บทที่ 5

การออกแบบและสร้างแผงวงจร

จุดประสงค์ของวิทยานิพนธ์ฉบับนี้คือสร้างแผงวงจรเพื่อให้ PC สามารถติดต่อสื่อสารกับข่าย ISDN ผ่านจุดเชื่อมโยง S/T ได้ โดยมีโครงสร้างการเชื่อมโยงแบบ Basic(2B+D)

การทำให้ PC สามารถเชื่อมโยงกับข่าย ISDN ที่จุดอ้างอิง S/T ได้นั้นสามารถทำได้ 2 วิธีคือ

ก. โดยใช้อุปกรณ์เทอร์มินัลอแดพเตอร์ (Terminal Adapter) แบบเอกเทศ (Standalone)

ข. โดยใช้แผงวงจรควบคุมการติดต่อสื่อสารภายใน PC

จุดประสงค์ของโครงการนี้ต้องการให้ PC สามารถควบคุมการติดต่อสื่อสารทั้งที่กระทำในช่องสัญญาณ D และนอกช่องสัญญาณ D ได้ จึงได้เลือกนำวิธีการที่สองมาใช้ทำให้ PC ติดต่อกับข่าย ISDN ได้

5.1 หลักการออกแบบ[11][12][13]

การออกแบบได้กำหนดให้แผงวงจรมีลักษณะเป็นการ์ดที่เสียบเข้ากับ Slot ภายใน PC โดยมีความสามารถครอบคลุมโปรโตคอลชั้นที่ 1 และ 2 โดยมีการทำงานดังนี้

ก. รับส่งข้อมูลกับข่าย ISDN ที่จุดอ้างอิง S/T โดยมีรูปแบบของข้อมูลตามที่ได้มีการกำหนดใน Rec. CCITT I.430

ข. สร้างและถอดเฟรม LAPD ได้

ค. สามารถสั่งงานแผงวงจรเพื่อเริ่มการทำงานจาก PC ได้

ง. ให้ PC สามารถสั่งงานแผงวงจรให้ทำการเชื่อมต่อและปลดวงจรเชื่อมต่อข้อมูลตามแบบโปรโตคอล LAPD ได้

จ. ให้แผงวงจรสามารถรับข้อมูลจากภายนอก โดยกรณีที่ได้รับข่าวสารให้แผงวงจรส่งข่าวสารนั้นมาแสดงผลที่จอ PC

ฉ. เมื่อ PC ต้องการส่งข่าวสารออก ให้สามารถป้อนข้อมูลผ่านทางคีย์บอร์ดของ PC ได้

การกำหนดให้แผงวงจรสามารถทำงานได้ในลักษณะนี้ จะทำให้สามารถเปลี่ยนแปลงการทำงานของแผงวงจรในอนาคตได้ โดยการพัฒนาซอฟต์แวร์ที่ทำหน้าที่โปรโตคอลชั้นที่ 3 เพื่อ

ห้องสมุดกลาง สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

ทำหน้าที่ป้อนข้อมูลให้ชั้นที่ 2 แทนการรับจากคีย์บอร์ดของ PC ซึ่งการพัฒนาแผงวงจรจะแยกเป็น 2 ส่วนคือ

ก. ส่วนฮาร์ดแวร์

ข. ส่วนซอฟต์แวร์

5.2 การออกแบบส่วนฮาร์ดแวร์

5.2.1 ฟังก์ชันของแผงวงจร

แผงวงจรจะมีหน้าที่ดังต่อไปนี้

ก. เชื่อมโยงกับจุดอ้างอิง S/T

ข. ฟังก์ชันสำหรับโปรโตคอลชั้นที่ 1

ค. ฟังก์ชันบางส่วนของโปรโตคอล LAPD เพื่อลดความยุ่งยากของซอฟต์แวร์

ง. ติดต่อสื่อสารกับ PC โดยผ่านทางบัสภายใน

ในการออกแบบแผงวงจรนั้น เพื่อให้การทำงานของแผงวงจรเป็นอิสระจาก PC สามารถตอบสนองการรับข้อมูลจากข่ายและทำฟังก์ชันที่กล่าวมาแล้วได้ด้วยความรวดเร็ว จึงได้กำหนดให้แผงวงจรมีการทำงานโดยอยู่ภายใต้การควบคุมของ CPU เฉพาะของตนเองอยู่บนแผงวงจร โดยไม่ต้องอาศัยความสามารถของ CPU ของ PC ที่เสียบแผงวงจร ดังนั้นจะสามารถเขียนส่วนประกอบต่างๆของแผงวงจรได้ดังรูปที่ 5.1

ก. วงจรไมโครโปรเซสเซอร์ เพื่อควบคุมการทำงานของแผงวงจรทั้งหมด วงจรส่วนนี้จะมี CPU 8085 เป็นอุปกรณ์หลัก

ข. วงจรเชื่อมโยงกับบัสที่จุดอ้างอิง S/T ตามแบบโปรโตคอลชั้นที่ 1 วงจรส่วนนี้ทำหน้าที่โดยไอซี SBC

ค. วงจรทำหน้าที่โปรโตคอลชั้นที่ 2 บางส่วน วงรส่วนนี้ทำหน้าที่โดยไอซี ICC

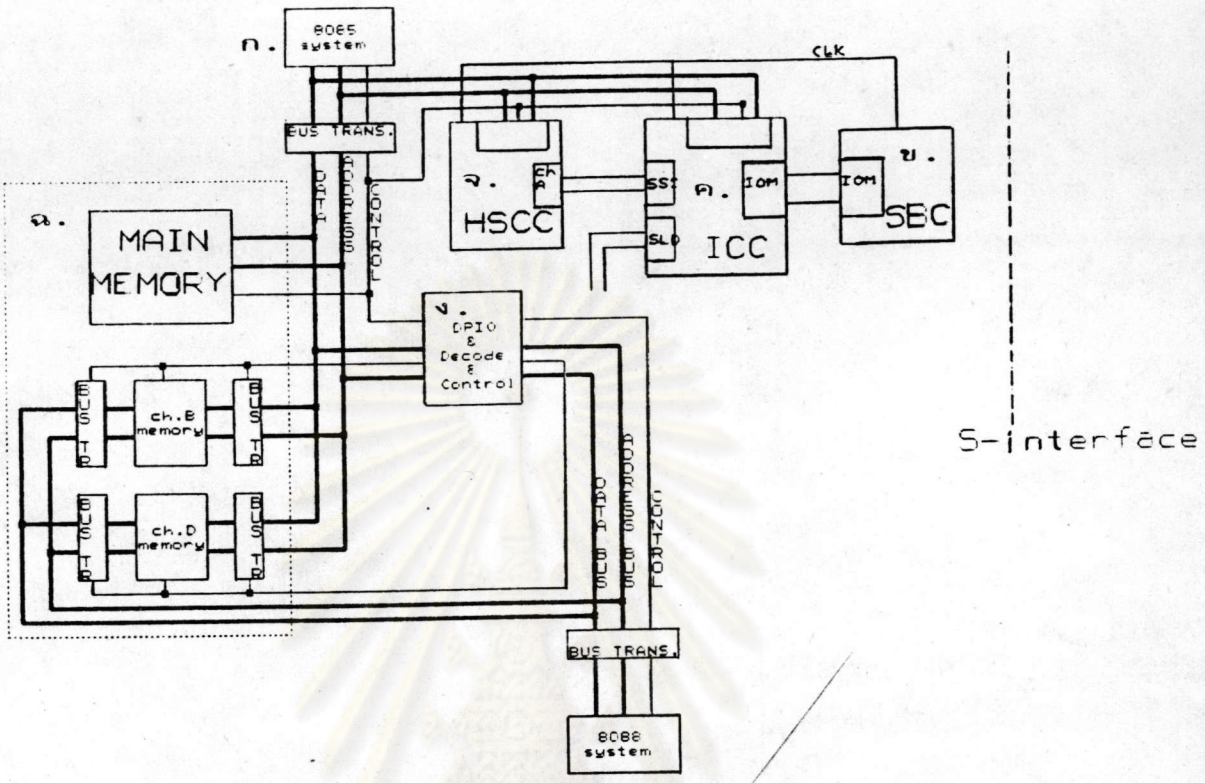
ง. วงจรควบคุมการติดต่อระหว่าง CPU ของแผงวงจรกับ CPU ของ PC เพื่อให้ CPU ทั้ง 2 ตัวสามารถติดต่อสื่อสารและทำงานร่วมกันได้

จ. วงจรแปลงข้อมูลแบบขนานเป็นแบบอนุกรม สำหรับการรับส่งข้อมูลจากหน่วยความจำสำรองของ PC ไปในช่องสัญญาณ B วงจรส่วนนี้ทำหน้าที่โดย HSCC

ฉ. วงจรหน่วยความจำ เพื่อเป็นที่เก็บข้อมูลชั่วคราวที่ได้รับหรือจะส่งออก โดยแยกเป็น 2 ส่วนสำหรับช่องสัญญาณ B และ D ช่องละ 1 ส่วน

5.2.2 วงจรไมโครโปรเซสเซอร์

วงจรนี้จะทำหน้าที่ควบคุมการทำงานภายในแผงวงจรทั้งหมด โดยมีอุปกรณ์หลักคือ CPU ซึ่งจะรับผิดชอบการเริ่มต้นการทำงานของแผงวงจร ขบวนการเชื่อมต่อและปลดวงจร



รูปที่ 5.1 ส่วนประกอบของแผงวงจร

เชื่อมต่อข้อมูล และการโต้ตอบภายในช่องสัญญาณ D ตามแบบโปรโตคอล LAPD โดยอาศัยโปรแกรมที่เก็บไว้ในหน่วยความจำ EPROM บนแผงวงจร

การทำงานของแต่ละขบวนการของ CPU บนแผงวงจร สามารถสั่งงานได้จาก PC โดยการติดต่อระหว่าง CPU บนแผงวงจรกับ PC จะทำโดยอาศัยสัญญาณขัดจังหวะและพอร์ทไอโอแบบ 2 ทาง(DPIO:Dual-Port I/O) ซึ่งสามารถอ่านและเขียนได้ทั้งจาก PC และ CPU บนแผงวงจร การทำงานในลักษณะนี้ทำให้ CPU บนแผงวงจรสามารถทำงานได้โดยอิสระและไม่จำเป็นต้องเลือกใช้ CPU ที่สามารถทำงานเข้ากันได้กับ CPU ของ PC และสำหรับแผงวงจรนี้ได้เลือกใช้ CPU เบอร์ 8085 เนื่องจากเป็น CPU ที่มีขาแอดเดรสและข้อมูลถูกมัลติเพลกซ์มาด้วยกันซึ่งจะทำให้สามารถต่อเข้ากับพอร์ทไมโครโปรเซสเซอร์ของ ICC และ HSCC ได้โดยตรง

5.2.3 วงจรควบคุมการติดต่อระหว่าง CPUs

จากรูปที่ 5.1 จะเห็นได้ว่าวงจรหน่วยความจำจะถูกเรียกใช้โดย CPU ทั้ง 2 ตัว ดังนั้นเพื่อหลีกเลี่ยงการชนกันของข้อมูลบนบัสอันเนื่องมาจาก CPU ทั้ง 2 ตัวต้องการใช้งาน

หน่วยความจำพร้อมกัน จึงได้ออกแบบให้วงจรส่วนนี้ประกอบด้วยวงจร Bus transceiver ทำงานร่วมกับ DPIO และวงจรสำหรับเลือกสัญญาณควบคุม(CE,RD และ WR)เพื่อใช้งานหน่วยความจำ โดยการทำงานจะเป็นดังนี้เมื่อ PC หรือ CPU บนแผงวงจรต้องการอ่านหรือเขียนหน่วยความจำ จะต้องตรวจสอบสถานะของหน่วยความจำจาก DPIO ก่อน ถ้าขณะนั้นหน่วยความจำว่างอยู่ CPU ตัวที่จะใช้งานจะต้องทำการเปลี่ยนสถานะของหน่วยความจำใน DPIO และปิดวงจร Bus transceiver ของ CPU อีกตัวหนึ่งพร้อมกับการเปิดวงจร Bus transceiverของตัวเองก่อน จึงจะสามารถเข้าใช้หน่วยความจำได้และเมื่อใช้งานเรียบร้อยแล้วจะต้องปิดวงจร Bus transceiver ของตัวเองและเปลี่ยนสถานะของหน่วยความจำใน DPIO ให้ว่างเพื่อให้หน่วยความจำสามารถถูกใช้งานได้ใหม่ ดังนั้นในขณะใดๆ จะมี Bus transceiver ของ CPU เพียงตัวเดียวเท่านั้นที่ทำงานอยู่

สัญญาณที่ใช้ในการเปิดและปิดวงจร Bus transceiver ทั้ง 2 ด้านของหน่วยความจำ จะได้จากสัญญาณ CS ที่ใช้สำหรับการอ้างแอดเดรสของหน่วยความจำชุดนั้น ๆ

