บทที่ 2

ทฤษฎีและความรู้พื้นฐาน

แนวทางของโพรโตคอลมาตรฐาน

(สมใจ บุญศิริ, 2535) หน่วยงานที่เรียกว่าไอเอสโอ (ISO : International Standard Organization) ได้กำหนดมาตรฐานในการสื่อสารข้อมูลทางคอมพิวเตอร์ เรียกว่า โอเอสไอ (OSI : Open System Interconnection Model) ประกอบด้วย 7 ระดับชั้น (layer) ดังรูปที่ 2.1

7	Application Layer
6	Presentation Layer
5	Session Layer
4	Transport Layer
3	Network Layer
2	Data Link Layer
1	Physical Layer

รูปที่ 2.1 แสดงระดับชั้นมาตรฐานโอเอสไอ

หน้าที่ของแต่ละระดับชั้นแสดงได้ดังนี้

1. ระดับชั้นฟิสิคอล (Physical Layer) ทำหน้าที่ในการรับ-ส่งบิต (bit) ระหว่างจุดต่อ (node) หนึ่ง กับจุดต่อถัดไป โดยอาศัยการประสานการทำงานระหว่างอุปกรณ์ด้วยคุณสมบัติทางไฟฟ้า

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

2. ระดับดาต้าลิงค์ (Data Link Layer) ทำหน้าที่ในการรับ-ส่งชุดของข้อมูล (frame or packet) ระหว่างจุดต่อ โดยมีการตรวจสอบความถูกต้องของข้อมูล และการควบคุมอัตราการส่งถ่ายข้อมูล

3. ระดับชั้นเน็ตเวอร์ค (Network Layer) ทำหน้าที่ในการกำหนดเส้นทางการติดต่อ (path or route) จากต้นทางไปยังปลายทาง ในระบบเครือข่ายเดียวกันและระหว่างเครือข่าย

4. ระดับชั้นทรานสปอร์ต (Transport Layer) ทำหน้าที่ในการเตรียมการส่งข้อมูลระหว่างผู้ใช้ เลือกระดับการบริการที่เหมาะสมในการควบคุมการส่งข้อมูล

5. ระดับเซสชั่น (Session Layer) ทำหน้าที่ในการจัดโครงสร้าง และประสานจังหวะการติดต่อที่เกิดขึ้นระหว่างผู้ใช้ และการจัดการแลกเปลี่ยนข้อมูลในการรับ และส่งข้อมูล ให้สามารถรับและส่งข้อมูลพร้อม ๆ กันหรือสลับกันได้

6. ระดับชั้นพรีเซนเตชั่น (Presentation Layer) หน้าที่ในการเปลี่ยนแปลงรูปแบบ หรือรหัสของข้อมูลที่ส่งผ่านระหว่างจุดต่อ

7. ระดับชั้นแอพพลิเคชั่น (Application Layer) ทำหน้าที่ในการให้บริการแก่ผู้ใช้ เช่น การโอนข้อมูล (file transfer) การส่งจดหมายอิเล็กทรอนิกส์ (electronic mail) การใช้เทอร์มินอลเสมือน (virtual terminal emulation) การเข้าถึงระบบข้อมูลที่ห่างไกล (remote access)

หน่วยงานที่ได้รับการยอมรับ ในการกำหนดมาตรฐานด้านระบบเครือข่ายเฉพาะบริเวณหน่วยงานหนึ่งคือ ไอทริพเพิลอี (IEEE : The Institute of Electrical and Electronics Engineers) ได้มีโครงการ 802 ซึ่งเป็นการกำหนดรหัสระบบเครือข่ายเฉพาะบริเวณมาตรฐานในระดับชั้นฟิสิคอล และ ระดับชั้นดาต้าลิงค์ของมาตรฐานโอเอสไอ คือ

มีเดียมแอ็กเซสคอนโทรล (MAC : Medium Access Control) โดยคอยควบคุมการเข้าถึงสื่อกลางที่ใช้ในการส่งข้อมูล

ชุดมาตรฐาน 802 มีดังนี้

802.1 Overview, Internetworking and System Management
802.2 Logical link control
802.3 Carrier Sense Multiple Access with Collision Detection (CSMA/CD)
802.4 Token Passing Bus
802.5 Token Passing Ring
802.6 Metropolitan Area Networks (MANs)
802.7 Advisory group for broadband transmission
802.8 Advisory group for fiber optics
802.9 Integrated voice and data LANs

อีเทอร์เน็ต (Ethernet) เป็นข้อกำหนดของเครือข่ายเบสแบนด์ (baseband) ที่เทียบเท่ากับมาตรฐาน 802.3 ซึ่งพัฒนามาพร้อม ๆ กับทีซีพี/ไอพี และเป็นเทคโนโลยีแรกของระบบเครือข่ายเฉพาะบริเวณที่ใช้ทีซีพี/ไอพี

โพรโตคอลทีซีพี/ไอพี

เป็นโพรโตคอลที่ถูกพัฒนาขึ้นโดย Advance Research Project Agency (ARPA) of Department of Defense (DoD) ได้รับการยอมรับให้เป็นมาตรฐานในระบบการสื่อสารข้อมูลสามารถใช้ติดต่อสื่อสารกับทุก ๆ ระบบเครือข่ายที่อยู่ในกลุ่มเดียวกัน ซึ่งอินเตอร์เน็ต (Internet) เป็นเครือข่ายที่สำคัญที่ใช้โพรโตคอลนี้

โพรโตคอลทีซีพี/ไอพีประกอบด้วยระดับชั้นเทียบกับระดับชั้นมาตรฐานโอเอสไอ ดังรูปที่ 2.2 และในแต่ละระดับชั้นประกอบด้วยโพรโตคอลต่าง ๆ ดังรูปที่ 2.3

