

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เครื่องเล่น MP3

MP3 Player



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องเล่น MP3

MP 3 Player

โดย

นายสุวิทย์ คุ่มภัย เลขประจำตัว 47015230

นายโสภณ อินริสพงศ์ เลขประจำตัว 47015231

อาจารย์ที่ปรึกษา

ดร.กิตติพล ชิตสกุล

ปริญญานิพนธ์สำหรับปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2549

ภาควิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เครื่องเล่น MP3 (MP3 Player)

ผู้จัดทำ

1. นายสุวิทย์ คุ่มภัย เลขประจำตัว 47015230
2. นายโสภณ อินริสพงษ์ เลขประจำตัว 47015231



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีก้ารนำไปใช้

## เครื่องเล่น MP3

นายสุวิทย์ กุ่มกัย รหัส 47015230  
 นายโสภณ อินริสพงษ์ รหัส 47015231  
 ดร.กิตติพล ชิตสกุล อาจารย์ที่ปรึกษา  
 ปีการศึกษา 2549

### บทคัดย่อ

โครงการเครื่องเล่น MP3 นี้เป็นการประยุกต์ใช้ CD-ROM มาสร้างเป็นเครื่องเล่น MP3 ที่สามารถเล่นแผ่น CD ได้โดยไม่จำเป็นต้องต่อกับไมโครคอมพิวเตอร์ ฟังก์ชันต่างๆ ได้แก่ Eject, Close, Play, Stop จะถูกควบคุมการด้วยไมโครคอนโทรลเลอร์ตระกูล AVR เบอร์ ATMEGA162 ผ่านทางพอร์ต IDE Interface ATAPI ส่วนการถอดรหัสใช้ไอซีถอดรหัสเบอร์ VS1011B เปลี่ยนไฟล์ MP3 ให้เป็นสัญญาณเสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## MP3 PLAYER

Mr. Suwit KHUMPAI ID. 47015230

Mr. Sapon INRITSAPONG ID. 47015231

Dr. Kitiphol CHITSAKUL Advisor

Educational Year 2006

### Abstract

This MP3 Player is an application of CD-ROM to be able to play audio files type MP3 without interfacing to a microcomputer. The function such as Eject, Close, Play, Stop can be controlled by microcontroller AVR family ATMEGA162 via a port IDE interface ATAPI. A VS1011B is used as decoder to convert MP3 file into voice signal.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## กิตติกรรมประกาศ

ในการศึกษา การทดลอง จนกระทั่งตัวโครงการรวมทั้งวิทยานิพนธ์ฉบับนี้สำเร็จลุล่วง ซึ่ง  
จะสำเร็จไปไม่ได้เลยถ้าหากไม่ได้รับความกรุณาจากอาจารย์ที่ปรึกษาคือ ดร. กิตติพล ชิตสกุล ที่คอย  
ให้ความรู้ ให้คำปรึกษาที่มีประโยชน์ รวมทั้งต้องกล่าวขอบคุณรุ่นพี่ที่ให้คำปรึกษา แนวทางการ  
แก้ปัญหาตลอดมา



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ

	หน้า
บทคัดย่อ	I
Abstract	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูป	IX
สารบัญตาราง	XI
<b>บทที่ 1 บทนำ</b>	<b>1</b>
1.1 ความเป็นมาของโครงการ	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 ประโยชน์ที่คาดว่าจะได้รับ	1
1.4 วิธีการดำเนินงาน	2
1.5 โครงสร้างปริญญานิพนธ์	2
<b>บทที่ 2 ทฤษฎีและหลักการ</b>	<b>3</b>
2.1 ความหมายของเอ็มเป็ก (MPEG)	3
2.2 เอ็มเป็ก 1 เลเยอร์ 3 (MP3)	3
2.2.1 ระบบโมโน	4
2.2.2 ระบบสเตอริโอ	4
2.2.3 ระบบสเตอริโอ	4
2.2.4 ระบบจอยท์-สเตอริโอ	4
2.3 คุณภาพเสียงที่ได้	5
2.4 โครงสร้างของข้อมูลไฟล์เอ็มเป็ก 1 เลเยอร์ 3	9
2.5 ชิพถอดรหัสข้อมูลเอ็มเป็ก VS1002d	10
2.5.1 คุณสมบัติของ VS1002d	10
2.5.2 ข้อมูลของขาใน VS1002d	11
2.5.3 การนำชิพ VS1002d ไปต่อใช้งาน	13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
2.6 ส่วนที่เชื่อมต่อข้อมูลอนุกรม SPI ของชิพ VS1002d สำหรับคาค้า	13
(SDI: Serial Protocol for Serial Data Interfacd)	
2.6.1 หน้าทีสำหรับขาส่งข้อมูลอนุกรม SPI	14
2.6.2 คุณลักษณะสำหรับการส่งแบบอนุกรมสำหรับการส่งข้อมูล (SDI)	14
2.7 ส่วนเชื่อมต่อข้อมูลอนุกรม SPI ของชิพ VS1002d สำหรับคอนโทรล	15
(SCI: Serial Protocol for Serial Command Interface)	
2.7.1 การอ่านข้อมูลสำหรับ SCI (SCI Read)	16
2.7.2 การเขียนข้อมูลสำหรับ SCI (SCI Write)	16
2.7.3 ลักษณะการทำงาน	17
2.8 SCI รีจิสเตอร์	18
2.8.1 รีจิสเตอร์โหมค, โหมค (RW)	18
2.8.2 รีจิสเตอร์สถานะ, STATUS (RW)	20
2.8.3 INT_FCNTLH (-)	20
2.8.4 รีจิสเตอร์สัญญาณนาฬิกา, CLOCKF (RW)	20
2.8.5 รีจิสเตอร์ SRATE (R)	20
2.8.6 รีจิสเตอร์ AUDATA (R)	20
2.8.7 รีจิสเตอร์ WRAM (W)	21
2.8.8 รีจิสเตอร์ WRAMDDR (W)	21
2.8.9 รีจิสเตอร์ HDATA and HDAT1 (R)	21
2.8.10 รีจิสเตอร์ AIADDR (RW)	22
2.8.11 รีจิสเตอร์ VOL (RW)	22
2.8.12 รีจิสเตอร์ A1CTRL[x] (RW)	23
2.9 การทำงานส่วนต่าง ๆ ของชิพ VS1002d	23
2.9.1 สัญญาณนาฬิกา	23
2.9.2 โหมคประหัตพลังงาน	23
2.9.3 การรีเซตตัวเครื่อง	23
2.9.4 การรีเซตโปรแกรม	24
2.9.5 การถอดรหัสข้อมูลเอ็มเป็ก	24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
2.9.6 การตรวจสอบว่ามีการทำงานอย่างปกติหรือไม่	24
2.9.7 การทดสอบ	24
2.9.8 อัตราเร็วสูงสุดของสัญญาณนาฬิกา	25
2.9.9 ความเร็วสูงสุดในการส่งข้อมูล	25
2.9.10 อัตราเร็วสูงสุดของบิตข้อมูลในการถอดรหัส	25
2.9.11 การทำงานเมื่อตอนรีเซต	25
2.9.12 การส่งข้อมูลสู่สายสัญญาณ SCI	26
2.9.13 การส่งข้อมูลสู่ส่วน SDI (ข้อมูลเอ็มเป็ก)	26
<b>บทที่ 3 ไมโครคอนโทรลเลอร์</b>	<b>27</b>
3.1 สถาปัตยกรรม และคุณสมบัติโดยทั่วไปของ ATMEGA162	27
3.2 สถาปัตยกรรมภายในและรีจิสเตอร์ใช้งานทั่วไป	28
3.2.1 สถาปัตยกรรมภายใน	28
3.2.2 รีจิสเตอร์ใช้งานทั่วไป	32
3.3 การรีเซ็ตและการอินเทอร์รัพท์	34
3.3.1 การรีเซ็ตและการอินเทอร์รัพท์	34
3.3.2 สัญญาณรีเซ็ต	35
3.4 Interrupt Handling	37
3.5 พอร์ตอินพุต/เอาต์พุต	37
3.5.1 พอร์ต A	37
3.5.2 พอร์ต B	38
3.5.3 พอร์ต C	39
3.8.4 พอร์ต D	39
3.6 Timer & Counter	40
3.6.1 TIMER0 & COUNTER 0	40
3.6.2 TIMER1 & COUNTER 1	41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
<b>บทที่ 4 โพรโทคอลที่ใช้ในการเชื่อมต่อกับซีดี-รอม</b>	<b>43</b>
4.1 ATA	43
4.1.1 การเชื่อมต่อทางกายภาพ	43
4.1.2 ขาเชื่อมต่อสำหรับรับส่งข้อมูล	44
4.1.3 รีจิสเตอร์ภายใน	46
4.1.4 โพรโทคอลการส่งข้อมูล	47
4.2 ATAPI	49
4.2.1 ATAPI Protocol	49
4.2.2 ATAPI PACKET Command	50
4.2.3 Status Register for Packet Command	50
4.3 ขั้นตอนการส่งคำสั่งแบบ PIO ที่มีข้อมูลเข้าสู่ Host	55
<b>บทที่ 5 ISO 9660</b>	<b>57</b>
5.1 บทนำ	57
5.2 ภาพรวมโครงสร้าง ISO9660	57
5.2.1 The Volume Descriptor	58
5.2.2 โครงสร้างของไครเรคทอรี	61
5.2.3 Path Table	63
<b>บทที่ 6 การสร้างและการออกแบบ</b>	<b>65</b>
6.1 การสร้างและออกแบบการควบคุมซีดีรอม	65
6.2 การสร้างและออกแบบการควบคุมการทำงานของวงจรถอดครีส์	66
6.3 การสร้างและออกแบบ Key Switch	67
6.4 การสร้างและออกแบบแหล่งจ่ายไฟตรง	68
6.5 การสร้างและออกแบบวงจรมายเสียบ	69
6.6 โฟลว์ชาร์ตการทำงานของส่วนต่างๆ	70
6.6.1 โฟลว์ชาร์ตของโปรแกรมหลัก	70
6.6.2 โฟลว์ชาร์ตของการ Play	71

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
6.6.3 โพลีชาร์ตของการเพิ่มและลดเสียง	72
<b>บทที่ 7 การทดลองและผลการทดลอง</b>	<b>73</b>
7.1 การทดลองและผลการทดลองของการเขียนโปรแกรมควบคุมภาคถอดรหัส	73
7.2 การทดลองและผลการทดลองของโครงการงาน	75
<b>บทที่ 8 บทสรุป</b>	<b>76</b>
<b>บรรณานุกรม</b>	<b>77</b>



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป

	หน้า
รูปที่ 2.1 ความไวต่อเสียงความถี่ต่าง ๆ ของมนุษย์	6
รูปที่ 2.2 การบังทางด้านความถี่ (Frequency Masking) ที่ความถี่มาส์ก 1 กิโลเฮิร์ตซ์	7
รูปที่ 2.3 ย่านความถี่วิกฤต (Critical Band) ที่ความถี่ต่าง	8
รูปที่ 2.4 ขอบเขตของช่วงความถี่ที่ถูกบังในการใช้ไซโคอะคูสติก (Psychacoustic)	8
รูปที่ 2.5 ส่วนประกอบภายในหัวข้อมูล (Header)	10
รูปที่ 2.6 ขาต่าง ๆ ของชิพ VS1002d	11
รูปที่ 2.7 การนำชิพ VS1002d ไปต่อใช้งานกับอุปกรณ์ไมโครคอนโทรลเลอร์	13
รูปที่ 2.8 รูปสัญญาณ BSYNC	15
รูปที่ 2.9 สัญญาณการอ่านข้อมูลผ่านพอร์ตอนุกรม	16
รูปที่ 2.10 สัญญาณการเขียนข้อมูลลงพอร์ตอนุกรม	17
รูปที่ 2.11 แผนผังเวลาของพอร์ตอนุกรม	18
รูปที่ 3.1 แผนผังโครงสร้างของ ATMEGA162	29
รูปที่ 3.2 สถาปัตยกรรมแบบ RISC ของ ATMEGA162	30
รูปที่ 3.3 การจัดตำแหน่งหน่วยความจำ SRAM	31
รูปที่ 3.4 การเข้าถึงข้อมูล SRAM ภายใน CPU	32
รูปที่ 3.5 โครงสร้างของรีจิสเตอร์ใช้งานทั่วไป	32
รูปที่ 3.6 ตำแหน่งรีจิสเตอร์ SREG ถูกจัดวางไว้ที่ตำแหน่ง \$3F	33
รูปที่ 3.7 ตำแหน่งรีจิสเตอร์ Stack Pointer ถูกจัดวางไว้ที่ตำแหน่ง \$3E และ \$3D	34
รูปที่ 3.8 Port A Data Register: PORTA	37
รูปที่ 3.9 Port B Data Register: PORTB	38
รูปที่ 3.10 Port C Data Register: PORTC	39
รูปที่ 3.11 Port D Data Register: PORTD	39
รูปที่ 3.12 โครงสร้างของ TIMER/COUNTER0	41
รูปที่ 4.1 การเชื่อมต่อระหว่างตัวแม่กับอุปกรณ์ 2 ตัว	43
รูปที่ 4.2 การเชื่อมต่อระหว่างตัวแม่กับอุปกรณ์ตัวเดียว	44
รูปที่ 4.3 ตัวเชื่อมต่อขนาด 40 ขา ตามมาตรฐาน ATA	44
รูปที่ 4.4 ลักษณะของขั้วไฟเลี้ยงของเครื่องเล่นซีดี	45

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

	หน้า
รูปที่ 4.5 โพลีชาร์ตขั้นตอนการส่งคำสั่งแบบ PIO ที่มีข้อมูลเข้าสู่ Host และขั้นตอนการส่งคำสั่งแบบ PIO ที่มีข้อมูลออกจาก Host	56
รูปที่ 5.1 โครงสร้าง ISO 9660	57
รูปที่ 5.2 d-characters	59
รูปที่ 5.3 a-character	60
รูปที่ 5.4 ระดับชั้นไดเรกทอรี	61
รูปที่ 5.5 ไดเรกทอรีแม่	62
รูปที่ 6.1 วงจรที่ทำการทดลองติดต่อกับ CD-ROM	65
รูปที่ 6.2 วงจรที่ทำการทดลองภาคถอดรหัส	66
รูปที่ 6.3 วงจร Key Switch	67
รูปที่ 6.4 วงจรแหล่งจ่ายไฟตรง	68
รูปที่ 6.5 วงจรขยายเสียง	69
รูปที่ 6.6 โพลีชาร์ตโปรแกรมหลัก	70
รูปที่ 6.7 โพลีชาร์ตโปรแกรมย่อยส่วน Play	71
รูปที่ 6.8 โพลีชาร์ตของการเพิ่มและลดเสียง	72
รูปที่ 7.1 สัญญาณอินพุตความถี่ 1 KHz	73
รูปที่ 7.2 สัญญาณเอาต์พุตที่ได้จากวงจรถอดรหัส	74
รูปที่ 7.3 การตอบสนองความถี่ของวงจรถอดรหัส	74

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญตาราง

	หน้า
ตารางที่ 2.1 อัตราการบีบอัดข้อมูลและความเร็วในการส่งข้อมูลจากเครื่องอ่านของข้อมูล ที่ถูกบีบอัดตามมาตรฐาน MPEG-1	4
ตารางที่ 2.2 ความสัมพันธ์ระหว่างคุณภาพเสียงที่ต้องการกับขนาดของข้อมูลที่ถูกบีบอัด	5
ตารางที่ 2.3 เวลาที่เสียไปในการแปลงข้อมูลเอ็มเป็ก	5
ตารางที่ 2.4 รูปแบบข้อมูลเอ็มเป็ก 1 เลขอร์ 3	9
ตารางที่ 2.5 หน้าทีของขาในชีพ VS1002d	12
ตารางที่ 2.6 หน้าทีของพอร์ตอนุกรม SPI	14
ตารางที่ 2.7 ชุดคำสั่งของชีพ VS1002d	15
ตารางที่ 2.7 SCI รีจิสเตอร์	18
ตารางที่ 2.8 รีจิสเตอร์โหมด	19
ตารางที่ 2.9 รีจิสเตอร์ HDATA and HDAT1 (R)	22
ตารางที่ 3.1 รีเซ็ทและอินเตอร์รัพท์เวกเตอร์	36
ตารางที่ 4.1 รายละเอียดและหน้าทีของแต่ละขาตามมาตรฐาน IDE	45
ตารางที่ 4.2 แอดเดรสของรีจิสเตอร์ตามมาตรฐาน ATA	47
ตารางที่ 4.3 รายละเอียดเกี่ยวกับรีจิสเตอร์ต่างๆทีใช้ใน ATAPI	51
ตารางที่ 5.1 Primary Volume Descriptor	58
ตารางที่ 5.2 ความยาวของ path	62
ตารางที่ 5.3 File Identifier	63
ตารางที่ 7.1 การทำงานของวงจรที Bit Rate ต่างๆ	75
ตารางที่ 7.2 การทำงานของวงจรเมื่อใช้ CD-ROM ทีห้อต่างๆ	75

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาของโครงการ

ในปัจจุบันการเก็บข้อมูลเสียง (Audio) สามารถเก็บบันทึกลงในสื่อต่างๆ ได้หลากหลายรูปแบบ ข้อมูลเสียงที่บันทึกสามารถแบ่งตามลักษณะของสัญญาณได้เป็น 2 รูปแบบ คือ ข้อมูลเสียงประเภทสัญญาณอะนาล็อก และ ข้อมูลเสียงประเภทสัญญาณดิจิทัล ซึ่งข้อมูลเสียงประเภทสัญญาณอะนาล็อกนั้นอาจเก็บข้อมูลในรูปของเทปคาสเซตต์ หรือแผ่นเสียง กล่าวคือข้อมูลที่เก็บสามารถอ่านออกมาได้ในรูปของสัญญาณไฟฟ้าที่มีแรงดันต่าง ๆ กันต่อเนื่องกันไปเรื่อย ๆ เมื่อนำสัญญาณเหล่านี้มาผ่านวงจรขยายและส่งต่อไปยังลำโพงก็จะทำให้เราได้ยินเสียงต้นฉบับ

แต่เนื่องจากการเก็บข้อมูลแบบอะนาล็อกนั้น ไม่สามารถเก็บข้อมูลของเสียงที่มีขนาดใหญ่ได้ นอกจากนั้นเมื่อเก็บไว้เป็นระยะเวลานาน ๆ คุณภาพของสื่อที่ใช้เก็บข้อมูลก็จะเสื่อมลง ทำให้ข้อมูลเสียงที่อ่านออกมาได้แยกลงไปอีก จึงได้มีการคิดค้นการเก็บข้อมูลเสียงแบบใหม่ในรูปของสัญญาณดิจิทัลหรือที่เราคุ้นเคยกันเป็นอย่างดีในปัจจุบันก็คือซีดีรอมนั่นเอง

จึงได้เกิดแนวคิดที่จะเอาเครื่องเล่นซีดีออดิโอ เสียงที่บันทึกลงแผ่นซีดีมีการบีบอัดข้อมูลแบบ MP3 เพื่อให้เก็บข้อมูลได้มากๆ โดยการใช้ไมโครคอนโทรลเลอร์ ในการควบคุมการทำงานของเครื่องเล่นซีดีออดิโอ ผ่านสาย IDE ตามมาตรฐานของ ATAPI ส่วนวงจรถอดรหัสใช้ชิพเบอร์ VS1002d ในตอนแรกจะทำการศึกษาการทำงานของซีดีรอม และศึกษาการทำงานของชิพถอดรหัส หลังจากนั้นจะทำการเขียนโปรแกรมควบคุมการทำงานของซีดีรอมและวงจรถอดรหัส

### 1.2 วัตถุประสงค์ของโครงการ

1. เพื่อศึกษาและทำความเข้าใจกับรูปแบบการติดต่อของเครื่องซีดีรอม
2. เพื่อศึกษาและทำความเข้าใจการใช้งานไมโครคอนโทรลเลอร์ชนิด AVR
3. เพื่อศึกษาและทำความเข้าใจการสร้าง และทดลองบอร์ดควบคุมเครื่องเล่นซีดีออดิโอ
4. เพื่อศึกษาและทำความเข้าใจการบีบอัดข้อมูลแบบ MP3
5. เพื่อเพิ่มพูนประสบการณ์และนำความรู้ที่ศึกษามาใช้งานจริง
6. เพื่อเพิ่มทักษะในการแก้ปัญหาและอุปสรรคที่ได้พบ

### 1.3 ประโยชน์ที่คาดว่าจะได้รับ

1. เพิ่มพูนทักษะและความรู้ต่างๆที่ได้ศึกษามา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับบริการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีกรณีนำไปใช้

2. เพื่อส่งเสริมและสร้างความภาคภูมิใจให้เกิดขึ้น
3. เข้าใจหลักการบีบอัดข้อมูลแบบ MP3
4. สามารถสร้างเครื่องเล่น MP3 ได้
5. เป็นแนวทางในการพัฒนาต่อไปในส่วนที่มีความซับซ้อนมากขึ้น
6. สามารถแก้ปัญหาและอุปสรรคที่ค้นพบได้

#### 1.4 วิธีการดำเนินงาน

1. ศึกษาหลักการและทฤษฎีที่ต้องใช้ในโครงการเป็นอันดับแรก
2. ทดลองเขียน โปรแกรมควบคุมวงจรถอดรหัส
2. ทดลองต่อบอร์ดคิดต่อกับซีดีรอม
4. เก็บรายละเอียด ปัญหา และสรุปผลการทดลอง

#### 1.5 โครงสร้างปริญญานิพนธ์

ปริญญานิพนธ์ได้รวบรวม แนวคิดของการทำโครงการ ทฤษฎีต่างๆที่เกี่ยวข้อง การทดลอง และผลที่ได้ โดยรวบรวมไว้เป็นบทตอนดังนี้

บทที่ 1 กล่าวถึงความเป็นมา วัตถุประสงค์ ขั้นตอนของการทำโครงการ

บทที่ 2 กล่าวถึงความหมายและทฤษฎีของการบีบอัดข้อมูลแบบเอ็มเป็ก (MPEG) รวมถึงคุณสมบัติของซีพียูถอดรหัสข้อมูลเอ็มเป็ก VS1002d

บทที่ 3 กล่าวถึงทฤษฎีไมโครคอนโทรลเลอร์ AVR เบอร์ ATMEGA162

บทที่ 4 กล่าวถึงโปรโตคอลที่ใช้ในการเชื่อมต่อกับ CD-ROM ตามมาตรฐาน ATAPI

บทที่ 5 กล่าวถึง ISO 9660 เป็นมาตรฐานที่ผู้ผลิต CD-ROM ทุกรายใช้เป็นมาตรฐาน

บทที่ 6 กล่าวถึงการสร้างและการออกแบบ โดยใช้ไมโครคอนโทรลเลอร์รวมโฟลว์ชาร์ต ในการเขียนโปรแกรมต่างๆ

บทที่ 8 กล่าวถึงการทดลองและผลการทดลอง

บทที่ 9 กล่าวถึงบทสรุปของโครงการและปัญหาของโครงการนี้

## บทที่ 2

### ทฤษฎีและหลักการ

#### 2.1 ความหมายของเอ็มเป็ก (MPEG)

เอ็มเป็ก (MPEG) ย่อมาจาก Moving Picture Experts Group ซึ่งเป็นชื่อของกลุ่มบุคคลที่ร่วมมือกันสร้างมาตรฐานสากล (International Standard) เพื่อใช้ในการเข้ารหัสข้อมูลภาพและเสียงที่อยู่ในรูปของสัญญาณดิจิทัล ก่อตั้งขึ้นตั้งแต่ปี ค.ศ. 1988 โดยมาตรฐานที่สร้างขึ้นนั้นถูกรับรู้เป็นมาตรฐานสากล ISO/IEC หมายความว่า มาตรฐานนี้มาใช้เข้ารหัสหรือถอดรหัสข้อมูลของตัวเองได้โดยไม่ต้องขออนุญาตหรือลิขสิทธิ์ให้กับผู้ใด

มาตรฐาน MPEG นั้น แบ่งย่อยออกเป็นกลุ่มๆ ตามชนิดของข้อมูลที่ถูกเข้ารหัสและการนำไปใช้งาน ซึ่งในปัจจุบันแบ่งออกได้เป็น 5 กลุ่ม ดังนี้

MPEG-1 เข้ารหัสข้อมูลภาพและเสียง ใช้ในระบบวีดิโอซีดีและเสียงเพลง

MPEG-2 เข้ารหัสข้อมูลภาพและเสียง ใช้ในระบบโทรทัศน์ดิจิทัลและดีวีดี

MPEG-3 เข้ารหัสข้อมูลมัลติมีเดียที่ใช้กันอยู่ในเว็บเพจ

MPEG-7 เป็นมาตรฐานในการใส่คำอธิบายชนิดของข้อมูลมัลติมีเดียในตัวของมันเอง มีประโยชน์ในด้านการหาข้อมูลบนระบบอินเทอร์เน็ต

MPEG-21 เป็นมาตรฐานใหม่ล่าสุดที่เพิ่งกำเนิดขึ้น ว่าด้วยเรื่องเกี่ยวกับ Multimedia Framework

#### 2.2 เอ็มเป็ก 1 เลเยอร์ 3 (MP3)

MP3 ที่เรารู้จักกันจะอยู่ในมาตรฐาน MPEG-1 ใช้ในการเข้ารหัสข้อมูลภาพและเสียง แต่ตัว MPEG-1 นี้ยังแบ่งออกเป็น 3 เลเยอร์ (layer) ตามความสามารถและความซับซ้อนในการเข้ารหัสข้อมูล โดยเลเยอร์ 1 มีความซับซ้อนในการเข้ารหัสน้อยทำให้บีบอัดข้อมูลได้น้อย และในทางกลับกัน เลเยอร์ 3 มีความซับซ้อนในการเข้ารหัสมากที่สุดทำให้บีบอัดข้อมูลได้มากที่สุด แต่ทั้งสามเลเยอร์จะมีพื้นฐานในการบีบอัดเหมือนกันทั้งหมด

MP3 นี้คือการเข้ารหัสข้อมูล MPEG-1 Layer 3 มีกระบวนการในการเข้ารหัสที่ซับซ้อนที่สุด ทำให้บีบอัดข้อมูลได้มาก หรือทำให้เหลือขนาดของข้อมูลเก็บมาก แต่ขนาดข้อมูลที่เสิร์ฟใช้ว่าจะไม่เสียอะไรไปเลย เนื่องจากการบีบอัดข้อมูลแบบ MPEG นั้น ได้ผลลัพธ์เป็นข้อมูลที่เรียงต่อกัน (Data Stream) ซึ่งจะรองรับทั้งในแบบ 1 ช่องสัญญาณ และ 2 ช่องสัญญาณ แยกออกเป็นระบบเสียงต่างๆ ได้ถึง 4 ระบบ คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1 ระบบโมนโอ คือ ได้ผลลัพธ์เป็นข้อมูลเพียง 1 ช่องสัญญาณ ซึ่งอาจจะเป็นเสียงจากช่องซ้ายหรือขวาก็ได้

2.2.2 ระบบดูอัลโมนโอ (Dual-Mono) ได้ข้อมูลผลลัพธ์ออกมา 2 ช่องสัญญาณ โดยช่องหนึ่งเป็นเสียงจากข้อมูลฝั่งซ้าย อีกช่องเป็นเสียงจากลำโพงฝั่งขวา

2.2.3 ระบบสเตอริโอ ข้อมูลที่ได้ประกอบด้วย 2 ช่องสัญญาณเช่นกัน แต่ช่องหนึ่งจะเป็นผลรวมของเสียงลำโพงซ้ายกับลำโพงขวา อีกช่องหนึ่งเป็นผลต่างของลำโพงซ้ายกับลำโพงขวา

2.2.4 ระบบจอยท์-สเตอริโอ (Joint-Stereo) มีลักษณะคล้ายกับระบบสเตอริโอ แต่จะมีการรวมสัญญาณที่ความถี่ต่ำ ๆ ไว้ในช่องสัญญาณเดียว และแยกเสียงที่ความถี่สูงขึ้นมาเหมือนกับระบบสเตอริโอปกติทั่วไป เนื่องจากธรรมชาติของหูมนุษย์จะแยกแยะตำแหน่งของแหล่งกำเนิดเสียงได้ยากถ้าเสียงนั้นมีความถี่ต่ำ

มาตรฐานในการบีบอัด	อัตราส่วน	ความเร็วในการส่งข้อมูล
MPEG-1 Layer 1	1 : 4	384 กิโลบิต/s
MPEG-1 Layer 2	1 : 6 ถึง 1 : 8	256 – 192 กิโลบิต/s
MPEG-1 Layer 3	1 : 10 ถึง 1 : 12	128 – 115 กิโลบิต/s

ตารางที่ 2.1 อัตราการบีบอัดข้อมูลและความเร็วในการส่งข้อมูลจากเครื่องอ่านของข้อมูลที่ถูกบีบอัดตามมาตรฐาน MPEG-1

จากข้างต้น จะเห็นว่าข้อมูลที่อ่านได้จากซีดีรอมเพลงธรรมดาที่มีจำนวนถึง 1,411,200 บิต ต่อความยาว 1 วินาที ดังนั้นความเร็วในการส่งข้อมูลจากเครื่องอ่าน ไปยังตัวแปลงข้อมูลต้องสูงตามไปด้วย คือประมาณ 1.4 เมกะบิตต่อวินาที จากตารางแสดงการบีบอัดข้อมูลแสดงให้เห็นถึงอัตราการบีบอัดข้อมูลและความเร็วในการส่งข้อมูล จากเครื่องอ่านของข้อมูลที่ถูกบีบอัดตามมาตรฐาน MPEG-1 เลเยอร์ต่างๆ โดยอ้างอิงเสียงที่ได้จากการแปลงกลับให้อยู่ระดับคุณภาพเสียงซีดี เห็นได้ว่าข้อมูลที่บีบอัดตามมาตรฐาน MP3 นั้นจะเหลืออัตราความเร็วในการส่งข้อมูลเพียง 128 กิโลบิตต่อวินาที หมายความว่าข้อมูลเสียงเพลง 1 นาที จากเดิมที่มีขนาดประมาณ 10 เมกะไบต์จะเหลือเพียงประมาณ 1 เมกะไบต์เท่านั้น นั่นหมายถึงแผ่นซีดีรอมจากเดิมที่เก็บเพลงได้ 14 – 18 เพลง รวมความยาวประมาณ 65 นาที ถ้านำมาเก็บข้อมูลที่ถูกเข้ารหัส MP3 จะสามารถเก็บเพลงได้มากกว่า 200 เพลง รวมความยาวมากกว่า 600 นาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คุณภาพเสียง	แบนด์วิธ	ระบบเสียง	อัตราการส่งข้อมูล	อัตราส่วนการบีบอัดข้อมูล
โทรศัพท์	2.5 กิโลเฮิร์ตซ์	โมโน	8 กิโลบิต/s	96 : 1
ดีกว่าคลื่น Shortwave	4.5 กิโลเฮิร์ตซ์	โมโน	16 กิโลบิต/s	48 : 1
ดีกว่าคลื่นวิทยุ AM	7.5 กิโลเฮิร์ตซ์	โมโน	32 กิโลบิต/s	24 : 1
ใกล้เคียงวิทยุ FM	11 กิโลเฮิร์ตซ์	สเตอริโอ	56 - 64 กิโลบิต/s	26 - 24 : 1
ใกล้เคียงวิทยุ CD	15 กิโลเฮิร์ตซ์	สเตอริโอ	96 กิโลบิต/s	16 : 1
CD	> 15 กิโลเฮิร์ตซ์	สเตอริโอ	112-128 กิโลบิต/s	14 - 12 : 1

