สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เครื่องถอดรหัสเอ็มเป็ก1 เลเยอร์ 3

MP3 DECODER

โดย

นายลภน สุภาพ

นางสาวศิริธร สาทพันธ์

นายสบโชค ยุทธวงศ์

อาจารย์ที่ปรึกษา

อาจารย์ชินภัทร นันทจิวากรชัย

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2542

เลขหมู่.....
เลขทะเบียน 36876
วัน, เดือน, ปี 29 ส.ค. 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ ปีการศึกษา 2542

ภาควิชา อิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร ลาดกระบัง

เรื่อง เครื่องถอดรหัสเอ็มเป็ก1 เลเยอร์3

ผู้จัดทำ

1.นาย ลภน สุภาพ

2. นางสาว ศิริธร สาทพันธ์

3. นาย สบโชค ยุทธวงศ์

ลงชื่อ.....แทนอาจารย์ที่ปรึกษา

(อาจารย์เทอดศักดิ์ ลิ่วหาทอง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องถอดรหัสเอ็มเป็ก เลเยอร์3

นาย ลภน สุภาพ
นางสาว ศิริธร สาทพันธ์
นาย สบโชค ยุทธวงศ์
อ.ชินภัทร นันทจิวากรชัย (อาจารย์ที่ปรึกษา)
ปีการศึกษา 2542

บทคัดย่อ

โครงงานนี้เป็นการพัฒนาเครื่องถอดรหัสเอ็มเป็ก 1 เลเยอร์ 3 ให้สามารถทำงานได้โดยไม่ต้องอาศัยอุปกรณ์สนับสนุน เพื่อประโยชน์ในการประยุกต์ใช้งานต่อไปในอนาคต โดยเนื้อหาหลักจะแบ่งออกเป็นสองส่วน ส่วนแรกเป็นการศึกษาและทดลองสร้างเครื่องถอดรหัสเอ็มเป็ก 1 เลเยอร์ 3 ส่วนที่สองเป็นส่วนที่ทำหน้าที่เป็นแหล่งข้อมูลเพื่อป้อนให้กับตัวถอดรหัส โดยจะมุ่งเน้นที่ซีดีรอมเป็นหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MP3 DECODER

Mr.Laphon Suparb
Miss.Siritorn Sattapant
Mr.Sobchock Yuttawong
Mr.Chinnapat Nuntajiwakornchai Advisor
1999

ABSTRACT

This project is to develop the MP3 decoder to work without a support device for further application in the future.We will devided the contents of this project in two parts.The first part is to study and make the MP3 decoder, and the second part is to make a data generator to generate the signal for the decoder.The main of the second part is the CD-ROM.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อ	i
ABSTRACT	ii
สารบัญ	iii
สารบัญรูปภาพ	vi
สารบัญตาราง	viii
บทที่ 1 บทนำ	1
1.1 กล่าวนำ	1
1.2 ขอบเขต	1
บทที่ 2 การบีบอัดข้อมูลเสียงแบบเอ็มเป็ก	3
2.1 บทนำ	3
2.2 หลักการพื้นฐานและการใช้งาน	3
2.3 การเข้ารหัสแบบเอ็มเป็กโดยทั่วไป	4
2.4 รูปแบบของข้อมูล	7
บทที่ 3 MAS 3507D	9
3.1 บทนำ	9
3.2 การทำงานและรายละเอียดภายใน MAS 3507D	10
3.2.1 ตัวประมวลผลข้อมูลเชิงเลข (DSP)	10
3.2.2 เฟิร์มแวร์ (Firmware)	11
3.2.3 ความสามารถในการโหลดโปรแกรมภายนอก	12
3.2.4 การประมวลผลในด้านความถี่	12
3.2.5 การจัดการสัญญาณนาฬิกา	13
3.2.6 การทำงานของส่วนจ่ายไฟ	13
3.2.7 การเชื่อมต่อ	14
3.3 การเชื่อมต่อการควบคุม	18
3.3.1 การเชื่อมต่อบัส I^2C	18
3.3.2 โครงสร้างคำสั่ง	20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
บทที่ 4 DAC 3550A	21
4.1 บทนำ	21
4.2 การทำงานและรายละเอียดของส่วนต่างๆ	24
4.2.1 การเชื่อมต่อ I^2S	24
4.2.2 อินเทอร์โพเลชั่นฟิลเตอร์	25
4.2.3 วงจรกรองความถี่ต่ำผ่าน	25
4.2.4 ส่วนเลือกช่องสัญญาณและการมิกซ์สัญญาณ	26
4.2.5 โพสท์ฟิลเตอร์ออปแอมป์ (Postfilter Op-Amps), และสายต่อออกภายนอก (Line Out)	26
4.2.6 การควบคุมระดับเสียง	27
4.2.7 วงจรขยายสำหรับหูฟัง	27
4.2.8 ระบบสัญญาณนาฬิกา	27
4.2.9 การเชื่อมต่อบัส I^2C	28
4.2.10 รีจิสเตอร์	29
บทที่ 5 โปรโตคอลที่ใช้ในการเชื่อมต่อกับ CD-ROM	30
5.1 ATA	30
5.1.1 การเชื่อมต่อทางกายภาพ	30
5.1.2 ขาเชื่อมต่อสำหรับการส่งข้อมูล	31
5.1.3 รีจิสเตอร์ภายใน	32
5.1.4 โปรโตคอลในการส่งข้อมูล	33
5.1.5 ลำดับเวลาในการส่งสัญญาณ	35
5.2 ATAPI	36
5.2.1 รีจิสเตอร์ภายใน	36
5.2.2 โปรโตคอลในการส่งข้อมูล	36
5.2.3 แพ็คเกตคอมมานด์ของ ATAPI	37
5.2.4 การใช้รีจิสเตอร์ Byte Count (Cylinder Low/High) สำหรับแพ็คเกตคอมมานด์	38
5.2.5 การใช้รีจิสเตอร์ Sector Count สำหรับแพ็คเกตคอมมานด์	39

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

5.2.6 ลำดับการส่งชุดคำสั่งที่มีการส่งข้อมูลกลับในแบบ PIO 39

5.2.7 ลำดับการส่งชุดคำสั่งที่ไม่มีการส่งข้อมูลกลับ 40

บทที่ 6 ISO9660 42

6.1 บทนำ 42

6.2 ภาพรวมโครงสร้างของ ISO9660 42

6.2.1 Volume Descriptor 43

6.2.2 โครงสร้างของไดเรคทอรี 45

6.2.3 Path Table 48

บทที่ 7 เครื่องถอดรหัสเอ็มเป็ก 1 เลเยอร์ 3 49

7.1 โครงสร้างฮาร์ดแวร์ 49

7.1.1 หน้าที่การทำงานของแต่ละส่วน 49

7.1.2 การเชื่อมต่อระหว่างแต่ละส่วน 52

7.2 ซอฟท์แวร์ควบคุมการทำงาน 54

7.2.1 โปรแกรมควบคุมการทำงานภายในไมโครคอนโทรลเลอร์ 89S8252 54

7.2.2 โปรแกรมควบคุมการทำงานภายในไมโครคอนโทรลเลอร์ 89C51 56

บทที่ 8 การทดลองและสรุปผลการทดลอง 62

8.1 การทดลอง 62

8.1.1 การเล่นเพลงที่บีบอัดข้อมูลด้วยอัตราแซมปลิ้งต่าง ๆ กัน 62

8.1.2 การทดสอบการเชื่อมต่อกับไดรฟ์ซีดีรอมจากบริษัทต่าง ๆ 62

8.1.3 การทดสอบการอ่านแผ่นซีดีรอมที่ถูกเขียนขึ้นมาในระบบไฟล์ที่ต่างกัน 63

8.2 สรุปผลการทดลอง 64

บรรณานุกรม

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

	หน้า
รูปที่ 1.1 แสดงบล็อคไดอะแกรมของเครื่องถอดรหัส MP3	3
รูปที่ 2.1 กราฟเปรียบเทียบอัตราส่วนการบีบอัดเสียงของการเข้ารหัสเลเยอร์ต่าง ๆ เมื่อเทียบกับคุณภาพเสียงต้นแบบจากคอมแพ็คดิสก์ (CD)	4
รูปที่ 2.2 แสดงระดับความดังเสียงที่เริ่มได้ยินที่ความถี่ต่าง ๆ	5
รูปที่ 2.3 แสดงผลของการปิดกั้นเสียง	5
รูปที่ 2.4 แสดงการปิดกั้นเสียงที่ไม่ได้ยินในย่านความถี่ย่อยหนึ่ง ๆ ของการเข้ารหัสเอ็มเป็ก	7
รูปที่ 2.5 แสดงรูปแบบข้อมูลเอ็มเป็ก1 เลเยอร์3	7
รูปที่ 2.6 แสดงบิตต่าง ๆ ในส่วนหัวข้อมูล	8
รูปที่ 3.1 แสดงบล็อกไดอะแกรมของ MAS 3507D	9
รูปที่ 3.2 บล็อคไดอะแกรมของตัวถอดรหัสเอ็มเป็ก	10
รูปที่ 3.3 timing diagram ของ I^2S อินพุท	14
รูปที่ 3.4 timing diagram ของการเชื่อมต่อ SDO ในโหมด 16 บิต/แซมเปิล	14
รูปที่ 3.5 timing diagram ของการเชื่อมต่อ SDO ในโหมด 32 บิต/แซมเปิล	15
รูปที่ 3.6 โปรโตคอลของบัส I^2C ของ MAS 3507D	17
รูปที่ 4.1 แสดงบล็อกไดอะแกรมของ DAC 3550A	21
รูปที่ 4.2 แสดงการประยุกต์ใช้งานทั่วไป : เครื่องเล่น MP3	22
รูปที่ 4.3 แสดงบล็อคไดอะแกรมของ DAC 3550A	23
รูปที่ 4.4 I^2S โหมด 16 บิต (LR_SEL=0)	24
รูปที่ 4.5 I^2S โหมด 32 บิต (LR_SEL=0)	24
รูปที่ 4.6 การตอบสนองความถี่ของอินเทอร์โพเลชั่นฟิลเตอร์ในช่วงความถี่ : 0..22 kHz	25
รูปที่ 4.7 แสดงแมทริกซ์การสวิทช์	26
รูปที่ 4.8 แสดงโพสท์ฟิวเตอร์ ออปแอมป์, ดีมฟาซิส ออปแอมป์ และสายออก	26
รูปที่ 4.9 แสดงโปรโตคอลของบัส I^2C ในการเขียน	29
รูปที่ 5.1 แสดงการเชื่อมต่อระหว่างโฮสท์กับอุปกรณ์ 2 ตัว	30
รูปที่ 5.2 แสดงการเชื่อมต่อระหว่างโฮสท์กับอุปกรณ์ตัวเดียว	31
รูปที่ 5.3 ตัวเชื่อมต่อขนาด 40 ขา ตามมาตรฐาน ATA	31

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
รูปที่ 5.4 ตัวเชื่อมต่อจ่ายไฟเลี้ยงให้อุปกรณ์ตามมาตรฐาน ATA	31
รูปที่ 5.5 แสดงลำดับเวลาในการส่งสัญญาณ	35
รูปที่ 6.1 โครงสร้า ISO9660	42
รูปที่ 6.2 Primary Volume Descriptor	43
รูปที่ 6.3 d-character	44
รูปที่ 6.4 a-character	44
รูปที่ 6.5 ลำดับชั้นของไดเรคทอรี	46
รูปที่ 6.6 ไดเรคทอรีแม่	47
รูปที่ 7.1 แสดงโครงสร้างภายนอกของ 89S8252	50
รูปที่ 7.2 แสดงการเชื่อมต่อพอร์ตสื่อสารอนุกรมแบบ SPI	51
รูปที่ 7.3 แสดง Timing Diagram ของการเชื่อมต่อแบบ SPI	51
รูปที่ 7.4 โฟลว์ชาร์ทการอ่านโครงสร้างข้อมูลของแผ่นและจัดรูปแบบข้อมูล	57
รูปที่ 7.5 โฟลว์ชาร์ทการอ่านข้อมูลจากแผ่นซีดีรอมในตำแหน่งเพลงที่ผู้ใช้เลือก	58
รูปที่ 7.6 โฟลว์ชาร์ทของโปรแกรมบริการอินเตอร์รัพท์ของ Time0	59
รูปที่ 7.7 โฟลว์ชาร์ทของโปรแกรมบริการอินเตอร์รัพท์พอร์ตอนุกรมและสัญญาณดีมานด์	60
รูปที่ 7.8 โฟลว์ชาร์ทแสดงการทำงานของ 89C51	61

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้า
ตารางที่ 2.1 แสดงความกว้างของย่านความถี่วิกฤตย่านต่าง ๆ	6
ตารางที่ 3.1 การตั้งค่าสำหรับแมทริกซ์ดิจิตอล volume	11
ตารางที่ 3.2 ความถี่ CLKO	12
ตารางที่ 3.3 แสดงถึงการกำหนดโหมดการทำงานเริ่มต้นผ่านทางขา PI ต่างๆ	15
ตารางที่ 3.4 สัญญาณเอาท์พุทของ PIO ในการถอดรหัสเอ็มเป็ก	16
ตารางที่ 3.5 แสดงแผนแอดเดรส I^2C	17
ตารางที่ 3.6 แสดงตำแหน่งย่อย	18
ตารางที่ 3.7 แสดงการกำหนดบิทของรีจิสเตอร์ควบคุม	18
ตารางที่ 3.8 การกำหนดบิตของรีจิสเตอร์ข้อมูล	18
ตารางที่ 3.9 คำสั่งคอนโทรลเลอร์พื้นฐาน	19
ตารางที่ 4.1 การควบคุมระดับเสียง	27
ตารางที่ 4.2 โหมดการทำงาน	28
ตารางที่ 4.3 ตำแหน่งรีจิสเตอร์ I^2C	29
ตารางที่ 5.1 แสดงแอดเดรสของรีจิสเตอร์ตามมาตรฐาน ATA	32
ตารางที่ 5.2 การใช้รีจิสเตอร์ Byte Count	38
ตารางที่ 6.1 ความยาวของ path	46
ตารางที่ 6.2 File Identifier	47
ตารางที่ 8.1 การทดลองเล่นเพลงที่บีบอัดข้อมูลด้วยอัตราแซมปลิ้งต่าง ๆ	62
ตารางที่ 8.2 การทดสอบการเชื่อมต่อกับไดรฟ์ซีดีรอมจากบริษัทต่างๆ	63
ตารางที่ 8.3 การทดลองเล่นเพลงจากซีดีรอมของบริษัทต่างๆ	63

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1. กล่าวนำ

ในปัจจุบันเพลงที่ถูกบีบอัดในรูปแบบ MP3 นั้นได้รับความนิยมอย่างแพร่หลาย เนื่องจากเสียงที่ได้จะมีคุณภาพสูง และใช้เนื้อที่น้อยในการเก็บข้อมูล แต่ในการใช้งานยังคงต้องอาศัยคอมพิวเตอร์ในการถอดรหัสอยู่ ซึ่งทำให้ความสามารถของเครื่องคอมพิวเตอร์ในการทำงานลดลง ทำให้เครื่องคอมพิวเตอร์ที่มีความสามารถสูงไม่พอ จะไม่สามารถใช้งานอื่นไปพร้อม ๆ กับการถอดรหัสเอ็มเปก1 เลเยอร์3 (MP3) ได้

ด้วยเหตุนี้ผู้จัดทำจึงได้ศึกษาและทดลองสร้างเครื่องถอดรหัสเอ็มเปก ที่สามารถทำงานได้ด้วยตัวมันเอง (stand alone) โดยไม่ต้องอาศัยเครื่องคอมพิวเตอร์

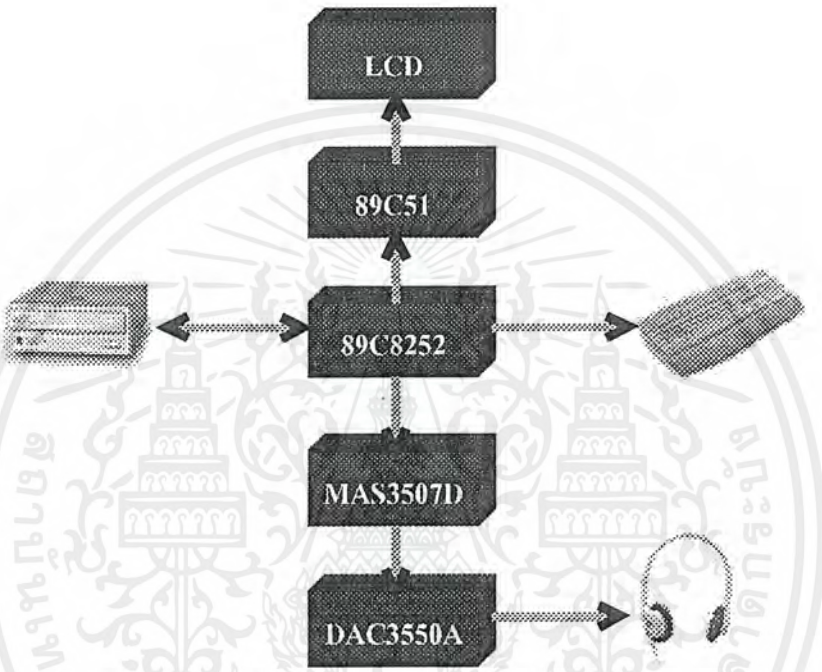
1.2. ขอบเขต

ทดลองสร้างเครื่องถอดรหัสเอ็มเปก 1 เลเยอร์ 3 (MP3) โดยมีแหล่งข้อมูลมาจากซีดีรอม ได้ทำการศึกษาและทดลองดังนี้

- ศึกษาการถอดรหัสแบบเอ็มเปก
- ศึกษามาตรฐานการบีบอัดข้อมูลตามมาตรฐาน ISO 11172-3
- ศึกษาการทำงานของ MAS3507D และ DAC3550A
- สร้างวงจรถอดรหัสเอ็มเปก1 เลเยอร์ 3 โดยใช้ MAS3507D และ DAC3550A
- สร้างวงจรควบคุมการทำงานของตัวถอดรหัสโดยสื่อสารผ่าน I^2C บัส
- ศึกษามาตรฐาน ATA และ ATAPI ซึ่งเป็นโปรโตคอลที่ใช้ในการติดต่อกับ CD-ROM
- ศึกษามาตรฐาน ISO9660 ซึ่งเป็นมาตรฐานในการจัดการระบบไฟล์ (file system)
- ศึกษาการใช้งานจอ LCD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เขียนโปรแกรมควบคุมการเชื่อมต่อระหว่าง 89S8252 กับ CD-ROM
- เขียนโปรแกรมควบคุมการเชื่อมต่อระหว่าง 89S8252 กับ 89C51
- เขียนโปรแกรมควบคุมการเชื่อมต่อระหว่าง 89S8252 กับ MAS3550D
- เขียนโปรแกรมควบคุมการแสดงผลบนจอ LCD โดยใช้ 89C51



รูปที่ 1.1 แสดงบล็อคไดอะแกรมของเครื่องถอดรหัส MP3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
การบีบอัดข้อมูลเสียงแบบเอ็มเป็ก
(MPEG Audio Compression)

2.1 บทนำ

การบีบอัดข้อมูลเสียงแบบเอ็มเป็ก (MPEG Audio Compression) เป็นวิธีการบีบอัดข้อมูลเสียงแบบดิจิตอลที่มีความเหมือนจริงสูง (High Fidelity : HiFi) การบีบอัดข้อมูลแบบเอ็มเป็กสำเร็จลงในปี 1993 โดยคณะกรรมการสากลแห่งการเชี่ยวชาญการบีบอัดเสียงเหมือนจริง ซึ่งรู้จักกันดีในชื่อ Motion Picture Expert Group (MPEG) และได้รับมาตรฐาน ISO/IEC 11172-3

2.2 หลักการพื้นฐานและการใช้งาน

หลักการบีบอัดข้อมูลแบบเอ็มเป็ก จะใช้ประโยชน์จากขีดจำกัดในการได้ยินของมนุษย์โดยไม่จัดเก็บข้อมูลเสียงที่มนุษย์ฟังไม่ได้ยิน เพื่อให้ข้อมูลมีขนาดเล็กลง ซึ่งใช้หลักการที่ว่า ถ้าเราได้ยินเสียง 2 เสียงที่มีความถี่ใกล้เคียงกัน โดยให้เสียงที่ 2 ดังกว่าเสียงที่ 1 มาก เมื่อเราฟัง 2 เสียงนี้พร้อมกันจะได้ยินเสียงที่ 2 (ดังกว่า) เพียงเสียงเดียว ดังนั้นในการจัดเก็บข้อมูลสามารถตัดข้อมูลเสียงที่ค่อยกว่าออกไปได้บางส่วน

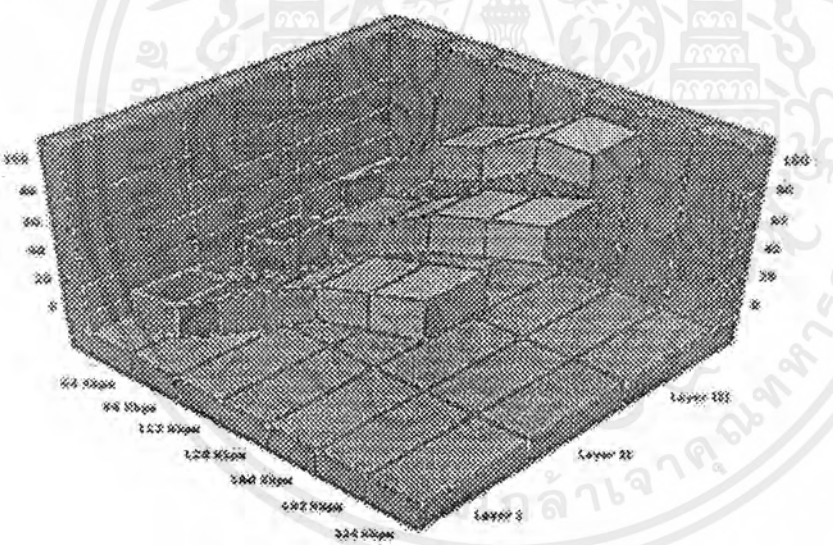
การบีบอัดข้อมูลเอ็มเป็กสามารถเลือกวิธีการบีบอัดเสียงได้หลายแบบ ดังนี้

- อัตราการสุ่ม (sampling rate) 32, 44.1, หรือ 48 กิโลเฮิร์ต (kHz)
- รองรับระบบเสียงได้ทั้ง 1 และ 2 ช่องเสียง ซึ่งมีอยู่ 4 แบบดังนี้
  - แบบโมโนเดี่ยว (Monophonic mode)
  - แบบโมโนคู่ (Dual Monophonic mode)
  - แบบสเตอริโอ (Stereo mode)
  - แบบจอยท์-สเตอริโอ (Joint-Stereo mode)
- สามารถเลือกค่าอัตราการส่งข้อมูล (bit rate) ได้ตั้งแต่ 32 – 224 กิโลบิตต่อวินาที ต่อหนึ่งช่องเสียง ขึ้นอยู่กับอัตราการสุ่ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- รูปแบบของการเข้ารหัสไฟล์เอ็มเป็ก มี 3 เลเยอร์ (Layer) คือ เลเยอร์ 1, เลเยอร์ 2 และเลเยอร์ 3 แต่ละเลเยอร์มีลักษณะต่างกันดังนี้
  - เลเยอร์ 1 มีความซับซ้อนน้อยที่สุด ต้องใช้อัตราการส่งข้อมูลสูงถึง 384 กิโลบิตต่อวินาที เพื่อที่จะให้ได้เสียงคุณภาพสูง
  - เลเยอร์ 2 มีความซับซ้อนมากขึ้น คุณภาพเสียงสูงกว่า เลเยอร์ 1 ใช้อัตราการส่งข้อมูล 160 กิโลบิตต่อวินาที จะได้เสียงคุณภาพสูง และ 192 กิโลบิตต่อวินาที จะได้คุณภาพเสียงไม่แตกต่างจากเสียงต้นแบบ
  - เลเยอร์ 3 มีความซับซ้อนมากที่สุดแต่สามารถให้คุณภาพเสียงที่ดีที่สุด ใช้อัตราการส่งข้อมูล 160 กิโลบิตต่อวินาที ซึ่งจะให้คุณภาพเสียงไม่แตกต่างจากเสียงต้นแบบ

สามารถแสดงความสัมพันธ์ของอัตราการส่งข้อมูลเสียง (กิโลบิตต่อวินาที) สำหรับการเข้ารหัสเลเยอร์ต่าง ๆ กับคุณภาพเสียงเมื่อเทียบกับเสียงต้นแบบได้ดังรูปที่ 2.1



รูปที่ 2.1 กราฟเปรียบเทียบอัตราส่วนการบีบอัดเสียงของการเข้ารหัสเลเยอร์ต่าง ๆ เมื่อเทียบกับคุณภาพเสียงต้นแบบจากคอมแพ็คดิสก์ (CD)

2.3 การเข้ารหัสแบบเอ็มเป็กโดยทั่วไป

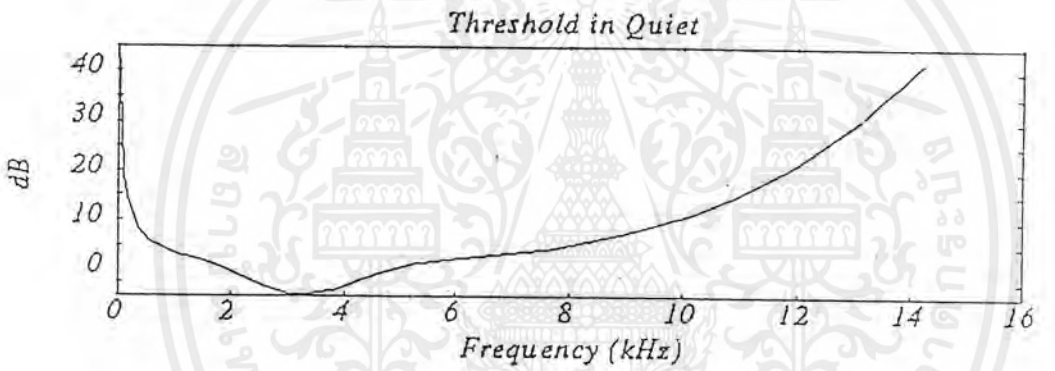
การเข้ารหัสแบบเอ็มเป็กโดยทั่วไป จะตัดข้อมูลเสียงที่จัดเก็บบางส่วนออก แต่สามารถคงรายละเอียดของเสียงที่ได้ยินไว้เท่าเดิม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

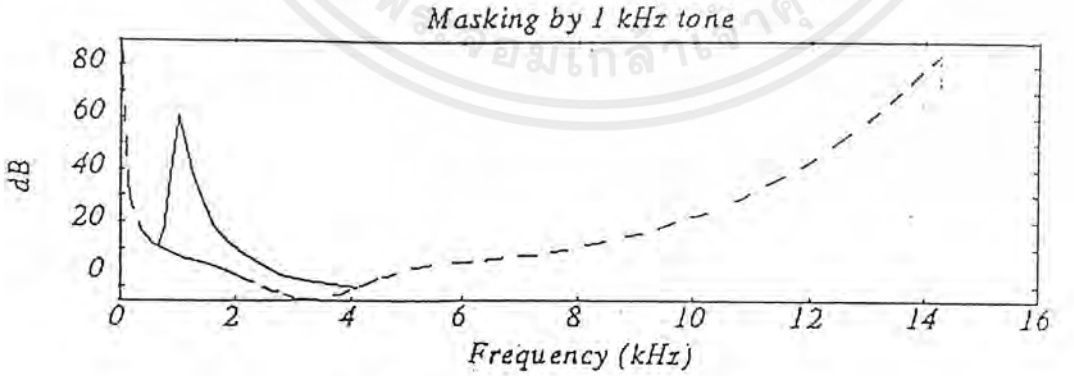
กระบวนการถอดรหัสบิตสตรีม (bitstream) จะนำข้อมูลผ่านกระบวนการปรับข้อมูลย้อนกลับ และทำการรวบรวมข้อมูลแต่ละช่วงความถี่กลับเป็นข้อมูลเสียง

การเข้ารหัสข้อมูลเสียงแบบเอ็มเป็ก ใช้หลักการตัดเสียงบางส่วนที่ฟังไม่ได้ยินออก เนื่องจากความได้เปรียบในการรับฟังเสียงของมนุษย์ที่ไม่สามารถได้ยินเสียงรบกวนภายใต้เงื่อนไขการปิดกั้นการได้ยิน (Auditory masking) การปิดกั้นการได้ยินเป็นคุณสมบัติอย่างหนึ่งในการรับฟังเสียงของมนุษย์ เกิดขึ้นเมื่อเรารับฟังเสียงที่มาจากแหล่งกำเนิดสองแหล่ง จะไม่สามารถได้ยินเสียงจากแหล่งกำเนิดที่ค่อยกว่า ถ้าแหล่งกำเนิดทั้งสองมีความถี่ใกล้เคียงกัน

ความสามารถในการได้ยินเสียงของมนุษย์ จะมีช่วงความถี่ที่สามารถรับฟังประมาณ 20 เฮิร์ต ถึง 20 กิโลเฮิร์ต และความถี่เสียงพูดปกติ 500 เฮิร์ตถึง 2 กิโลเฮิร์ต โดยช่วงเสียงที่มนุษย์สามารถรับฟังได้ไวที่สุดอยู่ในช่วง 2 ถึง 4 กิโลเฮิร์ต



รูปที่ 2.2 แสดงระดับความดังเสียงที่เริ่มได้ยินที่ความถี่ต่าง ๆ



รูปที่ 2.3 แสดงผลของการปิดกั้นเสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการทดลองเรื่องการปิดกั้นเสียงโดยให้คนฟังเสียงที่ความถี่ 1 กิโลเฮิร์ต ความดัง 80 เดซิเบล แล้วเล่นเสียงจากแหล่งกำเนิดอีกแหล่ง วัดระดับความดังของเสียงที่ได้ยินจากแหล่งกำเนิดที่สอง แปรความถี่ วาดกราฟ ได้ดังรูปที่ 2.3 ซึ่งแสดงให้เห็นว่าที่ความถี่ใกล้กับความถี่ 1 กิโลเฮิร์ต ระดับความดังเสียงของแหล่งกำเนิดที่สองต้องมีค่าสูงเพื่อให้รับฟังเสียงได้