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ISO-OSI Model		Internet Model	User Model
7 Application		Process / Applications	Network Applications
6 Presentation			
5 Session			
4 Transport		Transport	Network
3 Network	internet	Internet	
	convergence	Network	
	subnet		
2 Data Link		Link	Infrastructure
1 Physical		Physical	

รูปที่ 2.2 แสดงการเปรียบเทียบระหว่างโพรโตคอลทีซีพี/ไอพีกับระดับชั้นมาตรฐานโอเอสไอ

(Carl-Mitchell and Quarterman, 1993)

Layer	Protocols					
Process/ Application	TELNET (login)	FTP (files)	SMTP (mail)	DNS (names)	NTP (time)	NFS (files)
Transport	TCP			UDP		
Internet	IP					
Network	Ethernet	ISO 8802-2		X.25	SLIP	PPP
Link	(CSMA/CD)	ISO 8802-3 IEEE 802.3	ISO 8802-5 IEEE 802.5 (token ring)	ISO 7776 X.25 LAPB (HDLC)	(serial)	
Physical	(various)					

รูปที่ 2.3 แสดงโพรโตคอลต่างๆ ในแต่ละระดับชั้นของโพรโตคอลทีซีพี/ไอพี

(Carl-Mitchell and Quarterman, 1993)

โพรโตคอลทีซีพี/ไอพี ประกอบด้วยชุดของโพรโตคอลในระดับต่าง ๆ ที่สำคัญ คือ (Richard, 1991)

1. โพรโตคอลทีซีพี (TCP, Transmission Control Protocol)

เป็นโพรโตคอลที่เทียบเท่ากับโพรโตคอลในระดับชั้นทรานสปอร์ต ตามมาตรฐานโอเอสไอ ให้บริการแบบมีการเชื่อมต่อ (connection-oriented) คือ บริการที่มีการเชื่อมต่อกันตลอดเวลาในเชิงตรรกะ ให้ความเชื่อถือได้ในการรับส่งข้อมูล (reliable) มีการรับส่งข้อมูลไปและกลับได้ในเวลาเดียวกัน (full-duplex) และเป็นลักษณะของกระแสไบต์สำหรับมุมมองของขบวนการ (process) โปรแกรมประยุกต์ส่วนใหญ่ในอินเตอร์เน็ต จะใช้โพรโตคอลนี้ ซึ่ง ทีซีพี เองจะไปขอใช้บริการของโพรโตคอลไอพี อีกทีหนึ่ง

2. โพรโตคอลยูดีพี (UDP, User Datagram Protocol)

เป็นโพรโตคอลในระดับชั้นเดียวกับทีซีพี แต่ให้บริการที่แตกต่างกัน ยูดีพี ให้บริการแบบปราศจากการเชื่อมต่อ (connectionless) คือ ไม่มีการเชื่อมต่อกันตลอดเวลาในเชิงตรรกะ ดังนั้นจึงไม่ประกันว่าข้อมูลที่ส่งไปยังปลายทางจะถึงหรือไม่

3. โพรโตคอลไอซีเอ็มพี (ICMP, Internet Control Message Protocol)

เป็นโพรโตคอลที่ทำหน้าที่ดูแลจัดการ เกี่ยวกับข้อผิดพลาดหรือข่าวสารในการควบคุมการรับส่งข้อมูล ระหว่างเกตเวย์ (gateway) และแม่ข่าย โดยในการส่งข้อมูลข้อผิดพลาดต่าง ๆ เหล่านี้ ก็อาศัยบริการของโพรโตคอลไอพี เช่นกัน

4. โพรโตคอลไอพี (IP, Internet Protocol)

เป็นโพรโตคอลที่เทียบเท่ากับ โพรโตคอลระดับชั้น เน็ตเวอร์ค ตามมาตรฐานโอเอสไอ ให้บริการในการรับส่งกลุ่ม (packet) สำหรับทีซีพี, ยูดีพี และ ไอซีเอ็มพี

5. โพรโตคอลเออาร์พี (ARP, Address Resolution Protocol)

เป็นโพรโตคอลที่แปลงส่งอินเตอร์เน็ตแอดเดรส (Internet address) ไปเป็นฮาร์ดแวร์แอดเดรส (hardware address)

6. โพรโตคอลอาร์เออาร์พี (RARP, Reverse Address Resolution Protocol)

เป็นโพรโตคอลที่แปลงส่งฮาร์ดแวร์แอดเดรส ไปเป็นอินเตอร์เน็ตแอดเดรส ซึ่งทำหน้าที่ตรงกันข้ามกับโพรโตคอลเออาร์พี

7. โพรโตคอลเทลเน็ต (TELNET)

เป็นโพรโตคอลในระดับชั้นแอพพลิเคชั่นให้บริการในการเข้าถึงระบบข้อมูลที่ห่างไกล สามารถลงบันทึกเข้าสู่ระบบห่างไกลเชื่อมต่อเป็นเทอร์มินอลหนึ่งของ ระบบนั้นได้

8. โพรโตคอลเอฟทีพี (FTP, File Transfer Protocol)

เป็นโพรโตคอลในระดับชั้นแอพพลิเคชั่นเช่นกัน ให้บริการในการรับ-ส่ง แฟ้มข้อมูลระหว่างระบบหนึ่งไปยังอีกระบบหนึ่ง

นอกจากนี้ยังมีโพรโตคอลอื่น ๆ อีกซึ่งจะทำหน้าที่ต่าง ๆ เทียบเท่ากับหน้าที่ของระดับชั้นต่าง ๆ ตามมาตรฐานโอเอสไอ (Comer, 1991) ดังรูปที่ 2.3

