หอสมุดกลาง สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 2

ทฤษฎีและแนวความคิดที่นำมาใช้ในการวิจัย

ในบทนี้จะกล่าวถึง แนวเหตุผล ทฤษฎีที่สำคัญ หรือสมมติฐานที่ใช้ในการวิจัย

เหตุผลและหลักการเกี่ยวกับการอิมูเลตเทอร์มินัล และการถ่ายข้อมูล [2]

ปัจจุบันการเชื่อมต่อไมโครคอมพิวเตอร์เข้ากับเมนเฟรม เป็นงานที่มีความจำเป็นและสำคัญมากขึ้นเรื่อยๆ ซึ่งในการเชื่อมต่อระบบให้ใช้งานได้สมบูรณ์แบบและประสพผลสำเร็จนั้น จะต้องคำนึงถึงด้านเทคนิค และตัวแปรอื่นๆอีกหลายอย่าง

การเชื่อมต่อไมโครคอมพิวเตอร์กับเมนเฟรมนั้นมีเหตุผลหลายประการ คือ

1. ใช้ไมโครคอมพิวเตอร์ทำหน้าที่เป็นเทอร์มินัล เพื่อใช้วิ่งโปรแกรมที่มีอยู่ในเมนเฟรมโดยที่ไม่ต้องไปเสียเงินหาซื้อเทอร์มินัลซึ่งมีราคาแพงกว่าไมโครคอมพิวเตอร์มาใช้

2. ใช้ไมโครคอมพิวเตอร์ในการพัฒนาซอฟแวร์ใช้งานของเมนเฟรม ตั้งแต่ช่วงการคีย์โปรแกรม คอมไพล์ และแก้ไขโปรแกรมให้ถูกต้องก่อนแล้วค่อยถ่ายข้อมูลจากไมโครคอมพิวเตอร์ขึ้นไปเก็บที่เมนเฟรม เพื่อคอมไพล์โปรแกรมใหม่ และใช้งานโปรแกรมนั้นต่อไป ทำให้สามารถลดภาระการงาน และค่าใช้จ่ายของเมนเฟรมลงได้มาก

3. สามารถทำการแลกเปลี่ยนข้อมูลระหว่างไมโครคอมพิวเตอร์และเมนเฟรมซึ่งทำให้เกิดประโยชน์ในการใช้งานอย่างมาก

ในระบบของไมโครคอมพิวเตอร์และเมนเฟรมเองมีความแตกต่างกันในหลายเรื่อง คือ

1. โปรโตคอลการสื่อสาร (Communication Protocol)

การจัดเรียงข้อมูลเพื่อส่งไปตามสายสื่อสาร ซึ่งเมนเฟรมและไมโครคอมพิวเตอร์ใช้ไม่เหมือนกัน คือไมโครคอมพิวเตอร์ใช้แบบอะซิงโครนัส โดยมีพอร์ตอาร์เอส-232ซี (RS-232C) ส่วนของเมนเฟรมใช้การสื่อสารแบบซิงโครนัส

2. รหัสอักษร

ในไมโครคอมพิวเตอร์ใช้รหัสอักษรแบบแอสกี ส่วนเมนเฟรมโดยเฉพาะเครื่องของไอบีเอ็ม (IBM) ใช้รหัสอักษรแบบเอบซีดิก (EBCDIC)

การอิมูเลตเทอร์มินัล [2]

ในการถ่ายเทข้อมูลระหว่างไมโครคอมพิวเตอร์และเมนเฟรม สิ่งที่จะต้องทำอย่างแรกคือ การอิมูเลตเทอร์มินัล เพื่อให้สามารถติดต่อกันได้รู้เรื่องระหว่างระบบทั้งสอง

ในการแปลงระบบหรืออิมูเลตนั้น อุปกรณ์ต่างๆที่เชื่อมต่ออยู่กับระบบทั้งสองจะต้องมีการแปลงสัญญาณ และรหัสเสียใหม่ เพื่อที่จะได้คุยกันรู้เรื่อง ฮาร์ดแวร์และซอฟแวร์ที่ได้รับการพัฒนามาบนเมนเฟรมจะเป็นรหัสเอบซีดิกทั้งหมด สัญญาณและรหัสเหล่านี้ เมื่อส่งมายังไมโครคอมพิวเตอร์ผ่านทางสายโคแอคเชียล (COAXIAL) ที่มีโปรโตคอลเป็นแบบเอสดีแอลซี (SDLC) จะต้องได้รับการแปลงให้เป็นรหัสแบบแอสกี ที่จะใช้ได้กับไมโครคอมพิวเตอร์ก่อน

ในการแปลงระบบ หรืออิมูเลตนั้น มีวิธีการใหญ่ๆอยู่ 2 วิธีคือ

1. ใช้เทอร์มินัลอิมูเลเตอร์คาร์ดใส่เข้าที่ไมโครคอมพิวเตอร์
2. ใช้โปรโตคอลคอนเวอร์เตอร์

วิธีการใช้เทอร์มินัลอิมูเลเตอร์คาร์ด จะทำให้ไมโครคอมพิวเตอร์สามารถติดต่อสื่อสารกับเมนเฟรม หรือกับส่วนควบคุมการสื่อสาร (Communication Controller) ที่เชื่อมต่อกับเมนเฟรมได้ โดยเสียบไอบีเอ็ม3278เทอร์มินัลคาร์ดเข้าไปในไมโครคอมพิวเตอร์ แล้วต่อสายส่งข้อมูลจากเมนเฟรมเข้ากับไมโครคอมพิวเตอร์ และใช้ซอฟแวร์การติดต่อกับเมนเฟรมที่ให้มาพร้อมกับคาร์ด

สำหรับโปรโตคอลคอนเวอร์เตอร์ ซึ่งเป็นเครื่องคอมพิวเตอร์ขนาดเล็ก ทำหน้าที่เปลี่ยนสายเชื่อมต่อของเมนเฟรมที่เป็นแบบโคแอคเชียล ให้ต่อเข้ากับพอร์ต อาร์เอส-232ซี ที่ไมโครคอมพิวเตอร์ได้ เปลี่ยนรหัสแบบเอบซีดิกให้เป็นแบบแอสกี และเปลี่ยนชุดคำสั่งควบคุมการแสดงผลให้ใช้งานได้กับจอแสดงผลของไมโครคอมพิวเตอร์

โปรโตคอลคอนเวอร์เตอร์ที่ใช้ต่อกับเมนเฟรม จะทำหน้าที่เป็นส่วนควบคุมการสื่อสารไอบีเอ็มโมเด็ล 3274 ทางด้านไมโครคอมพิวเตอร์ก็ต่อกับโปรโตคอลคอนเวอร์เตอร์ผ่านทางพอร์ต

อาร์เอส-232ซี และมีการใช้โปรแกรมสื่อสารเพื่อใช้ควบคุมการเคลื่อนย้ายข้อมูลไปมา[3]

การถ่ายข้อมูล[2]

ส่วนเรื่องการรับส่งข้อมูล หรือการถ่ายข้อมูลระหว่างไมโครคอมพิวเตอร์กับเมนเฟรมนั้น มีวิธีการแบ่งได้เป็น 3 วิธีคือ

1. วิธีที่ง่ายที่สุด คือ ดึงข้อมูลจากเมนเฟรมมาเก็บลงบนดิสก์ของไมโครคอมพิวเตอร์โดยเก็บข้อมูลจากจอภาพลงสู่ดิสก์ โดยอาศัยหลักการของการกดชิป-พรินท์สกรีน (Shift-PrtSc) ที่ว่าข้อความทั้งหลายบนจอภาพจะถูกพิมพ์ออกทางเครื่องพิมพ์ ใช้หลักการนี้มาเก็บข้อความบนจอภาพลงสู่ดิสก์ ถ้าการเก็บข้อมูลจากจอภาพสู่ดิสก์เก็บต่อๆกันไปได้ ก็จะได้ข้อมูลจากเมนเฟรมที่ต่อเนื่องกันไป โดยจะต้องตัดส่วนเกินที่ปรากฏบนจอภาพที่ไม่ใช้ออกไป วิธีการนี้ไม่ค่อยมีประสิทธิภาพ

2. วิธีที่สองนี้ขึ้นกับโปรแกรมจัดระบบงานของไมโครคอมพิวเตอร์และเมนเฟรม คือ ทีเอสโอ (TSO : Time Sharing Option) และซีเอ็มเอส

การดึงข้อมูลจากเมนเฟรมมาเก็บลงในดิสก์ของไมโครคอมพิวเตอร์ โปรแกรมจำลองการทำงานของ 3278 เทอร์มินัลจะหลอก ทีเอสโอ ซึ่งเป็นโปรแกรมจัดระบบงานของเมนเฟรมว่ามีผู้ใช้ต้องการแก้ไขแฟ้มข้อมูล เมนเฟรมจะส่งข้อมูลที่ต้องการนั้นมาให้ทีละบรรทัด เมื่อไมโครคอมพิวเตอร์ได้รับก็จะนำข้อมูลที่ได้เก็บลงดิสก์ไปเรื่อยๆจนกว่าจะหมดแฟ้มข้อมูล การหลอกทีเอสโอนั้นที่ไมโครคอมพิวเตอร์จะสั่งคำสั่งในโปรแกรมอิดิต (EDIT) ไปให้ ทางด้านเมนเฟรมไม่รู้เลยว่ากำลังส่งแฟ้มข้อมูล เพราะคิดว่าถูกโปรแกรมอิดิตเรียกใช้งานตามปกติ

ส่วนการส่งแฟ้มข้อมูลจากไมโครคอมพิวเตอร์ไปให้เมนเฟรม โปรแกรมจำลองการทำงานของ 3278 เทอร์มินัล ก็จะหลอก ทีเอสโอ ว่าผู้ใช้กำลังจะสร้างแฟ้มข้อมูลใหม่ เมนเฟรมก็จะเปิดแฟ้มข้อมูลใหม่รอไว้ให้ โปรแกรมที่อยู่ในไมโครคอมพิวเตอร์ ก็จะเริ่มอ่านแฟ้มข้อมูลส่งไปให้เมนเฟรม โดยใช้อินเสริทโมด (insert mode) ทีละบรรทัดจนหมดแฟ้มข้อมูล และปิดท้ายด้วยคำสั่ง SAVE กับ END ของโปรแกรมอิดิต แฟ้มข้อมูลก็จะถูกส่งจากไมโครคอมพิวเตอร์ไปให้เมนเฟรมได้ เมนเฟรมจะไม่รู้ว่าเป็นการส่งแฟ้มข้อมูลจากไมโครคอมพิวเตอร์มาเก็บ เพราะคิดว่าผู้ใช้ใช้โปรแกรมอิดิต คีย์ข้อมูลเข้ามา

ถ้าเมนเฟรมที่ใช้อยู่ ใช้โปรแกรมจัดระบบงานแบบซีเอ็มเอส การส่งแฟ้มข้อมูลไปมาก็จะใช้หลักการคล้ายกันนี้ เพียงแต่ว่าคำสั่งการแก้ไขแฟ้มข้อมูลของซีเอ็มเอส จะเป็นคำสั่งของโปรแกรมเอ็กซ์อีดิต (XEDIT) ซึ่งเป็นอิดิเตอร์แบบเต็มจอภาพ (Full screen editor)

ข้อเสียของวิธีการนี้ คือเวลาที่ใช้ในการรับส่งแฟ้มข้อมูลค่อนข้างช้า และไม่มีการตรวจสอบความผิดพลาดในการส่งเท่าใดนัก

3. การส่งหรือรับแฟ้มข้อมูลทำอย่างตรงไปตรงมา คือฝ่ายเครื่องเมนเฟรมและฝ่ายที่ใช้ไมโครคอมพิวเตอร์จะคอยรับหรือส่งแฟ้มข้อมูลตามต้องการ โดยใช้ซอฟแวร์คอยรับส่งทั้งสองด้าน โดยฝ่ายส่งต้องเตรียมซอฟแวร์และแฟ้มข้อมูลให้พร้อม เมื่อฝ่ายรับพร้อมที่จะรับแล้ว จึงส่งแฟ้มข้อมูลไปให้ ความเร็วในการส่งแฟ้มข้อมูลวิธีนี้สูงที่สุดในทั้ง 3 วิธี

ปัจจุบันในตลาดมีรูปแบบหรือหลักการเกี่ยวกับอิมูเลตเทอร์มินัลและการถ่ายข้อมูลที่ใช้กันมี 2 ลักษณะใหญ่ๆ คือ

1. ใช้ไอบีเอ็ม3278เทอร์มินัลคาร์ด และมีการถ่ายข้อมูลระหว่างไมโครคอมพิวเตอร์กับเมนเฟรมเป็นแบบวิธีการที่ 2 คือหลอก ทีเอสโอ หรือซีเอ็มเอส ว่ามีผู้ใช้ต้องการแก้ไขหรือสร้างแฟ้มข้อมูลใหม่ในการถ่ายข้อมูล

ไอบีเอ็ม3278เทอร์มินัลคาร์ดมีมากมายในท้องตลาด เช่น เออร์ม่า (IRMA), PCOX/STANDARD, PCOX/PLUS, PERSYST COAX/3270, AST-3270/COAX, IDEA COMM 3278 เป็นต้น แต่ของเออร์ม่า เป็นที่นิยมและใช้กันแพร่หลายมาก

การใช้ในลักษณะนี้ ที่ไมโครคอมพิวเตอร์จะต้องเพิ่มทั้งส่วนฮาร์ดแวร์ (ไอบีเอ็ม 3278คาร์ด) และซอฟแวร์

2. ใช้โปรโตคอลคอนเวอร์เตอร์ และมีการรับส่งแฟ้มข้อมูลทำอย่างตรงไปตรงมา (วิธีการแบบที่ 3) มีหลายยี่ห้อด้วยกัน เช่น ไฮดรา-ทู

ส่วนการใช้ในลักษณะนี้ ที่ไมโครคอมพิวเตอร์เองใช้ซอฟแวร์ ส่วนที่เมนเฟรมก็ติดตั้งโปรโตคอลคอนเวอร์เตอร์

ในการเชื่อมต่อไมโครคอมพิวเตอร์กับเมนเฟรมนั้น การใช้โปรโตคอลคอนเวอร์เตอร์จะดีกว่าการใช้ไอบีเอ็ม3278เทอร์มินัลคาร์ด ทั้งนี้เพราะมีเหตุผลประกอบหลายประการ โดยจะเปรียบ

