สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เครื่องคำนวณกำลังงานไฟฟ้า

ELECTRICITY POWER MEASUREMENT

T046270

นาย มงคล กิตติโสภาวดี
นาย สมชาย แซ่เตียว

เลขหมู่.....
เลขทะเบียน 46270
วัน, เดือน, ปี 21 ส.ค. 2546

.b.....
.i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมอิเล็กทรอนิกส์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ประจำภาคเรียนที่ 2 ปีการศึกษา 2544

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2544

ภาควิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เครื่องคำนวณกำลังงานไฟฟ้า

Electricity Power Measurement

ผู้จัดทำ

1. นาย มงคล กิตติโสภาวดี รหัสประจำตัว 41014334
2. นาย สมชาย แซ่เตียว รหัสประจำตัว 41014439

ลงชื่อ.....อาจารย์ที่ปรึกษา

(รศ.อิทธิชัย อรุณศรีแสงไชย)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องคำนวณกำลังไฟฟ้า

นายมงคล กิตติโสภาวดี 41014334
นายสมชาย แซ่เตียว 41014439
อาจารย์อิทธิชัย อรุณศรีแสงไชย ที่ปรึกษา
ปีการศึกษา 2544

บทคัดย่อ

โครงงานนี้เป็นโครงงานที่ให้กำเนิดแนวคิดในการออกแบบและการสร้าง อุปกรณ์ที่สามารถวัดกำลังไฟฟ้าที่เราต้องสูญเสียไปในแต่ละวัน โดยพื้นฐานของการวัดมิเตอร์ นำมาประยุกต์ใช้โดยนำข้อมูลที่ได้มาคำนวณในแบบดิจิตอล และแสดงผลออกมาได้

โครงงานนี้เป็นโครงงานที่สร้างขึ้นมาด้วยแนวความคิด ในการสร้างอุปกรณ์ที่นอกเหนือไปจากที่มีอยู่แล้วและมีประโยชน์โดยทั่วๆไปกับทุกๆคน ระบบต่างจึงเป็นระบบที่คิดขึ้นมาใหม่ โดยอธิบายส่วนประกอบคร่าวๆได้ 4 ส่วนคือ ส่วนแรกเป็นส่วนของการรับอินพุทซึ่งเป็นค่าแรงดันที่จะนำไปใช้ในการคำนวณ ส่วนที่สองเป็นการแปลงสัญญาณที่ได้ออกมาเป็นข้อมูลทางดิจิตอลเพื่อนำมาใช้ในการคำนวณต่อไป ส่วนที่สามเป็นส่วนของการคำนวณด้วยไมโครคอนโทรลเลอร์ และส่วนสุดท้ายคือการนำข้อมูลที่คำนวณแล้วไปแสดงผลนั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Electricity Power Measurement

Mr.Mongkol Kittisopavadee 41014334
Mr.Somchai Sae-tiaw 41014439
Mr.Ittichai Arunsrisangchai Advisor
2001

Abstract

This project describes the design and construction of Electricity Power Meter. The meter is built to measure the average power and watt-hour power, display data to user

This system structure is divided into four parts, input receiving part, Analog to Digital Converter (ADC) part, calculating part and data display. This new system will provide convenience in calculating electricity power and is very easy to use which can be further developed as commercial product in the future.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์เล่มนี้คงจะสำเร็จขึ้นมาไม่ได้ หากไม่ได้รับความช่วยเหลือจากบุคคลหลายๆท่าน ซึ่งบุคคลแรกที่ต้องกล่าวถึง คือท่านอาจารย์ อิทธิชัย อรุณศรีแสงไชย อาจารย์ที่ปรึกษา ท่านได้พยายามสร้างแนวคิดใหม่ ให้คำปรึกษา และเอาใจใส่ตลอดมา ต้องขอขอบพระคุณเป็นอย่างมากครับ และความรู้ต่างๆที่นำมาซึ่งความสำเร็จของปริญญานิพนธ์เล่มนี้ ก็มาจากท่านอาจารย์หลายๆท่านที่ได้ประสิทธิประสาทวิชาความรู้ ให้ข้าพเจ้าได้นำความรู้มาใช้จนประสบผลสำเร็จได้ ขอขอบพระคุณท่านอาจารย์ทุกๆท่านด้วยครับ ไม่เพียงเท่านี้ตลอดเวลาที่ผ่านมายังมีทั้งรุ่นพี่ ทั้งเพื่อนหลายๆคนที่คอยให้คำปรึกษาในบางโอกาส และรุ่นน้องอีกหลายคนที่เป็นกำลังใจให้ จนถึงวันนี้วันที่ปริญญานิพนธ์เล่มนี้สมบูรณ์พร้อม ข้าพเจ้าคิดว่าผลงานนี้ถือเป็นของทุกๆท่านด้วย

สุดท้ายแล้วข้าพเจ้ายังต้องขอขอบคุณ ภาควิชาอิเล็กทรอนิกส์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง และองค์พระบาทสมเด็จพระจอมเกล้าเจ้าอยู่หัวผู้ทรงเป็นพระบิดาแห่งวิทยาศาสตร์ไทย และเป็นที่รักยิ่งของพวกเราลูกหลานพระจอมเกล้าทุกคน ข้าพเจ้าจะนำความรู้ที่ได้นำมาสร้างประโยชน์และพัฒนาประเทศชาติต่อไป

มงคล กิตติโสภาวดี
สมชาย แซ่เตียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อ	I
Abstract	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูป	VI
สารบัญตาราง	VII
บทที่ 1 บทนำ	1
1.1 ความสำคัญและที่มา	1
1.2 วัตถุประสงค์ของโครงงาน	1
1.3 แนวคิดในการออกแบบ	1
1.4 หลักการทำงานเบื้องต้น	2
บทที่ 2 ทฤษฎีที่ใช้ในการคำนวณ	3
2.1 กำลังงานไฟฟ้า	3
2.2 Watt-hour Power	3
2.3 Average Power	3
2.4 นิยามของค่าต่างๆ	5
บทที่ 3 ไมโครคอนโรลเลอร์และอุปกรณ์	9
3.1 ไมโครคอนโทรลเลอร์	9
3.2 Analog to Digital Converter (ADC)	16
3.3 LCD Display	17
บทที่ 4 โครงสร้างของอุปกรณ์	19
4.1 ส่วนของอินพุท	19
4.2 ส่วนของการแปลงสัญญาณ	20
4.3 ส่วนของคอนโทรลเลอร์	21
4.4 ส่วนของการแสดงผล	21
4.5 วงจรรวม	22
4.6 โปรแกรมการทำงานของเครื่องวัดกำลังไฟฟ้าภายในบ้าน	23

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
บทที่ 5 ผลการทดลอง	60
5.1 การทดลอง	60
5.2 ผลการทดลอง	60
บทที่ 6 บทสรุปและวิจารณ์	71
6.1 บทสรุป	71
6.2 บทวิจารณ์	71
ภาคผนวก ก	72
ภาคผนวก ข	103
ภาคผนวก ค	120
หนังสืออ้างอิง	128

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

หน้า

รูปที่ 1.1 แสดงลักษณะการเชื่อมต่อของอุปกรณ์ Electricity Power Device 1

รูปที่ 1.2 Block diagram ของการทำงานใน Electricity Cost Device 2

รูปที่ 2.1 (a) วงจรลดขนาดสัญญาณ โดยให้ Vout เป็นแรงดันจากไฟบ้าน (b) วงจรลดขนาดสัญญาณ โดยให้ Vout เป็นแรงดันที่คร่อมตัวต้านทานขนาด 100 โอห์ม 3

รูปที่ 3.1 การจัดวางขาของ 10

รูปที่ 3.2 วงจรสร้างClockของ 8051 13

รูปที่ 3.3 ผังหน่วยความจำสำหรับเก็บโปรแกรมสำหรับเบอร์ 8051 14

รูปที่ 3.4 ผังหน่วยความจำสำหรับเก็บโปรแกรมสำหรับเบอร์ 8052 14

รูปที่ 3.5 ผังหน่วยความจำสำหรับเก็บ Data Memory เบอร์ 8051 15

รูปที่ 3.6 ผังหน่วยความจำสำหรับเก็บ Data Memory เบอร์ 8052 15

รูปที่ 3.7 แสดงการต่อ ADC0804 เข้ากับไมโครคอนโทรลเลอร์ 16

รูปที่ 3.8 แสดงการต่อ LCD เข้ากับไมโครคอนโทรลเลอร์ 17

รูปที่ 4.1 แสดงลักษณะการต่อวงจรผ่านไปยังเครื่องใช้ไฟฟ้า 19

รูปที่ 4.2 (a) วงจรลดขนาดสัญญาณ โดยให้ Vout เป็นแรงดันจากไฟบ้าน (b) วงจรลดขนาดสัญญาณ โดยให้ Vout เป็นแรงดันที่คร่อมตัวต้านทานขนาด 100 โอห์ม 20

รูปที่ 4.3 แสดงการต่อ ADC0804 เข้ากับไมโครคอนโทรลเลอร์ 20

รูปที่ 4.4 แสดงการต่อ LCD เข้ากับไมโครคอนโทรลเลอร์ 21

รูปที่ 4.5 วงจรรวม 22

รูปที่ 5.1 กราฟแสดงการเปรียบเทียบค่า Pmeasure กับ Preal โดยกำหนด Vin2 = 2 โวลท์ 61

รูปที่ 5.2 กราฟแสดงการเปรียบเทียบค่า Pmeasure กับ Preal โดยกำหนด Vin2 = 4 โวลท์ 62

รูปที่ 5.3 กราฟแสดงการเปรียบเทียบค่า Pmeasure กับ Preal โดยกำหนด Vin2 = 6 โวลท์ 63

รูปที่ 5.4 กราฟแสดงการเปรียบเทียบค่า Pmeasure กับ Preal โดยกำหนด Vin2 = 8 โวลท์ 64

รูปที่ 5.5 กราฟแสดงการเปรียบเทียบค่า Pmeasure กับ Preal โดยกำหนด Vin2 = 10 โวลท์ 65

รูปที่ 5.6 กราฟแสดงการเปรียบเทียบค่า Pmeasure กับ Preal โดยกำหนด Vin2 = 12 โวลท์ 66

รูปที่ 5.7 กราฟแสดงการเปรียบเทียบค่า Pmeasure กับ Preal โดยกำหนด Vin2 = 14 โวลท์ 67

รูปที่ 5.8 กราฟแสดงการเปรียบเทียบค่า Pmeasure กับ Preal โดยกำหนด Vin2 = 16 โวลท์ 68

รูปที่ 5.9 กราฟแสดงการเปรียบเทียบค่า Pmeasure กับ Preal โดยกำหนด Vin2 = 18 โวลท์ 69

รูปที่ 5.10 กราฟแสดงการเปรียบเทียบค่า Pmeasure กับ Preal โดยกำหนด Vin2 = 20 โวลท์ 70

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

		หน้า
ตารางที่ 5.1	แสดงค่า Pmeasure เปรียบเทียบกับ Preal โดยกำหนด Vin2 = 2 โวลท์	61
ตารางที่ 5.2	แสดงค่า Pmeasure เปรียบเทียบกับ Preal โดยกำหนด Vin2 = 4 โวลท์	62
ตารางที่ 5.3	แสดงค่า Pmeasure เปรียบเทียบกับ Preal โดยกำหนด Vin2 = 6 โวลท์	63
ตารางที่ 5.4	แสดงค่า Pmeasure เปรียบเทียบกับ Preal โดยกำหนด Vin2 = 8 โวลท์	64
ตารางที่ 5.5	แสดงค่า Pmeasure เปรียบเทียบกับ Preal โดยกำหนด Vin2 = 10 โวลท์	65
ตารางที่ 5.6	แสดงค่า Pmeasure เปรียบเทียบกับ Preal โดยกำหนด Vin2 = 12 โวลท์	66
ตารางที่ 5.7	แสดงค่า Pmeasure เปรียบเทียบกับ Preal โดยกำหนด Vin2 = 14 โวลท์	67
ตารางที่ 5.8	แสดงค่า Pmeasure เปรียบเทียบกับ Preal โดยกำหนด Vin2 = 16 โวลท์	68
ตารางที่ 5.9	แสดงค่า Pmeasure เปรียบเทียบกับ Preal โดยกำหนด Vin2 = 18 โวลท์	69
ตารางที่ 5.10	แสดงค่า Pmeasure เปรียบเทียบกับ Preal โดยกำหนด Vin2 = 20 โวลท์	70

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความสำคัญและที่มา

ในชีวิตประจำวันของทุกๆคน เราคงจะปฏิเสธไม่ได้ว่า เครื่องใช้ไฟฟ้ามีส่วนเกี่ยวข้องกับกิจกรรมที่เราทำเกือบทุกเรื่อง ซึ่งในบ้านพักอาศัยที่เราอยู่นี้จะเห็นว่า มีเครื่องใช้ไฟฟ้าอยู่มากมาย ในการทำงานของเครื่องใช้ไฟฟ้าเหล่านี้ จำเป็นต้องใช้กำลังไฟฟ้าทั้งสิ้น ซึ่งนั่นหมายถึงค่าไฟฟ้าที่เราจะต้องจ่ายด้วย ดังนั้นการที่เราสามารถทราบได้ว่าอุปกรณ์ไฟฟ้าแต่ละตัวต้องใช้กำลังไฟฟ้าเท่าใดจะเป็นประโยชน์ต่อเราอย่างยิ่ง

1.2 วัตถุประสงค์ของโครงงาน

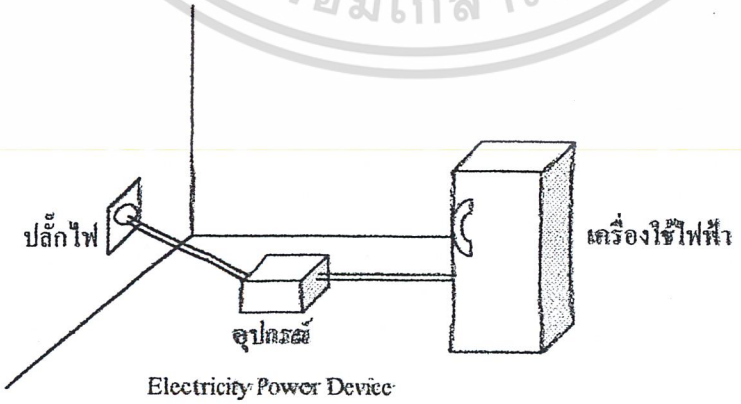
1.2.1 เพื่อสร้างแนวคิดในการออกแบบอุปกรณ์ที่มีประโยชน์ต่อทุกๆคนทั่วไป

1.2.2 สามารถออกแบบและสร้างเครื่องมือที่มีประโยชน์ต่อทุกๆคนทั่วไป

1.2.3 ศึกษา, ออกแบบและสร้างอุปกรณ์ที่นำไปใช้ประโยชน์ ในการตรวจสอบวัดกำลังไฟฟ้าได้

1.3 แนวคิดในการออกแบบ

แนวคิดในการสร้างโครงงานนี้ขึ้นมา ก็เพื่อที่จะสร้างอุปกรณ์ที่สามารถคำนวณกำลังไฟฟ้า ซึ่งเกี่ยวข้องกับชีวิตประจำวันของเราได้ โดยตัวโครงงานจะเป็นอุปกรณ์ที่เป็นตัวกลางในการส่งผ่านกำลังไฟฟ้าไปยังตัวเครื่องใช้ไฟฟ้า เสมือนเป็นมิเตอร์โดยที่ในตัวอุปกรณ์สามารถที่จะคำนวณกำลังไฟฟ้าและค่าไฟฟ้าออกมาเป็นตัวเลขได้เลย จะมีรูปแบบตามรูปที่ 1.1

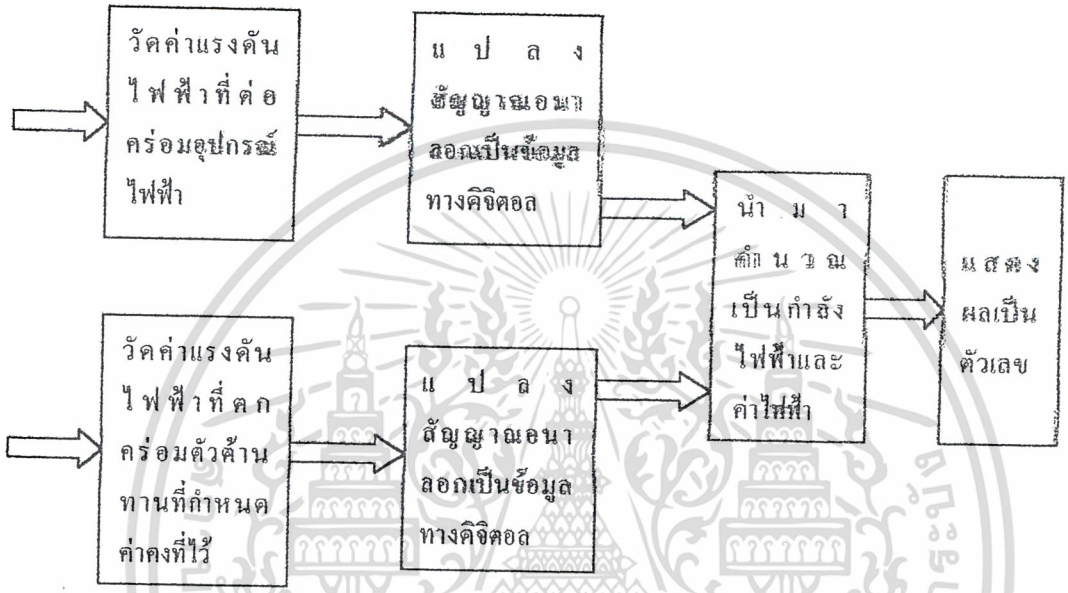


รูปที่ 1.1 แสดงลักษณะการเชื่อมต่อของอุปกรณ์ Electricity Power Device

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 หลักการทำงานเบื้องต้น

ตัวอุปกรณ์นี้ สามารถนำไปใช้ที่ใดก็ได้ เนื่องจากพื้นฐานการทำงาน จะทำการวัดค่าของแรงดันและกระแส นำมาคำนวณเป็นกำลังไฟฟ้า โดยรูปแบบการทำงานจะมีลักษณะดังรูปที่ 1.2



