

## วิชา Data Communication Laboratory

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

## การทดลองที่ 2 Serial Communication

## วัตถุประสงค์

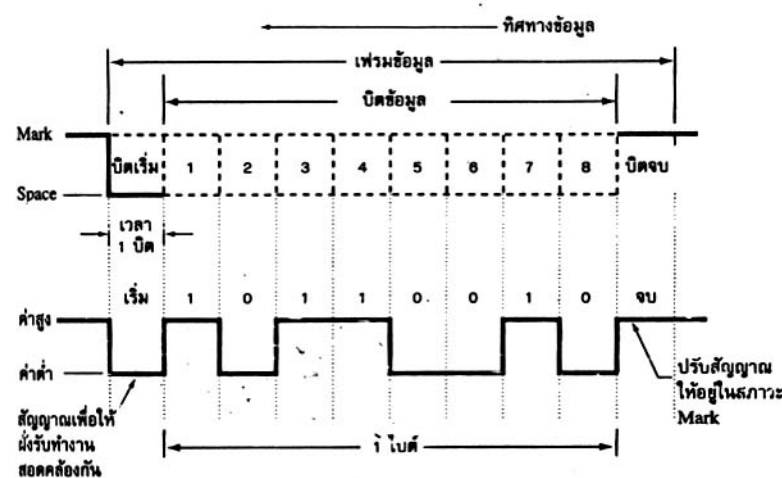
1. เพื่อให้เข้าใจหลักการสื่อสารผ่านพอร์ตอนุกรม
2. เพื่อให้เข้าใจการกำหนดค่าการคุณสมบัติการเชื่อมต่อพอร์ตอนุกรม
3. สามารถใช้โปรแกรมเพื่อติดต่อสื่อสารระหว่างคอมพิวเตอร์ผ่านพอร์ตอนุกรมได้

## ทฤษฎี

สำหรับการส่งข้อมูลอนุกรมแบบอะซิงโครนัส โดยใช้ชิพ UART (Universal Asynchronous Receiver/Transmitter) มักใช้ IC 8250 หรือ 16550 ซึ่งเป็นชิพที่ทำหน้าที่แปลงข้อมูลแบบขนานจากระบบไมโครโปรเซสเซอร์ให้เป็นข้อมูลแบบอนุกรมส่งออกไปยังสายสื่อสาร และในทางกลับกัน UART ก็ทำหน้าที่แปลงข้อมูลแบบอนุกรมที่มาถึงให้เป็นข้อมูลแบบขนานให้กับระบบไมโครโปรเซสเซอร์

## การรับส่งข้อมูลแบบอนุกรมอะซิงโครนัส

โดยปกติการสื่อสารแบบอะซิงโครนัสสถานะของสัญญาณในสายส่งเมื่อไม่มีข้อมูลจะมีสถานะเป็น High และในการสื่อสารผ่านพอร์ตอนุกรมจะทำการส่งข้อมูลที่ละไบต์ โดยมีบิตเริ่มต้น (Start Bit) แสดงถึงการเริ่มสถานะของการส่งของข้อมูลไบต์ ซึ่งสถานะของสายส่งมีค่าเป็น Low จากนั้นจะเป็นบิตของข้อมูล (บิตข้อมูลที่มีค่าเป็น 0 สถานะของสัญญาณมีค่าเป็น Low ส่วนบิตข้อมูลที่มีค่าเป็น 1 สถานะของสัญญาณมีค่าเป็น High) ตามด้วยพริตบิต (ขึ้นอยู่กับข้อกำหนดค่าการทำงาน) สุดท้ายคือบิตสิ้นสุด (Stop Bit) (1 หรือ 2 บิตขึ้นอยู่กับข้อกำหนดค่าการทำงาน) ดังรูปที่ 2.1

