

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เครื่องบันทึกคลื่นไฟฟ้าหัวใจ

ELECTROCARDIOGRAM RECORDER

โดย
นาย เลอศักดิ์ แซ่ลี

เลขหมู่.....
เลขทะเบียน.....**62400**
วัน,เดือน,ปี.....**17 ส.ค. 2549**

b. **11623032**
i.

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาอิเล็กทรอนิกส์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องบันทึกคลื่นไฟฟ้าหัวใจ
ELECTROCARDIOGRAM RECORDER

โดย

นาย เลอศักดิ์ แซ่ตั้ง เลขประจำตัวนักศึกษา 45015289



อาจารย์ที่ปรึกษา

ดร.กิติพล ชิตสกุล

ปริญญานิพนธ์สำหรับปริญญาวิทยาศาสตรบัณฑิต
สาขาวิชาอิเล็กทรอนิกส์
คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโท ปีการศึกษา 2548

ภาควิชา อิเล็กทรอนิกส์

คณะ วิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เครื่องบันทึกคลื่นไฟฟ้าหัวใจ

ELECTROCARDIOGRAM RECORDER

ผู้จัดทำ

นายเลิศศักดิ์ แซ่ลี

เลขประจำตัวนักศึกษา 45015289



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องบันทึกคลื่นไฟฟ้าหัวใจ
ELECTROCARDIOGRAM RECORDER

นายเลิศศักดิ์ แซ่ลี รหัส 45015289
ดร.กิติพล ชิตสกุล อาจารย์ที่ปรึกษา
ปีการศึกษา 2548

บทคัดย่อ

ผู้ป่วยโรคหัวใจระยะเริ่มแรกมักเกิดอาการไม่แน่นอนและอาจตรวจไม่พบเมื่อไปพบแพทย์หรือจำเป็นต้องใช้ขั้นตอนการตรวจวินิจฉัยที่ยุ่งยากจึงจะตรวจพบ เช่นเดียวกับผู้ที่มีอาการคล้ายกับโรคหัวใจ ซึ่งการวินิจฉัยจะใช้คลื่นไฟฟ้าหัวใจที่บันทึกในขณะที่เกิดอาการเป็นหลัก เครื่องบันทึกคลื่นไฟฟ้าหัวใจแบบพกติดตัวจึงเป็นที่ต้องการอย่างยิ่ง

โครงการนี้เป็นการศึกษาและออกแบบเครื่องบันทึกคลื่นไฟฟ้าหัวใจที่ใช้วงจรขยายอินสตุเมนเตชัน วงจรขยายสัญญาณและปรับศูนย์อัตโนมัติ และวงจรกำจัดสัญญาณรบกวน รับสัญญาณจากอิเล็กโทรดติดบริเวณหน้าอก แล้วแปลงคลื่นไฟฟ้าหัวใจเป็นสัญญาณดิจิทัลและใช้ไมโครคอนโทรลเลอร์ 89S8252 ควบคุมเพื่อนำไปบันทึกใน Flash Memory

ELECTROCARDIOGRAM RECORDER

Mr. Lersak Saelee ID 45015289

Dr. Kitipol Chitsakul (Advisor)

Academic year 2005

Abstract

Electrocardiogram (ECG) signals provide essential information to cardiologists for diagnosis heart diseases in patients. Since the symptoms of heart disease at preliminary stage occur in the unpredictable manner, it is difficult for physicians to diagnose from ECG. Thus a mobile ECG recorder is essentially needed.

An ECG recorder presented in this thesis employs an instrumentation circuit, auto zero adjust circuit and 50 Hz notch filter for acquiring the ECG from electrodes placed on the chest of the patient. The signal is then digitized by an analog to digital converter under control of a microcontroller, MCS-51 family (89S8252) for storing digitally in flash memories.

กิตติกรรมประกาศ

การจัดทำปริญญาบัตรฉบับนี้ ได้รับความสำเร็จเนื่องจากได้รับการสนับสนุนและกำลังใจจากบุคคลหลายฝ่าย ผู้เขียนขอขอบพระคุณคุณพ่อ คุณแม่ และครอบครัว ที่คอยให้การสนับสนุนและกำลังใจมาโดยตลอด

ขอขอบพระคุณ ดร. กิติพล ชิตสกุล ในฐานะอาจารย์ที่ปรึกษาที่คอยให้แนวความคิด คำแนะนำ ผลักดันให้มีกำลังใจทำงาน

ขอขอบพระคุณอาจารย์ทุกท่านที่อบรมสั่งสอน และให้ความรู้

ขอขอบคุณ นายศักดิ์พันธ์ คล้ายดอกจันทร์ ที่ได้สละเวลาอันมีค่าในการให้คำแนะนำ ตลอดจน และท้ายที่สุดนี้ ขอขอบคุณเพื่อนๆทุกคนที่ให้การสนับสนุนการทำปริญญาบัตรครั้งนี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VI
สารบัญรูป	VII
บทที่ 1 บทนำ	1
1.1 ความสำคัญและความเป็นมาของโครงการ	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 แนวคิดของโครงการ	1
1.4 โครงสร้างของปริิญาานิพนธ์	1
บทที่ 2 พื้นฐานของเครื่องวัดคลื่นไฟฟ้าหัวใจ	3
2.1 พื้นฐานของเครื่องวัดคลื่นไฟฟ้าหัวใจ	3
2.2 รูปร่างของสัญญาณคลื่นไฟฟ้าหัวใจ	5
2.3 วิธีการวัด	6
บทที่ 3 การออกแบบ	7
3.1 วงจรอนาล็อก	7
3.2 วงจรดิจิทัล	8
3.3 วงจรแปลงสัญญาณอนาล็อกเป็นดิจิทัล	8
3.4 Real Time Clock	11
3.5 ยูเอสบี	13
3.6 Flash Memory	16
3.7 LCD	24
บทที่ 4 หลักการและการออกแบบโปรแกรมบนเครื่องพีซี	32
4.1 หลักการทำงานของเครื่อง	32
4.2 โฟลชาร์ตแสดงการทำงานของเครื่อง	33
บทที่ 5 การทดลองและผลการทดลอง	38

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
5.1 การหาค่า CMRR	38
5.2 การวัดคลื่นไฟฟ้าหัวใจ	39
5.3 การทดลองซอฟต์แวร์	41
5.4 การแสดงผล	44
บทที่ 6 บทสรุป	45
6.1 สรุป	45
6.2 วิจารณ์และข้อเสนอแนะ	46
หนังสืออ้างอิง	47
ภาคผนวก	48



สารบัญตาราง

	หน้า
ตารางที่ 1 ตารางแนะนำการเลือกผลิตภัณฑ์	18
ตารางที่ 2 ตารางการทำงานของบัส	20
ตารางที่ 3 ตารางคำสั่งของ Flash Memory	22
ตารางที่ 4 ตารางแอดเดรสของเซกเตอร์	23
ตารางที่ 5 ตารางแสดงตำแหน่งขาต่างๆที่ใช้เชื่อมต่อกับ LCD	25
ตารางที่ 6 ตารางแสดงผลการทดลองหาค่า CMRR	39



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

	หน้า
2.1 รูปผังของเครื่อง อี.ซี.จี. และส่วนประกอบต่างๆ	4
2.2 รูปสัญญาณไฟฟ้าหัวใจที่เป็นปกติ	5
2.3 รูปตำแหน่งของการวัดคลื่นไฟฟ้าหัวใจ	6
2.4 รูปตำแหน่งการวัดคลื่นไฟฟ้าหัวใจที่หน้าอก	6
3.1 รูปแบบของไอซีเบอร์ LTC 1298	9
3.2 รูปการเชื่อมต่อ LTC1298 เข้ากับส่วนของ SPI	10
3.3 รูปโปรแกรมมิ่งไดอะแกรมของไอซีเบอร์ LTC 1298	10
3.4 รูปแบบของไอซีเบอร์ DS1302	11
3.5 รูปการต่อเข้ากับไมโครคอนโทรลเลอร์	12
3.6 รูปโปรแกรมมิ่งไดอะแกรมเวลาอ่านข้อมูล	12
3.7 รูปโปรแกรมมิ่งไดอะแกรมเวลาเขียนข้อมูล	13
3.8 รูปการเชื่อมต่อยูเอสบี	14
3.9 รูปแบบของไอซีเบอร์ AM29F016B	18
3.12 รูปวงจรถอดลอกเครื่องบันทึกคลื่นไฟฟ้าหัวใจ	29
3.13 รูปวงจรถอดลอกเครื่องบันทึกคลื่นไฟฟ้าหัวใจ 1	30
3.14 รูปวงจรถอดลอกเครื่องบันทึกคลื่นไฟฟ้าหัวใจ 2	31
4.1 รูปแสดง Flow Chart	33
5.1 รูปคลื่นไฟฟ้าหัวใจ	38
5.2 รูปตอนเปิดเครื่อง	39
5.3 รูปการกดสวิทช์ที่ 2 หรือ 3	39
5.4 รูปหน้าจอจากการกดสวิทช์ 1 ค้างไว้ 3 วินาที	40
5.5 หน้าจอการตั้งเวลา	40
5.6 รูปหน้าจอของ โหมดแสดงตนบาย	41
5.1 รูปวงจรถัดสัญญาณ common mode	38
5.2 รูปวงจรถัดสัญญาณ differential mode	38
5.3 รูปคลื่นไฟฟ้าหัวใจของสัญญาณช่อง 0	39

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

	หน้า
5.4 รูปคลื่นไฟฟ้าหัวใจของสัญญาณช่อง 1	40
5.5 รูปของสัญญาณเมื่อทำการเอามือแตะสายไฟที่จับอิเล็กทรอนิกส์ โตรด	40
5.6 รูปของสัญญาณเมื่ออิเล็กทรอนิกส์ โตรดที่จับยึดไม่แน่น	41
5.7 รูปตอนเปิดเครื่อง	41
5.8 รูปการกดสวิตซ์ที่ 2 หรือ 3	42
5.9 รูปหน้าจอจากการกดสวิตซ์ 1 ค้างไว้ 3 วินาที	42
5.10 หน้าจอการตั้งเวลา	43
5.11 รูปหน้าจอของ โหมดแสดงนบาย	43
5.12 รูปสัญญาณคลื่นไฟฟ้าหัวใจที่แสดงผลบนคอมพิวเตอร์	44

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความสำคัญและความเป็นมาของโครงการ

การวินิจฉัยโรคในระบบหัวใจและหลอดเลือดใช้คลื่นไฟฟ้าหัวใจเมื่อเกิดอาการเป็นหลัก และการวินิจฉัยอาการของโรคได้เร็วเท่าไรการรักษาจะได้ผลดีและสิ้นเปลืองค่าใช้จ่ายน้อยกว่า แต่เนื่องจากผู้ป่วยโรคหัวใจในระยะเริ่มต้นมักมีอาการไม่ต่อเนื่องทำให้อาจตรวจไม่พบเมื่อไปพบแพทย์ ดังนั้นหากสามารถบันทึกคลื่นไฟฟ้าหัวใจได้ทันทีเมื่อปรากฏอาการ จะทำการวินิจฉัยโรคจะทำได้เร็วขึ้นและมีประโยชน์อย่างมากในการวางแผนรักษา จึงได้ออกแบบและสร้างเครื่องต้นแบบวัดคลื่นไฟฟ้าหัวใจที่สามารถพกติดตามตัวสามารถบันทึกคลื่นไฟฟ้าหัวใจได้ทันทีมีอาการผิดปกติ แล้วนำผลที่ได้ไปให้แพทย์เพื่อวินิจฉัย

1.2 วัตถุประสงค์ของโครงการ

เพื่อศึกษาและพัฒนาเครื่องวัดคลื่นไฟฟ้าหัวใจแบบพกติดตามตัว สามารถใช้ในการวัดคลื่นไฟฟ้าหัวใจสำหรับผู้ป่วยเป็นโรคหัวใจ

1.3 แนวคิดของโครงการ

เครื่องบันทึกคลื่นไฟฟ้าหัวใจที่ใช้วงจรขยายอินสตูเมนเตชัน วงจรขยายสัญญาณและปรับศูนย์อัตโนมัติ และวงจรกำจัดสัญญาณรบกวน รับสัญญาณจากอิเล็กโทรดติดบริเวณหน้าอก แล้วแปลงคลื่นไฟฟ้าหัวใจเป็นสัญญาณดิจิทัลโดยวงจรแปลงอนาล็อกเป็นดิจิทัลโดยใช้ไมโครคอนโทรลเลอร์ 89S8252 ควบคุมเพื่อนำไปบันทึกใน Flash Memory

1.4 โครงสร้างของปริญญานิพนธ์

ปริญญานิพนธ์ได้รวบรวม แนวคิดของการทำโครงการ ทฤษฎีต่าง ๆ ที่เกี่ยวข้อง การทดลองและผลที่ได้ โดยรวบรวมไว้เป็นบทตอนดังนี้

บทที่ 1 บทนำ กล่าวถึงวัตถุประสงค์ ความเป็นมา และขอบเขตของโครงการ

บทที่ 2 พื้นฐานของเครื่องวัดคลื่นไฟฟ้าหัวใจ กล่าวถึงพื้นฐานของเครื่องวัดคลื่นไฟฟ้าหัวใจที่ใช้งานอยู่ทั่วไป

บทที่ 3 การออกแบบ กล่าวถึงการออกแบบในส่วนของวงจรอนาล็อก และวงจรดิจิทัล และหลักการทำงานคร่าวๆของโครงการ

บทที่ 4 หลักการและการออกแบบโปรแกรมบนเครื่อง พีซี กล่าวถึงขั้นตอนการทำงาน โดย Flow chart

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5 การทดลองและผลการทดลอง กล่าวถึงการทดลองทั้งทางด้าน ฮาร์ดแวร์ และ ซอฟต์แวร์ด้วย

บทที่ 6 บทสรุป กล่าวถึงการสรุปผลที่ได้จากการทดลอง รวมถึงการวิเคราะห์ถึงผลที่ได้จากการทดลอง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

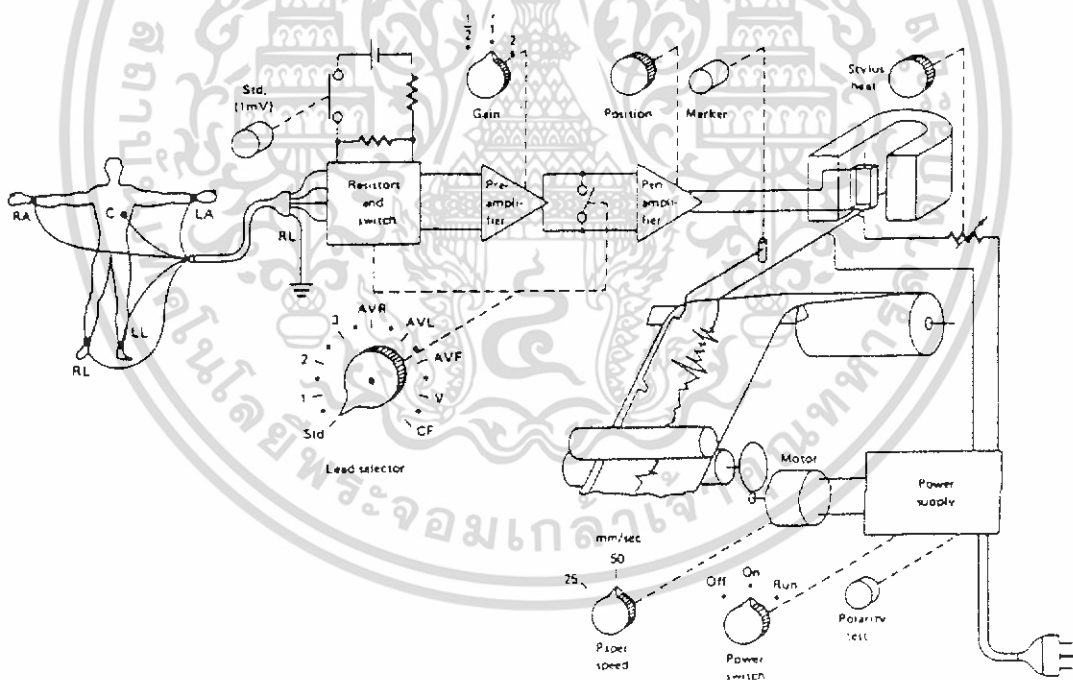
พื้นฐานของเครื่องวัดคลื่นไฟฟ้าหัวใจ

2.1 พื้นฐานของเครื่องวัดคลื่นไฟฟ้าหัวใจ

เครื่องวัดคลื่นไฟฟ้าหัวใจเรียกชื่อย่อว่า อีซีจี มีลักษณะเป็นคิฟเฟอร์เรนเชียล แอมพลิไฟเออร์ที่มีกำลังขยายสูง มีอินพุทอิมพีแดนซ์สูง และใช้บันทึกด้วยการเขียนโดยตรง เพื่อให้สัญญาณที่บันทึกไม่ผิดไปจากความจริง โดยคลื่นไฟฟ้าที่บันทึกนั้นจะมีขนาดอยู่ที่ 100-5000 ไมโครโวลต์ และช่วงความถี่ 0.2 – 100 เฮิรท์ ส่วนอิมพีแดนซ์คือ 0.1- 0.8 เมกะโอห์ม คลื่นไฟฟ้าที่ได้จากหัวใจเป็นผลรวมของไฟฟ้าที่เกิดจากเส้นใยกล้ามเนื้อส่วนใหญ่ และการบันทึกในทางโรงพยาบาลไม่ได้ใช้อิเล็กโทรดวางที่หัวใจโดยตรง หากแต่วางไว้ที่ผิวหนังร่างกาย เช่น ที่แขน ขา หน้าอก เพื่อทำการบันทึกไฟฟ้าที่กระจายมาจากหัวใจ ไปตามเนื้อของร่างกายที่นำไฟฟ้าได้ การที่ต้องวางอิเล็กโทรดบันทึกจากที่หลายแหล่งก็เพื่อที่จะให้ได้คลื่นไฟฟ้าที่มีรูปร่างต่างๆ กันซึ่งแสดงถึงการทำงานของหัวใจที่ส่วนต่างๆ กันด้วย ทั้งนี้ขึ้นอยู่กับเวกเตอร์ทางไฟฟ้าของหัวใจแผนผังอย่างง่าย ๆ ของเครื่อง อี ซี จี แสดงในรูป ที่ 2.1 ในแผนผังแสดงหน้าที่ของปุ่มที่ทำงาน โดยแสดงให้เห็นความสัมพันธ์กับวงจรภายในด้วย เริ่มด้วยมีสายอิเล็กโทรดต่อกับผู้ป่วย 5 สาย สายอิเล็กโทรด เหล่านี้มีรหัสเป็นสี หรือเขียนบอกไว้ว่าจะต่อกับส่วนใดของร่างกาย แล้วต่อกับอิเล็กโทรดชนิดแผ่น สำหรับติดที่แขน และขาซึ่งเรียกว่าสายต่อที่แขนขา (limb lead) 4 สาย ส่วนอีกสายหนึ่งโดยมากใช้ต่อกับอิเล็กโทรดรูปถ้วยที่ใช้ดูดเพื่อใช้วางที่หน้าอก ซึ่งต้องเปลี่ยนที่ได้หลายแห่ง เรียกว่า สายต่อบริเวณหน้าอก ในเครื่อง อี ซี จี ที่ใช้ในทางโรงพยาบาล โดยทั่วไป สายอิเล็กโทรดที่ต่อกับขาข้างขวา ใช้เป็นสายดินคือต่อกับแท่นเครื่อง อี ซี จี ด้วยส่วนอีก 4 สายนั้นต่อเข้ากับวงจรสวิทช์ เลือกสายต่อซึ่งนอกจากมีหน้าที่เลือกสายต่อที่ต้องการจะบันทึกแล้วยังเป็นวงจรรีซีตเตอร์ง่าย ๆ เพื่อช่วยในการบันทึกด้วย สวิทช์เลือกสายต่อสามารถใช้เลือกคู่อิเล็กโทรดที่ใช้บันทึกได้ถึง 12 เกณฑ์ อันเป็นวิธีการบันทึกที่ใช้กันโดยทั่วไปในทางโรงพยาบาล ตำแหน่งแรกของสวิทช์เลือกสายต่อที่แสดงในรูปที่ 2.1. อยู่ที่ STD ในตำแหน่งนี้เมื่อกดปุ่ม STD (1mV) จะทำให้ไฟตรง 1 มิลลิโวลต์ ต่อเข้ากับทางนำเข้าแอมพลิไฟเออร์ อันเป็นการปรับเทียบเครื่อง (calibration) คือเป็นการป้อนไฟฟ้า 1mV เข้าไปขยายใน เครื่องแทนไฟฟ้าจากหัวใจ โดยทั่วไปเครื่องจะต้องมีกำลังขยาย 1 ซม ต่อ 1 มิลลิโวลต์ คือปากกาที่เขียนเคลื่อนที่ไป 1 ซม ในแนวแกนตั้งเมื่อป้อนสัญญาณ 1 มิลลิโวลต์เข้าไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเทียบมาตรฐานนี้ (standardization) ต้องกระทำเมื่อทำการตรวจวัด อี ซี จี คือ สามารถนำเครื่อง อี ซี จี ไป บันทึกสัญญาณอื่นได้ ที่แอมพลิไฟเออร์ของปากกามีปุ่มที่ทำให้กำลังขยายเพิ่มเป็น 2 เท่า ในกรณีสัญญาณที่จะทำการบันทึกมากเกินไปหรือน้อยเกินไป ทั้งนี้เพื่อให้ได้คลื่น อี ซี จี ไม่สูงเกินไปหรือต่ำเกินไป ปริ๊นแอมพลิไฟเออร์เป็นชนิดคิฟเฟอร์เรนเซียลทางสัญญาณนำเข้ามาและมักเป็นไฟสลับด้วย สัญญาณที่ขยายในแอมพลิไฟเออร์แล้วจะส่งไปยังแอมพลิไฟเออร์ของปากกาซึ่งเป็นชนิดไฟตรง ซึ่งสามารถนำเครื่อง อี ซี จี ไปบันทึกสัญญาณอื่นได้ ที่แอมพลิไฟเออร์ของปากกาซึ่งทำหน้าที่ใส่ แรงดันออฟเซต (offset voltage) เพื่อให้ปากกามาอยู่ตรงกลางกระดาษบันทึก ก็คล้ายกับที่ใช้ในกัลป์วานอมิเตอร์ แต่ทำรูปร่างให้ใหญ่โตแข็งแรงกว่าเพื่อให้มีแรงไปทำให้ปากกาเคลื่อนที่ได้เร็ว เครื่อง อี ซี จี สมัยใหม่จะใช้ทรานสดิวเซอร์เพื่อการเปลี่ยนแปลงในตำแหน่งหรือความเร็วของปากกาที่ใช้เขียน แล้วเปลี่ยนสัญญาณให้เป็นศักย์ไฟฟ้าป้อนกลับไปยังแอมพลิไฟเออร์ของปากกา เพื่อช่วยปรับปรุงการสนองต่อความถี่ของเครื่องบันทึก



