บทที่ 3

การออกแบบและสร้างเครื่องมือตรวจสอบโพรโทคอลชนิด SDLC

3.1 ความสามารถของเครื่องมือตรวจสอบโพรโทคอลที่ต้องการ

ความสามารถของเครื่องมือตรวจสอบโพรโทคอลที่ทำการออกแบบและสร้างขึ้นมีดังนี้คือ

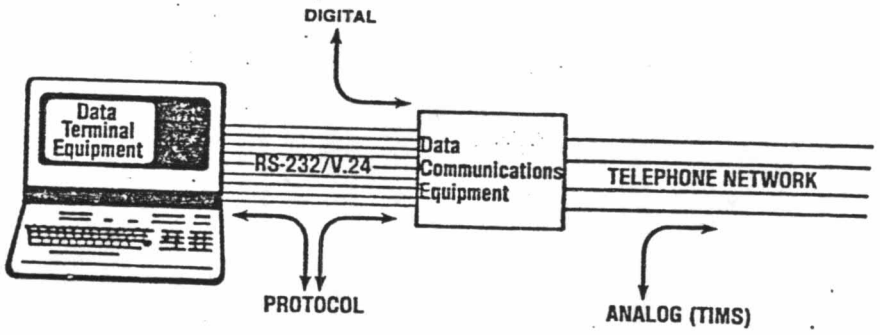
- ใช้ตรวจสอบโพรโทคอลชนิด SDLC
- มีลักษณะในการแสดงผลเป็นแบบ Layer1 และ Layer2 (รายละเอียดดูในบทที่ 4)
- มีอุปกรณ์ช่วยในการเก็บข้อมูล
- พิมพ์ผลที่ตรวจสอบออกทางเครื่องพิมพ์ได้
- ตรวจสอบโพรโทคอลทั้งในโมดฮาล์ฟดูเพลกซ์และฟูลดูเพลกซ์

3.2 หลักการในการออกแบบ (3)

สิ่งที่ต้องคำนึงถึงในการออกแบบคือจุดที่นำเครื่องมือตรวจสอบโพรโทคอลเข้าไปต่อ ในที่นี้จุดที่นำเข้าไปต่อคือจุดที่อยู่ระหว่าง DTE (Data Terminal Equipment) และ DCE (Data Circuit Terminating Equipment) ดังรูปที่ 3.1 สัญญาณที่อยู่ระหว่างจุดนั้นมีขนาดของแรงดันไฟตรงเท่ากับ ±12 โวลต์ เป็นสัญญาณเชิงเลข จุดประสงค์ของการต่ออุปกรณ์ตรวจสอบเข้าที่จุดนี้เพราะเป็นสัญญาณที่มีความแน่นอนสูง เมื่อมองจาก DTE จะเป็นสัญญาณที่ออกมาจากเครื่องคอมพิวเตอร์โดยตรงแต่ถ้ามองจาก DCE จะเป็นสัญญาณที่ได้จากการแปลงมาจากสัญญาณแอนะลอก (analog) ส่วนสัญญาณนาฬิกาก็ได้มาจากภายในระบบนั่นเอง ดังนั้นในการสุ่ม (sampling) ข้อมูลมาตรวจสอบก็สามารถสุ่มได้ กึ่งกลางของข้อมูลซึ่งทำให้เกิดความคลาดเคลื่อนได้ยาก อีกประการหนึ่งก็คือไม่ต้องมีส่วนแปลงสัญญาณแอนะลอกเป็นสัญญาณเชิงเลขภายใน

013036

เครื่องมือตรวจสอบ ซึ่งมีความสลับซับซ้อนพอสมควร เพราะว่าจะต้องมีการ coupling สัญญาณ มาจากคู่สายโทรศัพท์และวงจรที่ใช้ภายในเครื่องมือตรวจสอบกับโมเดม (modem) เป็นชุดที่ต่างกันย่อมเกิดความผิดพลาดในการแปลงข้อมูลได้ง่าย



รูปที่ 3.1 จุดที่นำข้อมูลออกมาตรวจสอบ

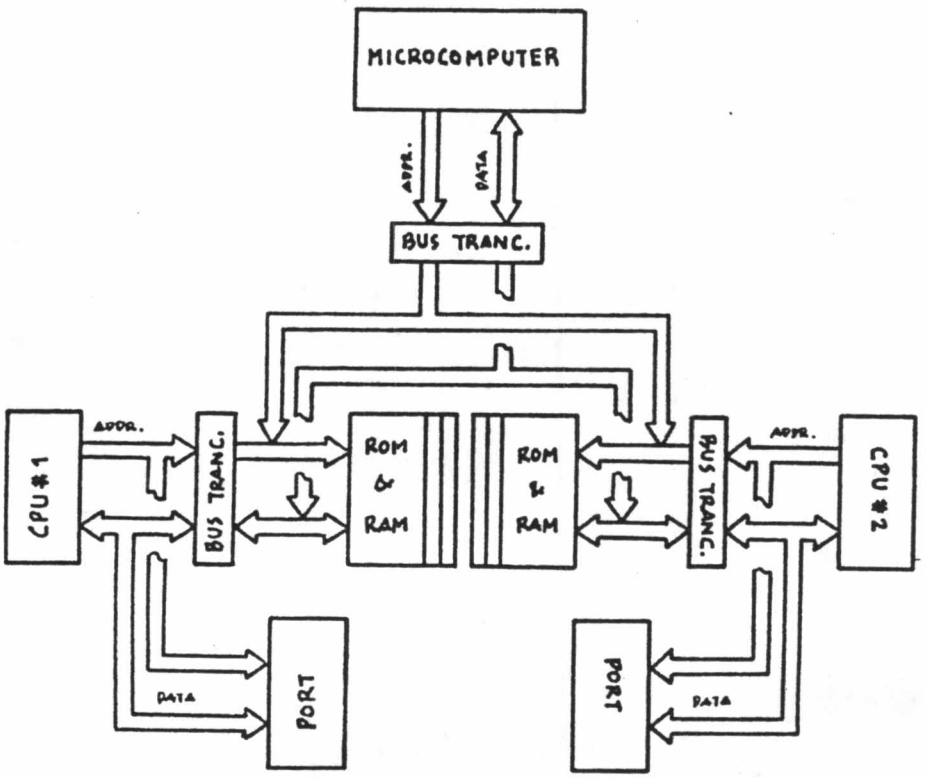
เพื่อที่จะให้เครื่องมือตรวจสอบโพรโทคอลที่สร้างขึ้นนี้มีความสามารถเป็นไปตามข้อที่กำหนดไว้ ในส่วนของการแสดงผล อุปกรณ์ช่วยในการเก็บข้อมูลและพิมพ์ผลออกทางเครื่องพิมพ์ จึงได้ออกแบบให้เครื่องมือตรวจสอบโพรโทคอลที่สร้างขึ้นนี้ทำงานร่วมกับเครื่องไมโครคอมพิวเตอร์ IBM PC/XT จุดประสงค์ที่ออกแบบเช่นนี้เพราะว่า