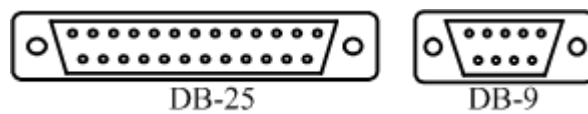


รูปที่ 2.1 เฟรมการส่งข้อมูลแบบอะซิงโครนัส

### มาตรฐาน RS232

ไมโครคอมพิวเตอร์จะมีพอร์ตอนุกรมที่เรียกว่า RS 232 อยู่ในตัว โดยพอร์ตนี้นำหน้าที่รับ และส่งข้อมูล ในแบบอนุกรมเรียกว่า Universal Asynchronous Adapter ตามมาตรฐาน RS 232 ซึ่งจัดพิมพ์ขึ้นเมื่อปี ค.ศ.1969 โดยที่ RS ย่อมาจาก Recommend Standard ส่วน 232 เป็นหมายเลขบังคับของมาตรฐาน ส่วน C เป็นหมายเลขท้ายสุดของมาตรฐานฉบับนี้ มาตรฐานนี้กำหนดคุณลักษณะของการเชื่อมต่ออุปกรณ์รับส่งข้อมูลปลายทาง (Data Terminal Equipment: DTE) กับอุปกรณ์สื่อสารข้อมูล (Data Communication Equipment : DCE) ซึ่งจะขึ้นอยู่กับผู้ผลิต

ความเร็วและระยะทางของการเชื่อมต่อ RS232 สามารถเชื่อมต่อการถ่ายโอนข้อมูลได้ประมาณ 0-20,000 บิตต่อวินาที ส่วนความยาวของสายเชื่อมต่อสัญญาณตามมาตรฐานของ RS232 จำกัดอยู่ที่ 50 ฟุต พอร์ตอนุกรมส่วนใหญ่จะมีรูปร่างขึ้นอยู่กับมาตรฐานของ RS-232 คือมีขาคอนเนคเตอร์ทั้งแบบ 25 ขาและแบบ 9 ขาแสดงดังรูปที่ 2.2 และขาคอนเนคเตอร์สำหรับใช้สัญญาณพื้นฐานของ RS-232 กำหนดดังตารางที่ 2.1



รูปที่ 2.2 DB25 Connector และ DB9 Connector

ตารางที่ 2.1 D Type 9 Pin and D Type 25 Pin Connectors

Common name	Description	Pin numbers 25-pin connector	Pin numbers 9-pin connector
TxD	Transmit Data	2	3
RxD	Receive Data	3	2
RTS	Request To Send	4	7
CTS	Clear To Send	5	8
DSR	Data Set Ready	6	6
SG	Signal Ground	7	5
CD	Carrier Detect	8	1
DTR	Data terminal ready	20	4
RI	Ring Indicator	22	9

#### คำอธิบายหน้าที่ของแต่ละขา

**Transmit Data (TxD)** เป็นสัญญาณที่ส่งออกจาก DTE (หรือไมโครคอมพิวเตอร์) ไปยังโมเด็มหรือต่อเข้าโดยตรงกับไมโครคอมพิวเตอร์ ข้อมูลแบบอนุกรมจะถูกส่งออกจากคอมพิวเตอร์ด้วยขานี้ สถานะของขานี้จะมีค่าเท่ากับ “1” หรือเทียบเท่ากับบิตหยุด

**Receive Data (RxD)** เป็นทางของสัญญาณเข้าไปยัง DTE หรือ ไมโครคอมพิวเตอร์ข้อมูลแบบอนุกรมจะรับเข้าเครื่องคอมพิวเตอร์ด้วยขานี้ เมื่อไม่มีสัญญาณเข้ามาขานี้จะมีสถานะทางลอจิกเป็น “1”