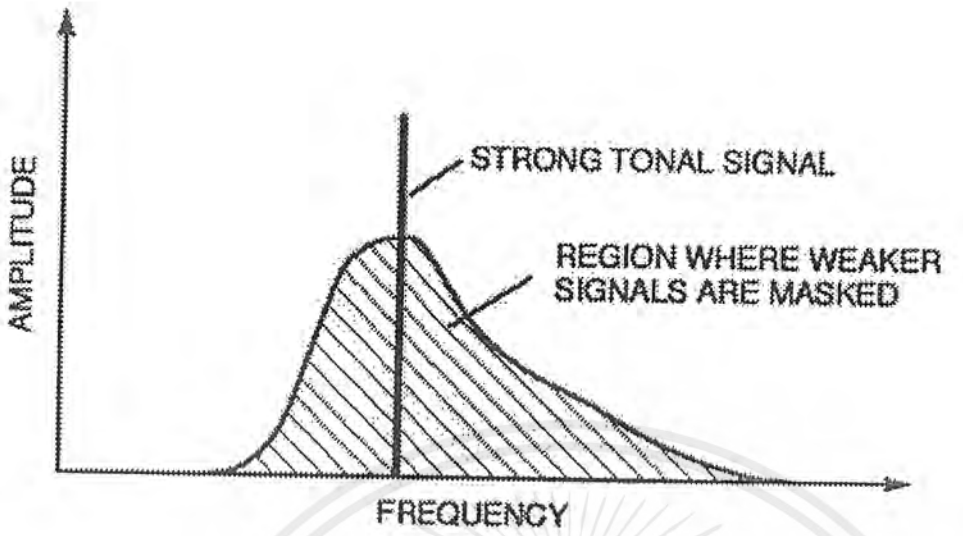
สามารถสรุปจากข้อมูลทั้งหมดได้ว่า มนุษย์สามารถรับฟังเสียงเป็นช่วงความถี่ และรับพลังงานเสียงได้ไม่เท่ากันในแต่ละย่านความถี่ ซึ่งเราเรียกว่าย่านความถี่วิกฤต (critical band) และในแต่ละย่านความถี่มนุษย์จะแยกสัญญาณจากแหล่งกำเนิดต่าง ๆ ออกจากกันได้ยาก ตารางที่ 2.1 แสดงความกว้างของย่านความถี่วิกฤตย่านต่าง ๆ

ตารางที่ 2.1 แสดงความกว้างของย่านความถี่วิกฤตย่านต่าง ๆ

Band Number	Frequency (Hz)	Band Number	Frequency (Hz)
0	50	14	1,970
1	95	15	2,340
2	140	16	2,720
3	235	17	3,280
4	330	18	3,840
5	420	19	4,690
6	560	20	5,440
7	660	21	6,375
8	800	22	7,690
9	940	23	9,375
10	1,125	24	11,625
11	1,265	25	15,375
12	1,500	26	20,250
13	1,735		

หลักการนี้ การเข้ารหัสแบบเอ็มเป็กยึดถือเป็นสำคัญ โดยแบ่งสัญญาณเสียงออกเป็นย่านความถี่ย่อยประมาณเท่ากับย่านความถี่วิกฤตแล้วจึงปรับข้อมูล (quantized) ในแต่ละย่านความถี่ตามความสามารถในการได้ยินในแต่ละย่านความถี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 แสดงการปิดกั้นเสียงที่ไม่ได้ยินในย่านความถี่ย่อยหนึ่ง ๆ ของการเข้ารหัสเอ็มเป็ก

2.4 รูปแบบของข้อมูลเอ็มเป็ก (Formatting)

เป็นการจัดรูปแบบข้อมูลให้ตรงตามมาตรฐาน ซึ่งมาตรฐานการจัดเรียงข้อมูลแบบเอ็มเป็กเลเยอร์ 3 แสดงได้ดังรูปที่ 2.5

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

รูปที่ 2.5 แสดงรูปแบบข้อมูลเอ็มเป็ก เลเยอร์ 3

1. ส่วนหัวข้อมูล (Header) เป็นข้อมูลขนาด 32 บิต แสดงลักษณะทั่วไปของไฟล์นั้น ๆ ซึ่งข้อมูลในส่วนนี้ประกอบด้วยบิตต่าง ๆ ดังรูปที่ 2.6
2. ส่วนตรวจสอบความผิดพลาด เป็นข้อมูลขนาด 16 บิต ตรวจสอบพาริตี้ของข้อมูลที่ถูกเข้ารหัส
3. ส่วนข้อมูลข้างเคียง เป็นข้อมูลขนาด 17 หรือ 32 ไบต์ (17 ไบต์ สำหรับการเข้ารหัสแบบโมโนเดี่ยว และ 32 ไบต์ สำหรับแบบอื่นๆ) ข้อมูลในส่วนนี้เก็บองค์ประกอบต่าง ๆ ที่ใช้ในการถอดรหัสโดยตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

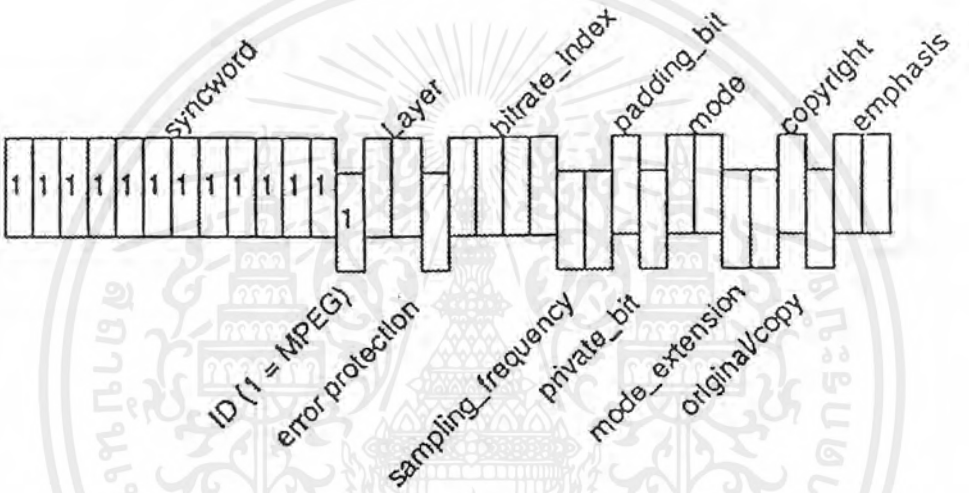
4. ข้อมูลหลัก จะไม่ถูกจำกัดความยาวข้อมูลขึ้นอยู่กับความถี่สุ่มตัวอย่างและอัตราการส่งข้อมูลดังสมการ

$$N = 144 \times \frac{\text{bitrate}}{\text{sampling frequency}}$$

เมื่อ N คือ ความยาวข้อมูลระหว่างสองซินค์เวิร์ด (syncword) ที่อยู่ติดกัน

bitrate คือ อัตราการส่งข้อมูล

sampling frequency คือ ความถี่สุ่มตัวอย่าง



รูปที่ 2.6 แสดงบิตต่าง ๆ ในส่วนหัวข้อมูล

- หมายเหตุ สามารถหารายละเอียดมาตราฐานการบีบอัดข้อมูลเพิ่มเติมได้จาก ห้องสมุดสำนักงานมาตราฐานอุตสาหกรรม (สมอ.)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

MAS 3507D

3.1. บทนำ

MAS 3507D เป็นวงจรรวมที่ผลิตโดยบริษัท Micronas Intermetall เพื่อทำหน้าที่ในการถอดรหัสออดิโอ เอ็มเป็ก 1 หรือ 2 ในเลเยอร์ 2 หรือ 3 โดยการถอดรหัสภายในตัววงจรรวมจะใช้ความสามารถของตัวประมวลผลข้อมูลเชิงเลข (Digital Signal Processor : DSP)

MAS 3507D มีความสามารถที่โดดเด่นอีกประการหนึ่งคือ ความสามารถในการประยุกต์ใช้งานเป็นอุปกรณ์พกพาได้ เนื่องจากใน MAS 3507D ได้รวมเอา DC/DC up-converter เอาไว้ภายใน เพื่อใช้งานกับแหล่งพลังงานที่มีแรงดันไม่คงที่เช่น แบตเตอร์รี่ เป็นต้น อีกทั้งยังเป็นอุปกรณ์ที่มีการใช้พลังงานที่ต่ำมาก ทำให้สามารถประยุกต์การใช้งานกับอุปกรณ์ซึ่งมีแหล่งพลังงานที่จ่ายกระแสต่ำได้เป็นอย่างดี

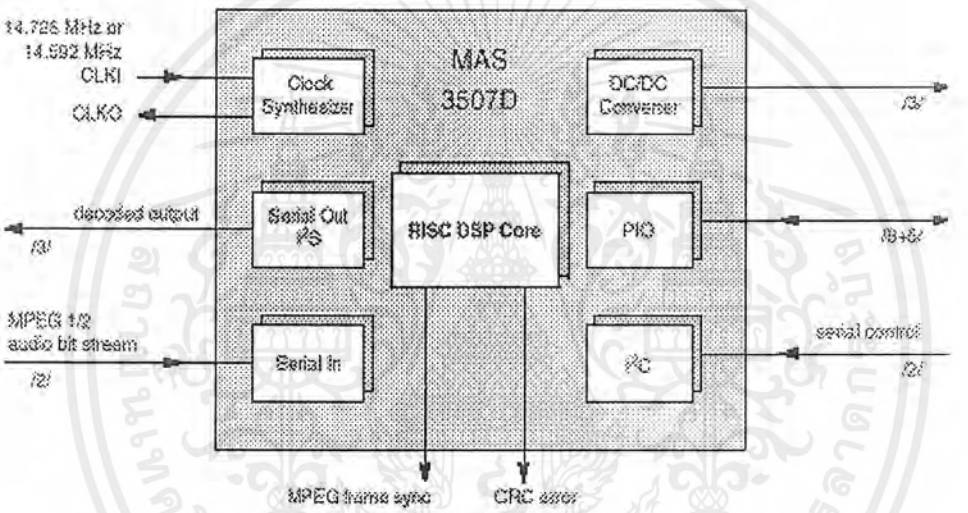
มาตรฐานการบีบอัดข้อมูลแบบเอ็มเป็ก (ISO 11172-3) ได้มีการแบ่งระดับการบีบอัดข้อมูลที่เป็นมาตรฐานอยู่ 3 เลเยอร์ ซึ่งมีความซับซ้อนและยุ่งยากแตกต่างกันออกไปขึ้นอยู่กับคุณภาพของเสียงและอัตราส่วนการบีบอัดข้อมูล การบีบอัดข้อมูลเลเยอร์ 3 เป็นมาตรฐานที่ได้รับความสนในมากที่สุดเพราะมีอัตราการบีบอัดข้อมูลสูงสุด (ประมาณ 12:1) และคุณภาพของเสียงที่ได้ยังคงคุณภาพเสียงในระดับ CD ไว้

ลักษณะเด่นที่สำคัญ

- เป็นชิปเดี่ยวสำหรับถอดรหัสข้อมูลเอ็มเป็ก1 หรือ 2 เลเยอร์ 2 หรือ 3
- การถอดรหัสตรงตามมาตรฐาน ISO
- มีความสามารถในการถอดรหัสเอ็มเป็ก 2 เลเยอร์ 3 สำหรับอัตราบิตต่ำ (MPEG2.5)
- สนับสนุนบิทสตรีมที่มีบิตเรทที่ปรับค่าได้ แบบอะซิงโครนัส
- มีบัฟเฟอร์ (buffer) ภายในสำหรับเก็บข้อมูลเอ็มเป็ก เพื่อไว้สำหรับการถอดรหัสทำให้สามารถถอดรหัสได้อย่างต่อเนื่อง
- มีการใช้สัญญาณในการร้องขอข้อมูลเอ็มเป็ก เพื่อเติมข้อมูลในบัฟเฟอร์
- ส่งข้อมูลออดิโอเอาท์พุทผ่านบัส I^2S

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สามารถปรับระดับเสียง ระดับเสียงเบส และระดับเสียงแหลมได้ตามความต้องการ
- สัญญาณนาฬิกาการแซมปลิ้งที่เอาท์พุทจะถูกสร้างและควบคุมจากภายใน
- ติดต่อและควบคุมชิปผ่าน I^2C บัส
- สามารถเข้าถึงข้อมูลสถานะได้โดยผ่านขา PIO หรือ I^2C
- มีการแสดงข้อมูลในส่วน 'CRC Error' และ 'MPEG Frame Synchronization'
- มีการจัดการด้านพลังงานเพื่อลดการใช้พลังงานที่ความถี่การแซมปลิ้งต่ำ
- ช่วงแรงดันไฟเลี้ยง : 1.6 V ถึง 3.6 V



รูปที่ 3.1 แสดงบล็อกไดอะแกรมของ MAS 3507D

3.2 การทำงานและรายละเอียดภายใน MAS 3507D

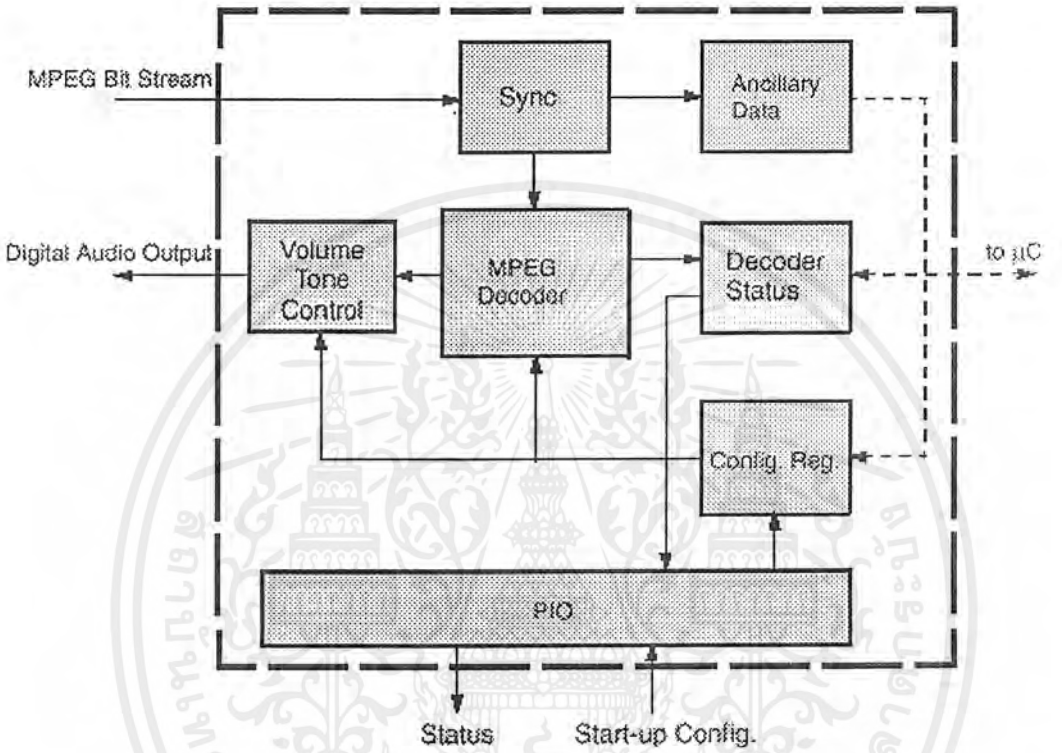
3.2.1 ตัวประมวลผลข้อมูลเชิงเลข (DSP)

รูปที่ 3.2 แสดงบล็อกไดอะแกรมส่วนวงจรถอดรหัสภายในของ MAS 3507D ซึ่งประกอบด้วย ตัวประมวลผลเชิงเลขความสามารถสูงที่ใช้สถาปัตยกรรมแบบ RISC (high performance RISC Digital Signal Processing) ซึ่งทำหน้าที่ในการถอดรหัสข้อมูล โดยตัวประมวลผลภายในทำงานด้วยเวิร์ด (word) ความยาว 20 บิตและขยายถึง 32 บิต ในแอคคิวมูเลเตอร์ (accumulator)

และเนื่องจากคำสั่งของ DSP นี้ออกแบบมาสำหรับการถอดรหัสโดยเฉพาะ จึงมีความเหมาะสมอย่างยิ่งสำหรับการถอดรหัสข้อมูลทางออดิโอ ด้วยเหตุนี้ จึงมีความต้องการพื้นที่เพียงเล็ก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

น้อยสำหรับ RAM และ ROM ภายใน นอกจากนั้นการส่งผ่านข้อมูลทั้งทางเข้าสู่ตัวชิปและออกจากตัวชิปจะกระทำผ่าน DMA ซึ่งจะไม่ไปขัดขวางการทำงานหลักคือการถอดรหัส ทำให้สามารถถอดรหัสข้อมูลต่อเนื่องได้อย่างราบรื่น



รูปที่ 3.2 บล็อคไดอะแกรมของตัวถอดรหัส MPEG

3.2.2 เฟิร์มแวร์ (Firmware)

ภายใน MAS 3507D จะมีโปรแกรมซึ่งทำหน้าที่ถอดรหัสข้อมูลฝังอยู่ภายในตัวชิปอยู่แล้ว โดยโปรแกรมจะทำการถอดรหัสข้อมูลเอ็มเป็กตาม ขั้นตอนการถอดรหัส คือ

- การซิงโครไนซ์
- การดึงข้อมูลข้างเคียง (side information extraction)
- การถอดรหัสแบบฮัฟแมน
- การดึงข้อมูลเพิ่มเติม (ancillary data extraction)
- การควบคุมระดับเสียงและโทน (tone)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3 ความสามารถในการโหลดโปรแกรมภายนอก

MAS 3507D มีความสามารถในปฏิบัติโปรแกรมอื่นนอกเหนือจากโปรแกรมที่อยู่ภายใน หากต้องการใช้ความสามารถพิเศษนี้ ก่อนอื่นต้องทำการส่งโปรแกรมไปเก็บในหน่วยความจำผ่านทาง I^2C และในการสั่งให้ทำงานในโปรแกรมนี้ก็จะต้องสั่งงานผ่านทาง I^2C เช่นกัน เงื่อนไขอีกสองประการที่จำเป็นต้องพิจารณาคือ ขนาดของโปรแกรมนี้ต้องมีขนาดไม่เกิน 1 กิโลเวิร์ด และขณะที่ทำงานโปรแกรมนี้ จะไม่สามารถทำการถอดรหัสข้อมูลได้ตามปกติ

3.2.4 การประมวลผลในด้านความถี่

การควบคุมระดับเสียงและมิกเซอร์

การควบคุมระดับเสียงของ MAS3507D อยู่ในรูปแบบดิจิตอล นอกจากนั้นยังสามารถควบคุมความสมดุลย์ (balance) ซึ่งสามารถปรับค่าความดังของสัญญาณซ้ายและขวาได้ละเอียดถึง 20 บิตโดยผ่านทาง I^2C บัส

การตัดเสียงและบายพาสส่วนโทนคอนโทรล (Tone Control)

ในการตัดเสียง (mute) ของตัวถอดรหัสนั้นสามารถควบคุมได้โดยใช้บิตข้อมูลเพียงบิตเดียวโดยการทำงานจะเป็นไปอย่างรวดเร็วและไม่เปลี่ยนระดับเสียง (Volume) ที่ใช้งานอยู่ และนอกจากนั้นยังสามารถบายพาสส่วนของโทนคอนโทรลที่ปรับระดับเสียงทุ้มแหลม และปรับระดับเสียงได้ในกรณีที่ไม่ต้องการใช้งาน

การควบคุมระดับเสียงทุ้ม-แหลม

การควบคุมระดับเสียงทุ้ม - แหลมจะสามารถควบคุมได้ในช่วง $\pm$ 15 dB ดังตาราง 3 และเพื่อป้องกันการโอเวอร์โฟลว์ (overflow) หรือการคลิปปิ้ง (clipping effect) จึงได้มีการเพิ่มส่วนพรีสเกลเลอร์ (prescaler) เพื่อลดอัตราขยายของระดับเสียงที่สูงเกินไป ดังนั้นที่อัตราขยายสูงสุด +15 dB ก็จะไม่มีการคลิป

ตาราง 3.1 การตั้งค่าสำหรับแมทริกซ์ดิจิตอล volume

ดัทออฟ	Bass	Treble
เอ็มเปิก 1	100 Hz	10 kHz
เอ็มเปิก 2	200 Hz	10 kHz
เอ็มเปิก 2.5	400 Hz	10 kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากช่วงความถี่แตกต่างกันในเอ็มเป็ก 1, เอ็มเป็ก 2 และ เอ็มเป็ก 2.5 จึงทำให้มีความถี่คัทออฟของเสียงทุ้มต่างกันไปด้วย

3.2.5 การจัดการสัญญาณนาฬิกา

MAS 3507D จะทำงานโดยใช้สัญญาณนาฬิกาที่ความถี่ 14.592 MHz หรือ 14.725 MHz สัญญาณ CLKI จะทำหน้าที่เป็นตัวอ้างอิงของการสังเคราะห์สัญญาณนาฬิกาซึ่งให้สัญญาณนาฬิกาภายในระบบ และจากการอ้างอิงสัญญาณนาฬิกาอินพุท (CLKI) จะทำให้ตัวชิบสร้างสัญญาณนาฬิกาเอาท์พุท (CLKO) ที่ซิงโครไนซ์กับความถี่การแซมเปิลออดิโอของบิตสตรีมที่ถูกถอดรหัส ซึ่งสัญญาณนาฬิกานี้จะเป็นสัญญาณนาฬิกาหลัก (master clock) ให้กับ D/A Converter ภายนอก (DAC บางตัวต้องการสัญญาณนาฬิกาหลักที่มีความสัมพันธ์ที่ถูกกำหนดโดยความถี่การแซมปลิ้ง) ซึ่งความถี่ที่ได้เป็นไปตามตารางที่ 3.2 นอกจากนั้นยังสามารถกำหนดให้มีการปรับความถี่ของสัญญาณนาฬิกาเอาท์พุทให้เปลี่ยนตามความถี่แซมปลิ้งได้ด้วย

ตารางที่ 3.2 ความถี่ CLKO

f_s / kHz	CLKO / MHz scaler on	CLKO / MHz scaler off
48, 32	24.576	24.576
44.1	22.5792	22.5792
24, 16	12.288	24.576
22.05	11.2896	22.5792
12, 8	6.144	24.576
11.025	5.6448	22.5792

3.2.6 การทำงานของส่วนจ่ายไฟ

MAS 3507D มีวงจรดีซีทูดีซีคอนเวอร์เตอร์ (DC/DC Control) อยู่ภายในเพื่อให้สามารถใช้งานกับแหล่งจ่ายไฟแบบแบตเตอรี โดยจะทำงานแบบปรับแรงดันขึ้น (up convertor)

ตัวตรวจสอบแรงดัน (Voltage Monitor)

ตัวตรวจสอบแรงดัน จะเปรียบเทียบแรงดันอินพุทที่ขา VSENS ด้วยค่าอ้างอิงภายในซึ่งปรับค่าได้โดยบัส I^2C และได้เอาท์พุทที่ขา PUP ซึ่งจะไม่แอคทีฟเมื่อแรงดันที่ขา VSENS ตกลงต่ำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กว่าค่าแรงดันอ้างอิง การทำงานของตัวตรวจสอบแรงดันสามารถทำงานได้อย่างอิสระโดยไม่ขึ้นอยู่การทำงานของส่วนดีซีทูดีซีคอนเวอร์เตอร์ ซึ่งการทำงานของส่วนนี้มีประโยชน์มากในการทำงานที่อาศัยแหล่งจ่ายไฟแบบแบตเตอร์รี่

ในการใช้งาน จะนำเอาท์พุทที่ได้จากส่วนตรวจสอบแรงดันนี้ไปป้อนให้กับขาเปิดการทำงานส่วนดิจิตอลของตัวชิป เพื่อป้องกันไม่ให้ตัวชิปทำงานในกรณีที่แรงดันไฟต่ำเกินไป ซึ่งอาจเกิดความเสียหายได้

ดีซีทูดีซีคอนเวอร์เตอร์

ดีซีทูดีซีคอนเวอร์เตอร์ของ MAS 3507D ทำหน้าที่สร้างแหล่งจ่ายแรงดันคงที่ เพื่อให้ตัวชิปได้รับแรงดันที่คงที่แม้จะใช้แหล่งจ่ายจากแบตเตอรี่สำหรับใช้ในการประยุกต์ใช้งานในแบบพกพา ซึ่งตัวดีซีทูดีซีคอนเวอร์เตอร์นี้ถูกออกแบบมาสำหรับแหล่งจ่ายที่เป็นแตเตอร์รี่ 2 ก้อนโดยเฉพาะ

การแสตนบาย (Stand-by)

การทำงานของ MAS 3507D ทั้งในส่วนดิจิตอลและดีซีทูดีซีคอนเวอร์เตอร์ จะมีขา เริ่มการทำงานแยกออกจากกัน (WSEN, DCEN) ด้วยเหตุนี้เราจึงสามารถใช้ดีซีทูดีซีคอนเวอร์เตอร์ เพื่อป้อนให้แก่วงจรส่วนอื่นๆ ในขณะที่ไม่มีการถอดรหัสได้

3.2.7. การเชื่อมต่อ (Interface)

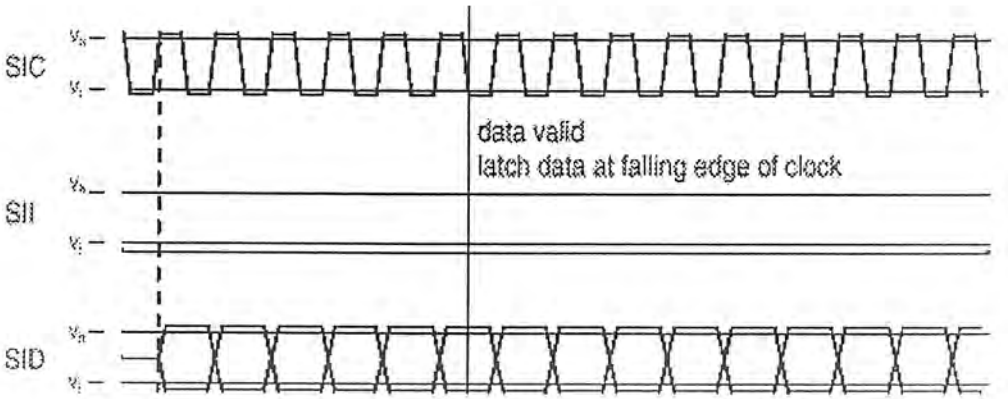
การเชื่อมต่ออุปกรณ์ภายนอกกับ MAS 3507D มีการเชื่อมต่อกันหลายรูปแบบ เช่น ผ่านทาง I^2C เป็นการเชื่อมต่อเพื่อควบคุม หรือการส่งข้อมูลระหว่างคอนโทรลเลอร์กับ MAS 3507D แต่สำหรับอินพุทบิตสตรีมเอ็มเป็กและเอาท์พุทที่เป็นดิจิตอลออดิโอสำหรับข้อมูลออดิโอที่ถอดรหัสแล้ว จะใช้การเชื่อมต่อแบบ I^2S และยังมีการใช้การเชื่อมต่อแบบขนาน (PIO) สำหรับการตรวจสอบและการเลือกโหมดการทำงานด้วย

การเชื่อมต่อบิตสตรีมเอ็มเป็ก

การเชื่อมต่ออินพุทบิตสตรีมเอ็มเป็กจะใช้ 3 ขา คือ SIC, SII และ SID โดยแต่ละขาจะส่งข้อมูลต่างกันดังนี้

- SIC ส่งสัญญาณนาฬิกา
- SID ส่งตัวสตรีมข้อมูล
- SII ส่งสัญญาณบอกเฟรมข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 timing diagram ของ I^2S อินพุท

การเชื่อมต่อเอาท์พุทออดิโอ

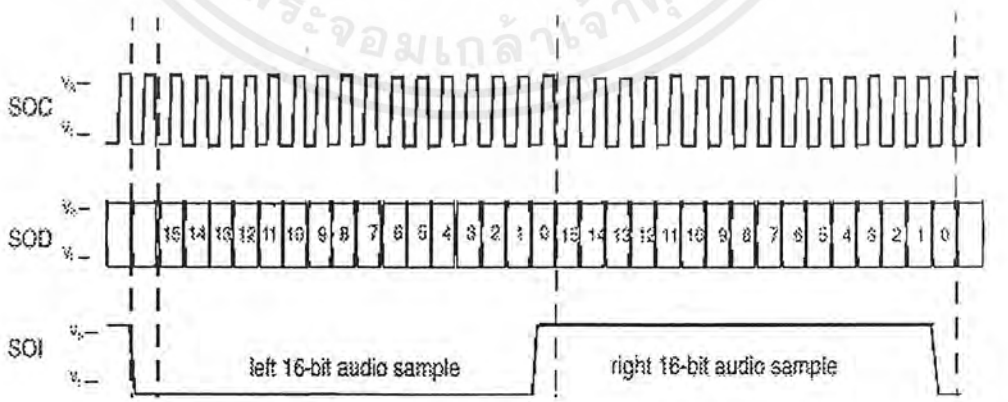
การเชื่อมต่อเอาท์พุทออดิโอของ MAS 3507D เป็นมาตรฐานการเชื่อมต่อแบบ I^2S สามารถเลือกการเชื่อมต่อได้ 2 รูปแบบ (16 บิตหรือ 32 บิต) โดยการทำงานของแต่ละโหมดเป็นดังนี้

- โหมด 16 บิต

การทำงานของสัญญาณต่างๆ เป็นดังรูปที่ 3.4

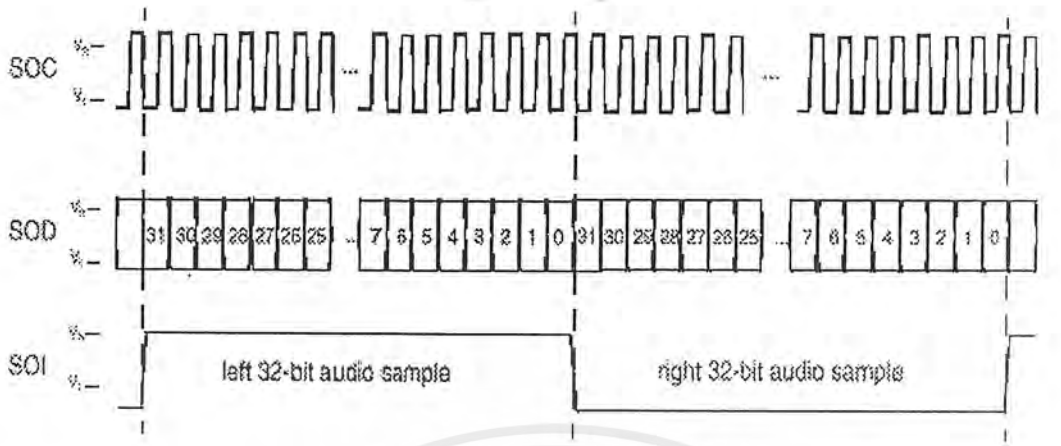
- โหมด 32 บิต

ในโหมด 32 บิตนี้ตัวชิปจะสร้างเอาท์พุทขนาด 32 บิตต่อข้อมูล 1 แซมเปิ้ล แต่จะเป็นข้อมูลเสียงเพียง 20 บิตแรกเท่านั้น ส่วนที่เหลืออีก 12 บิตจะเป็น 0 การทำงานของสัญญาณต่างๆ เป็นดังรูปที่ 3.5



รูปที่ 3.4 timing diagram ของการเชื่อมต่อ SDO ในโหมด 16 บิต/แซมเปิล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 timing diagram ของการเชื่อมต่อ SDO ในโหมด 32 บิต/แชมเปิล

