

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ไมโครคอนโทรลเลอร์ควบคุมซีดีรอม
MICROCONTROLLER CONTROL CD-ROM



เลขหมู่.....
เลขทะเบียน..... 62752
วัน,เดือน,ปี..... 21 ส.ค. 2549

b..... 62752
i.....

ปฏิญานี้เป็นส่วนหนึ่งของการศึกษาค้นคว้าหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาอิเล็กทรอนิกส์
เทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไมโครคอนโทรลเลอร์ควบคุมซีดีรอม
MICROCONTROLLER CONTROL CD-ROM



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาอิเล็กทรอนิกส์
คณะวิศวกรรมศาสตร์
เทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไมโครคอนโทรลเลอร์ควบคุมซีดีรอม

MICROCONTROLLER CONTROL CD-ROM

1. นาย สำเนาวิ บุญเดิม เลขประจำตัว 46015202

2. นาย รชานนท์ ชมภูจันทร์ เลขประจำตัว 46015191

โครงการได้รับการตรวจสอบแล้วพร้อมที่จะทำการสอบได้



..... อาจารย์ที่ปรึกษา

(ผศ. พลผดุง ผดุงกุล)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใบแนบที่ 3

ปริญญาบัตร ปีการศึกษา 2548

ภาควิชา อิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ไมโครคอนโทรลเลอร์ควบคุมซีดีรอม

ผู้จัดทำ

1. นาย สำเนาวิ บุญเดิม เลขประจำตัว 46015202
2. นาย ราชานนท์ ชมภูจันทร์ เลขประจำตัว 46015191



อาจารย์ที่ปรึกษา

(ผศ. พงศคุณ ผดุงกุล)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไมโครคอนโทรลเลอร์ควบคุมซีดีรอม

นาย สำเนาวิ บุญเติม รหัส 46015202
นาย ราชานนท์ ชมภูจันทร์ รหัส 46015191
ศศ. พลผดุง ผดุงกุล อาจารย์ที่ปรึกษา
ปีการศึกษา 2548

บทคัดย่อ

รายงานฉบับนี้อธิบายการออกแบบและการสร้าง ชุดควบคุมและตั้งงานซีดีรอม เพื่อใช้เป็นเครื่องเสียง ซึ่งได้รวมเอาวงจรขยายขนาด56วัตต์และแสดงผลการทำงานด้วยจอแอลซีดีขนาด16*2 ตัวอักษร ใช้ไมโครคอนโทรลเลอร์ MCS-51 เป็นตัวประมวลผลการทำงาน ใช้ภาษาแอสเซมบลีในการเขียนโปรแกรมควบคุม MCS-51จะติดต่อกับซีดีรอมโดยผ่าน IDE Interface ที่ใช้เป็นมาตรฐานสำหรับการสื่อสารแบบ ATPI ชุดควบคุมมีฟังก์ชันการทำงาน Play, Pause, Stop, Eject, Next, Back โดยสามารถเลือกควบคุมได้จากรีโมทคอนโทรล หรือ จากสวิทช์ปุ่มกดแบบธรรมดา สามารถปรับเพิ่มลดเสียงได้และเลือกเล่นเพลงได้15เพลงจากรีโมทคอนโทรล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MICROCONTROLLER CONTROL CD-ROM

Mr.Sumnouw Boonterm ID. 46015202

Mr.Rachanon Chompujun ID. 46015191

Assoc. Prof. Ponpadung Phadungkul Advisor

Educational Year 2005

Abstract

This report describes a design and construction of the control module for audio CD-ROM playback which include 56 watt audio amplifier and 16*2 alphabet characters LCD module. In a control module, used microcontroller MCS-51 and Assembly Language programe. In order to contact with CD-ROM use ATPI protocol pass through IDE interfacing. The control module have function Play, Pause, Stop, Eject, Next, Back . We can select mode control from remote or push button on the front panel of the module, control volume and provides fifteen songs selection from remote.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญาโทฉบับนี้จัดทำขึ้นจนสำเร็จได้เนื่องจากได้รับความช่วยเหลือจากหลายๆฝ่ายด้วยกัน โดยบุคคลท่านแรกที่จะกล่าวถึงคือ ผศ. พลหญิง ผดุงกุล เป็นอาจารย์ที่ปรึกษาปริญญาโทฉบับนี้ที่ให้ความช่วยเหลือพวกเราทุกเรื่อง คอยให้คำแนะนำ ให้ความรู้ช่วยแก้ปัญหา ซึ่งขอขอบพระคุณเป็นอย่างสูง

ขอขอบคุณบุคคลสำคัญที่ทำให้ข้าพเจ้าได้มีวันนี้ ซึ่งก็คือ บิดา มารดา ที่ได้ให้การเลี้ยงดู ข้าพเจ้าเป็นอย่างดี พร้อมทั้งให้โอกาสทางการศึกษาอย่างเต็มที่ สั่งสอนเอาใจใส่จนทำให้ข้าพเจ้าเป็นคนดีในวันนี้ ข้าพเจ้าขอสำนึกในพระคุณอันสูงค่านี้ตลอดไป

และสุดท้ายต้องขอขอบคุณ พี่ๆ ปริญญาโทที่คอยให้ความช่วยเหลือให้คำแนะนำที่เป็นประโยชน์จึงขอขอบคุณมา ณ ที่นี้ด้วยครับ

นาย สำเนาวิ บุญเดิม

นาย ราชานนท์ ชมภูจันทร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทที่ 1 บทนำ	1
1.1 บล็อกไดอะแกรมของโครงการ	1
บทที่ 2 ทฤษฎีเกี่ยวข้องกับโครงการ	2
2.1 ระบบพื้นฐานของไมโครคอนโทรลเลอร์ตระกูล MCS-51	2
2.2 โครงสร้างภายในของไมโครคอนโทรลเลอร์ตระกูล MCS-51	3
2.3 การจัดตำแหน่งขาของไมโครคอนโทรลเลอร์ตระกูล MCS-51(AT89C52)	4
2.3.1 PO.0-PO.7 (ขาที่ 32-39) พอร์ต 0	4
2.3.2 P1.0-P1.7 (ขาที่ 1-8) พอร์ต 1	5
2.3.3 ขา P2.0-P2.7 (ขาที่ 21-28) พอร์ต 2	5
2.3.4 ขา P3.0-3.7 (ขาที่ 10-17) พอร์ต 3	5
2.3.5 ขา PSEN (Program Store Enable ขาที่ 29)	5
2.3.6 ขา ALE (Address Latch Enable ขาที่ 30)	6
2.3.7 ขา EA (External Access ขาที่ 31)	6
2.3.8 ขา RST (RESET ขาที่ 9)	6
2.3.9 ขาสัญญาณนาฬิกา (ขาที่ 18-19)	6
2.3.10 แหล่งจ่ายไฟ (Power Supply)	6
2.4 การจัดตำแหน่งขาของไมโครคอนโทรลเลอร์ตระกูล MCS-51(AT89C2051)	6
2.5 สัญญาณนาฬิกาของ MCS-51	7
2.6 การรีเซ็ต	7
2.7 โครงสร้างโดยทั่วไปของ LCD มอดูล	8
2.7.1 ส่วนแสดงผล (Display)	8
2.7.2 ส่วนควบคุม (Controller)	8
2.7.3 ส่วนขับ (Drive)	9
2.8 มอดูล LCD ขนาด 16 ตัวอักษร 2 บรรทัด (LCD 16 x 2)	9
2.9 หลักการทำงานของรีโมตคอนโทรล	10
2.10 หลักการทำงานของโฟโต้ไดโอดมอดูล	11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
2.11 ลักษณะของโฟโต้ไดโอดคอมพูล	12
2.12 โครงสร้างส่วนเชื่อมต่อด้านหลังของซีดีรอม	13
2.13 ไฟเลี้ยงที่จ่ายให้แก่ซีดีรอม	13
2.14 มาตรฐานการติดต่อกับไดร์ฟ	14
2.15 รีจิสเตอร์ที่ใช้ในการควบคุมการทำงานของซีดีรอม	15
2.16 รายละเอียดภายในแต่ละส่วนของรีจิสเตอร์	16
2.17 วงจรขยายเสียง	17
2.18 แหล่งจ่ายไฟให้กับซีดีรอมและวงจรควบคุม	17
บทที่ 3 การออกแบบวงจร	19
3.1 ส่วนฐานเวลาของ MCS-51	19
3.2 ส่วนชุดรีเซต	20
3.3 ส่วนควบคุมการทำงานโดยใช้สวิทช์ 6 ตัว	21
3.4 ส่วนชุดรับสัญญาณรีโมตคอนโทรล	21
3.5 ส่วนแสดงผล LCD	22
3.6 ส่วนการเชื่อมต่อขาที่ใช้งานซีดีรอม	22
3.7 แหล่งจ่ายไฟให้กับวีดีรอมและวงจรควบคุม	24
3.8 วงจรเพาเวอร์แอมป์	25
3.9 วงจรไมโครคอนโทรลเลอร์ควบคุมซีดีรอม	26
3.10 โพลีชาร์ตโปรแกรมรวม	29
3.11 โพลีชาร์ตย่อยคีย์สวิทช์	30
3.12 โพลีชาร์ตย่อยส่วน REQUEST SENE	31
3.13 โพลีชาร์ตย่อยส่วน READ TOC	32
3.14 โพลีชาร์ตย่อยส่วน PLAY	33
3.15 โพลีชาร์ตย่อยส่วน EJECT	34
3.16 โพลีชาร์ตย่อยส่วน STOP	35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
3.17 โพล์ซาร์ตย่อยส่วน NEXT	36
3.18 โพล์ซาร์ตย่อยส่วน BACK	37
3.19 โปรแกรมที่ใช้ในการติดต่อกับซีดีรอมใช้ใน AT89C2051	38
3.20 โปรแกรมที่ใช้ในการติดต่อกับซีดีรอมใช้ใน AT89C252	59
บทที่ 4 การทดลองและผลการทดลอง	108
4.1 การวัดสัญญาณของวงจขยาย(Power Amplifier)	108
4.2 ผลการทดลองวงจขยายสัญญาณ	108
4.3 คำสั่ง TEST UNIT READY COMMAND	109
4.4 คำสั่ง REQUEST SENSE COMMAND	111
4.5 คำสั่ง REQUEST SENSE READ DATA	112
4.6 คำสั่ง READ TOC COMMAND	113
4.7 คำสั่ง READ TOC DATA	114
4.8 คำสั่ง MODE SENSE COMMAND	115
4.9 คำสั่ง MODE PARAMETER HEADER	116
4.10 คำสั่ง START/STOP UNIT COMMAND	117
4.11 คำสั่ง PLAY AUDIO MSF COMMAND	118
4.12 คำสั่ง READ SUBCHANNEL COMMAND	119
4.13 คำสั่ง CD-ROM CURRENT POSITION DATA FORMAT	120
4.14 คำสั่ง MSF ADDRESS FORMAT	121
4.15 สรุปผลรหัสคำสั่งที่ใช้ในซีดีรอม	121
4.16 รายละเอียดรีจิสเตอร์ SENSE KEY/ ASC	122
บทที่ 5 สรุปและวิเคราะห์ผลการดำเนินการ	123

สารบัญรูป

	หน้า
รูป 1.1 บล็อกไดอะแกรมของโครงการ	1
รูป 2.1 บล็อกไดอะแกรมระบบพื้นฐานของไมโครคอนโทรลเลอร์	2
รูป 2.2 โครงสร้างภายในของไมโครคอนโทรลเลอร์ตระกูล MCS-51	3
รูป 2.3 ตำแหน่งขาของไมโครคอนโทรลเลอร์ตระกูล MCS-51(AT89C52)	4
รูป 2.4 การจัดตำแหน่งขาของ MCS-51(AT89C2051)	6
รูป 2.5 การต่อคริสตอล	7
รูป 2.6 การต่อชุดรีเซต	8
รูป 2.7 โครงสร้างโดยทั่วไปของ LCD มอดูล	9
รูป 2.8 รูปร่างและการจัดขามอดูล LCD แบบอักษร	9
รูป 2.9 รูปแบบของสัญญาณพัลส์ที่มีการเข้ารหัสโดยใช้ความกว้างพัลส์	10
รูป 2.10 บล็อกไดอะแกรมของวงจรภายในมอดูลกราฟิกรับอินฟราเรด	11
รูป 2.11 การประยุกต์ใช้งานกับไมโครคอนโทรลเลอร์	12
รูป 2.12 โครงสร้างส่วนเชื่อมต่อด้านหลังของซีดีรอม	13
รูป 2.13 ขาสัญญาณของ ATA/IDE Connector	14
รูป 2.14 ไอซีเพาเวอร์แอมป์เบอร์ LM3875T	17
รูป 2.15 ลักษณะการวางขาใช้งานไอซี LM2576HVT	17
รูป 2.16 การประยุกต์ LM2576HVT ใช้งานทั่วไป	18
รูป 3.1 สัญญาณนาฬิกา และแมชชีนไจเนล	19
รูป 3.2 การต่อชุดรีเซต	20
รูป 3.3 ส่วนควบคุมการทำงานโดยใช้สวิตช์ 6 ตัว	21
รูป 3.4 ส่วนชุดรับสัญญาณรีโมตคอนโทรล	21
รูป 3.5 ส่วนแสดงผล LCD	22
รูป 3.6 ส่วนการเชื่อมต่อขาที่ใช้งานซีดี	23
รูป 3.7 วงจรแหล่งจ่ายไฟให้กับซีดีรอมและวงจรควบคุม	24
รูป 3.8 วงจรเพาเวอร์แอมป์	25
รูป 3.9 วงจรไมโครคอนโทรลเลอร์ควบคุมซีดีรอม	26
รูป 3.10 ภาพชิ้นงานคานบน	27

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูป 3.11 ภาพขึ้นงานด้านหน้า	28
รูป 3.12 โฟลว์ชาร์ตโฟลว์ชาร์ตโปรแกรมรวม	29
รูป 3.13 โฟลว์ชาร์ตย่อยส่วนคีย์สวิตช์	30
รูป 3.14 โฟลว์ชาร์ตย่อยส่วน REQUEST SENE	31
รูป 3.15 โฟลว์ชาร์ตย่อยส่วน READ TOC	32
รูป 3.16 โฟลว์ชาร์ตย่อยส่วน PLAY	33
รูป 3.17 โฟลว์ชาร์ตย่อยส่วน EJECT	34
รูป 3.18 โฟลว์ชาร์ตย่อยส่วน STOP	35
รูป 3.19 โฟลว์ชาร์ตย่อยส่วน NEXT	36
รูป 3.20 โฟลว์ชาร์ตย่อยส่วน BACK	37
รูป 4.1 สัญญาณวงจรรขยาย	108
รูป 4.2 สถานะการทำงานของคำสั่ง TEST UNIT READY COMMAND	109



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้า
ตาราง 2.1 ความสัมพันธ์ในการทำงานของขา RS,R/W และ E	10
ตาราง 2.2 ไฟเลี้ยงที่จ่ายให้แก่ซีดีรอม	13
ตาราง 2.3 รูปแบบตำแหน่งของรีจิสเตอร์ I/O	15
ตาราง 2.4 สถานะรีจิสเตอร์ (ATAPI STATUS REGISTER)	16
ตาราง 2.5 การ กำหนดไบร์รับส่งข้อมูล (ATAPI BYTE COUNT REGISTER)	16
ตาราง 2.6 เลือกบิต (ATAPI DRIVE SELECT REGISTER)	16
ตาราง 4.1 รหัสคำสั่ง TEST UNIT READY COMMAND	110
ตาราง 4.2 รหัสคำสั่ง REQUEST SENSE COMMAND	111
ตาราง 4.3 เป็นการอ่านข้อมูลของ REQUEST SENSE READ DATA	112
ตาราง 4.4 รหัสคำสั่งของ TOC COMMAND	113
ตาราง 4.5 รหัสคำสั่งของ READ TOC DATA	114
ตาราง 4.6 ตัวกระทำคำสั่ง Control Field	114
ตาราง 4.7 รหัสคำสั่ง MODE SENSE COMMAND	115
ตาราง 4.8 ตัวกระทำคำสั่ง Page control	115
ตาราง 4.9 รหัสคำสั่งชุด MODE PARAMETER HEADER	116
ตาราง 4.10 ตัวกระทำคำสั่งของ Media Type Code	116
ตาราง 4.11 รหัสคำสั่ง START/STOP UNIT COMMAND	117
ตาราง 4.12 ตัวกระทำคำสั่ง Start/Stop And Eject	117
ตาราง 4.13 รหัสคำสั่ง PLAY AUDIO MSF COMMAND	118
ตาราง 4.14 รหัสคำสั่ง READ SUBCHANNEL COMMAND	119
ตาราง 4.15 รหัสคำสั่ง CD-ROM CURRENT POSITION DATA FORMAT	120
ตาราง 4.16 รหัสคำสั่ง Audio Status	120
ตาราง 4.17 รหัสคำสั่ง MSF ADDRESS FORMAT	121
ตาราง 4.18 สรุปรหัสที่ใช้ในซีดีรอม	121
ตาราง 4.19 รหัสคำสั่ง SENSE KEY/ ASC	122

บทที่ 1

บทนำ

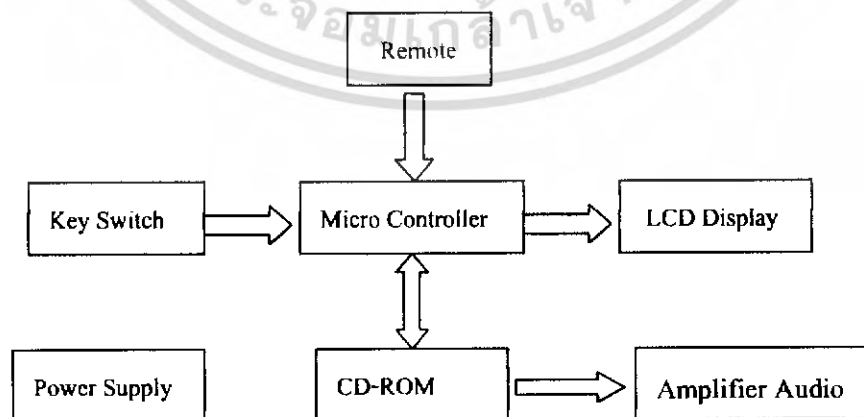
เนื่องจากในปัจจุบันอุปกรณ์ของคอมพิวเตอร์มีราคาถูกลงมาก จึงทำให้มีเครื่องคอมพิวเตอร์ใช้งานกันอย่างมากมายแต่การพัฒนาการของเทคโนโลยีเปลี่ยนแปลงอย่างรวดเร็ว จึงทำให้อุปกรณ์ที่ตกยุค ถูกถอดเก็บไว้ไม่ได้ใช้งาน อย่างเช่น ซีดีรอม เป็นจำนวนมาก ทำให้สูญเสียทรัพยากรโดยเปล่าประโยชน์

ดังนั้นจึงมีการคิดที่จะนำเอาอุปกรณ์ที่ถูกถอดเก็บไว้ไม่ได้ใช้งานมาดัดแปลงใหม่ เพื่อที่จะได้ใช้งานตามต้องการได้ โดยที่ไม่ต้องเสียอุปกรณ์นั้นโดยเปล่าประโยชน์ โดยในโครงการนี้ได้ทำการออกแบบชุดควบคุมและสั่งงานซีดีรอมที่ยังใช้งานได้อยู่ เอามาดัดแปลงเพื่อใช้ในด้านการบินเหิง โดยใช้เป็นเครื่องเสียงฟังเพลง จะใช้ไมโครคอนโทรลเลอร์ตระกูล MCS-51 เบอร์ AT89S8252 แบบแฟลช เป็นตัวควบคุมการติดต่อและสั่งงานกับซีดี-รอม เพื่อที่จะสามารถควบคุมการเล่นเพลงได้

1.1 บล็อกไดอะแกรมของโครงการ

บล็อกไดอะแกรมจะแบ่งออกเป็น 7 บล็อก ดังนี้

- บล็อกแหล่งจ่ายไฟ (Power Supply) จะใช้ 5 โวลต์ และ 12 โวลต์
- บล็อกแสดงผล (Display) จะใช้จอผลึก LCD แบบ 16 ตัวอักษร 2 บรรทัด
- บล็อกสวิตช์ (Key Switch) จะใช้อยู่ 5 กีบ คือ Play, Next, Back, Stop, Eject
- บล็อก รีโมท (Remote) จะใช้ของ Creative เป็นตัวสั่งการทำงาน
- บล็อกไมโครคอนโทรลเลอร์ (Micro Controller) ใช้เบอร์ AT89S8252 เป็นตัวควบคุม
- บล็อกซีดีรอม (CD-ROM) จะใช้ซีดีรอม ของ SONY
- บล็อกวงจรขยายเสียง (Amplifier Audio) ใช้ไอซีเพาเวอร์แอมป์เบอร์ LM3875T



รูป 1.1 บล็อกไดอะแกรมของโครงการ

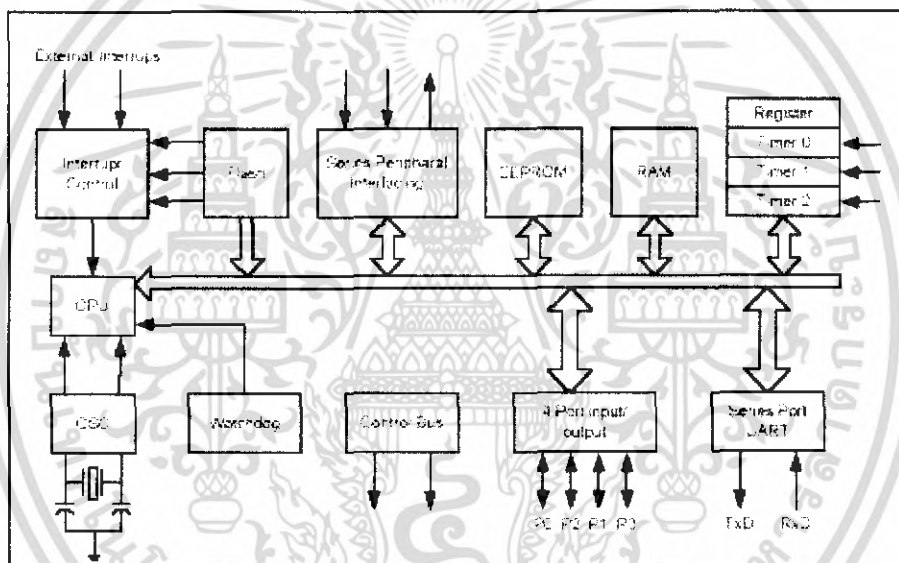
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีเกี่ยวข้องกับโครงการ

2.1 ระบบพื้นฐานของไมโครคอนโทรลเลอร์ตระกูล MCS-51

ไมโครคอนโทรลเลอร์ตระกูล MCS-51 ทุก ๆ เบอร์จะมีโครงสร้างและขาใช้งานพื้นฐานที่เหมือน ๆ กัน ซึ่งถูกออกแบบมาเพื่อสนองความต้องการของผู้ใช้แบบสำเร็จในไอซีตัวเดียว คือ มีสายสัญญาณอินพุต/เอาต์พุตภายในตัว โดยที่พอร์ตอินพุตและเอาต์พุตเป็นแบบบัฟเฟอร์ที่ใช้เชื่อมต่อกับวงจรภายนอก และสายสัญญาณควบคุมอื่น ๆ ที่ใช้สำหรับแยกสัญญาณข้อมูลกับสายสัญญาณกำหนดตำแหน่งหน่วยความจำ และยังมีชุดคำสั่งพิเศษเพื่อจัดการข้อมูลเพิ่มเติมขึ้นอีก นอกจากนั้นยังมีวงจรมับเวลาและตั้งเวลาด้วย



รูป 2.1 บล็อกไดอะแกรมระบบพื้นฐานของไมโครคอนโทรลเลอร์

บล็อกไดอะแกรมดังรูป 2.1 นั้นเป็นระบบพื้นฐานของไมโครคอนโทรลเลอร์ตระกูล MCS-51 โดยทั่วไป ซึ่งจะมีคุณสมบัติโดยรวมดังนี้

เป็นไมโครคอนโทรลเลอร์ขนาด 8 บิต

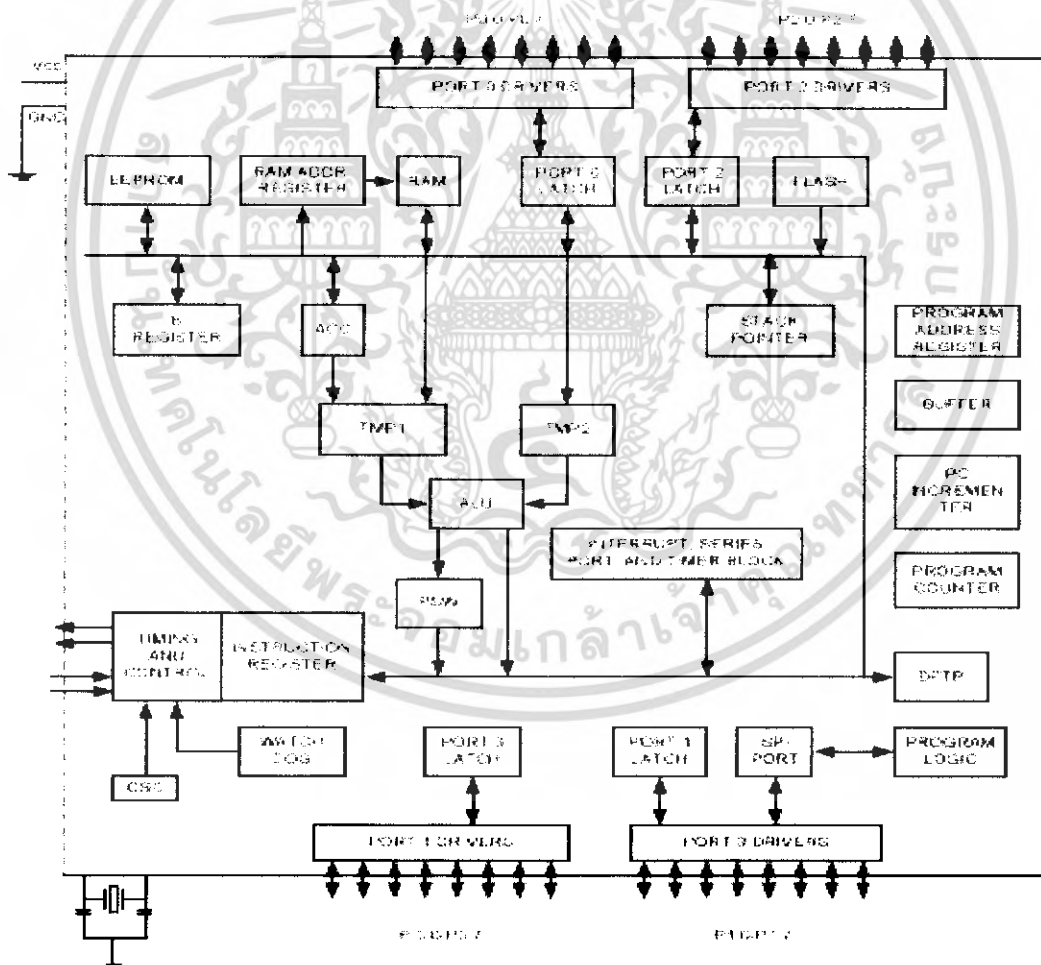
1. หน่วยความจำโปรแกรมเป็นแบบแฟลช สามารถลบและเขียนใหม่ได้เป็นพันครั้ง
2. มีหน่วยความจำภายในเป็นแบบแรม และเป็นแบบ EEPROM ในบางเบอร์ ซึ่งไม่ต้องกลัวว่าข้อมูลจะหาย เมื่อไม่มีไฟมาเลี้ยงวงจร
3. พอร์ตใช้งานเป็นแบบสองทิศทาง สามารถใช้งานได้ทั้งแบบอินพุตและเอาต์พุต
4. สามารถขยายหน่วยความจำโปรแกรมภายนอกได้สูงสุด 64 kbyte

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. มีไทเมอร์/คาเตอร์ขนาด 16 บิต อย่างน้อย 2 ตัว
6. มีพอร์ตสื่อสารอนุกรมแบบฟลูตเพลกซ์
7. รองรับแหล่งกำเนิดอินเทอร์รัปต์ได้ 6 ประเภท
8. มีวงจรกำเนิดสัญญาณนาฬิกาอยู่ภายใน