**Request to Send (RTS)** สัญญาณที่ขานี้จะเป็นส่วนที่บอกโมเด็มว่า UART พร้อมที่จะส่งข้อมูล ซึ่งขานี้ใช้สำหรับส่งสัญญาณไปยังโมเด็มหรือเครื่องพิมพ์ เป็นการร้องขอที่จะส่งสัญญาณมาทางขา 2 สัญญาณนี้ใช้คู่กับ CTS (Clear to Send) อุปกรณ์รับหากได้สัญญาณ RTS จะตรวจสอบตัวเองว่าพร้อมจะรับสัญญาณได้หรือยัง หากพร้อมที่จะรับก็จะส่งสัญญาณออกไปที่ขา CTS

**Clear to Send (CTS)** สัญญาณที่ขานี้จะเป็นส่วนที่แสดงว่าโมเด็มพร้อมที่จะส่งข้อมูล เมื่อสัญญาณนี้อยู่ในสถานะออฟ (แรงดันมีค่าเป็นลบ หรือ ลอจิก 1) ซึ่งหมายความว่าอุปกรณ์พร้อมที่จะรับข้อมูล

**Data Set Ready (DSR)** สัญญาณที่ขานี้จะเป็นตัวบอก UART ว่าโมเด็มพร้อมที่จะทำการเชื่อมต่อ เมื่อสัญญาณนี้อยู่ในสถานะออน (ลอจิก 0) เป็นการบอกโมโครคอมพิวเตอร์หรือฝ่ายส่งว่าโมเด็มต่อเข้ากับสายโทรศัพท์เรียบร้อยแล้วและพร้อมที่จะส่งได้แล้ว โมเด็มที่มีการหมุนหมายเลขอัตโนมัติจะส่งสัญญาณนี้ออกไปบอกให้คอมพิวเตอร์รู้ว่าต่อโทรศัพท์ได้สำเร็จแล้ว

**Signal Ground (SG)** ขากราวด์ทำหน้าที่เป็นระดับแรงดันอ้างอิงสำหรับทุกๆ สัญญาณ

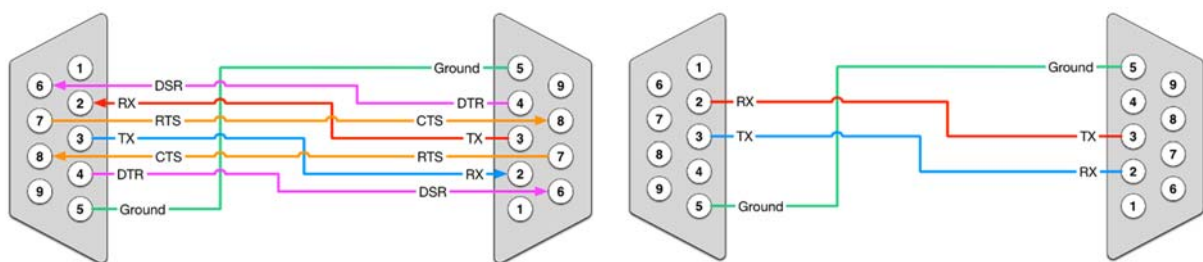
**Carrier Detect (CD)** โมเด็มจะทำการส่งสัญญาณนี้ให้กับเครื่องคอมพิวเตอร์เมื่อได้รับ สัญญาณ carrier จากโมเด็มปลายทางอีกฝั่งหนึ่ง

**Data Terminal Ready (DTR)** สัญญาณดังกล่าวจะตรงกันข้ามกับสัญญาณ DSR นั่นคือสัญญาณที่ขานี้จะเป็นตัวบอกโมเด็มว่า UART พร้อมที่จะทำการเชื่อมต่อ