ตารางที่ 2.2 ความสัมพันธ์ระหว่างคุณภาพเสียงที่ต้องการกับขนาดของข้อมูลที่ถูกบีบอัด

เลขอร์	อัตราส่วนการบีบอัด	เวลาที่ใช้ในการถอดรหัส
1	4 : 1	19 ns
2	6 : 1	35 ns
3	12 : 1	59 ns

ตารางที่ 2.3 เวลาที่เสียไปในการแปลงข้อมูลเอ็มเป็ก

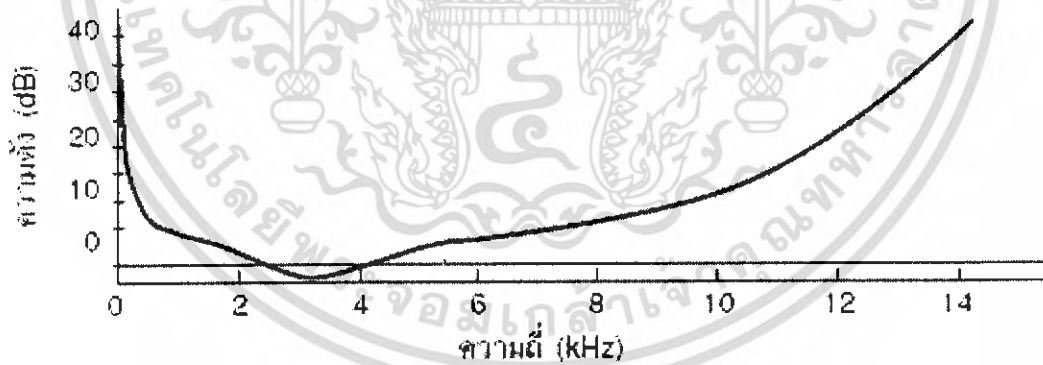
### 2.3 คุณภาพเสียงที่ได้

การบีบอัดข้อมูลตามมาตรฐาน MPEG นั้น สามารถกำหนดได้ว่าต้องการให้ข้อมูลที่ได้นั้นมีคุณภาพเสียงในระดับใด ถ้าต้องการให้มีคุณภาพเสียงที่ดีก็จะมีขนาดของข้อมูลใหญ่กว่าข้อมูลที่ต้องการคุณภาพเสียงที่ด้อยลงมา ขนาดของข้อมูลที่ถูกบีบอัดสามารถแทนได้ด้วยอัตราเร็วในการส่งข้อมูล เนื่องจากถ้าอัตราเร็วในการส่งข้อมูลสูง หมายความว่า ใน 1 วินาที ต้องใช้ข้อมูลในจำนวนที่สูงตามไปด้วย

ถ้าหากมาดูกันถึงความสัมพันธ์ระหว่างคุณภาพเสียงที่ต้องการ กับขนาดของข้อมูลที่ถูกบีบอัดจะเห็นความสามารถในการบีบอัดแบบ MP3 ได้อย่างชัดเจน ในกรณีที่เราต้องการคุณภาพเสียงในระดับของโทรศัพท์ จะเหลืออัตราการส่งข้อมูลเพียง 8 กิโลบิตต่อวินาทีเท่านั้น (ปัจจุบันระบบ

โทรศัพท์ที่ดิจิตอลส่งข้อมูลเสียงด้วยความเร็ว 64 กิโลบิตต่อวินาที ตามมาตรฐาน (ISDN) หรือถ้าหากเราต้องการเสียงที่มีคุณภาพในระบบวิทยุ FM โดยใช้การบีบอัดแบบ MP3 ก็จะใช้อัตราการส่งข้อมูลเพียง 64 กิโลบิตต่อวินาทีเท่านั้น ด้วยความเร็วเท่านี้ ทำให้เราสามารถสร้างระบบวิทยุดิจิตอลซึ่งส่งข้อมูลดิจิตอลออกอากาศให้เป็นความจริงได้ แต่ความสามารถในการบีบอัดที่สูงก็ใช่ว่าจะได้อะไรโดยไม่เสียอะไรเลย เนื่องจากกระบวนการในการเข้ารหัสที่ซับซ้อนทำให้การแปลงข้อมูลต้องใช้เวลาช่วงหนึ่งในการทำงาน ดังเห็นได้จากตารางที่ 2.3

การลดขนาดข้อมูลตามมาตรฐาน MPEG นั้น ใช้พฤติกรรมในการได้ยินเสียงของมนุษย์มาเป็นเครื่องมือในการลดขนาดข้อมูล อย่างที่เรารู้กันโดยทั่วไปว่า หูของมนุษย์นั้นมีขีดจำกัดในด้าน การรับฟัง โดยขีดจำกัดที่อยู่ด้านความถี่ 20 – 20,000 เฮิร์ตซ์ นั้นหมายความว่ามนุษย์จะได้ยินเสียงในช่วงความถี่สูงกว่า 20 เฮิร์ตซ์ แต่ต่ำกว่า 20 กิโลเฮิร์ตซ์เท่านั้น ถ้าอยู่เกินช่วงนี้ไปหูเราจะไม่ได้ยินเสียงนั้น ๆ แต่ใช้ว่าหูเราจะมีไหวต่อทุก ๆ ความถี่ที่เราได้ยินเท่า ๆ กัน จากผลการทดลองของนักวิทยาศาสตร์โดยสร้างห้องปิดที่ไม่มีเสียงรบกวนจากภายนอกขึ้น ให้ผู้ทดลองเข้าไปข้างในห้อง จากนั้นก็เริ่มสร้างเสียงที่ความถี่เท่าหนึ่ง (ที่มนุษย์ได้ยิน) โดยค่อย ๆ เพิ่มความดังของเสียงขึ้นเรื่อย ๆ จนครบตลอดย่านความถี่ที่มนุษย์ได้ยินนำข้อมูลที่ได้อาวาตกราฟจะได้ดังรูป

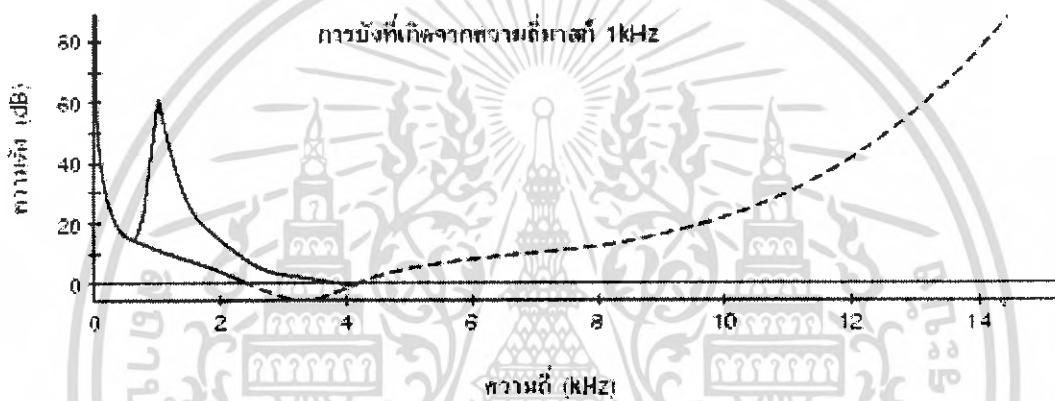


รูปที่ 2.1 ความไวต่อเสียงความถี่ต่าง ๆ ของมนุษย์

จากรูปที่ 2.1 จะเห็นได้อย่างชัดเจนว่า หูของมนุษย์เรามีความไวต่อเสียงแตกต่างกันที่ความถี่ต่าง ๆ กัน โดยจะมีความไวมากที่สุดในช่วงความถี่ 2 – 4 กิโลเฮิร์ตซ์ และถ้าเป็นความถี่ที่ต่ำมาก ๆ หรือสูงมาก ๆ ก็ยังต้องใช้ความดังมาก ๆ เพื่อให้เราได้ยิน การทดลองในช่วงที่ผ่านมาเป็นเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดลองกับความถี่เพียงความถี่เดียว ต่อไปเราจะมาดูว่าถ้ามีเสียง 2 ความถี่ที่ใกล้เคียงกันกำเนิดพร้อม ๆ กัน เราจะได้ยินอย่างไร

ในห้องปิดห้องเดิมสร้างเสียงความถี่ 1 กิโลเฮิร์ตซ์ที่ระดับความดัง 60 dB จากนั้นก็สร้างเสียงที่มีความถี่ใกล้เคียงกันขึ้นมา เช่น 0.9 กิโลเฮิร์ตซ์ แล้วเพิ่มความดังขึ้นเรื่อย ๆ จนกว่าหูจะได้ยินแล้วบันทึกค่าไว้ แล้วปรับความถี่ที่สร้างขึ้นเป็นความถี่ที่ 2 ทดลองไปเรื่อย ๆ แล้วบันทึกค่า นำมาวาดกราฟจะได้ดังรูปที่ 2.2

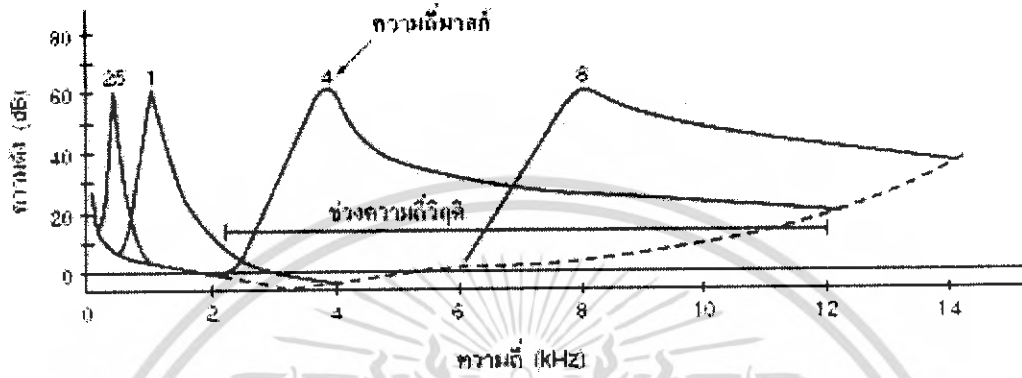


รูปที่ 2.2 การบังทางค่านความถี่ (Frequency Masking) ที่ความถี่มาสก์ 1 กิโลเฮิร์ตซ์

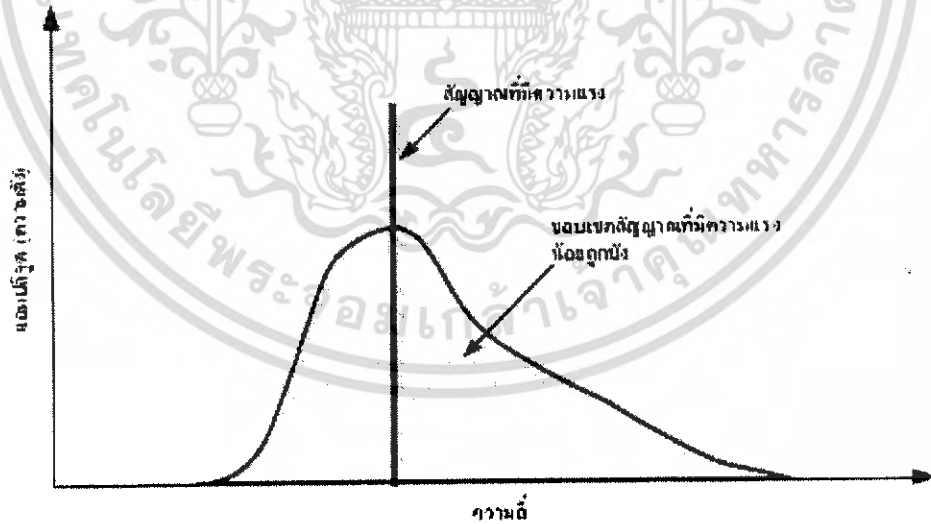
จากรูปที่ 2.2 เห็นได้ว่ามีความถี่หนึ่งดังขึ้นมา เสียงความถี่ที่ใกล้เคียงที่จะทำให้เราได้ยินนั้น จำเป็นต้องมีความดังมาก ๆ หากมีความดังน้อยกว่าเส้นกราฟหมายความว่าเราจะไม่ได้ยินเสียงนั้น ๆ ไปเลย คุณสมบัติของหูมนุษย์ในจุดนี้ทำให้เกิดย่านความถี่วิกฤต (Critical Band) ถ้ามองจากรูปก็คือช่วงความถี่ที่อยู่ในส่วนฐานของสามเหลี่ยม และเราเรียกความถี่ที่เป็นยอดของสามเหลี่ยมนี้ว่าความถี่มาสก์ (Masking Tone) สรุปแล้วย่านความถี่วิกฤตก็คือย่านความถี่ในช่วงที่ได้ยินหรือแยกแยะเสียงได้ยาก ถ้ามีเสียงความถี่มาสก์ของแบนด์นั้น ๆ อยู่ เราเรียกคุณสมบัติข้อนี้ของหูมนุษย์ว่าการบังทางความถี่ (Frequency Masking) นอกจากการไม่ได้ยินเสียงที่อยู่ในย่านความถี่วิกฤตแล้วยังมีคุณสมบัติอีกข้อหนึ่งที่ทำให้หูเราไม่ได้ยินเสียงไปชั่วคราว เราเรียกคุณสมบัติข้อนี้ว่า การบังชั่วคราว (Temporal Masking) คือเมื่อเราได้ยินเสียงที่เป็นความถี่มาสก์ดังขึ้นมาในระดับหนึ่งหลังจากเสียงนั้นหยุดลง เราจะต้องเสียเวลาช่วงหนึ่งก่อนที่จะได้ยินเสียงที่มีความถี่ใกล้เคียงกับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความถี่มาศกันั้น ดังรูปที่ 2.3



รูปที่ 2.3 ย่านความถี่วิกฤติ (Critical Band) ที่ความถี่ต่าง



รูปที่ 2.4 ขอบเขตของช่วงความถี่ที่ถูกบังในการใช้ไซโคอะคูสติก (Psychacoustic)

เป็นกราฟที่ได้จากการทดลองสร้างเสียงความถี่มาศกั 1 กิโลเฮิร์ตซ์ ที่มีความดัง 60 dB หลังจากนั้นปิดเสียงที่เป็นความถี่มาศกัทดลองสร้างที่ใกล้เคียง (ในที่นี้คือ 1.1 กิโลเฮิร์ตซ์) ที่เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความดังค่าหนึ่งขึ้นมาจับเวลาที่เสียงไปก่อนจะได้ยินเสียงนั้น แล้วทดลองซ้ำโดยเปลี่ยนระดับความดังของเสียงที่เกิดขึ้นทีหลังไปเรื่อย ๆ นำข้อมูลทั้งหมดมาวาดกราฟ ซึ่งจะเห็นได้ว่ายิ่งเสียงที่เกิดขึ้นมีความดังน้อยเท่าไร ก็ยังมีระยะที่เราไม่ได้ยินเสียงยาวขึ้นเท่านั้น

คุณสมบัติด้านการฟังทั้งหมดที่กล่าวมานั้น เรียกว่าไซโคอคูสติกโมเดล (Psychoacoustic Model) ถือเป็นเครื่องมือสำคัญในการลดขนาดข้อมูลของการบีบอัดตามมาตรฐาน MPEG ซึ่งมีลำดับกระบวนการในการบีบอัดดังนี้

1. นำข้อมูลเสียงดิจิทัลอลป้อนเข้าฟิลเตอร์เพื่อแยกเสียงออกเป็นช่วงความถี่ย่อยๆ (Subbands) ซึ่งมีความกว้างเท่ากับย่านความถี่วิกฤต จำนวน 32 ช่วงความถี่ เรียงขั้นตอนนี้ว่า Sub-band Filtering

2. ใช้ไซโคอคูสติกโมเดลเป็นเครื่องมือในการวิเคราะห์ข้อมูล ส่วนที่ไม่มีผลต่อการได้ยินของมนุษย์ออกไป โดยพิจารณาระหว่างช่วงความถี่ 2 ช่วงที่ติดกันและพิจารณาย่อยลงไปในแต่ละช่วงความถี่ด้วย

3. ถ้าวิเคราะห์แล้วพบว่าเสียงช่วงใดไม่มีผลต่อการได้ยิน ให้ตัดข้อมูลส่วนนั้นออกไป ไม่นำไปเข้ารหัสในส่วนถัดไป

4. นำข้อมูลที่เหลือมาเข้ารหัสซึ่งจะมีวิธีที่แตกต่างกันขึ้นอยู่กับแต่ละเลขอร์

#### 2.4 โครงสร้างของข้อมูลไฟล์เอ็มเป็ก 1 เลขอร์ 3

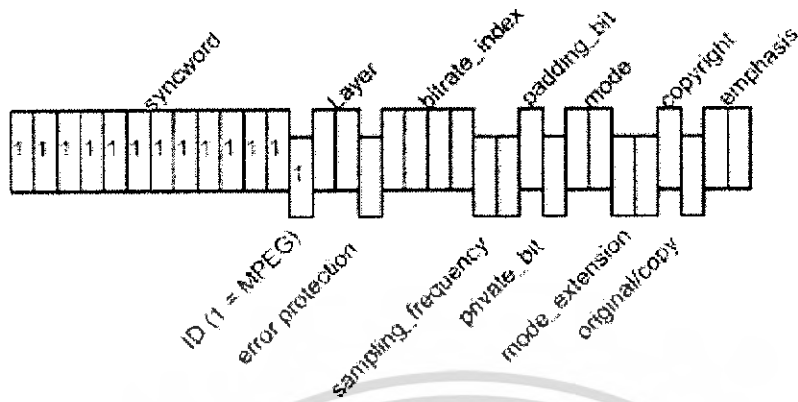
ข้อมูลที่ถูกรีบอัดตามมาตรฐาน MP3 นั้นจะอยู่ในลักษณะของเฟรมข้อมูล โดยในแต่ละเฟรมข้อมูลจะมีส่วนประกอบภายในอยู่ 4 ส่วน คือ

1. หัวข้อมูล (Header) เป็นข้อมูลขนาด 32 บิต แสดงลักษณะทั่วไปของไฟล์นั้น ๆ
2. ส่วนตรวจสอบความผิดพลาด (CRC) เป็นข้อมูลขนาด 16 บิต ใช้ตรวจสอบข้อมูลภายในเฟรมว่าถูกต้องหรือไม่จะมีหรือไม่ก็ได้
3. ข้อมูลข้างเคียง (Side Information) มีขนาด 17 หรือ 32 ไบต์ (17 ไบต์สำหรับระบบโมโน 32 ไบต์ สำหรับระบบอื่น ๆ) เป็นส่วนที่เก็บองค์ประกอบที่ใช้ในการถอดรหัส
4. ข้อมูลหลัก (Main Data) มีความยาวขึ้นอยู่กับอัตราการส่งข้อมูล (Baud rate) และอัตราการสุ่มข้อมูลในการแปลงกลับเป็นสัญญาณอะนาลอก (Sampling Frequency)

หัวข้อมูล (Header)	ตรวจสอบความผิดพลาด (CRC)	ข้อมูลข้างเคียง (Side Information)	ข้อมูลหลัก (Main Data)
-----------------------	-----------------------------	---------------------------------------	---------------------------

#### ตารางที่ 2.4 รูปแบบข้อมูลเอ็มเป็ก 1 เลขอร์ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 ส่วนประกอบภายในหัวข้อมูล (Header)

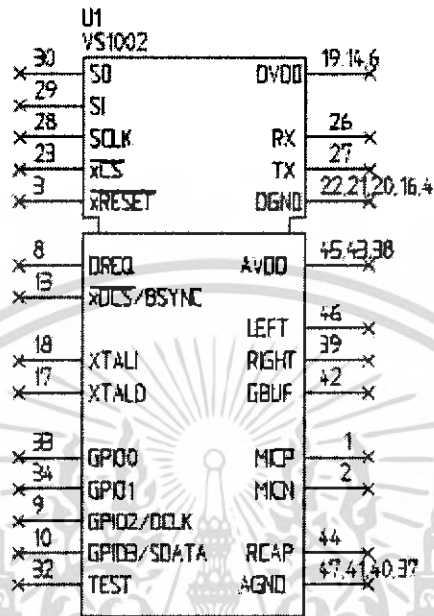
## 2.5 ชิพถอดรหัสข้อมูลเอ็มเป็ก VS1002d

### 2.5.1 คุณสมบัติของ VS1002d

1. สามารถถอดรหัสข้อมูลเอ็มเป็กออกดีโอได้ทั้งเลขอร์ 1, 2 และ 3
2. สนับสนุนการถอดรหัสข้อมูลเอ็มเป็ก 1 และ 2 ทั้งทุกเลขอร์ในเอ็มเป็ก 3 สามารถถอดรหัสได้ถึงเลขอร์ 2.5 โดยสนับสนุนข้อมูลทั้งแบบโมโนและสเตริโอ
3. สามารถส่งข้อมูลด้วยความเร็วได้หลายระดับ
4. ทำงานที่สัญญาณนาฬิกา 12.288 – 16 เมกะเฮิร์ตซ์ หรือ 24.576 – 26 เมกะเฮิร์ตซ์ (สำหรับความเร็วในการส่งค่า)
5. ประหยัดพลังงาน
6. ในชิพประกอบด้วย ดิจิตอลอะนาลอกคอนเวอร์เตอร์ (DAC) คุณภาพสูงโดยปราศจากเพสเอรอร์ระหว่างแชลแนล
7. สำหรับสัญญาณอะนาลอกทำงานด้วยระดับแรงดัน 2.6 – 3.6 V
8. สำหรับสัญญาณดิจิตอลทำงานด้วยระดับแรงดัน 2.1 – 3.6 V
9. มีแรมบนชิพถึง 4 กิโลบิต สำหรับผู้ใช้
10. มีฟังก์ชันใหม่เพิ่มเติม เช่น ข้อมูลเข้าเป็น PCM ข้อมูลเข้าเป็นสตรีม (Streaming)
11. ประมวลผลใน 16 บิตเวิร์สของข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.5.2 ข้อมูลของขาใน VS1002d



รูปที่ 2.6 ขาต่าง ๆ ของชิพ VS1002d

ชื่อขา	ขาที่	ชนิดของขา	หน้าที่ของขา
DREQ	1	DO	ขาแสดงร่องขอข้อมูล
DCIK	2	DIO	ขาสัญญาณนาฬิกาสำหรับข้อมูลอินพุตแบบอนุกรม
SDATA	3	DI	ขาข้อมูลอินพุตแบบอนุกรม
BSYNC	4	DI	ขาสัญญาณแสดงข้อมูลซิงโครไนส์
DVDD1	5	PWR	ขาไฟเลี้ยงดิจิทัล
DGND1	6	PWR	ขาราวด์ดิจิทัล
XTALO	7	CLK	ขาคริสตอลเอาต์พุต
XTALI	8	CLK	ขาคริสตอลอินพุต
DVDD2	9	PWR	ขาไฟเลี้ยงดิจิทัล
DGND2	10	PWR	ขาราวด์ดิจิทัล
XCS	11	DI	ขาเลือกอินพุตข้อมูลว่าเป็นข้อมูลหรือคอนโทรล (active low)
SCLK	12	DI	ขาสัญญาณนาฬิกาสำหรับข้อมูลอินพุตแบบอนุกรม
SI	13	DI	ขาอินพุตอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อขา	ขาที่	ชนิดของขา	หน้าที่ของขา
SO	14	DO3	ขาเอาต์พุตอนุกรม
TEST0	15	DI	ขาสำรองไว้สำหรับทดสอบ ต่อเข้ากับ DVDD
TEST1	16	DIO	ขาสำรองไว้สำหรับทดสอบ ไม่ต้องต่อ
TEST2	17	DIO	ขาสำรองไว้สำหรับทดสอบ ไม่ต้องต่อ
AGND1	18	PWR	ขากราวด์อะนาล็อก
AVDD1	19	PWR	ขาไฟเลี้ยงอะนาล็อก
RIGHT	20	AO	ขาเอาต์พุตเซนแนลขวา
AGND2	21	PWR	ขากราวด์อะนาล็อก
RCAP	22	AIO	ขาเปรียบเทียบความจุไฟฟ้า
AVDD2	23	PWR	ขาไฟเลี้ยงอะนาล็อก
LEFT	24	AO	ขาเอาต์พุตเซนแนลซ้าย
AGND3	25	PWR	ขากราวด์อะนาล็อก
XRESET	26	DI	ขารีเซ็ตแบบอะซิงโครไนส์ ทำงานที่ระดับสัญญาณ "0"
DGND3	27	PWR	ขากราวด์ดิจิตอล
DVDD3	28	PWR	ขาไฟเลี้ยงดิจิตอล

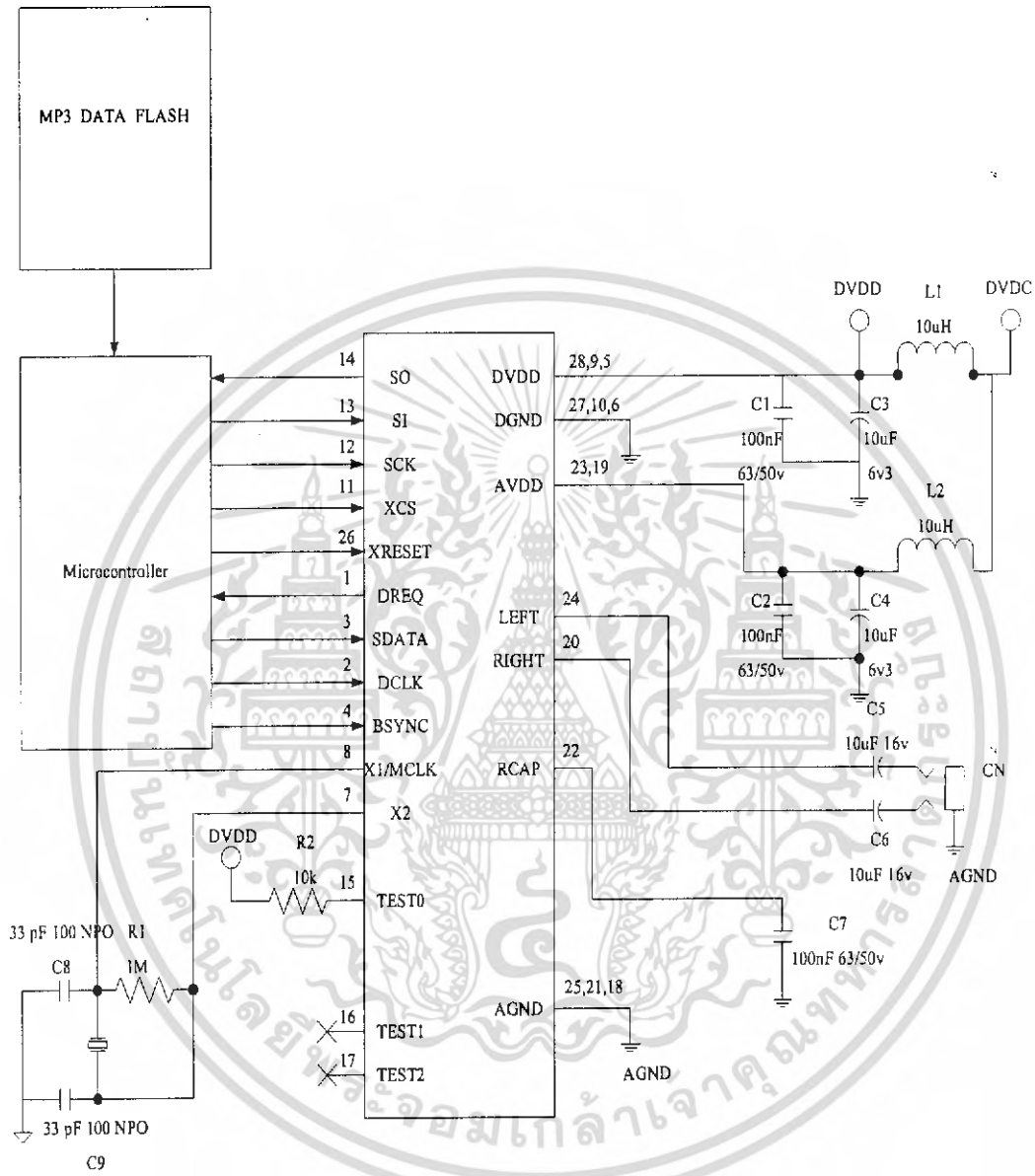
ตารางที่ 2.5 หน้าที่ของขาในชิพ VS1002d

ความหมายของชนิดของขา

- DI ขาดิจิตอลอินพุต
- DO ขาดิจิตอลเอาต์พุต
- DIO ขาดิจิตอลอินพุต/เอาต์พุต
- AO ขอนาฬิกาเอาต์พุต
- CLK ขาสัญญาณนาฬิกา/ขาแสดงการต่อกับคริสตอล
- PWR ขาไฟเลี้ยงหรือกราวด์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.5.3 การนำชิพ VS1002d ไปต่อใช้งาน



รูปที่ 2.7 การนำชิพ VS1002d ไปต่อใช้งานกับอุปกรณ์ไมโครคอนโทรลเลอร์

### 2.6 ส่วนที่เชื่อมต่อข้อมูลอนุกรม SPI ของชิพ VS1002d สำหรับดาต้า (SDI: Serial Protocol for Serial Data Interfacd)

สามารถแบ่งได้เป็น 2 แบบ คือ

1. ส่วนเชื่อมต่อข้อมูลแบบอนุกรมสำหรับส่งข้อมูล
2. ส่วนเชื่อมต่อข้อมูลอนุกรมสำหรับการควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.6.1 หน้าที่สำหรับขาส่งข้อมูลอนุกรม SPI