รูปที่ 1.2 Block diagram ของการทำงานใน Electricity Cost Device

จาก Block diagram สามารถอธิบายการทำงานได้โดย ส่วนแรกจะเป็นการรับอินพุทเข้ามา ซึ่งอินพุทนี้มี 2 ค่า ค่าแรกจะเป็นแรงดันที่วัดคร่อมเครื่องใช้ไฟฟ้า โดยจะทำการลดขนาดสัญญาณโดยอัตราส่วน 80 : 1 ทำให้สัญญาณจะมีขนาดลดลงมาอยู่ในช่วง 0-5 โวลท์ ส่วนค่าที่สองจะเป็นแรงดันที่คร่อมตัวต้านทานขนาด 1 กิโลโอห์ม ซึ่งถูกลดขนาดเป็น 80 : 1 เช่นกัน

ส่วนที่สองจะเป็นส่วนของการแปลงค่าจากสัญญาณที่เข้ามา ซึ่งเป็นสัญญาณอนาลอกนั้น จะทำการแปลงค่าเป็นข้อมูลทางดิจิตอล

ส่วนที่สามจะนำข้อมูลที่ได้ทั้งสองชุด มาคำนวณเป็นค่าแรงดันที่ตกคร่อมเครื่องใช้ไฟฟ้า และข้อมูลชุดที่สองจะนำมาคำนวณเป็นกระแส ที่ผ่านเข้าไปในเครื่องใช้ไฟฟ้า จากนั้นจะนำมาคำนวณเป็นค่ากำลังไฟฟ้าโดย

กำลังไฟฟ้า = แรงดัน X กระแส

และส่วนสุดท้าย คือส่วนของการแสดงผลโดยจะนำค่าไฟฟ้าที่ได้ มาแสดงผลบนหน้าจอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีที่ใช้ในการคำนวณ

2.1 กำลังงานไฟฟ้า

นิยามคือ ค่าของพลังงานที่ใช้ในช่วงเวลาหนึ่งสูตรคือ

$$p = \frac{dW}{dt} = VI$$

โดยที่

p	คือ กำลังงาน หน่วยเป็น Watt
W	คือ พลังงาน หน่วยเป็น Joule
t	คือ เวลา หน่วยเป็น วินาที
V	คือ ค่าแรงดันไฟฟ้า
I	คือ ค่ากระแสไฟฟ้า

2.2 Watt-hour power

คือค่า power ที่คูณกับเวลาโดยใช้หน่วยเป็นชั่วโมง ในประเทศไทยใช้ หน่วยนี้ในการคิดคำนวณค่า ไฟฟ้าซึ่งใช้เป็น kW-hour (kWh)

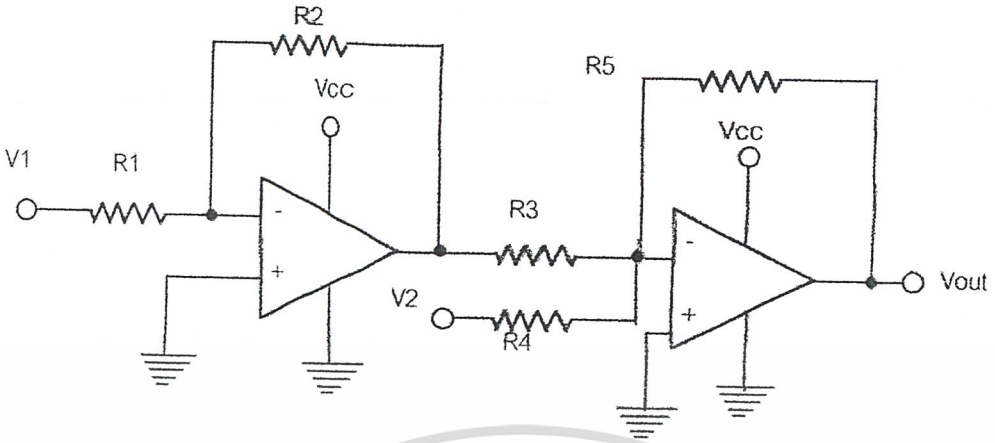
$$P_{Wat\text{-}hour} = P_{avg} \times t_{hour}$$

2.3 Average Power

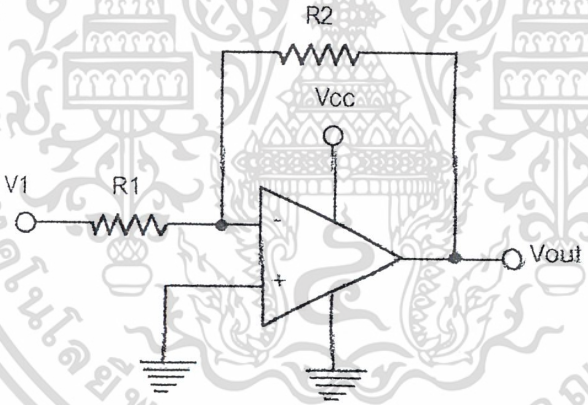
คือ ค่าเฉลี่ยของกำลังงานโดยคิดจากสูตร

$$P_{avg} = \frac{1}{T}\int_{t+T}^{t} P\,dt$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(a)



(b)

รูปที่ 2.1 (a) วงจรลดขนาดสัญญาณโดยให้ Vout เป็นแรงดันจากไฟบ้าน (b) วงจรลดขนาดสัญญาณโดยให้ Vout เป็นแรงดันที่คร่อมตัวต้านทานขนาด 100 โอห์ม

วงจรในรูป (a)

Vout = -R5 [(R2/R1R3)V1 + (V2/R4)]

วงจรในรูป (b)

Vout = -(R2/R1)V1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 นิยามของค่าต่างๆ

Watt-hour

วัตต์ต่อชั่วโมง(watt-hour) มีสัญลักษณ์เป็นWh เป็นหน่วยของพลังงานหนึ่งวัตต์ที่ใช้ใน 1 ชั่วโมง ค่าวัตต์ต่อชั่วโมงนี้ไม่ใช่หน่วยมาตรฐานในระบบที่เป็นทางการ แต่เรามักจะนำมาใช้ในการประยุกต์ทางไฟฟ้า

การหาค่าในหน่วยจูลเมื่อเราทราบค่าวัตต์ต่อชั่วโมง ให้นำค่าวัตต์ต่อชั่วโมงมาคูณกับ $3.6x10^{3}$ ส่วนการหาค่าในหน่วยวัตต์ต่อชั่วโมงเมื่อเราทราบค่าจูล ให้นำค่าจูลมาคูณกับ $2.778x10^{-4}$

โดยทั่วไป ค่าพลังงาน (E) จะมีค่าเท่ากับ พลังงาน (P) คูณกับเวลา (t) การกำหนดค่าพลังงานเป็นวัตต์ต่อชั่วโมง ต้องใช้ค่าพลังงานเป็นหน่วยวัตต์ และใช้เวลาเป็นหน่วยของชั่วโมง เช่น ถ้าเรามีค่ากำลังงานเท่ากับ 60 วัตต์ และเวลาเท่ากับ 3 ชั่วโมงจะได้

$E = P \times t = 60 \times 3 = 180$ Wh

ถ้า P และ t ไม่อยู่หน่วยวัตต์กับหน่วยชั่วโมง เราจะต้องทำการแปลงหน่วยก่อน

ค่าวัตต์ต่อชั่วโมงนี้ ยากที่จะใช้ในการกำหนดพลังงานในรูปอื่นทางไฟฟ้า ปริมาณของพลังงานศักย์ในแก๊สโซลีน น้ำมัน หรือถ่านหิน จะปลดปล่อยออกมาเมื่อเชื้อเพลิงได้รับการเผาไหม้ พลังงานความร้อนจะเป็นผลจากการเผาไหม้ของเชื้อเพลิง มีค่าเป็นหน่วยจูลตามระบบของ International System of Units (SI) หรือ British thermal Units (Btu)

พลังงาน

พลังงานมีหลายรูปแบบ ทั้งในการเคลื่อนที่ทางกล ความร้อน หรือพลังงานไฟฟ้าที่ถูกสร้างขึ้น แผ่ออกมา สูญเสียไป หรือได้รับล้วนเป็นพลังงานทั้งสิ้น สัญลักษณ์ทั่วไปของพลังงานแทนด้วย E ในระบบมาตรฐาน (The standard unit) จะมีหน่วยเป็นจูล ในบางระบบเช่น ระบบ Btu จะแทนด้วย 1 Btu = 1,055 J.

1 จูล คือ พลังงานที่เท่ากับ 1 วัตต์ของกำลังงานสำหรับ 1 วินาที พลังงานในวงจรที่เป็นกระแสตรง(DC) จะใช้ค่าของแรงดันในหน่วยโวลท์ กระแสในหน่วยแอมแปร์ และเวลาในหน่วยวินาที ส่วนในวงจรกระแสสลับ(AC) จะใช้ค่าพลังงานที่ซับซ้อนกว่า

สัญลักษณ์ทั่วไปของกระแสคือ I เมื่อแหล่งกำเนิดแรงดันที่เป็นDC ให้ค่ากระแสเป็นแอมแปร์สำหรับเวลาในหน่วยวินาที พลังงานนี้จะใช้เป็นหน่วยจูล คือ

$E = Vit$

เมื่อกระแสในหน่วยแอมแปร์วิ่งผ่านตัวความต้านทานในหน่วยโอห์มสำหรับเวลาในหน่วยวินาทีจะได้พลังงานสูญเสีย(Elost) เป็น

$E = I^2Rt$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อแรงดันDC ตกคร่อมตัวความต้านทานที่มีหน่วยเป็นโอห์ม สำหรับเวลาในหน่วยวินาที จะได้พลังงานดูดซับ(Eabsorb) เป็น

$E = V^2t/R$

วัตต์

วัตต์(watt) เป็นหน่วยมาตรฐานของกำลังงาน(พลังงานต่อหน่วยเวลา) และมีค่าเท่ากับ 1 จูลต่อวินาที ค่าวัตต์นี้ใช้ในการกำหนดอัตราของพลังงานไฟฟ้าที่สูญเสีย หรืออัตราพลังงานสนามแม่เหล็กที่แผ่ออกมา ดูดซับเข้าไป หรือสูญเสีย

ในวงจรและระบบที่เป็นDC และACที่ความถี่ต่ำ จะคำนวณค่ากำลังงานด้วยสูตรการคำนวณพื้นฐาน เป็นค่าคงที่ โดย

$P = EI$

$P = E^2/R$

$P = I^2R$

เมื่อ P คือ กำลังงานไฟฟ้า

E คือ ค่าแรงดัน

I คือ ค่ากระแส

ในระบบที่ใช้ความถี่คลื่นวิทยุหรือที่ความถี่สูงนั้น ค่ากำลังงานจะไม่ขึ้นกับค่า resistance เท่านั้น แต่จะขึ้นกับค่า reactance ด้วย จึงไม่สามารถใช่สูตรการคำนวณที่เป็นค่าคงที่ได้ จำเป็นต้องทำการวัดค่าโดยตรง

ในกรณีที่กำลังงานสูงมากหรือต่ำมาก อัตราการสูญเสียกำลังงานจะสูงขึ้น

กำลังงาน

กำลังงานทางไฟฟ้า คือ อัตราการเปลี่ยนแปลงของพลังงานไฟฟ้าไปเป็นพลังงานในรูปอื่น เช่น การเคลื่อนที่ ความร้อน หรือสนามแม่เหล็ก สัญลักษณ์ทั่วไปของกำลังงานแทนด้วย P มีหน่วยเป็นวัตต์ แทนด้วย W ในวงจรส่วนใหญ่จะใช้เป็นกิโลวัตต์ (kW) โดย 1 kW = 1,000 W 1 วัตต์เป็นผลของกำลังงาน จากพลังงานที่สูญเสีย หรือเปลี่ยนแปลงเท่ากับ 1 จูลต่อวินาที ค่ากำลังงานนี้บางครั้งเราจะเรียกว่า wattage ค่า wattage ในDC จะเป็นค่าจากแรงดันในหน่วยโวลท์ และค่ากระแสในหน่วยแอมแปร์ นี้เป็นกฎโดยทั่วไปว่าที่ความถี่ต่ำจะไม่มีการเก็บสะสมหรือปลดปล่อยพลังงาน แต่ที่ความถี่สูงจะมีการเก็บสะสมหรือปลดปล่อยพลังงาน ค่ากำลังงานจึงมีการคำนวณที่ซับซ้อนกว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในDC แหล่งจ่ายแรงดัน E ให้กระแส I ออกมาเป็นหน่วยแอมแปร์ จะให้ค่ากำลังงานออกมาเป็น

$P = EI$

เมื่อกระแสในหน่วยแอมแปร์วิ่งผ่านความต้านทานในหน่วยโอห์ม กำลังงานในหน่วยวัตต์ที่จ่ายออกมาจะมีค่าเป็น

$P = I^2R$

เมื่อแรงดัน(E) ในหน่วยโวลท์ตกคร่อมความต้านทานในหน่วยโอห์ม กำลังงานที่สูญเสียหรือเปลี่ยนแปลงจะเป็น

$P = E^2/R$

ในวงจรDC ค่ากำลังงานจะเป็นปริมาณสเกลาร์ ส่วนในกรณีของวงจรAC ค่ากำลังงานจะเป็นปริมาณเวกเตอร์ ถ้าหากเราสมมติว่าไม่มีค่า reactance เราจะใช้สูตรคำนวณเป็นDC ได้เลย แต่ถ้าใช้AC เราต้องคำนวณโดยใช้การคำนวณที่เป็นค่า root-mean-square ของค่าทางกระแสสลับ

เวลา

เวลาเป็นปรากฏการณ์ที่มีความหมายการเปลี่ยนแปลงของสภาพแวดล้อมในจักรวาลตามความหมายที่แท้จริงนั้นยากที่จะอธิบาย บางครั้งเราเรียกมันว่าภาพลวงตา หรือเหตุการณ์ที่เกิดขึ้นในสถานที่เดียวกัน

หน่วยพื้นฐานของเวลาในระบบ SI คือ วินาที สัญลักษณ์คือ s หรือ sec. โดย 1 วินาทีจะมีค่าเท่ากับ 9,192,631,770 ของรอบการแผ่รังสีของซีเซียม 133 อะตอม

จูล

จูลเป็นหน่วยพลังงานมาตรฐาน ในวิชาอิเล็กทรอนิกส์และในวิชาวิทยาศาสตร์ประยุกต์ทั่วๆไป 1จูลคือ พลังงานที่ปล่อยออกมาเมื่อแรง 1 นิวตัน ทำให้เกิดการเคลื่อนที่ 1 เมตร 1 จูลเสมือนกับ 1 วัตต์ของกำลังงานที่แผ่หรือสูญเสียใน 1 วินาที

ในวิทยาศาสตร์ปนะยุกต์บางอย่าง หน่วยอุณหภูมิของระบบอังกฤษ(Btu) ใช้แสดงเป็นหน่วยพลังงาน 1Btu จะมีค่าประมาณ 1,055 จูล

แรงดัน

แรงดัน(Voltage) มีชื่อเรียกอีกอย่างหนึ่งว่า electromotive force (EM) มีความหมายได้ทั้งเป็นพลังงานศักย์ของประจุไฟฟ้าและพลังงานศักย์ตกคร่อม ถ้าตัวนำหรือสารกึ่งตัวนำที่อยู่ระหว่าง 2 จุดซึ่งมีศักย์แตกต่างกัน จะมีกระแสไฟฟ้าไหลผ่าน สัญลักษณ์ที่เหมือนกันของแรงดันคือ V หรือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

E หน่วยมาตรฐานคือโวลท์(V) 1 โวลท์คือแรงดันที่ทำให้ประจุ 1 คูลอมบ์เคลื่อนที่ผ่านใน 1 วินาที