2.2 โครงสร้างภายในของไมโครคอนโทรลเลอร์ตระกูล MCS-51

วงจรรภายในของไมโครคอนโทรลเลอร์ตระกูล MCS-51 ประกอบด้วยวงจรพอร์ตอินพุตและเอาต์พุตทั้งหมด 4 พอร์ต โดยแต่ละพอร์ตจะเป็นแบบ 8 บิต มีหน่วยความจำโปรแกรมภายในแบบ EEPROM และมีหน่วยความจำที่เป็นข้อมูลภายในแบบ RAM ซึ่งรวมอยู่ในวงจหลักของไมโครคอนโทรลเลอร์รวมทั้งวงจรการคำนวณทางคณิตศาสตร์และลอจิก (ALU) วงจรรีจิสเตอร์ทั่วไป และรีจิสเตอร์ฟังก์ชันการใช้งานเฉพาะ

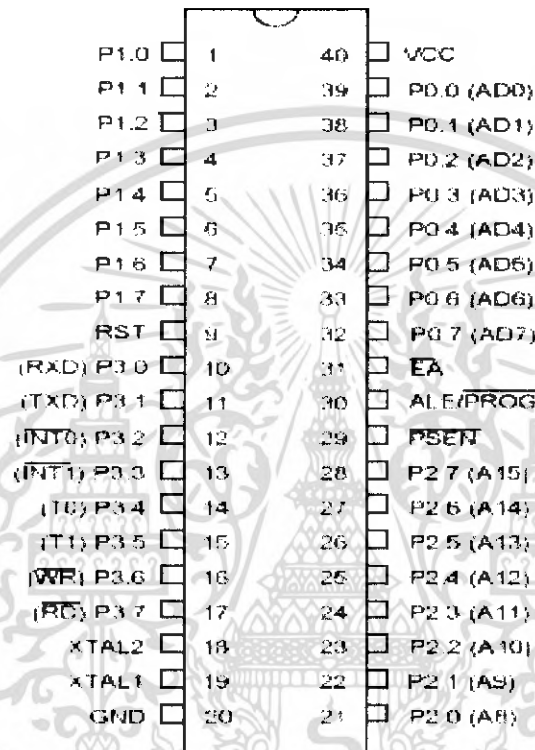


รูป 2.2 โครงสร้างภายในของไมโครคอนโทรลเลอร์ตระกูล MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 การจัดตำแหน่งขาของไมโครคอนโทรลเลอร์ตระกูล MCS-51(AT89C52)

ไมโครคอนโทรลเลอร์ตระกูล MCS-51 ทุกเบอร์จะมีขาใช้งานพื้นฐานที่เหมือนกัน ตัวอย่างเช่น แบบดิว (DIP) ซึ่งได้แบ่งการใช้งานออกเป็นขาอินพุตและเอาต์พุต ขาสัญญาณควบคุม ขาสัญญาณกำหนดตำแหน่งหน่วยความจำ และขาสัญญาณข้อมูลดังรูป 2.3



รูป 2.3 ตำแหน่งขาของไมโครคอนโทรลเลอร์ตระกูล MCS-51(AT89C52)

ตำแหน่งขาของไมโครคอนโทรลเลอร์ตระกูล MCS-51 และหน้าที่การทำงาน มีดังนี้

2.3.1 P0.0-P0.7 (ขาที่ 32-39) พอร์ต 0 ทำหน้าที่เป็นสัญญาณควบคุมอุปกรณ์นอกได้ 2 ทิศทาง สามารถรับข้อมูลอินพุตและส่งข้อมูลเอาต์พุตได้ มีขนาด 8 บิต การตั้งค่าให้พอร์ต 0 รับข้อมูลอินพุตทำได้ด้วยการส่งค่าสถานะ "1" ไปยังบิตที่ต้องการให้รับข้อมูลอินพุต วงจรภายในจะทำให้บิตนั้นมีค่าความต้านทานสูงและสามารถรับข้อมูลอินพุตได้ และยังใช้เป็นขาสัญญาณกำหนดตำแหน่งหน่วยความจำ (AO-A7) และขาสัญญาณข้อมูล (DO-D7) โดยการใช้ตัวแยกสัญญาณ (D-latch 74LS373) ทำหน้าที่เป็นมัลติเพล็กซ์ (multiplex) โดยเลือกช่วงเวลาของสัญญาณกำหนดตำแหน่งหน่วยความจำและสัญญาณข้อมูลออกจากกัน ในขณะที่ใช้เป็นพอร์ต

อินพุตและเอาต์พุต วงจรภายในจะไม่มีวงจรเพิ่มกระแสไฟฟ้า (pull up) จึงจำเป็นต้องต่อวงจรเพิ่มกระแสไฟฟ้าจากภายนอกเข้าไป

2.3.2 P1.0-P1.7 (ขาที่ 1-8) พอร์ต 1 ทำหน้าที่เป็นสัญญาณควบคุมอุปกรณ์ภายนอกได้ 2 ทิศทาง สามารถเป็นได้ทั้งอินพุตและเอาต์พุต มีขนาด 8 บิต สามารถอ้างถึงการทำงานได้ที่ละบิต และวงจรภายในมีตัวต้านทานเพิ่มกระแสไฟฟ้า (pull up) ในกรณีที่ต้องการให้รับข้อมูลอินพุตก็สามารถทำได้เหมือนพอร์ต 0

2.3.3 ขา P2.0-p2.7 (ขาที่ 21-28) พอร์ต 2 ทำหน้าที่เป็นสัญญาณควบคุมอุปกรณ์ภายนอกได้ 2 ทิศทาง คือ เป็นได้ทั้งอินพุตและเอาต์พุต มีขนาด 8 บิต สามารถใช้เป็นขาสัญญาณกำหนดตำแหน่งหน่วยความจำ (A8-A15) และมีวงจรเพิ่มกระแสไฟฟ้าในการกำหนดให้เป็นขาอินพุตทำได้โดยการส่งค่าข้อมูลสถานะ “1” ไปยังบิตที่ต้องการให้เป็นอินพุต ก็สามารถรับค่าข้อมูลอินพุตได้

2.3.4 ขา P3.0-3.7 (ขาที่10-17) พอร์ต 3 ทำหน้าที่ เป็นสัญญาณควบคุมอุปกรณ์ภายนอกอินพุตและเอาต์พุต 2 ทิศทาง มีขนาด 8 บิต คุณสมบัติทั่วไปจะเหมือนกับพอร์ตอื่น ๆ แต่จะมีคุณสมบัติที่ต่างออกไป คือ ใช้ทำหน้าที่พิเศษเป็นสัญญาณควบคุมการทำงานต่าง ๆ ของไมโครคอนโทรลเลอร์ ดังนี้

P3.0 มีหน้าที่ รับข้อมูลจากพอร์ตอนุกรม

P3.1 มีหน้าที่ ส่งข้อมูลจากพอร์ตอนุกรม

P3.2 มีหน้าที่ รับสัญญาณอินเทอร์รัพต์จากภายนอกตัวที่ 0

P3.3 มีหน้าที่ รับสัญญาณอินเทอร์รัพต์จากภายนอกตัวที่ 1

P3.4 มีหน้าที่ ใช้ตั้งเวลา/นับเวลาตัวที่ 0

P3.5 มีหน้าที่ ใช้ตั้งเวลา/นับเวลาตัวที่ 1

P3.6 มีหน้าที่ เป็นสัญญาณเขียนข้อมูลหน่วยความจำหรืออุปกรณ์ภายนอก

P3.7 มีหน้าที่ เป็นสัญญาณอ่านข้อมูลหน่วยความจำหรืออุปกรณ์ภายนอก

2.3.5 ขา PSEN (Program Store Enable ขาที่ 29) ขาที่ทำงานที่สถานะลอจิกเป็น “0” ไมโครคอนโทรลเลอร์ต้องอ่านค่าจากหน่วยความจำภายนอกที่เป็นข้อมูล โดยโปรแกรมจะเก็บในหน่วยความจำถาวร (ROM, EPROM, EEPROM) ส่วนมากใช้ต่อเป็นขาเลือกทำงาน (Enable : OE) แต่ถ้าไมโครคอนโทรลเลอร์ใช้หน่วยความจำภายในขานี้ก็จะไม่ได้ใช้งาน และมีค่าลอจิกเป็น “1”

2.3.6 ขา ALE (Address Latch Enable ขาที่ 30) ทำหน้าที่ ควบคุมการทำงานของสัญญาณ กำหนดตำแหน่งกับสัญญาณข้อมูล โดยใช้การเลือกเส้นทาง (data select หรือ multiplex) โดยปกติเมื่อไมโครคอนโทรลเลอร์ทำงานจะส่งสัญญาณกำหนดตำแหน่งออกมา ก่อนพร้อมกับส่งสัญญาณให้ขา ALE ทำงาน เพื่อเลือกให้สัญญาณกำหนดตำแหน่ง (A0-A7) ผ่าน ไอซี (74LS373) จะไม่ทำงาน ข้อมูลก็จะถูกส่งไปที่สายสัญญาณข้อมูล

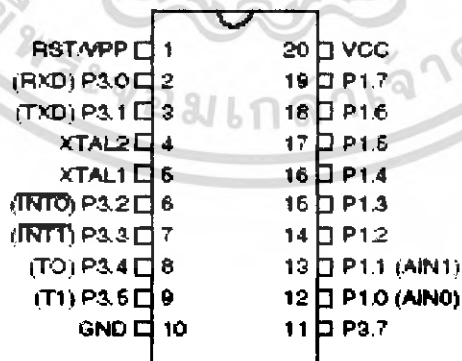
2.3.7 ขา EA (External Access ขาที่ 31) ทำหน้าที่ เลือกการทำงานของหน่วยความจำ ถ้าลอจิกเป็น "1" หมายถึง ใช้ข้อมูลจากหน่วยความจำภายในตัวไมโครคอนโทรลเลอร์ แต่ถ้ามีค่าลอจิกเป็น "0" หมายถึง ใช้ข้อมูลจากหน่วยความจำภายนอก

2.3.8 ขา PST (RESET ขาที่ 9) ทำหน้าที่ เริ่มต้นการทำงานใหม่ของไมโครคอนโทรลเลอร์ การทำงานของลอจิก "1" นี้จะทำให้ไมโครคอนโทรลเลอร์เริ่มต้นทำงานที่ตำแหน่ง 0000 เพื่ออ่านข้อมูลโปรแกรมและจัดระบบการทำงาน

2.3.9 ขาสัญญาณนาฬิกา (ขาที่ 18-19) ทำหน้าที่ เป็นตัวกำหนดสัญญาณนาฬิกาให้กับไมโครคอนโทรลเลอร์ใช้เป็นฐานเวลาในการทำงาน โดยใช้แผ่นผลึก (crystal) ที่มีความถี่ตั้งแต่ 0-24 เมกะเฮิรตซ์ (MHz) ร่วมกับตัวเก็บประจุขนาด 20-33 pF

2.3.10 แหล่งจ่ายไฟ (Power Supply) ขาที่ 20 จะเป็นขากราวด์ (Ground) และขาที่ 40 จะเป็นแหล่งจ่ายไฟบวกให้กับไมโครคอนโทรลเลอร์ ซึ่งใช้แหล่งจ่ายไฟฟ้าขนาดไม่เกิน 5 โวลต์

2.4 การจัดตำแหน่งขาของไมโครคอนโทรลเลอร์ตระกูล MCS-51(AT89C2051)



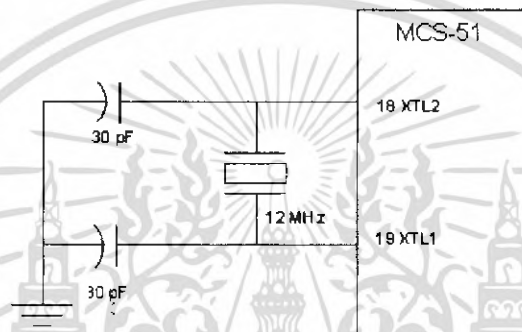
รูป 2.4 การจัดตำแหน่งขาของ MCS-51(AT89C2051)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการจัดวางตำแหน่งขาของ MCS-51(AT89C2051) ซึ่งมีอยู่ 20 ขาจะแตกต่างจาก AT89C52 คือมีเพียง P1.0-P1.7 และ P3.0-P3.5 และจะข้ามไป P3.7 ซึ่งไม่มี P3.6 ซึ่งการทำงานคล้ายกัน และจะมีหน่วยความจำภายในจำนวน 2k

2.5 สัญญาณนาฬิกาของ MCS-51

เนื่องจากไมโครคอนโทรลเลอร์ตระกูล MCS-51 มีวงจรสร้างสัญญาณนาฬิกาอยู่ภายใน เพื่อสร้างสัญญาณนาฬิกาให้กับระบบ ทำให้ระบบทำงานสอดคล้องกันทั้งหมด เพียงแต่ต่อคริสตัลและตัวเก็บประจุเข้าไป

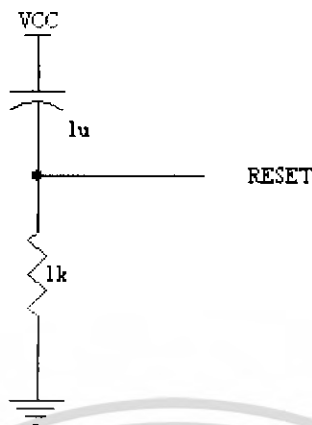


รูป 2.5 การต่อคริสตัล

ลักษณะการใช้งาน คริสตัล จะทำหน้าที่ ผลิตความถี่พินคาเมนทอล เสมือนเป็นตัวเหนี่ยวนำ โดยต่ออยู่กับตัวเก็บประจุภายนอก การกำหนดค่าของคริสตัลและตัวเก็บประจุ C1, C2 ไม่ได้มีค่าที่กำหนดแน่นอนอนตยตัวมากนัก อาจจะมีค่าประมาณ 30 pF สำหรับทุกความถี่ของตัวคริสตัลที่มีคุณภาพดี แต่หากใช้เซรามิกเรโซเนเตอร์ ค่าของ C จะมีค่าสูงกว่าโดยประมาณ 47 pF การใช้ค่าตัว C อาจเปลี่ยนแปลงได้ขึ้นอยู่กับตัวเซรามิกเรโซเนเตอร์นั้น ๆ

2.6 การรีเซต

การรีเซตไมโครคอนโทรลเลอร์ตระกูล MCS-51 เป็นการหน่วงเวลาช่วงสั้น ๆ เพื่อให้ CPU มีเวลาสำหรับจัดการกับวงจรทั้งหมดภายในชิป การรีเซตนั้นเป็นสิ่งจำเป็น ในกรณีเมื่อเริ่มจ่ายพลังงานหรือเมื่อระบบเกิดการทำงานผิดพลาด ทั้งนี้เพื่อให้ CPU สามารถเริ่มต้นทำงานใหม่ได้อย่างถูกต้อง



รูป 2.6 การต่อชุดรีเซ็ต

การรีเซ็ต MCS-51 ทำได้โดยการบังคับให้ขา RST มีสถานะเป็น 1 เป็นเวลาอย่างน้อย 2 เมกซีไนซ์เกิด โดยในขณะที่ขา RST มีสถานะเป็น 1 วงจรออสซิลเลเตอร์จะต้องทำงานอยู่ด้วย เมื่อ MCS-51 ได้รับสัญญาณ Reset CPU ภายใน MCS-51 จะตอบสนองโดยการ Reset วงจรภายใน โดยจากรูปที่ 2.5 จะพบว่าถ้ายังไม่มีการกดสวิตช์สัญญาณ RST จะเป็นลอจิก “0” และเมื่อมีการกดสวิตช์จะทำให้สัญญาณ RST เป็นลอจิก “1”

2.7 โครงสร้างโดยทั่วไปของ LCD มอดูล

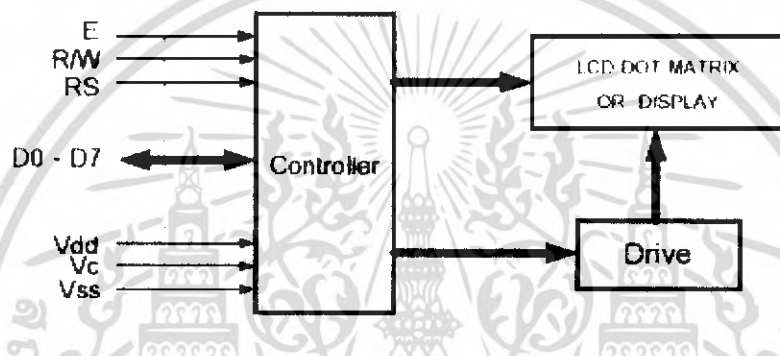
หน่วยแสดงผลแบบ LCD (Liquid Crystal Display) ที่มีโครงสร้างแบบคอตเมทริกซ์เป็นอุปกรณ์ชนิดหนึ่งที่นิยมนำมาใช้งาน เนื่องจากใช้พลังงานต่ำ เป็นอุปกรณ์มีขนาดเล็ก มีความละเอียดสูง สามารถแสดงตัวอักษรและรูปภาพได้หลายแบบ และการที่จะทำให้ LCD แต่หลอสดว้างเป็นรูปร่างหรือตัวอักษรต่าง ๆ ได้ นั้น จะต้องใช้วงจรขั้วที่มีความซับซ้อนมาก ซึ่งในปัจจุบันจึงได้มีการสร้างบอร์ด LCD สำเร็จรูป (LCD Module) เพื่อให้สะดวกต่อการใช้งาน ดังรูปที่ 2.6 โดยที่มอดูล LCD จะมีส่วนประกอบหลัก ๆ 3 ส่วนดังนี้

2.7.1 ส่วนแสดงผล (Display) หรือ LCD DOT MATRIX ซึ่งจะประกอบด้วยจุดขนาดเล็ก ๆ จำนวนมาก โดยแต่ละจุดจะถูกบังคับให้ติดสว่างเป็นรูปต่าง ๆ ด้วยวงจรขั้ว โดยที่ภายในเป็นผลึกเหลวที่สามารถแสดงผลให้เห็นโดยอาศัยแสงจากภายนอก ดังนั้นจะต้องมีมุมในการมองข้อมูลที่แสดงผลบนจอ LCD

2.7.2 ส่วนควบคุม (Controller) จะประกอบด้วยหน่วยความจำและรีจิสเตอร์ต่าง ๆ ซึ่งทำหน้าที่กำหนดลักษณะการทำงานของ LCD หรือ รูปภาพของการแสดงผล รวมทั้งควบคุมส่วน

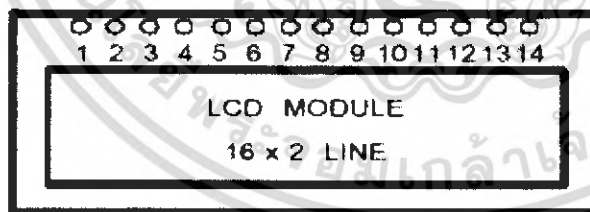
ต่าง ๆ ภายในมอดูล ส่วนควบคุมนี้จะใช้ชิปควบคุมโดยเฉพาะชิปที่นิยมใช้ก็คือ เบอร์ส่วนต่าง ๆ ภายในมอดูล ส่วนควบคุมนี้จะใช้ชิปควบคุมโดยเฉพาะชิปที่นิยมใช้ก็คือ เบอร์ HD44780 จะใช้ควบคุม LCD แบบอักษร

2.7.3 ส่วนขับ (Drive) เป็นตัวรับสัญญาณจากตัวควบคุมมาขับให้แสดงผลข้อมูลตามที่กำหนด ชิปที่ทำหน้าที่เป็นตัวขับได้แก่ เบอร์ HD44100 และ MSM5259 เป็นต้น



รูป 2.7 โครงสร้างโดยทั่วไปของ LCD มอดูล

2.8 มอดูล LCD ขนาด 16 ตัวอักษร 2 บรรทัด (LCD 16 x 2)



รูป 2.8 รูปร่างและการจัดขามอดูล LCD แบบอักษร

มอดูล LCD ขนาด 16 x 2 มีขาต่อใช้งานทั้งสิ้น 14 ขามีการจัดขาดังนี้

- Vss (ขา 1) : ขานี้จะต่อกับกราวด์ของแหล่งจ่ายไฟ
- Vdd (ขา 2) : ขานี้จะต่อกับแหล่งจ่ายไฟ + 5 โวลต์
- Vc (ขา 3) : ขานี้จะเป็นขาที่ควบคุมความสว่างของ LCD
- RS (ขา 4) : ขานี้จะเป็นขาที่ใช้ควบคุมรีจิสเตอร์ภายในหน่วยควบคุม เนื่องจากมีรีจิสเตอร์อยู่สองประเภท คือรีจิสเตอร์คำสั่ง (Command Register) และรีจิสเตอร์





เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูล (Data Register) โดยที่ ถ้าหากขานี้เป็นลอจิก “0” หมายความว่า เลือก
รีจิสเตอร์คำสั่ง และถ้าหากขานี้เป็นลอจิก “1” หมายความว่า เลือกรีจิสเตอร์ข้อมูล

- R/W (ขา 5) : ขานี้จะเป็นขาที่ใช้เลือกว่าจะทำการอ่านหรือเขียนข้อมูลให้กับมอดูล
LCD โดยที่ ถ้าหากขานี้เป็นลอจิก “0” หมายความว่า จะเป็นการอ่านข้อมูล
- E (ขา 6) : ขานี้จะเป็นขาสำหรับรับสัญญาณพัลส์เอ็นเอเบิลมอดูล LCD ให้ทำงาน
- D0-D7 (ขา 7-14) : ขานี้จะเป็นขาที่ใช้เป็นทางผ่านของข้อมูลระหว่าง LCD กับ
อุปกรณ์ภายนอกขนาด 8 บิต

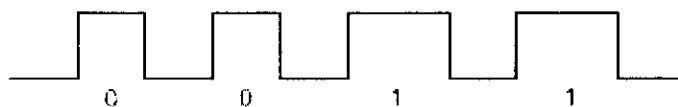
อนึ่งขา RS,R/W และ E จะถูกเลือกใช้งาน ดังตาราง 2.1

ตาราง 2.1 ความสัมพันธ์ในการทำงานของขา RS,R/W และ E

RS	R/W	E	การทำงาน
0	0		เขียนคำสั่ง
0	1		อ่านสถานะของโมดูล
1	0		เขียนข้อมูล
1	1		อ่านข้อมูล

2.9 หลักการทำงานของรีโมตคอนโทรล

รีโมตคอนโทรลประเภทที่ใช้แสงอินฟราเรด ทุกชนิด จะมีการส่งข้อมูลหรือสัญญาณพัลส์
ผ่านตัวส่งอินฟราเรด (LEDIR) โดยจะใช้ค่าความถี่ในช่วง 30-40 กิโลเฮิร์ตซ์ เหตุที่ใช้ค่าความถี่
สูงเพื่อเป็นการป้องกันแสงจากแหล่งกำเนิดแสงแหล่งอื่นๆ มารบกวน โดยสัญญาณที่ส่งมาจาก
รีโมตคอนโทรลจะมีรูปแบบเหมือนเลขฐานสอง และเนื่องจากผู้ผลิตรีโมตคอนโทรลมีอยู่หลาย
บริษัทฯ ซึ่งจะทำให้การเข้ารหัสของรีโมดแต่ละยี่ห้อไม่เหมือนกันแต่อยู่ในพื้นฐานเดียวกันคือจะ
ให้รหัสเลขฐานสองและความกว้างของพัลส์ในช่วง “High” และในช่วง “Low” เป็นตัว
กำหนดการทำงาน อย่างเช่น ดังรูปที่ 2.8 จะเป็นการใช้ความกว้างของพัลส์กำหนดรหัส (Pulse-
Width-Code Signals) โดยกำหนดความกว้างของพัลส์แทนลอจิก “0” หรือ “1” เป็นต้น

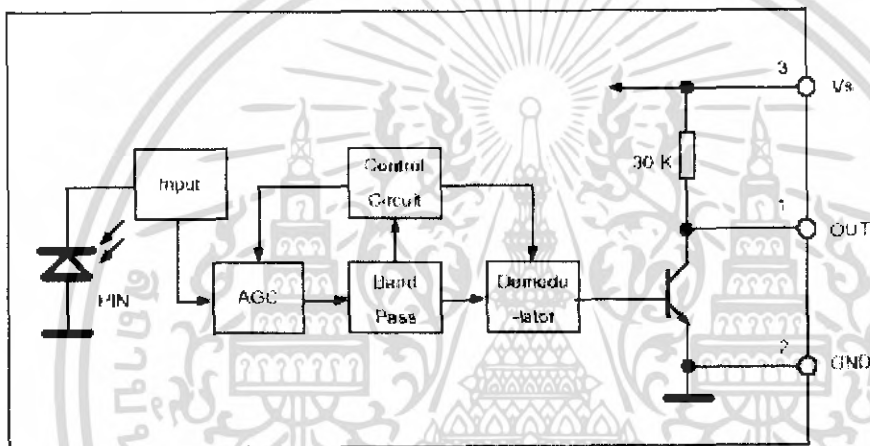


รูป 2.9 รูปแบบของสัญญาณพัลส์ที่มีการเข้ารหัสโดยใช้ความกว้างพัลส์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10 หลักการทำงานของโฟโตไดโอดมอดูล

ตัวรับอินฟราเรดจะแบ่งเป็น 2 ชนิดใหญ่ ๆ คือ ตัวรับแสงอินฟราเรดแบบธรรมดา และ ตัวรับแสงอินฟราเรดแบบมอดูล ซึ่งในโครงงานนี้เราใช้ตัวรับแสงอินฟราเรดแบบมอดูลหรือที่เรียกว่ามอดูลภาครับอินฟราเรดเบอร์ TSOP4838 สามารถตอบสนองความถี่ได้ดีที่สุดที่ 38 กิโลเฮิร์ตซ์ และโครงสร้างการทำงานภายในของมอดูลภาครับอินฟราเรด ที่เราเรียกว่ามอดูลนั้น เนื่องจากภายในจะประกอบด้วยส่วนของการทำงานต่าง ๆ มากกว่าหนึ่งส่วนจากรูปที่ จะเห็นว่า ภายในตัวมันจะมีตัวรับแสงอินฟราเรดแบบธรรมดาบรรจุอยู่ด้วยนอกจากนั้นก็จะมีวงจรปรีแอมป์, วงจรควบคุมและวงจรมอดูเลเตอร์บรรจุร่วมอยู่ด้วย

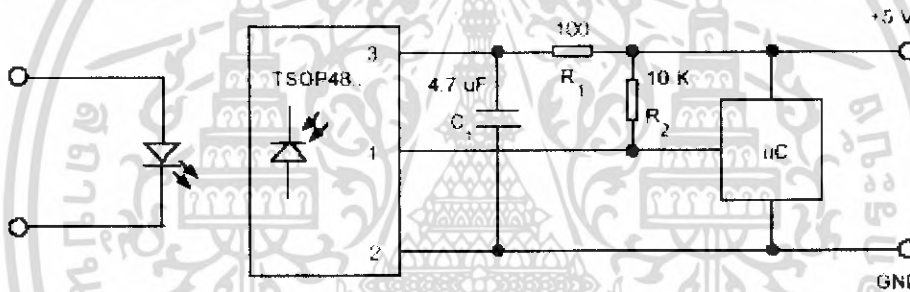


รูป 2.10 บล็อกไดอะแกรมของวงจรมอดูลภาครับอินฟราเรด

การทำงานโดยรวมของมอดูลภาครับอินฟราเรดแสดงไว้ในรูปที่ 2.9 คือเมื่อตัวรับแสงอินฟราเรดได้รับแสง ก่อนอื่นสัญญาณทั้งหมดจะถูกขยายโดยวงจรปรีแอมป์ และจากนั้นสัญญาณที่ผ่านการขยายแล้วจะมาควบคุมอัตราขยายอัตโนมัติหรือ (AGC) แล้วผ่านวงจรรองความถี่ผ่านและถูกดีมอดูเลเตอร์แปลงสัญญาณความถี่ออกจากสัญญาณข้อมูล โดยมีอินเวอร์เตอร์ที่ทำหน้าที่โดยทรานซิสเตอร์แปลงสัญญาณข้อมูลนั้นออกมาเป็นแอกทีฟ “0” ออกทางเอาต์พุตขา 1 ในส่วนของไดโอดอินฟราเรดจะมีฟิลเตอร์กรองแสงอินฟราเรดประกอบรวมอยู่บนตัวตั้งด้วย เพื่อเป็นการป้องกันแสงรอบ ๆ ข้างที่มีความเข้มของแสงใกล้เคียงกับแสงของอินฟราเรดไม่ให้สามารถผ่านแผ่นกรองแสงเข้ามารบกวนการทำงานของโฟโตมอดูลนี้ได้ อันอาจจะก่อให้เกิดความผิดพลาดของข้อมูลที่จะถูกดีมอดูเลเตอร์ออกทางเอาต์พุต และจากการที่มีอินเวอร์เตอร์ก่อนขับสัญญาณออกทางเอาต์พุตนี้เอง จึงทำให้สามารถต่อใช้งานร่วมกับวงจร TTL และคอม CMOS ได้โดยตรง

