ ประกอบด้วย เฟิร์มแวร์ (Firmware) ที่ติดตั้งอยู่ในอุปกรณ์ต่างๆ เพื่อควบคุมการกำเนิดและตรวจจับแรงดันไฟฟ้าเหล่านี้ ตารางที่ 3 แสดงรายการบริการต่างๆ ที่ Physical layer จัดให้กับ Data Link layer และกล่าวถึงหน้าที่บางอย่างที่ Physical layer ดำเนินการอยู่

โปรโตคอลสำหรับ Physical Layer ที่เกี่ยวข้องกับทางกล, ทางไฟฟ้า, การทำงานตามหน้าที่และขั้นตอนปฏิบัติต่างๆ สำหรับ Activating, Maintaining และ Deactivating การเชื่อมต่อทางฟิสิคอล สิ่งนี้รวมถึงข้อกำหนดของตัวต่อทางฟิสิคอล, เคเบิลและสัญญาณไฟฟ้าต่างๆ ความจริงข้อนี้มาตรฐานต่างๆทางฮาร์ดแวร์มีความสำคัญมาก เครื่องคอมพิวเตอร์ต้องมีตัวต่อที่ติดตั้งไว้ในตำแหน่งที่จัดไว้ให้โดยเฉพาะ ซึ่งเคเบิลสามารถต่อเข้าได้ง่าย เคเบิลต้องมีตัวต่อที่จัดไว้โดยเฉพาะประกอบติดเข้ากับปลายของเคเบิลแต่ละปลาย เทอร์มินอลก็เช่นเดียวกันก็ต้องมีตัวต่อติดตั้งไว้ตามที่กำหนดไว้โดยเฉพาะ เพื่อสะดวกในการต่อเข้ากับเคเบิล โดยเฉพาะเครื่องคอมพิวเตอร์กับเทอร์มินอลต้องมีการกำหนดเกี่ยวกับระดับแรงดันไฟฟ้าที่จะใช้ส่งสัญญาณต่างๆ ระหว่างอุปกรณ์ทั้งสองกันเสียก่อน ข้อกำหนดต่างๆ สำหรับซอฟต์แวร์พื้นฐานเป็นสิ่งสำคัญมาก โดยเฉพาะที่เกี่ยวกับเฟิร์มแวร์ที่ใช้ดำเนินการใน Physical layer เช่นตัวอย่าง เฟิร์มแวร์ที่ดำเนินการในเครื่องคอมพิวเตอร์ และเฟิร์มแวร์ที่ดำเนินการในเทอร์มินอลต้องตกลงกันเสียก่อนว่า ช่วงระยะเวลาของแต่ละบิทจะนานเท่าใด และจะจำแนกถึงความแตกต่าง ระหว่างบิท 1 กับบิท 0 ได้อย่างไร Physical layer ต้องส่งบิทต่างๆ ในขั้นตอนอย่างเดียวกันดังที่ได้กล่าวมาแล้ว และจะต้องรายงานถึงความผิดพลาดทั้งหมดที่ตรวจพบให้กับชั้นที่สูงถัดไปทราบการเชื่อมต่อทางฟิสิ คอ ลอาจรวมถึงการเชื่อมต่อร่วมของวงจรข้อมูลที่ต่อแบบอนุกรมด้วย วิธีการส่งที่ดำเนินการโดย Physical layer มีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การเชื่อมโยงแบบ Duplex หรือ Half-duplex
- การเชื่อมต่อแบบ Point-to-point หรือ Multipoint
- การส่งแบบ Synchronous หรือ Asynchronous

ถ้านำอุปกรณ์ของผู้ใช้มาใช้กับวงจรแอนาลอกเช่นวงจรโทรศัพท์ที่มีใช้กันอยู่เดิมอุปกรณ์จะต้องต่อเข้ากับโมเด็ม การเชื่อมโยงอุปกรณ์เข้ากับโมเด็มต้องทำตามข้อกำหนด ซึ่งเป็นที่ยอมรับกันทั่วไป เช่น EIA RS-232-C และ RS-449 Standard หรือ CCITT Rec. V.24 ถ้าหากว่านำมาใช้กับวงจรดิจิตอล ข้อเสนอแนะสำหรับการเชื่อมโยงทางฟิสิคอลต้องเปลี่ยนใหม่ เช่น CCITT Rec. X.21 ที่กล่าวถึงการเชื่อมโยง เพื่อให้เทอร์มินอลทำงานในลักษณะ Synchronous Mode ส่งผ่าน Circuit-switched Network

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

035034

**Services to the Data Link Layer**

- **Physical Connections** : transparent transmission of bit streams across physical connections
- **Physical Service-Data-Units** : definition of the unit of data that is handled by the physical layer (one bit for serial transmission and n bits for paralleld transmission)
- **Physical Connection End Points** : definition of identifiers that are used by the data link layer to refer to physical connection endpoints
- **Data Circuit Identification** : definition of identifiers that are used by the data link layer to refer to physical circuits
- **Sequecing** : delivery of bits in the order in which they were transmitted
- **Fault Condition Notification** : notification of the data link layer when fault conditions occur
- **Quality -of-Service parameters** : definition of parameters that define quality of service, including error rate, service availability, transmission rate, and transit delay

**Physical Layer Functions**

- Activating and deactivating physical connections upon request from the data link layer.
- Transmitting bits over a physical connection in a synchronous or asynchronous fashion.
- Handling physical layer management activities, including activation and error control.

**ตารางที่ 3 The OSI Physical layer**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 1.2 THE DATA LINK LAYER

Data Link layer มีหน้าที่เกี่ยวกับการจัดส่งข้อมูลที่เชื่อถือได้จาก Network Entity หนึ่ง หรือ Physical Node ไปให้กับที่อื่นๆและให้ความคุ้มครองชั้นต่างๆที่สูงกว่า เนื่องจากอุปกรณ์การส่งทางฟิสิคอลที่เกี่ยวข้องทั้งหมด ทำให้การส่งข้อมูลของ Data-link-service-data-units หรือเฟรมปราศจากข้อผิดพลาด ตารางที่ 4 เป็นรายการบริการต่างๆ ที่ Data Liink Layer ให้กับ Network layer และกล่าวถึงหน้าที่บางอย่างของ Data Link layer ที่ดำเนินการในการให้บริการเหล่านั้น ชั้นนี้มีหน้าที่ที่เกี่ยวข้องกับ เช่น

- ต้องกำหนดลงไปว่า แต่ละบล็อกที่ส่งออกไปเริ่มต้นและสิ้นสุดที่ไหน
- เมื่อปรากฏว่าการส่งมีข้อผิดพลาดเกิดขึ้น ก็จะต้องทำให้กลับคืนสู่สภาพเดิมเพื่อให้การสื่อสารเป็นไปอย่างมีประสิทธิภาพ

มาตรฐานของโปรโตคอลที่ได้พัฒนาขึ้นมาสำหรับ Data Link Layer นั้น คือ High Level Data Link Control (HDLC) HDLC เป็นแบบ Bit-Oriented เป็น Synchronous Protocol สำหรับการส่งข้อมูล SDLC Protocol เป็นมาตรฐานโปรโตคอลสำหรับการเชื่อมโยงข้อมูลที่กำหนดโดย SNA ตามความเป็นจริงแล้ว SDLC เป็นชุดประกอบของ HDLC Protocol LAP-D และ LAP-B Data Link Protocol นั้น เป็นตอนหนึ่งของ CCITT Rec. X.25 ซึ่งสอดคล้องกันกับ Data Link-Layer ของ OSI Model และเป็นชุดประกอบของ HDLC ด้วยเช่นเดียวกัน

| Services to the Network Layer |
| --- |
| - **Data Link Connection** : establishment of a data link connection between two or more entities defined by the network layer (data link connections are established and released dynamically) |
| - **Data Link Service-Data-Units** : definition of the unit of data handled by the physical layer (generally called a frame consisting of a sequence of bits) |

**ตารางที่ 4 OSI Data Lnk Layer Functions**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **Data Link Connection Endpoint Identifiers** : definition of identifiers that are used by the network layer to refer to data link connection end points
- **Sequencing** : delivery of frames in the same order in which they were transmitted
- **Error Notification** : notification of the network layer when unrecoverable errors occur
- **Flow Control** : dynamic control over the rate at which a network entity receives frames from a data link connection
- **Quality-of-Service parameters** : definition of parameters that define quality of service, including medan time between unrecoverable errors, residual error rate, serviice availability, throughput, and transit delay

**Data Link Layer Functions**

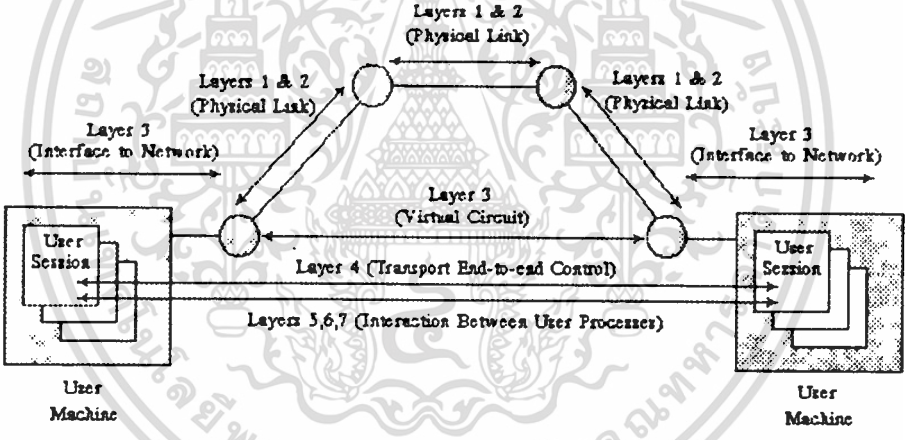
- Establishing and releasing data link connections for use by the network layer
- Building a data link connection using one or more physical connections
- Delimiting data so that it can be sent as frames over a physical connection
- Synchronizing the receipt of data that has been split over multiple physical connections
- Maintaining the sequential order of frames that are transmitted over a data link connection
- Detecting and correcting transmission error with retransmission of frames if necessary
- Proviiding flow control including dynamically altering the rate at which data units are accepted, and temporanly stopping transmission to a particular receiving entity upon request

**ตารางที่ 4 OSI Data Lnk Layer Functions (ต่อ)**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 1.3 THE NETWORK LAYER

Network layer มีความเกี่ยวข้องกับ Virtual Circuits เส้นทางระหว่างคอมพิวเตอร์ จะเกิดขึ้นในชั่วระยะเวลาหนึ่ง อาจผ่าน Physical Communication Links หลายลิงค์ด้วยกันดังแสดงในรูปPhysical Link แต่ละลิงค์ที่วางอยู่ระหว่างเครือข่ายคอมพิวเตอร์ 2 ชุด ซึ่งจะต้องใช้อย่างน้อยที่สุดก็คือ Physical และ Data Link สำหรับการดำเนินการเพื่อแลกเปลี่ยนข้อมูลผู้ใช้ที่ปลายด้านใดด้านหนึ่งไม่จำเป็นต้องทราบเส้นทางที่ข้อมูลเดินทางผ่าน หรือ ข้อมูลเดินทางผ่าน Physical Data Link กี่ลิงค์ ผู้ใช้เครื่องต้องการการเชื่อมโยงโดยตรงกับ Virtual Circuits เท่านั้น Network Layer จะควบคุม การทำให้เกิด Virtual Circuit และจัดให้มีการเชื่อมโยงชั้นที่สูงกว่าเข้ากับตัวเองเท่านั้น




รูปที่ 2 ทางเดินระหว่าง User Machines

เกี่ยวกับกฎเกณฑ์บางอย่าง เส้นทางที่กำหนดให้ข้อมูลเดินทางระหว่างผู้ใช้ทั้งสองนั้นอาจเปลี่ยนจากเส้นทางหนึ่งไปเป็นอีกเส้นทางหนึ่งได้ทันที โดยไม่มีแผนล่วงหน้า โครงข่ายของเครื่องอาจจะต้องแบ่งข้อมูลออกเป็นชิ้นเล็กๆ ที่เรียกว่า แพคเกท ให้มีความยาวตามที่กำหนด หลังจากส่งไปแล้วแพคเกทต่างๆเหล่านั้นต้องประกอบกลับคืนเป็นข้อมูลเดิมได้เกี่ยวกับโครงข่ายต่างๆ บางโครงข่าย แพคเกทต่างๆอาจไม่เรียงกันไปตามลำดับ ระหว่างการส่งข้อกำหนดต่างๆสำหรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Network Layer State นั้น ต้องส่งแพคเกทต่างๆไปให้กับผู้ใช้ในขั้นตอนอย่างเดียวกันกับต้นทางที่ส่งมามีความซับซ้อนอยู่หลายประการเกี่ยวกับวิธีการและการทำงานของ Virtual Circuit. Network Layer จะกำหนดมาตรฐานการเชื่อมโยงที่เกี่ยวกับ Virtual Circuit ที่เป็นไปได้รวมถึงส่วนประกอบต่างๆ ของโครงสร้างที่เกี่ยวกับการทำงานของตัวเองจากซอฟต์แวร์ในชั้นที่สูงกว่า

การกำหนดที่เหมาะสมที่เกี่ยวกับ Network Layer State นั้น ที่ชั้นนี้จะรับผิดชอบในเรื่อง Establishing, maintaining และ Terminating การเชื่อมต่อโครงข่ายระหว่าง Transport Entities ทั้งสอง และที่เกี่ยวกับการส่งถ่ายข้อมูลในระหว่างการดำเนินการอยู่นั้นจะต้องให้การคุ้มครองกับชั้นที่สูงกว่าทั้งหมด เนื่องจากสิ่งที่เกี่ยวข้องที่อยู่รอบข้างหรือต้องเข้าใจคือถึงลักษณะของการเชื่อมต่อนั้นๆ ตลอดเวลาที่กำหนดจะมีเพียงโครงข่ายเดียวเท่านั้นที่สามารถเชื่อมต่อกันระหว่าง Entities ทั้งสองที่กำหนดให้ ถึงแม้ว่าจะสามารถมีได้หลาย ๆ Physical Route ระหว่าง Transport Entities ทั้งสองถ้าหากเส้นทางที่กำหนดเกี่ยวพันกับ Multiple Network , Network Layer จะกำหนดเส้นทางของข้อมูลที่ส่งผ่านโหนดระหว่างทางทั้งหมดที่จำเป็น Network Layer ให้ความคุ้มครองชั้นที่สูงกว่า เนื่องจากเกี่ยวกับภาวะที่แตกต่างกันในอุปกรณ์ส่งผ่านต่างๆ Network Layer ที่ให้กับ TransportLayer และหน้าที่ต่างๆอีกหลายอย่างที่ดำเนินการในการให้บริการเหล่านั้นได้แสดงไว้ในตารางที่ 5

ปกติกฎเกณฑ์ที่นำมาใช้นั้นเพื่อให้สอดคล้องกันกับแนวทางที่ได้กำหนดโดย CCITTRec. X.25 level 3 สำหรับ Network Layer ข้อเสนอแนะของ X.25 มีอยู่ 3 ระดับ ซึ่งแต่ละระดับมีการเตรียมการในแต่ละชั้นไม่เหมือนกัน Level 1 มีการเตรียมการเกี่ยวกับ Physical Layer Level 2 เกี่ยวกับ Data Link Layer และ Level 3 บางทีก็เรียกว่า Packet Level เตรียมการเกี่ยวกับ Network Layer ข้อเสนอแนะใน X.25 ได้กล่าวถึงการเชื่อมโยง Packet Terminal เข้ากับ Packet-switched Network ได้อย่างไร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Services to the transport Layer**

- **Network Addresses** : definition of network addresses to be used in identifying entities defined by the transport layer
- **Network Connections** : transfer of data between transport entities that are identified by network addresses (each network connection is point-to-point, and multiple network connections can exist between transport entities)
- **Network Connection Endpoint Identifiers** : definition of unique end point idnetifier that are associated with the network addresses used by the transport layer
- **Network Service-Data-Unit Tranfer** : definition of the unit of data that is handled by the network layer (generally called a packet, on which no maximum size limit is imposed by the architecture)
- **Quality-of-Service paremeters** : definition of parameters that define quality of service, including residual error rate, service availability, throughput, transit delay, and connection-establishment delay : a selected quality of service is maintained for the duration of the network Connection
- **Error Notification** : notification of the transport layer when unercverable errors occur (error notification may or may not lead to the disconnection of network connection)
- **Sequencing** : delivery of frames in the same order in which they were transmitted if requested by the transport layer
- **Flow Control** : provision for stopping the transfer of packets upon request by the transport entity that is receiving packets Expedited Network Servie-Data-Unit Transfer : an optional, expedited service for handling information exchange over a network connection

**ตารางที่ 5 The OSI Network Layer**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **Reset** : an optional service than when requested by a transport entity, causes the network layer to discard all packets currently in transit and to notify the transport entity at the other end of the connection that a reset has occurred
- **Release** : release of a network connection upon request by a transport Entity

**Network Layer Functions**

- Determining an optimum routing over the possible network connections that con exist between two network addresses and then relaying packets over the various point-to-point connections that make up that route
- Providing a network connection between two transport entities and ransferring data over the network connection in a transparent fashion
- Multiplexing multiple network connections onto a single data link connection in order to optimize their use
- Segmenting and/or blocking packets for the purposes of facilitating the transfer of packets over netword connections
- Detecting errors and recovering from them
- Selecting an appropriate quality of service and maintaining this quality of service even when a network connection spans subnetworks of dissimilar quality
- Handling network layer management activities, including activation and error control

**ตารางที่ 5 The OSI Network Layer (ต่อ)**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 1.4 THE TRANSPORT LAYER

ชั้นต่ำสุด 3 ชั้น ของรูปที่ 1 นั้นแทนโครงข่ายร่วม ดังนั้น เครื่องจักรกลจำนวนมากสามารถร่วมส่วนแบ่งกันได้ซึ่งแต่ละเครื่องจะเป็นอิสระไม่เกี่ยวข้องกัน เช่นเดียวกับการบริการทางไปรษณีย์ ซึ่งผู้ใช้จำนวนมากอาจใช้ร่วมกันได้ อาจจะเป็นไปได้ การบริการทางไปรษณีย์อาจจะมีจดหมายสูญหายได้เพื่อให้แน่ใจได้ว่าสิ่งนี้จะไม่เกิดขึ้นอีกผู้ใช้ 2 คนที่ใช้บริการทางไปรษณีย์สามารถจะนำ End-to-End Control มาใช้เป็นของตัวเองได้ เช่น การให้หมายเลขกับจดหมายเหล่านั้นเรียงตามลำดับ Transport Layer มีความเกี่ยวข้องทำนองเดียวกันกับ End-to-End Control ของการส่งผ่านระหว่างผู้ใช้ทั้งสองที่เกิดขึ้นในช่วงของการดำเนินการ

ภาพประกอบในรูปที่ 2 ด้วยเหตุที่ว่า Network Layer ที่เกี่ยวข้องกับการเชื่อมโยงระหว่างผู้ใช้กับโครงข่าย. Transport Layer (และชั้นที่สูงกว่า) เกี่ยวข้องกับ End-to-End Interaction ระหว่างผู้ใช้เครื่องหน้าที่ต่างๆ ที่ได้ดำเนินการใน Transport Layer อาจรวมอยู่ใน End-to-End ที่มีการควบคุมร่วมกัน เพื่อป้องกันการสูญหาย หรือประมวลผลเป็น 2 เท่า ของสิ่งที่ดำเนินการควบคุมขบวนการปฏิบัติการเกี่ยวกับสิ่งที่ดำเนินการ และแอดเดรสซึ่งของผู้ใช้ปลายทางหรือระบบต่างๆ Trans-port Layer ร่วมกันกับ 3 ชั้นล่างจะให้บริการเกี่ยวกับการขนส่ง ทั้งหมดนี้จะเกี่ยวข้องกับการขนส่งข่าวสารจากผู้ใช้ระบบหนึ่งไปยังผู้ใช้คนอื่นๆ แต่ทั้งหมดนี้จะไม่เปลี่ยนแปลงข้อมูลที่ประกอบในรูปของข่าวสาร

การบริการการขนส่ง อาจดำเนินการได้หลายวิธี OSI Model ไม่ได้กำหนดรายละเอียดของวิธีการเหล่านี้เอาไว้ ในบางกรณีอาจนำรูปแบบของ Packet Switching Network มาใช้ตามข้อเสนอแนะของ X.25 สำหรับ ชั้นที่ 1,2 และ 3 เมื่อนำเอา X.25 มาใช้ Transport Layer ต้องแบ่งข้อมูลออกเป็นแพคเกทเล็กๆ สำหรับการส่งผ่านโครงข่าย ในกรณีอื่น ถ้านำเอา Point-to-Point Circuis มาใช้ สามารถนำมาใช้กับ Circuit Switching ได้ และยังสามารถใช้ได้กับ Satellite Circuit อีกด้วย เมื่อเอาเทคนิคต่างๆ เหล่านี้มาใช้ ข่าวสารที่จะส่งก็ไม่จำเป็นต้องแบ่งออกเป็นหลายๆ แพค เก็ตอีกการเชื่อมโยงจากชั้นที่สูงกว่า หรือ จากผู้ใช้เข้ากับ Transport Layer นั้นความมุ่งหมายเพื่อกำหนดมาตรฐานการเชื่อมโยงกับผู้ใช้ทั้งหลาย ในเรื่องของการบริการการขนส่งข่าวสารไม่ขึ้นอยู่กับประเภทของโครงข่ายที่ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการกำหนดรูปแบบของ OSI Model Transport Layer มีหน้าที่เกี่ยวกับการส่งถ่ายของ Transport-service-data-units ระหว่าง 2 Session Entities ที่ได้กำหนดในเรื่องคุณภาพของการบริการ Transport Layer ไม่เกี่ยวข้องกับการกำหนดเส้นทางที่ใช้ตลอดโครงข่าย ที่เกี่ยวข้องโดยตรงมี Reliable, cost-effective Transfer ของข้อมูล Transport Layer มีหน้าที่เกี่ยวกับการกำหนดการเชื่อมต่อสำหรับการขนส่งระหว่าง entities การส่งถ่ายข้อมูลและการปลดวงจรการเชื่อมต่อ

Transport Layer ได้ให้ข้อกำหนดในเรื่องของ Service Classes ไว้อย่างชัดเจน Service Classes จัดให้มีการรวมกันของแฟคเตอร์ต่างๆ เช่น Throughput , Transit Delay , Connection Setup Delay , Error Rate และเท่าที่จะหาได้ เมื่อการเชื่อมต่อการขนส่งได้ถูกกำหนดขึ้นแล้วโดย Session Entity. Transport Layer มีหน้าที่สำหรับมอนิเตอริ่งการส่งผ่านเพื่อให้แน่ใจว่าคุณภาพของบริการมีความเหมาะสมที่จะดำเนินการต่อไปได้ และถ้าไม่เหมาะสมที่จะดำเนินการต่อไปต้องรายงานกับ Entity ที่จัดไว้ โดยเฉพาะในเรื่องของความผิดพลาดหรือการลดลงของสัญญาณที่เกิดขึ้นการบริการที่เตรียมการโดย Transport Layer จะส่งต่อให้กับ Session Layer และหน้าที่อีกหลายอย่างที่ได้ดำเนินการโดย Transport Layer ดังได้สรุปไว้ในตารางที่ 6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Services to the Session Layer**

- **Transport Connection Establishment** : establishment of a transport connection, of an agreed-upon quality of service, between two entities defined by the seession layer (at the time of connection establishment, a class of service can be selected from other session entity is also notified of the connection release)
- **Data Transfer** : transfer of data in accordance with the quality of service negotiated when the connection was established
- **Transport Connection Release** : release of a transport connection upon request by a session entity (the other session entity is also notified of the connection release)

**Transport Layer Functions**

- Converting transport addresses for entities into network addresses; multiple tansport transport addresses might be mapped onto a single network address
- Multiplexing multiple transport connections onto a single network connection or spliting a transport connection over multiple network connections as needed to optimize use of network connections
- Sequencing of data units transfered to ensure that they are delivered in the same sequence in which they were sent
- Detecting and recovering from errors Segmenting blocking, and concatenation of data units
- Controlling data flow to prevent overloading of network resources
- Handling transport layer supervisory activities
- Providing for expedited transfer of data units between session enties

**ตารางที่ 6 OSI Transport Layer**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 1.5 THE SESSION LAYER

มีหน้าที่เกี่ยวกับการกำหนด Session ระหว่างผู้ใช้ที่ประกอบด้วยส่วนต่างๆที่ซับซ้อน เนื่องจากมีวิธีการที่ต่างกันอยู่หลายประการซึ่งเครื่องจักรกลสามารถทำงานร่วมกันได้เหมือนกัน กับผู้จัดการธุรกิจสองคนได้ตกลงที่จะลงทุนร่วมกันซึ่งทั้งสองจะต้องมีการตกลงกันก่อนเกี่ยวกับ กฎเกณฑ์ต่างๆของเกมในทางปฏิบัติทั้งสองจะต้องเซ็นสัญญากำหนดวิธีการกระทำซึ่งทั้งสองจะ ร่วมมือกัน Session Layer กำหนดขั้นตอนการปฏิบัติที่เกี่ยวกับการรกำหนดและยุติเกี่ยวกับ Session ถ้าหากว่ามีบางสิ่งดำเนินการผิดพลาดในช่วงของ Midsession Session Layer จะต้อง ดำเนินการให้กลับคืนสู่สภาพปกติโดยไม่ทำให้ข้อมูลสูญหาย หรือถ้าหากเป็นไปไม่ได้จะต้องยุติ Session ให้สอดคล้องกับขั้นตอนตามลำดับ Error-Checking ก็เป็นอีกวิธีการหนึ่งคือเมื่อตรวจพบ ว่ามีข้อผิดพลาดเกิดขึ้นต้องดำเนินการให้กลับคืนสู่สภาพเดิมทันทีซึ่งเป็นสิ่งที่จะต้องดำเนิน การในชั้นนี้

กฎต่างๆ สำหรับการโต้ตอบที่ดำเนินการอยู่มีอะไรบ้างที่ต้องกำหนดในขณะที่ Session ได้ถูกกำหนดขึ้นในรูปแบบบางอย่างของ Session การโต้ตอบที่เกิดขึ้นระหว่างเครื่องจักรกล ทั้ง สองในขณะเดียวกันโปรโตคอลก็ต้องกำหนดว่าใครจะเป็นผู้พูดก่อนและนานเท่าใดในบางกรณี เครื่องจักรทั้งสองจะทำการแบบผลัดกันส่งและรับในกรณีอื่นเครื่องจักรกลทางด้านหนึ่งอาจ ส่ง ข้อมูลที่ต่อเนื่องกันได้ก่อนที่จะอนุญาตให้สถานีตรงข้ามตอบก็ได้ Session บางอย่างเครื่องจักร กลทางด้านหนึ่งอาจจะระงับการส่งของสถานีตรงข้ามได้แต่ในบางเครื่องก็ไม่สามารถกระทำได้

Session Layer เน้นในเรื่องของการให้บริการต่างๆ ที่ได้กำหนดไว้แน่นอนนั่นคือ การจัด ระบบและทำให้สอดคล้องกับการโต้ตอบที่เกิดขึ้นระหว่าง Presentation Entities และควบคุม Session-service-data-units ที่มีการแลกเปลี่ยนระหว่างกัน ถึงแม้ว่าจะมีข้อมูลสูญหายไปที่ Transport level. Session layer ยังคงรักษาการโต้ตอบนั้นไว้ Session Layer มีหน้าที่รับผิดชอบ เกี่ยวกับ Establishing, managing และ Releasing Session การเชื่อมต่อระหว่าง Presentation Entities การเชื่อมต่อของ Session ระหว่างคู่ของ Entities ต่างๆที่กำหนดให้ สามารถมีจุดร่วมกัน ได้หลายจุด Session Con-nection สามารถทำแผนผังเชื่อมกับ Transport Connection ได้ สมมติ ว่าเช่นตัวอย่าง Transport Connection ต้องยุติลงเนื่องจากมี Error หรือข้อบกพร่องเกิดขึ้นในกรณี อื่น Session Connection ที่ต่ออนุกรมเรียงตามลำดับอาจจะ ใช้อย่างเดียวกันกับ Transport Connection ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากที่กล่าวมาแล้วตั้งแต่ต้น ที่เกี่ยวข้องอันดับแรกของ Session Layer คือควบคุมการโต้ตอบที่ทำงานร่วมกัน Session Layer จะกำหนดประเภทของการโต้ตอบไว้ 3 ประเภท คือ

- Two-way ที่กระทำได้พร้อมกัน ซึ่ง Entities ทั้งสองสามารถส่งและรับได้ในเวลาเดียวกัน (หรือแบบ Full-duplex)
- Two-way ผลัดกันกระทำ ซึ่ง Entity ทั้งสองจะผลัดกันรับและส่ง (หรือแบบ Half-duplex)
- One-way Interaction ซึ่งมี Entity หนึ่งเท่านั้นที่ส่งได้ และ Entity อื่นจะรับเพียงอย่างเดียวเท่านั้น (หรือแบบ Simplex)

เมื่อ Entities ต่างๆ ทำให้เกิดการส่งและรับสลับกันการเปลี่ยนแปลงจากภาวะหนึ่งไปยังอีกภาวะหนึ่งสามารถเกิดขึ้นเองซึ่งเป็นหน้าที่ของ Entities ต่างๆหรือสามารถผลักดันจาก Session layer ได้เมื่อ Presentation Entity ต้องการได้ ตารางที่ 7 แสดงรายการต่างๆ ที่ Session Layer ดำเนินการอยู่ ปัจจุบันยังมีหน้าที่ต่างๆ ที่เพิ่มขึ้นมาอีกที่ยังไม่ได้กำหนดไว้สำหรับ Session Layer แต่ก็ได้มีการพิจารณาถึงการเสนอตัวสำหรับการขยายต่อไปอีกในอนาคตซึ่งประกอบด้วยสิ่ง ต่อไปนี้

- Sequence Numbering ของ Session-service-data-untis
- Brackets
- Stop-go Transmission
- Security Functions

| Services to the Presentation Layer |
| --- |
| - **Session Connection Establishment** : establishment of a session connection between two entities defined by the presentation layer |
| - **Session Connection Release** : release of a session connection, in an orderly fashion and without loss of data, upon request of a presentation entity (this service permits either of the presentation entities to abort a session connection, in which case data might be lost ; the release of session connection can also be initiated by one of the session entities supporting it ) |

**ตารางที่ 7 The OSI Session Laye**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **Normal Data Exchange** : normal transfer of data between a sending presentation entity and a receiving presentation entity
- **Quarantine Service** : request by a sending presentation entity that a specific number of session data units be held and not made available to the receiving presentation entity until specifically released by the sending entity (the sending entity can later specify that the data units be discarded instead of being sent to the reciving entity)
- **Expedited Data Exchange** : transfer of data on an expedited basis for high-priority traffic
- **Interaction Management** : explicit control over whose turn is to exercise certain control functions
- **Session Connection Synchronization** : definition by presentation entities of synchronization points (the session layer provides services to reset a session connection to a resynchronization point and to restore entities to a defined state )
- **Exception Reporting** : notification to presentation entities by the session layer about exceptional conditions

**Session Layer Functions**

- Providing a one-to-one mapping between a session connection and a presentation connection at any given instant; over time, however, a transport connection can use several consecutive session connections, and several consecutive transport connections might use a single session connection
- Preventing a presentation entity from being overloaded with data by using transport flow control (there is no explicit flow control in the session layer)
- Restablishing a transport connection to support a session connection in the event of a reported failure of the underlying transport connection
- Handling session layer management activities

**ตารางที่ 7 The OSI Session Layer (ต่อ)**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 1.6 THE PRESENTATION LAYER

Presentation layer มีหน้าที่เกี่ยวข้องกับการเข้ารหัสข้อมูล และชุดตัวอักษรที่ประกอบด้วยตัวอักษร,ตัวเลข หรือ สัญลักษณ์พิเศษที่เครื่องพิมพ์หรือเทอร์มินอลสามารถแสดงได้ ชุดของตัวอักษรต่างๆที่มาถึงเทอร์มินอลจะมีขั้นตอนต่างๆที่แน่นอนและมีความหมายถูกต้อง ซึ่งจะปรากฏผลบนจอภาพ หรือเทอร์มินอลออกมาชุดของตัวอักษรต่างๆอาจบรรจุตัวอักษร, ตัวเลข หรือสัญลักษณ์ต่างๆ ที่นำมาเรียบเรียงเป็นข้อมูล, เปลี่ยนบรรทัดข้อมูล, การเลื่อนตำแหน่งข้อมูลในคอลัมน์, การเพิ่มหัวเรื่องของคอลัมน์, ทำให้ส่วนของข้อความเด่นชัดขึ้น, การให้แสงที่เหมาะสมและอื่นๆ อาจจะวางรูปแบบการเรียงพิมพ์ให้กับโอเปอร์เรเตอร์ใส่ข้อมูล โดยใส่ข้อมูลที่ต้องการส่งเข้าไปเท่านั้น ข้อมูลที่เข้ารหัสแล้วจะถูกส่งไปให้กับ Intelligent Terminal ซึ่งอาจจะเลือกหน้าออกมาพิมพ์ และเปลี่ยนรูปข้อมูลเข้าไปพิมพ์ สิ่งเหล่านี้มีหน้าที่เกี่ยวข้องกับ Presentation ของข้อมูลไม่มากนัก

Presentation Layer ที่เกี่ยวข้องกับ Syntax ที่นำมาใช้กับ Representing of Data และ Formatting of Data Representing of Data เกี่ยวกับชุดตัวอักษรต่างๆ และโครงสร้างของการเข้ารหัส Formatting of Data เกี่ยวกับลักษณะรูปแบบของข้อมูลที่ต้องการโดยเฉพาะ Input และ Output Device เนื่องจากแต่ละประเภทของ Syntax ก็ยังแบ่งประเภทต่างๆ ของ Syntax ได้อีก 3 ประเภท ที่เกี่ยวกับช่วงเวลาที่เกิด Session ระหว่าง Application Entities ทั้งสอง

- Syntax ที่ใช้กับ Sending Entity
- Syntax ที่ใช้กับ Receiving Entity
- Syntax ที่ใช้สำหรับการส่งถ่ายข้อมูล

Syntax ทั้ง 3 ประเภทนี้ อาจจะแตกต่างกันทั้งหมด หรือแตกต่างกันเพียง 2 ประเภท หรือเหมือนกันหมดทั้ง 3 ประเภท ซึ่งสามารถใช้กันได้เหมือนกัน ถ้าหากว่ามันต่างกัน Presentation Layer จะรับผิดชอบกับการเปลี่ยนแปลงจาก Syntax หนึ่งไปยังอีก Syntax หนึ่งเท่านั้น การเจรจากันระหว่าง Presentation Entities ต่างๆ นั้นหมายถึง Application Entities ต่างๆ กำหนดการเปลี่ยนแปลงที่จำเป็นและจุดไหนที่จะดำเนินการปกติการเจรจาเหล่านี้เกิดขึ้นเมื่อ Session ได้เริ่มขึ้นแล้วการเจรจาต่างๆ สามารถเกิดขึ้นในขณะ Session กำลังดำเนินการอยู่ได้ เช่นเดียวกัน ตารางที่ 8 เป็นรายการการบริการต่างๆ ที่ Presentation Layer จัดให้กับ Application Layer และกล่าวถึงหน้าที่ต่างๆ ที่ได้ดำเนินการให้กับบริการต่างๆ เหล่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Services to the Application Layer**

- **Session Services** : provision to the application layer of all the services provided by the session layer
- **Syntax Selection** : initial selection of a syntax and subsequent modification of the initial syntax selection
- **Syntax Transformation** : code and character set conversion and modification of data layout

**Presentation Layer Functions**

- Issuing a request to the session layer for the establishment of a session
- Initiation a data transfer from one application entity, or user, to another
- Negotiating and renegotiating the choice of a syntax to be used in the data transfer
- Performing any required data transformation or conversion
- Issuing a request to the session layer for the termiantion of a session

**ตารางที่ 8 The OSI Pressentation Layer**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 1.7 THE APPLICATION LAYER

Application layer มีหน้าที่ที่เกี่ยวข้องกับ Higher-level ที่ให้การสนับสนุนเกี่ยวกับการใช้งานหรือระบบที่ดำเนินการอยู่ เช่นตัวอย่าง สนับสนุนการทำงานของ โอเปอร์เรเตอร์, การใช้ประโยชน์จากข้อมูลทางไกล, การควบคุมการส่งถ่ายแฟ้มข้อมูล, การแจกจ่ายฐานข้อมูลสำหรับดำเนินการต่างๆ, หน้าที่ต่างๆสำหรับการโต้ตอบใน Higher-lever และอื่นๆ ขอบเขตที่จะได้รับการสนับสนุนในสิ่งเหล่านี้จะอยู่ใน Network Architecture และอยู่ในซอฟต์แวร์สำรองที่ใช้กับ Network Architecture เช่น ซอฟต์แวร์ฐานข้อมูล ซึ่งในแต่ละระบบจะไม่เหมือนกัน

เมื่อได้แจกจ่ายฐานข้อมูลและแฟ้มข้อมูลต่างๆที่จะนำไปใช้แล้ว การควบคุมต่างๆเป็นสิ่งจำเป็น เพื่อป้องกันปัญหาที่จะเกิดขึ้นกับข้อมูล หรือการหยุดชะงักการทำงานของเครื่องการควบคุมบางประเภทต้องมีการเชื่อมต่อกับโครงข่ายโดยเฉพาะ เช่น ลงเวลาเกี่ยวกับสิ่งที่ดำเนินการและสิ่งที่ดำเนินการอยู่นั้นเรียงลำดับไปทันทีที่ได้ลงเวลาไว้แล้ว (บางทีเรียกว่า Pipelining) การกำหนดความเร็วเป็นสิ่งที่จำเป็นสำหรับการดำเนินการบางอย่าง เพื่อที่จะให้เครื่องส่งสามารถส่งข้อมูลได้อย่างต่อเนื่อง โดยไม่ทำให้ข้อมูลเกิดการท่วมท้นที่เครื่องรับหรือเพื่อที่จะให้เครื่องพิมพ์ปลายทางสามารถรับที่อัตราความเร็วสูงสุดได้

Application layer กำหนดขั้นตอนต่างๆของการใช้โดยมีจุดที่จะเข้าไปสู่ระบบ และการกำหนดวิธีการต่อเชื่อมระหว่างอุปกรณ์ต่างๆเพื่อแลกเปลี่ยนข้อมูล และยังได้กำหนดหน้าที่ต่างๆที่เกี่ยวกับการสื่อสารระหว่างระบบต่างๆที่ไม่ได้กำหนดจากชั้นที่ต่ำกว่า หน้าที่ต่างๆเหล่านี้รวมถึงที่ดำเนินการโดยบุคคล เช่นเดียวกันกับที่ดำเนินการโดยโปรแกรมประยุกต์ในจำนวนของ 7 ชั้นด้วยกัน Application layer มีความยุ่งยากที่สุดที่จะให้ได้ตามมาตรฐานตามที่กำหนด และในเวลานี้ OSI Model ยังไม่ได้บัญญัติข้อกำหนดที่เกี่ยวกับหน้าที่ต่างๆ ที่ได้ดำเนินการโดย Application Layer อย่างไรก็ตาม ก็ยังได้แบ่งหน้าที่ต่างๆ ออกเป็น 2 ประเภทด้วยกัน ตารางที่ 9 แสดงรายการ การบริการต่างๆ ที่ Application layer กำหนดขั้นตอนต่างๆ ของการใช้หรือผู้ใช้ และกล่าวถึงประเภทของหน้าที่ต่างๆทั้ง 2 ประเภทไว้อย่างกว้างๆที่ได้ดำเนินการโดย Application Layer

นอกจากที่ได้แสดงในตารางที่ 9 แล้วยังได้จัดหาบริการต่างๆเพิ่มเติมให้อีกด้วย นอกจากนี้ Application Layer ยังให้การดำเนินการที่เกี่ยวข้องกับการบริการต่างๆทั้งในการดำเนินการที่เกี่ยวกับขั้นตอนต่างๆที่ได้นำมาประยุกต์ใช้และที่เกี่ยวข้องกับระบบที่นำมาต่อร่วมกันการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดำเนินการที่เกี่ยวข้องกับขั้นตอนต่างๆที่นำมาประยุกต์ใช้ที่ประกอบด้วย Initializing, Maintaining และ Terminating ของกระบวนการต่างๆวิธีการจัดสรรและยกเลิกการจัดสรร,การตรวจสอบและการ ป้องกันการหยุดชะงักของเครื่อง, ให้ความเชื่อถือ และมั่นใจ, มาตรการการป้องกัน, จุดตรวจสอบ และการควบคุมทำให้กลับคืนสู่สภาพปกติ รายการต่างๆที่กล่าวมานี้ยังไม่หมด การดำเนินการ ของระบบอาจประกอบด้วย Activating, maintaining,และ Deactivating ที่มีวิธีการหลายรูปแบบ การบรรจุโปรแกรม, การเฝ้าตรวจ และการรายงานสถานะและสถิติ, การตรวจสอบข้อผิดพลาด การวิเคราะห์ข้อผิดพลาด และ การทำให้กลับคืนสู่สภาพปกติ และการจัดองค์ประกอบใหม่และการให้เครื่องเริ่มต้นใหม่

| Services to the Application Process |
|---|
| - **Identification** : identification of the intended communication partners (e.g., by name, address, description, etc.) |
| - **Availability** : determination of the current availability of the intended communication partners |
| - **Authority** : establishment of the authority of the partners to communicate |
| - **Privacy** : agreement on the privacy mechanisms to be used in communication |
| - **Security** : authentication of the identity of the indeded communication partners |
| - **Cost Allocation** : determination of cost allocation methodology to be used in communication |
| - **Resource Allocation** : determination of the adequacy of available resources |
| - **Quality of Service** : determination of the acceptable quality of service error rate, and cost |
| - **Synchronization** : synchronization of cooperating applications |
| - **Diolog Discipline** : seliction of the dialog discipline to bo used in communication, including the procedures to be used in initiation and release |
| - **Error Recovery** : agreement on responsibility for error recovery |
| - **Data integriry** : agreement on the procedures for control of data integrity |

