สำนักหอสมุดกลาง_พระจอมเกล้าลาดกระบัง

การติดตามอัตโนมัติ

AUTOMATIC TRACKING

นายจักรกฤษณ์ อุปยโส
นายสมบัติ เปกวง
นายสุมิทธิ์ ยิ้มงาม

เลขหมู่.....
เลขทะเบียน 45848
วัน, เดือน, ปี 19 ก.พ. 2546

.b.....
.i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมการวัดคุม
ภาควิชาวิศวกรรมการวัดคุม คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2544

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AUTOMATIC TRACKING

CHAKKRIS UPAYASO

SOMBUTT PEGUANG

SUMITH YIMNGAM

A THESIS SUBMITTED PARTAIL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

BACHELOR OF ENGINEERING INSTRUMENTATION ENGINEERING

DEPARTMENT OF INSTRUMENTATION ENGINEERING

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2001

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาควิชาวิศวกรรมการวัดคุม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ใบรับรองปริญญานิพนธ์

หัวข้อปริญญานิพนธ์ การติดตามอัตโนมัติ

AUTOMATIC TRACKING

นักศึกษาผู้จัดทำ

นายจักรกฤษณ์	อุปยโส	รหัสประจำตัว	42015427
นายสมบัติ	เปกวง	รหัสประจำตัว	42015455
นายสุมิทธิ์	ยิ้มงาม	รหัสประจำตัว	42015457

ปริญญา วิศวกรรมศาสตรบัณฑิต

สาขาวิชา วิศวกรรมการวัดคุม

ปีการศึกษา 2544

อาจารย์ผู้ควบคุมปริญญานิพนธ์	ลายมือชื่อ
รศ.ดร.ฟูศักดิ์ ชีวสุวิทย์	

วัน/เดือน/ปี ที่สอบ วันอังคารที่ 26 มีนาคม 2545

สถานที่สอบ ณ ห้องสอบปริญญานิพนธ์ ภาควิชาวิศวกรรมการวัดคุม

ภาควิชารับรองแล้ว

(ผศ.ประสิทธิ์ จุลเสรีวงศ์)

หัวหน้าภาควิชาวิศวกรรมการวัดคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	การติดตามอัตโนมัติ	
	AUTOMATIC TRACKING	
นักศึกษาผู้จัดทำ	นายจักรกฤษณ์	อุปยโส
	นายสมบัติ	เปกวง
	นายสุมิทธิ์	ยิ้มงาม
อาจารย์ที่ปรึกษา	รศ.ดร.ฟูศักดิ์	ชีวสุวิทย์
ปีการศึกษา	2544	

บทคัดย่อ

ในปัจจุบันได้ มีการนำวิธีการประมวลผลภาพมาใช้ในงานด้านต่างๆ เช่น การสื่อสารข้อมูลภาพ การสำรวจทรัพยากรทางดาวเทียม การควบคุมการผลิตในอุตสาหกรรม สื่อการสอน เป็นต้น ในปริญญานิพนธ์นี้ได้นำวิธีการดังกล่าวมาประยุกต์ใช้กับการติดตามอัตโนมัติซึ่งจะเป็นการติดตามบุคคลหรือวัตถุที่กำลังเคลื่อนที่ โดยนำกล้องซีซีดี และสเต็ปปิ้งมอเตอร์มาทำงานร่วมกัน และจะควบคุมการทำงานโดยใช้โปรแกรมบอร์แลนด์ซีพลัสพลัส จากการทำงานร่วมกันของส่วนต่างๆ แล้วจะเห็นได้ว่าสามารถนำไปใช้ในงานรักษาความปลอดภัยตามสถานที่ต่างๆและตรวจจับบุคคลหรือวัตถุที่กำลังเคลื่อนที่ได้ และมีราคาไม่แพงมากนักอุปกรณ์สามารถหาได้ง่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	**Automatic Tracking**
Authors	**Mr. Chakkris Upayaso**
	Mr. Sombutt Peguang
	Mr. Sumith Yimngam
Thesis Advisor	**Assoc. Prof. Dr Fusak Cheevasuvit**
Year	**2001**

ABSTRACT

Image processing is now widely used in many applications such as image data communication, remote sensing, multimedia and etc. This thesis is developed from image processing to apply in automatic tracking. It can be tracked the moving person or object by using camera CCD and stepping motor which joined and controlled by Borland C++. Therefor, this project can be used for home security.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ขอขอบพระคุณ รศ.ดร.ฟูศักดิ์ ชีวสุวิทย์ ที่ได้ประสิทธิ์ประสาทวิชา พี่ภาวิน โกมลวิภาต ที่ได้ให้คำแนะนำปรึกษาในด้านการศึกษาแก่ผู้เขียนขอขอบพระคุณท่านคณะกรรมการสอบทุกท่าน ที่ได้สละเวลาอันมีค่า ขอขอบพระคุณ คุณพ่อคุณแม่และขอบคุณ เพื่อน ๆ ที่ได้ช่วยเหลือและให้กำลังใจในการทำ ปริญญานิพนธ์

คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....I

บทคัดย่อภาษาอังกฤษ.....II

กิตติกรรมประกาศ.....III

สารบัญ.....IV

สารบัญตาราง.....VII

สารบัญภาพ.....VIII

บทที่ 1 บทนำ.....1

1.1 ความเป็นมาและเหตุจูงใจของการวิจัย.....1

1.2 วัตถุประสงค์ของปริญญานิพนธ์.....1

1.3 ขอบเขตปริญญานิพนธ์.....1

1.4 ขั้นตอนการศึกษา.....2

บทที่ 2 การเพิ่มรายละเอียดให้กับข้อมูลภาพ.....3

2.1 กล่าวนำ.....3

2.2 การคำนวณความถี่ที่ใช้ในการสุ่มสัญญาณภาพ.....3

2.3 แนวทางการออกแบบ.....5

2.3.1 ส่วนของอินพุตบัฟเฟอร์.....5

2.3.2 ส่วนของการแยกซิงค์.....5

2.3.3 ส่วนของการกำเนิดสัญญาณนาฬิกา.....7

2.3.4 ส่วนของการแปลงสัญญาณแอนะล็อกเป็นดิจิตอล.....8

2.3.5 ส่วนของการมัลติเพล็กซ์.....8

2.3.6 ส่วนการกำเนิดสัญญาณแอดเดรส.....9

2.3.7 ส่วนของหน่วยความจำ.....10

2.3.8 ส่วนของการควบคุม.....12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

บทที่ 3 สเตปมอเตอร์.....14

3.1 กล่าวนำ.....14

3.2 ทฤษฎีพื้นฐานและหลักการการทำงานของสเตปมอเตอร์.....14

3.2.1 หลักการทำงานของสเตปมอเตอร์.....15

3.2.2 ชนิดของสเตปมอเตอร์.....16

3.2.2.1 สเตปมอเตอร์ชนิดปรับค่ารีลัคแตนซ์ได้ (VARIABLE RELUCTANCE STEPPING MOTOR).....16

3.2.2.2 สเตปมอเตอร์แบบแม่เหล็กถาวร.....20

3.2.2.3 สเตปมอเตอร์แบบไฮบริดจ์.....22

3.2.2.4 คุณสมบัติที่สำคัญของไฮบริดจ์สเตปมอเตอร์.....22

3.3 วงจรขับสเตปปิ้งมอเตอร์.....23

3.3.1 ระบบขับ (DRIVE SYSTEM).....23

บทที่ 4 สัญญาณต่าง ๆ บนสล็อตของ IBM/PC.....25

4.1 กล่าวนำ.....25

4.2 รายละเอียดเกี่ยวกับสัญญาณต่างๆ.....25

4.3 การจัดสัญญาณบนสล็อตของ IBM PC / XT.....33

4.4 บัสไซเคิลของระบบ.....34

4.5 บัสไซเคิลต่าง ๆ ในระบบ.....35

4.5.1 บัสไซเคิลในการอ่านข้อมูลจากหน่วยความจำ.....35

4.5.2 บัสไซเคิลในการเขียนข้อมูลลงบนหน่วยความจำ.....37

4.5.3 บัสไซเคิลในการอ่านข้อมูลจากพอร์ท.....38

4.5.4 บัสไซเคิลในการเขียนข้อมูลลงบนพอร์ท.....39

บทที่ 5 ผลการทดลอง.....41

5.1 กล่าวนำ.....41

5.2 ผลการทดลองการตรวจจับการเคลื่อนที่ของวัตถุ.....41

5.3 สรุปผลการทดลอง.....47

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

บทที่ 6 สรุปผลการวิจัยและข้อเสนอแนะ.....48

6.1 บทสรุป.....48

6.2 ข้อเสนอแนะ.....48

บรรณานุกรม.....49

ภาคผนวก.....50

ภาคผนวก ก.....51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ **หน้า**

2.1 แสดงพารามิเตอร์ของระบบโทรทัศน์สามระบบ คือ NTSC, PAL และ SECAM.....3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

ภาพที่ หน้า

2.1 แสดงรายละเอียดสัญญาณซิงค์ทางแนวนอน....4
2.2 แสดงบล็อกไดอะแกรมของแผงวงจรเก็บข้อมูลภาพ....6
2.3 แสดงส่วนของการแยกสัญญาณซิงค์....7
2.4 แสดงผังเวลาของการหน่วงทางแนวนอน....7
2.5 แสดงบล็อกไดอะแกรมส่วนการแปลงสัญญาณแอนะล็อกเป็นดิจิตอล....8
2.6 แสดงการอ้างตำแหน่งหน่วยความจำบนแผงวงจรเก็บข้อมูลภาพ....11
2.7 แสดงการอ้างตำแหน่งของเครื่องคอมพิวเตอร์....11
2.8 แสดงบล็อกไดอะแกรมของหน่วยความจำ....12
2.9 แสดงภาพที่ได้จากแผงวงจรเก็บข้อมูลภาพภาพที่....13
2.10 แสดงภาพที่ได้จากแผงวงจรเก็บข้อมูลภาพ....13
3.1 แสดงภาพตัดขวางของเตปมอเตอร์แบบ 3 เฟส....17
3.2 แสดงตำแหน่งสมดุลย์ เมื่อเฟสใดเฟสหนึ่งของสเตปมอเตอร์ถูกกระตุ้น....17
3.3 แสดงแรงภายนอกที่มีผลต่อเส้นแรงแม่เหล็ก....17
3.4 แสดงขั้นตอนการคลื่อนที่ของโรเตอร์เมื่อสเตปมอเตอร์ถูกกระตุ้น....18
3.5 แสดงขั้นตอนการเคลื่อนที่ของสเตปมอเตอร์....19
3.6 แสดงการเปรียบเทียบเส้นแรงแม่เหล็กระหว่างช่องว่างที่กว้างและแคบ....19
3.7 (a) แสดงมอเตอร์ 3 เฟส (b) แสดงมอเตอร์ 4 เฟส....20
3.8 แสดงโครงสร้างของสเตปมอเตอร์แบบแม่เหล็กถาวร....21
3.9 แสดงการทำงานของสเตปมอเตอร์แม่เหล็กถาวรขนาด 4 เฟส....21
3.10 แสดงโครงสร้างของไฮบริดจ์สเตปมอเตอร์....22
3.11 แสดงบล็อกไดอะแกรมของระบบขับสเตปมอเตอร์....24
4.1 แสดงสัญญาณต่างๆ บนของสล็อต IBM / PC....26
4.2 แสดงบัสไซเคิลของการอ่านข้อมูลจากหน่วยความจำ....36
4.3 แสดงบัสไซเคิลของการเขียนข้อมูลลงบนหน่วยความจำ....37
4.4 แสดงบัสไซเคิลของการอ่านข้อมูลจากพอร์ท I/O....38
4.5 แสดงบัสไซเคิลของการเขียนข้อมูลลงบนพอร์ท I/O....40
5.1 แสดงบล็อกไดอะแกรมของการตรวจจับการเคลื่อนที่ของวัตถุ....41
5.2 แสดงชุดตรวจจับการเคลื่อนที่ของวัตถุ....42

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ(ต่อ)

ภาพที่ หน้า

5.3 (ก) ภาพอดีต $f_{n-1}(x,y)$ (ข) ภาพปัจจุบัน $f_n(x,y)$ (ค) ภาพผลลัพธ์ของภาพ ปัจจุบันลบด้วยภาพอดีตขณะที่วัตถุยังไม่มีการเคลื่อนที่.....43

5.4 (ก) ภาพอดีต $f_{n-1}(x,y)$ (ข) ภาพปัจจุบัน $f_n(x,y)$ (ค) ภาพผลลัพธ์ของภาพ ปัจจุบันลบด้วยภาพอดีตในขณะที่วัตถุเคลื่อนที่ขึ้นไปทางซ้าย.....44

5.5 แสดงทิศทางการเคลื่อนที่ของกล้องตามแนวแกน x และแกน y เมื่อวัตถุเคลื่อนที่ขึ้นไปทางด้านบนซ้าย.....45

5.6 (ก) ภาพอดีต $f_{n-1}(x,y)$ (ข) ภาพปัจจุบัน $f_n(x,y)$ (ค) ภาพผลลัพธ์ของภาพ ปัจจุบันลบด้วยภาพอดีตในขณะที่วัตถุเคลื่อนที่ลงมาทางด้านขวา.....46

5.7 แสดงทิศทางการเคลื่อนที่ของกล้องตามแนวแกน x และแกน y เมื่อวัตถุเคลื่อนที่ลงมาทางด้านล่างขวา.....47

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและเหตุจูงใจของการวิจัย

จากการที่ศาสตร์ทางด้านการประมวลผลภาพได้เข้ามามีบทบาทในงานด้านต่างๆ ไม่ว่าเป็นด้านการพาณิชย์ การอุตสาหกรรม การสำรวจทรัพยากรธรรมชาติ การสื่อสาร ความมั่นคงของประเทศ และรวมไปถึงการช่วยพัฒนาทางด้านการศึกษาของประเทศด้วย ซึ่งในปัจจุบันสถาบันทางการศึกษาได้มีการศึกษาและวิจัยศาสตร์ทางด้านนี้มากขึ้นจึงได้นำศาสตร์ทางด้านการประมวลผลภาพมาประยุกต์ใช้ในการตรวจจับการเคลื่อนที่ของวัตถุ เพราะสามารถตรวจจับการเคลื่อนที่ของวัตถุนั้นมีประสิทธิภาพสูง

1.2 วัตถุประสงค์ของปริญญานิพนธ์

1.2.1 เพื่อศึกษาแนวทางในการออกแบบอุปกรณ์เก็บข้อมูลภาพ คือ ออกแบบแผงวงจรเก็บข้อมูลภาพนั่นเอง เป็นการเพิ่มทักษะในการพัฒนาระบบของการประมวลผลภาพด้วย

1.2.2 เพื่อช่วยลดงบประมาณในการจัดซื้อแผงวงจรควบคุมภาพรวมทั้งได้มีการเขียนซอฟต์แวร์ขึ้นมาควบคุมการทำงานแผงวงจรเก็บข้อมูลภาพดังกล่าวด้วย

1.2.3 สามารถนำแผงวงจรเก็บข้อมูลภาพที่สร้างขึ้นไปใช้งานได้จริง

1.2.4 เพื่อศึกษาการทำงานของสเตปมอเตอร์ และสร้างวงจรชุดขับสเตปมอเตอร์ขึ้นมา

1.2.5 เพื่อศึกษาและประยุกต์โปรแกรมในการตรวจจับการเคลื่อนที่ของวัตถุ

1.3 ขอบเขตของปริญญานิพนธ์

1.3.1 สร้างแผงวงจรเก็บข้อมูลภาพขนาด 512 x 512 จุดต่อภาพ ขนาด 8 บิตต่อภาพ โดยใช้ตัวแปลงสัญญาณแอนะล็อกเป็นดิจิตอลที่ความถี่ 12 MHz ใช้หน่วยความจำขนาด 256 กิโลไบต์ต่อภาพ อินเตอร์เฟสผ่านทาง XT BUS ใช้หน่วยความจำในการอินเตอร์เฟสกับเครื่องคอมพิวเตอร์ที่ D000:0000-D000:FFFF

1.3.2 สร้างแผงวงจรชุดขับสเต็ปปิ้งมอเตอร์จำนวน 2 ชุด เพื่อใช้ในการควบคุมทิศทางการหมุนของกล้อง CCD ในแนวนอน และแนวตั้ง

1.3.3 เขียนโปรแกรมควบคุมการทำงานของแผงวงจรเก็บข้อมูลภาพเพื่อนำข้อมูลภาพมาประมวลผล

1.3.4 เขียนโปรแกรมควบคุมทิศทางการหมุนของกล้องCCD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3.5 เขียนโปรแกรมควบคุมการตรวจจับการเคลื่อนที่ของวัตถุ

1.4 ขั้นตอนการศึกษา

ขั้นแรกทำการศึกษาการทำงานของแผงวงจรเก็บข้อมูลภาพและทำการสร้างแผงวงจรเก็บข้อมูลภาพขึ้นมาใช้งานพร้อมกับเขียนโปรแกรมควบคุมการเก็บข้อมูลภาพและทำการแสดงภาพออกที่หน้าจอ จากนั้นสร้างแผงวงจรขับสเต็ปปิ้งมอเตอร์พร้อมกับเขียนโปรแกรมควบคุมทิศทางการหมุนของสเต็ปปิ้งมอเตอร์ ต่อจากนั้นสร้างฐานกล้องโดยกำหนดให้มีทิศทางการหมุนในแนวแกนนอนและแนวแกนตั้งและนำข้อมูลภาพมาประมวลผลเพื่อควบคุมทิศทางการหมุนของกล้องให้เป็นไปตามทิศทางการของวัตถุที่เคลื่อนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
การเพิ่มรายละเอียดให้กับข้อมูลภาพ

2.1 กล่าวนำ

การแปลงสัญญาณภาพที่ได้จากกล้องวีดีโอหรือเครื่องเล่นวีดีโอต่างๆ ให้เป็นข้อมูลภาพที่เป็นดิจิตอลสำหรับนำไปประมวลผลต่อด้วยคอมพิวเตอร์นั้น มีกระบวนการแปลงที่เรียกว่าการสุ่มและการเข้ารหัส กระบวนการเหล่านี้จะร่วมเป็นอุปกรณ์ที่เรียกว่าตัวแปลงสัญญาณแอนะล็อกเป็นดิจิตอล ซึ่งข้อมูลภาพที่ได้นั้นจะบอกรายละเอียดของภาพเป็นจำนวนจุดต่อภาพและระดับของจุดภาพที่มีได้ ดังนั้นถาทำการเพิ่มจำนวนจุดภาพและระดับค่าของจุดภาพจะทำให้สามารถเก็บจำนวนข้อมูลจากภาพได้มากขึ้น และแต่ละจุดภาพสามารถให้รายละเอียดสูงขึ้น ในการเพิ่มจำนวนจุดภาพนั้นในที่นี้คือการเพิ่มอัตราการสุ่มต่อหนึ่งหน่วยเวลา และการเพิ่มระดับของจุดภาพจะเป็นการเพิ่มจำนวนบิตของตัวแปลงสัญญาณแอนะล็อกเป็นดิจิตอล

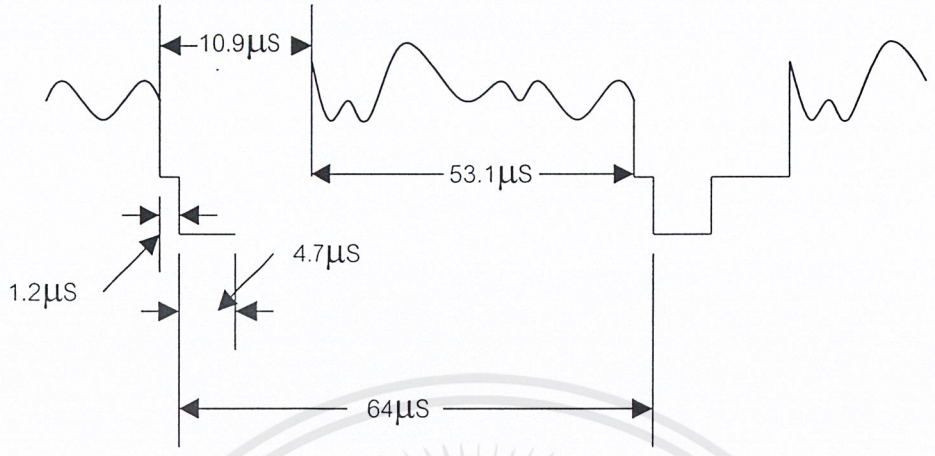
2.2 การคำนวณความถี่ที่ใช้ในการสุ่มสัญญาณภาพ

ในมาตรฐานของระบบโทรทัศน์ทั่วไปนั้นจะกำหนดคุณลักษณะของสัญญาณภาพรวม ซึ่งจะพิจารณาถึงเวลาในการสแกนทางแนวนอนหนึ่งเส้นว่า ใช้เวลาเท่าไรจากนั้นก็นำเวลาดังกล่าวมาทำการหารด้วยจำนวนจุดภาพตามต้องการเพื่อที่จะได้เวลาในการสุ่มแต่ละจุดในระบบโทรทัศน์ของประเทศไทยจะเป็นระบบ PAL มีพารามิเตอร์ของการสแกนตามตารางที่ 2.1 และภาพที่ 2.1

ตารางที่ 2.1 แสดงพารามิเตอร์ของระบบโทรทัศน์สามระบบคือ NTSC, PAL และ SECAM

Scanning	NTSC	PAL	SECAM
Line / frame	525	625	625
Frames / second	30	25	25
Interlace ratio	2 : 1	2 : 1	2 : 1
Aspect ratio	4 : 3	4 : 3	4 : 3
Color subcar. (Hz)	3579545	4433619	4.406250

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 2.1 แสดงรายละเอียดของสัญญาณซิงค์ทางแนวนอน

จากตารางที่ 2.1 จะเห็นได้ว่าความถี่ในการสแกนทางแนวนอนของระบบ PAL เท่ากับ 15,625 Hz (625×25) หรือเท่ากับ 64 μs ต่อหนึ่งเส้นสแกนไลน์ แต่มีช่วงเวลาของการเกิดสัญญาณภาพจริงเพียง 53.1 $64\mu s$ จากเวลาที่ได้นี้นำไปคำนวณหาความถี่ของการสุ่มด้วยสมการตามนี้

$$T_s = \frac{T_{hor}}{N} = \frac{53.1\ \mu s}{512} = 1.037 \times 10^{-7}\ \ S \tag{2.1}$$

T_s คือเวลาในการสุ่มสัญญาณภาพแต่ละจุด

T_{hor} คือเวลาในการสแกนหนึ่งเส้นทางแนวนอนของโทรทัศน์

N คือจำนวนจุดภาพที่ต้องการในหนึ่งเส้นสแกนไลน์

ดังนั้นความถี่ในการสุ่มแต่ละจุดภาพเท่ากับ

$$f_s = \frac{1}{T_s} = \frac{1}{1.037 \times 10^{-7}} = 9.64\ \ MHz \tag{2.2}$$

จากความถี่ที่คำนวณได้ในการสุ่มขนาด 512 จุดภาพต่อหนึ่งเส้นสแกนไลน์นี้ จะเห็นเป็นการยากที่จะสร้างวงจรเพื่อกำเนิดสัญญาณให้มีความถี่เท่านี้ จึงเลือกความถี่ที่สามารถกำเนิดได้ง่ายคือ 12 MHz

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 แนวทางการออกแบบ

จากวิธีการที่กล่าวผ่านมาแล้วนั้นก็จะถึงขั้นตอนของการออกแบบวงจรเพื่อให้มีการทำงานตามจุดประสงค์ ในหัวข้อนี้จะได้กล่าวถึงการออกแบบแผงวงจรเก็บข้อมูลภาพ โดยใช้ตัวแปลงสัญญาณแอนะล็อกที่ความถี่ 12 MHz และรายละเอียดของวงจรแสดงไว้ในภาพที่ ก-2 ที่ภาคผนวก ก โดยรายละเอียดในการออกแบบและการทำงานของแต่ละบล็อกนั้นจะได้กล่าวต่อไปในหัวข้อย่อยต่อไปตามนี้

2.3.1 ส่วนของอินพุตบัฟเฟอร์

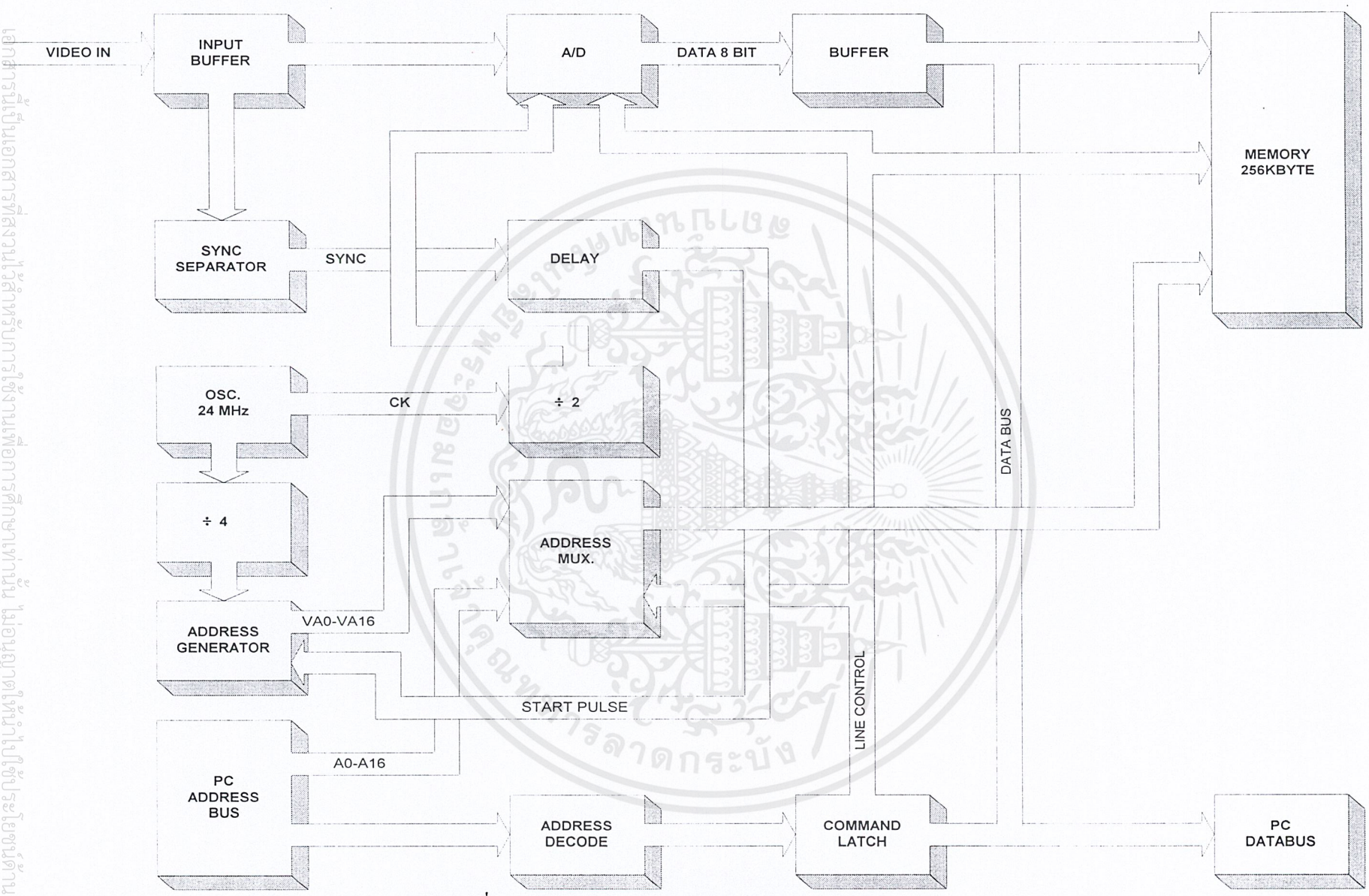
หน้าที่ส่วนนี้คือการปรับระดับแรงดันของสัญญาณภาพเพื่อส่งต่อไปในส่วนของการแยกสัญญาณซิงค์ และส่วนของ ตัวแปลงสัญญาณแอนะล็อกเป็นดิจิตอล