เทียบระหว่าง เออร์ม่าคาร์ดกับโปรโตคอลคอนเวอร์เตอร์ไฮดรา-ทู

1. ค่าใช้จ่าย

การใช้เออร์ม่าคาร์ดนั้นที่ไมโครคอมพิวเตอร์แต่ละตัวจะต้องมีคาร์ดนี้เสียบอยู่ ยิ่งเมื่อมีความต้องการในการที่จะอิมูเลตเทอร์มินัลมากเท่าไร ก็จะทำให้ค่าใช้จ่ายสูงมากขึ้นตามจำนวนคาร์ด แต่ในกรณีของไฮดรา-ทู ค่าใช้จ่ายก็คงที่อยู่ที่ตัวไฮดรา-ทู เพียงอย่างเดียว โดยในส่วนของไมโครคอมพิวเตอร์แต่ละตัวนั้น ไม่ต้องใช้ฮาร์ดแวร์อะไรเพิ่มเติมมากนัก เพียงแต่ใช้ซอฟแวร์เท่านั้น

(เออร์ม่าคาร์ด 1 ชุดราคาประมาณ 30,000 บาท ในขณะที่ ไฮดรา-ทู เองราคาประมาณ 300,000 บาท ซึ่งมี 16 พอร์ตสามารถใช้ต่อไมโครคอมพิวเตอร์ หรือ แอสกีเทอร์มินัลพร้อมๆกันได้ 16 เครื่อง)[3]

2. ความเร็วในการแลกเปลี่ยนข้อมูล

การใช้ไฮดรา-ทู ซึ่งวิธีการถ่ายข้อมูลเป็นแบบตรงไปตรงมา สามารถที่จะทำงานได้เร็วกว่าที่จะใช้เออร์ม่าคาร์ด

3. ความยืดหยุ่นในการพัฒนา

การใช้เออร์ม่าคาร์ดมักจะติดอยู่ที่ฮาร์ดแวร์คือ เออร์ม่าคาร์ด ถึงแม้ว่าจะมีส่วนของซอฟแวร์ติดมาด้วย แต่ก็ไม่สามารถที่จะทำการปรับแต่งอะไรได้มากนัก

ส่วนการใช้ไฮดรา-ทู ซึ่งเป็นโปรโตคอลคอนเวอร์เตอร์นั้น มีความยืดหยุ่นมากกว่า คือสามารถที่จะปรับแต่งในส่วนของซอฟแวร์ ไม่ว่าจะเป็นในด้านไมโครคอมพิวเตอร์เอง หรือที่เมนเฟรม[3]

<u>โปรโตคอลคอนเวอร์เตอร์ไฮดรา-ทู</u>[3]

ไฮดรา-ทู เป็นโปรโตคอลคอนเวอร์เตอร์ที่เชื่อมต่อกับชาร์แนล (channel) โดยตรง ถูกสร้างขึ้นโดยบริษัท JDS MicroProcessing เพื่อใช้เชื่อมต่อระหว่างเมนเฟรม และแอสกีดีไวซ์แบบซีเรียล (serial ASCII devices) สามารถเชื่อมต่อได้กับแอสกีดีไวซ์ต่างๆ เช่น เทอร์มินัล เครื่องพิมพ์ พล๊อตเตอร์ (plotters) และด๊อกคิวเมนท์สแกนเนอร์ (document scanners)

ไฮดรา-ทู มีแบบ 8, 16 และ 32 พอร์ต โดยเชื่อมต่อกับชาร์แนลโดยตรงเข้ากับเครื่องไอบีเอ็มเมนเฟรมในตระกูล 360, 370, 43xx, 30xx หรือกับเมนเฟรมอื่นๆ ที่คล้ายคลึงกัน สามารถทำงานได้เฉพาะบนชาร์แนลแบบไบท์มัลติเพล็กเซอร์ (byte multiplexer channel) เท่านั้น และไม่สามารถต่อเพื่อใช้บนชาร์แนลแบบ บล็อก (block) หรือซีเล็กเตอร์ (selector channel) ได้

การเชื่อมต่อจากแอสกีดีไวซ์ไปยังไฮดรา-ทู ใช้ผ่านซีเรียลอินเตอร์เฟสแบบอาร์เอส-232 ซี ไฮดรา-ทูสามารถใช้ลักษณะของการรับส่งข้อมูลได้หลายๆลักษณะ เช่น

ความเร็วในการรับส่งข้อมูล - 50, 75, 110, 150, 300, 600, 1200, 2400, 4800 หรือ 9600 บิตต่อวินาที

จำนวนบิตของข้อมูล - 7, 8 บิตต่อ 1 ตัวอักษร

จำนวนของสต๊อปบิต - 1, 1.5, 2 บิต

พาริตี้ - คู่, คี่, หรือไม่มี

แต่ละพอร์ตสามารถทำงานเป็นอิสระไม่ขึ้นต่อกัน สามารถกำหนดลักษณะของแต่ละพอร์ตให้แตกต่างกันได้

ทุกอุปกรณ์ที่สามารถใช้กับไฮดรา-ทูได้ อาจจะเชื่อมต่อแบบโลคอล (local) หรือแบบรีโมด (remote) ก็ได้ การต่อแบบโลคอลก็เชื่อมต่อโดยตรงกับไฮดรา-ทู ส่วนการต่อแบบรีโมดก็เชื่อมต่อโดยผ่านโมเด็ม

ไฮดรา-ทูเป็นอุปกรณ์ซึ่งทำงานภายใต้การควบคุมของไมโครโปรเซสเซอร์ที่มีความเร็วสูง ไมโครโปรเซสเซอร์แต่ละตัวทำงานภายใต้การควบคุมของแต่ละโปรแกรมไมโครโค๊ด (microcode) โดยโปรแกรมทั้งหมดนี้ถูกออกแบบและพัฒนาโดยบริษัท JDS MicroProcessing โดยใช้ภาษาเครื่อง (machine language) แต่ละกลุ่มของ 8 พอร์ตจะทำงานภายใต้การควบคุมของไมโครโปรเซสเซอร์แบบ 16 บิต ไซล็อค แซ็ด 8000 (16-bit Zilog Z8000) แต่ละไมโครโปรเซสเซอร์จะมีหน่วยความจำ 64 กิโลไบท์เพื่อใช้เป็นเนื้อที่ในการทำงาน โดยที่ 32 กิโลไบท์สำหรับอ่านได้อย่างเดียว (ROM, read-only memory) และ 32 กิโลไบท์สำหรับอ่านและเขียนได้ด้วย (RAM, random-access memory)

นอกจากนี้ยังมีโปรเซสเซอร์อีกตัวหนึ่งควบคุมการทำงานของชาร์แนลอินเตอร์เฟส เพื่อดูแล และติดตามชาร์แนล เมื่อเมนเฟรมส่งคำสั่งมายังไฮดรา-ทู ไมโครโปรเซสเซอร์ตัวนี้จะสั่งงานไปที่โปรเซสเซอร์แชีด 8000 ที่ควบคุมพอร์ตนั้นๆอยู่

ไฮดรา-ทู มีหน้าที่หลักๆ อยู่ 2 ประการคือ

1. อิมูเลตตัวเองเป็นคอนโทรลยูนิต
2. อิมูเลตเทอร์มินัล

ในที่นี้จะกล่าวถึงหน้าที่ หรือความสามารถหลักๆของไฮดรา-ทู ส่วนหน้าที่อื่นๆนั้น สามารถหารายละเอียดได้ในคู่มือที่มาพร้อมกับไฮดรา-ทู

<u>การอิมูเลตคอนโทรลยูนิต</u>

เพราะว่าไฮดรา-ทูต่อโดยตรงเข้ากับชาร์แนลของเมนเฟรม หน้าที่ของมันจึงเป็นเสมือนคอนโทรลยูนิต และยังทำหน้าที่เพิ่มเติมเป็น โปรโตคอลคอนเวอร์เตอร์ โดยการแปลและตอบสนองต่อคำสั่งชาร์แนล เมื่อไฮดรา-ทูได้รับข้อมูลจากเมนเฟรม จะแปลเปลี่ยนข้อสนเทศนั้นในรูปแบบที่แอสกีดีไวซ์สามารถรับรู้ได้ และในทางกลับกัน ข้อมูลที่ได้รับจากแอสกีดีไวซ์จะถูกแปลเปลี่ยนไปในรูปแบบซึ่งเมนเฟรมสามารถรับรู้ได้ เสมือนว่าได้รับจากคอนโทรลยูนิต ไฮดรา-ทูสามารถอิมูเลตได้ทุกฟังก์ชั่นของคอนโทรลยูนิตไอบีเอ็มแบบ3272 ในแง่ซอฟแวร์ของเมนเฟรม จะไม่มีความแตกต่างระหว่างไฮดรา-ทู และคอนโทรลยูนิตไอบีเอ็มแบบ3272 ทุกอุปกรณ์ที่ต่อเข้ากับไฮดรา-ทูจึงดูเสมือนเป็นอุปกรณ์แบบโลคอลของเมนเฟรม

นอกจากนี้ไฮดรา-ทูสามารถอิมูเลตฟังก์ชั่นของเครื่องพิมพ์ไอบีเอ็มโมเด็ล 1403, 3211 ได้อีกด้วย

<u>การอิมูเลตเทอร์มินัล</u>

เทอร์มินัลไอบีเอ็มแบบ3277 เป็นแบบสกรีน-โอเรียนเต็ด (screen-oriented) โดยที่ข้อมูลรับส่งในรูปของบล็อกๆละ 1920 ตัวอักษร สกรีนทั้งหมดจะถูกทำงานโดยผู้ใช้ก่อนที่การส่งข้อมูลจริงๆจะเกิดขึ้น เทอร์มินัลไอบีเอ็มแบบ3277 จะมีไมโครโปรเซสเซอร์ควบคุมฟังก์ชั่นของเทอร์มินัล เช่น TAB, ERASE EOF, CLEAR ในทางกลับกันแอสกีเทอร์มินัลเป็นแบบคาร์เรคเตอร์-โอเรียนเต็ด (character-oriented) โดยแสดงตัวอักษรที่ละตัวที่ได้รับมาทางจอภาพ และส่งตัวอักษรที่

ละตัวที่คีย์เข้ามา ฟังก์ชั่นเบื้องต้นของโปรโตคอลคอนเวอร์เตอร์ไม่เพียงแต่แปลรหัสอักษรแบบเอบซีดิกไปเป็นรหัสอักษรแบบแอสกี และในทางกลับกัน แต่จะต้องสามารถทำฟังก์ชั่นของไมโครโปรเซสเซอร์ของเทอร์มินัลไอบีเอ็มแบบ3277 ซึ่งทำให้แอสกีเทอร์มินัลสามารถทำงานแบบสกรีน-โอเรียนเต็ดในทางตรรกได้ ไฮดรา-ทูถูกออกแบบมาโดยเฉพาะเพื่อสามารถทำให้แอสกีเทอร์มินัลสามารถอิมูเลตฟังก์ชั่นทั้งหมดของเทอร์มินัลไอบีเอ็มแบบ3277 ได้ ไฮดรา-ทูมีแบบของแอสกีเทอร์มินัลที่สร้างขึ้นภายในให้เลือกมากกว่า 30 แบบ และยังสามารถสร้างแบบเพิ่มเติมขึ้นได้อีก 8 แบบของเทอร์มินัลต่อ 1 โปรเซสเซอร์ที่ผู้ใช้สามารถสร้างขึ้นมาเองได้

ไฮดรา-ทูสามารถติดตามและแก้ไขบนสกรีนเสมือน (บัฟเฟอร์ภายใน) ของสกรีนที่แท้จริงของแต่ละเทอร์มินัลที่เชื่อมต่อ เมื่อเมนเฟรมส่งคำสั่งเขียน (write) สกรีนเสมือนอีกสกรีนหนึ่งจะถูกสร้างขึ้นภายในไฮดรา-ทู จากนั้นไฮดรา-ทูจะทำการเปรียบเทียบระหว่างสกรีนเสมือนที่มีอยู่ก่อนและสกรีนเสมือนอันใหม่ที่ถูกสร้างขึ้นโดยคำสั่งจากเมนเฟรม ตัวอักษรเฉพาะที่มีการเปลี่ยนแปลงเท่านั้น พร้อมกับตำแหน่งของเคอร์เซอร์ในการควบคุมตัวอักษรจะถูกส่งไปยังเทอร์มินัล เทคนิคนี้เรียกว่า "การลดความซ้ำซ้อนของข้อมูล" ซึ่งลดเวลาในการเปลี่ยนแปลงสกรีนที่เทอร์มินัล

เมื่อมีการคีย์ที่เทอร์มินัล เทอร์มินัลจะส่งคีย์นั้นไปยังไฮดรา-ทู ไฮดรา-ทูจะแปลฟังก์ชั่นนั้นแล้วส่งตัวอักษรที่เหมาะสมกลับไปยังเทอร์มินัล ถ้าตัวอักษรถูกคีย์อยู่ตรงกลางของสดมภ์ข้อมูล ไฮดรา-ทูจะส่งตัวอักษรกลับไปยังเทอร์มินัล ทำให้เกิดการเปลี่ยนแปลงภายในสกรีนเสมือนซึ่งเป็นบัฟเฟอร์ได้อย่างเหมาะสม เช่น ถ้า TAB คีย์ถูกกดเข้ามา ไฮดรา-ทูจะหาในบัฟเฟอร์ที่ใช้เป็นสกรีนเสมือน เพื่อหาจุดเริ่มต้นของสดมภ์ถัดไป จากนั้นส่งคำสั่งกำหนดเคอร์เซอร์ (set cursor) ไปยังเทอร์มินัลเพื่อย้ายเคอร์เซอร์ไปยังตำแหน่งที่เหมาะสมบนสกรีน พวก attention generating keys (เช่น ENTER, PA2, ...) จะไม่เปลี่ยนแปลงสกรีนเสมือน แต่ทำให้ไฮดรา-ทูส่งคำสั่งเซอร์วิซรีเควซ (service request) ไปที่เมนเฟรม การตอบรับของเมนเฟรมต่อคำสั่งนี้ทำให้เกิดการโต้ตอบกันระหว่างไฮดรา-ทู และเมนเฟรม ผลทำให้เกิดการเปลี่ยนแปลงสกรีนที่ถูกสร้างขึ้นและส่งไปยังไฮดรา-ทูโดยเมนเฟรม

รายชื่อฟังก์ชั่น และความสามารถของเทอร์มินัลไอบีเอ็มแบบ 3277 ที่ถูกอิมูเลตตามมาตรฐานของแอสกีเทอร์มินัล

