คอมพิวเตอร์ควบคุมการหรี่ไฟ

COMPUTER CONTROL DIMMER

โดย

นายกนกศักดิ์ ศรีทองแท้ 34131101

นายรัฐสธน โพธิ์เงิน 34131129

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาอุตสาหกรรมศาสตร์บัณฑิต

ภาควิชาเทคนิคอุตสาหกรรม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2535

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการใช้

032772

ปริญญานิพนธ์ ปีการศึกษา 2535

ภาควิชาเทคนิคอุตสาหกรรม

คณะวิศวกรรมศาสตร์

เรื่อง คอมพิวเตอร์ควบคุมการหรี่ไฟ (COMPUTER CONTROL DIMMER)

ผู้จัดทำ

นายกนกศักดิ์ ศรีทองแท้ 34131101

นายรัฐสธน โพธิ์เงิน 34131129

.....อาจารย์ที่ปรึกษา

(อาจารย์อุทัย ศรีธีระวิรรจน์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

032772

หัวข้อปริญญานิพนธ์	คอมพิวเตอร์ควบคุมการหรี่ไฟ
โดย	นายกนกศักดิ์ ศรีทองแท้
	นายรัฐสธน วิพธิ์เงิน
อาจารย์ที่ปรึกษา	อาจารย์อุทัย ศรีธีระวิโรจน์
ภาควิชา	เทคนิคอุตสาหกรรม
ปีการศึกษา	2535

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังอนุมัติให้นับปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาอุตสาหกรรมศาสตร์บัณฑิต

.....คณบดีคณะวิศวกรรมศาสตร์

( )

คณะกรรมการสอบปริญญานิพนธ์

.....ประธานกรรมการ

( )

.....กรรมการ

( )

.....กรรมการ

( )

.....กรรมการ

( )

.....กรรมการ

( )

ลิขสิทธิ์ของคณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	คอมพิวเตอร์ควบคุมการหรี่ไฟ
ชื่อ	นายกนกศักดิ์ ศรีทองแท้
	นายรัฐสธน โพธิ์เงิน
อาจารย์ที่ปรึกษา	อาจารย์อุทัย ศรีธีระวิรรจน์
ภาควิชา	เทคนิคอุตสาหกรรม
ปีการศึกษา	2535

บทคัดย่อ

ปัจจุบันนี้มีการนำ Dimmer มาใช้งานกันอย่างแพร่หลาย ทั้งในส่วนของงานทางด้านงานบันเทิง และในโรงแรม ซึ่งการศึกษาถึงลักษณะการควบคุมใช้งาน Dimmer มีอยู่ 2 วิธีด้วยกันคือ

1. การควบคุมด้วยระบบอนาล็อก ซึ่งเป็นการควบคุมแบบนี้ มีข้อจำกัด คือ ไม่สามารถเปลี่ยนแปลงการควบคุมจะต้องทำการแก้ไขวงจรในด้าน Hard ware

2. การควบคุมด้านไมโครคอมพิวเตอร์ เป็นแบบที่เป็นที่นิยมกันเพิ่มมากขึ้น เพราะการควบคุมสามารถควบคุมด้าน Soft ware ทำให้การเปลี่ยนแปลงรูปแบบการควบคุมเพียงแค่แก้ไขในส่วนของ Soft ware เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Project Report Title	Computer Control Dimmer
Name	Mr. Kanoksak Srithongthae
	Mr. Rathsathon Po-Ngen
Project Report Advisor	Mr. U-thai Sritheeravirojana
Department of	Industrial Technology
Academic Year	1992

Abstract

Nowadays, Dimmer is used in many ways both in entertainment and hotel. There are 2 forms of digital control of Dimmer.

1. Control with analog circuit but it is limited because we cannot change form of control. We can change in the case of changing in the part of hardware.

2. Control with microcomputer , that is more popular than the first one because we can change form of control with software which is easier than the first way.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทที่ 1 บทนำ.....1

-Dimmer.....1

-ประเภทของ Dimmer.....1

-หลักการทำงานเบื้องต้นของ Computer Control Dimmer.....3

บทที่ 2 โครงสร้างของคอมพิวเตอร์ IBM-AT.....4

-เครื่องไอบีเอ็มเอที.....4

-วงจรซีพียู.....6

-ไอซี 80286.....8

-ชิป 82284.....11

-ชิป 82284 ตัวควบคุมบัส.....12

-การต่อ 80286 82288 และ 82284 แบบพื้นฐาน.....13

-พอร์ทอินพุตและเอาท์พุตของพีซีเอที

โครงสร้างการวางพอร์ทอินพุตและเอาท์พุต.....15

-วงจรไทเมอร์,อินเตอร์รัพท์.....21

-ดีเอ็มเอ.....23

-การเขียนโปรแกรมดีเอ็มเอแบบ 16 บิต.....32

-สล็อตหรือช่องต่อสำหรับอินพุตหรือเอาท์พุต.....33

-วงจรลำโพง.....40

-วงจรควบคุมคีย์บอร์ด.....40

-รีจิสเตอร์แสดงสถานะ.....42

-วงจรกำเนิดเวลาจริงและซีมอสแรม.....45

บทที่ 3 อินเตอร์เฟส การ์ด.....52

-บทนำ.....52

-การจัดแอดเดรสสำหรับอินพุตเอาท์พุต ในไอบีเอ็ม พีซี.....52

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-เทคนิคการดีโค้ดแอดเดรส.....54
-การเชื่อมต่ออุปกรณ์อินพุต เอาท์พุต กับไมโครคอมพิวเตอร์.....66
-8255 PPI.....76
-8255 Interface Card.....90
-การทดสอบและวงจรการทำงาน.....91
บทที่ 4 การทำงานของ Dimmer.....99
-การนำไปใช้งาน.....100
บทที่ 5 Digital to Analog Converter (DAC).....101
-วงจร R-2R Ladder Converter.....102
บทที่ 6 วงจร Decoder.....103
บทที่ 7 PROGRAM ควบคุมการ Dimming.....105
บทที่ 8 บทสรุปและข้อเสนอแนะ.....119
เอกสารอ้างอิง.....120
ภาคผนวก.....121

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์นี้สำเร็จลงได้ด้วย ความช่วยเหลือของท่านอาจารย์ อุทัย ศรีธีระวิโรจน์ ที่กรุณาให้คำปรึกษาในเรื่องต่าง ๆ และขอขอบคุณ นายมนต์เทพ เหมือนเดช ที่ช่วยเหลือทางด้านโปรแกรมกราฟิกและการควบคุม ขอขอบคุณ นายจิตรกร วิริยางกรูภาพ และน.ส.รุ่งรัตน์ คล้ายแย้ม ที่ช่วยเหลือในด้านการพิมพ์หนังสือ และนายบุญเสม สารจิตต์ ที่ช่วยเหลือในหลาย ๆ ด้าน และทุกท่านที่ทำให้ปริญญานิพนธ์นี้สำเร็จลุล่วงด้วยดี

นายกนกศักดิ์ ศรีทองแท้

นายรัฐสธน โพธิ์เงิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

ในปัจจุบันความเจริญก้าวหน้าทางเทคโนโลยี ได้พัฒนาให้เครื่องมือ เครื่องใช้และอุปกรณ์ต่าง ๆ เอื้ออำนวยความสะดวกให้แก่ผู้ใช้เป็นอย่างมาก ทั้งยังรวดเร็วและแม่นยำอีกด้วย โดยที่คอมพิวเตอร์ได้เข้ามามีบทบาทเป็นอย่างมาก ในการอำนวยความสะดวกสบายให้แก่มนุษย์ในชีวิตประจำวัน

Computer Control Dimmer ก็เช่นกันเป็นการนำเอาความก้าวหน้าทางเทคโนโลยีของคอมพิวเตอร์เข้ามาในการควบคุมหลอดไฟ โดยจะควบคุมให้สามารถเปิด/ปิดและควบคุมปริมาณความสว่างของหลอดไฟได้ตามต้องการ ซึ่งจะทำให้เกิดความสะดวกและแม่นยำมากกว่าการควบคุมโดยใช้คน

Dimmer คือ อุปกรณ์ทางอิเล็คทรอนิกส์ที่ทำหน้าที่ปรับระดับแรงดันไฟฟ้าให้แก่หลอดไฟเพื่อให้ได้ปริมาณความสว่างตามที่ต้องการ

ประเภทของ Dimmer

สามารถแบ่งได้ตามลักษณะการใช้งานได้ 2 ประเภทคือ

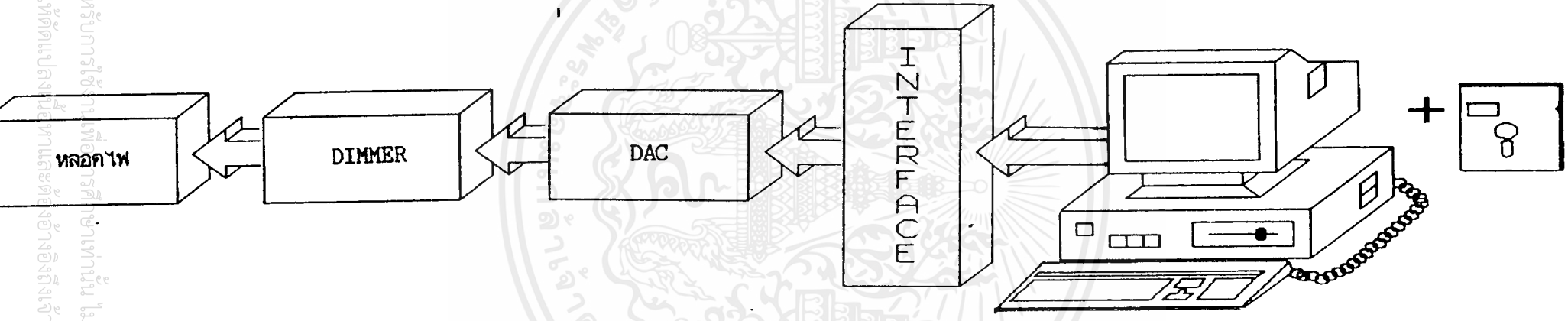
1. ARCHITECT TURAL DIMMER

   เป็น Dimmer ที่ใช้งานสถาปัตยกรรมเช่นในห้อง lobby ของโรงแรมในหอแสดงงานศิลปะ ห้องประชุม

2. Entertainment Dimmer

   เป็น Dimmer ที่ใช้ในงานเวทีต่าง ๆ เช่น เวทีคอนเสริต์ เวทีละคร สร้างบรรยากาศของแสงให้เข้ากับงานแสดงนั้น ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1.4 Block Diagram แสดงหลักการทำงานเบื้องต้นของ

Computer Control Dimmer

หลักการทางานเบื้องต้นของ Computer Control Dimmer

จาก Block Diagram Computer จะเป็นตัว Run โปรแกรม Control Dimmer โดยจะส่ง Output ออกทางการ์ด Interface ซึ่งใช้ 8255 เป็นตัวส่งข้อมูลแบบประยุกต์โดยใช้สายสัญญาณ 3 เส้น เส้นที่ 1 เป็น Data level and chanal select ส่วนเส้นที่ 2 เป็นสายสัญญาณ Clock และเส้นที่ 3 เป็น Enable (Control Latch) เพื่อเป็นการประหยัดสายในการส่งข้อมูล ซึ่งจะส่งผ่านมายังชุด Decoder เพื่อถอดรหัสจาก Serial to Parallel และทาการแยก Chanal และ Data level ออกจากกัน และส่ง Data level ไปสู่ภาค Digital to Analog เพื่อแปลงสัญญาณ Digital 8 บิต (256 Step) เป็นสัญญาณ Analog เปลี่ยนแปลง 0-10v แล้วส่งไปยังภาค Control ของ Dimmer เพื่อควบคุม load ต่อไป

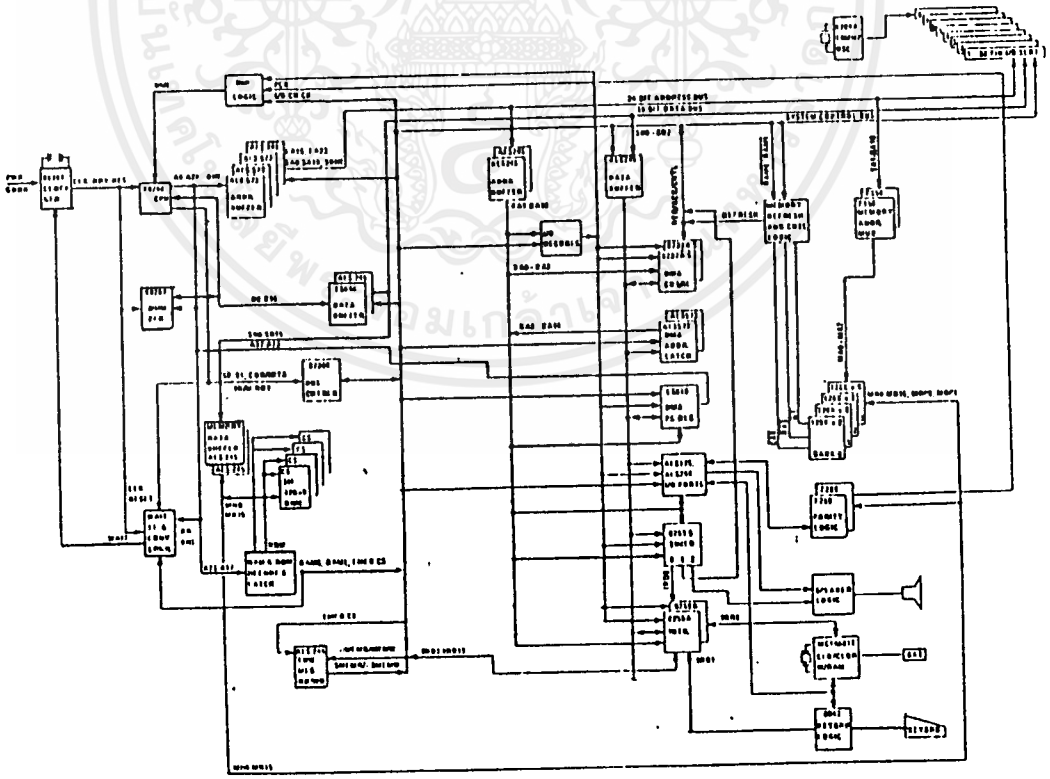
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

โครงสร้างของคอมพิวเตอร์ IBM-AT

2.1 เครื่องไอบีเอ็มเอที

เครื่องไอบีเอ็มเอทีเป็นผลของการพัฒนาพีซีของบริษัทไอบีเอ็มในปี พ.ศ. 2528 พัฒนาการของไอบีเอ็มเอทีในสมัยนั้นยังจัดได้ว่าเป็นการก้าวที่สำคัญในระดับพีซี ไอบีเอ็มเอทีใช้ซีพียู 80286 ทำงานที่ความถี่ของสัญญาณนาฬิกา 6 เมกะเฮิรตซ์ และทำงานร่วมโปรเซสเซอร์คณิตศาสตร์ 80287 ไอบีเอ็มเอที มีโครงสร้างระบบบัสเป็น 16 บิตเต็ม และมีส่วนขยายของระบบทางฮาร์ดแวร์ที่เพิ่มเติมจากไอบีเอ็มเอ็กซีทีหลายส่วน เพื่อให้เห็นรูปโครงของไอบีเอ็มเอที จึงขอสรุปโครงสร้างดังแสดงในรูปที่ 2.1



รูปที่ 2.1 บล็อกไดอะแกรมของเมนบอร์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบไมโครคอมพิวเตอร์พีซีเอทีนี้ เป็นการปรับปรุงเทคโนโลยีจากเดิม โดยให้ระบบเอ็กซ์ทีเดิมยังคงใช้งานได้เหมือนเดิม ด้วยเหตุนี้ซอฟต์แวร์ทุกระบบที่เคยใช้ได้กับพีซีเอ็กซ์ทีต้องใช้กับพีซีเอทีได้ โครงสร้างของระบบตามรูปที่ 2.2 ซึ่งอยู่บนเมนบอร์ดประกอบด้วย

ไมโครโปรเซสเซอร์เบอร์ 80286 ซึ่งเป็นไมโครโปรเซสเซอร์ที่มีระบบบัสข้อมูล 16 เส้น และแอดเดรส 24 เส้น การอ้างอิงแอดเดรสทำได้ถึง 16 เมกะไบต์ นอกจากนี้ยังทำงานร่วมกับโปรเซสเซอร์คณิตศาสตร์เบอร์ 80287 ได้อีกด้วย

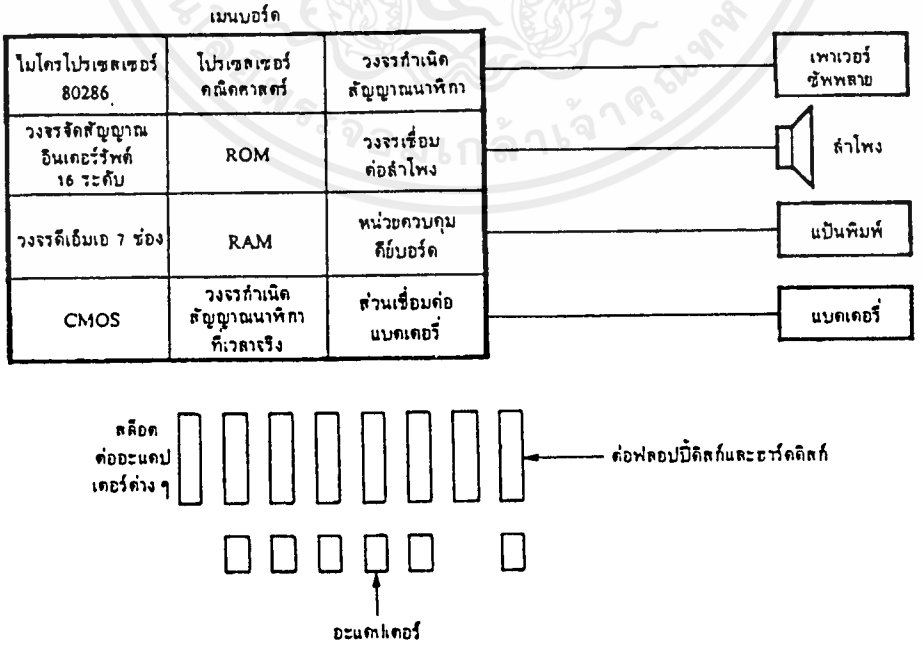
วงจรสนับสนุนระบบ ประกอบด้วย การจัดการดีเอ็มเอ ได้เพิ่มขยายช่องการทำดีเอ็มเอไปเป็น 7 ช่อง เพิ่มการจัดการอินเตอร์รัพต์จากเดิม 8 ระดับไปเป็น 16 ระดับ มีการจัดการระบบสัญญาณนาฬิกาที่โปรแกรมได้

ต่อเชื่อมรอม สำหรับเก็บโปรแกรมแบบถาวร เช่น ไบออสขนาด 64 กิโลไบต์ และเพิ่มขยายต่อเป็น 128 กิโลไบต์

ต่อหน่วยความจำแรม ระบบพีซีเอทีนี้ต่อกับหน่วยความจำแรมบนเมนบอร์ด 512 กิโลไบต์ หรือขยายต่อเป็น 1 เมกะไบต์ หรือมากกว่านั้น

วงจรควบคุมลำโพง สำหรับเปิด/ปิดเสียง ต่อกับลำโพงขนาดเล็ก

ซีมอสแรม เก็บข้อมูลระบบในพีซีเอ็กซ์ทีใช้สวิตช์สำหรับบอกการติดตั้งระบบในพีซีเอทีใช้หน่วยความจำซีมอสขนาด 64 ไบต์ เป็นตัวเก็บข้อมูลโดยมีแบตเตอรี่สำรองไว้



รูปที่ 2.2 โครงสร้างเมนบอร์ดของพีซีเอที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีชิปนาฬิกาทำงานตลอดเวลา นาฬิกานี้จะทำงานแม้ปิดเครื่อง เวลาของระบบจึงไม่ต้องเซตใหม่

มีวงจรเชื่อมกับแบตเตอรี่สำรอง ใช้สำหรับจ่ายให้กับซีมอสแรม และนาฬิกา

ต่อสล็อตได้ 8 สล็อต ซึ่งเป็นการเพิ่มสล็อตเป็นแบบเอทีเองโดยเพิ่มซ็อกเกตขนาด 36 ขา ต่อจากของเดิม 62 ขา ในระบบเอ็กซีทีออกมาอีก 6 สล็อต

เมนบอร์ดของพีซีเอทีที่กล่าวมานี้เมื่อเขียนแสดงโครงสร้างพื้นฐานทั้งหมดของวงจร-แสดงดังรูปที่ 2.2

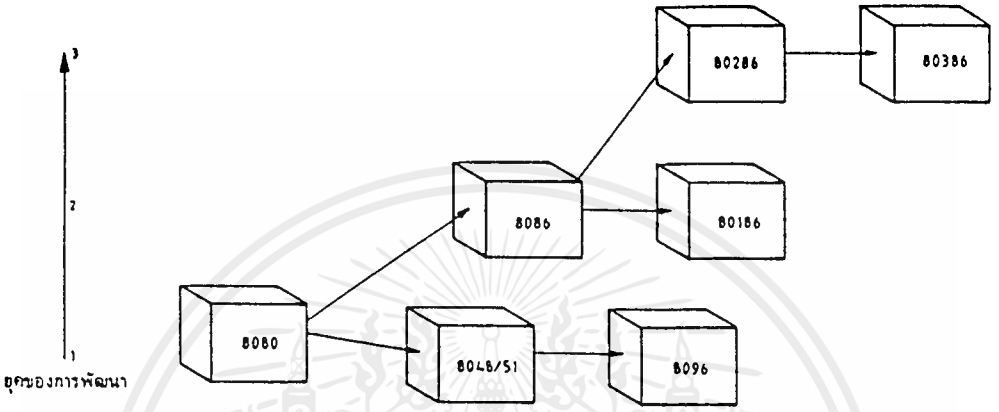
2.2 วงจรซีพียู

ซีพียู 80286 เป็นชิปในยุคที่สามของการพัฒนาไมโครโปรเซสเซอร์ของบริษัทอินเทล การพัฒนาไมโครโปรเซสเซอร์นั้น อินเทลได้ให้โครงสร้างการทำงานที่ใช้งานร่วมกันได้ขีดความสามารถที่เพิ่มขึ้นจึงเป็นซุปเปอร์เซตของเดิม ยุคของไมโครโปรเซสเซอร์ที่อินเทลพัฒนามาเป็นดังรูปที่ 2.3 ดังนั้นฐานของ 80286 จึงก้าวต่อเข้าสู่ 80386 ได้เป็นอย่างดี เพื่อที่จะเข้าใจระบบเมนบอร์ดของพีซีเอทีได้ดี ผู้เขียนขอเน้นในระบบพื้นฐานของตัว 80286 ก่อน เพื่อให้เห็นระบบบัสและการเชื่อมต่อกับอุปกรณ์อื่น ๆ ของ 80286 โดยปกติ 80286 จะทำงานร่วมกับชิปทำเนิดสัญญาณนาฬิกาคือ 82284 และชิปควบคุมบัส 82288 ดังนั้นแก่นของซีพียูจึงประกอบด้วยชิป 3 ตัว คือ 80286 82284 และ 82288 ประกอบรวมกันเข้าเป็นระบบเพื่อสร้างบัสเชื่อมโยงกับอุปกรณ์หน่วยความจำ หรืออุปกรณ์ประกอบจำพวกเพอริเฟอรัล หากเขียนระบบไมโครคอมพิวเตอร์พื้นฐานที่ใช้ซีพียู 80286 เป็นหลักจะได้ดังรูปที่ 2.4 โครงสร้างในรูปนี้จึงเป็นเสมือนโครงสร้างที่ระบบพีซีเอที จะต้องใช้ในการสร้างระบบบัสเพื่อเชื่อมต่อกับสิ่งต่าง ๆ

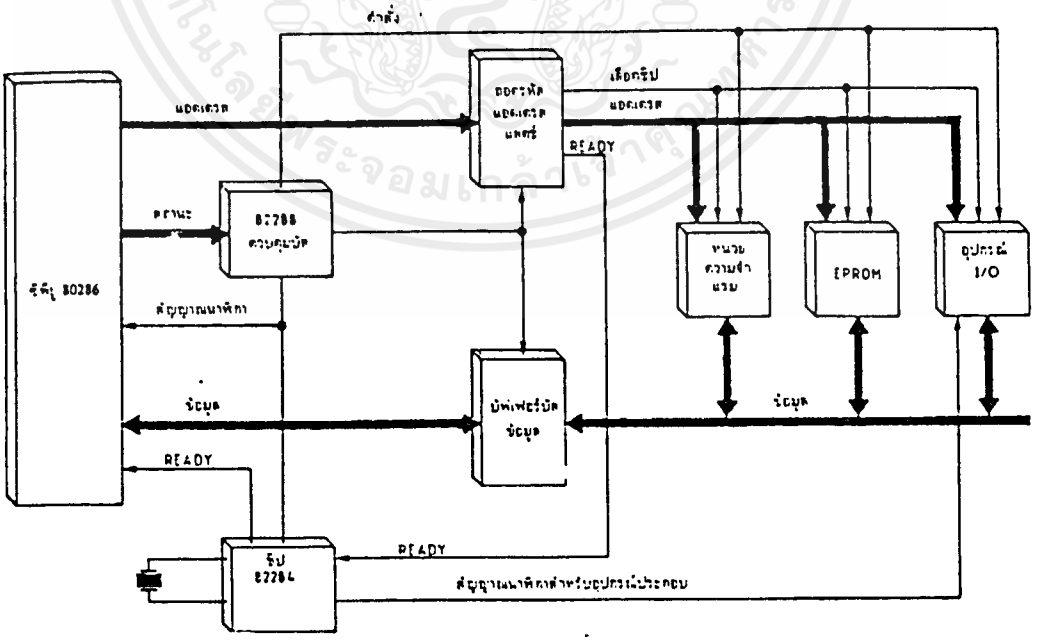
จากระบบไมโครคอมพิวเตอร์ทุกระบบที่จัดโครงสร้างเป็นบัส จะต้องประกอบด้วยบัสข้อมูล บัสแอดเดรสและสัญญาณสำหรับการควบคุมระบบต่าง ๆ ในระบบ 80286 ก็เช่นเดียวกันที่ต้องมีส่วนของการควบคุมระบบที่ประกอบด้วยบัสทั้งสามแบบนี้ ไมโครโปรเซสเซอร์ 80286 เป็นไอซีที่ต้องใช้สัญญาณนาฬิกาที่สร้างมาจากชิป 82284 ส่วนชิป 82288 เป็นชิปที่ทำการสร้างสัญญาณควบคุมระบบทั้งหมดก่อนที่จะต่อเข้ากับหน่วยความจำภายนอกแรมและรอม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากระบบดังกล่าวนี้เราจึงควรมารู้จักกับชิปทั้งสามนี้ก่อน ชิป 80286 เป็นชิปที่มีรูปร่างลักษณะชิปแบบ pin grid array ตัวชิปมีลักษณะเป็นตัวสี่เหลี่ยมจตุรัส มีขาทั้งสิ้น 68 ขา เรียงล้อมรอบทั้งสี่ด้านส่วน 82284 เป็นไอซีแบบดีนตะขาบ 18 ขาและ 82288 เป็นแบบ 20 ขา



รูปที่ 2.3 พัฒนาการของไมโครโปรเซสเซอร์ในกลุ่ม 80xxx

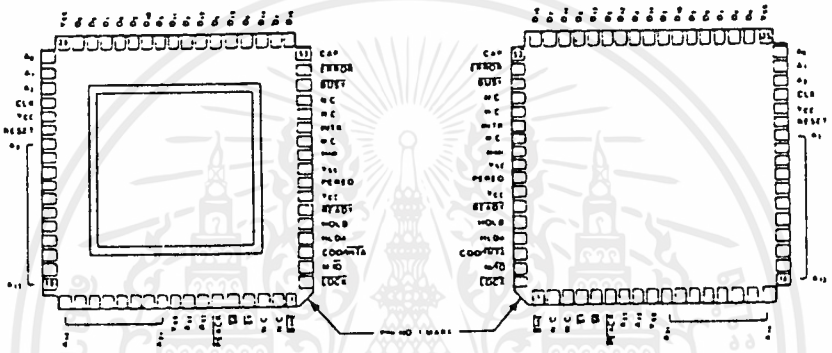


รูปที่ 2.4 ระบบไมโครคอมพิวเตอร์ที่มีซีพียูเบอร์ 80286

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

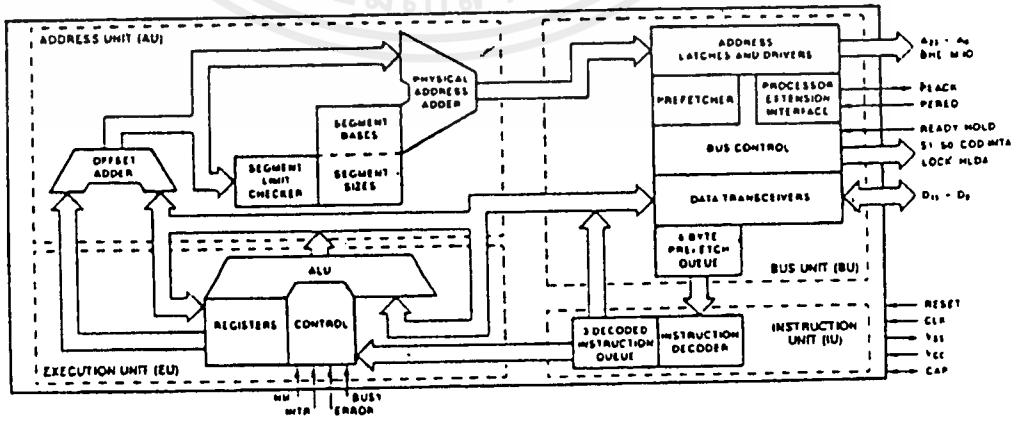
2.3 ไอซี 80286

ไอซีไมโครโปรเซสเซอร์ 80286 มีขีดความสามารถสูงกว่า 8088 ที่ใช้ในพีซีธรรมดามาก อินเทลเขียนบอกไว้ในคู่มือซิบว่ามีขีดความสามารถสูงกว่า 8086 มากกว่า 6 เท่า 80286 มี โครงสร้างภายในที่พัฒนามาให้มีสถาปัตยกรรมใช้กับตระกูล 8086 ได้ การจัดวางขาของไอซี 80286 แสดงดังรูปที่ 2.5



ก. มองทางด้านตัวอุปกรณ์ ข. มองทางด้าน PC บอร์ด

รูปที่ 2.5 การจัดเรียงขาของ 80286



รูปที่ 2.6 บล็อกไดอะแกรมภายในของ 80286

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ห้องสมุดคณะวิศวกรรมศาสตร์

สำหรับบล็อกไดอะแกรมภายในของ 80286 แสดงดังรูปที่ 2.6 ภายในแบ่งออกเป็นหน่วยจัดการแอดเดรส (AU-address unit) หน่วยการเอ็กซีคิวต์ (EU-execution unit) หน่วยควบคุมบัส (BU-bus unit) และหน่วยจัดการวงจรด้วยการถอดรหัสคำสั่ง (IU-instruction unit)

การทำงานของ 80286 ใช้สัญญาณที่ปรากฏที่ขาต่าง ๆ ในขั้นต้นนี้ลองมาทำความเข้าใจกับตัว 80286 ในฟังก์ชันการทำงานของแต่ละขาดูดังนี้

CLK เป็นอินพุตรับสัญญาณนาฬิกาจากภายนอกเข้ามา สัญญาณนี้จะได้รับการหารด้วย 2 เพื่อกำหนดเป็นสัญญาณนาฬิกาของระบบการหารสองจะกระทำภายในชิป และจะทำการซิงโครนัสกับการเริ่มต้นทำงานของสัญญาณรีเซต โดยเริ่มจาก "0" ไป "1" ของสัญญาณรีเซต

D_{15}-D_0 เป็นบัสสองทิศทาง จะทำหน้าที่เป็นอินพุตเมื่อสัญญาณขอการอ่านจากอินพุตเอาต์พุต จากหน่วยความจำหรือขณะตอบรับการอินเตอร์รัพต์ ส่วนจะเป็นเอาต์พุตขณะทำการเขียนข้อมูลให้กับอุปกรณ์เอาต์พุตหรือหน่วยความจำ

A_{23}-A_0 เป็นบัสแอดเดรส ทำหน้าที่เป็นเอาต์พุตแอดเดรสให้กับบัสแอดเดรส

BHE ชื่อสัญญาณ bus high enable เป็นสัญญาณเอาต์พุตบอกสถานะการติดต่อระหว่างซีพียูกับอุปกรณ์ภายนอกผ่านทางบัสข้อมูลในไบต์บน (D_{15}-D_8) การทำงานจะร่วมกับ A_0 ดังตารางที่ 2.1

ตารางที่ 2.1

BHE	A_0	ฟังก์ชัน
0	0	การติดต่อเป็นเวิร์ด
0	1	ติดต่อผ่านทาง 8 บิต D_{15}-D_8
1	0	ติดต่อผ่านทาง 8 บิต D_7-D_0
1	1	ไม่ได้ใช้

S_1, S_0 เป็นสัญญาณเอาต์พุตที่บอกสถานะของบัส ซึ่งต้องถอดรหัสร่วมกับสัญญาณเลือกระหว่างหน่วยความจำและไอโอ และสัญญาณ COD/INTA สัญญาณ S_1 S_0 เป็นลอจิกสามสถานะ และจะเป็นอิมพีแดนซ์สูง เมื่อเกิดการขอใช้บัส (bus acknowledge) เมื่อถอดรหัสจะได้ดังตารางที่ 2.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

03277

M/IO เป็นสัญญาณเอาต์พุตบอกสถานะการติดต่อระหว่างซีพียูกับอินพุต เอาต์พุต (I/O) หรือหน่วยความจำ

COD/INTA เป็นสัญญาณเอาต์พุต บอกการเฟตช์หรือการตอบสนองต่ออินเตอร์รัพต์

LOCK เป็นสัญญาเอาต์พุต เพื่อบอกสถานะการล็อกของบัสเอาไว้ใช้งาน

READY เป็นอินพุต เพื่อหยุดทำงานหรือยืดไซเคิลของบัสออกไปอีก เพื่อการซิงโครนัสกับอุปกรณ์ที่ทำงานได้ช้ากว่าซีพียู

HOLD เป็นอินพุต เป็นสัญญาณการขอใช้บัสจากภายนอก

ตารางที่ 2.2

COD/ $\overline{INTA}$	M/ $\overline{S_0}$	$\overline{S_1}$	$\overline{S_0}$	ไซเคิลของบัสเป็น
0	0	0	0	ตอบสนองอินเตอร์รัพต์ (INTA)
0	0	0	1	*สงวนไว้
0	0	1	0	สงวนไว้
0	0	1	1	ไม่ใช้
0	1	0	0	ถ้า A_1 = 1 จะเป็น Halt ถ้าไม่อยู่ในสถานะ shut down
0	1	0	1	อ่านจากหน่วยความจำ
0	1	1	0	เขียนหน่วยความจำ
0	1	1	1	ไม่ใช้
1	0	0	0	สงวนไว้
1	0	0	1	อ่านจาก I/O
1	0	1	0	เขียน I/O
1	0	1	1	ไม่ใช้
1	1	0	0	สงวนไว้
1	1	0	1	เฟตช์
1	1	1	0	สงวนไว้
1	1	1	1	ไม่ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

HOLA เป็นเอาต์พุต เป็นสัญญาณการตอบรับให้ใช้บัสจากภายนอก เช่นในสภาวะ DMA

INTR เป็นอินพุต เป็นการขออินเตอร์รัพต์ซีพียู

PEREQ เป็นอินพุต มีชื่อว่า processor extension operand request ทำหน้าที่ในการจัดการหน่วยความจำในการขยายหน่วยความจำออกไป

PEACK เป็นเอาต์พุต เป็นสัญญาณตอบสนองของ PEREQ

BUSY เป็นสัญญาณอินพุต มีชื่อว่า processor extension busy เป็นสัญญาณที่ใช้ต่อร่วมกับการทำงานภายนอก เช่น 80287

ERROR เป็นสัญญาณอินพุต มีชื่อว่า processor extension error เป็นระบบการทำงานจากโปรเซสเซอร์ภายนอก

RESET เป็นสัญญาณอินพุต สำหรับการรีเซตซีพียู

V_{SS} แหล่งจ่ายไฟ 0 โวลต์

V_{CC} แหล่งจ่ายไฟ 5 โวลต์

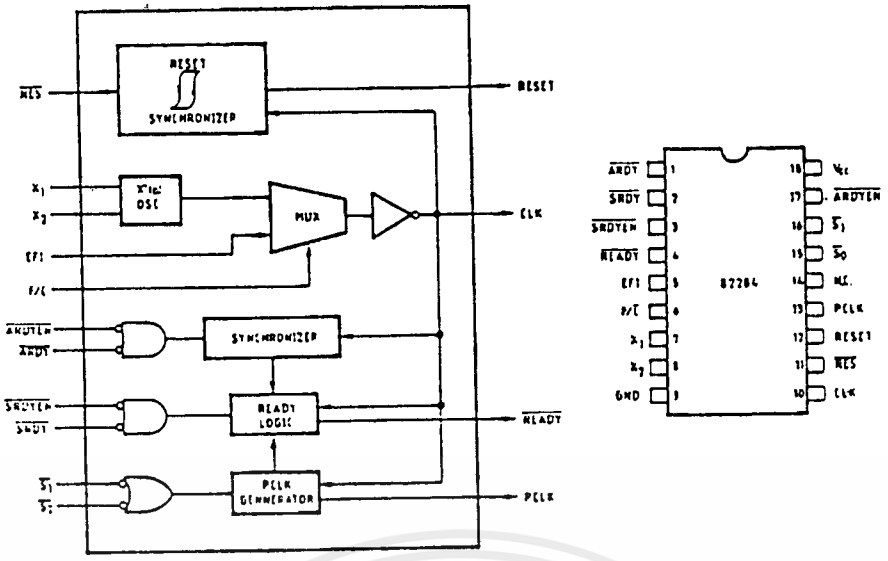
CAP ต่อกับตัวเก็บประจุลงกราวนด์ ค่าตัวเก็บประจุใช้ขนาด 0.047 ไมโครฟาราด

2.4 ชิป 82284

ชิป 82284 เป็นชิปสำหรับสร้างสัญญาณนาฬิกาให้กับ 80286 เช่น ถ้าใช้คริสตอล 16 เมกะเฮิรตซ์ ใส่ให้กับ 82284 82284 จะสร้างสัญญาณนาฬิกาขนาด 16 เมกะเฮิรตซ์ให้กับ 80286 ส่วนของ 80286 จะมีวงจรหารสองทำการลดความถี่ 16 เมกะเฮิรตซ์ให้เหลือ 8 เมกะเฮิรตซ์ และจัดว่าเป็นสัญญาณนาฬิกาของระบบ เดิมบริษัทไอบีเอ็มได้ใช้คริสตอล 12 เมกะเฮิรตซ์ จึงใช้กับความเร็วของสัญญาณนาฬิกาในระบบเป็น 6 เมกะเฮิรตซ์ แต่เครื่องที่ทำคล้ายคลึงส่วนใหญ่จะใช้สัญญาณนาฬิกาสูงกว่า ในปัจจุบันรุ่นเอทีจะใช้สัญญาณนาฬิการะบบที่ 10,12 หรือ 16 เมกะเฮิรตซ์ นั่นคือต้องใช้ความถี่คริสตอลสูงเป็นสองเท่า

ชิปและบล็อกไดอะแกรมของ 82284 โดยสัญญาณต่าง ๆ นี้อินเทลได้ออกแบบมาให้ใช้กับ 80286 พอดี ดังรูปที่ 2.7

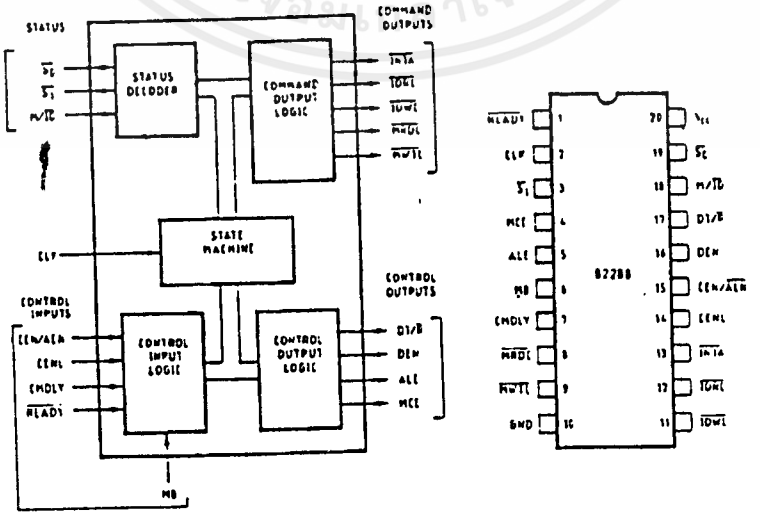
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 การจัดทางขา 82284 และวงจรบล็อกไดอะแกรม

2.5 ชิป 82288 ตัวควบคุมบัส

หากให้สัญญาณนาฬิกาของระบบเป็น 8 เมกะเฮิรตซ์ 80286 สามารถทำงานได้ประมาณ 1.5 ล้านคำสั่งในเวลา 1 วินาที หรือมีความเร็วขนาด 1.5 MIPS การที่ 80286 จะทำงานควบคุมติดต่อกับหน่วยความจำ หรือ อินพุต-เอาต์พุตจำเป็นต้องแยกสัญญาณควบคุมออกมาให้อย่างชัดเจนก่อน อินเทลได้ออกชิป 82288 เป็นชิปสำหรับควบคุมบัส เพื่อแยกสัญญาณควบคุมต่างๆ ดังจะกล่าวต่อไปออกมา ชิป 82288 บล็อกไดอะแกรมแสดงไว้ดังรูปที่ 2.8



รูปที่ 2.8 ชิป 82288 และบล็อกไดอะแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซีพียูจะควบคุมฮาร์ดแวร์ระบบอื่น ๆ ได้นั้นจะต้องสร้างไซเคิลของบัสขึ้นมา แต่ละไซเคิลจะแทนการทำงานอย่างใดอย่างหนึ่งขึ้นอยู่กับซีพียูขณะนั้น ว่าได้รับคำสั่งอะไรเข้ามา ซีพียูไซเคิลพื้นฐานประกอบด้วย

- ไซเคิลการอ่านหน่วยความจำ
- ไซเคิลการเขียนหน่วยความจำ
- การอ่านอินพุต
- การเขียนเอาต์พุต
- การตอบสนองต่ออินเตอร์รัพต์

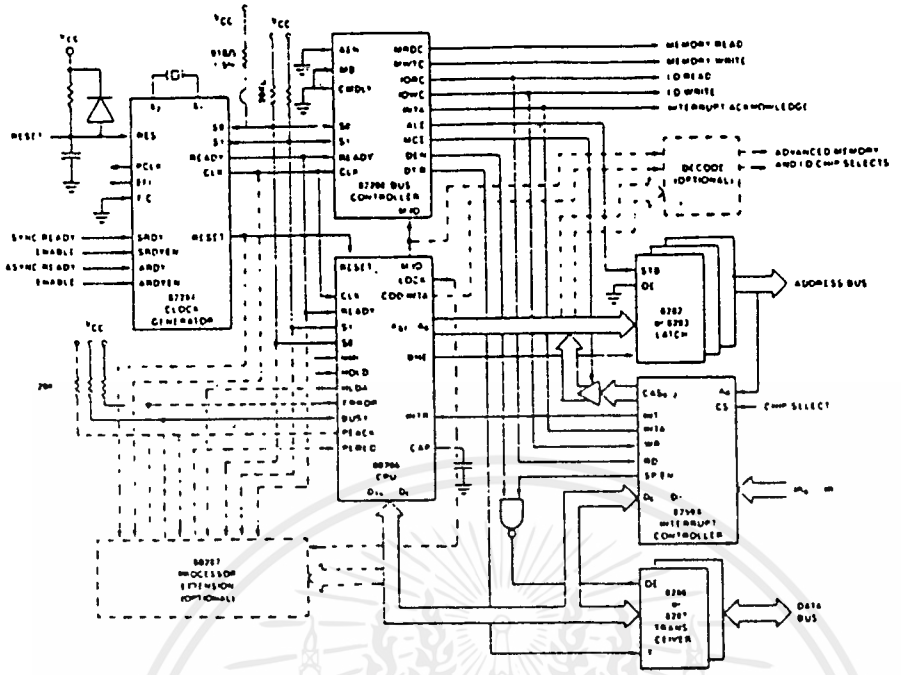
2.6 การต่อ 80286 82288 และ 82284 แบบพื้นฐาน

การทำงานเป็นระบบของ 80286 ต้องประกอบชิปทั้งสามชิปนี้ประกอบกันบนพีซีเอทีก็มีการต่อชิปทั้งสามนี้เป็นระบบเช่นเดียวกัน 82288 จะอยู่ในตำแหน่งระหว่างซีพียูกับระบบที่จะต่อเชื่อมกับซีพียู 82288 จะทำการแปลงสัญญาณ S_0 S_1 M/IO เพื่อใช้ในการทำงานกับระบบ