2.3.2 ส่วนของการแยกซิงค์

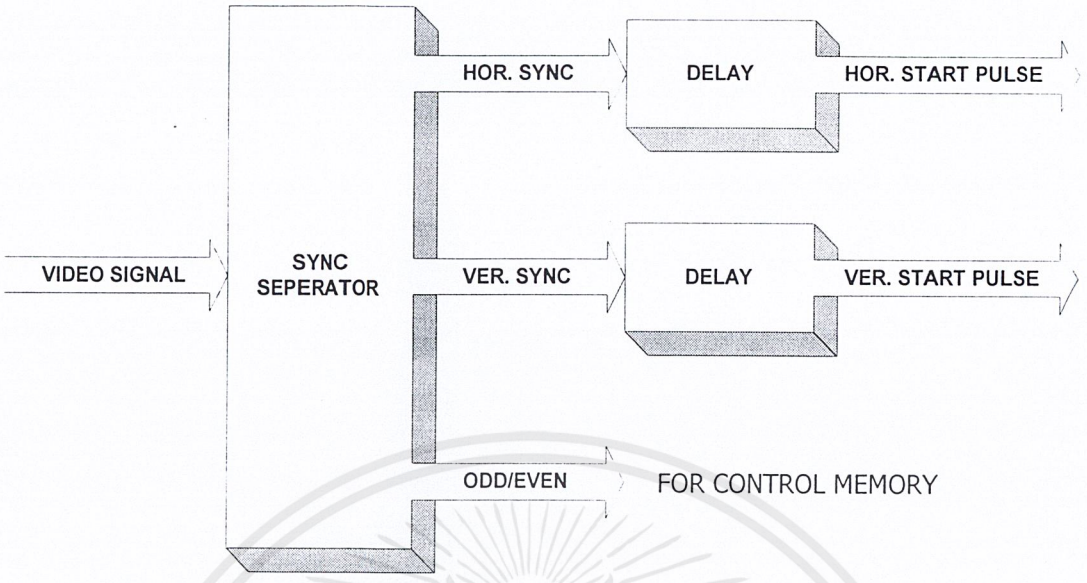
ส่วนของการแยกซิงค์เป็นตัวแยกสัญญาณซิงค์ทางแนวนอน ซิงค์ทางแนวตั้ง และสัญญาณบ่งบอกฟิลด์คี่หรือฟิลด์คู่จากสัญญาณภาพรวม โดยเอาต์พุตที่ได้จะเป็นพัลซ์ซึ่งเป็นระดับแรงดันที่ใช้กับไอซีจำพวก TTL ซึ่งสัญญาณต่างๆ เช่น สัญญาณซิงค์ทางแนวนอนกับสัญญาณซิงค์ทางแนวตั้งจะถูกส่งต่อไปยังส่วนของการมัลติเพล็กซ์เพื่อรวมกับสัญญาณภาพที่ได้จากส่วนของเอาต์พุตสำหรับแสดงผลออกทางจอมอนิเตอร์ ขณะเดียวกันสัญญาณซิงค์ทั้งสองจะถูกนำไปเป็นตัวควบคุมการนับของตัวกำเนิดสัญญาณแอดเดรสเพื่อให้การอ้างตำแหน่งการจัดเก็บหรือการอ่านข้อมูลทำได้อย่างถูกต้อง สัญญาณซิงค์นี้จะถูกหน่วงเวลาสำหรับการเก็บจุดภาพทั้งนี้เพราะความถี่ที่ใช้ในการสุ่มเป็น 12 MHz แทนที่จะเป็น 9.6 MHz จึงต้องหน่วงเวลาให้เลยช่วงแบลงค์กิ้งไประยะหนึ่งก่อนที่จะทำการสุ่มให้ได้จำนวนจุดภาพที่กำหนด สำหรับตัวหน่วงนั้นจะทำหน้าที่ในการส่งพัลซ์ไปกดระตุ้นให้ตัวกำเนิดสัญญาณแอดเดรสทางแนวนอนและแนวตั้งตามลำดับ จากที่ได้กล่าวมานี้จะให้บล็อกไดอะแกรมดังแสดงในภาพที่ 2.3 ส่วนผังเวลาของการหน่วงทางแนวนอนแสดงได้ดังภาพที่ 2.4

ส่วนสัญญาณบ่งบอกฟิลด์คู่นั้นใช้อ้างตำแหน่งของหน่วยความจำว่าจะเป็นส่วนที่เป็นฟิลด์คู่หรือฟิลด์คี่ดังที่กล่าวมาแล้วในหัวข้อ การเก็บภาพลงหน่วยความจำ ในกรณีที่สัญญาณนี้มีค่าเป็น “0”จะเป็นการเก็บภาพที่เป็นข้อมูลฟิลด์ ขนาด 64 กิโลไบต์ และในกรณีกลับกันถ้าสัญญาณนี้มีค่าเป็น “1” จะเป็นการอ้างถึงหน่วยความจำที่เก็บข้อมูลภาพฟิลด์คี่ ขนาด 64 กิโลไบต์เช่นกัน

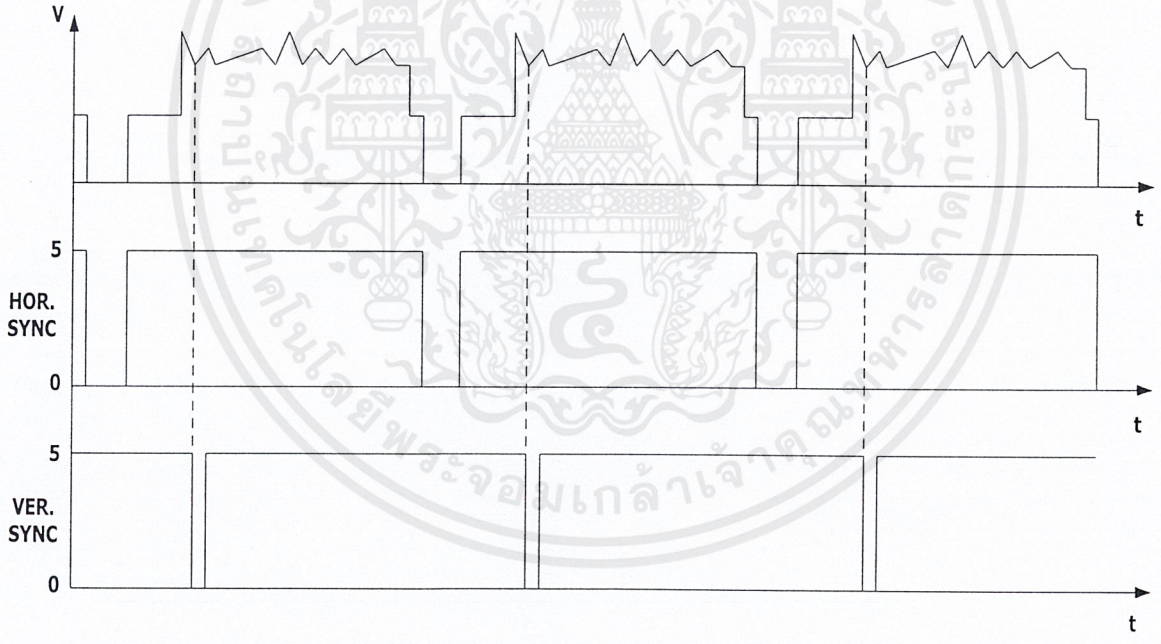
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 2.2 แสดงบล็อกไดอะแกรมของแผงวงจรเก็บข้อมูลภาพ



ภาพที่ 2.3 แสดงส่วนของการแยกสัญญาณซิงค์



ภาพที่ 2.4 แสดงผังเวลาของการหน่วงเวลาทางแนวนอน

2.3.3 ส่วนของการกำเนิดสัญญาณนาฬิกา

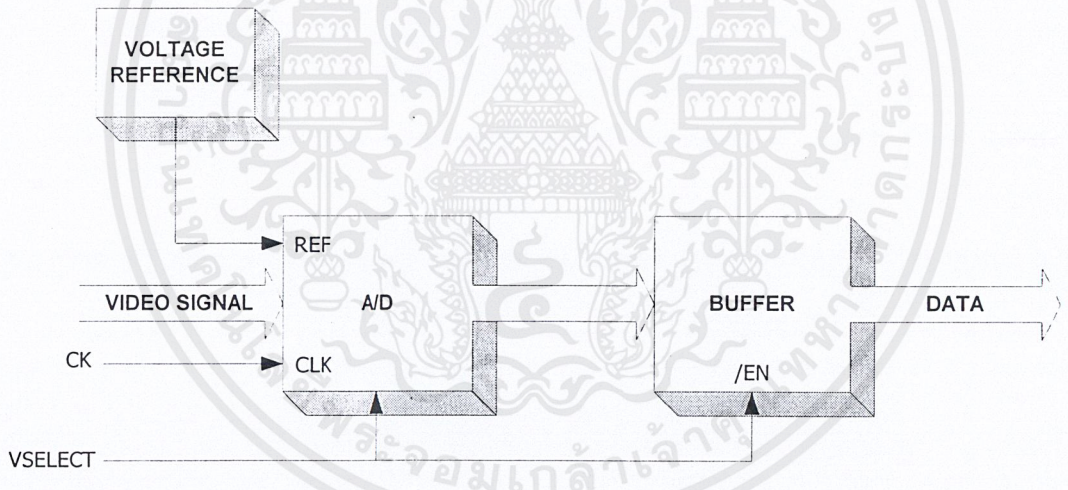
ส่วนกำเนิดสัญญาณนาฬิกาที่ใช้ในการสุ่มนั้นจะใช้คริสตอล 24MHz ต่อร่วมกับอินเวอร์ติ้งเกต โดยสัญญาณนาฬิกา 24MHz นั้นจะถูกนำไปใช้สองด้านด้วยกัน ทางที่หนึ่งจะใช้เป็นสัญญาณนาฬิกาสำหรับตัวนับเพื่อสร้างสัญญาณแอดเดรส และสัญญาณซิงค์ ส่วนการใช้งานอีกด้านหนึ่งจะนำไปหารความถี่ลงสองเท่าด้วย D flip-flop 1 ชุด ให้เหลือความถี่ 12 MHz เพื่อนำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไปใช้เป็นความถี่ในการสุ่มของตังแปลงสัญญาณแอนะล็อกเป็นดิจิตอล แล้วทำการหารลงอีกสองเท่าด้วย D flip-flop อีก 1 ชุด เพื่อให้เหลือความถี่ 6 MHz ใช้เป็นสัญญาณแอดเดรส A0 บนแผงวงจรเก็บข้อมูลภาพ

2.3.4 ส่วนของการแปลงสัญญาณแอนะล็อกเป็นดิจิตอล

ในส่วนนี้ประกอบไปด้วย ตัวแปลงสัญญาณแอนะล็อกเป็นดิจิตอลโดยใช้ความถี่ในการสุ่มขนาด 12 MHz ข้อมูลที่ออกจากตัวแปลงสัญญาณแอนะล็อกเป็นดิจิตอลทั้งสองชุดนั้น จะถูกส่งผ่านเข้าสู่หน่วยความจำทางบัฟเฟอร์โดยมีสัญญาณควบคุมการทำงานด้วยสัญญาณ VSELECT ที่มาจากส่วนควบคุม สำหรับควบคุมจังหวะการติดต่อกับหน่วยความจำ และส่วนประกอบอีกส่วนหนึ่งคือส่วนของแรงดันอ้างอิงที่จ่ายให้กับ ตัวแปลงสัญญาณแอนะล็อกเป็นดิจิตอล นั้นสามารถปรับให้เหมาะสมกับสัญญาณภาพที่เข้ามาได้ สำหรับภาพที่ 2.5 เป็นบล็อกไดอะแกรมของ ส่วนตัวแปลงสัญญาณแอนะล็อกเป็นดิจิตอล



ภาพที่ 2.5 แสดงบล็อกไดอะแกรมของส่วนการแปลงสัญญาณแอนะล็อกเป็นดิจิตอล

2.3.5 ส่วนของการมัลติเพล็กซ์

วงจรมัลติเพล็กซ์จประกอบด้วยส่วนสำคัญ 3 ส่วนดังนี้

- ส่วนของการมัลติเพล็กซ์สัญญาณซิงค์

ทำหน้าที่เลือกระหว่างสัญญาณซิงค์ทางแนวนอน ซิงค์ทางแนวตั้ง และสัญญาญบ่งบอกฟิลด์คี่และฟิลด์คู่ โดยสัญญาณซิงค์ทางแนวนอนจะถูกนำไปกำหนดการเริ่มนับของตัวนับทางแนวนอน สัญญาณซิงค์ทางแนวตั้งนั้นก็จะเป็นตัวกำหนดการเริ่มนับของตัวนับทางแนวตั้ง สัญญาณบ่งบอกฟิลด์คี่และฟิลด์คู่นั้นจะใช้เป็นตัวอ้างถึงตำแหน่งของหน่วยความจำในส่วนของฟิลด์คี่และฟิลด์คู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ส่วนของการมัลติเพล็กซ์แอดเดรส

หน้าที่ของส่วนนี้คือการเลือกระหว่างแอดเดรสสองชุด ชุดแรกเป็นแอดเดรสจากตัวนับส่วนชุดหลังเป็นแอดเดรสจากเครื่องคอมพิวเตอร์ ในกรณีที่มีการเก็บข้อมูลภาพลงหน่วยความจำหรืออ่านข้อมูลภาพจากหน่วยความจำเพื่อนำไปแสดงออกทางจอมอนิเตอร์ จะมีการอ้างถึงตำแหน่งของหน่วยความจำ ในกรณีนี้ต้องสัมพันธ์กับสัญญาณซิงค์ของภาพ เพื่อให้มีการเก็บภาพที่ถูกต้องหรือแสดงข้อมูลาภพออกมาได้ถูกต้อง ดังนั้นจึงใช้แอดเดรสที่สร้างจากตัวนับทั้งสองชุดที่มีการควบคุมการนับจากสัญญาณซิงค์สำหรับกรณีที่ต้องการนำข้อมูลภาพจากหน่วยความจำ บนแผงวงจรควบคุมหรือการนำข้อมูลจากดิสเก็ตลงสู่หน่วยความจำบนแผงวงจรควบคุม จะต้องใช้แอดเดรสจากเครื่องคอมพิวตอร์โดยตรงซึ่งการควบคุมแอดเดรสในการอ่านหรือเขียนข้อมูลนั้นจะขึ้นอยู่กับเครื่องคอมพิวเตอร์ที่เป็นตัวควบคุม

- ส่วนของการมัลติเพล็กซ์สัญญาณควบคุม

การมัลติเพล็กซ์สัญญาณควบคุมในส่วนนี้จะเป็นการเลือกสัญญาณที่จะนำไปควบคุมการทำงานของแผงวงจรเก็บข้อมูลภาพ

2.3.6 ส่วนการกำเนิดสัญญาณแอดเดรส

ในการสร้างสัญญาณแอดเดรสเพื่อใช้ในการอ้างตำแหน่ง ของหน่วยความจำบนแผงวงจรควบคุมการเก็บภาพ สามารถแบ่งออกได้เป็น 2 ส่วนด้วยกันคือ ส่วนของวงจรสร้างสัญญาณแอดเดรสสำหรับการอ้างตำแหน่งในการเก็บภาพทางแนวนอน สัญญาณแอดเดรสและส่วน ของวงจรสร้างสัญญาณแอดเดรสสำหรับการอ้างตำแหน่งในแนวตั้ง ซึ่งการสร้างแอดเดรสทั้งสองส่วนสามารถทำได้ดังนี้

- ตัวนับทางแนวนอน

จากการที่ต้องทำการเก็บภาพทางแนวนอนขนาด 512 จุดต่อหนึ่งเส้นสแกนไลน์ด้วยความถี่ 512 MHz ทำให้ต้องมีการอ้างตำแหน่งของหน่วยความจำ 512 ตำแหน่ง จากการออกแบบสายแอดเดรส A1- A8 ที่มีขนาด 8 เส้น สำหรับตัวนับขนาด 8 บิต ทำการป้อนอินพุตด้วยสัญญาณนาฬิกาขนาด 6MHz ที่สำคัญคือการเริ่มต้นนับต้องให้สัมพันธ์กับสัญญาณภาพทางแนวนอนจึงต้องใช้พัลซ์ที่มาจากตัวหน่วงเพื่อช่วยในการกำหนดการเริ่มเก็บข้อมูลภาพทางแนวนอนตามที่ได้กล่าวไว้แล้วในส่วนของการแยกซิงค์ หัวข้อ ส่วนของการแยกซิงค์ หลักจากทำการนับครบ 256 ตำแหน่งแล้วตัวนับก็จะทำการส่งพัลซ์ออกมาหยุดการนับพร้อมกับรีเซ็ตให้แอดเดรสทั้งหมดมีค่าเป็นศูนย์เพื่อรอการกระตุ้นจากสัญญาณพัลซ์ที่มีจากตัวหน่วงใหม่แล้วจึงจะเริ่มนับใหม่ จะเห็นได้ว่ามีการนับเพียงแค่ 256 ตำแหน่งเท่านั้น จึงใช้สัญญาณนาฬิกาขนาด 6 MHz เป็นสัญญาณแอดเดรส A0 เพิ่มเข้าไปทำให้สามารถนับได้ครบทั้ง 512 ตำแหน่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ตัวนับทางแนวตั้ง

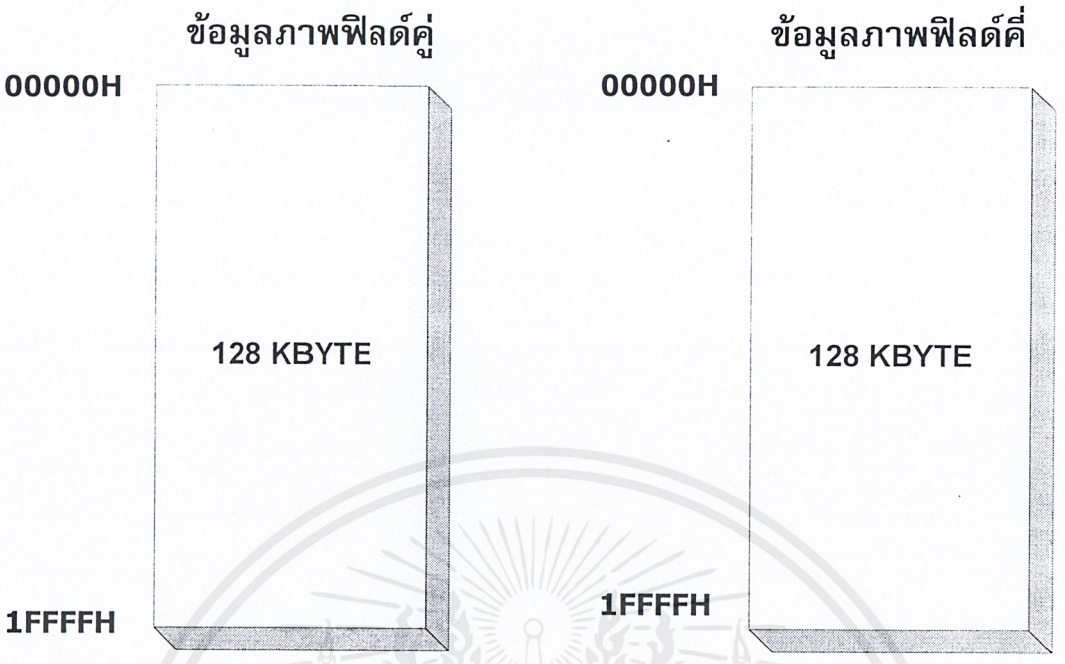
ในการกำเนิดสัญญาณแอดเดรสเพื่อใช้เก็บภาพจำนวน 256 เส้นต่อหนึ่งฟิลด์ภาพ ก็มีลักษณะการออกแบบเช่นเดียวกับการสร้างสัญญาณแอดเดรสทางทางแนวนอน โดยจะใช้ตัวนับขนาด 8 บิต แต่สัญญาณที่จะเข้ามาเป็นอินพุดจะแตกต่างออกไป เพราะการเปลี่ยนแปลงตำแหน่งของแอดเดรสทางแนวตั้งนั้นจะใช้เวลาเท่ากับหนึ่งเส้นสแกนไลน์ดังนั้นจึงใช้สัญญาณพัลซ์ที่เกิดจากตัวนับทางแนวนอนมาเป็นอินพุดของตัวนับทางแนวตั้งแทน ซึ่งเมื่อนับครบ 256 เส้นแล้วก็จะทำการหยุดและรีเซ็ตตัวเอง ซึ่งจะมีการทำงานในลักษณะเดียวกับการสร้างสัญญาณแอดเดรสทางแนวนอนนั้นเอง

2.3.7 ส่วนของหน่วยความจำ

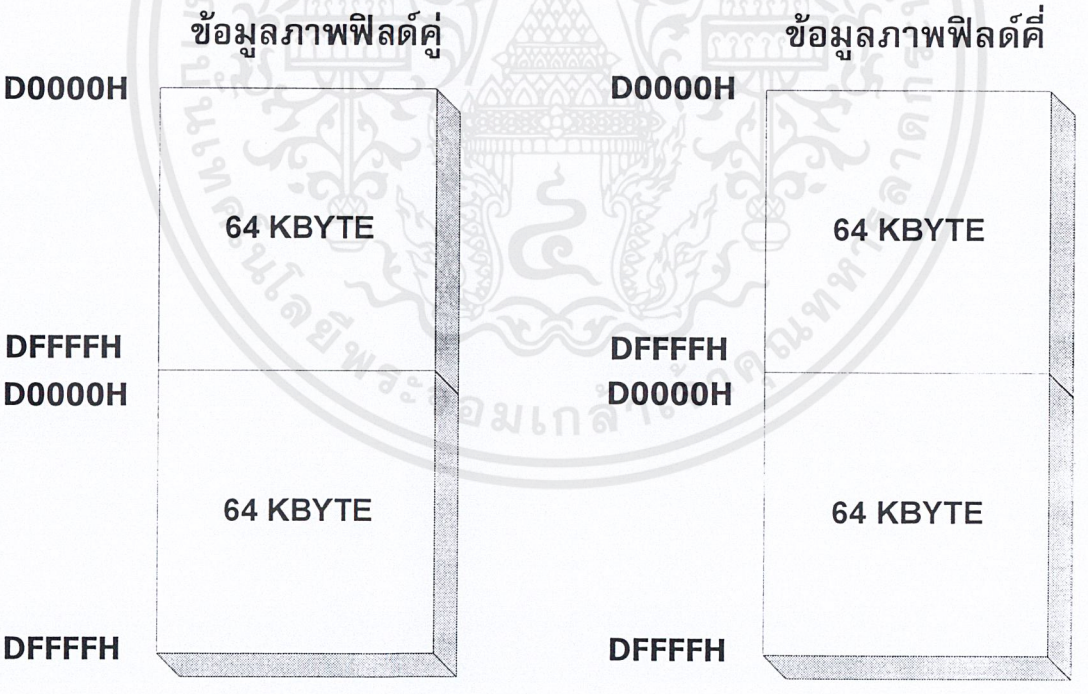
ตำแหน่งของหน่วยความจำในส่วนนี้แบ่งออกได้เป็นสองลักษณะ ลักษณะแรกเป็นตำแหน่งที่แผงวงจรเก็บข้อมูลภาพ ใช้อ้างเวลาเก็บข้อมูลภาพแอดเดรสที่ใช้อ้างตำแหน่งนี้จะได้จากส่วนของตัวนับ โดยมีตำแหน่งสำหรับการเก็บภาพเริ่มจาก 00000H-1FFFFH มีขนาด 128 กิโลไบต์ แสดงได้ดังภาพที่ 2.6 แต่หน่วยความจำที่ใช้เก็บข้อมูลภาพมีขนาด 256 กิโลไบต์ จึงใช้ RAM ขนาด 128 กิโลไบต์ 2 ตัวโดยตัวแรกจะใช้ในการเก็บข้อมูลของฟิลด์คู่ ส่วนอีกตัวจะใช้เก็บข้อมูลฟิลด์คี่ โดยใช้สัญญาณ O/E_MUX จากส่วนแยกสัญญาณซิงค์ เป็นตัวบ่งบอกการเก็บข้อมูลฟิลด์คู่และฟิลด์คี่ดังกล่าว

