



ปีการศึกษา 2534

Neumerical Control

โดย

นายสมเกียรติ คล่องพานิชภักดิ์ 311294

นายสมชัย สุขเจริญชัยกุล 311296

นายสมบูรณ์ วงศ์ศิริอำนาจ 311304

อาจารย์ที่ปรึกษา

อ. สุรพันธ์ เอื้อไพบุลย์

อ. ยทธนา คัดใจเดียว

008500

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโท
ภาควิชา
คณะวิศวกรรมศาสตร์
เรื่อง
ผู้จัดทำ

ปีการศึกษา 2534
วิศวกรรมอิเล็กทรอนิกส์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
NEUMERICAL CONTROL
นายสมเกียรติ คล่องพานิชภักดิ์ 311294
นายสมชัย สุขเจริญชัยกุล 311296
นายสมบูรณ์ วงศ์ศิริอำนวย 311304



..... (อาจารย์ที่ปรึกษา)
(อ. สรพันธ์ เอื้อไพบูลย์)

..... (อาจารย์ที่ปรึกษา)
(อ. ยุกธนา คัดใจเดียว)

Neumerical Control

นายสมเกียรติ คล่องพาทิชภักดิ์ 311294
นายสมชัย สุขเจริญชัยกุล 311296
นายสมบูรณ์ วงศ์ศิริอานวย 311304
อ.สุรพันธุ์ เอื้อาพบูลย์ อาจารย์ที่ปรึกษา
อ.ยุทธนา คีตใจเดี้ยว อาจารย์ที่ปรึกษา
ปีการศึกษา 2534

บทคัดย่อ

โครงงานนี้เป็นการศึกษาเกี่ยวกับเครื่อง Neumerical Control และออกแบบส่วนที่เข้า Interface กับ ไมโครคอมพิวเตอร์ 8088 ซึ่งจะต่อไปยังวงจรเพื่อเข้าขับ Stepping Motor ซึ่งจะต่อไปยังเครื่องเจาะ โดยมีการควบคุมในด้าน 2 ทิศทางคือ X และ Y ซึ่งการเคลื่อนที่จะใช้ Stepping ดังนั้นในโครงงานนี้จึงเน้นหนักทางด้าน การควบคุม Stepping Motor ให้เคลื่อนที่ได้ตามระยะทางที่ต้องการ โดยจะใช้ Stepping Motor 4 เฟส 8 Volt/เฟส 0.69 Amp/เฟส 1.8 degree/Step (แบบ Ground ร่วม) 2 ตัว และใช้การกระตุ้นแบบทีละ 2 เฟส ซึ่งการดำเนินงานจะ ใช้ร่วมกับ Software ที่เขียนขึ้น โดยการควบคุมจะทำให้โดยการส่งข้อมูลตามที่ต้องการ ผ่านทาง Software เพื่อให้ไมโครคอมพิวเตอร์ 8088 เป็นตัวส่งงานออกมาภายนอกเพื่อ ไปโปรแกรม 8255 เพื่อให้ได้สัญญาณทิศทางการหมุน และโปรแกรม 8253÷5 ซึ่งเป็นตัว Counter เพื่อเพื่อจ่าย Pulse ให้แก่วงจร Drive Stepping Motor ทั้ง 2 ตัวให้ เคลื่อนที่ตามระยะที่ต้องการ

โดยรายละเอียดการทำงานของส่วนต่างวทั้ง Hardware และ Software รวมจนถึงการดำเนินงานจะนำเสนออยู่ภายในปริญาานิพนธ์ฉบับนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Neumerical Control

Somkiat Klongpanichapuk 311294

Somchai Sukchareonchaikul 311296

Somboon Wongsiriamnuey 311304

Surapan Aeurpaibul Advisor

Yuttana Kidjaideaw Advisor

1991

Abstract

This project is concerned the study about Neumerical Control Equipment and is designed to interface with Micro-computer 8088 which will co-circuit in order to drive stepping motor which will co-ordinate with drilling machine, by have a control system into 2 directions; X and Y which the motion will use stepping. Therefore, this project is concentrated on control aspect of stepping motor to move to distance as required by using stepping motor 4 phases 8 volt/phase 0.69 amp/phase 1.8 degree/step (with co-ground system) 2 units and have motivation with each 2 phases which the utilization will co-with written software. The control can be done by transmitting desired data through software in order to have Micro-computer 8088 work as instruction unit to outsider to program 8255 in order to have circulate direction signal and program 8253-5 which is the counter to generate pulse for 2 drive stepping motor circuit to move to required distance.

This thesis will have all the detail of working function and utilization for both hardware and software.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นับเป็นเอกสารที่เผยแพร่
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงชื่อของเอกสารผู้จัดทำที่มีการนำไปใช้

สารบัญ

| | หน้าที่ |
|--|---------|
| บทนำ | 1 |
| บทที่ 1 ทฤษฎีและหลักการทำงานของ Stepping Motor | 2 |
| บทที่ 2 การ Interface กับ Computer (8088) | 15 |
| บทที่ 3 8253-5 | 21 |
| บทที่ 4 วงจรที่ใช้ในโครงงาน | 32 |
| บทที่ 5 Software | 44 |
| บทที่ 6 การทดลองและผลการทดลอง | 46 |
| บทที่ 7 บทสรุปและวิจารณ์ | 48 |
| ภาคผนวก ก Software ที่ใช้ในโครงงาน | 49 |
| ภาคผนวก ข | 54 |
| กิตติกรรมประกาศ | 81 |
| บรรณานุกรม | 82 |

สารบัญรูปภาพ

| | หน้า |
|--|------|
| บทที่ 1 | |
| รูปที่ 1.1 ภาพหน้าตัดของ Stepping Motor แบบ 4 เฟส | 2 |
| รูปที่ 1.2 Stepping Motor 4 เฟสแบบ Ground ร่วม | 3 |
| รูปที่ 1.3 โครงสร้างของ Variable-reluctance motors | 4 |
| รูปที่ 1.4 โครงสร้างของ Claw-pole permanent-magnet motors | 5 |
| รูปที่ 1.5 โครงสร้างของ Hybrid Motors | 6 |
| รูปที่ 1.6 แสดงขดลวดจากการพันชนิดต่างๆ | 7 |
| รูปที่ 1.7 เส้นแรงแม่เหล็กที่ทำให้เกิดแรงบิด | 7 |
| รูปที่ 1.8 แสดงการทำงานของ Stepping Motor | 8 |
| รูปที่ 1.9 ระบบควบคุม Stepping Motor | 8 |
| รูปที่ 1.10 แสดงการเคลื่อนที่ทีละ Step เมื่อกระตุ้นเฟส 1 เฟส 3 | 9 |
| รูปที่ 1.11 แสดงการกระตุ้นแบบเดินหน้าและถอยหลัง | 9 |
| รูปที่ 1.12 แสดงการกระตุ้นแบบเฟสเดียว | 10 |
| รูปที่ 1.13 แสดงการกระตุ้นแบบ 2 เฟส | 10 |
| รูปที่ 1.14 แสดงการกระตุ้นแบบครึ่ง Step | 10 |
| รูปที่ 1.15 เส้นแรงแม่เหล็กเมื่อขับแบบ 2 เฟส | 11 |
| รูปที่ 1.16 ลักษณะเส้นแรงแม่เหล็กเมื่อขับแบบ 2 เฟส | 11 |
| รูปที่ 1.17 Block diagram control stepping motor | 13 |
| รูปที่ 1.18 แสดงความสัมพันธ์ระหว่าง Torque กับกระแสและ Speed | 14 |
| บทที่ 2 | |
| รูปที่ 2.1 แสดงการจัดขา IC 8255 และแผนผังวงจรภายใน | 17 |
| รูปที่ 2.2 ความหมายของบิตต่างๆของรหัสควบคุม | 19 |

บทที่ 3

| | | |
|------------|--|----|
| รูปที่ 3.1 | ไดอะแกรมเวลาการทำงานของ 8253-5 | 24 |
| รูปที่ 3.2 | ไดอะแกรมเวลาการทำงานใน Mode 1 ของ 8253-5 | 25 |
| รูปที่ 3.3 | ไดอะแกรมเวลาการทำงานใน Mode 2 ของ 8253-5 | 26 |
| รูปที่ 3.4 | ไดอะแกรมเวลาการทำงานใน Mode 3 ของ 8253-5 | 28 |
| รูปที่ 3.5 | ไดอะแกรมเวลาการทำงานใน Mode 4 ของ 8253-5 | 28 |
| รูปที่ 3.6 | ไดอะแกรมเวลาการทำงานใน Mode 5 ของ 8253-5 | 29 |

บทที่ 4

| | | |
|------------|---|----|
| รูปที่ 4.1 | Block diagram ของการทำงานของวงจร | 32 |
| รูปที่ 4.2 | ส่วน Interface | 33 |
| รูปที่ 4.3 | ส่วน Input controller | 36 |
| รูปที่ 4.4 | แสดงสัญญาณควบคุมจำนวน Pulse หรือ Step | 37 |
| รูปที่ 4.5 | แสดงสัญญาณควบคุมจำนวนความถี่ | 37 |
| รูปที่ 4.6 | ส่วน Logic Sequencer | 39 |
| รูปที่ 4.7 | ส่วน Driver | 41 |
| รูปที่ 4.8 | ส่วน Supply ของ Driver | 42 |
| รูปที่ 4.9 | วงจรรวม | 43 |

บทที่ 6

| | | |
|------------|--|----|
| รูปที่ 6.1 | แสดงการเปรียบเทียบระหว่างความถี่ที่ป้อนกับความถี่จริงกับแรงดันไฟ ... | 47 |
|------------|--|----|

สารบัญตาราง

| | หน้าที่ |
|---|---------|
| บทที่ 2 | |
| ตาราง 2.1 แสดงข้อความควบคุมของ 8255 | 18 |
| ตาราง 2.2 แสดงสัญญาณควบคุมของ Port C | 19 |
| ตาราง 2.3 แสดงสัญญาณตรวจสอบของ Port C | 20 |
| บทที่ 3 | |
| ตาราง 3.1 แสดงหน้าที่ของบิตต่างๆใน Register Mode Control | 22 |
| ตาราง 3.2 แสดงข้อมูลของบิต SC1 และ SC2 | 22 |
| ตาราง 3.3 แสดงหน้าที่ของข้อมูลในบิต D4 และ D5 | 23 |
| ตาราง 3.4 แสดงหน้าที่ของข้อมูลในบิต D3-D1 | 23 |
| ตาราง 3.5 หน้าที่ของบิต A0 และ A1 ในการโปรแกรม Register Counter . | 30 |
| ตาราง 3.6 แสดงหน้าที่ของขา Input Gate ของ 8253 | 31 |
| บทที่ 4 | |
| ตาราง 4.1 แสดงค่าของ PO-P7 ที่ Set ในวงจร | 34 |
| ตาราง 4.2 แสดงตำแหน่ง Address ที่ใช้ควบคุม Port ต่างๆ | 34 |
| ตาราง 4.3 แสดงตำแหน่ง Address ที่ใช้ในวงจร | 35 |
| ตาราง 4.4 แสดงหน้าที่โดยสรุปของ Address ต่างๆ | 35 |
| ตาราง 4.5 แสดงหน้าที่บิตต่างๆของ Port A 8255 ในวงจร | 38 |
| บทที่ 6 | |
| ตาราง 6.1 แสดงการเปรียบเทียบระหว่างความถี่ที่ป้อนกับที่ออกมาและแรงดันไฟ . | 46 |

บทนำ

ในปัจจุบันนี้ระบบคอมพิวเตอร์ได้เข้ามามีบทบาท และใช้งานกันอย่างแพร่หลายในด้านธุรกิจและอุตสาหกรรมอย่างมากมาย ซึ่งนับวันระบบคอมพิวเตอร์จะมีราคาถูกลงแต่มีขีดความสามารถมากขึ้นเรื่อยๆ จึงมีการนำมาประยุกต์ใช้กับเครื่องมือต่างๆ อย่างมากมาย ซึ่งจะทำให้สะดวกและรวดเร็วขึ้นมาก ซึ่งในทางอุตสาหกรรม จะต้องมีการเจาะอุปกรณ์ต่างๆ มากมาย จึงมีการนำระบบคอมพิวเตอร์มาประยุกต์ใช้กับเครื่องเจาะ โดยควบคุมตำแหน่งการเจาะด้วยคอมพิวเตอร์ ซึ่งเครื่องดังกล่าวคือเครื่อง Numerical Control แต่ในปัจจุบันนี้เครื่องดังกล่าวยังมีราคาสูงมาก ซึ่งในโครงการนี้ได้ทำการศึกษาและออกแบบอุปกรณ์แล้ว Interface กับคอมพิวเตอร์ 8088 ซึ่งจะใช้คู่กับ Software ที่เขียนขึ้น โดยต่อไปยังวงจร Logic Sequencer เพื่อส่งข้อมูลไปยังวงจร Drive Stepping Motor เพื่อควบคุมให้วัตถุเคลื่อนที่ไปตามตำแหน่งที่ต้องการ

คณะผู้จัดทำ

บทที่ 1

ทฤษฎีและหลักการทำงานของ Stepping Motor

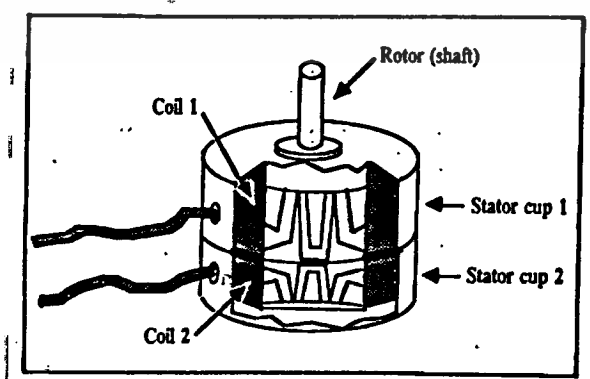
1.1 นิยาม

เป็น Motor ที่ตัวเพลลา (Rotor) จะหมุนเป็น Step โดยที่ Step การหมุนจะขึ้นอยู่กับการป้อน Pulse ที่ป้อนให้กับวงจรจัดลำดับที่จะนำสัญญาณไปทำให้เกิดการจ่ายไฟให้ขดลวดแต่ละเฟสโดยวงจร Drive อย่างถูกต้อง ทำให้เกิดการหมุนอย่างต่อเนื่อง และสามารถควบคุมตำแหน่งการหมุนได้จากจำนวน Pulse ที่ป้อนเข้าไป

Stepping Motor มีข้อได้เปรียบมอเตอร์ไฟตรงธรรมดา คือ Stepping Motor มีการควบคุมแบบ Loop เปิด ทำให้ง่ายต่อการควบคุม และเรายังสามารถที่จะรู้ตำแหน่งของมอเตอร์ได้ตลอดเวลาอย่างแน่นอนแม่นยำ ดังนั้นจึงเหมาะกับการใช้งานที่ต้องการควบคุมตำแหน่งที่แน่นอน

1.2 โครงสร้างของ Stepping Motor

ภายใน Stepping Motor ประกอบด้วย สเตเตอร์, โรเตอร์, และขดลวด ประกอบเข้าด้วยกันดังรูปที่ 1.1 ที่เป็นแบบ 4 เฟส (เสมือนการนำเอามอเตอร์มาประกบกัน 2 ตัว) ซึ่งขึ้นที่นิยมใช้กันมากที่สุด

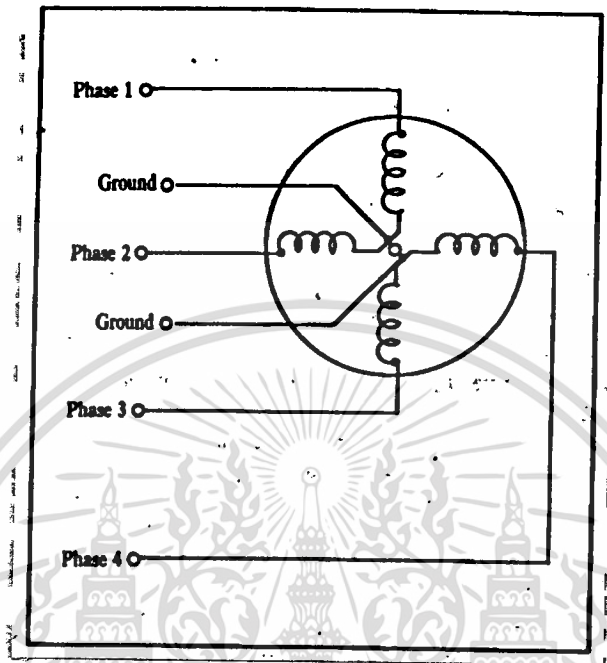


รูปที่ 1.1 ภาพหน้าตัดของ Stepping Motor แบบ 4 เฟส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า Stepping Motor แบบ 4 เฟสอาจมีสายไฟ 8 เส้นถ้าหากไม่มีการต่อแบบ

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารนี้ทุกประการ

ground ร่วม แต่อาจเหลือสายไฟเพียง 5 หรือ 6 เส้นเท่านั้นถ้าเป็นการต่อแบบ ground ร่วมดังรูปที่ 1.2

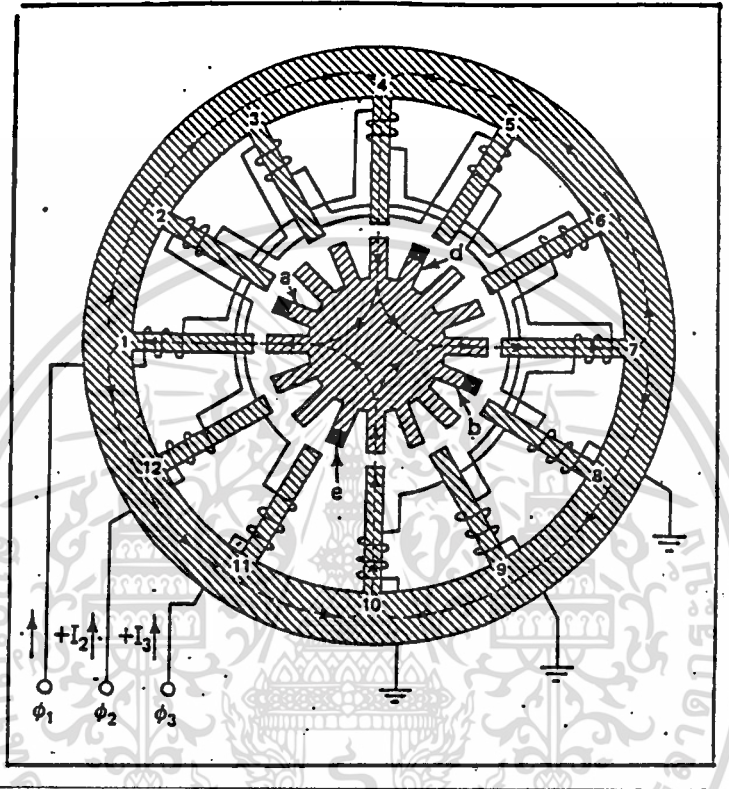


รูปที่ 1.2 Stepping Motor 4 เฟสแบบ ground ร่วม

1.3 ชนิดของ Stepping Motor

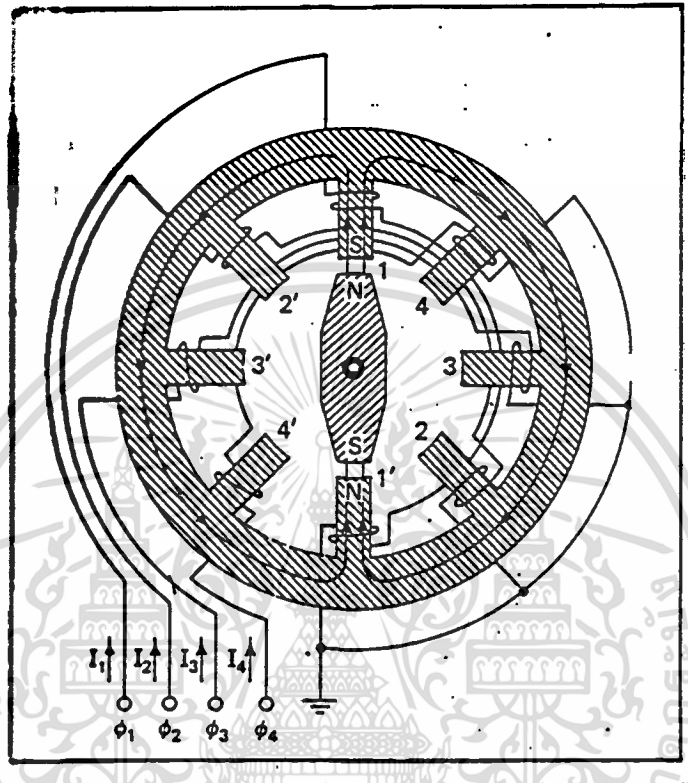
Stepping Motor แบ่งออกได้เป็น 3 ชนิดใหญ่ๆคือ

1. Variable-reluctance motors ตัวอย่างโครงสร้างของ Stepping Motor แบบนี้แสดงดังรูปที่ 1.3 โดยในรูปมี 3 เฟส แต่ละเฟสใช้ขดลวดพันบน 4 ขั้วหรือซี่ฟันของ Stator ขั้วของ Stator ที่อยู่ตรงกันข้ามจะพันด้วยขดลวดที่ลักษณะที่ต่างกันไป เพื่อให้มีความสมดุลระหว่างเส้นแรงแม่เหล็กเข้าและออกจาก Rotor ไม่เป็นที่นิยมใช้ โดยทั้ง Stator และ Rotor ทำจากโลหะที่มีค่า Permeability สูง กำลังถูกแทนที่ด้วย Servo Motor



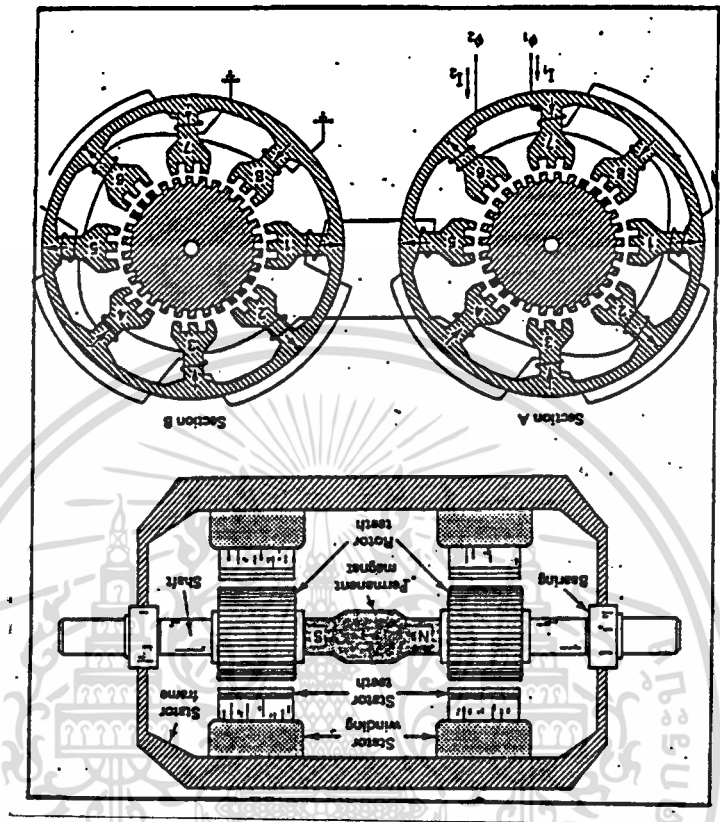
รูปที่ 1.3 โครงสร้างของ Variable-reluctance motors

2. Claw-pole permanent-magnet motors โครงสร้างของ Stepping Motor แบบนี้แสดงดังรูปที่ 1.4 ซึ่งในรูปเป็น Stepping Motor แบบ 4 เฟส แต่ละเฟสมีขดลวดอยู่บน 2 ขั้วของ Stator ดังนั้นในการออกแบบนี้ Stator จะต้อง มี 8 ขั้ว Rotor ทำจากแม่เหล็กถาวร มีราคาค่อนข้างถูก



รูปที่ 1.4 โครงสร้างของ Claw-pole permanent-magnet motors

3. Hybridmotors โครงสร้างของไฮบริดสเตปมอเตอร์ (HSM) แสดงดังรูปที่ 1.5 โดยแสดงถึงโครงสร้างของ HSM ประกอบด้วยซี่ฟันของ Rotor และ Pole ของ Stator ที่มีซี่ฟันเช่นกัน และฟันด้วยขดลวด โดย Stepping Motor แบบนี้เป็น การนำเอา 2 แบบแรกมาผสมกัน ใช้แม่เหล็กถาวรทำ Rotor ส่วน Stator คล้ายกับแบบ 1 แต่ต่างกันที่การพันขดลวดที่สามารถให้ขั้วแม่เหล็กได้ 2 ขั้ว



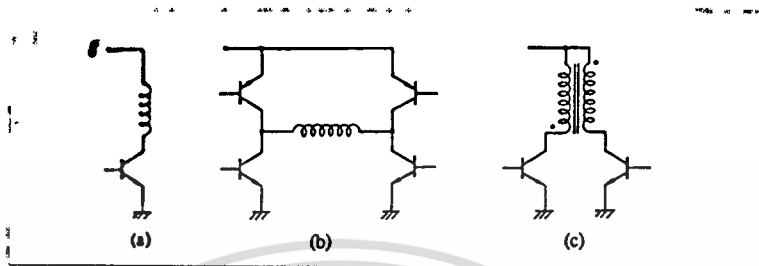
รูปที่ 1.5 โครงสร้างของ Hybrid Motors

1.4 การผันขดลวด

การผันขดลวดต่างกันทำให้คุณสมบัติของ Stepping Motor ต่างกันด้วย โดยเราสามารถแบ่งออกได้เป็น 3 ชนิด คือ

1. Monofilar winding, Unipolar Excitation แต่ละขั้วสามารถสร้างขั้วแม่เหล็กได้เพียงขั้วเดียว อาจเป็นเหนือหรือใต้ ใช้สำหรับ VR Motor
2. Monofilar winding, Bipolar Excitation สามารถสร้างขั้วแม่เหล็กได้ 2 ขั้ว ใช้สำหรับแบบ PM และ Hybrid
3. Bifilar winding, Bipolar Excitation เหมือนแบบ 2 แต่สามารถให้ขั้วแม่เหล็กได้ 2 ขั้วพร้อมกัน

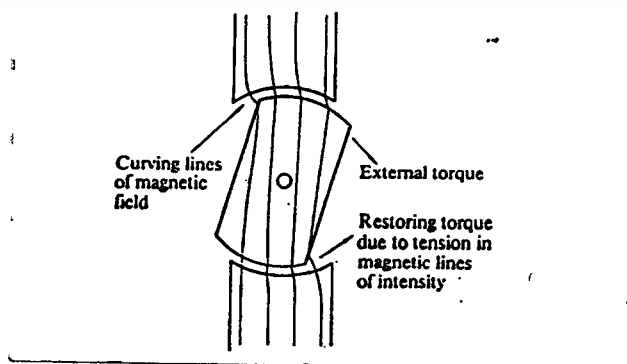
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ทั้ง 3 แบบ ต้องใช้วงจร drive ต่างกัน ดังรูปที่ 1.6
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหา และต้องอยู่ภายใต้เงื่อนไขของเอกสารทุกครั้งที่มีการนำไปใช้



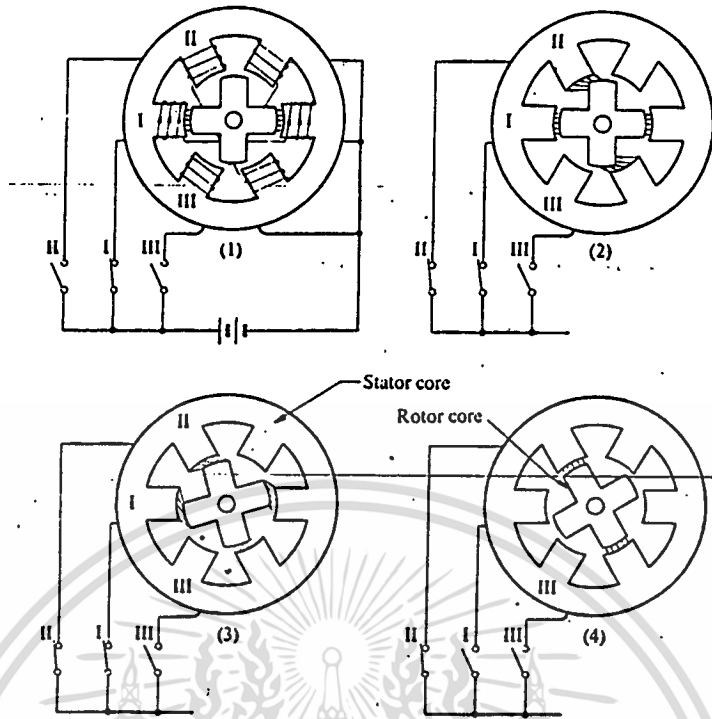
รูปที่ 1.6 (a) monofilar and unipolar, (b) monofilar and bipolar
(c) bifilar and bipolar