5.2.4 <u>วงจรแปลงข้อมูลแบบขนานเป็นแบบอนุกรม</u>

วงจรส่วนนี้ทำหน้าที่แปลงข้อมูลแบบขนานที่รับมาจากหน่วยความจำสำหรับช่องสัญญาณ B ให้เป็นข้อมูลแบบอนุกรมเพื่อส่งต่อให้ ICC จัดการส่งออกไปในช่องสัญญาณ B และเนื่องจากการส่งข้อมูลจากหน่วยความจำสำรองของ PC ไปในช่องสัญญาณ B นั้นต้องการให้ข้อมูลที่รับส่งสามารถตรวจสอบความถูกต้องได้ และอาจจะมีการพัฒนาซอฟต์แวร์เพิ่มเติมเพื่อให้สามารถใช้ PC ในการติดต่อสื่อสารระหว่างผู้เช่าด้วยกันได้ ซึ่งในลักษณะนี้ 8085 จำเป็นต้องสามารถควบคุมการรับส่งข้อมูลได้ ซึ่งจะสามารถทำได้ถ้าข้อมูลที่รับส่งอยู่ในรูปเฟรมที่ประกอบด้วยข้อมูลสำหรับตรวจสอบความถูกต้องด้วย จึงได้เลือกใช้ไอซีเบอร์ SAB82520 (HSCC)(รายละเอียดเพิ่มเติมดูได้ในภาคผนวก ก.) ซึ่งสามารถทำฟังก์ชันเหล่านี้ได้[13]คือ สร้างและถอดเฟรม HDLC แทรกและดึงบิตศูนย์ สร้างและตรวจสอบข้อมูลส่วนตรวจสอบความผิดพลาดได้

การติดต่อระหว่าง HSCC กับ CPU บนแผงวงจรจะทำผ่านพอร์ทสำหรับไมโครโปรเซสเซอร์ โดยอาศัยรีจิสเตอร์ภายในและสัญญาณขัดจังหวะ การทำงานจะเป็นดังนี้ เมื่อมีเหตุการณ์สำคัญใด ๆ เกิดขึ้นภายใน HSCC จะสร้างสัญญาณขัดจังหวะส่งให้ CPU และ CPU จะทราบสาเหตุการเกิดสัญญาณขัดจังหวะได้โดยการอ่านข้อมูลในรีจิสเตอร์ของ HSCC และเมื่อ CPU ต้องการสั่งงาน HSCC ก็จะทำได้โดยเขียนข้อมูลลงในรีจิสเตอร์ของ HSCC เช่นเดียวกัน นอกจากนั้น CPU จะสามารถทราบสถานะของ HSCC ในขณะใด ๆ ได้โดยการอ่านข้อมูลสถานะจากรีจิสเตอร์ของ HSCC

ในการแปลงข้อมูลนั้น HSCC จะรับข้อมูลแบบขนานจาก 8085 มาคำนวณเพื่อหาข้อมูลส่วนตรวจสอบความผิดพลาด แล้วจึงนำไปจัดให้อยู่ในรูปเฟรมที่มีแฟล็กเป็นอักขระคั่น ก่อนจะส่งออกไปในรูปของข้อมูลแบบอนุกรมผ่านการพอร์ทสำหรับช่อง A ไปให้ ICC ต่อไป และเมื่อทำหน้าที่เป็นผู้รับ HSCC ก็จะนำเอาข้อมูลอนุกรมที่ได้มาตรวจสอบความผิดพลาดก่อนแล้วจึงแยกข้อมูลออกเป็นส่วน ๆ ตามรูปแบบของเฟรม HDLC และจัดส่งต่อให้ CPU ต่อไป ถ้าข้อมูลที่ได้รับนั้นถูกต้อง โดยอาศัยสัญญาณนาฬิกาจาก SBC

และภายใน HSCC มีรีจิสเตอร์ที่ใช้สำหรับการสั่งงานและเก็บข้อมูลชั่วคราว ซึ่งจะอ้างถึงได้จาก 8085 ดังนั้นเพื่อความรวดเร็วในการอ่านและเขียนรีจิสเตอร์เหล่านี้ จึงได้กำหนดให้รีจิสเตอร์เหล่านี้อยู่ในตำแหน่งที่เป็นหน่วยความจำ โดยสามารถอ้างได้ที่แอดเดรส E800H ถึง E85FH

5.2.5 <u>วงจรทำหน้าที่โปรโตคอลชั้นที่ 2</u>

การทำงานในระดับโปรโตคอลชั้นที่ 2 หรือ LAPD นั้นได้เลือกใช้ไอซีเบอร์ PEB2080 (ICC)(รายละเอียดดูได้จากภาคผนวก ข.) สำหรับทำหน้าที่ร่วมกับซอฟต์แวร์บางส่วนที่จะได้พัฒนาขึ้นมา เหตุผลในการเลือก ICC มาใช้งาน[13]คือ ICC ได้รับการพัฒนาโดยมีฟังก์ชันหลักคือ จัดการการติดต่อสื่อสารภายในช่องสัญญาณ D ตามแบบโปรโตคอล LAPD โดยอาศัยข้อมูลที่ได้รับจากรีจิสเตอร์ภายในและจากจุดอ้างอิง S โดยผ่านวงจรเชื่อมโยง IOM (รายละเอียดเพิ่มเติมดูได้ในภาคผนวก ข.) นอกจากนั้น ICC ยังสามารถทำฟังก์ชันต่อไปนี้ได้ด้วยคือการแทรกบิตศูนย์ การสร้างและถอดเฟรม LAPD การสร้างและตรวจสอบข้อมูลที่ส่งในส่วนตรวจสอบความผิดพลาดของเฟรมซึ่งทำให้สามารถลดความยุ่งยากในการพัฒนาซอฟต์แวร์สำหรับโปรโตคอล LAPD ได้

การติดต่อระหว่าง ICC กับ CPU บนแผงวงจรจะทำผ่านพอร์ทสำหรับไมโครโปรเซสเซอร์ โดยอาศัยรีจิสเตอร์ภายในและสัญญาณขัดจังหวะ การทำงานจะเป็นดังนี้ เมื่อมีเหตุการณ์สำคัญใด ๆ เกิดขึ้นภายใน ICC จะสร้างสัญญาณขัดจังหวะส่งให้ CPU และ CPU จะทราบสาเหตุการเกิดสัญญาณขัดจังหวะได้โดยการอ่านข้อมูลในรีจิสเตอร์ของ ICC และเมื่อ CPU ต้องการสั่งงาน ICC ก็จะทำได้โดยเขียนข้อมูลลงในรีจิสเตอร์ของ ICC เช่นเดียวกัน นอกจากนั้น CPU จะสามารถทราบสถานะของ ICC ในขณะใด ๆ ได้โดยการอ่านข้อมูลสถานะจากรีจิสเตอร์ของ ICC

ในการทำงาน ICC จะนำข้อมูลที่ได้รับจาก 8085 และข้อมูลที่ ICC สร้างขึ้นเองในรีจิสเตอร์ต่าง ๆ มาจัดให้อยู่ในรูปของเฟรม LAPD โดยทำการคำนวณข้อมูลส่วนตรวจสอบความผิดพลาดและการแทรกบิตศูนย์ให้กับข้อมูลที่จะส่งออกด้วย จากนั้นจะส่งต่อไปให้ SBC พร้อม

กับข้อมูลที่ได้รับจาก HSCC จากพอร์ท SSI ซึ่งเป็นพอร์ตสำหรับข้อมูลทั่วไปที่อัตราเร็วบิต 128 Kbps โดยในการส่งนั้น ICC จะนำข้อมูลทั้ง 2 ส่วนนี้มาทำการมัลติเพลกซ์เชิงเวลาให้อยู่ในรูปแบบ IOM(IOM-format) ก่อนแล้วจึงส่งออกไปที่พอร์ท IOM ด้วยอัตราเร็วบิต 256 Kbps โดยอาศัยสัญญาณนาฬิกาจาก SBC

ภายใน ICC จะประกอบด้วยรีจิสเตอร์ที่ใช้สำหรับการสั่งงานและเก็บข้อมูลชั่วคราว ซึ่งจะอ้างถึงได้จาก 8085 ดังนั้นเพื่อความรวดเร็วในการอ่านและเขียนรีจิสเตอร์เหล่านี้ จึงได้กำหนดให้รีจิสเตอร์เหล่านี้อยู่ในตำแหน่งที่เป็นหน่วยความจำ โดยสามารถอ้างได้ที่แอดเดรส EC00H ถึง EC38H

5.2.6 วงจรเชื่อมโยงกับจุดเชื่อมโยง S

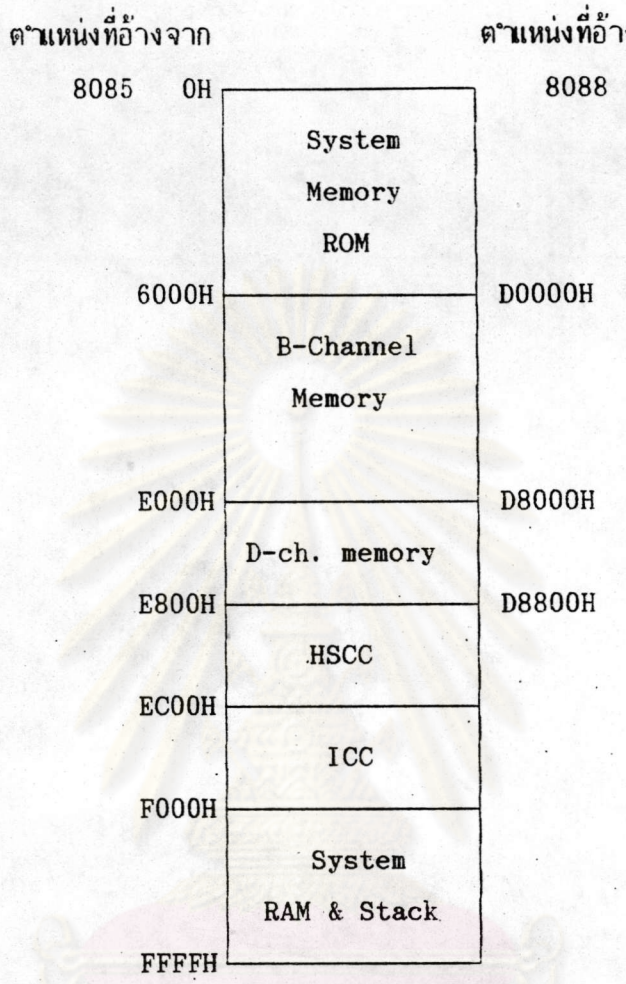
วงจรส่วนนี้ทำหน้าที่เชื่อมโยงแผงวงจรเข้ากับบัสที่จุดอ้างอิง S ซึ่งเป็นฟังก์ชันในระดับโปรโตคอลชั้นที่ 1 และเป็นงานทางด้านฮาร์ดแวร์ทั้งหมด จึงได้เลือกใช้ไอซีประเภท VLSI ที่มีความสามารถในการทำฟังก์ชันของโปรโตคอลชั้นที่ 1 มาใช้งานคือไอซีเบอร์ PEB2070 (SBC)(รายละเอียดดูเพิ่มเติมได้ที่ภาคผนวก ค.)และ SBC นี้จะสามารถสั่งงานได้จาก CPU บนแผงวงจรเพื่อทำขบวนการบางชนิดที่ต้องมีการสั่งงานเช่น ขบวนการ Activate/Deactivate โดยผ่านวงจรเชื่อมโยง IOM

ในการทำงาน SBC จะนำสัญญาณข้อมูลที่ได้รับจาก ICC ทางพอร์ท IOM มาทำการระมวลผลและนำเอาข้อมูลจริงที่ต้องการส่งออก ทำการมัลติเพล็กซ์กับสัญญาณควบคุมอื่น ๆ ที่จำเป็นสำหรับการส่งในระดับโปรโตคอลชั้นที่ 1 จากนั้นนำจึงไปแปลงรหัสให้เป็นแบบ 100% pulsewidth-pseudo-ternary แล้วจึงส่งออกไปที่จุดอ้างอิง S ต่อไปด้วยอัตราเร็วบิต 192 Kbps นอกจากนั้น SBC ยังทำหน้าที่จ่ายสัญญาณนาฬิกาให้กับ ICC และ HSCC เพื่อการทำงานและรับส่งข้อมูลด้วย

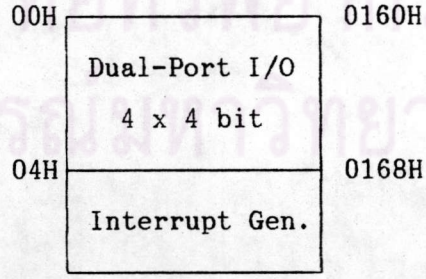
5.2.7 วงจรหน่วยความจำ

วงจรหน่วยความจำแบ่งได้เป็น 2 ส่วนคือหน่วยความจำสำหรับ 8085 และหน่วยความจำร่วมที่ใช้งานโดย 8085 และ CPU ของ PC(ในที่นี้คือ 8088)

5.2.6.1 หน่วยความจำสำหรับ 8085 หน่วยความจำส่วนนี้ประกอบด้วย EPROM ขนาด 24 KB และมีแอดเดรสอ้างถึงได้ที่ตำแหน่ง 0000H ถึง 5FFFH ใช้สำหรับเก็บโปรแกรมที่ 8085 ใช้ในการควบคุมการทำงานของแผงวงจรทั้งหมด และ RAM หลักขนาด 4 KB ซึ่งจะถูกใช้งานโดยโปรแกรมใน EPROM และ CPU ในการเก็บข้อมูลชั่วคราวและใช้เป็น Stack ในระหว่างที่มีการดำเนินโปรแกรม(Run) สามารถอ้างได้ที่ตำแหน่ง F000H ถึง FFFFH