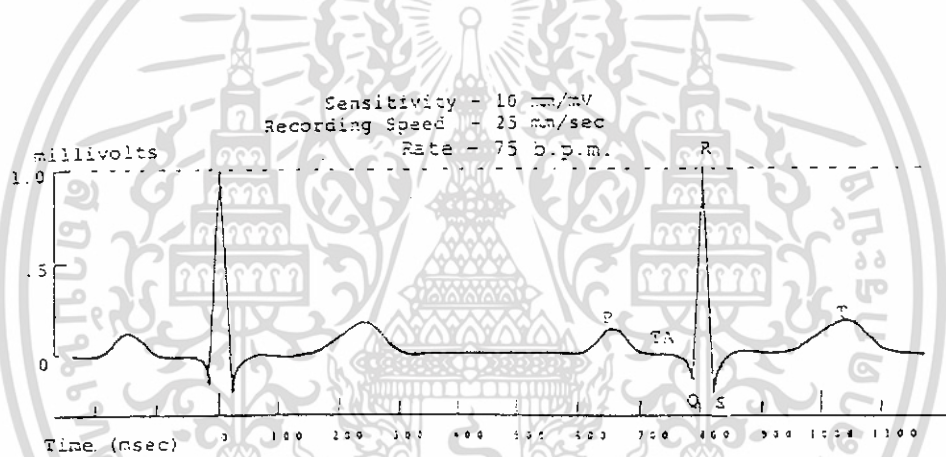
รูปที่ 2.1 ผังของเครื่อง อี.ซี.จี. และส่วนประกอบต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อใช้เข็มร้อน (heated stylus) สำหรับบันทึกปุ่ม หัวเข็มร้อน (stylus heat) ช่วยควบคุมความเข้มและความกว้างของเส้นที่เขียน คือทำให้ร้อนมากหรือน้อยลงนั่นเอง และปุ่มทำเครื่องหมาย นั้นใช้สำหรับทำเครื่องหมายบนกระดาษบันทึก เป็นการแสดงว่าได้ทำการบันทึกจากสายต่อใด และมีสวิตช์เพื่อเลือกความเร็วกระดาษ โดยมากใช้ความเร็ว 25 มิลลิเมตรต่อวินาที ซึ่งเป็นความเร็วมาตรฐานหรืออาจใช้ความเร็ว 50 มิลลิเมตรต่อวินาที ในกรณีที่อัตราการทำงานของโรคหัวใจเร็วมาก

2.2 รูปร่างของสัญญาณคลื่นไฟฟ้าหัวใจ

ลักษณะของสัญญาณคลื่นไฟฟ้าหัวใจที่ปกติ แสดงตัวรูปที่ 2.2. ภาพคลื่นไฟฟ้าหัวใจที่บันทึกได้ จะเริ่มตั้งแต่ก่อนการบีบตัวของหัวใจจนกระทั่งมีการคลายตัวในแต่ละครั้ง ดังนั้นจึงเกิดสัญญาณขึ้นเป็นจังหวะ โดยมีความถี่เท่ากับอัตราการเต้นของหัวใจ



รูปที่ 2.2 สัญญาณไฟฟ้าหัวใจที่เป็นปกติ

ภาพคลื่นไฟฟ้าหัวใจในแต่ละ จังหวะประกอบด้วยคลื่นไฟฟ้าย่อย 3 คลื่น คือ

2.2.1 ช่วงคลื่น P เป็นผลรวมทางไฟฟ้าขบวนการดีโพลาร์ไรซ์ที่เกิดขึ้นที่หัวใจห้องบนทั้งซ้ายและขวา ซึ่งเกิดก่อนที่หัวใจทั้งสองห้องจะมีการบีบตัว

2.2.2 ช่วงคลื่น QRS เป็นผลรวมทางไฟฟ้าจากขบวนการดีโพลาร์ไรซ์ของหัวใจห้องล่างด้านซ้ายและขวา ซึ่งเกิดขึ้นก่อนที่หัวใจทั้งสองข้างจะมีการบีบตัว โดยที่ขนาดของคลื่นสัญญาณ R สำหรับการทำงานปกติของหัวใจมีค่าประมาณ 1 มิลลิโวลต์

2.2.3 ช่วงคลื่น T เป็นผลรวมทางไฟฟ้าจากขบวนการรีโพลาร์ไรซ์ของหัวใจห้องล่างทั้งซ้ายและขวา และเกิดขึ้นก่อนที่หัวใจทั้งสองห้องจะมีการคลายตัว โดยขนาดของสัญญาณ T มีค่า ประมาณ 1/3 ของขนาดของสัญญาณ R

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับขบวนการรีโพลาร์ไรเซชันของหัวใจห้องบน อาจเกิดขึ้นในช่วงระหว่างที่หัวใจห้องล่างมีการบีบตัว แต่ค่าขนาดจะไม่ปรากฏ เนื่องจากค่าของสัญญาณช่วงคลื่น QRS มีค่ามากกว่า

2.3 วิธีการวัด

ในการวัดคลื่นไฟฟ้าหัวใจตามโรงพยาบาลทั่วไปนั้น จะใช้อิเล็กโทรดจับตามจุดต่างๆที่ทำการวัด โดยมีลักษณะการจับตามจุดต่างๆซึ่งเป็นมาตรฐานที่ใช้ทั่วไปดังรูป



รูปที่ 2.3 ตำแหน่งของการวัดคลื่นไฟฟ้าหัวใจ

รูปที่ 2.4 ตำแหน่งการวัดคลื่นไฟฟ้าหัวใจที่หน้าอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบ

การออกแบบวงจรจะแบ่งวงจรถูกออกเป็น 2 ส่วน คือวงจรถอนาล็อก และวงจรถิจิตอล ซึ่งการทำงานของวงจรถือบายได้ดังนี้

3.1 วงจรถอนาล็อก

จะประกอบไปด้วย 3 ส่วนหลัก ๆ ซึ่งมีการทำงานดังนี้

3.1.1 ส่วนที่ 1 เป็นวงจรถายความแตกต่างซึ่งใช้วงจรถายแบบอินสตรูเมนต์เรชัน (Instrumentation) สัญญาณคลื่นไฟฟ้าหัวใจที่เข้ามามีขนาดน้อยมาก เพียงประมาณ 1 มิลลิโวลต์ โดยการรับสัญญาณจาก อิเล็กโทรดที่ติดบนผิวหนังซึ่งมีค่าความต้านทานสูง และจะมีสัญญาณรบกวนจากไฟฟ้าบ้านกระแสสลับความถี่ 50 Hz ปนมาด้วย ทำให้ต้องเลือกใช้วงจรที่เป็น ดิฟเฟอเรนเชียล แอมป์ลิไฟเออร์ (Differential Amplifier) เพื่อลดสัญญาณรบกวนให้ได้มากที่สุด ดังนั้นวงจรถายที่นำมาใช้ต้องมีคุณสมบัติดังนี้

ก มีอินพุทอิมพีแดนซ์สูงมาก เมื่อเทียบกับความต้านทานของผิวหนัง เพื่อป้องกันการเสียนวลของวงจรและการบั่นทอนของสัญญาณที่เข้าสู่อินพุท เพราะจะมีผลเสียต่อวงจรถายคือสัญญาณรบกวนที่เข้ามาในลักษณะคอมมอนโหมด (commonmode signal) ไม่สามารถกำจัดออกไปได้ และยังทำให้เกิดศักดาไฟฟ้าออฟเซต (offset voltage) ซึ่งจะถูกลบออกให้มิลค่ามากขึ้นที่เอาต์พุท ถ้ามีค่ามวงจรถายจะอิมตัวศักดาไฟฟ้าเอาต์พุทจะค้างอยู่ที่ค่านี้อีกเท่าศักดาไฟฟ้าของแหล่งจ่าย และวงจรไม่สามารถทำงานได้

ข มีค่า CMRR (Common Mode Rejection Ratio) สูง ซึ่งเป็นคุณสมบัติของวงจรถายความแตกต่างที่สามารถกำจัดสัญญาณรบกวนได้ คือการมีอัตราขยายของสัญญาณดิฟเฟอเรนเชียลโหมด (differential mode signal) สูง และมีอัตราขยายของสัญญาณคอมมอนโหมดต่ำ เนื่องจากสัญญาณที่ต้องการขยายจะเข้าไปที่อินพุทในลักษณะสัญญาณดิฟเฟอเรนเชียลโหมด ส่วนสัญญาณรบกวนจะเข้าไปที่อินพุทในลักษณะสัญญาณคอมมอนโหมด

3.1.2 ส่วนที่ 2 เป็นวงจรถายสัญญาณและปรับศูนย์อัตโนมัติ เมื่อสัญญาณคลื่นไฟฟ้าหัวใจผ่านวงจรถายความแตกต่าง จะมีศักดาไฟฟ้าออฟเซตขยายเข้ามาด้วยซึ่งเป็นศักดาไฟฟ้ากระแสตรง สามารถแยกออกจากคลื่นไฟฟ้าหัวใจได้ โดยการส่งผ่านสัญญาณทางตัวเก็บประจุ เนื่องจากคลื่นไฟฟ้าหัวใจมีความถี่ต่ำ เพื่อให้สัญญาณผ่านไปได้ ตัวเก็บประจุที่ใช้ต้องมีค่ามาก ปัญหาที่ตามมาคือ ถ้ามีการรบกวนที่อิเล็กโทรดทำให้ความต้านทานตรงรอยสัมผัสมีการเปลี่ยนแปลง จะเกิดศักดาไฟฟ้าออฟเซตที่เอาต์พุทของจอยขยายความแตกต่าง ทำให้คลื่นไฟฟ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวใจที่ผ่านตัวเก็บประจุจะลอยตัวออกจากระดับศูนย์ และกลับเข้าสู่ระดับศูนย์ช้ามาก เนื่องจากค่า time constant ของตัวเก็บประจุกับความต้านทานอินพุทของวงจรถัดไปมีค่ามาก ดังนั้นวงจรนี้จะช่วยลดค่า time constant ให้เหลือน้อยมากเมื่อเกิดศักดาไฟฟ้าออฟเซ็ทถึงระดับที่กำหนดไว้คือ ประมาณ +1.2 โวลต์ หรือ -1.2 โวลต์ ซึ่งเท่ากับศักดาไฟฟ้าที่ตกคร่อมไดโอด D1 หรือ D2 กับขา EB ของทรานซิสเตอร์ Q1 หรือ Q2 ทรานซิสเตอร์จะทำหน้าที่เป็นทางผ่านของสัญญาณลงกราวด์ ทำให้ค่า time constant มีค่าลดลงเหลือเพียงเท่ากับผลคูณของ C1 กับความต้านทานภายในของ ทรานซิสเตอร์ซึ่งมีค่าน้อยมาก

3.2 วงจรดิจิตอล

จะใช้ตัวไมโครคอนโทรลเลอร์ตระกูล MCS-51 เบอร์ AT89C2051 และ AT89S8252 ในการควบคุมการทำงานต่างๆทั้งหมดของวงจร ซึ่งมีการทำงานคร่าวๆดังนี้

3.2.1 เริ่มจากสัญญาณที่รับมาจากวงจรถอดจะเข้ามาที่วงจรแปลงอนาลอกเป็นดิจิตอล ซึ่งใช้ตัว A/D converter เบอร์ LTC1298 ในการแปลงสัญญาณจากสัญญาณอนาลอก ให้เป็นสัญญาณดิจิตอล โดยจะใช้การติดต่อผ่านส่วนเชื่อมต่ออุปกรณ์อนุกรม (SPI) ซึ่งมีอยู่ในตัวไมโครคอนโทรลเลอร์เบอร์ AT89S8252

3.2.2 สัญญาณที่เป็นดิจิตอลที่เข้ามาในตัวไมโครคอนโทรลเลอร์นั้น จะถูกส่งไปบันทึกไว้ที่ตัวแฟลชเมมโมรี่ (Flash Memory) โดยที่ตัวไมโครคอนโทรลเลอร์เป็นตัวติดต่อและควบคุมการทำงานทั้งหมด

3.2.3 สัญญาณจะนำมาแสดงบนเครื่องคอมพิวเตอร์โดยผ่านทางพอร์ตยูเอสบี

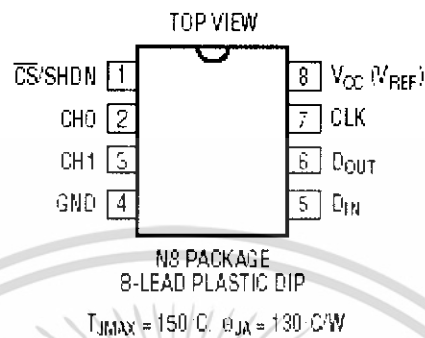
3.3 วงจรแปลงสัญญาณอนาล็อกเป็นดิจิตอล

จะใช้ไอซีเบอร์ LTC1298 ซึ่งเป็นไอซีที่ทำงานแบบ 12 บิต ทำงานแบบ Successive approximation sampling A/D converters เพื่อจะส่งสัญญาณไปยังตัวไมโครคอนโทรลเลอร์ มีรายละเอียดดังนี้

- เป็นไอซีขนาด 8 ขา
- กินกระแสต่ำ 250 ไมโครแอมป์
- ใช้เวลาในการแปลงสัญญาณ 1 ครั้ง ที่ 60 ไมโครเซค
- แรงดันการทำงานอยู่ที่ 4.5 – 5.5 โวลต์
- ความถี่คัลลอกอยู่ที่ 200 กิโลเฮิรตซ์
- กระแสชดดาว์นอัตโนมัติ 1 นาโนแอมป์
- มีวงจรมัลติเพล็กซ์ 2 ช่อง ทางอินพุท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.1 รูปแบบของไอซี



รูปที่ 3.1 รูปแบบของไอซีเบอร์ LTC1298

จากรูป ขาต่างๆจะทำงานดังนี้

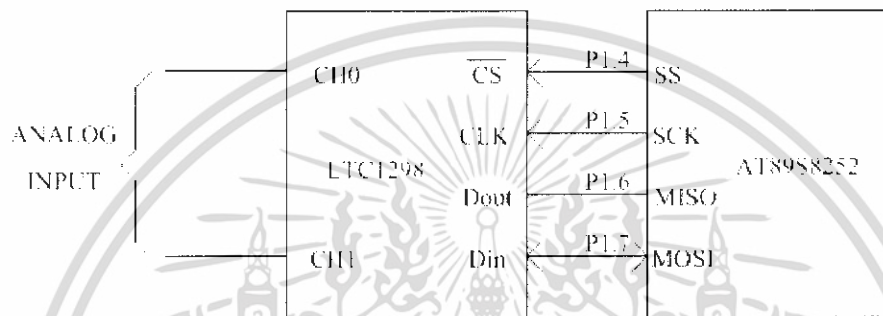
- $\overline{CS}/SHDN$ (ขา 1) เป็นขาที่บอกว่าอินพุตทำงานหรือไม่ ถ้าทำจะแสดงลอจิกต่ำ
- (0) ถ้าไม่ทำจะแสดงลอจิกสูง (1)
- CH0 (ขา 2) ใช้ต่อกับวงจรมัลติเพล็กซ์
- CH1 (ขา 3) ใช้ต่อกับวงจรมัลติเพล็กซ์
- GND (ขา 4) ใช้ต่อกับกราวด์ของวงจร
- D_{IN} (ขา 5) ใช้รับสัญญาณดิจิตอล โดยจะเข้าที่วงจรมัลติเพล็กซ์เซอร์
- D_{OUT} (pin 6) จะส่งผลการแปลงจากสัญญาณอนาล็อก เป็นดิจิตอล ไปยังไมโครคอนโทรลเลอร์
- CLK (pin 7) เป็นขาสัญญาณที่กำหนดจังหวะการทำงาน
- V_{CC}/V_{REF} (pin 8) ใช้ป้อนไฟให้กับตัวไอซี

3.3.2 การต่อเข้ากับไมโครคอนโทรลเลอร์

ในการต่อตัว LTC1298 เข้ากับไมโครคอนโทรลเลอร์นั้น จะต่อเข้ากับส่วนเชื่อมต่ออุปกรณ์อนุกรมหรือ SPI ซึ่งลักษณะการต่อมีดังนี้ขา

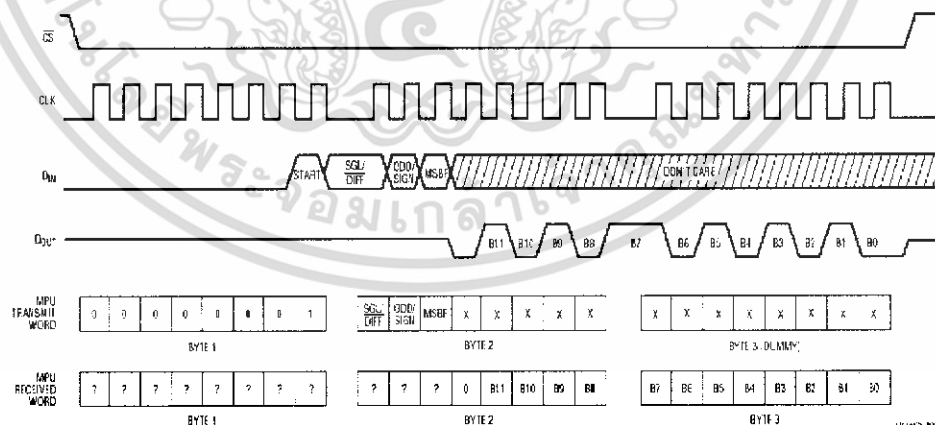
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ขา \overline{CS} เช้ากับขา P1.4 หรือขา SS ของ SPI
- ขา CLK เช้ากับขา P1.7 หรือขา SCK ของ SPI
- ขา D_{OUT} เช้ากับขา P1.6 หรือขา MISO ของ SPI
- ขา D_{IN} เช้ากับขา P1.5 หรือขา MOSI ของ SPI



รูปที่ 3.2 การเชื่อมต่อ LTC1298 เช้ากับส่วนของ SPI

3.3.3 ไทม์มิ่ง ไดอะแกรม (Timing Diagram)



รูปที่ 3.3 ไทม์มิ่ง ไดอะแกรมของไอซีเบอร์ LTC1298

ในการทำงานนั้นจะรับสัญญาณอนาล็อกเข้ามาทาง ขาCH0 และขา CH1 แล้วทำการแปลงสัญญาณเป็นสัญญาณดิจิทัลเพื่อส่งไปยังตัวไมโครคอนโทรลเลอร์ที่ขา Dout และจะเริ่มทำการแปลงสัญญาณที่ขอบขาของสัญญาณ \overline{CS} โดยจะเริ่มจากการสตาร์ตบิต (Start bit) และก็จะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตเห็นไปใช้โดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

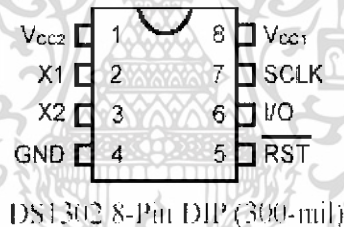
ทำการในกระบวนการของวงจรมัลติเพล็กซ์เซอร์ (Multiplexer) แล้วจึงเริ่มทำการแปลงสัญญาณแล้วส่งสัญญาณไปแบบ 12 บิตเมื่อส่งสัญญาณไปครบแล้วจะทำการเคลียร์บิตเพื่อรอรับสัญญาณครั้งต่อไป

3.4 Real Time Clock

ใช้ไอซีเบอร์ DS1302 ในการนับ เวลา , วัน , เดือน และปี ซึ่งใช้ต่อเข้ากับไมโครคอนโทรลเลอร์ โดยผ่านทางพอร์ตอนุกรม

3.4.1 คุณสมบัติของ DS1302

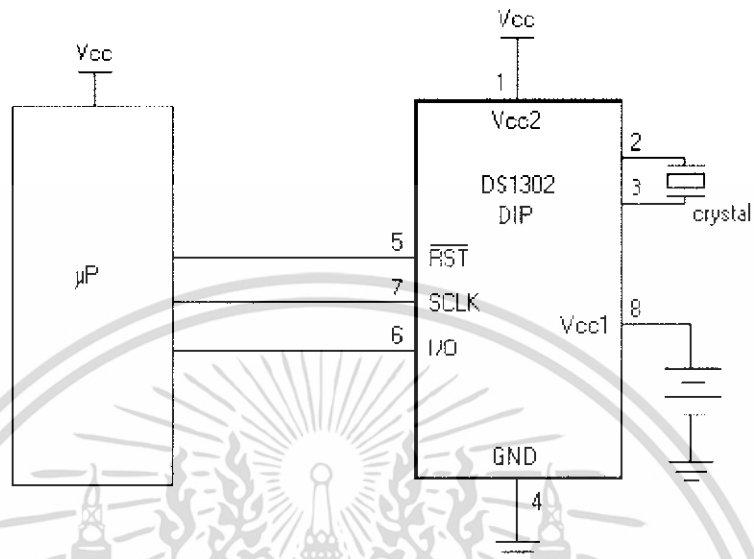
- ทำงานที่แรงดัน 2.0 – 5.5 โวลต์
- ใช้กระแสทำงาน 200 นาโนแอมป์ ที่ 2.0 โวลต์
- จะกินกำลังในการทำงานต่ำที่ 1 ไมโครวัตต์
- ทำงานที่ย่านอุณหภูมิ -45 องศาเซลเซียส ถึง +80 องศาเซลเซียส



รูปที่ 3.4 แบบของไอซีเบอร์ DS1302

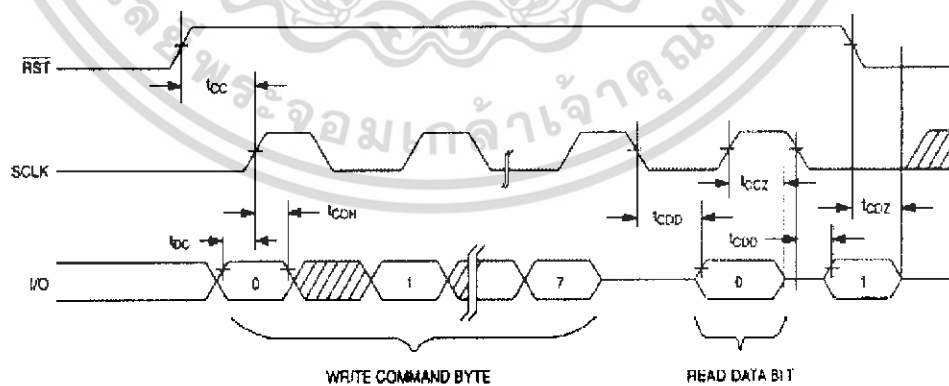
ไอซีเบอร์ DS1302 จะมีลักษณะการทำงานเหมือนแรม (RAM) โดยทำงานแบบ 31 ไบต์ ไบต์ละ 8 บิต

3.4.2 การต่อใช้งานเข้ากับไมโครคอนโทรลเลอร์



รูปที่ 3.5 การต่อเข้ากับไมโครคอนโทรลเลอร์

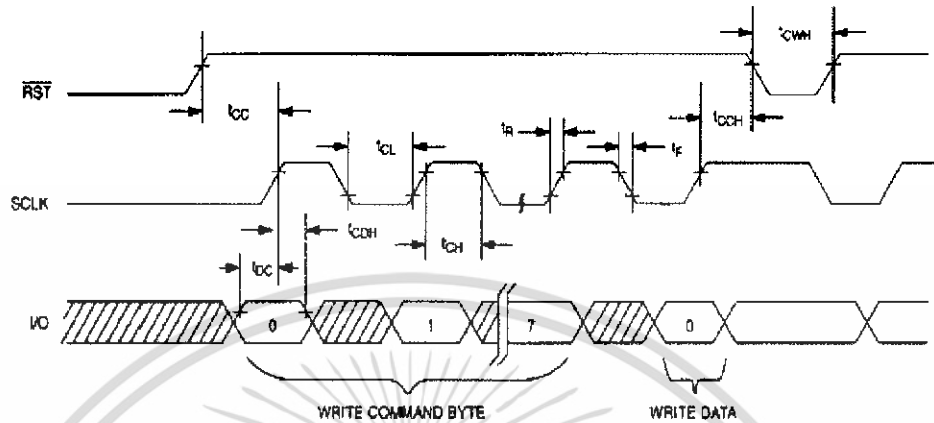
3.4.3 ไทม์มิ่งไดอะแกรม - เวลาอ่านข้อมูล



รูปที่ 3.6 ไทม์มิ่งไดอะแกรมเวลาอ่านข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เวลาเขียนข้อมูล



