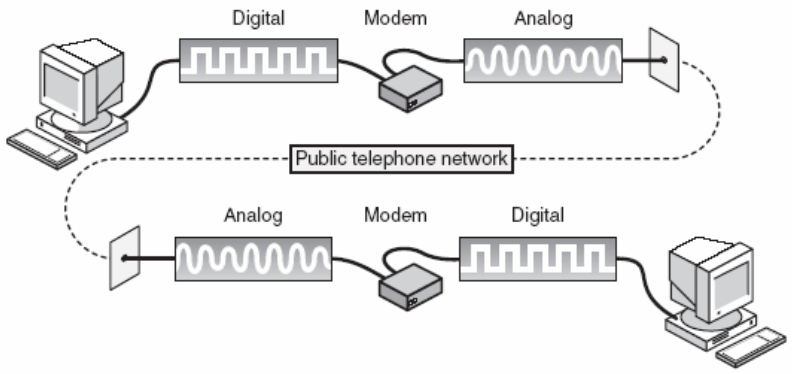
บทที่ 10
การขยายการเชื่อมต่อระบบเครือข่ายคอมพิวเตอร์

ในบทนี้จะอธิบายถึงการนำอุปกรณ์ระบบเครือข่ายต่างๆ มาใช้ในการจัดตั้งระบบเครือข่ายคอมพิวเตอร์ รวมถึงการขยายการเชื่อมต่อเพื่อเพิ่มขนาดของระบบเครือข่าย โดยอุปกรณ์พื้นฐานที่สำคัญคือโมเด็ม จากนั้นจะอธิบายการใช้งานอุปกรณ์ระบบเครือข่ายอื่นๆ เช่น สวิตช์ บริดจ์ เราท์เตอร์ และเกตเวย์ และสุดท้ายจะมีรายละเอียดของบริการต่างๆ สำหรับการขยายการเชื่อมต่อระบบเครือข่ายในพื้นที่กว้าง (**Wide Area Network**) หรือ **WAN**

10.1 เทคโนโลยีของโมเด็ม

ในกรณีที่เครื่องคอมพิวเตอร์ลูกข่ายมีที่ตั้งทางภูมิศาสตร์อยู่ห่างจากเครื่องเซิร์ฟเวอร์เป็นระยะทางไกลมาก จนกระทั่งสายสื่อสารสัญญาณ ที่อธิบายในบทที่ 2 ไม่สามารถเชื่อมต่อถึงกันได้ จึงจำเป็นต้องใช้เครือข่ายสายโทรศัพท์สาธารณะที่มีอยู่ในการขยายการเชื่อมต่อระบบเครือข่าย อย่างไรก็ตามเครื่องคอมพิวเตอร์ไม่สามารถที่จะทำการสื่อสารระหว่างกันโดยตรงผ่านทางสายโทรศัพท์พื้นฐานได้ เนื่องจากสัญญาณที่ถูกส่งโดยเครื่องคอมพิวเตอร์จะอยู่ในรูปแบบดิจิตอล (**Digital**) แต่สายโทรศัพท์พื้นฐานได้รับการออกแบบมาให้ทำการส่งเฉพาะสัญญาณอนาล็อก (**Analog**) ดังนั้นจึงจำเป็นต้องมีอุปกรณ์ที่เรียกว่า "โมเด็ม" เป็นตัวกลางในการแปลงสัญญาณ ดังแสดงตามรูปที่ **10 – 1**



รูปที่ **10 – 1** การใช้โมเด็มแปลงสัญญาณก่อนส่งออกไปบนเครือข่ายสายโทรศัพท์

โดยทั่วไปการเชื่อมต่อระหว่างเครื่องคอมพิวเตอร์กับโมเด็ม จะใช้พอร์ตสื่อสารมาตรฐาน **RS-232** เป็นตัวกำหนดการเชื่อมต่อ ใน **RS-232C lingo** จะเรียกโมเด็มว่าเป็นอุปกรณ์ **DCE (Data Communications Equipment)** และเรียกเครื่องคอมพิวเตอร์ว่าเป็นอุปกรณ์ **DTE (Data Terminal Equipment) RS-232** ซึ่งกำหนดการเชื่อมต่อใน **physical layer** นี้เป็นมาตรฐาน **de facto** สำหรับการสื่อสารของโมเด็ม และเครื่องพิมพ์มาเป็นเวลานานแล้ว ซึ่งในภายหลังได้รับการขยายให้เป็นมาตรฐาน **RS-232D** ซึ่งเพิ่มคุณสมบัติการตรวจสอบการส่งสัญญาณระหว่างโมเด็ม จึงเป็นหัวต่อเชื่อมประเภทพิเศษ สำหรับ **RS-232** ที่เรียกได้ว่าเป็นมาตรฐานได้รับการปรับปรุงให้ดีขึ้นโดย **EIA** และเทียบได้กับคุณลักษณะเฉพาะของโมเด็ม **V.24** และ **V.28** ของ **ITU**

10.1.1 มาตรฐานของโมเด็ม

ในปี **1980** บริษัท **Hayes Microcomputer Product** ได้พัฒนาโมเด็มสำเร็จเป็นรายแรก และใช้เป็นมาตรฐานในการสื่อสารของเครื่องคอมพิวเตอร์ผ่านเครือข่ายสายโทรศัพท์สาธารณะ จนกระทั่งผู้ผลิตรายอื่นใช้คำว่า **Hayes Compatible** ในการผลิตโมเด็มภายใต้มาตรฐานของ **Hayes** จึงทำให้ระบบเครือข่าย **LAN** ทั้งหมดสามารถสื่อสารระหว่างกันได้ ต่อมา **ITU (International Telecommunication Union)** จึงได้กำหนดมาตรฐานสากลของโมเด็มที่เรียกว่า "**V-Series**" ตามด้วยตัวเลขที่บ่งบอกมาตรฐานเหล่านั้น เช่น **V.22bis** หมายถึงโมเด็มที่มีความเร็วในการส่งข้อมูล **2400** บิตต่อวินาที และใช้เวลา **18** วินาที ในการส่งข้อมูลขนาด **1000** เวิร์ด ส่วน **V.42bis** จะมีการบีบอัดข้อมูล เพื่อให้ทำการส่งข้อมูลขนาดเดียวกันนี้ได้ในเวลาเพียง **3** วินาที ตารางที่ **10 – 1** แสดงมาตรฐานการบีบอัดข้อมูลใน "**V-Series**" เพื่อเพิ่มประสิทธิภาพในการทำงานของโมเด็ม

ตารางที่ **10 – 1** วิวัฒนาการของมาตรฐานโมเด็ม

มาตรฐาน	ความเร็ว (bps)	ปี ค.ศ.	คำอธิบาย
V.21	**1200**	**1980**	ส่งสัญญาณแบบ **Duplex V.21**
V.22bis	**2400**	**1982**	ส่งสัญญาณแบบ **Duplex V.22bis**
V.26bis	**1200/2400**	**1983**	ส่งสัญญาณแบบ **Half-Duplex**
V.26ter	**1200/2400**	**1983**	ส่งสัญญาณแบบ **Full-Duplex**
V.27ter	**2400/4800**	**1984**	ส่งสัญญาณแบบ **Full-Duplex**
V.32	**9600**	**1984**	ส่งสัญญาณแบบ **Full-Duplex**
V.32bis	**14400**	**1991**	ส่งสัญญาณแบบ **Full-Duplex**
V.32 turbo	**19200**	**1993**	ติดต่อสื่อสารกับโมเด็ม **V.32 turbo** เท่านั้น
V.FastClass	**28800**	**1993**	ส่งสัญญาณแบบ **Full-Duplex**
V.34	**28800**	**1994**	ปรับปรุง **V.FC** ให้เข้ากันได้กับโมเด็มทุกรุ่น
V.42	**57600**	**1995**	มีมาตรฐานการตรวจสอบความผิดพลาดของข้อมูล
V.90	**56600**	**1998**	เป็นมาตรฐานสำหนับโมเด็ม **56K**

หมายเหตุ คำว่า **bis** หรือ **ter** ที่ตามหลังข้อเสนอแนะบางข้อ แสดงให้เห็นจำนวนที่ทำการแก้ไขปรับปรุง **(bis)** หรือการแก้ไขปรับปรุงจากข้อเสนอแนะเดิม **(ter)** โดยคำทั้งคู่นี้เป็นภาษาฝรั่งเศส ของคำว่า **second (bis)** และ **third (ter)**

10.1.2 ประสิทธิภาพของโมเด็ม

หน่วยวัดความเร็วของโมเด็มคือ บิตต่อวินาที **(Bit per second – bps)** และ **Baud Rate** ซึ่งอ้างถึงความเร็วในการส่งคลื่นเสียงที่นำพาข้อมูลที่ต้องการส่งไปตามสายโทรศัพท์ หน่วยวัดความเร็วเป็น **Baud** นี้ได้มาจากชื่อของวิศวกรสื่อสารทางโทรเลขชาวฝรั่งเศส ชื่อ **Jean Maurice Email Baudot** ซึ่งในขณะนั้นหน่วยวัด **Baud Rate** ยังมีค่าเท่ากับความเร็วในการส่งข้อมูลเป็นบิตต่อวินาที แต่ต่อมาเมื่อเริ่มมีการนำเทคโนโลยีการบีบอัดข้อมูลมาใช้ทำให้การรวมสัญญาณ **(Modulate)** แต่ละครั้ง สามารถนำพาข้อมูลไปได้มากกว่า **1** บิต ตัวเลขของหน่วยวัดความเร็วทั้งสองนี้จึงไม่เท่ากัน เช่นโมเด็มที่มีความเร็ว **28,800 Baud** อาจจะสามารถทำการส่งข้อมูลได้เร็วถึง **115,200 bps**

10.2 ชนิดของโมเด็ม

10.2.1 โมเด็ม 56 Kbps

ในปัจจุบันโมเด็มสำหรับเครื่องคอมพิวเตอร์ส่วนบุคคลซึ่งมีความเร็วมากที่สุด ได้รับการออกแบบให้ทำงานบนสายโทรศัพท์แบบอนาล็อก มีความสามารถในการรับข้อมูลข่าวสารได้ด้วยความเร็ว **56 Kbps** ถึงแม้ว่าโดยพื้นฐานของสายโทรศัพท์จะสามารถรองรับการส่งสัญญาณได้ด้วยความเร็วเพียง **33.6 Kbps**

ในอดีตที่ผ่านมา **2-3** ทศวรรษ บริษัทผู้ให้บริการโทรศัพท์ได้มีการเปลี่ยนแปลงมาใช้เทคโนโลยีดิจิตอลเท่าที่จะเป็นไปได้ โดยเปลี่ยนแปลงโครงสร้างพื้นฐานการแลกเปลี่ยนและระบบ **switching** ของเครือข่ายโทรศัพท์ ทำให้เครือข่ายโทรศัพท์เกือบทั้งหมดที่มีในปัจจุบันเป็นระบบดิจิตอล และสถานที่ซึ่งใช้สายทองแดงเป็นมาตรฐานในส่วนของเครือข่ายโทรศัพท์ ก็จะเรียกว่า **analog local loop** – สายโทรศัพท์ที่ขึงจากสำนักงานศูนย์กลางของบริษัทโทรศัพท์ไปยังบ้าน

เนื่องจากการเชื่อมต่อภายในบริษัทผู้ให้บริการโทรศัพท์เป็นดิจิตอล โมเด็ม **56 Kbps** จึงทำงานผ่านปลายทางการเชื่อมโยง **56 Kbps** (ที่ซึ่งใช้เทคโนโลยี **56 Kbps** ในการสนับสนุนผู้ส่งและผู้รับ) อาจจะขึ้นอยู่กับกระแสของข้อมูลที่ได้รับจากเครื่องเซิร์ฟเวอร์ของระบบเครือข่าย (เครื่องเซิร์ฟเวอร์บนเครือข่ายอินเตอร์เน็ต) ซึ่งเดินทางผ่านเครือข่ายโทรศัพท์ในรูปแบบดิจิตอลจริงๆ นั่นคือข้อมูลจะต้องการการแปลงจากสัญญาณอนาล็อกเป็นดิจิตอลเพียงในการเดินทางจากเครื่องเซิร์ฟเวอร์ไปยังผู้รับ แม้ว่าจะมีการเคลื่อนที่ผ่านเครือข่ายโทรศัพท์ เพียงเมื่อสัญญาณข้อมูลออกจากบริษัทผู้ให้บริการโทรศัพท์ จึงจะมีการแปลงจากสัญญาณดิจิตอลเป็นอนาล็อก และจะทำการแปลงสัญญาณโดยสมบูรณ์ นั่นคือสัญญาณดิจิตอลที่แปลงไปเป็นสัญญาณอนาล็อกจะต้องประกอบด้วยค่าที่ไม่มีขีดจำกัด ซึ่งการแปลงสัญญาณข้อมูลด้วยวิธีนี้จะไม่ทำให้ความเร็วในการส่งลดลง และการทำในลักษณะนี้จะทำให้สัญญาณข้อมูลเดินทางผ่านการเชื่อมต่อของบริษัทผู้ให้บริการโทรศัพท์ไปยังเครื่องคอมพิวเตอร์แบบตั้งโต๊ะของผู้ใช้ ที่ซึ่งมีการแปลงสัญญาณจากอนาล็อกเป็นดิจิตอลเกิดขึ้น

อย่างไรก็ตาม เนื่องจากธรรมชาติของการส่งสัญญาณด้วยความเร็ว **56 Kbps** ที่ความเร็วขนาดนี้จะเกิดขึ้นเพียงทิศทางเดียว คือในขาลง (**downstream**) นั่นคือการไหลของข้อมูลจากเครือข่ายของบริษัทผู้ให้บริการโทรศัพท์ไปยังโมเด็มจะมีความเร็ว **56 Kbps** หากสมมติว่ามีการใช้เทคโนโลยี **56 Kbps** ตลอดทั้งการเชื่อมโยง ในทางกลับกับการไหลของข้อมูลในขาขึ้น (**upstream**) จากโมเด็มไปยังเครือข่ายโทรศัพท์ก็ไม่สามารถที่จะมีความเร็วได้มากกว่า **33.6 Kbps** ทำไม? เนื่องจากว่าการไหลของข้อมูลในขาขึ้นจะมีการแปลงข้อมูลจากดิจิตอลเป็นอนาล็อกมากกว่า **1** ครั้ง และการแปลงข้อมูลเหล่านี้เป็นสาเหตุให้ส่งผลกระทบที่เรียกว่า **quantization noise** ที่ลดประสิทธิภาพในการแปลงข้อมูล ทำให้ลดความเร็วในการส่งสัญญาณข้อมูลด้วย เทคโนโลยี **56 Kbps** จะมีประสิทธิภาพ ตราบเท่าที่

- มีเทคโนโลยี **56 Kbps** ใช้ตลอดเส้นทางการขนส่งข้อมูล
- การส่งสัญญาณเกี่ยวข้องกับการแปลงสัญญาณอนาล็อกเป็นดิจิตอลเพียงครั้งเดียว
- การเชื่อมต่อไปยังเครื่องเซิร์ฟเวอร์เป็นดิจิตอล
- ไม่มีสัญญาณรบกวนในสายโทรศัพท์

10.2.2 เคเบิลโมเด็ม (Cable Modems)