ก. การแทนข้อมูลสำหรับหน่วยความจำ



ข. การแทนข้อมูลสำหรับพอร์ทไอโอ

รูปที่ 5.2 การแทนข้อมูลบนแผงวงจร

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

5.2.6.2 หน่วยความจำร่วม หน่วยความจำส่วนนี้ CPU ทั้ง 2 ตัวสามารถเขียนและอ่านได้ แยกเป็น 2 ชุดคือสำหรับเก็บข้อมูลชั่วคราวสำหรับจ้อมูลที่จะส่งไปในช่องสัญญาณ B และ D ช่องสัญญาณละ 1 ชุดขนาด 32 และ 2 KB ตามลำดับ การเข้าใช้หน่วยความจำในส่วนนี้ CPU ทั้ง 2 ตัวสามารถทำได้โดยผ่านวงจร Bus transceiver และวงจรเลือกสัญญาณควบคุม ซึ่งเป็นวงจรที่ทำหน้าที่ป้องกันการชนกันของข้อมูลบนบัสของ CPU ทั้ง 2 ตัวโดยในขณะใดขณะหนึ่งจะมี CPU เพียงตัวเดียวเท่านั้นที่สามารถเข้าใช้หน่วยความจำได้

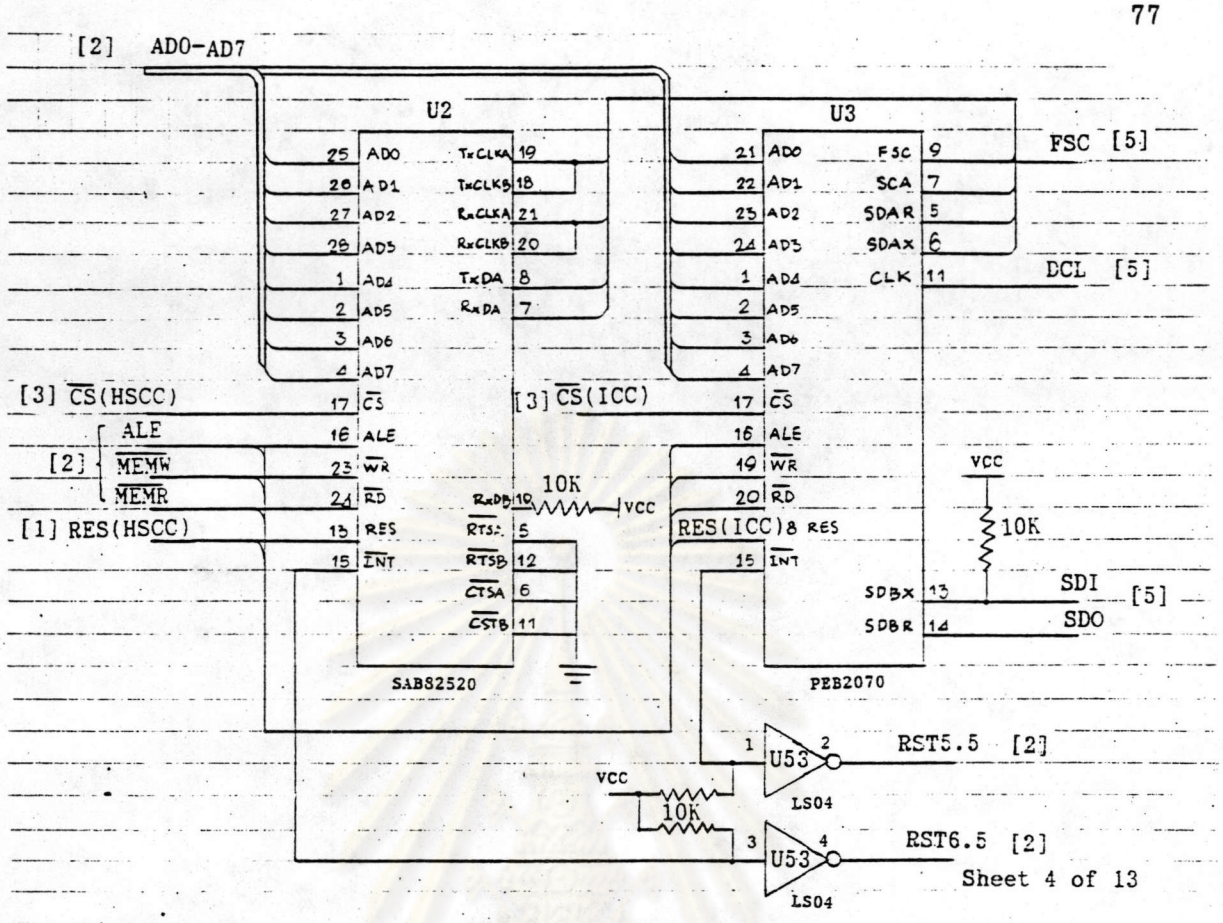
การอ้างแอดเดรสของหน่วยความจำส่วนนี้ 8085 อ้างได้ที่ตำแหน่ง 6000H ถึง DFFFH และ E000H ถึง E7FFH ในขณะที่ 8088 อ้างได้ที่ตำแหน่ง D0000H ถึง D7FFFH และ D8000H ถึง D87FFH สำหรับช่องสัญญาณ B และ D ตามลำดับ การแทนข้อมูลให้หน่วยความจำส่วนนี้เมื่อมองจาก 8088 ที่ตำแหน่งแอดเดรสนี้เนื่องจาก เป็นตำแหน่งที่ไม่มีการใช้งานหรืออ้างถึงในการใช้งานประมวลผลปกติ ทำให้สามารถนำเอาแผงวงจรนี้ไปใช้งานกับ PC ที่มีการใช้งานหน่วยความจำหลักครบทั้ง 640 KB ได้ นอกจากนั้นการถ่ายเทข้อมูลจากหน่วยความจำสำรองเช่นจานแม่เหล็ก เพื่อส่งออกไปในช่องสัญญาณทั้ง 2 ชนิดจะทำได้ด้วยความรวดเร็วเนื่องจากเป็นการถ่ายเทข้อมูลเข้าไว้ในหน่วยความจำของตัว 8088 เอง

การจัดตำแหน่งของหน่วยความจำและพอร์ทไอโอ(พอร์ท)ทั้งหมด ทั้งตำแหน่งที่อ้างโดย 8085 และ 8088 สามารถแสดงได้ในรูปที่ 5.2ก และ 5.2ข ตามลำดับ

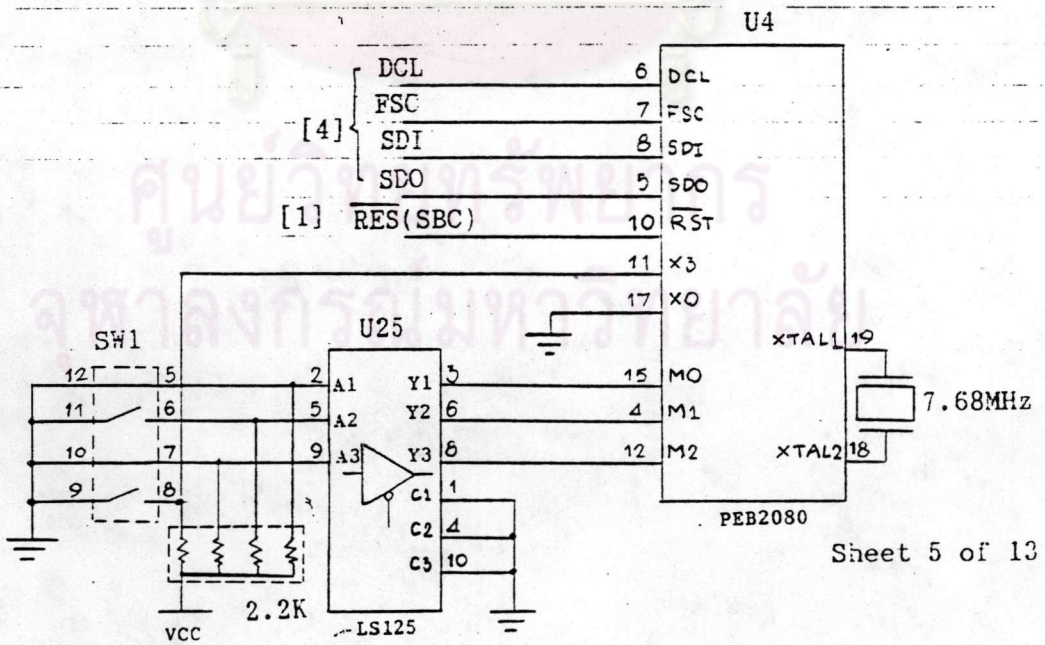
5.3 รายละเอียดส่วนฮาร์ดแวร์ [15][16]

สำหรับการอ้างถึงหน่วยความจำบนแผงวงจรโดย 8088 ซึ่งได้กำหนดให้อยู่ที่ตำแหน่งแอดเดรส D0000H เป็นต้นไป เนื่องจากเป็นพื้นที่ส่วนที่ยังไม่มีการใช้งาน และสามารถอ้างถึงได้โดยตรงจาก 8088

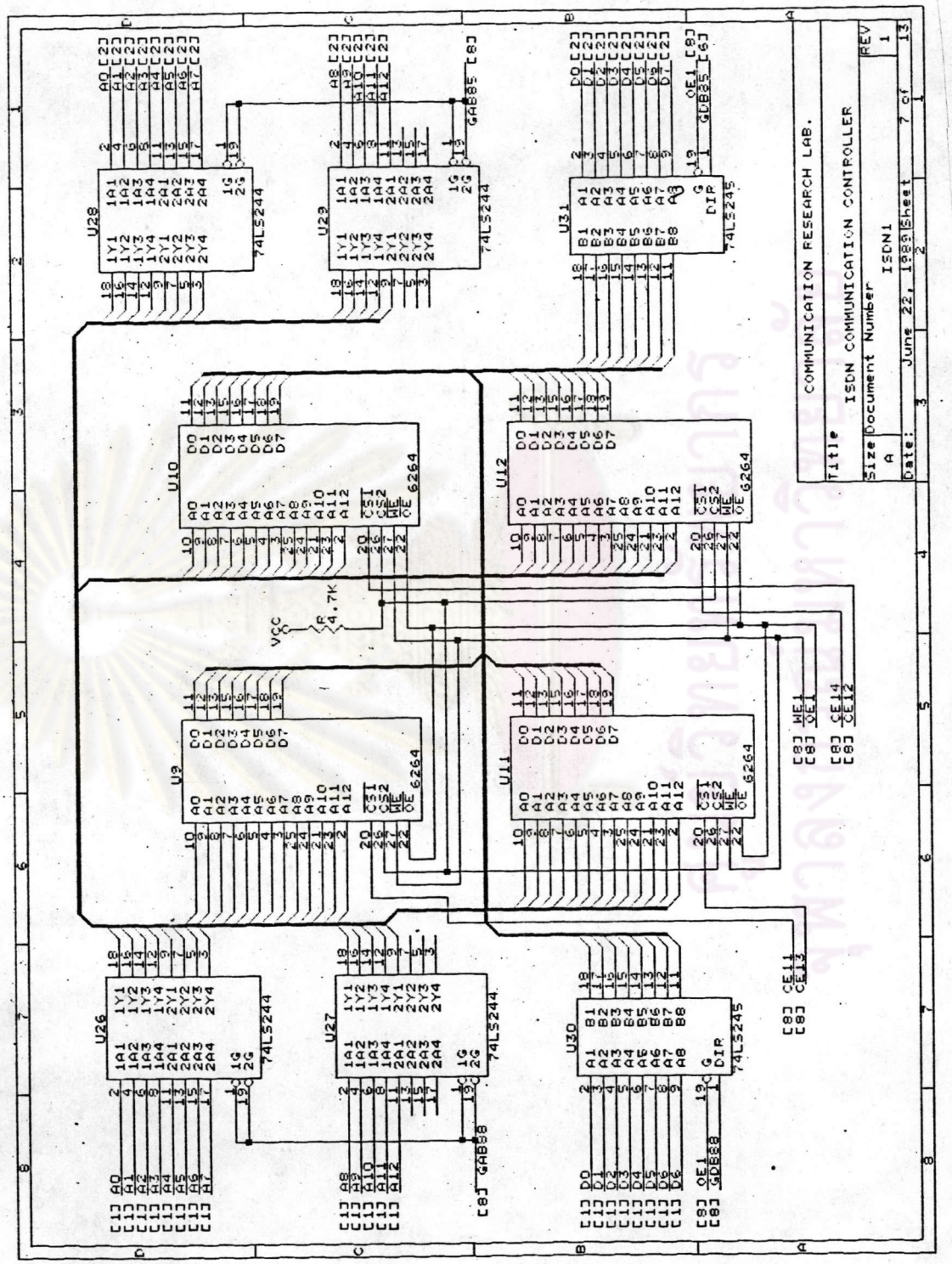
รูปที่ 5.3 เป็นวงจรทำหน้าที่แปลงข้อมูลแบบขนานเป็นแบบอนุกรมโดยใช้ไอซี HSCC (U2) และวงจรทำหน้าที่โปรโตคอลชั้นที่ 2 โดยใช้ไอซี ICC(U3) ไอซีทั้ง 2 ตัวนี้จะต่อโดยตรงจากบัสแบบมัลติเพลกซ์ของ 8085 และไอซีทั้ง 2 ตัวนี้สามารถสั่งงานหรือควบคุมได้โดย 8085 โดยการอ่านหรือเขียนรีจิสเตอร์ภายในที่ได้กำหนดให้มีการแทนข้อมูลแบบหน่วยความจำ การออกแบบกำหนดให้ใช้งาน HSCC เพียง 1 ช่องคือช่อง A โดยจะส่งข้อมูลให้ ICC ทางพอร์ท SSI ของ ICC และให้ทำงานด้วยสัญญาณนาฬิกาโมด 1 ซึ่งจะได้รับสัญญาณนาฬิกาสำหรับการรับและส่งข้อมูลจาก ICC ผ่านทางขา RxCLK และในกรณีที่เกิดเหตุการณ์ที่ทำให้มีการสร้างสัญญาณขัดจังหวะ HSCC จะส่งให้ 8085 ที่ขา RST6.5 และ ICC ส่งที่ขา RST5.5