รูปที่ 3.7 ไทม์มิ่งไดอะแกรมเวลาเขียนข้อมูล

3.5 ยูเอสบี (Universal Serial Bus:USB)

ยูเอสบี เป็นพอร์ตที่ใช้รับส่งข้อมูลอนุกรมแบบอนุกรมแบบอนุกรมแบบอนุกรมในการเชื่อมต่อกับคอมพิวเตอร์ มีคุณสมบัติดังนี้

- คอนเน็กเตอร์เพียงตัวเดียว ซึ่งมี 2 รูปแบบ สำหรับเชื่อมต่ออุปกรณ์ทุกๆชนิดเข้ากับคอมพิวเตอร์
- สามารถเชื่อมต่ออุปกรณ์หลายๆ ชนิดรวมเข้าสู่คอนเน็กเตอร์ตัวเดียว สูงสุด 127 ตัว
- ไม่เกิดการขัดแย้งกันของการเข้าใช้ทรัพยากรของระบบ (IRQ)
- ตรวจสอบการเชื่อมต่อและตั้งค่าการทำงานต่างๆอัตโนมัติ ระหว่างที่เครื่องกำลังทำงานอยู่ (hot attachment)

-ที่ขาพอร์ต ยูเอสบี มีแรงดันไฟตรง 5+ โวลต์จ่ายออกมาด้วย ทำให้อุปกรณ์ต่อพ่วงที่ใช้ พลังงาน ไม่มากนัก สามารถใช้แรงดันจากพอร์ตยูเอสบีนี้เป็นไฟเลี้ยงเพื่อทำงานได้ โดยไม่ต้องอาศัยแหล่งจ่ายไฟเลี้ยงจากภายนอกเพิ่มเติม

3.5.1 การเชื่อมต่อ

โครงสร้างการเชื่อมต่อของยูเอสบี นั้นเป็นสตาร์ (STAR) โดยฮับทำหน้าที่เป็นศูนย์กลางจุดเชื่อมต่อไปยังอุปกรณ์ต่าง ๆ ในแต่ละระดับชั้น สายเชื่อมต่อแต่ละเส้นเป็นการต่อแบบจุดต่อจุด (point to point)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจาก “พอร์ตยูเอสบี มีรูปแบบการเชื่อมต่อเป็นระบบบัส แต่มีโครงสร้างการเชื่อมต่อแบบสตาร์” ซึ่งอาจสร้างความสับสนในจุดนี้ได้ จึงขอขยายความหมายและเปรียบเทียบให้เห็นข้อแตกต่าง ตรงจุดนี้ จากคำว่า “โครงสร้างการเชื่อมต่อ” จะอธิบายลักษณะการต่อสายส่งสัญญาณของอุปกรณ์แต่ละตัวเข้าด้วยกัน ซึ่งก็คือ ลักษณะการต่อสายไฟระหว่างกันทำอย่างไรนั่นเอง แต่คำว่า “รูปแบบการเชื่อมต่อ” นั้นจะอธิบายถึงการเดินทางของสัญญาณข้อมูลภายในสายไฟที่เชื่อม ต่อกันอยู่จากอุปกรณ์ตัวหนึ่งไปยังอุปกรณ์แต่ละตัวที่ต่อเชื่อมอยู่” ความหมายโดยรวมก็คือ พอร์ต USB ต่อสายเข้าด้วยกัน แบบสตาร์รับส่งข้อมูลแบบบัส (โครงสร้างคล้ายกับระบบแลนอีเทอร์เน็ต)



รูปที่ 3.8 การเชื่อมต่อของยูเอสบี

3.5.2 กระบวนการกำหนดการทำงานของอุปกรณ์

เมื่อเริ่มการทำงานของยูเอสบี หรือเมื่อมีการเชื่อมต่ออุปกรณ์ตัวใหม่เข้ามาในระบบ ลำดับขั้นตอนคร่าวๆ ในการทำงานจะเป็นดังนี้

3.5.2.1 ฮับตรวจสอบพบว่ามี การเชื่อมต่ออุปกรณ์ตัวใหม่เข้าสู่ระบบ แล้วแจ้งผลกลับกลับไปยังโฮสต์คอนโทรลเลอร์

3.5.2.2 โฮสต์คอนโทรลเลอร์สั่งให้ฮับเปิดการทำงานของแหล่งจ่ายไฟในโหมดประหยัด เพื่อให้อุปกรณ์ที่อาศัยพลังงานจากบัสสามารถทำงานได้

3.5.2.3 โฮสต์คอนโทรลเลอร์สั่งให้ฮับรีเซตพอร์ตที่อุปกรณ์ตัวใหม่มาเชื่อมต่อ เพื่อให้ตัวอุปกรณ์รีเซตค่าแอดเดรสและเอ็นด์พอยต์ของตัวเองให้เป็นค่าเริ่มต้น (default)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีกรนำไปใช้

3.5.2.4 โสสต์อ่านดิสคริปเตอร์ต่างๆ จากตัวอุปกรณ์และพิจารณาว่าทรัพยากร ว่าทรัพยากร ของระบบพอเพียงต่อความต้องการของตัวอุปกรณ์หรือไม่ ทรัพยากรในที่นี้คือ พลังงานไฟฟ้าและปริมาณข้อมูลที่จะส่งของตัวอุปกรณ์ หากพิจารณาแล้วว่าไม่สามารถทำงานได้ก็ จะสั่งให้ฮับปิดการทำงานของพอร์ตนั้น

3.5.2.5 เมื่อโสสต์พิจารณาแล้วว่าสามารถให้บริการแก่อุปกรณ์ตัวที่มาเชื่อมต่อ ได้ จะควบคุมให้แหล่งจ่ายไฟจ่ายพลังงานตามที่อุปกรณ์ต้องการ รวมไปถึงการตั้งค่าแอดเดรสและ กำหนดค่าต่างๆ

3.5.2.6 หลังจากตั้งค่าเรียบร้อยแล้ว คอมพิวเตอร์ก็จะรู้จักกับอุปกรณ์ตัวใหม่และ สามารถติดต่อกับอุปกรณ์ได้ทันทีโดยไม่ต้องปิดและเปิดคอมพิวเตอร์ใหม่

3.5.3 ชนิดของการถ่ายทอดสัญญาณของยูเอสบี

3.5.3.1 การถ่ายทอดสัญญาณแบบไอโซโครนัส (Isochronous transfer)

จะมีการส่งหรือรับข้อมูลทุกๆเฟรมข้อมูล (1 มิลลิวินาที) จึงต้องมีการจอง การรับ ส่งข้อมูลไว้ตลอดทุกเฟรม แต่หากเกิดความผิดพลาดของข้อมูลจะไม่มีพยายามส่ง ข้อมูลที่ผิดนั้นกลับไปใหม่ จึงทำการถ่ายทอดข้อมูลแบบนี้จะไม่มีการรับประกันความถูกต้องของ ข้อมูลก่อนเริ่มการทำงาน อุปกรณ์จะต้องทำการซิงโครไนซ์การทำงานของตัวเองเข้ากับสัญญาณ เริ่มต้นของแต่ละเฟรม (SOP : Start Of Frame) ของยูเอสบี เพื่อให้อุปกรณ์สามารถรับหรือส่ง ข้อมูลในทุกๆ 1 มิลลิวินาทีได้ทันที ทำให้เกิดการรับส่งข้อมูลในจังหวะที่ถูกต้อง

3.5.3.2 การถ่ายทอดสัญญาณแบบบัลค์ (Bulk transfer)

ใช้กับอุปกรณ์ที่ไม่ต้องการส่งข้อมูลแบบคงที่ เช่น เครื่องพิมพ์ การรับส่งจะ เกิดขึ้นก็ต่อเมื่อบัสว่างจากการส่งข้อมูลอื่นๆทั้งหมดแล้ว หรืออาจกล่าวได้ว่าการถ่ายทอดสัญญาณ แบบบัลค์จะเกิดขึ้นได้ก็ต่อเมื่อเหลือที่ว่างของข้อมูลในแต่ละเฟรมเท่านั้น หากไม่มีที่ว่างเหลือก็จะ ต้องรอไปเรื่อยๆ จนกว่าจะสามารถส่งได้ แต่การถ่ายทอดสัญญาณแบบบัลค์ก็ไม่สามารถส่งข้อมูล ขนาดใหญ่ไปในคราวเดียวได้ เพราะเสี่ยงต่อการเกิดความผิดพลาดของข้อมูลระหว่างส่ง ข้อมูลจะ ต้องถูกแบ่งออกเป็นส่วนๆ แล้วทยอยส่งต่อกันไปในเฟรมเดียวกัน โดยขนาดของชิ้นข้อมูลที่ใหญ่ ที่สุดหลังจากแบ่งแล้วจะเท่ากับ 8,16,32 หรือ 64 ไบต์

3.5.3.3 การถ่ายทอดสัญญาณแบบอินเตอร์รัปต์ (Interrupt transfer)

การรับส่งข้อมูลแบบนี้จะเรียนแบบการสร้างสัญญาณอินเตอร์รัปต์ ให้แก่ ระบบ โดยวิธีการโพลลิง และการอ่านข้อมูลจะต้องไม่ช้าและเร็วจนเกินไป โดยที่คาบเวลาใน การอ่าน จะขึ้นอยู่กับความต้องการของอุปกรณ์แต่ละชนิด การส่งข้อมูลแต่ละครั้งสามารถ กำหนดให้ส่งได้มากที่สุด 64 ไบต์ ซึ่งเป็นการอ่านที่คงที่คล้ายกับการส่งแบบไอโซโครนัส การ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ่ายถอดสัญญาณแบบนี้จึงต้องมีการตรวจสอบความถูกต้องของข้อมูลใหม่ในรอบการวนอ่านถัดไป หากเกิดความผิดพลาดในการส่ง ก็สามารถส่งใหม่ได้

3.5.3.4 การถ่ายถอดสัญญาณแบบควบคุม (Control transfer)

ใช้สำหรับส่งคำสั่งควบคุมการทำงานทั้งหมดของอุปกรณ์ทุกๆ ตัว ตั้งแต่เริ่มแรกเมื่ออุปกรณ์ถูกเชื่อมต่อเข้ากับระบบ เกิดการอ่านข้อมูลดิสทริบิวเตอร์ต่างๆ เมื่อได้ข้อมูลมาโฮสต์ก็จะพิจารณาว่าสามารถรับรองการทำงานได้หรือไม่ เมื่อพิจารณาว่าได้ ก็จะตั้งค่าการทำงานต่างๆ และเริ่มการทำงาน การถ่ายถอดสัญญาณควบคุมประกอบด้วย การส่งข้อมูล 3 ช่วง ช่วงที่หนึ่งคือ ช่วงเตรียมตัวตั้งค่า (setup stage) ช่วงที่ 2 คือ ช่วงข้อมูล (data stage) และช่วงสุดท้ายคือ ช่วงตรวจสอบ (handshake stage) ยูเอสบี จะสำรองแบนด์วิดท์การส่งข้อมูลของบัสไว้ 10 เปอร์เซ็นต์ เพื่อให้สำหรับการถ่ายถอดสัญญาณควบคุมโดยเฉพาะ เพื่อไม่ให้เกิดปัญหาการควบคุมอุปกรณ์ถ้าช้าเมื่อมีการรับส่งข้อมูลปริมาณมากๆ เพราะอาจจะถูกอุปกรณ์บางตัวดึงแบนด์วิดท์ไปใช้รับส่งข้อมูลจนไม่สามารถส่งสัญญาณควบคุมได้

3.6 แฟลชเมมโมรี่ (FLASH MEMORY AM29F016B)

3.6.1 คุณสมบัติเฉพาะตัว

- ทำงานโดยใช้แหล่งจ่ายไฟเดี่ยว 5 V ซึ่งเป็นความต้องการอย่างน้อยที่สุดของระบบ
- ผลิตด้วยเทคโนโลยี 0.32 ไมโครเมตร
- ประสิทธิภาพการทำงานสูงสุด ใช้เวลาเข้าถึงข้อมูล 70 นาโนเซค
- กินกำลังงานต่ำ
 - ใช้กระแสขณะอ่าน 25 มิลลิแอมป์
 - ใช้กระแสขณะโปรแกรมหรือลบ 30 มิลลิแอมป์
 - ใช้กระแสขณะสแตนด์บาย (standby) 1 ไมโครแอมป์
- โครงสร้างของเซกเตอร์ (sector) สามารถยืดหยุ่นได้
- มีอัลกอริทึมที่ฝังมากับตัวชิป
- สามารถโปรแกรมหรือลบได้อย่างน้อยที่สุด 1,000,000 รอบต่อเซกเตอร์
- สามารถเก็บรักษาข้อมูลได้นานถึง 20 ปี ที่อุณหภูมิ 125 องศาเซลเซียส
- มีหลายแพ็คเกจให้เลือก
 - แบบ TSOP 48ขา และ 40 ขา
 - แบบ SO 44ขา
- ใช้ได้กับมาตรฐาน JEDEC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สามารถตรวจสอบความสมบูรณ์ของการ โปรแกรมหรือการลบทางซอฟต์แวร์ได้
- สามารถตรวจสอบความสมบูรณ์ของการ โปรแกรมหรือการลบทางฮาร์ดแวร์ได้
- สามารถพักขบวนการลบแล้วกลับมาทำการลบใหม่ได้
- สามารถใช้ขารี่เซ็ท (#RESET) ภายในเพื่อไปที่ โหมคอ่าน

3.6.2 ลักษณะโดยทั่วไป

Am29F016B เป็นแฟลชเมมโมรี่ขนาด 16 เม็กกะบิต,และ 2,097,152 ไบต์ ใช้ไฟ 5 โวลต์เท่านั้น ข้อมูลขนาด 8 บิตจะปรากฏที่ DQ0 – DQ7 Am29F016B ที่นำเสนอนี้มีทั้ง 48 ขาและ 40ขา แบบ TSOP และ 44 ขา แบบ SO อุปกรณ์ตัวนี้ถูกออกแบบเพื่อระบบการ โปรแกรมภายใน กับระบบมาตรฐานแบบแหล่งจ่ายไฟ(Vcc) 5 โวลต์ ขบวนการ โปรแกรมหรือการลบไม่ต้องการ แหล่งจ่ายไฟขนาด 12 โวลต์ และอุปกรณ์สามารถโปรแกรมได้ด้วยเครื่องโปรแกรมอีพรอม (EPROM) แบบมาตรฐาน

อุปกรณ์นี้ผลิตขึ้นโดยใช้เทคโนโลยี 0.32 ไมโครเมตร ของ AMD การนำเสนอ คุณสมบัติและประโยชน์ทั้งหมดของ Am29F016B ซึ่งเป็นการผลิตโดยใช้เทคโนโลยีขบวนการ 0.5 ไมโครเมตร

อุปกรณ์มาตรฐานที่นำเสนอใช้เวลาในการเข้าถึงข้อมูล ,120 ,90 ,70 และ 150นาโนเซค จึงยอม ให้ตัวประมวลความเร็วสูงทำงานโดยไม่ต้องคอย กำจัดการแย่งบัส โดยอุปกรณ์มีการแยก ชิพอีนเบิล(CE#), ไรท์อีนเบิล(WE#) และ เอาท์พุทอีนเบิล (OE#)

อุปกรณ์ต้องการแหล่งจ่ายไฟ 5 โวลต์เท่านั้นสำหรับการอ่านและเขียน ผลิตและควบคุมแรงดันโดยภายในเพื่อจัดเตรียมสำหรับขบวนการ โปรแกรมและลบ

ชุดคำสั่งทั้งหมดตรงกับมาตรฐานของ JEDEC single power – supply Flash คำสั่งใดๆ จะถูกเขียนไปที่รีจิสเตอร์คำสั่ง โดยใช้เวลาในการเขียนของไมโครโปรเซสเซอร์แบบมาตรฐาน รายการที่อยู่ในรีจิสเตอร์จะใช้เป็นอินพุทของสเตจแมชชีน (state machine) ที่ควบคุมการลบและการ โปรแกรมในช่วงของการเขียนภายในจะเก็บที่อยู่และข้อมูลที่จำเป็นสำหรับการ โปรแกรมและการลบด้วย การอ่านข้อมูลออกจากอุปกรณ์จะเหมือนกับการอ่านข้อมูลจากแฟลช หรืออีพรอม

กลไกการ โปรแกรมเกิดขึ้นโดยการ ทำคำสั่งของ โปรแกรมตามลำดับ เริ่มต้นด้วย ขั้นตอนการ โปรแกรมที่ถูกฝังมาภายใน ขั้นตอนภายในที่เกิดขึ้น โดยอัตโนมัติกำหนดด้วยความกว้างของพัลส์โปรแกรมและตรวจสอบความถูกต้องเกี่ยวกับขอบของพัลส์

กลไกการลบเกิดขึ้นโดยการทำการลบตามลำดับคำสั่ง เริ่มต้นด้วยขั้นตอนการลบที่ฝังภายใน เป็นขั้นตอนที่เกิดขึ้นโดยอัตโนมัติก่อนขบวนการลบ ระหว่างการลบ ,กลไกโดยอัตโนมัติกำหนดด้วยความกว้างของพัลส์ในการลบ และตรวจสอบเกี่ยวกับขอบของพัลส์

ระบบแม่สามารถตรวจสอบว่าขบวนการ โปรแกรมหรือการลบสมบูรณ์หรือไม่โดยสังเกตที่ขา RY/BY# หรือโดยการอ่าน สถานะของบิต ที่ DQ7 (Data# Polling) และ DQ6 (toggle) หลังจากรอบของการ โปรแกรมและการลบจบแล้ว อุปกรณ์พร้อมที่จะอ่านข้อมูล หรือยอมรับคำสั่งอื่น

โครงสร้างของการลบเซกเตอร์ ยอมให้ลบเซกเตอร์ และ โปรแกรมใหม่ โดยไม่มีผลกับข้อมูลที่บรรจุอยู่ในเซกเตอร์อื่นๆ อุปกรณ์ถูกลบมาแล้วขณะที่ส่งมาจากโรงงาน

ตารางที่ 1 ตารางแนะนำการเลือกผลิตภัณฑ์

Family Part Number	Am29F016B				
Speed Options	(V _{CC} = 5.0 V ± 5%)	-75			
	(V _{CC} = 5.0 V ± 10%)		-90	-120	-150
Max Access Time (ns)	70	90	120	150	
CE# Access (ns)	70	90	120	150	
OE# Access (ns)	40	40	50	75	

Note: See the AC Characteristics section for more information.



รูปที่ 3.9 รูปแบบของไอซีเบอร์ AM29F016B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6.3 การป้องกันข้อมูลทางฮาร์ดแวร์

กระบวนการวัดใดๆ รวมทั้งตัวตรวจวัดระดับแรงดัน (Vcc) จะยับยั้งขบวนการเขียนโดยอัตโนมัติ ระหว่าง การผ่านกำลังงาน ลักษณะเฉพาะของการป้องกันเซกเตอร์ ทางฮาร์ดแวร์ คือ การดีสเอเบิล (disable) ขบวนการโปรแกรมและการลบในเซกเตอร์ ใดๆที่รวมกัน ของหน่วยความจำ สามารถทำได้โดยผ่านทางโปรแกรม

ลักษณะเฉพาะของการพักการลบ คือ การที่ผู้ใช้สามารถหยุดการลบไว้ระหว่างคาบเวลาใดๆ เพื่อไปอ่านข้อมูลหรือโปรแกรมข้อมูลจากเซกเตอร์ ใดๆที่ไม่ถูกเลือกสำหรับการลบ การลบขั้นพื้นฐานสามารถที่จะทำให้สำเร็จได้เช่นนี้

ขา RESET# จะทำให้การทำงานใดๆ ที่ดำเนินอยู่จบสิ้นลง และรีเซ็ต (reset) สภาพภายในเพื่อทำการเขียนข้อมูล ขา RESET# อาจจะมีสัมพันธ์กับการรีเซ็ต ระบบทางไฟฟ้า การรีเซ็ตระบบจะทำได้ด้วยการรีเซ็ตทุกไคร้ครั้งนี้ ทำให้ระบบไมโครโปรเซสเซอร์สามารถอ่านโปรแกรมบูทอัพ (boot-up) จากแฟลชเมมโมรี่

ระบบสามารถพักการทำงานด้วยโหมดสแตนด์บาย (standby mode) ลดการกินกำลังไฟได้ดียิ่งในโหมดนี้

เทคโนโลยีแฟลช ของ AMD ได้รวบรวมเอาประสบการณ์การสร้างแฟลชเมมโมรี่หลายปีเพื่อที่จะผลิตให้มีคุณภาพ และ ความน่าเชื่อถือ สูง อุปกรณ์จะลบบิต ภายในเซกเตอร์ทั้งหมดในเวลาเดียวกันด้วยไฟฟ้าโดยผ่านทาง Fowler – Nordheim tunneling ข้อมูลจะถูกโปรแกรมด้วยการพ่นอิเล็กตรอนที่ร้อน

3.6.4 การทำงานของบัส

ในส่วนนี้จะอธิบายถึงความต้องการต่างๆ และการใช้งานการทำงานของบัส ซึ่งเริ่มต้นด้วยรีจิสเตอร์คำสั่งภายใน รีจิสเตอร์คำสั่งจะไม่กินเนื้อที่ในหน่วยจำที่ใช้แอสแอดเดรส โดยรีจิสเตอร์จะประกอบด้วยเลขที่ที่ใช้เก็บคำสั่งต่างๆ พร้อมด้วยแอดเดรสและข้อมูลที่ต้องการให้ทำงาน สิ่งที่บรรจุอยู่ในรีจิสเตอร์จะใช้เป็นอินพุทของสเตจเมกซ์ทรีซึน ภายใน และเอาท์พุทจะกำหนดตามฟังก์ชันของอุปกรณ์ ตารางการทำงานของบัสจะมีรายการของอินพุท และความต้องการของระดับความคุมต่างๆ และผลของเอาท์พุท ส่วนย่อยดังต่อไปนี้นี้จะอธิบายถึงแต่ละการทำงานในรายละเอียดต่อไป

ตารางที่ 2 ตารางการทำงานของบัส

Operation	CE#	OE#	WE#	RESET#	A0–A20	DQ0–DQ7
Read	L	L	H	H	A _{IN}	D _{OUT}
Write	L	H	L	H	A _{IN}	D _{IN}
CMOS Standby	V _{CC} ± 0.5 V	X	X	V _{CC} ± 0.5 V	X	High-Z
TTL Standby	H	X	X	H	X	High-Z
Output Disable	L	H	H	H	X	High-Z
Hardware Reset	X	X	X	L	X	High-Z
Temporary Sector Unprotect (See Note)	X	X	X	V _{ID}	A _{IN}	D _{IN}

Legend:

L = Logic Low = V_{IL}, H = Logic High = V_{IH}, V_{ID} = 12.0 ± 0.5 V, X = Don't Care, D_{IN} = Data In, D_{OUT} = Data Out, A_{IN} = Address In

Note: See the sections on Sector Group Protection and Temporary Sector Unprotect for more information