ขา SDI	ขา SCI	หน้าที่
-	XCS	ถ้าขา XCS ถูกป้อนเข้าด้วยระดับสัญญาณ “0” เป็นการเลือกว่าข้อมูลที่ส่งเป็นเอ็มเป็ก ถ้าขา XCS ถูกป้อนเข้าด้วยระดับสัญญาณ “1” เป็นการเลือกว่าข้อมูลที่ส่งเป็นข้อมูลสำหรับควบคุม
DCLK	SCK	สัญญาณนาฬิกาของข้อมูลอนุกรม จะถูกใช้เหมือนเป็นสัญญาณนาฬิกาหลักสำหรับอุปกรณ์อินเตอร์เฟส ในกรณีที่ XCS เป็น “0” ที่ขอบขาขึ้นแรกจะทำให้บิตแรกถูกเขียนลงไป ในขณะที่สัญญาณนาฬิกาที่กำลังให้สัญญาณต่อเนื่องไปเรื่อย ๆ
SDATA	SI	ข้อมูลอินพุตแบบอนุกรม ในกรณีที่ XCS มีค่าเป็น “0” SI จะเริ่มส่งเอาข้อมูลที่ขอบขาขึ้นของสัญญาณ SCK
-	SO	ข้อมูลเอาต์พุตแบบอนุกรมในการอ่าน ข้อมูลจะถูกเลื่อนออกที่ขอบขาของสัญญาณ SCK ที่ขอบขาลง ในการเขียน SO จะอยู่ในสถานะ high impedance

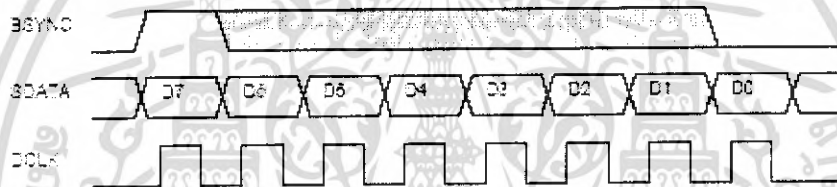
ตารางที่ 2.6 หน้าที่ของพอร์ตอนุกรม SPI

### 2.6.2 คุณลักษณะสำหรับการส่งแบบอนุกรมสำหรับการส่งข้อมูล (SDI)

สำหรับการส่งข้อมูลแบบอนุกรมจะสามารถทำงานได้ 2 โหมด คือ มาสเตอร์และสเลฟ ในการทำงานโหมดมาสเตอร์ VS1002d จะทำการสร้างสัญญาณนาฬิกา DCLK ขึ้นมาเอง โดยจะสามารถเลือกระดับความเร็วของสัญญาณนาฬิกาได้ 2 ระดับ คือ 512 และ 1024 เมกะเฮิรตซ์ ในโหมดสเลฟสัญญาณนาฬิกา DCLK จะถูกสร้างจากวงจรภายนอก สำหรับข้อมูล (SDATA signal) จะสามารถเปลี่ยนแปลงได้ทั้งที่ขอบขาขึ้นหรือขอบขาลงของ DCLK ได้ตามการโปรแกรมของเรา ชิพ VS1002d จะสมมติเอาข้อมูลอินพุตของไบต์ซิงโครไนส์ เช่น การทำงานภายในตัวดีโคเดอร์ ภายในชิพจะไม่หาไบต์ซิงโครไนส์สำหรับเฟรมจากข้อมูลที่เข้ามาเรื่อย ๆ แต่จะสมมติเอาข้อมูลที่เรียงเข้ามาอย่างถูกต้องเท่านั้น โดยไบต์ข้อมูลจะถูกส่งมาได้ทั้งแบบ MSB หรือ LSB ขึ้นอยู่กับการเซตในโหมด

เพื่อความแน่ใจว่าข้อมูลที่เข้ามาเป็นไบต์จะถูกต้องจริงๆ การส่งข้อมูลแบบอนุกรมจะมีสัญญาณ BSYNC เข้ามา ระหว่างที่สัญญาณ BSYNC เป็น “1” สัญญาณนาฬิกา จะให้กำเนิดพัลส์แรกออกมาและสัญญาณ BSYNC จะตกลงเป็น “0” ก่อนที่ข้อมูลจะส่งครบ ถ้าสัญญาณ BSYNC ไม่ใช่เราสามารถต่อเข้ากับไฟเลี้ยงภายนอก และอุปกรณ์ที่จะส่งข้อมูลเข้าจะต้องส่งได้อย่างถูกต้อง

สัญญาณ DREQ จะเป็นสัญญาณที่แสดงการร้องขอข้อมูลของ VS1002d ถ้า DREQ เป็น “0” ตัวส่งควรจะหยุดการส่งข้อมูลใหม่เพราะว่าจะเหลือเนื้อที่ 32 ไบต์เป็นเนื้อที่ปลอดภัย ซึ่งจะง่ายสำหรับพวกอุปกรณ์ไมโครคอนโทรลเลอร์ที่มีความเร็วต่ำ ถ้าหากไม่มีการตรวจสอบสัญญาณ DREQ ตัวส่งอาจส่งมากเกินไป 32 ไบต์ ทำให้อาจไม่เกิดการดีโค้ดของข้อมูลได้



รูปที่ 2.8 รูปสัญญาณ BSYNC

## 2.7 ส่วนเชื่อมต่อข้อมูลอนุกรม SPI ของชิพ VS1002d สำหรับคอนโทรล (SCI: Serial Protocol for Serial Command Interface)

โดยปกติ ชุดคำสั่งของการส่งข้อมูลอนุกรมสำหรับคอนโทรลจะประกอบด้วยชุด คำสั่งแอดเดรสไบต์ และ 16 บิต คำคำสั่ง (instructions + address byte + 16 bit data word) โดยชุดคำสั่ง (instruction) จะเป็นข้อมูล 8 บิต ซึ่งเป็นคำสั่งสำหรับอ่านและเขียนดังตาราง

ชุดคำสั่ง (Instruction)		
ชนิด	ออปโค้ด (Opcode)	ลักษณะการทำงาน
READ	0000 0011	Read Data
WRITE	0000 0010	Write Data

ตารางที่ 2.7 ชุดคำสั่งของชิพ VS1002d

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

\*\*\* หลังจากได้ส่งคำสั่งควบคุมแล้วตัวชิพ VS1001h จะไม่ยอมให้ส่ง SCI หรือ SDI ได้อีกเป็นระยะเวลา 5 ไมโครวินาที

### 2.7.1 การอ่านข้อมูลสำหรับ SCI (SCI Read)

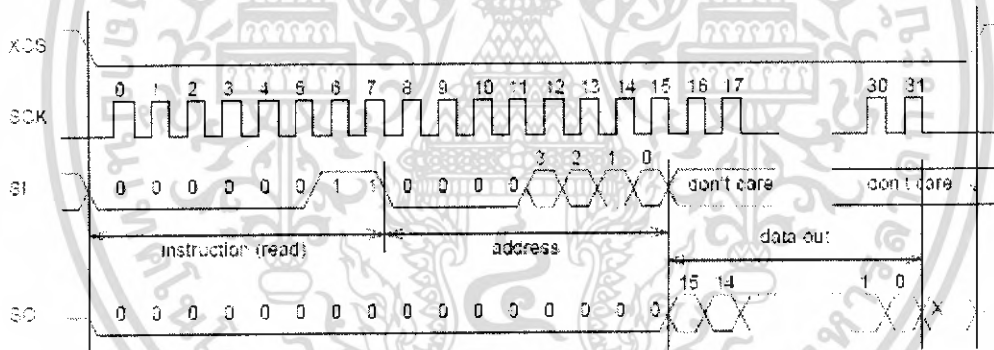
การอ่านข้อมูลจากชิพ VS1002d สามารถทำได้ดังนี้

1. ให้สัญญาณ XCS เป็น “0” เพื่อเลือกชิพให้ทำงานที่ SCI
2. จากนั้นส่ง ออปโค้ดอ่าน (Read Opcode) 03H ผ่านสาย SI จากนั้นส่ง 8 บิต

แอดเดรสตามลำดับ

3. หลังจากส่งแอดเดรสไปแล้ว หากมีข้อมูลใดส่งไปตามหลังชิพ VS1002d จะไม่สนใจ แต่จะส่งข้อมูล 16 บิตที่อยู่ในแอดเดรสที่ต้องการ โดยชิพ (shifted) ออกไปทางสาย SO

4. ให้สัญญาณ XCS เป็น “1” หลังจากชิพ (shifted) ข้อมูลเรียบร้อยแล้ว เพื่อให้จบกระบวนการอ่านข้อมูล



รูปที่ 2.9 สัญญาณการอ่านข้อมูลผ่านพอร์ตอนุกรม

### 2.7.2 การเขียนข้อมูลสำหรับ SCI (SCI Write)

การเขียนข้อมูลให้แก่ชิพ VS1002d สามารถทำได้ดังนี้

1. ให้สัญญาณ XCS เป็น “0” เพื่อเลือกชิพให้ทำงานที่ SCI
2. จากนั้นส่ง ออปโค้ดเขียน (Write Opcode) 02H ผ่านสาย SI จากนั้นส่ง 8 บิต

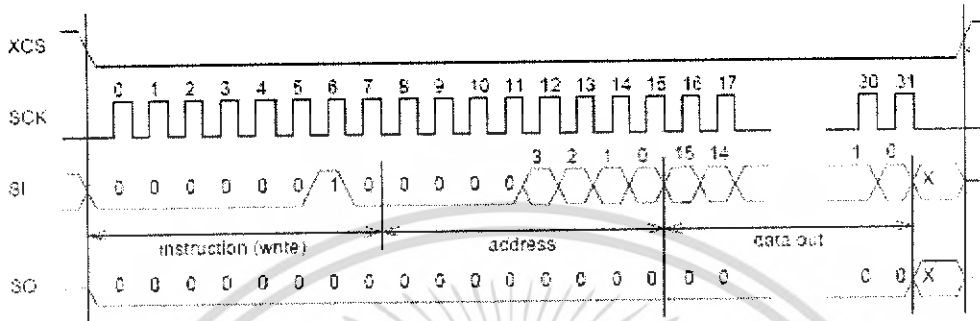
แอดเดรสตามลำดับ

3. จากนั้นจึงส่งข้อมูลตามไป ข้อมูลจะถูกชิพ (Shifted) ออกไปตามสาย SI

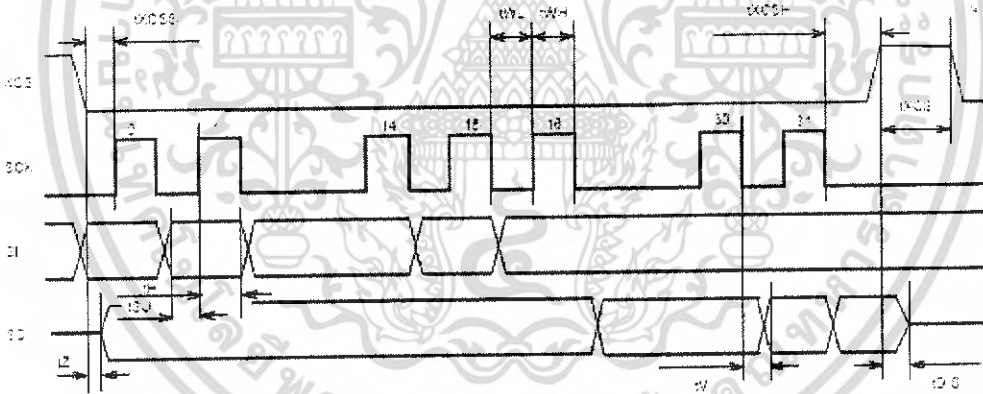
4. ให้สัญญาณ XCS เป็น “1” เพื่อจบกระบวนการเขียนข้อมูลลง SCI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. ระดับสัญญาณ XCS เปลี่ยนจาก “0” ไป “1” จะเกิดหลังจากสัญญาณ SCLK เปลี่ยนจาก “1” ไป “0” การเปลี่ยนแปลงจะเกิดตรงกับบิต LSB ของข้อมูล



รูปที่ 2.10 สัญญาณการเขียนข้อมูลลงพอร์ตอนุกรม



รูปที่ 2.11 แผนผังเวลาของพอร์ตอนุกรม

### 2.7.3 ลักษณะการทำงาน

VS1002d ใช้คุณสมบัติพื้นฐานการประมวลผลสัญญาณดิจิทัล VS\_DPS ภายในบรรจุกิตและหน่วยความจำที่จำเป็นสำหรับการถอดรหัสเอ็มเป็กทั้งหมด พร้อมด้วยพอร์ตอนุกรมกับตัวขยายและควบคุมเอาต์พุตที่เป็นสัญญาณอะนาลอก สเตริโอ DAC หลายระบบ

ข้อมูลและบิตเรตในส่วนเพิ่มเติมสามารถส่งข้อมูลด้วยความเร็วหลายระดับ(VBR)

เป็นตัวสนับสนุน ระดับ VBR ที่ใช้สำหรับเพลงจากใกล้เคียง VS1002d สามารถเล่นได้ทั้งไฟล์เอ็มเอกส์อาร์นเป็นเอกส์อาร์นที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกและเผยแพร่ข้อมูลนี้ไปยังผู้อื่นโดยไม่ได้รับอนุญาตจากสำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เป็ก 1 และ 2 เลขอร์ 1, 2 และ 3 ส่วนประกอบไฟล์ทั้ง อัตราสุ่ม CD นั้นประมาณ 100 kบิต/s สำหรับตัวอย่างเพลงระบบสเตอริโอที่ความถี่ 44100 Hz ด้วยเหตุที่การเข้ารหัสแบบเก่าต้องใช้ 128 kบิต/s สำหรับการทำงานเต็มที่ VBR การเข้ารหัสคุณภาพสูงได้ถูกนำมาใช้ประโยชน์อย่างกว้างขวาง

## 2.8 SCI รีจิสเตอร์

Name	Type	Addr	Function
โหมด	RW	0	โหมดคอนโทรล
STATUS	RW	1	สถานะของชิพ VS1002d
INT_FCTLH	-	2	รีจิสเตอร์ภายใน
CLOCKF	RW	3	ความเร็วสัญญาณนาฬิกา
SRATE	R	4	อัตราการสุ่มข้อมูล
SUDATA	R	5	ข้อมูลเสียงความดัง
WRAM	W	6	ส่วนการเขียนโปรแกรมลงหน่วยความจำ
WRAMADDR	W	7	แอดเดรสของหน่วยความจำที่ต้องการเขียน
HDAT0	R	8	จุดอ่านข้อมูล
HDAT1	R	9	จุดอ่านข้อมูล
AIADDR	RW	10	แอดเดรสที่ทำการเริ่มโปรแกรม
VOL	RW	11	ควบคุมความดังของเสียง
AICTRL[x]	RW	12	รีจิสเตอร์สำหรับการประยุกต์ใช้งานต่าง

ตารางที่ 2.7 SCI รีจิสเตอร์

### 2.8.1 รีจิสเตอร์โหมด, โหมด (RW)

บิต	Function	คำอธิบาย
โหมด[0]	ไม่ใช้งาน	
โหมด[1]	ไม่ใช้งาน	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บิต	Function		คำอธิบาย
โหมค[3]	ไม่ใช้งาน		
โหมค[4]	ประหยัดพลังงาน	0 1	ไม่มี มี
โหมค[2]	โปรแกรมรีเซต	0 1	รีเซต ไม่รีเซต
โหมค[5]	ไม่ใช้งาน		
โหมค[6]	ไม่ใช้งาน		
โหมค[7]	ไม่ใช้งาน		
โหมค[8]	สัญญาณ DCLK ทำงานที่ขอบขาขึ้น หรือขอบขาลง	0 1	ขอบขาขึ้น ขอบขาลง
โหมค[9]	ส่งบิตไหน ก่อนหน้าหรือหลัง	0 1	MSB ก่อน MSB หลัง
โหมค[10]	โหมคอินพุตข้อมูล	0 1	สเลฟ มาสเตอร์
โหมค[11]	ถ้าให้เป็นมาสเตอร์จะมีสัญญาณนาฬิกา เท่าใด	0 1	512 กิโลเฮิร์ตซ์ 1024 กิโลเฮิร์ตซ์

เครื่องจะรีเซต เมื่อตั้งค่า บิต 2 เป็น 1

#### ตารางที่ 2.8 รีจิสเตอร์โหมค

หน้าที่ของแต่ละบิตในรีจิสเตอร์โหมค มีดังนี้

บิต 4 จะครอบคลุมตัวอย่าง โดยจะเป็นตัวเปลี่ยน VS1002d เป็น โหมคประหยัดพลังงานที่เฉพาะส่วนการทำงาน คอนโทรลสายสัญญาณ

บิต 8 กำหนดขอบเขตการทำงานของ data clock สำหรับ SDI

บิต 9 กำหนด data order ภายในไบต์ สำหรับ SDI

บิต 10 ตั้งค่าสายสัญญาณอินพุตเป็นมาสเตอร์

บิต 11 ตั้งค่า bus clock speed เมื่อ VS1002d เป็นมาสเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.8.2 รีจิสเตอร์สถานะ, STATUS (RW)

รีจิสเตอร์สถานะบรรจุข้อมูลบน สถานะปัจจุบันของ VS1002d บิต 1 และ 0 ใช้ควบคุมอะนาล็อกเอาต์พุต ระดับเสียง 0 = -0 dB, 1 = -6 dB, 3 = -12 dB บิต 2 คืออะนาล็อก powerdown บิต เมื่อเซตเป็นอะนาล็อกเป็น powerdown

### 2.8.3 INT\_FCNTLH (-)

INT\_FCNTLH ไม่ใช่ user-accessible รีจิสเตอร์

### 2.8.4 รีจิสเตอร์สัญญาณนาฬิกา, CLOCKF (RW)

รีจิสเตอร์สัญญาณนาฬิกา สัญญาณนาฬิกาซึ่งทำงานบางอย่างด้วยความถี่มากกว่า 24.576 เมกะเฮิร์ตซ์ ความเร็วของสัญญาณนาฬิกาจะเซตในทุก ๆ 2 กิโลเฮิร์ตซ์ ดังนั้นสูตรการคำนวณค่าที่ถูกต้อง สำหรับรีจิสเตอร์นี้ คือ  $\text{ClockIn Hz} / 2000$  ค่านี้จะอยู่ระหว่าง 0.32767 ถึงแม้ว่าฮาร์ดแวร์จะมีขอบเขตความเร็วสูงสุด ที่ความเร็วต่ำกว่า 24.576 เมกะเฮิร์ตซ์ ทั้งอัตราสุ่มข้อมูลและบิตสตรีมจะทำงานไม่นานนัก ถ้ากำหนดให้ MSB ของ CLOCKF เป็น 1 จะทำงานภายใน clock-doubling clock ที่สูงถึง 15 เมกะเฮิร์ตซ์ อาจจะเป็น doubled ถ้าทำงานเป็น clock doubler 15 LSBs ของรีจิสเตอร์จะเป็นค่า clock-doubled ที่ใช้คำนวณ ดังนั้น รีจิสเตอร์นี้ต้องเซตค่าก่อนเริ่มการถอดรหัสข้อมูล MP3 มิฉะนั้นอัตราการสุ่มข้อมูลจะเซตไม่ถูกต้อง

ตัวอย่างที่ 1 : สำหรับความเร็วสัญญาณนาฬิกาที่ 26 เมกะเฮิร์ตซ์ ค่าจะเป็น 13000

ตัวอย่างที่ 2 : สำหรับความเร็วสัญญาณนาฬิกาภายนอกและใช้ clock-doubling

สำหรับ 27 เมกะเฮิร์ตซ์ ค่าจะเป็น  $0 \times 8000 + 13500 = 46263$

### 2.8.5 รีจิสเตอร์ SRATE (R)

เมื่อถอดรหัสข้อมูลถูกต้อง อডিโออัตราการสุ่มข้อมูลสามารถพบใน SRATE ไม่เป็นจำนวนเต็ม

### 2.8.6 รีจิสเตอร์ AUDATA (R)

เมื่อถอดรหัสข้อมูลถูกต้อง อัตราบิตเรตในปัจจุบันใน k บิต/s สามารถพบใน บิต/s 0.8 ของ AUDATA สำหรับบิตสตรีม variable บิตเรต บิต/s 14.9 จะไม่ใช่และจะเซตเป็น 0 บิต/s 15 เป็น 0 สำหรับข้อมูลแบบโมโนและ 1 สำหรับแบบสเตอริโอ

### 2.8.7 รีจิสเตอร์ WRAM (W)

WRAM ใช้โหลดโปรแกรมประยุกต์เป็นโปรแกรมแอดเดรสของแรมเริ่มต้นตั้งเป็นตัวแรกโดยการเขียนเป็น WRAMADDR รีจิสเตอร์ ลำดับถัดไปเป็นการเรียกครั้งแรกของ WRAM ค่าที่ใช้ 16 บิตs ของข้อมูลสามารถส่งได้กับการเขียน WRAM และโปรแกรมเวิร์สเป็น 32 บิตสองการเขียนที่ตามมาจำเป็นสำหรับแต่ละโปรแกรมเวิร์สไบต์ ลำดับคือบิตที่มีค่าสูงสุดก่อน

### 2.8.8 รีจิสเตอร์ WRAMDDR (W)

WRAMDDR ใช้เซตแอดเดรสของโปรแกรมสำหรับติดตามการเขียน WRAM พื้นที่สำหรับผู้เขียนโปรแกรมระหว่างแอดเดรส 4096 ... 5119 (ที่แอดเดรส 4096 ... 4111 จะถูกสำรองโดยระบบ) แต่สำหรับการเขียน WRAM ทั้งหมดจะสนใจแอดเดรสที่สูงกว่า 4096 ดังนั้นถ้าโปรแกรมเมอร์ต้องการเขียนที่แอดเดรส 4567 ต้องเขียนเป็น  $4567 + 4096 = 8663$  WRAMADDR

### 2.8.9 รีจิสเตอร์ HDATA and HDAT1 (R)

บิต	การทำงาน	ค่า	คำอธิบาย
HDAT [4 : 3]	ID	3	MPG 2.5 (1/4-rate)
		2	MPG 2.5 (1/4-rate)
		1	ISO 11172-3 1.0
		0	MPG 2.0 (1/2-rate)
HDAT [2 : 1]	เลขออร์	3	I
		2	II
		1	III
		0	สำรองไว้
HADT [0]	โปรเทกต์บิต	1	มีการโปรเทกต์ด้วย CRC
		0	ไม่มี CRC
HDAT0 [15 : 12]	บิตrate		c.f. 11172-3
HDAT0 [11 : 10]	อัตราสุ่มข้อมูล	3	สำรองไว้
		2	32/16/8 กิโลเฮิร์ตซ์
		1	48/24/12 กิโลเฮิร์ตซ์
		0	44/22/11 กิโลเฮิร์ตซ์

บิต	การทำงาน	ค่า	คำอธิบาย
HDATA0 [8]	Private บิต		Not defined
HDATA0 [9]	Padrate	1	Addition slot
		0	Normal frame
HDATA0 [7 : 6]	โหมด	3	โมโน
		2	ช่องสัญญาณคู่
		1	จอยส์ตรีโอ
		0	สเตริโอ
HDATA0 [5 : 4]	ส่วนขยาย		c.f. 11172-3
HDATA0 [3]	ลิขสิทธิ์	1	มีลิขสิทธิ์
		0	ฟรี
HDATA0 [2]	ต้นฉบับ	1	ต้นฉบับ
		0	ก๊อปปี้
HDATA [1 : 0]	Emphasis	3	CCITT J.17
		2	สำรองไว้
		1	50/5 ไมโครวินาที
		0	None

ตารางที่ 2.9 รีจิสเตอร์ HDATA and HDAT1 (R)

### 2.8.10 รีจิสเตอร์ AIADDR (RW)

AIADDR เป็นตัวแสดงแอดเดรสเริ่มต้นของการเขียนรหัสประยุกต์ที่เร็วกว่า รีจิสเตอร์ WRAMADDR และ WRAM ถ้าไม่ใช้รหัสประยุกต์ รีจิสเตอร์นี้จะไม่เป็นตัวแรกหรือตัวแรกจะเป็น 0

### 2.8.11 รีจิสเตอร์ VOL (RW)

VOL ตัวคุมระดับสัญญาณเสียงสำหรับเครื่องเล่น สำหรับแต่ละช่องสัญญาณ ค่า ในช่วงของ 0 ... 255 อาจกำหนดให้มันเบาลงจากระดับสัญญาณเสียงสูงสุด (ระดับใน 0.5 dB) ทางช่องซ้ายจะมีหลายค่าโดย 256 และค่าที่เพิ่มขึ้น ดังนั้นระดับสัญญาณเสียงสูงสุดเป็น 0 และ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เทียบถ้าเป็น 65535 ตัวอย่าง : สำหรับระดับสัญญาณเสียง  $-2.0$  dB ของช่องซ้ายและ  $-3.5$  dB ของช่องขวา :  $(4 \times 256) + 7 = 1031$  ดังนั้นที่ startup ระดับสัญญาณเสียงเซตเป็นเต็ม 1 ระดับสัญญาณเสียงการรีเซตซอฟต์แวร์ต้องไม่มีเซตระดับสัญญาณเสียง ดังนั้นการเซตระดับสัญญาณเสียงเป็น total silence (255 สำหรับช่องซ้ายและขวา) จะปิดอะนาล็อก power แต่ทำให้เกิดการสั้นในหูฟัง ถ้าคุณต้องการปิดระดับสัญญาณเสียง แต่ไม่ต้องการการสั้นนี้ ต้องปรับระดับสัญญาณเสียงเป็น 254 ทั้ง 2 ช่อง (0 x FEFE)

### 2.8.12 รีจิสเตอร์ A1CTRL[x] (RW)

A1CTRL[x] – รีจิสเตอร์ (  $x = [0,2]$  ) สามารถใช้เข้ากับโปรแกรมประยุกต์ของผู้ใช้ได้

## 2.9 การทำงานส่วนต่าง ๆ ของชิพ VS1002d

### 2.9.1 สัญญาณนาฬิกา

ชิพ VS1002d จะทำงานบนความถี่  $24.576$  เมกะเฮิร์ตซ์ ของสัญญาณนาฬิกาหลักนี้สามารถกำเนิดจากวงจรถ่ายนอก (ต่อที่ขา XTAL1) หรือโดยภายในคริสตัลออสซิลเลเตอร์เฟส (ขา XTAL1 และ XTAL0) นี้เป็นสัญญาณนาฬิกาที่พ่วงเพียงกับเอาต์พุตอดีโอคุณภาพสูงสำหรับทุกมาตรฐานอัตราตัวอย่าง

### 2.9.2 โหมดประหยัดพลังงาน

ในโหมดประหยัดพลังงาน ชิพแสดงเพียงสายคอนโทรลตัวขับเคลื่อนเอาต์พุตจะปิดและการประมวลผลตกค้างในสถานะคงค่า (Hold State)

### 2.9.3 การรีเซตตัวเครื่อง

เมื่อขา xRESET ป้อนพัลส์ “0” เข้าไป VS1002d จะรีเซตและทุกรีจิสเตอร์คอนโทรลและค่ากำหนดภายในต้องเซตเป็นค่าเริ่มต้น สัญญาณ xRESET เป็นอะซิงโครนัส ไม่ขึ้นกับสัญญาณนาฬิกาภายนอก รีเซตโหมดดับเบิลจะเป็นฟูลเพาเวอร์คาว์โหมด เมื่อทั้งส่วนสัญญาณดิจิทัลและอะนาล็อกของชิพ VS1002d เป็นพาวเวอร์น้อยสุด จะทำให้เมื่อสัญญาณนาฬิกาหยุด

หลังจากตัวเครื่องรีเซตแล้ว จะเซตให้ตัวโปรแกรมในรีจิสเตอร์เช่นรีจิสเตอร์ควบคุมเสียงให้พร้อมแล้วจึงทำการเริ่มถอดรหัสสัญญาณเสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 2.9.4 การรีเซตโปรแกรม

ระหว่างข้อมูลเอ็มเป็ก หากมีการรีเซตซอฟต์แวร์ จะมีการกระตุ้นให้บิตที่ 2 ในรีจิสเตอร์ SCI และต้องรอนาน้อย 2 ไมโครวินาที จากนั้นขา DREQ จะเป็น “0” ประมาณ 6000 เมทซินไซเคิล (ประมาณ 250 ไมโครวินาที) จะเป็นการหน่วงเวลาเอาไว้เพราะว่าชิพ VS1002d จะทำงานที่สัญญาณนาฬิกา 25.576 เมกะเฮิร์ตซ์ เมื่อสัญญาณ DREQ เป็น “1” ให้ส่งข้อมูลใด ๆ เข้าสู่ส่วน SDI หลังจากนั้นตัวชิพก็จะทำงานต่อไปได้อย่างปกติ

ถ้าต้องการให้แน่ใจว่าตัวชิพ VS1002d จะไม่หยุดกลางคันถ้าหากมีการส่งข้อมูลมาซ้ำมากให้ส่ง “2048H” เข้าสู่ส่วน SDI ก่อนที่จะทำการรีเซตโปรแกรม

#### 2.9.5 การถอดรหัสข้อมูลเอ็มเป็ก

โดยการทำงานปกติส่วนข้อมูลเอ็มเป็กเมื่อถูกถอดรหัส ตัวถอดรหัสจะทำการสุ่มเอาข้อมูลเอ็มเป็ก แล้วมาทำการถอดรหัสเป็นสัญญาณดิจิตอลภายในส่วน DAC ถ้าหากมีการผิดพลาดระหว่างการถอดรหัสแล้วจะมีการเซตบิตขึ้นในรีจิสเตอร์คอนโทรล (SCI) ที่บิต HDAT0 และ HDAT1 ในกรณีที่เกิดการล้มเหลวขึ้นในการส่งข้อมูล ตัวถอดรหัสจะยังคงทำงานแต่สัญญาณอะนาลอกจะไม่มีสัญญาณออกมา

#### 2.9.6 การตรวจสอบว่ามีการทำงานอย่างปกติหรือไม่

ถึงแม้ว่าตัวชิพเองจะมีการตรวจสอบอยู่แล้ว แต่ก็อาจมีเหตุให้เกิดการไม่ถอดรหัสข้อมูลบางตัว ทำให้โปรแกรมทำงานผิดพลาดได้ โดยอาจเกิดในกรณีการเร่งสปีดในการถอดรหัส