Screen Attributes	:	Normal-Intensity
		High-Intensity
		Non-Display
Keyboard Support	:	Clear
		Enter
		New Line
		Tab
		Back Tab
		Home
		Cursor Left
		Cursor Right
		Cursor Up
		Cursor Down
		Dup Key
		Field Mark
		Insert Mode
		Delete Char
		Reset
		PA1 - PA3
		PF1 - PF24
		Test Request
		Erase EOF
		Erase Input

แอสกีเทอร์มินัลซึ่งมีอินเตอร์เฟสอาร์เอส-232ซี ส่วนมากสามารถเชื่อมต่อกับไฮดรา-ทูได้ โดยต้องมีสิ่งจำเป็นเพิ่มเติมเพียงเล็กน้อย คือ เทอร์มินัลต้องมีความสามารถในการรับรู้ ลำดับของคำสั่งที่ได้รับจากซีเรียลพอร์ตอาร์เอส-232ซี ได้แก่ คำสั่งควบคุมตำแหน่งของเคอร์เซอร์ และคำสั่งเคลียร์สกรีน กรณีเทอร์มินัลต้องการใช้วีดีโอแอ็ททริบิวท์ที่แตกต่างกัน เช่น high-intensity, low-intensity field เทอร์มินัลต้องสามารถรับคำสั่งวีดีโอแอ็ททริบิวท์เหล่านี้จากพอร์ตอาร์เอส-232ซีได้ แต่ก็ไม่จำเป็นสำหรับเทอร์มินัลในการสนับสนุนวีดีโอแอ็ททริบิวท์ที่แตกต่างกัน ถ้าเทอร์มินัลไม่สามารถแสดงแอ็ททริบิวท์ของสดมภ์ที่แตกต่างกันได้ ไฮดรา-ทูจะแสดงทุกสดมภ์ด้วยความเข้มที่เหมือนกัน

เพราะเหตุว่าคีย์บอร์ดของแอสกีเทอร์มินัลส่วนมากไม่มี พีเอฟ (PF) และ พีเอ (PA) ฟังก์ชั่น ไฮดรา-ทูจึงอิมูเลตการทำงานของเทอร์มินัลไอบีเอ็มแบบ3277 บนแอสกีเทอร์มินัลโดยการใช้คอนโทรล (Ctrl) และหรือ เอสเคป (Esc) ซีเควันซ์ ที่ถูกสร้างโดยแอสกีเทอร์มินัล โดยอาจจะคีย์รหัสคอนโทรลพิเศษเหล่านี้โดยตรงจากคีย์บอร์ด หรือรหัสพิเศษเหล่านี้ถูกเปลี่ยนอย่างอัตโนมัติโดยตารางสำหรับแปลรหัสภายในไฮดรา-ทู ฟังก์ชั่นของเทอร์มินัลที่ใช้เอสเคปซีเควันซ์แทน จะถูกแปลงโดยตารางสำหรับแปลรหัสเท่านั้น

ไฮดรา-ทูทำการเปลี่ยนแปลงบัฟเฟอร์ภายในที่ใช้เป็นสกรีนเสมือน โอเปอร์เรชั่นทั้งหมดที่มีต่อเทอร์มินัลที่เชื่อมต่อกับไฮดรา-ทูจะถูกกระทำภายในไฮดรา-ทูก่อนโดยการใช้สกรีนเสมือน สิ่งนี้ทำให้โอเปอเรชั่นส่วนใหญ่เสร็จอย่างรวดเร็ว โดยที่ไม่ได้ขึ้นอยู่กับการติดต่อสื่อสารกับเทอร์มินัล ถ้าผลของโอเปอเรชั่นทำให้เกิดการเปลี่ยนแปลงสกรีนเสมือน ไฮดรา-ทูจะส่งเฉพาะสิ่งที่เปลี่ยนแปลงไปยังเทอร์มินัล เพื่อทำให้สกรีนของเทอร์มินัลมีข้อมูลที่ทันสมัย การใช้วิธีนี้จะทำให้อุปกรณ์ที่เชื่อมต่อแบบรีโมด สามารถตอบสนองต่อโอเปอเรชั่นส่วนใหญ่ได้อย่างรวดเร็ว เสมือนว่าเชื่อมต่อแบบโลคอล

<u>การอิมูเลตโปรโตคอล 3270</u>

ในที่นี้จะกล่าวถึง กิริยาและปฏิกิริยา ของไฮดรา-ทูที่มีต่อคำสั่ง 3270 และวิธีที่คำสั่ง 3270 ถูกอิมูเลตบนเทอร์มินัล โดยจะกล่าวถึงแต่ละฟังก์ชั่นของเทอร์มินัลไอบีเอ็มแบบ3270

คำสั่งเริ่มต้นสดมภ์ (Start field Order) เป็นคำสั่ง 3270 ที่ใช้เพื่อกำหนดแอ็ททริบิวท์ของสดมภ์ที่ตามมา แต่ละสดมภ์ที่แยกจากกันบนสกรีนต้องถูกกำหนดโดยคำสั่งนี้

เมื่อเมนเฟรมส่งคำสั่งนี้มายัง ไฮดรา-ทู ไฮดรา-ทูจะจำตำแหน่งในบัฟเฟอร์ที่ใช้เป็นสกรีนเสมือน ณ.ตำแหน่งที่อ้างถึงในคำสั่ง และจะสำรองจุดนั้นสำหรับแอ็ททริบิวท์ไบท์ที่ถูกอิมูเลต ทั้งนี้เนื่องจากว่าแอสกีเทอร์มินัลไม่ใช้แอ็ททริบิวท์ไบท์

คำสั่งนี้ถูกใช้เป็นตัวชี้ (index) ไปยังตารางตัวอักษรที่ใช้เป็นวีดิโอแอ็ททริบิวท์ ที่ถูกกำหนดไว้ในการนิยามเทอร์มินัล (TDF, terminal definition) สำหรับเทอร์มินัลนี้ โดยการใช้ไบท์นี้เป็นตัวชี้ ทำให้ไฮดรา-ทูสามารถเลือกไบท์ที่ใช้ควบคุมการแสดงภาพ (video display control byte) ที่ถูกต้อง เพื่อส่งไปยังเทอร์มินัลเพื่ออิมูเลตชนิดของสดมภ์ เช่น low intensity เป็นต้น

คำสั่งกำหนดตำแหน่งบัฟเฟอร์ (Set Buffer address Order) เมื่อเมนเฟรมส่งคำสั่งนี้มายังไฮดรา-ทู ไฮดรา-ทูจะอิมูเลตคำสั่งนี้โดยการแก้ไขตัวชี้ ที่ชี้ไปยังบัฟเฟอร์ที่ใช้เป็นสกรีนเสมือน คำสั่งนี้จึงไม่มีความจำเป็นในการอ้างอิงถึงอุปกรณ์จริงๆ เช่น เทอร์มินัล เพราะว่าสกรีนเสมือนจะถูกติดตาม และแก้ไขให้ข้อมูลทันสมัยโดยไฮดรา-ทู คำสั่งนี้จะทำให้ตำแหน่งใหม่ภายในบัฟเฟอร์ถูกเลือกขึ้น และบันทึกตัวชี้ไว้ ผลลัพธ์ที่ตามมาจากคำสั่งนี้ จะถูกใส่ลงบัฟเฟอร์ ณ.ตำแหน่งใหม่นั้น

คำสั่งสอดแทรกเคอร์เซอร์ (Insert Cursor Order) เมื่อไฮดรา-ทูได้รับคำสั่งนี้จากเมนเฟรม ไฮดรา-ทูจะอิมูเลตคำสั่งนี้โดยจะส่งคำสั่งกำหนดตำแหน่งเคอร์เซอร์ (set cursor address sequence) ไปยังเทอร์มินัลเพื่อวางตำแหน่งของเคอร์เซอร์ ณ.ตำแหน่งปัจจุบันที่ถูกแสดงโดยบัฟเฟอร์ภายในที่ใช้เป็นสกรีนเสมือน ตำแหน่งใหม่ของเคอร์เซอร์จะถูกบันทึกไว้ เพื่อว่าเมื่อมีการป้อนข้อมูลเข้ามาทางเทอร์มินัล ข้อมูลนั้นๆจะได้ถูกบันทึกลงในบัฟเฟอร์เสมือน ณ.ตำแหน่งที่ถูกต้อง

คำสั่งโปรแกรมแท็บ (Program Tab Order) คำสั่งนี้ทำให้ไฮดรา-ทูค้นหาในบัฟเฟอร์ที่ใช้เป็นสกรีนเสมือน เพื่อหาตำแหน่งของสดมภ์ที่ถูกต้องถัดไปสำหรับการอินพุต ตำแหน่งของบัฟเฟอร์จะถูกคำนวณหา จากนั้นไฮดรา-ทูจะส่งคำสั่งกำหนดตำแหน่งเคอร์เซอร์ไปยังเทอร์มินัล เพื่อวางตำแหน่งของเคอร์เซอร์

คำสั่งทำซ้ำจนถึงตำแหน่ง (Repeat to address Order) คำสั่งนี้ไฮดรา-ทูจะเติม

016436

ในเนื้อที่ระหว่างตำแหน่งเก่า และตำแหน่งใหม่ในบัฟเฟอร์ที่ใช้เป็นสกรีนเสมือน ด้วยตัวอักษรที่เจาะจง หลังจากที่บัฟเฟอร์ของสกรีนเสมือนใหม่ถูกสร้างขึ้น ไฮดรา-ทูจะส่งคำสั่งไปยังเทอร์มินัลเพื่อทำการเปลี่ยนแปลงข้อมูลบนสกรีนของเทอร์มินัลให้ทันสมัย เพื่อให้เท่าเทียมกับเนื้อหาของบัฟเฟอร์ภายในไฮดรา-ทูที่ใช้เป็นสกรีนเสมือน

คำสั่งลบตำแหน่งที่ไม่มีการป้องกันจนถึงตำแหน่ง (Erase unprotected to address Order) เมื่อไฮดรา-ทูได้รับคำสั่งนี้จากเมนเฟรม ไฮดรา-ทูจะเคลียร์สกรีนทั้งหมด ณ.ตำแหน่งที่ไม่มีการป้องกัน (unprotected) ในบัฟเฟอร์เสมือน และรวมถึงตำแหน่งที่ถูกเจาะจงในคำสั่งนี้ หลังจากที่บัฟเฟอร์เสมือนถูกสร้างขึ้นใหม่ สกรีนของเทอร์มินัลที่เชื่อมต่อกับไฮดรา-ทู จะต้องถูกปรับปรุงแก้ไขให้ทันสมัย เพื่อสะท้อนให้เห็นเนื้อหาใหม่ของบัฟเฟอร์เสมือนที่ใช้เป็นสกรีนเสมือน ในบางกรณีไฮดรา-ทูจะเคลียร์สกรีนทั้งหมดก่อน และเขียนข้อมูลเก่า ซึ่งวิธีนี้จะมีประสิทธิภาพมากกว่าที่จะเคลียร์เฉพาะตำแหน่งที่เจาะจง แต่ไม่ว่าวิธีการใดก็ให้ผลลัพธ์ที่เหมือนกัน

คำสั่งลบ/เขียน (Erase/Write Command) เมื่อไฮดรา-ทูได้รับคำสั่งนี้จากเมนเฟรม ไฮดรา-ทูจะเคลียร์บัฟเฟอร์เสมือนก่อนการเปลี่ยนแปลงสกรีนเสมือนด้วยข้อมูลที่ถูกเขียนโดยคำสั่งนี้ หลังจากสกรีนเสมือนถูกสร้างขึ้นใหม่ ไฮดรา-ทูจะส่งชุดคำสั่งไปยังเทอร์มินัลเพื่อทำให้สกรีนของเทอร์มินัลที่เชื่อมต่อ เปลี่ยนแปลงให้ทันสมัยเพื่อสะท้อนให้เห็นเนื้อหาของสกรีนเสมือนล่าสุด

คำสั่งลบ/เขียน สลับกัน (Erase/Write Alternate) คำสั่งนี้ไฮดรา-ทูไม่รับรู้ ถ้าไฮดรา-ทูได้รับคำสั่งนี้จากเมนเฟรม ไฮดรา-ทูจะปฏิเสธคำสั่งนี้กลับไปที่เมนเฟรม คือไม่สามารถอิมูเลตคำสั่งนี้ได้

เพื่อหลีกเลี่ยงการปฏิเสธคำสั่งนี้ ไฮดรา-ทูมีตารางที่ใช้เปลี่ยนแปลงคำสั่ง ที่ใช้ในการเปลี่ยนออพโค๊ดของคำสั่งชาร์แนล (CCW opcode) ไปเป็นออพโค๊ดตัวอื่นที่ใช้ได้ การแปลงนี้สามารถถูกทำให้สำเร็จได้ เมื่อมีคำสั่งอื่นที่ทำงานเหมือนกันกับคำสั่งเก่า คำสั่งนี้สามารถถูกจำลองโดยใช้คำสั่งลบ/เขียน ในการทำการแปลงคำสั่งนี้ คำสั่งเก่าและคำสั่งใหม่จะต้องถูกกำหนดเข้าไปในไฮดรา-ทู ด้วยคำสั่ง LOAD SYS diagnostic

คำสั่งอ่านบัฟเฟอร์ (Read Buffer) เมื่อไฮดรา-ทูได้รับคำสั่งนี้จากเมนเฟรม ไฮดรา-ทูจะส่งกลับด้วยเนื้อหาของบัฟเฟอร์เสมือน ถ้าคำสั่งนี้ใช้โดดๆ ข้อมูลตำแหน่งแรกที่ถูกส่งกลับ

ไปที่เมนเฟรม คือ ไบท์แรกของบัฟเฟอร์เสมือน (สกรีนตำแหน่ง 0 หรือ แถวที่ 1 แนวที่ 1) แต่ถ้าคำสั่งนี้ใช้รวมกับคำสั่งอื่นๆ ข้อมูลในบัฟเฟอร์เสมือนที่ส่งออกมาจะเริ่มต้นที่ตำแหน่งสุดท้ายที่ถูกกำหนดโดยคำสั่งที่มาก่อน คำสั่งนี้จะไม่มีผลเปลี่ยนแปลงต่อสกรีนเสมือน และไม่มีการติดต่อสื่อสารเกิดขึ้นระหว่างไฮดรา-ทูกับเทอร์มินัล