แรงดันนี้เป็นได้ทั้งไฟตรงหรือไฟสลับ แรงดันไฟตรงจะมีทิศทางเดียวตลอดเวลา แต่ในแรงดันไฟสลับนั้นจะมีการเปลี่ยนแปลงทิศทางไปตามคาบของเวลา จำนวนรอบที่สมบูรณ์ต่อเวลาซึ่งเป็นวินาทีคือความถี่ซึ่งวัดเป็นเฮิรทซ์ เอาท์พุทของเพาเวอร์ซัพพลายที่กรองแล้วจะเป็นตัวอย่างของไฟตรง

ความต่างศักย์จะสร้างสนามไฟฟ้าสถิตย์ก็ต่อเมื่อไม่มีกระแสไหลผ่าน เพราะว่าแรงดันเพิ่มขึ้นระหว่าง 2 จุดจะถูกแบ่งโดยระยะทางที่จำเพาะ สนามไฟฟ้าสถิตย์เข้มข้นมากกว่าเดิม จกกการเพิ่มขึ้นของความต่างศักย์ระหว่าง 2 จุด ความหนาแน่นของฟลักซ์ไฟฟ้าสถิตย์จะลดลงในช่วงที่วัตถุมีประจุเดียวล้อมรอบ โดยสนามไฟฟ้าสถิตย์ที่ความจุเป็นสัดส่วนโดยตรงของแรงดันของวัตถุต่อวัตถุที่อยู่ข้างเคียง

กระแสไฟฟ้า

กระแสไฟฟ้าคือการไหลของประจุไฟฟ้า โดยปกติแล้วจะเป็นอิเล็กตรอนหรืออะตอมที่สูญเสียอิเล็กตรอน ใช้สัญลักษณ์แทนด้วย I หน่วยมาตรฐานคือแอมแปร์ใช้สัญลักษณ์ A 1แอแปร์ของกระแสไฟฟ้าคือการเคลื่อนที่ของประจุไฟฟ้าขนาด 1 คูลอมบ์ ($6.24x10^{18}$ ประจุ) ผ่านจุดๆหนึ่งใน 1 วินาที นักฟิสิกส์กำหนดว่ากรแสไฟฟ้าไหลจากจุดประจุบวกไปยังจุดประจุลบ เรียกว่า Conventional Current หรือ กระแสโดยปกติที่ตัวเคลื่อนที่คืออิเล็กตรอน ทำให้การเคลื่อนที่เป็นการเคลื่อนที่จากลบไปบวก

กระแสไฟฟ้าสามารถเป็นได้ทั้งกระแสตรงหรือกระแสสลับ กระแสตรงไหลผ่านจุดทุดจุดในทิศทางเดิมตลอดเวลา ถึงแม้ว่าขนาดของกระแสจะเปลี่ยนแปลงก็ตาม ในกระแสสลับการไหลของประจุจะไหลกลับทิศทางเป็นช่วงๆ จำนวนรอบของกระแสสลับที่สมบูรณ์ต่อวินาทีคือความถี่ใช้ในหน่วยเฮิรทซ์ ตัวอย่างของไฟตรงคือการผลิตกระแสไฟฟ้าในกระบวนการทางไฟฟ้าเคมีของเซลล์

กระแสต่อพื้นที่ตัดขวางคือความหนาแน่นกระแส มีหน่วยเป็นแอมแปร์ต่อพื้นที่ โดยทั่วไปกระแสขนาดใหญ่ในตัวนำจะมีค่าสูงกว่าความหนาแน่นกระแส อย่างไรก็ตามบางครั้งความหนาแน่นกระแสจะแปรตามส่วนที่แตกต่างกันทางไฟฟ้า

กระแสในทางไฟฟ้าจะสร้างสนามแม่เหล็กขึ้น กระแสยิ่งสูงสนามแม่เหล็กจะยิ่งเข้มข้น นี่เป็นพื้นฐานของระบบไร้สาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
ไมโครคอนโทรลเลอร์และอุปกรณ์

3.1 ไมโครคอนโทรลเลอร์

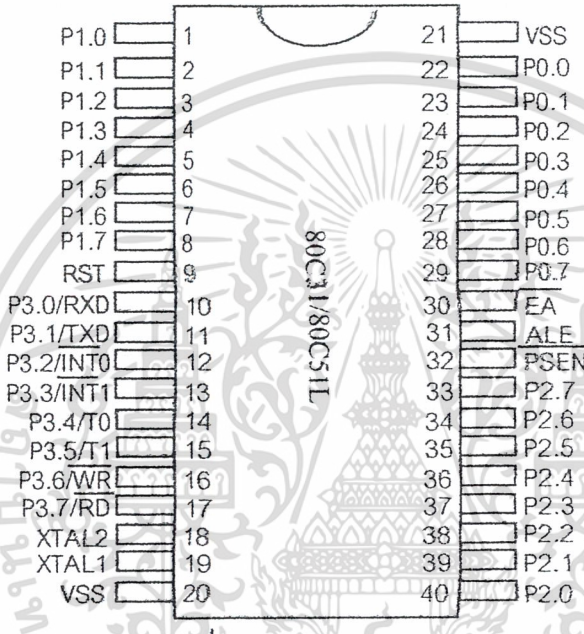
คุณสมบัติของ MCS-51

คุณสมบัติที่สำคัญๆ ของชิปไมโครคอนโทรลเลอร์ตระกูล MCS-51 มีดังนี้

1. ต้องการแหล่งจ่ายไฟ 5 โวลท์ เพียงชุดเดียว
2. มีหน่วยความจำสำหรับเก็บโปรแกรมควบคุมการทำงานอยู่ภายในชิปจำนวน 4 กิโลไบท์เบอร์ 8031,8032 ไม่มีหน่วยความจำส่วนนี้ ส่วนเบอร์ 8052 มี ความจำส่วนนี้ 8 กิโลไบท์
3. มีหน่วยความจำสำหรับเก็บข้อมูลทั่วไป (RAM) อยู่ภายในชิปจำนวน 128 ไบต์ (ใน 8031,8051)หรือ 256 ไบต์
4. สามารถใช้หน่วยความจำสำหรับโปรแกรมและข้อมูลที่อยู่ภายนอกชิปได้อย่างละ 64 ไบต์แยกจากกัน
5. คำสั่งส่วนใหญ่ใช้เวลาทำงานเพียง 1 ไมโครวินาที เมื่อใช้คริสตอลความถี่ 12 เมกกะเฮิรต์
6. พอร์ตสามารถรับหรือส่งข้อมูลได้ 2 ทิศทาง จำนวน 4 พอร์ตๆ ละ 8 บิต หรือสามารถใช้งานรวมทั้งสิ้น 32 พอร์ต
7. รับและส่งข้อมูลแบบอนุกรมได้ในตัว โดยสามารถกำหนดอัตราเร็วในการรับและส่งข้อมูล (Band Rate) ได้ตั้งแต่ 300 ถึง 375 กิโลไบต์ต่อวินาที
8. จัดลำดับความสำคัญของสัญญาณอินเตอร์รัปต์ได้ 2 ระดับ
9. มีรีจิสเตอร์สำหรับใช้งานเป็นไทม์เมอร์หรือเคาน์เตอร์เพื่อนับจำนวนสัญญาณนาฬิกาภายในชิป หรือนับการเปลี่ยนสถานะของสัญญาณภายนอกขนาด 16 บิต จำนวน 2 ตัว เพื่อใช้สำหรับนับจำนวนพัลส์ วัดความกว้างของพัลส์หรือใช้วัดช่วงเวลา (ในเบอร์ 8052 จะมี 3 ตัว)
10. หน่วยความจำสำหรับเก็บข้อมูลภายในส่วนสามารถเข้าถึงข้อมูลได้ทั้งระดับไบต์และระดับบิต เพื่อให้การออกแบบโปรแกรมและควบคุมระบบทำได้ง่ายขึ้น
11. มีคำสั่งคูณและหารเลขขนาด 8 บิต ในตัวเอง
12. สามารถประมวลผลแบบบูลีนเพื่อใช้ในงานควบคุมโดยเฉพาะ
13. ใช้โปรแกรมของไมโครคอนโทรลเลอร์ตระกูล MCS-48 (Upwardly Compatible) ได้ ในโครงงานครั้งนี้จะใช้เบอร์ 8031 ซึ่งเบอร์นี้จะไม่มีหน่วยความจำสำหรับเก็บโปรแกรมภายในชนี้มีไว้ใช้ในกรณีที่โปรแกรมมีขนาดเล็กกว่า 4 กิโลไบต์ หรือมากกว่า 4 กิโลไบต์มาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับเบอร์ 8051 และ 8751 เราสามารถทำให้มีคุณสมบัติเหมือน 8031 ได้โดยการต่อขา EA ลงกราวด์ คือให้ใช้โปรแกรมจากหน่วยความจำภายนอกเพียงอย่างเดียว



รูปที่ 3.2 การจัดวางขาของ

พอร์ทของ 8051 มี 40 ขา ดังนี้

- Vcc (ขา40) ต่อกับ +5V
- Vss (ขา20) เป็นขา GND
- พอร์ท 0 (ขา 32-39) มีทั้งหมด 8 บิต คือ (P0.7-P0.0) ใช้งานได้ 2หน้าที่ คือ แอดเดรสบัส และดาต้าบัสเมื่อต้องการติดต่อกับหน่วยความจำภายนอกหรือเป็นไอโอพอร์ท ถ้าต้องการให้ทำงานเป็นอินพุทพอร์ทต้องส่งลอจิก "1" ไปยังพอร์ทนี้ จะมีผลให้ Q ของ D-FF เป็น "0" ทำให้ FET ตัวล่างมีสถานะ OFF สัญญาณที่ใช้อ่านอินพุทพอร์ทแลทช์โดยส่งสัญญาณ READ LATCH ไปกระตุ้นที่ Tri-State Buffer ตัวบนและการอ่าน Port (pin) จะใช้สัญญาณ Read (pin)มีโครงสร้างแบบ Open-Drain ,Bi-directional

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- พอร์ท 1 (ขา 1-8) มีทั้งหมด 8 บิต คือ (P1.0-P1.7) มีโครงสร้างคล้าย พอร์ท 0 แต่จะใช้ความต้านทานภายในพูลอัพแทน Internal Pull up Register
- พอร์ท 2 (ขา 21-28) มีทั้งหมด 8 บิต คือ (P2.7-P2.0) มีโครงสร้างคล้าย พอร์ท 0 โดยมี FET ตัวล่างตัวเดียวส่วนด้านบนใช้ความต้านทานพูลอัพแทน (Internal pull up) พอร์ทนี้ทำงาน 2 หน้าที่ คือสามารถใช้เป็นแอดเดรสบัสขนาด 8 บิต (A15-A8) และเป็นไอโอพอร์ทใช้งานทั่วไปเมื่อจะใช้งานเป็นอินพุทพอร์ทต้องส่งลอจิก "1" มาที่พอร์ทนี้ก่อนเพื่อบังคับให้ FET อยู่ในสภาวะ OFF
- พอร์ท 3 (ขา 10-17) มีทั้งหมด 8 บิต คือ ขา (P3.7-P3.0) มีโครงสร้างคล้ายพอร์ท 1 ทำงานได้ 2 หน้าที่คือเป็นไอโอพอร์ทถ้าจะโปรแกรมให้เป็นอินพุทพอร์ทต้องส่งลอจิก "1" มาที่พอร์ทนี้ก่อน และอีกหน้าที่หนึ่งคือใช้ส่งสัญญาณควบคุมออกมา และรับสัญญาณเข้าไปมีสัญญาณต่างๆ ดังนี้

P3.0/RXD (Serial Input Port) เป็นขาที่ใช้รับข้อมูลแบบอนุกรม (UART)

P3.1/TXD (Serial Output Port) เป็นขาที่ใช้ส่งข้อมูลแบบอนุกรม (UART)

P3.2/INT0 (External Interrupt 0) ใช้รับสัญญาณการขัดจังหวะจากภายนอกเบอร์ 0

P3.3/INT1 (External Interrupt 1) ใช้รับสัญญาณการขัดจังหวะจากภายนอกเบอร์ 1

P3.4/T0 (Counter 0 External Input) ขารับสัญญาณพัลซ์อินพุทเข้าไปยังวงจร Counter 0 (เป็นอินพุทโหมดเคาน์เตอร์)

P3.5/T1 (Counter 1 External Input) ขารับสัญญาณพัลซ์อินพุทเข้าไปยังวงจร Counter 0 (เป็นอินพุทโหมดเคาน์เตอร์)

P3.6/$\overline{\text{WR}}$ (External Data Memory Write Strobe) ขาสัญญาณควบคุมการเขียนข้อมูลลงหน่วยความจำข้อมูลภายนอก

P3.7/$\overline{\text{RD}}$ (External Data Memory Read Strobe) ขาสัญญาณควบคุมการอ่านข้อมูลจากหน่วยความจำข้อมูลภายนอก

-ALE (ขา 30) เป็นขาส่งสโตรบสำหรับใช้ในการแลตช์แอดเดรสไบต์ต่ำ (A7-A0)ที่ส่งออกมาจาก (พอร์ท 0) สัญญาณนี้จะแอคทีฟทุกๆ 2 ครั้ง ใน 1 แมชชีนไซเคิล

-$\overline{\text{PSEN}}$ (ขา 29) เป็นขาสโตรบที่ใช้สำหรับอ่านข้อมูลจาก Program Memory ภายนอก สัญญาณนี้จะส่งออกมา 2 ครั้งในแต่ละแมชชีนไซเคิลแต่ถ้าเป็นการอ่าน Internal Memory จะไม่มีสัญญาณออกที่ขานี้

-$\overline{\text{EA}}$ (ขา 31) ใช้เลือกหน่วยความจำโปรแกรมภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ป้อน "0" จะอ่านโปรแกรมจากภายนอกชิพ

ป้อน "1" จะอ่านโปรแกรมจากภายในชิพ

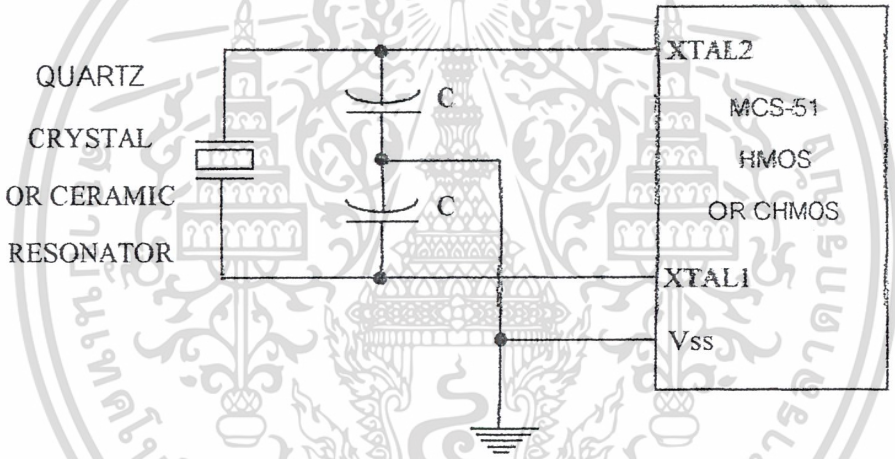
-RST (ขา 9) ขารีเซ็ท จะรีเซ็ทได้ก็ต่อเมื่อป้อนลอจิก "1" เข้าที่ขานี้อย่างน้อย 2 แมชชีนไซเคิล

