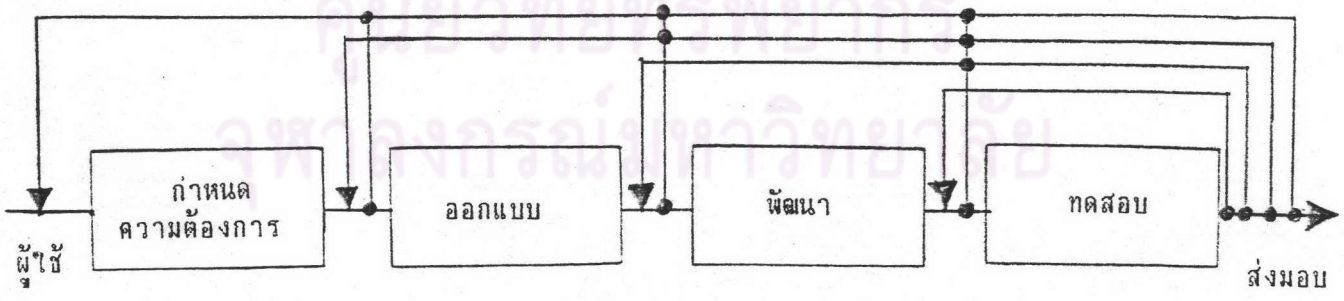
หอสมุดกลาง สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 4

ศึกษาการออกแบบแผ่นวงจรเครือข่ายท้องถิ่นโดยใช้ชุดไอซี TMS380

4.1 ขั้นตอนในการพัฒนาโดยใช้แบบจำลองขบวนการ

หลักการที่จะนำมาเป็นแนวทางในการพัฒนาครั้งนี้ได้ใช้ แบบจำลองขบวนการ (Process model) ในแบบที่เรียกว่า แบบจำลองวนซ้ำเต็มรูป (Fully iterative model) (Beam 1990) โดยในแบบจำลองนี้ได้แบ่งขั้นตอนการพัฒนาผลิตภัณฑ์ไว้ 4 ขั้น คือ กำหนดความต้องการ ออกแบบ พัฒนา และ ทดสอบ ซึ่งเมื่อเสร็จสิ้นทั้ง 4 ขั้นตอนนี้แล้วก็จะมีการส่งมอบออกไปใช้หรือเข้าสู่ตลาด หลักการที่สำคัญของแบบจำลองขบวนการแบบนี้ก็คือ ในแต่ละขั้นตอนการพัฒนานั้นสามารถมีการส่งผลสะท้อนกลับให้ไปเริ่มทำขั้นตอนก่อนหน้านั้นได้ทุกระยะ ซึ่งจะเห็นได้ว่ามีความยืดหยุ่นมากจึงสามารถสนองความต้องการได้ตรงจุดและรวดเร็วที่สุด ทั้งยังตรงกับสภาพความเป็นจริงในการพัฒนาอีกด้วย



รูปที่ 4.1 แบบจำลองขบวนการ ชนิด แบบจำลองวนซ้ำเต็มรูป (Beam 1990)

ในการพัฒนาครั้งนี้ ได้ทำเพียง 2 ขั้นตอน คือ กำหนดความต้องการ และออกแบบ แต่ก็ได้ให้แนวทางที่จะทำขั้นตอนต่อไปไว้ด้วย

4.2 ข้อกำหนดในการออกแบบ

ในขั้นตอนการกำหนดความต้องการในการพัฒนาผลิตภัณฑ์นั้น สามารถแบ่งเป็น 2 ขั้นตอนย่อย (Protopapas 1988) คือ ข้อกำหนดตามความต้องการของผู้ใช้ (Product requirements specification) และ ข้อกำหนดในการออกแบบระบบ (System design specification) ซึ่งข้อกำหนดในแบบแรกจะมาจากการรวบรวมข้อมูลเกี่ยวกับความต้องการของผู้ใช้ ซึ่งในวงการอุตสาหกรรมจริงๆแล้วผู้ที่ทำหน้าที่นี้ คือ ฝ่ายการตลาด ซึ่งจะส่งต่อข้อกำหนดนี้ไปยังฝ่ายวิศวกรรม ฝ่ายวิศวกรรมก็จะทำการนำความต้องการเบื้องต้นนี้ มาวิเคราะห์ความเป็นไปได้และตั้งข้อกำหนดต่างๆทางด้านเทคนิคขึ้นมา ตามที่เรียกว่า ข้อกำหนดในการออกแบบระบบ ดังที่กล่าวข้างต้น

สำหรับในการวิจัยครั้งนี้ ได้รวบรวมข้อกำหนดทั้ง 2 ขั้นตอนไว้ด้วยกัน เพราะเป็นการทำวิจัยโดยผู้เดียว และ มุ่งเน้นในด้านการศึกษาเป็นหลัก

ข้อกำหนดในการออกแบบแผ่นวงจรอะแดปเตอร์สำหรับเครือข่ายท้องถิ่น มีดังนี้

4.2.1 เป็นแผ่นวงจรที่ใช้โพรโตคอลแบบ โทเกนริง

4.2.2 ใช้กับเครื่องไมโครคอมพิวเตอร์รุ่น IBM PC AT และรุ่นที่คอมแพติเบิล

4.2.3 สามารถต่อกับตัวกลางนำสัญญาณ และอุปกรณ์ต่อสายต่างๆ ได้ตามมาตรฐานของ IEEE802.5

4.2.4 กำหนดระบบสัญญาณเชื่อมต่อกับระบบบัสของ เครื่อง IBM PC AT โดยไม่มีปัญหาหรือมีปัญหาน้อยที่สุดกับระบบสัญญาณของอินเตอร์เฟสการ์ดอื่นๆที่ต่อบนเครื่องเดียวกัน

4.2.5 มีไฟสัญญาณแสดงสถานะในการต่อเข้ากับริงหรือถอดออกจากริง

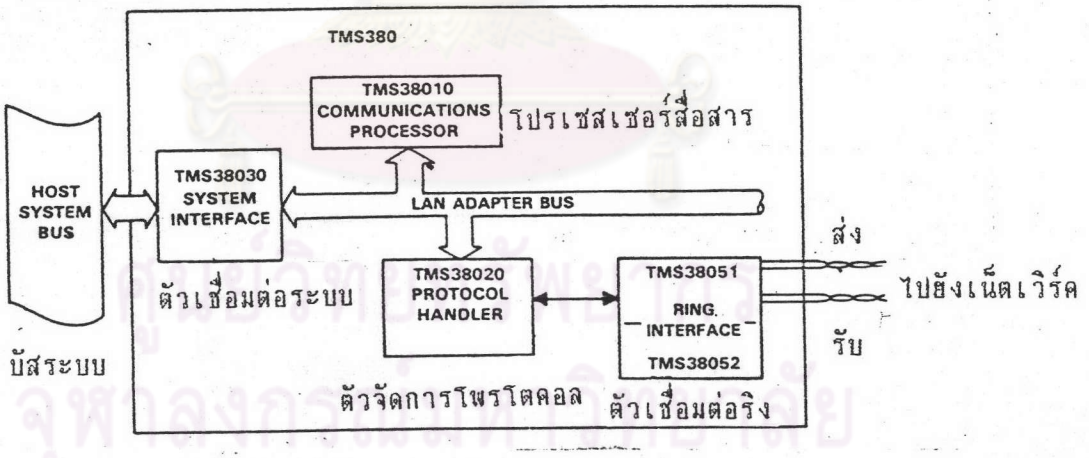
4.2.6 แสดงแนวทางในการพัฒนาซอฟต์แวร์ที่ใช้ในการควบคุมแผ่นวงจร

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

I17340718

4.3 การใช้ TMS380 ชิปเซตเป็นอุปกรณ์หลักในการออกแบบแผ่นวงจร (TMS380 Adapter Chipset User's Guide 1986)

อุปกรณ์หลักที่ใช้ในการพัฒนาแผ่นวงจรนี้ เป็นชุดแผงวงจรรวม(Integrated circuit) เรียกว่าชุด TMS380 ชิปเซต ของบริษัท เท็กซัสอินสตรูเม้นท์(Texas Instruments) ซึ่งออกมาตั้งแต่ปี ค.ศ.1986 เหตุผลที่เลือกชิปเซตชุดนี้มาเป็นอุปกรณ์หลักเพราะว่า เป็นของบริษัทเดียวที่ผลิตชิปโทเกนริงออกสู่ตลาด (Non proprietary)เพื่อให้บริษัททั่วไปนำไปพัฒนาแผ่นวงจรเครือข่ายท้องถิ่น และเป็นชิปเซตที่บริษัทผู้ผลิตแผ่นวงจรเครือข่ายท้องถิ่นนำไปใช้มากที่สุด ในการผลิตแผ่นวงจรโทเกนริง (Greenfield 1989) นอกจากบริษัทไอบีเอ็มซึ่งใช้ชิปเซตของตัวเอง(Proprietary) ซึ่งบริษัทเหล่านี้ได้แก่ 3COM, Gateway Communication, NCR, Proteon, Pure data, Racore และ Western Digital เป็นต้น



รูปที่ 4.2 แสดงโครงสร้างของ TMS380 ชิปเซต
(TMS380 Adapter Chipset User's Guide 1986)

ดังนั้นในการเลือกใช้ชุดชิปเซตเบอร์นี้จึงเป็นที่เชื่อถือได้ และตรงตามมาตรฐานอุตสาหกรรมที่มีใช้กันอยู่ และแม้ว่าภายหลังบริษัทเท็กซัสอินสตรูเม้นท์ได้ออกชิปรุ่น TMS380C 16/04 เมื่อปี 1989 ก็ตามแต่โครงสร้างภายในและสถาปัตยกรรมต่างๆยังคงเป็นแนวเดิมเกือบทั้งหมด เพียงแต่ลดจำนวนชิปลงและเพิ่มความเร็วในการทำงานขึ้น

4.3.1 ลักษณะทั่วไปของชิปเซต

TMS380 เป็น ชิปเซตเกี่ยวกับแลนของบริษัทเท็กซัสอินสตรูเม้นท์ซึ่งได้ทำการพัฒนาร่วมกับบริษัทไอบีเอ็มประกอบด้วยไอซี 5 ตัว คือ TMS38030 System Interface , TMS38010 Communications Processor , TMS38020 Protocol Handler และ TMS38051 กับ TMS38052 Ring Interface ต่อกันดังรูปที่ 4.2 โดยทั้งนี้จะมีคุณสมบัติต่างๆเข้ากันได้กับมาตรฐานแลนอินเตอร์เฟสการ์ด ของไอบีเอ็มโทเกนริง

ก. TMS380 ชิปเซต ใช้เทคโนโลยีโทเกนพาสซิ่งแอคเซส ซึ่งตรงกับมาตรฐาน IEEE 802.5

ข. ชิปเซต สามารถส่งข้อมูลด้วยความเร็ว 4 Mbit/sec โดยใช้สื่อการส่งเป็นสาย ตีเกลียวคู่ หรือ สายใยแสง

ค. มีตัวที่ทำให้เน็ตเวิร์คมีความเชื่อถือได้ เช่น มีวงจรตรวจสอบความผิดพลาด และการตรวจสอบโดยซอฟต์แวร์ที่อยู่ในตัวชิป และนอกจากนี้ยังมีความสามารถทางด้านการบริหารเน็ตเวิร์ค อื่นๆอีก

ง. มีระบบการเชื่อมต่อกับระบบบัสของโฮสต์(Host System Bus) แบบ ดีเอ็มเอ (Direct Memory Access) ซึ่งจะทำการส่งผ่านข้อมูลไปมาทำได้รวดเร็วมาก

จ. มี ซีพียู (CPU : Central Processing Unit) 16 บิตช่วยในการทำงาน โดยมี แรม(RAM) บัฟเฟอร์(buffer) อยู่ภายในชิปด้วย

ฉ. มี รอม (ROM) ภายในชิป ซึ่งจะเก็บซอฟต์แวร์ที่ช่วยในการจัดการเกี่ยวกับโพรโตคอล เรียก โพรโตคอลแฮนเดลอร์(Protocol handler)

ช. มีชิป 2 ตัวสำหรับอินเตอร์เฟสกับอุปกรณ์เชื่อมต่อ

4.3.2 สถาปัตยกรรมของแลนอินเตอร์เฟสการ์ดที่ใช้ TMS380

4.3.2.1 การเชื่อมต่อกับระบบ (SIF : System interface)

ก. SIF สามารถส่งผ่านข้อมูลกับโฮสต์ ได้สูงถึง 40 Mbit/s โดยผ่านทาง ดีเอ็มเอ โดยที่ระบบบัสของโฮสต์เลือกได้ 2 แบบคือ 16 บิต ของ ซีพียู ตระกูล iAPX86 และตระกูล 3200016 และ 32 บิต ของซีพียูตระกูล 68000

ข. SIF ควบคุมโดย บล็อกคำสั่ง(Command block) โดยมีลักษณะเหมือนภาษาระดับสูง เช่น TRANSMIT,RECIEVE

ค. SIF มีแอดเดรส 24 บิต ต่อกับ ระบบโฮสต์ และมีการทำ ดีเอ็มเอ แบบ "Scatter write gather read" ซึ่งจะทำให้ส่งข้อมูลที่ไม่อยู่ใน บล็อกหน่วยความจำ(Memory block) ที่ติดกัน สามารถรับส่งได้โดยผ่านลิงค์ลิสต์(Link-list) และสามารถโปรแกรมการทำงาน ได้ว่าเป็นแบบ การส่งผ่านแบบเบิสต์ (Burst transfer) หรือ แบบขโมยไซเคิล (Cycle steal) ได้หรือเลือกการป้องกันโดยใช้พาริตี้ (Parity protection) ได้ ซึ่งทำให้ผู้ออกแบบออกแบบให้เหมาะสมกับแต่ละระบบบัสได้

4.3.2.2 โปรเซสเซอร์สื่อสาร (CP : Communication Processor)

ก. CP มี ซีพียู ขนาด 16 บิต และมีแรมในชิป ขนาด 2.75 Kbyte

ข. CP จะเอ็กซีคิวต์ (Execute) ซอฟต์แวร์ที่บรรจุอยู่ในแรมในชิปของ TMS38020 จะทำหน้าที่เป็นบัฟเฟอร์ ของ เฟรมข้อมูล รับและส่ง

ค. ซีพียู สามารถจัดการทำงานเป็นแบบ ไซเคิลเดี่ยว (Single-cycle) ของ แลนอินเตอร์เฟสการ์ดแบบ 3 MHz เพื่อเพิ่มอัตราปริมาณงานให้สูงที่สุด

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ง. สามารถขยายหน่วยความจำ ได้ถึง 42 Kbyte

ค. แรมในชิป และ หน่วยความจำขยาย (Expansion memory) เป็นแบบป้องกันพาริตี้

4.3.2.3 โพรโตคอลแฮนเดลอร์ (PH : Protocol Handler)

ก. PH จะเป็นแบบโพรโตคอลฟังก์ชันที่ใช้ฮาร์ดแวร์สำหรับแลนอินเตอร์เฟสการ์ด ซึ่งเข้ากันได้ (Compatible) กับ IEEE802.5 ตัวรอมในชิป ซึ่งมีขนาด 16 Kbyte จะเก็บซอฟต์แวร์ซึ่งจะ เอ็กซีคิวต์ โดย CP โปรแกรมนี้จะจัดการเรื่องการปฏิบัติการของริง , การบริการ ในเรื่องการบริหารแลน , การ วินิจฉัย (Diagnostic) ชิปเซต

ข. PH ใช้ระบบเข้ารหัส ถอดรหัส และ เฟรม โดยวิธีดิฟเฟอร์เรนเชียลแมนเชสเตอร์

ค. PH มี สเตทแมชีน (State machine) ซึ่งจะเป็นตัวคอยจับโทเกนส่งและรับเฟรม จัดการเกี่ยวกับบัฟเฟอร์แรม ของ ชิปเซต และควบคุมระบบโทเกนพาริตี้

ง. มี ดีเอ็มเอ 4 ช่อง(รับ 2 ส่ง 2) จะทำให้เพิ่มความเร็วในการทำการส่งผ่านเฟรม ระหว่าง ริง และ บัฟเฟอร์แรม

4.3.2.4 ริงอินเตอร์เฟส (RI : Ring Interface)

ก. TMS38051, TMS38052 ทำหน้าที่เป็น ริงอินเตอร์เฟสทรานซีฟเวอร์ (Ring interface tranciever) และ ริงอินเตอร์เฟสคอนโทรลเล่อร์ (Ring interface controller)

ข. ชิปนี้จะมีวงจรทั้งดิจิตอล (Digital)และแอนะลอก (Analog) เพื่อจะต่อชิปเซตกับโทเกนริงแลน 4Mbit/s ผ่าน ช่องส่ง และ ช่องรับ ที่แยกจากกัน

ค. RI จะมีสัญญานนาฬิกาของริง เมื่ออยู่ในโหมดแอคทีฟมอนิเตอร์ และมี เฟสล็อกลูป(Phase lock loop) สำหรับทำการกู้สัญญาณ

นาฬิกาการตรวจจับข้อมูล และ การปรับเฟส

ง. RI ยังมีสัญญาน แฟนทอมไดร์ว (Phantom drive) เพื่อส่งไปยัง ตัวรวมสาย (Wriring concentrator) มีเส้นทางวนลูปกลับ สำหรับการทำ การทดสอบการวินิจฉัย และ ตรวจจับข้อผิดพลาดของสาย

4.3.4 การให้บริการการบริหารแลน

TMS380 จะทำการบริการการบริหารแลน โดยการส่งผ่านทาง MAC เฟรม ซึ่งจะทำการแลกเปลี่ยนกันระหว่างอินเตอร์เฟสการ์ดบนริง

4.3.4.1 MAC เฟรม เป็นเฟรมพิเศษ ซึ่งใช้ในการ ควบคุมการปฏิบัติการของริงแต่ไม่ได้ใช้ในการส่งข้อมูล

4.3.4.2 มาตรฐาน IEEE802.5 ได้กำหนด MAC เฟรมมาตรฐานไว้ 6 เฟรม แต่ TMS380 ได้บรรจุ MAC เฟรม ไว้ใช้งานถึง 20 กว่า เฟรม เพื่อใช้ในการรวบรวม ความผิดพลาดต่างๆซึ่งจะรายงานโดย ชิปเซตโดยอัตโนมัติ คอยตรวจสอบและแก้ไขสถานะและพารามิเตอร์ (Parameter)ต่างๆ ของ แลน ตัวอย่าง MAC เฟรมที่ใช้ใน TMS380 เช่น response , beacon , ring purge ,stand by monitor present,lobe media test , change parameters , request เป็นต้น