คำสั่งอ่านตำแหน่งที่ถูกแก้ไข (Read Modified) เมื่อไฮดรา-ทูได้รับคำสั่งนี้จากเมนเฟรม ไฮดรา-ทูจะหาตำแหน่งของสดมภ์แรกในบัฟเฟอร์เสมือนจากตำแหน่งบัฟเฟอร์ปัจจุบันที่ถูกแก้ไข ไฮดรา-ทูจะตอบกลับด้วยตำแหน่งของเคอร์เซอร์ปัจจุบัน และเอดไบท์ (AID byte, attention Identification byte) และตามด้วยคำสั่งกำหนดตำแหน่งบัฟเฟอร์ และตำแหน่งของเคอร์เซอร์ของแอ็ททริบิวท์ของสดมภ์บวกด้วย 1 สำหรับแต่ละสดมภ์ที่ถูกแก้ไขในสกรีนเสมือน

ถ้าคำสั่งนี้ได้รับโดดๆ สดมภ์แรกบนสกรีนจะถูกตอบกลับไป แต่ถ้าคำสั่งนี้ใช้ร่วมกับคำสั่งอื่นๆ สดมภ์แรกบนสกรีนที่ถูกตอบกลับไปนั้น จะเป็นสดมภ์แรกที่พบหลังจากตำแหน่งบัฟเฟอร์ปัจจุบันที่ถูกกำหนดจากคำสั่งที่มาก่อนหน้า คำสั่งนี้จะไม่มีการติดต่อสื่อสารเกิดขึ้นระหว่างไฮดรา-ทูและเทอร์มินัล

คำสั่งเขียน (Write) เมื่อเมนเฟรมส่งคำสั่งนี้ไปที่ไฮดรา-ทู เพื่อส่งข้อมูลไปที่เทอร์มินัล ข้อมูลที่ส่งไปนั้น ไฮดรา-ทูจะบันทึกลงในบัฟเฟอร์เสมือนปัจจุบัน ณ.ตำแหน่งของบัฟเฟอร์ปัจจุบัน ขณะที่ข้อมูลใหม่ได้รับเข้ามา ตำแหน่งของบัฟเฟอร์จะถูกเปลี่ยนแปลงไป เพื่อว่าข้อมูลอื่นๆที่ตามมาจะถูกใส่เข้าไปในบัฟเฟอร์เสมือนได้ทันทีทันใดตามหลังข้อมูลนี้ โดยไม่ต้องใช้คำสั่งกำหนดตำแหน่งบัฟเฟอร์ก่อน

เมื่อคำสั่งนี้เสร็จ สกรีนเสมือนที่ถูกสร้างขึ้นใหม่จะถูกเปรียบเทียบกับอันเก่า ส่วนที่เกิดการเปลี่ยนแปลงเท่านั้นที่ถูกส่งไปยังเทอร์มินัลเพื่อแก้ไขเนื้อหาบนสกรีนของเทอร์มินัลให้ทันสมัย

<u>ตารางนิยามเทอร์มินัล</u>

ไฮดรา-ทูสามารถใช้เชื่อมต่อกับแอสกีเทอร์มินัลได้หลายๆแบบ โดยการใช้ตารางนิยามเทอร์มินัลที่สร้างขึ้นแล้วเก็บไว้ในหน่วยความจำ (ROM, Read Only Memory) และไฮดรา-ทูยังสำรองเนื้อที่ไว้สำหรับให้ผู้ใช้กำหนดหรือนิยามเทอร์มินัลเพิ่มเติมได้อีก โปรเซสเซอร์ละ 8 นิยามเก็บไว้ในหน่วยความจำ ROM

เทอร์มินัลที่สามารถใช้เชื่อมต่อกับไฮดรา-ทูได้ จะต้องมีสิ่งต่อไปนี้ คือ

1. เทอร์มินัลต้องเป็นแอสกีดีไวซ์

2. ต้องมีอินเตอร์เฟสแบบซีเรียล อาร์เอส-232ซี ทั้งนี้เพราะว่าไฮดรา-ทูมีการติดต่อสื่อสารโดยการใช้ อาร์เอส-232ซี เท่านั้น

3. เทอร์มินัลต้องมีคอนโทรลซีเควนซ์ เพื่อใช้สำหรับการอ้างอิงตำแหน่งของเคอร์เซอร์โดยตรงโดยใช้แถวและแนว คือต้องมีคำสั่งที่ใช้วางตำแหน่งเคอร์เซอร์ โดยการส่งข้อมูลของแถวและแนวใหม่ รูปแบบของแถวและแนว สำหรับคำสั่งนี้ อาจเป็นได้หลายรูปแบบ เพื่อว่าเทอร์มินัลส่วนใหญ่จะได้ไม่มีปัญหาในเรื่องนี้

4. เทอร์มินัลต้องมีคอนโทรลซีเควนซ์เพื่อเคลียร์สกรีน

สำหรับความสามารถอื่นๆที่จะทำให้เทอร์มินัลใช้ประโยชน์ได้มาก คือ ความสามารถในการแสดงภาพตัวอักษร ด้วยความเข็มที่แตกต่างกัน ซึ่งถ้าเทอร์มินัลมีความสามารถนี้ จะทำให้ไฮดรา-ทูสามารถอิมูเลตความเข็มอ่อนของสดมภ์ได้ แต่ความสามารถนี้ถ้าไม่มีก็ไม่เป็นไร

กรณีที่เทอร์มินัลมีฟังก์ชั่นคีย์บนคีย์บอร์ดมากๆ จะทำให้ง่ายในการกำหนดคีย์เพื่ออิมูเลตพีเอฟ/พีเอ คีย์ แต่อย่างไรก็ตาม พีเอฟ/พีเอ คีย์อาจจะถูกอิมูเลตโดยการใช้คีย์ เอสเคป ตามด้วยคีย์อื่นๆก็ได้ เช่น เอสเคป 1 อาจกำหนดให้ใช้แทน พีเอฟ 1 เป็นต้น

ต่อไปจะขอกล่าวถึงหลักการที่ไฮดรา-ทูใช้ ตารางนิยามเทอร์มินัล ในการอิมูเลตเทอร์มินัล

เมื่อป้อนตัวอักษร หรือกดปุ่มฟังก์ชั่นบนคีย์บอร์ดที่เทอร์มินัล เทอร์มินัลจะส่งตัวอักษร หรือรหัสที่เกิดขึ้นไปยังไฮดรา-ทู ไฮดรา-ทูจะแปลงตัวอักษรหรือรหัสนั้นด้วยตารางนิยามเทอร์มินัล เพื่อหาตัวอักษรหรือรหัสที่แท้จริงที่ต้องการใช้ แล้วส่งต่อให้เมนเฟรม ด้วยวิธีการนี้ทำให้ไฮดรา-ทูสามารถอิมูเลตคีย์บอร์ดใดๆ หรือกำหนดความหมายใหม่ให้กับคีย์บนคีย์บอร์ด และสิ่งนี้ทำให้เกิดความยืดหยุ่นในการกำหนดคีย์บอร์ดในทางที่เหมาะสมกับการใช้งานได้

เมื่อเมนเฟรมส่งคำสั่งไปยังเทอร์มินัล คำสั่งนั้นจะถูกส่งไปยังไฮดรา-ทูก่อน ไฮดรา-ทูจะทำการอิมูเลตคำสั่งนั้นโดยการใช้ข้อมูลในตารางนิยามเทอร์มินัล เช่น เมื่อเมนเฟรมส่งคำสั่งเคลียร์สกรีน ไฮดรา-ทูจะดึงคำสั่งเคลียร์สกรีนจากตารางนิยามเทอร์มินัล และส่งไปยังเทอร์มินัล สำหรับการวางตำแหน่งของเคอร์เซอร์ ไฮดรา-ทูจะส่งคำสั่งการวางตำแหน่งเคอร์เซอร์พร้อมกับแถวและแนว

จากแนวความคิดที่กล่าวนี้ ทำให้ไฮดรา-ทูมีความยืดหยุ่นมากในการใช้เชื่อมต่อกับเทอร์มินัลต่างๆ ได้โดยไม่ต้องมีการเพิ่มโสหุ้ยอื่นๆ

ในกรณีที่มีเทอร์มินัลใหม่ๆ ที่ยังไม่เคยปรากฏในตารางนิยามเทอร์มินัล ผู้ใช้ก็สามารถที่จะกำหนดนิยามใหม่โดยรวบรวมข้อมูลจากคู่มือทางเทคนิคของเทอร์มินัลนั้น และใส่ข้อมูลนั้นลงในตารางนิยามเทอร์มินัล เพื่อว่าไฮดรา-ทูจะได้รู้วิธีในการติดต่อสื่อสารกับเทอร์มินัลนั้น

ในการอิมูเลตโปรโตคอลของเทอร์มินัลไอบีเอ็มแบบ3270 นั้น ไฮดรา-ทู รับคำสั่งต่างๆ จากเมนเฟรม และทำตามคำสั่งเหล่านั้นกับบัฟเฟอร์ภายในที่ใช้เป็นสกรีนเสมือน สิ่งที่เกิดการเปลี่ยนแปลงในสกรีนเสมือนเท่านั้นจะถูกไฮดรา-ทูส่งคำสั่งไปยังเทอร์มินัล เพื่อแก้ไขเนื้อหาบนสกรีนให้ทันสมัย โดยคำสั่งที่ไฮดรา-ทูส่งไปยังเทอร์มินัลมีคำสั่งเคลียร์สกรีน คำสั่งกำหนดตำแหน่งเคอร์เซอร์ คำสั่งกำหนดแอ๊ททริบิวท์ของสดมภ์

นอกจากนี้ในไฮดรา-ทูยังมีตารางที่ไฮดรา-ทูใช้ในการเปลี่ยนรหัสของข้อมูลที่มาจากเมนเฟรม แบบเอบซีดิกให้เป็นรหัสแบบแอสกีซึ่งแอสกีดีไวซ์เข้าใจ และในทางกลับกัน เปลี่ยนรหัสของข้อมูลจากแอสกีดีไวซ์ จากรหัสแบบแอสกีไปเป็นแบบเอบซีดิกก่อนส่งให้เมนเฟรม

<u>ส่วนสนับสนุนในเรื่องอินพุตเอาท์พุตทั่วไป</u>

ไฮดรา-ทูนอกจากจะใช้เชื่อมต่อเข้ากับอุปกรณ์ต่างๆที่ได้กล่าวถึงแล้ว อุปกรณ์อื่นๆซึ่งปกติไม่ได้เชื่อมต่อเข้ากับเมนเฟรม ยังสามารถใช้ได้โดยการเชื่อมต่อผ่านไฮดรา-ทู ในรูปของอินพุตเอาท์พุตทั่วไป ในลักษณะนี้สามารถมีโลจิคอลพาธ (logical path) เพื่อส่งผ่านและรับข้อมูลจากอุปกรณ์นั้นๆได้ ทำงานโดยใช้โปรแกรมของผู้ใช้ ซึ่งสนับสนุนทางตรรกสำหรับอุปกรณ์นั้นๆโดยเฉพาะ เช่น อุปกรณ์ที่มีอินเตอร์เฟสแบบซีเรียล อาร์เอส-232ซี รวมถึงพล๊อตเตอร์ เครื่องอ่านบาร์โค๊ด และด๊อกคิวเมนท์สแกนเนอร์ เหล่านี้สามารถถูกใช้โดยผ่านอินเตอร์เฟสนี้

<u>โปรแกรมอาร์คอม (RCOM)</u>[1]

อาร์คอม เป็นชื่อชุดของซอฟแวร์ที่พัฒนาขึ้นโดยบริษัท Computer Vectors, Inc. เพื่อใช้งานบนเครื่องพีซี และเมนเฟรม ในการอิมูเลตเทอร์มินัล และการถ่ายแฟ้มข้อมูลระหว่างพีซีกับพีซี และระหว่างเมนเฟรมกับพีซี ซึ่งมีโปรแกรมอยู่ 3 ชุด คือ

1. อาร์คอม เป็นโปรแกรมที่วิ่งบนเครื่องไอบีเอ็มพีซี หรือไอบีเอ็มคอมแพตติเบิ้ล เพื่ออิมูเลตพีซีเป็นเทอร์มินัลที่เชื่อมต่อกับเมนเฟรม โดยผ่านคอนโทรลเลอร์ไฮดรา-ทู อิมูเลตฟังก์ชันคีย์บนพีซีให้ใช้ได้เสมือนคีย์พิเศษที่ปรากฏบนเทอร์มินัล นอกจากนี้อาร์คอมยังสามารถอิมูเลตพีซีเป็นเทอร์มินัลของเครื่อง DEC รุ่น VT52 สำหรับเมนเฟรมที่เทอร์มินัลรุ่นนี้ใช้ได้

ถ้าเมนเฟรมติดตั้งคอนโทรลเลอร์ไฮดรา-ทู อาร์คอมสามารถอิมูเลตพีซีเป็นเทอร์มินัลไอบีเอ็มรุ่น 3277 และถ้าพีซีมีอะแด็ปเตอร์ที่เป็นสี กราฟริก และจอภาพสี อาร์คอมก็สามารถอิมูเลตพีซีเป็นเทอร์มินัลไอบีเอ็มรุ่น 3279 ได้

หน้าที่ของอาร์คอม คือ อิมูเลตเทอร์มินัล และไม่มีความสามารถในเรื่องการถ่ายแฟ้มข้อมูล แต่สามารถเก็บเนื้อหาบนสกรีนลงในแฟ้มข้อมูลบนดิสก์ไว้สำหรับการพิมพ์ หรือการประมวลผลภายหลังได้ คือสามารถพิมพ์เนื้อหาบนสกรีน หรือข้อสนเทศที่ถูกส่งมาจากเมนเฟรมได้

2. อาร์คอม/เอฟที เป็นโปรแกรมที่วิ่งบนเครื่องพีซี มีความสามารถในการอิมูเลตเทอร์มินัล ซึ่งเป็นความสามารถของโปรแกรมอาร์คอม และใช้ถ่ายแฟ้มข้อมูลไป หรือกลับระหว่างพีซีได้ โดยซอฟแวร์นี้ไม่จำเป็นต้องใช้ไฮดรา-ทูเป็นตัวกลาง ใช้เพียงพอร์ตอาร์เอส-232ซี และโมเด็ม