ตัวไมโครคอนโทรลเลอร์เองจะเก็บค่าความเร็วที่ VS1002d ต้องการไว้ ถ้าตัวข้อมูลเกิดการหยุดส่งหรือถ้าตัวชิพต้องการข้อมูลมากกว่า 60 กิโลบิตในทุก ๆ วินาที มันจะส่งผลต่อตัวไมโครคอนโทรลเลอร์เวลาตัวโปรแกรมเกิดการรีเซต ถ้าหากไม่มีการตรวจสอบแล้วเกิดการรีเซตทางฮาร์ดแวร์จะยังคงทำงาน

#### 2.9.7 การทดสอบ

มีการทดสอบหลายอย่างในการทดสอบชิพ VS1002d โดยอาจมีการทดสอบหน่วยความจำ ส่วนส่งสัญญาณอนุกรม (SPI) และเอาท์พุตสัญญาณไซน์ที่มีความถี่ตั้งแต่ 250 Hz ถึง 1500 Hz

การทดสอบสามารถทำได้หลายทางหากมีข้อมูลเอ็มเป็กถูกถอดรหัส ตัวถอดรหัส จะจบการถอดรหัสโดยทำการส่งข้อมูล 1024H ออกมา ทำให้เชื่อได้ว่าตัวถอดรหัสมองหาข้อมูลตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต่อไปและจะยังไม่ถอดรหัสข้อมูลโดยปราศจากข้อมูลเอ็มเป็ก ถ้าไม่มีข้อมูลเอ็มเป็กถอดรหัสตั้งแต่การรีเซตครั้งก่อนขั้นตอนนี้ก็จะไม่เกิด การทดสอบแต่ละครั้งจะเริ่มโดยการส่งคำสั่ง 4 ไบต์ตามลำดับ โดยการส่งแต่ละครั้งจะให้เอาท์พุทออกมา

### 2.9.8 อัตราเร็วสูงสุดของสัญญาณนาฬิกา

การใช้สัญญาณนาฬิกาอื่น ที่ไม่ใช่ความเร็วของสัญญาณนาฬิกาที่ 24.576 เมกะเฮิร์ตซ์ นี้จะส่งผลกระทบต่อความเร็วของการสุ่มเอาข้อมูล ความเร็วในการถอดรหัสบิตข้อมูลและผลกระทบต่อตัว DSP

### 2.9.9 ความเร็วสูงสุดในการสุ่มข้อมูล

ที่สัญญาณนาฬิกา 24.576 เมกะเฮิร์ตซ์หรือสูงกว่า อัตราสุ่มเอาข้อมูลสูงสุดอยู่ที่ 48000 Hz สำหรับที่ความเร็วต่ำกว่าสามารถคำนวณได้จาก  $(4800 \times \text{ความเร็วสัญญาณนาฬิกา (เมกะเฮิร์ตซ์)}) / 24.576$  จะเห็นว่าถ้าความเร็วของสัญญาณนาฬิกาลดลง จะทำให้ความเร็วในการสุ่มเอาข้อมูลลดลงด้วย

หากนำเอาสัญญาณนาฬิกาที่ 26 เมกะเฮิร์ตซ์ มาใส่จะเป็นความเร็วสูงสุดของสัญญาณนาฬิกาในการสุ่มเอาข้อมูลโดยจะมีความเร็วในการสุ่มข้อมูลถึง 50781 Hz ดังนั้น ทุก ๆ ข้อมูลเอ็มเป็กจะถูกถอดรหัสออกมาอย่างถูกต้อง แต่ที่ 48000 Hz แล้วจะไม่สามารถถอดรหัสได้อย่างถูกต้อง

### 2.9.10 อัตราเร็วสูงสุดของบิตข้อมูลในการถอดรหัส

ที่สัญญาณนาฬิกา 24.576 เมกะเฮิร์ตซ์ อัตราในการถอดรหัสข้อมูลเอ็มเป็กได้ 256 kbps และหากความเร็วของสัญญาณนาฬิกาต่างกันจะทำให้อัตราเร็วในการถอดรหัสต่างกันไปด้วย

### 2.9.11 การทำงานเมื่อตอนรีเซต

เมื่อชิพ VS1002d ทำการรีเซต (ขา xRESET เป็น "1") เมื่อเวลาผ่านไปประมาณ 4096 เมทซินไซเคิล ขา DREQ ควรจะเป็น "0" ถ้าหากไม่เป็น "0" แล้วหลังจากเวลาผ่านไปประมาณ 6000 เมทซินไซเคิล ขา DREQ ควรจะเป็น "1" ถ้าขา DREQ ไม่เปลี่ยนแปลงตามนี้แล้วการทำงานของโปรแกรมจะไม่ถูกต้อง

### 2.9.12 การส่งข้อมูลสู่สายสัญญาณ SCI

ถ้าเราต้องการส่งข้อมูลที่ทำให้เอาท์พุตออกมาเป็น  $-2\text{dB}$  ที่ช่องสัญญาณซ้าย และ  $-3.5\text{dB}$  ที่ช่องสัญญาณขวา ดังนั้นเพื่อให้เป็นดังข้างต้นเราต้องส่งข้อมูล 0407H ไปที่รีจิสเตอร์เสียง (VOL register)

- ให้ส่วน SCI ทำงาน
- ส่งข้อมูล 4 ไบต์ 02H, 0BH, 04H, 07H ผ่านสาย SCI
- อ่านข้อมูลเข้ามาส่วนรีจิสเตอร์ MISO ทีละ 8 บิต โดยจะเป็น MSB และ LSB

ตามลำดับ

- ข้อมูลที่ได้จะเป็น 16 บิต
- เมื่อจบการอ่านข้อมูลก็เซตให้ขา GIO1 เป็น “1”

### 2.9.13 การส่งข้อมูลสู่ส่วน SDI (ข้อมูลเอ็มเป็ก)

เราสามารถส่งข้อมูล 32 ไบต์ หรือเล็กกว่าข้อมูลเอ็มเป็กคู่ VS1002d ได้

- รองนกว่าขา DREQ เป็น “1”
- ให้ขา DCLK มีสัญญาณนาฬิกา
- ในการส่งแต่ละไบต์ให้ทำตามขั้นตอนดังนี้
  1. ให้ขา BSYNC เป็น “1”
  2. ทำให้ SPI ทำงาน
  3. รองนกว่าเรารู้ว่าข้อมูลบิตแรกถูกส่งออกไปแล้วแต่บิตสุดท้ายยังไม่ถูกส่งออกไป
- เซตให้ขา BSYNC เป็น “0”
- รองนกว่าไมโครคอนโทรลเลอร์ส่งข้อมูลผ่าน SPI จนจบ

### บทที่ 3

#### ไมโครคอนโทรลเลอร์

ไมโครคอนโทรลเลอร์ (Microcontroller) เป็นชื่อของอุปกรณ์อิเล็กทรอนิกส์แบบหนึ่ง ที่รวมเอาหน่วยประมวลผล หน่วยคำนวณทางคณิตศาสตร์และลอจิก วงจรรับสัญญาณอินพุต วงจรขับสัญญาณเอาต์พุต หน่วยความจำ วงจรกำเนิดสัญญาณนาฬิกาไว้ด้วยกัน ทำให้สามารถนำไปใช้งานแทนวงจรอิเล็กทรอนิกส์ที่ซับซ้อนได้เป็นอย่างดี ช่วยลดจำนวนอุปกรณ์และขนาดของระบบ ในขณะที่มีขีดความสามารถสูงขึ้น

ไมโครคอนโทรลเลอร์ มาจากคำ 2 คำรวมกัน คือ ไมโคร (Micro) ซึ่งหมายถึง ไมโครโปรเซสเซอร์ (Microprocessor) ซึ่งเป็นอุปกรณ์ประมวลผลขนาดเล็กภายในประกอบด้วย หน่วยประมวลผล กลางหรือซีพียู (CPU : Central Processing Unit) หน่วยคำนวณทางคณิตศาสตร์และลอจิก (ALU : Arithmetic Logic Unit) วงจรเชื่อมต่อหน่วยความจำและ วงจรสัญญาณนาฬิกาอีกคำหนึ่งเรียกว่า “คอนโทรลเลอร์” (Controller) หมายถึง อุปกรณ์ควบคุม ดังนั้นไมโครคอนโทรลเลอร์จึงเป็นอุปกรณ์ที่ใช้ในการควบคุม โดยที่จะสามารถเขียนโปรแกรมเพื่อกำหนดรูปแบบการควบคุมได้อย่างอิสระ

#### 3.1 สถาปัตยกรรม และคุณสมบัติโดยทั่วไปของ ATMEGA162

- สถาปัตยกรรมภายในถูกออกแบบให้ใช้สถาปัตยกรรมแบบ RISC (Reduce Instruction Set Computer)
- หน่วยความจำแบบ FLASH สำหรับบันทึก PROGRAM MEMORY ขนาด 16 KByte
- หน่วยความจำแบบ EEPROM สำหรับบันทึก DATA MEMORY ขนาด 512 Byte
- หน่วยความจำแบบ RAM ขนาด 1 KByte
- ระบบเปลี่ยนสัญญาณ ANALOG TO DIGTL ขนาด 10 บิต จำนวน 8 CHANNEL
- กลุ่มรีจิสเตอร์ใช้งานทั่วไป 8 บิต จำนวน 32 ตัว
- พอร์ตอินพุตและเอาต์พุตขนาด 8 บิต จำนวน 4 พอร์ต
- ระบบสื่อสารข้อมูลดิจิทัลแบบอะซิงโครนัส (UART) 1 CHANNEL
- ระบบสื่อสารข้อมูลแบบซิงโครนัส (SPI) 1 CHANNEL
- ความถี่สัญญาณนาฬิกา 0 – 16 MHz
- ระบบการรีเซตแบบฮาร์ด โนมัติเมื่อเริ่มจ่ายกระแสไฟฟ้าเข้าไมโครคอนโทรลเลอร์
- ระบบการกำเนิดความถี่สัญญาณแบบ PWM จำนวน 3 CHANNEL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ระบบการตรวจจับสัญญาณอนาล็อก (Analog Comparator)
- ระบบการป้องกันการ COPY ข้อมูลภายในหน่วยความจำ
- ระบบตรวจจับการทำงานผิดพลาดของ CPU (WATCHDOG TIMER ON-CHIP OSCILATOR)
- ระบบอินเตอร์รัพท์จากภายนอก (EXTERNAL INTERRUPT)
- TIMER/COUNTER ขนาด 16 บิต 2 CHANNEL
- TIMER/COUNTER ขนาด 8 บิต 2 CHANNEL

ATMEGA162 เป็นไมโครคอนโทรลเลอร์ขนาด 8 บิต ที่สถาปัตยกรรมแบบ RISC (Reduce Instruction Set Computer) ซึ่งทำให้การประมวลผลมีความเร็ว 1 คำสั่ง/1 Clock หรือ CPU สามารถประมวลคำสั่งได้ 1 MIPS / MHz จะมีหน่วยความจำสำหรับ PROGRAM MEMORY แบบ FLASH ขนาด 16 KByte หน่วยความจำสำหรับ DATA MEMORY แบบ EEPROM ขนาด 512 Byte และหน่วยความจำแบบ RAM ขนาด 1 KByte มีพอร์ตที่สามารถทำงานได้สองทิศทางจำนวน 32 เส้น สัญญาณ PWM และระบบ TIME/COUNTER จำนวน 4 ชุด ที่มีโหมดการทำงานเสริมในส่วนของการสร้างสัญญาณ Input Capture มีอุปกรณ์สื่อสารข้อมูลอนุกรมแบบ UART และ SPI และยังมีระบบการแปลง สัญญาณ Analog to digital ขนาด 10 บิต จำนวน 8 ช่องสัญญาณ ที่มาพร้อมกับ MCU มี Watchdog Timer เพื่อตรวจสอบการทำงานของไมโครคอนโทรลเลอร์ และมีระบบการประหยัดพลังงานอีก 3 ระบบ

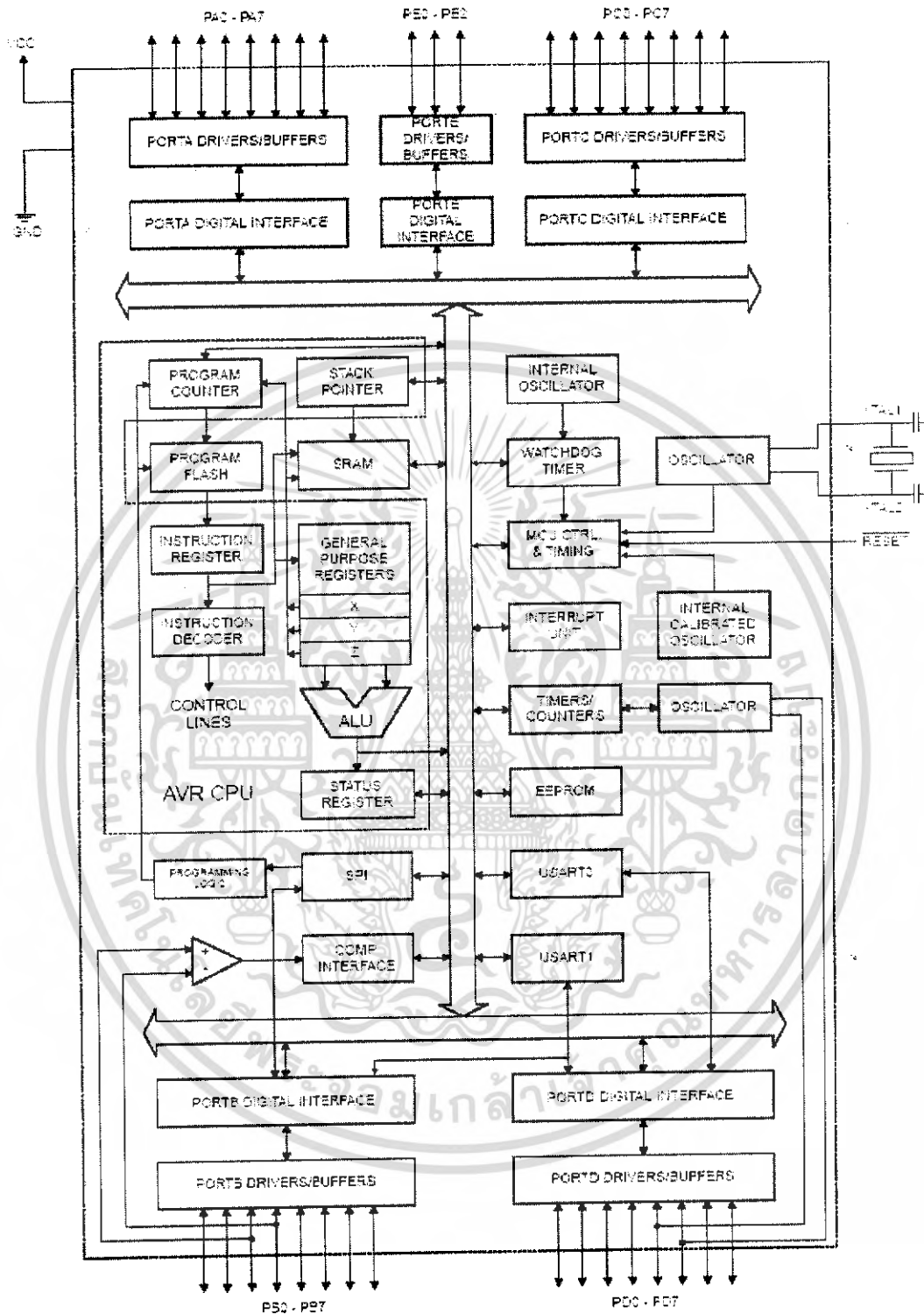
### 3.2 สถาปัตยกรรมภายในและรีจิสเตอร์ใช้งานทั่วไป

#### 3.2.1 สถาปัตยกรรมภายใน

โครงสร้างภายในจะประกอบด้วยรีจิสเตอร์ใช้งานทั่วไปขนาด 8 บิต จำนวน 32 ตัว ที่สามารถเข้าถึงข้อมูลได้ใน 1 Clock ซึ่งหมายความว่า MCU สามารถจัดการข้อมูลภายในรีจิสเตอร์ใช้งานทั่วไปได้เสร็จภายใน 1 Clock ของสัญญาณนาฬิกาโดยรีจิสเตอร์ R26-F31 เป็นรีจิสเตอร์ขนาด 8 บิต จำนวน 6 ตัว สามารถจับคู่เพื่อใช้เป็นรีจิสเตอร์ขนาด 16 บิต 3 ตัว โดยใช้ชื่อว่ารีจิสเตอร์ X, Y, Z

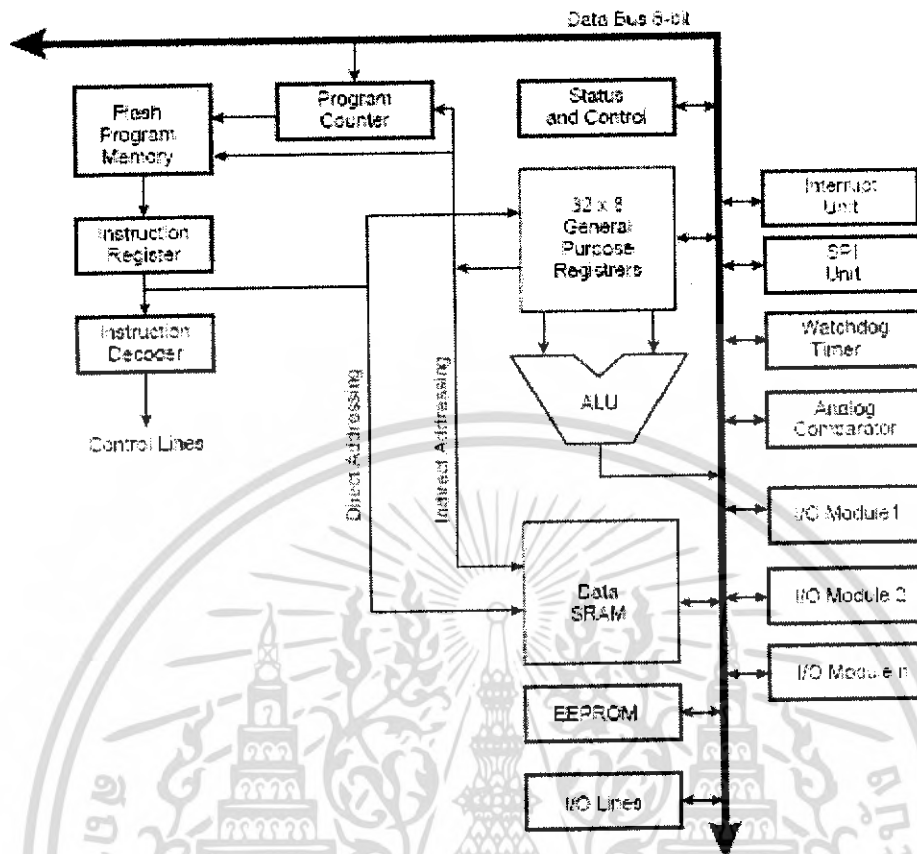
ALU จะสนับสนุนการกระทำทางคณิตศาสตร์และลอจิก ระหว่าง รีจิสเตอร์กับรีจิสเตอร์ หรือ ระหว่าง รีจิสเตอร์กับค่าคงที่ ซึ่งการเรียกรับรีจิสเตอร์ใช้งานทั่วไป สามารถกระทำได้ โดยการอ้างหน่วยความจำภายในที่ตำแหน่ง S00 – S1F จำนวน 32 ตำแหน่ง และใน MCU ได้จัดแบ่งให้มีรีจิสเตอร์ที่ใช้ควบคุมการทำงานของหน่วยความจำที่ตำแหน่ง S20 – S1F

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1 แผนผังโครงสร้างของ ATMEGA162

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 สถาปัตยกรรมแบบ RISC ของ ATMEGA162

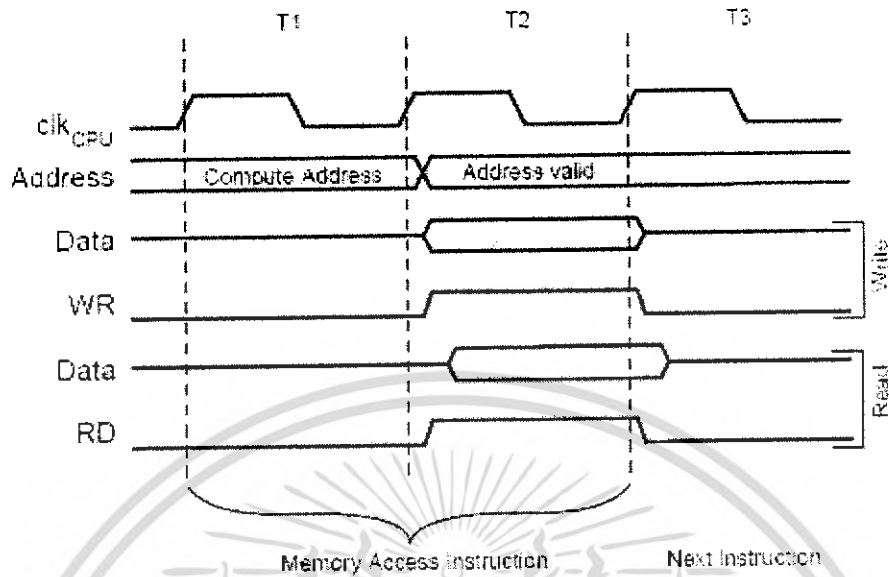
ระบบการทำงานของไมโครคอนโทรลเลอร์ใช้หลักการออกแบบของ HAVARD ด้วยการแยกระบบบัสของ PROGRAM และ DATA ออกจากกัน โดยโปรแกรมจะมีการประมวลผลด้วย SINGLE LEVEL PIPELINING ซึ่งทำให้ CPU สามารถ Fetch และ Execute คำสั่งได้ภายใน 1 คาบเวลา ด้วยคำสั่ง JUMP และ CALL แบบ RELATIVE ที่สามารถกระโดดข้ามการทำงานได้ไกลถึง 2K/4K ซึ่งใน 1 คำสั่งจะใช้รหัสการทำงาน 16 Bit หรือ 1 WORD โดยทุกครั้งที่มีการอินเตอร์รัพท์ หรือ การข้ามไปทำงานในโปรแกรมย่อยค่าของ PROGRAM COUNTER (PC) จะถูกเก็บลง STACK ซึ่งจะใช้พื้นที่หน่วยความจำใน SRAM บางส่วนเพื่อทำเป็นพื้นที่ของ STACK

### 3.2.1.1 หน่วยประมวลผลทางคณิตศาสตร์ และ ลอจิก

ระบบการประมวลผลที่มีประสิทธิภาพของ AVR คือ ALU สามารถสื่อสารข้อมูลโดยตรงกับรีจิสเตอร์ใช้งานทั่วไปได้ทั้ง 32 บิต โดย ALU ได้จัดแบ่งระบบการจัดการข้อมูลไว้ 3 ส่วน คือ ส่วนของการจัดการทางคณิตศาสตร์ ส่วนการกระทำทางลอจิกและในส่วนของการกระทำกับบิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





รูปที่ 3.4 การเข้าถึงข้อมูล SRAM ภายใน CPU

### 3.2.2 รีจิสเตอร์ใช้งานทั่วไป

โครงสร้างของรีจิสเตอร์ใช้งานทั่วไปทั้ง 32 ตัว แสดงดังรูปที่ 3.5

	7	0	Addr.
	R0		0x00
	R1		0x01
	R2		0x02
	...		
	R13		0x0D
	R14		0x0E
	R15		0x0F
General Purpose Working Registers	R16		0x10
	R17		0x11
	...		
	R26		0x1A
	R27		0x1B
	R28		0x1C
	R29		0x1D
	R30		0x1E
	R31		0x1F

รูปที่ 3.5 โครงสร้างของรีจิสเตอร์ใช้งานทั่วไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รีจิสเตอร์ทั้งหมดสามารถใช้ชุดคำสั่ง เพื่อเข้าถึงได้โดยตรงและจะใช้ช่วงเวลาที่การเข้าถึงเพียง 1 Clock โดยคำสั่ง SBCL, SUBI, CPI, ANDI และ ORI ซึ่งกระทำระหว่างรีจิสเตอร์กับค่าคงที่ และรีจิสเตอร์กับรีจิสเตอร์ และคำสั่ง LDI ที่ใช้โหนดค่าคงที่เข้าไปในรีจิสเตอร์ จะต้องใช้งานกับรีจิสเตอร์ R16 ถึง R31 ส่วนคำสั่ง SCB, SUB, CP, AND และ OR และคำสั่งใช้งานอื่น ๆ สามารถใช้งานได้ทั้งในรีจิสเตอร์ทั่วไป

### 3.2.2.1 The Status Register – SREG

Bit	7	6	5	4	3	2	1	0	
	I	T	H	S	V	N	Z	C	SREG
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

รูปที่ 3.6 ตำแหน่งรีจิสเตอร์ SREG ถูกจัดวางไว้ที่ตำแหน่ง \$3F

- Bit 7: Global Interrupt Enable

บิต I จะต้องถูก SET เป็น I เมื่อต้องการกำหนดให้ INTERRUPT ทั้งหมดทำงาน โดยอินเทอร์รัพท์แต่ละแหล่งจะสามารถแบ่งแยกกันกำหนดได้ในรีจิสเตอร์ GIMSK และ TIMSK แต่ละถ่าบิต 1 ถูก CLAR เป็น 0 อินเทอร์รัพท์ทุกตัวจะถูก DISABLE โดยบิต I จะถูก Clear โดยอัตโนมัติ เมื่อเกิดการอินเทอร์รัพท์และถูก SET โดยการใช้นี้คำสั่ง RET

- Bit 6: Bit Copy Storage

การใช้นี้คำสั่ง BLD และคำสั่ง BST จะเป็นคำสั่งที่ใช้บิต T ทำหน้าที่เป็น SOURCE หรือ DESTINATION โดยบิตต่าง ๆ ในรีจิสเตอร์สามารถ COPY บิต T ลงในบิตของรีจิสเตอร์ได้โดยใช้นี้คำสั่ง BST

- Bit 5: Half Carry Flag

บิต H แสดงการเกิด Half Carry Flag

- Bit 4: Sign Bit,  $S = N + V$

เป็นบิตที่ใช้แสดงเครื่องหมาย

- Bit 3: Two's Complement Overflow Flag

แสดงการทำ Two's Complement

- Bit 2: Negative Flag

จะแสดงการทำคำสั่งที่เป็นลบของการกระทำทางคณิตศาสตร์และลอจิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Bit 1: Zero Flag

แสดงการทำคำสั่งทางคณิตศาสตร์และลอจิกที่ให้ผลเป็น 0

- Bit 0: Carry Flag

แสดงการเกิด Carry Flag

### 3.2.2.2 Stack Pointer: SP

ใน ATMEGA162 จะใช้รีจิสเตอร์ขนาด 8 บิต bit 2 ตัว ในตำแหน่ง \$3E และ \$3D ทำหน้าที่เป็น STACK POINTER มีหน้าที่เก็บข้อมูลของรีจิสเตอร์ต่างๆ ไว้ชั่วคราว เมื่อโปรแกรมทำการตอบสนองการอินเทอร์รัพท์จากแหล่งกำเนิดนั้นๆ โดยก่อนที่โปรแกรมจะกระโดดไปตอบสนองโปรแกรมย่อย จะมีการเก็บค่าที่สำคัญเอาไว้ก่อน มิฉะนั้นแล้วอาจจะทำให้ค่าของรีจิสเตอร์เปลี่ยนแปลงไป เมื่อโปรแกรมกลับมาทำคำสั่งต่อไป จะทำให้เกิดการผิดพลาดได้

Bit	15	14	13	12	11	10	9	8	
	SP16	SP14	SP13	SP12	SP11	SP10	SP9	SP8	SPH
	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0	SPL
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	

รูปที่ 3.7 ตำแหน่งรีจิสเตอร์ Stack Pointer ถูกจัดวางไว้ที่ตำแหน่ง \$3E และ \$3D

## 3.3 การรีเซ็ตและการอินเทอร์รัพท์

### 3.3.1 การรีเซ็ตและการอินเทอร์รัพท์

ไมโครคอนโทรลเลอร์ ATMEGA162 มีระบบการตอบสนองสัญญาณอินเทอร์รัพท์จาก 16 แหล่งสัญญาณ โดยได้แยกอินเทอร์รัพท์เวกเตอร์ของแต่ละอินเทอร์รัพท์ออกจากกัน ในการควบคุมการตอบสนองของอินเทอร์รัพท์แต่ละแหล่ง สามารถแยกการควบคุมได้จาก บิต Enable ของอินเทอร์รัพท์นั้นๆ และบิต 1 ซึ่งใช้ควบคุมการอินเทอร์รัพท์ทั้งหมด โดยตำแหน่งแรกๆ ใน PROGRAM MEMORY จะเป็นตำแหน่งที่ถูกใช้เป็นที่ของอินเทอร์รัพท์เวกเตอร์และรีเซ็ตซึ่งตารางที่ 2 แสดงอินเทอร์รัพท์เวกเตอร์ต่าง ๆ โดยเริ่มจากอินเทอร์รัพท์ที่มีระดับความสำคัญสูงสุด คือ KESSET จนถึงอินเทอร์รัพท์ที่มีระดับความสำคัญต่ำสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3.2 สัญญาณรีเซ็ต

ไมโครคอนโทรลเลอร์ ATMEGA162 มีสัญญาณรีเซ็ต 3 แหล่ง คือ

1. POWER ON RESET ไมโครคอนโทรลเลอร์จะรีเซ็ตเมื่อมีการจ่ายไฟให้กับขา Vcc และ GND (ตอนจ่ายไฟเลี้ยงครั้งแรก)
2. EXTERNAL RESET ไมโครคอนโทรลเลอร์จะรีเซ็ตเมื่อมีสัญญาณลอจิก LOW เข้ามาที่ขารีเซ็ตเป็นระยะเวลามากกว่า 2 คาบเวลาของสัญญาณ XTAL
3. WATCHDOG RESET ไมโครคอนโทรลเลอร์จะทำการรีเซ็ต เมื่อถึงคาบเวลาของ WATCHDOG และ WATCHDOG จะต้องถูกกำหนดให้ทำงาน