Start-up Configuration

โดยพื้นฐานแล้ว MAS 3507D สามารถที่จะทำงานได้โดยไม่ต้องมีคอนโทรลเลอร์คอยควบคุม ข้อมูลสถานะส่วนมากสามารถกำหนดได้จากการกำหนดสถานะขณะเปิดเครื่อง (Start-up Configuration) การกำหนดสถานะขณะเปิดเครื่องนี้จะกระทำผ่านทางการเชื่อมต่อแบบขนาน (PIO) และหลังจากเปิดเครื่องแล้วขาเหล่านี้จะทำหน้าที่เป็นเอาท์พุทเพื่อแสดงข้อมูลเกี่ยวกับไฟล์เอ็มเป็กที่ถอดรหัสได้

ตารางที่ 3.3 แสดงถึงการกำหนดโหมดการทำงานเริ่มต้นผ่านทางขา PI ต่างๆ

PIO PIN	'0'	'1'
PI8	หาร CLKO ด้วย 1,2 หรือ 4 (ตาม MPEG1, 2 หรือ 2.5)	CLKO คงที่ที่ 24.576 หรือ 22.5792 MHz
PI4	สัญญาณนาฬิกาอินพุท 14.725 MHz	สัญญาณนาฬิกาอินพุท 14.592 MHz
PI3	Enable Layer3	Disable Layer3
PI2	Enable Layer2	Disable Layer2
PI1	เอาท์พุท SDO 32 บิต	เอาท์พุท SDO 16 บิต
PI0	อินพุท : โหมดมัลติมิเดีย (PLL off)	อินพุท : โหมดกระจายเสียง (PLL on)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ตาราง 3.4 สัญญาณเอาท์พุทของ PIO ในการถอดรหัส MPEG

PIO Pin	Name	คำอธิบาย
PI19	Demand PIN	
	%0	No input data exp.
	%1	Input data request
PI18, PI17	MPEG INDEX	
	%00	MPEG2.5
	%01	Reserved
	%10	MPEG2
	%11	MPEG1
PI13, PI12	MPEG Layer ID	
	%00	Reserved
	%01	Layer3
	%10	Layer2
	%11	Layer1
PI8	MPEG CRC-ERROR	
	%0	No error
	%1	CRC-error decoding not successful
PI4	MPEG-FRAME-SYNC	See following text
PI3, PI2	Sampling frequency	In KHz
	%00	44.1 / 22.1 / 11.0
	%01	48 / 24 / 12
	%10	32 / 16 / 8
	%11	reserve
PI1, PI0	Deemphasis	
	%00	None
	%01	50/15 μs
	%10	reserved
	%11	CCITT J.17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
36876

3.3. การเชื่อมต่อการควบคุม (Control Interface)

3.3.1. การเชื่อมต่อบัส I^2C

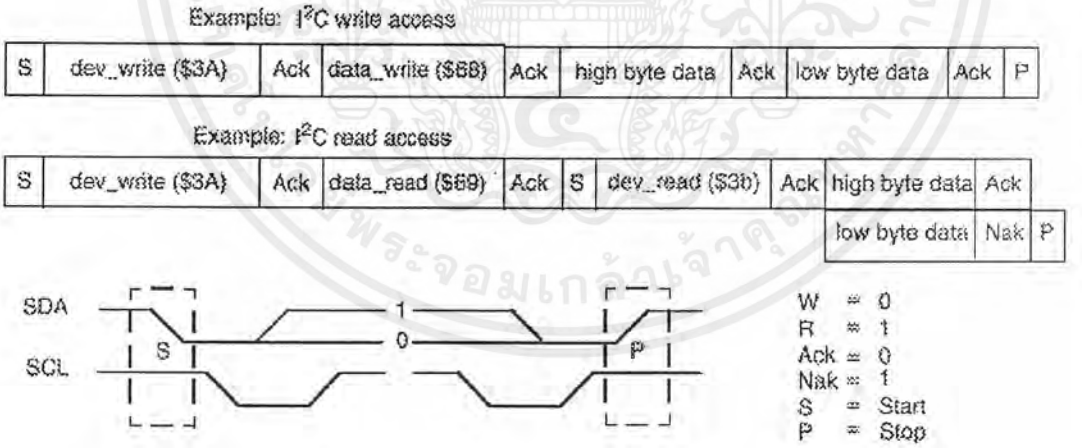
ทั่วไป

การติดต่อสื่อสารระหว่าง MAS 3507D และคอนโทรลเลอร์ภายในจะกระทำผ่านบัส I^2C โดย I^2C จะถ่ายโอนข้อมูลที่มีความยาวเวิร์ดต่ำสุด คือ 16 บิต การเชื่อมต่อจะใช้ ระดับของตำแหน่งย่อย (subaddress) เพียงระดับเดียว แผนแอดเดรสของ MAS 3507D (device address) แสดงในตารางที่ 3.5

ตารางที่ 3.5 แสดงแผนแอดเดรส I^2C

A7	A6	A5	A4	A3	A2	A1	W/R
0	0	1	1	1	0	1	0/1

รีจิสเตอร์ข้อมูลและรีจิสเตอร์ควบคุมของ I^2C ของ MAS 3507D จะมีขนาดข้อมูล 16 บิต สามารถเข้าถึงได้โดยการเขียน/อ่าน 2 เวิร์ดข้อมูล 8 บิต



รูปที่ 3.6 โปรโตคอลของบัส I^2C ของ MAS 3507D

ตำแหน่งย่อย (Subaddress)

การควบคุมผ่านทาง I^2C คอนโทรลเลอร์ต้องทำการส่งคำสั่ง เพื่อแสดงความต้องการว่าจะอ่านหรือเขียนข้อมูล ดังแสดงในตารางที่ 3.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.6 แสดงตำแหน่งย่อย

ตำแหน่งย่อย	คำอธิบาย
$68 / write	คอนโทรลเลอร์เขียนรีจิสเตอร์ข้อมูลของ MAS 3507D
$69 / read	คอนโทรลเลอร์อ่านจากรีจิสเตอร์ข้อมูลของ MAS 3507D
$6A / write	คอนโทรลเลอร์เขียนรีจิสเตอร์ควบคุมของ MAS 3507D

แอดเดรส ($6A) ใช้ในการควบคุมพื้นฐาน เช่น รีเซทและเลือกชิ้นงาน แอดเดรสอื่นใช้สำหรับถ่ายโอนข้อมูลจาก/ถึง MAS 3507D

รีจิสเตอร์ I^2C

รีจิสเตอร์ควบคุม I^2C

รีจิสเตอร์ควบคุม I^2C เป็นรีจิสเตอร์ที่เขียนได้อย่างเดียวเท่านั้นและ จุดประสงค์ของรีจิสเตอร์นี้คือการรีเซททางซอฟท์แวร์ของ MAS 3507D

ตารางที่ 3.7 แสดงการกำหนดบิทของรีจิสเตอร์ควบคุม

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
X	X	X	X	X	X	X	R	0	0	0	0	T3	T2	T1	T0

X = don't care, R = reset, T3...T0 = ตัวกำหนดในการโหลดซอฟท์แวร์

รีเซททางซอฟท์แวร์จะกระทำโดยการเขียนข้อมูล 16 บิตให้ MAS 3507D โดยกำหนดให้บิตที่ 8 เป็น 1 สำหรับ 4 บิตท้ายเป็นตัวกำหนดซึ่งใช้ในการดาวน์โหลดซอฟท์แวร์ (download software) ในการถอดรหัสเอ็มเป็กโดยทั่วไปนั้น บิทเหล่านี้ต้องเซทเป็น "0" เสมอ

รีจิสเตอร์ข้อมูล I^2C

รีจิสเตอร์ข้อมูลของ I^2C สามารถอ่าน (subaddress data_read) และเขียนได้(subaddress data_write) โดยมีความยาว 16 บิต การถ่ายโอนข้อมูลจะกระทำที่ MSB (m) เป็นบิตแรก

ตารางที่ 3.8 การกำหนดบิตของรีจิสเตอร์ข้อมูล

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
m															1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 โครงสร้างคำสั่ง

การควบคุม MAS 3507D จะกระทำอย่างสมบูรณ์ผ่านรีจิสเตอร์ข้อมูล I^2C โดยใช้คำสั่งพิเศษ คำสั่งจะถูกนำไปปฏิบัติโดย MAS 3507D ในขณะทำงานปกติที่ไม่มีการสูญเสียหรือเกิดการอินเตอร์รัพของข้อมูลที่กำลังมาหรือสตรีมข้อมูลออดิโอที่กำลังออกไป คำสั่ง I^2C เหล่านี้คอนโทรลเลอร์สามารถเข้าถึงสถานะภายใน,หน่วยความจำภายใน,รีจิสเตอร์ควบคุมฮาร์ดแวร์ภายนอก และแม้กระทั่งดาวน์โหลดของซอฟท์แวร์

เฟิร์มแวร์ของ MAS 3507D จะสแกน (scan) การเชื่อมต่อ I^2C แบบคาบเวลาและตรวจสอบคำสั่งที่ค้างอยู่หรือคำสั่งใหม่ อย่างไรก็ตาม เนื่องจากบางส่วนของเฟิร์มแวร์มีความสำคัญมาก จึงต้องมีการคาดคะเนช่วงเวลาแฝงในการตอบสนอง ในทางทฤษฎี ช่วงเวลาการตอบสนองต้องไม่เกิน 4 ms แต่ช่วงเวลาการตอบสนองทั่วไปจะน้อยกว่า 0.4 ms ตารางที่ 3.9 แสดงคำสั่งคอนโทรลเลอร์พื้นฐานซึ่งใช้ได้ใน MAS 3507D

ตารางที่ 3.9 คำสั่งคอนโทรลเลอร์พื้นฐาน

รหัส	คำสั่ง	คำอธิบาย
$0 $1	Run	เริ่มการปฏิบัติของโปรแกรมภายใน (0 หมายความว่า ค้างระบบการทำงาน)
$3	Read Control Information and Ancillary Data	อ่านข้อมูลในบล็อกของข่าวสาร จัดเป็นเวิร์ดที่มีความยาว 16 บิต
$9	write register	สามารถเขียนรีจิสเตอร์ภายในของ MAS 3507D ได้โดยตรงโดยใช้คอนโทรลเลอร์
$A $B	write to memory	สามารถเขียนบล็อกหน่วยความจำ DSP ได้โดยคอนโทรลเลอร์ ลักษณะเด่นนี้อาจจะใช้ในการดาวน์โหลดโปรแกรมสลับกัน
$D	Read register	คอนโทรลเลอร์สามารถอ่านรีจิสเตอร์ภายในของ MAS 3507D
$E $F	Read memory	สามารถอ่านบล็อกหน่วยความจำ DSP ได้โดยคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

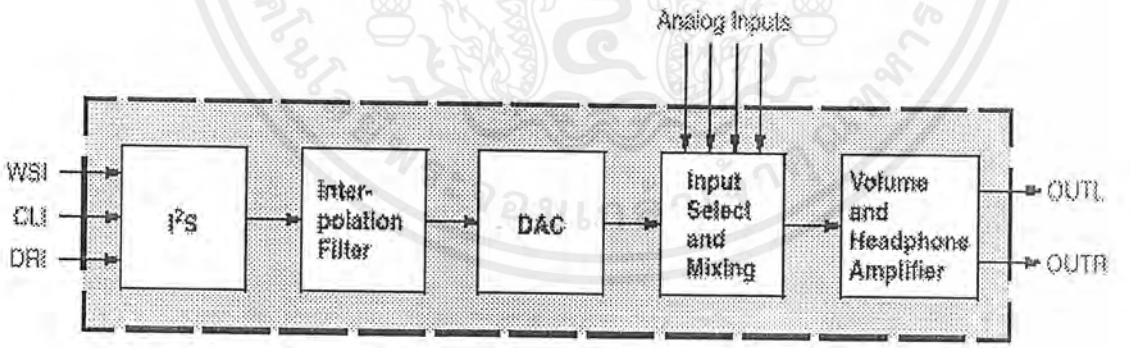
DAC 3550A

4.1 บทนำ

DAC 3550A เป็นชิปเดี่ยวความแม่นยำสูง ภายในประกอบด้วยวงจรแปลงสัญญาณดิจิตอลเป็นอนาลอกสองตัวที่ออกแบบมาสำหรับการประยุกต์ใช้งานทางออดิโอโดยเฉพาะ โดยมันถูกออกแบบมาสำหรับการประยุกต์ใช้งานในทางออดิโอและมัลติมิเดียทุกชนิด เช่น เครื่องเล่น MPEG, เครื่องเล่น CD, เครื่องเล่น DVD, เครื่องเล่น CD-ROM เป็นต้น

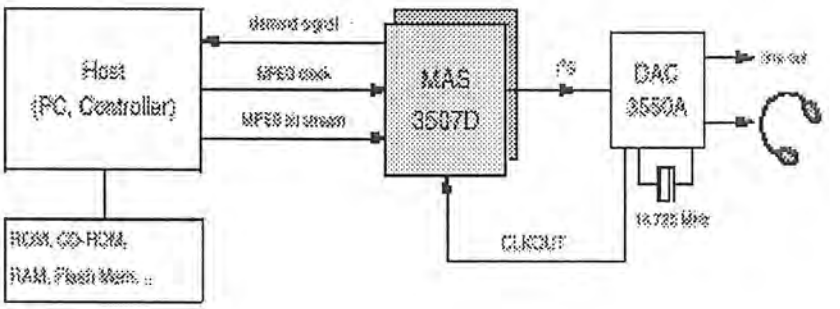
ความสามารถอีกข้อหนึ่งคือมันไม่ต้องใช้คริสตอลสำหรับการประยุกต์ใช้งานมาตรฐานด้วยอัตราแซมเปิลจาก 32 ถึง 48 kHz แต่จะต้องการคริสตอลเพียงสำหรับ

- การตรวจสอบอัตราการแซมเปิลอัตโนมัติที่ต่ำกว่า 32 kHz
- ใช้สัญญาณนาฬิกาเอาท์พุท CLKOUT



รูปที่ 4.1 แสดงบล็อกไดอะแกรมของ DAC 3550A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

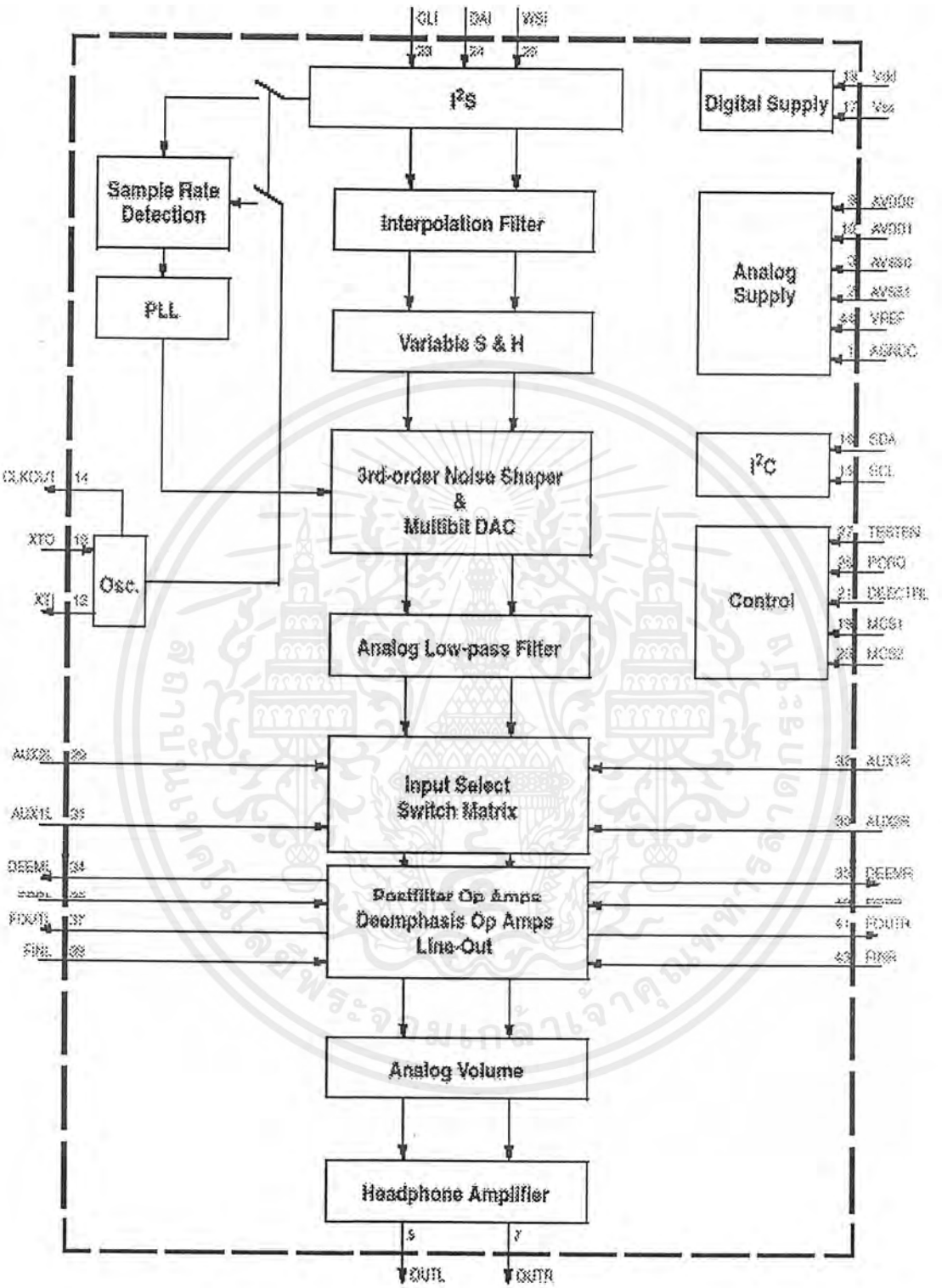


รูปที่ 4.2 แสดงการประยุกต์ใช้งานทั่วไป : เครื่องเล่น MP3

ลักษณะเด่นที่สำคัญ

- ไม่ต้องการสัญญาณนาฬิกาหลัก
- ภายในมีวงจรขยายสำหรับหูฟัง
- SNR 103 dB
- ใช้ระบบบัส I^2C,และบัส I^2S
- มีวงจรออสซิเลเตอร์ภายใน
- สามารถประยุกต์ใช้งานโดยไม่ใช้ I^2C ได้
- อัตราการแซมเปิลต่อเนื่องจาก 8 kHz ถึง 50 kHz
- THD ดีกว่า 0.01%
- ใช้พลังงานต่ำ
- สามารถใช้ได้กับแรงดันได้หลายระดับ ตั้งแต่ 2.7 V ..5.5 V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 แสดงบล็อกไดอะแกรมของ DAC 3550A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การทำงานและรายละเอียดของส่วนต่างๆ

4.2.1. การเชื่อมต่อ I^2S

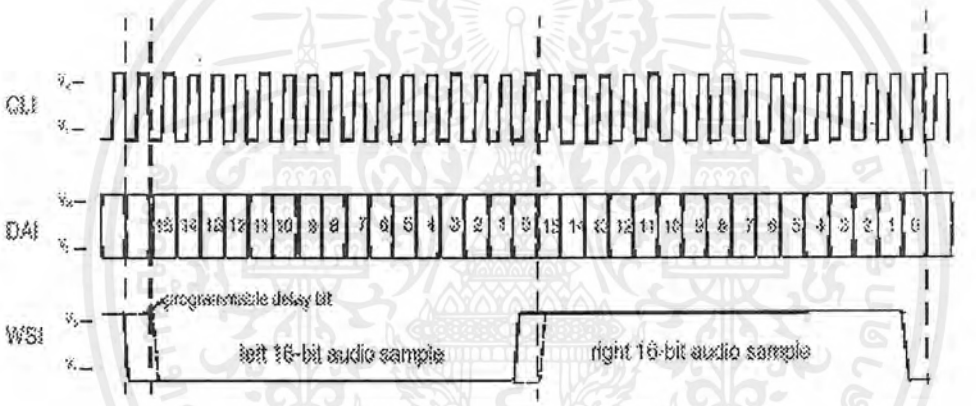
การเชื่อมต่อ I^2S เป็นการเชื่อมต่อทางออดิโอระหว่าง DAC 3550A กับ แหล่งกำเนิดดิจิตอล ออดิโอ เช่น เครื่องเล่น CD/DAT , ตัวถอดรหัส MPEG เป็นต้น

โหมด 16 บิต

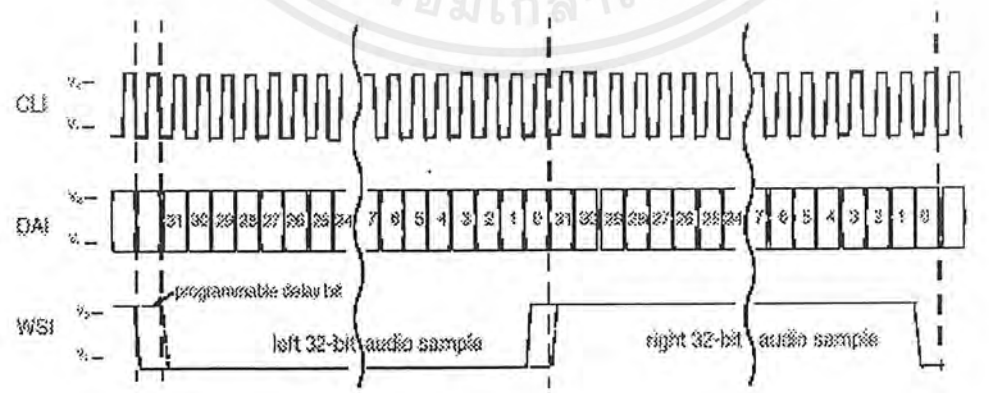
ในกรณีนี้ bit clock คือ 32 × fs_{aodio} ความยาวเวิร์ดสูงสุด 16 บิต

โหมด 32บิต

ในกรณีนี้ bit clock คือ 64 × fs_{aodio} ความยาวเวิร์ดสูงสุด 32 บิต



รูปที่ 4.4 I^2S โหมด 16 บิต (LR_SEL=0)



รูปที่ 4.5 I^2S โหมด 32 บิต (LR_SEL=0)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

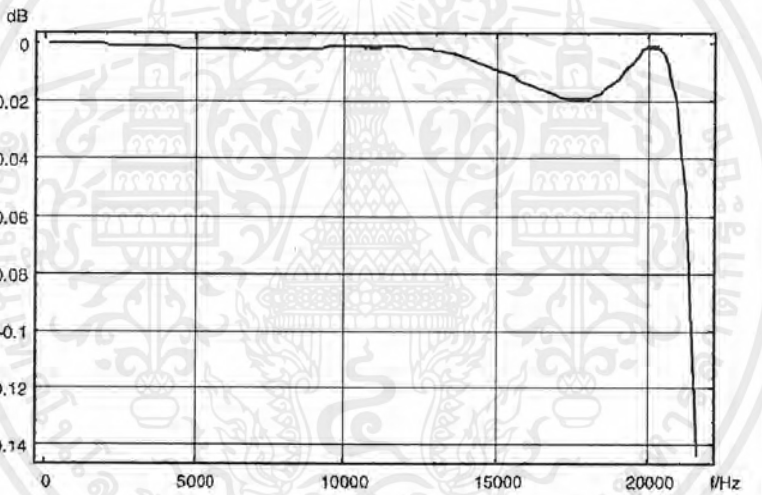
การตรวจจับเพื่อเลือกโหมดการทำงานอัตโนมัติ

การสับเปลี่ยนโหมดระหว่าง 16 และ 32 บิต ไม่จำเป็นต้องควบคุมด้วยอุปกรณ์ภายนอก เพราะ DAC 3550A จะทำการตรวจสอบเอง โดยสำหรับออดิโอคุณภาพสูง แนะนำให้ใช้โหมด 32 บิตของการเชื่อมต่อ I^2S เพื่อที่จะสามารถใช้ งานได้ครอบคลุมย่านความถี่ทั้งหมดที่ใช้งานได้

ในการแปลงข้อมูลสัญญาณจากขา WSI จะเป็นตัวบอกว่าข้อมูลนั้นๆ เป็นของแชนแนลขวาหรือซ้าย

4.2.2 อินเทอร์โพเลชั่นฟิลเตอร์

เป็นฟิลเตอร์ที่แทรกเข้ามาเพื่อจำกัดความถี่ของสัญญาณอินพุทดิจิตอลที่มีความถี่สูงกว่า22 KHz ออกไป



รูปที่ 4.6 การตอบสนองความถี่ของอินเทอร์โพเลชั่นฟิลเตอร์ในช่วงความถี่ : 0...22 kHz

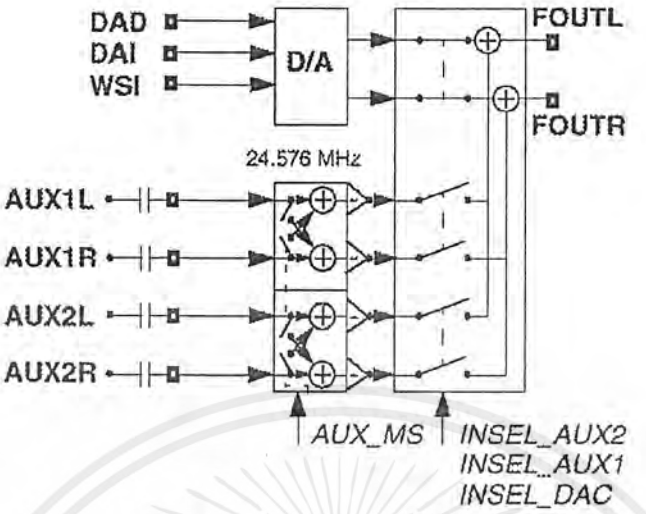
4.2.3 วงจรกรองความถี่ต่ำผ่าน (Low pass Filter)

วงจรกรองความถี่ต่ำผ่านนี้เป็นตัวกรองอันดับหนึ่งที่มีความถี่คัทออฟประมาณ 1.4 MHz ซึ่งจะกำจัดองค์ประกอบความถี่สูงของสัญญาณรบกวนออกไป

4.2.4 ส่วนเลือกช่องสัญญาณและการมิกซ์สัญญาณ

ส่วนนี้ใช้ในการเลือกระหว่างการมิกซ์อินพุทอื่นๆ (auxillary input) และสัญญาณที่มาจาก DAC เพื่อใช้งานในกรณีที่มีสัญญาณจากแหล่งอื่น แมทริกซ์การสวิทช์นี้จะอนุญาตให้เลือกระหว่างโหมดโมโนและโหมดสเตอริโอ แสดงดังรูปที่ 4.7

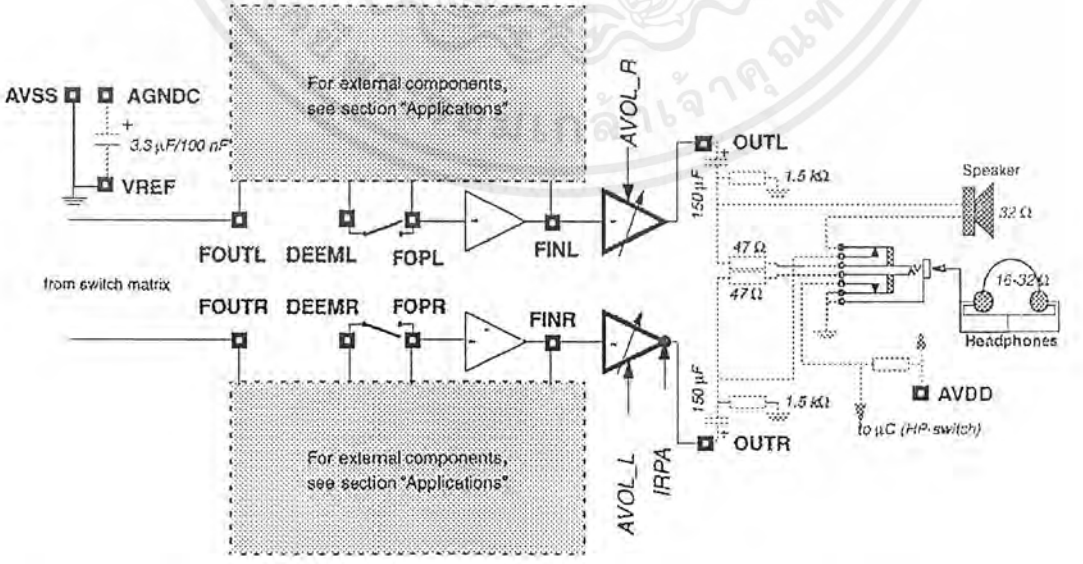
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 แสดงแมทริกซ์การสวิทช์

4.2.5 โพสท์ฟิลเตอร์ออปแอมป์ (**Postfilter Op-Amps**), และสายต่อออกภายนอก **(Line Out)**

บล็อกการทำงานของส่วนนี้แสดงดังรูปที่ 4.8 ในส่วนของโพสท์ฟิลเตอร์จะเป็นวงจรกรองความถี่ต่ำผ่านแบบแอคทีฟที่อาศัยออปแอมป์ภายใน โดยสามารถเลือกใช้วงจรฟิลเตอร์ลำดับต่างๆ ได้ตามต้องการ



รูปที่ 4.8 แสดงโพสท์ฟิวเตอร์ ออปแอมป์, ดีมฟาซิส ออปแอมป์ และสายออก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.6 การควบคุมระดับเสียง (Volume)

การควบคุมระดับเสียง จะสามารถกำหนดได้ครอบคลุมจาก +18 dB ถึง –75 dB โดยที่ระดับต่ำที่สุดจะเป็นการตัดเสียง (mute)

4.2.7 วงจรขยายสำหรับหูฟัง (Headphone Amplifier)

เอาท์พุทของวงจรขยายสำหรับหูฟังจะอยู่ที่ขา OUTL และ OUTR ซึ่งต่ออยู่กับหูฟังสเตอริโอหรือลำโพงเดี่ยว หูฟังสเตอริโอต้องต่อความต้านทาน 47 Ω แบบขนานภายนอกในทั้งสองช่องเสียง ถ้าต่อลำโพงเข้ากับเอาท์พุทเหล่านี้เครื่องขยายกำลังของช่องเสียงขวาต้องกลับขั้ว

ตารางที่ 4.1 การควบคุมระดับเสียง

Volume / dB	AVOL
18.0	111000
16.5	110111
15.0	110110
13.5	110101
-	-
0.0	101100 (dufault)
-1.5	101011
-	-
-54.0	001000
-57.0	000111
-	-
-75.0	000001
Mute	000000

4.2.8 ระบบสัญญาณนาฬิกา

ข้อดีของระบบสัญญาณนาฬิกาของ DAC 3550A ก็คือไม่ต้องการสัญญาณนาฬิกาหลักจากภายนอก DAC สัญญาณนาฬิกาภายในจะถูกสร้างโดย วงจร PLL ซึ่งจะล็อคเข้ากับความถี่จากสัญญาณนาฬิกาของ I^2S (CLI) เราสามารถกำหนดการทำงานของได้ 2 โหมด คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