**ตารางที่ 9 The OSI Application Layer**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

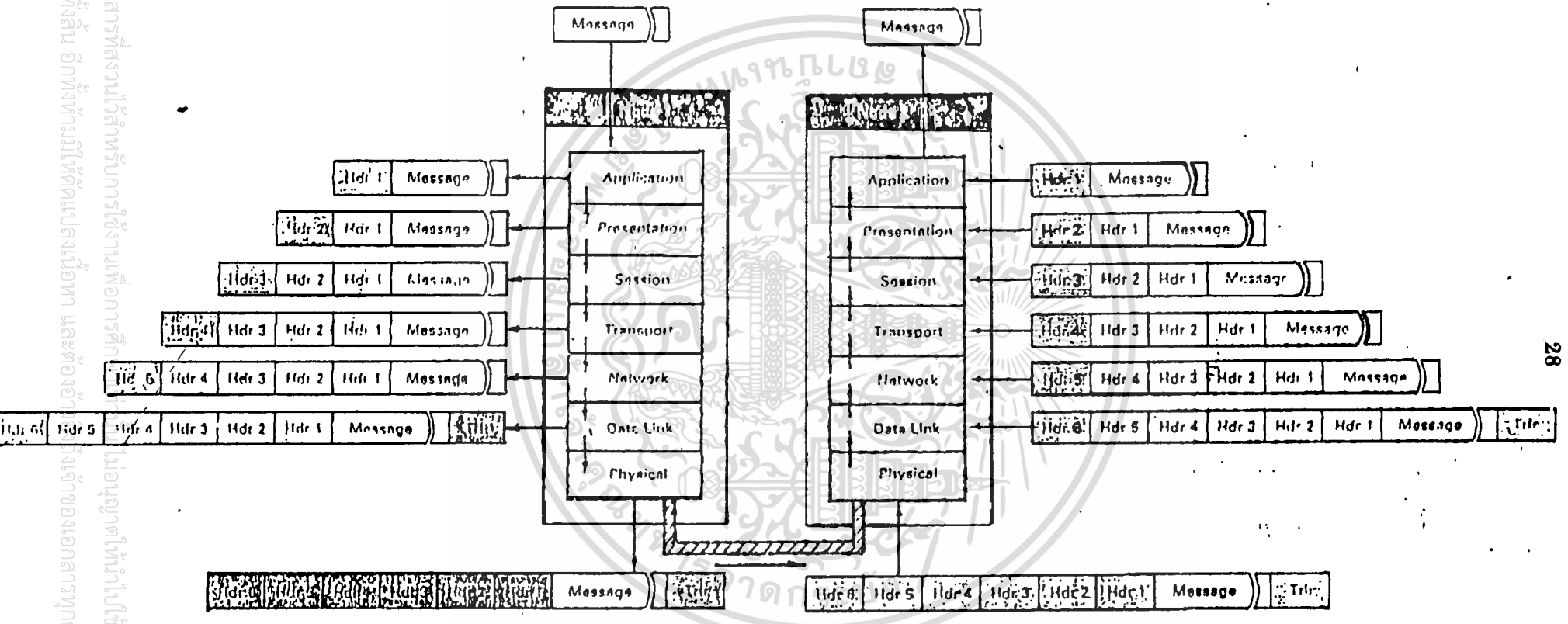
| |
|---|
| - **Data Syntax** : identification of constraints on data syntax (character sets, data structures,etc.) |
| **Application Layer Functions** |
| - Perfoming common application functions, which are functions that provide capabilities that are useful to many app,ications performing specific application functions, which are functions that are required to service the needs of a particular application |

ตารางที่ 9 The OSI Application Layer (ต่อ)

**MEASSAGE HEADERS**

ในขณะที่ข้อมูลเคลื่อนผ่านซอฟต์แวร์ของชั้นต่างๆ จะมี Headers ถูกเสริมเข้ากับข้อมูลเดิมเมื่อเข้าสู่ระบบที่ Application Layer ที่ Data Link Layer จะมีทั้ง Headers และ Trailer เสริมเข้าไปตามตัวอย่าง Headers และ Trailer เหล่านี้จะมีข้อมูลที่ใช้ในการควบคุมเพื่อที่จะให้ระบบปลายทางตรงข้ามใช้ประโยชน์ในการแปลงความหมายของข้อมูลที่พบใน Transmission Frame ดังแสดงในรูปที่ 3 ดังนั้น Transmission Frame ที่ส่งผ่าน Physical Circuit จะมีจำนวนบิทมากกว่าบิทข้อมูลเดิม ขณะที่ข้อมูลเคลื่อนที่ผ่านชั้นต่างๆ ที่ปลายทางตรงกันข้าม แต่ละชั้นจะขจัด Headers ที่ตรงกันออกไป และใช้ประโยชน์ที่มีข้อมูลอยู่ในนั้นจัดการกับข่าวสารได้อย่างถูกต้อง ดูรูปที่ 4 ความหมายที่เกี่ยวกับ Hearders และ Trailer ต้องกำหนดให้แน่ชัดตามมาตรฐานและโปรโตคอลที่ใช้ควบคุมหน้าที่ต่างๆ ของแต่ละชั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 3 การเติมส่วน Headers และ Trailers

รูปที่ 4 การแยกส่วน Headers และ Trailers

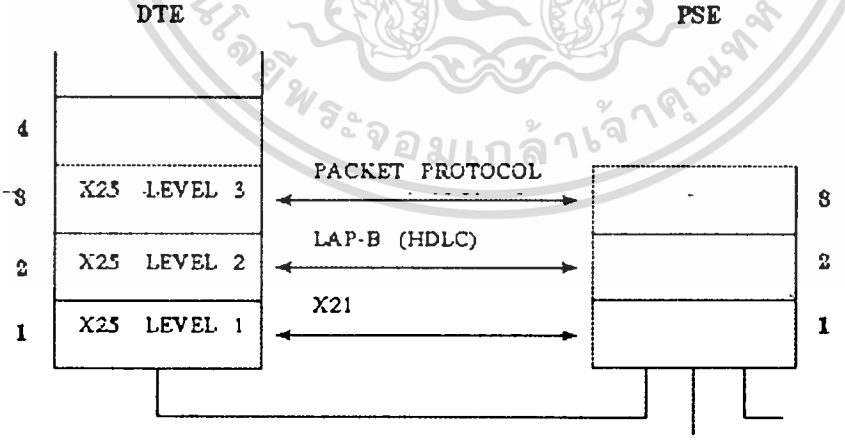
# บทที่ 2
# ลักษณะทั่วไปของ X.25

CCITT ได้เสนอแนะไว้ใน X.25 ภายใต้หัวข้อเรื่อง " *การเชื่อมโยงระหว่าง DTE กับ DCE สำหรับเทอร์มินอลที่ทำงานในลักษณะของแพคเกทที่เกี่ยวกับ Public Data Networks* " ระบบคอมพิวเตอร์หลัก , cluster controller (terminal computer) หรือ เทอร์มินอล สามารถ นำมาใช้เป็น DTE ได้ X.25 จะใช้ X.21 เป็นตัวเชื่อมโยงทางฟิสิคซ์ (level 1) Addressing จะดำเนินการโดย X.25 packet level (level 3) ดังนั้น X.21 จะไม่นำมาใช้สำหรับ addressing X.25 เป็นฐานการกำหนด virtual circuits ซึ่งเป็นหน้าที่ของ packet level (level 3)

X.25 level 3 ให้บริการที่แตกต่างกัน 3 อย่างดังนี้

- Virtual Call (VC)
- Permanent Virtual Circuit (PVC)
- Datagram (DG)

X.25 level 2 จะให้การแลกเปลี่ยนข้อมูลที่ data link level ถูกต้องซึ่งจะมีการใช้ โปรโตคอลที่เรียกว่า Link Access Protocol Balanced (LAP-B) สำหรับ X.25 ใช้ของ HDLC




**รูปที่ 5 แสดง Level ทั้ง 3 ของ X.25 กับ PSE (Packet Switching Exchange)**

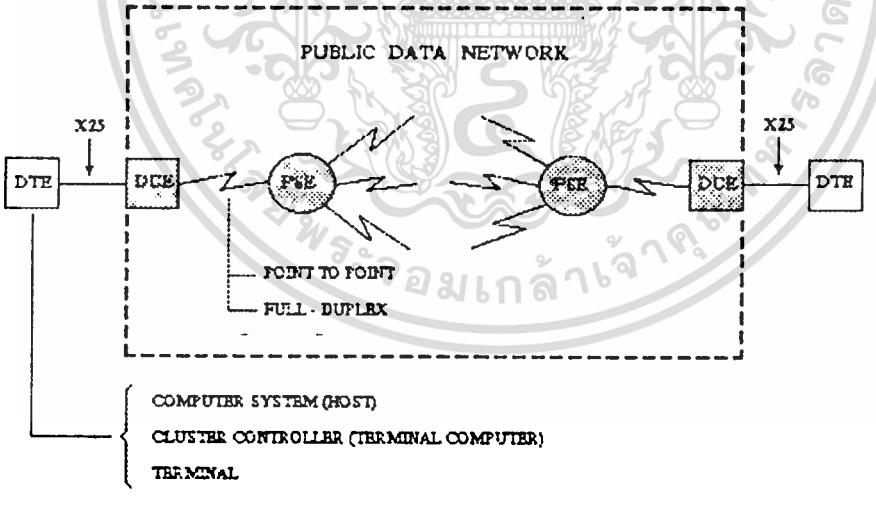
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.1 PACKET SWITCHING

ความคิดที่เกี่ยวกับ packet switching เกิดจากปัญหาที่เกี่ยวกับโครงข่ายโทรคมนาคมที่มีอยู่เดิมใน circuit switching การต่อทางฟิสิคซ์ต้องกำหนดและให้คงอยู่ตลอดช่วงระยะที่ต้องการสำหรับการส่งถ่ายข้อมูล ถ้าข่าวสารค่อนข้างสั้นและเกิดขึ้นในบางตอนที่มีรูปแบบที่ไม่แน่นอนทั่วไปแล้วจะเกิดจาก enqury/response ซึ่งมีผลกระทบต่อระบบทำให้การส่งถ่ายข้อมูลช้าและทำให้การใช้ทรัพยากรของสวิทชิ่งเป็นไปอย่างไม่มีประสิทธิภาพ

ระบบ message switching ทำให้การใช้ทรัพยากรที่มีอยู่แล้วมีประสิทธิภาพเพิ่มขึ้นแต่มันก็ยังมีข้อเสีย คือ คุณสมบัติโดยทั่วไปยังช้ามาก ที่เกี่ยวกับผลกระทบของการแลกเปลี่ยนข่าวสารซึ่งกันและกัน

Packet switching ได้รวบรวมข้อดีของ circuit switching และ message switching นั่นคือผู้ใช้รู้สึกเหมือนกับการต่อวงจรแบบ point-to-point กับผู้ใช้ที่ไกลออกไปแต่ทว่าความจริงแล้วไม่ใช่เป็นการต่อโดยตรง วงจรนี้เหมาะที่จะเรียกว่า **virtual circuit**




**รูปที่ 8** แสดง Packet Switching Network (PSN)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Packet switched Network คือ โครงข่ายสื่อสารที่ส่งข้อมูลในกลุ่มที่รู้จักกัน คือ แพคเกท (Packets) โครงข่ายประกอบด้วยชุดที่ต่อระหว่าง Packet switched Nodes อุปกรณ์จะต้องต่อเข้ากับโหนดใดโหนดหนึ่งของ Packet switched Nodes ข้อมูลที่ถูกส่งจะเป็นแพคเกทที่เรียงกันตามลำดับแต่ละแพคเกทจะถูกกำหนดเส้นทางโดยโครงข่ายแต่ละโหนดที่อยู่ระหว่างทางจะต้องบรรจบกัน แพคเกทต่างๆที่โหนดได้รับจะถูกเก็บไว้ชั่วคราวและจะถูกส่งออกไปในเส้นทางที่กำหนดที่เชื่อมโยงกับโหนดถัดไป การแลกเปลี่ยนแพคเกทข้อมูลระหว่างระบบปลายทางทั้งสองนั้นสามารถกระทำได้ทั้งสองทิศทาง วิธีการที่ใช้สำหรับควบคุมการส่งถ่ายและกำหนดเส้นทางของแพคเกทข้อมูลต่างๆ นั้นแบ่งออกได้เป็น 2 วิธี ด้วยกัน คือ

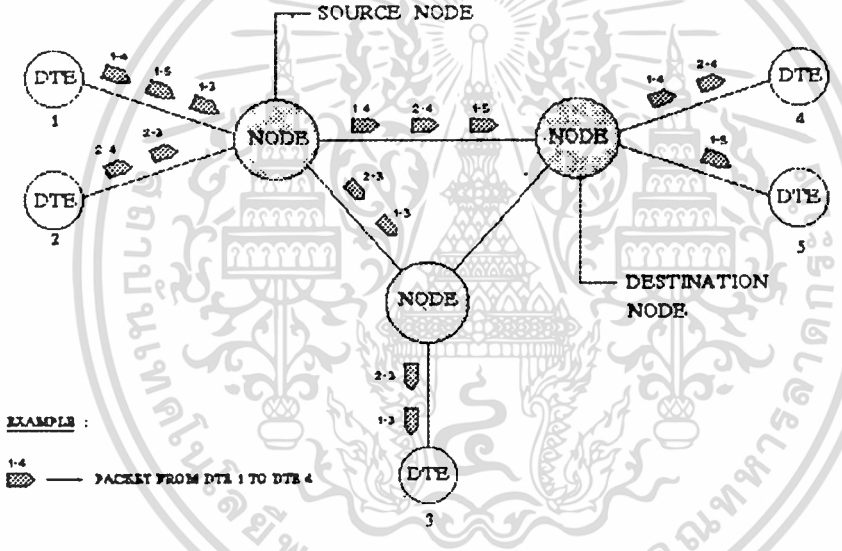
- Datagram และ
- Virtual Circuit

## 2.2 PACKETs

ข้อมูลของผู้ใช้ที่ส่งให้กับ network ต้องมีขนาดของแพคเกทค่อนข้างเล็ก แพคเกท หนึ่งๆ สามารถบรรจุได้ตั้งแต่ 16 ถึง 1024 ไบท์ ของข้อมูลผู้ใช้ แต่ละแพคเกทมี packet header อยู่ด้วย header จะบรรจุข้อมูลที่จำเป็นต่อเส้นทางของแพคเกทนอกจากจะมีแพคเกท ข้อมูลแล้วยังมีแพคเกทที่ใช้สำหรับทำหน้าที่ในการควบคุมอีกด้วย header ยังกำหนดประเภทของแพคเกทอีกด้วย ข้อเสนอแนะของ X.25 ยังได้กำหนดโครงสร้างและความสำคัญของประเภทของแพคเกทไว้หลายลักษณะ เทอร์มินอล และ อุปกรณ์ (DTEs) อื่นๆ ต้องผลิตแพคเกทให้สอดคล้องกับข้อเสนอแนะ

สำหรับ DTEs ซึ่งไม่สามารถทำตามนี้ได้ สามารถใช้อุปกรณ์ Packet Assembly/Disassembly(PAD) นี้เปลี่ยนข้อมูลของผู้ใช้ชนิด start/stop mode ให้เป็นแบบ packet mode ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 7 แสดงลักษณะของแพคเกตสวิทชิ่งใน PSDN

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.3 การเชื่อมโยงระหว่าง DTE กับ DCE

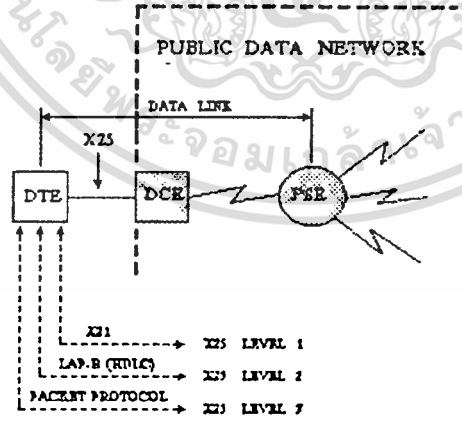
ข้อเสนอแนะของ X.25 ได้กำหนดทิศทางการเชื่อมโยงของ DTE กับ DCE ไว้การกำหนดนี้ได้จัดไว้เป็น 3 ระดับ เพื่อให้สอดคล้องกับ ISO Open System Interconnection Model มีดังนี้

Level 1 - การเชื่อมโยงทางฟิสิคซ์และทางไฟฟ้า
ใช้หลักการเดียวกันกับข้อเสนอแนะของ X.21

Level 2 - Link Access Procedure
เป็น logical link level และ วิธีการปฏิบัติเป็นชุดประกอบของ HDLC ใช้สำหรับ framing , synchronization และ error recovery สำหรับ point-to-point data link ระหว่าง DTE กับ network mode

Level 3 - Packet level
level นี้ทำหน้าที่ให้เกิด virtual circuit และให้มีการขนส่งข้อมูลระหว่าง DTE กับ network

Level เหล่านี้ แต่ละ Level จะได้พูดถึงรายละเอียดเพิ่มเติมในตอนต่อไป

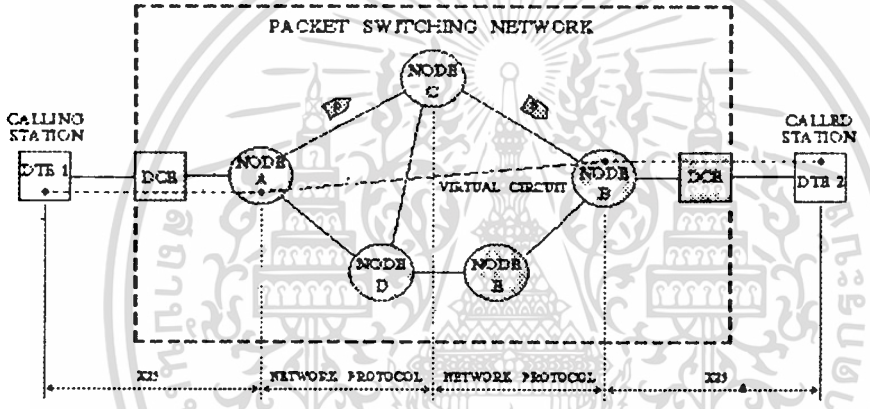



รูปที่ 8 แสดงการเชื่อมต่อระหว่าง DTE /DCE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.4 VIRTUAL CIRCUITs

ข้อได้เปรียบที่สำคัญ ที่เกี่ยวกับแนวความคิดของ virtual circuit คือว่า ศูนย์กลางคอมพิวเตอร์หลัก กล่าวได้ว่ามีการต่อเชื่อมเพียง 1 physical network เท่านั้น สามารถจะติดต่อกับ remoteDTEs ได้พร้อมกันหลายราย ที่เป็นไปได้เนื่องจากแพคเกทต่างๆสำหรับ virtual circuits จำนวนมากเป็น interleased ที่เกี่ยวกับการเชื่อมโยงทางฟิสิคซ์ เช่น DTE หนึ่งสามารถติดต่อกับ DTE อื่นๆอีกได้พร้อมกันหลายราย เรียกว่า **multi-channel DTE** ซึ่งตรงข้ามกับ **single-channel DTE**




รูปที่ 9 Virtual Circuit

Virtual Circuit มีอยู่ 2 ประเภท กล่าวคือ

### 2.4.1 A Virtual Call

ระหว่าง DTE ทั้งสองจะเป็นแบบ **temporary virtual circuit** ทางด้านลอจิคอลเปรียบได้กับการเรียกใน circuit switching network

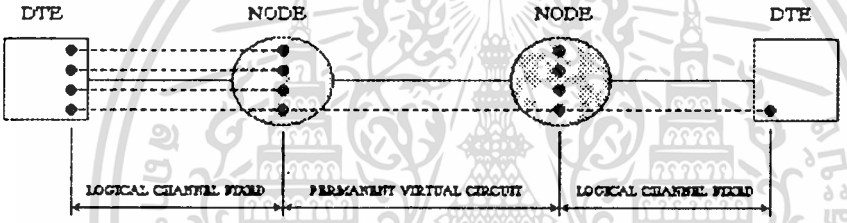
Virtual call มีขั้นตอนต่างๆขณะที่มันผ่านไป สิ่งเหล่านี้คือ call set-up, data transfer และ call clear Data transfer อาจเป็นแบบ one-way หรือ two-way พร้อมกันก็ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Data transfer ปกติจะมีผลต่อการส่งถ่ายของแพคเกทต่างๆจำนวนมาก แพคเกทต่างๆ เหล่านี้จะนำส่งเรียงตามลำดับไม่เปลี่ยนแปลงไปให้กับ network ถ้าตรวจพบมี error ในแพคเกท แพคเกทที่ตามกันมาจะคอยจนกว่ามีการแก้ไขให้กลับคืนสู่สภาพเดิมใหม่นั่นคือจะไม่มีการไล่ทัน

### 2.4.2 A Permanent Virtual Circuit

ระหว่าง DTE ทั้งสองเป็นแบบ fixed virtual circuit และ ทางด้านลอจิกคอลเปรียบได้กับ leased circuit นั่นคือ ไม่จำเป็นต้องมีการเรียก DTE ปลายทางก่อนที่จะส่งถ่ายข้อมูล แล้ว local node จะให้ข้อมูลที่เกี่ยวกับวงจรเหล่านี้เมื่อเปิดการรับ DTE เป็นสมาชิก




รูปที่ 10 Permanent Virtual Circuit (PVC)

## 2.5 LOGICAL CHANNELs

เพื่อให้ virtual calls และ/หรือ permanent virtual circuits ใช้ logical channel ได้พร้อมกันหลายวงจร virtual call และ prtmanent vircuit circuit แต่ละอย่าง ได้กำหนด logical channel group number ไว้ (น้อยกว่าหรือเท่ากับ 15) และ logical channel number (น้อยกว่าหรือเท่ากับ 255) สำหรับ virtual calls หมายเลขเหล่านี้ ได้ถูกกำหนดขึ้นระหว่าง call set-up phase สำหรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

permanent virtual circuit หมายเลขเหล่านี้ได้ถูกกำหนดขึ้นเมื่อมีการทำสัญญากับฝ่ายการจัดการขณะสมัครเป็นสมาชิกเข้ารับการบริการ

ในทางทฤษฎีมี logical channel ได้ถึง 4095 วงจร มีอยู่ที่ X.25 DTE/DCE interface แต่ละแห่งในทางปฏิบัติมีจำนวนจำกัดขึ้นอยู่กับการใช้แต่ละกรณี จำนวนของวงจรมีการตกลงขณะที่สมัครเป็นสมาชิก

Virtual circuit สามารถกล่าวได้ว่าเป็นการต่อทาง logical ระหว่าง logical channel 2 วงจรไม่ว่าจะเป็นการต่อแบบ temporary (virtual call) หรือ แบบ permanent (virtual circuit) หมายเลขของ logical channel จำเป็นสำหรับการกำหนดเส้นทาง เมื่อมีการส่งแพคเกทของ ข้อมูลเท่านั้น

## 2.6 DATAGRAMs

เป็นบริการที่อาจจัดให้ในบางโครงข่าย ถึงแม้ว่ามันจะไม่ได้รับการพิจารณาว่ามีความสำคัญอย่างเดียวกันกับ Virtual call service แต่โดยหลักการแล้วมันง่ายกว่า Virtual call concept มาก และคล้ายกันกับ Message switching

Datagram service ไม่จำเป็นต้องมี call set-up ก่อนที่จะส่งแพคเกทข้อมูล แต่ละ datagrams จะมีแพคเกทข้อมูลเป็นของตัวเอง พร้อมด้วย full DTE addressing นั่นคือไม่เหมาะกับ logical channel number และไม่มีการรับประกันว่า datagrams ที่ส่งไปให้กับโครงข่ายโดย DTE และแอดเดรสไปให้กับอีก DTE หนึ่งนั้นจะถูกนำส่งในขั้นตอน อย่างเดียวกัน

ถ้าผู้ใช้ต้องการที่จะส่งแพคเกทต่างๆไปในแพคเกทเดียวกัน ในช่วงระยะที่มีการติดต่อระหว่าง DTE กับ DTE แล้วประสิทธิภาพของระบบ datagram จะเพิ่มมากขึ้น เนื่องจากไม่มี call set-up และ clearance อย่างไรก็ตามโดยทั่วไปแล้วช่วงระยะเวลาจะมีผลต่อการแลกเปลี่ยนแพคเกทจำนวนมากดังนั้นจะน่าเลือกใช้ Virtual call concept ได้เช่นเดียวกับ Virtual call data packets ซึ่งไม่ได้กระโดดข้ามเส้นทางของข้อมูล โดยทั่วไปแล้วการดำเนินไปตาม เส้นทางง่ายกว่ามาก

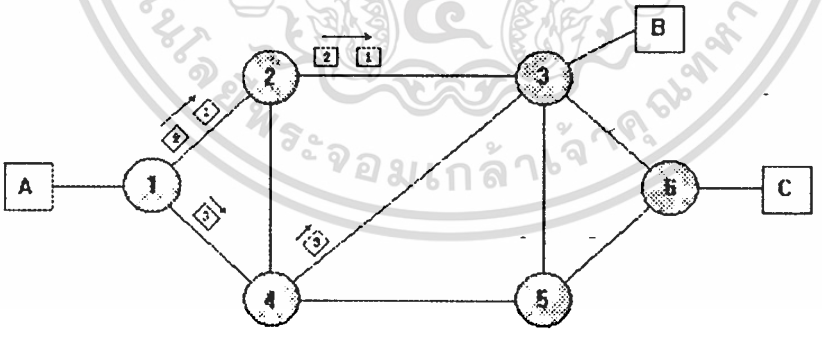
Logical channels ที่ DTE/DCE interface บางแห่งจะสำรองไว้สำหรับ datagram service ถ้ามันต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.7 วิธีการของ Datagram และ Virtual Circuit

### 2.7.1 วิธีการของ Datagram

แต่ละแพคเกทจะดำเนินการอย่างอิสระ จะพิจารณาสิ่งที่เกี่ยวข้องกับวิธีการนี้สมมติว่าสถานี A มีแพคเกทข้อมูลอยู่ 3 แพคเกท ที่จะส่งไปให้กับสถานี E สถานี A ส่งแพคเกท 1-2-3 เรียงกันตามลำดับไปให้กับโหนด 4 เมื่อโหนด 4 ได้รับแพคเกทที่ส่งเรียงกันมานั้นจะต้องทำการกำหนดเส้นทางให้กับแพคเกทเหล่านั้น เช่นเมื่อโหนด 4 ได้รับ แพคที่ 1 มาจะต้องกำหนดว่าคิวของแพคเกท สำหรับโหนด 5 หรือโหนด 7 ใครจะสั้นกว่ากันเมื่อเห็นว่าคิวของแพคเกทสำหรับโหนด 5 สั้นกว่าโหนด 4 ก็จะผ่านไปให้กับโหนด 5 สำหรับแพคเกทที่ 2 ก็จะดำเนินการเช่นเดียวกัน แต่แพคเกทที่ 3 โหนด 4 พบว่าคิวของแพค เกทสำหรับโหนด 7 สั้นกว่าโหนด 5 ดังนั้นจึงจัดคิวของแพคเกทที่ 3 ส่งไปให้กับโหนด 7 ดังนั้นแพคเกทต่างๆ แต่ละแพคเกทจะมีแอดเดรสปลายทางอย่างเดียวกันแต่ไม่จำเป็นที่จะต้องเดินทางไปในเส้นทางเดียวกันทั้งหมด โดยเฉพาะอย่างยิ่งมันอาจเป็นไปได้ว่าแพคเกทที่ 3 อาจไปถึงโหนด 6 ก่อนแพคเกทที่ 2 ด้วยเหตุนี้อาจเป็นไปได้ว่าแพคเกทต่างๆที่ส่งไปให้กับสถานี E จะมีแพค เกทหนึ่งไม่เรียงลำดับกับแพคเกทอื่นที่ส่งไป เมื่อไปถึงสถานี E จะต้องแก้ปัญหาว่าจะจัดลำดับให้กับมันใหม่ได้อย่างไร จากที่ได้เสนอแนะเกี่ยวกับ Datagram จะเป็นไปอย่างอิสระ

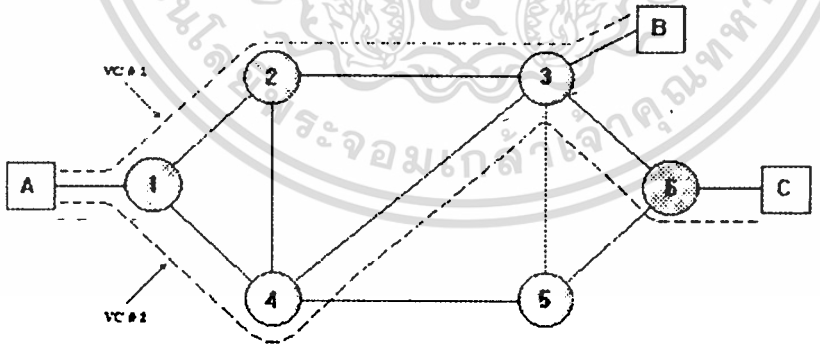



**รูปที่ 11 (ก)** Datagram แต่ละแพคเกตดำเนินการอิสระโดยโครงข่าย แพคเกตต่างๆ จะมีเครื่องหมายบอกแอดเดรสปลายทาง และอาจไปยังโนดปลายทางโดยไม่เรียงลำดับกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.7.2 วิธีการของ Virtual Circuit

คือ กำหนดให้มีการเชื่อมต่อทางลอจิกคอลก่อนที่จะมีการส่งแพคเกทข้อมูลใดๆออกไป เช่น ตัวอย่างสมมุติว่าสถานี A มีข้อมูลจำนวนหนึ่งหรือมากกว่าที่จะส่งไปให้กับสถานี E ขั้นตอนแรก สถานี A จะส่ง Call Request Packet ไปให้กับโหนด 4 แจ้งความประสงค์ที่จะเชื่อมต่อกับสถานี E โหนด 4 จะกำหนดเส้นทางตามต้องการและส่งข้อมูลที่ตามมา ทั้งหมดนั้นไปให้กับโหนด 5 ซึ่งโหนด 5 จะกำหนดเส้นทางที่ต้องการ และส่งข้อมูลที่ตามมา ทั้งหมดไปให้กับโหนด 6 ซึ่งในที่สุดจะส่ง Call Request Packet ไปให้กับสถานี E ถ้าสถานี E พร้อมที่จะรับการเชื่อมต่อ สถานี E จะส่ง Call Accept Packet ไปให้กับโหนด 6 แพคเกทนี้จะถูกส่งผ่านกลับไปให้โหนด 5 และ 4 จนถึงสถานี A ตอนนี้สถานี A และ E สามารถและเปลี่ยนข้อมูลกันได้โดยผ่านการเชื่อมต่อทางลอจิคอล หรือ Virtual Circuit ตามที่ ได้กำหนดไว้แล้ว ดังนี้ขณะนี้ข้อมูลที่มีอยู่ใน Virtual Circuit แต่ละแพคเกทจะถูกกำหนดลำ ดับหมายเลขของแพคเกทด้วยเหมือนกัน ดังนั้นข้อมูลทุกแพคเกทจากสถานี A จะถูกส่งผ่าน โหนด 4,5 และ 6 และในทำนองเดียวกันข้อมูลทุกแพค เกทจากสถานี E ก็จะถูกส่งผ่านโหนด 6 , 5 และ 4 ในที่สุดเมื่อสถานีใดสถานีหนึ่งต้องการยุติ การเชื่อมต่อก็จะส่ง Clear Request Packet ไปให้กับโครงข่าย




**รูปที่ 11 (ข)** Virtual Circuit แพคเกตต่างๆ ต้องกำหนดลำดับหมายเลข และเส้นทางระหว่าง 2 สถานี แพคเกตต่างๆ สำหรับ Virtual Circuit จะเดินทางไปเส้นทางเดียวกัน และดึงที่ปลายทางตามลำดับของแพคเกต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 3

# X.25 Level 1 - Physical Layer

## 3.1 บทนำ

X.25 level 1 ประกอบด้วย ทางกล,ทางไฟฟ้า,หน้าที่ควบคุมและวิธีการปฏิบัติที่มีเฉพาะกับการต่อการคงอยู่และปลดออกจาก physical line ระหว่าง DTE กับโครงข่าย (DCE และ packet switching exchange) X.25 level 1 ตรงกับ Physical layer ใน OSI model

CCITT ได้เสนอแนะเกี่ยวกับการเชื่อมโยงของ X.21 หรือ X.21-bis ที่นำมาใช้กับ level 1 ของ X.25 การเชื่อมต่อระหว่าง DTE กับ packet switching exchange เป็นแบบ **full-duplex point-to-point** data circuit ที่นำมาใช้กับวงจรเช่าทั่วๆ ไป ข้อบังคับต่างๆ ของ X.21 เฉพาะวงจรเช่าที่การเชื่อมต่อแบบ point-to-point นั้นได้นำมาประยุกต์ใช้กับ level 1

X.25 ไม่ได้ใช้ในส่วนของแอดเดรสซิ่ง แอดเดรสซิ่งใน X.21ใช้เฉพาะกับ circuit-switched switched public data networks เท่านั้น แอดเดรสซิ่งใน X.25 จะมีอยู่ใน X.25 packet level (level 3) เท่านั้น สำหรับ X.25 state 12 ของ X.21 (read for data) เป็นสิ่งจำเป็นใน state นี้ c = ON และ i = ON ซึ่งหมายความว่า แลกเปลี่ยนข้อมูลกันได้

ขั้นตอนปฏิบัติสำหรับ DTE ที่ถูกต่อเข้ากับ packet-switched data transmission service ที่เป็นผลสืบเนื่องมาจากวงจรควรจะมีดังต่อไปนี้

- **DCE NOT READY**

DCE NOT READY บอกให้ทราบว่าไม่สามารถให้บริการได้ โครงข่ายอาจจะอยู่ในสภาวะที่ผิดปกติหรือกำลังทำการ test loop อยู่ ภาวะนี้สัญญาณไบนารี่จะทรงตัว คือ r = 0 และ i = OFF

- **DTE UNCONTROLLED NOT READY**

DTE จะบอกให้ทราบว่าไม่สามารถปฏิบัติการได้ โดยมีสัญญาณ t = 0 และ c = OFF

- **การทำงานของ Interchange Circuits**

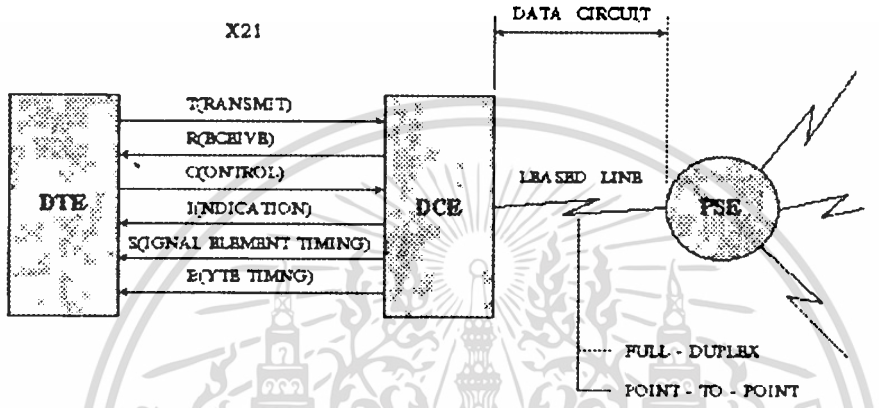
เมื่อ c = ON , ข้อมูลจะส่งผ่านวงจร T นำส่งไปยังโครงข่าย (network)

เมื่อ c = OFF , DTE ต้องส่งสัญญาณ t = 1 (DTE READY)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 3.2 INTERCHANGE CIRCUITs

Interchange Circuits สำหรับ X.21 interface (X.25 level 1) ซึ่ง CCITT ได้เสนอแนะไว้ใน X21




รูปที่ 12 X.21 INTERCHANGE CIRCUITS

**Circuit T - Transmit**

สัญญาณข้อมูลในรูปไบนารี่เริ่มจาก DTE ถูกส่งไปยัง DCE นั้นกระทำที่วงจรนี้เพื่อการส่งถ่ายไปยัง remote DTE หนึ่งหรือหลายๆ สถานี

วงจร T ยังใช้ส่งสัญญาณเพื่อ call control โดยใช้รหัส IA5 (International Alphabet No.5) โดย DTE ส่งไปให้กับ DCE เพื่อกำหนดการเรียกและดำเนินการสำหรับการควบคุมหน้าที่อื่นๆ อีก

**Circuit R - Receiver**

สัญญาณข้อมูลในรูปไบนารี่ ถูกส่งถ่ายจาก DCE ไปให้กับ DTE ที่กำลังรับอยู่โดยผ่านวงจร R นี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจร R นี้ยังใช้ส่งสัญญาณควบคุมการเรียกในรูปของรหัส IA5 จาก DCE ไปให้กับ DTE สัญญาณนี้จำเป็นสำหรับการกำหนดการเรียก และดำเนินการสำหรับการควบคุมหน้าที่อื่นๆ อีก

**Circuit C - Control**

วงจรนี้มีอยู่ 2 สภาวะ (ON/OFF) ซึ่งทำให้ DTE สามารถควบคุม DCE ได้ การทำงานของมันจะทำร่วมกับ วงจร T วงจร C ใช้เพื่อกำหนด call request,call acceptance และกำหนด call clearance indication และ confirmation ในระหว่างที่มีการส่งถ่ายข้อมูลอยู่นั้นต้องอยู่ในสภาวะ ON

**Circuit I - Indication**

วงจรนี้มีอยู่ 2 สภาวะ (ON/OFF) ซึ่งทำให้ DCE สามารถแจ้งไปยัง DTE ให้ทราบถึงสภาวะขบวนการเรียกที่กำลังดำเนินอยู่ วงจร I นี้ทำงานร่วมกับวงจร R เมื่อวงจร I อยู่ในสภาวะ ON วงจร R สามารถส่งข้อมูลจาก distant DTE ได้ เมื่อวงจร I อยู่ในสภาวะ OFF วงจร R จะใช้ส่งเฉพาะข้อมูลที่ใช้ในการควบคุมเท่านั้น

**Circuit S - Signal Element Timing**

DCE จะส่ง Signal element timing ไปให้กับ DTE โดยผ่านวงจรนี้ ปกติแล้วสัญญาณนี้จะมีช่วงระยะเวลา ON กับ OFF เท่านั้น อย่างไรก็ตามก็ยอมให้ OFF มีช่วงระยะเวลาเท่ากับผลรวมของคาบเวลาที่เป็นจำนวนคี่ของสภาวะ ON ได้สัญญาณนี้จะเบี่ยงเบนได้ไม่เกิน 1 เปอร์เซนต์