**Ring Indicator (RI)** จะทำงานเมื่อโมเด็มได้รับสัญญาณ Ringing จากโครงข่าย PSTN สัญญาณทั้งหมดนี้

การรับส่งข้อมูลแบบอนุกรมอะซิงโครนัสกับอุปกรณ์ชนิดเดียวกัน

เนื่องจากอุปกรณ์ชนิดเดียวกันย่อมใช้ขาคอนเนคเตอร์ที่เหมือนกัน ดังนั้นหากต้องการการรับส่งข้อมูลระหว่างอุปกรณ์ชนิดเดียวกันจำเป็นต้องใช้สายสัญญาณที่เป็นแบบครอส (Crossover | Null Modem) ซึ่งจะเชื่อมต่อขาคอนเนคเตอร์ที่ทำหน้าที่ตรงข้ามกัน เช่น สลับระหว่าง TxD กับ RxD หรือ สลับระหว่าง RTS กับ CTS ดังรูปที่ 2.3 ก) โดยสายสัญญาณแบบครอสที่ง่ายที่สุดเป็นดังรูป 2.3 ข) (Decisive Tactics. *Crossover or "Null Modem" vs. Straight Through Serial Cable*. Retrieved January 22, 2017, Web site: <https://www.decisivetactics.com/support/view?article=crossover-or-null-modem-vs-straight-through-serial-cable>)



ก) Null Modem Cable with Handshake

ข) Simple Null Modem Cable

รูปที่ 2.3 สายสัญญาณที่เป็นแบบครอส (Crossover | Null Modem)

## การทดลองที่ 2.1 การส่งข้อมูลผ่านพอร์ตอนุกรมในเครื่องเดียวกัน

1. เชื่อมต่อพอร์ตอนุกรมในเครื่องเดียวกันด้วยสายสัญญาณที่เป็นแบบครอสดังรูปที่ 2.4

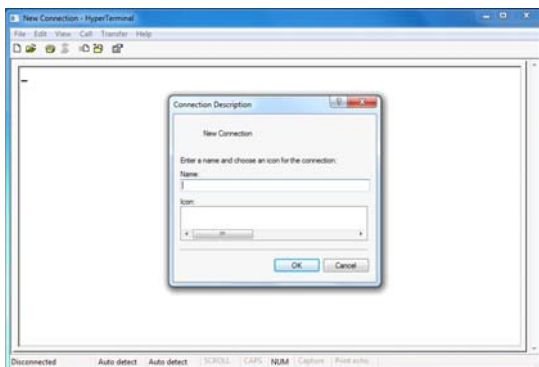


ก) พอร์ตอนุกรม

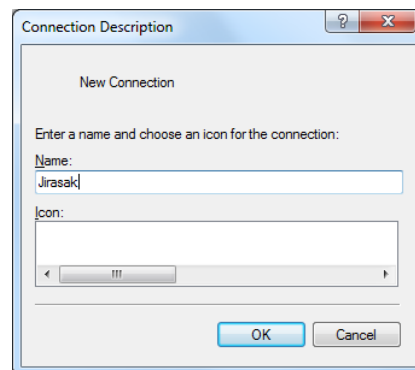
ข) เชื่อมต่อพอร์ตอนุกรมในเครื่องเดียว

รูปที่ 2.4 พอร์ตอนุกรมของเครื่องคอมพิวเตอร์ในห้องปฏิบัติการ

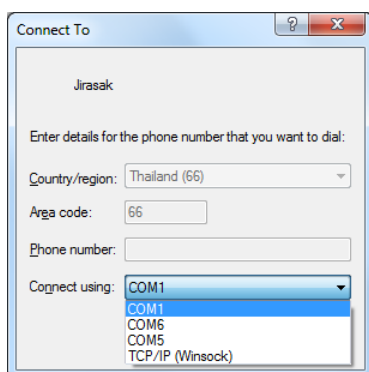
2. เปิดโปรแกรม HyperTerminal และกำหนดค่าดังรูปที่ 2.5 และตารางที่ 2.2 (รูปแบบ A)