– โหมดมาตรฐาน

– โหมดเอ็มเป็ก

การกำหนดโหมดการทำงานทำได้โดยการป้อนลอจิกที่ขา MCS1 และ MCS2

ตารางที่ 4.2 โหมดการทำงาน

MCS1	MCS2	Mode	Subaddress	Default Sample Rate
0	0	Standard	ADR0	32-48 kHz
0	1	Standard	ADR1	32-48 kHz
1	0	Standard	ADR2	32-48 kHz
1	1	MPEG	ADR3	Automatic

ในที่นี้จะขอกล่าวถึงเฉพาะโหมดเอ็มเป็ก

โหมดเอ็มเป็ก

โหมดนี้ควรจะใช้ร่วมกับ MAS 3507D ในการประยุกต์ใช้งานในเครื่องเล่น MPEG ในกรณีนี้ตัว MAS3507D ต้องการสัญญาณนาฬิกา 14.725 MHz ดังนั้นจึงต้องต่อคริสตัลความถี่ 14.725 MHz เพื่อที่จะให้สัญญาณนาฬิกาสำหรับ MAS 3507D และให้มีการตรวจจับอัตราการแซมเปิลอัตโนมัติในทุกอัตราการแซมเปิลจาก 8 ถึง 48 kHz ทำให้ไม่ต้องอาศัย I^2C บัสในการควบคุมอัตราแซมเปิล

อัตราการแซมเปิล MPEG :

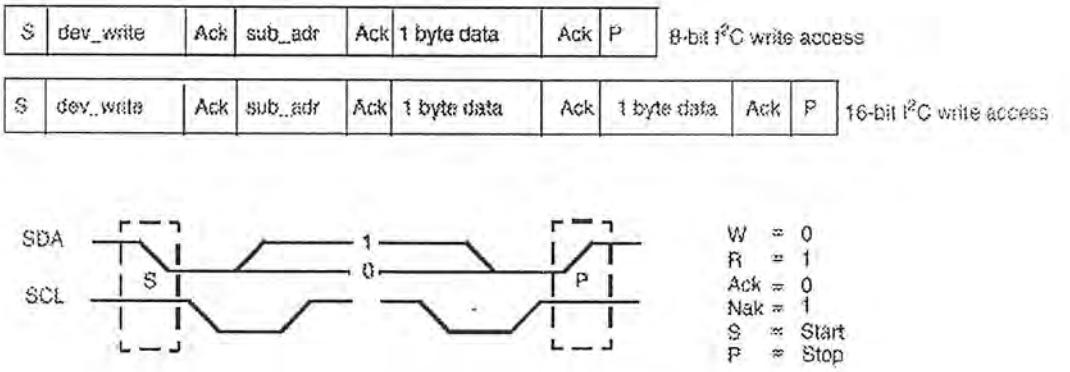
8 kHz, 11.025 kHz, 12 kHz, 16 kHz, 22.05 kHz, 24 kHz, 32 kHz, 44.1 kHz, 48 kHz

4.2.9 การเชื่อมต่อบัส I^2C

MAS 3550A จะถูกติดตั้งโดยการเชื่อมต่อ I^2C ซึ่งจะใช้การอ้างอิงตำแหน่งย่อย 1 ระดับ บัสแอดเดรส I^2C จะใช้เพื่อติดต่อกับ IC

ตำแหน่งย่อย จะยอมให้มีการเลือกชิปในการประยุกต์ใช้งานที่มี DAC หลายตัวและเลือกรีจิสเตอร์ภายใน รีจิสเตอร์เหล่านี้เขียนได้อย่างเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.9 แสดงโปรโตคอลของบัส I^2C ในการเขียน

4.2.10 รีจิสเตอร์

การรีเซททางฮาร์ดแวร์จะเซทค่าเริ่มต้นให้กับรีจิสเตอร์ควบคุมทุกตัวเป็นศูนย์ ในขณะที่เปิดเครื่องมาตัวชิปจะมีการโหลดค่าเริ่มต้นด้วยค่าดีฟอลท์ ดังตาราง 4.3

ตาราง 4.3 ตำแหน่งรีจิสเตอร์ I^2C

RA1	RA0	นีโมนิก
0	1	SR_REG
1	0	AVOL
1	1	GCFG

รีจิสเตอร์ของ DAC 3550A จะมีข้อมูลขนาด 8 หรือ 16 บิต การเข้าถึงรีจิสเตอร์ที่มี 16 บิต สามารถทำได้โดยการเขียน 2 เวิร์ดข้อมูล 8 บิต

นอกจากนี้ยังมีรายละเอียดของคำสั่งพิเศษ ซึ่งสามารถดูได้ที่ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
โปรโตคอลที่ใช้ในการเชื่อมต่อกับซีดี-รอม

ซีดี-รอมที่ผลิตขึ้นมาใช้กับเครื่องคอมพิวเตอร์ในปัจจุบันนั้นมีการเชื่อมต่อ เป็นไปตามมาตรฐาน ATAPI ซึ่งเป็นมาตรฐานที่เพิ่มเติมมาจากมาตรฐาน ATA ซึ่งเป็นมาตรฐานในการเชื่อมต่อฮาร์ดดิสก์เข้ากับเครื่องคอมพิวเตอร์ โดยการเชื่อมต่ออุปกรณ์ตามมาตรฐาน ATAPI จะเหมือนกับมาตรฐาน ATA ทั้งหมด จะมีส่วนที่แตกต่างก็คือการส่งคำสั่งเข้าไปยังตัวอุปกรณ์จะต้องส่งเป็นชุดคำสั่ง

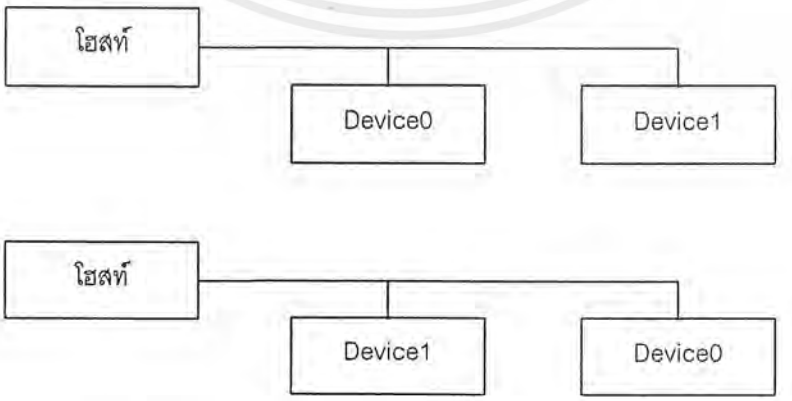
5.1 ATA

5.1.1 การเชื่อมต่อทางกายภาพ

ตามมาตรฐาน ATA สามารถนำอุปกรณ์มาต่อเชื่อมกันได้ 1-2 ตัว โดยอุปกรณ์นั้นๆ ต้องเชื่อมต่อกับส่วนที่ทำหน้าที่ควบคุมมัน หรือที่เรียกว่าโฮสท์อะแดปเตอร์ (host adapter) และถ้าอุปกรณ์ที่เชื่อมต่อมี 2 ตัว อุปกรณ์ทั้งสองจะต้องต่อเชื่อมกันแบบ daisy chain configuration ซึ่งอุปกรณ์ตัวหนึ่งจะถูกกำหนดเป็น Device0 และอีกตัวหนึ่งจะเป็น Device1

การกำหนดว่าอุปกรณ์ตัวใดจะเป็น Device0 หรือ Device1 นั้นอาจทำได้หลายวิธี

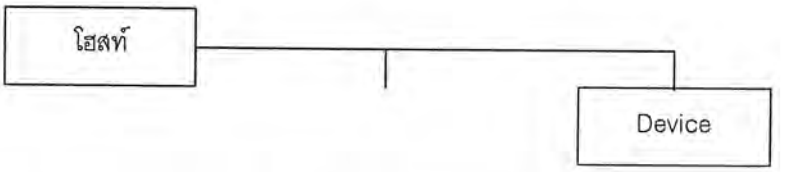
- สวิทช์หรือจัมเปอร์ (jumper) บนอุปกรณ์
- การใช้ขา Cable Select (CSEL)



รูปที่ 5.1 แสดงการเชื่อมต่อระหว่างโฮสท์กับอุปกรณ์ 2 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

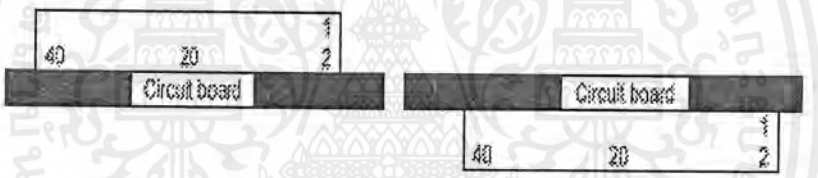
ในการเชื่อมต่ออุปกรณ์ 2 ตัวนั้นตำแหน่งในการต่อสายไฟจะไม่มีผลต่อการเชื่อมต่อ ถ้ามีอุปกรณ์ที่เชื่อมต่อแบบ ATA เพียงตัวเดียว แล้วโฮสท์กับอุปกรณ์ควรจะอยู่ที่ปลายทั้งสองด้านของสายไฟ



รูปที่ 5.2 แสดงการเชื่อมต่อระหว่างโฮสท์กับอุปกรณ์ตัวเดียว

5.1.2 ขาเชื่อมต่อสำหรับรับส่งข้อมูล

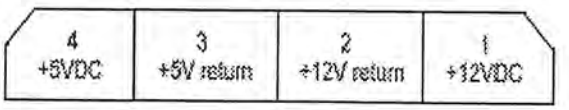
ตามมาตรฐาน ATA ขาเชื่อมต่อระหว่างอุปกรณ์และโฮสท์เป็นตัวเชื่อมต่อขนาด 40 ขา ซึ่งมีการจัดวางตำแหน่งตามรูปที่ 5.3



รูปที่ 5.3 ตัวเชื่อมต่อขนาด 40 ขา ตามมาตรฐาน ATA

จากจำนวนขาทั้งหมด 40 ขา เป็นส่วนของขาส่งข้อมูลขนาด 16 บิตอยู่ 16 ขา ส่วนที่ทำหน้าที่กำหนดตำแหน่งรีจิสเตอร์ที่จะเขียนอีก 5 ขา ส่วนที่ใช้ควบคุมการส่งข้อมูลในแบบ DMA 2 ขา ขารีเซท 1 ขา และส่วนที่เหลือเป็นขาสำหรับควบคุมการทำงานต่างๆ ของตัวอุปกรณ์ เช่น สัญญาณอ่าน-เขียน สัญญาณอินเทอร์รัพ เป็นต้น

ในส่วนของไฟเลี้ยงอุปกรณ์ที่เป็นไปตามมาตรฐาน ATA นั้นจะเป็นตัวเชื่อมต่อขนาด 4 ขา ดังรูปที่ 5.4 โดยจะมีไฟเลี้ยงสองชุด ชุดหนึ่งจ่ายไฟตรง 12 V อีกชุดหนึ่งจ่ายไฟตรง 5 V ซึ่งสามารถดูรายละเอียดเกี่ยวกับขาต่างๆ ได้ที่ภาคผนวก



รูปที่ 5.4 ตัวเชื่อมต่อจ่ายไฟเลี้ยงให้อุปกรณ์ตามมาตรฐาน ATA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1.3 รีจิสเตอร์ภายใน

ตามมาตรฐาน ATA นั้น การสั่งงานเพื่อควบคุมการทำงานของอุปกรณ์จะต้องกระทำผ่านรีจีสเตอร์ต่างๆ ภายในตัวอุปกรณ์นั้นๆ โดยการกำหนดรีจีสเตอร์ที่จะใช้งานนั้นจะกำหนดผ่านสายสัญญาณ 5 เส้น คือ CS0-1 และ DA0-2 นอกจากจะกำหนดรีจิสเตอร์ผ่านสายสัญญาณทั้ง 5 แล้ว การเขียนหรืออ่านรีจีสเตอร์ในตำแหน่งเดียวกันนั้นยังเป็นการอ้างถึงรีจีสเตอร์คนละตัวกันด้วย

ตารางที่ 5.1 แสดงแอดเดรสของรีจิสเตอร์ตามมาตรฐาน ATA

แอดเดรส					หน้า ที่	
CS0-	CS1-	DA2	DA1	DA0	READ (DIOR-)	WRITE (DIOW-)
N	N	x	x	x	บัสข้อมูลอยู่ในสถานะอิมพิแดนซ์สูง	ไม่ใช้
รีจิสเตอร์ Control Block						
N	A	0	x	x	บัสข้อมูลอยู่ในสถานะอิมพิแดนซ์สูง	ไม่ใช้
N	A	1	0	x	บัสข้อมูลอยู่ในสถานะอิมพิแดนซ์สูง	ไม่ใช้
N	A	1	1	0	Alternate Status	Device Control
N	A	1	1	1	ไม่ใช้	ไม่ใช้
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
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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รีจีสเตอร์ทั้งหมดสามารถแบ่งออกได้เป็นสองส่วนใหญ่ๆ คือ

- รีจิสเตอร์ Command Block จะถูกใช้เพื่อส่งคำสั่งไปยังอุปกรณ์หรือส่งสถานะของอุปกรณ์กลับมายังโฮสท์

- รีจิสเตอร์ Control Block จะถูกใช้เพื่อให้อุปกรณ์เข้าควบคุม และส่งค่าสถานะสำรอง (alternate status)

ตำแหน่งของรีจีสเตอร์ต่างๆ ตามมาตรฐาน ATA เป็นไปตามตาราง 5.1
รีจีสเตอร์แต่ละตัวมีขนาด 16 บิต และมีหน้าที่การทำงานที่ต่างกันไป ต่อไปจะขอกล่าวถึงหน้าที่การทำงานของรีจีสเตอร์บางตัวที่สำคัญ

- **รีจีสเตอร์ข้อมูล (Data Register)** เป็นตำแหน่งของรีจีสเตอร์ที่ทำหน้าที่ส่งข้อมูลจากตัวอุปกรณ์ออกไปยังโฮสท์ และรับข้อมูลจากโฮสต์เข้ามา
- **รีจีสเตอร์ Error** เก็บสถานะของการทำงานคำสั่งล่าสุด ใช้สำหรับตรวจสอบความผิดพลาดในการทำงาน
- **รีจีสเตอร์ Device/Head** ใช้กำหนดตัวอุปกรณ์ที่จะใช้งาน เนื่องจากการต่ออุปกรณ์ตามมาตรฐาน ATA สามารถต่อได้ 2 ตัว จึงต้องมีการเลือกตัวอุปกรณ์ที่ใช้งานด้วย
- **รีจีสเตอร์สถานะ (Status Register)** เก็บสถานะปัจจุบันของตัวอุปกรณ์
- **รีจีสเตอร์คำสั่ง (Command Register)** เป็นตำแหน่งรีจีสเตอร์ที่ใช้เขียนคำสั่งเข้ามาเพื่อควบคุมการทำงานของอุปกรณ์

รีจีสเตอร์ที่กล่าวมาข้างต้นนั้น บางตัวจะมีการกำหนดหน้าที่การทำงานเป็นบิตๆ โดยแต่ละบิตแยกกันอย่างอิสระ สามารถดูรายละเอียดได้ที่ภาคผนวก

5.1.4 โปรโตคอลการส่งข้อมูล

การควบคุมอุปกรณ์ตามมาตรฐาน ATA นั้น ต้องส่งข้อมูลเข้าไปยังรีจีสเตอร์ต่างๆ ของตัวอุปกรณ์นั้นๆ โดยการส่งข้อมูลเข้าไปยังรีจีสเตอร์ภายในนั้นจำเป็นจะต้องมีลำดับในการเขียนรีจีสเตอร์ต่างๆ อย่างถูกต้องจึงจะสามารถส่งคำสั่งได้อย่างถูกต้อง

โปรโตคอลในการส่งคำสั่งเพื่อควบคุมการทำงานของอุปกรณ์สามารถแบ่งออกได้เป็นประเภทต่างๆ ตามข้อมูลที่อ่านออกมาจากตัวอุปกรณ์ ซึ่งสามารถแบ่งได้เป็น 3 ประเภท

- คำสั่งที่ไม่มีการส่งข้อมูลกลับจากตัวอุปกรณ์
- คำสั่งที่มีการส่งข้อมูลกลับในแบบ PIO
- คำสั่งที่มีการส่งข้อมูลกลับในแบบ DMA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรโตคอลทั้งสามจะมีส่วนที่คล้ายกันในและต่างกันในบางส่วน ซึ่งจะขอกล่าวแยกเป็นชนิดๆ ดังนี้

โปรโตคอลการส่งคำสั่งที่ไม่มีการส่งข้อมูลกลับ

ลำดับการส่งคำสั่งที่ไม่มีการส่งข้อมูลจากตัวอุปกรณ์กลับมายังโฮสท์เป็นไปดังนี้

1. โฮสต์ต้องเช็คค่าบิต BSY ภายในรีจีสเตอร์สถานะ (Status Register) จนกว่าบิต BSY จะเท่ากับ 0
2. เขียนรีจีสเตอร์ Device/Head เพื่อเลือกอุปกรณ์ที่ต้องการควบคุม
3. ตรวจสอบค่าภายในรีจีสเตอร์สถานะ (Status Register) โดยรอจนกว่าบิต BSY จะเท่ากับ 0 และบิต DRDY จะเท่ากับ 1
4. เขียนพารามิเตอร์ต่างๆ ที่แต่ละคำสั่งต้องการลงในรีจีสเตอร์ต่างๆ ตามความต้องการของแต่ละคำสั่ง
5. เขียนคำสั่งที่ต้องการลงในรีจีสเตอร์คำสั่ง (Command Register)
6. เมื่ออุปกรณ์ได้รับการเขียนรีจีสเตอร์คำสั่ง ตัวอุปกรณ์จะเซ็ตบิต BSY เป็น 1 และปฏิบัติตามคำสั่งที่สั่ง
7. เมื่อทำงานตามคำสั่งนั้นๆ เสร็จสิ้น อุปกรณ์จะเคลียร์ค่าบิต BSY และส่งสัญญาณอินเทอร์รัพท์กลับมายังโฮสท์
8. ในกรณีที่เกิดความผิดพลาดในการส่งข้อมูลจะรายงานความผิดพลาดในรีจีสเตอร์ Error

โปรโตคอลการส่งคำสั่งที่มีการส่งข้อมูลกลับในแบบ PIO

การส่งคำสั่งที่มีการส่งข้อมูลกลับในแบบ PIO นี้จะมีส่วนที่เหมือนกับการส่งข้อมูลที่ไม่มีการส่งข้อมูลกลับตั้งแต่ลำดับที่ 1 ถึงลำดับที่ 5

1. โฮสต์ต้องเช็คค่าบิต BSY ภายในรีจีสเตอร์สถานะ (Status Register) จนกว่าบิต BSY จะเท่ากับ 0
2. เขียนรีจีสเตอร์ Device/Head เพื่อเลือกอุปกรณ์ที่ต้องการควบคุม
3. ตรวจสอบค่าภายในรีจีสเตอร์สถานะ (Status Register) โดยรอจนกว่าบิต BSY จะเท่ากับ 0 และบิต DRDY จะเท่ากับ 1
4. เขียนพารามิเตอร์ต่างๆ ที่แต่ละคำสั่งต้องการลงในรีจีสเตอร์ต่างๆ ตามความต้องการของแต่ละคำสั่ง

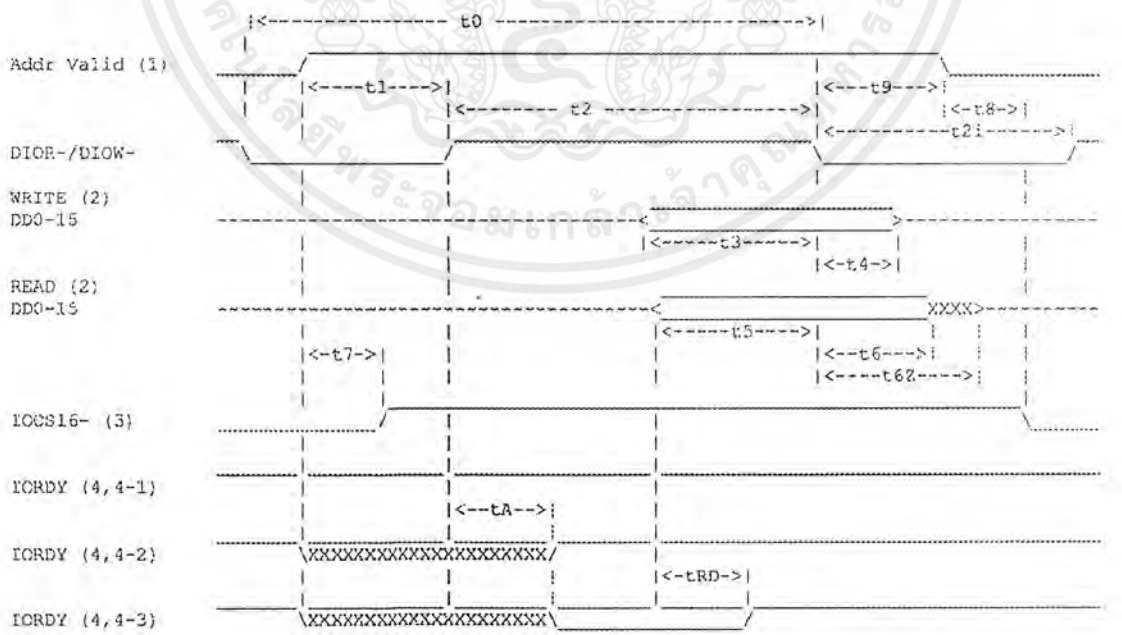
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. เขียนคำสั่งที่ต้องการลงในรีจีสเตอร์คำสั่ง (Command Register)
6. อุปกรณ์จะเซ็ตบิต BSY และปฏิบัติตามคำสั่งที่ส่งเข้าไป และเตรียมส่งข้อมูลชุดแรกไปยังโฮสท์
7. เมื่อพร้อมที่จะส่งข้อมูลออกมา อุปกรณ์จะเซ็ตบิต DRQ เคลียร์บิต BSY และส่งสัญญาณอินเทอร์รัพท์มายังโฮสท์
8. โฮสท์อ่านข้อมูลจากตัวอุปกรณ์
9. เช็คบิต DRQ ภายในรีจีสเตอร์สถานะ ถ้าบิต DRQ เท่ากับ 1 หมายถึงยังมีข้อมูลเวิร์ดถัดไปที่จะส่งออกมายังโฮสท์ แต่ถ้าหมดข้อมูลที่จะส่งแล้ว บิต DRQ จะเท่ากับ 0 และบิต BSY เท่ากับ 0

โปรโตคอลการส่งคำสั่งที่มีการส่งข้อมูลกลับในแบบ DMA

การส่งคำสั่งที่มีการส่งข้อมูลกลับในแบบ DMA นี้จะคล้ายกับการส่งข้อมูลที่ส่งข้อมูลที่ไม่มีการส่งข้อมูลกลับตั้งแต่ลำดับที่ 1 ถึง 5 เช่นเดียวกับการส่งข้อมูลกลับแบบ PIO แต่หลังจากนั้นการส่งข้อมูลมูลจากตัวโฮสท์จะเป็นไปตามมาตรฐานการส่งข้อมูลแบบ DMA ซึ่งในชิ้นงานนี้ไม่ได้ใช้โปรโตคอลนี้จึงไม่ขอกล่าวโดยละเอียด

5.1.5 ลำดับเวลาในการส่งสัญญาณ (Timing Diagram)



รูปที่ 5.5 แสดงลำดับเวลาในการส่งสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังที่กล่าวไปแล้วในหัวข้อที่แล้วว่าในการส่งคำสั่งแต่ละคำสั่งนั้นจำเป็นต้องมีลำดับในการเขียนรีจีสเตอร์ที่ถูกต้อง แต่ในการเขียนรีจีสเตอร์แต่ละครั้งก็จำเป็นจะต้องมีลำดับเวลาในการส่งสัญญาณเพื่อเขียนหรืออ่านรีจีสเตอร์อย่างถูกต้องด้วย จึงจะสามารถเขียนหรืออ่านข้อมูลได้อย่างถูกต้อง ลำดับเวลาในการส่งสัญญาณเป็นไปดังรูป 5.5

5.2 ATAPI

มาตรฐาน ATAPI เป็นมาตรฐานที่อยู่บนมาตรฐาน ATA อีกทีหนึ่ง จากเดิมที่การควบคุมอุปกรณ์ในมาตรฐาน ATA ใช้คำสั่งเพียงเวิร์ดเดียวในการสั่งงาน ในมาตรฐาน ATAPI นี้จะเปลี่ยนรูปแบบการส่งคำสั่งจากเวิร์ดเดียวเป็นแพ็คเก็ตของคำสั่งแทน เพื่อความยืดหยุ่นในการสั่งงานอุปกรณ์ แต่ถึงอย่างอย่างไรก็ตามมาตรฐาน ATAPI เป็นมาตรฐานที่อยู่บนมาตรฐาน ATA ดังนั้นการเชื่อมต่อขาเชื่อมต่อ (Connector) และลำดับการส่งสัญญาณ (Timing Diagram) จะเหมือนกับมาตรฐาน ATA ทั้งหมด ส่วนที่แตกต่างไปบ้างคือ รีจีสเตอร์ภายในอุปกรณ์ที่มีการเปลี่ยนแปลงหน้าที่การทำงานไปบางส่วน และส่วนที่แตกต่างกันกับมาตรฐาน ATA อย่างชัดเจนก็คือโปรโตคอลที่ใช้ส่งคำสั่งควบคุมอุปกรณ์

5.2.1 รีจิสเตอร์ภายใน

ในมาตรฐาน ATAPI รีจีสเตอร์ภายในส่วนใหญ่จะยังคงทำหน้าที่เดิมอยู่ แต่จะมีรีจีสเตอร์อยู่ 2 ตัวที่มีการเปลี่ยนแปลงหน้าที่การทำงาน และบางตัวจะยังคงทำงานเหมือนเดิม คือ

- รีจีสเตอร์ Cylinder High/Low เปลี่ยนเป็นรีจีสเตอร์ Byte Count ซึ่งทำหน้าที่ควบคุมปริมาณข้อมูลที่จะส่งจากตัวอุปกรณ์ไปยังโฮสท์หรือจากโฮสท์มายังตัวอุปกรณ์ ดูการทำงานของรีจีสเตอร์นี้ได้ในหัวข้อ 5.2.4
- รีจีสเตอร์ Sector Count เปลี่ยนเป็นรีจีสเตอร์ Interrupt Reason ซึ่งทำหน้าที่บอกสาเหตุของการอินเทอร์รัพท์แต่ละครั้ง ดูการทำงานของรีจีสเตอร์นี้ได้ในหัวข้อ 5.2.5

นอกจากหน้าที่ของรีจิสเตอร์ภายในบางตัวจะมีการเปลี่ยนแปลงแล้ว หน้าที่การทำงานของบิตบางบิตภายในตัวรีจีสเตอร์ก็มีการเปลี่ยนแปลงเช่นเดียวกัน

5.2.2 โปรโตคอลการส่งข้อมูล

การสั่งการอุปกรณ์ ATAPI กระทำได้ 2 วิธี คือ วิธีทาสก์ไฟล์ (Task File) และวิธีใหม่ที่เรียกว่า แพ็คเกตคอมมานด์ (Packet Command) สำหรับทั้งสองวิธีอุปกรณ์จะถูกสั่งการโดย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คอมพิวเตอร์โฮสท์ (Host) เพื่อที่จะแสดงคำสั่งและส่งค่าสถานะกลับมายังโฮสท์เมื่อเสร็จสิ้นคำสั่ง หากมีการเชื่อมต่อกับอุปกรณ์ 2 ตัวแล้ว คำสั่งจะต้องถูกเขียนในรูปแบบขนานไปยังอุปกรณ์ที่เชื่อมต่อทั้งหมด และสำหรับคำสั่งของ ATA แล้วอุปกรณ์ที่ถูกเลือก (ดูจากบิต DRV ในรีจิสเตอร์ Drive/Head) เท่านั้นที่จะปฏิบัติตามคำสั่ง

โปรโตคอลสำหรับ ATAPI นั้นเน้นที่การใช้คำสั่ง ATA แบบใหม่ที่เรียกว่า แพ็คเกตคอมมานด์ กติกาและโปรโตคอลของ ATA ทั่วไปจะใช้กับแพ็คเกตคอมมานด์ แต่ในการส่งคำสั่งออกไปแต่ละครั้งจะต้องประยุกต์ใช้หลักเกณฑ์ใหม่ ๆ ด้วย คือ

1. บิต DRQ ในรีจิสเตอร์สถานะจะต้องใช้ร่วมกับรีจิสเตอร์ Interrupt Reason เพื่อที่จะระบุชนิดของการอินเตอร์รัพที่เกิดขึ้นจริง
2. คำสั่งที่อุปกรณ์จะปฏิบัตินั้น จะส่งเป็นแพ็คเกต (packet) ผ่านรีจิสเตอร์ข้อมูล ไม่ใช่ทาสก์ไฟล์
3. ใช้ Byte Count ในการกำหนดปริมาณข้อมูลที่โฮสท์จะถ่ายโอนในแต่ละ DRQ Interrupt
4. รีจิสเตอร์ Features ของ ATAPI ใช้เพื่อบอกอุปกรณ์ให้ใช้การถ่ายโอนข้อมูลแบบ DMA
5. สถานะสุดท้ายจะส่งไปที่โฮสท์เสมือนการอินเตอร์รัพ หลังการถ่ายโอนข้อมูลสุดท้าย แทนที่จะส่งไปกับบล็อคข้อมูลสุดท้าย

โปรโตคอลของ ATAPI จะถูกประยุกต์ใช้เมื่อมีการส่งแพ็คเกตคอมมานด์จนกระทั่งโฮสท์อ่าน Completion Status หลังจากอ่าน Completion Status แล้วนิยามของรีจิสเตอร์ทาสก์ไฟล์และโปรโตคอลจะกลับไปใช้นิยามของ ATA มาตรฐาน

5.2.3 แพ็คเกตคอมมานด์ของ ATAPI

การส่งแพ็คเกตคอมมานด์ของ ATAPI ทำได้โดยเซทบิต DRV, เขียนรหัสคำสั่งมายังรีจิสเตอร์คำสั่งเพื่อบอกอุปกรณ์ให้ใช้โปรโตคอล ATAPI สำหรับคำสั่ง ATA ทั่วไป สัญญาณ DRQ จะใช้เพื่อแสดงว่าข้อมูลสำหรับคำสั่งนั้น ๆ สามารถถูกถ่ายโอนมาจาก/ไปยังอุปกรณ์ได้ ส่วนถ้าเป็นแพ็คเกตคอมมานด์ สัญญาณ DRQ ในครั้งแรกจะแสดงว่ามีการเขียนข้อมูลแพ็คเกตคอมมานด์มายังอุปกรณ์แล้ว จากนั้นคำสั่งก็จะถูกดำเนินการเหมือนคำสั่ง ATA ทั่วไป คอมมานด์แพ็คเกตจะถูกถ่ายโอนผ่าน PIO เสมอ และไม่ใช้ DMA