รูปที่ 5.3 วงจรแปลงข้อมูลแบบขนานเป็นแบบอนุกรมโดย HSCC
และวงจรทำหน้าที่โปรโตคอลชั้นที่ 2 โดย ICC



รูปที่ 5.4 วงจรเชื่อมโยงกับจุดเชื่อมโยง S โดย SBC



รูปที่ 5.5 วงจรหน่วยความจำร่วมสำหรับช่องสัญญาณ B

รูปที่ 5.4 เป็นวงจรทำหน้าที่เชื่อมโยงกับจุดเชื่อมโยง S โดยใช้ไอซี SBC(U4) การกำหนดโมดการทำงานของ SBC อาศัย SW1 และได้กำหนดให้ SBC ทำงานในโมด TE Normal Mode ซึ่งทำให้ SBC จะเป็นตัวที่ส่งสัญญาณนาฬิกาให้กับไอซีหลักบนแผงวงจรโดยอาศัย Crystal ขนาด 7.68 MHz

รูปที่ 5.5 เป็นวงจรหน่วยความจำร่วมสำหรับเก็บข้อมูลที่รับส่งในช่องสัญญาณ B และเนื่องจากหน่วยความจำในส่วนนี้สามารถเข้าถึงได้ทั้งจาก 8085 และ 8088 ดังนั้นจึงต้องอาศัยไอซีประเภท Bus transceiver คือเบอร์ 74LS244(U26,U27,U28 และ U29) และเบอร์ 74LS245(U30 และ U31) มาควบคุมการอ่านและเขียนเพื่อให้เป็นไปตามลำดับ การใช้งานหน่วยความจำส่วนนี้ โดยปกติจะกำหนดให้อยู่ภายใต้การควบคุมของ 8085 เมื่อ 8088 ต้องการอ่านหรือเขียน จะต้องตรวจสอบสถานะของหน่วยความจำนี้ก่อน โดยการอ่านข้อมูลใน DPIO ถ้าพบว่าขณะนั้นหน่วยความจำว่าง 8088 จึงจะสามารถอ่านหรือเขียนได้โดยต้องเปลี่ยนสถานะของหน่วยความจำใน DPIO ก่อนที่จะทำการบิด U28,U29 และ U31 ซึ่งทำหน้าที่เบิดบัสจาก 8085 ขณะเดียวกับที่เบิด U26,U27 และ U30 เพื่อเบิดบัสของหน่วยความจำให้กับ 8088 และและเมื่ออ่านหรือเขียนเสร็จก็จะต้องเปลี่ยนสถานะใน DPIO ให้กลับสู่สถานะเดิมด้วย และถ้า 8085 ต้องการอ่านหรือเขียน RAM ส่วนนี้ก็จะต้องทำงานตามขั้นตอนเดียวกันนี้ด้วย

รูปที่ 5.6 เป็นวงจรที่ใช้ควบคุมการอ่านและเขียนหน่วยความจำร่วมสำหรับข้อมูลที่จะใช้ช่องสัญญาณ B โดยใช้ไอซีเบอร์ 74LS157(U32 และ U33) ในการกำหนดสัญญาณควบคุม(RD WR และ CE) ให้กับหน่วยความจำส่วนนี้ โดยสัญญาณที่ทำหน้าที่ควบคุม U32 และ U33 จะได้มาจากไอซีเบอร์ 7474(U34) โดยมีการทำงานดังนี้ วงจรส่วนนี้โดยปกติจะเบิดสัญญาณควบคุมจาก 8085 ให้กับหน่วยความจำ เมื่อใดก็ตามที่ 8088 ต้องการอ่านหรือเขียนหน่วยความจำ 8088 จะต้องใช้คำสั่ง OUT ที่พอร์ท 0164H เพื่อบิดบัฟเฟอร์ที่ต่อกับบัสของ 8085 ในขณะเดียวกันจะต้องเบิดบัฟเฟอร์ที่ต่อกับบัสของ 8088 ดังนั้นขณะนี้ 8088 จะสามารถอ่านหรือเขียนหน่วยความจำได้จนกว่า 8088 จะส่งคำสั่ง OUT ที่พอร์ท 0165H ซึ่งจะเป็นการบิดบัฟเฟอร์ของตนเองจากบัสของหน่วยความจำทำให้ 8085 เป็นผู้ที่ได้เข้าใช้หน่วยความจำอีกครั้งหนึ่ง

รูปที่ 5.7 เป็นวงจรหน่วยความจำร่วมขนาด 2 KB สำหรับเก็บข้อมูลที่รับส่งในช่องสัญญาณ D ซึ่งสามารถอ่านและเขียนได้จาก CPU ทั้ง 2 ตัว ดังนั้นจึงต้องอาศัยไอซีประเภท Bus transceiver คือเบอร์ 74LS244(U37,U38 และ U39) และเบอร์ 74LS245(U41 และ U42) มาควบคุมเพื่อให้การอ่านและเขียนหน่วยความจำเป็นไปตามลำดับ การทำงานของวงจรส่วนนี้จะใช้วิธีการเดียวกันกับวงจรในรูปที่ 5.5

รูปที่ 5.6 แสดงวงจรควบคุมการติดต่อกับ ISDN

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

รูปที่ 5.7 วงจรส่วนติดต่อกับไมโครโปรเซสเซอร์ 8085

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

รูปที่ 5.8 วงจรในรูปนี้แบ่งออกเป็น 2 ส่วน ส่วนบนของรูปเป็นวงจรที่ใช้ควบคุมการอ่านและเขียนหน่วยความจำร่วมสำหรับช่องสัญญาณ D โดยใช้ไอซีเบอร์ 74LS157 (U43) และเบอร์ 7474 (U34) การทำงานจะใช้วิธีเดียวกันกับวงจรในรูปที่ 5.6 ยกเว้นพอร์ทที่ 8088 ใช้ในการเปิดบัฟเฟอร์ U34 จะใช้พอร์ท 0166H และพอร์ทที่ใช้ปิดบัฟเฟอร์จะใช้พอร์ท 0167H แทน ส่วนล่างของรูป วงจรประกอบด้วยไอซีเบอร์ 7474(U44) ซึ่งจะทำหน้าที่เป็นวงจรสร้างสัญญาณขัดจังหวะเพื่อส่งให้ 8088 เมื่อมีการเปลี่ยนแปลงใด ๆ เกิดขึ้นบนแผงวงจรและต้องการแจ้งให้ 8088 ทราบเพื่อทำงานขั้นต่อไปโดย 8088 จะทราบสาเหตุการเกิดสัญญาณขัดจังหวะนี้ได้โดยการอ่านสถานะใน DPIO การทำงานจะเป็นดังนี้ เมื่อ 8085 ต้องการสร้างสัญญาณขัดจังหวะก็จะใช้คำสั่ง OUT ที่พอร์ท 04H และเมื่อ 8088 รับทราบก็จะใช้คำสั่ง OUT ที่พอร์ท 0169H เพื่อ Clear สัญญาณขัดจังหวะที่ได้รับ

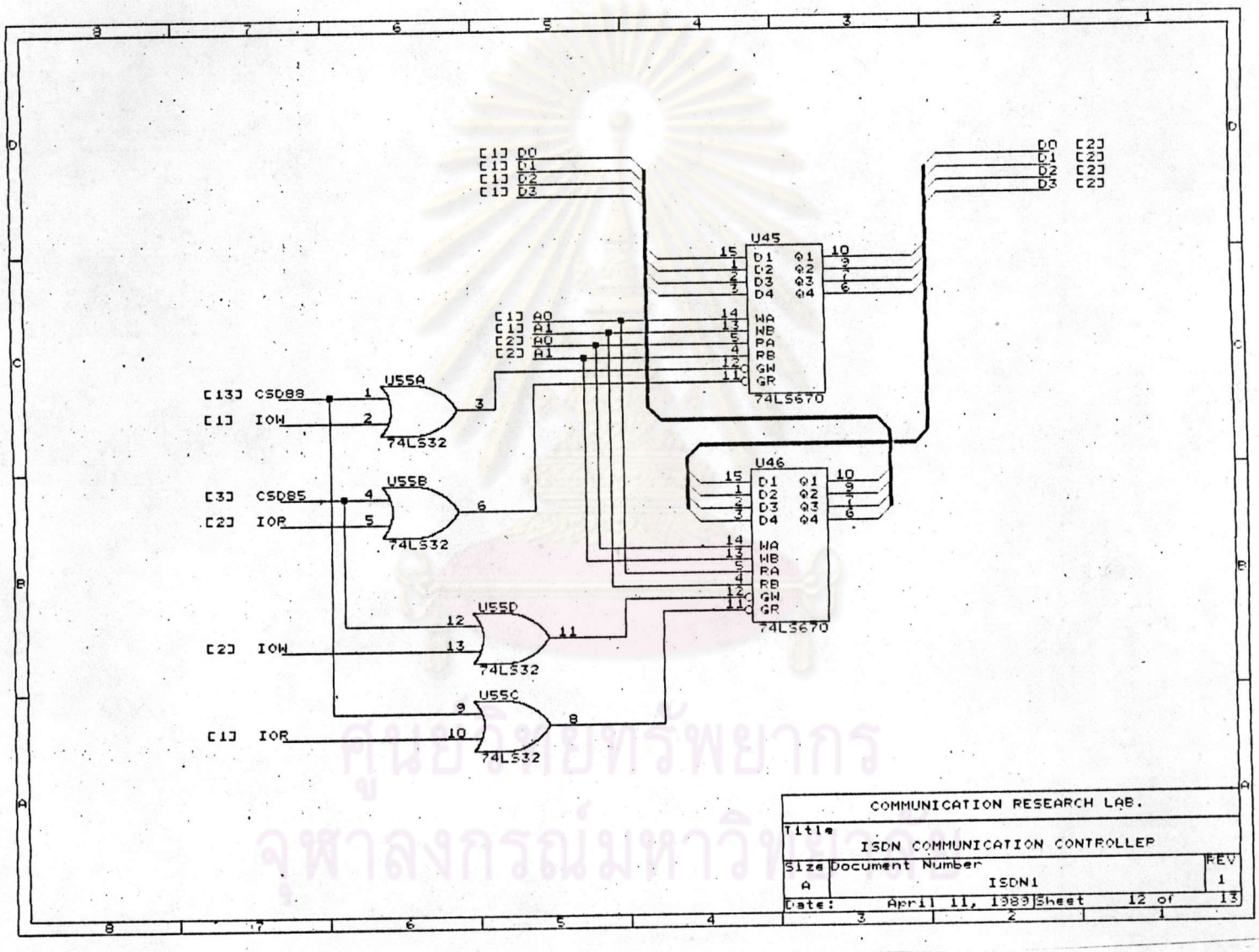
รูปที่ 5.9 เป็นวงจรทำหน้าที่เป็น DPIO ขนาด 4x4 บิต ซึ่งสามารถอ่านและเขียนได้พร้อมกัน สำหรับเก็บสถานะของแผงวงจรและหน่วยความจำร่วมโดยใช้ไอซีเบอร์ 74LS670 (U45 และ U46) ร่วมกับเบอร์ 74LS32(U55) พอร์ทส่วนนี้สามารถอ้างได้ที่ตำแหน่งแอดเดรส 00H ถึง 03H จาก 8085 และที่แอดเดรส 0160H ถึง 0163H จาก 8088 การทำงานจะให้ CPU แต่ละตัวสามารถอ่านและเขียนได้ที่แอดเดรสเดียวกันแต่ตำแหน่งทางกายภาพ(Physical Location) ต่างกันโดยอาศัยสัญญาณควบคุม IOR และ IOW ร่วมกับสัญญาณ CS ที่ใช้อ้างถึงแอดเดรสชุดนี้

รูปที่ 5.10 เป็นวงจรทำหน้าที่ถอดรหัสเพื่อใช้อ้างถึงพอร์ทบนแผงวงจรจาก 8088 และเพื่อสร้างสัญญาณขัดจังหวะสำหรับส่งให้ 8085 รวมทั้งวงจรสร้างสัญญาณ RESET ให้กับไอซีหลักบนแผงวงจรคือ HSCC,ICC และ SBC สำหรับการถอดรหัสเพื่ออ้างถึงพอร์ทบนแผงวงจรจะแยกเป็น 2 ขั้นตอน ขั้นแรกเป็นการถอดรหัสเพื่อให้ได้สัญญาณ CS ที่สามารถอ้างพอร์ทได้ชุดละ 4 พอร์ทโดยใช้ไอซี 74LS138(U49) ได้สัญญาณ CS ที่ขา 15(Y0) จะนำไปใช้สำหรับการอ้างถึง DPIO และขั้นที่ 2 ใช้ไอซีเบอร์ 74LS139(U50)ในการถอดรหัสเพื่อใช้เป็นพอร์ทสำหรับการอ่านและเขียนหน่วยความจำร่วมทั้ง 2 ส่วน โดยอาศัยสัญญาณ CS จากขา 14 และ 13 (Y1 และ Y2)ของ U49 เป็นสัญญาณเปิดเกต สัญญาณออกที่ได้จะนำไปใช้งานดังนี้พอร์ท 0164H ถึง 0167H ใช้สำหรับการเปิดปิด Buffer ที่ต่ออยู่กับหน่วยความจำร่วม พอร์ทที่ 0168H ใช้เป็นสัญญาณขัดจังหวะส่งให้ 8085 พอร์ทที่ 0169H ใช้สำหรับการ Clear สัญญาณขัดจังหวะที่ 8085 ส่งมาและพอร์ทที่ 016AH จะนำไปต่อที่ขา CLR ของ U44 เพื่อสร้างสัญญาณสำหรับการ RESET ไอซีหลักบนแผงวงจร