เคเบิลโมเด็มไม่ต้องอาศัยเครือข่ายโทรศัพท์โดยสิ้นเชิง จึงแตกต่างจากโมเด็มมาตรฐานและโมเด็มตระกูลต่างๆ ที่ทำความเร็วได้สูงถึง **56 Kbps** แต่จะใช้เครือข่ายของระบบเคเบิลทีวีในการถ่ายโอนข้อมูล จึงทำให้สามารถดาวน์โหลดข้อมูลได้ด้วยความเร็วสูงประมาณ **10 Mbps** ถึง **36 Mbps** ซึ่งเร็วกว่าโมเด็มที่ใช้เครือข่ายสายโทรศัพท์มาก โดยทั่วไปเคเบิลโมเด็มจะเชื่อมต่อเครื่องคอมพิวเตอร์ **PCs** เข้ากับปลั๊กของระบบเคเบิลทีวีที่ผนัง ถึงแม้ว่าจะเป็นโมเด็มที่ต้องทำการ **modulate** และ **demodulate** สัญญาณข้อมูล แต่เคเบิลโมเด็มก็ยังใช้อุปกรณ์อย่างอื่นประกอบด้วย ซึ่งรวมถึงการ์ดเชื่อมต่อระบบเครือข่าย ที่ต่อเชื่อมเข้ากับระบบเครือข่ายอีเธอร์เน็ต **10BaseT** ซึ่งติดตั้งอยู่ภายในเครื่องคอมพิวเตอร์ด้วย เคเบิลโมเด็มสามารถส่งสัญญาณในขาลงได้เร็วกว่าขาขึ้น เช่นเดียวกับโมเด็ม **56 Kbps** และถึงแม้ความเร็วในการส่งสัญญาณข้อมูลในขาลงจะสูงถึง **36 Mbps** แต่ดังที่กล่าวมาแล้วว่าความเร็วในการส่งสัญญาณข้อมูลในขาขึ้นจะช้ากว่า ซึ่งอาจเป็นไปได้ที่จะเร็วถึง **10 Mbps** แต่น่าจะทำความเร็วได้เพียง **2 Mbps** สำหรับการเชื่อมต่อ เคเบิลโมเด็มเพื่อใช้งาน จะเกี่ยวข้องกับ

- เครื่องคอมพิวเตอร์ **PCs** ที่มีการ์ดอีเธอร์เน็ต และเชื่อมต่ออยู่กับเคเบิลโมเด็ม
- สายเชื่อมต่อจากเคเบิลโมเด็มไปยังปลั๊กของระบบเคเบิลทีวีที่ผนัง
- สายภายใน (**drop cable**) ที่ต่อไปยัง **feeder cable** หรือกล่าวได้ว่าเชื่อมต่อกับสายเคเบิล **trunk**
- การควบคุมที่ส่วนควบคุมต้นทางของบริษัทซึ่งให้บริการเคเบิลทีวี ในการส่งสัญญาณออกไปตามสายเคเบิลและปรับแต่งการส่งสัญญาณข้อมูลที่มาจากเคเบิลโมเด็ม

ระบบเคเบิลทีวีใช้เทคโนโลยีการออกอากาศแบบ **broadband** ดังนั้น เทคโนโลยีที่ใช้จะต้องอยู่บนพื้นฐานของระบบเคเบิลทีวี ซึ่งมีการทำงานในย่านความถี่ **40 MHz** ถึง **550 MHz** และแบ่งออกเป็นช่องสัญญาณขนาด **6 MHz** ซึ่งแต่ละช่องจะใช้สำหรับ **1** ช่องโทรทัศน์ ในการส่งสัญญาณข้อมูลขาลง ส่วนควบคุมต้นทางจะ **modulate** สัญญาณข้อมูลและใส่เข้าไปในช่องสัญญาณ **6 MHz** ที่ไม่ได้ใช้ มีการนำวิธี **modulation** รูปแบบต่างๆ มาใช้ แต่ตามปกติจะใช้เทคนิคที่เรียกว่า **Quadrature Phase Shift Keying (QPSK)** ซึ่งทำให้มีการส่งสัญญาณข้อมูลด้วยความเร็วถึง **10 Mbps** และ **64QAM** ที่เร็วกว่าแต่มีความอ่อนไหวต่อสัญญาณรบกวนมากกว่า (มีกลุ่มของสัญญาณ **64** จุด) ซึ่งทำให้มีการส่งสัญญาณข้อมูลด้วยความเร็วถึง **36 Mbps** อย่างไรก็ตามก็จะทำการ **modulate** สัญญาณจากนั้นสัญญาณก็จะเดินทางผ่านสายเคเบิล **trunk, feeder** และ **drop cable** จนกระทั่งมาถึงเคเบิลโมเด็มซึ่งจะทำการ **demodulate** สัญญาณและส่งผ่านการ์ดเชื่อมต่อระบบเครือข่ายอีเธอร์เน็ต **10BaseT** เข้าไปยังเครื่องคอมพิวเตอร์ ในทางกลับกันในการส่งสัญญาณข้อมูลขาขึ้นเคเบิลโมเด็มก็จะรับคำสั่งจากส่วนควบคุมต้นทาง ซึ่งจะบอกว่าจะทำการส่งเมื่อใดและเป็นเวลานานเท่าใดและใช้ช่องความถี่ใดในการส่ง

10.2.3 โมเด็ม ISDN

ISDN หมายถึง **Integrated Services Digital Network** เป็นเทคโนโลยีการสื่อสารในระบบดิจิตอลซึ่งจะอธิบายในรายละเอียดต่อไปภายหลัง อย่างไรก็ตามในส่วนที่เกี่ยวกับโมเด็ม อย่างน้อยเรื่องของ **ISDN** ก็สมควรที่จะได้รับการกล่าวถึงเล็กน้อยอย่างรวดเร็วเกี่ยวกับ **adapter** ที่เชื่อมต่อเครื่องคอมพิวเตอร์เข้ากับสายสื่อสาร **ISDN** และการที่ **ISDN** จัดให้มีช่องกว้างของสัญญาณ หรือแบนด์วิธ (**bandwidth**) มากถึงประมาณ **2 Mbps**

ISDN adapter หรือที่เรียกอย่างถูกต้องได้มากกว่าว่า **terminal adapter (TA)** แม้ว่า **TA** จะให้บริการในการส่งและรับสัญญาณข้อมูลแต่ก็ไม่ใช่โมเด็ม และถ้าเป็นแบบที่ติดตั้งภายนอกก็จะเป็นบางสิ่งที่ทำงานเหมือนโมเด็มในการโต้ตอบกับโปรแกรมการสื่อสารของเครื่องคอมพิวเตอร์ อย่างไรก็ตามจะไม่มีหน้าที่ในการ **modulate** และ **demodulate** สัญญาณข้อมูลเหมือนโมเด็มโดยทั่วไป **ISDN adapter** จะเป็นอุปกรณ์ที่ให้บริการปรับสัญญาณให้เหมาะสม (สัญญาณข้อมูลที่ถูกส่งแยกไปยังช่องสัญญาณที่แตกต่างกัน) และให้เป็นไปตามมาตรฐานการสื่อสาร ที่สามารถเข้าใจได้โดยเครื่องคอมพิวเตอร์ โดยที่ **ISDN adapter** อาจจะเป็นแบบติดตั้งภายในโดยมีลักษณะเป็นการ์ดต่อขยาย หรือเป็นแบบติดตั้งภายนอกโดยต่อเข้ากับเครื่องคอมพิวเตอร์ทางพอร์ตสื่อสารอนุกรมหรือขนาน

10.2.4 โมเด็ม DSL

DSL หมายถึง **Digital Subscriber Line** เป็นเทคโนโลยีการสื่อสารระบบดิจิตอลเช่นเดียวกับ **ISDN** ซึ่งจะอธิบายในรายละเอียดต่อไปในภายหลัง **DSL** มีความเร็วในการส่งสัญญาณข้อมูลสูง (เป็นไปได้ที่จะส่งสัญญาณข้อมูลในขาลงด้วยความเร็วมากกว่า **7 Mbps**) และต้องการโมเด็มประเภทนี้ในการทำงาน ในความเป็นจริงแล้ว **DSL** ต้องการโมเด็ม 2 ตัว ตัวหนึ่งเชื่อมต่อกับเครื่องคอมพิวเตอร์ที่จะ **access** เข้าไปยังเครือข่ายอินเตอร์เน็ตหรือระบบเครือข่ายอื่น ส่วนอีกตัวหนึ่ง (บริหารจัดการโดยบริษัทผู้ให้บริการโทรศัพท์) ติดตั้งอยู่ที่อีกปลายหนึ่งของสายโทรศัพท์ที่เป็นทองแดงซึ่งเดินจากบ้านหรือสำนักงานของผู้ที่เป็นสมาชิกไปยังบริษัทผู้ให้บริการโทรศัพท์ โมเด็ม**DSL** มีลักษณะเป็นการ์ดต่อขยายที่ติดตั้งอยู่ภายในเครื่องคอมพิวเตอร์ โดยจะเหมือนกับโมเด็มโดยทั่วไปที่เชื่อมต่อเครื่องคอมพิวเตอร์เข้ากับปลั๊กโทรศัพท์ที่ผนัง อย่างไรก็ตามเนื่องจากการส่งสัญญาณข้อมูลของ **DSL** เป็นแบบดิจิตอลอย่างสมบูรณ์ โมเด็ม **DSL** จึงไม่ต้องทำการแปลงสัญญาณข้อมูลจากดิจิตอลเป็นอนาล็อก และจากอนาล็อกเป็นดิจิตอลยิ่งกว่านั้นโมเด็ม **DSL** อาจจะมีชิปซึ่งเรียกว่า "**splitter**" ที่แยกช่องสัญญาณในสายโทรศัพท์ออกตามลักษณะของสัญญาณเป็นสายของสัญญาณเสียงและสายของสัญญาณข้อมูล เพื่อให้การสื่อสารของโมเด็มไม่ไปรบกวนการโทรศัพท์ตามปกติ

10.3 ประเภทของการส่งสัญญาณข้อมูล (Transmission Types)

ถึงแม้ว่าโมเด็มและอุปกรณ์ซึ่งทำหน้าที่เหมือนโมเด็มจะเป็นหัวใจสำคัญของการสื่อสารระหว่างเครื่องคอมพิวเตอร์ แต่อุปกรณ์เหล่านี้ก็ไม่สามารถแลกเปลี่ยนกระแสของข้อมูลได้ตามตั้งใจและเต็มใจ โมเด็มตัวหนึ่งไม่สามารถเรียกไปยังโมเด็มตัวอื่นและส่งสัญญาณไปบอกได้ว่า "มีการส่งสัญญาณมาถึงแล้ว" หรือไม่สามารถทำให้โมเด็มตัวอื่นนั่งรอเฉยๆ เมื่อมีการติดต่อและเต็มไปด้วยอะไรก็แล้วแต่ ที่ผู้ส่งต้องการส่งมา

โมเด็มจะต้องทำงานด้วยวิธีที่ทั้งสองฝ่ายเข้าใจซึ่งกันและกัน และต้องแลกเปลี่ยนข้อมูลในรูปแบบที่ทั้งสองฝ่ายสามารถจำแนกและรองรับได้ กฎเกณฑ์ตามมาตรฐานและโปรโตคอลเริ่มที่จะมีความซับซ้อนมากขึ้นเมื่อเกี่ยวข้องกับบิตและไบต์ของข้อมูลอย่างแท้จริง แต่ในระดับของการออกอากาศ การส่งสัญญาณข้อมูลของโมเด็ม จึงจัดแบ่งได้โดยง่ายออกเป็น 2 ประเภทคือ การส่งสัญญาณแบบ **asynchronous** และการส่งสัญญาณแบบ **synchronous** (ขนานไปกับวิธีพื้นฐานในการสื่อสารของคน) ในการเลือกใช้โมเด็มเพื่อติดต่อกับระบบเครือข่าย จึงขึ้นอยู่กับว่าสภาพแวดล้อมในการสื่อสารนั้นเป็นแบบ **Asynchronous** หรือ **Synchronous**

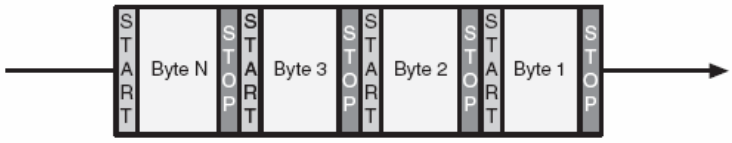
10.3.1 การสื่อสารแบบ Asynchronous

Asynchronous หมายความว่า **not synchronous** หรือไม่ถูกกำกับดูแลโดยช่วงจังหวะเวลา ซึ่งในเรื่องการสื่อสารของเครื่องคอมพิวเตอร์หมายความว่า เครื่องคอมพิวเตอร์ผู้ส่งและเครื่องคอมพิวเตอร์ผู้รับที่ทำการสื่อสารกันแบบ **asynchronous** ไม่ต้องทำการปรับช่วงจังหวะเวลาให้เข้ากันก่อนที่การส่งสัญญาณข้อมูลจะเกิดขึ้น เครื่องคอมพิวเตอร์ผู้ส่งสามารถที่จะทำการส่งได้เมื่อพร้อม จากนั้นจะหยุดและจะทำการส่งต่อหลังจากเวลาผ่านไปช่วงระยะเวลาหนึ่ง ส่วนเครื่องคอมพิวเตอร์ผู้รับจะต้องทราบว่าในกระแสของข้อมูลที่ส่งมาเป็นไบต์นั้นๆ บิตเริ่มต้น (**Start bit**) และบิตสิ้นสุด (**Stop bit**) อยู่ที่ใด การดำเนินการเกี่ยวกับบิตและไบต์กับจุดเริ่มต้นและจุดสิ้นสุดนี้ดูเหมือนจะเป็นเรื่องที่เข้าใจได้ยาก แต่ในความเป็นจริงแล้วเป็นหนึ่งในหลักเกณฑ์ที่ง่ายสำหรับความเข้าใจในการสื่อสารระหว่างเครื่องคอมพิวเตอร์ ให้เริ่มต้นจากการดูที่ชุดของบิตในการส่งสัญญาณข้อมูล ที่มีลักษณะดังนี้ (เป็นเลขฐานสองที่เทียบได้กับตัวอักษร **h** และ **i** เป็นคำว่า "**hi**")

h i
0110 1000 0110 1001

การรวมกลุ่มเลข **1s** และ **0s** ที่แตกต่างกันเหล่านี้ (โดยมากจะมีกลุ่มละ 8 ตัวเลข) ถูกนำมาใช้ในการสร้างหน่วยที่ใหญ่ขึ้นเรียกว่าไบต์ และดังที่ทราบแล้วว่าตามปกติ **1** ไบต์จะแทน **1** ตัวอักษร คือตัวอักษร เช่น **a, b** หรือ **c** ตัวเลข เช่น **1, 2** หรือ **3** เครื่องหมายวรรคตอนหรือสัญลักษณ์พิเศษ เช่น **?, !** และ **%** หรืออักขระอื่นที่ไม่สามารถพิมพ์ให้เห็นได้ เช่น สาเหตุที่ทำให้เครื่องคอมพิวเตอร์ส่งเสียง การรู้จักคีย์ **Esc** และอื่นๆ