3.6.4.1 ข้อกำหนดสำหรับการอ่านข้อมูลแบบอาร์เรย์

การอ่านข้อมูลอาร์เรย์จากเอาต์พุตใดๆ ระบบต้องทำให้ขา CE# และ OE# เป็น V_{IL} ขา CE# ใช้เลือกตัวอุปกรณ์ ขา OE# เป็นขาที่ใช้ควบคุมเอาต์พุตและข้อมูลไปที่ขาเอาต์พุต WE# ยังคงอยู่ที่ V_{IH}

แสดงเมซ์ท์ขึ้น ภายในจะเซ็ทเพื่อการอ่านข้อมูลอาร์เรย์ในขณะที่อุปกรณ์เริ่มทำงานหรือหลังจากเกิดการรีเซ็ท รอบของการอ่านของไมโครโปรเซสเซอร์แบบมาตรฐานที่อ้างแอดเดรสอย่างถูกต้องเหมาะสมที่อื่นพุทจะทำให้เกิดข้อมูลที่ถูกต้องที่เอาต์พุตอุปกรณ์ยังคงสามารถอ่านจนกระทั่งรายการที่อยู่ในรีจิสเตอร์คำสั่งเปลี่ยน

3.6.4.2 การเขียนคำสั่งและลำดับของคำสั่ง

เมื่อเขียนคำสั่งหรือลำดับคำสั่ง (ซึ่งรวมถึงการโปรแกรมข้อมูลไปที่อุปกรณ์และการลบเช็คเตอร์ของหน่วยความจำ) ระบบต้องทำให้ WE# และ CE# เป็น V_{IL} และ OE# เป็น V_{IH}

การลบครั้งหนึ่งสามารถลบได้หนึ่งเช็คเตอร์, หลายเช็คเตอร์หรือลบทั้งหมด ตารางเช็คเตอร์แอดเดรสแสดงเนื้อที่แต่ละเช็คเตอร์ครอบครอง เช็คเตอร์แอดเดรสหนึ่งประกอบด้วยบิตหลายบิตเพื่อที่จะเลือกเช็คเตอร์หนึ่งโดยเฉพาะ

หลังจากระบบทำการเขียนลำดับคำสั่งถูกเลือกแบบอัตโนมัติ อุปกรณ์เข้าสู่โหมดซีลโหมด ระบบสามารถอ่านโค้ดโดยอัตโนมัติจากรีจิสเตอร์ภายใน (ซึ่งแยกจากหน่วยความจำอาร์เรย์) บน DQ0-DQ7 รอบของเวลาการอ่านแบบมาตรฐานใช้ในโหมดนี้

3.6.4.3 สถานะของการโปรแกรมและการลบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระหว่างการลบหรือการโปรแกรมครั้งหนึ่ง ระบบอาจจะตรวจสอบสถานะของการทำงานโดยการอ่านสถานะของบิตที่ DQ7 – DQ0

3.6.5 นิยามของคำสั่ง

การเขียนแอดเดรสแบบเจาะจงและข้อมูลหรือลำดับคำสั่งเข้าไปที่รีจิสเตอร์คำสั่งเป็นการเริ่มการทำงานของอุปกรณ์ ตารางของคำสั่ง (Command Definitions) กำหนดลำดับคำสั่งของรีจิสเตอร์ให้ถูกต้อง เหมาะสม การเขียนแอดเดรสและค่าของข้อมูลที่ไม่ถูกต้อง หรือการเขียนลำดับที่ไม่ถูกต้องจะเป็นการรีเซ็ตอุปกรณ์ไปสู่การอ่านข้อมูลอาร์เรย์

แอดเดรสทั้งหมดจะถูกแลทซ์เก็บไว้ในขณะขอบขาของ WE# หรือ CE# อันไหนก็ตามที่เกิดทีหลัง ข้อมูลทั้งหมดจะถูกแลทซ์เก็บไว้ในขณะขอบขาขึ้นของ WE# หรือ CE# อันไหนก็ตามที่เกิดก่อน

3.6.5.1 การอ่านข้อมูลอาร์เรย์

อุปกรณ์จะรีเซ็ตโดยอัตโนมัติเพื่อไปอ่านข้อมูลหลังจากที่อุปกรณ์ทำงาน ไม่มีคำสั่งใดๆ ที่จะกู้ข้อมูลกลับมาได้ อุปกรณ์พร้อมที่จะอ่านข้อมูลหลังจากขบวนการโปรแกรมหรือการลบที่ฝังตัวมากับชิพทำงานเสร็จสมบูรณ์

หลังจากอุปกรณ์ยอมรับคำสั่งพักการลบ อุปกรณ์จะเข้าสู่โหมดพักการลบ ระบบสามารถ อ่านข้อมูลอาร์เรย์ ยกเว้นถ้าอ่านที่แอดเดรสภายใน เซกเตอร์ใดๆที่พักการลบ หลังจากขบวนการโปรแกรมในโหมดพักการลบเสร็จสมบูรณ์แล้ว ระบบอาจจะอ่านข้อมูลอีกครั้ง

ระบบต้องออกคำสั่งรีเซ็ตเพื่อทำให้อุปกรณ์สามารถกลับมาอ่านข้อมูล ถ้า DQ5 เป็นลอจิกสูง (High) หรือ ขณะที่อยู่ในโหมดคอคิวต์ซีล

3.6.5.2 คำสั่งรีเซ็ต

การเขียนคำสั่งรีเซ็ตเพื่อรีเซ็ตอุปกรณ์ เพื่อให้อุปกรณ์อ่านข้อมูล แอดเดรสบิตจะไม่สนใจสำหรับคำสั่งนี้

คำสั่งรีเซ็ตอาจเขียนระหว่างลำดับของรอบในลำดับคำสั่งการลบก่อนการลบ จะเริ่มขึ้น การรีเซ็ตอุปกรณ์นี้เพื่ออ่านข้อมูลอาร์เรย์ การลบครั้งหนึ่งเริ่มต้นขึ้น อย่างไรก็ตาม อุปกรณ์จะเพิกเฉยต่อคำสั่งรีเซ็ตจนกระทั่งกระบวนการทำงานจะเสร็จสมบูรณ์

คำสั่งรีเซ็ตอาจเขียนระหว่างลำดับของรอบในลำดับคำสั่งของการ โปรแกรมก่อนการโปรแกรมเริ่ม การรีเซ็ตอุปกรณ์นี้เพื่ออ่านข้อมูล (ใช้การโปรแกรมในโหมดพักการลบอีกด้วย) การโปรแกรมครั้งหนึ่ง อย่างไรก็ตาม อุปกรณ์จะเพิกเฉยต่อคำสั่งรีเซ็ตจนกระทั่งกระบวนการทำงานเสร็จสมบูรณ์

ถ้า DQS เป็นลจิกสูง ระหว่างกระบวนการ การโปรแกรมหรือการลบ การเขียนคำสั่งรีเซ็ตจะทำให้อุปกรณ์กลับไปอ่านข้อมูล (ใช้ระหว่างการพักการลบ)

3.6.5.3 ลำดับคำสั่งการเลือกอัตโนมัติ

ลำดับคำสั่งการเลือกอัตโนมัติ ยอมให้ระบบเข้าถึงการทำงานและโค้ดของอุปกรณ์และกำหนดว่าจะ ป้องกันเซ็คเตอร์หรือไม่ ตารางคำสั่ง แสดงแอดเดรสและข้อมูลที่จำเป็น

ตารางที่ 3 ตารางคำสั่งของ Flash Memory

Command Sequence (Note 1)	Cycles	Bus Cycles (Notes 2-4)											
		First		Second		Third		Fourth		Fifth		Sixth	
		Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data	Addr	Data
Read (Note 5)	1	RA	RD										
Reset (Note 6)	1	XXX	F0										
Autoselect (Note 7)	Manufacturer ID	4	555	AA	2AA	55	555	90	X00	01			
	Device ID	4	555	AA	2AA	55	555	90	X01	AD			
	Sector Group Protect Verify (Note 8)	4	555	AA	2AA	55	555	90	SGA X02	XX00 XX01			
Program	4	555	AA	2AA	55	555	A0	PA	PD				
Chip Erase	6	555	AA	2AA	55	555	80	555	AA	2AA	55	555	10
Sector Erase	6	555	AA	2AA	55	555	80	555	AA	2AA	55	SA	30
Erase Suspend (Note 9)	1	XXX	B0										
Erase Resume (Note 10)	1	XXX	30										

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4 ตารางแอดเดรสของเซกเตอร์

Sector	A20	A19	A18	A17	A16	Address Range
SA0	0	0	0	0	0	00000h-00FFFFh
SA1	0	0	0	0	1	01000h-01FFFFh
SA2	0	0	0	1	0	02000h-02FFFFh
SA3	0	0	0	1	1	03000h-03FFFFh
SA4	0	0	1	0	0	04000h-04FFFFh
SA5	0	0	1	0	1	05000h-05FFFFh
SA6	0	0	1	1	0	06000h-06FFFFh
SA7	0	0	1	1	1	07000h-07FFFFh
SA8	0	1	0	0	0	08000h-08FFFFh
SA9	0	1	0	0	1	09000h-09FFFFh
SA10	0	1	0	1	0	0A000h-0AFFFFh
SA11	0	1	0	1	1	0B000h-0BFFFFh
SA12	0	1	1	0	0	0C000h-0CFFFFh
SA13	0	1	1	0	1	0D000h-0DFFFFh
SA14	0	1	1	1	0	0E000h-0EFFFFh
SA15	0	1	1	1	1	0F000h-0FFFFh
SA16	1	0	0	0	0	10000h-10FFFFh
SA17	1	0	0	0	1	11000h-11FFFFh
SA18	1	0	0	1	0	12000h-12FFFFh
SA19	1	0	0	1	1	13000h-13FFFFh
SA20	1	0	1	0	0	14000h-14FFFFh
SA21	1	0	1	0	1	15000h-15FFFFh
SA22	1	0	1	1	0	16000h-16FFFFh
SA23	1	0	1	1	1	17000h-17FFFFh
SA24	1	1	0	0	0	18000h-18FFFFh
SA25	1	1	0	0	1	19000h-19FFFFh
SA26	1	1	0	1	0	1A000h-1AFFFFh
SA27	1	1	0	1	1	1B000h-1BFFFFh
SA28	1	1	1	0	0	1C000h-1CFFFFh
SA29	1	1	1	0	1	1D000h-1DFFFFh
SA30	1	1	1	1	0	1E000h-1EFFFFh
SA31	1	1	1	1	1	1F000h-1FFFFh

Note: All sectors are 64 Kbytes in size.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7 แอล ซี ดี (LCD)

3.7.1 ความรู้ทั่วไปเกี่ยวกับ แอล ซี ดี

แอล ซี ดี เป็นจอแสดงผลที่สามารถแสดงข้อความที่เป็นตัวเลข ,ตัวอักษรและสัญลักษณ์อื่นๆ ซึ่งคงจะเคยพบการใช้งานอยู่บ้างในเครื่องมือวัด เลเซอร์ปริ้นเตอร์และงานทางด้านอิเล็กทรอนิกส์อื่นๆ รูปที่ 1 เป็นตัวอย่างจอ แอล ซี ดี เพื่อใช้งานแสดงผลข้อความ สาเหตุที่มีการนำแอล ซี ดี ไปใช้งานกันมากเนื่องจาก ความสะดวกความสมบูรณ์ของข้อความ

คอนโทรลเลอร์ HD44780 เป็นคอนโทรลเลอร์แบบยัดติดผิวหน้าขนาด 80 ขาจากฮิตาชิ ทำหน้าที่ควบคุมการทำงานต่าง ๆ ให้จอ แอล ซี ดี โดยมีสัญญาณติดต่อกับภายนอกเพียง 14 เส้น

แผงของ แอล ซี ดี จะประกอบด้วยเซกเมนต์แสดงผลขนาดเล็กจำนวนมากในเซกเมนต์จะบรรจุ ชั้นของเหลวเป็นแผ่นบางๆ อยู่ระหว่างชั้นของแก้ว ของเหลวนี้เป็นสารประกอบทำงานโดยอาศัยพลังงานไฟฟ้ามาควบคุมการทำงานหรือการแสดงผลของ แอล ซี ดี เกิดขึ้นเนื่องจากการควบคุมแรงดันตกคร่อมเซกเมนต์ ก็จะทำให้เกิดสีดำหรือทึบแสง แต่ถ้าเอาแรงดันนั้นออก เซกเมนต์นั้นก็สว่างหรือโปร่งแสง ด้วยวิธีการจ่ายแรงดันและกระจายแรงดันนี้ก็เพียงพอที่จะควบคุมการแสดงตัวเลขตัวอักษร และสัญลักษณ์ต่างๆ ได้ และจากสาเหตุที่จอ แอล ซี ดี ใช้แรงดันควบคุม ดังนั้นจึงกินกำลังงานต่ำและขนาดเล็กแบบราบ

โมดูลของ แอล ซี ดี บางรุ่นอาจจะมี 1 แถว หรือมากกว่า การแสดงผลของจอ แอล ซี ดี จะอยู่ในรูปเมตริกซ์ เช่นบางรุ่นแสดงเมตริกซ์ที่มีขนาดกว้าง 5 เซกเมนต์ สูง 8 เซกเมนต์ และสำหรับรุ่น HD44780 สามารถควบคุมการแสดงผลได้สูงถึง 11 เซกเมนต์ ซึ่งเป็นผลดีกับการแสดงตัวอักษรบางตัวเช่น g, p และ q

ตัวอักษรจะถูกสร้างโดยการปรับตำแหน่งแต่ละเซกเมนต์ให้เหมาะสม เช่น ตัวอักษร "L" จะสร้างจากแนวตั้ง 1 แถว และแนวนอน 1 แถว

ตารางที่ 1 แสดงขาสัญญาณต่างๆ ที่ใช้ในการเชื่อมต่อกับไมโครคอนโทรลเลอร์ เนื่องจาก ว่าการควบคุม แอล ซี ดี ต้องการเวลาเพื่อรอทำงานตามคำสั่งหรือรับสัญญาณดังนั้นถ้าใช้กับคอมพิวเตอร์ จะต้องพิจารณาเรื่องเวลาด้วย แต่การเชื่อมต่อกับไมโครคอนโทรลเลอร์หรือไมโครโปรเซสเซอร์สามารถต่อโดยตรงได้ไม่ต้องมีอุปกรณ์อื่นมาต่อเพิ่ม หรือถ้าที่ก็เพียงเล็กน้อยเท่านั้น

ตารางที่ 5 ตารางแสดงตำแหน่งขาต่างๆ ที่ใช้เชื่อมต่อกับ LCD

ขา	สัญลักษณ์	ฟังก์ชัน
1	V_{ss}	กราวด์
2	V_{dd}	+5 โวลต์
3	V_o	ปรับความสว่างด้วยแรงดัน
4	RS	เลือกรีจิสเตอร์
5	R/\bar{W}	เลือกการอ่านหรือเขียน
6	E	อินเเบิลการอ่านหรือเขียน
7	D_0	คาต้าอินพุต/เอาต์พุตบิตต่ำสุด
8	D_1	คาต้าอินพุต/เอาต์พุตบิตที่ 2
9	D_2	คาต้าอินพุต/เอาต์พุตบิตที่ 3
10	D_3	คาต้าอินพุต/เอาต์พุตบิตที่ 4
11	D_4	คาต้าอินพุต/เอาต์พุตบิตที่ 5
12	D_5	คาต้าอินพุต/เอาต์พุตบิตที่ 6
13	D_6	คาต้าอินพุต/เอาต์พุตบิตที่ 7
14	D_7	คาต้าอินพุต/เอาต์พุตบิตสูงสุด

3.7.2 ไฟเลี้ยงสำหรับ แอล ซี ดี

จอแสดงผล แอล ซี ดี จะใช้ไฟเลี้ยง 5+ โวลต์ ป้อนให้ที่ขา 2 ซึ่งตัวมันกินกระแสเพียงไม่กี่มิลลิแอมป์ ส่วนขา 3 ต่อเพื่อปรับมุมมองการแสดงผลให้เหมาะสม ทั้งนี้ก็ขึ้นอยู่กับผลของแสงในขณะ นั้นด้วยรวมไปถึงตำแหน่งการติดตั้งและอุณหภูมิ

เมื่อเปรียบเทียบคุณสมบัติระหว่าง แอล อี ดี (LED) กับ แอล ซี ดี จะพบว่าในที่ที่มีแสงสว่างค่อนข้างสูง แอล อี ดี เกือบจะมองไม่เห็น ส่วน แอล ซี ดี นั้นสามารถอ่านในที่ที่มีแสงสว่างได้ เนื่องจากการทำงานของ แอล อี ดี นั้นจะปล่อยพลังงานแสงออกมาส่วน แอล ซี ดี นั้นจะใช้การหักเหแสง โดยให้แสงส่งผ่านตัวมัน ซึ่งบางสภาวะในที่ที่มีแสงสว่างน้อยก็ไม่สามารถอ่านค่าจอแสดงผล แอล ซี ดี ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7.3 คอนโทรลเลอร์และการควบคุม

การที่จะใช้โมดูลของ แอล ซี ดี ในงานใดงานหนึ่งนั้น จะต้องทำความเข้าใจกับตัวควบคุมก่อน HD44780 เป็นตัวควบคุมขนาดเล็กที่คล้ายกับคอมพิวเตอร์โดยจะทำงานทั้งหมด 11 คำสั่งแสดงดังในตารางที่ 2 เพื่อควบคุมการทำงานต่าง ๆ เช่นเคลียร์หน้าจอแสดงผล ,เขียนตัวอักษร ,เลือกตำแหน่งที่จะแสดงผลและอ่านข้อมูลจากจอแสดงผล

หน่วยความจำภายใน HD44780 มี 2 ชนิด คือ Character Generator (CG) ROM และ Character – Generator (CG) RAM

CGROM ใช้สำหรับเก็บตัวอักษรเกือบ 200 รูปแบบ เช่นตัวอักษรภาษาอังกฤษ , ตัวเลขเครื่องทางคณิตศาสตร์ ,สัญลักษณ์พิเศษ และอักษรญี่ปุ่น ซึ่งจะกำหนดเองในรอมไว้แล้วไม่สามารถแก้ไขหรือเปลี่ยนแปลงได้

CGRAM ใช้เก็บตัวอักษรที่ผู้ใช้ออกแบบขึ้นเองเช่น โลโก้ ,สัญลักษณ์พิเศษอักษรกราฟิกต่างๆ ที่สามารถออกแบบบนเมตริกซ์ขนาด 5 x 8 ได้ อักษรที่เขียนขึ้นนี้จะเขียนครั้งละ 5 บิตหลายๆ คำ แต่ละคำจะแทนรูปแบบเซกเมนต์ 1 แถว แล้วเก็บไว้ใน CGRAM รูปแบบอักษรนี้จะหายไปเมื่อปิดเครื่อง และเมื่อจะใช้งานต้องเรียกข้อมูลมาใหม่ หลังจากเปิดเครื่อง

อักษรใน CGROM และ CGRAM เป็นอักษรขนาด 8 บิต ซึ่งบางตำแหน่งก็ไม่ได้ใช้ ตำแหน่งแอดเดรสที่ใช้กันมากจากช่วง 21 H ถึง 7 DH ซึ่งตรงกับตำแหน่งรหัสแอสกีบนคอมพิวเตอร์ เช่น “A” จะถูกเก็บไว้ที่ตำแหน่ง 41 H และ “B” จะถูกเก็บไว้ที่ตำแหน่ง 42 H เป็นต้น โดยตำแหน่งแอดเดรสจะถูกเก็บอยู่ในเลขฐาน 16

ไอซี HD44780 มีรีจิสเตอร์ 2 ตัวคือ รีจิสเตอร์คำสั่ง (IR) ซึ่งใช้สำหรับเก็บรหัสของคำสั่ง และรีจิสเตอร์ข้อมูล (DR) ซึ่งใช้สำหรับเก็บรหัสอักษรเมื่อต้องการเขียนหรืออ่านข้อมูลจาก ไอซีจะต้องเลือกรีจิสเตอร์ให้เหมาะสมเพื่อนำไปต่อกับขา 4 ของ แอล ซี ดี ตามหน้าที่การทำงานที่เราต้องการ

ก) เป็นการเขียนข้อมูลลงบน โมดูล แอล ซี ดี

ข) เป็นการอ่านข้อมูลจากโมดูล แอล ซี ดี

หน่วยความจำแรมเก็บข้อมูลแสดงผล Display Data (DD) RAM จะเก็บรหัสตัวอักษรขนาด 8 บิตได้มากกว่า 80 ตัวอักษร ไว้ในแอดเดรสแต่ละตำแหน่ง โดยรหัสของตัวอักษรที่เก็บไว้ใน DDRAM จะกำหนดว่าให้แสดงอักษรที่ตำแหน่งไหน

ในกรณีที่ แอล ซี ดี มีสองแถว ถ้าต้องการลบแถวแรก (การทำงานของมันจะทำการเลื่อน

ตำแหน่งซ้ายสุดของแถวบนซึ่งเป็นตำแหน่งแอดเดรสสูงก่อน และตำแหน่งต่อไปก็จะเลื่อนตามมาตามลำดับ จนกว่าจะหมดแถวที่ 1 แล้วแถวที่ 2 จะตามมา เช่นสมมติว่าแถวที่ 1 มีตำแหน่งแอดเดรสเริ่มที่ 39-0 H พอหมดแถวที่ 1 แล้วแถวที่ 2 จะถูกลบต่อไปเป็น 40 H 41 H 42H 43H และต่อๆ ไปจนครบ

แต่ละครั้งที่ทำการเขียนอักษรลงในแอดเดรสของ DDRAM แต่ละตำแหน่งแอดเดรสจะเพิ่มตำแหน่งขึ้น โดยอัตโนมัติตามลำดับของการเขียนอักษร ยกเว้นกรณีที่ผู้ใช้กำหนดตำแหน่งของแอดเดรสที่จะเขียนข้อมูลเองในบางครั้ง

อย่างไรก็ตามเนื่องจากว่าแถวที่ 2 เริ่มที่ตำแหน่งแอดเดรส 40 H ดังนั้นถ้าหากผู้ใช้ต้องการให้แสดงที่แถวที่ 2 จะต้องอ้างแอดเดรส ให้ถูกต้องด้วยเช่นถ้าจอแสดงผลมี 16 ตำแหน่งแถวที่ 1 จะสิ้นสุดที่ตำแหน่ง 0 FH และแถวที่ 2 จะเริ่มต้นที่ 40 H ดังนั้นถ้าต้องการย้ายอักษรจากตำแหน่งขวาสุดของแถวที่ 1 ไปยังตำแหน่งซ้ายสุดของแถวที่ 2 ผู้ใช้จะต้องอ้างตำแหน่งแอดเดรสไปที่ 40 H

นอกจากนี้จอแสดงผลบางรุ่น จะมีแถวแสดงอักษรเป็นแบบฟิสิกอล (Physical) 1 แถวและแบบลอจิก อีก 2 แถว ซึ่งจะคล้ายกับจอแสดงผลแบบ 2 แถว ในกรณีที่เป็นจอแสดงผลแบบ 16 ตัวอักษร 8 ตัวอักษร แรกจะเริ่มที่แอดเดรส 0 ถึง 7 และ 8 ตัวอักษรหลังจะเริ่มที่แอดเดรส 40 H ถึง 47 H ดังนั้นถ้าจะเขียนข้อมูลไปที่ตำแหน่ง 8 ตัวหลังต้องกำหนดตำแหน่งไปที่ 40 H

อย่างไรก็ตามในจอแสดงผลขนาดเล็กจะไม่มี DDRAM เนื่องจากว่า DDRAM เป็นแรมที่ใช้สำรองไว้ให้ผู้ใช้เลือกใช้ตามวัตถุประสงค์