- ลดฮาร์ดแวร์ในส่วนของการแสดงผล พิมพ์ผล และบันทึกข้อมูล
- สามารถนำเครื่องไมโครคอมพิวเตอร์มาใช้ประโยชน์ได้มากขึ้น
- สถาปัตยกรรมทางด้านฮาร์ดแวร์ของเครื่องไมโครคอมพิวเตอร์รุ่นนี้ได้รับการเปิดเผยพอสมควร (4,5)
- นำฮาร์ดแวร์บางอย่างที่มีอยู่ในเครื่องไมโครคอมพิวเตอร์มาใช้ประโยชน์ได้
- เครื่องไมโครคอมพิวเตอร์รุ่นนี้เป็นรุ่นที่มีการใช้งานอย่างแพร่หลาย

ดังนั้นเครื่องมือตรวจสอบโพรโทคอลที่สร้างขึ้นจึงได้ถูกออกแบบเป็นการ์ดสำหรับเสียบลงไปบนบอร์ดของเครื่องไมโครคอมพิวเตอร์ โดยจะมีส่วนที่ใช้ในการตรวจสอบโพรโทคอลอยู่บนการ์ด แต่ส่วนที่ใช้ในการแสดงผล พิมพ์ผล และบันทึกข้อมูล ถูกจัดการโดยเครื่องไมโครคอมพิวเตอร์

3.3 การออกแบบส่วนฮาร์ดแวร์ (6,7)

แผนภาพกรอบของ เครื่องมือตรวจสอบโพรโทคอลที่ออกแบบแสดงดังรูปที่ 3.2 ซึ่งประกอบด้วยส่วนใหญ่ ๆ 3 ส่วนคือ ส่วนที่เป็น เครื่องไมโครคอมพิว เตอร์ ส่วนของซีพียู1 และส่วนของซีพียู2 โดยส่วนทั้งหมดที่อยู่ในรูปนี้ยก เว้นส่วนที่ เป็น เครื่องไมโครคอมพิว เตอร์ได้ถูกออกแบบให้อยู่บนการ์ดที่ใช้ เสียบและทำงานร่วมกับ เครื่องไมโครคอมพิว เตอร์ ส่วนของซีพียู1 และ2 ได้ออกแบบให้มีลักษณะการทำงานคล้ายกันมากและการทำงานนี้จะไม่ขึ้นต่อกัน แต่จะแตกต่างกันเพียง เล็กน้อย เท่านั้นซึ่งขอกล่าวในภายหลัง จุดประสงค์ที่ออกแบบให้ใช้ซีพียู 2 ตัวนั้น เพื่อต้องการให้ เครื่องมือตรวจสอบโพรโทคอลนี้สามารถตรวจสอบโมดของการรับส่งข้อมูลทั้งที่ เป็นชนิดฮาล์ฟ-ดูเพลกซ์ และฟูลดูเพลกซ์ การออกแบบในส่วนต่าง ๆ มีดังต่อไปนี้



รูปที่ 3.2 แผนภาพกรอบของ เครื่องมือตรวจสอบโพรโทคอล SDLC

3.3.1 ส่วนซีพียู ในการออกแบบนี้จะเลือกใช้ซีพียูเบอร์ Z-80 ทั้งซีพียู1 และ2 จุดประสงค์ในการเลือกซีพียู Z-80 (8) เพราะพอร์ทและหน่วยความจำที่จะนำมาใช้มีขาแอดเดรสแยกออกจากขาข้อมูลจึงไม่จำเป็นจะต้องใช้ซีพียูชนิดที่ขาแอดเดรสและข้อมูลถูกมัลติเพลกซ์ (multiplex) มาด้วยกัน ซึ่งจะมีผลทำให้อุปกรณ์ทางด้านฮาร์ดแวร์เพิ่มขึ้น คำสั่งของซีพียู Z-80 แต่ละคำสั่งมีช่วงเวลากำกับไว้ทำให้สามารถคำนวณเวลาที่ใช้ไปในการดำเนินโปรแกรมต่าง ๆ ได้ ซึ่งจะทำให้รู้ถึงขีดจำกัดของเครื่องมือตรวจสอบโพรโทคอลที่สร้างขึ้น ซีพียู Z-80 จะมีหน้าที่ในการนำข้อมูลเก็บลงในหน่วยความจำแต่ถ้าข้อมูลที่ตรวจสอบได้เป็นรหัสต่าง ๆ เช่น แฟลก FCS ฯลฯ จะต้องแปลงรหัสให้อยู่ในรูปที่เหมาะสมแล้วจึงเก็บลงในหน่วยความจำ

3.3.2 ส่วนพอร์ท ในการออกแบบนี้จะเลือกใช้พอร์ทเบอร์ 8273 (protocol controller) (รายละเอียดของหน้าที่และฟังก์ชันการทำงานอยู่ในภาคผนวก ก) เป็นพอร์ทที่ใช้ในการตรวจสอบโพรโทคอลชนิด SDLC จุดประสงค์ในการเลือกใช้ 8273 (9) เพราะเป็นพอร์ทที่มีความสามารถในการตรวจสอบโพรโทคอลชนิด SDLC ได้ครบถ้วน กล่าวคือ พอร์ท 8273 สามารถแยกเขตต่าง ๆ ภายใน 1 เฟรมของโพรโทคอล SDLC ได้ สามารถดึงบิตศูนย์ออกจากการแทรกบิตศูนย์มาจากต้นทางได้ สามารถตรวจสอบว่าเฟรมที่รับได้นั้นถูกต้องหรือมีข้อผิดพลาด สามารถตรวจสอบรหัสต่าง ๆ เช่น abort ได้ ฯลฯโดยจะรับข้อมูลที่อยู่ในรูปอนุกรมซึ่งเป็นจุดที่นำโพรโทคอลออกมาตรวจสอบและแปลงข้อมูลนั้นให้อยู่ในรูปขนานเพื่อง่ายต่อการประมวลผลของซีพียู พอร์ท 8273 จะบอกสถานะต่าง ๆที่เกิดขึ้นแก่ซีพียูโดยการขัดจังหวะซึ่งทำให้ลดเวลาการประมวลผลของซีพียูได้อย่างมาก