โดยทั่วไปแล้วในการระบุถึงเครื่องคอมพิวเตอร์ปลายทางในระบบเครือข่ายจำเป็นที่จะต้องมีการระบุที่อยู่ (address) หรือเส้นทางที่จะส่งข้อมูลไปถึง โพรโตคอลทีซีพี/ไอพี มีระบบในการกำหนดที่อยู่ของแต่ละเส้นทางและเครื่องแม่ข่ายเรียกว่า อินเตอร์เน็ตแอดเดรส ซึ่งประกอบด้วยบิตจำนวน 32 บิต ซึ่งรวมเอา เน็ตเวอร์คไอดี (Network ID) และ โฮสไอดี

(Host ID) ไว้ด้วยกัน ทุก ๆ แม่ข่ายบน โพรโตคอลทีซีพี/ไอพี ในอินเตอร์เน็ต จะต้องมีอินเตอร์เน็ตแอดเดรสที่ไม่ซ้ำกัน ซึ่งอินเตอร์เน็ตแอดเดรสของเครื่องแม่ข่ายเหล่านี้ ถูกกำหนดโดย Network Information Center (NIC)

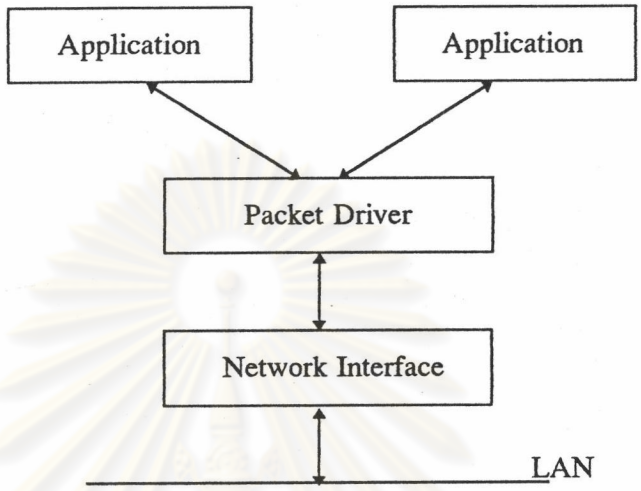
การนำเอาไมโครคอมพิวเตอร์มาดัดแปลงให้ทำงานเป็นเทอร์มินอลภายใต้ระบบเครือข่ายที่มีโพรโตคอลทีซีพี/ไอพี นั้น จำเป็นที่จะต้องมีอุปกรณ์สื่อสาร คือ เน็ตเวอร์คอินเตอร์เฟสการ์ด (Network Interface Card) เพิ่มเข้าไป ซึ่งอาจจะเป็นชนิดอีเทอร์เน็ต โทคเคนริง หรือชนิดอื่น ๆ ซึ่งเป็นการสื่อสารในระดับชั้นฟิสิคอล และมีโปรแกรมในการติดต่อกับการ์ดในระดับชั้นดาต้าลิงค์ซึ่งมีวิธีการที่ใช้กันอยู่แตกต่างกันถึง 4 ประเภท ด้วยกันคือ (Carl-Mitchell and Quarterman, 1993)

1. Packet Driver
2. Network Driver Interface Specification (NDIS)
3. Open Datalink Interface (ODI)
4. Adapter Support Interface (ASI)

1. แพกเก็ตไดร์เวอร์ (Packet Driver)

เป็นข้อกำหนดที่ใช้กันโดยทั่วไป สำหรับโปรแกรมประยุกต์ที่ใช้ โพรโตคอลทีซีพี/ไอพี โดยจะเป็นโปรแกรมประเภทฝังตัวในหน่วยความจำ (TSR : Terminate and Stay Resident) ในระบบปฏิบัติการดอส (DOS) ซึ่งโปรแกรมประยุกต์ต่าง ๆ จะสามารถขอใช้บริการจากแพกเก็ตไดร์เวอร์ร่วมกันโดยจะให้บริการรับและส่งเฟรม (frame) ข้อมูล ดังรูปที่ 2.4 (Carl-Mitchell and Quarterman, 1993)

แพกเก็ตไดร์เวอร์เป็นโปรแกรมที่หาได้ง่าย แพร่หลาย และสามารถสำเนาได้จากโดเมนสาธารณะ



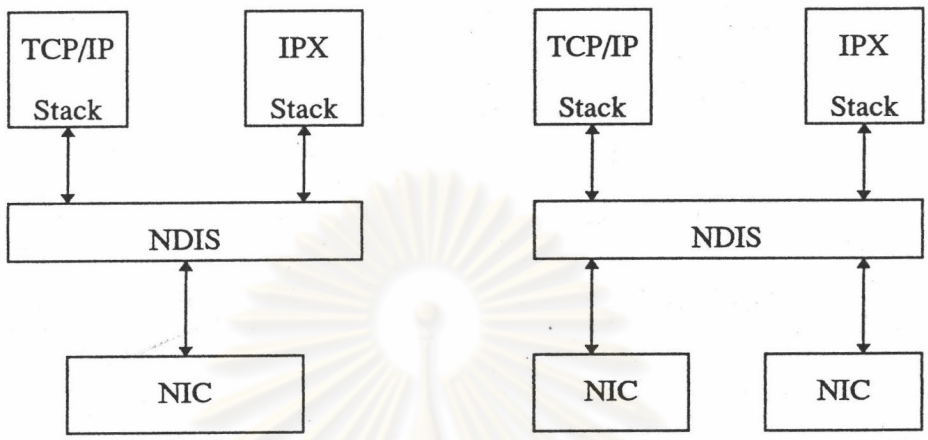
รูปที่ 2.4 แสดงตำแหน่งของแพกเก็ตไดร์เวอร์ ในระบบ