-XTAL1 (ขา 19) ใช้ต่อคริสตอลภายนอกโดยเป็นอินพุทเข้าสู่วงจรออสซิลเลเตอร์ภายใน

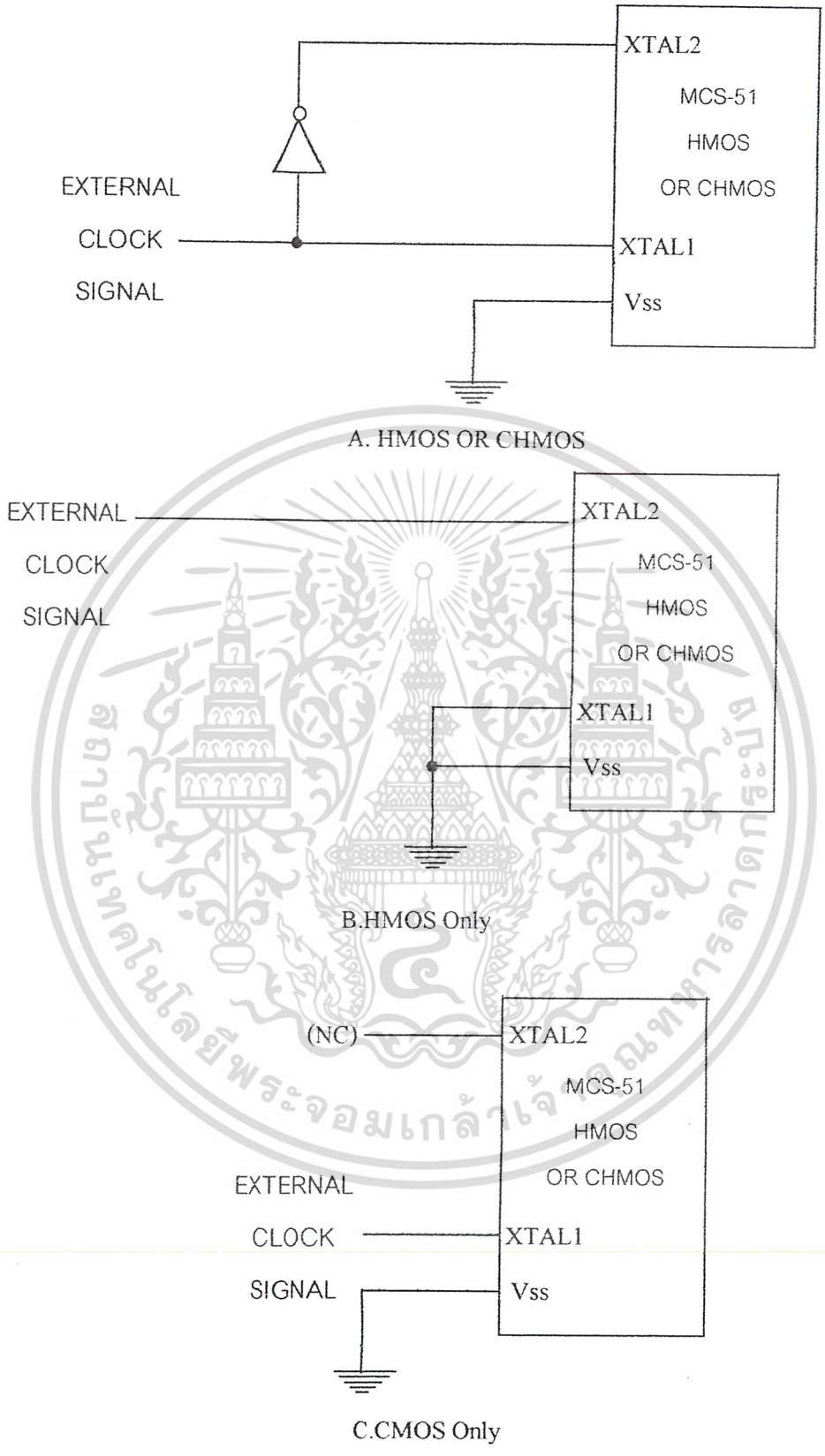
-XTAL2 (ขา 18) ใช้ต่อคริสตอลภายนอกโดยเป็นเอาท์พุทเข้าสู่วงจรออสซิลเลเตอร์ภายใน

วงจรClockของMCS-51

มีการต่ออยู่ 2 รูปแบบ คือแบบใช้คล็อกภายในและคล็อกจากภายนอกมีรูปแบบการต่อดังรูปที่ 2.3



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(b) Using the External Clock

รูปที่ 3.3 วงจรสร้างClockของ 8051

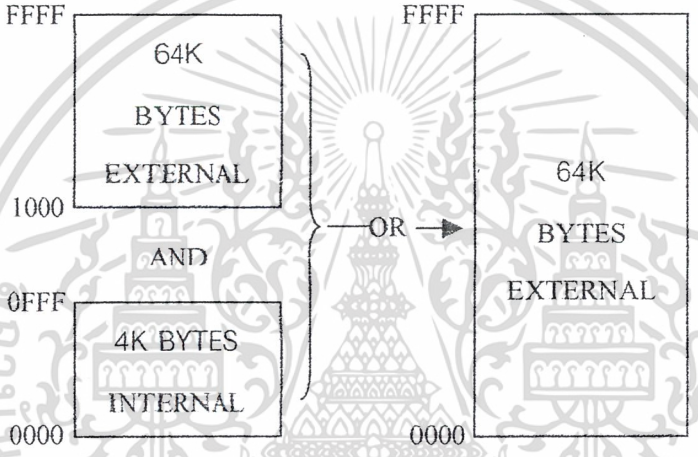
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน่วยความจำที่ใช้กับ MCS-51 มีอยู่ด้วยกัน 2 ชนิด คือ

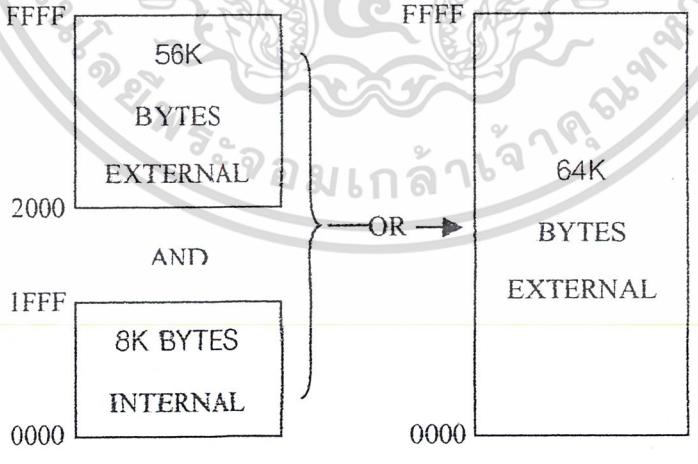
-หน่วยความจำสำหรับเก็บโปรแกรม (Program Memory)

-หน่วยความจำสำหรับเก็บข้อมูล (Data Memory)

หน่วยความจำสำหรับเก็บโปรแกรม เป็นหน่วยความจำที่ใช้เก็บโปรแกรมสั่งงานบรรจุอยู่ในชิพ8051 ส่วนที่เป็น Program Memory ก็คือ ROM ขนาด 4 กิโลไบต์นั่นเอง แต่ถ้าเป็นเบอร์ 8052 จะมี ROM ขนาด 8 กิโลไบต์ ดังรูปที่ 3.4 และ 3.5



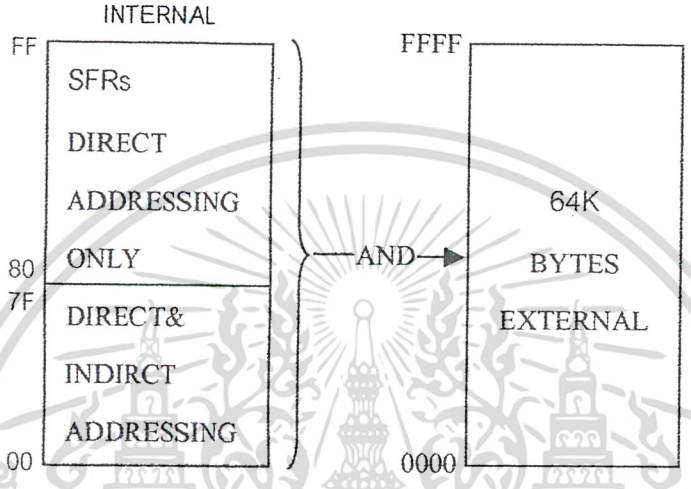
รูปที่ 3.4 ผังหน่วยความจำสำหรับเก็บโปรแกรมสำหรับเบอร์



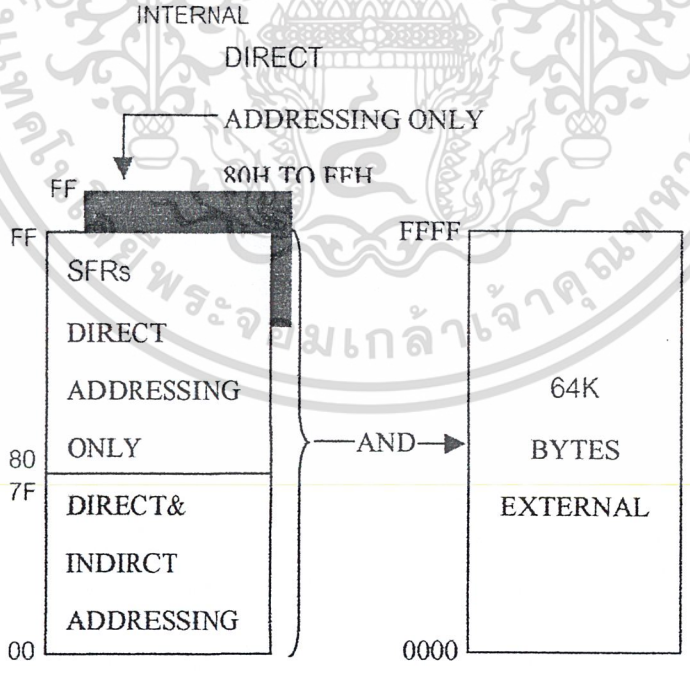
รูปที่ 3.5 ผังหน่วยความจำสำหรับเก็บโปรแกรมสำหรับเบอร์ 8052

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน่วยความจำสำหรับเก็บData Memory (RAM) แบ่งเป็นส่วนคือ หน่วยความจำข้อมูลภายในชิพมีเพียง 128 ไบต์ สำหรับเบอร์ 8051 และ 256 ไบต์ สำหรับเบอร์ 8052 ขึ้นไปและหน่วยความจำข้อมูลภายนอกชิพมีความจุ 64 กิโลไบต์ ดังรูปที่ 3.6 และ 3.7



รูปที่ 3.6 ผังหน่วยความจำสำหรับเก็บ Data Memory เบอร์ 8051



รูปที่ 3.7 ผังหน่วยความจำสำหรับเก็บ Data Memory เบอร์ 8052

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

* ชิพเบอร์ 8052 จะมีพื้นที่บริเวณ 82h-FFh ซึ่งถ้าจะเขียนอ่านข้อมูล ณ บริเวณนี้จะเข้าถึงข้อมูลโดยทางอ้อมเท่านั้น ดังผังหน่วยความจำดังรูป 3.6 *

ไทม์เมอร์/เคาน์เตอร์

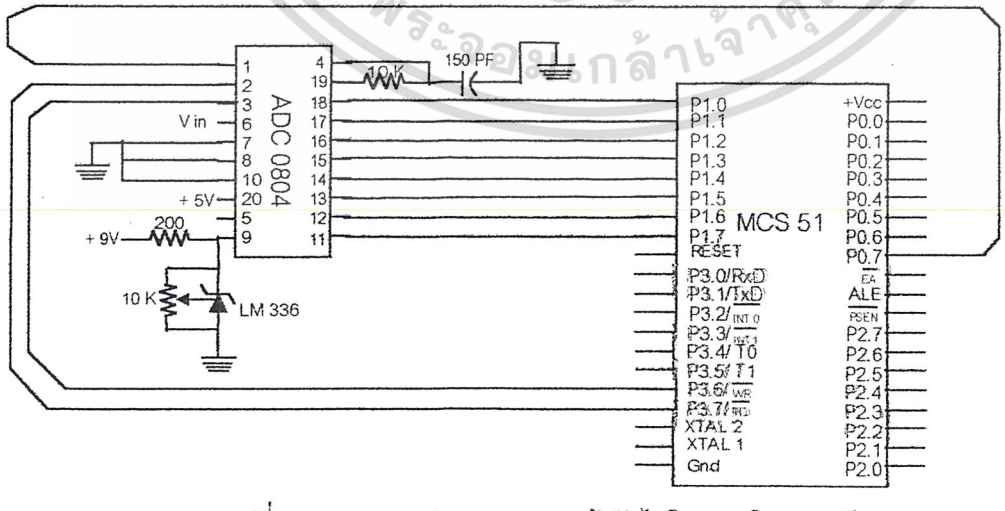
สามารถเลือกให้มีการทำงานเป็นไทม์เมอร์ หรือเคาน์เตอร์อย่างใดอย่างหนึ่ง โดยเลือกที่บิต C/T ในรีจิสเตอร์ใช้งานเฉพาะ TMOD โหมดไทม์เมอร์และเคาน์เตอร์จะใช้ up Counter Register (TH_X , TL_X) ตัวเดียวกันซึ่งเป็นแบบนับขึ้น

โหมดไทม์เมอร์ up Counter Register (TH_X , TL_X) จะถูกเพิ่มค่าทุกๆ 1 แมชชีนไซเคิล (12 คาบเวลาของ CPU osc) โหมดนี้ไม่ต้องป้อนสัญญาณจากภายนอกเข้ามาแต่จะใช้สัญญาณ CPU osc

โหมดเคาน์เตอร์ up Counter Register (TH_X , TL_X) จะถูกเพิ่มค่าทีละหนึ่งเมื่อป้อนสัญญาณคล็อกจากภายนอกเข้ามา 1 ลูกเข้ามาทางขา T0 (pin) หรือ T1 (pin) อยู่ที่ขา 14 และ 15 ตามลำดับ โดยไม่สนใจคาบเวลาของพัลซ์แต่ละลูก การตรวจสอบสัญญาณที่เข้าทางขานี้จะตรวจสอบทุกๆ S5P2 ของแต่ละแมชชีนไซเคิล ดังนั้นการตรวจสอบคล็อก 1 ลูกจะต้องใช้ถึง 2 แมชชีนไซเคิล (1/24 คาบเวลาของ CPU osc)

3.2 Analog to Digital Converter (ADC)

เป็น single chip ที่ใช้ในการแปลงสัญญาณ Analog ให้เป็นสัญญาณ Digital ซึ่งในกรณีนี้ใช้ IC เบอร์ ADC0804 ซึ่งเป็น chip ที่สามารถแปลง Voltage ได้ตั้งแต่ 0 – 5 V ไปเป็นสัญญาณ Digital 8 bit 00H – FFH และมี conversion time 100 μs รายละเอียดนอกจากนี้ดูได้ที่ภาคผนวก



รูปที่ 3.8 แสดงการต่อ ADC0804 เข้ากับไมโครคอนโทรลเลอร์

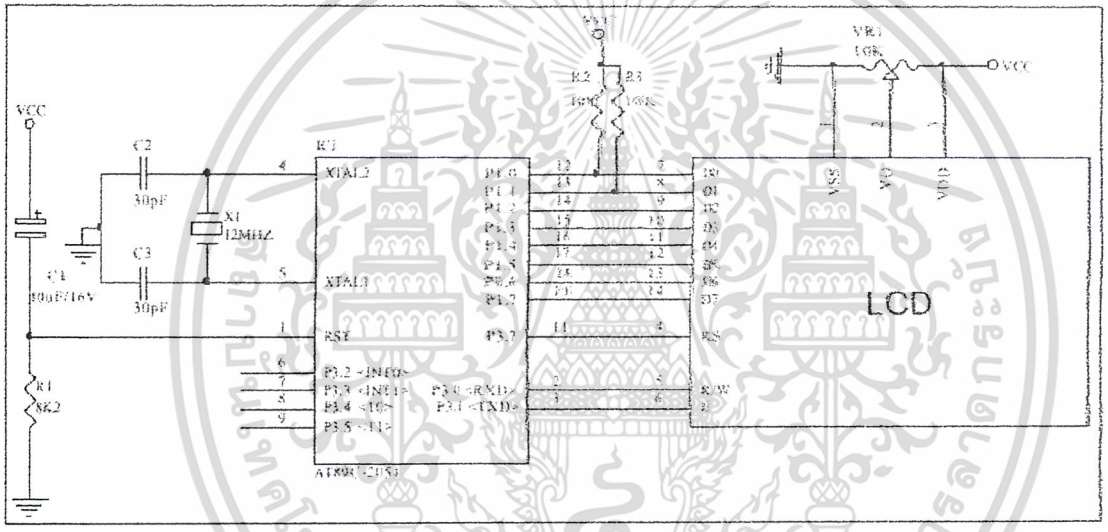
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

หลักการทำงานคือ นำ Vin เข้าที่ ขา 6 ของ ADC 0804 chip นี้จะทำการแปลงให้โดยอัตโนมัติ โดยเทียบค่าจาก Vref/2 ที่ขา 9 และ clk จาก ขา 4 และ 19 ซึ่งเป็นตัว oscillate โดยจะเริ่มจาก ส่งสัญญาณจากขา P0.7 ไปที่ขา 1 (Chip select) ถึงจะเริ่มการแปลงสัญญาณ โดยใช้เวลา 100 μs แล้วจึงนำข้อมูลเข้าสู่ port ที่ต้องการ ในกรณีนี้เราป้อนเข้าที่ port 2

3.3 LCD Display

ในกรณีนี้เราใช้ LCD ชนิด 1 line 16 character จะมีขา 14 ขาดังรูป



รูปที่ 3.9 แสดงการต่อ LCD เข้ากับไมโครคอนโทรลเลอร์

แบ่งเป็น

ขาไฟเลี้ยง 3 สาย

ขา 1	เป็น	Vss	(ground)
ขา 2	เป็น	Vo	(contrast)
ขา 3	เป็น	Vdd	(ไฟเลี้ยง)

control 3 สาย

ขา 4	เป็น	RS	(บอกว่าเป็นข้อมูลแบบโครงสร้างหรือเป็นข้อมูลที่จะแสดง)
ขา 5	เป็น	R/W	(บอกว่าจะ Read หรือจะ Write)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

46270

ขา 6 เป็น Enable

และ Data bus ขา 7-14 โดยที่ขา 14 จะใช้เป็น Busy flag ด้วย

ในกรณีนี้ใช้ Port 1 ต่อกับ Data bus และ ต่อขา Enable ที่ Port 3.7

การทำงานคือ เริ่มจาก Enable Display ก่อนแล้ว ทำการ initial หน้าจอให้พร้อมจริงเริ่มป้อนโปรแกรม

*เรื่องของโปรแกรมจะกล่าวถึงในภาคผนวก

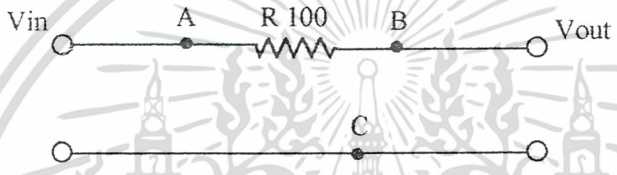
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