**Circuit B - Byte timing**

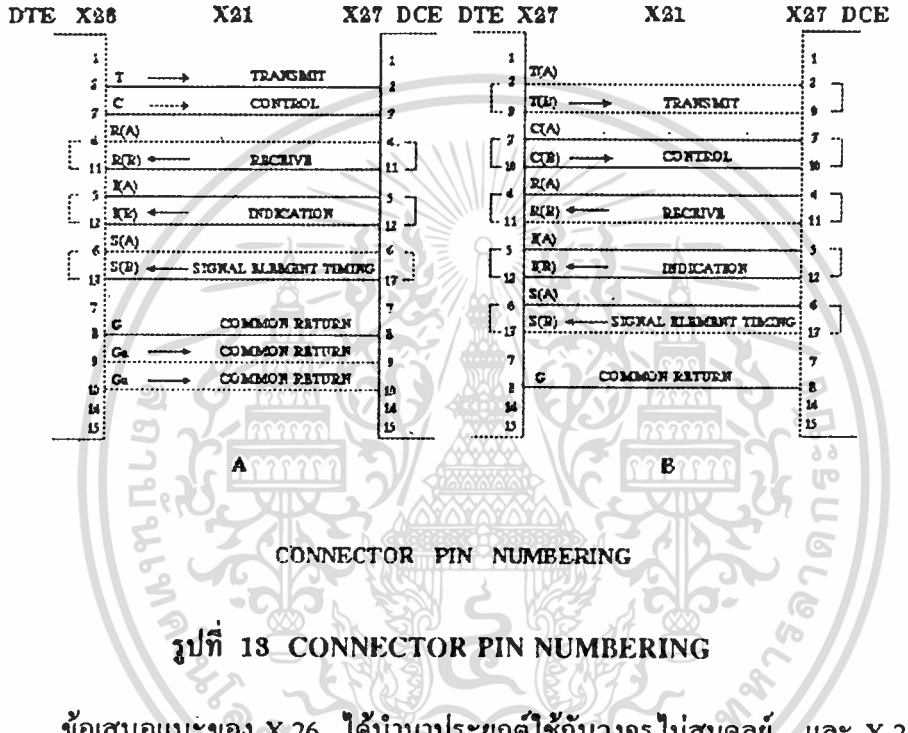
DCE จะส่งสัญญาณนี้ไปให้กับ DTE โดยใช้ 8-bit byte timing มันจะอยู่ในภาวะ OFF สำหรับ ช่วงเวลาที่เกี่ยวกับ S-circuit ON ซึ่งแสดงถึงบิทสุดท้ายของไบท์ เวลาอื่นมันจะอยู่ในภาวะ ON ใน X.21 ถ้าเป็นการกระทำใน DCE ไม่จำเป็นต้องใช้วงจรนี้

**ข้อสังเกตุ** : ข้อกำหนดของ interchange circuits ใช้ได้ทั้ง circuit-switching และ packet-switching service

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 3.3 คุณลักษณะทางไฟฟ้า

คุณลักษณะทางไฟฟ้า ของวงจรระหว่าง DTE กับ DCE ได้ระบุไว้ในข้อเสนอแนะของ CCITT เกี่ยวกับ X.26 และ X.27




รูปที่ 18 CONNECTOR PIN NUMBERING

ข้อเสนอแนะของ X.26 ได้นำมาประยุกต์ใช้กับวงจรไม่สมดุลย์ และ X.27 ได้นำมาใช้กับวงจรสมดุลย์(ข้อสังเกตุ : X.26 และ X.27 เทียบได้กับข้อเสนอแนะของ V.10 และ V.11 ตามลำดับ)

สำหรับ X.21 จุดเด่นคือ วงจรสามารถทำงานด้วยอัตราความเร็วของบิทที่น้อยกว่าหรือเท่ากับ 9600 บิทต่อวินาที และสูงกว่า 9600 บิทต่อวินาทีได้

การใช้วงจรไม่สมดุลย์ (unbalanced) จะต้องหลีกเลี่ยงสิ่งดังต่อไปนี้

- ไม่ควรใช่สายเคเบิลที่ต่อร่วม DTE กับ DCE ให้ยาวเกินไป
- ในที่ ๆ ซึ่งมีแหล่งกำเนิดเสียงรบกวน (noise) ภายนอก ซึ่งจะทำให้วงจรไม่สมดุลย์ไม่สามารถทำงานได้
- ในที่ๆซึ่งมีความจำเป็นที่จะต้องทำให้สิ่งรบกวน (interference) ลดลงด้วยสัญญาณอื่นๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กฎต่างๆที่นำมาประยุกต์ใช้ที่เกี่ยวกับระดับสัญญาณที่มีผลต่อวงจรสมดุลย์และไม่สมดุลย์มีดังนี้

- **Data circuits** : '0' Va' - Vb' มากกว่า +0.3 โวลท์
  (T และ R) 'I' Va' - Vb' น้อยกว่า -0.3 โวลท์
- **Control**
  **Timing circuits** : ON Va' - Vb' มากกว่า +0.3 โวลท์
  (C,I,S และ B) OFF Va' - Vb' น้อยกว่า -0.3 โวลท์

สำหรับวงจรต่างๆ ข้อบังคับได้กำหนดให้ใช้ตัวอักษรตัวพิมพ์ใหญ่ (T,R,C,I,S และ B) และภาวะของสัญญาณที่วงจรเหล่านี้ได้กำหนดให้ใช้ตัวอักษรตัวพิมพ์เล็ก (t,r,c,i,s และ b)

คำนิยามต่างๆ ของ **Interchange circuits** ได้กำหนดไว้ในข้อเสนอแนะของ X.24

Va' และ Vb' เทียบได้กับระดับของโวลท์เตจที่จุด A' และ B' สัมพันธ์กับจุด C' สำหรับวงจรไม่สมดุลย์ อิมพีแดนซ์ของเครื่องส่งควรเท่ากับหรือน้อยกว่า 50 โอห์ม สำหรับวงจรสมดุลย์ อิมพีแดนซ์ของเครื่องส่งควรเท่ากับหรือน้อยกว่า 100 โอห์ม และจะต้องสมดุลย์จริงๆ เมื่อเทียบกับจุด C

รูปภาพที่ 13 แสดงได้แสดงถึงการเชื่อมโยงที่ใช้ X.21 ไว้ 2 ลักษณะ คือ ในภาพ A ใช้ทั้งวงจรสมดุลย์และไม่สมดุลย์ ในภาพ B ใช้วงจรสมดุลย์อย่างเดียว ในภาพยังแสดงถึงจำนวนขาของตัวต่อชนิด 15 ขาสำหรับข้อมูลที่ต้องการเพิ่มที่เกี่ยวกับตัวต่อดูได้จาก ISO 4903

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 4

# X.25 Lever 2 - Data Link Layer

## 4.1 บทนำ

X.25 level 2 ประกอบด้วย Link Access Procedure ซึ่งใช้สำหรับการแลกเปลี่ยนข้อมูลระหว่าง DCE (network) กับ DTE ที่ทำงานใน user class of service 8-11 การดำเนินการได้ใช้หลักการและข้อกำหนดของ High level Data Link Control (HDLC) วิธีการปฏิบัติได้กำหนดโดย International Organization for Standardization (ISO)

ที่นำมาใช้กันมาก คือ LAP-B (Link Access Procedure Balanced) วิธีการปฏิบัติที่จะกล่าวถึงนี้เป็นชุดประกอบของ "HDLC balanced" บวกกับ option 2 และ 8

## 4.2 ข้อสังเกตุที่เกี่ยวกับตาราง HDLC

RR,RNR และ REJ commands ไม่ได้ส่งจาก DCE แต่ได้รับมาจาก DTE

## 4.3 วิธีการปฏิบัติที่เกี่ยวกับ Addressing

Address ที่อยู่ใน address field ของ เฟรม มีรหัสดังนี้

| Address | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| A | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| B | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

วิธีการใช้ของ addresses A และ B ดังในตาราง

## 4.4 Poll/Final bit

เมื่อ DCE ได้รับ SABM,DISC,Supervisory command หรือ I-frame ที่มี P = 1 ในการตอบเฟรมต่อไป DCE จะต้องให้ F = 1

Response frame ที่ส่งจาก DCE ไปให้กับ SABM หรือ DISC command ที่มี P = 1 จะตอบด้วย UA หรือ DM มี F = 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Response frame ที่ส่งจาก DCE ไปให้กับ I-frame ที่มี P = 1 มันจะตอบด้วย RR,REJ หรือ RNR มี F = 1

Response frame ที่ส่งจาก DCE ไปให้กับ S-command frame ที่มี P = 1 มันจะตอบด้วย RR หรือ RNR มี F = 1

| FORMAT | COMMANDS | RESPONSES | CONTROL FIELD | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| I-FRAME | I | | 0 | N(S) | | | P | N(R) | | |
| S-FRAMES | RR | RR | 1 | 0 | 0 | 0 | P/F | N(R) | | |
| | RNR | RNR | 1 | 0 | 1 | 0 | P/F | N(R) | | |
| | REJ | REJ | 1 | 0 | 0 | 1 | P/F | N(R) | | |
| U-FRAMES | | DM | 1 | 1 | 1 | 1 | F | 0 | 0 | 0 |
| | SABM | | 1 | 1 | 1 | 1 | P | 1 | 0 | 0 |
| | DISC | | 1 | 1 | 0 | 0 | P | 1 | 0 | 1 |
| | | UA | 1 | 1 | 0 | 0 | F | 1 | 0 | 0 |
| | | FRMR | 1 | 1 | 1 | 0 | F | 0 | 0 | 1 |

ตารางที่ 10 HDLC TABLE

## 4.5 ขั้นตอนต่างๆ สำหรับ link set-up

DTE อาจกำหนดการเชื่อมโยงโดยการส่งแฟลกต่อเนื่องกันได้ (active channel state)

### 4.5.1 Link set-up

เมื่อได้รับ SABM command DCE จะตอบด้วย UA ไปให้กับ DTE และเป็นการกำหนด send และ receive state variable V(S) และ V(R) ไว้ที่ 0

**ข้อสังเกตุ** : SABM command ส่งจาก DTE เท่านั้น

### 4.5.2 Information transfer phase

หลังจากที่ได้ส่ง UA response ไปให้กับ SABM command แล้ว DCE จะตอบสนองและส่ง I และ S-frammes ต่อไป เมื่อได้รับ SABM command ในขณะที่อยู่ในขั้นตอนการส่งถ่ายข้อมูลนั้น DCE จะต้องทำให้ตรงกันกับขั้นตอนที่ได้มีการปรับใหม่ (resetting procedure)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 4.5.3 Link disconnection

ระหว่างที่อยู่ในขั้นตอนการส่งถ่ายข้อมูลอยู่นั้น DTE จะแจ้งถึงการปลดออกจากวงจรการเชื่อมโยงโดยการส่ง DISC command ไปให้กับ DCE

เมื่อ DCE ได้รับ DISC command จะส่ง UA response ไปให้กับ DTE และจะเข้าสู่ขั้นตอนของ disconnection

ข้อสังเกตุ : DISC command ส่งจาก DTE เท่านั้น

### 4.5.4 Disconnected phase

หลังจากที่ได้รับ DISC command จาก DTE และ ตอบด้วย UA response ไปให้กับ DTE DCE จะเข้าสู่ขั้นตอน disconnected phase

ใน disconnected phase DCE จะเริ่มต้นใหม่ได้ใหม่นั้นจะต้องได้รับ SABM command พร้อมกับ UA response ไปให้กับ DTE และ จะส่ง DM response เป็นคำตอบเมื่อได้รับ DISC command

เมื่อได้รับ command frame อะไรก็ตามที่มี P = 1 DCE จะส่ง DM response ที่มี F = 1

เมื่อได้รับเฟรมอื่นๆ อีกใน disconnected phase DCE จะไม่ขอมรับรู้ทั้งสิ้น

DCE อาจจะเข้าสู่ disconnected phase ได้เช่นเดียวกันเมื่อตรวจพบภาวะของ error หรือมีการผิดปกติไม่สามารถปฏิบัติการได้ชั่วคราวหลังจากทำให้มันกลับคืนสู่สภาพเดิม ในกรณีนี้ DCE จะส่งDM และเริ่มต้นด้วย timer T1 ถ้า timer T1 หมดเวลาแล้วก่อนที่จะได้รับ SABM หรือ DISC command จาก DTE DCE จะส่ง DM response และเริ่มต้น timer T1 ใหม่

หลังจากส่ง DM response N2 (system parameter) เป็นพักๆ จาก DCE DCE จะยังคงอยู่ใน dicconnected phase และจะไม่สนใจที่จะส่ง DM response จะหยุดชั่วคราว

## 4.6 ขั้นตอนต่างๆ สำหรับการส่งถ่ายข่าวสาร

### 4.6.1 Transmitting I-frames

เมื่อ DCE ได้ส่ง I-frames ออกไปนั้นจะมี N(S) = V(S) และ N(R) = V(R) เมื่อส่ง I-frame สุดท้ายออกไปแล้วมันจะเพิ่ม V(S) อีก 1 ขั้น ถ้า timer T1 ไม่เริ่มทันทีที่มันส่ง I-frame มันจะเริ่มต้นใหม่

ถ้า V(S) เท่ากับ N(R) ที่ได้รับครั้งสุดท้ายบวกด้วย k (k คือจำนวนสูงสุดของเฟรมที่ยังค้างอยู่) DCE จะไม่ส่ง I-frame อื่นๆ อีก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อที่จะให้มั่นใจได้ว่ามีการส่งถ่ายข้อมูล DTE จะไม่ส่ง I-frame อื่นๆ อีก ถ้า V(S) = N(R) หลังสุดที่ได้รับจาก DCE บวก 7

เมื่อ DCE อยู่ในภาวะ "busy" หรือ "frame reject" มันอาจยังคงส่ง I-frame ได้ โดยมีข้อแม้ว่า DTE โดยตัวมันจะต้องไม่ 'busy' ถ้า DCE อยู่ในภาวะ "frame rejection " เนื่องจากได้รับ N(R) ไม่ถูกต้อง DCE จะระงับการส่ง I-frames

**4.6.2 Receiving I-frames**

เมื่อ DCE ไม่อยู่ในภาวะ 'busy' และได้รับโดยมี FCS ถูกต้อง นั่นคือ I-frame ซึ่งมี N(S) = V(R) DCE จะยอมรับ info-field ของเฟรมนี้ จะเพิ่ม V(R) = V(R) + 1

- ถ้า I-frame เป็นการส่งจาก DCE มันจะส่ง I-frame นี้ไปให้กับ DTE
- ถ้าไม่มี I-frame ที่จะส่งจาก DCE มันก็จะส่ง RR

ข้อสังเกตุ : เมื่อ DCE อยู่ในภาวะ 'busy' มันอาจจะไม่ยอมรับรู้ข่าวสารที่บรรจุอยู่ใน I-frame ที่ได้รับทั้งหมด

**4.6.3 การได้รับเฟรมที่ไม่ถูกต้อง**

เมื่อ DCE ได้รับเฟรมที่มี FCS ไม่ถูกต้อง เฟรมนี้จะถูกขจัดออกไป เมื่อ DCE ได้รับเฟรมที่มี FCS ถูกต้อง แต่มี N(S) ไม่ถูกต้อง นั่นคือ N(S) # V(R) มันจะขจัดข้อมูลที่บรรจุอยู่ในเฟรมและส่ง REJ response ที่มี N(R) ซึ่งตั้งไว้สูงกว่า N(S) ของเฟรมล่าสุดที่ได้รับ I-frame ( N(R) = V(R) ) ถูกต้องอยู่ 1 ขั้น

DCE จะพิจารณาข้อมูลที่บรรจุอยู่ใน I-frames ทุกเฟรมจนกระทั่งคาดว่า I-frame ที่ได้รับถูกต้อง เมื่อได้รับ I-frame ที่คาดหมายไว้ DCE จะ acknowledge เฟรมด้วย I หรือ RR-frames DCE จะใช้ N(R) และ P-bit ที่แสดงใน I-frames ที่ถูกขจัด

**4.6.4 Receiving acknowledgement**

เมื่อได้รับ I หรือ S-frame (RR,RNR,หรือ REJ) ถูกต้อง แม้ว่าจะอยู่ในภาวะ 'busy' หรือ 'frame rejection' DCE จะพิจารณา N(R) ที่บรรจุอยู่ในเฟรมนี้ตามที่ acknowledgement สำหรับ I-frames ทุกเฟรมที่มันได้ส่งออกไปโดยมี N(S) จนถึง N(R) ที่ได้รับ ลบด้วย 1

ถ้ายังมี I-frames ที่ยังตกค้างอยู่ที่ยังไม่ได้ acknowledgement มันจะเริ่มต้น timer T1 ถ้า timer หมดเวลาไป DCE จะดำเนินการส่งใหม่ในส่วนที่เกี่ยวกับ I-frames ที่ยังไม่ได้ตอบรับ (ดูหัวข้อ 4.6.5 กับ 4.6.8)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**4.6.5 Receiving reject**

เมื่อได้รับ REJ DEC จะทำให้ V(S) ของมันเท่ากับ N(R) ที่ได้รับใน REJ control field มันจะส่ง I-frame ตรงกันกับที่ต้องการที่จะให้ส่งใหม่ทันที การส่งใหม่จะทำให้สอดคล้องกับสิ่งที่ตามมา

- ถ้า DCE กำลังส่ง S หรือ U command หรือ response เมื่อ DCE ได้รับ REJ มันจะทำให้การส่งสิ้นเสร็จก่อนที่จะเริ่มต้นส่ง I-frame ที่ต้องการ

- ถ้า DCE กำลังส่ง I-frame เมื่อได้รับ REJ มันอาจจะขจัด I-frame และเริ่มต้นส่ง I-frame ที่ต้องการนั้นทันทีที่หลังจากขจัด I-frame นั้นออกไปแล้ว

ในทุกกรณี ถ้ายังไม่มีการ acknowledged I-frame อื่นๆ ที่ได้ส่งออกไปสิ่งหนึ่งที่เป็นผลตามมาที่แสดงให้รู้ในสภาพ REJ นั่นคือ I-frame เหล่านั้นจะต้องถูกส่งใหม่จาก DCE สืบเนื่องมาจากการส่งใหม่ของ I-frame ที่ต้องการ

ถ้า REJ-frame ได้รับจาก DTE เช่นเดียวกับ command มี P=1 DCE จะส่ง RR,หรือ RNR response มี F-1 ก่อนการส่งหรือการส่งใหม่ให้ตรงกันกับ I-frame ที่ต้องการ

**4.6.6 Receiving RNR**

หลังจากที่ได้รับ RNR DCE อาจส่ง I-frame มี N(S) เท่ากับ N(R) ที่ได้แสดงใน RNR-frame ถ้า timer T1 หมดไปหลังจากได้รับ RNR DCE จะดำเนินการต่อไปดังที่ได้กล่าวไว้ในหัวข้อ 4.6.8 ในทุกกรณี DCE จะไม่ส่ง I-frame อื่น ๆ ก่อนที่จะได้รับ RR หรือ REJ-frames

**4.6.7 DCE busy condition**

เมื่อ DCE เข้าสู่ภาวะ 'busy' มันจะส่ง RNR response ให้ในโอกาสแรก ในขณะที่อยู่ในภาวะ 'busy' นั้น ถ้ามันได้รับ S หรือ I-command frame ที่มี P=1 DCE จะ ตอบสนองและดำเนินการโดย S-frames และตอบด้วย RNR response โดยมี F=1 เพื่อเคลียร์ภาวะ 'busy' DCE จะส่ง REJ response หรือ RR response อย่างใดอย่างหนึ่งที่มี N(R) = V(R) ขึ้นอยู่กับ information field ว่าจะถูกขจัดออกไปหรือไม่

**4.6.8 Waiting acknowledgement**

DCE ยังคงดำเนินการต่อไปในส่วนที่เกี่ยวกับ "retransmission count variable" ซึ่งได้ตั้งไว้ที่ 0 เมื่อ DCE ได้รับ UA หรือ RNR หรือเมื่อ DEC ได้รับ I หรือ S-frames โดยมี N(R) สูงกว่าที่ได้รับ N(R) ครั้งสุดท้ายถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้า timer T1 หมดไป DCE จะเข้าสู่สภาวะ "timer recovery" เพิ่มอีก 1ให้กับ "retransmission count variable" ของมัน และตั้งค่าภายใน variable X ให้ตรงกับค่า V(S) ของมันที่มีอยู่ในปัจจุบัน

DCE จะเริ่มต้น timer T1 ใหม่ ตั้งค่า V(S) ของมันให้ตรงกับค่า N(R) สุดท้ายที่ได้รับจาก DTE และเริ่มส่งให้ใหม่ให้ตรงกับ I-frame โดยมี P=1 ภาวะของ "timer recovery" จะถูกเคลียร์เมื่อ DCE ได้รับ S-frame จาก DTE ถูกต้องโดยมี F=1

ถ้าในขณะนั้นอยู่ในภาวะ "timer recovery" DCE ได้รับเฟรมโดยมี N(R) อยู่ภายในขอบเขตจากค่า V(S) ปัจจุบันถึงค่า X ที่รวมอยู่ด้วยถูกต้อง ในขณะที่หรือก่อนที่จะเคลียริ่งภาวะ "timer recovery" มันจะยังไม่เข้าไปสู่ภาวะ rejection และมันจะปรับค่า V(S) ของมันให้ตรงกับ N(R) ที่ได้รับ

ถ้า "retransmission count variable" เท่ากับ N2(system parameter) DCE เริ่มต้นขั้นตอนการกำหนดสำหรับทิศทางของการส่งจาก DCE ใหม่ DCE อาจขอให้ DTE กำหนดการเชื่อมโยงโดยการส่ง DM response หลังจากที่ได้ส่ง DM response ไปแล้ว DCE จะเข้าสู่ภาวะ disconnected phase ดังที่ได้กล่าวมาแล้วในหัวข้อ 4.5.4

## 4.7 Procedure for resetting

วิธีการจัดลำดับใหม่ได้นำมาประยุกต์ใช้ในระหว่างขั้นตอนการส่งถ่ายข้อมูลนี้เท่านั้น DTE จะแจ้งให้ทราบถึงการจัดลำดับใหม่โดยการส่ง SABM command หลังจากได้รับ SABM command แล้ว DCE จะตอบด้วย UA response ไปให้กับ DTE และจัดลำดับ V(S) และ V(R) ของมันไว้ที่ 0 ใหม่ ในขณะเดียวกันมันจะเคลียร์ DCE และ/หรือ DTE ที่มีอยู่ให้อยู่ในภาวะว่าง "busy"

ภายใต้ภาวะ rejection บางอย่าง DCE อาจขอให้ DTE กำหนดการเชื่อมโยงใหม่ โดยการส่ง DM response หลังจากที่ส่ง DM response ไปแล้ว DCE จะเข้าสู่ภาวะ disconnection phase ดังที่ได้กล่าวมาแล้วในหัวข้อ 4.5.4

rejection condition จะเริ่มต้นขั้นตอนการจัดลำดับใหม่ มีดังนี้

- ถ้า "retransmission count variable" เท่ากับ N2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เมื่อ DCE ได้รับเฟรมที่มี FCS ถูกต้องระหว่างขั้นตอนของการส่งถ่ายข้อมูลพร้อมกับ address A และประกอบด้วยภาวะใดภาวะหนึ่งซึ่งตามมาดังนี้

- UA หรือ DM response

- ประกอบด้วย F=1 ยกเว้นภาวะระหว่าง "timer recovery" ดังได้กล่าวมาแล้วในหัวข้อ 4.6.8

ภายใต้ภาวะ rejection บางอย่าง DCE อาจขอให้ DTE จัดการเชื่อมโยงใหม่ โดยการส่ง FRMR response หลังจากที่ส่ง FRMR response ไปแล้ว DCE จะเข้าสู่ภาวะ "frame rejection" ภาวะ "frame rejection" จะถูกเคลียร์เมื่อ DCE ได้รับ SABM หรือ DISC command หรือ DM response เมื่อได้รับ command อื่น ขณะที่อยู่ในภาวะ "frame rejection" จะเป็นเหตุให้ DCE ส่ง FRMR response กลับไปใหม่พร้อมกับ information field อย่างเดียวกันกับที่ส่งในครั้งแรก

DCE จะเริ่มการจัดลำดับใหม่ ดังได้กล่าวไว้ในตอนต้น เมื่อได้รับเฟรมที่มี FCS ถูกต้องในระหว่างขั้นตอนของการส่งถ่ายข้อมูลพร้อมกับ address A หรือ B และมีภาวะใดภาวะหนึ่งดังต่อไปนี้

- เฟรมที่ไม่ทราบว่าเป็น command หรือ response

- N(R) ที่บรรจุอยู่ใน Information field ไม่ถูกต้อง

N(R) ที่ถูกต้องอยู่ภายในช่วงตั้งแต่ N(S) ต่ำสุดของเฟรมที่ยังไม่ได้ acknowledged ถึง DCE V(S) ปัจจุบันที่รวมอยู่ด้วย ถ้า DCE อยู่ในภาวะ "rejection" แต่ต้องไม่ใช่ DCE อยู่ในภาวะ "timer recovery"

## 4.8 System parameters

system parameter

- Timer T1

- จำนวนสูงสุดของการส่ง N2

- จำนวนสูงสุดของบิทใน I-frame N1

- จำนวนสูงสุดของ I-frames k ที่ยังคงค้างอยู่

parameter นี้จะต้องไม่เกิน 7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 5

# X.25 Level 3 - Network layer

## 5.1 บทนำ

Level 3 ของ X.25 ข้อแนะนำที่เกี่ยวกับ packet layer ของ DTE/DCE interface ประกอบด้วยส่วนสำคัญที่เกี่ยวกับ level 3 มีความซับซ้อนพอสมควรรวมทั้งมีจำนวนของแพคเกทชนิดต่างๆ เป็นจำนวนมากทีเดียว

เป็นความตั้งใจที่จะแนะนำขั้นตอนต่างๆ เหล่านี้ให้เป็นไปตามลำดับและวิธีการที่ถูกต้อง ข้อความ 2-3 ประโยคในที่นี้อาจจะช่วยให้ผู้อ่านเข้าใจวิธีการ และโครงสร้างโดยทั่วไปของบทนี้

**Ready phase**

ก่อนที่การสื่อสารข้อมูลระหว่าง DTE ทั้งสองจะเริ่มต้น DTE แต่ละแห่ง ขั้นแรกต้องพร้อมและเริ่มทางฟิสิกส์ด้วยตัวของมันเอง การต่อทางลอจิกคอล และแพคเกทเข้ากับ network การใช้บริการนี้เป็นอิสระไม่ว่าจะเป็นอย่างเดียวหรือทั้งสามอย่าง นั่นคือ virtual call, permanent virtual circuit และ datagram จะใช้อย่างเดียว หรือจะใช้รวมกันทั้ง 3 อย่าง เพราะฉะนั้นขั้นตอนนี้จะได้กล่าวก่อน

**Virtual Call Procedures**

การบริการทั้ง 3 อย่างนั้น virtual call นี้ซับซ้อนมากที่สุดมีหัวข้อใหญ่ๆอยู่ 3 หัวข้อเรียกว่า Call set-up phase, Data Transfer phase และ Clear phase ซึ่งจะได้กล่าวในตอนต่อไป

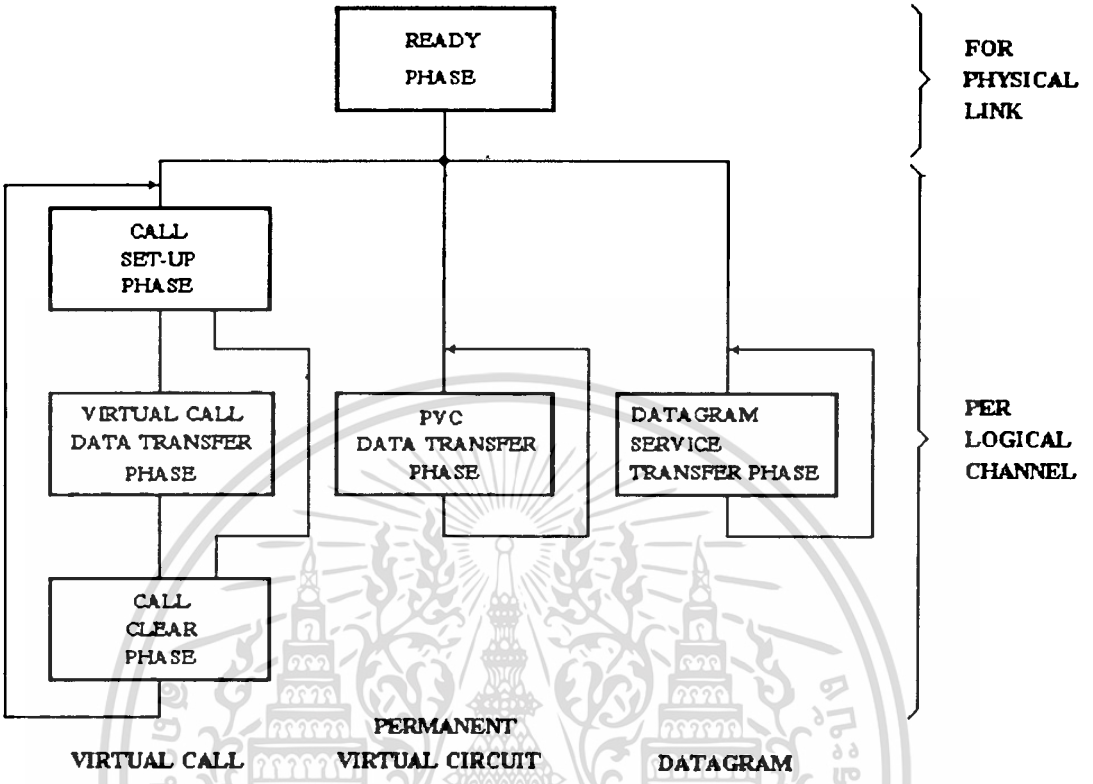
**Permanent Virtual Circuit Procedures**

PVC procedures เป็นลำดับต่อไป วิธีการโดยพื้นฐานเป็นชุดประกอบของ virtual call call set-up และ call clear phases ไม่จำเป็นต้องใช้ เพราะฉะนั้นหัวข้อนี้จะกล่าวเพียงย่อๆเท่านั้น

**Datagram Procedures**

การให้บริการ datagram ขั้นตอนบางอย่างจะต่างกันกับ virtual call มาก แต่ก็ยังมีเนื้อหาที่คาบเกี่ยวกันอยู่ call set-up และ clearing phases ไม่จำเป็นต้องใช้ data packets ก็ยังแตกต่างกันกับที่ใช้ใน virtual calls และ PVCs ด้วยเหมือนกัน ซึ่งจะได้กล่าวในหัวข้อต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 14 โครงสร้างโดยทั่วไปของ Network Layer

## 5.2 Ready Phase

### 5.2.1 ทั่วไป

Logical channel ต้องอยู่ในภาวะ "READY" ก่อนที่จะเข้าสู่ขั้นตอนของการเรียกและการส่งถ่ายข้อมูล ก่อนที่จะเข้าสู่ภาวะนี้ physical และ link level ทั้งสองนี้จะต้องอยู่ในภาวะที่ถูกต้องเสียก่อน และการดำเนินการในขั้นตอนต่อไปถึงจะเริ่มดำเนินการได้

### 5.2.2 Physical level

การต่อเชื่อมทางฟิสิกส์ใน packet-switching network กระทำอย่างเดียวกันกับวงจรเช่าวงจรเช่าใน X.21 และ X.21-bit มีแต่ Quiescent และ data transfer phases เท่านั้น

Physical level ต้องอยู่ในภาวะ "READY FOR DATA" ใน X.21 c ต้อง ON และ i = ON วงจร T และ R สำหรับการส่งถ่ายข้อมูลจะถูกกำหนดโดยระดับที่สูงกว่า สำหรับ X.21-bi52525 105,106,107,108 และ 109 ต้อง ON ส่วนวงจร 103 และ 104 จะถูกกำหนดโดยระดับที่สูงกว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**5.2.3 Link level**

สำหรับ link level ใช้ LAP-B protocal และ unnumbered frames เพื่อกำหนด operating mode และกำหนด logical link ใน LAP-B operating mode เป็นแบบ asynchronous balanced mode แพคเกทต่างๆ จากระดับสูงจะส่งผ่าน LAP-B I-frames นี้เท่านั้น

**5.2.4 Packet level**

ขณะนี้ packet level อยู่ในภาวะ "PACKET LEVEL READY" อย่างไรก็ตามก่อนหน้านี้ logical channel ทั้งหมดต้องอยู่ในภาวะ "READY" โดยวิธีการของ Restart Procedure

**5.2.5 Restart Procedure**

Restart procedure ใช้เพื่อเริ่มต้นหรือมีการเริ่มต้นใหม่ของ packet level DTE/DCE interface วิธีการนี้เป็นการเคลียร์ Virtual calls และ เป็นการเริ่มต้นใหม่ทั้งหมดของ permanent virtual circuits และ datagram channels ที่ DTE/DCE interface

Restart โดย DTE

DTE อาจต้องการเริ่มต้นใหม่โดยการส่ง Restart Request packet ผ่าน DTE/DCE interface หมายเลขของ logical channel ตั้งไว้ที่ 0 เช่นเดียวกับแพคเกทที่นำมาใช้กับทุก logical channel แชนแนลทั้งหมดเหล่านี้บรรจุอยู่ในภาวะ DTE RESTART REQUEST

Restart โดย DCE

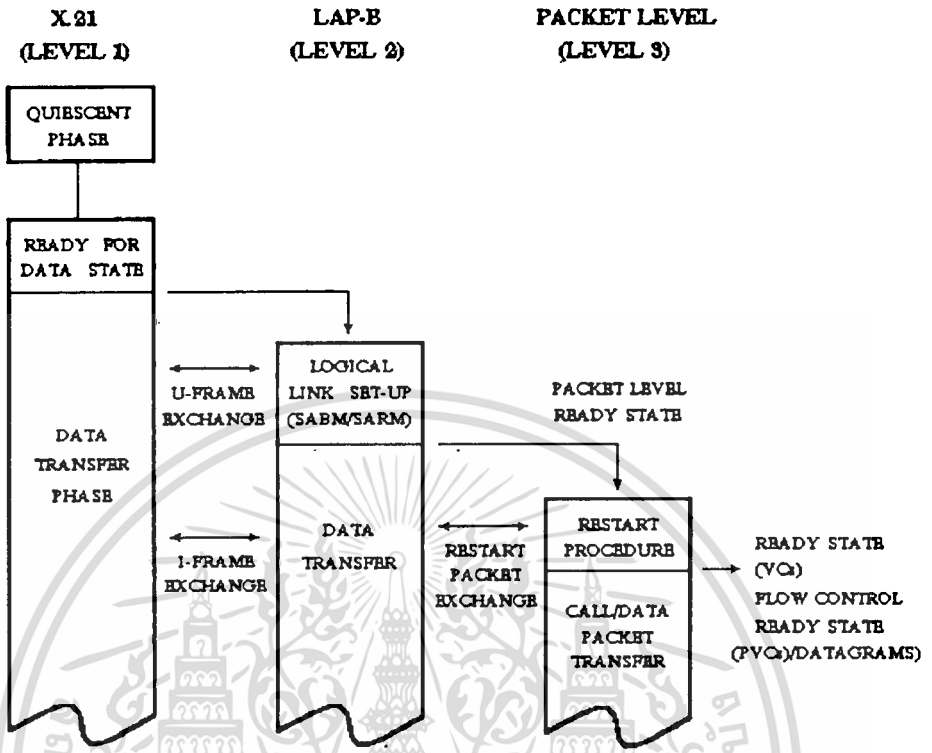
DCE สามารถทำให้เกิดการเริ่มต้นสำหรับ restart procedure ได้เช่นเดียวกัน DCE จะส่ง Restart Indication packet ไปให้กับ DTE ดังนั้นตำแหน่งของ logical channel ต่างๆจะอยู่ในภาวะ DCE RESTART INDICATION

DTE ตอบด้วย DTE Restart Confirmation packet สำหรับ Virtual calls ตำแหน่งของ logical channel ต่างๆจะอยู่ในภาวะ READY และ แชนแนลของ permanent virtual circuits และ datagrams จะอยู่ในภาวะ FLOW CONTROL READY

Restart Collision

Restart Collision เกิดขึ้นเมื่อ DTE และ DCE ส่ง Restart Request packet และ Restart Indication packet พร้อมๆกัน DCE จะพิจราณาการเริ่มต้นใหม่ได้กระทำเรียบร้อยแล้วและไม่จะเป็นต้องส่ง confirmation packet อีก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**รูปที่ 15 Ready Phase**

## 5.3 Virtual call procedure

### 5.3.1 Call set-up phase

#### 5.3.1.1 Call request packet

ขณะที่กำหนดให้ DTE ทำการเรียก DTE อื่นๆในโครงข่ายนั้น level 3 ของ calling DTE จะได้รับข้อมูลการเรียกที่จำเป็นจาก level 4 ของมันเอง level 3 จะสร้าง Call Request packet ซึ่งจะส่งผ่าน DTE/DCE interface ไปให้กับ local node ใน HDLC frame ที่วงจร T รูปแบบของ Call Request packet ได้แสดงไว้ดังรูปที่ 16 calling DTE จะส่งแพคเกทไปที่ logical channel ที่มีหมายเลขสูงสุดที่อยู่ในภาวะ READY logical channel นี้จะเดินทางไปที่ภาวะของ DTE WAITING

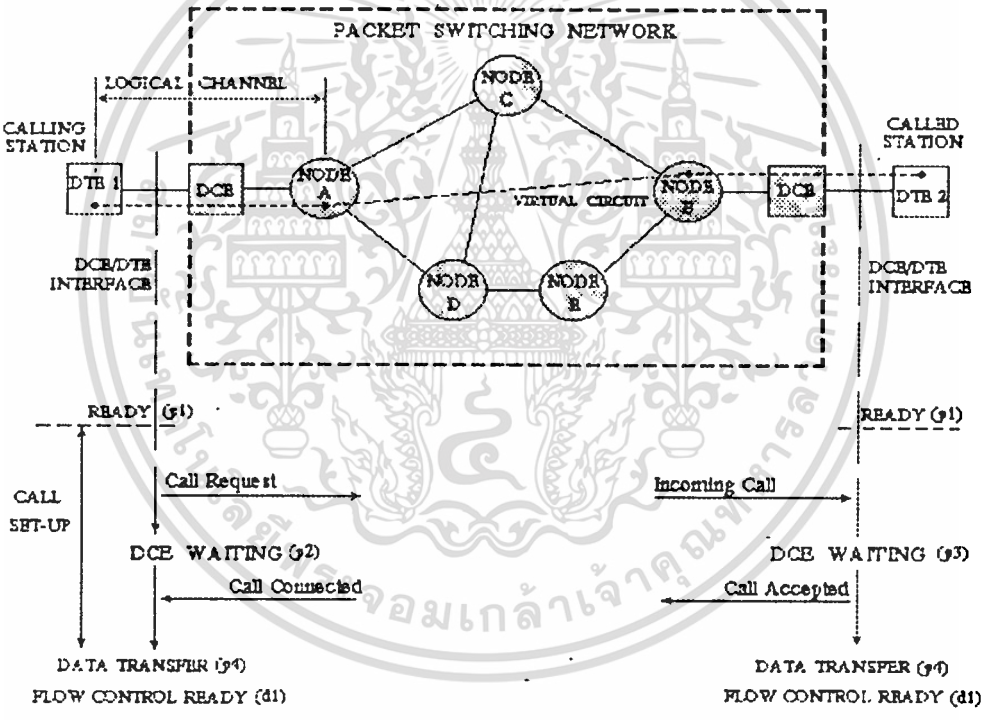
Call Request packet จะบรรจุแอดเดรสของ called DTE แอดเดรสนี้อาจเป็นแบบ full DTE network address,abbreviated address หรือ เป็นการกำหนดชื่อของ DTE เป็นอย่างอื่นก็ได้ตามที่ได้ทำการตกลงกันระหว่าง DTE กับโครงข่าย (network)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3.1.2 Incoming call packet

ถ้า calling DTE ได้รับความยินยอมให้ติดต่อกับ Called DTE ได้ โครงข่ายจะส่งข้อมูลที่จำเป็นสำหรับ Call Request packet ไปให้กับ distant node ซึ่งมี DTE ปลายทางต่ออยู่ โหนดนี้จะผ่าน Incoming Call packet ไปยัง called DTE ซึ่งเป็นสัญญาณของการเรียก แพคเกทนี้มีรูปแบบเดียวกันกับ Call Request packet Incoming Call packet จะส่งผ่านไปยัง logical channel ที่มีหมายเลขต่ำสุดที่อยู่ในภาวะ READY

ข้อสังเกตุ : การส่ง Call Requests ไปใน logical channel ที่หมายเลขสูงสุด และ Incoming Calls ที่ logical channel ที่หมายเลขต่ำสุดนั้นจะทำให้เกิด collision ลดลงได้มาก




**รูปที่ 16 Call Set-Up Procedure**

5.3.1.3 Call accepted packet

ถ้า called DTE ตกลงจะแลกเปลี่ยนข้อมูลกับ calling DTE แล้ว มันจะตอบด้วย Call Accepted packets ส่งผ่านไป logicall channel เดียวกันกับที่ใช้กับ Incoming Call packet logical channel นี้จะสับไปที่ภาวะของ DATA TRANSFER called DTE ถูกจำกัดเวลาไว้เพียง 3 นาที ที่จะสร้าง Call Accepted packet ถ้าเกินเวลากว่านี้โครงข่ายจะเข้าสู่ขั้นตอนของการเคลียร์เพื่อตัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Virtual call ออก โครงข่ายจะเข้าสู่ขั้นตอนเคลียร์ เช่นเดียวกันถ้า called DTE ปฏิเสธการเรียก โดยส่ง Clear Request packet (จะได้กล่าวในตอนต่อไป)