4.3.5 การส่งและการรับเฟรม

4.3.5.1 TMS380 จะจัดให้มีลิงค์ลิสต์ อินเตอร์เฟสกับระบบบัสของโฮสต์ซึ่งจะ ควบคุมโดยคำสั่งระดับสูง ในระหว่างการทำขบวนการเริ่มต้น (Initialize)ชิปเซตนั้นโฮสต์ก็จะจัดระบบ อินเตอร์เฟสต่างๆ โดยที่ ชิปเซต จะถูกจัดให้เหมาะสมกับระบบบัสของโฮสต์ เช่น เงื่อนไขการอินเตอร์รัปต์ (Interrupt) , ความถี่ และ ขนาดบัฟเฟอร์ เป็นต้น ข้อมูลคำสั่งและ สถานะ ถูกส่งผ่านระหว่างอินเตอร์เฟสการ์ดกับโฮสต์ โดยใช้ บล็อกควบคุม 2 แบบ ดังนี้

ก. SCB (System Command Block)

ข. SSB (System Status Block)

การจะส่งเฟรมนั้น ตอนแรกโฮสต์ต้องทำการอินเตอร์รัปต์ชิปเซต แล้ว ชิปเซตก็จะทำ การอ่านแบบ ดีเอ็มเอ SCB ซึ่ง SCB นี้จะ บรรจุ คำสั่งในการส่ง และ แอดเดรสเริ่มต้น ของ ทรานสมิตลิสต์ (Transmit list) ซึ่งอยู่ในหน่วยความจำของโฮสต์ จากนั้น ชิปเซต จะส่งผ่านทรานสมิตลิสต์ จากโฮสต์ไปยังบัฟเฟอร์แรมใน ชิปเซต แล้ว ชิปเซต ก็จะจับโทเกน และส่ง เฟรม ไปใน แลนเฟรม จะวิ่งวนไปหาปลายทาง ซึ่งจะ ก้อปปี้ข้อมูลและคืนเฟรม กลับไปยังโหนดที่ส่ง ซึ่งจะดึงเฟรมออก และปล่อยโทเกนออกมาเหมือนเดิม SSB จะถูกอัพเดท (Update) โดยการ์ดและส่งผ่านไปยังโฮสต์ สำหรับการรับก็คล้ายกัน เพียงแต่ทำในทางตรงกันข้าม

4.4 การออกแบบระบบบัสเพื่อติดต่อกับระบบบัสของ IBM PC AT

การส่งผ่านข้อมูลไปมาระหว่างเน็ตเวิร์คอินเตอร์เฟสการ์ด กับ หน่วยความจำของไมโครคอมพิวเตอร์นั้นมีเทคนิค 3 แบบ ใหญ่ๆ (Dawkins 1989) คือ

1. ดีเอ็มเอ (Direct Memory Access)

จะทำการส่งผ่านข้อมูลโดยไม่ต้องผ่าน ซีพียู เพราะตัวควบคุม ดีเอ็มเอ จะควบคุมระบบบัสทั้งหมดแทนซีพียู ข้อดีของระบบนี้ คือ ทำให้การส่งข้อมูลจำนวนมากๆทำได้รวดเร็ว เพราะไม่ต้องผ่านซีพียู (วิธีนี้ได้นำไปใช้ในการส่งผ่านข้อมูลระหว่างดิสก์และหน่วยความจำในเครื่อง) แต่ข้อเสียคือ ในการส่งแต่ละครั้งต้องมีการจัดค่าต่างๆเสียก่อน และพื้นที่ข้อมูลที่จะส่งก็ต้องจัดให้อยู่ติดกันเสียก่อนจึงจะส่งผ่านได้ เพราะฉะนั้นการส่งผ่านข้อมูลจำนวนไม่มากทำให้เสียเวลาในการจัดเตรียมต่างๆนี้

2. โปรแกรมไอโอ (Programmed I/O)

วิธีการส่งแบบนี้จะทำได้รวดเร็วกว่าดีเอ็มเอเพราะข้อมูลในหน่วยความจำจะสามารถส่งผ่านได้โดยการกอปปี้โดยตรง ไปยังช่องทาง I/O (I/O port) ซึ่งในระบบไมโครคอมพิวเตอร์ที่ใช้ ซีพียูตระกูล 80X86 จะสามารถใช้คำสั่ง inport(..) และ outport(..) ได้ทันที และข้อดีอีกอย่างคือมันจะไม่เปลืองเนื้อที่หน่วยความจำด้วย

3. หน่วยความจำร่วม (Shared memory หรือ Shared RAM)

จะเป็นวิธีการที่รวดเร็วที่สุดในการส่งผ่านข้อมูล และคำสั่งที่จะใช้ในการส่งผ่านข้อมูลก็เป็นคำสั่งเดียวกับการส่งผ่านกับหน่วยความจำ ซึ่งคำสั่งเหล่านี้มีความยืดหยุ่นในการใช้งานมาก เช่นสามารถใช้ระบบดัชนีได้ หน่วยความจำบนแผ่นวงจรเน็ตเวิร์คอินเตอร์เฟสการ์ด จะเสมือนหนึ่งเป็นหน่วยความจำของระบบด้วยดังนั้นข้อเสียที่สำคัญของระบบนี้คือ ต้องกินเนื้อที่หน่วยความจำของระบบไปจริงๆ และอาจมีปัญหาตามมาเกี่ยวกับการแย่งใช้เนื้อที่หน่วยความจำเดียวกัน โดยโปรแกรมประยุกต์อื่นๆ

จากเทคนิคทั้ง 3 วิธีนี้ ในการออกแบบแผ่นวงจรเน็ตเวิร์คอินเตอร์เฟสการ์ดจริงๆอาจมีการใช้เทคนิคมากกว่า 1 แบบผสมกันไป เช่น ของ ไอบีเอ็มใช้เทคนิคหน่วยความจำร่วม และ โปรแกรมไอโอ แต่บริษัทผู้ผลิตรายอื่นที่ใช้ชิปเซตของเท็กซัสอินสตรูเม้นท์จะใช้เทคนิค ดีเอ็มเอ และ โปรแกรมไอโอ ผสมกัน เนื่องจากตัวชิปได้สนับสนุนการทำงานในแบบนี้ ซึ่งรายละเอียดได้กล่าวแล้ว ในหัวข้อ 4.3 และในการพัฒนาครั้งนี้ก็จะยึดเทคนิคตามแนวนี้ด้วยเช่นกัน

4.4.1 สิ่งที่ต้องพิจารณาในการเชื่อมต่อระบบบัสของเครื่อง PC AT

เนื่องจากระบบบัสของเครื่อง IBM PC AT นั้นมีการต่อใช้งานร่วมกันของอุปกรณ์หลายชนิดและยังมีอุปกรณ์เพิ่มเติมภายนอกที่ต่อเข้าไปอีกหลายชนิดเช่นกัน ดังนั้นสิ่งที่จะต้องพิจารณาในการออกแบบการเชื่อมต่อระบบบัส คือ

1. แอดเดรสของอินพุทเอาต์พุตพอร์ต (I/O Port address)
2. แอดเดรสของหน่วยความจำบนบอร์ด
3. ช่องของระบบ ดีเอ็มเอ (ดีเอ็มเอ channel)
4. ระดับของการเรียกอินเตอร์รัปต์ (Level of Interrupt)
5. การจัดระบบบัสต่างๆ เช่น การมัลติเพล็กซ์สัญญาณ ระบบสัญญาณนาฬิกา ขนาดของบัสต่างๆ และการเรียงลำดับบิตของบัส เป็นต้น

เนื่องจากระบบการส่งผ่านข้อมูลโดยใช้ TMS380 ชิปเซต ได้ใช้

เทคนิคแบบ ดีเอ็มเอ และ โปรแกรมไอโอ ดังนั้นจึงไม่มีเนื้อที่หน่วยความจำบนแผ่นวงจรที่ไปซ้ำกับเนื้อที่หน่วยความจำของไมโครคอมพิวเตอร์ ดังนั้นจึงไม่ต้องพิจารณาในหัวข้อที่ 2.

4.4.2 การออกแบบระบบแอดเดรสของอินพุตเอาต์พุตพอร์ต

บนระบบบัสของเครื่อง IBM PC AT นั้นกำหนดแอดเดรสของพอร์ตหรือหมายเลขพอร์ตไว้ดังตารางที่ 4.1 หมายเลขพอร์ตทั้งหมดของเครื่องรุ่น IBM PC AT จะเริ่มตั้งแต่ 0000h ถึง 03FF โดยที่ อุปกรณ์บนบอร์ดของเครื่องไมโครคอมพิวเตอร์จะใช้พอร์ตหมายเลข 0000h-01FFh ส่วนพอร์ต 0200h ถึง 03FFh จะใช้กับอุปกรณ์เชื่อมต่อเพิ่มเติมที่ผ่านทางช่องเสียบบนเครื่อง ซึ่งแอดเดรสในส่วนนี้เป็นส่วนที่เราสามารถนำมาใช้ได้ จากตารางจะเห็นว่า ช่วงที่หมายเลขพอร์ตที่สำรองไว้อยู่จะมีสองช่วง คือ 360h-36Fh และ 3C0h-3CFh ซึ่งในการออกแบบครั้งนี้จะเลือกเอาหมายเลขพอร์ตช่วง 360h-363h เป็นหมายเลขพอร์ตของแผ่นวงจร ซึ่งพอร์ตที่ใช้นี้จะใช้ทำการติดต่อและส่งผ่านข้อมูลกับรีจิสเตอร์ 4 ตัวของชิปเซต คือ รีจิสเตอร์ข้อมูล, รีจิสเตอร์ข้อมูลที่มีการเพิ่มแอดเดรสโดยอัตโนมัติ, แอดเดรสรีจิสเตอร์ และ อินเตอร์รัปต์รีจิสเตอร์ โดยที่หมายเลขพอร์ตของรีจิสเตอร์แต่ละตัวจะเป็น 360h, 361h, 362h และ 363h ตามลำดับ การอ้างอิงไปยังพอร์ตต่างๆเหล่านี้ต้องมีการถอดรหัสของแอดเดรสบัส เพื่อให้ได้ตรงตามพอร์ตที่กำหนดด้วย ซึ่งการถอดรหัสแอดเดรสนั้นจะใช้วิธีการแบบ แอดเดรสคงที่ (Fixed address decoding) โดยจะนำเอาสายแอดเดรสสายที่ 0-15 มาผ่านวงจรตรรกะแล้ว AND กันกับสัญญาณ IOR และ IOW เพื่อจะให้ได้สัญญาณ CS (Chip Select) เพื่อไปเลือกอุปกรณ์อินพุตเอาต์พุตที่ต้องการ

4.4.3 การออกแบบช่องทางของระบบ ดีเอ็มเอ

ในเครื่อง IBM PC AT มีไอซีเบอร์ 8237 เป็นตัวควบคุม ดีเอ็มเอ จำนวนสองตัวเพิ่มจาก รุ่น XT ซึ่งมีเพียงแค่ตัวเดียว ทำให้สามารถมีช่องของระบบ ดีเอ็มเอ ได้ถึง 8 ช่องตามตารางที่ 4.2

ในช่องที่ 0-3 นั้นออกแบบใช้กับข้อมูล 8 บิตแต่ว่า ช่อง 4-7 ที่เพิ่ม

หมายเลขพอร์ตฐานสิบหก	ชื่ออุปกรณ์
000-01F	ดีเอ็มเอคอนโทรลเลอร์หมายเลข 1, 8237A-5
020-03F	อินเตอร์รัพต์คอนโทรลเลอร์หมายเลข 1, 8259A ตัวหลัก
040-05F	ไทเมอร์ 8254-2
060-06F	8042 คีย์บอร์ด
070-07F	นาฬิกา และ NMI และซีมอสแรม
080-09F	DMA เพจรีจิสเตอร์
0A0-0BF	อินเตอร์รัพต์คอนโทรลเลอร์หมายเลข 2, 8259A
0C0-0DF	ดีเอ็มเอคอนโทรลเลอร์หมายเลข 2, 8237A-5
0F0	เคลียร์โปรเซสเซอร์คณิตศาสตร์
0F1	รีเซตโปรเซสเซอร์คณิตศาสตร์
0F8-0FF	โปรเซสเซอร์คณิตศาสตร์
1F0-1F8	ฮาร์ดดิสก์
200-207	เกมไอโอ
278-27F	พอร์ตเครื่องพิมพ์หมายเลข 2
2F8-2FF	พอร์ตอนุกรมหมายเลข 2
300-31F	โปรโตไทป์การ์ด
360-36F	สำรอง
378-37F	พอร์ตเครื่องพิมพ์หมายเลข 1
380-38F	SDLC, ไบซิงค์ 2
3A0-3AF	ไบซิงค์ 1
3B0-3BF	โมโนโครมและเครื่องพิมพ์
3C0-3CF	สำรอง
3D0-3DF	จอภาพสี
3F0-3F7	ควบคุมดิสเกตต์
3F8-3FF	พอร์ตอนุกรมหมายเลข 1

ตารางที่ 4.1 หมายเลขพอร์ตที่ใช้กับ IBM PC AT

(ยืน ภู่วรวรรณ 2533)

DMA ตัวแรก	หน้าที่
ช่อง 0	สงวนไว้(แต่เดิมบนเครื่อง XT ใช้สำหรับรีเฟรช)
ช่อง 1	ใช้สำหรับการส่งผ่านกับ SDLC
ช่อง 2	ใช้สำหรับการติดต่อรับส่งข้อมูลกับดิสเก็ต
ช่อง 3	สงวนไว้
DMA ตัวที่สอง	
ช่อง 4	ต่อร่วมกับชิป 8237 ตัวแรก
ช่อง 5	สงวนไว้
ช่อง 6	สงวนไว้
ช่อง 7	สงวนไว้

ตารางที่ 4.2 รายละเอียดช่อง ดีเอ็มเอ ของเครื่อง IBM PC AT

(ยืน ภู่วรวรรณ 2533)

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

เติมเข้ามาใหม่ในรุ่น AT นั้นสามารถส่งข้อมูลได้ครั้งละ 16 บิต และช่องที่ยังว่างอยู่ก็คือ ช่อง 5-7 เพราะฉะนั้นจึงเลือกช่องที่ 5 มาใช้ในการออกแบบครั้งนี้

ข้อที่ต้องคำนึงถึงในการทำ ดีเอ็มเอ บนระบัสของเครื่อง AT ก็คือ การรีเฟรชหน่วยความจำไดนามิคแรมของระบบ ที่ต้องทำทุกๆ 15 μsec ซึ่งในการทำการรีเฟรชจะต้องมีการใช้บัสของระบบ แต่ในขณะที่มีการทำ ดีเอ็มเอนั้นบัสจะถูกยึดครองไปหมด เพราะฉะนั้นจะต้องมีขบวนการที่ทำให้ ดีเอ็มเอ ปล่อยระบบบัสออกมา ทุกๆ 15 μsec เช่นกันในขณะทำ ดีเอ็มเอ ซึ่งการแก้ปัญหาที่ง่ายที่สุดทำโดยการเซตค่า burst size ของการทำ ดีเอ็มเอ เวลาในการส่งไม่ให้เกินเวลาที่กำหนดในการทำรีเฟรชหน่วยความจำ แต่วิธีการแบบนี้จะทำให้เสียเวลามากถ้ามีข้อมูลที่ต้องส่งผ่าน ดีเอ็มเอ มากๆ เพราะจะต้องมีการจัดเตรียมการทำ ดีเอ็มเอ ใหม่ทุกครั้งหลังจากปล่อยบัสไปแล้ว ในการออกแบบจึงไม่ใช้วิธีการนี้ แต่ใช้การควบคุมโดยฮาร์ดแวร์แทน

4.4.4 การออกแบบระดับการเรียกอินเตอร์รัปต์

ในเครื่อง IBM PC AT มีไอซีเบอร์ 8259A เป็นตัวควบคุมการทำอินเตอร์รัปต์ จำนวนสองตัวเพิ่มจาก รุ่น XT ซึ่งมีเพียงแค่ตัวเดียว ทำให้สามารถมีการร้องขอการทำอินเตอร์รัปต์ทางฮาร์ดแวร์ได้ถึง 16 ระดับ ซึ่งแต่ละระดับจะมีการจัดลำดับความสำคัญไว้และเนื่องจากตัว 8259A ตัวที่สองนั้นต่อสัญญาณ INT เข้า IRQ2 ของ 8259A ตัวแรก เพราะฉะนั้นจึงทำให้ลำดับความสำคัญของ IRQ8-15 ที่เพิ่มเข้ามาอยู่ในระดับที่แทรกระหว่าง IRQ2 และ IRQ3 ซึ่งแสดงได้ตามตารางที่ 4.3

จากตารางจะเห็นว่าอินเตอร์รัปต์แต่ละหมายเลขมีการใช้งานอยู่แล้ว โดยส่วนใหญ่ ที่ว่างอยู่ก็มี IRQ9-IRQ12 และ IRQ15 การเลือก IRQ นี้จะมีผลในการทำงานอยู่ด้วยเพราะแต่ละหมายเลขตัวซีพียูจะให้ความสำคัญในการให้บริการไม่เท่ากัน ดังนั้นเราจึงควรเลือก IRQ หมายเลขที่ว่างอันดับแรกซึ่งก็คือ IRQ9 ซึ่งตามตารางจะเห็นว่าอินเตอร์รัปต์หมายเลขนี้จะเท่ากับ IRQ2 เดิมบนเครื่อง XT นั่นเอง แต่เพื่อความสะดวกในการเปลี่ยนแปลงหมายเลขอินเตอร์รัปต์ ในกรณีที่อาจไปซ้ำซ้อนกับอุปกรณ์เชื่อมต่ออื่นๆที่มีใช้อยู่แล้ว จึงได้ออกแบบให้เลือกหมายเลขอินเตอร์รัปต์ได้โดยใช้ตัวต่อวงจร (Jumper)

ลำดับความสำคัญ	หน้าที่
NMI	รับสัญญาณการตรวจสอบพาริตี้และตรวจสอบช่องไอโอ
IRQ0	ไทเมอร์เอาต์พุต 0
IRQ1	คีย์บอร์ด
IRQ2	อินเตอร์รัปต์มาจากตัวควบคุมตัวที่สอง
IRQ8	สัญญาณกำหนดเวลา
IRQ9	เรียกมาจาก IRQ2 เดิม
IRQ10	สงวนไว้
IRQ11	สงวนไว้
IRQ12	สงวนไว้
IRQ13	โปรเซสเซอร์ร่วม
IRQ14	ฮาร์ดดิสก์
IRQ15	สงวนไว้
IRQ3	พอร์ตอนุกรม 2
IRQ4	พอร์ตอนุกรม 1
IRQ5	จากแผงควบคุมฮาร์ดดิสก์
IRQ6	วงจรควบคุมดิสก์
IRQ7	พอร์ตขนาน 1

ตารางที่ 4.3 การวางอินเตอร์รัปต์ตามลำดับความสำคัญ

(ยืน ภู่วรวรรณ 2533)

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

เป็นตัวเสียบเพื่อให้เลือกเบอร์อินเตอร์รัปต์ โดยให้เลือก IRQ10 ,IRQ11, IRQ12 ได้เพิ่มเติมจาก IRQ2