โครงสร้างของอุปกรณ์

4.1 ส่วนของอินพุท

เนื่องจากอุปกรณ์นี้ทำหน้าที่วัดกำลังไฟฟ้าภายในบ้าน โครงสร้างในส่วนอินพุทจะรับสัญญาณจากไฟฟ้าภายในบ้าน นำมาลดขนาดของสัญญาณลงเพื่อนำไปวัดค่า โดยจะรับสัญญาณเป็นค่าแรงดันซึ่งโดยทั่วไป ไฟฟ้าที่ใช้ตามบ้านจะมีค่าแรงดันอยู่ที่ประมาณ 220 Vrms โดยประมาณ เราจะนำสัญญาณนี้มาลดขนาดลง 80 เท่า ให้แรงดันนี้มีค่าอยู่ในช่วง 0-5 โวลท์ โดยจะมีแรงดัน 2 ค่า คือ แรงดันไฟฟ้าภายในบ้าน และแรงดันที่ตกคร่อมตัวต้านทานขนาด 100 โอห์มทางด้านอินพุทก่อนที่กระแสจะวิ่งเข้าไปในเครื่องใช้ไฟฟ้า ซึ่งแรงดันนี้จะนำมาคำนวณเป็นกระแสนั่นเอง



รูปที่ 4.1 แสดงลักษณะการต่อวงจรผ่านไปยังเครื่องใช้ไฟฟ้า

จากรูปที่ 4.1

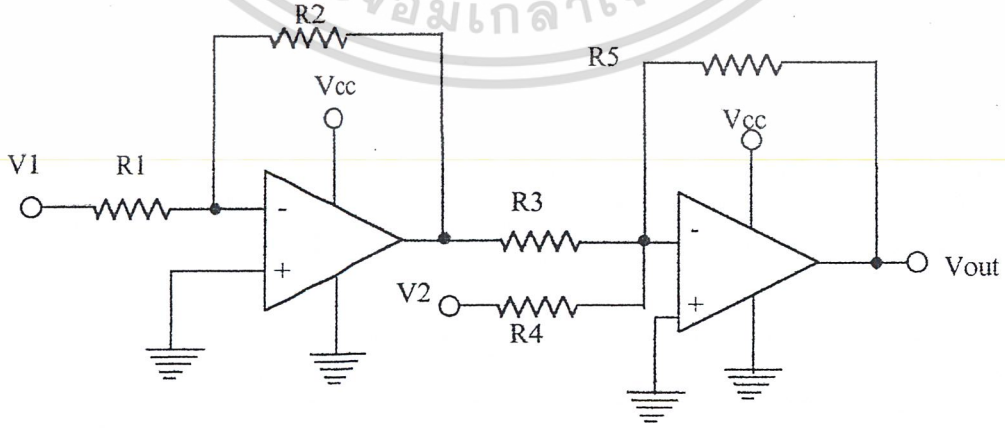
Vin คือ แรงดันจากไฟฟ้าตามบ้าน

Vout คือ แรงดันที่ต่อคร่อมเครื่องใช้ไฟฟ้า

จุด A คือ กราวนด์ของแรงดันอินพุทที่นำไปคำนวณ

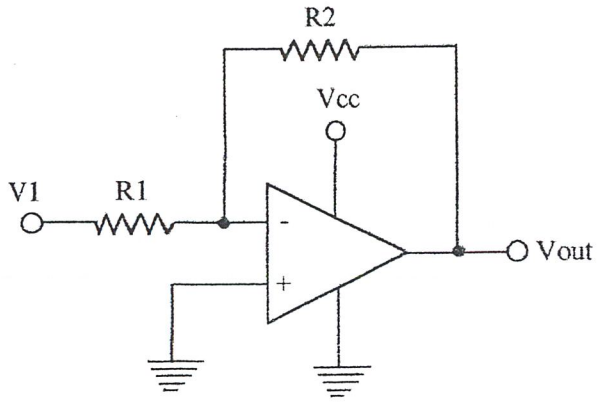
จุด B คือ แรงดันอินพุทที่ตกคร่อมความต้านทานขนาด 100 โอห์ม

จุด C คือ แรงดันอินพุทจากไฟฟ้าตามบ้าน



(a)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(b)

รูปที่ 4.2 (a) วงจรลดขนาดสัญญาณโดยให้ Vout เป็นแรงดันจากไฟบ้าน (b) วงจรลดขนาดสัญญาณโดยให้ Vout เป็นแรงดันที่คร่อมตัวต้านทานขนาด 100 โอห์ม

จากรูปที่ 4.2

V1 คือ แรงดันที่จุด C ในรูปที่ 4.1

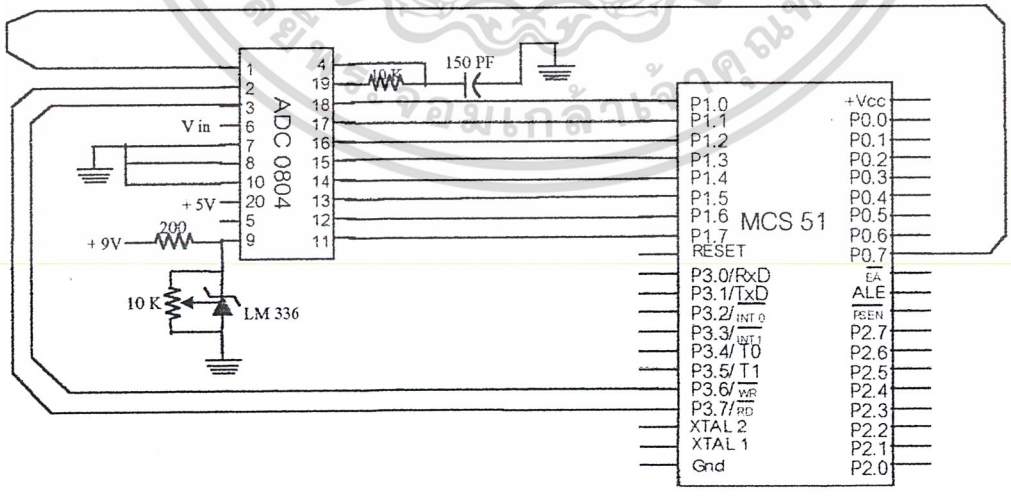
V2 คือ แรงดันที่จุด B ในรูปที่ 4.1

Ground คือ แรงดันที่จุด A ในรูปที่ 4.1

ค่าความต้านทานจะมีอัตราส่วนเป็น R1 = 80R2 และ R4 = 80R3 = 80R5

4.2 ส่วนของการแปลงสัญญาณ

เนื่องจากสัญญาณที่ได้จะเป็นไฟกระแสสลับ ซึ่งมีลักษณะเป็นรูปคลื่น Sinusioudal เราจะนำมาแปลงสัญญาณเป็นข้อมูลทางดิจิตอล โดยผ่าน ADC0804 ซึ่งจะได้ข้อมูลออกมาขนาด 8 บิท โดยจะนำข้อมูลทั้งสองชุด ไปคำนวณต่อไป



รูปที่ 4.3 แสดงการต่อ ADC0804 เข้ากับไมโครคอนโทรลเลอร์

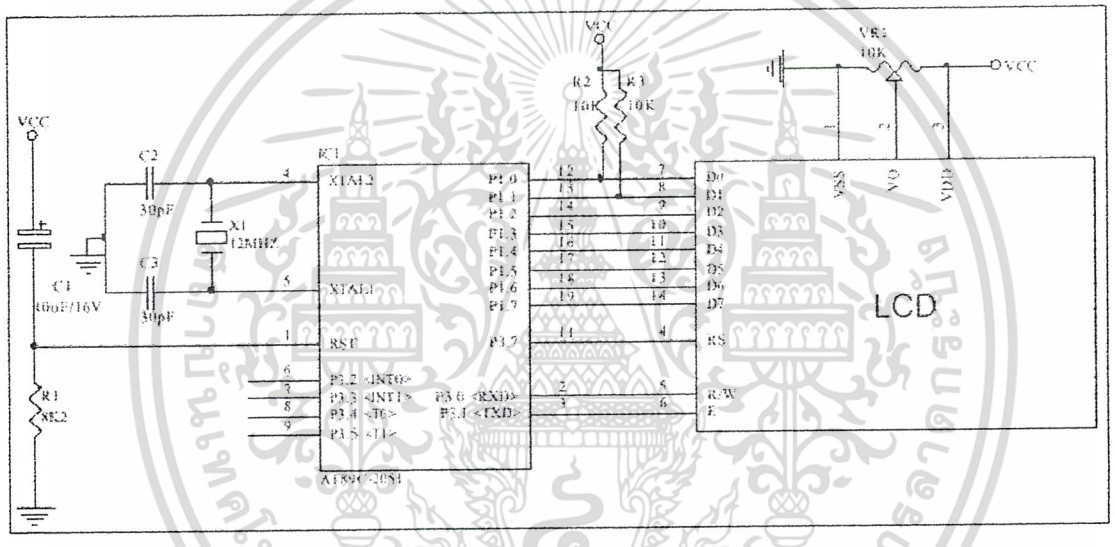
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 ส่วนของคอนโทรลเลอร์

ส่วนนี้เป็นส่วนควบคุมหลัก โดยใช้ไมโครคอนโทรลเลอร์ในการควบคุมการทำงาน ทั้งการรับข้อมูลจาก ADC0804 นำมาคำนวณโดยทฤษฎีการคำนวณทางกำลังไฟฟ้า แล้วทำการส่งข้อมูลไปแสดงผลทางหน้าจอ LCD

4.4 ส่วนของการแสดงผล

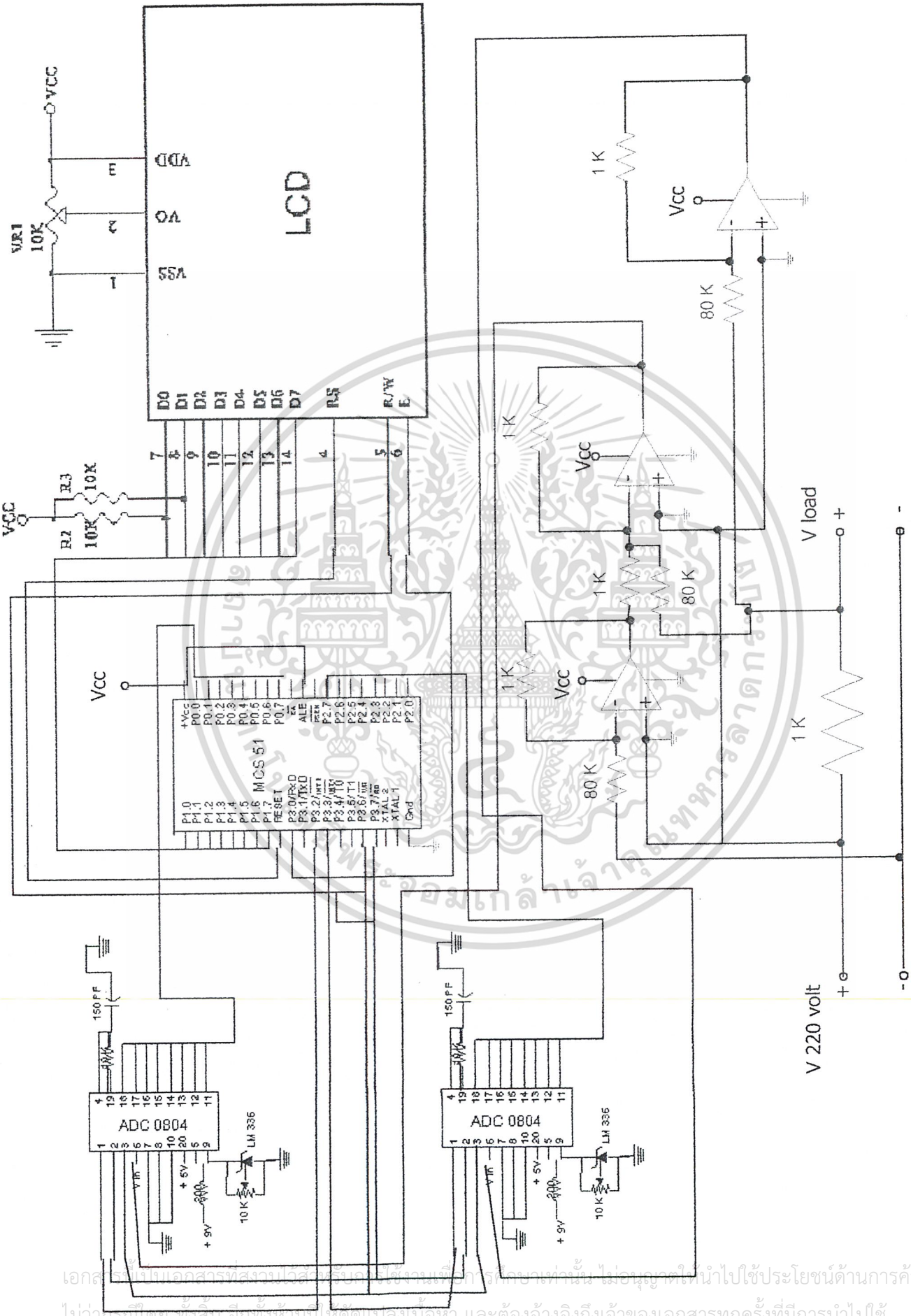
ส่วนนี้จะเป็นส่วนของหน้าจอ LCD (Liquid Crystal Device) โดยจะถูกควบคุมด้วยไมโครคอนโทรลเลอร์ จะทำการแสดงผลเป็นค่ากำลังไฟฟ้าที่วัดได้



รูปที่ 4.4 แสดงการต่อ LCD เข้ากับไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 วงจรรวม

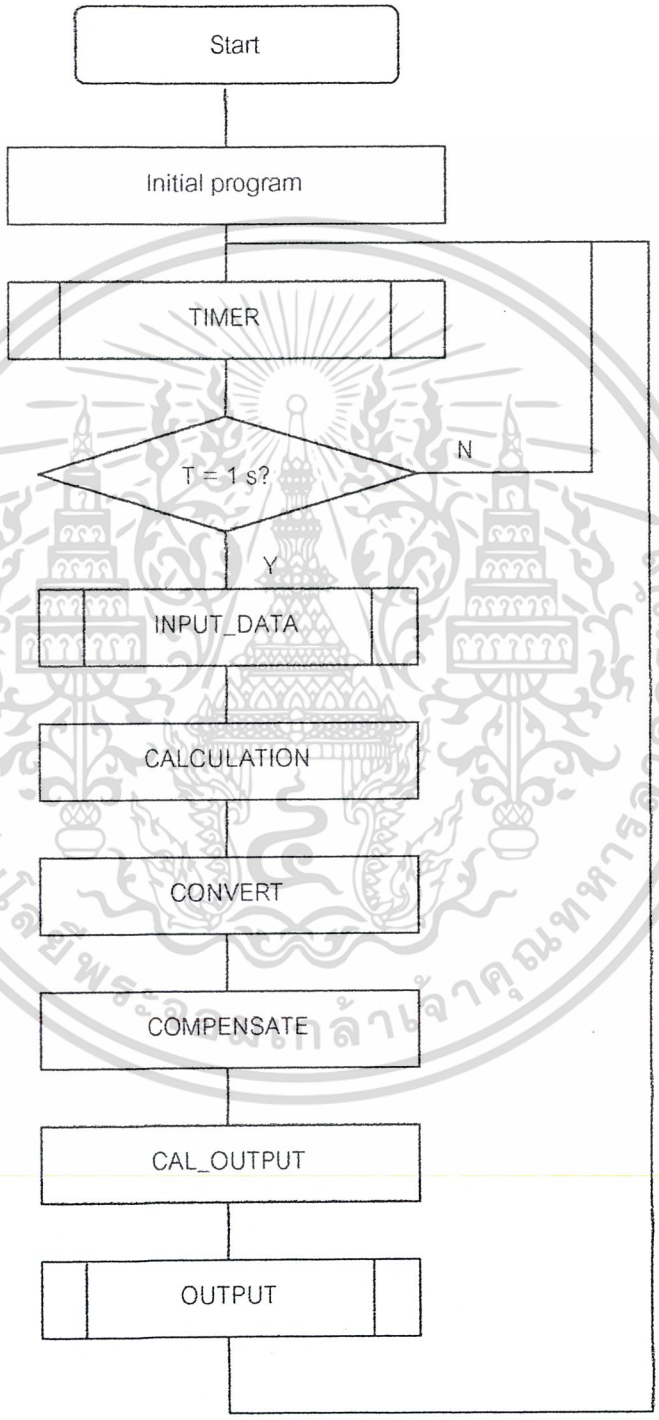


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.6 โปรแกรมการทำงานของเครื่องวัดกำลังไฟฟ้าภายในบ้าน