3.3.3 หน่วยความจำ ส่วนนี้ใช้ในการเก็บโปรแกรมที่ใช้ในการตรวจสอบโพรโทคอลและใช้ในการเก็บข้อมูลโดยจะแบ่งดังนี้คือ หน่วยความจำที่เป็น ROM (Read Only Memory) หน่วยความจำที่เป็น RAM หลัก (Random Access Memory) และหน่วยความจำที่เป็น RAM สำรอง ส่วนที่เป็น ROM จะถูกอ้างโดยซีพียู Z-80 เท่านั้นโดยมีแอดเดรสอยู่ในช่วง 0000H-1FFFH ใช้เก็บโปรแกรมที่ใช้ในการตรวจสอบโพรโทคอลชนิด SDLC ส่วนที่เป็น RAM หลักจะถูกอ้างโดยซีพียู Z-80 และซีพียู 8088 ที่อยู่บนเครื่องไมโครคอมพิวเตอร์แต่ในเวลาที่ไม่พร้อมกัน การอ้างนี้จะถูกอ้างโดยผ่าน bus transceiver (ซึ่งจะกล่าวต่อไป) RAM หลักนี้จะมีแอดเดรส

อยู่ในช่วง 2000H-9FFFH เมื่อมองจากซีพียู Z-80 ของทั้งสองตัว แต่จะมีแอดเดรสอยู่ในช่วง 40000H-4FFFFH เมื่อมองจากซีพียู 8088 ส่วนที่เป็น RAM สำรองจะถูกอ้างโดยซีพียู Z-80 เท่านั้นโดย มีแอดเดรสอยู่ในช่วง A000H-BFFFH ใช้ในการเก็บข้อมูลเช่นเดียวกับ RAM หลัก แต่ในเวลาที่ไม่พร้อมกัน นอกจากนี้ RAM สำรองยังใช้เป็นสแตคของซีพียู Z-80 ด้วย จุดประสงค์ที่ต้องมีการกำหนด RAM เป็น 2 ช่วง เพราะต้องการให้การตรวจสอบโพรโทคอลเป็นไปอย่างต่อเนื่องซึ่งอธิบายได้ดังนี้ เมื่อซีพียู 8088 ทำการย้ายข้อมูลจากหน่วยความจำหลักของซีพียู Z-80 ไปยังหน่วยความจำของเครื่องไมโครคอมพิวเตอร์ที่ใช้ในการแสดงผล จะต้องใช้เวลาช่วงหนึ่ง ถ้าขณะนั้นมีข้อมูลเข้ามาตลอดเวลา จะเห็นได้ว่าข้อมูลในช่วงนั้นจะสูญหายไปถ้าไม่มีการกำหนด RAM สำรองในการช่วยเก็บข้อมูลในช่วงเวลานี้ ส่วนการกำหนดแอดเดรสจากซีพียู 8088 ที่ต้องมีการกำหนดในตำแหน่งนี้เพราะ เครื่องไมโครคอมพิวเตอร์ IBM PC/XT ใช้ซีพียู 8088 ซึ่งสามารถอ้างแอดเดรสได้ทั้งหมด 1 เมกกะไบต์ แต่สามารถอ้างหน่วยความจำที่ใช้ในการประมวลผลได้ 640 กิโลไบต์ โดยแบ่งเป็น 256 กิโลไบต์อยู่บน system board อีก 384 กิโลไบต์เป็นส่วนขยายที่เพิ่มขึ้นซึ่งไม่อยู่บน system board ส่วนแอดเดรสต่าง ๆ ที่เหลือนั้นถูกใช้ไปในส่วนของ monochrome display, color display, ROM BIOS ฯลฯ และบางส่วนก็มีการจองไว้ (reserved) ในส่วนที่เหลือก็ไม่สามารถใช้ได้เพราะ ช่วงแอดเดรสที่เหลือนั้นไม่มีความต่อเนื่องกันโดยตลอดซึ่งจะทำให้การอ้างแอดเดรสมีความยุ่งยากมาก แต่ในขณะเดียวกันโปรแกรมที่เขียน ขึ้นเพื่อใช้ในการแสดงผลและบันทึกข้อมูลมีขนาดไม่ใหญ่มากซึ่งพอที่จะดำเนินภายใต้หน่วยความจำขนาด 256 กิโลไบต์ได้ ดังนั้นจึงเลือกใช้แอดเดรสที่อยู่ต่อจาก 256 กิโลไบต์นี้