และแน่นอนว่าเลข **1s** และ **0s** เหล่านี้เป็นเลขที่ใช้จริง ในการสื่อสารอาจจะแสดงแทนตัวเลขนี้โดยการเปลี่ยนแปลงความถี่ การเปลี่ยนแปลงความสูงของคลื่น การเปลี่ยนแปลงเฟส หรือทั้งสองแบบรวมกันเช่นในกรณีของ **quadrature amplitude modulation** ภายในเครื่องคอมพิวเตอร์ส่วนเล็กๆ ของข้อมูลเหล่านี้จะแทนด้วยแรงดันไฟฟ้าที่มีค่าแตกต่างกัน หากไม่คำนึงถึงว่าข้อมูลทั้งหมดจะถูกรวมกันเป็นชุดที่สร้างมาจากเลขฐานสอง ก็จะไม่มีชุดข้อมูลใดๆ ที่ประกอบด้วยเลข **1s** และ **0s** ที่ระบุได้ว่าแต่ละไบต์จะเริ่มต้นและสิ้นสุดที่ใด แล้วเครื่องคอมพิวเตอร์ผู้ส่งจะระบุเครื่องคอมพิวเตอร์ผู้รับได้อย่างไรเมื่อมีบิตเริ่มต้น และบิตสิ้นสุดโดยเฉพาะ? ถ้าการส่งสัญญาณข้อมูลเป็นแบบ **asynchronous** จะใช้เวลาเป็นตัวแยกอักขระต่างๆ ออกจากกันหรือไม่? คำตอบก็คือจะใช้บิตเริ่มต้น และบิตสิ้นสุดเป็นกรอบสำหรับบิตต่างๆ ที่รวมกันเป็นอักขระ โดยที่บิตเริ่มต้นจะระบุจุดเริ่มต้นของตัวอักษร และบิตสิ้นสุด (อาจจะมีขนาด **1, 1.5** และ **2 bits**) จะเป็นเครื่องหมายแสดงจุดสิ้นสุดของตัวอักษร ดังนั้นสิ่งที่อยู่ระหว่างบิตเริ่มต้นและบิตสิ้นสุดก็จะเป็นบิตที่แทนค่าตัวอักษรนั้นๆ และอาจจะมีบิตเพิ่มเติมอีกที่เรียกว่า "**parity bit**" ซึ่งใช้ในการตรวจสอบความผิดพลาดของข้อมูล ดังนั้นไบต์ที่จะทำการส่งก็จะมีรูปร่างดังแสดงตามรูปที่ **10 – 2** (ส่วนที่ระบายทึบคือ **start bit** และ **stop bit** ส่วนที่เป็นสีเทาคือบิตที่รวมกันเป็นตัวอักษร



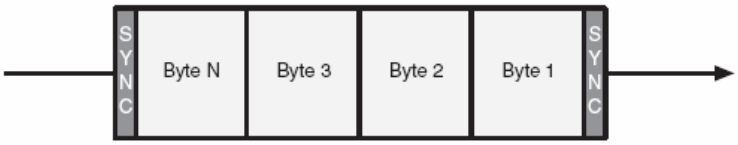
รูปที่ **10 – 2** กระแสข้อมูลที่ส่งทีละบิตแบบ **Asynchronous**

โมเด็มส่วนใหญ่จะอาศัยการส่งสัญญาณข้อมูลอนุกรมแบบ **asynchronous** ซึ่งทำงานได้ดีบนสายโทรศัพท์ (คำว่าอนุกรมมีความหมายว่าทำการส่งและรับข้อมูลตามลำดับ บิตต่อบิต) พอร์ตสื่อสารอนุกรม **RS-232** จะถูกใช้ในการสื่อสารลักษณะนี้ได้เป็นอย่างดี เนื่องจาก **RS-232** กำหนดคุณลักษณะเฉพาะของสัญญาณที่ใช้ในการส่งสัญญาณข้อมูลประเภทนี้ไว้แล้ว

10.3.2 การสื่อสารแบบ Synchronous

ด้วยเหตุที่การส่งสัญญาณข้อมูลแบบ **Asynchronous** นับว่าเป็นมาตรฐานการสื่อสารระหว่างโมเด็มกับโมเด็ม ในระบบดิจิตอลนั้นจะต้องอาศัยการสื่อสารแบบ **Synchronous** ซึ่งเป็นผลให้ส่งสัญญาณข้อมูลได้เร็วขึ้น แต่ก็มีความซับซ้อนและราคาสูงกว่าด้วย

การส่งสัญญาณแบบ **synchronous** อยู่บนพื้นฐานของการใช้เฟรม (บิตที่แบ่งออกเป็นส่วนๆ ตามช่วงระยะเวลาที่เท่ากัน) จึงแตกต่างจากการส่งสัญญาณแบบ **asynchronous** ที่มีการจัดโครงสร้างของไบต์ ด้วยบิตเริ่มต้น และบิตสิ้นสุด ในการสื่อสารแบบ **synchronous** จะต้องอาศัยพื้นฐานของช่วงระยะเวลาของสัญญาณนาฬิกา ดังนั้นก่อนที่จะทำการส่งสัญญาณข้อมูลและระหว่างการส่ง เครื่องคอมพิวเตอร์ทั้งสองจะต้องปรับแต่งช่วงจังหวะเวลาเพื่อที่จะเริ่มการสื่อสารและใช้คุณสมบัตินี้เป็นระยะๆ เพื่อตรวจสอบความเที่ยงตรงของช่วงจังหวะเวลา



รูปที่ **10 – 3** กระแสข้อมูลที่ส่งแบบ **Synchronous**

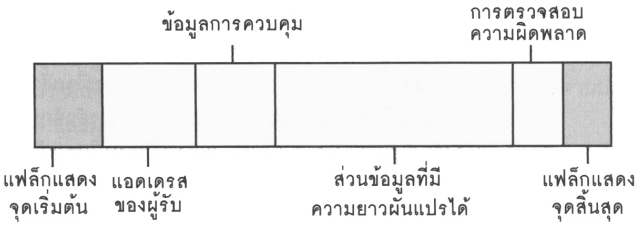
การส่งสัญญาณข้อมูลแบบ **synchronous** จะขึ้นอยู่กับโปรโตคอลที่กำกับดูแลการจัดรูปแบบของเฟรมข้อมูล รวมทั้งการควบคุมและตรวจสอบข้อผิดพลาดที่จะรวมอยู่ในเฟรมที่ทำการส่ง สำหรับโปรโตคอลการสื่อสารข้อมูลแบบ **Synchronous** ที่พบได้บ่อยและมีการทำงานใน **data link layer** มีดังนี้

10.3.2.1 High-level data link control (HDLC)

เป็นโปรดตคอลที่พัฒนามาจาก **SDLC** ที่ได้รับความคุ้มครองโดย **ISO** ดังนั้น **HDLC** จึงมีลักษณะเป็น **bit-oriented protocol** เช่นเดียวกับ **SDLC** อย่างไรก็ตามก็ใช้กันอย่างกว้างขวางในสาธารณะ มากกว่าที่จะใช้ในระบบปิดที่เป็นการส่วนตัว **HDLC** มี **subsets** เป็นจำนวนมาก โดยเวอร์ชั่นหนึ่งที่เรียกว่า **HDLC NRM (Normal Respond Mode)** มีลักษณะเดียวกับ **SDLC** ที่ว่าสนับสนุนความสัมพันธ์ระหว่าง **nodes** แบบ **master-slave** เช่นเดียวกัน อย่างไรก็ตามเมื่อกล่าวถึง **HDLC** โดยปกติจะหมายถึงเวอร์ชั่นที่เรียกว่า **LAPB (Link Access Procedure, Balanced)** ซึ่งสนับสนุนการสื่อสารแบบ **full duplex** สำหรับการเชื่อมต่อแบบ **Peer-to-peer** ซึ่งทั้งผู้ส่งและผู้รับไม่มีการควบคุมซึ่งกันและกัน **HDLC (LAPB version)** ถูกนำมาใช้ในระบบเครือข่าย **packet-switching X.25** ที่จะอธิบายในบทต่อไป โดย **HDLC frame** จะเหมือนกับ **SDLC frame**

10.3.2.2 Synchronous data link control (SDLC)

เป็นโปรโตคอลที่ได้รับการพัฒนาโดยบริษัท **IBM** สำหรับใช้ในโครงสร้าง **SNA (Systems Network Architecture)** ที่ได้รับการออกแบบมาเพื่อทำให้อุปกรณ์ประเภทต่างๆ ของ **IBM** สามารถสื่อสารกันได้ **SDLC** มีลักษณะเป็น **bit-oriented protocol** ซึ่งหมายความว่าข้อมูลจะถูกส่งในรูปแบบกระแสของบิตข้อมูล แทนที่จะเป็นตัวอักษรที่เข้ารหัส เช่น **ASCII (American Standard Code for Information Interchange)** เนื่องจากบิตจะไม่มีจุดอ้างอิงในความหมายของชุดตัวอักษร **SDLC** จึงมีบิตพิเศษตามลำดับการเกิดตัวอักษรควบคุมในลักษณะเช่นเดียวกับโปรโตคอลแบบ **Bit-oriented** ตัวอื่น สำหรับ **SDLC frame** จะมีการจัดตามรูปที่ **10 – 4** (ส่วนสีเทามีขนาด **1** ไบต์ เรียกว่า **flags** ที่เป็นเครื่องหมายแสดงจุดเริ่มต้นและจุดสิ้นสุดของเฟรมข้อมูล)

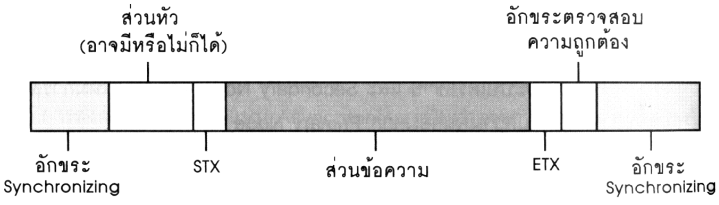


รูปที่ **10 – 4 SDLC frame**

SDLC มีพื้นฐานบนแนวความคิดของ **primary nodes** ที่ควบคุมการส่งสัญญาณข้อมูลบนระบบเครือข่าย และ **secondary nodes** ที่อนุญาตให้มีการส่งข้อมูลเฉพาะเพียงเมื่อได้รับความยินยอมจาก **primary nodes** โปรโตคอลนี้อาจจะถูกนำมาใช้ในสถานการณ์ซึ่งเครื่องคอมพิวเตอร์เมนเฟรมทำการสื่อสารกับเครื่องเทอร์มินอลหรือเครื่องเวิร์กสเตชั่นหลายๆ เครื่อง และในการเชื่อมต่อระหว่างผู้ส่งและผู้รับแบบ **point-to-point** ดังนั้นจึงมีการใช้อย่างแพร่หลายในระบบเครือข่ายแบบปิด เช่นใน **WANs** ที่อยู่บนพื้นฐานของเครื่องคอมพิวเตอร์เมนเฟรม

10.3.2.3 Binary synchronous communications protocol (Bisync)

เป็นโปรโตคอลซึ่งถูกแทนที่โดย **SDLC** ได้รับการพิจารณาให้เป็นคู่ต่อสู้เดิมของบริษัท **IBM** ในการเชื่อมโยงข้อมูลของระบบเครือข่าย โดย **Bisync** เป็นโปรโตคอลซึ่งมีลักษณะเป็น **byte-oriented** ที่มีการเข้ารหัสตัวอักษรโดยใช้ **ASCII** หรือ **EBCDIC (Extended Binary Coded Decimal Interchange Code)** ซึ่งเป็นที่รู้จักดีว่าเป็นวิธีการเข้ารหัสโดยการรวมบิตข้อมูลจำนวน **8** บิต เป็นกลุ่มทำให้มีตัวอักษรที่เป็นไปได้ **255** ตัว จึงแตกต่างจาก **SDLC** และตระกูลของ **HDLC** ดังนั้นข่าวสารใน **Bisync** จึงมีความยาวที่เปลี่ยนแปลงได้แต่มักจะเริ่มต้นและสิ้นสุดด้วยตัวอักษรการปรับแต่ง **(synchronizing characters)** และที่จุดเริ่มต้นของข้อความจะมีตัวอักษรควบคุมที่เรียกว่า **STX** นำหน้าข้อความ และที่จุดสิ้นสุดของข้อความจะตามด้วยตัวอักษรควบคุมที่เรียกว่า **ETX** และตามด้วยชุดของตัวอักษรที่ยืนยันความเที่ยงตรงในการส่งสัญญาณข้อมูล รูปที่ **10 – 5** จะแสดงให้เห็นโครงสร้างเฟรมข้อมูลของ **bisync** (ข่าวสารที่เป็นข้อความจะระบายทึบ และส่วนที่เป็นสีเทาคือตัวอักษรการปรับแต่ง)



รูปที่ **10 – 5 Bisync Frame**

10.4 ระบบเครือข่าย LAN ขนาดใหญ่

เมื่อบริษัท หรือองค์กรมีการเติบโตขึ้น ระบบเครือข่ายก็จะมีขนาดใหญ่ขึ้น ระบบ **LAN** ที่วางไว้จึงมีแนวโน้มที่จะต้องขยายตัวออกจากรูปแบบเดิมที่วางไว้ ในการใช้งานระบบเครือข่ายจะมีความรู้สึกว่าระบบเครือข่ายเดิมที่วางไว้มีขนาดเล็กลงก็ต่อเมื่อเกิดเหตุการณ์ต่างๆ ดังต่อไปนี้

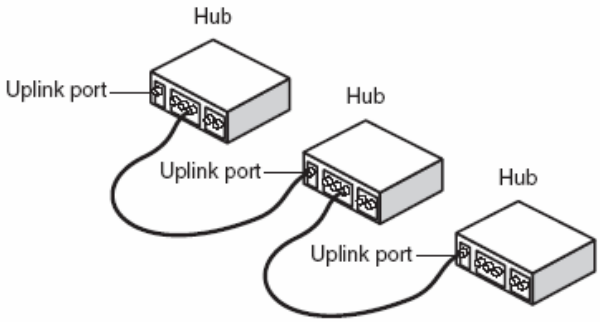
- มีความหนาแน่นของสัญญาณภายในระบบสายสัญญาณ
- การส่งไฟล์ไปยังเครื่องพิมพ์ใช้เวลานานขึ้น
- ระบบงานในเครือข่ายที่ต้องมีการรับ-ส่งข้อมูลอยู่ตลอดเวลา เช่นระบบฐานข้อมูล จะใช้เวลาในการตอบสนองต่อการร้องขอนานมากขึ้น

และเมื่อถึงเวลานั้น ผู้บริหารระบบเครือข่ายจะจำเป็นต้องขยายขนาดของระบบเครือข่ายเพื่อเพิ่มประสิทธิภาพของระบบเครือข่าย การดำเนินการดังกล่าวไม่ใช่การเพิ่มจำนวนเครื่องคอมพิวเตอร์ หรือสายสัญญาณเข้าไปในระบบ เนื่องจากสถาปัตยกรรมโครงสร้างของระบบเครือข่ายแต่ละแบบต่างมีข้อจำกัดในตัวเอง ดังนั้นจึงมีความจำเป็นต้องเพิ่มอุปกรณ์ระบบเครือข่ายบางอย่างเข้าไปในระบบเครือข่ายเดิม เพื่อขยายการเชื่อมต่อระบบเครือข่ายโดยอุปกรณ์เหล่านั้นจะต้องมีความสามารถในการ แยกกลุ่มของวง **LAN** เดิมออกจากกันเป็น **LAN** วงย่อยๆ เชื่อมต่อวง **LAN** ย่อยๆ ต่างๆ เหล่านั้นเข้าด้วยกัน และเชื่อมต่อไปยัง **LAN** วงอื่น ทำให้ระบบเครือข่ายขยายขึ้นเป็น **WAN**