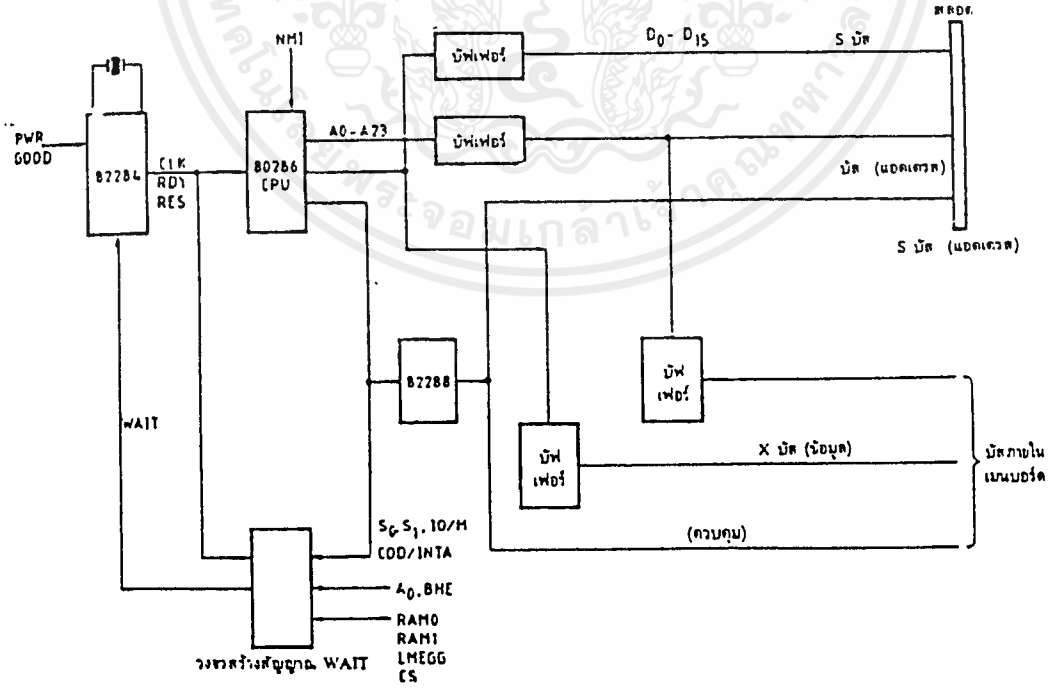
สำหรับในซีพีเอทีการต่อซีพียูกับชิป 82284 ก็มีลักษณะเดียวกัน บนส่วนของ 82284 นี้ ขารีเซตมาจากสัญญา POWER GOOD ซึ่งแอกทีฟ "1" นั้นคือ ถ้าหากมีปัญหาทางด้านแหล่งจ่ายไฟเลี้ยงจะทำให้ 82284 หยุดจ่ายสัญญาณนาฬิกาให้กับ 80286 วงจรส่วนของการสร้างบัสเพื่อต่อกับอุปกรณ์ภายนอกของพีซีเอทีมีด้วยกัน 4 แผ่น ดังที่แสดงในตอนท้ายบทนี้ อนึ่งวงจรทั้งหมดของซีพีเอทีเฉพาะส่วนเมนบอร์ดมีทั้งหมด 22 แผ่น ซึ่งประกอบต่อกัน ในตอนนี้จะขอเน้นเฉพาะส่วนของซีพียูก่อน

สำหรับวงจรพื้นฐานการต่อระหว่าง 80286 กับ 80288 และ 80284 และโครงสร้างระบบบัสของพีซีเอที แสดงดังรูปที่ 2.9 และรูปที่ 2.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 วงจรพื้นฐานการต่อระหว่าง 80286 กับ 80288 และ 80284



รูปที่ 2.10 โครงสร้างระบบบัสของพีซีเอที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.11 วงจรแผ่นที่ 1 เป็นการต่อ 82284 กับ 80286 โดยมี 80287 ต่อร่วมอยู่ด้วยสัญญาณเอาต์พุตของแผ่นนี้ส่วนใหญ่เป็นสัญญาณบัสของระบบ

รูปที่ 2.12 วงจรแผ่นที่ 2 เป็นการต่อ 82288 เข้ากับระบบเพื่อถอดรหัสสัญญาณควบคุมระบบทั้งหมดคือสัญญาณการอ่านหน่วยความจำ การเขียนหน่วยความจำ การเขียนอ่านอินพุต/เอาต์พุต การตอบสนองต่ออินเตอร์รัพท์ และการแลตช์แอดเดรสในแผ่นที่ 2 นี้มีจุดที่น่าสนใจคือการใส่บัฟเฟอร์ให้กับบัสข้อมูล ในที่นี้ใช้ไอซี LS646 เป็นบัฟเฟอร์สำหรับข้อมูลไบต์ต่ำ (D_0-D_7) และ LS245 เป็นบัฟเฟอร์ข้อมูลทางไบต์สูง (D_8-D_{15}) สิ่งที่สนใจคือวงจรถอดรหัสที่ใช้ 74F10 หรือแนนเกต 3 อินพุต เพื่อเลือกการติดต่อข้อมูลเป็นเวิร์ดหรือเป็นไบต์ ทั้งไบต์บนและไบต์ล่างทั้งนี้เพราะซีพียูมีสัญญาณการอีนาเบิลไบต์สูง BHE และ A_0 ร่วมกันในการคัดเลือกการติดต่อกับบัสข้อมูล สำหรับสัญญาณบางส่วนที่ปรากฏเป็นสัญญาณที่จะไปปรากฏที่สล็อต

อนึ่ง 82288 มีสัญญาณ DT/R เป็นตัวบอกทิศทางของการรับส่งข้อมูลในบัสข้อมูลซึ่งนำมาใช้กำหนดในตัวบัฟเฟอร์ข้อมูลนี้โดยตรง

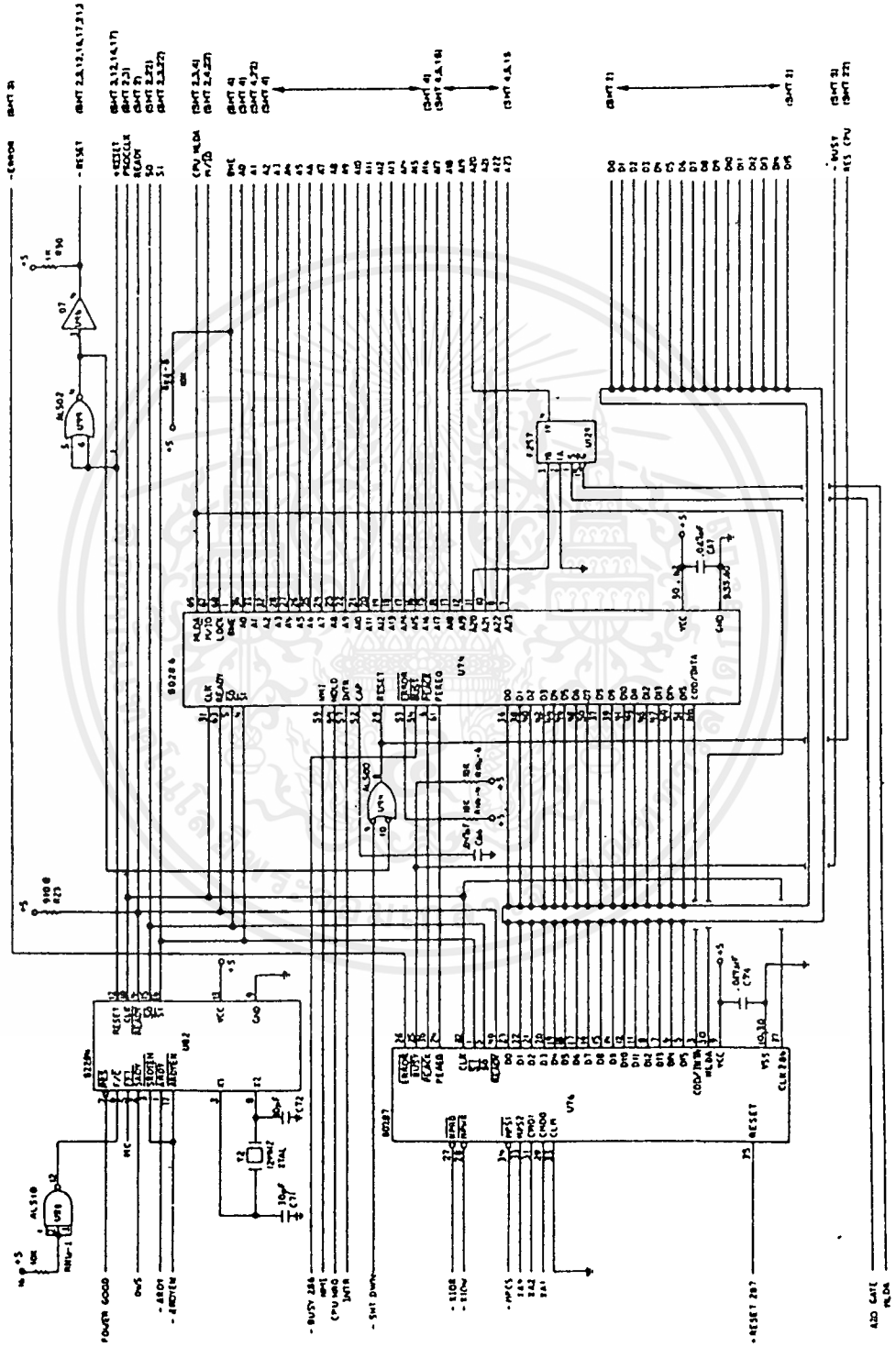
รูปที่ 2.13 วงจรแผ่นที่ 3 เป็นการสร้างสัญญาณควบคุมระบบเฉพาะต่าง ๆ สัญญาณควบคุมส่วนนี้ จะได้นำมาอธิบายต่อไป สิ่งที่น่าสังเกตคือ มีการใช้ PAL (programmble array logic) ทำการถอดรหัสสัญญาณต่าง ๆ เพื่อการสร้างสัญญาณควบคุมระบบขึ้น

ส่วนในรูปที่ 2.14 วงจรแผ่นที่ 4 เป็นวงจรบัฟเฟอร์ของแอดเดรสบัส สำหรับเชื่อมต่อกับสล็อต โครงสร้างของการเชื่อมแอดเดรสเข้ากับสล็อตนี้ การจัดการบัสของระบบพีซีเอทีนี้แยกบัสเป็นสองชุด ชุดหนึ่งเมื่อผ่านบัฟเฟอร์ก็จะส่งต่อไปยังช่องต่อสล็อต ส่วนบัสที่ใช้ในเมนบอร์ดเราเรียกว่า X บัส ส่วนของ X บัสนี้จะเป็นบัสที่ผ่านบัฟเฟอร์อีกชุดหนึ่ง สำหรับใช้ภายในเมนบอร์ด

2.7 พอร์ตอินพุตและเอาต์พุตของพีซีเอที โครงสร้างการวางพอร์ตอินพุตและเอาต์พุต

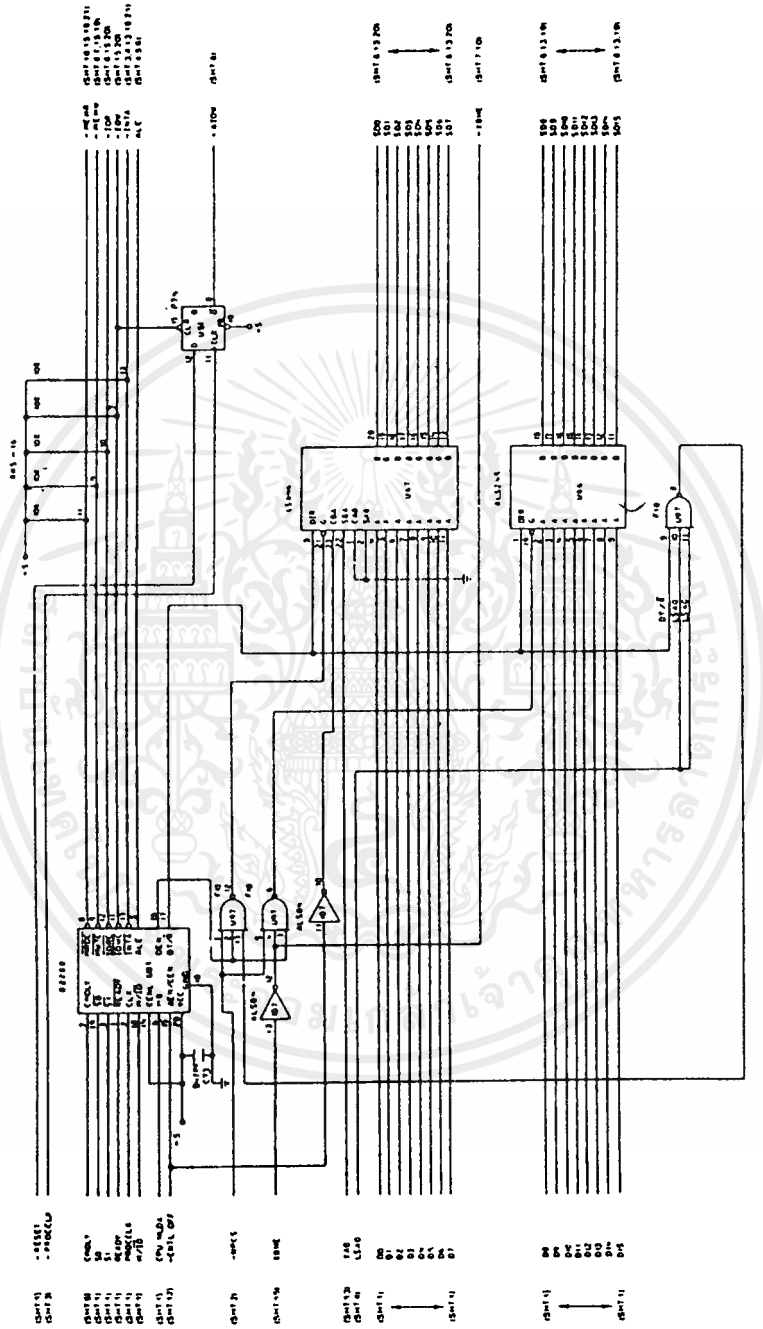
เพื่อให้โครงสร้างทางฮาร์ดแวร์ของพีซีเอทีใช้ซอฟต์แวร์ร่วมกับพีซีเอ็กซ์ทีได้ จำเป็นต้องให้โครงสร้างอินพุตและเอาต์พุตเหมือนกัน หมายเลขพอร์ตที่ใช้ของพีซีเอทีแสดงดังตารางที่ 2.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



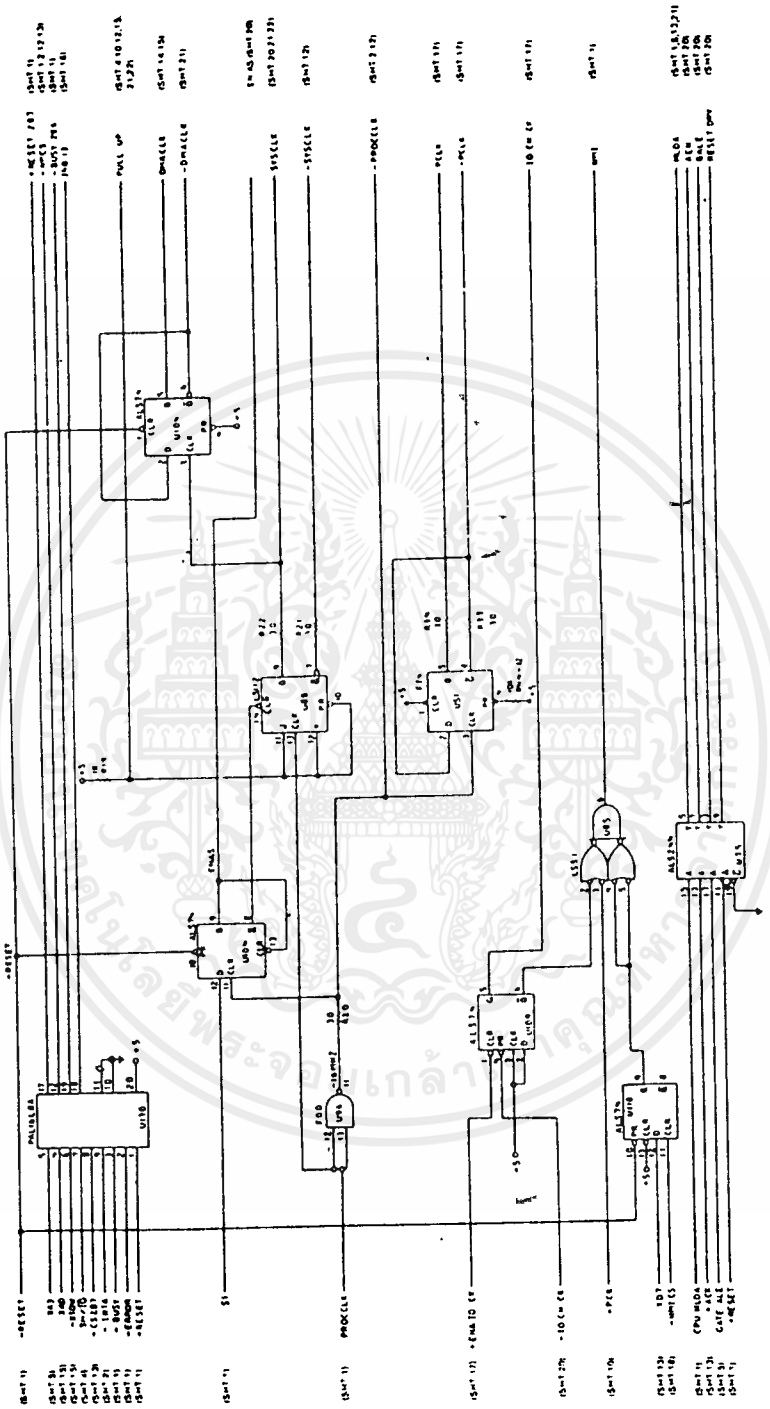
รูปที่ 2.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



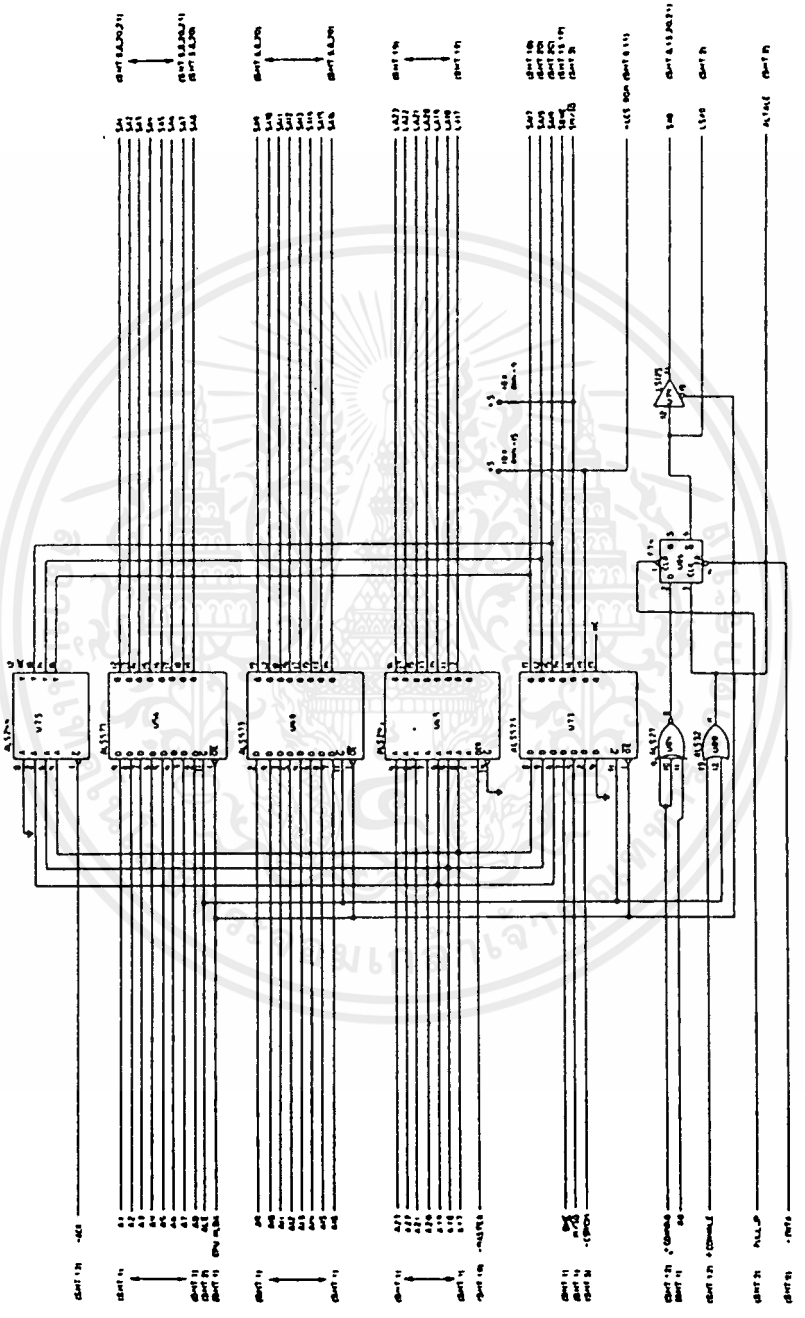
รูปที่ 2.12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.3 หมายเลขพอร์ตที่ใช้กับพีซีเอที

หมายเลขพอร์ตรหัสฐานสิบหก	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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8 วงจรไทเมอร์

ไทเมอร์หรือวงจรกำหนดเวลาของระบบสำหรับไมโครคอมพิวเตอร์เอทีแตกต่างจากพีซีเอ็กซ์ทีบ้างเล็กน้อย ในพีซีเอ็กซ์ทีใช้ชิป 8253 แต่สำหรับพีซีเอทีใช้ชิป 8254-2 วงจรที่แสดงส่วนของการต่อชิปนี้แสดงไว้ดังรูปที่ 2.19 วงจรแผ่นที่ 9 โครงสร้างของชิป 8254 นี้มีวงจรตั้งเวลาอยู่สามช่อง โดยทั้งสามช่องใช้สัญญาณกำหนดอินพุตหรือที่เรียกว่า CLK IN ด้วยความถี่ 1.19 MHz แต่ละช่องมีการเชื่อมต่อระหว่างอินพุตกับสัญญาณดังนี้

สัญญาณเชื่อมต่อของช่อง 0

GATE0 เป็นขาที่เชื่อมต่อกับแรงดัน V_{cc}

CLK IN0 เป็นสัญญาณอินพุต 1.19 MHz

CLK OUT0 เป็นสัญญาณที่ส่งไปอินเตอร์รัพต์ทางช่อง 0 ของ 8259A

สัญญาณเชื่อมต่อของช่อง 1

GATE1 ต่อเข้ากับ V_{cc}

CLK IN1 เป็นสัญญาณอินพุต 1.19 MHz

CLK OUT1 เป็นการต่อเข้ากับวงจรที่ใช้ในการกำหนดรอบการรีเฟรชสัญญาณช่อง 1 ที่เอาต์พุตนี้จะได้รับการโปรแกรมให้ส่งไปทำการรีเฟรชด้วยคาบเวลาประมาณ 15 ไมโครวินาที

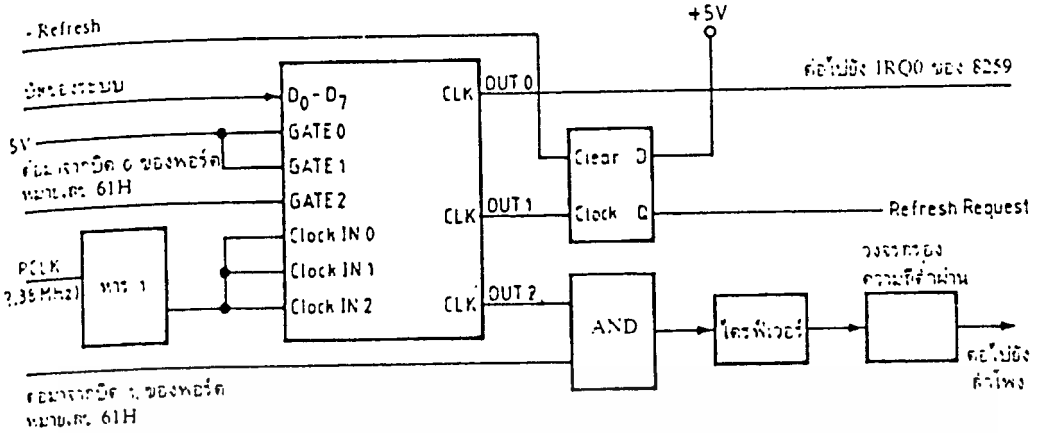
สัญญาณเชื่อมต่อของช่อง 2

GATE2 ถูกควบคุมโดยบิต 0 ของพอร์ตหมายเลข 61H ของสัญญาณควบคุมชิปไมโครคอมพิวเตอร์ 8042 เพื่อใช้ในการเปิดปิดสัญญาณเอาต์พุต

CLK IN2 ต่อกับสัญญาณอินพุตขนาด 1.19 MHz

CLK OUT2 ใช้เป็นสัญญาณเอาต์พุตในการขับลำโพง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.15 บล็อกไดอะแกรมของ 8254

2.8 อินเตอร์รัพต์

ผู้ออกแบบอินเตอร์รัพต์สำหรับเครื่องพีซี เอ็กซ์ทีได้วางโครงสร้างของการอินเตอร์-รัพต์ไว้ 8 ระดับ โดยใช้ไอซี 8259A โดยมี IRQ2 ว่างไว้ ครั้นมาออกแบบพีซี เอทีจึงจำเป็นต้องใช้อินเตอร์รัพต์เพิ่มเติมไปจากเดิม จึงต้องขยายอินเตอร์รัพต์เพิ่มเติมออกมาอีก โดยจัดวางอินเตอร์รัพต์ตามระดับความสำคัญดังตารางที่ 2.4

สังเกตุว่าการขยายอินเตอร์รัพต์ออกมาอีก 8 ช่องนี้ใช้สัญญาณ CTRL2 เป็นตัวส่งให้กับ IRQ2 ของ 8259A ตัวแรก ดังนั้นวงจรที่ใช้จึงใช้ 8259 สองตัว โดยให้เอาต์พุตของตัวหนึ่งเข้าที่ขา IRQ2 ของอีกตัวหนึ่ง วิธีการขยายแบบนี้ทำให้ระดับความสำคัญของ IRQ8-IRQ15 มีความสำคัญอยู่ในระดับ 2 เดิม หรือแทรก 8 ระดับอยู่ระหว่างระดับ 2 และ 3 เดิม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.4 การวางอินเตอร์รัพต์ตามระดับความสำคัญ

ระดับความสำคัญ	หน้าที่
นอนมาสเคเบิล NMI	รับสัญญาณจากตรวจสอบพาริตี้ และ ตรวจสอบแชนเนลไอโอ
IRQ0	ไทเมอร์เอาต์พุต 0
IRQ1	คีย์บอร์ด (เมื่อบัฟเฟอร์เต็ม)
IRQ2	อินเตอร์รัพต์มาจาก CTRL 2
IRQ8	สัญญาณกำหนดเวลา
IRQ9	เรียกมาจากซอฟต์แวร์ INT 0AH หรือ IRQ2 เดิม
IRQ10	สงวนไว้
IRQ11	สงวนไว้
IRQ12	สงวนไว้
IRQ13	โปรเซสเซอร์ร่วม (coprocessor)
IRQ14	จากแผงควบคุมฮาร์ดดิสก์
IRQ15	สงวนไว้
IRQ3	พอร์ตอนุกรม 2
IRQ4	พอร์ตอนุกรม 1
IRQ5	วงจรควบคุมฮาร์ดดิสก์
IRQ6	วงจรควบคุมดิสก์
IRQ7	พอร์ตขนาน 1

การจัดหมายเลขพอร์ตนั้นเป็นตามข้อกำหนดเดิมคือให้ 8259A ตัวแรก มีหมายเลขพอร์ตเป็น 020-03F และอีกตัวหนึ่งมีหมายเลข 0A0-0BF วงจรของการต่อ 8259 ทั้งสองตัว แสดงดังรูปที่ 2.20 วงจรแผ่นที่ 9

2.9 ดีเอ็มเอ

ดีเอ็มเอที่ใช้กับเครื่องพีซีเอ็กซ์ที ใช้ชิป 8237A เพียงชิปเดียว แต่สำหรับเอทีนี้ได้เพิ่มขยายช่องของดีเอ็มเอ ซึ่งจากเดิมมี 4 ช่อง ให้เป็น 8 ช่อง ดังนั้นจึงต้องใช้ชิป 8237A เพิ่มขึ้นอีกหนึ่งตัว โดยมีการเปลี่ยนแปลงโครงสร้างบ้างเล็กน้อย ช่องของดีเอ็มเอเดิมมี 4 ช่องคือ แชนเนล 0-3 และในเครื่องพีซีเอทีได้เพิ่มขยายช่อง 4-7 เพิ่มเติมจากเดิม จึงมีทั้งหมด 8 ช่อง โดยแต่ละช่องทำหน้าที่ดังตารางที่ 2.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อให้แชนเนลดีเอ็มเอเดิม 0-3 ทำงานเหมือนเดิม แต่เมื่อเครื่องพีซีเอทีไม่ต้องการรีเฟรชจากดีเอ็มเอ ดังนั้นแชนเนล 0 จึงว่างไว้ ดีเอ็มเอคอนโทรลเลอร์ตัวที่ 1 สนับสนุนการรับส่งข้อมูลแบบ 8 บิตระหว่างอินพุต เอาต์พุต กับหน่วยความจำของระบบ ในขณะที่ถ้าให้รับส่งกับหน่วยความจำและรับส่งแบบ 16 บิตได้ ในแต่ละแชนเนลถูกกำหนดด้วยโปรแกรมที่ระบบจะส่งไป การโปรแกรมก็โปรแกรมให้เกิดการส่งถ่ายข้อมูลทีละบล็อกด้วยขนาดบล็อกที่ใหญ่ที่สุด 64 กิโลไบต์ โดยติดต่อกับหน่วยความจำได้ตลอดสายแอดเดรส 24 เส้น หรือ 16 MB

ตารางที่ 2.5 รายละเอียดหน้าที่ช่องของดีเอ็มเอ

8237 ตัวแรก	หน้าที่
แชนเนล 0	สงวนไว้ (แต่เดิมของเครื่องพีซีเอ็กซ์ทีใช้สำหรับรีเฟรช)
แชนเนล 1	ใช้สำหรับการส่งผ่านกับ SDLC
แชนเนล 2	ใช้สำหรับการดิสค์อรับ ส่งข้อมูลกับดิสเกตต์
แชนเนล 3	สงวนไว้
8237 ตัวที่สอง	
แชนเนล 4	ต่อร่วมสำหรับชิป 8237 ตัวแรก
แชนเนล 5	สงวนไว้
แชนเนล 6	สงวนไว้
แชนเนล 7	สงวนไว้

สำหรับดีเอ็มเอคอนโทรลเลอร์ตัวที่สอง มีช่องการส่งรับข้อมูลคือ แชนเนล 4 ถึง แชนเนล 7 แชนเนล 4 ใช้สำหรับการต่อร่วม (cascade) กับแชนเนล 0-3 แชนเนล 4-7 นี้ได้รับการออกแบบให้ส่งถ่ายข้อมูลแบบ 16 บิตระหว่างอุปกรณ์อินพุตเอาต์พุตกับหน่วยความจำระบบโดยกำหนดขนาดข้อมูลเป็นบล็อกได้สูงสุดถึง บล็อกละ 128 กิโลไบต์ แต่อย่างไรก็ตาม ช่อง 5,6,7 ที่จะใช้งานนั้นตัววงจรได้ใช้ขา XA1 ต่อจากแอดเดรส ขา A0 ของ 8237 ทำให้การส่งถ่ายข้อมูลจะไม่สามารถส่งถ่ายในลักษณะเริ่มจากไบต์คี่ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัววงจรของ 8237 แสดงไว้ในรูปที่ 2.18 วงจรแผ่นที่ 7 การกำหนดแอดเดรสอินพุต-เอาต์พุตที่เกี่ยวข้องกับดีเอ็มเอทั้งสองชุดคือ แอดเดรส 000-01F ติดต่อกับดีเอ็มเอ โดยมีช่วงแอดเดรส 080-09F เป็นอินพุตเอาต์พุตแอดเดรสของเพจรีจิสเตอร์ และอินพุตเอาต์พุตของดีเอ็มเอคอนโทรลเลอร์ของตัวที่สองอยู่ที่แอดเดรส 0C0-0DF

การกำหนดหมายเลขแอดเดรสของพอร์ตอินพุตเอาต์พุตทำบนชิป 74ALS138 ซึ่งอยู่ในรูปที่ 12.17 วงจรแผ่นที่ 6 โดยเอาต์พุตกำหนดกลุ่มแอดเดรสของดีเอ็มเอ อินเตอร์รัพต์คอนโทรลเลอร์ไทเมอร์ พอร์ตขยาย สำหรับการกำหนดสถานะ (ในที่นี้ใช้ชิปไมโครคอมพิวเตอร์ชิปเดี่ยว 8042) พอร์ตกำหนดเพจรีจิสเตอร์และโปรเซสเซอร์คณิตศาสตร์

ชิป 74ALS138 ที่ถอดรหัสเป็นสัญญาณ PG REGCS นี้ส่งต่อไปยังชิป LS612 เพื่อกำหนดเพจรีจิสเตอร์ให้ทำงานร่วมกับดีเอ็มเอคอนโทรลเลอร์ในการกำหนดว่าการดีเอ็มเอจะเริ่มที่เพจใด โดยแต่ละเพจจะมีขนาดใหญ่ได้ถึง 64 กิโลไบต์ ดังนั้น LS612 จึงให้แอดเดรสของการดีเอ็มเอ ด้วยแอดเดรสตั้งแต่ A16-A23 นั่นหมายความว่าการกำหนดตำแหน่งหน่วยความจำสำหรับการดีเอ็มเอใด ๆ ต้องกำหนดแอดเดรสในบิต A16-A23 หรือเพจแอดเดรสนี้ด้วย วงจรการใช้ LS612 แสดงไว้ในรูปที่ 2.19 วงจรแผ่นที่ 8

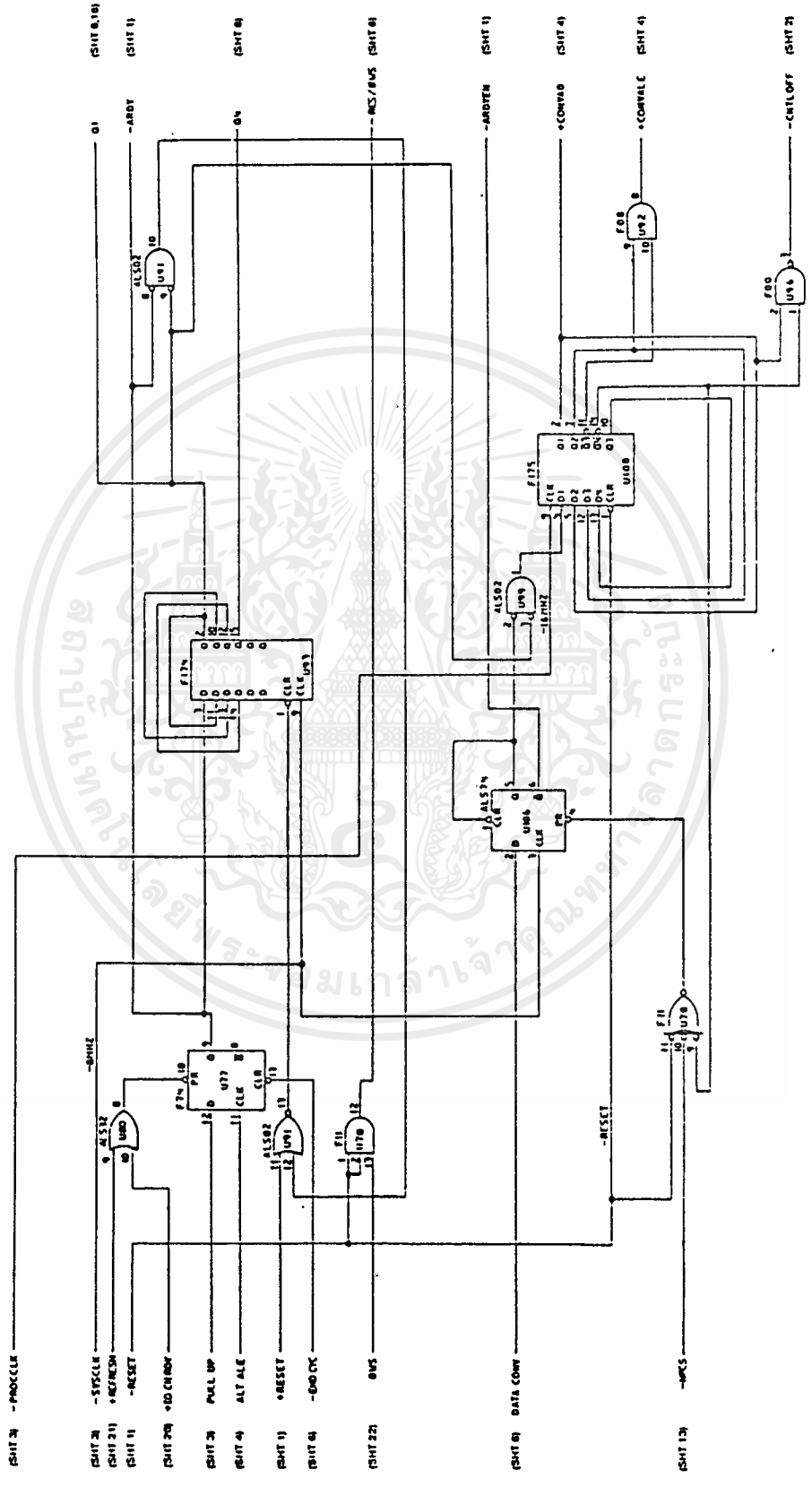
ผู้ใช้สามารถกำหนดค่าแอดเดรส A16-A23 ให้กับชิป LS612 นี้โดยผ่านทางพอร์ตหมายเลข 080-08F โดยแต่ละหมายเลขของพอร์ตมีความหมายดังต่อไปนี้

ตารางที่ 2.5 หมายเลขพอร์ต ของแต่ละแชนเนล

ดีเอ็มเอแชนเนล 0	ใช้พอร์ตหมายเลข 0087
ดีเอ็มเอแชนเนล 1	ใช้พอร์ตหมายเลข 0083
ดีเอ็มเอแชนเนล 2	ใช้พอร์ตหมายเลข 0081
ดีเอ็มเอแชนเนล 3	ใช้พอร์ตหมายเลข 0082
ดีเอ็มเอแชนเนล 5	ใช้พอร์ตหมายเลข 008A
ดีเอ็มเอแชนเนล 6	ใช้พอร์ตหมายเลข 0089
ดีเอ็มเอแชนเนล 7	ใช้พอร์ตหมายเลข 008B
รีเฟรชแอดเดรส	ใช้พอร์ตหมายเลข 008F

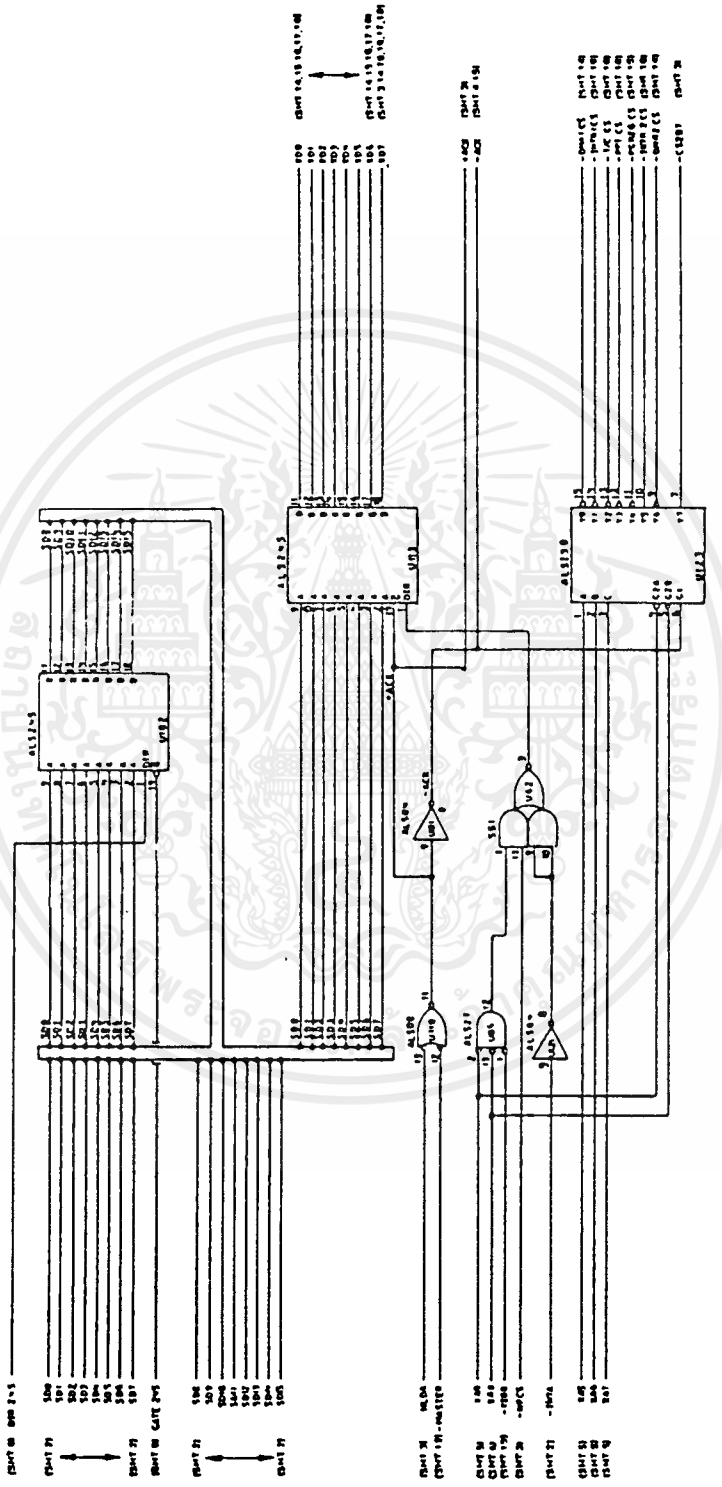
ในการกำหนดแอดเดรสเพื่อทำการดีเอ็มเอ ผู้ใช้ต้องกำหนดแอดเดรสเริ่มต้นให้กับ 8237A และ LS612 ด้วย หากกำหนดให้กับดีเอ็มเอแชนเนล 0-3 การกำหนดจะกระทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



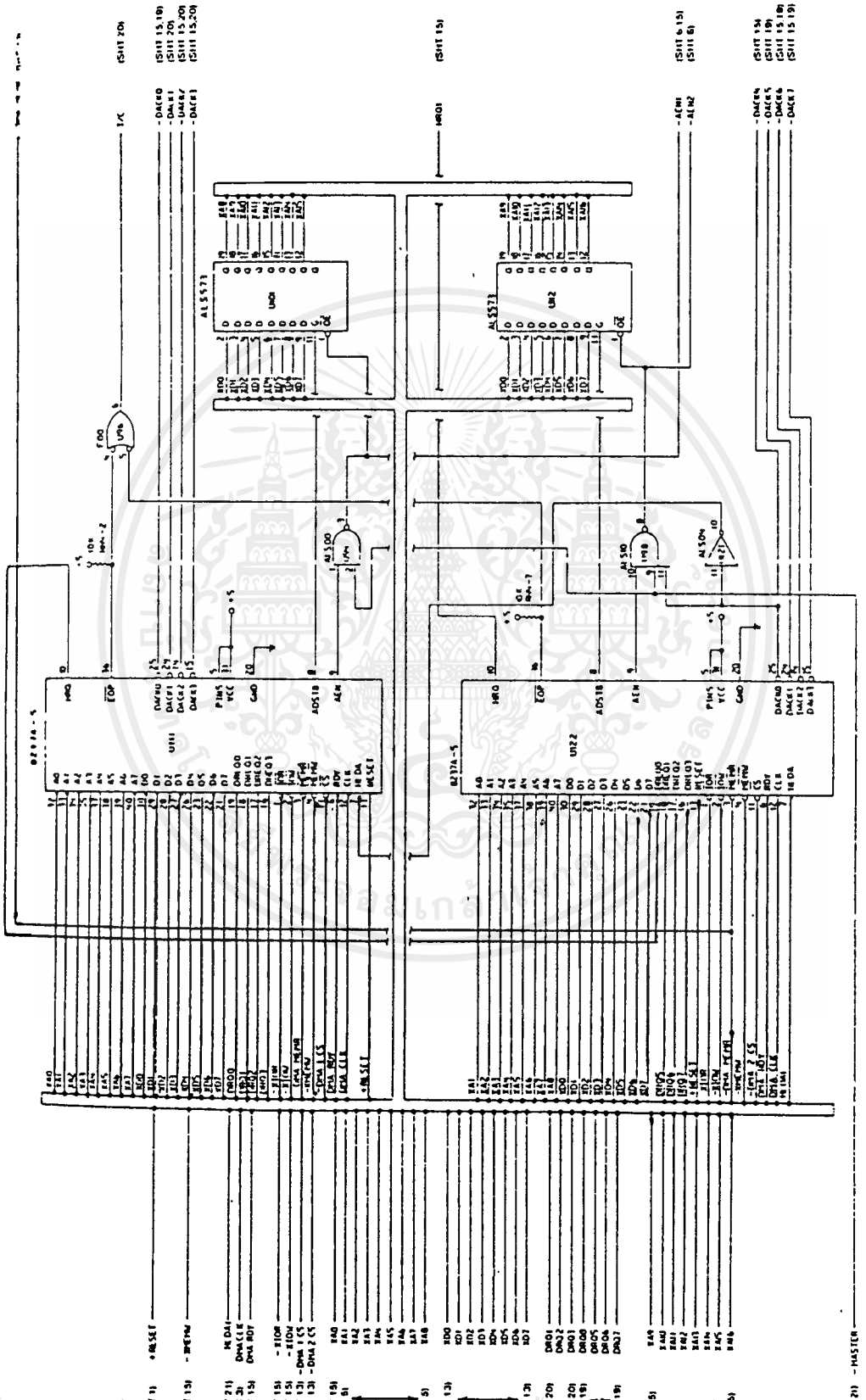
รูปที่ 2.16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