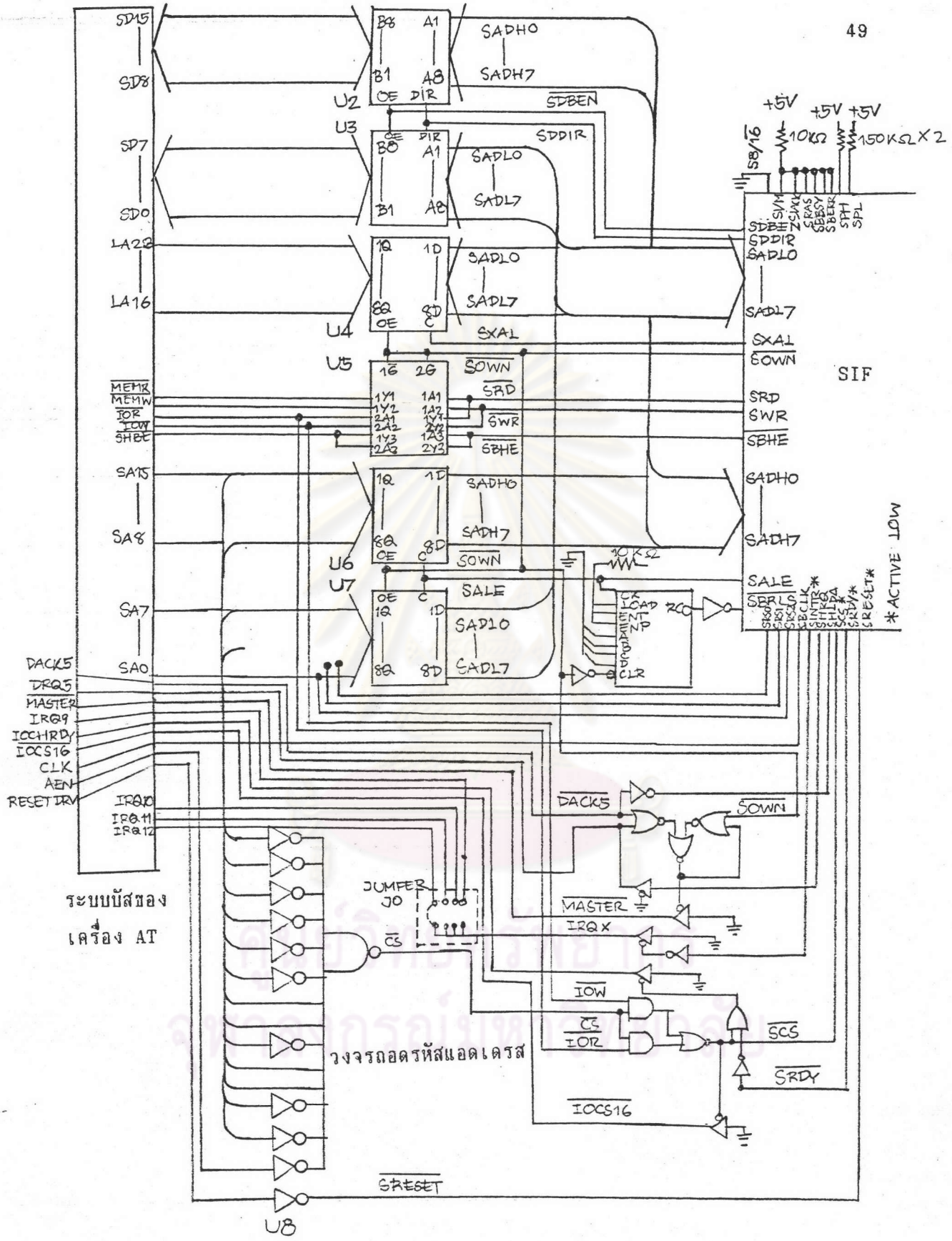
4.4.5 การออกแบบการจัดระบบบัส

เนื่องจากระบบบัสข้อมูลและบัสแอดเดรสของ TMS380 ที่จะเชื่อมต่อกับระบบบัสของเครื่อง AT นั้นมีความแตกต่างกันอย่างมากอยู่ 2 ประการคือ การจัดลำดับนัยสำคัญของบิตของสองระบบจะสลับกันคือ บิตที่มีนัยสำคัญสูงสุดของ AT บัสจะกลายเป็นบิตที่นัยสำคัญต่ำสุดของ TMS380 บัส เพราะฉะนั้นในการเชื่อมต่อจึงต้องสลับบิตกันโดยที่บิตที่มีนัยสำคัญสูงสุดของ AT บัสจะไปต่อกับบิตที่มีนัยสำคัญต่ำสุดของ TMS380 บัส และสิ่งที่แตกต่างที่สำคัญอีกประการหนึ่งก็คือ เครื่อง AT จะมีระบบบัสของแอดเดรสและ ข้อมูลแยกกันโดยเด็ดขาด คือ 24 บิต และ 16 บิต แต่ว่าบัสของ TMS380 จะมีการมัลติเพล็กซ์สัญญาณระหว่าง สัญญาณแอดเดรสและข้อมูลโดยใช้บัสร่วมกัน คือ SADL0-SADL7 และ SADH0-SADH7 ดังนั้นจึงต้องมีการควบคุมในส่วนนี้โดยใช้ไอซีที่ทำหน้าที่ เป็น แอดเดรสแลตช์ และ ดาต้าทรานซีฟเวอร์ โดยใช้สัญญาณ SALE และ SXAL เป็นตัวควบคุม สำหรับสัญญาณนาฬิกาที่ใช้ควบคุมการทำงานของ TMS38030 ซิสเต็มอินเตอร์เฟสจะใช้สัญญาณที่มาจากเครื่อง AT คือสัญญาณ CLK ความเร็ว 6 MHz

4.5 รายละเอียดวงจรการอินเตอร์เฟสตัวการ์ดกับระบบบัสของเครื่อง PC AT

4.5.1 ระบบบัสดีมัลติเพล็กซิ่ง (Bus Demultiplexing)

เพราะว่าระบบบัสของเครื่องรุ่น AT นั้นจะยอมอนุญาติให้ระบบมาสเตอร์บัสถูกยึดครองโดยเอ็กแพนชันการ์ดได้ TMS38030 จึงสามารถส่งผ่านข้อมูลจากเน็ตเวิร์คได้โดยตรงทั้งเข้าและออกจากหน่วยความจำของพีซี แต่อย่างไรก็ตามชิสเต็มบัสของ PC AT จะเป็นระบบ นอนมัลติเพล็กซ์ (Non-multiplex) นั่นก็คือดาต้าและแอดเดรสบัสจะแยกจากกัน ดังนั้นระบบบัสอินเตอร์เฟสของ TMS38030 จึงต้องถูก ดีมัลติเพล็กซ์ด้วย ตามรูป 4.3 จะเห็นว่าชุดไอซีเบอร์ 74LS373 (แอดเดรสแลตช์) และ 74LS245(ดาต้าบัสทรานซีฟเวอร์)จะเป็นตัวที่ทำหน้าที่ ดีมัลติเพล็กซ์ให้กับบัสอินเตอร์เฟสของ TMS38030



รูปที่ 4.3 วงจรอินเตอร์เฟส PC AT บัสกับชิป TMS38030

การทำแอดเดรสแลตช์นั้นทำโดยการที่ TMS38030 ส่งสัญญาณ SXAL (Extended Address Latch) และ SALE (Address Latch Enable) ออกมา และ TMS38030จะส่งสัญญาณ $\overline{\text{SOWN}}$ (System Bus Owned),SSDIR(Bus Direction) และ $\overline{\text{SDBEN}}$(Bus Enable) เพื่อจะทำการควบคุม และ บัฟเฟอร์ข้อมูล โดยใช้วงจรลอจิกภายนอกให้น้อยที่สุด

4.5.2 อินพุต/เอาต์พุตโดยตรง (Direct Input/Output)

TMS38030 จะบรรจุรีจิสเตอร์ขนาด 16 บิต 4 ตัวเพื่อใช้ในการทำขบวนการเริ่มต้น และ การควบคุมอินเตอร์รัปต์ และรีจิสเตอร์เหล่านี้จะถูกแมป(Map) ไปยัง I/O แอดเดรสสเปซ ของซีพียูของ เครื่องพีซีและจะถูกแอคเซสโดยการโปรแกรมแบบ DIO (Direct I/O control) DIO ไซเคิลจะเริ่มเมื่อซีพียูให้สัญญาณแอดเดรสออกมาที่แอดเดรสบัส SA15-SA0 และส่ง Address Enable(AEN) ออกมาด้วย สายแอดเดรส SA15-SA3 และ AEN จะถูกถอดรหัสโดย U8 ซึ่งเป็น PAL (Programmable Array Logic) จะให้สัญญาณ $\overline{\text{CS}}$ ออกมา ส่วนสาย SA0-SA2 จะต่อกับ SRS2, SRS1 และ SRS0 ซึ่งจะเป็นตัวเลือกรีจิสเตอร์ของ TMS38030 ที่ต้องการ สัญญาณ $\overline{\text{CS}}$ จะถูกควบคุมโดย $\overline{\text{IOR}}$ และ $\overline{\text{IOW}}$ ซึ่งเป็นสัญญาณการควบคุมการเขียนและอ่านของโปรเซสเซอร์ จะผลิตสัญญาณ SCS ให้แก่ TMS38030 $\overline{\text{IOR}}$ และ $\overline{\text{IOW}}$ จะวิ่งไปยัง U5 ที่ $\overline{\text{SRD}}$ และ $\overline{\text{SWR}}$ เพื่อจะเป็นตัวแยกแยะว่าเป็น DIO read หรือ DIO write และจะส่งสัญญาณไปที่ $\overline{\text{IOCS16}}$ ของชิสเต็มอินเตอร์เฟสด้วย เพื่อจะบอกโฮสต์ชิสเต็มบัสว่าขณะนี้กำลังทำการส่งผ่านข้อมูล 16 บิตอยู่ และในขณะที่ TMS38030 รับทราบการทำ DIO cycle มันจะแทรกสัญญาณทั้ง SDBEN และ SDDIR ไปที่ ดาต้าทรานซีฟเวอร์ U2 และ U3 เพื่อให้มันทำงานและทำการกำหนดทิศทางในการส่งข้อมูล และเมื่อการส่งข้อมูลเสร็จเรียบร้อยแล้วTMS38030 ก็จะดึง $\overline{\text{SRDY}}$ เป็นโลว์ (Low)เพื่อจะบอกว่าการทำงานเสร็จแล้ว ถ้าสัญญาณ $\overline{\text{SRDY}}$เป็นไฮ (High) มันจะเท่ากับบอกชิสเต็มบัสว่า "not ready" ซึ่งมันจะทำให้ IOCHRDY เป็นโลว์และถ้า $\overline{\text{SRDY}}$ เป็นโลว์จะทำให้ IOCHRDY เป็นไฮ ซึ่งจะเท่ากับ "ready" เมื่อถึงจุดนี้ DIO cycle ก็จะสมบูรณ์และ

โฮสต์โปรเซสเซอร์จะดึงสัญญาณควบคุมของมันออกทั้งหมด

4.5.3 การเรียกแอดเดรสหน่วยความจำโดยตรงหรือดีเอ็มเอ

การทำ ดีเอ็มเอ จะทำได้โดยการที่ TMS38030 ทำการควบคุมโฮสต์ชิสเต็มบัส TMS38030 จะร้องขอการควบคุมบัสโดยการส่งสัญญาณให้ SHRQ เป็นไฮ ซึ่งจะส่งผลให้ DRQ5 ของชิสเต็มบัสถูกกระตุ้น ซีพียูจะรู้ว่าเป็นการร้องขอการทำ ดีเอ็มเอ ดังนั้นมันจะตอบกลับโดยส่ง $\overline{\text{DACK5}}$ ตอบกลับและถูกอินเวิร์ตโดย U9 ซึ่งเป็น PAL ให้สัญญาณส่งไปยัง SHLDA ของ TMS38030 และ TMS38030 จะตอบสนองโดยส่ง $\overline{\text{SOWN}}$ ออกมาเป็นโลว์ ซึ่งจะเท่ากับการแสดงความเป็นเจ้าของบัส และสัญญาณนี้จะไปเปิดตัวแลทช์ U4, U6, U7 และจะมีผลไปถึงการผลิตสัญญาณ $\overline{\text{MASTER}}$ ออกมาเพื่อส่งไปยังชิสเต็มบัสเพื่อบอกว่า TMS38030 จะควบคุมการทำงานของโฮสต์ชิสเต็มบัสแล้ว นอกจากนี้สัญญาณ $\overline{\text{SOWN}}$ จะต่อกับ U5 เพื่อบอกทิศทางการทำ ดีเอ็มเอ ในขณะที่ทำการควบคุมบัสอยู่นั้น TMS38030 จะส่ง SDDIR ให้ U2 และ U3 เพื่อเป็นการบอกทิศทางในการส่งข้อมูล ในการทำ ดีเอ็มเอ ไซเคิล รอบแรกนั้น TMS38030 จะทำให้แอดเดรสไบต์สูงสุดถูกแลตช์โดยส่ง SXAL ไปยัง U4 ส่วนไบต์กลางและไบต์ต่ำสุดจะถูกแลตช์โดยสัญญาณ SALE ที่ส่งเข้าไปใน U6 และ U7 TMS38030 จะส่ง $\overline{\text{SRD}}$ หรือ $\overline{\text{SWR}}$ ขึ้นอยู่กับตอนนั้นจะทำการเขียนหรืออ่าน สัญญาณทั้งสองนี้จะถูกแมปไปยัง $\overline{\text{MEMR}}$, $\overline{\text{MEMW}}$ โดยผ่านทาง U5 จากนั้น TMS38030 จะส่ง $\overline{\text{SDBEN}}$ เพื่ออีนาเบิ้ลดาต้าทรานซีฟเวอร์ U2 และ U3 ไซเคิลจะจบลงตามปรกติเว้นเสียแต่ว่า จะมีการส่ง wait state แทรกเข้ามาโดยอุปกรณ์ สเลฟ (Slave) ซึ่งจะเซต IOCHRDY เป็นโลว์ และเพื่อให้แน่ใจว่าไซเคิลจะทำงานเหมือนๆกัน แอดเดรสไบต์สูงสุดจะไม่เปลี่ยนแปลงสถานะเว้นเสียแต่ว่ามีการปัดตัวทดจากไบต์ต่ำสุดขึ้นมาไบต์กลาง 74LS161A (U11) ซึ่งเป็นชิงโครนัสเคาน์เตอร์ขนาด 4 บิตจะเป็นกลไกทางฮาร์ดแวร์ที่จะจำกัดขนาด burst size ของ ดีเอ็มเอ ให้ได้แค่ 15 และมันจำเป็นในการที่จะทำให้ แรม ของ PC AT มีการรีเฟรช (Refresh) เป็นจังหวะซึ่งเมื่อเคาน์เตอร์นับ SALE ได้ 15 ลูกมันก็จะส่ง $\overline{\text{SBRLS}}$ ออกมา เมื่อ $\overline{\text{SBRLS}}$ ถูกส่งออกมา TMS38030 ก็จะส่งคืนการควบคุมพีซีบัส แล้วส่ง $\overline{\text{SOWN}}$ เป็นไฮ ตัว

เคาน์เตอร์ก็จะรีเซต ต่อจากนั้น TMS38030 ก็จะทำการร้องขอใช้บัสโดยทันทีทันใด(ส่ง SHRQ)

การเลิกใช้โฮสต์ชิสเต็มบัสนั้น TMS38030 จะทำให้ SHRQ เป็นโลว์ ทำสัญญาณ I/O เป็น ไตร-สเตท(Tri-state) แล้วปล่อยสัญญาณ $\overline{\text{SOWN}}$ ซึ่งมันจะทำการปล่อยสัญญาณ DRQ5 และ $\overline{\text{MASTER}}$ ของโฮสต์ชิสเต็มบัสเพื่อแสดงว่าตอนนี้บัสนั้นว่าง

4.6 รายละเอียดการออกแบบวงจรริงอินเตอร์เฟส

การอินเตอร์เฟสระหว่างอินเตอร์เฟสการ์ดกับสายตีเกลียวคู่จะผ่านวงจรริงอินเตอร์เฟส (RI) ซึ่งจะประกอบด้วยชิป MSI สองตัวคือ TMS38051 และ TMS38052 ซึ่งภายในชิปจะประกอบด้วยส่วนประกอบย่อยๆหลายส่วนด้วยกัน

4.6.1 ส่วนประกอบของ TMS38051 และ TMS38052

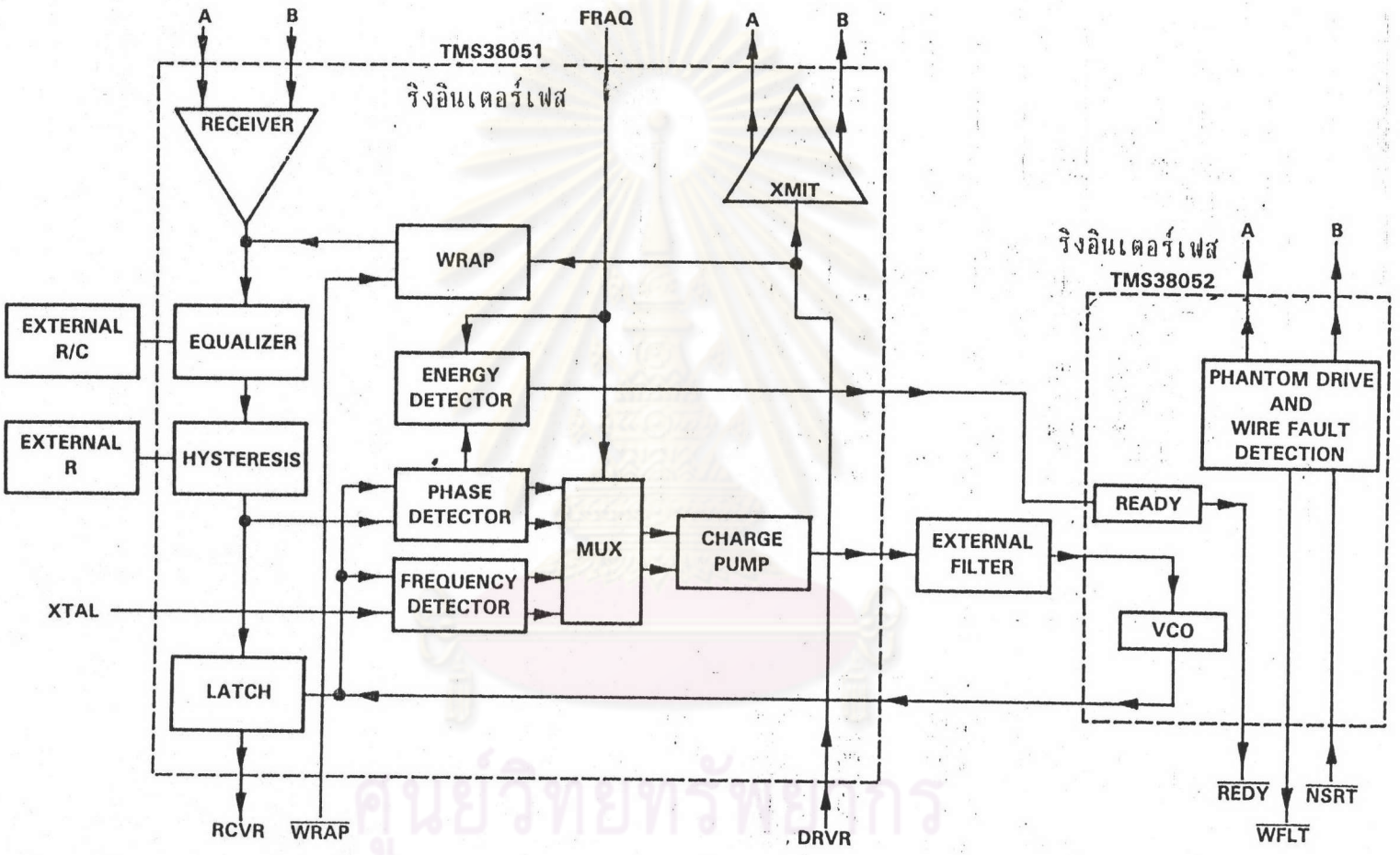
ชิปเซตทั้งสองตัวนี้เป็นชิปแบบอนาล็อก ซึ่งจะทำหน้าที่ให้การอินเตอร์เฟสที่ตรงตามมาตรฐาน IEEE802.5 เพื่อจะรับและส่งข้อมูลระหว่าง TMS38020 โพรโตคอลแฮนเดลอร์ กับ ระบบสายเคเบิ้ลของโทเกนริง ชิปเบอร์ TMS38051 จะเป็นทรานซีฟเวอร์ที่จะมีฟังก์ชันการรับและส่ง ในขณะที่ชิปเบอร์ TMS38052 จะเป็นตัวควบคุมซึ่งจะมีฟังก์ชันในการควบคุมริงอินเตอร์เฟส