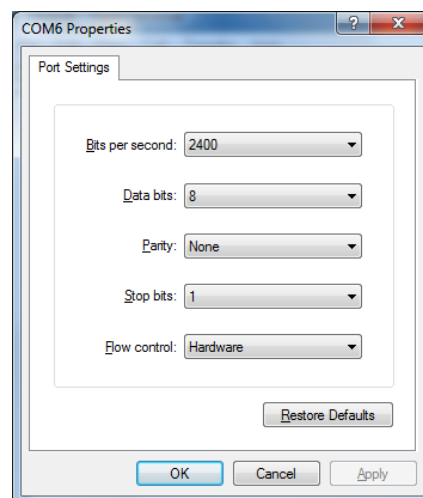
ก) โปรแกรม HyperTerminal



ข) กำหนดชื่อ Connection



ค) เลือก Com Port



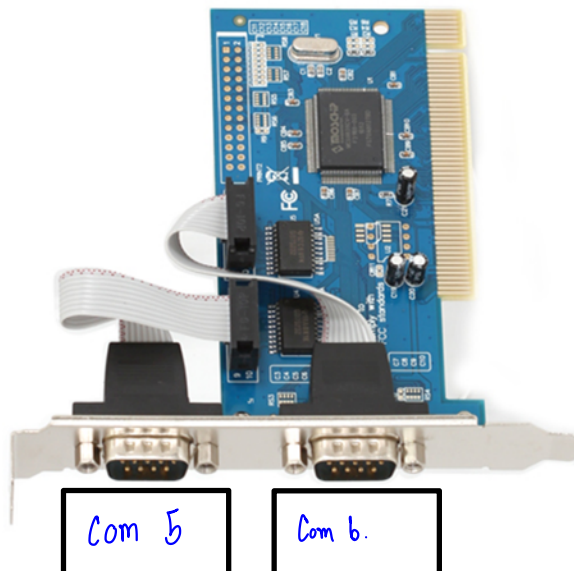
ง) กำหนดค่าคุณสมบัติการเชื่อมต่อ

รูปที่ 2.5 โปรแกรม HyperTerminal

## ตารางที่ 2.2 การกำหนดค่าคุณสมบัติการเชื่อมต่อพอร์ตอนุกรม

รูปแบบ/ Name	A	B	C	D	E	F	G
Connect using	Com 6	Com 5	Com 6	Com 6	Com 6	Com 6	Com 6
Bits per second	2400	2400	2400	2400	110	9600	115200
Data Bits	8	8	7	6	8	8	8
Paritty	None						
Stop Bits	1						
Flow Control	None						

- เปิดโปรแกรม HyperTerminal ใหม่ (เพิ่ม) และกำหนดค่าดังรูปที่ 2.5 และตารางที่ 2.2 (รูปแบบ B)
- ทดสอบส่งข้อความระหว่างหน้าต่างโปรแกรมในข้อ 2 และ ข้อ 3 (มี 2 หน้าต่างโปรแกรม)
- ให้ระบุหมายเลข Com Port ที่ใช้ในรูปที่ 2.6



รูปที่ 2.6 PCI to Serial 2 Port Card

- อธิบายลักษณะการส่งข้อมูลผ่านพอร์ตอนุกรมในเครื่องเดียวกันด้วยโปรแกรม HyperTerminal (รูปแบบ A-B)

การส่งข้อมูลโดยพิมพ์ ข้อมูลใน A จะไปต้นขงผู้รับ B ส่วนการส่งข้อมูลใน B จะไปเก็บขงผู้รับก็ A  
แสดงว่า A และ B เป็นได้ทั้งผู้ส่งและผู้รับ (Asynchronous balanced mode)

- หากต้องการส่งข้อมูลผ่านพอร์ตอนุกรมในเครื่องเดียวกันโดยใช้พอร์ตอนุกรมเพียง 1 พอร์ต ต้องทำอย่างไร?