1.5 ทฤษฎีของ Stepping Motor

เนื่องจาก Stepping Motor นี้ Rotor เป็นเหล็กอ่อน ซึ่งมีคุณสมบัติพยายามปรับตัวเองให้อยู่ในแนวที่เส้นแรงแม่เหล็กผ่านมากที่สุด ดังรูปที่ 1.7 เมื่อเกิดเส้นแรงแม่เหล็กขึ้นที่ Stator ตัดผ่าน Rotor ตัว Rotor ก็พยายามปรับตัวเองให้เส้นแรงแม่เหล็กตัดผ่านตัวเองมากที่สุด โดยการหมุนตัวเอง ทำให้เกิดมุมของการหมุนขึ้น และมอเตอร์จะหยุดหมุน เมื่อเส้นแรงแม่เหล็กที่ตัดผ่านตัวมันถึงจุดที่มากที่สุด ดังรูปที่ 1.8

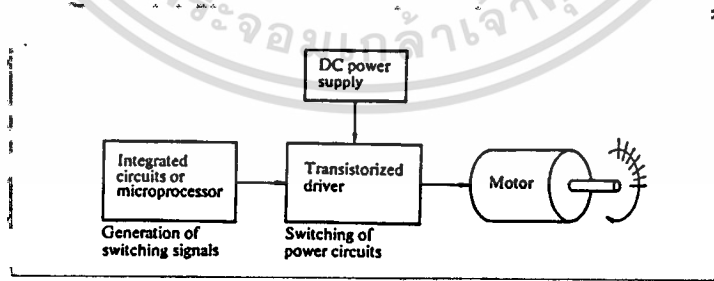


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 1.7 เส้นแรงแม่เหล็กที่ทำให้เกิดแรงบิด
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงแหล่งที่มาของเอกสารทุกครั้งที่มีการนำไปใช้



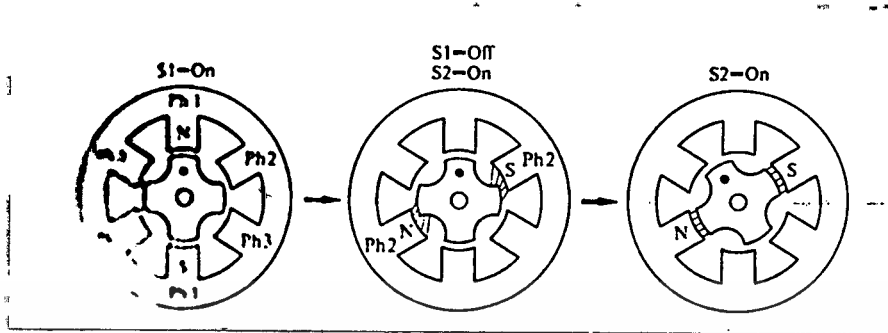
รูปที่ 1.8 แสดงการทำงานของ Stepping Motor

Stepping Motor เป็น Electric Motor ชนิดหนึ่งที่มีการควบคุมการทำงานของมันจะต้องมีใช้ทั้ง Analog Signal (DC Supply) เพื่อเป็นตัวจ่ายไฟให้แก่เฟสต่างๆ และ Digital Signal ที่ใช้ในการควบคุมการหมุน โดยระบบควบคุมของ Stepping Motor เป็นไปดังรูปที่ 1.9



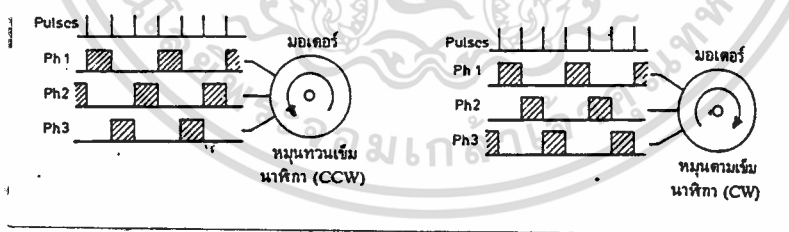
รูปที่ 1.9 ระบบควบคุม Stepping Motor

กล่าวคือใช้ DC Supply เป็นตัวจ่ายไฟเลี้ยงให้แก่ Stator ของเฟสต่างๆ และ Stepping Motor จะถูกควบคุมด้วย Digital Signal กล่าวคือ การหมุนของ Rotor จะหมุนเป็น Step ถ้าเราจ่าย Pulse ให้ 1 ลูก Rotor ก็จะหมุนไป 1 Step แต่เราจะต้องให้ Pulse ลูกต่อไปในเฟสต่อไปเพื่อให้เกิดการหมุนอย่างต่อเนื่องตามรูปที่ 1.10



รูปที่ 1.10 แสดงการเคลื่อนที่ทีละ Step เมื่อกระตุ้นเฟส 1 เฟส 2

และถ้าหากเราต้องการให้หมุนในทิศทางที่ตรงกันข้ามเราก็สามารถทำได้โดยการให้ Pulse กลับเฟสดังรูปที่ 1.11 แต่ถ้าต้องการหยุดก็หยุดป้อน Pulse มอเตอร์ก็จะหยุด ณ. ตำแหน่งสุดท้ายของการป้อน Pulse ครั้งสุดท้าย ทำให้เราสามารถรู้ตำแหน่งที่แน่นอนของ Stepping Motor ได้โดยการนับจำนวน Pulse ที่ป้อนให้มอเตอร์โดยใช้สูตร

$$\text{มุมที่เปลี่ยนไป} = \text{ค่ามุมต่อ Step} \times \text{จำนวน Pulse ที่ป้อนให้}$$


รูปที่ 1.11 แสดงการกระตุ้นแบบเดินหน้า และถอยหลัง :

1.6 การกระตุ้นเฟสขดลวด Stator

การกระตุ้นเฟสของ Stator มีอยู่ 3 แบบ คือ

1. การกระตุ้นเฟสเดียว เรียกว่า Single Phase Excitation ดังรูปที่ 1.12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของลิขสิทธิ์ที่มีการนำไปใช้

| สัญญาณนาฬิกาที่ | R | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|-----------------|---|---|---|---|---|---|---|---|---|
| เฟสที่ 1 | ■ | | | ■ | | | ■ | | |
| เฟสที่ 2 | | ■ | | | ■ | | | ■ | |
| เฟสที่ 3 | | | ■ | | | ■ | | | ■ |

รูปที่ 1.12 แสดงการกระตุ้นแบบเฟสเดียว

2. การกระตุ้นสองเฟส เรียกว่าแบบ Two Phase Excitation ดังรูปที่ 1.13

| สัญญาณนาฬิกาที่ | R | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|-----------------|---|---|---|---|---|---|---|---|---|
| เฟสที่ 1 | ■ | | ■ | | | ■ | | | |
| เฟสที่ 2 | | | | ■ | | | ■ | | |
| เฟสที่ 3 | | ■ | | | ■ | | | ■ | |

รูปที่ 1.13 แสดงการกระตุ้นแบบสองเฟส

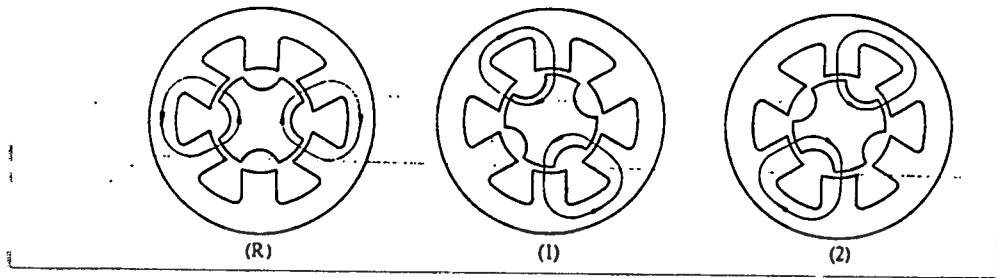
3. การกระตุ้นโดยใช้เฟส 1 และ เฟส 2 สลับกันไป เรียกว่าแบบ One-Two-Phase Excitation หรือแบบ Half Step Operation ดังรูปที่ 1.14

| สัญญาณนาฬิกาที่ | R | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|-----------------|---|---|---|---|---|---|---|---|---|
| เฟสที่ 1 | ■ | | | | | | ■ | ■ | ■ |
| เฟสที่ 2 | | ■ | | | | | | | ■ |
| เฟสที่ 3 | | | | ■ | | | | | |

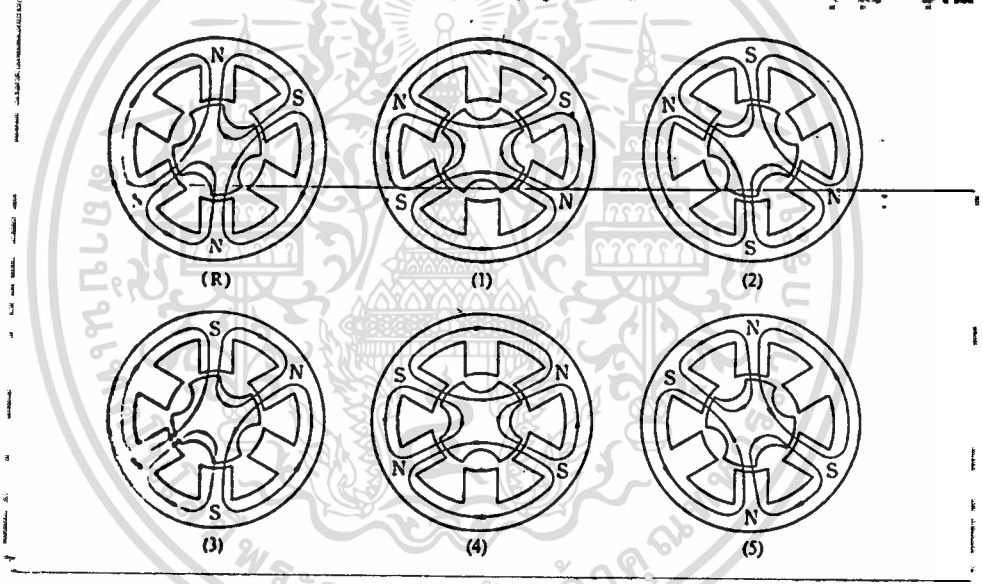
รูปที่ 1.14 แสดงการกระตุ้นแบบครึ่ง Step

ในการขับแบบกระตุ้น 2 เฟส เส้นแรงแม่เหล็กจะไม่ผ่านแกนเหล็กเป็นเส้นตรงเลยทีเดียวเหมือนแบบกระตุ้นเฟสเดียว แต่จะวกกลับมาเข้าสู่แกนทางด้านข้างๆดังรูปที่ 1.15 และเส้นแรงแม่เหล็กส่วนหนึ่งจะมาจากแกนตรงข้ามดังรูปที่ 1.16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1.15 เส้นแรงแม่เหล็กเมื่อขับแบบ 2 เฟส



รูปที่ 1.16 ลักษณะเส้นแรงแม่เหล็กเมื่อขับแบบ 2 เฟส

การขับ Stepping Motor แบบกระตุ้น 2 เฟสนี้จะมีลักษณะเดียวกันกับการขับแบบกระตุ้นเฟสเดียว แต่ในการกระตุ้นแต่ละครั้งนั้นจะกระตุ้นที่เดียวพร้อมกันทั้ง 2 เฟส ซึ่งจะเข้าตำแหน่งแต่ละ step ได้เร็วกว่าแบบเฟสเดียว และแรงบิดมากกว่าการขับแบบเฟสเดียวด้วย

1.7 การเกิด Oscillator ของ Stepping Motor

ความเร็วของการหมุนของ Stepping Motor เป็นลิเนียร์กับความถี่ที่ป้อนให้ Stepping Motor แต่เมื่อเราป้อนความถี่เพิ่มขึ้นเรื่อยๆจนถึงความถี่ค่าหนึ่ง Stepping Motor ก็จะหยุดหมุน เนื่องจากที่ Rotor หมุนตามฟลักซ์แม่เหล็กไม่ทัน เราเรียกว่า

มอเตอร์เกิดการ Oscillate ซึ่งมอเตอร์แต่ละตัวจะเกิดการ Oscillate ที่ความถี่ต่างกันไป โดยทั่วไปจะ Oscillate ที่ความถี่ประมาณ 500 Hz ในการขับแบบกระตุ้น 1 หรือ 2 เฟส และจะมีความถี่ประมาณ 1 kHz เมื่อขับแบบครึ่ง Step

แต่เมื่อลดความถี่ให้ต่ำลงมอเตอร์จะไม่หมุนทันที และจะเริ่มหมุนอีกที่ที่ความถี่ประมาณ 200 Hz ในการขับแบบ 1 หรือ 2 เฟส และมีความถี่ประมาณ 150 Hz เมื่อขับแบบครึ่ง Step

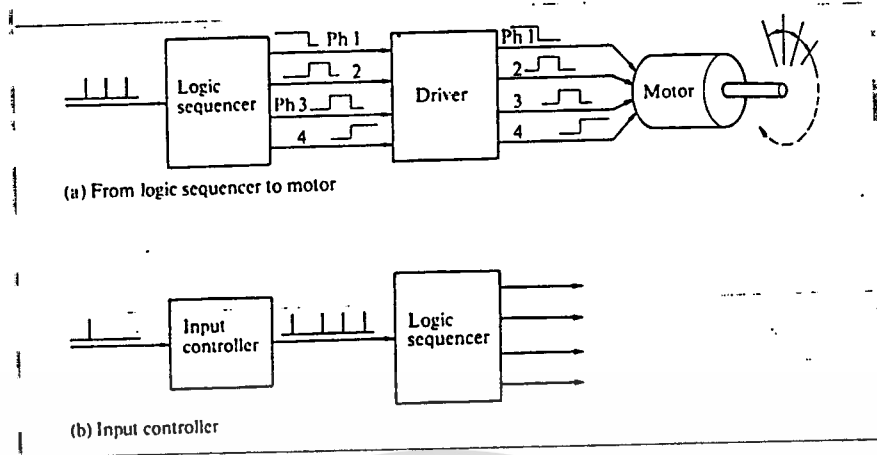
1.8 วิธีการควบคุม Stepping Motor

การควบคุม Stepping Motor โดยทั่วไปจะมีอยู่ 2 วิธี

1. ใช้ Computer หรือ Single board ซึ่งสามารถควบคุมการทำงานของ Speed และจำนวน Pulse ได้อย่างถูกต้อง แต่ข้อเสียคือค่าใช้จ่ายสูงและต้องใช้ความรู้ทาง Hardware เพื่อใช้ในการ Interface นอกจากนี้ต้องใช้ความรู้ทาง Software ในการเขียนโปรแกรมควบคุม ซึ่งถ้าไม่ต้องการทำส่วน Interface Card เองก็สามารถหาซื้อได้แต่ราคาค่อนข้างแพง
2. ใช้วงจร Digital ทั่วไปในการควบคุม ซึ่งจะมีราคาถูกกว่ามากและไม่ต้องใช้ความรู้ทางด้าน Computer มาก แต่การควบคุมยังไม่แม่นยำพอจึงไม่เหมาะกับการนำมาใช้งานจริง เหมาะกับการศึกษามากกว่า

1.9 การควบคุม Stepping Motor

การที่จะให้ Stepping Motor ทำงานตามที่เรต้องการนั้นเราต้องมีการป้อน Pulse rate ตามความเร็วที่ต้องการ และต้องมีการป้อนสัญญาณทิศทางว่าจะทวนเข็มหรือตามเข็ม โดยที่เราสามารถเขียนเป็นรูปการควบคุมอย่างง่ายดังรูปที่ 1.17



รูปที่ 1.17 Block diagram control Stepping Motor

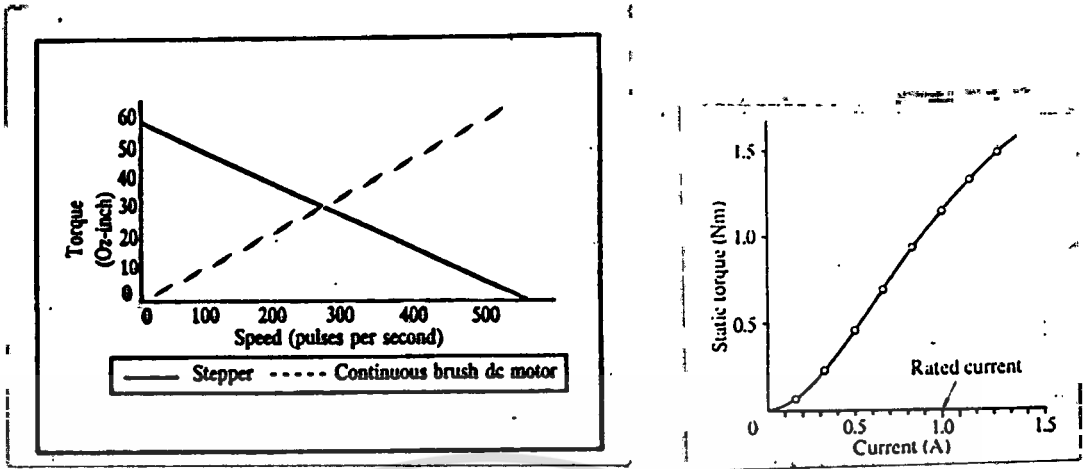
โดยที่หน้าที่ต่างๆของแต่ละ Block มีดังนี้

-Input Controller ทำหน้าที่ให้กำเนิดสัญญาณ Pulse rate ตามความถี่ที่ต้องการโดยอาจใช้ IC Timer 555 หรือใช้ Clock จาก Computer แต่ต้องมีวงจรหารความถี่เพื่อให้ Stepping Motor ทำงานได้โดยไม่เกิดการ Oscillate และหมุนตามความถี่ที่ต้องการ นอกจากนี้ยังทำหน้าที่ในการให้สัญญาณ Direction เพื่อให้ได้ทิศทางหมุนตามต้องการ โดยอาจใช้การสับสวิทช์หรือสัญญาณจาก Computer ก็ได้เช่นกัน

-Logic Sequencer เป็นวงจรทาง Logic โดยทั่วไปใช้ Logic gate ต่างๆ และ Flip-flop มีหน้าที่ควบคุมให้เกิดการกระตุ้นเฟสต่างๆของ Stepping Motor ตามที่ต้องการว่าจะให้เกิดการกระตุ้นแบบใด มักมีวงจรมาให้สำเร็จรูปตามต้องการโดยที่เราต้องเลือกว่า Stepping Motor ของเรามีกี่เฟส และต้องการการกระตุ้นแบบใด และต้องการจะควบคุมทิศทางด้วยหรือไม่ นอกจากนี้ยังมี IC ที่ใช้ในการควบคุม Stepping Motor โดยเฉพาะ เช่น PMM8713 หรือ L297

-Driver มีหน้าที่รับ Signal จาก Logic sequencer มา On/Off Transister ให้มีการจ่ายไฟเลี้ยง (ต้องมี Volt และ Amp มากพอ เพื่อให้ได้ Torque สูง) ในเฟสตามที่เรต้องการ (อาจใช้ IC L298 แทนการใช้วงจรทั่วไปได้)

การที่ Stepping Motor จะให้ Torque สูงหรือไม่ จะขึ้นกับ Speed ที่ใช้ขณะนั้นและกระแสที่จ่ายให้แต่ละเฟสว่ามีค่าสูงเพียงใดดังในรูปที่ 1.18 แต่ต้องไม่เกินค่าเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น มิชอบญาติให้นำไปใช้ประโยชน์ทางการค้าที่ Stepping Motor ตัวนั้นจะทนได้ ไม่เช่นนั้น Stepping Motor อาจจะเสียได้



รูปที่ 1.18 แสดงความสัมพันธ์ระหว่าง Torque กับกระแสและ Speed



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

การ Interface กับ Computer(8088)

โครงงานนี้ใช้ Computer(8088) ในการควบคุมในลักษณะ Open Loop ดังรูป



โดยวงจรส่วน Interface ที่ใช้ในโครงงานนี้เป็นดังรูปที่ 19 ซึ่งประกอบด้วยส่วนต่างๆซึ่งมีหน้าที่คือ

74LS688

เป็นตัวที่ทำหน้าที่ Decode โดยใช้สวิตช์เลือก โดยจะเชื่อมต่อระหว่าง Computer กับอุปกรณ์ภายนอก โดยหน้าที่ของ 74LS688 นี้จะทำการเปรียบเทียบค่าของ Input 2 ชุดที่ถูกส่งเข้าทางขา P0-P7 และขา Q0-Q7 ถ้า Input ทั้ง 2 ชุดนี้เท่ากันแล้ว Output ที่ขา P=Q จะให้ Output เป็น Logic "0" ซึ่งจากในวงจร P1-P6 ต่อกับแอดเดรสบิต A4-A9 และ P0 กับ P7 ต่อกับ Ground ในขณะที่ขา Q0-Q4 และ Q7 ต่อลง Ground ส่วนขา Q5 และ Q6 ต่อกับสัญญาณ Logic "1" และในการที่จะใช้งาน Port ที่ต่อขึ้นนี้ ขา G (GATE) จะต่อกับสัญญาณ AEN เพื่อป้องกันมิให้ 74LS688 ทำการ Decode ระหว่างขบวนการ DMA สำหรับการอ้าง Address ถึง Port ที่อยู่นอกเครื่อง Computer นั้น A9 จะต้องมามีค่าเป็น 1 ดังนั้น Address ที่สามารถอ้างถึงได้ในโครงงานนี้คือ 0200H จนถึง 03FFH ซึ่งในโครงงานนี้ใช้ Address Port 300H-30BH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญตาเห็นว่าไม่เหมาะสมในการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

74139

เป็น IC ที่ทำหน้าที่เป็น Chip Select โดย A2, A3 จะถูกนำไปใช้เป็นสัญญาณ Select เพื่อเลือกให้ 8255 , 8253-5(1) หรือ 8253-5(2) ตัวใดตัวหนึ่งทำงาน โดยจะใช้คู่กับขา A0, A1 โดย

A2, A3 เป็น Logic 0 ทั้งคู่จะทำให้ 74139 Decoder ส่ง Chip Select ให้ 8255 ทำงาน ในขณะที่เดียวกันจะส่ง A0, A1 มาเพื่อกำหนด Port Register ภายในที่เชื่อมต่อกับ Port Input Output ของ 8255

A2, A3 เป็น Logic 0, 1 ตามลำดับ จะทำให้ 74139 ส่ง Chip Select เพื่อเลือกให้ 8253(1) ทำงาน และจะส่ง A0, A1 มาเพื่อโปรแกรม Register Mode Control และ Register Counter ของ Channel ต่างๆ

A2, A3 เป็น Logic 1, 0 ตามลำดับ จะทำให้ 74139 ส่ง Chip Select เพื่อเลือกให้ 8253(2) ทำงาน และจะส่ง A0, A1 มาเพื่อโปรแกรม Register Mode Control และ Register Counter ของ Channel ต่างๆ

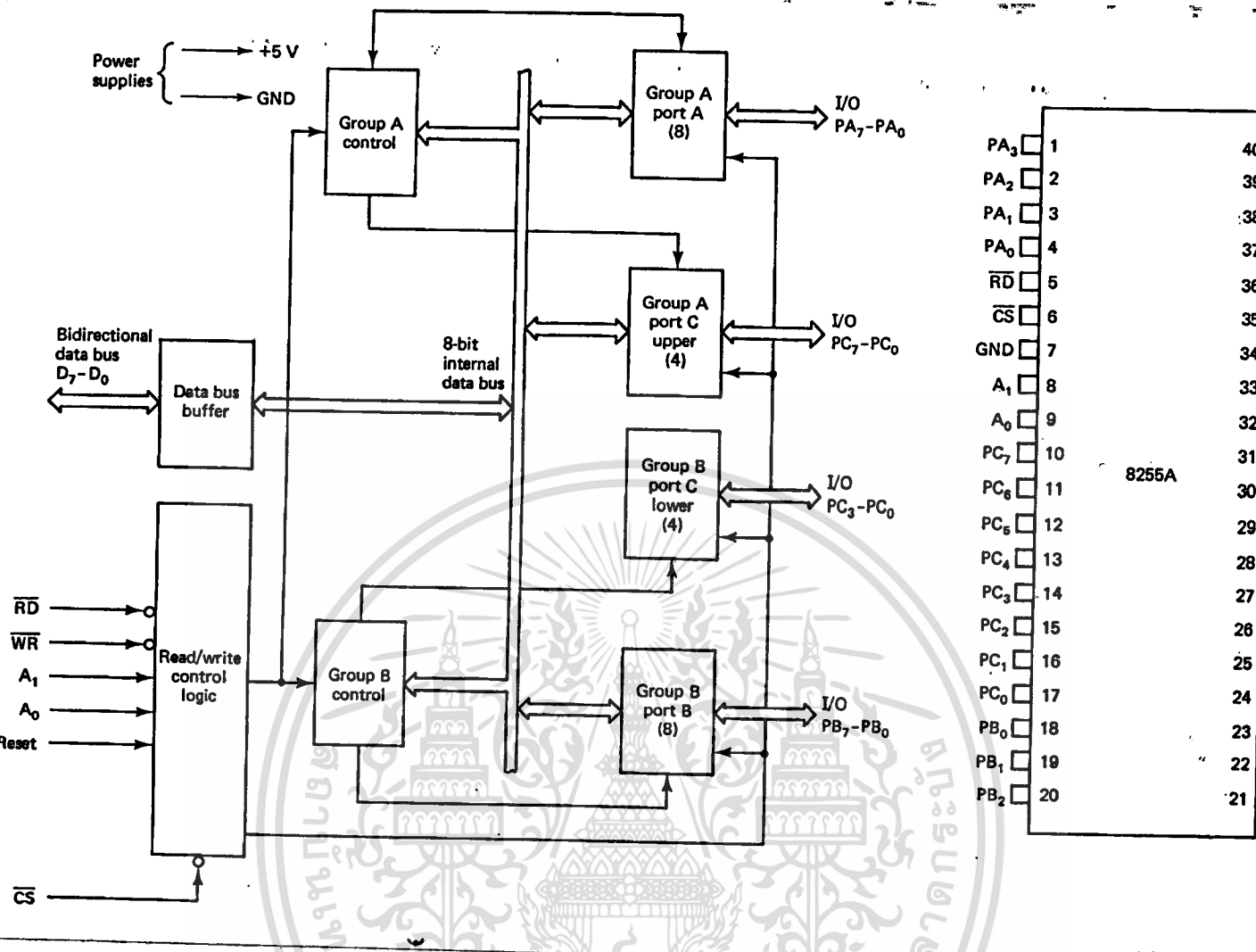
74LS74

เป็น D-Flip Flop ในวงจรเป็นตัวหารความถี่ของสัญญาณ Clock ที่นำมาจาก Slot เนื่องจากสัญญาณ Clock สูงสุดของ Slot ประมาณ 4 MHz แต่ความถี่ที่ใช้ทำงานสูงสุดคือ 1.33 MHz จึงใช้ D-Flip Flop 2 ตัวต่อเป็น Toggle 2 ภาคเป็นวงจรหาร 4 เพื่อให้ได้ Output ไปเข้าที่ขา CLK0 และ CLK1 ของ 8253-5 เพื่อเป็นความถี่มูลฐานของวงจรนับต่อไป

8255

เป็น IC ที่ต่อเป็น Port ได้ 3 Port โดยจะแบ่งเป็น Port A, B และ C โดย Port C จะแบ่งเป็น PC0-PC3 เรียก Port C ล่าง และ Port C บนคือ PC4-PC7 โดย Port ทุก Port เป็นได้ทั้ง Port Input และ Port Output โดยลักษณะการจัดขา IC 8255 และแผนผังวงจรภายในเป็นดังรูปที่ 2.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 แสดงการจัดหา IC 8255 และแผนผังวงจรภายใน