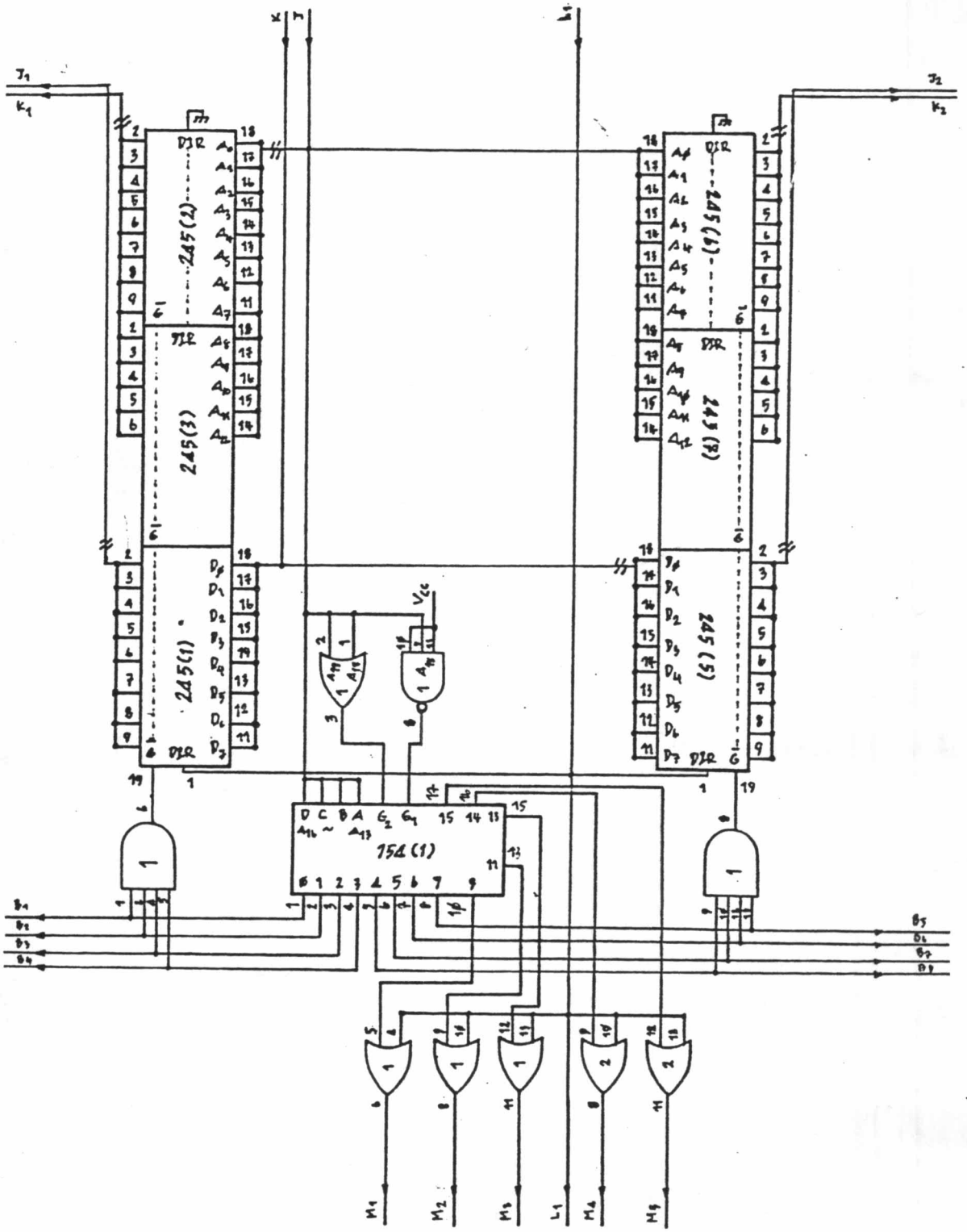
3.3.4 ส่วน bus transceiver จากรูป 3.2 จะเห็นได้ว่า bus transceiver นี้ต่ออยู่กับส่วนไมโครคอมพิวเตอร์ ส่วนซีพียู1 และส่วนซีพียู2 bus transceiver ใช้ในการอ้างหน่วยความจำโดยมีลำดับในการทำงานดังนี้คือ ถ้า bus transceiver ในส่วนของเครื่องไมโครคอมพิวเตอร์ทำงาน (active) bus transceiver ในส่วนของซีพียู1 และซีพียู2 จะไม่ทำงาน (idle) ในลักษณะนี้จะเป็นการเคลื่อนย้ายข้อมูลจาก RAM หลักของซีพียู1 และ2 ไปยัง RAM ของเครื่องไมโครคอมพิวเตอร์ ในทางกลับกันถ้า bus transceiver ในส่วนของเครื่องไมโครคอมพิวเตอร์ไม่ทำงาน (idle) และ bus transceiver ในส่วนของซีพียู1 และ2 ทำงาน (active) จะเป็นลักษณะของการเก็บข้อมูลลงใน RAM หลักของซีพียู Z-80 ทั้ง

สองตัว ดังนั้นจะเห็นได้ว่า bus transceiver ทำหน้าที่ในการอ้างแอดเดรสในส่วนของ RAM หลักจากซีพียูทั้ง 3 โดยมีลำดับการทำงานดังกล่าวข้างต้น

อนึ่งในการย้ายข้อมูลจากหน่วยความจำหลักของซีพียู Z-80 นั้นจะเกิดการย้ายข้อมูลขึ้นก็ต่อเมื่อซีพียู 8088 ได้รับสัญญาณจากซีพียู Z-80 ดังนั้นในการออกแบบนี้จึงกำหนดให้ส่วนของซีพียู1 มีความแตกต่างจากส่วนของซีพียู2 ตรงที่ ซีพียู1 สามารถส่งสัญญาณไปยังซีพียู 8088 ให้มาทำการย้ายข้อมูลได้ ในขณะที่ซีพียู2ไม่สามารถทำได้