(Carl-Mitchell and Quarterman, 1993)

2. เอ็นดีไอเอส (NDIS, Network Driver Interface Specification)

เป็นข้อกำหนดของเน็ตเวอร์คไดร์เวอร์ (network driver) ที่ถูกกำหนดขึ้นโดยไมโครซอฟต์ (Microsoft) เพื่อให้สามารถรองรับโปรแกรมประยุกต์ หรือ กองซ้อนโพรโตคอล (protocol stack) ได้หลายๆ โพรโตคอล และหลายเน็ตเวอร์คอินเตอร์เฟสการ์ด ดังรูปที่ 2.5 (Sheldon, 1994)

ข้อกำหนดนี้ยอมให้สถานีงาน (workstation) มีเน็ตเวอร์คอินเตอร์เฟสการ์ดได้ถึง 4 การ์ด และแต่ละการ์ดสามารถรองรับได้ถึง 4 กองซ้อนโพรโตคอล

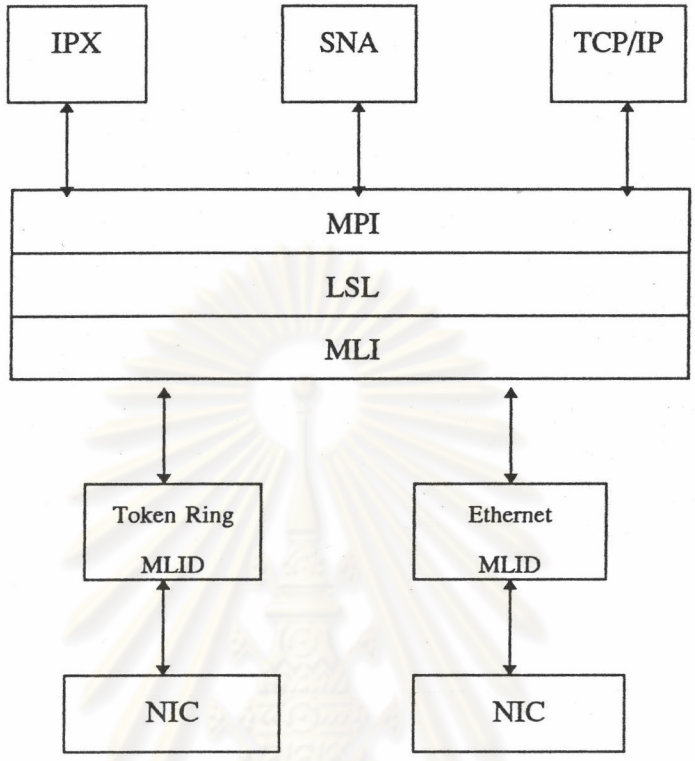


รูปที่ 2.5 แสดงโครงสร้างของเอ็นดีไอเอส (Sheldon, 1994)

3. โอดีไอ (ODI, Open Data-link Interface)

เป็นข้อกำหนดของเน็ตเวอร์คไดร์เวอร์ที่กำหนดโดย โนเวลล์ (Novell) เพื่อให้สามารถรองรับโปรแกรมประยุกต์หรือ กองซ้อนโพรโตคอล ได้หลายๆโพรโตคอล และหลายๆ เน็ตเวอร์คอินเตอร์เฟสการ์ด เช่นเดียวกับ เอ็นดีไอเอส ผู้ขายเน็ตเวอร์คอินเตอร์เฟสการ์ดไม่จำเป็นต้องเขียนตัวขับสำหรับกองซ้อนโพรโตคอลแต่ละชนิด เพียงแต่เขียนตัวขับให้สามารถติดต่อกับ แอลเอสแอล (LSL : Link Support Layer) ซึ่งเป็นส่วนหนึ่งของโอดีไอ (Sheldon, 1990)

โอดีไอประกอบด้วยองค์ประกอบ 3 ส่วนดังรูปที่ 2.6



รูปที่ 2.6 แสดงองค์ประกอบของโอดีไอ (Sheldon, 1990)

3.1 เอ็มแอลไอ (MLI, Multiple Link Interface)

เป็นชั้นสำหรับเชื่อมต่อกับตัวขับของเน็ตเวอร์คอินเตอร์เฟสการ์ด ซึ่งถูกเขียนมาให้สามารถเชื่อมต่อกับ แอลเอสแอล ได้ ตัวขับนี้จะหมายถึง Multiple Link Interface Drivers (MLIDs)

3.2 แอลเอสแอล (LSL, Link Support Layer)

แอลเอสแอลจะให้การเชื่อมต่อระหว่างตัวขับและ กองซ้อนโพรโตคอล ซึ่งมันจะทำหน้าที่เป็นเหมือนตัวสวิตซ์ทิศทางของ กลุ่มข้อมูลจาก MLID ไปยังโพรโตคอลที่

ถูกต้องหรือทำหน้าที่เป็นสะพานข้ามของกลุ่มข้อมูลจาก MLID หนึ่งไปยังอีก MLID หนึ่ง แอลเอสแอลนั้นสามารถรองรับโพรโตคอลได้ถึง 16 โพรโตคอล และ 8 MLID

3.3 เอ็มพีไอ (MPI, Multiple Protocol Interface)

ทำหน้าที่ให้การเชื่อมต่อกับกองซ้อนโพรโตคอล เช่น IPX, TCP/IP, OSI และ SNA เป็นต้น

4. เอเอสไอ (ASI, Adapter Support Interface)