การใช้งานพอร์ทจาก 8088 สามารถสรุปได้เป็นตารางดังแสดงในตารางที่ 5.1

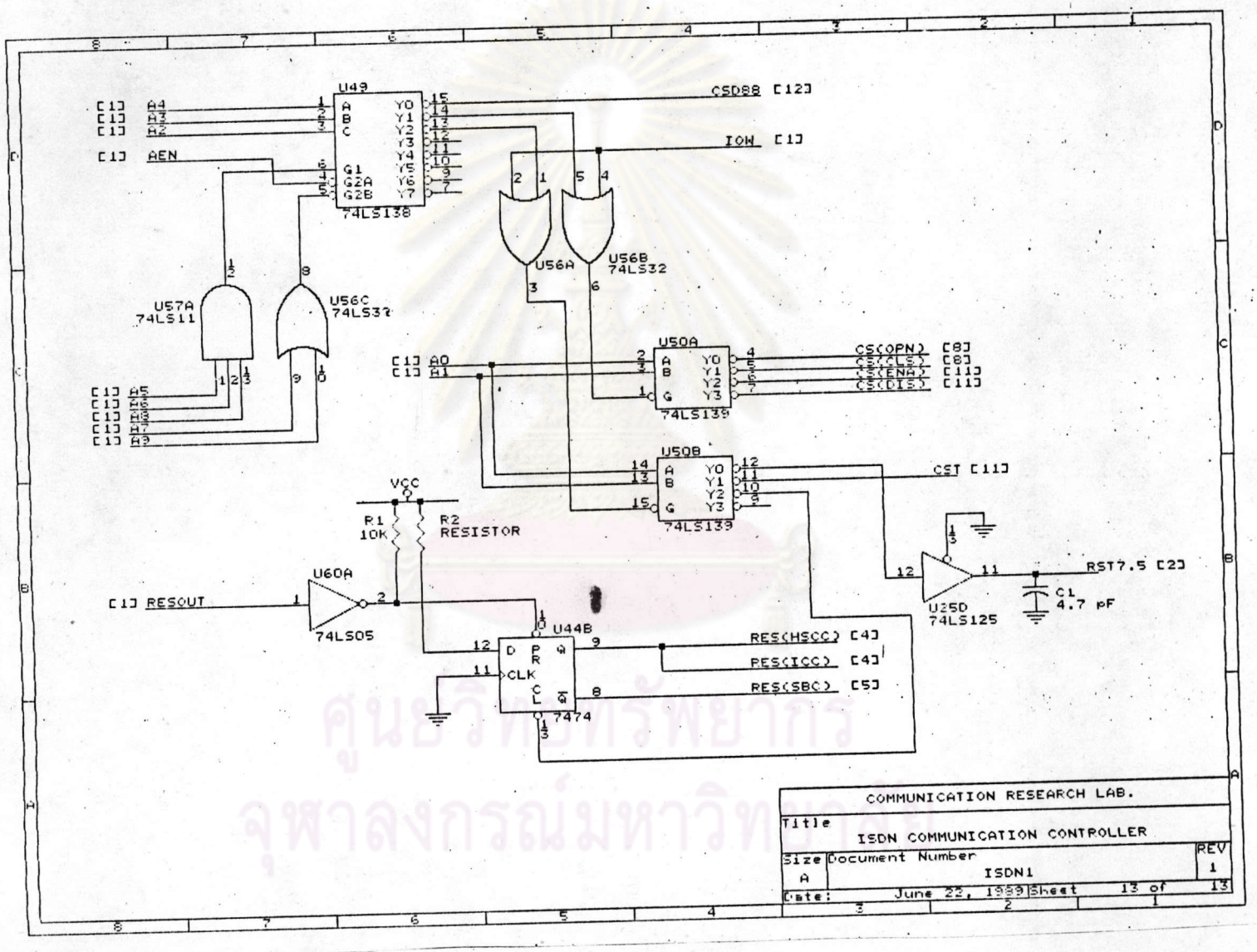
รูปที่ 5.8 แสดงวงจรส่วนควบคุมการทำงานของ 8085 และ D ที่ใช้ในการติดต่อกับ ISDN

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 5.9 วงจร DPIO ขนาด 4x4 บิต

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 5.10 วงจรของตัวสื่อสารแบบอนุกรมใช้ไอซี 8088

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ตารางที่ 5.1 การใช้งานพอร์ทจาก 8088

พอร์ท	หน้าที่
0160H-0163H	- อ่านหรือเขียน DPIO
0164H	- เปิด Buffer หน่วยความจำร่วม B
0165H	- ปิด Buffer หน่วยความจำร่วม B
0166H	- เปิด Buffer หน่วยความจำร่วม D
0167H	- ปิด Buffer หน่วยความจำร่วม D
0168H	- สร้างสัญญาณขัดจังหวะ
0169H	- Clear สัญญาณขัดจังหวะจาก 8085
016AH	- สร้างสัญญาณ RESET

5.4 การออกแบบส่วนซอฟต์แวร์ [17][18][19][20]

ซอฟต์แวร์ควบคุมการทำงานของแผงวงจรจะแยกเป็น 2 ส่วน คือ

ก. ซอฟต์แวร์ทำงานบน PC โดย 8088 เป็นโปรแกรมที่เขียนโดยภาษา C และภาษาแอสเซมบลีของ 8088 (8088-Assembly) ทำหน้าที่ติดต่อกับผู้ใช้ผ่านทางคีย์บอร์ด และจอแสดงผลและติดต่อกับแผงวงจรโดยอาศัยสัญญาณขัดจังหวะ

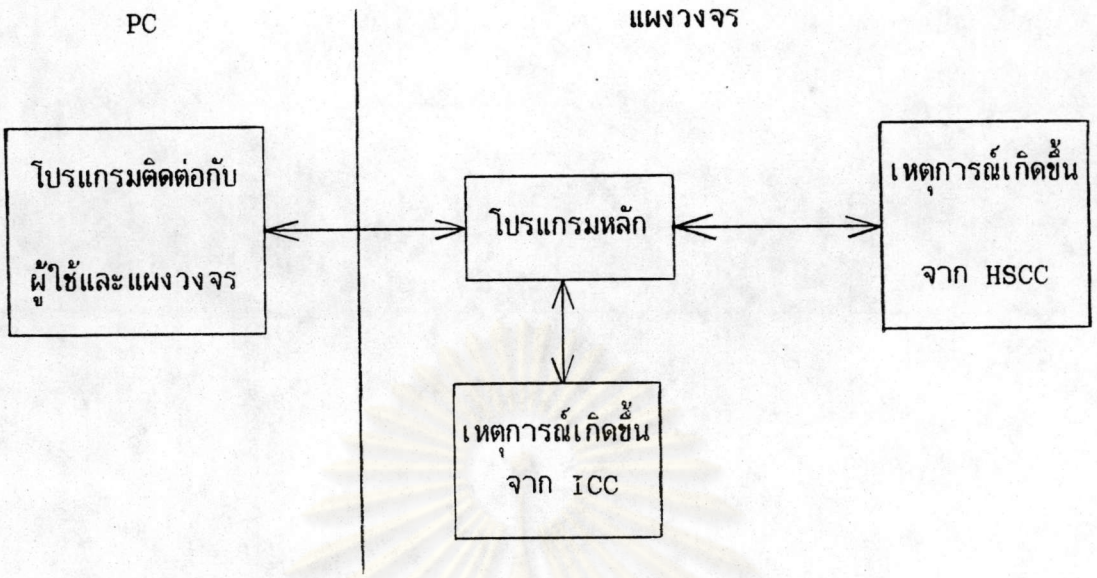
ข. ซอฟต์แวร์ทำงานบนแผงวงจรโดย 8085 เป็นโปรแกรมที่เขียนโดยภาษาแอสเซมบลีของ 8085 (8085-Assembly) ทำหน้าที่ควบคุมการทำงานทั้งหมดของแผงวงจรและติดต่อกับโปรแกรมบน PC โดยอาศัยสัญญาณขัดจังหวะ

ซอฟต์แวร์ทั้ง 2 ส่วนนี้ได้กำหนดให้มีการทำงานในลักษณะ Interrupt Driven คือ โปรแกรมแต่ละส่วนจะถูกเรียกใช้โดยโปรแกรมหลักเมื่อมีสัญญาณขัดจังหวะเกิดขึ้น ซึ่งจะสามารถเขียนแผนผังการทำงานของซอฟต์แวร์ของทั้งระบบได้ดังแสดงในรูปที่ 5.11 ในการออกแบบฮาร์ดแวร์ของแผงวงจรได้กำหนดให้ 8085 รับสัญญาณขัดจังหวะทั้งหมด 3 เส้นคือ RST7.5, RST6.5 และ RST5.5 จาก PC,HSCC และ ICC ตามลำดับ

5.5 รายละเอียดส่วนซอฟต์แวร์

5.5.1 ซอฟต์แวร์บน PC

ซอฟต์แวร์บน PC ได้กำหนดให้สามารถติดต่อกับผู้ใช้ผ่านทางคีย์บอร์ด จอแสดงผล โดยอาศัยสัญญาณขัดจังหวะและติดต่อกับแผงวงจรโดยสัญญาณขัดจังหวะเช่นเดียวกัน การทำ



รูปที่ 5.11 ความสัมพันธ์ของซอฟต์แวร์

งานของซอฟต์แวร์ให้เป็นแบบตรวจสอบเงื่อนไขเมื่อเกิดสัญญาณขัดจังหวะ ซึ่งสามารถตรวจสอบได้โดยการอ่านค่าจากคีย์บอร์ด และจาก DPIO ตามลำดับ จากนั้นจึงทำงานฟังก์ชันต่าง ๆ ตามผลที่ได้จากการตรวจสอบ ฟังก์ชันที่ผู้ใช้สามารถใช้บริการได้ผ่านทางคีย์บอร์ดมีดังนี้

5.5.1.1 ฟังก์ชันขอเชื่อมต่อวงจรข้อมูล ทำหน้าที่ส่งสัญญาณขัดจังหวะให้ 8085 เพื่อให้ทำขบวนการเชื่อมต่อวงจรข้อมูล (Establishment procedure) ถ้าขณะนั้นยังไม่มีการเชื่อมต่อวงจรข้อมูล

5.5.1.2 ฟังก์ชันขอปลดวงจรข้อมูล ทำหน้าที่ส่งสัญญาณขัดจังหวะให้ 8085 เพื่อให้ทำขบวนการปลดวงจรข้อมูล(Release procedure) ถ้าขณะนั้นมีวงจรข้อมูลที่มีการเชื่อมต่อใช้งานอยู่

5.5.1.3 ฟังก์ชันขอส่งข้อมูลในช่องสัญญาณ B ทำหน้าที่อ่านข้อมูลจากแผ่นจานแม่เหล็กลงในหน่วยความจำร่วมบนแผงวงจร เพื่อรอให้โปรแกรมบนแผงวงจรอ่านเพื่อส่งไปในช่องสัญญาณ B

5.5.1.4 ฟังก์ชันขอส่งข่าวสารในช่องสัญญาณ D ในวิทยานิพนธ์ฉบับนี้ฟังก์ชันนี้เป็นการจำลองการส่งข่าวสารจากชั้นที่ 3 ไปในส่วนข่าวสารของเฟรม I โดยการรับข่าวสารที่จะส่งจากคีย์บอร์ดเพื่อเขียนลงในหน่วยความจำร่วม เพื่อรอให้โปรแกรมบนแผงวงจรอ่านส่งไปในช่องสัญญาณ D ในรูปของเฟรม LAPD

5.5.1.5 <u>ฟังก์ชันขอทราบรายชื่อแฟ้มข้อมูล</u> สำหรับช่วยผู้ใช้งานในกรณีที่ต้องการส่งข้อมูลในช่องสัญญาณ B แต่ไม่ทราบชื่อแฟ้มข้อมูล โปรแกรมจะทำการอ่านชื่อแฟ้มข้อมูลทั้งหมดของแผ่นจานแม่เหล็กเพื่อแสดงผลที่จอแสดงผล

5.5.1.6 <u>หยุดทำงาน</u> ใช้งานเมื่อผู้ใช้ต้องการหยุดการทำงานของซอฟต์แวร์บน PC ทั้งหมด

และฟังก์ชันที่ทำงานเนื่องจากได้รับสัญญาณขัดจังหวะจากแผงวงจร ดังนี้

5.5.1.7 <u>ฟังก์ชันรับการเชื่อมต่อวงจรข้อมูล</u> ทำหน้าที่แจ้งให้ผู้ใช้ทราบเมื่อได้รับสัญญาณขัดจังหวะจากแผงวงจร แจ้งการขอเชื่อต่อวงจรข้อมูลจากข่าย