ในระหว่างที่เกิดการรีเซ็ต รีจิสเตอร์ทั้งหมดจะถูกกำหนดให้ค่าเริ่มต้น และ โปรแกรมจะเริ่มทำงานที่ตำแหน่ง \$0000 โดยคำสั่งที่ตำแหน่ง \$0000 จะต้องเป็นคำสั่ง RJMP แต่ถ้าโปรแกรมไม่มีการกำหนดให้มีการใช้อินเตอร์รัพท์พื้นที่ส่วนที่เป็นอินเตอร์รัพท์เวกเตอร์จะถูกใช้เป็นพื้นที่ของโปรแกรม

#### 3.3.2.1 Power On Reset

วงจรของ Power On Reset (POR) ถูกสร้างขึ้นเพื่อให้แน่ใจว่า MCU จะไม่ทำงานถ้าระดับของแรงดันไฟเลี้ยงวงจรยังไม่ถึงระดับที่จะทำให้ระบบการทำงานภายในของ MCU ทำงานได้อย่างถูกต้อง

ถ้าต้องการใช้ระบบ Power On Reset ภายใน MCU จะต้องต่อขารีเซ็ตเข้ากับ Vcc หรือ ต่อผ่านความต้านทาน Pull Up ค่าประมาณ 100K – 500K

โดยบิต FSTRT ใช้ในการเลือกคาบเวลา TIME – OUT ซึ่งเมื่อบิต FSTRT ถูกโปรแกรมจะทำให้ช่วงเวลา TIME – OUT มีค่านาน้อยลง ซึ่งมีประโยชน์อย่างมากเมื่อใช้ OSC ที่เป็น Ceramic Resonator

#### 3.3.2.2 External Reset

การรีเซ็ตจากภายนอกจะเกิดขึ้นโดยการให้ลอจิก LOW ที่ขา RESET อย่างน้อย 2 Clock ของสัญญาณนาฬิกา ซึ่งเมื่อสัญญาณที่ขอบขาขึ้นถึงระดับ Treshold Voltage จะทำให้วงจร Time Delay เริ่มทำงาน ซึ่ง CPU จะถูกรีเซ็ตเมื่อคาบเวลาของ Time Delay มากกว่าคาบเวลา Time – Out

### 3.3.2.3 Watchdog Reset

เมื่อ WTD TIME – OUT เริ่มกำเนิดสัญญาณ PULSE ที่มีคาบเวลาเท่ากับ Clock ของสัญญาณนาฬิกาโดยช่วงขอบขาของ PULSE จะทำให้ระบบการนับเวลาเริ่มทำงานจนหมดช่วงเวลาของ TIME – OUT จึงจะทำให้ CPU รีเซ็ต

Vector No.	Program Address <sup>(2)</sup>	Source	Interrupt Definition
1	0x000 <sup>(1)</sup>	RESET	External Pin, Power-on Reset, Brown-out Reset, Watchdog Reset, and JTAG AVR Reset
2	0x002	INT0	External Interrupt Request 0
3	0x004	INT1	External Interrupt Request 1
4	0x006	INT2	External Interrupt Request 2
5	0x008	PCINT0	Pin Change Interrupt Request 0
6	0x00A	PCINT1	Pin Change Interrupt Request 1
7	0x00C	TIMER3 CAPT	Timer/Counter3 Capture Event
8	0x00E	TIMER3 COMPA	Timer/Counter3 Compare Match A
9	0x010	TIMER3 COMPB	Timer/Counter3 Compare Match B
10	0x012	TIMER3 OVF	Timer/Counter3 Overflow
11	0x014	TIMER2 COMP	Timer/Counter2 Compare Match
12	0x016	TIMER2 OVF	Timer/Counter2 Overflow
13	0x018	TIMER1 CAPT	Timer/Counter1 Capture Event
14	0x01A	TIMER1 COMPA	Timer/Counter1 Compare Match A
15	0x01C	TIMER1 COMPB	Timer/Counter1 Compare Match B
16	0x01E	TIMER1 OVF	Timer/Counter1 Overflow
17	0x020	TIMER0 COMP	Timer/Counter0 Compare Match
18	0x022	TIMER0 OVF	Timer/Counter0 Overflow
19	0x024	SPI, STC	Serial Transfer Complete
20	0x026	USART0, RXC	USART0, Rx Complete
21	0x028	USART1, RXC	USART1, Rx Complete
22	0x02A	USART0, UDRE	USART0 Data Register Empty
23	0x02C	USART1, UDRE	USART1 Data Register Empty
24	0x02E	USART0, TXC	USART0, Tx Complete
25	0x030	USART1, TXC	USART1, Tx Complete
26	0x032	EE_RDY	EEPROM Ready
27	0x034	ANA_COMP	Analog Comparator
28	0x036	SPM_RDY	Store Program Memory Ready

ตารางที่ 3.1 รีเซ็ตและอินเตอร์รัพท์เวกเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4 Interrupt Handling

ไมโครคอนโทรลเลอร์ ATMEGA162 มีรีจิสเตอร์ที่ใช้งานอินเทอร์รัพท์ 2 ตัว คือ GIMSK และ TIMSK ที่ใช้บรรจุมิติควบคุมและแสดงบิตสถานะของอินเทอร์รัพท์ต่างๆ

เมื่ออินเทอร์รัพท์เกิดขึ้นจะทำให้บิต I ซึ่งเป็น บิตที่ใช้ควบคุมอินเทอร์รัพท์ถูกเคลียร์เป็น 0 และจะทำให้อินเทอร์รัพท์ทั้งหมดจะถูก Disable ผู้ใช้สามารถเซ็ทบิต I เป็น 1 เมื่อต้องการ Enable Interrupt

เมื่อเกิดอินเทอร์รัพท์จะทำให้ CPU กระโดดไปทำโปรแกรมบริการอินเทอร์รัพท์ และเคลียร์ Interrupt Flag (ของอินเทอร์รัพท์ที่กำลังเกิดขึ้น) ให้เป็น 0 หรือผู้ใช้สามารถเคลียร์ Interrupt Flag ได้โดยการเขียนลอกจาก 1 ลงใน Flag ที่ต้องการ

### 3.5 พอร์ตอินพุต/เอาต์พุต

#### 3.5.1 พอร์ต A

เป็นพอร์ต 2 ทิศทางขนาด 8 บิต โดยมีหน่วยควบคุมต่างๆ ของพอร์ต คือ รีจิสเตอร์ (PORTA REGISTER) อยู่ที่ตำแหน่ง \$1B (\$3B) รีจิสเตอร์ DDRA (DATA DIRECTION REGISTER) อยู่ที่ตำแหน่ง \$1A (\$3A) และ PINA (PORT A INPUT PINS) อยู่ที่ตำแหน่ง \$19 (\$39) โดย PINA จะสามารถอ่านได้อย่างเดียว ไม่สามารถเขียนข้อมูลลงได้ ในขณะที่ PORTA และ DDRA สามารถอ่านเขียนได้ โดยแต่ละขาสัญญาณของพอร์ต A จะสามารถกำหนดให้ค่าความต้านทาน PULL LOW ในขณะที่ความต้านทาน PULL UP ภายใน ACTIVE จะ ทำให้ MCU ข่ายกระแสออกมาภายนอก

พอร์ต A จะถูกนำไปใช้งานอีกอย่างหนึ่ง คือ เป็นขาสัญญาณอินพุตของวงจร ANALOG TO DIGITAL โดยถ้าบางขาของพอร์ต A ในขณะที่วงจร ANALOG TO DIGITAL กำลัง Conversion สัญญาณ ซึ่งอาจจะทำให้การทำการแปลงสัญญาณถูกขัดจังหวะ จะมีผลทำให้การแปลงสัญญาณผิดพลาด การทำงานในโหมดประหยัคพลังงานจะทำให้ SCHMITT TRIGGER ของส่วนเอาต์พุตถูกตัดออก

Bit	7	6	5	4	3	2	1	0	
	PORTA7	PORTA6	PORTA5	PORTA4	PORTA3	PORTA2	PORTA1	PORTA0	PORTA
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

รูปที่ 3.8 Port A Data Register: PORTA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.5.2 พอร์ต B

พอร์ต B เป็นพอร์ตสองทิศทาง ขนาด 8 บิต โดยมีรีจิสเตอร์ที่ใช้ควบคุมพอร์ต B อยู่ 3 ตัว คือ รีจิสเตอร์ PORTB อยู่ที่ตำแหน่ง \$18 (\$38), รีจิสเตอร์ DDRB อยู่ที่ตำแหน่ง \$16 (\$36) โดยพอร์ต B แต่ละขาสามารถแยกกำหนดให้มีความต้านทาน PULL UP ได้ตามต้องการ ซึ่งแต่ละขาสามารถรับกระแส (SINK CURRENT) ได้ 20 mA โดยขาของพอร์ต B ยังมีหน้าที่อื่นๆอีก คือ

Bit	7	6	5	4	3	2	1	0
	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

รูปที่ 3.9 Port B Data Register: PORTB

ฟังก์ชันอื่นๆ ที่ใช้ขาสัญญาณของพอร์ต B

- SCK – PORTB, Bit 7  
เป็นขา CLOCK ที่ใช้งานในส่วนของการสื่อสารแบบ SPI
- MISO – PORTB, Bit 6  
เป็นขารับข้อมูลการสื่อสารแบบ SPI
- MOSI – PORTB, Bit 5  
เป็นขาส่งข้อมูลในการสื่อสารแบบ SPI
- SS – PORTB, Bit 4  
ใช้เป็นขาควบคุมการทำงานในโหมด SPI
- AIN1 – PORTB, Bit 3  
เป็นขาอินพุต 1 ของการทำงานในส่วนของ Analog Compare
- AIN0 – PORTB, Bit 2  
เป็นขาอินพุต 0 ของการทำงานในส่วนของ Analog Compare
- T1 – PORTB, Bit 1  
เป็นขาอินพุตที่ใช้ในส่วนของ Timer 1
- T0 – PORTB, Bit 0  
เป็นขาอินพุตที่ใช้ในส่วนของ Timer 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.5.3 พอร์ต C

พอร์ต C เป็นพอร์ตสองทิศทาง ขนาด 8 บิต โดยมีหน่วยควบคุมการทำงานของพอร์ตจำนวน 3 หน่วย คือ PORTC (Data Register – PORTC) อยู่ที่ตำแหน่งหน่วยความจำ \$15(\$35), DDRC (Data Direction Register – DDRC) อยู่ที่ตำแหน่งหน่วย \$14 (\$34) และ PINC (Port C Input Pin – PINC) อยู่ที่ตำแหน่ง \$13 (\$33) โดย PINC จะสามารถอ่านได้อย่างเดียว ในขณะที่ PORTC และ DDRC จะสามารถทั้งอ่านและเขียนได้โดยแต่ละขาของพอร์ต C สามารถแยกกำหนดให้มีความต้านทาน PULL UP ได้ ในขณะที่พอร์ต C แต่ละขาสามารถรับกระแส (SINK CURRENT) ได้ 20 mA โดยถ้าภายในกำหนดให้มีความต้านทาน PULL UP และภายนอกมีความต้านทาน PULL LOW จะทำให้ MCU จ่ายกระแสออกภายนอก

Bit	7	6	5	4	3	2	1	0	
	PORTC7	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0	Port C
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

รูปที่ 3.10 Port C Data Register: PORTC

### 3.5.4 พอร์ต D

พอร์ต D เป็นพอร์ตสองทิศทาง ขนาด 8 บิต ที่มีหน่วยควบคุมการทำงานของพอร์ตคือ PORTD (DATA Register – PORTD) อยู่ที่ตำแหน่ง \$12 (\$30) โดย PIND สามารถอ่านได้อย่างเดียว ในขณะที่ PORTD และ DDRD สามารถทั้งอ่านและเขียน โดยพอร์ต D สามารถรับกระแสได้ 20 mA ซึ่งแต่ละขาของพอร์ต D สามารถเลือกฟังก์ชันการทำงานอื่น ๆ ได้อีก

Bit	7	6	5	4	3	2	1	0
	PORTD7	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

รูปที่ 3.11 Port D Data Register: PORTD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีก้ารนำไปใช้

การทำงานของฟังก์ชันอื่นของพอร์ต D

- OC2 – PORTD, Bit 7

ขา OC2 ใช้งานเป็นขาเอาต์พุตของ Compare Output Match ของ Timer/ Counter2

- ICP – PORTD, Bit 6

ขา ICP ใช้งานเป็นขาอินพุตของ Input Capture Mode ของ Timer/Counter1

- OC1A – PORTD, Bit 5

ขา OC1A ใช้งานเป็นขาเอาต์พุตของ Output CompareA Match ของ Timer/ Counter1

- OC1B – PORTD, Bit 4

ขา OC1B ใช้งานเป็นขาเอาต์พุตของ Output CompareB Match ของ Timer/ Counter1

- INT1 – PORTD, Bit 3

ขา INT1 ใช้งานเป็นขาอินพุตของ อินเทอร์รัพท์จากภายนอก

- INT0 – PORTD, Bit 2

ขา INT0 ใช้งานเป็นขาอินพุตของอินเทอร์รัพท์ จากภายนอก

- TXD – PORTD, Bit 1

ขา TXD ใช้งานเป็นขาส่งสัญญาณของการ สื่อสารอนุกรมแบบ UART

- RXD – PORTD, Bit 0

ขา RXD ใช้งานเป็นขารับสัญญาณของการ สื่อสารอนุกรมแบบ UART

### 3.6 Timer & Counter

ภายใน ATMEGA162 จัดให้มี Timer/Counter 4 ชุด โดยจัดเป็น Timer/Counter ขนาด 8 บิต 2 ชุด และ Timer/Counter ขนาด 16 บิต 2 ชุด Timer/Counter2 สามารถรับสัญญาณ CLOCK จากภายนอก ซึ่งเป็น Option ที่จะนำ Timer/Counter2 มาทำเป็น RTC โดยใช้ XTAL ที่มีค่าความถี่ เท่ากับ 32.768 KHz มาทำเป็นฐานเวลา Timer/Counter0 และ Timer/Counter1 ใช้วงจร Prescaling ขนาด 10 บิตรวมกัน ส่วน Timer/Counter2 ใช้วงจร Prescaling แยกจากกัน

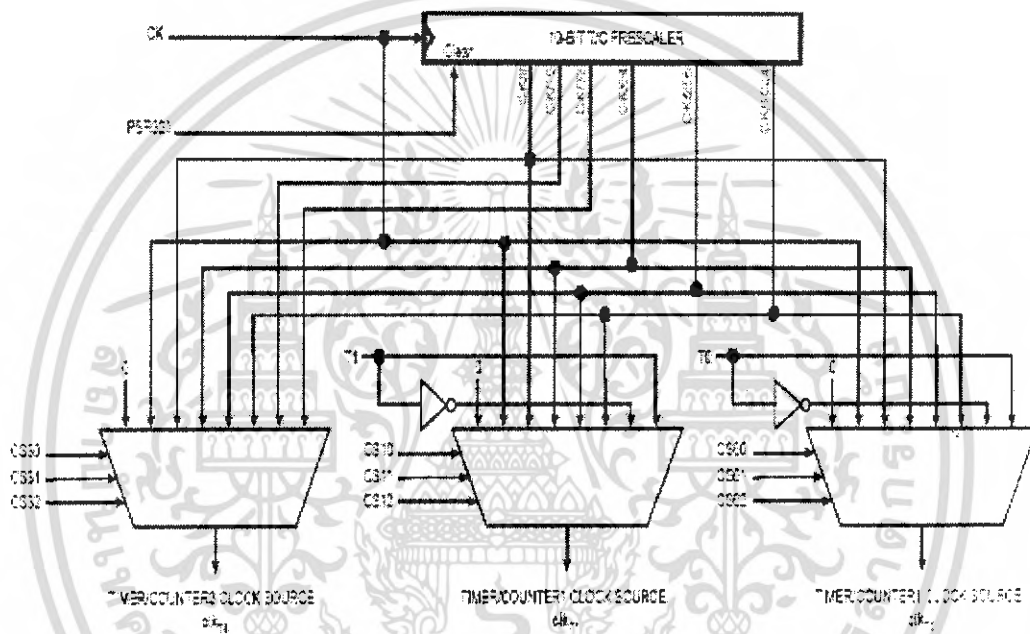
#### 3.6.1 TIMER0 & COUNTER 0

โครงสร้างของ Timer/Counter 0 ขนาด 8 บิต แสดงในรูปที่ ซึ่งสามารถเลือก สัญญาณ CLOCK ได้จาก CK (Clock ของระบบ) หรือสัญญาณ CLOCK ของระบบที่ถูกหาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Prescaling) หรือ สัญญาณจากภายนอก โดยการใช้งานจะอธิบายในรีจิสเตอร์ TCCR0 ซึ่งการควบคุมการอินเทอร์รัพท์จะควบคุมได้จากรีจิสเตอร์ TIMSK เมื่อ Timer/Counter0 ได้รับสัญญาณจากภายนอก ซึ่งสัญญาณดังกล่าว จะซิงโครไน (Synchronized) กับสัญญาณนาฬิกาภายใน CPU

โดย TIMER/COUNTER 0 จะเป็นวงจรมับขึ้นที่สามารถเขียนและอ่านข้อมูลได้ตลอดเวลา โดยเมื่อทำการเขียนข้อมูลลงใน TIMER/COUNTER 0 ในขณะที่มีสัญญาณ CLOCK จะทำให้ TIMER/COUNTER 0 นับค่าต่อเนื่องจากค่าที่ถูกเขียนลงไป



รูปที่ 3.12 โครงสร้างของ TIMER/COUNTER0

### 3.6.2 TIMER1 & COUNTER1

Timer/Counter1 จะมีขนาด 16 บิต โดยสามารถเลือกสัญญาณนาฬิกาได้จาก CK หรือ สัญญาณที่ได้รับจากการหาร CK (Prescaling) ซึ่งการหยุด Timer1/Counter1 จะอธิบายไว้ในรีจิสเตอร์ TCCR1A (Timer/Counter1 Control Register) และ TCCR1B โดยแฟร็กที่แสดงสถานะต่าง ๆ (Overflow, Compare math, Capture even) ส่วนสัญญาณควบคุมจะอธิบายไว้ในรีจิสเตอร์ TCCR1A และ TCCR1B การควบคุมสัญญาณอินเทอร์รัพท์จะควบคุมได้จากรีจิสเตอร์ TIMSK (TIMER / COUNTER INTERRUPT MASK REGISTER) เมื่อ TIMER1/COUNTER1 ถูกกำหนดให้ใช้สัญญาณนาฬิกาจากภายนอก สัญญาณดังกล่าวจะถูกซิงโครไน (Synchronized) กับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณนาฬิกาภายใน CPU

TIMER1/COUNTER1 จะประกอบด้วยส่วนของการเปรียบเทียบเอาต์พุต (Output Compare Function) 2 ฟังก์ชัน โดยจะใช้รีจิสเตอร์ OCR1A (Output Compare Register 1A) และ OCR1B (Output Compare Register 1B) เป็นส่วนหนึ่ง ของการเก็บค่าข้อมูลของการเปรียบเทียบ TIMER1/COUNTER1 จะสามารถเลือกใช้ฟังก์ชัน PWM ได้ทั้ง 8, 9 และ 10 บิต



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีนำไปใช้

## บทที่ 4

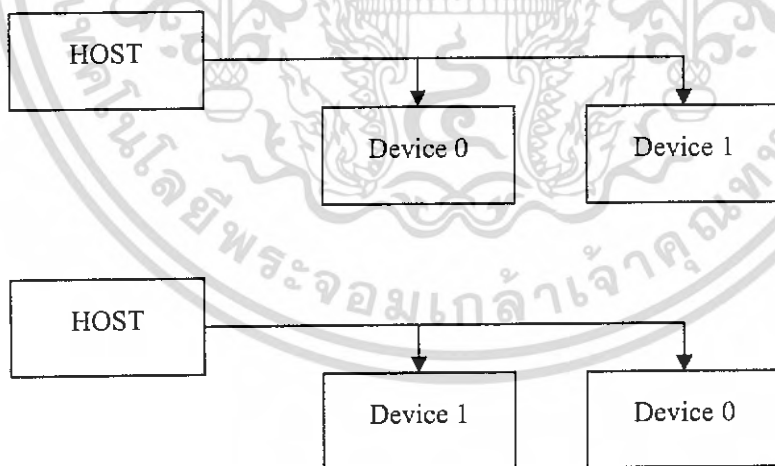
### โปรโตคอลที่ใช้ในการเชื่อมต่อกับซีดี-รอม

ซีดี-รอมที่ผลิตขึ้นมาใช้กับเครื่องคอมพิวเตอร์ในปัจจุบันนั้นมีการเชื่อมต่อ เป็นไปตามมาตรฐานATAPI ซึ่งเป็นมาตรฐานที่เพิ่มเติมมาจากมาตรฐาน ATA ซึ่งเป็นมาตรฐานในการเชื่อมต่อฮาร์ดดิสก์เข้ากับเครื่องคอมพิวเตอร์ โดยการเชื่อมต่ออุปกรณ์ตามมาตรฐานATAPI จะเหมือนกับมาตรฐาน ATA ทั้งหมด จะมีส่วนที่แตกต่างก็คือการส่งคำสั่งเข้าไปยังตัวอุปกรณ์จะต้องส่งเป็นชุดคำสั่ง

#### 4.1 ATA

##### 4.1.1 การเชื่อมต่อทางกายภาพ

ตามมาตรฐาน ATA สามารถนำอุปกรณ์มาเชื่อมต่อกันได้ 1-2 ตัว โดยอุปกรณ์นั้นๆต้องเชื่อมต่อกับส่วนที่ทำหน้าที่ควบคุมมัน หรือที่เรียกว่าโฮสต์อะแดปเตอร์ (Host adapter) และถ้าอุปกรณ์ที่เชื่อมต่อมี 2 ตัว อุปกรณ์ทั้งสองจะต้องต่อเชื่อมกันแบบ daisy chain configuration ซึ่งอุปกรณ์ตัวหนึ่งจะถูกกำหนดเป็น Device0 และอีกตัวหนึ่งเป็น Device1



รูปที่ 4.1 การเชื่อมต่อระหว่างตัวแม่กับอุปกรณ์ 2 ตัว

การกำหนดว่าอุปกรณ์ตัวใดเป็น Device0 หรือ Device1 นั้นอาจทำได้หลายวิธี

- สวิตช์หรือจัมเปอร์ (jumper) บนอุปกรณ์
- การใช้ขา Cable Select (CSEL)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการเชื่อมต่ออุปกรณ์ 2 ตัวนั้นตำแหน่งในการต่อสายไฟจะไม่มีผลต่อการเชื่อมต่อ ถ้ามีอุปกรณ์ที่เชื่อมต่อแบบ ATA เพียงตัวเดียว แล้วโฮสต์กับอุปกรณ์ควรจะอยู่ที่ปลายทั้ง 2 ด้านของสายไฟ



รูปที่ 4.2 การเชื่อมต่อระหว่างตัวแม่กับอุปกรณ์ตัวเดียว

**4.1.2 ขาเชื่อมต่อสำหรับรับส่งข้อมูล**

ตามมาตรฐาน ATA ขาเชื่อมต่อระหว่างอุปกรณ์และตัวแม่เป็นตัวเชื่อมต่อขนาด 40 ขา ซึ่งมีการจัดวางตำแหน่งตามรูปที่ 4.3



(ก) แผนผังขาทางด้านของตัวอุปกรณ์

(ข) แผนผังขาทางด้านของสายเคเบิล

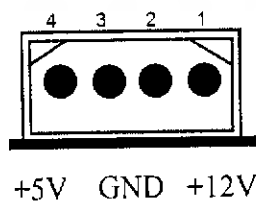
รูปที่ 4.3 ตัวเชื่อมต่อขนาด 40 ขา ตามมาตรฐาน ATA

จากจำนวนขาทั้งหมด 40 ขา เป็นส่วนของขาส่งข้อมูลขนาด 16 บิตอยู่ 16 ขา ส่วนที่ทำหน้าที่กำหนดตำแหน่งรีจิสเตอร์ที่จะเขียนอีก 5 ขา ส่วนที่ใช้ควบคุมการส่งข้อมูลในแบบ DMA 2 ขา ขารีเซท 1 ขา และส่วนที่เหลือเป็นขาสำหรับควบคุมการทำงานต่างๆของตัวอุปกรณ์ เช่น สัญญาณอ่าน-เขียน สัญญาณอินเตอร์รัพต์ รายละเอียดหน้าที่ของขาแต่ละขาแสดงได้ตามตารางที่ 4.1 เป็นต้น

ในส่วนของไฟเลี้ยงอุปกรณ์ที่เป็นไปตามมาตรฐาน ATA นั้นจะเป็นตัวเชื่อมต่อขนาด 4 ขา ดังรูปที่ 4.4 โดยจะมีไฟเลี้ยงขนาดสองชุด ชุดหนึ่งจ่ายไฟตรง 12 V อีกชุดหนึ่งจ่ายไฟตรง 5 V

ขาที่	ชื่อขา	หน้าที่	ขาที่	ชื่อขา	หน้าที่
1	- RESET	Reset	21	N/C	Not Connect
2	GND	Ground	22	GND	Ground
3	DD7	DATA7	23	-IOW	Write Strobe
4	DD8	DATA8	24	GND	Ground
5	DD6	DATA6	25	-IOR	Read Strobe
6	DD9	DATA9	26	GND	Ground
7	DD5	DATA5	27	IORDY	I/O Ready
8	DD10	DATA10	28	ALE	Address Latch
9	DD4	DATA4	29	N/C	Not Connect
10	DD11	DATA11	30	GND	Ground
11	DD3	DATA11	31	IRQ	Interrupt
12	DD12	DATA12	32	-IOCS	Chip Select
13	DD2	DATA2	33	DA1	Address
14	DD13	DATA13	34	N/C	Not connect
15	DD1	DATA1	35	DA0	Address
16	DD14	DATA14	36	DA2	Address
17	DD0	DATA0	37	-CS0	(1F0-1F7)
18	DD15	DATA15	38	-CS1	(3F6-3F7)
19	GND	Ground	39	-ACTIVE	LED Driver
20	KEY	Key	40	GND	Ground

ตารางที่ 4.1 รายละเอียดและหน้าที่ของแต่ละขาตามมาตรฐาน IDE



รูปที่ 4.4 ลักษณะของหัวไฟเลี้ยงของเครื่องเล่นซีดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 4.1.3 รีจิสเตอร์ภายใน

ตามมาตรฐาน ATA นั้น การสั่งงานเพื่อควบคุมการทำงานของอุปกรณ์จะต้องกระทำผ่านรีจิสเตอร์ ต่างๆ ภายในตัวอุปกรณ์นั้นๆ โดยการกำหนดรีจิสเตอร์ที่จะใช้งานนั้นจะกำหนดผ่านสายสัญญาณทั้ง 5 แล้ว การเขียนหรืออ่านรีจิสเตอร์ในตำแหน่งเดียวกันนั้นยังเป็นการอ้างอิงรีจิสเตอร์คนละตัวกันด้วย

รีจิสเตอร์ทั้งหมดสามารถแบ่งออกได้เป็นสองส่วนใหญ่ๆคือ

-รีจิสเตอร์ Command Block จะถูกใช้เพื่อส่งคำสั่งไปยังอุปกรณ์หรือส่งสถานะของอุปกรณ์กลับมายังโฮสต์

- รีจิสเตอร์ Control Block จะถูกใช้เพื่อให้อุปกรณ์เข้าควบคุม และส่งค่าสถานะสำรอง (alternate status)

ตำแหน่งของรีจิสเตอร์ต่างๆตามมาตรฐาน ATA เป็นไปตามตาราง 3.2 รีจิสเตอร์แต่ละตัวมีขนาด 16 บิต และมีหน้าที่การทำงานที่ต่างกันไป ต่อไปจะขอกล่าวถึงหน้าที่การทำงานของรีจิสเตอร์บางตัวที่สำคัญ

แอดเดรส					หน้าที่	
CS0-	CS1-	DA2	DA1	DA0	READ(DIOR-)	WRITE(DIOW-)
N	N	X	X	X	บัสข้อมูลอยู่ในสถานะอิมพีแดนซ์สูง	ไม่ใช่
รีจิสเตอร์ Control Block						
N	A	0	X	X	บัสข้อมูลอยู่ในสถานะอิมพีแดนซ์สูง	ไม่ใช่
N	A	1	0	X	บัสข้อมูลอยู่ในสถานะอิมพีแดนซ์สูง	ไม่ใช่
N	A	1	1	0	Alternate Status	Device Control
N	A	1	1	1	ไม่ใช่	ไม่ใช่
รีจิสเตอร์ Command Block						
A	N	0	0	0	Data	Data
A	N	0	0	1	Error	Error
A	N	0	1	0	Sector Count	Sector Count
A	N	0	1	1	Sector Number	Sector Number
A	N	1	0	0	Cylinder Low	Cylinder Low
A	N	1	0	1	Cylinder High	Cylinder High
A	N	1	1	0	Device/Head	Device/Head

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

A	N	1	1	1	Status	Status
A	A	X	X	X	Invalid Address	Invalid Address

A=signal asserted, N=signal negated, x=don't care

#### ตารางที่ 4.2 แอคเตสของรีจิสเตอร์ตามมาตรฐาน ATA

-รีจิสเตอร์ข้อมูล (Data Register) เป็นตำแหน่งของรีจิสเตอร์ที่ทำหน้าที่ส่งข้อมูลจากตัวอุปกรณ์ออกไปยัง โฮสต์ และรับข้อมูลจากโฮสต์เข้ามา

-รีจิสเตอร์ Error เก็บสถานะของการทำงานคำสั่งล่าสุด ใช้สำหรับตรวจสอบความผิดพลาดในการทำงาน

-รีจิสเตอร์ Device/Head ใช้กำหนดตัวอุปกรณ์ที่จะใช้งานเนื่องจากการต่ออุปกรณ์ตามมาตรฐาน ATA สามารถต่อได้ 2 ตัว จึงต้องมีการเลือกอุปกรณ์ที่ใช้งานด้วย

-รีจิสเตอร์สถานะ (Status Register) เก็บสถานะปัจจุบันของตัวอุปกรณ์

-รีจิสเตอร์คำสั่ง (Command Register) เป็นตำแหน่งรีจิสเตอร์ที่ใช้เขียนคำสั่งเข้ามาเพื่อควบคุมการทำงานของตัวอุปกรณ์

รีจิสเตอร์ที่กล่าวมาข้างต้น บางตัวจะมีการกำหนดหน้าที่การทำงานเป็นบิตๆ โดยแต่ละบิตแยกกันอย่างอิสระ