4.6.2 วงจรวอทช์ด็อกไทเมอร์ (Watchdog Timer)

ไอซีเบอร์ 74LS122 เป็นวงจรโมโนสเตเบิ้ลมัลติไวเบรเตอร์ (Monostable-multivibrator) ซึ่งจะทำหน้าที่เหมือนเป็นสุนัขเฝ้าระวังภัยเวลามีอะไรเกิดขึ้นในวงจรมันก็จะทำการเตือนออกมา โดยที่มันจะให้พัลซ์ขนาด 30 ms เมื่อสัญญาณ $\overline{\text{PHNSRT}}$ ถูกทำให้เป็นโลว์ เอาต์พุตที่ออกที่ Q จะเป็นสัญญาณ $\overline{\text{NSRT}}$ หรืออินเสิร์ท (Insert) ส่งไปยังริงอินเตอร์เฟสเมื่อ $\overline{\text{NSRT}}$ เป็นโลว์วงจรแฟนทอมไดร์วจะถูกกระตุ้นและทำให้สายเคเบิ้ลทำการต่อเข้ากับตัวรวมสาย คือต่อเข้ากับริงนั่นเอง แต่ถ้าสัญญาณนี้เป็นไฮมันก็จะถูกถอดออกจากริง ตัวโปรแกรมของอินเตอร์เฟสการ์ดจะทำการ เปิดปิด (Toggle) สัญญาณ $\overline{\text{PHNSRT}}$ ทุกๆ 10 ms เมื่อไรก็ตามที่ยังต้องการการอินเสิร์ทอยู่ ซึ่ง

ตารางที่ 4.4 สัญญาณที่ใช้ระหว่างโพรโตคอลแฮนเดลอร์และริงอินเตอร์เฟส

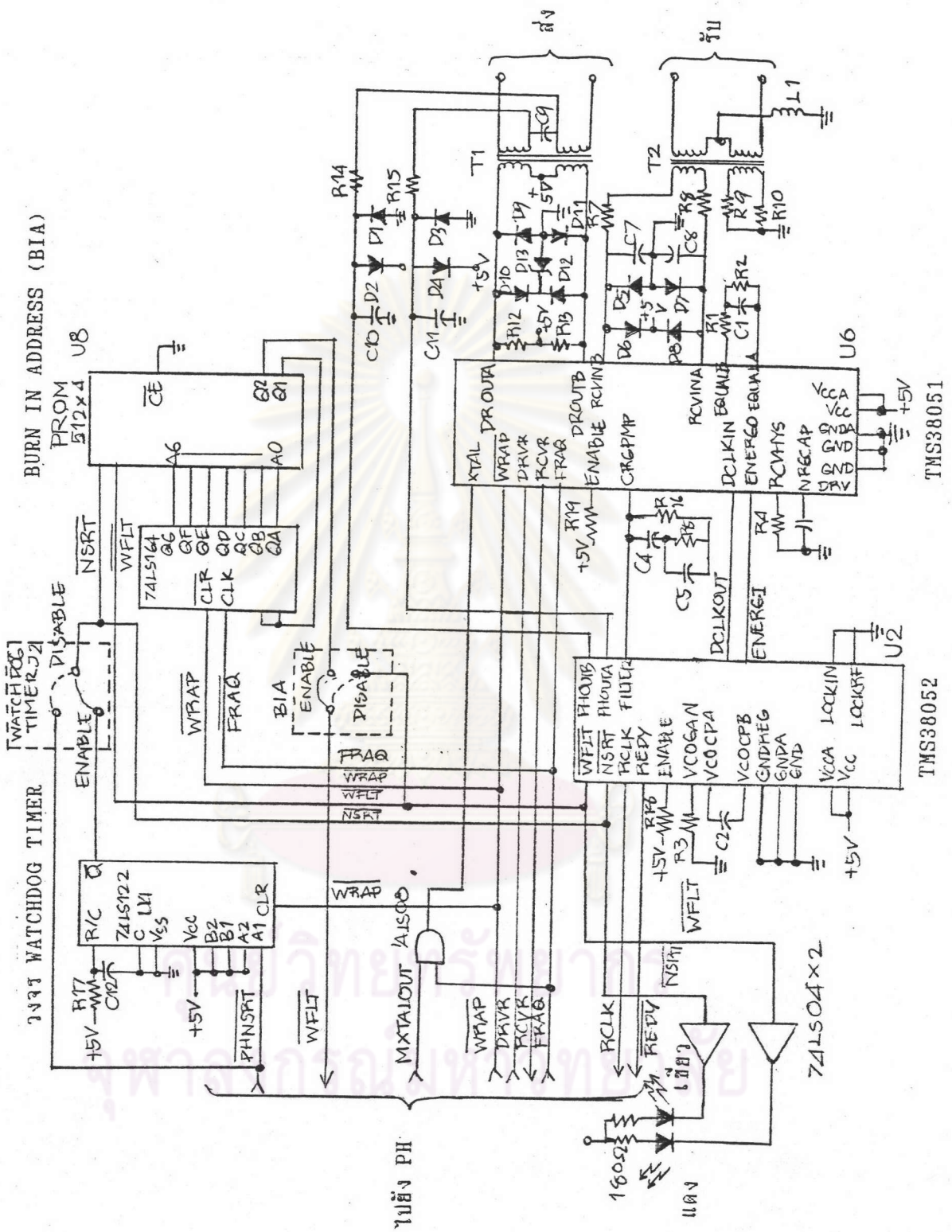
DRVR	เป็นข้อมูลขาออกของ PH ไปยังริงอินเตอร์เฟส ริงอินเตอร์เฟสจะทำการขยายกระแสให้สัญญาณข้อมูลที่จะส่ง
FRAQ	เมื่อสัญญาณนี้ถูกส่งมาโดย PH PLL จะล็อกสัญญาณกับ XTAL
$\overline{\text{NSRT}}$	เป็นอินพุตให้ริงอินเตอร์เฟส เมื่อเป็นโลว์มันจะทำให้วงจรแฟนทอมไดร์วทำการแทรกทางกายภาพเข้ามา สัญญาณนี้จะถูกขับโดยผ่านทางวงจร วอทช์ด็อกไทเมอร์
RCLK	เป็นเอาต์พุตจาก VCO ของ TMS 38052 จะให้สัญญาณนาฬิกาทำการสุ่มข้อมูลซึ่งมีเฟสล็อกกับข้อมูลที่เข้ามา
RCVR	สัญญาณนี้จะให้ข้อมูลเข้ารหัสดิฟเฟอร์เรนเชียลแมนเชสเตอร์จากริงในขณะที่ถูกตรวจจับโดย TMS38051
$\overline{\text{REDY}}$	มันจะเป็นโลว์ถ้า แรงขับสัญญาณถูกตรวจสอบได้ว่าต่ำสุดจากคู่สายอินพุต หรือ PLL ล็อกสัญญาณอินพุต
$\overline{\text{WFLT}}$	จะเป็นโลว์ถ้ามันตรวจสอบว่ามีการลัดวงจรลงกราวด์ของคู่สายรับหรือส่ง หรือมีการเปิดวงจรคู่รับหรือคู่ส่งคู่ใดคู่หนึ่ง
WRAP	ถ้าถูกทำให้เป็นโลว์มันจะบังคับให้ริงอินเตอร์เฟสไปกระตุ้นให้ป้อนกลับ (Feedback) จากสัญญาณส่งไปสัญญาณรับ ทั้งนี้เพื่อใช้ทำการวินิจฉัยเมื่อทำการรีเซตอะแดปเตอร์



รูปที่ 4.4 บล็อกไดอะแกรมของวงจรริงอินเตอร์เฟส

(TMS380 Adapter Chipset User's Guide 1986)

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 4.5 การต่อวงจรกันระหว่างชิปเซตส่วนวงจรริงอินเตอร์เฟส

มันจะทำให้ NSRT เป็นโลว์อยู่ตลอด ถ้าเงื่อนไขยังคงที่จะป้องกันโปรเซสเซอร์สื่อสาร จากการเปิดปิดสัญญาณ PHNSRT วงจรวันชอต (One-shot) ก็จะหมดกำหนดเวลา(Time -out) และอินเตอร์เฟสการ์ดก็จะถูกถอดจากริง

วงจรนี้มีเพื่อเพิ่มความเชื่อถือได้ของริง โดยจะทำให้หลีกเลี่ยงในการที่จะเกิดปัญหาในเรื่องการบกพร่องของสัญญาณนาฬิกาซึ่งจะทำให้ริง เกิดการสะดุดลงได้เมื่อมีความผิดพลาดเกิดขึ้นเช่น TMS38010 โปรเซสเซอร์สื่อสารไม่สามารถรักษาสัญญาณ PHNSRT ไว้ได้ อินเตอร์เฟสการ์ดก็จะถูกดึงออกจากริงโดยอัตโนมัติ

4.6.3 สัญญาณ TMS38020โพรโตคอลแฮนเดลอร์ไปยังริงอินเตอร์เฟส

ตารางที่ 4.4 จะอธิบายถึงสัญญาณที่ใช้ระหว่างโพรโตคอลแฮนเดลอร์และวงจรริงอินเตอร์เฟส

4.6.4 ริงอินเตอร์เฟสทรานสมิตเตอร์ (Ring Interface Transmitter)

ฟังก์ชันในการส่งได้ถูกรวมอยู่ใน TMS38051 จะแปลงสัญญาณ TTL ซึ่งทำการเข้ารหัสแบบดิฟเฟอร์เรนเชี่ยล บนขาดาต้าอินพุต (DRVR) ไปเป็นสัญญาณดิฟเฟอร์เรนเชี่ยลเคอร์เร้นท์ที่จะส่งไปในสายตีเกลียวคู่ซึ่งมี คาร์แรคเตอร์ริสติคอิมพีแดนซ์ 150 โอห์ม ตัวทรานสมิตเตอร์นี้จะทำการตอบสนองสัญญาณ DRVR โดยการจะควบคุมการปล่อยกระแสจากแทปกลางของทรานสฟอร์เมอร์(Transformer) ว่าจะป้อนออกทาง DROUTA หรือ DROUTB เมื่อ DRVR เป็นไฮกระแสจะทำการป้อนทาง DROUTA ถ้าเป็นโลว์จะออกทาง DROUTB ทั้ง DROUTA และ DROUTB นี้เป็นเอาท์พุทแบบคอลเลคเตอร์เปิด ซึ่งกระแสที่ควบคุมจะอยู่ในช่วง 25 ma +-5 ma วงจรที่เทียบเท่ากันนั้นแสดงดังรูปที่ 4.6 โดยจะแทน DROUTA และ DROUTB โดยรีซิสเตอร์(Resistor) 75 โอห์มที่ต่อกับ Vcc

4.6.5 ริงอินเตอร์เฟสรีซีฟเวอร์ (Ring Interface Receiver)

ช่องรับสัญญาณของ TMS38051 จะมีฟังก์ชันในการทำ equalization, signal shaping และ retiming received data TMS38051

จะต่อทรานฟอร์เมอร์กับตัวกลางในการส่งเรียกว่าการต่อแบบทรานฟอร์เมอร์คัปเปิ้ล ทั้งนี้ก็เพื่อจะตัด DC อ็อฟเซตในสัญญาณดิฟเฟอร์เรนเชี่ยลทิ้งไป วงจรรับใน TMS38051 จะมีการไบแอส(Bias) ตัวดิฟเฟอร์เรนเชี่ยลอินพุต RCVINA และ RCVINB แยกกันเป็น Vcc/2มีคอมมอนโหมดรีเจ็กชันขนาด 40dB ให้กับการส่งข้อมูลด้วยความเร็ว 4 Mbit/s บนสายตีเกลียวคู่ ส่วนวงจรกรองความถี่ต่ำได้ต่อคั่นระหว่างทรานสฟอร์เมอร์และ TMS38051 เพื่อจะลดทอนสัญญาณรบกวนความถี่สูง

4.6.6 ริงอินเตอร์เฟสเฟสล็อกลูป (Ring Interface Phase Lock Loop)

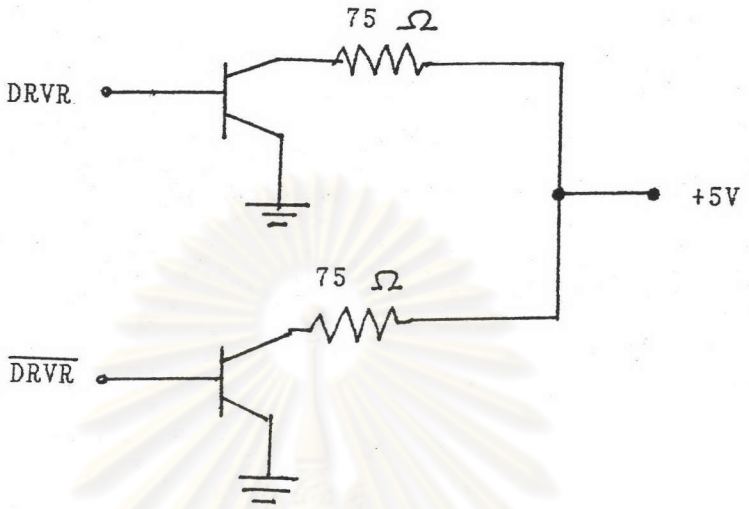
การรีโคเวอรี่สัญญาณ data clock 8 MHz ที่ฝังอยู่ในสัญญาณดิฟเฟอร์เรนเชี่ยลเมนเชสเตอร์ของข้อมูล จะทำได้โดยการใช้วงจรเฟสล็อกลูป(PLL) ซึ่งมันจะฝังอยู่ในตัว TMS38051 และ TMS38052 ตัวลูปฟิลเตอร์จะเป็นเพียงฟังก์ชันของ PLL ซึ่งจะเป็นส่วนที่แยกออกมาโดยเฉพาะ แม้ว่าทรานเฟอร์ฟังก์ชันของโวลเตจคอนโทรลออสซิเลเตอร์จะถูกเซตโดยส่วนอื่นที่แยกออกมาก็ตาม ตามรูปที่ 4.7 จะแสดงการต่อของวงจรเฟสล็อกลูปแบบ ออร์เดอร์ที่ 2 โดยมีแถบความถี่กว้างขนาด 97.6 KHz และลูปแดมปิ้งแฟคเตอร์มีค่า 25.7

4.6.7 วงจรตรวจสอบพลังงาน (Energy Detect)

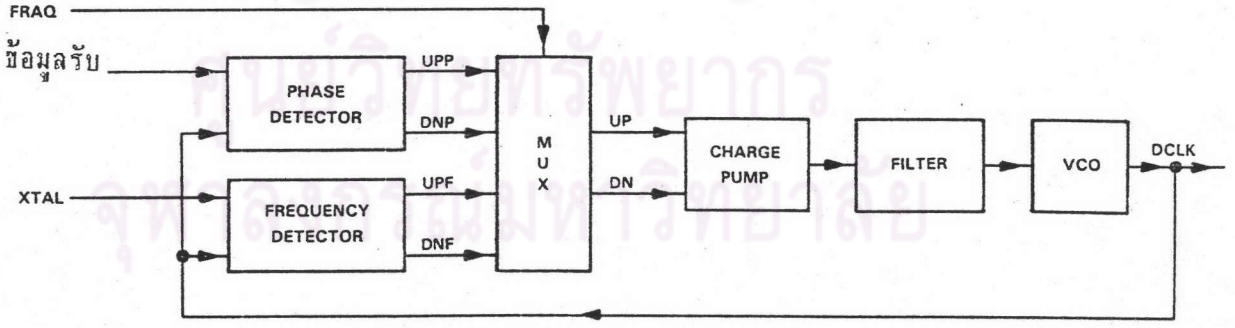
วงจรนี้จะทำหน้าที่ตรวจสอบการปรากฏของ สัญญาณที่ช่องรับอินพุต เวลาที่มีตรวจขอบขาขึ้นของสัญญาณได้ข้อมูลอินพุต ก็จะมีการชาร์จประจุเข้าไปในตัวคาปาซิเตอร์(C3 ที่ต่อกับ NRGCAP ของ TMS38052) และถ้ามีการตรวจไม่พบสัญญาณนี้ในข้อมูลที่ได้รับจะทำให้ประจุรั่วซึม (Leak) ออกมาจากตัวคาปาซิเตอร์ตัวนี้

4.6.8 แหรบโหมด (Wrap Mode)

TMS38051 จะมีฟังก์ชันแหรบ (Wrap) เพื่อทำการวินิจฉัยวงจรซึ่งจะกระตุ้นโดยทารให้สัญญาณโลว์ที่ขา $\overline{\text{WRAP}}$ วงจรนี้จะส่งข้อมูลเข้าที่ช่องรับแทนการส่งไปที่วงจรขับของสายตีเกลียวคู่ ไลน์ไดร์เวอร์จะถูกดิสเอเบิล(Disable)จึงไม่มีดิฟเฟอร์เรนเชี่ยลเอาต์พุตที่ DROUTA และ DROUTB และส่วนวงจรการรับของสายตีเกลียวคู่จะถูก