5.5.1.8 <u>ฟังก์ชันรับการปลดวงจรข้อมูล</u> ทำหน้าที่แจ้งให้ผู้ใช้ทราบเมื่อได้รับสัญญาณขัดจังหวะจากแผงวงจร แจ้งการขอปลดวงจรข้อมูลจากข่าย

5.5.1.9 <u>ฟังก์ชันรับข้อมูลจากช่องสัญญาณ B</u> โปรแกรมนี้เมื่อได้รับข้อมูลครั้งแรกจะทำการเปิดแฟ้มข้อมูลบนแผ่นจานแม่เหล็ก และอ่านข้อมูลจากหน่วยความจำร่วมลงเก็บในแฟ้มข้อมูลนี้และแจ้งให้ผู้ใช้ทราบผ่านทางจอแสดงผล โดยกำหนดให้แฟ้มข้อมูลมีชื่อเป็นเวลาปัจจุบันที่เปิดแฟ้มข้อมูล.REC

5.5.1.10 <u>ฟังก์ชันรับข่าวสารจากช่องสัญญาณ D</u> เป็นโปรแกรมจำลองการรับข่าวสารซิกแนลลิงจากส่วนข่าวสารของเฟรม I เพื่อการประมวลผลในชั้นที่ 3 โดยข่าวสารที่ได้รับในหน่วยความจำร่วมจะถูกอ่านขึ้นมาแสดงที่จอแสดงผลในลักษณะเฟรมต่อเฟรม

5.5.2 <u>ซอฟต์แวร์บนแผงวงจร</u>

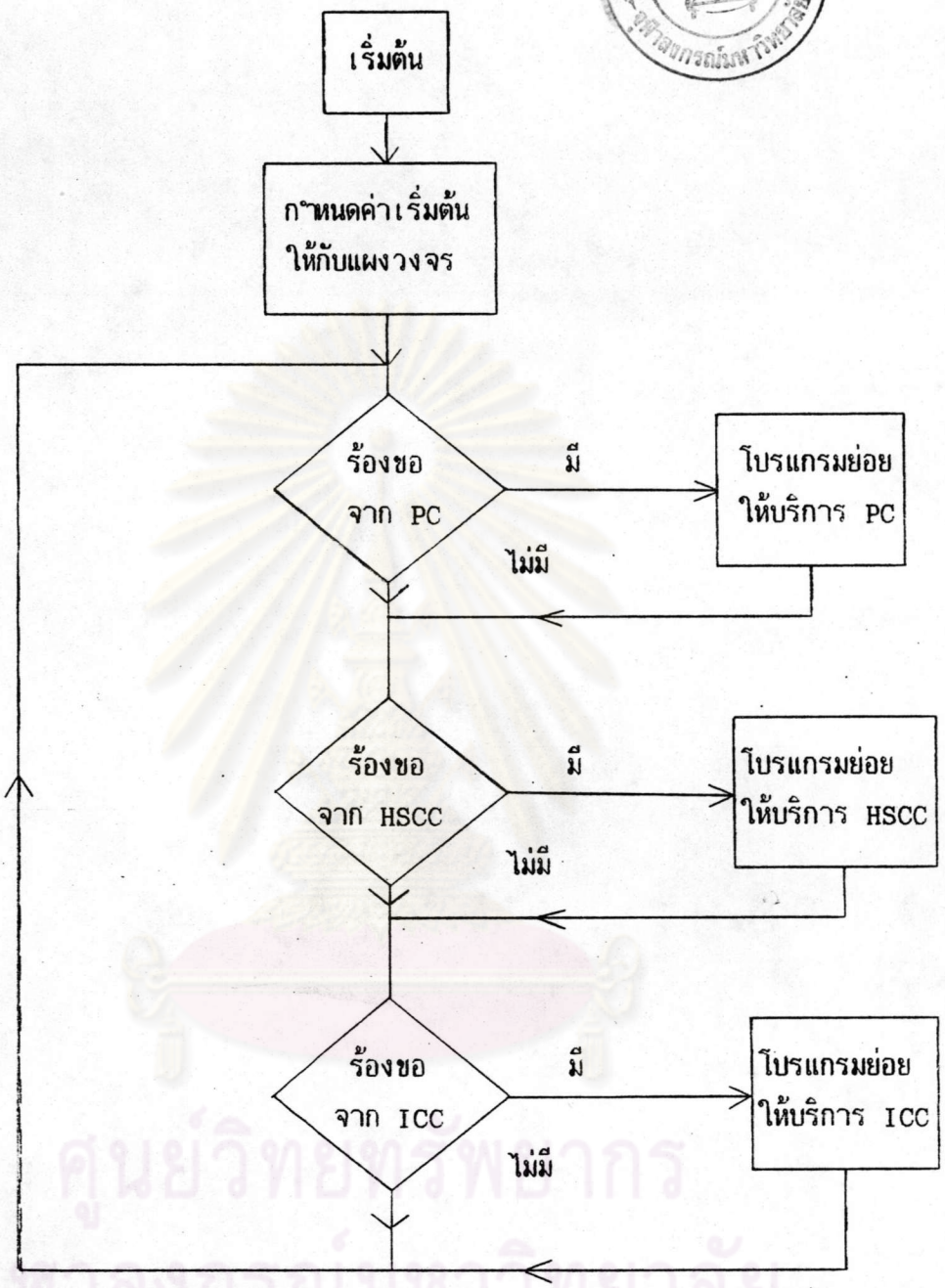
บนแผงวงจรซอฟต์แวร์จะแบ่งเป็น 3 ส่วน คือ

ก. ซอฟต์แวร์ให้บริการ PC

ข. ซอฟต์แวร์ให้บริการ HSCC

ค. ซอฟต์แวร์ให้บริการ ICC

เมื่อมีสัญญาณขัดจังหวะเกิดขึ้น โปรแกรมหลักจะตรวจสอบตัวแปรเก็บสถานะที่เกี่ยวข้อง เพื่อเรียกใช้โปรแกรมย่อยสำหรับเป็นหลักในการให้บริการวงจรแต่ละส่วน ภายในโปรแกรมย่อยเหล่านี้ จะมีฟังก์ชันต่าง ๆ ที่ให้บริการสำหรับอุปกรณ์หลักของวงจรส่วนนั้น ๆ และเมื่อทำงานตามฟังก์ชันที่ต้องการเรียบร้อยก็จะกลับเข้าสู่โปรแกรมหลักอีกครั้งหนึ่ง เพื่อรอการทำงานครั้งต่อไป ขั้นตอนการตรวจสอบตัวแปรสถานะและการเรียกใช้โปรแกรมย่อยแสดงเป็นผังงานได้ในรูปที่ 5.12



รูปที่ 5.12 ผังงานซอฟต์แวร์บนแผงวงจร

ภายในโปรแกรมหลักส่วนที่กำหนดค่าเริ่มต้น จะกำหนดค่าต่อไปนี้

ก. กำหนดค่าเริ่มต้นให้ DPIO จาก 8085 เป็น 0

ข. กำหนดค่าเริ่มต้นสำหรับตำแหน่งหน่วยความจำ RAM ที่ใช้เป็น Stack ที่แอดเดรส FFFFH

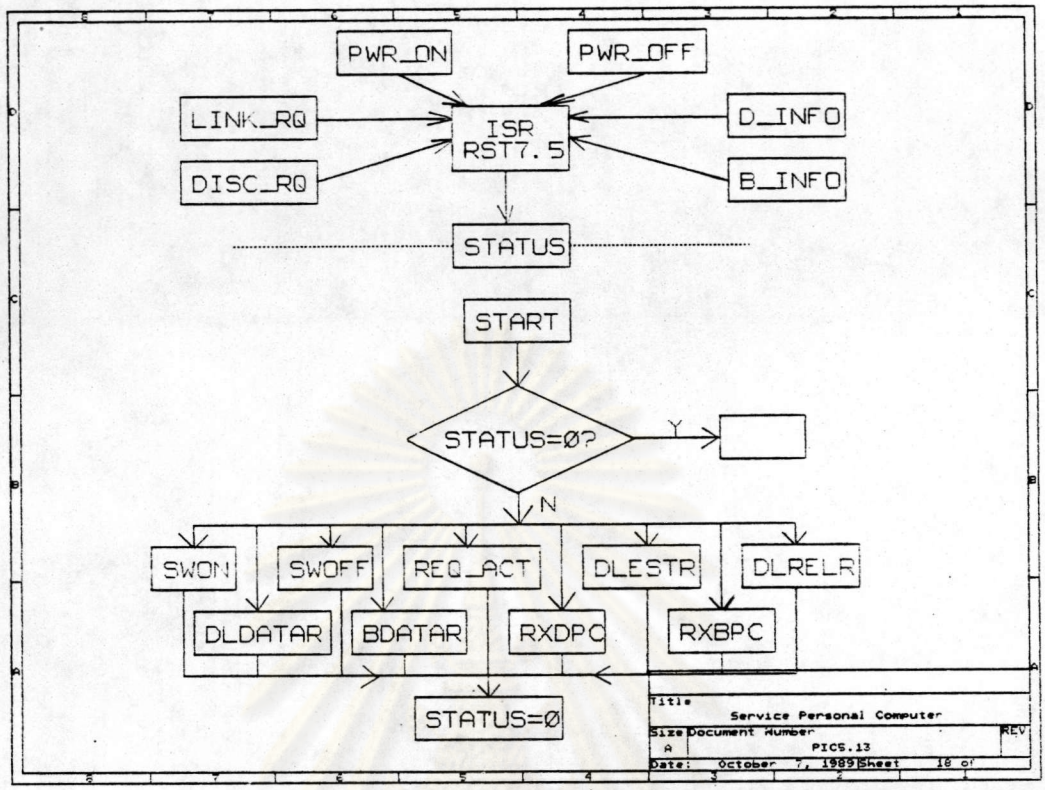
ค. กำหนดค่าเริ่มต้นสำหรับตัวแปรที่ใช้งานโดยโปรแกรมทั้งหมด

5.5.2.1 ซอฟต์แวร์ให้บริการ PC ทำหน้าที่ติดต่อกับ PC โดยกำหนดให้มีโปรแกรมหลักในส่วนนี้ทำงานควบคู่กับโปรแกรมใช้ประจำบริการการขัดจังหวะ (ISR:Interrupt Service Routine) ผ่านตัวแปรที่เก็บสถานะที่เกิดขึ้นจากการทำงานของ ISR และสามารถเขียนเป็นแผนผังได้ดังแสดงในรูปที่ 5.13 และตารางที่ 5.2

ตารางที่ 5.2 ฟังก์ชันสำคัญของโปรแกรมให้บริการ PC

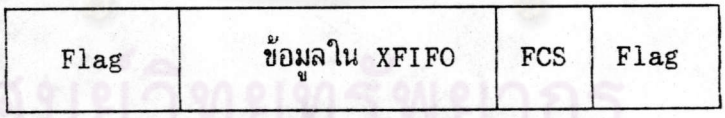
ฟังก์ชัน	หน้าที่
SWON	เริ่มต้นทำงานสำหรับวงจรบนแผงวงจร
SWOFF	สั่งให้แผงวงจรกลับสู่โมด Power down
REQ_ACT	ทำขบวนการ Activate วงจรเชื่อมโยงที่จุด S
DLESTR	ทำขบวนการเชื่อมต่อวงจรข้อมูล
DLRELR	ทำขบวนการปลดวงจรข้อมูล
DLDATAR	ทำขบวนการรับส่งข่าวสารด้วยเฟรม I
BDATAR	ทำขบวนการถ่ายเทข้อมูลในช่องสัญญาณ B
RXDPC	อ่านข้อมูลจากหน่วยความจำร่วมไปเก็บไว้ใน Buffer เพื่อส่งออกในช่องสัญญาณ D
RXBPC	อ่านข้อมูลจากหน่วยความจำร่วมเพื่อส่งออกในช่องสัญญาณ B

5.5.2.2 ซอฟต์แวร์ให้บริการ HSCC ทำหน้าที่ติดต่อกับ HSCC โดยมีวิธีการทำงานเช่นเดียวกับซอฟต์แวร์ให้บริการ PC และสำหรับการกำหนดโมดการทำงานให้ HSCC นั้นเนื่องจาก HSCC ถูกนำมาใช้เพื่อแปลงข้อมูลแบบขนานเป็นแบบอนุกรม เพื่อที่จะสามารถรับส่งข้อมูลจากแพ้มข้อมูลบนจานแม่เหล็กของ PC ไปในช่องสัญญาณ B ได้ ซึ่งการส่งอาจจะทำแบบ Transparent หรือโดยมีโปรโตคอลควบคุมได้[7] สำหรับในที่นี้ได้กำหนดให้ส่งข้อมูลแบบ Transparent โดยให้ข้อมูลที่ส่งอยู่ในรูปของเฟรมที่มีแฟล็กเป็นอักขระคั่นระหว่างเฟรมและมีข้อมูลสำหรับตรวจสอบความถูกต้องต่อท้ายข้อมูลที่ทำการส่ง ทั้งนี้เพื่อความง่ายในการพัฒนาโปรแกรมที่สามารถทดสอบการส่งข้อมูลได้ โดยการคำนวณค่าที่ส่งในส่วนตรวจสอบความถูกต้องจะทำโดย HSCC และเมื่อรับข้อมูลเข้ามา HSCC ก็จะตรวจสอบข้อมูลส่วนนี้ก่อน ถ้าถูกต้องก็จะแจ้งให้ CPU ทราบเพื่อรับข้อมูลได้ ถ้าไม่ถูกต้องก็จะแจ้งให้ CPU ทราบเช่นกัน เพื่อให้โปรแกรมที่รับผิดชอบการถ่ายเทข้อมูลได้จัดการแก้ไขต่อไป