3.4 รายละเอียดในส่วนฮาร์ดแวร์

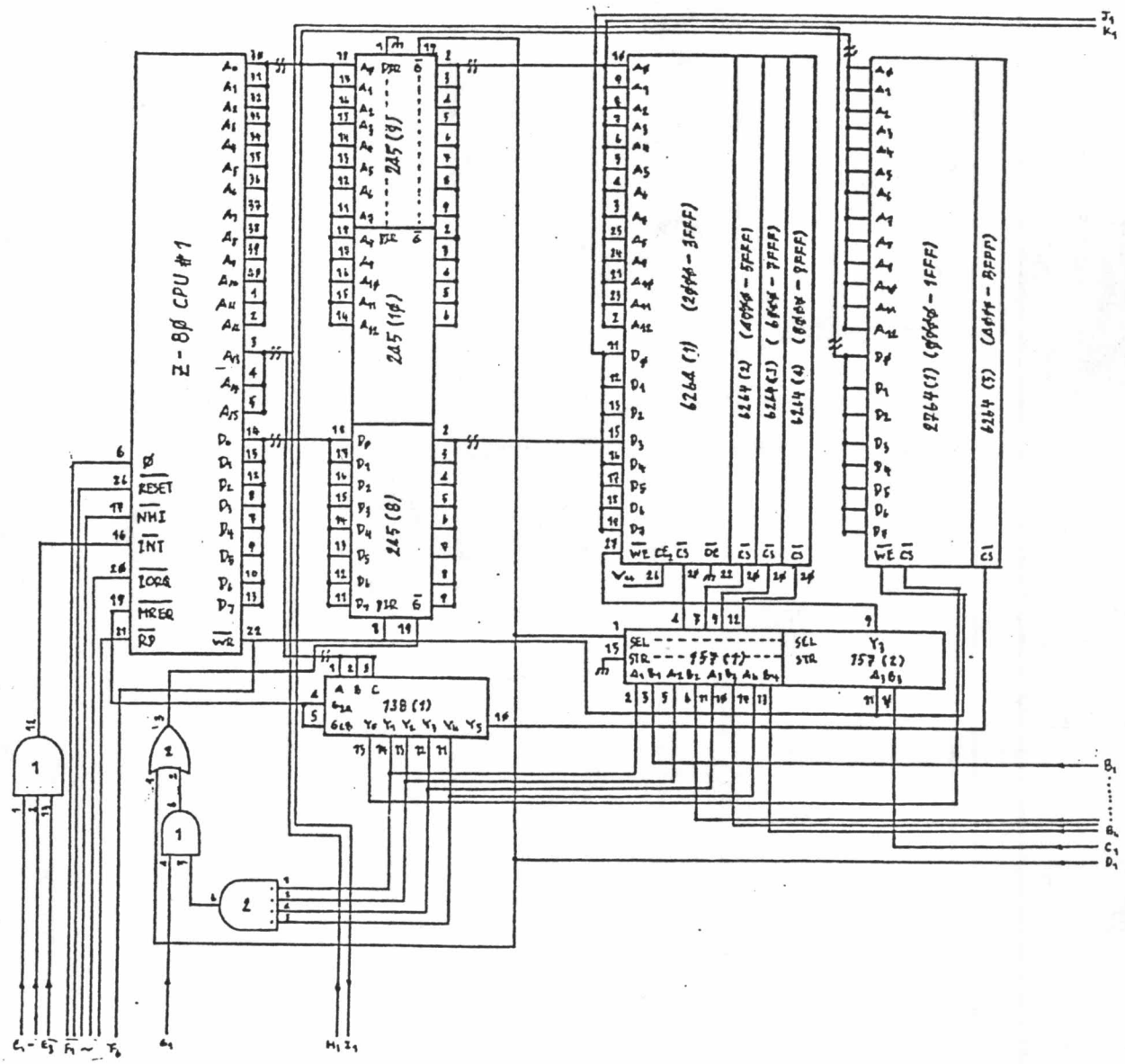
รูปที่ 3.3 เป็นฮาร์ดแวร์ที่ 8088 ใช้ติดต่อกับหน่วยความจำหลักของซีพียู1 และหน่วยความจำหลักของซีพียู2 การติดต่อนี้จะเกิดขึ้นก็ต่อเมื่อซีพียู 8088 ทำการย้ายข้อมูลจากหน่วยความจำหลักซีพียูทั้งสองไปเก็บในหน่วยความจำของเครื่องไมโครคอมพิวเตอร์ เพื่อใช้ในการแสดงผล ในส่วนของ bus transceiver จะถูกแทนด้วยไอซีเบอร์ 74LS245 ที่ใช้ไอซีเบอร์นี้เพราะ 74LS245 เป็นบัส 2 ทาง (bi-directional) ภายในตัวเดียวกัน แอดเดรสและข้อมูลของหน่วยความจำหลักของซีพียู1 จะถูกอ้างโดยผ่าน 74LS245 (2,3) และ 74LS245 (1) ตามลำดับ ในส่วนของซีพียู2 ก็เช่นเดียวกันจะถูกอ้างโดยผ่าน 74LS245 (6,7) และ 74LS245 (5) ตามลำดับ 74LS154 ใช้ในการอ้างช่วงของหน่วยความจำที่จะเลือก โดยช่วงหนึ่ง ๆ จะอ้างได้ 8 กิโลไบต์ ส่วน OR gate และ NAND gate ที่ต่ออยู่กับ 74LS154 นั้นใช้ในการกำหนดช่วงของหน่วยความจำให้อยู่ในช่วง 40000H-4FFFFH



รูปที่ 3.3 ฮาร์ดแวร์ที่ใช้ติดต่อกับซีพียู3 (8088)

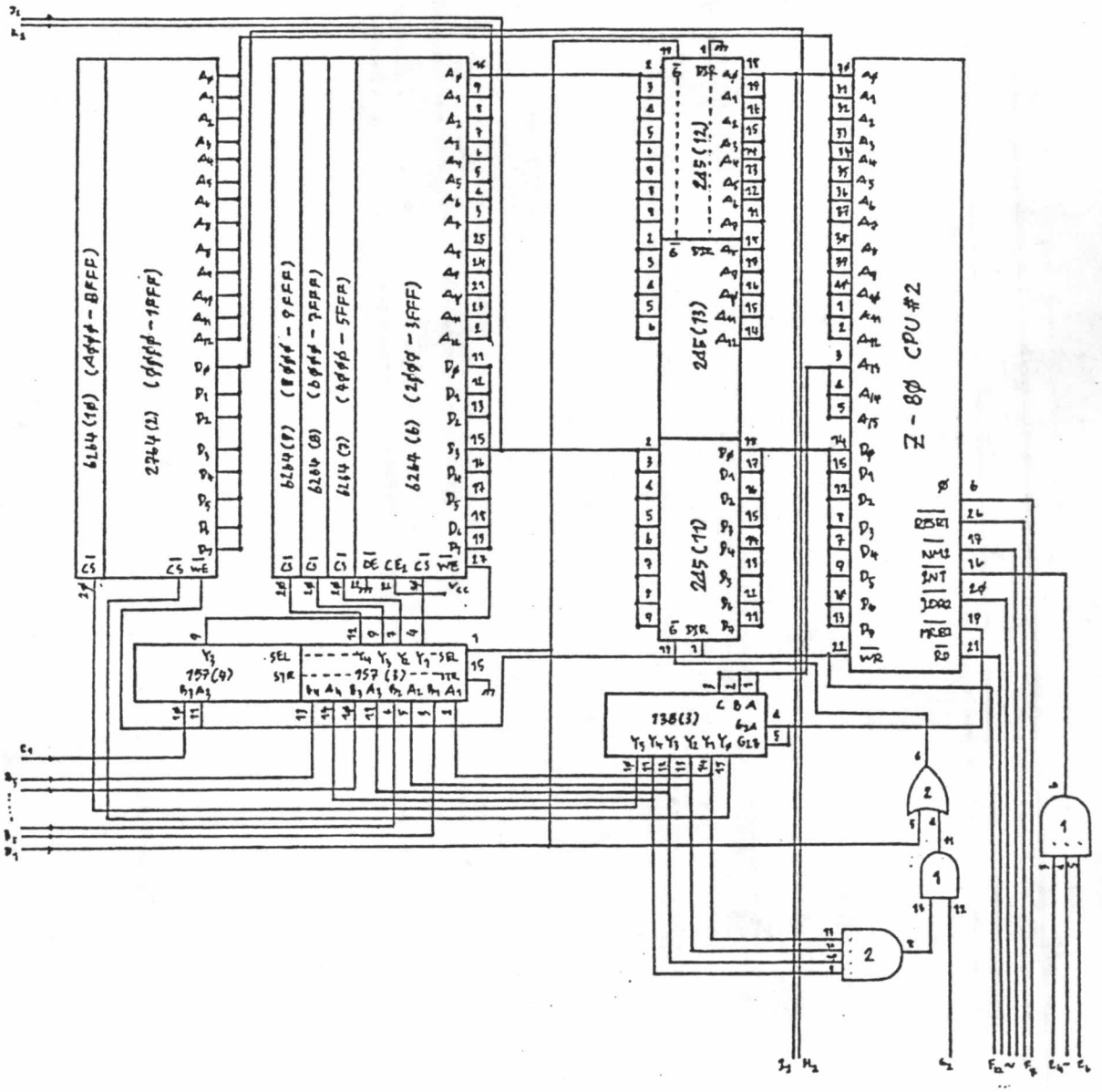
รูปที่ 3.4 เป็นฮาร์ดแวร์ที่ใช้ในการตรวจสอบโพรโทคอลในส่วนเก็บข้อมูล ในที่นี้กำหนดให้ซีพียู1 ตรวจสอบโพรโทคอลทางด้าน DTE ในรูปจะเห็นได้ว่าซีพียู Z-80 สามารถอ้างหน่วยความจำได้ 2 แบบคือ อ้างโดยตรง อ้างโดยผ่าน bus transceiver ในส่วนที่อ้างโดยตรงนั้นจะอ้างได้เฉพาะ ROM และ RAM ที่ตำแหน่ง A000H-BFFFH เท่านั้น ในส่วนที่อ้างโดยผ่าน bus transceiver นั้นจะอ้างได้เฉพาะ RAM ในช่วง 2000H-9FFFH เท่านั้น ROM และ RAM ที่ใช้แต่ละตัวจะมีความจุ 8 กิโลไบต์ 74LS138 ใช้ในการเลือกช่วงของหน่วยความจำ โดยช่วงหนึ่ง ๆ ที่อ้างได้มีความจุ 8 กิโลไบต์ การอ้างแอดเดรสนี้ได้รับสัญญาณจากแอดเดรส 3 บิตบนของซีพียู Z-80 รวมกับสัญญาณ MREQ ซึ่งใช้ในการถอดรหัสและ enable ตามลำดับ 74LS157 ใช้ในการกำหนดสัญญาณควบคุม (CS,WR) ให้กับหน่วยความจำ โดยมีลำดับการทำงานดังนี้ ถ้าอยู่ในช่วงของการเคลื่อนย้ายข้อมูลจากหน่วยความจำหลักไปยังหน่วยความจำของเครื่องไมโครคอมพิวเตอร์สัญญาณควบคุมนี้จะมาจากซีพียู 8088 ถ้าอยู่ในช่วงของการเก็บข้อมูลลงหน่วยความจำโดยซีพียู Z-80 สัญญาณนี้จะมาจากซีพียู Z-80