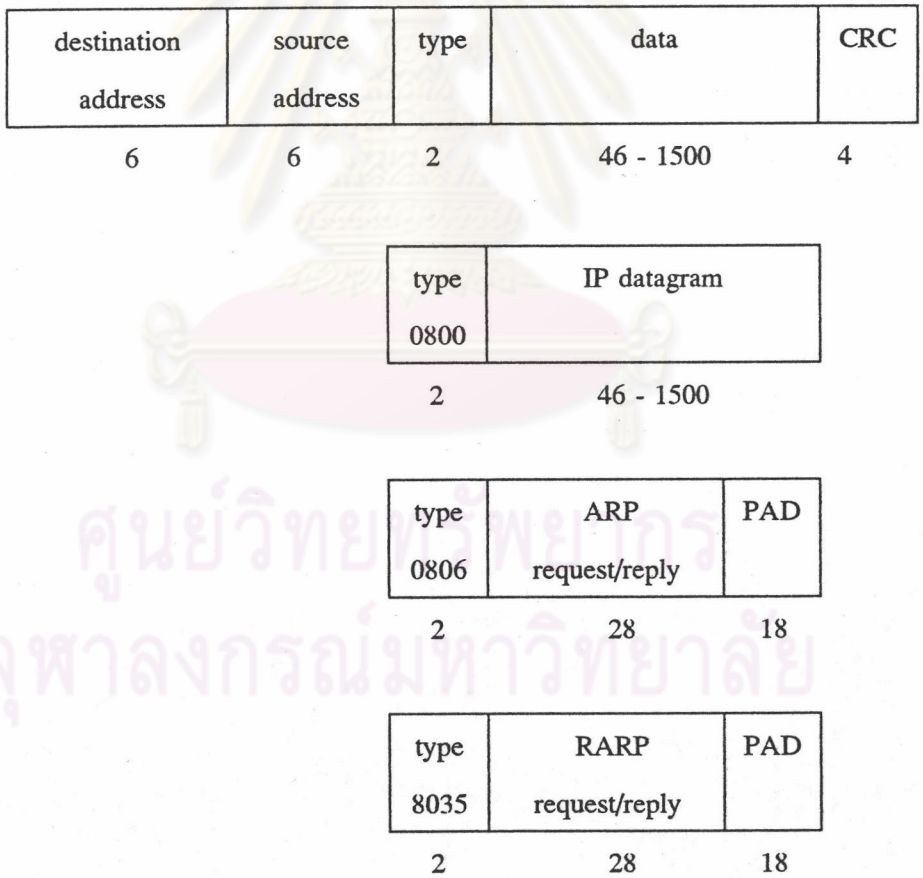
เป็นข้อกำหนดของไอบีเอ็ม (IBM) เพื่อรองรับกองซ้อนโพรโตคอล สำหรับเน็ตเวอร์คอินเตอร์เฟสการ์ดแบบโทคเคนริง ซึ่งมักจะใช้กับ AS/400, LAN Server และ Systems Network Architecture (SNA) (Stark, 1994)

จะเห็นได้ว่าวัตถุประสงค์ของวิธีการที่ใช้กันอยู่หรือข้อกำหนดที่กล่าวมานั้นมีความต้องการที่ตรงกัน คือ ต้องการให้หลายๆ โปรแกรมประยุกต์ หรือกองซ้อนโพรโตคอลต่างๆ สามารถเชื่อมต่อ และใช้บริการโพรโตคอลในระดับล่าง อย่างเป็นมาตรฐาน ไม่ต้องขึ้นกับเน็ตเวอร์คอินเตอร์เฟสการ์ดใดๆ แพกเก็ตไดร์เวอร์เป็นโปรแกรมที่หาได้ง่าย แพร่หลาย และสามารถสำเนาได้จากโดเมนสาธารณะ จึงมีความเหมาะสมที่จะนำมาทำการศึกษาวิจัยและใช้งาน สำหรับรายละเอียดจะกล่าวถึงต่อไป

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

แพกเก็ตไดร์เวอร์

แพกเก็ตไดร์เวอร์เป็นข้อกำหนดหนึ่งที่กำหนดโดย FTP Software Inc. เพื่อให้ง่ายต่อการเขียนโปรแกรมเชื่อมต่อ (programming interface) และเพื่อเป็นแบบฉบับ ซึ่งจะอนุญาตให้โปรแกรมประยุกต์หลาย ๆ โปรแกรมสามารถติดต่อในระดับดาต้าลิงค์ร่วมกันได้ แพกเก็ตไดร์เวอร์ จะเป็นตัวที่ทำการดีมัลติเพลกซ์ กลุ่ม (packet) ต่าง ๆ ที่ส่งมาจากโปรแกรมประยุกต์โดยใช้เขตที่ระบุชนิดของกลุ่ม (type) หรือจุดบริการเข้าถึง (service access point) ที่เป็นมาตรฐานของสื่อเครือข่าย (network media) นั้น ๆ (FTP Software Inc., 1989) ดังรูปที่ 2.7



รูปที่ 2.7 แสดงรูปแบบของกลุ่มของอีเทอร์เน็ต โดยแพกเก็ตไดร์เวอร์จะดีมัลติเพลกซ์โดยอาศัยเขตที่ระบุชนิดของกลุ่ม (Richard, 1994)

วัตถุประสงค์ของข้อกำหนดนี้คือ เพื่อให้การทำงานของกองซ้อนโพรโตคอล (protocol stack) ไม่ต้องขึ้นอยู่กับเน็ตเวอร์คอินเตอร์เฟสการ์ดของผู้ผลิตรายใด หรือรุ่นใดรุ่นหนึ่ง ดังนั้นโปรแกรมประยุกต์ต่างๆ ที่ใช้แพกเก็ตไดร์เวอร์จะสามารถทำงานบนอุปกรณ์ใหม่ ๆ ในเครือข่ายที่เป็นชั้น (class) เดียวกันได้โดยไม่ต้องทำการแก้ไขใด ๆ เพียงแต่เปลี่ยนแพกเก็ตไดร์เวอร์ตัวใหม่เท่านั้น ด้วยเหตุนี้ไม่ว่าโพรโตคอลในระดับเน็ตเวอร์คจะเป็น TCP/IP DECnet หรือ Novell ก็สามารถที่จะทำงานร่วมกับเน็ตเวอร์คอินเตอร์เฟสการ์ดได้โดยไม่ยาก