5.3.1.4 Call_connected_packet

เมื่อได้รับ Call Accepted packet distant node จะส่ง ข้อมูลที่จำเป็นไปให้กับ node ที่ calling DTE ต่ออยู่ node นี้จะส่ง Call Connected packet ไปให้กับ calling DTE แพคเกทนี้จะมีรูปแบบอย่างเดียวกันกับ Call Accepted packet Call Connected packet จะส่งผ่าน logical channel เดียวกันกับที่ใช้กับ Call Request แชนแนลนี้จะถูกสับไปยังภาวะของ DATA TRANSFER logical channel นี้อาจจะอยู่ในภาวะ DTE WATTING เป็นการคอยสำหรับ Call Connected packet ช่วงระยะเวลานี้ต้องไม่เกิน 200 วินาที

5.3.1.5 End-to-end protocal

โปรโตคอลที่ใช้กับขั้นตอนของ set-up นี้เป็นการแลกเปลี่ยนข่าวสารระหว่าง DTE 2 DTE ด้วยกัน เพราะฉะนั้นพิจารณาได้ว่าเป็น end-to-end protocal

5.3.1.6 DTE addresses

Address structure

DTE address มีจำนวนถึง 14 หลัก ตามแผนภาพที่ 17 ได้แสดงถึงการแบ่งแอดเดรสออกเป็น 3 ส่วนด้วยกัน ดังนี้

The Data Network Identification Code (DNIC)

The Network Number (NN) - จะเหลื่อมกับ DNIC อยู่ 1 หลัก และ

The optional subaddress

The Data Network Identification Code (DNIC)

หลักที่ 1-3 : รหัสของประเทศ

หลักที่ 4 : รหัสของ network ที่อยู่ภายในประเทศ

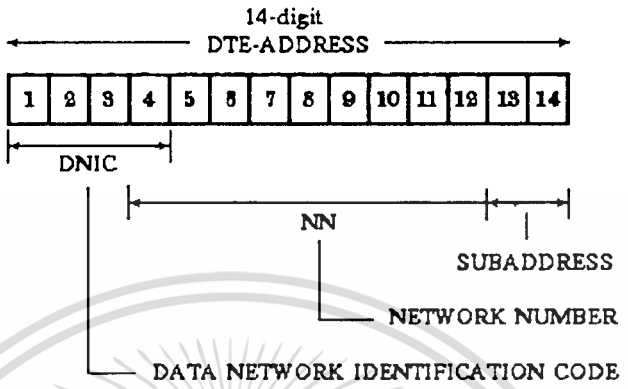
The Network Number (NN)

หลักที่ 4-12 : หมายเลขของ DTE ที่อยู่ภายในประเทศ

Subaddress

หลักที่ 13-14 : จัดไว้ให้ DTE เลือกใช้ได้ที่เกี่ยวกับ per-call basic และเมื่อมีใช้จะส่งผ่านตรงไปยัง network อาจจะนำมาใช้กับแอดเดรสเพียง 1 หรือ 2 หลัก โดยเฉพาะนำมาใช้ภายใน called DTE เมื่อมีการกำหนดการเรียก network จะไม่ตรวจสอบความถูกต้องของ subaddress หมายความว่าเป็นการกำหนดกันขึ้นมาเองระหว่างผู้ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 17 DTE ADDRESSING IN X.25

**DTE Address ใน Call Request และ Incoming Call Packets**

Address_lengths_field :

ออคเททที่ 4 ประกอบด้วย field length สำหรับ called และ calling DTE ในครึ่งออคเทตแรกบิทที่ 4,3,2 และ 1 บอกถึงความยาวของ called DTE address ครึ่งออคเททหลังบิทที่ 8,7,6 และ 5 บอกถึงความยาวของ calling DTE address ความยาวของแต่ละแอดเดรสบอกให้ทราบด้วย binary code และ บิทที่ 1 หรือ 5 มีค่าทางตัวเลขต่ำ

Address field :

ออคเททที่ 5 และ ออคเททต่อไปประกอบด้วย called DTE address ในขณะที่มีอยู่ต่อไปเป็น calling DTE address ในขณะที่มีอยู่

แต่ละหลักของแอดเดรสในครึ่งออคเททรหัสเป็นแบบ binary coded decimal โดยมีบิทที่ 5 หรือ 1 เป็นบิทที่มีค่าต่ำของหลัก

เริ่มจากหลักที่มีลำดับสูงก่อน แอดเดรสจะถูกเข้ารหัสในออคเททที่ 5 และ ออคเททต่อไปตามลำดับแต่ละออคเททจะมี 2 หลัก ในแต่ละออคเททหลักที่มีค่าทางตัวเลขสูงบิทที่ถูกเข้ารหัสคือ บิทที่ 8,7,6 และ 5

Address filed ถูกไล่เป็นวงกลมเป็นตัวเลขจำนวนเต็มของออคเททโดยการสอดบิท 0 ในบิทที่ 4,3,2 และ 1 เมื่อต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**5.3.2 Data transfer phase**

5.3.2.1 ทั่วไป

ใน DATA TRANSFER state เท่านั้นจะมี logical channel 2 channel ที่อยู่ในภาวะที่ยอมให้ DTEs แลกเปลี่ยน Data packets กันได้

5.3.2.2 Data packets

รูปแบบของ DTE/DCE Data packets มี fields และ indicators จำนวนมากซึ่งจะได้กล่าวตามลำดับไป

ถึงแม้จะกล่าวว่าการแลกเปลี่ยน Data packets เป็นการแลกเปลี่ยนกันระหว่าง DTEs แต่ก็ไม่ได้หมายความว่าจะไม่มีการเปลี่ยนแปลงเมื่อมันส่งผ่าน network เนื่องจากมีการควบคุมข้อมูลบางอย่างเช่น logical channel number ซึ่งจะมีความสำคัญต่อ local DTE/DCE interface เท่านั้นแน่ละจะมีแต่เพียงข้อมูลของมันเองเท่านั้นที่ส่งผ่านโดยไม่มีการเปลี่ยนแปลง X.25 ไม่ได้กำหนดรูปแบบของแพคเกทที่ส่งผ่านระหว่าง end-nodes

5.3.2.3 Packet identifier

Data packets ของออคเททที่ 3 ของแพคเกทมีบิทที่ 1 เป็น 0 ในแพคเกทอื่นๆ ทั้งหมดบิทนี้จะเป็น 1 เนื่องจากบางครั้ง Data packets นี้อ้างถึง "even"packets และแพคเกทอื่นๆเป็น "old" packets ในออคเททที่ 3 ของ Data packets ยังคงเหลืออยู่อีก 7 บิท มีไว้สำหรับ packet numbering ซึ่งจะได้กล่าวถึงอย่างย่อๆ

5.3.2.4 Data field

โดยทั่วไปความยาวสูงสุดของ user data field ที่เสนอแนะคือ 128 ออคเทท แต่ก็ยังให้เลือกใช้ได้คือ 16,32,64,256,215 และ 1024 ออคเทท การเลือกใช้นี้อาจจะมีการตกลงกันแต่ละครั้งการเรียก

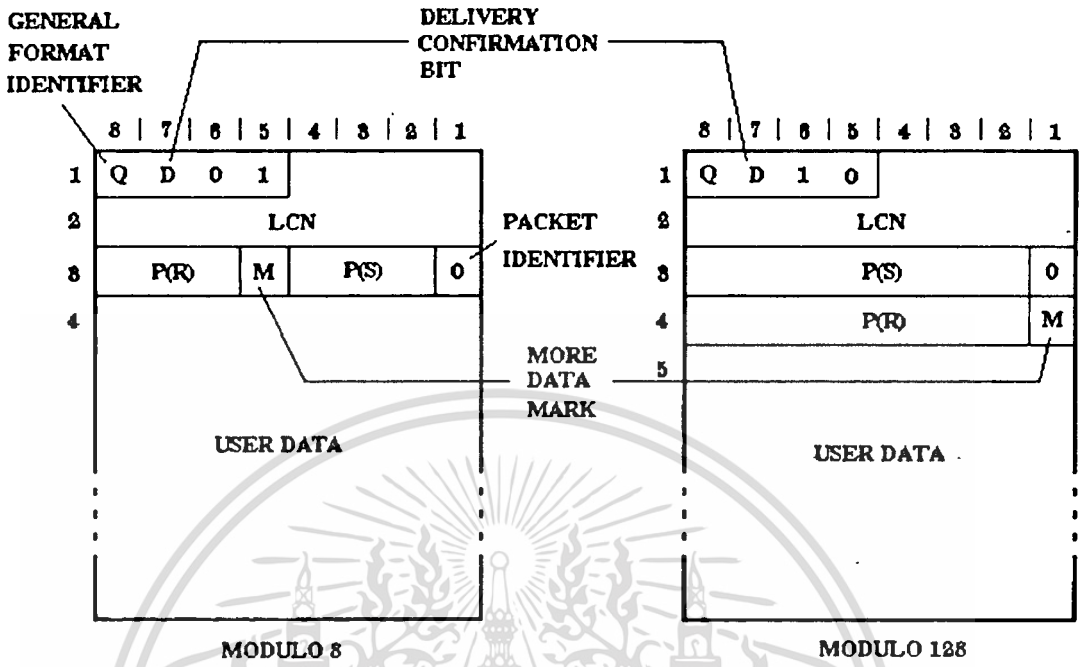
แพคเกทอาจบรรจุจำนวนบิทได้ถึงจำนวนสูงสุดที่ได้มีการตกลงกันไว้

5.3.2.5 Delivery confirmation bit (D-bit)

การกำหนด Delivery Confirmation bit (D-bit) ใช้เพื่อแจ้งให้ทราบว่า DTE ประสงค์ที่จะรับ end-to-end acknowledgement ของการนำส่งสำหรับข้อมูลที่มันกำลังอยู่หรือไม่

สำหรับ Data packets confermation ส่งผ่าน Packet Receiver Number P (R) หมายเลขนี้จะกล่าวต่อไปในการอธิบายถึงเรื่อง flow control

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**รูปที่ 18 DTE and DCE data packet**

5.3.2.6 The More data mark (M-bit) และ Packet Sequences

ถ้า DTE หรือ DCE ประสงค์ที่จะแจ้งลำดับของการส่งแพคเกทที่มากกว่าหนึ่งแพคเกท มันจะใช้ More Data Mark (M-bit) ถ้าแพคเกทไม่ใช่เป็นส่วนที่ต่อเนื่องกัน M=0 เมื่อ M=1 มันจะบอกให้ทราบว่ายังมีข้อมูลแพคเกทตามมาอีก

แพคเกทที่ส่งโดยมี M=0 ถูกนำส่งไปยังสถานีปลายทาง ถ้าสถานีปลายทางมีความเท่ากันหรือมากกว่า data field ที่นำส่ง D-bit ยังมีค่าเท่ากับต้นทาง ถ้าสถานีปลายทางมีความยาวของ data field สั้นกว่าสถานีต้นทาง ดังนั้น network จะแบ่งแพคเกทของต้นทางออกเป็นส่วนๆ สำหรับการนำส่งแพคเกทที่ประกอบกันเป็น 1 แพคเกทของแพคเกทต้นทางก็จะกลายเป็นแพคเกทที่เรียงกันตามลำดับโดยมี M=1 ในแพคเกททั้งหมด ยกเว้นแพคเกทสุดท้ายที่มี M=0

ที่เรียกว่า complete packet sequence นั้น แพคเกทยังแบ่งออกเป็น 2 ประเภท คือ ประเภท A แพคเกทประกอบด้วย full data field มี M bit = 1 และ D bit = 0 ประเภท B แพคเกทอาจจะหรืออาจจะไม่ประกอบด้วย full data field ก็ได้ โดยมี M bit = 0 และ D bit = 1 หรือ 0 ตามที่กำหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

complete packet sequence ประกอบด้วย ประเภท A แพคเกทที่มีหนึ่งแพคเกทหรือมากกว่าและแพคเกท B อีกหนึ่งแพคเกท ถ้า source DTE ส่งแพคเกทต่อเนื่องกัน ลำดับที่จะนำส่งไปยัง DTE ปลายทางจะไม่เปลี่ยนแปลง ถ้า DTEs ทั้งสองมีแพคเกทที่ทีความยาวของ data field สูงสุดเท่ากัน ถ้าปลายทางมีความยาวของ data field มากกว่าแพคเกทต้นทางอาจจะรวมกัน

ในทางกลับกัน ถ้าปลายทางมีความยาวของ data field สั้นกว่าต้นทาง network ต้องแบ่งออกเป็นส่วนๆ

5.3.2.7 Qualfier bit (Q-bit)

Complete packet sequence อาจจะเป็น 1 ในจำนวน 2 ระดับ DTE ต้องการส่งข้อมูลมากกว่า 1 ระดับ มันจะใช้บอกให้ทราบที่เรียกว่า Qualifier bit หรือ Q-bit ถ้าเป็นเพียง 1 ระดับใช้ Q=0 เสมอ ถ้าเป็น 2 ระดับ ทุกแพคเกทใน complete packet sequence Q-bit ควรจะกำหนดค่าอย่างเดียวกัน ไม่ว่าจะเป็น 0 หรือ 1 ลำดับที่น่าส่งควรจะมีค่า Q-bit อย่างเดียวกัน

5.3.2.8 Flow_control

**ทั่วไป**

ที่ DTE/DCE interface logical channel ที่ใช้ส่ง Data packets การควบคุมจะแยกออกไปแต่ละทาง ซึ่งดำเนินการโดยเครื่องรับ หมายความว่าเครื่องรับสามารถป้องกันตัวมันเองจากการท่วมท้นของ incoming data packet ที่มีอัตราความเร็วสูงมากได้

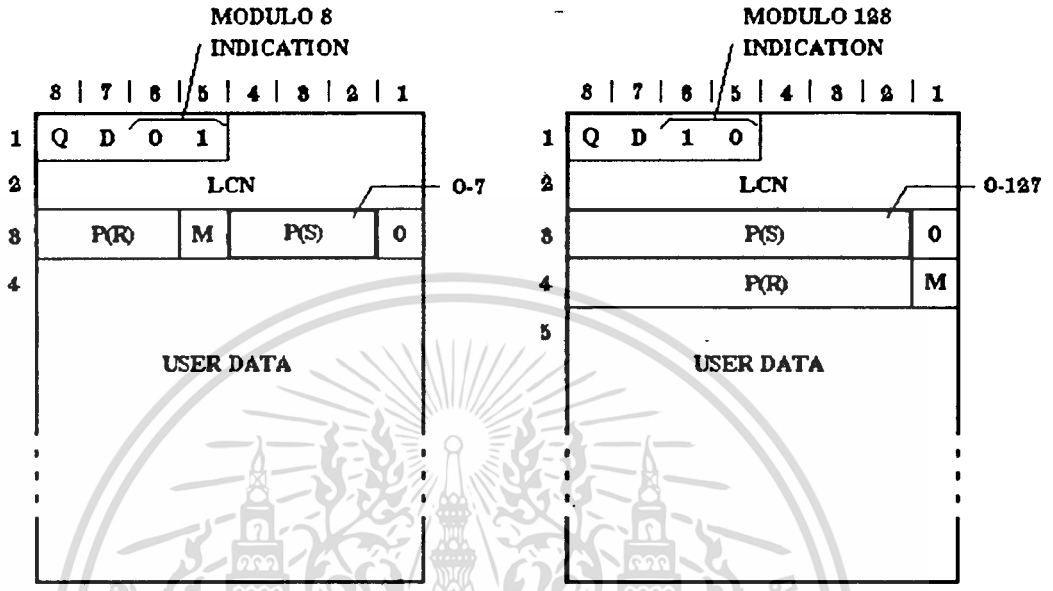
DTE ที่กำลังรับสามารถจำกัดอัตราความเร็วของการส่งจาก DTE ต้นทาง end-to-end โดยการขับยั้งการส่งจาก network network ควรจะกำหนดจำนวนของแพคเกทซึ่งอาจจะส่งผ่านที่เกี่ยวกับ virtual circuit ทั้งหมด และเมื่อจำนวนของ data packets ขึ้นมาถึงขีดจำกัด network จะขับยั้ง DTE ต้นทางไว้

**Numbering ของ Data Packets**

แต่ละ Data packet ที่ logical channel จะกำหนด send sequence number P(S) ทางด้านส่ง ลำดับหมายเลขนี้นำมาใช้กับ DTE/DCE interface เท่านั้น ไม่ใช้กับ end-to-end ระบบหมายเลขปกติเป็นแบบ module 8 P(S) เป็นวงรอบในย่าน 0-7 (modulo 128 ย่าน 0-127 อาจจะนำมาใช้ได้)

Data packet แรกที่ส่งไปที่ logical channel หลังจากนั้นมันจะเข้าไปสู่ภาวะ FLOW CON-TROL READY (sub-sate ของ DATA TRANSFER state) มี P(S) = 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 19 PACKET SEND SEQUENCE NUMBER P(S)

**Window Mechanism**

Window ได้กำหนดด้วย W ที่ประกอบด้วย packet send sequence numbers P(S) ของ Data packets ใน logical channel ผ่านจุดเชื่อมโยงในทิศทางหนึ่ง หมายเลขต่ำสุดใน window เรียกว่า lower window edge ครั้งแรก virtual call ได้มีการกำหนด lower window edge ทั้งสองทิศทางไว้ที่ 0 (นั่นคือ Data packet แรกจะมีหมายเลข P(S) = 0) Window size W นี้ให้เลือกใช้ได้ แต่ที่กำหนดคือ 2 ถ้า packet send sequence number ของ Data packet แรกไม่ได้กำหนดไว้ใน logical channel ที่ผ่านจุดเชื่อมโยง ค่าของ lower window edge บวกกับ W (module 8 หรือ 128) เช่นตัวอย่าง ถ้า W=2 และ lower window edge คือ 0 แพคเกทแรกที่ไม่ได้กำหนดจะมีค่า P(S) = 2 เมื่อส่งมาถึงจุดนี้จะหยุดส่ง

เพื่อให้เข้าใจเพิ่มขึ้น เครื่องรับต้องเลื่อน window ถ้าการส่งเริ่มต้นใหม่ เครื่องรับจะใช้ packet receive sequence number P(R) หมายเลขนี้อาจจะซ้อนกันอยู่ใน Data packet ในทิศทางตรงข้ามกับการส่ง หรือการส่งถ่ายใน Receive Ready หรือ Receive Not Ready packet ค่าของ P(R) เปลี่ยนเป็น lower window edge ใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

P(R) ที่มีค่าน้อยกว่าหรือ เท่ากับ sequence number ของ Data packet ที่คาดว่าจะมีการส่งของ Data packet และพิสูจน์ได้ว่า DTE หรือ DCE ส่ง P(R) ไปเป็นการยอมรับว่า Data packets ทั้งหมดจนถึงและรวมด้วย P(R)-1 โดยการเลื่อน window ส่งกลับไปให้

ถ้าไม่ได้มีการกำหนดขนาดของ window ไว้ก่อน เครื่องส่งจะเคลื่อนไปโดยไม่มีการหยุดชะงัก

**Delivery Confirmation**

เมื่อ DTE ต้นทางส่ง Data packet โดยมี D-bit ตั้งที่ 1 และมี P(S) = p ความสำคัญของการที่ตอบด้วย P(R) ให้ตรงกับ Data packet นั้น (นั่นคือ P(R) จะมากกว่าหรือ เท่ากับ p+1) แสดงว่า P(R) ที่ได้รับจาก remote DTE ที่เกี่ยวกับ Data packet ที่ส่งมาตอนแรก หมายความว่าการปรับปรุง local window จะขึ้นอยู่กับเวลาที่ใช้ในการเดินทางไประหว่าง DTE-to-DTE ที่ผ่าน network เพราะฉะนั้นการใช้ D-bit ตลอดเวลาจะทำให้การใช้ที่เกี่ยวกับ network ไม่มีประสิทธิภาพหรือต้องการขนาดของ window เพิ่มขึ้นเพื่อให้สอดคล้องกับ buffer capacity ที่เพิ่มขึ้นใน network กับ DTEs

**DTE/DCE Receive Ready (RR) Rackets**

RR packets ที่ใช้กับ DTE หรือ DCE บอกให้ทราบว่าพร้อมที่จะรับ W Data packets ที่อยู่ภายใน window เริ่มด้วย P(R) ซึ่ง P(R) จะแสดงให้ทราบใน RR packets

RR packet ได้นำมาใช้สำหรับในกรณีที่ไม่มี Data packet ที่จะนำส่ง

**DTE/DCE Receive Not Ready (RNR) packets**

RNR packets ที่ใช้กับ DTE หรือ DCE บอกให้ทราบว่าไม่สามารถสนองต่อ Data packets ที่เพิ่มขึ้นที่ logical channel ที่กำหนดไว้ได้ชั่วคราว lower window edge ได้มีการปรับปรุงค่า P(R) ที่อยู่ใน RNR packet แต่ทางด้านส่งต้องหยุดการส่งชั่วคราว RR packet สามารถใช้เพื่อเคลียร์ภาวะ RNR ได้

5.3.2.9 Interrupt procedure

**ทั่วไป**

Interrupt procedure ยอมให้ DTE ส่งข้อมูลไปให้กับ remote DTE ได้ โดยไม่ต้องมี flow control procedure สนับสนุนอย่างที่ใช้กับ Data packets

Interrupt procedure นำมาใช้กับ FLOW CONTROL READY substate (d1) ของ DATA TRANSFER state (p4) เท่านั้น ไม่มีผลกระทบเกี่ยวกับ flow control procedure

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Interrupt packet**

การระงับการส่งนั้น sending DTE จะส่ง Interrupt packet ไปให้กับ remote DTE DTE จะไม่ส่ง interrupt อีกจนกว่าการส่งไปครั้งนั้นจะได้รับการยืนยัน DCE ที่ remote end ส่งผ่าน Interrupt packet ที่ประกอบด้วยข้อมูลจาก source DTEInterrupt packet ไปให้กับ remote DTE

**Interrupt Confirmation Packet**

Remote DTE ได้รับ interrupt แล้วก็แจ้งให้ทราบโดยการส่ง Interrupt Confirmation packet ไปให้กับ source DTE

สุดท้าย DCE ที่ source end ส่ง Interrupt Confirmation packet ไปให้กับ source DTE และยุติการดำเนินการ

5.3.2.10 DTE Originated restart procedure

ทั่วไป

เป็นไปได้เหมือนกันที่การสื่อสารระหว่าง DTEs สูญเสียการควบคุมที่เกี่ยวกับ Data flow ไปชั่วขณะและทำให้เกิดการสับสนอย่างแน่นอนและจะทำให้มันกลับคืนสู่สภาพเดิมได้อย่างไรในสถานะการณ์อย่างนี้ DTE ใด DTE หนึ่งหรือทั้งสองอาจตัดสินใจที่จะเริ่มต้น virtual call ใหม่ได้

ขั้นตอนการ reset อาจเกิดขึ้นได้เมื่อ DTE ทั้งสองอยู่ใน FLOW CONTROL READY substate ของ DATA TRANSFER state เท่านั้น

การเริ่มต้นใหม่เพื่อขจัด Data และ Interrupt packets ทั้งหมดที่มีอยู่ในทั้งสองทางใน network สำหรับ virtual call โดยเฉพาะอย่างยิ่ง lower windowedge ที่ DTE ทั้งสองจะ reset ที่ 0

**Reset Request Packet**

DTE จะแสดงความต้องการสำหรับ reset โดยการส่ง Reset Request packet ไปยัง logical channel ที่เกี่ยวข้อง logical channel ไปที่ DTE RESET REQUEST state DTE จะกำหนด Resetting Cause bits ที่ 0 และสามารถเลือกใช้ที่จะกำหนด diagnostic code ที่จะส่งผ่านไปยัง remote DTE รหัสเหล่านี้จะกล่าวต่อไป

**Reset Indication Packet**

Network จะตอบสนองต่อ Reset Request packet โดยการส่ง Request Indication packet ไปให้กับ distant DTE แพคเกทนี้จะประกอบด้วย Resetting cause และ diagnostic code (อย่างใดอย่างหนึ่ง) จาก Reset Request packet ตอนนี้ logical channel เข้าสู่ DCE RESET INDICATION state ที่ภาคนี้ DCE ไม่สนใจข้อมูล RR และ RNR packets ทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Reset Confirmation Packet**

Remote DTE จะดำเนินการกับ Reset Request packet และตอบด้วย DTE Reset Confir mation packet ไปให้กับ network logical channel เข้าสู่ FLOW CONTROL READY state

Network ส่งกลับไปให้ DTE ด้วย DCE Reset Confirmation packet reset procedure ก็เริ่มต้น ขณะนี้ logical channel ไปที่ FLOW CONTROL READY state และการไหลของข้อมูลก็เริ่มต้นดำเนินการได้ใหม่อีก

5.3.2.11 Reset initiated by the network

ในบาง network อาจจะมีภาวะการณ์บางอย่างเกิดขึ้น เช่นการเชื่อมโยงหยุดชะงัก ซึ่งเป็นเหตุให้ network ต้องมีการปรับ virtual call ใหม่ ในกรณีนี้ DTE ทั้งสองจะได้รับ Reset Indication packet พร้อมกับ cause และ diagnostic fields ที่จัดไว้ให้

DTE ทั้งสองต้องตอบด้วย Reset Confirmation packets

5.3.2.12 Reset collisions

Reset collision เกิดขึ้นเมื่อ DTE และ DCE ส่ง Reset Request packet และ Reset Indication packet พร้อมกันและกำหนด logical channel เดียวกัน DCE คาดว่าจะไม่ได้รับ DTE Reset Confirmation packet และจะไม่ส่ง DCE Reset Confirmation packet ตอนนี้ logical channel ไปที่ FLOW CONTROL READY state

**5.3.3 Call clearing phase**

5.3.3.1 บทนำ

ในระหว่างที่มีภาวะปกติ Call clearing คือการปลดวงจรของ virtual call หลังจากที่การส่งข้อมูลได้เสร็จสิ้นแล้ว

ยังมีเหตุการณ์อื่นๆอีกมากที่มีผลต่อ clearing procedures ประการแรกมีเหตุการณ์ที่ call set-up ไม่สำเร็จ ประการที่ 2 called DTE อาจปฏิเสธไม่ยอมรับของการเรียก ประการที่ 3 calling DTE อาจจะยกเลิกการเรียกก่อนที่มันจะทำให้เกิดผลสำเร็จ

สาเหตุต่างๆ เหล่านี้จะได้กล่าวต่อไปดังนี้

5.3.3.2 DTE เริ่มเคลียร์ริ่ง หลังจากการส่งถ่ายข้อมูล

<u>ขั้นตอนของ DTE Call Clearing โดยทั่วไป</u>

โดยทั่วไปแล้วขั้นตอนของเคลียร์ริ่งถือว่าเป็นการกระทำที่มีผลต่อภายในทั้งหมดนั่นคือกล่าวได้ว่าไม่มีผลที่เกิดจาก end-to-end ในทางปฏิบัติ และ DTE ทั้งสองเริ่มเคลียร์ริ่ง DTE/DCE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

interface ด้วยตัวของมันเอง ขั้นตอนการสิ้นสุดของการส่งถ่ายข้อมูลปกติแล้วได้มาจากซอฟแวร์ที่มีระดับที่สูงกว่าใน DTE ทั้งสอง

DTE แต่ละแห่งส่ง Clear Request packet ผ่านไปสู่ DTE Clear Request state (p6) เมื่อ local node ได้รับคำขอมันจะส่ง DCE Clear Confirmation packet ไปให้กับ DTE ดังนั้นการเชื่อมโยงย้อนกลับไปสู่ READY state (p1) ใหม่

Call Clearing ที่มีผลต่อ End-to-end

Network บางแห่งจัดหา Clear Procedure ที่มีผลต่อ end-to-end ถึงแม้ว่าจะไม่ได้กำหนดความต้องการจาก X.25 recommendation ก็ตาม

DTE 1 ใน 2 สถานีที่เกี่ยวพันกับ virtual call อาจเริ่มดำเนินการโดยการส่ง Clear Request packet ไปให้กับ network network จะส่ง Clear Indication packet ไปให้กับ distant DCE distant DCE เชื่อมโยงไปสู่ DCE Clear Indication state (p7) distant ต้องตอบด้วย DTE Clear Confirmation packet และการเชื่อมโยงจะกลับไปสู่ READY state(p1) DTE เริ่มดำเนินการทันทีเมื่อได้รับ DCE Clear Confirmation packet แสดงว่าทั้ง local และ distant interfaces ได้ปลดวงจรออกแล้ว การเชื่อมโยงภายในขณะนี้มีสภาพในสภาวะ READY Clearing Cause field ใน Clear Request และ Clear Indication packet จะบรรจุรหัส 00 แสดงว่าการปฏิบัติเริ่มจาก DTE

ในขณะที่สาเหตุของเคลียริ่งเริ่มจาก DTE diagnostic code field ใน Clear Request packet จะไม่มีการเปลี่ยนแปลงเพิ่มขึ้นกับ Clear Indication packet

Time Limit

ในทุกกรณีที่เวลาถูกใช้ใน DTE CLEAR REQUEST state (p6) โดย logical channel ต้องไม่เกิน 3 นาที

5.3.3.3 Unsuccessful call

สมมติว่ามีเหตุผลบางอย่างที่การเรียกไม่สามารถกำหนดได้ network จะตอบด้วย Clear Indication packet ระบุ logical channel ที่เกี่ยวข้อง Clearing cause field จะบรรจุรหัสที่บอกเหตุผลของการเรียกที่ไม่สามารถติดต่อได้ สาเหตุเหล่านี้เรียกว่า "Call Progress Signals" DTE ต้องตอบด้วย DTE Clear Confirmation packet

5.3.3.4 Call collision

Call collision เกิดขึ้นเมื่อ DTE และ DCE ส่ง Call Request packet และ Incoming Call packet ในเวลาเดียวกันระบุ logical channel เดียวกัน DCE จะดำเนินการที่เกี่ยวกับ Call request และตัด Incoming Call ออกไป อันหลังนี้ถือได้ว่าเป็น unsuccessfull call ดังที่ได้กล่าวมาแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3.3.5 Call refusal

DTE อาจจะปฏิเสธ incoming call โดยการตอบด้วย Clear Request packet แทนที่จะตอบด้วย Call Accept packet ใน logical channel ที่เกี่ยวข้อง ผลก็คือ calling DTE จะได้รับ Clear Indication packet แทน Call Connected packet จาก network calling DTE ต้องตอบด้วย DTE Clear Confirmation packet ทางด้าน calling DTE คล้ายกับว่าจะไม่ประสบความสำเร็จในการเรียกแต่ clearing cause จะบอกถึงสาเหตุให้กับ DTE originated ทราบ

5.3.3.6 Call abort

DTE สามารถที่จะเริ่มดำเนินการเคลียร์ได้ทุกเวลา สมมติว่าได้ส่ง Call Request และก่อนที่จะได้รับ Call Accepted packet DTE ต้องการที่จะยกเลิกการเรียกด้วยเหตุบางอย่าง DTE สามารถกระทำได้โดยส่ง Clear Request packet ซึ่งจะได้รับการตอบรับจาก network

5.3.3.7 Clear collision

Clear collision เกิดขึ้นเมื่อ DTE และ DCE ส่ง Clear Request packet และ Clear Indication packet ในเวลาเดียวกันและได้กำหนด logical channel เดียวกันภายใต้สถานะการณ์อย่างนี้ DTE จะพิจารณาว่าการเคลียร์ได้กระทำเสร็จแล้ว DCE จะไม่ถือว่าต้องได้รับ DTE Clear Confirmation packet และจะไม่ส่ง DCE Clear Confirmation ไป logical channel ตอนนี้อยู่ในภาวะ READY

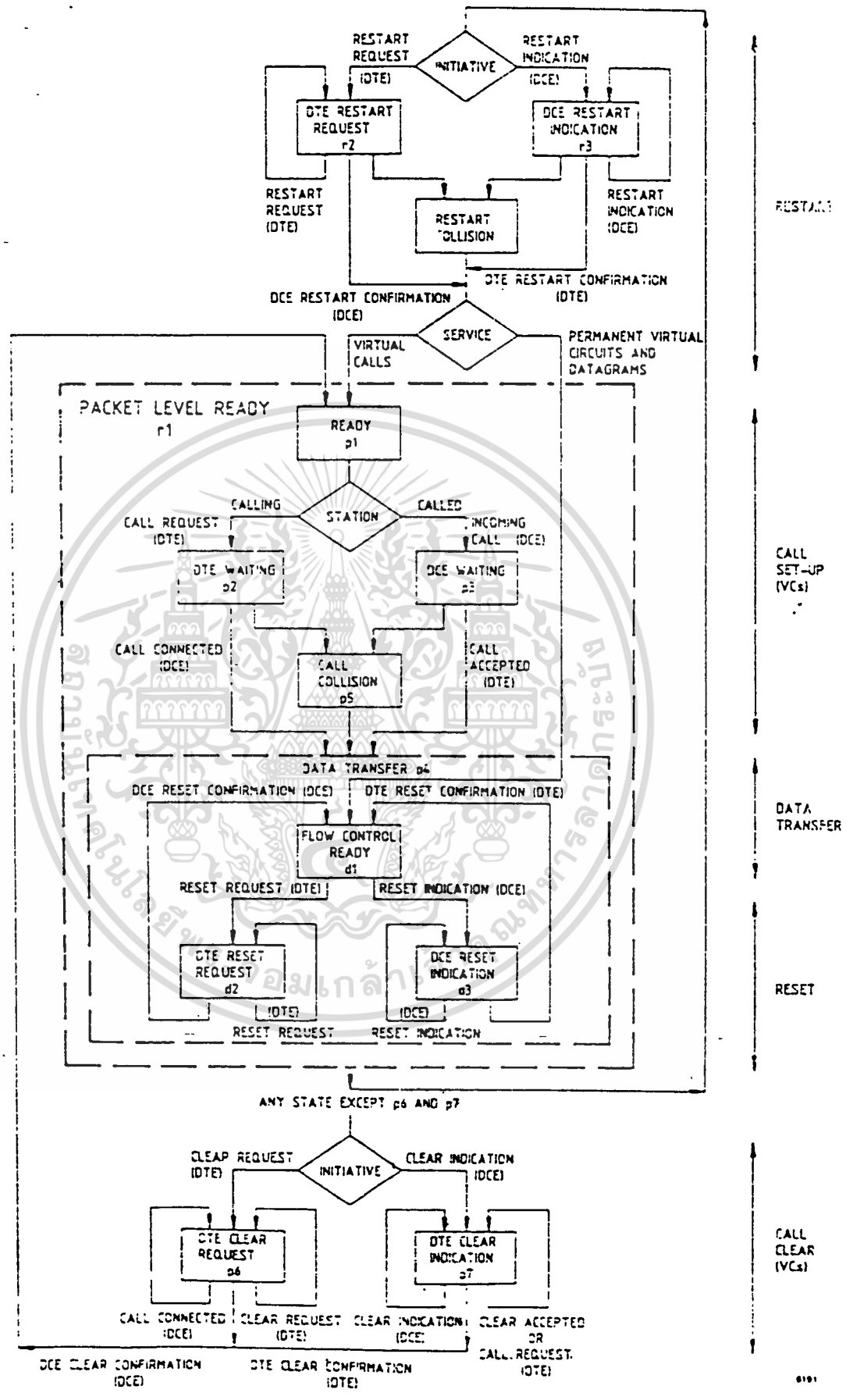
**5.3.4 State diagram**

ในตอนแรกๆ ของหัวข้อนี้ได้อ้างถึงภาวะต่างๆที่ได้เกิดขึ้นใน DTE/DCE interface ที่ได้กล่าวถึงมาก่อนแล้ว สามารถบอกถึงภาวะในรายละเอียดต่างๆได้

Recommendation X.25 แสดงถึงความสัมพันธ์ระหว่างภาวะต่างๆ และการกระทำที่ตอบสนองสำหรับการเปลี่ยนจากภาวะหนึ่งไปสู่อีกภาวะหนึ่ง state diagram ได้แสดงให้เห็นดังรูป

ในการที่จะดูแผนภูมิให้เข้าใจนั้น ประการแรกจำเป็นต้องพิจารณาเครื่องหมายที่ใช้ในการอ่านแผนภูมินั้นจะพบว่าเขาได้กล่าวถึงขั้นตอนต่างๆทั้งหมดที่ได้ให้ไว้ก่อนในหัวข้อแนวทางปฏิบัติต่างๆ ดังนั้นจะพบว่า Restart procedure, Call set-up procedure, Reset procedure และ Call Clear procedure เพราะฉะนั้นแผนภูมิได้อธิบายถึงความถูกต้องตามขั้นตอนทั้งหมดที่ DTE/DCE interface

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**รูปที่ 20 State Diagram**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 5.3.5 Diagnostic field และ diagnostic packet

5.3.5.1 ทั่วไป

ในขณะที่มี error เกิดขึ้น ฟิลด์นี้ทำให้ DCE สามารถให้รายละเอียดสาเหตุของ error กับ DTE ได้ packet type นี้ประกอบด้วย diagnostic field จะขึ้นอยู่กับภาวะของการเชื่อมต่อ ถ้าไม่เหมาะสมที่จะใช้แพคเกทอย่างใดอย่างหนึ่งใน 3 อย่าง ดังที่ได้กล่าวข้างบนนั้นแล้วบาง network จะจัดหา diagnostic packet ให้ เพื่อใช้ส่ง diagnostic code

ถึงแม้ว่าจะมี error เกิดขึ้นในสภาพผิดปกติใดๆที่มากกว่าหนึ่ง การตรวจพบครั้งแรกเท่านั้นที่จะถูกรายงานให้ทราบ หรือจะพูดอีกอย่างหนึ่งว่ามี diagnostic code เดียวเท่านั้นที่จะบรรจุใน diagnostic field ได้ จงสังเกตุว่า DTE ไม่ต้องดำเนินการใดๆ ที่ได้กำหนดการกระทำที่เกี่ยวกับข้อมูลที่ได้รับนี้ ถึงแม้ว่ามันอาจจะช่วยแยกแยะเหตุการณ์ที่เกิดขึ้นใหม่ได้

5.3.5.2 การใช้แพคเกทที่มี diagnostic field

Error ที่เกิดขึ้นใน Restart Phase ปกติแล้วจะใช้ Restart Indication packet สำหรับส่ง diagnostic code

ปัญหาที่เกิดขึ้นใน Call set-up และ Clearing phase ซึ่งจะถูกรายงานใน Clearing Indication packet

ปัญหาที่เกิดขึ้นใน Data Transfer phase จะถูกรายงานใน Reset Indication packet

สำหรับกรณีพิเศษ เมื่อแพคเกทมีความยาวน้อยกว่า 2 ออคเทท หรือมี general formatidentifier ไม่ถูกต้อง หรือไม่ได้กำหนด logical channel ไว้ Diagnostic packet จะถูกนำมาใช้ (ถ้ามีการสนับสนุนจาก network) Diagnostic packet ใช้อย่างเดียวกันกับ time-out error บางอย่าง ซึ่งมีดังต่อไปนี้

- DCE ส่ง Restart Indication และไม่ได้รับ Restart Confirmation หรือ Request จาก DTE ภายในน 60 วินาที
- DCE ส่ง Reset Indication และไม่ได้รับ Reset Confirmation หรือ Request จาก DTE ภายใน 60 วินาที
- DCE ส่ง Clear Indication และไม่ได้รับ Clear Confirmation หรือ Request จาก DTE ภายใน 60 วินาที

X.25 ได้กำหนดรายการต่างๆ ที่ได้ขึ้นบัญชีไว้เป็นจำนวนมาก ทั้งเหตุการณ์ปกติและ erroneous ที่อาจเกิดขึ้นได้ทุกภาวะที่เกี่ยวกับ DTE/DCE interface ที่ได้กำหนดไว้แล้ว

Table C-1/X.25 Special Cases

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table C-2/X.25 Restart Phase

Table C-3/X.25 Call set-up และ Clearing Phase

Table C-4/X.25 Data Transfer Phase

Table D-1/X.25 Time-outs

ตารางรายการเหล่านี้ได้กล่าวถึง diagnostic code ที่ได้ประยุกต์ใช้สำหรับสภาพที่เกิด error ที่ได้กำหนดไว้ทั้งหมด

5.3.5.3 Diagnostic field coding

Diagnostic field เป็นออคเททเดี่ยวที่มีรหัสไบนารี่ที่บอกให้ทราบถึง error ที่ตรวจพบ รหัสที่แบ่งออกเป็นกลุ่มเป็นประเภทต่างๆ มีดังนี้

| | |
|---|---|
| ไม่มีข้อมูลเพิ่มเติม | - codes 0-15 |
| Packet Type Invalid | - codes 16-31 |
| Packet Not Allowed | - codes 32-47 |
| Timer expired | - codes 48-63 |
| Call set-up problem | - codes 64-79 |
| Unassigned | - codes 80-95 |
| Unassigned | - codes 96-111 |
| Unassigned | - codes 112-127 |
| Reserved for network | - codes 128-225 |
| Specific diagnostic info. | |

