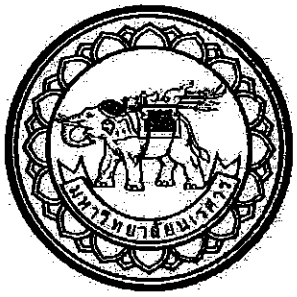


คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร
3 1001 00382400 1

การออกแบบและพัฒนาเครื่องเล่นเอ็มพีสาม

Development and Design of MP3 Player

นางสาวณัฐกาญจน์ มีสุขโข รหัส 48380248
นางสาวสุรีรัตน์ จันทร์เมือง รหัส 48380367

12 ก.ย. 2556
เลขทะเบียน 163814 92
เลขเรียกหนังสือ ฝ5.
มหาวิทยาลัยนเรศวร ส4322 9
2551

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร
ปีการศึกษา 2551

ใบรับรองปริญญานิพนธ์

หัวข้อโครงงาน	การออกแบบและพัฒนาเครื่องเล่นเอ็มพีสาม
ผู้ดำเนินโครงงาน	นางสาวณัฐภาญจน์ มีสุขโข รหัส 48380248
	นางสาวสุรีรัตน์ จันทร์เมือง รหัส 48380367
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร. สุชาติ แย้มเม่น
สาขาวิชา	วิศวกรรมไฟฟ้า
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2551

.....

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมไฟฟ้า

.....ที่ปรึกษาโครงงาน
(ผู้ช่วยศาสตราจารย์ ดร. สุชาติ แย้มเม่น)

.....กรรมการ
(รองศาสตราจารย์ ดร. ไพศาล มุณีสว่าง)

.....กรรมการ
(ดร. แดทรียา สุวรรณศรี)

หัวข้อโครงงาน	การออกแบบและพัฒนาเครื่องเล่นเอ็มพีสาม
ผู้ดำเนินโครงงาน	นางสาวณัฐกาญจน์ มีสุขโข รหัส 48380248
	นางสาวสุรีรัตน์ จันทร์เมือง รหัส 48380367
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร. สุชาติ แย้มเม่น
สาขาวิชา	วิศวกรรมไฟฟ้า
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2551

บทคัดย่อ

โครงงานนี้เป็นการออกแบบและสร้างเครื่องเล่นเอ็มพีสาม (MP3) กับไฟล์เพลงที่บันทึกลงในเอสดีการ์ด (SD CARD) หรือไฟล์เพลงที่บันทึกลงในยูเอสบีแฟลชไดรฟ์ (USB FLASH DRIVE) ที่มีขนาดเล็ก มีความจุระหว่าง 64 MB ถึง 2 GB โดยใช้ไมโครคอนโทรเลอร์ตระกูลเอวีอาร์ดึงข้อมูลจากเอสดีการ์ดหรือยูเอสบี และใช้อุปกรณ์ถอดรหัสเป็นตัวแปลงไฟล์เสียงในรูปแบบสัญญาณดิจิตอลให้เป็นเสียงเพื่อให้สามารถรับฟังได้

จากการทดลองใช้งานเครื่องที่พัฒนาขึ้นเมื่อนำข้อมูลไฟล์เพลงเอ็มพีสามบันทึกข้อมูลลงในสื่อบันทึกข้อมูลเอสดีการ์ดและสื่อบันทึกข้อมูลยูเอสบี เครื่องเล่นเพลงเอ็มพีสามที่พัฒนาขึ้นสามารถรับฟังได้ที่บิตเรท 128 Kbps และ 192 Kbps และสามารถทำงานได้ครบตามฟังก์ชั่นที่ได้ออกแบบไว้ทุกประการ แต่อย่างไรก็ตามไม่สามารถรับฟังได้ที่บิตเรท 320 Kbps เนื่องจากมีเสียงรบกวนและเสียงยาน

Project Title	Development and Design of MP3 Player
Name	Miss. Nutthakarn Meesukkho ID. 48380248
	Miss. Sureerat Janmuang ID. 48380367
Project Advisor	Assistant Professor Suchart Yammen,Ph.D.
Major	Electrical Engineering
Department	Electrical and Computer Engineering
Academic Year	2012

ABSTRACT

This project is to design and make an MP3 Player with the files that could be saved in an SD Card or a mini USB Flash drive, capacity between 64 MB and 2 GB. The AVR micro controller is used to process the files stored in the SD Card or the mini USB Flash drive and also as a decoder device to transfer digital data to a sound.

From a test, it is shown that the device can perform its function successfully. It can play songs at 128 Kbps and 192 Kbps but not at 320 Kbps for there will be noise and sound will be distorted.

กิตติกรรมประกาศ

โครงงานนี้สำเร็จลุล่วงไปได้ด้วยดี ด้วยความกรุณาจาก ผู้ช่วยศาสตราจารย์ ดร.สุชาติ แย้มเม่น ซึ่งเป็นอาจารย์ที่ปรึกษาโครงงานและให้ความกรุณาในการตรวจทาน ปริญญานิพนธ์ คณะผู้ดำเนินโครงงานขอกราบขอบพระคุณเป็นอย่างสูงและขอระลึกถึงความกรุณาของท่านไว้ตลอดไป

ขอขอบพระคุณ ศาสตราจารย์ ดร. ไพศาล มุณีสว่าง และ ดร.แคทรียา สุวรรณศรี ซึ่งเป็นคณะกรรมการในการสอบโครงงานที่ให้คำแนะนำ ชี้แนะแนวทาง และข้อคิดเห็นต่างๆที่เป็นประโยชน์ในโครงงานนี้ ทำให้โครงงานออกมาสมบูรณ์แบบยิ่งขึ้น

ขอขอบคุณ ดร. พรพิศุทธิ์ วรจิรันตน์ และ อาจารย์แสงชัย มังกรทอง ที่ให้คำแนะนำเกี่ยวกับการออกแบบวงจรอิเล็กทรอนิกส์และการเขียนโปรแกรมไมโครคอนโทรลเลอร์ที่เป็นประโยชน์ต่อโครงงานนี้รวมถึงความรู้ใหม่ๆ

เหนือสิ่งอื่นใด คณะผู้จัดทำโครงงานขอกราบขอบพระคุณคุณพ่อ คุณแม่ ผู้มอบความรัก ความเมตตา สติปัญญา เป็นที่ปรึกษาปัญหาในทุกๆ เรื่อง รวมทั้งเป็นผู้ให้ทุกสิ่งทุกอย่างตั้งแต่วัยเยาว์จวบจนถึงปัจจุบัน คอยเป็นกำลังใจทำให้ได้รับความสำเร็จอย่างทุกวันนี้และขอขอบคุณทุกๆ คนในครอบครัวของคณะผู้ดำเนินโครงงานที่ไม่ได้กล่าวไว้ ณ ที่นี้ด้วย

นางสาวณัฐกาญจน์ มีสุขโข รหัส 48380248
นางสาวสุรีรัตน์ จันทร์เมือง รหัส 48380367

สารบัญ

หน้า

ใบรับรองปริญญานิพนธ์ ..... ก

บทคัดย่อภาษาไทย ..... ข

บทคัดย่อภาษาอังกฤษ ..... ค

กิตติกรรมประกาศ ..... ง

สารบัญ ..... จ

สารบัญตาราง ..... ช

สารบัญรูป ..... ฌ

บทที่ 1 บทนำ 1

1.1 ความเป็นมาและความสำคัญของโครงงาน 1

1.2 วัตถุประสงค์ของโครงงาน 1

1.3 ขอบเขตของโครงงาน 1

1.4 ขั้นตอนดำเนินงาน 1

1.5 แผนการดำเนินงาน 2

1.6 ประโยชน์ที่คาดว่าจะได้รับ 2

1.7 งบประมาณ 3

บทที่ 2 หลักการและทฤษฎีพื้นฐาน 4

2.1 โครงสร้างและหลักการทำงานของ SD CARD 4

2.1.1 รายละเอียดขาสัญญาณของ SD CARD 5

2.2 USB (Universal Serial Bus) 6

2.3 การทำงานของไมโครคอนโทรเลอร์ 7

2.3.1 โครงสร้างทั่วไป 7

2.3.2 ไมโครคอนโทรเลอร์ AVR ATmega128 8

2.4 การบีบอัดข้อมูลแบบเอมเป็ก (MPEG) 9

2.4.1 หลักการและพื้นฐานในการใช้งานเอ็มพีสาม 10

2.4.2 การเข้ารหัสแบบ MPEG 11

2.4.3 โครงสร้างของข้อมูลเอ็มพีสาม 11

2.5 FAT: File Allocation Table 12

สารบัญ (ต่อ)

หน้า

บทที่ 3 วิธีการออกแบบของเครื่องเล่นเอ็มพีสาม (MP3) 15

3.1 การออกแบบอุปกรณ์ฮาร์ดแวร์ (Hardware) 15

3.1.1 แหล่งจ่ายไฟ 16

3.1.2 ส่วนติดต่อกับ SD CARD 17

3.1.3 ส่วนติดต่อกับ USB 18

3.1.4 ส่วนแปลงสัญญาณดิจิตอลเป็นสัญญาณอะนาลอก 19

3.1.5 ไมโครคอนโทรเลอร์ AVR ATmega128 21

3.1.6 Switch ควบคุม 25

3.1.7 จอ LCD6610 26

3.2 การโปรแกรมทางด้านซอฟต์แวร์ (Software) 27

3.2.1 การใช้งานฟังก์ชั่นการทำงานของเครื่องเล่นเอ็มพีสาม (MP3) 28

3.2.2 การเขียนโปรแกรมสำหรับเครื่องเล่นเอ็มพีสาม (MP3) 35

3.2.3 ขั้นตอนการเบิร์นโปรแกรม 38

3.2.4 ผลของการลงโปรแกรมตามการใช้งานของสวิตซ์ 42

บทที่ 4 ผลการทดลองและวิเคราะห์ผลการทดลอง 48

4.1 ผลการทดลอง 48

4.1.1 ผลการประกอบชิ้นงานของเครื่องเล่นเอ็มพีสาม (MP3) 48

4.1.2 ผลการทดลองการใช้งานร่วมกับสื่อบันทึกข้อมูล 49

4.2 วิเคราะห์ผลการทดลอง 56

บทที่ 5 บทสรุป 58

5.1 สรุปผลการทดลอง 58

5.2 ปัญหาและแนวทางแก้ไข 59

5.3 แนวทางในการพัฒนาต่อไป 59

เอกสารอ้างอิง 60

ภาคผนวก ก วงจรรวมทั้งหมดของเครื่องเล่นเอ็มพีสาม (MP3) 61
ภาคผนวก ข สวิตซ์ควบคุมเครื่องเล่นเอ็มพีสาม (MP3) 63

ประวัติผู้ดำเนินโครงงาน 65

สารบัญตาราง

ตารางที่ หน้า

1.1 แสดงระยะเวลาดำเนินงาน 2

2.1 แสดงรายละเอียดขาสัญญาณต่างๆ เมื่อใช้การเชื่อมต่อในโหมด SD MODE 5

2.2 แสดงรายละเอียดขาสัญญาณต่างๆ เมื่อใช้การเชื่อมต่อในโหมด SPI MODE 5

2.3 แสดงอัตราการบีบอัดข้อมูลและความเร็วในการส่งข้อมูลจากเครื่องอ่านของข้อมูลที่ถูกบีบอัดตามมาตรฐาน MPEG-1 10

2.4 แสดงเวลาที่ใช้ในการแปลงข้อมูล 11

2.5 แสดงรูปแบบข้อมูลเอมเป็กเลเยอร์สาม 12

2.6 แสดงค่าข้อจำกัดต่างๆ ของระบบไฟล์ FAT 13

2.7 เปรียบเทียบขนาดคลัสเตอร์ระหว่าง FAT 32 และ FAT 16 14

3.1 แสดงขาการทำงานของ SD CARD 17

3.2 แสดงขาที่ใช้งานการของ CH375 19

3.3 แสดงขาที่ใช้งานการของ VS1002D 20

3.4 แสดงขา PORT A (PA0-PA7) 23

3.5 แสดงขา PORT B (PB0-PB7) ในที่นี้ใช้เพียง PB0-PB4 23

3.6 แสดงขา PORT C (PC0-PC7) ในที่นี้ใช้เพียง PC2-PC7 24

3.7 แสดงขา PORT D (PD0-PD7) 24

3.8 แสดงขา PORT E (PE0-PE7) ในที่นี้ใช้ PE2-PE7 24

3.9 แสดงขา PORT F (PF0-PF7) ในที่นี้ใช้ PF0-PF4 25

3.10 แสดงหน้าที่ของขาสัญญาณจอ LCD6610 27

3.11 แสดงหน้าที่ของขาสัญญาณของเครื่องโปรแกรม 39

4.1 แสดงผลของการใช้งานเครื่องเล่นเอ็มพีสาม (MP3) กับไฟล์เพลง 128Kbps 50

4.2 แสดงผลของการใช้งานเครื่องเล่นเอ็มพีสาม (MP3) กับไฟล์เพลง 192Kbps 52

4.3 แสดงผลของการใช้งานเครื่องเล่นเอ็มพีสาม (MP3) กับไฟล์เพลง 320Kbps 53

4.4 แสดงผลของการใช้งานเครื่องเล่นเอ็มพีสาม (MP3) กับไฟล์เพลง 128Kbps 54

4.5 แสดงผลของการใช้งานเครื่องเล่นเอ็มพีสาม (MP3) กับไฟล์เพลง 192Kbps 55

4.6 แสดงผลของการใช้งานเครื่องเล่นเอ็มพีสาม (MP3) กับไฟล์เพลง 320Kbps 56

สารบัญรูป

รูปที่	หน้า
2.1 ลักษณะคอนเทคของหน่วยความจำแบบ SD Card และ MMC Card	4
2.2 แสดงพอร์ต USB	6
2.3 แสดงการติดต่อของ USB กับ USB Host Controller	6
2.4 แสดงรายละเอียดขาของไมโครคอนโทรเลอร์ AVR ATmega128	8
2.5 แสดงโครงสร้างภายในของไมโครคอนโทรเลอร์ AVR ATmega128	9
3.1 แสดงรูปการออกแบบของฮาร์ดแวร์ (Hardware)	15
3.2 แสดงรูปวงจรแหล่งจ่ายไฟ	16
3.3 แสดงวงจรส่วนติดต่อกับ SD CARD	17
3.4 แสดงวงจรส่วนติดต่อกับ USB โดยผ่าน CH375	18
3.5 แสดงวงจรการต่อขาของ VS1002D	20
3.6 แสดงวงจรการต่อขาของ AVR ATmega128	22
3.7 แสดงวงจรการต่อสวิตซ์ซึ่งจะต่อกับไมโครคอนโทรเลอร์	26
3.8 แสดงวงจรการต่อขาของ LCD6610	26
3.9 แสดงรูปการออกแบบทางด้านซอฟต์แวร์ (Software)	27
3.10 แสดงหน้าต่างโปรแกรมการสร้างโปรเจค	35
3.11 แสดงหน้าต่างโปรแกรมเลือกฟังก์ชั่นการทำงาน	36
3.12 แสดงหน้าต่างโปรแกรมเลือกเบอร์ไอซี	36
3.13 แสดงหน้าต่างโปรแกรมที่เขียนโปรแกรมไว้ภายใน	37
3.14 แสดงหน้าต่างโปรแกรมสำหรับการเซตค่า	37
3.15 แสดงหน้าต่างโปรแกรมที่ทำการ Build and Compile	38
3.16 แสดงโครงสร้างของเครื่องโปรแกรม	38
3.17 แสดงตำแหน่งขาของเครื่องโปรแกรม	39
3.18 แสดงเครื่องโปรแกรม	40
3.19 แสดงหน้าต่างโปรแกรมการเซตค่า Interface	40
3.20 แสดงหน้าต่างโปรแกรมการเซตค่า Command	41
3.21 แสดงหน้าต่างเมื่อทำการโปรแกรม	41
3.22 แสดงการเชื่อมต่อกับเครื่องเล่น MP3 เพื่อสำหรับโปรแกรม	42
3.23 แสดงหน้าจอ LCD ของการใช้งานฟังก์ชั่น PLAY/PAUSE	42

สารบัญ (ต่อ)

รูปที่	หน้า
3.24 แสดงหน้าจอ LCD ของการใช้งานฟังก์ชั่นเลือกโหมดการเล่นเพลง	43
3.25 แสดงหน้าจอ LCD ของการใช้งานฟังก์ชั่นลดระดับเสียง	44
3.26 แสดงหน้าจอ LCD ของการใช้งานฟังก์ชั่นเพิ่มระดับเสียง	45
3.27 แสดงหน้าจอ LCD ของการใช้งานฟังก์ชั่นเลื่อนเพลงไปข้างหน้า	46
3.28 แสดงหน้าจอ LCD ของการใช้งานฟังก์ชั่นย้อนเพลงไปข้างหลัง	47
4.1 แสดงลายวงจรพิมพ์ของเครื่องเล่นเอ็มพีสาม	48
4.2 แสดงส่วนประกอบชิ้นงานที่เสร็จสมบูรณ์แล้ว	49
4.3 แสดงเพลงที่อยู่ใน SD CARD ที่มีเพลงขนาดบิตเรท 128Kbps	50
4.4 แสดงเพลงที่อยู่ใน SD CARD ที่มีเพลงขนาดบิตเรต 192Kbps	51
4.5 แสดงเพลงที่อยู่ใน SD CARD ที่มีเพลงขนาดบิตเรต 320Kbps	52
4.6 แสดงเพลงที่อยู่ใน USB ที่มีเพลงขนาดบิตเรต 128Kbps	53
4.7 แสดงเพลงที่อยู่ใน USB ที่มีเพลงขนาดบิตเรต 192Kbps	54
4.8 แสดงเพลงที่อยู่ใน USB ที่มีเพลงขนาดบิตเรต 320Kbps	55

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของโครงงาน

เนื่องจากเทคโนโลยีในปัจจุบันได้มีการพัฒนาไปอย่างรวดเร็ว และอุปกรณ์สื่อบันทึกข้อมูลเอสดีการ์ด (SD CARD) เป็นอีกชิ้นหนึ่งได้ถูกสร้างขึ้นมาและเป็นอุปกรณ์ที่ได้รับความนิยม ซึ่งในปัจจุบันได้มีการนำสื่อบันทึกข้อมูลเอสดีการ์ดไปเป็นหน่วยความจำสำรองในการใช้งานอุปกรณ์อิเล็กทรอนิกส์ด้านต่างๆ และยังมีแฟลชไดรฟ์ (Flash Drive) หรือยูเอสบีแฟลชไดรฟ์ (USB Flash Drive) ซึ่งเป็นอุปกรณ์สำหรับเก็บข้อมูลโดยใช้หน่วยความจำแบบแฟลช ทำงานร่วมกับหน่วยความจำยูเอสบี 1.1 หรือ 2.0 ซึ่งอุปกรณ์เอสดีการ์ดได้นำไปใช้งานในอุปกรณ์ โทรศัพท์มือถือ หรือกล้องถ่ายรูป และยูเอสบีแฟลชไดรฟ์ก็ถูกนำไปใช้งานในการเก็บข้อมูลต่างๆ ซึ่งลักษณะพิเศษของอุปกรณ์ทั้ง 2 ชนิดนี้ก็คือมีขนาดเล็กสามารถนำติดตัวไปไหนมาไหน และสามารถเก็บข้อมูลได้เป็นจำนวนมาก

1.2 วัตถุประสงค์ของโครงงาน

1.2.1 เพื่อศึกษาการใช้งานสื่อบันทึกข้อมูลเอสดีการ์ด (SD CARD) และยูเอสบี (USB)

1.2.2 เพื่อศึกษาโครงสร้างและหลักการทำงานของไมโครคอนโทรลเลอร์ AVR ATmega128

1.2.3 เพื่อศึกษาการออกแบบและพัฒนาเครื่องเล่นเพลงเอ็มพีสาม (MP3)

1.3 ขอบเขตของโครงงาน

สร้างเครื่องเล่นไฟล์เสียง MP3 ที่บันทึกในอุปกรณ์ SD CARD และ USB ได้

1.4 ขั้นตอนดำเนินงาน

1.4.1 ศึกษาและค้นคว้าข้อมูลเกี่ยวกับสื่อบันทึกข้อมูล SD CARD สื่อบันทึกข้อมูล USB และไมโครคอนโทรลเลอร์ AVR ATmega128

1.4.2 ออกแบบวงจร

1.4.3 พัฒนาอุปกรณ์ของเครื่องเล่นเพลงเอ็มพีสาม

1.4.4 ทดสอบการใช้งานเครื่องเล่นเอ็มพีสาม พร้อมทั้งปรับปรุงการใช้งาน

1.4.5 จัดทำรูปเล่ม

1.5 แผนการดำเนินงาน

ตารางที่ 1.1 แสดงระยะเวลาดำเนินงาน

กิจกรรม	ช่วงระยะเวลา								
	ปี 2551							ปี 2552	
	มิ.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.
1. ศึกษาและค้นคว้าข้อมูลอุปกรณ์เอสดีการ์ด ยูเอสบีและไมโครคอลโทรเลอร์ AVR เบอร์ ATmega128	←	—	—	→					
2. ศึกษาและทำการเขียนโปรแกรม				←	—	—	→		
3. ออกแบบวงจรฮาร์ดแวร์และซอฟต์แวร์				←	—	—	—	→	
4. พัฒนาและปรับปรุงแก้ไขเครื่องเล่นเพลงเอ็มพีสาม					←	—	—	→	
5. สรุปผลการทดลองและจัดทำรูปเล่ม							←	—	→

1.6 ประโยชน์ที่คาดว่าจะได้รับ

1.6.1 สามารถนำเครื่องบันทึกข้อมูลเอสดีการ์ดและยูเอสบีที่บันทึกข้อมูลทางการศึกษามาเปิดฟังเพื่อทบทวนบทเรียนได้

1.6.2 สามารถให้ความบันเทิงแก่ผู้ใช้งาน โดยเลือกสื่อบันทึกข้อมูลตามความสะดวกของผู้ใช้งานได้ทั้งสื่อบันทึกข้อมูลเอสดีการ์ดและสื่อบันทึกข้อมูลยูเอสบี

1.6.3 สามารถรองรับความต้องการสื่อบันทึกข้อมูลได้ทั้งสองแบบในเครื่องเล่นเอ็มพีสามเครื่องเดียว

1.6.4 สามารถพกพาได้ง่าย

1.7 งบประมาณ

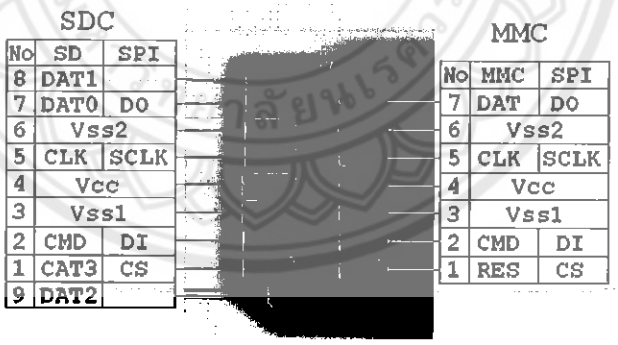
1.7.1 ค่าอุปกรณ์ในการทำชิ้นงาน	1,000 บาท
1.7.2 ค่าหนังสือเกี่ยวกับข้อมูล	700 บาท
1.7.4 ค่าถ่ายเอกสารและค่าจัดทำรูปเล่ม	2,000 บาท
1.7.5 ค่าแผ่นซีดี	50 บาท
รวมเป็นเงิน	3,750 บาท

บทที่ 2
หลักการและทฤษฎีพื้นฐาน

ในบทนี้กล่าวถึงหลักการและทฤษฎีพื้นฐานที่ใช้ในเครื่องเล่นเอ็มพีสาม (MP3) โครงสร้างและหลักการทำงานของ SD CARD **USB (Universal Serial Bus)** การทำงานของไมโครคอนโทรเลอร์ การบีบอัดข้อมูลแบบเอมเป็ก (MPEG) และ FAT:File Allocation Table

2.1 โครงสร้างและหลักการทำงานของ SD CARD

ในปัจจุบันหน่วยความจำแบบ SD Card (Secure Digital Memory Card) นี้เป็นหน่วยความจำที่ได้รับความนิยมมากเพราะเป็นอุปกรณ์ที่สามารถพกพาได้สะดวก ซึ่งหน่วยความจำแบบ SD Card นี้จะมีฟังก์ชั่นการทำงานคล้ายกับหน่วยความจำชนิดหนึ่ง MMC Card (Multi Media Card) ซึ่งภายในหน่วยความจำแบบ SD Card นี้จะมีไมโครคอนโทรเลอร์ แฟรชเมม โมรี่คอนโทรล (erase, read,write and error control) ซึ่งการโอนย้ายข้อมูลโดยปกติระหว่างเมมโมรีกับโฮสคอนโทรลนี้จะย้ายข้อมูล 512 ไบต์ต่อบล็อกโดยอัตโนมัติ ดังนั้นมันจะดูเหมือนฮาร์ดดิสก์ทั่วๆ ไปนั่นเอง ซึ่งในจุดนี้เราสามารถนำไปประยุกต์ต่างๆได้ โดยทั่วไปไฟล์ระบบนั้นจะเป็นแบบ FAT 32 นั้นความจุของหน่วยความจำของการ์ดจะต้องมีค่ามากกว่า 2 Gbyte



รูปที่ 2.1 ลักษณะคอนแทคของหน่วยความจำแบบ SD Card และ MMC Card

จากรูปที่ 2.1 จะแสดงรูปคอนแทคซึ่งใช้ในการติดต่อสื่อสารของหน่วยความจำแบบ SD Card และ MMC Card ซึ่งหน่วยความจำแบบ MMC Card จะมาช่องติดต่อ 7 คอนแทค ส่วนหน่วยความจำแบบ SD Card นั่นจะเพิ่มมาอีก 2 คอนแทค ในการโอนย้ายข้อมูลระหว่างโฮลกับการ์ดนั้นจะกระทำในสัญญาณนาฬิกา การติดต่อระหว่างโฮสกับหน่วยความจำแบบ SD Card นั้น

จะมีอยู่ 2 Mode คือ SD Mode และSPI Mode ซึ่งในการติดต่อนี้จะทำการติดต่อหน่วยความจำแบบ SPI Mod

2.1.1 รายละเอียดขาสัญญาณของ SD CARD

การเชื่อมต่อกับ SD Card สามารถทำได้ 2 แบบ ก็คือ การเชื่อมต่อแบบ SD MODE และเชื่อมต่อแบบ SPI MODE ดังรายละเอียดตามตาราง 2.1 และ 2.2 ตามลำดับดังนี้

ตารางที่ 2.1 แสดงรายละเอียดขาสัญญาณต่างๆ เมื่อใช้การเชื่อมต่อในโหมด SD MODE

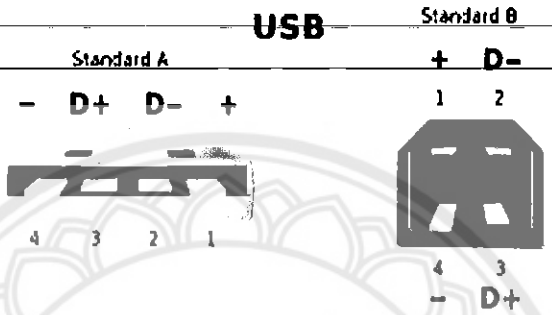
SD MODE			
Pin No.	Name	Type	Description
1	CD/DAT3	I/O,PP	Card Detection/Data Line 3
2	CMD	I/O,PP	Command/Response
3	Vss	S	Supply Voltage Ground
4	Vdd	S	Supply Voltage
5	CLK	I	Clock
6	Vss2	S	Supply Voltage Ground
7	DAT0	I/O,PP	Data Line 0
8	DAT1	I/O,PP	Data Line 0
9	DAT2	I/O,PP	Data Line 0