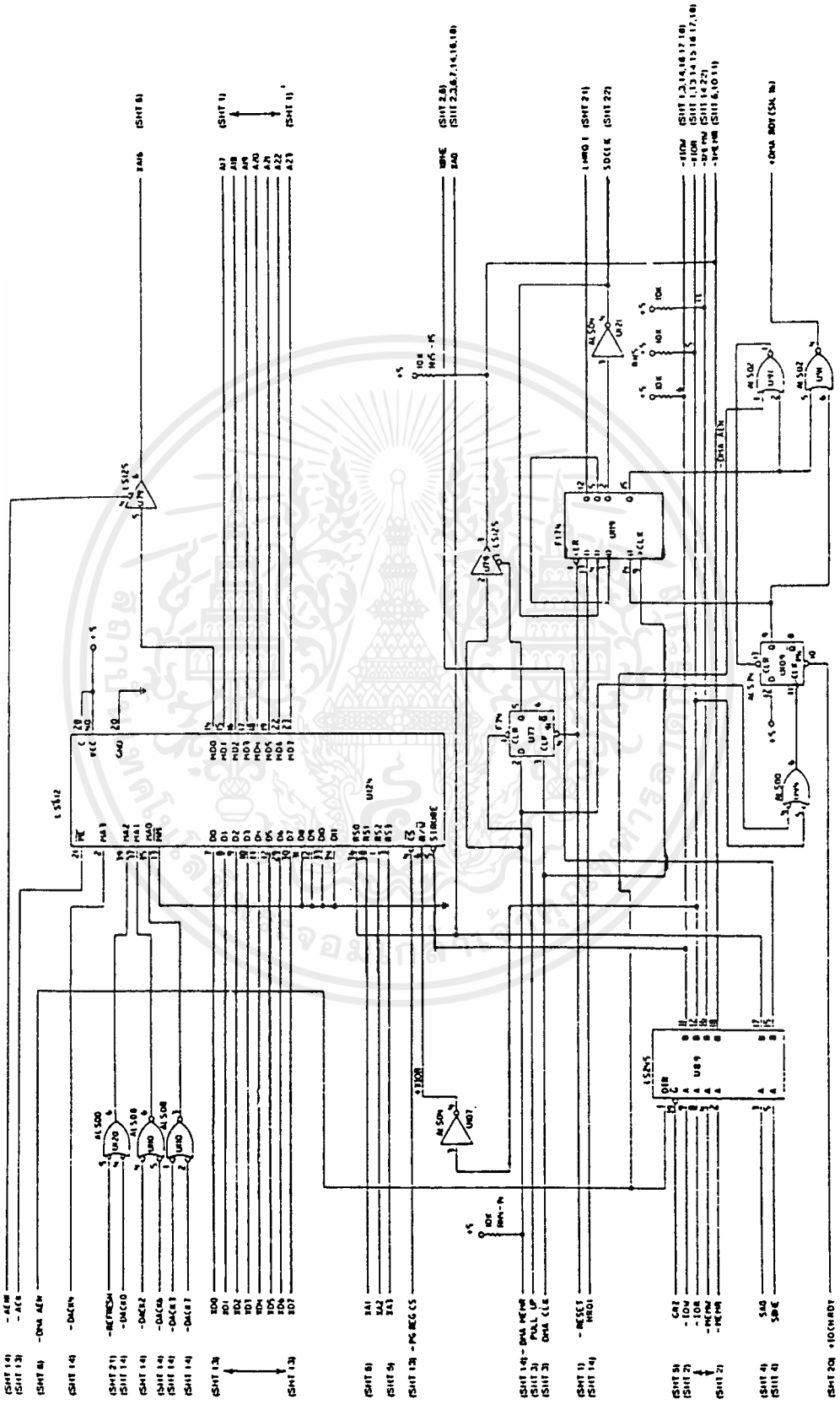
รูปที่ 2.17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



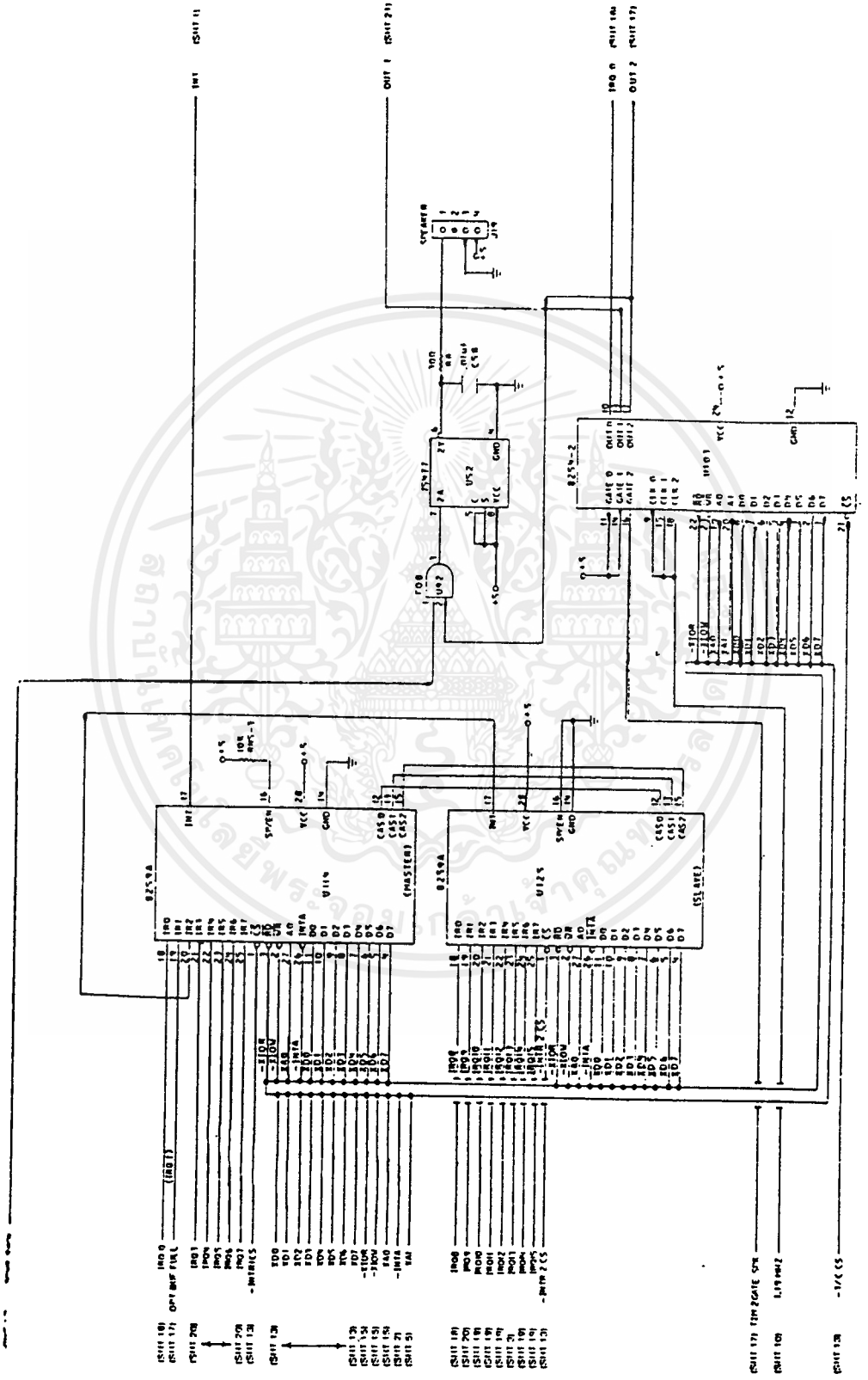
รูปที่ 2.18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



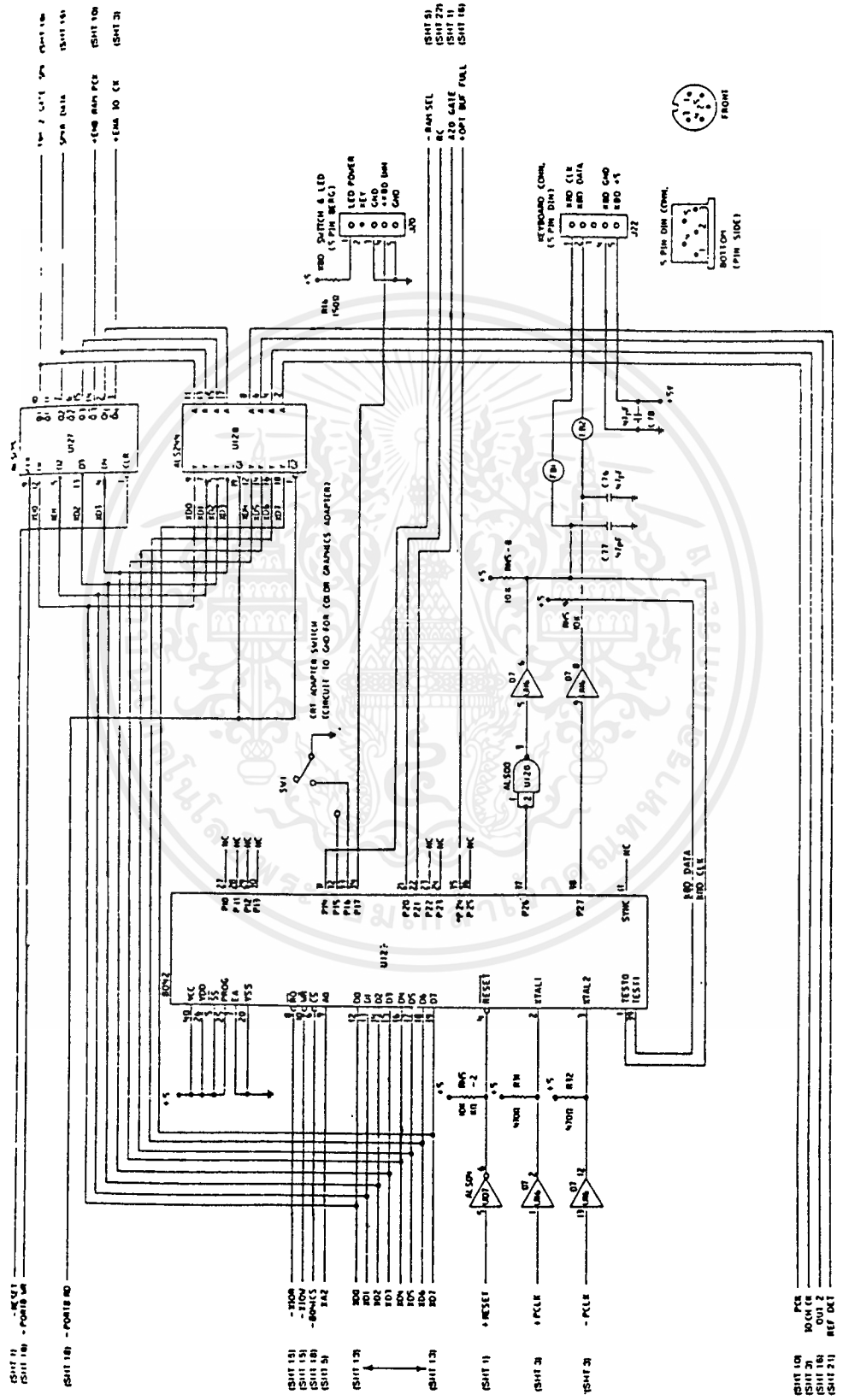
รูปที่ 2.19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.21

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยบอกแอดเดรสเริ่มต้นที่ 8237A บิต A0-A15 และบอกไปที่ LS612 ให้กับ 8237 และการกำหนดแอดเดรสบิต A17-A23 ให้กับดีเอ็มเอเพจรีจิสเตอร์ หรือ LS612 นั่นเอง สังเกตความแตกต่างกันระหว่างดีเอ็มเอ 8237A ทั้งสองตัว โดยตัวแรกจะทำงานเหมือนกับเครื่องเอ็กซ์ที แต่ตัวหลังเพิ่มขีดความสามารถให้ทำงานได้เต็ม 16 บิต สำหรับการส่งถ่ายข้อมูลแทนที่จะเป็น 8 บิต ตามระบบบัสข้อมูลของ 8088 ทั้งนี้เพราะบัสข้อมูลของ 80286 มี 16 เส้นแล้ว อย่างไรก็ตามเมื่อทำการดีเอ็มเอแบบ 16 บิตของชิป 8237 ตัวใหม่คือ ช่อง 5-7 จะต้องให้สัญญาณ BHE และ A0 เป็นลอจิก "0"

เนื่องจากการทำดีเอ็มเอต้องมีการกำหนดเพจแอดเดรสเสียก่อน ดังนั้นการเคลื่อนย้ายข้อมูลจะทำภายในตัว 8237 จึงทำให้ค่าแอดเดรสที่ตั้งไว้เปลี่ยนค่าเฉพาะในชิป 8237 เท่านั้นด้วยเหตุนี้เอง ทำให้เราจะกำหนดขนาดบล็อกเกินกว่า 64 กิโลไบต์ สำหรับตัวแรกไม่ได้ เพราะบล็อกจะไม่เกิน 128 กิโลไบต์ สำหรับชิปตัวหลัง

2.10 การโปรแกรมดีเอ็มเอแบบ 16 บิต

จากการที่กล่าวแล้วว่า แชนเนล 5-7 สำหรับเครื่องพีซีเอทีนี้ได้ออกแบบไว้สำหรับการทำดีเอ็มเอแบบ 16 บิต การควบคุมจะทำได้ด้วยการกำหนดค่าลงไปในพอร์ตแอดเดรส 0C0-0DF โดยแต่ละพอร์ตมีความหมายดังนี้

พอร์ต 0C0 เบสและแอดเดรสปัจจุบันของแชนเนล 0
พอร์ต 0C2 เบสแอดเดรสและตัวนับเวิร์ดของแชนเนล 0
พอร์ต 0C4 เบสและแอดเดรสปัจจุบันของแชนเนล 1
พอร์ต 0C6 เบสแอดเดรสและตัวนับเวิร์ดของแชนเนล 1
พอร์ต 0C8 เบสและแอดเดรสปัจจุบันของแชนเนล 2
พอร์ต 0CA เบสแอดเดรสและตัวนับเวิร์ดของแชนเนล 2
พอร์ต 0CC เบสและแอดเดรสปัจจุบันของแชนเนล 3
พอร์ต 0CE เบสแอดเดรสและตัวนับเวิร์ดของแชนเนล 3
พอร์ต 0D0 พอร์ตรีจิสเตอร์สถานี (อ่าน)/รีจิสเตอร์คำสั่ง (เขียน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พอร์ต 0D2 รีจิสเตอร์ Request (เขียน)

พอร์ต 0D4 มาส์กบิตรีจิสเตอร์ (เขียน)

พอร์ต 0D6 โหมดรีจิสเตอร์ (เขียน)

พอร์ต 0D8 ฟลิปฟลอปสำหรับเคลียร์ไบท์พอยน์เตอร์

พอร์ต 0DA รีจิสเตอร์ชั่วคราว (อ่าน)/มาสเตอร์เคลียร์ (เขียน)

พอร์ต 0DC เคลียร์มาส์กรีจิสเตอร์

พอร์ต 0DE เขียนมาส์กรีจิสเตอร์บิต

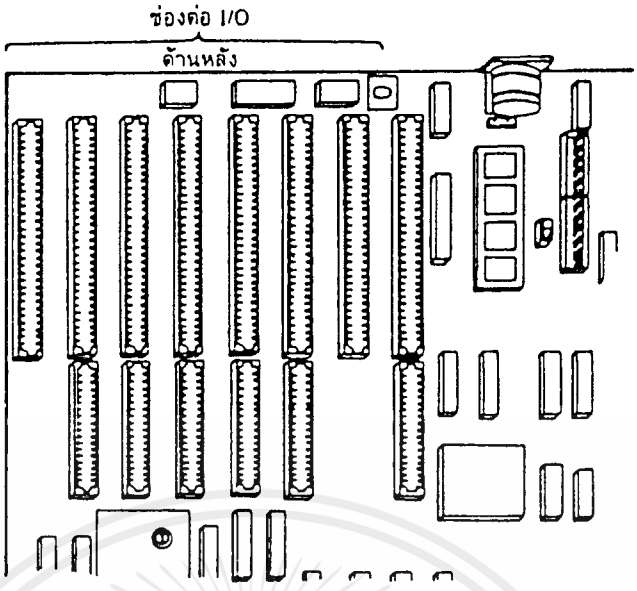
ในการเริ่มต้นทำงาน โปรแกรมไบออสจะเริ่มต้นตั้งค่าให้กับ 8237 เพื่อกำหนดโหมดการทำงาน การกำหนดโหมดการทำงานนี้จะต้องคล้องจองกับการทำงานในส่วนต่าง ๆ ด้วย

2.11 สล็อตหรือช่องต่อสำหรับอินพุตและเอาต์พุต

ไมโครคอมพิวเตอร์แบบเอ็กซ์ที มีสล็อตแบบ 62 จำนวน 8 สล็อตไว้ต่อเชื่อมกับอินพุต-เอาต์พุต แต่เมื่อพัฒนามาเป็นเครื่องแบบเอที ทำให้ขีดความสามารถบางอย่างเพิ่มขึ้น ดังนั้นจึงจำเป็นต้องปรับปรุงสล็อตเพิ่มเติมและเพื่อให้ใช้วงจรกับของเดิมได้ บริษัทไอบีเอ็มจึงกำหนดสล็อตเพิ่มเติมจากเดิมโดยมีโครงสร้างรูปแบบของจริงดังรูปที่ 2.21 สำหรับจุดมุ่งหมายของช่องต่ออินพุตและเอาต์พุต หรือสล็อตนี้มีเพื่อสนับสนุนดังนี้

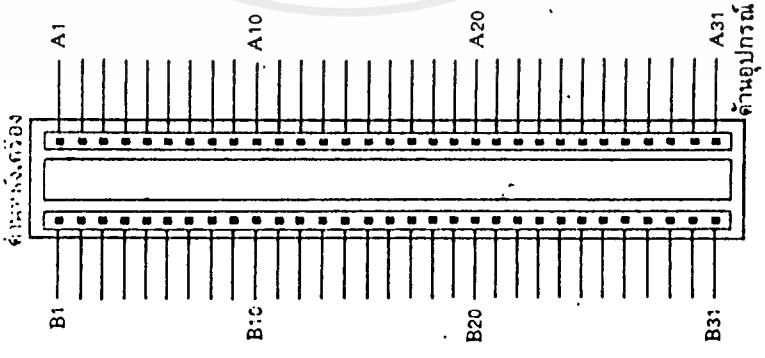
- แอดเดรสหมายเลขพอร์ตจากพอร์ตหมายเลข 100 ถึง 3FF
- ให้มีแอดเดรสครบ 24 เส้นตามโครงสร้างของ 80286 เพื่ออ้างอิงหน่วยความจำได้ 16 MB
- แชนเนลดีเอ็มเอ
- สร้างสถานะการรอของอินพุตหรือเอาต์พุต (I/O wait state)
- เปิดสถานะของการเชื่อมต่อเพื่อให้อุปกรณ์ภายนอกเชื่อมโยงกับระบบในส่วนต่างๆ ได้ง่าย
- รีเฟรชหน่วยความจำจากแชนเนลของไมโครโปรเซสเซอร์ภายใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



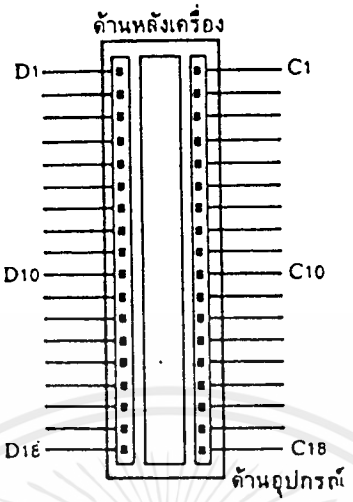
รูปที่ 2.22 การเพิ่ม J10-J16 เป็นสล็อตเพิ่มจากเดิม

จากรูปที่ 2.22 ได้แสดงจำนวนของช่องอินพุต/เอาต์พุตโดยแบ่งเป็น 2 ส่วน คือ ส่วนแรกมีขนาด 62 ขา ส่วนที่ 2 มีขนาด 36 ขา สำหรับวงจรการเชื่อมต่อกับสล็อตแสดงไว้ในรูปที่ 2.28 วงจรแผ่นที่ 13 ซึ่งเป็นสล็อตแบบ 62 ขา และรูปที่ 2.27 วงจรแผ่นที่ 12 เป็นส่วนขยายแบบ 36 ขา ส่วนสล็อตแบบ 62 ขา และสล็อตแบบ 36 ขา แสดงได้จากรูปที่ 2.23 และรูปที่ 2.24 ตามลำดับ และตำแหน่งขาบนสล็อตแสดงดังตารางที่ 2.6



รูปที่ 2.23 การนับขาของสล็อตแบบ 62 ขา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.24 การนับขาของสล็อตแบบ 36 ขา

ตารางที่ 2.6 ชื่อของสัญญาณขาต่าง ๆ ของสล็อต

ขาอินพุต เอาต์พุต	ชื่อสัญญาณ	อินพุต/ เอาต์พุต	ขาอินพุต/ เอาต์พุต	ชื่อสัญญาณ	อินพุต/ เอาต์พุต
A 1	-I/O CH CK	I	B 1	GND	กราวนด์
A 2	SD7	I/O	B 2	RESET DRV	O
A 3	SD6	I/O	B 3	-5 Vdc	แหล่งจ่ายไฟเลี้ยง
A 4	SD5	I/O	B 4	IRQ9	I
A 5	SD4	I/O	B 5	-5 Vdc	แหล่งจ่ายไฟเลี้ยง
A 6	SD3	I/O	B 6	DRQ2	I
A 7	SD2	I/O	B 7	-12 Vdc	แหล่งจ่ายไฟเลี้ยง
A 8	SD1	I/O	B 8	OWS	I
A 9	SD0	I/O	B 9	+12 Vdc	แหล่งจ่ายไฟเลี้ยง
A 10	-I/O CH RDY	I	B 10	GND	กราวนด์
A 11	AEN	O	B 11	-SMEMW	O
A 12	SA19	I/O	B 12	-SMEMR	O
A 13	SA18	I/O	B 13	-IOW	I/O
A 14	SA17	I/O	B 14	-IOR	I/O
A 15	SA16	I/O	B 15	-DACK3	O
A 16	SA15	I/O	B 16	DRQ3	I
A 17	SA14	I/O	B 17	-DACK1	O
A 18	SA13	I/O	B 18	DRQ1	I
A 19	SA12	I/O	B 19	-Refresh	I/O
A 20	SA11	I/O	B 20	CLK	O
A 21	SA10	I/O	B 21	IRQ7	I
A 22	SA9	I/O	B 22	IRQ6	I

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.6 (ต่อ) ชื่อของสัญญาณขาต่าง ๆ ของสล๊อต

ขาอินพุต/เอาต์พุต	ชื่อสัญญาณ	อินพุต/เอาต์พุต
A24	SA7	I/O
A25	SA6	I/O
A26	SA5	I/O
A27	SA4	I/O
A28	SA3	I/O
A29	SA2	I/O
A30	SA1	I/O
A31	SA0	I/O

อินพุต/เอาต์พุตแชนเนลด้าน A J1 ถึง J8

ขาอินพุต/เอาต์พุต	ชื่อสัญญาณ	อินพุต/เอาต์พุต
B24	IRQ4	I
B25	IRQ3	I
B26	-DACK2	O
B27	T/C	O
B28	BALE	O
B29	+5 Vdc	แหล่งจ่ายไฟเลี้ยง
B30	OSC	O
B31	GND	กราวนด์

อินพุต/เอาต์พุตแชนเนลด้าน B J1 ถึง J8

ขาอินพุต/เอาต์พุต	ชื่อสัญญาณ	อินพุต/เอาต์พุต
C1	SBHE	I/O
C2	LA23	I/O
C3	LA22	I/O
C4	LA21	I/O
C5	LA20	I/O
C6	LA19	I/O
C7	LA18	I/O
C8	LA17	I/O
C9	-MEMR	I/O
C10	-MEMW	I/O
C11	SD08	I/O
C12	SD09	I/O
C13	SD10	I/O
C14	SD11	I/O
C15	SD12	I/O
C16	SD13	I/O

ขาอินพุต/เอาต์พุต	ชื่อสัญญาณ	อินพุต/เอาต์พุต
D1	-MEM CS16	I
D2	-I/O CS16	I
D3	IRQ10	I
D4	IRQ11	I
D5	IRQ12	I
D6	IRQ15	I
D7	IRQ14	I
D7	-DACK0	O
D9	DRQ0	I
D10	-DACK5	O
D11	DRQ5	I
D12	-DACK6	O
D13	DRQ6	I
D14	-DACK7	O
D15	DRQ7	I
D16	+5 Vdc	แหล่งจ่ายไฟเลี้ยง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.6 (ต่อ) ชื่อสัญญาณขาต่าง ๆ ของสล็อต

ขาอินพุต/เอาต์พุต	ชื่อสัญญาณ	อินพุต/เอาต์พุต
C17	SD14	I/O
C18	SD15	I/O

อินพุต/เอาต์พุตแชนเนลด้าน C
J10 ถึง J14 และ J16

ขาอินพุต/เอาต์พุต	ชื่อสัญญาณ	อินพุต/เอาต์พุต
D17	-MASTER	I
D18	GND	กราวนด์

อินพุต/เอาต์พุตแชนเนลด้าน D
J10 ถึง J14 และ J16

สัญญาณที่ต่อเชื่อมกับอุปกรณ์อินพุต/เอาต์พุต เป็นสัญญาณที่มีขนาด 5 โวลต์ ตามมาตรฐาน TTL โดยที่แต่ละสล็อตจะเชื่อมต่อกับ ทีทีแอลแบบ LS ได้ 2 อินพุต ดังนั้นการต่อกับสล็อตจำเป็นต้องถึงโหลดดังกล่าวนี้ด้วย สัญญาณที่ขาต่าง ๆ ของสล็อตมีความหมายดังนี้

SA0-SA19 (อินพุต/เอาต์พุต) เป็นแอดเดรสของระบบที่ใช้ติดต่อกับหน่วยความจำและอุปกรณ์อินพุต/เอาต์พุต สายสัญญาณนี้จะต่อกับหน่วยความจำได้ 1 MB แต่ถ้าต้องการเชื่อมขยายแอดเดรสจะต้องใช้สายแอดเดรส LA17-LA23 การใช้สัญญาณ SA0-SA19 จะต้องแอกทีฟขณะที่สัญญาณ BALE เป็น "1" และจะแลตช์ไบท์ใช้ขณะเปลี่ยนจาก "1" ไป "0" สัญญาณ BALE เป็นสัญญาณที่มาจากไมโครโปรเซสเซอร์ หรือดีเอ็มเอคอนโทรลเลอร์

LA17-LA23 (อินพุต/เอาต์พุต) สัญญาณนี้เป็นสัญญาณที่ไม่ผ่านการแลตช์มาเลย เป็นสัญญาณที่ขยายเพิ่มต่อให้ระบบใช้กับหน่วยความจำได้เต็มที่ 16 MB สัญญาณนี้จะใช้ได้ต่อเมื่อ BALE เป็น "1" สัญญาณนี้จะไม่มีการแลตช์มาเลยจากไมโครโปรเซสเซอร์ ทั้งนี้เพื่อให้ใช้สำหรับการสร้างสถานะการรอ (wait state) ได้ สัญญาณนี้ได้รับการควบคุมโดยโปรเซสเซอร์และดีเอ็มเอคอนโทรลเลอร์ เพื่อควบคุมการเข้าถึงข้อมูล

CLK (เอาต์พุต) เป็นสัญญาณนาฬิกาของระบบ ในกรณีของไอบีเอ็มเอที จะส่งสัญญาณนี้เป็นสัญญาณขนาด 6 MHz โดยมีช่วงเวลาประมาณ 167 นาโนวินาที สัญญาณเป็นรูปสี่เหลี่ยมมี duty cycle 50 เปอร์เซ็นต์ สัญญาณนี้มีจุดมุ่งหมายเพื่อใช้ในการซิงโครนัสระบบ มิได้มีจุดมุ่งหมายสำหรับให้ใช้เป็นฐานเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RESET DRV (เอาต์พุต) สัญญาณนี้ใช้สำหรับรีเซตระบบในขณะปิดเครื่อง หรือขณะที่แหล่งจ่ายไฟเลี้ยงขาด หรือไฟตก สัญญาณนี้จะแอกทีฟเมื่อเป็นลอจิก "1"

SD0-SD15 (อินพุต/เอาต์พุต) เป็นสัญญาณข้อมูลขนาด 16 บิต ที่ใช้ติดต่อกับหน่วยความจำไมโครโปรเซสเซอร์ และอุปกรณ์อินพุต/เอาต์พุต บิต D0 เป็นบิตที่มีนัยสำคัญน้อยที่สุดในการติดต่อกับอุปกรณ์บางอย่างที่ต้องใช้ 8 บิต จะมีวิธีการแปลงบิตข้อมูลจาก SD8-SD15 เข้ามาใน 8 บิตล่างได้ เพื่อให้การติดต่อเป็นไปได้ทั้งแบบ 16 บิตและ 8 บิต

BALE (เอาต์พุต) เป็นสัญญาณที่ใช้สำหรับการแลตช์แอดเดรสของระบบ สัญญาณนี้มาจาก 82288 ตัวควบคุมบัสสัญญาณที่จะใช้แลตช์แอดเดรสเมื่อเปลี่ยนจาก "1" กับ "0" และสัญญาณนี้จะได้รับการทำให้เป็น "1" ขณะที่กำลังทำดีเอ็มเอ

I/O CHK (อินพุต) สัญญาณตรวจสอบของอินพุต/เอาต์พุต เพื่อบอกข้อมูลกับระบบเช่นเดียวกับการตรวจสอบพาริตี้ ดังนั้นถ้าบนอินพุต/เอาต์พุตมีข้อผิดพลาด สัญญาณนี้จะแอกทีฟ เพื่อให้ส่งสัญญาณเตือนในลักษณะ parity error

I/O CHRDY (อินพุต) สัญญาณนี้จะได้รับการทำให้เป็น "0" ด้วยหน่วยความจำหรืออุปกรณ์อินพุต/เอาต์พุต การใช้สัญญาณนี้ก็เพื่อให้อุปกรณ์อินพุต/เอาต์พุตที่ช้าจะได้ติดต่อกับระบบด้วยการส่งสัญญาณมายังซีพียู เพื่อซิงโครไนส์ระบบได้

IRQ3-IRQ7,IRQ9-IRQ12 และ IRQ14-IRQ15 (อินพุต) สัญญาณอินเตอร์รัพท์เหล่านี้ต่อเข้าเป็นสัญญาณอินพุต 8259A สองตัว เพื่อให้สัญญาณ INT เข้าสู่ไมโครโปรเซสเซอร์การจัดลำดับความสำคัญเป็นไปตามที่กล่าวมาแล้วในเรื่องของวงจรอินเตอร์รัพท์ โดยมี IRQ7 มีลำดับความสำคัญน้อยที่สุด IRQ9 มีลำดับความสำคัญสูงสุด IRQ8 ใช้สำหรับสัญญาณนาฬิกาที่กำหนดเวลาจริง

IOR (อินพุต/เอาต์พุต) สัญญาณอินพุต/เอาต์พุตเป็นสัญญาณที่ส่งมาจากซีพียู การควบคุมสัญญาณนี้มาจาก 80286 และดีเอ็มเอคอลโทรลเลอร์ สัญญาณนี้แอกทีฟ "0"

IOW (อินพุต/เอาต์พุต) สัญญาณเขียนข้อมูลลงบนอุปกรณ์อินพุต/เอาต์พุต สัญญาณนี้ควบคุมจากไมโครโปรเซสเซอร์หรือดีเอ็มเอคอลโทรเลอร์ สัญญาณนี้แอกทีฟด้วยลอจิก "0"

SMEAR (เอาต์พุต) MEMR (อินพุต/เอาต์พุต) สัญญาณนี้เป็นสัญญาณควบคุมการอ่านข้อมูลจากหน่วยความจำ SMEMR ใช้สำหรับติดต่อกับหน่วยความจำในส่วน 1 MB แรกหรือถอดรหัสมาจากแอดเดรสส่วนล่าง ส่วน MEMR นี้แอกทีฟกับหน่วยความจำได้หมด 16 MB

SMEMW (เอาต์พุต) MEMW (อินพุต/เอาต์พุต) สัญญาณนี้เป็นสัญญาณควบคุมการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เขียนข้อมูลลงหน่วยความจำ โครงสร้างอย่างอื่นเหมือนกับ SMEMR และ MEMR

DRQ0–DRQ3 และ DRQ5–DRQ7 (อินพุต) สัญญาณการขอดีเอ็มเอแชลแนล 0-3 และ 5-7 สัญญาณนี้จะมาจากอุปกรณ์อินพุต/เอาต์พุต DRQ0 มีลำดับความสำคัญสูงสุดและ DRQ7 มีลำดับความสำคัญต่ำสุด DRQ0–DRQ3 ใช้กับดีเอ็มเอแบบ 8 บิต ส่วน DRQ5–7

AEN (เอาต์พุต) อีนาเบิลแอดเดรส เพื่อเป็นสัญญาณเพื่อใช้สำหรับการแยกบัสแอดเดรสในการทำดีเอ็มเอ เมื่อสัญญาณนี้แอกทีฟจะเป็นการทำให้ดีเอ็มเอคอนโทรลเลอร์สามารถควบคุมการทำงานแอดเดรสแทนการควบคุมของซีพียู

REFRESH (อินพุต/เอาต์พุต) เป็นสัญญาณที่ใช้ในการแสดงสัญญาณรีเฟรชไซเคิล สัญญาณนี้ส่งมาจากไมโครโปรเซสเซอร์ผ่านทางช่องอินพุต/เอาต์พุต

T/C (เอาต์พุต) สัญญาณ Terminal Count เป็นสัญญาณพัลซ์เมื่อดีเอ็มเอนับจำนวนมาครบตามที่กำหนด

SBHE (อินพุต/เอาต์พุต) ชื่อสัญญาณ Bus High Enable เป็นสัญญาณบ่งบอกการถ่ายข้อมูลจาก SD8–SD15 เข้าสู่บัฟเฟอร์

MASTER (อินพุต) สัญญาณนี้ใช้กับ DRQ เพื่อควบคุมระบบ สัญญาณนี้มีจุดมุ่งหมายบ่งบอกการควบคุมบัสทั้งหมดว่ามาจากระบบซีพียูหลักนี้ หรือมาจากที่อื่น ถ้าหากสัญญาณนี้แอกทีฟ หมายความว่า ซีพียูเดิมส่งอำนาจการควบคุมให้กับสล็อต ซึ่งอาจจะมีซีพียูอื่น เข้ามาควบคุมระบบก็ได้ อนึ่งหากสัญญาณนี้แอกทีฟเกินกว่า 15 ไมโครวินาที โดยไม่มีกลไกรีเฟรชช่วยอาจทำให้ข้อมูลในหน่วยความจำหายได้

MEM CS16 (อินพุต) สัญญาณนี้เป็นตัวส่งมาบอกเมนบอร์ด ถ้าหากการถ่ายเทข้อมูลต้องการสถานะรอ

IO CS16 (อินพุต) สัญญาณนี้เป็นตัวส่งบอกเมนบอร์ดว่าอินพุต/เอาต์พุตต้องการสถานะรอ

OSC (เอาต์พุต) สัญญาณนาฬิกา 70 นาโนวินาที หรือประมาณ 14.31818 เมกะเฮิรตซ์สัญญาณนี้ไม่ได้ซิงโครนัสกับระบบ

OWS (อินพุต) เป็นสัญญาณที่จะบอกซีพียูว่า การทำงานในหนึ่งรอบของบัสนั้นจำเป็นต้องแทรกสถานะรอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.12 วงจรสัญญาณเสียง

วงจรสัญญาณเสียงออกมาทาง OUT2 ของ 8254 ผ่าน AND เกตเข้ากับสัญญาณ SPKR DATA ซึ่งมาจากการแลตช์พอร์ต เพื่อ ON หรือ OFF เสียง การกำเนิดเสียงจึงเกิดจากการส่ง ON หรือ OFF ผ่าน U_{27} คือ ALS175 ส่วนสัญญาณความถี่ของเสียงเกิดจากวงจรตั้งเวลาในตัว 8254 ซึ่งค่าความถี่ของเสียงสามารถโปรแกรมได้

2.13 วงจรควบคุมคีย์บอร์ด

มีสิ่งที่น่าสนใจแตกต่างจากเครื่อง เอกซ์ที คือใช้ไมโครโปรเซสเซอร์ชิปเดียวคือ 8042 สำหรับแทน 8255 เดิมของ เอ็กซ์ที ในเครื่องที่นี้การเชื่อมต่อกับคีย์บอร์ดจำต้องผ่านทาง 8042 นี้ โครงสร้างของไมโครโปรเซสเซอร์ 8042 ที่ต่ออยู่กับระบบ แสดงเป็นบล็อกไดอะแกรมเพื่อความเข้าใจได้ง่ายดังรูปที่ 2.24

ผู้ออกแบบพีซี เอทีเห็นว่าการใช้ 8255 และดิพสวิตช์ในพีซี เอ็กซ์ทีมีจุดอ่อน เพราะต้องการใช้ฮาร์ดแวร์เพิ่มในการตรวจสอบแป้นพิมพ์ และหากเปลี่ยนแปลงแก้ไขรหัสแกนที่มาจากแป้นพิมพ์ก็จะยุ่งยาก และยังทำให้ฮาร์ดแวร์ส่วนนี้มีมากขึ้น การใช้ 8042 จึงเป็นความคิดที่ดีมาก

8042 นี้มีรอมภายใน 2 กิโลไบต์ และมีแรม 128 ไบต์ มีพอร์ตอินพุต/เอาต์พุตที่สามารถเชื่อมต่อกันภายนอกได้โดยตรง

การรับข้อมูลจากแป้นพิมพ์รับเข้าทางขา T_0 และ T_5 โดยสัญญาณนาฬิกาที่ซิงโครนัสกับข้อมูลเข้าทาง T_0 และตัวข้อมูลคีย์บอร์ดเข้าทาง T_1 จะเห็นจุดเชื่อมต่อสายออกมาสองปลั๊ก ปลั๊กแรกเป็นตัวเชื่อมกับคีย์บอร์ดมี 5 ขา เหมือนขาของพีซี เอ็กซ์ที โดยแต่ละขามีรายละเอียดดังนี้

ขา 1 เป็นสัญญาณนาฬิกาของคีย์บอร์ด

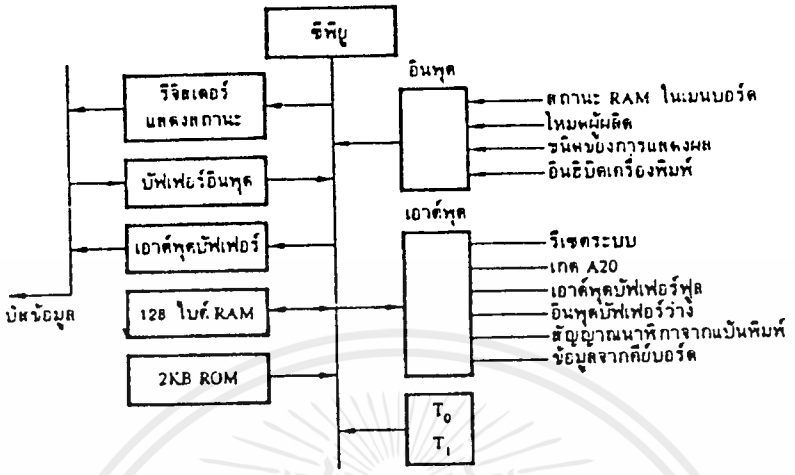
ขา 2 เป็นสัญญาณข้อมูล

ขา 3 ว่าง

ขา 4 กราวนด์

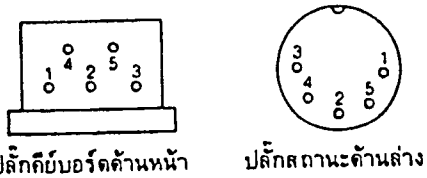
ขา 5 แรงดัน 5 โวลท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.25 วงจรบล็อกไดอะแกรม 8042 สำหรับเชื่อมต่อกับแป้นพิมพ์

การจัดวางขาแสดงดังรูปที่ 2.26 ส่วนปลั๊กอีกตัวคือ ปลั๊กต่อแสดงสถานะและการล็อกคีย์บอร์ด โดยมีปลั๊กขนาด 5 ขา ดังนี้



ปลั๊กคีย์บอร์ดด้านหน้า ปลั๊กสถานะด้านล่าง

รูปที่ 2.26 ลักษณะปลั๊กที่เชื่อมต่อกับ 8042

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขา 1 ต่อกับ LED แสดงไฟเลี้ยง

ขา 2 ว่าง

ขา 3 กราวนด์

ขา 4 ขาล็อกคีย์บอร์ดจากกุญแจล็อก

ขา 5 กราวนด์

สำหรับไมโครคอมพิวเตอร์ชิปเดียว 8042 นี้ต่อเข้ากับระบบโดยการเลือกแอดเดรสทาง -8041CS ซึ่งเป็นการเลือกพอร์ตเหมือนการเลือก 8255 ของพีซีเอ็กซ์ที การเขียนการอ่านผ่านทาง XIOR และ XIOW ทำให้ซีพียู 80286 สามารถติดต่อผ่านเข้าทางบัสข้อมูลในการอ่านสถานะได้ พอร์ตของ 8042 พอร์ตแรกคือ พอร์ต P10-P17 จะเห็นว่า P10-P13 ไม่ได้ใช้ ส่วน P14-P17 ใช้สำหรับอินพุต P14 เป็นตัวตรวจสอบแรม P16 เป็นตัวอ่านค่าดิพสวิตช์เพื่อเลือกว่าขณะนี้ระบบต่ออยู่กับจอสีหรือจอโมโนโครม ส่วน P17 เป็นตัวตรวจสอบสถานะอินฮิบิตคีย์บอร์ดเพื่อล็อกคีย์บอร์ด

การรับข้อมูลจะอ่านเข้ามาทางขา TEST0 และ TEST1 อย่างไรก็ตามมีพอร์ต P26 และ P27 เป็นเอาต์พุตลูปเพื่อการทดสอบคีย์บอร์ด สำหรับการอ่านค่าคีย์บอร์ดจะได้ค่ารหัสสแกนซึ่งต้องแปลงค่าเป็นรหัสเอสกีภายในโปรแกรมไบออส

2.14 รีจิสเตอร์แสดงสถานะ

แต่เดิมบนพีซีเอ็กซ์ทีใช้สวิตช์เล็ก ๆ เป็นตัวแสดงสถานะ แต่กรณีของพีซีเอทีนี้ใช้รีจิสเตอร์ภายในของ 8042 ในส่วนพอร์ตหมายเลข 64H (ดูจากซีพียู 80286) ซึ่งจะอ่านข้อมูลจาก 8042 ได้ 8 บิต เรียกว่ารีจิสเตอร์แสดงสถานะ โดยที่แต่ละบิตมีความหมายแตกต่างกันออกไปตามโปรแกรมที่ตั้งไว้ โดยมีรายละเอียดดังนี้

บิต 0 บอกบัฟเฟอร์ของแป้นพิมพ์เต็ม ถ้ามีค่า "0" หมายถึงบัฟเฟอร์ของแป้นพิมพ์ยังว่างอยู่ ส่วนของบัฟเฟอร์เอาต์พุตอยู่ที่พอร์ตหมายเลข 60H ดังนั้นถ้าบิตนี้มีค่าเป็น "1" และถ้าหากมีการอ่านพอร์ต 60H จะทำให้บิตนี้กลายเป็น "0"

บิต 1 บัฟเฟอร์อินพุตเต็ม บอกว่าบัฟเฟอร์การควบคุมแป้นพิมพ์พอร์ตแอดเดรส 60 และ 64 ว่าว่างหรือเต็ม ถ้าบิตนี้เป็น "0" จะบอกว่าว่าง ถ้าเป็น "1" จะบอกว่าข้อมูลได้รับการเขียนลงไปในบัฟเฟอร์แล้ว และคอนโทรลเลอร์ยังไม่ได้อ่านไป เมื่อคอนโทรลเลอร์อ่านไปจะทำให้เป็น "0"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บิต 2 แฟลกของระบบ บิตนี้จะได้รับการเขียนให้เป็น "0" เมื่อเริ่มต้นรีเซตระบบ บิตนี้จะได้รับการเซตค่าด้วยการส่งคำสั่งเข้ามาควบคุมแป้นพิมพ์

บิต 3 บอกว่าเป็นคำสั่งหรือข้อมูล โดยบัฟเฟอร์ของคอนโทรลเลอร์พอร์ต 60H หรือ 64H อาจจะเป็นข้อมูลหรือคำสั่ง โดยแอดเดรสพอร์ต 60H กำหนดให้เป็นพอร์ตข้อมูล และ 64H ให้เป็นคำสั่ง หากต้องการเขียนไปยังพอร์ต 64H ต้องเซตบิตนี้ให้เป็น "1" และถ้าเซตเป็น "0" หมายถึงต้องการติดต่อกับพอร์ตแอดเดรส 60H

บิต 4 สถานะการอินฮิบิตคีย์บอร์ด ถ้าเป็น "0" แสดงว่าอินฮิบิตคีย์บอร์ดคือล็อคคีย์บอร์ดนั่นเอง

บิต 5 หมดเวลาส่ง ถ้าเป็น "1" หมายถึงการส่งข้อมูลจากการส่งข้อมูลคอนโทรลเลอร์ไม่สมบูรณ์ เป็นบิตเพื่อบอกสถานะข้อผิดพลาด

บิต 6 หมดเวลารับ เป็น "1" แสดงว่าการส่งข้อมูลมาแล้วโปรแกรมไม่สามารถรับไปในเวลาที่กำหนด

บิต 7 ข้อผิดพลาดจากพาริตี้ เป็น "0" แสดงว่าข้อมูลที่ได้รับเป็นพาริตี้คี่ เป็น "1" แสดงว่าเป็นพาริตี้คู่