องค์ประกอบต่างๆ ที่ทำให้ประสบผลสำเร็จในการขยายเครือข่าย คืออุปกรณ์ระบบเครือข่าย เช่นฮับ (**Hub**) รีพีตเตอร์ (**Repeater**) สวิตช์ (**Switch**) บริดจ์ (**Bridge**) เราท์เตอร์ (**Router**) เบราท์เตอร์ (**Brouter**) และเกตเวย์ (**Gateway**) ตามที่แนะนำไว้ในบทที่ **1** ในหัวข้อนี้จะอธิบายการนำอุปกรณ์ต่างๆ เหล่านี้มาใช้งาน

10.5 การใช้ฮับ (Hub) ในการขยายระบบเครือข่าย

ในบทที่ **1** ได้กล่าวถึงการใช้ฮับในการจัดโครงสร้างสถาปัตยกรรมแบบดาว (**Star**) และแบบโทเก็นริง มาแล้ว ถึงแม้ว่าฮับจะไม่สามารถใช้ในการขยายการเชื่อมต่อระบบเครือข่ายจาก **LAN** เป็น **WAN** ได้ แต่การเพิ่มฮับเข้าไปในระบบเครือข่ายจะทำให้สามารถเพิ่มเครื่องคอมพิวเตอร์เข้าไปในระบบเครือข่ายเพื่อทำให้ระบบเครือข่ายมีขนาดใหญ่ขึ้นได้ และเป็นวิธีที่ได้รับความนิยมอย่างสูง ภายใต้ขอบเขตที่จำกัดในการออกแบบ รูปที่ **10 – 6** แสดงให้เห็นวิธีการใช้ฮับต่อขยายระบบเครือข่ายอีเธอร์เน็ต **10BaseT** แบบอนุกรม



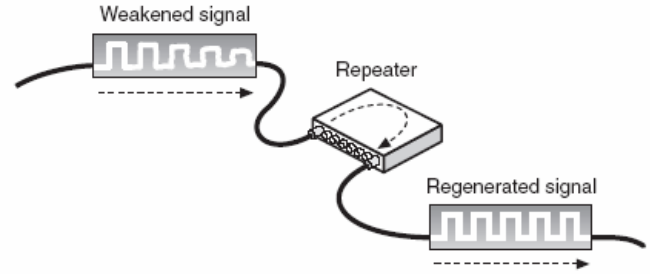
รูปที่ **10 – 6 Ethernet Hub** ต่อพ่วงกันแบบอนุกรม

<u>หมายเหตุ</u> การเชื่อมต่อฮับโดยใช้ **Uplink Port** จะต้องใช้สาย **Crossover** ที่มีความแตกต่างจากการเข้าหัวสายแบบธรรมดา การเรียงสาย **UTP** ไม่ถูกต้องจะส่งผลต่อการทำงานในภาพรวมของระบบเครือข่าย

10.6 การใช้รีพีตเตอร์ (Repeater) ในการขยายระบบเครือข่าย

เมื่อสัญญาณข้อมูลถูกส่งออกไปตามสายเคเบิล สัญญาณจะเริ่มอ่อนกำลังลง เนื่องจากการเกิดดูดซับสัญญาณ (**Attenuation**) และถ้าสายสัญญาณมีความยาวมากๆ เครื่องคอมพิวเตอร์ผู้รับจะไม่สามารถเข้าใจได้เลย การติดตั้งรีพีตเตอร์จะช่วยให้สัญญาณถูกส่งไปได้ไกลขึ้น

รีพีตเตอร์เป็นอุปกรณ์อย่างง่ายที่สุดในการขยายการไปถึงของระบบเครือข่าย หัวใจสำคัญก็คือรีพีตเตอร์เป็นอุปกรณ์ที่ได้รับการออกแบบมาเพื่อรับสัญญาณ ทำให้สัญญาณสมบูรณ์ ทำให้สัญญาณแรงขึ้น และส่งออกไปใหม่ ดังนั้นการใช้รีพีตเตอร์จึงทำให้ **LANs** ขนาดเล็กสามารถเติบโตเป็น **LANs** ที่มีขนาดใหญ่ขึ้น โดยเคลื่อนการส่งสัญญาณข้อมูลจากส่วนหนึ่งของระบบเครือข่ายไปยังอีกส่วนหนึ่ง พึงระลึกไว้เสมอว่าเมื่อข้อมูลเดินทางไปตามระบบเครือข่าย สัญญาณข้อมูลจะมีแนวโน้มที่จะอ่อนกำลังลง (สัญญาณข้อมูลถูกทำให้ผิดรูปร่างไป) รีพีตเตอร์ซึ่งมีการทำงานใน **physical layer** จะทำหน้าที่เป็นตัวกลางโดยการทำสัญญาณข้อมูลที่อ่อนกำลังลงนี้ ให้มีความแรงเพิ่มมากขึ้นก่อนที่จะส่งต่อออกไปยังส่วนอื่นของระบบเครือข่าย ดังแสดงตามรูปที่ **10 – 7**



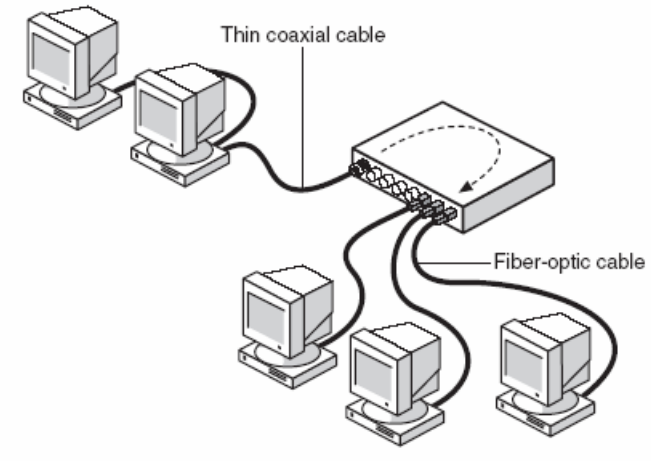
รูปที่ **10 – 7** รีพีตเตอร์สร้างสัญญาณที่อ่อนกำลังลงขึ้นมาใหม่

10.6.1 การทำงานของรีพีตเตอร์

รีพีตเตอร์จะรับสัญญาณที่ส่งอกมาจากเครื่องคอมพิวเตอร์กลุ่มหนึ่งซึ่งมีสัญญาณอ่อนลง และส่งต่อออกไปยังส่วนอื่นของระบบเครือข่าย โดยแพ็กเก็ตข้อมูล และโปรโตคอลใน **Logical Link Control (LLC) Sub-layer** จะต้องสามารถบ่งชี้ถึงเครื่องคอมพิวเตอร์ในแต่ละกลุ่มได้ ดังนั้นถึงแม้ว่ารีพีตเตอร์จะสามารถเพิ่มขนาดของระบบเครือข่ายได้ แต่ก็ไม่สามารถนำมาใช้ในการขยายระบบเครือข่ายออกไปนอกเหนือความสามารถของโครงสร้างสถาปัตยกรรมหลักของระบบเครือข่าย และไม่สามารถใช้ในการเชื่อมต่อส่วนของระบบเครือข่ายซึ่งอาศัยวิธีการ **access** ที่แตกต่างกันได้ เช่นไม่สามารถใช้รีพีตเตอร์ในการเชื่อมต่อเครือข่ายอีเธอร์เน็ต (**802.3**) ที่มีการ **access** ด้วยวิธี **CSMA/CD** เข้ากับเครือข่ายโทเก้นริง (**802.5**) ซึ่งใช้วิธีการ **access** แบบ **Token passing**

นอกจากนั้นยังไม่สามารถใช้รีพีตเตอร์ในการกรองสัญญาณข้อมูลที่เสียหายได้ เพราะรีพีตเตอร์ไม่มีความสามารถนี้ อย่างไรก็ตามก็สามารถนำรีพีตเตอร์มาใช้ในการเคลื่อนย้ายการส่งสัญญาณข้อมูลระหว่างสื่อประเภทต่างๆ ได้ เช่นระหว่างสายโคแอ็กซ์เชียล กับเคเบิลใยแก้วนำแสง และยังเป็นวิธีที่ง่ายและราคาถูกในการขยายขนาดของระบบเครือข่าย หรือจัดแบ่งระบบเครือข่ายออกเป็นส่วนย่อยที่รับภาระในการขนส่งข้อมูลน้อยลง รูปที่ **11 – 8** แสดงให้เห็นว่ารีพีตเตอร์สามารถทำการส่งแพ็กเก็ตข้อมูล จากสายสัญญาณชนิดหนึ่งไปยังสายสัญญาณอีกชนิดหนึ่งได้ กล่าวคือสามารถรับ **Ethernet packet** จากสายโคแอ็กเชียล และส่งต่อออกไปยังเคเบิลใยแก้วนำแสงได้ รีพีตเตอร์แบบหลายพอร์ต

บางรุ่นจะทำหน้าที่เหมือนฮับหลายพอร์ต และสนับสนุนการเชื่อมต่อสายสัญญาณแบบต่างๆ ซึ่งทำให้สามารถขยายระบบเครือข่ายออกไปยังเซ็กเมนต์อื่นได้ จึงแตกต่างจากฮับที่ไม่สามารถขยายไปยังเซ็กเมนต์อื่นได้



รูปที่ **10** – 8 การใช้รีพีตเตอร์เชื่อมต่อสายสัญญาณชนิดต่างๆ

10.6.2 การพิจารณาใช้รีพีตเตอร์

การใช้รีพีตเตอร์ เป็นวิธีการที่มีราคาถูกที่สุดในการขยายขนาดของระบบเครือข่ายในกรณีที่ระบบเครือข่ายมีข้อจำกัดด้านระยะทาง และมีปริมาณข้อมูลที่ไหลเวียนในระบบไม่มากนัก โดยรีพีตเตอร์จะทำการส่งข้อมูลทุกบิตจากเซ็กเมนต์หนึ่งไปยังอีกเซ็กเมนต์หนึ่ง ถึงแม้ว่าข้อมูลนั้นจะไม่อยู่ในรูปแบบที่ถูกต้อง หรือจุดหมายปลายทางของข้อมูลนั้นจะไม่ได้อยู่ในเซ็กเมนต์นั้นก็ตาม นั่นหมายความว่าปัญหาที่เกิดขึ้นในเซ็กเมนต์หนึ่งจะถูกส่งต่อไปยังเซ็กเมนต์อื่นด้วย ทั้งนี้เนื่องจากรีพีตเตอร์ไม่ได้ทำหน้าที่ในการกรองข้อมูล (**Filtering**) นั่นเอง ดังนั้นรีพีตเตอร์จะทำการแพร่กระจายสัญญาณข้อมูลนั้นกลับไปกลับมาจากเซ็กเมนต์หนึ่งสู่เซ็กเมนต์อื่นๆ ในระบบเครือข่ายที่แพร่สัญญาณอย่างต่อเนื่อง (**Broadcast Stream**) เหตุการณ์ในลักษณะนี้จะเกิดขึ้นเมื่อมีการบอร์ดคาสต์สัญญาณข้อมูลเป็นจำนวนมากจนเกินความสามารถของสายสัญญาณที่จะรองรับได้ และถ้ารีพีตเตอร์มีการตอบสนองต่อข้อมูลที่มีปัญหาอย่างต่อเนื่องในการที่จะส่งไปยังระบบที่ไม่มีการตอบสนอง สิ่งที่เกิดขึ้นก็คือประสิทธิภาพของระบบเครือข่ายจะลดลง และสุดท้ายระบบเครือข่ายจะไม่สามารถทำงานต่อไปได้ หรือที่เรียกว่า “ระบบล่ม (**Network Down**)”

ดังนั้นจึงสามารถสรุปได้ว่าให้พิจารณาใช้รีพีตเตอร์ในการขยายระบบเครือข่าย ในกรณีดังต่อไปนี้

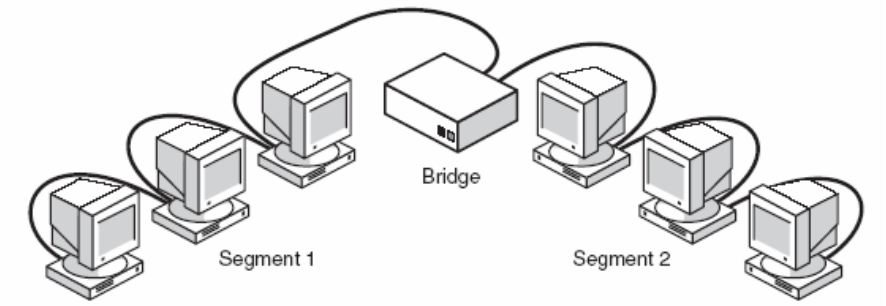
- เชื่อมต่อเซ็กเมต์ 2 เซ็กเมนต์ของระบบเครือข่ายที่ใช้สายสัญญาณแตกต่างกัน
- เพิ่มความคมชัดให้กับสัญญาณ เพื่อขยายระยะทางในการส่งข้อมูล
- ต้องการส่งข้อมูลในลักษณะ 2 ทิศทาง

และไม่ควรใช้รีพีตเตอร์ในกรณีต่างๆ ดังต่อไปนี้

- มีปริมาณข้อมูลในระบบเครือข่ายเป็นจำนวนมาก
- แต่ละเซ็กเมนต์มีวิธีการ **access** ที่แตกต่างกัน
- มีความต้องการในการกรองข้อมูลที่เสียหาย

10.7 การใช้บริดจ์ (Bridge) ในการขยายระบบเครือข่าย

Bridge (แปลว่า สะพาน) เป็นสิ่งก่อสร้างที่ข้ามแม่น้ำกว้างจะเชื่อมโยงอะไรที่ไม่สามารถเชื่อมถึงกันได้ และจัดให้มีวิธีในการเดินทางไปมาได้ตราบเท่าที่ไม่มีทางตันทางภูมิศาสตร์ สำหรับบริดจ์ของระบบเครือข่ายก็จะทำหน้าที่เช่นเดียวกัน ถึงแม้ว่าการเดินทางที่ว่านี้จะเป็นการขนส่งสัญญาณข้อมูลที่เป็นอิเล็กทรอนิกส์ และบริดจ์ในระบบเครือข่ายก็มีความฉลาดมากกว่าถนนที่เป็นเหล็กและคอนกรีต ในความเป็นจริงแล้วบริดจ์จะทำงานอย่างใดอย่างหนึ่งขึ้นอยู่กับว่าจะมองบริดจ์อย่างไร บริดจ์จะทำหน้าที่เหมือนรีพีตเตอร์ที่สามารถใช้ในการเชื่อมต่อเซ็กเมนต์ 2 ส่วนเข้าด้วยกันได้ นั่นคือเชื่อมต่อส่วนของ **LAN** ดังแสดงตามรูปที่ **10 – 9** หรือใช้ในการแยกส่วนของระบบเครือข่ายในกรณีที่มีกลุ่มของเครื่องคอมพิวเตอร์มีการส่งสัญญาณข้อมูลออกมาเป็นปริมาณมาก ซึ่งส่งผลให้ประสิทธิภาพของระบบเครือข่ายลดลง จะสามารถใช้บริดจ์ในการแยกเครื่องคอมพิวเตอร์กลุ่มที่มีปัญหานั้นออกจากระบบโดยรวมได้ หรือแยกโหนดที่มีปัญหาออกจากส่วนที่เหลือของระบบเครือข่าย