ตารางที่ 2.2 แสดงรายละเอียดขาสัญญาณต่างๆ เมื่อใช้การเชื่อมต่อในโหมด SPI MODE

SPI MODE			
Pin No.	Name	Type	Description
1	CS	I	Chip Select (Active Low)
2	DI	I	Host-to-card Command Data input
3	VSS	S	Supply Voltage Ground
4	VDD	S	Supply Voltage
5	SCLK	I	Clock
6	VSS2	S	Supply Voltage Ground
7	DO	O	Card-to-Host Data and Status
8	RVS	-	Reserved
9	RVS	-	Reserved

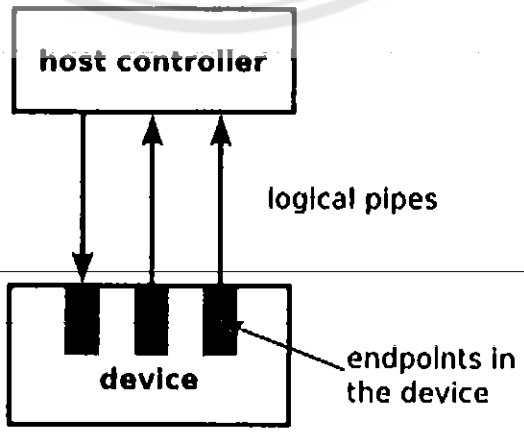
2.2 USB (Universal Serial Bus)

ระบบเชื่อมต่ออนุกรมความเร็วสูงของคอมพิวเตอร์ ซึ่งเป็นช่องทางในการสื่อสารระหว่างคอมพิวเตอร์กับอุปกรณ์ I/O (Input/output devices) อื่นๆ ที่นำมาเชื่อมต่อกับคอมพิวเตอร์ไม่ว่าจะเป็น Printer, Modem, Mouse, Keyboard, Digital Camera และอื่นๆ อีกมากมาย โดยในปัจจุบันถือเป็นมาตรฐานอุตสาหกรรมที่คอมพิวเตอร์ยุคใหม่ควรจะต้องมี ทุกวันนี้ USB พัฒนามาถึง version 3.0



รูปที่ 2.2 แสดงพอร์ต USB

การสื่อสารข้อมูลทั้งหมดบนระบบ USB เริ่มมาจาก Host ที่ติดตั้งอยู่บนเมนบอร์ดซึ่งทำงานภายใต้การควบคุมของซอฟต์แวร์และฮาร์ดแวร์ ซึ่ง Host นี้ประกอบไปด้วย USB Host Controller ซึ่งเป็นระบบควบคุมที่ทำให้เกิดการถ่ายเทข้อมูลบน USB Bus นอกจากนี้ก็มี Root Hub ซึ่งเป็น Port ซึ่งใช้เป็นที่เชื่อมต่อกับอุปกรณ์ USB บนเมนบอร์ด ซึ่งจะมีชิปที่ทำหน้าที่เป็น USB Host Controller



รูปที่ 2.3 แสดงการติดต่อของ USB กับ USB Host Controller

2.3 การทำงานของไมโครคอนโทรเลอร์

ไมโครคอนโทรเลอร์คือชิปประมวลผลอย่างหนึ่งซึ่งทำหน้าที่ประมวลผลตามโปรแกรมหรือชุดคำสั่งเหมือนกับไมโครโพรเซสเซอร์ โครงสร้างภายในจะเป็นวงจรรวมขนาดใหญ่ประกอบไปด้วย หน่วยความจำทางคณิตศาสตร์และลอจิก บัสข้อมูล บัสควบคุม บัสที่อยู่ พอร์ตขนาน พอร์ตอนุกรม รีจิสเตอร์ หน่วยความจำ วงจรนับ วงจรจับเวลา และวงจรอื่นๆรวมกันอยู่ภายในชิปไอซี ไมโครคอนโทรเลอร์ถูกออกแบบมาเพื่อใช้ในงานควบคุม สามารถติดต่อกับอุปกรณ์อินพุตและเอาต์พุตได้สะดวกใช้งานง่ายสามารถทำงานได้โดยใช้ชิปเดียว มีคำสั่งที่สนับสนุนในการเขียนโปรแกรมควบคุมและสามารถเข้าถึงข้อมูลระดับบิตได้

2.3.1 โครงสร้างทั่วไป

ไมโครคอนโทรเลอร์เป็นวงจรรวมขนาดใหญ่มีวงจรซับซ้อน ดังนั้นเพื่อให้ง่ายต่อการทำความเข้าใจจึงอธิบายโครงสร้างของสถาปัตยกรรมของไมโครคอนโทรเลอร์ดังนี้

(ก). หน่วยประมวลผลกลางหรือซีพียู (CPU : Central Processing Unit)

มีหน้าที่รับคำสั่งและนำคำสั่งนั้นมาประมวลผลหรือตีความคำสั่ง ภายในซีพียูจะประกอบไปด้วยหน่วยย่อยที่สำคัญที่ทำหน้าที่คำนวณทางคณิตศาสตร์ และกระทำทางตรรกะ หน่วยย่อยนี้เรียกว่า ALU (Arithmetic Logic Unit)

(ข). หน่วยความจำ (Memory) สามารถแบ่งได้ออกเป็น 2 กลุ่มคือ

1. หน่วยความจำโปรแกรม (ROM : Read Only Memory) ใช้ในการเก็บโปรแกรมการทำงานที่ผู้เขียนพัฒนาโปรแกรมสร้างขึ้น ซึ่งแยกประเภทได้ดังนี้

- Mask Rom : ติดมากับโรงงานผลิตชิป สามารถอ่านได้อย่างเดียว เขียนทับไม่ได้
- PROM (Programmable Read-Only Memory) : สามารถเขียนได้ครั้งเดียว โดยป้อนพลังงานพัลส์แรงดันสูง (High voltage pulse)
- EPROM (Erasable Programmable Read-Only Memory) : สามารถลบข้อมูลโดยการนำไปตากแดดจัดๆ ที่มีรังสีอัลตร้าไวโอเลต
- EEPROM/FLASH (Electrically Erasable Programmable Read-Only Memory) : สามารถลบและเขียนข้อมูล แก้ไขได้หลายครั้ง

2. หน่วยความจำข้อมูล (RAM : Random Access Memory) เป็นหน่วยความจำที่ใช้ในการอ่านและเขียนข้อมูล สามารถเพิ่มหน่วยความจำส่วนนี้ได้ แบ่งออกเป็น 4 ประเภทคือ

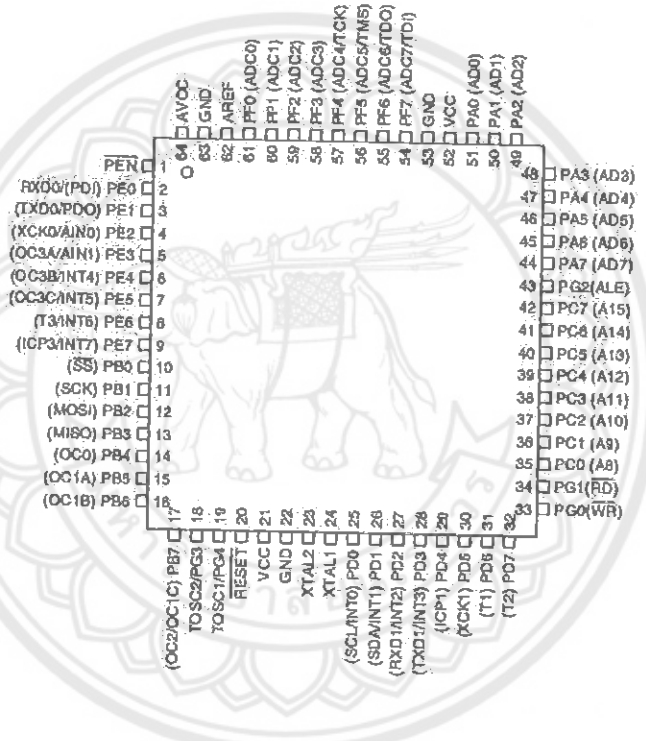
- เก็บข้อมูลแบบบิต
- เก็บข้อมูลแบบรีจีสเตอร์
- เก็บข้อมูลแบบทั่วไป
- เก็บข้อมูลแบบ stack

(ค). อินพุต/เอาท์พุต (I/O Port)

ทำหน้าที่ในการรับ – ส่งสัญญาณดิจิตอลเพื่อเชื่อมต่อกับตัวไมโครคอนโทรเลอร์ อินพุตพอร์ต เอาท์พุตพอร์ต และยังทำหน้าที่ติดต่อกับอุปกรณ์ภายนอกโดยผ่านวงจรอินเตอร์เฟส– หรือวงจรเชื่อม (Interface)

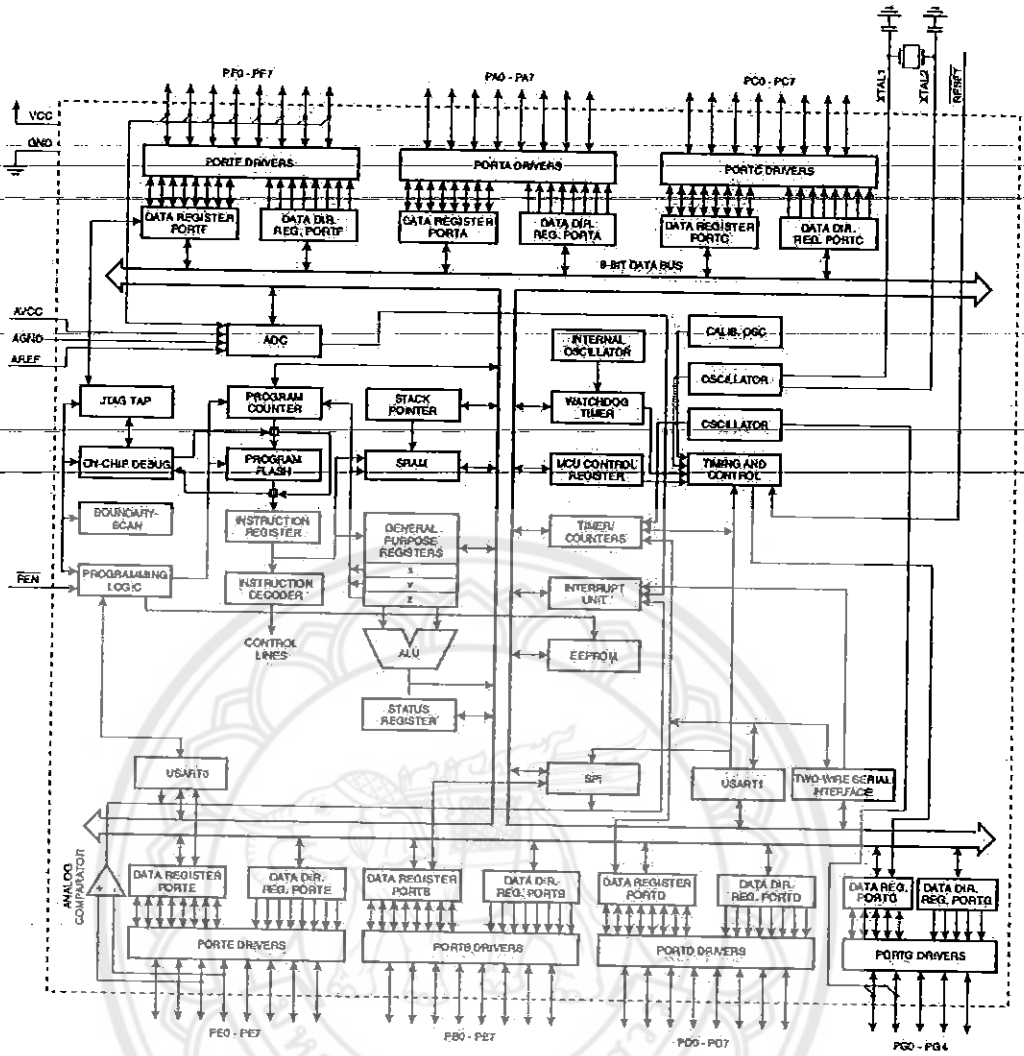
2.3.2 ไมโครคอนโทรเลอร์ AVR ATmega128

ไมโครคอนโทรเลอร์ AVR ATmega128 เป็นไมโครคอนโทรเลอร์ขนาด 8 บิต โดยที่มีการจัดวางรูปแบบขาดังรูปที่ 2.4 นี้



รูปที่ 2.4 แสดงรายละเอียดขาของไมโครคอนโทรเลอร์ AVR ATmega128

ส่วนประกอบหลักคือ ซีพียู หน่วยความจำโปรแกรม หน่วยความจำข้อมูล ส่วนติดต่อพอร์ต ส่วนจัดการสัญญาณนาฬิกาของระบบ วงจรรีเซต (Reset) หลัก ส่วนควบคุมกาตอบสนองอินเตอร์รัปต์ (Interrupt) และส่วนจัดการด้านไฟเลี้ยง แต่จะมีโมดูลพิเศษเพิ่มเติมเข้ามาเพื่อช่วยให้ AVR ATmega128 มีความสามารถสูงมากขึ้น ได้แก่ โมดูลไทเมอร์/เคาน์เตอร์ โมดูลสร้างสัญญาณ PWM โมดูลเชื่อมต่ออนุกรม โมดูลวัอตดอกซ์ไทเมอร์ โมดูลแปลงสัญญาณอะนาลอกเป็นสัญญาณดิจิตอลขนาด 10 บิต โมดูลเปรียบเทียบสัญญาณอะนาลอก ดังแสดงในรูป 2.5 นี้



รูปที่ 2.5 แสดงโครงสร้างภายในของไมโครคอนโทรเลอร์ AVR ATmega128

2.4 การบีบอัดข้อมูลแบบเอมเป็ก (MPEG)

เอมเป็ก (MPEG) นั้นย่อมาจาก Moving Picture Expert Group ซึ่งเป็นชื่อของกลุ่มคนที่ร่วมมือกันสร้างมาตรฐานสากล (International Standard) เพื่อที่จะใช้ในการเข้ารหัสข้อมูลภาพและเสียงที่อยู่ในรูปแบบของสัญญาณดิจิตอล ต่อจากนั้นได้มีกลุ่มวิศวกรชาวยุโรปได้มีการพัฒนาบีบอัดไฟล์ในรูปแบบของเอมเป็ก (MPEG) ต่อจนกลายมาเป็นการบีบอัดไฟล์แบบ เอ็มพี3 (MP3 : MPEG-1 Audio Layer 3) ซึ่งใช้กันอย่างแพร่หลายในปัจจุบันภายให้มาตรฐาน ISO/IEC ซึ่งเป็นมาตรฐานที่นำมาใช้เข้ารหัสหรือถอดรหัสข้อมูลของตนเองได้โดยไม่ต้องขออนุญาตหรือจ่ายค่าลิขสิทธิ์ให้กับผู้ใด โดยมาตรฐานของ MP3 คือ ISO/IEC 11172-3

มาตรฐานการบีบอัดไฟล์แบบเอมเป็กนั้นแบ่งออกได้เป็นกลุ่มๆ ตามชนิดข้อมูลที่ถูกเข้ารหัสและการนำไปใช้งานซึ่งแบ่งออกได้เป็น 5 กลุ่มดังนี้

1. MPEG-1 ข้อมูลภาพและเสียง ใช้ในระบบวีดีโอซีดี และเสียงเพลง

2. MPEG-2 เข้ารหัสข้อมูลภาพและเสียง ใช้ในระบบโทรทัศน์ดิจิตอลและดีวีดี

3. MPEG-4 เป็นส่วนขยายของ MPEG-1 เพื่อรับรูปแบบมัลติมีเดียต่างๆ เช่น 3D หรือการเข้ารหัสที่มีประสิทธิภาพมากขึ้น MPEG-4 แบ่งออกเป็นหลายส่วนตามหน้าที่แต่ละส่วน และจะปล่อยให้ผู้ผลิตซอฟท์แวร์เป็นผู้พัฒนาโปรแกรมที่ใช้จริงๆเอง ไม่จำเป็นต้องตาม MPEG-4 เต็มชุดก็ได้ พัฒนาได้เป็นบางส่วนก็พอ (แบบเดียวกับ MPEG-3 ที่หยิบแต่ส่วนออดิโอไปทำ)

4. MPEG-7 ไม่ใช่มาตรฐานเกี่ยวกับภาพและเสียงเหมือนอันอื่นๆ แต่เป็นมาตรฐานที่ใช้เก็บข้อมูลเกี่ยวกับตัวมีเดีย (metadata) เช่น หนังแผ่นนี้ชื่ออะไร หรือถ้าหนังเล่นมาถึงตอนนี้ ให้เล่นเพลงนี้พร้อมขึ้นซับไทเทิลไฟล์นี้เป็นต้น อิมพลีเมนต์โดย XML

5. MPEG-21 เป็นมาตรฐานที่กำหนดขึ้นว่าด้วยเรื่องเกี่ยวกับ Multimedia Framework นอกจากนี้ทั้ง 5 กลุ่มนี้ ยังมีกลุ่มแบบอื่นๆอีก แต่ไม่ได้ใช้ในปัจจุบันคือ

MPEG-3 เป็นมาตรฐานที่เตรียมใช้กับ HDTV(High Definition Television หรือโทรทัศน์ความละเอียดสูง) แต่สุดท้ายไม่ได้ใช้ เพราะว่าแค่เทคโนโลยี MPEG-2 ที่มีอยู่เดิมก็เพียงพอสำหรับ HDTV แล้ว

2.4.1 หลักการและพื้นฐานในการใช้งานเอ็มพีสาม

เอ็มพี 3 คือรูปแบบไฟล์ที่ใช้กันอย่างแพร่หลายในปัจจุบัน โดยมีรูปแบบการบีบอัดไฟล์อยู่รูปแบบมานคฐานแบบ MPEG-1 ซึ่งเป็นรูปแบบของการเข้ารหัสข้อมูลแบบข้อมูลภาพและ เสียง โดยตัวของ MPEG-1 นี้จะแบ่งออกเป็น 3 เลเยอร์ (layer) ตามความสามารถและซับซ้อนในการเข้ารหัสข้อมูลโดยเลเยอร์ที่ 1 จะมีความซับซ้อนในการเข้ารหัสน้อย ทำให้สามารถบีบอัดข้อมูลได้น้อย และในทางกลับกัน เลเยอร์ที่ 3 ก็มีความซับซ้อนในการเข้ารหัสข้อมูลมากที่สุด นั่นคือ เอ็มพี 3 จะมีรูปแบบการเข้ารหัสที่มีความซับซ้อนสูงและสามารถที่จะบีบอัดข้อมูลได้มากจึงสามารถจดเก็บข้อมูลได้เป็นจำนวนมาก แต่ถึงอย่างไรรูปแบบในการบีบอัดของทั้ง 3 เลเยอร์ก็มีรูปแบบมาตรฐานเหมือนกันทั้งหมด โดยความเร็วในการบีบอัดและลักษณะที่แตกต่างกันของทั้ง 3 เลเยอร์แสดงได้ดังตารางที่ 2.3 และตารางที่ 2.4

ตารางที่ 2.3 แสดงอัตราการบีบอัดข้อมูลและความเร็วในการส่งข้อมูลจากเครื่องอ่านของข้อมูลที่ถูกบีบอัดตามมาตรฐาน MPEG-1

มาตรฐานการบีบอัด	อัตราส่วน	ความเร็วในการส่งข้อมูล
MPEG-1 Layer 1	1:4	348 กิโลบิต/วินาที
MPEG-1 Layer 2	1:6 ถึง1:8	256-192 กิโลบิต/วินาที
MPEG-1 Layer 3	1:10 ถึง 1:12	128-115 กิโลบิต/วินาที

ตารางที่ 2.4 แสดงเวลาที่ใช้ในการแปลงข้อมูล

มาตรฐานการบีบอัด	อัตราส่วน	เวลาที่ใช้ในการถอดรหัส
MPEG-1 Layer 1	4:1	19ns
MPEG-1 Layer 2	6:1	35ns
MPEG-1 Layer 3	12:1	59ns

2.4.2 การเข้ารหัสแบบ MPEG

การเข้ารหัสแบบ MPEG นั้นโดยทั่วไปแล้วจะทำการตัดข้อมูลเสียงที่จัดเก็บบางส่วนออก แต่สามารถคงรายละเอียดของเสียงที่ได้ยินไว้เท่าเดิม ทั้งนี้มาจากการลดขนาดข้อมูลตามมาตรฐาน MPEG นั้น ใช้พฤติกรรมการได้ยินเสียงของมนุษย์มาเป็นเครื่องมือในการลดขนาดข้อมูลอย่างที่รู้กันโดยทั่วไปว่า หูของมนุษย์นั้นมีขีดจำกัดในการรับฟัง โดยที่ความถี่ที่มนุษย์สามารถรับฟังได้ คือ ความถี่ระหว่าง 20-20,000 เฮิรตซ์ ซึ่งถ้าอยู่เกินช่วงนี้ไป หูของมนุษย์จะไม่สามารถรับฟังได้ยินเสียงนั้นได้ แต่ถึงอย่างไรหูของมนุษย์ใช่ว่าจะมีความไวต่อทุกๆความถี่ที่ได้ยินเท่ากันหมดทุกคน ที่ความถี่ 2-4 กิโลเฮิรตซ์ หูของมนุษย์มีความไวต่อเสียงมากที่สุดและเมื่อสังเกตที่ความถี่ความถี่ต่างๆ จะพบว่าที่ความถี่สูงมากหรือต่ำมาก มีความจำเป็นที่จะต้องใช้ความดังมากๆเพื่อที่จะทำให้มนุษย์ได้ยินจากคุณสมบัติการรับฟังทั้งหมดที่กล่าวมานั้นเรียกรวมกันว่า "ไซโคอคูสติกโมเดล" (Pshychoacoustic Model) ซึ่งเป็นเครื่องมือที่สำคัญมากในการนำมาเป็นตัวกรองเพื่อลดขนาดในการบีบอัดข้อมูลเสียงตามมาตรฐาน MPEG ซึ่งมีลำดับกระบวนการในการบีบอัดดังนี้

1. นำข้อมูลเสียงดิจิตอลป้อนเข้าฟิลเตอร์(Filter) เพื่อแยกเสียงออกเป็นช่วงความถี่ย่อยๆ (Sub-bands) ซึ่งมีความกว้างเท่ากับย่านความถี่วิกฤต จำนวน 32 ช่องความถี่ เรียกขั้นตอนนี้ว่า Sub-band Filtering

2. ใช้ไซโคอคูลติกโมเดลเป็นเครื่องมือในการวิเคราะห์ข้อมูลส่วนที่ไม่มีผลต่อการได้ยินของมนุษย์ออกไป โดยพิจารณาระหว่างช่วงความถี่ 2 ช่วงที่ติดกันและพิจารณาย่อยลงไปในแต่ละช่วงความถี่ด้วย

3. ถ้าวิเคราะห์แล้วพบว่าเสียงช่วงใดไม่มีผลกระทบต่อการได้ยิน ให้ตัดข้อมูลส่วนนั้นออกไปและไม่นำไปเข้ารหัสในส่วนถัดไป

4. นำข้อมูลที่เหลือมาเข้ารหัสซึ่งมีวิธีแตกต่างกัน ซึ่งอยู่แต่ละเลเยอร์

2.4.3 โครงสร้างของข้อมูลเอ็มพีสาม

ข้อมูลที่ถูกบีบอัดตามมาตรฐาน เอ็มพี 3 นั้นจะอยู่ในลักษณะเฟรมข้อมูล โดยในแต่ละเฟรมข้อมูลจะมีส่วนประกอบภายในอยู่ 4 ส่วนคือ

1. หัวข้อมูล (Header) เป็นข้อมูลขนาด 32 บิต แสดงลักษณะทั่วไปของไฟล์นั้นๆ

2. ส่วนตรวจสอบความผิดพลาด (CRC) เป็นขนาดข้อมูล 16 บิต ใช้ตรวจสอบข้อมูลภายในเฟรมว่าถูกต้องหรือไม่ จะมีหรือไม่ก็ได้

3. ข้อมูลข้างเคียง (Side Information) มีขนาด 17 หรือ 32 ไบต์ (17 ไบต์สำหรับระบบโมโน 32 ไบต์สำหับระบบอื่นๆ) เป็นส่วนที่เก็บองค์ประกอบที่ใช้ในการถอดรหัส

4. ข้อมูลหลัก (Main Data) มีความยาวขึ้นอยู่กับอัตราการส่งข้อมูล (Baud rate) และอัตราการสุ่มข้อมูลในการแปลงกลับเป็นสัญญาณอะนาลอก(Sampling Frequency)

ตารางที่ 2.5 แสดงรูปแบบข้อมูลเอมเป็กเลเยอร์สาม

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

2.5 FAT: File Allocation Table

เป็นระบบไฟล์ที่ใช้ในระบบปฏิบัติการในตระกูล Microsoft และเป็นระบบไฟล์ที่มีพัฒนาการมาต่อเนื่อง ระบบไฟล์ในตระกูลนี้มีลักษณะคือ เป็นการกำหนดหมายเลขให้กับทุกๆ คลัสเตอร์ (Cluster) ในแต่ละ พาร์ทิชั่น (Partition) แล้วทำการสร้างตารางที่มีจำนวนช่องตามจำนวนคลัสเตอร์นั้นเพื่อเป็นการระบุสถานที่หรือคลัสเตอร์ที่ทำการเก็บขอ้มูลของไฟล์แต่ละไฟล์ และมีตารางอีกตารางหนึ่งที่เรียกว่า "ไดเร็คทอรี่" (Directory) สำหรับเก็บข้อมูลรายละเอียดของไฟล์ เช่น คุณสมบัติต่างๆ และ หมายเลขคลัสเตอร์เริ่มต้นที่เก็บข้อมูลจริงๆ ระบบๆไฟล์ FAT มีหลายรุ่นดังต่อไปนี้

FAT 12 เป็นระบบไฟล์ที่ใช้ใน Floppy Disk และ Hard Dish ที่มีขนาดไม่เกิน 16 MBs หมายเลขคลัสเตอร์มีขนาด 12 บิต จึงสามารถอ้างถึง คลัสเตอร์ ได้เพียง 4096 คลัสเตอร์เท่านั้น

FAT 16 ใช้ตัวเลขขนาด 16 บิต ในการกำหนดหมายเลข คลัสเตอร์ จึงกำหนดหมายเลขได้ 65536 หมายเลข ระบบไฟล์นี้มีใช้ในระบบปฏิบัติการของ Microsoft ทุกรุ่น พาร์ทิชั่นที่จะใช้ระบบไฟล์นี้ได้ต้องมีขนาดไม่เกิน 2 GBs. FAT 16 ได้รับการปรับปรุงให้มีความสามรถมากขึ้นใน Windows 95 เพื่อให้สามารถใช้งานกับไฟล์ท่ามีชื่อยาวได้ไม่เกิน 256 ตัว เรียก FAT 16 รุ่นนี้ว่า Virtual FAT หรือ VFAT

FAT 32 ระบบไฟล์ระบบนี้จะใช้หมายเลขขนาด 28 บิต ซึ่งตามทฤษฎีจะสามรถกำหนดคลัสเตอร์ ได้มากถึง 268,435,456 คลัสเตอร์ และสามารถใช้กับพาร์ทิชั่นที่มีขนาดใหญ่ได้ถึง 2 Terabytes ระบบ ไฟล์ FAT 32 นี้มีใช้ใน Windows 95 OSR2 ขึ้นไป แต่ไม่ได้ใช้ใน Windows NT

