บทที่ 5

การสื่อสารข้อมูลแบบอนุกรม (Serial Communication)

5.1) การส่งข้อมูลแบบอนุกรมและแบบขนาน

โดยอุดมคติแล้วการแลกเปลี่ยนข้อมูลกันระหว่างคอมพิวเตอร์กับคอมพิวเตอร์ หรือระหว่างคอมพิวเตอร์กับอุปกรณ์โดยรอบ (อย่างเช่น เครื่องพิมพ์) จะทำในลักษณะเป็นหน่วยแปดบิตหรือหนึ่งไบต์ต่อครั้ง การแลกเปลี่ยนข้อมูลดังกล่าว เรียกว่า การส่งข้อมูลแบบขนาน เนื่องจากส่งทั้งแปดบิตไปในขณะเดียวกันผ่านสายที่ขนานกันแปดสาย อย่างไรก็ดี ในสายโทรศัพท์จะทำการส่งข้อมูลแบบขนานไม่ได้เพราะสายโทรศัพท์มีเพียงสายเดียวในการส่งข้อมูลไปในทิศทางหนึ่ง และอีกสายหนึ่งในการส่งข้อมูลไปในอีกทิศทางหนึ่ง ดังนั้นในการส่งข้อมูลของคอมพิวเตอร์ไปทางสายโทรศัพท์ (ซึ่งเป็นวิธีที่ใช้ติดต่อกับคอมพิวเตอร์เครื่องอื่นเป็นส่วนใหญ่)จึงต้องมีการปรับปรุงบ้างเล็กน้อย โดยแต่ละไบต์จะต้องแตกออกเป็นบิตย่อยๆ และส่งบิตเหล่านั้นไปครั้งละหนึ่งบิต วิธีการส่งข้อมูลแบบนี้เรียกว่า การส่งข้อมูลแบบอนุกรม

ความยากในการส่งข้อมูลทีละบิตที่อาจจะมีอยู่นั้นมองเห็นได้ชัดเจนเพราะว่าในการส่งข้อมูลแบบขนาน ไบต์ใดเป็นไบต์เริ่มต้นและไบต์ใดเป็นไบต์จบนั้นทราบได้แน่ชัด (ทั้งหมดมีหน่วยเป็นไบต์) แต่การส่งข้อมูลแบบอนุกรมมักจะเสี่ยงต่อการที่คอมพิวเตอร์รับข้อมูลไม่ทราบว่าไบต์หนึ่งจบที่ไหน และอีกไบต์หนึ่งเริ่มต้นที่ไหน และหากมีการสูญหายไปแม้แต่บิตเดียวเนื่องจากความบกพร่องของสายโทรศัพท์แล้ว จะเป็นผลให้ข้อมูลที่ได้รับหลังจากนั้นไปกลายเป็นขยะที่อ่านไม่รู้เรื่องได้

5.2) ลักษณะของฮาร์ดแวร์ (Hardware Specification)

มาตรฐาน RS-232C

เพื่อที่จะทำให้อุปกรณ์จากผู้ผลิตต่างกันทำงานร่วมกันได้ มาตรฐานหลายชนิดจึงได้รับการออกแบบขึ้น โดยมาตรฐานที่ใช้กันอย่างกว้างขวางที่สุดคือ RS-232C ซึ่งถูกประกาศขึ้นในปี 1969 โดย Electronic Industries Association มาตรฐาน RS-232C ที่ร่างขึ้นในตอนเริ่มแรกสำหรับกำหนดการเชื่อมต่อระหว่างเทอร์มินัล (terminal) และโมเด็ม ระบุคุณลักษณะทางไฟฟ้าของวงจรระหว่างอุปกรณ์สองตัว และกำหนดชื่อและหมายเลขแก่สายที่จำเป็นสำหรับการเชื่อมต่อวงจร ชื่อวงจรตามมาตรฐาน RS-232C (AA, AB เป็นต้น) จำได้ยากในทางปฏิบัติจึงใช้ชื่อย่อแทน