การทำงานของวงจรจะใช้สัญญาณควบคุม Computer โดยจะส่งคำสั่งมาโปรแกรมการทำงานหรือกำหนดรูปแบบของ Port ให้เป็น Input หรือ Output โดยหน้าที่การทำงานของแต่ละขาจะเป็นดังนี้

- D₀-D₇ เป็นขาข้อมูล Input-Output ที่จะต้องผ่านเข้าออกจากส่วนนี้
- CS ขานี้เป็นขา Input ที่จะรับสัญญาณจากภายนอกเพื่อเลือก Chip 8255 นี้ โดยเมื่อขานี้เป็น Logic "0" จะทำให้ตัว 8255 ต่อเข้ากับระบบ Bus ของ Computer
- RD เป็นสัญญาณ Input ที่ต้องส่งมาจาก Computer เมื่อสัญญาณที่ขานี้และขา CS เป็น "0" จะทำให้ Computer อ่านข้อมูลจาก Bus ในขณะที่เป็น Port Input
- WR จะ Active เมื่อสัญญาณที่ขานี้และขา CS เป็น Logic "0" สัญญาณนี้จะมาจาก Computer เมื่อต้องการเขียนข้อมูลลง Port ที่กำหนด

— A₀-A₁ Logic ของขาทั้ง 2 จะถอดรหัสออกเป็น 4 เพื่อกำหนด Register

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภายในที่เชื่อมต่อกับ Port Input-Output ของ 8255

- RESET เป็นสัญญาณที่ส่งมาจากภายนอกเข้ามาทำการ Reset 8255 เพื่อเคลียร์สถานะต่างๆของ 8255 เมื่อ 8255 ได้รับการ Reset มันจะกลับเข้าสู่ Mode Input

- PA0-PA7, PBO-PB7 และ PC0-PC7 เป็นสายสัญญาณที่เป็น Port ของ 8255 โดยการกำหนด Port นี้จะกำหนดโดยขา A0-A1

การโปรแกรมให้ 8255 ทำงานจะต้องโปรแกรมโดยกำหนดควบคู่กับสัญญาณขา RD และ WR โดยสัญญาณของขาควบคุมที่ประกอบกันจะแสดงความหมายดังตาราง

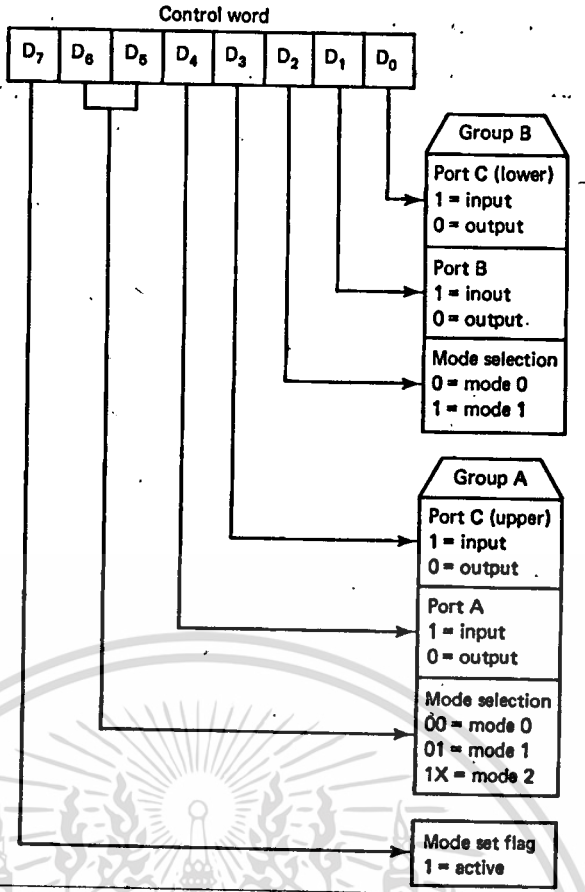
| RD | WR | A1 | AO | ความหมาย |
|----|----|----|----|-------------------------------|
| 1 | 0 | 0 | 0 | เขียน Port A ซึ่งเป็นข้อมูล |
| 0 | 1 | 0 | 0 | อ่าน Port A ซึ่งเป็นข้อมูล |
| 1 | 0 | 0 | 1 | เขียน Port B ซึ่งเป็นข้อมูล |
| 0 | 1 | 0 | 1 | อ่าน Port B ซึ่งเป็นข้อมูล |
| 1 | 0 | 1 | 0 | เขียน Port C ซึ่งเป็นข้อมูล |
| 0 | 1 | 1 | 0 | อ่าน Port C ซึ่งเป็นข้อมูล |
| 1 | 0 | 1 | 1 | เขียนข้อมูลซึ่งเป็นรหัสควบคุม |
| 0 | 1 | 1 | 1 | อ่านเข้ามาซึ่งไม่มีความหมายใด |

ตาราง 2.1 แสดงขาควบคุมของ 8255

การควบคุมการทำงานของ 8255 แบ่งเป็น 3 Mode โดยการกำหนด Mode การทำงานจะต้องส่งข้อมูลคำสั่งเข้าไปโปรแกรมใน Mode ควบคุมของ 8255 โดยแต่ละ Mode มีลักษณะดังนี้

- Mode 0 : เป็น Mode ที่กำหนดให้ Port ทุก Port บนตัว 8255 เป็น Port Input-Output แบบพื้นฐาน โดยการโปรแกรม 8255 คือการให้ค่ารหัสบิตต่างๆเข้าไปในรหัสควบคุมแล้วส่งไปยัง Register ของ Port ควบคุม ซึ่งความหมายแต่ละบิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นับผูกพันให้ใช้หรือเผยแพร่โดยไม่ได้รับอนุญาต
 เอกสารนี้สงวนลิขสิทธิ์ไว้สำหรับงานวิจัยและพัฒนา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 ความหมายของบิตต่างๆของรหัสควบคุม

- Mode 1 : เป็น Mode ที่ทำให้ Input-Output มีการตรวจสอบสัญญาณ โดยใช้ Input-Output ของ Port A และ B เป็นหลัก และใช้ Port C บน เป็นสัญญาณ Handshake ของ Port A ส่วน Port C ล่างเป็นสัญญาณ Handshake B โดยสัญญาณควบคุมแต่ละบิตของ Port C เป็นดังตาราง

| ขา | กรณี Input | กรณี Output |
|-----|------------|-------------|
| PC0 | $INTR_B$ | $INTR_B$ |
| PC1 | IBF_B | OBF_B |
| PC2 | STB_B | ACK_B |
| PC3 | $INTR_A$ | $INTR_A$ |
| PC4 | STB_A | I/O |
| PC5 | IBF_A | I/O |
| PC6 | I/O | ACK_A |
| PC7 | I/O | OBF_A |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 2.2 แสดงสัญญาณควบคุมของ Port C

- Mode 2 : Port C จะทำหน้าที่เป็นสัญญาณตรวจสอบโดยสัญญาณแต่ละขาเป็นดังตาราง

| Port C | ความหมาย |
|--------|-------------------|
| PC0 | I/O |
| PC1 | I/O |
| PC2 | I/O |
| PC3 | INTR _A |
| PC4 | STB _A |
| PC5 | IBF _A |
| PC6 | ACK _A |
| PC7 | OBF _A |

ตาราง 2.3 แสดงสัญญาณตรวจสอบของ Port C

Mode 2 จะทำได้เฉพาะ Port A โดยจะใช้ Port A ทำหน้าที่เป็น Port แบบ 2 ทิศทาง คือ สามารถเป็นได้ทั้ง Input และ Output ซึ่งโครงสร้างของ Port A ทั้ง Input และ Output จะมี Handshake ทั้งคู่

บทที่ 3

8253-5

การสร้าง Pulse โดยใช้ 8253-5 Programmable Timer เป็นส่วนที่สำคัญที่สุดในวงจร โดยภายใน 8253-5 จะมีวงจรมับ 16 บิต 3 แชนแนลคือ แชนแนลที่ 0, 1 และแชนแนลที่ 2 โดยที่เราสามารถโปรแกรมและใช้งานแชนแนลทั้ง 3 นี้แยกกันได้อิสระภายในแต่ละแชนแนลของ 8253-5 จะมีสัญญาณ Input อยู่ 2 เส้น คือ สัญญาณ Clock และสัญญาณ Gate ซึ่งป้อนเข้าทางขา CLK และขา GATE ของแต่ละแชนแนล

นอกจากสัญญาณ Input ทั้ง 2 เส้นแล้ว แต่ละแชนแนลจะมีสัญญาณ Output อีก 1 เส้น คือ สัญญาณที่ถูกส่งออกมาทางขา OUT ของแต่ละแชนแนล ซึ่ง Output ที่ถูกส่งออกมาทางขา OUT นี้จะขึ้นอยู่กับ Mode การทำงานของแต่ละแชนแนลที่เราได้ทำการโปรแกรมไว้ก่อนหน้าที่แต่ละแชนแนลจะเริ่มต้นการทำงาน

สำหรับการโปรแกรม 8253-5 จะทำได้โดยการใช้คำสั่ง OUT ส่งข้อมูล(คำสั่ง)ให้กับ Register ต่างๆของ 8253-5 ซึ่ง Register ของ 8253-5 จะแบ่งออกได้เป็น 2 ประเภทคือ Register Mode Control และ Register Counter

3.1 Register Counter

ภายใน 8253-5 จะมี Register Counter ซึ่งเป็น Register ขนาด 16 บิต อยู่ 3 ตัว โดยข้อมูลใน Register แต่ละตัวจะควบคุมช่วงเวลาของ Output ในแต่ละแชนแนล

เมื่อ 8253-5 เริ่มต้นการทำงานในแชนแนลใดแล้ว ข้อมูลใน Register Counter ของแชนแนลนั้นจะถูกลดค่าลง 1, 2 หรือ 3 (ขึ้นอยู่กับ Mode การทำงาน) เมื่อแชนแนลนั้นได้รับ Clock แต่ละลูก ข้อมูลใน Register Counter จะถูกลดลงจนมีค่าเป็น "0" จากนั้นจึงจะเกิดการเปลี่ยนแปลงทางด้าน Output และการทำงานของแชนแนลนั้น สำหรับการลดค่าของ Register Counter นี้ อาจจะลดค่าลงแบบ BCD หรือ Binary ก็ได้ขึ้นอยู่กับ การ Set ข้อมูลในบิต DO ของ Register Mode Control และเนื่องจาก Register Counter เป็น Register ขนาด 16 บิตๆ แต่ Data Bus

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของ 8253-5 มีขนาดเพียง 8 บิต ดังนั้นการเขียนหรืออ่านข้อมูลจาก Register Counter จะถูกแบ่งออกเป็น 2 ส่วน คือ 8 บิตบน (Most Significant Bit ; D15-D8) และ 8 บิตล่าง (Least Significant Bit ; D7-D0) ส่วนการที่จะกำหนดว่าข้อมูลที่ส่งให้กับ Register Counter นี้เป็นข้อมูลสำหรับ 8 บิตบน หรือ 8 บิตล่างนั้น เราจะกำหนดได้โดยการ Set ค่าในบิต D5 และ D4 ของ Register Mode Control โดยที่ Register Counter จะรับข้อมูลก็ต่อเมื่อสัญญาณที่ขา CLK ถูกเปลี่ยนจาก Logic "0" เป็น "1" และเปลี่ยนจาก Logic "1" กลับเป็น "0" อีกครั้งก่อน และสำหรับหน้าที่ของบิตต่างๆใน Register Mode Control จะแสดงดังตารางข้างล่างนี้

| | | | | | | | | |
|---------|-----|-----|-----|-----|----|----|----|-----|
| บิต | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| หน้าที่ | SC1 | SC0 | RL1 | RLO | M2 | M1 | M0 | BCD |

ตาราง 3.1 แสดงหน้าที่ของบิตต่างๆใน Register Mode Control

บิต D7-D6 : (SC1, SC0) : ข้อมูลในบิตทั้ง 2 นี้จะใช้สำหรับเลือกแชนแนลที่ต้องการ ซึ่งแสดงดังตารางข้างล่าง

| SC1 | SC2 | แชนแนลที่ถูกเลือก |
|-----|-----|-------------------|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 2 |

ตาราง 3.2 แสดงข้อมูลของบิต SC1 และ SC2

บิต D4-D5 : ใช้ในการกำหนดไบท์ (8 บิตล่างหรือ 8 บิตบน) ของข้อมูลที่ต้องการจะอ่านจาก Register Counter ซึ่งเป็น Register ขนาด 16 บิต และใช้ในการกำหนดว่าโปรแกรม Register Counter นี้จะทำการโปรแกรมเฉพาะไบท์ใดไบท์หนึ่งหรือ

โปรแกรมทั้ง 2 ไบท์ นอกจากนี้ยังใช้ในการแลทช์ (Latch) ค่าใน Register Counter ได้อีกด้วย สำหรับผลการ Set ข้อมูลภายในบิต D4 และ D5 จะแสดงดังตารางข้างล่าง

| RLO | RL1 | |
|-----|-----|---|
| 0 | 0 | ทำการแลทช์ค่าใน Register Counter |
| 0 | 1 | อ่าน/เขียนเฉพาะข้อมูลเฉพาะใน 8 บิตล่าง |
| 1 | 0 | อ่าน/เขียนเฉพาะข้อมูลเฉพาะใน 8 บิตบน |
| 1 | 1 | อ่าน/เขียนเฉพาะข้อมูลทั้ง 16 บิต โดยเริ่มจาก 8 บิตล่างก่อน จากนั้นจึงอ่าน/เขียนข้อมูลใน 8 บิตบน |

ตาราง 3.3 แสดงหน้าที่ของข้อมูลในบิต D4 และ D5

บิต D3-D1 : (M2-M1) : ข้อมูลในบิตนี้ใช้สำหรับเลือกโหมดการทำงานให้กับ แชนแนลที่เรากำหนดในบิต SC0 และ SC1 สำหรับ Mode การทำงานของแชนแนลต่างๆ ใน 8253-5 จะมี 6 Mode โดยผลการ Set เป็นดังตารางข้างล่าง

| M2 | M1 | M0 | โหมดการทำงาน |
|----|----|----|--------------------------------------|
| 0 | 0 | 0 | Mode 0 : Interrupt on Terminal Count |
| 0 | 0 | 1 | Mode 1 : Programmable One-Shot |
| 0 | 1 | 0 | Mode 2 : Rate Generator |
| 0 | 1 | 1 | Mode 3 : Square Wave Generator |
| 1 | 0 | 0 | Mode 4 : Software Triggered Strobe |
| 1 | 0 | 1 | Mode 5 : Hardware Triggered Strobe |

ตาราง 3.4 แสดงหน้าที่ของข้อมูลในบิต D3-D1

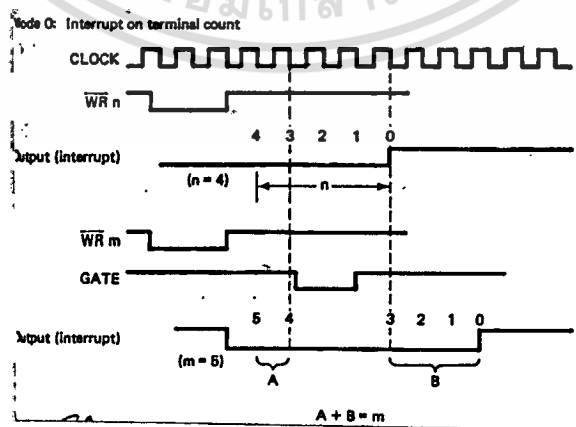
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
บิต D0 : (BCD) : ข้อมูลในบิตนี้จะใช้สำหรับกำหนดว่าการลดค่าของข้อมูลในไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำใน

Register Counter นั้นจะทำการลดในลักษณะของ BCD หรือ Binary โดยถ้าข้อมูลในบิตนี้ถูก Set เป็น "1" ค่าของข้อมูลใน Register Counter จะถูกลดลงในลักษณะของ BCD และถ้าถูก Set เป็น "0" ค่าของข้อมูลใน Register Counter ก็จะถูกลดลงในลักษณะของ Binary

3.2 การทำงานของ 8253-5

การทำงานของ 8253-5 สามารถทำได้ 6 Mode ดังนี้

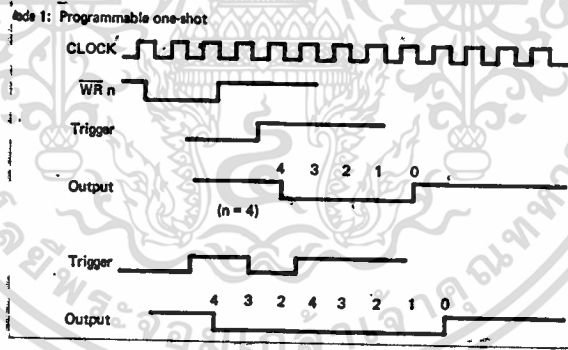
1. **Mode 0 : Interrupt On Terminal Count** แชนแนลที่ถูกโปรแกรมให้ทำงานใน Mode นี้ Output จะมีระดับ Logic เป็น "0" เมื่อค่า Counter ถูกส่งให้กับ Register Counter ของแชนแนลนั้นแล้ว แชนแนลนั้นก็จะเริ่มนับจำนวน Clock ที่รับเข้ามา โดย Output ของแชนแนลนั้นยังคงเป็น Logic "0" อยู่ ซึ่งในระหว่างการนับข้อมูลใน Register Counter จะถูกลดค่าลงเรื่อยๆ เมื่อข้อมูลใน Register Counter ถูกลดจนเป็น "0" Output ของแชนแนลนั้นก็จะถูกเปลี่ยนเป็น Logic "1" และจะคงสภาพ Logic "1" นี้อยู่จนกว่าจะมีการส่งข้อมูลให้กับ Register Mode Control ใหม่ และถ้ามีการโหลดค่า Counter ใหม่ให้กับแชนแนลที่อยู่ระหว่างการนับใน Mode 0 จะมีผลคือ เมื่อมีการโหลดไบต์แรกของ Counter ค่าใหม่ การนับเดิมจะหยุดนับ และเมื่อโหลดไบต์ที่ 2 แชนแนลนั้นจะเริ่มต้นการนับใหม่ตาม Counter ค่าใหม่ที่ได้รับ โดยการทำงานงานของ Mode 0 สามารถแสดงดังรูปที่ 3.1



รูปที่ 3.1 ไต่อะแกรมเวลาการทำงานของ 8253-5 ใน Mode 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 2. **Mode 1 : Programmable One Shot** การทำงานของ Mode นี้ เมื่อค่า
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีกรณ

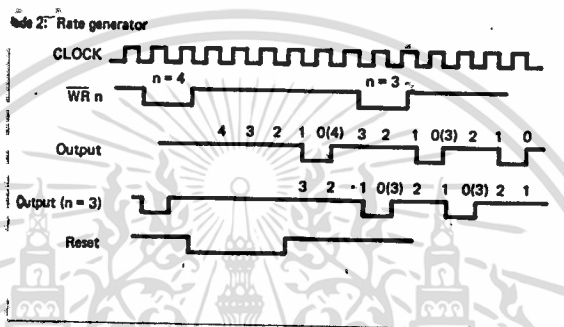
Counter ถูกโหลดให้กับ Register Counter แล้ว Output ของแชนแนลนั้นจะถูกปรับให้เป็น Logic "1" และเมื่อ Input ที่ขา GATE ของแชนแนลนั้นถูกเปลี่ยนจาก Logic "0" เป็น "1" Output ก็จะเปลี่ยนเป็น Logic "0" และจะเริ่มนับจำนวน Clock ที่ได้รับเข้ามาทางขา CLK ของแชนแนลนั้น เมื่อนับครบตามจำนวนที่กำหนดไว้ใน Register Counter แล้ว Output ก็จะเปลี่ยนกลับเป็น Logic "1" อีกครั้ง ถ้ามีการโหลด Counter ค่าใหม่ในระหว่างการทำงานอยู่ ค่า Counter ค่าใหม่จะยังไม่มีผลใดๆกับการนับเดิมจนกว่าแชนแนลนั้นจะเสร็จจากการนับตามค่า Counter ที่ Set ไว้เดิมเสียก่อน และเมื่อสัญญาณที่ขา GATE เปลี่ยนจาก Logic "0" เป็น "1" อีกครั้ง 8253-5 ก็จะเริ่มต้นการนับใหม่โดยใช้ค่า Counter ใหม่ที่ได้รับ สำหรับค่า Counter ที่เก็บไว้ใน Register Counter นั้น เราสามารถทำการอ่านออกมาได้ตลอดเวลาโดยไม่กระทบการนับของแชนแนลที่กำลังนับอยู่เลย สำหรับ Input ที่เข้ามาทางขา GATE ของแชนแนลที่ทำงานใน Mode 1 นั้นจะมีลักษณะเป็น Retriggerable ดังแสดงในรูปที่ 3.2



รูปที่ 3.2 ไตอะแกรมเวลาการทำงานในโหมด 1 ของ 8253-5

3. Mode 2 : Rate Generator การทำงานใน Mode นี้จะมีลักษณะเป็นการหารความถี่ของ Clock ที่ได้รับด้วยค่าที่โหลดให้กับ Register Counter คือสัญญาณ Output ของแชนแนลที่ทำงานในโหมดนี้จะมียระดับ Logic เป็น "1" และ "0" สลับกันไป โดยที่ความกว้างของช่วงเวลาที่ เป็น Logic "1" จะเท่ากับช่วงเวลาของ Clock จำนวน N-1 ลูก (ค่า N ในที่นี้จะแทนค่า Counter ที่โหลดให้กับ Register Counter) และความกว้างของช่วงเวลาที่ เป็น Logic "0" จะเท่ากับช่วงเวลาของ

Clock จำนวน 1 ลูก ดังนั้นคาบเวลาของ Output จากเซนแนลที่ทำงานใน Mode นี้จึงเท่ากับช่วงเวลาของ Clock N ลูก ซึ่งก็คือความถี่ของ Output จากเซนแนลนี้จะเท่ากับความถี่ที่ได้รับหารด้วย N นั่นเอง และในระหว่างคาบเวลาของ Output จากเซนแนลที่ทำงานใน Mode 2 ถ้าเราทำการโหลด Counter ค่าใหม่ ค่า Counter ที่โหลดเข้าไปใหม่จะมีผลต่อคาบเวลาต่อไปเท่านั้น โดยจะไม่มีผลต่อคาบเวลาของ Output ที่เซนแนลนี้กำลังสร้างอยู่ ดังแสดงในรูปที่ 3.3



รูปที่ 3.3 ไตอะแกรมเวลาการทำงานใน Mode 2 ของ 8253-5

สำหรับระดับสัญญาณ Input ที่ขา GATE จะมีหน้าที่ในการควบคุมการทำงานด้วย โดยถ้าสัญญาณ Input ที่ขา GATE นี้มี Logic เป็น "0" Output จะเป็น Logic "1" ตลอดเวลา และจะเริ่มการนับเมื่อ Logic ของ Input ที่ขา GATE เปลี่ยนเป็น "1" และสำหรับหลังจากที่เราทำการ Set การทำงานใน Mode 2 ให้กับเซนแนลใดแล้ว เซนแนลนั้นจะยังไม่เริ่มการทำงาน (Output เป็น Logic "1" ตลอดเวลา) จนกว่าเราจะโหลดค่า Counter ให้กับ Register Counter ของเซนแนลนั้นเสียก่อน

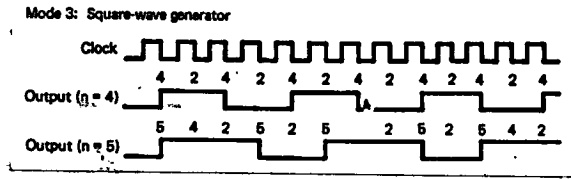
4. Mode 3 : Square Wave Rate Generator การทำงานใน Mode นี้จะมีลักษณะคล้ายกับใน Mode 2 คือเป็นการหารความถี่ของสัญญาณ Clock ที่ป้อนให้กับเซนแนลที่ทำงานใน Mode นี้ด้วย N (ค่า Counter ที่โหลดให้กับ Register Counter) แต่ต่างกันคือ ช่วงเวลาที่ Output เป็น Logic "1" และ Logic "0" ใน 1 คาบจะเท่ากัน (หรือเกือบเท่ากัน) การทำงานของ Mode 3 จะแบ่งออกได้เป็น 2 ลักษณะตามค่าของ Counter ที่โหลดให้กับ Register Counter (ในที่นี้จะแทนค่า Counter ด้วยตัวแปร N) ดังนี้

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ถ้า N เป็นเลขคู่ : Output จะมีช่วงเวลาที่ เป็น Logic "1" นานเท่ากับช่วงที่เป็น Logic "0" นานเท่ากับช่วงเวลาของ Clock $N/2$ ลูก โดยมีการทำงานคือ หลังจากเริ่มการทำงาน Output ของแชนแนลนี้เป็น Logic "1" จากนั้นทุกครั้งที่ Clock ที่ป้อนให้เปลี่ยน Logic จาก "1" เป็น "0" ค่าใน Register Counter จะถูกลดลง 2 ซึ่งจะลดจนกว่า Register Counter มีข้อมูลเป็น "0" ช่วงนี้ Output จะเปลี่ยนเป็น Logic "0" และ Register Counter จะถูกเปลี่ยนข้อมูลเป็นค่า Counter เริ่มต้นที่เราส่งให้ก่อนที่จะเริ่มต้นการทำงาน (ในที่นี้คือค่า N) โดยอัตโนมัติ และเมื่อสัญญาณ Clock เปลี่ยนจาก Logic "1" เป็น "0" ค่าใน Register Counter ก็จะถูกลดลงอีก 2 จนค่าใน Register Counter เป็น "0" Output จึงจะกลับเป็น Logic "1" อีกครั้ง ค่า N ก็จะถูกโหลดให้กับ Register Counter อีกโดยอัตโนมัติ จากนั้นจึงทำงานตามขั้นตอนต่างๆต่อไป

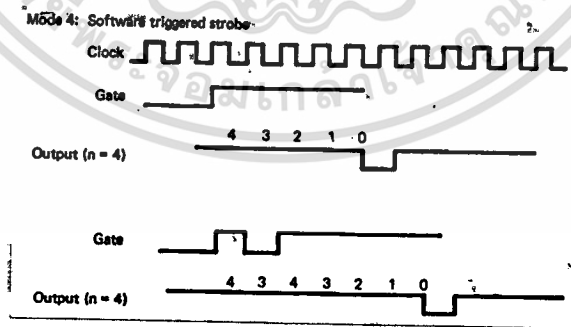
- ถ้า N เป็นเลขคี่ : ในกรณีนี้ช่วงเวลา 1 คาบของ Output จะมีช่วงเวลาที่ เป็น Logic "1" อยู่ยาวนานเท่ากับช่วงเวลาของ Clock $(N+1)/2$ ลูก และจะมีช่วงเวลาที่ มี Logic "0" นานเท่ากับช่วงเวลาของ Clock $(N-1)/2$ ลูก โดยในขณะที่เริ่มทำงานใน Mode 3 Output จะมีระดับ Logic เป็น "1" และเมื่อได้รับ Clock ลูกแรก ข้อมูลใน Register Counter จะถูกลดลง "1" จากนั้นเมื่อแชนแนลนี้ได้รับ Clock ลูกต่อไป ข้อมูลใน Register Counter ก็จะถูกลดลงครั้งละ "2" จนกระทั่งข้อมูลใน Register Counter เป็น "0" ในช่วงนี้ Output จะถูกเปลี่ยนระดับเป็น "0" และข้อมูลใน Register Counter จะถูกโหลดด้วยค่าเริ่มต้น (ในที่นี้คือ N) โดยอัตโนมัติ จากนั้นเมื่อได้รับ Clock ลูกแรกข้อมูลใน Register Counter จะถูกลดลง "3" และเมื่อแชนแนลนี้ได้รับ Clock ลูกต่อไป ข้อมูลใน Register Counter ก็จะถูกลดลงครั้งละ "2" จนกระทั่งค่าใน Register Counter เป็น "0" แชนแนลนี้ก็จะเริ่มทำงานตามขั้นตอนแรกใหม่ต่อไป