2.11 ลักษณะของโฟโต้ไดโอดคอมมอดูล

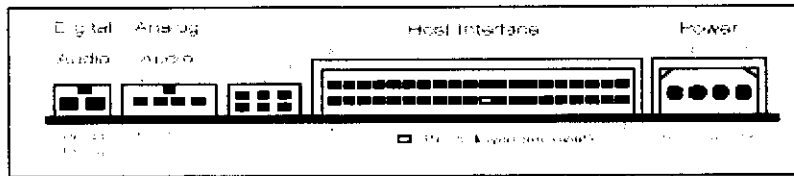
- มีตัวตรวจจับแสงและวงจรปรีแอมป์ บรรจุอยู่ในตัวถังเดียวกัน
- ให้สัญญาณเอาต์พุตออกมาแอกทีฟที่สถานะลอจิก “0”
- มีวงจรฟิลเตอร์สำหรับความถี่ PCM ภายใน
- มีความปลอดภัยจากแสงโคจรอบสูง
- มีชิลด์ป้องกันการรบกวนจากสนามไฟฟ้า
- สิ้นเปลืองพลังงานต่ำ
- สามารถต่อใช้งานร่วมกับอุปกรณ์ไอซี TTL และ CMOS ได้โดยตรง



รูป 2.11 การประยุกต์ใช้งานกับไมโครคอนโทรลเลอร์

ในรูป 2.11 นั้นจะเป็นการประยุกต์ใช้งานของโฟโต้ไดโอดคอมมอดูลโดยต่อร่วมกับไมโครคอนโทรลเลอร์ เมื่อพิจารณาจากวงจรก็จะสังเกตได้ว่าอุปกรณ์ R_1 และ C_1 จะต่อไว้เพื่อป้องกันและลดปัญหาอันเนื่องมาจากการรบกวนทางสนามไฟฟ้า หรือเกิดการกระชากของกระแสที่เกิดจากอุปกรณ์เครื่องใช้ไฟฟ้าอื่น ๆ ทำให้มีการรบกวนผ่านเข้ามาในระบบของแหล่งจ่ายแรงดันไฟเลี้ยงของโฟโต้ไดโอดคอมมอดูล ส่วนตัวต้านทาน R_2 เป็นตัวต้านทานพูลอัพและจะต้องมีค่าไม่ต่ำกว่า 10 กิโลโอห์ม ตามความเหมาะสมของค่าแรงดันไฟเลี้ยงและระดับลอจิกอินพุตของไมโครคอนโทรลเลอร์

2.12 โครงสร้างส่วนเชื่อมต่อด้านหลังของซีดีรอม



รูป 2.12 โครงสร้างส่วนเชื่อมต่อด้านหลังของซีดีรอม

โครงสร้างด้านหลังของซีดีรอมไครว์ ดังรูป 2.12 แสดงส่วนเชื่อมต่อด้านหลังของซีดีรอมไครว์ ซึ่งจะมีส่วนที่ใช้ในการเชื่อมต่ออยู่ดังต่อไปนี้

- ช่องเสียบสาย Power เป็นจุดที่จะต้องนำสายไฟจากเพาเวอร์ซัพพลายมาจ่ายไฟให้กับซีดีรอมไครว์
- ช่องเสียบสัญญาณ IDE ซึ่งจะใช้ในการเชื่อมต่อระหว่างซีดีรอมกับคอนโทรลเลอร์
- ช่องสำหรับเซตจัมเปอร์ (Jumper Set) เพื่อกำหนดการทำงานของซีดีรอมให้ทำงานเป็นเครื่องหลัก (Master) หรือ เครื่องรอง (Slave)
- ช่องเสียบสัญญาณ Audio ใช้ในการส่งสัญญาณเสียงจากซีดีรอมไปยังลำโพงหรืออาจจะเข้าภาคขยายก่อนที่จะไปยังลำโพง ก็ได้

2.13 ไฟเลี้ยงที่จ่ายให้แก่ซีดีรอม

ไฟเลี้ยง 5 V และ 12 V จะจ่ายให้กับแต่ละส่วนภายในซีดีรอม เช่น มอเตอร์ และภาพวงจรควบคุม

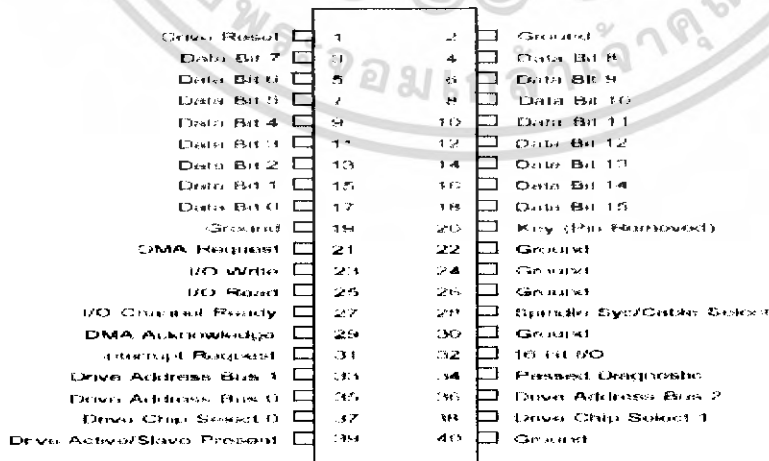
ขา	ไฟเลี้ยง
1	12 VDC
2	GND
3	GND
4	5 VDC

ตาราง 2.2 ไฟเลี้ยงที่จ่ายให้แก่ซีดีรอม

2.14 มาตรฐานการติดต่อกับไดรฟ์

ATA / IDE เป็น IDE Drive โดยหากพูดถึง ซีดีรอมไดรฟ์ โดยปราศจากการระบุเจาะจงชนิดของไดรฟ์แสดงว่าเขามายถึง ATA / IDE Drive โดยที่ ATA / IDE interface ใช้สาย Pair พร้อมด้วยคอนเน็กเตอร์จำนวน 40 ขา (บางที่เรียกว่า Header Type Connector) จะต้องมีความยาวไม่เกิน 18 นิ้ว และถูกใช้เป็นสายที่นำพา สัญญาณจาก IDE Drive/Controller ไปยัง ซีดีรอม ซึ่งเป็นมาตรฐานของ ANSI

- ขา Drive Reset (ขา 1) สัญญาณที่ขานี้จะทำให้ซีดีรอมกลับเข้าสู่สถานะ Power On ใช้เพื่อ กำหนดสถานะเริ่มต้นของอุปกรณ์ต่างๆ ที่อยู่บน ซีดีรอม
- ขา Ground (ขา 2, 19, 22, 24, 26, 30, 40) ขานี้เป็นขาสัญญาณ Ground ของซีดีรอม
- ขา Data (ขา 3-18) ขานี้จะมีทั้งหมด 16 ขา โดยมีขาที่ 3 ไปจนถึงขาที่ 18 มีไว้เพื่อการนำพาข้อมูลขนาด 16 บิต จากไมโครคอนโทรลเลอร์ไปยังซีดีรอมไดรฟ์
- ขา I/O Write (ขา 23) และ I/O Read (ขา 25) ในระหว่างที่มีการส่งผ่านข้อมูลนั้น สัญญาณ I/O Write ที่ขา 23 จะถูกใช้เพื่อบ่งบอกว่า ข้อมูลที่กำลังปรากฏอยู่บนบั๊สสามารถเขียนให้กับซีดีรอมได้ ส่วนสัญญาณ I/O Read ที่ขา 25 มีไว้เพื่อบอกให้ทราบว่าข้อมูลบั๊สขณะนี้เต็มไปด้วยตาต้า หลังจากที่มีการสั่งให้อ่านซีดีรอมแล้ว
- ขา DA0, DA1, DA2 (ขาที่ 35, 33, 36) เป็นสัญญาณที่มาจาก Bus Address เป็นสัญญาณที่มีไว้เพื่อเรียก Register ตัวใดตัวหนึ่งที่อยู่ภายในจากซีดีรอมไดรฟ์ ออกมาใช้งาน ซึ่งได้แก่ Command Register หรือ Control Register
- ขา CS0, CS1 (ขาที่ 37, 38) สัญญาณนี้มีไว้เพื่อใช้เลือกชุดรีจิสเตอร์ที่เป็น Control Block Register หรือ Command Block Register



รูป 2.13 ขาสัญญาณของ ATA/IDE Connector

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.15 รีจิสเตอร์ที่ใช้ในการควบคุมการทำงานของซีดีรอม

การที่จะควบคุมการทำงานหรือติดต่อสื่อสารกับ ซีดีรอมนั้นจำเป็นที่จะต้องมีการใช้รูปแบบฟอร์มที่มาจากมาตรฐานที่ได้กำหนดเอาไว้ โดยจะต้องติดต่อผ่านรีจิสเตอร์ที่ระบุตำแหน่งรวมทั้งการรับและการส่งข้อมูล สามารถเลือกบล็อกของ Control Block Register และ Command Block Register ได้รับสัญญาณ CS1, CS0 และเลือกรีจิสเตอร์ภายในแต่ละบล็อกได้จากสัญญาณ DA2, DA1, DA0 และยังสามารถเขียนหรืออ่านข้อมูลจากรีจิสเตอร์ได้จากสัญญาณ DIOW และ DIOB ดังตาราง 2.3

ตาราง 2.3 รูปแบบตำแหน่งของรีจิสเตอร์ I/O

Addresses					Functions	
CS1	CS0	DA2	DA1	DA0	(DIOR)	(DIOB)
					Control Block Register	
N	A	0	0	0	FloppyA Status	Unused
N	A	0	0	1	FloppyB Status	Unused
N	A	0	1	0	Unused	Floppy Digital Output
N	A	0	1	1	Floppy ID/Tape Control	RESERVED
N	A	1	0	0	Floppy Controller Status	RESERVED
N	A	1	0	1	Floppy Data Register	
N	A	1	1	0	Alternate ATAPI Status	Device Control
N	A	1	1	1	Not Used	Not Used
					Command Block Register	
A	N	0	0	0	Data	
A	N	0	0	1	ATAPI Error Register	ATAPI Feature
A	N	0	1	0	ATAPI Interrupt Reason	Unused
A	N	0	1	1	Reserved	
A	N	1	0	0	ATAPI Byte Count Register (Bits 0-7)	
A	N	1	0	1	ATAPI Byte Count Register (Bits 8-15)	
A	N	1	1	0	Drive Select	
A	N	1	1	1	ATAPI Status	ATA Command

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อ A (Signal Asserted) เป็นค่าลอจิก “1”

N (Signal Negated) เป็นค่าลอจิก “0”

2.16 รายละเอียดภายในแต่ละส่วนของรีจิสเตอร์

รายละเอียดภายในแต่ละส่วนของรีจิสเตอร์ ที่ใช้ในโครงงานนี้จะประกอบไปด้วย ตาราง 2.4 ถึง ตาราง 2.6

ตาราง 2.4 ATAPI STATUS REGISTER

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

Bit7 BSY บิตนี้จะถูกเช็คว่าพร้อม หรือไม่พร้อมในการรับ-ส่งข้อมูล

ตาราง 2.5 ATAPI BYTE COUNT REGISTER

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

Bit7 - Bit0 ใช้ในการกำหนดจำนวนไบต์ในการรับ-ส่งข้อมูลทางด้านต่ำ

Bit15- Bit8 ใช้ในการกำหนดจำนวนไบต์ในการรับ-ส่งข้อมูลทางด้านสูง

ตาราง 2.6 ATAPI DRIVE SELECT REGISTER

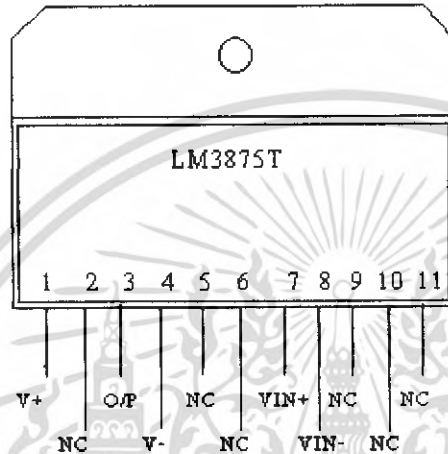
D7	D6	D5	D4	D3	D2	D1	D0
1	Reserved	1	DRV	Reserved			

Bit4 DRV บิตนี้ใช้เลือก Device0 (Master) หรือ Device 1 (Slave)

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

2.17 วงจรขยายเสียง (POWER AMPLIFIER)

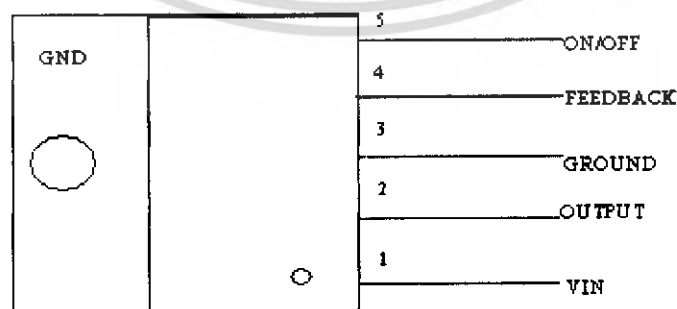
เนื่องจากสัญญาณเสียงจากที่ออกจากซีดีรอมนั้นมีกำลังขับลำโพงไม่มากจึงต้องสร้างเพาเวอร์แอมป์ขึ้นมา ซึ่งได้เลือกใช้ไอซีเพาเวอร์แอมป์เบอร์ LM3875T ของบริษัทเนชั่นแนลเซมิคอนดักเตอร์ คุณสมบัติโดยทั่วไปของไอซีตัวนี้คือ ให้กำลังขับเสียงได้ 56วัตต์ ใช้กับลำโพง 8 Ohm ลักษณะการวางขาใช้งานของไอซีเป็นดังรูป



รูป 2.14 ไอซีเพาเวอร์แอมป์เบอร์ LM3875T

2.18 แหล่งจ่ายไฟให้กับซีดีรอมและ วงจรควบคุม

เนื่องจากซีดีรอมต้องการไฟ 5 โวลต์ 1.5แอมป์ไปเลี้ยงวงจร และ 12 โวลต์ 1.5แอมป์ไปขับส่วนของมอเตอร์ ส่วนวงจรที่สร้างขึ้นเพื่อใช้ควบคุมการทำงานทั้งหมดของโครงงานนี้ก็ใช้ไฟ 5 โวลต์เช่นกัน ดังนั้นเราจึงต้องการไฟ 5 โวลต์และ 12 โวลต์ 1.5แอมป์ เลือกใช้เรกติเพอร์ไอซีเบอร์ LM2576HVT-5 และ LM2576HVT-12 ซึ่งทั้งสองสามารถจ่ายกระแสไฟได้ 3 แอมป์ เป็นตัวรักษาแรงดันและปรับแรงดันไฟ ลักษณะการวางขาใช้งานของไอซีเป็นดังรูป

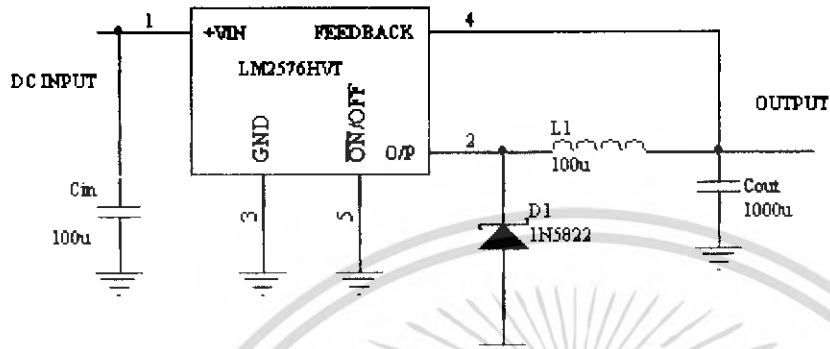


รูป 2.15 ลักษณะการวางขาใช้งานของไอซี LM2576HVT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเอกสารอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

62752

การประยุกต์ LM2576HVT ใช้งานทั่วไป



รูป 2.16 การประยุกต์ LM2576HVT ใช้งานทั่วไป

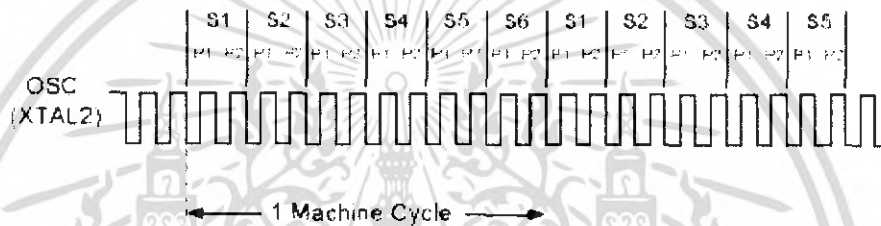
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบวงจร

3.1 ส่วนฐานเวลาของ MCS-51

เนื่องจากระบบไมโครคอนโทรลเลอร์ทำงานได้นั้นจะต้องมีฐานเวลาและสัญญาณนาฬิกาเป็นตัวควบคุมการทำงาน ซึ่งจะต้องออกแบบให้วงจรออสซิลเลเตอร์สำหรับสร้างสัญญาณนาฬิกาให้กับมัน โดยปกติแล้วไมโครคอนโทรลเลอร์ตระกูล MCS-51 จะมีวงจรออสซิลเลเตอร์อยู่ภายใน การกำหนดจังหวะเวลาก็ทำได้เพียงแค่ต่อคริสตัลภายนอกให้กับมันเท่านั้น



รูป 3.1 สัญญาณนาฬิกา และแมชชีน ไซเคิล

จากรูป 3.1 ถ้าไมโครคอนโทรลเลอร์ใช้ฐานเวลาความถี่ที่มีความถี่ 12 MHz และหนึ่งแมชชีน ไซเคิลต้องใช้สัญญาณนาฬิกา 12 ลูกจะทำให้หนึ่งรอบคำสั่งต้องใช้เวลา 1 ไมโครวินาทีหรือความเร็วในการทำงานเท่ากับ 1 MHz แต่ถ้าหากให้ใช้ความถี่สัญญาณนาฬิกาสูงขึ้น ก็จะทำให้ระบบไมโครคอนโทรลเลอร์ ทำงานได้เร็วขึ้น

และสามารถที่จะคำนวณหาค่าเวลาที่ใช้ในการกระทำคำสั่งใด ๆ จนเสร็จ โดยจะต้องดูว่าคำสั่งนั้นใช้จำนวนแมชชีน ไซเคิลเป็นเท่าไรในการประมวลผล ซึ่งสามารถหาเวลาที่ใช้ได้จากสูตร

$$T = \frac{C \times 12}{\text{Crystal Frequency}} \quad (3.1)$$

โดยที่ C เป็นค่าจำนวนแมชชีน ไซเคิลของคำสั่ง
Crystal Frequency เป็นค่าความถี่ของคริสตัลที่ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ส่วนชุดรีเซต

การออกแบบส่วนชุดรีเซตนั้นมีความจำเป็นอยู่เหมือนกัน เพราะการรีเซตนั้นจะเป็นการจัการเพื่อให้ระบบไมโครคอนโทรลเลอร์พร้อมที่จะทำงานเริ่มต้นได้อย่างถูกต้อง เนื่องจากการทำงานในระบบไมโครคอนโทรลเลอร์เกิดการผิดพลาดขึ้น หรือใช้จัดการระบบเมื่อเริ่มต้นง่าย พลังงาน โดยมีสิ่งที่ควรคำนึงถึง ก็คือจะต้องใช้เวลาให้เหมาะสมไม่มากหรือน้อยเกินไป และต้องคำนึงถึงอุปกรณ์รอบข้างด้วย

โดยเวลาวงรอบแมชชีนสามารถคำนวณได้จากสัญญาณนาฬิกา อย่างเช่นถ้าระบบไมโครคอนโทรลเลอร์ ต้องการใช้คริสตอล 12 MHz จะได้วงจรีเซตดังนี้

$$\text{จากความต้องการของขา RST} = 24 \text{ คาบของสัญญาณออสซิลเลเตอร์} \quad (3.2)$$

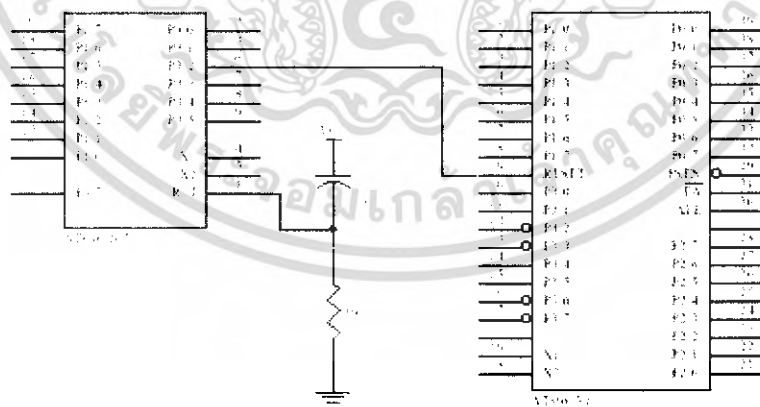
$$\text{และ 1 คาบของสัญญาณออสซิลเลเตอร์} = \frac{1}{12} \text{ MHz} \quad (3.3)$$

$$= 83.3 \text{ nSec}$$

$$\text{จะได้} = 83.3 \text{ nSec} * 24$$

$$= 2 \text{ uSec}$$

ดังนั้นจะได้เวลาในการรีเซตอย่างน้อยคือ 2 uSec วงจรีเซตที่ออกแบบจะใช้ RC time constant ซึ่งสามารถเลือกใช้ค่าของ R และ C ได้ตามความเหมาะสม ในที่นี้จะเลือกใช้ค่าตามดังรูป 3.2

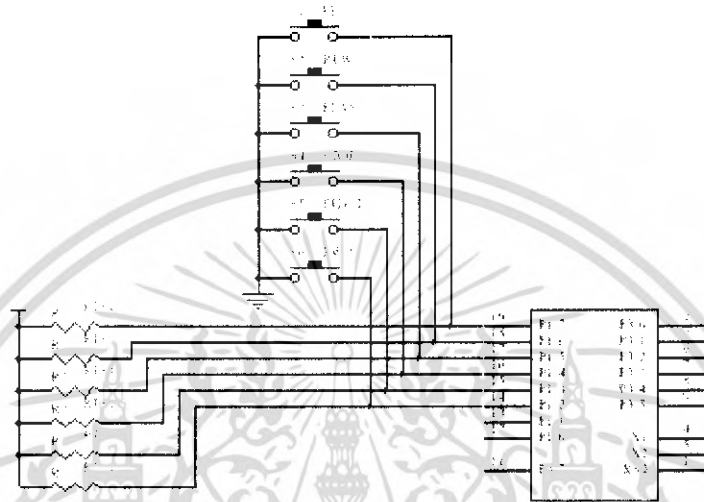


รูป 3.2 การต่อชุดรีเซต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 ส่วนควบคุมการทำงานโดยใช้สวิทช์ 6 ตัว

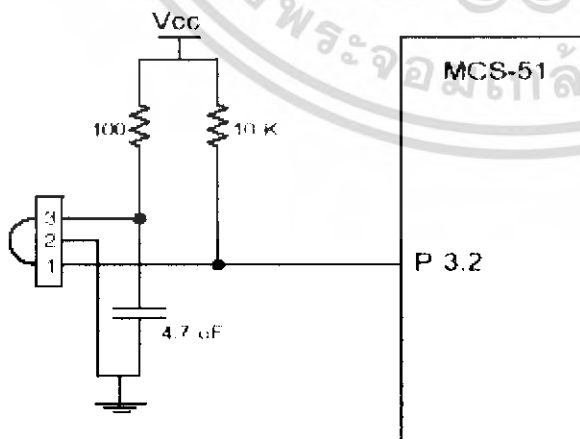
เนื่องจากมีพอร์ตของไมโครคอนโทรลเลอร์เหลืออยู่ 5 พอร์ต ดังนั้นจึงได้ทำการเพิ่มฟังก์ชันขึ้นมาโดยการต่อสวิทช์ 6 ตัว ซึ่งมีการใช้ฟังก์ชันที่จำเป็นโดยทั่วไป เช่น Play, Stop, Next, Back, Eject และ Power



รูป 3.3 ส่วนควบคุมการทำงานโดยใช้สวิทช์ 6 ตัว

3.4 ส่วนชุดรับสัญญาณรีโมตคอนโทรล

เนื่องจากในโครงการนี้ใช้รีโมตคอนโทรลของ Creative เป็นตัวส่ง ดังนั้นจึงต้องมีตัวรับสัญญาณทางด้านภากรับอินฟราเรด โดยจะใช้ตัวรับแสงอินฟราเรดแบบมอดูล เบอร์ TSOP4838 สามารถตอบสนองความถี่พาห้ได้ดีที่สุดที่ 38 กิโลเฮิร์ตซ์ และใช้ขาด้านเอาต์พุตของตัวรับแสงอินฟราเรด ต่อเข้ากับขา P3.2 ดังรูป 3.4

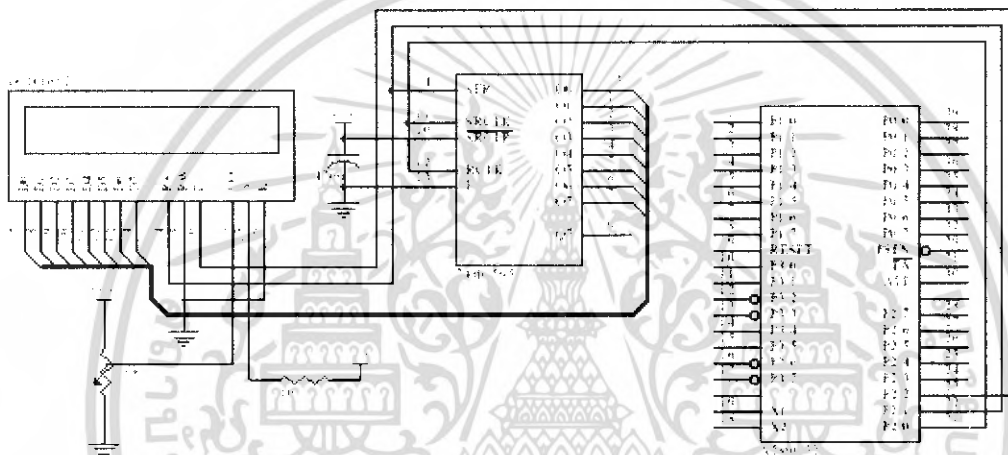


รูป 3.4 ส่วนชุดรับสัญญาณรีโมตคอนโทรล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 ส่วนแสดงผล LCD

การออกแบบส่วนแสดงผล LCD จะต้องพิจารณาว่าจะใช้แสดงผลแบบไหนและต้องการที่จะใช้ในการแสดงผลกี่บรรทัด ในโครงการนี้จะใช้ LCD (16 x 2) แบบอักษร แล้วจะมาพิจารณาถึงการเชื่อมต่อใช้งานเข้ากับระบบไมโครคอนโทรลเลอร์ โดยจากรูปที่ 3.5 จะพบว่าใช้พอร์ต P1.0-P1.7 ซึ่งทำหน้าที่เป็นทางผ่านของข้อมูลและคำสั่ง LCD และมีตัวควบคุมการทำงานของ LCD อีก 3 เส้น คือ ขา RS, และ ขา E จะต่อกับพอร์ต P3.0 และ P3.1 โดยเรียงตามลำดับ ส่วนขา R/W นั้นจะต้องอยู่กับกราวด์ เนื่องจากใช้เขียนคำสั่ง และเขียนข้อมูลเท่านั้น