เอาต์พุตบัฟเฟอร์ ที่กล่าวถึงคือบัฟเฟอร์แบบอ่านได้อย่างเดียว อยู่ที่พอร์ตหมายเลข 60H ตัวคีย์บอร์ดคอนโทรลเลอร์รับข้อมูลมาจากคีย์บอร์ดแล้วใส่ไว้ในบัฟเฟอร์ตัวนี้เพื่อให้โปรแกรมไบออสอ่านข้อมูลไป

อินพุตบัฟเฟอร์ เป็นรีจิสเตอร์ที่เขียนได้อย่างเดียว อยู่ที่พอร์ต 60H หรือ 64H ถ้าเขียนที่แอดเดรส 60H จะเซตแฟลกเพื่อบอกเป็นการเขียนข้อมูล ถ้าเขียนที่แอดเดรส 64H จะได้รับการเซตแฟลกเป็นการเขียนคำสั่ง ข้อมูลที่เขียนลงในพอร์ต 60H จะส่งต่อให้กับคีย์บอร์ด

พอร์ต 64H พอร์ตคำสั่ง โปรแกรมคอนโทรลเลอร์สามารถส่งรหัสเพื่อควบคุมการทำงานของคีย์บอร์ดคอนโทรลเลอร์ผ่านเข้าทางพอร์ตนี้ คำสั่งที่เขียนมีรหัสดังนี้

20 อ่านไบต์คำสั่งของคอนโทรลเลอร์ เพื่อบอกให้คีย์บอร์ดคอนโทรลเลอร์ส่งรหัสคำสั่งขณะนั้นมาที่เอาต์พุตบัฟเฟอร์

60 เขียนไบต์คำสั่งของคอนโทรลเลอร์ เป็นการบอกว่าต้องการเขียนคำสั่งโดยไบต์ที่ตามมาในพอร์ต 60H จะเป็นคำสั่งโดยแต่ละบิตมีความหมายดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บิต 7 สงวนไว้ (โดยปกติให้เป็น "0")

บิต 6 โหมดไอบีเอ็ม หมายถึงการกำหนดรหัสสแกนจะตามที่บริษัทไอบีเอ็มกำหนดไว้

บิต 5 โหมดไอบีเอ็ม เป็นการบอกว่าแป้นพิมพ์เป็นของไอบีเอ็ม ให้บิตนี้เป็น "1" คอนโทรลเลอร์ จะไม่ตรวจสอบพาริตี้หรือแปลงรหัสสแกน

บิต 4 ดิสเอเบิลคีย์บอร์ด ถ้าเขียนด้วย "1" จะเป็นการดิสเอเบิล โดยการทำให้สายสัญญาณนาฬิกาเป็น "0" ข้อมูลจะรับส่งกันไม่ได้

บิต 3 ถ้าเป็น "1" จะทำให้ฟังก์ชันการอินฮิบิตคีย์บอร์ดทำงานไม่ได้

บิต 2 แฟลกระบบ ถ้าเขียนอะไรไปจะอ่านได้ที่รีจิสเตอร์สถานะ

บิต 1 สงวนไว้ ปกติให้ค่า "0"

บิต 0 อีนาเบิลอินเตอร์รัพท์ของบัฟเฟอร์เอาต์พุต ถ้าเขียน "1" หมายถึงอีนาเบิลให้อินเตอร์รัพท์เกิดขึ้นได้

AA ทดสอบตัวเองเพื่อหาข้อผิดพลาด โดยได้รหัส **55H** ที่บัฟเฟอร์เอาต์พุต หมายถึงไม่มีข้อผิดพลาด

AB ทดสอบอินเตอร์เฟส โดยทดสอบสายข้อมูลและสัญญาณนาฬิกา ผลลัพธ์ของการทดสอบจะปรากฏที่เอาต์พุตบัฟเฟอร์ดังนี้

00 ไม่มีข้อผิดพลาด

01 สายสัญญาณนาฬิกาแสดงสถานะ "0" ตลอด

02 สายสัญญาณนาฬิกาแสดงสถานะ "1" ตลอด

03 สายข้อมูลคีย์บอร์ดแสดงสถานะ "0" ตลอด

04 สายสัญญาณข้อมูลคีย์บอร์ดแสดงสถานะ "1" ตลอด

AC คัมพ์ข้อตรวจสอบระบบคีย์บอร์ด เป็นการส่งข้อมูล 16 ไบต์จากหน่วยความจำภายใน 8042 สถานะของพอร์ตอินพุต สถานะปัจจุบันของพอร์ตเอาต์พุต สถานะของคอนโทรลเลอร์ข้อมูลทั้งหมดจะส่งมาในฟอร์แมตของรหัสสแกน

AD ดิสเอเบิลคีย์บอร์ด คำสั่งนี้จะเซตบิต 4 ของคำสั่งที่สั่งคอนโทรลเลอร์ เพื่อดิสเอเบิลการรับส่งข้อมูล

AE อีนาเบิลการเชื่อมต่อคีย์บอร์ด คำสั่งนี้จะเคลียร์บิต 4 ของไบต์คำสั่งเพื่อให้รับส่งข้อมูลได้

C0 อ่านพอร์ตอินพุต คำสั่งนี้ทำให้คอนโทรลเลอร์อ่านพอร์ตอินพุต แล้วนำข้อมูลใส่ไว้ในบัฟเฟอร์เอาต์พุต คำสั่งนี้จะใช้ได้ต่อเมื่อบัฟเฟอร์เอาต์พุตว่างอยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

D0 อ่านพอร์ตเอาต์พุต คำสั่งนี้ทำให้คอนโทรลเลอร์อ่านพอร์ตเอาต์พุตแล้วเก็บข้อมูลไว้ในบัฟเฟอร์เอาต์พุต คำสั่งนี้ใช้ได้ต่อเมื่อบัฟเฟอร์เอาต์พุตว่างอยู่

D1 เขียนพอร์ตเอาต์พุต ข้อมูลไบต์ต่อมาที่เขียนในพอร์ต **60H** จะใส่ลงไปในพอร์ตเอาต์พุตของคอนโทรลเลอร์ (8042)

อนึ่งบิต 0 ของพอร์ตเอาต์พุตของคอนโทรลเลอร์ต่อกับรีเซตของระบบ

E0 อ่านอินพุตทดสอบ คำสั่งนี้ทำให้คอนโทรลเลอร์อ่าน T_0 และ T_1 และข้อมูลที่ได้จะเก็บไว้ที่บัฟเฟอร์เอาต์พุต บิตข้อมูลเป็น 0 แทน T_0 บิตข้อมูลเป็น 1 แทน T_1

F0-FF พัลส์ของพอร์ตเอาต์พุต บิต 0 ถึงบิต 3 ของเอาต์พุตของพอร์ตคอนโทรลเลอร์ส่งเป็นพัลส์ออกไปโดยมีพัลส์ลอจิก "0" ประมาณ 6 ไมโครวินาที บิต 0-3 ของคำสั่งนี้แสดงว่าจะให้พัลส์ไปปรากฏที่บิตใดของเอาต์พุต ถ้าเป็น 0 แสดงว่าจะมีพัลส์

อนึ่งบิต 0 ของพอร์ตเอาต์พุตเป็นการต่อรีเซต ถ้าส่งพัลส์ออกทางบิตนี้จะทำให้รีเซตระบบ

2.15 วงจรกำเนิดเวลาจริงและซีมอสแรม

พีซีเอทีมีข้อแตกต่างกับพีซีเอ็กซ์ทีคือ มีนาฬิกาจริงอยู่ที่สามารถปรับปรุงเวลาเองอย่างอัตโนมัติ มีแบตเตอรี่สำรองอยู่ ทำให้นาฬิกาเดินได้ตลอดเวลา ชิพที่ได้ในที่นี้คือชิพรุ่น **MC146818** โดยมี **RAM** อยู่ภายในเพื่อเก็บข้อมูลสถานะและเวลา การเลือกชิพซีมอสเพราะต้องการให้ใช้กำลังงานต่ำมาก จึงใช้กับแบตเตอรี่แบบลิเทียมก้อนเล็ก ๆ ได้นานเป็น ปี ๆ

วงจร **MC146818** นี้ต่อเข้ากับไฟฟ้าสำรองแบตเตอรี่ โดยมี **MC14069** เป็นวงจรกำเนิดสัญญาณฐานเวลา 32.768 กิโลเฮิรตซ์ ดังนั้นวงจรสำรองไฟฟ้าจึงต้องสนองตอบต่อชิพทั้งสองนี้ได้ตลอดเวลา แม้จะเลิกจ่ายไฟฟ้าให้กับระบบแล้ว

MC146818 นี้จะส่งสัญญาณอินเตอร์รัพต์ทาง **IRQ8** ให้กับระบบอย่างต่อเนื่องเพื่อให้ซีพียูอ่านข้อมูลเวลาจากชิพ เพื่อนำไปปรับปรุงเวลาของระบบ

โปรแกรมที่เขียนทางด้าน 80286 นี้จะมองเห็นแรมในชิพนี้ 64 ไบต์ โดย 14 ไบต์แรกเป็นเวลาและปฏิทิน

การอ่านซีมอสแรม **(CMOS RAM)** จะทำได้ด้วยการส่งค่าแอดเดรสของซีมอสแรมออกไปทางพอร์ต **70H** เสียก่อน ค่าของแอดเดรสจะได้รับการแลตช์ไว้ภายในหลังจากนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จึงเขียนหรืออ่านทางพอร์ตหมายเลข 71H

ข้อมูลในซีมอสเรมนี้มีความสำคัญมาก ดังนั้นการเขียนข้อมูลหรือปรับปรุงข้อมูลบางไบต์จำเป็นต้องแก้ไขไบต์ตรวจสอบผลรวมด้วย การเก็บข้อมูลทั้ง 64 ไบต์เป็นดังนี้

ข้อมูลเกี่ยวกับนาฬิกา ไบต์ 00-0D จำนวน 14 ไบต์ (ไม่นำข้อมูลมาตรวจสอบผลรวม)

ไบต์ 00 เวลาหน่วยวินาที

ไบต์ 01 หลักวินาทีที่ตั้งเวลาไว้

ไบต์ 02 เวลาหน่วยนาที

ไบต์ 03 หลักนาทีที่ตั้งเวลาไว้

ไบต์ 04 หลักชั่วโมง

ไบต์ 05 หลักชั่วโมงที่ตั้งเวลาไว้

ไบต์ 06 วันที่ในสัปดาห์

ไบต์ 07 วันที่ของเดือน

ไบต์ 08 เดือน

ไบต์ 09 ปี

ไบต์ 0A รีจิสเตอร์แสดงสถานะ A

ไบต์ 0B รีจิสเตอร์แสดงสถานะ B

ไบต์ 0C รีจิสเตอร์แสดงสถานะ C

ไบต์ 0D รีจิสเตอร์แสดงสถานะ D

ไบต์ 0E เป็นไบต์แสดงสถานะการตรวจสอบ เช่น ตรวจสอบแหล่งจ่ายไฟเสื่อมหายไปจากการจ่ายให้นาฬิกา ตรวจสอบผลรวมการติดตั้งระบบผิดจากที่กำหนด ขนาดของหน่วยความจำอะแดปเตอร์ฮาร์ดดิสก์เวลาที่อ่านได้ผิดพลาด

ไบต์ 0F แสดงสถานะของไฟเสื่อม

ไบต์ 10 แสดงสถานะการติดตั้งฟลอปปี้ดิสก์แบบ 1.2 MB หรือ 360 KB หรืออื่น ๆ

ไบต์ 11 สงวนไว้

ไบต์ 12 แสดงการติดตั้งฮาร์ดดิสก์โดยบอกชนิดของฮาร์ดดิสก์

ไบต์ 13 สงวนไว้

ไบต์ 14 บอกสถานะของคอนฟิกูเรชั่นของระบบ

ไบต์ 15,16 บอกขนาดของหน่วยความจำบนเมนบอร์ด

ไบต์ 17,18 บอกขนาดหน่วยความจำที่ขยาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไบต์ 19 ถึง 2D สงวนไว้

ไบต์ 2E และ 2F ใช้สำหรับตรวจสอบ **check sum**

ไบต์ 30 และ 31 บอกขนาดหน่วยความจำที่ขยาย

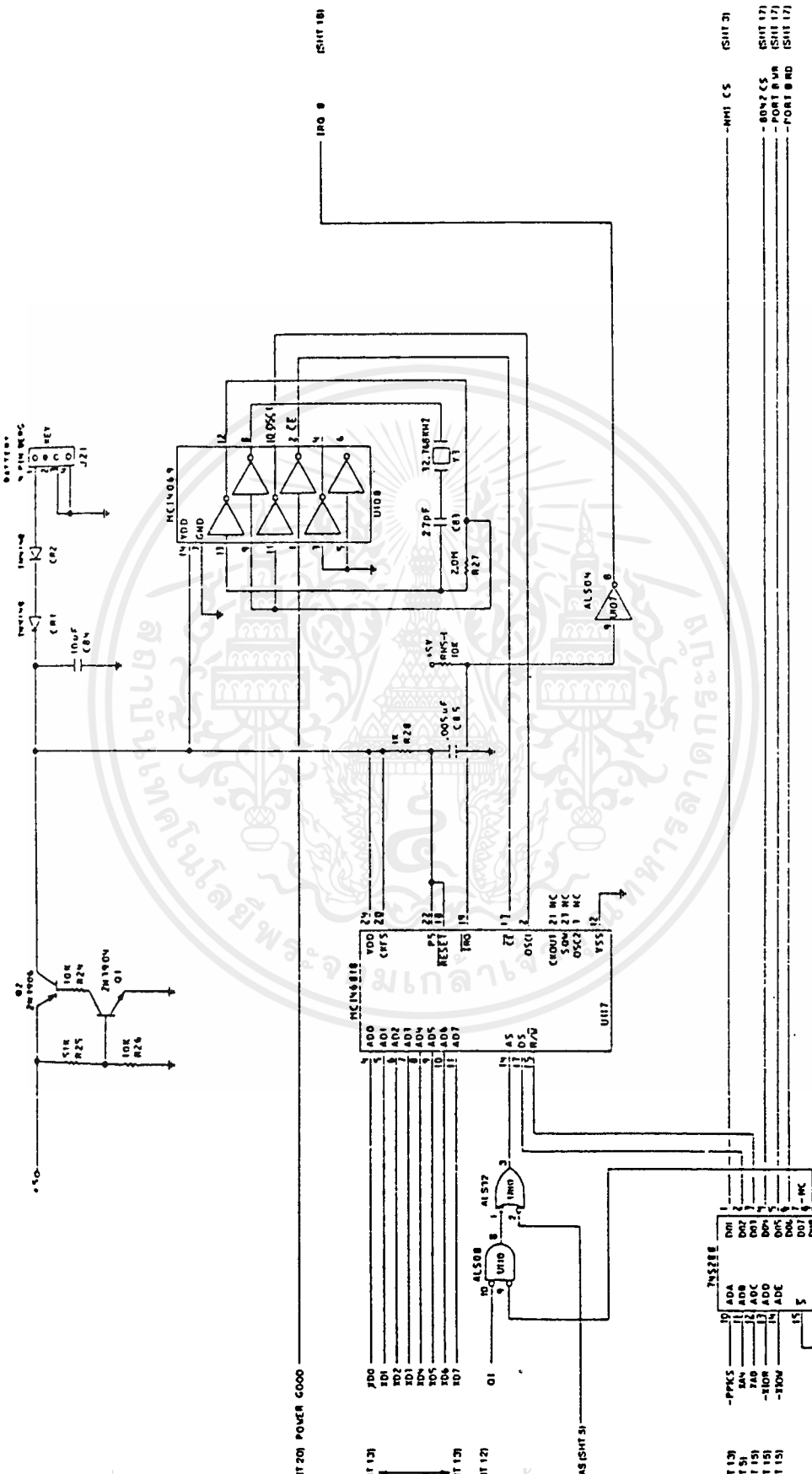
ไบต์ 32 บอกตัวเลข BCD เป็นศตวรรษให้ไบออสใช้

ไบต์ 33 แฟลกข่าวสาร

ในการใช้งานพีซีเอทีไบออสจะตรวจสอบหน่วยความจำนี้ทุกครั้งที่เริ่มต้นระบบ หากแบตเตอรี่หายไปทำให้ข้อมูลในซีมอสเรมหายไป จะทำให้ระบบไม่สามารถทำงานได้

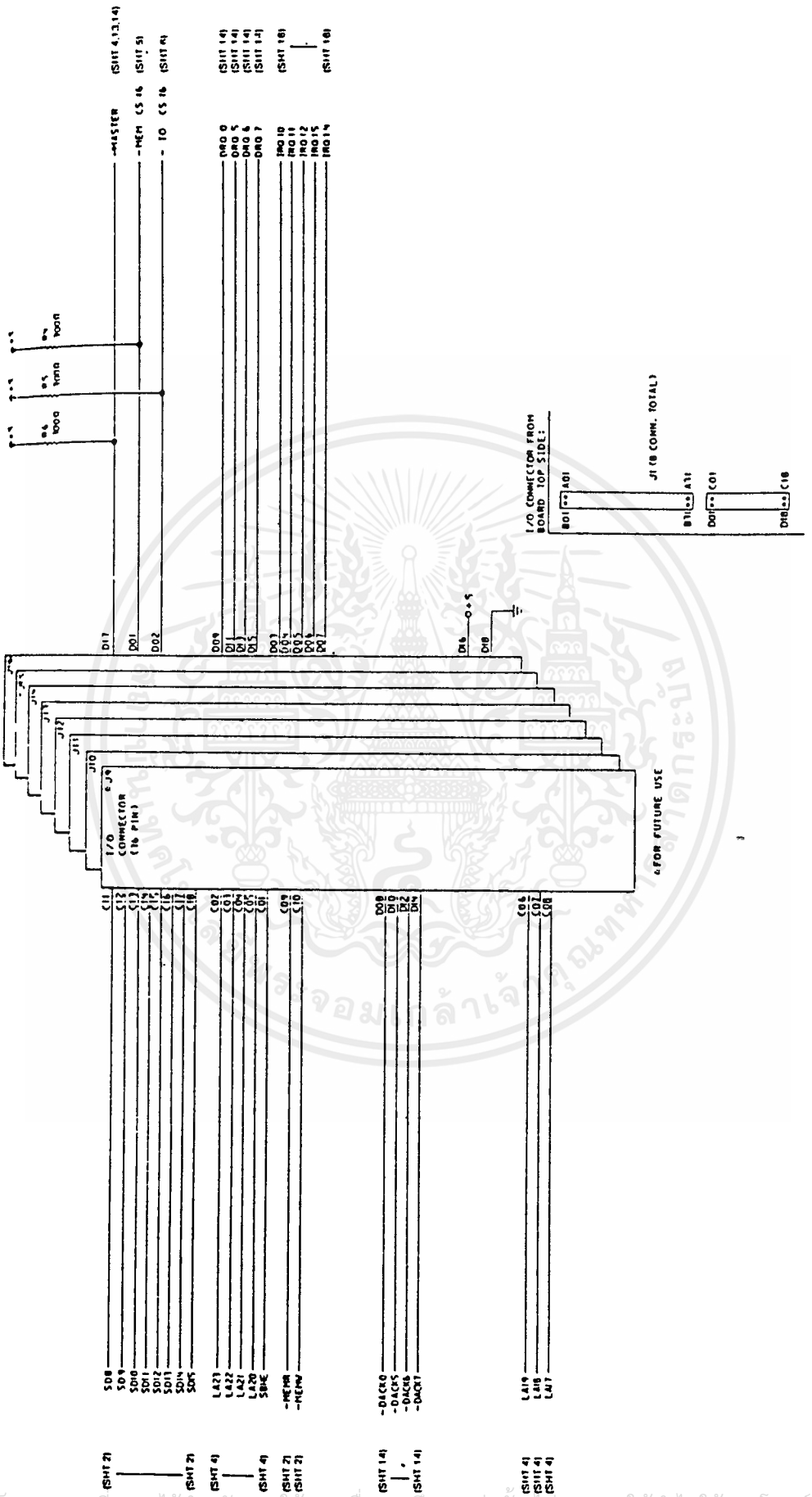
การทำงานใด ๆ ของระบบจะได้รับการติดตั้งค่าเริ่มต้นไว้ที่ไบออส โปรแกรมไบออสจะกำหนดค่าต่าง ๆ เหล่านี้เอาไว้ การศึกษาให้เข้าใจระบบได้ดีควรจะต้องรู้เรื่องฮาร์ดแวร์คือ วงจรของระบบและดูโหมดการทำงานต่าง ๆ ในโปรแกรมไบออส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



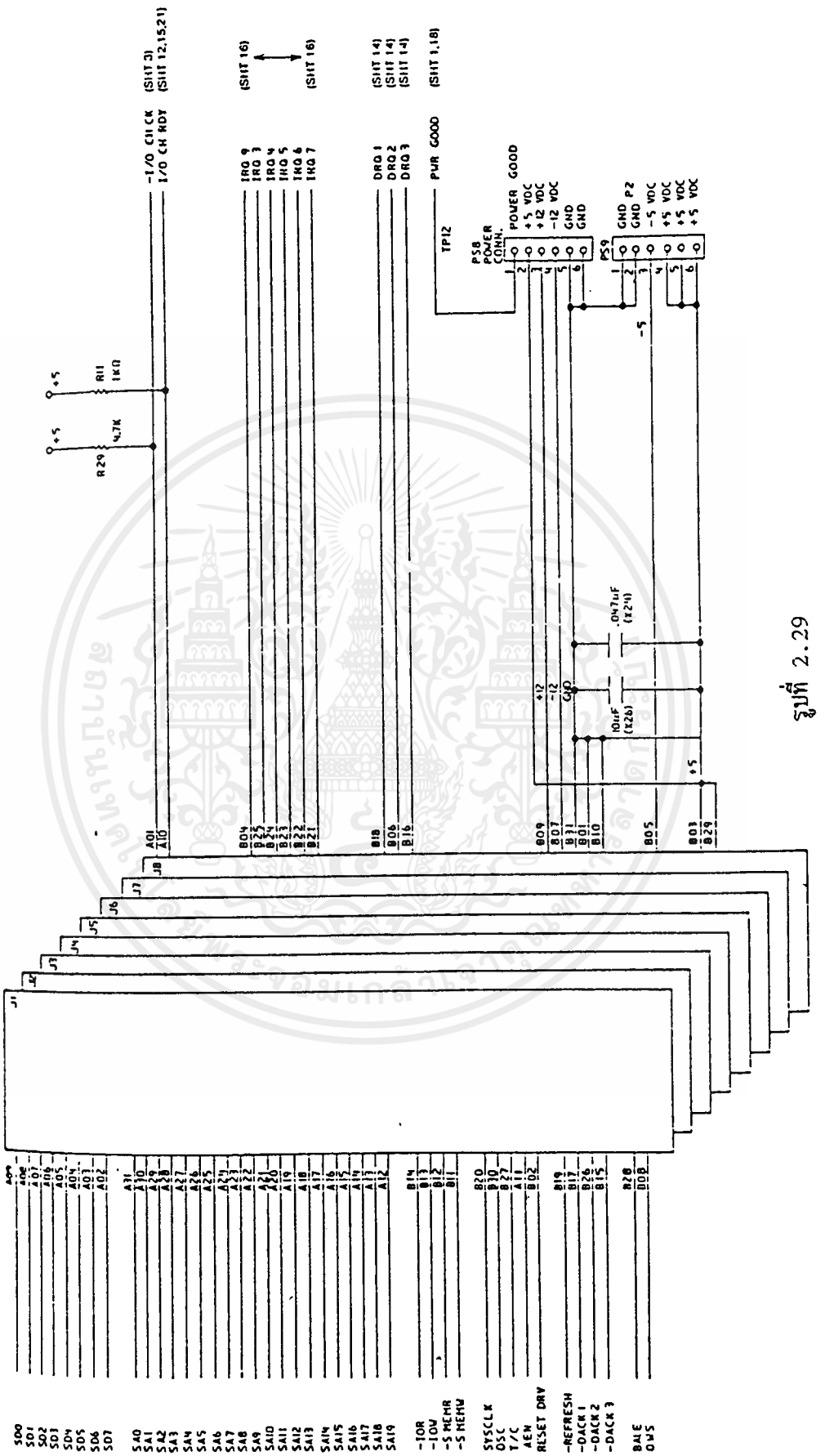
รูปที่ 2.27

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



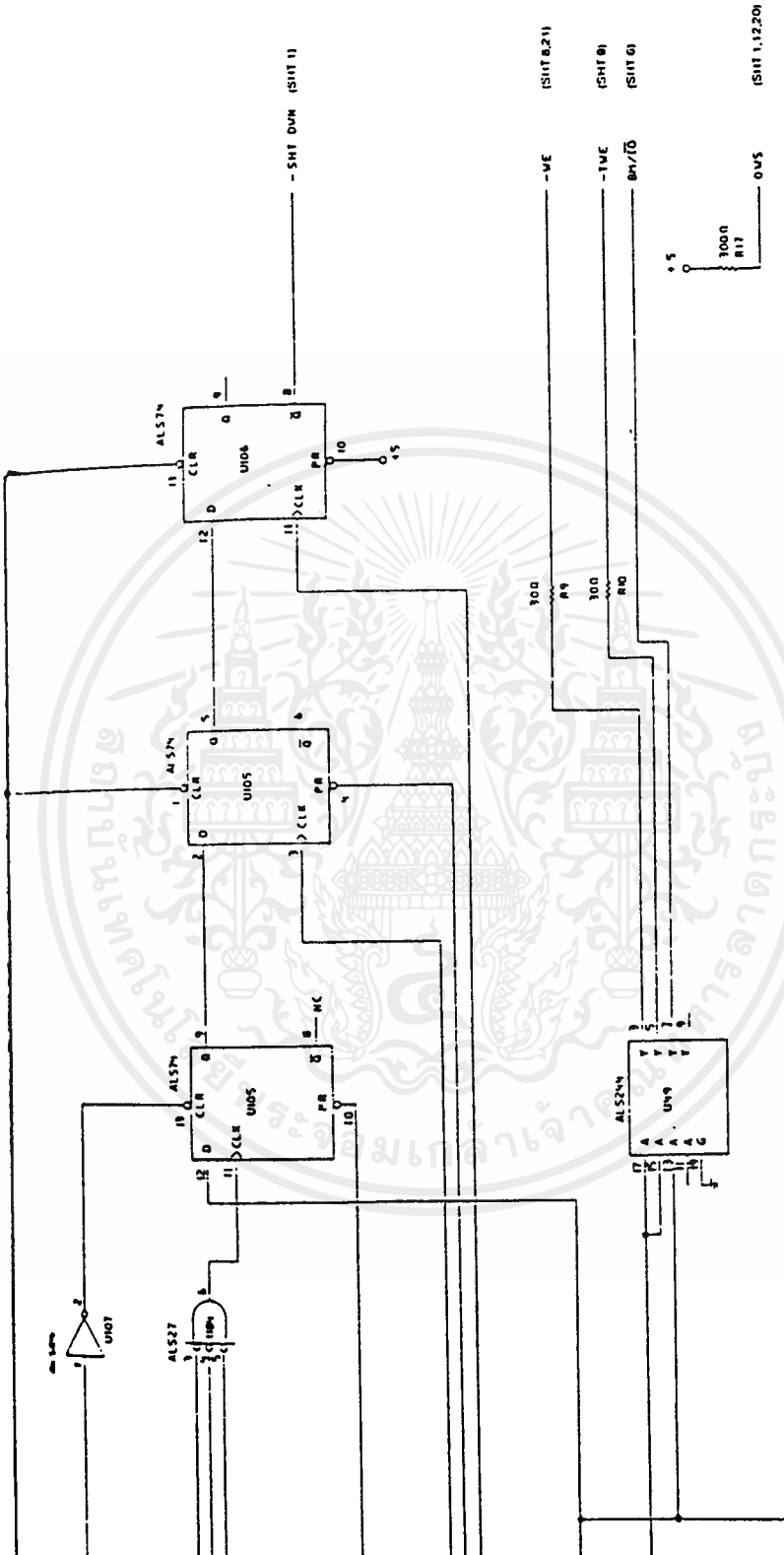
รูปที่ 2.28

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.29

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.30

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
อินเตอร์เฟส การ์ด

3.1 บทนำ

ปัจจุบัน คอมพิวเตอร์ได้มีบทบาทในสังคมเพิ่มมากขึ้น ช่วยในการอำนวยความสะดวก และความบันเทิง ทั้งในชีวิตประจำวัน และในการทำงานต่าง ๆ ตลอดจนในงานอุตสาหกรรม ทั้งนี้เพราะคอมพิวเตอร์ทำงานได้อย่างรวดเร็ว และถูกต้อง ดังนั้น การศึกษาการทำงานของเครื่องคอมพิวเตอร์ และศึกษาลักษณะการนำเครื่องคอมพิวเตอร์มาใช้ในงานต่าง ๆ จึงมีความสำคัญ และจำเป็นมากยิ่งขึ้น

ในบทนี้จะกล่าวถึงหลักการในการนำเอาคอมพิวเตอร์ มาประยุกต์ใช้งาน ซึ่งจำเป็นที่จะต้องรู้จักกับการเชื่อมต่อระหว่างอุปกรณ์ภายนอก โดยใช้หลักการ Interface ซึ่งจำเป็นที่จะต้องทราบถึงโครงสร้างของ เครื่องคอมพิวเตอร์ และอุปกรณ์ชิ้นส่วนต่าง ๆ ที่นำมาใช้พอสมควร

รวมถึงการจัดแอดเดรสสำหรับหน่วยความจำและพอร์ตไอโอ ต่าง ๆ ภายในไอบีเอ็ม พีซี ซึ่งจะแสดงถึงแอดเดรสต่างๆที่ถูกใช้งานโดยไอโอ พอร์ต และหน่วยความจำนอกจากนี้จะได้กล่าวถึง เทคนิคการดีโค้ดในแบบต่างๆ ด้วย

3.2 การจัดแอดเดรสสำหรับอินพุตเอาท์พุตพอร์ต ในไอบีเอ็ม พีซี

ในการควบคุมและตรวจสอบสภาวะการทำงาน รวมทั้งการอ่านข้อมูลจากอุปกรณ์ที่เป็นชิปชับพอร์ตหรือการ์ดต่างๆที่ใช้ในระบบของ ไอบีเอ็ม พีซี นั้น จะกระทำโดยผ่านทางพอร์ตไอโอของระบบ ดังนั้นในการที่จะใช้งานหรือควบคุมการทำงานของอุปกรณ์เหล่านี้ จึงจำเป็นต้องศึกษาถึงวิธีการควบคุมพอร์ตไอโอต่างของระบบด้วย และเนื่องจากการควบคุมหรือติดต่อกับ พอร์ตเหล่านี้ต้องกระทำโดยการอ้างถึงแอดเดรสของพอร์ตไอโอ เหล่านั้นโดยตรง เราจึงจำเป็นต้องศึกษาถึงหลักการอ้างแอดเดรสบนไอบีเอ็ม พีซี ด้วย

การอ้างแอดเดรสบนไอบีเอ็ม พีซี เอที (ใช้ซีพียู 80286) ถูกออกแบบมาให้ใช้แอดเดรสของไอโอพอร์ตได้ 10 บิท คือ A0-A9 เท่านั้น จึงสามารถที่จะอ้างแอดเดรสของไอโอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ในการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พอร์ตสูงสุดเพียง 1024 พอร์ต ใน 1024 พอร์ตนี้ยังแบ่งออกเป็น 2 ส่วน(ส่วนละ512 พอร์ต) โดยใช้ข้อมูลในบิท A9 ถ้าA9เป็น "0" เราจะอ่านข้อมูลได้เฉพาะพอร์ตของซิปซัพพอร์ตต่าง ๆ ที่อยู่บนเมนบอร์ดของไอบีเอ็ม พีซี แต่ถ้า A9 เป็น "1" ก็จะอ่านข้อมูลได้เฉพาะจากพอร์ตของการ์ดต่าง ๆ เท่านั้น

ดังนั้นถ้าเราต้องการสร้างแผ่นเชื่อมโยงกับไอบีเอ็ม พีซี จึงต้องเลือกแอดเดรสสำหรับพอร์ตที่อยู่บนการ์ดตามที่เสดงไว้ในตารางที่ 3.1 สำหรับโครงงานปริญญานิพนธ์นี้จะเลือกแอดเดรสของพอร์ตช่วง 300H-31FH ซึ่งใช้กับโปรโตไทป์การ์ด ทั้งนี้เพื่อความสะดวกของผู้ออกแบบและป้องกันการอ้างพอร์ตซ้ำกับการ์ดที่มีอยู่ แต่ก็สามารถใช้งานในเบอร์พอร์ตได้ถ้าไม่มีการ์ดนั้นติดตั้งอยู่ในระบบ

ตารางที่ 3.1 การใช้งานแอดเดรสสำหรับพอร์ตบนไอบีเอ็ม พีซี

หมายเลขพอร์ต	ชื่ออุปกรณ์
000H - 01FH	ดีเอ็มเอคอนโทรลเลอร์หมายเลข1,8237A-5
020H - 03FH	อินเตอร์รัพท์คอนโทรลเลอร์หมายเลข 1,8259A
040H - 05FH	ไทเมอร์ 8254-2
060H - 06FH	8042 คีย์บอร์ด
070H - 07FH	นาฬิกา และ NMI และซีมอสแรม
080H - 09FH	DMA เพจรีจิสเตอร์
0A0H - 0BFH	อินเตอร์รัพท์คอนโทรลเลอร์หมายเลข2,8259A
0C0H - 0DFH	ดีเอ็มเอคอนโทรลเลอร์หมายเลข 2,8237A-5
0F0H	เคลียร์โปรเซสเซอร์คณิตศาสตร์
0F1H	รีเซ็ตโปรเซสเซอร์คณิตศาสตร์
0F8H - 0FFH	โปรเซสเซอร์คณิตศาสตร์
1F0H - 1F8H	ฮาร์ดดิสค์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.1 การใช้งานแอดเดรสสำหรับพอร์ตบนไอบีเอ็ม พีซี (ต่อ)

หมายเลขพอร์ต	ชื่ออุปกรณ์
200H - 207H	เกมไอโอ
278H - 27FH	พอร์ตเครื่องพิมพ์หมายเลข2
28FH - 2FFH	พอร์ตอนุกรมหมายเลข 2
300H - 31FH	โปรโตไทป์การ์ด
360H - 36FH	สำรอง
378H - 37FH	พอร์ตเครื่องพิมพ์หมายเลข 1
380H - 38FH	SDLC, ไบซิงค์ 2
3A0H - 3AFH	ไบซิงค์ 1
3B0H - 3BFH	โมโนโครัมและเครื่องพิมพ์
3C0H - 3CFH	สำรอง
3D0H - 3DFH	จอภาพสี
3F0H - 3F7H	ควบคุมดิสเกตต์
3F8H - 3FFH	พอร์ตอนุกรมหมายเลข 1

3.3 เทคนิคการดีโค้ดแอดเดรส

วงจรดีโค้ดแอดเดรสถือว่าเป็นหัวใจสำคัญของการอินเตอร์เฟสระหว่างเมนบอร์ดกับอุปกรณ์ไอโอพอร์ต จะอยู่ร่วมกันเพื่อทำหน้าที่สองอย่าง คือ กำหนดให้เป็นพอร์ตอินพุตอ่านข้อมูลหรือให้พอร์ตเอาท์พุตเขียนข้อมูลบนอุปกรณ์ อินพุต/เอาท์พุต สัญญาณที่จะต้องนำมาใช้เพื่อเป็นตัวเลือกกระบวนการที่เหมาะสมที่ CPU จะติดต่อกับอุปกรณ์ อินพุต/เอาท์พุต หรืออุปกรณ์อินพุต/เอาท์พุตติดต่อกับ CPU คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แอดเดรส ใช้ตั้งแต่ A_0 - A_9 (10 บิท) สำหรับแอดเดรสพอร์ต

IOR แอคทีฟที่ลอจิค "0" เพื่อบอกว่าขณะนี้เป็นการอ่านข้อมูลจากอินพุตพอร์ต

IOW แอคทีฟที่ลอจิค "0" เพื่อบอกว่าขณะนี้เป็นการเขียนข้อมูลไปยังเอาท์พุตพอร์ต

AEN แอคทีฟที่ลอจิค "1" เพื่อดิสเอเบิ้ลวงจรอื่น ๆ ขณะเกิดขบวนการ DMA ในการดีโค้ดจะใช้ลอจิค "0"

รูปที่ 3.1 แสดงถึงพื้นฐานของการดีโค้ดซึ่งแยกออกเป็น 3 ส่วน ส่วนแรก เป็นการนำแอดเดรส A0 - A9 มาทำการดีโค้ดสร้างสัญญาณ - ADDRESS SEL และนำสัญญาณนี้ไปรวมกับ -AEN ได้สัญญาณ -I/O SELECT เพื่อแยกว่าเป็นอุปกรณ์ไอโอ ถ้าใช้สัญญาณนี้ร่วมกับ -IOR จะได้สัญญาณสำหรับเลือกอุปกรณ์อินพุต -IOW จะเป็นสัญญาณสำหรับเลือกอุปกรณ์เอาท์พุต

เทคนิคการดีโค้ดสามารถแบ่งได้ 2 กลุ่มดังนี้

3.3.1 เทคนิคในการดีโค้ดแอดเดรสแบ่งตามการใช้อุปกรณ์

1. การดีโค้ดแบบใช้ไอซีลอจิกเกทธรรมดา เช่น AND,OR,NOR,NAND และ NOT GATE ในการดีโค้ดมักใช้เมื่อต้องการพอร์ตเพียงพอร์ตเดียว

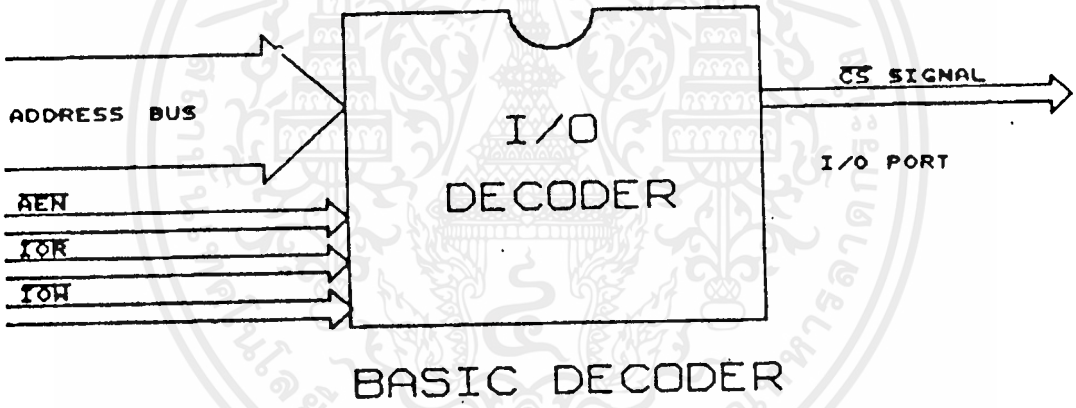
2. การดีโค้ดแบบ Fixed ใช้ไอซีดีโค้ดเบอร์ต่าง ๆ เช่น 74LS138,74 LS139 ฯลฯ วิธีการดีโค้ดแบบนี้เป็นวิธีที่ง่ายและสะดวกมักใช้เมื่อต้องการพอร์ตตั้งแต่ 4 พอร์ตขึ้นไป

3. การดีโค้ดแบบใช้สวิทช์เลือกโดยใช้ไอซีเปรียบเทียบระหว่าง 2 อินพุต เช่น EXCLUSIVE-OR หรือไอซีเบอร์ 74LS688 (8 BIT COMPARATOR) การดีโค้ดทั้งสองแบบแรกที่กล่าวมามีข้อเสียคือแอดเดรสที่เราเลือกใช้งานนั้น อาจจะชนกับแอดเดรสของการ์ดอื่นที่เรานำมาเพิ่มเข้าไปในระบบภายหลังก็ได้ และอีกข้อหนึ่งคือ ไม่สามารถเปลี่ยนแปลงแอดเดรสปัญหาเช่นนี้เราสามารถแก้ไขได้โดยใช้วงจรดีโค้ดที่สามารถเปลี่ยนแปลงค่าแอดเดรสได้โดยเพียงแต่เปลี่ยนตำแหน่งของสวิตช์ (DIP SW.)