ตัวอย่างเช่น สายเส้นที่ 2 มีชื่อ BA แต่ใช้กันทั่วไปว่า TXD (Transmitted Data) ตามมาตรฐาน RS-232C สายเส้นที่ 2 จะนำข้อมูลจากเทอร์มินัลไปสู่โมเด็ม เพื่อให้การทำงานถูกต้องเทอร์มินัลต้องส่งเอาต์พุตออกที่สายเส้นที่ 2 และโมเด็มต้องรับข้อมูลบนสายเส้นที่ 2 เพราะฉะนั้นสายเส้นที่ 2 เป็นสายส่งข้อมูลสำหรับอุปกรณ์บางอย่าง และเป็นสายรับข้อมูลสำหรับอุปกรณ์อย่างอื่น การเชื่อมต่อโดยตรงจากสายเส้นที่ 2 บนอุปกรณ์หนึ่งเข้ากับสายเส้นที่ 2 บนอุปกรณ์อีกตัวหนึ่งสามารถทำได้ต่อเมื่ออุปกรณ์หนึ่งส่งข้อมูลบนสายเส้นที่ 2 และอีกตัวหนึ่งรับข้อมูลบนสายเส้นที่ 2 เพื่อป้องกันไม่ให้อุปกรณ์ส่งข้อมูลบนสายเส้นเดียวกัน อุปกรณ์จึงถูกแบ่งออกเป็นสองชนิด อุปกรณ์อย่างเช่นเทอร์มินัลซึ่งใช้สายเส้นที่ 2 สำหรับเอาต์พุต เรียกว่า DTE (Data Terminal Equipment) อุปกรณ์อย่างเช่นโมเด็มซึ่งใช้สายเส้นที่ 2 สำหรับอินพุต เรียกว่า DCE (Data Communication Equipment)

5.3) ประเภทของการสื่อสารข้อมูลแบบอนุกรม

5.3.1) การสื่อสารทางเดียว

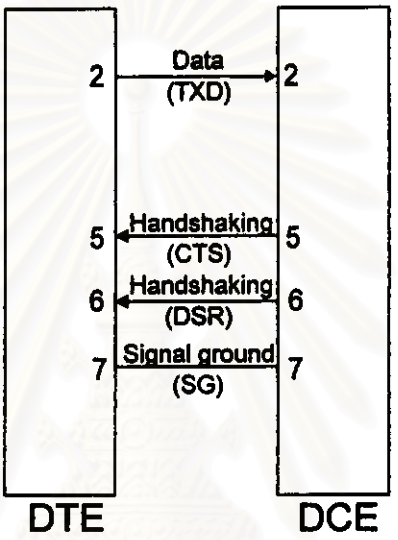
วงจรหลักที่ถูกใช้สำหรับการสื่อสารมีอยู่สามวงจร คือสายเส้นที่ 2 สำหรับข้อมูลจาก DTE ไปยัง DCE สายเส้นที่ 3 สำหรับข้อมูลจาก DCE ไปยัง DTE และสายเส้นที่ 7 สำหรับซิกแนลกราวนด์ (signal ground) ซึ่งเป็นจุดอ้างอิงร่วมสำหรับขั้วและแรงดันไฟฟ้าของสายอื่น ในกรณีที่ง่ายที่สุดซึ่งมีเพียงอุปกรณ์หนึ่งส่งและอีกตัวรับใช้สายเพียงสองเส้นก็เพียงพอ คือสายเส้นที่ 2 หรือ 3 และสายเส้นที่ 7 สำหรับเป็น signal ground