อาร์คอม/เอฟที สามารถถ่ายข้อมูลได้ 2 โมด คือ โมดไบนารี และ โมดข้อความ

โมดไบนารี เหมาะที่จะใช้ถ่ายข้อมูลระหว่างพีซี เพราะว่า พีซีตัวรับจะได้รับแฟ้มข้อมูลที่เหมือนกับแฟ้มข้อมูลของพีซีตัวส่งทุกประการ

โมดข้อความ เครื่องหมายบอกจบระเบียน (CR/LF, carriage return/line feed) จะถูกแปลงเป็น LF และส่งเพื่อเป็นเครื่องหมายบอกจบระเบียน จากนั้นพีซีตัวรับ ซึ่งรับในโมดข้อความ จะแปลง LF ทุกๆตัว ให้เป็นเครื่องหมาย CR/LF การถ่ายข้อมูลในโมดนี้ จะเกิดปัญหาเมื่อแฟ้มข้อมูลมีเครื่องหมาย LF ปะปนอยู่ในระเบียน

3. อาร์คอม/เอชเอฟที ถูกออกแบบเพื่อใช้วิ่งบนเครื่องไอบีเอ็มเมนเฟรม ที่ติดตั้งคอนโทรลเลอร์ไฮดรา-ทู โปรแกรมนี้วิ่งภายใต้ระบบปฏิบัติการดอส/วีเอสอี หรือ วีเอ็ม/ซีเอ็มเอส โปรแกรมนี้เป็นโปรแกรมที่ใช้ในการถ่ายแฟ้มข้อมูลทั้งขึ้นและลงระหว่างเมนเฟรมกับพีซีที่ใช้โปรแกรมอาร์คอม/เอฟที

โปรแกรมอาร์คอม อาร์คอม/เอฟที ทำงานโดยยึดถือ พารามิเตอร์ไฟล์เป็นหลัก โดย

โปรแกรมจะทำงานตามพารามิเตอร์แต่ละตัวที่กำหนดไว้ ซึ่งผู้ใช้สามารถกำหนดได้ตามต้องการ พารามิเตอร์นี้รวมถึงลักษณะของการติดต่อสื่อสาร โปรโตคอล ซึ่งรายละเอียดปรากฏอยู่ในคู่มือ การใช้อาร์คอม

ในการถ่ายแฟ้มข้อมูลระหว่างเครื่องคอมพิวเตอร์ 2 เครื่องนั้น โปรโตคอลการสื่อสารข้อมูลเป็นสิ่งที่มีความสำคัญมาก ซึ่งเป็นการกำหนด กฎระเบียบ วิธีที่ถูกต้อง ที่คอมพิวเตอร์ 2 เครื่องใช้ส่งข้อสนเทศผ่านชาร์แนลการสื่อสาร โปรโตคอลสำคัญมากด้วยเหตุผล 2 ประการ เหตุผลแรกทั้งคนและเครื่องต้องรู้ และยอมรับกฎระเบียบ วิธี คอมพิวเตอร์ที่เป็นตัวรับ และตัวส่ง จะต้องใช้โปรโตคอลที่เหมือนกัน เหตุผลที่สอง บางโปรโตคอลเป็นแบบธรรมดา ง่ายต่อการใช้งาน ซึ่งโปรโตคอลนี้จะใช้งานได้ดี ก็ต่อเมื่อ ทุกสิ่งดำเนินการเป็นไปตามแผน แต่โปรโตคอลนี้จะไร้ประโยชน์เมื่อเกิดเหตุการณ์ที่ไม่คาดคิดขึ้น ซึ่งจะทำให้คอมพิวเตอร์เกิดความสูญเสีย หรือได้รับข้อมูลที่ผิดไป สำหรับโปรโตคอลที่ดีนั้น ต้องมีความสามารถในการเอาข้อมูลที่หายไปกลับคืนได้ หรือป้องกันข้อผิดพลาดได้ก่อนที่จะเกิดเป็นปัญหาใหญ่ภายหลัง

โปรโตคอล ที่อาร์คอม/เอฟที มีให้เลือกใช้

1. เอเอฟทีพี (AFTP : acronym for "asynchronous file transfer protocol") มีการควบคุมความถูกต้องของข้อมูล เพราะมีการตรวจสอบข้อผิดพลาด และการเอาข้อมูลที่หายไปกลับคืนมา โดยมีการใช้เทคนิค การเรียงลำดับตามบล็อก การคำนวณหา และการตรวจสอบ ซีอาร์ซี (CRC : cyclic redundancy check) ทุกๆบล็อกของข้อมูลที่ส่งไป มีการตอบ แอ็ก หรือ แน็ก เพื่อควบคุมความถูกต้อง และการส่งใหม่เมื่อมีข้อผิดพลาด

2. เอ็กซ์ออฟ โปรโตคอล เอ็กซ์ออน/เอ็กซ์ออฟ ใช้ได้เฉพาะการถ่ายแฟ้มข้อมูลแบบแอสกีไฟล์ โปรโตคอลเป็นแบบง่ายๆ ไม่มีการตรวจสอบข้อผิดพลาด

3. เอ็กซ์โมเด็ม เป็นโปรโตคอลที่เหมาะในการถ่ายข้อมูลกับคอมพิวเตอร์อื่นในแบบรีโมด โดยต้องการการตอบรับจากพีซีรีโมด หลังจากข้อมูลแต่ละบล็อกถูกส่งไป

การถ่ายข้อมูลระหว่างพีซีกับพีซี พีซีทั้งสองเครื่องอยู่ในฐานะที่เท่าเทียมกัน แต่ในการถ่ายข้อมูลระหว่างเมนเฟรมกับพีซีนั้น เมนเฟรมจะเป็นตัวควบคุมการถ่ายข้อมูล ซึ่งเมนเฟรมจะอยู่ในฐานะเป็นเครื่องหลัก ส่วนพีซีซึ่งเป็นคอมพิวเตอร์ที่มีขนาดเล็กกว่าจะอยู่ในฐานะเครื่องรอง

(subordinate หรือ slave)

การถ่ายข้อมูลส่งจากเมนเฟรมไปยังพีซี เรียกว่า ดาวน์โลด ส่วนการถ่ายข้อมูลส่งจากพีซีไปยังเมนเฟรม เรียกว่า อัพโลด

ในการถ่ายข้อมูลระหว่างเมนเฟรมกับพีซีนั้น ที่คอมพิวเตอร์แต่ละเครื่อง คือที่เมนเฟรมและพีซี จะต้องมีส่วนประกอบ หรือสิ่งที่มีความจำเป็น คือ

ที่เมนเฟรม

1. เมนเฟรมที่ต้องการถ่ายข้อมูลขึ้น หรือลง (อัพโลด หรือ ดาวน์โลด) จะต้องมีโปรแกรมอาร์คอม/เอชเอฟที ใช้ในการควบคุมการถ่ายข้อมูล

2. เมนเฟรมนั้นต้องเป็นไอบีเอ็มเมนเฟรม ที่มีระบบปฏิบัติการแบบดอส/วีเอสอี, เอ็มวีเอส, หรือ วีเอ็ม/ซีเอ็มเอส

3. เมนเฟรมต้องติดตั้งไฮดรา-ทู

ที่พีซี

1. พีซีจะต้องมีโปรแกรมอาร์คอม/เอฟที

2. โมเด็ม กรณีที่พีซีนั้นต้องการเป็นเทอร์มินัลแบบไดอัล-อัพ (dial-up) ในการเชื่อมต่อกับไฮดรา-ทู

จากการที่ซอฟแวร์ อาร์คอม/เอชเอฟที ซึ่งเป็นซอฟแวร์บนเมนเฟรมที่จะใช้ควบคุมการถ่ายข้อมูลระหว่างเมนเฟรมกับพีซี ไม่ได้ซื้อมา เพราะมีราคาแพง การที่ขาดซอฟแวร์ตัวนี้ทำให้ไม่สามารถถ่ายข้อมูลระหว่างเมนเฟรมกับพีซีได้

จากความสามารถของโปรแกรมอาร์คอม/เอฟที ที่ใช้ถ่ายข้อมูลไปหรือกลับระหว่างพีซีได้ ประกอบกับหลักการในการถ่ายข้อมูลที่ว่า คอมพิวเตอร์ 2 เครื่องนั้นจะต้องใช้โปรโตคอลการสื่อสารข้อมูลที่เหมือนกัน ฉะนั้นถ้าทำการศึกษาการทำงานของซอฟแวร์อาร์คอม/เอฟที แล้วนำหลักการที่ได้มาเขียนโปรแกรมซึ่งจะทำหน้าที่เหมือนกับโปรแกรมอาร์คอม/เอชเอฟที

โปรแกรมอาร์คอม/เอฟที ที่มีอยู่มีแต่ออบเจ็คโค๊ด (object) ไม่มีซอสโปรแกรม (source program) ในการศึกษาการทำงานของโปรแกรมอาร์คอม/เอฟทีนี้ ใช้โปรแกรมยูทิลิตี้ดีบัก ติดตามการทำงานของโปรแกรมนี้ โดยในการถ่ายแฟ้มข้อมูล จะเลือกศึกษาเฉพาะกรณีของ

โปรโตคอลแบบเอเอฟทีพี

หลักการและขั้นตอนการทำงานของโปรแกรมอาร์คอม/เอฟที

จากการศึกษาการทำงานของโปรแกรมอาร์คอม/เอฟที พอสรุปหลักการ และขั้นตอนการทำงานได้ดังนี้ (ดังแสดงในรูปที่ 2.1)

1. ทำงานเริ่มต้น เป็นการกำหนดค่าเริ่มต้น กำหนดอุปกรณ์ ทรัพยากรต่างๆ ตามพารามิเตอร์ไพล์

2. อ่าน หรือรอรับ อินพุตจากคีย์บอร์ด ถ้าไม่มีอินพุตจากคีย์บอร์ด ก็ข้ามไปทำข้อ 4

3. ตรวจสอบอินพุตที่ได้รับเข้ามา

3.1 ถ้าเป็นคีย์ที่กำหนดไว้เป็นอาร์คอมโปรแกรมคีย์ ก็ทำการประมวลในแต่ละรูทีนตามแต่ละฟังก์ชั่นที่กำหนดไว้

3.2 ถ้าเป็นฟังก์ชันคีย์ ก็ทำการเปลี่ยนรหัสสำหรับฟังก์ชั่นคีย์นั้น เป็นในรูปแบบของเอสเคปซีเควนซ์ ที่สัมพันธ์กับแต่ละฟังก์ชั่นคีย์ แล้วส่งรหัสที่ได้ไปที่พอร์ตการสื่อสาร

3.3 กรณีอื่นๆ ก็ส่งตัวอักษร หรือรหัสนั้นๆ ไปที่พอร์ตการสื่อสาร

4. อ่าน หรือรอรับ อินพุตจากพอร์ตการสื่อสาร แล้วเก็บลงในบัฟเฟอร์

5. อ่านตัวอักษร หรือรหัส จากบัฟเฟอร์ ถ้าไม่มีข้อมูลอยู่ หรือบัฟเฟอร์ว่าง ก็ข้ามไปทำข้อ 2

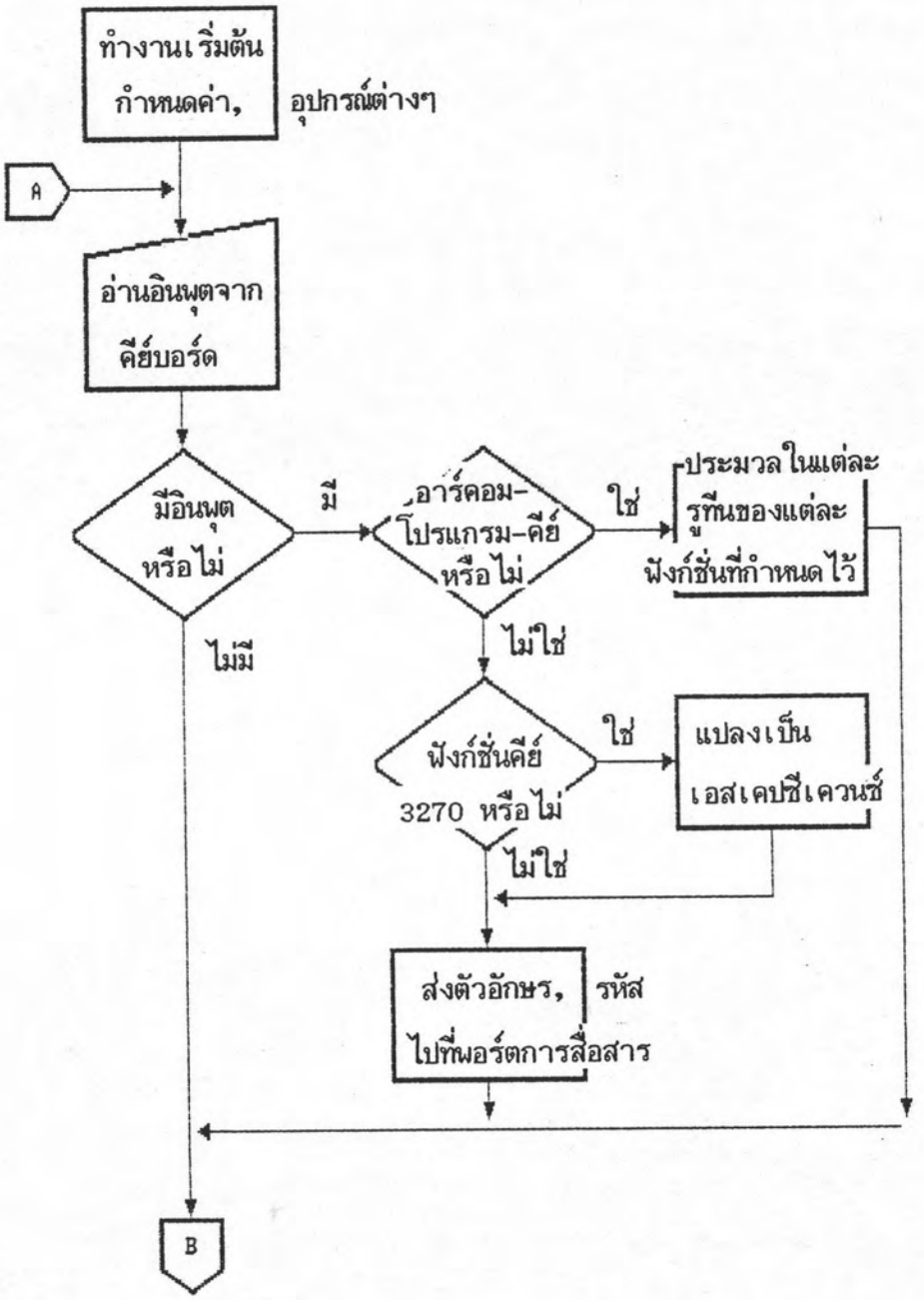
6. ตรวจสอบตัวอักษร หรือรหัสที่ได้จากบัฟเฟอร์