4. การดีโค้ดโดยใช้ PROM (Programmable Read Only Memory) มักจะใช้เมื่อพอร์ตที่เราต้องการมีแอดเดรสแยกกันอย่างอิสระ ต้องมีการโปรแกรมก่อนนำมาใช้งาน หรืออุปกรณ์ที่ทำการดีโค้ดไม่มากเกินไปจะนำไปใช้ดีโค้ด การทำงานที่อยู่บนการ์ดต่าง ๆ มารวมไว้บนการ์ดเดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. การดีโค้ดโดยใช้ PAL(PROGRAMMABLE ARRAY LOGIC) และ PLA (PRO-GRAMABLE LOGIC ARRAY) เทคนิคการดีโค้ดแบบนี้ต้องเขียนอินพุตและเอาท์พุตให้อยู่ในรูปฟังก์ชั่นลอจิก และนำไปโปรแกรมลงบน PAL ชนิดของ PAL มีให้เลือกตามจำนวนอินพุต เอาท์พุทที่เหมาะสม วิธีการใช้งานนั้นสามารถศึกษาได้จากคู่มือ วิธีนี้สามารถลดจำนวนอุปกรณ์ในวงจรดีโค้ดลงให้เหลือเพียง PAL ตัวเดียว



รูปที่ 3.1 แสดงถึงพื้นฐานของการดีโค้ด

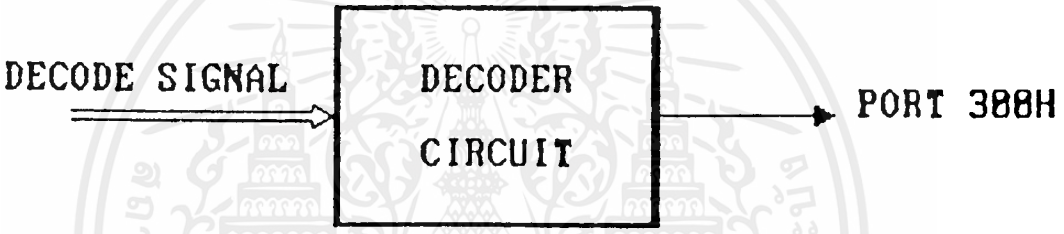
3.3.2 เทคนิคการออกแบบวงจรดีโค้ดตามลักษณะของพอร์ต

การพิจารณาเลือกเทคนิคใดมาใช้ในการดีโค้ด จึงขึ้นอยู่กับพอร์ตที่เราต้องการมีลักษณะอย่างไร บางครั้งก็จำเป็นต้องนำแต่ละเทคนิคมารวมกัน โครงการปริญญา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นิพนธ์นี้จะสนใจเฉพาะ 3 แบบแรกเท่านั้น เพราะมีการใช้งานทั่วไป ส่วนการใช้ PROM และ PAL นั้นมีความยุ่งยากและใช้ในกรณีพิเศษเท่านั้น ดังนั้นจึงจัดการออกแบบวงจรดีโคดแอดเดรสแบบต่าง ๆ เพื่อสร้างพอร์ตที่เราต้องการ ซึ่งพอจะแบ่งลักษณะของพอร์ตได้ 4 ลักษณะดังนี้

1. Single Port Single Address เป็นลักษณะวงจรดีโคดที่สร้างพอร์ตขึ้นมาเพียงพอร์ตเดียวและมีแอดเดรสที่จะติดต่อกับพอร์ตนี้เพียงพอร์ตเดียว มีโครงสร้างดังรูป



รูปที่ 3.2 โครงสร้างของ Single Port Single Address

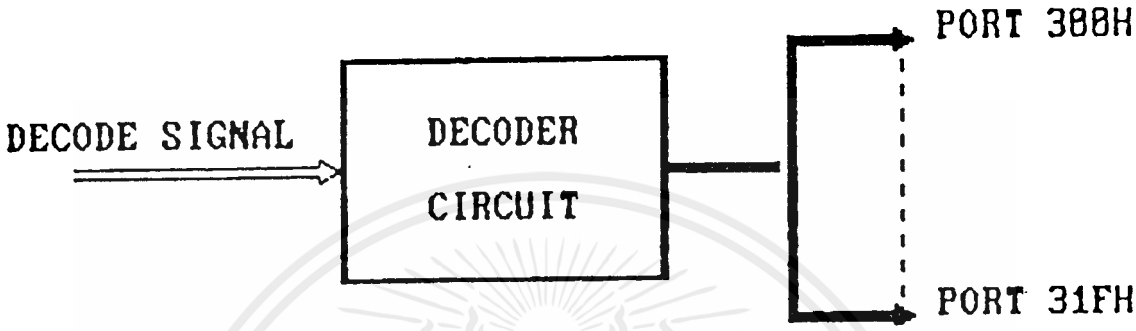
ตัวอย่างที่ 3.1 วงจรนี้มีสัญญาณตั้งแต่ A0 ถึง A9 มาดีโคดแอดเดรสทั้งหมด โดยเรากำหนดการดีโคดออกที่พอร์ต 300H ดังนั้น A0 ถึง A9 จึงต้องมีลอจิกดังนี้

A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
1	1	0	0	0	0	0	0	0	0

ซึ่งสามารถออกแบบวงจรได้ดังรูปที่ 3.2

2. Single Port Multiple Address เป็นลักษณะวงจรดีโคดที่สร้างพอร์ตขึ้นมาพอร์ตเดียว เช่นกันแต่มีแอดเดรสที่จะติดต่อกับพอร์ตนี้ได้หลายเบอร์มีโครงสร้างดังรูปที่ 3.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 โครงสร้างของ Single Port Multiple Address

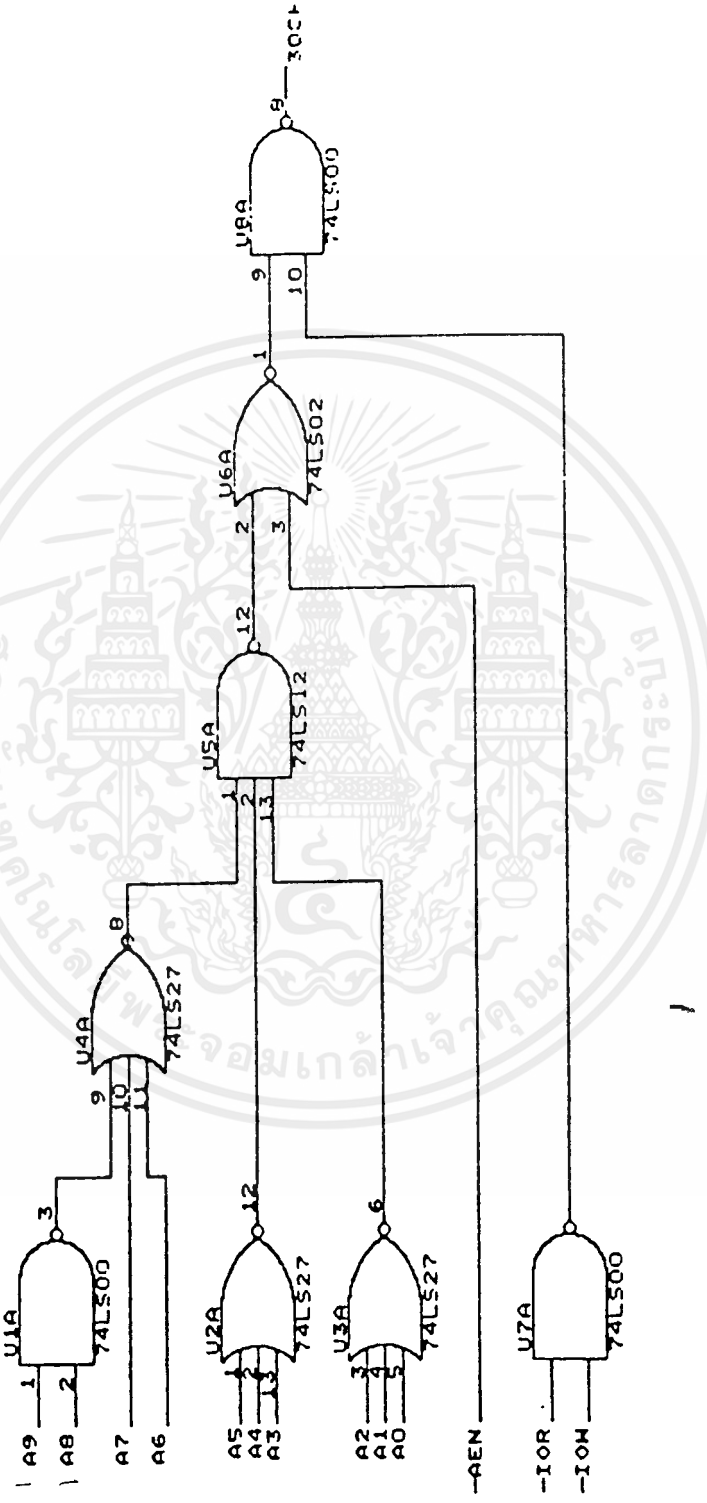
วงจรนี้ เราจึงไม่ได้นำสัญญาณแอดเดรสมาดีโค้ดทั้งหมดโดยจะละช่วงแอดเดรสบิทล่างไว้

ตัวอย่างที่ 3.3 กำหนดให้ดีโค้ดออกพอร์ตในช่วง 300H ถึง 31FH ดังนั้นสัญญาณลอจิกของแอดเดรสมีลักษณะดังนี้

A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
1	1	0	0	0	X	X	X	X	X

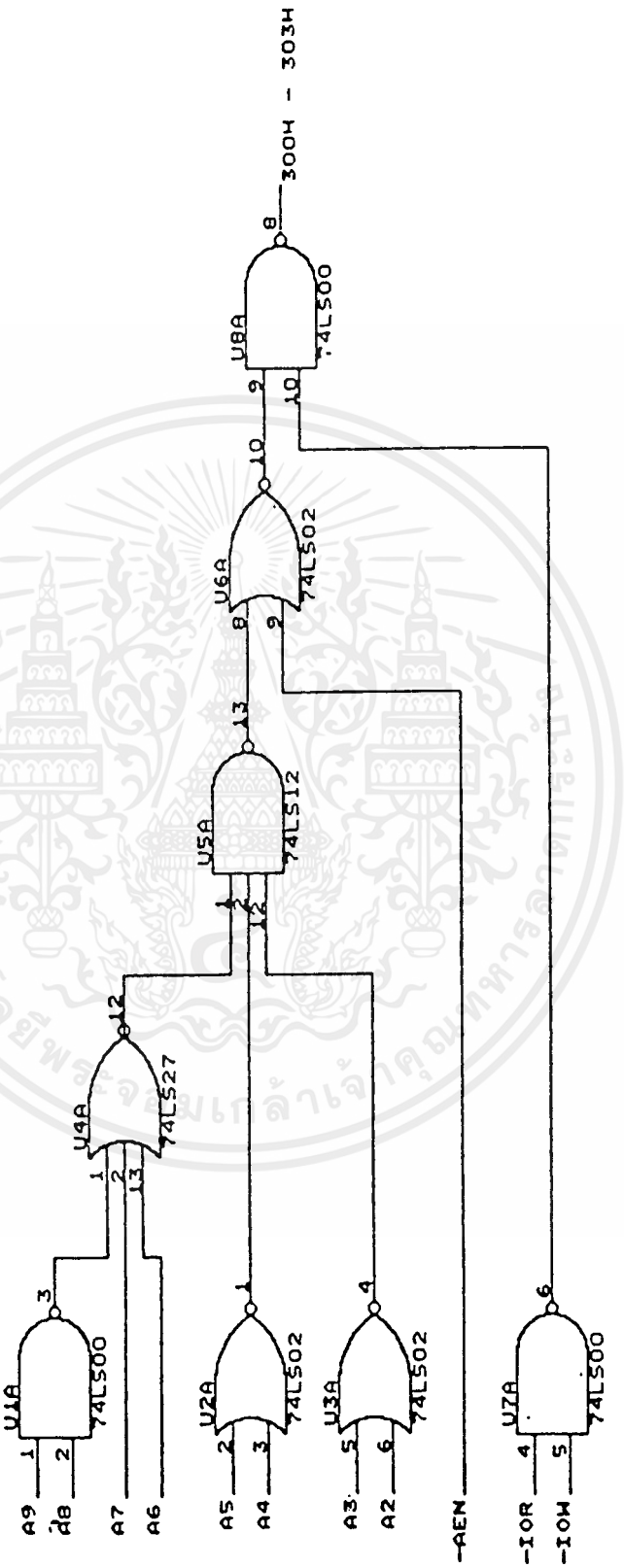
ซึ่งสามารถออกแบบวงจรได้ดังรูปที่ 3.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 วงจร Single Port Single Address

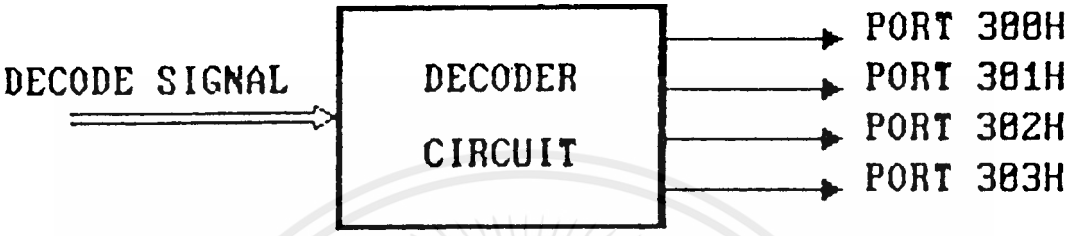
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 วงจร Single port Multiple address

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. Multiple Port Single Address เป็นลักษณะวงจงดีโค้ดที่สร้างพอร์ตขึ้นมาหลายพอร์ต และมีแอดเดรสที่จะติดต่อกับพอร์ตนี้เพียงพอร์ตละหนึ่งแอดเดรสเท่านั้น มีโครงสร้างดังรูปที่ 3.6



รูปที่ 3.6 โครงสร้างของ Multiple Port Single Address

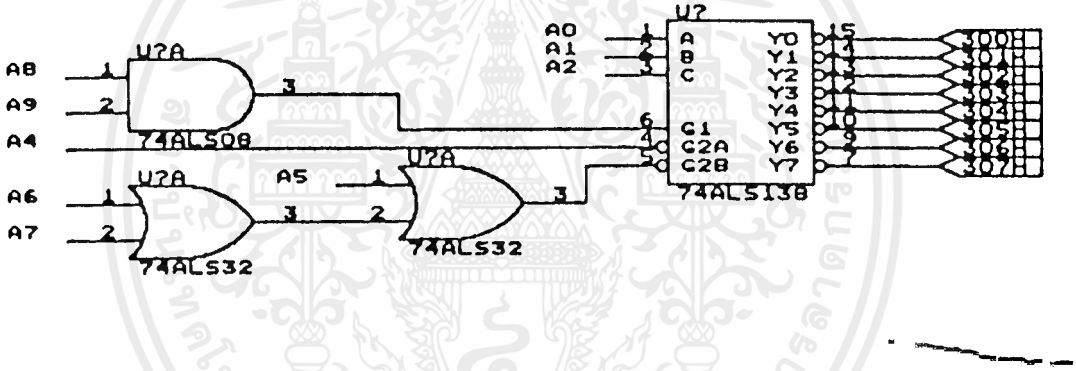
จากวงจรทั้งสองแบบเป็นการดีโค้ดแบบพอร์ตเดียว เรามักใช้ไอซีลอจิกเกทธรรมดาในการดีโค้ด แต่ถ้าเราต้องการใช้งานมากกว่าหนึ่งพอร์ต เช่น 4 พอร์ต เราจะใช้ไอซีดีโค้ดเบอร์ 74LS138 หรือ 74LS139 เลือกช่วงในการดีโค้ด

ตัวอย่างที่ 3.4 กำหนดให้การดีโค้ดแอดเดรสมี 4 พอร์ต คือ พอร์ต 300H,301H,302H 303H,304H,305H เมื่อนำสัญญาณแอดเดรสมาเขียนลอจิกดูจะมีลักษณะดังนี้

A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	
1	1	0	0	0	0	0	0	0	0	PORT 300H
1	1	0	0	0	0	0	0	0	1	PORT 301H
1	1	0	0	0	0	0	0	1	0	PORT 302H
1	1	0	0	0	0	0	0	1	1	PORT 303H
1	1	0	0	0	0	0	1	0	0	PORT 304H
1	1	0	0	0	0	0	1	0	1	PORT 305H

ซึ่งสามารถออกแบบวงจรได้ดังรูปที่ 3.7

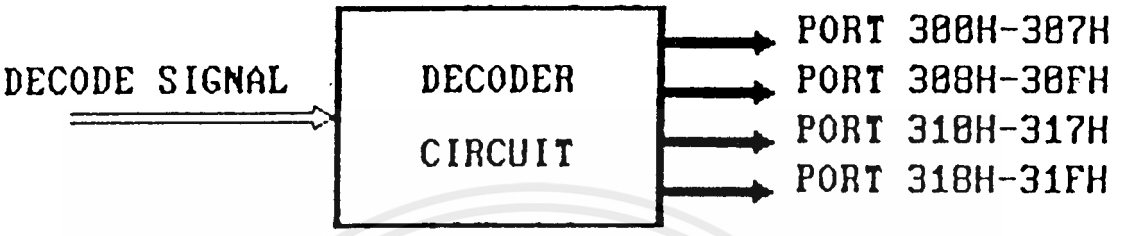
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 วงจร Multiple Port Single Address

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. Multiple Port Multiple Address เป็นลักษณะวงจรดีโค้ดที่สร้างพอร์ตขึ้นมาหลายพอร์ตและมีแอดเดรสที่จะติดต่อกับพอร์ตนี้ได้พอร์ตละหลายแอดเดรส เป็นช่วงแอดเดรสมีโครงสร้างดังรูปที่ 3.8



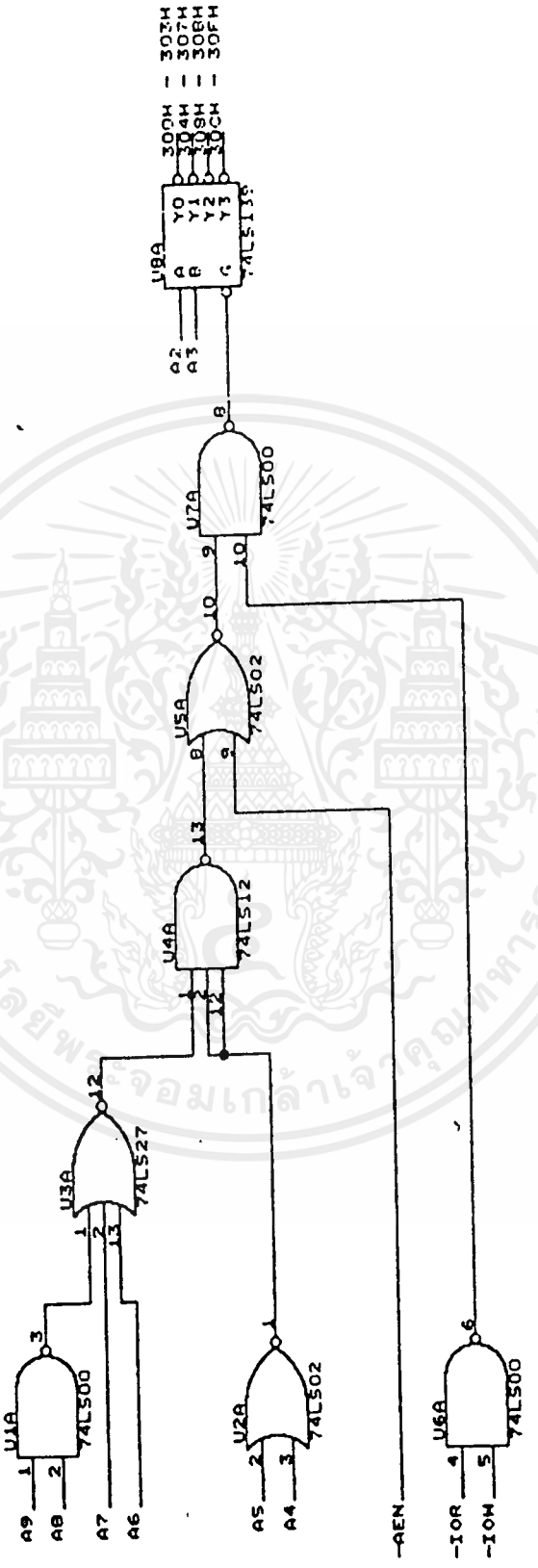
รูปที่ 3.8 โครงสร้างของ Multiple Port Multiple Address

ตัวอย่างที่ 3.6 กำหนดให้การดีโค้ดแอดเดรสมี 8 กลุ่ม ซึ่งแอดเดรสทั้ง 8 กลุ่มแสดงได้ดังตารางข้างล่างนี้

กลุ่ม	แอดเดรส
0(Y0)	2F0H - 2F3H
1(Y1)	2F4H - 2F7H
2(Y2)	2F8H - 2FBH
3(Y3)	2FCH - 2FFH
4(Y4)	3F0H - 3F3H
5(Y5)	3F4H - 3F7H
6(Y6)	3F8H - 3FBH
7(Y7)	3FCH - 3FFH

ซึ่งสามารถออกแบบได้ดังรูปที่ 3.9

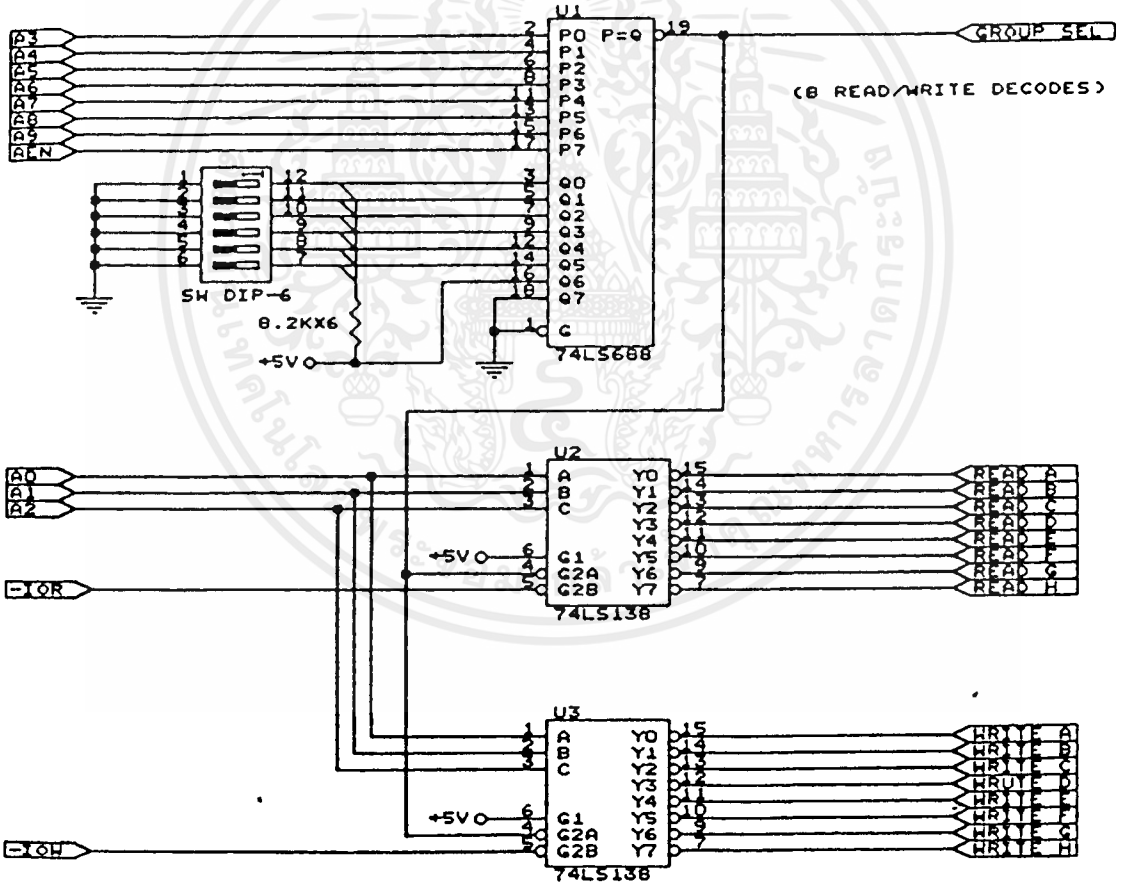
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 วงจรแบบ Multiple Port Multiple Address

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ ห้ามมิให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างที่ 3.7 การออกแบบวงจรที่ทำการดีโคดกลุ่มแอดเดรสขนาด 8 แอดเดรส ซึ่งการเรียกกลุ่มแอดเดรสที่จะทำการดีโคดจะทำได้โดยการเซ็ท Dip Switch ที่ขา Q0 ถึง Q5ของไอซีเบอร์ 74LS688 หน้าที่ของไอซีเบอร์ 74LS688 นี้จะทำการเปรียบเทียบค่าของอินพุต 2 ชุดที่ถูกส่งมาทางขา P0 ถึง P7 และขา Q0 ถึง Q7 ถ้าอินพุตทั้งสองชุดเท่ากันแล้ว เอาท์พุตที่ขา P=Q จะได้เอาท์พุตเป็นลอจิก "0" ดังรูปที่ 3.10



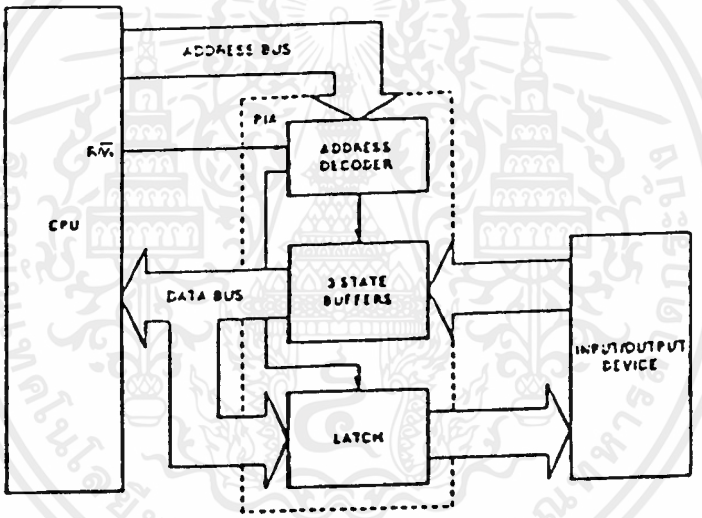
รูปที่ 3.10 วงจรดีโคด Multiple Port Multiple Address

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 การเชื่อมต่ออุปกรณ์อินพุต เอาต์พุตกับระบบไมโครคอมพิวเตอร์

3.4.1 การเชื่อมโยงอุปกรณ์อินพุต เอาต์พุตกับระบบขนาดข้อมูล 8 บิต

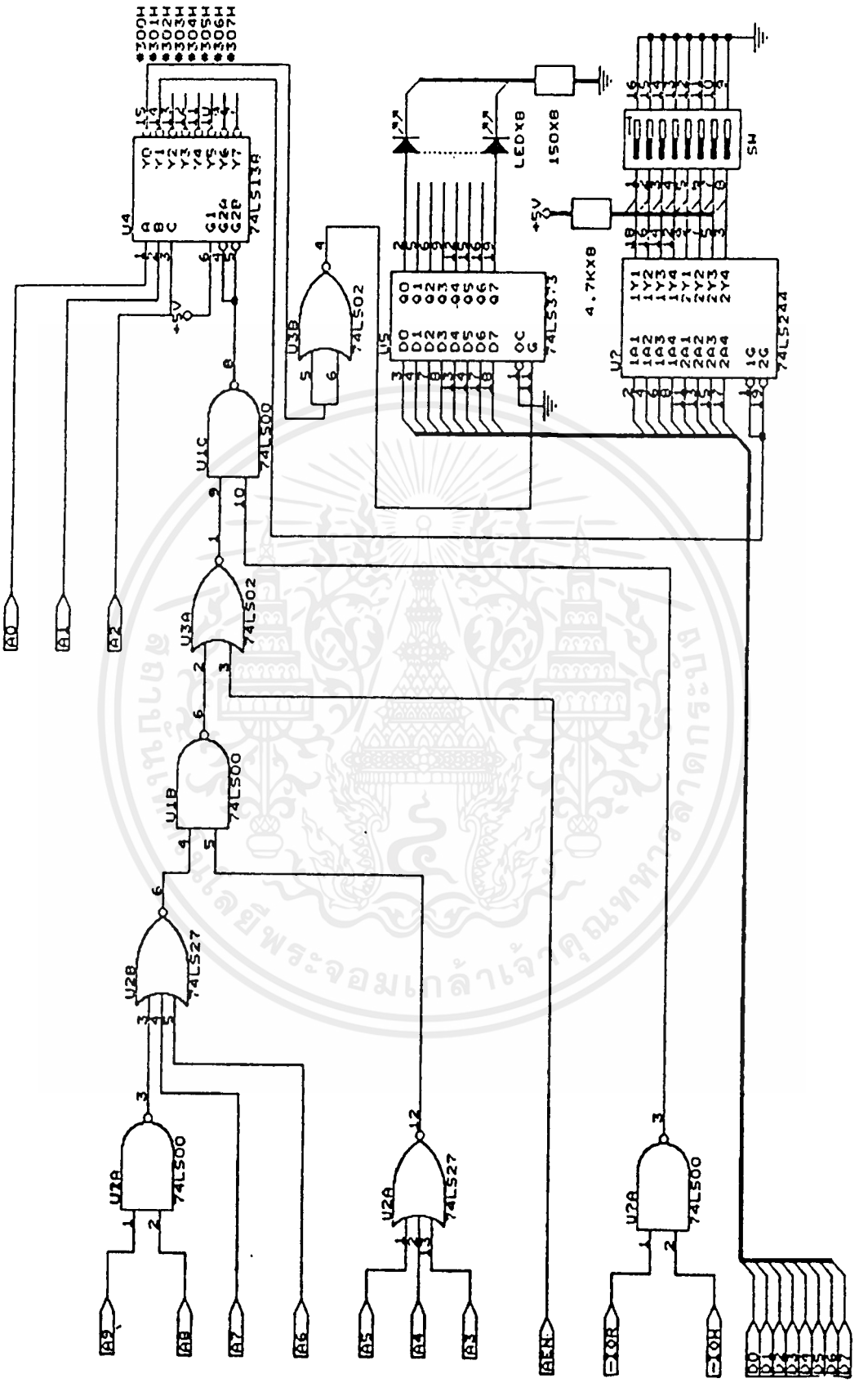
จากหัวข้อ 3.2 เราได้กล่าวถึงเทคนิคการดีโคดแบบต่าง ๆ ต่อไปจะขอกล่าวถึงการเชื่อมต่ออุปกรณ์อินพุต เอาต์พุตกับระบบไมโครคอมพิวเตอร์ เพื่อรับ-ส่งข้อมูลขนาด 8 บิตรูปที่ 3.11 สัญญาณจากวงจรดีโคดใช้เลือกพอร์ตโดยควบคุมการรับ-ส่งข้อมูลด้วย -IOR และ -IOW



รูปที่ 3.11 แสดงการเชื่อมต่ออินพุต เอาต์พุต กับระบบไมโครคอมพิวเตอร์

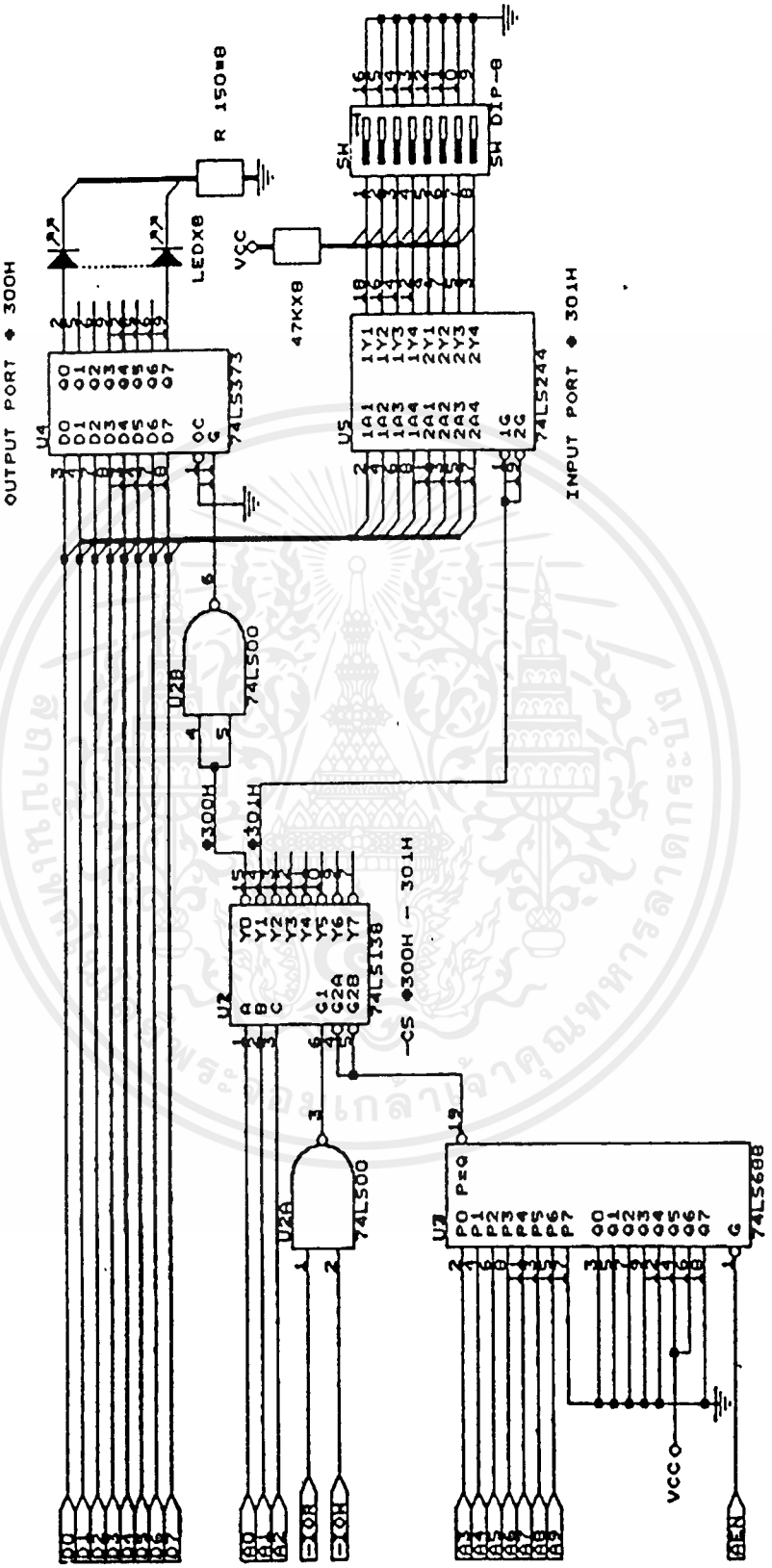
อุปกรณ์ที่ทำหน้าที่เป็นบัฟเฟอร์ข้อมูลทางด้านอินพุต ได้แก่ ไอซีเบอร์ 74LS244,74LS245 ฯลฯ ทางด้านเอาต์พุตเป็นไอซีประเภทแลตช์ข้อมูล เช่น 74LS373,74LS374,74LS377 ฯลฯ อุปกรณ์ทั้ง 2 ชนิดที่กล่าวมานี้สามารถนำมาต่อ เพื่อรับข้อมูลจากดิพสวิตช์ ส่งข้อมูลไปแสดงผลที่หลอดแอลอีดีได้โดยตรง หรือยังสามารถนำมาต่อเป็นบัฟเฟอร์กั้นระหว่างอุปกรณ์อินพุต อินพุตชนิดที่โปรแกรมได้กับบัสข้อมูล รูปที่ 3.12 แสดงถึงวงจรที่ได้ทำการออกแบบ เพื่อทดลองการรับ-ส่ง ข้อมูลขนาด 8 บิต รับข้อมูลทางดิพสวิตช์ และแสดงผลทางแอลอีดี ทดสอบการทำงานด้วยคำสั่งโปรแกรมดีบัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.12 วงจรการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.13 วงจรการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.11 แสดงถึงการใช้ไอซี 74LS688 ร่วมกับไอซีดีโค้ด 74LS138 ในวงจรดีโค้ด ที่อินพุตของไอซี74LS688 จะถูกเซตค่าไว้ให้ตรงกับแอดเดรสที่ต้องการ ซึ่งสามารถเปลี่ยนแปลงค่าได้

3.4.2 การเชื่อมโยงอุปกรณ์อินพุต เอาต์พุตกับระบบขนาดข้อมูล 16 บิต

ในโครงงานนี้มีจุดประสงค์หลักคือ ต้องการสร้างแผ่นวงจรอินเตอร์เฟสบนไอบีเอ็มพีซี เอที เราทราบแล้วว่าการเชื่อมโยงทำได้โดยการสร้างแผ่นวงจรอินเตอร์เฟสเสียบลงในช่องสล๊อตที่ได้เตรียมไว้ โครงสร้างของสล๊อตเอที (เอที บัส) จะมีสัญญาณส่วนที่เพิ่มจากเอ็กซ์ที 36 ขา สัญญาณที่นำใช้มีดังนี้

สัญญาณข้อมูล D8-D15 เป็นสัญญาณข้อมูล 8 บิตบน ที่ใช้ติดต่อกับหน่วยความจำและอุปกรณ์อินพุต เอาต์พุต ถ้ามีการติดต่อกับบัสข้อมูล 8 บิตบน จะสามารถตรวจสอบได้จากสัญญาณ -SBHE

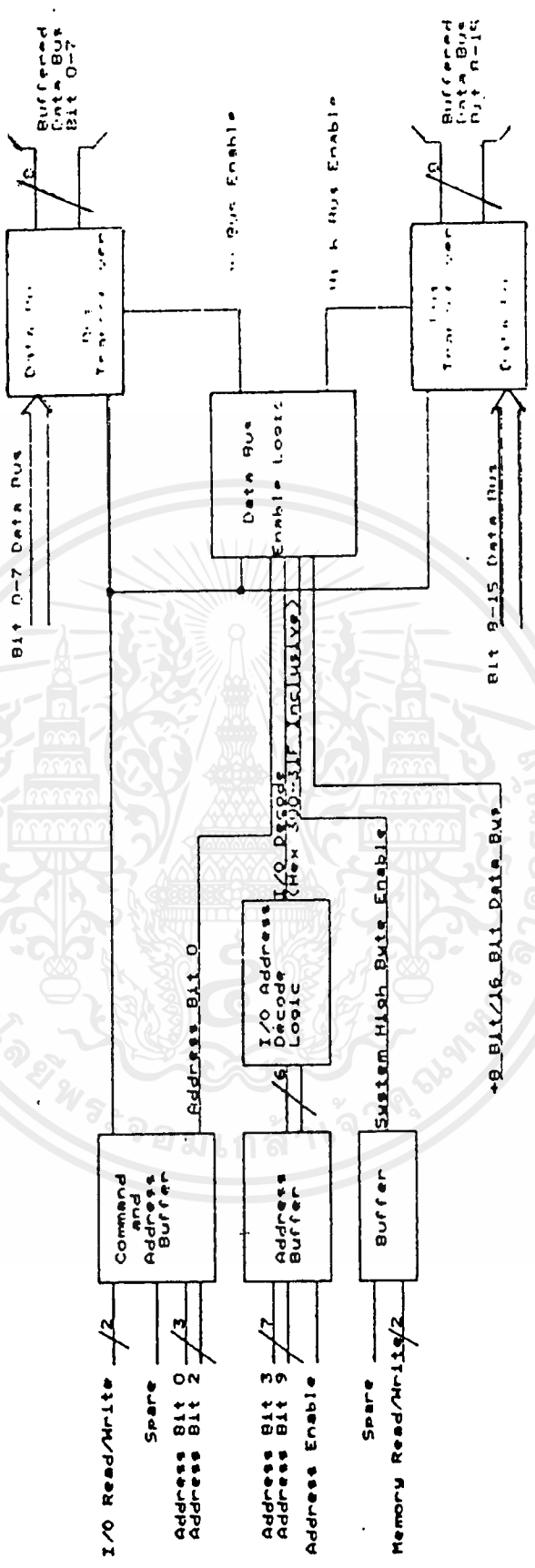
สัญญาณ -SBHE (เฉพาะรุ่น เอที) เป็นขาสัญญาณที่ใช้แสดงว่ามีการรับส่งข้อมูลในบิตที่ D8-D15

สัญญาณ -I/O CS16(เฉพาะรุ่น เอที)เป็นขาสัญญาณที่ใช้บอกระบบว่าต้องการรับส่งข้อมูลกับอุปกรณ์อินพุต เอาต์พุต ครั้งละ 16 บิต ,1 WAIT STATE(ต้องให้สัญญาณขานี้เป็น"0") และมีกระแสซิ้งค์ 20 มิลลิแอมป์ ขับโดยไอซีประเภทคอลเลคเตอร์เปิดหรือบัฟเฟอร์แบบ 3 สภาวะ

รูป 3.13 เป็นบล็อคไดอะแกรมของการเชื่อมโยงอุปกรณ์อินพุต เอาต์พุตกับระบบ ขนาดข้อมูล 16 บิต โดยปกติแล้วอุปกรณ์อินพุต เอาต์พุตจะมีขนาด 8 บิต ดังนั้นถ้าต้องการขนาดข้อมูล 16 บิตต้องใช้อุปกรณ์ 2 ชุด คือ ชุดสำหรับ 8 บิตบน(D8-D15) และ 8 บิตล่าง(D0-D7) การดีโค้ดแอดเดรสจะต้องสร้างสัญญาณเลือกชิปไบท์ต่ำ (D0-D7) และไบท์สูง (D8-D15) จากการทดลองสามารถแบ่งการสร้างสัญญาณเลือกชิปได้อีก 3 วิธี

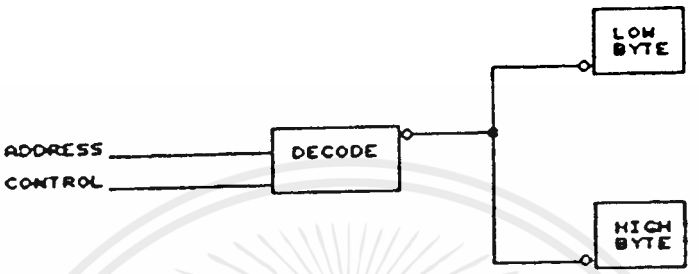
1.ใช้สัญญาณเลือกชิปสัญญาณเดียว (หมายเลขพอร์ตเดียว)เลือกชิปไบท์ต่ำ (D0-D7) และไบท์สูง (D8-D15)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



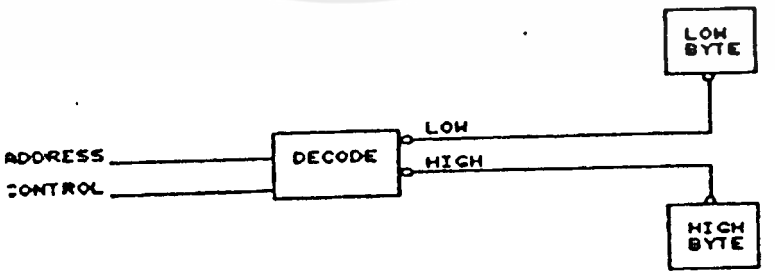
รูปที่ 3.14 บล็อคไดอะแกรมของการเชื่อมโยงอุปกรณ์อินพุต เอาต์พุตกับระบบขนาดข้อมูล 16 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.15 สัญญาณเลือกชิปสัญญาณเดียว

2. สัญญาณเลือกชิป 2 สัญญาณ (เบอร์พอร์ต 2 เบอร์เรียงกัน)



รูปที่ 3.16 สัญญาณเลือกชิป 2 สัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.ใช้สัญญาณ A0 และ -SBHE ใช้เลือกไบต์สูงและไบต์ต่ำ รูปที่ 3.16 พิจารณาสภาวะลอจิกของทั้ง 2 สัญญาณ สัญญาณทั้ง 2 นี้จะใช้แสดงการติดต่อกับอินพุต เอาต์พุตพอร์ตเป็นแบบไบต์หรือเวิร์ด สภาวะต่างๆ เป็นดังนี้

ตารางที่ 3.2 การทำงานของ A0 และ -SBHE

-SBHE	A0	การทำงาน
0	0	การกระทำแบบเวิร์ด (word access)
0	1	ไบต์คู่ (ไบต์ต่ำ)
1	0	ไบต์คี่ (ไบต์สูง)
1	1	ไม่ใช้งาน

รูปที่ 13.17 เป็นวงจรที่ออกแบบเพื่อทดลองการรับส่งข้อมูลแบบ 16 บิต ทางด้านอินพุตใช้ไอซี 74LS245 เป็นบัฟเฟอร์ข้อมูลทางอินพุตพอร์ต และเอาต์พุตพอร์ตใช้ไอซีแบบแลตช์74LS373 ในส่วนของการดีโคดใช้ไอซี 74LS688 สร้างสัญญาณ -IOSEL หรือ-GROUP SELโดยสัญญาณแอดเดรส A2-A9 จะต่อกับขา P0-P7 ตามลำดับ เพื่อเปรียบเทียบกับอินพุตด้าน Q ซึ่ง Q6-Q7 เป็นลอจิก "1" Q4-Q5 เป็นลอจิก "0" และ Q0-Q3 ต่อกับดิพสวิตช์สามารถปรับเปลี่ยนแอดเดรสได้ เมื่ออินพุตทางด้าน P=Q จะได้เอาต์พุต P=Q เป็น "0" ช่วงแอดเดรสดังนี้

A9 A8 A7 A6 A5 A4 A3 A2 A1 A0

1 1 0 0 X X X X X X พอร์ตช่วง 300H-33FH

เมื่อได้สัญญาณ P=Q เป็น "0" (-IOSEL หรือ -GROUP SEL) เมื่อนำมาออร์กับ -IORD เป็นสัญญาณสำหรับอีนาเบิ้ลอินพุต และนำมาออร์กับสัญญาณ -IOW จะได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณสำหรับอีนาเบิ้ลเอาต์พุต สัญญาณ -GROUP SEL จะต้องนำมาออร์กับ "0" เพื่อสร้างสัญญาณ -IO CS16 เป็นอินพุตของสล็อต ในช่วงที่ต้องการอ่านข้อมูลจากอินพุตพอร์ตครั้งละ 16 บิตเท่านั้นถ้าสัญญาณ -IO CS16 เป็น "0" ตลอดจะทำให้เครื่องแฮงค์

ไอซี 74LS139 U2A ทำหน้าที่อีนาเบิ้ลเอาต์พุตพอร์ต ที่ขา G จะได้รับสัญญาณ [-IORD] + [-IO SEL] ส่วน U2B ทำหน้าที่อีนาเบิ้ลเอาต์พุตพอร์ต ที่ขา G จะได้รับสัญญาณ [-IOW] + [-IO SEL] สามารถเขียนเป็นสภาวะทางลอจิกได้ดังนี้

ที่ U2A:

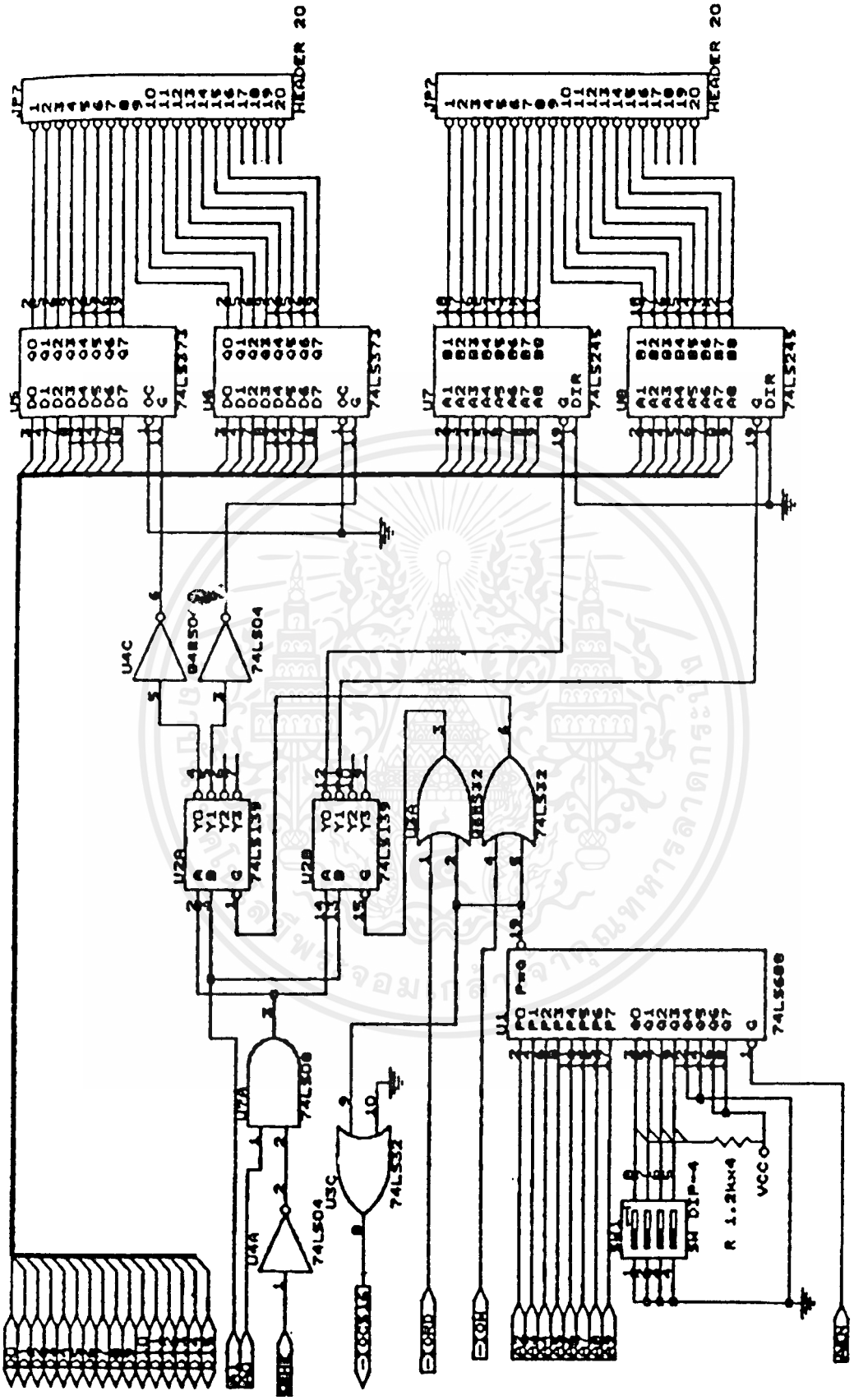
[-IORD + -IO SEL]	A1	-SBHE	A0	OUTPUT [Y]	
0	0	1	0	Y0	อ่านไบต์ต่ำ
0	0	0	1	Y1	อ่านไบต์สูง
0	0	0	0	Y0,Y1	อ่านแบบเวิร์ด
0	1	1	0	Y2	อ่านไบต์ต่ำ
0	1	0	1	Y3	อ่านไบต์สูง
0	1	0	0	Y2,Y3	อ่านแบบเวิร์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ U2B:

[-IOW + -IO SEL]	A1	-SBHE	A0	OUTPUT [Y]	
0	0	1	0	Y0	เขียนไบต์ต่ำ
0	0	0	1	Y1	เขียนไบต์สูง
0	0	0	0	Y0,Y1	เขียนแบบเวิร์ด
0	1	1	0	Y2	เขียนไบต์ต่ำ
0	1	0	1	Y3	เขียนไบต์ต่ำไบต์
0	1	0	0	Y2,Y3	เขียนแบบเวิร์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