ลักษณะที่สองคือการอ้างตำแหน่งของเครื่องคอมพิวเตอร์ ลักษณะการอ้างตำแหน่งในกรณีนี้ต้องขึ้นอยู่กับตำแหน่งของหน่วยความจำที่ระบบจัดเตรียมไว้ให้ หน่วยความจำที่ว่างไว้สำหรับเครื่องคอมพิวเตอร์ที่กำหนดไว้ให้สำหรับอุปกรณ์ที่จะนำมาเชื่อมต่อด้วยนั้นมีขนาดสูงสุดเพียงแค่ 64 กิโลไบต์ มีตำแหน่งอยู่ในช่วง D0000H-DFFFFH การอ้างตำแหน่งของเครื่องคอมพิวเตอร์แสดงได้ดังภาพที่ 2.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 2.6 แสดงการอ้างตำแหน่งหน่วยความจำบนแผงวงจรเก็บข้อมูลภาพ

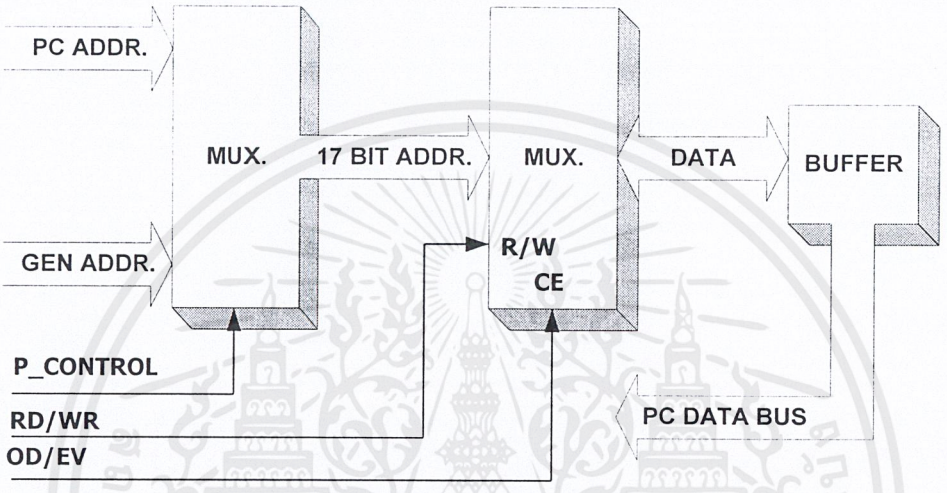


ภาพที่ 2.7 แสดงการอ้างตำแหน่งของเครื่องคอมพิวเตอร์

เนื่องจากขนาดของข้อมูลที่อยู่บนแผงวงจรเก็บข้อมูลภาพ นั้นมีถึง 256 กิโลไบต์ ประกอบด้วยกลุ่มข้อมูลภาพจุดคี่และคู่ แต่ละกลุ่มมีขนาด 128 กิโลไบต์ ทำให้ต้องส่งสัญญาณควบคุมเพื่อเลือกการติดต่อกับหน่วยความจำบนแผงวงจรเก็บข้อมูลภาพ ว่าจะอ่านจากหน่วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความจำส่วนใดโดยมี P_CONTROL ร่วมกับ OD/EV เป็นสัญญาณที่ใช้ในการควบคุม สัญญาณ P_CONTROL จะใช้เป็นตัวเลือกว่าจะให้หน่วยความจำติดต่อกับแอดเดรสที่มาจากเครื่องคอมพิวเตอร์หรือติดต่อกับแอดเดรสที่มาจากส่วนของการกำหนดสัญญาณแอดเดรส ส่วนสัญญาณ OD/EV จะเป็นตัวเลือกว่าต้องการติดต่อกับหน่วยความจำที่เก็บข้อมูลภาพฟิลด์คี่หรือฟิลด์คู่ ดังบล็อกไดอะแกรมที่แสดงในภาพที่ 2.8



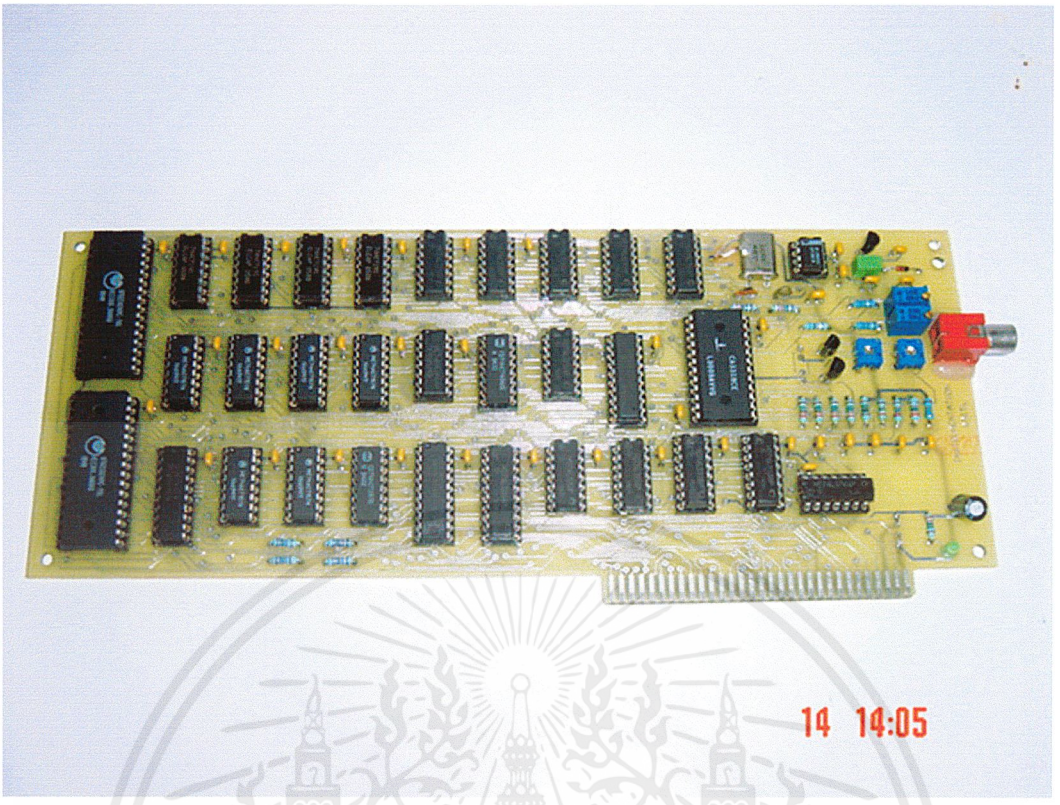
ภาพที่ 2.8 แสดงบล็อกไดอะแกรมของส่วนหน่วยความจำ

2.3.8 ส่วนของการควบคุม

ส่วนนี้จะเป็นตัวควบคุมการทำงานของวงจรเก็บข้อมูลภาพ ทั้งหมด โดยคำสั่งต่างๆ จะถูกส่งออกมาทางซอฟต์แวร์ ด้วยการส่งค่าออกมาที่พอร์ตเอาต์พุต ตำแหน่งของพอร์ตที่จะส่งค่าออกมานั้นจะเลือกใช้ตำแหน่งช่วง 310H ถึง 31FH ซึ่งเป็นตำแหน่งพอร์ตที่เครื่องคอมพิวเตอร์เผื่อไว้สำหรับอุปกรณ์อินเตอร์เฟสทั่วไป แต่การเลือกตำแหน่งพอร์ตต้องคำนึงถึงว่ามีอุปกรณ์อินเตอร์เฟสใดที่ใช้ซ้ำกันบ้าง สำหรับคำสั่งหลักๆ ที่ใช้คือ คำสั่งการเริ่มต้นเก็บภาพ คำสั่งเพื่อการอ่านข้อมูลภาพจาหน่วยความจำ เป็นต้น

โดยแผงวงจรเก็บข้อมูลภาพที่สร้างขึ้นและภาพที่เก็บได้จากวงจรเก็บข้อมูลภาพแสดงดังภาพที่ 2.9 และ 2.10 ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 2.9 แสดงแผงวงจรเก็บข้อมูลภาพที่สร้างขึ้น

ภาพที่ 2.10 แสดงภาพที่ได้จากแผงวงจรเก็บข้อมูลภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
สเตปมอเตอร์

3.1 กล่าวนำ

สเตปมอเตอร์ คืออุปกรณ์ที่เคลื่อนที่เป็นสเตปโดยการกระตุ้นด้วยวิธีการทางแม่เหล็กไฟฟ้า ซึ่งเปลี่ยนสัญญาณดิจิตอลทางอินพุทซึ่งเป็นพัลซ์ไปเป็นการเคลื่อนที่แบบอนาลอกที่เอาท์พุท สเตปมอเตอร์บางครั้งถูกเรียกว่า สเตปปิ้งมอเตอร์หรือ สเตปเปอร์มอเตอร์ ที่ได้สมญานามเช่นนี้ก็เพราะว่า มอเตอร์เป็นอุปกรณ์ซึ่งเคลื่อนที่เมื่อถูกกระตุ้นโดยโวลท์เตจหรือกระแส ซึ่งโดยมากจะเป็น ไฟฟ้ากระแสตรง เอาท์พุทของสเตปมอเตอร์จะมีจำนวนเท่ากับจำนวนของคำสั่งอินพุทซึ่งมีลักษณะเป็นพัลซ์ โดยเมื่อป้อนกระแสให้กับสเตปมอเตอร์แล้วสเตปมอเตอร์จะมีการเคลื่อนที่เพิ่มขึ้น 1 สเตป

สเตปมอเตอร์มีมาประมาณ 40 กว่าปีมาแล้ว สมัยก่อนการใช้งานสเตปมอเตอร์มีประสิทธิภาพต่ำสู้เอซีมอเตอร์และดีซีมอเตอร์ไม่ได้ แต่เมื่อไม่นานมานี้ได้มีการนำดิจิตอลคอมพิวเตอร์เข้ามาใช้จึงได้เปลี่ยนรูปแบบการควบคุมสเตปมอเตอร์ใหม่ โดยใช้การควบคุมด้วยไมโครโปรเซสเซอร์หรือไมโครคอนโทรลเลอร์ซึ่งจะช่วยให้ใช้งานสเตปมอเตอร์ได้สะดวกขึ้นและมีประโยชน์ในการใช้สอยมากขึ้น ในปัจจุบันนี้สเตปมอเตอร์ได้ถูกนำมาใช้ในงานอุปกรณ์ที่ใช้งานร่วมกับคอมพิวเตอร์เช่น พริ้นเตอร์ ดิสค์ไดรว์และนอกจากนี้ยังถูกใช้ในอุปกรณ์ที่ใช้ในการควบคุมเชิงเลข (NUMERICAL CONTROL) การควบคุมกระบวนการ (PROCESS CONTROL) และการควบคุมอุปกรณ์ทางเครื่องมือกล (MACHINE TOOL CONTROL) เป็นต้น

การควบคุมการทำงานของสเตปมอเตอร์โดยทั่วไปใช้วงจรขับ ซึ่งให้สเตปมอเตอร์ตอบสนองต่อสัญญาณพัลซ์ซึ่งทำให้เกิดการออสซิลเลทระหว่างสเตปของการเคลื่อนที่ทำสเตปมอเตอร์สั่นระหว่างใช้งาน การแก้ปัญหาดังกล่าวกระทำได้ด้วยวิธีการแด้มพ์ (DAMPING METHOD) ของสเตปมอเตอร์ ซึ่งวิธีนี้เป็นการแก้ปัญหาตามลักษณะเฉพาะของมอเตอร์แต่ละตัว

3.2 ทฤษฎีพื้นฐานและหลักการการทำงานของสเตปมอเตอร์

อีกแนวทางหนึ่งที่ใช้แก้ปัญหาของการติดต่อระหว่าง คอมพิวเตอร์กับมอเตอร์ ก็คือ การใช้มอเตอร์ที่ง่ายต่อการควบคุมโดยคอมพิวเตอร์ สเตปมอเตอร์เป็นมอเตอร์ที่เหมาะกับงานเช่นนี้มาก เพราะ มันสามารถควบคุมให้หมุนได้โดยใช้สภาวะเพียง 2 สถานะ (NO – OFF) ซึ่งสัญญาณที่ว่านี้เครื่องคอมพิวเตอร์สามารถสร้างขึ้นได้ง่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สเตปมอเตอร์มีขดลวดอยู่หลายขด โดยที่แต่ละขดควบคุมการจ่ายกระแสโดยสัญญาณ NO- OFF ทำให้สามารถควบคุมตำแหน่งการเคลื่อนที่ได้ตราบใดเท่าที่ยังไม่เกิด การเกินกำลัง (OVERLOAD) ตราบนั้นมันยังคงรักษาตำแหน่งเดิม จนกระทั้งมีการกระตุ้นการจ่ายกระแสอีกครั้ง และเนื่องจากสเตปมอเตอร์สามารถควบคุมตำแหน่งได้โดยการส่งสัญญาณกระตุ้นทำให้ไม่จำเป็นต้องใช้ตัววัดตำแหน่ง (ENCODER) ในการทำงาน ข้อเสียของสเตปมอเตอร์คือ มันมีข้อจำกัดด้านความเร็วเมื่อเปรียบเทียบกับ DC SERVO MOTOR ที่มีขนาดเท่ากันและยังมีการสั่นสะเทือน (VIBRATION) สูงทำให้การเคลื่อนที่เป็นไปอย่างไม่ราบเรียบ

3.2.1 หลักการทำงานของสเตปมอเตอร์

สเตปมอเตอร์สามารถแบ่งโครงสร้างทางกายภาพออกเป็น 2 ส่วน คือ สเตเตอร์ (STATOR) และโรเตอร์ (ROTOR) ตัวสเตเตอร์เป็นส่วนที่อยู่กับที่ ประกอบด้วย ขดลวดทองแดงซึ่งพันอยู่รอบแกนเหล็ก เพื่อสร้างสนามแม่เหล็กเมื่อมีการจ่ายกระแสผ่านขดลวด ส่วนโรเตอร์เป็นส่วนที่เคลื่อนที่ได้ มีลักษณะเป็นแท่งเหล็กทรงกลม และที่ผิวรอบนอกมีลักษณะเป็นซีกฟันซึ่งทำจากแม่เหล็กถาวร

เมื่อยังไม่มีการจ่ายกระแสให้กับขดลวดของมอเตอร์ ซีกฟันอันใดอันหนึ่งของโรเตอร์จะอยู่ในตำแหน่งที่ตรงกันกับซีกฟันอันใดอันหนึ่งของสเตเตอร์ ทั้งนี้เป็นเพราะแม่เหล็กถาวรที่ตัวมอเตอร์ที่ตัวของโรเตอร์พยายามที่จะทำให้ความต้านทานทางแม่เหล็กไฟฟ้า (RELUCTANCE) มีค่าน้อยที่สุด ซึ่ง ณ ที่จุดนี้ซีกฟันของตัวโรเตอร์และสเตเตอร์ตรงกันนั้นมีค่าความต้านทานทางแม่เหล็กไฟฟ้าน้อยที่สุด ทำให้เกิดเส้นแรงแม่เหล็ก – ไฟฟ้ามากที่สุด และจากรูป เส้นแรงแม่เหล็กไฟฟ้าจะทำให้เกิดขั้วแม่เหล็กเหนือและใต้ขึ้นมา 2 คู่ ทั้งที่ตัวสเตเตอร์และตัวโรเตอร์ดังภาพ ค่าทอร์ค (TORQUE) ที่ทำให้ตัวโรเตอร์สามารถยึดอยู่ในตำแหน่งดังกล่าวนี้เรียกว่า ดีเท็นท์ทอร์ค (DETENT TORQUE) (หมายความว่า การที่จะทำให้มอเตอร์เคลื่อนที่ในขณะที่ไม่ได้จ่ายกระแส ให้กับขดลวดของมอเตอร์จะต้องออกแรงมากกว่าค่าของดีเท็นทอร์ค จึงจะทำให้โรเตอร์เริ่มหมุนได้) จากภาพข้างต้น นั้นมี 12 สเตป

เมื่อจ่ายกระแสให้กับขดลวดที่อยู่ในสเตเตอร์คู่ใดคู่หนึ่ง ดังรูปข้างล่าง จะทำให้เกิดขั้วแม่เหล็กเหนือและใต้ที่ซีกฟันของตัวสเตเตอร์ ซึ่งจะดึงดูดซีกฟันของตัวโรเตอร์ที่มีขั้วแม่เหล็กที่มีศักย์ต่างกันที่อยู่ใกล้ที่สุดเข้าไว้ ตำแหน่งนี้เรียกว่าสเตเบิ้ลโพสซิชั่ (STABLE POSITION) ของโรเตอร์ จะมีจำนวนเท่ากับจำนวนซีกฟันของโรเตอร์ และแรงที่จะทำให้ โรเตอร์เปลี่ยนตำแหน่งไปจากตำแหน่งสเตเบิ้ลโพลซิชั่นไดนี้เรียกว่า โฮลดิ้ง ทอร์ค (HOLDING TORQUE) เมื่อสับเปลี่ยนการจ่ายกระแสให้แก่ขดลวด จากขดหนึ่งไปยังอีกขดหนึ่ง เนื่องจากขดลวดวางอยู่ในตำแหน่งที่ต่างกัน I' ก็จะทำให้ตัวสเตเตอร์ดึงดูดซีกฟันของตัวโรเตอร์อีกซีกหนึ่งที่ใกล้ที่สุดเข้าไว้ ซึ่งจะทำให้ตัวโรเตอร์เคลื่อนที่ไป 1 สเตปหรือ 30° จากนั้นก็เปลี่ยนไป จ่ายกระแสให้กับขดลวด ชุดแรกโดยใน

เอกสารนี้มีลิขสิทธิ์ ไม่อนุญาตให้ทำซ้ำ ดัดแปลง เพื่อการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คราวนี้เปลี่ยนทิศทางการไหลของกระแสให้ตรงกันข้ามกับครั้งแรก ซึ่งทำให้ตัว โรเตอร์เคลื่อนที่ไปอีก 1 สเตป (เคลื่อนที่ไป 30°) หลังจากนั้นก็ไปจ่ายกระแสให้กับขดลวดชุดที่สอง โดยกลับทิศทางของกระแสที่ป้อนให้อีกเช่นกันทำให้โรเตอร์หมุนไป 90° ถ้าหากเราต้องการเคลื่อนที่หนึ่งรอบ เราต้องทำการกระตุ้นให้มอเตอร์เคลื่อนที่ไปจนครบ 12 สเตป และถ้าต้องการให้โรเตอร์หมุนไปอีก ทิศทางหนึ่ง ก็ทำการสลับลำดับการจ่ายกระแส ตามลำดับ

ถ้าจ่ายกระแสให้แก่ขดลวดทั้งสองพร้อม ๆ กัน ซีกฟันของโรเตอร์จะอยู่ ณ ตำแหน่งระหว่างซีกฟันของสเตเตอร์ เพราะฉะนั้นการจ่ายกระแสให้แก่มอเตอร์แบบนี้จะให้ทอร์คมากกว่าแบบที่จ่ายกระแสในเวลาขณะใดขณะหนึ่งเพียงขดเดียว การป้อนกระแสให้กับขดลวดแต่ละขดเพื่อให้สเตปมอเตอร์เคลื่อนที่ แต่ถ้าในตอนแรกจ่ายกระแสให้กับขดลวดเพียงขดเดียวแล้ว สเตปต่อไปจะจ่ายกระแสให้กับขดลวดพร้อม ๆ กันสองขดลวด

ทำอย่างนี้สลับกันไปจะทำให้โรเตอร์เคลื่อนที่ไปสเตปละ 15° และในการขับเคลื่อนแบบนี้จะทำให้สเตปต่อรอบ (STEP/REV) เพิ่มขึ้นเป็นเท่าตัว การขับสเตปมอเตอร์แบบนี้เรียกว่า การขับแบบครึ่งสเตป (HALF STEPPING) ซึ่งนิยมใช้กันในงานอุตสาหกรรมเป็นอันมาก แม้บางครั้งจะให้ทอร์คน้อยกว่า แต่มีข้อดีกว่าการขับแบบเต็มสเตป (FULL STEPPING) ก็คือ การเคลื่อนที่เป็นไปอย่างราบเรียบ (ที่ความเร็วไม่สูงมากนัก) อีกทั้งมีความแม่นยำสูงและการ สั่นสะเทือนน้อยกว่า

3.2.2 ชนิดของสเตปมอเตอร์

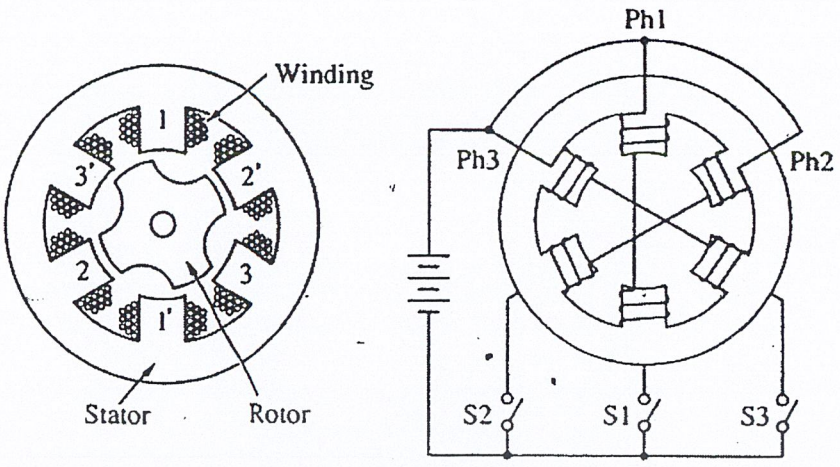
สเตปมอเตอร์สามารถแบ่งได้หลายชนิดตามลักษณะโครงสร้างและการใช้งานดังต่อไปนี้

3.2.2.1 สเตปมอเตอร์ชนิดปรับค่ารีลัคแตนซ์ได้ (VARIABLE RELUCTANCE STEPPING MOTOR)

สเตปมอเตอร์ชนิดนี้สามารถปรับค่ารีลัคแตนซ์ได้ ซึ่งภาพที่ 3.1 แสดงภาพตัดขวางของสเตปมอเตอร์แบบ 3 เฟส โดยที่สเตเตอร์มีฟันทั้งหมด 6 ซี่ ซี่ที่อยู่ตรงข้ามกันหรือทำมุม 180° ซึ่งกันและกันจะเป็นเฟสเดียวกัน ขดลวดที่พันอยู่ที่ฟันของสเตเตอร์ในแต่ละเฟส จะต่ออนุกรมหรือขนานก็ได้ จากภาพที่ 3.1 เป็นการต่อแบบอนุกรม ส่วนโรเตอร์นั้นมีฟัน 4 ซี่ ทั้งโรเตอร์และสเตเตอร์ทำมาจากโลหะซิลิกอน ซึ่งมีสภาพซึมซับทางแม่เหล็กสูงและยอมให้ สนามแม่เหล็กจำนวนมากไหลผ่านได้ ฟันของสเตเตอร์ในเฟสเดียวกันจะมีขั้วต่างกันโดยซี่ฟัน โดยซี่ I, II, III เป็นขั้วเหนือและซี่ I′, II′, III′ เป็นขั้วใต้หลังจากถูกกระตุ้น

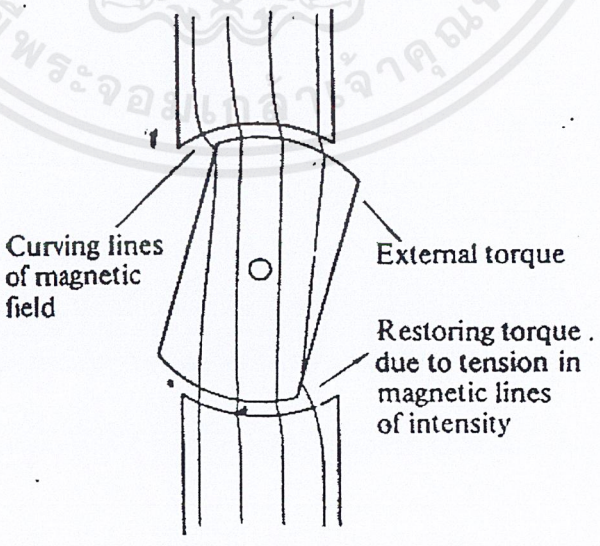
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



ภาพที่ 3.1 แสดงภาพตัดขวางของสเตปมอเตอร์แบบ 3 เฟส

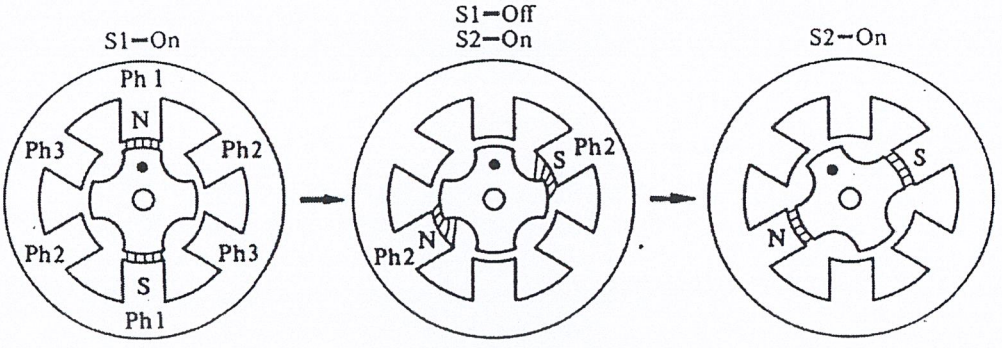
ภาพที่ 3.2 แสดงตำแหน่งสมดุลย์ เมื่อเฟสใดเฟสหนึ่งของสเตปมอเตอร์ถูกกระตุ้น



ภาพที่ 3.3 แสดงแรงภายนอกที่มีผลต่อเส้นแรงแม่เหล็ก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