3.7.4 การอ่านและการเขียน

ขั้นตอนของการเขียนเริ่มจากมีสัญญาณ RS เข้ามาและให้สัญญาณ R/\bar{W} มีสถานะเป็นลอจิกต่ำ (LOW) หลังจากนั้นประมาณ 140 นาโนวินาที สัญญาณอีนาเบิลจะมีสถานะเป็นลอจิกสูง และคงสถานะอยู่อย่างน้อย 450 นาโนวินาทีเพื่อที่จะให้ขาเอาต์ D0-D7 ส่งข้อมูลอย่างน้อย 195 นาโนวินาที ก่อนที่สัญญาณอีนาเบิลจะเป็น LOW

ส่วนขั้นตอนในการอ่านข้อมูลจะคล้ายกับเขียนแต่สัญญาณ R/\bar{W} จะเป็นลอจิกสูง ส่วนสัญญาณข้อมูล D0-D7 จะทำงานหลังจากสัญญาณอีนาเบิลเป็นลอจิกสูง แล้วประมาณ 320 นาโนวินาที

ไอซี HD44780 จะไม่ทำคำสั่งใหม่ที่เข้ามาจนกว่าจะทำคำสั่งที่ทำงานอยู่ขณะเสร็จก่อน ซึ่งในรอบแยกที่ 1 จะแสดงเวลาที่มากที่สุดที่แต่ละคำสั่งใช้ในการประมวลผล แต่ถ้าใช้ภาษาเบสิกหรือภาษาระดับสูงในการโปรแกรมค่าเวลาเหล่านี้อาจไม่ต้องใส่ใจกับมันมากนัก เพราะว่าโปรแกรมจะเข้าถึงคำสั่งโดยอัตโนมัติอยู่แล้วถ้าหากต้องการใช้ชุดโมดูล แอล ซี ดี รับคำสั่งต่อมา

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อผู้จัดทำเห็นไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อาจทำได้โดยเขียนโปรแกรมหน่วงเวลาจากทำคำสั่งเหล่านั้นหรืออาจจะอ่านแฟลกว่า) บิต (7 ก็ได้
คำสั่งในตารางที่ 2 แสดงการเชื่อมต่อของข้อมูลขนาด 8 บิต D0-D7 แต่ถ้าเป็น
คอนโทรลเลอร์ HD44780 จะใช้ข้อมูลขนาด 4 บิตเท่านั้น บางครั้งถ้าเรามีไมโครคอนโทรลเลอร์
ขนาด 8 บิต เราสามารถที่จะต่อใช้ข้อมูลเพียง 4 บิตก็ได้ เพื่อเป็นการประหยัดฮาร์ดแวร์ การใช้
ข้อมูลทั้งหมดเพียง 7 เส้น เท่านั้น (D4-D7, RS, R/\bar{W} และอีน่าเบิล (หรืออาจจะใช้เพียง 6 เส้น ถ้า
ให้สัญญาณ R/\bar{W} เป็นลอจิกต่ำตลอด ก็จะสามารถอ่านข้อมูล D0-D7 แสดงบนจอแสดงผลได้
อย่างไรก็ตามการที่จะส่งคำสั่งเพื่อใช้กับข้อมูลขนาด 8 บิตนั้น คุณจะต้องส่งข้อมูล
บิต D4-D7 ไปพร้อมกับสัญญาณ RS และ R/\bar{W} โดยไม่ต้องบิต D0-D3

ตัวอย่างการเขียนตัวอักษร "Z" ให้ออกจอแสดงผลที่ใช้บิตข้อมูลขนาด 8 บิตจะ
ใช้คำสั่งดังนี้คือ

RS	R/\bar{W}	D7	D6	D5	D4	D3	D2	D1	D0
1	0	0	1	0	1	1	0	1	0

ส่วนกรณีที่ใช้แบบ 4 บิตจะใช้คำสั่งเหมือนกันแต่จะต้องป้อนออกไป 2 ครั้งดังนี้คือ

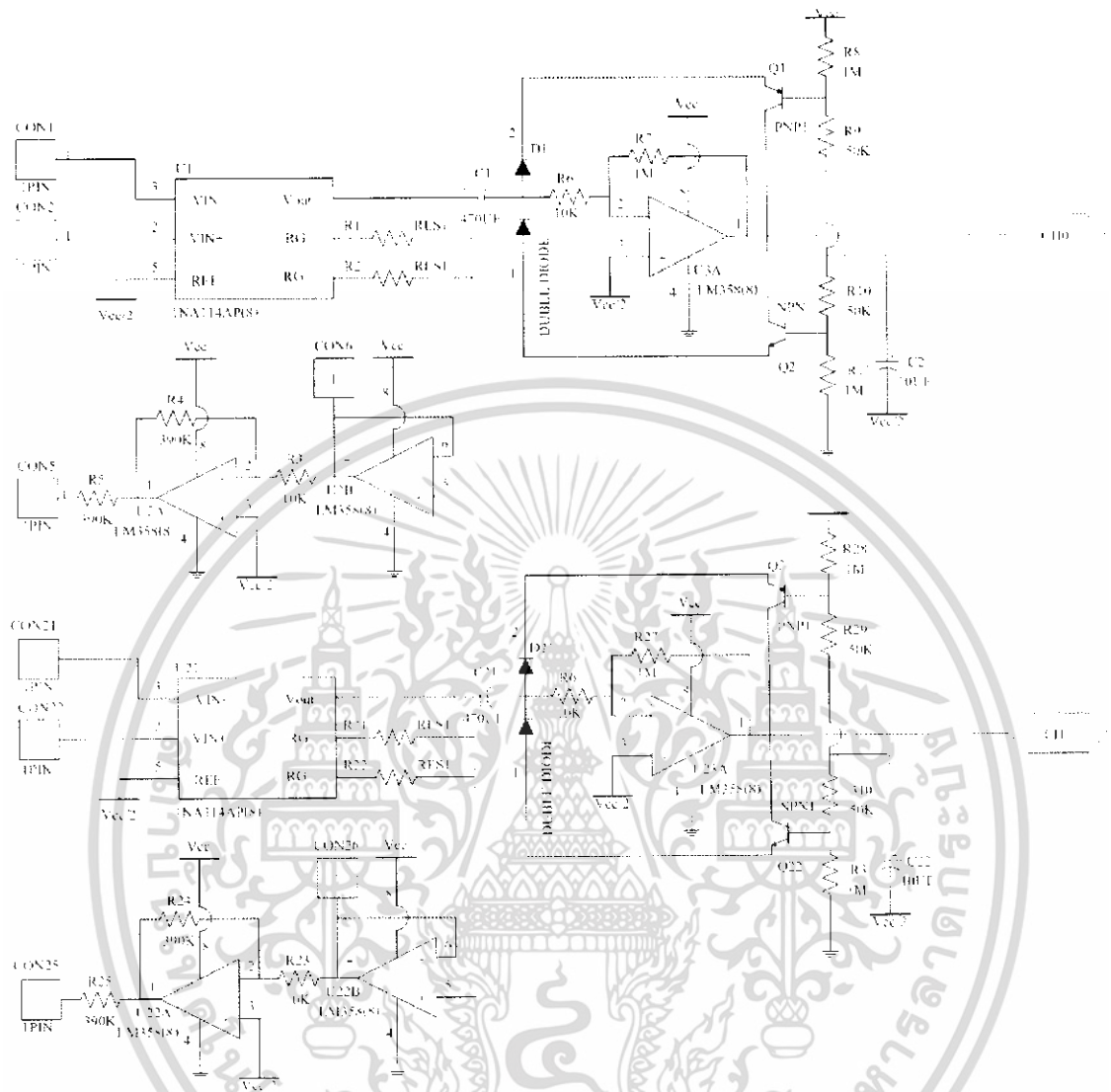
RS	R/\bar{W}	D7	D6	D5	D4
1	0	0	1	0	1
1	0	1	0	1	0

การใช้งานแบบ 4 บิต จะทำงานช้าและโปรแกรมยาวกว่าแบบ 8 บิต แต่ข้อดีก็คือใช้
การต่อวงจร หรือขาสัญญาณน้อยกว่าแบบ 8 บิต

3.7.5 การเชื่อมต่อ

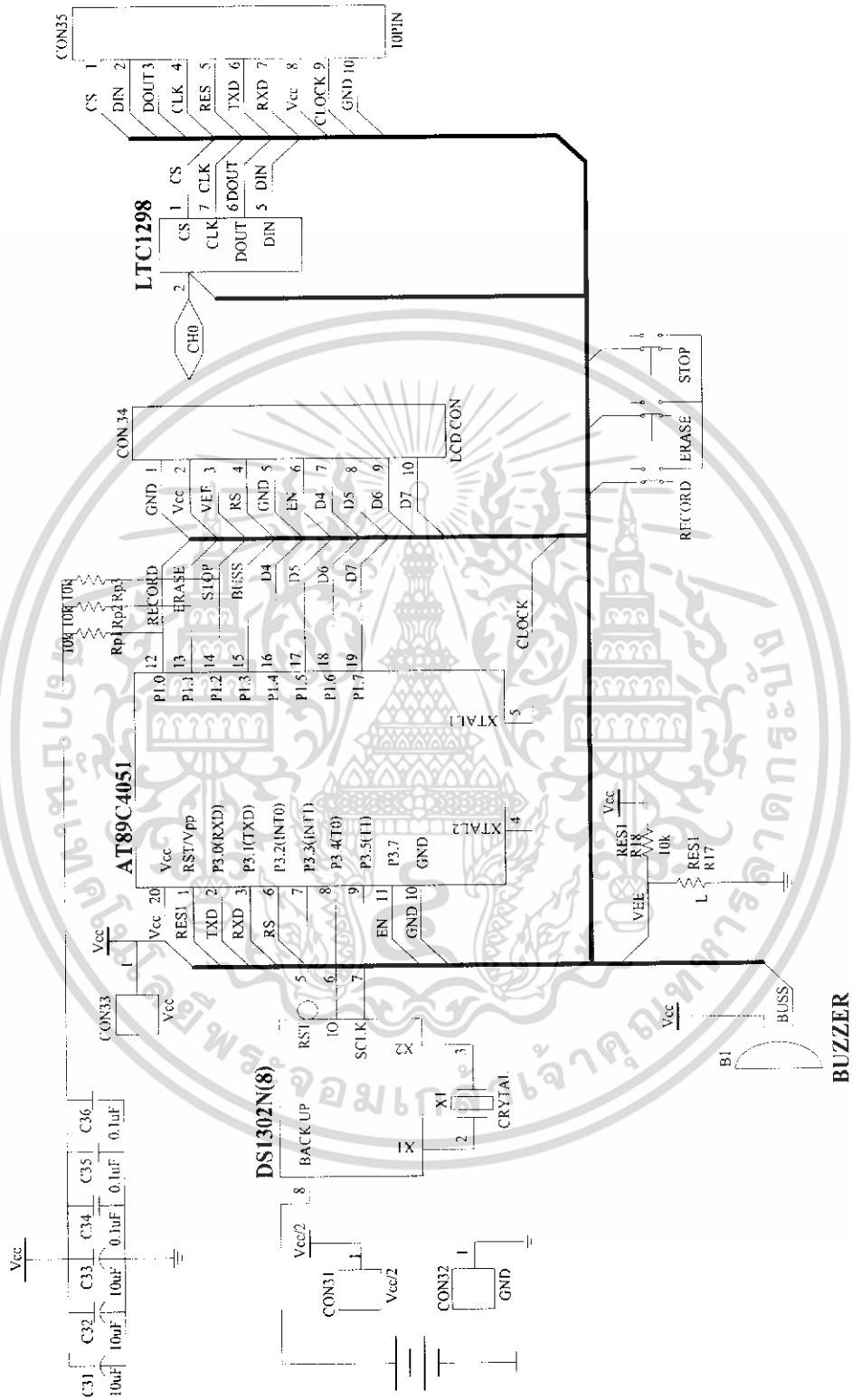
โมดูลของ แอล ซี ดี สามารถที่เชื่อมต่อกับไมโครคอนโทรลเลอร์ได้หลายเบอร์รูปที่
5 แสดงการเชื่อมต่อโมดูล แอล ซี ดี เข้ากับไมโครคอนโทรลเลอร์เบอร์ 8051 โดยมีหน้าที่การ
ทำงานของแต่ละขา ดังนี้คือ ขา 1 ต่อกราวด์ ขา 2 ต่อไฟ 5 – โวลท์ ขา 3 ต่อกับตัวต้านทางปรับค่าได้
แบบโพเทนชิโอมิเตอร์ เมื่อปรับความเข้มและความสว่าง ขา 5 ,4 และ 6 ใช้ต่อสัญญาณควบคุมซึ่ง
ในที่นี้จะใช้พอร์ต 3 ของ 8051 เป็นตัวควบคุม ส่วนบัสข้อมูล (Data Bus) ที่ขา 7 ถึงขา 14 จะต่อ
พอร์ต 1 ของ 8051 เพื่ออ่านและเขียนข้อมูลลงไปบน โมดูล แอล ซี ดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



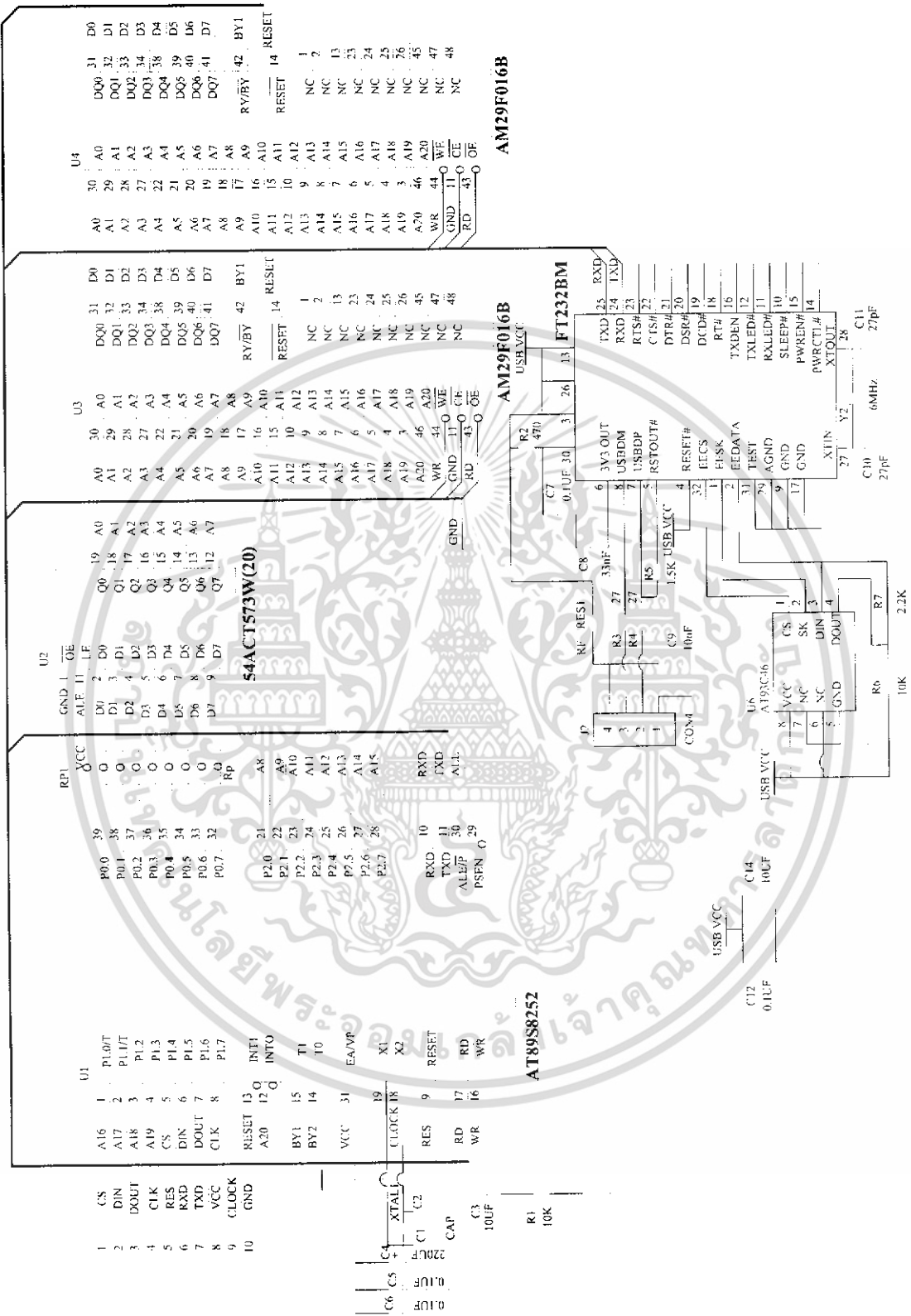
รูปที่ 3.12 รูปวงจรมอลอกของเครื่องบันทึกคลื่นไฟฟ้าหัวใจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.13. รูปวงจรถัดจอของเครื่องบันทึกคลื่นไฟฟ้าหัวใจ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.14 รูปวงจรดิจิทัลของเครื่องบันทึกคลื่นไฟฟ้าหัวใจ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

หลักการและการออกแบบโปรแกรมบนเครื่อง พี ซี (PC)

4.1 หลักการทำงานของเครื่อง

หลักการทำงานของคือ การเริ่มต้นให้ไอซีไมโครคอนโทรลเลอร์ เบอร์ AT89C4051 เซ็ตค่าเริ่มต้นให้ แอลซีดี และทำการรอรับการตั้งค่าเวลา หลังจากนั้นจะเข้ามาสู่โหมดแทนบาย เพื่อรอรับการกดสวิทช์จากผู้ใช้ โดยมีการทำงานอยู่ 3 อย่างคือ

4.1.1 บันทึกสัญญาณคลื่นไฟฟ้าหัวใจลงบนแฟลชเมมโมรี่ โดยใช้ไอซีเบอร์ AT89S8252 ควบคุมการทำงานของไอซีเบอร์ LTC1298 ในการแปลงสัญญาณอนาล็อกเป็นดิจิตอล แล้วทำการบันทึกสัญญาณลงบนตัวไอซีแฟลชเมมโมรี่ (AM29F016B)

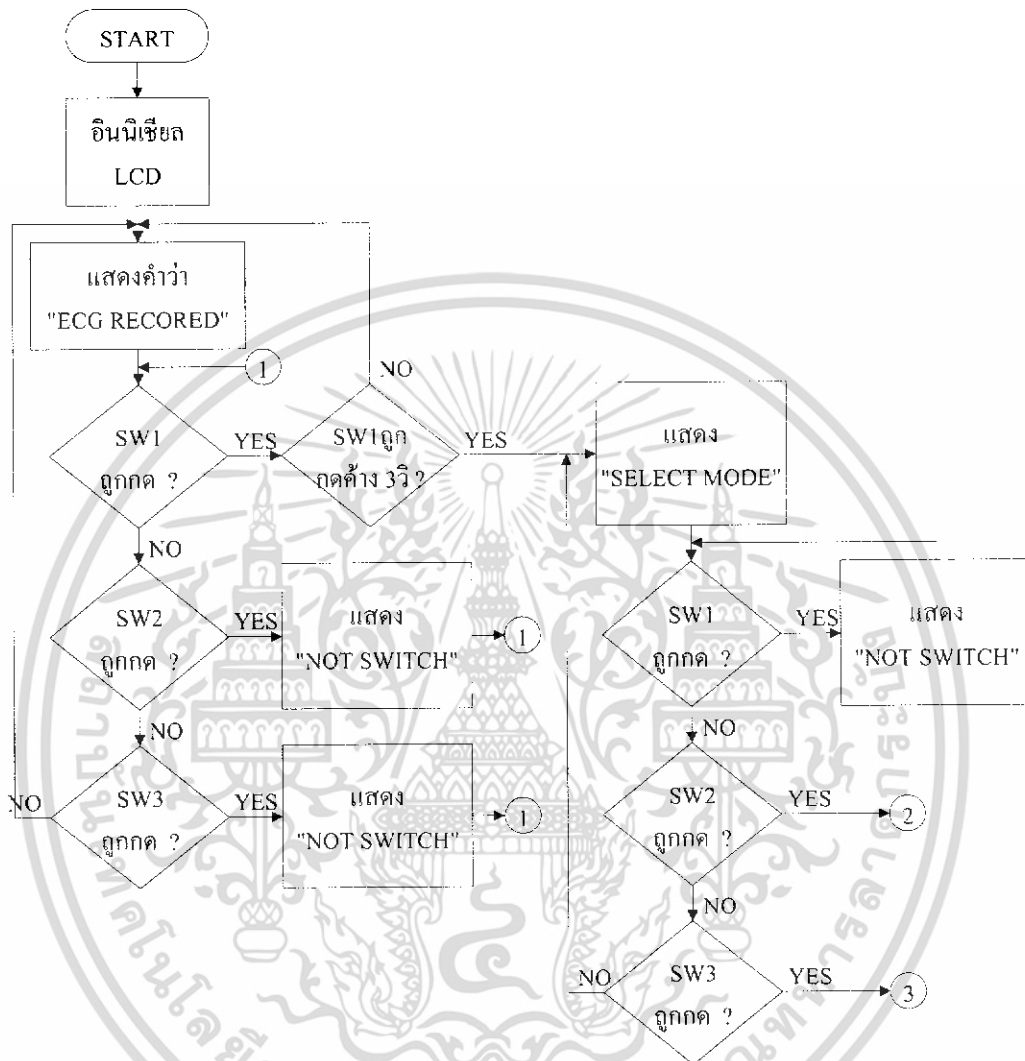
4.1.2 ลบข้อมูลที่อยู่ในแฟลชเมมโมรี่

4.1.3 หยุดการทำงาน โดยใช้การติดต่อสื่อสารผ่านพอร์ตอนุกรม จากไอซีเบอร์ AT89C4051 ไปยัง ไอซีเบอร์ AT89S8252 เพื่อหยุดการทำงานของแฟลชเมมโมรี่