1. ระดับของแพกเก็ตไดร์เวอร์

แพกเก็ตไดร์เวอร์ สามารถแบ่งออกได้เป็น 3 ระดับ ดังนี้

1.1 ระดับเบสิก (Basic Packet Driver)

แพกเก็ตไดร์เวอร์ในระดับนี้จะมีฟังก์ชั่นต่าง ๆ ที่ไม่ยากนัก ซึ่งประกอบด้วยการกระจายและรับกลุ่ม ข้อดีคือ ง่ายต่อการพัฒนาและใช้ทรัพยากรน้อย

1.2 ระดับเอกซ์เทน (Extended Packet Driver)

แพกเก็ตไดร์เวอร์ในระดับนี้จะรวมเอาการทำงานของระดับเบสิกไว้ทั้งหมด แต่จะเพิ่มฟังก์ชั่นพิเศษต่าง ๆ เช่น การมัลติแคส (multicast), การยอมให้โปรแกรมประยุกต์สามารถตรวจสอบสถานะในการใช้เน็ตเวอร์คอินเตอร์เฟสการ์ด เป็นต้น

1.3 ระดับไฮเพอร์ฟอแมนซ์ฟังก์ชั่น (High-Performance Function)

แพกเก็ตไดร์เวอร์ในระดับนี้ จะรองรับการพัฒนาและปรับเปลี่ยนประสิทธิภาพให้เหมาะสม ซึ่งอาจจะมีหรือไม่มีฟังก์ชั่นในระดับเบสิกหรือระดับเอกซ์เทนก็ได้

2. การระบุถึงเน็ตเวอร์คอินเตอร์เฟสการ์ด

การระบุถึงเน็ตเวอร์คอินเตอร์เฟสการ์ดนั้นจะระบุโดยตัวเลข 3 ค่า คือ ชั้น (class) ชนิด (type) และ อินเตอร์เฟสนัมเบอร์ (interface number)

2.1 ชั้น จะระบุถึงชั้นของเน็ตเวอร์คอินเตอร์เฟส ซึ่งจะบอกถึงชนิดของสื่อที่เน็ตเวอร์คอินเตอร์เฟสรองรับ เช่น DEC/Intel/Xerox (DIX or Bluebook) Ethernet, IEEE 802.3 Ethernet, IEEE 802.5 Token Ring, ProNet-10, Appletalk, serial line และอื่น ๆ

2.2 ชนิด จะบอกถึงชนิดของเน็ตเวอร์คอินเตอร์เฟสการ์ดของชั้นนั้น ๆ เช่น Ethernet อาจจะมีชนิดเป็น 3Com, 3C503, 3C505, Interlan Ni5210, Univation, BICC Data Networks, Ungermann-Bass PC-NIC และอื่น ๆ ชนิดของเน็ตเวอร์คอินเตอร์เฟสการ์ดสำหรับ IEEE 802.5 อาจจะเป็น IBM Token Ring adapter, Proteon P1340 เป็นต้น

2.3 อินเตอร์เฟสนัมเบอร์ ในกรณีที่เครื่องคอมพิวเตอร์นั้น ประกอบด้วยหลาย ๆ เน็ตเวอร์คอินเตอร์เฟสที่เป็นชั้น และชนิดเดียวกัน ตัวเลขนี้จะเป็นตัวระบุว่าเป็นเน็ตเวอร์คอินเตอร์เฟสการ์ดตัวใด

รายละเอียดสำหรับชั้นและชนิดต่าง ๆ จะแสดงอยู่ในส่วนของภาคผนวก ก.

ชั้นของเน็ตเวอร์คอินเตอร์เฟสการ์ด จะเป็นจำนวนเต็มซึ่งมีขนาด 8 บิต ส่วนชนิดจะเป็นจำนวนเต็มขนาด 16 บิต FTP Software Inc. จะเป็นผู้ดูแลและจัดการกับค่าที่ระบุในชั้นและชนิด โดยถ้ามีชั้นและชนิดใหม่ ๆ เพิ่มขึ้นจะต้องแจ้งและลงทะเบียนตัวเลขค่านั้น ๆ กับ FTP Software Inc.

ชนิดที่มีค่า 0xFFFF จะเป็นชนิด วายคาร์ด (wildcard type) ซึ่งจะเข้ากับเน็ตเวอร์คอินเตอร์เฟสการ์ดใดๆ ก็ได้ในชั้นนั้นๆ ส่วนการให้ค่าของอินเตอร์เฟสนัมเบอร์เป็น

วายคาร์ดนั้นไม่มีความจำเป็น เพราะ 0 มักจะระบุถึงเน็ตเวอร์คอินเตอร์เฟสการ์ดตัวแรกของชั้นและชนิดนั้น ๆ

ข้อกำหนดนี้ มิได้กำหนดเพื่อไว้สำหรับการรองรับ เน็ตเวอร์คอินเตอร์เฟสการ์ดหลาย ๆ ตัว (multiple network interface) ที่มีลักษณะต่าง ๆ เหมือนหรือแตกต่างกันในลักษณะที่ใช้แพกเก็ตไดร์เวอร์ตัวเดียวควบคู่กับการใช้การขัดจังหวะ ซึ่งในประเด็นนี้จะสามารถแก้ปัญหาได้โดย การใช้แพกเก็ตไดร์เวอร์หลายๆ ตัว หนึ่งตัวต่อแต่ละเน็ตเวอร์คอินเตอร์เฟสการ์ด และใช้การขัดจังหวะที่แตกต่างกัน แพกเก็ตไดร์เวอร์ถูกเรียกโดยใช้การขัดจังหวะในช่วง 0x60 กับ 0x80 ซึ่งจะไม่เจาะจงว่าจะต้องเป็นการขัดจังหวะใด