แต่ละประเภทไม่ได้ใช้รหัสที่มีอยู่ทั้งหมด

ตารางรายการ แสดงถึงรหัสที่ได้กำหนดไว้ในปัจจุบัน และขั้นตอนและภาวะซึ่งรหัสได้สร้างขึ้น ข้อสังเกตุ รหัสตัวแรกในแต่ละประเภทเป็นแบบทั่วไป นั่นคือมันแทนทุกประเภท รหัสทั่วๆไปอาจใช้แทนรหัสที่ได้กำหนดไว้ได้มากขึ้นเป็นการจำกัด diagnostic code ให้มีจำนวนรหัสน้อยลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 5.4 Permanent virtual circuit service

ดังที่ได้กล่าวมาแต่แรกในหัวข้อที่ 2 แล้วว่า permanent virtual circuit ไม่จำเป็นต้องมีขั้นตอนของ Call set-up และ Clearing

ตารางที่ ได้กล่าวถึงแพคเกทสำหรับ set-up และ Clearing ซึ่งไม่เป็นประโยชน์ต่อ PVCs แพคเกทที่ยังค้างอยู่ที่จะใช้อธิบายสำหรับ virtual call services เพราะฉะนั้น หมายความว่า state p1 ถึง p7 ไม่ได้นำมาใช้กับ PVCs และเช่นเดียวกันกับ diagnostic codes ทั้งหมดที่เกี่ยวกับภาวะเหล่านี้ ไม่ได้นำมาใช้ใน PVC procedures

| DIAGNOSTICS | CODE (DEC.) | SPEC. CASES (TAB.C-1) | RESTART PHASE (TAB.C-2) | SET-UP, CLEARING PHASES (TAB.C-3) | DATA TRANSF PHASE (TAB.C-4) |
|---|---|---|---|---|---|
| NO ADDITIONAL INFORMAITON | 0 | ----- G E | N E R I C | ---------- | --------- |
| Invalid P(S) | 1 | | | | d1 |
| Invalid P(R) | 2 | | | | d1 |
| PACKET TYPE INVALID | 16 | ----- G E | N E R I C | ---------- | --------- |
| For state r1 | 17 | | r1 | | |
| " " r2 | 18 | | r2 | | |
| " " r3 | 19 | | - | | |
| " " p1 | 20 | | | p1 | |
| " " p2 | 21 | | | p2 | |
| " " p3 | 22 | | | p3 | |
| " " p4 | 23 | | | p4 | |
| " " p5 | 24 | | | p5 | |
| " " p6 | 25 | | | p6 | |
| " " p7 | 26 | | | - | |
| " " d1 | 27 | | | | d1 |
| " " d2 | 28 | | | | d2 |
| " " d3 | 29 | | | | - |
| PACKET NOT ALLOWED | 32 | ------G E | N E R I C | ---------- | --------- |
| Unidentifiable packet | 33 | | r2 | p1,2,3,5,6 | |
| Call on one way logical channel | 34 | | | p3 | |
| Invalid packet type on a PVC | 35 | | | | d1,d2 |
| Packet on unassigned logical channel | 36 | ANY | | | |
| Reject not subscribed to | 37 | | | | d1,d2 |
| Packet too short | 38 | ANY | r2 | p1,2,3,5,6 | |
| Packet too long | 39 | | r1,r3 | p1,5,7 | d1,d3 |
| Invalid general format id. | 40 | ANY | | | |
| Restart with nonzero on bits 1-4, 9-16 | 41 | | r2 | p1,2,3,5,6 | d1,2 |
| Packet type not compatible with facility | 42 | | | p3 | |
| Unauthorized interrupt confirmation | 43 | | | | d1 |
| Unauthorized interrupt | 44 | | | | d1 |
| TIMER EXPIRED | 48 | ------G E | N E R I C | ---------- | --------- |
| For incoming call | 49 | | | | |
| For clear indication | 50 | | | | |
| For reset indication | 51 | | | | |
| For restart indication | 52 | | | | |
| CALL SET-UP PROBLEM | 64 | | | | |
| Facility code not allowed | 65 | | | p3 | |
| Facility parameter not allowed | 66 | | | p3 | |
| Invalid called address | 67 | | | p3 | |
| Invalid calling address | 68 | | | p3 | |

**ตารางที่ 11 Packet types and their use in various services**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น และห้ามมิให้ดัดแปลงเนื้อหา รวมถึงห้ามลบข้อความนี้ของเอกสารทุกครั้งที่มีการนำไปใช้

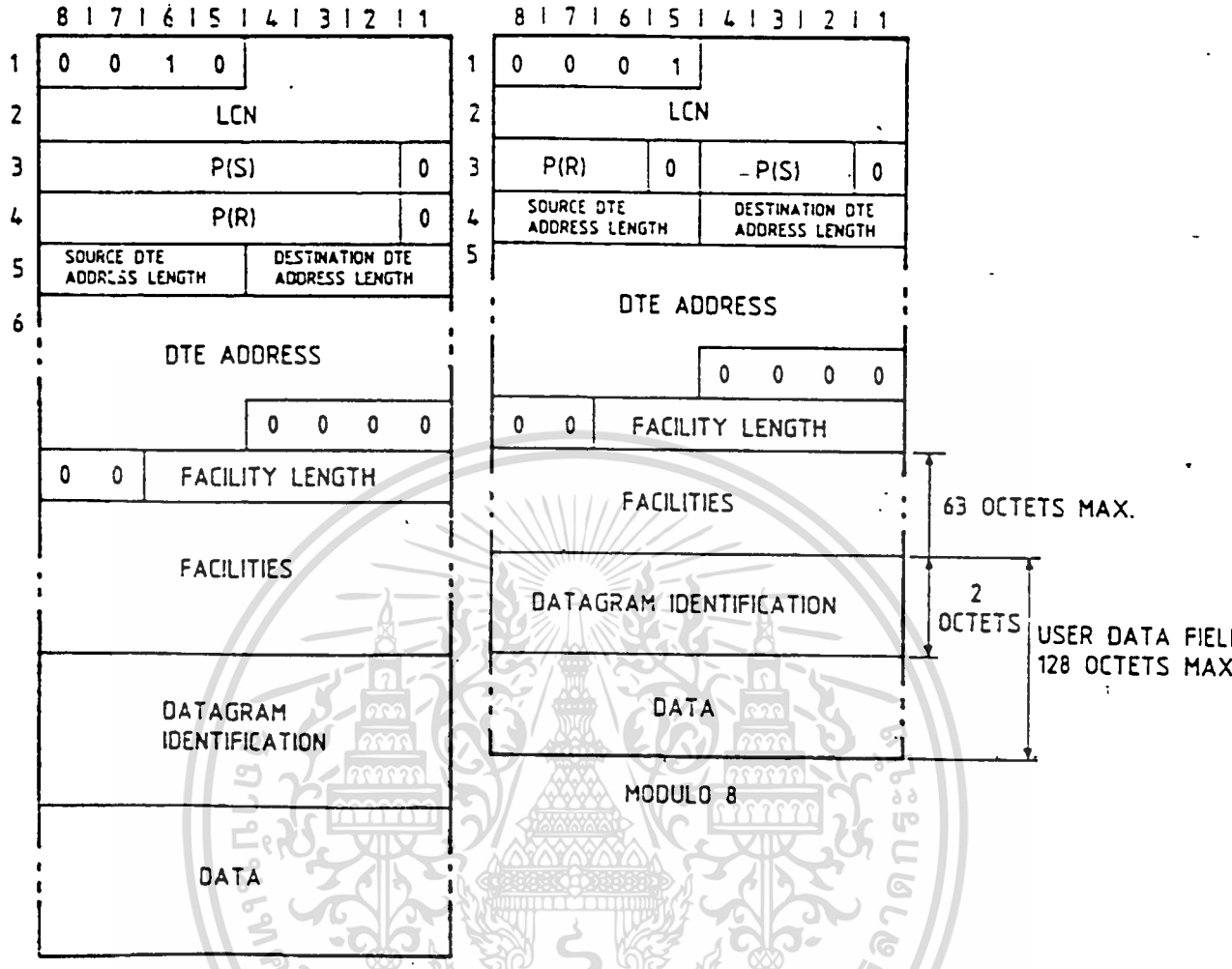
| PACKET TYPE | | SERVICE | | |
|---|---|---|---|---|
| from DCE to DTE | from DTE to DCE | VC | PVC | DG |
| **Call set-up and clearing** | | | | |
| INCOMING CALL | CALL REQUEST | X | | |
| CALL CONNECTED | CALL ACCEPTED | X | | |
| CLEAR INDICATION | CLEAR REQUEST | X | | |
| DCE CLEAR CONFIRMATION | DTE CLEAR CONFIRMATION | X | | |
| **Data and interrupt** | | | | |
| DCE DATA | DTE DATA | X | X | |
| DCE INTERRUPT | DTE INTERRUPT | X | X | |
| DCE INTERRUPT CONFIRMATION | DTE INTERRUPT CONFIRMATION | X | X | |
| **Datagram** | | | | |
| DCE DATAGRAM | DTE DATAGRAM | | | X |
| DATGRAM SERVICE SIGNAL | | | | X |
| **Flow control and reset** | | | | |
| DCE RR | DTE RR | X | X | X |
| DCE RNR | DTE RNR | X | X | X |
| | DTE REJ | X | X | X |
| RESET INDICATION | RESET REQUEST | X | X | X |
| DCE RESET CONFIRMATION | DTE RESET CONFIRMATION | X | X | X |
| **Restart** | | | | |
| RESTART INDICATION | RESTART REQUEST | X | X | X |
| DCE RESTART CONFIRMATION | DTE RESTART CONFIRMATION | X | X | X |
| **Diagnostic** | | | | |
| DIAGNOSTIC | | X | X | X |

04938A

VC VIRTUAL CALL
PVC PERMANENT VIRTUAL CIRCUIT
DG DATAGRAM

**ตารางที่ 12 Diagnostic Code and DTE/DCE Interface States**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 21 DTE and DCE DATAGRAME

## 5.5 Datagram service

### 5.5.1 บทนำ

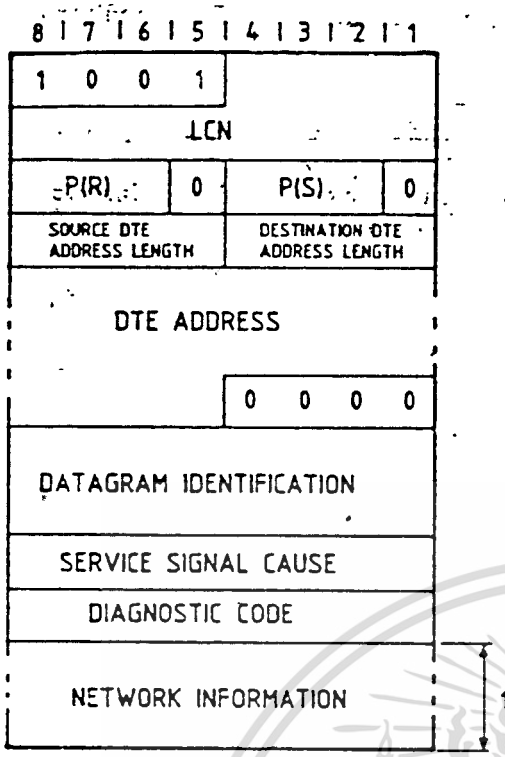
ตามที่ได้กล่าวแล้วในหัวข้อที่ 2 ไม่ต้องมี call set-up และ clearing สำหรับ datagram service Datagram เป็นแพคเกทที่ดำเนินการเป็นอิสระ และ network ไม่จำเป็นต้องมีการส่งต่อไปยัง destination DTE ในลำดับเดียวกันกับที่ได้รับจาก source DTE

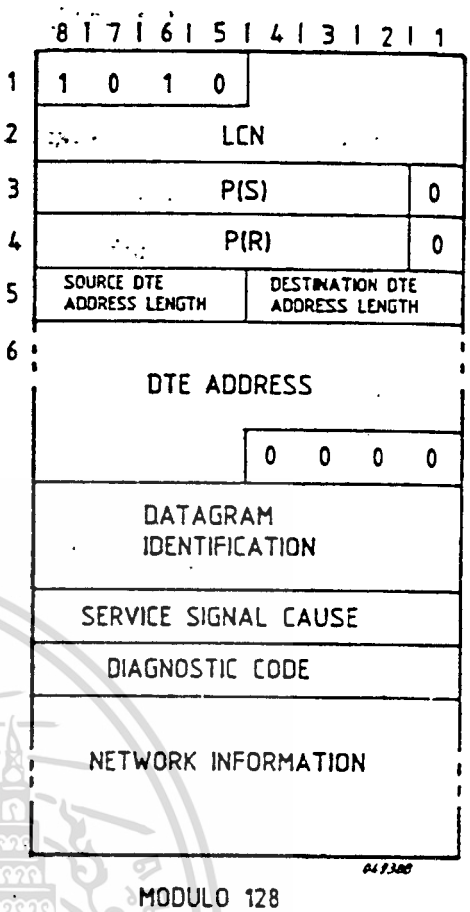
ขั้นตอนต่างๆ ที่ได้กล่าวไปแล้วได้นำมาประยุกต์ใช้กับ datagram logical channel แต่ละแชนแนลเป็นของตัวเองที่ DTE/DCE interface แพคเกทต่างๆ ผ่านไปตรงๆ เหมือนเดิมตลอด network ระหว่าง DTE ทั้งสอง

### 5.5.2 DTE และ DCE datagram format

Datagram ประกอบ field ของ Call และ Data packets ที่ได้กล่าวสำหรับ Virtual circuit service และจำเป็นที่จะกล่าวเพิ่มเติมอีกเล็กน้อยในที่นี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




| SERVICE SIGNAL CAUSE | | HEX.DEC. CODE |
|---|---|---|
| **DATAGRAM SERVICE SIGNAL - SPECIFIC** | | |
| **Datagram Rejected** | | |
| LOCAL PROCEDURE ERROR | | 13 |
| INVALID FACILITY REQUEST | | 03 |
| ACCESS BARRED | | 0B |
| NOT OBTAINABLE | | 0D |
| INCOMPATIBLE DESTINATION | | 21 |
| REVERSE CHARGING ACCEPTANCE NOT SUBSCRIBED | | 19 |
| **Datagram Non-Delivery Indication** | 1) | |
| NETWORK CONGESTION | | 05 |
| OUT OF ORDER | | 09 |
| NUMBER BUSY (DESTINATION QUEUE FULL) | | 01 |
| REMOTE PROCEDURE ERROR | | 11 |
| **Datagram Delivery Confirmation** | 2) | |
| DELIVERY CONFIRMATION | | 31 |
| **DATAGRAM SERVICE SIGNAL - GENERAL** | | |
| LOCAL DCE QUEUE OVERFLOW | 3) | 7F |
| NETWORK CONGESTION | | 47 |
| NETWORK OPERATIONAL | | 4F |

1) ISSUED ONLY WHEN THE NON-DELIVERY INDICATION FACILITY HAS BEEN REQUESTED
2) ISSUED ONLY WHEN THE DELIVERY CONFIRMATION FACILITY HAS BEEN REQUESTED
3) FOR FURTHER STUDY

**รูปที่ 22 DATAGRAME SERVECE SIGNAL**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Data field บรรจุได้สูงสุดถึง 128 ออคเทท ไม่มีการกำหนดให้เลือกใช้ 2 ออคเททแรก เรียกว่า Datagram Identification โดยยอมให้ DTE มีหมายเลขของ datagram โดยเฉพาะสำหรับกำหนดวัตถุประสงค์ให้สอดคล้องกับ algorithm ทั้งหมด network จะไม่ดำเนินการเกี่ยวกับการกำหนดสิ่งนี้ แต่ตอบด้วย Datagram Service Signal packet เมื่อต้องการ

**5.5.3 Datagram service signals**

Node ซึ่งมี DTE ต่ออยู่จะสร้าง Datagram Service packets เมื่อต้องการและส่งกลับไปให้กับ DTE

service signals อยู่ 2 ประเภท คือ specific และ general types :

- Specific type อ้างถึง specific Datagram packet และปัญหาหรือภาวะอื่นๆที่เกี่ยวข้องกับมัน
- General type อ้างถึงภาวะที่เกี่ยวข้องของข้อมูลทั้งหมด และไม่กำหนดแพคเกทใดๆ

Specific signals แบ่งออกได้เป็น 3 ประเภท

1. Datagram rejected - datagram ที่ถูกขจัดออกไปโดย network มันจำเป็นที่มีการทดสอบให้ถูกต้องก่อนที่จะดำเนินการ เหตุผลสำหรับการ rejected ได้ให้ไว้ใน diagnosis

2. Datagram non-delivery indication. - datagram ที่ถูกขจัดออกไปแต่ต่อมาการทดสอบปรากฏผลว่าถูกต้อง เหตุผลสำหรับ non-delivery ที่ได้ให้ไว้ โดยทั่วไปเป็นภาวะชั่วคราว

ข้อสังเกตุ : การบอกให้ทราบนี้ได้กำหนดขึ้นถ้า DTE subscriber มี non-delivery indication facility

3. Datagram delivery confirmation - datagram ที่มีการยอมรับโดย destination DTE

ข้อสังเกตุ : การบอกให้ทราบนี้ได้กำหนดขึ้นถ้า DTE subscribers มี delivery confirmation facility เท่านั้น

**5.5.4 Flow control**

Datagrams และ Datagram Service Signal packets จะถูกรวมกันที่ logical channels ได้มีการยกเว้นที่เป็นสมาชิกให้เลือกใช้ได้เป็นพิเศษซึ่ง service signal จะเคลื่อนที่ใน logical channel ที่แยกไว้ต่างหาก

เมื่อรวมกันแล้วทั้งสองอย่างนี้จะอยู่ภายใต้การควบคุมของ flow control mechanism อันเดียวกันในแต่ละ logical channel Packet numbering และ window mechanism จะอยู่ภายใต้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การดำเนินการของเครื่องรับ มีวิธีการอย่างเดียวกันกับที่ได้กล่าวในเรื่อง virtual circuit logical channels

แพคเกทแต่ละ logical channel จะเรียงกันตามลำดับ service signal packet ที่มีความสวำคัญสูงกว่า datagrams จะอยู่ในตำแหน่งหัวแถว

RR และ RNR packets ใช้สำหรับควบคุมการเคลื่อนที่ของข้อมูลดังที่ได้กล่าวมาแล้วในเรื่อง virtual circuit oreration

## 5.6 Operation user facilities

### 5.6.1 บทนำ

ใน X.25 มีจำนวนของ option facilities ค่อนข้างมากที่มีให้กับผู้ใช้ปลายทาง ผู้ใช้จะต้องบอกเป็นสมาชิกกับ facilities เหล่านี้ ทางด้านฝ่ายการจัดการอาจจะคิดค่าธรรมเนียมสำหรับการเป็นสมาชิกแต่ละอย่างไป

ในบางกรณีเพื่อที่จะใช้ facilities บางชนิด DTEs และ DCEs ต้องมีข้อมูลของ facility รวมอยู่ในแพคเกทที่ส่งออกไปด้วย ข้อมูลนี้ถูกส่งไปใน facility field ของแพคเกทที่เกี่ยวข้อง

### 5.6.2 Facility field

Facility field จะมีเมื่อ DTE ใช้ option user facility ที่ต้องการแจ้งให้ทราบบางอย่างใน Call Request ,Call Accepted, Call Connected, Clear Request, Clear Indication, DTE Datagram หรือ DCE Datagram packet เท่านั้น

Facility field ประกอบด้วย "facility elements" หนึ่งหรือมากกว่า ออกเททแรกของ facility element จะบรรจุ "facility code"

Facilities field บิทที่ 7 และ 8 กำหนด facility paremeter field ของ 1,2,3 หรือจำนวนของออคเททอาจแปรผันได้

สำหรับประเภท D ออคเททที่ 2 ของ facility element จะบอกถึงความยาวของ parameter field ความยาวนี้บอกด้วยรหัสไบนารี่

แต่ละประเภทอาจจะมี facility code ได้ถึง 64 ยกเว้นประเภท D ซึ่งมีเพียง 63 เท่านั้น (facility code 1111 1111 ใช้สำหรับการเพิ่มออกไปอีก)

รหัสของ parameter field ขึ้นอยู่กับ facility ที่ต้องการใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 5.6.3 Optional packet สำหรับ Virtual Circuit และ Datagram Services

5.6.3.1 Extended packet Sequence Numbering

Potional นี้สมัครเป็นสมาชิกสำหรับช่วงระยะเวลาที่กำหนดและนำมาใช้ได้กับทุก logical channel ที่ DTE/DCE interface

Facility นี้ซึ่งได้แสดงใน packet formats ที่ได้แสดงในตอนต้นๆ ของบทนี้ที่ยอมให้ผู้ใช้ใช้ modulo 128 numbering สำหรับ P(S) และ P(R) ได้ ถ้า option นี้ไม่ใช้ numbering นี้คือ modulo 8

5.6.3.2 Non-standard default window sizes

ถ้า option นี้สมัครเป็นสมาชิกสำหรับช่วงระยะที่กำหนด ผู้ใช้สามารถเลือก default window size จากรายการของ Window sizes ที่สนับสนุนโดยฝ่ายการจัดการ ถ้าไม่ใช้ facility นี้ default window size คือ 2

ค่าอื่นๆ ที่ต่างจาก default value อาจจะมีการตกลงสำหรับ virtual call โดยวิธีการของ flow control parameter ที่จะได้มีการตกลงเกี่ยวกับ facility ซึ่งจะกล่าวต่อไป

ค่าอื่นๆ ที่ต่างจาก default window size อาจจะมีการตกลงกันสำหรับช่วงระยะเวลาที่กำหนดสำหรับแต่ละ permanent virtual circuit และ แต่ละ datagram logical channel

ถ้าไม่มีอื่นที่จะเลือกใช้ สามารถเลือก default value มาใช้ได้

บางทีทางฝ่ายการจัดการอาจต้องการ window ที่มีขนาดเดียวกันทั้งสองทิศทาง

5.6.3.3 Default throughput classes assignment

Facility นี้สามารถบอกเป็นสมาชิกสำหรับช่วงระยะเวลาที่กำหนดได้ จะยอมให้สมาชิกเลือก default throughput class จากรายการของประเภทที่สนับสนุนโดยฝ่ายการจัดการ บางทีฝ่ายการจัดการอาจต้องการที่เกี่ยวกับการรับส่งข้อมูลที่มีประเภทเดียวกันทั้งสองทิศทาง ถ้าไม่ใช้ facility นี้ default throughtput class จะตกลงกับ throughtput class ของ DTE แต่ต้องไม่เกิน throughtput class สูงสุดที่สนับสนุนโดย network

ค่าอื่นๆ ที่ต่างจาก default value อาจตกลงกันสำหรับ virtual call โดยวิธีการที่เกี่ยวกับ throughtput class การตกลงที่เกี่ยวกับ facility นี้จะได้กล่าวต่อไป

ค่าอื่นๆ อาจจะมีการตกลงกันสำหรับช่วงระยะเวลาที่กำหนดสำหรับแต่ละ PVC และแต่ละ datagram logical channel

5.6.3.4 Packet retransmission - the DTE reject packet (DTE REJ)

ผู้ใช้สามารถเลือกใช้ facility นี้ร่วมกับ logical channel ได้ทั้งหมดที่ DTE/DCE interface ซึ่งจะต้องมีการตกลงสำหรับช่วงระยะเวลาที่กำหนดให้

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ ห้ามมิให้ทำซ้ำ ดัดแปลง เผยแพร่ หรือกระทำการใดๆ โดยไม่ได้รับอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Facility นี้จะยอมให้ DTE ขอร้องให้ DCE ส่ง flow controlled packets (นั่นคือ Data packets ตามปกติ) หนึ่งหรือมากกว่ามาให้ใหม่ DTE ขอร้อง DCE โดยการส่ง DTE REJ packet แพคเกทนี้จะระบุ logical channel ที่เกี่ยวข้องและกำหนด P(R) value P(R) value อาจจะอ้างถึงแพคเกทที่ได้รับซึ่งยังไม่ได้มีการตอบรับและอาจจะเห็นได้ชัดเจนว่าไม่เท่ากันหรือมากกว่าสำหรับแพคเกทที่จะส่งต่อไป

DCE จะเริ่มการส่งของ Data packets กลับไปให้ใหม่ เริ่มด้วย P(S) เท่ากับ P(R) จาก DTE REJ packet

เมื่อได้รับ DTE REJ packet มากกว่าหนึ่งก่อนที่จะเริ่มต้นส่งตามคำขอครั้งแรก DCE จะพิจารณาขบวนการของ DTE ที่ทำให้เกิด error และ DCE จะจัดการปรับ logical channel ใหม่

ถ้า DTE อยู่ในสภาวะที่ยังไม่พร้อมที่จะรับ มันจะแจ้งให้ทราบโดยการส่ง RNR packet ไปให้ถ้าจะเคลียร์ภาวะของ RNR มันจะส่ง DTE REJ packet แทน นั่นคือ ไม่จำเป็นต้องส่ง RR packet

5.6.3.5 Incoming calls barred

Facility นี้ต้องทำการตกลงกันสำหรับช่วงระยะเวลาที่ใช้และนำมาใช้ได้กับ Logical channel ทั้งหมดที่ DTE/DCE interface

ถ้าเป็นสมาชิก Virtual calls และ Datagrams จะไม่มีอยู่ที่ DTE DTE อาจเริ่มต้น VCs และ Datagrams ได้เลย

ข้อสังเกตุ : แชนแนลยังคงเป็นแบบดูเพล็กซ์สำหรับ flow control และหน้าที่ควบคุมอื่นๆ

5.6.3.6 Outgoing calls barred

Facility นี้ต้องมีการตกลงกันสำหรับช่วงระยะเวลาที่ใช้ และนำมาใช้ได้กับ logical channel ทั้งหมดที่ DTE/DCE interface

ถ้าเป็นสมาชิก DTE อาจจะรับ VCs และ Datagrams ได้อย่างเดียวเท่านั้นแต่ไม่สามารถเริ่มต้นจากอย่างใดอย่างหนึ่ง

5.6.3.7 One-way logical channel outgoing

Facility นี้ต้องมีการตกลงกันสำหรับช่วงระยะเวลาที่ใช้ และมีวิธีการอย่างเดียวกันกับ incoming calls barred ยกเว้นสำหรับการใช้กับลอจิตอลแชลแนลเดียวเท่านั้น

5.6.3.8 One-way logical channel incoming

Facility นี้ต้องมีการตกลงกันสำหรับช่วงระยะเวลาที่ใช้ และมีวิธีการอย่างเดียวกันกับ outgoing calls barred สำหรับการใช้กับ logicall channel นั้นให้ได้เพียงแชลแนลเดียวเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.6.3.9 Closed user group (CUG)

Facility นี้ต้องมีการตกลงกำหนดช่วงระยะเวลาการใช้สำหรับ VCs และ datagrams DTE อาจจะขึ้นอยู่กับ closed user group กลุ่มเดียวหรือมากกว่าได้ CUG ยอมให้ DTEs ที่ขึ้นอยู่กับกลุ่มนี้สามารถติดต่อสื่อสารระหว่างกันได้ แต่ป้องกันไม่ให้ติดต่อสื่อสารกับ DTEs กลุ่มอื่นๆ

Calling DTE จะกำหนด CUG สำหรับ VC หรือ datagram โดยการใช้ facility parameters ใน Call Request หรือ DTE Datagram packet

ถ้าเป็นการแจ้งไปยัง called DTE โดยการใช้ facility parameters ใน Incoming Call หรือ DCE Datagram packets

รหัสของ facility field แสดงให้เห็นในแผนภาพหน้าตรงข้าม parameter field ประกอบด้วย BCD 2 หลัก และ CUG index index นี้แสดงถึง local node ซึ่ง CUG นั้นถูกเลือก จงสังเกตุว่า index ที่ DTE ต้นทางและปลายทางของ การเชื่อมโยงไม่จำเป็นต้องเหมือนกัน

เมื่อ DTE ที่ขึ้นอยู่กับ CUG เดียวเท่านั้นไม่จำเป็นต้องมี facility field รวมอยู่ใน packets

5.6.3.10 Closed user group with outgoing access

Facility นี้ก็เลือกใช้ได้เช่นเดียวกันโดยมีการกำหนดช่วงระยะเวลาการใช้ DTE ที่เกี่ยวข้องสามารถจะขึ้นกับ CUG ได้มากกว่าหนึ่งกลุ่มได้ โดยมี bothway access ระหว่าง DTEs และเริ่มต้นจาก VCs หรือ datagrams ไปยังส่วนที่แยกออกของ Network ได้เช่นเดียวกัน อย่างไรก็ตามมันจะไม่ยอมรับ incoming VCs หรือ datagrams

เมื่อทำการเรียกไปยัง CUG members ควรจะใช้ facilities field เมื่อเรียกเข้าไปยังส่วนที่แยกออกของ network ในกรณีไม่จำเป็นต้องใช้ facility field สำหรับ option นี้

5.6.3.11 Closed user group with incoming access

Facility นี้คล้ายกับกรณีก่อนๆนอกจากไม่อนุญาตให้มี outgoing access แต่จะยอมให้มี incoming access มาจากส่วนที่แยกออกของ network

อนึ่งผู้ใช้อาจจะไม่ใช้ facility field สำหรับ option นี้เมื่อเป็นการต่อเชื่อมกับส่วนที่แยกออกของ network

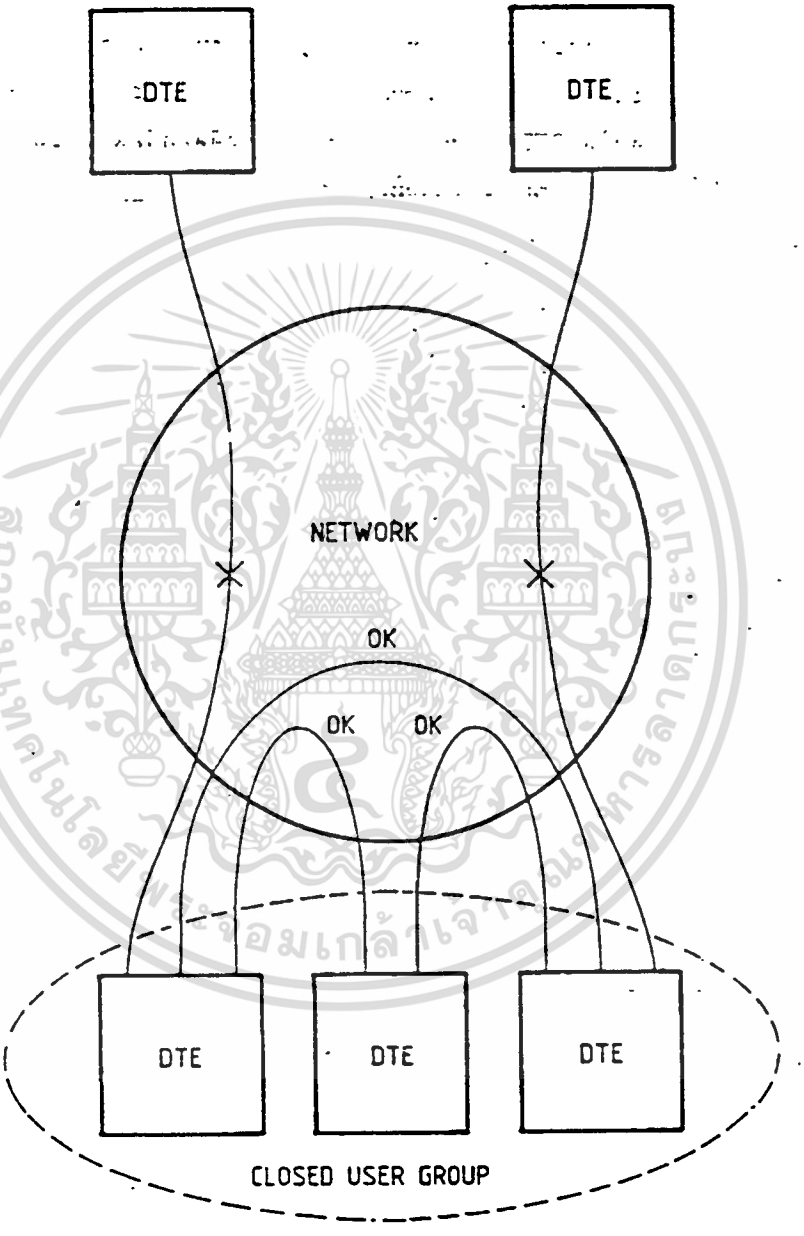
5.6.3.12 Incoming calls barred within a CUG

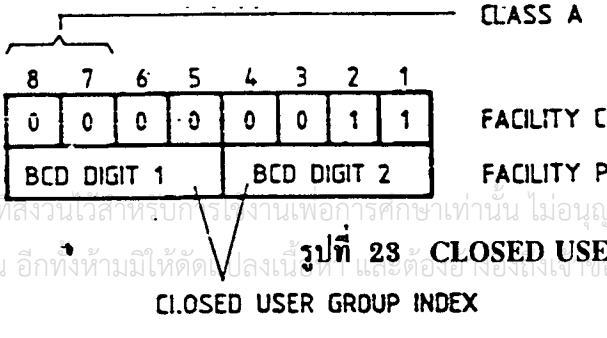
option นี้ก็เช่นเดียวกันที่ต้องมีการตกลงกำหนดเวลาการใช้ที่ยอมให้ DTE ใน CUG เริ่มต้น VCs หรือ datagrams กับ CUG members อื่นๆได้ แต่ไม่สามารถรับจาก CUG members อื่นๆได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.6.3.13 Outgoing calls barred within a CUG

Optical นี้จะกลับกันกับ option ก่อน มันจะยอมให้ DTE รับจาก CUG members ของมันเท่านั้น แต่ไม่ยอมให้ส่งออกไป




**รูปที่ 23 CLOSED USER GROUP**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.6.3.14 Bilateral CUG

Facility นี้ก็เลือกใช้ได้โดยทำการตกลงกำหนดระยะเวลาการใช้สำหรับ VCs และ datagrams DTE ที่มี facility นี้อาจจะขึ้นอยู่กับ bilateral CUG ได้มากกว่าหนึ่งกลุ่ม bilateral CUG จะยอมให้คู่ของ DTEs ซึ่งทั้งสองฝ่ายตกลงที่จะติดต่อกันและกันได้ แต่จะป้องกันการติดต่อกับ DTEs อื่นๆ ทั้งหมด

Calling DTE จะกำหนดชื่อ CUG สำหรับ VC หรือ datagram โดยการใช้ facility parameters ใน Call Request หรือ DTE Datagram packet

ที่จะแจ้งไปให้กับ Called DTE โดยใช้ facility parameters ใน Incoming Call หรือ DCE Datagrams packets

5.6.3.16 Reverse charging

Reverse charging เป็น facility ที่ผู้ใช้เลือกใช้ได้ DTE แสดงความต้องการขอให้โอนการคิดค่าบริการสำหรับ virtual call หรือ datagram

รหัสสำหรับ reverse charging ใช้อย่างเดียวกันกับ Call Request, Incoming Call, DTE Datagram และ DCE Datagram packets ซึ่งได้แสดงในแผนภาพหน้าตรงข้าม บิท 1 เฉพาะออกเทท facility parameter field ใช้สำหรับการเลือกใช้ reverse charging บิท 2 ถึง 6 ใช้สำหรับ facilities อื่นๆ facility ต้องการบิทที่มีภาวะ 1 เท่านั้น บิท 7 และ 8 ใช้สำหรับ fast select facility ซึ่งจะได้กล่าวต่อไป

5.6.3.17 Reverse charging Acceptance

ผู้ใช้ที่ใช้ facility นี้ เข้าเป็นสมาชิก เมื่อได้รับอนุญาตแล้ว DCE ส่งไปยัง DTE incoming calls หรือ datagrams ซึ่งต้องการ reverse charging facility แพคเกทนี้จะไม่ถูกส่งออกไปถ้าไม่มี facility นี้อยู่

5.6.3.18 RPOA selection

เมื่อผู้ใช้ต้องการใช้ facility นี้จะยอมให้ calling DTE กำหนดรายละเอียดโดยเฉพาะ Recognized Private Operating Agence (RPOA) ส่งผ่าน network โดยวิธีที่ซึ่ง Call หรือ datagram ที่เป็นเส้นทางระหว่างประเทศเมื่อที่ RPOA มากกว่าหนึ่ง transit network เหมาะที่จะใช้เป็น international gateway

รหัสที่เกี่ยวกับ facility field ได้แสดงไว้ในหน้าตรงข้าม PROA network number อยู่ในรูปของ 4 BCD digits

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 5.6.4 Optinal facilities ที่เหมาะจะใช้ร่วมกับ virtual circuit services

5.6.4.1 Non-standard default packet sizes

ถ้ามีการเลือกใช้ facility นี้ โดยการบอกเป็นสมาชิกสำหรับช่วงระยะเวลาหนึ่ง ผู้ใช้สามารถเลือกใช้ default packet size (เป็นการอ้างถึงความยาวสูงสุดของ user data field) ได้จากหัวข้อของ packet size ที่สนับสนุนโดยผ่ายการจัดการ ถ้าไม่ใช้ facility นี้ default packet size คือ 128 ออคเททของ user data

ค่าอื่นๆที่ต่างไปจาก default packet sizes อาจจะตกลงส่วนที่เกี่ยวกับ virtual call โดยหมายถึง flow control parameter ซึ่งจะได้มีการประชุมหารือกันต่อไป

ค่าอื่นๆที่ต่างไปจาก default packet sizes อาจจะมีการตกลงกันสำหรับช่วงระยะเวลาการใช้สำหรับ permanent virtual circuit แต่ละวงจรไป

บางครั้งทางฝ่ายการจัดการอาจกำหนด packet size ให้เป็นอย่างเดียวกันทั้งสองทิศทาง

5.6.4.2 Flow control parameter negotiation

ผู้ใช้ที่เลือกใช้ facility นี้ โดยกำหนดสำหรับช่วงระยะเวลาการใช้ โดย DTE สำหรับ virtual calls facility นี้ถ้าเป็นสมาชิกแล้วขอมให้มีการโยกย้ายกันได้ที่ per-call basis ของ flow control parameters parameter ได้อ้างถึงแพคเกทและ window sizes ที่ DTE/DCE interface สำหรับแต่ละทิศทางของ data transmission

ถ้าไม่มี facility นี้ การใช้ default packet และ window sizes ไม่ว่าจะเป็นแบบ option หรือ standard ก็ตาม ต้องบอกเป็นสมาชิกเสียก่อน

การแลกเปลี่ยนกันของพารามิเตอร์เหล่านี้ไม่ใช่เป็นแบบ end-to-end operation การเชื่อมโยงทั้งสองจะแยกออกจากกันไม่เกี่ยวกับการแลกเปลี่ยน ทางด้านผู้เรียก DTE กับ DCE แลกเปลี่ยนกันโดยใช้ facility field ของ Call Request และ Call Connected packets ที่ทางด้านผู้ถูกเรียก DTE กับ DCE แลกเปลี่ยนด้วย facility fields ของ Incoming Call และ Call Acceptance packets โดยทั่วไปถ้า window มีขนาดใหญ่กว่าหรือเท่ากับ 2 ตามที่กำหนด การตอบรับอาจจะเท่ากันหรือต่ำกว่ แต่จะไม่มากกว่าที่ได้กำหนด ถ้า window ที่มี 1 ที่ได้ตั้งไว้คือ 1 หรือ 2 ซึ่งยอมรับได้

สำหรับขนาดของแพคเกท การตอบรับอาจจะเท่ากับค่าที่ได้กำหนด หรือใกล้เคียงกับค่า 128 แต่จะไม่เลยจาก 128 ออคเทท ที่เป็นค่าที่กำหนดไว้

5.6.4.3 Throughput class negotiation

ประการแรกควรจะสังเกตว่าค่าของ default throughput คือจำนวนข้อมูลสูงสุดที่อาจเกี่ยวเนื่องกับ vitual call ที่ DTE/DCE interface

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Throughput class negotiation facility ซึ่งผู้ใช้อาจบอกเป็นสมาชิกสำหรับช่วงระยะเวลาหนึ่ง ที่ยอมให้มีการแลกเปลี่ยนที่ per-call basic การแลกเปลี่ยนสำหรับแต่ละทิศทางนั้นจะเป็นอิสระ ถึงแม้ว่าข้อมูลนั้นจะบรรจุในแพคเกทอย่างเดียวกัน มีการเลือกอีกวิธีการหนึ่งคือการกำหนดให้น้อยกว่า หรือให้มากที่สุดเท่ากับ default value ที่จุดเชื่อมโยงที่เกี่ยวข้องกัน

สรุปสั้นๆได้ว่า ประเภทจำนวนของข้อมูลในครั้งหนึ่งๆสำหรับการเรียกทุกครั้งคือจำนวนข้อมูลต่ำสุดที่จุดเชื่อมโยงด้านใดด้านหนึ่งไม่ว่าจะได้รับโดยการเลือกหรือโดย default

ข้อมูลที่จะใช้แลกเปลี่ยนถูกนำส่งใน Call Request, Incoming Call, Call Acceptance และ Call Connected packets ถ้า DTE ทั้งสองเป็นสมาชิกกับ facility นี้ ถ้ามี DTE เดียวเป็นสมาชิกจะมีเพียงแต่แพคเกทที่เกี่ยวข้องกันที่นำส่งใน negotiation fields ถ้า DTE ทั้งสองไม่ได้เป็นสมาชิก ดังนั้นจะไม่มีแพคเกทที่เกี่ยวข้องอยู่