รูปที่ 4.6 ไลน์ไดร์เวอร์ของ TMS38051



รูปที่ 4.7 บล็อกไดอะแกรมของวงจรเฟสล็อกลูป

(TMS380 Adapter Chipset User's Guide 1986)

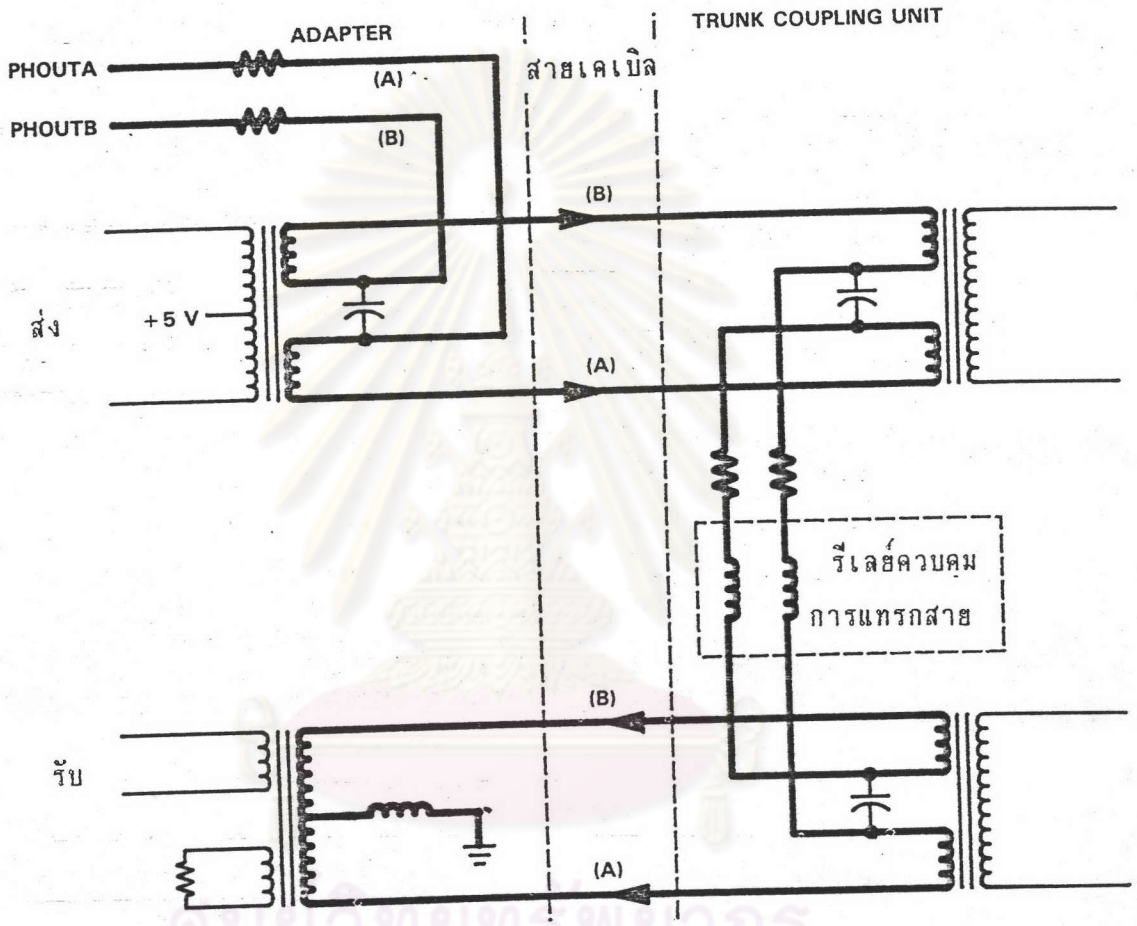
ดิสเอเบิ้ลโดยการดึงการไบแอสบนขา RCVINA และ RCVINB แหรบโหมดจะไม่มีผลกระทบต่อฟังก์ชันอื่นๆของช่องการรับ ในขณะที่ TMS38051 อยู่ในโหมดปรกตินั้นสัญญาณ $\overline{\text{WRAP}}$ จะเป็นไฮอยู่

4.6.9 ริงอินเตอร์เฟสแฟนทอมไดร์ว (Ring Interface Phantom Drive)

วงจรแฟนทอมไดร์วนี้เป็นวงจรที่อยู่ในชิป TMS38052 จะใช้สำหรับการกดสัญญาณดีซีบนอุปกรณ์ในการต่อวงแหวน และคอยตรวจสอบว่ามันเปิดหรือปิดวงจรอยู่ คำว่าแฟนทอมไดร์วนี้ ได้มาจากการที่สัญญาณดีซีนั้นหายไปในการทำดิฟเฟอร์เรนเชี่ยลโวลเตจซึ่งใช้ในการส่งข้อมูล วงจรนี้จะทำงานโดยการป้อนสัญญาณ $\overline{\text{NSRT}}$ เข้าไป โดยสัญญาณนี้จะถูกขับโดยวงจรวอชท์ด็อกไทเมอร์ เมื่อป้อนเข้าไปแล้วก็จะมีดีซีโวลเตจอยู่บนขา PHOUTA และ PHOUTB ผลของกระแสดีซีได้ไหลไปตามสายทวิสต์แพร์ไปยังไวริ่งคอนเซ็นเตรเตอร์ ซึ่งจะไปกระตุ้นรีเลย์ให้สวิตช์อินเตอร์เฟสการ์ดไปต่อกับริงกระแสก็จะวิ่งกลับลงกราวด์ ผ่านตัวอินดัคเตอร์(Inductor)ที่ต่อกับแทปกลางของทรานสฟอร์เมอร์ด้านปฐมภูมิ

เงื่อนไขการผิดปรกติของสายนั้น จะถูกตรวจจับโดยกระแสที่ป้อนโดย PHOUTA และ PHOUTB แลกรณีที่มีกระแสสูงหรือต่ำผิดปรกติจากดาต้าชีตของ TMS38051 /TMS38052 มันจะถูกตรวจจับโดยดูว่ามีสัญญาณ $\overline{\text{WFLT}}$ จากแฟนทอมไดร์วหรือไม่

สิ่งที่กำหนดในการออกแบบอย่างหนึ่งคือ การให้มีไฟแสดงสถานะว่าขณะนั้นแผ่นวงจรต่อเข้าไปในริงหรือเปล่า ซึ่งในสภาพปรกติการ เราไม่สามารถทราบได้เลยถ้าไม่เรียกตรวจสอบดูโดยทางซอฟต์แวร์ หลักการที่ทำก็คือเพิ่มเติมวงจรขับ LED ต่อกับ สายสัญญาณ $\overline{\text{NSRT}}$ ที่เข้าไปยัง TMS38052 ซึ่งสัญญาณนี้เป็นไฮเมื่อไม่มีการแทรกสายเข้าสู่ริง และเป็นโลว์เมื่อมีการแทรกสายเข้าสู่ริง ดังนั้นเมื่อต่อกับ LED ที่อีกข้างหนึ่งต่อกับไฟ +5 V จึงทำให้ไฟ LED สว่างเมื่อสัญญาณ $\overline{\text{NSRT}}$ เป็นโลว์ซึ่งแสดงว่ามีการแทรกสายเข้าสู่ริง และไฟ LED จะดับเมื่อสัญญาณ $\overline{\text{NSRT}}$ เป็นไฮเพื่อแสดงว่าสายเคเบิลไม่ได้ต่อกับริง



รูปที่ 4.8 วงจรแฟนทอมไดร์ว

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

4.7 การต่อวงจรภายในบอร์ดระหว่างชิปเซต

4.7.1 จังหวะนาฬิกาพื้นฐานของแลนอินเตอร์เฟสการ์ด

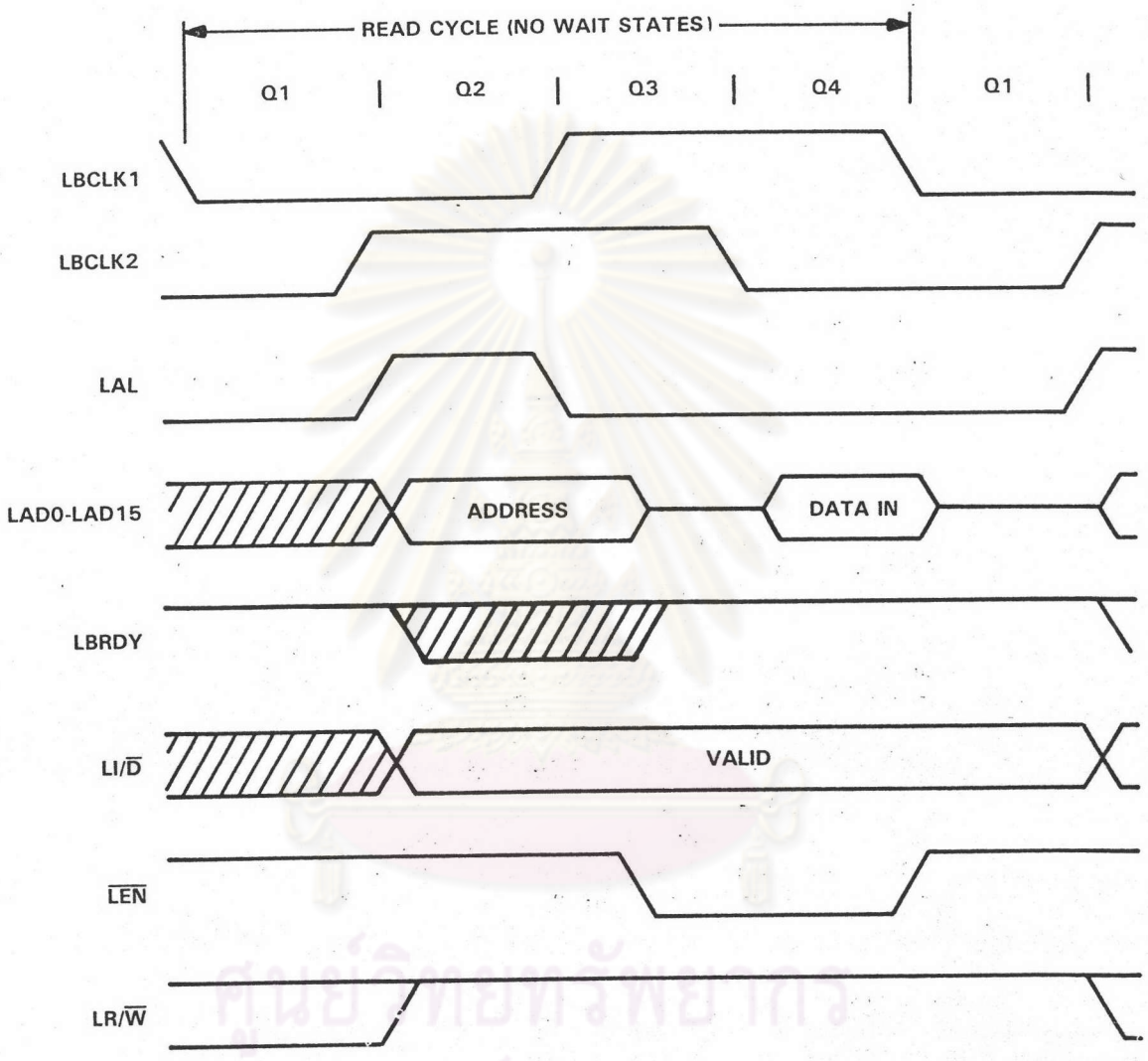
ตารางสัญญาณนาฬิกา (Timing Diagram)ของการส่งข้อมูลบนของอินเตอร์เฟสการ์ดบัสที่ทำงานโดยตัวโปรเซสเซอร์สื่อสาร ตามรูป 4.9-4.12 จะแสดงไซเคิลการเขียนและไซเคิลการอ่าน อุปกรณ์ภายนอกที่ต่ออยู่สามารถเพิ่มช่วงเวลาของสัญญาณ LBCLK1/LBCK2 ซึ่งเรียกว่า "wait state" การทำให้สัญญาณ LBRDY เป็นโลว์ก่อนที่จะเริ่ม Q4 ก็จะทำให้มันขยายช่วงเวลาการส่งนี้ สัญญาณ LBRDY จะยังคงเป็นโลว์จนกระทั่งถึงขาลงของสัญญาณ LBCLK1 สัญญาณส่งผ่านข้อมูลของบัสนี้จะยืดไปจนกระทั่งบัสมาสเตอร์จะตรวจได้ว่าสัญญาณ LBRDY เป็นไฮในขณะสัญญาณ LBCLK2 เป็นขาลง

ทั้งตัวบัสมาสเตอร์และอุปกรณ์สเลฟบนบัส จะดูสัญญาณ LBRDY เมื่อตอนเริ่มต้นของ Q4 ของแต่ละไซเคิล ถ้า LBRDY เป็นไฮ อุปกรณ์ทั้งหมดจะสรุปว่าบัสไซเคิลต่อมาจะเป็นค่าแอดเดรสในช่วง Q2แต่ถ้า LBRDY เป็นโลว์มันก็จะถือว่าบัสจะมีการทำงานตามปรกติต่อไป

ในตอนเริ่มต้นในแต่ละไซเคิลไม่ว่าจะเป็นการเขียน หรือ การอ่าน สาย LR/W และ LI/D จะเป็นค่าที่ถูกต้อง ในเวลาขณะนี้ข้อมูลแอดเดรสจะปรากฏอยู่บนสาย LAD0-LAD15 ซึ่งจะควบคู่ไปกับพัลซ์ของสัญญาณ LAL สัญญาณขาลงของ LAL นั้นอุปกรณ์ภายนอกจะใช้ในการทำแลตช์แอดเดรส ตัวบัสสเลฟจะทำการแลตช์แอดเดรสและถอดรหัสออกมา ในขณะที่อยู่ในดาต้าไซเคิลตัวบัสมาสเตอร์จะส่งสัญญาณ $\overline{\text{LEN}}$ ซึ่งสัญญาณนี้จะไม่ใช้ในการทำการถอดรหัสแอดเดรสแต่ใช้เป็นตัว "output enable" ในขณะทำไซเคิลการอ่าน

4.7.2 ไซเคิลการอ่าน (Read cycle)

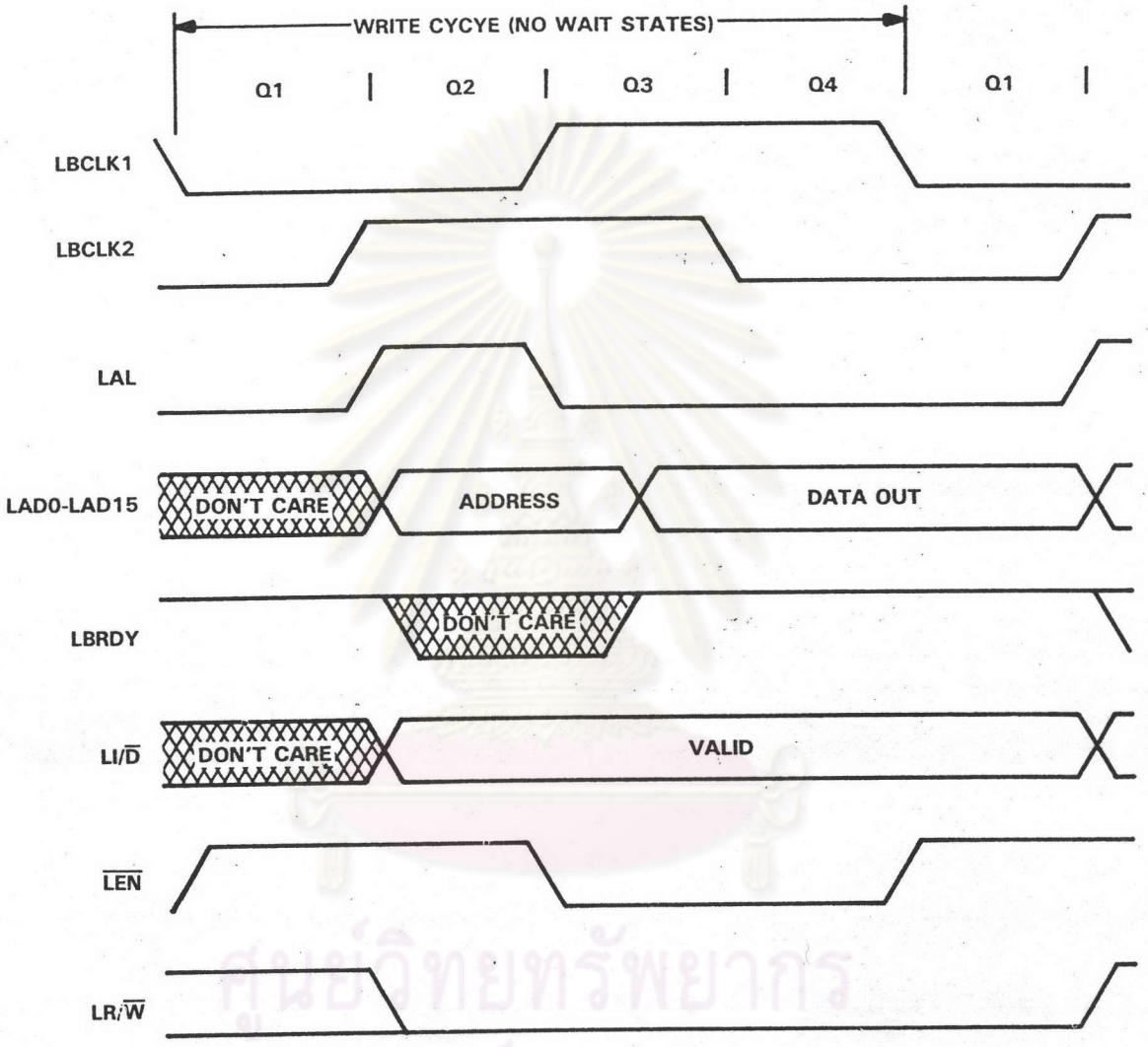
ในขณะทำไซเคิลการอ่าน ตัวบัสมาสเตอร์จะไม่ป้อนสัญญาณ LEN จนกระทั่งมีสภาวะไตร-สเตทบนสัญญาณ LAD ในการเตรียมอ่านข้อมูล ตัวบัสสเลฟต้องไม่ส่งสัญญาณบนสาย LAD จะกระทั่งบัสมาสเตอร์ส่ง LEN ออกมา ข้อมูลเอาท์พุทจะใช้ได้ภายในเวลาดีเลย์ที่กำหนดจากขอบขาขึ้นของสัญญาณ Q3 แม้ว่าสัญญาณไทมิ่งนั้นตัวสเลฟ