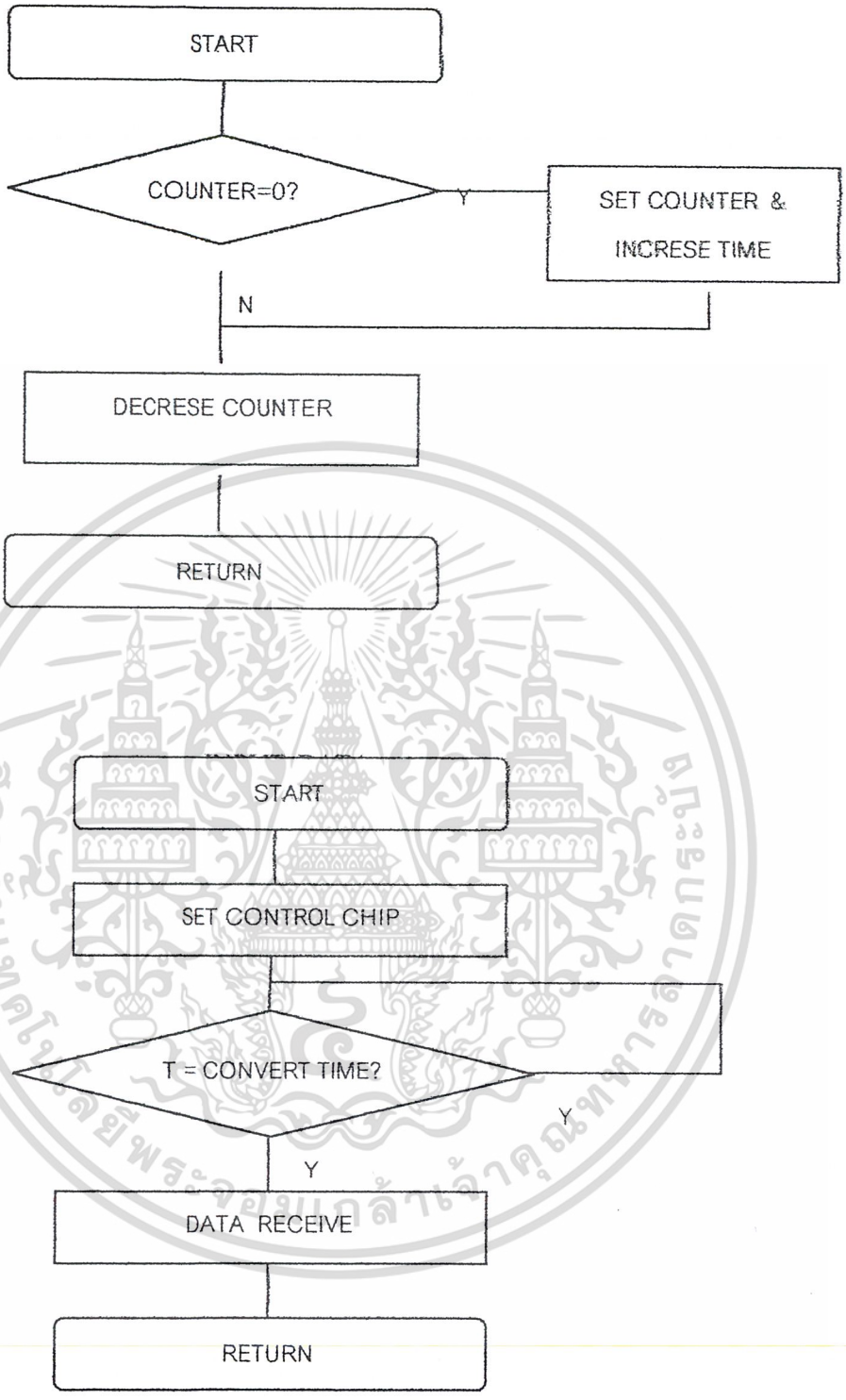
Flow Chart

MAIN :

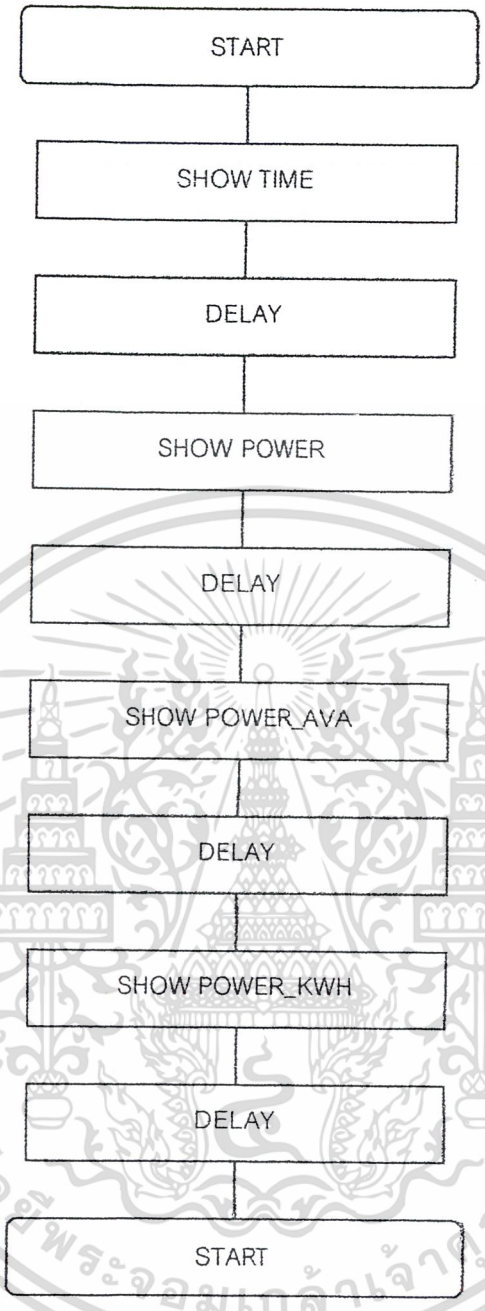


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TIMER :