45848



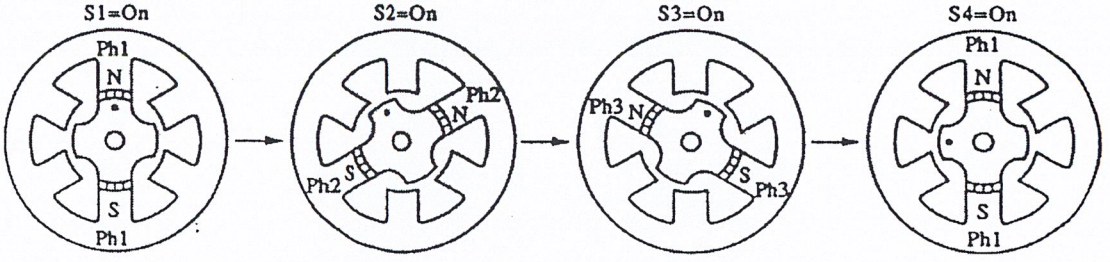
ภาพที่ 3.4 แสดงขั้นตอนการเคลื่อนที่ของโรเตอร์เมื่อสเตปมอเตอร์ถูกกระตุ้น

กระแสที่ไหลในแต่ละเฟสถูกควบคุมโดยสวิทช์ ปิด/เปิด ถ้าเฟส I ถูกกระตุ้นจะมีกระแสไหลและเกิดฟลักซ์แม่เหล็กดังแสดงในภาพที่ 3.2 แกนโรเตอร์จะอยู่ตำแหน่งเดียวกับซี่ I และ I′ ทำให้ทั้งโรเตอร์และสเตเตอร์อยู่ในแนวเดียวกันกรณีนี้จะทำให้ค่ารีลัคแตนซ์มีค่าน้อยที่สุดซึ่งเป็นตำแหน่งที่สมดุลย์ ถ้าโรเตอร์ถูกกระทำจากแรงภายนอกจะทำให้เปลี่ยนตำแหน่ง ดังภาพที่ 3.3 แรงบิดกระทำกับโรเตอร์ในทิศตามเข็มนาฬิกาทำให้ตำแหน่งเปลี่ยนไป มีผลทำให้เส้นแรงแม่เหล็กเคลื่อนที่จากซี่ของโรเตอร์ และสเตเตอร์ เมื่อโรเตอร์ และสเตเตอร์ไม่ได้อยู่ในแนวเดียวกันแล้วค่ารีลัคแตนซ์จะมีค่ามากจากนั้นสเตปมอเตอร์จะทำตัวให้มีค่ารีลัคแตนซ์น้อยที่สุด พอเฟส II ถูกกระตุ้นดังภาพที่ 3.4 โรเตอร์ถูกแรงภายนอกกระทำให้เคลื่อนที่ไป 30° ในทิศทวนเข็มนาฬิกา จากนั้นก็จะย้ายจากมุมที่เกิดการกระตุ้นแต่ละครั้งให้กลับไปยังตำแหน่งเดิมเรียกว่า สเตป

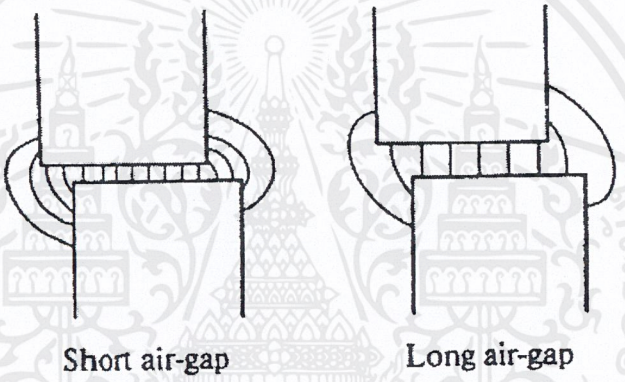
- ช่องว่าง (AIR GAP) ต้องมีขนาดเล็กที่สุด

ช่องว่างระหว่างฟันของโรเตอร์และสเตเตอร์ต้องเล็กที่สุดเท่าที่จะเป็นไป ได้เพื่อให้ทอร์คที่เกิดขึ้นมีค่ามากและมีความแม่นยำทางตำแหน่งมากขึ้น ภาพที่ 3.6 แสดงการเปรียบเทียบระหว่างช่องว่างที่กว้างและแคบในขณะที่แหล่งกำเนิดสนามแม่เหล็กแหล่งเดียวกันหรือมีระดับความรุนแรงเท่ากันช่องอากาศที่เล็กกว่าจะให้ทอร์คมากกว่าและทำให้เคลื่อนที่จากจุดสมดุลย์น้อยกว่าช่องว่างขนาดใหญ่เมื่อมีแรงบิดจากภายนอกมากระทำต่อโรเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.5 แสดงขั้นตอนการเคลื่อนที่ของสเตปมอเตอร์

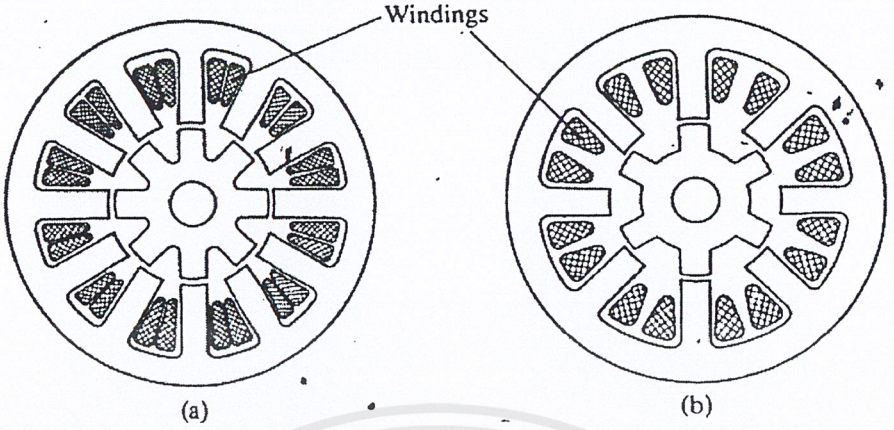


ภาพที่ 3.6 แสดงการเปรียบเทียบเส้นแรงแม่เหล็กระหว่างช่องว่างที่กว้างและแคบ

- มุมของการสเตปแคป

คุณสมบัติอีกประการหนึ่งของสเตเตอร์คือจะต้องมีมุมของการสเตปเล็กที่สุดเท่าที่จะเป็นไปได้ มุมที่แสดงในภาพที่ 3.2 ยังไม่ถือว่าเป็นมุมที่เล็ก แต่ภาพที่ 3.7 (a) แสดงมอเตอร์ 3 เฟสซึ่งมีจำนวนฟันของโรเตอร์ และสเตเตอร์เป็น 2 เท่าของภาพที่ 3.2 ส่วนภาพที่ 3.7 (b) แสดงมอเตอร์ 4 เฟส มุมของการสเตปของทั้ง 2 โครงสร้างนี้เท่ากับ 15° นอกจากนี้ยังมีบางชนิดที่มีมุมของการสเตป 7.5° โดยฟันของโรเตอร์ และสเตเตอร์มี 12 และ 16 ซี่ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.7 (a) แสดงมอเตอร์ 3 เฟส (b) แสดงมอเตอร์ 4 เฟส

ความสัมพันธ์ของมุมของการสเตป θ_S, มุมเฟส m, จำนวนซี่ฟันของโรเตอร์ N_r, จำนวนสเตป S แสดงดังสมการ

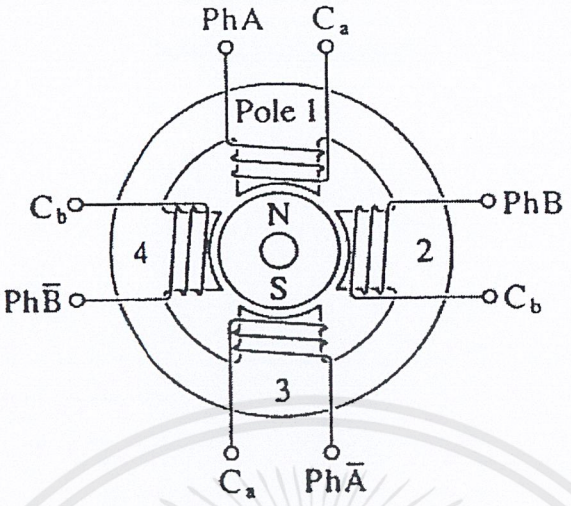
$$S = \frac{360}{\theta_S} = m \cdot N_r \tag{3.1}$$

เพื่อที่จะลดขนาดของมุมสเตปลงต้องเพิ่มจำนวนซี่ฟันของโรเตอร์โดยที่โครงสร้างของแต่ละขั้วของเฟสใดๆ สเตเตอร์จะมีหลายซี่ฟัน แต่ก็ไม่ใช่องค์ประกอบโดยตรงที่จะกำหนดมุมของสเตปมอเตอร์

3.2.2.2 สเตปมอเตอร์แบบแม่เหล็กถาวร

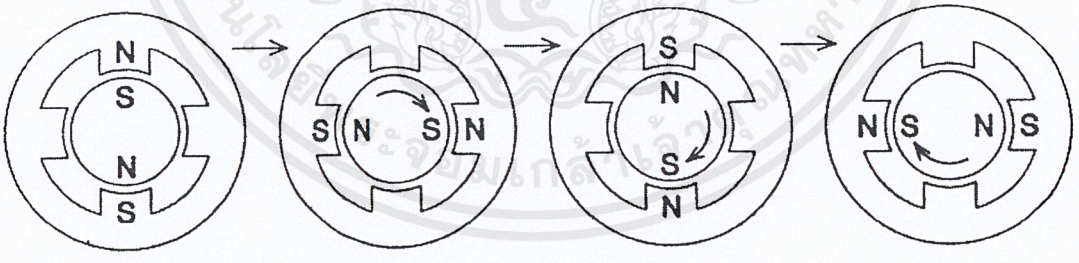
สเตปมอเตอร์ชนิดนี้ใช้แม่เหล็กถาวรเป็นโรเตอร์ และมีซี่ฟันของสเตเตอร์ถูกพันด้วยขดลวดสำหรับสร้างสนามแม่เหล็ก เมื่อต้องการให้สเตปมอเตอร์แบบ แม่เหล็กถาวรมีขนาดของสเตปเล็กลงจะต้องเพิ่มขั้วแม่เหล็กของโรเตอร์และจำนวนซี่ฟันของสเตเตอร์แต่มีขีดจำกัดในการเพิ่มจำนวนขั้วแม่เหล็กของโรเตอร์ เนื่องจากการสร้างแม่เหล็กถาวรสร้างให้มีโครงสร้างโดยมีขั้วแม่เหล็กหลายขั้วทำได้ยาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.8 แสดงโครงสร้างของสเตปมอเตอร์แบบแม่เหล็กถาวร

ตัวอย่างการทำงานของสเตปมอเตอร์แบบแม่เหล็กถาวร สมมติว่าสเตปมอเตอร์แบบแม่เหล็กถาวร ขนาด 4 เฟส มีโรเตอร์เป็นแม่เหล็กถาวรทรงกระบอกและสเตเตอร์มี 4 ซี่ฟันซึ่งรอบ ๆ พันด้วยขดลวด มีรูปแบบพื้นฐานของการทำงาน คือ เมื่อสร้างสัญญาณกระตุ้นตามลำดับเฟส โรเตอร์จะหมุนไปตามทิศทางของการกระตุ้น ดังแสดงในภาพที่ 3.9



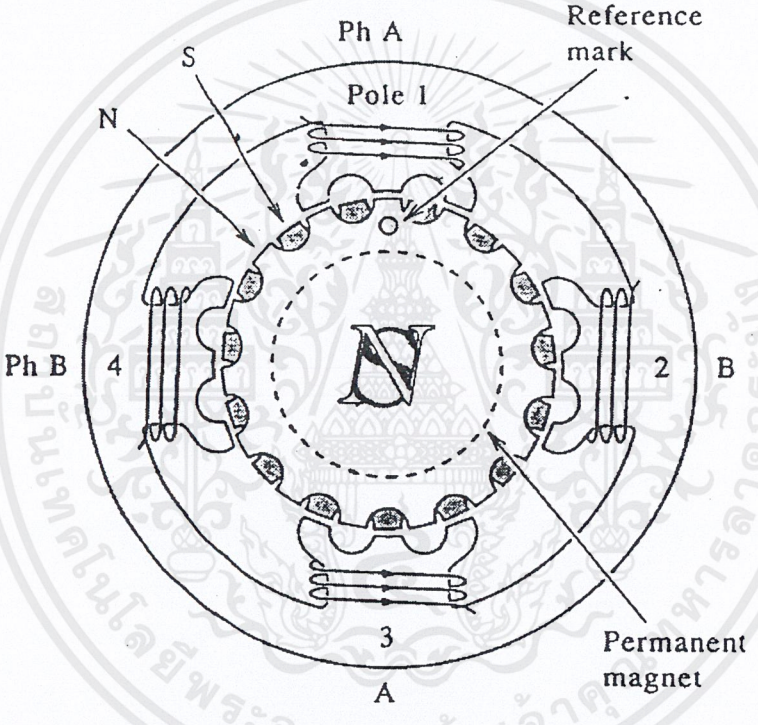
ภาพที่ 3.9 แสดงการทำงานของสเตปมอเตอร์แบบแม่เหล็กถาวรขนาด 4 เฟส

ข้อเสียของสเตปมอเตอร์แบบแม่เหล็กถาวรคือ มีขนาดมุมสเตปใหญ่ทำให้ความละเอียดของสเตปต่อรอบน้อยเนื่องจากโครงสร้างของโรเตอร์เป็นแบบแม่เหล็กถาวรการสร้างแม่เหล็กถาวรให้มีขั้วหลายขั้วทำได้ยากทำให้ไม่สามารถสร้างสเตปขนาดเล็ก ทำให้ค่าทอร์คที่ได้ต่อหน่วยต่อปริมาตรต่ำ ถ้าต้องการปรับประสิทธิภาพในเรื่องของทอร์ค แม่เหล็กถาวรที่ใช้ต้องทำจากสารแม่เหล็กที่มีความเป็นแม่เหล็กสูง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2.3 สเตปมอเตอร์แบบไฮบริดจ์

สเตปมอเตอร์ชนิดนี้มีแกน โรเตอร์เป็นแบบแม่เหล็กถาวร โดยมีการทำงานร่วมกันของมอเตอร์แบบแม่เหล็กถาวร และมอเตอร์แบบปรับค่ารีลัคแตนซ์ได้ ไฮบริดจ์สเตปมอเตอร์นี้มีโครงสร้างของสเตเตอร์คล้ายกับโครงสร้างของสเตปมอเตอร์แบบปรับค่ารีลัคแตนซ์ได้ แต่ต่างกันที่การต่อขดลวดโดยที่แต่ละเฟสของสเตปมอเตอร์แบบปรับค่ารีลัคแตนซ์ได้จะมีขดลวด 2 ขดแต่ละขดมีขั้วต่างกัน แต่ไฮบริดจ์สเตปมอเตอร์ขดลวดทั้งสองจะพันอยู่ที่ ขั้วเดียวกัน เรียกว่า ไบโฟลาร์ (BIFILAR) ซึ่งในการกระตุ้นแต่ละครั้งจะให้ขั้วที่แตกต่างกัน



ภาพที่ 3.10 แสดงโครงสร้างของไฮบริดจ์สเตปมอเตอร์

3.2.2.4 คุณสมบัติที่สำคัญของไฮบริดจ์สเตปมอเตอร์

โครงสร้างของมอเตอร์จะมีแม่เหล็กถาวรอยู่ตรงกลาง ระหว่างเฟสทั้งสองการเหนี่ยวนำสนามแม่เหล็กทำได้โดยใช้สนามแม่เหล็กซึ่งสร้างจากสเตเตอร์ซึ่งเป็นสนามแม่เหล็กแบบเฮทเทอโพลาร์ (HETEROPOLAR FIELD) ดังนั้นทอร์คเกิดจากการทำงานร่วมกันของสนามแม่เหล็ก 2 ชนิด คือ สนามจากแม่เหล็กถาวรและสนามแม่เหล็กเหนี่ยวนำที่เกิดจากการกระตุ้นของขดลวดแต่ละขด โครงสร้างของซี่ฟันของสเตเตอร์จะใหญ่กว่าซี่ฟันของโรเตอร์เล็กน้อยเพื่อเพิ่มความถูกต้องแม่นยำทางตำแหน่งของการเคลื่อนที่

หลักการทำงานของไฮบริดจ์สเตปมอเตอร์ที่แตกต่างจากสเตปมอเตอร์แบบแปรค่ารีลัคแตนซ์ได้คือแรงบิดที่เกิดขึ้นจากสนามแม่เหล็กจะไม่ขึ้นอยู่กับกระแสที่ไหลผ่านขดลวดเพียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อย่างเดียวแต่ขึ้นอยู่กับโครงสร้างของซี่ฟันด้วย ซึ่งซี่ฟันถูกออกแบบเพื่อให้ได้โครงสร้างขนาดเล็กและใช้แม่เหล็กถาวรเป็นแกนกลางเพื่อลดผลของการออสซิลเลททางแมคคานิคส์

ข้อดีของไฮบริดจ์สเตปมอเตอร์คือ มีขนาดสเตปขนาดเล็ก มีความละเอียดสเตปต่อรอบสูง มีค่าทอร์คสูงกว่าสเตปมอเตอร์แบบแปรค่ารีลัคแตนซ์ได้ แต่สเตปมอเตอร์แบบแปรค่ารีลัคแตนซ์ได้มีแรงเฉื่อยทางแมคคานิคส์น้อยกว่าไฮบริดจ์สเตปมอเตอร์

นอกจากสเตปมอเตอร์ทั้ง 3 ชนิดที่กล่าวมาแล้วยังมีสเตปมอเตอร์แบบอื่น ๆ ที่ไม่ได้กล่าวถึงอีกเช่น ลิเนียร์สเตปมอเตอร์ ซึ่งเป็นมอเตอร์ที่ได้รับการออกแบบให้มีการเคลื่อนที่แบบเชิงเส้น อิเล็คโตรดไฮดรอลิคสเตปมอเตอร์ซึ่งเป็นสเตปมอเตอร์กำลังสูงที่ใช้งานอุตสาหกรรม เป็นต้น

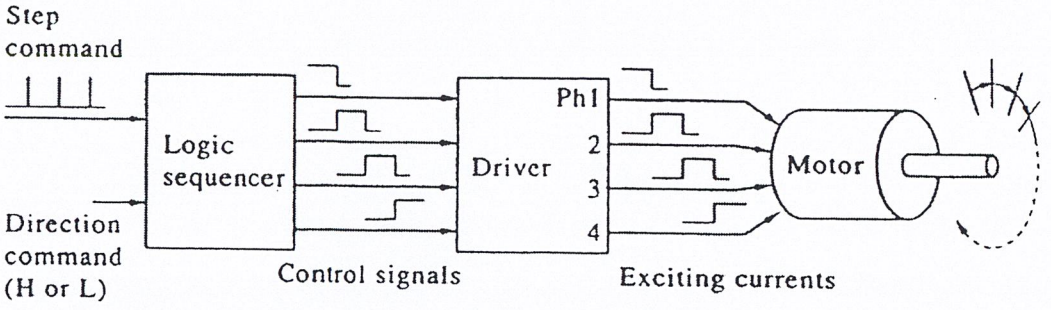
3.3 วงจรขับสเตปมอเตอร์

วงจรขับสเตปมอเตอร์ (STEPPER DRIVE) เป็นวงจรที่แปลงสัญญาณที่มีขนาดเล็กๆ (SMALL SIGNAL) จากระบบควบคุม (CONTROL SYSTEM) ให้เป็นสัญญาณไฟฟ้าที่มีกำลังมากขึ้น เพื่อส่งมอบให้กับมอเตอร์ในการทำให้สเตปมอเตอร์เคลื่อนที่มอเตอร์เป็นอุปกรณ์ที่ให้ผลิตผลเป็นทอร์คและทอร์คนี้ได้ถูกสร้างโดยสนามแม่เหล็ก (MAGNETIC FIELD) ซึ่งเกิดขึ้นเมื่อมีการจ่ายกระแสให้กับขดลวดของสเตปมอเตอร์ทำให้เกิดแรงที่เรียกว่าแรงขับเคลื่อนแม่เหล็ก (MAGNETO MOTIVE FORCE หรือ M.M.F.)

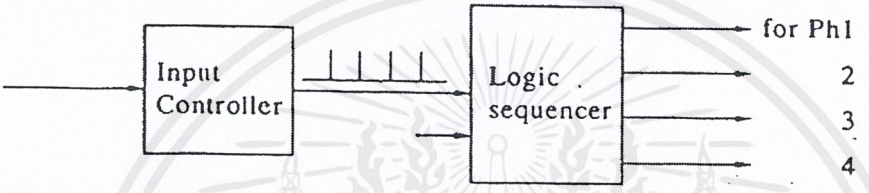
3.3.1 ระบบขับ (DRIVE SYSTEM)

ในภาพที่ 3.11 ได้แสดงถึงพื้นฐานของระบบขับสเตปมอเตอร์ ในภาพที่ 3.11 (a) แสดงถึงส่วนของลอจิกซีเควนเซอร์ (LOGIC SEQUENCER), ไดรเวอร์ (DRIVER) และสเตปมอเตอร์ชนิด 4 เฟส (FOUR-PHASE STEPMOTOR) เมื่อมีสัญญาณสเตปพัลส์เข้ามาทางลอจิกซีเควนเซอร์ จะทำให้เกิดสัญญาณควบคุมออกจากลอจิกซีเควนเซอร์ในลักษณะต่างๆ กันดังภาพ เพื่อทำให้ไดรเวอร์ทำการขยายสัญญาณและส่งไปควบคุมให้สเตปมอเตอร์หมุนไปตามทิศทางที่ต้องการ ทิศทางของการหมุนถูกกำหนดโดยสถานะของสัญญาณกำหนดทิศทาง (DIRECTION SIGNAL) เช่นถ้ามีสถานะที่เข้ามาเป็นสถานะสูง (HIGH LEVEL) ก็ให้หมุนไปตามทิศทางตามเข็มนาฬิกาเป็นต้น ในบางครั้งเมื่อมอเตอร์ต้องการจำนวนสเตปหลายๆ สเตปในการเคลื่อนที่จึงจำเป็นต้องมีอินพุตคอนโทรลเลอร์ (INPUT CONTROLLER) ในการจัดเตรียมสัญญาณที่เข้ามาให้เป็นสัญญาณที่ลอจิกซีเควนเซอร์ สามารถทำให้มอเตอร์เคลื่อนที่ได้ ดังภาพที่ 3.11 (b) แสดงให้เห็นว่าเมื่อมีสัญญาณอินพุตเข้ามา 1 อินพุตหน่วยอินพุตคอนโทรลเลอร์จะกำเนิดสัญญาณพัลส์จำนวน 4 พัลส์ให้กับลอจิกซีเควนเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(a) From logic sequencer to motor



(b) Input controller

ภาพที่ 3.11 แสดงบล็อกไดอะแกรมของระบบขับสเตปมอเตอร์

โดยวงจรที่ใช้ในการขับสเตปมอเตอร์ของปริญญานิพนธ์นี้ แสดงได้ในภาพที่ ก-1 ใน ภาคผนวก ก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
สัญญาณต่าง ๆ บนสล็อตของ IBM/PC

4.1 กล่าวนำ

ภายใน IBM/PCได้มีการออกแบบให้สามารถที่จะเพิ่มเติมวงจรอินเทอร์เฟสเข้าไปในภายหลังได้ โดยผ่านทางสล็อตที่อยู่บนเมนบอร์ด (Main Board) สำหรับสล็อตบนเมนบอร์ดนี้จะมีจำนวน 5 สล็อต (สำหรับใน IBM PC/XT จะมี 8 สล็อต; จะกล่าวถึงในภายหลัง) ซึ่งแต่ละสล็อตจะมีจำนวนขาทั้งสิ้น 62 ขา แบ่งออกเป็น 2 ข้างๆละ 31 ขา ส่วนการเรียกตำแหน่งขาของสล็อตเหล่านี้จะขึ้นอยู่กับว่าขานั้นอยู่ข้างใด (ซ้ายหรือขวา) ของสล็อตโดยขาที่อยู่ทางด้านซ้ายของสล็อตจะเรียกโดยใช้อักษร "B" นำหน้าเลขตำแหน่งของขา เช่น ขา B16 ก็คือขาทางด้านซ้ายของสล็อตขาที่ 16 (นับจากทางด้านท้ายของเครื่อง) ส่วนขาที่อยู่ทางด้านขวาของสล็อตจะเรียกโดยใช้อักษร "A" นำหน้าเลขตำแหน่งของขา เช่น ขาA24 ก็คือขาทางด้านขวาของสล็อตขาที่ 24 (นับจากทางด้านท้ายของเครื่อง)

แต่ละขาของสล็อตเหล่านี้จะเชื่อมต่อกับเส้นสัญญาณต่างๆบนเมนบอร์ด ทำให้การสร้างวงจรอินเทอร์เฟสกับ IBM/PC สามารถทำได้โดยสะดวก ซึ่งเส้นสัญญาณที่เชื่อมต่อกับขาของสล็อตเหล่านี้จะประกอบไปด้วย เส้นสัญญาณของบัสแอดเดรส (Address Bus), บัสข้อมูล (Data Bus), บัสควบคุมสำหรับการเขียน/อ่านข้อมูลจากหน่วยความจำ หรือพอร์ท I/O, เส้นสัญญาณสำหรับการขออินเทอร์รัพท์ของวงจรอินเทอร์เฟส, เส้นสัญญาณสำหรับการขอ DMA, สัญญาณฐานเวลา (Timing- Signal) ต่างๆที่ใช้ในระบบ, เส้นสัญญาณแสดงการรีเฟรชหน่วยความจำ และสัญญาณสำหรับการตรวจสอบความผิดพลาด (I/O CHCK)