<u>จาก DTE ไปสู่ DCE</u>

เมื่ออุปกรณ์ DTE ส่งข้อมูลไปที่อุปกรณ์ DCE ข้อมูลถูกส่งไปตามสายเส้นที่ 2 และใช้สายเส้นที่ 7 เป็นซิกแนลกราวนด์ ตามปกติอุปกรณ์ DCE ควบคุมการส่งแฮนด์เช็คกิ้งจากอุปกรณ์ DTE บนสายเส้นที่ 6 ชื่อว่า DSR (Data Set Ready) ถ้าเครื่องพิมพ์เป็น DCE และคอมพิวเตอร์เป็น DTE สายเส้นที่ 6 บนคอมพิวเตอร์ควรถูกเชื่อมต่อเข้ากับสายเส้นที่ 6 บนเครื่องพิมพ์ และเครื่องพิมพ์จะรักษาแรงดันไฟฟ้าบวกบนสายเส้นที่ 6 ในขณะที่เครื่องพิมพ์กำลังรับข้อมูล และเมื่อต้องการที่จะบอกคอมพิวเตอร์ให้หยุดการส่งข้อมูล เครื่องพิมพ์จะลดแรงดันไฟฟ้าบนสายเส้นที่ 6 ให้เป็นสถานะลบ

บ่อยครั้งที่วงจรแฮนด์เช็คกิ้งชุดที่สอง คือสายเส้นที่ 5 ถูกใช้โดยอุปกรณ์ DCE เพื่อควบคุมการส่งจากอุปกรณ์ DTE วงจรนี้มีชื่อว่า CTS (Clear To Send) เมื่อสายแฮนด์เช็คกิ้งทั้งสองเส้นถูกใช้ อุปกรณ์ DTE ต้องได้รับการออกแบบให้ส่งข้อมูลก็ต่อเมื่อสายทั้งสองเป็นไฮ (high) หรือแรงดันไฟฟ้าบวก บางครั้งสายนั้นอาจมีความหมายต่างไป เช่นเส้นหนึ่งอาจบอก

อุปกรณ์ฝ่ายส่งให้หยุดการพิมพ์จนกระทั่งข้อมูลถูกพิมพ์ไปได้จำนวนหนึ่ง และเส้นที่เหลืออาจจะแจ้งว่ากระดาษของเครื่องพิมพ์หมด อย่างไรก็ตามความหมายเหล่านี้ไม่ได้มีมาตรฐาน แต่ถึงกระนั้น คอมพิวเตอร์หลายชนิดถูกโปรแกรมไม่ให้ส่งข้อมูล ถ้าสายแฮนด์เช็คกิ้งทั้งสองไม่เป็นไฮ แม้แต่กับเครื่องพิมพ์ที่ไม่ได้กำหนดความหมายพิเศษกับสายชุดที่สอง อย่างน้อยก็ควรรักษาแรงดันไฟฟ้าบวกไว้ อย่างไรก็ตามบางครั้งสัญญาณชุดที่สองต้องถูกสร้างหลอกขึ้นมาโดยการต่อมันเข้ากับชุดแรก

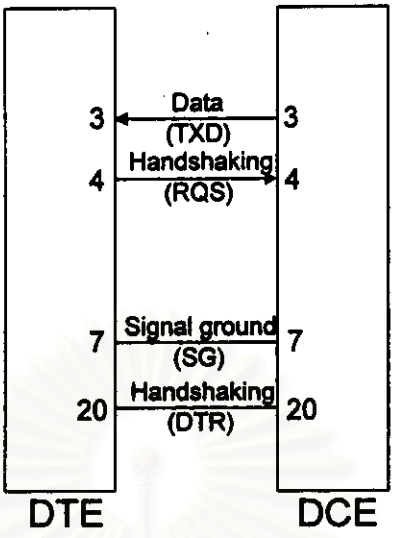


รูปที่ 5.1 แสดงการสื่อสารทางเดียวพร้อมด้วยแฮนด์เช็คกิ้งจาก DTE ไปยัง DCE

DCE ไปยัง DTE

เมื่อให้อุปกรณ์ DCE จะสื่อสารกับอุปกรณ์ DTE สายเส้นที่ 3 ต้องถูกใช้สำหรับการส่งผ่านข้อมูล และถ้าต้องการแฮนด์เช็คกิ้ง จะใช้สายเส้นที่ 20 เพื่อส่งแฮนด์เช็คกิ้งจากอุปกรณ์ DTE ไปยังอุปกรณ์ DCE สายเส้นที่ 20 มีชื่อว่า DTR (Data Terminal Ready) สายแฮนด์เช็คกิ้งชุดที่สองคือ สายเส้นที่ 4 เรียกว่า Request To Send (RQS หรือ RT1S4) ไม่ได้ถูกนำมาใช้เสมอไป