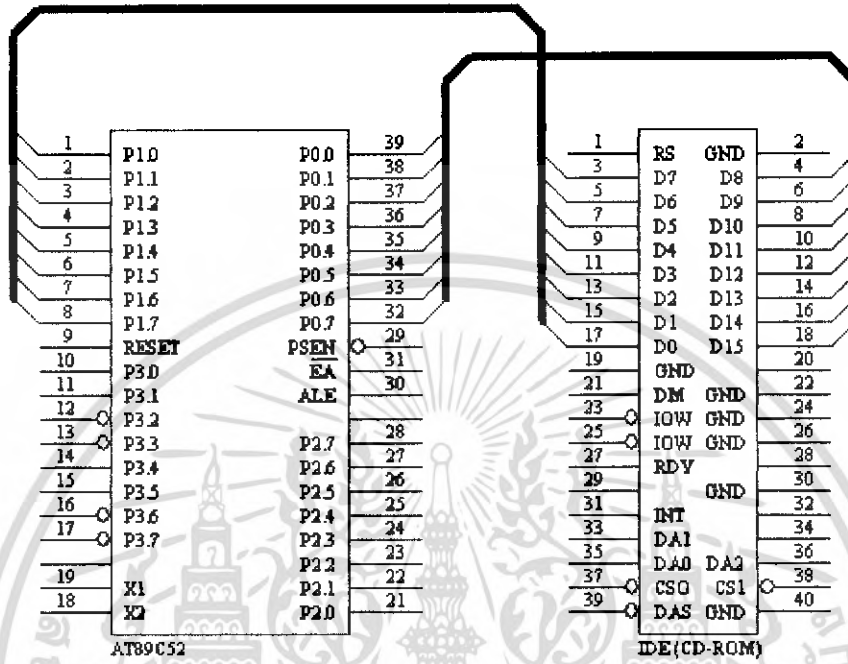


รูป 3.5 ส่วนแสดงผล LCD

3.6 ส่วนการเชื่อมต่อขาที่ใช้งานซีดีรอม

ในการติดต่อกับ ซีดีรอม นั้นจะต้องใช้สัญญาณ 5 สัญญาณในการควบคุมเพื่อเลือกกรีจิสเตอร์ในการติดต่อกัน ก็คือ CS0, CS1, DA0, DA1 และ DA2 ส่วน DIOR และ DIOW จะใช้ในการเขียนและอ่านข้อมูล และในส่วนของกรับและส่งข้อมูลเป็นแบบ 16 บิต (DD0-DD7 และ DD8-DD15) โดยกรีจิสเตอร์ภายในมี 2 บล็อก ซึ่งสามารถทำการเลือกบล็อกได้จาก CS0, CS1 และเลือกกรีจิสเตอร์ภายในแต่ละบล็อกจาก DA0, DA1, DA2 และยังสามารถทำการอ่าน หรือเขียนสัญญาณได้จากกรีจิสเตอร์ DIOR, DIOW ได้ ลักษณะการต่อใช้งานนั้นจะใช้พอร์ต P0.0-P0.7 กับ P1.0-P1.7 เชื่อมต่อกับการรับและส่งข้อมูล 16 บิต ของซีดีรอม ส่วนพอร์ต P2.0-P2.7 นั้นจะใช้เชื่อมต่อกับชุดของสัญญาณควบคุมทั้งหมด ของซีดีรอม โดยทำการต่อเรียงกันตามรูป

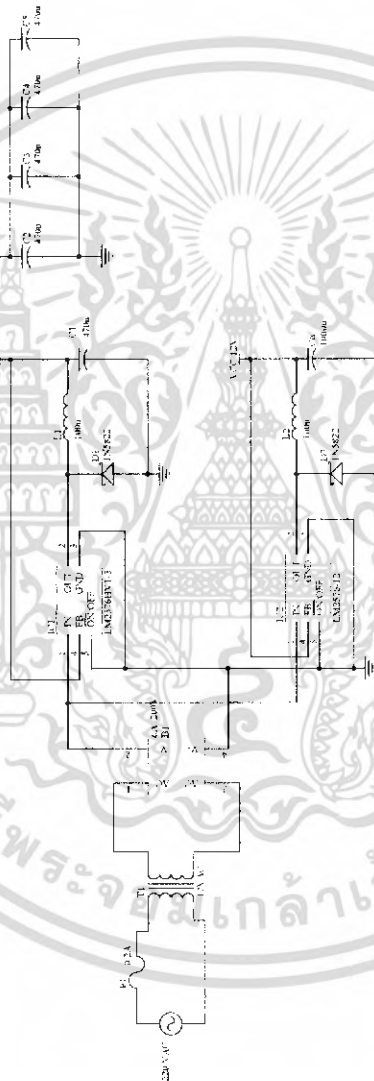
3.6



รูป 3.6 ส่วนการเชื่อมต่อขาที่ใช้งานซีดี

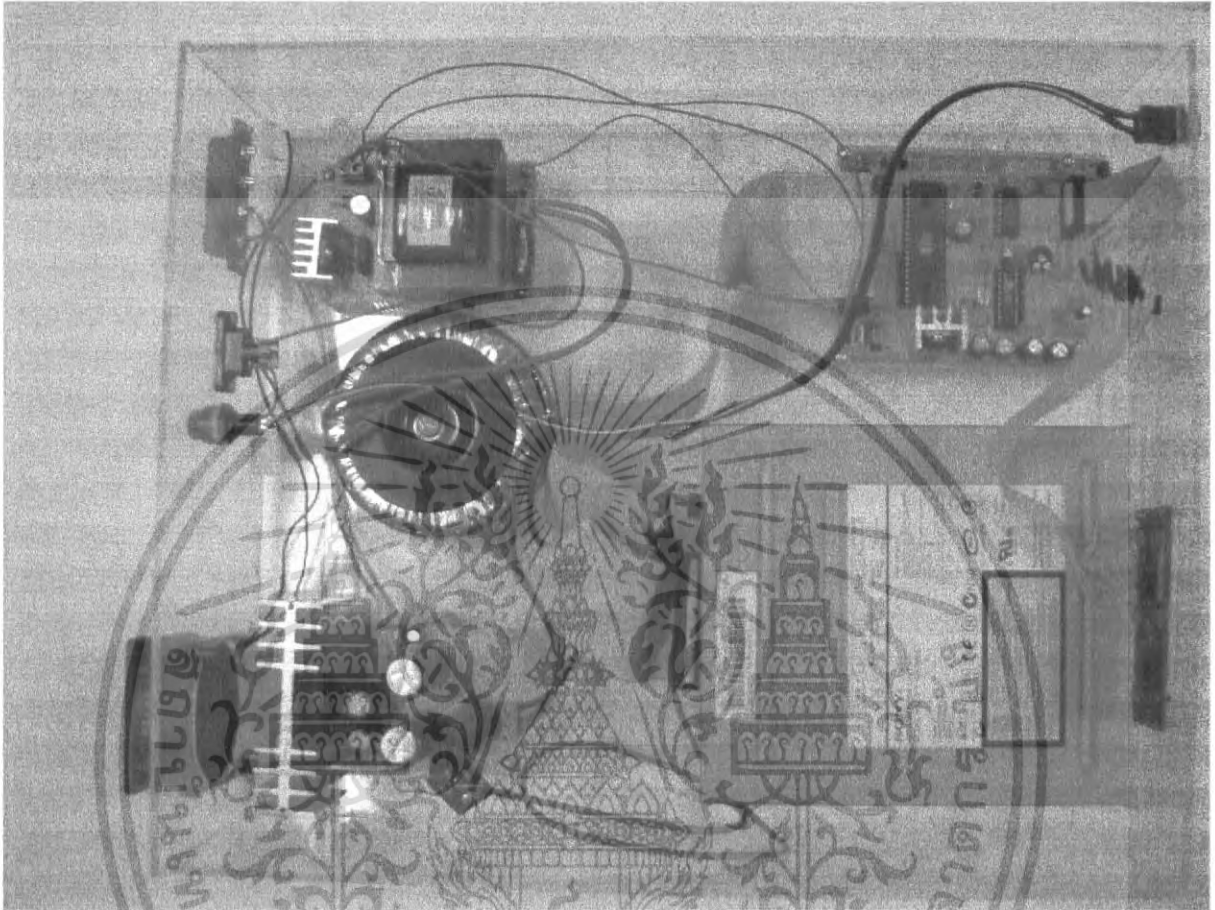
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7 แหล่งจ่ายไฟให้กับซีดีรอมและวงจรถวลุม



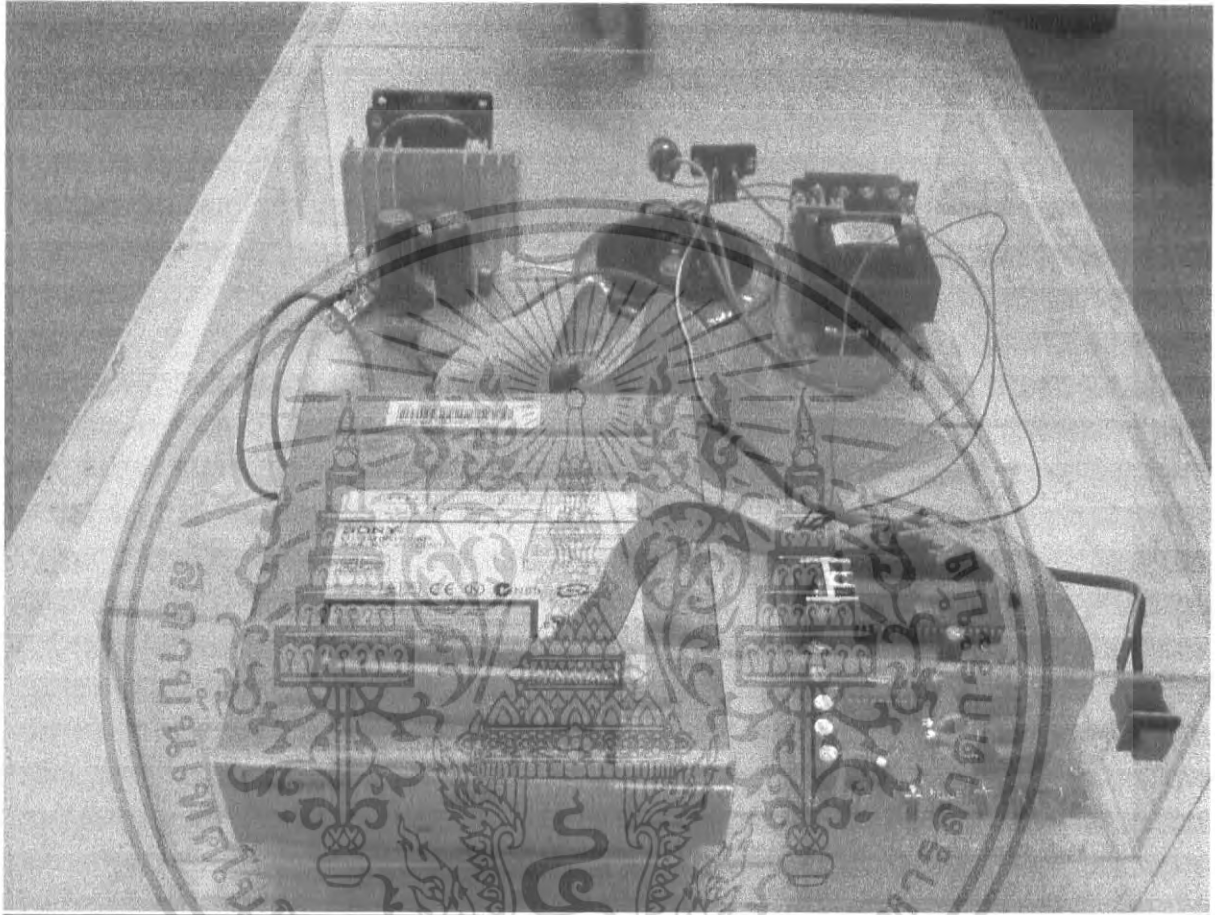
รูป 3.7 แหล่งจ่ายไฟให้กับซีดีรอมและวงจรถวลุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.10 ภาพชิ้นงานด้านบน

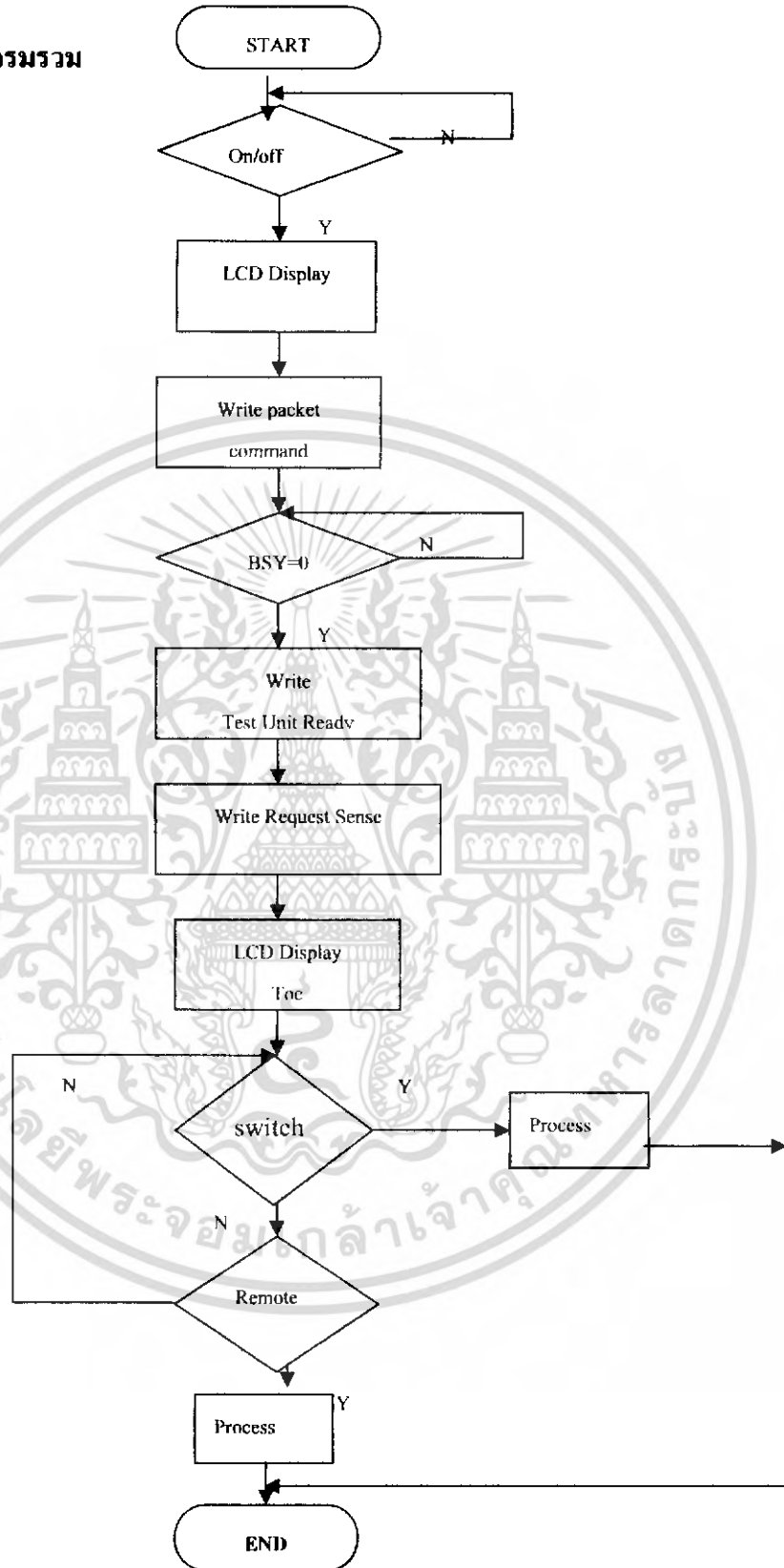
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.11 ภาพชิ้นงานด้านหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

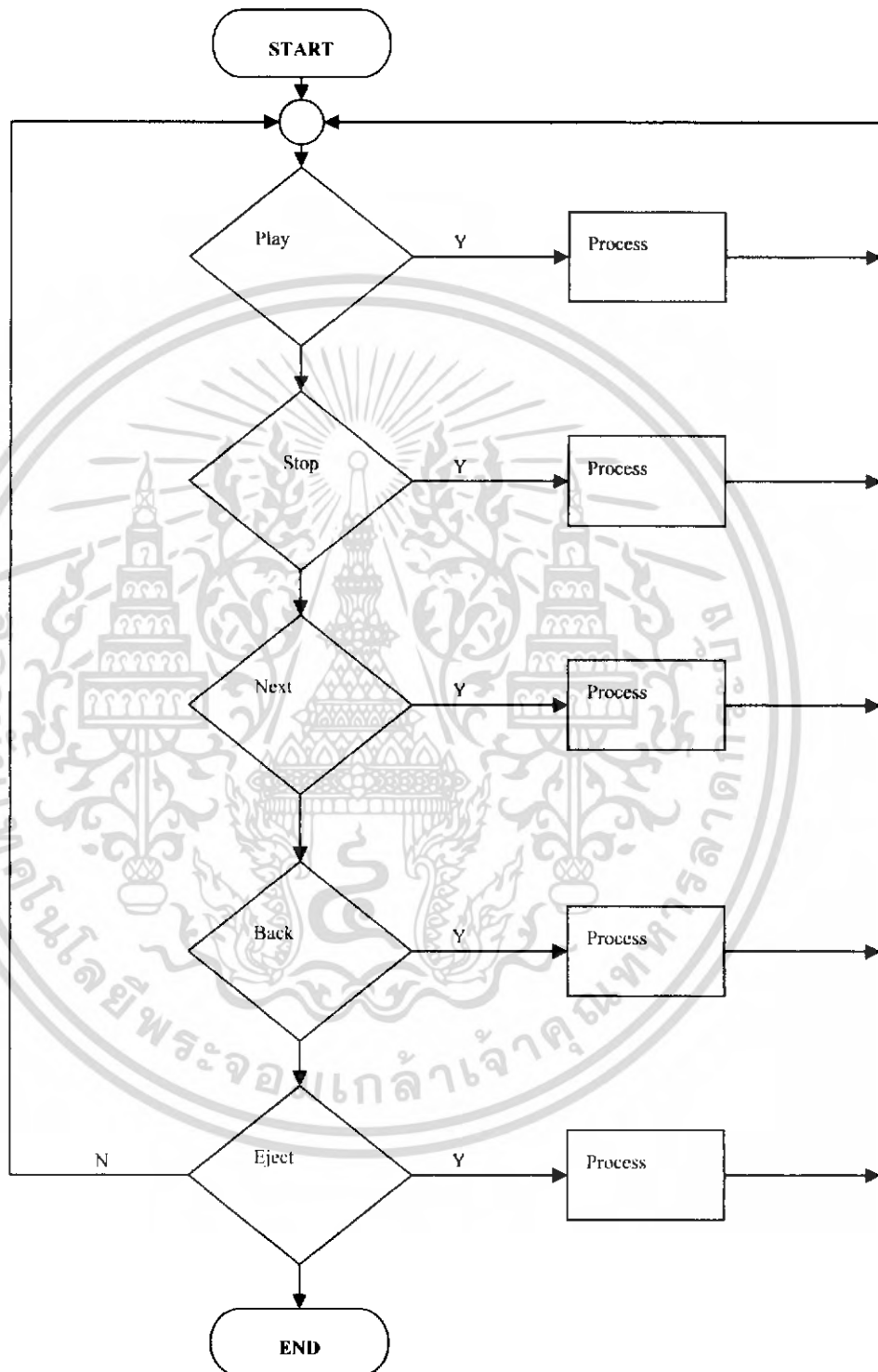
3.10 โฟลว์ชาร์ตโปรแกรมรวม



รูป 3.12 โฟลว์ชาร์ต โปรแกรมรวม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

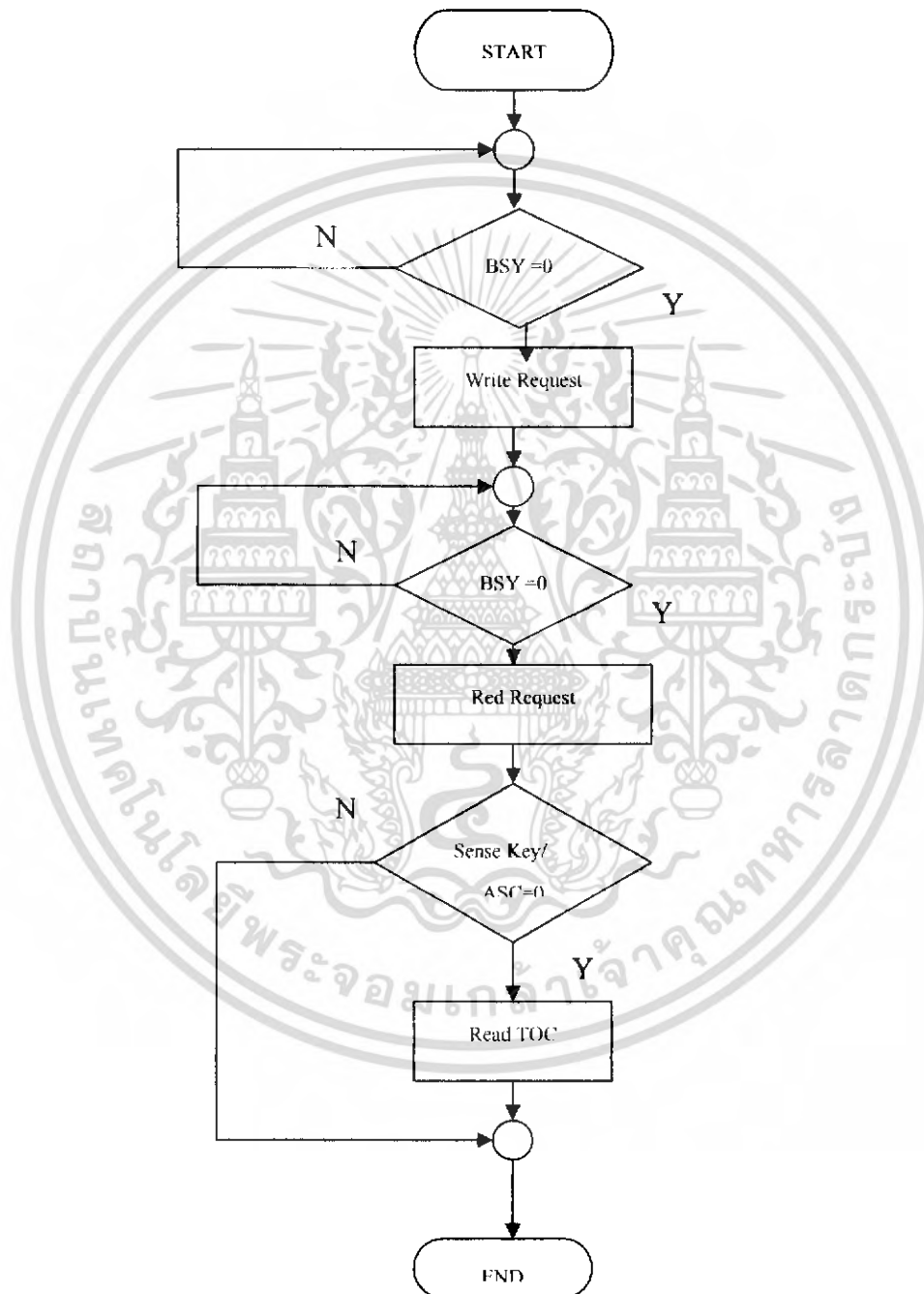
3.11 โฟลว์ชาร์ตย่อยส่วนคีย์สวิตช์



รูป 3.13 โฟลว์ชาร์ตย่อยส่วนคีย์สวิตช์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

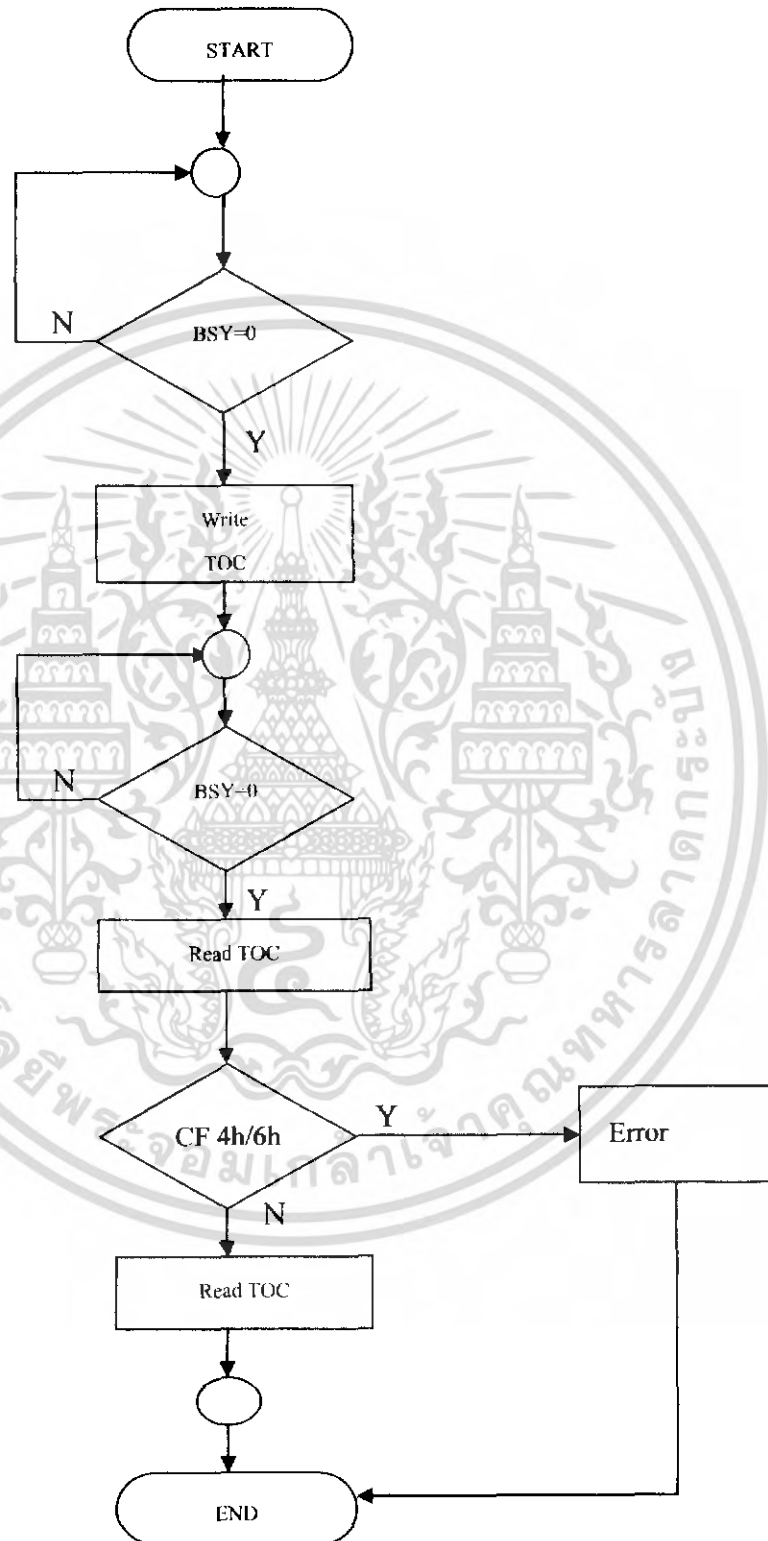
3.12 โพลีชาร์ตย่อยส่วน REQUEST SENE



รูป 3.14 โพลีชาร์ตย่อยส่วน REQUEST SENE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