5.6.4.4 Fast select facility

**Fast Select Packets**

ผู้อ่านอาจสังเกตได้ว่า setting up และ clearing virtual call มีขั้นตอนต่างๆมากทีเดียว ถ้าการส่งถ่ายข้อมูลที่มีผลต่อการแลกเปลี่ยนของแพคเกทที่มีองค์ประกอบเดียวในแต่ละทิศทางจริงๆ ขั้นตอนการดำเนินการของ virtual call ค่อนข้างคับคั่งมาก

Fast select facility นี้จะให้ประโยชน์กับผู้ใช้เพื่อแก้ปัญหาโดยแลกกับการแลกเปลี่ยนข้อมูลที่ตัดให้สั้นและต่อเนื่อง

Fast select packets ได้แสดงไว้ดังรูป การตรวจสอบอย่างละเอียดจะแสดงให้ทราบว่า call packets ต่างกันจาก call set-up packets ทั่วไปในรูปแบบส่วนใหญ่อย่างหนึ่งที่อยู่ใน data field Call Request และ Incoming Call packets สำหรับ fast select มี data fied สูงสุด 128 ออคเทท เปรียบเทียบให้เห็นถึงความแตกต่างกับ call set-up packets ทั่วไปที่มีเพียง 16 ออคเทท fast select Call Accepted และ Call Connected packets มี data field สูงสุด 128 ออคเททเช่นเดียวกัน ซึ่งตรงกันข้ามกับ call packet ทั่วไปไม่มี data field อยู่เลย

Clear packets สำหรับ fast select แสดงให้เห็นความแตกต่างอย่างมากกับภาวะทั่วๆไป และคล้ายกันมากกับภาวะทั่วๆไป ในรูปแบบปัจจัยที่สำคัญที่สุดถึงแม้ว่าความจริงแล้วแพคเกทเหล่านี้มี data field อยู่เช่นเดียวกัน ที่มีใช้อยู่ในขณะนี้

**Fast Select Procedure - Restricted Response**

ส่วนประกอบที่สำคัญ fast select procedure อาศัยหลักการของ call refusal ดังนั้นเมื่อ Incoming Call packet มาถึงที่ DTE มันจะปฏิเสธการเรียกโดยการตอบด้วย Clear Request

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

packet ความจริง facility field นี้เท่านั้นที่ยอมให้ called DTE ตอบด้วยวิธีนี้นั่นคือการตอบจะถูกจำกัด อย่างไรก็ตามแพคเกทเหล่านี้ประกอบด้วย data field ที่พอจะบรรจุข้อมูลของผู้ใช้เทียบได้กับ Data packet ทั่วไป ถึงแม้ว่าวิธีการนี้การเรียกจะถูกปฏิเสธ แต่การแลกเปลี่ยนข้อมูลก็ยังมีอยู่ นอกเหนือจากนั้น ลักษณะของการดำเนินการเป็นไปโดยอัตโนมัติ หมายความว่าเป็นการนำส่ง end-to-end confirmation

การแลกเปลี่ยนแพคเกทนี้เทียบได้กับ Call set-up, data transfer และ Clearance procedure และทำให้ประหยัดเวลาได้อย่างมากสำหรับการแลกเปลี่ยนข้อมูลสั้นๆ

**Fast select - Non - Restricted Response**

สามารถใช้ fast select packets สำหรับกำหนด virtual calls ได้ ได้ถูกเสนอขึ้นมาถ้า data transfer ประกอบด้วยแพคเกทจำนวนมาก facility field แสดงให้ทราบว่าการตอบไม่ได้ถูกจำกัดต่อ Clear Request packet และ DTE สามารถตอบด้วย Call Accepted packet ได้

ประเภทของ call set-up ต่างจากการเรียกทั่วไป เนื่องจาก data field ในแพคเกทมีความซับซ้อน

**Fast Select Not Requested**

Facilities field อาจจะบอกให้ทราบว่าไม่ต้องการ fast select facility ทำให้แพคเกทมีผลต่อ call set-up กลับไปสู่สภาวะปกติ และเป็นสิ่งจำเป็นสำหรับการติดต่อกับ DTEs ซึ่งไม่ได้เป็นสมาชิกของ fast select facility

**Facility Field Coding**

รหัสนี้ได้แสดงไว้ดังรูป จะสังเกตว่าบิทที่ 1 เป็น 1 พร้อมที่จะใช้สำหรับ reverse charging facility ซึ่งได้กล่าวในตอนต้นๆแล้ว

**Fast Select Acceptance Facility**

Facility นี้ต้องมีการกำหนดระยะเวลาการใช้ DCE จะให้ fast select Incoming Call packets กับ DTE ใน facility นี้เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 5.6.5 Optional facilities ที่เหมาะสมกับ datagram service

5.6.5.1 Abbreviated address

Option นี้ต้องทำสัญญากำหนดช่วงระยะเวลาการใช้ อนุญาตให้รหัสของ addresses ถูกตัดให้สั้นกว่าเดิมแทน ต้องมีการตกลงกันระหว่างฝ่ายการจัดการกับ DTE จะเริ่มต้นที่ 1-1 mapping ที่คัดเลือกเอาจากแอดเดรส

A1 to n mapping

Facility นี้จะช่วยทำให้เวลาของการใช้แอดเดรสเป็นประจำสั้นลง

5.6.5.2 Datagram queue length selection

การเลือกใช้ facility นี้ต้องทำสัญญากำหนดช่วงระยะเวลาการใช้สำหรับ logical datagram channel แต่ละวงจรไว้ facility นี้สามารถเลือกจำนวนของ Datagram และ Datagram Service Signal packets ที่จะเก็บเรียงกันตามลำดับโดย destination DCE เมื่ออัตราความเร็วของแพคเกทที่มาถึงที่ destination DCE จาก source DTE อื่นๆที่มีอัตราความเร็วเกินกว่าของที่จะนำส่งไปให้กับ destination DTE

5.6.5.3 Datagram service signal logical channel

Option นี้ต้องทำสัญญากำหนดช่วงระยะเวลาการใช้ มันจะเตรียมการแยก logical channel ให้กับ DTE ที่รับ Datagram Service Signals เท่านั้น ยอมให้ DTE แยก flow control ของ Datagram Service Signal packets ออกจาก Datagram packet ได้

5.6.5.4 Datagram non-delivery indication

Option นี้สามารถทำความตกลงสำหรับช่วงระยะเวลาการใช้สำหรับ Datagrams หรือที่เกี่ยวกับ per-datagram basis อย่างหนึ่งอย่างใดได้

การจัดหาสัญญาณสำหรับ non-delivery service ที่กำเนิดจาก network แต่ละครั้งที่ Datagram ไม่สามารถนำส่งไปยัง DTE ปลายทางได้

รหัสสำหรับ per-Datagram option ได้แสดงดังในรูป

5.6.5.5 Datagram delivery confirmation

Option นี้ก็เช่นเดียวกันสามารถที่จะทำความตกลงสำหรับช่วงระยะเวลาการใช้สำหรับ datagrams ทั้งหมดหรือที่เกี่ยวกับ per-datagram basis อย่างใดอย่างหนึ่งได้

Facility นี้จัดหาสัญญาณสำหรับ delivery confirmation service ที่กำเนิดโดย network หลังจากที่ datagram ได้มีการยอมรับจาก DTE ปลายทาง

สังเกตว่าถ้าเป็นการกระทำกับ datagrams ทั้งหมดมันจะทำให้เกิด service data มีจำนวนมากขึ้น

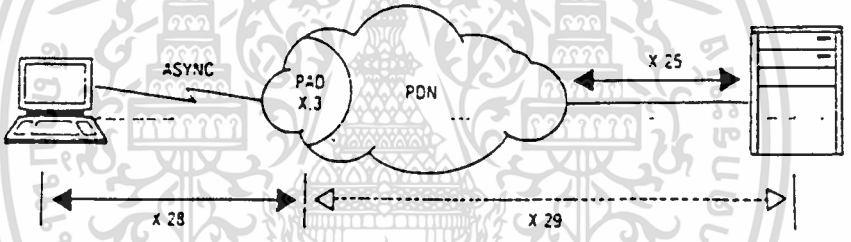
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 6

# มาตรฐานของ PAD , X.3 , X.28 และ X.29

## 6.1 บทนำ

ความสัมพันธ์ของ X.3, X.28 และ X.29 มีดังรูปที่ คือ




**รูปที่ 24 ความสัมพันธ์ของ X.3, X.28 และ X.29**

X.3 จะอธิบายถึง configuration parameter ของผู้ใช้แต่ละคน ซึ่งอยู่ใน PAD แต่ละ Synchronous port ที่อาจจะอยู่บน PAD เดียวกัน เช่น PAD ที่มี 10 พอร์ต ก็จะมี X.3 Paremeter ได้ทั้งหมด 16 แบบ

คำสั่งหรือ procedure ที่ใช้ระหว่างอุปกรณ์ที่ต่ออยู่กับ X.3 จะเรียกว่า X.28 ซึ่งแบ่งออกได้เป็น 2 ส่วนคือ PAD command และ PAD response โดย PAD command คือ configuration message ที่ส่งไปยัง PAD หรืออาจเป็นคำสั่ง Call Setup/Clearing ที่ส่งมาจากเทอร์มินอลและตอบสนองส่งกลับมาในทางตรงกันข้าม จะเรียกว่า PAD service signal

สำหรับ X.29 นั้น ใช้ในการ control และ modify X.3 PAD port Parameter บน network โดย remote packet-mode DTE เช่น X.25 host computer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 6.2 X.3 Parameter

ตาม X.3 recommendation นั้น จะมีทั้งหมด 22 parameter ดังแสดงในตารางที่ 13

| X.3 PAD PARAMETER | | |
|---|---|---|
| CATEGORY | NUMBER | FUNCTION |
| BREAK HANDLING | 7 | Action on Break |
| | 8 | Discard Output |
| DATA FORWARDING | 3 | Data Forwarding Character(s) |
| | 4 | Idle Timer |
| EDITING | 15 | Editing |
| | 16 | Character-Delete Character |
| | 17 | Buffer-Delete Character |
| | 18 | Buffer-Display Character |
| | 19 | Editing-Service Signals |
| FLOW CONTROL | 5 | Flow Control by PAD |
| | 12 | Flow Control by Terminal |
| LINE CHARACTERISTICS | 11 | Speed |
| | 21 | Parity Treatment |
| PAD RECALL | 1 | Escape From Data Transfer |
| DISPLAY FORMATTING | 9 | Carriage-Return Padding |
| | 10 | Line Folding |
| | 13 | Line-Feed Insertion |
| | 14 | Padding After Line Feed |
| | 22 | Page Wall |
| TERMINAL DISPLAY | 2 | Echo |
| | 6 | Control of PAD-Service Signals |
| | 22 | Echo Mask |

**ตารางที่ 13 X.3 PAD parameter**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Parameter 1 : Escape from Data Transfer**

Parameter 1 เป็นตัวกำหนดว่าในระหว่างการทำงานอยู่นั้นจะยอมให้มีการสลับ mode การทำงานมาเป็น PAD command หรือไม่

การส่งสัญญาณ Escape from Data Transfer นั้นอาจทำได้โดยการส่ง Data link Escape sequence เช่น Ctrl-P ไปยัง PAD เมื่อกลับเข้ามาใน PAD command mode แล้ว ผู้ใช้สามารถใช้ X.28 command ได้ และยังสามารถกลับไปยัง data transfer mode ได้อีกด้วย

ค่าของ Parameter ที่เป็นไปได้จะมีดังนี้

0 ไม่อนุญาตให้มีการ Escape from data transter

1 อนุญาตให้มีการ Escape from data transfer ได้

**Parameter 2 : Echo**

Parameter 2 เป็นตัวกำหนดว่า จะให้มีการแสดงตัวอักษรขึ้นบนหน้าจอเองโดย PAD หรือไม่ หรืออาจจะให้การแสดงตัวอักษรนี้ทำโดย remote DTE โดยใช้ X.29

ค่าของ Parameter ที่จะเป็นไปได้ มีดังนี้

0 ไม่ให้มีการ Echo

1 ให้มีการ Echo

**Parameter 3 : data forwarding Character(s)**

นอกจากเราจะให้มีการส่งแพคเกท เมื่อแพคเกทเต็มแล้วเราอาจส่งทันทีที่ได้รับอักษรที่บอกให้ทำการส่งก็ได้ โดยวิธีนี้ข้อมูลจะถูกเก็บอยู่ใน buffer จนกว่าจะมีอักษรที่ส่งให้ทำการส่งแพคเกทออกไป ดังนั้นอักษรที่เป็นตัวบอกให้มีการส่งแพคเกทนั้นจะเป็นอักษรตัวสุดท้ายในทุกๆ แพคเกท

อักษรนี้ เป็นตัวที่มีบทบาทสำคัญในการกำหนดว่า ก่อนที่จะส่งแพคเกทออกไปนั้น แต่ละแพคเกทมีข้อมูบอยู่เต็มหรือไม่มากน้อยเพียงใด ถ้าต้องการที่จะลดปริมาณข้อมูลที่วิ่งอยู่บน network เราควรจะให้มีจำนวนข้อมูลอยู่ในแต่ละแพคเกทมากที่สุด โดยทั่วๆไปอักษรที่ใช้เป็นตัวกำหนดในการส่งข้อมูลมักใช้เป็น carriage return (CR)

ค่าของ Parameter ที่จะเป็นไปได้ มีดังนี้

0 ไม่มีการใช้อักษรใดเป็นตัวกำหนดในการส่งข้อมูล

1 ใช้อักษรที่เป็นตัว alphanumeric (A-Z, a-z, 0-9) เป็นตัวกำหนด

2 ใช้อักษร carriage return (CR) เป็นตัวกำหนด

4 ใช้อักษร ESC, BEL, ENQ, ACK เป็นตัวกำหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8 ใช้อักษร DEL, CAN, DC2 เป็นตัวกำหนด

16 ใช้อักษร ETX,EOT เป็นตัวกำหนด

32 ใช้อักษร HT, LF, VT, FF เป็นตัวกำหนด

64 ใช้อักษรทุกๆตัวที่เป็น ASCII ที่นอกเหนือจากที่กำหนดไว้ข้างบน(ยกเว้น DEL) เป็นตัวกำหนด

ตัวอย่างการกำหนด Parameter 3 นี้ที่เป็นไปได้เช่น (2+4) = 6 หรือ (1+2+4+8+16 +32+64) = 127

**Parameter 4 : Idle Timer**

วิธีตรงกันข้ามกับการส่งแบบ data forward ที่กำหนดโดย Parameter 3 ก็คือวิธีการใช้ Idle Timer เป็นตัวกำหนด โดยในการส่งแพคเกทออกไปนั้นจะขึ้นอยู่กับเวลาที่กำหนดให้ว่าข้อมูลที่จะส่งแต่ละตัวจะห่างกันนานเท่าไหร่ ค่าของ Idle time จะอยู่ระหว่าง 0 ถึง 255

สำหรับในกรณีที่เป็น interactive นั้น การกำหนดให้ Idle time มีค่าต่างๆจะทำให้ได้ response ที่ดี แต่ในขณะเดียวกันก็เป็นการสิ้นเปลืองเพราะแพคเกทมีขนาดเล็ก

ค่าของ Parameter ที่จะเป็นไปได้มีดังนี้

0 ไม่ใช้ Idle time

1-255 Delay เป็น 20 ส่วนของวินาที (ค่า 20 = 1 วินาที)

**Parameter 5 : Flow Control by PAD**

Parameter นี้เป็นตัวกำหนดว่าจะให้การ flow นั้นถูกควบคุมโดย PAD หรือไม่ และจะใช้วิธีการควบคลุมแบบ X-ON/X-OFF หรือไม่ การควบคุมแบบนี้ไม่จำเป็นสำหรับการทำงานแบบ interactive เพราะความสามารถในการพิมพ์ของคนจะไม่มีทางล้น input butter ของ PAD แต่ในการทำงานที่ข้อมูลมีการส่งอย่างต่อเนื่องเช่นการ transfer file นั้น จะต้องการการควบคุมโดยวิธีนี้ เพื่อป้องกันการสูญหายของข้อมูล

ค่าของ Parameter ที่จะเป็นไปได้ มีดังนี้

0 ไม่ใช้ X-ON/X-OFF flow control

1 ใช้ X-ON (ASCII DC1) /X-OFF (ASCII DC3) flow control

**Parameter 6 : Control of PAD Service Signals**

PAD service Signal เป็น network status message ที่ส่งโดย PAD ไปยัง เทอร์มินอลต่างๆ เพื่อให้ข้อมูลที่จำเป็นสำหรับผู้ใช้ที่แต่ละ เทอร์มินอล ซึ่งเกี่ยวกับ status ของผู้ใช้ที่ทำการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรียกใช้ นอกจากนี้มันยังอาจใส่เข้าไปใน application ที่เป็น noninteractive เพื่อบอกความผิดพลาดที่เกิดขึ้นได้อีกด้วย

ค่าของ Parameter ที่จะเป็นไปได้ มีดังนี้

0 ไม่มีการส่ง service signal

1 มีการส่ง service signal

**Parameter 7 : Action on Break**

การกำหนด Parameter นี้จะเป็นการบอกถึงสิ่งที่ PAD จะต้องทำเมื่อได้รับสัญญาณ break โดยทั่วไปการที่ผู้ใช้ส่งสัญญาณ break นั้น เพื่อบอกให้ยกเลิกการทำงานของโปรแกรมนั้นๆ Parameter ที่ 7 นี้จะใช้เพื่อให้การทำเช่นนี้เร็วขึ้นหรือเพื่อไม่ให้มีการ break เกิดขึ้น

ค่าของ Parameter ที่จะเป็นไปได้ มีดังนี้

0 ไม่มีการทำอะไรเมื่อได้รับสัญญาณ break

1 PAD ส่ง X.25 Interrupt Packet

2 PAD ส่ง X.25 Reset Packet

4 PAD ส่ง Indication-of-Break message

8 ออกจาก data transfer mode

9 ไม่มีการแสดงผลออกทางเทอร์มินอล

ตัวอย่างการกำหนด Parameter 7 นี้ที่เป็นไปได้เช่น (1+4) = 5 หรือ (1+4+16) = 21

**Parameter 8 : Discard Output**

เมื่อ Parameter นี้ถูกใช้ระหว่างการเรียกใช้ ข้อมูลที่ถูก disassemble โดย PAD และส่งไปยังเทอร์มินอลนั้น จะถูก discard

Parameter 8 นี้จะใช้ร่วมกับ Parameter 7 กล่าวคือ ถ้า Parameter 7 ถูกกำหนดเป็นค่า 16 จะส่งผลให้ Parameter 8 เป็น 1 คือจะ discard output

ค่าของ Parameter ที่จะเป็นไปได้ มีดังนี้

1 ส่งข้อมูลปกติ

2 Discard output

**Parameter 9 : Carriage-Return Padding**

จำนวน padding character ที่จะส่งไปหลังจากอักษร Carriage Return (CR) จะกำหนดได้โดย Parameter ที่ 9 นี้ คุณสมบัตินี้มีไว้สำหรับ printer ที่ช้าๆ ที่ต้องใช้เวลาในการเลื่อนหัวอ่านเมื่อจะต้องขึ้นบรรทัดใหม่ padding character นั้นมักจะใช้เป็น ASCII NULL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าของ Parameter ที่จะเป็นไปได้ มีดังนี้

0 ไม่ต้องส่ง padding character

1-7 จำนวน padding character ที่ต้องส่งหลัง CR

**Parameter 10 : Line Folding**

Parameter 10 นี้ใช้สำหรับในกรณีที่จำนวนอักษรต่อ 1 บรรทัดของ host และเทอร์มินอล มีขนาดไม่เท่ากัน บาง host จะส่งข้อมูลโดยมีจำนวนตัวอักษรต่อบรรทัดเท่ากับจำนวนสูงสุดที่จะเป็นไปได้ (132 อักษรต่อบรรทัด) แต่ในขณะที่เทอร์มินอลอาจแสดงได้เพียง 80 อักษรต่อบรรทัด และ เทอร์มินอลนั้นอาจไม่มีระบบ wraparound ดังนั้น Parameter 10 นี้จะต้องถูกกำหนดให้เป็น 80 เพื่อให้ Carriage Return เข้าไปทุกๆ 80 ตัวอักษรที่ได้รับมา

ค่าของ Parameter ที่จะเป็นไปได้ มีดังนี้

0 ไม่ต้องมี line folding

1 จำนวนอักษรต่อบรรทัด

**Parameter 11 : Speed**

Parameter 11 นี้จะไม่สามารถกำหนดได้เองโดยผู้ใช้ จะขึ้นอยู่กับว่า network ของแต่ละ port นั้นๆสามารถรองรับได้ที่ความเร็วเท่าใด

ค่าของ Parameter ที่จะเป็นไปได้ มีดังนี้

| | | | |
|---|---|---|---|
| 0 | 110 bit/s | 1 | 134.5 bit/s |
| 2 | 300 bit/s | 3 | 1,200 bit/s |
| 4 | 600 bit/s | 5 | 75 bit/s |
| 6 | 150 bit/s | 7 | 1,800 bit/s |
| 8 | 200 bit/s | 9 | 100 bit/s |
| 10 | 50 bit/s | 11 | 75/1,200 bit/s (Videotex) |
| 12 | 240 bit/s | 13 | 4,800 bit/s |
| 14 | 9,600 bit/s | 15 | 19,200 bit/s |
| 16 | 48,000 bit/s | 17 | 56,000 bit/s |
| 18 | 64,000 bit/s | | |

**Parameter 12 : Flow Control by Terminal**

เป็น Parameter ที่ใช้กำหนด Flow control ในทิศทางตรงกันข้ามกับที่กำหนดโดย Parameter 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าของ Parameter ที่จะเป็นไปได้ มีดังนี้

0 ไม่ใช้ X-ON/X-OFF flow control

1 ใช้ X-ON (ASCII DC1) /X-OFF (ASCII DC3) flow control

**Parameter 13 : Line-Feed Insertion**

Parameter 13 ใช้ในการกำหนด line feed ที่รับและส่งกับเทอร์มินอลที่แตกต่างกันหลัง carriage-return

ค่าของ Parameter ที่จะเป็นไปได้ มีดังนี้

0 ไม่มีการเพิ่ม line-feed

1 เพิ่ม line-feed หลัง carriage-return ที่ส่งไปยังเทอร์มินอล

2 เพิ่ม line-feed หลัง carriage-return ที่รับมาจากเทอร์มินอล

4 เพิ่ม line-feed หลัง echo carriage-return ไปยังเทอร์มินอล

ตัวอย่างการกำหนด Parameter 13 นี้ที่เป็นไปได้เช่น (1+4) = 5 หรือ (1+2+4) = 7

**Parameter 14 : Padding after Line-feed**

Parameter 14 นี้ทำหน้าที่เหมือนกับ Parameter 9 แต่มีข้อแตกต่างกันที่ Parameter 14 นี้จะส่ง padding Character หลังจาก line-feed แทนที่จะส่ง carriage-return

ค่าของ Parameter ที่จะเป็นไปได้ มีดังนี้

0 ไม่ต้องส่ง padding character

1-7 จำนวน padding character ที่ต้องส่งหลัง LF

**Parameter 15 : Editing**

Parameter 15 นี้จะใช้ร่วมกับ Parameter 16, 17 และ 18 ในการ edit อักษรที่อยู่ใน buffer ของ PAD ก่อนที่จะส่งไป ดังนั้นถ้ากำหนดให้การส่งแบบ Idle time จะต้องกำหนดให้ Parameter 15 นี้เป็นแบบไม่มีการ edit

ค่าของ Parameter ที่จะเป็นไปได้ มีดังนี้

0 ไม่มีการ edit

1 มีการ edit

**Parameter 16 : Character-delete Character**

ASCII ที่กำหนดโดย Parameter 16 นี้จะเป็นตัวที่ใช้เป็น delete character ในการ edit โดยทั่วไปจะใช้ DEL (ASCII 127)

ค่าของ Parameter ที่จะเป็นไปได้ มีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

0 ไม่มีการใช้

1-127 ASCII ที่ใช้เป็น delete-character

**Parameter 17 : Buffer-delete Character**

อักษรที่กำหนดโดย Parameter 17 นี้จะเป็นอักษรที่ใช้บอกให้มีการลบข้อมูลที่อยู่ใน buffer ของ PAD ออก โดยทั่วไปจะใช้ Ctrl-X (ASCII 24)

ค่าของ Parameter ที่จะเป็นไปได้ มีดังนี้

0 ไม่มีการใช้

1-127 ASCII ที่ใช้เป็นอักษรสำหรับส่ง buffer delete

**Parmeter 18 : Buffer-display Character**

อักษรที่กำหนดโดย Parameter 18 นี้จะเป็นอักษรที่ใช้บอกให้มีการ redisplay ข้อมูลที่อยู่ใน buffer ของ PAD โดยทั่วไปจะใช้ Ctrl-R (ASCII 18)

ค่าของ Parameter ที่จะเป็นไปได้ มีดังนี้

0 ไม่มีการใช้

1-127 ASCII ที่ใช้เป็นอักษรสำหรับส่ง buffer display

**Parameter 19 : Edit Service Signals**

ในการ edit ของ hard-copy กับ vidio-display terminal นั้นจะแตกต่างกัน เนื่องจากในกรณีของ hard-copy นั้น เราไม่สามารถกลับไปลบส่วนที่พิมพ์ไปแล้วไม่ได้ ดังนั้นเมื่อต้องการลบนั้น เราจึงใช้วิธีส่ง "BS space BS" ไปเพื่อลบข้อมูล ในกรณีของ hard-copy นั้นเมื่อลบจนหมดบรรทัดใน buffer ของ PAD แล้ว PAD จะ response "XXX", CR, LF

ค่าของ Parameter ที่จะเป็นไปได้ มีดังนี้

0 ไม่มีการ edit ของ PAD service signal

1 Hard-copy terminal

2 video-display terminal

8 BS จะถูกส่งเป็น response สำหรับการลบ

32-126 ASCII ที่จะถูกส่งเป็น response สำหรับการลบ

**Parameter 20 : Echo Mask**

Parameter 20 นี้ใช้ในการกำหนดว่าอักษรตัวใดจะให้มีการ echo ตัวใดไม่ให้มีการ echo

ค่าของ Parameter ที่จะเป็นไปได้ มีดังนี้

0 ให้ echo ทุกตัวอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1 ไม่ให้ echo อักษร CR

2 ไม่ให้ echo อักษร LF

4 ไม่ให้ echo อักษร VT, HT, FF

8 ไม่ให้ echo อักษร BEL, BS

16 ไม่ให้ echo อักษร ESC, ENQ

32 ไม่ให้ echo อักษร ACK, NAK, STX, SOH, EOT, ETB, ETX

64 ไม่ให้ echo อักษรที่กำหนดเป็น edit-character ใน Parameter 16, 17, 18

128 ไม่ให้ echo อักษร DEL และอักษรอื่นที่เป็น Control-character นอกเหนือจากที่กำหนดไว้ด้านบน

**Parameter 21 : Parity Treatment**

Parameter 21 นี้เป็นตัวกำหนดว่าให้มีการตรวจสอบ parity หรือ การสร้าง parity ส่งไป หรืออาจทำทั้ง 2 อย่างควบคู่กันไป Parameter 21 นี้จะต้องกำหนดให้เป็น 0 ถ้ามีการส่งข้อมูลขนาด 8 bit ทั้งในกรณีของการส่งข้อมูลแบบ graphic หรือการส่งใน binary mode

ค่าของ Parameter ที่จะเป็นไปได้ มีดังนี้

0 ไม่มีการตรวจสอบ parity

1 มีการตรวจสอบ parity

2 มีการ generate parity

3 มีการตรวจสอบ และ generate parity

**Parameter 22 : Page Wait**

ในกรณีเทอร์มินอลที่เป็นแบบ scholling เราสามารถกำหนดโดย Parameter 22 นี้ได้ว่าให้หยุดทุกครั้งเมื่อจำนวน line-feed เท่ากับจำนวนบรรทัดของ full screen แล้ว

ค่าของ Parameter ที่จะเป็นไปได้ มีดังนี้

0 ไม่มีการหยุดรอแต่ละหน้า

1-255 จำนวน line-feed ที่ส่งมาโดย PAD ก่อนที่จะหยุดรอแต่ละหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

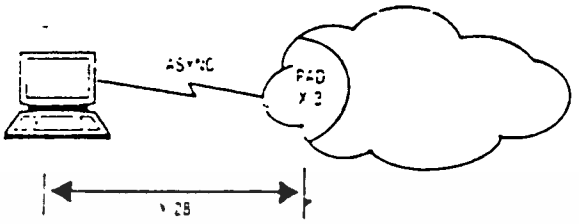
## 6.3 X.28 Command and response

หลังจากที่ผู้ใช้ได้เข้าสู่ PDN node X.28 Command จะอนุญาตให้ผู้ใช้ติดต่อโดยตรงเข้ากับ PAD และทำการ Call เข้ากับ Network ได้ ดังรูปที่ 25 ถ้าจำเป็น คำสั่งต่างๆเหล่านี้สามารถเปลี่ยนแปลงได้ที่ X.3 Parameter โดยใช้คำสั่งทั้ง 9 คำสั่งของ PAD ดังตารางที่ 14 โดยสามารถแบ่งออกได้ 3 ประเภท คือ การทำการ Call, อ่านและแปลง X.3 Parameter และตรวจสอบหรือ reset การ Call เมื่อผู้ใช้ส่ง Command ไปแล้ว PAD จะทำการตอบสนองด้วย PAD Service Signal ทั้ง 12 คำสั่งดังตารางที่ 15

| PAD Command signal format | Function | PAD Service Signal sent in response |
|---|---|---|
| STAT | To request status information regard- ing a virtual call connected to the DTE | FREE or ENGAGED |
| CLR | To clear a virtual call | CLF CONF or CLR ERR (in case of local procedure error) |
| PAR? | To request the current values of specified parameters | PAR (list of parameter references with their current values or INV) |
| SET? | To rewuest changing or setting of the current values of the specified parameters and to rewuest the current values of specified parameters | PAR (list of parameter references with their current values or INV) |
| PROF | To give PAD parameters a standard set of values | Acknowledgement |
| RESET | To reset the virtual call | Acknowledgement |
| INT | To transmit an interrupt packet | Acknowledgement |
| SET | To set or change parameter values | Acknowledgement |
| Selection PAD command signal | To set up a virtual call | Acknowledgement |

**ตารางที่ 14 PAD Command Signal**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

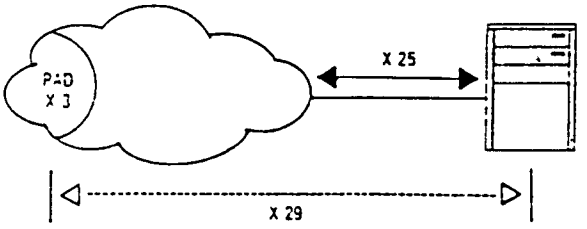



รูปที่ 25 X.3 และ X.28

## 6.4 X.29 and Remote PAD Configuration

นอกจากการใช้ X.28 ในการเปลี่ยนแปลง Configuration ของ PAD แล้วเรายังสามารถอนุญาตให้ Remote Host ทำการเปลี่ยนแปลง Config ของ PAD ได้ทันทีอัตโนมัติหลังจากมีการ Log On โดยการใช้ X.29 Message ระหว่าง PAD และ Remote DTE หรือ Host

X.29 Message จะส่งไปบน Network ใน User Data Field ของ X.25 Packet ดังรูปที่ 26 Packet นี้จะถูกสังเกตุได้ว่าเป็น X.29 Packet ได้เพราะจะมี Q bit ที่หัวของ Packet เป็น 1 ส่วน X.29 Code นั้นจะอยู่ใน 4 บิทล่างของออคเททแรกใน User Data Field




รูปที่ 26 X.25 และ X.29

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| Format of the PAD service signal | | Explanation |
|---|---|---|
| RESET | DTE 1,2,or 3 characters witch represent the<br>ERR decimal value of the diagostic code<br>NC | Indication that the remote DTE has reset the virtual call<br>Indication of a reset of a virtual call due to a local procedure error<br>Indication of a reset of a virtual call due to network congestion |
| CLR | | Indication of clearing |
| COM | | Indication of call connected |
| PAD identification PAD service signal | The characters to be sent are network dependent | |
| ERROR | ERR | Identification that a PAD command is in error |
| (Network dependent) | | Indication of incoming call |
| XXX | | Indication of line delete function completed |
| ENGAGED | | Response to status PAD command signal when a call is not established |
| FREE | | Response to status PAD command signal when a call has bee established |
| PAD | Decimal value of parameter: parameter value or INV | Pesponse to set-and-read or read PAD command signal |
| (Network dependent) | | Prompt PAD service signal |
| Format effect | | Acknowledgement PAD service signal |

**ตารางที่ 15 PAD Service Signals**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 7
# โปรแกรมด้านการสื่อสารข้อมูล

## 7.1 ความรู้ด้านโปรแกรมสื่อสาร

ในการศึกษาการทำงานของการรับส่งข้อมูลขนาด 1 ไบต์ออกทางพอร์ตอนุกรมเราต้องทำความเข้าใจในส่วนของฮาร์ดแวร์บ้างพอสมควรเพื่อช่วยในเรื่องการตรวจสอบการทำงานบางอย่างของการ์ดอินพุต/เอาท์พุตพอร์ท (I/O PORT) และในเรื่องของรีจิสเตอร์ที่นำมาใช้งานในส่วนของการสื่อสารข้อมูล โดยพอจะสามารถแจงรายละเอียดได้ดังนี้

**Interrupt Identification Register (IIR) หรืออาจเรียกว่า Interrupt Status Register (ISR)**

เป็นรีจิสเตอร์ที่ใช้งานควบคู่กับรีจิสเตอร์ IER ใช้กับค่าความสำคัญของการอินเตอร์รัพต์ ซึ่งมี 4 ลักษณะในการขออินเตอร์รัพต์กับตัว UART ซึ่งมีระดับการขออินเตอร์รัพดังนี้

-การขออินเตอร์เมื่อมีการเปลี่ยนแปลงสถานะของการรับข้อมูลจากพอร์ตอนุกรม (Highest Priority)

-การขออินเตอร์เมื่อมีการรับข้อมูล

-การขออินเตอร์เมื่อมีการส่งข้อมูล

-การขออินเตอร์เมื่อมีการเปลี่ยนแปลงสถานะของโมเด็ม (Lowest Priority)

รีจิสเตอร์ IIR นี้ใช้งานเพียง 3 บิทแรก (Bito-Bit2) เท่านั้นที่ใช้งาน โดยการใช้งานทั้ง 3 บิท สัมพันธ์กันดังนี้คือ

| Bit2 | Bit1 | Bit0 | Priority | Interrupt |
|---|---|---|---|---|
| 0 | 0 | 1 | None | ไม่มีการอินเตอร์รัพท์ |
| 0 | 0 | 0 | 1 (Lowest) | เมื่อเกิดการเปลี่ยนแปลงสภานะของโมเด็ม |
| 0 | 1 | 0 | 2 | เมื่อเกิด Transmit Buffer empty |
| 1 | 0 | 0 | 3 | เมื่อมีการรับข้อมูล |
| 1 | 1 | 0 | 4 (Highest) | เมื่อมีการผิดพลาดของข้อมูลที่รับเข้ามา |

**Line control Register (LCR)**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นรีจิสเตอร์ที่ใช้ในการกำหนดพารามิเตอร์ต่างๆ ที่ใช้ในการสื่อสารข้อมูล เช่น ขนาดของข้อมูล การกำหนดพาริตี้บิทที่ใช้ในการตรวจสอบจำนวนสตอปบิท ที่ใช้ในการสื่อสารข้อมูลบนสายสัญญาณ RS-232C

| Bit | Name | Description |
|---|---|---|
| 0-1 | Character Length | กำหนดขนาดของข้อมูลที่ใช้ในการสื่อสารข้อมูล |

| Bit0 | Bit1 | Data Bit |
|---|---|---|
| 0 | 0 | 5 |
| 0 | 1 | 6 |
| 1 | 0 | 7 |
| 1 | 1 | 8 |

| | | |
|---|---|---|
| 2 | Number of Stop Bit | กำหนดจำนวนบิทสิ้นสุดของการสื่อสารข้อมูล |

| Bit2 | Stop Bit |
|---|---|
| 0 | 1 |
| 1 | 2 หรือ 1.5 |

| | | |
|---|---|---|
| 3-5 | Parity Bit | กำหนดการเช็คข้อมูลที่ใช้การสื่อสารข้อมูล |

| Bit5 | Bit4 | Bit3 | Parity |
|---|---|---|---|
| 0 | 0 | 0 | None |
| 0 | 0 | 1 | Odd |
| 0 | 1 | 1 | Even |
| 1 | 0 | 1 | Mark |
| 1 | 1 | 1 | Space |

บิท Parity ถ้าถูกกำหนดเป็น

-แบบ Odd นั้นหมายความว่าเมื่อมีการเช็คบิทที่เซ็ทเป็น 1 ทั้งหมดที่ใช้ในการสื่อสาร ถ้านับบิทนี้ได้เป็นจำนวนเลขคี่ บิท Parity ที่ส่งร่วมไปกับข้อมูลจะถูกเซ็ทเป็น 1 ด้วย แต่ถ้านับได้เป็นจำนวนเลขคู่ บิท Parity จะเป็น 0

-แบบEven จะตรงข้ามกับ odd คือถ้านับบิทที่เป็น 1 ได้จำนวนเลขคู่ บิท P)arity จะถูกเซ็ทเป็น 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | | |
|---|---|---|
| | | -แบบ Mark ที่บิท Parity จะถูกเซ็ทให้เป็น'1'เสมอไม่กรณีใดๆ<br>-แบบ Space ที่บิท Parity จะถูกรเซ็ทเป็น'0' เสมอ ซึ่งตรงข้ามกับแบบ mark |
| 6 | Break Control | ถ้าเซ็ทเป็น '1' จะหยุดการสื่อสารข้อมูลจนกว่าบิทนี้จะถูกรีเซ็ทให้เป็น 0 |
| 7 | Divisor Latch Access bit (DLAB) | ถ้าบิทนี้ถึงรีเซ็ท จะเป็นการเข้าถึงรีจิสเตอร์ RBR, THR และ IER แต่ถูกเซ็ทเป็น '1' จะใช้เป็นการกำหนดความเร็วในการสื่อสารข้อมูล |

**Modem Control Register (MCR)**

เป็นรีจิสเตอร์ที่ใช้งานในด้านการติดตั้งโมเด็มเพื่อช่วยในการควบคุมการไหลของข้อมูล (Flow Control) และการยกสายหรือวางสายโมเด็มแทนหูโทรศัพท์ เป็นรีจิสเตอร์ที่ส่งสัญญาณควบคุมจาก 8250 ไปยังโมเด็มหรือ Emulating modem

| Bit | Name | Description |
|---|---|---|
| 0 | Data Terminal Ready (DTR) | บิทนี้จะต่อใช้งาน โดยตรงกับโมเด็ม เซ้ทบิทให้เป็น '1' ขา DTR จะเป็น '1' (High) ด้วย |
| 1 | Request To | บิทนี้ใช้งานในลักษณะของ Flow Control โดยบิทนี้ถูกเซ็ทเป็น '1'จะทำให้ขา RTS ของพอร์ตอนุกรมถูกเซ็ทเป็น '1' (High) ด้วย ส่งผลให้เกิดการส่งข้อมูลไปยัง Remote DTE |
| 2-3 | GPO1,GPO2 | ใช้กำหนดคุณสมบัติของสัญญาณเอาท์พุทบิทนี้จะถูก |
| 4 | Loopback | เซ็ท '1' เมื่อต้องการทำ Self test ของ 8250 |
| 5-7 | Unused | จะถูกรีเซ็ทเป็น '0' เสมอ |