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Program

```
; เข้า 400 V นะ

    time_h$    equ    50h          ;เวลาเป็นนาทีนะ
    time_l$           equ    51h
    multiplyer_h$     equ    52h          ;ผลหารจากข้อมูลหน้าสุดกับตัวเวลา
    multiplyer_l$     equ    53h
    aba_h$     equ    54h          ;เศษจากการหาร
    aba_l$            equ    55h
    multi_h$   equ    56h
    multi_l$          equ    57h
       min_val$            equ    58h
       temp1$         equ    10h
    temp2$            equ    11h
    tmp_ab1$  equ    12h
    tmp_ab2$  equ    13h
    chk_loop$ equ    14h          ; ตรวจสอบ loop ถ้าไม่ใช่ ตอบ 0 ใช่ ตอบ 1
    count_min$        equ    15h
    tmp_inp1$ equ    16h
    tmp_inp2$ equ    17h
    ini_time1$ equ   18h
    ini_time2$ equ   19h
    time_hbyte$       equ    1ah
    rare_data5$       equ    1bh
    rare_data4$       equ    1ch
    rare_data3$       equ    1dh
    rare_data2$       equ    1eh
    rare_data1$       equ    1fh
    substate5$ equ   60h
    substate4$ equ   61h
    substate3$ equ   62h
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
substate3$ equ     62h
substate2$ equ     63h
substate1$ equ     64h
   compan1$        equ     65h
   compan2$        equ     66h
   compan3$        equ     67h
   compan4$        equ     68h
   compan5$        equ     69h
   compan6$        equ     6Ah
   compan7$        equ     6Bh
   compan_mul$             equ     6Ch
   compan_div$             equ     6Dh
   comp_div_mul$           equ     6Eh
   comp_div_aba$equ        6Fh
   temp_cal5$      equ     70h
temp_cal4$         equ     71h
temp_cal3$         equ     72h
temp_cal2$         equ     73h
temp_cal1$         equ     74h
; $$$$$$$$$$$$  Start  $$$$$$$$$$$$
      org      0000h
      jmp      main
;========== Procedure ==========
   org         1000h
;****************  DIV4_4 **********************
div4_4: acall    adder          ; input เป็น16 bit  2 ชุด ตัวตั้ง(r0,r1) ตัวหาร(r2,r3) ผลหาร
(multplyer_l)
              mov     a,chk_loop
              jz      cont4
              mov     a,temp2$
              sjmp    in_loop
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                mov     b,r2
                div     ab
                jnz     cont
                xch     a,r7
                jz      cont1
                xch     a,r7
                inc     r6
                clr     c
                xch     a,r1
                subb    a,r3
                xch     a,r1
                xch     a,r0
                subb    a,r2
                xch     a,r0
                sjmp    cont2
cont1:          xch     a,r7
                sjmp    cont3
cont4:  mov     a,r2
                rlc     a
                jc      exceed_half
                sjmp    cont2
cont:           add     a,r6
                mov     temp2$,a
                sjmp    test
exceed_half:mov         a,#01h          ;ผลหารเป็น 1
                mov     temp2$,a
                sjmp    test
cont3:  mov     temp2$,a                ;ใช้ temp2 เป็น temp เก็บตัวตามคิดคูณ
in_loop:        mov     b,r3
                mul     ab                      ;คูณตัวหลัง
                mov     temp1$,b
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                xch     a,r1
                clr     c
                subb    a,r1
                xch     a,r5                    ; store ค่าเศษหลังที่ได้
                xch     a,r0
                subb    a,#00h          ; clr ทด
                xch     a,r0
                mov     a,temp2$
                mov     b,r2
                mul     ab                      ; คูณตัวหน้า
                add     a,temp1$
                jc      have_c          ; ถ้าตัวลบเกิน FFh
                xch     a,r0
                subb    a,r0
                mov     temp1$,a
                jnc     enddiv          ; ถ้าลบแล้วติดลบ
have_c: mov     a,#0ffh
                clr     c
                subb    a,r0
                clr     c
                subb    a,temp1$
                jc      have_c1                 ; ถ้าเกินตัวตั้ง
                xch     a,temp1$
                sjmp    enddiv
have_c1:        dec     temp2$          ; ลดค่า ผลลัพธ์
test:           mov     chk_loop,#01h ; set ค่า chk_loop เป็น false
                sjmp    div4_4
enddiv: xch     a,r4
                mov     multiplyer_1$,temp2$
                ret
;**************** END DIV4_4 ********************
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;********************  DIV4_2 ************************
div4_2: acall   div2_2
        acall   adder       ; input เป็น16 bit ตัวตั้ง(r0,r1) 8 bit ตัวหาร(r3) ผลหาร
(r6,r7)
next3:  mov     a,r0
        mov     b,r3
        div     ab
        jnz     next1
        mov     a,r7
        jz      next2
        mov     a,r1
        clr     c
        subb    a,r3
        mov     r1,a
        jnc     next2
        dec     r0
        sjmp    next3
next2:  mov     a,r1
        mov     b,r3
        div     ab
        mov     multiplyer_l$,a
        mov     r5,b
        jmp     fin
next1:  mov     multiplyer_h$,a
        mov     r0,b
        mov     a,b
loop:   anl     a,#0f0h     ; เลือก 4 bit แรก
        swap    a
        mov     r7,a        ; ใช้ r7 เก็บตัวคูณไว้
        mov     b,temp1$
        mul     ab
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        swap    a
        add     a,multiplyer_l$
        mov     multiplyer_l$,a
        mov     a,r7
        mov     b,temp2$
        mul     ab
        mov     r6,a        ; ใช้ r6 เก็บค่าเศษ
        mov     a,r0
        anl     a,#0fh
        swap    a
        mov     r0,a
        mov     a,r1
        swap    a
        anl     a,#0fh
        orl     a,r0
        mov     r0,a
        mov     a,r1
        anl     a,#0fh
        mov     r1,a
        mov     a,r0
        add     a,r6
        mov     r0,a
        mov     a,b
        jnc     next4
        inc     a
next4:  jz      next9
        mov     r7,a
        mov     b,temp1$
        mul     ab
        swap    a
        add     a,multiplyer_l$
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                mov     multiplyer_l$,a
                mov     a,r7
                mov     b,temp2
                mul     ab
                add     a,r0
                mov     r0,a
                mov     a,b
                jnc     next5
                inc     a
                sjmp    next5
next9:          mov     a,r0
                mov     b,r3
                div     ab
                swap    a
                add     a,multiplyer_l$
                mov     multiplyer_l$,a
                mov     r0,b
                sjmp    next10
next5:          jnz     next4
next10: mov     a,r0
                anl     a,#0f0h         ; เลือก 4 bit แรก
                swap    a
loop1:          mov     r7,a            ; ใช้ r7 เก็บตัวคูณไว้
                mov     b,temp1$
                mul     ab
                add     a,multiplyer_l$
                mov     multiplyer_l$,a
                mov     a,r7
                mov     b,temp2$
                mul     ab
                mov     r6,a            ; เก็บค่าเศษ
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        mov     a,r0
        anl     a,#0fh
        swap    a
        orl     a,r1
        add     a,r6
        mov     r0,a
        mov     r1,#00h
next8:  mov     a,b
        jnc     next6
        inc     a
next6:  jz      next7
        mov     r7,a        ; ใช้ r7 เก็บตัวคูณไว้
        mov     b,temp1$
        mul     ab
        add     a,multiplyer_l$
        mov     multiplyer_l$,a
        mov     a,r7
        mov     b,temp2$
        mul     ab
        add     a,r0
        mov     r0,a
        sjmp    next8
next7:  mov     a,r0
        mov     b,r3
        div     ab
        inc     a
        add     a,multiplyer_l$
        mov     multiplyer_l$,a
        mov     r5,b
        jnc     fin
        inc     multiplyer_h$
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
fin:            mov     r4,#00h
                ret
;***************  END DIV4_2 ********************
;******************  DIV2_2  ******************
div2_2: mov     a,#0ffh
                mov     b,r3
                div     ab
                xch     a,temp1$                    ; ใช้เก็บค่าผลหารชั่วคราว
                inc     b
                mov  temp2$,b
                xch     a,b
                clr     c
                subb    a,r3
                jz      inc_div
                ret
inc_div:        inc     temp1$                      ; แสดงว่าหารลงตัว
                mov     temp2$,#00h
                ret
;***************  END DIV2_2 ********************
;******************  ADDER  ********************
adder:  mov     r0,ini_time1$
                mov     r1,ini_time2$
                mov     a,tmp_ab2$
                add     a,r1
                xch     a,r1
                xch     a,r0
                addc    a,tmp_ab1$
                xch     a,r0
                mov     a,#00h
                addc    a,#00h
                xch     a,r7
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ret
;****************  END  ADDER ********************
;************  SELECT FIRST BIT  ****************
select_first_bit: mov   a,r1                    ; ส่งค่ากลับมาที่ r0
                swap    a
                mov     r7,a                    ; r7 temp
                anl     a,#0fh
                xch     a,r0
                xch     a,r7                    ; r7 restore
                anl     a,#0f0h
                xch     a,r1
                xch     a,r2
                swap    a
                mov     r7,a                    ; r7 temp
                anl     a,#0fh
                orl     a,r1
                xch     a,r1                    ; update $$## 2 bit แรก
                xch     a,r7                    ; r7 restore
                anl     a,#0f0h
                xch     a,r2                    ; update ##$0   bit3 bit4 is 0
                ret
;**********  END  SELECT FIRST BIT  ***************
; ***************  ABA MULTIPLY**********************
aba_mul:        mov     a,r0
                mov     b,multiplyer_l$
                mul     ab
                mov     r7,b                    ; r7 temp
                xch     a,r5                    ; r5 output bit ต่ำ
                mov     a,r0
                mov     b,multiplyer_h$
                mul     ab
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                add     a,r7                    ; r7 restore
                xch     a,r4                    ; r4 output bit สูง
                xch     a,b
                addc    a,#00h
                xch     a,r3                    ; r3 ทด bit เกิน
                mov     a,r0
                mov     b,aba_l$
                mul     ab
                add     a,r2
                xch     a,r2                    ; update OOXX 2 bit หลัง
                xch     a,b
                addc    a,#00h
                xch     a,r7                    ; r7 temp
                mov     a,r0
                mov     b,aba_h$
                mul     ab
                add     a,r7                    ; r7 restore
                add     a,r1
                xch     a,r1                    ; update XXOO 2 bit หน้า
                xch     a,b
                addc    a,#00h
                xch     a,r0                    ; r0 output for multiply next
end_aba:        ret
; ************* END  ABA MULTIPLY **************
; ******************* SWAP  MUL  ******************
swap_mul:       mov     a,multiplyer_l$
                swap    a
                mov     r7,a                    ; r7 temp
                anl     a,#0f0h
                xch     a,multi_l$
                xch     a,r7                    ; r7 restore
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                anl     a,#0fh
                xch     a,multi_h$
                mov     a,multiplyer_h$
                swap    a
                anl     a,#0f0h
                orl     a,multi_h$
                xch     a,multi_h$
                ret
;****************** END SWAP MUL ****************
;******************   EXCHANGE    ******************
exchange:       xch     a,multiplyer_l$
                xch     a,multi_l$
                xch     a,multiplyer_l$
                xch     a,multiplyer_h$
                xch     a,multi_h$
                xch     a,multiplyer_h$
                ret
;****************** END EXCHANGE    *************
; ***************  CHECK OVER ***************
check_over:mov  a,r0
                clr     c
                subb    a,time_h$
                xch     a,r0
                mov     a,r1
                clr     c
                subb    a,time_l$
                xch     a,r1
                inc     rare_data1$
end_check:      ret
;******************  END CHECK OVER ************
;************* GET  RARE  DATA *****************
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                org    1300h
get_rare_data:  mov    a,time_h$
                jz     get_c
                ljmp   get_con
get_c:  mov     a,time_l$
                dec    a
                jnz    below_10
                mov    rare_data4$,substate4$
                mov    rare_data3$,substate3$
                mov    rare_data2$,substate2$
                mov    rare_data1$,substate1$
                ljmp   get_fin
below_10:       mov    r1,substate2$
                mov    r2,substate1$
                mov    r0,substate3$
below_con1:mov         a,r0
                mov    b,multiplyer_l$
                mul    ab
                add    a,rare_data1$
                xch    a,rare_data1$
                mov    a,b
                addc   a,#00h
                xch    a,r7                    ; r7 temp
                mov    a,r0
                mov    b,multiplyer_h$
                mul    ab
                add    a,r7
                add    a,rare_data2$
                xch    a,rare_data2$
                xch    a,b
                addc   a,rare_data3$
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                xch     a,rare_data3$
                mov     a,r0
                mov     b,aba_1$
                mul     ab
                add     a,r2
                xch     a,r2
                mov     a,r1
                addc    a,#00h
                mov     r1,a
                mov     r0,#01h
                jc      below_con1
                mov     r0,#00h
                mov     a,r1
                mov     b,time_1$
                div     ab
                mov     r1,b
                add     a,rare_data2$
                xch     a,rare_data2$
                mov     a,#00h
                addc    a,rare_data3$
                xch     a,rare_data3$
                mov     a,r1
below_con2:mov  b,temp1$
                mul     ab
                add     a,rare_data1$
                xch     a,rare_data1$
                mov     a,#00h
                addc    a,rare_data2$
                xch     a,rare_data2$
                mov     r7,b
                mov     a,r1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        mov     b,temp2$
        mul     ab
        add     a,r7
        add     a,r2
        xch     a,r2
        mov     r1,#01h
        mov     a,r1
        jc      below_con2
        mov     r1,#00h
        mov     a,r2
        mov     b,time_1$
        div     ab
        add     a,rare_data1$
        xch     a,rare_data1$
        ljmp    get_fin
get_con: mov    a,substate5$          ; ooxxxxxxxx load bit 1,2
        swap    a
        mov     r7,a                  ; r7 temp
        anl     a,#0fh                ; select oxxxxxxxxx
        mov     r0,a                  ; r0 = bit1
        mov     a,substate3$
        swap    a
        anl     a,#0fh
        mov     r2,a                  ; store last bit ###@ ที่ r2
        mov     a,substate4$
        swap    a
        mov     r6,a                  ; r6 temp
        anl     a,#0f0h
        orl     a,r2
        xch     a,r2                  ; store bit 2 ##@$ ที่ r2
        xch     a,r6                  ; r6 restore
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                anl     a,#0fh
                mov     r1,a                    ; store bit 3 #@$$ ที่ r1
                xch     a,r7                    ; r7 restore
                anl     a,#0f0h
                orl     a,r1
                xch     a,r1                    ; store bit 4 @$$$ ที่ r1
get_con5:       mov     a,r0                    ; r0 คือ bit ที่เกินออกมา r7 output
                mov     b,multiplyer_1$
                mul     ab
                swap    a
                anl     a,#0f0h
                add     a,rare_data3$
                mov     rare_data3$,a           ; OXXXXX output
                mov     a,r0
                mov     b,aba_1$
                mul     ab
                add     a,r2
                xch     a,r2                    ; update bit 1,2 ##@@
                xch     a,r1
                addc    a,b
                xch     a,r1
                mov     a,r0
                mov     b,aba_h$                ; r0 restore
                mul     ab
                add     a,r1
                mov     r1,a                    ; update bit 3,4 @@$$
                jnc     get_con1
                inc     b
get_con1:       xch     a,b
                jz      get_con3
                mov     r0,a
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                sjmp    get_con5
get_con3:       acall   select_first_bit                 ; ส่งค่ากลับมาที่ r0 getcon3
CSOOOOXXXX C is complete S is select
                mov     a,substate3$
                anl     a,#0fh
                orl     a,r2
                xch     a,r2
get_con6:       acall   aba_mul
                xch     a,r5
                add     a,rare_data3$                   ; output OOXXXX
                xch     a,rare_data3$
                mov     a,r0
                jnz     get_con6
                acall   select_first_bit                ; ส่งค่ากลับมาที่ r0 format :
CCSOOOOXXX
                mov     a,substate2$
                swap    a
                anl     a,#0fh
                orl     a,r2
                xch     a,r2
                acall   exchange
get_con7:       acall   aba_mul
                mov     a,r5
                add     a,rare_data2$                   ; output XXOXXX
                xch     a,rare_data2$
                mov     a,r4
                addc    a,rare_data3$                   ; ouput OOXXXX
                xch     a,rare_data3$
                mov     a,r0
                jnz     get_con7
                acall   exchange
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                acall   select_first_bit                ; ส่งค่ากลับมาที่ r0 format :
CCCSOOOOXX
                mov     a,substate2$
                anl             a,#0fh
                orl             a,r2
                xch             a,r2
get_con8:       acall   aba_mul
                xch             a,r5
                add     a,rare_data2$                   ; outputXXOOXX
                xch             a,rare_data2$
                xch             a,r4
                addc    a,rare_data3$                   ; output OOXXXX
                xch             a,rare_data3$
                mov     a,r0
                jnz             get_con8
                acall   select_first_bit                ; ส่งค่ากลับมาที่ r0 format :
CCCCSOOOOX
                mov     a,substate1$
                swap    a
                anl             a,#0fh
                orl             a,r2
                xch             a,r2
                acall   exchange
get_con9:       acall   aba_mul
                xch             a,r5
                add     a,rare_data1$
                xch             a,rare_data1$
                mov     a,r4
                addc    a,rare_data2$
                xch             a,rare_data2$
                mov     a,r3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                addc    a,rare_data3$