หอสมุดกลาง สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 3.4 หน่วยความจำที่ใช้เก็บข้อมูลทางด้าน DTE

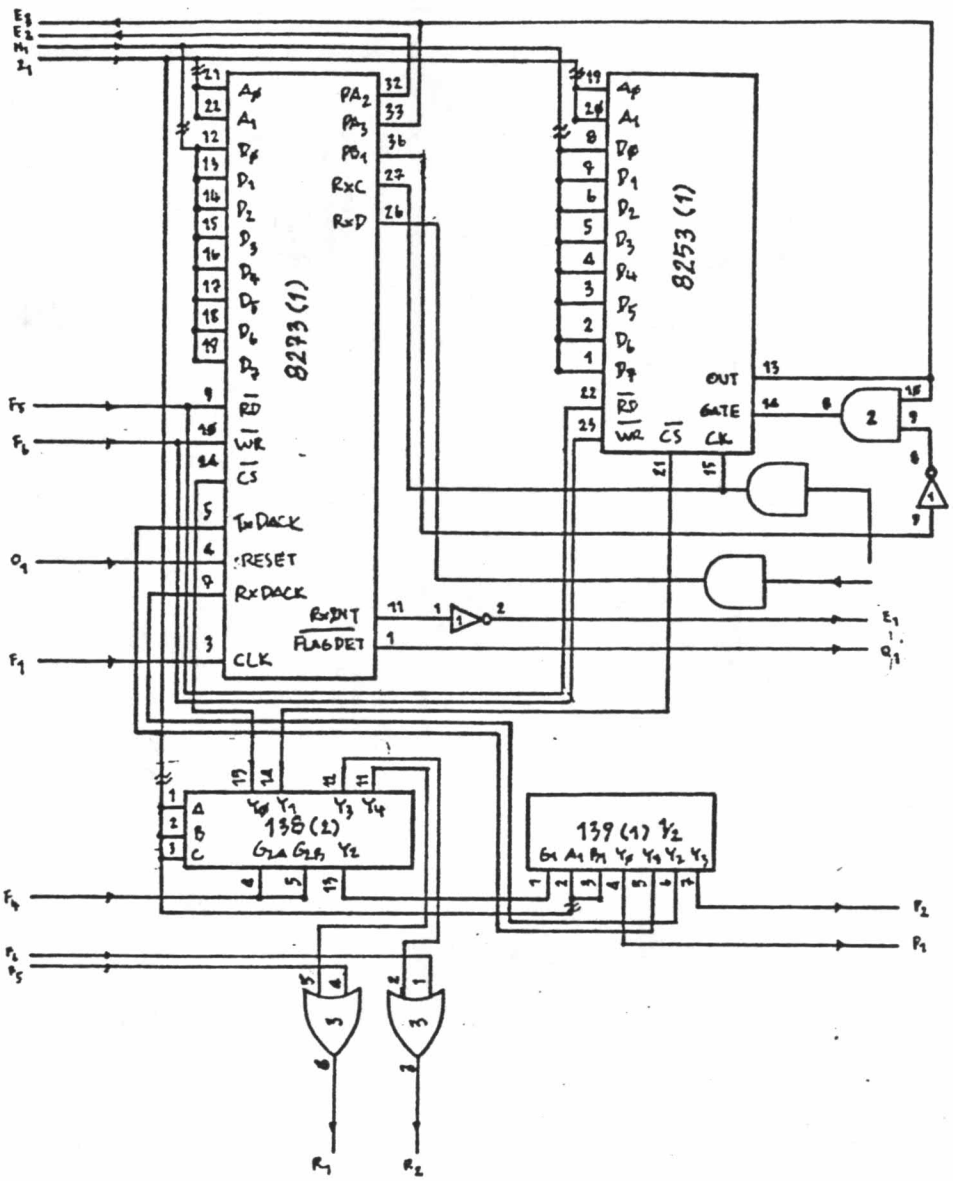
รูปที่ 3.5 เป็นฮาร์ดแวร์ที่ใช้ในการตรวจสอบโพรโทคอลโดยกำหนดให้ซีพียู2 ตรวจสอบโพรโทคอลทางด้าน DCE ฮาร์ดแวร์ในส่วนนี้จะมีการทำงานเช่นเดียวกับฮาร์ดแวร์ในรูปที่ 3.4 แต่จะแตกต่างกันที่หมายเลขที่ใช้ในการกำหนดเบอร์ไอซีเท่านั้น