ตั้ง Port เป็น loopback เพื่อให้งส่งข้อมูลผ่าน Port เองวนเองกันได้

## การทดลองที่ 2.2 การรับ-ส่งข้อมูลผ่านพอร์ทอนุกรมระหว่างเครื่องคอมพิวเตอร์ด้วยโปรแกรม HyperTerminal

1. เชื่อมต่อพอร์ทอนุกรมระหว่างเครื่องคอมพิวเตอร์ด้วยสายสัญญาณที่เป็นแบบครอส
2. เปิดโปรแกรม HyperTerminal และกำหนดค่าดังรูปที่ 2.5 และตารางที่ 2.2 (รูปแบบ A-A) \*
3. ให้ทดสอบการส่งข้อความระหว่างเครื่องคอมพิวเตอร์ หากไม่สามารถเชื่อมต่อได้ให้สลับสายที่พอร์ทอนุกรมให้ถูกต้อง
4. ผลการในการทดลองที่ 2.1 ข้อ 5 ถูกต้องหรือไม่ เพราะเหตุใด

ถูกต้อง เพราะว่า การทำงานของโปรแกรมปกติ

5. ทดสอบการส่งข้อความระหว่างเครื่องคอมพิวเตอร์ พร้อมอธิบายผลการทดลอง

ผลการทดลองไม่เปลี่ยนแปลง

6. เปลี่ยนการเชื่อมต่อที่ HyperTerminal ใหม่โดย กำหนดค่าดังรูปที่ 2.5 และตารางที่ 2.2 (รูปแบบ A-C) \*

7. ทดสอบการส่งข้อความระหว่างเครื่องคอมพิวเตอร์ พร้อมอธิบายผลการทดลอง

ผลการทดลองของตัวอักษรที่อยู่ในขอบเขต 0-127 ของ Ascii table จะเปลี่ยนแปลงเวลาส่งข้อมูล ส่วนตัวอักษรที่อยู่นอก Ascii table จะเปลี่ยนเมื่อส่งจาก A ไป C

8. เปลี่ยนการเชื่อมต่อที่ HyperTerminal ใหม่โดย กำหนดค่าดังรูปที่ 2.5 และตารางที่ 2.2 (รูปแบบ A-D) \*

9. ทดสอบการส่งข้อความระหว่างเครื่องคอมพิวเตอร์ พร้อมอธิบายผลการทดลอง

ผลการทดลองของตัวอักษรที่อยู่ในขอบเขต 0-63 พยายามจะไม่เปลี่ยนแปลงแต่ได้ส่งจาก A-D โดยตัวอักษรที่อยู่ในขอบเขต เช่น 2 คือ 01111010 จะตัดเหลือ 6 bits คือ 111010 คือ 58 (:)

10. วิเคราะห์ผลการทดลองข้อ 5 7 และ 9

เมื่อส่งข้อมูลตามข้อ 5 ที่มี data bit เท่ากัน ข้อมูลจะไม่เปลี่ยนแปลง ถ้าส่งข้อมูลตามข้อ 6 ที่ส่งจาก data bit 9 ไปยัง data bit 7. การส่งข้อมูลที่เป็นตัวอักษรในขอบเขต Ascii table จะไม่เปลี่ยนแปลง แต่ถ้าเกินขอบเขต เช่น unicode จะผลในตัวอักษร บางตัวเปลี่ยนไป ถ้าส่งข้อมูลตามข้อ 7 Ascii code ที่เกินขอบเขต 0-63 จะเพิ่มเพราะเกิดการตัด bit นี้ออก 2 ตัว ได้ส่งจาก A → D

\* เป็นรูปแบบการกำหนดค่าคุณสมบัติการเชื่อมต่อผ่านพอร์ทอนุกรมระหว่างเครื่องคอมพิวเตอร์ 2 เครื่อง

11. สร้างไฟล์ข้อความด้วยโปรแกรม Notepad โดยให้มีตัวอักษรอย่างน้อย 500 ตัว
12. เปลี่ยนการเชื่อมต่อที่ HyperTerminal ใหม่โดย กำหนดค่าดังรูปที่ 2.5 และตารางที่ 2.2 (รูปแบบ E-E) \*
13. ทดสอบการส่งไฟล์ระหว่างเครื่องคอมพิวเตอร์ พร้อมอธิบายผลการทดลอง