รูปที่ **10 – 9** การใช้บริดจ์เชื่อมต่อระบบเครือข่าย 2 เซ็กเมนต์

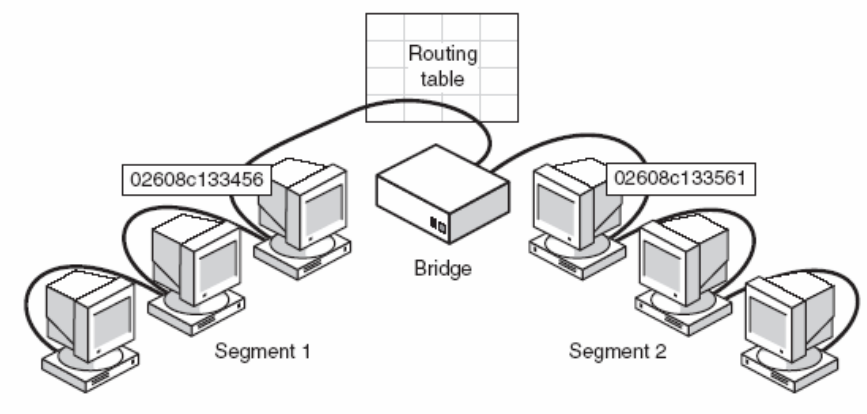
10.7.1 การทำงานของบริดจ์

บริดจ์จะมีการทำงานใน **data link layer** ทำการกรองสัญญาณและส่งผ่านแพ็กเก็ตข้อมูลไประหว่างส่วนของระบบเครือข่าย ซึ่งอาจจะเป็นส่วนของระบบเครือข่ายที่มีโครงสร้างสถาปัตยกรรมที่แตกต่างกันได้ จึงแตกต่างจากรีพีตเตอร์ซึ่งทำงานใน **physical layer** และทำการส่งผ่านสัญญาณข้อมูลระหว่างส่วนของระบบเครือข่ายที่เหมือนกัน ดังนั้นบริดจ์จะสามารถเชื่อมโยงส่วนของเครือข่ายอีเธอร์เน็ตเข้ากับส่วนของเครือข่ายโทเก้นริงได้ และถึงแม้ว่าระบบเครือข่ายทั้งคู่จะใช้โปรโตคอลที่แตกต่างกัน บริดจ์ก็สามารถที่จะโยกย้ายแพ็กเก็ตข้อมูลระหว่างระบบเครือข่ายทั้งสองได้ แต่ในการที่จะส่งผ่านแพ็กเก็ตข้อมูลจากส่วนหนึ่งไปยังอีกส่วนหนึ่งนั้น บริดจ์จะต้องเข้าใจว่าผู้ส่งและผู้รับอยู่ในเซ็กเมนต์เดียวกันหรือไม่

- ถ้าผู้ส่งและผู้รับอยู่ในเซ็กเมนต์เดียวกัน ไม่มีความจำเป็นที่จะแปลงแพ็กเก็ต ข้อมูลก่อนส่งไปยังส่วนอื่นของระบบเครือข่าย
- ถ้าผู้ส่งและผู้รับอยู่คนละเซ็กเมนต์กัน บริดจ์จะต้องทราบว่าผู้รับอยู่ที่ใดเพื่อที่จะส่งแพ็กเก็ตข้อมูลไปยังฝ่ายที่ถูกต้อง

บริดจ์จะทำเช่นนี้ได้อย่างไร? เริ่มจากวิธีการเฝ้าตรวจสอบการขนส่งข้อมูลในส่วนที่เชื่อมต่อกับบริดจ์ด้วยการทำงานใน **promiscuous mode** เมื่อมีโหนดทำการส่งข้อมูล บริดจ์จะตรวจสอบแอดเดรสของแหล่งต้นทางและแอดเดรสปลายทางของแต่ละแพ็กเก็ต แต่บริดจ์จะรู้ได้อย่างไรว่าแอดเดรสต้นทางและปลายทางนั้นอยู่ในเซ็กเมนต์เดียวกันของระบบเครือข่ายหรือคนละเซ็กเมนต์กัน ก็ต้องตรวจสอบจากฐานข้อมูลที่อยู่ในหน่วยความจำ ฐานข้อมูลนี้

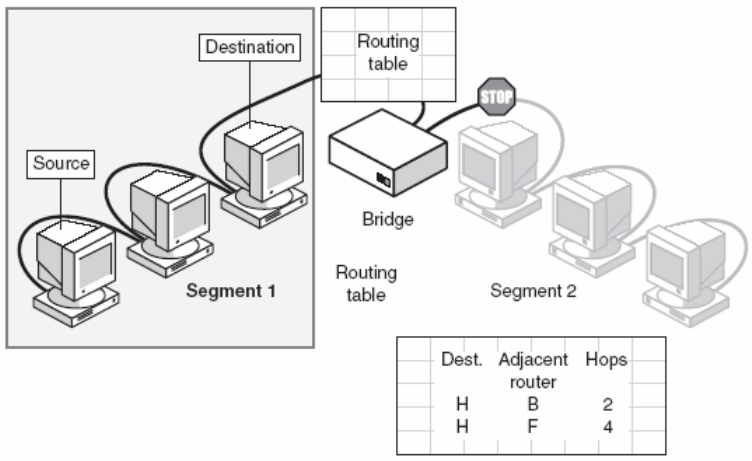
เรียกว่า “**routing table**” เมื่อบริดจ์เริ่มทำงานครั้งแรกฐานข้อมูลนี้จะว่างเปล่า แต่เมื่อมีโหนดทำการส่งข้อมูลไปมาและบริดจ์ทำงานในการตรวจสอบแอดเดรสของผู้ส่งและผู้รับ บริดจ์ก็จะเริ่มสร้างตารางโดยการเพิ่มแอดเดรสทางกายภาพของโหนดที่ยังไม่มีในรายการเข้าไปในตาราง ด้วยวิธีนี้บริดจ์ก็จะมีข้อมูลการเชื่อมต่อโหนดทั้งหมดตลอดเวลาที่เข้าไปอยู่ในเซ็กเมนต์ของระบบเครือข่าย ดังแสดงตามรูปที่ **10 – 10**



รูปที่ **10 – 10 Routing Table**

จะเกิดอะไรขึ้น เมื่อบริดจ์สร้าง **routing table** เสร็จแล้ว และได้รับแพ็กเก็ตข้อมูลที่ส่งจาก **node A** ไปยัง **node B** การที่จะทราบว่าเกิดอะไรขึ้นบ้างจะขึ้นอยู่กับว่า **node A** และ **node B** อยู่ที่ส่วนใดของระบบเครือข่าย

- ถ้าทั้ง **node A** และ **node B** อยู่ในเซ็กเมนต์เดียวกันของระบบเครือข่าย บริดจ์จะไม่สนใจแพ็กเก็ตข้อมูลนั้น และคาดว่าระบบเครือข่ายจะดูและการจัดส่งแพ็กเก็ตข้อมูลนั้นเอง
- ถ้า **node A** และ **node B** อยู่คนละเซ็กเมนต์ของระบบเครือข่าย และบริดจ์พบว่ามีแอดเดรสของ **node B** อยู่ใน **routing table** ก็จะส่งผ่านแพ็กเก็ตข้อมูลไปยัง **node B**
- ถ้า **node A** และ **node B** อยู่คนละเซ็กเมนต์ของระบบเครือข่าย แต่ยังไม่มีแอดเดรสของ **node B** ใน **routing table** บริดจ์ก็จะส่งแพ็กเก็ตข้อมูลไปยังทุกส่วนของระบบเครือข่ายยกเว้นเซ็กเมนต์ที่ **node A** อาศัยอยู่ หรืออาจกล่าวอีกอย่างหนึ่งได้ว่าบริดจ์จะแพร่กระจายแพ็กเก็ตข้อมูลไปยังทุกส่วนของระบบเครือข่ายยกเว้นส่วนที่บรรจุโหนดผู้ส่ง

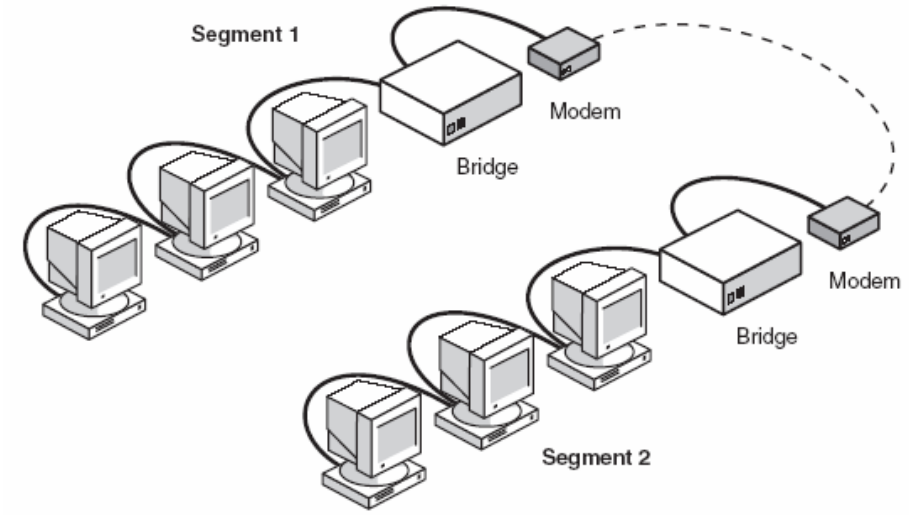


รูปที่ **10 – 11** การใช้ **Routing Table** จัดการเซ็กเมนต์ในระบบเครือข่าย

วิธีการในการเรียนรู้ว่าโหนดต่างๆ อยู่ในเซ็กเมนต์ส่วนใดของระบบเครือข่ายในลักษณะนี้ เรียกว่า "**backward learning**" ซึ่งเป็นวิธีที่เหมือนกับการเรียนรู้ถนนหนทางในเมืองใหม่ ยิ่งคุณรู้จักจุดหมายปลายทางมากเท่าใดก็จะทำให้คุณเดินทางไปได้อย่างมีประสิทธิภาพมากขึ้นเท่านั้น เช่นเดียวกับบริดจ์ ยิ่งเรียนรู้แอดเดรสของโหนดปลายทางมากเท่าใด ก็จะทำให้สามารถกำหนดเส้นทางการขนส่งแพ็กเก็ตข้อมูลได้อย่างมีประสิทธิภาพมากขึ้นเท่านั้น และจะช่วยให้ผ่อนคลายความแออัดในกระบวนการทำงานโดยการไม่สนใจแพ็กเก็ตข้อมูลที่มาจากเซ็กเมนต์ส่วนเดียวกันของระบบเครือข่าย ดังนั้นบริดจ์จะใช้ **Routing Table** เพื่อลดปริมาณการขนส่งข้อมูลในระบบเครือข่าย โดยควบคุมการไหลเวียนของแพ็กเก็ตข้อมูลระหว่างเซ็กเมนต์ กรรมวิธีนี้เรียกว่า "**Segmenting Network Traffic**" ดังนั้นเราสามารถที่จะใช้ **Bridge** หลายตัวในการรวมเซ็กเมนต์ต่างๆ เข้าเป็นระบบเครือข่ายขนาดใหญ่ได้

10.7.2 Remote Bridge

การที่บริดจ์เป็นอุปกรณ์ที่ใช้ในการเชื่อมต่อ และแยกส่วนของระบบเครือข่าย ดังนั้นจึงมักจะใช้บริดจ์ในการเชื่อมต่อระบบเครือข่ายขนาดใหญ่ที่ประกอบด้วยหลายเซ็กเมนต์ และมีพื้นที่การติดตั้งอยู่ห่างกันมาก โดยแต่ละเซ็กเมนต์จะทำการติดต่อกันผ่านทางสายโทรศัพท์ โดยใช้โมเด็มติดตั้งที่บริดจ์แต่ละตัว ดังแสดงตามรูปที่ **10 – 12**



รูปที่ **10 – 12 Remote Bridge**

เนื่องจากเครือข่ายที่อยู่ต่างสถานที่กันสามารถเชื่อมต่อถึงกันผ่านทางเครือข่ายโทรศัพท์สาธารณะ และสามารถขนส่งข้อมูลระหว่างเครือข่ายได้ **IEEE 802.1** จึงได้พัฒนา **Spanning Tree Algorithm (STA)** เพื่อให้บริดจ์สามารถรับรู้การเกิดขึ้นของข้อมูลได้มากกว่า **1** ทิศทาง ดังนั้นจึงสามารถเลือกใช้เส้นทางการรับข้อมูลได้อย่างเหมาะสม โดยโปรแกรม **STA** จะทำการปิดเส้นทางการขนส่งข้อมูลที่ไม่ได้เลือกใช้ และจะทำการเปิดเส้นทางเหล่านี้ใหม่เมื่อเส้นทางที่ใช้อยู่เดิมมีปัญหา

10.7.3 ความแตกต่างระหว่างบริดจ์ กับรีพีตเตอร์

บริดจ์ทำงานอยู่ในเลเยอร์ที่ 2 ของ **OSI Reference Model** ซึ่งสูงกว่ารีพีตเตอร์ จึงหมายความว่าบริดจ์มีความฉลาดและสามารถควบคุมเส้นทางการขนส่งข้อมูลได้ดีกว่ารีพีตเตอร์ อย่างไรก็ตามทั้งบริดจ์และรีพีตเตอร์จะมีการทำงานที่เหมือนกันในเรื่องของการเพิ่มความแรงให้กับสัญญาณ โดยบริดจ์จะทำการขยายสัญญาณในระดับแพ็กเก็ต

ดังนั้นบริดจ์จึงสนับสนุนการส่งข้อมูลในระยะไกล โดยผ่านเครือข่ายสายสัญญาณที่สนับสนุนการทำงาน ซึ่งแตกต่างจากรีพีตเตอร์ ตรงที่รีพีตเตอร์ไม่สามารถขยายสัญญาณผ่านเครือข่ายสายสัญญาณสาธารณะได้

10.7.4 การพิจารณาใช้บริดจ์

บริดจ์มีหน้าที่ทุกอย่างเหมือนกับรีพีตเตอร์ แต่สามารถต่อเข้ากับโหนดได้มากกว่า และช่วยในการเพิ่มประสิทธิภาพให้กับระบบเครือข่ายได้ดีกว่ารีพีตเตอร์ เนื่องจากสามารถแยกส่วนของระบบเครือข่ายออกเป็นหลายเซ็กเมนต์ จึงทำให้การทำงานในแต่ละเซ็กเมนต์เป็นไปอย่างรวดเร็ว นอกจากนั้นการแบ่งเซ็กเมนต์ยังช่วยลดปริมาณการชนกันของข้อมูล (**Collision**) จึงทำให้ประสิทธิภาพของระบบเครือข่ายดีขึ้น โดยบริดจ์จะใช้ **Routing Table** เป็นตัวจัดการ **Traffic** ในระบบเครือข่าย บริดจ์จึงสามารถใช้งานได้ในกรณีดังต่อไปนี้

- ขยายระยะทางการเชื่อมต่อของแต่ละเซ็กเมนต์
- สนับสนุนการเพิ่มจำนวนเครื่องคอมพิวเตอร์ในระบบเครือข่าย
- ลดสภาวะการเกิดคอขวด (**bottleneck**) อันเนื่องมาจากมีจำนวนเครื่องคอมพิวเตอร์ในเซ็กเมนต์มากจนเกินไป
- แบ่งส่วนระบบเครือข่ายออกเป็นหลายเซ็กเมนต์ เพื่อลดปริมาณการขนส่งข้อมูลระหว่างเซ็กเมนต์ ซึ่งจะทำให้ประสิทธิภาพของระบบเครือข่ายดีขึ้น
- เชื่อมต่อเซ็กเมนต์ที่ใช้สายสัญญาณที่แตกต่างกันได้