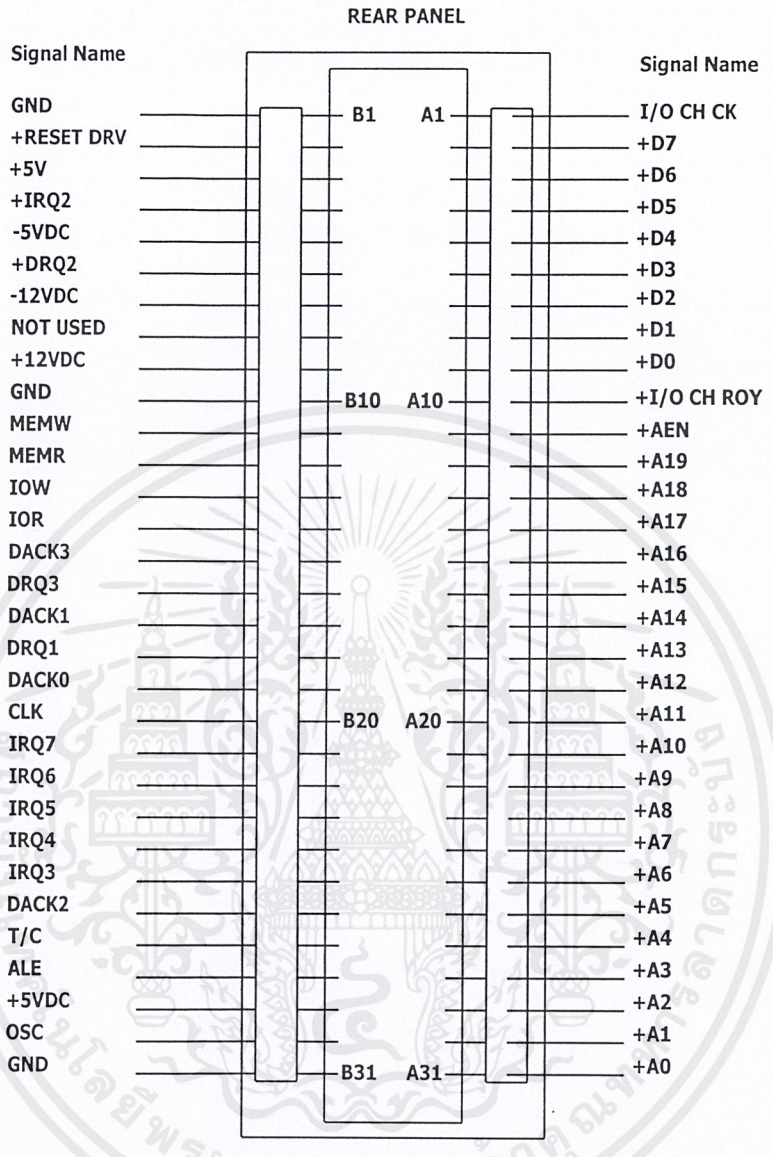
นอกจากเส้นสัญญาณเหล่านี้แล้ว สล็อตบนเมนบอร์ดยังเชื่อมต่อกับแหล่งจ่ายไฟต่างๆที่ใช้ในระบบอีกด้วย คือ +5Vdc , -5Vdc , +12Vdc ,-12vdc

4.2 รายละเอียดเกี่ยวกับสัญญาณต่างๆ

OSC (Oscillator; ขา B30):

ขานี้เป็นเอาท์พุทที่เชื่อมต่อกับสัญญาณคล็อกที่มีค่าความถี่สูงสุดบนเมนบอร์ด คือ 14.31818 MHz ซึ่งมีคาบเวลาประมาณ 70 nanosec. และมี Duty Cycle (ช่วงเวลาใน 1 คาบที่สัญญาณคล็อกมีลอจิกเป็น " 1 " หารด้วยคาบเวลาทั้งหมด) ประมาณ 50% สัญญาณคล็อกอื่นๆของระบบ เช่น คล็อกที่ป้อนให้กับ 8088 หรือ ชิพพอร์ทต่าง ๆ นั้นจะถูกสร้างขึ้นโดยการหารสัญญาณคล็อกนี้ อย่างไรก็ตามสิ่งหนึ่งที่จะต้องคำนึงถึงในการใช้งานสัญญาณ OSC ก็คือ สัญญาณนี้จะไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.1 แสดงสัญญาณต่างๆ บนสล็อตของ IBM/PC

Synchronize กับสัญญาณอื่นๆบนบัสของระบบ ดังนั้นจึงไม่ควรที่จะนำสัญญาณจากขา OSC นี้ไปใช้เป็นสัญญาณคล็อกสำหรับวงจรภายนอกอื่นๆ ที่ทำงานร่วมกับระบบ

CLK (Clock; ขา B20):

ขาสัญญาณนี้เป็นเอาท์พุท ซึ่งต่อกับสัญญาณคล็อกที่ถูกสร้างขึ้นโดยการหารสัญญาณ OSC ด้วย 3 ทำให้ได้ความถี่ประมาณ 4.77 MHz (14.31818 MHz/3) หรือ มีช่วงเวลาใน 1 คาบ (ช่วงเวลาของคล็อก 1 ลูก) เท่ากับ 2.10 nanosec. (1/4.77 MHz) สำหรับค่า Duty Cycle ของสัญญาณนี้จะมี ค่าประมาณ 1/3 คือ ใน 1 คาบจะมีช่วงเวลาที่เป็นลอจิก "1" เท่ากับ 1/3 ของคาบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เวลาทั้งหมด หรือประมาณ 70 nanosec. และช่วงเวลาที่เป็นลอจิก "0" เท่ากับ 2/3 ของคาบเวลาทั้งหมด หรือประมาณ 140 nanosec. สัญญาณนี้เป็นสัญญาณที่ถูกใช้เป็นคล็อกของระบบ

RESET DRV (ขา B2):

ขาสัญญาณนี้เป็นเอาท์พุท ซึ่งจะแอคทีฟ (ลอจิก "1") ในช่วงที่เราเริ่มจ่ายไฟให้กับระบบ และจะยังคงแอคทีฟไปจนกว่าระบบต่างๆภายใน IBM/PC จะพร้อมที่จะทำงานได้จากนั้นสัญญาณนี้ก็จะเปลี่ยนกลับเป็นลอจิก "0" นอกจากนี้ในระหว่างการทำงานของ IBM/PC ถ้าระดับแรงดันของแหล่งจ่ายไฟตก สัญญาณนี้ก็จะถูกทำให้แอคทีฟเช่นกัน โดยทั่วไปแล้วสัญญาณนี้จะถูกนำไปใช้ ในการรีเซ็ทวงจรอินเทอร์เฟสหรืออุปกรณ์ I/O ต่างๆในช่วงที่เริ่มจ่ายไฟให้กับระบบ ซึ่งจะทำเป็นการทำให้วงจรหรืออุปกรณ์เหล่านี้ถูกปรับให้อยู่ในสภาวะที่แน่นอน ก่อนที่จะเริ่มต้นการทำงานในระบบ (สภาวะนี้เป็นสภาวะที่เราทราบ และต้องการให้วงจรทำงานในขณะที่ระบบถูกรีเซ็ท)

A0-A19 (Address Bus; ขา A31-A12):

ขาสัญญาณทั้ง 20 ขานี้เป็นเอาต์พุต ซึ่งใช้สำหรับกำหนดแอดเดรสของหน่วยความจำ หรืออุปกรณ์ I/O ที่ 8088 ต้องการติดต่อด้วย โดยที่สัญญาณ A0 จะมีนัยสำคัญต่ำสุด (Least-Significant Bit) และ A19 จะมีนัยสำคัญสูงสุด (Most Significant Bit) สำหรับค่าแอดเดรสบน บัสแอดเดรส A0-A19 นี้ จะถูกกำหนดโดย 8088 ในระหว่างขบวนการอ่าน/เขียนข้อมูลลงในหน่วยความจำหรืออุปกรณ์ I/O แต่ในช่วงของขบวนการ DMA นั้น DMA-Controller จะเป็นผู้กำหนดค่าแอดเดรสบนบัสแอดเดรสเอง (ในระหว่างนี้ 8088 จะถูกตัดออกจากระบบ)

จะเห็นได้ว่าจำนวนเส้นแอดเดรสนี้มีอยู่จำนวน 20 เส้น ซึ่งสามารถจะอ้างแอดเดรสของหน่วยความจำได้ถึง 1 Mbyte แต่อย่างก็ตามจะมีแอดเดรสบางแอดเดรสที่ถูกใช้งานโดย IBM/PC อยู่ก่อนแล้ว คือแอดเดรสของหน่วยความจำ RAM บนเมนบอร์ดที่ถูกใช้โดยระบบ จำนวน 64 Kbyte (สำหรับ IBM PC/XT จะเป็นจำนวน 256 Kbyte) และแอดเดรสสำหรับหน่วยความจำ ROM อีก 48 Kbyte ซึ่งถูกจัดในช่วงของแอดเดรสบนสุดใน 1 Mbyte คือ 0FC00H จนถึง 0FFFFFH (สำหรับ IBM PC/XT จะเป็น 64 Kbyte)

สำหรับการอ้างแอดเดรสของพอร์ท I/O นั้น จะได้เส้นแอดเดรสเพียง16 เส้น คือA0-A15 ซึ่งจะทำให้อ้างแอดเดรสของพอร์ทได้ 64K พอร์ท โดยผ่านทางชุดคำสั่ง IN และ OUT ส่วนเส้นแอดเดรสที่เหลือคือ A16-A19 นั้นจะไม่ถูกใช้งาน อย่างไรก็ตามภายใน IBM/PC จะใช้เส้น แอดเดรสที่ใช้งานจะต้องอยู่ในช่วง 0200H จนถึง 03FFH เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

D0 - D7 (Data Bus; ขา A9 - A2):

ขาสัญญาณนี้จะเป็นแบบ Bi-Directional ซึ่งต่อกับบัสข้อมูลของระบบ เพื่อทำหน้าที่ในการส่งผ่านข้อมูลระหว่างพอร์ท I/O กับ IBM/PC โดยบิท D0 จะมีนัยสำคัญต่ำสุด และบิท D7จะมีนัยสำคัญสูงสุด

สำหรับในบัสไซเคิลของการเขียนข้อมูลที่สร้างขึ้นโดย 8088 นั้น ข้อมูลจะถูกส่งออกมาบนบัสข้อมูล ก่อนที่สัญญาณ $\overline{\text{MEMR}}$ (ในกรณีที่ต้องการส่งข้อมูลให้พอร์ท) หรือ $\overline{\text{MEMW}}$ (ในกรณีที่ต้องการส่งข้อมูลให้กับหน่วยความจำ) จะเปลี่ยนจากลอจิก " 0 " เป็นลอจิก " 1 " (ขอบขาขึ้น) ซึ่งโดยทั่วไปขอบขาขึ้นของสัญญาณ $\overline{\text{LOW}}$ หรือ $\overline{\text{MEMW}}$ นี้ จะถูกใช้เพื่อสั่งให้พอร์ท I/O หรือหน่วยความจำที่มีแอดเดรสตรงกับค่าแอดเดรสบนบัสแอดเดรสนั้นรับข้อมูลไปเก็บไว้

สำหรับบัสไซเคิลของการอ่านข้อมูลที่สร้างขึ้นโดย 8088 นั้น พอร์ท I/O หรือ หน่วยความจำที่ถูกอ้างถึงจะต้องส่งข้อมูลออกมาบนบัสข้อมูล ก่อนที่สัญญาณ $\overline{\text{IOR}}$ (ในกรณีที่ต้องการอ่านข้อมูลจากพอร์ท) หรือ $\overline{\text{MEMW}}$ (ในกรณีที่ต้องการอ่านข้อมูลจากหน่วยความจำ) จะเปลี่ยนจากลอจิก " 0 " เป็นลอจิก " 1 " (ขอบขาขึ้น)

ALE (Address Latch Enable: ขา B28):

ขาสัญญาณนี้เป็นขาสัญญาณเอาท์พุทที่ 8088 Bus Controller สร้างขึ้นเพื่อใช้สำหรับแสดงการเริ่มต้นของบัสไซเคิล และแสดงให้อุปกรณ์ภายนอกทราบว่าแอดเดรสที่ 8088 ต้องการติดต่อด้วยนั้นถูกส่งออกมาบนบัสแอดเดรสแล้ว โดยที่สัญญาณ ALE นี้จะเปลี่ยนจากลอจิก " 1 " เป็น " 0 " เมื่อค่าแอดเดรสที่ถูกต้องถูกส่งออกมาบนบัสข้อมูลเรียบร้อยแล้ว ดังนั้นขอบขาลงของสัญญาณ ALE นี้จะถูกใช้ในการแลทช์ค่าแอดเดรสจากบัสแอดเดรส / ข้อมูล (Address / Data Bus: AD0 – AD7) ของ 8088 ทำให้สามารถแยกค่าแอดเดรส (A0 – A19) และข้อมูล (A0 – A7) ออกจากกัน อย่างไรก็ตามสัญญาณ ALE จะแอคทีฟเฉพาะในบัสไซเคิลที่สร้างขึ้นโดย 8088 เท่านั้น โดยจะไม่แอคทีฟระหว่างขบวนการ DMA

$\overline{\text{I/O CHCK}}$ (I / O Channel Check: ขา A1) :

ขาสัญญาณนี้เป็นอินพุทที่ใช้แสดงความผิดพลาดเกี่ยวกับพาริตี้ ที่เกิดขึ้นในการทำงาน ของวงจรอินเทอร์เฟสหรืออุปกรณ์ I/O เมื่อขาสัญญาณนี้ได้รับลอจิก " 0 " จะทำให้ 8088 ถูกอินเทอร์รัพท์แบบ Non – Maskable (NMI) อย่างไรก็ตามเราสามารถที่จะกำหนดให้วงจรของ IBM/PC ทำการขออินเทอร์รัพท์ (เมื่อได้รับสัญญาณ $\overline{\text{I/O CHCK}}$) หรือไม่ก็ได้ โดยการกำหนดลอจิกของบิทข้อมูลของพอร์ทที่ควบคุมการขออินเทอร์รัพท์แบบ NMI คือบิท D7 ของพอร์ท 00A0H ในกรณีที่บิท D7 ของพอร์ท 00A0H ถูกเซ็ทเป็น "1" ก็จะทำให้วงจรภายนอกของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อินเทอร์รัพท์แบบ NMI ได้ (Enable) แต่บิท D7 ของพอร์ท 00A0H ถูกเซ็ทเป็น "0" ก็จะเป็นการดิสเอเบิล (Disable) การขออินเทอร์รัพท์แบบ NMI ดังนี้

nable : ใช้คำสั่ง OUT ส่งข้อมูล 80H ไปยังพอร์ท 00A0H

isable : ใช้คำสั่ง OUT ส่งข้อมูล 00H ไปยังพอร์ท 00A0H

และเนื่องจากยังมีอุปกรณ์อื่นที่สามารถขออินเทอร์รัพท์แบบ NMI ได้อีก ดังนั้นซอฟท์แวร์ที่ใช้งานจะต้องสามารถตรวจสอบว่าการขออินเทอร์รัพท์นั้นเกิดขึ้นจากแหล่งใดได้ด้วย

I/O CHRDY (I/O Channel Ready: ขา A10):

ขาสัญญาณนี้เป็นอินพุทที่ใช้เพิ่มช่วงเวลาในบัสไซเคิลในกรณีที่อุปกรณ์ I/O หรือหน่วยความจำที่เกี่ยวข้องกับขบวนการในบัสไซเคิลที่เกิดขึ้นนั้น ไม่สามารถทำงานได้ทันตามช่วงเวลาปกติของบัสไซเคิลนั้น ๆ ได้ (ช่วงเวลาของบัสไซเคิลที่เกี่ยวกับหน่วยความจำใช้ช่วงเวลาเท่ากับช่วงเวลาของคล็อก 4 ลูก หรือ 840 nanosec. ในขณะที่บัสไซเคิลที่เกี่ยวกับ I/O จะใช้ช่วงเวลาเท่ากับช่วงเวลาของคล็อก 5 ลูก หรือ 1.05 μsec)

เมื่ออุปกรณ์ I/O หรือหน่วยความจำต้องก่ารที่จะเพิ่มช่วงเวลาในบัสไซเคิลให้นานขึ้นอีกนั้น จะสามารถทำได้โดยการป้อนลอจิก "0" ให้กับขา I/O CHRDY ในช่วงเวลาที่ I/O หรือ หน่วยความจำที่ถูกกำหนดนั้น ได้รับสัญญาณจาการดีโค้ดแอดเดรส และ สัญญาณ $\overline{\text{MEMR}}$, $\overline{\text{MEMW}}$, $\overline{\text{IOR}}$ หรือ $\overline{\text{IOW}}$ แอคทีฟ

IRQ2 – IRQ7 (Interrupt Request 2 through 7: ขา B4 และ B25 – B21)

ขาสัญญาณทั้ง 6 นี้เป็นขาอินพุทที่ใช้สำหรับขออินเทอร์รัพท์จาก 8088 โดยสัญญาณเหล่านี้จะต่อเข้ากับ 8259A บนเมนบอร์ดโดยตรง โปรแกรมในส่วน BIOS ของ IBM/PC จะทำการโปรแกรม 8259A ให้ IRQ2 มีลำดับความสำคัญสูงสุด (Highest Priority) และ IRQ7 มีลำดับความสำคัญต่ำสุด ในกรณีที่มีการขออินเทอร์รัพท์เกิดขึ้นคือ ระดับลอจิกที่ขา IRQ ขาใดขาหนึ่งถูกเปลี่ยนจากลอจิก "0" เป็นลอจิก "1" (ขอบขาขึ้น) 8259A ก็จะทำการส่งสัญญาณ INT ให้กับ 8088 เพื่อทำการขออินเทอร์รัพท์

สิ่งสำคัญในการขออินเทอร์รัพท์โดยผ่านทาง IRQ2 – IRQ7 นี้ ก็คืออุปกรณ์ที่ทำการขออินเทอร์รัพท์โดยผ่านทาง IRQ ขาใดก็จะต้องรักษาระดับสัญญาณที่ขา IRQ นั้น ให้แอคทีฟ (ลอจิก "1") อยู่จนกว่าจะได้รับสัญญาณ INTA (Interrupt Acknowledge) จาก 8088 เสียก่อน ถ้าไม่เช่นนั้นการขออินเทอร์รัพท์จถูกยกเลิก และอินทอร์รัพท์ Level 7 (IRQ7) ก็จะถูกสร้างขึ้นโดยอัตโนมัติ ไม่ว่าการขออินเทอร์รัพท์ที่ถูกยกเลิกนั้นจะเป็นการขออินเทอร์รัพท์ใน Level 1 หรือขาใด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้นไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่อย่างไรก็ตามสัญญาณ INT นี้จะไม่ถูกต่อออกมาที่ขาของสล็อตด้วย ดังนั้นโปรแกรมที่ทำการตอบสนองต่อการขออินเทอร์รัพท์ (Interrupt Service Routine) ต้องทำการรีเซ็ทสัญญาณ IRQ เอง โดยใช้คำสั่ง OUT ไปยังพอร์ท I/O ที่เกี่ยวข้อง

$\overline{\text{IOR}}$ (I/O Read: ขา B14):

ขาสัญญาณนี้เป็นเอาท์พุทแอคทีฟที่ลอจิก "0" ที่สร้างขึ้นโดย 8088 Bus Controller เพื่อใช้ในการแสดงว่าบัสไซเคิลที่เกิดขึ้นนี้ เป็นบัสไซเคิลของการอ่านข้อมูลจากพอร์ท I/O เพื่อให้พอร์ท I/O ที่มีแอดเรดสตรงกับอดเดรสนั้นส่งข้อมูลออกมาบนบัสข้อมูล โดยข้อมูลจะต้องถูกส่งออกมาบนบัสข้อมูลก่อนขอบขาขึ้นของสัญญาณ $\overline{\text{IOR}}$ ประมาณ 30 nanosec. เพื่อให้มั่นใจได้ว่า 8088 สามารถรับข้อมูลได้ถูกต้อง

สำหรับขบวนการ DMA 8237A – 5 DMA จะทำการสร้างสัญญาณ $\overline{\text{IOR}}$ เอง โดยที่ค่าแอดเดรสที่อยู่บนบัสแอดเดรสจะเป็นค่าแอดเดรสของหน่วยความจำ (แทนที่จะเป็นแอดเดรสของพอร์ท I/O) ที่พอร์ท I/O ที่ขอ DMA ต้องการจะนำข้อมูลไปเก็บ การที่พอร์ทใดจะส่งข้อมูลออกมาบนบัสข้อมูลนั้น จะอาศัยสัญญาณ DACK จาก DMA Controller เป็นตัวกำหนด เช่นกรณีที่สัญญาณ DACK 1 แอคทีฟก็จะแสดงว่าพอร์ท I/O ที่จะต้องส่งข้อมูลออกมาบนบัสข้อมูลก็คือพอร์ท I/O ที่ขอ DMA ต้องการจะอ่านข้อมูล

$\overline{\text{MEMW}}$ (Memory Read: ขา B12):

ขานี้เป็นเอาท์พุทจาก 8088 ซึ่งสัญญาณนี้จะแอคทีฟ (ลอจิก "0") ในระหว่างบัสไซเคิลในการเขียนข้อมูลลงในหน่วยความจำของ 8088 สัญญาณ $\overline{\text{MEMW}}$ นี้จะถูกส่งออกมา เพื่อให้หน่วยความจำที่แอดเดรสตรงกับค่าแอดเดรสบนบัสแอดเดรสนั้น ทำการรับข้อมูลที่อยู่บนบัสข้อมูลไปเก็บไว้ โดยทั่วไปหน่วยความจำจะรับข้อมูลในช่วงขอบขาขึ้นของสัญญาณ $\overline{\text{MEMW}}$

สำหรับในระหว่างขบวนการ DMA นั้น 8237A – 5 DMA – Controller จะทำการควบคุมบัสต่าง ๆ ของระบบแทน 8088 และสัญญาณ $\overline{\text{MEMW}}$ จะถูกใช้ในบัสไซเคิลของการเขียนข้อมูลลงในหน่วยความจำ (ข้อมูลถูกส่งจากอุปกรณ์ I/O ไปให้กับหน่วยความจำ)

$\overline{\text{MEMR}}$ (Memory Read: ขา B12):

ขานี้เป็นเอาท์พุทจาก 8288 ซึ่งสัญญาณนี้จะแอคทีฟ (ลอจิก "0") ในระหว่างบัสไซเคิลของการอ่านข้อมูลจากหน่วยความจำของ 8088 เพื่อให้หน่วยความจำที่มีแอดเดรสตรงกับค่าแอดเดรสบนบัสแอดเดรสนั้น ทำการส่งข้อมูลออกมาบนบัสข้อมูลนั้น โดยหน่วยความจำนั้นจะต้องส่งข้อมูลออกมาในช่วงเวลา 30 nanosec. ก่อนที่สัญญาณ $\overline{\text{MEMR}}$ จะกลับเป็นลอจิก "1" ทั้งนี้ก็เพื่อให้ 8088 ได้รับข้อมูลที่ถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับในระหว่างขบวนการ DMA นั้น DMA – Controller จะควบคุมบัสต่าง ๆ ของระบบแทน 8088 และสัญญาณ $\overline{\text{MEMR}}$ จะถูกใช้ในบัสไซเคิลของการอ่านข้อมูลจากหน่วยความจำ (ข้อมูลถูกส่งจากหน่วยความจำไปให้กับอุปกรณ์ I/O)

DRQ1 – DRQ3 (DMA Request 1 – 3: ขา B18, B6 และ ขา B16):

ขาสัญญาณทั้งสามนี้เป็นสัญญาณอินพุทแอคทีฟที่ลอจิก "1" ซึ่งอุปกรณ์ภายนอกสามารถใช้ในการขอ DMA จากระบบ โดยการป้อนระดับสัญญาณลอจิก "1" ให้กับขา DRQ ขาใดขาหนึ่ง (ขาDRQ ทั้งสามนี้จะต่อเข้ากับ DRQ1 – DRQ3 ของ 8237A – 5)

เมื่อ 8237A – 5 ได้รับสัญญาณนี้แล้วก็จะตรวจสอบว่ามีการขอ DMA ในแซลแนลที่มีลำดับความสำคัญ (Priority) สูงกว่าหรือไม่ ถ้าไม่ก็จะทำการขอ DMA จาก 8088 และตอบรับการขอ DMA จากอุปกรณ์ภายนอก (สัญญาณ $\overline{\text{DACK}}$ ของแชนเนลที่ขอ DMA จะแอคทีฟ) แต่ถ้ามี 8237A – 5 ก็จะทำการขอ DMA ให้กับแชนแนลที่มีลำดับความสำคัญต่ำกว่า ภายใน ROM BIOS ของ IBM/PC จะโปรแกรม 8237A – 5 ให้ DRQ 1 มีลำดับความสำคัญสูงสุดและ DRQ3 มีลำดับความสำคัญต่ำสุด ดังนั้นถ้ามีการขอ DMA ของอุปกรณ์ภายนอกผ่านทางแชนแนลที่ (DRQ1) และแชนแนลที่ 2 (DRQ2) 8237A – 5 ก็จะทำการขอ DMA ให้กับแชนแนลที่ 1 ก่อน จากนั้นเมื่อเสร็จจากขบวนการ DMA ของแชนแนลที่ 1 แล้ว จึงจะทำการขอ DMA ให้กับแชนแนลที่ 2 .