6.1 ถ้าเป็นคำสั่ง เอ็นคิว (ENQ) จากคอมพิวเตอร์อีกตัวหนึ่ง ก็ทำการประมวลในรูทีน การถ่ายแฟ้มข้อมูล

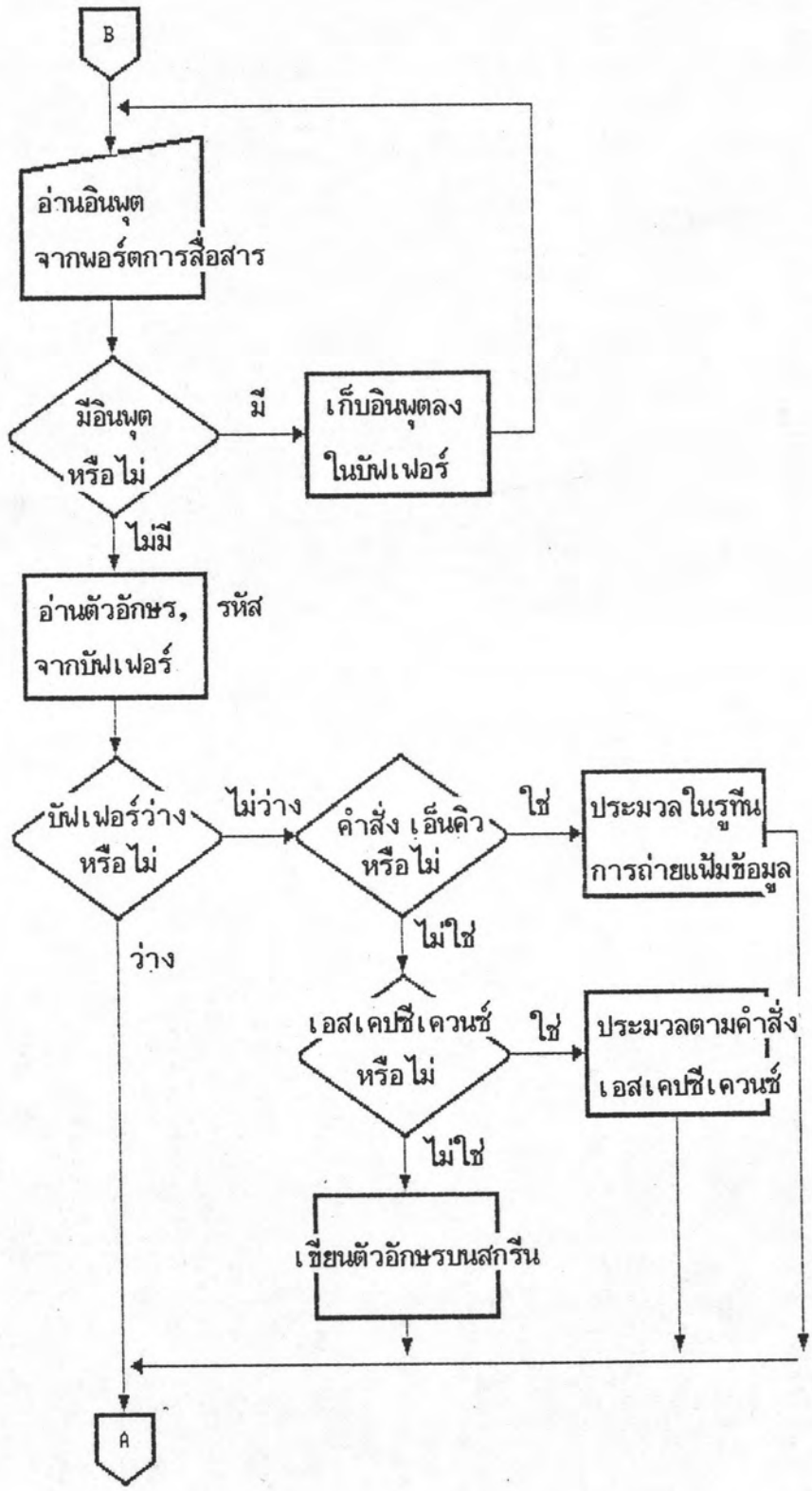
6.2 ถ้าเป็นคำสั่ง เอสเคปซีเควนซ์ จากไฮดรา-ทู ก็ทำการประมวลตามแต่ละคำสั่ง คำสั่งเหล่านี้ได้แก่ คำสั่งเคลียร์สกรีน คำสั่งกำหนดตำแหน่งเคอร์เซอร์ คำสั่งกำหนดวีดิโอแอ็ททริบิวท์

6.3 กรณีอื่นๆ ก็เขียนตัวอักษร หรือรหัสนั้นๆ บนสกรีนเทอร์มินัล

7. กลับไปทำข้อ 2



รูปที่ 2.1 แสดงขั้นตอนการทำงานของโปรแกรมอาร์คอม/เอฟที



รูปที่ 2.1 (ต่อ) แสดงขั้นตอนการทำงานของโปรแกรมอาร์คอม/เอฟที

คีย์ หรือ ฟังก์ชั่นคีย์ที่กำหนดไว้เป็น อาร์คอมโปรแกรมคีย์ มีสรุปไว้ในตารางที่ 2.1 ส่วนรายละเอียด และวิธีการใช้แต่ละฟังก์ชันมีอยู่ในคู่มือการใช้อาร์คอม

คีย์	ฟังก์ชั่น
CTRL/F1	ส่งแฟ้มข้อมูลไปที่พีซีที่ปลายทาง
CTRL/F2	รับแฟ้มข้อมูลจากพีซีที่ปลายทาง
CTRL/F3	เปลี่ยนชื่อแฟ้มข้อมูลที่ใช้ส่ง
CTRL/F4	เปลี่ยนชื่อแฟ้มข้อมูลที่ใช้รับ
CTRL/F5	เปลี่ยนชื่อแฟ้มข้อมูลที่ใช้เก็บผลการถ่ายแฟ้มข้อมูล
CTRL/F6	ยกเลิก หรือหยุดการถ่ายแฟ้มข้อมูล
CTRL/F7	ลบบรรทัดที่ 25 บนจอภาพ (ซึ่งแสดงสถานะ)
CTRL/F8	เรียก หรือทำฟังก์ชั่นออโต้ไดล
CTRL/F9	เรียกเมนูการเลือกฟังก์ชั่น
CTRL/F10	เรียกดูออนไลน์เฮวพ์

ตารางที่ 2.1 แสดงอาร์คอมโปรแกรมคีย์

หอสมุดกลาง สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

คีย์	ฟังก์ชั่น
SHIFT/F1	เปิด-ปิด เครื่องพิมพ์
SHIFT/F2	กำหนดแอ็ททริบิวท์ของจอภาพสี
SHIFT/F3	กำหนดให้พีซีที่ปลายทางส่งชื่อแฟ้มข้อมูล
SHIFT/F4	กำหนดให้พีซีที่ปลายทางรับชื่อแฟ้มข้อมูล
SHIFT/F5	บันทึกข้อมูลที่แสดงบนจอภาพลงในดิสก์
SHIFT/F6	ออกจากโปรแกรมอาร์คอมโดยไม่ขัดจังหวะไลน์ที่เชื่อมอยู่
SHIFT/F7	เรียกใช้คำสั่งของดอส
SHIFT/F8	แสดงบรรทัดที่แสดงสถานะ
SHIFT/F9	แสดงไดเรคทอรี
SHIFT/F10	เรียกใช้อาร์คอมไดเอคนอสติกโมด
SHIFT/PRTSC	พิมพ์เนื้อหาบนจอภาพไปที่เครื่องพิมพ์
CTRL/BREAK	ออกจากโปรแกรมอาร์คอมโดยขัดจังหวะไลน์ที่เชื่อมอยู่

ตารางที่ 2.1 (ต่อ) แสดงอาร์คอมโปรแกรมคีย์

ไอบีเอ็มพีซี คีย์ที่ถูกใช้	รหัสภายในพีซีที่ส่งให้ไฮดรา-ทู	รหัสที่ไฮดราแปลงและส่งต่อให้เมนเฟรม	ฟังก์ชั่นของ 3270 ที่ต้องการใช้
F1	1B 61	B1	PF1
F2	1B 62	B2	PF2
F3	1B 63	B3	PF3
F4	1B 64	B4	PF4
F5	1B 65	B5	PF5
F6	1B 66	B6	PF6
F7	1B 67	B7	PF7
F8	1B 68	B8	PF8
F9	1B 69	B9	PF9
F10	1B 6A	BA	PF10
ALT/F1	1B 31	BB	PF11
ALT/F2	1B 32	BC	PF12
ALT/F3	1B 33	81	PF13
ALT/F4	1B 34	82	PF14
ALT/F5	1B 35	83	PF15
ALT/F6	1B 36	84	PF16

ตารางที่ 2.2 แสดงฟังก์ชั่นคีย์ ซึ่งอิมูเลตฟังก์ชั่นของไอบีเอ็ม 3270 เอสเคปซีเควนซ์และรหัสที่ส่งให้เมนเฟรม

ไอบีเอ็มพีซี คีย์ที่ถูกใช้	รหัสภายในพีซีที่ส่งให้ไฮดรา-ทู	รหัสที่ไฮดราแปลงและส่งต่อให้เมนเฟรม	ฟังก์ชั่นของ 3270 ที่ต้องการใช้
ALT/F7	1B 37	85	PF17
ALT/F8	1B 38	86	PF18
ALT/F9	1B 39	87	PF19
ALT/F10	1B 3A	88	PF20
ALT/1	1B 1B	AC	PA1
ALT/2	1B 56	AE	PA2
ALT/3	1B 57	AB	PA3
ALT/4	1B 45	B0	TEST REQUEST
ALT/C หรือ			
CTRL/C	1B 4A	03	CLEAR
ENTER	0D	0D	ENTER
KEYPAD '+'	1B 6E	0E	RETURN
DEL	1B 7F	04	DELETE CHARACTER
INS	1B 50	07	SET INSERT MODE
ALT/R หรือ			
CTRL/R	1B 30	12	RESET INSERT MODE

ตารางที่ 2.2 (ต่อ) แสดงฟังก์ชั่นคีย์ ซึ่งอิมูเลตฟังก์ชั่นของไอบีเอ็ม 3270 เอสเคป ซีเควนซ์ และรหัสที่ส่งให้เมนเฟรม

ไอบีเอ็มพีซี คีย์ที่ถูกใช้	รหัสภายในพีซีที่ส่งให้ไฮดรา-ทู	รหัสที่ไฮดราแปลงและส่งต่อให้เมนเฟรม	ฟังก์ชั่นของ 3270 ที่ต้องการใช้
END	1B 49	06	ERASE EOF
CTRL/E	1B 4B	05	ERASE INPUT
HOME	1B 5A	16	HOME
CTRL/HOME	1B 58	0F	HOME-2 LOCATION
LEFT ARROW	1B 44	08	LEFT ARROW
RIGHT ARROW	1B 43	0C	RIGHT ARROW
UP ARROW	1B 41	0B	UP ARROW
DOWN ARROW	1B 42	0A	DOWN ARROW
TAB	09	09	TAB
BACK TAB	1B 47	02	BACK TAB
INS	1B 71	01	INSERT CHARACTER
CTRL/2	00	-	DISCONNECT FROM HYDRA
CTRL/Y	19	-	DIAGNOSTIC FUNCTIONS
CTRL/T	14	-	FORCE TIMEOUT
CTRL/U	15	-	SELECT USER MESSAGE

ตารางที่ 2.2 (ต่อ) แสดงฟังก์ชั่นคีย์ ซึ่งอิมูเลตฟังก์ชั่นของไอบีเอ็ม 3270 เอสเคป ซีเควนซ์ และรหัสที่ส่งให้เมนเฟรม

ส่วนฟังก์ชั่นคีย์ ซึ่งอิมูเลตฟังก์ชั่นของไอบีเอ็ม 3270 มีรายละเอียดของแต่ละฟังก์ชัน รวมถึงเอสเคปซีเควนซ์ และรหัสที่ส่งให้เมนเฟรม สรุปไว้ในตารางที่ 2.2

ในการถ่ายข้อมูลระหว่างคอมพิวเตอร์ 2 เครื่องนั้น จะต้องมีเครื่องหนึ่งเป็นตัวออกคำสั่งเริ่มการทำงานไปยังอีกเครื่องหนึ่ง ไม่ว่าจะเป็นการรับ หรือส่งข้อมูล ในที่นี้จะขอเรียก คอมพิวเตอร์เครื่องที่ออกคำสั่งเริ่มการทำงานว่า "ตัวออกคำสั่ง" และอีกเครื่องหนึ่งว่า "ตัวรับคำสั่ง" ทั้งนี้เพื่อให้เรียกได้ง่ายในการอธิบาย

ในกรณีของการถ่ายข้อมูลระหว่างพีซีกับพีซี พีซีแต่ละตัวมีฐานะเท่าเทียมกัน ตัวใดจะเป็นตัวออกคำสั่ง ขึ้นอยู่กับแต่ละเหตุการณ์ไป ส่วนในกรณีของการถ่ายข้อมูลระหว่างเมนเฟรมกับพีซี เมนเฟรมเป็นเครื่องที่มีขนาดใหญ่กว่า จะเป็นตัวควบคุมในการถ่ายข้อมูล คือจะทำหน้าที่เป็นตัวออกคำสั่ง และพีซีจะทำหน้าที่เป็น ตัวรับคำสั่ง

ในโปรแกรมอาร์คอม/เอฟที มีโปรแกรมย่อยที่ทำหน้าที่เกี่ยวกับการถ่ายแฟ้มข้อมูล อยู่ 2 โปรแกรมย่อย คือ

1. โปรแกรมย่อยที่ทำหน้าที่ถ่ายข้อมูลทั้งรับและส่ง ในฐานะที่เป็น ตัวออกคำสั่ง การทำงานในโปรแกรมย่อยนี้ ถูกเรียกใช้จาก คีย์ หรือฟังก์ชั่นคีย์ ที่ใช้เป็นอาร์คอมโปรแกรมคีย์