สามารถส่งไฟล์ได้ปกติ

14. เปลี่ยนการเชื่อมต่อที่ HyperTerminal ใหม่โดย กำหนดค่าดังรูปที่ 2.5 และตารางที่ 2.2 (รูปแบบ F-F) \*
15. ทดสอบการส่งไฟล์ระหว่างเครื่องคอมพิวเตอร์ พร้อมอธิบายผลการทดลอง

สามารถส่งไฟล์ได้

16. เปลี่ยนการเชื่อมต่อที่ HyperTerminal ใหม่โดย กำหนดค่าดังรูปที่ 2.5 และตารางที่ 2.2 (รูปแบบ G-G) \*
17. ทดสอบการส่งไฟล์ระหว่างเครื่องคอมพิวเตอร์ พร้อมอธิบายผลการทดลอง

สามารถส่งไฟล์ได้ปกติ

18. วิเคราะห์ผลการทดลองข้อ 13 15 และ 17

จากการทดลอง ความเร็วในการรับส่งข้อมูลจะขึ้นอยู่กับค่าที่เราตั้ง bits per second โดย 1000 bits/second จะใช้เวลานานที่สุด ทด: 10000 bits/second จะใช้เวลาน้อยที่สุด

สรุปได้ว่า ยิ่ง bits/second สูงๆ ความเร็วในการรับส่งข้อมูลก็จะมากขึ้นด้วย

### การทดลองที่ 2.3 การรับ-ส่งข้อมูลผ่านพอร์ทอนุกรมระหว่างเครื่องคอมพิวเตอร์ด้วยโปรแกรมที่แตกต่างกัน

1. เชื่อมต่อพอร์ทอนุกรมระหว่างเครื่องคอมพิวเตอร์ด้วยสายสัญญาณที่เป็นแบบครอส
  - 1.1. เครื่องแรกให้ใช้โปรแกรม HyperTerminal ในการรับ-ส่งข้อมูล
  - 1.2. เครื่องที่สองให้ใช้โปรแกรมอื่นๆ ในการรับ-ส่งข้อมูล
2. โปรแกรมใดบ้างที่สามารถรับ-ส่งข้อมูลผ่านพอร์ทอนุกรมระหว่างเครื่องคอมพิวเตอร์ได้ (ตอบในตารางที่ 2.3)
3. ทดสอบว่าโปรแกรมที่สามารถรับ-ส่งข้อมูลผ่านพอร์ทอนุกรม รองรับคุณสมบัติการเชื่อมต่อใดบ้าง (ตอบในตารางที่ 2.3)

ตารางที่ 2.3 ตารางสรุปโปรแกรมที่สามารถการรับ-ส่งข้อมูลผ่านพอร์ทอนุกรมระหว่างเครื่องคอมพิวเตอร์กับโปรแกรม HyperTerminal

โปรแกรม	คุณสมบัติการเชื่อมต่อที่รองรับ	หมายเหตุ
Tera Term	<ul style="list-style-type: none"> <li>- เลือก Port เป็นคู่ๆตามเส้นพวงแดงก็ได้</li> <li>- data bit เลือกได้แค่ 7-8</li> <li>- Speed ตามที่เว็บถามค่า</li> <li>- BPS/parity /stop ฯลฯ</li> </ul>	
PuTTY	<ul style="list-style-type: none"> <li>- Speed</li> <li>- data bits</li> <li>- BPS / Parity / stop / port</li> </ul>	
KITTY	ค่อนข้างคล้ายกับ putty แต่มีลูกเล่นมากกว่า เช่น หน้า UI เปลี่ยน	
SmartTY	รองรับการคัดลอกไฟล์และไจเรกทอรีด้วย SCP ได้ทันที และแก้ไขไฟล์ในสถานที่	