ตารางที่ 2.6 แสดงค่าข้อจำกัดต่างๆ ของระบบไฟล์ FAT

คุณลักษณะ	FAT 12	FAT 16	FAT 32
ใช้สำหรับ	Floppies and very smallhard disk volumes	Small to moderate-sized hard disk	Medium-sized to very large hard volumes
ขนาดของการเข้ารหัสข้อมูล	12 bits	16 bits	28 bits
จำนวนมากสุดของคลัสเตอร์	4,096	65,536	268,436,456
ขนาดของคลัสเตอร์ที่ใช้งาน	0.5 KB to 4 KB	2 KB to 32 KB	4 KB to 32 KB
ขนาดความจุสูงสุด	16,736,256	2,147,1323,200	about 2^41

FAT ที่นิยมใช้กันอยู่ในของระบบปฏิบัติการของ Windows คือ FAT 16 โดย FAT จะทำหน้าที่จัดข้อมูลหลายๆเซ็กเตอร์ โดยในแต่ละเซ็กเตอร์จะแบ่งย่อยออกเป็นอีกหลายๆคลัสเตอร์ (Cluster) ซึ่งในระบบ FAT 16 (16 bit) นั้นสามารถอ้างอิงหรือชี้ตำแหน่งคลัสเตอร์ได้สูงสุด 65,536 คลัสเตอร์ (ข้อมูลทางดิจิตอล 1 บิต สามารถแบ่งได้เป็น 2 สถานะ คือ 0 กับ 1 ดังนั้นถ้า 16 บิต สามารถเป็นได้เท่ากับ 2 ยกกำลัง 16 ซึ่งเท่ากับ 65,536) แต่ในระบบ FAT 16 นั้นสามารถมีขนาดคลัสเตอร์ใหญ่ที่สุด 32 KB (Kilobyte) ดังนั้นในระบบ FAT 16 จึงสามารถอ้างข้อมูลในหนึ่งพาร์ทิชั่น (Partition) ได้สูงสุดที่ 2 GB(Gigabyte) (32 KB คูณ 65,536 คลัสเตอร์ เท่ากับ 2,097,152 หรือ 2,048 MB

ดังนั้นถ้าต้องการใช้ฮาร์ดดิสก์ขนาด 2 GB หรือฮาร์ดดิสก์ที่มีพาร์ทิชั่นขนาดเท่ากับ 2 GB หมายความว่าขนาดของคลัสเตอร์ที่เล็กที่สุดเท่ากับ 32 KB ซึ่งหมายความว่า ไม่ว่าไฟล์ที่ต้องการเก็บในฮาร์ดิสก์จะมีขนาดเล็กแค่ไหนก็ตาม ฮาร์ดิสก์ก็จะต้องจองพื้นที่ให้ไฟล์นี้ไม่ต่ำกว่า 32 KB ตัวอย่างเช่น ถ้าต้องการเก็บไฟล์ขนาด 1 KB ลงในฮาร์ดิสก์ที่เป็น FAT 16 ฮาร์ดิสก์จะต้องจองพื้นที่เพื่อเก็บไฟล์นี้ 32 KB ซึ่งหมายความว่าจะต้องสูญเสียพื้นที่ในฮาร์ดิสก์ไปโดยไม่สามารถใช้ได้ถึง 31 KB ดังนั้นยิ่งมีไฟล์ขนาดเล็กกว่า 32 KB มากเท่าไร หมายความว่า จะสูญเสียพื้นที่ว่างบนฮาร์ดิสก์ไปโดยไม่ได้ใช้ประโยชน์มากขึ้นเท่านั้น วิธีแก้ปัญหาการสูญเสียเนื้อที่นี้อาจจะทำได้ 2 กรณี คือ

1. ถ้ายังต้องการใช้ฮาร์ดิสก์ที่เป็น FAT 16 อยู่จะต้องใช้วิธีแบ่งพาร์ทิชั่นฮาร์ดิสก์ให้มีขนาดเล็กลง เพื่อให้ฮาร์ดิสก์มีขนาดคลัสเตอร์เล็กลง จะได้ทำให้เนื้อที่ว่างที่ไม่สามารถใช้งานเหลือน้อยลง ดังตารางข้างล่างนี้คือ ถ้าใช้แบ่งพาร์ทิชั่นฮาร์ดิสก์ไว้ที่ขนาด 512 MB ต่อพาร์ทิชั่น ขนาด

ของคลัสเตอร์จะลดลงเหลือแค่ 8 KB ซึ่งจะทำให้ความสูญเสียเนื้อที่บนฮาร์ดิสก์โดยเปล่าประโยชน์ลดน้อยลองถึง 3 เท่าเมื่อเทียบกับการแบ่งพาร์ทิชั่นไว้ที่ขนาด 2 GB แต่วิธีการแบ่งฮาร์ดิสก์ออกเป็นหลายๆพาร์ทิชั่นนี้อาจทำให้เกิดความยุ่งยากเช่น มีไดร์ฟฮาร์ดิสก์หลายๆ ไดร์ฟอาจทำให้สับสนเวลาใช้งาน

2. ให้ใช้ฮาร์ดิสก์ที่เป็น FAT 32 ซึ่งเป็นระบบ FAT แบบใหม่ 32 บิต ซึ่งมีในระบบปฏิบัติการwindows 95 OSR2 (Windows 95 OEM Service Release 2) หรือใน windows 98 หรือใช้โปรแกรมที่ช่วงแปลง FAT 16 ให้เป็น FAT 32 อย่างเช่น Partition-It, Partition Magic เป็นต้น สำหรับ FAT 32 นี้ จำนวนคลัสเตอร์ที่จะอ้างอิงถึงได้เท่ากับ 2 ยกกำลัง 28 หรือเท่ากับ 268,436,456 คลัสเตอร์ ดังนั้นเมื่อใช้ขนาดของคลัสเตอร์ 4 KB ขนาดของพาร์ทิชั่นสูงสุดที่จะมีได้เท่ากับ 8 GB และขนาดของคลัสเตอร์สูงสุดที่ 32 KB จะทำให้ฮาร์ดิสก์สามารถมีพาร์ทิชั่นได้สูงสุดที่ 2 TB(1 TB = 1024 GB

ตารางที่ 2.7 เปรียบเทียบขนาดคลัสเตอร์ระหว่าง FAT 32 และ FAT 16

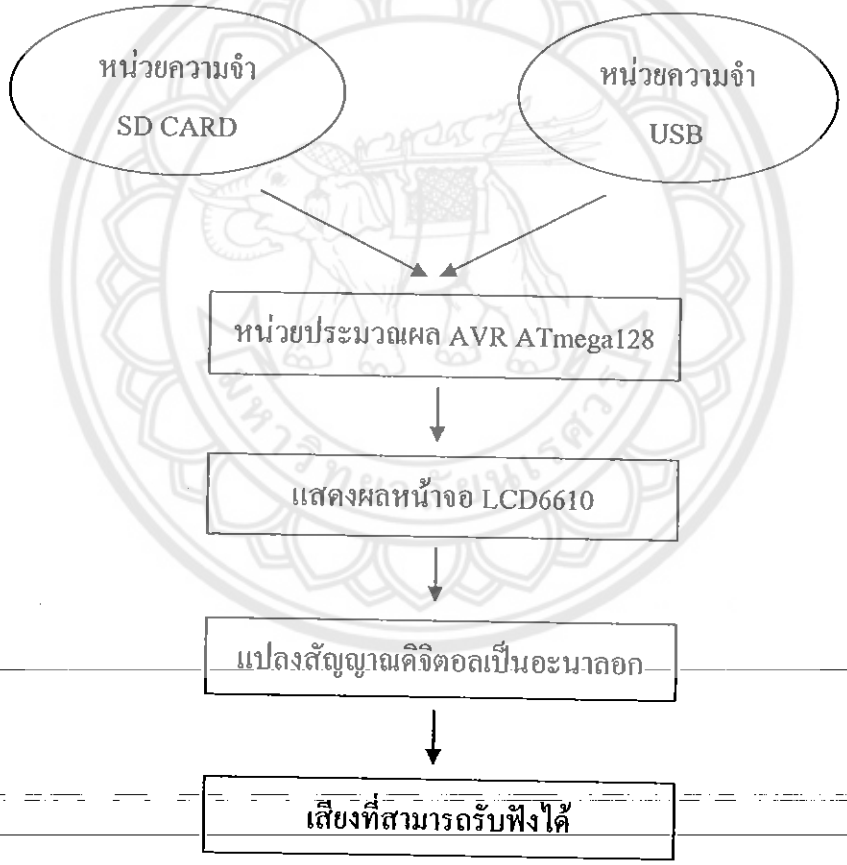
ขนาดของพาร์ทิชั่น	ขนาดของคลัสเตอร์ FAT 32	ขนาดของคลัสเตอร์ FAT 16
น้อยกว่า 260 Megabyte	512 byte	4 kilobyte
260-511 Megabyte	4 kilobyte	8 kilobyte
512-1023 Megabyte	4 kilobyte	8 kilobyte
1024-2048 Megabyte	4 kilobyte	16 kilobyte
2-8 Gigabyte	4 kilobyte	32 kilobyte
8-16 Gigabyte	8 kilobyte	32 kilobyte
16-32 Gigabyte	16 kilobyte	32 kilobyte
มากกว่า 32 Gigabyte	32 kilobyte	32 kilobyte

ในเครื่องคอมพิวเตอร์ที่ใช้ในระบบปฏิบัติการ windows 95 นั้น ถ้าสำรวจดูในโฟล์เดอร์ (Folder) ต่างๆ จะพบว่ามีขนาดไฟล์เล็กๆเป็นจำนวนหลายร้อยไฟล์ ตัวอย่างเช่น ไฟล์ที่มีนามสกุล .dll ซึ่งไฟล์เหล่านี้จะเป็นต้นตอของการทำให้เกิดการสูญเสียเนื้อที่ในฮาร์ดิสก์อย่างไม่มีประโยชน์ โดยไม่ว่าฮาร์ดิสก์จะมีคลัสเตอร์เป็นแบบไฟล์แฟต 32 (FAT 32) หรือไฟล์แฟต 16 (FAT 16) ก็ตาม

บทที่ 3
วิธีการออกแบบเครื่องเล่นเอ็มพีสาม (MP3)

ในบทนี้จะกล่าวถึงวิธีการออกแบบของเครื่องเล่นเอ็มพีสาม (MP3) การออกแบบอุปกรณ์ Hardware และการโปรแกรมทางด้าน Software ซึ่งแยกออกเป็น แหล่งจ่ายไฟ ส่วนติดต่อกับ SD CARD ส่วนติดต่อ USB ส่วนแปลงสัญญาณดิจิตอลเป็นสัญญาณอะนาลอก ไมโครคอนโทรเลอร์ AVR ATmega128 Switch ควบคุม จอ LCD6610 การเขียนโปรแกรมสำหรับเครื่องเล่นเอ็มพีสาม (MP3) ขั้นตอนการเบิร์นโปรแกรม และผลของการลงโปรแกรม

3.1 การออกแบบอุปกรณ์ฮาร์ดแวร์ (Hardware)



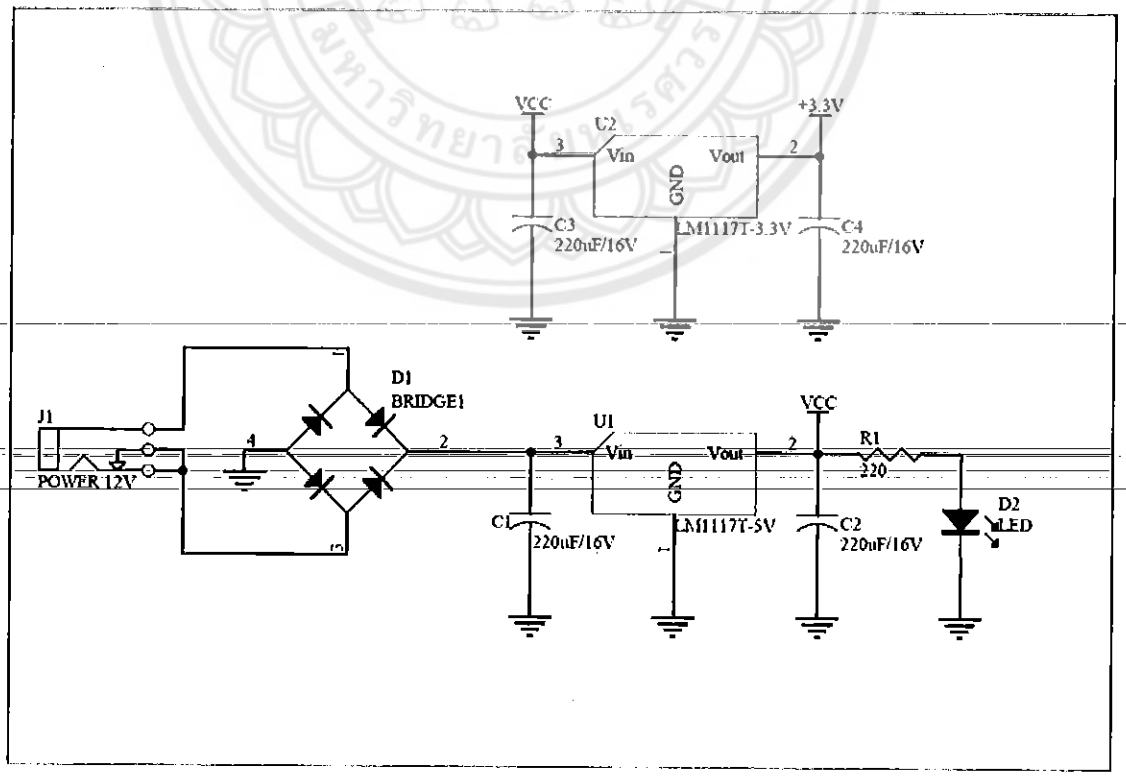
รูปที่ 3.1 แสดงรูปการออกแบบของฮาร์ดแวร์ (Hardware)

จากรูปที่ 3.1 จะเป็นการออกแบบการทำงานของอุปกรณ์ฮาร์ดแวร์ (Hardware) ในการออกแบบเครื่องเล่นเอ็มพีสาม (MP3) นี้ จุดประสงค์ของโครงงานก็เพื่อนที่จะให้เครื่องเล่น MP3

ต้องกำหนดส่วนรับข้อมูลไว้สองทาง ไมโครคอนโทรเลอร์ และส่วนแปลงสัญญาณดิจิตอลให้เป็นสัญญาณอนาลอคหรือให้เป็นสัญญาณเสียงเพื่อที่จะหูของมนุษย์สามารถรับฟังได้ โดยเราจำแนกเครื่องเล่นเอ็มพีสาม (MP3) ออกเป็นส่วนๆดังนี้

3.1.1 แหล่งจ่ายไฟ

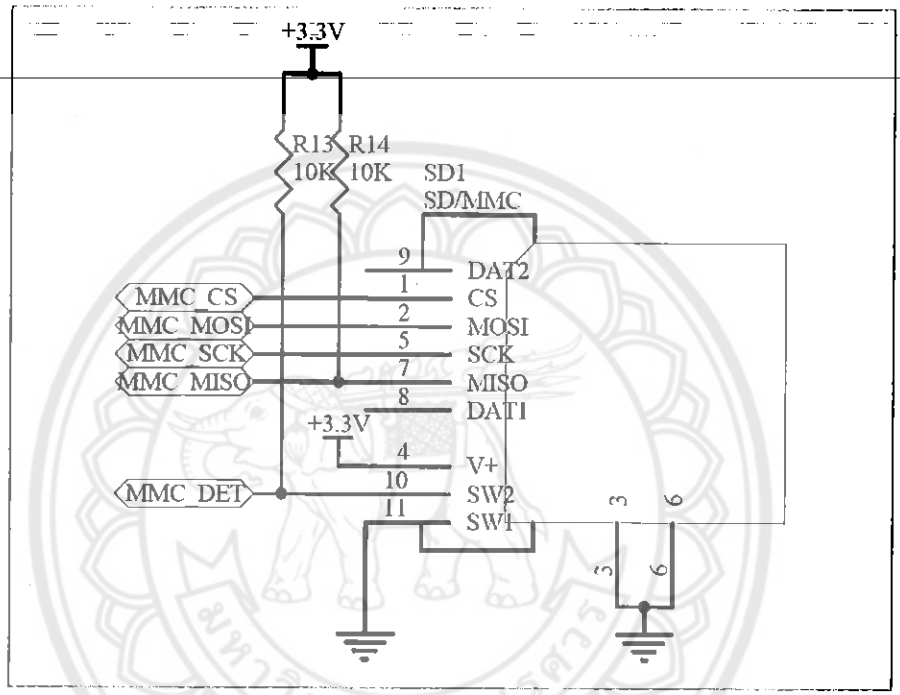
เครื่องเล่นเพลงเอ็มพีสามจะทำงานได้จะต้องจ่ายแรงดันไฟฟ้าให้ให้กับเครื่องเล่นก่อน ซึ่งเครื่องเล่นเพลงเอ็มพีสาม มีอุปกรณ์อิเล็กทรอนิกส์คือไมโครคอนโทรเลอร์ หน้าจอ LCD 6610 สวิตซ์ ส่วนติดต่อกับกับอุปกรณ์สื่อบันทึกข้อมูลเอสดีการ์ด และส่วนติดต่อกับสื่อบันทึกข้อมูลยูเอสบี ซึ่งตัวไมโครคอนโทรเลอร์นั่นใช้แรงดันไฟที่ 2.7V – 5.5V แต่ในส่วนยูเอสบีนั่นใช้แรงดันไฟฟ้าที่ -5V และ +5V ที่ขาสัญญาณของยูเอสบี ฉะนั้นจึงต้องออกแบบในส่วนของแหล่งจ่ายไฟให้แปลงแรงดันไฟฟ้าให้เหมาะสมกับอุปกรณ์ เพื่อให้อุปกรณ์แต่ละตัวสามารถทำงานได้และไม่เกิดการเสียหายจากการใช้งาน ซึ่งได้ออกแบบตามรูป 3.1 ซึ่งเมื่อจ่ายไฟเข้ามาจะผ่านบริดจ์ไดโอดเพื่อแปลงไฟฟ้ากระแสสลับให้เป็นไฟฟ้ากระแสตรง เมื่อแปลงแรงดันไฟเรียบร้อยแล้วจากผ่านมายังตัวไอซี LM ซึ่งเราสามารถเลือกเปลี่ยนเบอร์ไอซีตามความต้องการแรงดันเอาท์พุตได้ ซึ่งเราจะแยกไว้สองตัวเพื่อที่จะให้ได้แรงดันไฟเอาท์พุตตามต้องการ เพื่อที่จะจ่ายให้กับไอซีของเรา ตามรูป 3.2 นี้



รูปที่ 3.2 แสดงรูปวงจรแหล่งจ่ายไฟ

3.1.2 ส่วนติดต่อกับ SD CARD

จากรูป 3.3 เป็นส่วนของสล๊อต SD CARD ของเครื่องเล่น MP3 ที่ออกแบบให้อ่านไฟล์เพลงได้จากสื่อบันทึกข้อมูล SD CARD โดยจะต้องจ่ายแรงดันไฟฟ้าที่ 3.3V เพื่อให้ SD CARD สามารถทำงานได้ ซึ่งการทำงานนั้นในส่วนของการรับและส่งข้อมูลนั้นจะอยู่ที่ขา 2 และขา 7 โดยที่กำหนดจังหวะการทำงานโดยขา 5 ซึ่งปกติแล้วขาสัญญาณของสื่อบันทึกข้อมูลนั่นจะมีเพียง 9 ขา แต่สำหรับ Socket นั่นจะมีทั้งหมด 11 ขา โดยขาแต่ละขามีหน้าที่ตามตาราง 3.1 ดังนี้



รูปที่ 3.3 แสดงวงจรส่วนติดต่อกับ SD CARD

ตารางที่ 3.1 ตารางแสดงขาการทำงานของ SD CARD

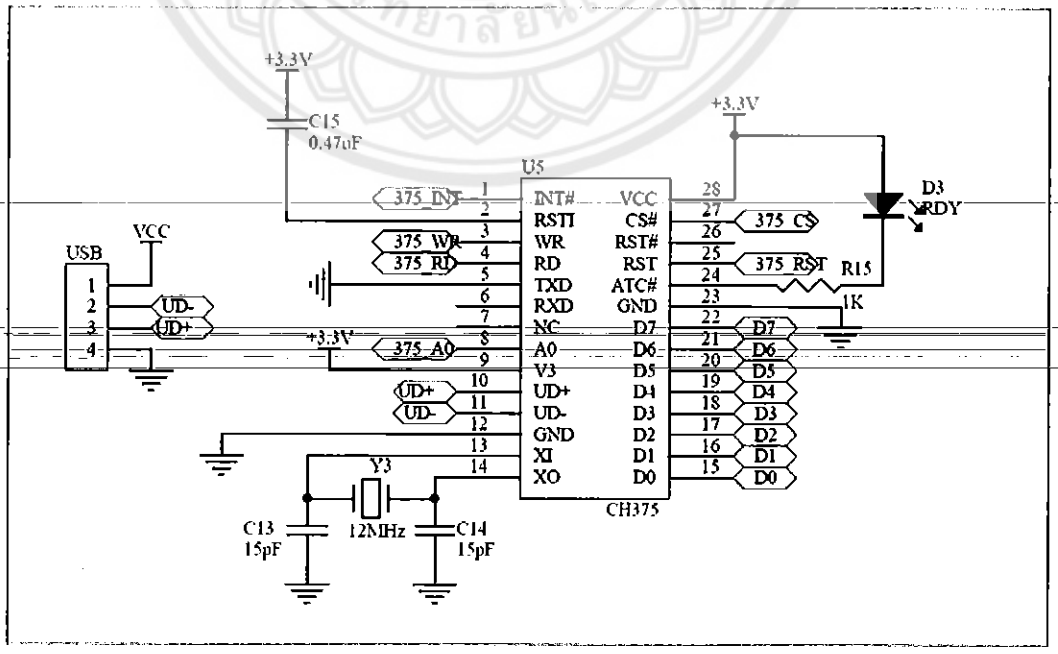
Pin No.	ทำหน้าที่
1	ขาสัญญาณ CS เป็นขาเส้นทางของข้อมูล
2	ทำหน้าที่เป็น MOSI ใช้ในการส่งข้อมูลออก
3	ขากราวน์ GND
4	ขาไฟเลี้ยง
5	ทำหน้าที่เป็นขา SCK ขาสัญญาณนาฬิกา
6	ขากราวน์ GND
7	ทำหน้าที่เป็น MISO ใช้ในการรับข้อมูล

ตารางที่ 3.1 ตารางแสดงขาการทำงานของ SD CARD [ต่อ]

Pin No.	ทำหน้าที่
8	ขาสัญญาณข้อมูล Data Line [1]
9	ขาสัญญาณข้อมูล Data Line [2]
10	ขาตรวจสอบว่ามี SD CARD หรือไม่
11	ขากราวน์ GND

3.1.3 ส่วนติดต่อกับ USB

จากรูป 3.4 เป็นส่วนของการติดต่อกับสื่อบันทึกข้อมูลยูเอสบี (USB) ซึ่งต้องใช้ซ็อกเก็ตไว้เพื่อรองรับการใช้งาน สัญญาณนั้นจะมีเพียงแค่สัญญาณแรงดันไฟบวกและสัญญาณแรงดันไฟลบซึ่งเป็นแบบ Twist pair โดยที่จะต้องจ่ายไฟเลี้ยงและมีขากราวน์เพื่อให้ USB นั่นทำงานสื่อสารได้ ซึ่งสัญญาณที่ส่งเข้ามานั้นจะส่งมาในลักษณะส่งสัญญาณความต่างของระดับแรงดัน กรณีในการส่งสัญญาณ "0" สัญญาณที่ขา UD- จะมีระดับแรงดันที่สูงกว่าขา UD+ กรณีในการส่งสัญญาณ "1" สัญญาณที่ขา UD- จะมีระดับแรงดันที่ต่ำกว่าขา UD+ และเราได้ใช้ไอซี CH375 เพื่อมารองรับการใช้งานสื่อบันทึกข้อมูล USB โดยที่ไอซีจะตรวจจับสื่อบันทึกข้อมูล USB โดยอัตโนมัติเมื่อมีการเสียบของ USB เข้ามา ซ้ำยังรองรองไฟล์ระบบ FAT12/16/32 และสนับสนุน USB 2.0 ซึ่งเป็นมาตรฐานของการส่งสัญญาณในความเร็ว 480 Mb ต่อวินาที ซึ่งมีรายละเอียดขอขาต่างๆดังตารางที่ 3.2



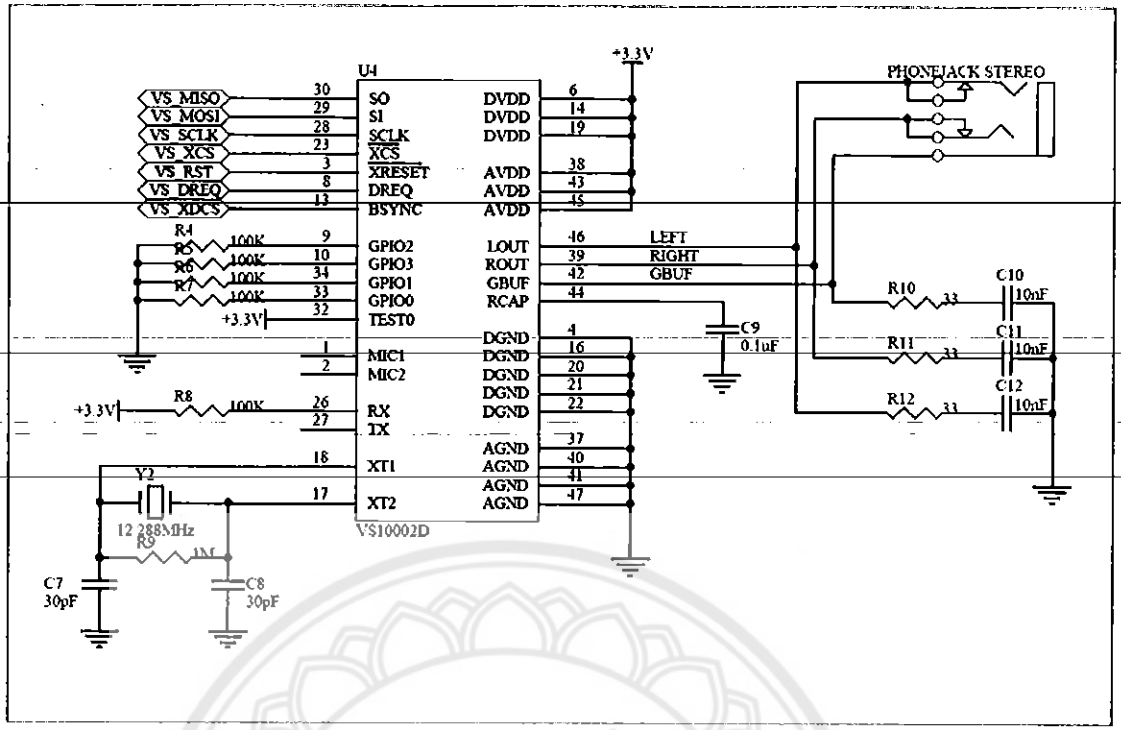
รูปที่ 3.4 แสดงวงจรส่วนติดต่อกับ USB โดยผ่าน CH375