อย่างไรก็ตาม 8237A – 5 ยังมีแชนแนลสำหรับการขอ DMA ให้กับแชนแนลที่ 2 แชนแนลที่ 0 (DRQ 0) ซึ่งในความเป็นจริงแล้วแชนแนลนี้จะมีลำดับความสำคัญที่สูงกว่าแชนแนลที่ 1 แต่จะไม่ถูกต่อออกมายังขาของสล็อต เนื่องจาก IBM/PC จะใช้แชนแนลที่ 0 นี้ในการรีเฟรชหน่วยความจำที่เป็น Dynamic RAM

ในการขอ DMA เน้นสัญญาณ DRQ นี้ จะต้องแอคทีฟอยู่ในช่วงเวลาหนึ่งเท่านั้นถ้าสัญญาณนี้แอคทีฟอยู่นานเกินไป จะทำให้เกิดขบวนการ DMA ขึ้นมากกว่า 1 ขบวนการได้ สำหรับวงจรที่ขอ DMA โดยทั่วไปแล้วจะใช้สัญญาณตอบรับการขอ DMA หรือสัญญาณ $\overline{\text{DACK}}$ ของแชนแนลที่ขอ DMA นั้น ในการรีเซ็ทสัญญาณ DRQ เช่นอุปกรณ์ภายนอกที่ขอ DMA ผ่านทางแชนแนลที่ 1 (DRQ 1) ก็จะคอยตรวจสอบการตอบรับการขอ DMA จากสัญญาณ $\overline{\text{DACK}}$ ของแชนแนลที่ 1 ($\overline{\text{DACK1}}$) เมื่อได้รับสัญญาณจาก $\overline{\text{DACK1}}$ แล้ว ก็จะรีเซ็ทสัญญาณ DRQ1 (เปลี่ยนจากลอจิก "1" เป็น "0")

$\overline{\text{DACK}}$ 0 – $\overline{\text{DACK}}$ 3 (DMA Acknowledge 0 – 3: ขา B19, B17, B26 และ B15)

สัญญาณทั้ง 4 นี้เป็นเอาท์พุทแอคทีฟที่ลอจิก "0" ซึ่ง 8237A- 5 สร้างขึ้นเพื่อแสดงให้วงจรภายนอกที่ขอ DMA ทราบว่าการขอ DMA นั้นได้รับการตอบสนองแล้ว และ 8237A-5 จะเข้าสู่ขบวนการ DMA เพื่อให้การส่งข้อมูลระหว่างอุปกรณ์ I/O ที่ขอ DMA กับหน่วยความจำเกิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขึ้นได้โดยตรง (คือไม่ต้องผ่าน 8088) โดยสัญญาณ DACK นี้จะแอคทีฟในแชนแนลใดขึ้นอยู่กับว่าขบวนการ DMA ที่เกิดขึ้นนั้น เป็นการตอบสนองต่อการขอ DMA ในแชนแนลใด เช่นถ้าขบวนการ DMA ที่จะเกิดขึ้นนั้นเป็นการตอบสนองต่อการขอ DMA ในแชนแนลที่ 2 (DRQ 2) สัญญาณ $\overline{\text{DACK2}}$ ก็จะแอคทีฟ เป็นต้น

ดังที่กล่าวแล้วว่าสัญญาณ DRQ0 นั้น จะไม่ถูกต่อออกมายังขาของสล็อต ดังนั้นวงจรอินเทอร์เฟสจึงไม่สามารถจะขอ DMA ผ่านทางแชนแนล 0 ได้ แต่สัญญาณ $\overline{\text{DACK0}}$ จะถูกต่อออกมายังสล็อตด้วย (ขา B19) ทั้งนี้ก็เพื่อที่จะแสดงให้วงจรอินเทอร์เฟสต่าง ๆ ทราบว่าขบวนการ DMA ที่กำลังเกิดขึ้นในช่วงเวลาที่ $\overline{\text{DACK0}}$ แอคทีฟนั้น เป็นขบวนการที่ใช้สำหรับการรีเฟรชหน่วยความจำที่เป็น Dynamic RAM ซึ่งวงจรอินเทอร์เฟสที่ใช้หน่วยความจำประเภทนี้สามารถจะนำไปใช้ในการรีเฟรช Dynamic RAM ที่อยู่ในวงจรได้

โดยที่การรีเฟรชหน่วยความจำนั้นจะเกิดขึ้นในทุก ๆ 15.12 μsec. หรือ ทุก ๆ 72 คล็อก ดังนั้นสัญญาณ $\overline{\text{DACK0}}$ นี้ก็จะแอคทีฟในทุก ๆ 15.12 μsec. ด้วย

AEN (Address enables: ขา A11):

สัญญาณนี้เป็นเอาท์พุทที่มช้แสดงว่าบัสไซเคิลที่เกิดขึ้นในช่วงเวลาที่สัญญาณ AEN แอคทีฟ (ลอจิก "1") นั้น เป็นบัสไซเคิลของขบวนการ DMA

สำหรับเมนบอร์ดของ IBM/PC นั้น จะใสัญญาณนี้ในการดิสเอเบิล (Disable) 8288 Bus Controller และจะใช้ดิสเอเบิลพอร์ท I/O ต่าง ๆ ที่ไม่เกี่ยวข้องกับขบวนการ DMA ที่เกิดขึ้นนี้ ที่จำเป็นต้องทำเช่นนี้ก็เพราะในระหว่างขบวนการ DMA นั้น 8237A-5 จะส่งแอดเดรสของหน่วยความจำออกมาบนบัสแอดเดรส และจะทำให้สัญญาณ $\overline{\text{IOR}}$ หรือ $\overline{\text{IOW}}$ แอคทีฟด้วย ดังนั้นถ้าไม่ทำการดิสเอเบิลพอร์ท I/O ที่ไม่เกี่ยวข้องไว้ ก็อาจทำให้พอร์ท I/O ที่มีแอดเดรสตรงกับค่า แอดเดรสบนบัสแอดเดรส (ซึ่งเป็นแอดเดรสของหน่วยความจำ) นั้น ทำการอ่านหรือส่งข้อมูลออกมาบนบัสข้อมูลทำให้เกิดความผิดพลาดขึ้นได้

T/C (Terminal Count: ขา B27):

สัญญาณนี้ถูกสร้างขึ้นจากการนำเอาสัญญาณเอาท์พุทที่ขา EOP ของ 8237A-5 มากลับ ลอจิก (โดยใช้เกท Inverter) ทำให้สัญญาณ T/C นี้แอคทีฟที่ลอจิก "1"

สำหรับสัญญาณนี้จะแอคทีฟเมื่อจำนวนไบท์ในการส่งผ่านข้อมูลของขบวนการ DMA ในแชนแนลใดแชนแนลหนึ่ง ครบตามจำนวนที่กำหนดไว้ โดยทั่วไปแล้วสัญญาณที่จะถูกใช้ในการสิ้นสุดขบวนการ DMA ที่มีการส่งข้อมูลข้อมูลเป็นบล็อก เนื่องจากสัญญาณนี้จะแอคทีฟโดยไม่แสดงว่าเป็นสัญญาณของแชนแนลใด ดังนั้นจึงต้องทำการนำสัญญาณ T/C นี้ผ่านเกท Inverter แล้วนำไป OR กับสัญญาณ DACK เพื่อให้สามารถทราบได้ว่า สัญญาณ T/C ที่เกิดขึ้นนั้นเป็นสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของแชนแนลใด สำหรับแชนแนลที่ 0 นั้นสัญญาณ T/C จะแอคทีฟในช่วงวเวลาที่คงที่คือ ทุก ๆ 990.804 millisec. ซึ่งก็คือช่วงเวลาที่ใช้ในการรีเฟรชหน่วยความจำขนาด 64 Kbyte นั่นเอง

+ 5 Vdc (ขา B3 และ B29):

ขาทั้งสองนี้ต่อกับแหล่งจ่ายไฟ DC+ 5 V ของระบบโดยจะมีค่าความเที่ยงตรง (Regulated) ± 5 % คือ อยู่ในช่วง + 4.75 ถึง +5.25 Vdc

+12 Vdc (ขา B9):

ขานี้จะต่อกับแหล่งจ่ายไฟ DC +12 V ของระบบ โดยจะมีค่าความเที่ยงตรง (Regulate) ± 5 % คือ อยู่ในช่วง +11.4 ถึง + 12.6 Vdc

- 5 Vdc (ขา B5):

ขานี้จะต่อกับแหล่งจ่ายไฟ DC – 5 V ของระบบ โดยจะมีค่าความเที่ยงตรง (Regulate) ± 10 % คือ อยู่ในช่วง – 5.5 ถึง – 4.5 Vdc

- 12 Vdc (ขา B7):

ขานี้จะต่อกับแหล่งจ่ายไฟ DC – 5 V ของระบบ โดยจะมีค่าความเที่ยงตรง (Regulate) ± 10 % คือ อยู่ในช่วง – 13.2 ถึง –10.8 Vdc

GND (ขา B1, B10 และ B31):

ขาทั้งสามนี้จะต่อเข้ากับกราวด์ (Ground)

4.3 การจัดสัญญาณบนสล็อตของ IBM PC / XT

สำหรับใน IBM PC / XT นั้นจะมีสล็อตสำหรับเชื่อมต่อวงจรภายนอกได้มากขึ้น คือ ใน IBM PC / XT จะทำการเพิ่มจำนวนสล็อตบนเมนบอร์ดขึ้นเป็น 8 สล็อต จากเดิมที่มีอยู่เพียง 5 สล็อตบน IBM PC โดยการจัดสัญญาณต่าง ๆ ในทั้ง 8 สล็อตจะยังคงเหมือนกับใน IBM PC เพียงแต่สัญญาณต่าง ๆ ที่จะถูกส่งออกมายังขาของสล็อตที่ 8 นั้น จะถูกต่อผ่านวงจรขับกระแส (Buffer) ก่อน และในสล็อตที่ 8 นี้ขา B8 จะถูกใช้งานด้วย โดยจะใช้เป็นขา $\overline{\text{CRAD SLCTD}}$ (หรือ Card Selected) ซึ่งขาสัญญาณนี้จะเป็นขาสัญญาณอินพุทจากวงจรภายนอกที่เสียบอยู่บนสล็อตที่ 8 เพื่อให้วงจรบนเมนบอร์ททราบว่าคาร์ดที่อยู่บนสล็อตนี้ถูกเลือกใช้งานอยู่ ซึ่งจะทำให้ Driver บนเมนบอร์ดทำการอ่านหรือส่งข้อมูลไปยังสล็อตที่ 8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 บัสไซเคิลของระบบ

บัสไซเคิล คือ ขบวนการของสัญญาณที่เกิดขึ้นในช่วงของการส่งผ่านข้อมูลระหว่างหน่วยความจำ , อุปกรณ์อินพุท / เอาท์พุท (Input / Output I/O) และไมโครโปรเซลเซอร์ 8088 ดังนั้นในการที่จะออกแบบวงจรอินเทอร์เฟส (Interface) สำหรับเครื่อง IBM/PC ซึ่งใช้ไมโครโปรเซสเซอร์เบอร์ 8088 นั้น เราจะต้องเข้าใจถึงบัสไซเคิลของการส่งผ่านข้อมูลในลักษณะต่าง ๆ ของระบบเสียก่อน

สำหรับบัสไซเคิลของระบบนั้น เราสามารถจะแบ่งออกตามลักษณะของอุปกรณ์ที่สร้างบัสไซเคิลนั้นได้เป็น 2 กลุ่ม คือ

1. บัสไซเคิลที่สร้างโดย 8088 (8088 Driver Bus Cycle) : สำหรับบัสไซเคิลในกลุ่มนี้จะเป็นบัสไซเคิลที่ 8088 สร้างขึ้นในการส่งข้อมูลระหว่าง 8088 กับ I/O หรือระหว่าง 8088 กับหน่วยความจำ โดยในบัสไซเคิลเหล่านี้ 8088 จะส่งแอดเดรสของหน่วยความจำหรือพอร์ท ที่ต้องการจะติดต่อด้วย ออกมาบนแอดเดรสบัส, ส่งหรือรับข้อมูลจากบัสข้อมูล และส่งสัญญาณควบคุมที่จำเป็นออกมาบนบัสควบคุม ซึ่งบัสไซเคิลในกลุ่มนี้แบ่งออกเป็น 5 ชนิด คือ บัสไซเคิลในการอ่านข้อมูลจากหน่วยความจำ (Memory – Read Bus Cycle), บัสไซเคิลในการเขียนข้อมูลลงหน่วยความจำ (Memory – Write Bus Cycle), บัสไซเคิลในการรับข้อมูลจากพอร์ท I/O (I/O Port – Read Bus Cycle), บัสไซเคิลในการเขียนข้อมูลลงบนพอร์ท I/O (I/O Port – Write Bus Cycle) และบัสไซเคิลในการตอบรับการขออินเทอร์รัพท์ (Interrupt Acknowledge Bus cycle) สำหรับบัสไซเคิลในการตอบรับการขออินเทอร์รัพท์นี้จะเป็นบัสไซเคิลที่เกิดขึ้นเฉพาะบน Main Board ของ IBM/PC เท่านั้น โดยจะไม่ปรากฏบนสล็อต (Slot) ทั้งห้าของ IBM/PC (8 สล็อตในกรณีของ IBM/PC)

2. บัสไซเคิลที่สร้างขึ้นโดย DMA Controller (DMA – Driver Bus Cycle) : สำหรับบัสไซเคิลในกลุ่มนี้จะเป็นบัสไซเคิลที่ DMA Controller (ในที่นี้คือ 8237A – 5) สร้างขึ้นในขบวนการ DMA โดยในระหว่างขบวนการ DMA นี้ 8088 จะทำให้บัสแอดเดรส, บัสข้อมูล และบัส ควบคุมบางเส้นของ 8088 มีสถานะเป็น High Impedance ซึ่งจะทำให้ 8237 – 5 DMA Controller สามารถควบคุมบัสต่าง ๆ เหล่านี้ได้ จากนั้น 8237 – 5 ก็จะสร้างบัสไซเคิลขึ้นเพื่อให้อุปกรณ์ I/O สามารถรับข้อมูลให้กับหน่วยความจำได้โดยตรง (โดยไม่ต้องผ่าน 8088 หรือ 8237 – 5 ก่อน ซึ่งจะทำให้ประหยัดเวลาและลดความยุ่งยากในการส่งข้อมูลลงได้) โดยในบัสไซเคิลเหล่านี้ 8237-5จะทำหน้าที่ในการส่งแอดเดรสและสัญญาณที่จำเป็นออกมาบนบัสแอดเดรสและบัสควบคุมแทน 8088 สำหรับบัสไซเคิลในกลุ่มนี้จะแบ่งออกเป็น 2 ชนิด คือ บัสไซเคิลในการอ่านข้อมูลจากอุปกรณ์ I/O เพื่อส่งให้กับหน่วยความจำ (DMA Memory – Write Bus Cycle) และบัสไซเคิลในการอ่านข้อมูลจากหน่วยความจำเพื่อส่งข้อมูลให้กับอุปกรณ์ I/O (DMA Memory – Read Bus Cycle)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 บัสไซเคิลต่าง ๆ ในระบบ

การทำงานในแต่ละบัสไซเคิลของ 8088 นั้น จะใช้ช่วงเวลาเท่ากับช่วงเวลาของคล็อกที่ป้อน ให้กับ 8088 จำนวน 4 ลูก สำหรับ IBM/PC คล็อกที่ป้อนให้กับ 8088 จะมีความถี่ประมาณ 4.77 MHz หรือ มีคาบของคล็อก 1 ลูกประมาณ 210 nanosec. ดังนั้นใน 1 บัสไซเคิล จะใช้เวลาประมาณ 4 x 210 nanosec. หรือ 840 nanosec. อย่างไรก็ตามในกรณีที่อุปกรณ์ที่ทำงานร่วมกับ 8088 เช่น หน่วยความจำหรือพอร์ท I/O นั้น มีความเร็วในการทำงานที่ต่ำไม่สามารถที่จะตอบสนองการทำงานในบัสไซเคิลของ 8088 ได้ทัน อุปกรณ์นั้นก็สามารถที่จะเพิ่มช่วงเวลาในบัสไซเคิลได้ โดยการป้อนลอจิก "0" ให้กับขา READY ของ 8088

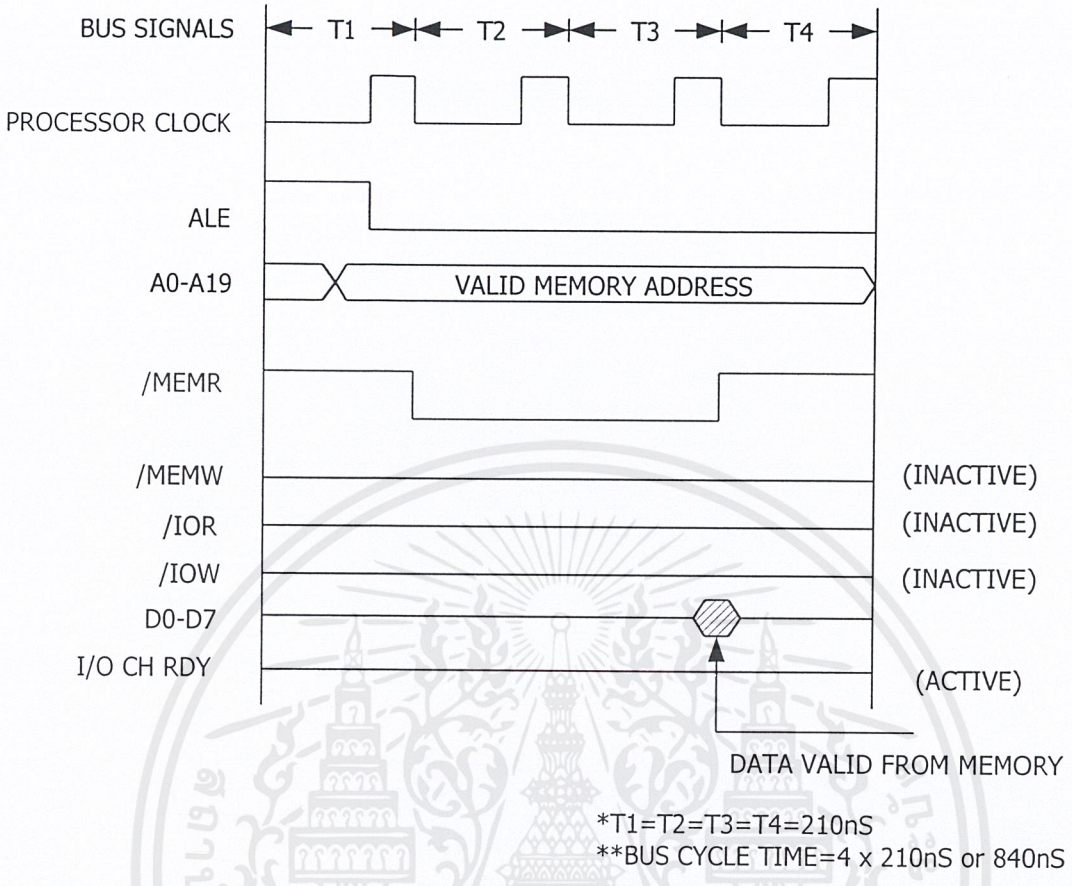
สำหรับอุปกรณ์ที่เป็นหน่วยความจำ เช่น ROM หรือ RAM นั้น โดยทั่วไปจะมีความเร็วในการทำงานสูงพอที่จะตอบสนองต่อการทำงานในบัสไซเคิลของ 8088 ได้ ดังนั้นใน IBM/PC จึงไม่มีการเพิ่มช่วงเวลาของบัสไซเคิลในการอ่าน / เขียนข้อมูลลงในหน่วยความจำ คือ ใน 1 บัสไซเคิลจะยังคงใช้ช่วงเวลาเท่ากับช่วงเวลาคล็อก 4 ลูก (840 nanosec.) ใน 1 บัสไซเคิลเป็นคล็อก 5 ลูก (1.05 μsec.) ใน 1 บัสไซเคิล สำหรับคล็อกที่เพิ่มเข้ามานี้จะเรียกว่า " TW " และ สภาวะที่ 8088 หยุดรอ เพื่อให้อุปกรณ์ภายนอกรับหรือส่งข้อมูลได้ทันนั้นก็คือ " Wait State "

ในส่วนต่อไปจะกล่าวถึงบัสไซเคิลต่าง ๆ ที่ถูกสร้างขึ้นโดย 8088 ซึ่งจะประกอบด้วย บัสไซเคิลในการอ่านข้อมูลจากหน่วยความจำ , บัสไซเคิลในการเขียนข้อมูลลงบนหน่วยความจำ , บัสไซเคิลในการรับข้อมูลจากพอร์ท I/O, บัสไซเคิลในการส่งข้อมูลออกทางพอร์ท I/O และบัสไซเคิลในการตอบรับการขออินเทอร์รัพท์ โดยบัสไซเคิลที่จะกล่าวถึงนี้เป็นบัสไซเคิลที่ปรากฏบนสล็อตทั้ง 5 ของ IBM/PC (หรือ 8 สล็อตใน IBM PC/XT) ซึ่งสัญญาณต่าง ๆ จะแตกต่างจากบัสไซเคิลที่ปรากฏบนบัสของ 8088 อยู่บ้าง เช่น ขา A0 – A19 และ D0 – D7 ซึ่งแยกจากกันอยู่แทนที่จะอยู่ร่วมกันเหมือนบนบัสของ 8088 เป็นต้น อย่างไรก็ตามเนื่องจากบัสไซเคิลในการตอบรับการขออินเทอร์รัพท์นั้น เป็นบัสไซเคิลที่เกิดขึ้นเฉพาะในบัสของระบบเท่านั้น โดยจะไม่ปรากฏบนสล็อตทั้ง 5 ของ IBM/PC ดังนั้นในที่นี้จึงไม่กล่าวถึงบัสไซเคิลในการตอบรับการขออินเทอร์รัพท์นี้

4.5.1 บัสไซเคิลในการอ่านข้อมูลจากหน่วยความจำ

8088 จะทำการสร้างบัสไซเคิลนี้ขึ้นในช่วงที่ 8088 ทำการเฟทช์ (Fetch) คำสั่งหรืออ่านข้อมูลจากหน่วยความจำ ซึ่งหน่วยความจำที่กล่าวถึงนี้อาจจะเป็นหน่วยความจำที่อยู่บนเมนบอร์ด (Main Board), บนการ์ดที่เสียบอยู่บนสล็อต, ROM หรือ RAM ก็ได้สำหรับขบวนการของสัญญาณที่เกิดขึ้นในบัสไซเคิลนี้ สามารถจะแสดงได้ดังภาพที่ 4.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



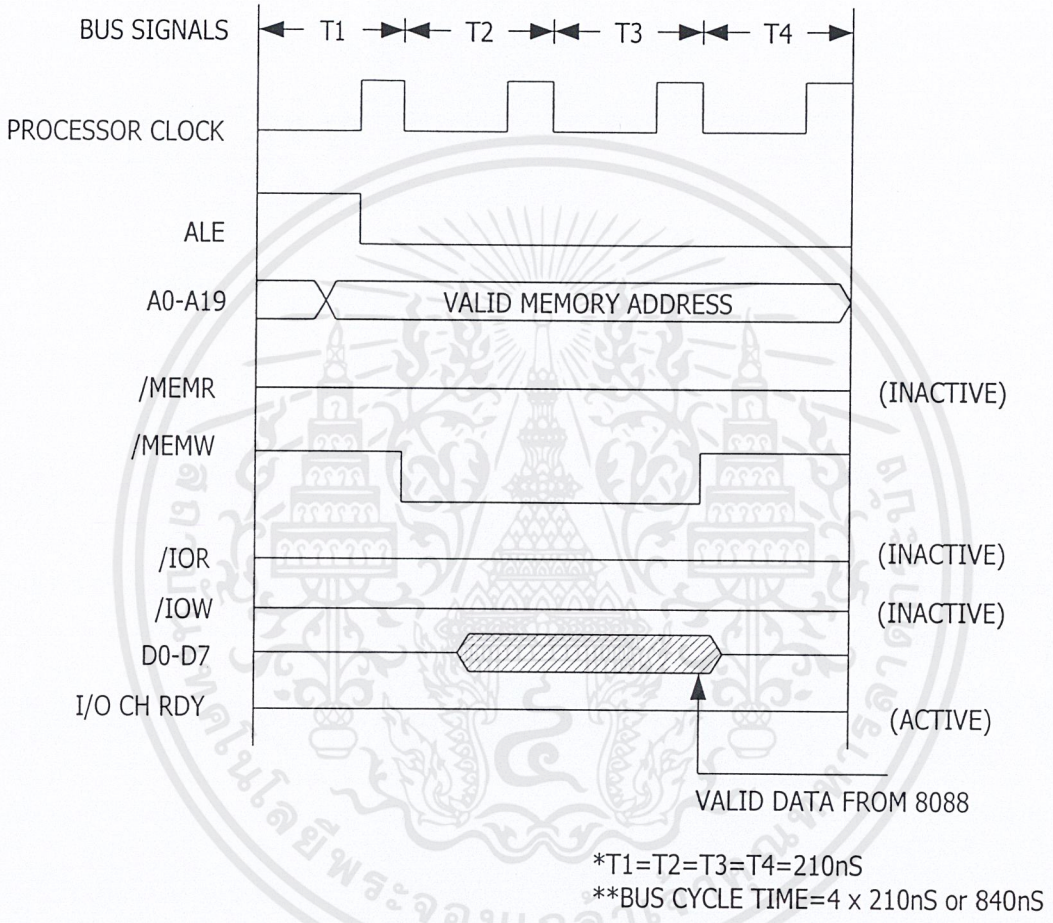
ภาพที่ 4.2 แสดงบัสไซเคิลของการอ่านข้อมูลจากหน่วยความจำ

จากภาพที่ 5.2 บัสไซเคิลนี้ จะเริ่มต้นในช่วงเวลาของคล็อก T1 (ในที่นี้จะเรียกคล็อกลูกแรกในบัสไซเคิลที่ 8088 สร้างขึ้นเป็น T1, ลูกที่สองเป็น T2,...) ซึ่งเป็นช่วงเวลาที่สัญญาณ ALE แอคทีฟ (ลอจิก "1") สัญญาณ ALE นี้จะถูกใช้เพื่อแสดงให้อุปกรณ์ที่ทำงานร่วมกับ 8088 ทราบว่าข้อมูลที่อยู่บนบัสแอดเดรสในช่วงขอบขาลงของสัญญาณ ALE เป็นแอดเดรสของหน่วยความจำที่ 8088 ต้องการจะติดต่อด้วย (ในที่นี้คือ แอดเดรสของหน่วยความจำที่ 8088 ต้องการอ่านข้อมูล) หลังจากนั้นในช่วงของคล็อก T2 สัญญาณ $\overline{\text{MEMR}}$ จะแอคทีฟ (ลอจิก "0") ซึ่งเป็นการแสดงที่ทำให้อุปกรณ์ที่ทำงานร่วมกับ 8088 ทราบว่าบัสไซเคิลนี้เป็นบัสไซเคิลในการอ่านข้อมูลจากหน่วยความจำ (Memory – Read Bus Cycle) และเป็นการทำให้หน่วยความจำที่มีแอดเดรสตรงกับค่าแอดเดรสที่อยู่บนบัสแอดเดรสนั้นส่งข้อมูลออกมาบนบัสข้อมูล เมื่อหน่วยความจำส่งข้อมูลออกมาบนบัสข้อมูลแล้ว 8088 จะทำการอ่านข้อมูลนั้นในช่วงเริ่มต้นของคล็อก T4 จากนั้นสัญญาณ $\overline{\text{MEMR}}$ จะถูกปรับให้กลายเป็นลอจิก "1" และจะสิ้นสุดการทำงานในบัสไซเคิลนี้เมื่อสิ้นสุดคล็อก T4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.2 บัสไซเคิลในการเขียนข้อมูลลงบนหน่วยความจำ