10.8 การใช้เราท์เตอร์ (Router) ในการขยายระบบเครือข่าย

ในสภาวะแวดล้อมที่ระบบเครือข่ายประกอบด้วยหลายเซ็กเมนต์ การเลือกใช้บริดจ์อาจจะทำให้ไม่สามารถติดต่อระหว่างเซ็กเมนต์ต่างๆ ได้อย่างรวดเร็ว ทั้งนี้ขึ้นอยู่กับโครงสร้างของระบบเครือข่ายและโปรโตคอลที่ใช้ ในระบบเครือข่ายขนาดใหญ่ นอกจากอุปกรณ์ที่ใช้จะต้องจดจำแอดเดรสของเครื่องคอมพิวเตอร์ทุกเครื่องในระบบเครือข่ายได้แล้ว ยังต้องสามารถตัดสินใจได้ด้วยว่าเส้นทางการขนส่งข้อมูลเส้นทางใดที่จะเหมาะสมที่สุด นอกจากนั้นยังจำเป็นต้องมีการกรองข้อมูลที่เสียหายออกเพื่อไม่ให้เกิดความหนาแน่นในการขนส่งข้อมูล

ในขณะที่รีพีตเตอร์ และบริดจ์ส่งผ่านสัญญาณข้อมูลจากเซ็กเมนต์หนึ่งของระบบเครือข่ายไปยังอีกเซ็กเมนต์หนึ่ง แต่สำหรับเราท์เตอร์จะรับหน้าที่ในกระบวนการที่ยิ่งใหญ่ต่อไปอีกขั้นหนึ่ง โดยการ

- ส่งผ่านแพ็กเก็ตข้อมูลจากระบบเครือข่ายหนึ่งไปยังระบบเครือข่ายอื่น ถึงแม้ว่าระบบเครือข่ายเหล่านั้นจะแยกห่างจากกันเป็นระยะทางไกลและมีระบบเครือข่ายอื่นอยู่ระหว่างกลาง
- ค้นหาเส้นทางการขนส่งข้อมูลที่ดีที่สุดในการจัดส่งข้อมูล
- ทำการกรองข้อมูลที่เสียหายและแยก **Traffic** ในการขนส่งข้อมูล

เราท์เตอร์สามารถที่จะโยกย้าย **packet** ข้อมูลระหว่างระบบเครือข่ายที่มีโครงสร้างสถาปัตยกรรมที่แตกต่างกันได้ เช่นเดียวกับบริดจ์ แต่เราท์เตอร์จะมีการทำงานในเลเยอร์ระดับที่สูงกว่าในโครงสร้างของระบบเครือข่าย นั่นคือใน **network layer** ของ **OSI Model**

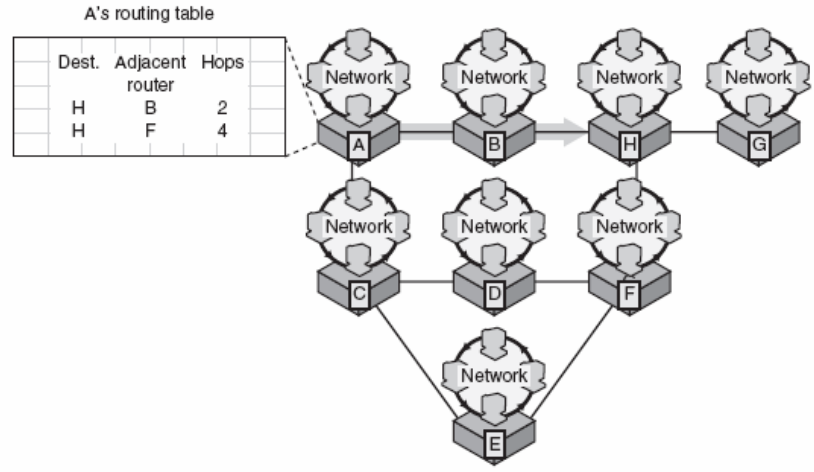
10.8.1 การทำงานของเราท์เตอร์

เราท์เตอร์มีการกำหนดเส้นทางการขนส่งข้อมูลได้อย่างไร ? เริ่มจากจะต้องทราบว่าเราท์เตอร์จะทำการสื่อสารกับเราท์เตอร์ด้วยกันเองเท่านั้น นั่นคือเราท์เตอร์จะคุยกับเราท์เตอร์ของระบบเครือข่ายอื่น โดยจะไม่ติดต่อโดยตรงกับเครื่องคอมพิวเตอร์ นอกจากนั้นเราท์เตอร์จะต้องอาศัย **routing table** ในการค้นหาระบบเครือข่ายอื่น อย่างไรก็ตาม **routing table** นี้จะแตกต่างจาก **routing table** ในบริดจ์ที่เก็บข้อมูล **physical address** (แอดเดรสของการ์ดเชื่อมต่อระบบเครือข่ายของโหนดต่างๆ) โดยเราท์เตอร์จะใช้ข้อมูล **network address** ซึ่งเป็นตัวเลขที่ระบุระบบเครือข่าย และอาจจะถึงระบบเครือข่ายย่อยที่โหนดเชื่อมต่ออยู่ ในความหมายเช่นนี้ **network address** จะคล้ายกับเลขที่บ้านในเมือง

นอกจาก **network address** แล้ว **routing table** ของเราท์เตอร์ยังประกอบด้วยข้อมูลอื่น เป็นต้นว่าเส้นทางที่เป็นไปได้ระหว่างเราท์เตอร์ ระยะห่างระหว่างเราท์เตอร์ และมีการเชื่อมต่อกับระบบเครือข่ายอื่นอย่างไร ด้วยข้อมูลที่มีอยู่เหล่านี้ เราท์เตอร์จะใช้ข่าวสารในแพ็กเก็ตข้อมูลในการพิจารณาเส้นทางที่ดีที่สุดในการเดินทางไปยังระบบเครือข่ายปลายทาง ตามอัลกอริทึมในการค้นหาเส้นทางที่ดีที่สุด ซึ่งมีปัจจัยดังนี้

- จำนวน **hops** (จำนวนครั้งที่กระโดดจากเราท์เตอร์ตัวหนึ่งไปยังเราท์เตอร์อีกตัวหนึ่ง)
- ความเร็วของสายสื่อสาร
- ความหนาแน่นของการใช้เส้นทางในเวลานั้น
- ค่าใช้จ่ายในการส่งสัญญาณข้อมูลที่เหมาะสม

เมื่อเราท์เตอร์ตกลงใจเลือกเส้นทางการส่งข้อมูลที่ดีที่สุดแล้ว ก็จะส่งผ่านแพ็กเก็ตข้อมูลไปยังเราท์เตอร์ตัวถัดไปที่อยู่ในเส้นทาง และหากจำเป็นก็จะทำแม้กระทั่งการแตกแพ็กเก็ตข้อมูลให้เล็กลง ถ้าแพ็กเก็ตข้อมูลเดิมมีขนาดใหญ่เกินกว่าที่จะเดินทางไปตามเส้นทางที่เลือกไว้ นั่นคือส่วนของงานที่ทำโดยอุปกรณ์ระบบเครือข่ายที่ฉลาด



รูปที่ **10 – 13** เราท์เตอร์ติดต่อกับเราท์เตอร์อื่น

การที่เราท์เตอร์ต้องทำงานที่ซับซ้อนมากขึ้น ทำให้เราท์เตอร์ทำงานได้ช้ากว่าบริดจ์ โดยแพ็กเก็ตข้อมูลที่ถูกส่งจากเราท์เตอร์ตัวหนึ่งไปยังเราท์เตอร์อีกตัวหนึ่ง จะถูกถอดออกและสร้างใหม่ใน **Data link Layer** เพื่อปรับปรุงแอดเดรสต้นทางและแอดเดรสปลายทาง จึงทำให้เราท์เตอร์สามารถส่งข้อมูลโดย **TCP/IP** จากระบบเครือข่ายอีเธอร์เน็ต (**802.3**) ออกไปยังระบบเครือข่ายโทเก้นริง (**802.5**) ได้ และเนื่องจากเราท์เตอร์จะอ่านเฉพาะข้อมูลที่อยู่ใน

แพ็กเก็ต จึงไม่อนุญาตให้ข้อมูลที่ไม่สมบูรณ์ถูกส่งออกไปยังระบบเครือข่าย โดยไม่ทำการแพร่กระจายข้อมูลที่ไม่สมบูรณ์เหล่านั้น จึงไม่สร้างความสับสนให้กับระบบเครือข่าย

เราท์เตอร์จะไม่สนใจโหนดปลายทาง แต่จะสนใจเฉพาะ **Network Address** ปลายทางเท่านั้น และจะทำการส่งข้อมูลออกไปยังปลายทางก็ต่อเมื่อรู้จักกับเราท์เตอร์ปลายทางแล้วเท่านั้น ซึ่งเป็นวิธีการหนึ่งที่เราท์เตอร์ใช้ในการควบคุมการหมุนเวียนของกระแสข้อมูลในระบบเครือข่าย ซึ่งจะช่วยลด **Traffic** ในระบบเครือข่าย จึงทำให้ เราท์เตอร์สามารถใช้เส้นทางในการเชื่อมต่อระหว่างเซ็กเมนต์ของระบบเครือข่ายได้อย่างมีประสิทธิภาพมากยิ่งขึ้น

การเลือกเส้นทางการขนส่งข้อมูลของเราท์เตอร์จะแตกต่างจากบริดจ์ตรงที่เราท์เตอร์สามารถเชื่อมต่อระบบเครือข่ายได้หลายระบบในเวลาเดียวกัน และจากการที่เราท์เตอร์สามารถทำการเชื่อมต่อเซ็กเมนต์ของระบบเครือข่ายต่างๆ ซึ่งมีแพ็กเก็ตข้อมูลแตกต่างกันได้ จึงทำให้เราท์เตอร์มีเส้นทางการขนส่งข้อมูลเพิ่มมากขึ้น ดังนั้นหากเราท์เตอร์ตัวใดหยุดการทำงาน ระบบเครือข่ายยังสามารถส่งข้อมูลไปในเส้นทางอื่นได้ นอกจากนั้นเราท์เตอร์ยังสามารถตรวจสอบความหนาแน่นในการขนส่งข้อมูลตามเส้นทางต่างๆ ได้ และเราท์เตอร์จะใช้ข้อมูลนี้ในการตัดสินใจว่าจะจัดส่งข้อมูลไปในเส้นทางใด สำหรับอัลกอริทึมที่ใช้ในการเลือกเส้นทางการขนส่งข้อมูลมีดังนี้

- **OSPF (Open Shortest Path First)** เป็นอัลกอริทึมที่ทำงานในขณะทำการรับหรือส่งข้อมูลในสถานะ **Link** โดยจะควบคุมกระบวนการในการตอบสนองต่อการเปลี่ยนแปลงเส้นทางการขนส่งแพ็กเก็ตข้อมูล
- **RIP (Routing Information Protocol)** เป็นโปรโตคอลที่มีกลไกในการตรวจสอบระยะทางในการขนส่งข้อมูล เพื่อใช้ในการตัดสินใจเลือกเส้นทางการขนส่งข้อมูลที่ใช้เวลาน้อยที่สุด โปรโตคอลนี้สามารถทำงานร่วมกับโปรโตคอล **TCP/IP** และ **IPX** ได้
- **NLSP (NetWare Link Service Protocol)** เป็นอัลกอริทึมที่ทำงานในขณะทำการรับหรือส่งข้อมูลในสถานะ **Link** ของ **NetWare**

10.8.2 Routable Protocol

ถึงแม้ว่างานทั้งหมดของเราท์เตอร์จะเป็นแบบเดียวกัน แต่ไม่ใช่ว่าโปรโตคอลทุกตัวจะสามารถทำงานร่วมกับ **Router** ในการกำหนดเส้นทางการขนส่งข้อมูลได้ โปรโตคอลที่สามารถสนับสนุนการทำงานของเราท์เตอร์ได้ คือ โปรโตคอลซึ่งมีความสามารถในการกำหนดเส้นทางการขนส่งข้อมูล ดังนี้

- **DECnet**
- **IP (Internet Protocol)**
- **IPX (Internetwork Pacet Exchange)**
- **XNS (Xerox Network System**
- **DDP** ใน **AppleTalk**

และโปรโตคอลที่ไม่สามารถกำหนดเส้นทางการขนส่งข้อมูลได้ คือ

- **LAT (Local Area Transport)** ของบริษัท **Digital Equipment Corporation**
- **NetBUEI (NetBIOS Extended User Interface)**

<u>หมายเหตุ</u> ในปัจจุบันเราท์เตอร์สามารถกำหนดเส้นทางการขนส่งข้อมูลได้โดยใช้โปรโตคอลมากกว่า **1** ตัว เช่นสามารถใช้ **RIP** และ **DEC** ได้ในเวลาเดียวกัน

10.8.3 ชนิดของเราท์เตอร์

เราท์เตอร์แบ่งออกเป็น 2 ชนิดใหญ่ๆ คือ

10.8.3.1 Static Router

เป็นเราท์เตอร์ที่ต้องอาศัยผู้บริหารระบบเครือข่ายในการตั้งค่าการใช้งาน โดยเป็นผู้กำหนด **Route Table** ในแต่ละเส้นทางให้กับเราท์เตอร์ และจะใช้หลักการพื้นฐานของเส้นทางเดิมที่เคยติดต่อในการตัดสินใจส่งแพ็กเก็ตข้อมูลออกไปบนระบบเครือข่าย เส้นทางในการขนส่งข้อมูลจึงอาจไม่ใช่เส้นทางที่ดีที่สุดเสมอไป เราท์เตอร์ชนิดนี้จะมีความปลอดภัยสูงเนื่องจากผู้บริหารระบบเครือข่ายจะเป็นผู้ตัดสินใจเลือกเส้นทางการขนส่งข้อมูล อย่างไรก็ตามการใช้เราท์เตอร์ชนิดนี้จะต้องการผู้บริหารระบบเครือข่ายที่มีความรู้ ความสามารถสูง และมีประสบการณ์ในการทำงานเกี่ยวข้องกับระบบเครือข่ายเป็นเวลานาน