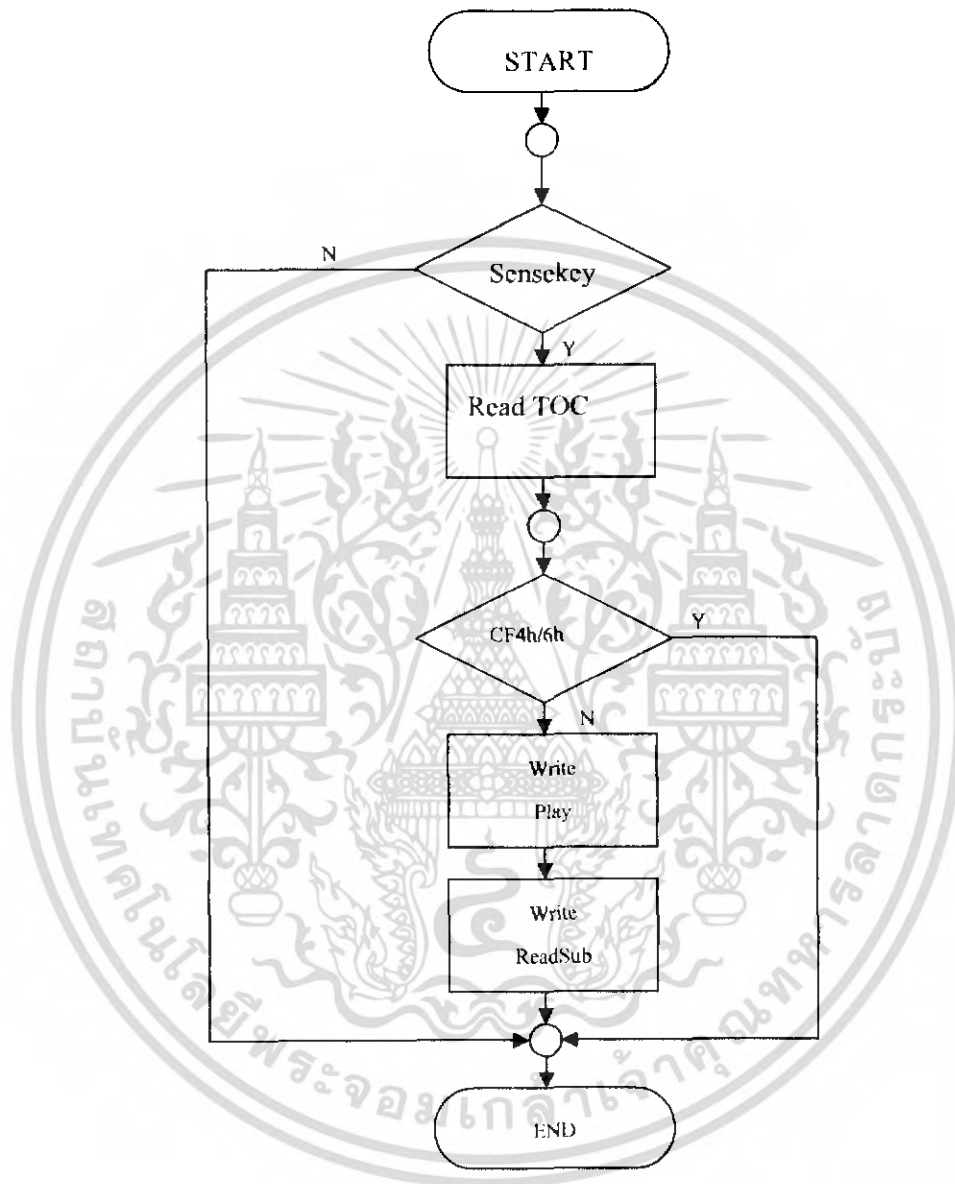
3.13 โฟลว์ชาร์ตย่อยส่วน READ TOC



รูป 3.15 โฟลว์ชาร์ตย่อยส่วน READ TOC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

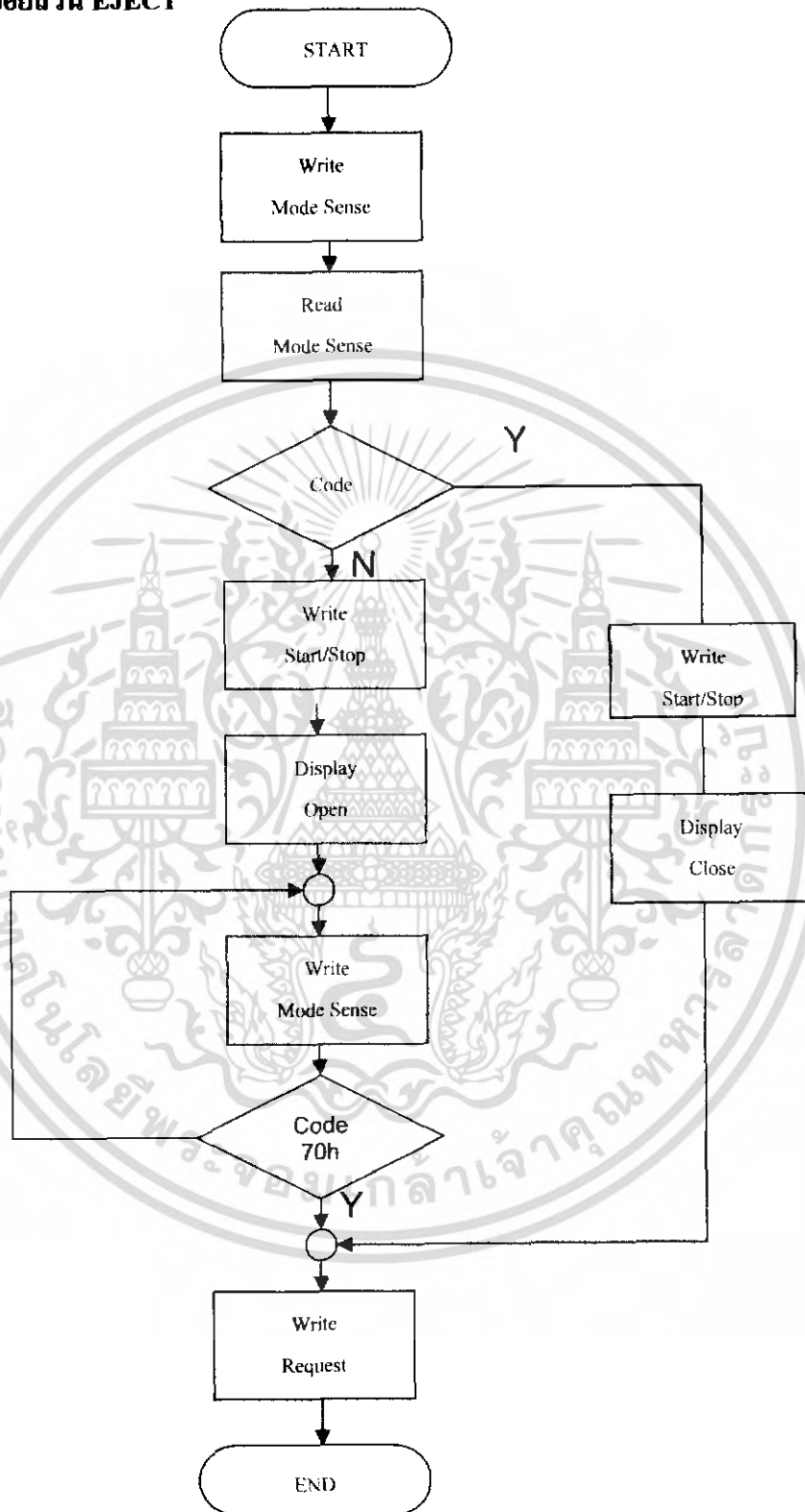
3.14 โฟลว์ชาร์ตย่อยส่วน PLAY



รูป3.16 โฟลว์ชาร์ตย่อยส่วน PLAY

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

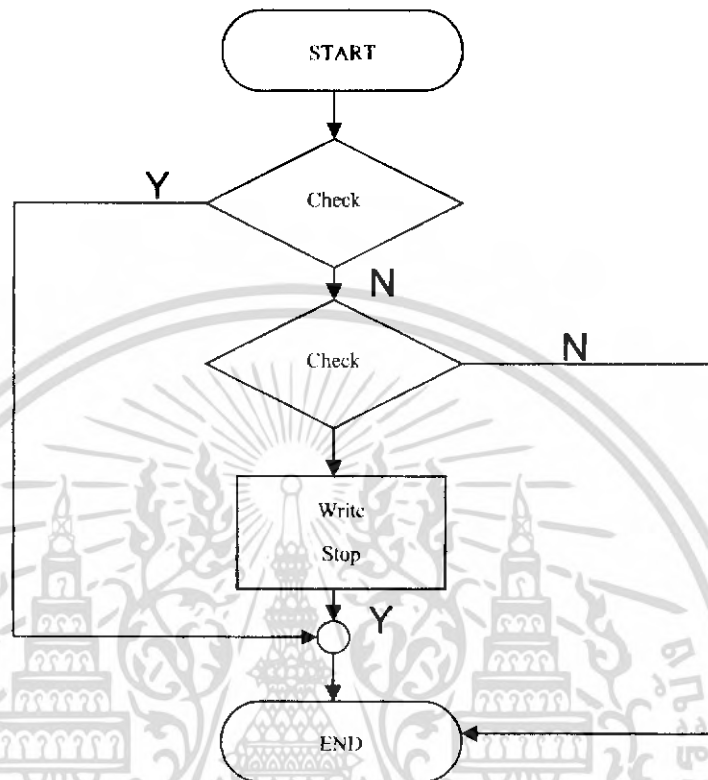
3.15 โฟลว์ชาร์ตย่อยส่วน EJECT



รูป 3.17 โฟลว์ชาร์ตย่อยส่วน EJECT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

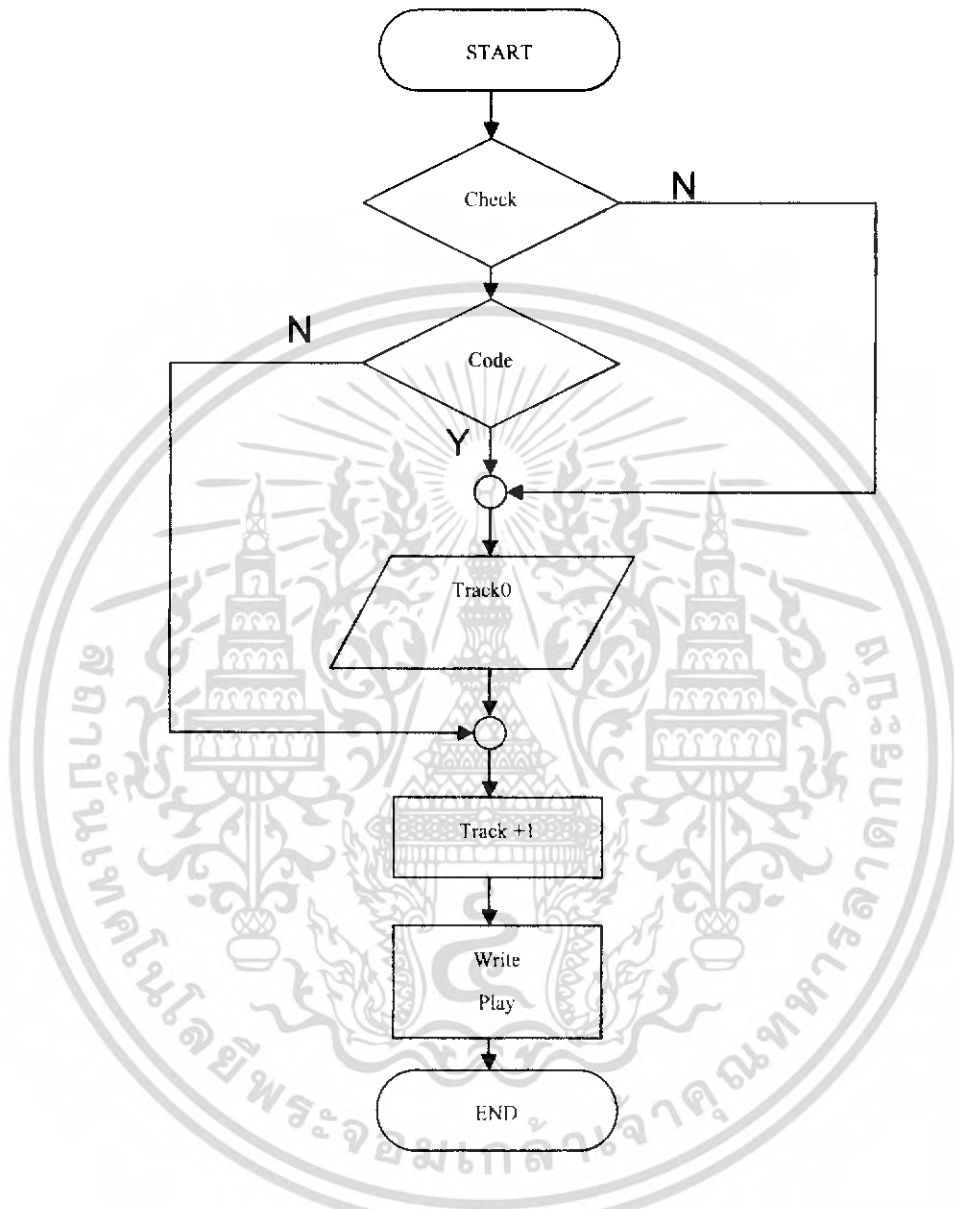
3.16 โฟลว์ชาร์ตย่อยส่วน STOP



รูป 3.18 โฟลว์ชาร์ตย่อยส่วน STOP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

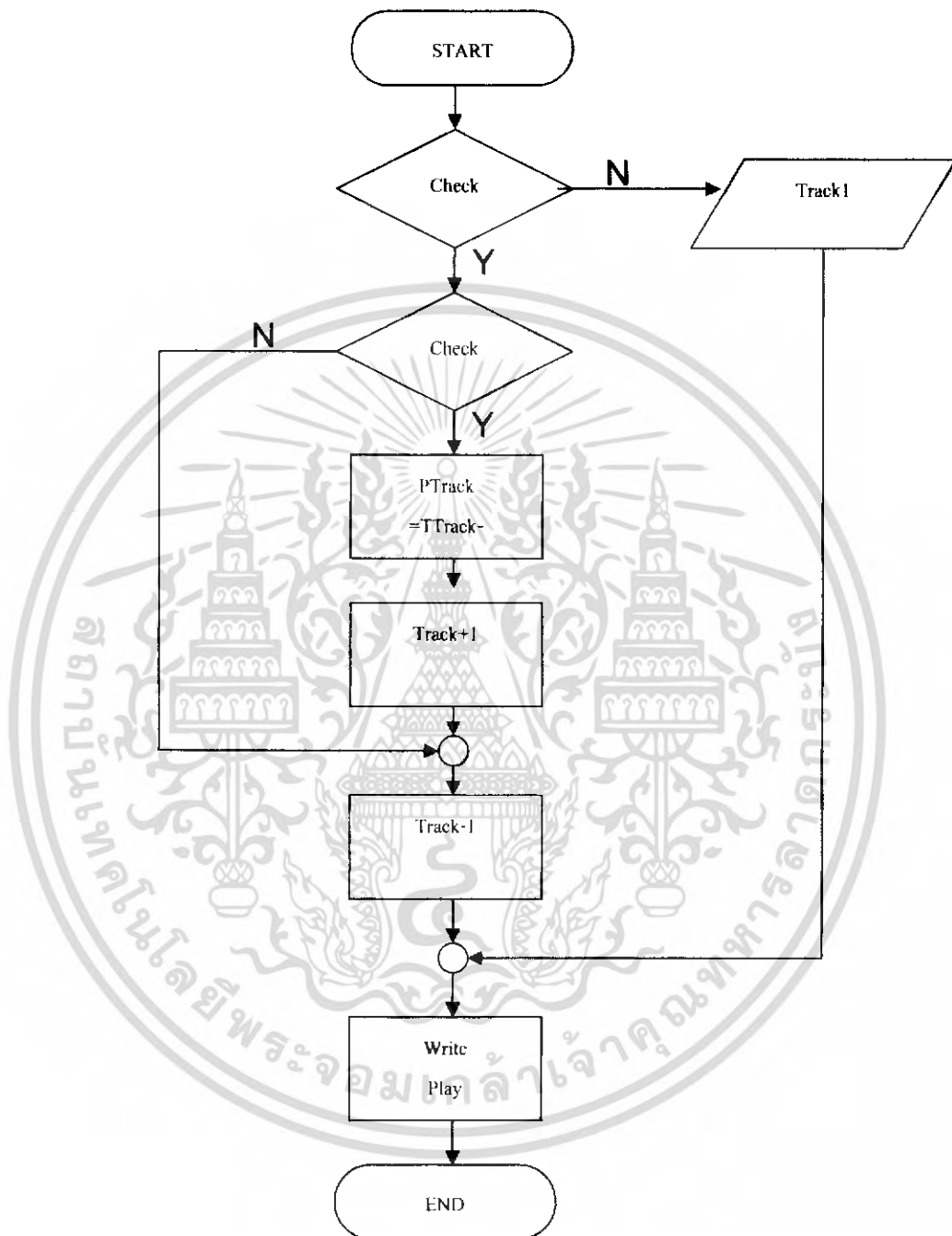
3.17 โพลีชาร์ตย่อยส่วน NEXT



รูป 3.19 โพลีชาร์ตย่อยส่วน NEXT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.18 โฟลว์ชาร์ตย่อยส่วน BACK



รูป 3.20 โฟลว์ชาร์ตย่อยส่วน BACK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.19 โปรแกรมที่ใช้ในการติดต่อกับซีดีรอมไว้ใน AT89C2051

```

;
;Command for control CD-ROM
;
;
CMD_PWR EQU 'u' ;POWER
CMD_EJECT EQU 'e' ;EJECT
CMD_STOP EQU 's' ;STOP
;
CMD_PLAY EQU 'p' ;PLAY
CMD_FF EQU 'f' ;FORWORD TRACK
CMD_REW EQU 'r' ;REWORD TRACK
;
CMD_PAUSE EQU 'x' ;PAUSE
CMD_REPEAT EQU 'g' ;REPEAT DISC
CMD_TRACK EQU 'u' ;TRACK SEEK
;2 BYTE ASCII THEN CANCEL 'c'

CMD_BEFOREOFF EQU 'n' ;before off CD-ROM

CMD_TRACKONCE EQU 'o' ;TRACK 1-9
BYTE1 EQU 30H
BYTE2 EQU 31H
BYTE3 EQU 32H
BYTE4 EQU 33H
CH_REMOTE: BIT 01H

REMOTE_PWR EQU 0A8H ;POWER
REMOTE_EJECT EQU 042H ;EJECT (CD SELECT)
REMOTE_STOP EQU 01CH ;STOP

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;
REMOTE_PLAY EQU 4CH ;PLAY
REMOTE_FF EQU 8CH ;FF
REMOTE_REW EQU 0CH ;REW
;
REMOTE_PAUSE EQU 9CH ;PAUSE
REMOTE_REPEAT EQU 38H ;REPEAT DISC (DISPLAY)

REMOTE_MORE EQU 0B0H ;MORE THAN 10 TRACK

ORG 20H
FLAG: DS 1 ;TEMP FLAG
FLAG_TIMEOUT BIT FLAG.0 ;TIMEOUT
FLAG_REMOTE BIT FLAG.1

FLAG_ERR BIT FLAG.2
FLAG_FIRST_T BIT FLAG.3
FLAG_SECOND BIT FLAG.4

BIT_COMMAND: DS 1 ;CMD FOR REMOTE
BIT_ADDRESS: DS 1 ;ADDRESS FOR SONY
STACK: DS 1

COUNTER EQU 0C0H ;02, 92H = 652 uS
;02, B0H = 775 uS
;02, C0H = 840 uS

MAX EQU 0FFH

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;I/O
```

```
REMOTE BIT P1.0 ;IR = 1.0
```

```
RST BIT P3.2 ;RESET NEXT CPU
```

```
TXD_BIT BIT P3.1 ;TX 110 BOUD RATE
```

```
LED BIT P1.1
```

```
K_POWER BIT P1.2
```

```
K_EJECT BIT P1.3
```

```
K_STOP BIT P1.4
```

```
K_PLAY BIT P1.5
```

```
K_REW BIT P1.6
```

```
K_FFW BIT P1.7
```

```
ORG 0
```

```
LJMP MAIN
```

```
ORG 3H
```

```
RETI
```

```
ORG 0BH
```

```
PUSH ACC
```

```
PUSH PSW
```

```
CLR EX0
```

```
LCALL READ_REMOTE
```

```
POP PSW
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
POP ACC
SETB EX0
RETI
```

```
ORG 23H
RETI
```

```
ORG 50H
REMOTE_TBL: DB 30H ;0
DB 000H, 080H, 040H ;123
DB 0C0H, 020H, 0A0H ;456
DB 060H, 0E0H, 010H ;789
DB 0FFH ;0 command 't'
ORG 80H
MAIN:
MOV SP,#STACK ;STACK POINTER
MOV SCON,#01010010b ;set Mode 1 8 bit Asynchronus
;ren = 1 Enable Serial Port
;TB8,RB8 = 0 stop bit
;TI,RI
MOV TMOD,#00100000B ;set mode (timer)
;mode 2 Auto Reload 8 bit

MOV TH1,#0FDH

SETB TR1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV FLAG,#0 ;
SETB RST ;
NOP
NOP
CLR ES
CLR EA ;DISABLE ALL INTERRUPT
:CLOCK GEN
SETB REMOTE ;SET ACTIVE HIGH REMOTE IN
SETB LED
:START HERE SONY COMMAND IS
MOV FLAG,#0
START:
CLEAR_BIT:
MOV BIT_COMMAND,#0FFH ;CLEAR COMMAND
MOV BIT_ADDRESS.#0FFH ;CLEAR ADDRESS
NOP
NOP
LCALL GET_REMOTE
GET_KEY:
NOP
MOV A,P1
ANL A,#11111100B
XRL A,#11111100B
JZ START
;-----
CHECK_KEY:

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
JB FLAG_FIRST_T,K_EXIT
```

```
SETB K_POWER
```

```
NOP
```

```
JB K_POWER,K2
```

```
MOV A,#CMD_PWR
```

```
JB RST.EXIT_K1
```

```
LCALL ONOFF
```

```
EXIT_K1:
```

```
CPL RST ;RESET POWER ON / OFF
```

```
CPL LED
```

```
JNB K_POWER,$
```

```
AJMP K7
```

```
K2:
```

```
SETB K_EJECT
```

```
NOP
```

```
JB K_EJECT,K3
```

```
MOV A,#CMD_EJECT
```

```
LCALL TX
```

```
JNB K_EJECT,$
```

```
AJMP K7
```

```
K3:
```

```
SETB K_PLAY
```

```
NOP
```

```
JB K_PLAY,K4
```

```
MOV A,#CMD_PLAY
```

```
LCALL TX
```

```
JNB K_PLAY,$
```

```
AJMP K7
```

```
K4:
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

SETB K_STOP
NOP
JB K_STOP,K5
MOV A,#CMD_STOP
LCALL TX
JNB K_STOP,$
AJMP K7

```

K5:

```

SETB K_REW
NOP
JB K_REW,K6
MOV A,#CMD_REW
LCALL TX
JNB K_REW,$
AJMP K7

```

K6:

```

SETB K_FFW
NOP
JB K_FFW,K7
MOV A,#CMD_FF
LCALL TX
JNB K_FFW,$

```

K7:

K_EXIT:

```

LCALL UTIL_PRESS
LJMP START

```

```

;-----
GET_REMOTE:

```

```

    JNB CH_REMOTE,CHECK_KEY

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CLR CH_REMOTE
```

```
LCALL CH_INPUT
```

```
RET
```

```
-----  
CH_INPUT:  PUSH  04
```

```
MOV  R4,BYTE1
```

```
CJNE R4,#7BH,NOT_EJ
```

```
MOV  R4,BYTE2
```

```
CJNE R4,#0CAH,NOT_EJ
```

```
MOV  R4,BYTE3
```

```
CJNE R4,#0FEH,PLAY
```

```
MOV  R4,BYTE4
```

```
CJNE R4,#01H,NOT_EJ
```

```
MOV  A,#CMD_EJECT
```

```
LJMP NOT_EJ
```

```
;/;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;/
```

```
PLAY:      MOV  R4,BYTE3
```

```
CJNE R4,#0FDH,STOP
```

```
MOV  R4,BYTE4
```

```
CJNE R4,#02H,NOT_EJ
```

```
MOV  A,#CMD_EJECT    ;EJECT
```

```
LJMP EXIT_GET_CMD
```

```
;/;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;/
```

```
STOP:      MOV  R4,BYTE3
```

```
CJNE R4,#0FCH,BACK
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV R4,BYTE4
CJNE R4,#03H,NOT_EJ

MOV A,#CMD_STOP ;STOP
LJMP EXIT_GET_CMD

;////////////////////////////////////

BACK: MOV R4,BYTE3
CJNE R4,#0FAH,NEXT

MOV R4,BYTE4
CJNE R4,#05H,NOT_EJ

MOV A,#CMD_PLAY ;PLAY
LJMP EXIT_GET_CMD

;////////////////////////////////////

NEXT: MOV R4,BYTE3
CJNE R4,#0F8H,REM_A

MOV R4,BYTE4
CJNE R4,#07H,NOT_EJ

MOV A,#CMD_FF ;FF
LJMP EXIT_GET_CMD

;////////////////////////////////////

REM_A:
LCALL FIND_TRACK
JB FLAG_ERR,NOT_SEND
JB FLAG_FIRST_T,SEND_2BYTE

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV B,A
XRL A,#'0' ;TRACK 0 NOT HAVE
JZ EXIT_D ;IF PRESS "0" THEN NOT ACTIVE
PUSH ACC
MOV A,#'t'
LCALL SEND_DELAY
SETB FLAG_SECOND
MOV A,#'0'
LCALL SEND_DELAY
POP ACC
MOV A,B
LCALL SEND_DELAY
LJMP SUCCESS_2BYTE
SEND_2BYTE:
LCALL FIND_TRACK
JNB FLAG_ERR,REM_B
MOV A,#'c'
LCALL SEND_DELAY
; MOV A,#'c'
; LCALL TX
LJMP EXIT_D
REM_B:
LCALL TX
JB FLAG_SECOND,SUCCESS_2BYTE
SETB FLAG_SECOND ;
LJMP EXIT_D
SUCCESS_2BYTE:
CLR FLAG_FIRST_T
CLR FLAG_SECOND
NOT_SEND:

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
AJMP EXIT_D
```

```
EXIT_GET_CMD:
```

```
LCALL TX
```

```
EXIT_D:
```

```
MOV R2,#3
```

```
LCALL DTSEC
```

```
RET
```

```
-----
```

```
FIND_TRACK:
```

```
MOV B,#0
```

```
CLR FLAG_ERR ;CLEAR FLAG ERROR
```

```
MOV DPTR,#REMOTE_TBL
```

```
FIND_T_NEXT:
```

```
CLR A
```

```
MOVC A,@A+DPTR
```

```
XRL A,BIT_COMMAND
```

```
JZ FIND_T_EXIT
```

```
INC B
```

```
INC DPTR
```

```
MOV A,B
```

```
XRL A,#12 ;11 KEY COMPARE
```

```
JNZ FIND_T_NEXT
```

```
SETB FLAG_ERR ;SET FLAG ERROR
```

```
RET
```

```
FIND_T_EXIT: MOV A,B
```

```
LCALL HTOA
```

```
MOV A,R3 ;SEND LSB
```

```
RET
```

```
-----
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ONOFF:

```

MOV R2,#10
LCALL DTSEC
MOV A,#n'
LCALL TX
MOV R2,#30
LCALL DTSEC
NOP
NOP
RET

```

SEND_DELAY:

```

LCALL TX
MOV R2,#4
LCALL DTSEC
RET

```

BYTE_ADDRESS EQU 04H ;ANL D4H,44H

BYTE_ERROR EQU 0DH ;ERROR

GET_IR:

```

CLR FLAG_REMOTE
MOV R5,#3 ;SONY "L" 2.4 mS

```

BUSY:

```

LCALL BUSY SIGNAL
DJNZ R5,BUSY
JNB REMOTE,$ ;WAIT SIGNAL ACTIVE HIGH

```

RECIVE:

;THIS STEP READ COMMAND

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV R0,#BIT_COMMAND
MOV B,#8 ;8 BIT COMMAND
MOV R7,#0

```

D_READ:

```

DEC R7
MOV A,R7
XRL A,#0
JZ EXIT_IR
JB REMOTE.D_READ
LCALL FIND_BIT

```

```
DJNZ B,D_READ
```

;NEXT STEP READ ADDRESS

```

MOV R0,#BIT_ADDRESS ;4 BIT ADDRESS
MOV B,#4
MOV R7,#0

```

A_READ:

```

DEC R7
MOV A,R7
XRL A,#0
JZ EXIT_IR

JB REMOTE,A_READ

```

;AFTER GET FILE FROM REMOTE READY

:

```

LCALL FIND_BIT
DJNZ B,A_READ

```

;NONE COMPARE BIT ADDRESS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;
;   LCALL CH_CMD

MOV   A,BIT_ADDRESS
ANL   A,#0FH   ;MARK 4 BIT (LSB)
XRL   A,#BYTE_ADDRESS
JNZ   NONE_SONY
SETB  FLAG_REMOTE

NONE_SONY:
EXIT_IR:
RET

;-----
UTIL_PRESS:
MOV   R7,#MAX
WAITMAX:
MOV   R6,#MAX
DJNZ  R6,$
DJNZ  R7,WAITMAX
RET

;-----
CH_CMD:
;MAY BE IF ADD FUNCTION TO LCD DISPLAY
MOV   A,BIT_COMMAND
LCALL SEND_TEST
MOV   A,BIT_ADDRESS
LCALL SEND_TEST
MOV   A,#0AH
LCALL TX
MOV   A,#0DH
LCALL TX