2. โปรแกรมย่อยที่ทำหน้าที่ถ่ายข้อมูลทั้งรับและส่ง ในฐานะที่เป็น ตัวรับคำสั่ง การทำงานในโปรแกรมย่อยนี้ ถูกเรียกใช้จากคำสั่ง เอ็นคิว ที่ได้รับจากพอร์ตการสื่อสาร ซึ่งพีซีตัวออกคำสั่ง ส่งมาให้

<u>หลักการ และขั้นตอนการทำงาน ของโปรแกรมย่อย ที่ทำหน้าที่ถ่ายข้อมูลทั้งรับและส่งในฐานะตัวออกคำสั่ง</u>

กรณีเป็นตัวส่งแฟ้มข้อมูล

1. ส่งคำสั่ง เอ็นคิว-อาร์ ไปที่พอร์ตการสื่อสาร (เพื่อส่งไปบอกพีซีที่ปลายทางให้อยู่ในโมดรับข้อมูล

2. อ่าน หรือรอรับอินพุต การตอบรับ จากพอร์ตการสื่อสาร (พีซีที่ปลายทางจะส่งการตอบรับกลับมาเพื่อบอกว่าพร้อมแล้ว ให้ส่งข้อมูลไปได้)

3. เปิดแฟ้มข้อมูล (แฟ้มข้อมูลที่ต้องการส่ง)

4. อ่านระเบียนจากแฟ้มข้อมูล แล้วเก็บไว้ในบัฟเฟอร์ส่ง แล้วทำขั้นตอนย่อยในข้อ 4 จนกระทั่ง หมดแฟ้มข้อมูล

4.1 นำเอาข้อมูลจากบัฟเฟอร์ส่ง มาตัดทอนเป็นบล็อกๆละไม่เกิน 250 ตัวอักษร

4.2 ใส่ลำดับที่ของบล็อก ความยาวของข้อมูลในบล็อก รหัสปิดหัว-ท้าย บล็อก

4.3 คำนวณหา ซีอาร์ซี และใส่ลงในบล็อก

4.4 ส่งข้อมูลที่ละบล็อก ไปที่พอร์ตการสื่อสาร

4.5 อ่าน หรือรอรับอินพุต การตอบรับ จากพอร์ตการสื่อสาร

4.6 ตรวจสอบการตอบรับ และลำดับที่ของบล็อกที่ตอบรับ

4.7 กรณีได้รับการปฏิเสธการรับ ก็ทำการส่งข้อมูลบล็อกนั้นกลับไปใหม่

5. เมื่อหมดแฟ้มข้อมูล ก็ส่งคำสั่ง อีโอที ไปที่พอร์ตการสื่อสาร (เพื่อส่งไปบอกพีซีที่ปลายทางว่า หมดข้อมูลที่จะส่งแล้ว)

กรณีเป็นตัวรับแฟ้มข้อมูล

1. ส่งคำสั่ง เอ็นคิว-เอส ไปที่พอร์ตการสื่อสาร (เพื่อส่งไปบอกพีซีที่ปลายทางให้อยู่ในโมดส่งข้อมูล)

2. เปิดแฟ้มข้อมูล (แฟ้มข้อมูลที่ใช้เก็บบันทึกข้อมูลที่จะส่งมา)

3. อ่าน หรือรอรับ อินพุต บล็อกของข้อมูล จากพอร์ตการสื่อสาร แล้วเก็บไว้ในบัฟเฟอร์รับ แล้วทำขั้นตอนย่อยในข้อ 3 จนกระทั่งได้รับคำสั่ง อีโอที (ซึ่งส่งมาจากพีซีที่ปลายทางเพื่อบอกให้รู้ว่าหมดข้อมูลที่จะรับแล้ว)

3.1 ทำการตรวจสอบลำดับที่ของบล็อก

3.2 ทำการตรวจสอบ ซีอาร์ซี (เพื่อตรวจสอบว่าข้อมูลได้รับคลาดเคลื่อนหรือไม่)

3.3 ส่งการตอบรับ พร้อมลำดับที่ของบล็อก ไปที่พอร์ตการสื่อสาร

3.4 กรณีได้รับข้อมูลคลาดเคลื่อน ก็จะส่ง การปฏิเสธการรับ ไปที่พอร์ตการสื่อสาร (เพื่อส่งไปบอกให้พีซีที่ปลายทาง ส่งข้อมูลบล็อกนั้นกลับมาให้ใหม่)

3.5 นำข้อมูลที่ได้รับ เขียนลงในแฟ้มข้อมูล

4. ปิดแฟ้มข้อมูล

<u>หลักการ และขั้นตอนการทำงาน ของโปรแกรมย่อย ที่ทำหน้าที่ถ่ายข้อมูลทั้งรับและส่ง ในฐานะตัวรับคำสั่ง</u>

กรณีเป็นตัวรับแฟ้มข้อมูล (เนื่องจากได้รับคำสั่ง เอ็นคิว-อาร์ จากพีซีที่ปลายทาง)

1. ส่งการตอบรับ ไปที่พอร์ตการสื่อสาร (เพื่อส่งไปบอกพีซีที่ปลายทางว่าพร้อมแล้วที่จะรับข้อมูล)

2. เปิดแฟ้มข้อมูล (แฟ้มข้อมูลที่ใช้เก็บบันทึกข้อมูลที่จะส่งมา)

3. อ่าน หรือรอรับ อินพุต บล็อกของข้อมูล จากพอร์ตการสื่อสาร แล้วเก็บไว้ในบัฟเฟอร์รับ แล้วทำขั้นตอนย่อยในข้อ 3 จนกระทั่งได้รับคำสั่ง อีโอที (ซึ่งส่งมาจากพีซีที่ปลายทางเพื่อบอกให้รู้ว่าหมดข้อมูลที่จะรับแล้ว)

3.1 ทำการตรวจสอบลำดับที่ของบล็อก

3.2 ทำการตรวจสอบ ซีอาร์ซี (เพื่อตรวจสอบว่าข้อมูลได้รับคลาดเคลื่อนหรือไม่)

3.3 ส่งการตอบรับ พร้อมลำดับที่ของบล็อก ไปที่พอร์ตการสื่อสาร

3.4 กรณีได้รับข้อมูลคลาดเคลื่อน ก็จะส่ง การปฏิเสธการรับ ไปที่พอร์ตการสื่อสาร (เพื่อส่งไปบอกให้พีซีที่ปลายทาง ส่งข้อมูลบล็อกนั้นกลับมาให้ใหม่)

3.5 นำข้อมูลที่ได้รับ เขียนลงในแฟ้มข้อมูล

4. ปิดแฟ้มข้อมูล

กรณีเป็นตัวส่งแฟ้มข้อมูล (เนื่องจากได้รับคำสั่ง เอ็นคิว-เอส จากพีซีที่ปลายทาง)

1. เปิดแฟ้มข้อมูล (แฟ้มข้อมูลที่ต้องการส่ง)

2. อ่านระเบียนจากแฟ้มข้อมูล แล้วเก็บไว้ในบัฟเฟอร์ส่ง แล้วทำขั้นตอนย่อยในข้อ 2 จนกระทั่ง หมดแฟ้มข้อมูล

2.1 นำเอาข้อมูลจากบัฟเฟอร์ส่ง มาตัดทอนเป็นบล็อกๆละไม่เกิน 250 ตัวอักษร

2.2 ใส่ลำดับที่ของบล็อก ความยาวของข้อมูลในบล็อก รหัสปิดหัว-ท้าย บล็อก

2.3 คำนวณหา ซีอาร์ซี และใส่ลงในบล็อก

2.4 ส่งข้อมูลที่ละบล็อก ไปที่พอร์ตการสื่อสาร

2.5 อ่าน หรือรอรับอินพุต การตอบรับ จากพอร์ตการสื่อสาร

2.6 ตรวจสอบการตอบรับ และลำดับที่ของบล็อกที่ตอบรับ

2.7 กรณีได้รับการปฏิเสธการรับ ก็ทำการส่งข้อมูลบล็อกนั้นกลับไปใหม่

3. เมื่อหมดแฟ้มข้อมูล ก็ส่งคำสั่ง อีโอที ไปที่พอร์ตการสื่อสาร (เพื่อส่งไปบอกพีซีที่ปลายทางว่า หมดข้อมูลที่จะส่งแล้ว)

<u>คำสั่งเอสเคปซีเควนซ์ของไฮดรา-ทู</u>

คำสั่ง เอสเคปซีเควนซ์ จากไฮดรา-ทู ซึ่งเป็นคำสั่งที่ไฮดรา-ทู อิมูเลตขึ้นมาเพื่อใช้แทนคำสั่งสำหรับคอนโทรลเลอร์ไอบีเอ็ม แบบ 3270 มีดังนี้

1. คำสั่งเคลียร์สกรีน เมื่อโปรแกรมอาร์คอม/เอฟที ได้รับคำสั่งนี้ จะทำการเคลียร์สกรีนทั้งหมดที่พีซี รูปแบบของคำสั่งนี้

X'1B4A' หรือ ESC J

2. คำสั่งกำหนดตำแหน่งเคอร์เซอร์ เมื่อได้รับคำสั่งนี้ จะทำการเคลื่อนย้าย หรือกำหนดตำแหน่งเคอร์เซอร์ใหม่ ไป ณ. แถว และแนว ที่ระบุในตัวคำสั่ง รูปแบบของคำสั่งนี้

X'1B59' row col หรือ ESC Y row col

โดยที่ row, col คือ ค่าแถว และแนว เป็นค่าไบนารี โดยมีค่าเริ่มต้นที่ X'20'

เช่น X'1B593820' จะหมายถึง ให้กำหนดตำแหน่งเคอร์เซอร์ใหม่ ณ.แถวที่ 25 แนวที่ 1

3. คำสั่งกำหนดวีดิโอแอ็ททริบิวท์ เมื่อได้รับคำสั่งนี้ จะเป็นการกำหนดแอ็ททริบิวท์

ความเข็มอ่อนในการแสดงภาพ ของตัวอักษร หรือข้อความที่ตามมา รูปแบบของคำสั่งนี้

X'1B53' video-character หรือ ESC S video-character

โดยที่ video-character คือ ตัวอักษรกำหนดแอ็ททริบิวท์ มีค่าเป็น

X'02' - ความอ่อน

X'04' - ความเข็ม

ซอฟแวร์ทางด้านอินพุตและเอาท์พุต

ไฮดรา-ทู มีซอฟแวร์บนเครื่องเมนเฟรมที่ใช้เป็นอินเตอร์เฟสทางด้านอินพุต และเอาท์พุต ซึ่งสามารถใช้ติดต่อสื่อสารได้กับอุปกรณ์ที่มีอินเตอร์เฟสแบบซีเรียล อาร์เอส-232ซี ซอฟแวร์ตัวนี้เป็นโปรแกรมย่อย (subroutine) ซึ่งผู้ใช้สามารถเขียนโปรแกรมเพื่อติดต่อสื่อสารกับมินิคอมพิวเตอร์และพีซีได้ โดยให้โปรแกรมย่อยนี้คอยดูแลในด้านอินพุต และเอาท์พุตที่เกิดขึ้นจริง โปรแกรมย่อยนี้มีชื่อเรียกว่า เอชวาย 100 (HY100 Subroutine)

ในการใช้งานนั้น ผู้ใช้ส่งข้อสนเทศ หรือ การร้องขอเพื่อให้ไฮดรา-ทูปฏิบัติการให้ โดยส่งข้อสนเทศเหล่านั้นลงในพารามิเตอร์ซึ่งส่งผ่านให้โปรแกรมย่อยเอชวาย 100 ในโปรแกรมย่อยนี้จะมีคำสั่งซึ่งเป็นคำสั่งชาร์แนล สั่งให้ไฮดรา-ทูทำงาน จากนั้นไฮดรา-ทูจะปฏิบัติการด้านอินพุต หรือเอาท์พุต และส่งผลกลับคืนมาที่โปรแกรมของผู้ใช้ในรูปของรหัสเพื่อบ่งบอกสถานะของเหตุการณ์ หรือผลที่เกิดขึ้น การเรียกใช้ และพารามิเตอร์ที่เกี่ยวข้องกับโปรแกรมย่อยนี้ จะได้กล่าวถึงต่อไป

การเรียกใช้ โปรแกรมย่อยเอชวาย 100

การติดต่อ หรือการใช้งานโปรแกรมย่อยนี้ กระทำโดยใช้ชุดของพารามิเตอร์ ซึ่งมีการส่งเข้า และออกจากโปรแกรมย่อยนี้ ชุดของพารามิเตอร์นี้มีด้วยกันทั้งหมด 34 พารามิเตอร์ ในการเรียกใช้เอชวาย 100 ให้ทำงานหนึ่งอย่างนั้น จะใช้พารามิเตอร์บางตัวเท่านั้น ไม่ได้ใช้ทั้งหมด พารามิเตอร์บางตัวใช้ในกรณีการทำงานด้านอินพุต บางตัวใช้ในกรณีของเอาท์พุต

การเรียกใช้ เอชวาย 100

1. สำหรับภาษาเอสเซมบลีย์

10307813

```
CALL    HY100,(PARMLIST)
```

2. สำหรับภาษาโคบอล

```
CALL  'HY100'  USING  PARMLIST
```

โดยที่ HY100 คือ ชื่อของโปรแกรมย่อย

PARMLIST คือ ตำแหน่งที่อยู่ (address) ของพารามิเตอร์แรกในชุดของพารามิเตอร์

ชุดของพารามิเตอร์ ประกอบด้วย 34 พารามิเตอร์ ซึ่งแต่ละตัวนั้นเป็นสดมภ์แบบฟูลเวิร์ด (fullword) และมีข้อสนเทศที่จำเป็นสำหรับการติดต่อระหว่างโปรแกรมของผู้ใช้ และเอชวาย 100 ทั้งไปและกลับ แต่ละพารามิเตอร์มีการกำหนดเป็นไบนารีฟูลเวิร์ด มีรายละเอียดดังนี้

<u>พารามิเตอร์ทั่วไป</u>

พารามิเตอร์ตัวที่ 1 ถึง ตัวที่ 12 เป็นพารามิเตอร์ที่ใช้เก็บข้อสนเทศที่ต้องใช้ในการทำงานด้านอินพุต หรือเอาท์พุต

พารามิเตอร์ตัวที่ 1 กำหนดหน่วยของอุปกรณ์ทางตรรก ที่มีการเชื่อมต่อกับพอร์ตของไฮดรา-ทู ซึ่งมีการกำหนดไว้เป็นพอร์ตอินพุตเอาท์พุตทั่วไป (general I/O port)