โดยลักษณะการทำงานของ Modec 3 สามารถแสดงดังรูปที่ 3.4



รูปที่ 3.4 ไตอะแกรมเวลาการทำงานใน Mode 3 ของ 8253-5

5. Mode 4 : Software Triggered Strobe จะเริ่มต้นทำงานก็ต่อเมื่อเรา โหลดค่า Counter ให้กับ Register Counter แล้วเท่านั้น โดยเมื่อโหลดข้อมูลให้กับ Register Counter แล้ว Output ของแชนแนลนั้นจะมีระดับ Logic เป็น "1" และ จะเริ่มต้นนับจำนวน Clock ที่ได้รับเข้ามาทางขา CLK เมื่อครบตามจำนวนที่ได้กำหนดใน Register Counter แล้ว Output จะเปลี่ยนระดับ Logic เป็น "0" โดยช่วงเวลาที่ Output เป็น Logic "0" นี้จะนานเท่ากับช่วงเวลาของ Clock 1 ลก จากนั้นแชนแนลนั้นก็ยุติการทำงานและ Output ก็กลับเป็น Logic "1" อีกครั้ง ดังแสดงใน รูปที่ 3.5

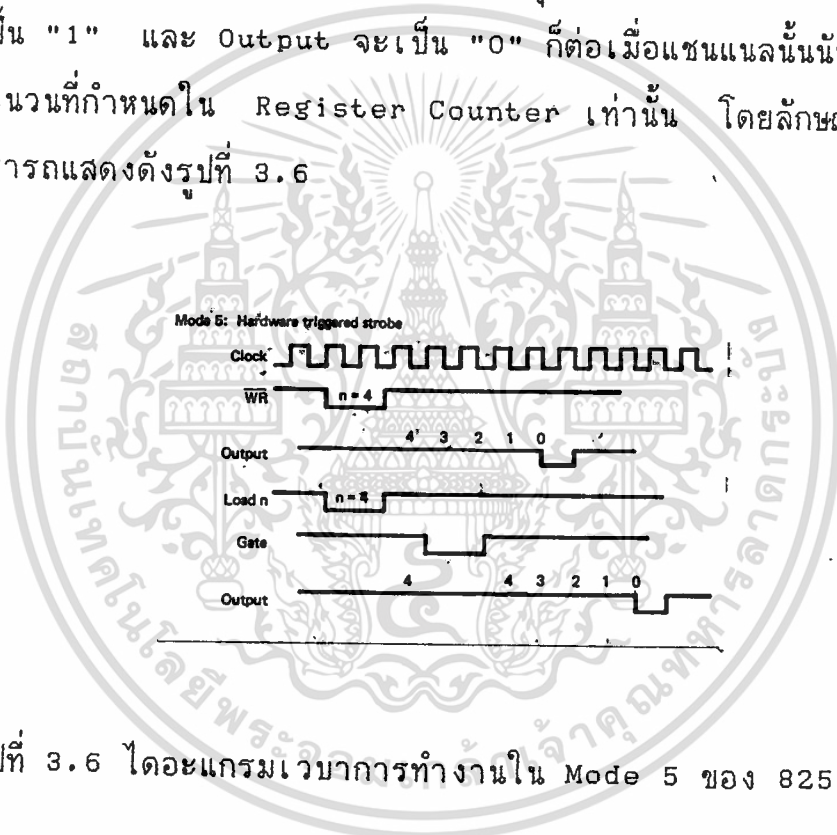


รูปที่ 3.5 ไตอะแกรมเวลาการทำงานใน Mode 4 ของ 8253-5

และในระหว่างที่กำลังทำงานใน Mode 4 นี้ ถ้าโหลด Counter ค่าใหม่นี้ให้กับ Register Counter ค่า Counter ที่โหลดเข้าไปใหม่จะยังไม่มียผลต่อการทำงานจนกว่า จะเสร็จจากการทำงานเสียก่อน และจากรูปจะเห็นว่าในระหว่างที่มีการนับจำนวน ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต่ออายุใช้งานของเอกสารนี้

Clock อยู่ นั่น ถ้าสัญญาณ GATE ถูกเปลี่ยนเป็น Logic "0" จะทำให้การนับนั้นสิ้นสุดลง และจะเริ่มนับใหม่อีกเมื่อสัญญาณที่ขา GATE เปลี่ยนเป็น Logic "1"

6. **Mode 5 : Hardware Triggered Strobe** จะเริ่มทำงานเมื่อสัญญาณที่ขา GATE ถูกเปลี่ยนจาก Logic "0" เป็น "1" เท่านั้น เมื่อเริ่มทำงาน Output จะมี Logic "1" เมื่อนับครบตามจำนวนที่กำหนดใน Register Counter แล้ว Output จึงเปลี่ยนเป็น Logic "0" นานเท่ากับ Clock 1 ลูก แล้ว Output จึงกลับเป็น "1" และการทำงานใน Mode 5 จะเริ่มทำการนับใหม่ทุกครั้งที่มี Logic ที่ขา GATE ถูกเปลี่ยนจาก "0" เป็น "1" และ Output จะเป็น "0" ก็ต่อเมื่อชนแนลนั้นนับจำนวน Clock ได้ครบตามจำนวนที่กำหนดใน Register Counter เท่านั้น โดยลักษณะการทำงานใน Mode 5 สามารถแสดงดังรูปที่ 3.6



รูปที่ 3.6 ไตอะแกรมเวเวการการทำงานใน Mode 5 ของ 8253-5

3.3 การโปรแกรม Register ของ 8253-5

การโปรแกรมเราต้องทำการโปรแกรมรีจิสเตอร์ของชนแนลที่ต้องการโปรแกรม 2 Register คือ Register Mode Control และ Register Counter ตามลำดับก่อน การโปรแกรม Register Mode Control โดยการ Set ให้ A0, A1 เป็น Logic "1" ทั้งคู่ก่อน แล้วส่งค่าให้ D4, D5 (RLO, RL1) เพื่อกำหนดจำนวนไบต์ที่ต้องส่งออกมาตามที่กำหนดข้างต้น พร้อมทั้งส่งค่า D6, D7 (SCO, SC1) เพื่อเลือกชนแนลที่ต้องการโปรแกรมตามที่กำหนดข้างต้น โดยในการโปรแกรมเราไม่จำเป็นต้องเรียงตามลำดับว่าจะต้องโปรแกรมชนแนลใดก่อน ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากโปรแกรม Register Mode Control แล้วเราจะโปรแกรม Register Counter โดยการ Set ค่า A0,A1 ตามตารางข้างล่าง

| A1 | A0 | Register ที่ถูกโปรแกรม |
|----|----|---------------------------|
| 0 | 0 | Register Counter แชนแนล 0 |
| 0 | 1 | Register Counter แชนแนล 1 |
| 1 | 0 | Register Counter แชนแนล 2 |
| 1 | 1 | Register Mode Control |

ตาราง 3.5 แสดงหน้าที่ของบิต A0 และ A1 ในการโปรแกรม Register Counter สำหรับลำดับของการโปรแกรม Register Counter เราจะเลือกโปรแกรมแชนแนลใดก่อนก็ได้ โดยไม่ขึ้นกับลำดับของการโปรแกรม Register Mode Control แต่สิ่งหนึ่งที่ต้องคำนึงถึงคือลำดับและจำนวนไบต์ของข้อมูลที่จะต้องส่งให้กับ Register Counter ซึ่งจะขึ้นอยู่กับค่าของบิต RL1 และ RLO ดังที่กล่าวไว้ตอนต้น

3.4 การใช้งานขา GATE INPUT

ในแต่ละ Mode การทำงานของ Counter จะมีการใช้ขา Input Gate ที่แตกต่างกันไป ซึ่งหน้าที่ทั้งหมดของขา Input Gate จะแสดงดังตาราง

| Modes | Signal Steatus | | |
|-------|--|--|---------------------|
| | Low Or Going Low | Rising | High |
| 0 | Disables Counting | | Enables Counting |
| 1 | — | 1)Initiates Counting 2)Resets Output after next clock | — |
| 2 | 1)Disables counting 2)Sets output immediately high | 1)Reloads counter 2)Initiates counting | Enables counting |
| 3 | 1)Disables counting 2)Sets output immediately high | Initiates counting | Enables counting |
| 4 | Disables counting | — | Enables counting |
| 5 | — | Initiates counting | — |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ตาราง 3.6 แสดงหน้าที่ของขา Input Gate ของ 8253
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

วงจรที่ใช้ในโรงงาน

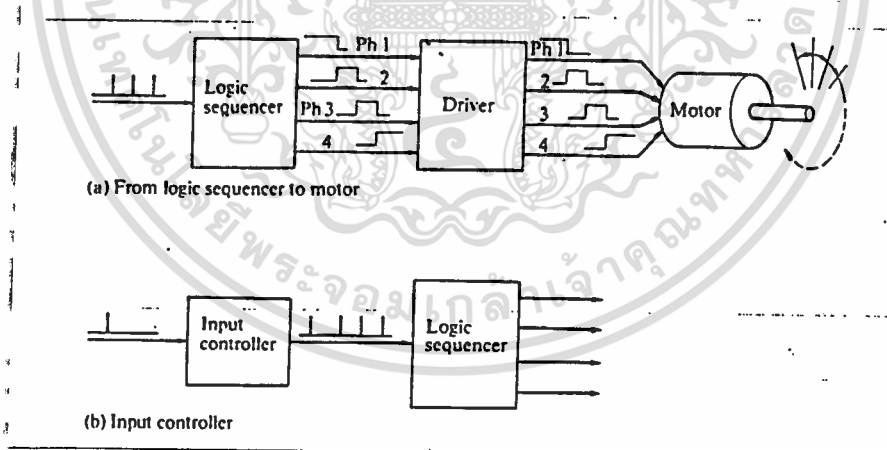
โรงงานนี้ได้แบ่งออกเป็น 2 ส่วนใหญ่ๆ คือ

- ส่วน Hardware
- ส่วน Software

โดยส่วน Hardware สามารถแบ่งเป็น 4 ส่วน คือ

- ส่วน Interface Computer (8088)
- ส่วน Input Controller
- ส่วน Logic Sequencer
- ส่วน Driver

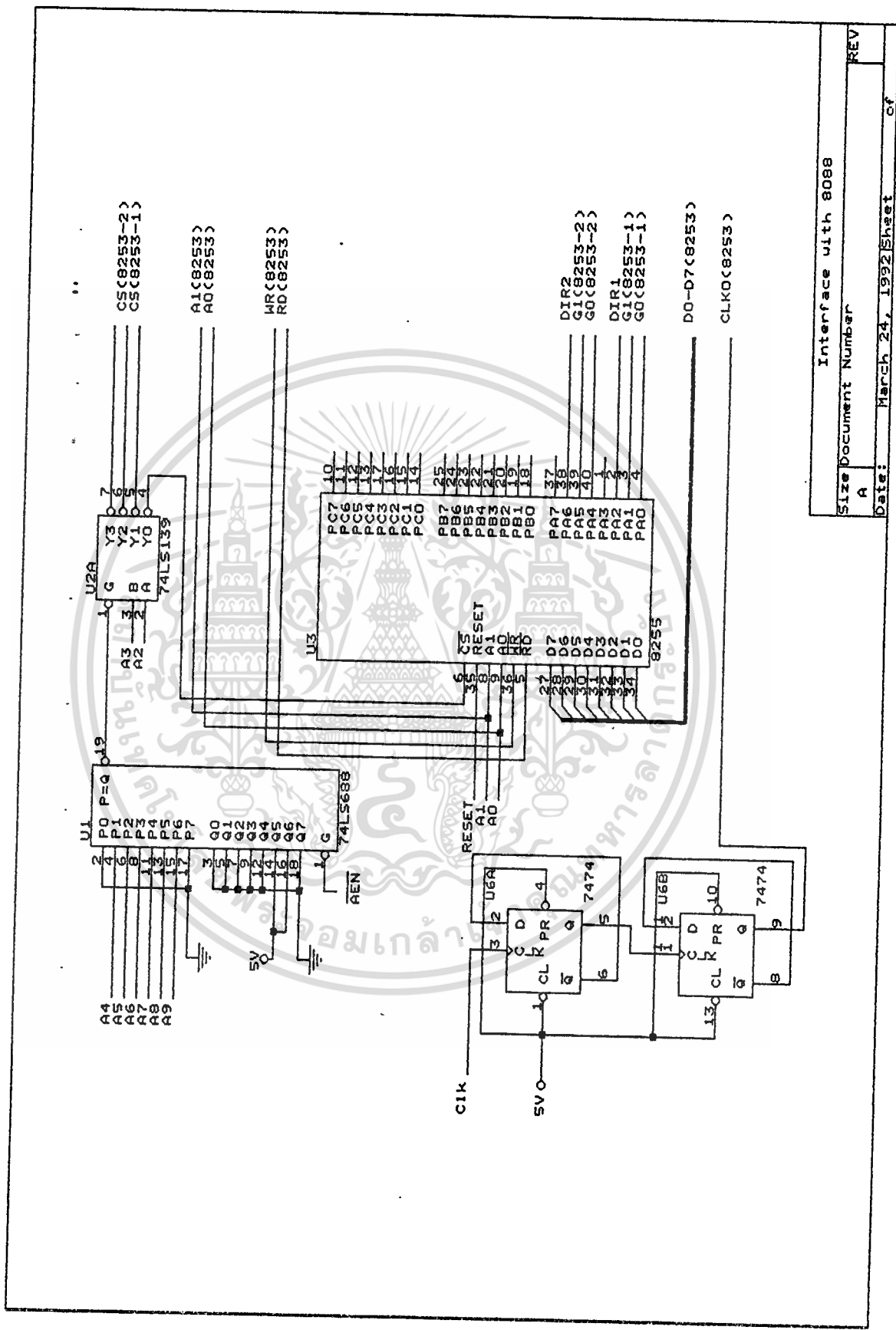
โดย Block Diagram ของการทำงานเป็นดังรูปที่ 4.1



รูปที่ 4.1 Block Diagram ของการทำงานของวงจร.

4.1 ส่วน Interface

ส่วน Interface เป็นวงจรที่ทำหน้าที่ในการเชื่อมต่อ Computer กับอุปกรณ์ภายนอก ซึ่งวงจรที่ใช้ในโรงงานมีลักษณะดังรูปที่ 4.2 โดยจะใช้ Address A0-A9 โดยที่ A9 จะต้องมีความเป็น Logic "1" ดังที่กล่าวมาแล้ว เมื่อต้องการให้เป็นไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Interface with 8088
 Size Document Number A
 Date: March 24, 1992 Sheet of REV

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Output Port โดยการทำงานของ Mux 74LS688 จะให้ Output ออกมาเป็น 0 เมื่อ $Q7-Q0 = P7-P0$ ที่ในวงจรนี้ค่า $Q0-Q7$ ได้ถูก set เอาไว้ดังนี้

| | | | | | | | |
|----|----|----|----|----|----|----|----|
| Q7 | Q6 | Q5 | Q4 | Q3 | Q2 | Q1 | Q0 |
| 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |

ตาราง 4.1 แสดงค่าของ $Q0-Q7$ ที่ set ในวงจร

ซึ่งค่า $P0, P7$ ก็ได้ถูก set เอาไว้เป็น 0 ด้วย ดังนั้นค่า $P1-P6$ ที่ต่อตรงกับ $A4-A9$ จะต้องมีค่าเป็น 000011 ตามลำดับ ในที่นี่เราใช้ AEN ต่อเข้ากับขาของ Mux เพื่อให้แน่ใจว่า Mux จะทำงานเมื่อมีติดต่อกับ Input หรือ Output เท่านั้น หลังจากได้ Output จาก Mux แล้วเราจะใช้สัญญาณดังกล่าวเป็นตัว Chip Select ให้กับ Decoder 139 เพื่อทำการเลือกให้ 8255 และ 8253 อีก 2 ตัวทำงานตามต้องการ โดยจะใช้ $A2, A3$ ในการ Decoder เพื่อแบ่งออกเป็น 4 Port ใหญ่ที่จะใช้ในการเลือก โดย Port ต่างๆจะถูกแบ่งดังนี้

| Port | Address |
|---------------|-----------|
| 8255 | 300H-303H |
| 8253 ตัวที่ 1 | 304H-307H |
| 8253 ตัวที่ 2 | 308H-30BH |
| no used | 30CH-30FH |

ตาราง 4.2 แสดงตำแหน่ง Address ที่ใช้ควบคุม Port ต่างๆ

ส่วน $A0, A1$ จะใช้ในการ Set Mode Control ต่างๆให้กับทั้ง 8255 และ

8253-5 ดังนี้คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| Outport No. | Function |
|-------------|----------------------|
| 300H | Port A 8255 |
| 301H | Port B 8255 (noused) |
| 302H | Port C 8255 (noused) |
| 303H | Control Word 8255 |
| 304H | CH 0 8253 1 |
| 305H | CH 1 8253 1 |
| 306H | CH 2 8253 1 (noused) |
| 307H | Mode Control 8253 1 |
| 308H | CH 0 8253 2 |
| 309H | CH 1 8253 2 |
| 30AH | CH 2 8253 2 (noused) |
| 30BH | Mode Control 8253 2 |

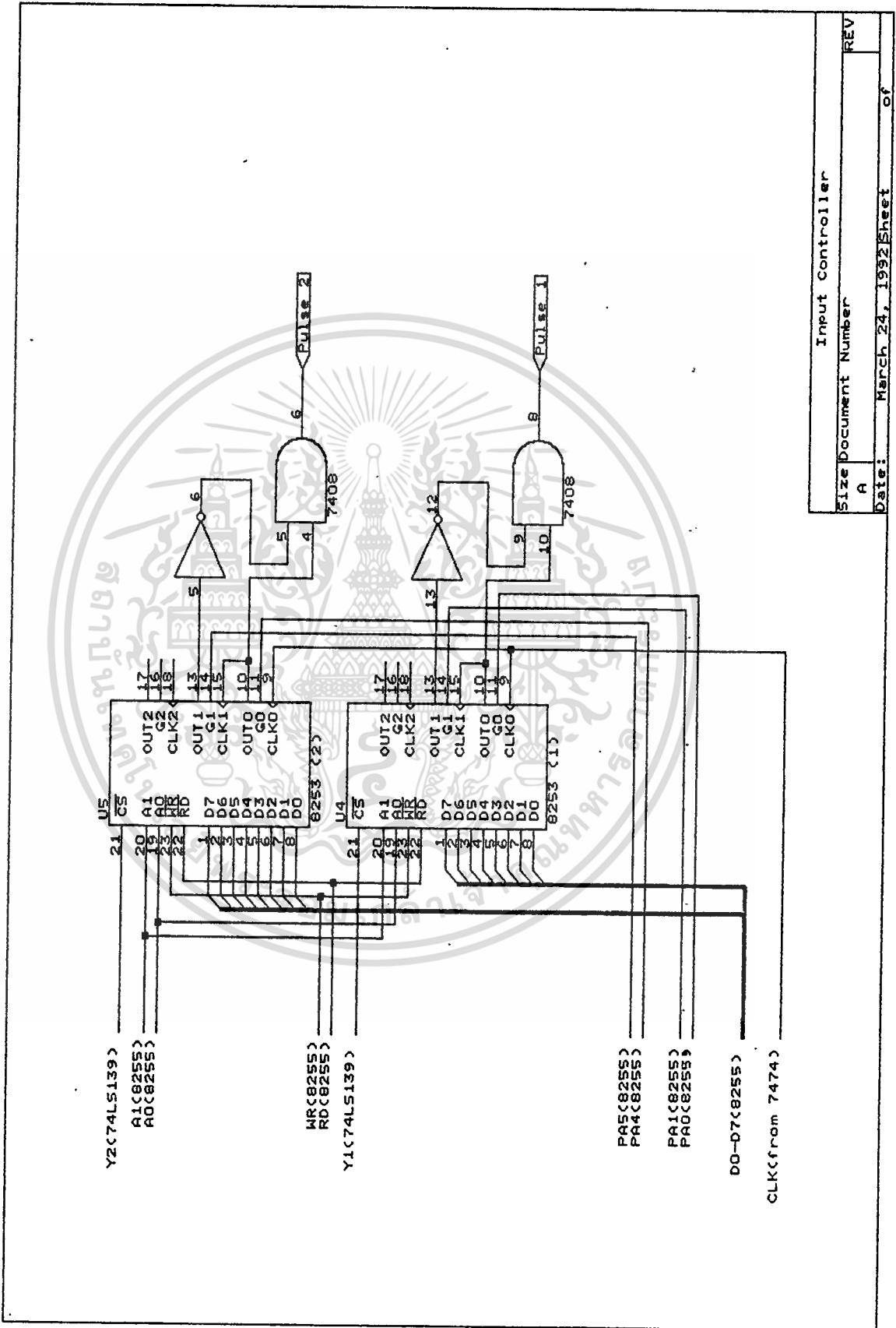
ตาราง 4.3 แสดงหน้าที่ของ Address ที่ใช้ในวงจร ทำให้เราเขียนตารางการทำงานโดยสรุปของ Address ต่างๆได้ดังนี้

| | |
|-------|--|
| A9-A4 | ใช้ในการ Mux เพื่อ Chip Select มายัง Decoder |
| A3,A2 | ใช้ในการ Decoder เพื่อเลือก 8255 และ 8253 |
| A1,A0 | ใช้ในการ Set Mode Control ให้กับทั้ง 8255 และ 8253 |

ตาราง 4.4 แสดงหน้าที่โดยสรุปของ Address ต่างๆ

4.2 ส่วน Input Controller

ในการควบคุม Stepping Motor เราต้องควบคุมตัวแปร 3 ตัวคือ จำนวนองศาที่หมุนไปของ Motor, ความเร็วในการหมุนและทิศทางในการหมุน โดยที่องศาการหมุนควบคุมได้โดยการควบคุมจำนวน Pulse ที่ป้อนให้กับ Stepping Motor ส่วนความเร็วไม่จากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

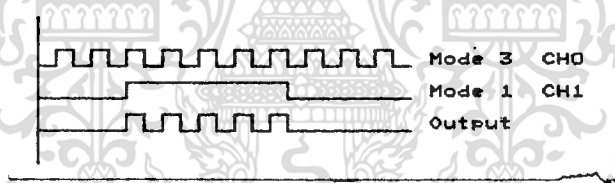


| | |
|------------------|-----------------|
| Input Controller | |
| Size | Document Number |
| A | REV |
| Date: | March 24, 1992 |
| Sheet | of |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

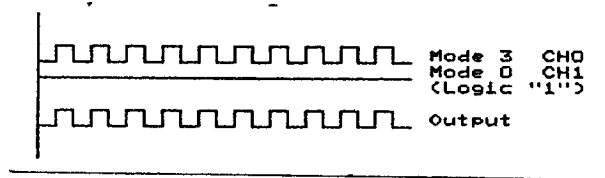
ในการหมุนควบคุมจากความถี่ที่เราจะป้อนให้กับ Stepping Motor โดยที่สัญญาณทั้ง 2 จะถูกส่งมาในสายสัญญาณเดียวกัน ส่วนทิศทางการหมุนจะถูกควบคุมโดย 8255 ถ้าต้องการให้หมุนตามเข็มนาฬิกาให้ส่งสัญญาณเป็น 1 มา และส่ง 0 ถ้าต้องการหมุนในทิศทางตรงกันข้าม ส่วนที่สำคัญที่สุดของ Input Controller คือการสร้างสัญญาณ Pulse ให้ได้ ความถี่และจำนวน Pulse ตามต้องการ โดยใช้ 8253-5 ในวงจรเราจะ Set CH0 8253 ให้ทำงานใน Mode 3 (Square Wave Generator) เพื่อจะสร้างความถี่ตามที่ เราต้องการ โดย Output CH0 จะถูกนำไปเป็น Clock ของ CH1

ในการควบคุมจำนวน Pulse หรือ Step ทำได้โดย Set mode 1 (Monostable) ให้กับ CH1 8253 แล้วทำการ Set Counter ที่ต้องการ จึงเสมือน การนับ Pulse จาก CH0 ซึ่ง Output CH1 จะถูกนำไปผ่าน Inverter ก่อน เพื่อให้ กลายเป็น Logic 1 แล้วจึงนำไป And กับ Output CH0 จะทำให้เราได้ Pulse Train ที่มีทั้งความถี่ และจำนวน Pulse ตามที่เราต้องการ



รูปที่ 4.4 แสดงสัญญาณควบคุมจำนวน Pulse หรือ Step

แต่ถ้าหากว่าเราต้องการควบคุมความถี่เพียงอย่างเดียวเราจะทำการ Set CH1 Mode 0 แต่ไม่ต้อง Set Counter เพียงเพื่อต้องการให้ได้ Output ออกมาเป็น 0 เพื่อนำไป ผ่าน Inverter แล้วนำไป And กับ Output CH0 ที่ให้ความถี่ที่เราต้องการออกมา



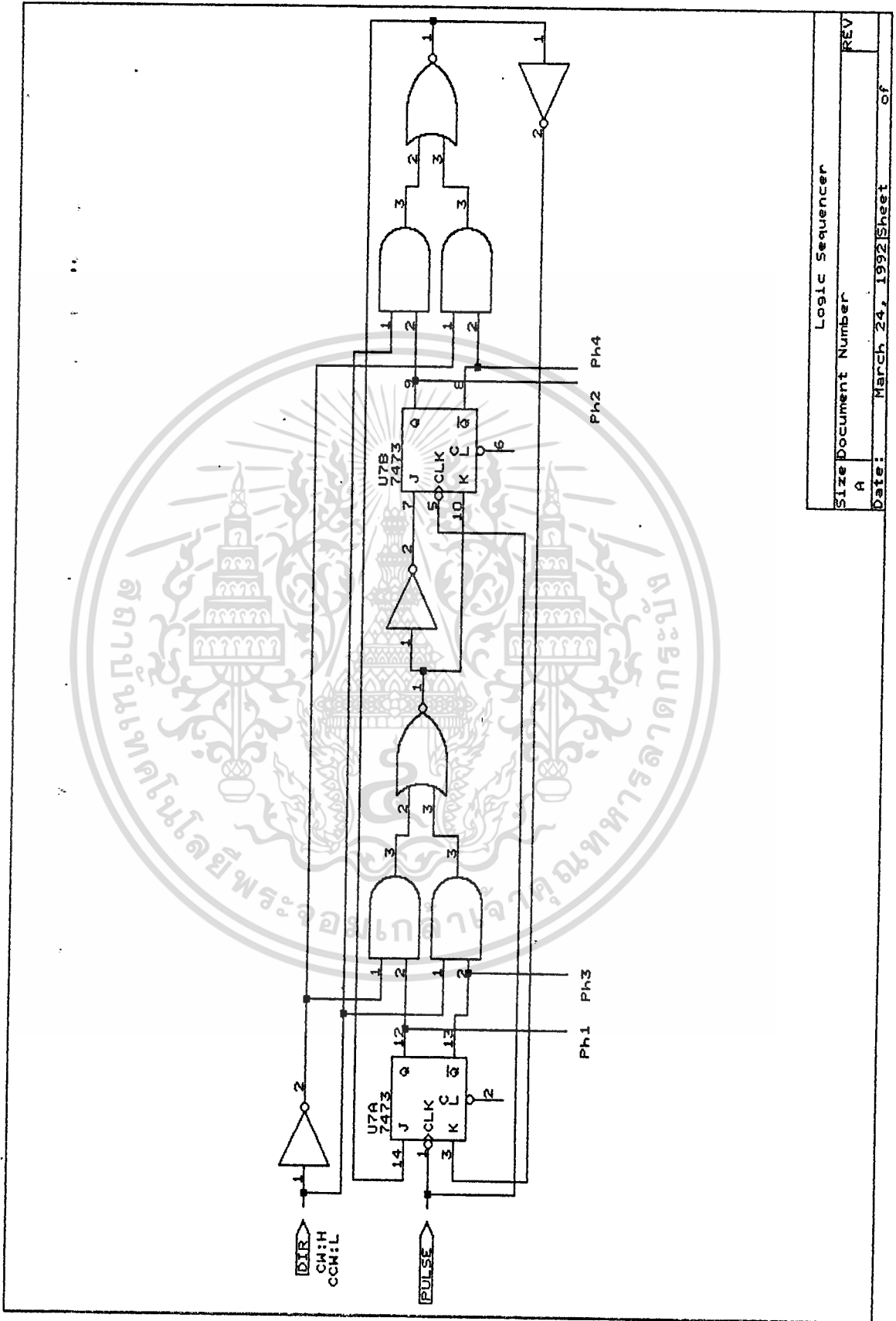
ในส่วนของ 8255 จะใช้ในการสร้างสัญญาณเพื่อควบคุมทิศทางและเป็นสัญญาณ Trigger เพื่อให้ 8253 CH ต่างๆเริ่มทำงานตามต้องการโดยเราจะใช้ Port A เพียงอันเดียวโดยบิต ต่างๆ จะทำหน้าที่ดังนี้

| บิต | หน้าที่ |
|-----|--------------------------------|
| PA0 | Trigger ให้ CH 0 8253 ตัว 1 |
| PA1 | Trigger ให้ CH 1 8253 ตัวที่ 1 |
| PA2 | ควบคุมทิศทาง Motor 1 |
| PA3 | Not used |
| PA4 | Trigger ให้ CH 0 8253 ตัว 2 |
| PA5 | Trigger ให้ CH 1 8253 ตัว 2 |
| PA6 | ควบคุมทิศทาง Motor 2 |
| PA7 | Not used |