10.8.3.2 Dynamic Router

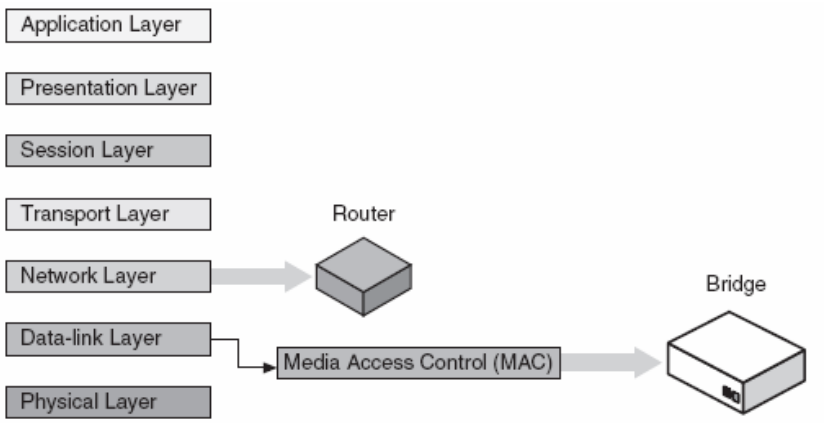
เป็นเราท์เตอร์ที่ได้รับการออกแบบมาให้มีความสามารถในการเรียนรู้ และจดจำเส้นทางในการขนส่งแพ็กเก็ตข้อมูลได้เอง จึงไม่ต้องมีการตั้งค่าการใช้งานมาก จะทำการตั้งค่าเริ่มต้นการใช้งานจากบริษัทผู้ผลิตเพียงครั้งเดียว นอกจากนั้น **Dynamic Router** ยังมีความสามารถในการศึกษาข้อมูลจากเราท์เตอร์ตัวอื่นเพิ่มเติมก่อนที่จะทำการส่งแพ็กเก็ตข้อมูลออกไปบนระบบเครือข่าย การตัดสินใจเลือกเส้นทางการขนส่งข้อมูลจะอยู่บนพื้นฐานของความสิ้นเปลืองและความหนาแน่นของเส้นทางในการขนส่งข้อมูล สำหรับด้านการรักษาความปลอดภัย หากในองค์กรมีผู้บริหารระบบเครือข่ายที่มีความรู้และประสบการณ์สูง สามารถที่จะเพิ่มการตั้งค่าการรักษาความปลอดภัยด้วยการเพิ่มการกรองระบบเครือข่ายบางส่วน เพื่อไม่ให้มีกระแสข้อมูลไหลเวียนออกไปบนระบบเครือข่ายมากจนเกินไป

10.8.4 ความแตกต่างระหว่างบริดจ์ กับเราท์เตอร์

บ่อยครั้งที่เกิดความสับสนในการใช้งานระหว่างบริดจ์ และเราท์เตอร์ ไม่ว่าจะเป็นการใช้งานในระบบเครือข่าย **LAN** หรือ **WAN** จึงมีคำถามเกิดขึ้นว่าเมื่อใดจึงควรใช้บริดจ์ และเมื่อใดควรใช้เราท์เตอร์

บริดจ์จะมีทำงานใน **Data Link Layer (MAC Sub-layer)** ของ **OSI Reference Model** จึงรู้จักเฉพาะแอดเดรสของโหนดในระบบเครือข่าย ในการทำงาน บริดจ์จะทำการตรวจสอบแพ็กเก็ตข้อมูลที่ถูกส่งออกมาบนระบบเครือข่าย เพื่อหาแอดเดรสของโหนดต้นทางและปลายทาง ถ้าบริดจ์ทราบที่อยู่ของแต่ละโหนดแล้ว แพ็กเก็ตข้อมูลนั้นจะถูกส่งออกไปยังเซ็กเมนต์ที่เหมาะสม แต่ถ้าบริดจ์ไม่ทราบที่อยู่ของโหนดปลายทางแล้ว บริดจ์จะทำการแพร่กระจายข้อมูลนั้นออกไปในทุกเซ็กเมนต์บนระบบเครือข่าย ยกเว้นเซ็กเมนต์ของโหนดต้นทาง

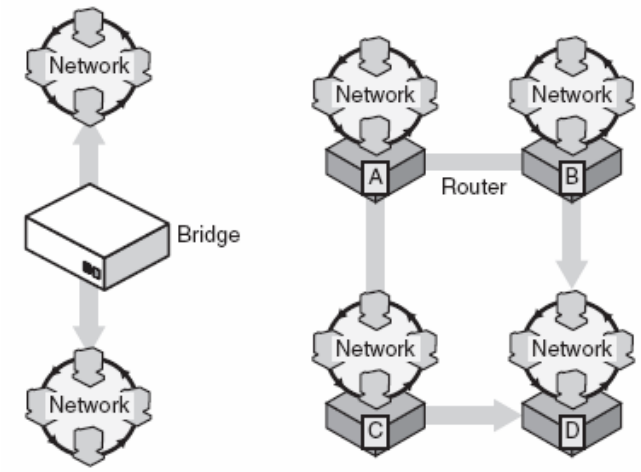
เราท์เตอร์จะมีการทำงานใน **Network Layer** ซึ่งมีส่วนเกี่ยวข้องกับแพ็กเก็ตข้อมูลมากกว่าบริดจ์ โดยนอกจากจะสามารถตัดสินใจได้ว่าจะทำการส่งแพ็กเก็ตข้อมูลออกไปในเส้นทางใดแล้ว ยังสามารถตัดสินใจได้ด้วยว่าจะทำการส่งข้อมูลนั้นออกไปอย่างไร โดยมีการตรวจสอบโปรโตคอลที่ใช้ และยิ่งไปกว่านั้นเราท์เตอร์ยังทำการตรวจสอบแอดเดรสของเราท์เตอร์ตัวอื่นด้วย เพื่อตัดสินใจว่าจะส่งข้อมูลผ่านเราท์เตอร์ตัวใดไปยังจุดหมายปลายทาง รูปที่ **10 – 14** แสดงให้เห็นความแตกต่างในการทำงานของเราท์เตอร์และบริดจ์ เมื่อเปรียบเทียบกับโครงสร้าง **OSI Reference Model**



รูปที่ **10 – 14** เราท์เตอร์ทำงานใน **Network Layer** ส่วนบริดจ์ทำงานใน **MAC Sub-layer** ของ **Data Link Layer**

การเรียนรู้เรื่องของการแพร่กระจายสัญญาณข้อมูล หรือบรอดคาสต์ (**broadcast**) จะช่วยให้เข้าใจความแตกต่างระหว่างบริดจ์ กับเราท์เตอร์ได้อย่างชัดเจน สำหรับ บริดจ์เมื่อไม่ทราบแอดเดรสปลายทางที่แน่นอนของโหนดจะทำการบรอดคาสต์สัญญาณข้อมูลออกไปทางทุกพอร์ตที่เชื่อมต่ออยู่ ยกเว้นพอร์ตซึ่งส่งแพ็กเก็ตข้อมูลออกมา ซึ่งหมายความว่าเครื่องคอมพิวเตอร์ทั้งหมดบนระบบเครือข่ายจะได้รับสัญญาณข้อมูลนั้น ยกเว้นเครื่องคอมพิวเตอร์ซึ่งอยู่ในเซ็กเมนต์เดียวกับเครื่องคอมพิวเตอร์ผู้ส่ง ซึ่งหากเป็นระบบเครือข่ายขนาดเล็กเหตุการณ์เช่นนี้จะไม่ส่งผลกระทบมากนัก แต่หากเป็นระบบเครือข่ายขนาดใหญ่ เหตุการณ์เช่นนี้จะเป็นการเพิ่มความหนาแน่นในเส้นทางการขนส่งข้อมูลทั้งหมด ซึ่งอาจส่งผลให้ประสิทธิภาพในการทำงานของระบบเครือข่ายลดลง และระบบล่มในที่สุด

ในการจัดส่งแพ็กเก็ตข้อมูล บริดจ์สามารถทำการส่งแพ็กเก็ตข้อมูลผ่านเส้นทางใดๆ ได้เพียงเส้นทางเดียว แต่เราท์เตอร์สามารถเลือกใช้เส้นทางการขนส่งข้อมูลได้หลายเส้นทาง และยังสามารถตัดสินใจได้ด้วยว่าเส้นทางใดเป็นเส้นทางที่ดีที่สุดในขณะนั้น เช่น หาก **Router A** ต้องการที่จะส่งข้อมูลไปยัง **Router D** จะสามารถเลือกได้ว่าจะทำการส่งผ่าน **Router B** หรือ **C** ดังแสดงตามรูปที่ **10 – 15** นอกจากนั้นในการจัดส่งแพ็กเก็ตข้อมูลของเราท์เตอร์จะต้องมีการทำงานร่วมกับโปรโตคอลที่สามารถกำหนดเส้นทางในการขนส่งข้อมูล (**Routable Protocol**) ได้เท่านั้น แต่สำหรับบริดจ์ไม่จำเป็น



รูปที่ **10 – 15** การเลือกเส้นทางการขนส่งแพ็กเก็ตข้อมูลของเราท์เตอร์

10.9 การใช้เกตเวย์ (Gateway) ในการขยายระบบเครือข่าย

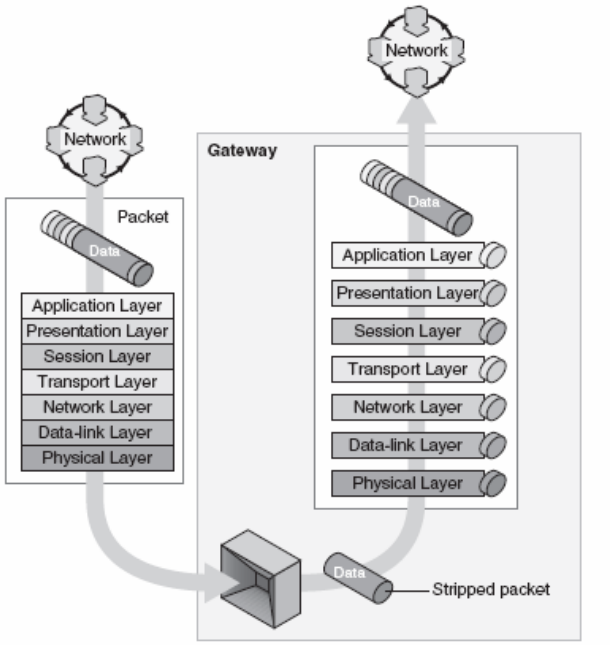
ในขณะที่รีพีตเตอร์ บริดจ์ และเราท์เตอร์ กล่าวถึงการส่งผ่านสัญญาณข้อมูลในระบบเครือข่าย หรือระหว่างระบบเครือข่าย แต่เกตเวย์ (Gateways) จะมีจุดมุ่งหมายที่แตกต่างออกไปเล็กน้อย ถึงแม้ว่าผลสุดท้ายก็เป็นงานในการสื่อสารระหว่างระบบเครือข่ายเช่นเดียวกัน ในกรณีของเกตเวย์จะเป็นการสื่อสารระหว่างระบบเครือข่ายที่ไม่เหมือนกัน เช่น คอมพิวเตอร์เมนเฟรมกับ **LANs** (ระหว่าง **IBM's SNA** กับระบบเครือข่ายที่รัน **Microsoft's Windows NT/2000 Server** และ **TCP/IP**) หรือระบบเครือข่าย **AppleTalk** กับระบบเครือข่าย **Token Ring**

10.9.1 การทำงานของเกตเวย์

เกตเวย์มีการทำงานในเลเยอร์ระดับที่สูงกว่าในโครงสร้างระบบเครือข่าย จึงแตกต่างจากบริดจ์ และอุปกรณ์อื่นที่ได้อธิบายก่อนหน้านี้ โดยทั่วไปเกตเวย์มักจะทำงานใน **application layer** แต่ก็มีบางกรณีที่ทำงานใน **presentation layer** และ **session layer** หรือในบางครั้งก็ทั้ง **7** เลเยอร์ ใน **OSI Model**

โดยมากเกตเวย์จะเป็นเครื่องคอมพิวเตอร์ที่ได้รับการอุทิศให้ทำงานในการแปลงแพ็กเก็ตข้อมูลให้อยู่ในรูปของโปรโตคอลซึ่งใช้โดยระบบเครือข่ายที่เชื่อมต่ออยู่ ในการทำงานนี้เกตเวย์จะต้องนำแพ็กเก็ตข้อมูลที่จะทำการส่งและแยกโปรโตคอลที่ใช้โดยระบบเครือข่ายที่ส่งออกมาตามลำดับชั้น อย่างไรก็ตามก่อนที่จะส่งแพ็กเก็ตข้อมูลต่อออกไปก็จะทำการจัดหีบห่อแพ็กเก็ตข้อมูลนั้นใหม่โดยห่อหุ้มด้วยโปรโตคอลที่ระบบเครือข่ายผู้รับใช้ กระบวนการทั้งหมดนี้เปรียบเทียบได้กับการเปลี่ยนชุดที่ใส่เวลากลางวันซึ่งเป็นที่ต้องการของระบบเครือข่ายของคุณ ไปเป็นชุดเล่นกีฬาตามประเภทของกีฬา เช่นในทีมบาสเกตบอล หรือทีมฟุตบอล

การแปลงโปรโตคอลนี้ทำให้แต่ละระบบเครือข่ายมองเห็นระบบเครือข่ายอื่นเป็นเหมือนกับระบบของตัวเอง ถึงแม้ว่าโครงสร้างสถาปัตยกรรมและโปรโตคอลที่ใช้อาจจะแตกต่างกันอย่างสิ้นเชิงก็ตาม และด้วยการใช้วิธีการนำ **Protocol Stack** เดิมออกจากแพ็กเก็ตข้อมูล และเพิ่ม **Protocol Stack** ใหม่เข้าไปนี้ เกตเวย์ก็จะทำให้ระบบเครือข่ายต่างๆ สามารถสื่อสารกันได้ ดังแสดงตามรูปที่ **10 – 16**



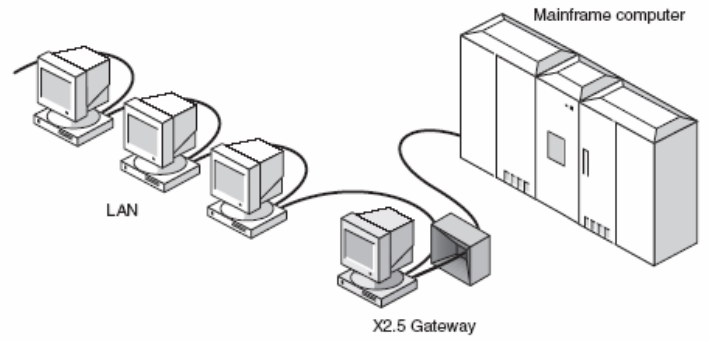
รูปที่ **10 – 16** เกตเวย์ทำงานโดยการถอด **Protocol Stack** เดิมออก แล้วใส่ **Protocol Stack** ใหม่เข้าไป

หากกล่าวโดยทั่วไปแล้วอาจทำให้เกิดความสับสนได้ โดยเฉพาะในการอธิบายเรื่องเกี่ยวกับเครือข่ายอินเตอร์เน็ต มักใช้คำว่า "เกตเวย์" ในการกล่าวถึงบางสิ่งที่นอกเหนือจากเกตเวย์นี้ เช่นในบางครั้งก็จะเรียกเราท์เตอร์ว่าเป็นเกตเวย์ เมื่อพูดถึงเรื่อง **Web** จะใช้เกตเวย์เป็น **access point** ในการเชื่อมเข้ากับ **backbone** ของเครือข่ายอินเตอร์เน็ต และยังมี **e-mail gateways** ซึ่งทำการแปลงข่าวสารที่ใช้โปรโตคอลที่แตกต่างกัน นอกจากนั้นยังมีเกตเวย์ที่สนับสนุนการสื่อสารจากระยะไกลและแพ็กเก็ตสวิตชิ่ง หรือแม้กระทั่งเกตเวย์ที่มีความสัมพันธ์กับซอฟต์แวร์ เช่น **CGI (Common Gateways Interface)** ซึ่งเป็นคุณลักษณะเฉพาะที่ได้รับการออกแบบมาเพื่อสนับสนุนการโต้ตอบระหว่างโปรแกรมประยุกต์กับเครื่องเว็บเซิร์ฟเวอร์ **(Web Servers)** อย่างไรก็ตามในแต่ละกรณีที่กล่าวมาแล้ว จะเป็นการช่วยได้มากหากจำไว้ว่า คำว่าเกตเวย์ หมายถึงฮาร์ดแวร์ ซอฟต์แวร์ และการอินเตอร์เฟซบางอย่างที่ทำให้เกิดการสื่อสารซึ่งอาจเกิดขึ้นไม่ได้