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



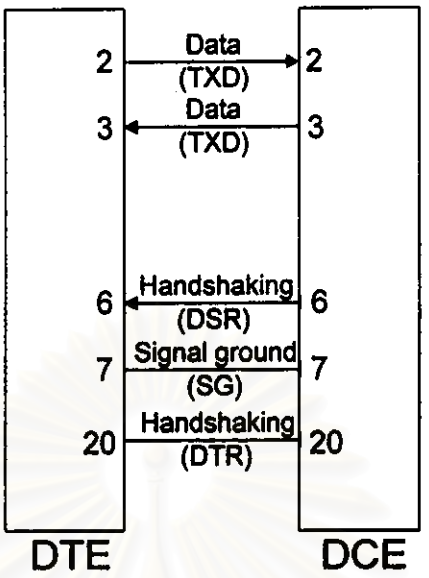
รูปที่ 5.2 แสดงการสื่อสารทางเดียวพร้อมด้วยแฮนด์เช็คกิ้งจาก DCE ไปยัง DTE

5.3.2) การสื่อสารสองทาง

ในหลายกรณีที่ข้อมูลถูกส่งผ่านในสองทิศทางโดยเฉพาะเมื่อคอมพิวเตอร์สองตัวสื่อสารกัน รวมทั้งในกรณีที่ใช้ซอฟต์แวร์แฮนด์เช็คกิ้งด้วยเช่นกัน จำนวนสายที่น้อยที่สุดที่จำเป็นในการสื่อสารสองทางคือ สามเส้น ได้แก่ สายข้อมูลในแต่ละทิศทาง และซิกแนลกราวด์ การเพิ่มเติมสายแฮนด์เช็คกิ้งในแต่ละทิศทาง ทำให้จำนวนสายรวมเป็นห้าเส้น ดังแสดงในรูปที่ 5.3

เมื่อสายแฮนด์เช็คกิ้งชุดที่สองถูกนำมาใช้เพิ่มเติมลงในแต่ละทิศทาง สายทั้งหมดที่ใช้คือเจ็ดเส้น บางครั้งอาจมีการเพิ่มสายอีกสองเส้น เพื่อทำให้โมเด็มสามารถให้ข้อมูลมากขึ้นแก่คอมพิวเตอร์หรือเทอร์มินัล ได้แก่ CD (Carrier Dectect) ซึ่งถูกเชื่อมต่อเข้ากับขา 8 เพื่อแจ้งการคงอยู่ของสัญญาณพาหะ และ RI (Ring Indicator) ถูกเชื่อมต่อกับขา 22 เพื่อแสดงว่าโมเด็มกำลังถูกเรียกโดยอุปกรณ์ระยะไกลซึ่งก็คือการตรวจสอบสัญญาณกริ่งของโทรศัพท์นั่นเอง จำนวนวงจรทั้งหมดจะกลายเป็นเก้าเส้น

แม้ว่ายังมีวงจรอื่นอีกหลายวงจรถูกกำหนดโดย RS-232C แต่ทั้งเก้าวงจรนี้ถูกใช้กันมากที่สุด และเป็นเพียงชุดเดียวที่โดยปกติถูกเชื่อมต่อกับไมโครคอมพิวเตอร์ ซึ่งเป็นเหตุผลที่ไมโครคอมพิวเตอร์ใช้หัวต่อ 9 ขาแทน 25 ขา สำหรับนำพาสัญญาณที่จำเป็นทั้งหมดของวงจร RS-232C