ตาราง 4.5 แสดงหน้าที่บิตต่างๆของ Port A 8255 ในวงจร

4.3 Logic Sequencer

ส่วน Logic Sequencer เป็นส่วนที่ทำหน้าที่ควบคุมการส่ง Pulse ออกไปขับเคลื่อน ลวด Stepping Motor แต่ละเฟสอย่างสัมพันธ์กัน ให้เป็นไปตามรูปแบบการขับและทิศทางการหมุนที่ต้องการ การควบคุมความเร็วโดยความถี่ของสัญญาณ Pulse ที่ป้อนให้ โดยในโครงงานนี้เลือกใช้แบบ Bidirectional Two-Phase-On Logic Sequencer For A Four-Phase Motor โดยสัญญาณ Direction จะนำมาจาก 8255 ส่วนสัญญาณ Pulse จะนำมาจาก 8253-5 และเนื่องจากโครงงานนี้ต้องใช้ Stepping Motor 2 ตัว ดังนั้นจึงต้องใช้วงจรส่วนนี้ 2 วงจรเช่นกัน โดยลักษณะวงจรเป็นดังรูปที่ 4.6



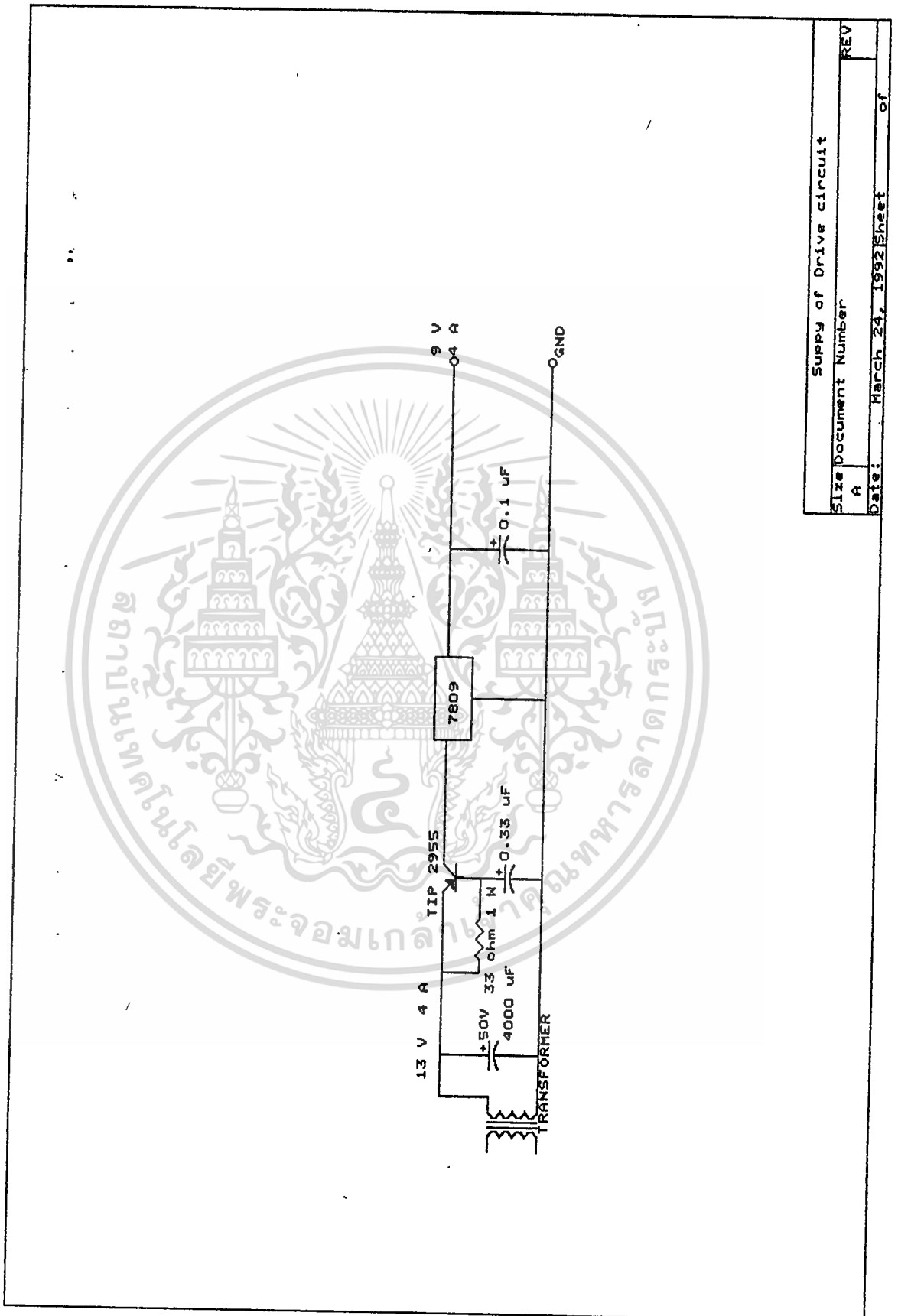
| | |
|----------------------|-------------------------|
| Logic Sequencer | |
| Size Document Number | REV |
| A | |
| Date: | March 24, 1992 Sheet of |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 Driver

วงจร Driver มีหน้าที่รับ Signal จาก Logic Sequencer มาเพื่อ On/Off Transister ให้มีการจ่ายไฟเลี้ยงในเฟสตามที่เรารต้องการโดยที่จะมี Optoisolate เพื่อทำหน้าที่ในการแยก Ground ของ Computer กับของ Supply ออกจากกันเพื่อป้องกันมิให้ Computer เสียหายถ้าหากเกิดการ Short Cot ที่ Supply การที่ต้องผ่าน Inverter ก็เพื่อให้ได้ Logic ตามที่ต้องการเนื่องจาก Optoisolate จะทำให้เกิดการกลับ Logic ในโครงงานนี้เลือกใช้ Four-Phase Driver with Condenser Suppressor โดยทำการกระตุ้นแบบ 2 เฟสเพื่อลดการสั่น (Damp) และเพิ่มแรงบิดเพื่อความแน่นอนทางตำแหน่ง แต่ทรานซิสเตอร์ใช้ต่อแบบ Darlington เพื่อช่วยในการจ่ายกระแส สำหรับ Stepping Motor ที่ใช้ในโครงงานนี้ใช้ 4 เฟส Spec 8 Volt/Phase , 0.6 9Amp/Phase วงจร Supply ที่ใช้มีขนาด 9 volt 4 Amp /Motor วงจร Drive และ Supply เป็น ดังรูปที่ 4.7

สำหรับตัวเก็บประจุที่ต่อกับ PH1 และ PH3 จะเป็นตัวป้องกันทรานซิสเตอร์ในช่วงที่มันไม่ทำงาน โดยจะเป็นตัวดูดคลื่นกระแสจากขดลวด และยังทำหน้าที่เป็น Electrical Damper ให้กระแสไหลวน Loop และเกิด Jule Heat ในขดลวด ค่าของตัวเก็บประจุยิ่งน้อย แรงบิดยิ่งมาก ในที่นี้ใช้ค่า 0.47 μF



Supply of Drive circuit

Size Document Number
A

REV

Date: March 24, 1992 Sheet of

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

Software

ในส่วนของ Software จะทำหน้าที่ควบคุมระบบ Hardware โดยส่งค่า Output ต่างๆออกไปควบคุม Motor ตาม Port ต่างๆที่ออกแบบเอาไว้แล้ว โดยลักษณะโดยทั่วไปจะเป็นการทำงานในแบบ Keystroke Control กล่าวคือเมื่อป้อนค่าต่างๆที่ต้องการเสร็จแล้วจะใช้การ Enter Key ต่างๆที่กำหนดไว้ให้ทำงานตามเงื่อนไขที่กำหนด โดยจะมีการทำงานตามโปรแกรมที่ตั้งไว้ กล่าวคือ จะมีการหมุนไปทางซ้ายหรือทางขวาตามแนวแกน X หรือหมุนไปหน้าหลังตามแนวของแกน Y ซึ่งการหมุนของ Stepping Motor แต่ละครั้งนั้นจะถูกอ้างอิงจาก Coordinate ของ X และ Y ที่จุด X และ Y มีค่าเท่ากับ 0 (จุดอ้างอิง) ด้วย โดยที่สามารถควบคุมความเร็วของ Stepping Motor แต่ละตัวจาก Software ในตอนเริ่มต้นได้ ซึ่งโดยปกติแล้ว Stepping Motor ไม่ควรมีความเร็วมากเกินไป เนื่องจากจะทำให้มีค่าความผิดพลาดต่างๆมากขึ้น ซึ่งความผิดพลาดต่างๆนั้นมาจากทางด้าน Mechanics โดยในโครงงานนี้ใช้ความเร็วที่ความถี่ประมาณ 50 Hz ซึ่งความผิดพลาดทาง Mechanics ต่างๆ (เกิดจากการ slip ของสายพาน) มีค่าไม่มากนัก ซึ่งบางส่วนจะสามารถแก้ไขได้ด้วย Software โดยสามารถตรวจสอบได้โดยการให้แทนของเครื่องเจาะเริ่มเจาะที่จุดเริ่มต้นและเจาะไปที่จุดอื่นๆ เมื่อเจาะตามจุดที่ต้องการเสร็จแล้วก็ให้กลับไปจุดเริ่มต้น พบว่าเกิดความผิดพลาดขึ้นประมาณ 1 mm ซึ่งการเช็คครั้งนี้เป็นการเช็คแบบวงรอบปิด กล่าวคือเริ่มที่จุดเริ่มต้น และเมื่อทำงานเสร็จแล้วก็ให้กลับไปจุดเริ่มต้นอีก

5.1 การใช้ File ควบคุมต่างๆ

ในการใช้งานของ File นี้ สามารถเซตจุดที่ต้องการเจาะได้มากที่สุด 250 จุด ซึ่งเหมาะสำหรับชิ้นงานขนาดไม่ใหญ่เกินไปนัก โดยจะมีการอ้างอิงตามแนวของแกน X และ Y โดยเราจะกำหนดจุดอ้างอิงให้เป็นจุดเริ่มต้นหรือ Coordinate $X = 0$, $Y = 0$ หลังจากใส่จำนวนจุดที่ต้องการเจาะแล้วก็กำหนดตำแหน่งของจุดที่จะเจาะต่างๆเป็นค่า Coordinate ของแต่ละจุดเมื่อเทียบกับจุดอ้างอิง ต่อจากนั้นเครื่องก็จะวิ่งไปหยุดรอที่จุด ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แรก หลังจากกด Enter Key แล้วเครื่องก็จะวิ่งไปรอที่จุดต่อไปจนกระทั่งหมดทุกจุดแล้ว เครื่องก็จะกลับไปยังจุดเริ่มต้นเพื่อรอการป้อนกลับค่าใหม่ต่อไป

5.2 Software

Software ที่เขียนขึ้นได้แสดงไว้ในภาคผนวกท้ายปริญญาบัตรฉบับนี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

การทดลองและผลการทดลอง

จากการทดลองพบว่า Stepping Motor1 Oscillator ที่ 1100 Hz

Stepping Motor2 Oscillator ที่ 1000 Hz

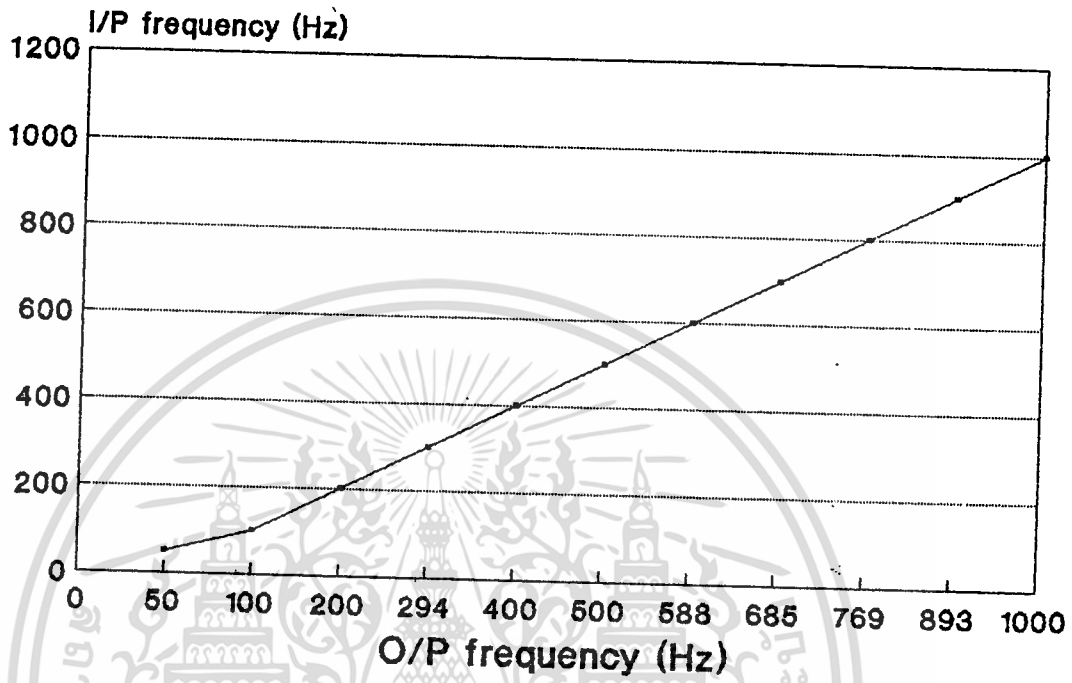
Output จาก Logic Sequencer ต้องมีค่ามากกว่า 1.8 Volt วงจร Drive จึงจะทำงาน ซึ่งจากการทดลองวัดได้ 4 Volt

โดยตารางความคลาดเคลื่อนระหว่างความถี่ที่ต้องการกับความถี่ที่วัดได้ (จาก Input Logic Sequencer) และกระแสที่จ่ายให้กับ Motor แต่ละ Phase เป็นดังนี้

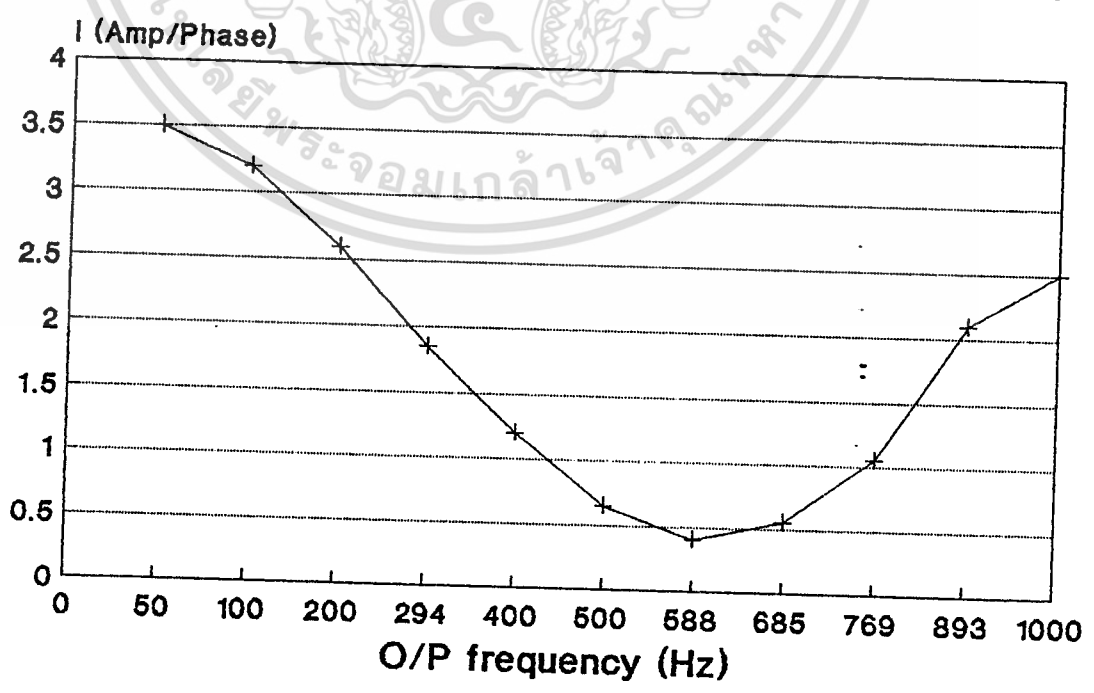
| ค่าที่ป้อนทาง Computer (Hz) | ค่าที่วัดได้จริง (Hz) | ค่าแรงดันไฟที่วัดได้ (Volt/Phase) |
|--------------------------------|--------------------------|--------------------------------------|
| 50 | 50 | 3.5 |
| 100 | 100 | 3.2 |
| 200 | 200 | 2.6 |
| 300 | 294 | 1.85 |
| 400 | 400 | 1.2 |
| 500 | 500 | 0.65 |
| 600 | 588 | 0.4 |
| 700 | 685 | 0.55 |
| 800 | 769 | 1.05 |
| 900 | 893 | 2.1 |
| 1000 | 1000 | 2.5 |

เอกสารนี้เป็นเอกสารที่ความไวสูงห้ามเผยแพร่โดยไม่ได้รับอนุญาตให้นำไปใช้ โดยอัตโนมัติในการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Relation Between O/P and I/P frequency



Relation Between O/P frequency and I(current)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7

บทสรุปและวิจารณ์

โครงการนี้ Software ที่เขียนขึ้น จะเน้นไปทางหาวิธีควบคุม Stepping Motor ให้หมุนตามระยะที่ต้องการ ซึ่งผลที่ได้นั้นจะไม่จำกัดอยู่ที่เครื่องเจาะเพียงอย่างเดียว แต่สามารถนำไปประยุกต์ใช้ต่อกับเครื่องอื่นได้อีก

สำหรับผลการทดลองของโครงการอาจมีส่วนผิดพลาดได้ ซึ่งอาจเกิดจากสาเหตุได้ 2 ส่วน คือ ทาง Software และทาง Mechnic โดยความผิดพลาดที่เกิดจาก Software นั้นคือ จากการที่ 8253 จะรับข้อมูลเฉพาะที่เป็น Integer แต่ในการคำนวณต่างๆ ผลที่ได้ไม่จำเป็นต้องเป็นค่า Integer ซึ่งทำให้มีการปัดเศษต่างๆ ทำให้ค่าเกิดการคลาดเคลื่อนได้ ส่วนทาง Mechnic เนื่องจากในโครงการนี้ส่วนที่ใช้ประกอบเข้ากับเครื่องนั้นมี 2 แกน โดยแกนหนึ่งใช้ระบบสายพาน แต่อีกแกนหนึ่งใช้เป็นตัวสวมเข้าไประหว่างแกนของเครื่องเจาะกับแกนของ Stepping Motor ซึ่งทำให้มีความหนืดแตกต่างกัน ทำให้อาจมีปัญหาเรื่อง Torque ได้ และสำหรับการใช้งานนั้นจะมีโอกาสผิดพลาดได้มากขึ้นเมื่อใช้งานที่ความเร็วมากขึ้น ทั้งนี้เพราะ Stepping Motor เมื่อใช้งานที่ความเร็วเพิ่มขึ้น Torque จะลดลง ซึ่งจะแก้ได้โดยใช้ Stepping Motor ที่มี Torque มากขึ้น หรือมีกระแสมากขึ้น สำหรับระยะที่เคลื่อนที่ถ้าต้องการค่าที่ละเอียดมากขึ้นอีก ก็สมารถทำได้โดยใช้เฟืองทดแต่จะทำให้ Torque ยิ่งลดลงไปอีก หรืออาจใช้ Stepping Motor ที่มีความละเอียดมากขึ้นก็ได้

- เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

#include<stdio.h>

#include<string.h>

#include<math.h>

main()

{

int dot,i,j;

float iy[255],ix[255];

float avgx,avgy;

int dismov;

unsigned int disx,disy;

int speed,sped;

int timer1,timer2;

int inda1,inda2;

char kkk,lop,change,copo;

kole: clrscr();

printf(" number of dot (max 255 hole):");

scanf("%9d",&dot);

printf(" speed rate (minnimum 50 Hz) :");

scanf("%9d",&speed);

printf("\n\n");

for (i=1;i<=dot;i++)

{

printf(" enter x*2d= ",i);

scanf("%f",&ix[i]);

printf(" enter y*2d= ",i);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

scanf("%f",&iy[i]);
)

avgx=0;
avgy=0;

printf("\n\n");

printf(" press (Y)es if all option is ok and press anykey to change
change = getch();

if (change != 'y')

goto kole;

for (i=1;i<=dot;i++)
(
avgx = (ix[i]-avgx);
if (avgx < 0)
inda1 = 3;
else if (avgx > 0)
inda1 = 7;
else inda1 = 0;
avgy = (iy[i]-avgy);
if (avgy < 0)

inda2 = 48;

else if (avgy > 0)

inda2 = 112;

else inda2 = 0;

printf("\n\n");

printf(" distance in x = %1fcm distance in y = %1fcm\n\n",avgx,avgy);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

disx = (fabs(avgx)*1500)-67;          /*step x to move*/
disy = (fabs(avgy)*1500)+500;        /*step y to move*/
avgx = (ix[i]);
avgy = (iy[i]);
outportb(0x303,128);
outportb(0x303,0);
sped  1200000/speed;
outportb(0x307,54);
outportb(0x304,(sped%256));
outportb(0x304,(sped/256));
outportb(0x30b,54);
outportb(0x308,(sped%256));
outportb(0x308,(sped/256));
outportb(0x307,114);
outportb(0x305,(disx%256));
outportb(0x305,(disx/256));
outportb(0x30b,114);
outportb(0x309,(disy%256));
outportb(0x309,(disy/256));
printf(" press (S) to start to move\n");
man: lop = getch();
if (lop == 's')
outportb(0x300,inda1+inda2);
else goto man;
timer1 = (disx/speed)+1000;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

timer2 = (disy/speed)*1000;

if (timer1 < timer2)

delay(timer2);

else

delay(timer1);

if (i<dot)

{

clrscr();

gotoxy(1,5);

printf("press (C)ontinue to move next position\n\n");

printf("press (E)xit to out\n");

yui: kkk = getch();

switch(kkk)

{

case 'c' : break;

case 'e' : ( clrscr(); exit(); break;);

default : goto yui;

}

}

}

printf("\n\n");

printf(" press (A)gain to set data again\n\n");

printf(" press (Q)uit to out\n");

wert: copo = getch();

switch(copo)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
(  
case 'q' : ( clrscr(); exit(); break; )  
case 'a' : goto kole;  
default : goto wert;  
)  
)
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.



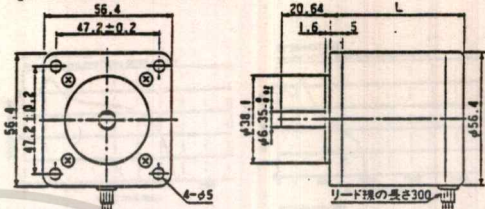
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

STEPPING MOTORS HB(HYBRID)type

●ステッピングモータ HB(ハイブリッド)型

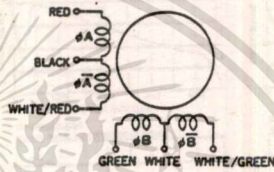
●外形寸法 DIMENSIONS(mm)

Fig.9 KP56RM, KP56KM, KP56LM, KP56RM



●結線方法 CONNECTION DIAGRAM

Fig.33



●標準仕様 STANDARD SPECIFICATIONS

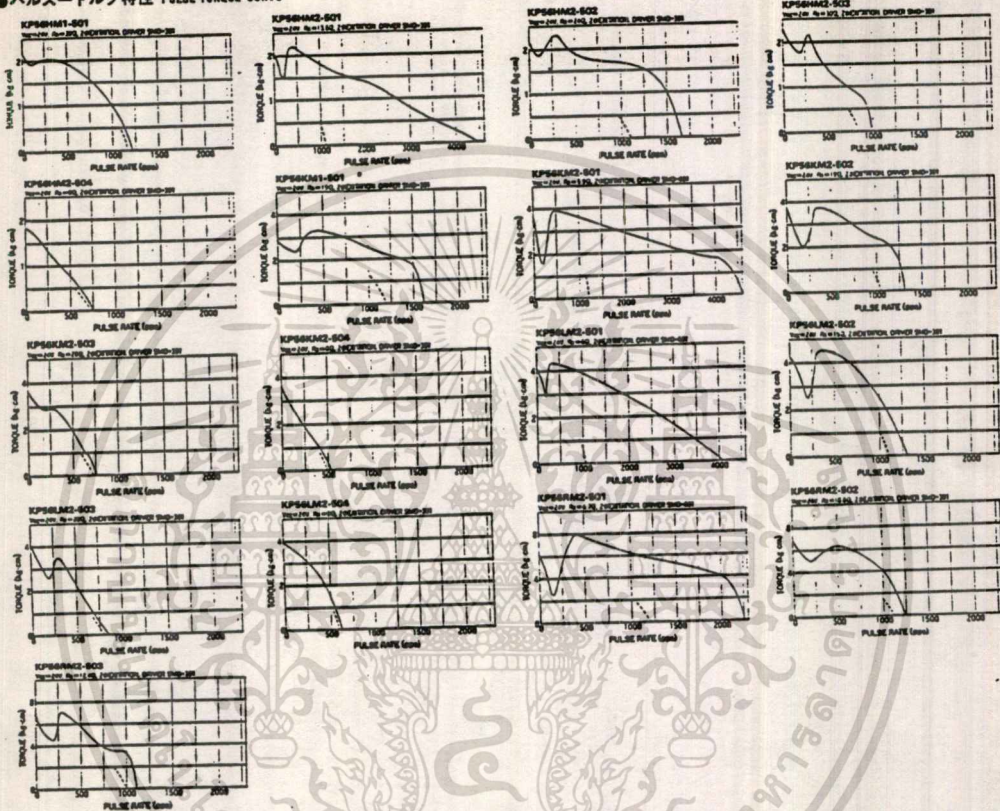
| 機種名 MODEL | KP56RM1 | KP56RM2 | | | | KP56KM1 | KP56KM2 | | | |
|--|--------------------|---------|------|------|------|--------------------|---------|------|------|------|
| | -501 | -501 | -502 | -503 | -504 | -501 | -501 | -502 | -503 | -504 |
| 相数 PHASE | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| ステップ角 STEP ANGLE deg STEP | 0.9 | 1.8 | 1.8 | 1.8 | 1.8 | 0.9 | 1.8 | 1.8 | 1.8 | 1.8 |
| 電圧 VOLTAGE V | 12 | 2.25 | 6 | 12 | 24 | 5 | 1.4 | 5 | 12 | 24 |
| 電流 CURRENT A PHASE | 0.4 | 1.6 | 0.75 | 0.4 | 0.2 | 1.0 | 3.8 | 1.0 | 0.43 | 0.21 |
| 巻線抵抗 RESISTANCE Ω/PHASE | 30 | 1.4 | 8 | 30 | 120 | 5 | 0.37 | 5 | 28 | 115 |
| インダクタンス INDUCTANCE mH/PHASE | 40 | 1.45 | 7 | 30 | 95 | 8.5 | 0.63 | 9 | 44 | 167 |
| 最大静止トルク HOLDING TORQUE kgf·cm | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 |
| ディテントトルク DETENT TORQUE gf·cm | 300 | 300 | 300 | 300 | 300 | 300 | 450 | 450 | 450 | 450 |
| ロータイナーシャ ROTOR INERTIA g·cm ² | 60 | 60 | 60 | 60 | 60 | 110 | 110 | 110 | 110 | 110 |
| 重量 WEIGHT kg | 0.35 | 0.35 | 0.35 | 0.35 | 0.35 | 0.5 | 0.5 | 0.5 | 0.5 | 0.5 |
| 外形寸法 DIMENSIONS | Fig. 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 |
| L寸法 L LENGTH mm | 38 | 38 | 38 | 38 | 38 | 50.8 | 50.8 | 50.8 | 50.8 | 50.8 |
| 結線方法 CONNECTION DIAGRAM | Fig. 33 | 33 | 33 | 33 | 33 | 33 | 33 | 33 | 33 | 33 |
| 絶縁階級 INSULATION CLASS | B | | | | | B | | | | |
| 絶縁抵抗 INSULATION RESISTANCE | 500V DC 100M Ω min | | | | | 500V DC 100M Ω min | | | | |
| 絶縁耐力 DIELECTRIC STRENGTH | 500V 50Hz 1min. | | | | | 500V 50Hz 1min. | | | | |
| 使用温度範囲 AMBIENT TEMPERATURE °C | -10-50 | | | | | -10-50 | | | | |
| 温度上昇 TEMPERATURE RISE deg. | 80 | | | | | 80 | | | | |
| リード線仕様 LEADS SPECIFICATION | AWG #22 UL3266 | | | | | AWG #22 UL3266 | | | | |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