#### 4.1.4 โปรโตคอลการส่งข้อมูล

การควบคุมอุปกรณ์ตามมาตรฐาน ATA นั้นต้องส่งข้อมูลเข้าไปยังรีจิสเตอร์ต่างๆ ของตัวอุปกรณ์นั้นๆ โดยการส่งข้อมูลเข้าไปยังรีจิสเตอร์ภายในนั้นจำเป็นต้องมีลำดับในการเขียนรีจิสเตอร์ต่าง ๆ อย่างถูกต้องจึงจะสามารถส่งคำสั่งได้อย่างถูกต้อง โปรโตคอลในการส่งคำสั่งเพื่อควบคุมการทำงานของอุปกรณ์ สามารถแบ่งออกได้เป็นประเภทต่าง ๆ ตามข้อมูลที่อ่านออกมาจากตัวอุปกรณ์ซึ่งสามารถแบ่งออกได้เป็น 3 ประเภท

- คำสั่งที่ไม่มีการส่งข้อมูลกลับจากตัวอุปกรณ์
- คำสั่งที่มีการส่งข้อมูลกลับในแบบ PIO
- คำสั่งที่มีการส่งข้อมูลกลับในแบบ DMA

โปรโตคอลทั้งสามจะมีส่วนที่คล้ายกันและต่างกันในบางส่วน ซึ่งจะขอกล่าวแยกเป็นชนิดๆ ดังนี้

#### 4.1.4.1 โพรโทคอลการส่งคำสั่งที่ไม่มีการส่งข้อมูลกลับ

ลำดับการส่งคำสั่งที่ไม่มีการส่งข้อมูลจากตัวอุปกรณ์กลับมายังโฮสต์เป็นไปดังนี้

1. โฮสต์ต้องเช็คค่าบิต BSY ภายในรีจิสเตอร์สถานะ (Status Register) จนกว่าบิต BSY จะเท่ากับ 0

2. เขียนรีจิสเตอร์ Device/Head เพื่อเลือกอุปกรณ์ที่ต้องการควบคุม

3. ตรวจสอบค่าภายในรีจิสเตอร์สถานะ (Status Register) โดยรอจนกว่าบิต BSY จะเท่ากับ 0 และบิต DRDY จะเท่ากับ 1

4. เขียนพารามิเตอร์ต่างๆที่แต่ละคำสั่งต้องการลงในรีจิสเตอร์ต่างๆ ตามความต้องการของแต่ละคำสั่ง

5. เขียนคำสั่งที่ต้องการลงในรีจิสเตอร์คำสั่ง (Command Register)

6. เมื่ออุปกรณ์ได้รับการเขียนรีจิสเตอร์คำสั่ง ตัวอุปกรณ์จะเซตบิต BSY เป็น 1 และปฏิบัติตามคำสั่งที่สั่ง

7. เมื่อทำงานตามคำสั่งนั้นๆเสร็จสิ้นอุปกรณ์จะเคลียร์ค่าบิต BSY และส่งสัญญาณอินเทอร์รัพท์กลับมายังโฮสต์

8. กรณีที่เกิดความผิดพลาดในการส่งข้อมูลจะรายงานความผิดพลาดในรีจิสเตอร์ Error

#### 4.1.4.2 โพรโทคอลการส่งคำสั่งที่มีการส่งข้อมูลกลับในแบบ PIO

การส่งคำสั่งที่มีการส่งข้อมูลกลับในแบบ PIO นี้จะมีส่วนที่เหมือนกับการส่งข้อมูลที่ไม่มีการส่งข้อมูลกลับตั้งแต่ลำดับที่ 1 ถึงลำดับที่ 5

1. โฮสต์ต้องเช็คค่าบิต BSY ภายในรีจิสเตอร์สถานะ (Status Register) จนกว่าบิต BSY จะเท่ากับ 0

2. เขียนรีจิสเตอร์ Device/Head เพื่อเลือกอุปกรณ์ที่ต้องการควบคุม

3. ตรวจสอบค่าภายในรีจิสเตอร์สถานะ (Status Register) โดยรอจนกว่าบิต BSY จะเท่ากับ 0 และบิต DRDY จะเท่ากับ 1

4. เขียนพารามิเตอร์ต่างๆที่แต่ละคำสั่งต้องการลงในรีจิสเตอร์ต่างๆ ตามความต้องการของแต่ละคำสั่ง

5. เขียนคำสั่งที่ต้องการลงในรีจิสเตอร์คำสั่ง (Command Register)

6. อุปกรณ์จะเซตบิต BSY และปฏิบัติตามคำสั่งที่ส่งเข้าไป และเตรียมส่งข้อมูลชุดแรกไปยังโฮสต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.เมื่อพร้อมที่จะส่งข้อมูลออกมา อุปกรณ์จะเซตบิต DRQ เคลียร์บิต BSY และส่งสัญญาณอินเทอร์รัพท์กลับมายังโฮสต์

8.โฮสต์อ่านข้อมูลจากตัวอุปกรณ์

9.เซตบิต DRQ ภายในรีจิสเตอร์สถานะ ถ้าบิต DRQ เท่ากับ 1 หมายถึงยังมีข้อมูลเวิร์ดถัดไปที่จะส่งออกมายังโฮสต์ แต่ถ้าหมดข้อมูลที่จะส่งแล้วบิต DRQ จะเท่ากับ 0 และบิต BSY จะเท่ากับ 0

#### 4.1.4.3 โพรโตคอลการส่งคำสั่งที่มีการส่งข้อมูลกลับในแบบ DMA

การส่งคำสั่งที่มีการส่งข้อมูลกลับในแบบ DMA นี้จะคล้ายกับการส่งข้อมูลที่ส่งข้อมูลที่ไม่มีการส่งข้อมูลกลับตั้งแต่ลำดับ 1 ถึง 5 เช่นเดียวกับการส่งข้อมูลกลับแบบ PIO แต่หลังจากนั้นการส่งข้อมูลจากตัวโฮสต์จะเป็นไปตามมาตรฐานการส่งข้อมูลแบบ DMA ซึ่งในงานนี้ไม่ได้ใช้โปรโตคอลนี้จึงไม่ขอกล่าวละเอียด

## 4.2 ATAPI

มาตรฐาน ATAPI เป็นมาตรฐานที่อยู่บนมาตรฐาน ATA อีกทีหนึ่ง จากเดิมที่การควบคุมอุปกรณ์ในมาตรฐาน ATA เพียงเวิร์ดเดียวในการสั่งงาน ในมาตรฐาน ATAPI นี้จะเปลี่ยนรูปแบบการส่งคำสั่งจากเวิร์ดเดียวเป็นแพ็คเกจขอคำสั่งแทน เพื่อความยืดหยุ่นในการสั่งงานอุปกรณ์ แต่ถึงอย่างไรก็ตามมาตรฐาน ATAPI เป็นมาตรฐานที่อยู่บนมาตรฐาน ATA ดังนั้นการเชื่อมต่อขาเชื่อมต่อ (Connector) และลำดับการส่งสัญญาณจะเหมือนกับมาตรฐาน ATA ทั้งหมด ส่วนที่แตกต่างไปบ้างคือ รีจิสเตอร์ภายในอุปกรณ์ที่มีการเปลี่ยนแปลงหน้าที่การทำงานไปบางส่วน และส่วนที่แตกต่างกับมาตรฐาน ATA อย่างชัดเจนก็คือ โพรโตคอลที่ใช้ส่งคำสั่งควบคุมอุปกรณ์

### 4.2.1 ATAPI Protocol

ATAPI Device จะถูกสั่งด้วยวิธีการ 2 วิธีด้วยกัน คือ

1. คำสั่งพื้นฐานของมาตรฐาน ATA
2. คำสั่งแบบเป็นแพ็คเกจ (Packet Command)

ทั้งสอง 2 วิธีอุปกรณ์จะคอยรับคำสั่งจาก Host เพื่อปฏิบัติตามและจะแจ้งสถานะของตัวอุปกรณ์ให้กับ Host เมื่อทำคำสั่งเสร็จ แต่ถ้ามีอุปกรณ์มากกว่า 1 ตัว ต่อพ่วงกันอยู่ คำสั่งจะถูกเขียนไปสู่ทุกตัวและจะมีเฉพาะคำสั่งเลือกอุปกรณ์เท่านั้นที่จะทำ โดยที่ Device 1 จะแจ้งสถานะของตัวมันสู่ Device 0 ผ่านทางบิต PDIAG “The Protocol” ของ ATAPI ก็คือการใช้คำสั่ง ATA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบใหม่ที่เรียกว่า “ATAPI Packet Command” โพรโตคอลของ ATA ทั้งหมดจะถูกส่งเป็นแบบ Packet Command แต่เมื่อมีการส่งคำสั่งออกไปแล้วจะมีกฎใหม่ ๆ เพิ่มเข้ามา

1. บิต DRQ ที่อยู่ใน Status Register จะถูกนำมาใช้ร่วมกับ Interrupt Reason Register เพื่อกำหนด Interrupt Type

2. คำสั่งของการจัดการกับตัวอุปกรณ์จะถูกส่งเป็น Packet ผ่าน Data Register

3. คำสั่งที่ส่งไปโดย Packet Command จะเหมือนกับของ Task File

4. Byte Count จะถูกใช้เพื่อกำหนดปริมาณข้อมูลที่ Host จะถ่ายโอนไปในแต่ละ DRQ Interrupt

5. The ATAPI Feature Register จะถูกใช้เพื่อบอกเมื่อ DMA ถูกใช้ด้วยการใช้ฮอปโคด (Opcode) ที่ต่างกัน

6. สถานะสุดท้ายที่จะชี้บอกกับตัว Host จะส่งเป็น interrupt หลังจากข้อมูลสุดท้ายได้ถูกถ่ายโอนไป โพรโตคอลที่ว่ามานี้จะถูกเพิ่มเข้ามาเมื่อมีการส่ง Packet Command ออกไปแล้วเท่านั้นจนกระทั่ง Host ได้อ่าน Completion Status หลังจาก Host ได้ทำการอ่าน Completion Status แล้ว Task File Register และ Protocol ต่างๆ จะกลับเข้าสู่รูปแบบมาตรฐานของ ATA

#### 4.2.2 ATAPI PACKET Command

ATAPI Packet Command จะคล้ายกับคำสั่งของ ATA ปกติ โดยการเริ่มต้นด้วย The Task Register และการเซต Drive Selection Bit และการเขียนไบต์คำสั่งเข้าไปใน Command Register ด้วยคำสั่ง ATA ปกติ บิต DRQ จะถูกสร้างขึ้นมาเพื่อบอกว่าข้อมูลสำหรับคำสั่งควรจะถูกส่งเข้าไปหรือออกมาจากตัวอุปกรณ์ แต่ถ้าเป็นรูปแบบ Packet Command บิต DRQ จะถูกสร้างขึ้นมาเพื่อบอกว่า Command Packet Data ควรจะถูกเขียนเข้าไปในตัวอุปกรณ์ในการส่งแบบ ATAPI Packet Command สามารถที่จะส่งคำสั่งออกไปได้โดยไม่ต้องสนใจสถานะของบิต DRDY Status

#### 4.2.3 Status Register for Packet Command

การติดต่อเข้าไปหรือออกมาจากตัวอุปกรณ์จะกระทำผ่าน I/O Register ซึ่งจะทำการผ่านรีจิสเตอร์ที่ถูกเลือกโดยการในใส่สัญญาณจาก Host (CS0-, CS1-, DA2, DA1, DA0, DIOR-, DIOW-) ซึ่งจะแสดงได้ดังตารางที่ 5.3

แอดเดรส					หน้าที่	
CS0-	CS1-	DA2	DA1	DA0	READ(DIOR-)	WRITE(DIOW-)
รีจิสเตอร์ Control Block						
N	A	0	0	0	สถานะฟลอปปี A	ไม่ใช่
N	A	0	0	1	สถานะฟลอปปี A	ไม่ใช่
N	A	0	1	0	ไม่ใช่	Floppy Digital Output
N	A	0	1	1	ควบคุม Floppy ID/Tape	สงวน
N	A	1	0	0	สถานะ Floppy Controller	สงวน
N	A	1	0	1	รีจิสเตอร์บล็อกคำสั่ง	
N	A	1	1	0	สถานะ Alternate ATAPI	ควบคุมอุปกรณ์
N	A	1	1	1	ไม่ใช่	ไม่ใช่
รีจิสเตอร์ Command Block						
A	N	0	0	0	Data	
A	N	0	0	1	Error	Error
A	N	0	1	0	Sector Count	Sector Count
A	N	0	1	1	Sector Number	Sector Number
A	N	1	0	0	Cylinder Low	Cylinder Low
A	N	1	0	1	Cylinder High	Cylinder High
A	N	1	1	0	Device/Head	Device/Head
A	N	1	1	1	Status	Status
A	A	X	X	X	Invalid Address	Invalid Address

ตารางที่ 4.3 รายละเอียดเกี่ยวกับรีจิสเตอร์ต่างๆที่ใช้ใน ATAPI

ใน ATAPI Register จะอ้างอิงด้วยการใช้ขนาด 8 บิตทุกตัว ยกเว้น Data Register จะใช้ขนาดในการติดต่อเป็น 16 บิตเราจะทำการอธิบายหน้าที่ต่างๆของรีจิสเตอร์แต่ละส่วนดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.2.3.1 ATAPI Device Control Register

D7	D6	D5	D4	D3	D2	D1	D0
Reserved				1	SRST	nIEN	0

- บิต 4 – 7: สงวนไว้เพื่อใช้ในอนาคต
- บิต 3 : เป็น 1 เสมอ
- บิต 2 : SRST บิตนี้คือ Software Reset ซึ่งขั้นตอนในการรีเซ็ตจะให้เป็นขั้นตอนซึ่งจะได้อธิบายไว้ในตอนต่อไป
- บิต 1 : nIEN บิตนี้จะเป็นตัว Enable/Disable ของ Interrupt ที่จะมีถึง Host  
เมื่อ nIEN = 0: ขา INTRQ จะอยู่ในสถานะ Tri-State Buffer  
เมื่อ nIEN = 1: ขา INTRQ จะอยู่ในสถานะ High Impedance
- บิต 0 : เป็น 0 เสมอ

#### 4.2.3.2 ATAPI Error Register

D7	D6	D5	D4	D3	D2	D1	D0
Sense Key				MCR	ABRT	EOM	ILI

- บิต 4 – 7: Sense Key ซึ่งจะได้กล่าวต่อไป
- บิต 3 : MCR (Media Change Requested) ถูกใช้ตามมาตรฐาน ATA
- บิต 2 : ABRT (Aborted Command) ถูกใช้ตามมาตรฐาน ATA
- บิต 1 : EOM (End of Media Detected)
- บิต 0 : ILI (Illegal Length Indication)

#### 4.2.3.3 ATAPI Feature Register

D7	D6	D5	D4	D3	D2	D1	D0
Reserved						Overlap	DMA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- บิต 2 – 7: สงวนไว้เพื่อใช้ในอนาคต  
 บิต 1 : Overlap (Optional)  
 บิต 0 : DMA (Optional) บางคำสั่งจะถ่ายโอนข้อมูลผ่านการเชื่อมต่อแบบ DMA แต่บิตนี้จะไม่ใช่ในการคำสั่งแบบแพ็คเกจ (Command Packet)

#### 4.2.3.4 ATAPI Interrupt Reason Register

D7	D6	D5	D4	D3	D2	D1	D0
Reserved					Release	IO	CoD

- บิต 3 – 7: สงวนไว้เพื่อใช้ในอนาคต  
 บิต 2 : RELEASE เป็นตัวบอกว่าอุปกรณ์ได้มีการปล่อย ATA bus ก่อนที่คำสั่งจะทำเสร็จ  
 บิต 1 : IO เป็นรีจิสเตอร์ที่ชี้บอกถึงทิศทางในการโอนถ่ายข้อมูลว่าข้อมูลเข้าไปใน Host หรือออกจาก Host  
 โดยถ้า IO = 1 แสดงว่าข้อมูลไหลเข้าไปใน Host  
 IO = 0 แสดงว่าข้อมูลไหลออกจาก Host  
 บิต 0 : CoD (Command or Data)  
 โดยถ้า CoD = 1 แสดงว่าข้อมูลที่โอนถ่ายกันเป็น คำสั่ง (Command)  
 CoD = 0 แสดงว่าข้อมูลที่โอนถ่ายกันเป็นข้อมูล (Data)

#### 4.2.3.5 ATAPI Byte Count Register

D7	D6	D5	D4	D3	D2	D1	D0
Byte Count (bits 0-7)							
Byte Count (bits 8-15)							

Byte Count จะใช้ใน PIO Mode เท่านั้น การกระทำคำสั่งจะทำการเช็คก่อนที่จะมีการส่งคำสั่งแพ็คเกจคำสั่งออกไป (Command Packet) ในรีจิสเตอร์นี้จะประกอบไปด้วยขนาดของคำสั่งทั้งหมดที่จะมีการโอนถ่ายเพียงหนึ่งกลุ่มข้อมูลเท่านั้น เช่น คำสั่ง Mode Select/Sense สำหรับคำสั่งที่ต้องการ DRQ Interrupt หลายๆครั้ง เช่น คำสั่ง Read / Write จะทำโดยการเช็คขนาดที่เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต้องการ เมื่อมีการส่งข้อมูลตัวอุปกรณ์จะทำการเซต Byte Count Register ให้มีขนาดเท่ากับจำนวนข้อมูลที่ Host จะทำการโอนถ่ายและจะทำการปล่อย DRQ Interrupt ข้อมูลภายในของรีจิสเตอร์จะไม่เปลี่ยนแปลงจนกว่าข้อมูลเวิร์ดแรกจะทำส่งเข้าไปหรือออกจาก Data Register เมื่อเริ่มมีการส่งข้อมูลตัวอุปกรณ์จะทำการเปลี่ยนขนาดของ byte count

#### 4.2.3.6 ATAPI Drive Select Register

D7	D6	D5	D4	D3	D2	D1	D0
1	Reserved	1	DRV	Reserved for SAM LUN			

บิต 4 : เป็นการเลือกตัวอุปกรณ์ว่าจะไปไหน โดยถ้า DRV = 0: แสดงว่าต้องการติดต่อกับ Device 0 โดยถ้า DRV = 1: แสดงว่าต้องการติดต่อกับ Device 1

#### 4.2.3.7 ATAPI Status Register

D7	D6	D5	D4	D3	D2	D1	D0
BSY	DRDY	DMA/DF	SERVICE/DSC	DRQ	CORR	Reserved	CHECK

บิต 7 : BSY (Busy) บิตนี้จะถูกเซตเมื่อตัวอุปกรณ์มีการเข้าถึงข้อมูลภายใน Command Block

บิต 6 : DRDY เป็นบิตที่บอกว่าตัวอุปกรณ์มีการตอบสนองต่อคำสั่ง ATA

บิต 5 : DMA READY / DF บิตนี้จะเป็นตัวบอกว่าตัวอุปกรณ์นั้นพร้อมที่จะทำการโอนถ่ายข้อมูลในโหมด DMA บิตนี้จะถูกสงวนไว้ใช้ในอนาคตเพื่อเพิ่มความสามารถในการ Overlap

บิต 4 : SERVICE / DSC บิตนี้จะเป็นตัวบอกว่าตัวอุปกรณ์กำลังร้องขอ Service หรือ Interrupt และจะไม่เคลียร์บิตนี้จนกว่าจะได้รับคำสั่ง Service Command

บิต 3 : DRQ (Data Request) เป็นบิตที่บอกว่าตัวอุปกรณ์พร้อมที่โอนถ่ายข้อมูลระหว่าง Host กับตัวอุปกรณ์ ข้อมูลใน ATAPI Interrupt Reason จะมีการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

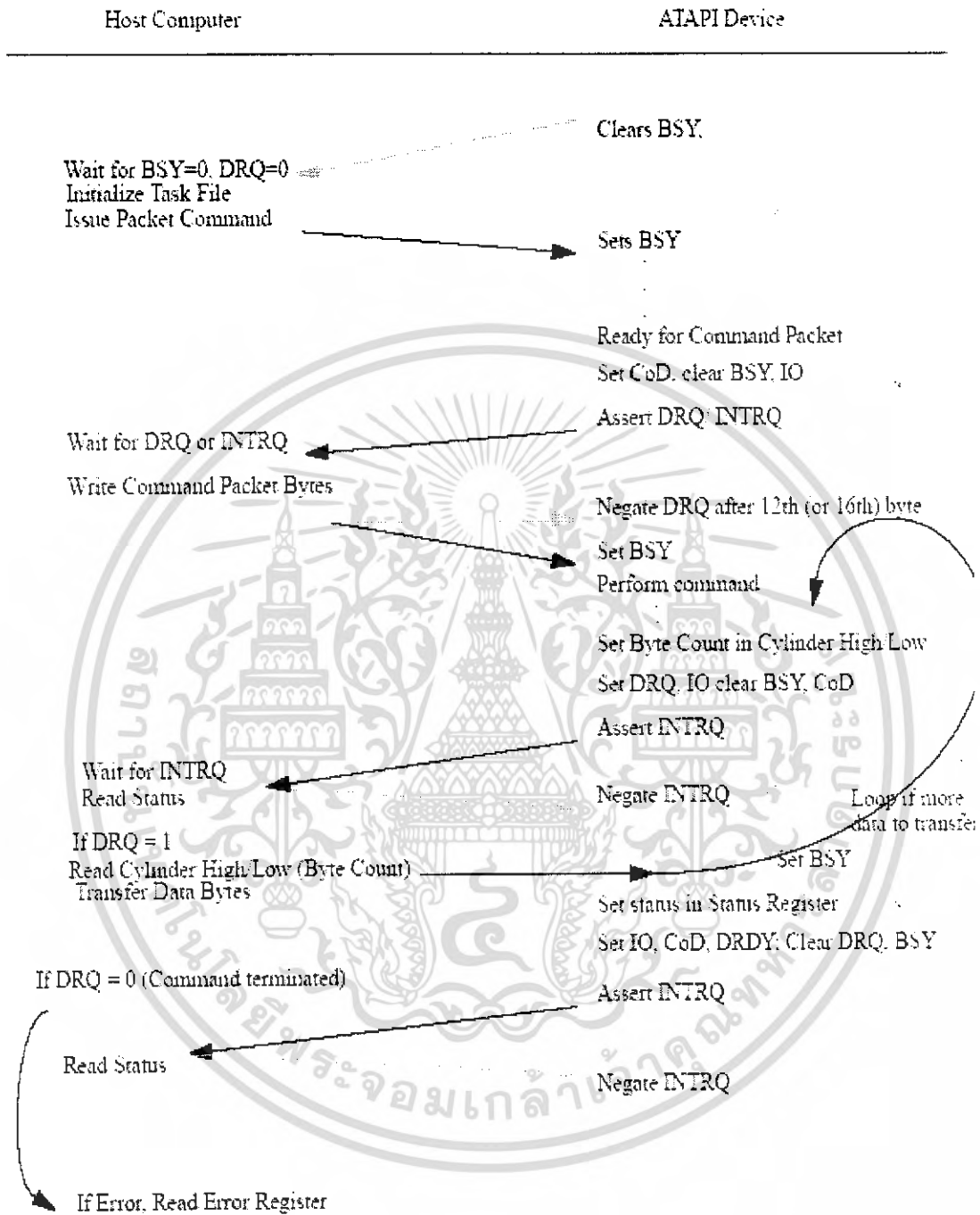
- เปลี่ยนแปลงในระหว่างที่มี Packet Command และ เมื่อ DRQ = 1
- บิต 2 : CORR บิตนี้จะเป็นตัวบอกว่ามีการเกิด Correctable Error
- บิต 1 : สงวนไว้เพื่อใช้ในอนาคต
- บิต 0 : CHECK จะเป็นตัวบอกว่ามีการเกิด Error ขึ้นในระหว่างที่มีการประมวลผลของคำสั่งก่อนหน้านี้

#### 4.3 ขั้นตอนการส่งคำสั่งแบบ PIO ที่มีข้อมูลเข้าสู่ Host

ขั้นตอนเหล่านี้จะประกอบด้วยคำสั่งเช่น คำสั่ง Inquiry, Read และอื่นๆ

- Host จะต้องคอยเช็คค่า BSY = 0, DRQ = 0 แล้วก็เขียนพารามิเตอร์ต่างๆลงใน Feature, Byte Count และ Drive/Head Register
- Host จะทำการเขียน Packet Command Code (AOH) เข้าไปใน Command Register
- ตัวอุปกรณ์จะทำการเช็ค BSY และทำการเตรียมรับคำสั่ง Packet Command
- เมื่อตัวอุปกรณ์พร้อมที่จะรับคำสั่ง Packet Command ตัวอุปกรณ์จะทำการเช็คบิต CoD และเคลียร์บิต IO ส่วนบิต DRQ จะทำการเช็คขึ้นมาทันทีหรือก่อนที่บิต BSY จะเคลียร์ลง
- หลังจากเจอสัญญาณ DRQ แล้ว Host ก็ทำการเขียนคำสั่งจำนวน 12 ไบต์ลงใน Data Register
- หลังจากนั้นตัวอุปกรณ์จะทำการเคลียร์ DRQ (เมื่อคำสั่งไบต์ที่ 12 ได้เขียนไปแล้ว) และจะทำการเช็คบิต BSY และจะทำการอ่าน Feature และ Byte Count ที่ส่งมาจาก Host และตัวอุปกรณ์จะทำการรอรับข้อมูลที่จะทำการคอนถ่ายมา
- เมื่อมีข้อมูลเข้ามาตัวอุปกรณ์จะทำการใส่จำนวนไบต์ข้อมูลลงใน Byte Count Register และจะทำการเช็ค IO และเคลียร์ CoD และจะทำการเช็ค DRQ และเคลียร์ BSY และเช็ค INTRQ
- หลังจากเจอสัญญาณ INTRQ แล้ว Host จะทำการอ่าน DRQ เพื่อจะได้กำหนดคำสั่งต่อไปถ้า DRQ = 0 ตัวอุปกรณ์จะทำการยุติคำสั่ง แต่ถ้า DRQ = 1 แล้ว Host จะทำการอ่านค่าข้อมูล (จำนวนไบต์ใน Byte Count Register) ผ่านทาง Data Register
- ตัวอุปกรณ์จะเคลียร์ DRQ (ถ้ามีความต้องการจะโอนถ่ายข้อมูลอีก BSY จะเท่ากับ 1 และให้ปฏิบัติตามตั้งแต่ข้อ 7. ลงมา
- เมื่อตัวอุปกรณ์พร้อมที่จะแสดงสถานะของตัวเองตัวอุปกรณ์จะทำการใส่ค่า Completion Status เข้าไปใน Status Register, และจะทำการเช็ค CoD, IO, DRDY และเคลียร์ BSY, DRQ และปล่อยสัญญาณ INTRQ
- หลังจากเจอสัญญาณ INTRQ & DRQ = 0 Host จะทำการอ่าน Status Register

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 ขั้นตอนการส่งคำสั่งแบบ PIO ที่มีข้อมูลออกจาก Host

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

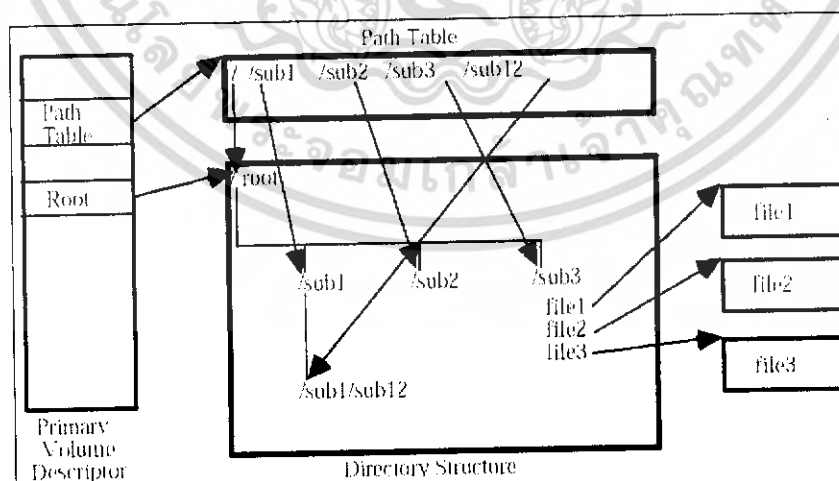
### ISO 9660

#### 5.1 บทนำ

คอมแพคดิสก์ (CD) ถูกเรียกได้ว่าเป็นผลิตภัณฑ์ที่ประสบความสำเร็จในตลาดผู้บริโภคอย่างสูงที่สุดเท่าที่เคยวางตลาดมา ตั้งแต่การแนะนำสินค้าชนิดนี้ในเดือนมิถุนายน 1980 ทำให้ CD มีอิทธิพลต่ออุตสาหกรรมดนตรีและเป็นทางเลือกสำหรับผู้ฟังดนตรีหลายล้านคน เนื่องจากความถูกต้องสูงเป็นพิเศษซึ่งเกิดจากเทคนิคการบันทึกแบบดิจิทัล และการแก้ไขข้อมูลนั้นสามารถทำได้ยากมากซึ่งเกิดจากการบันทึกแบบออปติคัล คุณสมบัติเหล่านี้ทำให้ CD มีเสน่ห์ที่น่าสนใจในการเป็นศูนย์กลางของข้อมูลดิจิทัล ลักษณะเด่นอื่นที่ทำให้ CD น่าสนใจ ก็คือการผลิตออกมาในปริมาณมากๆ ได้อย่างรวดเร็ว

แต่ในระยะแรกของการเกิดนั้นยังไม่มีมาตรฐานใดๆ เข้ามารับรองรูปแบบของการเขียนข้อมูลในตัวแผ่น ทำให้เกิดปัญหาในด้านความเข้ากันได้ของแผ่นซีดี ซึ่งเป็นปัญหาอย่างมากในด้านการพัฒนา จึงมีการจับมือกันระหว่างผู้ผลิตซีดีรอมหลายรายในการสร้างมาตรฐานร่วมกันให้เป็นมาตรฐานอุตสาหกรรมที่นิยมโดย Red Book, Yellow Book และ ISO 9660 ทำให้ปัจจุบันสามารถใช้ CD กับฮาร์ดแวร์และซอฟต์แวร์ได้เกือบจะทุกรูปแบบ

#### 5.2 ภาพรวมโครงสร้าง ISO9660



รูปที่ 5.1 โครงสร้าง ISO 9660

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงสร้างทางข้อมูลของ ISO 9660 แบ่งได้เป็น 3 ส่วนหลัก คือ Volume Descriptor, Directory Structure และ Path Table โครงสร้างเหล่านี้เกี่ยวข้องซึ่งกันและกันดังแสดงในรูปที่ 5.1 Volume Descriptor จะบอกตำแหน่งของ Directory Structure และ Path Table ไดรเรททอรีจะบอกตำแหน่งของไฟล์ และ Path Table จะเป็นวิธีลัดไปสู่แต่ละไดเรททอรี