**Line Status Register (LSR)**

เป็นรีจิสเตอร์ที่ใช้แสดงสถานะ การใช้งานของการสื่อสารข้อมูล เป็นรีจิสเตอร์ที่แสดงสถานะของสายสัญญาณของขา RS-232C โดยบิทแต่ละบิทมีความหมายดังนี้

| Bit | Name | Description |
|---|---|---|
| 0 | Data Ready (DR หรือ RxRDY) | บิทนี้จะถูกเซ็ทเป็น '1' เมื่อมีการอ่านข้อมูลเข้ามาไว้ใน 8250 ให้ทราบว่ามีข้อมูลส่งเข้ามาและเมื่อทำการอ่าน |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | | |
|---|---|---|
| | | ข้อมูลจากรีจิสเตอร์ RBR แล้วจะถูกรีเซ็ทบิทเป็น '0' |
| 1 | Overrun Error (OE) | บิทนี้จะถูกเซ็ทให้เป็น '1' เมื่อข้อมูลก่อนหน้านี้ยังไม่ได้ทำการ อ่านข้อมูลเข้าไปเก็บไว้ก่อน แต่มีข้อมูลถัดมาทำการรอให้ 8250 รับข้อมูลอยู่แล้ว |
| 2 | Parity Error (PE) | บิทนี้จะถูกเซ็ทให้เป็น '1' เมื่อบิทพาริตี้ที่รับเข้ามาไม่ถูกต้องตามที่ได้กำหนดไว้ในรีจิสเตอร์ LCR |
| 3 | Framing Error (FE) | บิทนี้จะถูกเซ็ทให้เป็น '1' เมื่อเกิดการผิดพลาดของ Farm ข้อมูล |
| 4 | Break Interrupt (BI) | บิทนี้จะถูกเซ็ทให้เป็น '1' เมื่อ 8250 มีการ Detect เงื่อนไขการขอ Break บนสัญญาณขา RD |
| 5 | Trasmitter Holding Register Empty (THRE) | บิทนี้จะถูกเซ็ทให้เป็น '1' เมื่อ 8250 พร้อมที่จะทำการรับข้อมูลตัวถัดไปที่ส่งเข้ามา หรือ Transmit Buffer Empty (TBE) |
| 6. | Transmitter Shift (TSRE) | บิทนี้จะถูกเซ็ทเป็น '1' เมื่อในรีจิสเตอร์ Transmitter |
| 7. | Unused | จะถูกรีเซ็ทเป็น '0' เสมอ |

**Modem Status Register (MSR)**

เป็นรีจิสเตอร์แสดงสถานะของโมเด็ม โดยจะทำการตรวจจับสัญญาณที่ส่งมาจากเครื่องโมเด็มมาให้กับเครื่องคอมพิวเตอร์ โดยบิทแต่ละบิทในตัวรีจิสเตอร์ MSR มีความหมายดังนี้

| Bit | Name | Description |
|---|---|---|
| 0 | Delta Clear To Send (DCTS) | บิทนี้จะถูกเซ็ทเป็น '1' เมื่อสัญญาณขาของ CTS เกิดการตรวจสอบสถานะของบิทสัญญาณ CTS |
| 1 | Delta Data Set Ready (DDSR) | บิทนี้จะถูกเซ็ทเป็น '1' เมื่อสัญญาณขาของ DSR เกิดการเปลี่ยนแปลงสถานะเช่นกัน |
| 2 | Trailing Edge Ring Indicator (TERI) | บิทนี้จะถูกเซ็ทเป็น '1' เมื่อมีการตรวจจับสัญญาณเรียกกริ่งโทรศัพท์ได้ |
| 3 | Delta Data Carrier Detect (DDCD) | บิทนี้จะถูกเซ็ทเมื่อเกิดการเปลี่ยนแปลงของสัญญาณ DCD |
| 4 | Clear To Send | บิทนี้จะถูกเซ็ท เพื่อให้ทราบถึงการส่งถ่ายข้อมูล |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | | |
|---|---|---|
| | (CTS) | ระหว่างตัว DTE กับ DCE ซึ่งเมื่อทำการเซ็ทหรือรีเซ็ทบิทจะมีผลกับสัญญาณขา CTS ของพอร์ตอนุกรม ทำให้เป็นเซ็ท(High) หรือ รีเซ็ท (Low) ตามไปด้วยเช่นกัน บิทนี้มีผลในการทำ Flow Control |
| 5 | Data Set Ready (DSR) | บิทนี้จะถูกเซ็ทเป็น '1' เมื่อ DCE พร้อมที่จะทำการรับส่งข้อมูล |
| 6 | Ring Indicator(RI) | บิทนี้จะถูกเซ็ทเมื่อมีสัญญาณกริ่งโทรศัพท์เข้ามาทาง DCE |
| 7 | Data Carrier Detect (DCD) | บิทนี้จะทำการตรวจจับสัญญาณโทรศัพท์บนสายสัญญาณ RS-232 |

**การจองเนื้อที่หน่วยความจำที่ใช้ในการรับส่งข้อมูล**

ในการกำหนดขนาดหน่วยความจำที่ใช้ในการเป็นที่พักข้อมูลที่ใช้ในการรับส่งข้อมูลจะเป็นการจองเนื้อที่หน่วยความจำที่เรียกว่าบัฟเฟอร์ ไว้ใช้ในการสื่อสารข้อมูล โดยที่ขณะที่มีการรับหรือส่งข้อมูลเราจะมีการพักข้อมูลไว้ที่หน่วยความจำดังกล่าวไว้ก่อน ก่อนที่จะทำการรับหรือส่งข้อมูล ในโปรแกรมที่สร้างขึ้นมานี้ได้ทำการจองขนาดเนื้อที่ที่นำมาใช้เป็นบัฟเอร์ไว้ขนาด 2 กิโลไบท์ จำนวน 2 ชุด โดยชุดแรกเป็นบัฟเฟอร์สำหรับการรับข้อมูลจากภายนอก และอีกชุดหนึ่งเป็นบัฟเฟอร์การส่งข้อมูลออกไปยังภายนอก บัฟเฟอร์ทั้ง 2 ชุดที่ใช้นี้เรากำหนดให้เป็นแบบบัฟเฟอร์วงแหวนคือจะเพิ่มตำแหน่งขึ้นไปที่ละหนึ่งเมื่อมีการรับข้อมูลหรือส่งข้อมูลผ่าน 8250 จนถึงตำแหน่งของหน่วยความจำที่ทำเป็นบัฟเฟอร์สูงสุด และเมื่อมีการับหรือส่งข้อมูลอีก 1 ไบท์ ตำแหน่งของหน่วยความจำจะกลับมาเริ่มที่ตำแหน่งต่ำสุดคือ 0 ใหม่เป็นเช่นนี้เรื่อยไปในโปรแกรมจะใช้ตัวแปรประเภทตัวอักษร (Charaxter แบบอะเรย์ (Array) ขนาด 2048 ไบท์ โดยกำหนดชื่อตัวแปร in_buffer [] สำหรับการรับข้อมูล และ out_buffer[] สำหรับการส่งข้อมูลและมีตัวแปรที่เป็นชนิดจำนวนเต็ม เป็นตัวกำหนดตำแหน่งในการรับและส่งข้อมูลใช้ชื่อว่า Start_inbuf,end_inbuf สำหรับบัฟเฟอร์รับข้อมูลและ start-outbuf, end_outbuf สำหรับบัฟเฟอร์การส่งข้อมูล โดยการเช็คตำแหน่งสูงสุดของบัฟเฟอร์จะใช้คำสั่งดังนี้

```
if ( start_inbuf >= 2048 )
        start_inbuf = 0;
else
        start_inbuf++;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 7.2 ลำดับขั้นตอนการทำงาน

1. ศึกษาคุณสมบัติของขาสัญญาณของ RS-232 ทั้ง 8 ขาสัญญาณ คือ Rx, Tx, RTS, CTS, DTR, DSR, DCD และขา SG

2. ศึกษาคุณสมบัติของ UART 8250 ทางด้านฮาร์ดแวร์ ขาที่ใช้งานต่างๆของ 8250

3.ศึกษาการอินเตอร์รัพต์หมายเลขต่างๆที่ใช้งานในด้านเกี่ยวกับการสื่อสารข้อมูลแบบอนุกรม

4. รู้จักหน้าที่การทำงานของตัวรีจิสเตอร์ที่ใช้งานในการสื่อสารข้อมูล

5. เขียนโปรแกรมการควบคุมขาสัญญาณ RTS และ DTR ของ RS-232

6. เขียนโปรแกรมการเช็คสถานะของขาสัญญาณ CTS, DSR และ DCD ของ RS-232

7. เขียนโปรแกรมการเช็คสถานะการเปลี่ยนแปลงของขาสัญญาณ CTS, DSR และ DCD

8. เขียนโปรแกรมการเซ็ทค่าพารามิเตอร์ให้กับตัวรีจิสเตอร์ที่ใช้งานในการสื่อสารข้อมูล

9. เขียนโปรแกรมการรับส่งข้อมูลผ่านทางขาสัญญาณ Tx และ Rx ของRS-232 โดยรับหรือส่งอักขระที่ละ 1 ไบท์

10. ศึกษารหัสแอสกี้ที่ใช้ในงานด้านการสื่อสาร เช่น STX, ETX, EOT หรือ SOH ว่ามีความหมายเช่นไร

11. เขียนโปรแกรมการใช้งานจริงของโปรแกรม TFNET โดยรวม

12. เขียนโปรแกรมการเปิดไฟล์ข้อมูล และนำข้อมูลในไฟล์ที่เปิดได้มาแสดงออกทางจอภาพ

13. เขียนโปรแกรมการเข้าถึงไฟล์ต่างๆของ Floppy Driver หรือ Harddisk ที่ใช้อยู่ในปัจจุบัน

14. เขียนโปรแกรมการเข้าถึงไฟล์ในส่วนของ Subdirectory ใน Floppy Driver หรือ Harddisk ปัจจุบัน

15. เขียนโปรแกรมการอ่านแอกตริบิวท์ไฟล์

16. ศึกษาคำสั่ง AT ที่ใช้งานในตัวโมเด็ม

17. เขียนโปรแกรมการควบคุมโมเด็ม โดยการโปรแกรมคำสั่ง AT จากโปรแกรมที่พัฒนาขึ้นมา

18. เขียนโปรแกรมการการรับส่งไฟล์แบบไม่มีการเช็คเงื่อนไขว่าข้อมูลที่รับเข้ามานั้นถูกต้องหรือไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

19. ศึกษาการตรวจเช็คข้อมูลที่นำมาทำการรับส่ง ในวิธีการต่างๆ เช่นการใช้หลัก CRC เป็นต้น

20. เขียนโปรแกรมการรับส่งไฟล์แบบ Time Sharing โดยมีการตรวจสอบข้อมูลที่ใช้ในการรับส่งไปด้วยว่าข้อมูลที่รับเข้ามาได้ถูกต้องหรือไม่

21. จัดการหน้าตาเมนูของจอภาพให้มีความเหมาะสมในเรื่องการตรวจสอบข้อมูลที่ใช้ในการรับส่งไฟล์

22. พัฒนาโปรแกรมที่ใช้ฟังก์ชั่นการทำงานแบบ Chat Mode ให้มีความสามารถในการติดต่อสื่อสารกันให้สามารถใช้งานร่วมกันได้มากกว่า 2 เครื่อง

23. สร้างรูปแบบการรับส่งข้อมูลในการใช้งานของตัว PC ในการใช้งานแบบหลายตัว

24. เขียนฟังก์ชั่นการเปิดไฟล์ที่ใช้เก็บค่าพารามิเตอร์ในการสื่อสารข้อมูลของโปรแกรม

25. เพิ่มฟังก์ชั่นในการแสดงข้อความจากไฟล์ข้อมูล (XXXXX.HLP) เพื่อนำข้อมูลมาช่วยในเรื่องการใช้งานของโปรแกรม TFNET

26. เพิ่มฟังก์ชั่นการอ่านค่าพารามิเตอร์จากการผ่านค่าอาร์กิวเมนท์จากระบบปฏิบัติการ

27. เพิ่มฟังก์ชั่นในส่วนของการเรียกใช้งานระบบปฏิบัติการชั่วคราว

28. เปลี่ยนแปลงหน้าตาของฟังก์ชั่นต่างๆของโปรแกรม

29. ทดสอบการทำงานโดยรวมของฟังก์ชั่นต่างๆของตัวโปรแกรม TFNET

30. เพิ่มฟังก์ชั่นการตรวจเช็คข้อมูลบางชนิดของระบบ เช่น วัน เวลา ของ BIOS

31.เพิ่มฟังก์ชั่นการตรวจจับชุดข้อความที่รับเข้ามาเพื่อนำข้อความชุดนี้ไปประมวลคำเพื่อรับทราบว่าทางฝ่ายส่งต้องการให้ฝ่ายรับทำหน้าที่อะไร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 7.3 การพัฒนาโปรแกรม

โปรแกรม TFNET ที่สร้างขึ้นจะถูกพัฒนาขึ้นโดยการโปรแกรมภาษาซี ในรูปของโปรเจ็กไฟล์เพื่อช่วยในการศึกษาการใช้งานพื้นฐานในส่วนของการสื่อสารข้อมูล ทั้งทางด้านฮาร์ดแวร์ และซอฟต์แวร์ เพื่อให้สามารถพัฒนาโปรแกรมอย่างมีประสิทธิภาพขึ้นไปได้อย่างเป็นขั้นตอน อีกทั้งยังรู้ถึงการเซ็ทใช้งานตัวโมเด็มทั้งในด้านของฮาร์ดแวร์ และซอฟต์แวร์ ซึ่งการโปรแกรมจะใช้คำสั่ง AT ที่เป็นคำสั่งเฉพาะที่ตัวโมเด็มจะเข้าใจได้เพียงอุปกรณ์เดียวเท่านั้น ซึ่งโปรแกรม TFNET จะประกอบไปด้วยไฟล์ต่างๆดังนี้

**DIR.C**

เป็นไฟล์ที่เก็บส่วนของฟังก์ชั่นที่ทำหน้าที่เกี่ยวกับ Directory ของไฟล์ หรือการเข้าถึงไฟล์ใน Directory นั้นๆ ซึ่งฟังก์ชั่นในไฟล์จะใช้งานร่วมกับหน้าที่ของการรับส่งไฟล์ โดยในไฟล์ DIR.C จะประกอบไปด้วยฟังก์ชั่นที่ใช้งานดังนี้

ฟังก์ชั่น Put_Screen (); เป็นฟังก์ชั่นแสดงข้อความต่างๆเพื่อบอกให้ผู้ใช้ทราบว่าตารางช่องไหนเป็นการแสดงชื่อไฟล์ หรือช่องไหนแสดงอะไร

ฟังก์ชั่น File_Box (); เป็นฟังก์ชั่นตีกรอบของเมนูการแสดงชื่อไฟล์

ฟังก์ชั่น Sort_File (); เป็นฟังก์ชั่นจัดเก็บชื่อไฟล์ลงในตัวแปรที่ใช้งานร่วมกันในหลายๆไฟล์เพื่อการใช้งานในด้านการเลือกไฟล์ที่จะนำมาใช้

ฟังก์ชั่น Get_File_Name (); เป็นฟังก์ชั่นที่ใช้งานควบคู่กับฟังก์ชั่น Sort_File ();

ฟังก์ชั่น line_file (); เป็นฟังก์ชั่นแสดงจำนวนการเลือกใช้ไฟล์ และจำนวนขนาดของหน่วยความจำโดยรวมของไฟล์ที่ทำการเลือก

ฟังก์ชั่น Display_File (); เป็นฟังก์ชั่นที่ใช้งานควบคู่กันกับฟังก์ชั่น Show_File ();

ฟังก์ชั่น Select_File (); เป็นฟังก์ชั่นเลือกการเข้าถึงข้อมูลในแต่ละไฟล์ โดยไฟล์ที่ถูกเลือกจะแสดงตัวอักษรที่มีสีต่างไปจากปรกติ

ฟังก์ชั่น Show_File (); เป็นฟังก์ชั่นแสดงชื่อไฟล์ครั้งละ 17 ไฟล์ในเมนูแสดงไฟล์ โดยจะแสดงทั้งขนาดข้อมูล เวลาและวันที่ ที่จัดเก็บครั้งหลังสุดของไฟล์แต่ละไฟล์

**FUNCTION.C**

เป็นไฟล์ที่รวมส่วนหลักของหน้าที่การทำงานในโปรแกรม TFNET ตามที่ได้แสดงบนเมนูหลักที่ปรากฏบนจอ โดยไฟล์นี้จะรวมหน้าที่การทำงานทั้งหมดที่มีการเรียกใช้ในเมนูฟังก์ชั่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชั่นโดยหลักที่แสดงอยู่บนจอมอนิเตอร์ โดยในไฟล์ FUNCTION.C นี้จะประกอบไปด้วยฟังก์ชั่นที่ใช้งานดังนี้

ฟังก์ชั่น Talk_PC (); เป็นฟังก์ชั่นการรับส่งอักขระทั้งแบบแพคเกท (สูงสุด 70 ตัวอักขระ) หรือแบบ 1 อักขระ (1ไบท์) โดยสามารถใช้งานในลักษณะการรับส่งเฉพาะเครื่องได้ คือสามารถกำหนดหมายเลขเครื่องที่ต้องการติดต่อได้ โดยยังสามารถติดต่อกับเครื่องหมายเลขอื่นได้เช่นกันแต่ทางฝั่งเครื่องหมายเลขอื่นนี้ไม่สามารถมองเห็นข้อความที่ส่งมาได้ฟังก์ชั่น Transfer_File (); เป็นฟังก์ชั่นการรับส่งไฟล์ระหว่าง DTE 2 ตัว โดยสามารถใช้งานรับส่งไฟล์ทั้งแบบผ่าน MODEM หรือ Null MODEM

ฟังก์ชั่น Do_Dial_Modem (); เป็นฟังก์ชั่นช่วยในการหมุนเลขหมายโทรศัพท์ให้กับตัวโมเด็มโดยในฟังก์ชั่นนี้จะมีการใช้งานด้านคำสั่ง AT ในการหมุนเลขหมายโทรศัพท์

ฟังก์ชั่น Dial_And_HangUp (); เป็นฟังก์ชั่นเลือกการใช้งานของตัวโมเด็มว่า จะทำการหมุนหมายเลขโทรศัพท์ หรือทำการยกหูโทรศัพท์เมื่อมีการโทรเข้ามายังเครื่อง

ฟังก์ชั่น Set_Comm_Status (); เป็นฟังก์ชั่นกำหนดการทำงานของพอร์ตอนุกรมว่าการใช้งานด้านการสื่อสารเป็นแบบผ่าน DCE ไปยัง DCE Remote หรือการต่อใช้งานในลักษณะ Null MODEM ซึ่งจะไม่มีการตรวจเช็กบิท DSR และบิท DTR

ฟังก์ชั่น Auto_Answer (); เป็นฟังก์ชั่นที่ใช้การตอบรับเมื่อมีการโทรเข้ามายังเครื่องโมเด็มโดยฟังก์ชั่นนี้จะเข้าถึงส่วนของฮาร์ดแวร์โมเด็ม และการโปรแกรมผ่านทางคำสั่ง AT

ฟังก์ชั่น Pass_Word (); เป็นฟังก์ชั่นที่เพิ่มเข้ามาเพื่อช่วยในการป้องกันการเข้ามาทำการเซ็ทค่าพารามิเตอร์ของสถานะการสื่อสารข้อมูลของผู้อื่นที่ไม่ใช่ผู้ใช้งานจริง

ฟังก์ชั่น Chang_Pass (); เป็นฟังก์ชั่นที่ใช้แก้ไขเปลี่ยนแปลงรหัสผ่านการเข้าไปแก้ไขพารามิเตอร์ด้านการสื่อสาร

ฟังก์ชั่น Init_Port (); เป็นฟังก์ชั่นการติดตั้งการสื่อสารข้อมูล ที่ใช้กำหนดมาตราฐานในการสื่อสารระหว่างเครื่อง PC 2 เครื่อง

ฟังก์ชั่น Get_Information (); เป็นฟังก์ชั่นที่จะแสดงข้อมูลเกี่ยวกับส่วนประกอบต่างๆ ภายนอกรอบๆโปรแกรม

ฟังก์ชั่น OS_Shell (); เป็นฟังก์ชั่นที่เรียกใช้งานคำสั่งด้านระบบปฏิบัติการชั่วคราว

ฟังก์ชั่น Check_Exit_Program (); เป็นฟังก์ชั่นตรวจสอบการยืนยันการขอยกเลิกการใช้งานโปรแกรมการสื่อสารข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

GLOBAL.C

เป็นไฟล์ที่เก็บรวบรวมฟังก์ชั่นต่างๆที่ใช้งานบ่อยๆ ในโปรแกรม TFNET โดยส่วนมากจะเป็นฟังก์ชั่นเกียวกับการจัดการในเรื่องของจอภาพ และถูกเรียกใช้งานจากฟังก์ชั่นต่างๆอยู่บ่อยครั้งโดยในไฟล์ GLOBAL.C นี้จะประกอบไปด้วยฟังก์ชั่นที่ใช้งานดังนี้

ฟังก์ชั่น Wait_Press_Keybroad_botton (); เป็นฟังก์ชั่นที่รอการกดปุ่มใดๆก็ได้บนแป้นพิมพ์เพียงพอยกเลิกการทำงานที่ทำอยู่ในขณะนั้น

ฟังก์ชั่น Cursor_Off (); เป็นฟังก์ชั่นปิดการแสดงผลของตัวเคอร์เซอร์เพื่อความสวยงามในการแสดงผล

ฟังก์ชั่น Cursor_On (); เป็นฟังก์ชั่นเปิดการแสดงผลของตัวเคอร์เซอร์เพื่อให้ทราบถึงว่ามีการกดปุ่มเพื่อป้อนตัวอักษรไว้ทำงานใดงานหนึ่ง

ฟังก์ชั่น Box (); เป็นฟังก์ชั่นแสดงการตีกรอบของหน้าต่างเมนูฟังก์ชั่นต่างๆ

ฟังก์ชั่น BGBox (); เป็นฟังก์ชั่นแสดงเงาของกรอบเมนูฟังก์ชั่นที่ทำขึ้นมา

ฟังก์ชั่น Screen_Bar (); เป็นฟังก์ชั่นแสดงสถานะค่าพารามิเตอร์ที่กำหนดขึ้นมาสำหรับการสื่อสารข้อมูล

ฟังก์ชั่น Move_Up (); เป็นฟังก์ชั่นการเคลื่อนย้ายบล็อกในแนวทแยงขึ้น

ฟังก์ชั่น Move_Down (); เป็นฟังก์ชั่นการเคลื่อนย้ายบล็อกในแนวทแยงลง

ฟังก์ชั่น Show_Name (); เป็นฟังก์ชั่นที่ใช้แสดงลักษณะของโปรแกรมที่พัฒนาขึ้นมา

ฟังก์ชั่น Show_Programmer (); เป็นฟังก์ชั่นเช่นเดียวกันกับฟังก์ชั่น Show_Name ();

ฟังก์ชั่น ClearTo (); เป็นฟังก์ชั่นลบข้อมูล(เคลียร์หน้าจอภาพ)ในบริเวณที่กำหนดขึ้น

ฟังก์ชั่น Show_Current_Drive (); เป็นฟังก์ชั่นแสดงการใช้งานของ Driver ปัจจุบันที่ใช้งานร่วมกันกับโปรแกรม TFNET

ฟังก์ชั่น Show_Dos_Ver (); เป็นฟังก์ชั่นแสดงรุ่นของระบบปฏิบัติการที่ใช้งานอยู่ในปัจจุบัน

ฟังก์ชั่น Show_Memory (); เป็นฟังก์ชั่นแสดงขนาดของหน่วยความจำทั้งหมดที่ใช้งานของระบบทั้งที่ถูกใช้งานไปแล้ว หรือยังไม่ถูกใช้งานก็ตาม ซึ่งก็คือการแสดงขนาดของความจุของหน่วยความจำสำรอง ประเภท Harddisk หรือ Floppydisk

ฟังก์ชั่น Get_Video_Mode (); เป็นฟังก์ชั่นแสดงโหมดการใช้งานของจอมอนิเตอร์ที่ใช้งานอยู่ในขณะนั้น

ฟังก์ชั่น Get_Mouse_Status (); เป็นฟังก์ชั่นตรวจสอบการติดตั้งการใช้งานเมาท์

ฟังก์ชั่น Get_Date (); เป็นฟังก์ชั่นแสดงวันที่ใน BIOS ที่ผู้ใช้กำหนดขึ้นมาในปัจจุบัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชั่น Get_Parameter (); เป็นฟังก์ชั่นแสดงค่าพารามิเตอร์ของการสื่อสาร

ฟังก์ชั่น Get_Time (); เป็นฟังก์ชั่นแสดงเวลาใน BIOS ที่กำลังเดินอยู่ในปัจจุบัน โดยผู้ใช้จะเป็นผู้กำหนดเวลาที่จะตั้งขึ้นมาในปัจจุบันเอง

ฟังก์ชั่น Error_Message (); เป็นฟังก์ชั่นแสดงข้อความที่ต้องการออกทางจอภาพ ในลักษณะเช่นเดียวกันกับการแสดงหน้าต่างเมนู โดยอาจจะมีการเช็คการกดคีย์ก่อนการยกเลิกการแสดงผลหรือให้สามารถแสดงผลตัวอักษรแบบกระพริบก็ได้

**HELPNET.C**

เป็นไฟล์ที่เรียกใช้งานในขณะที่ผู้ใช้งานต้องการความช่วยเหลือในเรื่องของข้อมูลในการใช้งานส่วนต่างๆของฟังก์ชั่นในโปรแกรม TFNET เพื่อให้ผู้ใช้งานสามารถใช้งานได้อย่างสะดวกรวดเร็ว โดยในไฟล์ HELPNET.C นี้จะประกอบไปด้วยฟังก์ชั่นที่ใช้งานในด้านการแสดงข้อความในลักษณะของโปรแกรมอิดิเตอร์ แต่ไม่สามารถเข้าไปแก้ไขข้อมูลที่ปรากฏในเมนูนั้นได้ นอกเสียจากการเข้าไปแก้ไขไฟล์ข้อมูลนั้นๆในโปรแกรมอิดิเตอร์ต่างๆที่มีใช้งานอยู่ทั่วๆไป ในไฟล์ HELPNET.C จะประกอบไปด้วยฟังก์ชั่นดังนี้

ฟังก์ชั่น Help_Box (); เป็นฟังก์ชั่นดีกรอบของบล็อกที่ใช้ในการแสดงข้อความ หลักการทำงานคล้ายกับการใช้งานฟังก์ชั่น Box ();

ฟังก์ชั่น Show_Help (); เป็นฟังก์ชั่นที่ใช้เช็คการกดคีย์ในการพลิกหน้าของข้อความในหน้าต่างของเมนูที่มีข้อความยาวๆ

ฟังก์ชั่น Select_Help (); เป็นฟังก์ชั่นที่เลือกการเปิดไฟล์ข้อความที่เก็บข้อความในการอธิบายการทำงาน ตลอดจนการใช้งานด้านคีย์ต่างๆที่นำมาใช้งานในโปรแกรม TFENT

**INITIAL.C**

ไฟล์ initial.c เป็นไฟล์ที่รวมฟังก์ชันในการ Initialize พอร์ตอนุกรมของเครื่อง คอมพิวเตอร์เซ็ทการทำงานของพอร์ตอนุกรมตั้งแต่เริ่มกำหนดค่าความเร็วในการรับส่งข้อมูล กำหนดขนาดของข้อมูล จำนวนบิทสิ้นสุดการเช็คบิทข้อมูลที่ใช้ในการรับส่งข้อมูล โดยจะเข้าถึงการใช้งานกับตัวรีจิสเตอร์ที่ใช้ในการรับส่งข้อมูลในตัว 8250 UART (Universal Asynchronous Receiver /T^ransmitter) โดยข้อมูลที่นำมาเซ็ทค่าพารามิเตอร์นี้ จะได้มาจาก 3 ทาง คือ 1. จากการพิมพ์ตัวแปร Angument ในขณะเรียกใช้งานเข้ามาจากระบบปฏิบัติ (Dos) 2. จากทางไฟล์ Modem. PAR 3. จากทางตัวโปรแกรมภายในที่กำหนดขึ้นเมื่อไม่สามารถหาค่าพารามิเตอร์ได้จาก 2 วิธีก่อนหน้านี้ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยในการเซ็ทค่าพารามิเตอร์นั้นจะให้ความสำคัญของการรับค่า Argument มาก่อนตามลำดับ และมายังการรับค่าพารามิเตอร์ที่อยู่ในไฟล์ MODEM.PAR โดยภายในไฟล์จะประกอบไปด้วยข้อมูล หมายเลขพอร์ตอนุกรมที่ใช้งาน ความเร็วในการสื่อสาร จำนวนบิทข้อมูล จำนวนสต๊อปบิท การเช็คพาริตี้ ชุดอักขระ ป้องกันการเข้าไปทำการเซ็ทค่าพารามิเตอร์การสื่อสาร (Password) หมายเลข Terminal Node และหมายเลข Remote Node ในไฟล์ Initial.c นี้ยังประกอบไปด้วยการรีเซ็ทบิทใช้งานของขาสัญญาณในพอร์ตอนุกรมอีกด้วย โดยจะเป็นฟังก์ชั่น Reset_Signal (); ในไฟล์ INITIAL.C นี้จะประกอบไปด้วย

ฟังก์ชั่น Initialize_Commu(); เป็นฟังก์ชั่นจัดการเก็บค่าตารางเวคเตอร์เก่าของพอร์ตอนุกรมทั้ง COM1 และ COM2 และทำการเซ็ทค่าเริ่มต้นของบัฟเฟอร์ที่ใช้ในการรับส่งข้อมูล ฟังก์ชั่นนี้จะใช้งานร่วมกันกับการเซ็ทค่าพารามิเตอร์ในการรับส่งข้อมูลที่อ่านมาจากไฟล์ MODEM.PAR หรือการผ่านค่าอาร์กิวเมนต์จากระบบปฏิบัติการ (DOS)

ฟังก์ชั่น Initialize_Port(); เป็นฟังก์ชั่นที่มีหน้าที่เดียวกันกับฟังก์ชั่น Initialze_Commu(); แต่จะแตกต่างกันในเรื่องการใช้งาน ซึ่งจะทำงานเมื่อการอ่านค่าในไฟล์ MODEM.PAR เกิดการผิดพลาด หรือการผ่านค่าอาร์กิวเมนต์เข้ามาผิดพลาด จึ้งจะมาทำการเซ็ทค่าพารามิเตอร์ที่ฟังก์ชั่นนี้

ฟังก์ชั่น Init_ComPort (); เป็นฟังก์ชั่นเช่นเดียวกับฟังก์ชั่น Initialize_Commu ();

ฟังก์ชั่น interrupt Com_Interrupt_Handler (); เป็นฟังก์ชั่นอินเตอร์รัพต์

ฟังก์ชั่น Set_Vecter_ComPort (); เป็นฟังก์ชั่นจัดเก็บค่าตารางเวคเตอร์ที่ใช้งานก่อนทำการเซ็ทการใช้งานตามที่กำหนดในโปรแกรม TFNET

ฟังก์ชั่น Reset_Signal (); เป็นฟังก์ชั่นรีเซ็ทบิท RTS และบิท DTR ก่อนยกเลิกการทำงานของโปรแกรม TFNET

ฟังก์ชั่น Return_Vecter_ComPort (); เป็นฟังก์ชั่นคืนค่าตารางเวคเตอร์ให้กับตำแหน่งเวคเตอร์เดิม ก่อนออกจากโปรแกรม TFNET

**MAIN.C**

เป็นไฟล์หลักของโปรแกรมที่เรียกใช้งานฟังก์ชั่นต่างๆของเมนู และเช็คการผ่านค่าอาร์กิวเมนต์ Argument จากระบบปฏิบัติการ (DOS) โดยในไฟล์ MAIN.C นี้จะประกอบไปด้วยฟังก์ชั่นที่ใช้งาน ดังนี้

ฟังก์ชั่น Read_Parameter (); เป็นฟังก์ชั่นการเปิดอ่านไฟล์ MODEM.PAR ซึ่งเป็นไฟล์ที่เก็บค่าตัวแปรที่ใช้งานในการติดตั้งมาตราฐานของการสื่อสาร

ฟังก์ชั่น Save_Parameter (); เป็นฟังก์ชั่นจัดเก็บค่าตัวแปรด้านการกำหนดพารามิเตอร์ด้านการสื่อสารลงไฟล์ MODEM.PAR ก่อนทำการยกเลิกการทำงานของโปรแกรม TFNET

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ฟังก์ชั่น** Read_Argv (); เป็นฟังก์ชั่นตรวจสอบการส่งผ่านค่าอาร์กิวเมนท์ จากระบบปฏิบัติการก่อนเรียกใช้งานโปรแกรม TFNET

**PACKET.C**

**ไฟล์** PACKET.C จะมีหน้าที่โดยรวมของไฟล์นี้คือ จัดการในเรื่องของแพคเกทข้อมูลที่จะใช้ในการรับส่งข้อมูลออกทางพอร์ตอนุกรม โดยยึดหลักการรับส่งข้อมูลแบบ XMODEM โดยประกอบไปด้วยฟังก์ชั่นต่างๆดังนี้

**ฟังก์ชั่น** Put_Packet(); จะทำการเติมอักขระ SOH STX ไว้หน้าข้อมูลที่จะทำการส่งออกไป เพื่อเป็นการบอกจุดเริ่มต้นของแพคเกทที่ทำการส่งข้อมูล และที่ตำแหน่งท้ายข้อมูลจะทำการใส่โค๊ดในการเช็คข้อมูลที่ส่งข้อมูลที่ส่งไปขนาด 2 ไบท์พร้อมด้วยอักขระETX ที่ใช้ในการบอกจุดสิ้นสุดของแพคเกทที่ใช้ส่งโดยมีรูปแบบแพคเกทข้อมูลดังนี้

[ SOH | STX | Data | CRC Code | ETX ]

**ฟังก์ชั่น** Out_Packet (); เป็นฟังก์ชั่นที่ทำการถอดแพคเกทที่ใส่ข้อมูลส่งมา โดยจะทำการถอดอักขระ SOH, STX, CRC Code และ ETX ตามลำดับ

**ฟังก์ชั่น** Put_ID_Node(); จะมีหน้าที่คล้ายกับฟังก์ชั่น Put_Packet(); แต่จะมีการเพิ่มเติมชุดข้อมูลอีก 2 ชุดคือ ชื่อ Node ของทางด้าน Terminal DTE ที่ได้กำหนดขึ้นและชื่อ Node ของทางด้าน Remote DTE ที่จะส่งออกทางพอร์ตอนุกรม โดยข้อมูลแพคเกทที่ใช้ส่งมีรูปแบบดังนี้ [ SOH | STX | ID REMOTE | | Data | CRC Code | | ID TERMINAL | ETX ]

**ฟังก์ชั่น** Out_ID_Node(); มีหน้าที่คล้ายกับฟังก์ชั่น Out_Packet(); เช่นกัน ซึ่งถ้าชื่อ Node ID REMOTE ที่ส่งเข้ามา ถูกต้องตรงกับชื่อ Node ID Terminal ของทางด้านรับจะทำการนำข้อมูลที่รับมาได้แสดงข้อมูลออกทางจอภาพแสดงผล แต่ถ้าไม่ถูกต้องทางด้าน REMOTE DTE จะไม่ทำการรับข้อมูลแพคเกทนั้น และไม่แสดงผลออกทางจอ

**PHONE.C**

เป็นไฟล์ที่จัดการกับ Modem และการติดต่อสื่อสารข้อมูลระหว่าง DTE 2 ตัวขึ้นไป ซึ่งไฟล์จัดการในการติดต่อกับโมเด็ม และการรับส่งข้อมูลขนาด 1 ไบท์ หรือแบบแพคเกทโดยมีฟังก์ชั่นใช้งานดังนี้

**ฟังก์ชั่น** Check_Modem(); เป็นฟังก์ชั่นที่เช็คการตอบสนองของตัวโมเด็มซึ่งถ้ามีการต่อใช้งานกับโมเด็ม โมเด็มจะมีการส่งสัญญาณที่ขา CTS (Clear To Send)และขาสัญญาณ DSR (Data Set Ready) กลับมาให้ทาง PC ซึ่งในการตรวจเช็คว่า PC มีการต่อใช้งานกับโมเด็มหรือไม่นั้น จึงทำการตรวจสอบที่ขาสัญญาณ CTS และ DTR นี้ ซึ่งฟังก์ชั่นนี้ได้ทำการตรวจจับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณ CTS หรือ DTR ในบิทที่ 4 หรือในบิทที่ 5 ของรีจิสเตอร์ MSR ซึ่งถ้ามีต่อใช้งานกับโมเด็มจะมีการส่งค่า '1' กลับ แต่ถ้าไม่มีการต่อใช้งานร่วมกับโมเด็มจะส่งค่า '0' กลับมา

ฟังก์ชั่น Dial_Modem(char PhoneNumber[]); เป็นฟังก์ชั่นที่ใช้ในการหมุนหมายเลขโทรศัพท์ที่ส่งไปให้โมเด็มช่วยต่อคู่สายโทรศัพท์อีกด้านหนึ่งให้ โดยมีการใช้งานคำสั่งของ AT COMMAND กับตัวโมเด็มสำหรับการควบคุมการใช้งานของตัวโมเด็ม เช่น ยกหู หรือวางหูของตัวโมเด็ม หรือการส่งหมายเลขโทรศัพท์ไปต่อยังคู่สายโทรศัพท์ที่ต้องการเรียก

ฟังก์ชั่น Talk(); เป็นฟังก์ชั่นรับส่งข้อมูลขนาด 1 ไบท์โดยจะแสดงอักขระที่ส่งและรับในจอภาพแสดงผลตามไปด้วย ซึ่งอักขระที่ใช้ในการรับส่งข้อมูล จะเป็นชุดของอักขระ ASCII (A-Z, a-z หรือตัวเลข0-9 หรืออักขระเครื่องหมายพิเศษทั่วๆไป)

ฟังก์ชั่น Chat_Mode(); เป็นฟังก์ชั่นรับส่งข้อมูลแบบแพคเกทโดยสามารถมีข้อมูลได้สูงสุด 70 ตัวอักขระในการรับส่งข้อมูลแต่ละครั้ง ซึ่งหลักการแสดงผลออกทางจอภาพก็เช่นเดียวกับการใช้งานฟังก์ชั่น Talk();

ฟังก์ชั่น Contact_Node(); เป็นฟังก์ชั่นที่ต้องมีการกำหนดโหนดการใช้งานของเครื่องแต่ละเครื่องว่าจะต่อใช้งานกับ Node ที่มีเลขหมายอะไร ซึ่งต้องมีการกำหนดชื่อ Node ก่อน (ซึ่งสามารถกำหนดชื่อ Node ได้ทั้งตัวเลข ตัวอักขระ A-Z a-z หรือตัวอักขระเครื่องหมายพิเศษบางตัว) ซึ่งการทำหน้าที่ของฟังก์ชั่นนี้ก็คือ PC ทางด้าน Terminal DTE (ซึ่งมีชื่อ Node ใช้งานซึ่งต้องไม่ซ้ำกับ DTE ตัวอื่น) จะส่งแพคเกทไปยัง Remote DTE โดยจะกำหนดชื่อ Node ที่จะติดต่อสื่อสารกันโดยตรงจากทาง Terminal DTE เป็นตัวกำหนดชื่อ Node ทางด้าน Remote DTE ใน การรับข้อมูลทุกครั้งจะทำการเช็คชื่อ Node ว่าเป็นชื่อเดียวกันกับ Remote DTE ที่ต้องการจะส่งหรือไม่ ถ้าเป็น Node ที่จะติดต่อสื่อสารด้วยจะทำการรับข้อมูลที่ส่งมาแสดงออกทางจอแสดงผล ฟังก์ชั่นนี้สนับสนุนการสื่อสารข้อมูลในลักษณะของการใช้ PC ต่อใช้งานร่วมกันมากกว่า2 เครื่องขึ้นไปในลักษณะการต่อสัญญาณ RS-232 แบบ Null Modem (ซึ่งยังอยู่ในการพัฒนาโปรแกรม)

RS232.C

เป็นไฟล์ที่ใช้ในการรับส่งข้อมูลออกทางพอร์ตอนุกรม RS-232 ที่เป็นแบบแพคเกทหรือเป็นแบบตัวอักขระเพียงตัวเดียว โดยในไฟล์ RS232.C นี้จะประกอบไปด้วยฟังก์ชั่นที่ใช้งานดังนี้

ฟังก์ชั่น Send_Char (); เป็นฟังก์ชั่นส่งข้อมูลขนาด 1 ไบท์ออกทางพอร์ตอนุกรม

ฟังก์ชั่น Send_Command (); เป็นฟังก์ชั่นส่งคำสั่งการใช้งานไปควบคุมการทำงานของตัวโมเด็ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชั่น Send_Packet (); เป็นฟังก์ชั่นส่งแพคเกทข้อมูลออกทางพอร์ตอนุกรม โดยสามารถส่งข้อมูลได้สูงสุด 128 ไบท์

ฟังก์ชั่น get_in_buffer (); เป็นฟังก์ชั่นเก็บค่าข้อมูลที่รับมาได้ในบัฟเฟอร์วงแหวนของการรับข้อมูล

ฟังก์ชั่น Initialize_Node (); เป็นฟังก์ชั่นกำหนดชื่อโหนดของตัวเครื่อง และชื่อโหนดของตัวเครื่องที่จะติดต่อด้วย

**TRANSFER.C**

เป็นไฟล์ที่ทำหน้าที่จัดการในเรื่องของการรับส่งข้อมูลในไฟล์ ระหว่างตัวเครื่องคอมพิวเตอร์ 2 ตัว โดยมีฟังก์ชั่นที่นำมาใช้งานดังนี้

ฟังก์ชั่น CrcUpData (); เป็นฟังก์ชั่นประมวลผลค่าข้อมูลที่จะทำการรับส่งข้อมูล เพื่อตรวจเช็คค่าความผิดพลาดในการรับส่งสำหรับสถานีส่ง และสถานีรับ โดยจะทำการคำนวณค่าข้อมูลเพื่อหาน้ำหนักของข้อมูลที่นำมาทำการรับส่งดังนี้