รูปที่ 4.9 ตารางสัญญาณนาฬิกาของไซเคิลการอ่านของแลนอะแดปเตอร์บัส

(TMS380 Adapter Chipset User's Guide 1986)

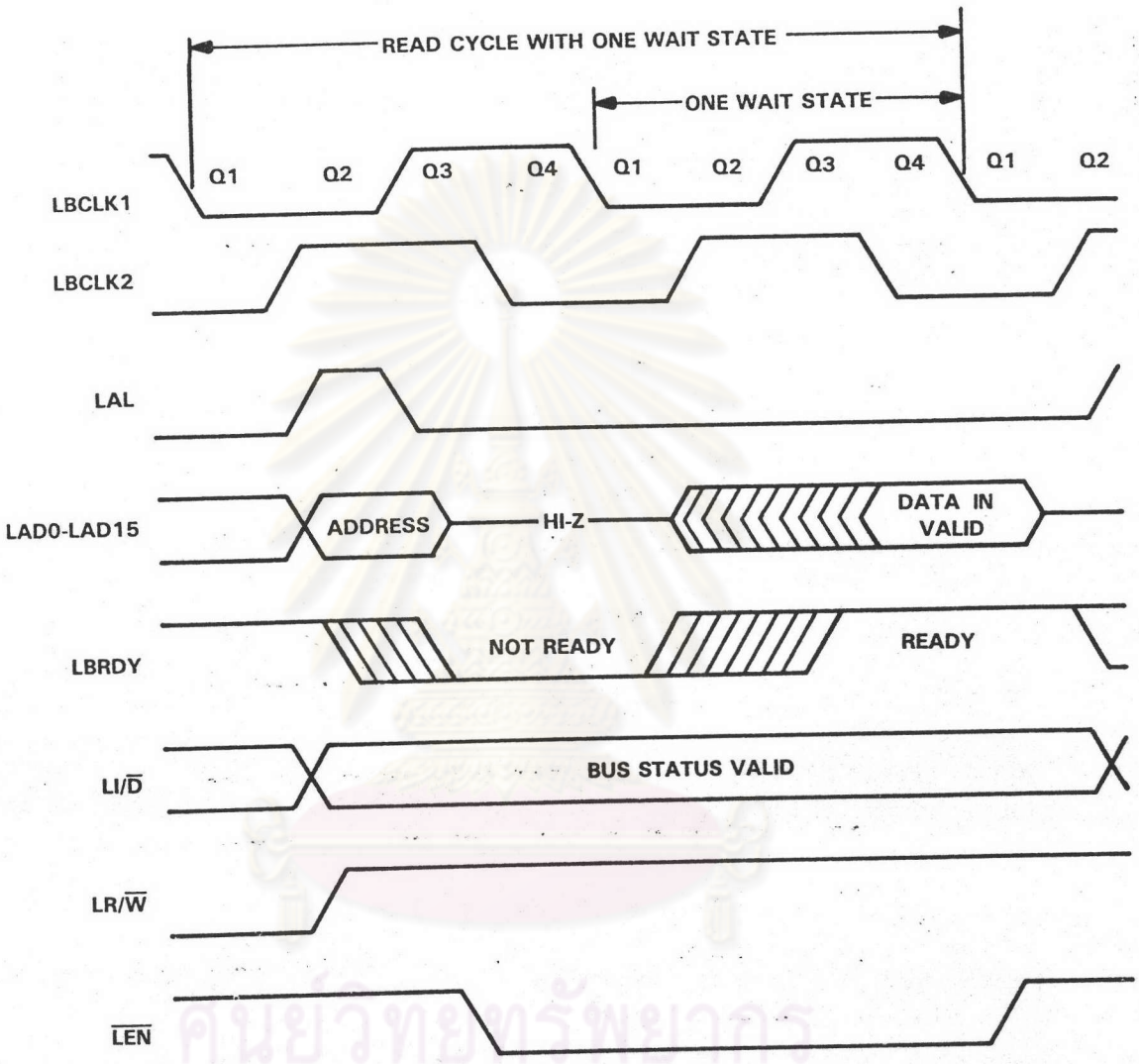
ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 4.10 ตารางสัญญาณนาฬิกาของไซเคิลการเขียนของแลนอะแดปเตอร์บัส

(TMS380 Adapter Chipset User's Guide 1986)

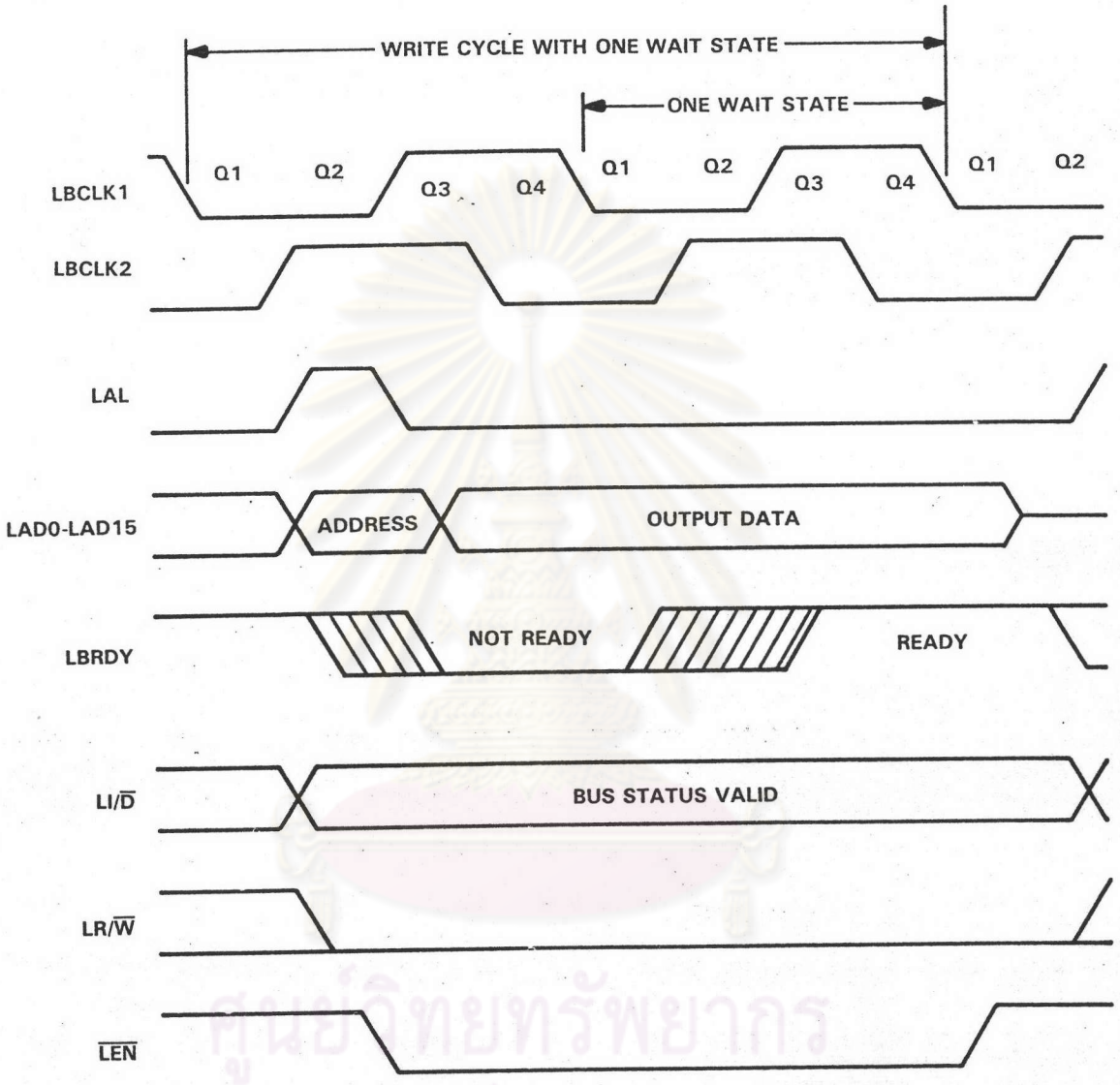
ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 4.11 ตารางสัญญาณนาฬิกาของไซเคิลการอ่านแบบ 1 wait-state

(TMS380 Adapter Chipset User's Guide 1986)

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 4.12 ตารางสัญญาณนาฬิกาของไซเคิลการเขียนแบบ 1 wait-state (TMS380 Adapter Chipset User's Guide 1986)

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

ต้องเลือกที่จะขับดาต้าออกมาในช่วงขาขึ้นของ Q4 ตัวบัสมาสเตอร์จะตรวจสอบอินพุทดาต้าในขณะขอบขาลงของ Q4 ตัวบัสสเลฟ จะยังคงสัญญาณเอาท์พุทดาต้าไว้จนกระทั่งเงื่อนไขใดเงื่อนไขหนึ่งในสองเงื่อนไขนี้เกิดขึ้น คือ

ก. บัสมาสเตอร์ไม่ทำให้ $\overline{\text{LEN}}$ เป็นโลว์อีกต่อไป หรือ

ข. LBRDY ถูกส่งออกมา

บัสสเลฟต้องทำให้เอาท์พุทเป็นไตร-สเตทภายในเวลาดีเลย์ที่กำหนดหลังจากการหายไปของสัญญาณ $\overline{\text{LEN}}$

4.7.3 ไซเคิลการเขียน (Write cycle)

ในการทำไซเคิลการเขียนนั้น ตัวบัสมาสเตอร์จะส่งสัญญาณ $\overline{\text{LEN}}$ ในตอนเริ่ม Q3 และเขียนข้อมูลออกมาที่สาย LAD การเขียนข้อมูลนี้จะไม่มีความหมายถ้ายังไม่ได้ส่ง $\overline{\text{LEN}}$ ออกมา บัสสเลฟจะทำการรับรู้ข้อมูลตอนขอบขาลงของ LBCLK1 (ตอนท้ายของQ4) บัสมาสเตอร์จะให้ค่า$\overline{\text{LEN}}$เป็นแอคทีฟหลังจากขาลงของสัญญาณ LBCLK1เพื่อที่จะให้บัสไดร์เวอร์ภายนอกยังคงค่าดาต้าไว้หลังจาก write strobe บัสมาสเตอร์จะแทรกสัญญาณ $\overline{\text{LEN}}$ ในช่วง Q1 ก่อนที่จะเริ่มบัสไซเคิลต่อไป

4.7.4 วงจรผลิตสถานะการคอย (Wait State Generation)

แต่ละบัสไซเคิลจะใช้ช่วงเวลาอย่างน้อยหนึ่งช่วง LBCLK ดังแสดงอยู่ในรูป 4.9 แต่สามารถที่ขยายต่ออีกหนึ่งไซเคิลถ้าหน่วยความจำนั้นทำงานช้าเวลาที่เพิ่มนี้เรียกว่าสถานะการคอย (wait state) การทำสถานะการคอยนั้นจะอนุญาตให้เฉพาะโปรแกรมที่อยู่ในโปรเซสเซอร์สื่อสารเท่านั้น โดยจะยอมให้เกิดถ้าเป็นแรมสัญญาณ LBRDY จะถูกทำให้เป็นโลว์ได้ทั้งโดยบัสมาสเตอร์และบัสสเลฟ เพื่อทำการขยายอินเตอร์เฟสการ์ด บัสหน่วยความจำและบัสไซเคิลภายในดังแสดงในรูป 4.9 ไม่ว่า CP ชิปจะเป็นมาสเตอร์หรือสเลฟ โดยมันจะรับ LBRDY เป็นอินพุต PH ชิปจะดู LBRDY เมื่อมันถูกแอคเซสเป็นตัวสเลฟ (เช่น มาสเตอร์ภายนอกอาจร้องขอสถานะการคอย เมื่อกำลังทำการแอคเซส PHรอม หรือ PH คอนโทรลรีจิสเตอร์) แต่มันจะไม่สนใจสถานะการคอย ถ้ามันไม่ใช่บัสมาสเตอร์ ตัวมาสเตอร์(ในขณะที่ทำการเขียน) หรือ

สเลฟ(ในขณะทำการอ่าน) จะทำการส่งข้อมูลออกเอาต์พุตออกมาตลอดตราบใดที่สัญญาณ LBRDY ยังเป็นโลว์อยู่

วงจรภายนอกจะให้สัญญาณสถานะการคอย ออกมาโดยให้ LBRDY เป็นโลว์(โดยวงจรคอลเลคเตอร์เปิด) ก่อนที่จะถึงขอบขาลงของ LBCLK2 สถานะการคอยจะถูกส่งออกมาจนกระทั่ง LBRDY ถูกทำให้เป็นไฮ LBRDY ต้องถูกปล่อยออกให้ทันเวลาทั้งนี้เพื่อป้องกันไม่ให้เกิด สถานะการคอยที่สองออกมา ตราบใดที่ LBDRY เป็นโลว์ โปรเซสเซอร์จะถูกบังคับให้ทำการรอ โดยที่มันจะหยุดตัว คำสั่งไมโคร (Micro-instruction) ในขณะนั้นและได้ถูกอนุญาตให้ดำเนินการสถานะต่อไปได้ ตัวซีพียูจะไม่สามารถตอบสนองการร้องขออินเตอร์รัปต์หรือการร้องขอบัสได้จนกระทั่ง LBRDY จะถูกปล่อย ในกรณีนี้จะนำไปใช้ได้ทั้งไซเคิลหน่วยความจำที่ถูกควบคุมโดย โปรเซสเซอร์สื่อสารหรือที่ไม่ใช่ไซเคิลหน่วยความจำก็ได้

ข้อสังเกตุอย่างหนึ่งคือตัวโปรเซสเซอร์สื่อสาร จะแอคเซสแรมในชิปหรือรีจิสเตอร์ LBRDY ต้องไม่ถูกทำให้เป็นโลว์

4.7.5 รหัสแสดงสถานะของบัส (Bus Status code)

สัญญาณทั้ง LI/D และ LR/W จะให้ข้อมูลสถานะของแลนอินเตอร์เฟสการ์ดบัสไซเคิลที่กำลังทำงานอยู่ ความหมายของสถานะของบัสที่ออกมาแสดงตามตาราง 4.5 DW จะถูกส่งมาเมื่อ CP เขียนข้อมูลไปยังหน่วยความจำหรือในเงื่อนไขการ RESET ส่วน DR จะถูกส่งออกมาเมื่อ CP อ่านข้อมูลจากหน่วยความจำ เมื่อ CP เรียกแอดเดรส แรม ภายใน มันจะส่งสถานะบัสและแอดเดรสที่เหมาะสมออกมาที่บัส

IOP รหัสสถานะของบัสจะถูกส่งออกมาเมื่อ CP อ่านคำสั่งหรือตัวโอเปอร์แรนด์ (Operand) ของคำสั่งนั้น การทำเช่นนี้ก็เพื่อเป็นการสร้างคุณภาพในการอ้างอิงคำสั่งเมื่อทำการดีบัก (Debug) และ เพื่อทำการแยกแยะคำสั่งอ้างอิงกับ PH prefetch unit

4.7.6 แผนที่หน่วยความจำของ (Bus Memory Map)

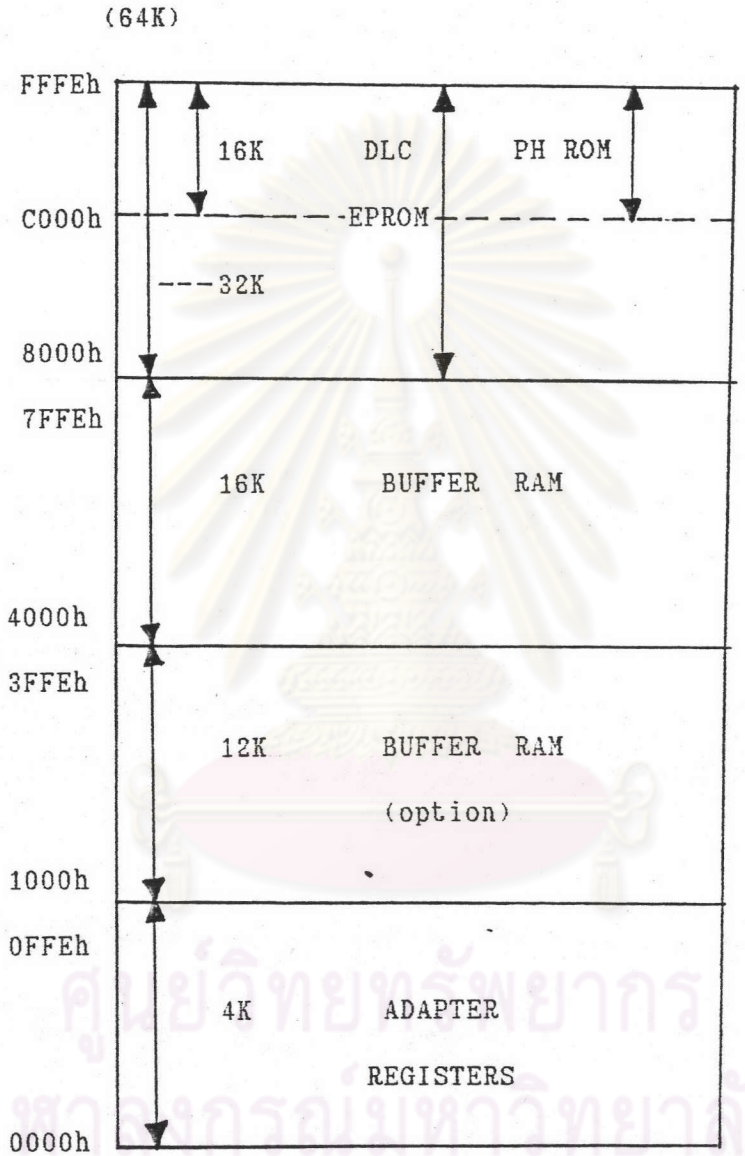
แผนที่หน่วยความจำสำหรับอินเตอร์เฟสการ์ดนั้น แสดงดังรูปที่ 4.13

LR/$\overline{W}$	LI/$\overline{D}$	ชื่อสัญญาณ	ความหมาย
0	0	DW	Data Write
0	1	DR	Data Read
1	0	—	invalid
1	1	IOP	Instruction or Immediate Operand