แพ็คเกตคอมมานด์ของ ATAPI สามารถส่งออกมาโดยไม่ต้องคำนึงถึงสถานะของบิตสถานะ DRDY

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าในช่วงการพูลลิ่งสัญญาณ BSY อุปกรณ์ยังคงอยู่ในสภาวะที่ไม่สามารถรับคำสั่งได้นานมากกว่า 5 วินาทีแล้วโฮสท์จะต้องไทม์เอาท์ (time out) และรีเซทอุปกรณ์

การถ่ายโอนข้อมูลนั้นอาจทำได้หลายวิธีมากกว่าที่ได้อธิบายไว้ แต่จะต้องใช้ตามลำดับดังที่อธิบายตามมาตรฐานนี้เพื่อให้สามารถเข้ากันได้ (compatible) กับอุปกรณ์ ATAPI ในปัจจุบันและในอนาคต

5.2.4 การใช้รีจิสเตอร์ Byte Count (Cylinder Low/High) สำหรับแพ็คเกตคอมมานด์

รีจิสเตอร์นี้ใช้ในการควบคุมจำนวนไบต์ที่โฮสท์จะถ่ายโอนในแต่ละ DRQ ใช้สำหรับข้อมูลพารามิเตอร์ของคำสั่ง (command parameter data) ที่จะถูกถ่ายโอนผ่าน PIO และไม่ใช้สำหรับ DMA หรือจำนวนข้อมูลในคอมมานด์แพ็คเกต

เนื่องจากความยาวของข้อมูลที่จะถ่ายโอนจริงมายัง/จากอุปกรณ์ซึ่งใช้ PIO นั้นจะถูกควบคุมโดยโฮสท์ และเนื่องจากอุปกรณ์ ATAPI ต้องการที่จะควบคุมจำนวนไบต์ที่จะถ่ายโอน ด้วยเหตุนี้จึงมีการเพิ่มความสามารถดังกล่าวเข้าไปโดยใช้รีจิสเตอร์ Byte Count

ตารางที่ 5.2 การใช้รีจิสเตอร์ Byte Count

การทำงาน	PIO	DMA
ส่งแพ็คเกตคอมมานด์	ใช้เพื่อเป็นพารามิเตอร์ให้กับแพ็คเกตคอมมานด์ และไม่ใช้ควบคุมการถ่ายโอนแพ็คเกตคอมมานด์	คอมมานด์แพ็คเกตจะส่งผ่านโหมด PIO เท่านั้น
เป็นพารามิเตอร์ให้แก่แพ็คเกตคอมมานด์	การเป็นพารามิเตอร์ให้แก่แพ็คเกตคอมมานด์ทั่วไปที่จะถ่ายโอนข้อมูลพารามิเตอร์ โฮสท์จะใช้ Byte Count ในการบอกปริมาณข้อมูลสูงสุดที่จะถูกถ่ายโอนในแต่ละ DRQ	อุปกรณ์สามารถที่จะเพิกเฉยต่อค่าใน Byte Count ในกรณีที่การถ่ายโอนที่เกิดขึ้นจริงนั้นถูกควบคุมโดยอุปกรณ์ ATAPI ไม่ใช่โฮสท์
เป็นข้อมูลพารามิเตอร์ที่ส่งจากอุปกรณ์ไปยังโฮสท์	ในแต่ละ DRQ/DMARQ Byte Count จะบรรจุจำนวนไบท์ของข้อมูลที่โฮสท์จะถ่ายโอนมาจากอุปกรณ์	อุปกรณ์ ATAPI สามารถที่จะถ่ายโอนข้อมูลเมื่อใดก็ได้ตามต้องการ
เป็นข้อมูลพารามิเตอร์ที่ส่งจากอุปกรณ์มายังโฮสท์	ในแต่ละ DRQ/DMARQ Byte Count จะบรรจุจำนวนไบท์ของข้อมูลที่โฮสท์จะถ่ายโอนไปยังอุปกรณ์	อุปกรณ์ ATAPI สามารถที่จะถ่ายโอนข้อมูลเมื่อใดก็ได้ตามต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก่อนที่จะเกิดการถ่ายโอนข้อมูล โฮสท์จะอ่าน 16 บิตของรีจิสเตอร์ Byte Count และจะถ่ายโอนตามความยาวนั้น ๆ ทั้งอุปกรณ์ ATAPI และโฮสท์ ต้องมีการนับไบต์ของตัวเองและจะถ่ายโอนจนกระทั่งการนับไบต์เป็นศูนย์ สำหรับบางคำสั่ง เช่น Mode Sense โฮสท์จะไม่รู้ถึงปริมาณข้อมูลที่จะถูกถ่ายโอน ดังนั้นโฮสท์จำเป็นต้องอ่านค่า Byte Count ของอุปกรณ์เพื่อที่จะถ่ายโอนข้อมูลด้วยจำนวนที่ถูกต้อง

นอกจากนี้รีจิสเตอร์ Byte Count ยังใช้ในการกำหนดปริมาณข้อมูลสูงสุดที่โฮสท์สามารถรับได้ในหนึ่งแพ็คเกต (single PIO DRQ Packet) หรือบอกขนาดของแพ็คเกต อุปกรณ์ ATAPI จะใช้ค่านี้เป็นขนาดสูงสุดสำหรับแต่ละ PIO หรือชุดข้อมูลของ DMA อุปกรณ์สามารถเลือกที่จะถ่ายโอนแพ็คเกตที่เล็กกว่าค่าที่กำหนดโดยโฮสท์ในรีจิสเตอร์ Byte Count ได้

5.2.5 การใช้รีจิสเตอร์ Sector Count สำหรับแพ็คเกตคอมมานด์

รีจิสเตอร์ Interrupt Reason จะบรรจุรายละเอียดของสัญญาณ DRQ เมื่อบิต DRQ ในรีจิสเตอร์สถานะถูกเซทเนื่องจากแพ็คเกตคอมมานด์แล้ว รีจิสเตอร์นี้จะแสดงว่าจะต้องถ่ายโอนแพ็คเกตคอมมานด์หรือข้อมูลของผู้ใช้ และใช้แสดงทิศทางของการถ่ายโอน

5.2.6 ลำดับการส่งชุดคำสั่งที่มีการส่งข้อมูลกลับในแบบ PIO

หัวข้อนี้รวมถึงบางคำสั่ง ได้แก่ Inquiry, Read เป็นต้น การปฏิบัติการนั้นรวมถึงการถ่ายโอนไบต์ข้อมูลที่ไม่ทราบจำนวนจากอุปกรณ์มายังโฮสท์

1. โฮสท์ทำการพูลลิ่งจน BSY=0 , DRQ=0 แล้วจะใส่ค่าเริ่ม (initialize) ลงในทาสก์ไฟล์โดยการเขียนพารามิเตอร์ที่ต้องการลงในรีจิสเตอร์ Features, Byte Count และ Driye/Head
2. โฮสท์เขียนรหัสแพ็คเกตคอมมานด์ (A0h) ไปยังรีจิสเตอร์คำสั่ง
3. อุปกรณ์เซท BSY และเตรียมการถ่ายโอนคอมมานด์แพ็คเกต
4. เมื่ออุปกรณ์พร้อมที่จะรับคอมมานด์แพ็คเกต อุปกรณ์ จะทำการเซท CoD และเคลียร์ IO ยืนยันสัญญาณ DRQ โดยทันทีหรือก่อนที่จะยืนยันสัญญาณ BSY อีกครั้ง บางอุปกรณ์จะยืนยันสัญญาณ INTRQ ต่อจากการยืนยัน DRQ หลังจากตรวจพบ DRQ โฮสท์จะเขียน 12 ไบต์ (6 เวิร์ด) ของคำสั่งไปยังรีจิสเตอร์ข้อมูล
5. อุปกรณ์จะ (1) เคลียร์ DRQ (เมื่อมีการเขียนไบต์ที่ 12)
   (2) เซท BSY
   (3) อ่านรีจิสเตอร์ Features และ Byte Count

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(4) เตรียมการถ่ายโอนข้อมูล

6. เมื่ออุปกรณ์เตรียมข้อมูลพร้อมแล้ว
   อุปกรณ์จะ (1) ใส่ Byte Count ของข้อมูลลงในรีจิสเตอร์ Cylinder High และ Low
   (2) เซท IO และเคลียร์ CoD
   (3) เซท DRQ และเคลียร์ BSY
   (4) เซท INTRQ
7. หลังจากตรวจพบ INTRQ โฮสท์จะอ่านบิต DRQ ในรีจิสเตอร์สถานะเพื่อที่จะตัดสินใจว่าจะต้องดำเนินการต่อไปอย่างไรกับคำสั่ง ถ้า DRQ=0 แล้ว อุปกรณ์จะจบคำสั่ง ถ้า DRQ=1 แล้วโฮสท์จะอ่านข้อมูล (เป็นจำนวนตามที่ระบุในรีจิสเตอร์ Cylinder High/Low) ผ่านรีจิสเตอร์ข้อมูล
8. อุปกรณ์ เคลียร์ DRQ ถ้าต้องการถ่ายโอนข้อมูลอีก อุปกรณ์จะเซท BSY และต่อจากนี้สำดับจะซ้ำตั้งแต่ขั้นตอน 7
9. เมื่ออุปกรณ์พร้อมที่จะแสดงสถานะ อุปกรณ์จะใส่ completion status ลงในรีจิสเตอร์สถานะ, เซท CoD, IO, DRDY และก่อนที่จะยืนยัน INTRQ จะเคลียร์ BSY และ DRQ
10. หลังจากตรวจพบ INTRQ&DRQ=0 โฮสท์จะอ่านรีจิสเตอร์สถานะ

5.2.7 ลำดับการส่งชุดคำสั่งที่ไม่มีการส่งค่ากลับ

หัวข้อนี้รวมถึงบางคำสั่ง ได้แก่ Seek เป็นต้น การปฏิบัติการของคำสั่งเหล่านี้เกี่ยวกับการถ่ายโอนไบต์ข้อมูลที่ไม่มีข้อมูล (no data transfer)

1. โฮสท์ทำการพูลลิ่งจน BSY=0 , DRQ=0 แล้วจะใส่ค่าเริ่ม (initialize) ลงในทาสก์ไฟล์ โดยการเขียนพารามิเตอร์ที่ต้องการลงในรีจิสเตอร์ Features, Byte Count และ Drive/Head
2. โฮสท์เขียนรหัสแพ็คเกตคอมมานด์ (A0h) ไปยังรีจิสเตอร์คำสั่ง
3. อุปกรณ์เซท BSY และเตรียมการถ่ายโอนคอมมานด์แพ็คเกต
4. เมื่ออุปกรณ์พร้อมที่จะรับคอมมานด์แพ็คเกต อุปกรณ์จะทำการเซท CoD และเคลียร์ IO ยืนยันสัญญาณ DRQ โดยทันทีหรือก่อนที่จะยืนยันสัญญาณ BSY อีกครั้ง บางอุปกรณ์จะยืนยันสัญญาณ INTRQ ต่อจากการยืนยัน DRQ หลังจากตรวจพบ DRQ โฮสท์จะเขียน 12 ไบต์ (6 เวิร์ด) ของคำสั่งไปยังรีจิสเตอร์ข้อมูล
5. อุปกรณ์เซท BSY และปฏิบัติคำสั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. เมื่ออุปกรณ์พร้อมที่จะแสดงสถานะ อุปกรณ์จะใส่ completion status ลงในรีจิสเตอร์สถานะ, เซท CoD, IO, DRDY และเคลียร์ BSY และ DRQ ก่อนที่จะยืนยัน INTRQ
7. หลังจากตรวจพบ INTRQ โฮสท์จะอ่านรีจิสเตอร์สถานะเพื่อดูสถานะ command completion

- **หมายเหตุ** หารายละเอียดเพิ่มเติมได้จาก Internet โดยใช้ Keyword ATA หรือ ATAPI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

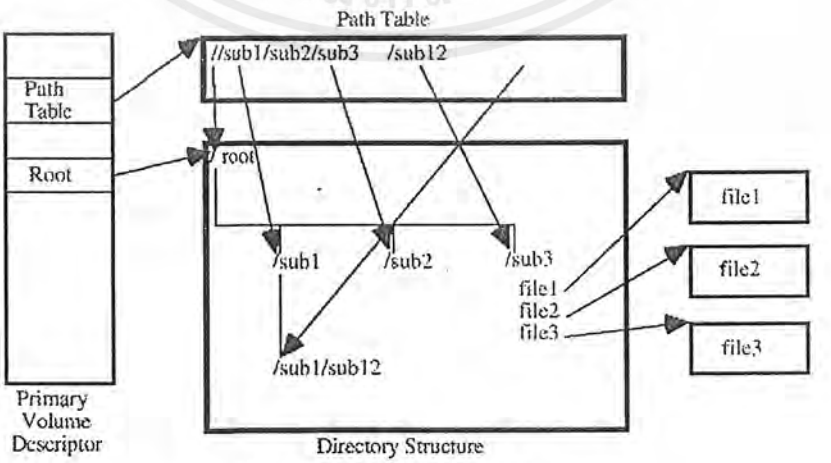
ISO 9660

6.1 บทนำ

คอมแพ็คดิสก์ (CD) ถูกเรียกได้ว่าเป็นผลิตภัณฑ์ที่ประสบความสำเร็จในตลาดผู้บริโภคอย่างสูงที่สุดเท่าที่เคยวางตลาดมา ตั้งแต่การแนะนำสินค้าชนิดนี้ในเดือนมิถุนายน 1980 ทำให้ CD มีอิทธิพลต่ออุตสาหกรรมดนตรีและเป็นทางเลือกสำหรับผู้ฟังดนตรีหลายล้านคน เนื่องจากความถูกต้องสูงเป็นพิเศษซึ่งเกิดจากเทคนิคการบันทึกแบบดิจิตอล และการแก้ไขข้อมูลนั้นสามารถทำได้ยากมากซึ่งเกิดจากการบันทึกแบบออบติคอล (optical) คุณสมบัติเหล่านี้ทำให้ CD มีเสน่ห์น่าสนใจในการเป็นตัวกลางของข้อมูลดิจิตอล ลักษณะเด่นอื่นที่ทำให้ CD น่าสนใจ ก็คือการที่สามารถผลิตออกมาในปริมาณมากได้อย่างรวดเร็วและราคาถูก

แต่ในระยะแรกของการเกิดนั้น ยังไม่มีมาตรฐานใดๆ เข้ามารองรับรูปแบบของการเขียนข้อมูลในตัวแผ่น ทำให้เกิดปัญหาในด้านความเข้ากันได้ของแผ่นซีดี ซึ่งเป็นปัญหาอย่างมากในด้านการพัฒนา จึงมีการจับมือกันระหว่างผู้ผลิตซีดีรอมหลายรายในการสร้างมาตรฐานร่วมกันให้เป็นมาตรฐานอุตสาหกรรมที่นิยามโดย Red Book, Yellow Book และ ISO 9660 ทำให้ในปัจจุบันสามารถใช้ CD กับฮาร์ดแวร์และซอฟท์แวร์ได้เกือบจะทุกรูปแบบ

6.2 ภาพรวมโครงสร้างของ ISO 9660



รูปที่ 6.1 โครงสร้าง ISO 9660

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงสร้างทางข้อมูลของ ISO 9660 แบ่งได้ 3 ส่วนหลัก คือ Volume Descriptor, Directory Structure และ Path Table โครงสร้างเหล่านี้เกี่ยวข้องซึ่งกันและกันดังแสดงในรูปที่ 6.1 Volume Descriptor จะบอกตำแหน่งของ Directory Structure และ Path Table, ไดเรคทอรีจะบอกตำแหน่งของไฟล์ และ Path Table จะเป็นวิธีลัดไปสู่แต่ละไดเรคทอรี

6.2.1 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) ก่อนที่ผู้ใช้จะสามารถเข้าถึง Volume ได้ แม้ว่า ISO 9660 จะไม่กำหนดว่าข้อมูลอะไรบ้างที่ต้องอยู่ใน Boot Record หรือวิธีที่จะใช้ข้อมูลเหล่านั้น Supplementary Volume Descriptor สามารถใช้ระบุ alternate character set สำหรับระบบที่ไม่สนับสนุน ISO646 character set ส่วน Volume Partition Descriptor สามารถใช้แบ่ง Volume ออกเป็น Volume ให้มีขนาดเล็กลง

Volume Descriptor จะเริ่มการอัดที่ Logical Sector16 (ซึ่งตอบสนองภายใน 2 วินาทีและ 16 เซคเตอร์ใน CD หรือใน CD "Atime",00:02:16)

Primary Volume Descriptor

Standard Identifier (CD001)
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
Time Stamps

รูปที่ 6.2 Primary Volume Descriptor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Primary Volume Descriptor เป็นจุดเริ่มต้นในการระบุ CD-ROM ประกอบด้วย Starndard 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

Standard Identifier เป็นกลุ่มของตัวอักษร และมีค่า CD001 ตามมาตรฐาน ISO 9660 ซึ่งจะบอกระบบปฏิบัติการว่านี่คือดิสก์ตาม ISO 9660 เพื่อที่จะแบ่งแยก volume ออกจากระบบไฟล์อื่นโดยใช้แบบแผนง่าย ๆ เช่น High Sierra ซึ่งมี Standard Identifier คือ CD-ROM และ Compact Disc Interactive ซึ่งมี Standard Identifier คือ CD-I

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

A B C D E F G H I J K L M N O P Q R

S T U V W X Y Z 0 1 2 3 4 5 6 7 8 9 _

รูปที่ 6.3 d-character

Volume Set Identifier เป็นการระบุชื่อของ Multiple Volume Set 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

System Identifier จะระบุว่าระบบนั้นสามารถยอมรับได้และสามารถทำให้เกิด logic บนเซคเตอร์ 0-15 ได้ ตัวอักษรที่ถูกใช้ในระบบ ID คือสิ่งที่ ISO 9660 เรียกว่า a-character และมีความยาวจำกัดที่ 31 ตัวอักษร

A B C D E F G H I J K L M N O P Q R S

T U V W X Y Z 0 1 2 3 4 5 6 7 8 9 _ sp

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

รูปที่ 6.4 a- character

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Volume Size เป็นจำนวนซึ่งบอกระบบปฏิบัติการว่ามี Logical Block อยู่ใน Volume นั้นเท่าไหร่ Logical Block เป็นวิธีทั่วไปในการหาตำแหน่งของข้อมูลที่อยู่ใน Volume ซึ่งแต่ละตำแหน่งทั้งหมดนี้จะมี Logical Block Number

Volume Set Size เป็นจำนวนซึ่งบอกระบบปฏิบัติการว่ามี Volume อยู่ใน Volume Set นั้นเท่าไหร่ Volume Sequence Number เป็นตำแหน่งใน Multiple Volume Set ที่ Volume นั้นอยู่ ยกตัวอย่างเช่น แผ่นดิสก์แผ่นหนึ่ง มี Volume Set Size = 5 และ Volume Sequence Number = 3 แสดงว่าดิสก์แผ่นนี้เป็นดิสก์แผ่นที่ 3 ของดิสก์ชุดที่ 5

Logical Block Size เป็นจำนวนไบต์ที่จัดรูปแบบแล้วเกิดที่ว่างที่น้อยที่สุดซึ่งถูกจัดสรรใน Volume นั้นๆ จำนวนนี้สามารถเป็น 512, 1024 หรือ 2048 ไบต์ก็ได้ ส่วนมากดิสก์ ISO 9660 จะใช้ Logical Block Size ขนาด 2048 ไบต์ ซึ่งมีขนาดเดียวกับขนาดของเซคเตอร์

Path Table Size จะบอกระบบปฏิบัติการว่ามีจำนวนไบต์เท่าไหร่ที่อยู่ใน Path Table ระบบปฏิบัติการส่วนมากจะใช้ Path Table ใน fast memory, local memory (RAM) การใช้ Path Table Size เป็นวิธีที่รวดเร็วสำหรับระบบปฏิบัติการที่จะรู้ว่าจะต้องจัดสรรหน่วยความจำเท่าไหร่ก่อนที่มันจะอ่าน Path Table วิธีนี้ทำให้ระบบปฏิบัติการอ่าน Path Table เพียงครั้งเดียวเท่านั้น เป็นการประหยัดเวลา

Root Directory Record จะบรรจุข้อมูลที่ระบบปฏิบัติการต้องการในการระบุตำแหน่งและอ่านไดเรคทอรี่ที่อยู่ระดับบนสุด รูปแบบของ Root Directory Record จะเหมือนกับ Directory Record อื่น ๆ

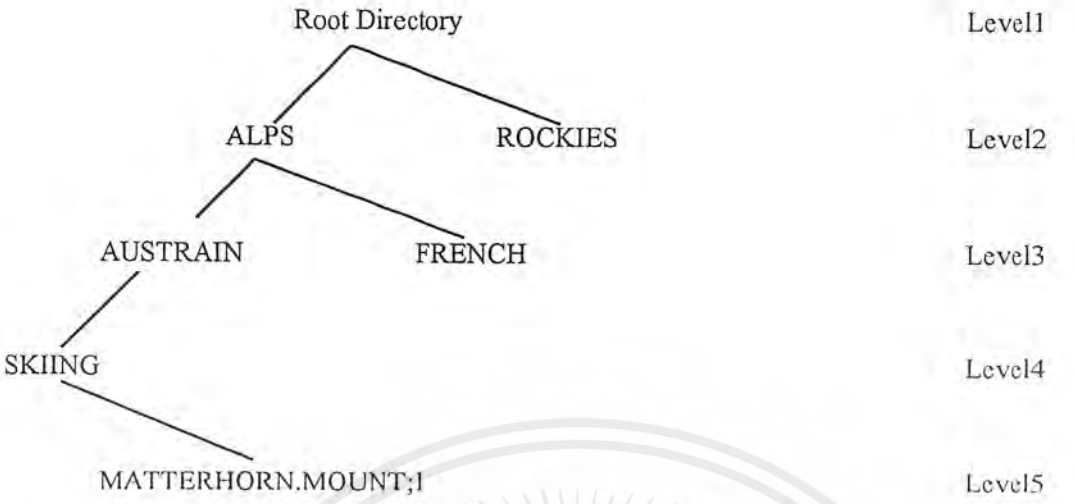
Identifier อื่น ๆ ใน Primary Volume Descriptor นั้นจะบรรจุข้อมูลเกี่ยวกับว่าใครเป็นผู้ตีพิมพ์ Volume นี้, ใครเตรียมข้อมูล, การประยุกต์ใช้งาน และชื่อของไฟล์ซึ่งบรรจุ copyright notice, บทคัดย่อ และบรรณานุกรม

Time Stamps เป็นฟิล์ว (field) ใน Primary Volume Descriptor ซึ่งบรรจุข้อมูลเกี่ยวกับว่า Volume นั้นถูกสร้างขึ้นเมื่อไหร่, ถูกปรับปรุงเมื่อไหร่, เมื่อไหร่ที่ข้อมูลนั้นใช้ได้และเมื่อไหร่ที่เลิกใช้ข้อมูล

6.2.2 โครงสร้างของไดเรคทอรี

โครงสร้างของไดเรคทอรีตาม ISO 9660 ได้ถูกจัดแบ่งเป็นระดับชั้นต่าง ๆ คล้ายกับระบบไฟล์ส่วนใหญ่ ส่วนบนสุดของระดับชั้น คือ Root Directory ซึ่งจะระบุตำแหน่งอยู่ใน Primary Volume Descriptor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.5 ระดับชั้นของไดเรคทอรี่

จากรูป Root Directory เป็นไดเรคทอรีที่อยู่ Level1 เท่านั้น ไดเรคทอรี่ย่อย ALPS และ ROCKIES จะอยู่ Level2, ไดเรคทอรีย่อย AUSTRAIN และ FRENCH จะอยู่ Level3, ไดเรคทอรีย่อย SKIING จะอยู่ Level4 และไฟล์ MATTERHORN.MOUNT;1 จะอยู่ Level5

ISO 9660 ได้กำหนดให้สามารถมีความกว้างของDirectory Structure ได้สูงสุด คือ Level8 นอกจากนี้ยังกำหนดความยาว path ของแต่ละไฟล์ ซึ่งหาได้จากผลรวมของความยาวไดเรคทอรี่ทั้งหมดที่เกี่ยวข้อง, ความยาวของ File Identifier และจำนวนไดเรคทอรี่ที่เกี่ยวข้อง ความยาว path จะต้องไม่เกิน 225 จากรูปที่ 6.5 จะมีความยาว path คือ 39 ดังตารางที่ 6.1

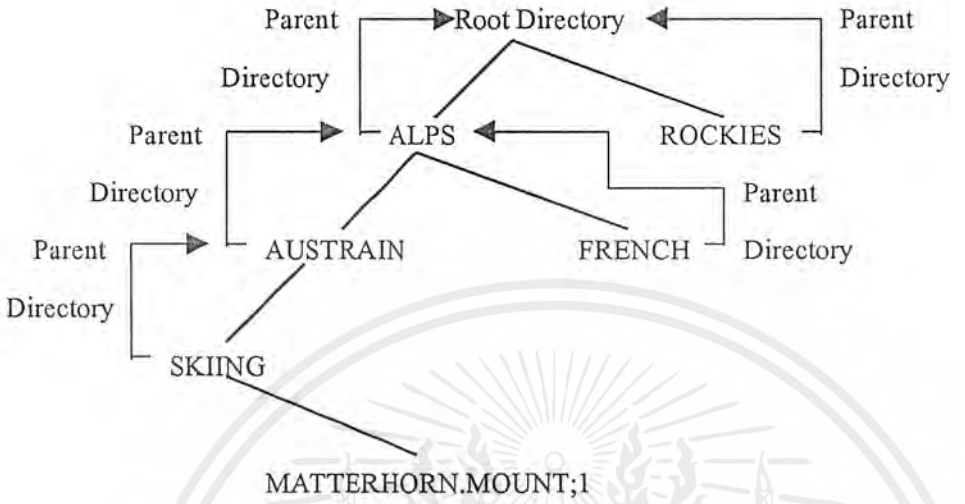
ตารางที่ 6.1 ความยาวของ path

Identifier	ความยาว
ALPS	4
AUSTRAIN	8
SKIING	6
MATTERHORN.MOUNT;1	18
จำนวนของไดเรคทอรี่	3
ผลรวมของความยาวและจำนวนไดเรคทอรี่	39

ไดเรคทอรี่ใน ISO 9660 Volume จะถูกบันทึกอยู่ในรูปของไฟล์ซึ่งบรรจุกลุ่มของ directory record ในแต่ละ directory record จะอธิบายถึงไฟล์หนึ่งหรือไดเรคทอรี่อื่น แต่ละไดเรค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทอรี่จะมีไดเรคทอรี่แม่ (parent directory) ในไดเรคทอรี่แม่จะบรรจุ directory record ซึ่งใช้ระบุไดเรคทอรี่นั้น แสดงรูปที่ 6.6



รูปที่ 6.6 ไดเรคทอรี่แม่

ชื่อไฟล์

ตามมาตรฐาน ISO 9660 ทุก ๆ ไฟล์และทุก ๆ ไดเรคทอรีจะต้องมีชื่อ และชื่อนั้นเรียกว่า File Identifier โดยที่ File Identifier จะประกอบไปด้วย 5 ส่วน ดังแสดงในตารางที่ 6.2

ตารางที่ 6.2 File Identifier

	1) ชื่อไฟล์	2) SEPARATOR 1	3) File Name Extension	4) SEPARATOR 2	5) File Version Number
contents	d-characters	.	d-characters	;	ตัวเลขจาก 1 ถึง 32767
ไฟล์1	MATTERHORN	.	MOUNT	;	1
ไฟล์2	PIKES_PEAK	.		;	1
ไฟล์3		.	HILLS	;	1
ไดเรคทอรี	SKIING				

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

File Identifier จะต้องเป็นไปตามเงื่อนไขดังนี้

- ถ้า File Name ไม่มีตัวอักษร แล้ว File Name Extension จะต้องมีอย่างน้อย 1 ตัวอักษร ดังแสดงในตารางที่ 6.2 ไฟล์3
- ถ้า File Name Extension ไม่มีตัวอักษร แล้ว File Name จะต้องมีอย่างน้อย 1 ตัวอักษร ดังแสดงในตารางที่ 6.2 ไฟล์2
- ผลรวมความยาวของจำนวนตัวอักษรของ File Name และ File Name Extension จะต้องไม่เกิน 30 ตัวอักษร

ส่วนไดเรคทอรี ตามมาตรฐาน ISO 9660 จะถูกกำหนดให้มีแค่ชื่อไฟล์เท่านั้น จะไม่มี SEPARATOR1(.), SEPARATOR2(;), File Name Extension หรือ File Version Number ดังแสดงในตารางที่ 6.2 ไดเรคทอรี

6.2.3 Path Table

Path Table เป็นวิธีลัดที่ระบบปฏิบัติการเลือกใช้ในการไปสู่แต่ละไดเรคทอรีบนดิสก์เพื่อที่จะให้ได้ไฟล์ที่ต้องการ Path Table จะเก็บชื่อไดเรคทอรี, ชื่อไดเรคทอรีแม่ และที่อยู่สำหรับแต่ละไดเรคทอรียกเว้น Root Directory

ส่วนใหญ่ระบบปฏิบัติการจะอ่าน Path Table ก่อน 1 ครั้งและเก็บไว้ในหน่วยความจำ มากกว่าที่จะอ่านหลาย ๆ ครั้ง ตัวอย่างดังรูปที่ 6.5 ถ้าระบบปฏิบัติการไม่ใช้ Path Table แล้ว จะต้องอ่าน Root Directory เพื่อที่จะหาตำแหน่งของไดเรคทอรี ALPS จากนั้นอ่านไดเรคทอรี ALPS เพื่อที่จะหาตำแหน่งของไดเรคทอรี AUSTRAIN แล้วอ่านไดเรคทอรี SKIING เพื่อที่จะหาตำแหน่งของไฟล์ MATTERHORN.MOUNT;1 แต่ถ้าใช้ Path Table ระบบปฏิบัติการจะค้นหาตำแหน่งของไดเรคทอรี SKIING ใน Path Table, อ่านไดเรคทอรี SKIING และหาตำแหน่งของไฟล์

จากตัวอย่างข้างต้นจะพบว่าการค้นหาตำแหน่งของไฟล์โดยไม่ใช้ Path Table จะต้องอ่านข้อมูลจาก CD-ROM ถึง 4 ครั้ง แต่ถ้าใช้ Path Table จะอ่านข้อมูลเพียงแค่ครั้งเดียว ซึ่งใน CD-ROM ทั่ว ๆ ไป การขับ (drive) จะใช้เวลาในการเข้าถึงข้อมูล (seek time) ประมาณ 0.25 วินาที ดังนั้นช่วงเวลาที่ต่างกันในการค้นหาจะเท่ากับ 0.75 วินาที ในกรณีที่ต้องการเข้าถึงไฟล์จำนวนมาก ช่วงเวลาที่ต่างกันนี้จะมีผลต่อการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7
เครื่องถอดรหัสเอ็มเป็ก 1 เลเยอร์ 3