10.9.2 Mainframe Gateway

เกตเวย์เพียงตัวเดียวไม่ใช่โครงสร้างที่ใช้ได้ทั่วไป เนื่องจากประเภทของการแปลงข้อมูลทำให้เกตเวย์สามารถรองรับการโยกย้ายข้อมูลได้เฉพาะระบบเครือข่าย ดังนั้นเกตเวย์สำหรับแปลงข้อมูลจาก **SNA** ไปเป็น **Windows NT/2000 TCP/IP** ไม่สามารถที่จะสนับสนุนการแปลงข้อมูลจาก **AppleTalk** ไปเป็น **Token Ring** ได้ ดังคติพจน์ที่ว่า "ของใครของมัน" ถึงแม้ว่าในบางครั้งอาจจะมีเกตเวย์ที่จัดให้มีฮาร์ดแวร์และซอฟต์แวร์ที่ทำให้สามารถทำการแปลงข้อมูลได้มากกว่า **1** ประเภท แต่ในการแปลงข้อมูลแต่ละประเภทก็ยังต้องการฮาร์ดแวร์และซอฟต์แวร์โดยเฉพาะในการสนับสนุนการแปลงข้อมูลแต่ละประเภท

การใช้งานอย่างหนึ่งของเกตเวย์ที่มีประโยชน์คือการเชื่อมต่อเครื่องคอมพิวเตอร์ **PC** เข้ากับ **LAN** ที่มีเครื่องมินิคอมพิวเตอร์ หรือเครื่องคอมพิวเตอร์เมนเฟรมเป็นศูนย์กลาง ภายใต้สภาวะแวดล้อมนี้เครื่องคอมพิวเตอร์เครื่องหนึ่งจะถูกนำมาใช้เป็นเกตเวย์ เพื่อให้เครื่องคอมพิวเตอร์เครื่องอื่นสามารถใช้เป็นช่องทางในการติดต่อกับเครื่องคอมพิวเตอร์เมนเฟรมได้ ดังแสดงตามรูปที่ **10 – 17**



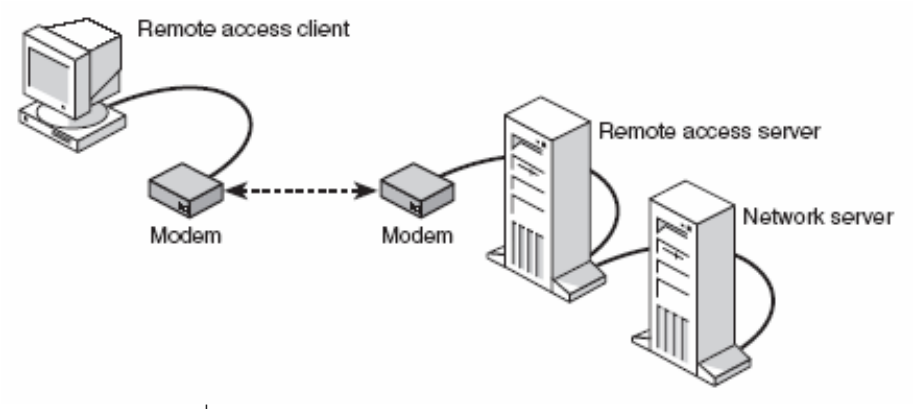
รูปที่ **10 – 17** การเชื่อมต่อเครื่องคอมพิวเตอร์ **PC** เข้ากับคอมพิวเตอร์เมนเฟรม

10.9.3 การพิจารณาใช้เกตเวย์

เกตเวย์จัดเป็นทางเลือกหนึ่งในการขยายระบบเครือข่าย โดยปกติเกตเวย์จะเป็นเครื่องคอมพิวเตอร์ **1** เครื่องที่มีลักษณะการทำงานเฉพาะ โดยคอมพิวเตอร์เครื่องนี้จะมีความสำคัญเท่ากับเครื่องเซิร์ฟเวอร์ เนื่องจากจะต้องทำการแปลงแพ็กเก็ตข้อมูล หากพิจารณาใช้เกตเวย์ให้ทำงานหลายชนิดพร้อมกัน การพิจารณาจำนวน **RAM** และประสิทธิภาพของ **CPU** จะส่งผลต่อประสิทธิภาพของระบบเครือข่ายในภาพรวม

10.10 บริการการเชื่อมต่อระยะไกล (Remote Access Service)

บ่อยครั้งที่องค์กรในภาคธุรกิจมีความจำเป็นต้องเชื่อมต่อระบบเครือข่ายมากกว่า 1 เครือข่าย หรือเชื่อมต่อเครื่องคอมพิวเตอร์จากบ้านเข้ามายังระบบเครือข่ายขององค์กร เพื่อสนองความต้องการนี้ระบบปฏิบัติการเครือข่าย (NOS) จึงมีบริการการเชื่อมต่อระยะไกล หรือ RAS รวมอยู่แล้ว ในการสนับสนุนการเชื่อมต่อระยะไกลจะต้องอาศัยบริการ 2 ชนิดคือ RAS และบริการบนเครื่องลูกข่าย หรือที่รู้จักกันในชื่อว่า Dial-up Networking โดยเครื่องเซิร์ฟเวอร์จะต้องมีบริการ RAS เพื่อรองรับการเชื่อมต่อจากเครื่องลูกข่าย ส่วนที่เครื่องลูกข่ายจะต้องใช้บริการ Dual-up Networking ในการติดต่อผ่านทางโมเด็มเข้าไปยังเครื่องเซิร์ฟเวอร์ซึ่งอยู่อีกฟากหนึ่งของระบบเครือข่าย เมื่อรวมบริการทั้งคู่เข้าด้วยกันจึงทำให้เกิดการขยายระบบเครือข่ายขึ้น ดังแสดงตามรูปที่ 10 – 18



รูปที่ 10 – 18 การขยายระบบเครือข่ายโดยใช้บริการ RAS

10.10.1 RAS Connection

การเชื่อมต่อเข้าสู่ RAS Server จากระยะไกล จำเป็นต้องอาศัยบริการเครือข่ายการสื่อสารพื้นฐาน ซึ่งจะได้อธิบายรายละเอียดของบริการต่างๆ ต่อไปภายหลัง บริการเหล่านี้ประกอบด้วย

- PSTN (Public Switching Telephone Network) หรือเครือข่ายโทรศัพท์สาธารณะ
- X.25 เป็นบริการ Packet Switching ที่สามารถรองรับการใช้บริการ Dial-up
- ISDN (Integrated Service Digital Network) ให้บริการการเชื่อมต่อด้วยความเร็วสูง แต่ราคาค่าบริการจะแพงกว่า Dial-up และต้องใช้ ISDN Card แทนโมเด็ม
- ADSL (Asynchronous Digital Subscriber Line) เป็นบริการการเชื่อมต่อความเร็วสูง ที่ทำบนเครือข่ายสายโทรศัพท์พื้นฐาน และต้องใช้ ADSL Modem

10.10.2 RAS Protocol

บริการ RAS สนับสนุนโปรโตคอล 3 ชนิด คือ

- SLIP (Serial Line Interface Protocol) เป็นโปรโตคอลชนิดแรกถูกนำมาใช้ตั้งแต่ปี ค.ศ.1984 แต่โปรโตคอล SLIP ไม่สนับสนุน Dynamic IP, NetBUEI หรือ IPS Protocol นอกจากนั้นยังไม่สนับสนุนการเข้ารหัสข้อมูล และเครื่องคอมพิวเตอร์ที่จะติดต่อเข้ามายัง RAS Server จะต้องทำการติดตั้ง RAS Client

- **PPP (Point-to-Point Protocol)** เป็นโปรโตคอลที่สนับสนุนการเข้ารหัสข้อมูล และสนับสนุนโปรโตคอล **TCP/IP, IPX, NetBUEI, AppleTalk** และ **DECnet** จึงทำให้เครื่องลูกข่ายสามารถติดต่อเข้ามายังระบบเครือข่ายผ่านเครือข่ายอินเตอร์เน็ตได้
- **PPTP (Point-to-Point Tunneling Protocol)** เป็นโปรโตคอลหลักที่สนับสนุนการจัดตั้งระบบเครือข่ายเสมือน **(Virtual Private Network – VPN)** โดยมีการรักษาความปลอดภัยให้กับข้อมูลในการติดต่อด้วย **TCP/IP** บนเครือข่ายสาธารณะ

10.10.3 การรักษาความปลอดภัยของ RAS

กระบวนการในการรักษาความปลอดภัยของบริการเชื่อมต่อจากระยะไกลจะขึ้นอยู่กับระบบปฏิบัติการที่ใช้ โดย **RAS** จะมีหน้าที่ในการรักษาความปลอดภัยให้กับข้อมูล ดังนี้

- **Auditing** เป็นการตรวจสอบข้อมูลในขณะที่ผู้ใช้ติดต่อเข้ามายังระบบเครือข่าย และบันทึกเวลาที่ใช้ในการติดต่อ
- **Call Back** เราสามารถตั้งค่าให้ **RAS** ทำการติดต่อกลับไปยัง **Host** ที่ต้องการให้ทำการติดต่อได้ โดยการจำกัดหมายเลขโทรศัพท์ที่จะสามารถติดต่อเข้ามายัง **RAS Server** ได้ เมื่อมีการจัดตั้งการเชื่อมต่อ **RAS Server** จะติดต่อกลับไปยังเครื่องลูกข่ายนั้น
- **Security Host** เป็นการเพิ่มขั้นตอนในการขอติดต่อกับ **RAS Server**
- **PPTP Filtering** เป็นกระบวนการในการกรองข้อมูลที่ไม่ได้ใช้โปรโตคอล **PPTP** ในการสื่อสารผ่านระบบเครือข่ายเสมือน **(VPN)** ที่จัดตั้งขึ้นบนเครือข่ายอินเตอร์เน็ต

10.10.4 การติดตั้ง RAS

การวางแผนติดตั้ง **RAS** จะเริ่มจากการรวบรวมข้อมูลของเครื่องคอมพิวเตอร์ในระบบเครือข่าย และผู้ใช้งานระบบเครือข่ายที่ต้องการติดต่อเข้ามาจากระยะไกล ข้อมูลดังกล่าวประกอบด้วย

- ชนิดของโมเด็ม การติดตั้ง และไดร์ฟเวอร์ โดยโมเด็มจะต้องรองรับบริการ **RAS**
- ชนิดของพอร์ตที่ต้องการใช้ในการติดต่อ
- ความต้องการในด้านการรักษาความปลอดภัยข้อมูล
- ลักษณะการทำงานของโมเด็มเป็นแบบ **Dial In** หรือ **Dial Out** ทั้งคู่
- โปรโตคอลของเครื่องไคลเอนต์ **(Client Protocol)**

10.10.5 การตั้งค่า RAS

หลังจากทำการติดตั้งบริการ **RAS** แล้ว จะต้องทำการตั้งค่าเพื่อการทำงาน โดยจะเกี่ยวกับพอร์ตที่ใช้ โปรโตคอลในการสื่อสาร และการเข้ารหัสข้อมูล อย่างไรก็ตามการใช้ **RAS** ก็ไม่ใช่ทางเลือกที่ดีที่สุดในการขยายระบบเครือข่าย แต่เป็นเพียงเพิ่มช่องทางในการติดต่อเข้ามายังระบบเครือข่ายจากระยะไกล ดังนั้นสิ่งจำเป็นที่ควรทราบคือ เมื่อใดควรใช้บริการ **RAS** และเมื่อใดควรใช้บริการอย่างอื่น การใช้ **RAS** จะเป็นสิ่งจำเป็นในกรณีที่มี

ช่องสัญญาณกว้างน้อยกว่า **128 Kbps** และไม่ต้องการที่จะติดต่อกับระบบเครือข่ายตลอดเวลา และอย่าเลือกใช้ **RAS** ถ้าต้องการแบนด์วิธมาก ที่ต้องใช้ **Synchronous Modem**

สำหรับผู้ใช้ที่ต้องการติดต่อเข้ามายังระบบเครือข่ายผ่านบริการ **RAS** จะต้องรับผิดชอบในการตั้งค่า **Dial-up Network** บนเครื่องคอมพิวเตอร์ของตนเอง ซึ่งจะเกี่ยวข้องกับหมายเลขโทรศัพท์ที่จะติดต่อเข้ามา พอร์ตการสื่อสาร ชื่อผู้ใช้ และรหัสผ่าน ดังแสดงตามรูปที่ **10 – 19**

รูปที่ **10 – 19** การตั้งค่าหมายเลขโทรศัพท์ใน **Dial-up Networking**

แบบฝึกหัดท้ายบท

1. เพราะเหตุใดโมเด็ม **56K** จึงสามารถส่งข้อมูลผ่านสายโทรศัพท์ **33.6 K** ได้ด้วยความเร็ว **56 Kbps**
2. ประสิทธิภาพของโมเด็มวัดจากอะไร มีการกำหนดมาตรฐานไว้อย่างไรบ้าง
3. **Cable Modem** คืออะไร มีการทำงานอย่างไร
4. จงอธิบายการทำงานของโมเด็ม **ISDN**
5. **ADSL** คืออะไร มีประโยชน์อย่างไรในการขนส่งข้อมูลดิจิตอล
6. จงอธิบายความแตกต่างระหว่างการขนส่งข้อมูลแบบ **Asynchronous** กับแบบ **Synchronous**
7. **Parity bit** คืออะไร มีประโยชน์อย่างไรต่อการขนส่งแพ็กเก็ตข้อมูล
8. ในการเชื่อมต่อฮับหลายๆ ตัวเข้าด้วยกันเพื่อขยายขนาดของระบบเครือข่ายจะต้องใช้สายชนิดใด
9. เมื่อใดจึงควรพิจารณาใช้รีพีตเตอร์ในการขยายขนาดของระบบเครือข่าย
10. จงอธิบายความแตกต่างระหว่างการใช้ฮับกับรีพีตเตอร์ในการขยายขนาดของระบบเครือข่าย
11. จงอธิบายความแตกต่างระหว่างการใช้บริดจ์กับรีพีตเตอร์ในการขยายขนาดของระบบเครือข่าย
12. จงอธิบายความแตกต่างระหว่างการใช้บริดจ์กับเราท์เตอร์ในการขยายขนาดของระบบเครือข่าย
13. สภาวะคอขวด (**bottleneck**) คืออะไร และจะเกิดขึ้นเมื่อใด
14. **Route Table** คืออะไร มีความสำคัญอย่างไรต่ออุปกรณ์ระบบเครือข่าย
15. **Routable Protocol** คืออะไร ให้ยกตัวอย่างโปรโตคอลประเภทนี้มา **5** ตัว
16. เราท์เตอร์แบ่งออกเป็นกี่ชนิด อะไรบ้าง จงอธิบาย
17. จงอธิบายการทำงานของเกตเวย์ (**Gateway**) ของระบบเครือข่าย
18. มีโปรโตคอลที่สนับสนุนการเชื่อมต่อระยะไกล (**Remote Access**) อยู่กี่ชนิด อะไรบ้าง
19. **RAS Server** คืออะไร มีหน้าที่อย่างไรในการขยายการเชื่อมต่อระบบเครือข่ายเป็น **WAN**
20. จงอธิบายการตั้งค่า **Dial-up Networking** ใน **Windows XP** มาพอสังเขป