ตารางที่ 3.2 ตารางแสดงขาที่ใช้งานการของ CH375

Pin No.	หน้าที่
1	ขาอินเตอร์รัปต์
2	ขารีเซตอินพุตโดยจ่ายไฟเลี้ยงให้ที่ขาสัญญาณ
3	ขาอินพุตเขียนข้อมูลสัญญาณแฟลช
4	ขาอินพุตอ่านข้อมูลสัญญาณแฟลช
5	กราวน์ GND
8	ขาคำสั่งในการอ่านหรือเขียนข้อมูลอินพุตลงใน Address
9	ไฟเลี้ยง
10	ขาข้อมูล USB สัญญาณบวก
11	ขาข้อมูล USB สัญญาณลบ
12	กราวน์ GND
13	ขาอินพุตสำหรับ crystal
14	ขาเอาต์พุตสำหรับ crystal
15-22	ขาสัญญาณ Data bus แบบ Bi-Directional
23	กราวน์ GND
25	ขารีเซต

3.1.4 ส่วนแปลงสัญญาณดิจิตอลเป็นสัญญาณอะนาลอก

ในส่วนของการแปลงสัญญาณดิจิตอลเป็นสัญญาณอนาลอกของเครื่องเล่น MP3 นั้น จะเป็นส่วนที่เอาไว้แปลงสัญญาณที่ได้รับมาจากสื่อบันทึกข้อมูลเอสดีการ์ด (SD CARD) หรือสื่อบันทึกข้อมูลยูเอสบี (USB) จากรูปแบบไฟล์เอ็มเปกเลเยอร์สามหรือจากสัญญาณดิจิตอลไปเป็นสัญญาณอนาลอกเพื่อให้สัญญาณที่ออกมานั้นเป็นสัญญาณเสียงที่หูของคนเราสามารถรับฟังได้ เครื่องเล่นเพลงเอ็มพีสามเครื่องนี้ได้ใช้ไอซีเบอร์ VS1002D ซึ่งเป็นไอซีประเภท MP3 audio decoder โดยสามารถเลือกใช้เบอร์ไอซีได้ตามสะดวกหรือตามที่สามารถหาซื้อได้ตามท้องตลาดไม่ว่าจะเป็น VS1003B VS1011E โดยไอซีประเภทนี้จะมีขาการใช้งานที่ใกล้เคียงกัน สามารถดูได้ในดาต้าชีท โดยรูปวงจรของส่วนการแปลงสัญญาณดิจิตอลเป็นสัญญาณอนาลอกนั้นจะเป็นตามรูปที่ 3.5 และมีรายละเอียดของขาสัญญาณแสดงตามตาราง 3.3



รูปที่ 3.5 แสดงวงจรการต่อขาของ VS1002D

ตารางที่ 3.3 ตารางแสดงขาที่ใช้งานการของ VS1002D

Pin No.	Pin Name	หน้าที่ในวงจร
3	XRESET	ขารีเซตสัญญาณอะซิงโครนัสที่ขอบขาลง
4	DGND0	ในวงจรต่อกราวน์ GND
6	DVDD0	ในวงจรต่อไฟเลี้ยง
8	DREQ	ขาข้อมูลรีเควสเอาต์พุต
9	GPIO2	ในวงจรต่อกราวน์
10	GPIO3	ในวงจรต่อกราวน์
13	BSYNC	ขาเลือกข้อมูล / byte sync
14	DVDD1	ในวงจรต่อไฟเลี้ยง
16	DGND1	ในวงจรต่อกราวน์ GND
17	XTAL1	Crystal output, Crystal Input
18	XTAL2	Crystal output, Crystal Input
19	DVDD2	ในวงจรต่อไฟเลี้ยง
20	DGND2	ในวงจรต่อกราวน์ GND
21	DGND3	ในวงจรต่อกราวน์ GND

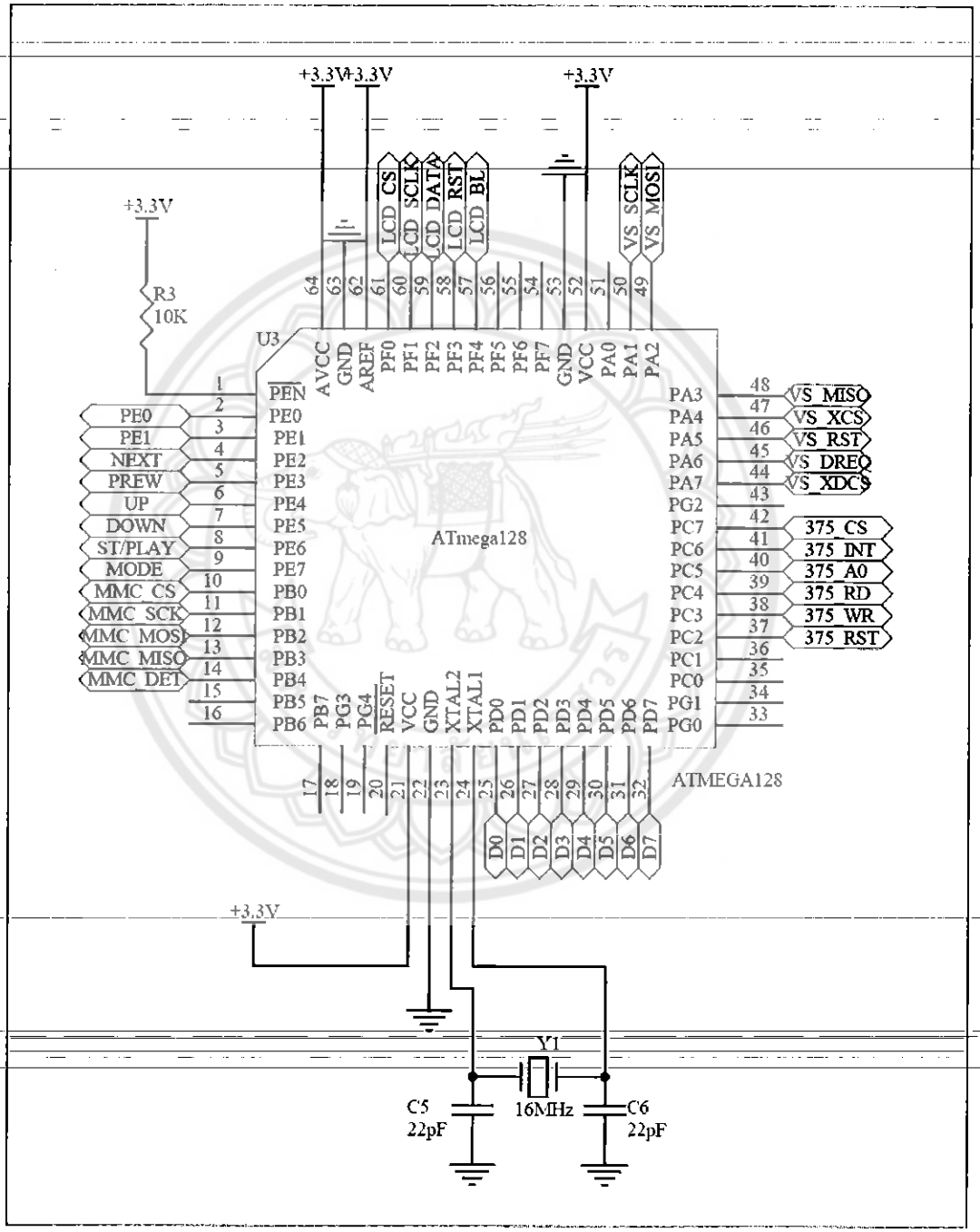
ตารางที่ 3.3 ตารางแสดงขาที่ใช้งานการของ VS1002D [ต่อ]

Pin No.	Pin Name	หน้าที่ในวงจร
22	DGND4	ในวงจรต่อกราวน์ GND
23	XCS	ขาเลือกข้อมูลอินพุตทำงานที่ขอบขาลง
26	RX	ขารับสัญญาณ UART / ในวงจรต่อไฟเลี้ยง
28	SCLK	ขาสัญญาณนาฬิกา
29	SI	ขาอินพุตอนุกรม
30	SO	ขาเอาท์พุตอนุกรม
33	GPIO0	ในวงจรต่อกราวน์
34	GPIO1	ในวงจรต่อกราวน์
37	AGND0	ในวงจรต่อกราวน์ GND
38	AVDD0	ในวงจรต่อไฟเลี้ยง
39	RIGHT	Right channel output
40	AGND1	ในวงจรต่อกราวน์ GND
41	AGND2	ในวงจรต่อกราวน์ GND
42	GBUF	ทำงานเสมือนกราวน์สำหรับเสียงเอาท์พุต
43	AVDD1	ในวงจรต่อไฟเลี้ยง
44	RCAP	ในวงจรต่อกราวน์
45	AVDD2	ในวงจรต่อไฟเลี้ยง
46	LEFT	Left channel output
47	AGBD3	ในวงจรต่อกราวน์ GND

3.1.5 ไมโครคอนโทรเลอร์ AVR ATmega128

เครื่องเล่นเอ็มพีสาม (MP3) เครื่องนี้ได้ใช้ไมโครคอนโทรเลอร์ของตระกูล AVR ของบริษัท ATMEL เบอร์ ATmega128 ในการติดต่อสื่อสารกับอุปกรณ์และไอซีต่อๆภายในเครื่องนี้ MP3 นี้ ซึ่งคุณสมบัติของไอซีเบอร์นี้นั้นได้แสดงไว้ในบทที่ 2 แล้ว ฉะนั้นเราจะมาดูรูปวงจรในการต่อขาแต่ละขาของไอซี ATmega128 ว่าขาแต่ละขานั้นต่อกับส่วนใดและต่อกับอุปกรณ์ใดบ้าง ดังแสดงดังรูปที่ 3.5 โดยที่ได้ออกแบบการติดต่อไว้โดยส่วนของพอร์ต A หรือ PA ติดต่อกับตัวแปลงสัญญาณดิจิตอลเป็นสัญญาณเสียง ส่วนของพอร์ต B หรือ PB เพื่อติดต่อกับสื่อบันทึกข้อมูล SD CARD ส่วนของพอร์ต C หรือ PC เพื่อติดต่อกับไอซี CH375 เพื่อแปลงสัญญาณแรงดันจากสื่อบันทึกข้อมูล USB ให้เป็นบิตข้อมูล ส่วนของพอร์ต D หรือ PD เพื่อติดต่อกับไอซี CH375

เพื่อรับข้อมูลดิจิตอลขนาด 8 บิต เพื่อที่จะส่งต่อไปส่วนของการแปลงสัญญาณดิจิตอลเป็นสัญญาณเสียงต่อไป ส่วนของพอร์ต E หรือ PE เพื่อการใช้งานในการควบคุมเครื่องเล่น MP3 ผ่านการกดสวิตซ์ และส่วนของพอร์ต F หรือ PF เพื่อติดต่อกับหน้าจอแสดงผล LCD6610 โดยรูปการต่อขานั้นจะเป็นตามดังรูป 3.6



รูปที่ 3.6 แสดงวงจรการต่อขาของ AVR ATmega128

และส่วนหน้าที่ของขาสัญญาณแต่ละขานั้นจะแยกออกเป็นแต่พอร์ต ซึ่งแต่ละพอร์ตนั่นจะติดต่อกับอุปกรณ์อิเล็กทรอนิกส์ ได้แก่ส่วนติดต่อกับไอซี VS1002D ซึ่งใช้ขาพอร์ต PA อธิบายรายละเอียดของขาสัญญาณได้ดังตารางที่ 3.4, ส่วนติดต่อกับ SD CARD ซึ่งใช้ขาพอร์ต PB อธิบายรายละเอียดของขาสัญญาณได้ดังตารางที่ 3.5, ส่วนติดต่อกับไอซี CH375 ซึ่งใช้ขาพอร์ต PC และขาพอร์ต PD อธิบายรายละเอียดของขาสัญญาณได้ดังตารางที่ 3.6 และตารางที่ 3.7 ตามลำดับส่วนติดต่อกับสวิตซ์ควบคุมซึ่งใช้ขาพอร์ต PE อธิบายรายละเอียดของขาสัญญาณได้ดังตารางที่ 3.8 และส่วนติดต่อกับหน้าจอ LCD 6610 ซึ่งใช้ขาพอร์ต PF อธิบายรายละเอียดของขาสัญญาณได้ดังตารางที่ 3.9 ดังนี้

ตารางที่ 3.4 แสดงขา PORT A (PA0-PA7)

ขาพอร์ต	รายละเอียดขา
PA7	เชื่อมต่อกับ VS1002D ที่ขา XDCS ซึ่งเป็นขาสัญญาณแสดงข้อมูลอะซิงโครไนท์
PA6	เชื่อมต่อกับ VS1002D ที่ขา DREQ ซึ่งเป็นขาร้องขอข้อมูลเอาท์พุต
PA5	เชื่อมต่อกับ VS1002D ที่ขา XRESET ซึ่งเป็นขาสัญญาณอะซิงโครไนท์
PA4	เชื่อมต่อกับ VS1002D ที่ขา XCS ซึ่งเป็นขาเลือกอินพุต
PA3	เชื่อมต่อกับ VS1002D ที่ขา MISO ซึ่งเป็นขาเอาท์พุตอนุกรม
PA2	เชื่อมต่อกับ VS1002D ที่ขา MOSI ซึ่งเป็นขาอินพุตอนุกรม
PA1	เชื่อมต่อกับ VS1002D ที่ขา SCLK ซึ่งเป็นขาสัญญาณนาฬิกา

ตารางที่ 3.5 แสดงขา PORT B (PB0-PB7) ในที่นี้ใช้เพียง PB0-PB4

ขาพอร์ต	รายละเอียดขา
PB4	เชื่อมต่อกับ SD CARD ที่ขา DET ซึ่งเป็นขาเปรียบเทียบและสร้างสัญญาณเอาท์พุต PWM สำหรับไทเมอร์/เคาน์เตอร์ 0 (ขนาด 8 บิต)
PB3	เชื่อมต่อกับ SD CARD ที่ขา MISO ซึ่งเป็นขาสัญญาณ อินพุตของมาสเตอร์/เอาท์พุตสเลฟ
PB2	เชื่อมต่อกับ SD CARD ที่ขา MOSI ซึ่งเป็นขาสัญญาณ เอาท์พุตมาสเตอร์/อินพุตสเลฟ ซึ่งเชื่อมต่อกับ SD CARD ที่ขา MOSI
PB1	เชื่อมต่อกับ SD CARD ที่ขา SCK ซึ่งเป็นขาสัญญาณนาฬิกา
PB0	เชื่อมต่อกับ SD CARD ที่ขา CS (Chip Select , Active low)

ตารางที่ 3.6 แสดงขา PORT C (PC0-PC7) ในที่นี้ใช้เพียง PC2-PC7

ขาพอร์ต	รายละเอียดขา
PC7	เชื่อมต่อกับ CH375 ที่ขา CS (Chip Select , Active low)
PC6	เชื่อมต่อกับ CH375 ที่ขา INT ซึ่งเป็นขา Interrup ร้องขอข้อมูลเอาท์พุต
PC5	เชื่อมต่อกับ CH375 ที่ขา AO ซึ่งเป็นขา address ส่งข้อมูลอินพุตให้กับขาข้อมูล
PC4	เชื่อมต่อกับ CH375 ที่ขา RD ซึ่งเป็นขาอ่านข้อมูลอินพุต
PC3	เชื่อมต่อกับ CH375 ที่ขา WR ซึ่งเป็นขาที่เขียนข้อมูลอินพุต
PC2	เชื่อมต่อกับ CH375 ที่ขา RST ซึ่งเป็นขาไว้รีเซตข้อมูลเอาท์พุต

ตารางที่ 3.7 แสดงขา PORT D (PD0-PD7)

ขาพอร์ต	รายละเอียดขา
PD7	เชื่อมต่อกับ CH375 ที่ขา D7 ซึ่งเป็นขาส่งข้อมูลแบบ 8 บิต
PD6	เชื่อมต่อกับ CH375 ที่ขา D6 ซึ่งเป็นขาส่งข้อมูลแบบ 8 บิต
PD5	เชื่อมต่อกับ CH375 ที่ขา D5 ซึ่งเป็นขาส่งข้อมูลแบบ 8 บิต
PD4	เชื่อมต่อกับ CH375 ที่ขา D4 ซึ่งเป็นขาส่งข้อมูลแบบ 8 บิต
PD3	เชื่อมต่อกับ CH375 ที่ขา D3 ซึ่งเป็นขาส่งข้อมูลแบบ 8 บิต
PD2	เชื่อมต่อกับ CH375 ที่ขา D2 ซึ่งเป็นขาส่งข้อมูลแบบ 8 บิต
PD1	เชื่อมต่อกับ CH375 ที่ขา D1 ซึ่งเป็นขาส่งข้อมูลแบบ 8 บิต
PD0	เชื่อมต่อกับ CH375 ที่ขา D0 ซึ่งเป็นขาส่งข้อมูลแบบ 8 บิต

ตารางที่ 3.8 แสดงขา PORT E (PE0-PE7) ในที่นี้ใช้ PE2-PE7

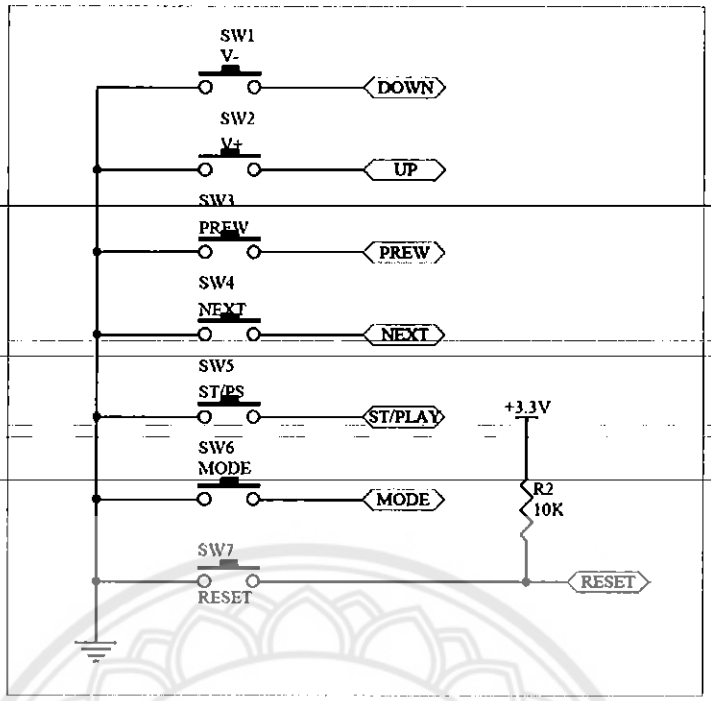
ขาพอร์ต	รายละเอียดขา
PE7	ขาเชื่อมต่อกับ Switch ในโหมดการทำงานแบบเลือกโหมด
PE6	ขาเชื่อมต่อกับ Switch ในโหมดการทำงาน Play/Stop
PE5	ขาเชื่อมต่อกับ Switch ในโหมดการทำงานลดระดับเสียง
PE4	ขาเชื่อมต่อกับ Switch ในโหมดการทำงานเพิ่มระดับเสียง
PE3	ขาเชื่อมต่อกับ Switch ในโหมดการทำงานแบบ Previous
PE2	ขาเชื่อมต่อกับ Switch ในโหมดการทำงานแบบ Next

ตารางที่ 3.9 แสดงขา PORT F (PF0-PF7) ในที่นี้ใช้ PF0-PF4

ขาพอร์ต	รายละเอียดขา
PF4	เชื่อมต่อกับ LCD ที่ขา BL ซึ่งเป็นขาควบคุมสัญญาณไฟ LED Back-Ligth
PF3	เชื่อมต่อกับ LCD ที่ขา RST ซึ่งเป็นขารีเซตของจอ LCD ทำงานที่ลอจิก 0
PF2	เชื่อมต่อกับ LCD ที่ขา DATA ซึ่งเป็นข้อมูลอนุกรมของจอ LCD
PF1	เชื่อมต่อกับ LCD ที่ขา SCLK ซึ่งเป็นสัญญาณอนุกรมของจอ LCD
PF0	เชื่อมต่อกับ LCD ที่ขา CS (Chip Select , Active low) สำหรับเปิด-ปิดการรับส่งข้อมูล

3.1.6 Switch ควบคุม

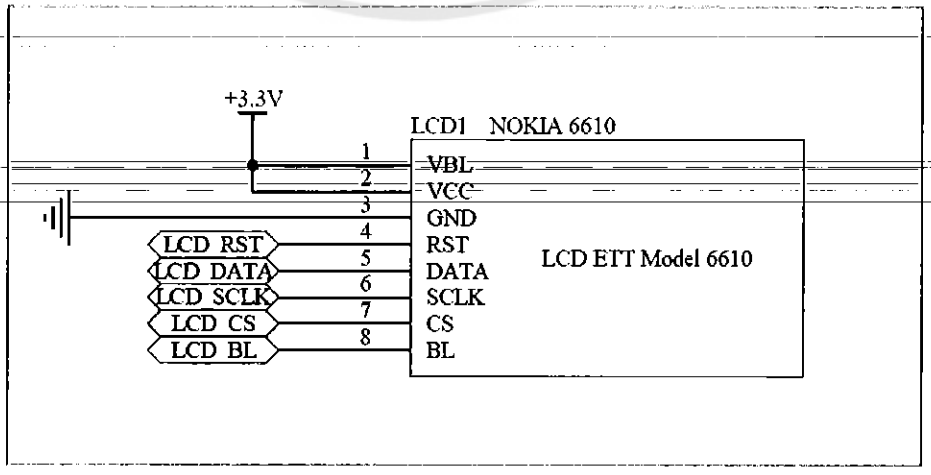
เครื่องเล่นเพลงเอ็มพีสาม (MP3) เครื่องนี้ได้ออกแบบให้ทำงานจากการควบคุมผ่านสวิตช์ สวิตช์แต่ละตัวก็จะทำหน้าที่ต่างกันไป สวิตช์ที่ 1 ของตัวเครื่องเล่นเอ็มพีสาม (MP3) นั่นจะทำหน้าที่ Reset ระบบ เพื่อให้เครื่องเล่นเอ็มพีสาม (MP3) กลับมาทำงานในส่วนของการแสดงผลเริ่มต้นเพื่อที่จะให้เลือกอ่านไฟล์เพลงจากสื่อบันทึกข้อมูลเอสดีการ์ด (SD CARD) หรือสื่อบันทึกข้อมูลยูเอส (USB) ซึ่งส่วนนี้ใช้สวิตช์ในการเลือกคือสวิตช์ "PREW" จะใช้เลือกอ่านไฟล์เพลงจากสื่อทันทึกข้อมูลเอสดีการ์ด และสวิตช์ "NEXT" จะใช้เลือกอ่านไฟล์เพลงจากสื่อทันทึกข้อมูลยูเอสบี สวิตช์ "MODE" ก็จะทำหน้าที่เลือกโหมดการทำงานการเล่นเพลง โดยมีโหมดการเล่นเพลงทั้งหมดสามโหมดด้วยกันได้แก่ โหมดการเล่นเพลงเดียว โหมดการเล่นเพลงทั้งหมดและโหมดเล่นเพลงแบบสลับ โดยที่การเปลี่ยนโหมดการเล่นเพลงจะใช้เพียงสวิตช์เพียงตัวเดียวใช้การใช้งานสวิตช์ที่ใช้ในการปรับระดับเสียงได้แก่ สวิตช์ "Volume Down" ก็ทำหน้าที่ปรับลดระดับเสียง สวิตช์ "Volume Up" ก็จะทำหน้าที่ปรับเพิ่มระดับเสียง และยังมีสวิตช์ "PREV" ที่นอกจากจะทำหน้าที่เพื่อเลือกอ่านข้อมูลจากสื่อบันทึกข้อมูลแล้วยังทำหน้าที่ย้อนเพลง สวิตช์ "NEXT" ก็เช่นกันนอกจากจะเลือกอ่านข้อมูลจากสื่อบันทึกข้อมูลแล้วยังทำหน้าที่เลื่อนเพลงไปข้างหน้า และสวิตช์ "PLAY/PAUSE" ก็จะทำหน้าที่สั่งการให้เครื่องเล่นเพลงเอ็มพีสามเล่นเพลงและหยุดเพลง โดยสวิตช์ต่างๆนี้จะต่อกับไมโครคอนโทรเลอร์ AVR ATmega 128 โดยสวิตช์แต่ละตัวจะกำกับหน้าที่เอาไว้และต่อตามรูปที่ 3.7



รูปที่ 3.7 แสดงวงจรการต่อสวิตช์ซึ่งจะต่อกับไมโครคอนโทรเลอร์

3.1.7 จอ LCD6610

จอ LCD6610 เป็นส่วนแสดงผลของเครื่องเล่นเอ็มพีสาม (MP3) ซึ่งได้เลือกหน้าจอของ ETT ซึ่งเป็นหน้าจอสำเร็จรูปและเป็นหน้าที่ที่แสดงผล GRAPHIC ขนาด 132 x 132 DOT หน้าจอมีความละเอียดของสี 4,096 สี เชื่อมต่อสัญญาณโดยตรงกับไมโครคอนโทรลเลอร์ได้ทั้งระบบไฟ 3.3V หรือ 5V ซึ่งการต่อหน้าจอ LCD กับไมโครคอนโทรเลอร์นั่นแสดงดังในรูปข้างบน ซึ่งรายละเอียดของขาสัญญาณต่างๆนั่นได้แสดงตามรูปที่ 3.8 และตารางที่ 3.10

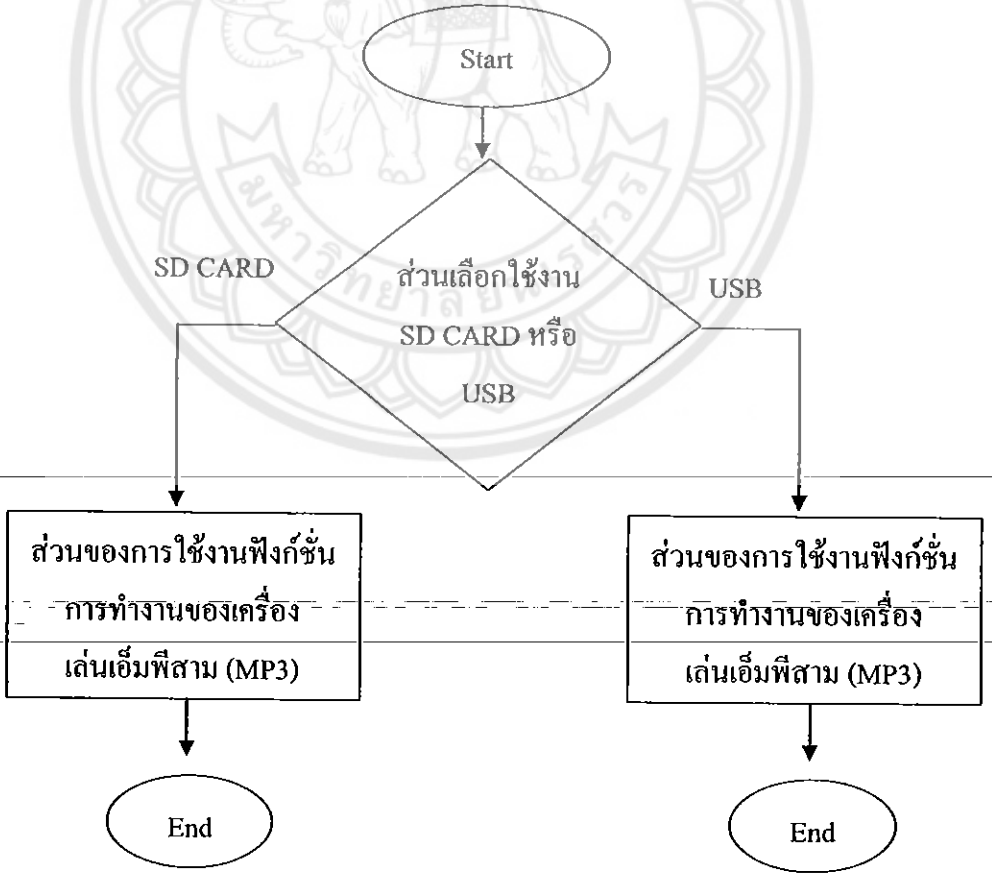


รูปที่ 3.8 แสดงวงจรการต่อขาของ LCD6610

ตารางที่ 3.10 แสดงหน้าที่ของขาสัญญาณจอ LCD6610

Pin No.	Pin Name	หน้าที่
1	VBL	ขาสัญญาณไฟบวก 3.3V
2	VCC	ไฟเลี้ยง 3.3V
3	GND	กราวน์
4	RST	ขารีเซตของหน้าจอ LCD
5	DATA	ขาข้อมูลอนุกรม
6	SCLK	ขาสัญญาณนาฬิกา
7	CS	ขาเลือกคำสั่งในการเลือก เปิด – ปิด การรับส่งข้อมูลของจอ LCD
8	BL	ขาสัญญาณควบคุมการ เปิด – ปิด LED Back - Light