STEPPING MOTORS HB(HYBRID)type

●ステッピングモータ HB(ハイブリッド)型

●パルストルク特性 PULSE-TORQUE CURVE



●標準仕様 STANDARD SPECIFICATIONS

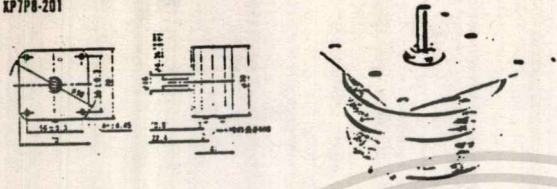
| 機種名 MODEL | KPS6LM2 | | | | KPS6RM2 | | |
|--|--------------------|------|------|------|---------|------|------|
| | -501 | -502 | -503 | -504 | -501 | -502 | -503 |
| 相数 PHASE | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| ステップ角 STEP ANGLE deg. STEP | 1.8 | 1.8 | 1.8 | 1.8 | 1.8 | 1.8 | 1.8 |
| 電圧 VOLTAGE V | 1.8 | 6 | 12 | 24 | 2.16 | 4.5 | 5.4 |
| 電流 CURRENT A. PHASE | 3.6 | 1.2 | 0.6 | 0.3 | 3.6 | 1.8 | 1.5 |
| 巻線抵抗 RESISTANCE Q / PHASE | 0.5 | 5 | 20 | 80 | 0.6 | 2.5 | 3.6 |
| インダクタンス INDUCTANCE mH / PHASE | 0.7 | 8.5 | 35 | 120 | 1.25 | 5 | 7 |
| 最大静止トルク HOLDING TORQUE kgf·cm | 5.5 | 5.5 | 5.5 | 5.5 | 8.0 | 8.0 | 8.0 |
| ディテントトルク DETENT TORQUE gf·cm | 450 | 450 | 450 | 450 | 750 | 750 | 750 |
| ロータイナーシャ Rotor INERTIA g·cm ² | 110 | 110 | 110 | 110 | 230 | 230 | 230 |
| 重量 WEIGHT kg | 0.6 | 0.6 | 0.6 | 0.6 | 0.95 | 0.95 | 0.95 |
| 外形寸法 DIMENSIONS | Fig. 9 | 9 | 9 | 9 | 9 | 9 | 9 |
| | L寸法 L LENGTH mm | 57 | 57 | 57 | 57 | 80 | 80 |
| 接続方法 CONNECTION DIAGRAM | Fig. 33 | 33 | 33 | 33 | 33 | 33 | 33 |
| 絶縁階級 INSULATION CLASS | B | | | | | | |
| 絶縁抵抗 INSULATION RESISTANCE | 500V DC 100M Q min | | | | | | |
| 絶縁耐力 DIELECTRIC STRENGTH | 500V 50Hz 1min. | | | | | | |
| 使用温度範囲 AMBIENT TEMPERATURE °C | -10-50 | | | | | | |
| 温度上昇 TEMPERATURE RISE deg. | 80 | | | | | | |
| リード線仕様 LEADS SPECIFICATION | AWG #22 UL3266 | | | | | | |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

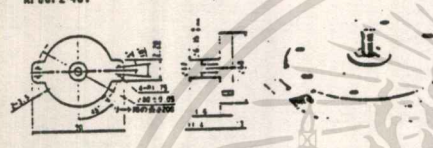
STEPPING MOTORS PM(PERMANENT MAGNET),VR(VARIABLE RELUCTANCE)type

●ステッピングモータ PM、VR型

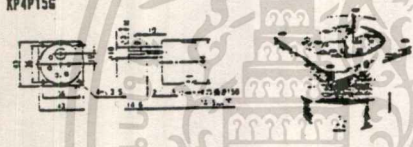
●外形寸法 DIMENSIONS(mm)
KP7P8-201



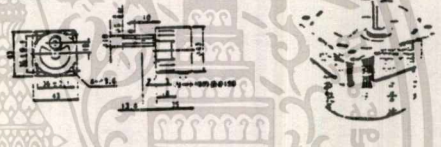
KP6P2-401



KP4P15G



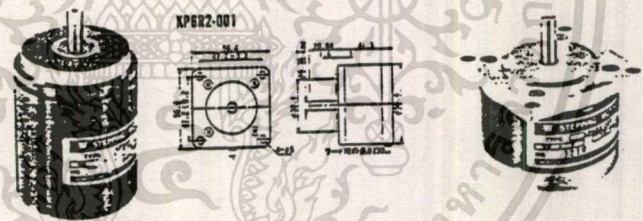
KP4P18GN



KP5AR15-2



KP6R2-001



●標準仕様 STANDARD SPECIFICATIONS

| 機種名 MODEL | KP7P8-201 | KP6P2-401 | KP4P15G | KP4P18GN |
|--|-------------------|----------------|------------|----------|
| 相数 PHASE | 4 | 4 | 4 | 4 |
| ステップ角 STEP ANGLE deg./STEP | 7.5 | 1.8 | 15 | 18 |
| 電圧 VOLTAGE V | 12 | 12 | 12 | 12 |
| 電流 CURRENT A/PHASE | 0.64 | 0.36 | 0.18 | 0.17 |
| 巻線抵抗 RESISTANCE Ω/PHASE | 18.8 | 33 | 66 | 70 |
| インダクタンス INDUCTANCE mH/PHASE | 30.5 | 18 | 27 | 37 |
| 最大静止トルク HOLDING TORQUE gf·cm | 3800 | 600 | 100 | 150 |
| ディテントトルク DETENT TORQUE gf·cm | 520 | 110 | 50 | 50 |
| ロータイナーシャ Rotor INERTIA g·cm ² | 194 | 53.8 | 3.44 | 5 |
| 重量 WEIGHT kg | 0.65 | 0.18 | 0.15 | 0.15 |
| 絶縁階級 INSULATION CLASS | A | A | A | A |
| 絶縁抵抗 INSULATION RESISTANCE | 500V DC 100MΩ min | | | |
| 絶縁耐力 DIELECTRIC STRENGTH | 500V 50Hz 1min. | | | |
| 使用温度範囲 AMBIENT TEMPERATURE °C | -10-40 | -10-50 | -10-40 | -10-45 |
| 温度上昇 TEMPERATURE RISE deg. | 65 | 50 | 65 | 60 |
| リード線仕様 LEADS SPECIFICATION | AWG #22 UL3266 | AWG #26 UL1007 | 10/0.12PVC | |

●ギヤヘッド

出力常用トルク
KP4P15G(at 100pps)

| 減速比 | トルク |
|-------|------|
| 1/10 | 0.33 |
| 1/20 | 0.55 |
| 1/30 | 0.87 |
| 1/50 | 1.37 |
| 1/60 | 1.5 |
| 1/75 | 1.5 |
| 1/100 | 1.5 |
| 1/150 | 1.5 |
| 1/200 | 1.5 |
| 1/300 | 1.5 |
| 1/500 | 1.5 |
| 1/600 | 1.5 |

KP4P18GN

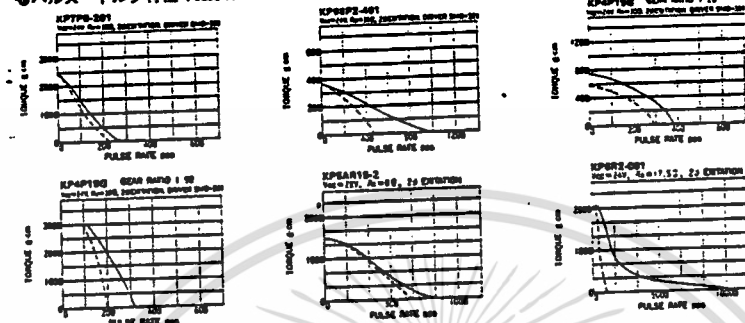
| 減速比 | トルク |
|------|-----|
| 1/90 | 3.0 |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

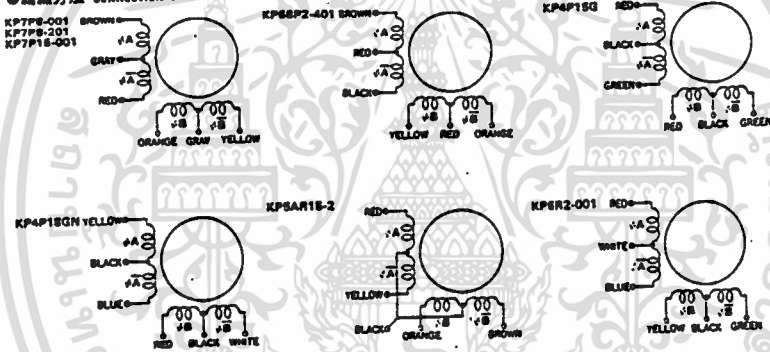
STEPPING MOTORS PM(PERMANENT MAGNET),VR(VARIABLE RELUCTANCE)type

●ステッピングモータ PM、VR型

●パルストルク特性 PULSE-TORQUE CURVE



●結線方法 CONNECTION DIAGRAM



●標準仕様 STANDARD SPECIFICATIONS

| 機種名 MODEL | KP5A15 -2 | KP6R2 -001 |
|--|-------------------|----------------|
| 相数 PHASE | 4 | 4 |
| ステップ角 STEP ANGLE deg. STEP | 15 | 1.8 |
| 電圧 VOLTAGE V | 28 | 4 |
| 電流 CURRENT A/PHASE | 1.4 | 1.14 |
| 巻線抵抗 RESISTANCE Ω/PHASE | 20 | 3.5 |
| インダクタンス INDUCTANCE mH/PHASE | — | — |
| 最大静止トルク HOLDING TORQUE kgf·cm | 2 | 2 |
| ロータイナーシャ ROTOR INERTIA g·cm ² | 10 | 45 |
| 質量 WEIGHT kg | 0.56 | 0.35 |
| 絶縁階級 INSULATION CLASS | H | B |
| 絶縁抵抗 INSULATION RESISTANCE | 500V DC 100MΩ min | |
| 絶縁耐力 DIELECTRIC STRENGTH | 500V 50Hz 1min. | |
| 使用温度範囲 AMBIENT TEMPERATURE °C | -10-50°C | |
| 温度上昇 TEMPERATURE RISE deg. | 許容温度100°C | 80°C |
| リード線仕様 LEADS SPECIFICATION | 7 0.12TEF | AWG #26 UL3256 |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| Hex Range | Usage |
|-----------------------|---|
| 000-00F | DMA Chip 8237A-5 |
| 020-021 | Interrupt 8259A |
| 040-043 | Timer 8253-5 |
| 060-063 | PPI 8255A-5 |
| 080-083 | DMA Page Registers |
| 0A [*] | NMI Mask Register |
| 0C ^X | Reserved |
| 0E ^X | Reserved |
| 200-20F | Game Control |
| 210-217 | Expansion Unit |
| 220-24F | Reserved |
| 278-27F | Reserved |
| 2F0-2F7 | Reserved |
| 2F8-2FF | Asynchronous Communications (Secondary) |
| 300-31F | Prototype Card |
| 320-32F | Fixed Disk |
| 378-37F | Printer |
| 380-38C ^{**} | SDLC Communications |
| 380-389 ^{**} | Binary Synchronous Communications (Secondary) |
| 3A0-3A9 | Binary Synchronous Communications (Primary) |
| 3B0-3BF | IBM Monochrome Display/Printer |
| 3C0-3CF | Reserved |
| 3D0-3DF | Color/Graphics |
| 3E0-3E7 | Reserved |
| 3F0-3F7 | Diskette |
| 3F8-3FF | Asynchronous Communications (Primary) |

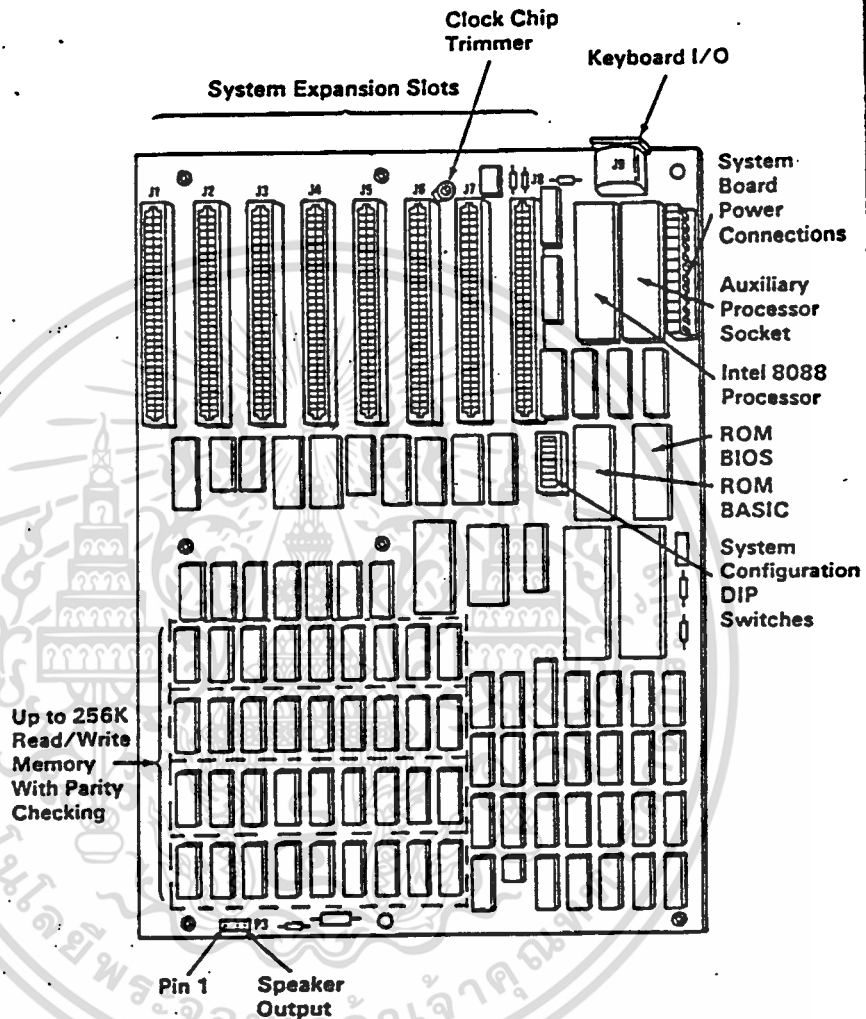
^{*} At power-on time, the Non Mask Interrupt into the 8088 is masked off. This mask bit can be set and reset through system software as follows:
 Set mask: Write hex 80 to I/O Address hex A0 (enable NMI)
 Clear mask: Write hex 00 to I/O Address hex A0 (disable NMI)

^{**} SDLC Communications and Secondary Binary Synchronous Communications cannot be used together because their hex addresses overlap.

I/O Address Map

1-8 System Unit

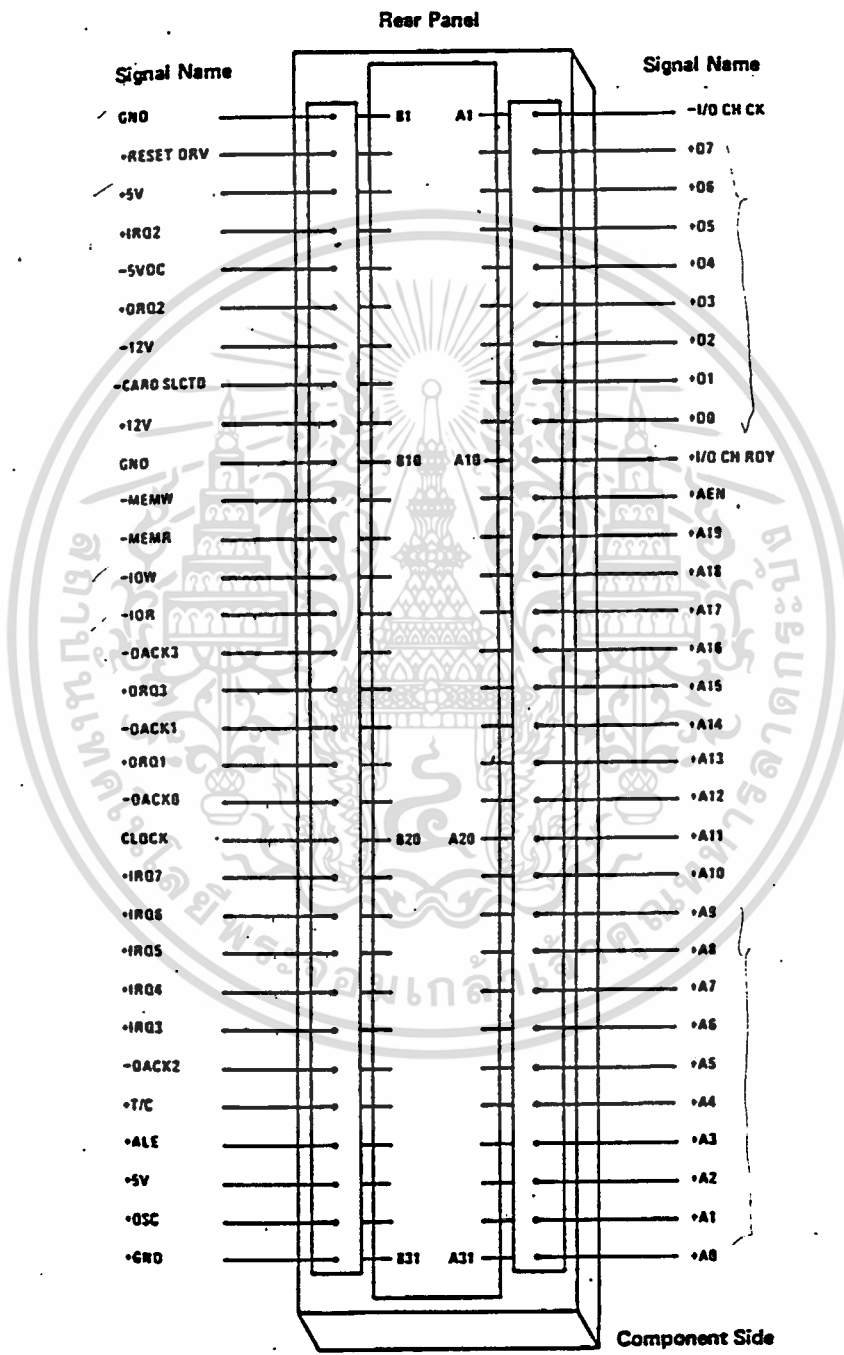
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



System Board Component Diagram

System Board Switch Settings

All system board switch settings for total system memory, number of diskette drives, and type of display are located in "Appendix G: Switch Settings."



I/O Channel Diagram

I/O Channel Description

The following is a description of the IBM Personal Computer XT I/O Channel. All lines are TTL-compatible.

| Signal | I/O | Description |
|-----------|-----|--|
| OSC | O | Oscillator: High-speed clock with a 70-ns period (14.31818 MHz). It has a 50% duty cycle. |
| CLK | O | System clock: It is a divide-by-three of the oscillator and has a period of 210 ns (4.77 MHz). The clock has a 33% duty cycle. |
| RESET DRV | O | This line is used to reset or initialize system logic upon power-up or during a low line voltage outage. This signal is synchronized to the falling edge of clock and is active high. |
| A0-A19 | O | Address bits 0 to 19: These lines are used to address memory and I/O devices within the system. The 20 address lines allow access of up to 1 megabyte of memory. A0 is the least significant bit (LSB) and A19 is the most significant bit (MSB). These lines are generated by either the processor or DMA controller. They are active high. |
| D0-D7 | I/O | Data Bits 0 to 7: These lines provide data bus bits 0 to 7 for the processor, memory, and I/O devices. D0 is the least significant bit (LSB) and D7 is the most significant bit (MSB). These lines are active high. |
| ALE | O | Address Latch Enable: This line is provided by the 8288 Bus Controller and is used on the system board to latch valid addresses from the processor. It is available to the I/O channel as an indicator of a valid processor address (when used with AEN). Processor addresses are latched with the falling edge of ALE. |

1-16 System Unit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| Signal | I/O | Description |
|-------------------------------|-----|---|
| $\overline{\text{I/O CH CK}}$ | I | -I/O Channel Check: This line provides the processor with parity (error) information on memory or devices in the I/O channel. When this signal is active low, a parity error is indicated. |
| I/O CH RDY | I | I/O Channel Ready: This line, normally high (ready), is pulled low (not ready) by a memory or I/O device to lengthen I/O or memory cycles. It allows slower devices to attach to the I/O channel with a minimum of difficulty. Any slow device using this line should drive it low immediately upon detecting a valid address and a read or write command. This line should never be held low longer than 10 clock cycles. Machine cycles (I/O or memory) are extended by an integral number of CLK cycles (210 ns). |
| IRQ2-IRQ7 | I | Interrupt Request 2 to 7: These lines are used to signal the processor that an I/O device requires attention. They are prioritized with IRQ2 as the highest priority and IRQ7 as the lowest. An Interrupt Request is generated by raising an IRQ line (low to high) and holding it high until it is acknowledged by the processor (interrupt service routine). |
| $\overline{\text{IOR}}$ | O | -I/O Read Command: This command line instructs an I/O device to drive its data onto the data bus. It may be driven by the processor or the DMA controller. This signal is active low. |
| $\overline{\text{IOW}}$ | O | -I/O Write Command: This command line instructs an I/O device to read the data on the data bus. It may be driven by the processor or the DMA controller. This signal is active low. |

| Signal | I/O | Description |
|--|-----|---|
| $\overline{\text{MEMR}}$ | O | Memory Read Command: This command line instructs the memory to drive its data onto the data bus. It may be driven by the processor or the DMA controller. This signal is active low. |
| $\overline{\text{MEMW}}$ | O | Memory Write Command: This command line instructs the memory to store the data present on the data bus. It may be driven by the processor or the DMA controller. This signal is active low. |
| DRQ1-DRQ3 | I | DMA Request 1 to 3: These lines are asynchronous channel requests used by peripheral devices to gain DMA service. They are prioritized with DRQ3 being the lowest and DRQ1 being the highest. A request is generated by bringing a DRQ line to an active level (high). A DRQ line must be held high until the corresponding DACK line goes active. |
| $\overline{\text{DACK0}}$ $\overline{\text{DACK3}}$ | O | -DMA Acknowledge 0 to 3: These lines are used to acknowledge DMA requests (DRQ1-DRQ3) and to refresh system dynamic memory (DACK0). They are active low. |
| AEN | O | Address Enable: This line is used to de-gate the processor and other devices from the I/O channel to allow DMA transfers to take place. When this line is active (high), the DMA controller has control of the address bus, data bus, read command lines (memory and I/O), and the write command lines (memory and I/O). |
| T/C | O | Terminal Count: This line provides a pulse when the terminal count for any DMA channel is reached. This signal is active high. |

1-18 System Unit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| Signal | I/O Description |
|---------------------|--|
| <u>CARD SLCTD</u> I | -Card Selected: This line is activated by cards in expansion slot J8. It signals the system board that the card has been selected and that appropriate drivers on the system board should be directed to either read from, or write to, expansion slot J8. Connectors J1 through J8 are tied together at this pin, but the system board does not use their signal. This line should be driven by an open collector device. |

The following voltages are available on the system board I/O channel:

- +5 Vdc \pm 5%, located on 2 connector pins
- 5 Vdc \pm 10%, located on 1 connector pin
- +12 Vdc \pm 5%, located on 1 connector pin
- 12 Vdc \pm 10%, located on 1 connector pin
- GND (Ground), located on 3 connector pins



8253/8253-5 PROGRAMMABLE INTERVAL TIMER

- MCS-85™ Compatible 8253-5
- 3 Independent 16-Bit Counters
- DC to 2.6 MHz
- Programmable Counter Modes
- Count Binary or BCD
- Single +5V Supply
- Available in EXPRESS.
 - Standard Temperature Range
 - Extended Temperature Range

The Intel® 8253 is a programmable counter/timer device designed for use as an Intel microcomputer peripheral. It uses nMOS technology with a single +5V supply and is packaged in a 24-pin plastic DIP. It is organized as 3 independent 16-bit counters, each with a count rate of up to 2.6 MHz. All modes of operation are software programmable.

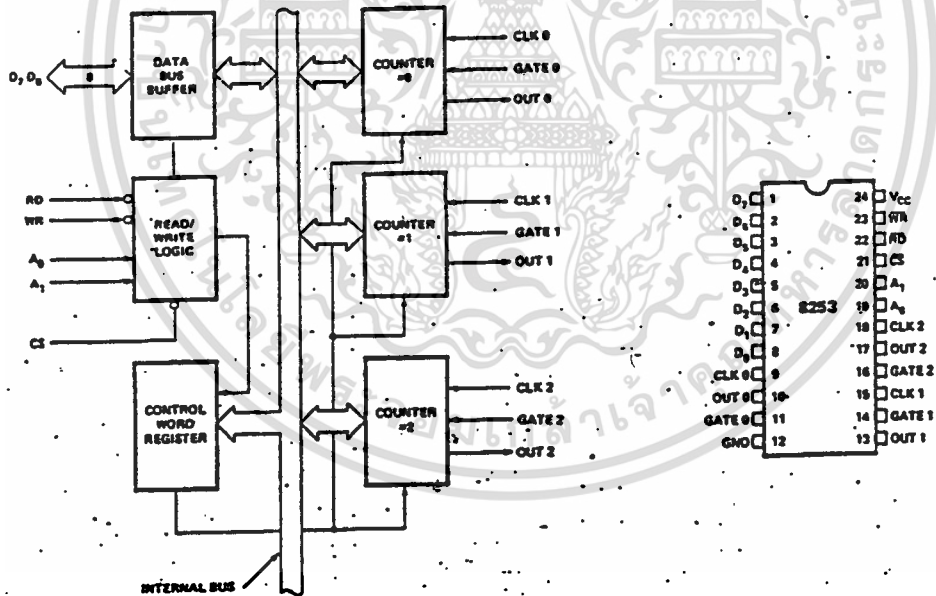


Figure 1. Block Diagram

Figure 2. Pin Configuration

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FUNCTIONAL DESCRIPTION

General

The 8253 is a programmable interval timer/counter specifically designed for use with the Intel™ Micro-computer systems. Its function is that of a general purpose, multi-timing element that can be treated as an array of I/O ports in the system software.

The 8253 solves one of the most common problems in any microcomputer system, the generation of accurate time delays under software control. Instead of setting up timing loops in systems software, the programmer configures the 8253 to match his requirements, initializes one of the counters of the 8253 with the desired quantity, then upon command the 8253 will count out the delay and interrupt the CPU when it has completed its tasks. It is easy to see that the software overhead is minimal and that multiple delays can easily be maintained by assignment of priority levels.

Other counter/timer functions that are non-delay in nature but also common to most microcomputers can be implemented with the 8253.

- Programmable Rate Generator
- Event Counter
- Binary Rate Multiplier
- Real Time Clock
- Digital One-Shot
- Complex Motor Controller

Data Bus Buffer

This 3-state, bi-directional, 8-bit buffer is used to interface the 8253 to the system data bus. Data is transmitted or received by the buffer upon execution of INput or OUTput CPU instructions. The Data Bus Buffer has three basic functions.

1. Programming the MODES of the 8253.
2. Loading the count registers.
3. Reading the count values.

Read/Write Logic

The Read/Write Logic accepts inputs from the system bus and in turn generates control signals for overall device operation. It is enabled or disabled by CS so that no operation can occur to change the function unless the device has been selected by the system logic.

\overline{RD} (Read)

A "low" on this input informs the 8253 that the CPU is inputting data in the form of a counters value.

\overline{WR} (Write)

A "low" on this input informs the 8253 that the CPU is outputting data in the form of mode information or loading counters.