### 5.2.1 The Volume Descriptor

มี Volume Descriptor อยู่ 4 ชนิดที่นิยามใน ISO 9660 คือ Primary Volume Descriptor, Boot Record, Supplementary Volume Descriptor และ Volume Partition Descriptor โดย Primary Volume Descriptor ถูกใช้ทั่วไป Boot Record ใช้สำหรับระบบที่ต้องแสดงบางอย่างของการตั้งค่าเริ่มต้น(initialization) ก่อนที่ใช้จะสามารถเข้าถึง Volume ได้ แม้ว่า ISO 9660 จะไม่กำหนดว่าข้อมูลอะไรบ้างที่ต้องอยู่ใน Boot Record หรือวิธีที่จะใช้ข้อมูลเหล่านั้น Supplementary Volume Descriptor สามารถใช้ระบุ alternate character set สำหรับระบบที่ไม่สนับสนุน ISO 646 character set ส่วน Volume Partition Descriptor สามารถใช้แบ่ง Volume ออกเป็น Volume ให้มีขนาดเล็กลง Volume Descriptor จะเริ่มการอัดที่ Logical Sector 16 (ซึ่งคอบสนองภายใน 2 วินาที และ 16เซคเตอร์ใน CD หรือใน CD "Atime", 00:02:16)

#### 5.2.1.1 Primary Volume Descriptor

Standard Identifier
Volume Identifier
Volume Set Identifier
System Identifier
Volume Size
Logical Block Size
Path Table Size
Root Directory Record
Identifier
Time Stamps

ตารางที่ 5.1 Primary Volume Descriptor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Primary Volume Descriptor เป็นจุดเริ่มต้นในการระบุ CD-ROM ประกอบด้วย Standard Identifier, Volume Identifier, Volume Set Identifier, System Identifier, Volume Size, Number of Volume in this set, Number of this Volume in the set, Size Logical Block, Size of the Path Table, Location of the Path Table, Root Directory Record, Other Identifiers และเวลาสำคัญที่เกี่ยวข้องกับ Volume

1. **Standard Identifier** เป็นกลุ่มของตัวอักษร และมีค่า CD001 ตามมาตรฐาน ISO 9660 ซึ่งจะบอกระบบปฏิบัติการว่านี่คือดิสก์ตาม ISO 9660 เพื่อที่จะแบ่งแยก Volume ออกจากระบบไฟล์อื่น โดยใช้แบบแผนง่ายๆ เช่น High Sierra ซึ่งมี Standard Identifier คือ CD-ROM และ Compact Disc Interactive ซึ่งมี Standard Identifier คือ CD-I

2. **Volume Identifier** ตัวอักษรที่ใช้จะถูกกำหนดโดย ISO 9660 ซึ่งเรียกว่า d-character และไม่สามารถยาวเกิน 31 ตัวอักษรได้

ABCDEFGHIJKLMN OPQR  
STUVWXYZ0123456789\_

รูปที่ 5.2 d-characters

3. **Volume Set Identifier** เป็นการระบุชื่อของ Multiple Volume ที่ Volume นั้นอยู่ Volume Set Identifier จะเหมือนกับ Volume Identifier ตรงที่จะถูกระบุโดย d-character และไม่สามารถยาวเกิน 31 ตัวอักษรได้ ตัวอย่างเช่น ถ้า Volume นี้ชื่อ DICTIONARY\_E\_H ก็อาจจะมี Volume Set Identifier เป็น DICTIONARY ซึ่งหมายความว่า Volume นี้มีเวิร์ดเริ่มด้วยตัวอักษร E ถึง H และ Volume Set จะเป็นกลุ่มของดิสก์สำหรับตัวอักษรทั้งหมด

4. **System Identifier** จะระบุว่าระบบนั้นสามารถยอมรับได้และสามารถทำให้เกิด logic บน เซกเตอร์ 0-15 ได้ ตัวอักษรที่ถูกใช้ในในระบบ ID คือสิ่งที่ ISO 9660 เรียกว่า a-character และมีความยาวจำกัดที่ 31 ตัวอักษร

5. **Volume Size** เป็นจำนวนซึ่งบอกระบบปฏิบัติการว่ามี Logical Block อยู่ใน Volume Set นั้นเท่าไร Logical Block เป็นวิธีทั่วไปในการหาตำแหน่งของข้อมูลที่อยู่ใน Volume ซึ่งแต่ละตำแหน่งทั้งหมดนี้จะมี Logical Block Number

ABCDEFGHIJKLMNOPQRS  
 TUVWXYZ0123456789\_sp  
 !"%'()\*+,-./:;<=>?

รูปที่ 5.3 a – character

6. **Volume Set Size** เป็นจำนวนซึ่งบอกระบบปฏิบัติการว่ามี Volume อยู่ใน Volume Set นั้นเท่าไร Volume Sequence Number เป็นตำแหน่งใน Multiple Volume Set ที่ Volume นั้นอยู่ยกตัวอย่างเช่น แผ่นดิสก์แผ่นหนึ่งมี Volume Set Size = 5 และ Volume Sequence Number = 3 แสดงว่าดิสก์แผ่นนี้เป็นดิสก์แผ่นที่ 3 ของดิสก์ชุดที่ 5

7. **Logical Block Size** เป็นจำนวนไบต์ที่จัดรูปแบบแล้วเกิดที่ว่างที่น้อยที่สุดซึ่งถูกจัดสรรใน Volume นั้นๆ จำนวนนี้สามารถเป็น 512, 1024 หรือ 2048 ไบต์ก็ได้ ส่วนมากดิสก์ ISO 9660 จะใช้ Logical Block Size ขนาด 2048 ไบต์ ซึ่งมีขนาดเดียวกันกับขนาดของเซกเตอร์

8. **Path Table Size** จะบอกระบบปฏิบัติการว่ามีจำนวนไบต์เท่าไรที่อยู่ใน Path Tableระบบปฏิบัติการส่วนมากจะใช้ Path Table ใน fast memory, local memory (RAM) การใช้ Path Table Size เป็นวิธีที่รวดเร็วสำหรับระบบปฏิบัติการที่จะรู้ว่าจะต้องจัดสรรหน่วยความจำเท่าไรก่อนที่จะมันจะอ่าน Path Table วิธีนี้ทำให้ระบบปฏิบัติการอ่าน Path Table เพียงครั้งเดียวเท่านั้น เป็นการประหยัดเวลา

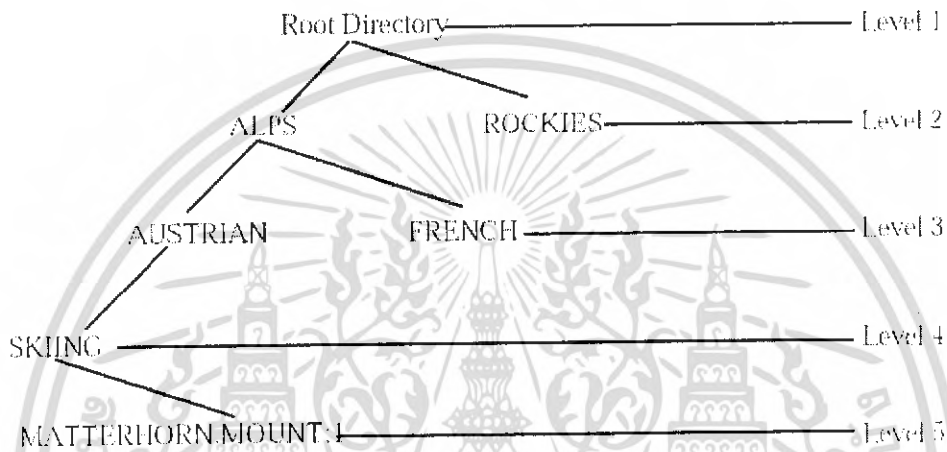
9. **Root Directory Record** จะบรรจุข้อมูลที่ระบบปฏิบัติการต้องการในการระบุตำแหน่งและอ่านไดเรกทอรีที่อยู่ระดับบนสุด รูปแบบของ Root Directory Record จะเหมือนกับ Directory Record อื่นๆ

10. **Identifier** อื่นๆ ใน Primary Volume Descriptor นั้นจะบรรจุข้อมูลเกี่ยวกับว่าใครเป็นผู้ตีพิมพ์ Volume นี้, ใครเตรียมข้อมูล, การประยุกต์ใช้งาน และชื่อของไฟล์ซึ่งบรรจุ copyright notice, บทคัดย่อ และบรรณานุกรม

11. **Time Stamps** เป็นฟิลด์ (field) ใน Primary Volume Descriptor ซึ่งบรรจุข้อมูลเกี่ยวกับว่า Volume นั้น ถูกสร้างขึ้นเมื่อไหร่, ถูกปรับปรุงเมื่อไหร่, เมื่อไหร่ที่ข้อมูลนั้นใช้ได้ และเมื่อไหร่ที่เลิกใช้ข้อมูล

## 5.2.2 โครงสร้างของไดเรกทอรี

โครงสร้างของไดเรกทอรีตาม ISO 9660 ได้ถูกจัดแบ่งเป็นระดับชั้นต่างๆคล้ายกับระบบไฟล์ส่วนใหญ่ ส่วนบนสุดของระดับชั้น คือ Root Directory ซึ่งจะระบุตำแหน่งอยู่ใน Primary Volume Descriptor



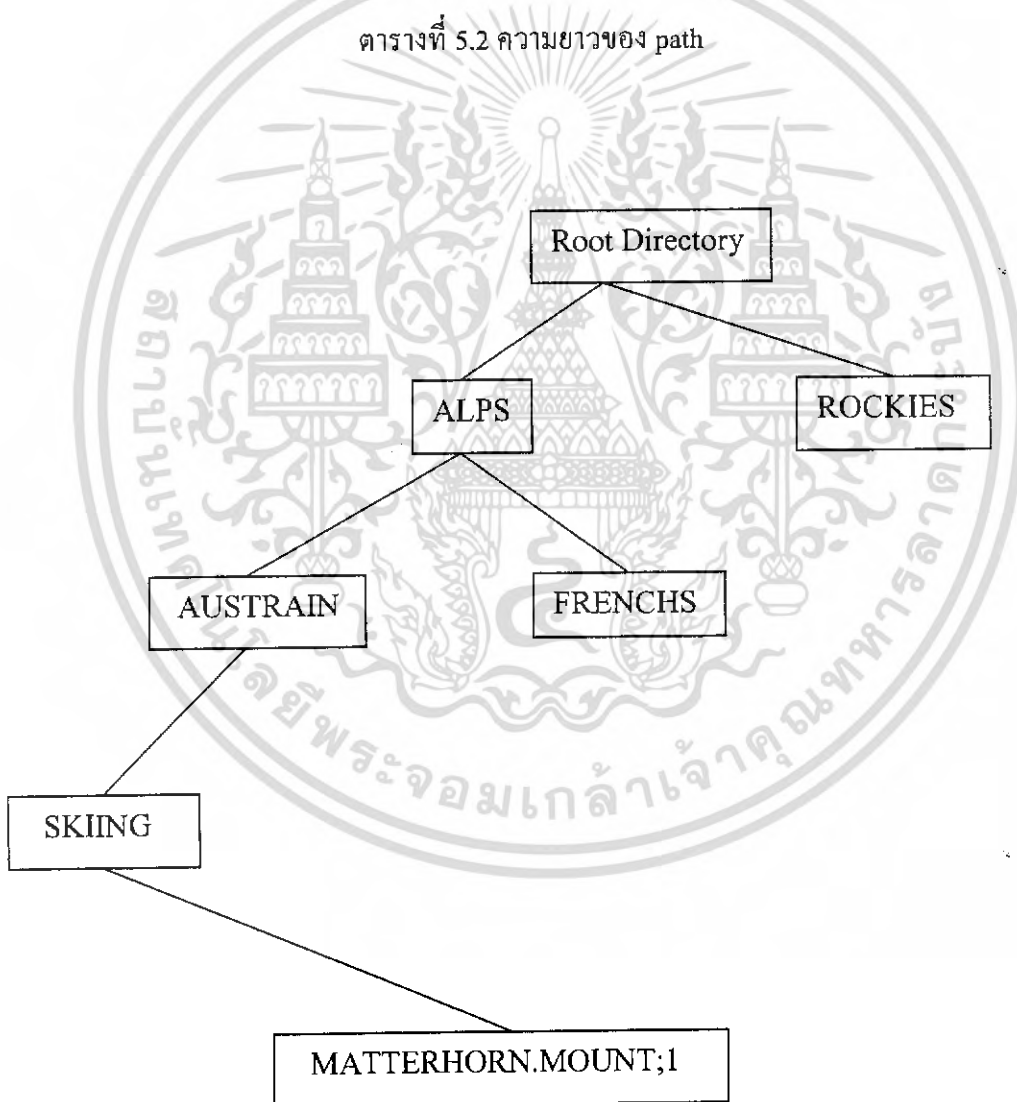
รูปที่ 5.4 ระดับชั้นไดเรกทอรี

จากรูป Root Directory เป็นไดเรกทอรีที่อยู่ Level 1 เท่านั้น ไดเรกทอรีย่อย ALPS และ ROCKIES จะอยู่ Level 2, ไดเรกทอรีย่อย AUSTRIAN และ FRENCH จะอยู่ Level 3 ไดเรกทอรีย่อย SKIING จะอยู่ Level 4 และไฟล์ MATTERHORN.MOUNT; 1 จะอยู่ Level 5

ISO 9660 ได้กำหนดให้สามารถมีความกว้างของ Directory Structure ได้สูงสุดคือ Level 8 นอกจากนี้ยังกำหนดความยาว path ของแต่ละไฟล์ ซึ่งทำได้จากผลรวมของความยาวไดเรกทอรีทั้งหมดที่เกี่ยวข้อง, ความยาวของ File Identifier และจำนวนไดเรกทอรีที่เกี่ยวข้อง ความยาว path จะต้องไม่เกิน 225 จากรูปที่ 5.5 จะมีความยาว path คือ 39 ดังตารางที่ 5.1

ไดเรกทอรีใน ISO 9660 Volume จะถูกบันทึกอยู่ในรูปของไฟล์ซึ่งบรรจุกลุ่มของ Directory record ในแต่ละ directory record จะอธิบายถึงไฟล์หนึ่งหรือไดเรกทอรีอื่น แต่ละไดเรกทอรีจะมีไดเรกทอรีแม่ (parent directory) ในไดเรกทอรีแม่จะบรรจุ directory record ซึ่งใช้ระบุไดเรกทอรีนั้น แสดงดังรูปที่ 5.6

Identifier	Length
ALPS	4
AUSTRIAN	8
SKIING	6
MATTERHORN.MOUNT;1	18
Number of directories	3
Sum of length and number of directories	39



รูปที่ 5.5 ไคเรคทอรีแม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ชื่อไฟล์

ตามมาตรฐาน ISO 9660 ทุกๆไฟล์และทุกๆไดเรกทอรีจะต้องมีชื่อ และชื่อนั้นเรียกว่า File Identifier โดยที่ File Identifier จะประกอบไปด้วย 5 ส่วน ดังแสดงในตารางที่ 5.2

	File Name	SEPARATOR	File Name	SEPARATOR	File Version
		1	Extension	2	Number
contents	d-characters	-	d-characters	-	A number from 1 to 32767
File 1	MATTERHORN	-	MOUNT	-	1
File 2	PIKES_PEAK	-	-	-	1
File 3	-	-	HILLS	-	1
directory	SKIING	-	-	-	-

ตารางที่ 5.3 File Identifier

File Identifier จะต้องเป็นไปตามเงื่อนไขดังนี้

- ถ้า File Name ไม่มีตัวอักษรแล้ว File Name Extension จะต้องมียังน้อย 1 ตัวอักษรดังแสดงในตารางที่ 5.2 ไฟล์ 3

- ถ้า File Name Extension ไม่มีตัวอักษรแล้ว File Name จะต้องมียังน้อย 1 ตัวอักษรดังแสดงในตารางที่ 5.2 ไฟล์ 2

- ผลรวมความยาวของจำนวนตัวอักษรของ File Name และ File Name Extension จะต้องไม่เกิน 30 ตัวอักษร

ส่วนไดเรกทอรี ตามมาตรฐาน ISO 9660 จะถูกกำหนดให้มีแค่ชื่อไฟล์เท่านั้น จะไม่มีSEPARATOR1 (.), SEPARATOR2 (.); File Name Extension หรือ File Version Number ดังแสดงในตารางที่ 5.2 ไดเรกทอรี

### 5.2.3 Path Table

Path Table เป็นวิธีตัดที่ระบบปฏิบัติการเลือกใช้ในการไปสู่แต่ละไดเรกทอรีบนดิสก์เพื่อที่จะให้ได้ไฟล์ที่ต้องการ Path Table จะเก็บชื่อไดเรกทอรี, ชื่อไดเรกทอรีแม่, ที่อยู่สำหรับเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่ละไดเรกทอรียกเว้น Root Directory

ส่วนใหญ่ระบบปฏิบัติการจะอ่าน Path Table ก่อน 1 ครั้งและเก็บไว้ในหน่วยความจำมากกว่าที่จะอ่านหลายๆครั้ง ตัวอย่างดังรูปที่ 5.5 ถ้าระบบปฏิบัติการไม่ใช่ Path Table แล้วจะต้องอ่าน Root Directory เพื่อที่จะหาตำแหน่งของไดเรกทอรี ALPS จากนั้นอ่านไดเรกทอรีALPS เพื่อที่จะหาตำแหน่งของไดเรกทอรี AUSTRAIN แล้วอ่านไดเรกทอรี SKIING เพื่อที่จะหาตำแหน่งของไฟล์ MATTERHORN.MOUNT; 1 แต่ถ้าใช้ Path Table ระบบปฏิบัติการจะค้นหาตำแหน่งของไดเรกทอรี SKIING ใน Path Table, อ่านไดเรกทอรี SKIING และหาตำแหน่งของไฟล์

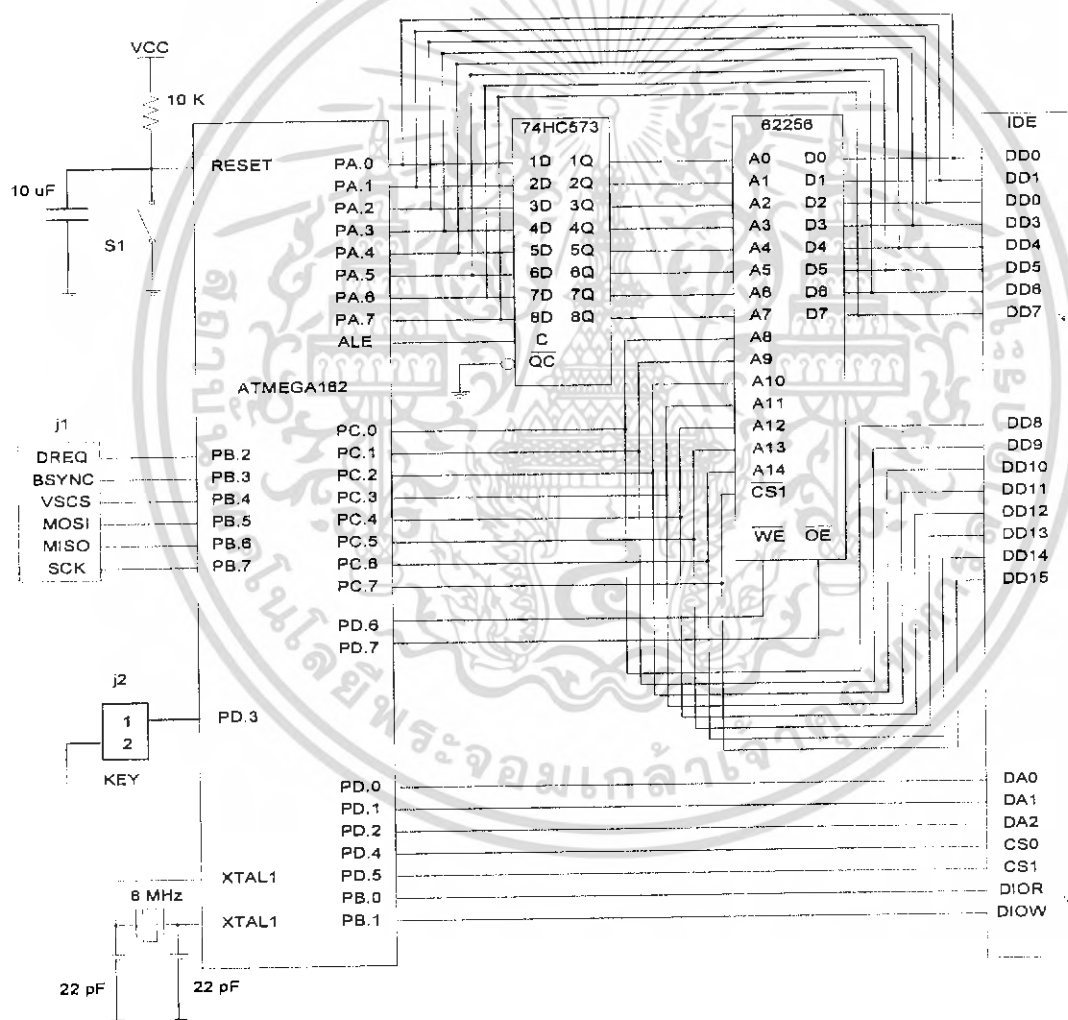
จากตัวอย่างข้างต้นพบว่าการค้นหาตำแหน่งของไฟล์โดยไม่ใช่ Path Table จะต้องอ่านข้อมูลจาก CD-ROM ถึง 4 ครั้ง แต่ถ้าใช้ Path Table จะอ่านข้อมูลเพียงแค่ครั้งเดียว ซึ่งใน CDROM ทั่ว ๆ ไป การขับ (Drive) จะใช้เวลาในการเข้าถึงข้อมูล (seek time) ประมาณ 0.25 วินาที ดังนั้นช่วงเวลาที่แตกต่างกันในการค้นหาจะเท่ากับ 0.75 วินาที ในขณะที่ต้องการเข้าถึงไฟล์จำนวนมาก ช่วงเวลาที่ต่างกันนี้จะมีผลต่อการทำงาน

## บทที่ 6

### การสร้างและการออกแบบ

#### 6.1 การสร้างและออกแบบการควบคุมซีดีรอม

โครงงานนี้จะใช้ไมโครคอนโทรลเลอร์ AVR ควบคุมการทำงานของ CD-ROM ซึ่งรายละเอียดของวงจรที่ทดลองแสดงดังรูปที่ 6.1

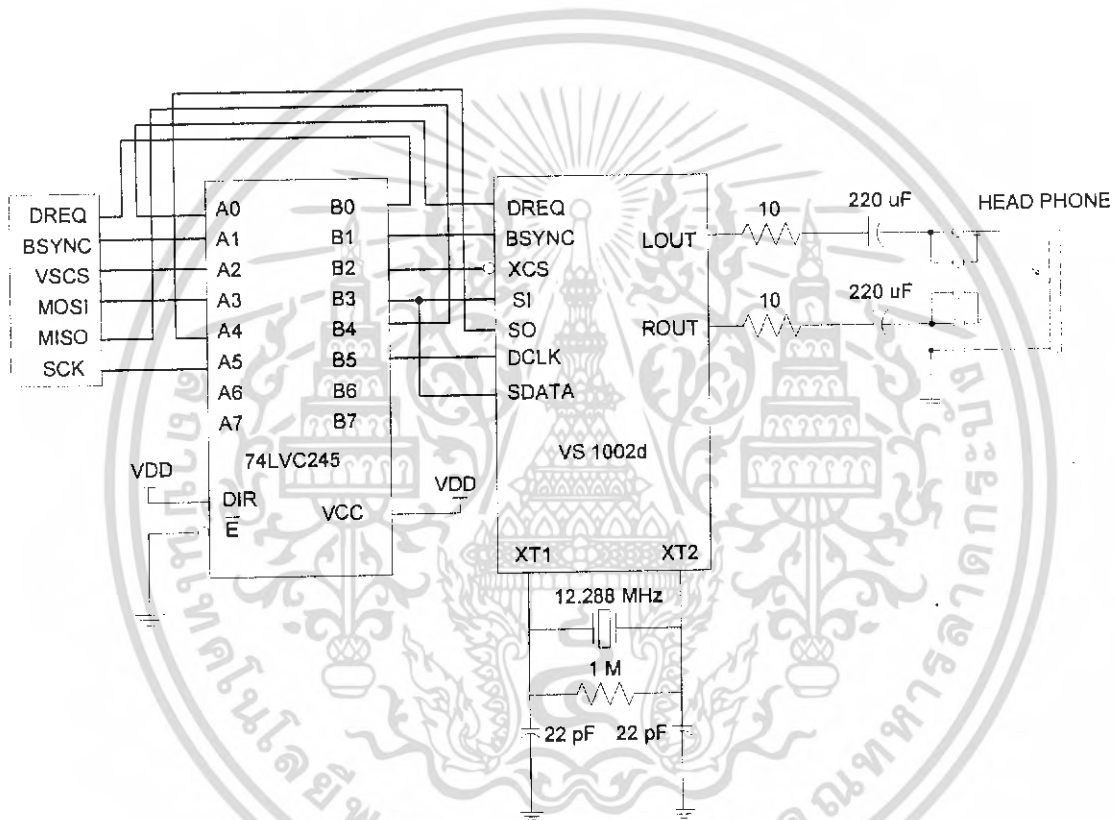


รูปที่ 6.1 วงจรที่ทำการทดลองติดต่อกับ CD-ROM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 6.2 การสร้างและออกแบบการควบคุมการทำงานของวงจรถอดรหัส

เนื่องจากชิพถอดรหัส VS1002d จะติดต่อกับไมโครคอนโทรลเลอร์แบบ SPI จึงทำให้การออกแบบวงจรถอดรหัสต้องใช้ไมโครคอนโทรลเลอร์ที่มีการติดต่อสื่อสารแบบ SPI ด้วย แต่การเชื่อมต่อระหว่างไมโครคอนโทรลเลอร์กับชิพถอดรหัสต้องผ่านไอซีเบอร์ 74LVC245 ก่อน เนื่องจากอุปกรณ์ทั้งสองมีขนาดแรงดันที่ไม่เท่ากัน ซึ่งการออกแบบวงจรถอดรหัสที่ใช้ในการทดลองนี้แสดงดังรูปที่ 6.2

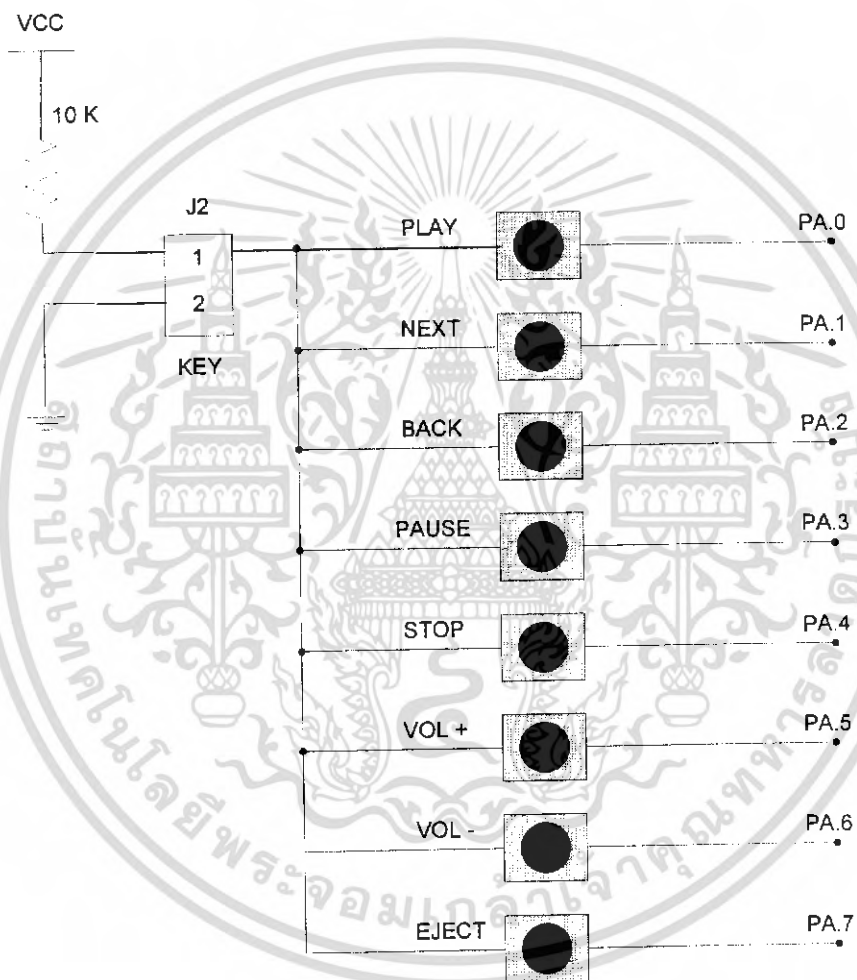


รูปที่ 6.2 วงจรที่ทำการทดลองถอดรหัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 6.3 การสร้างและออกแบบ Key Switch