```
VALUE = (*CAL >> 8) ^ DATA ;
*ACC = (*CAL << 8) ^ TABLE_CRC [VALUE] ;
```

ในบรรทัดแรกเป็นการคำนวณหาค่าตำแหน่งที่จะนำมากำหนดตำแหน่งในตารางข้อมูล CRC ต่อจากนั้นทำการคำนวณเพื่อให้ได้ค่าบิทเช็ค CRC ที่ใช้เช็คข้อมูล

ฟังก์ชั่น Calc_Crc (); เป็นฟังก์ชั่นการเรียกใช้งานคำนวณค่า CRC ของชุดข้อมูลโดยที่จะนำมาคำนวณที่ละ 1 ตัวอักขระ (1 ไบท์) ของข้อมูลจนสิ้นสุดชุดข้อมูลนั้น

ฟังก์ชั่น getstr (); เป็นฟังก์ชั่นในการจัดการเรื่องการรับข้อมูลจากคีย์บอร์ดว่าเป็นตัวเลขหรือเป็นตัวอักขระ หรือรวมกันทั้งตัวเลข และอักขระ

ฟังก์ชั่น Meter (); เป็นฟังก์ชั่นแสดงการทำงานในการรับส่งไฟล์เป็นอย่างรูปแบบบาร์

ฟังก์ชั่น Check_Status~ ~(); เป็นฟังก์ชั่นเช็คการเชื่อมต่อของเครื่องคอมพิวเตอร์กับตัวโมเด็มว่าต่อใช้งานกันหรือไม่ถ้าไม่ถูกต่อใช้งานจะยกเลิกการรับส่งไฟล์ทันที

ฟังก์ชั่น Init_Stack_Commu (); เป็นฟังก์ชั่นเซ็ทการเริ่มต้นใช้งานของบัฟเฟอร์ส่งข้อมูลและบัฟเฟอร์รับข้อมูล

ฟังก์ชั่น Error_Time_Out (); เป็นฟังก์ชั่นเช็กการรอคอยอักขระไม่ว่าจะทางด้านรับหรือส่งข้อมูลเพื่อไม่ให้เสียเวลาการรอคอยกันนานเกินไป

ฟังก์ชั่น Send_File (); เป็นฟังก์ชั่นหลักของการส่งไฟล์ออกทางพอร์ตอนุกรม โดยจะจัดการทั้งในเรื่องการเปิดไฟล์อ่านข้อมูล การนำข้อมูลทีละ 128 ไบท์ไปใส่รูปแบบแพคเกท ก่อนที่จะทำการส่งแพคเกทข้อมูลไปยังเครื่องคอมพิวเตอร์ด้าน Remote

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชั่น Receive_File (); เป็นฟังก์ชั่นหลักของการรับไฟล์ที่ส่งมาจากพอร์ตอนุกรม

WINDOWS.C

เป็นไฟล์ที่รวมฟังก์ชั่นเกี่ยวกับการจัดการทางหน้าจอภาพแสดงผลแบบเมนูคำสั่งเพื่อให้สามารถใช้งานได้อย่างสะดวกสบายมากยิ่งขึ้น ซึ่งการทำเมนูคำสั่งนี้เป็นการสร้างเมนูคำสั่งในลักษณะ Pull Down Menu ซึ่งในไฟล์นี้มีฟังก์ชั่นใช้งานที่รวมกันอยู่ดังนี้

ฟังก์ชั่น Menu(); เป็นฟังก์ชั่นที่จัดการแสดงหน้าต่างเมนูการใช้งานตลอดจนการใช้งานด้านการรับคีย์ โดยจะทำการเช็คการกดคีย์ลูกศรทั้ง เลื่อนซ้ายขวาบนล่าง Home End และ Hot Key ที่กำหนดขึ้นมาใช้งานในแต่ละหน้าของคำสั่งในแต่ละหน้าของเมนูฟังก์ชั่น จะถูกเรียกใช้งานในเมนูย่อยของโปรแกรม

ฟังก์ชั่น HighLight(); จะใช้งานในการแสดงการเลือกใช้เมนูหลักของฟังก์ชั่น

ฟังก์ชั่น Head_Command(); เป็นฟังก์ชั่นที่ใช้แสดงตัวอักขระที่นำมาใช้งานเป็น Hot key ในแต่ละฟังก์ชั่น เพื่อให้สามารถเลือกใช้งานฟังก์ชั่นต่างๆได้อย่างสะดวกรวดเร็ว

ฟังก์ชั่น Menu_Title(); เป็นฟังก์ชั่นเมนูหลักของคำสั่งทั้งหมดในโปรแกรม TFNET โดยจะมีหน้าที่การทำงานของฟังก์ชั่นคล้ายกับฟังก์ชั่น Menu(); ต่างกันตรงที่ ฟังก์ชั่น Menu_Title(); จะควบคุมการกดคีย์ในส่วนของเมนูฟังก์ชั่นหลักของโปรแกรมส่วนฟังก์ชั่น Menu(); จะควบคุมในส่วนของเมนูฟังก์ชั่นย่อยต่างๆของฟังก์ชั่นหลัก

ฟังก์ชั่น Draw_Main_menu(); เป็นฟังก์ชั่นสร้างเมนูของคำสั่งฟังก์ชั่นหลักทั้งหมดของโปรแกรม

ฟังก์ชั่น Main_Menu(); เป็นฟังก์ชั่นเรียกการใช้งานหลักของการสร้างเมนูฟังก์ชั่นหลักและฟังก์ชั่นย่อยของโปรแกรม

ฟังก์ชั่น Check_Menu(); เป็นฟังก์ชั่นการกำหนดการใช้งานฟังก์ชั่นหลักบนเมนูทางจอภาพแสดงผล เพื่อเลือกใช้งานฟังก์ชั่นที่เขียนขึ้นใช้งานเฉพาะอย่างไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 8
# บทวิจารณ์และสรุปผล

ในระหว่างทำการทดลองได้ประสบปัญหาหลายๆด้านทั้งในด้านของทางฮาร์ดแวร์ หรือทางซอฟ์ตแวร์ไวรัสคอมพิวเตอร์ การพัฒนาโปรแกรมผิด) โดยปัญหาส่วนใหญ่ที่เกิดขึ้น เกิดจากความผิดพลาดของทางฮาร์ดแวร์ซะส่วนใหญ่ ซึ่งแจงรายการต่างๆในการประสบปัญหาขึ้นในระหว่างการทำการทดลองได้ดังนี้

- การไม่เข้าใจในเรื่องการใช้งานบิทของการสื่อสาร

หลังจากได้ทำการศึกษาการใช้งานของตัว UART(8250) จากตำราหลายเล่ม ในช่วงแรกเกิดความสงสัยในเรื่องการเซ็ทบิทใช้งานบิท RTS และบิท CTS ที่รับเข้ามา ตำราบางเล่มเขียนว่า'ON' ซึ่งอ่านใจความในตำราจะเป็นส่วนของโลจิก '0' และในบางตำรา 'ON' จะเป็นการบอกให้ทราบว่าทำการติดต่อระหว่าง DTE และ DCE คือ บิท RTS ถูกเซ็ทบิทเป็น '1' ส่งไปยัง DCE ให้รับทราบว่า DTE ขอการ Connect กับทาง DCE และทาง DCE จะส่งโลจิก '1' ตอบมาให้ DTE งตำราขัดแย้งกันเอง

- การพัฒนาโปรแกรมในลักษณะกึ่งรีเดอร์นชั่น

โปรแกรมพัฒนาการใช้งานให้สามารถมีความยืดหยุ่นในเรื่องการเปิดใช้งานเมนูฟังก์ชั่นต่างๆเมื่อมีการเรียกใช้ฟังก์ชั่นต่างๆจะเกิดการเก็บค่าตำแหน่งการ Return ของฟังก์ชั่นที่เรียกผ่านมายังฟังก์ชั่นที่ใช้งานอยู่ในปัจจุบัน ถ้าเขียนโปรแกรมไม่ถูกต้องจะทำให้ Stack Function ใช้เก็บค่าตำแหน่งการกลับไปยังฟังก์ชั่นที่เรียกเข้ามา เกิดการเต็มของข้อมูล

- การรบกวนของไวรัส

เป็นปัญหาที่ไม่สามารถคาดเดาได้เลยว่าเกิดจากไวรัสคอมพิวเตอร์หรือไม่ ในหลายๆสาเหตุที่เกิด และที่สำคัญคือการทำลายข้อมูลที่พัฒนาโปรแกรมใน Harddisk โดยการไม่สามารถเข้าถึงข้อมูลใน Harddisk แม้จะบูตจาก Driver A ก็ตาม หรือการไม่สามารถคอมพาย์ลไฟล์ที่นำมาพัฒนาได้ บางครั้งเกิดการคอมพาย์ลไม่ได้ไปหลายวัน ทำการทดลองเปลี่ยนวันที่ใน BIOS เพิ่มหรือลดเพื่อทำการหลอกการตรวจวันที่ของการทำงานของตัวไวรัสคอมพิวเตอร์ ก็ยังไม่สามารถคอมพาย์ลไฟล์ได้อีกเช่นเคย และเมื่อคอมพาย์ลไฟล์ได้ก็ลองทำการเซ็ทวันที่ใหม่เช่นกันให้เป็นวันที่ไม่สามารถคอมพาย์ลได้ ก็สามารถคอมพาย์ลได้ และทดลองตรวจเช็คไวรัสคอมพิวเตอร์จากใน Harddisk หรือจาก Floppydisk ก็ไม่พบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การใช้งานของเครื่องคอมพิวเตอร์

ซึ่งก็เป็นปัญหาที่ไม่สามารถหาคำตอบได้ ซึ่งคิดว่าเกิดจากโครงสร้างของบอร์ดคอมพิวเตอร์ซึ่งแต่ละโรงงานจัดการในเรื่องหน่วยความจำมาไม่เหมือนกัน ทำให้เกิดสาเหตุในเรื่องหน่วยความจำการคอมพาย์ล ไม่สามารถจัดการเรื่องหน่วยความจำ บางครั้งคอมพาย์ลไปเกิดการรีเซ็ทเครื่องบางครั้งเกิดการเข้าถึงการเซ็ทค่าพารามิเตอร์ต่างๆใน BIOS

- การผิดพลาดของการ์ด I/O

การ์ดไม่สามารถทำการเซ็ทบิทต่างๆ ในพอร์ตอนุกรมได้ทำให้เกิดความเข้าใจผิดคิดว่าเกิดจากการผิดพลาดในเรื่องของโปรแกรมบ้าง เกิดจากการทำงานของไวรัสคอมพิวเตอร์บ้าง ซึ่งไม่ใช่สาเหตุ ตรวจสอบพบเมื่อทำการตรวจบิท RS-232 โดยผ่านทาง RS-232 mini tester ซึ่งพบว่าเกิดจากสาเหตุที่บัฟเฟอร์ภายในการ์ด I/O เสียคือไม่สามารถส่งกระแสออกทางพอร์ตอนุกรมได้จำนวนมากได้

- การผิดพลาดในการรับส่งของโมเด็ม

โมเด็มเกิดการผิดพลาดในส่วนของฮาร์ดแวร์ด้านการรับข้อมูล คือโมเด็มจะทำการส่งข้อมูลตลอดเวลา ปัญหานี้โมเด็มแบบ External จะมีประโยชน์มากเพราะโมเด็มชนิดนี้จะแสดงสถานะของบิทใน RS-232 ให้รับทราบทางหน้าเครื่องโมเด็มอยู่ตลอดเวลา ซึ่งโมเด็มแบบ Internal จะไม่มี LED แสดงผลในลักษณะนี้

- การเขียนโปรแกรมเรียกใช้งานฟังก์ชั่นในช่วงการใช้งานฟังก์ชั่นอินเตอร์รัพต์ไม่ได้

การเขียนฟังก์ชั่นภายใต้การใช้งานฟังก์ชั่นอินเตอร์รัพต์ โดยปรกติจะสามารถเขียนฟังก์ชั่นขึ้นมาใช้งานได้ซึ่งการเรียกใช้ฟังก์ชั่นก็สามารถเรียกใช้งานเหมือนการเรียกใช้งานฟังก์ชั่นในลักษณะปรกติ สาเหตุของการเขียนฟังก์ชั่นภายใต้ฟังก์ชั่นอินเตอร์รัพต์ไม่ได้ อาจเกิดจากการเขียนโปรแกรมในลักษณะของโปรเจ็กไฟล์ ซึ่งในแต่ละไฟล์จะมีขนาดของโปรแกรมฟังก์ชั่นที่ใหญ่พอสมควรทำให้การจัดการหน่วยความจำในการใช้งานฟังก์ชั่นอินเตอร์รัพต์ถูกจัดการเป็นอีกส่วนหนึ่งของหน่วยความจำ ทำให้การเรียกใช้งานฟังก์ชั่นในส่วนที่ถูกเรียกใช้งานอยู่ต่าง Segment กันจึงต้องทำการกระโดดเปียยังตำแหน่งของฟังก์ชั่นนั้นเกิดการผิดพลาดไปได้บ้าง

- การพัฒนาโปรแกรมโดยไม่คำนึงถึงการใช้งาน

โปรแกรมที่ทำการพัฒนาทดลองขึ้นนี้เป็นการเขียนโปรแกรมโดยใช้ในจอภาพแสดงผลแบบสีโดยไม่คิดถึงการใช้งานของจอแสดงผลแบบอื่น ซึ่งสามารถใช้งานได้เช่นเดียวกันกับจอภาพแสดงผลแบบสีแต่จะลำบากในเรื่องการใช้งานไปบ้างซักหน่อย คือไม่สามารถทราบว่ากำลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะเลือกใช้งานที่คำสั่งใดแต่ได้ทำการเพิ่มฟังก์ชั่นในการรับ Hot key เพื่อเพิ่มประสิทธิ์ภาพในการใช้งานให้มากขึ้น

- การปิดกั้นในเรื่องของการใช้งานเมาท์

การติดตั้งการใช้งานด้านการสื่อสารข้อมูล ในโปรแกรมสามารถเลือกใช้งานได้ 2 พอร์ตซึ่งสามารถเลือกติดตั้งได้ตลอดเวลาที่ใช้งานอยู่ในโปรแกรม TFNET ซึ่งถูกกั้นการใช้งานกับตัวเมาท์โดยสิ้นเชิง ซึ่งถ้าต้องการติดตั้งเมาท์จะถูกจัดการติดตั้งให้ในพอร์ตอนุกรมที่เพิ่มขึ้นอีก2 พอร์ตคือใน COM3 หรือ COM4 ซึ่งการ์ดบางการ์ดยุ่งยากในเรื่องการเซ็ทจัมเปอร์บนการ์ด บางการ์ดก็ไม่มีการเพิ่มพอร์ตอนุกรมนี้ขึ้นมาให้เลย อีกทั้งซอฟต์แวร์บางโปรแกรมไม่มีการเซ็ท COM3 หรือ COM4 มาให้อีกด้วย

- การสับสนในเรื่องของโปรแกรม

การพัฒนาโปรแกรมที่มีลักษณะเป็นแบบโปรเจ็กไฟล์นั้น จะมีลักษณะที่ยุ่งยากสับสนในเรื่องของฟังก์ชั่นการใช้งานเป็นอย่างมาก เพราะต้องคำนึงถึงผลกระทบของฟังก์ชั่นที่ถูกเรียกใช้งานมาก่อนการเรียกใช้งานในฟังก์ชั่นปัจจุบัน โดยเฉพาะในเรื่องของการจัดหน้าจอภาพแสดงผล ซึ่งจะมีผลกระทบโดยตรงกับฟังก์ชั่นที่มีการใช้งานกับจอภาพด้วยกัน ถ้าไม่ทำการเขียนฟังก์ชั่นอย่างรอบคอบพอสมควรเพื่อป้องกันการแสดงผลที่ผิดเพี้ยน

- การค้นคว้าตำราที่ขาดประสิทธิ์ภาพ

ในเรื่องของตำรานี้ ไม่แน่ใจว่าเกิดจากการค้นคว้าหาตำราไม่เป็นก็ไม่ทราบ ตั้งแต่การเดินหาตำรา หนังสือ ทั้งภาษาไทย และภาษาต่างประเทศ ตามร้านหนังสือก็ไม่ค่อยพบเนื้อหาที่เกี่ยวข้องกับโปรเจ็กที่ทำ จะมีตำราที่เกี่ยวกับ LAN หรือ Network มาก ตำราที่บอกเกียวกับเนื้อหาบางอย่างก็หาไม่ได้เช่น โค๊ดต่างๆของการใช้งานในการสื่อสารข้อมูลว่ามีรูปแบบอย่างไร (VT102 ,ANSI ฯ) หรือไม่ก็ตำราบางเล่มมีโปรแกรมตัวอย่างการสื่อสารข้อมูล แต่ขาดประสิทธิ์ภาพในการใช้งาน คือทำเป็นโปรแกรมตัวอย่างที่แสดงหลักการทำงานเพียงเท่านั้น หาได้สามารถใช้งานได้จริงซึ่งในตำราบอกว่าเป็นโปรแกรมการใช้งานจริงที่มีประสิทธิ์ภาพการใช้งาน

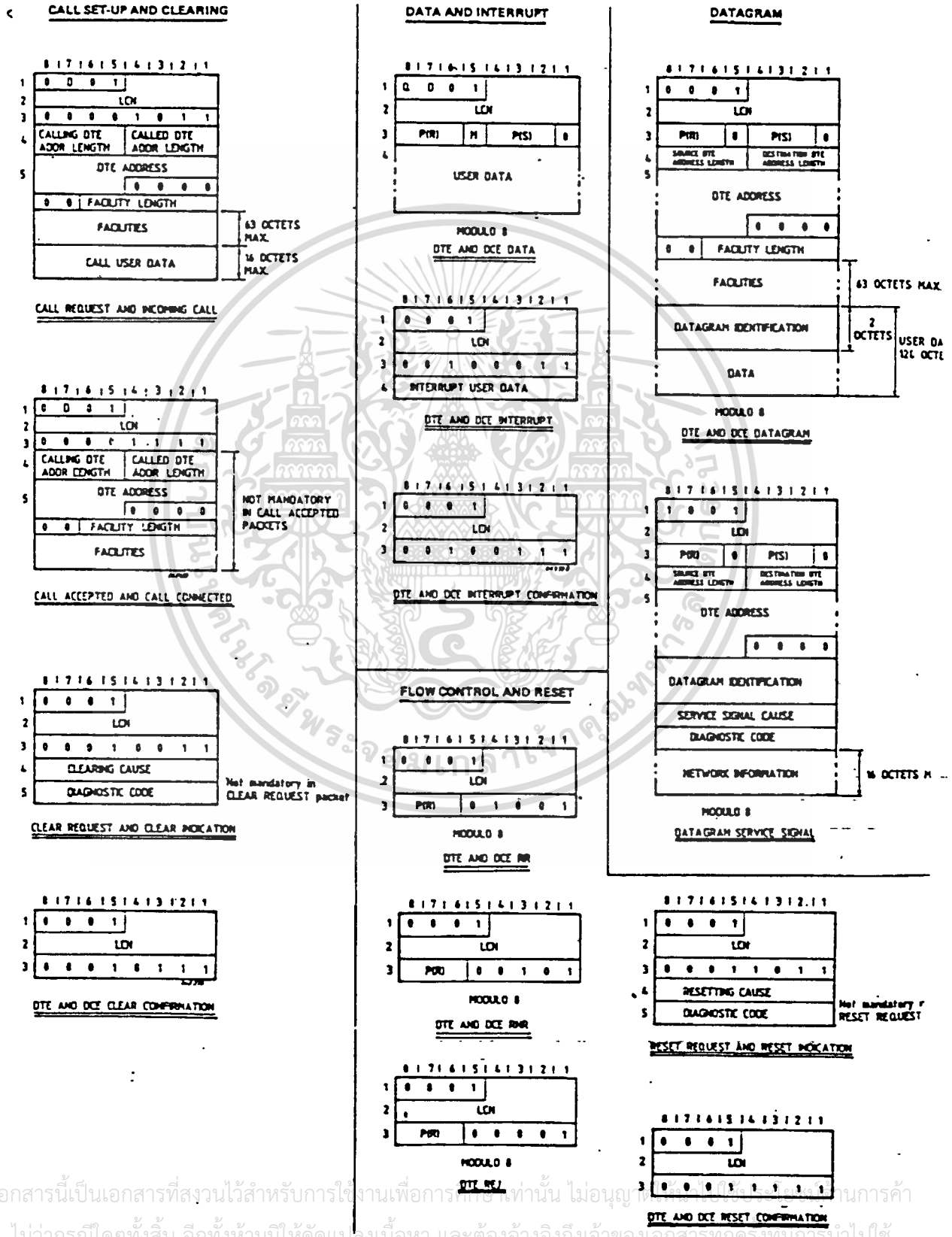
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

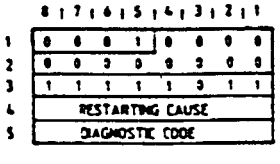
# ภาคผนวก

## Frames, Packets และ Status



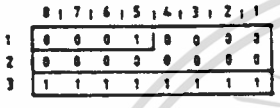

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## RESTART



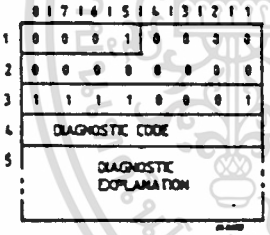

NOT mandatory in RESTART REQUEST PACKETS

RESTART REQUEST & RESTART INDICATION



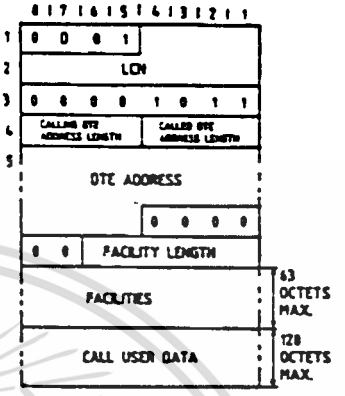

DTE & DCE RESTART CONFIRMATION

## DIAGNOSTIC

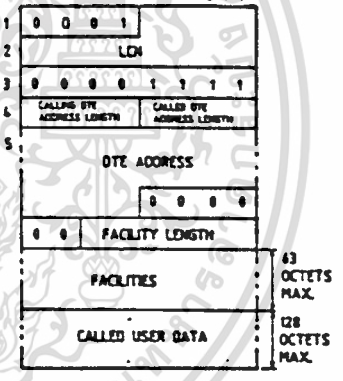



DIAGNOSTIC

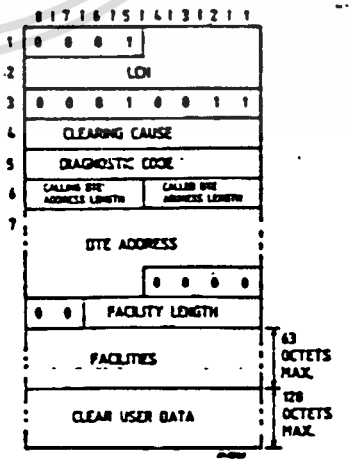
## FAST SELECT FACILITY




CALL REQUEST AND INCOMING CALL




CALL ACCEPTED AND CALL CONNECTED




CLEAR REQUEST AND CLEAR INDICATION

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## TABLES

| CLEARING CAUSE | HEX DEC CODE |
|---|---|
| DTE originated | 00 |
| Number busy | 01 |
| Out of order | 09 |
| Remote procedure error | 11 |
| Reverse charging acceptance not subscribed | 19 |
| Incompatible destination | 21 |
| Fast select acceptance not subscribed | 29 |
| Invalid facility request | 03 |
| Access barred | 0B |
| Local procedure error | 13 |
| Network congestion | 05 |
| Not obtainable | 0D |
| RPOA out of order | 15 |

| SERVICE SIGNAL CAUSE | | HEX DEC CODE |
|---|---|---|
| **DATAGRAM SERVICE SIGNAL - SPECIFIC** | | |
| **Datagram Rejected** | | |
| Local procedure error | | 13 |
| Invalid facility request | | 03 |
| Access barred | | 0B |
| Not obtainable | | 0D |
| Incompatible destination | | 21 |
| Reverse charging acceptance not subscribed | | 29 |
| **Datagram Non-Delivery Indication** | 1) | |
| Network congestion | | 05 |
| Out of order | | 09 |
| Number busy/destination queue full | | 01 |
| Remote procedure error | | 11 |
| **Datagram Delivery Confirmation** | 2) | |
| Delivery confirmation | | 21 |
| **DATAGRAM SERVICE SIGNAL - GENERAL** | | |
| Local DCE queue overflow | 3) | 1F |
| Network congestion | | 07 |
| Network operational | | 47 |

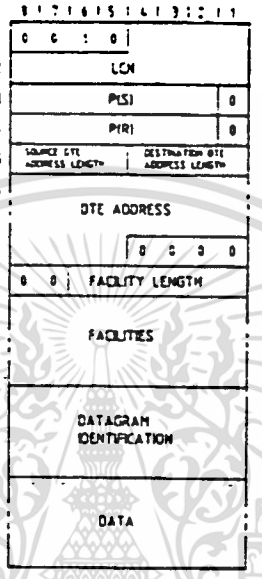
1) Issued only when the non-delivery indication facility has been requested
2) Issued only when the delivery confirmation facility has been requested
3) For further study

| RESETTING CAUSE | | HEX DEC CODE |
|---|---|---|
| DTE originated | 2) | 00 |
| Out of order | 1) | 01 |
| Remote procedure error | 2) | 03 |
| Local procedure error | | 05 |
| Network congestion | | 07 |
| Remote DTE operational | 1) | 09 |
| Network operational | 3) | 0F |
| Incompatible destination | 2) | 11 |

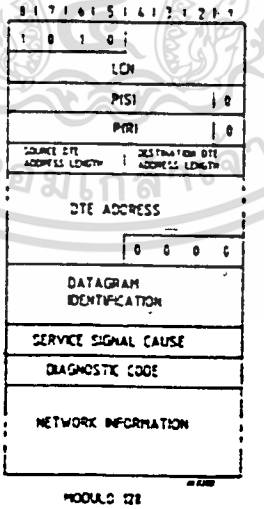
1) Applicable to PVC's only
2) Applicable to VC's and PVC's only
3) Applicable to PVC's and datagram logical channels only

| RESTARTING CAUSE | HEX DEC CODE |
|---|---|
| LOCAL PROCEDURE ERROR | 01 |
| NETWORK CONGESTION | 03 |
| NETWORK OPERATIONAL | 07 |

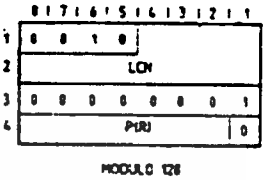
## MODULO 128



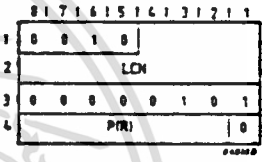

MODULO 128
DTE AND DCE DATAGRAM



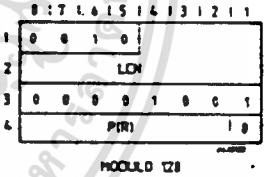

MODULO 128
DATAGRAM SERVICE SIGNAL



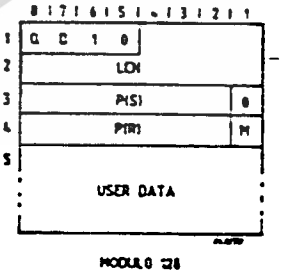

MODULO 128
DTE AND DCE RR




MODULO 128
DTE AND DCE RNR

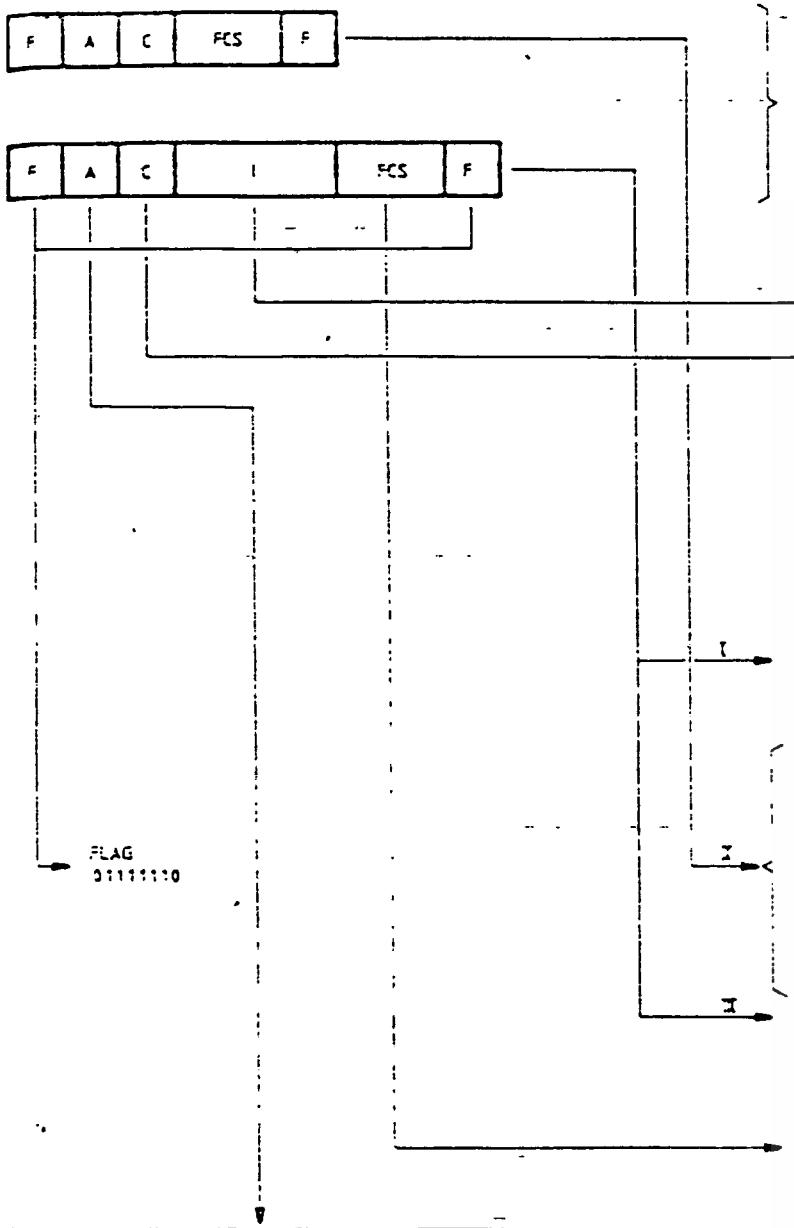



MODULO 128
DTE REJ




MODULO 128
DTE AND DCE DATA

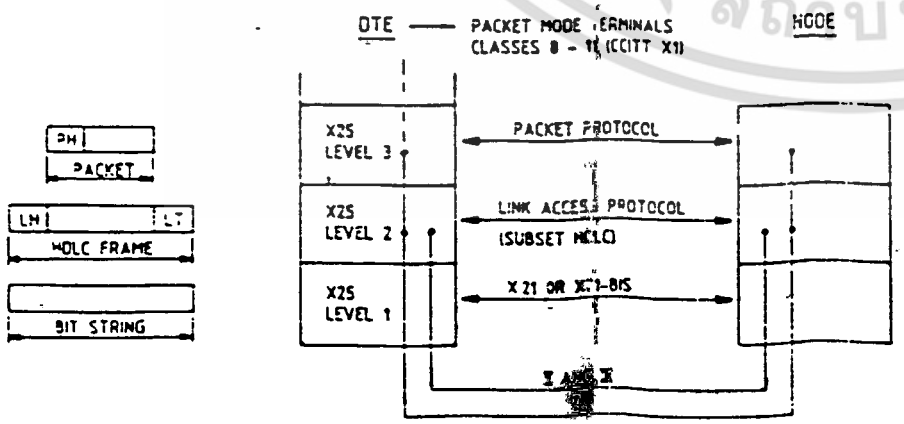
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




| COMMANDS | RESPONSES | CONTROL FIELD BITS 8 7 6 5 4 3 2 1 |
|---|---|---|
| **INFORMATION FRAMES** | | |
| I | | N(R) P N(S) 0 |
| **SUPERVISORY FRAMES** | | |
| RR (LAB-8) | RR | N(R) P/F 0 0 0 1 |
| RNR (LAP-8) | RNR | N(R) P/F 0 1 0 1 |
| REJ (LAP-8) | REJ | N(R) P/F 1 0 0 1 |
| **UNNUMBERED FRAMES** | | |
| SARM (LAP) | DM | 0 0 0 P/F 1 1 1 1 |
| SABM (LAP-8) | | 0 0 1 P 1 1 1 1 |
| DISC | | 0 1 0 P 0 0 1 1 |
| | UA | 0 1 1 F 0 0 1 1 |
| | CMDR (LAP) | 1 0 0 F 0 1 1 1 |
| | FRMR (LAP-8) | 1 0 0 F 0 1 1 1 |

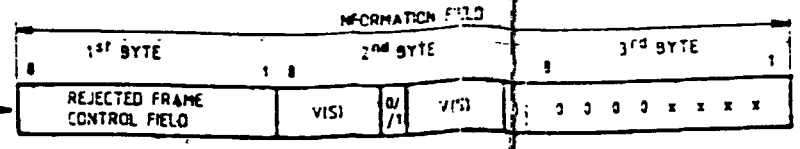
FRAME CHECK SEQUENCE (16 bits)
POLYNOMIAL $x^{16} + x^{12} + x^5 + 1$

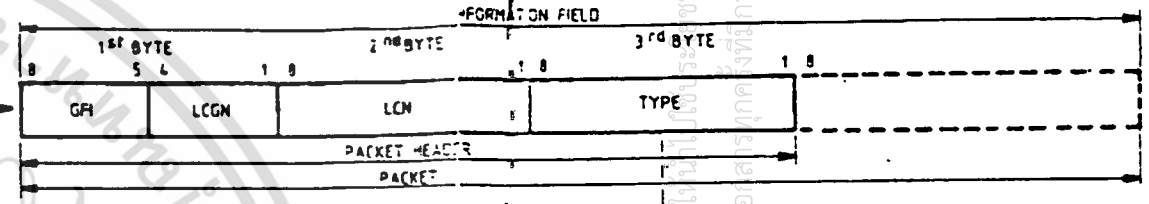
| ADDRESS FIELD (HEX: DEC.) | | |
|---|---|---|
| DIRECTION | COMMANDS | RESPONSES |
| DTE→DCE | 01 | 03 |
| DCE→DTE | 03 | 01 |










| PACKET TYPE (MODULO 8) | | OCTET 1 BITS | OCTET 3 BITS |
|---|---|---|---|
| FROM DCE TO DTE | FROM DTE TO DCE | 8 7 6 5 | 8 7 6 5 4 3 2 1 |
| **CALL SET-UP AND CLEARING** | | | |
| INCOMING CALL | CALL REQUEST | 0 0 0 1 | 0 0 0 0 1 0 1 1 |
| CALL CONNECTED | CALL ACCEPTED | 0 0 0 1 | 0 0 0 0 1 1 1 1 |
| CLEAR INDICATION | CLEAR REQUEST | 0 0 0 1 | 0 0 0 1 0 0 1 1 |
| DCE CLEAR CONFIRMATION | DTE CLEAR CONFIRMATION | 0 0 0 1 | 0 0 0 1 0 1 1 1 |
| **DATA AND INTERRUPT** | | | |
| DCE DATA | DTE DATA | 0 0 0 1 | P(R) M P(S) 0 |
| DCE INTERRUPT | DTE INTERRUPT | 0 0 0 1 | 0 0 1 0 0 0 1 1 |
| DCE INTERRUPT CONFIRMATION | DTE INTERRUPT CONFIRMATION | 0 0 0 1 | 0 0 1 0 0 1 1 1 |
| **DATAGRAM** • | | | |
| DCE DATAGRAM | DTE DATAGRAM | 0 0 0 1 | P(R) 0 P(S) 0 |
| DATAGRAM SERVICE SIGNAL | | 1 0 0 1 | P(R) 0 P(S) 0 |
| **FLOW CONTROL AND RESET** | | | |
| DCE RR | DTE RR | 0 0 0 1 | P(R) 0 0 0 0 1 |
| DCE RNR | DTE RNR | 0 0 0 1 | P(R) 0 0 1 0 0 1 |
| | DTE REJ • | 0 0 0 1 | P(R) 0 1 0 0 0 1 |
| RESET INDICATION | RESET REQUEST | 0 0 0 1 | 0 0 0 1 1 0 1 1 |
| DCE RESET CONFIRMATION | DTE RESET CONFIRMATION | 0 0 0 1 | 0 0 0 1 1 1 1 1 |
| **RESTART** | | | |
| RESTART INDICATION | RESTART REQUEST | 0 0 0 1 | 1 1 1 1 1 0 1 1 |
| DCE RESTART CONFIRMATION | DTE RESTART CONFIRMATION | 0 0 0 1 | 1 1 1 1 1 1 1 1 |
| **DIAGNOSTIC** | | | |
| DIAGNOSTIC • | | 0 0 0 1 | 1 1 1 1 0 0 0 1 |

04920

• NOT NECESSARILY AVAILABLE ON EVERY NETWORK

0 0 NOT ASSIGNED
0 1 MODULO 8
1 0 MODULO 128
1 1 EXTENSION

# กิตติกรรมประกาศ

โครงงานและปริญญานิพนธ์ฉบับนี้ได้สำเร็จลุล่วงไปได้ ด้วยการได้รับความช่วยเหลือจากหลายท่าน ขอขอบพระคุณ ดร. วรวัฒน์ ลิ้มโภคา ผู้ควบคุมให้โครงงานนี้ดำเนินไปตามขั้นตอนและให้คำเสนอแนะแนวทางแก้ไขเมื่อเกิดปัญหาขึ้น ขอขอบคุณ พี่สมศักดิ์ บริษัท โอจีเอ (OGA) จำกัด ที่ได้ให้ความรู้ในเรื่องโครงข่าย X.25 ขอขอบคุณ องค์การโทรศัพท์แห่งประเทศไทย ที่ได้ให้ข้อมูลเพื่อใช้ประกอบการทำปริญญานิพนธ์ฉบับนี้ ขอขอบคุณผู้ที่ได้ให้ความช่วยเหลืออีกหลายท่าน ตลอดจนเพื่อนๆ ที่ให้คำแนะนำต่างๆ และเป็นห่วงถามไถ่อยู่เสมอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# หนังสืออ้างอิง


ไพบูลย์ เอมพันธุ์, "คู่มือการใช้โปรแกรม Procomm Plus", ซีเอ็ดยูเคชั่น, 1989.

วิสันต์ อาชาเคโชพล, "ระบบโทรศัพท์ดิจิตอล", สำนักพิมพ์ฟิสิกส์เซ็นเตอร์, 1991.

มัณฑนา ปราการสมุทร, "การเขียนชุดคำสั่งภาษา C", ดวงกลมสมัย, 1989.

ศิวัฒน์ ศิวะบวร และ พรชัย จักรธำรงค์ม และ จิรศักดิ์ ชัยวิริยะกุล, "การประยุกต์ใช้งานภาษาซี ; ซีเอ็ดยูเคชั่น, 1988.

ศูนย์การฝึกอบรมองค์การโทรศัพท์แห่งประเทศไทย, "คู่มือช่าง X.25 DTE/DCE INTERFACE" ,โรงพิมพ์องค์การโทรศัพท์แห่งประเทศไทย, 1991.

Timothy S.Monk, "Windows Programmer's Guide to Serial Communications", SAMS Publishing, 1992.

Michael Holmes and Bob Flanders, "PC Magazine C++ Communications Utilities", Ziff-Davis Press, 1993.

Joe Campbell, "C Programmer's Guide to Serial Communications", Howard W.Sams & Company, 1987.

Uyless Black, "Data Networks Concepts,Theory andPractice", Prentice Hall, 1989.

Dave Williams,"The Programmer's Technical Referance : MS-DOS, IBM PC & Compatibles", John Wiley/Sigma Press, 1990.

Jack Purdum, "C Programmer's Toolkit", Indianapolis, 1989.

Scotts Valley, "Turbo C Reference Guide Version 2.0", Boloardland International, 1988.

DataTronics, "User's Manual Discovery 2400 BPS MODEM", Datatronics Technology, 1991.

George Sutty and Steve Blair, "Programmer's Guide to the EGA/VGA", Prentice Hall, 1988.

Herry and Rosella Kliewer and C.J. and Wilma Dyck, "EGA/VGA A Programmer's Reference Guide", McGraw-Hill, 1990.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# หนังสืออ้างอิง (ต่อ)

Stephen S.Wilson, "PC-Host Communications Strategies for Implementation",Addison Wesley Publishing Company, 1993.

Byron W.Putman, "RS-232 Simplified", Prentice-Hall, 1987.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้