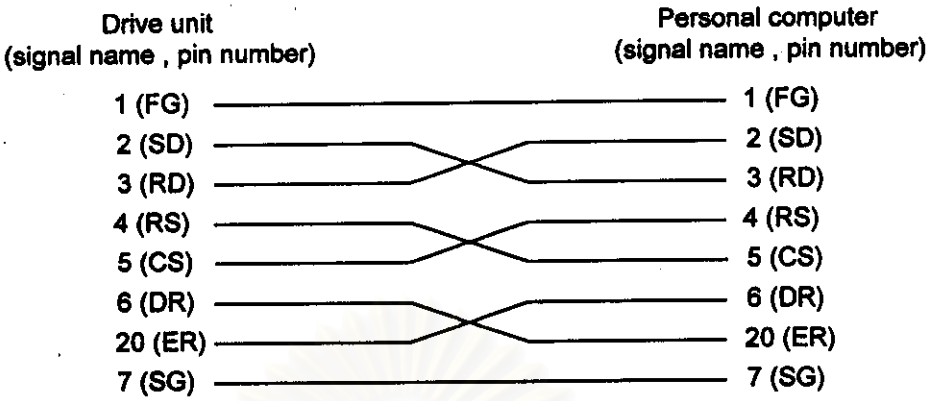


รูปที่ 5.3 แสดงการสื่อสารสองทางพร้อมด้วยวงจรแฮนด์เช็คกิ้งหลัก

5.3.3) นัลโมเด็ม

ดังที่ได้กล่าวมาก่อนหน้านี้แล้วว่าในตอนแรก RS-232C ถูกนำไปใช้สำหรับกำหนดการเชื่อมต่อระหว่างเทอร์มินัลซึ่งเป็น DTE กับโมเด็มซึ่งเป็น DCE ต่อมาได้มีการนำไปประยุกต์ใช้กับการเชื่อมต่อระหว่างอุปกรณ์อื่นอีกหลายชนิดที่ไม่ได้ถูกกำหนดให้เป็น DTE หรือ DCE อย่างเป็นกิจลักษณะ เช่น ไมโครคอมพิวเตอร์ หรือเครื่องพิมพ์

เนื่องจากไม่มีมาตรฐานที่ชี้ชัดว่าอุปกรณ์ควรเป็น DTE หรือ DCE บ่อยครั้งจึงจำเป็นต้องเชื่อมต่ออุปกรณ์ DTE สองตัวหรืออุปกรณ์ DCE สองตัวเข้าด้วยกัน ในลักษณะเช่นนี้ ต้องทำการเชื่อมต่อสาย 2 บนอุปกรณ์ตัวแรกกับสาย 3 บนอุปกรณ์ตัวที่สองและสาย 3 บนอุปกรณ์ตัวแรกกับสาย 2 บนตัวที่สอง สายแฮนด์เช็คกิ้งก็ต้องถูกไขว้ในทำนองเดียวกัน การไขว้สายอาจทำได้โดยการต่ออุปกรณ์ด้วยสายสัญญาณซึ่งถูกไขว้ไว้แล้ว หรือโดยการซื้อหัวต่อพิเศษที่เชื่อมต่ออุปกรณ์ทั้งสองและทำการไขว้สายที่จำเป็นไว้ภายใน ไม่ว่าในกรณีใดสายที่อยู่ตรงกลางหรือหัวต่อจะถูกเรียกว่า นัลโมเด็ม (null modem) มันทำให้อุปกรณ์ DTE สองตัวติดต่อกันได้ โดยไม่ต้องมีอุปกรณ์ DCE เป็นทางผ่าน หรือในทางกลับกันคือระหว่าง DCE กับ DTE การต่อสายสำหรับนัลโมเด็มแสดงไว้ในรูปที่ 5.4



รูปที่ 5.4 แสดงการเชื่อมต่อแบบนัลโมเด็ม

5.4) การตั้งค่าตัวเลือกทางการสื่อสาร (Communication Setting Parameter)