การออกแบบโปรแกรม

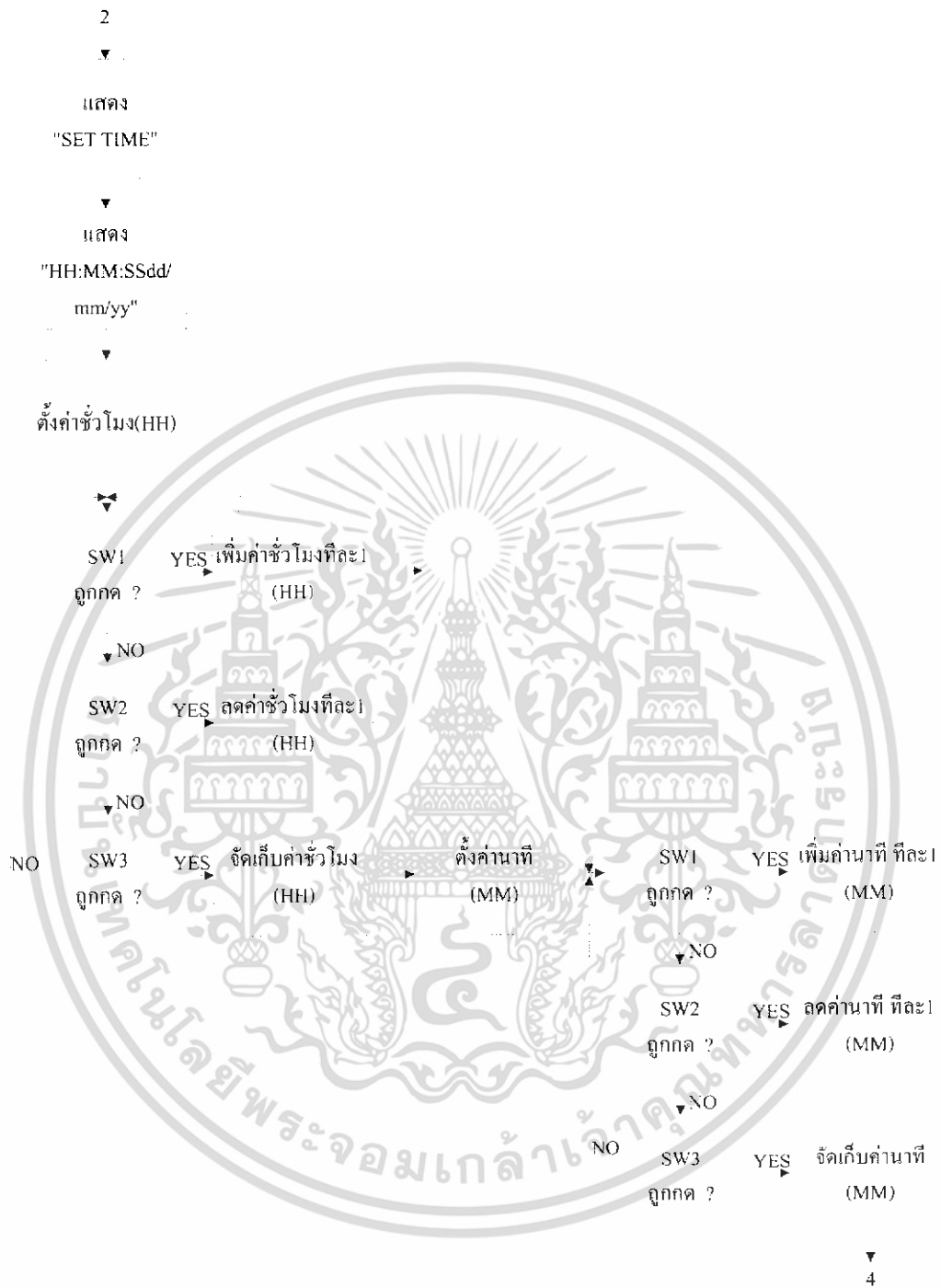
ในการเขียนโปรแกรมจะต้องเขียนออกเป็น 2 ส่วน คือส่วนของไอซีเบอร์ AT89C4051 ซึ่งไอซีเบอร์นี้จะต้องติดต่อกับ แอลซีดี, Real Time Clock(DS1302) และ ไอซีเบอร์ AT89S8252 ซึ่งใช้ควบคุมการทำงานของ แฟลชเมมโมรี่ และ ไอซีเบอร์ LTC1298 โดยพัฒนาโปรแกรมบนเครื่องพี ซี โดยใช้โปรแกรม RAD51

4.2 โฟลชาร์ตแสดงการทำงานของเครื่อง



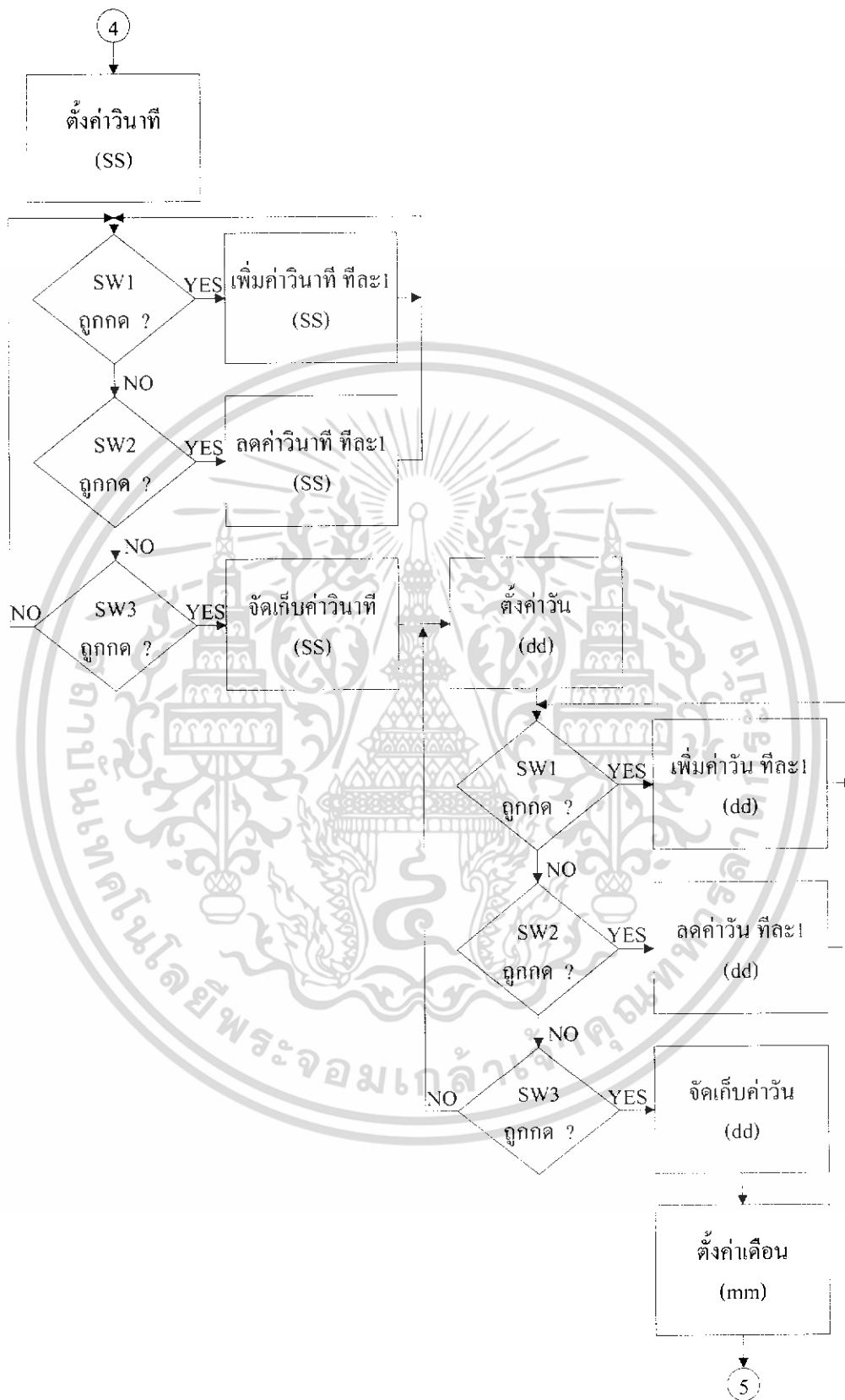
รูปที่ 4.1 รูปแสดง Flow Chart

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



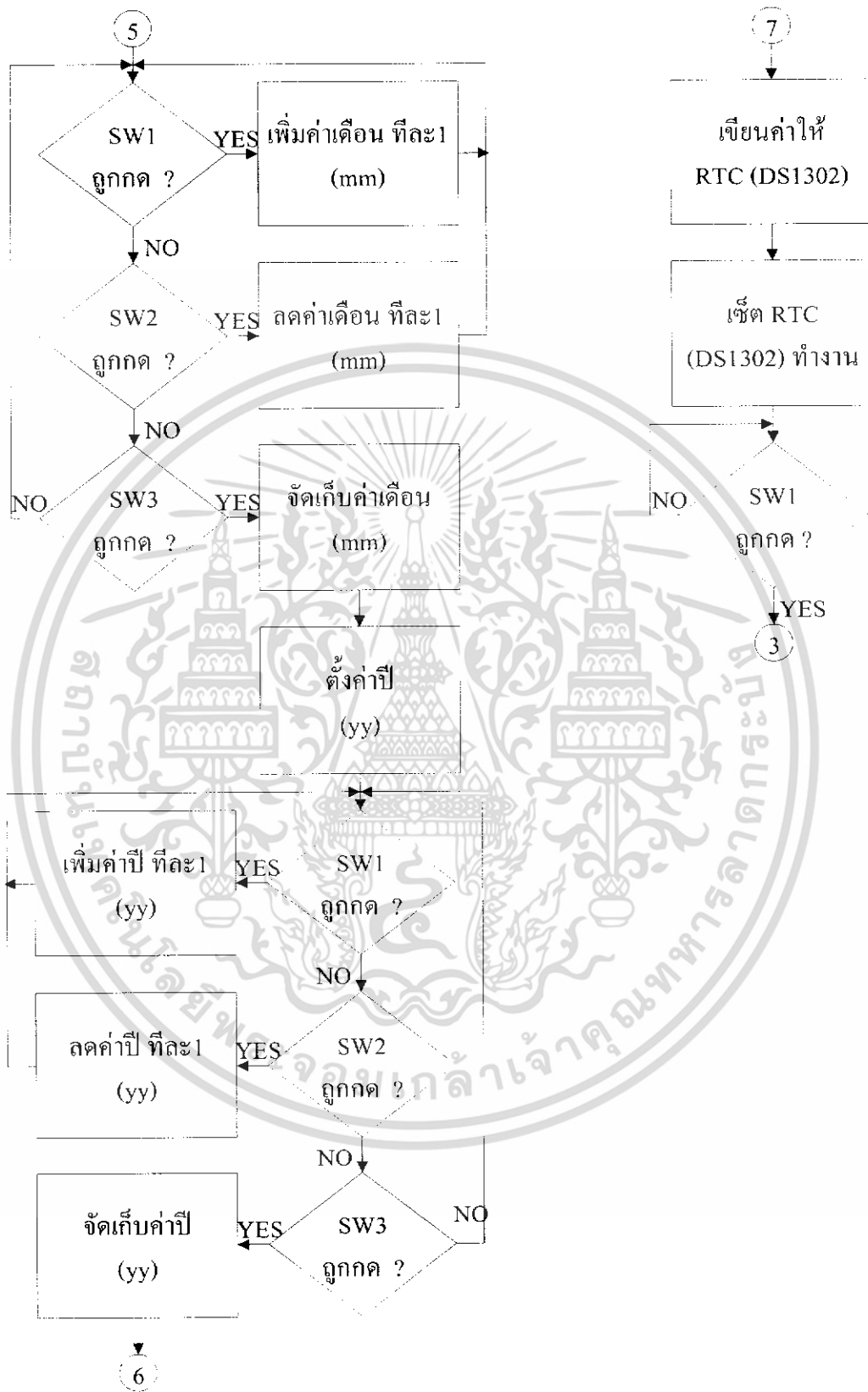
รูปที่4.1 รูปแสดง Flow Chart (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



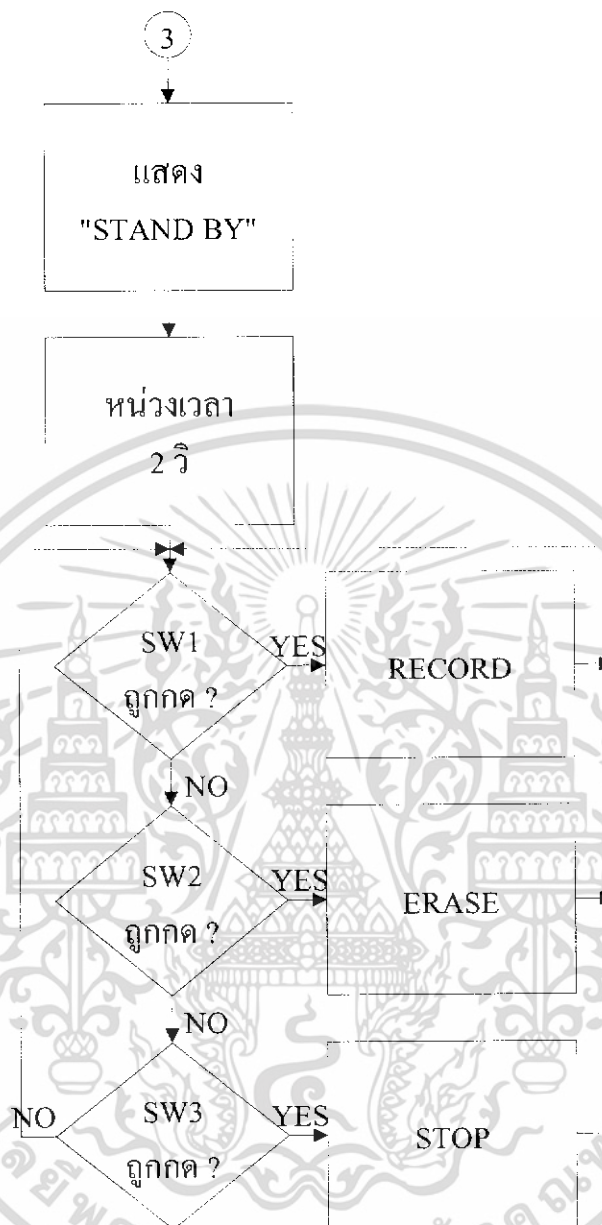
รูปที่4.1 รูปแสดง Flow Chart (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่4.1 รูปแสดง Flow Chart (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



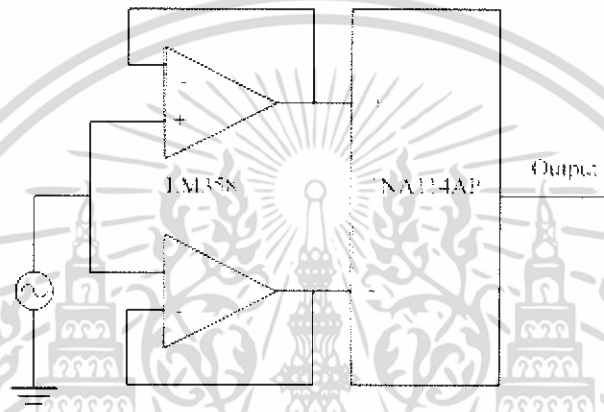
รูปที่ 4.1 รูปแสดง Flow Chart (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

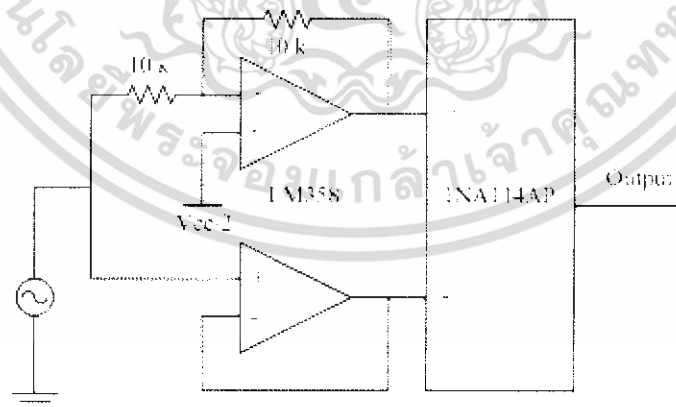
บทที่ 5

การทดลองและผลการทดลอง

5.1 การหาค่า CMRR จะใช้การป้อนสัญญาณรูปไซน์ ความถี่ 1 กิโลเฮิรต์ แล้ววัดสัญญาณจาก common mode และ differential mode ในค่าแอมพลิจูดต่างๆ ทำการบันทึก ค่าแอมพลิจูดของแต่ละโหมดไว้ แล้วนำนามาคิดค่า CMRR ในหน่วยเดซิเบล



รูปที่ 5.1 รูปวงจรใช้วัดสัญญาณ common mode



รูปที่ 5.2 รูปวงจกรใช้วัดสัญญาณ differential mode

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 6 ตารางแสดงผลการทดลองหาค่า CMRR

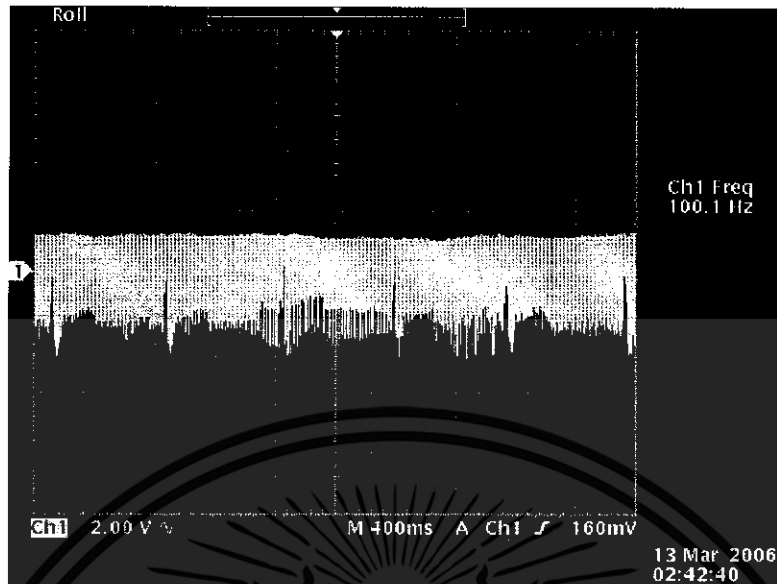
อินพุท	Output ที่ได้จาก common mode	Output ที่ได้จาก differential mode	ค่า CMRR (db)
360 mVp_p	หาค่าไม่ได้	4.56 V.	หาค่าไม่ได้
740 mVp_p	หาค่าไม่ได้	10.2	หาค่าไม่ได้
5 Vp_p	5 mVp_p	50 Vp_p	80 db
10 Vp_p	10 mVp_p	100 Vp_p	80 db

5.2 การวัดคลื่นไฟฟ้าหัวใจ



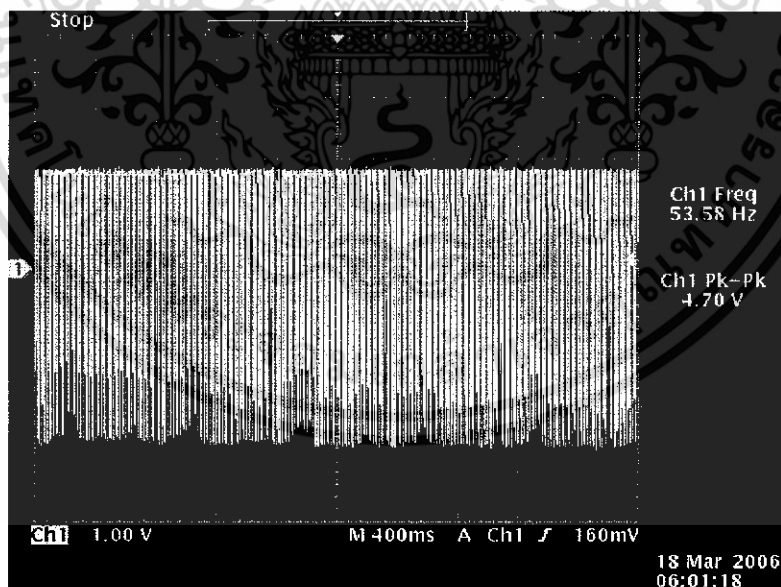
รูปที่ 5.3 รูปคลื่นไฟฟ้าหัวใจของสัญญาณช่อง 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



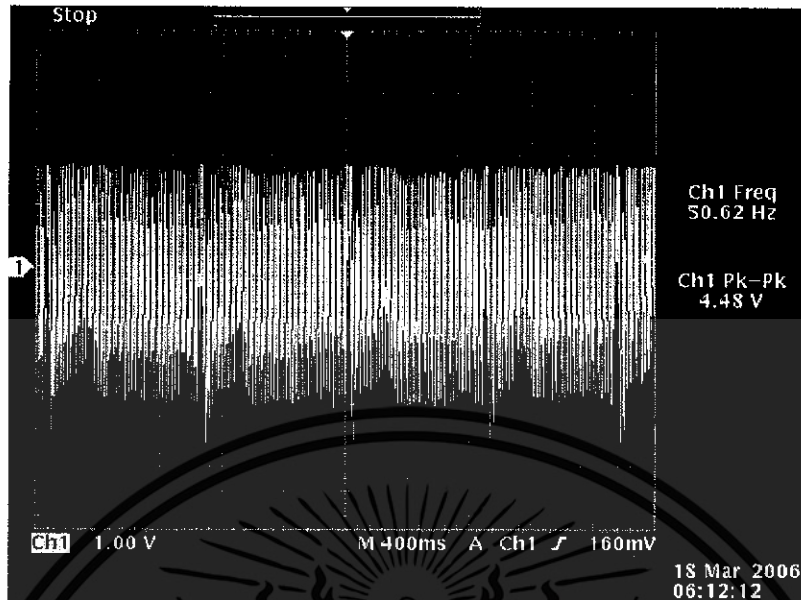
รูปที่ 5.4 รูปคลื่นไฟฟ้าหัวใจของสัญญาณช่อง 1

ในการทดลองนี้ขณะที่ทำการวัดคลื่นไฟฟ้าหัวใจอยู่ถ้ามีการแตะต้องสายที่ใช้จับ อิเล็กโทรด หรือว่าอิเล็กโทรดไม่แน่น จะทำให้สัญญาณเกิดน้อยขึ้นได้ (noise)



รูปที่ 5.5 รูปของสัญญาณเมื่อทำการเอามือแตะสายไฟที่จับอิเล็กโทรด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.6 รูปของสัญญาณเมื่ออิเล็กทรอนิกส์ที่จับยึดไม่แน่น

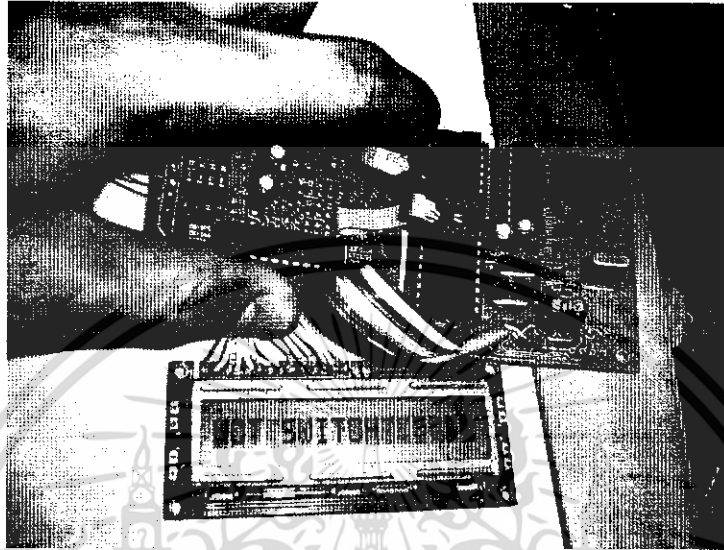
5.3 การทดลองซอฟต์แวร์ ทำการรันซอร์สโค้ด ลงบนไอซีเบอร์ AT89C4051 แล้วทำการเปิดเครื่องเพื่อให้เห็น “ECG RECORDER”