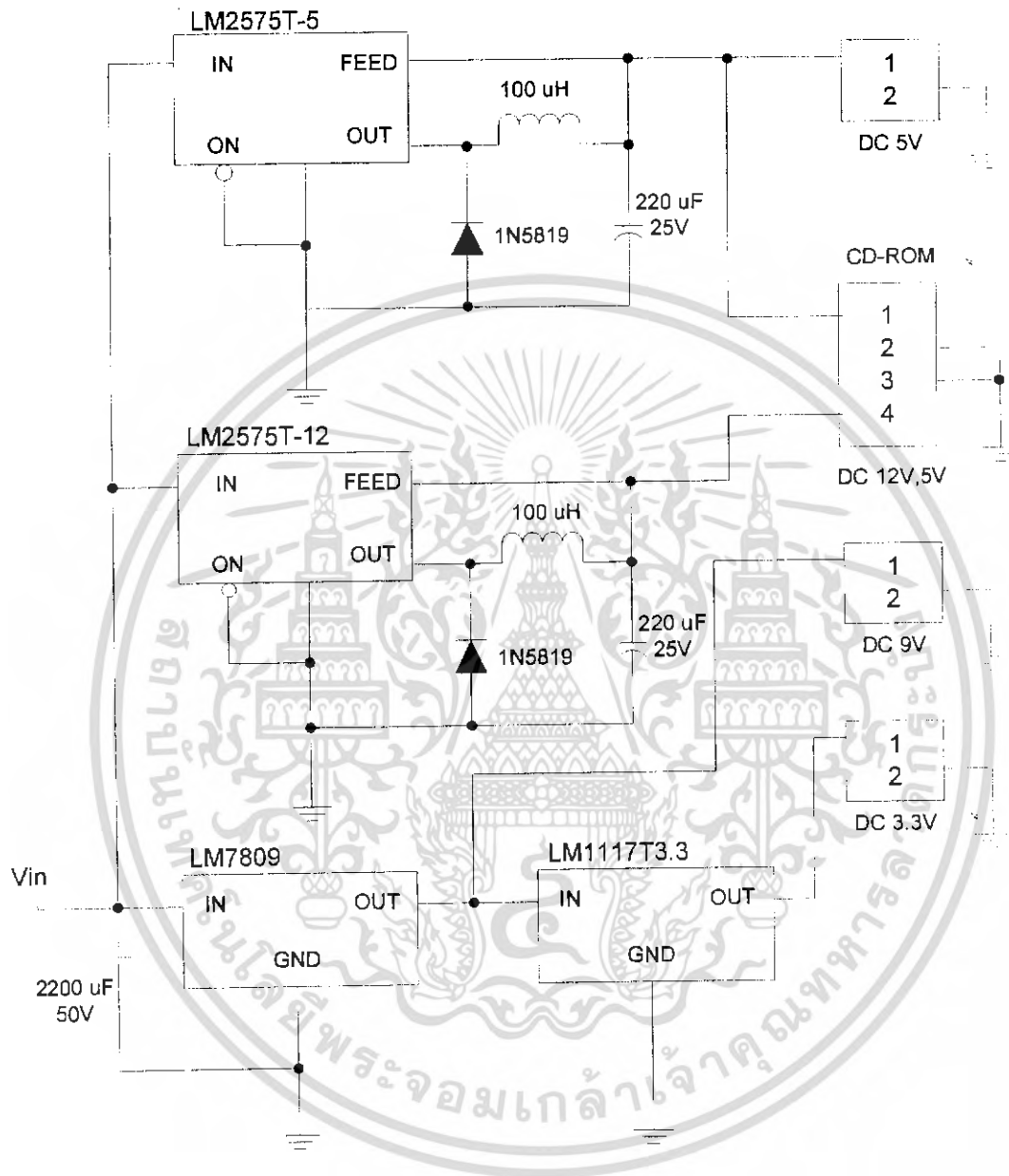
เนื่องจากจำนวนพอร์ตของไมโครคอนโทรลเลอร์มีจำนวนจำกัด ดังนั้นการออกแบบ Key Switch จึงใช้ความสามารถในการอินเตอร์รัพท์ของไมโครคอนโทรลเลอร์ ซึ่งกำหนดให้ตอบสนองการอินเตอร์รัพท์ที่ขอบขาลง เมื่อเกิดการอินเตอร์รัพท์ขึ้นแล้วค่อยไปตรวจสอบว่าเป็นปุ่มกดใด ซึ่งการออกแบบวงจร Key Switch ที่ใช้ในการทดลองนี้แสดงดังรูปที่ 6.3



รูปที่ 6.3 วงจร Key Switch

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

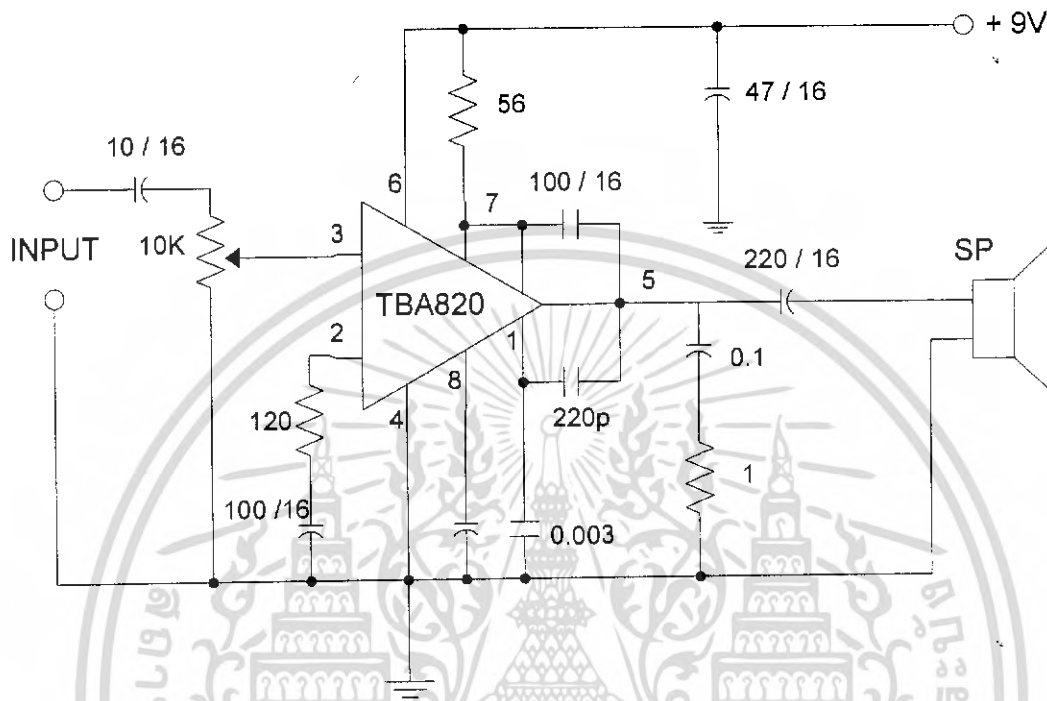
## 6.4 การสร้างและออกแบบแหล่งจ่ายไฟตรง



รูปที่ 6.4 วงจรแหล่งจ่ายไฟตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 6.5 การสร้างและออกแบบวงจรขยาย

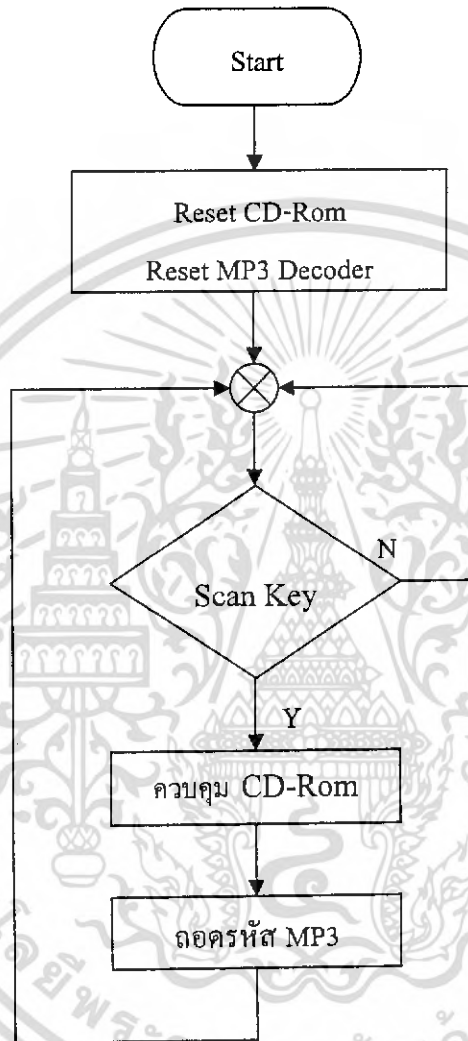


รูปที่ 6.5 วงจรขยาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 6.6 โฟลว์ชาร์ตการทำงานของส่วนต่างๆ

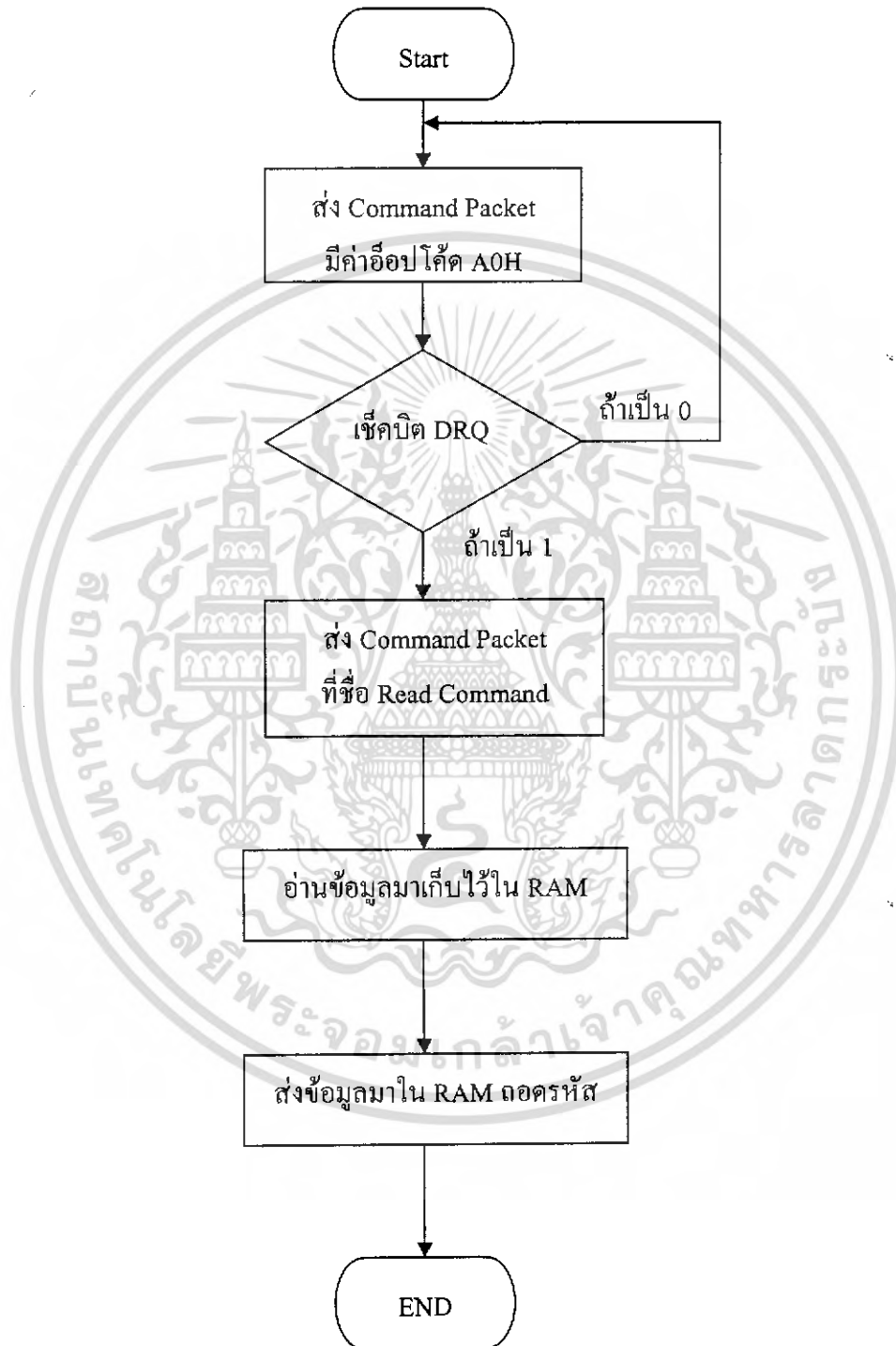
### 6.6.1 โฟลว์ชาร์ตของโปรแกรมหลัก



รูปที่ 6.6 โฟลว์ชาร์ตโปรแกรมหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีกรณีนำไปใช้

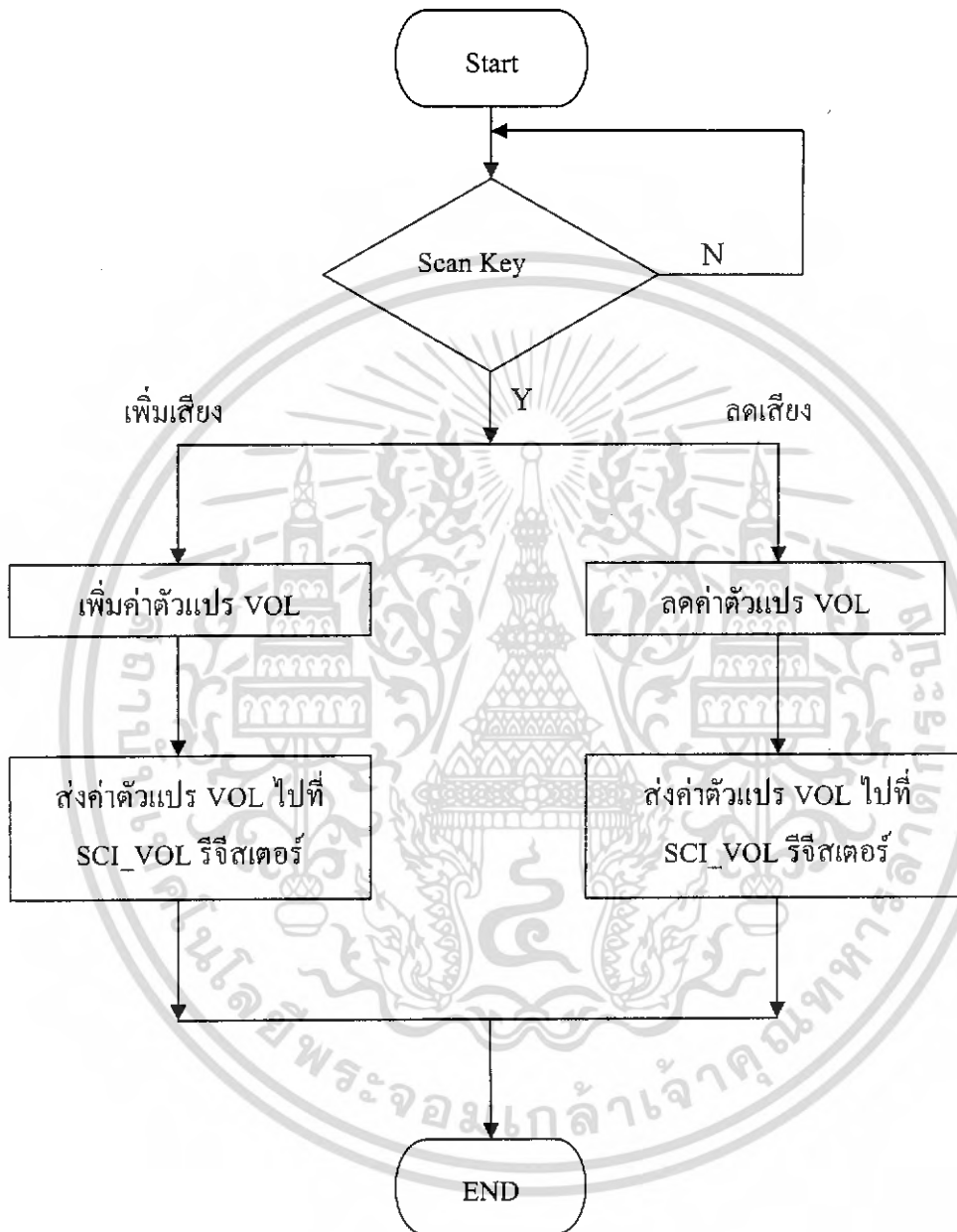
### 6.6.2 โฟลว์ชาร์ตของการ Play



รูปที่ 6.7 โฟลว์ชาร์ต โปรแกรมย่อยส่วน Play

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 6.6.3 โฟลว์ชาร์ตของการเพิ่มและลดเสียง



รูปที่ 6.8 โฟลว์ชาร์ตของการเพิ่มและลดเสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 7

### การทดลองและผลการทดลอง

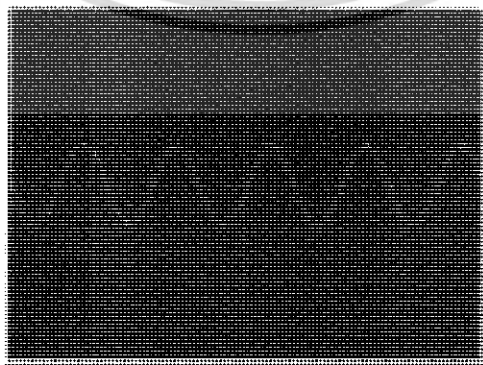
#### 7.1 การทดลองและผลการทดลองของการเขียนโปรแกรมควบคุมภาคถอดรหัส

การทดลองควบคุมภาคถอดรหัส ทำได้โดยใช้ไมโครคอนโทรลเลอร์ควบคุมผ่านทางพอร์ตอนุกรม SPI ที่ใช้ส่งคำสั่งและข้อมูลรหัสไฟล์ MP3 เข้าสู่ชิพ VS1002d ซึ่งการเขียนโปรแกรมควบคุมมีลำดับขั้นตอนดังนี้

1. ตั้งค่าเริ่มต้นพอร์ตอนุกรม SPI ของไมโครคอนโทรลเลอร์ ได้แก่ ความเร็วการส่งข้อมูล และส่ง MSB ก่อน ซึ่งถูกควบคุมโดยรีจิสเตอร์ SPCR
2. Reset ชิพ VS1002d
3. รอจนกว่าขา DREQ เป็น 1
4. ตั้งค่าเริ่มต้นรีจิสเตอร์ต่างๆ ของชิพ VS1002d ได้แก่ รีจิสเตอร์ SCI\_MODE รีจิสเตอร์ SCI\_CLOCKF รีจิสเตอร์ SCI\_AUDATA และรีจิสเตอร์ SCI\_VOL
5. ส่งข้อมูลทดสอบชิพโดยป้อนค่าดังนี้ 53H, EFH, 6EH, 30H, 00H, 00H, 00H, 00H, 45H, 78H, 69H, 74H

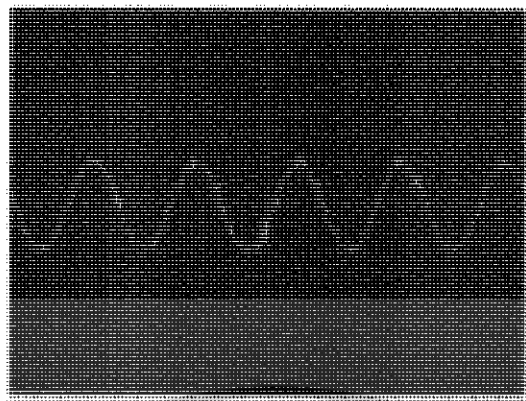
จากการทดลองป้อนค่าเข้าสู่ชิพ VS1002d สรุปได้ว่า การเขียนโปรแกรมควบคุมภาคถอดรหัส สามารถทำการถอดรหัสได้ ซึ่งจากการทดลองป้อนค่าต่างๆ ดังที่ได้กล่าวมาแล้วนั้นทำให้ได้สัญญาณเอาต์พุตเป็นสัญญาณอะนาล็อกออกดีโอที่มีความถี่อยู่ในช่วงที่มนุษย์สามารถรับฟังได้ เมื่อนำสัญญาณเอาต์พุตที่ได้ไปต่อกับลำโพงก็จะได้ยินเสียงออกมา

นอกจากนี้ยังได้ทำการทดลองนำคลื่นไซน์ความถี่ 1 ถึง 20 KHz แปลงเป็นข้อมูลไฟล์ MP3 นำไปป้อนให้กับวงจรถอดรหัสถอดรหัส วัสดุสัญญาณอินพุตเทียบกับสัญญาณเอาต์พุตได้ดังรูปที่ 7.1 และ 7.2 และพล็อตการตอบสนองความถี่ได้ดังรูปที่ 7.3

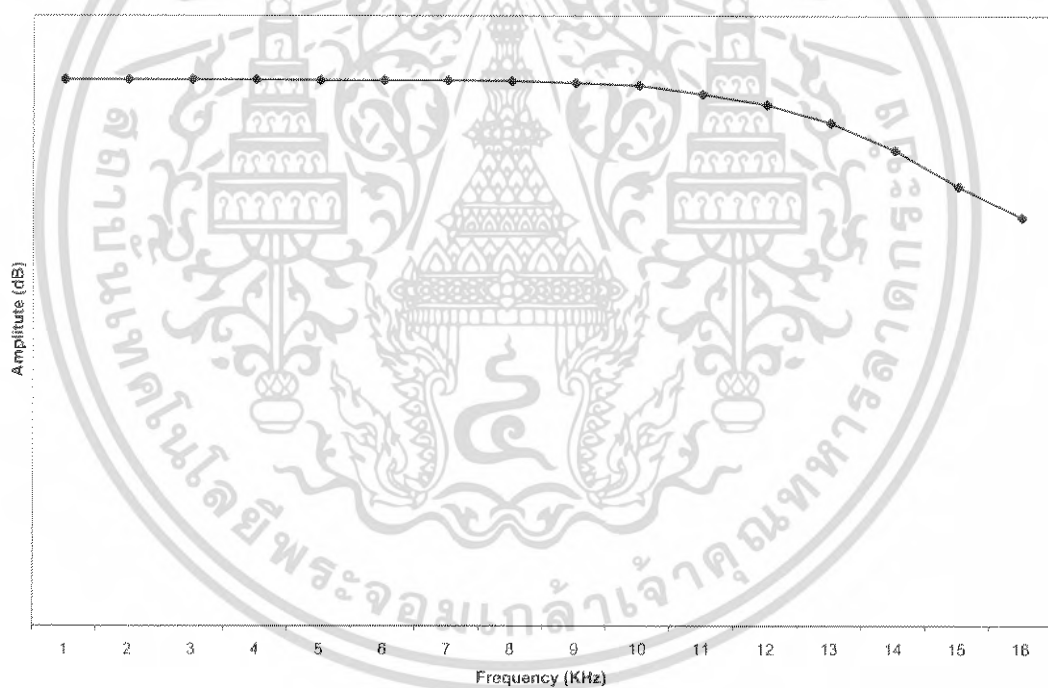


รูปที่ 7.1 สัญญาณอินพุตความถี่ 1 KHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.2 สัญญาณเอาต์พุตความถี่ 1 KHz ที่ได้จากวงจรทรานซิสเตอร์



รูปที่ 7.3 การตอบสนองความถี่ของวงจรทรานซิสเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 7.2 การทดลองและผลการทดลองของโครงการ

ในการทำชิ้นงานจริงสิ่งที่เป็นปัญหาของโครงการคือ การเขียนโปรแกรมควบคุมการทำงานของตัว CD-ROM ให้สามารถทำงานตามฟังก์ชันทั่วไปของเครื่องเล่น MP3 Player ให้ได้ การเขียนโปรแกรมควบคุมฟังก์ชันการทำงานของวงจรมีผู้จัดทำสามารถเขียนโปรแกรมควบคุมได้เพียงบางส่วนเท่านั้น กล่าวคือ สามารถควบคุมฟังก์ชันได้แก่ การเล่น (Play) การหยุด (Stop) การเปิด-ปิดถาด (Eject) และการเพิ่ม ลด เสียงเท่านั้น ส่วนการทำงานของฟังก์ชัน Next, Back, Pause, และการเลื่อน Track ยังไม่สามารถใช้งานได้

อัตรา Bit Rate	การทำงานของวงจรมี
80 kbps	ทำงานได้
96 kbps	ทำงานได้
112 kbps	ทำงานได้
128 kbps	ทำงานได้
160 kbps	ทำงานได้
192 kbps	ทำงานได้
256 kbps	ทำงานไม่ได้
320 kbps	ทำงานไม่ได้

ตารางที่ 7.1 การทำงานของวงจรมีที่อัตรา Bit Rate ต่างๆ

ชนิดของ CD-ROM	การทำงานของวงจรมี
ASUS 52X	ทำงานได้
ASUS DVD 16X	ทำงานได้
SAMSUNG 52X	ทำงานได้
LITE ON RW	ทำงานได้
LG 40X	ทำงานไม่ได้

ตารางที่ 7.2 การทำงานของวงจรมีเมื่อใช้ CD-ROM ยี่ห้อต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 8

### บทสรุป

ในปฏิญญาพนธ์ได้รายงานการออกแบบและการสร้างเครื่องเล่น MP3 ซึ่งใช้ซีดีรอมที่ใช้กับคอมพิวเตอร์ทั่วไป ซึ่งไฟล์ที่บันทึกลงบนแผ่นซีดีนั้น จะบันทึกในรูปแบบของไฟล์ MP3 มีผลทำให้จำนวนข้อมูลที่บันทึกได้มีขนาดที่มากกว่าการบันทึกเสียงแบบ Audio ที่ไม่มีการบีบอัดของข้อมูล

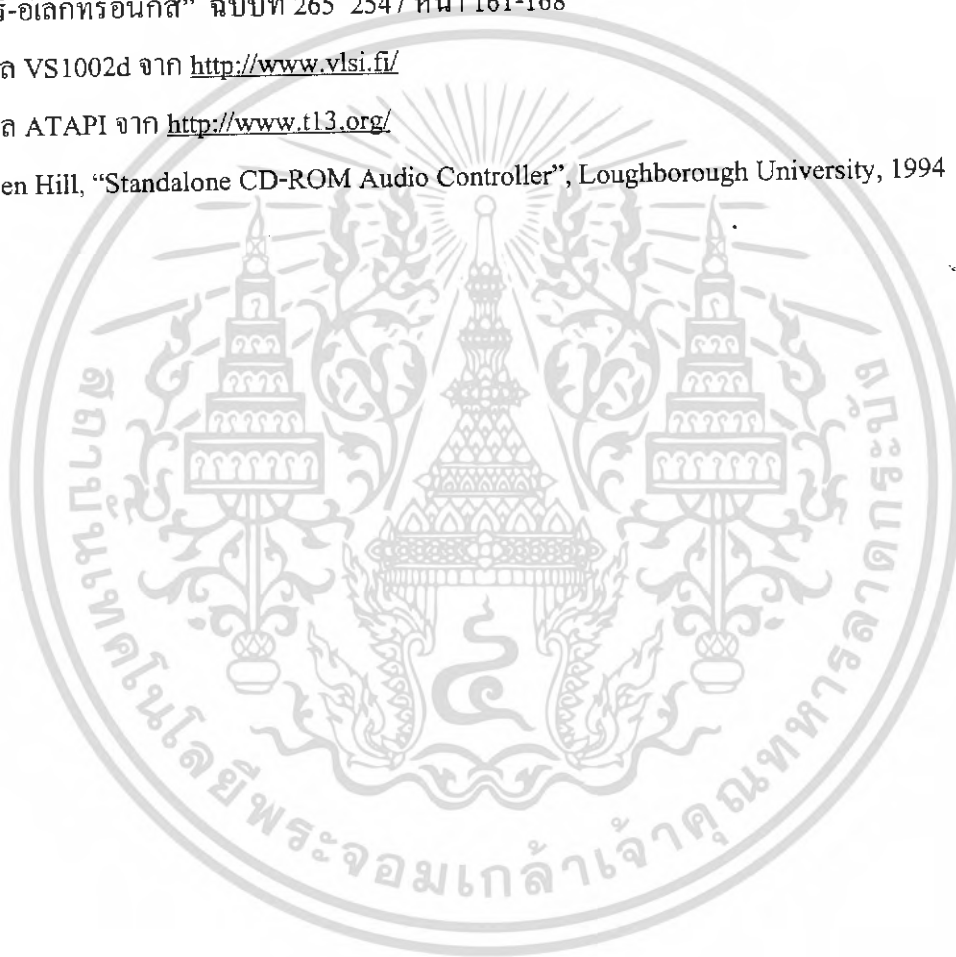
การออกแบบฮาร์ดแวร์ควบคุมฟังก์ชันการทำงานของซีดีรอม และส่วนของการถอดรหัสซึ่งใช้ชิปถอดรหัสเบอร์ VS1002d นั้น ทั้งสองส่วนจะถูกควบคุมโดยใช้ไมโครคอนโทรลเลอร์ AVR เบอร์ ATMEGA162 เพื่อไม่ต้องพึ่งพาไมโครคอมพิวเตอร์ และใช้งานหรือเคลื่อนย้ายโดยสะดวก จากผลการทดสอบการทำงานของวงจร ปรากฏว่าใช้งานได้เฉพาะการเล่น (Play) การหยุด (Stop) การเปิด ปิดถาด (Eject) และการเพิ่ม ลด เสียงเท่านั้น ส่วนการทำงานของฟังก์ชัน Next, Back, Pause, และการเลื่อน Track ยังไม่สามารถใช้งานได้

ปัญหาของระบบคือยังไม่สามารถใช้กับเครื่องเล่น CD-ROM บางยี่ห้อเท่านั้น และปัญหาอีกจุดหนึ่งคือ Power Supply เนื่องจากตัว CD-ROM จะใช้พลังงานสูง อยู่ระหว่าง 20 ถึง 25 W ทำให้ยังไม่สามารถใช้กับแบตเตอรี่ขนาดเล็กได้ และต้องหาวิธีให้ประหยัดพลังงานมากที่สุดด้วย

จากโครงการนี้ทางคณะผู้จัดทำเล็งเห็นว่าได้ประโยชน์ในการปรับปรุงพัฒนาต่อไปได้ในอนาคตในเรื่องของการอินเทอร์เฟซที่ต้องใช้มาตรฐาน ATA หรือ ATAPI ที่ใช้กันในฮาร์ดดิสก์หรือซีดีรอมไครฟ์ เพื่อที่จะทำการอ่านข้อมูลต่างๆ ที่บันทึกไว้ออกมาได้ นอกจากนี้ยังสามารถนำอุปกรณ์การเก็บข้อมูลที่มีขนาดใหญ่ เช่น ฮาร์ดดิสก์ ไปบันทึกข้อมูลตามที่ต้องการโดยไม่ต้องพึ่งพาไมโครคอมพิวเตอร์

## บรรณานุกรม

1. อุดม จีนประดับ, “ไมโครคอนโทรลเลอร์ MCS-51 Microcontroller MCS-51” สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ กรุงเทพฯ 2542
2. สมศักดิ์ เลอชุกต์, “เทคนิคสารพันสร้างสร้งงาน CD”, กรุงเทพฯ: ซอฟท์เพรส 2546 296 หน้า
3. อภิรักษ์ นามแต่ง เครื่องแปลง CD-ROM เป็นเครื่องเล่น CD สำหรับฟังเพลง “วารสารเซมิคอนดักเตอร์-อิเล็กทรอนิกส์” ฉบับที่ 265 2547 หน้า 161-168
4. ข้อมูล VS1002d จาก <http://www.vlsi.fi/>
5. ข้อมูล ATAPI จาก <http://www.t13.org/>
6. Darren Hill, “Standalone CD-ROM Audio Controller”, Loughborough University, 1994



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้