A0, A1

These inputs are normally connected to the address bus. Their function is to select one of the three counters to be operated on and to address the control word register for mode selection.

\overline{CS} (Chip Select)

A "low" on this input enables the 8253. No reading or writing will occur unless the device is selected. The \overline{CS} input has no effect upon the actual operation of the counters.

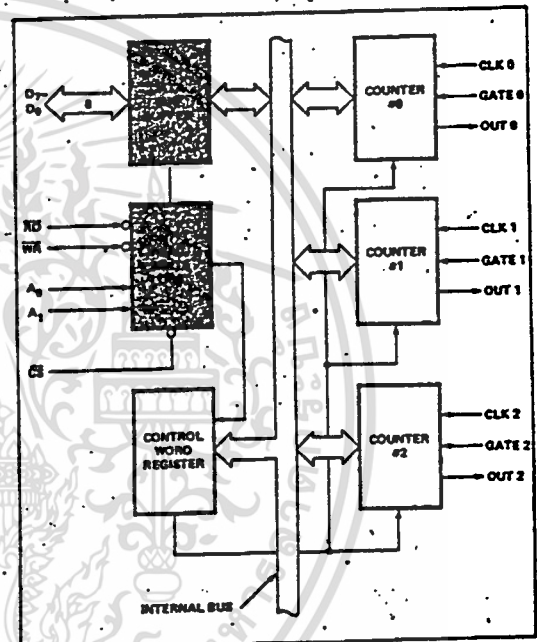


Figure 3. Block Diagram Showing Data Bus Buffer and Read/Write Logic Functions

| CS | \overline{RD} | \overline{WR} | A ₁ | A ₀ | |
|----|-----------------|-----------------|----------------|----------------|----------------------|
| 0 | 1 | 0 | 0 | 0 | Load Counter No. 0 |
| 0 | 1 | 0 | 0 | 1 | Load Counter No. 1 |
| 0 | 1 | 0 | 1 | 0 | Load Counter No. 2 |
| 0 | 1 | 0 | 1 | 1 | Write Mode Word |
| 0 | 0 | 1 | 0 | 0 | Read Counter No. 0 |
| 0 | 0 | 1 | 0 | 1 | Read Counter No. 1 |
| 0 | 0 | 1 | 1 | 0 | Read Counter No. 2 |
| 0 | 0 | 1 | 1 | 1 | No-Operation 3-State |
| 1 | X | X | X | X | Disable 3-State |
| 0 | 1 | 1 | X | X | No-Operation 3-State |

Control Word Register

The Control Word Register is selected when A0, A1 are 11. It then accepts information from the data bus buffer and stores it in a register. The information stored in this register controls the operational MODE of each counter, selection of binary or BCD counting and the loading of each count register.

The Control Word Register can only be written into; no read operation of its contents is available.

Counter #0, Counter #1, Counter #2

These three functional blocks are identical in operation so only a single Counter will be described. Each Counter consists of a single, 16-bit, pre-settable, DOWN counter. The counter can operate in either binary or BCD and its input, gate and output are configured by the selection of MODES stored in the Control Word Register.

The counters are fully independent and each can have separate Mode configuration and counting operation, binary or BCD. Also, there are special features in the control word that handle the loading of the count value so that software overhead can be minimized for these functions.

The reading of the contents of each counter is available to the programmer with simple READ operations for event counting applications and special commands and logic are included in the 8253 so that the contents of each counter can be read "on the fly" without having to inhibit the clock input.

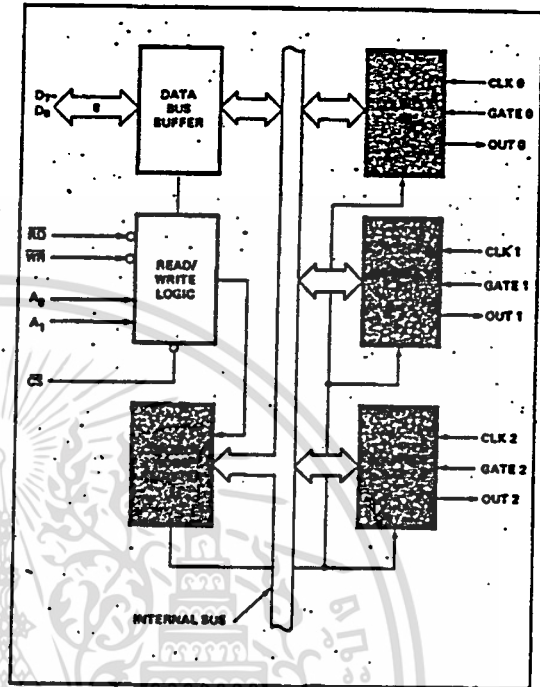


Figure 4. Block Diagram Showing Control Word Register and Counter Functions

8253 SYSTEM INTERFACE

The 8253 is a component of the Intel™ Microcomputer Systems and interfaces in the same manner as all other peripherals of the family. It is treated by the systems software as an array of peripheral I/O ports; three are counters and the fourth is a control register for MODE programming.

Basically, the select inputs A0, A1 connect to the A0, A1 address bus signals of the CPU. The CS can be derived directly from the address bus using a linear select method. Or it can be connected to the output of a decoder, such as an Intel® 8205 for larger systems.

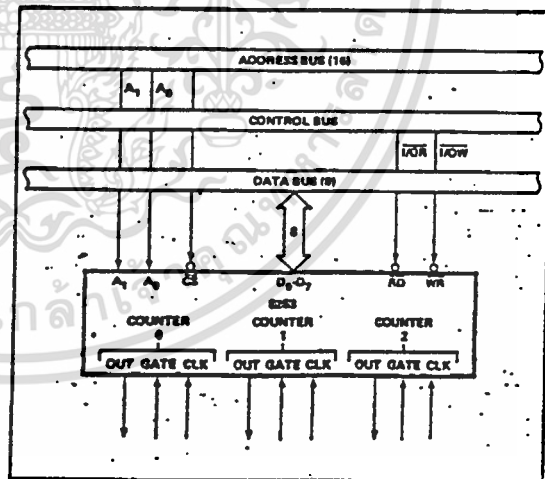


Figure 5. 8253 System Interface

OPERATIONAL DESCRIPTION

General

The complete functional definition of the 8253 is programmed by the systems software. A set of control words must be sent out by the CPU to initialize each counter of the 8253 with the desired MODE and quantity information. Prior to initialization, the MODE, count, and output of all counters is undefined. These control words program the MODE, Loading sequence and selection of binary or BCD counting.

Once programmed, the 8253 is ready to perform whatever timing tasks it is assigned to accomplish.

The actual counting operation of each counter is completely independent and additional logic is provided on-chip so that the usual problems associated with efficient monitoring and management of external, asynchronous events or rates to the microcomputer system have been eliminated.

Programming the 8253

All of the MODES for each counter are programmed by the systems software by simple I/O operations.

Each counter of the 8253 is individually programmed by writing a control word into the Control Word Register. (A0, A1 = 11)

Control Word Format

| | | | | | | | |
|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| D ₇ | D ₆ | D ₅ | D ₄ | D ₃ | D ₂ | D ₁ | D ₀ |
| SC1 | SC0 | RL1 | RL0 | M2 | M1 | M0 | BCD |

Definition of Control

SC — Select Counter:

| | | | | |
|-----|---|-----|---|------------------|
| SC1 | | SC0 | | |
| 0 | 0 | 0 | 0 | Select Counter 0 |
| 0 | 1 | 0 | 1 | Select Counter 1 |
| 1 | 0 | 1 | 0 | Select Counter 2 |
| 1 | 1 | 1 | 1 | Illegal |

RL — Read/Load:

| | | | | |
|-----|---|-----|---|---|
| RL1 | | RL0 | | |
| 0 | 0 | 0 | 0 | Counter Latching operation (see READ/WRITE Procedure Section) |
| 1 | 0 | 0 | 1 | Read/Load most significant byte only. |
| 0 | 1 | 1 | 0 | Read/Load least significant byte only. |
| 1 | 1 | 1 | 1 | Read/Load least significant byte first, then most significant byte. |

M — MODE:

| | | | |
|----|----|----|--------|
| M2 | M1 | M0 | |
| 0 | 0 | 0 | Mode 0 |
| 0 | 0 | 1 | Mode 1 |
| X | 1 | 0 | Mode 2 |
| X | 1 | 1 | Mode 3 |
| 1 | 0 | 0 | Mode 4 |
| 1 | 0 | 1 | Mode 5 |

BCD:

| | |
|---|--|
| 0 | Binary Counter 16-bits |
| 1 | Binary Coded Decimal (BCD) Counter (4 Decades) |

Counter Loading

The count register is not loaded until the count value is written (one or two bytes, depending on the mode selected by the RL bits), followed by a rising edge and a falling edge of the clock. Any read of the counter prior to that falling clock edge may yield invalid data.

MODE Definition

MODE 0: Interrupt on Terminal Count. The output will be initially low after the mode set operation. After the count is loaded into the selected count register, the output will remain low and the counter will count. When terminal count is reached the output will go high and remain high until the selected count register is reloaded with the mode or a new count is loaded. The counter continues to decrement after terminal count has been reached.

Rewriting a counter register during counting results in the following:

- (1) Write 1st byte stops the current counting.
- (2) Write 2nd byte starts the new count.

MODE 1: Programmable One-Shot. The output will go low on the count following the rising edge of the gate input.

The output will go high on the terminal count. If a new count value is loaded while the output is low it will not affect the duration of the one-shot pulse until the succeeding trigger. The current count can be read at any time without affecting the one-shot pulse.

The one-shot is retriggerable, hence the output will remain low for the full count after any rising edge of the gate input.

MODE 2: Rate Generator. Divide by N counter. The output will be low for one period of the input clock. The period from one output pulse to the next equals the number of input counts in the count register. If the count register is reloaded between output pulses the present period will not be affected, but the subsequent period will reflect the new value.

The gate input, when low, will force the output high. When the gate input goes high, the counter will start from the initial count. Thus, the gate input can be used to synchronize the counter.

When this mode is set, the output will remain high until after the count register is loaded. The output then can also be synchronized by software.

MODE 3: Square Wave Rate Generator. Similar to MODE 2 except that the output will remain high until one half the count has been completed (for even numbers) and go low for the other half of the count. This is accomplished by decrementing the counter by two on the falling edge of each clock pulse. When the counter reaches terminal count, the state of the output is changed and the counter is reloaded with the full count and the whole process is repeated.

If the count is odd and the output is high, the first clock pulse (after the count is loaded) decrements the count by 1. Subsequent clock pulses decrement the clock by 2. After timeout, the output goes low and the full count is reloaded. The first clock pulse (following the reload) decrements the counter by 3. Subsequent clock pulses decrement the count by 2 until timeout. Then the whole process is repeated. In this way, if the count is odd, the output will be high for $(N + 1)/2$ counts and low for $(N - 1)/2$ counts.

In Modes 2 and 3, if a CLK source other than the system clock is used, GATE should be pulsed immediately following WR of a new count value.

MODE 4: Software Triggered Strobe. After the mode is set, the output will be high. When the count is loaded, the counter will begin counting. On terminal count, the

output will go low for one input clock period, then will go high again.

If the count register is reloaded during counting, the new count will be loaded on the next CLK pulse. The count will be inhibited while the GATE input is low.

MODE 5: Hardware Triggered Strobe. The counter will start counting after the rising edge of the trigger input and will go low for one clock period when the terminal count is reached. The counter is retriggerable. The output will not go low until the full count after the rising edge of any trigger.

| Signal Status | Low Or Going Low | Rising | High |
|---------------|---|--|------------------|
| 0 | Disables counting | --- | Enables counting |
| 1 | --- | 1) Initiates counting 2) Resets output after next clock | --- |
| 2 | 1) Disables counting 2) Sets output immediately high | 1) Reloads counter 2) Initiates counting | Enables counting |
| 3 | 1) Disables counting 2) Sets output immediately high | 1) Reloads counter 2) Initiates counting | Enables counting |
| 4 | Disables counting | --- | Enables counting |
| 5 | --- | Initiates counting | --- |

Figure 8. Gate Pin Operations Summary

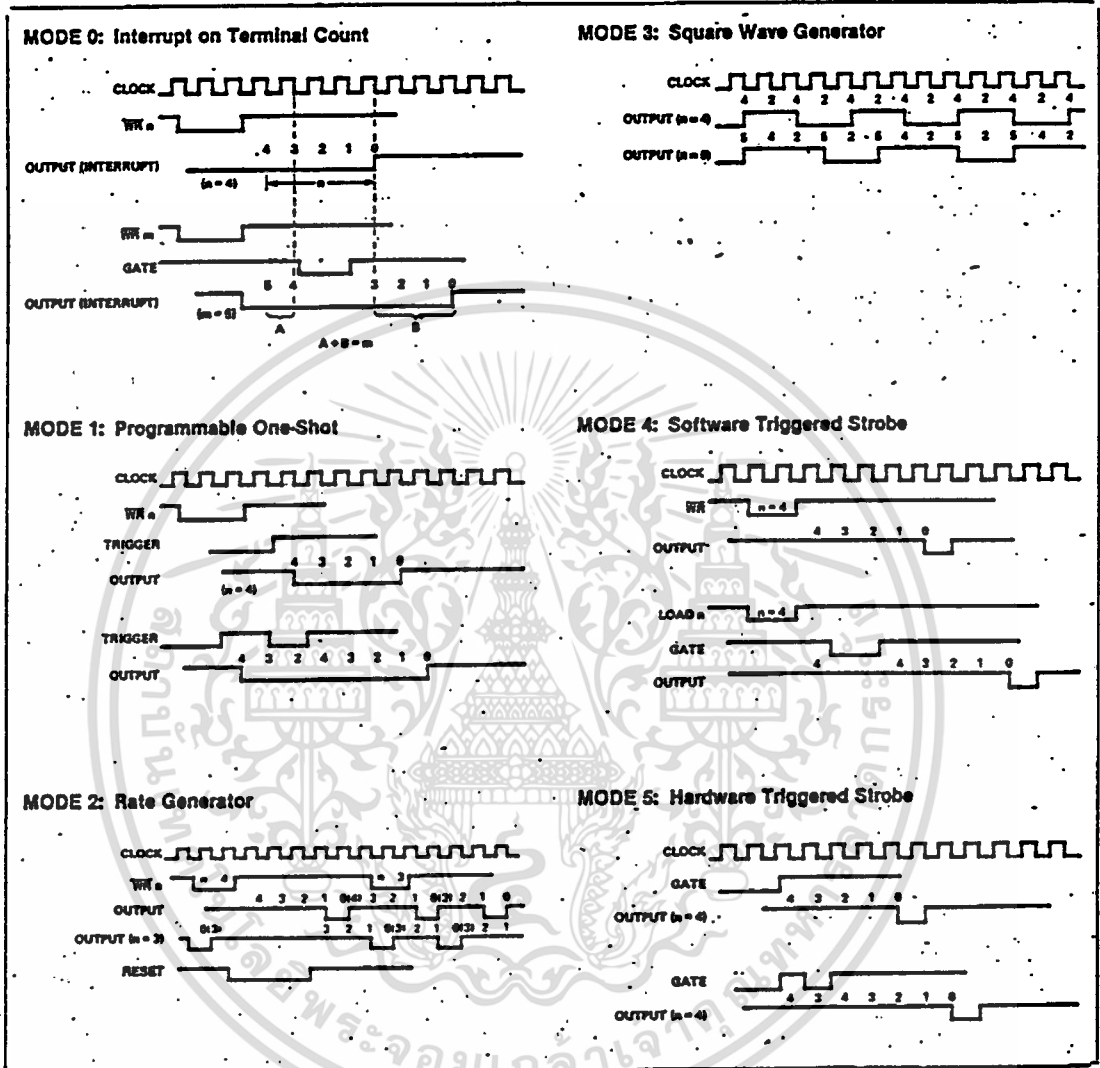


Figure 7. 8253 Timing Diagrams

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8253 READ/WRITE PROCEDURE

Write Operations

The systems software must program each counter of the 8253 with the mode and quantity desired. The programmer must write out to the 8253 a MODE control word and the programmed number of count register bytes (1 or 2) prior to actually using the selected counter.

The actual order of the programming is quite flexible. Writing out of the MODE control word can be in any sequence of counter selection, e.g., counter #0 does not have to be first or counter #2 last. Each counter's MODE control word register has a separate address so that its loading is completely sequence independent. (SC0, SC1)

The loading of the Count Register with the actual count value, however, must be done in exactly the sequence programmed in the MODE control word (RL0, RL1). This loading of the counter's count register is still sequence independent like the MODE control word loading, but when a selected count register is to be loaded it must be loaded with the number of bytes programmed in the MODE control word (RL0, RL1). The one or two bytes to be loaded in the count register do not have to follow the associated MODE control word. They can be programmed at any time following the MODE control word loading as long as the correct number of bytes is loaded in order.

All counters are down counters. Thus, the value loaded into the count register will actually be decremented. Loading all zeroes into a count register will result in the maximum count (2^{16} for Binary or 10^4 for BCD). In MODE 0 the new count will not restart until the load has been completed. It will accept one of two bytes depending on how the MODE control words (RL0, RL1) are programmed. Then proceed with the restart operation.

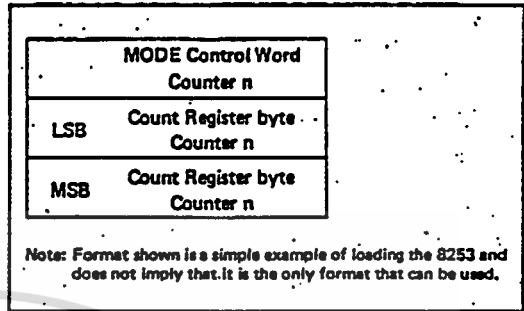


Figure 8. Programming Format

| | | A1 | A0 |
|-------|--------------------------------------|----|----|
| No. 1 | MODE Control Word Counter 0 | 1 | 1 |
| No. 2 | MODE Control Word Counter 1 | 1 | 1 |
| No. 3 | MODE Control Word Counter 2 | 1 | 1 |
| No. 4 | LSB Count Register Byte Counter 1 | 0 | 1 |
| No. 5 | MSB Count Register Byte Counter 1 | 0 | 1 |
| No. 6 | LSB Count Register Byte Counter 2 | 1 | 0 |
| No. 7 | MSB Count Register Byte Counter 2 | 1 | 0 |
| No. 8 | LSB Count Register Byte Counter 0 | 0 | 0 |
| No. 9 | MSB Count Register Byte Counter 0 | 0 | 0 |

Note: The exclusive addresses of each counter's count register make the task of programming the 8253 a very simple matter, and maximum effective use of the device will result if this feature is fully utilized.

Figure 9. Alternate Programming Formats

Read Operations

In most counter applications it becomes necessary to read the value of the count in progress and make a computational decision based on this quantity. Event counters are probably the most common application that uses this function. The 8253 contains logic that will allow the programmer to easily read the contents of any of the three counters without disturbing the actual count in progress.

There are two methods that the programmer can use to read the value of the counters. The first method involves the use of simple I/O read operations of the selected counter. By controlling the A0, A1 inputs to the 8253 the programmer can select the counter to be read (remember that no read operation of the mode register is allowed A0, A1=11). The only requirement with this method is that in order to assure a stable count reading the actual operation of the selected counter must be inhibited either by controlling the Gate input or by external logic that inhibits the clock input. The contents of the counter selected will be available as follows:

- first I/O Read contains the least-significant byte (LSB).
- second I/O Read contains the most significant byte (MSB).

Due to the internal logic of the 8253 it is absolutely necessary to complete the entire reading procedure. If two bytes are programmed to be read then two bytes must be read before any loading WR command can be sent to the same counter.

Read Operation Chart

| A1 | A0 | RD | |
|----|----|----|--------------------|
| 0 | 0 | 0 | Read Counter No. 0 |
| 0 | 1 | 0 | Read Counter No. 1 |
| 1 | 0 | 0 | Read Counter No. 2 |
| 1 | 1 | 0 | Illegal |

Reading While Counting

In order for the programmer to read the contents of any counter without effecting or disturbing the counting operation the 8253 has special internal logic that can be accessed using simple WR commands to the MODE register. Basically, when the programmer wishes to read the contents of a selected counter "on the fly" he loads the MODE register with a special code which latches the present count value into a storage register so that its contents contain an accurate, stable quantity. The programmer then issues a normal read command to the selected counter and the contents of the latched register is available.

MODE Register for Latching Count

A0, A1 = 11

| D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|-----|-----|----|----|----|----|----|----|
| SC1 | SC0 | 0 | 0 | X | X | X | X |

- SC1, SC0 — specify counter to be latched.
- D5, D4 — 00 designates counter latching operation.
- X — don't care.

The same limitation applies to this mode of reading the counter as the previous method. That is, it is mandatory to complete the entire read operation as programmed. This command has no effect on the counter's mode.

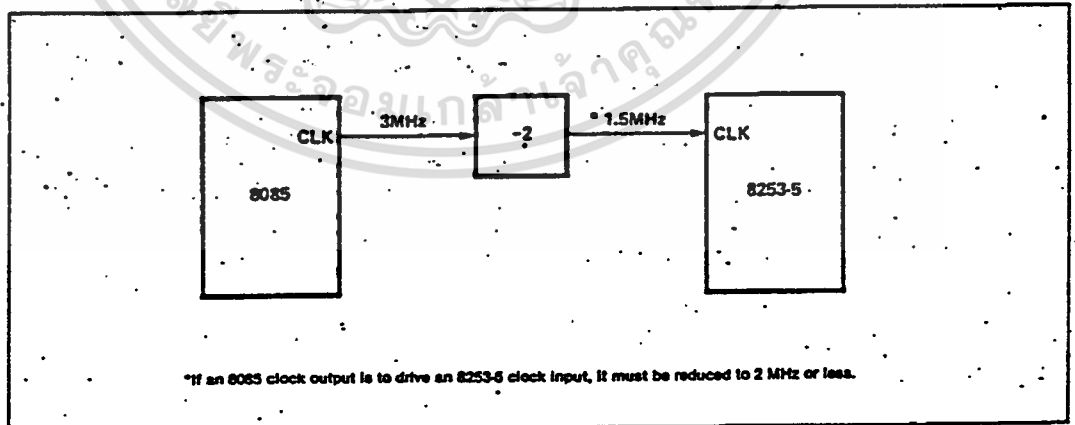


Figure 10. MCS-85™ Clock Interface*



8253/8253-5

ABSOLUTE MAXIMUM RATINGS*

| | | |
|--------------------------------|-------|-----------------|
| Ambient Temperature Under Bias | | 0°C to 70°C |
| Storage Temperature | | -65°C to +150°C |
| Voltage On Any Pin | | |
| With Respect to Ground | | -0.5V to +7V |
| Power Dissipation | | 1 Watt |

*NOTICE: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

D.C. CHARACTERISTICS ($T_A = 0^\circ\text{C}$ to 70°C , $V_{CC} = 5\text{V} \pm 10\%$)

| Symbol | Parameter | Min. | Max. | Unit | Test Conditions |
|-----------|-------------------------|------|------------------------|---------------|----------------------------|
| V_{IL} | Input Low Voltage | -0.5 | 0.8 | V | |
| V_{IH} | Input High Voltage | 2.2 | $V_{CC} + 0.5\text{V}$ | V | |
| V_{OL} | Output Low Voltage | | 0.45 | V | Note 1 |
| V_{OH} | Output High Voltage | 2.4 | | V | Note 2 |
| I_{IL} | Input Load Current | | ± 10 | μA | $V_{IN} = V_{CC}$ to 0V |
| I_{OFL} | Output Float Leakage | | ± 10 | μA | $V_{OUT} = V_{CC}$ to .45V |
| I_{CC} | V_{CC} Supply Current | | 140 | mA | |

CAPACITANCE ($T_A = 25^\circ\text{C}$, $V_{CC} = \text{GND} = 0\text{V}$)

| Symbol | Parameter | Min. | Typ. | Max. | Unit | Test Conditions |
|-----------|-------------------|------|------|------|------|--------------------------------------|
| C_{IN} | Input Capacitance | | | 10 | pF | $f_c = 1\text{ MHz}$ |
| $C_{I/O}$ | I/O Capacitance | | | 20 | pF | Unmeasured pins returned to V_{SS} |

A.C. CHARACTERISTICS ($T_A = 0^\circ\text{C}$ to 70°C , $V_{CC} = 5.0\text{V} \pm 10\%$, $\text{GND} = 0\text{V}$)

Bus Parameters (Note 3)

READ CYCLE

| Symbol | Parameter | 8253 | | 8253-5 | | Unit |
|----------|---|------|------|--------|------|---------------|
| | | Min. | Max. | Min. | Max. | |
| t_{AR} | Address Stable Before $\overline{\text{READ}}$ | 50 | | 30 | | ns |
| t_{RA} | Address Hold Time for $\overline{\text{READ}}$ | 5 | | 5 | | ns |
| t_{RR} | $\overline{\text{READ}}$ Pulse Width | 400 | | 300 | | ns |
| t_{RD} | Data Delay From $\overline{\text{READ}}^*$ | | 300 | | 200 | ns |
| t_{DF} | $\overline{\text{READ}}$ to Data Floating | 25 | 125 | 25 | 100 | ns |
| t_{rv} | Recovery Time Between $\overline{\text{READ}}$ and Any Other Control Signal | 1 | | 1 | | μs |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

A.C. CHARACTERISTICS (Continued)
WRITE CYCLE

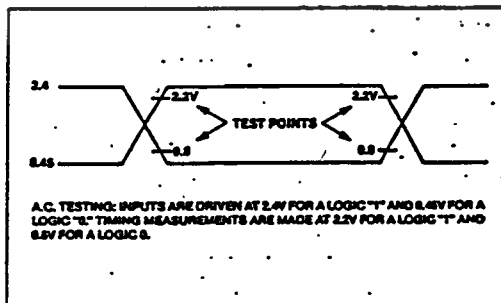
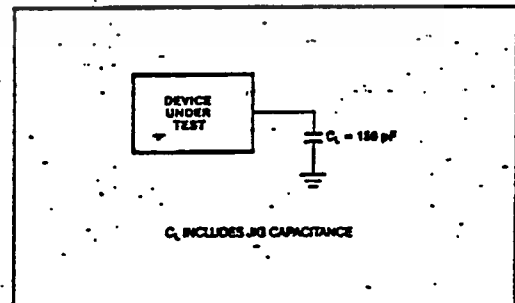
| Symbol | Parameter | 8253 | | 8253-5 | | Unit |
|----------|--|------|------|--------|------|---------|
| | | Min. | Max. | Min. | Max. | |
| t_{AW} | Address Stable Before WRITE | 50 | | 30 | | ns |
| t_{WA} | Address Hold Time for WRITE | 30 | | 30 | | ns |
| t_{WW} | WRITE Pulse Width | 400 | | 300 | | ns |
| t_{DW} | Data Set Up Time for WRITE | 300 | | -250 | | ns |
| t_{WD} | Data Hold Time for WRITE | 40 | | 30 | | ns |
| t_{RV} | Recovery Time Between WRITE and Any Other Control Signal | 1 | | 1 | | μ s |

CLOCK AND GATE TIMING

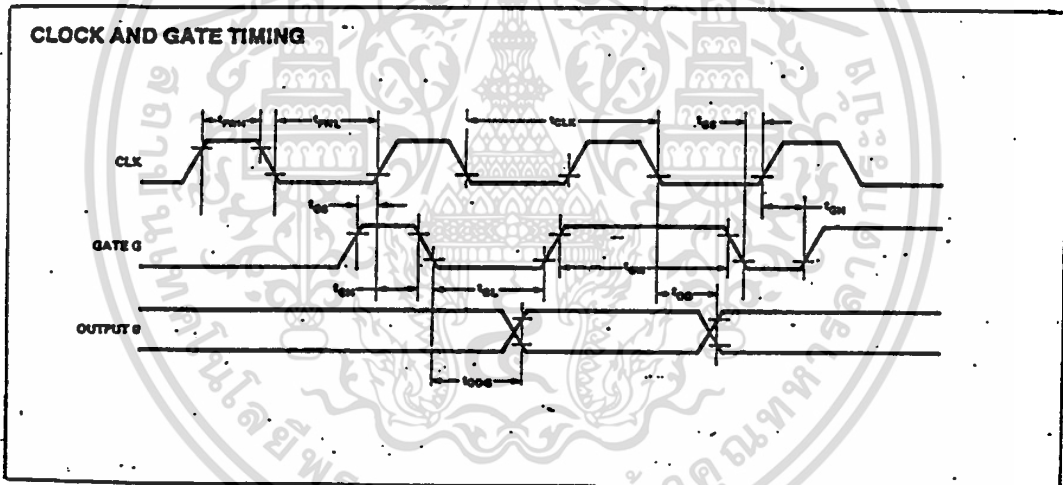
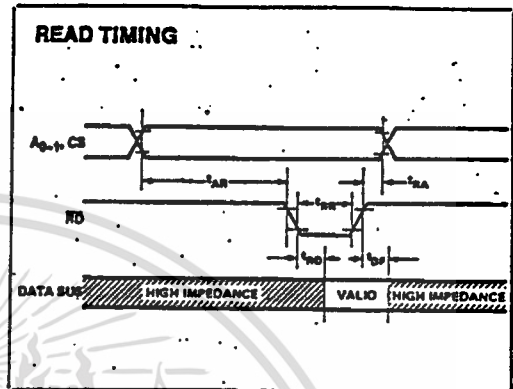
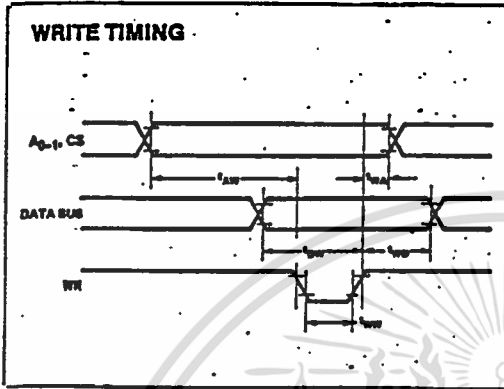
| Symbol | Parameter | 8253 | | 8253-5 | | Unit |
|-----------|---|------|------|--------|------|------|
| | | Min. | Max. | Min. | Max. | |
| t_{CLK} | Clock Period | 380 | dc | 380 | dc | ns |
| t_{PWH} | High Pulse Width | 230 | | 230 | | ns |
| t_{PWL} | Low Pulse Width | 150 | | 150 | | ns |
| t_{GW} | Gate Width High | 150 | | 150 | | ns |
| t_{GL} | Gate Width Low | 100 | | 100 | | ns |
| t_{GS} | Gate Set Up Time to CLK \uparrow | 100 | | 100 | | ns |
| t_{GH} | Gate Hold Time After CLK \uparrow | 50 | | 50 | | ns |
| t_{OD} | Output Delay From CLK \downarrow (4) | | 400 | | 400 | ns |
| t_{ODG} | Output Delay From Gate \downarrow (4) | | 300 | | 300 | ns |

NOTES:

1. $I_{OL} = 2.2$ mA.
 2. $I_{OH} = -400$ μ A.
 3. AC timings measured at $V_{OH} = 2.2$, $V_{OL} = 0.8$.
 4. $C_L = 150$ pF.
- * For Extended Temperature EXPRESS, use M8253 electrical parameters.