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

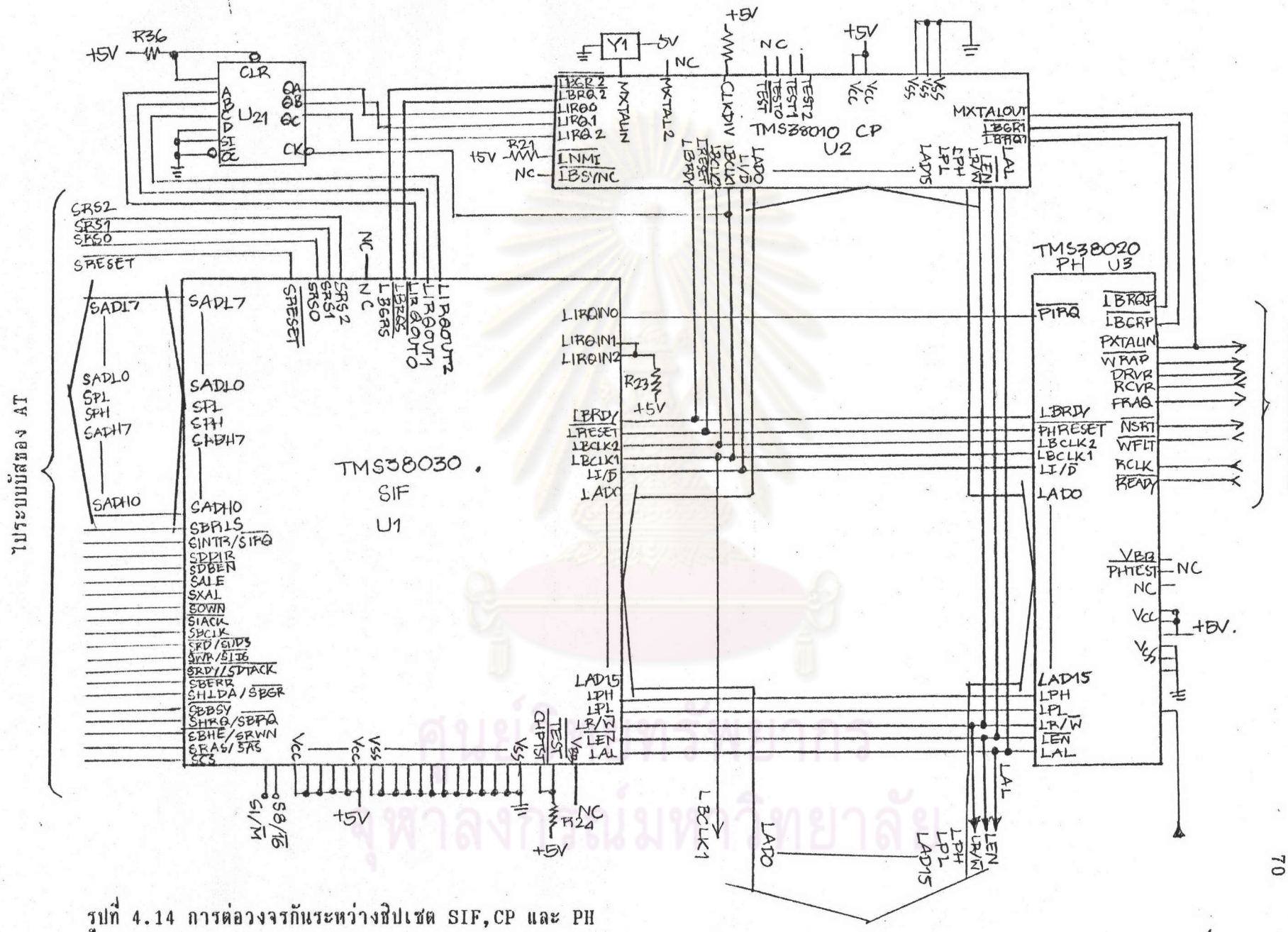
ตารางที่ 4.5 รหัสแสดงสถานะภาพของอะแดปเตอร์บัส

(TMS380 Adapter Chipset User's Guide 1986)

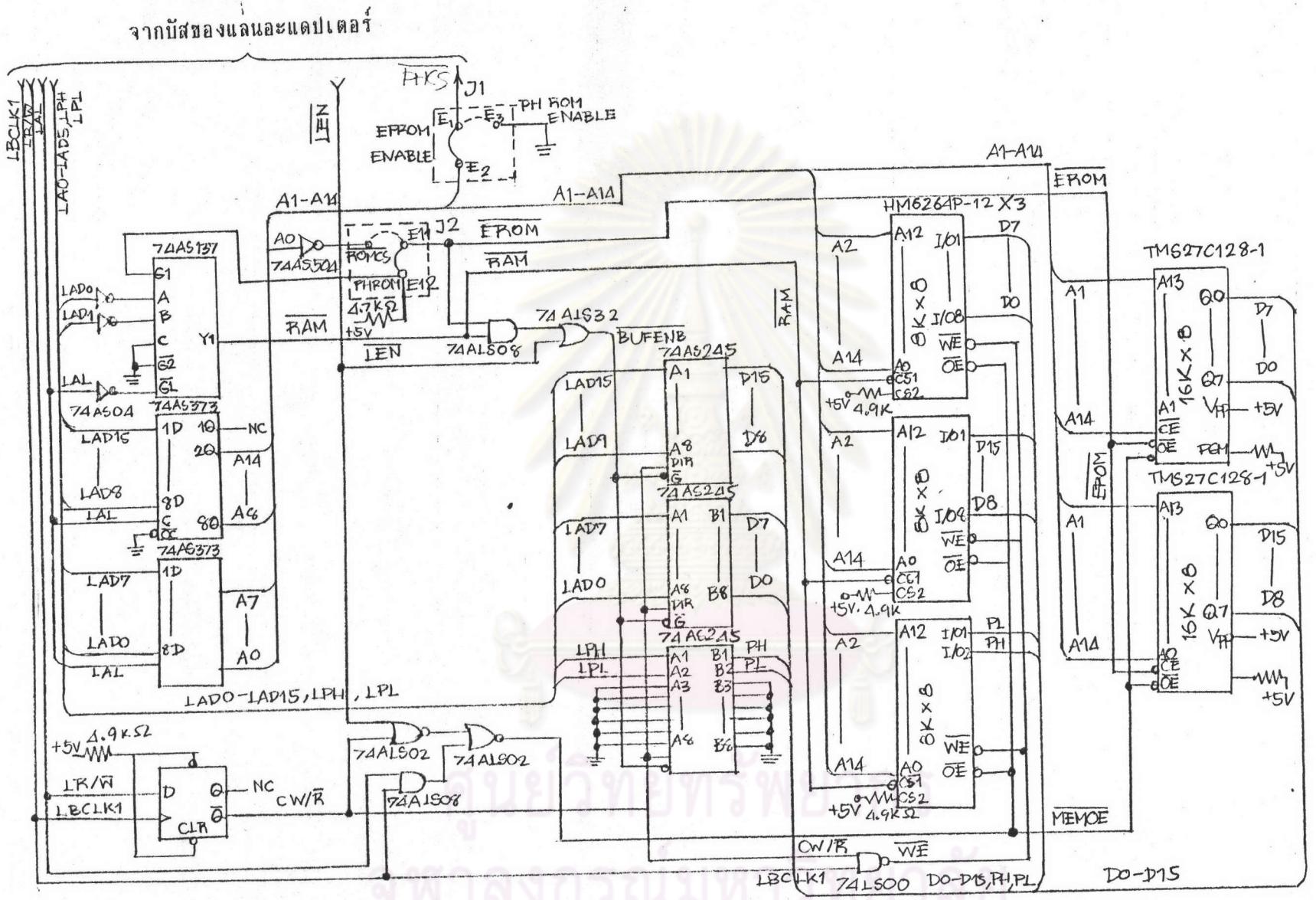


รูปที่ 4.13 แผนที่หน่วยความจำของอินเตอร์เฟสการ์ดบัส

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 4.14 การต่อวงจรกันระหว่างชิปเซต SIF,CP และ PH



รูปที่ 4.15 การต่อวงจรกันระหว่างชิปเซตส่วนการต่อหน่วยความจำ

ตารางที่ 4.6 รายการอุปกรณ์ของวงจรที่ต่อกันภายในอินเตอร์เฟสการ์ด

รายการ	ขนาด,เบอร์	รายการ	ขนาด,เบอร์
R1	121Ω	C3	6800pF
R2	604Ω	C4	8.2μF
R3,R4	2.49 KΩ	C5	680pF
R5	825Ω	C7,C8	62pF
R7,R8	121Ω	C9	0.1μF
R9,R10	75Ω	C10,C11	8.2μF
R12,R13	300Ω	C12	0.47μF
R14,R15	50Ω	L1	56μH
R16	330Ω	D1-D4	1N4004
R17	150Ω	D5-D12	1N4148
R18	2.3KΩ	D13	10V,5W 1N5347B
R19-R25	4.7KΩ	T1,T2	TN12837
R27-R36	1KΩ	Y1	Osc. KXO-01
C1	180pF		
C2	200pF		

โดยที่แผนที่หน่วยความจำนี้แสดงให้เห็นพื้นที่ทั้งหมดที่สามารถได้ในอะแดปเตอร์ คือ ตั้งแต่ 0000h ไปถึง FFFEh ซึ่งตามรูปจะเห็นว่าเนื้อที่ตั้งแต่ 0000h-0FFEh นั้นได้ถูกันไว้ใช้สำหรับรีจิสเตอร์ภายในและแรมในชิปเอง ส่วนเนื้อที่ตั้งแต่ 1000h-7FFEh จะใช้สำหรับหน่วยความจำแรมเพิ่มเติมภายนอก ซึ่งหน่วยความจำส่วนนี้โปรแกรมประยุกต์ต้องการใช้ สำหรับเนื้อที่ตั้งแต่ C000h-FFFEh นั้นจะเป็นเนื้อที่ของ รอมใน PH ซึ่งเก็บโปรแกรมต่างๆ แต่ถ้าเอาอีพรอม (EPROM)ของเท็กซัสอินสตรูเม้นท์ ซึ่งมีโปรแกรม DLC (Data Link Control) มาใส่จะทำให้เราสามารถเขียนโปรแกรมติดต่อในระดับ LLC ได้ และแอดเดรสของ อีพรอมนี้จะทับแอดเดรสของ PH รอมแต่ว่าในตัวอีพรอมนี้จะมีโปรแกรมที่แทนกันได้กับ PH อยู่ด้วยแล้ว

ตามรูป 4.15 จะแสดงให้เห็นว่า แรมส่วนขยาย ต่อเพิ่มเข้าไปในอินเตอร์เฟสการ์ดอย่างไร ซึ่งหน่วยความจำชุดนี้จะใช้แรม เบอร์ HM6264 8K x 8 สแตติกแรม (Static RAM) ต่อกันทำให้ได้หน่วยความจำเพิ่มเป็น 16 Kb เพื่อเป็นบัฟเฟอร์ในการส่งและรับเฟรม แอดเดรสจะถูกดีมัลติเพล็กซ์โดยไอซีแลตช์เบอร์ AS373 2 ตัว ข้อมูลจะถูกส่งไปมาในส่วนหน่วยความจำ โดยไอซีบัสทรานซีฟเวอร์เบอร์ AS245 สัญญาณ LR/$\overline{W}$ จะต้องถูกแลตช์ก่อนที่จะถูกใช้เป็นสัญญาณทิศทางให้แก่ AS245 เพราะว่า LR/$\overline{W}$ อาจเปลี่ยนไปก่อนที่ $\overline{LEN}$ จะถูกดึงออกไปสัญญาณ $\overline{LEN}$ จะถูกใช้ร่วมกับการถอดรหัสแอดเดรสในการที่จะอีนาเบิล(Enable)ตัว 245 สัญญาณ LR/$\overline{W}$ จะถูก NAND กับ LBCK1 เพื่อสร้างสัญญาณ write strobe ที่เป็นแอคทีฟโลว์สำหรับ แรม ส่วนสัญญาณชิปซีเล็คนั้นจะได้มาจากวงจรถอดรหัสที่เหมาะสม โดยที่ แรม 16 Kbนี้จะเริ่มตั้งแต่แอดเดรส 4000h ไปถึง 7FFEh

และที่แสดงตามรูป 4.15 มีการต่ออีพรอม(EPROM) 16K X 8 2 ตัว ซึ่งแอดเดรสจะเริ่มตั้งแต่ 8000h - FFFEh และโดยที่ อีพรอมภายนอกนี้สามารถแทนที่เนื้อที่ของรอม ในโพรโตคอลแฮนเดลอร์ ดังนั้นก็สามารถที่จะมีโปรแกรมอื่นๆให้โพรโตคอลแฮนเดลอร์ได้ เช่น ดีบัก หรือ โมดิฟายด์อินเตอร์เฟสการ์ดซอฟท์แวร์ในกรณีนี้จัมเปอร์ J1 และ J4 จะถูกเซตให้ดิสเอเบิล PH รอม และอีนาเบิลอีพรอม

4.8 การอินเตอร์เฟสทางซอฟต์แวร์

ในส่วนนี้จะกล่าวถึงการใช้ซอฟต์แวร์อินเตอร์เฟสระหว่างโปรเซสเซอร์ ของระบบคอมพิวเตอร์ที่ต่อกับตัวอินเตอร์เฟสการ์ด ซึ่งซอฟต์แวร์นี้จะใช้ในการควบคุมการทำงานของอินเตอร์เฟสการ์ด ในการทำการส่งข้อมูลและการทำงานตัวอินเตอร์เฟสการ์ดจะถูกควบคุมโดยการทำไดเร็กแอคเซสไปยังรีจิสเตอร์ 4 ตัว และช่องทางการทำ ดีเอ็มเอ นี้จะอยู่ในตัว ชิสเต็มอินเตอร์เฟส (SIF)ของ TMS38030 ชีสเต็มอินเตอร์เฟสรีจิสเตอร์จะใช้สำหรับทำขบวนการเริ่มต้นของอินเตอร์เฟสการ์ด ทำการอ่านสาเหตุการอินเตอร์รัปต์ที่ส่งไปยังระบบที่ต่ออยู่แล้วจะทำการส่งอินเตอร์รัปต์ไปยังอินตอร์เฟสการ์ด เพื่อกระตุ้นดีเอ็มเอ ทำการส่งหรือรับข้อมูลกับหน่วยความจำระบบ

ช่องดีเอ็มเอจะใช้ส่งผ่านคำสั่ง พารามิเตอร์และเฟรมไปยังอินเตอร์เฟสการ์ดและรับโค้ดแสดงว่าเรียบร้อยแล้วกลับ และรับเฟรมจากอินเตอร์เฟสการ์ด สรุปได้ว่าการเคลื่อนย้ายข้อมูลทั้งหมดที่เข้าและออกจากริง จะผ่านโดยการทำ ดีเอ็มเอ เท่านั้น

4.8.1 สรุปบัฟเฟอร์ที่ระบบต้องการ

ในรายชื่อต่อไปนี้จะเป็นจะเป็นส่วนหน่วยความจำที่ต้องการ โดยระบบ

SYSTEM COMMAND BLOCK (SCB) เป็นบัฟเฟอร์ขนาด 6 ไบต์ ซึ่งจะใช้ในการเก็บคำสั่งที่จะเอ็กซีคิวต์โดยอินเตอร์เฟสการ์ด และตัวชี้ (Pointer) ไปยังแอดเดรสขนาด 24 บิต

SYSTEM STATUS BLOCK (SSB) จะเป็นบัฟเฟอร์ขนาด 8 ไบต์ ซึ่ง จะใช้เก็บรหัสสถานะที่ส่งกลับหลังจากการทำงานของคำสั่งอินเตอร์เฟสการ์ดเสร็จเรียบร้อยแล้ว

COMMAND PARAMETER LIST คำสั่งบางคำสั่ง (เช่น OPEN) ต้องการบล็อกของหน่วยความจำเพื่อทำเป็นพารามิเตอร์บล็อก ในขณะที่คำสั่งทำการเอ็กซีคิวต์เสร็จเรียบร้อยแล้ว ก็จะปล่อยเนื้อที่ที่จองไว้นี้ให้ใช้ทำอย่างอื่นได้

RECEIVE LIST คำสั่ง RECEIVE จะต้องมีการจอง

RECEIVE LIST ในหน่วยความจำระบบ ขนาดของหน่วยความจำจะขึ้นอยู่กับขนาดและจำนวนลิสต์ที่ใช้ในโปรแกรมประยุกต์ การเลือกขนาดนี้จะเลือกได้ตอนทำขบวนการเริ่มต้นของอินเตอร์เฟสการ์ดว่าจะเป็น 14,24 หรือ 26 ไบต์

TRANSMIT LIST คำสั่ง TRANSMIT จะต้องมีการจอง TRANSMIT LIST ในหน่วยความจำระบบและหลักการทั้งหมดจะเหมือนกับ RECEIVE LIST ทุกประการ

PRODUCT ID คำสั่ง OPEN นั้นจะต้องการตัวชี้ที่จะชี้ไปยัง Product ID ขนาด 18 ไบต์ ซึ่งจะเป็นส่วนหนึ่งของ open parameter list ผู้ออกแบบซอฟต์แวร์ระบบควรจะกันเนื้อที่ 18 ไบต์ของหน่วยความจำระบบสำหรับทำคำสั่งนี้ เมื่อทำเสร็จแล้วมันจะปล่อยเนื้อที่ให้ทำอย่างอื่นได้

4.8.2 รายละเอียดรีจิสเตอร์ทั้งหมด

4.8.2.1 รีจิสเตอร์อินเตอร์รัปต์ (Interrupt Register)

จะใช้สำหรับการส่งอินเตอร์รัปต์ ไปยังอินเตอร์เฟสการ์ดและทำการอ่านสถานะการอินเตอร์รัปต์จากอินเตอร์เฟสการ์ด บิต 0-7 จะสามารถถูกเซตเป็น 1 แต่ไม่สามารถถูกรีเซตเป็น 0 ได้โดยระบบที่ต่ออยู่แต่จะถูกรีเซตได้โดย โปรเซสเซอร์สื่อสารเท่านั้น ในทางตรงกันข้ามบิตที่ 8 จะสามารถถูกรีเซตเป็น 0 แต่ไม่สามารถถูกเซตเป็น 1 ได้โดยระบบที่ต่ออยู่ แต่จะสามารถถูกเซตได้โดยโปรเซสเซอร์สื่อสาร บิต 9-15 จะสามารถถูกอ่านโดยระบบที่ต่ออยู่ได้อย่างเดียวเท่านั้น ซึ่งบิตเหล่านี้จะถูกเซตหรือรีเซตโดยโปรเซสเซอร์สื่อสาร

4.8.2.2 รีจิสเตอร์แอดเดรส (Address Register)

ตัวแอดเดรสรีจิสเตอร์ จะบรรจุตัวชี้ที่จะชี้ไปยัง แรมภายในของอินเตอร์เฟสการ์ด ซึ่งจะแอคเซสโดย Data หรือ Data/Auto-increment รีจิสเตอร์ บิตทั้ง 16 บิตสามารถจะถูกอ่านแม้ว่าเพียงแค่บิต 5-14 ซึ่งถูกเซตหรือรีเซตโดยระบบที่ต่ออยู่ โดยการนี้มันจะยอมให้ระบบที่ต่ออยู่ทำการแอคเซส แรม ภายในของอินเตอร์เฟสการ์ดขนาด 2 Kb ตำแหน่งเริ่มต้นของแรมซึ่งจะถูกอ่านเป็นฟังก์ชันกับอะแดป-