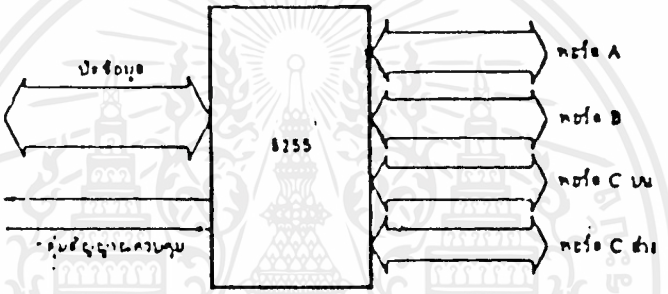
รูปที่ 3.17 16 บิต อินพุต-เอาท์พุตพอร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับ
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 8255 PPI

หลักการเบื้องต้นของ 8255

เป็นไอซี 40 ขา ตัวไอซี 8255 ได้รับการออกแบบมาเพื่อให้มีสัญญาณเชื่อมโยงกับ 8080 แต่สัญญาณนี้เหมาะที่จะใช้กับ IBM PC ได้ดีเช่นเดียวกัน 8255 เป็นไอซีที่ต่อเป็นพอร์ตให้กับไมโครโปรเซสเซอร์ได้ 3 พอร์ต โดยมีโครงสร้างพื้นฐานแสดงด้วยบล็อกไดอะแกรมได้ดังรูปที่ 3.18



รูปที่ 3.18 ไดอะแกรมของไอซี 8255

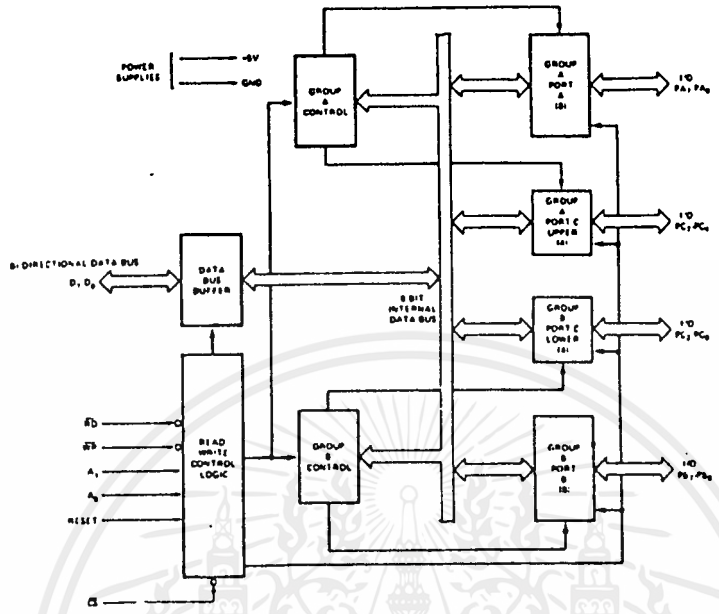
การเรียกพอร์ตของ 8255 จะเรียกพอร์ตว่า พอร์ต A,B และ C โดยพอร์ต C แยกเป็น 2 ส่วน คือ PC_0 - PC_3 เรียกว่าพอร์ต C ล่าง จำนวน 4 บิต และพอร์ต C บน PC_4 - PC_7 ที่พิเศษคือ พอร์ตทุกพอร์ตเป็นได้ทั้ง พอร์ตอินพุต และเอาต์พุต

3.5.1 ขาต่าง ๆ ของ 8255

เพื่อให้เข้าใจวิธีต่อใช้งานต่าง ๆ ระหว่าง IBM PC/AT กับ 8255 จึงจำเป็นต้องเข้าใจความหมายและตำแหน่งของขาต่าง ๆ เสียก่อน ขาทั้ง 40 ขา ของไอซีประกอบด้วย

D_0 - D_7 เป็นขาข้อมูลของอินพุต-เอาต์พุต ที่จะต้องผ่านเข้าออกจากส่วนนี้ D_0 - D_7 จึงต้องต่อเข้ากับระบบบัสของไมโครโปรเซสเซอร์ เพื่อให้ไมโครโปรเซสเซอร์ สามารถอ่านหรือเขียนข้อมูลออกทางพอร์ตผ่านทางบัสนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.19 ไดอะแกรมการทำงานของ 8255

CS (เลือกชิพ) ขานี้เป็นขาอินพุตที่จะรับสัญญาณจากภายนอก เพื่อเลือกชิพ 8255 นี้โดยเมื่อขานี้เป็นลอจิก "0" จะทำให้ตัว 8255 ต่อเข้ากับระบบของบัสของไมโครโปรเซสเซอร์เพื่อให้ไมโครโปรเซสเซอร์เขียนหรืออ่านข้อมูลออกจากพอร์ตได้

RD ขาสัญญาณการอ่านเป็นสัญญาณอินพุตที่จะส่งมาจาก CPU เมื่อสัญญาณที่ขานี้เป็น "0" และ CS เป็น "0" ตัว 8255 จะทำให้ตัวซีพียูอ่านข้อมูลจากบัส ในขณะเป็นอินพุต

WR ขาสัญญาณการเขียน จะแอคทีฟเมื่อสัญญาณ WR เป็น "0" และ CS เป็น "0" สัญญาณนี้จะมาจาก CPU เมื่อต้องการเขียนข้อมูลลงบนพอร์ตที่กำหนด

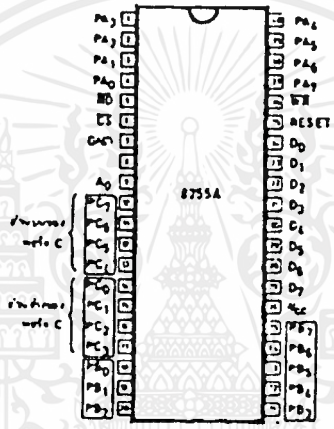
A_0 - A_1 ขาแอดเดรส ลอจิกของขาทั้งสองข้าง จะถอดรหัสเป็น 4 เพื่อกำหนดรีจิสเตอร์ภายในที่เชื่อมต่อกับพอร์ตอินพุต-เอาต์พุตของ 8255

RESET ขารีเซต เป็นสัญญาณที่ส่งจากภายนอกเข้ามาทำการรีเซต 8255 เพื่อเคลียร์สถานะต่าง ๆ ของ 8255 เมื่อ 8255 ได้รับการรีเซต มันจะกลับเข้าสู่โหมดอินพุตหรือทุกพอร์ตเป็นพอร์ตอินพุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PA_0 - PA_7 เป็นสายสัญญาณที่พอร์ตของ 8255 ที่ชื่อพอร์ต A การเลือกพอร์ตจะเลือกพอร์ตโดยขาแอดเดรส A_0 - A_1

PC_0 - PC_1 เป็นสายสัญญาณที่พอร์ต C ของ 8255 การกำหนดพอร์ตนี้จะได้รับการกำหนดโดยขาแอดเดรส A_0 - A_1 พอร์ต C นี้แบ่งเป็น 2 กลุ่ม คือกลุ่ม PC_0 - PC_3 และกลุ่ม PC_4 - PC_7



รูปที่ 3.20 แสดงตำแหน่งการวางขาของ 8255

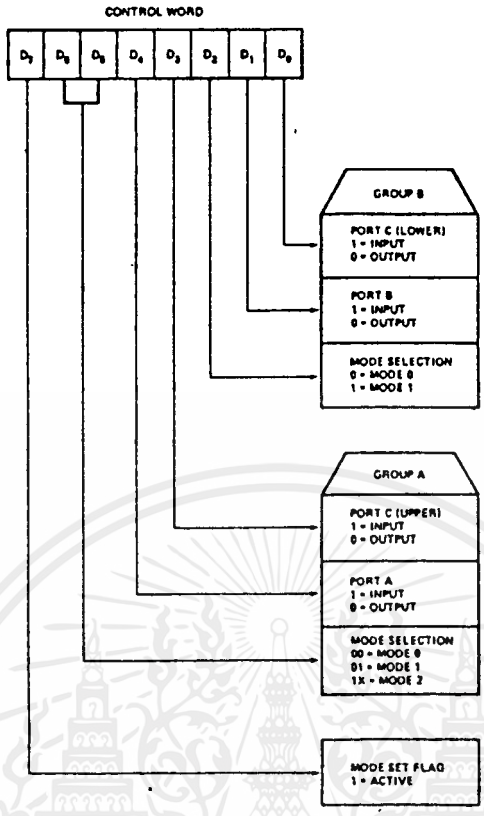
3.5.2 MODE การทำงานของ 8255

การเลือกการทำงานของ 8255 สามารถให้ทำงานได้หลายรูปแบบ โดยเรียกแต่ละแบบว่า โหมด (MODE) ต่าง ๆ มีอยู่ 3 โหมด คือ โหมด 0 โหมด 1 และโหมด 2

1. โหมด 0

โหมด 0 เป็นโหมดที่กำหนดให้ พอร์ตทุกพอร์ต บนตัว 8255 เป็นพอร์ตอินพุต-เอาต์พุต แบบพื้นฐาน รูปแบบความเป็นไปได้จึงมีทั้งสิ้น 16 รูปแบบ ตามลักษณะของพอร์ต A,B,C บน และ C ล่างลักษณะควบคุมแต่ละรหัสควบคุมแต่ละแบบ จะเป็นดังรูปที่ 3.22 และ 3.23

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

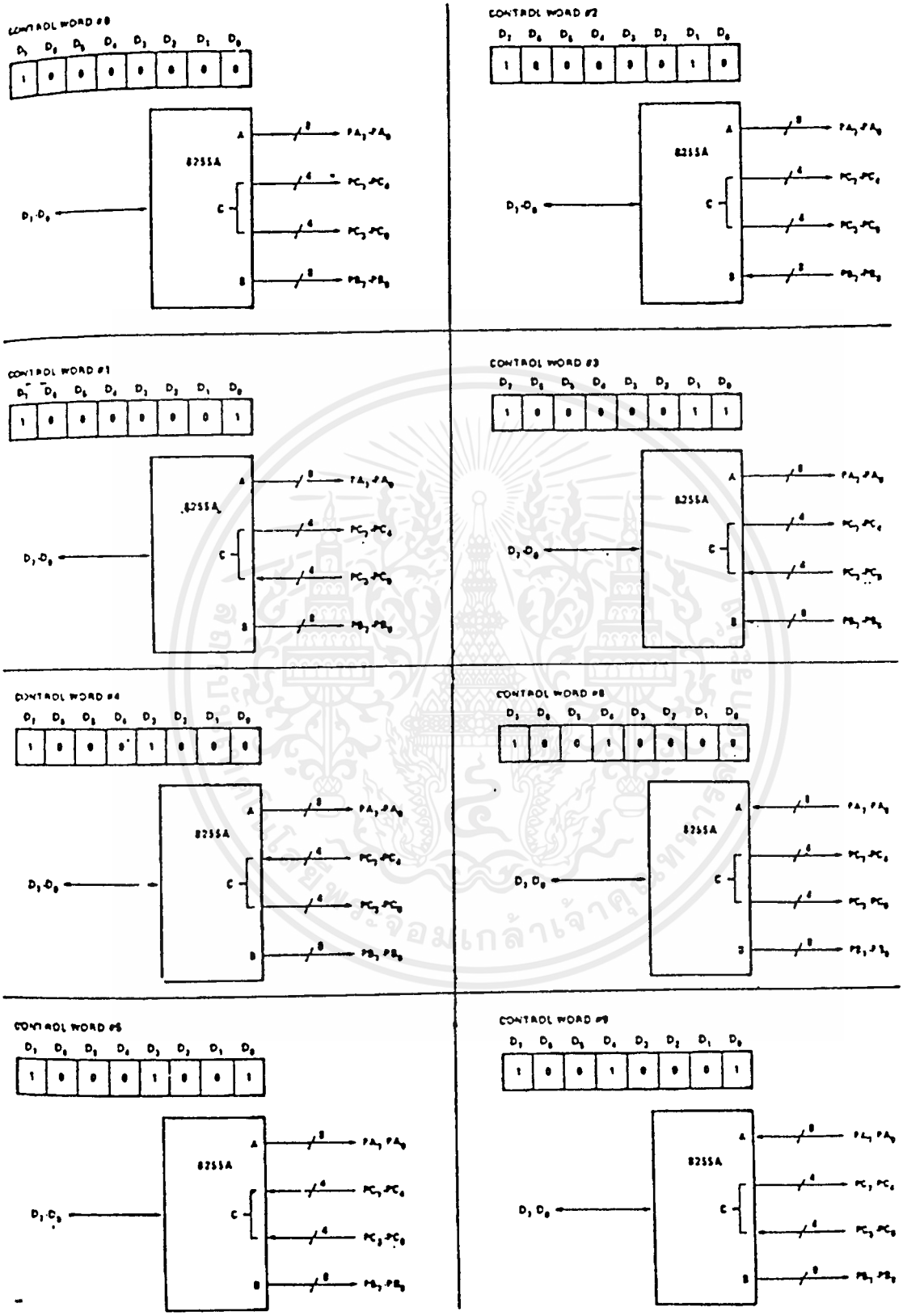


รูปที่ 3.21 ความหมายและบิตต่าง ๆ ของรหัสควบคุม

ตารางที่ 3.3 ตารางแสดงสัญญาณแต่ละ PORT

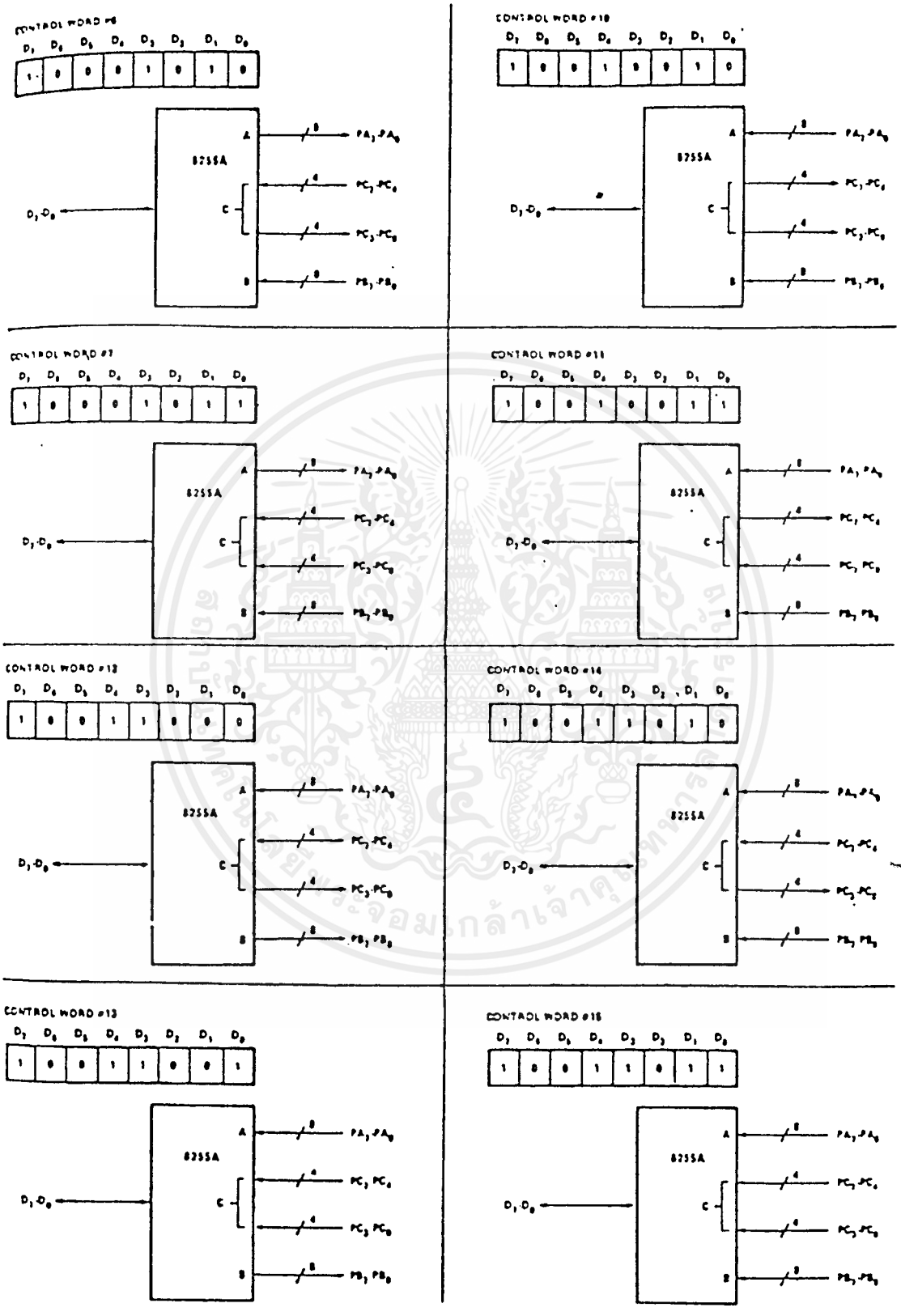
A		B		GROUP A			GROUP B	
D_4	D_3	D_1	D_0	PORT A	PORT C (UPPER)	#	PORT B	PORT C (LOWER)
0	0	0	0	OUTPUT	OUTPUT	0	OUTPUT	OUTPUT
0	0	0	1	OUTPUT	OUTPUT	1	OUTPUT	INPUT
0	0	1	0	OUTPUT	OUTPUT	2	INPUT	OUTPUT
0	0	1	1	OUTPUT	OUTPUT	3	INPUT	INPUT
0	1	0	0	OUTPUT	INPUT	4	OUTPUT	OUTPUT
0	1	0	1	OUTPUT	INPUT	5	OUTPUT	INPUT
0	1	1	0	OUTPUT	INPUT	6	INPUT	OUTPUT
0	1	1	1	OUTPUT	INPUT	7	INPUT	INPUT
1	0	0	0	INPUT	OUTPUT	8	OUTPUT	OUTPUT
1	0	0	1	INPUT	OUTPUT	9	OUTPUT	INPUT
1	0	1	0	INPUT	OUTPUT	10	INPUT	OUTPUT
1	0	1	1	INPUT	OUTPUT	11	INPUT	INPUT
1	1	0	0	INPUT	INPUT	12	OUTPUT	OUTPUT
1	1	0	1	INPUT	INPUT	13	OUTPUT	INPUT
1	1	1	0	INPUT	INPUT	14	INPUT	OUTPUT
1	1	1	1	INPUT	INPUT	15	INPUT	INPUT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.22 แสดงลักษณะการใช้งานของ MODE 0

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ ให้ใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

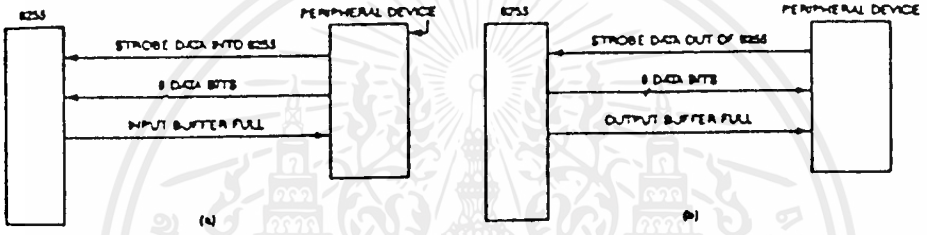


รูปที่ 3.23 แสดงลักษณะการใช้งานของ MODE 0 (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ ใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. โหมด 1

การทำงานของ 8255 ในโหมดที่ทำให้อินพุต-เอาต์พุต มีการตรวจสอบสัญญาณ (HANDSHAKING) โดยใช้อินพุต-เอาต์พุตของพอร์ต A และ B เป็นหลัก และใช้พอร์ต C บน เป็นสัญญาณ handshake ของพอร์ต A ส่วนพอร์ต C ล่าง เป็นสัญญาณ handshake ของพอร์ต B



รูปที่ 3.24 โครงสร้างของการจัดสัญญาณ HANDSHAKE

ตารางที่ 3.4 หน้าที่ต่าง ๆ ของพอร์ต C เป็นสัญญาณ handshake เมื่อ 8255 ทำงาน Mode 1

PC	IN	OUT
PC0	$INTR_B$	$INTR_B$
PC1	IBF_B	$\overline{OBF}_B$
PC2	$\overline{STB}_B$	$\overline{ACK}_B$
PC3	$INTR_A$	$INTR_A$
PC4	$\overline{STB}_A$	I/O
PC5	IBF_A	I/O
PC6	I/O	$\overline{ACK}_A$
PC7	I/O	$\overline{OBF}_A$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แนวความคิดของการใช้อินพุต-เอาต์พุตแบบ handshake ก็เพื่อให้มีการซิงโครไนซ์ ระหว่างอุปกรณ์ภายนอกที่ทำงานได้ช้ากับการทำงานของคอมพิวเตอร์ที่ทำงานได้เร็ว เช่น เครื่องพิมพ์ ทำงานได้ช้าเมื่อคอมพิวเตอร์ส่งตัวอักษรตัวแรกมาพิมพ์ เครื่องพิมพ์รับตัวอักษร และกำลังจะพิมพ์คอมพิวเตอร์ก็ส่งตัวที่ 2 และตัวที่ 3 ตามมา ทำให้การประมวลผลของอุปกรณ์เครื่องพิมพ์ทำงานไม่ทัน อาจจะทำให้ข้อมูลเสียหาย ดังนั้น เครื่องพิมพ์จึงส่งสัญญาณบอกคอมพิวเตอร์ว่า อย่าเพิ่งส่งมา เพราะยังไม่พร้อมรับ

ลักษณะของการรับส่งข้อมูล อินพุต-เอาต์พุต แบบมี handshake แสดงได้ดังรูป 3.24 และ PA_0 - PA_7 เป็นอินพุตโดยมีพอร์ต C เป็นสัญญาณ handshake ดังแผนผังในรูปที่ 3.40

เมื่อโปรแกรม 8255 เป็นโหมด 1 แล้วตัว 8255 จะให้พอร์ต C เป็นสัญญาณควบคุมโดยแต่ละบิตของพอร์ต เป็นไปตามที่กำหนดไว้แล้วดังตารางที่ 3.9

โดยปกติ 8255 ยังให้สัญญาณอินเตอร์รัพต์แบบอก ซีพียูด้วยสัญญาณอินเตอร์รัพต์ของ 8255 จะเกิดขึ้นที่ขา PC_0 และ PC_3 โดยที่บัฟเฟอร์พร้อมแล้ว และต้องการให้ซีพียูส่งอินพุต หรือเอาต์พุตมาที่บัฟเฟอร์ สัญญาณอินเตอร์รัพต์ที่เกิดขึ้น สังเกตุว่าสัญญาณอินเตอร์รัพต์เป็นสัญญาณแอคทีฟลอจิก "1" ซึ่งตรงกับของ 8080 เมื่อใช้กับ IBM สัญญาณ INT ของ IBM จะรับด้วยลอจิก "0"

โครงสร้างการ handshake ของ 8255 แสดงด้วยสัญญาณทางไฟฟ้าได้ดังรูปที่ 3.26 สังเกตุว่าการทำงานของ 8255 เกี่ยวกับสัญญาณ RD และ WR ซึ่งจะทำให้สัญญาณควบคุมเปลี่ยนแปลงไป

การตรวจสอบสัญญาณซึ่งกันและกันนี้ เป็นวิธีการรับส่งที่มีประสิทธิภาพ เช่น ในกรณีอินพุต เมื่ออุปกรณ์ภายนอกต้องการส่งข้อมูลให้ซีพียูก็จะส่งข้อมูลแบบขนานเข้ามาพร้อมทั้งสโตรบ (STB) บอก 8255 ตัว 8255 จะนำข้อมูลนั้นไปเก็บไว้ในรีจิสเตอร์ภายในก่อน แล้วส่งสัญญาณตอบบอกว่าบัฟเฟอร์ยังเต็มอยู่นะ (IBF) อย่าเพิ่งส่งมาอีก ครั้นเมื่อซีพียูอ่านข้อความจากรีจิสเตอร์ไปแล้ว ส่วนของสัญญาณบัฟเฟอร์อินพุต (IBF) ก็จะบอกว่าว่างแล้วส่งมาได้อุปกรณ์ภายนอกก็จะส่งมาให้อีก

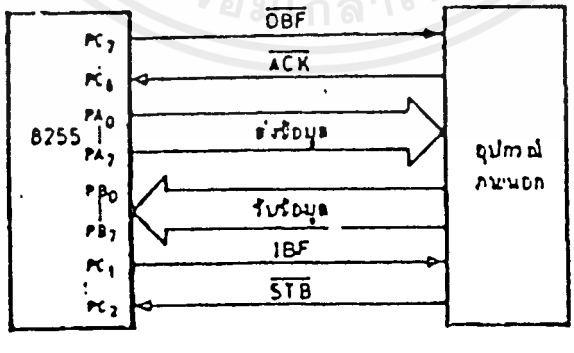
ทำนองเดียวกัน สำหรับพอร์ตเอาต์พุต เมื่อซีพียูส่งข้อมูลออกมาทางพอร์ตเอาต์พุตให้กับ 8255 ตัว 8255 ก็จะรับไว้ในรีจิสเตอร์ภายใน พร้อมทั้งส่งสัญญาณออกไปบอกอุปกรณ์ภายนอกว่า เอาต์พุตบัฟเฟอร์ของฉันมีข้อมูลอยู่นะ (OBF) มาอ่านเอาไปซิ อุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภายนอก เมื่อทราบ และพร้อมจะอ่านก็จะส่งสัญญาณตอบรับ (ACK) พร้อมกับอ่านข้อมูลไป โดยสัญญาณ ACK จะมีความหมายว่า ฉันอ่านข้อมูลของเธอไปแล้วนะ ตัว 8255 ก็จะตอบกลับว่า บัฟเฟอร์ฉันว่างแล้ว เธอรอก่อนนะ จะมีข้อมูลใหม่ส่งมาให้อีก

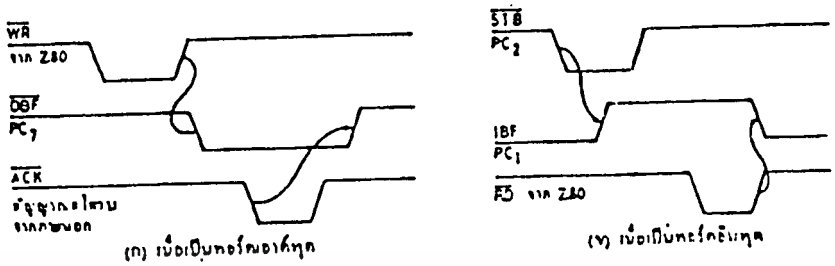
ในกรณีที่จะโปรแกรมโหมด 1 นี้นั้น เราจะให้รหัสควบคุมเป็น 101 (I/O) 01 (I/O)0 ในส่วน (I/O) หมายถึงถ้าเป็นอินพุตก็เป็น "1" เอาต์พุตก็เป็น "0" โดย I/O ตัวแรกเป็นของพอร์ต A ตัวที่ 2 พอร์ต B เช่น ถ้าต้องการให้พอร์ต A เป็นเอาต์พุต และพอร์ต B เป็นอินพุต เราจะใช้รหัสควบคุมเป็น 10100110 หรือ A6H

หากจะพิจารณาการทำงานของซีพียูจะเห็นว่า ทำอย่างไรจึงเขียน หรืออ่านพอร์ตได้ถูกต้อง วิธีง่ายวิธีหนึ่งคือซีพียูจะคอยตรวจสอบสัญญาณของ 8255 เช่น กรณีเอาต์พุต ซีพียู จะคอยอ่านพอร์ต C แล้วตรวจสอบบิต 7 (OBF) หลังจากที่ส่งข้อมูลไปแล้ว ถ้าบิต 7 ยังเป็น "0" แสดงว่ายังไม่ได้รับการสมบูรณ์ แต่ถ้าเป็น "1" แล้วแสดงว่าอุปกรณ์ภายนอกรับข้อมูลไปแล้ว สำหรับกรณีอินพุตคอยตรวจสอบจากสัญญาณ IBF ได้เช่นกันว่า มีข้อมูลใหม่เข้ามาหรือยังคือตรวจสอบบิต PC_1 ของพอร์ต C



รูปที่ 3.25 การต่อ 8255 ในโหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.26 แผนผังเวลาการรับ/ส่งข้อมูลแบบ handshake

สรุปโหมด 1

1 แบ่งออกเป็น 2 กลุ่มคือ กลุ่ม A และกลุ่ม B

2 แต่ละกลุ่มประกอบด้วย DATA PORT 8 BIT และ 4 BIT HAND SHAKING

3 สำหรับ 8 BIT DATA จะเป็นได้ทั้ง INPUT/OUTPUT

4 ใช้ 4 bit เป็น CONTROL, SATATUS ของ DATA 8 BIT

สัญญาณควบคุม ในการีอินพุต (INPUT SIGNAL CONTROL DEFINATION)

1 STB (STORBE INPUT) เป็นขาสัญญาณอินพุต ซึ่งเมื่อเป็น "LOW" ก็ได้แสดงว่าข้อมูลได้ถูกส่งเข้าไป LATCH เอาไว้

2 IBF (INPUT BUFFER FULL F/F) เป็นขาเอาต์พุต เมื่อเป็น "HIGH" ก็แสดงว่าข้อมูลได้ถูกส่งเข้าไปใน INPUT แล้วโดยที่ IBF จะถูกเซ็ทที่ขอบขาลงของสัญญาณ STB (FALLING EDGE OF RD INPUT) และถูกรีเซ็ทโดยสัญญาณขอบขาขึ้นของสัญญาณ RD (RISING EDGE OF RE INPUT)

3 INTR (INTERRUPT REQUEST) เป็นสัญญาณเอาต์พุตซึ่งใช้ INTERRUPT CPU เมื่ออุปกรณ์ I/O ต้องการขอทำการ INTERRUPT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กับ CPU ขา INTR จะถูกเซ็ทโดยสัญญาณขาขึ้นของ STB ถ้า IBF เป็น "1" และ INTR จะถูกรีเซ็ทโดยสัญญาณขาลงของสัญญาณ RD ซึ่งแสดงว่า CPU ยอมรับการ INTERRUPT แล้ว

4 INTE A ถูกควบคุมด้วย BIT SET/RESET ของ PORT C4

5 INTE B ถูกควบคุมด้วย BIT SET/RESET ของ PORT C2

สัญญาณควบคุมในกรณีเอาต์พุต (OUTPUT CONTROL SIGNAL DEFINITION)

1 OBF (OUTPUT BUFFER FULL F/F) เป็นขาสัญญาณเอาต์พุต จะเป็น "LOW" เพื่อแสดงว่าให้มีการ WRITE ข้อมูลออกมาที่พอร์ตที่กำหนดของ CPU มายัง 8255 ได้ OBF F/F จะเซ็ทเป็น "LOW" โดย RISING EDGE ของ WR INPUT SIGNAL

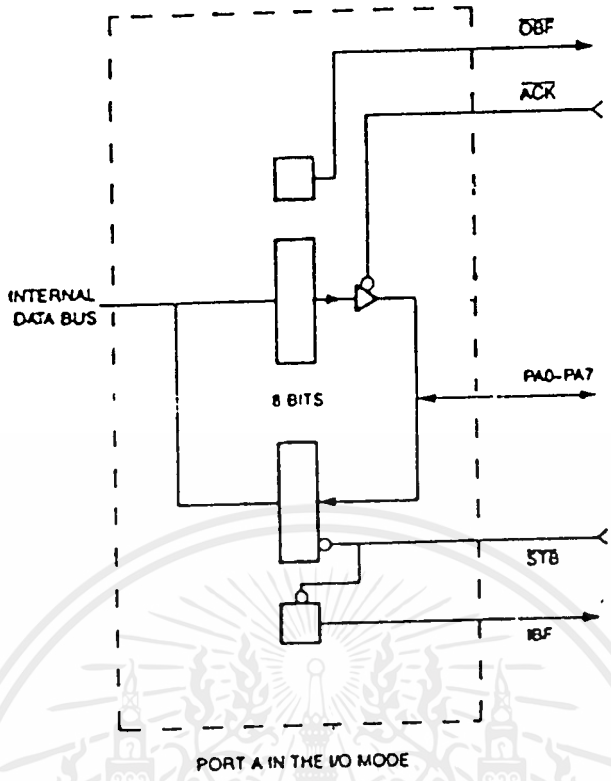
2 ACK (ACKNOLEDGE INPUT) เป็นสัญญาณอินพุท จะเป็น "LOW" เพื่อแสดงว่าข้อมูลที่พอร์ต A หรือพอร์ต B ของ 8255 ได้ถูกรับไปแล้วซึ่งเป็นสัญญาณตอบรับจากอุปกรณ์ภายนอกที่พอร์ต

3 INTR (INTERRUPT REQUEST) เป็นสัญญาณเอาต์พุตซึ่งจะเป็น "HIGH" เพื่อ INTERRUPT CPU เมื่ออุปกรณ์เอาต์พุตได้รับข้อมูลจาก CPU แล้วขา INTR จะเซ็ทที่ RISING EDGE ของ ACK ถ้า OBF เป็น "HIGH" INTE จะรีเซ็ทที่ FALLING EDGE ของสัญญาณ WR

3. โหมด 2

8255 ยังมีโหมดการทำงานอีกโหมดหนึ่งคือ โหมด 2 ซึ่งทำได้เฉพาะพอร์ต A ในโหมดนี้ 8255 จะใช้พอร์ต A ทำหน้าที่เป็นพอร์ตแบบสองทิศทางคือสามารถเป็นได้ทั้งอินพุตและเอาต์พุตโดยโครงสร้างของพอร์ต A ทั้งอินพุทมี handshake ทั้งคู่ ส่วนพอร์ต C ที่ใช้จะทำหน้าที่เป็นสัญญาณตรวจจสอบแต่ละขาดังตารางที่ 3.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.27 โครงสร้างของพอร์ต A ที่ทำงานแบบพอร์ต 2 ทิศทาง

ตารางที่ 3.5 หน้าที่ของพอร์ต C ในโหมด 2

PORT C LINE	DEFINITION
PC0	I/O
PC1	I/O
PC2	I/O
PC3	$INTR_A$
PC4	$\overline{STB}_A$
PC5	IBF_A
PC6	$\overline{ACK}_A$
PC7	$\overline{OBF}_A$

สังเกตว่า เมื่อโปรแกรมพอร์ต A เป็นโหมด 2 แล้ว พอร์ต B จะต้องโปรแกรมเป็นโหมด 0 หรือโหมด 1 ก็ได้ ซึ่งการทำงานได้แยกอิสระอีก ในการใช้งานพอร์ตแบบ 2 ทิศทางนี้มีใช้ได้กับงานบางประเภท เช่น ใช้ในการรับส่งข้อมูลของพอร์ตมาตรฐานบางประเภท เช่น IEEE 488 หรือใช้เชื่อมโยงระหว่างคอมพิวเตอร์ในการรับส่งข้อมูลสลับกันไปกลับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปโหมด 2 BASIC FUNCTION DEFFINITIONS

1 ใช้ GROUP A เพียง GROUP เดียว

2 มี 8 BIT PORT ใช้งานเพียงพอร์ตเดียวเป็น BI-DIRECTION BUS PORT (PORT A) และมีพอร์ตสำหรับควบคุมอีก 5 BIT คือ พอร์ต C

3 ทั้งอินพุตและเอาต์พุตเป็นแบบ LATCH-SIGANAL ได้

4 มีบิทสำหรับควบคุม 5 BIT (พอร์ต C ใช้สำหรับ CONTROL และ STATUS สำหรับ 8 BIT BI-DIRECTION BUS PORT (PORT A)

สัญญาณควบคุมในกรณี MODE 2 BI-DIRECTION BUS I/O CONTROL SIGNAL DEFINTION

INTR (INTERRUPT REQUEST) หรือขาสัญญาณ OUTPUT "HIGH" เมื่อต้องการ INTERRUPT CPU ทั้งการทำงานเป็นอินพุตและเอาต์พุต

กรณีที่ทำงานเป็น OUTPUT

- OBF (OUTPUT BUFFER FULL) OUTPUT SIGNAL "LOW" เมื่อขานี้เป็น LOW แสดงว่าขณะนี้ 8255 ได้รับข้อมูลมาจากพอร์ต A แล้ว

- ACK (ACKNOWLEDGE) INPUT SIGNAL จะเป็น "LOW" เพื่อบอกให้รู้ว่า ขณะนี้ได้มีข้อมูลมาอยู่ที่พอร์ต A แล้ว พร้อมที่จะส่งข้อมูลออกไปทางพอร์ต A และจะเซ็ทเอาต์พุตเป็นสภาวะ "HIGH" IMPLEDANCE" เมื่อเป็น "HIGH"

- INTE1 (THE INTERRUPT F/F ASSOCIATED WITH OBF) เป็นฟลิบฟลอบภายใน ซึ่งจะสร้างสัญญาณ INTR A โดยทำงานร่วมกับขา OBF ซึ่งสามารถเซ็ทหรือรีเซ็ทได้ที่ PC6

ในกรณี INPUT OPERATION

- STB (STOBE INPUT) เป็นสัญญาณอินพุทเป็น LOW เพื่อมีการ LATCH ข้อมูลเอาไว้ใน INPUT LATCH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- IBF (INPUT BUFFER FULL F/F) เป็นขาสัญญาณเอาต์พุท จะเป็น HIGH เพื่อแสดงให้รู้ว่าได้กระทำการโหลดข้อมูลไว้ใน INPUT LATCH เรียบร้อยแล้ว

- INTE 2 (THE INTE F/F ASSOCIATED WITH IBF) เป็นฟลิบ-ฟลอบภายในซึ่งจะสร้างสัญญาณ INTE A โดยทำงานร่วมกับขา IBF โดยสามารถเซ็ทหรือรีเซ็ทที่ขา PC4

- INTR (INTERRUPT REQUEST) เป็นขาสัญญาณเอาต์พุทจะเป็น HIGH เมื่อต้องการทำการอินเตอร์รัพต์กับ CPU

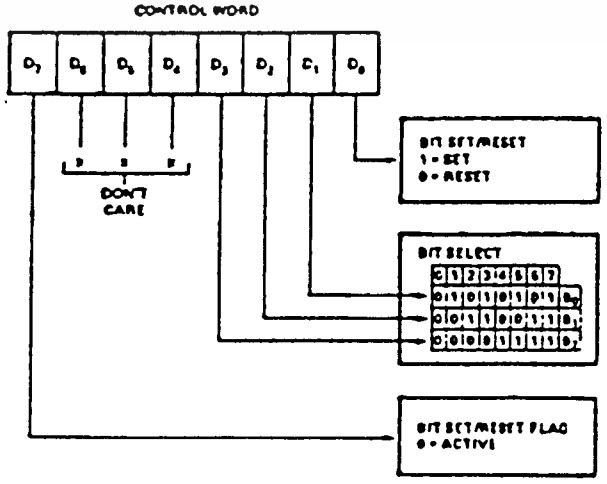
3.5.3 การเซ็ท/รีเซ็ทบิต

บิตทั้งสามของพอร์ต C สามารถที่จะเซ็ทหรือรีเซ็ทได้โดย CONTROL WORD ที่ส่งมาจาก CPU เพื่อทำการควบคุมให้พอร์ต C ทำตาม CONTROL WORD ที่ส่งมาจากพอร์ต C จะใช้เป็นฟังก์ชัน การควบคุมการอินเตอร์รัพต์

ในการทำงานของโหมด 1 หรือโหมด 2 สัญญาณควบคุมสามารถส่งผ่าน INTERRUPT REQUEST เข้าไปยัง CPU โดยสัญญาณนี้สร้างขึ้นโดยพอร์ต C เพื่อจะไปทำการ INHIBITED หรือ ENABLED โดยการเซ็ทหรือการรีเซ็ทขา INTE F/F ของ CPU โดยสัญญาณ BIT SET / RESET ของพอร์ต C

(BIT SET) ตอบสนองการอินเตอร์รัพต์ (INTERRUPT ENABLE)

(BIT RESET) ไม่ตอบสนองการอินเตอร์รัพต์ (INTERRUPT DISABLE)



รูปที่ 3.28 แสดงการเซ็ท - รีเซ็ทบิต

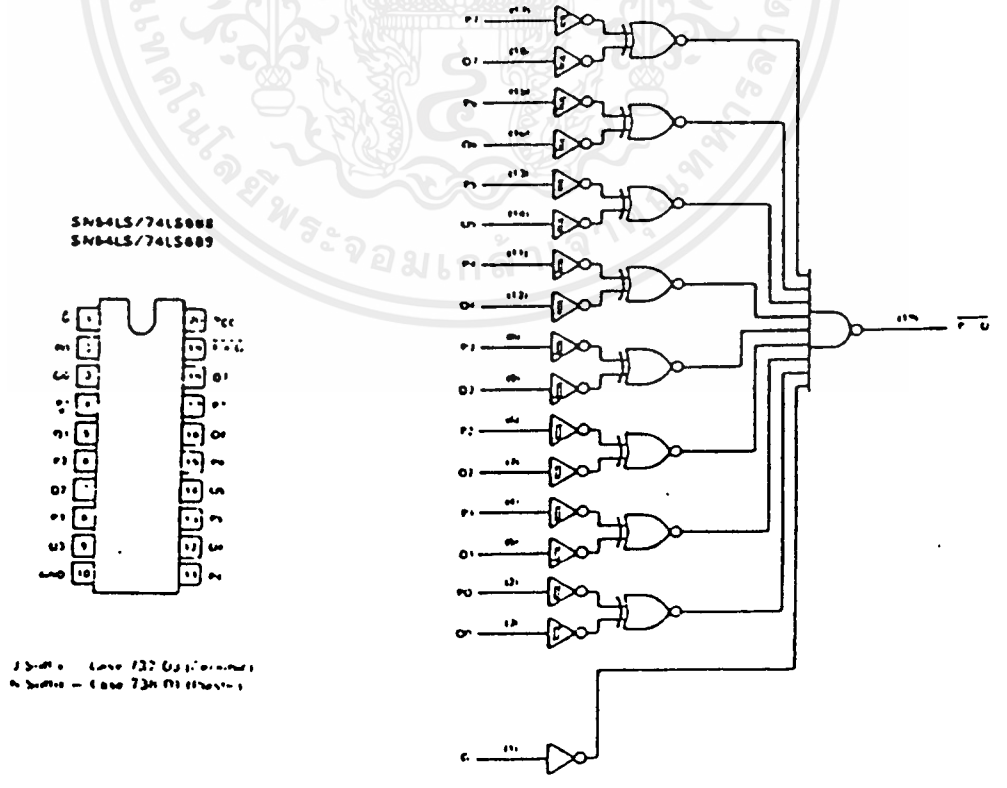
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6 8255 INTERFACE CARD

3.6.1 อุปกรณ์ที่ช่วยในการ ดีโค้ดสัญญาณต่าง ๆ บน IBM

เมื่อทราบถึงสัญญาณต่าง ๆ ที่ 8255 ต้องการ และเทคนิคการดีโค้ดแบบต่าง ๆ ที่สามารถนำไปใช้ได้แล้ว การที่ต่อ 8255 เข้ากับเครื่องคอมพิวเตอร์ IBM PC/AT นั้น ก็จะสะดวกขึ้น เพียงแต่จุดชุดวงจรเลือกแอดเดรสให้แก่ 8255 เท่านั้น และเพื่อให้การใช้งานได้สะดวกจึงใช้การดีโค้ดแบบเปลี่ยนแอดเดรสได้ด้วย ในวงจรนี้จึงเลือกอุปกรณ์ที่ช่วยในการดีโค้ด คือ 74LS688

74LS688 ไอซีตัวนี้ทำหน้าที่ในการเปรียบเทียบสัญญาณ 2 สัญญาณ ว่าเหมือนกันหรือไม่ ("0" หรือ "1") โดยสามารถเปรียบเทียบได้พร้อมกันทั้ง 8 คู่ โดยอินพุทแต่ละคู่จะอยู่ที่ขา P และ G และมีสัญญาณที่ขา input แต่ละคู่ตรงกัน จะทำให้มีสัญญาณออกมาที่ output (ขา 19) ของ 74LS688 เป็นลอจิก "0" นั่นก็คือ 74LS688 จะทำงานเต่ถ้า



รูปที่ 3.29 การจัดขาและตาราง Output ของ IC 74LS688

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อินพุทคู่ใดคู่หนึ่งมีสัญญาณเดกต่างกัน output จะออกเป็น "1" ทันทีนั่นก็คือ 688 จะไม่แอคตีฟ และ IC 74LS688 นั่นยังมีสัญญาณ Enable อีก 1 ขา คือ ขา G นี้มีลอจิกเป็น "0" จึงมีการเปลี่ยนแปลงสัญญาณต่าง ๆ ได้

3.7 การทดสอบและวงจรการทำงาน

เราศึกษาถึงโครงสร้างและการทำงานของ 8255 ไปแล้วในต่อไปนี้เราจะกล่าวถึงการนำ 8255 อินเตอร์เฟสกับ IBM/AT ซึ่งเป็นวงจรที่ใช่งานจริงรวมทึงรายละเอียดของอุปกรณ์ที่ใช่และแผ่นวงจรพิมพ์ในการอธิบายเป็นส่วน ๆ คือวงจรอินเตอร์เฟส 8255 นั้น เราสามารถแบ่งออกได้เป็น 3 ส่วน คือ