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        LCALL UTIL_PRESS
        RET
;-----
; 1 CYCLE TIME IS 650 uS
; DELAY TIME 650 uS DATA COUNTER IS 02H, 92H
FIND_BIT:
        CLR    C
        LCALL BUSY_SIGNAL
        JB     REMOTE,ABORT
        SETB   C
ABORT:
        MOV    A,@R0
        RLC    A
        MOV    @R0,A    ;STORE
        JNB   REMOTE,$    ;WAIT FOR ACTIVE HIGH AGAIN
        RET
;-----
; WAIT TIME IS MORE > 650 uS
BUSY_SIGNAL:
        MOV    R6,#02H
DELAY:   MOV    R7,#COUNTER    ;DATA FOR WAIT 1 PULSE CYCLE
        DJNZ  R7,$
        DJNZ  R6,DELAY
        RET

LF:     MOV    A,#0AH
        LCALL TX
        MOV    A,#0DI1

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
LCALL TX
```

```
RET
```

```
;-----  
RX:
```

```
JNB RI,$
```

```
CLR RI
```

```
MOV A,SBUF
```

```
RET
```

```
;-----  
TX:
```

```
JNB TI,$
```

```
CLR TI
```

```
MOV SBUF,A
```

```
RET
```

```
;-----  
HTOA:
```

```
PUSH ACC
```

```
SWAP A
```

```
ACALL HTOAS
```

```
MOV R2,A
```

```
POP ACC
```

```
ACALL HTOAS
```

```
MOV R3,A
```

```
RET
```

```
HTOAS: ANL A,#0FH
```

```
CJNE A,#0AH,$+3
```

```
JNC HTOAS1
```

```
ORL A,#30H
```

```
RET
```

```
HTOAS1: SUBB A,#9
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ORL A,#40H
```

```
RET
```

```
-----
```

```
SEND_TEST:
```

```
LCALL HTOA
```

```
MOV A,R2
```

```
LCALL TX
```

```
MOV A,R3
```

```
LCALL TX
```

```
MOV A,#'
```

```
LCALL TX
```

```
RET
```

```
-----
```

```
DTSEC: MOV R6,#179
```

```
DTSEC1: MOV R7,#0
```

```
DJNZ R7,$
```

```
DJNZ R6,DTSEC1
```

```
DJNZ R2,DTSEC
```

```
RET
```

```
:/://////
```

```
NOT_EJ: POP 04
```

```
RET
```

```
:/://////
```

```
ERROR: LCALL WAIT_1mS
```

```
RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

READ_REMOTE: JNB  REMOTE,$

                LCALL WAIT_2_5mS

                JNB  REMOTE,ERROR

                JB   REMOTE,$

                NOP

                NOP

                MOV  R1,#08

                JNB  REMOTE,$

ST_WAIT:       LCALL WAIT_750uS

                MOV  C,REMOTE

                CPL  CY

                RLC  A

                DJNZ R1,NOT_8

                LJMP STO_BYTE1

NOT_8:         JB   REMOTE,$

                NOP

                NOP

                JNB  REMOTE,$

                LJMP ST_WAIT

STO_BYTE1:    MOV  BYTE1,A

;////////////////////////////////////

                JB   REMOTE,$

                NOP

                NOP

                MOV  R1,#08

                JNB  REMOTE,$

ST_WAIT2:     LCALL WAIT_750uS

                MOV  C,REMOTE

                CPL  CY

                RLC  A

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

                DJNZ R1,NOT_82
                LJMP STO_BYTE2
NOT_82:         JB  REMOTE,$
                NOP
                NOP
                JNB REMOTE,$
                LJMP ST_WAIT2
STO_BYTE2:     MOV  BYTE2,A
                ;////////////////////////////////////
                JB  REMOTE,$
                NOP
                NOP
                MOV  R1,#08
                JNB REMOTE,$
ST_WAIT3:      LCALL WAIT_750uS
                MOV  C,REMOTE
                CPL  CY
                RLC  A
                DJNZ R1,NOT_83
                LJMP STO_BYTE3
NOT_83:         JB  REMOTE,$
                NOP
                NOP
                JNB REMOTE,$
                LJMP ST_WAIT3
STO_BYTE3:     MOV  BYTE3,A
                ;////////////////////////////////////
                JB  REMOTE,$
                NOP
                NOP

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV R1,#08
JNB REMOTE,$
ST_WAIT4: LCALL WAIT_750uS
MOV C,REMOTE
CPL CY
RLC A
DJNZ R1,NOT_84
LJMP STO_BYTE4
NOT_84: JB REMOTE,$
NOP
NOP
JNB REMOTE,$
LJMP ST_WAIT4
STO_BYTE4: MOV BYTE4,A
;////////////////////////////////////
SETB CH_REMOTE
LCALL WAIT_60mS
LCALL WAIT_60mS
RET
;////////////////////////////////////
WAIT_250uS: MOV R7,#1
AJMP DO_WAIT
WAIT_500uS: MOV R7,#2
SJMPDO_WAIT
WAIT_750uS: MOV R7,#3
SJMPDO_WAIT
WAIT_1mS: MOV R7,#4
SJMP DO_WAIT

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

WAIT_2mS:      MOV      R7,#8
                SJMP    DO_WAIT

WAIT_2_5mS:    MOV      R7,#10
                SJMP    DO_WAIT

WAIT_60mS:     MOV      R7,#240
                SJMP    DO_WAIT

```

```

DO_WAIT:       SETB    TR0

WAIT_LP:       JNB     TF0,$
                CLR     TF0
                DJNZ   R7,WAIT_LP
                CLR     TR0
                MOV    TH0,#6
                MOV    TL0,#6
                RET

```

```

END

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.20 โปรแกรมที่ใช้ในการติดต่อกลับซีดีรอมใช้ใน AT89S8252

```
;Project CD-ROM interface ATAPI
```

```
;By Arampee
```

```
;RD2
```

```
Bau_RD2 equ 0fah
```

```
;C52
```

```
Bau_C52 equ 0fdh
```

```
Start_CD equ 1
```

```
org 20h
```

```
flag: ds 1
```

```
flag_cmd bit flag.0 ;for lcd command
```

```
flag_serial bit flag.1
```

```
flag_disc bit flag.2 ;disc insert
```

```
flag_eject bit flag.3
```

```
flag_play bit flag.4
```

```
flag_subb bit flag.5
```

```
flag_pause bit flag.6
```

```
flag_error bit flag.7
```

```
org 21h
```

```
flag_II: ds 1
```

```
flag_repeat bit flag_II.0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
disbuff:    ds    17        ;17 byte buffer
```

```
;for function hex to decimal
```

```
mcm_buff:   ds    3
```

```
xbyte_d:    ds    1
```

```
xbyte_l:    ds    1
```

```
xbyte_h:    ds    1
```

```
audio_status: ds    1
```

```
packet_size: ds    1
```

```
start_track: ds    1
```

```
total_track: ds    1
```

```
track_play:  ds    1
```

```
index_play:  ds    1
```

```
min:         ds    1
```

```
sec:         ds    1
```

```
tmp:         ds    1
```

```
tmp0:        ds    1
```

```
tmp1:        ds    1
```

```
tmp2:        ds    1
```

```
tmp3:        ds    1
```

```
Start_M:     ds    1
```

```
Start_S:     ds    1
```

```
Start_F:     ds    1
```

```
End_M:       ds    1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End_S:    ds    1
```

```
End_F:    ds    1
```

```
packet:
```

```
packet_cmd: ds    1
```

```
packet_1:  ds    1
```

```
packet_2:  ds    1
```

```
packet_3:  ds    1
```

```
packet_4:  ds    1
```

```
packet_5:  ds    1
```

```
packet_6:  ds    1
```

```
packet_7:  ds    1
```

```
packet_8:  ds    1
```

```
packet_9:  ds    1
```

```
packet_10: ds    1
```

```
packet_11: ds    1
```

```
packet_12: ds    1
```

```
packet_13: ds    1
```

```
packet_14: ds    1
```

```
packet_15: ds    1
```

```
org 70h
```

```
TOC_file: ds    30*3 ;maximum 25 track (3 byte / track)
```

```
stack equ 255-30
```

```
org 0
```

```
ljmp main
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

retl

org 1bh      ;
retl

org 23h      ;service Serial Interrupt
ljmp  rx_int

main:
mov  sp,#stack
mov  psw,#0

mov  scon,#01010000B ;set Mode 1 8 bit Asynchronus
                    ;ren = 1 Enable Serial Port
                    ;TB8,RB8 = 0 stop bit
                    ;TI,RI
mov  tmod,#00100000B ;set mode (timer)
                    ;mode 2 Auto Reload 8 bit)
mov  tcon,#01010000B ;timer0,1 internal count

mov  th1,#Bau_C52
mov  tl1,#Bau_C52

setb  tr1

clr  ea      ;enable all interrupt

mov  flag,#0

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

mov    flag_II,#0        ;clear bit flag

lcall  init_lcd

mov    dptr,#scl_init

lcall  write_line

mov    dptr,#scl_wait

lcall  write_line

mov    r2,#5

lcall  dtsec

;

lcall  init_ata_device    ;initilize CD-ROM

lcall  atapi_identify_device ;check Company CD-ROM

lcall  wait_for_ready    ;check media

mov    dptr,#scl_cmd

lcall  write_line

mov    track_play,#1

setb  es

setb  ea

;-----
wait_serial:

    jb  flag_disc,none_Chk

    lcall wait_for_ready

;    ajmp wait_serial

none_Chk:

    jnb flag_serial.menu_exit

    mov  a,tmp

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        clr    flag_serial

key_1:
        cjne  a,'#c',key_2    ;Eject / Closed Tray
        clr    flag_repeat    ;clear flag repeat
        lcall atapi_eject
        ljmp  menu_exit

key_2:
        cjne  a,'#r',key_3    ;reverse
        jnb   flag_disc,menu_exit
        lcall atapi_rew_track
        ljmp  menu_exit

key_3:
        cjne  a,'#p',key_4
        jnb   flag_disc,menu_exit
        jnb   flag_play,menu_exit
        mov   track_play,#Start_CD ;play track
        lcall play_CD
        ljmp  menu_exit

key_4:
        cjne  a,'#f',key_5
        jnb   flag_disc,menu_exit
        lcall atapi_few_track ;forword track
        ljmp  menu_exit

key_5:
        cjne  a,'#s',key_6
        jnb   flag_disc,menu_exit
        clr    flag_repeat    ;clear flag repeat
        lcall atapi_stop     ;stop

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        clr    flag_play
        ljmp  menu_exit
key_6:
:may be Remote Control
        cjne  a,#'t',key_7      ;track 2 byte
        jnb  flag_disc,menu_exit
        clr  es
        lcall get_remote
        setb es
        ljmp  menu_exit
key_7:
        cjne  a,#'x',key_8
        jnb  flag_disc,menu_exit ;pause
        lcall atapi_pause
        ljmp  menu_exit
key_8:
        cjne  a,#'g',key_9      ;repeat track
        cpl  flag_repeat
        mov  r2,#10
        lcall dtsec
        ljmp  menu_exit
key_9:
        cjne  a,#'n',key_10     ;on /off
        lcall clear_screen
        ljmp  menu_exit
key_10:
menu_exit:
        jnb  flag_play,none_Display_Time
        jnb  flag_pause,none_Display_Time
        clr  es

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

lcall read_subchanCmd    ;display time
mov    a,audio_status
cjnc  a,#13h,none_stop

jnb   flag_repeat,repeat_disc
lcall atapi_stop
ljmp  none_stop

repeat_disc:
mov   track_play,#Start_CD
lcall Play_CD

none_stop:
mov   r2,#3
lcall dtsec
setb  es

none_Display_Time:
mov   tmp,#0
mov   a,SBUF
clr   ri
ljmp  wait_serial

;-----
dtsec:   mov   r6,#179
dtsec1:  mov   r7,#0h
         djnz  r7,$
         djnz  r6,dtsec1
         djnz  r2,dtsec
         ret

;-----

get_remote:
;first byte

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

mov  dptr,#scl_shiftTrack
lcall write_line
mov  a,#LINE2+12
lcall goto_xy

lcall rx_byte
cjne a,'#c',digit_one
ljmp Shift_exit

digit_one:
lcall write_lcd
clr  c
subb a,#30h
anl  a,#0fh
swap a
anl  a,#0f0h
mov  xbyte_h,a
;!'s
;second byte
lcall rx_byte
cjne a,'#c',next_digit
ljmp Shift_Exit

next_digit:
lcall write_lcd
clr  c
subb a,#30h
anl  a,#0fh
orl  a,xbyte_h

once_track:
mov  xbyte_d,a

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;success 2 byte
    mov  xbyte_h,#0
    mov  xbyte_l,#0

    lcall dtoh

    mov  r2,#2
    lcall dtsec

    mov  a,total_track ;
    mov  b,xbyte_l ;Reload
    clr  c
    subb a,b
    jc   Over_Track
    mov  a,b ;reload
    xrl  a,#0
    jz   Over_track
    mov  a,b
    mov  track_play,a ;play track
    lcall play_CD

Over_Track:
Shift_Exit:
    ret

;-----
power_down:
    clr  ATA_nRESET
    lcall Atapi_Delaytime
    setb ATA_nRESET
    mov  dptr,#scl_blank1
    lcall write_line

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

mov  dptr,#scl_blank2
lcall write_line
ret

;-----
clear_screen:
jnb  flag_disc,key9_exit
clr  flag_repeat      ;clear flag repeat
lcall atapi_stop     ;stop
clr  flag_play
mov  dptr,#scl_blank1
lcall write_line
mov  dptr,#scl_poweroff
lcall write_line
key9_exit:
ret

;-----
;0123456701234567
scl_cmd:  db  LINE1, "Press Command ? "
scl_init:  db  LINE1, " Initialize "
scl_wait:  db  LINE2, " Please Wait... "
scl_test:  db  LINE2, "***** done *****"
scl_readTOC: db  LINE1, " TOC Reading... "
scl_blank1: db  LINE1, " "
scl_blank2: db  LINE2, " "
scl_poweroff: db  LINE2, "---- OFF. ----"
scl_shiftTrack: db  LINE2, "Shift Track "

;-----
LINE1 equ 0
LINE2 equ 40h

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
point    equ    16    ;16 char
```

```
lcd_clk   bit    P2.0
```

```
lcd_data  bit    P2.1
```

```
lcd_e     bit    P2.2
```

```
-----
```

```
delay:
```

```
    mov    r4,#0h
```

```
    djnz  r4,$
```

```
    djnz  r2,delay
```

```
    ret
```

```
-----
```

```
print_lcd:
```

```
    lcall htoa
```

```
    mov   a,r2
```

```
    lcall write_lcd
```

```
    mov   a,r3
```

```
    lcall write_lcd
```

```
    ret
```

```
-----
```

```
write_led:
```

```
    setb  flag_cmd
```

```
    lcall write_byte
```

```
    ret
```

```
-----
```

```
write_byte:
```

```
    push b
```

```
    push acc
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
mov b,#8 ;shift 8 bit
```

```
send_lcd:
```

```
rlc a
```

```
mov lcd_data,c
```

```
nop
```

```
clr lcd_clk
```

```
nop
```

```
setb lcd_clk
```

```
nop
```

```
djnz b,send_lcd
```

```
clr lcd_clk
```

```
nop
```

```
setb lcd_clk
```

```
nop
```

```
jb flag_cmd,RS_HIGH
```

```
clr lcd_data
```

```
ajmp RS_exit
```

```
RS_HIGH:
```

```
setb lcd_data
```

```
RS_exit:
```

```
nop
```

```
clr lcd_e
```

```
mov r2,#1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

lcall delay

setb lcd_c
mov r2,#1
lcall delay
pop acc
pop b
ret

;-----
write_disbuf:
mov a,#LINE2
lcall goto_xy
;
setb flag_cmd
mov r0,#disbuff
mov b,#point
write_Char:
push b
mov a,@r0
lcall write_byte
inc r0
pop b
djnz b,write_Char
ret

;-----
write_line:
clr a
movc a,@a+dptr
lcall goto_xy

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

inc  dptr

setb  flag_cmd
mov  b,#point
se_lcd:
push  b
clr  a
movc  a,@a+dptr
lcall write_byte
inc  dptr
pop  b
djnz b,se_lcd
ret
;-----
goto_xy:
clr  flag_cmd
setb acc.7 ;sct bit 7 are goto_xy
lcall write_byte
ret
;-----
init_lcd:
setb lcd_e
setb lcd_clk
;delay time for initialize

mov  r2,#5
lcall delay

clr  flag_cmd ;if flag_cmd == "1" then RS = LOW;
mov  a,#38h ;cmd
lcall write_byte

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
mov a,#0fh ;cmd
```

```
lcall write_byte
```

```
mov a,#06h ;cmd
```

```
lcall write_byte
```

```
mov a,#01h ;cmd
```

```
lcall write_byte
```

```
ret
```

```
-----
```

```
:HEX TO ASC
```

```
HTOA: PUSH ACC
```

```
SWAP A
```

```
LCALL HTOAS
```

```
MOV R2,A
```

```
POP ACC
```

```
LCALL HTOAS
```

```
MOV R3,A
```

```
RET
```

```
HTOAS: ANL A,#0FH
```

```
CJNE A,#0AH,$+3
```

```
JNC HTOAS1
```

```
ORL A,#'0' ;0x30H
```

```
RET
```

```
HTOAS1: SUBB A,#9
```

```
ORL A,#'@' ;0x40H
```

```
RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        DA2      bit   P2.7
DA1      bit   P2.6
DA0      bit   P2.5

CS1      bit   P2.4
CS0      bit   P2.3

ATA_nDIOW bit   P3.4    ;Write Pulse
ATA_nDIOR bit   P3.5    ;Read Pulse
ATA_INTRQ bit   P3.6    ;interrupt Request
ATA_nRESET bit   P3.7

ATA_DATA_MS equ   P0     ;MSB D8-D15
ATA_DATA_LS equ   P1     ;LSB D0-D7

ATA_ADDR   equ   P2
ATAPI_LOW  equ   0
ATAPI_HIGH equ   0ffh

```

```

;-----

```

```
do_packet_cmd:
```

```

    mov r0,#packet
    mov r7,#16

```

```
clr_packet:
```

```

    mov @r0,#0
    inc r0
    djnz r7,clr_packet

```

```
lcall ndrq_wait
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

mov    a,#0a0h
lcall  set_cmd_register
ret
;-----

```

```
send_packet:
```

```

    lcall drq_wait

    mov    r0,#packet

    mov    r7,packet_size ;packet_size
send_packet_1:
    mov    a,@r0
    inc    r0
    push  acc
    mov    a,@r0
    inc    r0
    mov    b,a
    pop   acc

    lcall  set_data
    djnz  r7,send_packet_1

```

```

    lcall wait_irq
    lcall get_status_register
ret
;-----

```

```
get_MSF_field:
```

```

    lcall get_data
    mov    tmp1,b

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;
    lcall get_data
    mov  tmp2,a
    mov  tmp3,b
    ret
;-----
packet_cmd_error:
    mov  dptr,#scl_packet_error
    lcall write_line
;
    lcall get_error_register
    ret
;-----
init_ata_device:
    clr  ATA_nRESET
    lcall Atapi_Delaytime
    setb ATA_nRESET
    lcall Atapi_Delaytime
;
    lcall bsy_wait
    lcall get_devhead_register
    clr  ATA_DATA_LS.4      ;set device are "master"
    lcall set_devhead_register
;
    lcall bsy_wait
    lcall get_cyl_low_register
    cjne a,#14h,init_device_error
    lcall get_cyl_high_register
    cjne a,#0ebh,init_device_error

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        ljmp  init_device_done
init_device_error:
        mov  dptr,#sel_device_error
        lcall write_line
        ajmp $
init_device_done:                ;diagnostics Command
        mov  a,#90h
        lcall set_cmd_register
        lcall bsy_wait

        leall get_error_register
        jnb  acc.0,init_device_error
;After init successfull
;Display CD-ROM Ready
        mov  dptr,#sel_device_ready
        lcall write_line
        mov  r2,#5
        lcall dtsec
        rct
;-----
atapi_identify_device:

        mov  a,#2
        lcall set_cyl_high_register
        clr  a
        lcall set_cyl_low_register

        mov  a,#0a1h            ;Command
        lcall set_cmd_register
        lcall wait_irq

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

lcall get_data
clr a
jb acc.0,identify_C2
jb acc.1,identify_C2
mov a,#6
identify_C2:
jb acc.0,identify_C3
jb acc.1,identify_C3
mov a,#8
identify_C3:
mov packet_size,a
;skip next 9 word
mov r7,#26
lcall skip_device_fields
;get model number
mov a,#LINE2
lcall goto_xy

mov r7,#10
lcall print_device_ansi ;get Company Vendor

mov r7,#(10+33+1+1+46+1)
lcall skip_device_fields

lcall skip_rest_of_packet

lcall bsy_drdy_wait

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

mov  dptr,#scl_device_ready
lcall write_line
mov  r2,#5
lcall dtsec
ret

```

```

;-----

```

```

test_unit_ready:

```

```

lcall do_packet_cmd
lcall send_packet
jnb  STATUS_REG_ERR,test_unit_d
clr  c
ret

```

```

test_unit_d:

```

```

setb e
ret

```

```

;-----

```

```

wait_for_ready:

```

```

mov  dptr,#scl_chk_drive
lcall write_line

```

```

;

```

```

mov  r3,#2      ;5

```

```

wait_for_ready2:

```

```

mov  r2,#15     ;25

```

```

wait_for_ready1:

```

```

dec  r2
cjne r2,#0,xx
ajmp not_media

```

```

xx:

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    lcall Atapi_Delaytime
    lcall test_unit_ready
    jnc wait_for_ready1
    dec r3
    ejne r3,#0,wait_for_ready2
media_insert:
    lcall read_TOC
    setb flag_disc
    ret
not_media:
    mov dptr,#sel_nodisc
    lcall write_line
    clr flag_disc
    ret
;-----
read_TOC:
    mov dptr,#sel_toc_reading
    lcall write_line
    mov dptr,#sel_waiting
    lcall write_line
get_TOC:
    mov a,#0ffh
    lcall set_cyl_low_register
    lcall set_cyl_high_register

    lcall do_packet_cmd

    mov packet_cmd,#43h ;command
    mov packet_1,#00000010b ;MSF = 1
    mov packet_7,#0ffh ;allocation Lenght

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

mov   packet_8,#0ffh      ;(again, set it to full ON)

lcall send_packet

jb    STATUS_REG_ERR,read_toc_e
jb    STATUS_REG_DRQ,read_toc_get_file
setb  c
ret

read_toc_e:
mov   dptr,#scl_toc_error
lcall write_line
lcall delay_display
clr   c
ret

:-----
read_toc_get_file:
lcall get_data      ;data lenght

lcall get_data      ;get_track Limit
mov   start_track,a
;read_track_description
:

mov   r1,#TOC_file+3 ;for zero
mov   total_track,#0 ;clear

read_toc12:
lcall get_data      ;get address and Control Fields

lcall get_data      ;get Track number
mov   tmp0,a

next_MSF_read:

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;get MSF feilds
    lcall get_data
    mov tmp1,b    ;M
    mov @r1,b
    inc r1
;
    lcall get_data
    mov tmp2,a    ;S
    mov @r1,a
    inc r1
    mov tmp3,b    ;F
    mov @r1,b
    inc r1
;
    inc total_track
    lcall bsy_wait
    jb STATUS_REG_DRQ,read_toc12
    setb c
;Find total Track
    dec total_track ;total_track = total_track - 1
;
    mov dptr,#scl_TOCinfor
    lcall copy_write
;show total track
    mov xbyte_1,total_track
    lcall insert_Word

    mov disbuff+0,a    ;MSB Bit
    mov a,r3

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
mov disbuff+1,a ;LSB Bit
```

;Calculate Remaing Time

```
mov a,total_track
inc a ;find for last
mov dptr,#TOC_file
mov b,#3
mul ab
add a,dpl
mov r0,a
mov a,@r0
mov min,a
inc r0
mov a,@r0
mov sec,a
clr flag_subb
mov a,total_track
cjnc a,#1,none_equ
setb flag_subb
none_equ:
mov a,min
jb flag_subb,ad1
clr c
subb a,TOC_file+3 ;M_Start
ad1:
mov xbyte_1,a
lcall insert_Word ;Total Min
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;
    mov    disbuff+9,a
    mov    a,r3
    mov    disbuff+10,a
;
    mov    a,sec
    jb     flag_subb,ad2
    clr    c
    subb   a,TOC_file+4    ;S_Start
ad2:
    mov    xbyte_1,a
    lcall  insert_Word    ;Total Sec
    cjne   a,#'0',Wr_Char
    mov    a,#' '
Wr_Char:
    mov    disbuff+13,a
    mov    a,r3
    mov    disbuff+14,a
    lcall  write_disbuf
    ret
;-----
calculate_track:
;start play track
    mov    a,track_play    ;start index track
    mov    r1,#Start_M
    lcall  load_MSF
;end of play
    mov    a,total_track    ;stop end of TOC
    inc    a                ;add last MSF
    mov    r1,#End_M

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
lcall load_MSF
```

```
ret
```

```
-----
```

```
Load_MSF:
```

```
mov  dptr,#TOC_file
```

```
mov  b,#3
```

```
mul  ab
```

```
add  a,dpl
```

```
mov  r0,a
```

```
mov  b,#3
```

```
Load_MSF_file:
```

```
clr  a
```

```
mov  a,@r0
```

```
mov  @r1,a
```

```
inc  r0
```

```
inc  r1
```

```
djnz b,load_MSF_file
```

```
ret
```

```
-----
```

```
play_CD:
```

```
mov  dptr,#scl playCD
```

```
lcall write_line
```

```
;
```

```
Seek_track:
```

```
lcall calculate_track
```

```
;
```

```
lcall sc_test
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

lcall do_packet_cmd
mov packet_cmd,#47h
mov packet_3,Start_M
mov packet_4,Start_S
mov packet_5,Start_F
;

mov packet_6,End_M
mov packet_7,End_S
mov packet_8,End_F

lcall send_packet
jnb STATUS_REG_ERR,play_audio_cmd
lcall packet_cmd_error
clr c
clr flag_play
ret
play_audio_cmd:
setb flag_play
setb c
ret
;-----
read_subchanCmd:
mov a,#0fh
lcall set_cyl_low_register
lcall set_cyl_high_register
;

lcall do_packet_cmd
mov packet_cmd,#42h
mov packet_1,#00000010b :MSF = 1

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

mov packet_2,#01000000b ;subQ = 1
mov packet_3,#01h
mov packet_7,#0ffh ;allocation lenght
mov packet_8,#0ffh ;
;
lcall send_packet
jnb STATUS_REG_ERR,read_subcmdErr
jnb STATUS_REG_DRQ,read_subcmdRdy
ret
read_subcmdErr:
; lcall packet_cmd_error
lcall wait_for_ready
clr flag_play
ret
;-----
read_subcmdRdy:
lcall get_data ;gct audio status
mov a,b
mov audio_status,a ;wait for Stop
;
; lcall print_serial
; lcall lf

lcall get_data ;sub chanel data lenght

lcall get_data :format code ADDR

lcall get_data ;track inDex
mov index_play,a ;track <now> play
mov track_play,a ;store for seek

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

lcall get_MSF_field ;show inDex Remaing Time
;

lcall get_MSF_ficld

jnb flag_repeat,r_repeat
mov  dptr,#scl_Play_Time
ajmp ltrack
r_repeat:
mov  dptr,#scl_Play_repeat
ltrack:
lcall copy_write
;show track
mov  xbyte_1,index_play
lcall insert_Word
;Index Track
mov  disbuff+6,a ;MSB Bit
mov  a,r3
mov  disbuff+7,a ;LSB Bit

;Min Display
mov  xbyte_1,tmp1 ;Min
lcall insert_Word
cjne a,#'0',Wr_CharM
mov  a,#' '

Wr_CharM:
mov  disbuff+9,a ;MSB Bit

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

mov a,r3
mov disbuff+10,a ;LSB Bit
;Sec Display
mov xbyte_1,tmp2 ;Sec
lcall insert_Word

mov disbuff+13,a ;MSB Bit
mov a,r3
mov disbuff+14,a ;LSB Bit
;
lcall write_disbuf
acall skip_rest_of_packet
mov r2,#2
lcall dtsec
ret
;-----
atapi_rew_track:
jnb flag_play,Reword_Exit
mov a,track_play
xrl a,#1
jnz Reword_Track
Reword_Exit:
ret
Reword_track:
clr c
mov a,track_play
subb a,#1
mov track_play,a
ljmp Seck_Track
;-----

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

atapi_few_track:
    jnb  flag_play,Forword_Exit

    mov  a,total_track
    xrl  a,track_play
    jnz  Forword_track

Forword_Exit:
    ret

Forword_track:
    mov  a,track_play
    add  a,#1
    mov  track_play,a
    ljmp Seck_Track
;-----
atapi_pause:
    cpl  flag_pause
    jnb  flag_pause,none_pause

    mov  dptr,#scl_pause
    lcall write_line

none_pause:
    lcall do_packet_cmd
    mov  packet_cmd,#4bh
    clr  a

    mov  c,flag_pause
    cpl  c

    mov  acc.0,c
    mov  packet_8,a      ;pause / Resume

    lcall send_packet

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

jnb STATUS_REG_ERR,Pause_Cmd
lcall packet_cmd_error
ret