8088 จะทำการสร้างบัสไซเคิลขึ้น ในช่วงที่ 8088 ทำการเขียนข้อมูลลงในหน่วยความจำ สำหรับขบวนการของสัญญาณที่เกิดขึ้นในบัสไซเคิลนี้สามารถแสดงได้ดังภาพที่ 4.3



ภาพที่ 4.3 แสดงบัสไซเคิลของการเขียนข้อมูลลงบนหน่วยความจำ

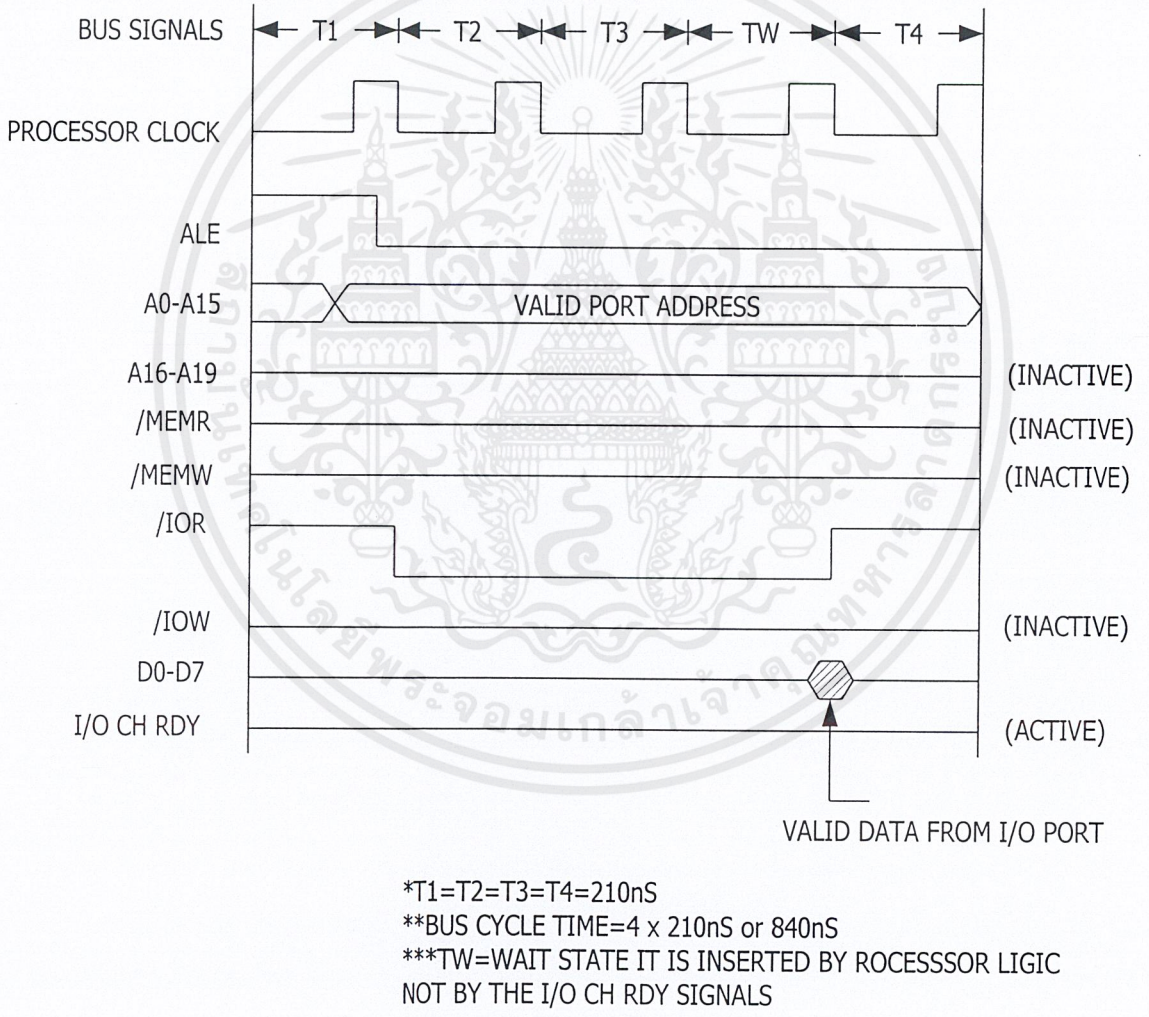
จากภาพที่ 4.3 บัสไซเคิลนี้จะเริ่มต้นในช่วงของคล็อก T1 ซึ่งเป็นช่วงเวลาที่ ALE แอคทีฟ (ลอจิก "1") สัญญาณ ALE นี้จะถูกใช้เพื่อแสดงให้อุปกรณ์ที่ทำงานร่วมกับ 8088 ทราบว่าข้อมูลที่บนบัสแอดเดรสในช่วงขอบขาลงของสัญญาณ ALE นั้น เป็นแอดเดรสของหน่วยความจำที่ 8088 ต้องการติดต่อด้วย (ในที่นี้คือ แอดเดรสของหน่วยความจำที่ 8088 ต้องการเขียนข้อมูลลงไป) หลังจากนั้นในช่วงของคล็อก T2 สัญญาณ $\overline{\text{MEMR}}$ จะแอคทีฟ (ลอจิก "0") ซึ่งเป็นการแสดงให้อุปกรณ์ที่ทำงานร่วมกับ 8088 ทราบว่าบัสไซเคิลนี้เป็นบัสไซเคิลในการเขียนข้อมูลที่ต้องการลงบนหน่วยความจำ (Memory – Write Bus Cycle) จากนั้น 8088 จะทำการส่งข้อมูลที่ต้องการจะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เขียนลงบนหน่วยความจำที่กำหนดนั้นออกมาบนบัสข้อมูล ในช่วงคล็อก T4 สัญญาณ $\overline{\text{MEMR}}$ จะถูกปรับให้กลับเป็นลอจิก "1" และจะสิ้นสุดการทำงานในบัสไซเคิลในการอ่านข้อมูลจากพอร์ท

4.5.3 บัสไซเคิลในการอ่านข้อมูลจากพอร์ท

ในขณะที่ 8088 เอ็กซีคิวท์ (Execute) คำสั่ง IN เช่น IN ac,DATA ซึ่งเป็นคำสั่งที่ทำให้ 8088 ทำการอ่านข้อมูลจากพอร์ทที่กำหนดในส่วนของโอเปอแรนด์ (Operand) นั้น 8088 จะสร้างบัสไซเคิลในการอ่านข้อมูลจากพอร์ท เพื่อให้พอร์ทที่ถูกกำหนดนั้นส่งข้อมูลออกมาบนบัสข้อมูลสำหรับขบวนการของสัญญาณที่เกิดขึ้นในบัสไซเคิลนี้สามารถแสดงได้ดังภาพที่ 4.4



ภาพที่ 4.4 แสดงบัสไซเคิลของการอ่านข้อมูลจากพอร์ท I/O

จากภาพที่ 4.4 บัสไซเคิลนี้จะเริ่มต้นในช่วงของคล็อก T1 ซึ่งเป็นช่วงเวลาที่สัญญาณ ALE แอคทีฟ (ลอจิก "1") สัญญาณ ALE นี้จะถูกใช้เพื่อแสดงให้อุปกรณ์ที่ทำงานร่วมกับ 8088 ทราบว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลที่อยู่บนบัสแอดเดรสในช่วงขอบขาลงของสัญญาณ ALE นั้น เป็นแอดเดรสของพอร์ทที่ 8088 ต้องการจะติดต่อด้วย (ในที่นี้คือ แอดเดรสของพอร์ทที่ 8088 ต้องการอ่านข้อมูล)

สำหรับใน 8088 สามารถอ้างแอดเดรสของพอร์ทได้เพียง 64 K พอร์ทเท่านั้นในขณะที่สามารถอ้างแอดเดรสของหน่วยความจำได้ถึง 1 Mbyte ดังนั้นในกรณีของบัสไซเคิลที่เกี่ยวกับการอ้างแอดเดรสของพอร์ท 8088 จะใช้เส้นแอดเดรสเพียง 16 เส้น คือ A0 – A15 เท่านั้น(ในขณะที่การอ้างแอดเดรสของหน่วยความจำจะใช้เส้นแอดเดรสทั้ง 20 เส้น คือ A0 – A19)

หลังจากนั้นในช่วงของคล็อก T2 สัญญาณ $\overline{\text{IOR}}$ จะแอคทีฟ (ลอจิก “0”) ซึ่งเป็นการแสดงให้อุปกรณ์ที่ทำงานร่วมกับ 8088 ทราบว่าบัสไซเคิลนี้เป็นบัสไซเคิลในการอ่านข้อมูลจากพอร์ท (I/O – Port Read Bus Cycle) และ เป็นการทำให้พอร์ทที่มีแอดเดรสตรงกับค่าแอดเดรสที่อยู่บนบัสแอดเดรสนั้นส่งข้อมูลออกมาบนบัสข้อมูล เมื่อพอร์ทที่ถูกอ้างข้อมูลนั้นในช่วงเริ่มต้นของคล็อก T4 จากนั้นสัญญาณ IOR จะถูกปรับให้เป็นลอจิก “1” และจะสิ้นสุดการทำงานในบัสไซเคิลเมื่อสิ้นสุดช่วงเวลาของคล็อก T4

จะเห็นได้ว่าโดยปกติแล้วบัสไซเคิล จะใช้ช่วงเวลาเท่ากับช่วงเวลาคล็อกจำนวน 4 ลูก เช่นเดียวกับบัสไซเคิลในการอ่าน / เขียนข้อมูลลงหน่วยความจำ แต่ภายใน IBM/PC นั้นจะเห็นช่วงเวลา (Tw) ในบัสไซเคิลนี้ขึ้นอีก 1 ลูกทำให้ช่วงเวลาในบัสไซเคิลเพิ่มขึ้นเป็น 1.05 μsec. โดย Tw นี้จะถูกเพิ่มเข้าไประหว่างช่วงต่อของคล็อก T3 และ T4 เพื่อให้พอร์ท I/O ซึ่งปกติมักจะมีความเร็วในการทำงานต่ำ สามารถที่จะส่งข้อมูลออกมาบนบัสข้อมูลได้ทัน อย่างไรก็ตามในกรณีที่เราต้องการจะเพิ่มช่วงเวลาในบัสไซเคิลนี้ขึ้นอีก (เพิ่มจำนวน Tw) ก็สามารถทำได้ดดยการป้อนลอจิก “0” ให้กับขา I/O CHRDY บนสล็อตของ IBM/PC

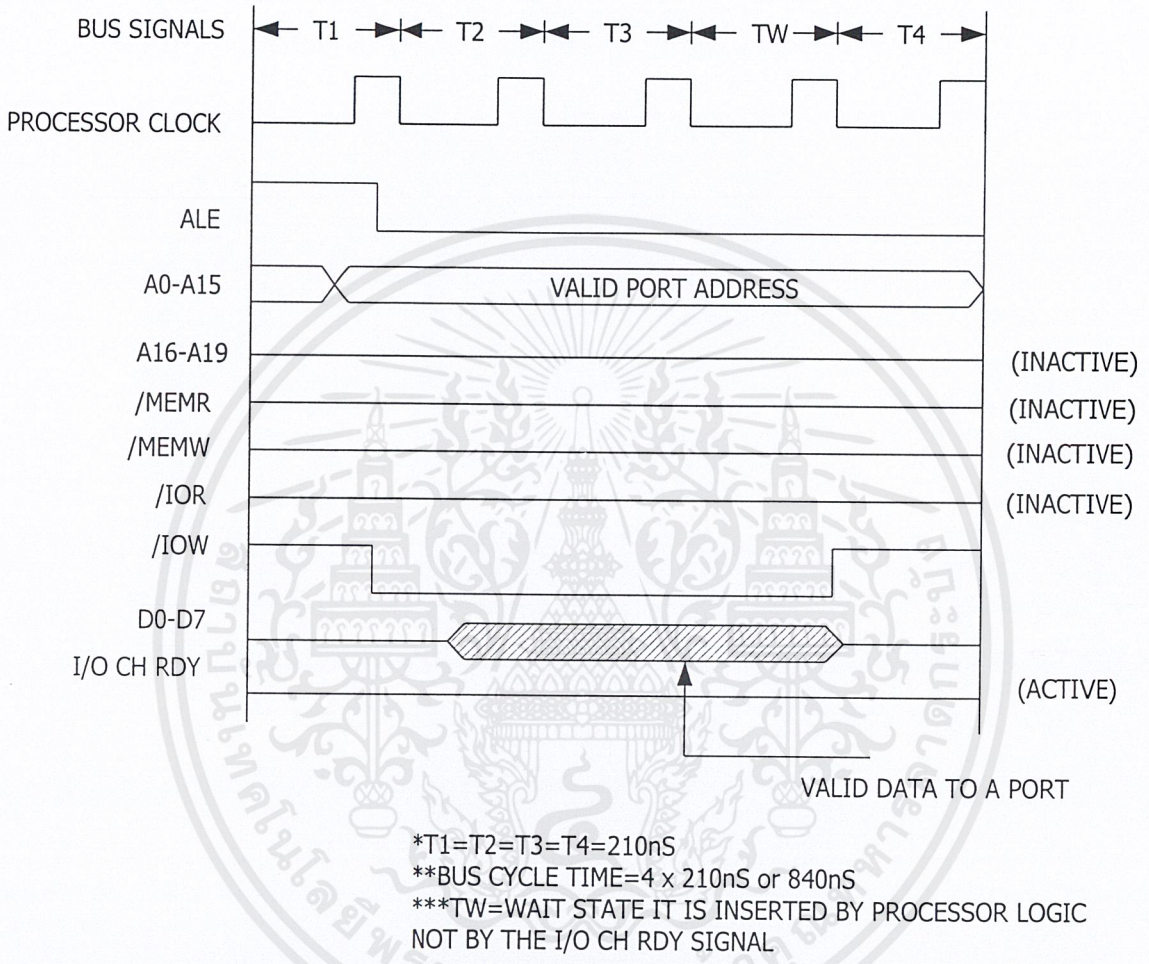
4.5.4 บัสไซเคิลในการเขียนข้อมูลลงบนพอร์ท

ในขณะที่ 8088 เอ็กซิคิวท์ชุดคำสั่ง OUT เช่น OUT DATA,ac ซึ่งเป็นชุดคำสั่งที่ให้ 8088 ทำการเขียนข้อมูลลงบนพอร์ทที่กำหนดในส่วนของโอเปอแรนด์นั้น 8088 จะสร้างบัสไซเคิลในการเขียนข้อมูลลงบนพอร์ท เพื่อให้พอร์ทที่ถูกกำหนดทำการรับข้อมูลที่อยู่บนบัสข้อมูลสำหรับขบวนการของสัญญาณต่าง ๆ ในบัสไซเคิลนี้จะแสดงได้ดังภาพที่ 4.5

บัสไซเคิลนี้จะเริ่มต้นในช่วงของคล็อก T1 ซึ่งเป็นช่วงเวลาที่สัญญาณ ALE แอคทีฟ (ลอจิก “1”) สัญญาณ ALE นั้นเป็นแอดเดรสของพอร์ทที่ 8088 ต้องการจะติดต่อด้วย (ในที่นี้คือ แอดเดรสของพอร์ทที่ 8088 ต้องการส่งข้อมูลให้) สำหรับเส้นแอดเดรสที่ใช้ในบัสไซเคิลนี้จะมีจำนวน 16 เส้นเท่ากับในกรณีของบัสไซเคิลในการอ่านข้อมูลจากพอร์ท คือใช้เส้นแอดเดรส A0 – A15 ในการอ้างแอดเดรสของพอร์ทนั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากนั้นในช่วงของคล็อก T2 สัญญาณ $\overline{IOW}$ จะแอคทีฟ (ลอจิก “0”) ซึ่งเป็นการแสดงให้อุปกรณ์ที่ทำงานร่วมกับ 8088 ทราบว่าบัสไซเคิลนี้เป็นบัสไซเคิลในการเขียนข้อมูลลงบนพอร์ท (I/O – Port Write Bus Cycle) จากนั้น 8088 จะทำการส่งข้อมูลที่ต้องการจะส่งให้กับพอร์ท



ภาพที่ 4.5 แสดงบัสไซเคิลของการเขียนข้อมูลลงบนพอร์ท I/O

ที่กำหนดนั้นออกมาบนบัสข้อมูล ในช่วงของคล็อก T4 สัญญาณ $\overline{IOW}$ จะถูกปรับให้กลับเป็นลอจิก “1” และจะสิ้นสุดการทำงานในบัสไซเคิลเมื่อสิ้นสุดช่วงเวลาของคล็อก T4

สำหรับในกรณีของบัสไซเคิลนี้ IBM/PC ก็จะทำเพิ่ม Tw เข้าไประหว่างช่วงต่อของคล็อก T3 และ T4 เช่นเดียวกับในกรณีของบัสไซเคิลในการอ่านข้อมูลจากพอร์ทเพื่อให้พอร์ท I/O สามารถที่จะทำงานในบัสไซเคิลนี้ได้ทัน อย่างไรก็ตามถ้าต้องการจะเพิ่มช่วงเวลาในบัสไซเคิลขึ้นอีก ก็สามารถจะทำได้โดยการป้อนลอจิก “0” ให้กับขา I/O CHRDY บนสล็อตของ IBM/PC

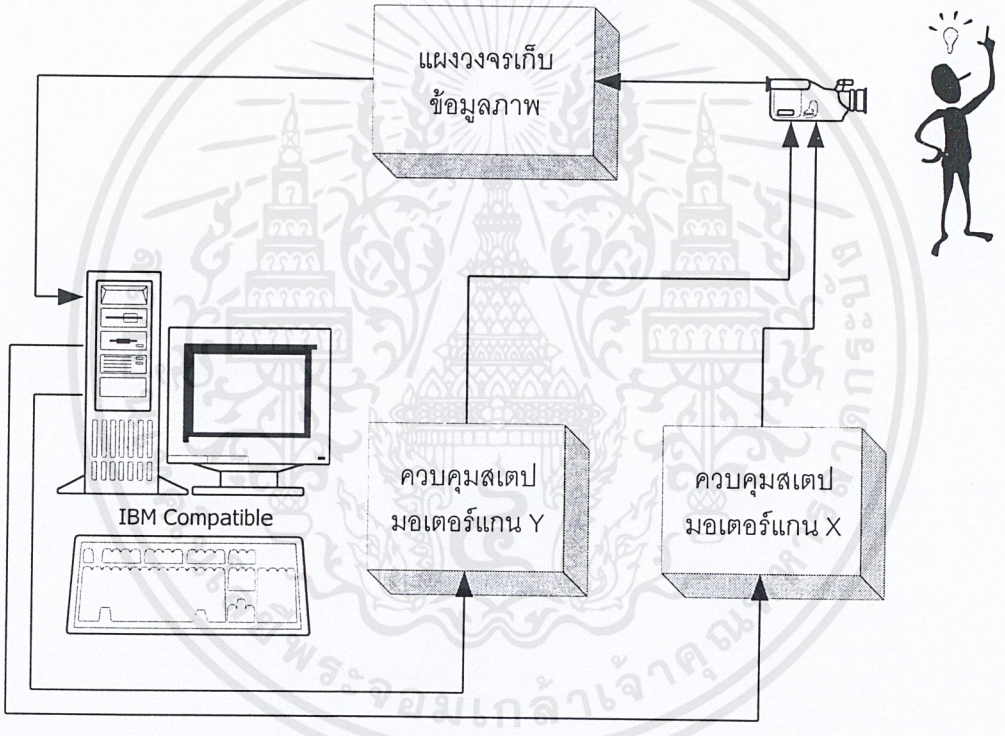
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

ผลการทดลอง

5.1 กล่าวนำ

จากข้อมูลการเก็บข้อมูลภาพ และการควบคุมทิศทางการหมุนของสเตปมอเตอร์ เราได้ทำการสร้างฐานควบคุมทิศทางการหมุนของกล้องขึ้นมา แล้วทำการเขียนโปรแกรมตรวจจับการเคลื่อนที่ของวัตถุที่กล้องสามารถจับได้ โดยมีบล็อกไดอะแกรมดังภาพที่ 5.1



ภาพที่ 5.1 แสดงบล็อกไดอะแกรมของการตรวจจับการเคลื่อนที่ของวัตถุ

5.2 ผลการทดลองการตรวจจับการเคลื่อนที่ของวัตถุ

หลักการทำงานของการตรวจจับการเคลื่อนที่ของวัตถุ เราจะนำเอาภาพที่เก็บได้ 2 ภาพ ที่คนละเวลากัน โดยให้ภาพแรกที่เก็บได้เป็นภาพอดีต และภาพที่เก็บได้ในเวลาต่อมาเป็นภาพปัจจุบัน นำเอามาลบกันด้วยสมการ 5.1

$$S(x,y) = |f_n(x,y) - f_{n-1}(x,y)| \tag{5.1}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6
สรุปผลการวิจัยและข้อเสนอแนะ

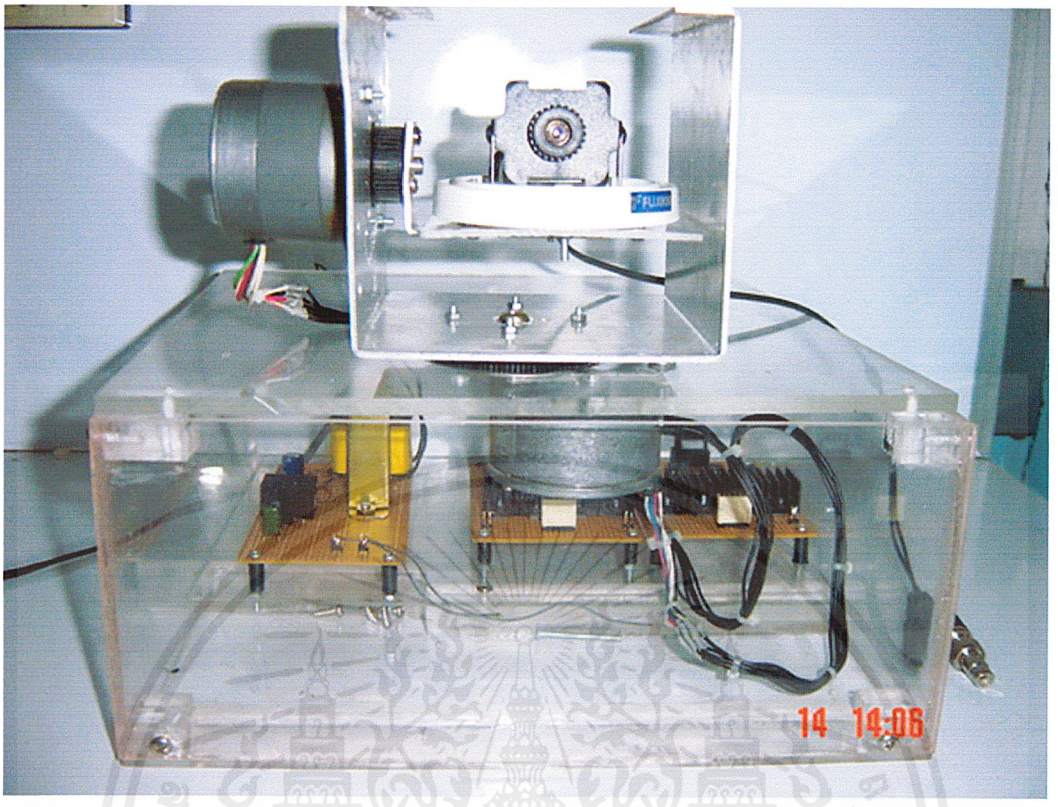
6.1 บทสรุป

ปริญญานิพนธ์นี้มีวัตถุประสงค์ในการศึกษาเพื่อสร้างชุดตรวจจับการเคลื่อนที่ของวัตถุโดยทำการสร้างแผงเก็บข้อมูลภาพขึ้นมาทำการเก็บข้อมูลภาพ เพื่อนำไปใช้ในการประมวลผลตรวจจับวัตถุที่มีการเคลื่อนที่ซึ่งในการออกแบบแผงวงจรเก็บข้อมูลภาพได้ใช้ตัวแปลงสัญญาณแอนะล็อกเป็นดิจิตอลเพื่อเก็บข้อมูลจากภาพขนาด 512×512 จุดต่อภาพ โดยภาพที่จะทำการเก็บข้อมูลนั้นจะต้องมีความละเอียดเพียงพอ ที่จะนำมาเป็นข้อมูลสำหรับการประมวลผล และในการเขียนโปรแกรมขึ้นมาสำหรับใช้ในการประมวลผลกับข้อมูลของภาพที่ได้จากแผงวงจรเก็บข้อมูลภาพนั้น แล้วจะนำผลลัพธ์ที่ได้จากการประมวลผลเพื่อนำไปควบคุมทิศทางการหมุนของกล้องตามการเคลื่อนที่ของวัตถุ แต่ในการอ่านข้อมูลภาพจากหน่วยความจำบนแผงวงจรเก็บข้อมูลภาพนั้น ในบางครั้งจะเกิดค่าผิดพลาดขึ้นมา เนื่องจากวงจรแรงดันอ้างอิงที่ใช้จ่ายให้กับตัวแปลงสัญญาณแอนะล็อกเป็นดิจิตอล มีเสถียรภาพน้อย เพราะใช้วงจรเปรียบเทียบแรงดันแบบธรรมดา สำหรับค่าความผิดพลาดที่เกิดขึ้นนั้น สามารถยอมรับได้ ซึ่งทำให้สามารถประหยัดค่าใช้จ่ายลงได้ เมื่อเรานำชุดตรวจจับการเคลื่อนที่ของวัตถุไปทำการตรวจจับการเคลื่อนที่ของคนที่อยู่ในห้องพบว่าสามารถตรวจจับการเคลื่อนที่ได้จริง แต่ความเร็วในการตรวจจับนั้นยังไม่เร็วเท่าที่ควร อันเนื่องมาจากสาเหตุหลายอย่าง เช่น ความเร็วในการเก็บข้อมูลภาพเข้ามาในคอมพิวเตอร์ ความละเอียดของสเตปมอเตอร์ยังน้อยเกินไป เป็นต้น