ในบทนี้จะกล่าวถึงส่วนประกอบและหน้าที่ของอุปกรณ์ต่างๆ รวมถึงหลักการทำงานของโปรแกรมควบคุม ลักษณะการเชื่อมต่อ

7.1 โครงสร้างฮาร์ดแวร์

7.1.1 หน้าที่การทำงานของแต่ละส่วน

เครื่องถอดรหัส MP3 นี้สามารถแบ่งการทำงานออกเป็นส่วนๆ โดยแต่ละส่วนมีหน้าที่การทำงานหลักๆ ดังนี้

1. ชุดถอดรหัสเอ็มเป็ก

ส่วนชุดถอดรหัสนี้มีหน้าที่การทำงานหลักคือแปลงข้อมูลเสียงที่ถูกบีบอัดตามมาตรฐาน MPEG1 Layer 1,2 และ 3 ให้กลับมาเป็นสัญญาณเสียงเหมือนเดิม โดยในการทำงานจะประกอบด้วยวงจรรวมสองตัวที่ทำงานร่วมกันคือ

- MAS3507D ทำหน้าที่ถอดรหัสข้อมูลที่ถูกบีบอัดตามมาตรฐาน MPEG1 Layer 1,2 และ 3เป็นสัญญาณออดิโอที่เป็นดิจิตอล
- DAC3550A ทำหน้าที่แปลงข้อมูลดิจิตอลที่ได้จาก MAS3507D เป็นสัญญาณอนาลอก และสร้างสัญญาณนาฬิกาป้อนแก่ตัว MAS3507D

2. ชุดควบคุมการอ่านข้อมูลจากซีดีรอม

หน้าที่การทำงานของส่วนนี้คือการควบคุมซีดีรอมให้ทำงานตามที่ต้องการ และรับคำสั่งจากผู้ใช้ ซึ่งสามารถแยกการทำงานออกเป็นข้อย่อยๆ ได้ดังนี้

- อ่านโครงสร้างข้อมูลภายในแผ่นซีดีรอมเพื่อเก็บข้อมูลชื่อ ตำแหน่ง ความยาว ของไฟล์หรือไดเรกทอรี่
- จัดโครงสร้างข้อมูลที่อ่านได้ให้อยู่ในรูปแบบที่นำไปใช้งานได้ง่าย และเก็บไว้ในเมมโมรี่ของเครื่องเพื่อใช้ในการเลือกเพลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

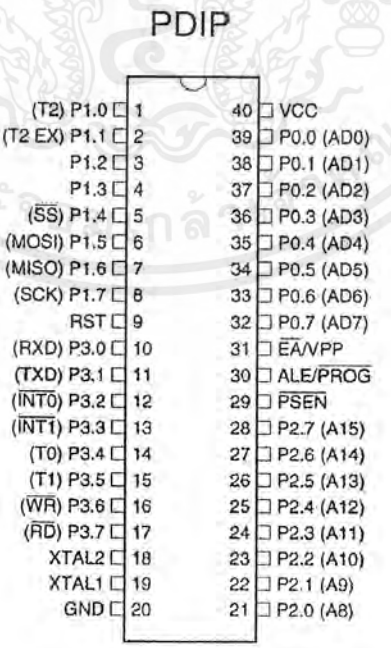
- อ่านข้อมูลเอ็มเป็กในตำแหน่งและความยาวตามที่ผู้ใช้ต้องการ
- จัดรูปแบบและส่งข้อมูลเอ็มเป็กเข้าสู่ตัวถอดรหัส MAS3507D
- คอยรับการสั่งงานจากผู้ใช้โดยอาศัยการ scan key
- ส่งข้อมูลที่ต้องการแสดงผลไปยัง 89C51 เพื่อนำไปแสดงผลบนจอ LCD ต่อไป

การทำงานของส่วนนี้ใช้ไมโครคอนโทรลเลอร์เบอร์ 89S8252 เป็นหัวใจหลักในการทำงาน ซึ่งต่อร่วมกับวงจรจัดลอจิกภายนอกเพื่อให้ข้อมูลที่ได้จากซีดีรอมซึ่งมีขนาด 16 บิตสามารถใช้งานกับไมโครคอนโทรลเลอร์ขนาด 8 บิตได้

ไมโครคอนโทรลเลอร์เบอร์ 89S8252 ซึ่งเป็นตระกูล MCS-51 ได้เพิ่มความสามารถขึ้นมาอีกหลายประการ เพื่อเป็นการง่ายต่อการทำความเข้าใจจะขออธิบายถึงความสามารถที่เพิ่มขึ้นนี้

ลักษณะเด่นของ 89S8252

- มีหน่วยความจำสำหรับเก็บโปรแกรม จำนวน 8 กิโลไบต์
- มีหน่วยความจำชนิด EEPROM จำนวน 2 กิโลไบต์
- สามารถใช้งานได้กับคริสตัลความถี่ 0-24 MHz
- มีรีจิสเตอร์สำหรับใช้งานเป็นไทม์เมอร์หรือเคาร์เตอร์ ขนาด 16 บิต จำนวน 3 ตัว
- มีพอร์ตสื่อสารอนุกรมแบบ SPI
- มีสัญญาณ Watchdog อยู่ภายใน ซึ่งสามารถใช้หรือไม่ก็ได้ตามความต้องการ
- มีรีจิสเตอร์ที่ใช้ชี้ตำแหน่งข้อมูล(DPTR) จำนวน 2 ตัว



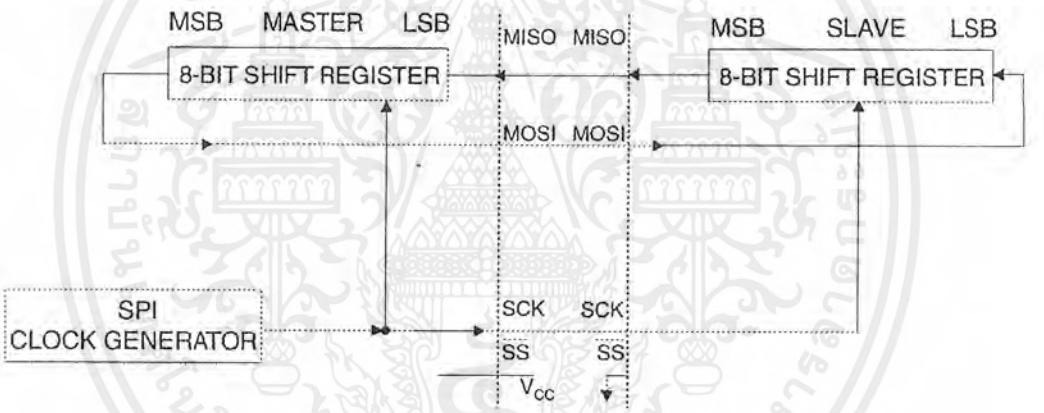
รูปที่ 7.1 แสดงโครงสร้างภายนอกของ 89S8252

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

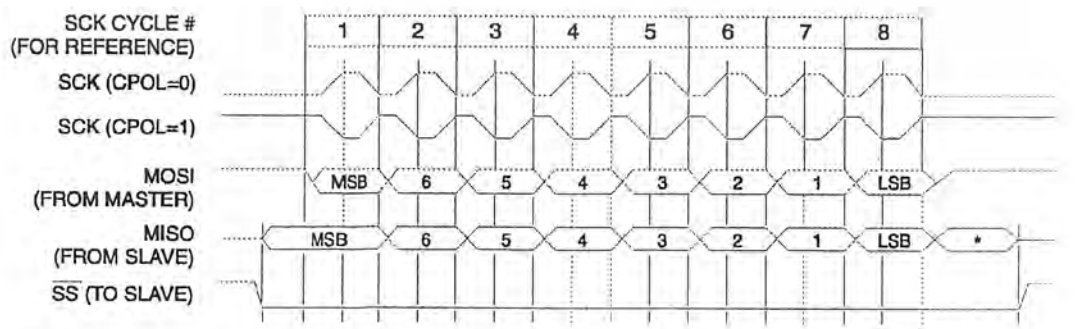
- พอร์ตสื่อสารอนุกรมแบบ SPI

SPI (Serial peripheral interface) เป็นการสื่อสารอนุกรมที่มีความเร็วสูงอีกทั้งยังเป็นการรับส่งแบบ Full-Duplex อีกด้วย การเชื่อมต่อแบบ SPI มีรูปแบบดังรูปที่ 7.2 โดยรายละเอียดของสัญญาณมีดังนี้

1. สัญญาณ SCK เป็นสัญญาณที่ตัว MASTER เป็นตัวสร้างขึ้นเพื่อใช้กำหนดจังหวะในการติดต่อสื่อสาร
2. สัญญาณ MOSI เป็นสัญญาณข้อมูลที่ตัว MASTER เป็นตัวส่งออกไปให้ SLAVE
3. สัญญาณ MIOS เป็นสัญญาณข้อมูลที่ตัว SLAVE เป็นตัวส่งมายัง MASTER
4. สัญญาณ SS\ เป็นสัญญาณทีส่งจาก MASTER ไปยัง SLAVE เพื่อเป็นการอนุญาตให้ทำการส่งข้อมูลได้



รูปที่ 7.2 แสดงการเชื่อมต่อพอร์ตสื่อสารอนุกรมแบบ SPI



รูปที่ 7.3 แสดง Timing Diagram ของการเชื่อมต่อแบบ SPI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สัญญาณ Watchdog

สัญญาณ Watchdog เป็นสัญญาณที่ใช้รีเซท CPU หาก CPU ไม่มีการเซตค่าในบิตที่กำหนดภายในเวลาที่ตั้งไว้ สัญญาณ Watchdog จะมีประโยชน์มากในระบบที่ต้องการการรีเซตหากระบบเกิดความผิดพลาด

89S8252 สามารถที่จะเลือกใช้สัญญาณ Watchdog หรือไม่ก็ได้ขึ้นกับความต้อง การตั้งค่าเวลาของสัญญาณ Watchdog สามารถตั้งได้ถึง 8 ระดับ โดยระยะเวลานี้ไม่ขึ้นกับความถี่ของคริสตัลที่เลือกใช้

- หน่วยความจำชนิด EEPROM ภายใน 89S8252

หน่วยความจำชนิด EEPROM ภายใน 89S8252 มีขนาด 2 กิโลไบต์ ในการเขียนหรืออ่านจะให้เฉพาะคำสั่ง MOVX เท่านั้น ในการใช้งานจำเป็นต้องเซตค่าในบิตที่กำหนดเพื่อบอก CPU ให้ทราบว่าเป็นการอ้างอิงหน่วยความจำชนิดใด เนื่องจากมีตำแหน่งตั้งแต่ 000h- 7FFh ซึ่งซ้อนทับกับตำแหน่งหน่วยความจำชนิด RAM ภายใน ในขั้นตอนการเขียนหน่วยความจำชนิดนี้แต่ละครั้งต้องใช้เวลาประมาณ 2.5mS จึงจะสามารถเขียนครั้งต่อไปได้ ดังนั้นก่อนการเขียนแต่ละครั้งจำเป็นต้องเช็คบิตสถานะของการเขียนก่อนว่าพร้อมแล้วหรือยัง

- ความสามารถในการสร้างสัญญาณนาฬิกา

ความสามารถนี้จะใช้ Timer/Counter 2 ในการกำหนดค่าความถี่ที่ต้องการโดยมีสูตรการคำนวณดังนี้

$$\text{Clock Out Frequency} = \frac{\text{Oscillator Frequency}}{4 \times [65536 - (\text{RCAP2H, RCAP2L})]}$$

หมายเหตุ RCAP2H, RCAP2L คือค่าใน Timer/Counter 2 ในโหมด auto reload

3. ส่วนแสดงผลการทำงานออกทางจอ LCD

หน้าที่การทำงานหลักของส่วนนี้คือการรับข้อมูลจากชุดควบคุมการอ่านซีดีรอมมาแสดงผลออกทาง LCD โดยจอ LCD ที่ใช้เป็นจอ LCD แสดงผลแบบกราฟฟิกเพื่อให้สามารถแสดงผลเป็นภาษาไทยได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.1.2 การเชื่อมต่อระหว่างแต่ละส่วน

1. การเชื่อมต่อจากไดรฟ์ซีดีรอมเข้ากับชุดควบคุมการอ่านซีดีรอม

ใช้การเชื่อมต่อแบบ IDE 40 pin โดยรายละเอียดของขาต่าง ๆ สามารถดูได้จากบทที่ 5 ในบทนี้จะกล่าวถึงสัญญาณที่สำคัญในการควบคุมการทำงานของไดรฟ์ซีดีรอมเท่านั้น

- CA 1 และ CA 2 เป็นสัญญาณที่ส่งจากชุดควบคุมมายังไดรฟ์ซีดีรอมเพื่อใช้เลือกกลุ่มของรีจิสเตอร์ภายในไดรฟ์ซีดีรอม
- DA0,DA1และ DA2 เป็นสัญญาณที่ส่งจากชุดควบคุมมายังไดรฟ์ซีดีรอมเพื่อเลือกรีจิสเตอร์ที่จะติดต่อด้วย
- DIOR- เป็นสัญญาณสโตรปในการอ่านข้อมูลจากชุดควบคุมส่งมายังไดรฟ์ โดยขอบขาลงของ DIOR- จะอนุญาตให้ข้อมูลจากรีจิสเตอร์เข้ามาอยู่ในสายสัญญาณได้
- DIOW- เป็นสัญญาณสโตรปในการเขียนข้อมูลจากชุดควบคุมส่งมายังไดรฟ์ซีดีรอม โดยขอบขาขึ้นของ DIOW- จะแลทช์ข้อมูลจากสายสัญญาณลงในรีจิสเตอร์หรือพอร์ตข้อมูลของอุปกรณ์
- RESET สัญญาณในการรีเซทอุปกรณ์

ในการเชื่อมต่อนี้จะใช้โปรโตคอล ATA และATAPI (ดูรายละเอียดต่างๆได้จากบทที่ 5)

เนื่องจากข้อมูลที่ติดต่อกับซีดีรอมเป็นข้อมูลขนาด 16 บิต แต่ไมโครคอนโทรลเลอร์ที่ใช้ในส่วนควบคุมเป็นไมโครคอนโทรลเลอร์ขนาด 8 บิต ทำให้จำเป็นต้องมีวงจรจัดลอจิกส่วนหนึ่งเป็นตัวคั่นกลางระหว่างไดรฟ์ซีดีรอมและตัวไมโครคอนโทรลเลอร์

2. การเชื่อมต่อระหว่างชุดควบคุมการอ่านซีดีรอมกับชุดถอดรหัส

ในการทำงานของตัว MAS3507D จะต้องโหลดข้อมูลเข้าไปเก็บไว้ในบัฟเฟอร์ภายในตัวของมันก่อน โดยในขณะที่บัฟเฟอร์ภายในของ MAS3507D ว่างหรือข้อมูลกำลังจะถูกใช้หมดจะมีการส่งสัญญาณออกมาบอกให้เติมบัฟเฟอร์ สัญญาณนั้นก็คือสัญญาณดีมานด์ (Demand) ดังนั้นจึงต้องต่อสัญญาณนี้เข้ากับ 89S8252 เพื่อให้ตัว 89S8252 ทราบว่าชุดถอดรหัสต้องการข้อมูล

ในส่วนของสตรีมข้อมูลเอ็มเป็กซึ่งต้องป้อนผ่านทาง I^2S บัสนั้นอาศัยความสามารถพิเศษของ 89S8252 ซึ่งสามารถส่งสัญญาณในรูปแบบ I^2S ที่มีความเร็วให้เลือกใช้งานได้หลายค่า ในชิ้นงานนี้เลือกใช้ความถี่ 375 Kbps

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. การเชื่อมต่อระหว่าง MAS3507D กับ DAC3550A

สัญญาณที่จำเป็นระหว่าง MAS3507D และ DAC3550A ก็คือสัญญาณออดิโอที่ถอดรหัสออกมาได้ในรูปแบบดิจิตอล และสัญญาณนาฬิกา

ในส่วนของข้อมูลออดิโอที่ถอดรหัสได้จะมีสัญญาณที่ต้องเชื่อมต่ออยู่ 3 สัญญาณ คือ ข้อมูลดิจิตอล สัญญาณนาฬิกา และสัญญาณบอกเฟรมข้อมูล

สำหรับสัญญาณนาฬิกานั้นตัว DAC3550A จะเป็นตัวสร้างจากวงจรออสซิลเลเตอร์ภายในเพื่อป้อนกลับไปยังตัว MAS3507D

4. การเชื่อมต่อระหว่างชุดควบคุมการอ่านซีดีรอมกับส่วนแสดงผลทาง LCD

การส่งข้อมูลระหว่างส่วนการทำงานสองส่วนนี้ก็คือการส่งข้อมูลระหว่างไมโครคอนโทรลเลอร์ 89S8252 ซึ่งเป็นตัวควบคุมชุดควบคุมการอ่านซีดีรอม กับไมโครคอนโทรลเลอร์ 89C51 ซึ่งเป็นตัวควบคุมการแสดงผลของ LCD

ข้อมูลที่ส่งในส่วนนี้จะติดต่อผ่านพอร์ตข้อมูลอนุกรมโหมด 1 ของไมโครคอลโทรลเลอร์ตระกูล MCS-51 โดยข้อมูลที่ส่งจะอยู่ในรูปของรหัส ASCII ในการใช้งานนั้น 89S8252 จะส่งข้อความที่ต้องการแสดงผลไปยัง 89C51 หลังจากที่ 89C51 ได้รับข้อมูลครบก็จะทำการจัดรูปแบบข้อมูลแล้วสั่งงานให้จอ LCD แบบ Graphic แสดงผลต่อไป

7.2 ซอฟท์แวร์ควบคุมการทำงาน

จากโครงสร้างทางฮาร์ดแวร์จะเห็นได้ว่ามีการทำงานสองส่วนที่จำเป็นต้องใช้ไมโครคอนโทรลเลอร์เป็นตัวควบคุมการทำงาน ซึ่งจำเป็นจะต้องมีการโปรแกรมการทำงานของไมโครคอนโทรลเลอร์นั้นๆ

7.2.1 โปรแกรมควบคุมการทำงานภายในไมโครคอนโทรลเลอร์ 89S8252

ไมโครคอลโทรลเลอร์เบอร์ 89S8252 เป็นหัวใจสำคัญของชุดควบคุมการอ่านซีดีรอม มีหน้าที่การทำงานหลักๆ ตามหัวข้อ 7.1.1 ซึ่งสามารถแยกอธิบายการทำงานของโปรแกรมออกเป็นช่วงๆ ได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **การอ่านโครงสร้างไฟล์ของแผ่นซีดีรอมและจัดรูปแบบข้อมูลที่ได้**

การทำงานนี้เป็นขั้นตอนแรกในการเริ่มใช้งานเครื่อง คือเป็นการอ่าน Primary Volume Descriptor ของแผ่นซีดีรอมเพื่อหาตำแหน่งของ Path Table ซึ่งตัว Path Table จะเป็นจุดรวมของตำแหน่งไดเรกทอรี่ทั้งหมดในแผ่นซีดีรอม จากนั้นก็นำข้อมูลที่ได้มาจัดในรูปแบบที่ตัวเครื่องสามารถเรียกใช้ได้อย่างรวดเร็วและเก็บไว้ในหน่วยความจำเพื่อใช้ในการแสดงผลและเป็นตำแหน่งในการเรียกอ่านไดเรกทอรี่ต่างๆ

ในการจัดข้อมูลที่อ่านได้จาก Path Table นั้นต้องตรวจสอบก่อนว่าแผ่นซีดีรอมนั้นๆ เขียนมาในมาตรฐานใด หลังจากตรวจสอบมาตรฐานของแผ่นซีดีรอมนั้นแล้ว ก็จะจัดข้อมูลไดเรกทอรี่ที่ได้ตามมาตรฐานนั้นๆ

การทำงานในส่วนนี้สามารถอธิบายในรูปของโฟลว์ชาร์ทได้ดังรูปที่ 7.4

- **การอ่านข้อมูลจากแผ่นซีดีรอมในตำแหน่งเพลงที่ผู้ใช้เลือก**

การทำงานในส่วนนี้จะเกิดขึ้นหลังจากผ่านการทำงานในส่วนแรกไปแล้วและ ผู้ใช้ได้กดเลือกอัลบัม และชื่อเพลงเรียบร้อยแล้ว การทำงานหลักของส่วนนี้เริ่มจากตรวจสอบตำแหน่งไฟล์และความยาวของไฟล์ที่ผู้ใช้เลือก เมื่อได้ตำแหน่งไฟล์และความยาวแล้ว ก็จะทำการอ่านข้อมูลเพื่อเติมบัฟเฟอร์ก่อนจะเริ่มเล่นเพลง (Pre-buffering) เมื่อเติมบัพเฟอร์จนครบที่กำหนดไว้แล้วก็จะเอนเนเบิลอินเทอร์รัพท์จากสัญญาณดีมานต์ของส่วนถอดรหัส และอินเทอร์รัพท์ของพอร์ตอนุกรม SPI หลังจากนั้นก็จะอ่านซีดีรอมเพื่อเติมบัฟเฟอร์ไปเรื่อยๆ เนื่องจากส่วนโปรแกรมบริการอินเทอร์รัพท์จะอ่านข้อมูลออกจากเมมโมรี่ส่งไปยังส่วนถอดรหัสอยู่ตลอดเวลา

การอ่านข้อมูลจากซีดีรอมจะกระทำเป็นเพจ โดยแต่ละเพจมีขนาด 16 Kbyte เมื่ออ่านข้อมูลครบแต่ละเพจจะตรวจสอบว่ามีการกดคีย์ใดๆ หรือไม่ ถ้ามีการกดคีย์ก็จะหยุดการเล่นเพลงกลับไปอยู่ในส่วนการรับคำสั่งจากผู้ใช้อีกครั้ง

การทำงานในส่วนนี้สามารถอธิบายในรูปของโฟลว์ชาร์ทได้ดังรูปที่ 7.5

- **โปรแกรมบริการอินเทอร์รัพท์ของ Timer0**

โปรแกรมบริการอินเทอร์รัพท์ของ Timer0 นี้ใช้ควบคุมการตรวจสอบการกดคีย์ (Scan Key) โดยจะทำการตรวจสอบการกดคีย์ทุก 30 ms เมื่อตรวจพบว่ามีการกดคีย์ก็จะเช็คว่าคีย์ที่ถูกกดเป็นคีย์ใด จากนั้นจะเซ็ตบิตแสดงสถานะการกดคีย์เพื่อให้โปรแกรมหลักรู้

การทำงานในส่วนนี้สามารถอธิบายในรูปของโฟลว์ชาร์ทได้ดังรูปที่ 7.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **โปรแกรมบริการอินเทอร์รัพท์ของพอร์ตอนุกรม SPI และ สัญญาณดีมานต์**

หน้าที่ของโปรแกรมบริการอินเทอร์รัพท์สองส่วนนี้มีหน้าที่การทำงานที่เหมือนกันคือจะอ่านข้อมูลที่เก็บบัฟเฟอร์ไว้ในเมมโมรี่แล้วส่งออกไปยังชุดถอดรหัส

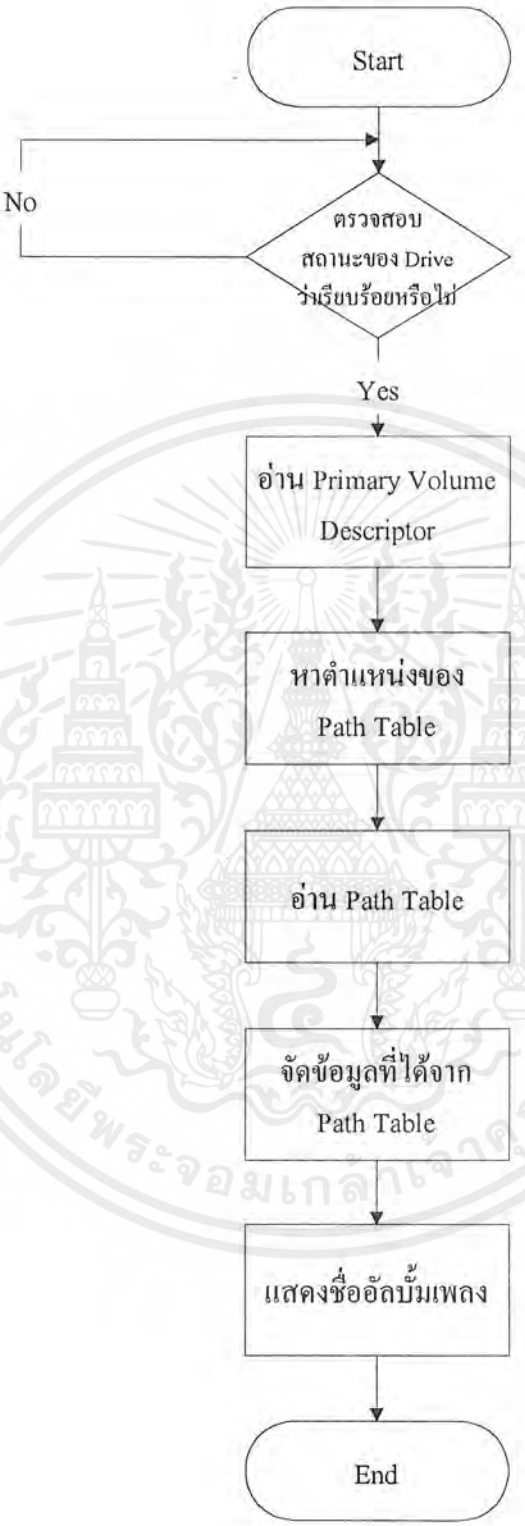
อินเทอร์รัพท์จากสัญญาณดีมานต์คืออินเทอร์รัพท์ภายนอก0 (Ex0) กล่าวคือเมื่อเติมบัฟเฟอร์ก่อนเล่นเพลงจนครบตามที่กำหนดแล้วโปรแกรมหลักจะเอนเนเบิลอินเทอร์รัพท์ซึ่งจะเกิดการอินเทอร์รัพท์จากสัญญาณดีมานต์นี้ขึ้นแทบจะในทันที การทำงานของโปรแกรมส่วนนี้ก็จะทำการอ่านข้อมูลจากเมมโมรี่ที่บัฟเฟอร์ไว้แล้วไปทางพอร์ตอนุกรม SPI

หลังจากข้อมูลไบต์แรกถูกส่งออกไปเรียบร้อยแล้ว การอินเทอร์รัพท์ครั้งต่อไปจะเป็นอินเทอร์รัพท์ที่เกิดจากพอร์ตอนุกรม SPI ต่อเนื่องไปเรื่อยๆ จนกว่าจะหมดสัญญาณดีมานต์จากส่วนถอดรหัส การทำงานในส่วนนี้สามารถอธิบายในรูปของโฟลว์ชาร์ทได้ดังรูปที่ 7.7

7.2.2 โปรแกรมควบคุมการทำงานภายในไมโครคอนโทรลเลอร์ 89C51

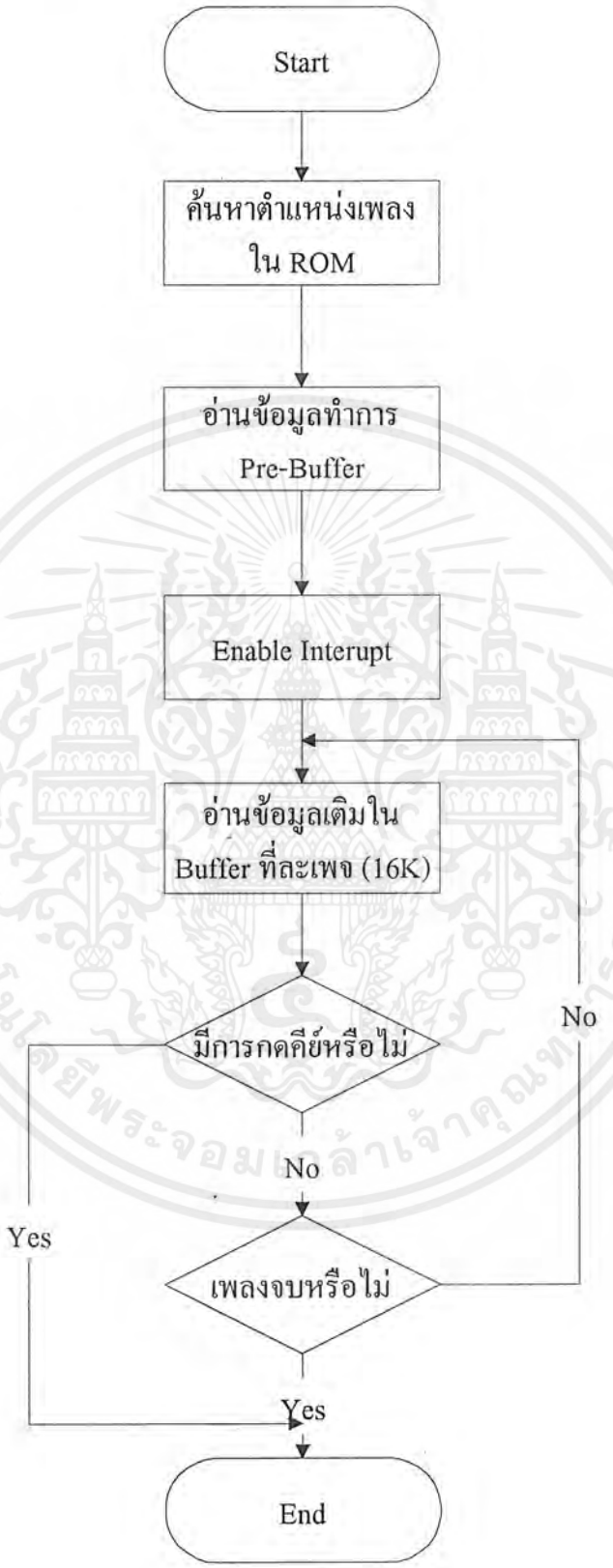
ไมโครคอนโทรลเลอร์ 89C51 นี้เป็นหัวใจหลักของส่วนแสดงผล หน้าที่หลักของมันคือแสดงผลออกทางจอ LCD ให้ได้ตามที่ส่วนควบคุมการอ่านซีดีรอมส่งมา ข้อมูลที่ส่งมาจะอยู่ในรูปแบบ ASCII ผ่านทางพอร์ตอนุกรม เนื่องจากความต้องการแสดงผลที่มีรูปแบบต่างๆกันหลายรูปแบบ เช่น การมีแถบดำ การเลื่อนข้อความเพื่อแสดงข้อความยาวๆ ฯลฯ ดังนั้นเพื่อเป็นการแบ่งเบาภาระของส่วนควบคุมการอ่านซีดีรอม หลังจากที่ 89C51 รับข้อมูลจนครบแล้ว 89C51 จะทำการเช็คก่อนว่าส่วนควบคุมต้องการให้แสดงผลแบบใด (โดยการเช็คไบต์แรกของข้อมูล) หลังจากนั้น 89C51 จึงทำการจัดรูปแบบของข้อมูลที่ได้รับมาแล้วส่งออกไปให้ LCD แสดงผลต่อไป หลังจากสั่งงาน LCD เสร็จแล้ว 89C51 ก็จะคอยรับข้อมูลจาก ส่วนควบคุมการอ่านซีดีรอมเพื่อแสดงผลในครั้งต่อไป การทำงานในส่วนนี้สามารถอธิบายในรูปของโฟล์ชาร์ทได้ดังรูปที่ 7.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