รูปที่ 3.5 หน่วยความจำที่ใช้เก็บข้อมูลทางด้าน DCE

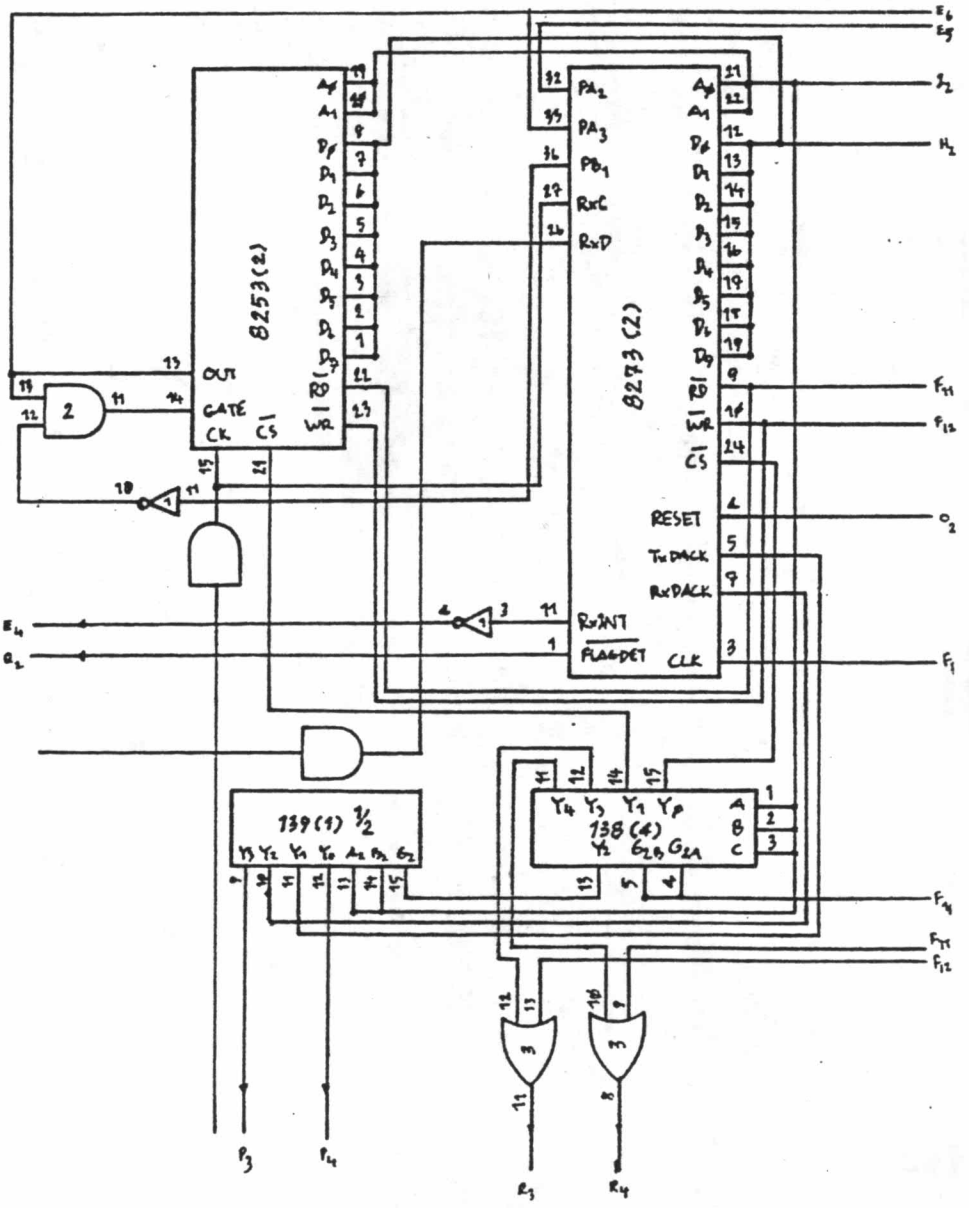
รูปที่ 3.6 เป็นพอร์ทที่ใช้ในการตรวจสอบโพรโทคอลทางด้าน DTE โดยที่ 8273 จะรับข้อมูลที่อยู่ในรูปอนุกรมเข้าทางขา RxD ส่วนสัญญาณนาฬิกาจะรับเข้าทางขา RxC แล้วจึง

ทำการประมวลผลการตรวจสอบโพรโทคอลภายในตัว 8273 ผลที่ได้จะมี 2 ทางคือ ถ้าการตรวจสอบพบว่าเป็นแฟลก 8273 จะส่งสัญญาณออกที่ขา FLAG DET แต่ถ้าเป็นกรณีอื่น ๆ เช่น ตรวจพบข้อมูล ตรวจพบ abort ฯลฯ 8273 จะส่งสัญญาณออกที่ขา RxINT สัญญาณทั้งสองนี้จะเกิดขึ้นไม่พร้อมกันแต่จะไปเข้าที่ขา INT ของซีพียู Z-80 เหมือนกับ 8253 (10) ใช้ช่วย 8273 ในการตรวจสอบสถานะ idle ทั้งนี้เนื่องจาก8273 มีข้อจำกัดที่ว่า ถ้า 8273 ตรวจพบสถานะ idle จะหยุดทำงานทันทีหลังจากแจ้งสถานะไปยังซีพียูเรียบร้อยแล้ว



รูปที่ 3.6 พอร์ทที่ใช้ในการตรวจสอบโพรโทคอลทางด้าน DTE

รูปที่ 3.7 เป็นพอร์ทที่ใช้ในการตรวจสอบโพรโทคอลทางด้าน DCE ซึ่งมีการทำงานเช่นเดียวกับพอร์ทในรูปที่ 3.6 แต่จะแตกต่างกันตรงที่หมายเลขกำกับเบอร์ไอซีเท่านั้น



รูปที่ 3.7 พอร์ทที่ใช้ในการตรวจสอบโพรโทคอลทางด้าน DCE

ตำแหน่งของพอร์ทต่าง ๆ ถูกกำหนดไว้ในตารางที่ 3.1 ไอซีเบอร์ 74LS670 (register file) ใช้เป็นพอร์ทที่ใช้แจ้งสถานะของซีพียู1 และ 2 ทำให้ซีพียูทั้งสองตัวรู้สถานะซึ่งกันและกัน