รูปที่ 5.7 รูปตอนเปิดเครื่อง

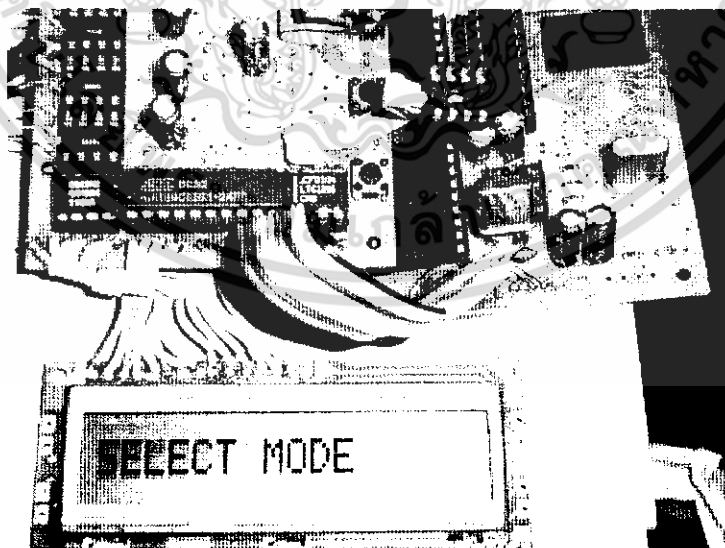
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากหน้าจอ “ECG RECORDER” จะรอตรวจสอบสวิตช์ ถ้ากดสวิตช์ 2 หรือ 3 จะขึ้นว่า “NOT SWITCH” ถ้าปล่อยสวิตช์ จะกลับไปหน้าจอเดิม



รูปที่ 5.8 รูปการกดสวิตช์ที่ 2 หรือ 3

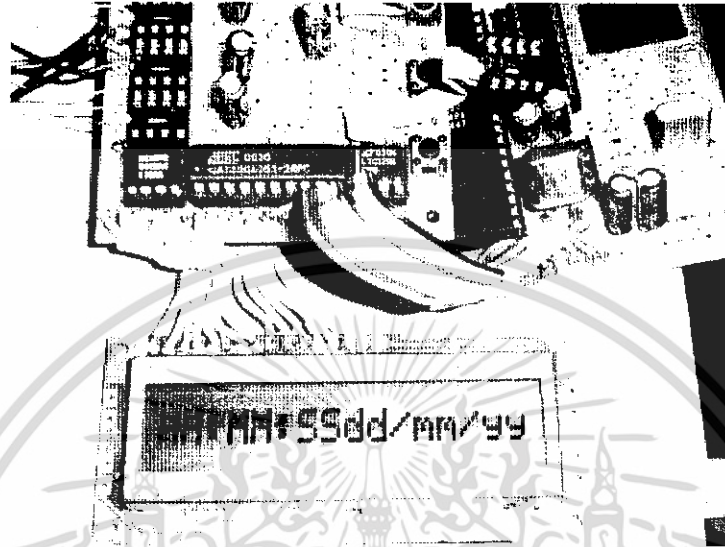
จากหน้าจอ “ECG RECORDER” ต้องกดสวิตช์ 1 ค้างไว้ 3 วินาที หน้าจอจะเปลี่ยนเป็น “SELECT MODE” แล้วจะรอตรวจสอบการกดสวิตช์ที่ 2 และ 3 ต่อ



รูปที่ 5.9 รูปหน้าจอจากการกดสวิตช์ 1 ค้างไว้ 3 วินาที

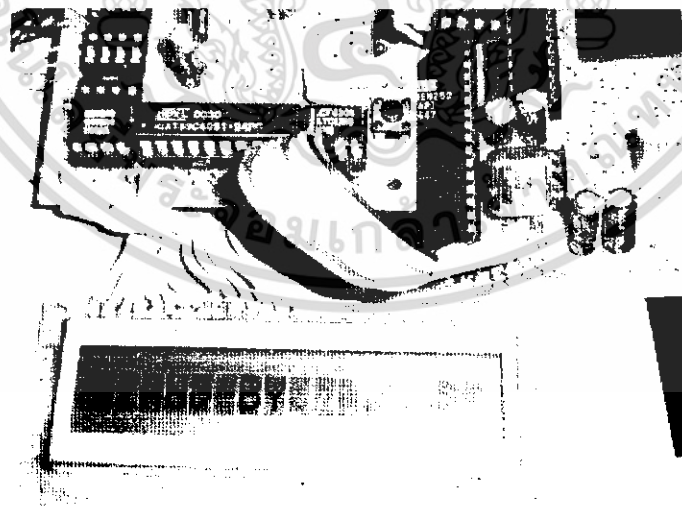
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเข้าสู่หน้าจอ “SELECT MODE” แล้ว จะรอกการกดสวิทช์ 2 เพื่อทำการตั้งเวลาและวันที่ โดยหน้าจอจะขึ้นว่า “HH:MM:SSdd/mm/yy”



รูปที่ 5.10 หน้าจอการตั้งเวลา

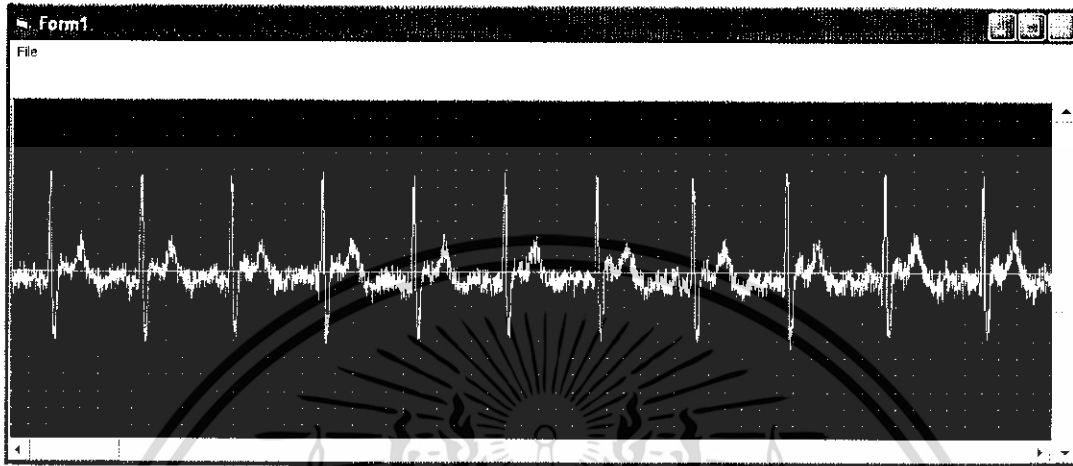
เมื่อตั้งเวลาเสร็จเรียบร้อยแล้ว เครื่องก็จะเริ่มนับเวลา และแสดงหน้าจอ หลังจากนั้นจะเข้าสู่หน้าจอ “STAND BY” เพื่อรอกการกดสวิทช์ เพื่อรอรับการทํางานในโหมดต่างๆ



รูปที่ 5.11 รูปหน้าจอของ โหมดแสดนบาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.4 การแสดงผล นำสัญญาณคลื่นไฟฟ้าหัวใจที่ทำการบันทึกและจัดเก็บไว้ในแฟลชเมมโมรี่ แล้วนำมาแสดงผลบนคอมพิวเตอร์



รูปที่ 5.12 รูปสัญญาณคลื่นไฟฟ้าหัวใจที่แสดงผลบนคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

บทสรุป

6.1 สรุป

จากการทำโครงการชิ้นนี้ มีวัตถุประสงค์เพื่อใช้ในการบันทึกคลื่นไฟฟ้าหัวใจของผู้ป่วยโรคหัวใจขณะที่มีอาการ โดยใช้วงจรขยายแบบอินสตุเมนเตชัน แล้วใช้ไมโครคอนโทรลเลอร์ควบคุมการทำงานของวงจรทั้งหมด เริ่มจากการแปลงสัญญาณคลื่นไฟฟ้าหัวใจเป็นดิจิตอล แล้วบันทึกสัญญาณที่ได้จัดเก็บไว้ในหน่วยความจำแบบแฟลช และนำมาแสดงผลบนคอมพิวเตอร์เพื่อให้แพทย์ได้ทำการวินิจฉัยอาการของผู้ป่วยจากสัญญาณที่ทำการบันทึกไว้

จากการทดลองทั้งหมดสามารถสรุปได้ดังนี้

การทดลองเพื่อหาค่า CMRR เมื่อป้อนสัญญาณที่แอมพลิจูดต่ำๆ สัญญาณที่ได้จากวงจร Differential mode จะถูกขยายประมาณ 10 เท่าตามเกณฑ์ที่ใช้ ส่วนสัญญาณที่ได้จาก Common mode จะไม่สามารถวัดค่าได้ แต่เมื่อลองปรับขนาดแหล่งจ่ายไฟเพิ่มขึ้น และเพิ่มสัญญาณอินพุตขึ้น สัญญาณจาก Common mode จะทำการวัดได้แต่จะมีขนาดต่ำมากๆ ส่วนสัญญาณที่ได้จาก Differential mode จะไม่สามารถวัดได้ แต่หาค่าได้จากการคำนวณ ทำให้ค่า CMRR ที่ได้มีค่า 80 เดซิเบล ส่วนค่า CMRR จากคู่มือของไอซีอินสทรูเมนต์มีค่า 115 เดซิเบล ที่อัตราขยาย 1000 เท่า แต่เนื่องจากวงจรที่ใช้ ต้องการอัตราขยายแค่ 10 เท่า เพื่อไม่ให้สัญญาณรบกวนถูกขยายเข้ามาด้วย ทำให้ค่า CMRR ที่ได้ไม่ตรงกับค่า CMRR จากคู่มือของไอซีอินสทรูเมนต์ เนื่องจากถ้าขนาดของเกณฑ์มีการเปลี่ยนแปลง ค่าแอมพลิจูดที่ได้จาก Differential mode จะเกิดการเปลี่ยนแปลงไปตามเกณฑ์ ทำให้ค่า CMRR ที่ได้ก็จะเปลี่ยนไปตามเกณฑ์ด้วย

การทดลองวัดคลื่นไฟฟ้าหัวใจ ที่ช่อง 0 จะเห็นได้ว่าสัญญาณที่วัดได้ ยังมีน้อยช้อยู่บ้าง ส่วนสัญญาณที่ช่อง 1 รูปสัญญาณที่จะเกิดการ Saturation ขึ้น เนื่องจากมีสัญญาณรบกวนที่ถูกขยายเข้ามาด้วย ทำให้สัญญาณถูกขยายขึ้นจนเกิดการ Saturation และเมื่อทำการเอามือจับที่สายไฟที่จับอิเล็กโทรด ทำให้มีสัญญาณรบกวนเกิดขึ้นและถูกขยายจน Saturation และถ้าอิเล็กโทรดที่ใช้จับยึดที่ผิวหนังติดไม่แน่น จะทำให้ค่าความต้านทานระหว่างหน้าสัมผัสของอิเล็กโทรดกับผิวหนังเกิดการเปลี่ยนแปลง และจะมีสัญญาณรบกวนปนเข้ามาด้วย

จากการทดลองวงจรอนาล็อกที่ใช้วัดคลื่นไฟฟ้าหัวใจ จะพบว่าสัญญาณที่ได้ยังมีน้อยช้อยู่ ซึ่งนำไปใช้ในการวินิจฉัยไม่ได้ แต่ใช้ในการมอนิเตอร์ได้ ซึ่งไม่เป็นไปตามวัตถุประสงค์ที่ต้องการทำเครื่องบันทึกคลื่นไฟฟ้าหัวใจเพื่อใช้ในการวินิจฉัยอาการของผู้ป่วย

การทดลองทางด้านซอฟต์แวร์ เมื่อทำการรันโปรแกรมลงบนไอซีเอ็มโครคอนโทรลเลอร์ แล้วทำการเปิดเครื่องจะเห็นได้ว่าการทำงานในโหมดต่างๆ สามารถทำได้ถูกต้อง

เมื่อนำสัญญาณที่ถูกบันทึกไว้ในแฟลชเมมโมรี่มาแสดงผลบนคอมพิวเตอร์ จะเห็นได้ว่าสัญญาณจะใกล้เคียงกับสัญญาณที่วัดได้จากวงจรอนาล็อก

6.2 วิจารณ์และข้อเสนอแนะ

จากการทดลองทั้งหมด จะพบว่าในการวัดสัญญาณคลื่นไฟฟ้าหัวใจ ถ้าเกิดการเปลี่ยนแปลงทางกายภาพขึ้น จะทำให้รูปสัญญาณที่ทำการวัดเกิดการเปลี่ยนแปลงขึ้นได้ เนื่องจากสัญญาณคลื่นไฟฟ้าหัวใจมีความถี่ต่ำมาก ส่วนรูปคลื่นไฟฟ้าหัวใจที่ทำการวัดได้ที่มีน้อยชั่ปะปนอยู่นั้น ถ้านำสัญญาณไปผ่านวงจรกรองความถี่ 50 เฮิร์ตออก จะทำให้สัญญาณมีน้อยชั่ลดลงและสามารถนำไปให้แพทย์ทำการวินิจฉัยได้

แนวทางการแก้ไขก็คือ การลดขนาดของเกณฑ์ในวงจรอินสตรูเมนต์แอมป์ลง เพื่อช่วยลดการขยายสัญญาณรบกวนที่เกิดขึ้นในระหว่างทำการวัด และการนำสัญญาณคลื่นไฟฟ้าหัวใจไปผ่านวงจรกรองความถี่ 50 เฮิร์ตออก เพื่อลดสัญญาณรบกวนที่ปะปนมากับไฟบ้าน ซึ่งจะช่วยให้สามารถทำการวัดสัญญาณคลื่นไฟฟ้าหัวใจได้อย่างถูกต้อง ส่วนทางด้านซอฟต์แวร์ จะต้องเขียนซอฟต์แวร์ให้กระชับเข้าใจง่าย และเขียนให้เป็นระเบียบ เพื่อจะได้ทำการแก้ไข และปรับปรุงได้ง่าย

หนังสืออ้างอิง

1. สักกะพันธ์ กล้ายดอกจันทร์. “การบีบอัดสัญญาณไฟฟ้าหัวใจในเวลาจริงโดยใช้การแปลงเวฟเลตบน MCS-51”, วิทยานิพนธ์, 2545, หน้า 4 – 12 .
2. ไพบุลย์ บุปผา, “เทคนิคการใช้งานแอลซีดีโมดูล”, วารสารเซมิคอนดักเตอร์อิเล็กทรอนิกส์, ฉบับที่ 166, 2539, หน้า48 – 53