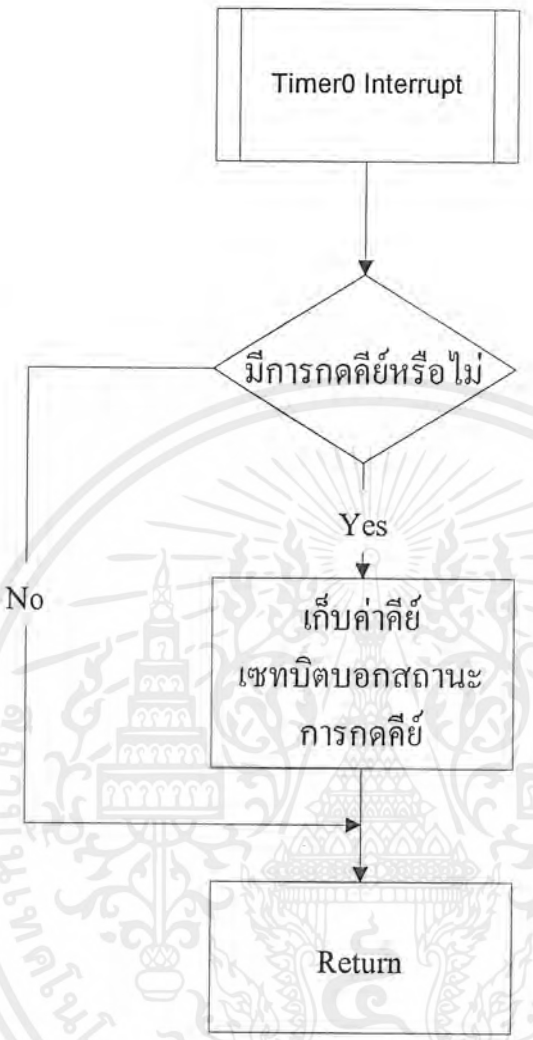
รูปที่ 7.4 โฟลว์ชาร์ทการอ่านโครงสร้างข้อมูลของแผ่นและจัดรูปแบบข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.5 โฟลว์ชาร์ทการอ่านข้อมูลจากแผ่นซีดีรอมในตำแหน่งเพลงที่ผู้ใช้เลือก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

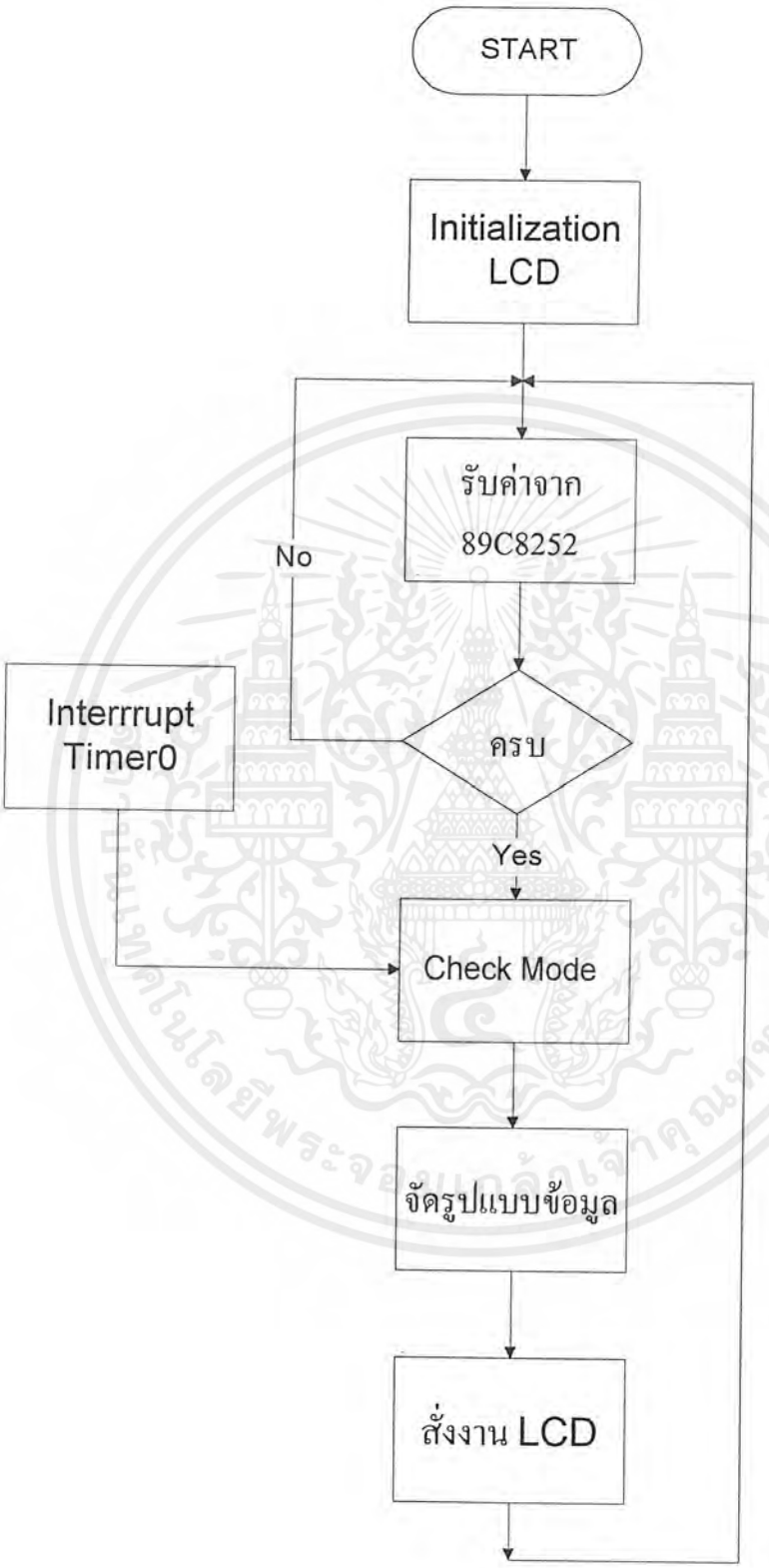


รูปที่ 7.6 โฟลว์ชาร์ทของโปรแกรมบริการอินเทอร์รัพท์ของ Timer0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 7.7 โฟลว์ชาร์ทของโปรแกรมบริการอินเทอร์รัพท์พอร์ตอนุกรมและสัญญาณดีมานด์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.8 โฟลว์ชาร์ตแสดงการทำงานของ 89C51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 8
การทดลองและสรุปผลการทดลอง

8.1 การทดลอง

การทดลองทั้งหมดจะแบ่งออกเป็น 4 ส่วน ดังนี้

8.1.1 การเล่นเพลงที่บีบอัดข้อมูลด้วยอัตราแซมปลิ้งต่างๆ กัน

การทดลองในขั้นนี้ทดลองโดยนำไฟล์เพลงที่ถูกบีบอัดตามมาตรฐาน MPEG1 Layer 3 ซึ่งมีอัตราแซมปลิ้งที่ต่างกันมาทดลองเล่นผลที่ได้เป็นดังนี้

ตารางที่ 8.1 การทดลองเล่นเพลงที่บีบอัดข้อมูลด้วยอัตราแซมปลิ้งต่างๆ

อัตราแซมปลิ้ง	ผลการทดลอง
11.025 K	สามารถถอดรหัสได้ถูกต้อง
22.05 K	สามารถถอดรหัสได้ถูกต้อง
44.1 K	สามารถถอดรหัสได้ถูกต้อง

8.1.2 การทดสอบการเชื่อมต่อกับไดรฟ์ซีดีรอมจากบริษัทต่างๆ

ในขั้นตอนนี้เป็นการทดสอบความเข้ากันได้ของชิ้นงานที่สร้างขึ้นว่าสามารถควบคุมการทำงานของซีดีรอมจากบริษัทผู้ผลิตต่างๆ กันได้มากน้อยเพียงใด ซึ่งการทดลองนี้แบ่งออกเป็นสองส่วนย่อยๆ คือ

- ทำการทดลองโดยนำซีดีรอมของบริษัทผู้ผลิตต่างๆ มาเชื่อมต่อเข้ากับชิ้นงานแล้วสั่งให้ตัวไดรฟ์อ่านข้อมูลจากแผ่นซีดีรอมขึ้นมาว่าสามารถอ่านข้อมูลได้หรือไม่ โดยข้อมูลที่อ่านมาได้จะแสดงผลออกทางโปรแกรม Hyper terminal บนวินโดวส์ เพื่อตรวจสอบความถูกต้องของข้อมูลที่อ่านขึ้นมาได้ ผลการทดลองที่ได้เป็นดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 8.2 การทดสอบการเชื่อมต่อกับไดรฟ์ซีดีรอมจากบริษัทต่างๆ

ยี่ห้อ/ความเร็ว	ผลการทดลอง
Wearnes 12X	อ่านข้อมูลได้ถูกต้อง
GoldStar 16X	อ่านข้อมูลได้ถูกต้อง
Sony 40X	อ่านข้อมูลได้ถูกต้อง
Creative 16X	อ่านข้อมูลได้ถูกต้อง
Pioneer 10X	มีความผิดพลาดเกิดขึ้นในการอ่าน
Pioneer 36X	มีความผิดพลาดเกิดขึ้นในการอ่าน
ASUS 40 X	มีความผิดพลาดเกิดขึ้นในการอ่าน

- การทดลองในขั้นถัดมาคือการทดลองนำไดรฟ์ต่างๆ มาเล่นเพลงจริงๆ ซึ่งจำเป็นต้องอ่านไฟล์อย่างต่อเนื่องได้ผลการทดลองดังนี้

ตารางที่ 8.3 การทดลองเล่นเพลงจากซีดีรอมของบริษัทต่างๆ

ยี่ห้อ/ความเร็ว	ผลการทดลอง
Wearnes 12X	สามารถเล่นเพลงได้อย่างต่อเนื่อง
GoldStar 16X	สามารถเล่นเพลงได้อย่างต่อเนื่อง
Sony 40X	สามารถเล่นเพลงได้อย่างต่อเนื่อง
Creative 16X	สามารถเล่นเพลงได้อย่างต่อเนื่อง
Pioneer 10X	ไม่สามารถเล่นเพลงได้ต่อเนื่อง
Pioneer 36X	ไม่สามารถเล่นเพลงได้ต่อเนื่อง
ASUS 40 X	ไม่สามารถเล่นเพลงได้ต่อเนื่อง

8.1.3 การทดสอบการอ่านแผ่นซีดีรอมที่ถูกเขียนขึ้นมาในระบบไฟล์ที่ต่างกัน

การทดลองในขั้นนี้ทดลองโดยนำแผ่นซีดีรอมที่ถูกเขียนในระบบไฟล์ที่ต่างกันมาทดสอบว่าสามารถอ่านข้อมูลได้หรือไม่

จะขอจำแนกระบบไฟล์ของซีดีรอมในปัจจุบันออกเป็น 2 ส่วนคือ

- ซีดีรอมที่เขียนตามมาตรฐาน ISO9660
- ซีดีรอมที่เขียนตามมาตรฐาน ISO9660 + Juliet

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อทดลองนำแผ่นซีดีรอมที่มีระบบไฟล์ทั้งสองมาตรฐานมาทดลองอ่านผลปรากฏว่าสามารถอ่านชื่อไฟล์ และระบุตำแหน่งได้ไฟล์หรือไดเรคทอรี่ได้ถูกต้อง

8.2 สรุปผลการทดลอง

จากผลการทดลองที่ได้จะเห็นได้ว่าเครื่องถอดรหัส MP3 ที่สร้างขึ้นนี้สามารถถอดรหัสไฟล์เพลงที่ถูกบีบอัดด้วยอัตราแซมปลิ้งต่างๆ กันได้อย่างถูกต้อง และสามารถใช้งานได้กับแผ่นซีดีรอมที่มีระบบไฟล์ตามมาตรฐาน ISO9660 และ ISO9660+Juliet

ในส่วนของความเข้ากันได้ของตัวเครื่องกับไดรฟ์ซีดีรอมจากบริษัทผู้ผลิตต่างๆ นั้น จะเห็นได้ว่ามีไดรฟ์ซีดีรอมจากบางผู้ผลิตที่ไม่สามารถใช้งานได้ ซึ่งเป็นผลมาบริษัทนั้นๆ ไม่ได้สร้างไดรฟ์ให้เป็นไปตามมาตรฐาน ATAPI จึงทำให้ไม่สามารถใช้ไดรฟ์จากบริษัทนั้นๆ อ่านแผ่นได้อย่างถูกต้อง การใช้งานไดรฟ์จากบริษัทเหล่านี้ จำเป็นจะต้องศึกษาไดรเวอร์ของไดรฟ์นั้นๆ ว่ามีความจำเป็นต้องเปลี่ยนคำสั่งส่วนใด และต้องเพิ่มคำสั่งส่วนใดบ้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

- Stephen G. Finch, "Information technology-AT Attachment Interface with Extensions (ATA-2)"
- Tom Hanan, Devon Worrell; "ATA Packet Interface for CD-ROMs (SFF-8020i)", Revision2.6 Proposed; January 22,1996

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

ความหมายของคำในไวยากรณ์ของข้อมูลเอ็มเป็ก 1 เลเยอร์ 3

ก.1 ลำดับสัญญาณเสียงทั่ว ๆ ไป

frame ส่วนของบิตสตรีม (bitstream) ที่สามารถถอดรหัสได้ บรรจุข้อมูล 1,152สัญญาณสุ่มความถี่

ก.2 เฟรมของสัญญาณ (Audio Frame)

header ส่วนของบิตสตรีมที่บรรจุข้อมูลเกี่ยวกับการซิงโครไนซ์ (synchronization) และข้อมูลสถานะ

error_check ส่วนของบิทสตรีมที่บรรจุข้อมูลสำหรับการตรวจสอบความผิดพลาด

audio_data ส่วนของบิทสตรีมที่บรรจุข้อมูลเกี่ยวกับข้อมูลเสียงที่สุ่มตัวอย่าง

ancillary_data ส่วนของบิทสตรีมที่บรรจุข้อมูลเพิ่มเติม

ก.3 ส่วนหัวของข้อมูล (Header)

32 บิทแรกของบิทสตรีมจะเป็นส่วนหัวของข้อมูล

syncword เป็นสตริง (string) เท่ากับ "1111 1111 1111"

ID 1 บิต แสดงถึงมาตรฐานการเข้ารหัส มีค่า "1" หมายถึงเข้ารหัสตามมาตรฐาน ISO/IEC11172-3 "0" หมายถึงไม่ใช่มาตรฐาน ISO

layer 2 บิต แสดงเลเยอร์ที่ใช้ ดังตารางที่ ก.1

ตารางที่ ก.1 ความหมายของรหัสข้อมูลใน **layer**

layer	ความหมาย
"00"	เลเยอร์ 1
"01"	เลเยอร์ 2
"10"	เลเยอร์ 3
"11"	ไม่ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

protection_bit 1 บิต บอกให้ทราบว่ามีการเพิ่มข้อมูลเกี่ยวกับการตรวจสอบความผิดพลาดมากับบิทสตรีมหรือไม่ โดย

"1" ไม่มีการเพิ่มข้อมูล

"0" มีการเพิ่มข้อมูล

bitrate_index แสดงอัตราข้อมูล (bitrate) ที่ใช้

sampling_frequency บอกความถี่ในการสุ่มตัวอย่าง

padding_bit ถ้าบิตนี้เป็น "1" แสดงว่าเฟรมของข้อมูลนั้น ๆ บรรจุสล็อต (slot) เพิ่มเติม ถ้าเป็น "0" แสดงว่าไม่ได้บรรจุสล็อตเพิ่มเติม

การบรรจุสล็อตเพิ่มเติมก็เพื่อปรับอัตราส่วนระหว่างอัตราการบีบอัดกับความถี่ในการสุ่มตัวอย่างให้ลงตัว

ในกรณีความถี่ในการสุ่มตัวอย่าง 44.1 กิโลเฮิร์ต ไม่มี padding คือเท่ากับ "0"เสมอ

private_bit ไม่ใช้ในการเข้ารหัสตามมาตรฐาน ISO/IEC11172-3

mode แสดงโหมดซึ่งเป็นไปตามตารางข้างล่าง

ตารางที่ ก.2 ความหมายของรหัสข้อมูลใน **mode**

mode	**ความหมาย**
"00"	สเตอริโอ
"01"	จอยท์-สเตอริโอ
"10"	สองช่องเสียง
"11"	หนึ่งช่องเสียง

mode_extension ใช้บอกชนิดของวิธีที่จอยท์-สเตอริโอใช้ ว่ามี ms_stereo และ intensity_stereo หรือไม่ อย่างไร ดังตารางที่ ก.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ ก.3 ความหมายของรหัสข้อมูลใน **mode_extension**

mode_extension	intensity	ms_stereo
"00"	off	off
"01"	on	off
"10"	off	on
"11"	on	on

copyright "1" หมายถึง ป้องกันการสำเนา "0" หมายถึง ไม่ป้องกันการสำเนา

original/copy "0" หมายถึง บิทสตรีมนั้นถูกสำเนามา "1" หมายถึง บิทสตรีมนั้นเป็นต้นฉบับ

emphasis แสดงถึงชนิดของเอ็มพาไซด์ (emphasis) ที่ใช้งาน

ตารางที่ ก.4 ความหมายของรหัสข้อมูลใน **emphasis**

Emphasis	ความหมาย
"00"	ไม่มีการใช้
"01"	50/15 ไมโครวินาที
"10"	สงวนไว้
"11"	CCITT J.17

ก.4 ส่วนตรวจสอบความผิดพลาด

crc_check ข้อมูล 16 บิต เพื่อตรวจสอบพาริตี้ของบิทสตรีมว่าถูกต้องหรือไม่

ก.5 ส่วนข้อมูลเสียง

main_data_begin แสดงตำแหน่งแรกของข้อมูลหลักในแต่ละเฟรม ซึ่งจะระบุตำแหน่งเป็นค่าออฟเซท (offset byte) เป็นค่าตำแหน่งที่ห่างออกมาจากไบต์แรกของซิงค์เวิร์ด (syncword) โดยไม่นับส่วนของหัวข้อมูลและข้อมูลข้างเคียง (side information)

private_bit จะไม่ใช้ในมาตรฐาน ISO/IEC

จำนวนของ private_bit จะขึ้นอยู่กับจำนวนช่องเสียง และจะนำไปใช้พิจารณาเพื่อเทียบกับจำนวนบิตของข้อมูลข้างเคียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

scfsi[ch][scfsi_band] ในการเข้ารหัสแบบเอ็มเป็ก 3 นั้น scfsi (scalefactor selection information) จะให้ข้อมูลเกี่ยวกับค่าสเกลแฟคเตอร์ (scalefactor) ของแต่ละย่านความถี่ (subband) และแต่ละช่องเสียง (ch)

ค่า scfsi_band ถูกใช้เพื่อเลือกกลุ่มของสเกลแฟคเตอร์ การใช้สเกลแฟคเตอร์ในแต่ละแกรนนูล (granule) จะถูกควบคุมโดย scfsi

ตารางที่ ก.5 ความหมายของ **scfsi[ch][scfsi_band]**

scfsi[ch][scfsi_band]	ความหมาย
"0"	เลือกใช้ในแต่ละแกรนนูลแยกกัน
"1"	ทั้ง 2 เลือกใช้สเกลแฟคเตอร์ตัวเดียวกัน

scfsi_band ควบคุมการใช้ scfsi สำหรับกลุ่มของสเกลแฟคเตอร์

ตารางที่ ก.6 ความหมายของรหัสข้อมูลใน **scfsi_band**

scfsi_band	ย่านความถี่ที่ถูกใช้งาน
0	0..5
1	6..10
2	11..15
3	16..20

part2_3_length[gr][ch] บอกจำนวนของบิทในส่วนที่เข้ารหัสแบบฮัฟแมน และสเกลแฟคเตอร์ เพื่อใช้หาตำแหน่งเริ่มต้นของข้อมูลหลักสำหรับแกรนนูลถัดไป

big_values[gr][ch] ค่าที่ถูกเข้ารหัสโดยใช้รหัสฮัฟแมน โดยอ้างค่าจากตารางฮัฟแมน

grobal_gain[gr][ch] เป็นตัวแปรที่ใช้ในขั้นตอนรีควอนไตซ์ (requantized)

scalefac_compress[gr][ch] เลือกจำนวนบิตที่ถูกใช้สำหรับส่งค่าสเกลแฟคเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ ก.7 ความหมายของรหัสข้อมูลใน **scalefac_compress[gr][ch]**

scalefac_compress[gr][ch]	slen1	slen2
0	0	0
1	0	1
2	0	2
3	0	3
4	3	0
5	1	1
6	1	2
7	1	3
8	2	1
9	2	2
10	2	3
11	3	1
12	3	2
13	3	3
14	4	2
15	4	3

block_type 0,1,3

slen : เป็นความยาวของสเกลแฟคเตอร์ย่าน 0 ถึง 10

slen : เป็นความยาวของสเกลแฟคเตอร์ย่าน 11 ถึง 20

block_type 2 และ **mixed_block_flag** เป็น 0

slen : เป็นความยาวของสเกลแฟคเตอร์ย่าน 0 ถึง 5

slen : เป็นความยาวของสเกลแฟคเตอร์ย่าน 6 ถึง 11

block_type 2 และ **mixed_block_flag** เป็น 1

slen : เป็นความยาวของสเกลแฟคเตอร์ย่าน 0 ถึง 7 (ย่าน long window)

เป็นความยาวของสเกลแฟคเตอร์ย่าน 3 ถึง 5 (ย่าน short window)

slen : เป็นความยาวของสเกลแฟคเตอร์ย่าน 6 ถึง 11

window_switching_flag[gr][ch] แสดงว่าข้อมูลไม่ได้ใช้วินโดว์ (window) ปกติ (type=0) ใน กรณีที่ window_switching_flag[gr][ch] = 1 จะมีผลดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

region_count = 7 กรณี **block_type**==1 หรือ **block_type**==3

หรือ **block_type**==2 และ **mixed_block_flag**

region_count = 8 **block_type**==2 และไม่ใช้ **mixed_block_flag**

region_count = 36 ค่าทั้งหมดของ **big_values** อยู่ใน **region1**

กรณีที่ window_switching_flag[gr][ch] = 0 แล้วค่า **block_type** จะเท่ากับศูนย์

block_type[gr][ch] ชี้ชนิดของวินโดว์ที่ใช้ของแกรนนูลนั้น

ตารางที่ ก.8 ความหมายของรหัสข้อมูลใน **block_type[gr][ch]**

Block_type[gr][ch]	ความหมาย
0	ไม่ใช้
1	เริ่มต้นบล็อค
2	วินโดว์บล็อคสั้น 3 บล็อก
3	ท้ายบล็อค

block_type และ **mixed_block_flag** ให้ข้อมูลเกี่ยวกับค่าที่อยู่ในบล็อค เกี่ยวกับความยาว และการนับสำหรับการแปลง ถ้า **window_switching_flag**==1 แล้ว **mixed_block_flag** จะเป็นตัวชี้ว่าโพลีเฟสด้านความถี่ต่ำย่านใดถูกถอดรหัสโดยใช้ type ปกติ

ในกรณีบล็อคยาว (block_type ไม่ใช่ 2 หรือย่านต่ำที่ block_type เป็น 2 เมื่อ mixed_block_flag =1) IMDCT จะให้เอาท์พุท 36 ค่าทุก ๆ อินพุท 18 ค่า เอาท์พุทจะขึ้นอยู่กับ block_type

ในกรณีบล็อคสั้น (ย่านที่เหนือกว่าบล็อกยาวของ block_type2 เมื่อ mixed_block_flag =0 หรือทุกย่านความถี่ของ block_type2 เมื่อ mixed_block_flag =0) IMDCT ให้เอาท์พุท 12 ค่า

mixed_block_flag[gr][ch] เป็นตัวกำหนดการแปลงที่ความถี่ต่ำว่าใช้ block_type แบบใด

กรณี **mixed_block_flag** เป็น 0 : ทุกบล็อคถูกแปลงค่าโดยชี้จาก **block_type[gr][ch]**

กรณี **mixed_block_flag** เป็น 1 : ย่านความถี่ต่ำสุด 2 ย่านถูกกำหนดให้แปลงด้วยวินโดว์ปกติ อีก 30 ย่านที่เหลือถูกแปลงค่าโดยชี้จาก **block_type[gr][ch]**

table_select[gr][ch][region] ใช้เลือกตารางฮัฟแมนจาก 32 ตาราง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

subblock_gain[gr][ch][window] บอกค่าอัตราการขยายที่เพิ่มขึ้น/ลดลงจาก **grobal_gain** ของแต่ละบล็อคย่อย

region0_count[gr][ch] ส่วนของสเปคตรัมที่ถูกแบ่งเพื่อใช้เพิ่มความสามารถของตัวเข้ารหัสให้มีความถูกต้องยิ่งขึ้น โดยแต่ละส่วนที่ถูกแบ่งจะเรียกว่า region0,1,2 แต่ละ region จะใช้ตารางถอดรหัสฮัฟแมนต่างกัน ขึ้นอยู่กับ maximum quantized value และ signal statistic ค่าของ region0_count และ region1_count ถูกใช้ชี้บอกขอบเขตของ region นั้น ๆ

region1_count[gr][ch] นับจำนวนของสเกลแฟคเตอร์ใน region1 ลบออก 1

preflag[gr][ch] เป็นค่าเสริมเพื่อขยายค่าที่ความถี่สูง ถ้าค่า preflag ถูกเซท จะนำค่าในตาราง preflag ไปคูณกับค่าสเกลแฟคเตอร์อีกครั้ง ในกรณี block_type=2 preflag ไม่ถูกใช้

scalefac_scal[gr][ch] เป็นค่าสเกลแฟคเตอร์ที่ถูกปรับค่าด้วยสเกลล็อกการิทึม โดยคูณค่าของแต่ละลำดับด้วย 2 หรือ $\sqrt{2}$ ขึ้นอยู่กับ **scalefac_scal**

ตารางที่ ก.9 ความหมายของข้อมูลใน **scalefac_scal[gr][ch]**

Scalefac_scal[gr][ch]	scalefac_multiplier
0	0,5
1	1

count1table_select[gr][ch] ใช้เลือกค่าจากตารางฮัฟแมน B7.A หรือ B7.Bเมื่อค่าที่ปรับระดับใน region คูณด้วย 4 แล้วไม่เกิน 1

ตารางที่ ก.10 ความหมายของรหัสข้อมูลใน **count1table_select[gr][ch]**

Count1table_select[gr][ch]	ความหมาย
0	ตาราง B7.A
1	ตาราง B7.B

scalfac_l[gr][ch], scalefac_s[gr][ch][sfb][window], is_pos[gr][ch] ถูกใช้ในการปรับระดับค่าคืน (requantization)

huffmancodebits0 ข้อมูลที่ถูกเข้ารหัสแบบฮัฟแมน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบของ huffmancodebits0 แสดงวิธีการเข้ารหัสข้อมูลลงใน **big_value** โดยข้อมูลที่ได้จะเป็นคู่ (x,y) ให้ค่าแต่ละค่ามีจำนวนบิตน้อยกว่า 15 บิต การเข้ารหัสจะถูกเลือกจากตารางฮัฟแมนที่ 0 ถึง 31 ถ้าข้อมูลที่เข้ามามีขนาดเกิน 15 บิต จะแยกเข้ารหัสต่างหาก ถ้าในคู่ของข้อมูลที่เข้ารหัสไม่เป็น 0 จะปรากฏเครื่องหมาย ให้ค่า + หรือ – ในตารางฮัฟแมนจะบรรจุองค์ประกอบ 3 ส่วน คือ

- hcod[x][y] รหัสข้อมูลฮัฟแมน
- hlen[x][y] ความยาวข้อมูลฮัฟแมน
- linbits ความยาวของ linbitsx หรือ linbitsy เมื่อถูกเข้ารหัส

รูปแบบของส่วนประกอบของ **huffmancodebits** ประกอบด้วยกลุ่มของข้อมูลประกอบดังนี้

- **sig nv** เครื่องหมายของค่า v (0 เป็นบวก 1 เป็นลบ)
- **sig nw** เครื่องหมายของค่า w (0 เป็นบวก 1 เป็นลบ)
- **sig nx** เครื่องหมายของค่า x (0 เป็นบวก 1 เป็นลบ)
- **sig ny** เครื่องหมายของค่า y (0 เป็นบวก 1 เป็นลบ)
- **linbitsx** ใช้เมื่อเข้ารหัสค่าของ x มากกว่าหรือเท่ากับ 15
- **linbitsy** ใช้เมื่อเข้ารหัสค่าของ y มากกว่าหรือเท่ากับ 15
- **is[1]** คือ ค่าที่ถูกปรับระดับสำหรับ frequency line ที่ 1

ก.6 Ancillary Data

ancillary_bit ผู้ใช้สามารถนิยามได้เอง

- หมายเหตุ สามารถหารายละเอียดมาตราฐานการบีบอัดข้อมูลเพิ่มเติมจาก ห้องสมุดสำนักงานมาตราฐานอุตสาหกรรม (สมอ.) หัวข้อ ISO 11172-3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข
รายละเอียดของคำสั่งพิเศษ

Run

S	dev_write	A	data_write	A	a3,a2	A	a1,a0	A	P

คำสั่ง run เป็นคำสั่งที่สั่งให้ชิปเริ่มทำงาน ณ ตำแหน่งของโปรแกรม a=(a3,a2,a1,a0) โดย a3 จะเป็นตัวกำหนดรูปแบบการทำงานตามตารางที่ 3.10 ซึ่งมีค่าได้ 2 ค่า คือ $1, $0

Read Control Interface Data

1) ส่งคำสั่ง

S	dev_write	A	data_write	A	S	$3,x2	A	x1,x0	A	P

2) รับข้อมูล

S	dev_write	A	data_read	A	S	dev_read		
	(ancillary word 0)				A	d3,d2	A	d1,d0
					....repeat for n data values...			
					A	d3,d2	A	d1,d0	A	P

x2...x0 : combined count, offset value

d3...d0 : 16-bit data values

คำสั่ง read control interface data เป็นคำสั่งที่ใช้ในการอ่านสถานะของการถอดรหัส (ดูได้จากตาราง 3-9) และยังสามารถเข้าถึงตำแหน่งของหน่วยความจำได้อย่างรวดเร็วโดยใช้ x2, x1 และ x0 เป็นตัวบอกตำแหน่งเริ่มต้นและจำนวนข้อมูลที่จะอ่าน

Write Register

S	dev_write	A	data_write	A	$9,r1	A	r0,d0	A	
					d4,d3	A	d2,d1	A	P

เป็นการเขียนข้อมูลจำนวน 20 บิต (d=d4,d3,d2,d1,d0) ลงในรีจิสเตอร์ (r=r1,r0)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Write D0 Memory

S	dev_write	A	data_write	A	$A,$0	A	$0,$0
				A	n3,n2	A	n1,n2
				A	a3,a2	A	a1,a0
				A	d3,d2	A	d1,d0
				A	$0,$1	A	$0,$4

...repeat for n data values...