3. การเขียนโปรแกรมเชื่อมต่อ

โดยปกติโปรแกรมประยุกต์ที่เรียกใช้ แพกเก็ตไดร์เวอร์ มักจะเขียนด้วยไมโครซอฟต์ซี (Microsoft C) และ ภาษาแอสเซมบลี ทุก ๆ ฟังก์ชั่นจะทำการเข้าถึงโดยใช้การขัดจังหวะทางซอฟต์แวร์ ในการเรียกใช้นั้น เรจิสเตอร์ AH จะเก็บรหัสของฟังก์ชั่นที่ต้องการ

โปรแกรมเลียนแบบเทอร์มินอล

โปรแกรมเลียนแบบเทอร์มินอล คือโปรแกรมที่ทำหน้าที่ในการเลียนแบบการทำงานของเทอร์มินอล โดยทำหน้าที่ในการรับตัวอักขระหรือข้อมูล ที่ส่งมาจากเครื่องคอมพิวเตอร์แม่ข่าย และนำมาแสดงผลหรือประมวลผลไปยังจอภาพ ในขณะเดียวกันจะต้องรอรับอักขระจากแผงแป้นอักขระ ที่ผู้ใช้ป้อนเข้ามาและส่งไปให้กับเครื่องคอมพิวเตอร์แม่ข่าย

ในสมัยก่อนเทอร์มินอลจะเป็นลักษณะของเครื่องพิมพ์ดีดที่มีม้วนกระดาษสำหรับพิมพ์อักขระให้ปรากฏ ซึ่งม้วนกระดาษก็จะเลื่อนขึ้นไปเรื่อย ๆ เมื่ออักขระถูกพิมพ์จนหมดบรรทัด เรียกว่า เทเลไทน์ (teletype) ต่อมามีการพัฒนาเทอร์มินอลให้เป็นลักษณะของจอภาพและแผงแป้นอักขระ เมื่ออักขระแสดงผลจนเต็มจอภาพ ก็จะมีการเลื่อนของ

บรรทัดอักขระที่อยู่บนสุดหายไปจากจอ เป็นลักษณะของการเลื่อนขึ้น ซึ่งการทำงานในลักษณะดังกล่าวอาศัยการควบคุมซึ่งเป็นมาตรฐาน โดยจะมีอักขระในการควบคุมการทำงานดังกล่าว เรียกว่า อักขระควบคุม (control character) หรือ เอสเค็ปซีเควนส์ (escape sequences) ดังนั้นเมื่อเทอร์มินอลได้รับอักขระควบคุม หรือ เอสเค็ปซีเควนส์ ก็จะปฏิบัติตามการควบคุมดังกล่าว เช่น ลบจอภาพ, เลื่อนบรรทัดขึ้น เป็นต้น อักขระควบคุม ดังกล่าวจะเป็นมาตรฐานสำหรับเทอร์มินอลแต่ละชนิด เช่น VT100, VT220 หรือ ANSI เป็นต้น ดังนั้นหน้าที่สำคัญประการหนึ่งที่โปรแกรมเลียนแบบเทอร์มินอลจะต้องประมวลผลคือ จะต้องปฏิบัติตามอักขระควบคุมดังกล่าว (Freed and Derfler, 1992)

อักขระควบคุมสำหรับเทอร์มินอลชนิด ANSI ซึ่งใกล้เคียงกับ VT100 มีรูปแบบดังนี้

<ESC>[<n>;<m>;...<SPACCHAR>

คือเริ่มต้นด้วยอักขระ "ESCAPE" (0x1B) และตามด้วย LEFT BRACKET (0x5B) และตามด้วยตัวเลขซึ่งโดยมากจะหมายถึง บรรทัดที่ และ สดมภ์ที่ โดยมีเครื่องหมาย semicolon เป็นตัวคั่น ส่วนตัวอักขระสุดท้ายจะหมายถึง คำสั่งของอักขระควบคุม

อักขระควบคุมของ เทอร์มินอล ANSI ที่สำคัญประกอบด้วย

<ESC> [<n> A ย้ายตัวชี้ตำแหน่งขึ้น <n> บรรทัด ถ้าไม่ใส่ <n> จะหมายถึง 1 บรรทัด

<ESC> [<n> B ย้ายตัวชี้ตำแหน่งลง <n> บรรทัด

<ESC> [<n> C ย้ายตัวชี้ตำแหน่งไปทางขวา <n> สดมภ์

<ESC> [<n> D ย้ายตัวชี้ตำแหน่งไปทางซ้าย <n> สดมภ์

<ESC> [<n> E ย้ายตัวชี้ตำแหน่งไปที่สดมภ์แรกและลง <n> บรรทัด

<ESC> [<n> F ย้ายตัวชี้ตำแหน่งไปยังสดมภ์แรก

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

นอกจากอักขระควบคุมดังกล่าวแล้วนี้ ยังมีอักขระควบคุมอีกมาก (Biggerstaff., 1986)

โมเด็ม

เพื่อให้การใช้งานคอมพิวเตอร์ที่มีประสิทธิภาพมากขึ้น การทำงานในลักษณะของเทอร์มินอลทางไกล (remote terminal) เป็นสิ่งจำเป็นอย่างหนึ่งในการใช้งานระบบคอมพิวเตอร์ ดังนั้นจึงจำเป็นที่จะต้องมีอุปกรณ์ในการติดต่อสื่อสารทางไกล เพื่อใช้ในการติดต่อสื่อสารระหว่างเทอร์มินอลกับเครื่องคอมพิวเตอร์แม่ข่าย