3.2 การโปรแกรมทางด้านซอฟต์แวร์ (Software)



รูปที่ 3.9 แสดงรูปการออกแบบทางด้านซอฟต์แวร์ (Software)

3.2.1 การใช้งานฟังก์ชั่นการทำงานของเครื่องเล่นเอ็มพีสาม (MP3)

- การใช้งานฟังก์ชั่น Play/Puase

ในส่วนของโค้ดโปรแกรมจะเป็นส่วนของการแสดงผลทางหน้าจอและส่วนของโค้ดการใช้งานฟังก์ชั่น โดยมีรายละเอียดดังนี้

```
void lcd_print_play (unsigned char mode) // ฟังก์ชั่น lcd_print_play โดยกำหนดชนิดตัวแปลเป็น
                                         // เป็น unsigned char ช่วงข้อมูล 0-255
{
    switch (mode)   // ฟังก์ชั่น switch เพื่อเปรียบเทียบข้อมูล
    {
        case  PLAY : nokia_lcd_puts (0,1,"Now Playing",WHITE,BLACK);
        //กรณีที่ 1 "PLAY" ให้จอ LCD แสดงผลโดยเริ่มที่อักษรตัวที่ 0 บรรทัดที่ 1 แสดง
        // คำว่า "Now Playing" ตัวอักษรสีขาว พื้นหลังสีดำ
            break;  // คำสั่งหยุด
        case  PAUSE :   nokia_lcd_puts (0,1,"Pause",WHITE,BLACK);
        // กรณีที่ 2 "PAUSE" ให้จอ LCD แสดงผลโดยเริ่มที่อักษรตัวที่ 0 บรรทัดที่ 1
        // แสดงคำว่า "Pause" ตัวอักษรสีขาว พื้นหลังสีดำ
            break;  // คำสั่งหยุด
            default : break;  // ฟังก์ชั่นหยุด
    }
}
//-------------------------------------------------------------------------------//
if((PINE&STOP)==0)   //key PLAY/PAUSE //กรณีของฟังก์ชั่น PLAY/PAUSE ที่ขา PE
{
    blkeyflag = 0;   //ประกาศตัวแปลให้ blkeyflag = 0
    Delay(100);      //ดีเลย์
    if(!(PINE&STOP))        //ฟังก์ชั่นตรวจสอบเงื่อนไข
    {
        while(!(PINE&STOP)); //ฟังก์ชั่นวนลูปการทำงานถ้าเป็นไปตามฟังก์ชั่น
        if(flag)             //ตรวจสอบเงื่อนไข
        {
            flag=0; //ประกาศตัวแปล flag=0
            if(lrc == 0 || HanziEnable == 0) //ถ้า lrc เท่ากับ 0 หรือ HanziEnable
```

```
                                        //เท่ากับ 0
                {
                    lcd_print_play (PAUSE); // lcd_print_play จะไปทำคำสั่งใน
                                            //กรณี PAUSE
                }
            }
            else            //ตรวจสอบเงื่อนไข
            {
                flag=1; //ประกาศตัวแปล flag=1
                if(lrc == 0 || HanziEnable == 0) //ถ้า lrc เท่ากับ 0 หรือ HanziEnable
                                                 //เท่ากับ 0
                {
                    lcd_print_play (PLAY); // lcd_print_play จะไปทำคำสั่งใน
                                           //กรณี PAUSE
                }
                }
                Delay(1000); //ดีเลย์
                }
        }
```

- การใช้งานฟังก์ชั่นเลือกโหมดการเล่นเพลง

ในส่วนของโค้ดโปรแกรมจะเป็นส่วนของการแสดงผลทางหน้าจอและส่วนของโค้ดการใช้งานฟังก์ชั่นมีดังนี้

```
void check (uint8_t mode)        // ฟังก์ชั่น check ขนาด 8 บิต
{
    switch(mode)                 // ฟังก์ชั่น switch เพื่อเปรียบเทียบข้อมูล
    {
        case REPET_ALL:nokia_lcd_puts (0,4,"Mode Repeat All",YELLOW,BLACK);
        //กรณีที่ 1 “REPET_ALL” ให้จอ LCD แสดงผลโดยเริ่มที่อักษรตัวที่ 0 บรรทัดที่ 4
        //แสดงคำว่า “Mode Repeat All” ตัวอักษรสีเหลือง พื้นหลังสีดำ
                break;   // คำสั่งหยุด
        case REPET_ONE:nokia_lcd_puts (0,4,"ModeRepeatOne",YELLOW,BLACK);
```

```
                    //กรณีที่ 2 "REPET_ONE" ให้จอ LCD แสดงผลโดยเริ่มที่อักษรตัวที่ 0 บรรทัด
                    //ที่ 4 แสดงคำว่า "Mode RepeatOne" ตัวอักษรสีเหลือง พื้นหลังสีดำ
                            break;  // คำสั่งหยุด
                    case RANDOM:nokia_lcd_puts (0,4,"Mode Shuffle",YELLOW,BLACK);
                    //กรณีที่ 3 "RANDOM" ให้จอ LCD แสดงผลโดยเริ่มที่อักษรตัวที่ 0 บรรทัดที่ 4
                    // แสดงคำว่า "Mode Shuffle" ตัวอักษรสีเหลือง พื้นหลังสีดำ
                            break;  // คำสั่งหยุด
                    break; // คำสั่งหยุด
                }
        }
        //-------------------------------------------------------------------------------//
        else if(((!(PINE & MODE)) || blkeyflag) && keylen==0 ) //ฟังก์ชั่นเลือกโหมดถ้า
        {
                Delay(50);      //ดีเลย์
                if(blkeyflag == 0)      ////ถ้าเป็นไปตามเงื่อนไข
                {
                        blkeyflag = 1;  //ตัวแปล blkeyflag = 1
                        keycnt = 0;     //ตัวแปล keycnt = 0
                }
                else    //นอกเหนือจากนั้น
                {
                        if(!(PINE & MODE))    ////ถ้าเป็นไปตามเงื่อนไข(!(PINE & MODE))
                        {
                                keycnt ++;      //ตัวแปล keycnt เพิ่มค่าไปทีละหนึ่งค่า
                                keylen=0xfff;   //ให้ตัวแปล keylen เท่ากับ 111111111111
                                if(keycnt>5)    //ถ้า keycnt มากกว่า 5
                                {
                                        blkeyflag = 0;  //ตัวแปล blkeyflag = 0
                                        keylen=0xffff;  //ตัวแปล keycnt เท่ากับ 1111111111111111
                                }
                        }
                        else    //นอกเหนือจากนั้น
```

```
{
        blkeyflag = 0;    //ตัวแปล blkeyflag = 0
        keylen=0xffff;    //ตัวแปล keylen เท่ากับ 1111111111111111
        if(mode==REPET_ALL)//ถ้าเป็นตามเงื่อนไข mode = REPET_ALL
        {
            mode=REPET_ONE;     //mode เท่ากับ REPET_ONE
        }
        else if(mode==REPET_ONE)     //ถ้าเป็นไปตามเงื่อนไข
        {
            mode=RANDOM;        //mode เท่ากับ RANDOM
            srandom(((unsigned long)TCNT1)<<16);  //
        }
        else      //นอกเหนือจากนั้น
        {
            mode=REPET_ALL;     //mode เท่ากับ REPET_ALL
        }
        if(lrc == 0 || HanziEnable == 0)   //ถ้าเป็นไปตามเงื่อนไข lrc เท่ากับ 0
                                           //หรือ HanziEnable เท่ากับ 0
        {
            check (mode);   //ทำการ check mode
        }
    }
}
```

- การใช้งานฟังก์ชั่นลดระดับเสียง

ในส่วนของโค้ดโปรแกรมจะเป็นส่วนของการแสดงผลทางหน้าจอและส่วนของโค้ดการใช้งานฟังก์ชั่นมีดังนี้

```
void lcd_print_volume (uint8_t vol)        //ฟังก์ชั่น lcd_print_volume ขนาด 8 บิต
{
    unsigned char volset[16];          //ประกาศตัวแปล unsigned char เซตค่า 16 ค่า
        memset (volset,' ',sizeof volset);  //จะมีรูปแบบระดับเสียงที่ใช้ขณะนี้ต่อระดับ
                                            //เสียงทั้งหมด
        sprintf (volset, "Vol ");  //ประกาศให้ใช้รูปแบบเป็น “Vol” แทน volset
```

```
                memset (volset + 4, 0x7f, vol);     //เซตค่า array
                volset [15] = '\0';          //ให้ volset ในตัวแปรอาร์เร 15 ตัว เท่ากับ \0
                nokia_lcd_puts (0,7,volset,RED,WHITE); //หน้าจอ LCD แสดงผลโดยเริ่มที่
                                                       //อักษรตัวที่ 0 บรรทัดที่ 7 แสดงคำว่า
                                                       //"Vol" ตัวสีแดง พื้นหลังสีขาว
        }
        //-------------------------------------------------------------------------------------------//
        if(!(PINE & DOWN) && keylen==0)    //ถ้าเป็นไปตามเงื่อนไข (!(PINE & UP)) และ
                                           // keylen เท่ากับ 0
        {
                blkeyflag = 0;    //ให้ตัวแปล blkeyflag = 0
                if(!(PINE & DOWN))    //ถ้าเป็นไปตามเงื่อนไข (!(PINE & DOWN))
                {
                        keylen=200;              // ประกาศตัวแปล keylen=200
                        if (voltmp > 0)          //ถ้าเป็นไปตามเงื่อนไข voltmp > 0
                        {
                                voltmp -= 1;     //ให้ voltmp ลดค่าทีละหนึ่งค่า
                                voltemp = volume_values[voltmp];//ให้ voltemp เท่ากับ
                                                                // volume_values[voltmp];
                                voltemp <<= 8;  //ค่า voltemp เลื่อนบิตไปทางซ้าย 8 บิต
                                voltemp += volume_values[voltmp]; //voltemp บวกค่าหรือเท่ากับ
                                                                  // volume_values[voltmp];
                                VS1002D_SetVolume (voltemp); //เรียกใช้ VS1002D ในโหมดของ
                                                             //การ SetVolume
                                lcd_print_volume (voltmp);   //เรียกใช้ lcd_print_volume
                        }
                }
        }
```

- การใช้งานฟังก์ชั่นเพิ่มระดับเสียง

ในส่วนของโค้ดโปรแกรมจะเป็นส่วนของการแสดงผลทางหน้าจอและส่วนของโค้ด การใช้งานฟังก์ชั่นมีดังนี้

```
void lcd_print_volume (uint8_t vol)        //ฟังก์ชั่น lcd_print_volume ขนาด 8 บิต
```

```
{
        unsigned char volset[16];          //ประกาศตัวแปล unsigned char เซตค่า 16 ค่า
            memset (volset,' ',sizeof volset); //จะมีรูปแบบระดับเสียงที่ใช้ขณะนี้ต่อระดับ
                                               //เสียงทั้งหมด
            sprintf (volset, "Vol ");          //ประกาศให้ใช้รูปแบบเป็น "Vol" แทน volset
            memset (volset + 4, 0x7f, vol);    //เซตค่า array
            volset [15] = '\0';                //ให้ volset ในตัวแปรอาร์เร 15 ตัว เท่ากับ \0
            nokia_lcd_puts (0,7,volset,RED,WHITE);//หน้าจอ LCD แสดงผลโดยเริ่มที่
                                                  //อักษรตัวที่ 0 บรรทัดที่ 7 แสดงคำว่า
                                                  //"Vol" ตัวสีแดง พื้นหลังสีขาว
}
else if(!(PINE & UP) && keylen==0)     //ถ้าเป็นไปตามเงื่อนไข (!(PINE & UP)) และ
                                       // keylen เท่ากับ 0
{
    blkeyflag = 0;          //ประกาศตัวแปล blkeyflag = 0
    if(!(PINE & UP))        //ถ้าเป็นไปตามเงื่อนไข (!(PINE & UP))
    {
        keylen=200;     // ประกาศตัวแปล keylen=200
        if (voltmp < 12) //ถ้าเป็นไปตามเงื่อนไข voltmp < 12
        {
            voltmp += 1;     //ให้ voltmp บวกค่าทีละค่า
            voltemp = volume_values[voltmp]; //ให้ voltemp เท่ากับ
                                             //volume_values[voltmp];
            voltemp <<= 8;  // ค่า voltemp เลื่อนบิตไปทางซ้าย 8 บิต
            voltemp += volume_values[voltmp]; //voltemp บวกค่าหรือเท่ากับ
                                              // volume_values[voltmp];
            VS1002D_SetVolume (voltemp); //เรียกใช้ VS1002D ในโหมดของ
                                         //การ SetVolume
            lcd_print_volume (voltmp);       //เรียกใช้ lcd_print_volume
        }
    }
}
```

- การใช้งานฟังก์ชั่นเลื่อนเพลงไปข้างหน้า

ในส่วนของโค้ดโปรแกรมจะเป็นส่วนของการแสดงผลทางหน้าจอและส่วนของโค้ดการใช้งานฟังก์ชั่นมีดังนี้

```
else if(!(PINE & NEXT))          //ถ้าเป็นไปตามเงื่อนไข (!(PINE & NEXT))
{
        blkeyflag = 0;           //ประกาศตัวแปล blkeyflag = 0
        Delay(50000);            //ประกาศดีเลย์
        if(!(PINE & NEXT))       //ถ้าเป็นไปตามเงื่อนไข (!(PINE & NEXT))
        {
                while(!(PINE & NEXT));          //เงื่อนไขวนลูป (!(PINE & NEXT))
                songs++;         //ให้เพิ่มค่า songs ไปหนึ่งค่า
                if(songs > totalsongs)songs=1;  //ถ้า songs มากกว่า totalsongs , songs เท่ากับ 1
                {
                        free(buffer);    //ให้ฟรีบัฟเฟอร์หรือฟรีสำหรับพักข้อมูล
                        goto next;       //ไปทำงานฟังก์ชั่น next
                }
        }
}
```

- การใช้งานฟังก์ชั่นเลื่อนย้อนเพลงไปข้างหลัง

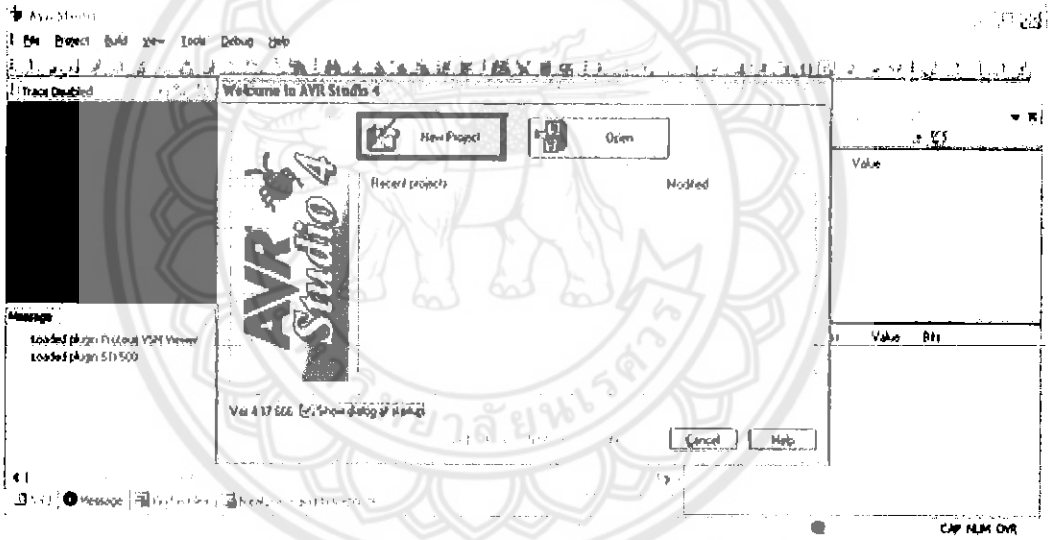
ในส่วนของโค้ดโปรแกรมจะเป็นส่วนของการแสดงผลทางหน้าจอและส่วนของโค้ดการใช้งานฟังก์ชั่นมีดังนี้

```
else if(!(PINE & PREV))//ถ้าเป็นไปตามเงื่อนไข (!(PINE & PREV))
{
        blkeyflag = 0;   //ประกาศตัวแปล blkeyflag = 0
        Delay(5000);     //ประกาศดีเลย์
        if(!(PINE & PREV))       //ถ้าเป็นไปตามเงื่อนไข (!(PINE & PREV))
        {
                while(!(PINE&PREV)); //เงื่อนไขวนลูป (!(PINE & PREV))
                if(songs == 1)songs=totalsongs;  //ถ้า songs เท่ากับ 1, songs เท่ากับ totalsongs
                else songs--;    //ให้ songs ลดค่าไปหนึ่งค่า
                {
                        free(buffer);    //ให้ฟรีบัฟเฟอร์หรือฟรีสำหรับพักข้อมูล
```

```
                goto next;          //ไปทำงานฟังก์ชั่น next
            }
        }
    }
}
```

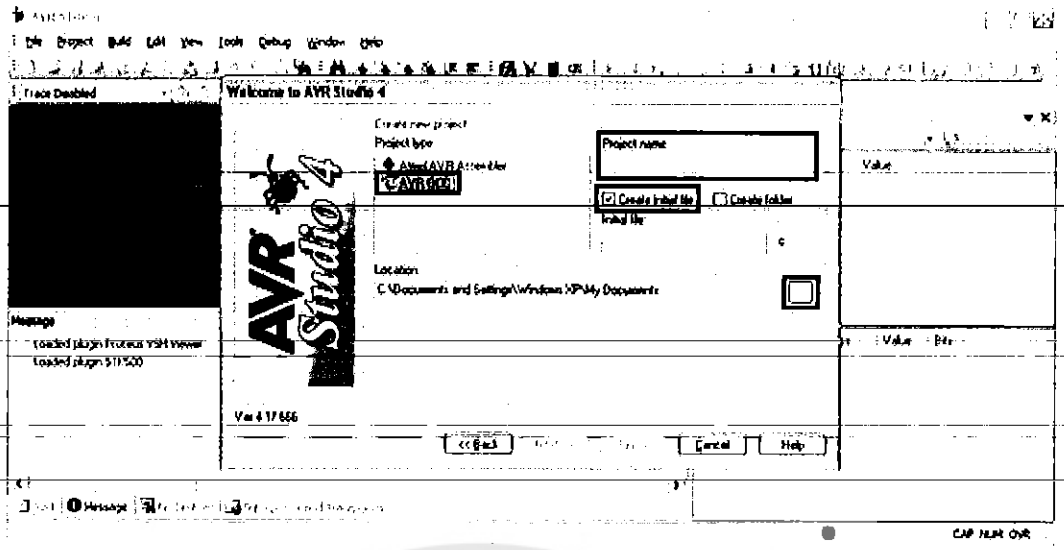
3.2.2 การเขียนโปรแกรมสำหรับเครื่องเล่นเอ็มพีสาม (MP3)

การเขียนโปรแกรมสำหรับเครื่องเล่น MP3 นั้นจะทำการเขียนโปรแกรมโดยผ่านโปรแกรม AVR Studio 4 ซึ่งเป็นฟรีโปรแกรมสามารถดาวน์โหลดได้จาก http://www.atmel.com เมื่อทำการลงโปรแกรมแล้วก็ทำการเขียนโปรแกรมโดยทำการสร้างโปรเจคขึ้นมาใหม่เพื่อสร้างโปรเจคและทำการเขียนโค๊ด ซึ่งหน้าต่างโปรแกรมจะแสดงดังรูปด่านล่างนี้ โดยที่เราทำการเลือกที่ New Project ตามรูป 3.10 นี้



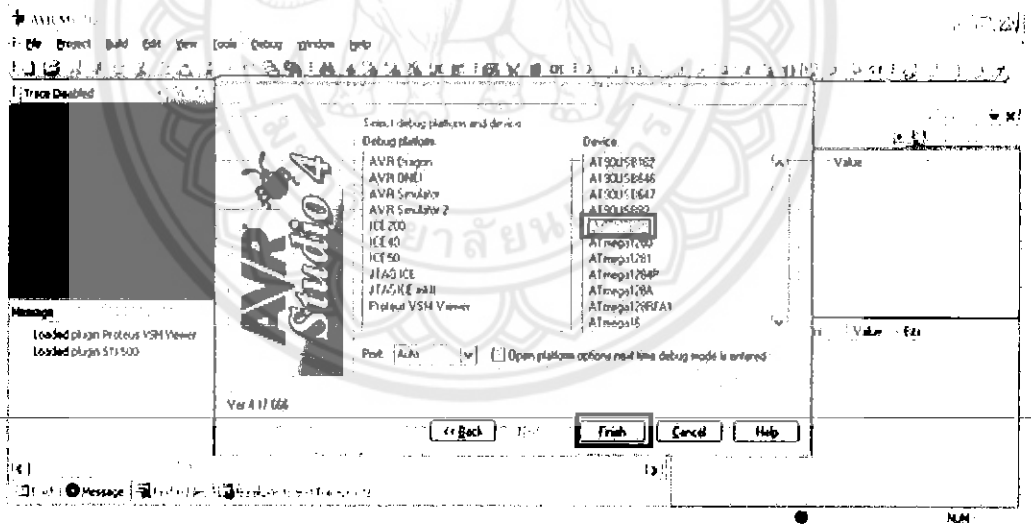
รูปที่ 3.10 แสดงหน้าต่างโปรแกรมการสร้างโปรเจค

หลังจากนั้นจะมีช่อง Project type ให้เราเลือก AVR GCC เพื่อที่จะเขียนโปรแกรมเป็นภาษา C จากนั้นก็ป้อนชื่อโปรเจคในช่อง Project name และเลือก Create initial file เพื่อสร้างไฟล์ภาษาซีให้กับโปรเจค เลือก Create folder เพื่อสร้างโฟลเดอร์ในการเก็บข้อมูลโปรเจค เลือก Location ที่ต้องการเก็บโปรเจค แล้วกด next ตามรูป 3.11



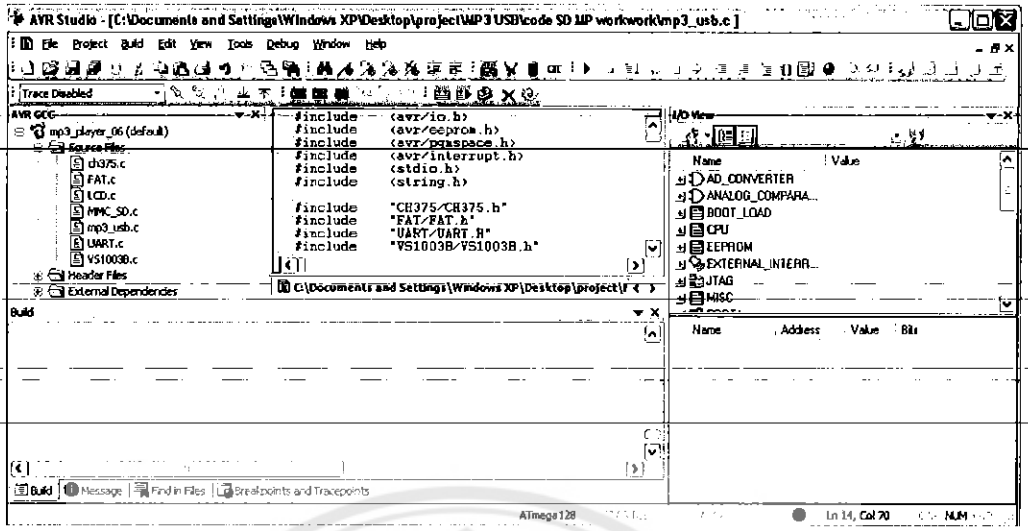
รูปที่ 3.11 แสดงหน้าต่างโปรแกรมเลือกฟังก์ชั่นการทำงาน

หลังจากนั่นให้เราทำการเลือกเบอร์ไอซีที่เราใช้ ซึ่งเครื่องเล่น MP3 เครื่องนี้ใช้ไอซีเบอร์ ATmega128 เมื่อเราเลือกเบอร์ไอซีแล้วให้กด Finish ตามรูป 3.12



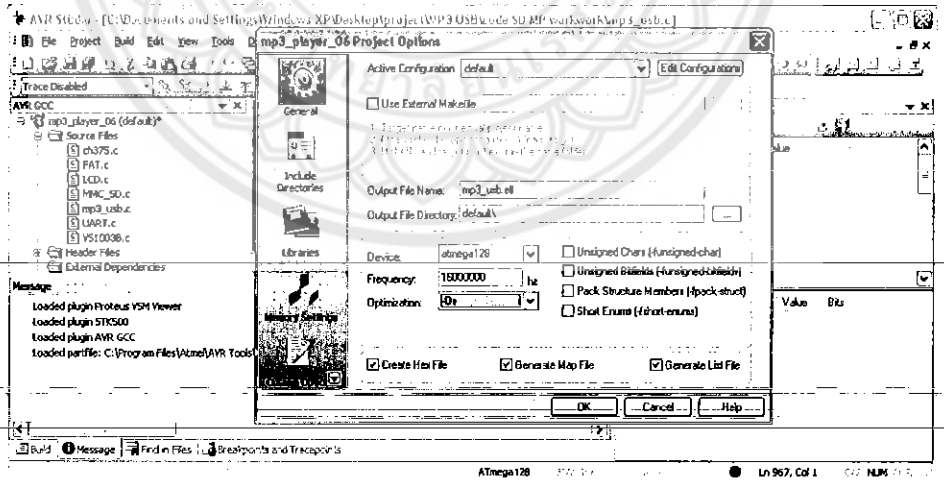
รูปที่ 3.12 แสดงหน้าต่างโปรแกรมเลือกเบอร์ไอซี

จากรูป 3.12 ขะแสดงหน้าต่างโปรแกรมด้านล่างนี้ก็พร้อมที่จะให้เราทำการเขียนโปรแกรมแล้ว ซึ่งในส่วนของการเขียนโค้ดโปรแกรมนั่นเราจะออกออกเป็นส่วนๆตามการใช้งาน คือ SD CARD, USB, LCD, CH375,VS1002D, USART และ FAT ซึ่งโค้ดโปรแกรมทั้งหมดนี้ได้ใส่ลงในตัว CD ของโปรเจคเรียบร้อยแล้ว