6.2 ข้อเสนอแนะ

สำหรับแนวทางในการพัฒนาต่อไปนั้น จะได้ทำการเพิ่มประสิทธิภาพให้กับแผงวงจรเก็บข้อมูลภาพในด้านความเร็วของการจัดการหน่วยความจำ เช่น การนำวิธีการของ DMA (Direct-Memory Access) มาช่วยในการส่งผ่านข้อมูลเข้าสู่เครื่องคอมพิวเตอร์ หรือการเพิ่มส่วนของการประมวลภาพที่ซับซ้อน เช่น การแปลงฟาสฟูเรียร์ (FFT; Fast Furrier Transform) การหาขอบภาพ (Edge detection) การแปลง DCT (Discrete Cosine Transform) เป็นต้น ซึ่งการทำงานด้วยฟังก์ชั่นเหล่านี้โดยปกติต้องอาศัยชิปการประมวลภาพ (DSP; Digital Signal Processing) ในการประมวลผล และในส่วนของสเตปมอเตอร์ควรใช้มอเตอร์ที่มีความละเอียดสูง จะทำให้การตรวจจับมีความแม่นยำและความเร็วที่สูงขึ้นได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



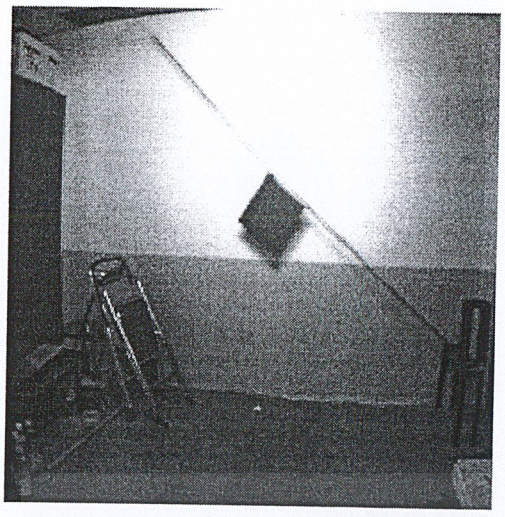
ภาพที่ 5.2 แสดงชุดตรวจจับการเคลื่อนที่ของวัตถุ

โดยให้ $f_n(x,y)$ เป็นข้อมูลภาพปัจจุบัน $f_{n-1}(x,y)$ เป็นข้อมูลภาพในอดีต และ $S(x,y)$ คือผลลัพธ์ที่ได้จากการลบภาพทั้ง 2 แต่จากการทดลองภาพ 2 ภาพที่เหมือนกันแต่เก็บในเวลาที่แตกต่างกันจะเกิดความแตกต่างขึ้นเล็กน้อย อันเนื่องมาจากผลกระทบหลายอย่าง เช่น แสงที่มาตกกระทบ หรืออาจจะเกิดจากความผิดพลาดที่แผงวงจรเก็บข้อมูลภาพ เราจึงใช้สมการที่ 5.2 ในการแก้ข้อผิดพลาดที่เกิดขึ้น

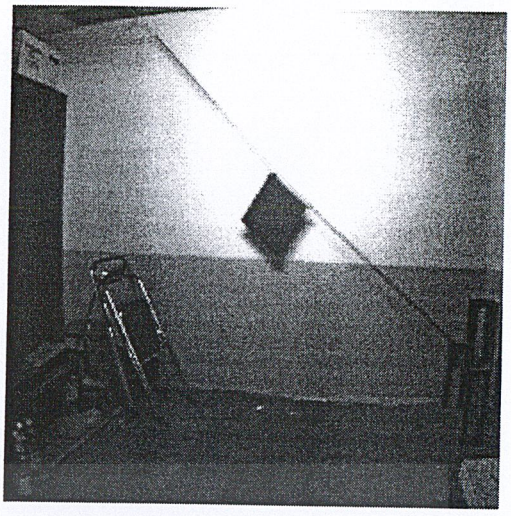
$$S(x,y) = \begin{cases} S(x,y) \;;| f_n(x,y) - f_{n-1}(x,y) > th \\ 0 \end{cases} \qquad (5.2)$$

จากสมการ 5.2 ให้ th เป็นค่า Threshold ที่ตั้งไว้ถ้าผลลัพธ์จากการลบภาพทั้ง 2 มากกว่าค่า Threshold จะเป็นผลต่างของภาพทั้ง 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ก) ภาพอดีต $f_{n-1}(x,y)$



(ข) ภาพปัจจุบัน $f_n(x,y)$

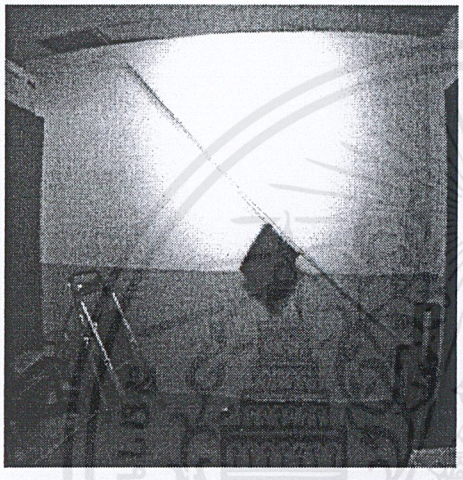
(ค) ภาพผลลัพธ์ของภาพปัจจุบันลบด้วยภาพอดีตในขณะที่วัตถุยังไม่มีการเคลื่อนที่

ภาพที่ 5.3 (ก) ภาพอดีต $f_{n-1}(x,y)$ (ข) ภาพปัจจุบัน $f_n(x,y)$ (ค) ภาพผลลัพธ์ของภาพ ปัจจุบันลบด้วยภาพอดีตขณะที่วัตถุยังไม่มีการเคลื่อนที่

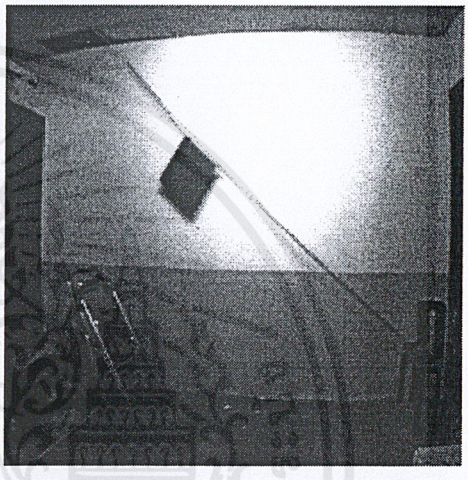
ซึ่งจากภาพที่ 5.3 (ก) และ 5.3 (ข) เป็นภาพวัตถุที่จับได้ แต่วัตถุไม่มีการเคลื่อนที่ เมื่อนำภาพ 2 ภาพมาลบกันตากมาการจะไม่มีผลต่างของภาพเกิดขึ้นดังแสดงในภาพที่ 5.3 (ค) ซึ่งเป็นผลลัพธ์ที่ได้จากการลบภาพทั้ง 2 เมื่อไม่มีผลต่างของภาพเกิดขึ้นส่วนควบคุมการหมุนของกล้องจะยังไม่ทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

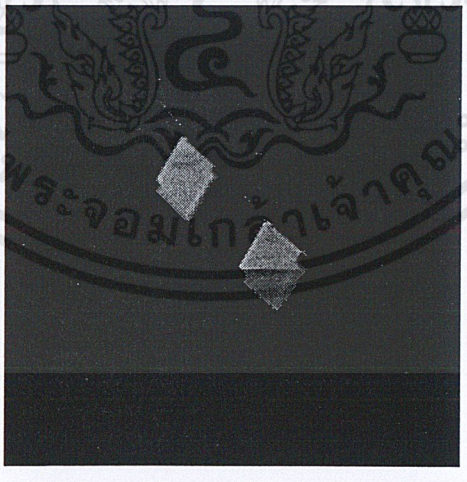
แต่เมื่อวัตถุที่จับได้มีการเคลื่อนที่ เมื่อนำภาพ 2 ภาพมาลบกัน จะเกิดกลุ่มผลต่างของข้อมูลภาพขึ้นมา 2 กลุ่ม คือกลุ่มผลต่างของข้อมูลภาพที่เป็นบวก กับกลุ่มผลต่างของข้อมูลภาพที่เป็นลบ โดยใช้หลักการที่ว่ากลุ่มข้อมูลที่เป็นวัตถุจะมีค่าระดับสีเทาต่ำคือเป็นสีดำที่อยู่ในภาพ ส่วน Background จะมีค่าระดับสีเทาสูง คือ เป็นสีขาวในภาพเมื่อนำภาพปัจจุบันลบด้วยด้วยภาพอดีต กลุ่มข้อมูลภาพที่เป็นวัตถุในภาพอดีตจะมีค่าเป็นบวก ส่วนกลุ่มข้อมูลภาพที่เป็นวัตถุในภาพปัจจุบันจะมีค่าเป็นลบดังนั้นเราจะได้ว่ากลุ่มข้อมูลภาพวัตถุที่เป็นบวกจะเคลื่อนที่ไปหากลุ่มข้อมูลภาพที่เป็นลบ โดยภาพผลลัพธ์ที่ได้แสดงดังภาพที่ 5.4



(ก) ภาพอดีต $f_{n-1}(x,y)$



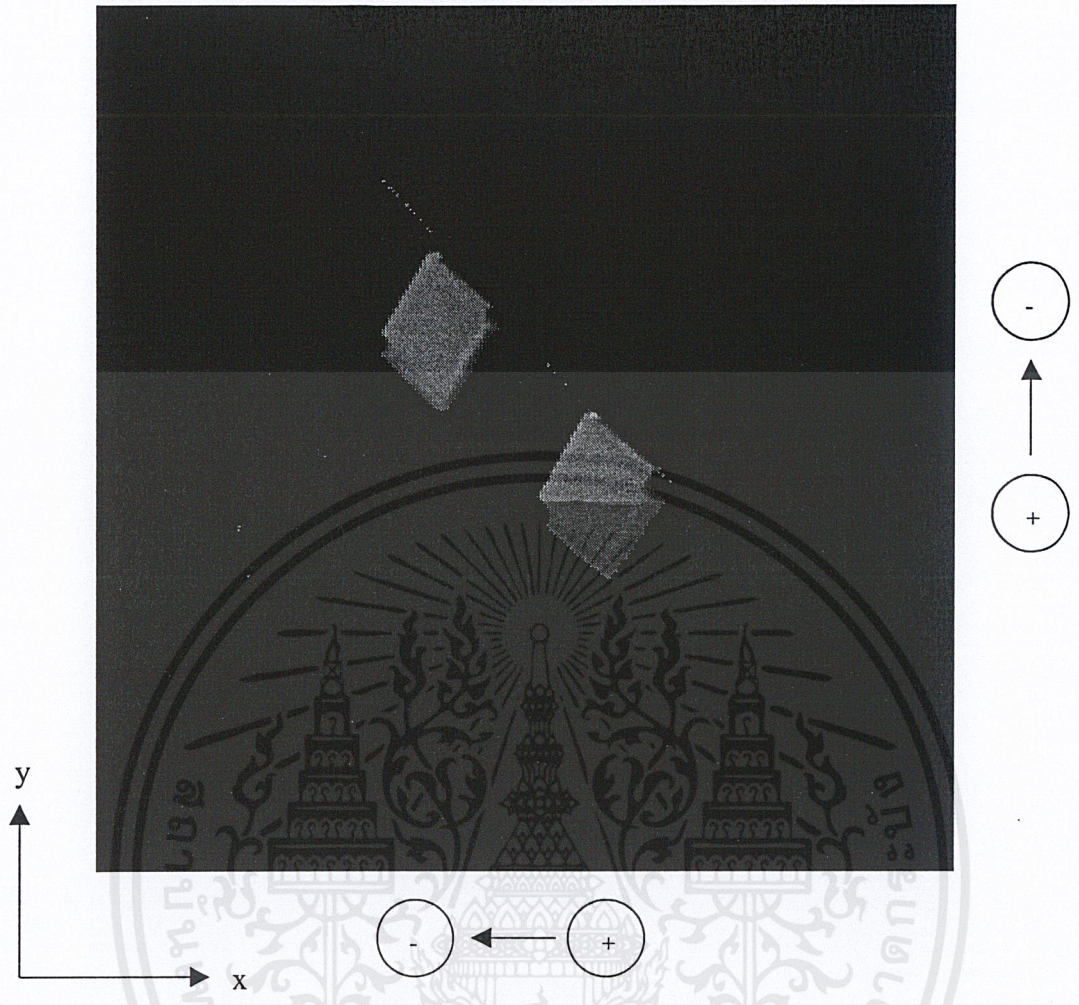
(ข) ภาพปัจจุบัน $f_n(x,y)$



(ค) ภาพผลลัพธ์ของภาพปัจจุบันลบด้วยภาพอดีตในขณะที่วัตถุเคลื่อนที่ขึ้นไปทางซ้าย

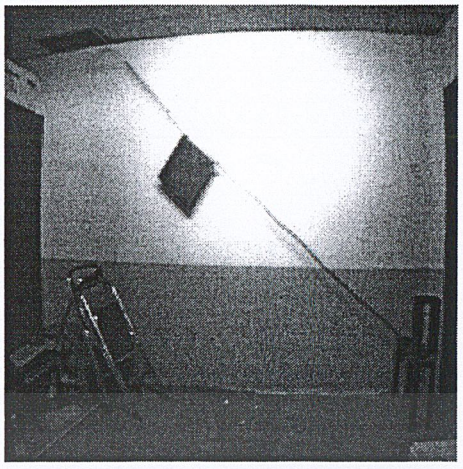
ภาพที่ 5.4 (ก) ภาพอดีต $f_{n-1}(x,y)$ (ข) ภาพปัจจุบัน $f_n(x,y)$ (ค) ภาพผลลัพธ์ของภาพ ปัจจุบันลบด้วยภาพอดีตในขณะที่วัตถุเคลื่อนที่ขึ้นไปทางซ้าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

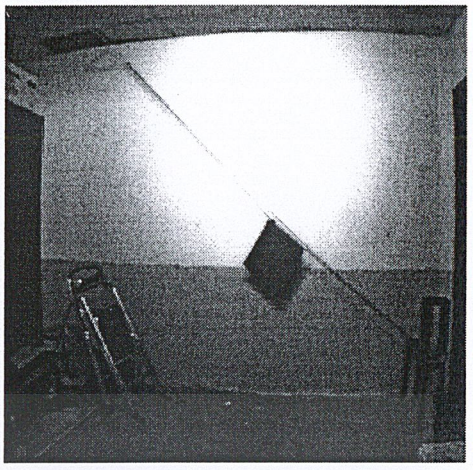


ภาพที่ 5.5 แสดงทิศทางการเคลื่อนที่ของกล้องตามแนวแกน x และแกน y เมื่อวัตถุเคลื่อนที่ขึ้นไปทางด้านบนซ้าย

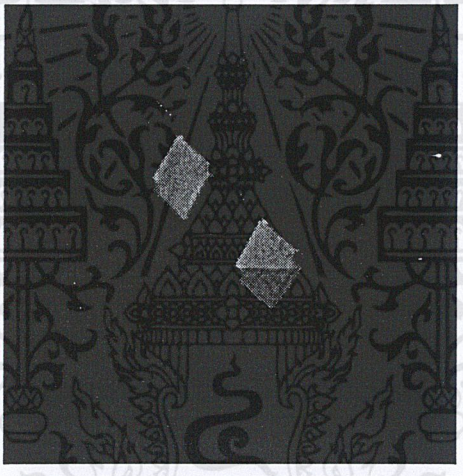
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ก) ภาพอดีต $f_{n-1}(x,y)$



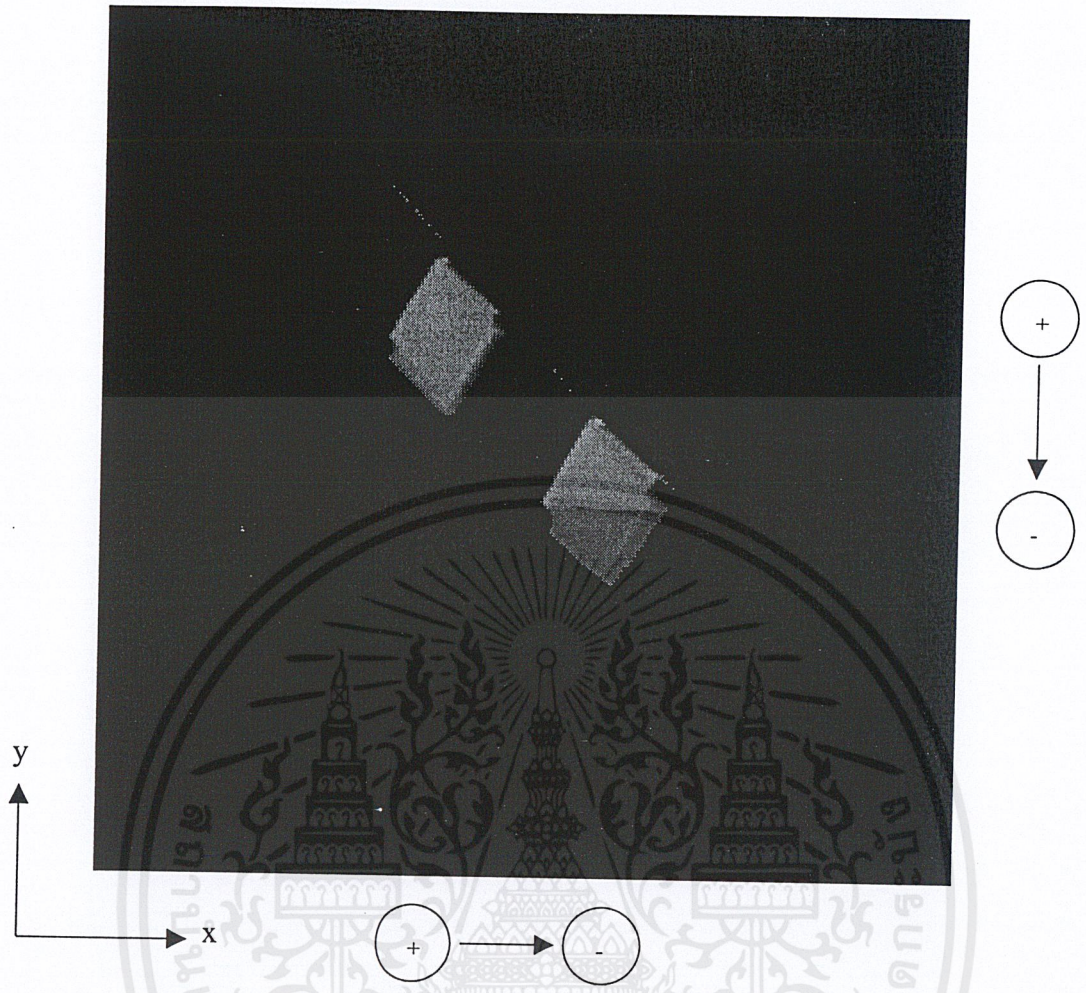
(ข) ภาพปัจจุบัน $f_n(x,y)$



(ค) ภาพผลลัพธ์ของภาพปัจจุบันลบด้วยภาพอดีตในขณะที่วัตถุเคลื่อนที่ลงมาทางด้านขวา

ภาพที่ 5.6 (ก) ภาพอดีต $f_{n-1}(x,y)$ (ข) ภาพปัจจุบัน $f_n(x,y)$ (ค) ภาพผลลัพธ์ของภาพ ปัจจุบันลบด้วยภาพอดีตในขณะที่วัตถุเคลื่อนที่ลงมาทางด้านขวา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 5.7 แสดงทิศทางการเคลื่อนที่ของกล้องตามแนวแกน x และแกน y เมื่อวัตถุเคลื่อนที่ลงมาทางด้านล่างขวา

จากหลักการที่ได้เราจะทำการเขียนโปรแกรมหาระยะห่างของวัตถุทั้ง 2 ที่เคลื่อนที่ไปทั้งแนวแกนตั้งและแนวแกนนอนเพื่อนำผลลัพธ์ที่ได้มาขับสเตปมอเตอร์ให้หมุนกล้องไปตามตำแหน่งที่วัตถุเคลื่อนที่ไป

5.3 สรุปผลการทดลอง

จากการทดลองนำชุดตรวจจับการเคลื่อนที่ของวัตถุทำการตรวจจับการเคลื่อนที่ของคนที่เดินอยู่ในห้อง พบว่าสามารถทำการตรวจจับการเคลื่อนที่ได้จริงตามทฤษฎีแต่ว่าความเร็วในการตรวจจับนั้นยังไม่เร็วเท่าที่ควร เนื่องจากสาเหตุหลายอย่าง เช่น ความเร็วในการอ่านเขียนข้อมูลระหว่างคอมพิวเตอร์กับแผงวงจรเก็บข้อมูลภาพ ความละเอียดของสเตปมอเตอร์ที่นำมาใช้ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. รองศาสตราจารย์ มัณฑนา ปราการสมุทร, " การเขียนชุดคำสั่งภาษาซี ", มหาวิทยาลัยจุฬาลงกรณ์มหาวิทยาลัย , มิถุนายน 2534
2. รองศาสตราจารย์ นุกูล กระจาย , " การเขียนโปรแกรมในดอส และวินโดวส์ด้วยบอร์แลนด์ C++ 5.0 " , ซีเอ็ดยูนิเคชั่น , 2540
3. Lewis C. Eggebrecht , " Interfacing to the IBM Personal Computer " , Howard w. Sams & Co.,lnc , 1983
4. บัณฑิต สุมนวัตนเดช , " แผงวงจรเก็บข้อมูลภาพที่ประกอบด้วยฟังก์ชั่นหน่วยความจำ " , วิทยานิพนธ์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง , 2539
5. ชนะธิป รักชลธี , " ไมโครสเตป สเตปมอเตอร์ คอนโทรลเลอร์ " , ปริญญานิพนธ์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง , 2538

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

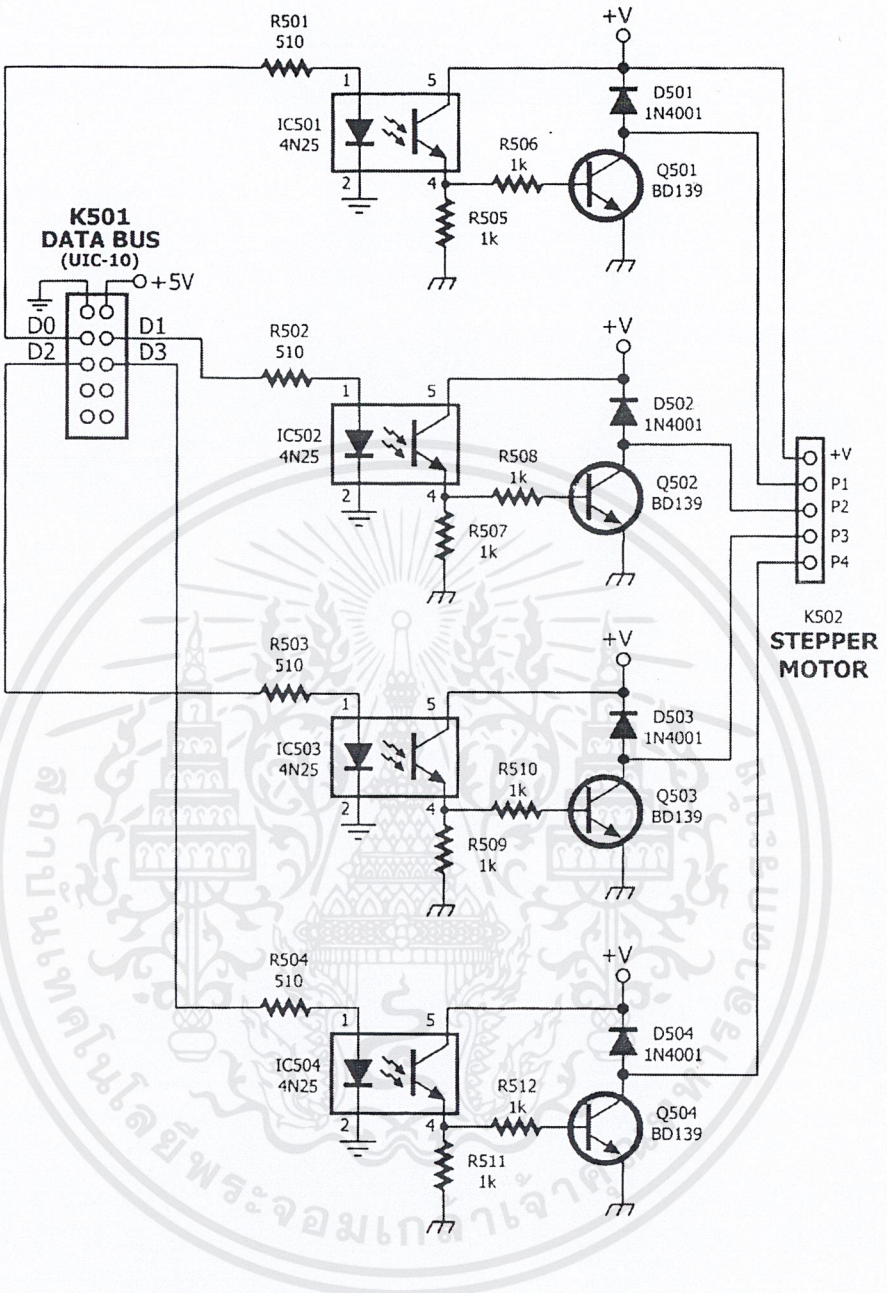
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

วงจรรวม

ภาพที่ ก-1 แสดงวงจรชุดขับสเตปมอเตอร์
ภาพที่ ก-2 แสดงวงจรของวงจรเก็บข้อมูลภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ ก-1 แสดงวงจรชุดขับสเตปมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ ก-2 แสดงวงจรของแผงวงจรเก็บข้อมูลภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้