5.4.1) Baud rate

Baud rate คือความเร็วในการส่งบิตข้อมูล เป็นบิตข้อมูลต่อวินาที (ย่อว่า bps) หาได้โดยหารจำนวนบิตเดียวของข้อมูลที่ส่งด้วยระยะเวลาที่ใช้ในการส่งบิตข้อมูลนั้น ดังนั้น baud rate 300 จึงหมายความว่าสามารถส่งบิตเดียวได้ภายใน 0.0033 วินาที หรือส่งข้อมูลไป 300 บิตในหนึ่งวินาที ในทำนองเดียวกัน baud rate 1200 แสดงว่าส่งได้ 1200 bps เนื่องจากในการส่งข้อมูลคาแรกเตอร์หนึ่งตัวใช้ 10 บิต (8 บิตข้อมูล 1 บิตเริ่มต้น และอีก 1 บิตหยุด) ดังนั้น 1200 bps จึงคำนวณออกมาได้เป็น 120 คาแรกเตอร์ต่อวินาที (cps) และ 300 bps ได้ 30 cps

อัตราบิตต่อวินาทีโดยทั่วไปอยู่ในอนุกรม 110, 150, 300, 600, 1200, 2400, 4800, 9600 และ 19200 อัตราที่ใช้กันมากที่สุดสำหรับการสื่อสารทางโมเด็มคือ 1200 และ 2400 อัตรา 1200 ใช้กันมากสำหรับการสื่อสารระหว่างคอมพิวเตอร์กับเครื่องพิมพ์ และ 9600 ใช้กันมาก สำหรับการเชื่อมต่อเทอร์มินัลกับคอมพิวเตอร์

5.4.2) บิตเริ่มต้น

บิตเริ่มต้นถูกใส่เพิ่มที่จุดเริ่มต้นของเฟรมเสมอ เพื่อเตือนอุปกรณ์ฝ่ายรับว่าข้อมูลกำลังมาถึง และเพื่อเข้าจังหวะกลไกที่แยกแต่ละบิต บิตเริ่มต้นคือ SPACE หรือไบนารี 0

ในการเชื่อมต่อโดยตรง SPACE หรือ 0 ถูกส่งเป็นแรงดันไฟฟ้าบวก แรงดันไฟฟ้าระหว่างเฟรมจะเป็นลบ ดังนั้นที่จุดเริ่มต้นของแต่ละเฟรมแรงดันไฟฟ้าจะเปลี่ยนจากลบเป็นบวก

5.4.3) บิตข้อมูล

มาตรฐานหรือโปรโตคอลการสื่อสารแบบอนุกรม ทำให้เกิดการส่งตัวอักษรที่ยาวต่างกัน เมื่อซอฟต์แวร์สื่อสารให้เลือกความยาวเวิร์ด นั่นคือกำลังถามว่าต้องการส่งตัวอักษรเจ็ดบิตหรือแปดบิต (บางครั้งความยาวอื่นก็ถูกใช้แต่แทบไม่ค่อยมี) ถ้าข้อมูลทั้งหมดถูกส่งในรูปแบบ ASCII เวิร์ดขนาดเจ็ดบิตก็เพียงพอ เพราะว่าตาราง ASCII กำหนดจำนวนจาก 0 ถึง 127 ซึ่งทั้งหมดสามารถแทนได้ด้วยเจ็ดบิต

5.4.4) บิตพาริตี้ (Parity bit)

การตรวจสอบพาริตี้เป็นวิธีหนึ่งในการทดสอบว่า ข้อมูลที่ส่งได้ถูกรับไปอย่างถูกต้องหรือไม่ อุปกรณ์ฝ่ายส่งจะเพิ่มบิตพาริตี้อีกหนึ่งบิต เป็นค่า 0 หรือ 1 ขึ้นอยู่กับบิตข้อมูล อุปกรณ์ฝ่ายรับจะตรวจสอบว่าบิตพาริตี้มีความสัมพันธ์ที่ถูกต้องกับบิตอื่นหรือไม่ ถ้าไม่ แสดงว่าบางสิ่งต้องผิดพลาดในระหว่างการส่ง พาริตี้สามารถคำนวณได้จากวิธีต่อไปนี้

- พาริตี้คู่ (Even Parity) หมายความว่า จำนวนของบิตข้อมูลที่เป็น 1 และค่าของบิตพาริตี้รวมกันเป็นจำนวนคู่ เช่น ตัว A ในฐานสองคือ 01000001 เมื่อนับจำนวนของบิตที่เป็น 1 จะได้ 2 ซึ่งเป็นเลขคู่ ดังนั้นบิตพาริตี้ต้องเป็น 0 ถ้าตัวอักษร A ที่รับได้มีพาริตี้เป็น 1 แสดงว่าเกิดความผิดพลาดในระหว่างการส่ง