A.C. TESTING INPUT, OUTPUT WAVEFORM

A.C. TESTING LOAD CIRCUIT


WAVEFORMS



5474/7474 Dual D-Type Positive-Edge-Triggered Flip-Flop with Preset and Clear

| | Schottky TTL | | | | High-Speed TTL | | | | Low-Power Schottky TTL | | | | Standard TTL | | | | Low-Power TTL | | | |
|------------|----------------|---|---------|----|----------------|---|---------|----|------------------------|---|---------|----|---------------|---|---------|----|----------------|---|---------|----|
| | Device Type | | Package | | Device Type | | Package | | Device Type | | Package | | Device Type | | Package | | Device Type | | Package | |
| | C | P | M | CF | C | P | M | CF | C | P | M | CF | C | P | M | CF | C | P | M | CF |
| T.I. | SN54S74 | J | D | W | SN54H74 | J | D | W | SN54LS74 | J | D | W | SN5474 | J | D | W | SN54L74 | J | D | W |
| FAIRCHILD | SN74S74 | J | D | W | SN74H74 | J | D | W | SN74LS74 | J | D | W | SN7474 | J | D | W | SN74L74 | J | D | W |
| MOTOROLA | FM54S74/FM9S74 | D | E | K | FM54H74/FM9H74 | D | E | K | FM54LS74/FM9LS74 | D | E | K | FM5474/FM974 | D | E | K | FM54L74/FM9L74 | D | E | K |
| N.S.C. | DM74S74 | | | | DM54H74 | J | D | W | DM54LS74 | J | D | W | DM5474 | J | D | W | DM54L74 | J | D | W |
| PHILIPS | N74S74 | | | | 0JJ131/74H74 | | | | N74LS74 | | | | FJJ131/7474 | | | | | | | |
| SIGNETICS | S54S74 | | | | S54H74 | F | A | J | S54LS74 | F | A | J | S5474 | F | A | J | S54L74 | F | A | J |
| SIEMENS | N74S74 | | | | N74H74 | F | A | J | N74LS74 | F | A | J | N7474 | F | A | J | | | | |
| FUJITSU | | | | | | | | | 74LS74 | | | | MB420 | | | | | | | |
| HITACHI | HD74S74 | | | | | | | | HD74LS74 | | | | HD7474/HD2510 | | | | | | | |
| mitsubishi | M74S74 | | | | | | | | M74LS74 | | | | MS327/MS374 | | | | | | | |
| NEC | 74S74 | | | | | | | | 74LS74 | | | | μPB214 | | | | | | | |
| TOSHIBA | | | | | | | | | | | | | TC3474A | | | | | | | |

Electrical Characteristics SN54LS74/SN74LS74
absolute maximum ratings over operating free-air temperature range

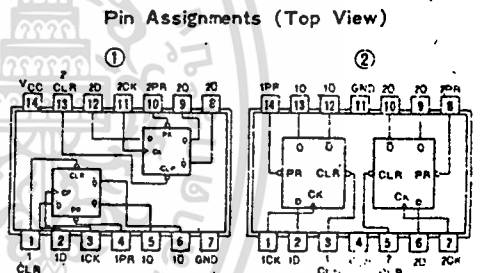
| | | | | |
|--------------------------------|------|-----------------------------------|--------|----------------|
| Supply voltage V _{CC} | 7V | Operating power temperature range | SN54LS | -55°C to 125°C |
| Input voltage | 5.2V | Storage temperature range | SN74LS | 0°C to 70°C |
| | | | | -65°C to 150°C |

recommended operating conditions

| | SN54LS74 | | | SN74LS74 | | | UNIT |
|---|---------------------|------|-----|----------|------|-----|------|
| | MIN | MAX | TYP | MIN | MAX | TYP | |
| Supply voltage, V _{CC} | 4.5 | 5.5 | 5 | 4.5 | 5.5 | 5 | V |
| High-level output current, I _{OH} | | -400 | | | -400 | | mA |
| Low-level output current, I _{OL} | | 4 | | | 4 | | mA |
| Power diss. to | Clear pin | 25 | | 25 | | | mW |
| | Preset or clear pin | 25 | | 25 | | | mW |
| Input setup time, t _{su} | High-level data | 25 | | 25 | | | ns |
| | Low-level data | 20 | | 20 | | | ns |
| Input hold time, t _{hd} | | 5 | | 5 | | | ns |
| Operating power temperature, T _A | | -55 | 125 | 0 | 70 | | °C |

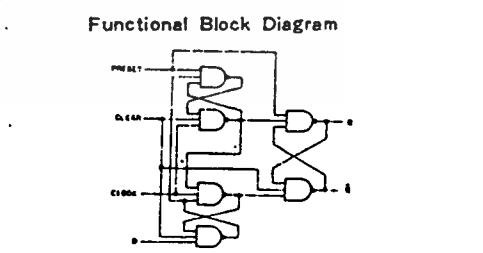
electrical characteristics over recommended operating free-air temperature range

| 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 = -18 mA | | -1.5 | V |
| V _{OH} | High-level output voltage | V _{CC} = MIN, V _{IH} = 2V, V _{IL} = 0.8 V, I _{OH} = MAX | 2.7 | 3.4 | V |
| V _{OL} | Low-level output voltage | V _{CC} = MIN, V _{IH} = 2V, V _{IL} = 0.8 V, I _{OL} = 4 mA | 0.25 | 0.4 | V |
| I _I | Input current at maximum input voltage | D, J, K, Clear | | 0.1 | mA |
| | | Preset | | 0.2 | |
| | | Clock | | 0.2 | |
| | | Clock | | 0.1 | |
| I _{IH} | High-level input current | D, J, K, Clear | | 20 | μA |
| | | Preset | | 40 | |
| | | Clock | | 40 | |
| | | Clock | | 20 | |
| I _{IL} | Low-level input current | D, J, K, Clear | | -0.4 | mA |
| | | Preset | | -0.8 | |
| | | Clock | | -0.8 | |
| | | Clock | | -0.4 | |
| I _{OS} | Short-circuit output current † | V _{CC} = MAX | -20 | -100 | mA |
| I _{CC} | Supply current (Average per flip-flop) | V _{CC} = MAX, See Note 1 | 4 | 8 | mA |
| f | clock frequency | V _{CC} = 5V, T _A = 25°C, C _L = 15 pF, R _L = 2kΩ | 25 | 33 | MHz |
| t _{PLH} | from clear, preset or clock (as appropriate) to 0 or 1 | | 13 | 25 | ns |
| t _{PHL} | | | 25 | 40 | ns |



Functional Table
74, 74L, LS74, S74 (See Note 2)

| INPUTS | | | OUTPUTS | | | |
|--------|-------|-------|---------|----------------|----------------|----------------|
| PRESET | CLEAR | CLOCK | D | Q | Q-bar | Q |
| L | H | X | X | H | L | L |
| H | L | X | X | L | H | H |
| L | L | X | X | H | H | H |
| H | H | ↑ | H | H | L | L |
| H | H | ↑ | L | L | H | H |
| H | H | L | X | Q ₀ | Q ₀ | Q ₀ |



74, S74, 74L, LS74, S74 DUAL D-TYPE FLIP-FLOP WITH CLEAR AND PRESET

NOTES: 1 With all outputs open, I_{CC} is measured with the Q and Q-bar outputs high in turn. At the time of measurement, the clock input is grounded.
2 H = high level (steady state), L = low level (steady state), X = irrelevant, ↑ = transition from low to high level.
3 Only the level of Q before the indicated input conditions were established.
4 This configuration is nonlatch, that is, it will not persist when preset and clear inputs return to their inactive (L or 1) level.

เอกสารนี้เป็นเอกสารของบริษัทเอกชนเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า
ไม่ให้นำไปใช้โดยไม่ได้รับอนุญาตจากทางบริษัท

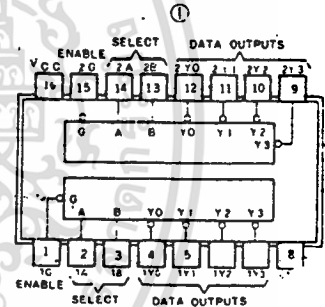
54139/74139 Dual 2-Line-to-4-Line Decoder/Demultiplexer

| | Schottky TTL | | | | High-Speed TTL | | | | Low-Power Schottky TTL | | | | Standard TTL | | | | Low-Power TTL | | | | |
|-----------|----------------------|---------|---|---|----------------|---------|---|---|------------------------|---------|------------------------|---|--------------|---------|----|--|---------------|---------|---|----|--|
| | Device Type | Package | | | Device Type | Package | | | Device Type | Package | | | Device Type | Package | | | Device Type | Package | | | |
| | | C | P | M | CF | | C | P | M | CF | | C | P | M | CF | | C | P | M | CF | |
| T.I. | SN54S139 | J | I | | W | | | | | | SN54LS139 | J | I | | W | | | | | | |
| | SN74S139 | J | I | N | | | | | | | SN74LS139 | J | I | N | J | | | | | | |
| FAIRCHILD | FM35139 | D | | | | | | | | | FM45139 / FM55139 | D | | | | | | | | | |
| | FC74S139 / FCN74S139 | D | | | | | | | | | FC74LS139 / FCN74LS139 | D | | | | | | | | | |
| MOTOROLA | | | | | | | | | | | SN74LS139 | | | | | | | | | | |
| N.S.C. | DM74S139 | | | | | | | | | | DM54LS139 | | | | | | | | | | |
| | | | | | | | | | | | DM74LS139 | | | | | | | | | | |
| PHILIPS | N74S139 | | | | | | | | | | N74LS139 | | | | | | | | | | |
| SIGNETICS | S54S139 | F | I | B | I | W | | | | | | | | | | | | | | | |
| | N74S139 | F | I | B | J | W | | | | | N74LS139 | | | | | | | | | | |
| SIEMENS | | | | | | | | | | | | | | | | | | | | | |
| FUJITSU | | | | | | | | | | | 74LS139 | | | | | | | | | | |
| HTACHI | | | | | | | | | | | HD74LS139 | | | | | | | | | | |
| MTSUBISHI | | | | | | | | | | | M74LS139 | | | | | | | | | | |
| NEC | | | | | | | | | | | 74LS139 | | | | | | | | | | |
| AMD | | | | | | | | | | | | | | | | | | | | | |

Electrical Characteristics SN54LS139/SN74LS139

| absolute maximum ratings over operating free-air temperature range | | | | | | | | | |
|--|--|--------------------------------------|-----------|----------------|------|------|-----|------|----|
| Supply voltage, V _{CC} | 7V | Operating free-air temperature range | SN54LS | -55°C to 125°C | | | | | |
| Input voltage | 7V | temperature range | SN74LS | 0°C to 70°C | | | | | |
| | | Storage temperature range | | -65°C to 150°C | | | | | |
| recommended operating conditions | | | | | | | | | |
| | | | SN54LS139 | SN74LS139 | UNIT | | | | |
| | | | MIN | NOM | MAX | MIN | NOM | MAX | |
| Supply voltage, V _{CC} | | | 4.5 | 5 | 5.5 | 4.75 | 5 | 5.25 | V |
| High-level output current, I _{OH} | | | | | | 400 | | 400 | mA |
| Low-level output current, I _{OL} | | | | | | 4 | | 8 | mA |
| 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 _{HI} High-level input voltage | | | 2 | | V | | | | |
| V _{LI} 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, I _{OH} = -1mA | SN54LS | 2.5 | 3.4 | V | | | | |
| | | SN74LS | 2.7 | 3.4 | | | | | |
| V _{OL} Low-level output voltage | V _{CC} = MIN., V _{IL} = 2V, I _{OL} = 8mA | | 0.35 | 0.5 | V | | | | |
| 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 = 7V | | 20 | | µA | | | | |
| I _{LI} Low-level input current | V _{CC} = MAX., V _I = 0.5V | | -0.4 | | mA | | | | |
| I _{CS} Short-circuit output current* | V _I = MAX. | | -20 | -100 | mA | | | | |
| I _{CC} Supply current | V _{CC} = MAX., Outputs enabled and open | | 6.8 | 11 | mA | | | | |
| t _{PLH} from Binary select | V _{CC} = 5V, T _A = 25°C, C _L = 15pF, R _L = ZK [¶] | 2 | 13 | 20 | ns | | | | |
| t _{PHL} to Any output | | | 22 | 33 | | | | | |
| t _{PLH} to Any output | | 3 | 18 | 29 | ns | | | | |
| t _{PHL} from Enable | | | 25 | 38 | | | | | |
| t _{PLH} to Any output | 2 | 16 | 24 | ns | | | | | |
| t _{PHL} to Any output | | 21 | 32 | | | | | | |

Pin Assignment (Top View)



positive logic: see function table

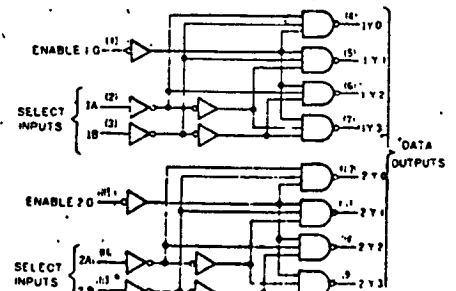
Function Table

'S139 'LS139 (EACH DECODER/DEMUTIPLEXER)

| INPUTS | | SELECT | | | | OUTPUTS | | | |
|--------|----|--------|---|---|----|---------|----|----|--|
| ENABLE | Z0 | Z1 | A | B | Y0 | Y1 | Y2 | Y3 | |
| H | X | X | H | H | H | H | H | H | |
| L | L | L | L | H | H | H | H | H | |
| L | L | L | H | H | L | L | L | L | |
| L | L | L | H | L | H | L | L | L | |
| L | L | L | L | L | H | L | L | L | |
| L | L | L | L | H | L | L | L | L | |

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

Functional Block Diagram



'S139 'LS139 DECODER/DEMUTIPLEXER

† 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 one output should be asserted at a time.

‡ t_{PLH} = propagation delay time, low-to-high-level output; t_{PHL} = propagation delay time, high-to-low-level output.

เอกสารนี้เป็นทรัพย์สินของสถาบันเพื่อการศึกษาเท่านั้น มิอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5473/7473 Dual J-K Master-Slave Flip-Flop with Clear

| | Schottky TTL | | | | High-Speed TTL | | | | Low-Power Schottky TTL | | | | Standard TTL | | | | Low-Power TTL | | | | |
|------------|--------------|---|---------|----|----------------|---|---------|----|------------------------|------------------|---------|----|--------------|----|---------------|---------------|---------------|----|---------|----|----|
| | Device Type | | Package | | Device Type | | Package | | Device Type | | Package | | Device Type | | Package | | Device Type | | Package | | |
| | C | P | M | CF | C | P | M | CF | C | P | M | CF | C | P | M | CF | C | P | M | CF | |
| T.I. | | | | | SN54H73 | J | Q | | W3 | SN54LS73 | J | Q | | ND | | W3 | SN5473 | J | Q | | ND |
| | | | | | SN74H73 | J | Q | ND | | SN74LS73 | J | Q | ND | | | | SN7473 | J | Q | ND | TD |
| FAIRCHILD | | | | | FMS4H73/FM9H73 | Q | Q | | FQ | FM54LS73/FM5LS73 | | | | | | FMS473/FM9N73 | Q | Q | | FQ | |
| | | | | | FC74H73/FC9H73 | Q | Q | PD | | FC74LS73/FC9LS73 | | | | | | FC7473/FC9N73 | Q | Q | PD | | |
| MOTOROLA | | | | | MC3163 | L | Q | | FQ | | | | | | MC5473 | L | Q | | FQ | | |
| | | | | | MC3063 | L | Q | PD | FQ | SN74LS73 | | | | | MC7473 | L | Q | PD | FQ | | |
| N.S.C. | | | | | DM54H73 | J | Q | ND | | DM54LS73 | | | | | DM5473 | J | Q | ND | | WJ | |
| | | | | | DM74H73 | J | Q | ND | | DM74LS73 | | | | | DM7473 | J | Q | ND | | WJ | |
| PHILIPS | | | | | N74H73 | | Q | | | N74LS73 | | Q | | | FJJ121/7473 | | Q | | | | |
| SIGNETICS | | | | | SS4H73 | F | Q | AD | WJ | DM54LS73 | | | | | SS473 | F | Q | AC | WJ | | |
| | | | | | N74H73 | F | Q | AD | | N74LS73 | | AD | | | N7473 | F | Q | AD | | | |
| SIEMENS | | | | | | | | | | | | | | | FLJ121 | | Q | | | | |
| FUJITSU | | | | | | | | | | | | | | | | | | | | | |
| HITACHI | | | | | | | | | | HD74LS73 | | P | Q | | HD7473/HD2515 | | Q | P | Q | | |
| MITSUBISHI | | | | | | | | | | M74LS73 | | P | Q | | M53273 | | P | Q | | | |
| NEC | | | | | | | | | | | | | | | μPB225 | | D | Q | C | | |
| TOSHIBA | | | | | | | | | | | | | | | TD3473A | | P | Q | | | |

Electrical Characteristics SN54LS73/SN74LS73A

absolute maximum ratings over operating free-air temperature range

| | | | | |
|---------------------------------|------|--------------------------------------|--------|----------------|
| Supply voltage, V _{CC} | 7V | Operating free-air temperature range | SN54LS | -55°C to 125°C |
| Input voltage | 5.5V | | SN74LS | 0°C to 70°C |
| | | Storage temperature range | | -65°C to 150°C |

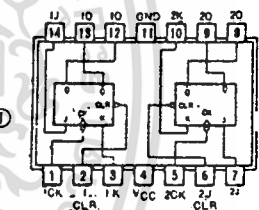
recommended operating conditions

| | SN54LS73A | | | SN74LS73A | | | UNIT |
|--|-----------|---------------------|------|-----------|---------|------|------|
| | MIN | NOM | MAX | MIN | NOM | MAX | |
| 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 | | | 4 | mA |
| Pulse width, t _p | | Clk high | 20 | | Clk low | 20 | ns |
| | | Preset or clear low | 25 | | 25 | | |
| Input setup time, t _{su} | | High-level data | 20 | | 25 | | ns |
| | | Low-level data | 20 | | 20 | | |
| Input hold time, t _{hd} | | | 0 | | 5 | | ns |
| Operating free-air temperature, T _a | | | -55 | | 125 | | °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} =MAX | 2.7 | 3.4 | V |
| V _{OL} | Low-level output voltage | V _{CC} =MIN, V _{IH} =2V, V _{IL} =0.8V, I _{OL} =4mA | 0.25 | 0.4 | V |
| I _I | Input current at maximum input voltage | D, J, K, Clear, Preset, Clock | | 0.1, 0.3, 0.3, 0.4 | mA |
| I _{IH} | High-level input current | D, J, K, Clear, Preset, Clock | | 20, 60, 60, 80 | μA |
| I _{IL} | Low-level input current | D, J, K, Clear, Preset, Clock | | -0.4, -0.8, -0.8, -0.8 | mA |
| I _{OS} | Short-circuit output current‡ | Series 54LS, Series 74LS, V _{CC} =MAX | | -20, -100, -20, -100 | mA |
| I _{CC} | Supply current (Average per flip-flop) | V _{CC} =MAX, See Note 1 | | 4, 6 | mA |
| f _{max} | maximum clock frequency | V _{CC} =5V, T _a =25°C, C _L =15pF, R _L =2kΩ | | 30, 45 | MHz |
| t _{PLH} | propagation delay, time, low-to-high-level output | | | 15, 20 | ns |
| t _{PHL} | propagation delay, time, high-to-low-level output | | | 15, 20 | ns |

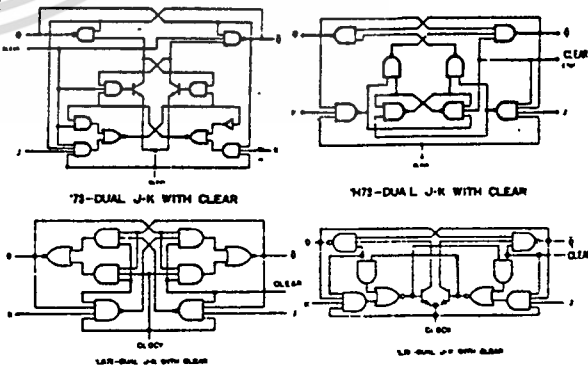
Pin Assignment (Top View)



Function Tables

| 73 H73, L73 (See Note 2) | | | | | | LS73 (See Note 2) | | | | | | |
|--------------------------|-------|---|---------|----------------|-----------------|-------------------|-------|---|---------|----------------|-----------------|-----------------|
| INPUTS | | | OUTPUTS | | | INPUTS | | | OUTPUTS | | | |
| CLEAR | CLOCK | J | K | Q | Q̄ | CLEAR | CLOCK | J | K | Q | Q̄ | |
| L | X | X | X | L | H | L | X | X | X | L | H | |
| H | ↓ | L | L | Q ₀ | Q̄ ₀ | H | ↓ | L | L | Q ₀ | Q̄ ₀ | |
| H | ↓ | L | H | L | L | H | ↓ | L | H | L | L | |
| H | ↓ | L | L | H | L | H | ↓ | L | H | L | H | |
| H | ↓ | H | H | TOGGLE | | H | ↓ | H | H | TOGGLE | | |
| H | ↓ | H | X | X | Q ₀ | Q̄ ₀ | H | ↓ | X | X | Q ₀ | Q̄ ₀ |

Function: Block Diagrams



NOTES: 1. with all outputs open, I_{CC} is measured with the Q and Q̄ outputs high in turn. At the time of measurement, the clock input is grounded.
 2. H=high level (steady state), L=low level (steady state), X=irrelevant, ↓=transition from high to low level.
 † high-level pulse, data inputs should be held constant while clock is high, data is transferred to output on the falling edge of the pulse.
 ‡ Each output changes in the complement of its previous level on each active transition (pulse) of the clock.

* 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 time.

t_{PLH} = propagation delay, time, low-to-high-level output.

t_{PHL} = propagation delay, time, high-to-low-level output.

§ ↓: The arrow indicates the edge of the clock pulse used for reference. † for the rising edge, ↓ for the falling edge.

กิตติกรรมประกาศ

ขอขอบคุณอาจารย์ยุทธนา คิดใจเดียว อาจารย์สุรพันธ์ เอื้อไพบูลย์ อาจารย์ผล ศาสตร์ และอาจารย์อื่นๆ เป็นอย่างสูงที่ช่วยให้คำปรึกษา และขอขอบคุณพี่มณฑา คุณศุภชัย น้อยสมทวน และภาคเครื่องกล เป็นอย่างมากสำหรับทาง Mechenic จนโครงการนี้สำเร็จลุล่วงไปได้ด้วยดี

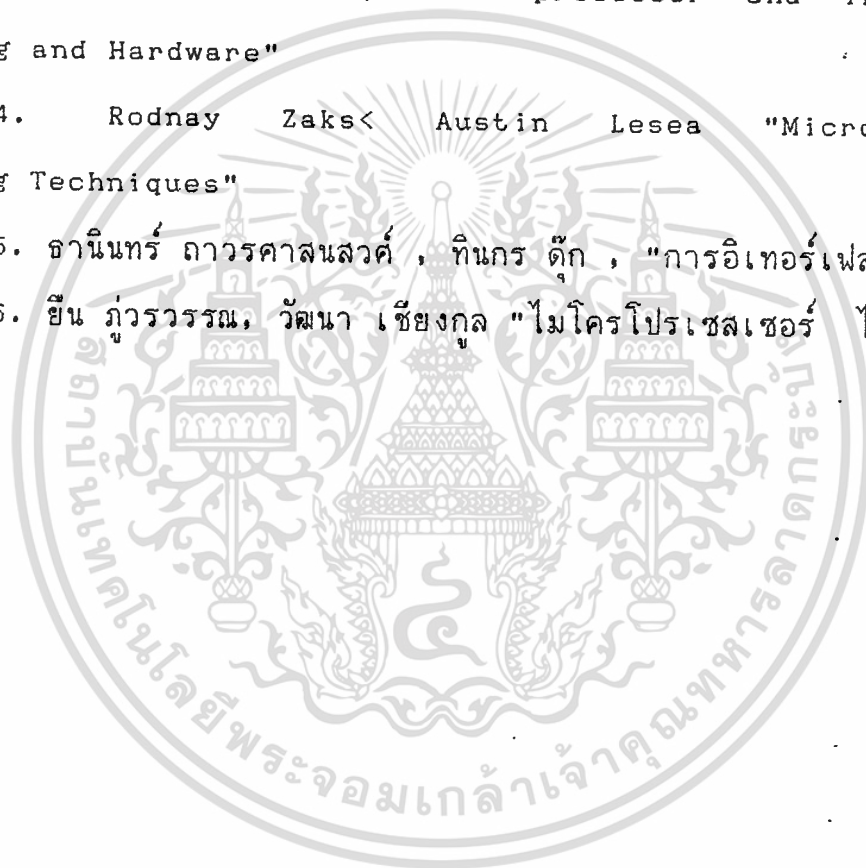


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. Takashi Kenjo, "Stepping motors and their microprocessor controls", Professor in the Department of Electrical Engineering, Institute of Vocational, Kanagawa, Japan
2. Borland International, "Turbo C User's Guide"
3. Douglas V. Hall, "Microprocessor and Interfacing Programming and Hardware"
4. Rodney Zaks & Austin Lesea "Microprocessor Interfacing Techniques"
5. ฉานินทร์ ถาวรศาสนสวัสดิ์ , ทิพนกร ตึก , "การอิเทอร์เฟส IBMPC"
6. ยืน ภู่วรวรรณ, วัฒนา เชียงกุล "ไมโครโปรเซสเซอร์ ไมโครคอมพิวเตอร์"

เตอร์"



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้