```

```

Pause_Cmd:
ret

```

```

;-----
:stop Function

```

```

atapi_stop:

```

```

mov  dptr,#scl_stopCD
lcall write_line

lcall do_packet_cmd ;clear
mov  packet_cmd,#4eh
lcall send_packet
jnb  STATUS_REG_ERR,Stop_playCmd
lcall packet_cmd_error
ljmp stop_exit

```

```

Stop_playCmd:

```

```

lcall wait_for_ready ;check disc and get TOC Disc
;after stop CD read New TOC

```

```

clr  flag_play

```

```

stop_exit:

```

```

ret

```

```

;-----
atapi_eject:

```

```

mov  dptr,#scl_stopCD

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

lcall write_line

lcall do_packet_cmd ;clear
mov packet_cmd,#1bh ;command
jb flag_eject,Load_tray

mov dptr,#scl_tray_open
lcall write_line

mov packet_4,#00000010b
lcall send_packet
cpl flag_eject
ret
Load_tray:
mov dptr,#scl_tray_close
lcall write_line

mov packet_4,#00000011b
lcall send_packet
cpl flag_eject
lcall wait_for_ready ;check disc and get TOC Disc
ret
;-----
skip_rest_of_packet:
lcall get_status_register
jnb STATUS_REG_DRQ,skip_rest_d
lcall get_data
sjmp skip_rest_of_packet
skip_rest_d:
ret

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
-----;
```

```
send_device_ansi:
```

```
    lcall get_data
    push acc
    mov a,b
    lcall tx_byte
    pop acc
    lcall tx_byte
    djnz r7,send_device_ansi
    ret
```

```
-----;
```

```
print_device_ansi:
```

```
    lcall get_data
    push acc
    mov a,b
    lcall write_lcd
    pop acc
    lcall write_lcd
    djnz r7,print_device_ansi

    mov r2,#5
    lcall dtsec
    ret
```

```
-----;
```

```
STATUS_REG_ERR bit acc.0
```

```
STATUS_REG_IDX bit acc.1
```

```
STATUS_REG_CORR bit acc.2
```

```
STATUS_REG_DRQ bit acc.3
```

```
STATUS_REG_DSC bit acc.4
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

STATUS_REG_DF bit acc.5

STATUS_REG_DRDY bit acc.6

STATUS_REG_BSY bit acc.7

get_status_register:

;set status

setb DA2

setb DA1

setb DA0

setb CS1

clr CS0

lcall read_cycle_1s

ret

get_sector_n_register:

clr DA2

setb DA1

setb DA0

setb CS1

clr CS0

lcall read_cycle_1s

ret

get_sector_c_register:

clr DA2

setb DA1

clr DA0

setb CS1

clr CS0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    lcall read_cycle_1s
    ret
;-----
ERROR_REG_AMNF bit   acc.0
ERROR_REG_TKONF bit  acc.1
ERROR_REG_ABRT bit   acc.2
ERROR_REG_MCR bit    acc.3
ERROR_REG_IDNF bit   acc.4
ERROR_REG_MC bit     acc.5
ERROR_REG_UNC bit    acc.6
ERROR_REG_RES bit    acc.7

get_error_register:
    clr  DA2
    clr  DA1
    setb DA0
    setb CS1
    clr  CS0
    lcall read_cycle_1s
    ret
;-----
get_cyl_high_register:
    setb DA2
    clr  DA1
    setb DA0
    setb CS1
    clr  CS0
    lcall read_cycle_1s
    ret
;-----

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
get_cyl_low_register:
```

```
    setb DA2
    clr DA1
    clr DA0
    setb CS1
    clr CS0
    lcall read_cycle_1s
    ret
```

```
-----
```

```
get_devhead_register:
```

```
    setb DA2
    setb DA1
    clr DA0
    setb CS1
    clr CS0
    lcall read_cycle_1s
    ret
```

```
-----
```

```
set_devhead_register:
```

```
    setb DA2
    setb DA1
    clr DA0
    setb CS1
    clr CS0
    lcall write_cycle_1s
    ret
```

```
-----
```

```
set_cyl_high_register:
```

```
    setb DA2
    clr DA1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    setb DA0
    setb CS1
    clr CS0
    lcall write_cycle_1s
    ret
;-----
set_cyl_low_register:
    setb DA2
    clr DA1
    clr DA0
    setb CS1
    clr CS0
    lcall write_cycle_1s
    ret
;-----
set_cmd_register:
    setb DA2
    setb DA1
    setb DA0
    setb CS1
    clr CS0
    lcall write_cycle_1s
    ret
;-----
;data Register
get_data:
    clr DA2
    clr DA1
    clr DA0
    setb CS1

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    clr    CS0
    lcall  read_cycle
    ret

;-----
set_data:
    clr    DA2
    clr    DA1
    clr    DA0
    setb   CS1
    clr    CS0
    lcall  write_cycle
    ret

;-----
Set_Data_Is:
    clr    DA2
    clr    DA1
    clr    DA0
    setb   CS1
    clr    CS0
    lcall  write_cycle_Is
    ret

;-----
Read_cycle:
    clr    ATA_nDIOR
    mov    a,ATA_DATA_LS
    mov    b,ATA_DATA_MS
    setb   ATA_nDIOR
    ret

;-----
read_cycle_Is:

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

mov  ATA_DATA_LS,#0ffh
clr  ATA_nDIOR
mov  a,ATA_DATA_LS
setb ATA_nDIOR
ret

```

```

;-----

```

```

write_cycle:

```

```

mov  ATA_DATA_LS,a
mov  ATA_DATA_MS,b
clr  ATA_nDIOW
nop
setb ATA_nDIOW
ret

```

```

;-----

```

```

write_cycle_1s:

```

```

mov  ATA_DATA_LS,a
clr  ATA_nDIOW
nop
setb ATA_nDIOW
ret

```

```

;-----

```

```

Wait_Irq:

```

```

setb ATA_INTRQ
jnb  ATA_INTRQ,Wait_Irq
ret

```

```

;-----

```

```

skip_device_fields:

```

```

lcall get_data
djnz r7,skip_device_fields
ret

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

;-----

bsy_wait:

```

nop
nop
nop
nop
lcall get_status_register
jb STATUS_REG_BSY,bsy_wait
ret

```

;-----

bsy_drdy_wait:

```

nop
nop
nop
nop
leall Atapi_Delaytime
lcall get_status_register
jb STATUS_REG_BSY,bsy_drdy_wait
jnb STATUS_REG_DRDY,bsy_drdy_wait
ret

```

;-----

drq_wait:

```

nop
nop
nop
nop
lcall get_status_register
jb STATUS_REG_BSY,drq_wait
jnb STATUS_REG_DRQ,drq_wait
ret

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;-----
ndrq_wait:
    nop
    nop
    nop
    nop
    lcall get_status_register
    jb STATUS_REG_BSY,drq_wait
    jb STATUS_REG_DRQ,drq_wait
    ret

```

```

;-----
Atapi_Delaytime:
    mov r4,#0ah
delay_1:  mov r5,#0ah
delay_2:  mov r6,#0
delay_3:  xch a,r7
    xch a,r7
    djnz r6,delay_3
    djnz r5,delay_2
    djnz r4,delay_1
    ret

```

```

;-----
delay_display:
    mov r2,#3
    lcall dtsec
    ret

```

```

;-----
convert_hd:
    mov xbyte_h,#0
    lcall htod16bit

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        mov     a,mem_buff+2
        lcall  htoa
        ret
;-----
insert_Word:
        lcall  convert_hd
        mov     a,r2
        ret
;-----

                :0123456701234567
scl_device_error:  db   LINE2, "ATAPI CD-ROM ERR"
scl_device_ready: db   LINE1, "ATAPI CD-ROM Rdy"

scl_chk_drive:    db   LINE1, "Disc insert Wait"
scl_nodisc:       db   LINE2, "Tray No Disc.!"

scl_toc_reading:  db   LINE1, " TOC Reading.."
scl_waitting:     db   LINE2, " Please Wait...."
scl_toc_error:    db   LINE2, " Read TOC Error!"

scl_description: db   LINE1, "Read M.S.F Files"

scl_tray_open:    db   LINE2, "-< Tray Open >- ";
scl_tray_close:  db   LINE2, "-> Tray Close<- ";

scl_playCD:       db   LINE1, "Now Playing  ";
scl_packet_error: db   LINE1, "Cmd Packet Error";

scl_TOCinfor:     db           " Track 00m 00s";

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

scl_Play_Time:    db    "Play 01 00m 00s";
scl_Play_repeat:  db    "Rep-D 01 00m 00s";
scl_Track_repeat: db    "Rep-T 01 00m 00s";

```

```

scl_stopCd:      db    LINE1, "Now Stop..  ";
scl_empty:       db    LINE2, "          ";

```

```

scl_pause:       db    LINE2, "Now CD Pause..!";

```

```

;-----

```

```

print_serial:

```

```

    lcall htoa
    mov  a,r2
    lcall tx_byte
    mov  a,r3
    lcall tx_byte
    ret

```

```

;-----

```

```

;SERVICE RX INTERRUPT

```

```

RX_INT:          PUSH  ACC

```

```

    clr  flag_serial

```

```

    jnb  ri,$

```

```

    CLR  RI

```

```

    MOV  A,SBUF          ;RECIVE DATA FROM SBUF (INTERRUPT)

```

```

    MOV  tmp,a

```

```

    setb flag_serial

```

```

SERETI:

```

```

    POP  ACC

```

```

    reti

```

```

;-----

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

RX_BYTE:  JNB  RI,$
           CLR  RI
           MOV  A,SBUF
           mov  tmp,a
           RET

```

```

-----
TX_BYTE:
           MOV  SBUF,A
           JNB  TI,$
           CLR  TI
           RET

```

```

-----
LF:      MOV  A,#0AH
           LCALL TX_BYTE
           MOV  A,#0DH
           LCALL TX_BYTE
           RET

```

```

copy_write:  mov  r0,#disbuff
             mov  b,#point
copy_Now:    clr  a
             move a,@a+dptr
             mov  @r0,a
             inc  r0
             inc  dptr
             djnz b,copy_Now
             ret

```

```

-----
;16 BIT H TO DECIMAL

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

; HEX TO DECIMAL

HTOD16BIT: CLR A ;CLEAR OUTPUT

MOV MEM_BUFF+2,A

MOV MEM_BUFF+1,A

MOV MEM_BUFF+0,A

MOV R4,#16 ;SHIFT 16 BIT

HTO16D2: MOV A,XBYTE_L

RLC A

MOV XBYTE_L,A

MOV A,XBYTE_H

RLC A

MOV XBYTE_H,A

MOV R5,#3 ;ADD DECIMAL

MOV R0,#MEM_BUFF+2 ;MSB MEM_BUFF +0

; MEM_BUFF +1

;LSB MEM_BUFF +2

HTO16D: MOV A,@R0

ADDC A,ACC

DA A

MOV @R0,A

DEC R0

DJNZ R5,HTO16D

DJNZ R4,HTO16D2

RET

DTOH: MOV R7,#8 ;8 BIT

DTOH1:

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV R0,#xbyte_d ;INDEX TO R1
CLR C
DTH2: MOV A,@R0
RRC A

PUSH PSW ;-[
JNB ACC.7,DTH3
CLR C
SUBB A,#30H
DTH3: JNB ACC.3,DTH4
CLR C
SUBB A,#03H
DTH4: MOV @R0,A
POP PSW ;-]
MOV A,xbyte_1
RRC A
MOV xbyte_1,A
DJNZ R7,DTH1
RET