- พาริตี้คี่ (Odd Parity) หมายความว่า จำนวนทั้งหมดของบิตข้อมูลที่เป็น 1 บวกกับค่าของบิตพาริตี้ เป็นจำนวนคี่ ดังนั้นสำหรับตัวอักษร A บิตพาริตี้ควรถูกกำหนดเป็น 1 เพื่อให้จำนวนของบิตที่เป็น 1 ทั้งหมด เป็น 3 ซึ่งเป็นจำนวนคี่

- ไม่มีพาริตี้ (Null Parity) หมายถึงไม่มีบิตพาริตี้

- SPACE (บางครั้งเรียกว่า Bit Trimming) คือบิตพาริตี้ที่เป็น 0 เสมอ มีประโยชน์ในการตรวจสอบข้อผิดพลาดบางอย่าง เมื่อการส่งข้อมูลเป็นขยะมากบางครั้งบิต พาริตี้อาจกลายเป็น 1 แสดงว่าเกิดข้อผิดพลาด พาริตี้แบบนี้สามารถใช้เพื่อส่งอักษรเจ็ดบิตให้กับอุปกรณ์ที่ต้องการตัวอักษรแปดบิตได้เช่นกัน อุปกรณ์ฝ่ายรับจะถือว่าบิตพาริตี้เป็นบิตสุดท้ายของข้อมูล

- MARK (บางครั้งเรียกว่า Bit Forcing) ทำงานเหมือนกับพาริตี้แบบ SPACE ยกเว้นแต่บิตพาริตี้จะเป็น 1 เสมอ เนื่องจาก 1 ในตำแหน่งนั้นสามารถที่จะถูกตีความรวมเข้ากับค่าของจำนวนได้ อุปกรณ์ หรือคอมพิวเตอร์ฝ่ายรับต้องถูกโปรแกรมไม่ให้สนใจค่าบิตนี้

5.4.5) บิตจบ (End bit)

ที่ท้ายของแต่ละเฟรม บิตจบจะถูกส่งออกมา บิตจบมีทั้งแบบหนึ่งบิต หนึ่งบิตครึ่ง หรือสองบิต อย่างน้อยต้องมีหนึ่งบิตเสมอ เพื่อประกันว่ามีแรงดันไฟฟ้าลบอย่างน้อยเป็นช่วงเวลาหนึ่งก่อนที่เฟรมถัดไปจะมาถึง เพื่อที่จะสามารถแยกแยะเฟรมถัดไปได้จากบิตเริ่มต้น

ที่เป็นบวกของข้อมูล บิตจบมากกว่าหนึ่งบิตโดยทั่วไปจะใช้เมื่ออุปกรณ์ฝ่ายรับต้องการเวลาเพิ่มขึ้นก่อนที่มันจะสามารถจัดการกับตัวอักษรที่เข้ามาตัวถัดไปได้

หนึ่งบิตครึ่ง หมายความว่า ความยาวของบิตนั้นมากกว่าบิตปกติ บิตจบบังคับให้มีช่องว่างอย่างน้อยระหว่างเฟรม โดยบิตเหล่านี้จะถูกส่งเป็นไบนารีหนึ่ง ซึ่งในการเชื่อมต่อโดยตรงจะเป็นแรงดันไฟฟ้าลบ

บิตจบสองบิตมักจะถูกใช้ที่อัตราบอด 110 ซึ่งเป็นอัตราการส่งข้อมูลต่ำสุดที่ใช้กันทั่วไป เพื่อให้สอดคล้องกับความต้องการของเครื่องโทรพิมพ์รุ่นเก่า ซึ่งใช้ baud rate ต่ำและต้องการเวลาพิเศษเพื่อประมวลตัวอักษร

5.4.6) แฮนด์เช็คกิ้ง (Hand chaking)