BIT	0	INTERRUPT ADAPTER
	1	ADAPTER RESET
	2	SSB CLEAR
	3	EXECUTE
	4	SCB REQUEST
	5	RECEIVE CONTINUE
	6	RECEIVE VALID
	7	TRANSMIT VALID
	8	RESET SYSTEM INTERRUPT
	9	X
	10	X
	11	X
	12	X
	13	X
	14	X
LSB	15	X

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

รูปที่ 4.16 อินเตอร์รัปต์รีจิสเตอร์

(TMS380 Adapter Chipset User's Guide 1986)

เตอร์เซตบิต 0-4 ในสภาวะปรกตินั้นบิต 0-4 จะเซตเป็น 00001 แต่ถ้าเกิด Adater Chaeck Interrupt มันจะถูกเซตเป็น 00000 บิต 15 จะถูกควบคุมโดยอินเตอร์เฟสการ์ดเช่นเดียวกับข้อมูลที่ส่งไปมาบนแลนอินเตอร์เฟสการ์ดบัสซึ่งจะเป็น 16 บิตเวิร์ดเท่านั้น

4.8.2.3 รีจิสเตอร์ข้อมูล (Data Register)

รีจิสเตอร์ข้อมูล เปรียบเสมือนเป็น'พอร์ต'(Port) ไปยังแรมภายในของอินเตอร์เฟสการ์ด ซึ่งตัวข้อมูลจะสามารถถูกเขียนหรืออ่านก็ได้ ตำแหน่งของแรมภายในนี้จะถูกอ่านหรือถูกเขียนโดยรีจิสเตอร์นี้ และจะถูกชี้โดยแอดเดรสที่บรรจุในแอดเดรสรีจิสเตอร์ความสามารถในด้านนี้ ทำให้ตัวแปรการทำขบวนการเริ่มต้นสามารถถูกส่งผานไปยังอินเตอร์เฟสการ์ดสำหรับทำการขบวนการเริ่มต้น และความสามารถส่วนนี้ยังทำให้ยังใช้ในขบวนการตรวจสอบอินเตอร์เฟสการ์ดที่มีปัญหาอีกด้วย หลังจากการทำขบวนการเริ่มต้นแล้ว ก็จะไม่ยอมให้มีการเขียนไปยังอินเตอร์เฟสการ์ดการ์ด เมื่อมีการเขียนข้อมูลโดยระบบที่ต่ออยู่แล้ว ก็จะมีการทำการเขียนแบบ ดีเอ็มเอ บนอินเตอร์เฟสการ์ดบัส เพื่อเป็นการเขียนข้อมูลลงบนตำแหน่งที่ถูกชี้โดยแอดเดรสรีจิสเตอร์ อย่างไรก็ตามถ้าต่อมามีการอ่านค่าจากรีจิสเตอร์ข้อมูลโดยระบบที่ต่ออยู่ จะไม่มีการทำโดยการอ่านแบบ ดีเอ็มเอ ตามตำแหน่งที่ถูกชี้โดยแอดเดรสรีจิสเตอร์ การอ่านค่าที่เพิ่งถูกเขียนไปโดยการเขียนแบบ ดีเอ็มเอ นี้ ตัวแอดเดรสรีจิสเตอร์จะต้องมีการมีการเขียนซ้ำใหม่ด้วยค่าแอดเดรสเดิมที่ใช้ตอนอ่านดาต้ารีจิสเตอร์ การทำเช่นนี้จะเป็นการบังคับให้การทำการอ่านแบบ ดีเอ็มเอ บนแลนอินเตอร์เฟสการ์ดบัส ทำการอัปเดตดาต้ารีจิสเตอร์ด้วยค่าใหม่เสมอ

4.8.2.4 รีจิสเตอร์ข้อมูลที่มีการเพิ่มแอดเดรสโดยอัตโนมัติ

(Data Register with Auto incrememt)

รีจิสเตอร์นี้จะเหมือนกับ รีจิสเตอร์ข้อมูลแบบปรกติทุกประการยกเว้นว่าแอดเดรสที่อยู่ภายในนั้น จะมีการเพิ่มค่าขึ้นโดยอัตโนมัติหลังจากมี

การเขียนหรือการอ่านกับรีจิสเตอร์ เมื่ออยู่ใน 8 บิตโหมด การเขียน MSB จะทำให้แอดเดรสรีจิสเตอร์มีการเพิ่มขึ้นครั้งละสอง ซึ่งจะทำให้แรมของอินเตอร์เฟสการ์ดมีการอ่านและเขียนอย่างต่อเนื่อง โดยไม่ต้องมีการเขียนใหม่ยังไปแอดเดรสรีจิสเตอร์อีกครั้งในระหว่างการแอคเซสแต่ละครั้ง

4.8.3 การทำขบวนการเริ่มต้นของอะแดปเตอร์(Adapter initialization)

หลังจากมีการเวอร์ริฟาย(Verify)และตรวจสอบตามปรกติเรียบร้อยแล้ว ตัวซอฟต์แวร์ระบบต้องทำการทำขบวนการเริ่มต้นอินเตอร์เฟสการ์ด การทำขบวนการเริ่มต้นนี้ จะเกี่ยวกับการส่งผ่านค่าพารามิเตอร์ ไปยังอินเตอร์เฟสการ์ดโดยการอินเตอร์เฟสแบบ DIO ตัวพารามิเตอร์ต่างๆจะทำการกำหนดในสิ่งต่อไปนี้

ก. แอดเดรสในหน่วยความจำระบบของ SCB และ SSB

ข. ตัวแปรในการควบคุมการทำอินเตอร์รัปต์

ก่อนที่ขบวนการทำขบวนการเริ่มต้นจะเสร็จสิ้น อินเตอร์เฟสการ์ดจะทำการทดสอบการทำ ดีเอ็มเอ อินเตอร์เฟส ซึ่งการทดสอบนี้จะรวมถึงสิ่งเหล่านี้

ค. ทำการเขียนแบบ ดีเอ็มเอ ไปยังทั้ง SCB และ SSB

ง. ทำการอ่านแบบ ดีเอ็มเอ ไปยังทั้ง SCB และ SSB แล้วทำการเทียบค่าที่ได้

การทำ ดีเอ็มเอ เหล่านี้ไม่ต้องทำการรันโปรแกรมจากระบบที่ต่ออยู่เลย แต่ทว่าในกรณีที่การทดสอบแล้วเกิดข้อผิดพลาดขึ้น อินเตอร์เฟสการ์ดจะทำการส่งค่าที่ผิดพลาดนั้นกลับไปยังอินเตอร์รัปต์รีจิสเตอร์

4.8.4 บล็อกคำสั่งและสถานะ (Command and Status Block)

ทั้งสองตัวนี้เป็นบล็อกควบคุมที่มีแอดเดรสคงที่ โดยระบบจะเป็นผู้จัดการมาให้เรียบร้อยแล้ว แอดเดรสเริ่มต้นของบล็อกทั้งสองนั้นจะถูกส่งไปยังอินเตอร์เฟสการ์ดในขณะที่กำลังทำขบวนการเริ่มต้นดังอธิบายไว้หัวข้อที่แล้ว บล็อกทั้งสองนั้นจะเป็นแบบการจัดแบบเวิร์ด(แอดเดรสคู่)

โดยทั่วๆไปแล้วระบบที่ต่ออยู่ จะส่งคำสั่งไปยังอินเตอร์เฟสการ์ด

โดยทำการโหลดการร้องขอ (Request)ลงใน SCB แล้วทำการอินเตอร์รัปต์อินเตอร์เฟสการ์ด จากนั้นอินเตอร์เฟสการ์ดก็จะทำการดาวน์โหลดคำสั่งและพารามิเตอร์ต่างๆ ผ่านช่องทาง ดีเอ็มเอ ของชิสเต็มอินเตอร์เฟส ถ้าบิต SCB REQUEST (บิต 4) ของอินเตอร์รัปต์รีจิสเตอร์ อินเตอร์เฟสการ์ดก็จะทำการอินเตอร์รัปต์ระบบที่ต่ออยู่หลังคำสั่งได้ถูกดาวน์โหลดลงแล้ว ก็จะเป็นการบอกระบบที่ต่ออยู่ว่า SCB พร้อมที่จะรับคำสั่งใหม่ได้แล้ว เมื่อสถานะของคำสั่งถูกส่งกลับมาแล้ว อินเตอร์เฟสการ์ดก็จะทำการโหลด SSB โดยใช้ ดีเอ็มเอ และทำการอินเตอร์รัปต์ระบบที่ต่อ หลังจากระบบได้อ่าน SSB แล้ว ระบบจะต้องให้อินเตอร์เฟสการ์ดรู้ด้วยว่า SSB นั้นว่างแล้วและพร้อมที่จะรับค่าสถานะต่างๆต่อไป สถานะภาพทั้งหมดของคำสั่งจะถูกส่งกลับมาใน SSB

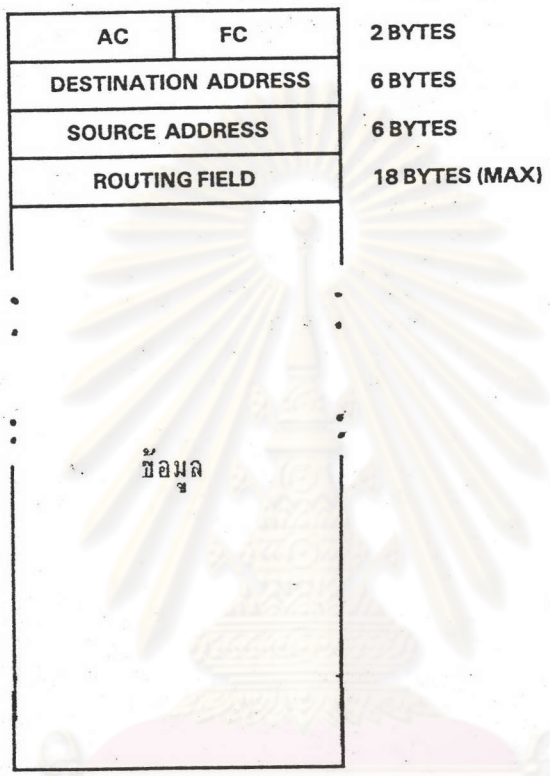
คำสั่งทั้งหมดของ SCB มีดังนี้

OPEN : ก่อนที่จะมีการสื่อสารข้อมูล ระบบที่ต่ออยู่ด้วยต้องเริ่มด้วยการสั่งคำสั่งนี้ก่อน มันจะทำการเซตแอดเดรสต่างๆของอินเตอร์เฟสการ์ดและอีนาเบิลให้รับเฟรมจากริงได้ และหลังจากคำสั่ง OPEN มันจะส่งคำสั่ง RECEIVE ตามมาทันทีทันใด ในขณะสั่ง OPEN การทำการร้องขออินเตอร์รัปต์จะถูกโฮลด์ไว้หมด

TRANSMIT : จะใช้สำหรับการส่งเฟรมไปยังโหนดอื่น โดยจะมีรูปแบบของเฟรมข้อมูลดังรูปที่ 4.17

TRANSMIT HALT : ใช้สำหรับการอินเตอร์รัปต์ ทรานสมิตลิสต์ หลังจากที่มีการรับรู้คำสั่งนี้แล้ว อินเตอร์เฟสการ์ดจะทำการสิ้นสุดทรานสมิตเซน เร็วที่สุดเท่าที่จะเร็วได้ เฟรมที่เข้าคิวอยู่ก็จะถูกล้างทิ้งทั้งหมด และคำสั่ง TRANSMIT จะถูกทำให้สิ้นสุดโดยจะส่ง สถานะ COMMAND COMPLETE ออกมา แต่ถ้าขณะนั้นไม่ได้สั่งคำสั่ง TRANSMIT อยู่ เมื่อสั่ง TRANSMIT HALT มันก็จะไม่สนใจ

RECEIVE : คำสั่งนี้จะใช้สำหรับรับเฟรมจากสถานีอื่น บนริงการสั่งคำสั่งนี้ปรกติจะสั่งเพียงครั้งเดียวเท่านั้น (หลังจากสั่ง OPEN) และการรับข้อมูลมันจะมีการเข้าไปเพิ่มใน receive parameter list chain โดยอัตโนมัติ ส่วนการยกเลิกคำสั่งนี้ก็โดยสั่งคำสั่ง CLOSE



รูปที่ 4.17 รูปแบบของทรานสมิตเฟรม

(TMS380 Adapter Chipset User's Guide 1986)

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

CLOSE : ใช้สำหรับการทำให้สิ้นสุดการส่งข้อมูลต่างๆบนริง หรือเป็นการสิ้นสุดคำสั่ง OPEN แบบแหรบ และมันจะทำการล้างทิ้งเฟรมทุกเฟรมในอินเตอร์เฟสการ์ด คำสั่ง CLOSE นี้จะถูกปฏิเสธโดยการส่งสถานะ COMMMAND REJECT ออกมาถ้าอินเตอร์เฟสการ์ดไม่ได้ OPEN อยู่

SET GROUP ADDRESS : ใช้สำหรับการเซต อินเตอร์เฟสการ์ดกรุ๊ปแอดเดรส ถ้ามันจะต้องมีการเปลี่ยนหลังคำสั่ง OPEN คำสั่งนี้จะถูกปฏิเสธโดยการส่งสถานะ COMMAND REJECTออกมาถ้าอินเตอร์เฟสการ์ดไม่ได้OPEN อยู่

SET FUNCTIONAL ADDRESS : ใช้สำหรับเซตและรีเซต อินเตอร์เฟสการ์ดฟังก์ชันนอลแอดเดรส ถ้ามันต้องมีการเปลี่ยนแปลงหลังจากคำสั่ง OPEN และคำสั่งนี้จะถูกรีเจ็กต์โดยการส่งสถานะ COMMAND REJECTD ออกมาถ้าอินเตอร์เฟสการ์ดไม่ได้ OPEN อยู่

READ ERROR LOG : จะใช้สำหรับการอ่าน และรีเซตตัวบันทึกความผิดพลาด (Error log) ของอุปกรณ์ที่ต่อกับอินเตอร์เฟสการ์ด หลังจากคำสั่งนี้สมบูรณ์ ตัวบันทึกความผิดพลาดจะกลายเป็นศูนย์ทั้งหมด

READ ADATER BUFFER : จะใช้สำหรับการส่งผ่าน ที่เก็บข้อมูลของอินเตอร์เฟสการ์ด ผ่านตัวอินเตอร์เฟสระบบไปยังหน่วยความจำของระบบที่ต่ออยู่

ในการเขียนโปรแกรมควบคุมการทำงานตามที่กล่าวมาทั้งหมดนี้ เป็นการติดต่อในระดับ MAC ของแผ่นวงจร แต่ถ้าจะติดต่อในระดับ LLC ซึ่งจะต้องมีรอม DLC (Data Link Control) ติดตั้งเพิ่มเข้าไปซึ่งบริษัทเท็กซัสอินสตรูเม้นท์มีจำหน่ายด้วย การติดต่อในระดับ LLC ก็จะมีหลักการเดียวกันเพียงแต่ว่าคำสั่งต่างๆเพิ่มขึ้นในส่วนท้าดาต้าลิงค์เท่านั้น ในบทที่ 6 จะกล่าวถึงวิธีการเขียนโปรแกรมติดต่อกับแผ่นวงจรโดยใช้ภาษาซี แต่ได้เขียนติดต่อกับการ์ดโทเกนริง ของไอบีเอ็ม ซึ่งหลักการโดยรวมแล้วเหมือนกัน ต่างกันเพียงในรายละเอียดของชื่อคำสั่งและบล็อกของพารามิเตอร์ที่ส่งผ่านให้กันเท่านั้น

4.9 ตัวแปรที่มีผลต่อสมรรถนะของแผ่นวงจร (TMS380 Adapter Chipset User's Guide 1986)

ตัวแปรต่างๆที่มีผลในการทำงานของแลนอินเตอร์เฟสการ์ด ซึ่งจะทำให้สมรรถนะเปลี่ยนไปก็คือ

4.9.1 ขนาดบัฟเฟอร์ภายใน

ตัวขนาดบัฟเฟอร์ภายในนี้ จะเป็นตัวจำกัดขนาด ของเบิสต์ดีเอ็มเอ โดยที่จะกำหนดขอบเขตด้านบนของจำนวนข้อมูลที่สามารถส่งไปมาได้ระหว่างโฮสต์ และอินเตอร์เฟสการ์ด โดยไม่ต้องทำการเอ็กซีคิวต์ซอฟต์แวร์ที่ทำการจัดการอะแดปเตอร์ บัฟเฟอร์ภายในดังนั้นสมรรถนะจะเพิ่มขึ้นเมื่อมีความต้องบัฟเฟอร์เพียงจำนวนน้อย เพื่อเก็บเฟรมข้อมูลไว้ และจากการทดสอบจะได้ว่าถ้าขนาดเฟรมที่เท่ากันบัฟเฟอร์ที่สามารถเก็บเฟรมไว้ได้หมดในบัฟเฟอร์เดียวหรือจำนวนบัฟเฟอร์น้อยกว่า จะมีสมรรถนะดีกว่า

4.9.2 จำนวนบัฟเฟอร์ภายในที่จัดไว้

สมรรถนะของอินเตอร์เฟสการ์ดจะเพิ่มขึ้นเมื่อดาต้าเฟรมสำหรับเฟรม i+1 มีการส่งผ่านจากโฮสต์ไปยังอินเตอร์เฟสการ์ดในขณะเฟรม i กำลังทำการส่งไปยังริงอยู่ การทำแบบนี้เรียกว่า"ไปป์ไลนิ่ง"(Pipelining) ซึ่งถ้าจะทำแบบนี้ได้ก็จะต้องมีบัฟเฟอร์ภายในเพียงพอที่จะเก็บเฟรมสองเฟรมไว้ได้ เพราะฉะนั้นถ้าเปรียบเทียบอินเตอร์เฟสการ์ดสองตัว ที่มีขนาดบัฟเฟอร์ภายในที่เท่ากันแล้ว ตัวที่มีจำนวน บัฟเฟอร์มากเพียงพอที่จะเก็บเฟรมข้อมูลได้สองเฟรมแล้ว ก็จะให้ผลรวมของการส่งข้อมูลที่ดีกว่า

4.9.3 เทคนิคการบริหารทรานสมิตลิสต์ของโฮสต์

จากการเปรียบเทียบว่าเมื่อ TRANSMIT COMMAND ถูกส่งเมื่อมีข้อมูลทุกๆ 7 เฟรม กับมีการส่งในทุกๆเฟรม จะเห็นได้ว่ามันจะให้สมรรถนะที่ดีที่สุดเมื่อลิงค์-ลิสต์ อินเตอร์เฟสมีการใช้งานอย่างเต็มที่ และการทำไปป์ไลน์ของการส่งผ่านเฟรมไม่ถูกอินเตอร์รัปต์เสียก่อน