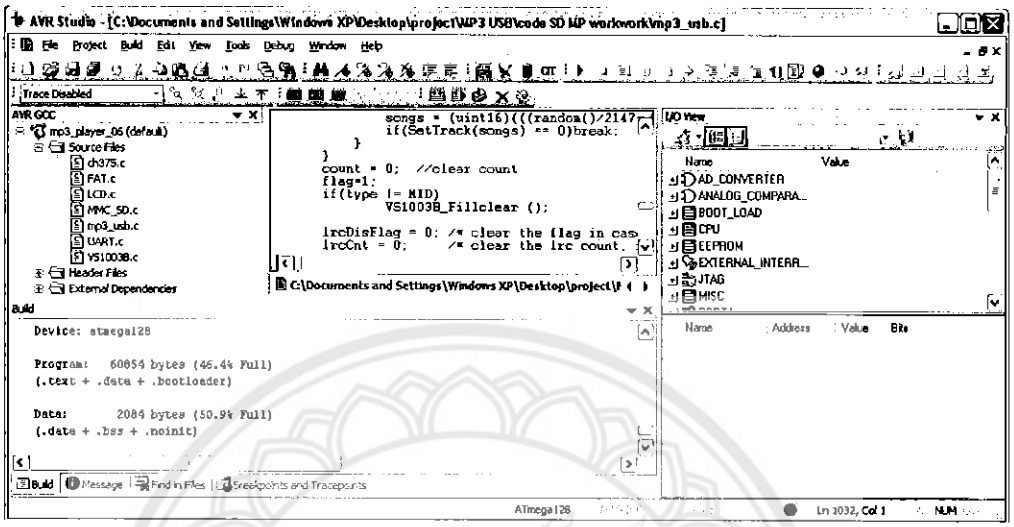
รูปที่ 3.13 แสดงหน้าต่างโปรแกรมที่เขียนโปรแกรมไว้ภายใน

ซึ่งเมื่อทำการเขียนโค้ดโปรแกรมเรียบร้อยแล้วเราจะต้องทำการเซตค่าของตัวไอซีโดยไปที่ Project เลือก Configuration Options โดยเราก็ทำการเลือก Device ตามเบอร์ไอซีของเรา Frequency ในที่นี้เราใช้ 16MHz และ Optimization ให้เลือกเป็น –Os และทำการติ๊กถูกที่ช่อง Create Hex File, Generate Map File และ Generate List File แล้วกด OK ตามรูป 3.14



รูปที่ 3.14 แสดงหน้าต่างโปรแกรมสำหรับการเซตค่า

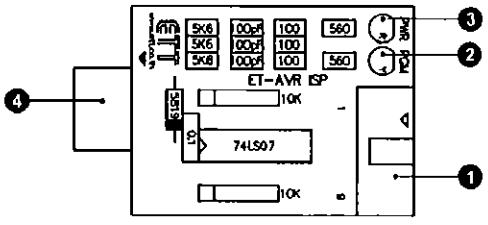
จากนั้นก็ทำการ Build หรือกด F7 ซึ่งโปรแกรมจะแสดงค่าต่างๆหรือแสดงส่วนโปรแกรมที่เขียนไว้ถูกหรือไม่อย่างไร แต่ถ้าโปรแกรมที่เขียนไว้ถูกต้องไม่มีการผิดพลาดใดๆ เกิดขึ้นก็จะแสดงคำว่า Build Succeeded และทำการ Compile โปรแกรม ตามรูป 3.15



รูปที่ 3.15 แสดงหน้าต่างโปรแกรมที่ทำการ Build and Compile

3.2.3 ขั้นตอนการเบิร์นโปรแกรม

ในส่วนของการเบิร์นโปรแกรมลงในเครื่องเล่น MP3 นั้นเราจะใช้โปรแกรม PonyProg2000 ซึ่งทางผู้ออกแบบได้เลือกสาบเบิร์นโปรแกรมของ ETT รุ่น ET-AVR ISP เป็นบอร์ดที่ออกแบบมาเพื่อใช้ในการดาว์นโหลด Hex File ให้กบ MCU ตระกูล AVR ของ Atmel โดยใช้วิธีการแบบ Serial Programming ซึ่งการดาวนโหลด Hex File จะกระทำผ่านทางพอร์ตขนานของคอมพิวเตอร์ โดยที่จะต้องใช้งานรวมกับ Software PonyProg2000 ซึ่ง PonyProg2000 เป็นโปรแกรม Download ข้อมูลแบบ HEX File ให้กับ CPU ตระกูล AVR โดยใช้วิธีการแบบ Serial Programming ซึ่งสามารถดาว์นโหลดโปรแกรมได้จาก http://www.lancos.com/ppwin95.html โดยมีโครงสร้างของบอร์ดเป็นดังรูป 3.16 ดังนี้



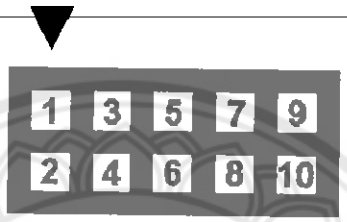
รูปที่ 3.16 แสดงโครงสร้างของเครื่องโปรแกรม

ส่วนหมายเลข 1 คือ พอร์ตสำหรับเชื่อมต่อกับสายสัญญาณของพอร์ตขนานของคอมพิวเตอร์พีซี เพื่อโปรแกรม Hex File ให้กับ MCU

ส่วนหมายเลข 2 คือ LED PGM (สีเขียว) แสดงสถานะของการโปรแกรมหรือดาว์นโหลด Hex File ลง MCU

ส่วนหมายเลข 3 คือ LED PWR (สีแดง) แสดงสถานะของไฟเลี้ยง

ส่วนหมายเลข 4 คือ พอร์ตสำหรับเชื่อมต่อกับตัวเครื่อง MP3 ในส่วนที่ไว้ใช้สำหรับโปรแกรม Hex File หรือดาว์นโหลดโปรแกรมลง MCU ให้กับเครื่องเล่น MP3 ของเรา

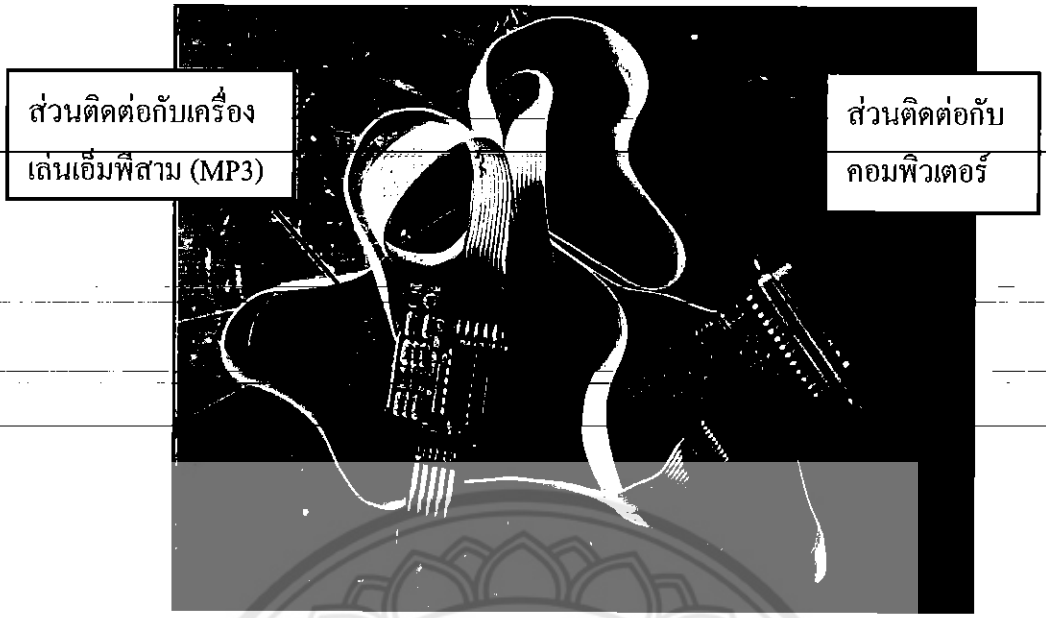


รูปที่ 3.17 แสดงตำแหน่งขาของเครื่องโปรแกรม

ซึ่งในส่วนของขาสัญญาณต่างๆของเครื่องเบิร์นโปรแกรมนั้นจะเป็นไปตามตาราง 3.11 ด้านล่างนี้ และจากรูป 3.17 ก็จะเป็นรูปแสดงเครื่องเบิร์นโปรแกรมพร้อมสายต่อเพร้อมใช้งาน

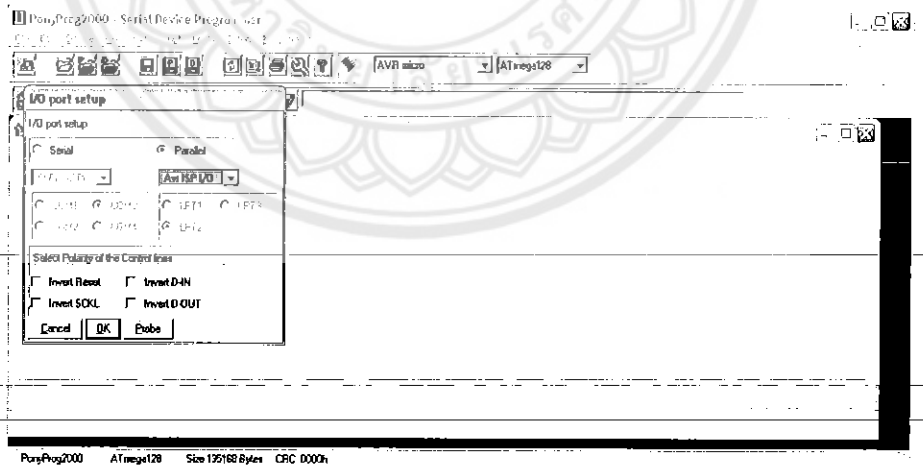
ตารางที่ 3.11 แสดงหน้าที่ของขาสัญญาณของเครื่องโปรแกรม

ตำแหน่งขา	ชื่อสัญญาณ
1	MOSI
2	VCC
3	ไม่ได้ใช้งาน
4	GND
6	GND
5	RESET
7	SCK
8	GND
9	MISO
10	GND



รูปที่ 3.18 แสดงเครื่องโปรแกรม

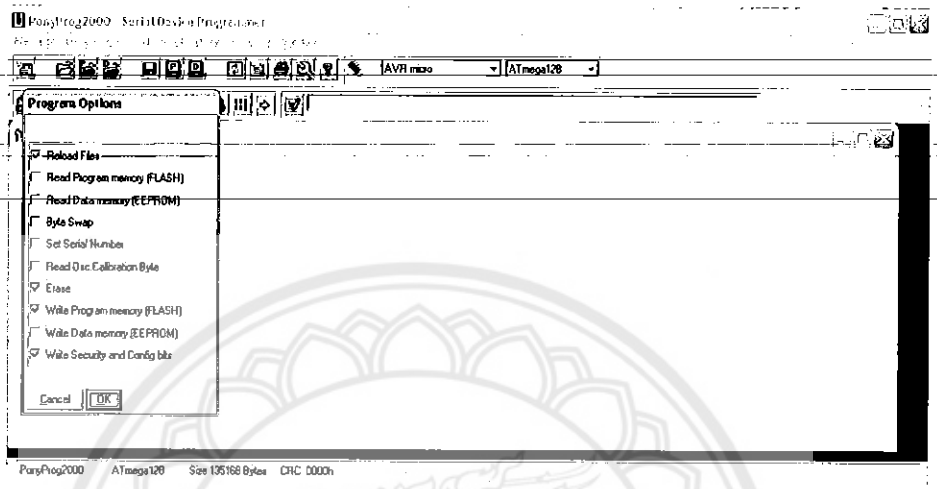
เราจะต้องทำการเรียกการใช้งานโปรแกรม PonyProg2000 ก่อนและเมื่อทำการเปิดโปรแกรมขึ้นมาแล้วนั้นเราจะต้องทำการเซตอัพค่าซะก่อนซึ่งให้ไปที่ Setup เลือก Interface Setup คลิกเลือก Parallel ในช่องให้เลือก Avr ISP I/O เสร็จแล้วให้กด ตามรูป 3.19 นี้



รูปที่ 3.19 แสดงหน้าต่างโปรแกรมการเซตค่า Interface

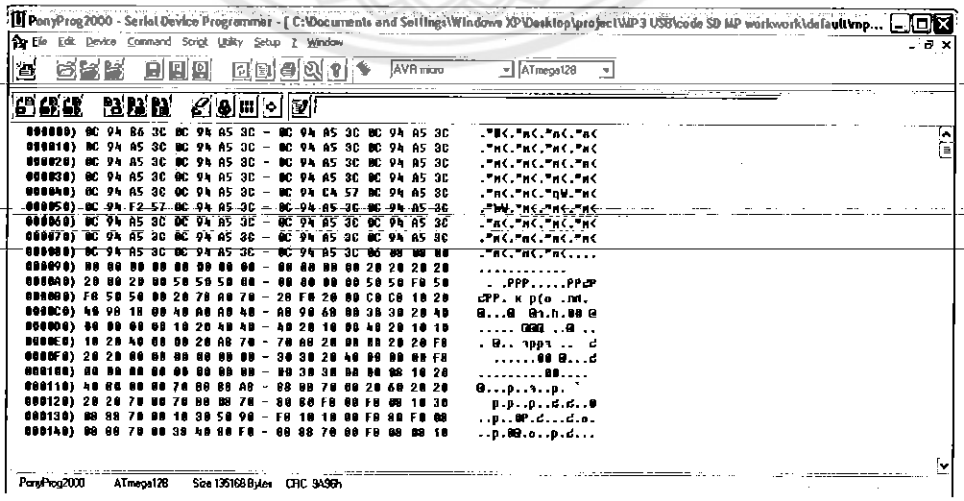
จากนั้นให้กลับไปที่ Setup ใหม่อีกครั้งโดยเลือกที่ Calibration กด yes อีกครั้งหนึ่งก็จะปรากฏหน้าต่าง Calibration OK จากนั้นให้กดปุ่ม Ok ต่อจากนั้นให้ทำการเลือก Device ให้ทำการ

เลือกไปที่ AVR micro และทำการเลือกเบอร์ไอซี ซึ่งเครื่องเล่น MP3 ของเรานั้นใช้เบอร์ ATmega128 จากนั้นให้ไปที่ Command แล้วไปที่ Program Options ทำการเลือกเครื่องหมายที่ช่อง Reload File, Erase, Write Program memory(FLASH) และ Write Security Config bits แล้วกด OK ตามรูป 3.20 นี้

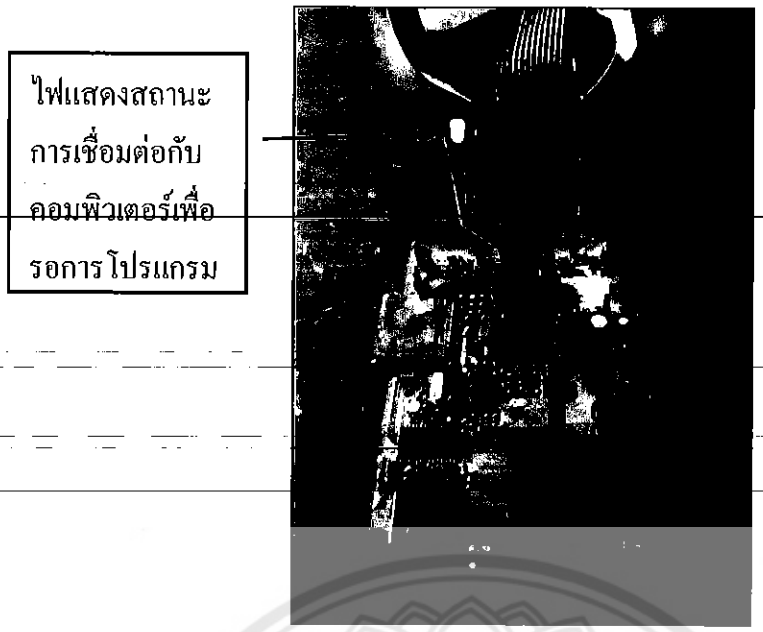


รูปที่ 3.20 แสดงหน้าต่างโปรแกรมการเซตค่า Command

จากนั้นให้ไปที่ File เลือก Open Program FLASH File ทำการเลือกไฟล์ที่เป็น .hex ที่เราสร้างขึ้นจากการใช้โปรแกรม AVR Studio 4 จากนั้นก็ไปที่ command เลือก Program หรือกด Ctrl-P โปรแกรมก็จะทำการโปรแกรมให้กับเครื่องเล่น MP3 ตามรูป 3.21 นี้ และรูปที่ 3.22 จะแสดงการรูปการโปรแกรมผ่านเครื่องเล่น MP3 ของจริง



รูปที่ 3.21 แสดงหน้าต่างเมื่อทำการโปรแกรม

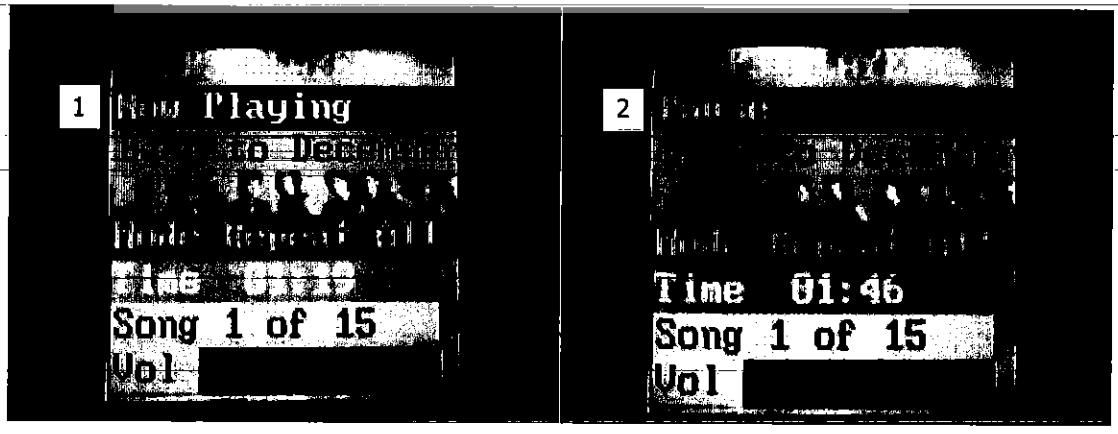


รูปที่ 3.22 แสดงการเชื่อมต่อกับเครื่องเล่น MP3 เพื่อสำหรับโปรแกรม

3.2.4 ผลของการลงโปรแกรมตามการใช้งานของสวิตซ์

(ก). การใช้งานฟังก์ชั่น PLAY/PLUASE ตามการกดสวิตซ์

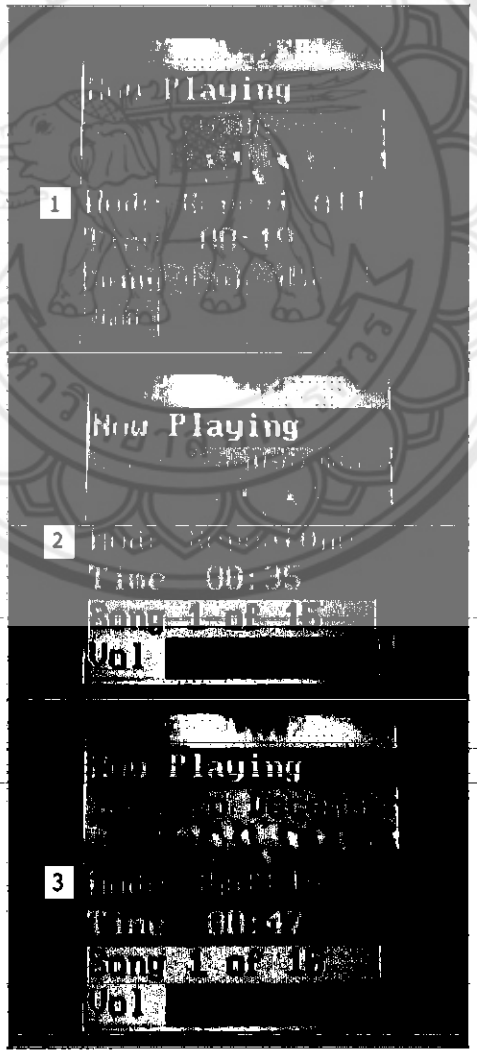
ทำการทดลองใช้งานสวิตซ์ PLAY/PAUSE จากอุปกรณ์เครื่องเล่นเอ็มพีสาม (MP3) เพื่อดูประสิทธิภาพของโค้ดโปรแกรมผ่านทางการแสดงผลทางหน้อจอ LCD 6610 เมื่อทำการกดสวิตซ์หนึ่งครั้ง เครื่องเล่นเอ็มพีสาม (MP3) จะทำการเล่นเพลงหรือ PLAY แสดงผลทางหน้าจอ LCD "Now Playing" แสดงตามภาพหมายเลข 1 จากรูป 3.24 เมื่อทำการกดสวิตซ์อีกครั้งหนึ่ง เครื่องเล่นเอ็มพีสาม (MP3) จะหยุดเพลงไว้ชั่วคราวหรือ PAUSE เพลงที่เล่น แสดงผลทางหน้าจอ LCD "Pause" แสดงตามภาพหมายเลข 2 จากรูป 3.24



รูปที่ 3.23 แสดงหน้าจอ LCD ของการใช้งานฟังก์ชั่น PLAY/PAUSE

(ข). การใช้งานฟังก์ชั่นเลือกโหมดการเล่นเพลง

ทำการทดลองใช้งานสวิตช์ MODE จากอุปกรณ์เครื่องเล่นเอ็มพีสาม (MP3) เพื่อเลือกโหมดการเล่นเพลง ดูประสิทธิภาพของโค้ดโปรแกรมผ่านทางการแสดงผลทางหน้าจอ LCD 6610 เมื่อทำการเล่นเพลงผ่านเครื่องเล่นเอ็มพีสาม (MP3) เริ่มต้นโหมดการเล่นเพลงจะอยู่ในโหมดเล่นเพลงทั้งหมดหรือ REPEAT ALL แสดงผลทางหน้าจอ LCD "Mode Repeat All" แสดงตามภาพหมายเลข 1 จากรูป 3.25 เมื่อทำการกดสวิตช์เลือกโหมดหนึ่งครั้ง เครื่องเล่นเอ็มพีสาม (MP3) จะเปลี่ยนโหมดการเล่นเพลงจาก REPEAT ALL ไปเป็นโหมด REPEAT ONE แสดงผลทางหน้าจอ LCD "Mode RepeatOne" แสดงตามภาพหมายเลข 2 จากรูป 3.25 เมื่อทำการกดสวิตช์เลือกโหมดอีกครั้ง เครื่องเล่นเอ็มพีสาม (MP3) จะเปลี่ยนโหมดการเล่นเพลงจาก REPEAT ONE ไปเป็นโหมด RANDOM แสดงผลทางหน้าจอ LCD "Mode Shuffle" แสดงตามภาพหมายเลข 3 จากรูป 3.25



รูปที่ 3.24 แสดงหน้าจอ LCD ของการใช้งานฟังก์ชั่นเลือกโหมดการเล่นเพลง

(ค). การใช้งานฟังก์ชั่นลดระดับเสียง

ทำการทดลองใช้งานสวิตซ์ Volume Down (Vol-) หรือลดระดับเสียงจากเครื่องเล่นเอ็มพีสาม (MP3) เพื่อดูประสิทธิภาพของโค้ดโปรแกรมผ่านทางการแสดงผลทางหน้าจอ LCD 6610 เริ่มต้นเมื่อมีการเล่นเพลงจากอุปกรณ์เครื่องเล่นเอ็มพีสาม (MP3) จะถูกตั้งค่าโปรแกรมไว้ที่ระดับเสียงสูงสุดแสดงตามภาพหมายเลข 1 จากรูป 3.26 เมื่อทำการกดสวิตซ์ลดระดับเสียงเครื่องเล่นเอ็มพีสาม (MP3) จะทำการปรับระดับเสียงตามการกดสวิตซ์ กดสวิตซ์ลดระดับเสียงหลายครั้งระดับเสียงก็จะยิ่งเบาลงเรื่อยๆ จนถึงระดับเสียงต่ำสุด รูปที่ 3.26 หมายเลข 2 จะแสดงผลของการกดสวิตซ์ลดระดับเสียงผ่านทางหน้าจอ LCD 6610

รูปที่ 3.25 แสดงหน้าจอ LCD ของการใช้งานฟังก์ชั่นลดระดับเสียง

(ง). การใช้งานฟังก์ชั่นเพิ่มระดับเสียง

ทำการทดลองใช้งานสวิตซ์ Volume UP (Vol+) หรือเพิ่มระดับเสียงจากเครื่องเล่นเอ็มพีสาม (MP3) พิจารณาประสิทธิภาพของโค้ดโปรแกรมผ่านทางการแสดงผลทางหน้าจอ LCD 6610เมื่อมีการเล่นเพลงจากอุปกรณ์เครื่องเล่นเอ็มพีสาม (MP3) และผู้ใช้งานหรือ User ได้ลดระดับเสียงไว้ตามหมายเลข 1 รูปที่ 3.27 เมื่อทำการกดสวิตซ์เพิ่มระดับเสียงเครื่องเล่นเอ็มพีสาม (MP3) จะทำการปรับระดับเสียงตามการกดสวิตซ์ เมื่อผู้ใช้งานกดสวิตซ์เพิ่มระดับเสียงหลายครั้ง ระดับเสียงก็จะเพิ่มขึ้นเรื่อยๆ จนถึงระดับเสียงสูงสุด รูปที่ 3.27 หมายเลข 2 จะแสดงผลของการกดสวิตซ์เพิ่มระดับเสียงผ่านทางหน้าจอ LCD 6610

รูปที่ 3.26 แสดงหน้าจอ LCD ของการใช้งานฟังก์ชั่นเพิ่มระดับเสียง

(จ). การใช้งานฟังก์ชั่นเลื่อนเพลงไปข้างหน้า

ทำการทดลองใช้งานสวิตซ์ NEXT หรือเลื่อนเพลงไปข้างหน้าจากเครื่องเล่นเอ็มพีสาม (MP3) พิจารณาประสิทธิภาพของโค้ดโปรแกรมผ่านทางการแสดงผลทางหน้าจอ LCD 6610 เมื่อมีการเล่นเพลงจากอุปกรณ์เครื่องเล่นเอ็มพีสาม (MP3) และผู้ใช้งานหรือ User ได้ทำการเล่นเพลงมาจนถึงลำดับเพลงที่ 8 จากทั้งหมด 15 เพลงตามภาพหมายเลข 1 รูปที่ 3.28 แสดงผลทางหน้าจอ LCD 6610 เมื่อทำการกดสวิตซ์หนึ่งครั้งเพื่อเลื่อนเพลงไปข้างหน้าหรือกดสวิตซ์ NEXT เครื่องเล่นเอ็มพีสาม (MP3) จะทำการเลื่อนเพลงไปข้างหน้าหนึ่งเพลง เพลงที่ทำการเล่นก็จะเป็นเพลงลำดับที่ 9 จากทั้งหมด 15 เพลงตามภาพหมายเลข 2 จากรูป 3.28 แสดงผลของการกดสวิตซ์เลื่อนเพลงไปข้างหน้าผ่านทางหน้าจอ LCD 6610

รูปที่ 3.27 แสดงหน้าจอ LCD ของการใช้งานฟังก์ชั่นเลื่อนเพลงไปข้างหน้า

(ฉ). การใช้งานฟังก์ชั่นเลื่อนย้อนเพลงไปข้างหลัง

ทำการทดลองใช้งานสวิตซ์ PREVIOUS หรือย้อนเพลงไปข้างหลังจากเครื่องเล่นเอ็มพีสาม (MP3) พิจารณาประสิทธิภาพของโค้ดโปรแกรมผ่านทางการแสดงผลทางหน้าจอ LCD 6610 เมื่อมีการเล่นเพลงจากอุปกรณ์เครื่องเล่นเอ็มพีสาม (MP3) และผู้ใช้งานหรือ User ได้ทำการเล่นเพลงมาจนถึงลำดับเพลงที่ 8 จากทั้งหมด 15 เพลงตามภาพหมายเลข 1 รูปที่ 3.29 แสดงผลทางหน้าจอแอลซีดี (LCD6610) เมื่อทำการกดสวิตซ์หนึ่งครั้งย้อนเพลงไปข้างหลังหรือกดสวิตซ์ PREVIOUS เครื่องเล่นเอ็มพีสาม (MP3) จะทำการเลื่อนเพลงไปข้างหลังหนึ่งเพลง และเพลงที่ทำการเล่นก็จะเป็นเพลงลำดับที่ 7 จากทั้งหมด 15 เพลงตามภาพหมายเลข 2 จากรูป 3.28 แสดงผลของการกดสวิตซ์เลื่อนเพลงไปข้างหน้าผ่านทางหน้าจอ LCD 6610