ในหลายๆ กรณีอุปกรณ์ฝ่ายส่งจำเป็นต้องรู้ว่าอุปกรณ์ฝ่ายรับพร้อมที่จะรับข้อมูลหรือไม่ ตัวอย่าง เช่นการส่งข้อมูลไปที่เครื่องพิมพ์ ความเร็วของการสื่อสารอาจจะเร็วกว่าความเร็วของเครื่องพิมพ์ ดังนั้นเครื่องพิมพ์ต้องระงับการส่งข้อมูลของคอมพิวเตอร์จนกว่ามันพร้อมที่จะรับข้อมูล ในทำนองเดียวกันกับการส่งข้อมูลจากคอมพิวเตอร์เครื่องหนึ่งไปอีกเครื่องหนึ่ง และคอมพิวเตอร์ตัวที่สองไม่สามารถประมวลผลข้อมูลได้เร็วเท่ากับอัตราที่ข้อมูลเข้ามา

แฮนด์เช็คกิ้งเป็นวิธีที่อุปกรณ์ฝ่ายรับใช้ควบคุมการไหลของข้อมูล จากอุปกรณ์ฝ่ายส่ง บางครั้งเครื่องพิมพ์ไม่สามารถพิมพ์ตัวอักษรได้เร็วเท่ากับที่มันได้รับจากคอมพิวเตอร์ เครื่องพิมพ์จึงต้องใช้แฮนด์เช็คกิ้งเพื่อบอกให้คอมพิวเตอร์หยุดส่งชั่วคราว แฮนด์เช็คกิ้งมีประโยชน์เมื่อเครื่องพิมพ์กระดาษหมด หรือเมื่อคอมพิวเตอร์ส่งข้อมูลไปให้คอมพิวเตอร์อีกเครื่อง และคอมพิวเตอร์ฝ่ายรับไม่สามารถประมวลผลตัวอักษรได้เร็วเท่ากับที่พวกมันกำลังถูกส่งมา แต่เมื่อทราบว่าอุปกรณ์ฝ่ายรับสามารถประมวลผลตัวอักษรที่รับเข้ามาได้เร็วกว่าอัตราการส่งก็สามารถยกเลิกแฮนด์เช็คกิ้งได้

ทั้งสองกรณี ต้องมีข่าวสารถูกส่งกลับจากอุปกรณ์ฝ่ายรับไปยังอุปกรณ์ฝ่ายส่งเพื่อแจ้งว่ามันพร้อมหรือไม่ ข่าวสารนี้เรียกว่า โฟลว์คอนโทรล (flow control) หรือแฮนด์เช็คกิ้ง (hand chaking) แฮนด์เช็คกิ้งมีสองแบบคือ ฮาร์ดแวร์ และ ซอฟต์แวร์ ทั้งคู่เกี่ยวข้องกับสัญญาณที่ส่งกลับมาจากอุปกรณ์ฝ่ายรับไปยังอุปกรณ์ฝ่ายส่ง ในฮาร์ดแวร์แฮนด์เช็คกิ้งอุปกรณ์ฝ่ายรับส่งแรงดันไฟฟ้าบวกไปตามวงจรแฮนด์เช็คกิ้งเมื่อมันพร้อมที่จะรับข้อมูล เมื่อคอมพิวเตอร์ฝ่ายส่งได้รับแรงดันไฟฟ้าลบ มันจะรู้ว่าต้องหยุดส่งข้อมูล ในซอฟต์แวร์แฮนด์เช็คกิ้งสัญญาแฮนด์เช็คกิ้งเป็นอักขระพิเศษที่ถูกส่งไปตามวงจรข้อมูลแทนวงจรแฮนด์เช็คกิ้ง การใช้ฮาร์ดแวร์แฮนด์เช็คกิ้งอย่างน้อยต้องมีการเชื่อมต่อเพิ่มเติมหนึ่งเส้นเพื่อนำสัญญาณทำให้จำนวนของสายทั้งหมดเป็นสามเส้น คือ สายข้อมูลซิกแนลกราวนด์ และแฮนด์เช็คกิ้ง