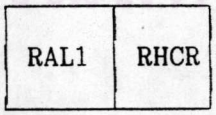


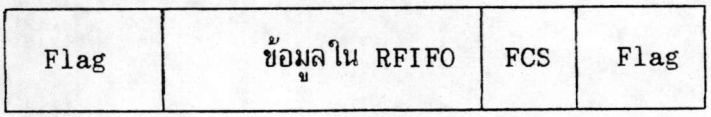
รูปที่ 5.13 ผังงานของโปรแกรมให้บริการ PC

เมื่อกำหนดให้ HSCC ทำงานในโมด Extended Transparent Mode 0 เฟรมที่ส่งออกไปโดย HSCC จะแสดงได้ในรูปที่ 5.14ก และในทิศทางการรับในรูปที่ 5.14ข



ก. การจัดเฟรมสำหรับการส่ง

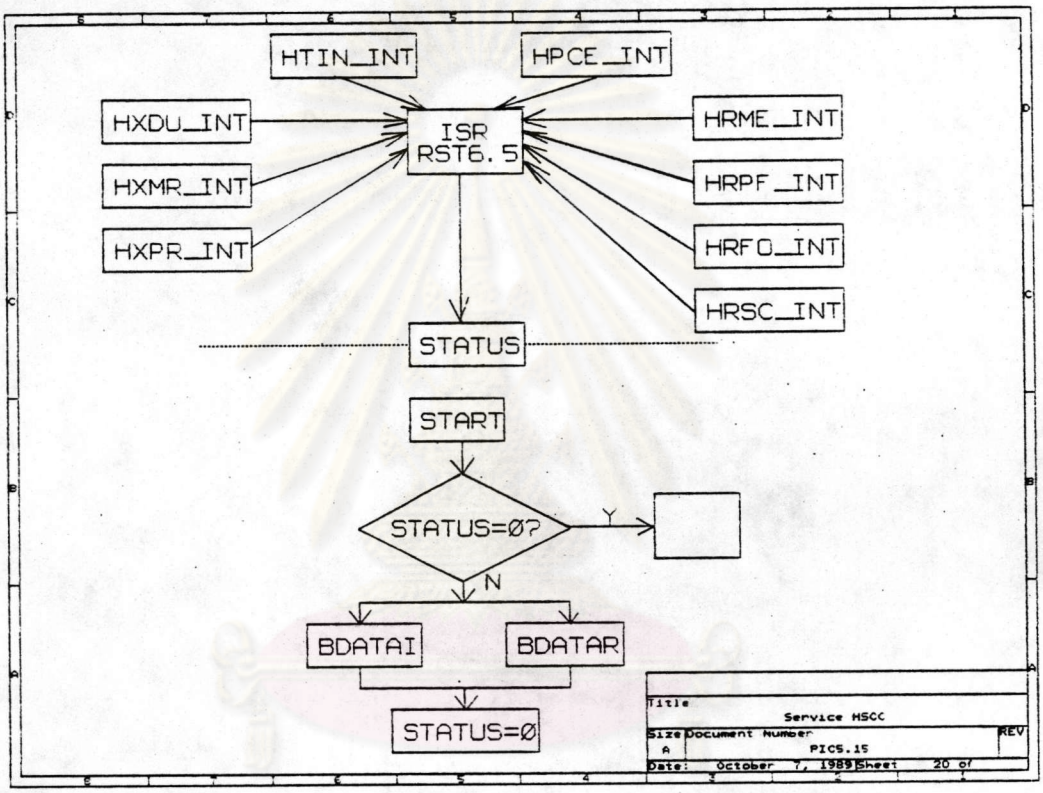




ข. เฟรมที่ได้รับ

รูปที่ 5.14 การจัดเฟรมในโมด Extended Transparent mode 0

ข้อมูลไบต์แรกและไบต์ที่สองต่อจากแฟล็กเบิด นอกจากจะอ่านได้จาก RFIFO แล้วยังสามารถอ่านได้จากรีจิสเตอร์ RAL1 และ RHCR ตามลำดับ การกำหนด HSCC ให้ทำงานในโมดนี้ทำให้โปรแกรมให้บริการ HSCC ในส่วนของโปรแกรมหลักมีเฉพาะโปรแกรมทำหน้าที่อ่านข้อมูลจากหน่วยความจำร่วมเพื่อส่งออก และโปรแกรมทำหน้าที่แจ้งสถานะของการรับส่งข้อมูลในช่องสัญญาณ B ให้ PC ทราบผ่าน 8085 เท่านั้น และสามารถเขียนเป็นแผนผังได้ดังแสดงในรูปที่ 5.15 และตารางที่ 5.3



รูปที่ 5.15 ผังงานของโปรแกรมให้บริการ HSCC

ตารางที่ 5.3 ฟังก์ชันสำคัญของโปรแกรมให้บริการ HSCC

ฟังก์ชัน	หน้าที่
BDATAR	อ่านข้อมูลจากหน่วยความจำร่วมและเขียนลงในรีจิสเตอร์ RFIFO เพื่อส่งออกในช่องสัญญาณ B
BDATAI	แจ้งให้ PC ทราบเมื่อมีข้อมูลในหน่วยความจำร่วมและต้องการให้ PC อ่านออกไปบันทึกลงจานแม่เหล็ก

5.5.2.3 ซอฟต์แวร์ให้บริการ ICC ทำหน้าที่ติดต่อกับ ICC โดยมีการทำงานลักษณะเดียวกันกับโปรแกรมให้บริการ HSCC

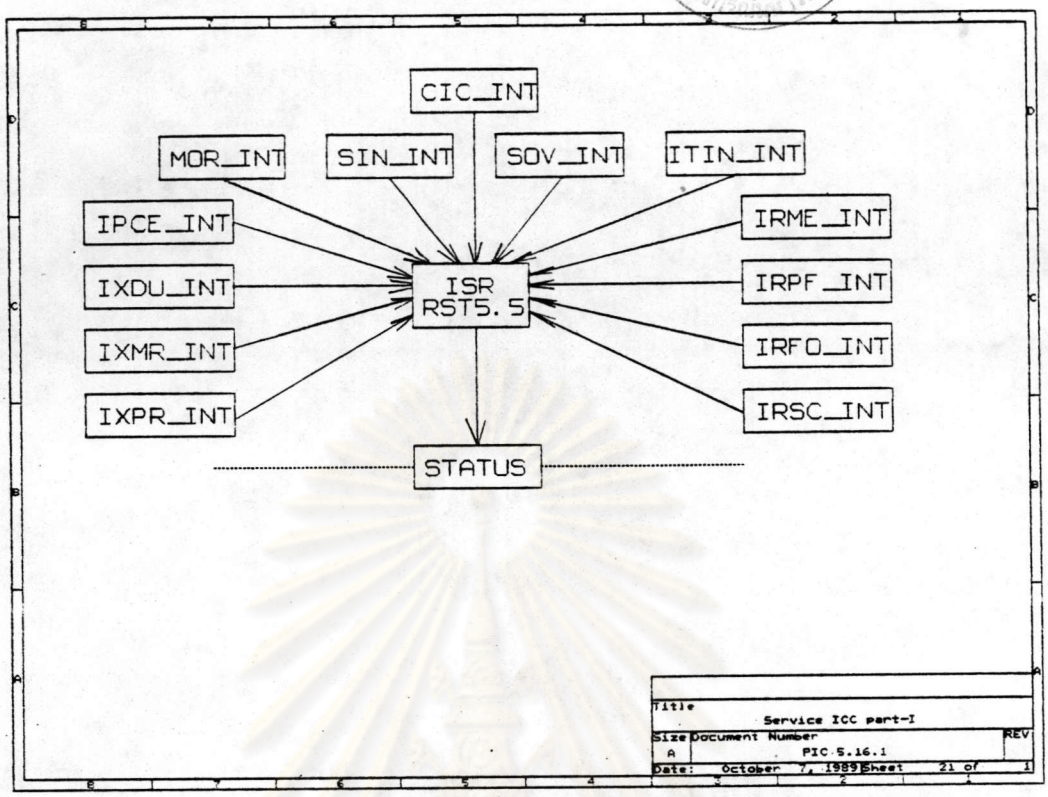
การกำหนดโมดการทำงานให้ ICC นั้นได้กำหนดให้ทำงานในโมด Auto ทั้งนี้เนื่องจากตามเอกสารอ้างอิงของ CCITT Rec.I.441[2] แล้วจำนวนเฟรม I ที่ค้างไม่ได้รับการตอบรับ (Outstanding Frame) สำหรับข่าวสารซิกแนลลิงที่ส่งในช่องสัญญาณ D ค่าโดยปริยายคือ 1 และ ICC ในโมด Auto นี้จะสามารถควบคุมการโต้ตอบตามโปรโตคอล LAPD ในขณะที่มีการรับส่งข่าวสารด้วยเฟรม I ได้นอกเหนือจากการทำฟังก์ชันหลักของชั้นที่ 2 คือการแทรกบิต การคำนวณและตรวจสอบข้อมูลส่วนตรวจสอบความถูกต้องเป็นต้น ซึ่งจะเพียงพอสำหรับการทำงานตามแบบโปรโตคอล LAPD สำหรับ TE ที่มีวงจรข้อมูลใช้งานเพียง 1 วงจร(1LAP) ซึ่งเป็นจำนวนวงจรข้อมูลที่เพียงพอสำหรับ TE ที่ใช้งานทั่วไป[14] ดังนั้นเมื่อกำหนดให้ ICC ทำงานในโมด Auto ทำให้สามารถลดความยุ่งยากของซอฟต์แวร์ในส่วนที่ควบคุมการโต้ตอบระหว่างคู่สนทนาในขณะที่มีการรับส่งข่าวสารด้วยเฟรม RR และ RNR ได้และนอกจากนั้นการเลือกใช้กรณีที่มีเฟรม I ที่ค้างการตอบรับได้เพียง 1 เฟรมทำให้สามารถลดการใช้งานเฟรมREJ และ FRMR ได้[21] โดยในกรณีที่เกิดเหตุการณ์ที่จะต้องใช้เฟรมเหล่านี้ก็จะมีการเรียกใช้ขบวนการฟื้นฟู หรือขบวนการแก้ไขความผิดพลาดที่เกี่ยวข้องแทน

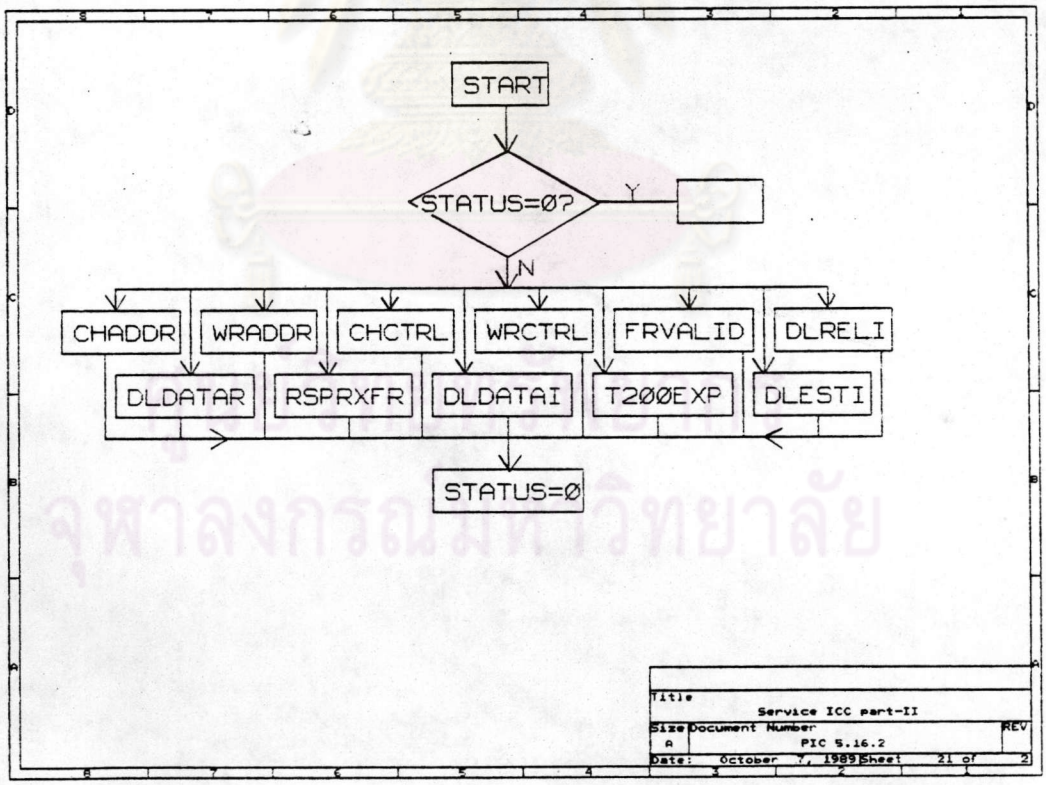
ซอฟต์แวร์ส่วนนี้จะเป็นซอฟต์แวร์หลักของการควบคุมการติดต่อสื่อสารระหว่างแผงวงจรกับข่ายให้เป็นไปตามโปรโตคอล LAPD โดยจะมีหน้าที่ดังต่อไปนี้