รูปที่ 3.28 แสดงหน้าจอ LCD ของการใช้งานฟังก์ชั่นย้อนเพลงไปข้างหลัง

บทที่ 4

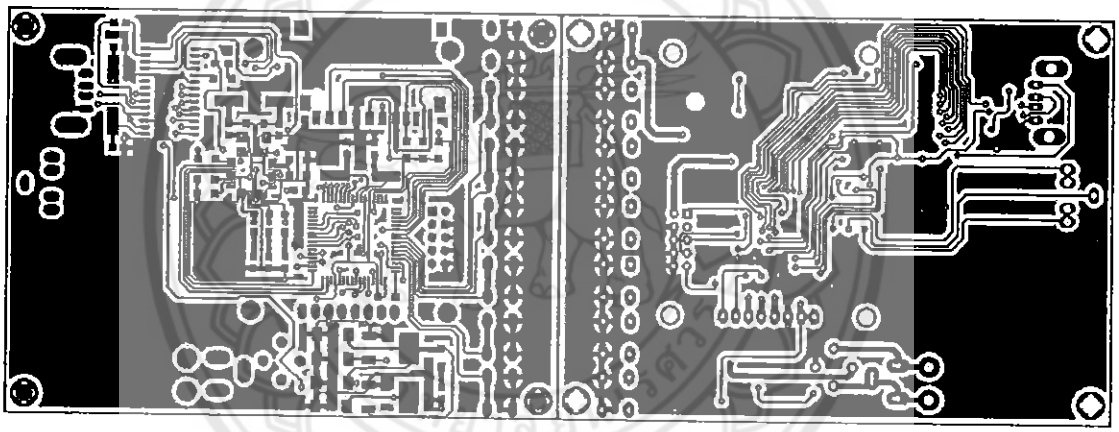
ผลการทดลองและวิเคราะห์ผลการทดลอง

ในบทที่ 4 นี้จะกล่าวถึงผลการทดลองและวิเคราะห์ผลการทดลองการใช้งานเครื่องเล่นเอ็มพีสาม (MP3) โดยที่ในบทนี้ผู้วิจัยจะกล่าวผลสรุปการประกอบชิ้นงานแต่ละชิ้นงานที่เสร็จสมบูรณ์และการทดลองในส่วนของการใช้งานร่วมกับสื่อบันทึกข้อมูล SD CARD และ USB จากการอ่านไฟล์เพลงขนาด 128 Kbps, 192 Kbps และ 320 Kbps จากการกดสวิตซ์ฟังก์ชั่นการทำงานต่างๆ

4.1 ผลการทดลอง

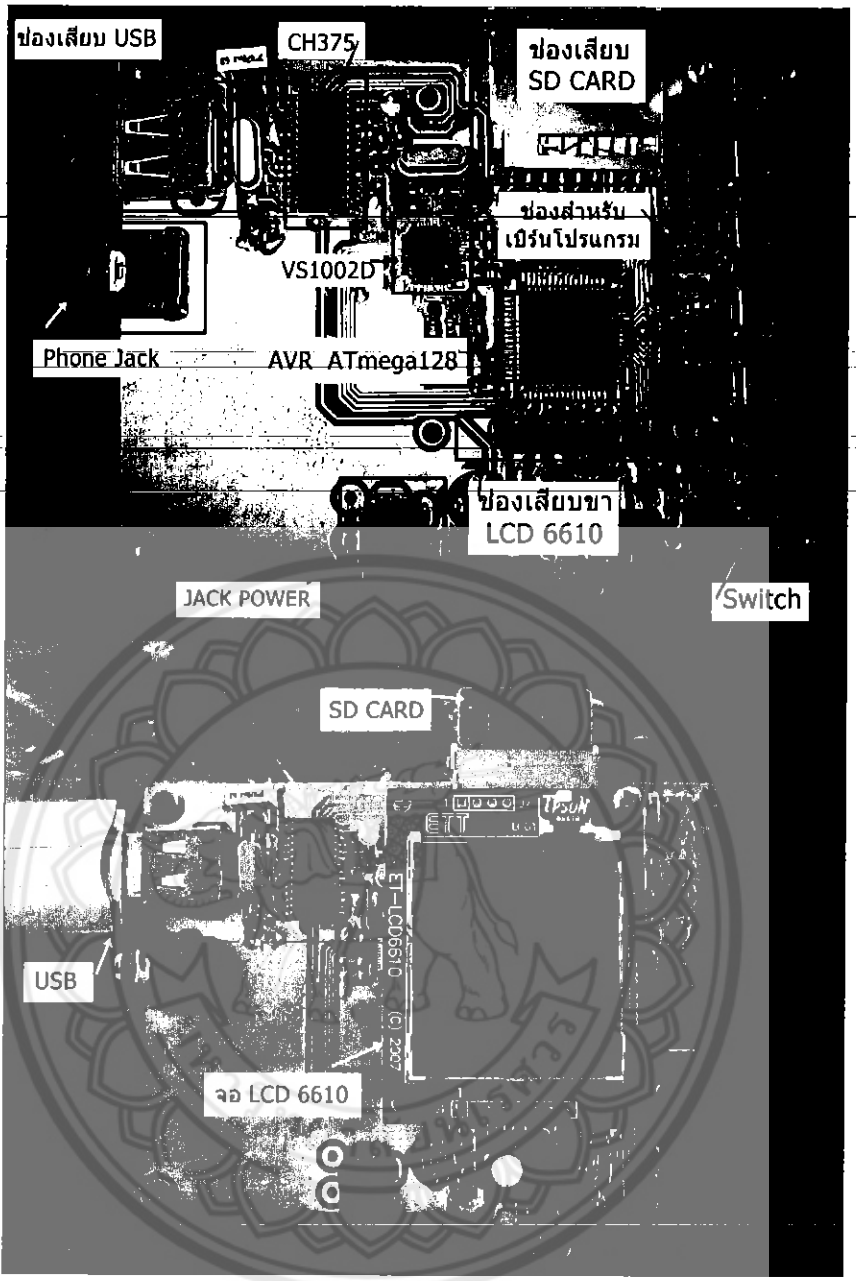
4.1.1 ผลการประกอบชิ้นงานของเครื่องเล่นเอ็มพีสาม (MP3)

จากการออกแบบเครื่องเล่นเอ็มพีสาม (MP3) ส่วนของฮาร์ดแวร์จากบทที่ 3 นำส่วนต่างๆมาออกแบบในโปรแกรมด้วย Protel99 SE ดังแสดงในรูปที่ 4.1



รูปที่ 4.1 แสดงลายวงจรพิมพ์ของเครื่องเล่นเอ็มพีสาม

จากที่ได้ออกแบบเครื่องเล่นเอ็มพีสาม (MP3) ตามที่กล่าวไว้แล้วในบทที่ 3 ที่พัฒนาขึ้นโดยที่เครื่องเล่นเอ็มพีสาม (MP3) นี้สามารถดึงข้อมูลจากสื่อบันทึกข้อมูลดิจิตอลได้ทั้ง SD CARD และ USB โดยแสดงผลของฟังก์ชั่นการทำงานผ่านจอ LCD6610 ที่รับค่าทำการจากสวิตซ์ควบคุม จากรูปด้านล่างนี้จะแสดงให้เห็นส่วนประกอบของวงจรเครื่องเล่น MP3 ที่ออกแบบไว้และผลประกอบเครื่องเล่น MP3 ที่เสร็จสมบูรณ์แล้ว ตามรูป 4.2 ข้างล่างนี้



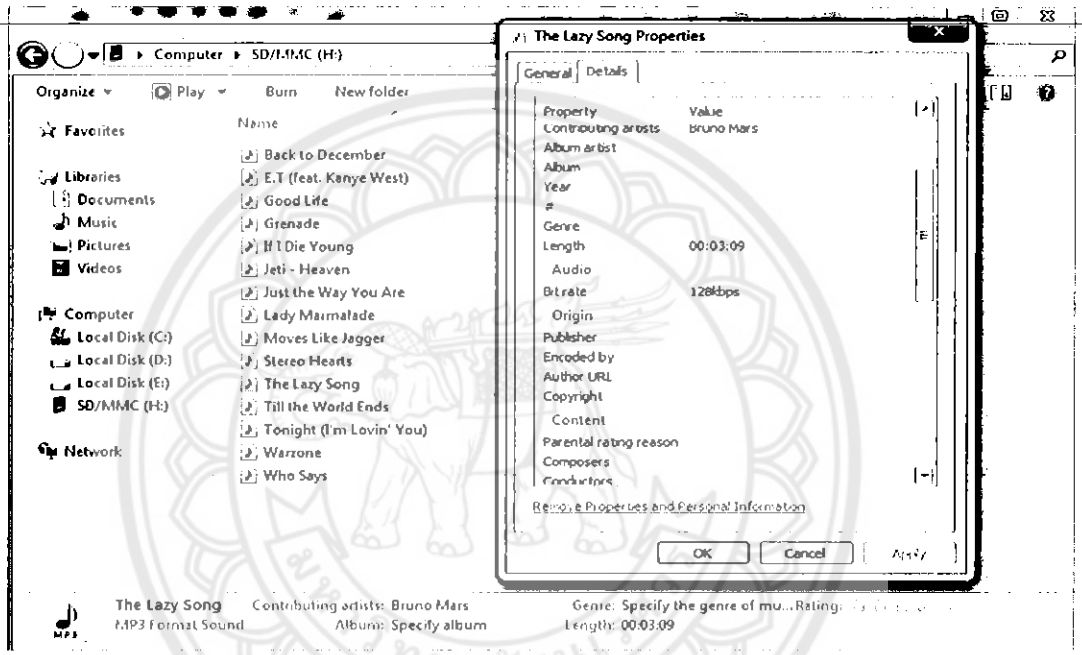
รูปที่ 4.2 แสดงส่วนประกอบชิ้นงานที่เสร็จสมบูรณ์แล้ว

4.1.2 ผลการทดลองการใช้งานร่วมกับสื่อบันทึกข้อมูล

การทดลองประสิทธิภาพการใช้งานเครื่องเล่นเพลงเอ็มพีสาม (MP3) ร่วมกับสื่อบันทึกข้อมูลเอสดีการ์ด SD CARD เพื่อให้แสดงผลประสิทธิภาพการใช้งานเครื่องเล่นเอ็มพีสามในการเล่นไฟล์เพลง MP3 ที่ขนาดบิตเรท 128Kbps 192Kbps และ 320Kbps และการทดลองการใช้งานเครื่องเล่นเอ็มพีสาม (MP3) ร่วมกับสื่อบันทึกข้อมูล USB เพื่อให้แสดงผลประสิทธิภาพการใช้งานเครื่องเล่นเอ็มพีสาม (MP3) ในการเล่นไฟล์เพลง MP3 ที่ขนาดบิตเรท 128Kbps 192Kbps และ 320Kbps

(ก).1 ผลการทดลองจากการเลือกใช้งาน SD CARD ของเพลงที่มีขนาดบิตเรท 128Kbps

จากผู้วิจัยทำการทดลองโดยทำการเล่นเพลงจากเครื่องเล่นเอ็มพีสาม (MP3) จากสื่อบันทึกข้อมูล SD CARD จำนวนเพลงทั้งหมด 15 เพลง แต่ละเพลงนั้นจะมี Bit Rate ที่ 128Kbps แสดงไว้ดังรูป 4.4 ทดลองการกดสวิตซ์เพื่อเล่นเพลง หยุดเพลง เปลี่ยนโหมดการเล่นเพลงทั้ง 3 โหมด ปรับเพิ่มระดับเสียง ปรับลดระดับเสียง เลื่อนเพลงไปข้างหน้า และทำการย้อนเพลงกลับที่ต้องการจะเล่น และดูผลการทดลองทางหน้าจอ LCD 6610 ตามรูปที่ 4.5 และแสดงผลตามตารางที่แสดงผลอยู่ที่ตารางที่4.1



รูปที่ 4.3 แสดงเพลงที่อยู่ใน SD CARD ที่มีเพลงขนาดบิตเรท 128Kbps

จากการทดลองการใช้งานเครื่องเล่นเอ็มพีสาม (MP3) โดยใช้งานร่วมกับสื่อบันทึกข้อมูล SD CARD ที่บรรจุไฟล์เพลงที่มีขนาดของ Bit Rate ที่ 128Kbps โดยใช้งานตามฟังก์ชั่นต่างๆ โดยการกดสวิตซ์ แสดงไว้ตามตาราง 4.1 ดังนี้

ตารางที่ 4.1 แสดงผลของการใช้งานเครื่องเล่นเอ็มพีสาม (MP3) กับไฟล์เพลง 128Kbps

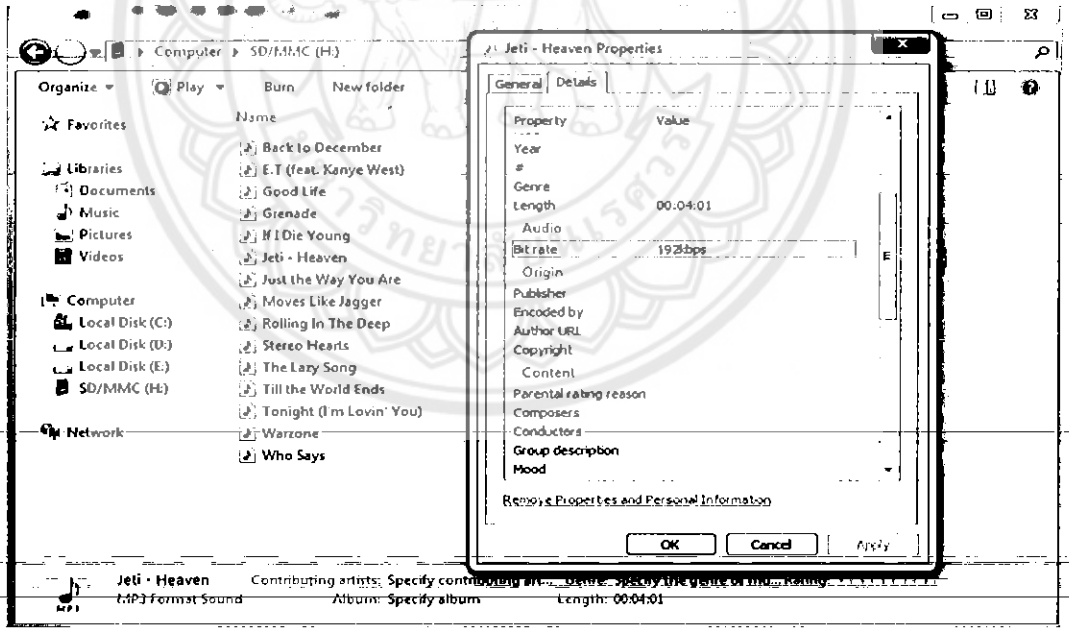
การใช้งานเครื่องเล่น MP3 จากสื่อบันทึกข้อมูล SD CARD ที่บรรจุไฟล์เพลงที่ Bit Rate 128Kbps		
การใช้งานสวิตซ์	**การแสดงผลหน้าจอ LCD**	**เพลงที่ทำการเล่น**
Play/Pause	ถูกต้อง	ถูกต้อง/รับฟังได้
Mode	ถูกต้อง	ถูกต้อง/รับฟังได้

ตารางที่ 4.1 แสดงผลของการใช้งานเครื่องเล่นเอ็มพีสาม (MP3) กับไฟล์เพลง 128Kbps [ต่อ]

การใช้งานเครื่องเล่น MP3 จากสื่อบันทึกข้อมูล SD CARD ที่บรรจุไฟล์เพลงที่ Bit Rate 128Kbps		
การใช้งานสวิตซ์	**การแสดงผลหน้าจอ LCD**	**เพลงที่ทำการเล่น**
Volume Up	ถูกต้อง	ถูกต้อง/รับฟังได้
Volume Down	ถูกต้อง	ถูกต้อง/รับฟังได้
Next	ถูกต้อง	ถูกต้อง/รับฟังได้
Previous	ถูกต้อง	ถูกต้อง/รับฟังได้

(ก).2 ผลการทดลองจากการเลือกใช้งาน SD CARD ของเพลงที่มีขนาดบิตเรท 192Kbps

ทำการทดลองโดยทำการเล่นเพลงจากเครื่องเล่นเอ็มพีสาม (MP3) จากสื่อบันทึกข้อมูล SD CARD จำนวนเพลงทั้งหมด 15 เพลง แต่ละเพลงนั้นจะมี Bit Rate ที่ 192Kbps แสดงไว้ดังรูป 4.6 ทดลองการกดสวิตซ์เพื่อเล่นเพลง หยุดเพลง เปลี่ยนโหมดการเล่นเพลงทั้ง 3 โหมด ปรับเพิ่มระดับเสียง ปรับลดระดับเสียง เลื่อนเพลงไปข้างหน้า และทำการย้อนกลับเพลงที่ต้องการจะเล่น และดูผลการทดลองทางหน้าจอ LCD 6610 ตามรูปที่ 4.7 พร้อมทั้งแสดงผลตามตารางที่ 4.2



รูปที่ 4.4 แสดงเพลงที่อยู่ใน SD CARD ที่มีเพลงขนาดบิตเรต 192Kbps

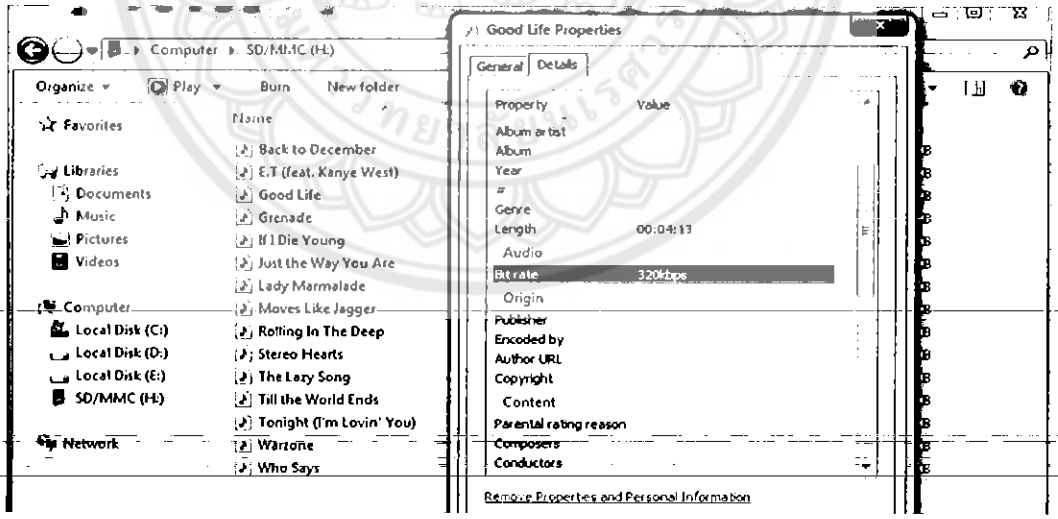
จากการทดลองการใช้งานเครื่องเล่นเอ็มพีสาม (MP3) โดยใช้งานร่วมกับสื่อบันทึกข้อมูล SD CARD ที่บรรจุไฟล์เพลงที่มีขนาดของ Bit Rate ที่ 192Kbps โดยใช้งานตามฟังก์ชั่นต่างๆ โดยการกดสวิตซ์ แสดงไว้ตามตาราง 4.2 ดังนี้

ตารางที่ 4.2 แสดงผลของการใช้งานเครื่องเล่นเอ็มพีสาม (MP3) กับไฟล์เพลง 192Kbps

การใช้งานเครื่องเล่น MP3 จากสื่อบันทึกข้อมูล SD CARD ที่บรรจุไฟล์เพลงที่ Bit Rate 192Kbps		
การใช้งานสวิตซ์	**การแสดงผลหน้าจอ LCD**	**เพลงที่ทำการเล่น**
Play/Pause	ถูกต้อง	ถูกต้อง/รับฟังได้
Mode	ถูกต้อง	ถูกต้อง/รับฟังได้
Volume Up	ถูกต้อง	ถูกต้อง/รับฟังได้
Volume Down	ถูกต้อง	ถูกต้อง/รับฟังได้
Next	ถูกต้อง	ถูกต้อง/รับฟังได้
Previous	ถูกต้อง	ถูกต้อง/รับฟังได้

(ก).3 ผลการทดลองจากการเลือกใช้งาน SD CARD ของเพลงที่มีขนาดบิตเรท 320Kbps

ทำการทดลองโดยทำการเล่นเพลงจากเครื่องเล่นเอ็มพีสาม (MP3) จากสื่อบันทึกข้อมูล SD CARD จำนวนเพลงทั้งหมด 15 เพลง แต่ละเพลงนั้นจะมี Bit Rate ที่ 320Kbps แสดงไว้ดังรูป 4.8 ทดลองการกดสวิตซ์เพื่อเล่นเพลง หยุดเพลง เปลี่ยนโหมดการเล่นเพลงทั้ง 3 โหมด ปรับเพิ่มระดับเสียง ปรับลดระดับเสียง เลื่อนเพลงไปข้างหน้า และทำการย้อนกลับเพลงที่ต้องการจะเล่น และดูผลการทดลองทางหน้าจอ LCD 6610 ตามรูปที่ 4.9 พร้อมแสดงผลตามตารางที่ 4.3



รูปที่ 4.5 แสดงเพลงที่อยู่ใน SD CARD ที่มีเพลงขนาดบิตเรต 320Kbps

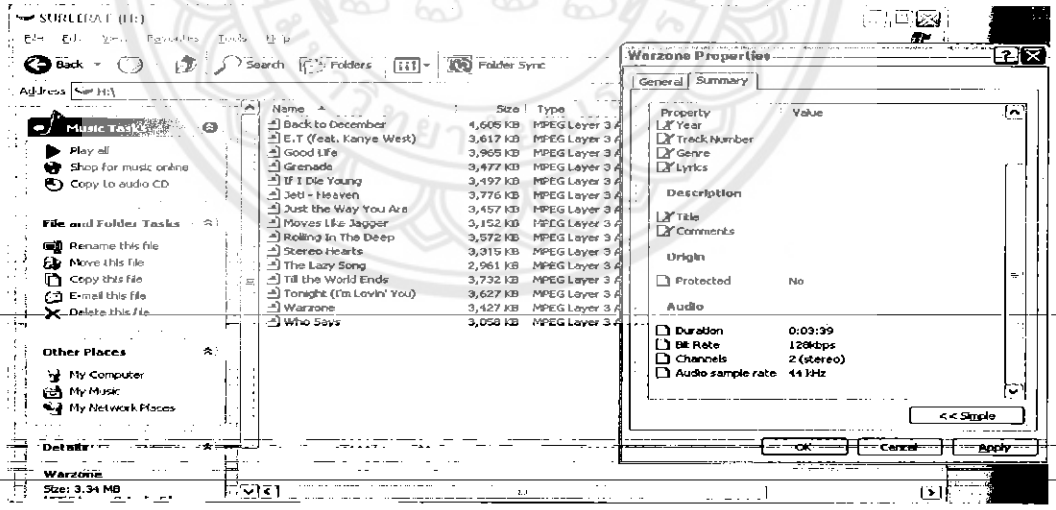
จากการทดลองการใช้งานเครื่องเล่นเอ็มพีสาม (MP3) โดยใช้งานร่วมกับสื่อบันทึกข้อมูล SD CARD ที่บรรจุไฟล์เพลงที่มีขนาดของ Bit Rate ที่ 320Kbps โดยใช้งานตามฟังก์ชั่นต่างๆ โดยการการกดสวิตซ์ แสดงไว้ตามตารางที่ 4.3 ดังนี้

ตารางที่ 4.3 แสดงผลของการใช้งานเครื่องเล่นเอ็มพีสาม (MP3) กับไฟล์เพลง 320Kbps

การใช้งานเครื่องเล่น MP3 จากสื่อบันทึกข้อมูล SD CARD ที่บรรจุไฟล์เพลงที่ Bit Rate 320 Kbps		
การใช้งานสวิตช์	**การแสดงผลหน้าจอ LCD**	**เพลงที่ทำการเล่น**
Play/Pause	ถูกต้อง	ถูกต้อง/ไม่สามารถรับฟังได้
Mode	ถูกต้อง	ถูกต้อง/ไม่สามารถรับฟังได้
Volume Up	ถูกต้อง	ถูกต้อง/ไม่สามารถรับฟังได้
Volume Down	ถูกต้อง	ถูกต้อง/ไม่สามารถรับฟังได้
Next	ถูกต้อง	ถูกต้อง/ไม่สามารถรับฟังได้
Previous	ถูกต้อง	ถูกต้อง/ไม่สามารถรับฟังได้

(ข).1 ผลการทดลองจากการเลือกใช้งาน USB ของเพลงที่มีขนาดบิตเรท 128Kbps

ทำการทดลองโดยทำการเล่นเพลงจากเครื่องเล่นเอ็มพีสาม (MP3) จากสื่อบันทึกข้อมูล USB จำนวนเพลงทั้งหมด 15 เพลง แต่ละเพลงนั้นจะมี Bit Rate ที่ 128Kbps แสดงไว้ดังรูป 4.10 ทดลองการกดสวิตช์เพื่อเล่นเพลง หยุดเพลง เปลี่ยนโหมดการเล่นเพลงทั้ง 3 โหมด ปรับเพิ่มระดับเสียง ปรับลดระดับเสียง เลื่อนเพลงไปข้างหน้า และทำการย้อนกลับเพลงที่ต้องการจะเล่น และดูผลการทดลองทางหน้าจอ LCD 6610 ตามรูปที่ 4.11 พร้อมแสดงผลตามตารางที่ 4.4



รูปที่ 4.6 แสดงเพลงที่อยู่ใน USB ที่มีเพลงขนาดบิตเรต 128Kbps

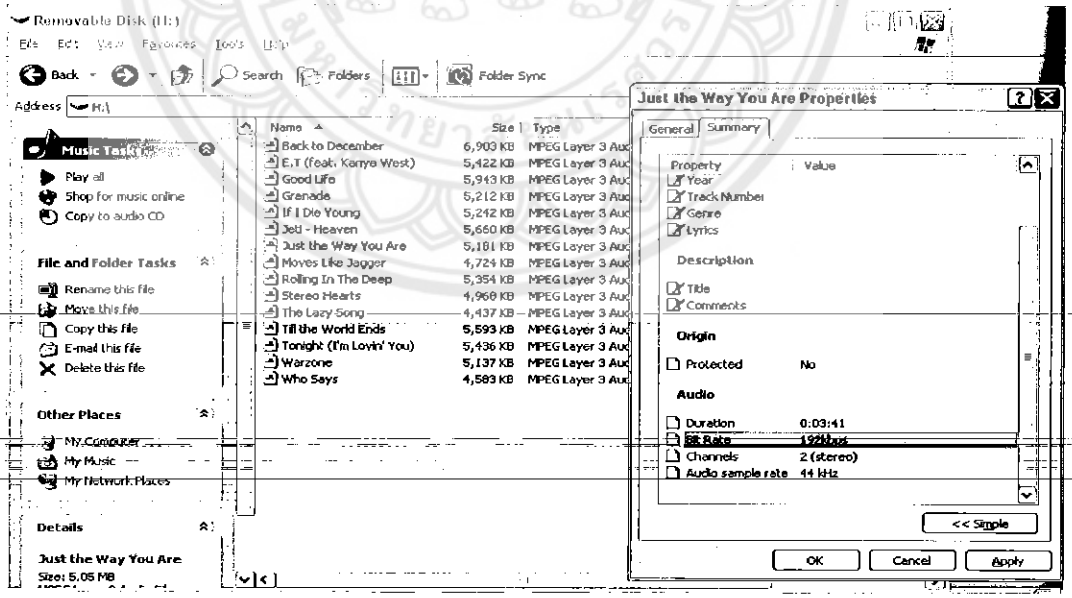
จากการทดลองการใช้งานเครื่องเล่นเอ็มพีสาม (MP3) โดยใช้งานร่วมกับสื่อบันทึกข้อมูล USB ที่บรรจุไฟล์เพลงที่มีขนาดของ Bit Rate ที่ 128Kbps โดยใช้งานตามฟังก์ชั่นต่างๆโดยการการกดสวิตช์ แสดงไว้ตามตารางที่ 4.4 ดังนี้