1. วงจรดีโค้ด
2. วงจรบัฟเฟอร์
3. ส่วนของซิพซัพพอร์ท

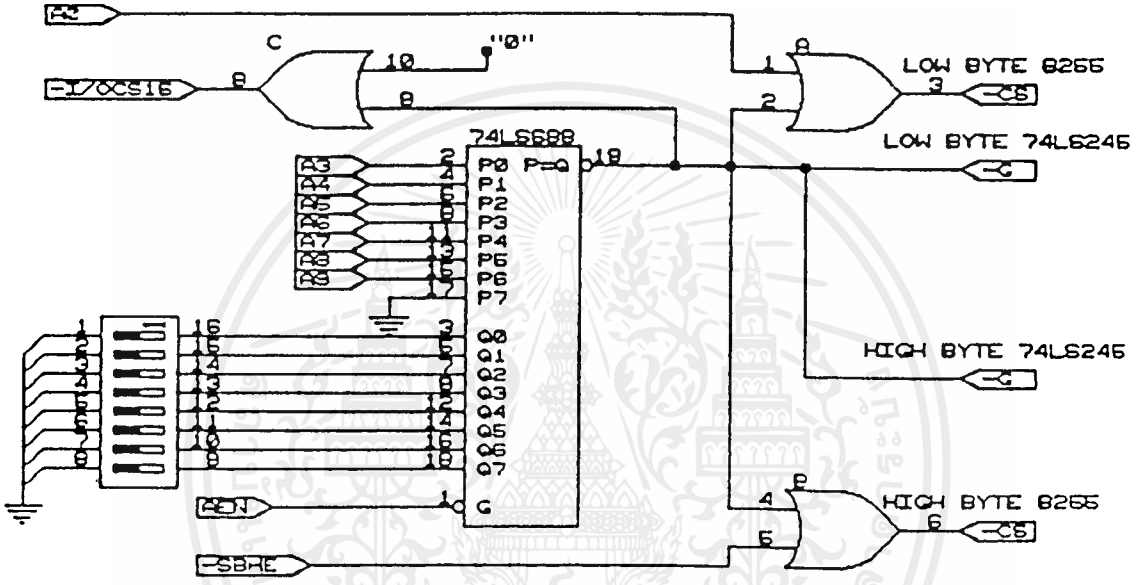
3.7.1 วงจรดีโค้ด

IC ที่เรานำมาทำการดีโค้ด เป็น IC 74LS688 การดีโค้ดสัญญาณต่างๆ จาก Slot ของ IBM/AT มาใช่เลือกแอดเดรสให้แก่ 5288 สัญญารที่จะนำมาทำการดีโค้ดก็มี A0-A9 แต่ 8255 ต้องการสัญญาณจาก A0 และ A1 ไปควบคุมรีจิสเตอร์ภายในตัวมันดังนั้นสัญญาณแอดเดรสที่เรานำมาใจ้จึงมี A3-A9 ส่วน A1 และ A2 เราจะนำไปต่อเข้ากับ 8255 โดยตรงโดยเราได้นำสัญญาณ A1 จาก SLOT ต่อเข้ากับ A0 จากSLOT ต่อเข้าโดยตรงกับ 8255 ก็เพราะว่าเราต้องนำสัญญาณ A0 ไปใช้ได้ดีโค้ดร่วมกับสัญญาณ SBHE และ สัญญาณจาก SLOT ที่เราได้นำมาทำการดีโค้ดยังมีสัญญาณ AEN และ IOCS16 อีกดังได้แสดงตามวงจรในหน้าถัดไป

ก่อนที่จะอธิบายถึงการทำงานของวงจร ดีโค้ดเตอร์ เราจะต้องรู้ก่อนว่าการดีโค้ดของ IBM PC/AT แยกออกเป็น 2 แบบ คือ การดีโค้ดหน่วยความจำ และ การดีโค้ดพอร์ตการดีโค้ดหน่วยความจำเราจะไม่ขอกล่าวถึง เพราะอินเตอร์เฟสการ์ดของเราต้อง

เอกสารนี้มีลิขสิทธิ์ ไม่อนุญาตให้นำไปใช้ในเชิงพาณิชย์ ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การติดต่อกับ I/O พอร์ตเท่านั้น และการดีโค้ดพอร์ตนั้นจะใช้ในการเลือกแอดเดรสที่จะใช้การดีโค้ดนั้นซึ่งในที่นี้จะใช้แอดเดรส 10 ตำแหน่งคือ A0-A9 เข้ามาทำการดีโค้ดส่วนตำแหน่ง A10-A15 จะไม่ได้นำมาใช้งานแต่ค่าแอดเดรสเหล่านี้ ยังคงเปลี่ยนแปลงตามค่าแอดเดรสที่กำหนดไว้ใน คำสั่ง OUT หรือ IN เพียงแต่ไม่ได้ถูกนำมาใช้ดีโค้ดร่วมกับ A0-A9 เท่านั้น



รูปที่ 3.30 การทำงานของวงจรดีโค้ดเดอร์

จากรูปที่ 3.30 เป็นการแสดงวงจรดีโค้ดของการอินเตอร์เฟส 8255 จะเห็นได้ว่านอกจากได้เอาสัญญาณแอดเดรส A3-A9, AEN แล้วยังเอาสัญญาณ IOCS16,A0 และ SBHE เข้ามาดีโค้ดด้วย คือ สัญญาณ A0 จะใช้ร่วมกับ SBHE ซึ่ง A0 จะเป็นตัวกำหนดการรับและส่งข้อมูล 8 บิตล่าง คือ (D0-D7) ส่วนสัญญาณ SEHE จะเป็นตัวกำหนดการรับและส่งข้อมูล 8 บิต D8-D15 ส่วนสัญญาณ IOCS16 เป็นตัวบอกให้มีการรับและส่งข้อมูลแบบ 16 บิต

การทำงานของ IC74LS688 นั้นจะเป็นลักษณะของวงจรเปรียบเทียบซึ่งรายละเอียดข้าต่าง ๆ ได้อธิบายไว้ก่อนหน้านี้แล้วการทำงานของ IC ตัวนี้จะเปรียบเทียบสัญญาณ อินพุตซึ่งแยกเป็น 2 ส่วน คือ P0-P7 และ Q0-Q7 สัญญาณเอาต์พุตจะเป็น "0" อินพุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

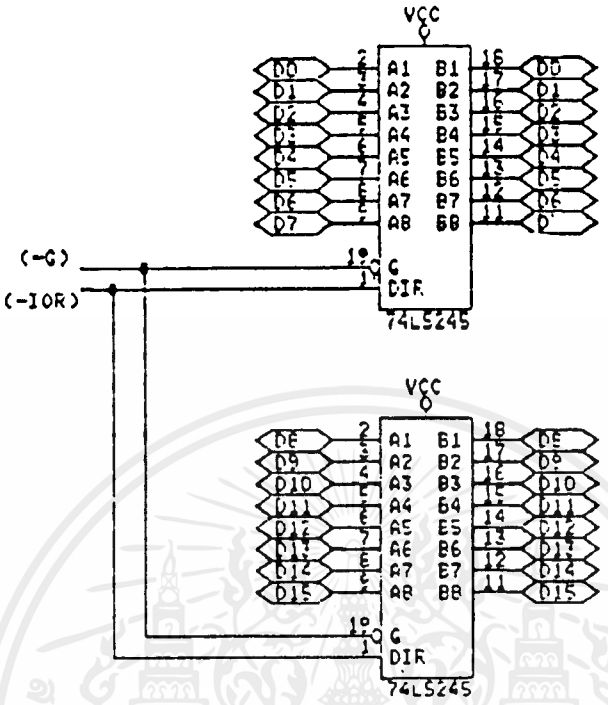
Q เราจะต่อเข้ากับ DIP SW และเราจะเช็คค่าที่ DIF SW ไว้ที่ พอร์ต SOG เมื่อมีสัญญาณแอดเดรส 306H มาที่อินพุต P ของ IC 688 ก็จะมีสัญญาณ LOW ออกมาที่ขา 19 (CS) และสัญญาณนี้ก็จะไปเป็นสัญญาณ CS ให้กับวงจรบัฟเฟอร์

แอดเดรส	ตำแหน่ง PORT
301	PORT A HIGH BYTE
302	PORT B LOW BYTE
303	PORT B HIGH BYTE
304	PORT C LOW BYTE
305	PORT C HIGH BYTE
306	CONTROL
307	CONTROL

ตารางที่ 3.6 แสดงแอดเดรส ของ พอร์ต

3.7.2 วงจรบัฟเฟอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.31 วงจรบัฟเฟอร์

ในวงจรการอินเตอร์เฟส 8255 นี้เราจะใช้ IC 74LS245 เป็นบัฟฟอร์ 2 ทิศทางสัญญาณ DATA ของ D0-D7 และ D8-D15 ซึ่งต่อออกมาจาก SLOT ของ IBM ซึ่งการทำงานของ IC จะอธิบายดังต่อไปนี้

วงจรบัฟเฟอร์ จะทำหน้าที่เหมือนกับสวิทช์ ปิด-เปิด ให้สัญญาณต่อถึงกัน ดังแสดงในรูปที่ 3.31 จะเห็นได้ว่า IC 74LS245 จะทำงาน 2 ทิศทาง เป็นทั้งรับและส่งข้อมูลโดยมีสัญญาณควบคุม 2 สัญญาณอีนาเบิล (-G) และสัญญาณควบคุมทิศทาง (DIR)

จากวงจรที่แสดงในรูปบัสข้อมูล D0-D7 จาก SLOT ต่อเข้ากับ A1-A8 ของ 74LS245 ตัวที่เป็น LOW BYTE และขา B1-B8 ต่อเข้ากับขาสัญญาณ D0-D7 ของ8255 ตัวที่เป็น LOW BYTE และบัสข้อมูล D8-D15 จาก ม SLOT ต่อเข้ากับขา A1-A8 ของ74LS245 ตัวที่เป็น HIGH BYTE ขาสัญญาณอีนาเบิล (-G) ของ IC 74LS245 ตัวที่เป็น LOW BYTE และขาสัญญาณอีนาเบิล (-G) ขา 19 ของ 688 ส่วนขาสัญญาณ DIR ขา 1 ของ

เอกสารนี้ใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

74LS245 ตัวที่เป็น HIGH BYTE และ LOW BYTE จะต่อเข้าด้วยกัน และไปต่อเข้ากับขาสัญญาณ DIR(245) จาก SLOT การทำงานของวงจร บัฟเฟอร์ นี้จะเกิดขึ้นเมื่อขาสัญญาณอินาเบิลเอดทีฟ นั้นคือเมื่อมีสัญญาณแอดเดรส #306 วงจรดีโคดจะทำงานและจะมีสัญญาณมาชิพให้กับ 74LS245 ทำงานด้วย ส่วนทิศทางของข้อมูลจะส่งข้อมูลหรืออ่านข้อมูลนั้นขึ้นอยู่กับ สัญญาณ DIR ดังนี้คือ

1. สัญญาณ DIR เป็นลอจิก "1" ข้อมูล D0-D15 จาก SLOT จะไปปรากฏที่ขาสัญญาณ D0-D7 ของ 8255 ทั้ง 2 ตัว คือทั้ง LOW BYTE และ HIGH BYTE

2. สัญญาณ DIR เป็นลอจิก "0" ข้อมูลที่ขาสัญญาณ D0-D7 จาก 8255 ทั้ง 2 ตัวจะไปปรากฏที่บัสข้อมูล D0-D15 ของ SLOT

3.7.3 วงจรการต่อชิพซัพพอร์ต 8255

8255 เป็นชิพซัพพอร์ตที่เป็นทั้ง อินพุต-เอาต์พุต พอร์ต เพื่อเชื่อมอุปกรณ์อินพุต-เอาต์พุตเข้ากับระบบคอมพิวเตอร์แสดงการต่อวงจรดังรูปที่ 3.32

การทำงานของวงจรส่วนของวงจรดีโคด และส่วนของบัฟเฟอร์ได้อธิบายไปแล้วคราวนี้ก็จะกล่าวถึงการทำงานของวงจร อินเตอร์เฟสการ์ด 8255 ทั้งวงจร ในการที่จะต่อ 8255 เข้ากับเครื่อง IBM PC/AT นี้จะต้องนำเอาสัญญาณต่าง ๆ ที่จะมาควบคุม 8255 ให้ทำงานตามต้องการ สัญญาณที่ต้องใช้คือ

A1-A2 เป็นสัญญาณแอดเดรสที่นำมาจาก SLOT ของ IBM ไม่ต้องผ่านวงจรดีโคดเพราะจะนำมาดีโคดบน 8255 ซึ่งการเปลี่ยนแปลงของ A1 และ A2 จะไปควบคุมรีจิสเตอร์ภายใน 8255 ซึ่งใช้ในการเลือกโหมดการทำงานต่าง ๆ ของ 8255 ซึ่งในการจะควบคุมให้ 8255 ทำงานในโหมดไหนนั้นต้อง CONTROL WORD ไปที่รีจิสเตอร์ของ 8255 ก่อน

RESET 8255 ต้องการสัญญาณ RESET ที่มีลอจิกเป็น "1" ซึ่งสัญญาณ RESET ของ IBM ก็มีลอจิก "1" ตรงกับ 8255 จึงสามารถต่อกับ RESET ของ IBM เข้ากับขา RESET ของ 8255 ได้โดยตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RD สัญญาณ RD ของ 8255 ต่อเข้าโดยตรงกับสัญญาณ RD ของ IBMโดยสัญญาณนี้จะ ACTIVE เมื่อเครื่องคอมพิวเตอร์ IBM ต้องการอ่านข้อมูลจากอุปกรณ์ภายนอกผ่านทาง 8255 WR ซึ่งเข้ากับขา WR ของ 8255 โดยตรงจะ ACTIVE เมื่อขา WR มีพอจิก "0" คือ ต้องการเขียนข้อมูลผ่านบน 8255

D0-D15 เป็น DATA BUS ของ IBM ซึ่งต่อเข้ากับ D0-D15 ของ 8255 ได้โดยตรงซึ่งเป็นเส้น DATA ที่ใช้ในการรับส่งข้อมูลต่าง ๆ ระหว่างเครื่อง IBM กับ 8255 ซึ่งที่ขา DATA ทุกเส้นจะต้องต่อความต้านทาน PULL UP เพื่อให้ข้อมูลที่ส่งมาไม่ให้เกิดความผิดพลาด

AEN ต่อเข้ากับชุด DECODE จากการใช้งานที่ต่อใช้งานกับ 8255 AEN จะต้องมีลอจิกเป็น "0"

A3-A9 ต่อเข้ากับชุดดีโค้ดเพื่อให้สามารถเลือกแอดเดรสให้กับ 8255 และสามารถเปลี่ยนแปลงแอดเดรสได้โดยเปลี่ยนค่าที่ DIP SWITCH

CS ของ 8255 จะต่อเข้ากับชุดดีโค้ดเพื่อเลือกให้ 8255 ตัวใดตัวหนึ่งทำงานโดยถ้าขา CS มีลอจิก "0" DATA BUS ของ 8255 จะต่อเข้ากับ DATA BUS ของ IBM จึงสามารถอ่านและเขียนข้อมูลบน 8255 ได้ ซึ่งวงจรที่สมบูรณ์ดูได้จากวงจร 9.14 ซึ่งจะดีโค้ดให้ 8255 ตัวที่ 1 ACTIVE เมื่อขา Q ของ 688 ส่งสัญญาณลอจิก "0" มาที่ขา CS ของ 8255 ซึ่ง 8255 ตัวที่ 1 จะเริ่มที่แอดเดรส 30GH ส่วน 8255 ตัวที่ 2 จะเริ่มที่ แอดเดรส 307H โดยจะ ACTIVE เมื่อขา Q ของ 688 ส่งสัญญาณลอจิก "0" มาที่ขา CS ของ 8255 ตัวที่ 2

3.7.4 การใช้งาน 8255 INTERFACE CARD

การตั้งแอดเดรส (SET ADDRESS) จากวงจรดีโค้ดเดอร์ดังรูปที่ 3.32 จะเห็นว่าบิตที่เซ็ตได้มี A3-A7 และ A8-A9 ต้องเป็น "1" 8255 จึงได้รับ CS ดังนั้นแอด

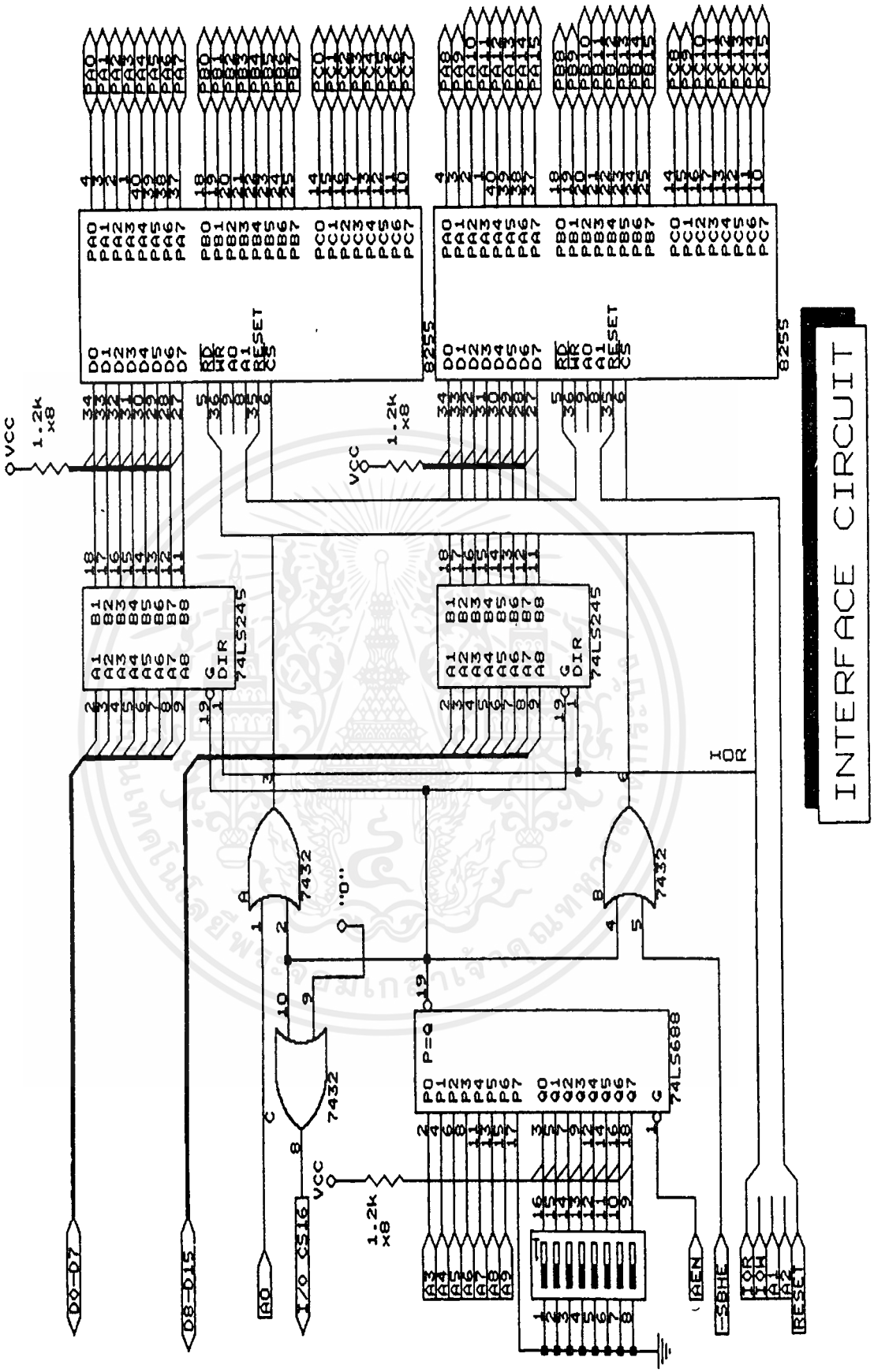
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เดรสที่เข็ทได้ทั้งหมดได้แก่ A3 A4 A5 A6 A7 A8 และ A9 เขียนได้ดังข้างล่างนี้

A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
X	X	X	X	X	X	1	1	0	0	0	0	0	X	X	X

ส่วนรายละเอียดเกี่ยวกับสัญญาณต่าง ๆ ที่ต่อออกมา 8255 ทั้ง 2 ตัว ซึ่งประกอบด้วย PORT A,PORT B และ PORT C ซึ่งต่อไปใช้งานควบคุมอุปกรณ์ต่าง ๆ หรือใช้ในการอ่านข้อมูลจากอุปกรณ์ภายนอกผ่านทาง 8255 รายละเอียดของสัญญาณต่าง ๆ ที่ต่อจาก SLOT ของ IBM ก็สามารถต่อไปใช้งานเพิ่มเติมได้อีกด้วยโดยได้ต่อผ่านทาง STRIP HEADER และบนการ์ดยังมีไฟเลี้ยงที่ระดับแรงดันต่าง ๆ ต่อไว้ให้อีกด้วย ซึ่งรายละเอียดอยู่ที่รูป 3.32

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



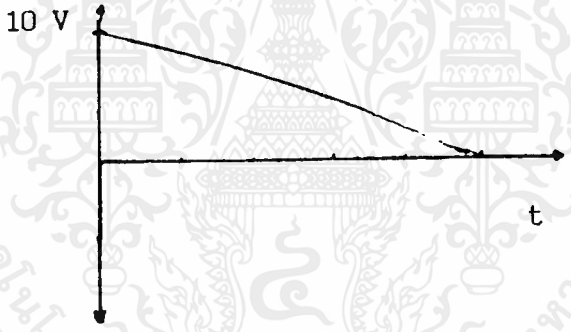
รูปที่ 3.32 วงจรการใช้งาน INTERFACE CARD 8255

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทำงานของ Dimmer

จะต่อกับไฟ 220v โดยใช้ fuse ขนาด 0.3 A เป็นตัวป้องกันจ่ายไปให้กับลง ไปจาก 220v เป็น 24-0-24 ทำการเร็คติไฟร์โดยใช้ Diode D1 และ D2 ผ่านมาเข้าที่ ไอซี recgulate 7815 จะเป็นไฟ D.C. + 15 v และจะจ่ายให้แก่ Zener-diode. 9.1v เป็นไฟ vcc แก่ชุดกำเนิดสัญญาณ 1+Cos wt ซึ่งมี VR1 เป็นตัวปรับระดับรูปสัญญาณให้ลงมาที่ระดับ 0 v



รูปสัญญาณ 1+Cos wt

เมื่อได้รูปสัญญาณ 1+Cos wt แล้วจะนำมาเปรียบเทียบสัญญาณที่ส่งมาจากส่วนควบคุมจาก DAC ซึ่งจะเป็นค่าแรงดัน 0-10 v เราจะทำการปรับ VR2 เพื่อให้ค่ากระแสที่จะไปทริกไดรแอดให้ทำงานนำกระแส 0%-100% ที่แรงดัน 0-10 v voltage ที่ได้จาก Comparator จะไปดูดซับกระแส Transistor ซึ่งไปควบคุมกระแสที่จะผ่าน LED ใน OPTO Transistor ทำให้ Transistor ในตัว OPTO MOC 3020 นำกระแสจากชุดบริดส์เร็คติไฟร์ 220 v ซึ่งผ่าน R 360 ซึ่งเป็นตัวกำหนดกระแส และนำกระแสส่วนนี้ไปทริกเกตส์ของ Triac ซึ่งจะมี วงจรสนับเบอร์เป็นตัวควบคุมสัญญาณรบกวนโดยจะใช้ R อนุกรมกับ C ต่อคร่อม Triac ไว้ เมื่อ Triac เกิดการทริกเกตที่ค่าต่าง ๆ กัน ทำให้เกิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การนำกระแสที่ต่างกัน และได้ระดับแรงดันที่จ่ายแก่หลอดไฟค่าต่าง ๆ กันตามไปด้วย ความสว่างของหลอดไฟก็จะเป็นไปตามระดับของแรงดันที่เข้ามา Control 0-10 v จะถูกเปรียบเทียบกับค่าความสว่าง 0-100% ในการ Dimmer.

การนำไปใช้งาน

จากระบบการควบคุมสามารถควบคุม Dimmer ได้หลาย chanal โดยใช้จำนวนสายที่น้อยและสามารถส่งได้ไกล เป็นการประหยัดสายไม่ต้องใช้สายจำนวนมากตามจำนวน Chanal และสามารถแสดงผลทางจอ monitor ซึ่งสามารถควบคุมแบบ Automatic และ Mannual ได้ซึ่งประโยชน์ของ Dimmer มีดังต่อไปนี้

1. เสริมสร้างบรรยากาศ ปรับความสว่างให้เหมาะสม เช่น ในโรงแรมจะติดตั้งในห้อง lobby เพื่อปรับบรรยากาศให้เทียบเคียงกับธรรมชาติ เพื่อให้ได้มุมมองและภาพพจน์ในการตกแต่งนั้น ๆ ให้แตกต่างกันออกไปได้หลายรูปแบบ

2. ใช้ในห้องประชุม เพื่อปรับความสว่าง ให้เหมาะกับการทำงาน เช่นในกรณีการสัมมนาจะเปิดไฟเต็มที่ เพื่อใช้ในการแลคเชอร์หรือในกรณีต้องการฉายสไลด์ที่จะต้องลดความสว่างลง แต่ในบางส่วนก็ให้มีแสงพอที่จะ lecture ได้ เช่น ข้างหน้าจะมืดข้างหลังจะพอดีแสงได้ เราสามารถปรับความสว่างได้เป็น % คือ 0-100%

ในงานเวทีจะเน้นไปในทางการสร้างรูปแบบของแสง ให้เข้ากับบรรยากาศในงานการแสดงนั้น ๆ เช่นในเวทีคอนเสริต์ รูปแบบของแสงจะแตกต่างจากการแสดงละคร

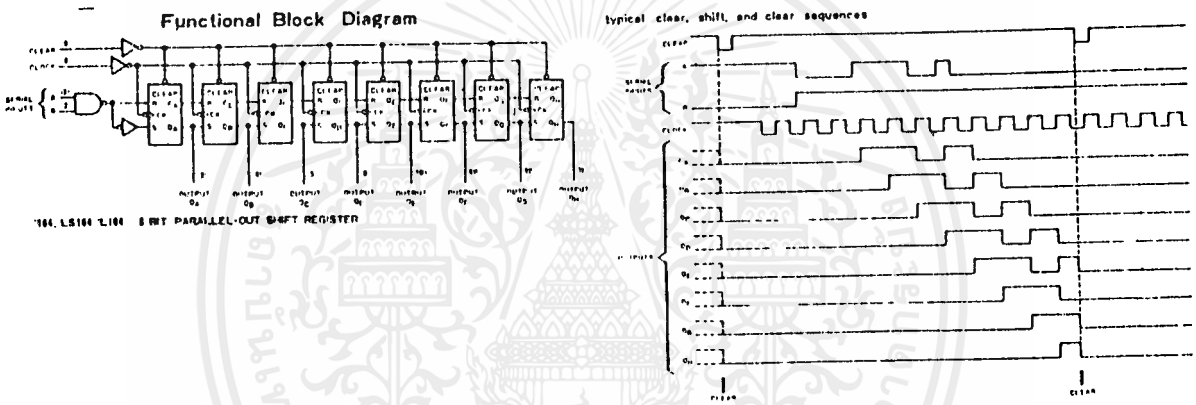
ในโรงงานจะใช้ในรูปแบบการประหยัดพลังงาน ช่วยลดค่าแอร์ตัวอย่าง เช่น ในโรงงานที่เป็นห้องกระจก เวลากลางคืนจะเปิดไฟเต็มที่ แต่เมื่อถึงตอนกลางวันจะมีแสงจากภายนอกเข้ามาช่วยทำให้เราสามารถ Dim ไฟลงได้ ช่วยลดความร้อนที่ออกมาจากหลอดไฟ.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

วรจร DECODER

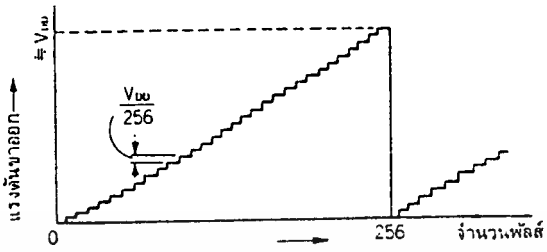
จากข้อมูล จาก Interface Card การส่งสัญญาณจะเป็นแบบ serial เรา ไม่สามารถนำไปใช้งานโดยตรง จึงต้องนำมาทำการถอดรหัสเป็นสัญญาณ Parallel โดย ใช้ ไอซี 74164 ซึ่งเป็น ไอซี serial shift Register 8 Parallel Output



รูปที่ 6.1 serial shift Register 8 Parallel Output

จาก Data Book เราให้ serial input B เป็น "high" logic (ดูจาก timeing diagram) และสัญญาณ Data lavel และ chanel select จะเข้าที่ขา serial input A โดยให้ขา clear เป็น "high" logic และสัญญาณ Clock เข้าที่ ขา Clock (9) เนื่องจากข้อมูลทั้ง Data level และ chanel select เป็นข้อมูล 16 บิตจึงต้องใช้ ไอซี 74164 2 ตัว เพื่อให้ทำงานได้ครบทั้ง 16 บิต และเนื่องจากข้อมูล เป็นแบบ serial จะ shift data อยู่ตลอดเวลา ทำให้ข้อมูลไม่คงที่จึงต้องใช้ ไอซี # 74373 เป็น ไอซี Octal-D-Type Transparant Latch and Edge-Triggered Fliggered Flig Flops เป็นตัว latch เพื่อให้ข้อมูลครบทั้ง 16 บิต แล้วจึงส่งข้อ มูลออกมาโดยให้สัญญาณ output control เป็น "Low" เมื่อมี Data level และ chanel Select มาครบทั้ง 16 บิตแล้วซึ่งจะทำให้ข้อมูลที่ได้ ถูกต้องแล้วจึงส่งไปยัง ภาค Digitol to Analog Converter ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.2 ข) แรงดันขาออกและจำนวนพัลส์

วงจร R-2R Ladder Converter

เป็นวงจรที่ได้รับความนิยมเนื่องจากใช้งานได้ดีพอสมควร เหมาะกับงานที่ไม่ต้องการความละเอียดสูง และมีต้นทุนต่ำ

สูตรการคำนวณค่า Resister DAC

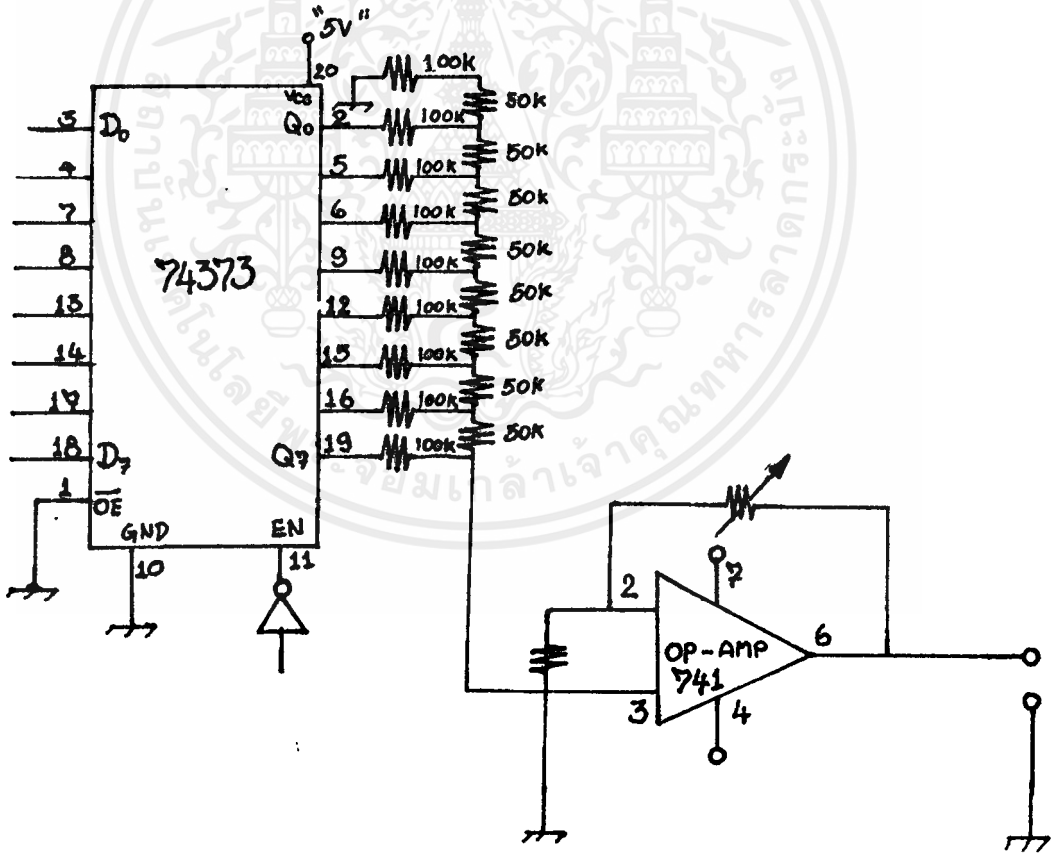
$$V_o = -[(D_7/2^0)+(D_6/2^1)+(D_5/2^2)+(D_4/2^3)+(D_3/2^4)+(D_2/2^5)+(D_1/2^6)+(D_o/2^7)].(R_f/$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

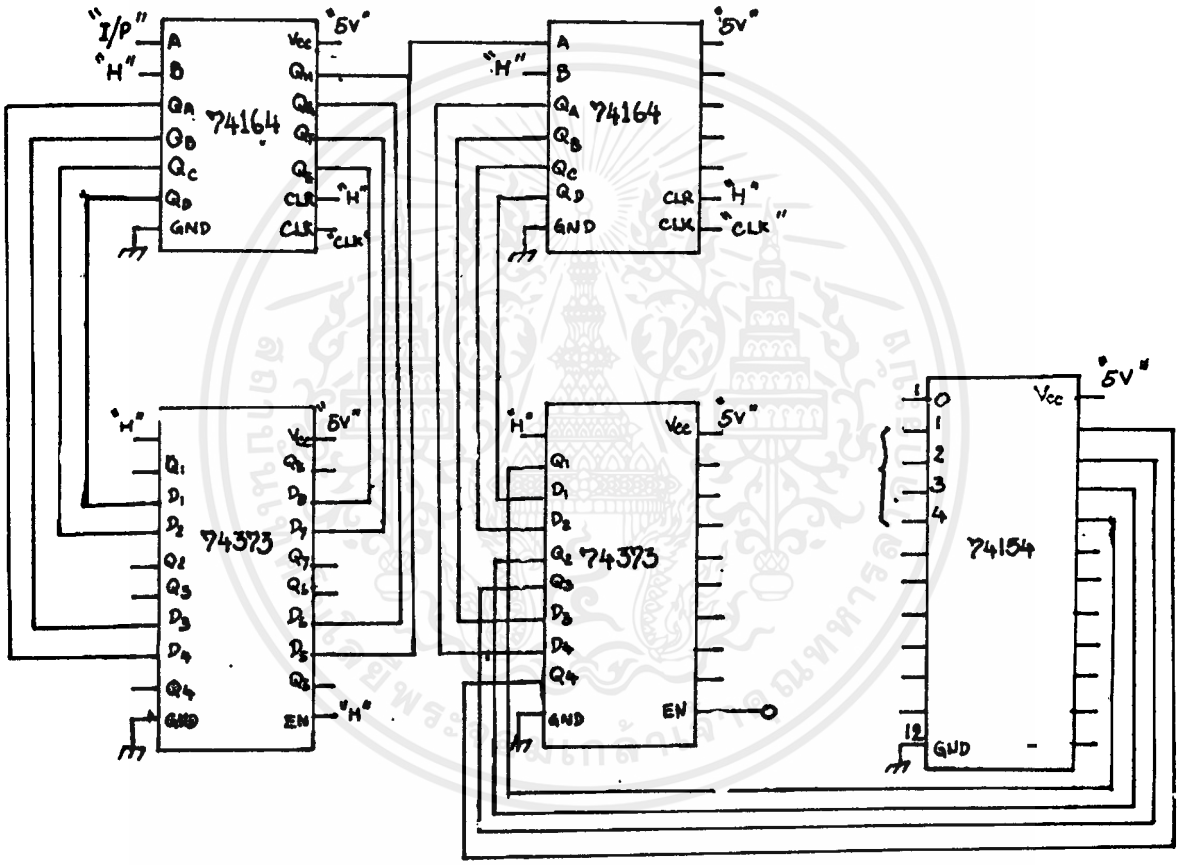
Digital to Analog Converter (DAC)

สำหรับในบทนี้จะกล่าวถึงหลักการทำงานของ Digital to Analog Converter (DAC) ที่ใช้กับเครื่อง Dimmer Control by Computer IBM/AT โดยได้ทำการออกแบบใช้ Digital to Analog Converter (DAC) แบบ R-2R ladder (แบบวงจรชั้นบันไดของตัวต้านทาน) ซึ่งมีวงจรดังแสดงในรูปที่ 4.1



รูปที่ 5.1 ก) วงจร แบบ R-2R ladder

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.2 Decoder Circuit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7

PROGRAM ควบคุมการ DIMMING

Program เป็นส่วนที่ใช้ในการควบคุมระบบ Hardware ให้ทำงานตาม Menu ที่สร้างขึ้น เพื่อให้มีความสะดวกในการสั่งงานจากผู้ใช้เครื่อง เพราะฉะนั้นโปรแกรมจึงใช้การเขียนแบบกราฟิก โดยใช้ภาษา C ในการเขียนโปรแกรม

7.1 หลักการทำงานของโปรแกรม

หลักการทำงานดูจากรูปที่ 7.1 ซึ่งเป็น Flow Chart ของ Main โปรแกรม จะมีขั้นตอนการทำงานของโปรแกรมดังนี้

เมื่อเปิดเครื่องโปรแกรมเริ่มทำงาน โดยจะทำการเช็ค Arrow key ก่อน หากไม่ได้ต่อ Arrow key เครื่องก็จะออกจากโปรแกรม แต่หากมีการต่อ Arrow key ไว้ เครื่องก็จะทำงานต่อไปตามโปรแกรม โดยจะเข้าไปใน MAIN MENU จะมี 7 Function ให้เลือกดังนี้ คือ

[1] MANUAL FUNCTION

[2] AUTO FUNCTION

[3] 1 >> 4 FUNCTION

[4] 1 << 4 FUNCTION

[5] FLASH FUNCTION

[6] CLEAR FUNCTION

[7] EXIT FUNCTION

[1] MANUAL FUNCTION

เป็นการควบคุมการหรี่ไฟ โดยใช้ Arrow key เป็นตัว click การเพิ่มหรือลดค่า DATA ความสว่างได้จากที่จะไปยัง Port แล้วส่งไปยังระบบ Hardware ซึ่งจะสามารถจะปรับได้ตั้งแต่ 0-100 % โดยจะมีแท่งกราฟแสดงการเพิ่มหรือลดของ DATA ความสว่างจาก0-100 %

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

[2] AUTO FUNCTION

เป็นการเพิ่มค่า DATA ความสว่างจาก 0-100 % ทีละ Chanel โดย เริ่มจาก Chanal ที่ 1 เพิ่มจาก 0-100 % แล้วลดลงมาที่ 0 % แล้วจึงเริ่ม Chanel ที่ 1, 2,3,4 ลำดับต่อไป แล้วจึงวนกลับมาเริ่มที่ Chanel ที่ 1ใหม่

[3] 1 >> 4 FUNCTION

เป็นการนำค่า DATA ความสว่างที่เราใส่เข้าไปในการควบคุมแบบ MANUAL มาออก Port โดยเริ่มแสดงจาก Chanel ที่ 1 ไปจนถึง Chanel ที่ 4 แล้วมาเริ่มที่ Chanel ที่ 1 ใหม่วนไปเรื่อย ๆ

[4] 1 << 4 FUNCTION

เป็นการนำค่า DATA ความสว่างที่เราใส่เข้าไปในการควบคุมแบบ MANUAL มาออก Port โดยเริ่มแสดงจาก Chanel ที่ 4 ไปจนถึง Chanel ที่ 1 แล้วมาเริ่มที่ Chanel ที่ 4 ใหม่วนไปเรื่อย ๆ

[5] FLASH FUNCTION

เป็นการนำค่า DATA ความสว่าง 100 % และ 0 % สลับกันมาแสดงออกแต่ละ Chanel โดยเริ่มแสดงจาก Chanel ที่ 1 ไปจนถึง Chanel ที่ 4 แล้วมาเริ่มที่ Chanel ที่ 1 ใหม่วนไปเรื่อย ๆ

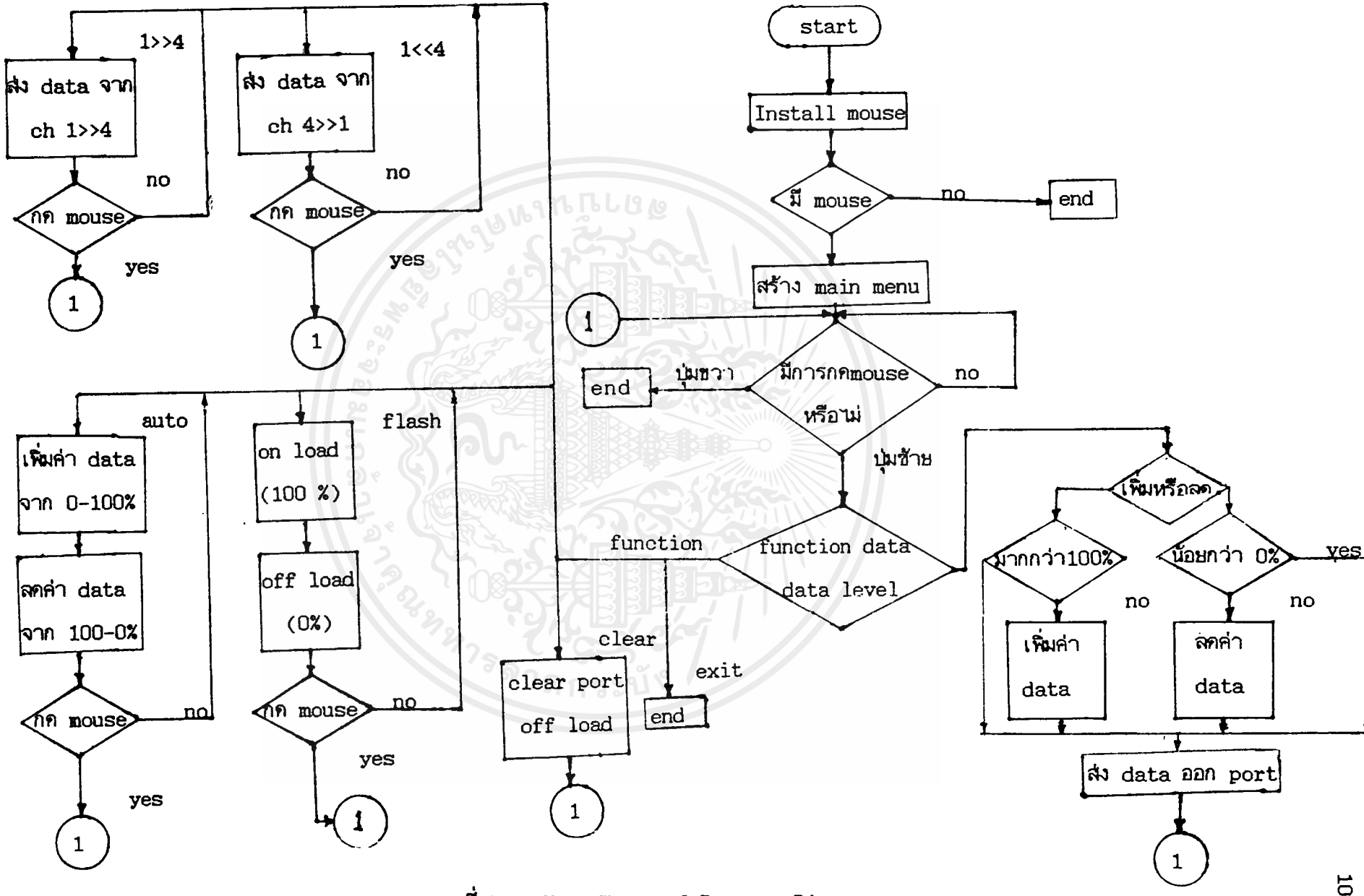
[6] CLEAR FUNCTION

เป็นการนำค่า DATA ความสว่าง 0 % ออกทั้งหมด 4 Chanel หลอดไฟจะดับหมด

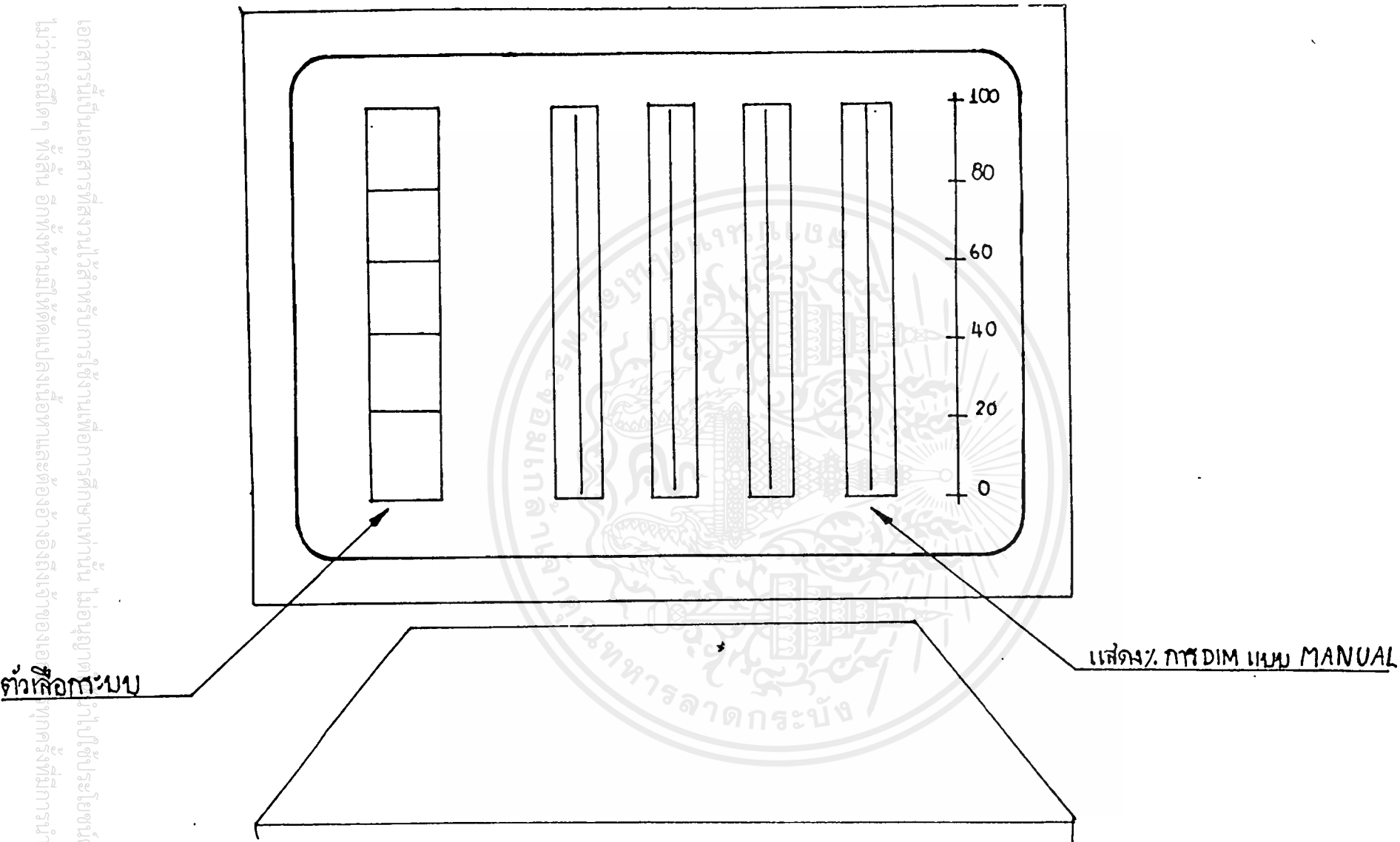
[7] EXIT FUNCTION

เป็นการออกจากโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.1 Flow Chart of Program Dimmer