ก. สร้างและถอดเฟรม LAPD
ข. ตรวจสอบแอดเดรสของเฟรมที่ได้รับ
ค. ทำขบวนการเชื่อมต่อวงจรข้อมูล
ง. ทำขบวนการปลดวงจรข้อมูล
จ. ทำขบวนการรับส่งข่าวสารด้วยเฟรม I
ฉ. ควบคุมและตรวจสอบการรับและส่งเฟรม I
ช. ควบคุมการโต้ตอบตามแบบโปรโตคอล LAPD สำหรับเฟรมที่ได้รับที่ไม่ใช่เฟรมชนิด I และ S

และสามารถเขียนเป็นแผนผังได้ดังแสดงในรูปที่ 5.16 และตารางที่ 5.4

การทำงานของซอฟต์แวร์บนแผงวงจรนั้น ในกรณีที่เป็นข่าวสารซิกแนลลิง ซอฟต์แวร์ให้บริการ PC กับซอฟต์แวร์ให้บริการ ICC จะทำงานต่อเนื่องกัน ซึ่งจะได้อธิบายเป็นตัวอย่างเพื่อความเข้าใจในหัวข้อ 5.5.3 ต่อไปนี้





รูปที่ 5.16 ผังงานของโปรแกรมให้บริการ ICC

ตารางที่ 5.4 ฟังก์ชันสำคัญของโปรแกรมให้บริการ ICC

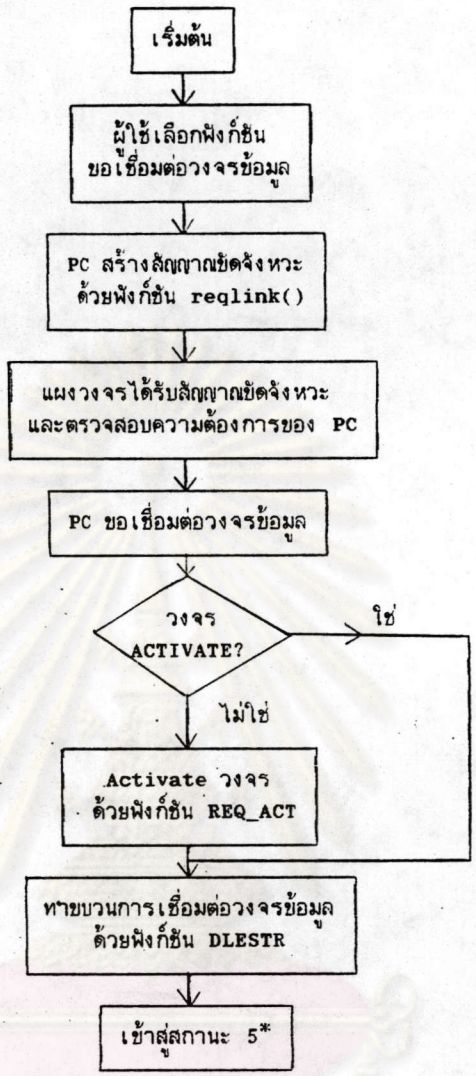
ฟังก์ชัน	หน้าที่
CHADDR	ตรวจสอบแอดเดรสของเฟรมที่ได้รับ
WRADDR	เตรียมข้อมูลส่วนแอดเดรส
CHCTRL	ตรวจสอบส่วนควบคุมเพื่อเลือกเฟรมที่จะใช้ในการโต้ตอบ
WRCTRL	เตรียมข้อมูลส่วนควบคุม
FRVALID	ตรวจสอบความถูกต้องของเฟรมที่ได้รับ
T200EXP	ทำการฟื้นฟูการส่งข่าวสารด้วยเฟรม I เมื่อหมดเวลา T200 โดยไม่ได้รับคำตอบ
IVRETX	ส่งเฟรม I ใหม่เมื่อไม่ได้รับคำตอบ
RSPRXFR	จัดการการโต้ตอบเฟรมที่ได้รับ
DLDATAR	ทำขบวนการส่งข่าวสารด้วยเฟรม I
DLDATAI	แจ้งการได้รับข่าวสารด้วยเฟรม I
DLESTI	แจ้งการได้รับการขอเชื่อมต่อวงจรข้อมูล
DLRELI	แจ้งการได้รับการขอปลดวงจรข้อมูล

5.5.3 ตัวอย่างการทำงานของซอฟต์แวร์

5.5.3.1 ตัวอย่างการขอเชื่อมต่อวงจรข้อมูล ในการขอเชื่อมต่อวงจรข้อมูลจากผู้ใช้นั้น สามารถแสดงเป็นขั้นตอนการทำงานของซอฟต์แวร์ดังรูปที่ 5.17

จากรูปที่ 5.17 จะอธิบายได้ดังนี้ เมื่อผู้ใช้ต้องการเชื่อมต่อวงจรข้อมูลก็จะเรียกใช้โปรแกรมบน PC แล้วเลือกฟังก์ชันขอเชื่อมต่อวงจรข้อมูล เมื่อโปรแกรมรับทราบความต้องการของผู้ใช้ก็จะเริ่มต้นขบวนการของเชื่อมต่อวงจรข้อมูล โดยเรียกใช้ฟังก์ชัน reqlink() ซึ่งจะทำหน้าที่กำหนดสถานะใน DPIO และสร้างสัญญาณขัดจังหวะให้แผงวงจร

เมื่อแผงวงจรได้รับสัญญาณขัดจังหวะจาก PC ก็จะตรวจสอบความต้องการของ PC โดยการอ่านสถานะจาก DPIO และเมื่อทราบความต้องการแล้ว PC ก็จะตรวจสอบว่าขณะนั้นวงจรชั้นที่ 1 ได้รับการ Activate หรือยัง ถ้ายังก็จะทำขบวนการ Activate ก่อน โดยฟังก์ชัน REQ_ACT แล้วจึงเริ่มขบวนการเชื่อมต่อวงจรข้อมูลต่อไป โดยฟังก์ชัน DLESTR

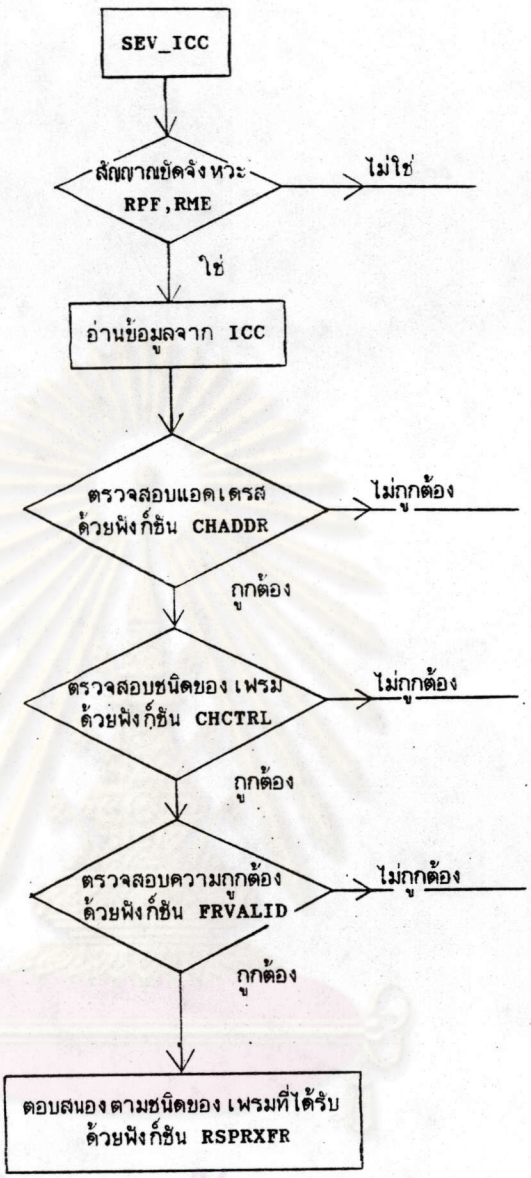


สถานะ 5 : วงจรรอการเชื่อมต่อ

รูปที่ 5.17 ขั้นตอนการขอเชื่อมต่อวงจรข้อมูล

การทำงานของฟังก์ชัน DLESTR จะเริ่มต้นจากการสร้างเฟรม LAPD ซึ่งในที่นี้คือเฟรม SABM โดยอาศัยข้อมูลที่ CPU ส่งมาให้ในรีจิสเตอร์ต่างๆ ของ ICC โดยเรียกใช้ฟังก์ชัน WRADDR, WRCTRL และเมื่อสร้างเฟรมเรียบร้อยก็จะจัดการส่งออกไปโดยฟังก์ชัน SENDFR จากนั้นก็จะอยู่ในสถานะรอการเชื่อมต่อวงจรข้อมูล

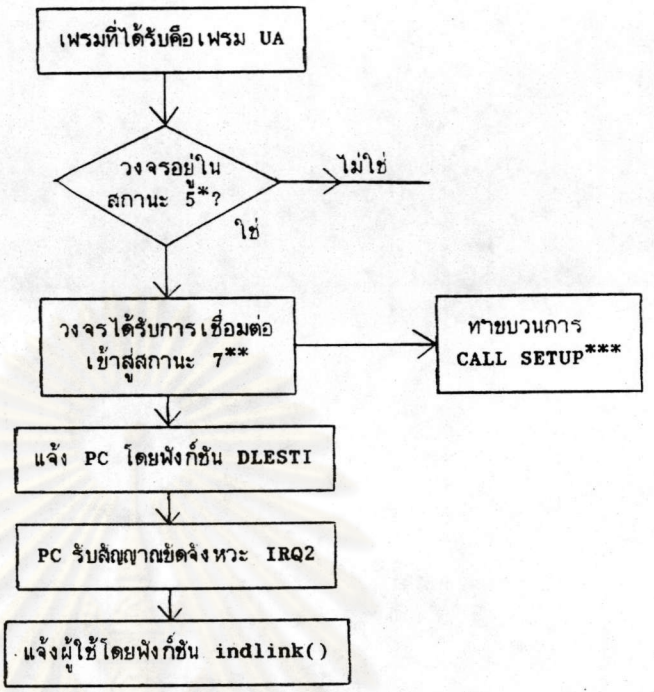
5.5.3.2 <u>ตัวอย่างวงจรข้อมูลได้รับการเชื่อมต่อ</u> เมื่อแผงวงจรได้รับข้อมูลเข้ามาโดยผ่านทาง ICC ในกรณีที่เป็นข่าวสารซิกแนลลิง ซอฟต์แวร์บนแผงวงจรจะทำงานดังแสดงได้ในรูปที่ 5.18



รูปที่ 5.18 ขั้นตอนการรับและตรวจสอบเฟรมข้อมูล

และในกรณีที่เฟรมที่ได้รับคือเฟรม UA ซอฟต์แวร์ก็จะมีการทำงานดังแสดงในรูปที่ 5.19

จากรูปที่ 5.18 อธิบายได้ดังนี้ เมื่อ 8085 ได้รับสัญญาณขัดจังหวะ RPF หรือ RME จาก ICC ซึ่งแสดงว่ามีข้อมูลที่ได้รับอยู่ในรีจิสเตอร์เอง ICC 8085 ก็จะไปอ่านข้อมูลที่ได้รับมานี้ ซึ่ง ICC ได้ทำการแยกข้อมูลส่วนต่างๆ ของเฟรมที่ได้รับเรียบร้อยแล้วและเก็บไว้ในรีจิสเตอร์ต่าง ๆ กัน จากนั้น นำข้อมูลส่วนแอดเดรสไปตรวจสอบโดยฟังก์ชัน CHADDR ถ้าถูกต้องก็ตรวจสอบส่วนควบคุมโดยฟังก์ชัน CHCTRL ต่อไป จากนั้นก็จะนำข้อมูลทั้งเฟรมมาตรวจสอบความถูกต้องของเฟรมอีกครั้งหนึ่ง โดยฟังก์ชัน FRVALID ถ้าทุกอย่างถูกต้องหมด 8085 ก็จะตอบสนองต่อเฟรมที่ได้รับนั้น โดยการเรียกใช้ฟังก์ชัน RSPRXFR ต่อไป



รูปที่ 5.19 ขั้นตอนวงจรข้อมูลได้รับการเชื่อมต่อ

จากรูปที่ 5.19 ถ้าหากว่าเฟรมที่ได้รับนั้นคือเฟรม UA 8085 ก็จะตรวจสอบว่าขณะนั้นแผงวงจรอยู่ในสถานะรอการเชื่อมต่อวงจรข้อมูลหรือไม่ ถ้าใช่ก็จะทำการเปลี่ยนสถานะสู่สถานะวงจรข้อมูลได้รับการเชื่อมต่อ จากจุดนี้จะสามารถทำการรับส่งเฟรม I เพื่อทำขบวนการ Call set up ในระดับโปรโตคอลชั้นที่ 3 ได้

และพร้อมกันนั้น 8085 ก็จะแจ้งให้ PC ทราบโดยฟังก์ชัน DLESTI และสัญญาณขัดจังหวะ เมื่อ PC ได้รับสัญญาณขัดจังหวะและตรวจสอบพบว่าเป็นการแจ้งผลการเชื่อมต่อวงจรข้อมูล PC ก็จะแจ้งให้ผู้ใช้ทราบโดยฟังก์ชัน indlink() ต่อไป

โดยสรุปแล้ว การพัฒนาซอฟต์แวร์สำหรับการควบคุมการโต้ตอบตามแบบโปรโตคอล LAPD นี้ ได้ยึดแนวทางตามเอกสารอ้างอิงหมายเลข[2][10]