end

```

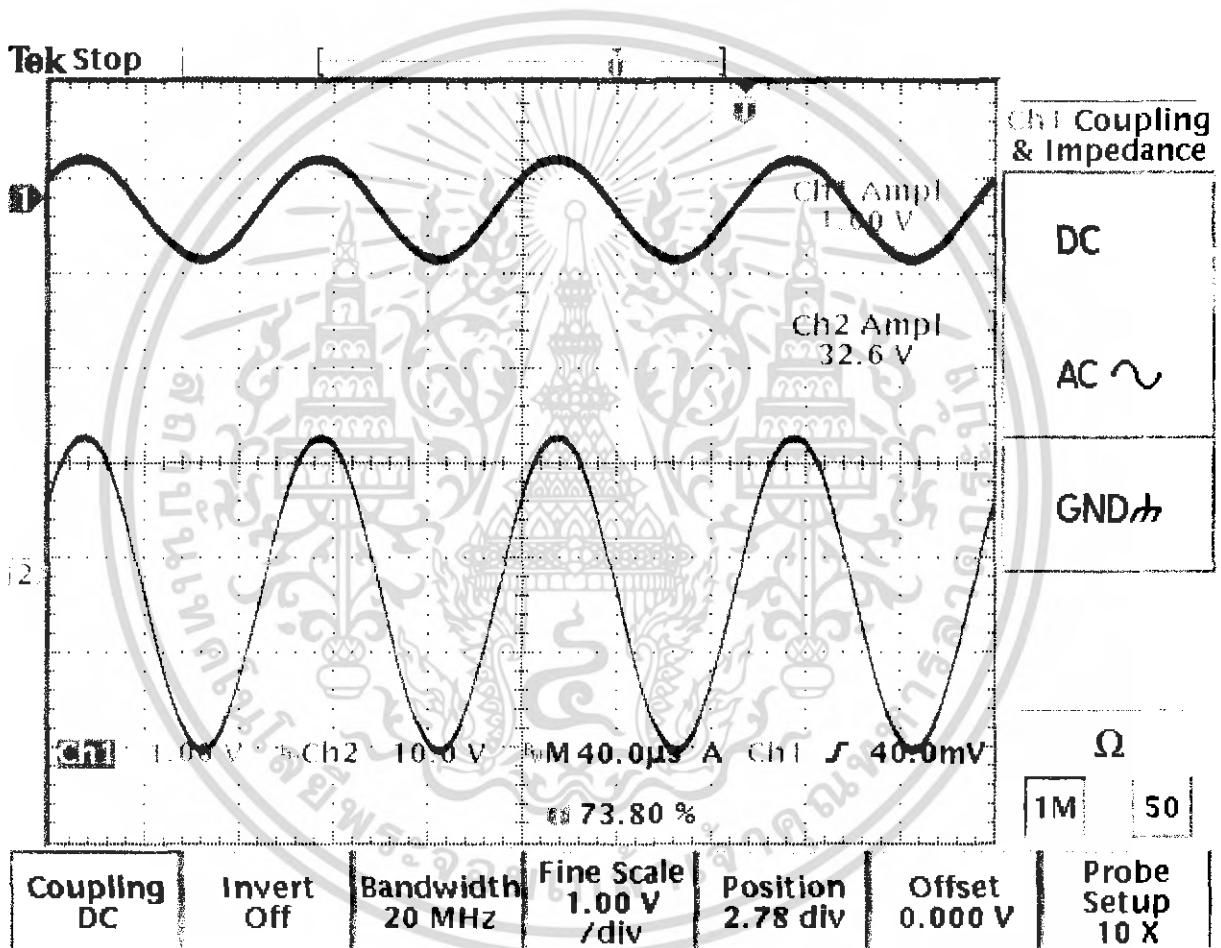
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

4.1 การวัดสัญญาณของวงจรขยาย (Power Amplifier)

ในการทดลองการวัดวงจรขยายโดยการจ่ายอินพุตเป็นรูปคลื่นไซน์ (sine wave) จำนวน 1 v และไปจับสัญญาณ เอาท์พุทได้ 32.6 v ซึ่งได้รูปคลื่นตามรูป



รูป 4.1 สัญญาณวงจรขยาย

4.2 ผลจากการทดลองวงจรขยายสัญญาณ

เมื่อจ่ายอินพุตเป็นรูปคลื่นไซน์ ขนาด 1 โวลต์ ได้ขนาดสัญญาณทางเอาท์พุทเป็นรูปคลื่นไซน์ ขนาด 32.6 โวลต์ จากผลการทดลองแสดงว่า ได้อัตราขยายเท่ากับ 32.6 เท่า ในขณะที่คำนวณจากวงจรได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

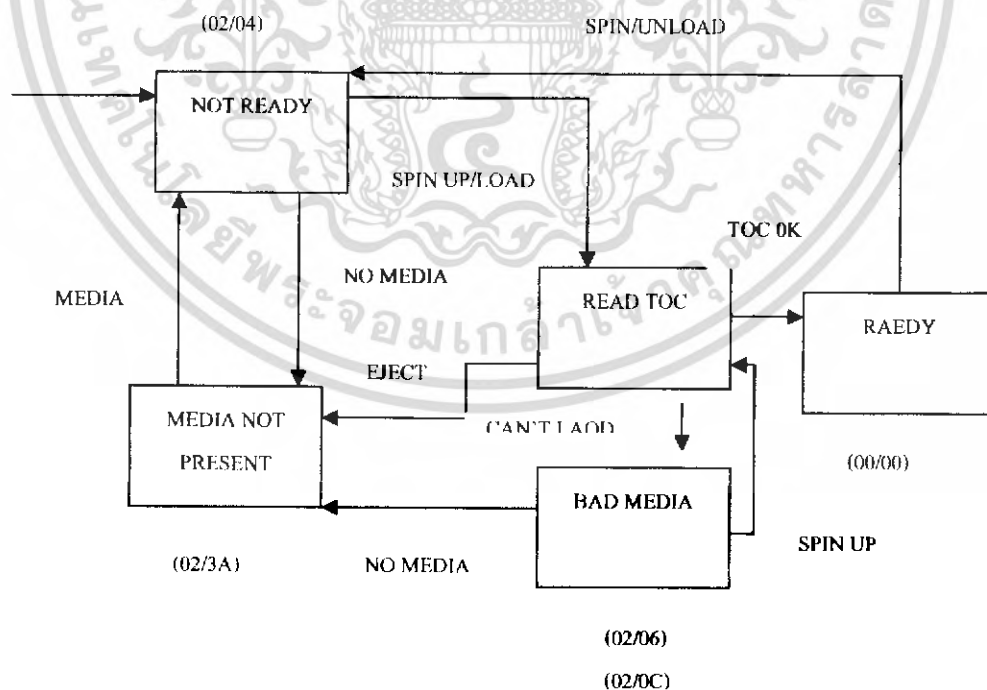
$$AV = [1 + R_f/R_i] \quad (4.1)$$

$$= [1 + 30k/1k] = 31 \text{ เท่า}$$

ขนาดของสัญญาณที่คำนวณได้เมื่อเทียบกับสัญญาณที่วัดได้ต่างกัน 1.2 โวลต์ ซึ่งอาจจะเป็นความผิดพลาดเนื่องจากค่าเปอร์เซ็นต์ error ของค่าความต้านทาน R_f กับ ค่าความต้านทาน R_i ซึ่งก็สามารถแก้ไขได้โดยใช้ค่าความต้านทานที่มีค่าเปอร์เซ็นต์ error น้อยๆ จะทำให้ได้ค่าใกล้เคียงกับค่าที่คำนวณได้มากที่สุด ส่วนความผิดเพี้ยนของสัญญาณเอาต์พุตก็จะเห็น ไม่มีซึ่งเอาต์พุตมีลักษณะเหมือนอินพุตทุกประการและจากคาส์ซีดีไอซีเบอร์ LM3875T ตัวที่อยู่ในคลาสบีนั้นก็คือคุณภาพเสียงอยู่ในระดับที่ดีกินพลังงานไฟฟ้าไม่มาก

4.3 คำสั่ง TEST UNIT READY COMMAND

คำสั่ง TEST UNIT READY COMMAND เป็นคำสั่งที่ใช้ตรวจสอบความพร้อมในการทำงานของซีดีรอม โดยมีชุดรหัสคำสั่งดังตารางที่ 4.2 และ 4.7 เป็นรายการแสดงสถานการณ์ทำงานของชุดคำสั่ง TEST UNIT READY COMMAND



รูป 4.2 สถานะการทำงานของคำสั่ง TEST UNIT READY COMMAND

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 4.1 รหัสคำสั่ง TEST UNIT READY COMMAND

Bit	7	6	5	4	3	2	1	0
Byte								
0	Operation code (00h)							
1	Reserved							
2	Reserved							
3	Reserved							
4	Reserved							
5	Reserved							
6	Reserved							
7	Reserved							
8	Reserved							
9	Reserved							
10	Reserved							
11	Reserved							

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 คำสั่ง REQUEST SENSE COMMAND

คำสั่ง REQUEST SENSE COMMAND เป็นคำสั่งที่ใช้บ่งบอกว่าในขณะนี้จะเป็นการร้องขอค่าของ Sense Key และ ASC เพื่อใช้ตรวจสอบสถานะว่าคอนแทกซ์หรือไดรฟ์ทำอะไรอยู่ โดยมีชุดคำสั่งดังตารางที่ 4.2

ตาราง 4.2 รหัสคำสั่ง REQUEST SENSE COMMAND

Bit	7	6	5	4	3	2	1	0
Byte								
0	Operation code (03h)							
1	Reserved							
2	Reserved							
3	Reserved							
4	Allocation Length							
5	Reserved							
6	Reserved							
7	Reserved							
8	Reserved							
9	Reserved							
10	Reserved							
11	Reserved							

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 คำสั่ง REQUEST SENSE READ DATA

คำสั่ง REQUEST SENSE READ DATA เป็นคำสั่งที่ใช้ในการอ่านค่า Sense Key และ ASC โดยการอ่านค่าจะต้องทำการเขียนคำสั่ง REQUEST SENSE COMMAND ก่อน

ตาราง4.3 เป็นการอ่านข้อมูลของ REQUEST SENSE READ DATA

Bit	7	6	5	4	3	2	1	0
0	Valid	Error Code (70h or 71h)						
1	Reserved							
2	Reserved	ILI	Reserved	Sense Key				
3	Information							
4	Information							
5	Information							
6	Information							
7	Additional Sense Length(n-7)							
8	Command Specific Information							
9	Command Specific Information							
10	Command Specific Information							
11	Command Specific Information							
12	Additional Sense Code(ASC)							
13	Additional Sense Length Qualifier(Optical)							
14	Field Replaceable Unit Code(Optical)							
15	SKSV	Sense Key Specific(Optical)						
16	SKSV	Sense Key Specific(Optical)						
17	Sense Key Specific(Optical)							
18	Additional Sense Length Byte							
n	Additional Sense Length Byte							

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.6 คำสั่ง READ TOC COMMAND

คำสั่ง READ TOC COMMAND เป็นคำสั่งที่ใช้บ่งบอกว่าต่อไปนี้จะเป็นการอ่านค่าจากสารบัญ เพื่อจะได้เก็บค่าของข้อมูลที่ได้จากแผ่น อย่างเช่น ใช้เก็บจำนวนและตำแหน่งของเพลง โดยมีชุดรหัสคำสั่งตามตารางที่ 4.4

ตาราง 4.4 รหัสคำสั่งของ TOC COMMAND

Bit	7	6	5	4	3	2	1	0
Byte								
0	Operation code (43h)							
1	Reserved					MSF	Reserved	
2	Reserved							
3	Reserved							
4	Allocation Length							
5	Reserved							
6	Starting Track							
7	Allocation Length(MSB)							
8	Allocation Length(LSB)							
9	Format			Reserved				
10	Reserved							
11	Reserved							

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.7 คำสั่ง READ TOC DATA

คำสั่ง READ TOC DATA เป็นคำสั่งที่ใช้ในการอ่านค่าเพื่อร้องขอตารางสารบัญ โดยก่อนการอ่านค่าจะต้องทำการเขียนคำสั่ง READ TOC COMMAND ก่อน

ตาราง 4.5 รหัสคำสั่งของ READ TOC DATA

Bit	7	6	5	4	3	2	1	0
Byte								
0	MSB TOC Data Length							
1	LSB							
2	First Track Number							
3	Last Track Number							
TOC Track Descriptors								
0	Reserved							
1	ADR				Control			
2	Track Number							
3	Reserved							
4	MSB							
5	Absolute CD-ROM Address							
6								
7	LSB							

ตาราง 4.6 ตัวกระทำคำสั่ง Control Field

Control Field	Description
00h	2 Audio With Out Pre-emphasis
01h	2 Audio With Pre-emphasis
02h	2 Audio With Out Pre-emphasis
03h	2 Audio With Pre-emphasis
04h	Digital Data
05h	Reserved
06h	Digital Data
07h	Reserved
08h-0fh	Broadcast Use

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.8 คำสั่ง MODE SENSE COMMAND

คำสั่ง MODE SENSE COMMAND เป็นคำสั่งที่ใช้ในการตรวจสอบว่ามีการเคลื่อนที่เข้าออกของถาดหรือไม่ โดยมีชุดคำสั่งดังตาราง 4.7 และ 4.8

ตาราง 4.7 รหัสคำสั่ง MODE SENSE COMMAND

Bit	7	6	5	4	3	2	1	0
Byte								
0	Operation code (5Ah)							
1	Reserved							
2	PC		Page Code					
3	Reserved							
4	Reserved							
5	Reserved							
6	Reserved							
7	Allocation Length(MSB)							
8	Allocation Length(LSB)							
9	Reserved							
10	Reserved							
11	Reserved							

ตาราง 4.8 ตัวกระทำคำสั่ง Page control

Code	Parameter
00b	Current Values
01b	Changeable Value
10b	Default Value
11b	Save Value

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.9 คำสั่ง MODE PARAMETER HEADER

คำสั่ง MODE PARAMETER HEADER เป็นคำสั่งที่ใช้ในการอ่านค่ารหัสเพื่อตรวจสอบว่าการเคลื่อนช่องถาดเข้าหรือเคลื่อนช่องถาดออกจากซีดีรอม โดยการอ่านค่าจะต้องเขียนคำสั่ง MODE SENSE COMMAND ก่อน

ตาราง 4.9 รหัสคำสั่งชุด MODE PARAMETER HEADER

Bit	7	6	5	4	3	2	1	0
Byte								
0	MSB		Mode Data Length					
1							LSB	
2			Media Type					
3			Reserved					
4			Reserved					
5			Reserved					
6			Reserved					
7								

ตาราง 4.10 ตัวกระทำคำสั่งของ Media Type Code

Code	Media Type
70h	Door Close
71h	Door Open

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.10 คำสั่ง START/STOP UNIT COMMAND

คำสั่ง START/STOP UNIT COMMAND เป็นคำสั่งที่ใช้ในการกำหนดสถานะการทำงาน Start/Stop/Eject/Load Disc โดยสามารถกำหนดสถานะการทำงานได้จากตาราง 4.11

ตาราง 4.11 รหัสคำสั่ง START/STOP UNIT COMMAND

Bit	7	6	5	4	3	2	1	0
Byte								
0	Operation code (1Bh)							
1	Reserved						Immed	
2	Reserved							
3	Reserved							
4	Reserved					I.oFj		Start
5	Reserved							
6	Reserved							
7	Reserved							
8	Reserved							
9	Reserved							
10	Reserved							
11	Reserved							

ตาราง 4.12 ตัวกระทำคำสั่ง Start/Stop And Eject

I.oFj	Start	Operation
0	0	Stop
0	1	Start
1	0	Eject
1	1	Load Disc

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.11 คำสั่ง PLAY AUDIO MSF COMMAND

คำสั่ง PLAY AUDIO MSF COMMAND เป็นคำสั่งที่ใช้ในการกำหนดจุดเริ่มต้นของเพลง และจุดสุดท้ายของเพลง โดยมีชุดรหัสคำสั่งตามตารางที่ 4.13

ตาราง 4.13 รหัสคำสั่ง PLAY AUDIO MSF COMMAND

Bit	7	6	5	4	3	2	1	0
Byte								
0	Operation code (47h)							
1	Reserved							
2	Reserved							
3	Starting M Field							
4	Starting S Field							
5	Starting F Field							
6	Ending M Field							
7	Ending S Field							
8	Ending F Field							
9	Reserved							
10	Reserved							
11	Reserved							

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.12 คำสั่ง READ SUBCHANNEL COMMAND

คำสั่ง READ SUBCHANNEL COMMAND เป็นคำสั่งที่ใช้ในการอ่านค่ารายละเอียดของแต่ละแทร็ก หรือใช้ในการเก็บสถานะของเพลง โดยมีชุดรหัสคำสั่งตามตารางที่ 4.14

ตาราง 4.14 รหัสคำสั่ง READ SUBCHANNEL COMMAND

Bit	7	6	5	4	3	2	1	0
Byte								
0	Operation code (42h)							
	Reserved						MSF	Reserved
2	Reserved	SUBQ	Reserved					
3	Reserved							
4	Reserved							
5	Reserved							
6	Track Number							
7	Allocation Length (MSB)							
8	Allocation Length (LSB)							
9	Reserved							
10	Reserved							
11	Reserved							

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.13 คำสั่ง CD-ROM CURRENT POSITION DATA FORMAT

คำสั่ง CD-ROM CURRENT POSITION DATA FORMAT เป็นคำสั่งที่ใช้ในการอ่านค่าที่ตำแหน่งปัจจุบันในขณะนั้น โดยก่อนการอ่านค่าจะต้องเขียนคำสั่ง READ SUBCHANNEL COMMAND ก่อน

ตาราง 4.15 รหัสคำสั่ง CD-ROM CURRENT POSITION DATA FORMAT

Bit	7	6	5	4	3	2	1	0
Byte								
Sub Channel Data Header								
0	Reserved							
1	Audio Status							
2	Sub Channel Data Length							
3								LSB
CD-ROM Current Position Data Block								
4	Sub Channel Data(Format Code 01h)							
5	ADR				Control			
6	Track number							
7	Track number							
8	Absolute CD-ROM Address							
9								
10								
11								LSB
12	Track Relative CD-ROM Address							
13								
14								
15								LSB

ตาราง 4.16 รหัสคำสั่ง Audio Status

Status	Description
00h	Audio Status Byte Not Supported/Not Valid
11h	Play Operation In Progress
12h	Play Operation Pause
13h	Play Operation Successfully Completed
14h	Play Operation Stop Due To Error
15h	No Current Audio Status To Return

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.14 คำสั่ง MSF ADDRESS FORMAT

คำสั่ง MSF ADDRESS FORMAT เป็นรูปแบบตำแหน่งที่ใช้ในการเก็บค่าของนาฬิกา วินาที และเฟรม

ตาราง 4.17 รหัสคำสั่ง MSF ADDRESS FORMAT

Bit	7	6	5	4	3	2	1	0
Byte								
0	Reserved							
1	M Field							
2	S Field							
3	F Field							

4.15 สรุปผลรหัสคำสั่งที่ใช้ในซีดีรอม

สรุปผลคำสั่งต่างๆที่รวบรวมรหัสชุดคำสั่งที่ใช้ในการควบคุมการเล่นเพลง AUDIO ในโครงการนี้คือ

ตาราง 4.18 สรุปรหัสที่ใช้ในซีดีรอม

Command Description	Opcode
PAUSE/RESUME	4BH
PLAY AUDIO MSF	47H
READ TOC	43H
REQUEST SENSE	03H
TEST UNIT READY	00H
MODE SENSE	5AH
START/STOP UNIT	1BH
RAED SUB-CHANNEL	42H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.16 รายละเอียดรีจิสเตอร์ SENSE KEY/ ASC

รีจิสเตอร์ SENSE KEY/ ASC เป็นรีจิสเตอร์ที่ใช้ในการบ่งบอกค่าตามตารางที่ 4.19 ว่าในขณะนั้นกำลังทำอะไรอยู่

ตาราง 4.19 รหัสคำสั่ง SENSE KEY/ ASC

Sense key	ASC	Description
00	00	NO ADDITIONAL SENSE INFORMATION
02	04	LOGICAL DRIVE NOT READY
02	06	NO REFERENCE POSITION FOUND
02	30	CANNOT READ MEDIUM
02	3A	MEDIUM NOT PRESENT
02	53	MEDIUM REMOVAL PREVENTED
02	57	UNABLE TO RECOVER TOC
03	02	NO SEEK COMPLETE
04	B6	MEDIUM LOAD MECHANISM FAILED
05	00	PLAY OPERATION IN PROGRESS
05	1A	PARAMETER LIST LENGTH ERROR
05	20	INVALID COMMAND OPERATION CODE
05	21	LOGICAL BLOCK ADDRESS OUT OF RANGE
05	24	INVALID FIELD IN COMMAND PACKET
05	39	SAVING PARAMETER
05	63	END OF USE AREA ENCOUNTERED ON THIS TRACK
05	64	ILLEGAL MODE FOR THIS TRACK
06	28	NOT ILLEGAL READY TO READY TRANSITION
06	29	POWER ON /RESET
0B	B9	PLAY OPERATION ABORTED

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปและวิเคราะห์ผลการดำเนินการ

โครงการนี้มีวัตถุประสงค์เพื่อศึกษาหลักการควบคุมและการเลือกใช้ฟังก์ชันจากการทำงานของซีดีรอม โดยใช้ซีดีรอมเป็นเครื่องเล่นเพลง (CD PLAYER) และได้ใช้การประยุกต์ตัวไมโครคอนโทรลเลอร์ตระกูล MCS-51 เบอร์ AT89S8252 เป็นตัวควบคุมการทำงานของซีดีรอมแทนที่จะใช้โปรแกรมจากในคอมพิวเตอร์ เพื่อที่จะได้เรียนรู้ถึงการติดต่อสื่อสารกับเครื่องซีดีรอม (IDE Interface) ซึ่งมีการติดต่อสื่อสารกันแบบมาตรฐานของ ATPI (ATPI Protocol) และการที่เลือกใช้ไมโครคอนโทรลเลอร์ตระกูล MCS-51 เบอร์ AT89C52 และ AT89C2051 เนื่องจากมีข้อดีหลายประการ คือมันมีหน่วยความจำแบบแฟลช ซึ่งสามารถที่จะลบและเขียนใหม่ได้เป็นพันครั้ง จึงเหมาะกับการใช้ในโครงการนี้

ในโครงการนี้จะเป็นการศึกษาการใช้งานซีดีรอมและการเลือกใช้ฟังก์ชันการทำงานเบื้องต้นเท่านั้น โดยสามารถทำการควบคุมการทำงานได้จากรีโมทและสวิตช์ปุ่มกดซึ่งมี PLAY, EJECT, STOP, NEXT, BACK ซึ่งได้ใช้ในการฟังเพลงจริงปรากฏว่ามันใช้ได้ติดตามวัตถุประสงค์
ปัญหาที่พบ

การทำงานของโครงการนี้มีปัญหา และ อุปสรรคคือยังทำความเข้าใจถึงการควบคุมการทำงานของซีดีรอม การเข้าถึงแทรคซีดีเพื่อที่จะเลือกเพลงยังไม่มากนักและการเขียนโปรแกรมภาษาแอสเซมบลีเพื่อใช้ให้ไมโครคอนโทรลเลอร์ควบคุมการทำงานของซีดีรอมก็ไม่มากทำให้การทำโครงการติดปัญหา

แนวทางการแก้ปัญหา

ต้องทำการค้นคว้าหาข้อมูลเกี่ยวกับการควบคุมการทำงานของซีดีรอม การเข้าถึงแทรคของแผ่นซีดี และการเขียนโปรแกรมภาษาแอสเซมบลีเพิ่ม

หนังสืออ้างอิง

1. ชีรวัฒน์ ประกอบผล, “ภาษาเอสเซมบลีสำหรับ MCS-51”, พิมพ์ครั้งที่ 2, กรุงเทพฯ, สำนักพิมพ์ ส.ศ.ท, หน้า 47-53
2. T. HANAN, “Small From Factor Committee,” ATA Packet Interface for CD-ROMs SFF-8020i, Revision2.6 Proposed, January 22, 1996
3. วารสาร เซมิคอนดักเตอร์ อิเล็กทรอนิกส์, ฉบับที่ 271 , มกราคม, 2548



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LM3875 Overture™ Audio Power Amplifier Series

High-Performance 56W Audio Power Amplifier

General Description

The LM3875 is a high-performance audio power amplifier capable of delivering 56W of continuous average power to an 8Ω load with 0.1% THD+N from 20Hz to 20kHz.

The performance of the LM3875, utilizing its Self Peak Instantaneous Temperature (°Ke) (SPIKe™) protection circuitry, puts it in a class above discrete and hybrid amplifiers by providing an inherently, dynamically protected Safe Operating Area (SOA). SPIKe protection means that these parts are completely safeguarded at the output against overvoltage, undervoltage, overloads, caused by shorts to the supplies, thermal runaway, and instantaneous temperature peaks.

The LM3875 maintains an excellent signal-to-noise ratio of greater than 95dB(min) with a typical low noise floor of 2.0μV. It exhibits extremely low THD+N values of 0.06% at the rated output into the rated load over the audio spectrum, and provides excellent linearity with an IMD (SMPTE) typical rating of 0.004%.

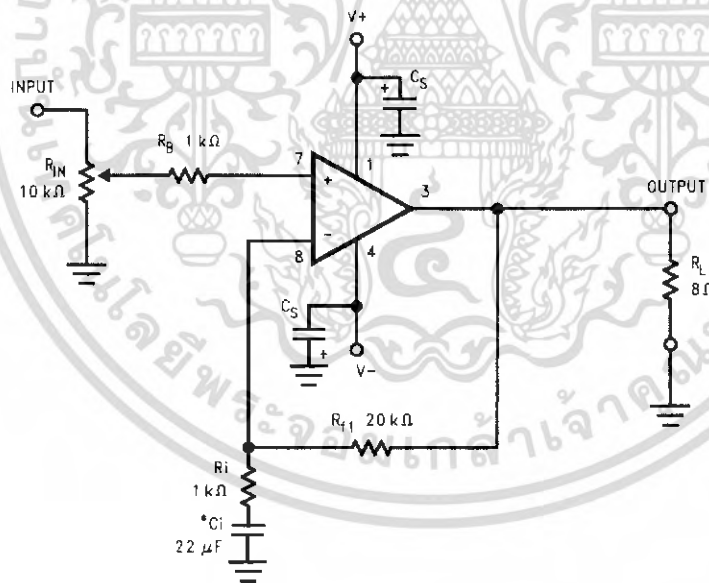
Features

- 56W continuous average output power into 8Ω
- 100W instantaneous peak output power capability
- Signal-to-Noise Ratio >95dB (min)
- Output protection from a short to ground or to the supplies via internal current limiting circuitry
- Output over-voltage protection against transients from inductive loads
- Supply under-voltage protection, not allowing internal biasing to occur when $|V+| + |V-| < 12V$, thus eliminating turn-on and turn-off transients
- 11 lead TO-220 package
- Wide supply voltage range: $|V+| + |V-| = 20V$ to 84V

Applications

- Component or compact stereos
- Self-powered speakers
- Surround-sound amplifiers
- High-end stereo TVs

Typical Application



*Optional components dependent upon specific design requirements. Refer to the External Components Description section for a component function description.

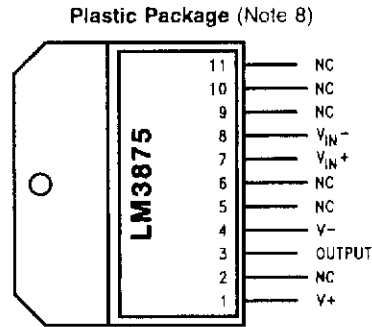
FIGURE 1. Typical Audio Amplifier Application Circuit

Overture™ and SPIKe™ are trademarks of National Semiconductor Corporation.

LM3875 Overture™ Audio Power Amplifier Series High-Performance 56W Audio Power Amplifier

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Connection Diagram



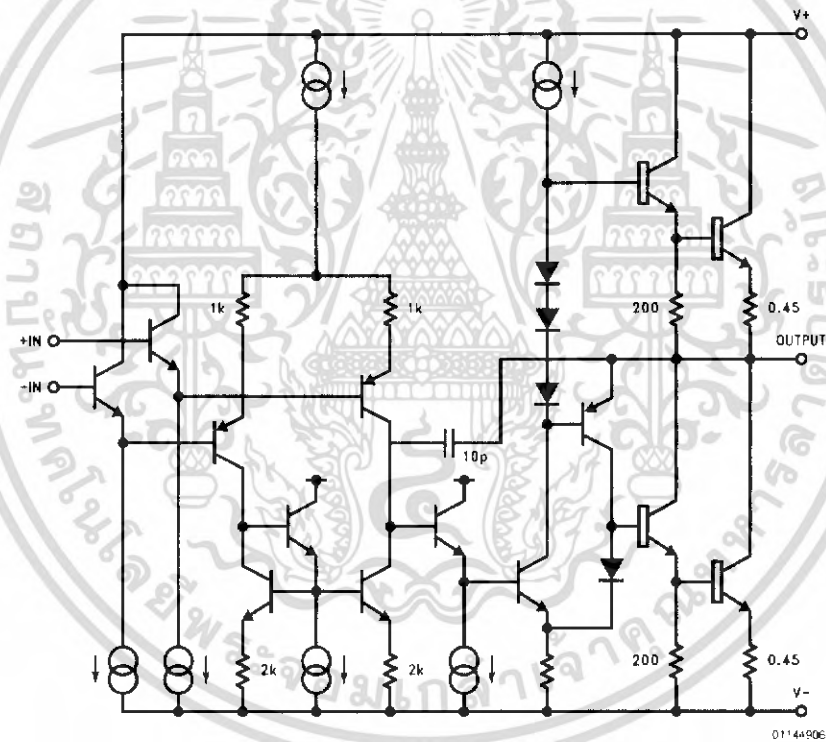
01144902

Top View

Order Number LM3875T or LM3875TF
See NS Package Number TA11B for
Staggered Lead Non-Isolated Package
or TF11B for Staggered Lead Isolated Package

Equivalent Schematic

(Excluding active protection circuitry)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Absolute Maximum Ratings (Notes 1,

2)

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/ Distributors for availability and specifications.

Supply Voltage $ V^+ + V^- $ (No Signal)	94V
Supply Voltage $ V^+ + V^- $ (Input Signal)	84V
Common Mode Input Voltage	$(V^+ \text{ or } V^-)$ and $ V^+ + V^- \leq 80V$
Differential Input Voltage	60V
Output Current	Internally Limited
Power Dissipation (Note 3)	125W
ESD Susceptibility (Note 4)	2500V
Junction Temperature (Note 5)	150°C
Soldering information	
T package (10 seconds)	260°C

Storage Temperature -40°C to $+150^\circ\text{C}$

Thermal Resistance

 θ_{JC} 1°C/W θ_{JA} 43°C/W**Operating Ratings** (Notes 1, 2)

Temperature Range

 $T_{MIN} \leq T_A \leq T_{MAX}$ $-20^\circ\text{C} \leq T_A \leq +85^\circ\text{C}$ Supply Voltage $|V^+| + |V^-|$ 20V to 84V

Note: Operation is guaranteed up to 84V, however, distortion may be introduced from the SPIKE Protection Circuitry when operating above 70V if proper thermal considerations are not taken into account. Refer to the Thermal Considerations section for more information. (See SPIKE Protection Response)

Electrical Characteristics (Notes 1, 2)

The following specifications apply for $V^+ = +35V$, $V^- = -35V$ with $R_L = 8\Omega$ unless otherwise specified. Limits apply for $T_A = 25^\circ\text{C}$.

Symbol	Parameter	Conditions	LM3875		Units (Limits)
			Typical (Note 6)	Limit (Note 7)	
$ V^+ + V^- $	Power Supply Voltage			20 84	V (Min) V (Max)
** P_O	Output Power (Continuous Average)	THD + N = 0.1% (Max) $f = 1 \text{ kHz}, f = 20 \text{ kHz}$	56	40	W (Min)
Peak P_O	Instantaneous Peak Output Power		100		W
THD + N	Total Harmonic Distortion Plus Noise	40W, $20 \text{ Hz} \leq f \leq 20 \text{ kHz}$ $A_V = 26 \text{ dB}$	0.06		%
**SR	Slew Rate (Note 9)	$V_{IN} = 1.414 V_{rms}$, $f = 10 \text{ kHz}$ Square-wave, $R_L = 2 \text{ k}\Omega$	11	5	V/ μs (Min)
* I_Q	Total Quiescent Power Supply Current	$V_{CM} = 0V$, $V_O = 0V$, $I_O = 0 \text{ mA}$	30	70	mA (Max)
* V_{OS}	Input Offset Voltage	$V_{CM} = 0V$, $I_O = 0 \text{ mA}$	1	10	mV (Max)
I_B	Input Bias Current	$V_{CM} = 0V$, $I_O = 0 \text{ mA}$	0.2	1	μA (Max)
I_{OS}	Input Offset Current	$V_{CM} = 0V$, $I_O = 0 \text{ mA}$	0.01	0.2	μA (Max)
I_O	Output Current Limit	$ V^+ = V^- = 10V$, $t_{on} = 10 \text{ ms}$, $V_O = 0V$	6	4	A (Min)
* V_{od}	Output Dropout Voltage	$ V^- - V_O $, $V^+ = 20V$, $I_O = +100 \text{ mA}$ $ V_O - V^+ $, $V^- = -20V$, $I_O = -100 \text{ mA}$	1.6 2.7	5 5	V (Max) V (Max)
*PSRR	Power Supply Rejection Ratio	$V^+ = 40V$ to $20V$, $V^- = -40V$, $V_{cm} = 0V$, $I_O = 0 \text{ mA}$ $V^+ = 40V$, $V^- = -40V$ to $-20V$, $V_{cm} = 0V$, $I_O = 0 \text{ mA}$	120 120	85 85	dB (Min)
*CMRR	Common Mode Rejection Ratio	$V^+ = 60V$ to $20V$, $V^- = -20V$ to $-60V$, $V_{cm} = 20V$ to $-20V$, $I_O = 0 \text{ mA}$	120	80	dB (Min)
* A_{VOL}	Open Loop Voltage Gain	$ V^+ = V^- = 40V$, $R_L = 2 \text{ k}\Omega$, $\Delta V_O = 60V$	120	90	dB (Min)
GBWP	Gain-Bandwidth Product	$ V^+ = V^- = 40V$ $f_O = 100 \text{ kHz}$, $V_{IN} = 50 \text{ mVrms}$	8	2	MHz (Min)
** e_{IN}	Input Noise	1HF - A Weighting Filter $R_{IN} = 600\Omega$ (Input Referred)	2.0	8.0	μV (Max)

Electrical Characteristics (Notes 1, 2) (Continued)

The following specifications apply for $V^+ = +35V$, $V^- = -35V$ with $R_L = 8\Omega$ unless otherwise specified. Limits apply for $T_A = 25^\circ C$.

Symbol	Parameter	Conditions	LM3875		Units (Limits)
			Typical (Note 6)	Limit (Note 7)	
SNR	Signal-to-Noise Ratio	$P_O = 1W$, A-Weighted, Measured at 1 kHz, $R_S = 25\Omega$	98 dB		dB
		$P_O = 40W$, A-Weighted, Measured at 1 kHz, $R_S = 25\Omega$	114 dB		dB
		$P_{pk} = 100W$, A-Weighted, Measured at 1 kHz, $R_S = 25\Omega$	122 dB		dB
IMD	Intermodulation Distortion Test	60 Hz, 7 kHz, 4:1 (SMPTE) 60 Hz, 7 kHz, 1:1 (SMPTE)	0.004 0.006		%

*DC Electrical Test; refer to Test Circuit #1

**AC Electrical Test; refer to Test Circuit #2.

Note 1: Absolute Maximum Ratings indicate limits beyond which damage to the device may occur. Operating Ratings indicate conditions for which the device is functional, but do not guarantee specific performance limits. Electrical Characteristics state DC and AC electrical specifications under particular test conditions which guarantee specific performance limits. This assumes that the device is within the Operating Ratings. Specifications are not guaranteed for parameters where no limit is given, however, the typical value is a good indication of device performance.

Note 2: All voltages are measured with respect to supply GND, unless otherwise specified.

Note 3: For operating at case temperatures above $25^\circ C$, the device must be derated based on a $150^\circ C$ maximum junction temperature and a thermal resistance of $\theta_{JC} = 1.0^\circ C/W$ (junction to case). Refer to the Thermal Resistance figure in the Application Information section under **Thermal Considerations**.

Note 4: Human body model, 100 pF discharged through a 1.5 k Ω resistor.

Note 5: The operating junction temperature maximum is $150^\circ C$, however, the instantaneous Safe Operating Area temperature is $250^\circ C$.

Note 6: Typicals are measured at $25^\circ C$ and represent the parametric norm.

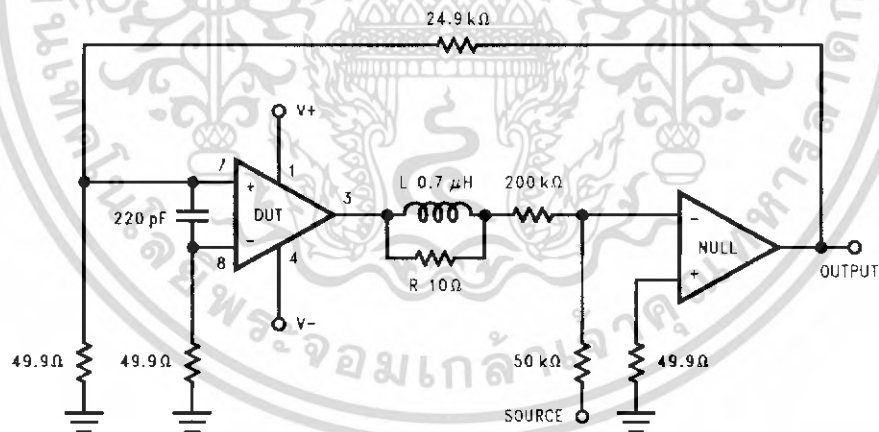
Note 7: Limits are guaranteed to National's AOQL (Average Outgoing Quality Level).

Note 8: The LM3875T package TA11B is a non-isolated package, setting the top of the device and the heat sink at V^- potential when the LM3875 is directly mounted to the heat sink using only thermal compound. If a mica washer is used in addition to thermal compound, θ_{CS} (case to sink) is increased, but the heat sink will be isolated from V^- .

Note 9: The feedback compensation network limits the bandwidth of the closed-loop response and so the slew rate will be reduced due to the high frequency roll-off. Without feedback compensation, the slew rate is typically $16V/\mu s$.

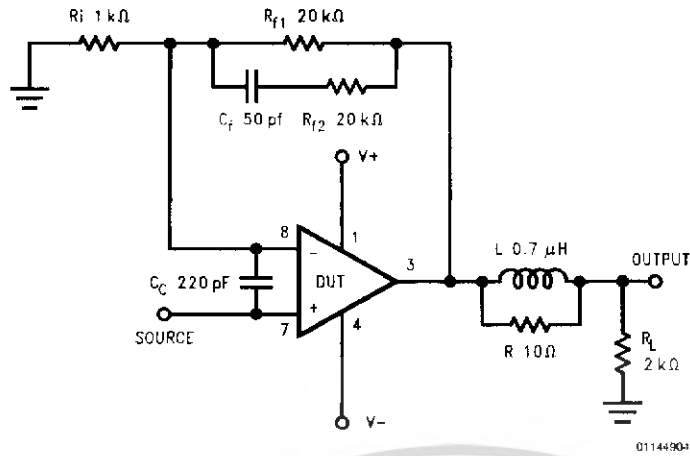
Note 10: The output dropout voltage is the supply voltage minus the clipping voltage. Refer to the Clipping Voltage vs. Supply Voltage graph in the **Typical Performance Characteristics** section.

Test Circuit #1 (DC Electrical Test Circuit)

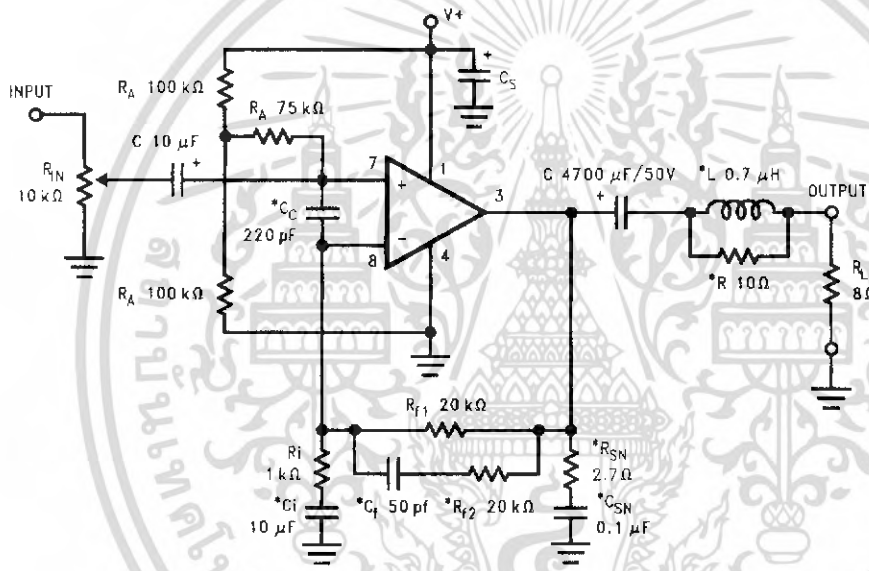


01449J3

Test Circuit #2 (AC Electrical Test Circuit)



Single Supply Application Circuit



*Optional components dependent upon specific design requirements. Refer to the External Components Description section for a component function description.

FIGURE 2. Typical Single Supply Audio Amplifier Application Circuit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

External Components Description

(Figure 1 and Figure 2)

Components	Functional Description
1. R_{IN}	Acts as a volume control by setting the voltage level allowed to the amplifier's input terminals.
2. R_A	Provides DC voltage biasing for the single supply operation and bias current for the positive input terminal.
3. C_A	Provides bias filtering.
4. C	Provides AC coupling at the input and output of the amplifier for single supply operation.
5. R_B	Prevents currents from entering the amplifier's non-inverting input which may be passed through to the load upon power-down of the system due to the low input impedance of the circuitry when the under-voltage circuitry is off. This phenomenon occurs when the supply voltages are below 1.5V.
6. $*C_C$	Reduces the gain (bandwidth of the amplifier) at high frequencies to avoid quasi-saturation oscillations of the output transistor. The capacitor also suppresses external electromagnetic switching noise created from fluorescent lamps.
7. R_i	Inverting input resistance to provide AC Gain in conjunction with R_{f1} .
8. $*C_i$	Feedback capacitor. Ensures unity gain at DC. Also a low frequency pole (highpass roll-off) at: $f_c = 1/(2\pi R_i C_i)$.
9. R_{f1}	Feedback resistance to provide AC Gain in conjunction with R_i .
10. $*R_{f2}$	At higher frequencies feedback resistance works with C_i to provide lower AC Gain in conjunction with R_{f1} and R_i . A high frequency pole (lowpass roll-off) exists at: $f_c = [R_{f1} R_{f2}] (s + 1/R_{f2} C_i) / [(R_{f1} + R_{f2}) (s + 1/C_i (R_{f1} + R_{f2}))]$.
11. $*C_f$	Compensation capacitor that works with R_{f1} and R_{f2} to reduce the AC Gain at higher frequencies.
12. $*R_{SN}$	Works with C_{SN} to stabilize the output stage by creating a pole that eliminates high frequency oscillations.
13. $*C_{SN}$	Works with R_{SN} to stabilize the output stage by creating a pole that eliminates high frequency oscillations. $f_c = 1/(2\pi R_{SN} C_{SN})$.
14. $*L$	Provides high impedance at high frequencies so that R may decouple a highly capacitive load and reduce the Q of the series resonant circuit due to capacitive load. Also provides a low impedance at low frequencies to short out R and pass audio signals to the load.
15. $*R$	
16. C_S	Provides power supply filtering and bypassing.

*Optional components dependent upon specific design requirements. Refer to the Application Information section for more information.

OPTIONAL EXTERNAL COMPONENT INTERACTION

Although the optional external components have specific desired functions that are designed to reduce the bandwidth and eliminate unwanted high frequency oscillations they may cause certain undesirable effects when they interact. Interaction may occur for components whose reactances are in close proximity to one another. One example would be the coupling capacitor, C_C , and the compensation capacitor, C_f .

These two components act as low impedances to certain frequencies which will couple signals from the input to the output. Please take careful note of basic amplifier component functionality when designing in these components.

The optional external components shown in Figure 2 and described above are applicable in both single and split voltage supply configurations.