โมเด็ม เป็นอุปกรณ์สื่อสารข้อมูลคอมพิวเตอร์ ที่ทำหน้าที่ในการแปลงสัญญาณดิจิตอล ให้เป็นสัญญาณอนาลอกหรือ สัญญาณเสียง ส่งไปในสายโทรศัพท์และทำการแปลงสัญญาณอนาลอกกลับมาเป็นสัญญาณดิจิตอล

1. มาตรฐานของโมเด็ม

โมเด็มมีการกำหนดมาตรฐานเพื่อให้สามารถติดต่อกันได้และ เพื่อประสิทธิภาพในการรับส่งข้อมูล มาตรฐานที่สำคัญ คือ

1.1 Bell 103 เป็นโมเด็มมาตรฐานแรกที่สำคัญในโลกของคอมพิวเตอร์ ซึ่งสามารถรับส่งข้อมูลด้วยอัตราส่งในระหว่าง 110 ถึง 300 bps โดยอาศัยการต่อเข้ากับสายโทรศัพท์โดย acoustic coupler ในลักษณะของ full-duplex และมีการส่งแบบ asynchronous

1.2 Bell 212, V.22 เป็นมาตรฐานที่ส่งข้อมูลได้เร็วขึ้นในถึง 1,200 bps และ สามารถลดระดับความเร็วในการส่ง (fall back) เพื่อรับข้อมูล 300 bps ในมาตรฐาน Bell 103 ได้ และใช้มาตรฐานอาร์เอส 232 ซี ในการเชื่อมต่อกับโมเด็ม

1.3 V.22 bis เป็นมาตรฐานที่ส่งข้อมูลได้ที่ความเร็ว 2,400 bps ซึ่งส่งข้อมูลเร็วเป็น 2 เท่าของ V.22 มีการใช้อย่างแพร่หลายมาก มีการเพิ่มความสามารถในการควบคุมโมเด็มด้วยคำสั่ง AT ซึ่งพัฒนาโดย เฮยส์ (Hayes)

1.4 V.32 สามารถส่งข้อมูลที่ความเร็ว 9,600 bps ซึ่งผลิตโดย U.S. Robotics และ Telebit

1.5 V.32 bis เป็นการพัฒนาให้โมเด็มมีความฉลาดขึ้นอีก สามารถส่งข้อมูลด้วยความเร็ว 9,600 bps จนถึง 14,400 bps

1.6 V.42 เป็นมาตรฐานที่กำหนดโดย CCITT สำหรับการควบคุมความผิดพลาดในการรับส่งข้อมูล (error-control) ของโมเด็มให้มีประสิทธิภาพสูงขึ้นในด้านความถูกต้องของข้อมูลในขณะที่รับส่งด้วยความเร็วสูง ๆ

1.7 V.42 bis เป็นมาตรฐานที่กำหนดโดย CCITT ในการลดขนาดของข้อมูลของโมเด็มเพื่อให้สามารถส่งข้อมูลได้มากขึ้น ส่งผลให้อัตราการส่งข้อมูลสูงขึ้นอีก

1.8 MNP class 2,3,4 เป็นมาตรฐานสำหรับการควบคุมความผิดพลาดในการรับส่งข้อมูลของโมเด็มซึ่งกำหนดโดย Microcom Inc.

1.9 MNP class 5 เป็นมาตรฐานการลดขนาดของข้อมูลของโมเด็มซึ่งกำหนดโดย Microcom Inc.

2. การควบคุมโมเด็มโดยใช้คำสั่ง AT

โมเด็มในระยะแรก ๆ นั้น เป็นแต่เพียงอุปกรณ์การสื่อสารที่ทำหน้าที่แปลงสัญญาณและส่งข้อมูลไปมาเท่านั้น ไม่มีความสามารถใด ๆ เรียกว่า Dumb modem ต่อมามีการเพิ่มความสามารถให้กับโมเด็ม ให้สามารถทำงานอย่างอื่นให้มากขึ้น เช่น สามารถ

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

หมุนโทรศัพท์ได้ รับโทรศัพท์ได้เป็นต้น มีการสั่งการและควบคุมโมเด็มโดยคำสั่ง และเป็นมาตรฐานที่สำคัญคือ คำสั่ง AT ซึ่งถูกออกแบบโดย เฮยส์ ทำให้โมเด็มมีความฉลาดเพิ่มมากขึ้น ตัวอย่างของคำสั่ง AT เช่น

ATH	เป็นคำสั่งในการเลิกการติดต่อและวางสายโทรศัพท์
ATD	เป็นคำสั่งในการหมุนโทรศัพท์
ATM	เป็นคำสั่งในการควบคุมลำโพงภายในโมเด็ม
ATA	เป็นคำสั่งให้โมเด็มรับโทรศัพท์อัตโนมัติ
ATL	เป็นคำสั่งในการเพิ่มลดเสียงของลำโพงในโมเด็ม

นอกจากคำสั่งดังกล่าวนี้แล้วยังมีคำสั่งอีกมากในการควบคุม (Freed and Derfler, 1992) ซึ่งทำให้การควบคุมโมเด็มสะดวกและง่ายขึ้น โดยโปรแกรมประยุกต์เพียงแต่ส่งคำสั่งเหล่านี้ให้แก่โมเด็ม โดยผ่านมาตรฐาน อาร์เอส 232 ซี โมเด็มก็จะปฏิบัติงานตามคำสั่งดังกล่าว

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย