

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบแสดงสัญญาณไฟฟ้าหัวใจ

ECG SIGNAL SYSTEM



เลขหมู่.....  
เลขทะเบียน..... 73182  
วันเดือนปี..... 10 ก.ค. 2550

b. 112 55009  
.....

ปฏิญานិพนธ์เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต  
สาขาวิชาอิเล็กทรอนิกส์  
สถาบันเทคโนโลยีพระจอมเกล้าคุณทหารลาดกระบัง  
ปีการศึกษา 2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ระบบแสดงสัญญาณไฟฟ้าหัวใจ

## ECG SIGNAL SYSTEM



โดย

นายวิศิษฐ์	สุธาเบญจประดิษฐ์	45010723
นายวีรพงษ์	ละโลงสูงเนิน	45010724
นายวีรพล	รัตตนิเชียร	45010725

อาจารย์ที่ปรึกษา

รศ.ดร.สุริกานต์ สมควรพาณิชย์

ปริญญาานิพนธ์สำหรับปริญญาวิศวกรรมศาสตรบัณฑิต  
สาขาวิชาอิเล็กทรอนิกส์  
คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ระบบแสดงสัญญาณไฟฟ้าหัวใจ

นายวิศิษฐ์ สุทธาญาประดิษฐ์ รหัส 45010723

นายวีรพงษ์ ละโลงสูงเนิน รหัส 45010724

นายวีรพล รัตตนิเชียร รหัส 45010725

ผศ.ดร.สุริภณ สมควรพาณิชย์ (อาจารย์ที่ปรึกษา)

ปีการศึกษา 2548

## บทคัดย่อ

เป็นการศึกษาการสร้างเครื่องจำลองและแสดงผลของสัญญาณหัวใจแบบปรกติทั้ง 12 แบบที่เราเก็บเป็นฐานข้อมูลไว้ใน Memory (EEPROM) โดยเราใช้ Microcontroller ตระกูล PIC เป็นตัวควบคุมการติดต่อกับ Memory (EEPROM) , ควบคุมการแสดงผลทาง Oscilloscope และส่งค่าจาก Memory (EEPROM) ไปแสดงผลทางจอคอมพิวเตอร์โดยโปรแกรม Visual Basic 6 มีส่วนรับสัญญาณคลื่นไฟฟ้าหัวใจที่เป็นสัญญาณอนาล็อกแล้วทำการแปลงค่าสัญญาณเพื่อนำไปแสดงผลทางคอมพิวเตอร์และนำค่าสัญญาณนี้มาทำการวิเคราะห์ว่าสัญญาณหัวใจที่รับมีความผิดปกติหรือไม่ เครื่องมือนี้ทำเพื่อเป็นสื่อที่ช่วยในการเรียนการสอนทางการแพทย์ และผู้ที่มีความสนใจในเรื่องสัญญาณไฟฟ้าหัวใจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ECG SIGNAL SYSTEM

Mr. Wisit Suthabenchapardit ID.45010723

Mr. Weerapong Lalongsoongneon ID.45010724

Mr. Weerapon Rattanavichain ID.45010725

Assoc. Prof. Dr. Suripon Somkuanparnit (Advisor)

Educational Year 2005

## Abstract

This project is a study of Simulate machine invention and to perform Normal ECG 12 Leads. We stored the Normal ECG 12 Leads in memory by using Micro-controller, PIC family. The Micro-controller controls memory communication, and display result by scope. Moreover, Micro-controller can deliver value from memory to perform result at computer from VB. 6 program. Hardware have received analog input. Analog signal is shown the result at computer from VB. 6 program. ECG wave from then analyze the heart signals are normal or not.

The purpose of ECG Project is not only to study the Simulate machine and Normal ECG 12 Leads, but also be a media to help medical studying and teaching, and for people who are interested in heart signal.

ปริญญาโทปีการศึกษา 2548  
ภาควิชา อิเล็กทรอนิกส์  
คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าคุณทหารลาดกระบัง  
เรื่อง ระบบแสดงสัญญาณหัวใจHEART SIGNAL MONITORING  
ผู้จัดทำ

1. นายวิสิทธิ์ สุทธาเบญาประดิษฐ์ 45010723
2. นายวีรพงษ์ ละโลงสูงเนิน 45010724
3. นายวีรพล รัตตนิเวียร์ 45010725



รศ.ดร.สุริภณ สมควรพาณิชย์ อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## กิตติกรรมประกาศ

โครงการชิ้นนี้สำเร็จลุล่วงไปได้ด้วยดี ขอขอบพระคุณ รศ.ดร.สุริภณ สมควรพาณิชย์ อาจารย์ที่ปรึกษา เป็นอย่างสูง ที่ให้ความรู้ คำปรึกษา และคำแนะนำต่างๆ ในโครงการ รวมทั้งอุปกรณ์ต่างๆ และสถานที่สำหรับทำโครงการนี้ ขอขอบคุณเพื่อนๆทุกคน และผู้เกี่ยวข้อง ที่ให้ความช่วยเหลือในด้านต่างๆเป็นอย่างดี



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ

บทที่ 1	บทนำ	
	คลื่นไฟฟ้าหัวใจ	1
บทที่ 2	ความรู้เบื้องต้นเกี่ยวกับคลื่นไฟฟ้าหัวใจ (ECG)	2
	2.1 หลักการวิเคราะห์หาค่าลักษณะของสัญญาณไฟฟ้าหัวใจ	5
	2.2 Lead แบบต่างๆในการวัด ECG	7
บทที่ 3	โครงสร้างโดยรวมของระบบ	9
	3.1 โครงสร้างทางฮาร์ดแวร์	9
	3.2 The Serial Port	14
	3.3 I2C BUS	16
	3.4 D/A Conversion	17
	3.4.1 Binary Weighted DAC	17
	3.4.2 R/2R Ladder	18
บทที่ 4	โครงสร้างทางซอฟต์แวร์	19
	4.1 ข้อมูลของกราฟ Normal ECG 12 Leads	19
	4.2 การทำงานของโปรแกรม Write Data in EEprom	23
	4.3 การทำงานของโปรแกรม VB Normal 12 Leads ECG	25
	ขั้นตอนการใช้งานโปรแกรม	26
บทที่ 5	ผลการทดลอง	30
	5.1 ผลการทดลอง Output (V) ของ DAC (DAC0808)	30
	5.2 ผลการทดลองการติดต่อกับ EEprom ด้วย Bus I2C	31
	5.3 กราฟการเปรียบเทียบการแสดงผลระหว่าง Oscilloscope และ Program VB	33
	5.4 ผลการทดลองการวิเคราะห์สัญญาณ ECG	45
	5.5 ผลการทดลองการปรับ Heart Rate	47
บทที่ 6	สรุปผลการทดลอง	50
	กิตติกรรมประกาศ	
	หนังสืออ้างอิง	
	ภาคผนวก	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป

รูปที่ 2.1	แสดงคลื่นไฟฟ้าหัวใจโดยมี Iso-electric line แบ่งระหว่าง cycle	2
รูปที่ 2.2	แสดงส่วนต่างๆของคลื่นไฟฟ้าหัวใจ	3
รูปที่ 2.3	พารามิเตอร์ของสัญญาณไฟฟ้าหัวใจ	5
รูปที่ 2.4	กิจกรรมทางไฟฟ้าที่เกิดขึ้นที่บริเวณต่างๆในหัวใจ	6
รูปที่ 2.5	แสดงจุดการวัดคลื่นไฟฟ้าหัวใจและสามเหลี่ยมของ Eindhoven	7
รูปที่ 2.6.1	แสดงจุดการวัดคลื่นไฟฟ้าหัวใจที่ตำแหน่งต่างๆ	7
รูปที่ 2.6.2	แสดงจุดการวัดคลื่นไฟฟ้าหัวใจที่ตำแหน่งต่างๆ	8
รูปที่ 2.7	Electrocardiography	8
รูปที่ 3.1.1	วงจรเรียงกระแสแบบเต็มคลื่น	9
รูปที่ 3.1.2	วงจร Regulator +15V, -15V, 5V	9
รูปที่ 3.1.3	บล็อกไดอะแกรมของวงจรในการทำงาน Mode 1 และ Mode 2	10
รูปที่ 3.2	โครงสร้าง PIC16F877	10
รูปที่ 3.3	Block Diagram ที่สำคัญของ Port B	11
รูปที่ 3.4	Block Diagram และ Function ที่สำคัญของ Port C	13
รูปที่ 3.5	Block Diagram และ Function ที่สำคัญของ Port D	13
รูปที่ 3.6	Block Diagram และ Function ที่สำคัญของ Port E	14
รูปที่ 3.7	Serial Port แบบ DB9	15
รูปที่ 3.8	การแปลงสัญญาณดิจิตอลเป็นสัญญาณอนาล็อก	17
รูปที่ 3.9	แสดงวงจร Binary Weighted DAC	17
รูปที่ 3.10	แสดงวงจร R/2R Ladder	18
รูปที่ 4.1	กราฟ Normal ECG 12 Leads	19 - 22
รูปที่ 4.2.1	โปรแกรม Write Data in EEprom	23
รูปที่ 4.2.2	โปรแกรม Write Data in EEprom	24
รูปที่ 4.2.3	โปรแกรม Write Data in EEprom	24
รูปที่ 4.3	แสดงโหมดการทำงานของวงจร	25
รูปที่ 4.3.1	โปรแกรม VB Normal 12 Leads ECG	26
รูปที่ 4.3.2	โปรแกรม VB Normal 12 Leads ECG	27
รูปที่ 4.3.3	โปรแกรม VB Normal 12 Leads ECG	28
รูปที่ 4.3.4	โปรแกรม VB Normal 12 Leads ECG	29

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.3.4 โปรแกรม VB Normal 12 Leads ECG	29
รูปที่ 5.1 กราฟแสดงความสัมพันธ์ I/P และ O/P ของ DAC0808	30
รูปที่ 5.2 แสดงผลการส่งข้อมูล 1 และ ข้อมูล 0 ในระบบบัส I <sup>2</sup> C	31
รูปที่ 5.3 แสดงผลการส่งข้อมูล 1 ชุดให้กับ EEPROM ในระบบบัส I <sup>2</sup> C	32
รูปที่ 5.4 แสดงผลการส่งข้อมูลในระบบบัส I <sup>2</sup> C หลายชุดข้อมูล	32
รูปที่ 5.5.1 กราฟการเปรียบเทียบการแสดงผลระหว่าง Oscilloscope และ Program VB	33
รูปที่ 5.5.2 กราฟการเปรียบเทียบการแสดงผลระหว่าง Oscilloscope และ Program VB	34
รูปที่ 5.5.3 กราฟการเปรียบเทียบการแสดงผลระหว่าง Oscilloscope และ Program VB	35
รูปที่ 5.5.4 กราฟการเปรียบเทียบการแสดงผลระหว่าง Oscilloscope และ Program VB	36
รูปที่ 5.5.5 กราฟการเปรียบเทียบการแสดงผลระหว่าง Oscilloscope และ Program VB	37
รูปที่ 5.5.6 กราฟการเปรียบเทียบการแสดงผลระหว่าง Oscilloscope และ Program VB	38
รูปที่ 5.5.7 กราฟการเปรียบเทียบการแสดงผลระหว่าง Oscilloscope และ Program VB	39
รูปที่ 5.5.8 กราฟการเปรียบเทียบการแสดงผลระหว่าง Oscilloscope และ Program VB	40
รูปที่ 5.5.9 กราฟการเปรียบเทียบการแสดงผลระหว่าง Oscilloscope และ Program VB	41
รูปที่ 5.5.10 กราฟการเปรียบเทียบการแสดงผลระหว่าง Oscilloscope และ Program VB	42
รูปที่ 5.5.11 กราฟการเปรียบเทียบการแสดงผลระหว่าง Oscilloscope และ Program VB	43
รูปที่ 5.5.12 กราฟการเปรียบเทียบการแสดงผลระหว่าง Oscilloscope และ Program VB	44
รูปที่ 5.6.1 แสดงการรับสัญญาณ ECG Lead 2 (Display) เข้ามา	45
รูปที่ 5.6.2 แสดงการรับสัญญาณ ECG Lead 2 (Display) เข้ามาแล้วแสดงผลตรงส่วน Analysis ส่วน AV-node เมื่อสัญญาณมีระดับปกติ	45
รูปที่ 5.6.3 แสดงการรับสัญญาณ ECG Lead 2 (Display) เข้ามาแล้วแสดงผลตรงส่วน Analysis ส่วน SA-node เมื่อสัญญาณมีระดับปกติแต่สัญญาณมีความสูงลดลง	46
รูปที่ 5.6.4 แสดงการรับสัญญาณ ECG Lead 2 (Display) เข้ามาแล้วแสดงผลตรงส่วน Analysis ส่วน AV-node เมื่อสัญญาณมีระดับความสูงเกินค่าปกติ	46
รูปที่ 5.6.5 แสดงการรับสัญญาณ ECG Lead 2 (Display) เข้ามาแล้วแสดงผลตรงส่วน Analysis ส่วน SA-node เมื่อสัญญาณมีความต่ำกว่าค่าผิดปกติ	47
รูปที่ 5.7.1 แสดง Heart Rate = 60 Rate/Min	47
รูปที่ 5.7.2 แสดง Heart Rate = 75 Rate/Min	48
รูปที่ 5.7.3 แสดง Heart Rate = 100 Rate/Min	48
รูปที่ 5.7.4 แสดง Heart Rate = 120 Rate/Min	49

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญตาราง

ตารางที่ 3.1	แสดง PIN ของ Serial Port แบบ DB9	15
ตารางที่ 5.1	แสดงผลการทดลอง Output (V) ของ DAC(DAC0808)34	30



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

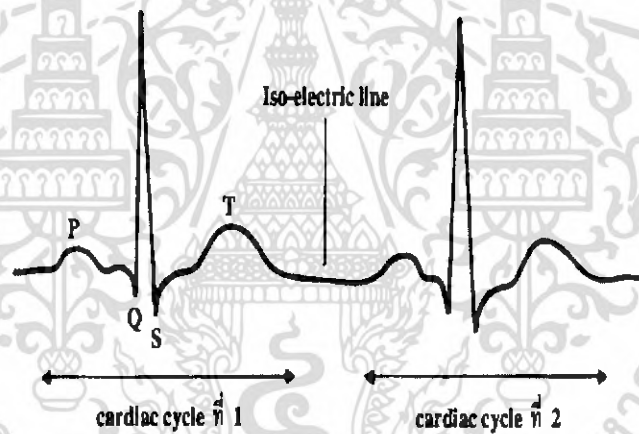
## บทที่ 1 บทนำ

คลื่นไฟฟ้าหัวใจ หรือกราฟหัวใจ หรือภาษาอังกฤษเรียกว่า **Electrocardiogram** ใช้คำย่อว่า **ECG หรือ EKG** ก็คือการตรวจจับกระแสไฟฟ้าที่ออกมาจากหัวใจ หัวใจคนเราเป็นอวัยวะมหัศจรรย์จริงๆ ประกอบไปด้วยส่วนของกล้ามเนื้อหัวใจที่แข็งแรง ทำงานตลอดชีวิตไม่มีวันหยุดพัก อวัยวะอื่นๆยังพักได้ แต่หัวใจไม่เคยพัก การที่กล้ามเนื้อหัวใจจะทำงานบีบตัวได้นั้นจะต้องอาศัยไฟฟ้ากระตุ้น ไฟฟ้านี้ก็มาจากหัวใจเอง โดยจะปล่อย ไฟฟ้าออกมาเป็นจังหวะ จากหัวใจห้องบนขวา ลงมายังหัวใจห้องล่าง ขณะที่ไฟฟ้าผ่านกล้ามเนื้อหัวใจ กล้ามเนื้อหัวใจจะเกิดการหดตัว (และตามมาด้วยการคลายตัว) หัวใจจึงบีบตัวไล่เลือดจากห้องบน มายังห้องล่าง อย่างสัมพันธ์กัน เมื่อเรานำเอาตัวจับสัญญาณ ไฟฟ้า (electrode) มาวางไว้ที่หน้าอก ใกล้หัวใจ เราก็สามารถบันทึกไฟฟ้าที่ออกจากหัวใจนี้ได้ การตรวจคลื่นไฟฟ้าหัวใจไม่ใช่เครื่องเช็กหัวใจ ไม่ใช่ตรวจออกมาเป็นกราฟแล้วจะบอกความผิดปกติ ทุกอย่าง เปล่าเลย ความเป็นจริงแล้วคลื่นไฟฟ้าหัวใจบอกให้เราทราบข้อมูลเกี่ยวกับหัวใจค่อนข้างจำกัด เช่น จังหวะการเต้น ความสม่ำเสมอ การนำไฟฟ้าในหัวใจ ชนิดของการเดินผิดจังหวะ (ตรวจขณะมีอาการ) หัวใจโตหรือไม่ กล้ามเนื้อหัวใจตาย กล้ามเนื้อหัวใจขาดเลือด (ในบางรายเท่านั้น เพราะต้องตรวจขณะมีอาการ) ความผิดปกติของระดับเกลือแร่บางชนิดในร่างกาย เป็นต้น ข้อมูลที่ได้มาก็ต้องนำมาแปลผลอีกครั้ง โดยอาศัยประวัติ การตรวจร่างกาย ความชำนาญของแพทย์ จึงจะสรุปอีกครั้งว่าคลื่นไฟฟ้า หัวใจผิดปกติหรือไม่ บ่อยครั้งที่ผู้ป่วยได้รับการบอกจากแพทย์ว่าเป็นโรคหัวใจโต หลอดเลือดตีบ 3 เส้น ทั้งๆที่ผู้ป่วยไม่ได้เป็นเลย เนื่องจากแพทย์ขาดความรู้ความชำนาญในการนำข้อมูลที่ได้จาก EKG มาแปลผลอย่างถูกต้อง คลื่นไฟฟ้าหัวใจปกติ ไม่ได้หมายความว่าหัวใจปกติ ปราศจากโรค ในโรคหัวใจขาดเลือด หรือ หลอดเลือดหัวใจตีบนั้น คลื่นไฟฟ้าหัวใจจะผิดปกติก็ต่อเมื่อเป็นโรคขั้นรุนแรงจนเกิดกล้ามเนื้อหัวใจ ตายแล้ว ถ้าเป็นเพียงหลอดเลือดตีบแต่ไม่รุนแรง ก็อาจตรวจไม่พบได้ นอกจากการตรวจคลื่นไฟฟ้าหัวใจขณะ ออกกำลังกายจะช่วยเพิ่มความไวในการตรวจขึ้น

## บทที่ 2 ความรู้เบื้องต้นเกี่ยวกับคลื่นไฟฟ้าหัวใจ (ECG)

Electrocardiogram เป็นชื่อเต็มที่รู้จักกันดีในอเมริกาด้วยชื่อย่อว่า EKG แต่สำหรับในยุโรป ออสเตรเลีย รวมทั้งนิวซีแลนด์แล้วพวกเขาเรียกกันในนาม ECG ซึ่งก็คือเรื่องเดียวกันนั่นเอง เพราะพวกอเมริกันเกรงว่าจะพูดหรือฟังผิด ระหว่างคลื่นสมอง (EEG) กับ คลื่นไฟฟ้าหัวใจ (ECG) นั่นเอง.

คลื่นไฟฟ้าหัวใจ ประกอบด้วยคลื่นที่สำคัญๆ 3 ส่วนด้วยกัน ได้แก่ P Wave, QRS Complex, T Wave ซึ่ง Eintoven เป็นผู้ตั้งชื่อทั้งหมดนี้รวมเป็น 1 cycle ของหัวใจ ซึ่ง P Wave เกิดจาก atrial depolarization, QRS Complex เกิดจาก ventricular depolarization และ T Wave เกิดจาก ventricular repolarization ดังนั้นจะเห็นได้ว่าหัวใจมี วงรอบทางไฟฟ้าอย่างสมบูรณ์ ซึ่งถ้าเปรียบเทียบกับทางสรีรวิทยาแล้วก็จะได้ว่าหัวใจมีการบีบตัวและคลายตัวเป็น physical cycle อย่างสมบูรณ์เช่นกัน



รูปที่ 2.1 แสดงคลื่นไฟฟ้าหัวใจโดยมี Iso-electric line แบ่งระหว่าง cycle

คลื่นต่างๆที่ได้ใน ECG นั้น อาจเป็นไปได้ทั้งบวกและลบ การที่จะเป็นคลื่นชนิดใดนั้นขึ้นอยู่กับตำแหน่งของ electrode ด้วย.

ในบางกรณีเราได้ทั้งส่วนที่เป็นบวกและลบภายในคลื่นเดียวกัน เราเรียกคลื่นชนิดนี้ว่า biphasic หรือ diphasic wave ซึ่งเกิดขึ้นได้เพราะ electrode อยู่ตรงจุดกึ่งกลางของเซลล์กล้ามเนื้อพอดี ดังนั้นคลื่นที่ได้ตอนแรกจะเป็นคลื่นบวก เพราะประจุบวกเคลื่อนที่ไปเรื่อยๆ ส่วนคลื่นตอนหลังซึ่งจะอยู่ใต้เส้นมาตรฐานเพราะเป็นผลของประจุลบที่ผ่านไปนั่นเอง.

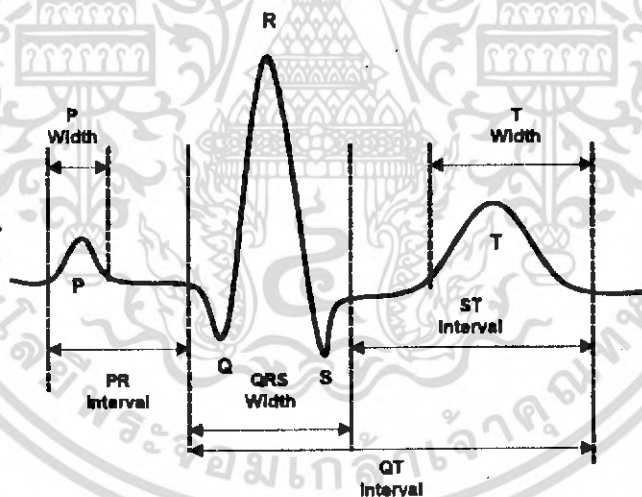
คลื่นแต่ละคลื่นของคลื่นไฟฟ้าหัวใจแยกออกจากกันด้วยเส้นมาตรฐานสั้น ๆ ซึ่งเรียกว่า segment ส่วนของ segment ที่มีความสำคัญมากได้แก่ ST segment ในบางครั้งการวัดความยาว

ของคลื่นอย่างเฉียว หรือความยาวของคลื่นรวมถึง segment มีประโยชน์มากในการอ่านคลื่นไฟฟ้าหัวใจ ระยะเช่นนี้เราเรียกว่า interval ซึ่ง interval เช่นนี้มีอยู่ 3 ชนิดที่สำคัญคือ

1. **P-R interval** ได้แก่ระยะเวลานับตั้งแต่จุดเริ่มต้นของ P wave ไปจนถึงจุดเริ่มต้นของ R wave ในกรณีที่มี Q wave ร่วมอยู่ด้วยใน QRS complex P-R interval จะเป็นระยะจากจุดเริ่มต้นของ P wave ไปจนถึงจุดเริ่มต้นของ Q wave ดังนั้นในกรณีเช่นนี้ ถ้าเราจะเรียกตามความเป็นจริงก็ต้องเรียกว่า P-Q interval ซึ่งมีผู้ใช้บ้างเหมือนกัน

2. **QRS interval** เป็นระยะเวลานับตั้งแต่จุดเริ่มต้นของ Q wave ไปจนถึงจุดสุดท้ายของ QRS complex ดังนั้นใน QRS interval จึงรวมคลื่นทุกคลื่นที่ประกอบเป็น QRS complex หมายความว่าในบางครั้ง QRS complex ไม่มี Q wave QRS interval จะเริ่มต้นจากจุดเริ่มต้นของ R wave เลย

3. **Q-T interval** เป็นระยะเวลาจากจุดเริ่มต้นของ QRS complex ไปจนถึงปลายสุดของ T wave ในบรรดา interval ทั้งหมด Q-T interval จะยาวมากที่สุด ประกอบด้วยความยาวของ 2 wave และ 1 segment นั่นคือ ความยาวของ QRS complex + ความยาว ST segment + ความยาวของ T wave



รูปที่ 2.2 แสดงส่วนต่างๆของคลื่นไฟฟ้าหัวใจ

ในปัจจุบันคลื่นไฟฟ้าหัวใจไม่ได้ประกอบด้วยกราฟเพียงแถวเดียวอย่างที่กล่าวมาแล้ว หากแต่ประกอบด้วยกราฟต่างๆรวมกันถึง 12 แถวด้วยกัน ซึ่งแต่ละแถวนั้นเราเรียกว่า Lead ดังนั้นคลื่นไฟฟ้าหัวใจที่สมบูรณ์จะต้องประกอบด้วย 12 Lead เสมอ แต่ละ lead เกิดจากการวาง electrode ซึ่งเป็นขั้วบวกและขั้วลบไว้บนที่ต่างๆของร่างกาย ซึ่งทั้ง 12 lead นี้สามารถแบ่งออกเป็น 2 ชนิดด้วยกันคือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**1. Standard lead หรือ Bipolar lead หรือ limb lead** เกิดจากการวาง electrode ไว้บนแขนและขา คือ electrode อันหนึ่งอยู่บนแขนขวา อีกอันหนึ่งอยู่บนแขนซ้าย และอันหนึ่งอยู่บนขาซ้ายหรือเท้าซ้าย แต่เรามักจะวางไว้บนขาซ้ายเพราะสะดวกกว่า และคลื่นไฟฟ้าที่ได้ก็จะมีรูปร่างเหมือนกันไม่ว่าเราจะวางไว้บนเท้าซ้ายหรือขาซ้าย คลื่นไฟฟ้าที่ได้นี้เป็นการวัดความต่างศักย์ระหว่างขั้วทั้งสองของ electrode ที่วางไว้ตามจุดต่าง ๆ จากการทำความเข้าใจทำให้เราได้คลื่นไฟฟ้า 3 lead ด้วยกันคือ

Lead I ได้จากการวัดความต่างศักย์ระหว่างแขนขวาและแขนซ้าย

Lead II ได้จากการวัดความต่างศักย์ระหว่างแขนขวาและขาซ้าย

Lead III ได้จากการวัดความต่างศักย์ระหว่างแขนซ้ายและขาซ้าย

จะเห็นได้ว่า standard lead นี้ เป็นการวัดความต่างศักย์ระหว่างจุด 2 จุดด้วยกัน และ electrode ที่จุดหนึ่งจะเป็นขั้วบวก electrode ที่อีกจุดหนึ่งจะเป็นขั้วลบ ดังนั้นเราจึงอาจเรียก lead ที่ได้จากการวาง electrode เช่นนี้ว่า bipolar lead แต่เนื่องจาก electrode ทั้ง 2 ขั้วนี้จะต้องวางอยู่บนแขน 2 ข้างหรือแขนกับขา เราจึงอาจเรียก lead ที่ได้ว่าเป็น limb lead บางคนนิยมเรียกรวมไปเลยว่าเป็น bipolar limb lead เพื่อให้แตกต่างจาก unipolar limb lead ซึ่งจะกล่าวต่อไป

**2. Unipolar lead** เกิดจากการวาง electrode ที่เป็นขั้วบวกไว้ตามตำแหน่งมาตรฐานต่าง ๆ ส่วน electrode อีกขั้วหนึ่งของ galvanometer นั้นทำให้มีค่าเป็นศูนย์คือเป็น zero potential หรือเรียกง่าย ๆ ว่าเป็น indifferent electrode ซึ่งก็อยู่ที่ตัวเครื่องมือนั่นเอง การวาง electrode เช่นนี้ทำให้เราสามารถวัด voltage ที่เกิดขึ้นจริง ๆ ได้ electrode ขั้วบวกได้ตามตำแหน่งต่าง ๆ ที่เราต้องการทราบโดยเราสามารถเคลื่อนย้าย electrode ขั้วบวกไปตามที่ต้องการ แต่ให้ indifferent electrode อยู่กับที่ด้วยเหตุผลข้อนี้บางคนจึงเรียก electrode ขั้วบวกว่า exploring electrode จุดมาตรฐานที่เราใช้

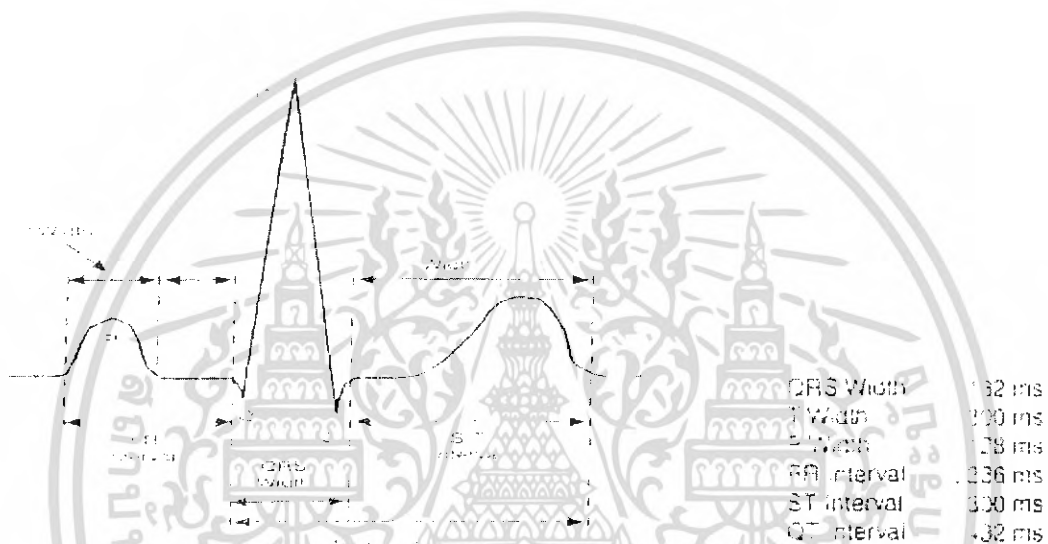
1. หัวไหล่ขวาซึ่งอาศัย electrode ที่ติดกับแขนขวาในการทำ standard lead นั้นเอง
2. หัวไหล่ซ้ายซึ่งอาศัย electrode ที่ติดกับแขนซ้าย
3. เท้าซ้ายซึ่งอาศัย electrode ที่ติดกับขาซ้าย

ยังมี unipolar lead อีกชนิดหนึ่งคือ chest lead หรือ precordial lead ซึ่งได้จากการวาง exploring electrode ลงบนตำแหน่งต่าง ๆ บนหน้าอก ได้แก่การวางให้ electrode อยู่บนหัวใจและรอบ ๆ หัวใจด้วย ตามมาตรฐานสากลการวางจะมี 6 ตำแหน่งด้วยกัน ทำให้ได้ chest lead 6 lead ซึ่งตามภาษาคลื่นไฟฟ้าหัวใจเราใช้ตัวย่อว่า V lead

## 2.1 หลักการวิเคราะห์คุณสมบัติของสัญญาณไฟฟ้าหัวใจ

**ECG Signal :** คือสัญญาณไฟฟ้าหัวใจที่นำมาใช้ในการวิเคราะห์ ในที่นี้นำมาจากฐานข้อมูลของสถาบัน MIT/BIH โดยสัญญาณนี้ได้ผ่านการแซมปลิงด้วยความถี่ 250 Hz

**ECG Parameters Identification :** ทำหน้าที่หาค่าพารามิเตอร์ของสัญญาณไฟฟ้าหัวใจ จากค่าสัมประสิทธิ์ของเวฟเล็ท เนื่องจากพารามิเตอร์ของสัญญาณไฟฟ้าหัวใจ (จุด P, Q, R, S, T, QRS Width, P-R Interval, ST Interval, P Width และ T Width) ดังแสดงในรูปที่ 3 บ่งบอกถึงสภาวะการทำงานของหัวใจ )



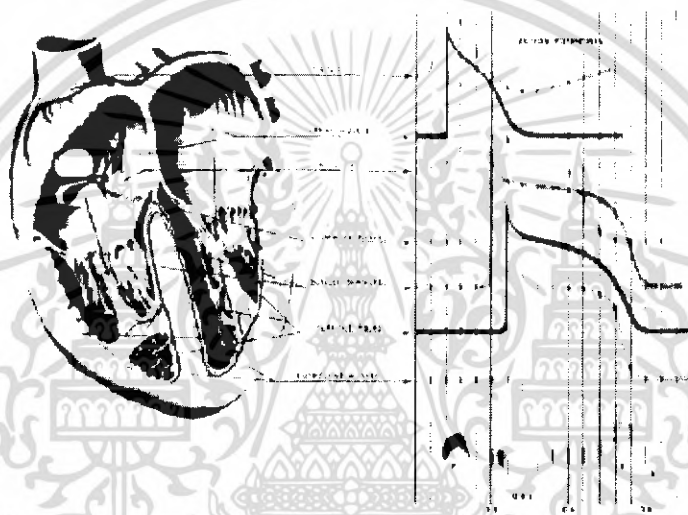
รูปที่ 2.3 พารามิเตอร์ของสัญญาณไฟฟ้าหัวใจ

**รูปคลื่น QRS (รูปคลื่น R) :** พิจารณาองค์ประกอบความถี่ของรูปคลื่น QRS พบว่ามีค่าประมาณ 3 Hz ถึง 40 Hz ซึ่งสอดคล้องกับองค์ประกอบความถี่ของสเกลที่ 3 และ 4 นำค่าสัมประสิทธิ์ของเวฟเล็ทที่สเกลสูงสุดก่อน มาทำการเปรียบเทียบกับระดับเทรสโฮล หลังจากนั้นทำการเก็บค่าตำแหน่งต่าง ๆ ที่อยู่ในช่วงนี้ ทำเช่นเดียวกันกับที่ค่าสเกลรองลงมา หลังจากนั้นทำการตรวจวัดหาชุดของข้อมูลที่ประกอบด้วย รูปคลื่นไบเฟซิก (Biphasic Waveform), จุดตัดศูนย์ของแต่ละสเกล โดยที่ตำแหน่งที่เกิดจุดตัดศูนย์นี้จะสอดคล้องกับตำแหน่งของรูปคลื่น R

**QRS Onset QRS Offset และความกว้างของรูปคลื่น QRS :** เนื่องจากค่า QRS Onset คือตำแหน่งเริ่มต้นที่เกิดรูปคลื่น QRS และค่า QRS Offset คือตำแหน่งที่สิ้นสุดรูปคลื่น QRS ดังนั้นความกว้างของรูปคลื่น QRS สามารถคำนวณได้จากค่าทั้งสอง การวิเคราะห์ค่าทั้งสองทำได้ยากเนื่องจากการรบกวนด้วยสัญญาณที่มีความถี่สูง แต่การวิเคราะห์ก็สามารถทำได้ที่สเกลค่าต่ำ จะพบว่าค่า Onset สามารถหาได้จากค่าเริ่มต้นของรูปคลื่นไบเฟซิก และค่า Offset สามารถหาได้จากค่าสิ้นสุดของรูปคลื่นไบเฟซิกนั่นเอง

**รูปคลื่น P และรูปคลื่น T :** พิจารณาองค์ประกอบความถี่ของรูปคลื่น P และรูปคลื่น T มีค่าประมาณ 0.5 Hz ถึง 10 Hz ซึ่งสอดคล้องกับองค์ประกอบความถี่ของสเกลที่ประมาณ 6 แต่เนื่องจากองค์ประกอบความถี่ในช่วงนี้จะค่อนข้างตรงกับ Motion artifact (ประมาณ 0.5 Hz ถึง 7 Hz) ดังนั้นในการพิจารณารูปคลื่น P และ T จึงเป็นงานที่ยากเนื่องจากการรบกวนด้วย Motion artifact นั้นเอง

**ระยะห่าง PR ระยะห่าง ST และระยะห่าง QT :** ค่าระยะห่าง PR คือช่วงเวลาค่า Onset ของรูปคลื่น P ถึง Onset ของรูปคลื่น R ค่าระยะห่าง ST คือช่วงเวลาค่า Offset ของรูปคลื่น S ถึงค่า Offset ของรูปคลื่น T ค่าระยะห่าง QT หาได้จากผลต่างระหว่างค่า Onset ของรูปคลื่น Q ถึงค่า Offset ของรูปคลื่น T ซึ่งนิยามของพารามิเตอร์เหล่านี้

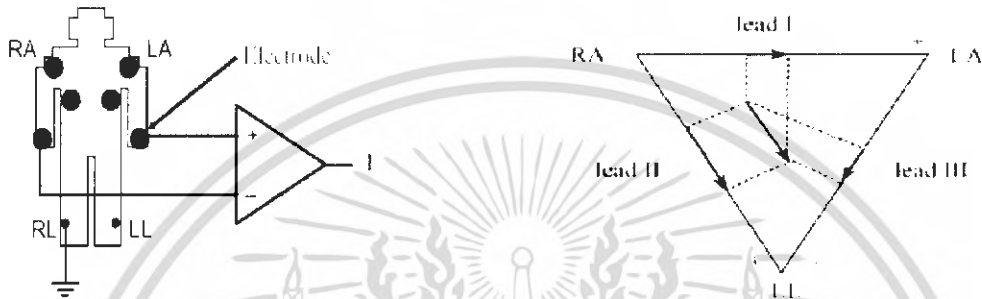


รูปที่ 2.4 กิจกรรมทางไฟฟ้าที่เกิดขึ้นที่บริเวณต่างๆในหัวใจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.2 Lead แบบต่างๆในการวัด ECG

1. Bipolar lead (วัดศักย์ไฟฟ้าระหว่างจุดสองจุดในสามเหลี่ยมของ Eindhoven)
2. Augmented Leads (วัดศักย์ไฟฟ้าของจุดๆหนึ่งเทียบกับค่าเฉลี่ยของอีกสองจุด)
3. Chest (V) Leads (วัดศักย์ไฟฟ้าที่ตำแหน่งต่างๆบนอก V1 ถึง V6 เทียบกับค่าเฉลี่ยของจุดทั้งสามของสามเหลี่ยมของ Eindhoven)



รูปที่ 2.5 แสดงจุดการวัดคลื่นไฟฟ้าหัวใจและสามเหลี่ยมของ Eindhoven

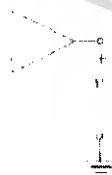
Bipolar Leads: lead I



Bipolar Leads: lead III



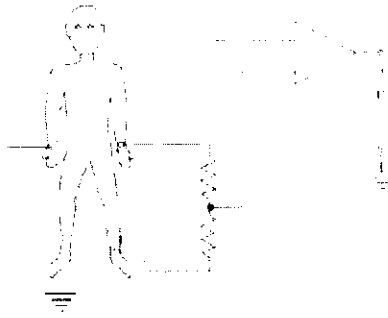
Bipolar Leads: lead II



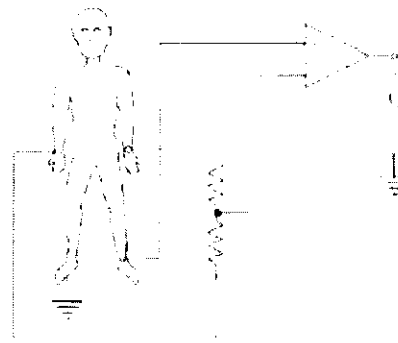
รูปที่ 2.6.1 แสดงจุดการวัดคลื่นไฟฟ้าหัวใจที่ตำแหน่งต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

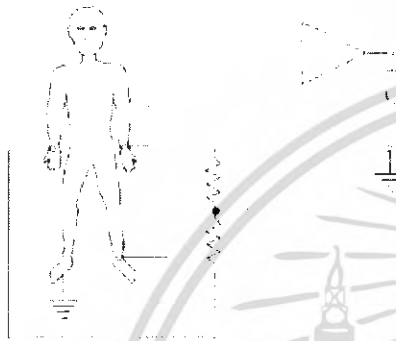
Augmented Leads: aVR



Augmented Leads: aVF



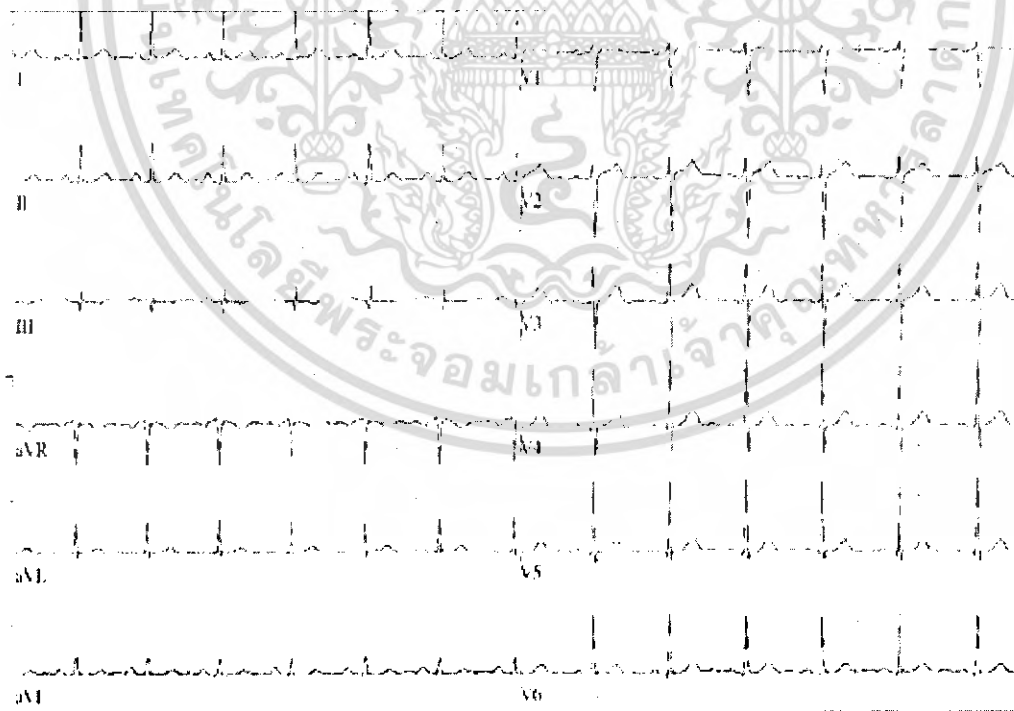
Augmented Leads: aVL



การวัดของ chest leads



รูปที่ 2.6.2 แสดงจุดการวัดคลื่นไฟฟ้าหัวใจที่ตำแหน่งต่างๆ



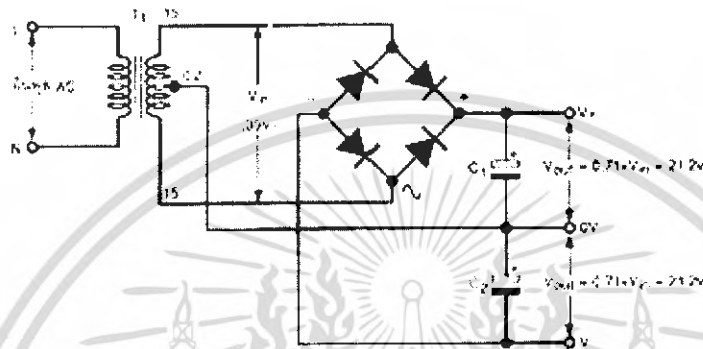
รูปที่ 2.7 Electrocardiography

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### บทที่ 3 โครงสร้างโดยรวมของระบบ

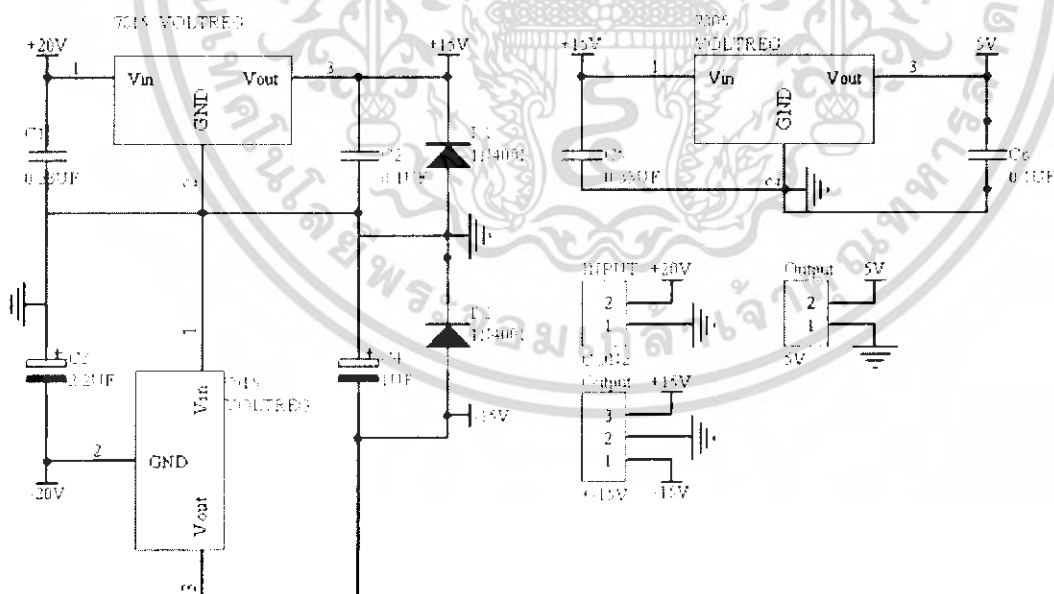
#### 3.1 โครงสร้างทางฮาร์ดแวร์

ในแหล่งจ่ายไฟฟ้ากระแสตรงมีหน้าที่เปลี่ยนแปลงแรงดันเอาต์พุตที่เป็นไฟฟ้ากระแสตรงที่ได้จากวงจรเรียงกระแสทั้งแบบครึ่งคลื่นและแบบเต็มคลื่นซึ่งจะมีการเพิ่มให้เป็นแรงดันไฟฟ้ากระแสตรงที่เรียบมากขึ้นมีการต่อตัวเก็บประจุเพื่อใช้กรองแรงดันให้ได้แรงดันที่เรียบ คือมีค่าริบเบิล น้อยๆนั่นเอง ต่อวงจรได้ดังรูปที่ 3.1.1 ค่า  $V_{out}$  จะหาได้ดังนี้  
แรงดันเอาต์พุตเท่ากับ  $V_{DC} \text{ (full wave)} = (0.636V_P) - 0.71$



รูปที่ 3.1.1 วงจรเรียงกระแสแบบเต็มคลื่น

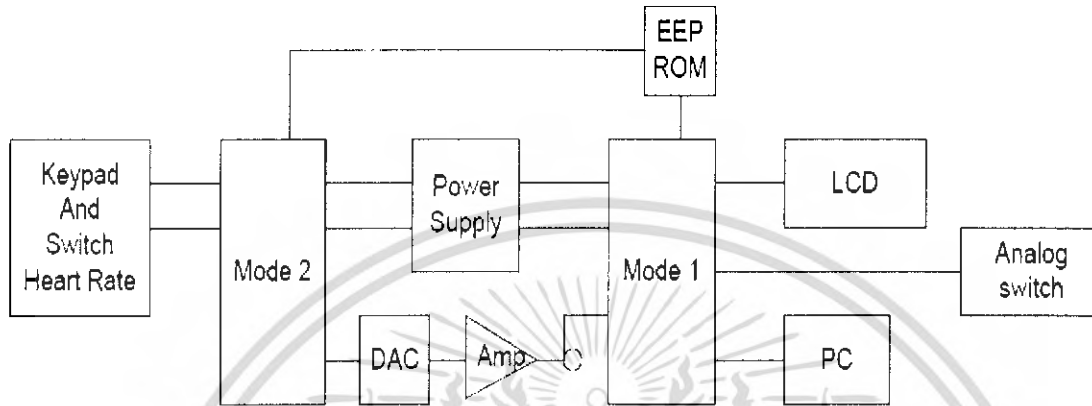
มีการสร้างวงจร Regulator เป็นแหล่งจ่ายไฟตรงให้กับวงจรในการทำงาน Mode 1 และวงจรการทำงานใน Mode 2 โดยต้องทำการสร้างวงจร +15V, -15V และ 5V ดังรูปที่ 3.1.2



รูปที่ 3.1.2 วงจรRegulator +15V,-15V,5V

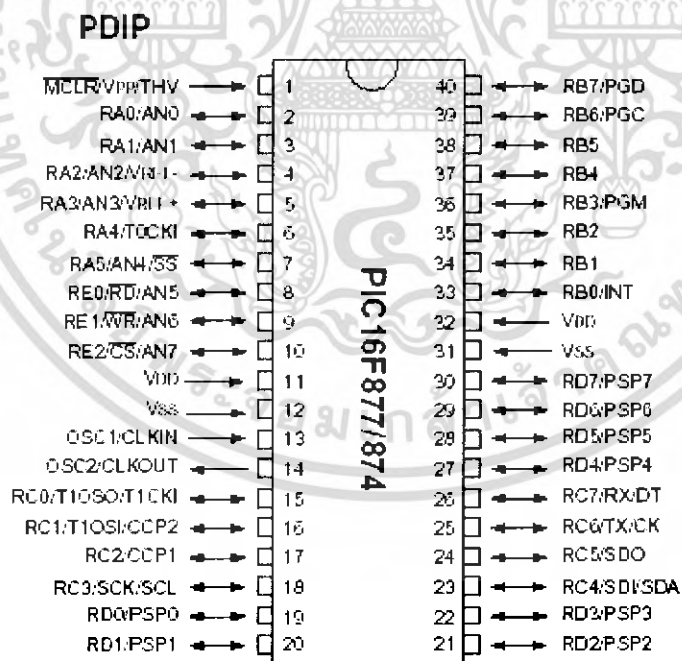
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นำเอาแหล่งจ่ายไฟมาต่อเข้ากับวงจรโดย Pic16f877 ต่อไฟ 5V ส่วน DAC0808 กับ Op-Amp ต่อไฟเลี้ยง +15V กับ -15V ซึ่งเป็นการต่อแบบ ขนนวน 8 bit การเชื่อมต่อกันระหว่าง Pic16f877 กับ EEprom เป็นการเชื่อมต่อในระบบบัส I2C การต่อระหว่าง Pic16f877 กับ PC ใช้แบบการส่งข้อมูลผ่าน MAX232 ในรูปแบบ Serial Port และการเชื่อมต่อกันระหว่าง Pic16f877 กับ LCD เป็นการส่งข้อมูลแบบ 8 bit นำมาต่อเป็นวงจรรวมได้ดังรูปที่ 3.1.3



รูปที่ 3.1.3 บล็อกไดอะแกรมของวงจรในการทำงาน Mode 1 และ Mode 2

โครงสร้าง PORT PIC16F877



รูปที่ 3.2 โครงสร้าง PIC16F877

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



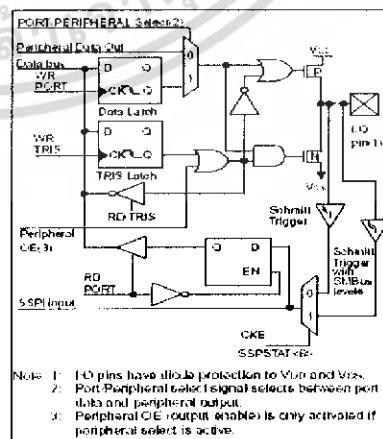
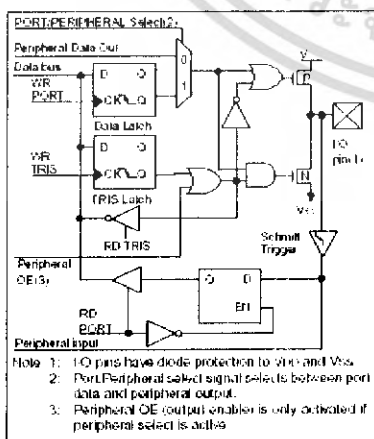
PORTB แต่ละ port จะมี weak pull-up อยู่ภายใน (ถ้าต้องการ pull-up แข็งๆ ต้องต่อวงจรภายนอก) เราสามารถกำหนดว่าจะใช้ pull-up ภายในหรือไม่จากการ set หรือ clear RBPU<sub>A</sub> (OPTION register บิต 7) โดยถ้าเรา clear RBPU<sub>A</sub> จะหมายถึง เราทำการ disable pull-up ภายใน และถ้าเรากำหนดให้ PORTB เป็น OUTPUT แล้ว pull-up จะถูก disable โดยอัตโนมัติ

สำหรับ PORTB นั้น ขา RB4-RB7 จะมี feature เพิ่มเติมก็คือ การกำหนดให้เกิด Interrupt เมื่อเกิดการเปลี่ยนแปลงของสถานะของสัญญาณไฟฟ้าที่ขา RB4-RB7 (โดยถ้าขาใดขาหนึ่งเกิดเปลี่ยนแปลงสถานะก็จะทำให้เกิด RB Port Change Interrupt ขึ้น ซึ่งจะทำให้ RBIF (INTCON.0) flag ถูก set โดยที่ Interrupt ประเภทนี้สามารถทำการ “wake” microcontroller จากสถานะ sleep mode ได้

RBIF flag จะถูก Clear ได้ 2 กรณี คือ

1. ทำการอ่านหรือเขียน PORTB
2. ทำการ clear RBIF flag โดยตรง

PORTC เป็นลักษณะแบบ Port แบบสองทิศทาง ซึ่ง register ที่จะเป็นตัวกำหนดว่า port ใดจะเป็นแบบ input/output จะถูกกำหนดโดย TRISC register ถ้า set TRISC bit ใด (=1) PORTC ที่บิตนั้นก็จะเป็น input ถ้า clear TRISC bit ใด (=0) PORTC ที่บิตนั้นก็จะเป็น output ที่ PORTC จะมีคุณสมบัติเพิ่มเติม เช่น IIC, UART, SPI, PWM, CAPTURE ขึ้นอยู่กับการเลือกใช้งาน โดยเมื่อเราทำการ enable คุณสมบัติเพิ่มเติมต่างๆ ที่ PORTC เราต้องระวังในเรื่องของการตั้งค่า TRISC ของแต่ละขาของ PORTC เพราะในการ enable คุณสมบัติบางตัวที่อยู่ PORTC (เช่น UART) ตัวมันเองก็จะทำการเปลี่ยน bit TRISC โดยอัตโนมัติ ดังนั้นไม่ควรที่จะตั้งค่า TRISC โดยตรงกับขาใดของ PORTC ที่ทำการ enable คุณสมบัติเพิ่มเติม PORTC แต่ละ port จะ Schmitt Trigger input buffers อยู่ภายในแต่ละขา



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TABLE 3-5 PORTC FUNCTIONS

Name	Bit#	Buffer Type	Function
RC0/T1OSC/T1CKI	bit0	ST	Input/output port pin or Timer1 oscillator output/Timer1 clock input
RC1/T1OSC/CF2	bit1	ST	Input/output port pin or Timer1 oscillator input or Capture2 input/Compare2 output/PWM2 output
RC2/CEP1	bit2	ST	Input/output port pin or Capture1 input/Compare1 output/PWM1 output
RC3/ASCK/SCL	bit3	ST	RC3 can also be the synchronous serial clock for both SPI and I <sup>2</sup> C modes
RC4/SD/SDA	bit4	ST	RC4 can also be the SPI Data In/SPI mode or data I/O I <sup>2</sup> C mode
RC5/SSCO	bit5	ST	Input/output port pin or Synchronous Serial Port data output
RC6/TXCK	bit6	ST	Input/output port pin or USART Asynchronous Transmit or Synchronous Clock
RC7/RXD1	bit7	ST	Input/output port pin or USART Asynchronous Receive or Synchronous Data

Legend: ST = Schmitt Trigger input

TABLE 3-6 SUMMARY OF REGISTERS ASSOCIATED WITH PORTC

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other resets
07h	PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	xxxx xxxx
87h	TRISC	PORTC Data Direction Register								1111 1111	1111 1111

Legend: x = unknown, - = unchanged

รูปที่ 3.4 Block Diagram และ Function ที่สำคัญของ PortC

สำหรับ PORTD และ PORTE นั้นจะไม่มีอยู่ใน pic ในตระกูลนี้ที่มีขนาดขา 28 ขา ก่อนอื่น มาพูดถึงถึง PORTD ก่อน PORTD จะเป็น port ขนาด 8 bits ซึ่งจะมี Schmitt Trigger input buffer อยู่ในตัว โดยที่เราสามารถกำหนดแต่ละบิตของ port ให้เป็น input หรือ output ได้โดยอิสระจากกัน PORTD สามารถที่จะทำตัวเป็น parallel slave port ได้อีกด้วย โดยทำได้โดยการ set PSMODE bit (TRISE<4>) ซึ่งใน mode นี้ buffer ภายในจะกลายเป็นแบบ TTL

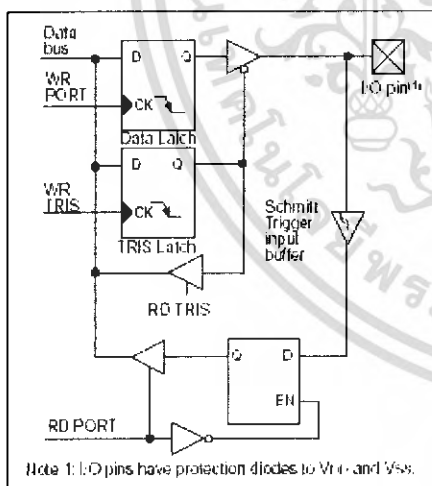


TABLE 3-7 PORTD FUNCTIONS

Name	Bit#	Buffer Type	Function
RD0/PSP0	bit0	ST-TTL <sup>1)</sup>	Input/output port pin or parallel slave port bit0
RD1/PSP1	bit1	ST-TTL <sup>1)</sup>	Input/output port pin or parallel slave port bit1
RD2/PSP2	bit2	ST-TTL <sup>1)</sup>	Input/output port pin or parallel slave port bit2
RD3/PSP3	bit3	ST-TTL <sup>1)</sup>	Input/output port pin or parallel slave port bit3
RD4/PSP4	bit4	ST-TTL <sup>1)</sup>	Input/output port pin or parallel slave port bit4
RD5/PSP5	bit5	ST-TTL <sup>1)</sup>	Input/output port pin or parallel slave port bit5
RD6/PSP6	bit6	ST-TTL <sup>1)</sup>	Input/output port pin or parallel slave port bit6
RD7/PSP7	bit7	ST-TTL <sup>1)</sup>	Input/output port pin or parallel slave port bit7

Legend: ST = Schmitt Trigger input, TTL = TTL input

Note 1: Input buffers are Schmitt Triggers when in I/O mode and TTL buffer when in Parallel Slave Port Mode.

TABLE 3-8 SUMMARY OF REGISTERS ASSOCIATED WITH PORTD

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other resets
10h	PORTD	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	xxxx xxxx	xxxx xxxx
69h	TRISD	PORTD Data Direction Register								1111 1111	1111 1111
69h	TRISE	IBF	OBF	IBOV	PSMODE	—	PORTE Data Direction Bits			11 111	11 111

Legend: x = unknown, - = unchanged, — = unimplemented read as '0'. Shaded cells are not used by PORTD.

รูปที่ 3.5 Block Diagram และ Function ที่สำคัญของ PortD

PORTE จะมีทั้งหมด 3 ขา คือ RE0/(RD)/AN5, RE1/(WR)/AN6 และ RE2/(CS)/AN7 ซึ่งจะมี Schmitt Trigger input buffer อยู่ในตัว โดยที่เราสามารถกำหนดแต่ละบิตของ port ให้เป็น input หรือ output I/O PORTE สามารถกลายเป็น control input สำหรับ microprocessor port เมื่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการ set PSPMODE(TRISE<4>) bit ข้อควรระวังเมื่ออยู่ในโหมดนี้ก็คือ ต้องตรวจดูให้ได้ว่า TRISE ตั้งแต่บิต 0-2 ถูก set (อยู่ในสถานะ input) และต้องแน่ใจว่า ADCON1 ถูก set ให้อยู่ใน mode digital I/O ซึ่งใน mode นี้ input buffer จะเป็น TTL

PORT E จะมีลักษณะคือ จะ multiplex กับ analog inputs โดยเมื่อ PORTE ถูก set เป็น analog inputs แล้ว ขาเหล่านี้เมื่อทำการอ่านค่าจะมีค่าเป็น 0 ส่วน TRISE ซึ่งเป็น control register นั้นจะต้อง set ให้เป็น input เมื่อ set ให้อยู่ใน mode analog input

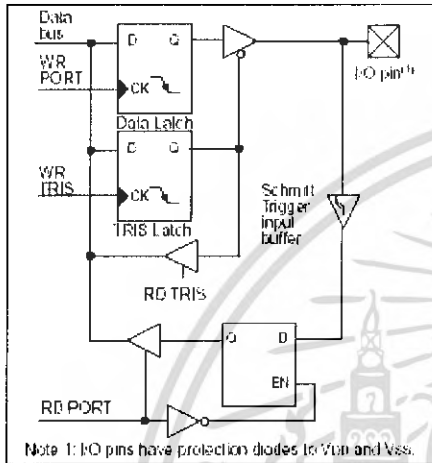


TABLE 3-9 PORT E FUNCTIONS

Name	Bit#	Buffer Type	Function
RE0:RTRAMP	TR0	ST-TTL <sup>1)</sup>	Input/output port pin or read control input in parallel slave port mode or analog input. RTD 1 = Not a read operation 0 = Read operation. Reads RTRAMP register of chip selected.
RE1:WRAMP	WR1	ST-TTL <sup>1)</sup>	Input/output port pin or write control input in parallel slave port mode or analog input. WTR 1 = Not a write operation 0 = Write operation. Writes WRAMP register of chip selected.
RE2:RSAMP	RS2	ST-TTL <sup>1)</sup>	Inputs/output port pin or chip select control input in parallel slave port mode or analog input. RS 1 = Device is not selected 0 = Device is selected

Legend: ST = Schmitt trigger input, TTL = TTL input.  
Note 1: Input buffers are Schmitt triggers when in I/O mode and TTL buffers when in Parallel Slave Port Mode

TABLE 3-10 SUMMARY OF REGISTERS ASSOCIATED WITH PORTE

Addr	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BCR	Value on all other resets
90h	PORTE	—	—	—	—	—	RE2	RE1	RE0	xxx	xxx
92h	TRISE	IGF	ODF	IBOV	PSPMODE	—	PORTE Data Direction Bits			111 111	1 01 111
9Fh	ADCON1	ADFM	—	—	—	PCFG3	PCFG2	PCFG1	PCFG0	—	—

Legend: x = unknown, — = unimplemented, — = unimplemented reads '0'. Shaded cells are not used by PORTE.

รูปที่ 3.6 Block Diagram และ Function ที่สำคัญของ Port E

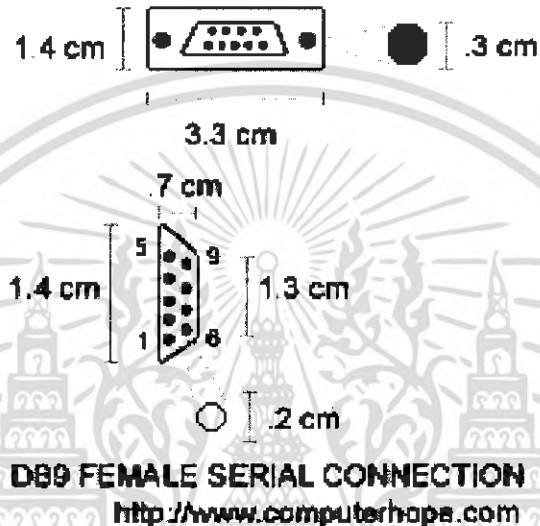
### 3.2 The Serial Port

การใช้ Serial Port มีการใช้ที่ยุงยกว่าการใช้ Parallel Port โดยส่วนใหญ่การต่อ Serial จะมีการต่อแบบ Parallel ควบคู่ไปด้วย โดยมีการใช้ UART ส่วนข้อดีของการใช้ Serial Data Transfer ที่เทียบกับการใช้ Parallel Data Transfer มีดังนี้

1. สายของ Serial จะทนกว่าสายของ Parallel เพราะ Serial Port มีการส่งค่า '1' ประมาณ -3 - -25 volts และส่งค่า '0' ประมาณ 3 - 25 volts แต่ Parallel Port มีการส่งค่า '1' ประมาณ 0 volts และส่งค่า '0' ประมาณ 5 volts จะเห็นว่า Serial Port มีการ Swing 50 v มากกว่า Parallel Port ที่มีการ Swing 5 v ซึ่งมีผลทำให้สายของ Serial จะทนกว่าสายของ Parallel
2. Serial ไม่ต้องใช้สายมากเท่า Parallel สามารถใช้ได้ในระยะที่ไกลกว่า Parallel
3. Infra Red ได้ถูกนำมาใช้อย่างกว้างขวาง ช่วยในการส่งข้อมูลได้ในระยะที่ไกลๆ ได้ Serial Port ได้ถูกนำมามีส่วนสำคัญในการส่งข้อมูลด้วย ซึ่งเป็นที่นิยมมากกว่า Parallel

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เมื่อ Microcontroller ได้ถูกนำมาใช้อย่างแพร่หลาย ส่วนใหญ่ถูกสร้างแบบ SCI (Serial Communications Interfaces) ซึ่งทำให้ลดจำนวน pin count ของ MPU's คือจะใช้แค่ 2 pin คือ Transmit Data (TXD) and Receive Data (RXD) แต่ Parallel ต้องใช้ทั้งหมด 8 pin แต่ใน Project เราจะใช้ Serial Port แบบ DB9 ซึ่งมี 9 pin



รูปที่ 3.7 Serial Port แบบ DB9

ตารางที่ 3.1 แสดง PIN ของ Serial Port แบบ DB9

PIN	PURPOSE	SIGNAL NAME
Pin 1	Data Carrier Detect	DCD
Pin 2	Received Data	RxDatA
Pin 3	Transmitted Data	TxDatA
Pin 4	Data Terminal Ready	DTR
Pin 5	Signal Ground	Gnd
Pin 6	Data Set Ready	DSR
Pin 7	Request To Send	RTS
Pin 8	Clear To Send	CTS
Pin 9	Ring Indicator	RI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3 I2C BUS

#### 3.3.1 คุณสมบัติพื้นฐาน

- เป็น two-wired bus
- ความเร็ว : 100 kbps (standard mode) 400 kbps (fast mode) 3.4 Mbps (high-speed mode)
- การส่งข้อมูลเป็นแบบอนุกรม ทีละ 8 bit
- สายสัญญาณเป็นแบบ 2 ทิศทาง
- ค่าความจุไฟฟ้ารวมบนสายต้องไม่เกิน 400pF
- การเข้าถึงอุปกรณ์มี 2 แบบคือ 7-bit addressing หรือ 10-bit addressing

#### 3.3.2 Data Transfer

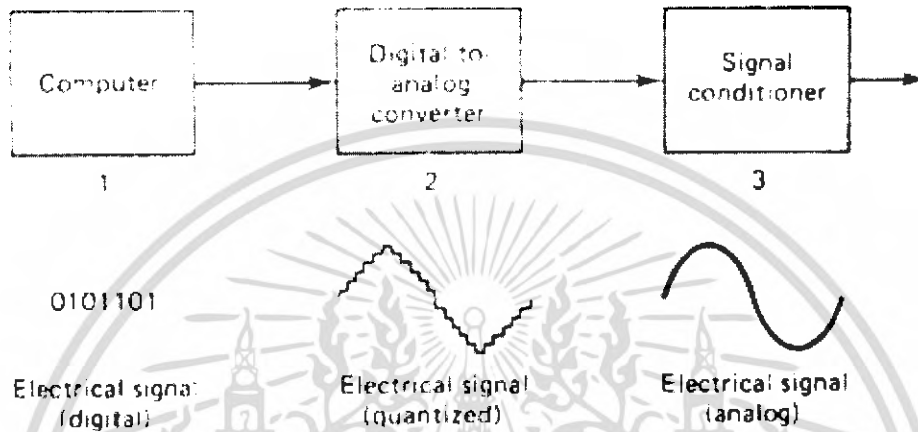
1. ข้อมูลแบบบิตจะถูกส่งผ่านหลังการเริ่มต้นเงื่อนไข
2. การส่งข้อมูลจะอยู่ในรูปของไบต์
3. ข้อมูล 1 ไบต์เท่ากับข้อมูล 8 บิตรวมกับอีก 1 บิตที่รู้ค่า
4. SLAVE ADDRESS เป็นที่เก็บข้อมูล
  - ข้อมูลไบต์แรกที่ถูกส่ง
  - ระหว่างการส่งข้อมูลไบต์แรก
5. MASTER ทำหน้าที่เป็นตัวส่งข้อมูล  
SLAVE ทำหน้าที่เป็นตัวรับข้อมูล
  - การรับครั้งต่อไปขึ้นอยู่กับบิตสุดท้ายใน ADDRESS ไบต์

## สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

### 3.4 D/A Conversion

#### วงจรแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาล็อก(D/A หรือ D2A)

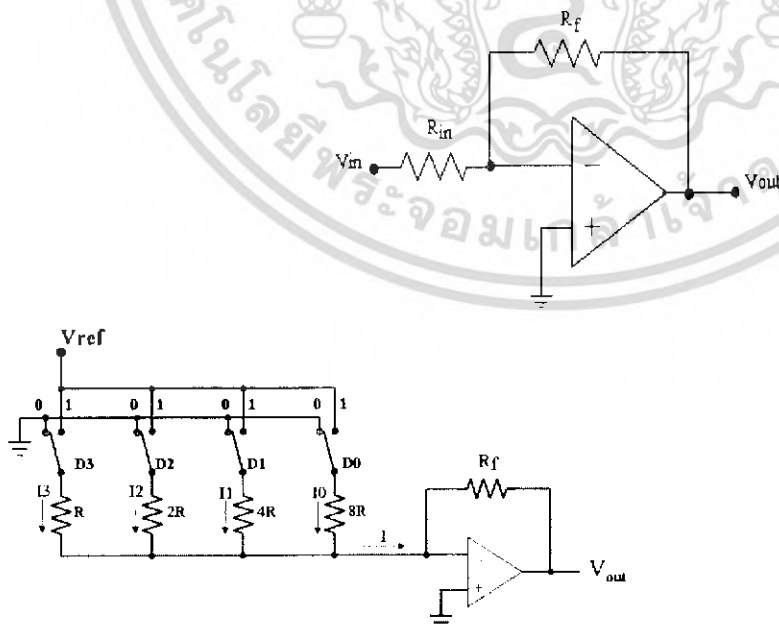
หมายถึงวงจรที่เปลี่ยนข้อมูลทางดิจิทัลให้เป็นค่าอนาล็อก ค่าอนาลอกนี้อาจเป็นกระแสไฟฟ้าหรือแรงดันไฟฟ้าก็ได้



รูปที่ 3.8 การแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาล็อก

#### 3.4.1 Binary Weighted DAC

- Uses switches & precision resistors
- Difficult to achieve high density due to large resistance values

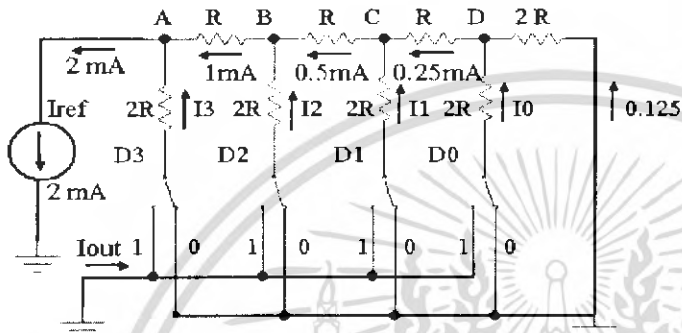


รูปที่ 3.9 แสดงวงจร Binary Weighted DAC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ 73182 ษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4.2 R/2R Ladder

- เป็นวิธีที่ง่ายและไม่ยุ่งยากที่สุด
- เป็นวิธีที่ทำให้ R รวมมีค่าน้อย
- ใช้ในการส่งแบบ INTRICATE



รูปที่ 3.10 แสดงวงจร **R/2R Ladder**

พบว่าค่า  $I_{out}$   $V_{out}$  ที่ได้นั้นมีลักษณะเป็นสัญญาณอนาล็อกโดยที่สัญญาณเข้ามีลักษณะที่คล้ายกับเป็นสัญญาณดิจิทัล

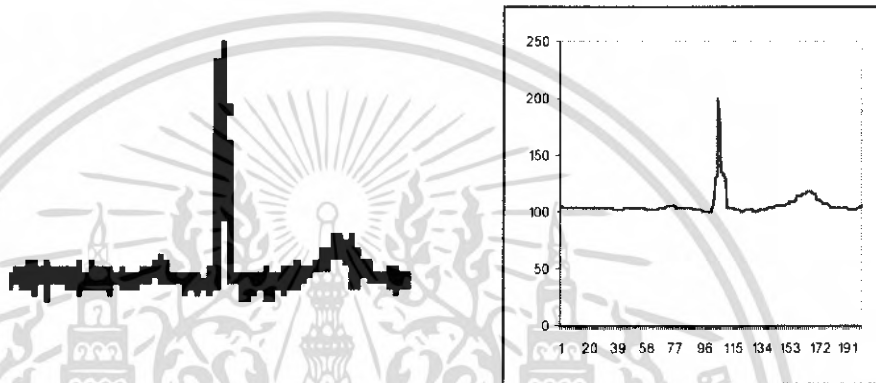
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4 โครงสร้างทางซอฟต์แวร์

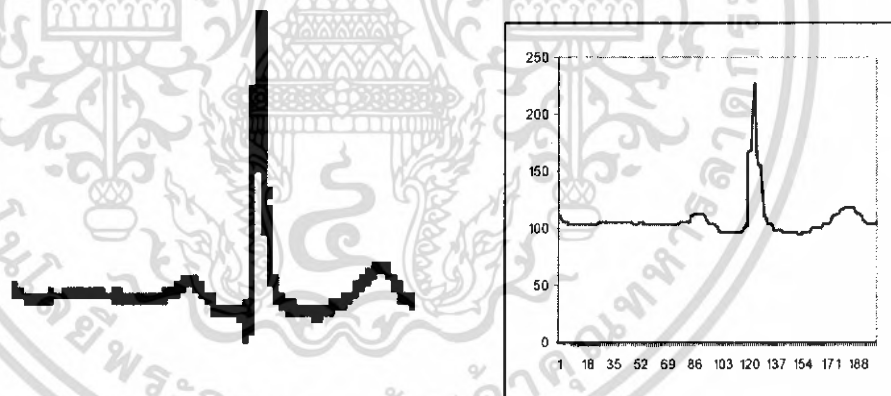
### 4.1 ข้อมูลของกราฟ Normal ECG 12 Leads

จากรูปกราฟ Normal ECG 12 Leads ที่ได้ทำการค้นหาจาก Internet เราได้ทำการปรับปรุงภาพโดยใช้หลักการของ วิชา Digital Image Processing ใช้โปรแกรม MatLab ในการปรับปรุงภาพให้ได้รูปที่ชัดเจนขึ้นเพื่อนำค่าที่ได้มาเก็บเป็น ข้อมูลใส่ไว้ใน EEprom ได้รูปกราฟดังนี้

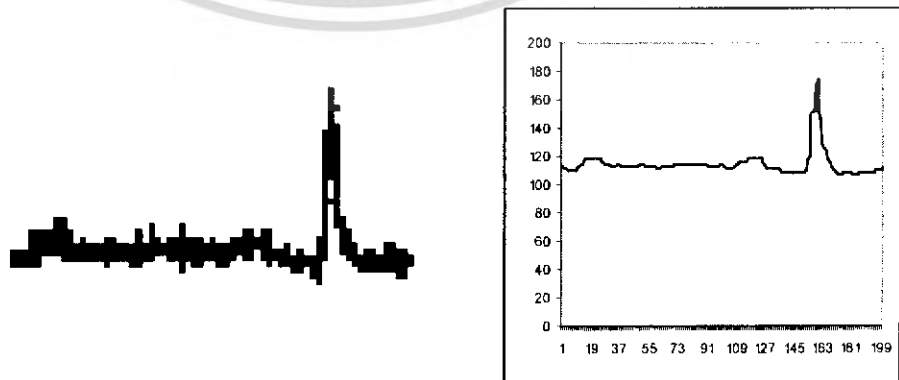
กราฟ Lead 1



กราฟ Lead 2

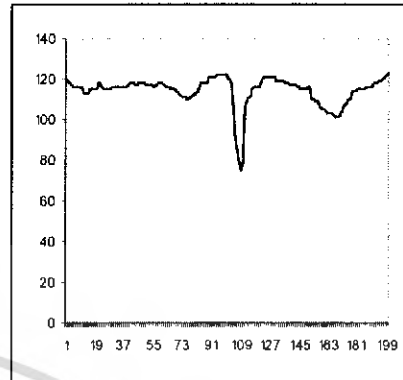


กราฟ Lead 3

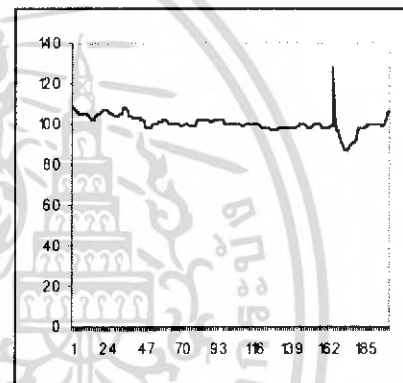


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

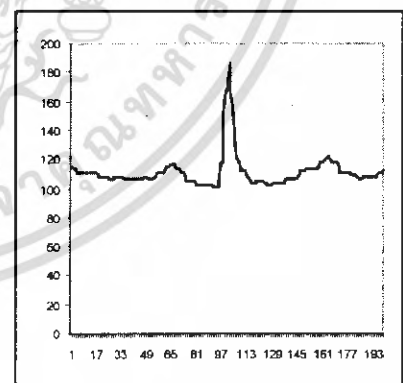
กราฟ aVR



กราฟ aVL

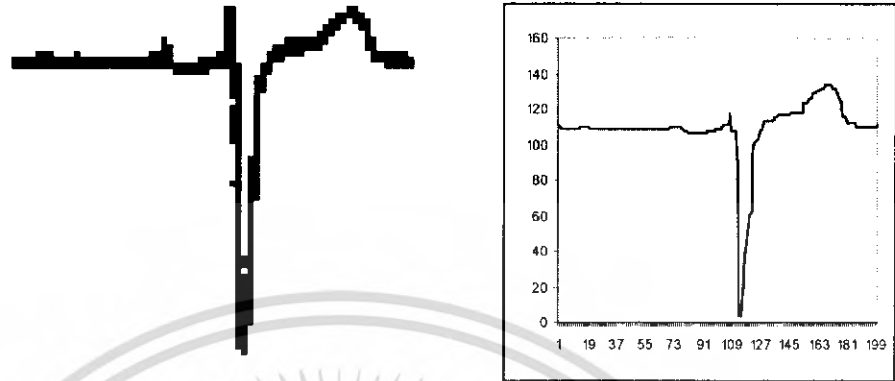


กราฟ aVF

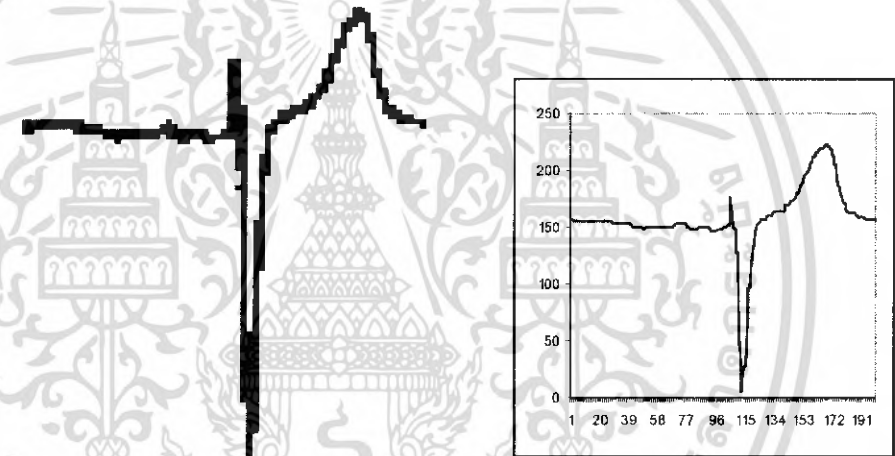


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

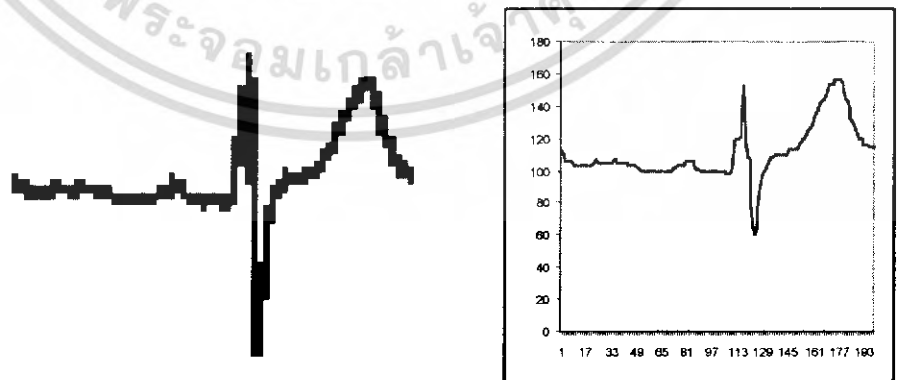
กราฟ V1



กราฟ V2

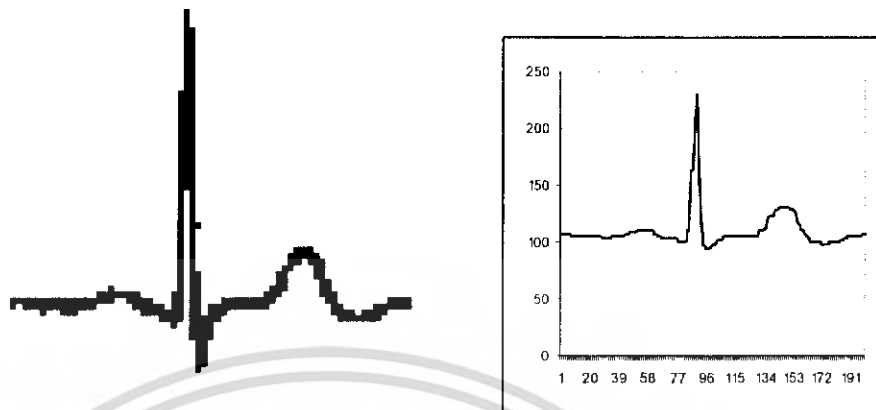


กราฟ V3

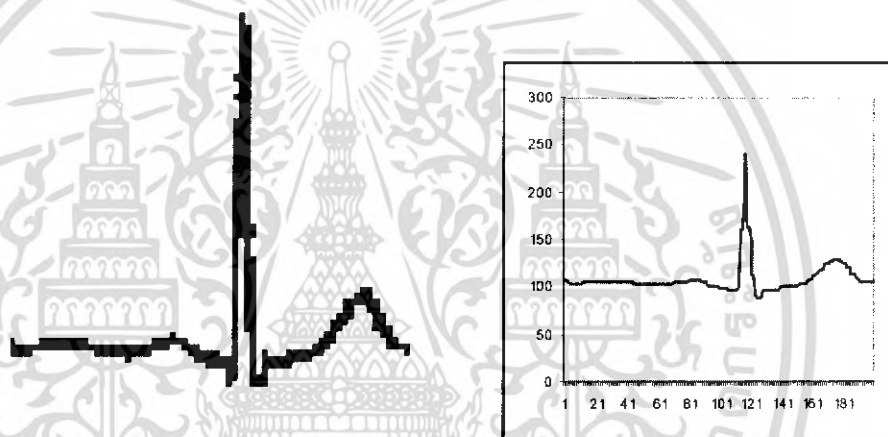


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

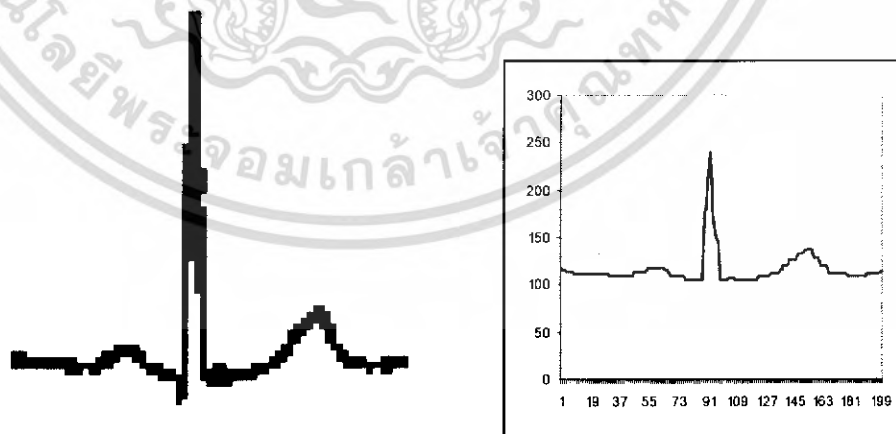
กราฟ V4



กราฟ V5



กราฟ V6

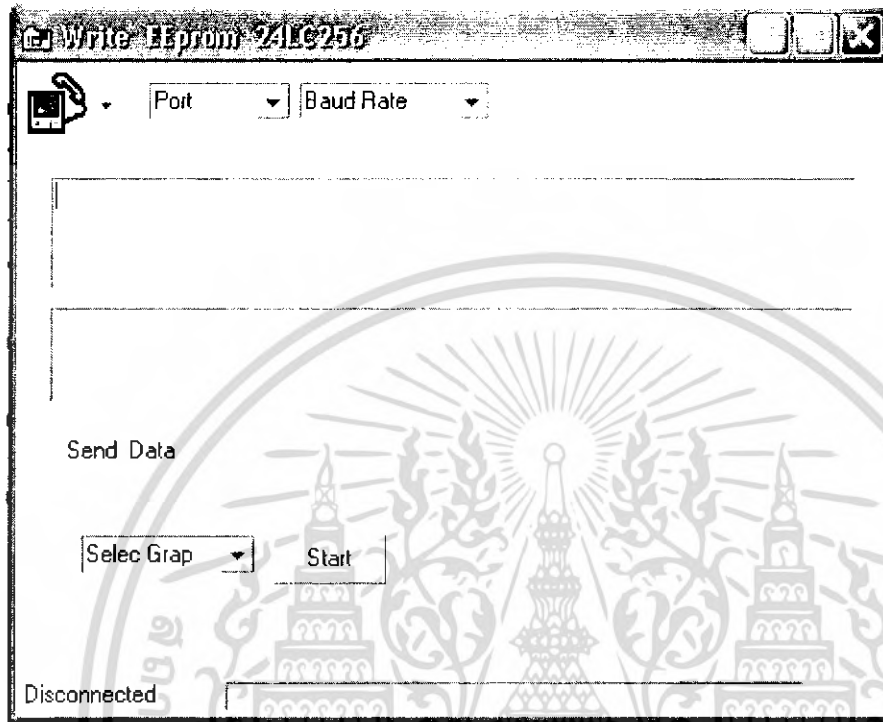


รูปที่ 4.1 กราฟ Normal ECG 12 Leads

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

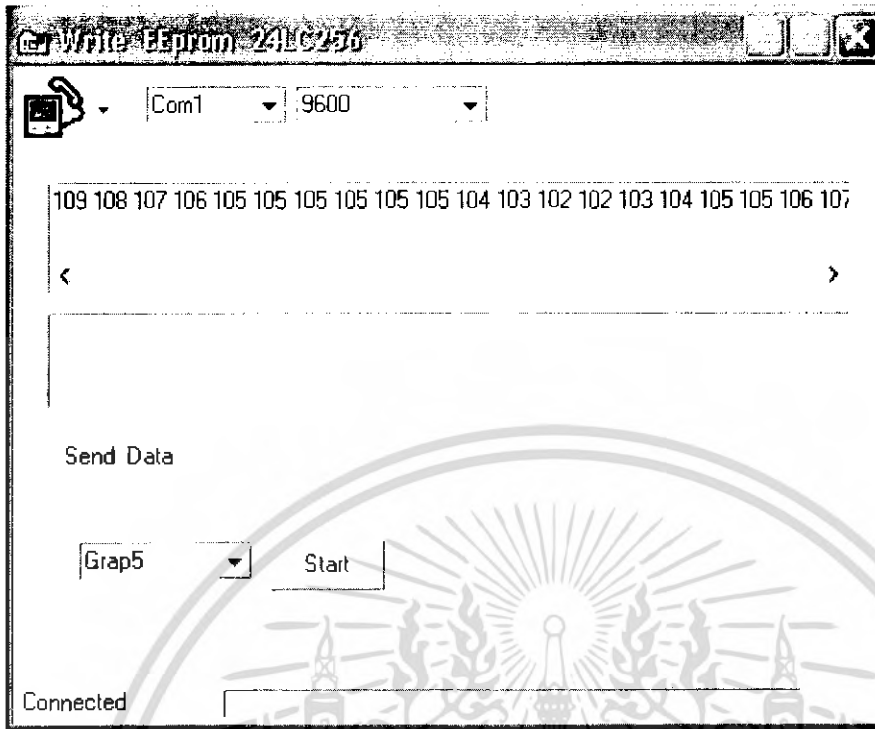
## 4.2 การทำงานของโปรแกรม Write Data in EEprom

เมื่อเปิด โปรแกรมขึ้นมาจะมีหน้าจอดังรูป



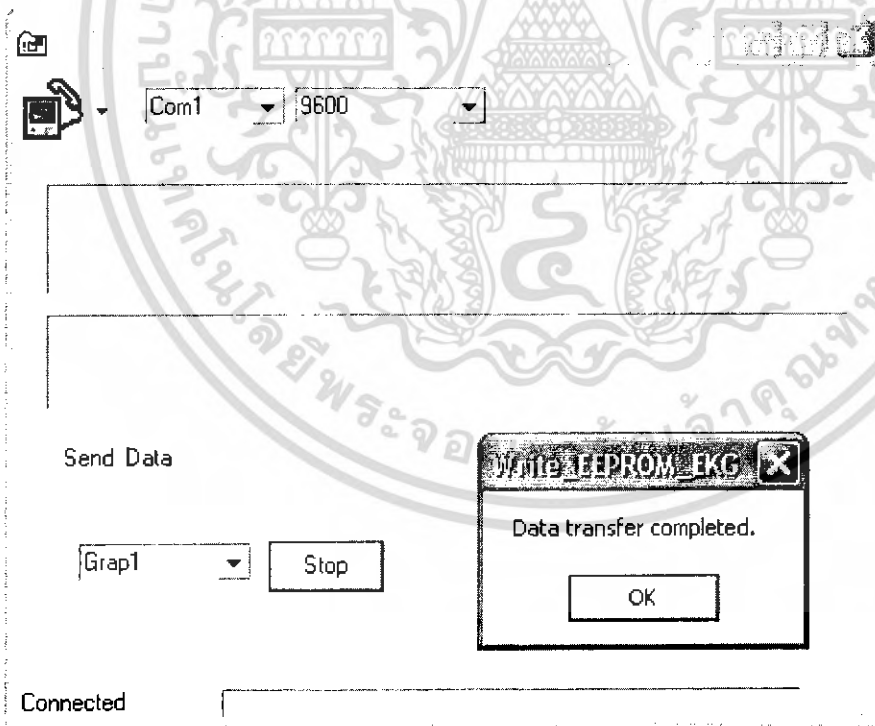
รูปที่ 4.2.1 โปรแกรม Write Data in EEprom

ให้เราทำการ set ค่า Port และ Baud Rate แล้วก็ทำการกด Connect โปรแกรมถึงจะเริ่มใช้งานได้ตรง ส่วน Send Data จะเป็นการเลือกว่าเราต้องการจะเก็บค่าข้อมูลของกราฟลง EEprom ที่ตำแหน่งไหน เมื่อเลือกแล้วโปรแกรมก็จะทำการส่งสัญญาณไปเลือกว่าจะเก็บข้อมูลของกราฟนี้ไว้ที่ตำแหน่งไหน แล้วเราก็ทำการใส่ค่าข้อมูลที่เราต้องการเก็บลงในช่อง TextBox ดังรูป



รูปที่ 4.2.2 โปรแกรม Write Data in EEprom

จากนั้นกดปุ่ม start เพื่อส่งข้อมูลไปเก็บใน EEPROM

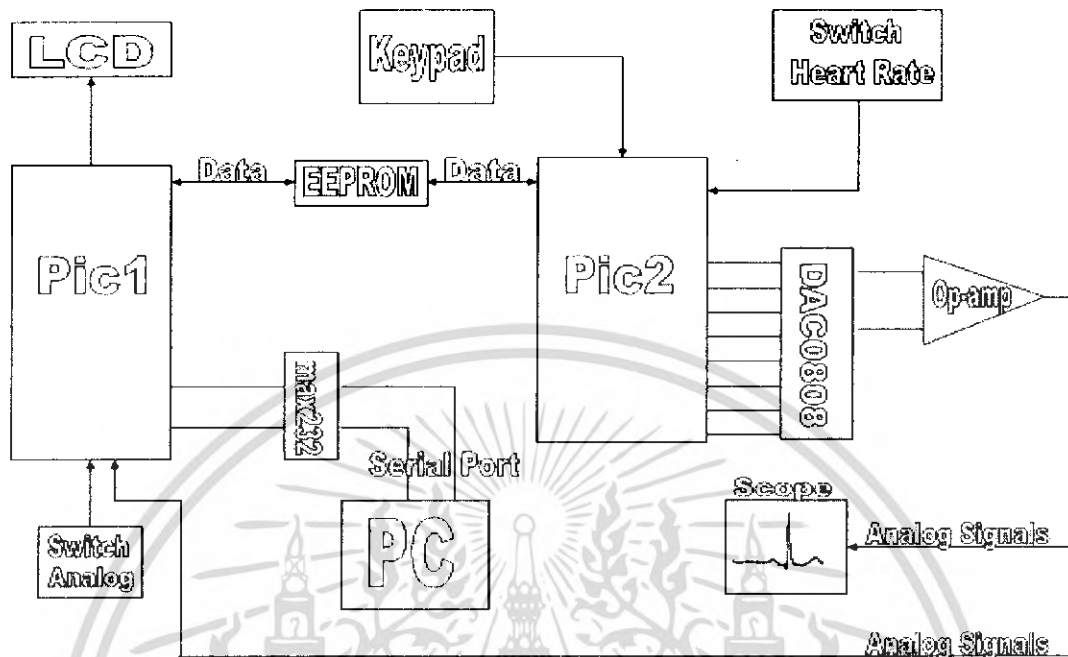


รูปที่ 4.2.3 โปรแกรม Write Data in EEprom

จะมีหน้าต่างบอก เมื่อทำการส่งข้อมูลเสร็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 4.3 การทำงานของโปรแกรม VB Normal 12 Leads ECG มี 2 Mode ในการทำงาน



รูปที่ 4.3 บล็อกไดอะแกรมของวงจรในการทำงาน Mode 1 และ Mode 2

#### MODE 1

1. การติดต่อระหว่าง PC กับ PIC16877 จะติดต่อกันทาง serial port โดยใช้ Max232 เป็นตัวแปลงไฟ
2. เมื่อต้องการแสดงผลที่หน้าจอคอมพิวเตอร์ PC จะส่งคำสั่งผ่าน serial port มายัง PIC16877 แล้ว PIC16877 จะเรียกอ่านข้อมูลจาก EEPROM แล้วจะส่งข้อมูลไปยัง PC เพื่อแสดงผลสามารถเลือกการแสดงผลได้ตาม lead ต่างๆ สามารถต่อเป็นวงจรได้ดังรูปวงจร Mode 1 ในภาคผนวก

#### MODE 2

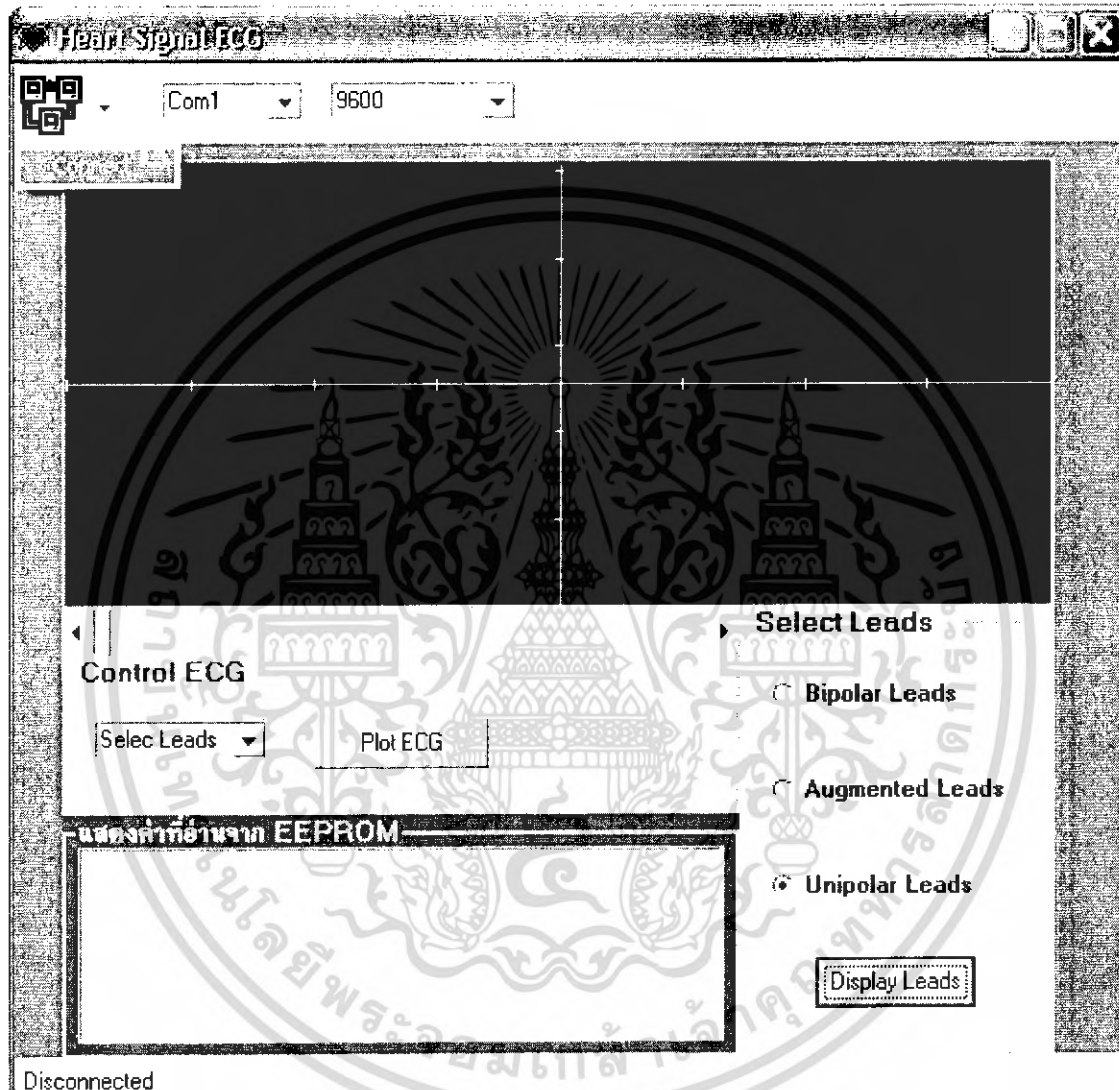
1. ปรับสวิตช์เลือก heart rate และกดคีย์แปดเพื่อเลือกกราฟ
2. PIC2 จะส่งสัญญาณไปบอกให้ PIC1 แสดงข้อมูลที่จอ LCD ว่ากำลังแสดงกราฟอะไรอยู่และ heart rate เท่าไหร่ ในขณะเดียวกัน PIC2 ยังทำการเรียกอ่านค่าจาก EEPROM และส่งข้อมูลออกไปยัง port B แล้วข้อมูลก็จะถูกเปลี่ยนจากข้อมูลดิจิทัลเป็นอนาล็อกโดย D/A (แบบขนาน) และขยายสัญญาณโดยออปแอม ก็จะได้สัญญาณไฟฟ้าหัวใจ ที่มี heart rate และ lead ตามต้องการ
3. ถ้าต้องการรับค่าสัญญาณอนาล็อกไปวิเคราะห์ทำได้โดย กดเปิดสวิตช์ Analog input แล้วเปิดโปรแกรมวิเคราะห์สัญญาณไฟฟ้าหัวใจก็จะรู้ว่าการทำงานของหัวใจเป็นไปในทิศทางใด ปกติหรือไม่

สามารถต่อเป็นวงจรได้ดังรูปวงจร Mode 2 ในภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ขั้นตอนการใช้งานโปรแกรม

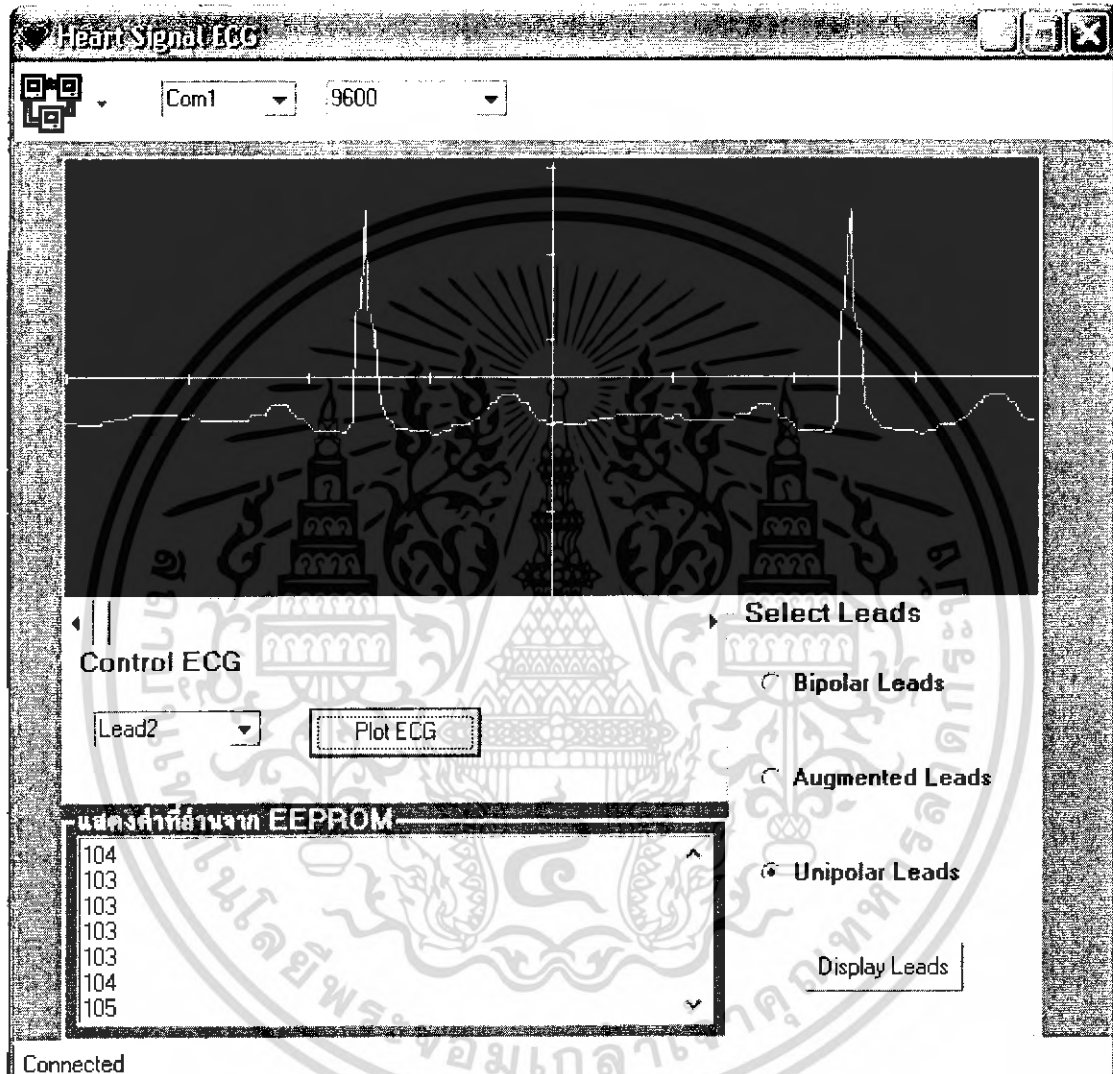
เริ่มต้นที่หน้าจอ Main Program จะมีให้เลือก set ค่า Port และ Baud Rate โดยมีต้องทำการset ค่าก่อนแล้วจึงกด Connect



รูปที่ 4.3.1 โปรแกรม VB Normal 12 Leads ECG

ที่ส่วนของ Control ECG จะเป็นการพล็อตกราฟสัญญาณหัวใจแบบปรกติทีละสัญญาณ Lead1,Lead2,Lead3,aVR,aVL,aVF,V1,V2,V3,V4,V5,V6 เมื่อเลือกแล้ว Program ก็จะมีการส่งสัญญาณไปบอกให้ EEPROM ส่งค่าออกมาแสดงที่ส่วนของ "แสดงค่าที่อ่านจาก EEPROM" แล้วก็ทำการกดปุ่ม "Plot ECG" ที่จอแสดงผลก็จะแสดงกราฟที่เราเลือกไว้และปุ่ม "Plot ECG" ก็จะเปลี่ยนชื่อเป็น "Pause" เมื่อเรากดกราฟที่แสดงอยู่ก็จะหยุดทำงาน เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

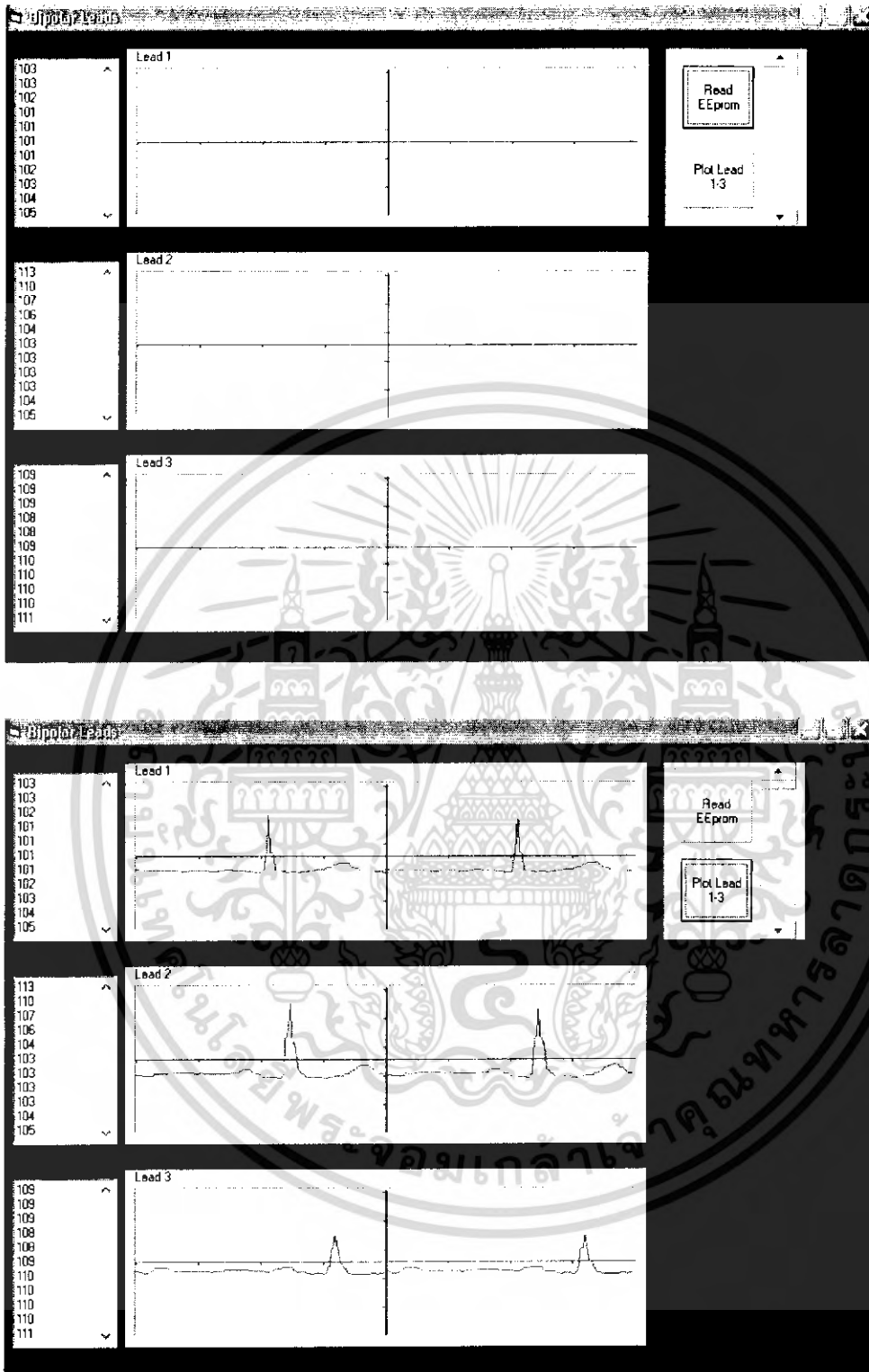
และปุ่มก็จะเปลี่ยนเป็น “Plot ECG” อีกครั้งเมื่อเรากดปุ่ม “Plot ECG” ก็จะแสดงผลอีกครั้งการทำงานก็จะวนไปตามที่เราต้องการ เราสามารถเลื่อนแถบ Scrollbar เพื่อปรับความเร็วของการที่แสดงผลได้ดังรูป



รูปที่ 4.3.2 โปรแกรม VB Normal 12 Leads ECG

ที่ส่วน Select Leads จะเป็นการแสดงกราฟสัญญาณแบบทีละ Leads แยกจากกันเมื่อเราเลือกแล้วก็กดปุ่ม “Display Leads” ก็จะปรากฏหน้าต่าง Program ใหม่มาเมื่อเรากดปุ่ม “Read EEprom” ก็จะทำการอ่านค่าจาก EEprom มาแสดงให้คุณ แล้วก็กดปุ่ม “Plot” เหมือนการแสดงผลแบบหน้าจอ Main ดังรูป

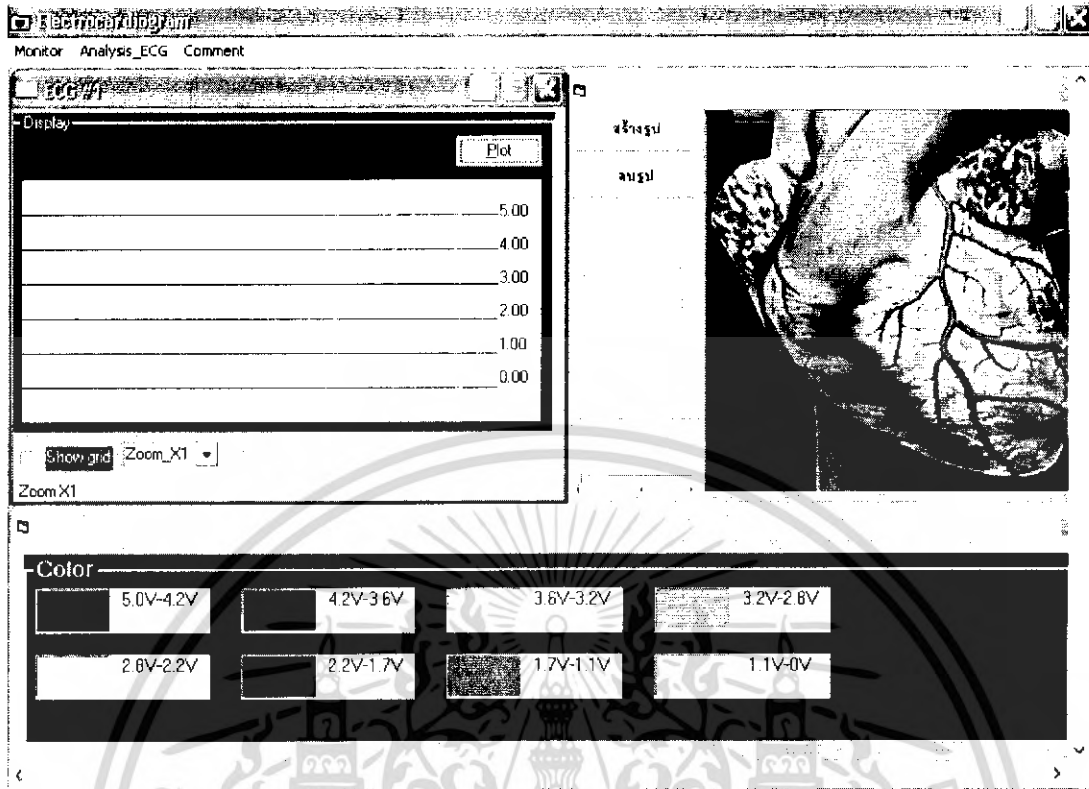
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3.3 โปรแกรม VB Normal 12 Leads ECG

โดยการทำงานในส่วนนี้จะเหมือนกันหมดทุกอย่างต่างกันตรงที่ค่าที่อ่านจาก EEPROM และรูปกราฟในการแสดงผลเท่านั้น เมื่อเลิกการใช้ Program ก็กดปุ่มปิด Program ก็จะปิดการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3.4 โปรแกรม VB Normal 12 Leads ECG

ในรูปการทำงานส่วนนี้เป็นหน้าจอแสดงผล Display และ Analysis สัญญาณ ECG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

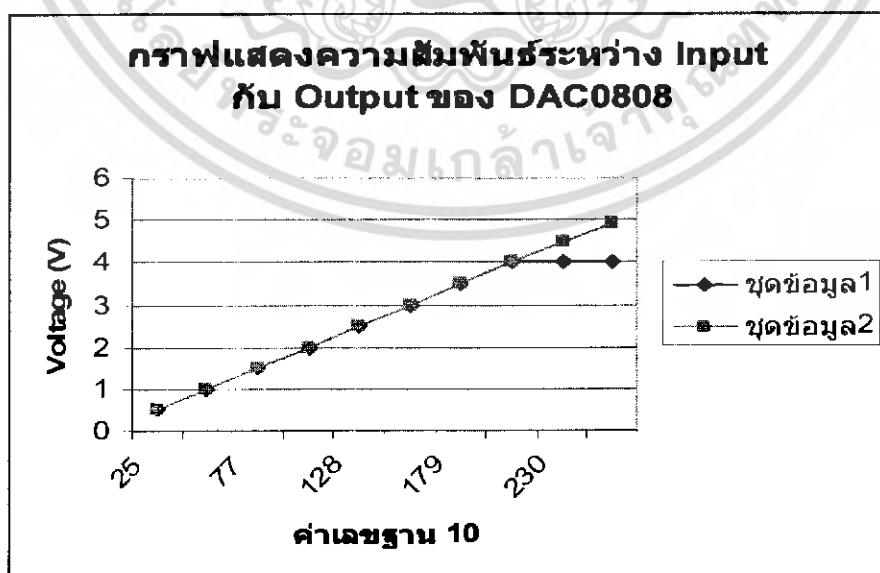
## บทที่ 5 ผลการทดลอง

### 5.1 ผลการทดลอง Output (V) ของ DAC (DAC0808)

เมื่อให้สัญญาณ Output แบบ Ramp Function โดยจ่ายไฟเลี้ยงที่ +5,-5V และ +15,-15V มาเปรียบเทียบกับเพื่อที่จะใช้ไฟเลี้ยงให้กับ DAC มี Output ที่ทำงานได้ครบทุกค่าของเลข 8-bit โดยที่ +5,-5V เป็นชุดข้อมูล 1 และ +15,-15V เป็นชุดข้อมูล 2

ตารางที่ 5.1 แสดงผลการทดลอง Output (V) ของ DAC (DAC0808)

ฐาน 10	ฐาน 2	V output (V)	+5,-5 V	+15,-15 V
25	00011001	0.5	0.5	0.5
51	00110011	1.0	1.0	1.0
77	01001100	1.5	1.5	1.5
102	01100110	2.0	2.0	2.0
128	01111111	2.5	2.5	2.5
153	10011001	3.0	3.0	3.0
179	10110010	3.5	3.5	3.5
204	11001100	4.0	4.0	4.0
230	11100101	4.5	4.0	4.5
255	11111111	5.0	4.0	4.9



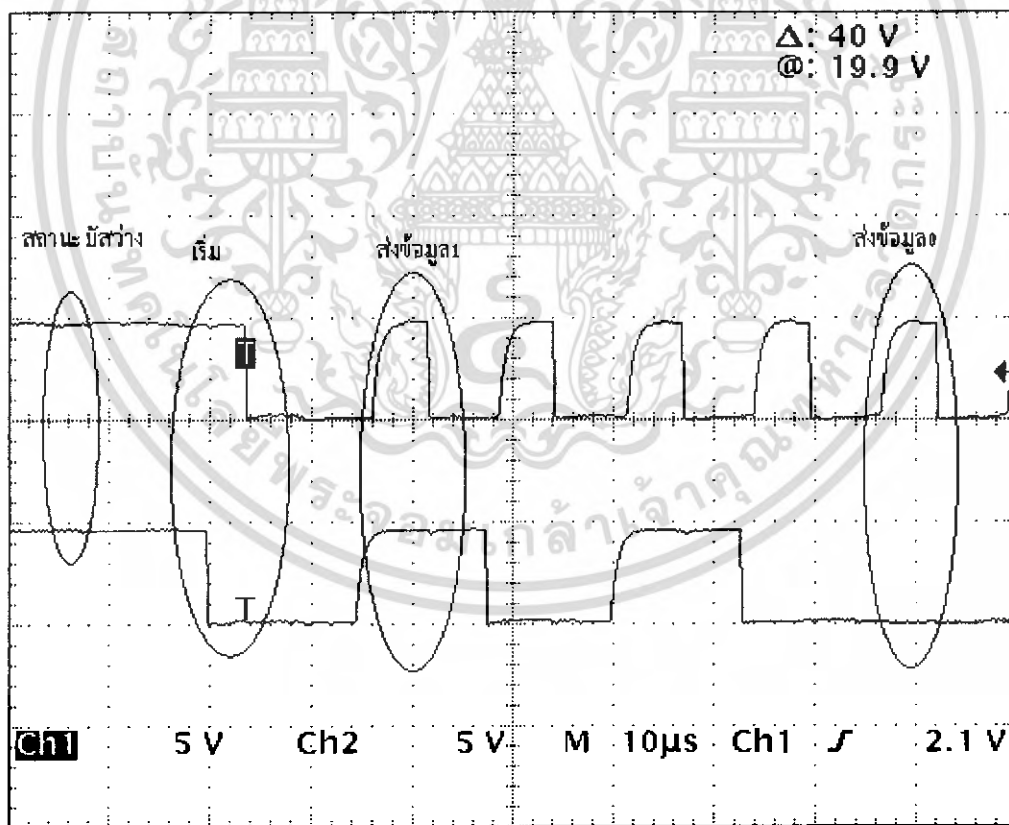
รูปที่ 5.1 กราฟแสดงความสัมพันธ์ I/P และ O/P ของ DAC0808

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 5.2 มอดการทดลองการติดต่อกับ EEprom ด้วย Bus I2C

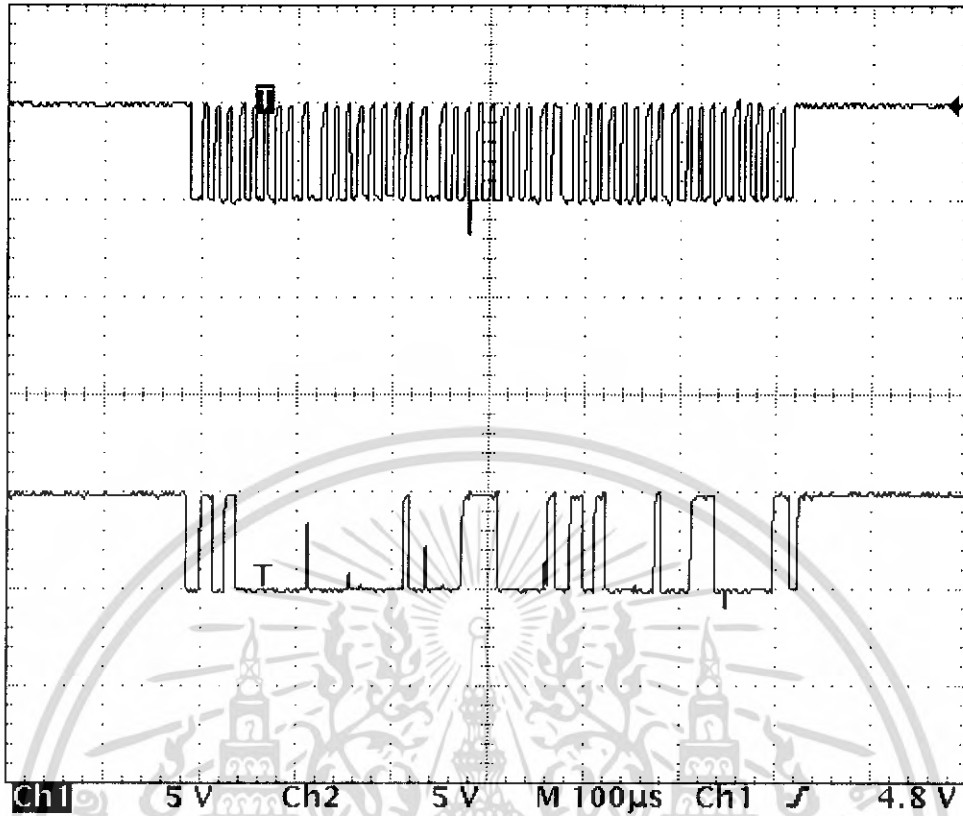
สภาวะที่เกิดขึ้นบนระบบบัส I<sup>2</sup>C มีด้วยกัน 5 สภาวะดังนี้

1. เริ่มต้นที่สภาวะบัสว่างโดยสถานะของ Logic ของขา SCL และ SDA จะมีสถานะเป็น Logic 1 ทั้งคู่
2. สภาวะเริ่มต้นจะเกิดขึ้นเมื่อ Logic ของ SDA เปลี่ยนจาก 1 เป็น 0 ในขณะที่ Logic ของ SCL ยังคงเป็น 1 อยู่
3. การส่ง Logic 1 ของ SDA เกิดขึ้นเมื่อ Logic ของ SDA เป็น 1 คงทีในขณะที่ Logic ของ SCL เป็น 1
4. การส่ง Logic 0 ของ SDA เกิดขึ้นเมื่อ Logic ของ SDA เป็น 0 คงทีในขณะที่ Logic ของ SCL เป็น 1
5. สภาวะการหยุดส่งข้อมูล เกิดขึ้นเมื่อ Logic ของ SDA เปลี่ยนแปลงจาก 0 เป็น 1 ในขณะที่ Logic ของ SCL เป็น 1 คงที่อยู่

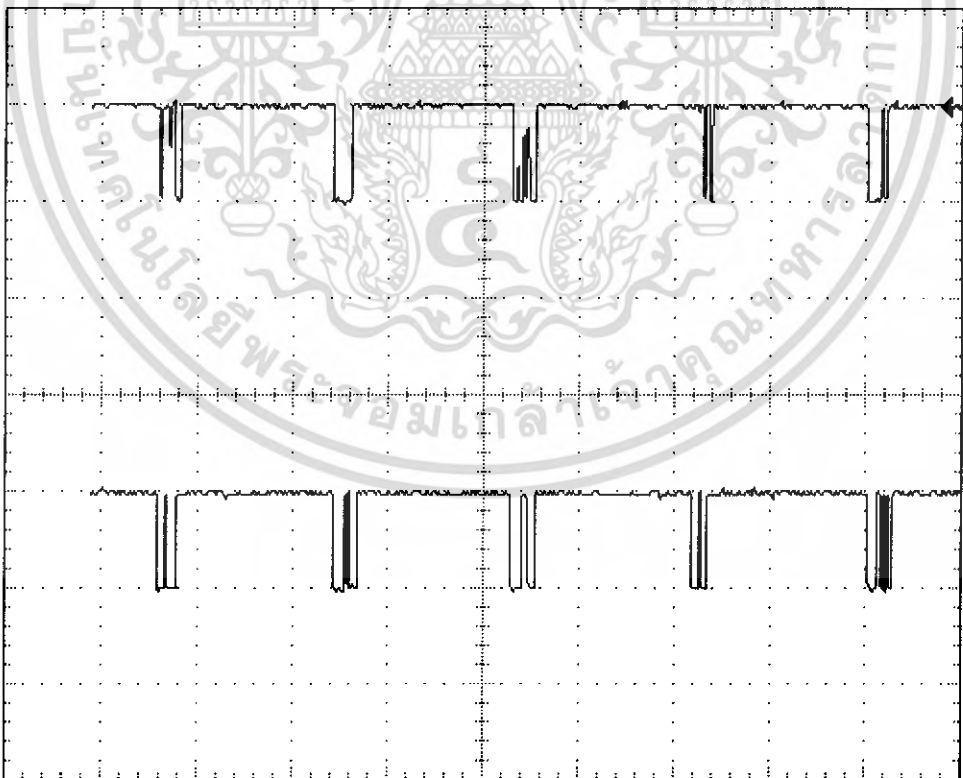


รูปที่ 5.2 แสดงผลการส่งข้อมูล 1 และ ข้อมูล 0 ในระบบบัส I<sup>2</sup>C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.3 แสดงผลการส่งข้อมูล 1 ชุดให้กับ EEprom ในระบบบัส I<sup>2</sup>C



รูปที่ 5.4 แสดงผลการส่งข้อมูลในระบบบัส I<sup>2</sup>C หลายชุดข้อมูล

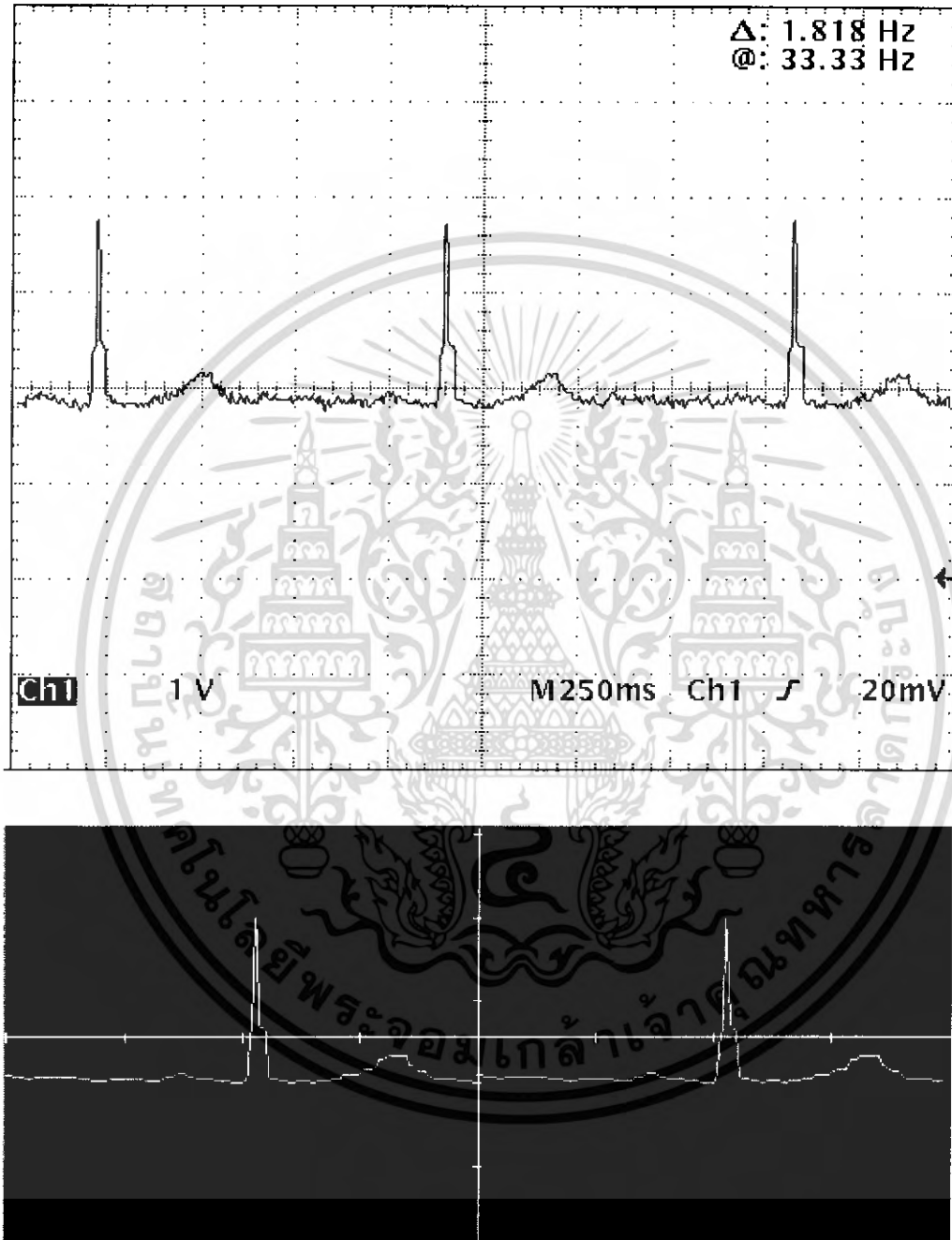
จากรูป กราฟเส้นบนเป็น กราฟ Clock (SCL) ส่วนกราฟด้านล่างเป็นกราฟข้อมูล (SDA)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 5.3 กราฟการเปรียบเทียบการแสดงผลระหว่าง Oscilloscope และ Program VB

Bipolar Leads

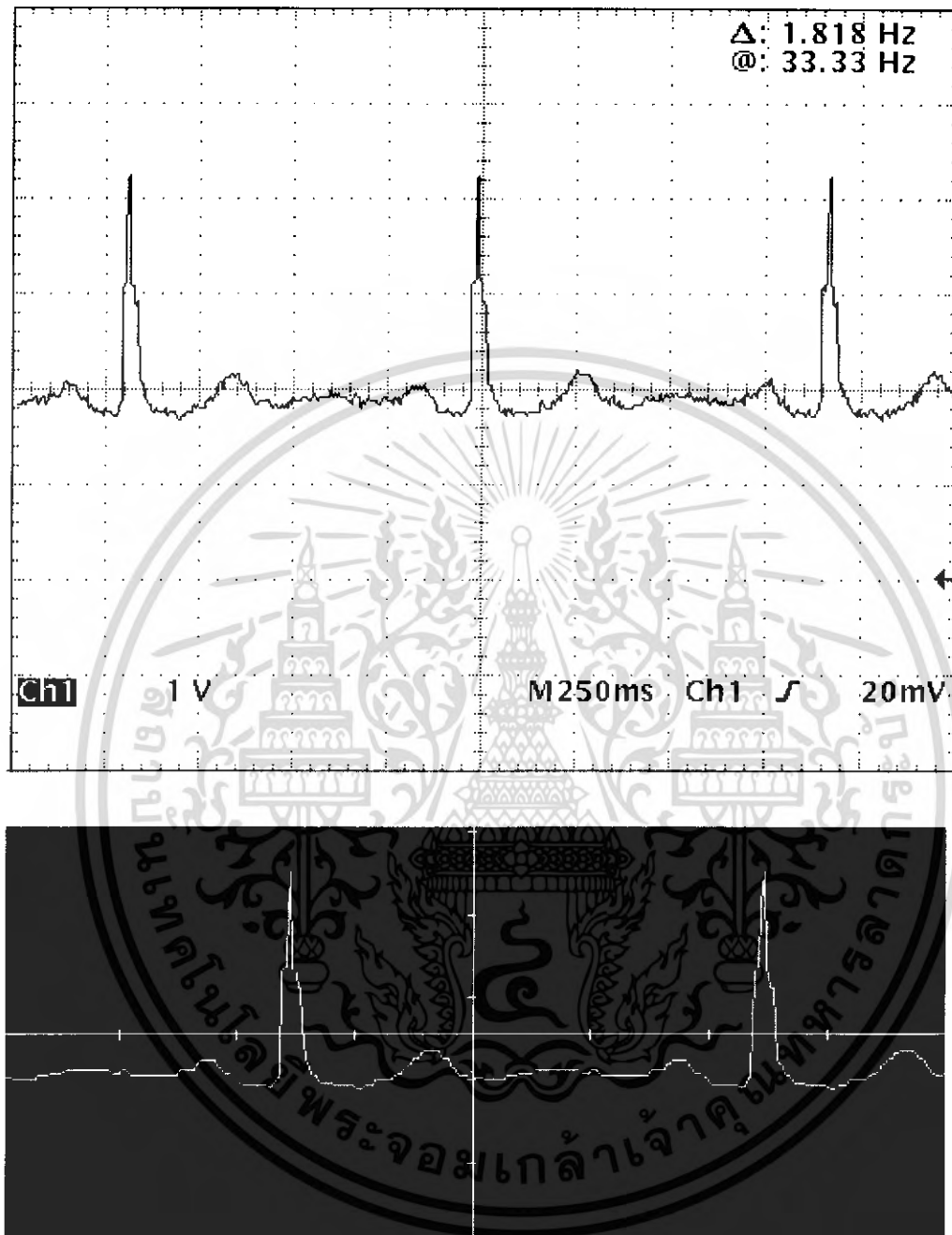
Lead1 Keypad \*1\*



รูปที่ 5.5.1 กราฟการเปรียบเทียบการแสดงผลระหว่าง Oscilloscope และ Program VB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

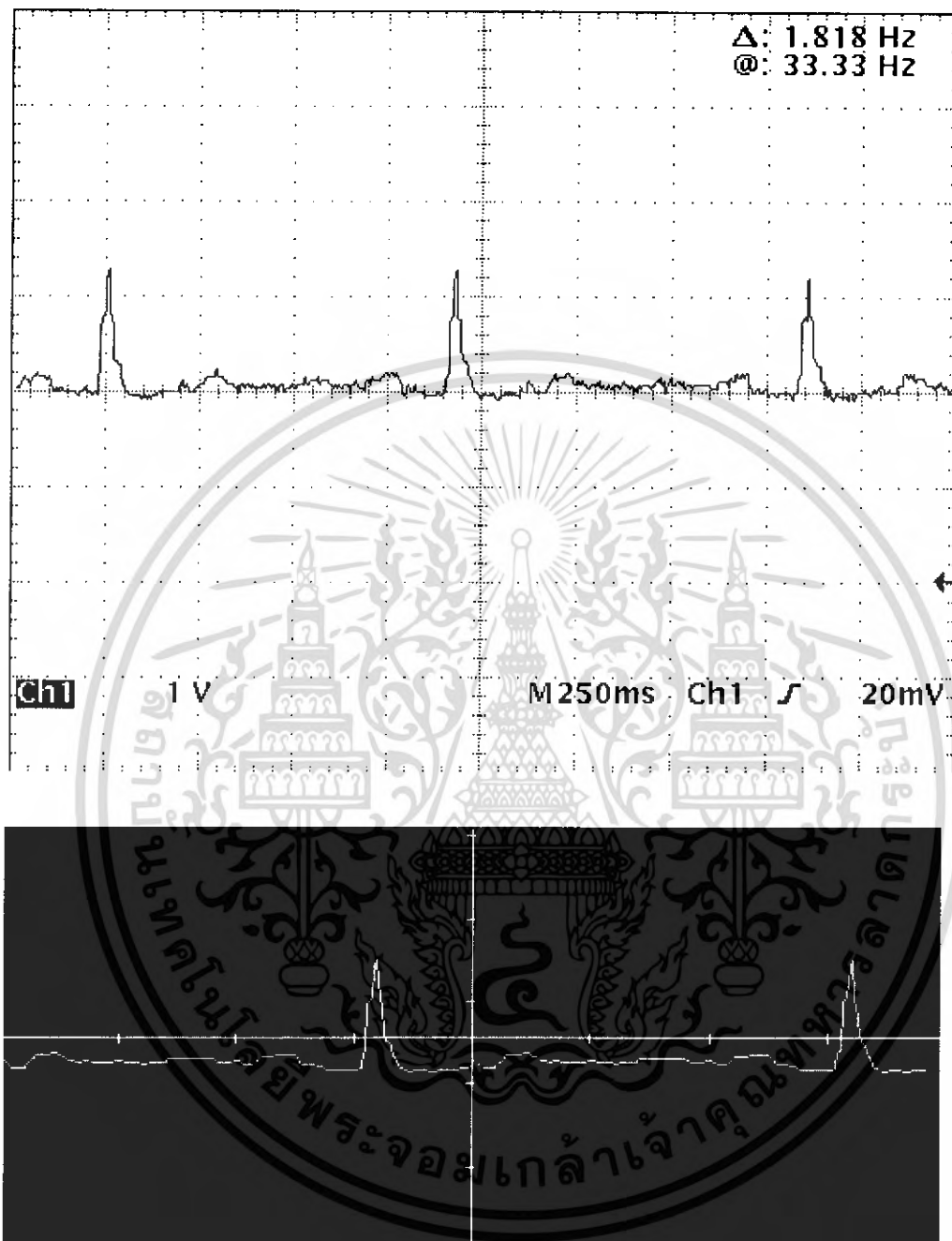
Lead2 Keypad \*2\*



รูปที่ 5.5.2 กราฟการเปรียบเทียบการแสดงผลระหว่าง Oscilloscope และ Program VB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Lead3 Keypad \*3\*

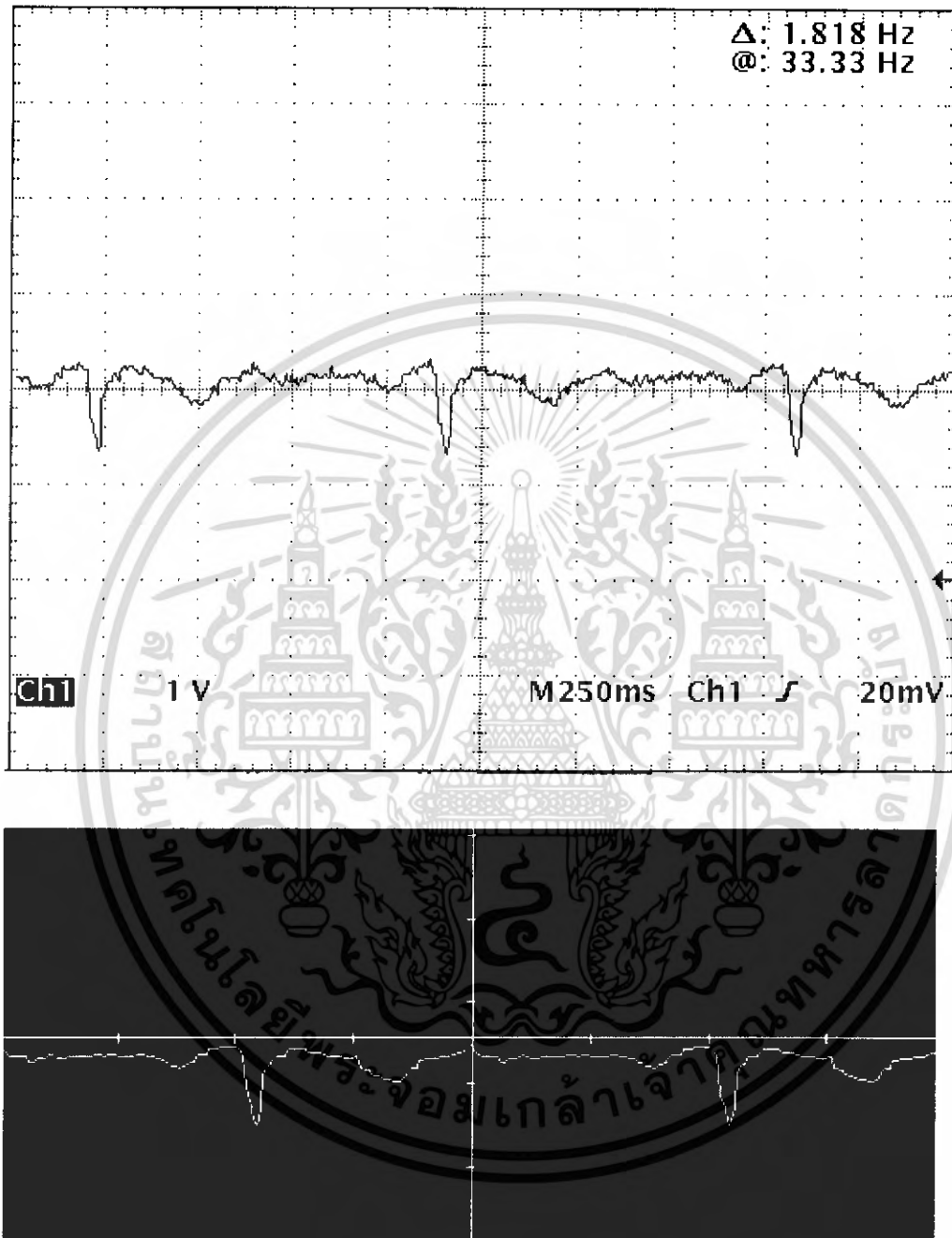


รูปที่ 5.5.3 กราฟการเปรียบเทียบการแสดงผลระหว่าง Oscilloscope และ Program VB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Augmented Leads

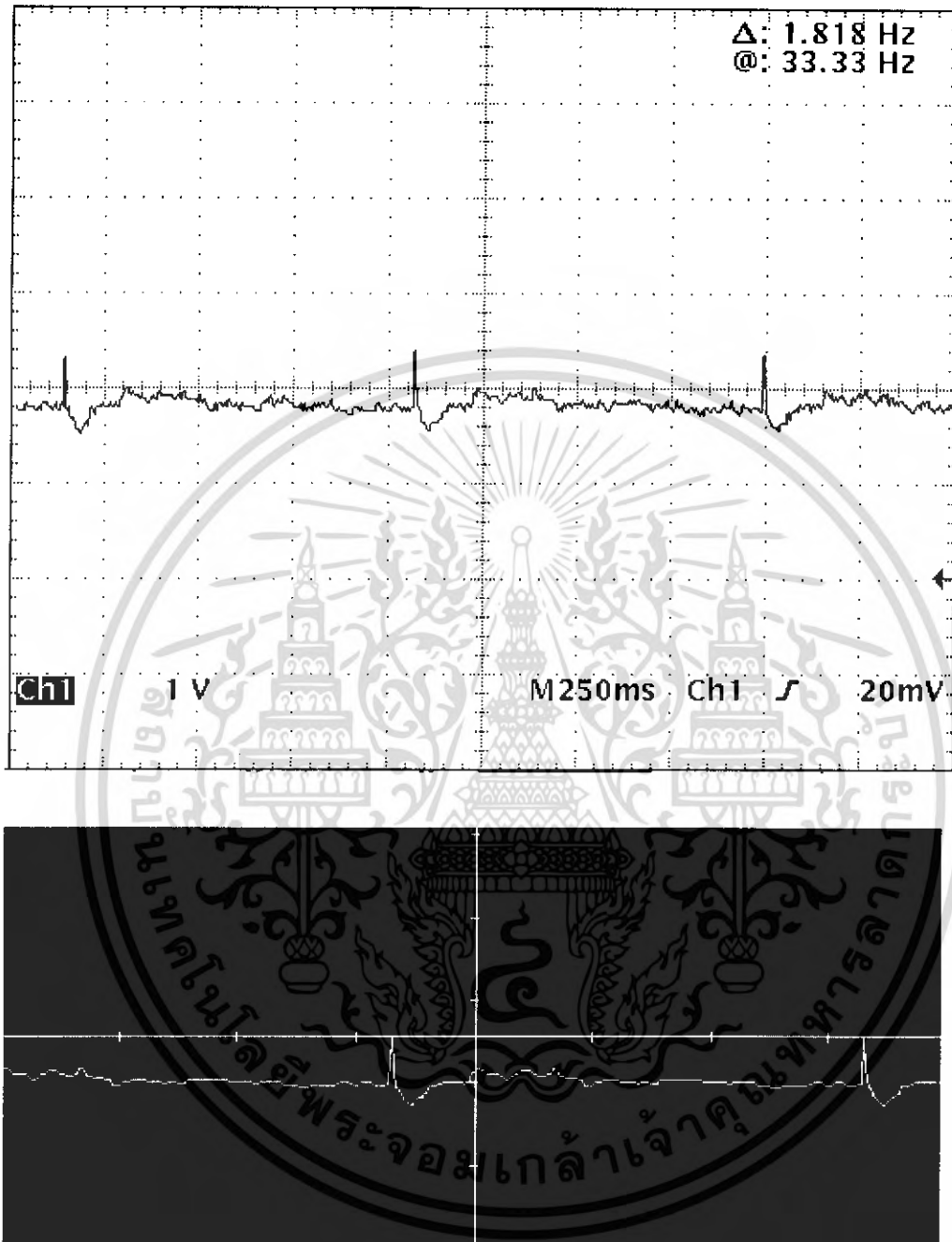
aVR Keypad \*4\*



รูปที่ 5.5.4 กราฟการเปรียบเทียบการแสดงผลระหว่าง Oscilloscope และ Program VB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

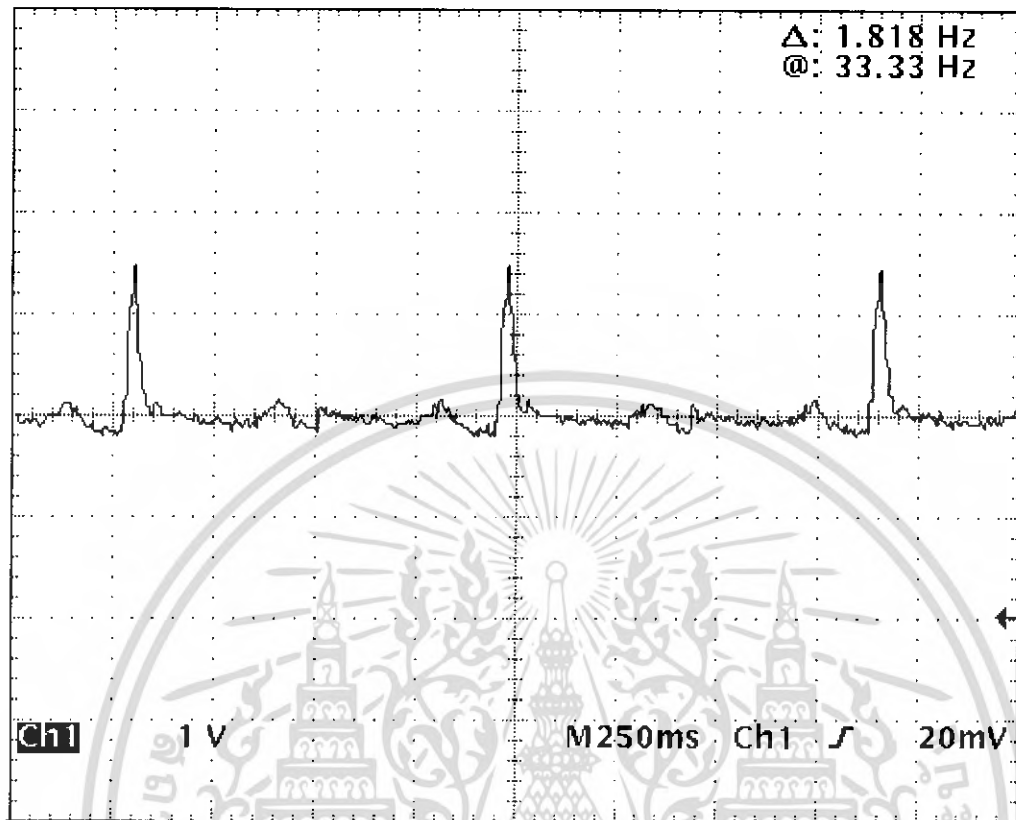
aVL Keypad \*5\*



รูปที่ 5.5.5 กราฟการเปรียบเทียบการแสดงผลระหว่าง Oscilloscope และ Program VB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

aVF Keypad \*6\*

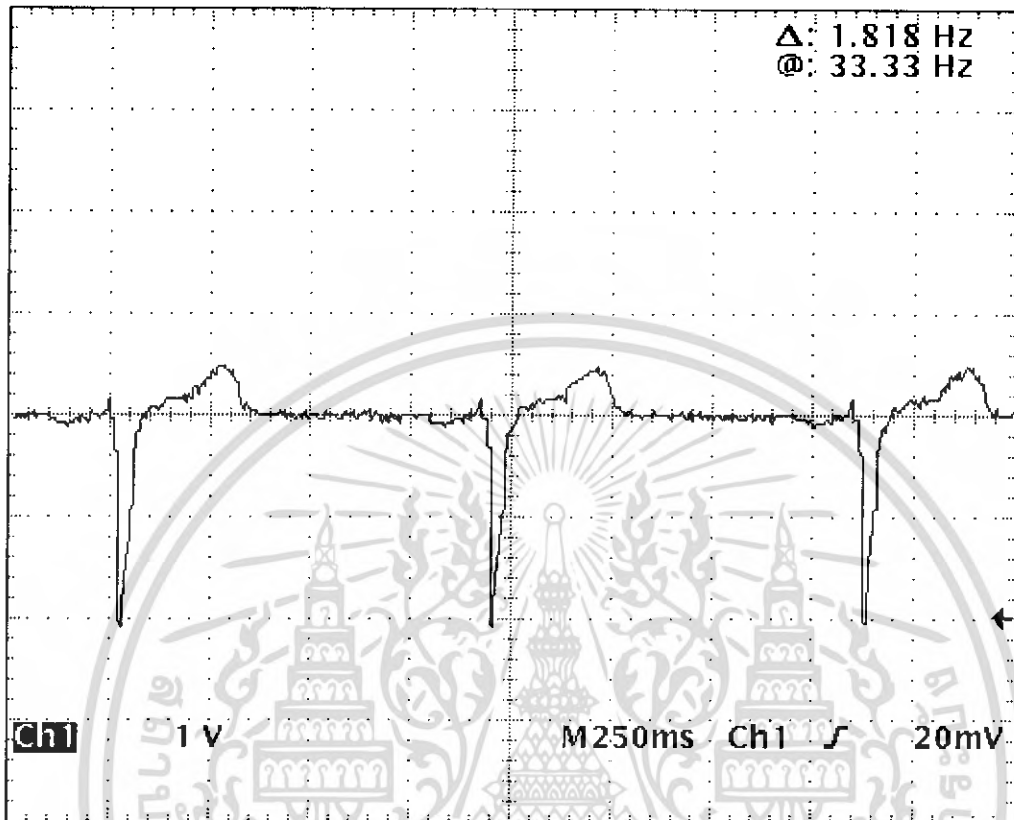


รูปที่ 5.5.6. กราฟการเปรียบเทียบการแสดงผลระหว่าง Oscilloscope และ Program VB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Unipolar Leads

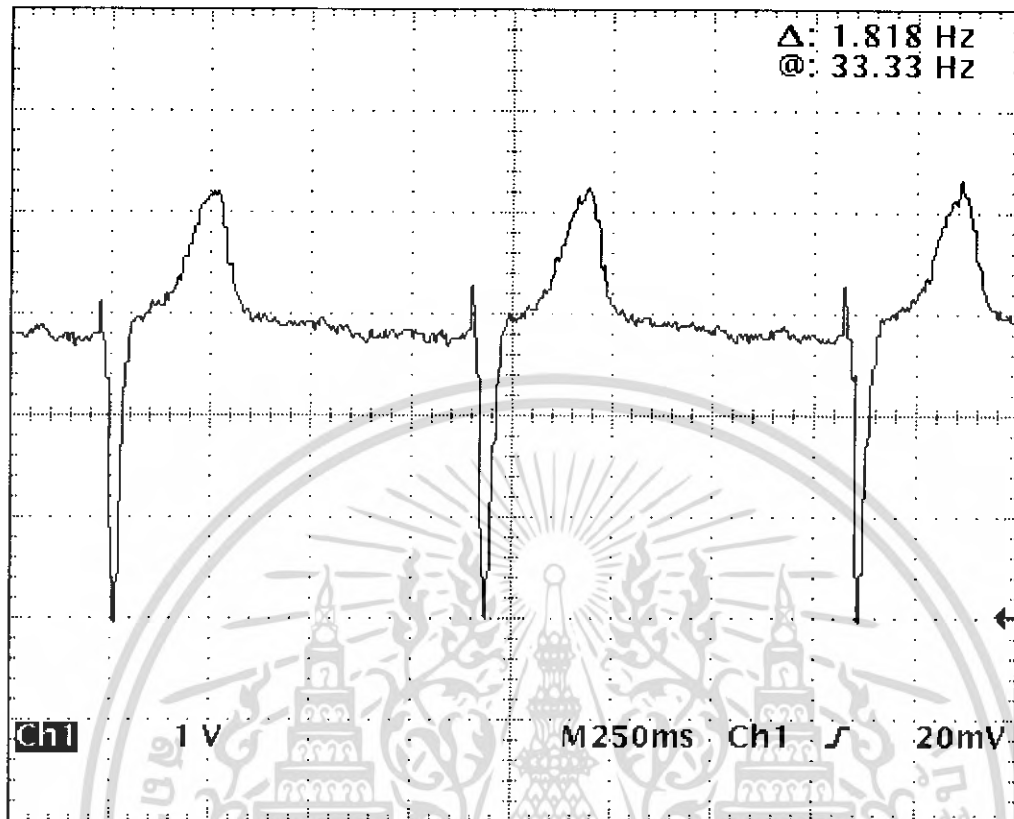
V1 Keypad \*7\*



รูปที่ 5.5.7 กราฟการเปรียบเทียบการแสดงผลระหว่าง Oscilloscope และ Program VB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

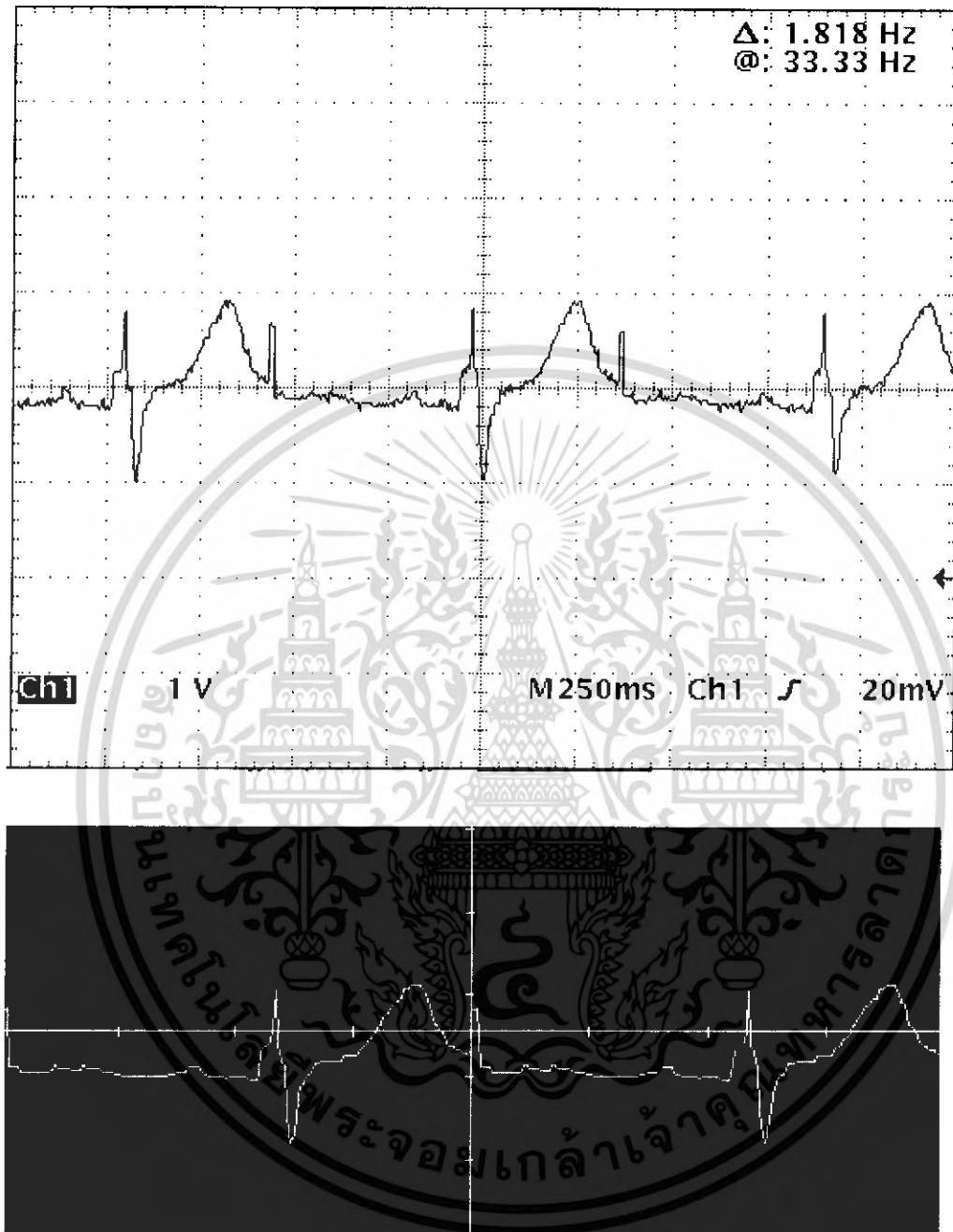
V2 Keypad \*8\*



รูปที่ 5.5.8 กราฟการเปรียบเทียบการแสดงผลระหว่าง Oscilloscope และ Program VB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

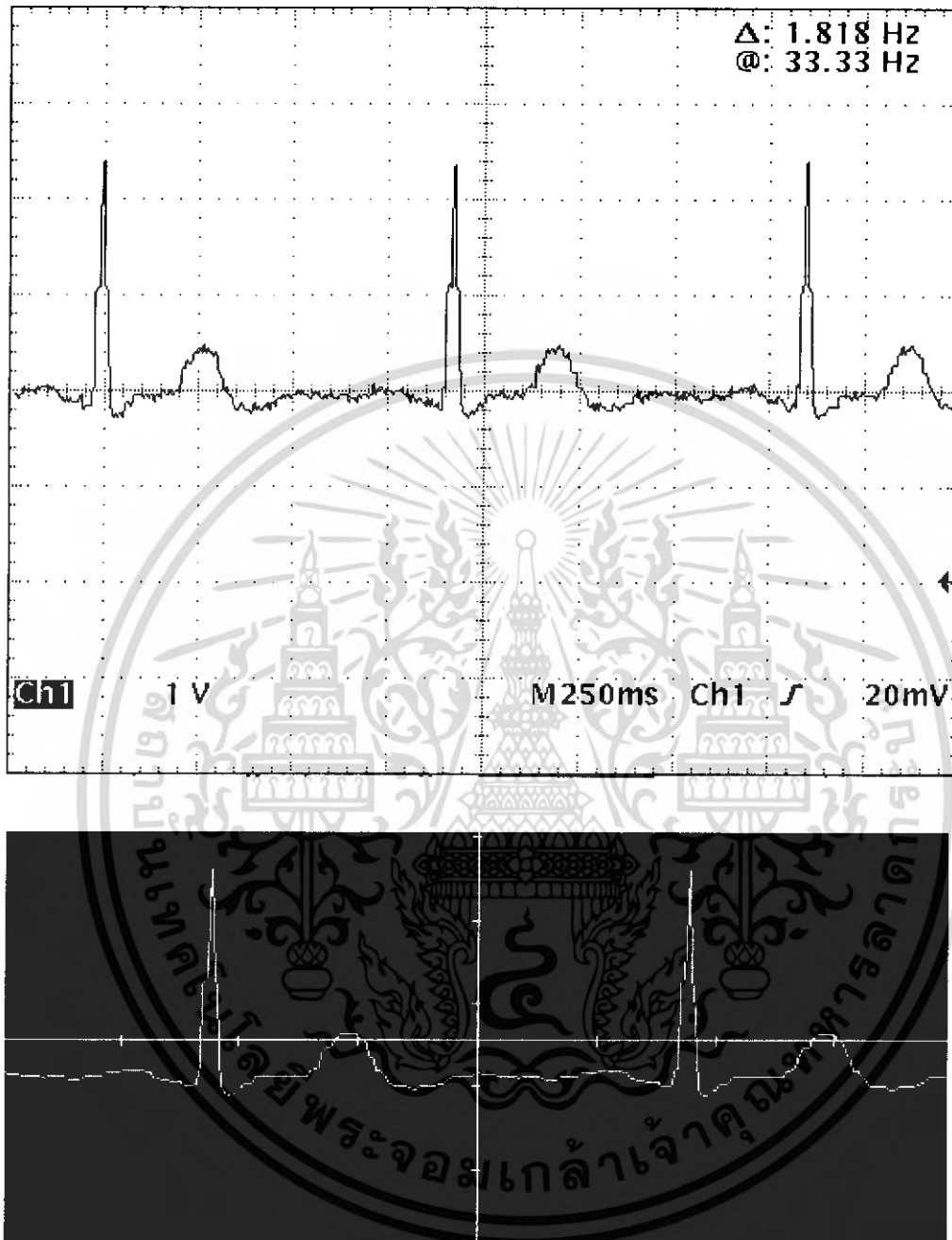
V3 Keypad \*9\*



รูปที่ 5.5.9 กราฟการเปรียบเทียบการแสดงผลระหว่าง Oscilloscope และ Program VB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

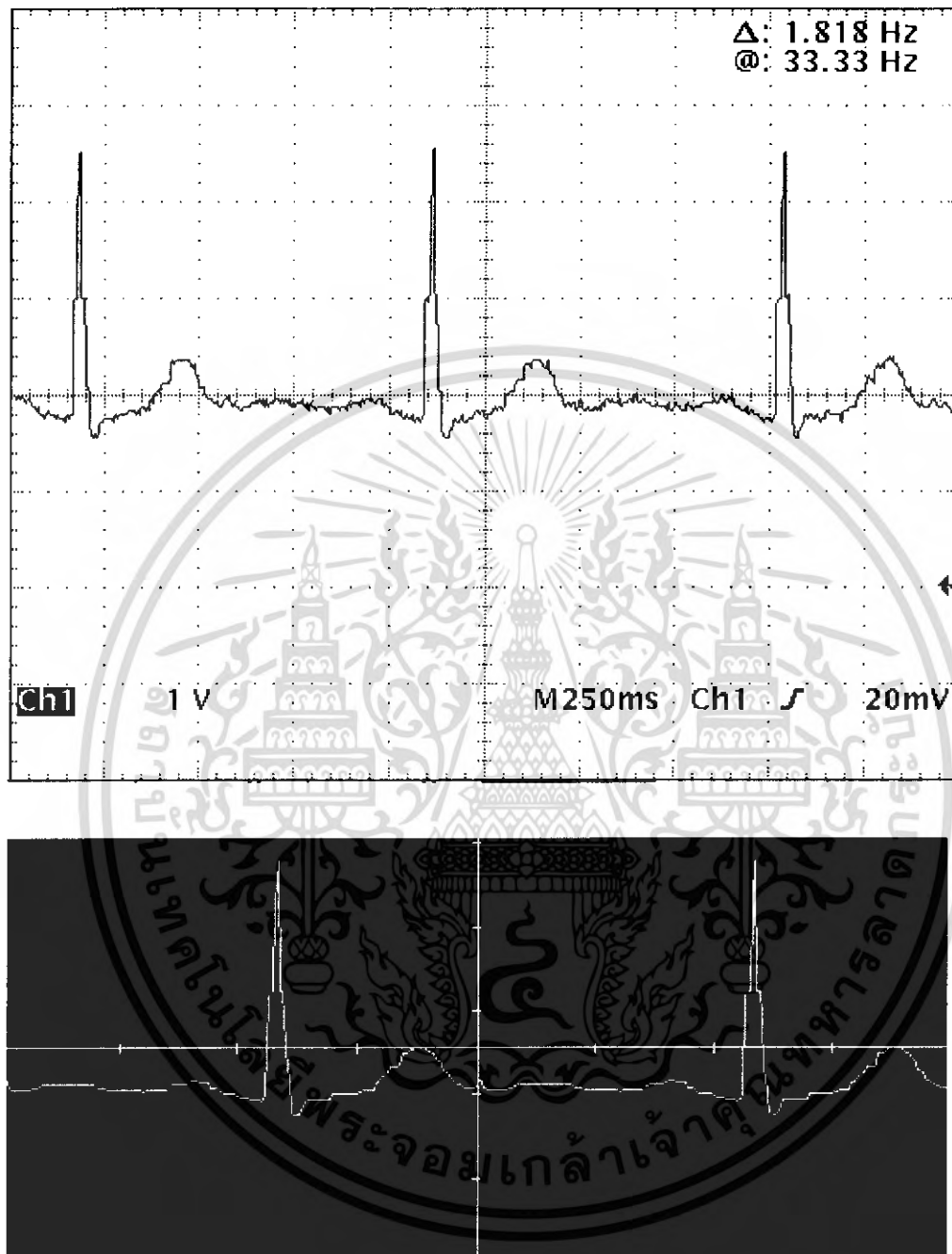
V4 Keypad \*10\*



รูปที่ 5.5.10 กราฟการเปรียบเทียบการแสดงผลระหว่าง Oscilloscope และ Program VB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

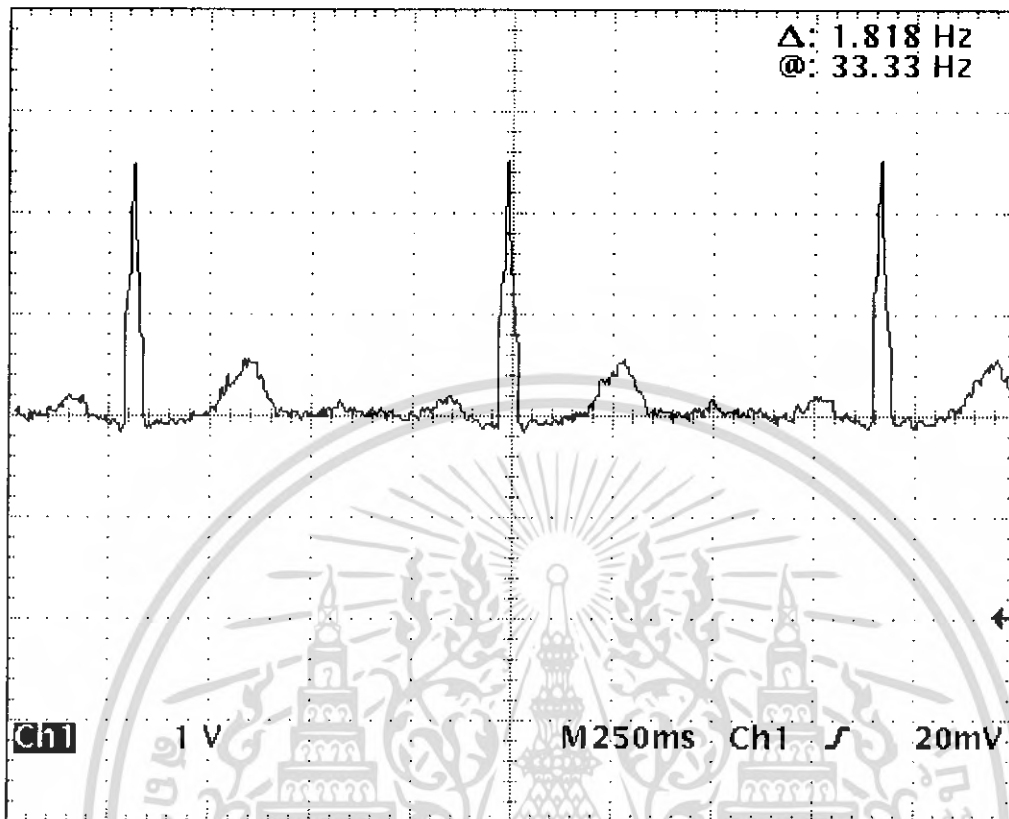
V5 Keypad \*11\*



รูปที่ 5.5.11 กราฟการเปรียบเทียบการแสดงผลระหว่าง Oscilloscope และ Program VB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

V6 Keypad \*12\*

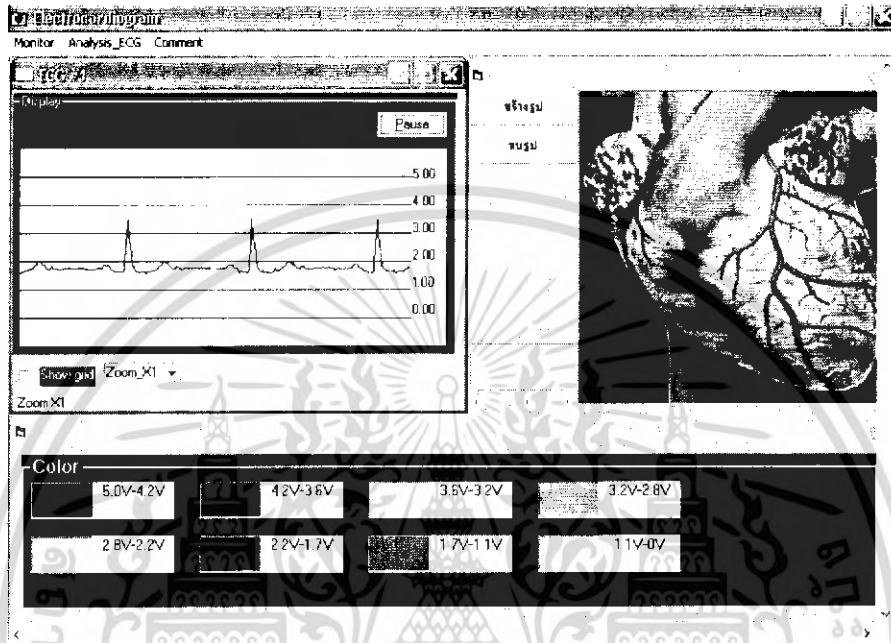


รูปที่ 5.5.12 กราฟการเปรียบเทียบการแสดงผลระหว่าง Oscilloscope และ Program VB

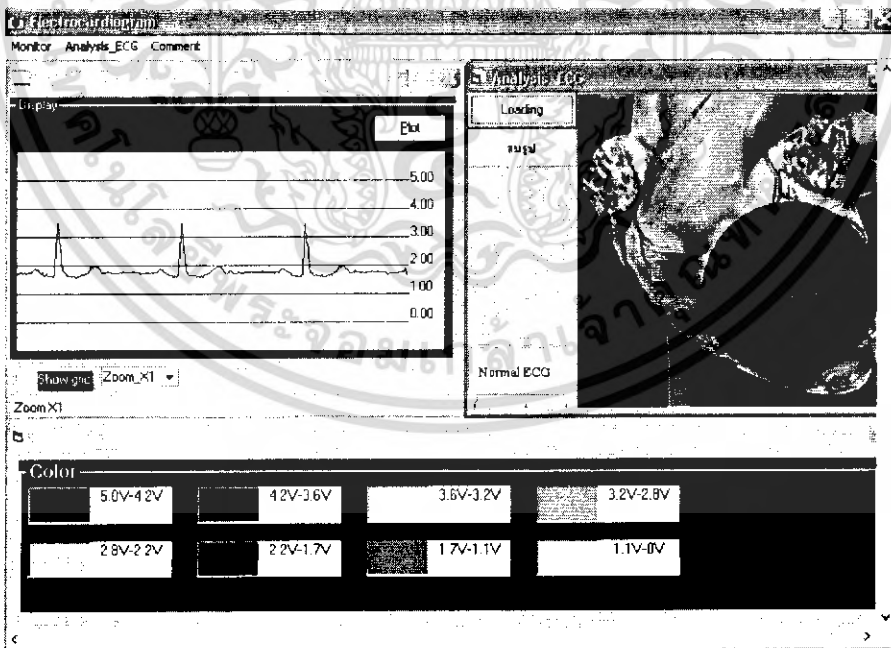
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 5.4 ผลการทดลองการวิเคราะห์สัญญาณ ECG

เราได้ทำการรับสัญญาณอนาล็อก ECG Lead 2 เข้ามาแล้วมาทำการวิเคราะห์และนำมาแสดงผลออกมาเป็นสีต่างๆ โดยเราได้แบ่งช่วงของสีตามสัญญาณ ECG ปกติของมนุษย์โดยเราได้ทำการตรวจสอบความสูงของสัญญาณ ECG แล้วนำมาแสดงผลทางสีแทน

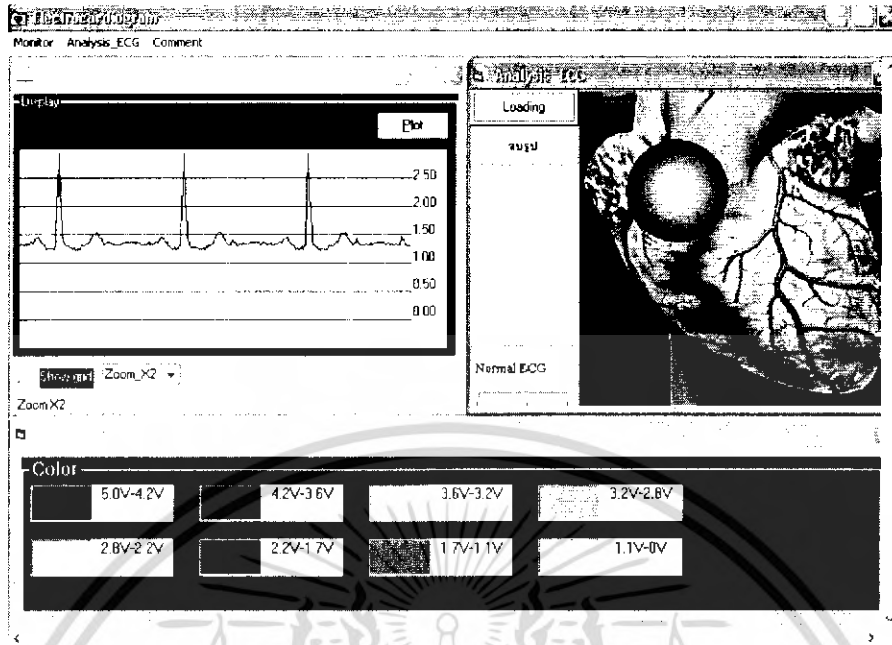


รูปที่ 5.6.1 แสดงการรับสัญญาณ ECG Lead 2 (Display) เข้ามา

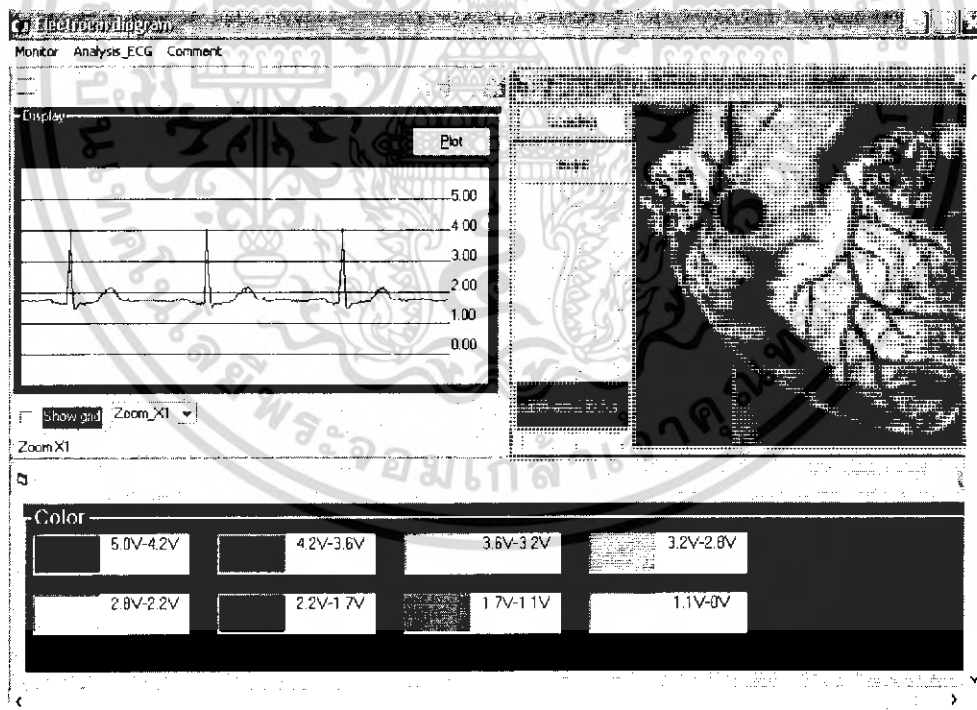


รูปที่ 5.6.2 แสดงการรับสัญญาณ ECG Lead 2 (Display) เข้ามาแล้วแสดงผลตรงส่วน Analysis ส่วน AV-node เมื่อสัญญาณมีระดับปกติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

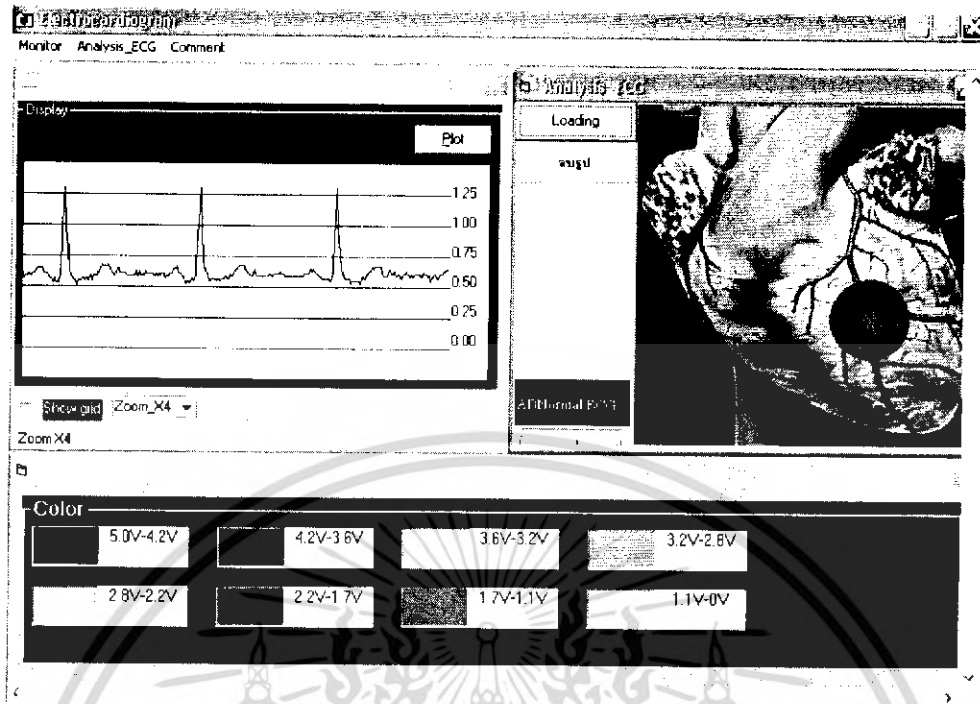


รูปที่ 5.6.3 แสดงการรับสัญญาณ ECG Lead 2 (Display) เข้ามาแล้วแสดงผลตรงส่วน Analysis ส่วน SA-node เมื่อสัญญาณมีระดับปกติแต่สัญญาณมีความสูงลดลง



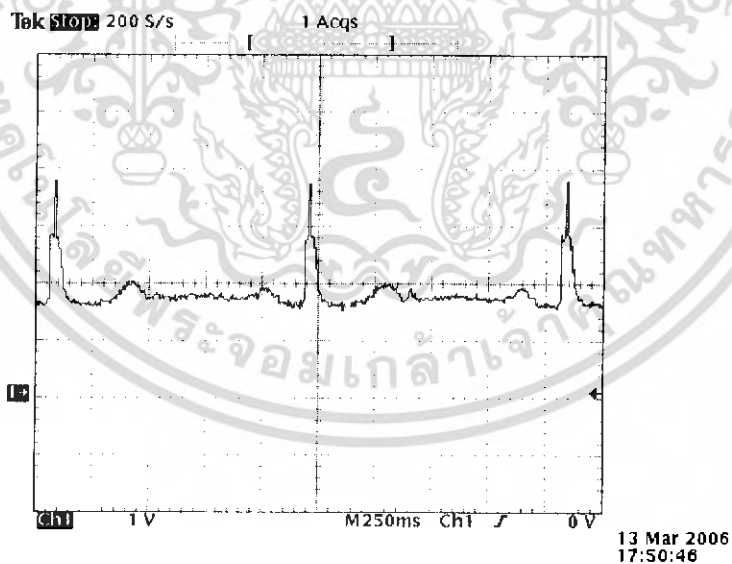
รูปที่ 5.6.4 แสดงการรับสัญญาณ ECG Lead 2 (Display) เข้ามาแล้วแสดงผลตรงส่วน Analysis ส่วน SA-node เมื่อสัญญาณมีความสูงเกินค่าผิดปกติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.6.5 แสดงการรับสัญญาณ ECG Lead 2 (Display) เข้ามาแล้วแสดงผลตรงส่วน Analysis ส่วน AV-node เมื่อสัญญาณมีความต่ำกว่าค่าผิดปกติ

### 5.5 ผลการทดลองการปรับ Heart Rate



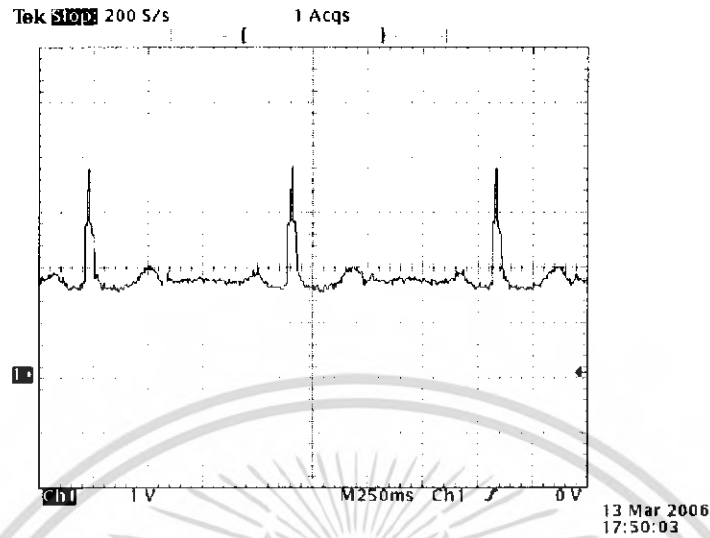
รูปที่ 5.7.1 แสดง Heart Rate = 60 Rate/Min

การคิดอัตราการ Sampling สัญญาณ จากข้อมูล 200 ค่า

$$\text{ค่าที่ Sampling} = 60 / (200 * 60) = 5\text{ms}$$

$$\text{สัญญาณมีความถี่} = 1 / (5\text{ms} * 200) = 1\text{Hz}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

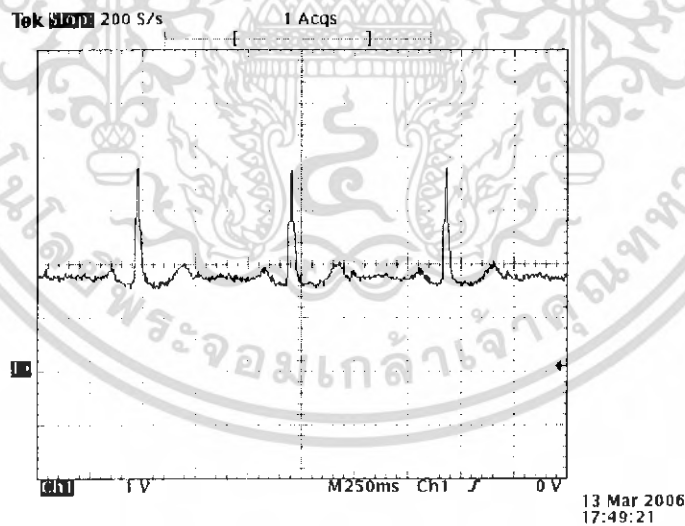


รูปที่ 5.7.2 แสดง Heart Rate = 75 Rate/Min

การคิดอัตราการ Sampling สัญญาณ จากข้อมูล 200 ค่า

$$\text{ค่าที่ Sampling} = 75 / (200 * 60) = 4\text{ms}$$

$$\text{สัญญาณมีความถี่} = 1 / (4\text{ms} * 200) = 1.25\text{Hz}$$



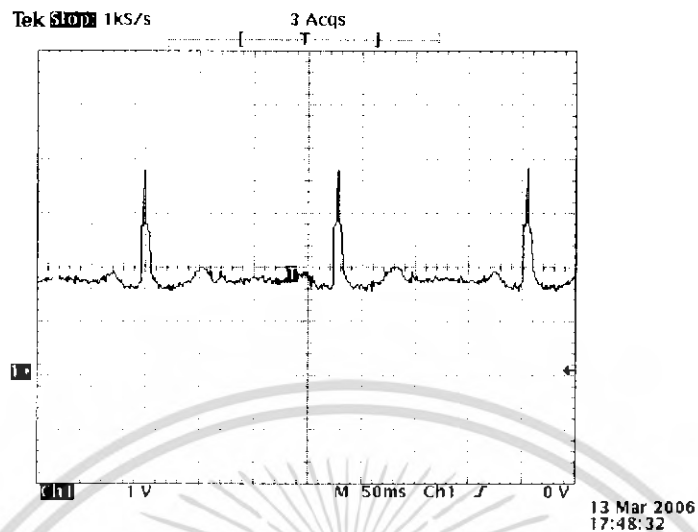
รูปที่ 5.7.3 แสดง Heart Rate = 100 Rate/Min

การคิดอัตราการ Sampling สัญญาณ จากข้อมูล 200 ค่า

$$\text{ค่าที่ Sampling} = 100 / (200 * 60) = 3\text{ms}$$

$$\text{สัญญาณมีความถี่} = 1 / (3\text{ms} * 200) = 1.67\text{Hz}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.7.4 แสดง Heart Rate = 120 Rate/Min

การคิดอัตราการ Sampling สัญญาณ จากข้อมูล 200 ค่า

$$\text{ค่าที่ Sampling} = 120 / (200 * 60) = 250 \mu\text{s}$$

$$\text{สัญญาณมีความถี่} = 1 / (250 \mu\text{s} * 200) = 2 \text{ Hz}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### บทที่ 6 สรุปผลการทดลอง

จากผลการทดลอง DAC พบว่าเมื่อเราป้อน Input ตั้งแต่ 0-255 พบว่าได้ Output เป็น 0-4.9V ซึ่งเป็น Linear Output แต่ค่า Input 255 ไม่ได้ Output 5 Volt ตามต้องการ แต่เมื่อพิจารณาข้อมูลที่ต้องใช้พบว่าค่าข้อมูลสูงสุดมีค่าไม่ถึง 250 ดังนั้นจึงไม่เป็นปัญหาต่อค่า Output

ผลการติดต่อแบบบัส I2C พบว่าได้สภาวะบัสว่าง สภาวะเริ่มต้น สภาวะส่ง Logic 1 สภาวะส่ง Logic 0 และสภาวะหยุดส่งตามทฤษฎีทุกอย่าง

ผลการแสดงข้อมูล ECG Lead ต่างๆที่แสดงทาง Oscilloscope เทียบกับ VB พบว่าได้รูปเดียวกันและตรงกับข้อมูลที่เรานำไปเก็บไว้ใน EEPROM ทุกอย่าง

ผลการแสดงและวิเคราะห์สัญญาณ ECG ได้ทำการแสดงออกมาเป็นการแพร่กระจายของสัญญาณ SA-node และแพร่ของสัญญาณ AV-node โดยได้แสดงทั้งแบบสัญญาณ ECG ที่ปกติและสัญญาณ ECG ที่มีความผิดปกติ

## หนังสืออ้างอิง

1. ยงยุทธ สหัทธกุล, "ECG ทางคลินิก", ปี 2544, หน้า 3-11
2. สัจจะ จรัสรุ่งรวีร์, "คู่มือการสร้างแอปพลิเคชันด้วย Visual Basic 6", ปี 2542, หน้า 3-188
3. อภิชาติ ภู่วัฒน์, "เริ่มต้นเขียน โปรแกรมติดต่อและควบคุมฮาร์ดแวร์ด้วย Visual Basic", ปี 2546, หน้า 5-18, 157-222
4. อภิชาติ สุครุธรรม, "Electrocardiogram", ปี 2525, หน้า 1-45

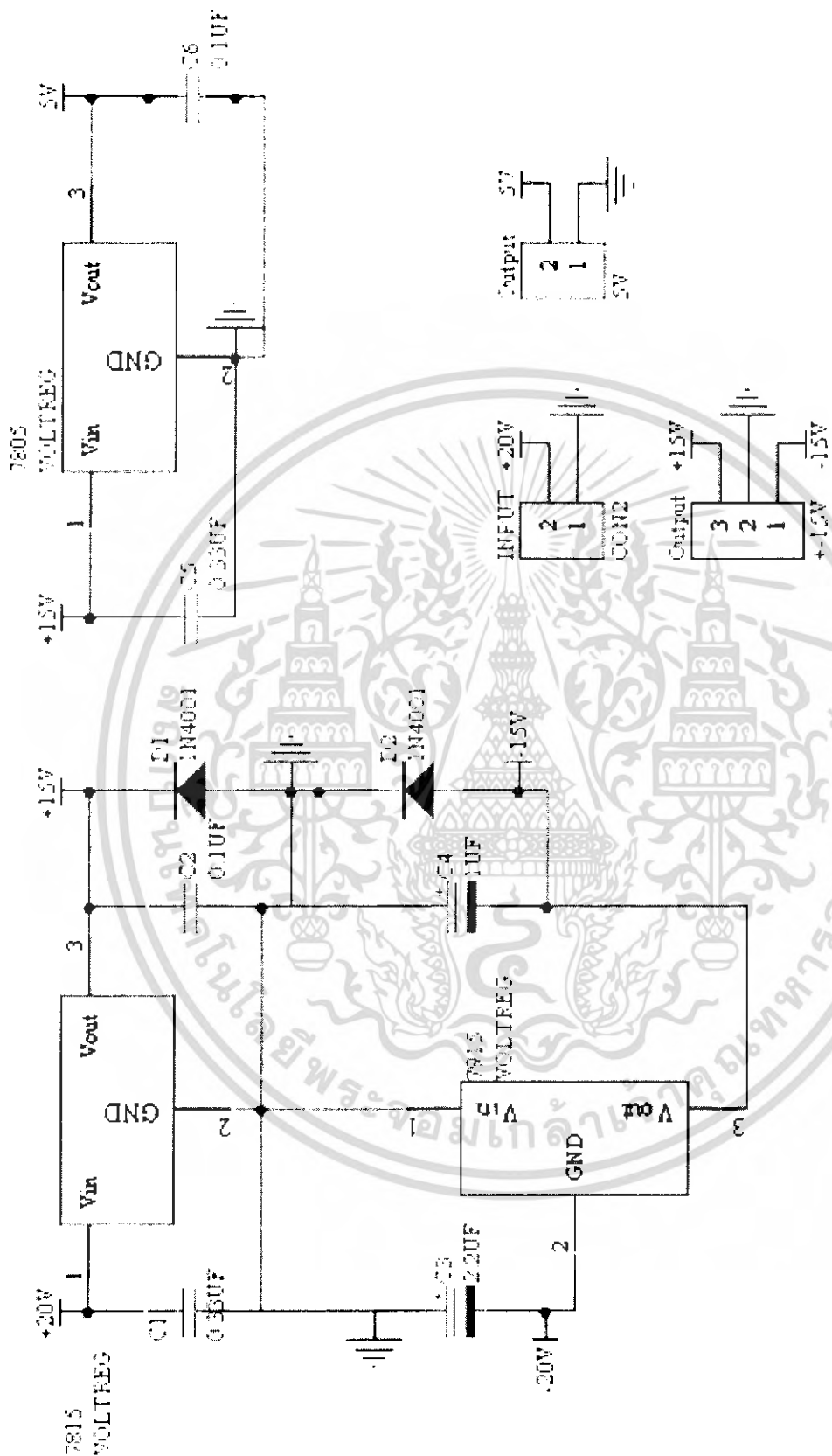


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



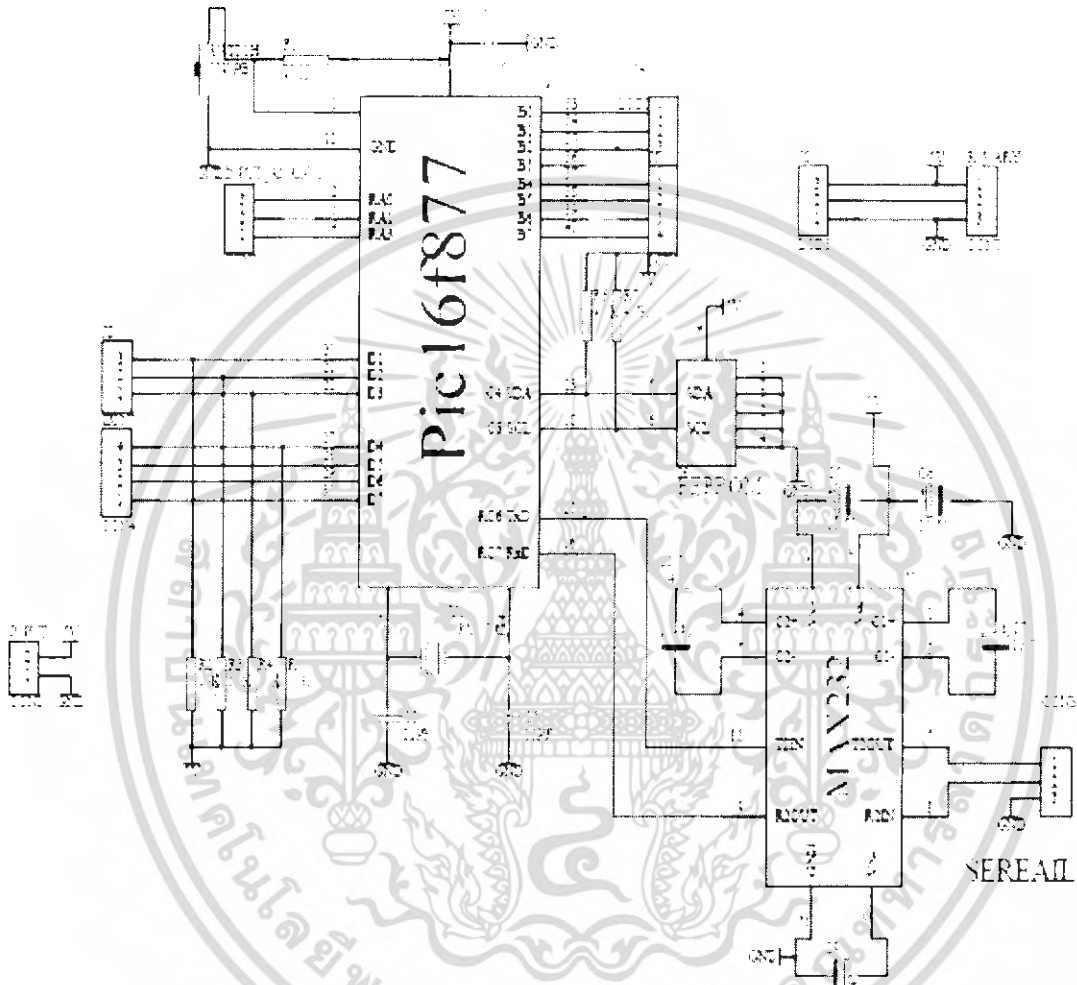
## ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



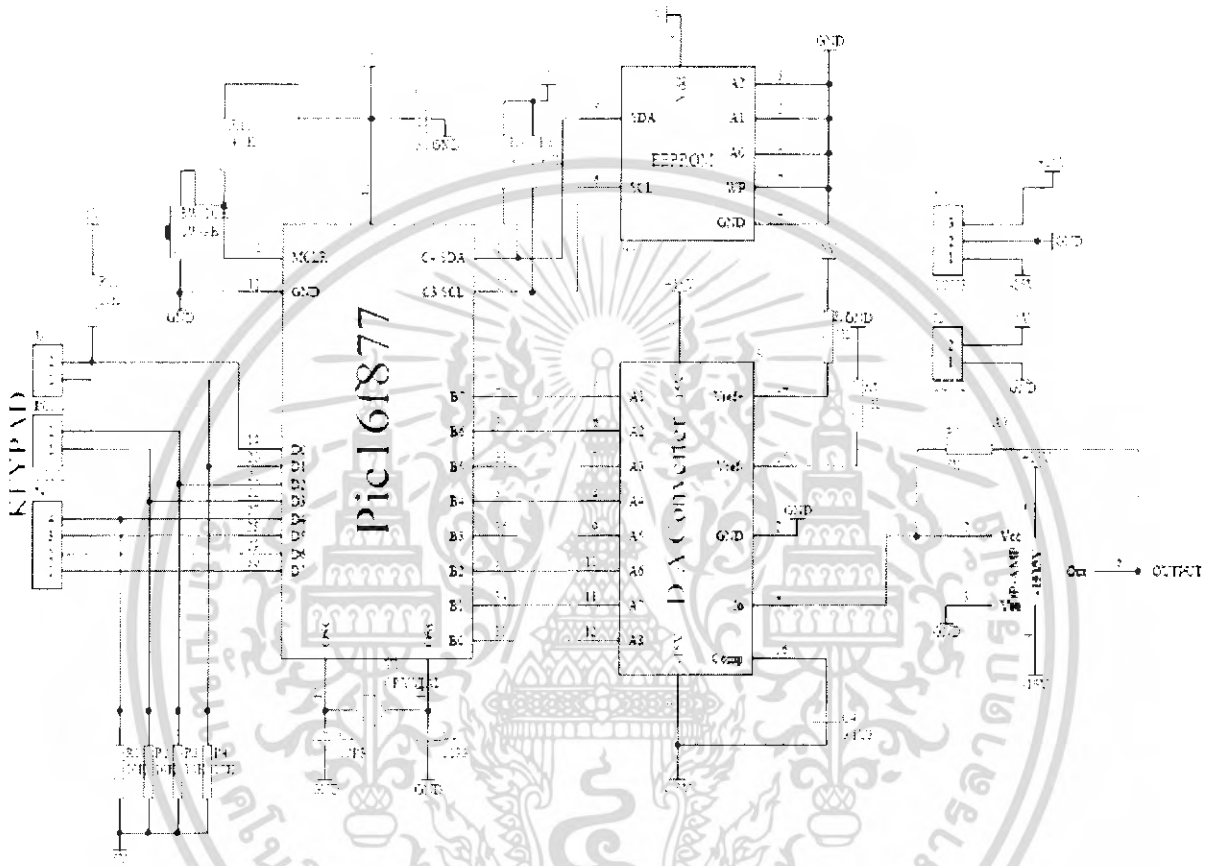
รูปวงจร Regulator +15V,-15Vและ 5V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปวงจรใน Mode 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปวงจรใน Mode 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## วิธีการใช้งาน เครื่อง ECG Signals System

1. เสียบปลั๊กไฟและกด Switch On ด้านหลังของเครื่อง
2. ต่อสาย Serial Port เข้าด้านหลังของเครื่อง
3. ปรับ Switch เลือก Heart Rate ตามที่เราต้องการ (60,75,100,120)
4. กด Keypad เพื่อเลือกกราฟสัญญาณ Normal ECG 12 Lead โดยกด \* กราฟ ECG \* เช่น \*1\* คือ การเลือกกราฟ ECG Lead 1 โดยเลือกได้ทั้งหมด 13 กราฟ
5. ก็จะได้สัญญาณ ECG ออกมาทางช่อง Output (สีแดง) สามารถใช้ Oscilloscope วัดสัญญาณได้
6. เมื่อเราต้องการที่จะรับสัญญาณอนาล็อกเข้ามาให้เรากด Switch Analog แล้วนำสัญญาณ Analog ต่อเข้ามาทางช่อง Analog Input (สีเหลือง) ค่าของสัญญาณนี้จะถูกส่งค่าไปทาง Serial Port ที่ต่อไว้ที่ PC

7. เมื่อใช้งานเสร็จก็ทำการปิด Switch Off พร้อมทั้งถอดปลั๊กสายไฟออกด้วย

หมายเหตุ

- \*1\* = ECG Lead 1
- \*2\* = ECG Lead 2
- \*3\* = ECG Lead3
- \*4\* = ECG aVR
- \*5\* = ECG aVL
- \*6\* = ECG aVF
- \*7\* = ECG V1
- \*8\* = ECG V2
- \*9\* = ECG V3
- \*10\* = ECG V4
- \*11\* = ECG V5
- \*12\* = ECG V6
- \*13\* = ECG All Lead

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## วิธีการใช้งานโปรแกรม ECG SIGNALS SYSTEM

1. กดเลือก โปรแกรม ECG SIGNALS SYSTEM แล้วก็ทำการกด Set Baud Rate และ Set Port
2. กด Connected โปรแกรมก็จะเริ่มใช้งานได้
3. ถ้ากด Select Leads ก็จะเป็นการอ่านค่าของกราฟที่เก็บไว้ใน EEprom มาแสดงแล้วก็ทำการกด Plot ECG กราฟสัญญาณก็จะทำการ Plot แสดงตรงส่วน Display ECG
4. ตรงส่วนที่เป็นการเลือกสัญญาณ ECG ที่หลายๆ กราฟ ให้กดเลือกว่าต้องการดูกราฟไหนก่อน (มีให้เลือก 3 แบบ 1.Bipolar Leads 2.Augmented Leads 3.Unipolar Leads)
5. หลังจากนั้นก็ทำการกดปุ่ม Display Leads ก็จะแสดงหน้าจอแค่ Leads ตามที่เราได้ทำการเลือกเอาไว้
6. กดปุ่ม Read EEprom ก็จะอ่านค่าข้อมูลจาก EEprom มาแสดงไว้ที่ Text Box
7. กด Plot ก็จะทำการแสดงกราฟสัญญาณ ECG ขึ้นมา
8. ตรงหน้าโปรแกรมหลักถ้าเราเลือกตรงส่วนวิเคราะห์สัญญาณ ก็จะแสดง หน้าจอวิเคราะห์สัญญาณขึ้นมา
9. โดยหน้าจอนี้จะมีส่วน Plot ECG ที่รับมาจากช่อง Analog Input มาแสดง และส่วนวิเคราะห์สัญญาณ เมื่อเรากด สร้างรูปก็จะทำการแสดงภาพ 2 มิติ เป็นสีแปรจาก SA-Node ไปยัง AV-Node (แสดงเป็นสีต่างๆ)
10. เมื่อเรากดวิเคราะห์สัญญาณแล้วก็มาทำการดูค่าสีเปรียบเทียบว่าค่าของสีอยู่ในช่วงไหน และมีความผิดปกติหรือไม่

## วิธีการใช้โปรแกรมเขียนค่าข้อมูลไปยัง EEPROM

1. เปิด โปรแกรม Write Data in EEPROM จะขึ้น โปรแกรมดังรูปที่ 4.2.1
2. กดเลือก Baud, Port และ Connect
3. เลือกกราฟที่จะเขียนลงใน EEPROM
4. พิมพ์ค่าที่ต้องการเขียนลงในช่องด้านบนดังรูปที่ 4.2.2 แล้วกดปุ่ม Start
5. เมื่อโปรแกรมทำการส่งข้อมูลเสร็จจะมีหน้าต่างบอกดังรูปที่ 4.2.3 ให้ตอบ OK เป็นอันเสร็จสิ้นการเขียนข้อมูลลงใน EEPROM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## SOURCE CODE MODE1

```

#include <16f877.h>
#define * =16
#define adc=8
#define fuses Hs,NOWDT,NOPROTECT,nolvp
#define use delay(clock=2000000)
#define use rs232(baud=9600, xmit=PIN_C6, rcv=PIN_C7) // Jumpers: 8 to 11, 7 to 12
#define use fast_io(B)
#include <24512.c>
#define BUFFER_SIZE 32
byte buffer[BUFFER_SIZE];
byte next_in = 0;
byte next_out = 0;
#define int_rda
void serial_isr() {
    int t;
    buffer[next_in]=getc();
    t=next_in;
    next_in=(next_in+1) % BUFFER_SIZE;
    if(next_in==next_out)
        next_in=t; // Buffer full !!
}
#define bkbhit (next_in!=next_out)
byte bgetc() {
    byte c;
    while(!bkbhit);
    c=buffer[next_out];
    next_out=(next_out+1) % BUFFER_SIZE;
    return(c);
}
long int adress;
byte buf_in,buf_out;
int i;
byte a,x,y,z;
void Write_Garp1()
{
    adress=0;i=1;
    do {
        buf_in=bgetc();

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        x=(buf_in-'0')*100;
        buf_in=bgetc();
        y=(buf_in-'0')*10;
        buf_in=bgetc();
        z=(buf_in-'0')*1;
        buf_in=x+y+z;
        write_ext_eeprom(address,buf_in);
        address++;
        i++;
    }while (address<200);
}

```

```
void Write_Garp20
```

```

{
    address=200;i=1;
    do {
        buf_in=bgetc();
        x=(buf_in-'0')*100;
        buf_in=bgetc();
        y=(buf_in-'0')*10;
        buf_in=bgetc();
        z=(buf_in-'0')*1;
        buf_in=x+y+z;
        write_ext_eeprom(address,buf_in);
        address++;
        i++;
    }while (address<400);
}

```

```
void Write_Garp30
```

```

{
    address=400;i=1;
    do {
        buf_in=bgetc();
        x=(buf_in-'0')*100;
        buf_in=bgetc();
        y=(buf_in-'0')*10;
        buf_in=bgetc();
        z=(buf_in-'0')*1;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        buf_in=x+y+z;
        write_ext_eeprom(address,buf_in);
        adress++;
        i++;
    }while (adress<600);
}

```

```
void Write_Garp40
```

```

{
    adress=3600;i=1;
    do {
        buf_in=bgetc();
        x=(buf_in-'0')*100;
        buf_in=bgetc();
        y=(buf_in-'0')*10;
        buf_in=bgetc();
        z=(buf_in-'0')*1;
        buf_in=x+y+z;
        write_ext_eeprom(address,buf_in);
        adress++;
        i++;
    }while (adress<3800);
}

```

```
void Write_Garp50
```

```

{
    adress=3800;i=1;
    do {
        buf_in=bgetc();
        x=(buf_in-'0')*100;
        buf_in=bgetc();
        y=(buf_in-'0')*10;
        buf_in=bgetc();
        z=(buf_in-'0')*1;
        buf_in=x+y+z;
        write_ext_eeprom(address,buf_in);
        adress++;
        i++;
    }while (adress<4000);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}

void Write_Garp6()
{
    adress=4000;i=1;
    do {
        buf_in=bgetc();
        x=(buf_in-'0')*100;
        buf_in=bgetc();
        y=(buf_in-'0')*10;
        buf_in=bgetc();
        z=(buf_in-'0')*1;
        buf_in=x+y+z;
        write_ext_eeprom(adress,buf_in);
        adress++;
        i++;
    }while (adress<4200);
}

void Write_Garp7()
{
    adress=1200;i=1;
    do {
        buf_in=bgetc();
        x=(buf_in-'0')*100;
        buf_in=bgetc();
        y=(buf_in-'0')*10;
        buf_in=bgetc();
        z=(buf_in-'0')*1;
        buf_in=x+y+z;
        write_ext_eeprom(adress,buf_in);
        adress++;
        i++;
    }while (adress<1400);
}

void Write_Garp8()
{
    adress=1400;i=1;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

do {
    buf_in=bgetc();
    x=(buf_in-'0')*100;
    buf_in=bgetc();
    y=(buf_in-'0')*10;
    buf_in=bgetc();
    z=(buf_in-'0')*1;
    buf_in=x+y+z;
    write_ext_eeprom(address,buf_in);
    adress++;
    i++;
}while (adress<1600);
}

void Write_Garp90
{
    adress=4600;i=1;
    do {
        buf_in=bgetc();
        x=(buf_in-'0')*100;
        buf_in=bgetc();
        y=(buf_in-'0')*10;
        buf_in=bgetc();
        z=(buf_in-'0')*1;
        buf_in=x+y+z;
        write_ext_eeprom(address,buf_in);
        adress++;
        i++;
    }while (adress<4800);
}

void Write_Garp100
{
    adress=1800;i=1;
    do {
        buf_in=bgetc();
        x=(buf_in-'0')*100;
        buf_in=bgetc();
        y=(buf_in-'0')*10;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        buf_in=bgetc();
        z=(buf_in-'0')*1;
        buf_in=x+y+z;
        write_ext_eeprom(address,buf_in);
        adress++;
        i++;
    }while (adress<2000);
}

```

```
void Write_Garp110
```

```

{
    adress=2000;i=1;
    do {
        buf_in=bgetc();
        x=(buf_in-'0')*100;
        buf_in=bgetc();
        y=(buf_in-'0')*10;
        buf_in=bgetc();
        z=(buf_in-'0')*1;
        buf_in=x+y+z;
        write_ext_eeprom(address,buf_in);
        adress++;
        i++;
    }while (adress<2200);
}

```

```
void Write_Garp120
```

```

{
    adress=2200;i=1;
    do {
        buf_in=bgetc();
        x=(buf_in-'0')*100;
        buf_in=bgetc();
        y=(buf_in-'0')*10;
        buf_in=bgetc();
        z=(buf_in-'0')*1;
        buf_in=x+y+z;
        write_ext_eeprom(address,buf_in);
        adress++;
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        i++;
    }while (adress<2400);
}

void Read_Garp10
{
    adress=0;
    do {
        buf_out=read_ext_eeprom(adress);
        printf("\r\n%3U",buf_out);
        adress++;
    }while (adress<200);
}

void Read_Garp20
{
    adress=200;
    do {
        buf_out=read_ext_eeprom(adress);
        printf("\r\n%3U",buf_out);
        adress++;
    }while (adress<400);
}

void Read_Garp30
{
    adress=400;
    do {
        buf_out=read_ext_eeprom(adress);
        printf("\r\n%3U",buf_out);
        adress++;
    }while (adress<600);
}

void Read_Garp40
{
    adress=3600;
    do {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        buf_out=read_ext_eeprom(address);
        printf("\r\n%3U",buf_out);
        adress++;
    }while (adress<3800);
}

```

```

void Read_Garp50

```

```

{
    adress=3800;
    do {
        buf_out=read_ext_eeprom(address);
        printf("\r\n%3U",buf_out);
        adress++;
    }while (adress<4000);
}

```

```

void Read_Garp60

```

```

{
    adress=4000;
    do {
        buf_out=read_ext_eeprom(address);
        printf("\r\n%3U",buf_out);
        adress++;
    }while (adress<4200);
}

```

```

void Read_Garp70

```

```

{
    adress=1200;
    do {
        buf_out=read_ext_eeprom(address);
        printf("\r\n%3U",buf_out);
        adress++;
    }while (adress<1400);
}

```

```

void Read_Garp80

```

```

{
    adress=1400;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

do {
    buf_out=read_ext_eeprom(address);
    printf("\r\n%3U",buf_out);
    adress++;
}while (adress<1600);
}

```

```

void Read_Garp90

```

```

{
    adress=4600;
    do {
        buf_out=read_ext_eeprom(address);
        printf("\r\n%3U",buf_out);
        adress++;
    }while (adress<4800);
}

```

```

void Read_Garp100

```

```

{
    adress=1800;
    do {
        buf_out=read_ext_eeprom(address);
        printf("\r\n%3U",buf_out);
        adress++;
    }while (adress<2000);
}

```

```

void Read_Garp110

```

```

{
    adress=2000;
    do {
        buf_out=read_ext_eeprom(address);
        printf("\r\n%3U",buf_out);
        adress++;
    }while (adress<2200);
}

```

```

void Read_Garp120

```

```

{

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

adres=2200;
do {
    buf_out=read_ext_eeprom(adres);
    printf("\r\n%3U",buf_out);
    adres++;
while (adres<2400);
}

```

```

void check_command() // check character transmitted by rs232

```

```

{char buf;
    buf=toupper(hgetc());
switch (buf)
{
    case '1' : Read_Garp10;
    break;
    case '2' : Read_Garp20;
    break;
    case '3' : Read_Garp30;
    break;
    case '4' : Read_Garp40;
    break;
    case '5' : Read_Garp50;
    break;
    case '6' : Read_Garp60;
    break;
    case '7' : Read_Garp70;
    break;
    case '8' : Read_Garp80;
    break;
    case '9' : Read_Garp90;
    break;
    case 'A' : Read_Garp100;
    break;
    case 'B' : Read_Garp110;
    break;
    case 'C' : Read_Garp120;
    break;
    case 'a' : Write_Garp10;
    break;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        case 'b' : Write_Garp20;
        break;

        case 'c' : Write_Garp30;
        break;

        case 'd' : Write_Garp40;
        break;

        case 'e' : Write_Garp50;
        break;

        case 'f' : Write_Garp60;
        break;

        case 'g' : Write_Garp70;
        break;

        case 'h' : Write_Garp80;
        break;

        case 'i' : Write_Garp90;
        break;

        case 'j' : Write_Garp100;
        break;

        case 'k' : Write_Garp110;
        break;

        case 'l' : Write_Garp120;
        break;
    }
}

void main(void)
{
    long int adress_j;
    byte buf_in,buf_out;
    enable_interrupts(global);
    enable_interrupts(int_rda);
    set_tris_B(0x00);
    setup_port_a(ALL_ANALOG);    // A0 A1 A2 A3 A5 E0 E1 E2 Ref=Vdd
    setup_adc(ADC_CLOCK_INTERNAL); // Clock RC
    // set_adc_channel(0);      // Read Analog Input RA0(channel=0)
    while(TRUE)
    {
        if(bkbhit)
            check_command();
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
else
    do {
        set_adc_channel(0);
        delay_us(10);
        value1 = Read_ADC0;
        PUTC(value1);
        delay_ms(1);
    } while (TRUE);
}
```

```
}
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**SOURCE CODE MODE2**

```

#include <16f877.h>
#include <fuses.h>,NOWDT,NOPROTECT,nolvp
#include <delay.h>(clock=20000000)
#include <fast_io.h>(B)
#include <24512.c>
#define use_portd_kbd
#include <kbd.c>

byte a,x,y,z,SUM;
long int adress;
byte buf_in,buf_out;
int i;
void main(void)
{
    long int adress,j,BB[20];
    int Output;
    char choice,check,shoot;
    char AA[5];
    kbd_init();
    set_tris_B(0x00);
    while(TRUE){
        Start:
        choice=kbd_getc();
        switch(choice)
        {
            case '#':{check=1;i=0;
                do{
                    loop2:
                    shoot=kbd_getc();
                    if (shoot=='0'){BB[i]=0;}
                    else if (shoot=='1'){BB[i]=1;}
                    else if (shoot=='2'){BB[i]=2;}
                    else if (shoot=='3'){BB[i]=3;}
                    else if (shoot=='4'){BB[i]=4;}
                    else if (shoot=='5'){BB[i]=5;}
                    else if (shoot=='6'){BB[i]=6;}
                    else if (shoot=='7'){BB[i]=7;}
                    else if (shoot=='8'){BB[i]=8;}
                    else if (shoot=='9'){BB[i]=9;}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

else goto loop2;
i++;
}while(i<1);
}
case**:{check=1;j=0;
do{
loop:
shoot=kbd_getc();
if (shoot==**){check=0;break;}
else if (shoot=='0'){AA[i]=shoot;}
else if (shoot=='1'){AA[i]=shoot;}
else if (shoot=='2'){AA[i]=shoot;}
else if (shoot=='3'){AA[i]=shoot;}
else if (shoot=='4'){AA[i]=shoot;}
else if (shoot=='5'){AA[i]=shoot;}
else if (shoot=='6'){AA[i]=shoot;}
else if (shoot=='7'){AA[i]=shoot;}
else if (shoot=='8'){AA[i]=shoot;}
else if (shoot=='9'){AA[i]=shoot;}
else goto loop;
i++;
}while(check);
if(i==1)
{
if (AA[0]=='1'){Output=1;}
else if (AA[0]=='2'){Output=2;}
else if (AA[0]=='3'){Output=3;}
else if (AA[0]=='4'){Output=4;}
else if (AA[0]=='5'){Output=5;}
else if (AA[0]=='6'){Output=6;}
else if (AA[0]=='7'){Output=7;}
else if (AA[0]=='8'){Output=8;}
else if (AA[0]=='9'){Output=9;}
}
if(i==2)
{
if (AA[0]=='1')
{
if(AA[1]=='0'){Output=10;}
if(AA[1]=='1'){Output=11;}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        if(AA[1]!='2'){Output=12;}
        else break;
    }
}

switch (Output)
{
    case (1):
        //Check1:
        xxx1:
        adress=0;
        do {

            buf_out=read_ext_ceprom(address);
            adress++;
            output_b(buf_out);
            delay_ms(4);
            if (input(PIN_D0)){goto Start;}
        }while (adress<4200);
        break;
        goto xxx1;
    case (2):
        //Check2:
        xxx2:
        adress=200;
        do {

            buf_out=read_ext_ceprom(address);
            adress++;
            output_b(buf_out);
            delay_ms(4);
            if (!input(PIN_D0)){goto Start;}
        }while (adress<4200);
        break;
        goto xxx2;
    case (3):
        //Check3:
        xxx3:
        adress=400;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

do {

    buf_out=read_ext_eeprom(address);
    address++;
    output_b(buf_out);
    delay_ms(4);
    if (!input(PIN_D0)){goto Start;}
}while (address<4200);
break;
goto xxx3;

case (4):
//Check4:
xxx4:
address=3600;
do {
    buf_out=read_ext_eeprom(address);
    address++;
    output_b(buf_out);
    delay_ms(4);
    if (!input(PIN_D0)){goto Start;}
}while (address<3800);
break;
goto xxx4;

case (5):
//Check5:
xxx5:
address=3800;
do {

    buf_out=read_ext_eeprom(address);
    address++;
    output_b(buf_out);
    delay_ms(4);
    if (!input(PIN_D0)){goto Start;}
}while (address<4000);
break;
goto xxx5;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

case (6):
    //Check6:
    xxx6:
    adress=4000;
    do {

        buf_out=read_ext_eeprom(adress);
        adress++;
        output_b(buf_out);
        delay_ms(4);
        if (!input(PIN_D0)){goto Start;}
    }while (adress<4200);
    break;
    goto xxx6;

case (7):
    //Check7:
    xxx7:
    adress=1200;
    do {

        buf_out=read_ext_eeprom(adress);
        adress++;
        output_b(buf_out);
        delay_ms(4);
        if (!input(PIN_D0)){goto Start;}
    }while (adress<1400);
    break;
    goto xxx7;

case (8):
    //Check8:
    xxx8:
    adress=1400;
    do {

        buf_out=read_ext_eeprom(adress);
        adress++;
        output_b(buf_out);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        delay_ms(4);
        if (!input(PIN_D0)){goto Start;}
    }while (adress<1600);
    break;
    goto xxx8;

case (9):
    //Check9:
    xxx9:
    adress=4600;
    do {

        buf_out=read_ext_cprom(address);
        adress++;
        output_b(buf_out);
        delay_ms(4);
        if (!input(PIN_D0)){goto Start;}
    }while (adress<4800);
    break;
    goto xxx9;

case (10):
    //Check10:
    xxx10:
    adress=1800;
    do {

        buf_out=read_ext_eprom(address);
        adress++;
        output_b(buf_out);
        delay_ms(4);
        if (!input(PIN_D0)){goto Start;}
    }while (adress<2000);
    break;
    goto xxx10;

case (11):
    //Check11:
    xxx11:
    adress=2000;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

do {

    buf_out=read_ext_eeeprom(address);
    adress++;
    output_b(buf_out);
    delay_ms(4);
    if (!input(PIN_D0)){goto Start;}
}while (adress<2200);
break;
goto xxx11;

case (12):
//Check12:
xxx12:
adress=2200;
do {
    buf_out=read_ext_eeeprom(address);
    adress++;
    output_b(buf_out);
    delay_ms(4);
    if (!input(PIN_D0)){goto Start;}
}while (adress<2400);
break;
goto xxx12;

case (13):
//Graph1
adress=0;
do {

    buf_out=read_ext_eeeprom(address);
    adress++;
    output_b(buf_out);
    delay_ms(4);
    if (!input(PIN_D0)){goto Start;}
}while (adress<200);
adress=0;
do {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        buf_out=read_ext_eeprom(address);
        adress++;
        output_b(buf_out);
        delay_ms(4);
        if (!input(PIN_D0)){goto Start;}
    }while (adress<200);
//Graph2
adress=200;
do {

        buf_out=read_ext_eeprom(address);
        adress++;
        output_b(buf_out);
        delay_ms(4);
        if (!input(PIN_D0)){goto Start;}
    }while (adress<400);
adress=200;
do {

        buf_out=read_ext_eeprom(address);
        adress++;
        output_b(buf_out);
        delay_ms(4);
        if (!input(PIN_D0)){goto Start;}
    }while (adress<400);
//Graph3
adress=400;
do {

        buf_out=read_ext_eeprom(address);
        adress++;
        output_b(buf_out);
        delay_ms(4);
        if (!input(PIN_D0)){goto Start;}
    }while (adress<600);
adress=400;
do {

        buf_out=read_ext_eeprom(address);
        adress++;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        output_b(buf_out);
        delay_ms(4);
        if (!input(PIN_D0)){goto Start;}
    }while (address<600);
//Graph4
    adress=3600;
    do {

        buf_out=read_ext_ceprom(address);
        adress++;
        output_b(buf_out);
        delay_ms(4);
        if (!input(PIN_D0)){goto Start;}
    }while (address<3800);
    adress=3600;
    do {

        buf_out=read_ext_ceprom(address);
        adress++;
        output_b(buf_out);
        delay_ms(4);
        if (!input(PIN_D0)){goto Start;}
    }while (address<3800);
//Graph5
    adress=3800;
    do {

        buf_out=read_ext_ceprom(address);
        adress++;
        output_b(buf_out);
        delay_ms(4);
        if (!input(PIN_D0)){goto Start;}
    }while (address<4000);
    adress=3800;
    do {

        buf_out=read_ext_ceprom(address);
        adress++;
        output_b(huf_out);
        delay_ms(4);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        if (!input(PIN_D0)){goto Start;}
    }while (adress<4000);
//Graph6
adress=4000;
do {

        buf_out=read_ext_eeprom(adress);
        adress++;
        output_b(buf_out);
        delay_ms(4);
        if (!input(PIN_D0)){goto Start;}
    }while (adress<4200);
adress=4000;
do {

        buf_out=read_ext_eeprom(adress);
        adress++;
        output_b(buf_out);
        delay_ms(4);
        if (!input(PIN_D0)){goto Start;}
    }while (adress<4200);
//Graph7
adress=1200;
do {

        buf_out=read_ext_eeprom(adress);
        adress++;
        output_b(buf_out);
        delay_ms(4);
        if (!input(PIN_D0)){goto Start;}
    }while (adress<1400);
adress=1200;
do {

        buf_out=read_ext_eeprom(adress);
        adress++;
        output_b(buf_out);
        delay_ms(4);
        if (!input(PIN_D0)){goto Start;}
    }while (adress<1400);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

//Graph8
adress=1400;
do {

    buf_out=read_ext_eeprom(adress);
    adress++;
    output_b(buf_out);
    delay_ms(4);
    if (!input(PIN_D0)){goto Start;}
}while (adress<1600);
adress=1400;
do {

    buf_out=read_ext_eeeprom(adress);
    adress++;
    output_b(buf_out);
    delay_ms(4);
    if (!input(PIN_D0)){goto Start;}
}while (adress<1600);
//Graph9
adress=4600;
do {

    buf_out=read_ext_eeeprom(adress);
    adress++;
    output_b(buf_out);
    delay_ms(4);
    if (!input(PIN_D0)){goto Start;}
}while (adress<4800);
adress=4600;
do {

    buf_out=read_ext_eeeprom(adress);
    adress++;
    output_b(buf_out);
    delay_ms(4);
    if (!input(PIN_D0)){goto Start;}
}while (adress<4800);
//Graph10
adress=1800;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

do {

    buf_out=read_ext_eeeprom(address);
    address++;
    output_b(buf_out);
    delay_ms(4);
    if (!input(PIN_D0)){goto Start;}
}while (address<2000);
address=1800;
do {

    buf_out=read_ext_eeeprom(address);
    address++;
    output_b(buf_out);
    delay_ms(4);
    if (!input(PIN_D0)){goto Start;}
}while (address<2000);
//Graph11
address=2000;
do {

    buf_out=read_ext_eeeprom(address);
    address++;
    output_b(buf_out);
    delay_ms(4);
    if (!input(PIN_D0)){goto Start;}
}while (address<2200);
address=2000;
do {

    buf_out=read_ext_eeeprom(address);
    address++;
    output_b(buf_out);
    delay_ms(4);
    if (!input(PIN_D0)){goto Start;}
}while (address<2200);
//Graph12
address=2200;
do {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    buf_out=read_ext_eeprom(address);
    address++;
    output_b(buf_out);
    delay_ms(4);
    if (!input(PIN_D0)){goto Start;}
}while (address<2400);
address=2200;
do {

```

```

    buf_out=read_ext_eeprom(address);
    address++;
    output_b(buf_out);
    delay_ms(4);
    if (!input(PIN_D0)){goto Start;}
}while (address<2400);
break;
}

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

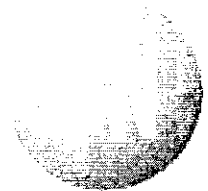
# MC78L00A Series

## Three-Terminal Low Current Positive Voltage Regulators

The MC78L00A Series of positive voltage regulators are inexpensive, easy-to-use devices suitable for a multitude of applications that require a regulated supply of up to 100 mA. Like their higher powered MC7800 and MC78M00 Series cousins, these regulators feature internal current limiting and thermal shutdown making them remarkably rugged. No external components are required with the MC78L00 devices in many applications.

These devices offer a substantial performance advantage over the traditional zener diode-resistor combination, as output impedance and quiescent current are substantially reduced.

- Wide Range of Available, Fixed Output Voltages
- Low Cost
- Internal Short Circuit Current Limiting
- Internal Thermal Overload Protection
- No External Components Required
- Complementary Negative Regulators Offered (MC79L00A Series)



**ON Semiconductor**  
 A Division of Motorola  
<http://onsemi.com>



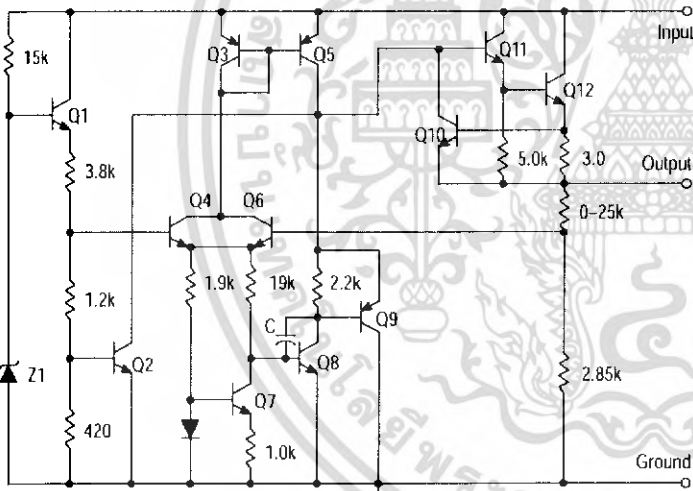
Pin: 1. Output  
 2. Ground  
 3. Input

**TO-92  
 P SUFFIX  
 CASE 29**



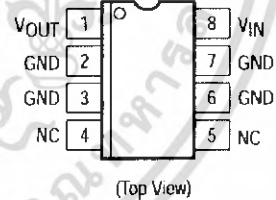
**SOP-8\*  
 D SUFFIX  
 CASE 751**

### Representative Schematic Diagram

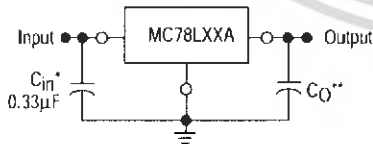


\*SOP-8 is an internally modified SO-8 package. Pins 2, 3, 6, and 7 are electrically common to the die attach flag. This internal lead frame modification decreases package thermal resistance and increases power dissipation capability when appropriately mounted on a printed circuit board. SOP-8 conforms to all external dimensions of the standard SO-8 package.

### PIN CONNECTIONS



### Standard Application



A common ground is required between the input and the output voltages. The input voltage must remain typically 2.0 V above the output voltage even during the low point on the input ripple voltage.

\* $C_{in}$  is required if regulator is located an appreciable distance from power supply filter.  
 \*\* $C_O$  is not needed for stability; however, it does improve transient response.

### DEVICE TYPE/NOMINAL VOLTAGE

5% Output Voltage Accuracy	Voltage
MC78L05AC	5.0
MC78L08AC	8.0
MC78L09AC	9.0
MC78L12AC	12
MC78L15AC	15
MC78L18AC	18
MC78L24AC	24

### ORDERING INFORMATION

See detailed ordering and shipping information in the package dimensions section on page 7 of this data sheet.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## MC78L00A Series

**MAXIMUM RATINGS** ( $T_A = +125^\circ\text{C}$ , unless otherwise noted.)

Rating	Symbol	Value	Unit
Input Voltage (2.6 V–8.0 V) (12 V–18 V) (24 V)	$V_I$	30 35 40	Vdc
Storage Temperature Range	$T_{stg}$	-65 to +150	$^\circ\text{C}$
Operating Junction Temperature Range	$T_J$	0 to +150	$^\circ\text{C}$

**ELECTRICAL CHARACTERISTICS** ( $V_I = 10\text{ V}$ ,  $I_O = 40\text{ mA}$ ,  $C_I = 0.33\ \mu\text{F}$ ,  $C_O = 0.1\ \mu\text{F}$ ,  $-40^\circ\text{C} < T_J < +125^\circ\text{C}$  (for MC78LXXAB),  $0^\circ\text{C} < T_J < +125^\circ\text{C}$  (for MC78LXXAC), unless otherwise noted.)

Characteristics	Symbol	MC78L05AC, AB			Unit
		Min	Typ	Max	
Output Voltage ( $T_J = +25^\circ\text{C}$ )	$V_O$	4.8	5.0	5.2	Vdc
Line Regulation ( $T_J = +25^\circ\text{C}$ , $I_O = 40\text{ mA}$ ) $7.0\text{ Vdc} \leq V_I \leq 20\text{ Vdc}$ $8.0\text{ Vdc} \leq V_I \leq 20\text{ Vdc}$	Reg <sub>line</sub>	–	55 45	150 100	mV
Load Regulation ( $T_J = +25^\circ\text{C}$ , $1.0\text{ mA} \leq I_O \leq 100\text{ mA}$ ) ( $T_J = +25^\circ\text{C}$ , $1.0\text{ mA} \leq I_O \leq 40\text{ mA}$ )	Reg <sub>load</sub>	–	11 5.0	60 30	mV
Output Voltage ( $7.0\text{ Vdc} \leq V_I \leq 20\text{ Vdc}$ , $1.0\text{ mA} \leq I_O \leq 40\text{ mA}$ ) ( $V_I = 10\text{ V}$ , $1.0\text{ mA} \leq I_O \leq 70\text{ mA}$ )	$V_O$	4.75 4.75	–	5.25 5.25	Vdc
Input Bias Current ( $T_J = +25^\circ\text{C}$ ) ( $T_J = +125^\circ\text{C}$ )	$I_{IB}$	–	3.8	6.0 5.5	mA
Input Bias Current Change ( $8.0\text{ Vdc} \leq V_I \leq 20\text{ Vdc}$ ) ( $1.0\text{ mA} < I_O < 40\text{ mA}$ )	$\Delta I_{IB}$	–	–	1.5 0.1	mA
Output Noise Voltage ( $T_A = +25^\circ\text{C}$ , $10\text{ Hz} \leq f \leq 100\text{ kHz}$ )	$V_n$	–	40	–	$\mu\text{V}$
Ripple Rejection ( $I_O = 40\text{ mA}$ , $f = 120\text{ Hz}$ , $8.0\text{ Vdc} \leq V_I \leq 18\text{ V}$ , $T_J = +25^\circ\text{C}$ )	RR	41	49	–	dB
Dropout Voltage ( $T_J = +25^\circ\text{C}$ )	$V_I - V_O$	–	1.7	–	Vdc

<http://onsemi.com>

2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## MC78L00A Series

**ELECTRICAL CHARACTERISTICS** ( $V_I = 14\text{ V}$ ,  $I_O = 40\text{ mA}$ ,  $C_I = 0.33\ \mu\text{F}$ ,  $C_O = 0.1\ \mu\text{F}$ ,  $-40^\circ\text{C} < T_J < +125^\circ\text{C}$  (for MC78LXXAB),  $0^\circ\text{C} < T_J < +125^\circ\text{C}$  (for MC78LXXAC), unless otherwise noted.)

Characteristics	Symbol	MC78L08AC, AB			Unit
		Min	Typ	Max	
Output Voltage ( $T_J = +25^\circ\text{C}$ )	$V_O$	7.7	8.0	8.3	Vdc
Line Regulation ( $T_J = +25^\circ\text{C}$ , $I_O = 40\text{ mA}$ ) $10.5\text{ Vdc} \leq V_I \leq 23\text{ Vdc}$ $11\text{ Vdc} \leq V_I \leq 23\text{ Vdc}$	Reg <sub>line</sub>	–	20 12	175 125	mV
Load Regulation ( $T_J = +25^\circ\text{C}$ , $1.0\text{ mA} < I_O < 100\text{ mA}$ ) ( $T_J = +25^\circ\text{C}$ , $1.0\text{ mA} \leq I_O \leq 40\text{ mA}$ )	Reg <sub>load</sub>	–	15 8.0	80 40	mV
Output Voltage ( $10.5\text{ Vdc} \leq V_I \leq 23\text{ Vdc}$ , $1.0\text{ mA} \leq I_O \leq 40\text{ mA}$ ) ( $V_I = 14\text{ V}$ , $1.0\text{ mA} \leq I_O \leq 70\text{ mA}$ )	$V_O$	7.6 7.6	– –	8.4 8.4	Vdc
Input Bias Current ( $T_J = +25^\circ\text{C}$ ) ( $T_J = +125^\circ\text{C}$ )	$I_{IB}$	– –	3.0 –	6.0 5.5	mA
Input Bias Current Change ( $11\text{ Vdc} \leq V_I \leq 23\text{ Vdc}$ ) ( $1.0\text{ mA} \leq I_O \leq 40\text{ mA}$ )	$\Delta I_{IB}$	– –	– –	1.5 0.1	mA
Output Noise Voltage ( $T_A = +25^\circ\text{C}$ , $10\text{ Hz} \leq f \leq 100\text{ kHz}$ )	$V_n$	–	60	–	$\mu\text{V}$
Ripple Rejection ( $I_O = 40\text{ mA}$ , $f = 120\text{ Hz}$ , $12\text{ V} \leq V_I \leq 23\text{ V}$ , $T_J = +25^\circ\text{C}$ )	RR	37	57	–	dB
Dropout Voltage ( $T_J = +25^\circ\text{C}$ )	$V_I - V_O$	–	1.7	–	Vdc

**ELECTRICAL CHARACTERISTICS** ( $V_I = 15\text{ V}$ ,  $I_O = 40\text{ mA}$ ,  $C_I = 0.33\ \mu\text{F}$ ,  $C_O = 0.1\ \mu\text{F}$ ,  $-40^\circ\text{C} < T_J < +125^\circ\text{C}$  (for MC78LXXAB),  $0^\circ\text{C} < T_J < +125^\circ\text{C}$  (for MC78LXXAC), unless otherwise noted.)

Characteristics	Symbol	MC78L09AC, AB			Unit
		Min	Typ	Max	
Output Voltage ( $T_J = +25^\circ\text{C}$ )	$V_O$	8.6	9.0	9.4	Vdc
Line Regulation ( $T_J = +25^\circ\text{C}$ , $I_O = 40\text{ mA}$ ) $11.5\text{ Vdc} \leq V_I \leq 24\text{ Vdc}$ $12\text{ Vdc} \leq V_I \leq 24\text{ Vdc}$	Reg <sub>line</sub>	–	20 12	175 125	mV
Load Regulation ( $T_J = +25^\circ\text{C}$ , $1.0\text{ mA} < I_O < 100\text{ mA}$ ) ( $T_J = +25^\circ\text{C}$ , $1.0\text{ mA} < I_O < 40\text{ mA}$ )	Reg <sub>load</sub>	–	15 8.0	90 40	mV
Output Voltage ( $11.5\text{ Vdc} \leq V_I \leq 24\text{ Vdc}$ , $1.0\text{ mA} \leq I_O \leq 40\text{ mA}$ ) ( $V_I = 15\text{ V}$ , $1.0\text{ mA} \leq I_O \leq 70\text{ mA}$ )	$V_O$	8.5 8.5	– –	9.5 9.5	Vdc
Input Bias Current ( $T_J = +25^\circ\text{C}$ ) ( $T_J = +125^\circ\text{C}$ )	$I_{IB}$	– –	3.0 –	6.0 5.5	mA
Input Bias Current Change ( $11\text{ Vdc} \leq V_I \leq 23\text{ Vdc}$ ) ( $1.0\text{ mA} \leq I_O \leq 40\text{ mA}$ )	$\Delta I_{IB}$	– –	– –	1.5 0.1	mA
Output Noise Voltage ( $T_A = +25^\circ\text{C}$ , $10\text{ Hz} \leq f \leq 100\text{ kHz}$ )	$V_n$	–	60	–	$\mu\text{V}$
Ripple Rejection ( $I_O = 40\text{ mA}$ , $f = 120\text{ Hz}$ , $13\text{ V} \leq V_I \leq 24\text{ V}$ , $T_J = +25^\circ\text{C}$ )	RR	37	57	–	dB
Dropout Voltage ( $T_J = +25^\circ\text{C}$ )	$V_I - V_O$	–	1.7	–	Vdc

<http://onsemi.com>

3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## MC78L00A Series

**ELECTRICAL CHARACTERISTICS** ( $V_I = 19\text{ V}$ ,  $I_O = 40\text{ mA}$ ,  $C_I = 0.33\text{ }\mu\text{F}$ ,  $C_O = 0.1\text{ }\mu\text{F}$ ,  $-40^\circ\text{C} < T_J < +125^\circ\text{C}$  (for MC78LXXAB),  $0^\circ\text{C} < T_J < +125^\circ\text{C}$  (for MC78LXXAC), unless otherwise noted.)

Characteristics	Symbol	MC78L12AC, AB			Unit
		Min	Typ	Max	
Output Voltage ( $T_J = +25^\circ\text{C}$ )	$V_O$	11.5	12	12.5	Vdc
Line Regulation ( $T_J = +25^\circ\text{C}$ , $I_O = 40\text{ mA}$ ) $14.5\text{ Vdc} \leq V_I \leq 27\text{ Vdc}$ $16\text{ Vdc} \leq V_I \leq 27\text{ Vdc}$	Reg <sub>line</sub>	–	120 100	250 200	mV
Load Regulation ( $T_J = +25^\circ\text{C}$ , $1.0\text{ mA} \leq I_O \leq 100\text{ mA}$ ) ( $T_J = +25^\circ\text{C}$ , $1.0\text{ mA} \leq I_O \leq 40\text{ mA}$ )	Reg <sub>load</sub>	–	20 10	100 50	mV
Output Voltage ( $14.5\text{ Vdc} \leq V_I \leq 27\text{ Vdc}$ , $1.0\text{ mA} \leq I_O \leq 40\text{ mA}$ ) ( $V_I = 19\text{ V}$ , $1.0\text{ mA} \leq I_O \leq 70\text{ mA}$ )	$V_O$	11.4 11.4	–	12.6 12.6	Vdc
Input Bias Current ( $T_J = +25^\circ\text{C}$ ) ( $T_J = +125^\circ\text{C}$ )	$I_{IB}$	–	4.2	6.5 6.0	mA
Input Bias Current Change ( $16\text{ Vdc} < V_I \leq 27\text{ Vdc}$ ) ( $1.0\text{ mA} \leq I_O \leq 40\text{ mA}$ )	$\Delta I_{IB}$	–	–	1.5 0.1	mA
Output Noise Voltage ( $T_A = +25^\circ\text{C}$ , $10\text{ Hz} < f < 100\text{ kHz}$ )	$V_n$	–	80	–	$\mu\text{V}$
Ripple Rejection ( $I_O = 40\text{ mA}$ , $f = 120\text{ Hz}$ , $15\text{ V} \leq V_I \leq 25\text{ V}$ , $T_J = +25^\circ\text{C}$ )	RR	37	42	–	dB
Dropout Voltage ( $T_J = +25^\circ\text{C}$ )	$V_I - V_O$	–	1.7	–	Vdc

**ELECTRICAL CHARACTERISTICS** ( $V_I = 23\text{ V}$ ,  $I_O = 40\text{ mA}$ ,  $C_I = 0.33\text{ }\mu\text{F}$ ,  $C_O = 0.1\text{ }\mu\text{F}$ ,  $-40^\circ\text{C} < T_J < +125^\circ\text{C}$  (for MC78LXXAB),  $0^\circ\text{C} < T_J < +125^\circ\text{C}$  (for MC78LXXAC), unless otherwise noted.)

Characteristics	Symbol	MC78L15AC, AB			Unit
		Min	Typ	Max	
Output Voltage ( $T_J = +25^\circ\text{C}$ )	$V_O$	14.4	15	15.6	Vdc
Line Regulation ( $T_J = +25^\circ\text{C}$ , $I_O = 40\text{ mA}$ ) $17.5\text{ Vdc} \leq V_I \leq 30\text{ Vdc}$ $20\text{ Vdc} \leq V_I \leq 30\text{ Vdc}$	Reg <sub>line</sub>	–	130 110	300 250	mV
Load Regulation ( $T_J = +25^\circ\text{C}$ , $1.0\text{ mA} \leq I_O \leq 100\text{ mA}$ ) ( $T_J = +25^\circ\text{C}$ , $1.0\text{ mA} \leq I_O \leq 40\text{ mA}$ )	Reg <sub>load</sub>	–	25 12	150 75	mV
Output Voltage ( $17.5\text{ Vdc} \leq V_I \leq 30\text{ Vdc}$ , $1.0\text{ mA} \leq I_O \leq 40\text{ mA}$ ) ( $V_I = 23\text{ V}$ , $1.0\text{ mA} \leq I_O \leq 70\text{ mA}$ )	$V_O$	14.25 14.25	–	15.75 15.75	Vdc
Input Bias Current ( $T_J = +25^\circ\text{C}$ ) ( $T_J = +125^\circ\text{C}$ )	$I_{IB}$	–	4.4	6.5 6.0	mA
Input Bias Current Change ( $20\text{ Vdc} \leq V_I \leq 30\text{ Vdc}$ ) ( $1.0\text{ mA} \leq I_O \leq 40\text{ mA}$ )	$\Delta I_{IB}$	–	–	1.5 0.1	mA
Output Noise Voltage ( $T_A = +25^\circ\text{C}$ , $10\text{ Hz} \leq f \leq 100\text{ kHz}$ )	$V_n$	–	90	–	$\mu\text{V}$
Ripple Rejection ( $I_O = 40\text{ mA}$ , $f = 120\text{ Hz}$ , $18.5\text{ V} \leq V_I \leq 28.5\text{ V}$ , $T_J = +25^\circ\text{C}$ )	RR	34	39	–	dB
Dropout Voltage ( $T_J = +25^\circ\text{C}$ )	$V_I - V_O$	–	1.7	–	Vdc

<http://onsemi.com>

4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## MC78L00A Series

**ELECTRICAL CHARACTERISTICS** ( $V_I = 27\text{ V}$ ,  $I_O = 40\text{ mA}$ ,  $C_I = 0.33\text{ }\mu\text{F}$ ,  $C_O = 0.1\text{ }\mu\text{F}$ ,  $0^\circ\text{C} < T_J < +125^\circ\text{C}$ , unless otherwise noted.)

Characteristics	Symbol	MC78L18AC			Unit
		Min	Typ	Max	
Output Voltage ( $T_J = +25^\circ\text{C}$ )	$V_O$	17.3	18	18.7	Vdc
Line Regulation ( $T_J = +25^\circ\text{C}$ , $I_O = 40\text{ mA}$ ) $21.4\text{ Vdc} \leq V_I \leq 33\text{ Vdc}$ $20.7\text{ Vdc} \leq V_I \leq 33\text{ Vdc}$ $22\text{ Vdc} \leq V_I \leq 33\text{ Vdc}$ $21\text{ Vdc} \leq V_I \leq 33\text{ Vdc}$	Regline	–	45	325	mV
Load Regulation ( $T_J = +25^\circ\text{C}$ , $1.0\text{ mA} \leq I_O \leq 100\text{ mA}$ ) ( $T_J = +25^\circ\text{C}$ , $1.0\text{ mA} \leq I_O \leq 40\text{ mA}$ )	Regload	–	30	170	mV
Output Voltage ( $21.4\text{ Vdc} \leq V_I \leq 33\text{ Vdc}$ , $1.0\text{ mA} \leq I_O < 40\text{ mA}$ ) ( $20.7\text{ Vdc} \leq V_I \leq 33\text{ Vdc}$ , $1.0\text{ mA} \leq I_O \leq 40\text{ mA}$ ) ( $V_I = 27\text{ V}$ , $1.0\text{ mA} \leq I_O \leq 70\text{ mA}$ ) ( $V_I = 27\text{ V}$ , $1.0\text{ mA} \leq I_O \leq 70\text{ mA}$ )	$V_O$	17.1	–	18.9	Vdc
Input Bias Current ( $T_J = +25^\circ\text{C}$ ) ( $T_J = +125^\circ\text{C}$ )	$I_{IB}$	–	3.1	6.5	mA
Input Bias Current Change ( $22\text{ Vdc} \leq V_I \leq 33\text{ Vdc}$ ) ( $21\text{ Vdc} \leq V_I \leq 33\text{ Vdc}$ ) ( $1.0\text{ mA} \leq I_O \leq 40\text{ mA}$ )	$\Delta I_{IB}$	–	–	1.5	mA
Output Noise Voltage ( $T_A = +25^\circ\text{C}$ , $10\text{ Hz} \leq f \leq 100\text{ kHz}$ )	$V_n$	–	150	–	$\mu\text{V}$
Ripple Rejection ( $I_O = 40\text{ mA}$ , $f = 120\text{ Hz}$ , $23\text{ V} \leq V_I \leq 33\text{ V}$ , $T_J = +25^\circ\text{C}$ )	RR	33	48	–	dB
Dropout Voltage ( $T_J = +25^\circ\text{C}$ )	$V_I - V_O$	–	1.7	–	Vdc

**ELECTRICAL CHARACTERISTICS** ( $V_I = 33\text{ V}$ ,  $I_O = 40\text{ mA}$ ,  $C_I = 0.33\text{ }\mu\text{F}$ ,  $C_O = 0.1\text{ }\mu\text{F}$ ,  $0^\circ\text{C} < T_J < +125^\circ\text{C}$ , unless otherwise noted.)

Characteristics	Symbol	MC78L24AC			Unit
		Min	Typ	Max	
Output Voltage ( $T_J = +25^\circ\text{C}$ )	$V_O$	23	24	25	Vdc
Line Regulation ( $T_J = +25^\circ\text{C}$ , $I_O = 40\text{ mA}$ ) $27.5\text{ Vdc} \leq V_I \leq 38\text{ Vdc}$ $28\text{ Vdc} \leq V_I \leq 80\text{ Vdc}$ $27\text{ Vdc} \leq V_I \leq 38\text{ Vdc}$	Regline	–	–	–	mV
Load Regulation ( $T_J = +25^\circ\text{C}$ , $1.0\text{ mA} \leq I_O \leq 100\text{ mA}$ ) ( $T_J = +25^\circ\text{C}$ , $1.0\text{ mA} \leq I_O \leq 40\text{ mA}$ )	Regload	–	40	200	mV
Output Voltage ( $28\text{ Vdc} \leq V_I \leq 38\text{ Vdc}$ , $1.0\text{ mA} \leq I_O \leq 40\text{ mA}$ ) ( $27\text{ Vdc} \leq V_I \leq 38\text{ Vdc}$ , $1.0\text{ mA} \leq I_O \leq 40\text{ mA}$ ) ( $28\text{ Vdc} \leq V_I \leq 33\text{ Vdc}$ , $1.0\text{ mA} \leq I_O \leq 70\text{ mA}$ ) ( $27\text{ Vdc} \leq V_I \leq 33\text{ Vdc}$ , $1.0\text{ mA} \leq I_O \leq 70\text{ mA}$ )	$V_O$	22.8	–	25.2	Vdc
Input Bias Current ( $T_J = +25^\circ\text{C}$ ) ( $T_J = +125^\circ\text{C}$ )	$I_{IB}$	–	3.1	6.5	mA
Input Bias Current Change ( $28\text{ Vdc} \leq V_I \leq 38\text{ Vdc}$ ) ( $1.0\text{ mA} \leq I_O \leq 40\text{ mA}$ )	$\Delta I_{IB}$	–	–	1.5	mA
Output Noise Voltage ( $T_A = +25^\circ\text{C}$ , $10\text{ Hz} \leq f \leq 100\text{ kHz}$ )	$V_n$	–	200	–	$\mu\text{V}$
Ripple Rejection ( $I_O = 40\text{ mA}$ , $f = 120\text{ Hz}$ , $29\text{ V} \leq V_I \leq 35\text{ V}$ , $T_J = +25^\circ\text{C}$ )	RR	31	45	–	dB
Dropout Voltage ( $T_J = +25^\circ\text{C}$ )	$V_I - V_O$	–	1.7	–	Vdc

<http://onsemi.com>

5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# MC78L00A Series

Figure 1. Dropout Characteristics

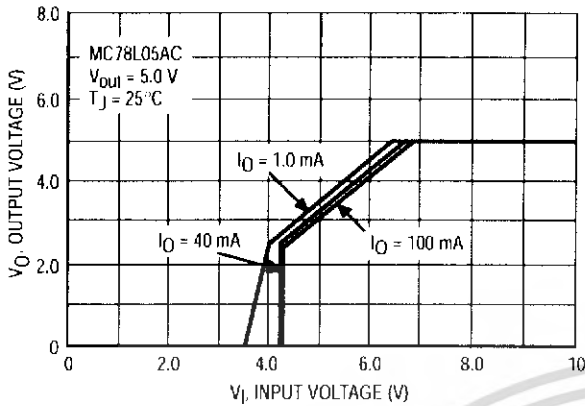


Figure 2. Dropout Voltage versus Junction Temperature

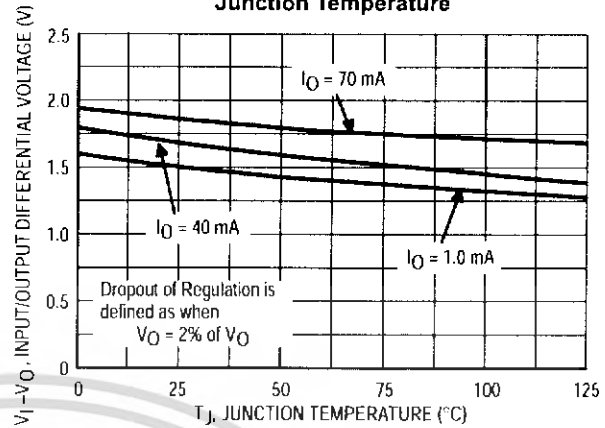


Figure 3. Input Bias Current versus Ambient Temperature

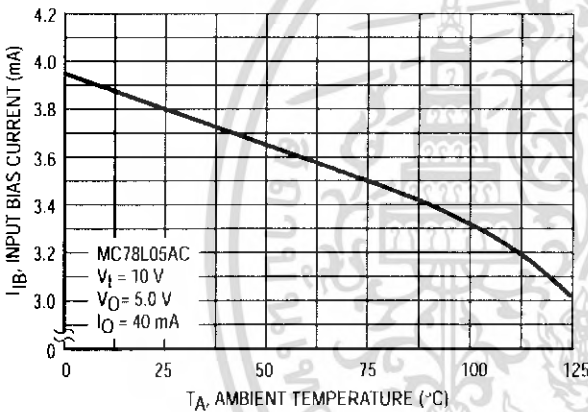


Figure 4. Input Bias Current versus Input Voltage

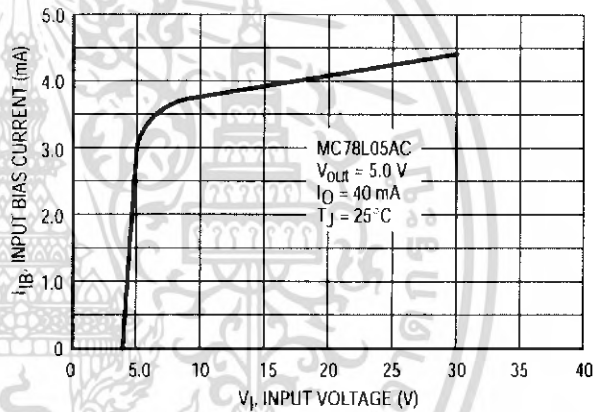


Figure 5. Maximum Average Power Dissipation versus Ambient Temperature – TO-92 Type Package

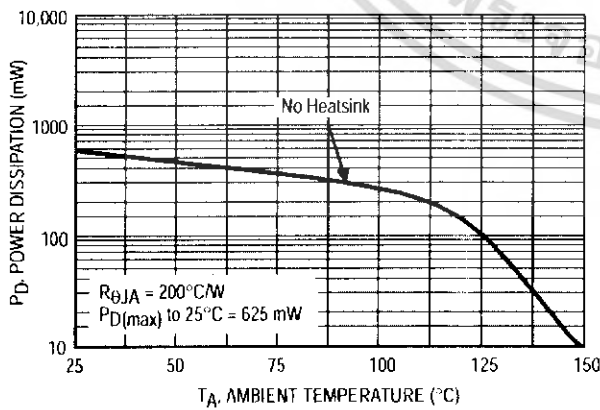
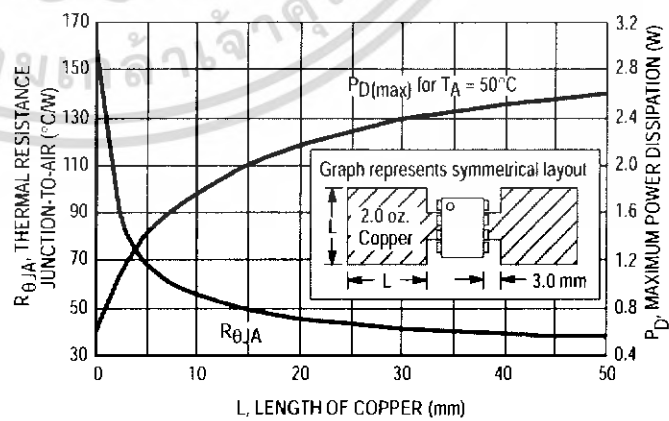


Figure 6. SOP-8 Thermal Resistance and Maximum Power Dissipation versus P.C.B. Copper Length



<http://onsemi.com>

# MC78L00A Series

## APPLICATIONS INFORMATION

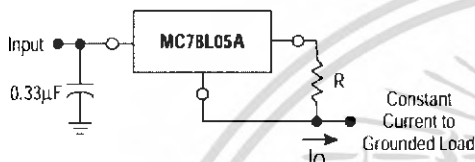
### Design Considerations

The MC78L00A Series of fixed voltage regulators are designed with Thermal Overload Protection that shuts down the circuit when subjected to an excessive power overload condition. Internal Short Circuit Protection limits the maximum current the circuit will pass.

In many low current applications, compensation capacitors are not required. However, it is recommended that the regulator input be bypassed with a capacitor if the regulator is connected to the power supply filter with long wire lengths, or if the output load capacitance is large. The

input bypass capacitor should be selected to provide good high-frequency characteristics to insure stable operation under all load conditions. A 0.33 μF or larger tantalum, mylar, or other capacitor having low internal impedance at high frequencies should be chosen. The bypass capacitor should be mounted with the shortest possible leads directly across the regulators input terminals. Good construction techniques should be used to minimize ground loops and lead resistance drops since the regulator has no external sense lead. Bypassing the output is also recommended.

Figure 7. Current Regulator



The MC78L00 regulators can also be used as a current source when connected as above. In order to minimize dissipation the MC78L05C is chosen in this application. Resistor R determines the current as follows:

$$I_O = \frac{5.0 \text{ V}}{R} + I_B$$

$I_B = 3.8 \text{ mA}$  over line and load changes

For example, a 100 mA current source would require R to be a 50 Ω, 1/2 W resistor and the output voltage compliance would be the input voltage less 7 V.

Figure 8. ± 15 V Tracking Voltage Regulator

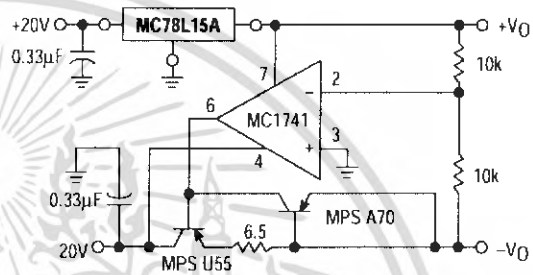
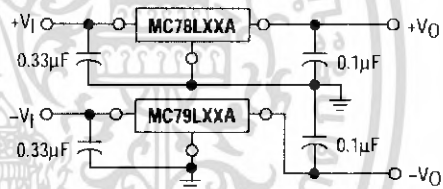


Figure 9. Positive and Negative Regulator



### ORDERING INFORMATION

Device	Operating Temperature Range	Package	Shipping
MC78LXXACD*	$T_J = 0^\circ \text{ to } +125^\circ \text{C}$	SOP-8	98 Units / Rail
MC78LXXACDR2*		SOP-8 / Tape & Reel	2500 Units / Tape & Reel
MC78LXXACP		TO-92	2000 Units / Bag
MC78LXXACPRA / MC78LXXACPRE		TO-92 / Tape & Reel	2000 Units / Tape & Reel
MC78LXXACPRM / MC78LXXACPRP		TO-92 / Ammo Pack	2000 Units / Ammo Pack
MC78LXXABD*	$T_J = -40^\circ \text{ to } +125^\circ \text{C}$	SOP-8	98 Units / Rail
MC78LXXABDR2*		SOP-8 / Tape & Reel	2500 Units / Tape & Reel
MC78LXXABP*		TO-92	2000 Units / Bag
MC78LXXABPRA / MC78LXXABPRE		TO-92 / Tape & Reel	2000 Units / Tape & Reel
MC78LXXABPRM / MC78LXXABPRP		TO-92 / Ammo Pack	2000 Units / Ammo Pack

XX indicates nominal voltage

\*Available in 5, 8, 9, 12 and 15 V devices.

<http://onsemi.com>

# DAC0808

## 8-Bit D/A Converter

### General Description

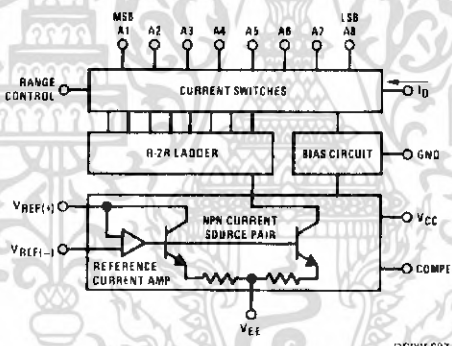
The DAC0808 is an 8-bit monolithic digital-to-analog converter (DAC) featuring a full scale output current settling time of 150 ns while dissipating only 33 mW with  $\pm 5V$  supplies. No reference current ( $I_{REF}$ ) trimming is required for most applications since the full scale output current is typically  $\pm 1$  LSB of  $255 I_{REF}/256$ . Relative accuracies of better than  $\pm 0.19\%$  assure 8-bit monotonicity and linearity while zero level output current of less than  $4 \mu A$  provides 8-bit zero accuracy for  $I_{REF} \geq 2$  mA. The power supply currents of the DAC0808 is independent of bit codes, and exhibits essentially constant device characteristics over the entire supply voltage range.

The DAC0808 will interface directly with popular TTL, DTL or CMOS logic levels, and is a direct replacement for the MC1508/MC1408. For higher speed applications, see DAC0800 data sheet.

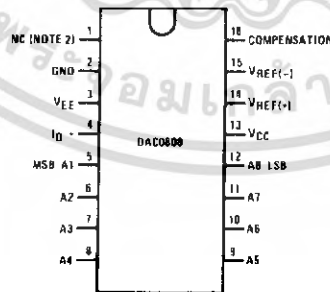
### Features

- Relative accuracy:  $\pm 0.19\%$  error maximum
- Full scale current match:  $\pm 1$  LSB typ
- Fast settling time: 150 ns typ
- Noninverting digital inputs are TTL and CMOS compatible
- High speed multiplying input slew rate: 8 mA/ $\mu s$
- Power supply voltage range:  $\pm 4.5V$  to  $\pm 18V$
- Low power consumption: 33 mW @  $\pm 5V$

### Block and Connection Diagrams



Dual-In-Line Package

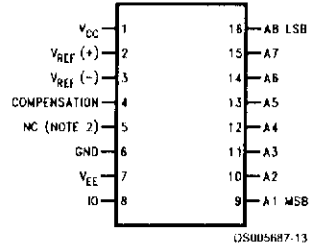


**Top View**  
**Order Number DAC0808**  
**See NS Package M16A or N16A**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Block and Connection Diagrams (Continued)

### Small-Outline Package



## Ordering Information

ACCURACY	OPERATING TEMPERATURE RANGE	N PACKAGE (N16A) (Note 1)		SO PACKAGE (M16A)
		DAC0808LCN	MC1408P8	DAC0808LCM
8-bit	0°C ≤ T <sub>A</sub> ≤ +75°C			

Note 1: Devices may be ordered by using either order number.



**Absolute Maximum Ratings** (Note 2)

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

## Power Supply Voltage

$V_{CC}$	+18 $V_{DC}$
$V_{EE}$	-18 $V_{DC}$
Digital Input Voltage, V5-V12	-10 $V_{DC}$ to +18 $V_{DC}$
Applied Output Voltage, $V_O$	-11 $V_{DC}$ to +18 $V_{DC}$
Reference Current, $I_{14}$	5 mA
Reference Amplifier Inputs, V14, V15	$V_{CC}$ , $V_{EE}$
Power Dissipation (Note 4)	1000 mW
ESD Susceptibility (Note 5)	TBD

Storage Temperature Range	-65°C to +150°C
Lead Temp. (Soldering, 10 seconds)	
Dual-In-Line Package (Plastic)	260°C
Dual-In-Line Package (Ceramic)	300°C
Surface Mount Package	
Vapor Phase (60 seconds)	215°C
Infrared (15 seconds)	220°C

**Operating Ratings**

Temperature Range	$T_{MIN} \leq T_A \leq T_{MAX}$
DAC0808	$0 \leq T_A \leq +75^\circ\text{C}$

**Electrical Characteristics**

( $V_{CC} = 5V$ ,  $V_{EE} = -15 V_{DC}$ ,  $V_{REF}/R14 = 2 \text{ mA}$ , and all digital inputs at high logic level unless otherwise noted.)

Symbol	Parameter	Conditions	Min	Typ	Max	Units
$E_r$	Relative Accuracy (Error Relative to Full Scale $I_O$ )	(Figure 4)				%
	DAC0808LC (LM1408-8)				<b>±0.19</b>	%
	Settling Time to Within ½ LSB (Includes $t_{PLH}$ )	$T_A = 25^\circ\text{C}$ (Note 7), (Figure 5)		150		ns
$t_{PLH}$ , $t_{PHL}$	Propagation Delay Time	$T_A = 25^\circ\text{C}$ , (Figure 5)		30	100	ns
$TCI_O$	Output Full Scale Current Drift			±20		ppm/°C
MSB $V_{IH}$ $V_{IL}$	Digital Input Logic Levels	(Figure 3)				
	High Level, Logic "1" Low Level, Logic "0"		2		0.8	$V_{DC}$ $V_{DC}$
MSB	Digital Input Current	(Figure 3)				
	High Level Low Level	$V_{IH} = 5V$ $V_{IL} = 0.8V$		0 -0.003	0.040 -0.8	mA mA
$I_{15}$	Reference Input Bias Current	(Figure 3)		-1	-3	µA
	Output Current Range	(Figure 3)				
		$V_{EE} = -5V$ $V_{EE} = -15V$ , $T_A = 25^\circ\text{C}$	0 0	2.0 2.0	2.1 4.2	mA mA
$I_O$	Output Current	$V_{REF} = 2.000V$ , $R14 = 1000\Omega$ , (Figure 3)	1.9	1.99	2.1	mA
	Output Current, All Bits Low	(Figure 3)		0	4	µA
	Output Voltage Compliance (Note 3)	$E_r \leq 0.19\%$ , $T_A = 25^\circ\text{C}$				
	$V_{EE} = -5V$ , $I_{REF} = 1 \text{ mA}$ $V_{EE}$ Below -10V				-0.55, +0.4 -5.0, +0.4	$V_{DC}$ $V_{DC}$
$SRI_{REF}$	Reference Current Slew Rate	(Figure 6)	4	8		mA/µs
	Output Current Power Supply Sensitivity	$-5V \leq V_{EE} \leq -16.5V$		0.05	2.7	µA/V
$I_{CC}$ $I_{EE}$	Power Supply Current (All Bits Low)	(Figure 3)				
				2.3 -4.3	22 -13	mA mA
$V_{CC}$ $V_{EE}$	Power Supply Voltage Range	$T_A = 25^\circ\text{C}$ , (Figure 3)	4.5 -4.5	5.0 -15	5.5 -16.5	$V_{DC}$ $V_{DC}$
	Power Dissipation					

## Electrical Characteristics (Continued)

( $V_{CC} = 5V$ ,  $V_{EE} = -15V$ ,  $V_{REF}/R14 = 2mA$ , and all digital inputs at high logic level unless otherwise noted.)

Symbol	Parameter	Conditions	Min	Typ	Max	Units
	All Bits Low	$V_{CC} = 5V$ , $V_{EE} = -5V$		33	170	mW
		$V_{CC} = 5V$ , $V_{EE} = -15V$		106	305	mW
	All Bits High	$V_{CC} = 15V$ , $V_{EE} = -5V$		90		mW
		$V_{CC} = 15V$ , $V_{EE} = -15V$		160		mW

**Note 2:** Absolute Maximum Ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications do not apply when operating the device beyond its specified operating conditions.

**Note 3:** Range control is not required.

**Note 4:** The maximum power dissipation must be derated at elevated temperatures and is dictated by  $T_{JMAX}$ ,  $\theta_{JA}$ , and the ambient temperature,  $T_A$ . The maximum allowable power dissipation at any temperature is  $P_D = (T_{JMAX} - T_A)/\theta_{JA}$  or the number given in the Absolute Maximum Ratings, whichever is lower. For this device,  $T_{JMAX} = 125^\circ C$ , and the typical junction to ambient thermal resistance of the dual-in-line J package when the board mounted is  $100^\circ C/W$ . For the dual-in-line N package, this number increases to  $175^\circ C/W$  and for the small outline M package this number is  $100^\circ C/W$ .

**Note 5:** Human body model, 100 pF discharged through a 1.5 k $\Omega$  resistor.

**Note 6:** All current switches are tested to guarantee at least 50% of rated current.

**Note 7:** All bits switched.

**Note 8:** Pin-out numbers for the DAL080X represent the dual-in-line package. The small outline package pinout differs from the dual-in-line package.

## Typical Application

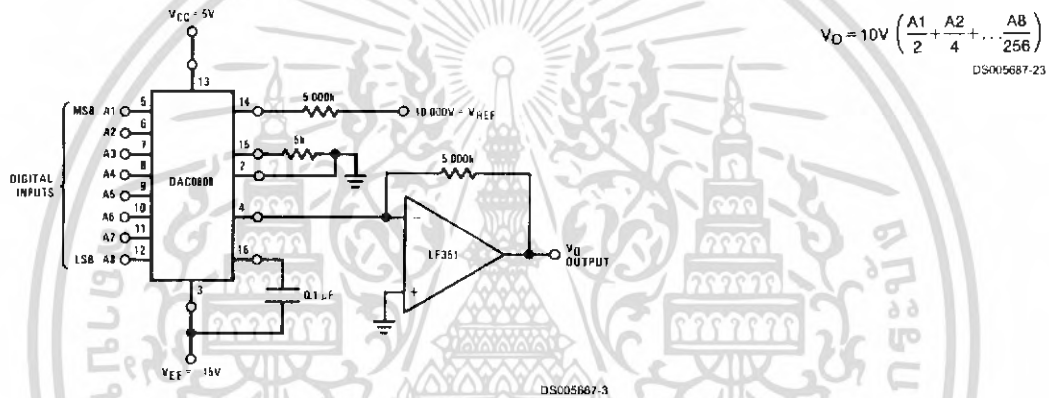
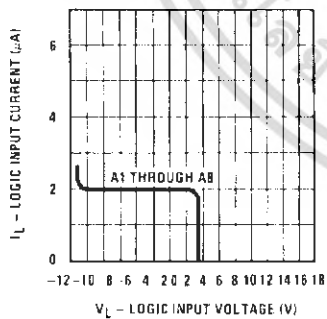


FIGURE 1. +10V Output Digital to Analog Converter (Note 8)

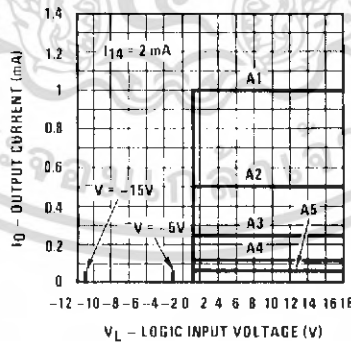
## Typical Performance Characteristics $V_{CC} = 5V$ , $V_{EE} = -15V$ , $T_A = 25^\circ C$ , unless otherwise noted

Logic Input Current vs Input Voltage



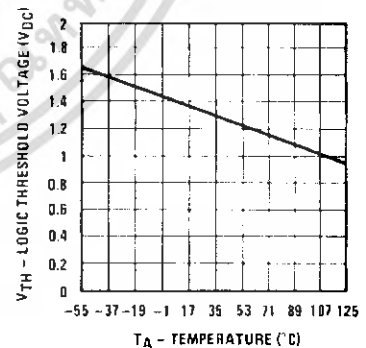
DS005667-14

Bit Transfer Characteristics



DS005687-15

Logic Threshold Voltage vs Temperature

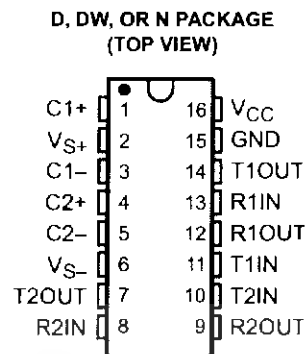


DS005687-16

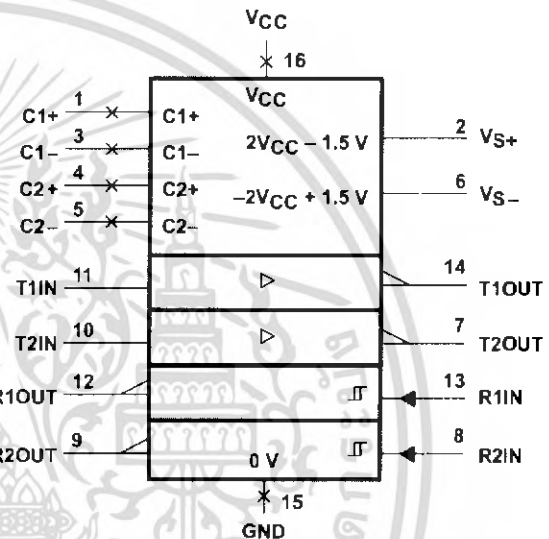
# MAX232, MAX232I DUAL EIA-232 DRIVER/RECEIVER

SLLS047G – FEBRUARY 1989 – REVISED AUGUST 1998

- Operates With Single 5-V Power Supply
- LinBiCMOS™ Process Technology
- Two Drivers and Two Receivers
- ±30-V Input Levels
- Low Supply Current . . . 8 mA Typical
- Meets or Exceeds TIA/EIA-232-F and ITU Recommendation V.28
- Designed to be Interchangeable With Maxim MAX232
- Applications
  - TIA/EIA-232-F
  - Battery-Powered Systems
  - Terminals
  - Modems
  - Computers
- ESD Protection Exceeds 2000 V Per MIL-STD-883, Method 3015
- Package Options Include Plastic Small-Outline (D, DW) Packages and Standard Plastic (N) DIPs



## logic symbol†



† This symbol is in accordance with ANSI/IEEE Std 91-1984 and IEC Publication 617-12.

## description

The MAX232 device is a dual driver/receiver that includes a capacitive voltage generator to supply EIA-232 voltage levels from a single 5-V supply. Each receiver converts EIA-232 inputs to 5-V TTL/CMOS levels. These receivers have a typical threshold of 1.3 V and a typical hysteresis of 0.5 V, and can accept ±30-V inputs. Each driver converts TTL/CMOS input levels into EIA-232 levels. The driver, receiver, and voltage-generator functions are available as cells in the Texas Instruments LinASIC™ library.

The MAX232 is characterized for operation from 0°C to 70°C. The MAX232I is characterized for operation from -40°C to 85°C.

## AVAILABLE OPTIONS

T <sub>A</sub>	PACKAGED DEVICES		
	SMALL OUTLINE (D)	SMALL OUTLINE (DW)	PLASTIC DIP (N)
0°C to 70°C	MAX232D‡	MAX232DW‡	MAX232N
-40°C to 85°C	MAX232ID‡	MAX232IDW‡	MAX232IN

‡ This device is available taped and reeled by adding an R to the part number (i.e., MAX232DR).



Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

LinASIC and LinBiCMOS are trademarks of Texas Instruments Incorporated.

PRODUCTION DATA information is current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

**TEXAS  
INSTRUMENTS**

POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

Copyright © 1998, Texas Instruments Incorporated

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

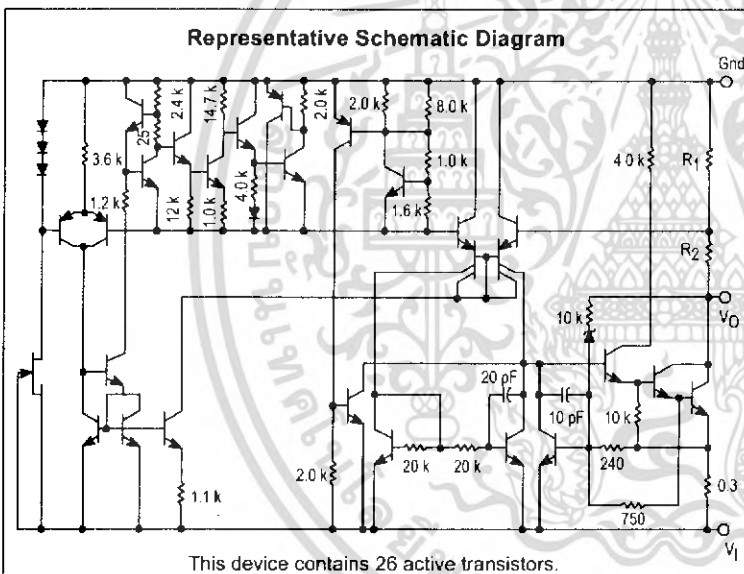


# Three-Terminal Negative Voltage Regulators

The MC7900 series of fixed output negative voltage regulators are intended as complements to the popular MC7800 series devices. These negative regulators are available in the same seven-voltage options as the MC7800 devices. In addition, one extra voltage option commonly employed in MECL systems is also available in the negative MC7900 series.

Available in fixed output voltage options from -5.0 V to -24 V, these regulators employ current limiting, thermal shutdown, and safe-area compensation - making them remarkably rugged under most operating conditions. With adequate heatsinking they can deliver output currents in excess of 1.0 A.

- No External Components Required
- Internal Thermal Overload Protection
- Internal Short Circuit Current Limiting
- Output Transistor Safe-Area Compensation
- Available in 2% Voltage Tolerance (See Ordering Information)



### ORDERING INFORMATION

Device	Output Voltage Tolerance	Operating Temperature Range	Package
MC79XXACO2T	2%	$T_J = 0^\circ \text{ to } +125^\circ\text{C}$	Surface Mount
MC79XXCD2T	4%		
MC79XXACT	2%		Insertion Mount
MC79XXCT	4%		
MC79XXBD2T	4%	$T_J = -40^\circ \text{ to } +125^\circ\text{C}$	Surface Mount
MC79XXBT			Insertion Mount

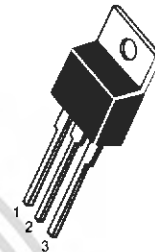
XX indicates nominal voltage.

## MC7900 Series

### THREE-TERMINAL NEGATIVE FIXED VOLTAGE REGULATORS

T SUFFIX  
PLASTIC PACKAGE  
CASE 221A

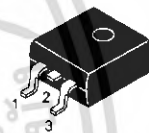
Heatsink surface connected to Pin 2.



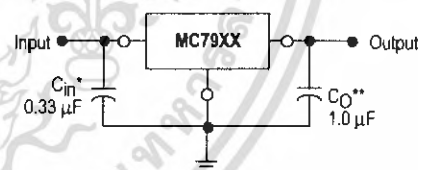
Pin 1. Ground  
Pin 2. Input  
Pin 3. Output

D2T SUFFIX  
PLASTIC PACKAGE  
CASE 936  
(D<sup>2</sup>PAK)

Heatsink surface (shown as terminal 4 in case outline drawing) is connected to Pin 2.



### STANDARD APPLICATION



A common ground is required between the input and the output voltages. The input voltage must remain typically 2.0 V above more negative even during the high point of the input ripple voltage.

XX, These two digits of the type number indicate nominal voltage.

- \*  $C_{in}$  is required if regulator is located an appreciable distance from power supply filter.
- \*\*  $C_O$  improve stability and transient response

### DEVICE TYPE/NOMINAL OUTPUT VOLTAGE

MC7905	5.0 V	MC7912	12 V
MC7905.2	5.2 V	MC7915	15 V
MC7906	6.0 V	MC7918	28 V
MC7908	8.0 V	MC7924	24 V

# MC7900

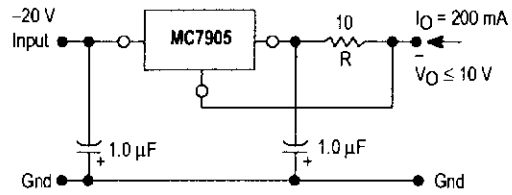
## APPLICATIONS INFORMATION

### Design Considerations

The MC7900 Series of fixed voltage regulators are designed with Thermal overload Protection that shuts down the circuit when subjected to an excessive power overload condition. Internal Short Circuit Protection that limits the maximum current the circuit will pass, and Output Transistor Safe-Area Compensation that reduces the output short circuit current as the voltage across the pass transistor is increased.

In many low current applications, compensation capacitors are not required. However, it is recommended that the regulator input be bypassed with a capacitor if the regulator is connected to the power supply filter with long wire lengths, or if the output load capacitance is large. An input bypass capacitor should be selected to provide good high-frequency characteristics to insure stable operation under all load conditions. A 0.33  $\mu\text{F}$  or larger tantalum, mylar, or other capacitor having low internal impedance at high frequencies should be chosen. The capacitor chosen should have an equivalent series resistance of less than 0.7  $\Omega$ . The bypass capacitor should be mounted with the shortest possible leads directly across the regulators input terminals. Normally good construction techniques should be used to minimize ground loops and lead resistance drops since the regulator has no external sense lead. Bypassing the output is also recommended.

**Figure 7. Current Regulator**

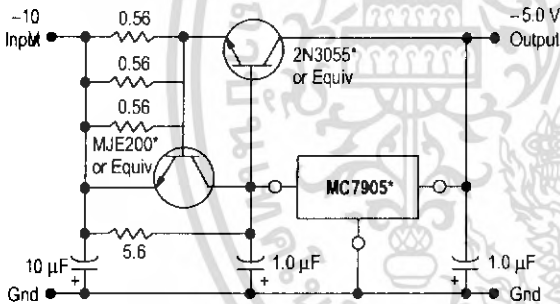


The MC7905, -5.0 V regulator can be used as a constant current source when connected as above. The output current is the sum of resistor R current and quiescent bias current as follows.

$$I_O = \frac{5.0 \text{ V}}{R} + I_B$$

The quiescent current for this regulator is typically 4.3 mA. The 5.0 V regulator was chosen to minimize dissipation and to allow the output voltage to operate to within 6.0 V below the input voltage.

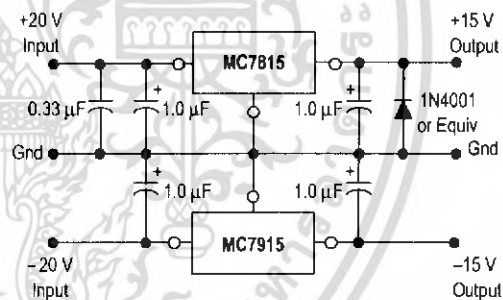
**Figure 8. Current Boost Regulator**  
(-5.0 V @ 4.0 A, with 5.0 A Current Limiting)



\*Mounted on heatsink.

When a boost transistor is used, short circuit currents are equal to the sum of the series pass and regulator limits, which are measured at 3.2 A and 1.8 A respectively in this case. Series pass limiting is approximately equal to  $0.6 \text{ V}/R_{SC}$ . Operation beyond this point to the peak current capability of the MC7905C is possible if the regulator is mounted on a heatsink; otherwise thermal shutdown will occur when the additional load current is picked up by the regulator.

**Figure 9. Operational Amplifier Supply**  
(±15 @ 1.0 A)



The MC7815 and MC7915 positive and negative regulators may be connected as shown to obtain a dual power supply for operational amplifiers. A clamp diode should be used at the output of the MC7815 to prevent potential latch-up problems whenever the output of the positive regulator (MC7815) is drawn below ground with an output current greater than 200 mA.



MICROCHIP

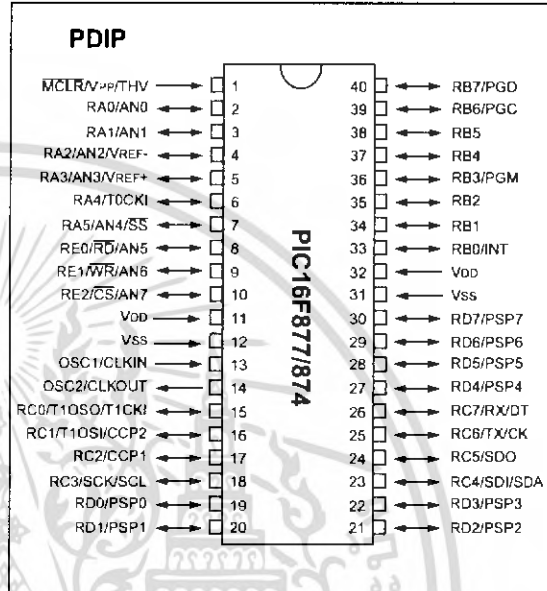
# PIC16F87X

## 28/40-pin 8-Bit CMOS EEPROM/Flash Microcontrollers

### Microcontroller Core Features:

- High-performance RISC CPU
- Only 35 single word instructions to learn
- All single cycle instructions except for program branches which are two cycle
- Operating speed: DC - 20 MHz clock input  
DC - 200 ns instruction cycle
- Up to 8K x 14 words of Flash Program Memory,  
Up to 368 x 8 bytes of Data Memory (RAM)  
Up to 256 x 8 bytes of EEPROM data memory
- ★ • Pinout compatible to the PIC16C73/74/76/77
- Interrupt capability (up to 14 internal/external interrupt sources)
- Eight level deep hardware stack
- Direct, indirect, and relative addressing modes
- Power-on Reset (POR)
- Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
- Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation
- Programmable code-protection
- Power saving SLEEP mode
- Selectable oscillator options
- Low-power, high-speed CMOS EPROM/EEPROM technology
- Fully static design
- In-Circuit Serial Programming™ via two pins
- ★ • Only single 5V source needed for programming
- ★ • In-Circuit Debugging via two pins
- Processor read/write access to program memory
- Wide operating voltage range: 2.5V to 5.5V
- High Sink/Source Current: 25 mA
- Commercial and Industrial temperature ranges
- Low-power consumption:
  - < 2 mA typical @ 5V, 4 MHz
  - 20 µA typical @ 3V, 32 kHz
  - < 1 µA typical standby current

### Pin Diagram



### Peripheral Features:

- Timer0: 8-bit timer/counter with 8-bit prescaler
- Timer1: 16-bit timer/counter with prescaler, can be incremented during sleep via external crystal/clock
- Timer2: 8-bit timer/counter with 8-bit period register, prescaler and postscaler
- Two Capture, Compare, PWM modules
- Capture is 16-bit, max. resolution is 12.5 ns, Compare is 16-bit, max. resolution is 200 ns, PWM max. resolution is 10-bit
- ★ • 10-bit multi-channel Analog-to-Digital converter
- ★ • Synchronous Serial Port (SSP) with SPI™ (Master Mode) and I<sup>2</sup>C™ (Master/Slave)
- ★ • Universal Synchronous Asynchronous Receiver Transmitter (USART/SCI) with 9-bit address detection
- Parallel Slave Port (PSP) 8-bits wide, with external  $\overline{RD}$ ,  $\overline{WR}$  and  $\overline{CS}$  controls (40/44-pin only)
- Brown-out detection circuitry for Brown-out Reset (BOR)

This is an advanced copy of the data sheet and therefore the contents and specifications are subject to change based on device characterization.

## 3.0 I/O PORTS

Some pins for these I/O ports are multiplexed with an alternate function for the peripheral features on the device. In general, when a peripheral is enabled, that pin may not be used as a general purpose I/O pin.

Additional information on I/O ports may be found in the PICmicro™ Mid-Range Reference Manual, (DS33023).

### 3.1 PORTA and the TRISA Register

PORTA is a 6-bit wide bi-directional port. The corresponding data direction register is TRISA. Setting a TRISA bit (=1) will make the corresponding PORTA pin an input, i.e., put the corresponding output driver in a hi-impedance mode. Clearing a TRISA bit (=0) will make the corresponding PORTA pin an output, i.e., put the contents of the output latch on the selected pin.

Reading the PORTA register reads the status of the pins whereas writing to it will write to the port latch. All write operations are read-modify-write operations. Therefore a write to a port implies that the port pins are read, this value is modified, and then written to the port data latch.

Pin RA4 is multiplexed with the Timer0 module clock input to become the RA4/T0CKI pin. The RA4/T0CKI pin is a Schmitt Trigger input and an open drain output. All other RA port pins have TTL input levels and full CMOS output drivers.

Other PORTA pins are multiplexed with analog inputs and analog VREF input. The operation of each pin is selected by clearing/setting the control bits in the ADCON1 register (A/D Control Register1).

**Note:** On a Power-on Reset, these pins are configured as analog inputs and read as '0'.

The TRISA register controls the direction of the RA pins, even when they are being used as analog inputs. The user must ensure the bits in the TRISA register are maintained set when using them as analog inputs.

#### EXAMPLE 3-1: INITIALIZING PORTA

```
BCF STATUS, RP0 ;
CLRF PORTA      ; Initialize PORTA by
                 ; clearing output
                 ; data latches
BSF STATUS, RP0 ; Select Bank 1
MOVLW 0xCF      ; Value used to
                 ; initialize data
                 ; direction
MOVWF TRISA     ; Set RA<3:0> as inputs
                 ; RA<5:4> as outputs
                 ; TRISA<7:6> are always
                 ; read as '0'.
```

FIGURE 3-1: BLOCK DIAGRAM OF RA3:RA0 AND RA5 PINS

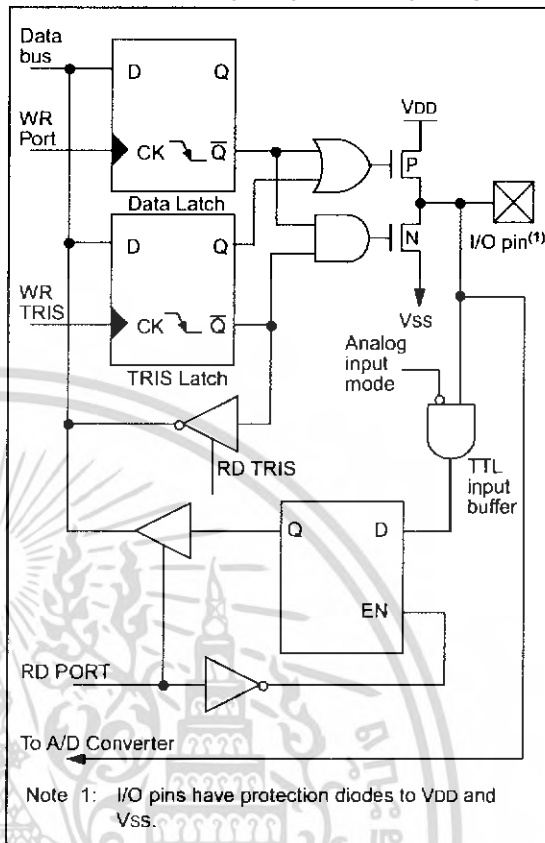
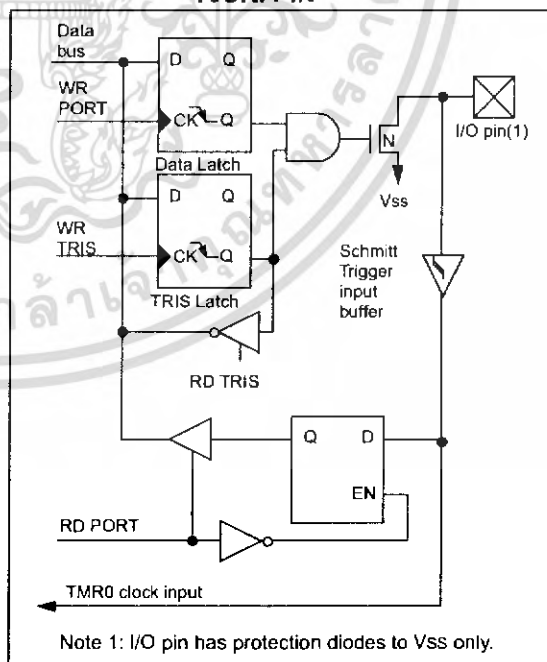


FIGURE 3-2: BLOCK DIAGRAM OF RA4/T0CKI PIN



# PIC16F87X

**TABLE 3-1 PORTA FUNCTIONS**

Name	Bit#	Buffer	Function
RA0/AN0	bit0	TTL	Input/output or analog input
RA1/AN1	bit1	TTL	Input/output or analog input
RA2/AN2	bit2	TTL	Input/output or analog input
RA3/AN3/VREF	bit3	TTL	Input/output or analog input or VREF
RA4/T0CK1	bit4	ST	Input/output or external clock input for Timer0 Output is open drain type
RA5/SS/AN4	bit5	TTL	Input/output or slave select input for synchronous serial port or analog input

Legend: TTL = TTL input, ST = Schmitt Trigger input

**TABLE 3-2 SUMMARY OF REGISTERS ASSOCIATED WITH PORTA**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets
05h	PORTA	—	—	RA5	RA4	RA3	RA2	RA1	RA0	--0x 0000	--0u 0000
85h	TRISA	—	—	PORTA Data Direction Register						--11 1111	--11 1111
9Fh	ADCON1	—	—	ADFM	—	PCFG3	PCFG2	PCFG1	PCFG0	--0- 0000	--0- 0000

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by PORTA.



## 3.2 PORTB and the TRISB Register

PORTB is an 8-bit wide bi-directional port. The corresponding data direction register is TRISB. Setting a TRISB bit (=1) will make the corresponding PORTB pin an input, i.e., put the corresponding output driver in a hi-impedance mode. Clearing a TRISB bit (=0) will make the corresponding PORTB pin an output, i.e., put the contents of the output latch on the selected pin.

Three pins of PORTB are multiplexed with the Low Voltage Programming function; RB3/PGM, RB6/PGC and RB7/PGD. The alternate functions of these pins are described in the Special Features Section.

### EXAMPLE 3-1: INITIALIZING PORTB

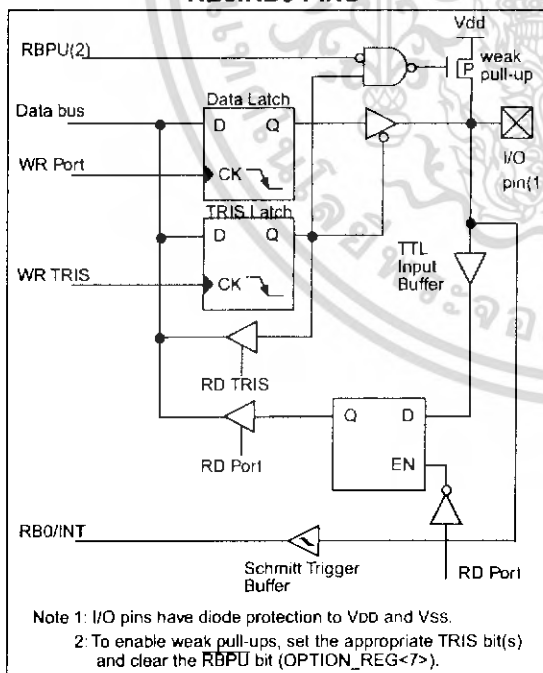
```
BCF STATUS, RPO ;
CLRF PORTB      ; Initialize PORTB by
                 ; clearing output
                 ; data latches

BSF STATUS, RPO ; Select Bank 1
MOVLW 0xCF      ; Value used to
                 ; initialize data
                 ; direction

MOVWF TRISB     ; Set RB<3:0> as inputs
                 ; RB<5:4> as outputs
                 ; RB<7:6> as inputs
```

Each of the PORTB pins has a weak internal pull-up. A single control bit can turn on all the pull-ups. This is performed by clearing bit  $\overline{\text{RBP}}\text{U}$  (OPTION\_REG<7>). The weak pull-up is automatically turned off when the port pin is configured as an output. The pull-ups are disabled on a Power-on Reset.

**FIGURE 3-3: BLOCK DIAGRAM OF RB3:RB0 PINS**



Four of PORTB's pins, RB7:RB4, have an interrupt on change feature. Only pins configured as inputs can cause this interrupt to occur (i.e. any RB7:RB4 pin configured as an output is excluded from the interrupt on change comparison). The input pins (of RB7:RB4) are compared with the old value latched on the last read of PORTB. The "mismatch" outputs of RB7:RB4 are OR'ed together to generate the RB Port Change Interrupt with flag bit RBIF (INTCON<0>).

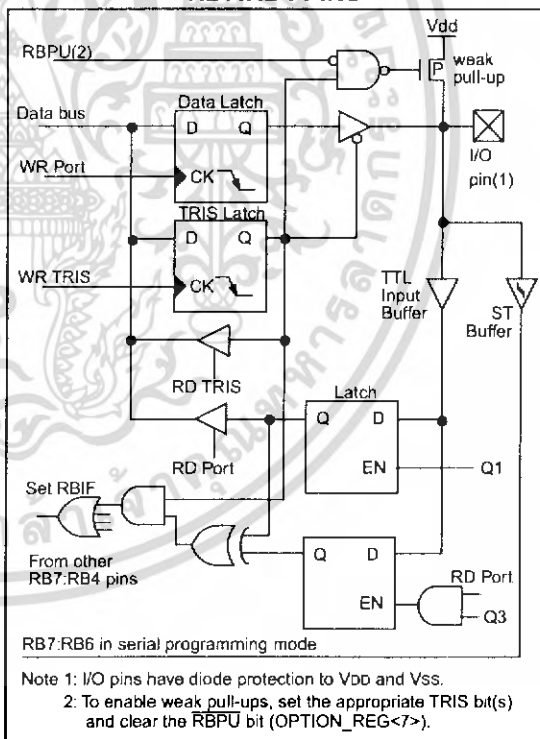
This interrupt can wake the device from SLEEP. The user, in the interrupt service routine, can clear the interrupt in the following manner:

- Any read or write of PORTB. This will end the mismatch condition.
- Clear flag bit RBIF.

A mismatch condition will continue to set flag bit RBIF. Reading PORTB will end the mismatch condition, and allow flag bit RBIF to be cleared.

The interrupt on change feature is recommended for wake-up on key depression operation and operations where PORTB is only used for the interrupt on change feature. Polling of PORTB is not recommended while using the interrupt on change feature.

**FIGURE 3-4: BLOCK DIAGRAM OF RB7:RB4 PINS**



# PIC16F87X

**TABLE 3-3 PORTB FUNCTIONS**

Name	Bit#	Buffer	Function
RB0/INT	bit0	TTL/ST <sup>(1)</sup>	Input/output pin or external interrupt input. Internal software programmable weak pull-up.
RB1	bit1	TTL	Input/output pin. Internal software programmable weak pull-up.
RB2	bit2	TTL	Input/output pin. Internal software programmable weak pull-up.
RB3/PGM	bit3	TTL	Input/output pin or programming pin in LVP mode. Internal software programmable weak pull-up.
RB4	bit4	TTL	Input/output pin (with interrupt on change). Internal software programmable weak pull-up.
RB5	bit5	TTL	Input/output pin (with interrupt on change). Internal software programmable weak pull-up.
RB6/PGC	bit6	TTL/ST <sup>(2)</sup>	Input/output pin (with interrupt on change) or In-Circuit Debugger pin. Internal software programmable weak pull-up. Serial programming clock.
RB7/PGD	bit7	TTL/ST <sup>(2)</sup>	Input/output pin (with interrupt on change) or In-Circuit Debugger pin. Internal software programmable weak pull-up. Serial programming data.

Legend: TTL = TTL input, ST = Schmitt Trigger input

Note 1: This buffer is a Schmitt Trigger input when configured as the external interrupt.

Note 2: This buffer is a Schmitt Trigger input when used in serial programming mode.

**TABLE 3-4 SUMMARY OF REGISTERS ASSOCIATED WITH PORTB**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets
06h, 106h	PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxx	uuuu uuuu
86h, 186h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
81h, 181h	OPTION_ REG	RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111

Legend: x = unknown, u = unchanged. Shaded cells are not used by PORTB.

### 3.3 PORTC and the TRISC Register

PORTC is an 8-bit wide bi-directional port. The corresponding data direction register is TRISC. Setting a TRISC bit (=1) will make the corresponding PORTC pin an input, i.e., put the corresponding output driver in a hi-impedance mode. Clearing a TRISC bit (=0) will make the corresponding PORTC pin an output, i.e., put the contents of the output latch on the selected pin.

PORTC is multiplexed with several peripheral functions (Table 3-5). PORTC pins have Schmitt Trigger input buffers.

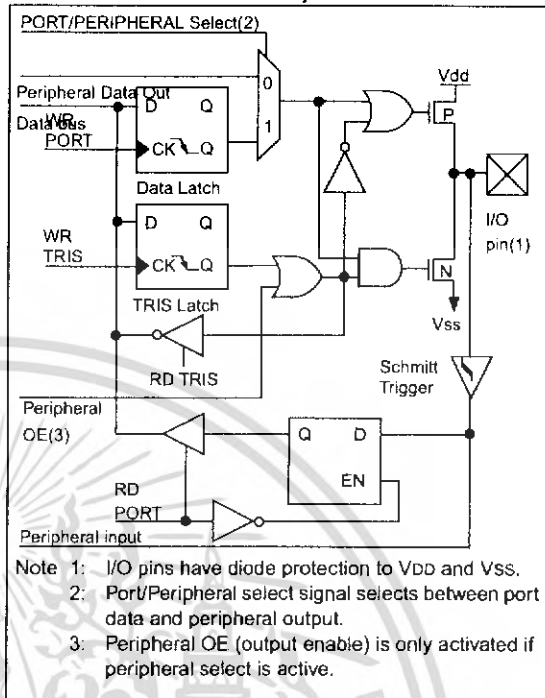
When the I<sup>2</sup>C module is enabled, the PORTC (3:4) pins can be configured with normal I<sup>2</sup>C levels or with SMBUS levels by using the CKE bit (SSPSTAT <6>).

When enabling peripheral functions, care should be taken in defining TRIS bits for each PORTC pin. Some peripherals override the TRIS bit to make a pin an output, while other peripherals override the TRIS bit to make a pin an input. Since the TRIS bit override is in effect while the peripheral is enabled, read-modify-write instructions (BSF, BCF, XORWF) with TRISC as destination should be avoided. The user should refer to the corresponding peripheral section for the correct TRIS bit settings.

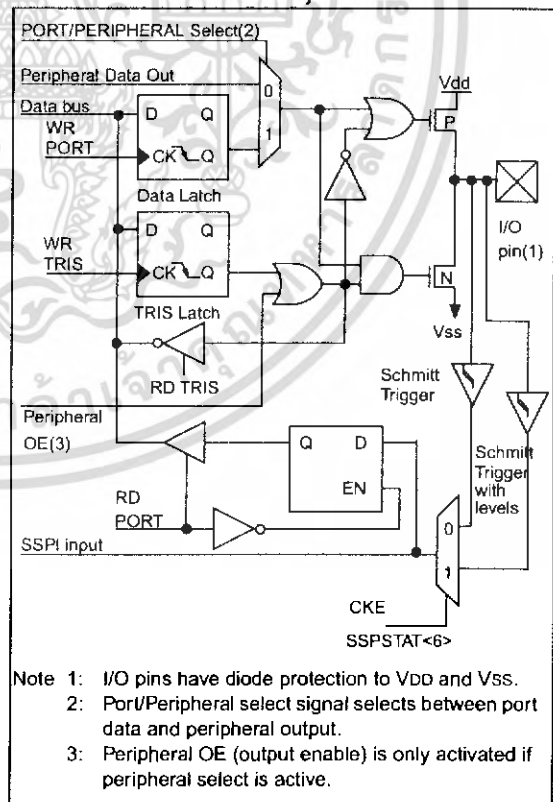
#### EXAMPLE 3-1: INITIALIZING PORTC

```
BCF STATUS, RP0 ; Select Bank 0
CLRF PORTC ; Initialize PORTC by
; clearing output
; data latches
BSF STATUS, RP0 ; Select Bank 1
MOVLW 0xCF ; Value used to
; initialize data
; direction
MOVWF TRISC ; Set RC<3:0> as inputs
; RC<5:4> as outputs
; RC<7:6> as inputs
```

**FIGURE 3-5: PORTC BLOCK DIAGRAM (PERIPHERAL OUTPUT OVERRIDE) RC<0:2> RC<5:7>**



**FIGURE 3-6: PORTC BLOCK DIAGRAM (PERIPHERAL OUTPUT OVERRIDE) RC<3:4>**



# PIC16F87X

**TABLE 3-5 PORTC FUNCTIONS**

Name	Bit#	Buffer Type	Function
RC0/T1OSO/T1CKI	bit0	ST	Input/output port pin or Timer1 oscillator output/Timer1 clock input
RC1/T1OSI/CCP2	bit1	ST	Input/output port pin or Timer1 oscillator input or Capture2 input/Compare2 output/PWM2 output
RC2/CCP1	bit2	ST	Input/output port pin or Capture1 input/Compare1 output/PWM1 output
RC3/SCK/SCL	bit3	ST	RC3 can also be the synchronous serial clock for both SPI and I <sup>2</sup> C modes.
RC4/SDI/SDA	bit4	ST	RC4 can also be the SPI Data In (SPI mode) or data I/O (I <sup>2</sup> C mode).
RC5/SDO	bit5	ST	Input/output port pin or Synchronous Serial Port data output
RC6/TX/CK	bit6	ST	Input/output port pin or USART Asynchronous Transmit or Synchronous Clock
RC7/RX/DT	bit7	ST	Input/output port pin or USART Asynchronous Receive or Synchronous Data

Legend: ST = Schmitt Trigger input

**TABLE 3-6 SUMMARY OF REGISTERS ASSOCIATED WITH PORTC**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets
07h	PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	nnnn nnnn
87h	TRISC	PORTC Data Direction Register								1111 1111	1111 1111

Legend: x = unknown, u = unchanged.

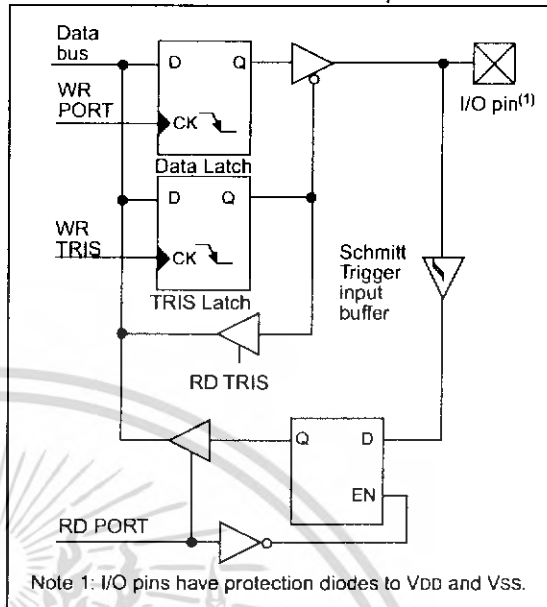
### 3.4 PORTD and TRISD Registers

This section is not applicable to the 28-pin devices.

PORTD is an 8-bit port with Schmitt Trigger input buffers. Each pin is individually configurable as an input or output.

PORTD can be configured as an 8-bit wide microprocessor port (parallel slave port) by setting control bit PSPMODE (TRISE<4>). In this mode, the input buffers are TTL.

**FIGURE 3-7: PORTD BLOCK DIAGRAM (IN I/O PORT MODE)**



**TABLE 3-7 PORTD FUNCTIONS**

Name	Bit#	Buffer Type	Function
RD0/PSP0	bit0	ST/TTL <sup>(1)</sup>	Input/output port pin or parallel slave port bit0
RD1/PSP1	bit1	ST/TTL <sup>(1)</sup>	Input/output port pin or parallel slave port bit1
RD2/PSP2	bit2	ST/TTL <sup>(1)</sup>	Input/output port pin or parallel slave port bit2
RD3/PSP3	bit3	ST/TTL <sup>(1)</sup>	Input/output port pin or parallel slave port bit3
RD4/PSP4	bit4	ST/TTL <sup>(1)</sup>	Input/output port pin or parallel slave port bit4
RD5/PSP5	bit5	ST/TTL <sup>(1)</sup>	Input/output port pin or parallel slave port bit5
RD6/PSP6	bit6	ST/TTL <sup>(1)</sup>	Input/output port pin or parallel slave port bit6
RD7/PSP7	bit7	ST/TTL <sup>(1)</sup>	Input/output port pin or parallel slave port bit7

Legend: ST = Schmitt Trigger input TTL = TTL input

Note 1: Input buffers are Schmitt Triggers when in I/O mode and TTL buffer when in Parallel Slave Port Mode.

**TABLE 3-8 SUMMARY OF REGISTERS ASSOCIATED WITH PORTD**

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets
08h	PORTD	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	xxxx xxxx	bbbb bbbb
88h	TRISD	PORTD Data Direction Register								1111 1111	1111 1111
89h	TRISE	IBF	OBF	IBOV	PSPMODE	—	PORTE Data Direction Bits			0000 -111	0000 -111

Legend: x = unknown, u = unchanged, - = unimplemented read as '0'. Shaded cells are not used by PORTD.

# PIC16F87X

## 3.5 PORTE and TRISE Register

This section is not applicable to the 28-pin devices.

PORTE has three pins RE0/ $\overline{RD}$ /AN5, RE1/ $\overline{WR}$ /AN6 and RE2/ $\overline{CS}$ /AN7, which are individually configurable as inputs or outputs. These pins have Schmitt Trigger input buffers.

I/O PORTE becomes control inputs for the microprocessor port when bit PSPMODE (TRISE<4>) is set. In this mode, the user must make sure that the TRISE<2:0> bits are set (pins are configured as digital inputs). Ensure ADCON1 is configured for digital I/O. In this mode the input buffers are TTL.

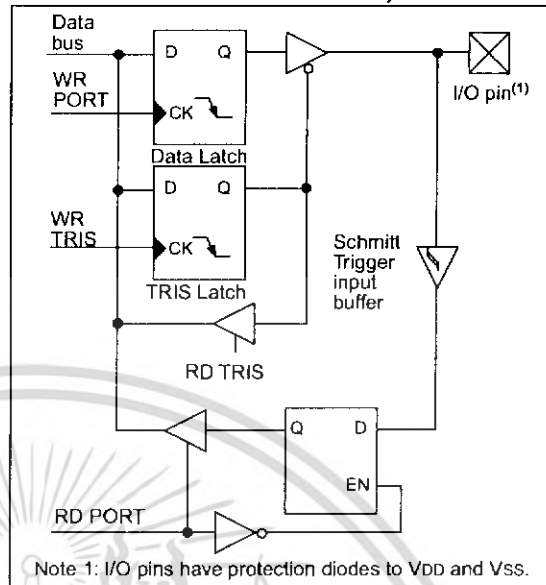
Figure 3-9 shows the TRISE register, which also controls the parallel slave port operation.

PORTE pins are multiplexed with analog inputs. When selected as an analog input, these pins will read as '0's.

TRISE controls the direction of the RE pins, even when they are being used as analog inputs. The user must make sure to keep the pins configured as inputs when using them as analog inputs.

**Note:** On a Power-on Reset these pins are configured as analog inputs.

FIGURE 3-8: PORTE BLOCK DIAGRAM (IN I/O PORT MODE)



Note 1: I/O pins have protection diodes to VDD and VSS.

FIGURE 3-9: TRISE REGISTER (ADDRESS 89h)

R-0	R-0	R/W-0	R/W-0	U-0	R/W-1	R/W-1	R/W-1	
IBF	OBF	IBOV	PSPMODE	—	bit2	bit1	bit0	R= Readable bit W= Writable bit U= Unimplemented bit, read as '0' - n= Value at POR reset
bit7								bit0
bit 7 : <b>IBF</b> : Input Buffer Full Status bit 1 = A word has been received and is waiting to be read by the CPU 0 = No word has been received								
bit 6 : <b>OBF</b> : Output Buffer Full Status bit 1 = The output buffer still holds a previously written word 0 = The output buffer has been read								
bit 5 : <b>IBOV</b> : Input Buffer Overflow Detect bit (in microprocessor mode) 1 = A write occurred when a previously input word has not been read (must be cleared in software) 0 = No overflow occurred								
bit 4 : <b>PSPMODE</b> : Parallel Slave Port Mode Select bit 1 = Parallel slave port mode 0 = General purpose I/O mode								
bit 3 : <b>Unimplemented</b> : Read as '0'								
<b>PORTE Data Direction Bits</b>								
bit 2 : <b>Bit2</b> : Direction Control bit for pin RE2/ $\overline{CS}$ /AN7 1 = Input 0 = Output								
bit 1 : <b>Bit1</b> : Direction Control bit for pin RE1/ $\overline{WR}$ /AN6 1 = Input 0 = Output								
bit 0 : <b>Bit0</b> : Direction Control bit for pin RE0/ $\overline{RD}$ /AN5 1 = Input 0 = Output								

**TABLE 3-9 PORTE FUNCTIONS**

Name	Bit#	Buffer Type	Function
RE0/ $\overline{RD}$ /AN5	bit0	ST/TTL <sup>(1)</sup>	Input/output port pin or read control input in parallel slave port mode or analog input: $\overline{RD}$ 1 = Not a read operation 0 = Read operation. Reads PORTD register (if chip selected)
RE1/ $\overline{WR}$ /AN6	bit1	ST/TTL <sup>(1)</sup>	Input/output port pin or write control input in parallel slave port mode or analog input: $\overline{WR}$ 1 = Not a write operation 0 = Write operation. Writes PORTD register (if chip selected)
RE2/ $\overline{CS}$ /AN7	bit2	ST/TTL <sup>(1)</sup>	Input/output port pin or chip select control input in parallel slave port mode or analog input: $\overline{CS}$ 1 = Device is not selected 0 = Device is selected

Legend: ST = Schmitt Trigger input TTL = TTL input

Note 1: Input buffers are Schmitt Triggers when in I/O mode and TTL buffers when in Parallel Slave Port Mode.

**TABLE 3-10 SUMMARY OF REGISTERS ASSOCIATED WITH PORTE**

Addr	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets
09h	PORTE	—	—	—	—	—	RE2	RE1	RE0	--- -xxx	--- -uuu
89h	TRISE	IBF	OBF	IBOV	PSPMODE	—	PORTE Data Direction Bits			0000 -111	0000 -111
9Fh	ADCON1	—	—	ADFM	—	PCFG3	PCFG2	PCFG1	PCFG0	--0- 0000	--0- 0000

Legend: x = unknown, u = unchanged, - = unimplemented read as '0'. Shaded cells are not used by PORTE.