รูปที่ 7.2 หน้าจอแสดงการควบคุมกราฟฟิก

```
#include <dos.h>
#include <conio.h>
#include <stdio.h>
#include <stdlib.h>
#include <mouse.h>
#include <dimmer.h>

#define PORT0x300
#definetdelay1000

int adj,err,chold,loop,fold,Dout[9],output[5][512];
void msc(void)
{
 m1 = 3;  mouseinit();
if (_BX!=0) err++;
}

void show(int sh)
{
setfillstyle(1,14);
bar(chx[sh]+1,374,chx[sh]+wide-1,374-(data[sh]*3));
setfillstyle(1,5);
bar(chx[sh]+1,76,chx[sh]+wide-1,374-(data[sh]*3));
}
void RTL(void)
{
int lp;
   for (lp=0;lp<=7;lp++) {
 outportb(PORT,Dout[lp]);
 outportb(PORT+1,0x00);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 outportb(PORT+1,0xFF);
 outportb(PORT+1,0x00);
   }
}
void LTR(void)
{
int lp;
   for (lp=7;lp>=0;lp--) {
 outportb(PORT,Dout[lp]);
 outportb(PORT+1,0x00);
 outportb(PORT+1,0xFF);
 outportb(PORT+1,0x00);
   }
}
void chout(int ch)
{
int dtemp,lp;
/********  Channal Dimmer Out **********/
outportb(PORT+2,0xFF);
dtemp = ch;
for (lp=0;lp<=7;lp++) {
Dout[lp] = dtemp % 2; dtemp /= 2;
}
chold = ch;
LTR();
}


void dataout(int dat)
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int dtemp,lpd;
if (func!=1) {
dtemp = (int)(dat*2.55);
if (dat!=0) dtemp++;
}
/********  Data Dimmer Out *********/
for (lpd=0;lpd<=7;lpd++) {
Dout[lpd] = dtemp % 2;  dtemp /= 2;
}
m1 = 3;  mouseinit();
if (_BX!=0) err=1;
LTR();
outportb(PORT+2,0x00);
msc();
}

void TEST(void)
{
int loop2,pan;
START:
for (loop=1;loop<=4;loop++) {
   for (loop2=0;loop2<255;loop2++) {
chout(loop);
dataout(loop2);
   }
   for (pan=255;pan>=0;pan--) {


chout(loop);
dataout(pan);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    }
    msc();
    if (err!=0) goto EX;
}
if (err==0) goto START;
EX:;
}
void FLASH(void)
{
int loop2,pan;
START:
for (loop=1;loop<=4;loop++) {
for (loop2=0;loop2<=1;loop2++) {
chout(loop);
show(loop);
if ((loop2%2)==0) pan=100; else pan=00;
datsout(pan);
delay(tdelay);
msc();
}
msc();
          display(loop);
if (err!=0) goto EX;
}
msc();
if (err==0) goto START;


EX:;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void REW(void)
{
START:
for (loop=4;loop>=1;loop--) {
chout(loop);
show(loop);
dataout(data[loop]);
if (err!=0) goto EX;
                display(loop);
delay(tdelay);
msc();
                display(loop);
if (err!=0) goto EX;

}
msc();
if (err==0) goto START;
EX:;
}
void FWD(void)
{
START:
for (loop=1;loop<=4;loop++) {
chout(loop);
show(loop);
dataout(data[loop]);


if (err!=0) goto EX;
delay(tdelay*2);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
msc();
                display(loop);
if (err!=0) goto EX;
}
msc();
if (err==0) goto START;
EX:;
}
void dimclear(void)
{
int lp;
for (lp=4;lp>=0;lp--) {
chout(lp);
datsout(00);
}
}
void main(void)
{
int c;
nosound();
func = 6; fold = 0;
system("A:\MOUSE.COM");
clrscr();
gotoxy(30,11);
textcolor(12+128);
cprintf("Wait to Clear Data\n");


m1 = 0;mouseinit();
if (_AX == 0) {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
gotoxy(1,24);
printf("\nMouse not found\n");
exit(1);
}
     delay(1);
frame();
outportb(PORT+3,0x80);
dimclear();
     button(6,0);
setcolor(12);
outtextxy(XMAX/2,YMAX-10,"PC Control Dimmer
by 34131101 & 34131129   KMITL");
settextjustify(1,1);
MouseSetX(10,600);
MouseSetY(10,410);
MouseSetXY(60,350);
mouseon();
setmousecursor(glove);
_BX = 0;

/************   Check Mouse Press  ***************/
while (_BX!=2){
while (_BX==0) {  m1 = 3;  mouseinit();  }
if (_BX == 1)  {  while (_BX!=0) {  m1=3; mouseinit();
x  = _CX; y  = _DX; adj = 0;
if ((_CX>=36)&&(_CX<=119)) {


for (c=1;c<=6;c++) {
if ((_DX>=begfy[c])&&(_DX<=endfy[c])) {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
mouseoff();
func=c;   err=0;
/*button(func,1);    */
if (func!=fold) button(func,1);
mouseon();
/*** ** ***/
switch (func) {
case 1: TEST();fold=func; break;
case 2: FWD();fold=func; break;
case 3: REW();  fold=func; break;
case 4: FLASH();    fold=func; break;
case 5: dimclear(); fold=func; break;
case 6: goto EX;
}
/**** * ****/
mouseoff(); button(func,0); mouseon();
outportb(PORT+3,0x80);
}
while (_BX!=0) {  m1 = 3;  mouseinit();  }

}
}
if ((_DX>=50)&&(_DX<=75)) {
for (c=1;c<=4;c++) {
if ((_CX>=chx[c])&&(_CX<=(chx[c]+wide))) {
mouseoff();


ch_no = c; adj++;
chbutton(ch_no,100,1);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if (ch_no!=0) show(ch_no);
if (data[c]<100) { data[c]++; display(ch_no); }
chbutton(ch_no,100,0);
mouseon();
chout(ch_no);
dataout(data[ch_no]);
adj = 0;
}
while (_BX!=0) {  m1 = 3;  mouseinit();  }
}
}

if ((_DX>=375)&&(_DX<=400)) {
for (c=1;c<=4;c++) {
if ((_CX>=chx[c])&&(_CX<=(chx[c]+wide))) {
mouseoff();
ch_no = c; adj++;
chbutton(ch_no,0,1);
if (ch_no!=0) show(ch_no);
if (data[c]>0) { data[c]--; display(ch_no); }
chbutton(ch_no,0,0);
mouseon();
chout(ch_no);
dataout(data[ch_no]);
adj = 0;
}

while (_BX!=0) {  m1 = 3;  mouseinit();  }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}

}

}
EX:
mouseoff();
closegraph();
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 8

บทสรุปและข้อเสนอแนะ

ในการจัดทำโครงงานนี้ เพื่อเป็นการศึกษาถึงการควบคุมการใช้งาน Dimmer ว่าเราจะสามารถควบคุมได้อย่างไร

ในปัจจุบันนี้การใช้งาน Dimmer มีอย่างแพร่หลายการศึกษาในครั้งนี้ จึงนับว่าจะสามารถจำนวน ประโยชน์ให้กับผู้สนใจในภายหน้าได้เป็นอย่างดี สำหรับคุณสมบัติของ Dimmer ที่ควรศึกษายังมีความละเอียดมากกว่าในหนังสือเล่มนี้ จะขอยกตัวอย่าง เช่น คุณลักษณะการป้องกันสัญญาณรบกวน ซึ่งมิได้กล่าวไว้เลย แต่สามารถศึกษาเพิ่มเติม ได้จากเอกสารอ้างอิงตามที่แสดงไว้

ในส่วนของวงจรควบคุมที่ใช้ IC#74164 เป็นตัวรับข้อมูลนับว่ามีความสะดวกและประหยัดมากกว่า การใช้ Micro Controler โดยการรับการควบคุมจาก Computer โดยตรง จากการทดลองด้วยโปรแกรมต่าง ๆ พบว่าการควบคุมแบบแสดงผลทางกราฟิกนั้นจะใช้งานได้ดี และแม่นยำมาก แต่ในการควบคุมยังมีข้อผิดพลาดอยู่มาก ทั้งนี้ทั้งนั้นเมื่อทำการวิเคราะห์ดูแล้วสามารถเห็นข้อผิดพลาดได้ดังนี้

การใช้งาน IC Digital to Analog # MC 1408 นั้น ค่อนข้างที่จะเสียได้ง่ายจึงได้เปลี่ยนมาใช้ วงจรแบบ R-2R Ladder เป็นชุด DAC แทนจะเหมาะสมกับลักษณะของงานนี้มากกว่า ซึ่งไม่ต้องการความละเอียดมาก

แต่อย่างไรก็ตามโครงการนี้ ก็มีวัตถุประสงค์เพื่อใช้ในการศึกษา ทดลอง เพื่อเป็นพื้นฐานแนวความคิด ในการนำไปใช้งานและเพื่อพัฒนาต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

[1] อาจารย์อุดมศักดิ์ ยั่งยืน , "Power Electronics I" , ภาควิชาวิศวกรรมไฟฟ้า คระวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าธนบุรี

[2] รศ. กฤษดา วิศวธีรานนท์ , "ไอซีดิจิตอล" , บริษัท ซีเอ็ดยูเคชั่น จำกัด , หน้า 204-208 พิมพ์ครั้งที่ 2 พ.ศ. 2531

[3] สุภาพ , โรจิต , วิชูชย์ ฯ , "Interface Card IBM PC/AT" , ปริญญานิพนธ์ปริญญาบัณฑิต ภาควิชาเทคโนโลยีไฟฟ้าอุตสาหกรรม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ

[4] "คู่มือ / เทียบเบอร์ ไอซี TTL" , บริษัท ซีเอ็ดยูเคชั่น จำกัด , 2522

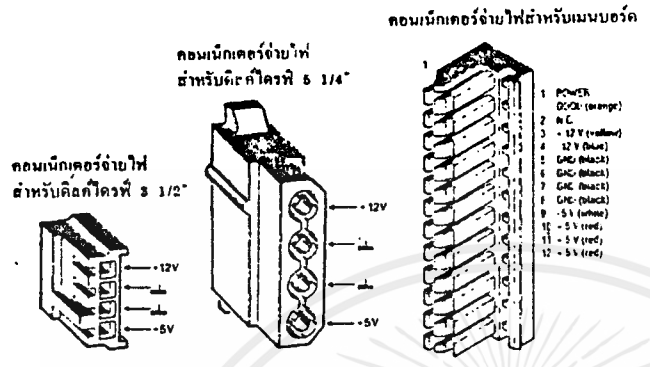
[5] THEO DORE EBOGART. JR , INTRODUCTION TO DIGITAL CIRCUITS , MC GRAW-HILL INTERNATIONAL EDITIONS , 1992

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

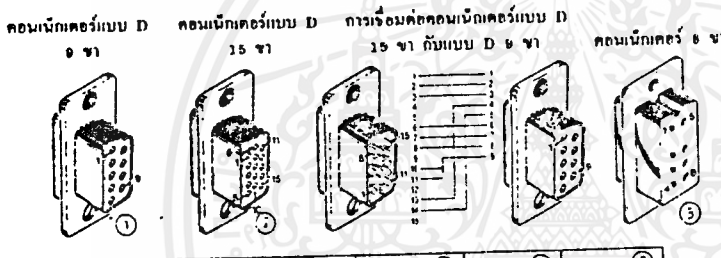
ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CONNECTOR กับตำแหน่งขาและสัญญาณ



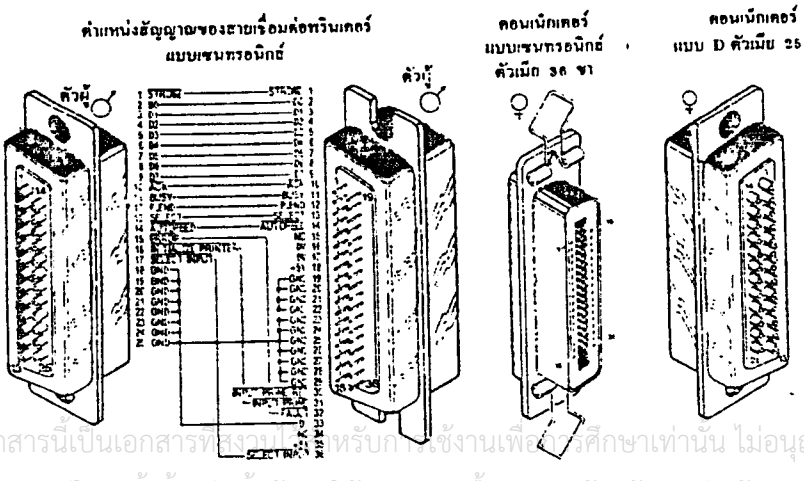
รูปที่ 1 คอนเน็กเตอร์
สำหรับภาคจ่ายไฟ
แบบสวิตชิ่ง



ขา	MDA ① (Hercules)	CGA ① dig. RGBI	EGA ② dig. RGBI	NEC ① dig. RGBI	VGA ② an. RGB
1	ground	ground	ground	(present)	R - out
2	ground	ground	R'	R	G - out
3	n.c.	R	R	G	B - out
4	n.c.	G	G	B	mon. ID bit 2
5	n.c.	B	B	ground	ground
6	intensity	intensity	G'	ground	R - return
7	video	n.c.	B'	h - sync (+)	G - return
8	h - sync (+)	h - sync (+)	h - sync (+)	v - sync (-)	B - return
9	v - sync (-)	v - sync (-)	v - sync (-)		(no pin)
10					sync - return
11					mon. ID bit 0
12					mon. ID bit 1
13					h - sync
14					v - sync
15					reserved

หมายเหตุ 1. dig. RGBI คือ สัญญาณ RGB แบบดิจิตอล
2. an. RGB คือ สัญญาณ RGB แบบอนาลอก
3 ตัวเลขในวงกลมคือแบบของคอนเน็กเตอร์ที่ใช้สัมพันธ์กับในรูป

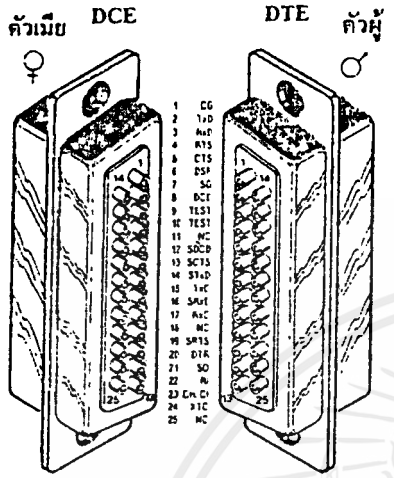
รูปที่ 2 แสดง
คอนเน็กเตอร์และสัญญาณ
สำหรับจอภาพแบบต่าง ๆ



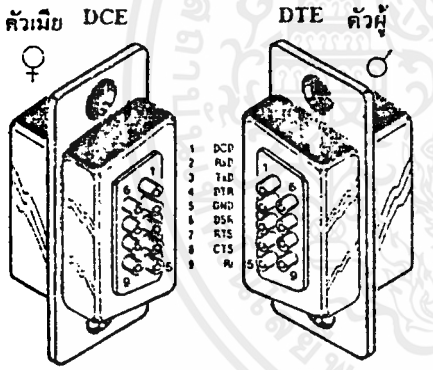
รูปที่ 3 สายเชื่อมต่อ
พริ้นเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

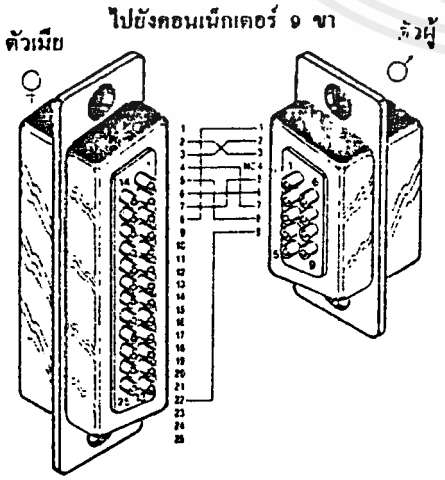
คอนเน็กเตอร์แบบ D 25 ขา



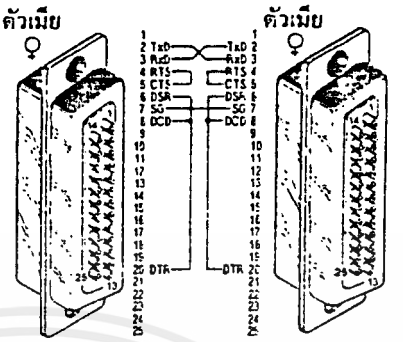
คอนเน็กเตอร์ 9 ขา



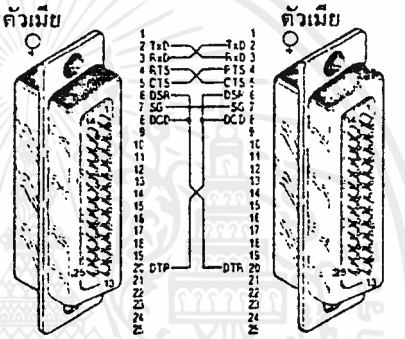
การเชื่อมต่อสัญญาณจากคอนเน็กเตอร์ 25 ขา
ไปยังคอนเน็กเตอร์ 9 ขา



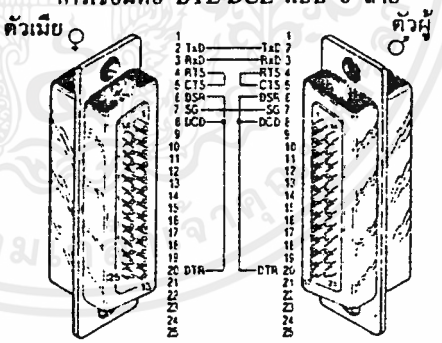
การเชื่อมต่อ DTE-DTE แบบ 3 สาย



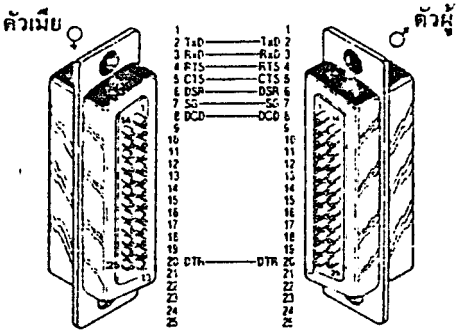
การเชื่อมต่อ DTE-DTE แบบ full connection



การเชื่อมต่อ DTE-DCE แบบ 3 สาย



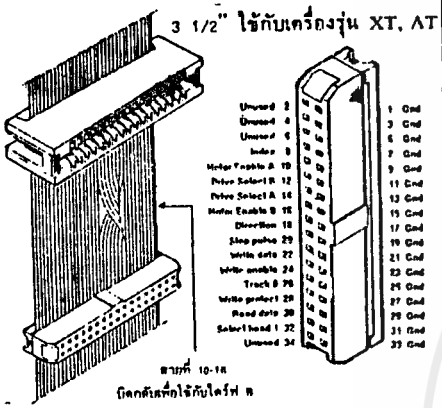
การเชื่อมต่อ DTE-DCE แบบ full connection



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

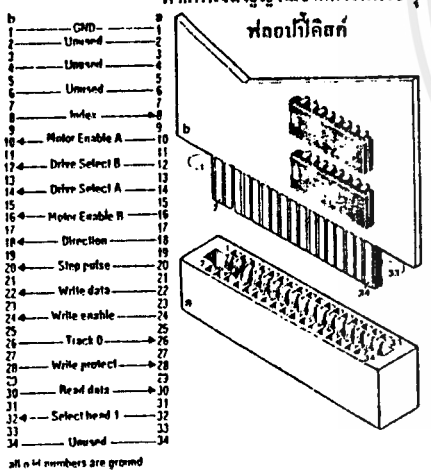
ฟลอปปี้ดิสก์อินเตอร์เฟส

หัวต่อสำหรับไดรฟ์ขนาด 3 1/2" ใช้กับเครื่องรุ่น XT, AT



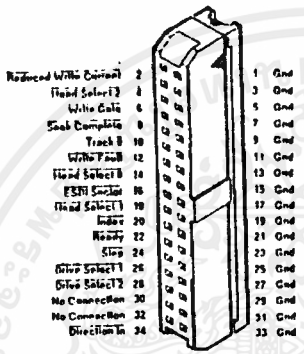
สายที่ 10-16
ปิดกลับหัวก่อนใช้กับไดรฟ์ B

ตำแหน่งสัญญาณจากการ์ดควบคุมฟลอปปี้ดิสก์

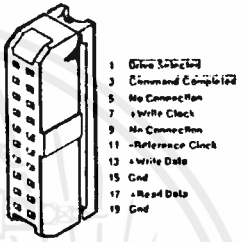


ฮาร์ดดิสก์อินเตอร์เฟสมาตรฐาน ST506

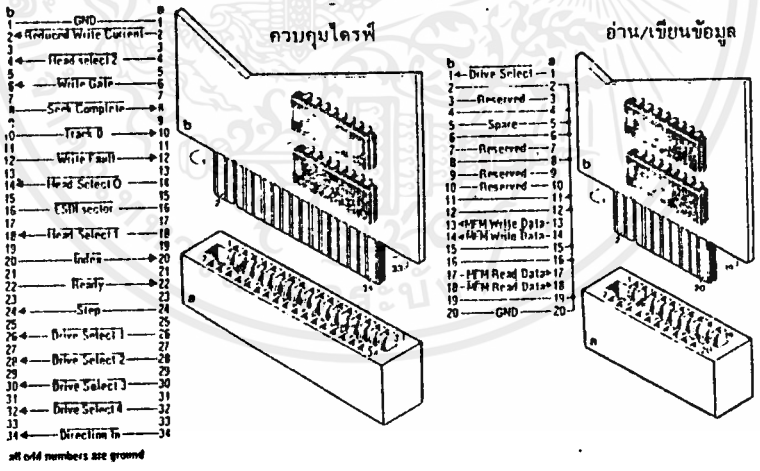
ควบคุมไดรฟ์



อ่าน/เขียนข้อมูล



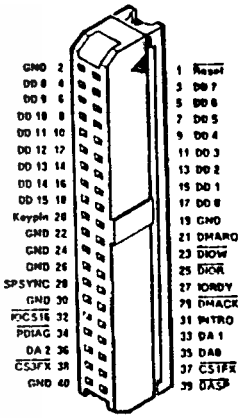
ตำแหน่งสัญญาณจากการ์ดควบคุมฮาร์ดดิสก์



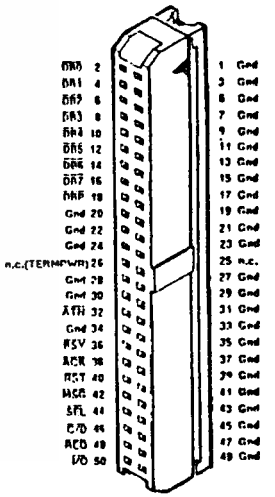
ควบคุมไดรฟ์

อ่าน/เขียนข้อมูล

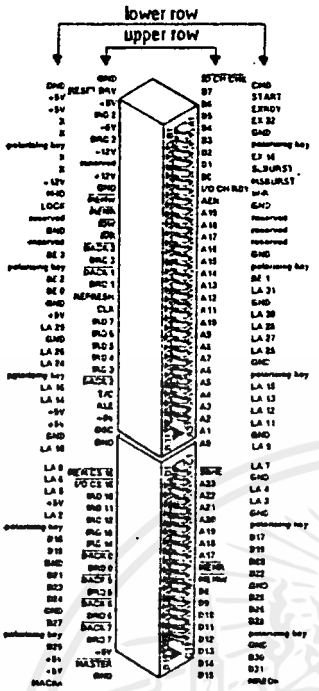
ฮาร์ดดิสก์อินเตอร์เฟสมาตรฐาน IDE



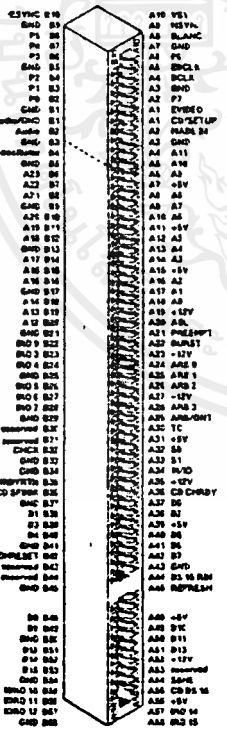
ฮาร์ดดิสก์อินเตอร์เฟสมาตรฐาน SCSI



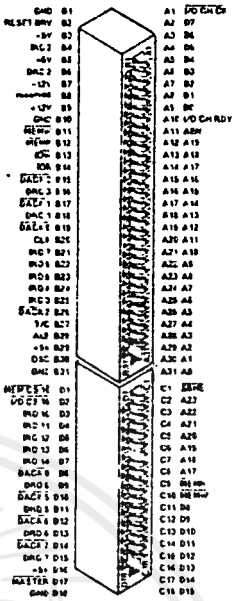
บัสมาตรฐาน EISA



บัสมาตรฐาน MCA



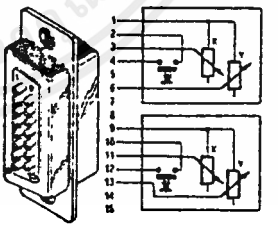
ขยายระบบมาตรฐาน IBM



คอนเน็กเตอร์สำหรับคีย์บอร์ดรุ่น PS2 (6 ขา)

คอนเน็กเตอร์คีย์บอร์ดแบบ DIN (5 ขา)

คอนเน็กเตอร์สำหรับจอยสติกแบบ D (15 ขา)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

54154/74154 4-Line-to-16-Line Decoder/Demultiplexer

	Schottky TLL					High-Speed TLL					Low-Power Schottky TLL					Standard TLL					Low-Power TLL				
	Device Type	Package C	P	M	CF	Device Type	Package C	P	M	CF	Device Type	Package C	P	M	CF	Device Type	Package C	P	M	CF	Device Type	Package C	P	M	CF
T.I.																SN54154	J①			W①	SN54L154	J①			
																SN74154	J①	N①			SN74L154	J①	N①		
FAIRCHILD																FC74154	⑭①								
																	⑭①	P①							
MOTOROLA																SN74154		P①							
N.S.C.											DM54LS154		J①			DM54154	J①			F①	DM54L154A	D①	N①		F①
											DM74LS154		J①			DM74154	J①	N①		F①	DM74L154A	D①	N①		F①
PHILIPS											N74LS154		⑭			N74154									
SIGNETICS																S54154	F①	N①		Q①					
																N74154		N①							
SIEMENS																FLY141		①							
FUJITSU																									
HITACHI											HD74LS154		P①			HD74154/HD2580	①	P①							
MITSUBISHI																M53354		P①							
NEC																μPB2154	D①								
TOSHIBA																									

Electrical Characteristics SN54154 SN74154

absolute maximum ratings over operating free-air temperature range

Supply voltage, V_{CC}	7V	Operating free-air temperature range	SN54*	55°C to 125°C
Input voltage	5.5V		SN74*	0°C to 70°C
		Storage temperature range		−65°C to 150°C

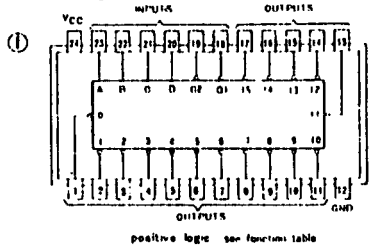
recommended operating conditions

	SN54154 MIN	NOM	MAX	SN74154 MIN	NOM	MAX	UNIT
Supply voltage, V_{CC}	4.5	5	5.5	4.75	5	5.25	V
High-level output current, I_{OH}			−800			−800	μA
Low-level output current, I_{OL}			16			16	mA
Operating free-air temperature, T_A	−55		125	0		70	°C

electrical characteristics over recommended operating free-air temperature range

PARMETER*		TEST CONDITIONS†		MIN	TYP‡	MAX	UNIT
V_{IH}	High-level input voltage			2			V
V_{IL}	Low-level input voltage					0.8	V
V_I	Input clamp voltage	V_{CC} = MIN, I_I = −12mA				−1.5	V
V_{OH}	High-level output voltage	V_{CC} = MIN, V_{IH} = 2V, V_{IL} = 0.8V, I_{OH} = −800μA		2.4	3.4		V
V_{OL}	Low-level output voltage	V_{CC} = MIN, V_{IH} = 2V, V_{IL} = 0.8V, I_{OL} = 16mA			0.2	0.4	V
I_I	Input current at maximum input voltage	V_{CC} = MAX, V_I = 5.5V				1	mA
I_{IH}	High-level input current	V_{CC} = MAX, V_I = 2.4V				40	μA
I_{IL}	Low-level input current	V_{CC} = MAX, V_I = 0.4V				−1.6	mA
I_{OS}	Short-circuit output current*	V_{CC} = MAX,	SN54*	−20		−55	mA
			SN74*	−18		−57	
I_{CC}	Supply current	V_{CC} = MAX	SN54*		34	49	mA
		See Note	SN74*		34	56	
t_{PLH}	from A, B, C, or D inputs through 3 levels of logic	V_{CC} = 5V, T_A = 25°C, C_L = 15pF, R_L = 400Ω			24	36	ns
t_{PHL}	from A, B, C, or D inputs through 3 levels of logic				22	33	ns
t_{PLH}	from either strobe input				20	30	ns
t_{PHL}	from either strobe input				18	27	ns

Pin Assignment (Top View)



positive logic see function table

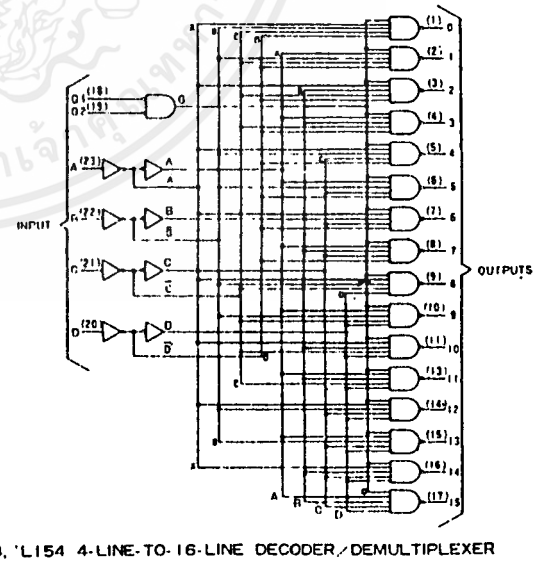
Function Table

154, 'L154

INPUTS G1	G2	D	C	B	A	OUTPUTS 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
L	L	L	L	L	L	L	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H
L	L	L	L	L	H	H	L	H	H	H	H	H	H	H	H	H	H	H	H	H	H
L	L	L	L	H	L	H	H	L	H	H	H	H	H	H	H	H	H	H	H	H	H
L	L	L	L	H	H	H	H	H	L	H	H	H	H	H	H	H	H	H	H	H	H
L	L	L	H	L	L	H	H	H	H	L	H	H	H	H	H	H	H	H	H	H	H
L	L	L	H	L	H	H	H	H	H	H	L	H	H	H	H	H	H	H	H	H	H
L	L	L	H	H	L	H	H	H	H	H	H	L	H	H	H	H	H	H	H	H	H
L	L	L	H	H	H	H	H	H	H	H	H	H	L	H	H	H	H	H	H	H	H
L	L	H	L	L	L	H	H	H	H	H	H	H	H	L	H	H	H	H	H	H	H
L	L	H	L	L	H	H	H	H	H	H	H	H	H	H	L	H	H	H	H	H	H
L	L	H	L	H	L	H	H	H	H	H	H	H	H	H	H	L	H	H	H	H	H
L	L	H	L	H	H	H	H	H	H	H	H	H	H	H	H	H	L	H	H	H	H
L	L	H	H	L	L	H	H	H	H	H	H	H	H	H	H	H	H	L	H	H	H
L	L	H	H	L	H	H	H	H	H	H	H	H	H	H	H	H	H	H	L	H	H
L	L	H	H	H	L	H	H	H	H	H	H	H	H	H	H	H	H	H	H	L	H
L	L	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	L
L	H	X	X	X	X	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H
H	L	X	X	X	X	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H
H	H	X	X	X	X	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H	H

H = high level, L = low level, X = irrelevant

Functional Block Diagram



154, 'L154 4-LINE-TO-16-LINE DECODER/DEMULTIPLEXER

NOTE: I_{CC} is measured with all inputs grounded and all outputs open.

† For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating conditions for the applicable type.
‡ All typical values are at V_{CC} = 5V, T_A = 25°C.
* Not more than one output should be shorted at a time.
* t_{PLH} = propagation delay time, low-to-high-level output
t_{PHL} = propagation delay time, high-to-low-level output

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

54164/74164 8-Parallel-Out Serial Shift Register

	Schottky TTL					High-Speed TTL					Low-Power Schottky TTL					Standard TTL					Low-Power TTL				
	Device Type	Package C	P	M	CF	Device Type	Package C	P	M	CF	Device Type	Package C	P	M	CF	Device Type	Package C	P	M	CF	Device Type	Package C	P	M	CF
T.I.											SN54LS164 SN74LS164	J J			W N	SN54164 SN74164	J J			W N	SN54L164 SN74L164	J J			
FAIRCHILD																									
MOTOROLA											SN74LS164		P			SN74164		P							
N.S.C.											DM74LS164 DM54LS164		N W			DM74164		N			DM54L164 DM74L164		J N		
PHILIPS											74LS164		N			74164		N							
SIGNETICS											74LS164		A			S54164 74164	J J	A A		W N					
SIEMENS																									
FUJITSU											74LS164		P												
HITACHI											HD74LS164		P			HD74164		P							
MITSUBISHI											M74LS164		P			M53164		P							
NEC																μPD2164	D								
TOSHIBA																TD3014		P							
AMD											Am74LS164														

Electrical Characteristics SN54LS164, SN74LS164

absolute maximum ratings over operating free-air temperature range

Supply voltage, V_{CC} . . . 7V
Input voltage . . . 7V

Operating free-air temperature range	SN54	−55°C to 125°C
	SN74	0°C to 70°C
Storage temperature range		−65°C to 150°C

recommended operating conditions

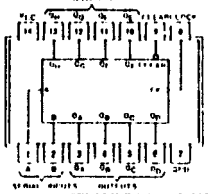
	SN54LS164 MIN	NOM	MAX	SN74LS164 MIN	NOM	MAX	UNIT
Supply voltage, V_{CC}	4.5	5	5.5	4.75	5	5.25	V
High-level output current, I_{OH}			−400			−400	μA
Low-level output current, I_{OL}			4			8	mA
Clock frequency, f_{clock}	0		25	0		25	MHz
Width of clock or clear input pulse, t_w	20			20			ns
Data setup time, t_{setup}	15			15			ns
Data hold time, t_{hold}	5			5			ns
Operating free-air temperature, T_A	−55		125	0		70	°C

electrical characteristics over recommended operating free-air temperature range

PARAMETER		TEST CONDITIONS†		MIN	TYP‡	MAX	UNIT
V_{IH}	High-level input voltage			2			V
V_{IL}	Low-level input voltage					0.8	V
V_I	Input clamp voltage	V_{CC} = MIN, I_I = −18mA				−1.5	V
V_{OH}	High-level output voltage	V_{CC} = MIN, V_{IH} = 2V, V_{IL} = 0.8V, I_{OH} = −400μA		2.7	3.5		V
V_{OL}	Low-level output voltage	V_{CC} = MIN, V_{IH} = 2V, V_{IL} = 0.8V, I_{OL} = 8mA			0.35	0.5	V
I_I	Input current maximum input voltage	V_{CC} = MAX, V_I = 7V				0.1	mA
I_{IH}	High-level input current	V_{CC} = MAX, V_I = 2.7V				20	μA
I_{IL}	Low-level input current	V_{CC} = MAX, V_I = 0.4V				−0.4	mA
I_{OS}	Short-circuit output current*	V_{CC} = MAX	SN54LS	−20		−100	mA
			SN74LS	−20		−100	
I_{CC}	Supply current	V_{CC} = MAX, See Note 1			16	27	mA
f_{max}	Maximum clock frequency	V_{CC} = 5V, T_A = 25°C, R_L = 2kΩ	C_L = 15pF	25	36		MHz
t_{PHL}	Propagation delay time, high-to-low level Q outputs from clear input		C_L = 15pF		24	36	ns
t_{PLH}	Propagation delay time, low-to-high level Q outputs from clock input		C_L = 15pF		17	27	ns
t_{PHL}	Propagation delay time, high-to-low level Q outputs from clock input		C_L = 15pF		21	32	ns

Pin Assignment (Top View)

①



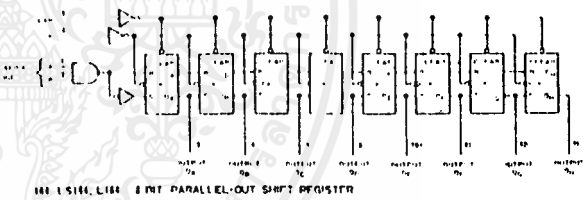
positive logic; see function table

Function Table

164, LS164, L164 (see Note 2)

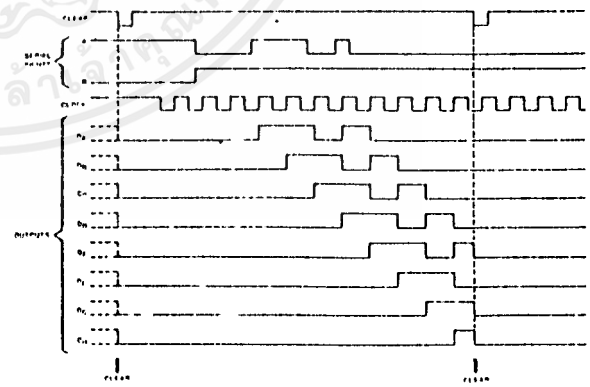
INPUTS CLEAR	CLOCK	A	B	OUTPUTS QA	QB	QH
L	X	X	X	L	L	L
H	L	X	X	Q_{A0}	Q_{B0}	Q_{H0}
H	↑	H	H	H	Q_{An}	Q_{Gn}
H	↑	L	X	L	Q_{An}	Q_{Gn}
H	↑	X	L	L	Q_{An}	Q_{Gn}

Functional Block Diagram



164, LS164, L164 8-BIT PARALLEL-OUT SHIFT REGISTER

typical clear, shift, and clear sequences



NOTES 1. I_{CC} is measured with outputs open, serial inputs grounded, and a momentary ground, then 4.5V, applied to clear.

2. H = high level (steady state), L = low level (steady state)
X = irrelevant (any input, including transitions)
↑ = transition from low to high level
Q_{A0}, Q_{B0}, Q_{H0} = the level of Q_A, Q_B, or Q_H, respectively, before the indicated steady-state input conditions were established
Q_{An}, Q_{Gn} = the level of Q_A or Q_G before the most recent ↑ transition of the clock; indicates a one-bit shift

†For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating conditions for the applicable device type.
‡All typical values are at V_{CC} = 5V, T_A = 25°C
*Not more than two outputs should be shorted at a time.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

54373/74373 Octal D-Type Transparent Latches and Edge-Triggered Flip-Flops

	Schottky TTL					High-Speed TTL					Low-Power Schottky TTL					Standard TTL					Low-Power TTL				
	Device Type	Package C	P	M	CF	Device Type	Package C	P	M	CF	Device Type	Package C	P	M	CF	Device Type	Package C	P	M	CF	Device Type	Package C	P	M	CF
T.I.	SN54S373 / SN74S373	J	N								SN54LS373 / SN74LS373	J	N												
FAIRCHILD																									
MOTOROLA																									
N.S.C.																									
PHILIPS																									
SIGNETICS																									
SIEMENS																									
FUJITSU																									
HITACHI																									
MITSUBISHI																									
NEC																									
TOSHIBA																									

Electrical Characteristics SN54LS373/SN74LS373

absolute maximum ratings over operating free-air temperature range

Supply voltage, V_{CC} 7V
Input voltage 7V
Operating free-air temperature range: SN54LS 55°C to 125°C; SN74LS 0°C to 70°C
Storage temperature range . . . . . −65°C to 150°C

recommend operating conditions

		SN54LS373 MIN	NOM	MAX	SN74LS373 MIN	NOM	MAX	UNIT
Supply voltage, V_{CC}		4.5	5	5.5	4.75	5	5.75	V
High-level output current, I_{OH}				−1			−2.6	mA
High-level output voltage, V_{OH}				5.5			5.5	V
Pulse width, t_w	Clock enable high	15			15			ns
	Clock enable low	15			15			
Setup time t_{su} (1)		0↓			0↓			ns
Hold time t_{hold}		10↓			10↓			ns
Operating free-air temperature T_A		−55		125	0		70	°C

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER		TEST CONDITIONS‡		MIN	TYP‡	MAX	UNIT
V_{IH}	High-level input voltage			2			V
V_{IL}	Low-level input voltage					0.8	V
V_{IK}	Input clamp voltage	V_{CC} = MIN, I_I = −18mA				−1.5	V
V_{OH}	High-level output voltage	V_{CC} = MIN, V_{IH} = 2V, V_{IL} = $V_{IL\,max}$, I_{OH} = MAX		2.4	3.1		V
V_{OL}	Low-level output voltage	V_{CC} = MIN, V_{IH} = 2V, V_{IL} = $V_{IL\,max}$	I_{OL} = 24mA		0.35	0.5	V
I_{OZH}	Off-state output current, high-level voltage applied	V_{CC} = MAX, V_{IH} = 2V, V_O = 2.7V				20	μA
I_{OZL}	Off-state output current, low-level voltage applied	V_{CC} = MAX, V_{IH} = 2V, V_O = 0.4V				−20	μA
I_I	Input current at maximum input voltage	V_{CC} = MAX, V_I = 7V				0.1	mA
I_{IH}	High-level input current	V_{CC} = MAX, V_I = 2.7V				20	μA
I_{IL}	Low-level input current	V_{CC} = MAX, V_I = 0.4V				−0.4	mA
I_{OS}	Short-circuit output current§	V_{CC} = MAX		−30		−130	mA
I_{CC}	Supply current	V_{CC} = MAX, Output control at 4.5V	'LS373		24	40	mA

switching characteristics, V_{CC} = 5V, T_A = 25°C

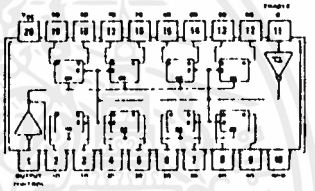
PARAMETER	FROM (INPUT)	TO (OUTPUT)	TEST CONDITIONS	MIN	TYP	MAX	UNIT
f_{max}							MHz
t_{PLH}	Data	Any Q	C_L = 45pF, R_L = 667Ω, See Notes 2 and 3		12	18	ns
t_{PHL}					12	18	
t_{PLH}	Clock or enable	Any Q			20	30	ns
t_{PHL}					18	30	
t_{PZH}	Output Control	Any Q			15	28	ns
t_{PZL}					25	36	
t_{PHZ}	Output Control	Any Q	C_L = 5pF, R_L = 667Ω, See Note 3		12	20	ns
t_{PLZ}					15	25	

† For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating conditions.
‡ All typical values are at V_{CC} = 5V, T_A = 25°C.
§ Not more than one output should be shorted at a time and duration of the short-circuit should not exceed one second.

NOTES 2. Maximum clock frequency is tested with all outputs loaded.
3. See load circuits and waveforms on page 3-11.

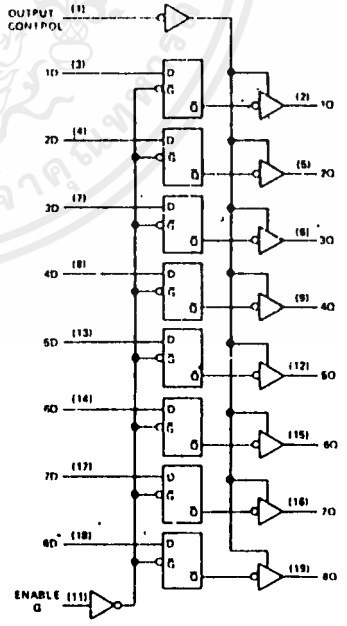
f_{max} = maximum clock frequency
t_{PLH} = propagation delay time, low-to-high-level output
t_{PHL} = propagation delay time, high-to-low-level output
t_{PZH} = output enable time to high level
t_{PZL} = output enable time to low level
t_{PHZ} = output disable time from high level
t_{PLZ} = output disable time from low level

Pin Assignments (Top View)



SN54LS373 (J) SN74LS373 (J, N)
SN54S373 (J) SN74S373 (J, N)

'LS373, 'S373
TRANSPARENT LATCHES



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้