พารามิเตอร์ตัวที่ 2 กำหนดประเภทของการทำงานที่ต้องการร้องขอ เช่น การเรียกหา (polling) เพื่อเลือกอัตราความเร็วในการรับ-ส่ง การตรวจสอบผลการปฏิบัติการก่อนหน้า การรอคอยให้การทำงานก่อนหน้าเสร็จ การเคลียร์บัฟเฟอร์ การเริ่มต้นทำงานด้านเอาท์พุต การเริ่มต้นทำงานด้านอินพุต การเริ่มต้นทำงานด้านเอาท์พุต/อินพุต การปลดปล่อยอุปกรณ์ที่เชื่อมต่อในทางตรรก

พารามิเตอร์ตัวที่ 3 กำหนดเลขที่ของพอร์ตที่ถูกเชื่อมต่อในทางตรรก เป็นอุปกรณ์ที่ต้องการทำงานอินพุต หรือเอาท์พุต

พารามิเตอร์ตัวที่ 4 กำหนดจำนวนบิตในแต่ละตัวอักษร

พารามิเตอร์ตัวที่ 5 กำหนดจำนวนบิตปิดท้าย (stop bits) ในแต่ละตัวอักษร

พารามิเตอร์ตัวที่ 6 กำหนดประเภทของพาริตี้ ในการสร้างและตรวจสอบ

พารามิเตอร์ตัวที่ 7-11 กำหนดอัตราความเร็วในการรับ-ส่งข้อมูล

พารามิเตอร์ตัวที่ 12 กำหนดเงื่อนไขเพิ่มเติมในการรับ-ส่ง เช่น ข้อมูลเอาท์พุตไม่ต้องแปลงรหัสเป็นแบบแอสกี ข้อมูลอินพุตไม่ต้องแปลงรหัสเป็นแบบเอบซีดริก โดยไฮดรา-ทู เป็นต้น

<u>พารามิเตอร์เอาท์พุต</u>

พารามิเตอร์ตัวที่ 13 ถึง ตัวที่ 23 เป็นพารามิเตอร์ที่ใช้เก็บข้อสนเทศที่ใช้ในการทำงานด้านเอาท์พุต

พารามิเตอร์ตัวที่ 13 บรรจุตำแหน่งที่อยู่ของระเบียนที่เป็นเอาท์พุตที่ต้องการส่ง

พารามิเตอร์ตัวที่ 14 บรรจุความยาวของระเบียนเอาท์พุต

พารามิเตอร์ตัวที่ 15 บรรจุตัวอักษรที่ใช้เป็นสัญญลักษณ์บอกไฮดรา-ทู ให้หยุดการทำงานเอาท์พุตนั้น

พารามิเตอร์ตัวที่ 16 บรรจุขีดจำกัดของเวลาในการทำเอาท์พุต (output timeout)

พารามิเตอร์ตัวที่ 17 บรรจุตัวอักษรที่ใช้เป็น เอ็กซ์ออน/เอ็กซ์ออฟ โปรโตคอล

พารามิเตอร์ตัวที่ 18 จะบรรจุรหัสที่เป็นผลจากการทำเอาท์พุต บอกว่าสำเร็จ หรือเกิดข้อผิดพลาด

พารามิเตอร์ตัวที่ 19 จะบรรจุตำแหน่งที่อยู่ของข้อความที่เป็นข้อผิดพลาดที่เกิดขึ้นจากการทำเอาท์พุต

พารามิเตอร์ตัวที่ 20 จะบรรจุความยาวของข้อความที่เป็นข้อผิดพลาด

พารามิเตอร์ตัวที่ 21 จะบรรจุจำนวนตัวอักษรที่ไม่ได้ถูกส่งไปที่อุปกรณ์เอาท์พุต

พารามิเตอร์ตัวที่ 22 จะบรรจุรหัสที่แสดงสถานะของชาร์แนล

พารามิเตอร์ตัวที่ 23 จะบรรจุดีไวซ์เซนซ์ไบท์ (device sense byte) ซึ่งจะเป็นรหัสที่แสดงข้อผิดพลาดที่เกิดขึ้น

<u>พารามิเตอร์อินพุต</u>

พารามิเตอร์ตัวที่ 24 ถึง ตัวที่ 34 เป็นพารามิเตอร์ที่ใช้เก็บข้อสนเทศที่ต้องใช้ในการ

ทำงานด้านอินพุต

พารามิเตอร์ตัวที่ 24 บรรจุตำแหน่งที่อยู่ของเนื้อที่ที่ใช้เป็นอินพุต

พารามิเตอร์ตัวที่ 25 บรรจุความยาวของระเบียนอินพุต

พารามิเตอร์ตัวที่ 26 บรรจุตัวอักษรที่ใช้เป็นสัญญลักษณ์บอกไฮดรา-ทู ให้หยุดการทำงานอินพุตนั้น

พารามิเตอร์ตัวที่ 27 บรรจุขีดจำกัดของเวลาในการทำอินพุต (input timeout)

พารามิเตอร์ตัวที่ 28 บรรจุตัวอักษรที่ใช้เป็น เอ็กซ์ออน/เอ็กซ์ออฟ โปรโตคอล

พารามิเตอร์ตัวที่ 29 จะบรรจุรหัสที่เป็นผลจากการทำอินพุต บอกว่าสำเร็จ หรือเกิดข้อผิดพลาด

พารามิเตอร์ตัวที่ 30 จะบรรจุตำแหน่งที่อยู่ของข้อความที่เป็นข้อผิดพลาดที่เกิดขึ้นจากการทำอินพุต

พารามิเตอร์ตัวที่ 31 จะบรรจุความยาวของข้อความที่เป็นข้อผิดพลาด

พารามิเตอร์ตัวที่ 32 จะบรรจุจำนวนตัวอักษรที่ไม่ได้รับจากอุปกรณ์อินพุต

พารามิเตอร์ตัวที่ 33 จะบรรจุรหัสที่แสดงสถานะของชาร์แนล

พารามิเตอร์ตัวที่ 34 จะบรรจุดีไวซ์เซนซ์ไบท์ ซึ่งจะเป็นรหัสที่แสดงข้อผิดพลาดที่เกิดขึ้น

จากการศึกษาหลักการทำงานของโปรแกรมอาร์คอม/เอฟที ซอฟแวร์บนเมนเฟรมที่จะต้องทำการพัฒนาขึ้นเพื่อใช้ในการควบคุมการถ่ายแฟ้มข้อมูลระหว่างเมนเฟรมกับพีซีนั้น จะมีหลักการและแนวความคิดเช่นเดียวกับ โปรแกรมย่อยที่ทำหน้าที่ถ่ายข้อมูลรับ และส่ง ในฐานะที่เป็นตัวออกคำสั่ง

เมื่อพิจารณาขั้นตอนการทำงานของโปรแกรมย่อยดังกล่าวแล้ว ส่วนที่มีความสำคัญคือ การทำงานทางด้านอินพุตและเอาท์พุตกับพอร์ตการสื่อสาร

จากการศึกษาโดยเขียนโปรแกรมเล็กๆ ทำการเรียกใช้โปรแกรมย่อยเอชวาย 100 ให้ทำงานด้านอินพุตและเอาท์พุตกับพอร์ตของไฮดรา-ทู ซึ่งเป็นพอร์ตแบบซีเรียล อาร์เอส-232ซี และติดตามผลที่จะเกิดขึ้น พอสรุปได้ดังนี้

1. พอร์ตที่จะใช้เป็นอินพุต หรือเอาท์พุตนั้น จะต้องมีการกำหนดไว้เป็นพอร์ตอินพุตเอาท์

พุตทั่วไป กรณีที่พอร์ตนั้นไม่ได้เป็นพอร์ตอินพุตเอาท์พุตทั่วไป และต้องการทำอินพุต หรือเอาท์พุตกับพอร์ตนั้น จะต้องมีการใช้พอร์ตที่เป็นพอร์ตอินพุตเอาท์พุตทั่วไปมาเชื่อมต่อแบบตรรกไว้ (เสมือนใช้เป็นสพานเชื่อมทางตรรก) โดยจะทำให้สามารถทำงานด้านอินพุต และเอาท์พุตกับพอร์ตนั้นๆได้

พอร์ตที่เป็นพอร์ตอินพุตเอาท์พุตทั่วไป กำหนดไว้ในพารามิเตอร์ตัวที่ 1 ส่วนพอร์ตที่ใช้เป็นอินพุต และเอาท์พุต ก็กำหนดไว้ในพารามิเตอร์ตัวที่ 3 กรณีที่พอร์ตที่ต้องการใช้เป็นอินพุต และเอาท์พุตมีสภาพเป็นพอร์ตอินพุตเอาท์พุตทั่วไป ก็ให้กำหนดไว้ในพารามิเตอร์ตัวที่ 1 เท่านั้น ส่วนพารามิเตอร์ตัวที่ 3 ไม่ต้องใช้

2. ในการเรียกใช้โปรแกรมย่อยเอชวาย 100 ไม่ว่าจะเป็นอินพุต หรือเอาท์พุต จะต้องกำหนดลักษณะของการติดต่อสื่อสารกับพอร์ตนั้น ไว้ในพารามิเตอร์ทั่วไป ได้แก่ กำหนดจำนวนบิตต่อหนึ่งตัวอักษรไว้ในพารามิเตอร์ตัวที่ 4 กำหนดจำนวนบิตปิดท้ายไว้ในพารามิเตอร์ตัวที่ 5 กำหนดพาริตี้ไว้ในพารามิเตอร์ตัวที่ 6 กำหนดอัตราความเร็วในการอินพุตเอาท์พุตไว้ในพารามิเตอร์ตัวที่ 7 ถึง ตัวที่ 11 และกำหนดลักษณะพิเศษอื่นๆไว้ในพารามิเตอร์ตัวที่ 12 เช่น อินพุต หรือเอาท์พุตไม่ต้องทำการแปลงรหัสข้อมูล การใช้รหัสพิเศษ หรือขีดจำกัดของเวลา เป็นตัวกำหนดการสิ้นสุดในการทำอินพุต หรือเอาท์พุต เป็นต้น

กรณีที่เรียกใช้เอชวาย 100 ในครั้งแรกสุด หรือเมื่อมีการเปลี่ยนแปลงค่าที่กำหนดในพารามิเตอร์ทั่วไป ซึ่งเป็นลักษณะของการสื่อสาร โปรแกรมเอชวาย 100 จะส่งคำสั่งชาร์แนล สั่งงานให้ไฮดรา-ทู รับรู้ และเชื่อมต่อกับพอร์ตด้วยลักษณะที่ได้กำหนดไว้ในพารามิเตอร์ต่างๆ

3. การเรียกใช้เอชวาย 100 เพื่อทำการอินพุตกับพอร์ต ต้องกำหนดค่าต่างๆในพารามิเตอร์สำหรับอินพุต (พารามิเตอร์ตัวที่ 24 - 34) และกำหนดการทำงานเป็นการอินพุตไว้ในพารามิเตอร์ตัวที่ 2 หลังจากเรียกใช้การทำงานนี้แล้ว ต้องเรียกใช้เอชวาย 100 ให้ทำงานหยุดรอจนกว่าการทำงานอินพุตที่ร้องขอก่อนหน้าเสร็จสิ้น โดยกำหนดการทำงานเป็นการรอคอยไว้ในพารามิเตอร์ตัวที่ 2 หลังจากเรียกใช้การทำงานนี้แล้ว เมื่อการอินพุตเสร็จสิ้น ไฮดรา-ทูจะส่งผลของการอินพุตกลับมาในรูปของรหัสไว้ในพารามิเตอร์ตัวที่ 29 เพื่อบ่งบอกว่าการอินพุตประสพผลสำเร็จ หรือเกิดข้อผิดพลาดขึ้น ซึ่งในส่วนของโปรแกรมผู้ใช้ ต่อมาก็สามารถตรวจสอบได้

4. การเรียกใช้เอชวาย 100 เพื่อทำการเอาท์พุตกับพอร์ต ต้องกำหนดค่าต่างๆ ใน

พารามิเตอร์สำหรับเอาท์พุต (พารามิเตอร์ตัวที่ 13 - 23) และกำหนดการทำงานเป็นการเอาท์พุตไว้ในพารามิเตอร์ตัวที่ 2 หลังจากเรียกใช้การทำงานนี้แล้ว ต้องเรียกใช้เอชวาย 100 ให้ทำงานหยุดรอจนกว่าการทำงานเอาท์พุตที่ร้องขอก่อนหน้าเสร็จสิ้น โดยกำหนดการทำงานเป็นการรอคอยไว้ในพารามิเตอร์ตัวที่ 2 หลังจากเรียกใช้การทำงานนี้แล้ว เมื่อการเอาท์พุตเสร็จสิ้น ไฮดรา-ทูจะส่งผลของการเอาท์พุตกลับมาในรูปของรหัสไว้ในพารามิเตอร์ตัวที่ 18 เพื่อบ่งบอกว่าการเอาท์พุตประสบผลสำเร็จ หรือเกิดข้อผิดพลาดขึ้น ซึ่งในส่วนของโปรแกรมผู้ใช้ ต่อมาก็สามารถตรวจสอบได้

5. เมื่อต้องการเลิกการอินพุต หรือเอาท์พุตกับพอร์ต จะต้องปลดปล่อยพอร์ตนั้นให้เป็นอิสระเพื่อทำงานตามปรกติได้ ก็เรียกใช้เอชวาย 100 โดยกำหนดการทำงานเป็นการปลดปล่อยไว้ในพารามิเตอร์ตัวที่ 2 หลังจากเรียกใช้การทำงานนี้แล้ว ต้องเรียกใช้เอชวาย 100 ให้ทำงานหยุดรอจนกว่าการปลดปล่อยพอร์ตจะเสร็จสิ้น เมื่อการปลดปล่อยเสร็จสิ้น ไฮดรา-ทูจะส่งผลที่เกิดในรูปของรหัสไว้ที่พารามิเตอร์ตัวที่ 18 เพื่อบ่งบอกว่า การปลดปล่อยพอร์ตประสบผลสำเร็จ หรือเกิดข้อผิดพลาดขึ้น

ส่วนรายละเอียดเกี่ยวกับความสามารถอื่นๆ และวิธีการเรียกใช้โปรแกรมย่อยเอชวาย 100 มีรายละเอียดอยู่ในหนังสือคู่มือที่มาพร้อมกับไฮดรา-ทู