ตารางที่ 4.4 แสดงผลของการใช้งานเครื่องเล่นเอ็มพีสาม (MP3) กับไฟล์เพลง 128Kbps

การใช้งานเครื่องเล่น MP3 จากสื่อบันทึกข้อมูล USB ที่บรรจุไฟล์เพลงที่ Bit Rate 128Kbps		
การใช้งานสวิตซ์	**การแสดงผลหน้าจอ LCD**	**เพลงที่ทำการเล่น**
Play/Pause	ถูกต้อง	ถูกต้อง/รับฟังได้
Mode	ถูกต้อง	ถูกต้อง/รับฟังได้
Volume Up	ถูกต้อง	ถูกต้อง/รับฟังได้
Volume Down	ถูกต้อง	ถูกต้อง/รับฟังได้
Next	ถูกต้อง	ถูกต้อง/รับฟังได้
Previous	ถูกต้อง	ถูกต้อง/รับฟังได้

(ข).2 ผลการทดลองจากการเลือกใช้งาน USB ของเพลงที่มีขนาดบิตเรท 192Kbps

ทำการทดลองโดยทำการเล่นเพลงจากเครื่องเล่นเอ็มพีสาม (MP3) จากสื่อบันทึกข้อมูล USB จำนวนเพลงทั้งหมด 15 เพลง แต่ละเพลงนั้นจะมี Bit Rate ที่ 192Kbps แสดงไว้ดังรูป 4.12 ทดลองการกดสวิตซ์เพื่อเล่นเพลง หยุดเพลง เปลี่ยนโหมดการเล่นเพลงทั้ง 3 โหมด ปรับเพิ่มระดับเสียง ปรับลดระดับเสียง เลื่อนเพลงไปข้างหน้า และทำการย้อนหลังเพลงที่ต้องการจะเล่น ดูผลการทดลองทางหน้าจอ LCD 6610 ตามรูปที่ 4.13พร้อมแสดงผลตามตารางที่ 4.5



รูปที่ 4.7 แสดงเพลงที่อยู่ใน USB ที่มีเพลงขนาดบิตเรต 192Kbps

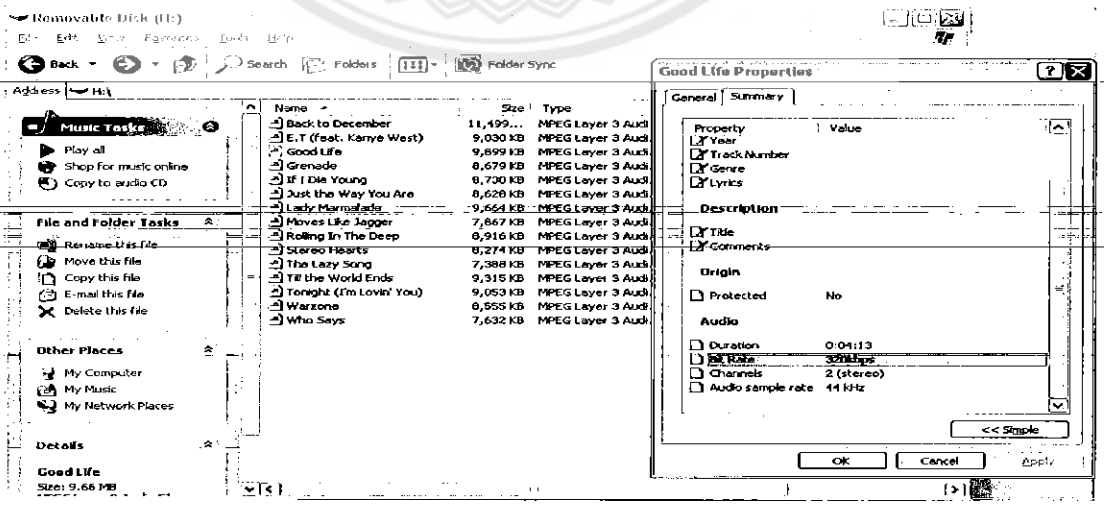
จากการทดลองการใช้งานเครื่องเล่นเอ็มพีสาม (MP3) โดยใช้งานร่วมกับสื่อบันทึกข้อมูล USB ที่บรรจุไฟล์เพลงที่มีขนาดของ Bit Rate ที่ 192Kbps โดยใช้งานตามฟังก์ชั่นต่างๆโดยการการกดสวิตซ์ แสดงไว้ตามตารางที่ 4.5 ดังนี้

ตารางที่ 4.5 แสดงผลของการใช้งานเครื่องเล่นเอ็มพีสาม (MP3) กับไฟล์เพลง 192Kbps

การใช้งานเครื่องเล่น MP3 จากสื่อบันทึกข้อมูล USB ที่บรรจุไฟล์เพลงที่ Bit Rate 192Kbps		
การใช้งานสวิตซ์	**การแสดงผลหน้าจอ LCD**	**เพลงที่ทำการเล่น**
Play/Pause	ถูกต้อง	ถูกต้อง/รับฟังได้
Mode	ถูกต้อง	ถูกต้อง/รับฟังได้
Volume Up	ถูกต้อง	ถูกต้อง/รับฟังได้
Volume Down	ถูกต้อง	ถูกต้อง/รับฟังได้
Next	ถูกต้อง	ถูกต้อง/รับฟังได้
Previous	ถูกต้อง	ถูกต้อง/รับฟังได้

(ข).3 ผลการทดลองจากการเลือกใช้งาน USB ของเพลงที่มีขนาดบิตเรท 320Kbps

ทำการทดลองโดยทำการเล่นเพลงจากเครื่องเล่นเอ็มพีสาม (MP3) จากสื่อบันทึกข้อมูล USB จำนวนเพลงทั้งหมด 15 เพลง แต่ละเพลงนั้นจะมี Bit Rate ที่ 320Kbps แสดงไว้ดังรูป 4.14 ทดลองการกดสวิตซ์เพื่อเล่นเพลง หยุดเพลง เปลี่ยนโหมดการเล่นเพลงทั้ง 3 โหมด ปรับเพิ่มระดับเสียง ปรับลดระดับเสียง เลื่อนเพลงไปข้างหน้า และทำการย้อนกลับเพลงที่ต้องการจะเล่นและดูผลการทดลองทางหน้าจอ LCD 6610 ตามรูปที่ 4.15พร้อมแสดงผลตามตารางที่ 4.6



รูปที่ 4.8 แสดงเพลงที่อยู่ใน USB ที่มีเพลงขนาดบิตเรต 320Kbps

จากการทดลองการใช้งานเครื่องเล่นเอ็มพีสาม (MP3) โดยใช้งานร่วมกับสื่อบันทึกข้อมูล USB ที่บรรจุไฟล์เพลงที่มีขนาดของ Bit Rate ที่ 320Kbps โดยใช้งานตามฟังก์ชั่นต่างๆ โดยการการกดสวิตซ์ แสดงไว้ตามตารางที่ 4.6 ดังนี้

ตารางที่ 4.6 แสดงผลของการใช้งานเครื่องเล่นเอ็มพีสาม (MP3) กับไฟล์เพลง 320Kbps

การใช้งานเครื่องเล่น MP3 จากสื่อบันทึกข้อมูล USB ที่บรรจุไฟล์เพลงที่ Bit Rate 320 Kbps		
การใช้งานสวิตซ์	**การแสดงผลหน้าจอ LCD**	**เพลงที่ทำการเล่น**
Play/Pause	ถูกต้อง	ถูกต้อง/ไม่สามารถรับฟังได้
Mode	ถูกต้อง	ถูกต้อง/ไม่สามารถรับฟังได้
Volume Up	ถูกต้อง	ถูกต้อง/ไม่สามารถรับฟังได้
Volume Down	ถูกต้อง	ถูกต้อง/ไม่สามารถรับฟังได้
Next	ถูกต้อง	ถูกต้อง/ไม่สามารถรับฟังได้
Previous	ถูกต้อง	ถูกต้อง/ไม่สามารถรับฟังได้

จากการทดลองการใช้งานเครื่องเล่นเอ็มพีสาม (MP3) การใช้งานฟังก์ชั่นกดสวิตซ์ในการเล่นไฟล์เพลง MP3 สามารถเล่นเพลงที่มี Bit Rate 128Kbps และ 192Kbps เพลงที่ออกมาสามารถรับฟังได้ แต่เครื่องเล่นเอ็มพีสาม (MP3) เครื่องนี้ไม่สามารถเล่นเพลงที่มีขนาดของ Bit Rate 320Kbps ได้เนื่องจากเพลงที่ออกมานั้นจะยาน เกิดการรบกวนของสัญญาณเสียง เกิดเสียงซ่าๆไม่สามารถรับฟังได้ แต่สามารถที่จะแสดงผลของฟังก์ชั่นการทำงานได้ปกติ เพราะฉะนั้น Bit Rate 320Kbps การแสดงผลของคุณภาพเสียงจะไม่สามารถฟังได้ทั้งจากการอ่านไฟล์เพลงจากสื่อบันทึกข้อมูล SD CARD และจากสื่อบันทึกข้อมูล USB

4.2 วิเคราะห์ผลการทดลอง

จากผลการทดลองใช้งานเครื่องเล่นเอ็มพีสาม (MP3) ที่ทำการเลือกไฟล์เพลงในส่วนของสื่อบันทึกข้อมูล SD CARD นั้นพบว่าสามารถใช้งานในฟังก์ชั่นที่ได้โปรแกรมไว้ทำงานตามที่เขียนโปรแกรมได้ตามที่ตั้งจุดประสงค์ สามารถที่จะกดสวิตซ์เพื่อที่จะให้เล่นเพลงและหยุดเพลงได้ เลือกโหมดการเล่นเพลงได้ทั้ง Repeat All, Repeat One และ Shuffle ได้ สามารถเลื่อนเพลงที่ต้องการจะเล่นได้ทั้งไปข้างหน้าและถอยหลังได้ ตัวเครื่องเล่น MP3 สามารถเพิ่มและลดระดับเสียงได้ โดยขนาดของเพลงแต่ละเพลงนั้นจะต้องไม่เกินกับหน่วยความจำของไมโครโปรเซสเซอร์ที่ผู้วิจัยใช้ด้วย ซึ่งในที่นี้เพลงที่มีบิตเรท 320Kbps นั่นไม่สามารถเล่นได้ ถึงแม้ว่าจะเล่นเสียงที่ออกมานั่นก็จะมีความผิดพลาด เกินเสียงที่ยานและมีเสียงซ่าๆแทรก เสียงที่ออกมานั่นไม่ไพเราะ แต่ถ้าผู้วิจัยเล่น

ไฟล์เพลงที่มีขนาดบิตเรท 128Kbps หรือ 192Kbps แล้วเสียงที่ออกมานั่นก็จะสามารถฟังได้ มีความไพเราะเหมาะสมกับการรับฟัง

และจากผลการทดลองใช้งานเครื่องเล่นเอ็มพีสาม (MP3) เมื่อเลือกไฟล์เพลงในส่วนของสื่อบันทึกข้อมูล USB นั้นพบว่าสามารถใช้งานในฟังก์ชั่นที่ได้โปรแกรมไว้เหมือนๆกับในส่วนที่ใช้งานกับสื่อบันทึกข้อมูล SD CARD สามารถที่จะกดสวิตซ์เพื่อที่จะให้เล่นเพลงและหยุดเพลงได้ เลือกโหมดการเล่นเพลงได้ทั้ง Repeat All, Repeat One และ Shuffle ได้ สามารถเลื่อนเพลงที่ต้องการจะเล่นได้ทั้งไปข้างหน้าและถอยหลังได้ และตัวเครื่องเล่น MP3 สามารถเพิ่มหรือลดระดับเสียงได้เช่นกัน และในส่วนของขนาดบิตเรทของไฟล์เพลงนั่นก็มีผลเช่นเดียวกันกับที่ดึงข้อมูลจาก SD CARD ซึ่งเพลงที่มีบิตเรท 128Kbps และ 192Kbps เครื่องเล่น MP3 สามารถเล่นเพลงแล้วฟังได้มีความไพเราะ ส่วนเพลงที่มีบิตเรท 320Kbps นั่นไม่สามารถฟังได้เช่นเดียวกัน

บทที่ 5
บทสรุป

ในบทนี้ จะกล่าวถึงสรุปผลการดำเนินงานในโครงงานที่ได้นำเสนอการการออกแบบเครื่องเล่นเอ็มพีสาม (MP3) การทำงานของเครื่องเล่นเอ็มพีสาม (MP3) และได้นำเสนอปัญหาต่างๆ และแนวทางแก้ไขดังต่อไปนี้

5.1 สรุปผลการทดลอง

จากการทดลองการใช้งานเครื่องเล่นเอ็มพีสาม (MP3) โดยการอ่านไฟล์เพลงจากสื่อบันทึกข้อมูล SC CARD และจากสื่อบันทึกข้อมูล USB นั้นสามารถสรุปผลได้ว่าเครื่องเล่นเอ็มพีสาม (MP3) ที่อ่านไฟล์เพลงจากสื่อบันทึกข้อมูล SD CARD และจากสื่อบันทึกข้อมูล USB นั้นสามารถใช้งานในส่วนฟังก์ชั่นต่างๆได้ดี สามารถทำงานตามการเลือกกดสวิตซ์ต่างๆได้ทุกสวิตซ์ควบคุม

1. สามารถที่จะเลือกอ่านไฟล์ได้จาก SD CARD หรือ USB โดยที่เมื่อเลือกอ่านไฟล์จากจากสื่อบันทึกข้อมูล SD CARD ก็จะทำการกดสวิตซ์ “PREV” หรือถ้าต้องการอ่านไฟล์เพลงจากสื่อบันทึกข้อมูล USB ก็ให้ไปกดสวิตซ์ “NEXT” แทน
2. สามารถเลือกโหมดเล่นเพลงผ่านการกดสวิตซ์ “MODE” ได้ซึ่งมีโหมดการเล่นเพลงทั้งหมด 3 โหมดด้วยกันคือ Repeat All, Repeat One และ Shuffle
3. สามารถที่จะเลื่อนไปข้างหน้าได้โดยการกดสวิตซ์ “Next” และย้อนเพลงไปข้างหลังโดยการกดสวิตซ์ “ Previous”
4. สามารถที่จะปรับเพิ่มระดับเสียงโดยการกดสวิตซ์ “Volume Up” หรือถ้าต้องการที่จะปรับลดระดับเสียงก็ให้ไปกดสวิตซ์ “Volume Down”

จากการทดลองเล่นกับไฟล์เสียง MP3 ตามมาตรฐานโดยใช้ Bit Rate ที่ 128 Kbps และ 192Kbps นั่นเครื่องเล่นเอ็มพีสาม (MP3) นั้นสามารถใช้งานได้ทุกฟังก์ชั่นผ่านสวิตซ์ควบคุมได้ดี และเพลงที่เล่นออกมานั่นสามารถฟังได้ไม่มีความผิดเพี้ยนของเสียง

ซึ่งจากวัตถุประสงค์ของโครงงานเครื่องเล่นเอ็มพีสาม (MP3) นั้นเราสามารถที่จะออกแบบเครื่องเล่นเอ็มพีสาม (MP3) จากอุปกรณ์ที่มีขายตามท้องตลาดได้ โดยที่เครื่องเล่น MP3 สามารถใช้งานกับตัวอุปกรณ์สื่อบันทึกข้อมูล SD CARD และสื่อบันทึกข้อมูล USB เพื่ออ่านไฟล์เพลงจากสื่อบันทึกข้อมูลทั้งสองได้ ซึ่งเราได้ใช้ไอซี AVR ATmega128 เป็นตัวควบคุมการทำงานของเครื่องเล่นเอ็มพีสาม (MP3) นี้ ซึ่งเป็นไปตามวัตถุประสงค์ของโครงงาน

5.2 ปัญหาและแนวทางแก้ปัญหา

5.2.1 เครื่องเล่นไม่สามารถที่จะเลือกเล่นเพลงเฉพาะเจาะจงในโฟลเดอร์ใดโฟลเดอร์หนึ่งได้ ฉะนั้นเมื่อไม่ต้องการฟังเพลงใดก็ให้กดสวิตซ์เพื่อเลื่อนเพลง หรือทำการลบไฟล์เพลงที่ไม่อยากฟังออกจาก SD CARD หรือ USB

5.2.2 เครื่องเล่นไม่สามารถเล่นไฟล์เพลงที่มี Bit Rate สูงๆได้ ซึ่งจะทำให้เพลงที่เล่นออกมานั่นเสียงไม่ไพเราะ เสียงยาน ฉะนั่นก่อนที่จะทำการบันทึกไฟล์เพลงลงใน SD CARD หรือ USB ก็ควรที่จะแปลงคุณภาพบิตเรทของเพลงก่อนที่จะทำการบันทึกไว้ใน SD CARD หรือ USB ซึ่งเราแนะนำที่บิตเรทที่ 128Kbps ซึ่งเครื่องเล่นสามารถเล่นได้และยังทำให้มีเนื้อที่ในการจัดเก็บเพลงเพิ่มขึ้นอีกด้วย

5.2.3 ขณะที่เครื่องเล่นเอ็มพีสาม (MP3) เล่นเพลงอยู่ไม่ว่าจะดึงจากหน่วยความจำใดความจำหนึ่ง เราจะไม่สามารถกดสวิตซ์เพื่อที่จะกลับไปยังหน้าจอหลักเพื่อเลือกเข้าสู่อีกหน่วยความจำหนึ่งได้ จำเป็นต้องกดรีเซท เพื่อกลับไปเริ่มต้น แนวทางการแก้ไขก็ควรที่จะสร้างสวิตซ์เพื่อขึ้นมาอีกหนึ่งสวิตซ์ และให้โปรแกรมในส่วนของหน้าจอหลักเพื่อกดสลับการใช้งานจากหน่วยความจำ

5.3 แนวทางในการพัฒนาต่อไป

ในการพัฒนาเครื่องเล่นเอ็มพีสาม (MP3) ต่อไปนั่นควรที่จะออกแบบให้เครื่องเล่น MP3 นั้นสามารถที่จะเลือกเล่นไฟล์เพลงเฉพาะโฟลเดอร์ที่ทำการจัดเก็บไฟล์ไว้ได้ ซึ่งเครื่องเล่น MP3 เครื่องนี้ไม่สามารถที่จะเล่นเพลงที่เฉพาะเจาะจงหรือเพลงที่ผู้ใช้ได้แยกไฟล์เป็นหมวดหมู่มาก่อนแล้ว และควรที่จะทำการเพิ่มสวิตซ์ในส่วนของหน้าจอหลักในการเลือกการใช้งานจากสื่อบันทึกข้อมูลใด หลังจากที่เราเล่นเพลงอยู่และอยากที่จะเลือกเพลงจากอีกส่วนหนึ่ง

เอกสารอ้างอิง

[1] ประจิน พลังสันติสุข,การเขียนโปรแกรมควบคุมไมโครคอนโทรเลอร์ AVR ด้วยภาษา C กับ WinAVR เล่ม 1,บริษัท แอพซอฟต์เทค จำกัด

[2] ประจิน พลังสันติสุข,การเขียนโปรแกรมควบคุมไมโครคอนโทรเลอร์ AVR ด้วยภาษา C เล่ม 2 ,บริษัท แอพซอฟต์เทค จำกัด

[3] http://elm-chan.org/docs/mmc/mmc_e.html

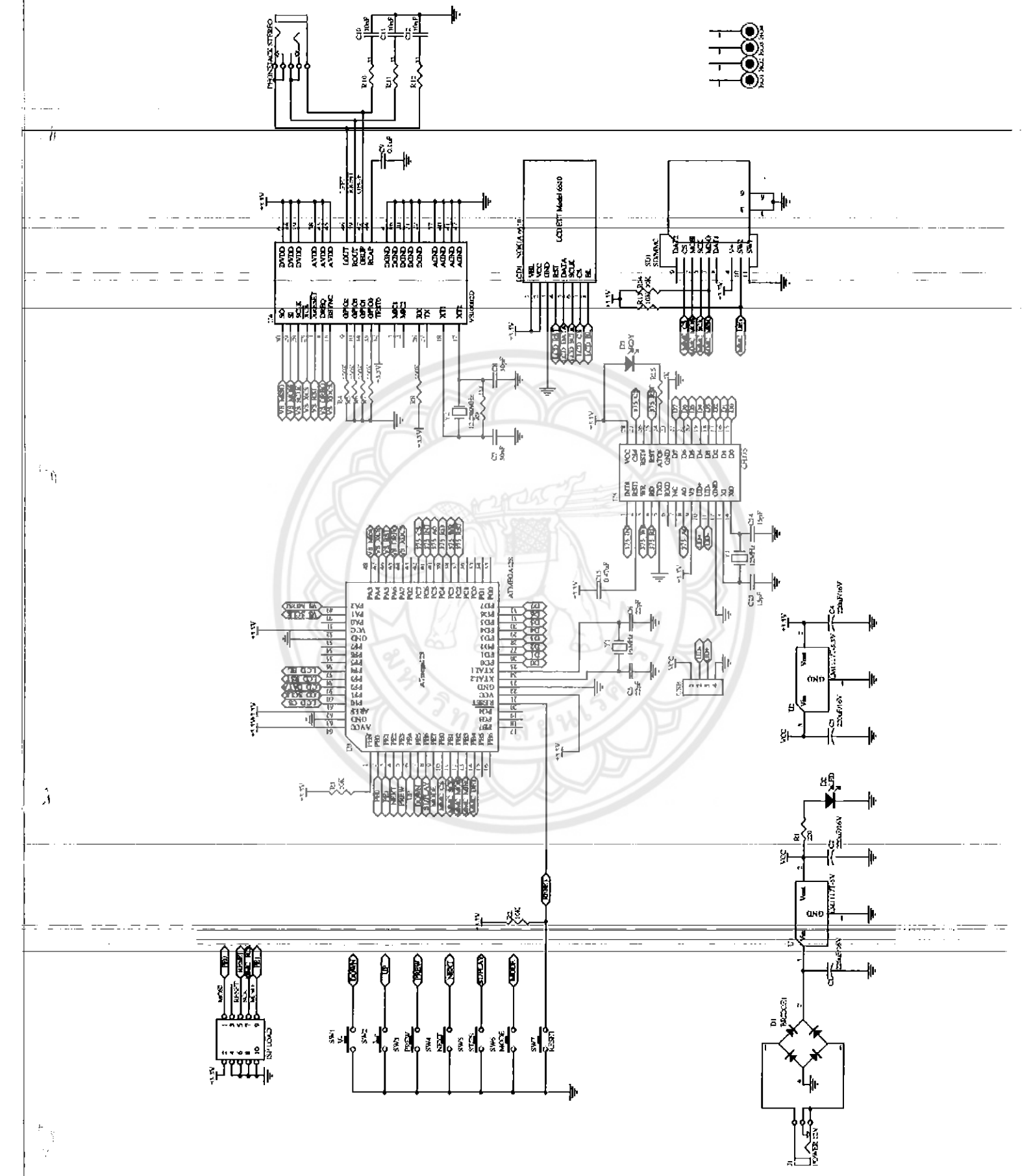
[4] http://www.ett.co.th

ภาคผนวก ก

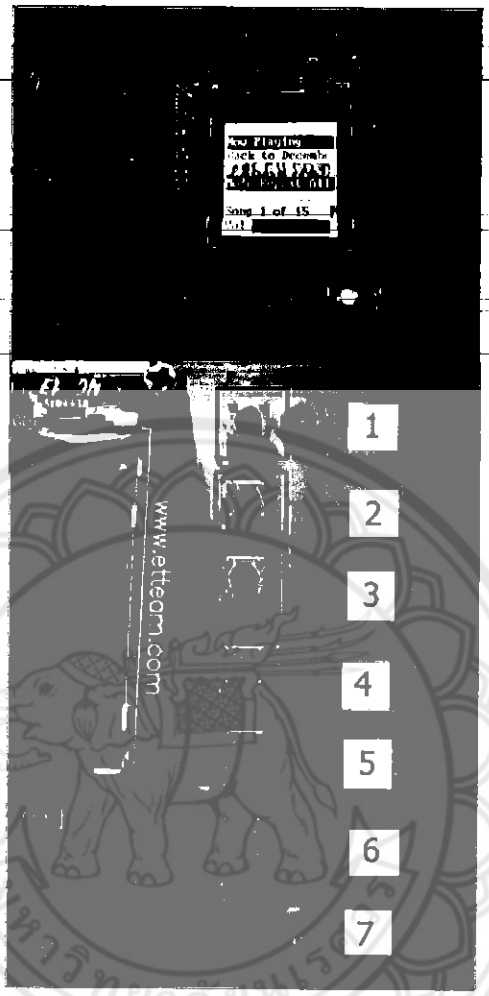
วงจรรวมทั้งหมดของเครื่องเล่นเอ็มพีสาม

ภาคผนวก ข

สวิตซ์ควบคุมเครื่องเล่นเอ็มพีสาม (MP3)



สวิตซ์ควบคุมเครื่องเล่นเอ็มพีสาม (MP3)



รูปที่ 1 แสดงสวิตซ์ควบคุมเครื่องเล่นเอ็มพีสาม (MP3)

สวิตซ์ควบคุมเครื่องเล่นเอ็มพีสาม (MP3)

- หมายเลข 1 คือสวิตซ์ Reset
- หมายเลข 2 คือสวิตซ์เลือกโหมดการเล่นเพลง MODE
- หมายเลข 3 คือสวิตซ์ PLAY/PAUSE
- หมายเลข 4 คือสวิตซ์ ลดระดับเสียง Volume Down (Vol-)
- หมายเลข 5 คือสวิตซ์ เพิ่มระดับเสียง Volume Up (Vol+)
- หมายเลข 6 คือ สวิตซ์ ย้อนเพลงหรือ Previous
- หมายเลข 7 คือสวิตซ์ เลื่อนเพลงไปข้างหน้าหรือ Next

ประวัติผู้เขียนโครงงาน

ชื่อ นางสาวณัฐกาญจน์ มีสุขโข
ภูมิลำเนา 285/1 หมู่ 9 ต.บ้านสวน อ.เมือง จ.สุโขทัย
ประวัติการศึกษา

- จบระดับมัธยมศึกษาจากโรงเรียนบ้านสวนวิทยาคม
- ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 4 สาขาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail: jai_ee@hotmail.com

ชื่อ นางสาวสุรีรัตน์ จันทร์เมือง
ภูมิลำเนา 97/1 หมู่ 3 ต.บ้านสวน อ.เมือง จ.สุโขทัย
ประวัติการศึกษา

- จบระดับมัธยมศึกษาจากโรงเรียนสุโขทัยวิทยาคม
- ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 4 สาขาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail: pum_orange@hotmail.com