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซอร์สโค้ดของไอซีเบอร์ AT89C4051

```

lcd_rs          bit        p3.2
lcd_en          bit        p3.7
lcd_data        equ        p1
record          bit        p1.0
erase          bit        p1.1
stop           bit        p1.2
record_l        equ        01h
erase_l         equ        02h
stop_l          equ        03h
read_l          equ        04h
flash_full      equ        55h
;*****      rtc *****
rtc_data        bit        P3.4
rtc_clk         bit        P3.5
rtc_rst        bit        P3.3
r_second        equ        40h
r_minute        equ        41h
r_hour          equ        42h
r_date          equ        43h
r_month         equ        44h
r_year          equ        45h
b_second        equ        50h
buff_ti         equ        30h
;*****
org             0000
ljmp            start
;*****
;
; Interrupt Serial
;*****
org            0023h
pop            acc
pop            psw
jnb           ri,recive
clr           ti
push          psw
push          acc
reti

recive:        mov            a,sbuf
               cjne         a,#read_l,recive__3
               clr          es

recive_1:      mov            r7,#05h
recive_2:      jnb           ri,$
               clr          ri
               mov          a,sbuf
               cjne         a,#0aah,recive_1
               djnz         r7,recive_2
               setb         es

recive_3:      push          psw
               push          acc
               reti

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;*****
;
;                               START
;*****

start:                            lcall    delay500ms
                                lcall    init_lcd
                                mov     r2,#00h
                                mov     r3,#00h
                                djnz   r3,$
                                djnz   r2,$-2
                                mov     scon,#50h
                                setb   ren
                                setb   ea
                                setb   es
                                mov     tmod,#21h
                                mov     th1,#0fch
                                mov     tl1,#0fch
                                setb   tr1
start_1:                          lcall    show_ecg

chk3s_sw1:                        jb     record,chk3s_sw2
                                mov     r4,#30d
chk3s_rec:                        lcall   delay100ms
                                jb     record,chk3s_sw2
                                djnz   r4,chk3s_rec
                                lcall   show_select_mode
                                lcall   delay500ms
                                ljmp   select_mode
chk3s_sw2:                        jb     erase,chk3s_sw3
                                lcall   show_press_other
                                jnb   erase,chk3s_sw2
                                ljmp   start_1
chk3s_sw3:                        jb     stop,chk3s_sw1
                                lcall   show_press_other
                                jnb   stop,chk3s_sw3
                                ljmp   start_1

;-----
;                               Time Setting
;-----
select_mode:                      jb     erase,sl_model1
                                ljmp   rtc_set

;-----
;                               Stand by
;-----
sl_model1:                        jb     stop,sl_model2
sl_mode_1:                        lcall  show_stand_by
                                lcall  delay1s
                                ljmp   press_key

sl_model2:                        jb     record,select_mode
                                sjmp   select_mode

;-----
;                               Record
;-----
press_key11:                      ljmp   press_key1
press_key:                        jb     record,press_key11

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        clr         es
        lcall      show_recording
        lcall      delayls
        mov        sbuf,#record_1
        jnb       ti,$
        clr       ti
        jnb       ri,$
        clr       ri
        mov        sbuf,#29h
        jnb       ti,$
        clr       ti
        jnb       ri,$
        clr       ri
        mov        sbuf,#59h
        jnb       ti,$
        clr       ti
        jnb       ri,$
        clr       ri
        mov        sbuf,#59h
        jnb       ti,$
        clr       ti
        jnb       ri,$
        clr       ri
        mov        sbuf,#08h
        jnb       ti,$
        clr       ti
        jnb       ri,$
        clr       ri
        mov        sbuf,#03h
        jnb       ti,$
        clr       ti
        jnb       ri,$
        clr       ri
        mov        sbuf,#49h
        jnb       ti,$
        clr       ti
        jnb       ri,$
        clr       ri

press_key_1:
        jnb       stop,press_key_2
        jnb       ri,press_key_1
        clr       ri
        mov        a,sbuf
        cjne      a,#flash_full,press_key_1
        lcall     show_flash_full
        lcall     delayls
        lcall     delayls
        setb      es
        ljmp      si_mode_1

press_key_2:
        mov        sbuf,#stop_1
        jnb       ti,$
        clr       ti
        jnb       ri,$
        clr       ri
        lcall     show_stop
        lcall     delayls

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

                                setb     es
                                ljmp     sl_mode_1
;-----
;                               Erase
;-----
press_key1:                    jb     erase,press_key2
                                clr     es
                                lcall   show_erasing
                                mov     sbuf,#erase_i
                                jnb     ti,$
                                clr     ti
                                jnb     ri,$
                                clr     ri
                                jnb     ri,$
                                clr     ri
                                icall   show_eration_finished
                                lcall   delay_s
                                lcall   delay_s
                                setb     es
                                ljmp     sl_mode_1

key_1:                          ljmp     press_key
press_key2:                      jb     stop,key_1
;-----
;                               Initial LCD
;-----
init_lcd:                       lcall   delay_lcd
                                clr     lcd_rs
                                clr     lcd_en
                                setb    lcd_en
                                mov     a,#28h
                                clr     lcd_en
                                djnz    r2,$
                                djnz    r3,$-2
                                lcall   writeinst
                                mov     a,#28H
                                lcall   writeinst
                                mov     a,#0CH
                                lcall   writeinst
                                mov     a,#01H
                                lcall   writeinst
                                mov     a,#06h
                                lcall   writeinst
                                mov     r2,#0Ch
                                mov     r3,#0Ch
                                djnz    r3,$
                                djnz    r2,$-2
                                ret
;-----
;                               writeinst
;-----
writeinst:                       clr     lcd_rs
                                lcall   write_2nibbles
                                setb    lcd_rs
                                ret
;-----

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;          writechar
;-----
writechar:          push      dph
                   push      dpl
                   setb      lcd_rs
                   lcall     write_2nibbles
                   pop       dpl
                   pop       dph
                   ret

;-----
;          write_2nibbles
;-----
write_2nibbles:    push      acc
                   orl       lcd_data,#0f0h
                   orl       a,#0fh
                   anl      lcd_data,a
                   lcall     delay_lcd
                   pop       acc
                   swap      a
                   orl       lcd_data,#0f0h
                   orl       a,#0fh
                   anl      lcd_data,a
                   lcall     delay_lcd
                   ret

;-----
;          deley_lcd
;-----
delay_lcd:        mov       r7,#00h
                   setb      lcd_en
                   djnz     r7,$
                   clr      lcd_en
                   ret

;-----
;          Deley 1 s.
;-----
delay1s:         mov       r5,#08h
                   mov       r6,#0Ch
                   mov       r7,#0Ch
del1:            djnz     r7,$
                   djnz     r6,del1
                   djnz     r5,del1
                   ret

;-----
;          Deley 500 ms.
;-----
delay500ms:      mov       r5,#04h
                   mov       r6,#00h
                   mov       r7,#00h
del500:          djnz     r7,$
                   djnz     r6,del500
                   djnz     r5,del500
                   ret

;-----
;          Deley 100 rs.
;-----
delay100ms:      mov       r7,#100
delay1ms:        mov       r6,#0e6h

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

delay_loop2:      nop
                  nop
                  djnz      r6,delay_loop2
                  djnz      r7,delay_1ms
                  ret

;-----
;          DISPLAY ON LCD
;-----
;SHOW "ECG RECORDER"
show_ecg:         mov        dptr,#table1
                  lcall      display_h
                  mov        dptr,#table1+8
                  lcall      display_l
                  ret

show_select_mode: mov        dptr,#table2
                  lcall      display_h
                  mov        dptr,#table2+8
                  lcall      display_l
                  ret

show_press_other: mov        dptr,#table3
                  lcall      display_h
                  mov        dptr,#table3+8
                  lcall      display_l
                  ret

show_time_setting: mov        dptr,#table4
                  lcall      display_h
                  mov        dptr,#table4+8
                  lcall      display_l
                  ret

show_time:        mov        dptr,#table5
                  lcall      display_h
                  mov        dptr,#table5+8
                  lcall      display_l
                  ret

show_stand_by:    mov        dptr,#table6
                  lcall      display_h
                  mov        dptr,#table6+8
                  lcall      display_l
                  ret

show_recording:   mov        dptr,#table7
                  lcall      display_h
                  mov        dptr,#table7+8
                  lcall      display_l
                  ret

show_stop:        mov        dptr,#table8
                  lcall      display_h
                  mov        dptr,#table8+8
                  lcall      display_l
                  ret

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

show_erasing:      mov     dptr,#table9
                  lcall   display_h
                  mov     dptr,#table9+8
                  lcall   display_l
                  ret

show_eration_finished: mov  dptr,#table10
                  lcall   display_h
                  mov     dptr,#table10+8
                  lcall   display_l
                  ret

show_flash_full:   mov     dptr,#table11
                  lcall   display_h
                  mov     dptr,#table11+8
                  lcall   display_l
                  ret

show_read_data:   mov     dptr,#table12
                  lcall   display_h
                  mov     dptr,#table12+8
                  lcall   display_l
                  ret

;-----
;               display_lcd
;-----
display_h:        mov     r2,#08h
                  mov     a,#80h
                  lcall   writeinst
lcd100:           mov     a,#00h
                  movc    a,@a+dptr
                  lcall   writechar
                  inc     dptr
                  djnz    r2,lcd100
                  ret

display_l:        mov     r2,#08h
                  mov     a,#0cCh
                  lcall   writeinst
lcd200:           mov     a,#00h
                  movc    a,@a+dptr
                  lcall   writechar
                  inc     dptr
                  djnz    r2,lcd200
                  ret

;-----
;               Table
;-----

table1:           db     "ECG RECORDER   "
table2:           db     "SELECT MODE   "
table3:           db     "PRESS OTHER   "
table4:           db     "TIME SETTING  "
table5:           db     "HH:MM:SSdd/mm/yy"
table6:           db     "STAND BY     "

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

table7:          db "RECORDER      "
table8:          db "STOP          "
table9:          db "ERASING       "
table10:         db "ERASION FINISHED"
table11:         db "MEMORY IS FULL "
table12:         db "READING DATA "

;*****
;*****
rtc_set:         lcall          show_time_setting
                lcall          delays
                lcall          delays
                lcall          show_time
                lcall          delays
                mov            buff_t1,#00h
loop_sw_1:      ljmp           display_hour
                jnb           record,chack_sw1_1
                jnb           erase,chack_sw2_1
                jnb           stop,chack_sw3_1
                ljmp          loop_sw_1

chack_sw1_1:    lcall          delays
                jb            record,loop_sw_1
                inc           buff_t1
                mov           a,buff_t1
                cjne          a,#18h,display_hour
                mov           buff_t1,#00h
                ljmp          display_hour

chack_sw2_1:    lcall          delays
                jb            erase,loop_sw_1
                mov           a,buff_t1
                cjne          a,#00h,switch2_1
                ljmp          display_hour

switch2_1:     dec            buff_t1
                ljmp          display_hour

display_hour:  lcall          show_lcd
                mov           r_hour,a
                lcall          show_hour
                ljmp          loop_sw_1

chack_sw3_1:   lcall          delays
                jb            stop,loop_sw_1
                mov           buff_t1,#00h
                ljmp          display_minute

loop_sw_2:     jnb           record,chack_sw1_2
                jnb           erase,chack_sw2_2
                jnb           stop,chack_sw3_2
                ljmp          loop_sw_2

;-----
;           Set Minute
;-----
chack_sw1_2:   lcall          delays
                jb            record,loop_sw_2
                inc           buff_t1
                mov           a,buff_t1
                cjne          a,#03ch,display_minute

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

                                mov     buff_t1,#00h
                                ljmp    display_minute

chack_sw2_2:                    lcall   delays
                                jnb    erase,loop_sw_2
                                mov     a,buff_t1
                                cjne   a,#00h,switch2_2
                                ljmp    display_minute
switch2_2:                      dec     buff_t1
                                ljmp    display_minute

display_minute:                 lcall   show_lcd
                                mov     r_minute,a
                                lcall   show_minute
                                ljmp    loop_sw_2

chack_sw3_2:                    lcall   delays
                                jnb    stop,loop_sw_2
                                mov     buff_t1,#00h
                                ljmp    display_second
loop_sw_3:                      jnb    record,chack_sw1_3
                                jnb    erase,chack_sw2_3
                                jrb    stop,chack_sw3_3
                                ljmp    loop_sw_3
;-----
;                               Set Second
;-----
chack_sw1_3:                    lcall   delays
                                jnb    record,loop_sw_3
                                inc     buff_t1
                                mov     a,buff_t1
                                cjne   a,#03ch,display_second
                                mov     buff_t1,#00h
                                ljmp    display_second

chack_sw2_3:                    lcall   delays
                                jnb    erase,loop_sw_3
                                mov     a,buff_t1
                                cjne   a,#00h,switch2_3
                                ljmp    display_second
switch2_3:                      dec     buff_t1
                                ljmp    display_second

display_second:                 lcall   show_lcd
                                mov     r_second,a
                                lcall   show_second
                                ljmp    loop_sw_3

chack_sw3_3:                    lcall   delays
                                jnb    stop,loop_sw_3
                                mov     buff_t1,#00h
                                ljmp    display_date
loop_sw_4:                      jnb    record,chack_sw1_4
                                jnb    erase,chack_sw2_4
                                jnb    stop,chack_sw3_4
                                ljmp    loop_sw_4
;-----

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;           Set Date
;-----
chack_sw1_4:    lcall    delay1s
                jb       record,loop_sw_4
                inc     buff_t1
                mov     a,buff_t1
                cjne   a,#20h,display_date
                mov     buff_t1,#00h
                ljmp    display_date

chack_sw2_4:    lcall    delay1s
                jb       erase,loop_sw_4
                mov     a,buff_t1
                cjne   a,#00h,switch2_4
                ljmp    display_date
switch2_4:     dec     buff_t1
                ljmp    display_date

display_date:  lcall    show_lcd
                mov     r_date,a
                lcall  show_date
                ljmp    loop_sw_4

chack_sw3_4:   lcall    delay1s
                jb       stop,loop_sw_4
                mov     buff_t1,#00h
                ljmp    display_month
loop_sw_5:     jnb     record,chack_sw1_5
                jnb     erase,chack_sw2_5
                jnb     stop,chack_sw3_5
                ljmp    loop_sw_5
;-----
;           Set Month
;-----
chack_sw1_5:   lcall    delay1s
                jb       record,loop_sw_5
                inc     buff_t1
                mov     a,buff_t1
                cjne   a,#0dh,display_month
                mov     buff_t1,#00h
                ljmp    display_month

chack_sw2_5:   lcall    delay1s
                jb       erase,loop_sw_5
                mov     a,buff_t1
                cjne   a,#00h,switch2_5
                ljmp    display_month
switch2_5:     dec     buff_t1
                ljmp    display_month

display_month: lcall    show_lcd
                mov     r_month,a
                lcall  show_month
                ljmp    loop_sw_5

chack_sw3_5:   lcall    delay1s
                jb       stop,loop_sw_5
                mov     buff_t1,#00h

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

loop_sw_6:          ljmp          display_year
                   jnb          record,chack_sw1_6
                   jnb          erase,chack_sw2_6
                   jnb          stop,chack_sw3_6
                   ljmp          loop_sw_6
;-----
;                   Set Year
;-----
chack_sw1_6:       lcall          delays
                   jb           record,loop_sw_6
                   inc          buff_t1
                   mov          a,buff_t1
                   cjne         a,#064h,display_year
                   mov          buff_t1,#00h
                   ljmp          display_year

chack_sw2_6:       lcall          delays
                   jb           erase,loop_sw_6
                   mov          a,buff_t1
                   cjne         a,#00h,switch2_6
                   ljmp          display_year
switch2_6:        dec          buff_t1
                   ljmp          display_year

display_year:     lcall          show_lcd
                   mov          r_year,a
                   lcall          show_year
                   ljmp          loop_sw_5

chack_sw3_6:       lcall          delays
                   jb           stop,loop_sw_6
                   lcall          delays
                   clr          rtc_rst
                   setb         rtc_clk
                   lcall          delay
                   lcall          rtc_init
                   lcall          rtc_display
                   sjmp          $

;-----
;                   Change dec to ASCII
;-----
show_lcd:         mov          a,buff_t1
                   mov          b,#0ah
                   div          ab
                   swap         a
                   orl          a,b

                   mov          b,a
                   anl          a,#0f0h
                   swap         a
                   add          a,#30h
                   mov          r3,a
                   mov          a,b
                   anl          a,#0fh
                   add          a,#30h
                   mov          r4,a

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

                                mov     a,b
                                ret
;-----
;                               Show Time Set
;-----
show_hour:                      mov     a,#080h
                                lcall   writeinst
                                mov     a,r3
                                lcall   writechar
                                mov     a,#081h
                                lcall   writeinst
                                mov     a,r4
                                lcall   writechar
                                ret

show_minute:                    mov     a,#083h
                                lcall   writeinst
                                mov     a,r3
                                lcall   writechar
                                mov     a,#084h
                                lcall   writeinst
                                mov     a,r4
                                lcall   writechar
                                ret

show_second:                    mov     a,#086h
                                lcall   writeinst
                                mov     a,r3
                                lcall   writechar
                                mov     a,#087h
                                lcall   writeinst
                                mov     a,r4
                                lcall   writechar
                                ret

show_date:                      mov     a,#0c0h
                                lcall   writeinst
                                mov     a,r3
                                lcall   writechar
                                mov     a,#0c1h
                                lcall   writeinst
                                mov     a,r4
                                lcall   writechar
                                ret

show_month:                     mov     a,#0c3h
                                lcall   writeinst
                                mov     a,r3
                                lcall   writechar
                                mov     a,#0c4h
                                lcall   writeinst
                                mov     a,r4
                                lcall   writechar
                                ret

show_year:                      mov     a,#0c6h
                                lcall   writeinst

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        mov     a,r3
        lcall  writechar
        mov     a,#0c7h
        lcall  writeinst
        mov     a,r4
        lcall  writechar
        ret

;-----
;           Initial RTC
;-----
rtc_init:    mov     r2,#8eh
            mov     r3,#00h
            lcall  rtc_write_ch
            mov     r2,#80h
            mov     r3,r_second
            lcall  rtc_write_ch
            mov     r2,#82h
            mov     r3,r_minute
            lcall  rtc_write_ch
            mov     r2,#84h
            mov     r3,r_hour
            lcall  rtc_write_ch
            mov     r2,#86h
            mov     r3,r_date
            lcall  rtc_write_ch
            mov     r2,#88h
            mov     r3,r_month
            lcall  rtc_write_ch
            mov     r2,#8ch
            mov     r3,r_year
            lcall  rtc_write_ch
            mov     r2,#8eh
            mov     r3,#80h
            lcall  rtc_write_ch
            ret

;-----
;           Display RTC
;-----
rtc_display: mov     r2,#85h
            lcall  rtc_read_ch
            mov     r0,#b_second
            lcall  htoa
            mov     @r0,#3ah
            inc     r0

            mov     r2,#83h
            lcall  rtc_read_ch
            lcall  htoa
            mov     @r0,#3ah
            inc     r0

            mov     r2,#81h
            lcall  rtc_read_ch
            lcall  htoa
            lcall  rtc_read_ch
            mov     r2,#87h

            lcall  htoa

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

mov     @r0,#2fh
inc     r0

mov     r2,#89h
lcall  rtc_read_ch
lcall  htoa
mov     @r0,#2fh
inc     r0

mov     r2,#8dh
lcall  rtc_read_ch
lcall  htoa

lcall  display
ret

;-----
; Write Command and parameter byte
; IN: R2 = COMMAND, R3 = DATA
;-----
rtc_write_ch:    clr     rtc_clk
                lcall  delay
                setb  rtc_rst
                lcall  delay
                mov   a,r2
                lcall  rtc_write_8bit
                mov   a,r3
                lcall  rtc_write_8bit
                clr   rtc_rst
                lcall  delay
                ret

;-----
; Transfer 8-bits data to DS1302
; IN: A = data
;-----
rtc_write_8bit:  mov   r4,#08h
wr8bit1:        rrc   a
                mov   rtc_data,c
                setb  rtc_clk
                lcall  delay
                clr   rtc_clk
                lcall  delay
                djnz  r4,wr8bit1
                ret

;-----
; READ Command and read parameter byte
; IN: R2 = COMMAND
; OUT: R3 = DATA
;-----
rtc_read_ch:    clr     rtc_clk
                lcall  delay
                setb  rtc_rst
                lcall  delay
                mov   a,r2
                lcall  rtc_write_8bit

;
                mov   r4,#8
                clr   a

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

rd_ch1:          clr          rtc_clk
                 lcall       delay
                 mov         c,rtc_data
                 rrc         a
                 setb       rtc_clk
                 lcall       delay
                 djnz       r4,rd_ch1
                 mov         r3,a

                 clr          rtc_rst
                 lcall       delay
                 ret

;-----
;          Display Time
;-----
display:        mov         r2,#08h
                 mov         a,#08Ch
                 lcall       writeinst
lcd1:           mov         r0,#50h
                 mov         a,@r0
                 lcall       writechar
                 inc         r0
                 djnz       r2,lcd1

                 mov         r2,#08h
                 mov         a,#0c0h
                 lcall       writeinst
lcd2:           mov         r0,#58h
                 mov         a,@r0
                 lcall       writechar
                 inc         r0
                 djnz       r2,lcd2
                 ret

;-----
;          Change Hex to ASCII
;-----
htoa:          mov         a,r3
                 swap        a
                 anl         a,#0fh
                 clr         c
                 add         a,#30h
                 mov         @r0,a
                 mov         a,r3
                 anl         a,#0fh
                 clr         c
                 add         a,#30h
                 inc         r0
                 mov         @r0,a
                 inc         r0
                 ret

;-----
;          Delay
;-----
delay:         nop
                 ret

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซอร์สโค้ดของไอซีเบอร์ AT89S8252

```

record          equ          01h
erase           equ          02h
stop            equ          03h
read            equ          04h
flash_full     equ          55h
;***** SPI *****
spcr            equ          0d5h
spsr            equ          0aah
spif            equ          1000000b
spdr            equ          86h
cs              bit          p1.4
mosi            bit          p1.3
miso            bit          p1.6
sck             bit          p1.7
;*****
wmcon           equ          96h
eemen           equ          00001000b
eemwe           equ          00010000b
wdrst           equ          00000010b
;*****
by1             bit          p3.5
a16             bit          p1.0
a17             bit          p1.1
a18             bit          p1.2
a19             bit          p1.3
a20             bit          p3.2
address_hh2     equ          40h
address_hh      equ          41h
address_h       equ          42h
address_l       equ          43h
data_1          equ          44h
;*****

                org          0000h
                ljmp         start

                org          000bh
                ljmp         int_timer

;*****

                org          0023h
                jb           ri,recive
                clr          ti
                push         psw
                push         acc

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

                                reti

recvie:                          mov     a,sbuf
                                clr     ri
                                cjne   a,#record,recvie_1
recvie_1:                        ljmp   recvie_record
                                cjne   a,#erase,recvie_2
                                ljmp   recvie_erase

recvie_2:                        cjne   a,#read,recvie_3
                                ljmp   recvie_read
recvie_3:                        push  psw
                                push  acc
                                reti

;*****
start:                            lcall  delayis
                                mov    scon,#50h
                                setb  ea
                                setb  es
                                mov    tmod,#21h
                                mov    th1,#0fdh
                                mov    tl1,#0fdh
                                setb  tr1
                                lcall  delayls

                                orl    wmcon,#eemen
                                orl    wmcon,#eemwe

                                mov    dptr,#00h
                                movx   a,@dptr
                                cjne   a,#0ffh,loop_4
                                mov    a,#00h
loop_1:                          movx   @dptr,a
                                mov    a,wmcon
                                anl    a,#wdtrst
                                jz     loop_1
                                inc    dptr
                                mov    a,#0Ch
loop_2:                          movx   @dptr,a
                                mov    a,wmcon
                                anl    a,#wdtrst
                                jz     loop_2
                                inc    dptr
                                mov    a,#00h
loop_3:                          movx   @dptr,a
                                mov    a,wmcon
                                anl    a,#wdtrst
                                jz     loop_3

loop_4:                          mov    dptr,#00h
                                movx   a,@dptr
                                mov    address_hh,a
                                inc    dptr
                                movx   a,@dptr
                                mov    address_h,a

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

inc      dptr
movx    a,@dptr
mov     address_1,a

xrl     wmcon,#eenwe
xrl     wmcon,#eemen

sjmp    $

;*****
recvie_record:  clr     es
mov     sbuf,a
jnb     ti,$
clr     ti
jnb     ri,$
clr     ri
mov     data_1,sbuf
lcall  write_data
mov     sbuf,data_1
jnb     ti,$
clr     ti
jnb     ri,$
clr     ri
mov     data_1,sbuf
lcall  write_data
mov     sbuf,data_1
jnb     ti,$
clr     ti
jnb     ri,$
clr     ri
mov     data_1,sbuf
lcall  write_data
mov     sbuf,data_1
jnb     ti,$
clr     ti
jnb     ri,$
clr     ri
mov     data_1,sbuf
lcall  write_data
mov     sbuf,data_1
jnb     ti,$
clr     ti
jnb     ri,$
clr     ri
mov     data_1,sbuf
lcall  write_data
mov     sbuf,data_1
jnb     ti,$
clr     ti
jnb     ri,$
clr     ri
mov     data_1,sbuf
lcall  write_data
mov     sbuf,data_1
jnb     ti,$
clr     ti

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        mov     th0,#0f6h
        mov     t10,#00h
        setb   et0
        setb   tr0
        reti

;*****
recive_erase:    clr     es
                 mov     r7,a
                 mov     sbuf,a
                 jnb    ti,$
                 clr     ti
                 lcall   chip_erase
                 mov     sbuf,r7
                 jnb    ti,$
                 clr     ti
                 setb   es
                 reti

;*****
recive_read:     clr     es
                 icall  read_data
                 setb   es
                 reti

;*****
int_timer:      mov     th0,#0f6h
                 mov     t10,#0Ch
                 setb   cs
                 setb   mosi
                 setb   miso
                 setb   sck
                 mov     spcr,#01010010b
                 clr     cs
                 mov     a,#01h           ;ch0
                 lcall  master_io
                 mov     a,#10100000b
                 lcall  master_io
                 andl   a,#0fh
                 mov     r6,a
                 mov     a,#0ffh
                 lcall  master_io
                 mov     r5,a

                 setb   cs
                 setb   mosi
                 setb   miso
                 setb   sck
                 mov     spcr,#01010010b
                 clr     cs
                 mov     a,#01h           ;ch0
                 lcall  master_io
                 mov     a,#11100000b
                 lcall  master_io
                 arl    a,#0fh
                 mov     r4,a
                 mov     a,#0ffh

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

                                lcall    master_ic
                                mov      r3,a

                                mov      a,address_hh
                                cjne     a,#01fh,int_loop1
                                mov      a,address_h
                                cjne     a,#0ffh,int_loop1
                                mov      a,address_l
                                and      a,#0f0h
                                cjne     a,#0f0h,int_loop1
                                mov      sbuf,#055h
                                jnb      ti,$
                                clr      ti
                                ljmp     int_break

int_loop1:
                                mov      data_l,r6
                                lcall    write_data
                                mov      data_l,r5
                                lcall    write_data
                                mov      data_l,r4
                                lcall    write_data
                                mov      data_l,r3
                                lcall    write_data
                                jb       ri,chk_st
                                reti

chk_st:
                                mov      a,sbuf
                                clr      ri
                                cjne     a,#stop,int_out
int_break:
                                clr      tr0
                                mov      th0,#0f6h
                                mov      tl0,#00h

                                mov      data_l,#55h
                                lcall    write_data
                                mov      data_l,#55h
                                lcall    write_data
                                mov      data_l,#55h
                                lcall    write_data
                                mov      data_l,#55h
                                lcall    write_data
                                mov      data_l,#55h
                                lcall    write_data
                                mov      data_l,#55h
                                lcall    write_data

                                orl      wmcon,#eemen
                                orl      wmcon,#eemwe

                                mov      dptr,#000h
                                mov      a,address_hh
                                movx     @dptr,a
int_loop4:
                                mov      a,wmcon
                                and     a,#wdtrst
                                jz      int_loop4
                                inc     dptr
                                mov      a,address_h
                                movx     @dptr,a
int_loop5:
                                mov      a,wmcon

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

                                anl        a,#wdtrst
                                jz          int_loop5
                                inc         dptr
                                mov         a,address_1
int_loop6:                       movx      @dptr,a
                                mov         a,wmccon
                                anl        a,#wdtrst
                                jz          int_loop6
                                mov        sbuf,#03h
                                jnb        ti,$
                                clr        ti

                                xrl        wmccon,#eemwe
                                xrl        wmccon,#eemen

int_out:                          setb     es
                                reti

master_io:                          mov      spdr,a
spi_flag:                          mov      a,spdr
                                anl      a,#spif
                                jz       spi_flag
                                mov      a,spdr
                                ret

;*****
;*****
write_data:                          clr      a16
                                clr      a17
                                clr      a18
                                clr      a19
                                clr      a20
                                mov      dptr,#0555h
                                mov      a,#0aah
                                movx     @dptr,a
                                mov      dptr,#02aah
                                mov      a,#055h
                                movx     @dptr,a
                                mov      dptr,#0555h
                                mov      a,#0a0h
                                movx     @dptr,a
                                mov      cpl,address_1
                                mov      dph,address_h
                                mov      a,address_hh
                                mov      c,acc.0
                                mov      a16,c
                                mov      c,acc.1
                                mov      a17,c
                                mov      c,acc.2
                                mov      a18,c
                                mov      c,acc.3
                                mov      a19,c
                                mov      c,acc.4
                                mov      a20,c
                                mov      a,data_1
                                movx     @dptr,a
                                jnb      by1,$

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

inc            dptr
mov            address_l,dpl
mov            address_h,dph
mov            a,address_l
cjne          a,#0Ch,add_hh2
mov            a,address_h
cjne          a,#0Ch,add_hh2
inc            address_hh
add_hh2:      ret
;*****
;*****
chip_erase:   clr            a16
              clr            a17
              clr            a18
              clr            a19
              clr            a20
              mov            dptr,#0555h
              mov            a,#0aah
              movx           @dptr,a
              mov            dptr,#02aah
              mov            a,#055h
              movx           @dptr,a
              mov            dptr,#0555h
              mov            a,#080h
              movx           @dptr,a
              mov            dptr,#0555h
              mov            a,#0aah
              movx           @dptr,a
              mov            dptr,#02aah
              mov            a,#055h
              movx           @dptr,a
              mov            dptr,#0555h
              mov            a,#010h
              movx           @dptr,a
              nop
              nop
              nop
              jnb            by1,$

              mov            address_hh,#00h
              mov            address_h,#00h
              mov            address_l,#00h

              orl            wmcn,#eemen
              orl            wmcn,#eemwe
              mov            a,#00h
              movx           @dptr,a
erase_loop_1: mov            a,wmcn
              anl            a,#wdtrst
              jz             erase_loop_1
              inc            dptr
              mov            a,#00h
              movx           @dptr,a
erase_loop_2: mov            a,wmcn
              anl            a,#wdtrst
              jz             erase_loop_2

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

inc            dptr
mov            a,#00h
movx          @dptr,a
erase_loop_3:  mov            a,wmcon
               ani            a,#wdtrst
               jz             erase_loop_3

               xrl            wmcon,#eemwe
               xrl            wmcon,#eemer.
               ret

;*****
;*****
read_data:    mov            sbuf,address_hh
               jnb           ti,$
               clr            ti
               mov            sbuf,address_h
               jnb           ti,$
               clr            ti
               mov            sbuf,address_l
               jnb           ti,$
               clr            ti

               mov            dptr,#0000h
               mov            address_hh2,#00h
               jnb           ri,$
               clr            ri

read_loop:    mov            a,address_hh2
               mov            c,acc.0
               mov            a16,c
               mov            c,acc.1
               mov            a17,c
               mov            c,acc.2
               mov            a18,c
               mov            c,acc.3
               mov            a19,c
               mov            c,acc.4
               mov            a20,c
               movx          a,@dptr
               mov            sbuf,a
               jnb           ti,$
               clr            ti
               mov            a,address_hh2
               cjre          a,address_hh,read_loop_1
               mov            a,dph
               cjne          a,address_h,read_loop_1
               mov            a,dpl
               cjne          a,address_l,read_loop_1
               ret

read_loop_1:  inc            dptr
               mov            a,dpi
               cjne          a,#00h,chk_add
               mov            a,dph
               cjne          a,#00h,chk_add
               inc            address_hh2
chk_add:      mov            a,address_hh2
               cjne          a,address_hh,read_loop

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

mov          a,dph
cjne        a,address_h,read_loop
mov         a,dpl
cjne        a,address_l,read_loop
ret

;+++++-----+++++-----+++++-----+++++-----+++++-----+++++-----+++++
delayls:    mov          r5,#07h
            mov          r6,#00h
            mov          r7,#00h
            djnz         r7,$
            djnz         r6,$-2
            djnz         r5,$-4
            ret

;+++++-----+++++-----+++++-----+++++-----+++++-----+++++-----+++++
delay500ms: mov          r5,#04h
            mov          r6,#00h
            mov          r7,#00h
del500:     djnz         r7,$
            djnz         r6,del500
            djnz         r5,del500
            ret

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้