A	d3,d2	A	d1,d0		
A	$0,$0	A	$0,$4	A	P

n3...n0 : number of word

a3...a0 : start address in MASD memory

d4...d0 : data value

เป็นคำสั่งในการเขียนค่าลงในหน่วยความจำ D0

Write D1 Memory

S	dev_write	A	data_write	A	$B,$0	A	$0,$0
				A	n3,n2	A	n1,n2
				A	a3,a2	A	a1,a0
				A	d3,d2	A	d1,d0
				A	$0,$1	A	$0,$4

...repeat for n data values...

A	d3,d2	A	d1,d0		
A	$0,$0	A	$0,$4	A	P

n3...n0 : number of word

a3...a0 : start address in MASD memory

d4...d0 : data value

เป็นคำสั่งในการเขียนค่าลงในหน่วยความจำ D1

Read Register

1) ส่งคำสั่ง

S	dev_write	A	data_write	A	$D,r1	A	r0,$0	A	P

2) รับค่าในรีจิสเตอร์

S	dev_write	A	data_read	A	S	dev_read			
A	d3,d2	A	d1,d0	A	X,X	A	X,d4	N	

n3...n0 : number of words to transmitted

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

a3…a0 : start address in MASD memory

d4…d0 : data value

เป็นการอ่านข้อมูลจำนวน 20 บิต ที่อยู่ในรีจิสเตอร์ (r=r1,r0)

Read D0 Memory

1) ส่งคำสั่ง

S	dev_write	A	data_write	A	$E,$0	A	$0,$0				
				A	n3,n2	A	n1,n0				
				A	a3,a2	A	a1,a0	A	P		

2) รับค่าจากหน่วยความจำ

S	dev_write	A	data_read	A	S	dev_read					
A	d3,d2	A	d1,d0	A	$0,$0	A	$0,d4				
…repeat for n data values…											
A	d3,d2	A	d1,d0	A	$0,$0	A	$0,d4	A	P		

n3…n0 : number of words to transmitted

a3…a0 : start address in MASD memory

d4…d0 : data value

เป็นการอ่านข้อมูลที่อยู่ในหน่วยความจำ D0

Read D1 Memory

1) ส่งคำสั่ง

S	dev_write	A	data_write	A	$F,$0	A	$0,$0				
				A	n3,n2	A	n1,n0				
				A	a3,a2	A	a1,a0	A	P		

2) รับค่าจากหน่วยความจำ

S	dev_write	A	data_read	A	S	dev_read					
A	d3,d2	A	d1,d0	A	$0,$0	A	$0,d4				
…repeat for n data values…											
A	d3,d2	A	d1,d0	A	$0,$0	A	$0,d4	A	P		

n3…n0 : number of words to transmitted

a3…a0 : start address in MASD memory

d4…d0 : data value

เป็นการอ่านข้อมูลที่อยู่ในหน่วยความจำ D1

- หมายเหตุ สามารถหารายละเอียดเพิ่มเติมได้จากบริษัท Micronas Intermetall

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค

รีจิสเตอร์ภายในตามมาตรฐาน ATAPI และรายละเอียด

การติดต่อสื่อสารจาก/ไปยังอุปกรณ์จะกระทำโดยผ่านรีจิสเตอร์ I/O ซึ่งการกำหนดเส้นทางข้อมูลอินพุทหรือเอาท์พุทจาก/ไปยังรีจิสเตอร์ (ที่ถูกเลือก) จะทำได้โดยการถอดรหัสสัญญาณจากโฮสท์ (CS1FX-, CS3FX-, DA2, DA1, DA0, DIOR- และ DIOW-)

ตารางที่ ค1 รีจิสเตอร์ภายในตามมาตรฐาน ATAPI

แอดเดรส					หน้าที่	
CS1FX	CS3FX	DA2	DA1	DA0	อ่าน (DIOR-)	เขียน(DIOW-)
					รีจิสเตอร์ควบคุมบล็อค	
N	A	0	0	0	สถานะ Floppy A	ไม่ใช้
N	A	0	0	1	สถานะ Floppy A	ไม่ใช้
N	A	0	1	0	ไม่ใช้	รีจิสเตอร์ Floppy Digital Output
N	A	0	1	1	ควบคุม Floppy ID/เทป	สงวน
N	A	1	0	0	สถานะ Floppy Controller	สงวน
N	A	1	0	1	รีจิสเตอร์ Floppy Data	
N	A	1	1	0	สถานะ Alternate ATAPI	ควบคุมอุปกรณ์
N	A	1	1	1	(หมายเหตุ1)	ไม่ใช้
					รีจิสเตอร์บล็อคคำสั่ง	
A	N	0	0	0	ข้อมูล	
A	N	0	0	1	รีจิสเตอร์ Error	รีจิสเตอร์ Feature
A	N	0	1	0	รีจิสเตอร์ Interrupt Reason	ไม่ใช้
A	N	0	1	1	สงวนสำหรับ SAM TAG Byte	
A	N	1	0	0	รีจิสเตอร์ Byte Count (บิต0-7)	
A	N	1	0	1	รีจิสเตอร์ Byte Count (บิต8-15)	
A	N	1	1	0	Drive Select	
A	N	1	1	1	สถานะ ATAPI	คำสั่ง ATA

หมายเหตุ1 : รีจิสเตอร์นี้ได้ยกเลิกการใช้ไปแล้ว แนะนำว่าอุปกรณ์ไม่ควรที่จะตอบสนองต่อการอ่านค่าที่แอดเดรสนี้ได้ ถ้าอุปกรณ์ไม่ตอบสนองแล้ว อุปกรณ์ก็ควรจะไม่ขับสัญญาณ DDF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค.1 รีจิสเตอร์สถานะของ ATAPI (รีจิสเตอร์สถานะของ ATA)

D7	D6	D5	D4	D3	D2	D1	D0
BSY	DRDY	DMA READY Or DF	SERVICE or DSC	DRQ	CORR	Reversed	CHECK

DRDY, DSC, CORR และ CHECK จะสามารถนำไปใช้ได้เมื่อสิ้นสุดคำสั่งแล้ว

บิต 7 BSY (Busy) จะถูกเซทเมื่ออุปกรณ์ทำการเข้าถึงบล็อคคำสั่ง

บิต 6 DRDY (Device Ready) แสดงว่าอุปกรณ์สามารถตอบสนองต่อคำสั่ง ATA ได้

บิต 5 DMA READY/DF บิตนี้แสดงว่าอุปกรณ์พร้อมที่จะเริ่มการถ่ายโอนข้อมูลแบบ DMA บิตนี้จะสงวนไว้สำหรับความสามารถในการโอเวอร์แลป (overlap) ที่จะพัฒนาในอนาคต

เมื่อไม่อนุญาตให้มีการโอเวอร์แลปหรือ Interleave DMA บิตนี้จะใช้สำหรับ Drive Fault (DF)

บิต 4 SERVICE/DSC บิตนี้จะเป็นใช้สัญญาณว่าอุปกรณ์กำลังต้องการการบริการหรืออินเตอร์รัพ บิตนี้จะถูกเซทเมื่อมีการขออินเตอร์รัพ และบิตนี้จะไม่เคลียร์จนกว่าจะมีการส่งคำสั่ง Service (A2h) ออกมา

เมื่อไม่อนุญาตให้มีการอินเตอร์รัพ บิตนี้จะใช้สำหรับฟังก์ชั่น DSC

บิต 3 DRQ (Data Request) แสดงว่า อุปกรณ์พร้อมที่จะถ่ายโอนไบต์หนึ่งหรือเวิร์ดหนึ่งของข้อมูลระหว่างโฮสท์และ Drive

เมื่อบิต DRQ ถูกเซท ข้อมูลที่อยู่ในรีจิสเตอร์ Interrupt Reason ของ ATAPI จะสามารถนำไปใช้ได้

บิต 2 CORR (Corrected Data) ใช้แสดง Correctable Error

บิต 0 CHECK แสดงการเกิดความผิดพลาดในระหว่างการปฏิบัติการของคำสั่งก่อนหน้านี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค.2 รีจิสเตอร์ Error ของ ATAPI (รีจิสเตอร์ Error ของ ATA)

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

บิต 7-4 Sense Key

บิต 3 MCR (Media Change Request) สงวนไว้สำหรับใช้กับ media device ที่สามารถเปลี่ยนได้ และใช้แสดงว่า อุปกรณ์ ได้ตรวจพบการร้องขอในการเปลี่ยน media

บิต 2 ABRT (Aborted Command) แสดงว่าคำสั่งที่ร้องขอมานั้นล้มเหลว เนื่องจากรหัสคำสั่งไม่ถูกต้องหรือ อุปกรณ์ อื่นเกิดความผิดพลาดขึ้น

บิต 1 EOM (End of Media) แสดงการตรวจพบตำแหน่งสุดท้ายของมิเดีย (media)

บิต 0 ILI (Illegal Length Indication)

ค.3 รีจิสเตอร์ Feature ของ ATAPI (รีจิสเตอร์ Feature ของ ATA)

D7	D6	D5	D4	D3	D2	D1	D0
Reserved						OVER LAP	DMA

บิต 1 OVERLAP (Optional) อุปกรณ์ อาจจะ release บัส ATA ก่อนที่คำสั่งจะเสร็จสิ้นสมบูรณ์

บิต 0 DMA (Optional) ข้อมูลสำหรับคำสั่งที่จะถูกถ่ายโอนผ่านการเชื่อมต่อแบบ DMA สังเกตว่าจะไม่มีการใช้คอมมานด์แพ็คเกต

ค.4 รีจิสเตอร์ Byte Count ของ ATAPI (รีจิสเตอร์ Cylinder High/Low ของ ATA)

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Byte Count จะใช้สำหรับ PIO เท่านั้น และจะต้องตั้งค่าก่อนที่จะส่งแพ็คเกตคอมมานด์ออกไป Byte Count จะบรรจุปริมาณข้อมูลในการถ่ายโอนทั้งหมดสำหรับคำสั่งที่ถ่ายโอนข้อมูลเพียงชุดเดียว (เช่น Mode Sense / Select, Inquiry) ส่วนคำสั่งที่ต้องการการอินเตอร์รัพ DRQ หลายครั้ง (เช่น Read หรือ Write) จะต้องตั้งค่า Byte Count ให้เป็นขนาดการถ่ายโอนที่ต้องการ เมื่อข้อมูลถูกถ่ายโอน ATAPI อุปกรณ์ จะตั้งค่า Byte Count ให้เป็นปริมาณข้อมูลที่โฮสท์จะต้องถ่ายโอน จากนั้นจะส่ง DRQ อินเตอร์รัพ ข้อมูลในรีจิสเตอร์นี้จะต้องไม่เปลี่ยนจนกระทั่งเวิร์ดแรกได้ถูกถ่ายโอนจาก/ไปยังรีจิสเตอร์ข้อมูลแล้ว

ค.5 รีจิสเตอร์ Interrupt Reason ของ ATAPI (รีจิสเตอร์ Sector Count ของ ATA)

D7	D6	D5	D4	D3	D2	D1	D0
Reserved					RELEAS E	IO	CoD

บิต 0 CoD (Command or Data)

เมื่อ CoD=0 แสดงว่าข่าวสารที่ถูกถ่ายโอนนั้นคือข้อมูล

เมื่อ CoD=1 แสดงว่าข่าวสารที่ถูกถ่ายโอนนั้นคือคำสั่ง

บิต 1 IO ทิศทางการถ่ายโอนข่าวสาร

เมื่อ IO=0 แสดงถึงทิศทางไปยัง อุปกรณ์

เมื่อ IO=1 แสดงถึงทิศทางไปยังโฮสท์

IO	DRQ	CoD	
0	1	1	คำสั่ง - พร้อมที่จะรับไบต์แพ็คเกตคอมมานด์
1	1	1	สาร (อนาคต) - พร้อมที่จะส่งข้อมูลไปยังโฮสท์
1	1	0	ข้อมูลไปโฮสท์ - ส่งข้อมูลพารามิเตอร์คำสั่ง (เช่น Read Data) ไปยังโฮสท์
0	1	0	ข้อมูลจากโฮสท์ - รับข้อมูลพารามิเตอร์คำสั่ง (เช่น Write Data) จากโฮสท์
1	0	1	สถานะ - รีจิสเตอร์จะบรรจุ Completion Status

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บิต 2 RELEASE แสดงว่า อุปกรณ์ ได้ release บัส ATA ก่อนการเสร็จสิ้นสมบูรณ์ของคำสั่งที่กำลังดำเนินการอยู่

ค.6 รีจิสเตอร์ Drive Select ของ ATAPI (รีจิสเตอร์ Drive / Head ของ ATA)

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

บิต 4 DRV บิตนี้จะเป็นการเลือก อุปกรณ์0 (DRV=0) หรือไม่ก็ อุปกรณ์1 (DRV=1)

ค.7 รีจิสเตอร์ อุปกรณ์ Control ของ ATAPI (รีจิสเตอร์ อุปกรณ์ Control ของ ATA)

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

บิต 2 SRST (Software Reset) เป็นบิตการรีเซททางซอฟท์แวร์

บิต 1 nIEN เป็นบิตอนุญาต/ไม่อนุญาตการอินเตอร์รัพของอุปกรณ์มายังโฮสท์

เมื่อ nIEN=0 และอุปกรณ์ได้ถูกเลือก แล้วจะอนุญาตสัญญาณ INTRQ ผ่านบัฟเฟอร์ 3 สถานะ

เมื่อ nIEN=1 หรืออุปกรณ์ไม่ได้ถูกเลือก แล้วสัญญาณ INTRQ จะอยู่ในสถานะอิม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ง
ตัวอย่าง File Systems ที่อ่านได้จากแผ่น CD-ROM

ตารางที่ ง.1 แสดงตัวอย่าง Volume Descriptor ที่อ่านมาได้

ค่าที่อ่านได้จาก CD-ROM	Field name	ความหมาย
01	Volume Descriptor Type	Primary Volume Descriptor
4344303031	Standard Identifier	CD001
01	Volume Descriptor Version	1
00	Unused Field	
20202020…20	System Identifier	
707432342e3120202020202020202020202020202020202020202020202020202020	Volume Identifier	Pt24.1
0000000000000000	Unused Field	
6011050000051160	Volume Space Size	85005920
0000000…00	Unused Field	
01000001	Volume Set Size	
01000001	Volume Sequence Number	
00080800	Logical Block Size	2048
aa010000000001aa	Path Table Size	426
15000000	Location of Occurrence of Type L Path Table	
00000000	Location of Optional Occurrence of Type L Path Table	
00000014	Location of Occurrence of Type M Path Table	
00000000	Location of Optional	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	Occurrence of Type M Path Table	
22001600000000000016000 80000000008000000000000 00000200000100000010100	Directory Record of Root Directory	
2020202020…20	Volume Set Identifier	
2020202020…20	Publisher Set Identifier	
2020202020…20	Data Preparer Identifier	
2020202020…20	Application Identifier	
2020202020…20	Copyright File Identifier	
2020202020…20	Abstract File Identifier	
2020202020…20	Bibliographic File Identifier	
31 39 39 39 30 31 32 38 30 33 32 32 35 35 30 30 00	Volume Creation Date and time	1999 01 28 03 22 55 00
31 39 39 39 30 31 32 38 30 33 32 32 35 35 30 30 00	Volume Modification Date and Time	1999 01 28 03 22 55 00
32303039303132353033323 23535303000	Volume Expiration Date and Time	1999 01 28 03 22 55 00
31 39 39 39 30 31 32 38 30 33 32 32 35 35 30 30 32	Volume Effective Date and Time	1999 01 28 03 22 55 00 2
00	File Structure Version	
000000…00	Reserved for future Standardization	
000000…00	Application Use	
000000…00	Reserved for future Standardization	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ ง.2 แสดงตัวอย่างค่าที่อ่านมาได้จาก Path Table ของ CD-ROM และความหมาย

ค่าที่อ่านได้จากCD-ROM	ความหมายของค่าที่ได้
05	ความยาวชื่อเท่ากับ 5 ตัวอักษร
00	ไม่มีความหมาย
0001f0f4	ตำแหน่งของข้อมูล
0001	Folder
42415a4f4f00	BAZOO
18	ความยาวชื่อเท่ากับ 24 ตัวอักษร
00	ไม่มีความหมาย
000071d4	ตำแหน่งของข้อมูล
0001	Folder
4249524420b8a7e4aac22053455256494 34520bed4e0c8c9	BIRD ธงไชย service พิเศษ
14	ความยาวชื่อเท่ากับ 20 ตัวอักษร
00	ไม่มีความหมาย
000093f7	ตำแหน่งของข้อมูลบนแผ่น
0001	Folder
434852495354494e412d355448204156 454e5545	CHRISTINA-5TH AVENUE
0c	ความยาวชื่อเท่ากับ 12 ตัวอักษร
00	ไม่มีความหมาย
0000e3bd	ตำแหน่งของข้อมูลบนแผ่น
0001	Folder
464c592de1c1c5a7e0bec5a7	FLY แมลงเพลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ ง.3 แสดงตัวอย่าง Directory Structure ที่อ่านมาได้จาก CD-ROM

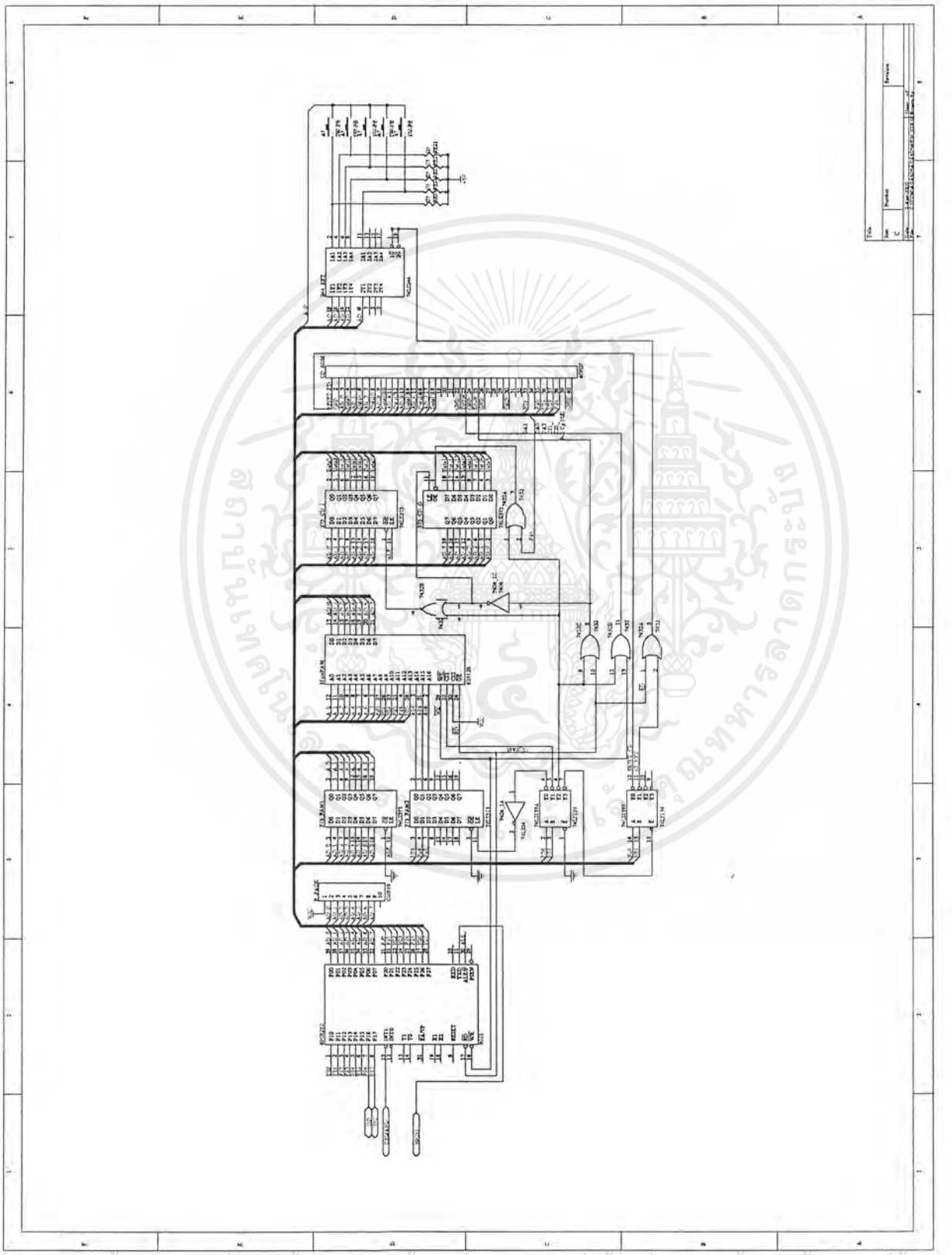
ค่าที่อ่านได้จาก CD-ROM	ความหมายของค่าที่ได้
42	จำนวนข้อมูล
00	ไม่มีความหมาย
d5710000000071d5	ข้อมูลอยู่ที่ตำแหน่ง 71d5h
3215360000361532	ข้อมูลมีขนาด 3,544,370 ไบต์
620c0e0b1a0e00	
00000001000001	
21	ความยาวชื่อเท่ากับ 33 ตัวอักษร
424952442dbed4e0c8c92d30312dabe8cdc1abd0e3cbe	BIRD-พิเศษ-01-ซ่อมซะให้เข็ด.MP3
9e0a2e7b42e4d50333b31	
3e	จำนวนข้อมูล
00	ไม่มีความหมาย
9878000000007898	ข้อมูลอยู่ที่ตำแหน่ง 7898h
4a1a2f00002f1a4a	ข้อมูลมีขนาด 3,086,922 ไบต์
620c0e0b1d0a00	
00000001000001	
1c	ความยาวชื่อเท่ากับ 28 ตัวอักษร
424952442dbed4e0c8c92d30322dbed9b4e0c5e8b9e62	BIRD-พิเศษ-02-พูดเล่นๆ.MP3
e4d50333b3100	
46	จำนวนข้อมูล
00	ไม่มีความหมาย
7c7e000000007e7c	ข้อมูลอยู่ที่ตำแหน่ง 7e7ch
72c23a00003ac272	ข้อมูลมีขนาด 3,850,866 ไบต์
620c0e0b203a00	
00000001000001	
24	ความยาวชื่อเท่ากับ 36 ตัวอักษร
424952442dbed4e0c8c92d30332da1e9cdb9cbd4b9a1d	BIRD-พิเศษ-03-ก้อนหินกับ
1bab9d2ccd4a1d22e4d50333b3100	นาฬิกา.MP3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก จ

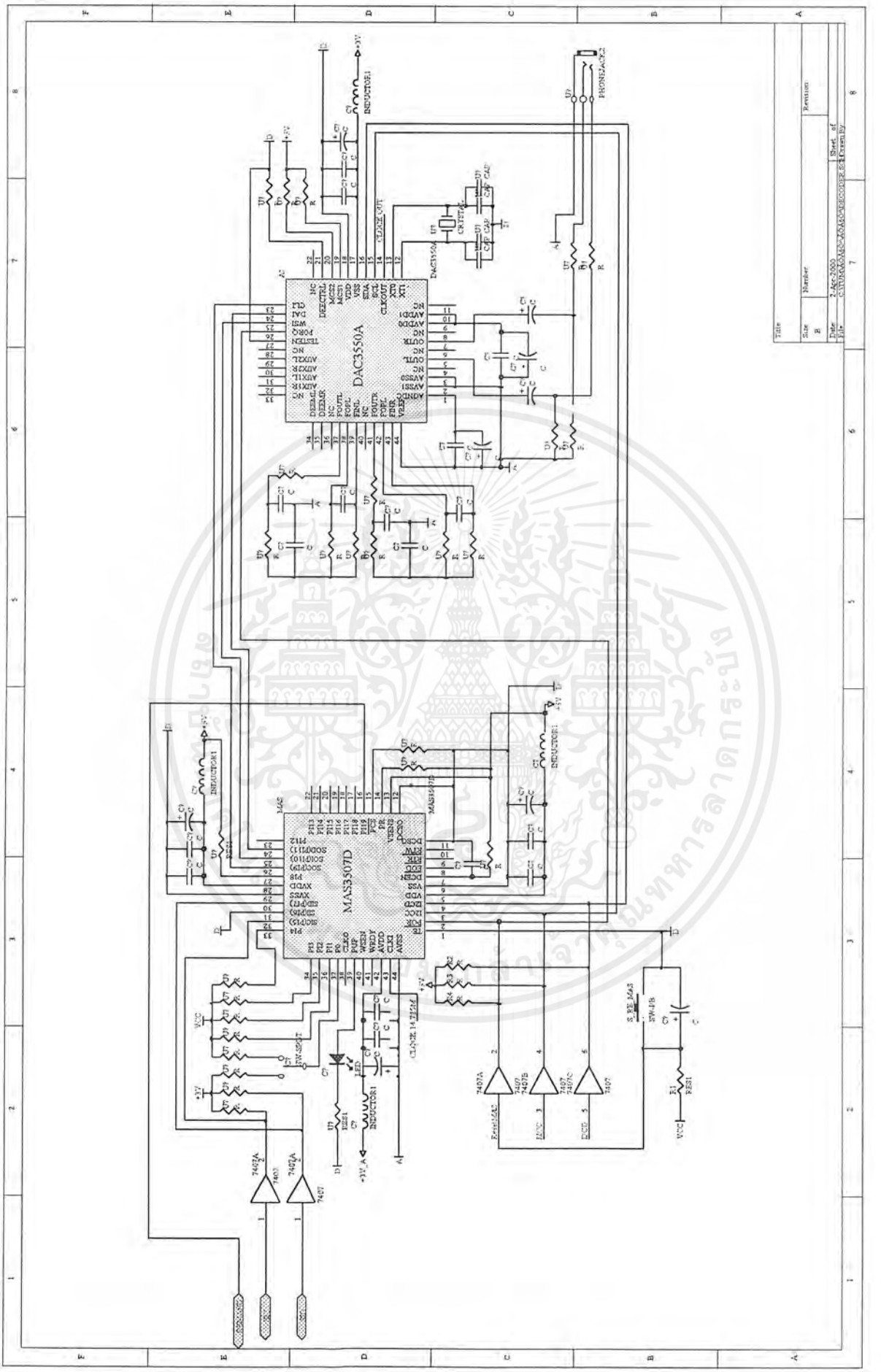
รูปวงจร

รูปที่ จ.1 รูปวงจรส่วนควบคุมการอ่าน CD-ROM



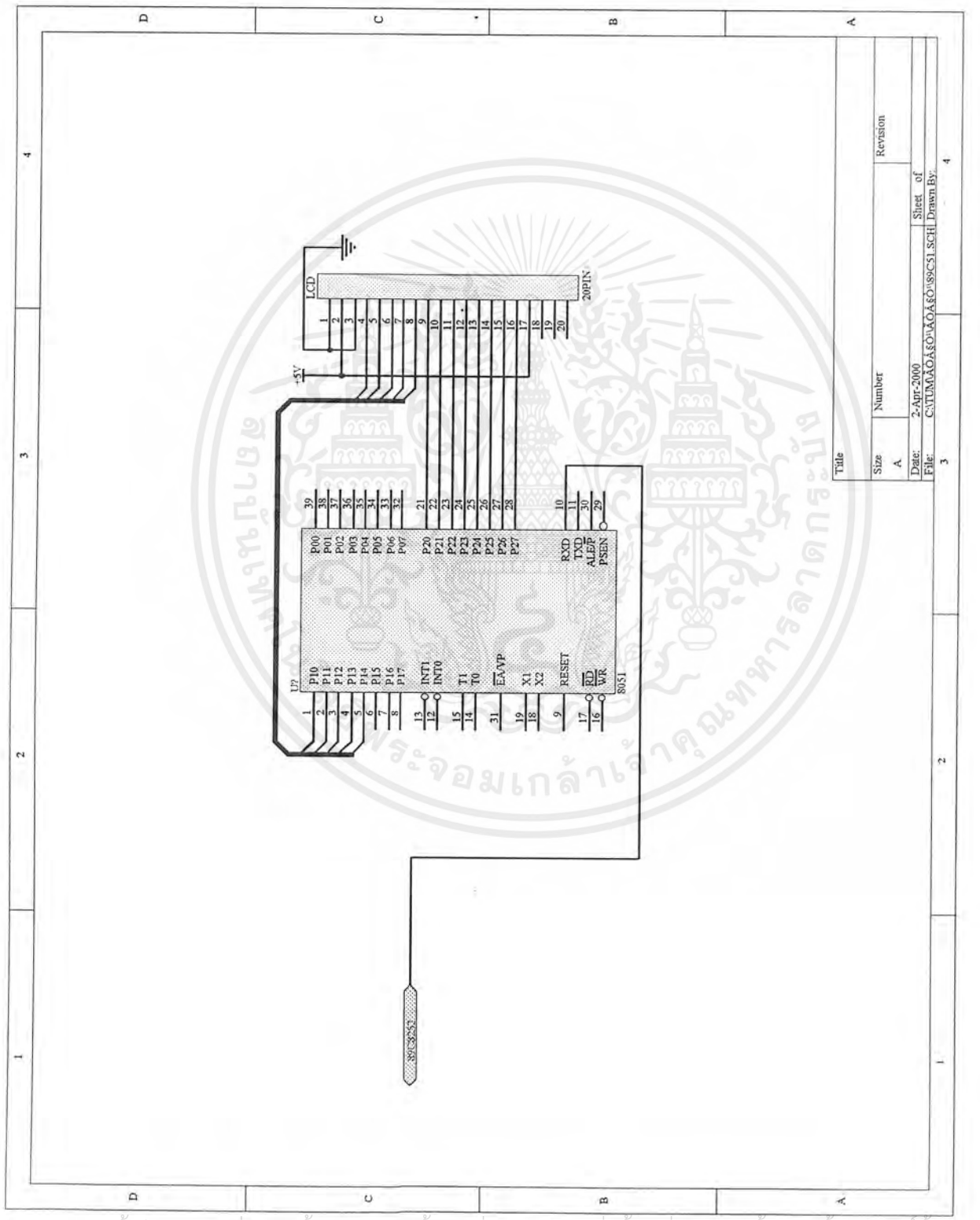
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ จ.2 รูปวงจรถอดรหัสสัญญาณ MP3



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ จ.3 รูปวงจรส่วนแสดงผลบนจอ LCD



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้