ตารางที่ 3.1 ตำแหน่งของพอร์ทในเครื่องมือตรวจสอบโพรโทคอล SDLC

เบอร์ไอซี	เบอร์พอร์ท	รีจิสเตอร์
8273	00-03	00 = command, status register
	09-0A	01 = parameter, result register
		02 = reset, TxINT result register
		03 = RxINT result register
		09 = Tx data register
		0A = Rx data register
8253	04-07	04-06 = timer 1-3
		07 = command
670(1)	0C-0F	0C-0F ตำแหน่งของ word ที่ 1-4
670(2)	10-13	10-13 ตำแหน่งของ word ที่ 1-4

3.5 การออกแบบทางด้านซอฟต์แวร์ (software)

3.5.1 ซอฟต์แวร์ที่ใช้ในการตรวจสอบ เนื่องจากซีพียูที่ใช้ในการตรวจสอบโพรโทคอลเป็น Z-80 ดังนั้นจึงใช้ภาษาแอสเซมบลี(11) (assembly) เป็นโปรแกรมควบคุมการทำงานทั้งหมดในการตรวจสอบโดยจะตรวจสอบเขตต่างๆที่อยู่ภายในเฟรมของ SDLC นอกจากนี้ยังมีสถานะบางชนิดที่ต้องการตรวจสอบคือ idle,abort ฯลฯ (ภาคผนวก ค เป็นโปรแกรมภาษาแอสเซมบลีของซีพียู Z-80 ที่ใช้ในการตรวจสอบทางด้าน DTE และ DCE) ในที่นี้จะกล่าวเฉพาะโปรแกรมย่อยเท่านั้น

ตารางที่ 3.2 ส่วนสำคัญของโปรแกรมที่ใช้ในการตรวจสอบ

โปรแกรมย่อย	หน้าที่การทำงาน
START - CM5	ใช้ในการกำหนดค่าเริ่มต้นให้กับ 8253 และ 8273
CHECK2 - CRBF3	ใช้ตรวจสอบว่าสัญญาณขัดจังหวะที่รับได้มาจากส่วนใด
CHKINT	ใช้ในการแยกแยะส่วนย่อยของสถานะในการขัดจังหวะที่มาจาก 8273 ซึ่งจะให้ค่าที่แตกต่างกันออกมาขึ้นอยู่กับผลที่ตรวจสอบได้
NON1 - NON2	เป็นส่วนที่ใช้ในการเก็บค่าต่าง ๆ เมื่อเกิดการขัดจังหวะชนิด non-maskable ขึ้น

3.5.2 ซอฟต์แวร์ที่ใช้ในการแสดงผล (12) ซอฟต์แวร์ในส่วนของภาคแสดงผลนั้นจะอยู่บนเครื่องไมโครคอมพิวเตอร์ ในที่นี้จะใช้ภาษาเทอร์โบปาสกาล(13) (turbo pascal) เป็นตัวจัดการทั้งหมดโดยจะเริ่มตั้งแต่การนำข้อมูลจากบัฟเฟอร์เข้ามาเก็บในส่วนของ RAM ที่ใช้ในการแสดงผลแต่ก่อนที่จะแสดงผลจะต้องนำข้อมูลมาแปลงรหัสจาก EBCDIC เป็น ASCII (American Standard Code for Information Interchange) เสียก่อน ทั้งนี้เพราะรหัสที่ใช้อยู่ในเครื่องไมโครคอมพิวเตอร์เป็นรหัส ASCII ในส่วนของการแสดงผลจะแยกเป็นสองลักษณะคือ

1. การแสดงผลในรูปแบบของข้อมูลในชั้นที่1 (layer1) เป็นการนำข้อมูลที่รับมาได้ทั้งหมดไปแสดงผล วิธีนี้จะแสดงผลเป็น DTE และ DCE สลับกันไปทุก ๆ บรรทัด

2. การแสดงผลในรูปแบบของข้อมูลในชั้นที่2 (layer2) เป็นการนำข้อมูลที่รับมาได้ต่อ 1 เฟรมมาแสดงเฉพาะรายละเอียดของคำสั่งและแอดเดรสเท่านั้น

ในกรณีที่ต้องการเก็บข้อมูลที่รับมาได้ไว้เป็นหลักฐานหรือข้อมูลทางสถิติถ้าจะพิมพ์ผลในขณะนั้นจะไม่สามารถทำได้ ทั้งนี้เพราะว่าความเร็วในการพิมพ์ผลจะช้ากว่าความเร็วของข้อมูลที่กำลังตรวจสอบอยู่ซึ่งผลที่พิมพ์ได้นั้นจะไม่มีความต่อเนื่อง และจะทำให้วิเคราะห์เหตุการณ์ผิดได้ ปัญหานี้จะถูกแก้ไขโดยนำข้อมูลที่ตรวจสอบนี้เก็บลงบนจานแม่เหล็กก่อนแล้วจึงนำไปพิมพ์ผลในภาย

หลัง (รายละเอียดของโปรแกรมแสดงในภาคผนวก ค) ในที่นี้จะขอกล่าวเฉพาะส่วนสำคัญ ๆ เท่านั้น

โปรแกรมจะถูกเขียนเป็นส่วนย่อย ๆ ที่เรียกว่ากระบวนความ (procedure) โปรแกรมที่แสดงผลจะแสดงในสองลักษณะคือ layer1 และ layer2

ตารางที่ 3.3 ส่วนสำคัญของโปรแกรมที่ใช้ในการแสดงผล

กระบวนความหรือข้อกำหนด	หน้าที่
const ebd:ebc	แปลงรหัสจาก EBCDIC เป็น ASCII
savdat	บันทึกข้อมูลลงบนจานแม่เหล็ก
ondte	นำข้อมูลจากหน่วยความจำของซีพียู1 มาแสดงผล
ondce	นำข้อมูลจากหน่วยความจำของซีพียู2 มาแสดงผล
nsend,nrec	ใช้แยก N(S),N(R) ออกจาก control field
pofi	ใช้แยก poll/final ออกจาก control field
untyp1,untyp2	ใช้แยกชนิดของคำสั่งหรือคำตอบรับออกจาก control field
twoside1,twoside2	ใช้แสดงผลในแบบ layer2 และขณะเดียวกันก็ใช้ในการตรวจสอบการสิ้นสุดหรือสถานะต่าง ๆ ที่จะเกิดขึ้นได้เช่น idle,abort, เฟรมมีขนาดน้อยกว่า 32 บิต