                xch             a,rare_data3$
                mov     a,r0
                jnz             get_con9
                acall   exchange
                acall   select_first_bit                ; ส่งค่ากลับมาที่ r0 format :
CCCCCSOOOO
                mov     a,substate1$
                anl             a,#0fh
                orl             a,r2
                xch             a,r2
get_con10:      acall   aba_mul
                xch             a,r5
                add     a,rare_data1$
                xch             a,rare_data1$
                mov     a,r4
                addc    a,rare_data2$
                xch             a,rare_data2$
                mov     a,r3
                addc    a,rare_data3$
                xch             a,rare_data3$
                mov     a,r0
                jnz             get_con10
                acall   select_first_bit
                acall   exchange
get_con11:      acall   aba_mul
                mov     a,r4
                jz              get_fin
                add     a,rare_data1$
                xch             a,rare_data1$
                mov     a,r3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            addc    a,rare_data2$
            xch     a,rare_data2$
            xch     a,r5
            swap    a
            anl     a,#0fh
            mov     r7,a                    ; r7 temp
            mov     b,time_l$
            mul     ab
            add     a,r2
            xch     a,r2
            xch     a,b
            addc    a,r1
            xch     a,r1
            xch     a,r7                    ; r7 restore
            mov     b,time_h$
            mul     ab
            add     a,r1
            xch     a,r1
            xch     a,r0
            addc    a,b
            mov     r0,a
            jnz     get_con11
            acall   exchange
            acall   select_first_bit
get_con12:  acall   aba_mul
            xch     a,r4
            add     a,rare_data1$
            xch     a,rare_data1$
            mov     a,r3
            addc    a,rare_data2$
            xch     a,rare_data2$
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        mov     a,#00h
        mov     a,r5
        mov     b,time_l$
        mul             ab
        add     a,r2
        xch             a,r2
        xch             a,b
        addc    a,r1
        xch             a,r1
        mov     a,r0
        addc    a,#00h
        xch             a,r0
        mov     a,r5
        mov     b,time_h$
        mul             ab
        add     a,r1
        xch             a,r1
        xch             a,r0
        addc    a,b
        mov     r0,a
get_con14:  clr     c
        subb    a,time_h$
        jc              get_con13
        mov     a,time_l$
        clr             c
        subb    a,r1
        jz              get_con15
        jnc             get_con13
get_con15:  acall   check_over
        mov     a,r0
        sjmp    get_con14
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                jnz             get_con12
get_con13:      mov     a,r0
                mov     a,time_h
                jnz             get_fin
                xch             a,r1
                mov     b,time_l
                div             ab
                add     a,rare_data1$
                xch             a,rare_data1$
get_fin: ret
;********** END GET RARE DATA ***************
; ************* SELECT MIN VAL  ****************
select_min: mov a,tmp_inp1$
                clr             c
                subb    a,tmp_inp2$
                jc              sel_con1
                mov     min_val$,tmp_inp2$
                ret
sel_con1:mov    min_val$,tmp_inp1$
                ret
; *********** END SELECT MIN VAL  ************
; *********** SET COMPANSATE VAL.  *************
set_comp:       mov     a,min_val$
                        clr             c
                        subb    a,#13h
                        jc              set_con1
        ; มากกว่า 30 V
                        mov     compan_mul$,compan1$
                        mov     compan_div$,compan5$
                        ret
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
set_con1:       mov     a,min_val$                      ; น้อยกว่า 30
V
                clr             c
                subb    a,#09h
                jc              set_con2
        ; น้อยกว่า 30 V มากกว่า 15 V
                mov     compan_mul$,compan2$
                mov     compan_div$,compan5$
                ret
set_con2:       mov     a,min_val$                      ; น้อยกว่า 15
V
                clr             c
                subb    a,#04h
                jc              set_con3
                mov     compan_mul$,compan3$
                mov     compan_div$,compan5$
                ret
set_con3:       mov     a,min_val$
                clr             c
                subb    a,#02h
                jc              set_con4
                mov     compan_mul$,compan4$
                mov     compan_div$,compan6$
                ret
set_con4:       mov     compan_mul$,compan4$
                mov     compan_div$,compan6$
                ret
; ****** END  SET  COMPANSATE VAL *******
; **********  FIND SUB COMPONENT  **********
find_sub_comp: mov      r3,compan_div$
                acall   div2_2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                mov     comp_div_mul$,temp1$
                mov     comp_div_aba$,temp2$
                mov     temp1$,#00h
                mov     temp2$,#00h
                ret
; ******** END FIND SUB COMPONENT ******
;========== End Procedure ========
;////////////////// interupt timer 0 (for display)///////////////
                org     0300h
calc:   xch     a,time_l$
                add     a,#01h
                xch     a,time_l$
                jnc     nn
                inc     time_h$
nn:             mov     multiplyer_h$,#00h
                mov     multiplyer_l$,#00h          ; r0,r1 ใช้เป็นตัวตั้ง
;========== Set ค่า ==========
; 10000/time ได้เท่าไหร่เก็บค่าไว้คำนวณ
                mov     ini_time1$,#0ffh
                mov     ini_time2$,#0ffh
                mov     r2,time_h$
                mov     r3,time_l$
                mov     tmp_ab1$,#00h
                mov     tmp_ab2$,#01h
                mov     r6,#00h
                mov     r7,#00h
                mov     chk_loop,#00h
;========= end set ค่า =========
                mov     a,r2
                jnz     d4
d42:    lcall   div4_2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                sjmp    endd4x
d4:             mov     a,r3
                jnz     d44
                mov     time_l$,time_h$
                lcall   div4_2
                mov     multiplyer_l$,multiplyer_h$
                sjmp    endd4x
d44:            lcall   div4_4          ; input: ตัวตั้งเป็น FFFFH เพราะใช้ในการ
เทียบค่า ตัวหารคือตัวเดิม
endd4x:         mov     aba_h$,r4
                mov     aba_l$,r5
                mov     count_min$,#60
end_dis:        ret
;/////////////////  end display ////////////////
;///////////////// period_s ////////////////
                org     0100h
period_s:       djnz    time_hbyte$,pe1
                sjmp    pe2
pe1:            jmp     endpe
pe2:            dec     count_min
                mov     th0,#0bdh
                mov     tl0,#0beh
                mov     time_hbyte$,#10h
                mov     p1,#00h
wait_inp1:      jnb     ie0,wait_inp1
                mov     tmp_inp1$,p0
                clr     ie0
                mov     p1,#20h
wait_inp2:jnb   ie0,wait_inp2
                clr     ie0
                mov     tmp_inp2$,p0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        lcall   select_min
        lcall   set_comp
        lcall   find_sub_comp
        mov     a,tmp_inp1$
        mov     b,tmp_inp2$
        mul     ab
; ###############  ใส่คำนวณกลับตรงนี้  ###############
        mov     r7,b                    ; r7 temp
        mov     r0,#1ah                 ; คูณกลับ ด้วย 400/F
        mov     b,r0
        mul     ab
        xch     a,temp_cal1$
        xch     a,r7
        mov     r7,b
        mov     b,r0
        mul     ab
        add     a,r7                    ; r7 restore
        xch     a,temp_cal2$
        mov     a,#00h
        addc    a,b
        xch     a,temp_cal3$
        mov     a,temp_cal1$
        mov     b,r0
        mul     ab
        xch     a,temp_cal1$
        mov     r7,b                    ; r7 temp
        mov     a,temp_cal2$
        mov     b,r0
        mul     ab
        add     a,r7                    ; r7 restore
        xch     a,temp_cal2$
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
mov     a,#00h
addc    a,b
xch     a,r7                    ; r7 temp
mov     a,temp_cal3$
mov     b,r0
mul     ab
add     a,r7                    ; r7 restore
xch     a,temp_cal3$
mov     a,#00h
addc    a,b
xch     a,temp_cal4$
mov     r0,compan_mul$
xch     a,temp_cal1$
mov     b,r0
mul     ab
xch     a,temp_cal1$
mov     r7,b                    ; r7 temp
xch     a,temp_cal2$
mov     b,r0
mul     ab
add     a,r7
xch     a,temp_cal2$
mov     a,#00h
addc    a,b
xch     a,r7
mov     a,temp_cal3$
mov     b,r0
mul     ab
add     a,r7
xch     a,temp_cal3$
mov     a,#00h
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        addc    a,b
        xch     a,r7
        mov     a,temp_cal4$
        mov     b,r0
        mul     ab
        add     a,r7
        xch     a,temp_cal4$
        mov     a,#00h
        addc    a,b
        xch     a,temp_cal5$
        mov     r0,#11h                 ; หารด้วย 255 2 ครั้ง
        mov     r1,#0ffh
        mov     r2,#01h
        mov     a,temp_cal5$
cal_con1:
        mov     b,r1
        mul     ab
        xch     a,r3                    ; output bit4
        xch     a,temp_cal5$
        add     a,temp_cal4$
        xch     a,temp_cal4$
        mov     chk_loop$,#02h
        jnc     cal_con2
        mov     temp_cal5$,r2
        sjmp    cal_con1
cal_con2:
        xch     a,temp_cal4$
        mov     b,r0
        div     ab
        add     a,r3
        xch     a,r3
        mov     r4,#00h
cal_con3:
        mov     r7,b                    ; r7 temp
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            xch     a,b
            mov     b,r1
            mul     ab
            add     a,r4
            xch     a,r4
            xch     a,r7                    ; r7 restore
            add     a,temp_cal3$
            xch     a,temp_cal3$
            mov     b,r2
            jc      cal_con3
            xch     a,temp_cal3$
            mov     b,r0
            div     ab
            add     a,r4
            xch     a,r4
            mov     r5,#00h
cal_con4:   mov     r7,b                    ; r7 temp
            xch     a,b
            mov     b,r1
            mul     ab
            add     a,r5
            xch     a,r5
            xch     a,r7                    ; r7 restore
            add     a,temp_cal2$
            xch     a,temp_cal2$
            mov     b,r2
            jc      cal_con4
            xch     a,temp_cal2$
            mov     b,r0
            div     ab
            add     a,r5
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                xch     a,r5
                mov     r6,#00h
cal_con5:       mov     r7,b                    ; r7 temp
                xch     a,b
                mov     b,r1
                mul     ab
                add     a,r6
                xch     a,r6
                xch     a,r7                    ; r7 restore
                add     a,temp_cal1$
                xch     a,temp_cal1$
                mov     b,r2
                jc      cal_con5
                xch     a,temp_cal1$
                mov     b,r0
                div     ab
                add     a,r6
                xch     a,r6
                mov     temp_cal5$,#00h
                mov     temp_cal4$,r3
                mov     temp_cal3$,r4
                mov     temp_cal2$,r5
                mov     temp_cal1$,r6
                mov     r3,#00h
                mov     r4,#00h
                djnz    chk_loop$,cal_con2
                mov     r0,compan_div$
                mov     r1,comp_div_mul$
                mov     r2,comp_div_aba$
                mov     r5,#00h
                mov     r6,#00h
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
cal_con6:   xch   a,r3
            add   a,temp_cal4$
            xch   a,r3
            mov   a,temp_cal4$
            mov   b,r2
            mul   ab
            add   a,temp_cal3$
            xch   a,temp_cal3$
            jc    cal_con6
            mov   r4,#00h
            xch   a,temp_cal3$
            mov   b,r0
            div   ab
            add   a,r3
            xch   a,r3
            xch   a,b
cal_con7:   mov   r7,a                ;r7 temp
            add   a,r4
            xch   a,r4
            xch   a,r7                ; r7 restore
            mov   b,r2
            mul   ab
            add   a,temp_cal2$
            xch   a,temp_cal2$
            mov   a,b
            addc  a,#00h
            jnz   cal_con7
            xch   a,temp_cal2$
            mov   b,r0
            div   ab
            add   a,r4
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                xch     a,r4
                xch     a,b
cal_con8:       mov     r7,a
                add     a,r5
                xch             a,r5
                xch             a,r7
                mov     b,r2
                mul             ab
                add     a,temp_cal1$
                xch             a,temp_cal1$
                mov     a,b
                addc    a,#00h
                jnz             cal_con8
                xch             a,temp_cal1$
                mov     b,r0
                div             ab
                add     a,r5
                xch             a,temp_cal1$
                mov     temp_cal2$,r4
                mov     temp_cal3$,r3
; ########################################
                mov     a,substate1$
                add     a,temp_cal1$
                xch             a,substate1$
                mov     a,substate2$
                addc    a,temp_cal2$
                xch             a,substate2$
                jnc             endpe
                clr             c
                xch             a,substate3$
                add     a,#01h
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                xch         a,substate3$
                jnc         endpe
                clr         c
                xch         a,substate4$
                add     a,#01h
                xch         a,substate1$
                jnc         endpe
                xch         a,substate5$
                inc         a
                xch         a,substate5$
endpe:  ret
;////////////////// end period_s  ////////////////
;//////////////////   MAIN   ////////////////
                org     0050h
main:   mov     compan1$,#0D5h
                mov     compan2$,#0DFh
                mov     compan3$,#0EBh
                mov     compan4$,#0FFh
                mov     compan5$,#0C8h
                mov     compan6$,#0B2h
                mov     compan7$,#78h
                mov     time_h$,#00h                    ; set time
                mov     time_l$,#00h
                mov     ie,#80h                         ; set interupt
                mov     tmod,#01h
                mov     th0,#0bdh
                mov     tl0,#0beh
                mov     time_hbyte$,#10h
                mov     count_min$,#60
                setb    tr0
wait:   jnb         tf0,wait
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        clr         tf0
        acall   period_s
        mov     a,count_min$
        jnz         wait
        acall   calc
        mov     rare_data5$,#00h
        mov     rare_data4$,#00h
        mov     rare_data3$,#00h
        mov     rare_data2$,#00h
        mov     rare_data1$,#00h
        lcall   swap_mul
        lcall   get_rare_data
        sjmp    wait
        end
;//////////////  END  MAIN ////////////
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

การทดลองและผลการทดลอง

5.1 การทดลอง

ในการทดลองเราจะทำการต่อวงจรรวม แล้วป้อนค่าแรงดันโดยให้ Vin1 เป็นค่าแรงดันไฟบ้าน และให้ Vin2 เป็นค่าแรงดันที่ตกคร่อมตัวความต้านทานขนาด 100 โอห์ม โดยจะป้อนค่าแรงดันเป็น 2 , 4 , 6 , 8 , 10 , 12 , 14 , 16 , 18 และ 20 โวลท์ โดยจะทำการกำหนดให้ Vin2 เป็นค่าคงที่ แล้ววัดค่ากำลังไฟฟ้า จากนั้นนำค่ากำลังไฟฟ้าที่คำนวณไว้กับที่วัดได้นำมา Plot Graph เปรียบเทียบกัน

5.2 ผลการทดลอง

ในตารางเราจะกำหนด

Vin1 เป็น แรงดันไฟบ้าน

Vin2 เป็น แรงดันที่ตกคร่อมตัวต้านทานขนาด 100 โอห์ม

Preal เป็น ค่ากำลังไฟฟ้าที่ได้จากการคำนวณ

Pmeasure เป็น ค่ากำลังไฟฟ้าที่วัดได้

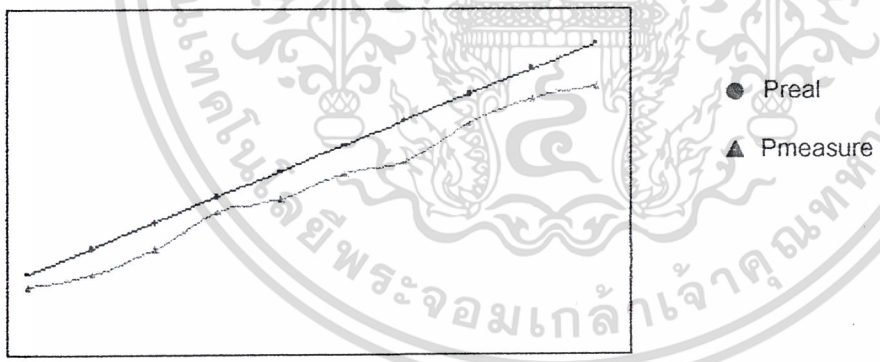
จากนั้นนำมา Plot Graph เปรียบเทียบความแตกต่างของค่า Preal กับ Pmeasure โดยค่า Preal จะคำนวณได้จากสูตร

$$Preal = (Vin1 \times Vin2) / 100$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Vin1	Vin2	Preal	Pmeasure
2	2	0.04	0.019607843
4	2	0.08	0.039215686
6	2	0.12	0.078431373
8	2	0.16	0.137254902
10	2	0.2	0.156862745
12	2	0.24	0.196078431
14	2	0.28	0.215686275
16	2	0.32	0.274509804
18	2	0.36	0.31372549
20	2	0.4	0.333333333

ตารางที่ 5.1 แสดงค่า Pmeasure เปรียบเทียบกับ Preal โดยกำหนด Vin2 = 2 โวลท์

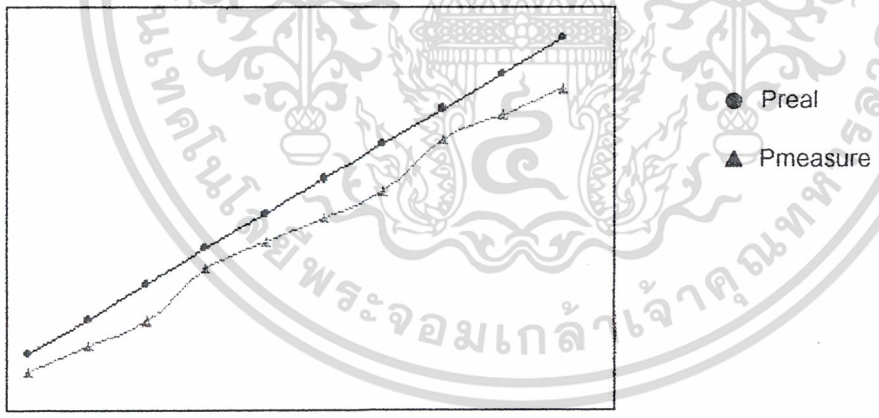


รูปที่ 5.1 กราฟแสดงการเปรียบเทียบค่า Pmeasure กับ Preal โดยกำหนด Vin2 = 2 โวลท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Vin1	Vin2	Preal	Pmeasure
2	4	0.08	0.039215686
4	4	0.16	0.098039216
6	4	0.24	0.156862745
8	4	0.32	0.274509804
10	4	0.4	0.333333333
12	4	0.48	0.392156863
14	4	0.56	0.450980392
16	4	0.64	0.568627451
18	4	0.72	0.62745098
20	4	0.8	0.68627451

ตารางที่ 5.2 แสดงค่า Pmeasure เปรียบเทียบกับ Preal โดยกำหนด Vin2 = 4 โวลท์

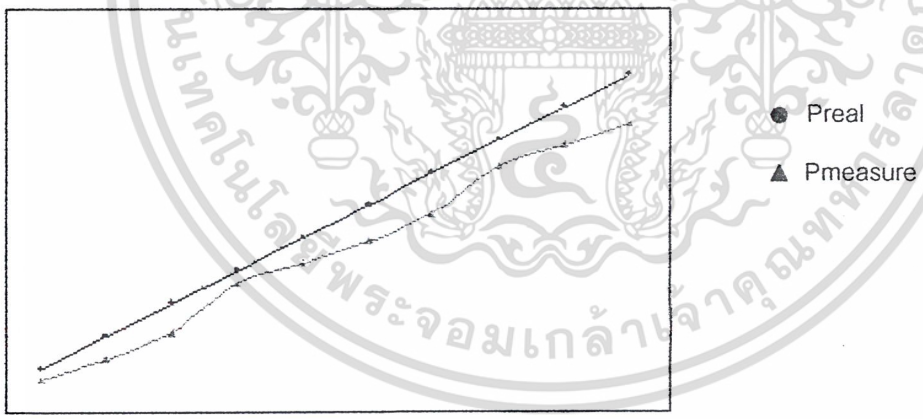


รูปที่ 5.2 กราฟแสดงการเปรียบเทียบค่า Pmeasure กับ Preal โดยกำหนด Vin2 = 4 โวลท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Vin1	Vin2	Preal	Pmeasure
2	6	0.12	0.078431373
4	6	0.24	0.156862745
6	6	0.36	0.254901961
8	6	0.48	0.431372549
10	6	0.6	0.509803922
12	6	0.72	0.588235294
14	6	0.84	0.68627451
16	6	0.96	0.862745098
18	6	1.08	0.941176471
20	6	1.2	1.019607843

ตารางที่ 5.3 แสดงค่า Pmeasure เปรียบเทียบกับ Preal โดยกำหนด Vin2 = 6 โวลท์

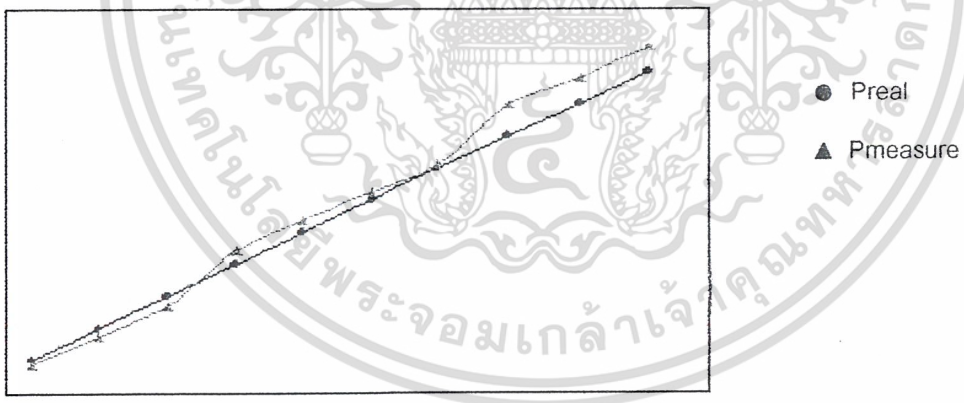


รูปที่ 5.3 กราฟแสดงการเปรียบเทียบค่า Pmeasure กับ Preal โดยกำหนด Vin2 = 6 โวลท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Vin1	Vin2	Preal	Pmeasure
2	8	0.16	0.137254902
4	8	0.32	0.274509804
6	8	0.48	0.431372549
8	8	0.64	0.705882353
10	8	0.8	0.862745098
12	8	0.96	1
14	8	1.12	1.137254902
16	8	1.28	1.431372549
18	8	1.44	1.568627451
20	8	1.6	1.725490196

ตารางที่ 5.4 แสดงค่า Pmeasure เปรียบเทียบกับ Preal โดยกำหนด Vin2 = 8 โวลท์

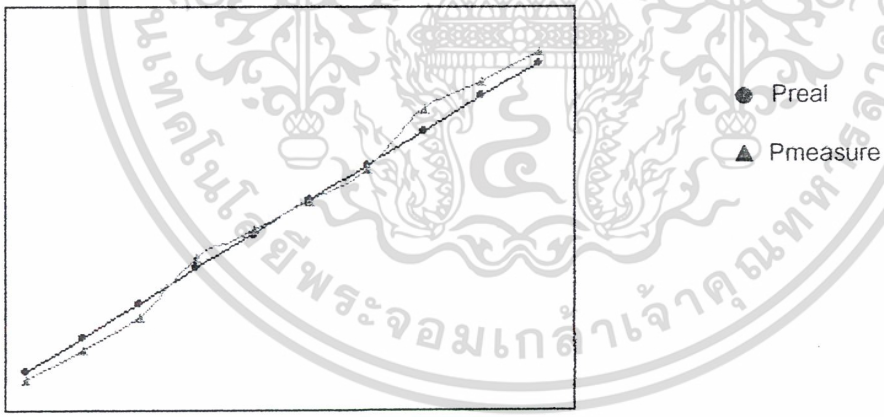


รูปที่ 5.4 กราฟแสดงการเปรียบเทียบค่า Pmeasure กับ Preal โดยกำหนด Vin2 = 8 โวลท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Vin1	Vin2	Preal	Pmeasure
2	10	0.2	0.156862745
4	10	0.4	0.333333333
6	10	0.6	0.509803922
8	10	0.8	0.862745098
10	10	1	1.019607843
12	10	1.2	1.196078431
14	10	1.4	1.37254902
16	10	1.6	1.725490196
18	10	1.8	1.882352941
20	10	2	2.058823529

ตารางที่ 5.5 แสดงค่า Pmeasure เปรียบเทียบกับ Preal โดยกำหนด Vin2 = 10 โวลท์

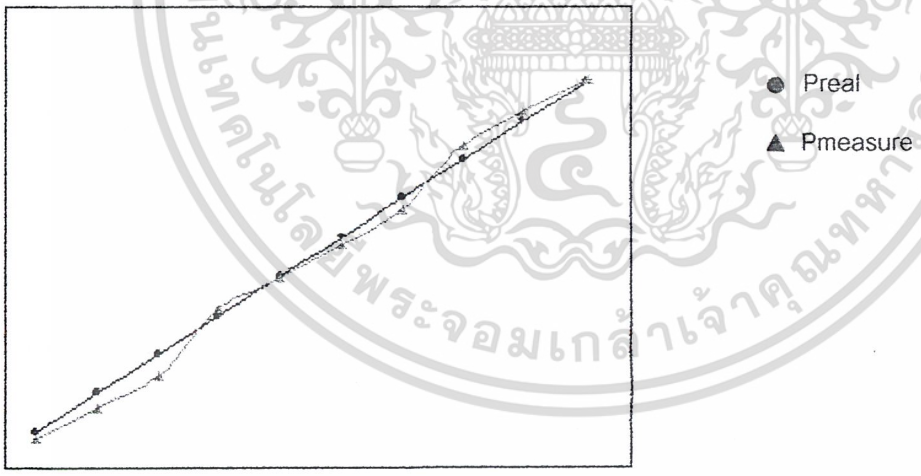


รูปที่ 5.5 กราฟแสดงการเปรียบเทียบค่า Pmeasure กับ Preal โดยกำหนด Vin2 = 10 โวลท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Vin1	Vin2	Preal	Pmeasure
2	12	0.24	0.196078431
4	12	0.48	0.392156863
6	12	0.72	0.588235294
8	12	0.96	1
10	12	1.2	1.196078431
12	12	1.44	1.392156863
14	12	1.68	1.607843137
16	12	1.92	2
18	12	2.16	2.215686275
20	12	2.4	2.411764706

ตารางที่ 5.6 แสดงค่า Pmeasure เปรียบเทียบกับ Preal โดยกำหนด Vin2 = 12 โวลท์

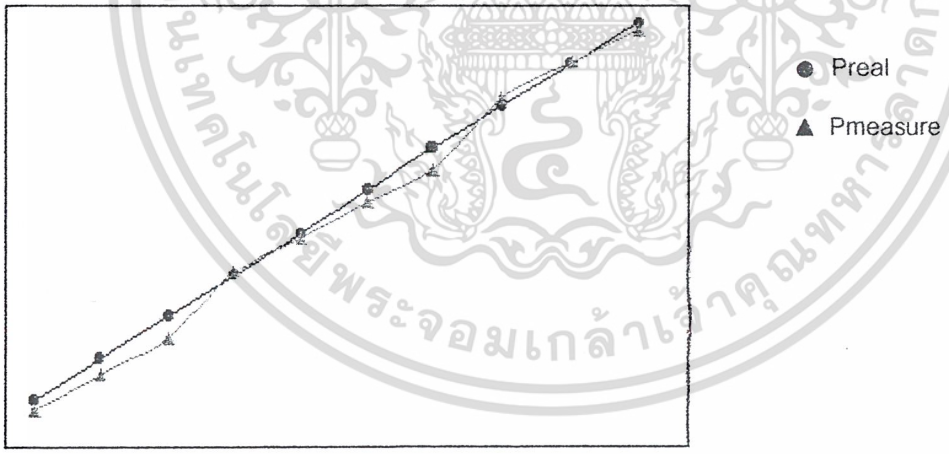


รูปที่ 5.6 กราฟแสดงการเปรียบเทียบค่า Pmeasure กับ Preal โดยกำหนด Vin2 = 12 โวลท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Vin1	Vin2	Preal	Pmeasure
2	14	0.28	0.215686275
4	14	0.56	0.450980392
6	14	0.84	0.68627451
8	14	1.12	1.137254902
10	14	1.4	1.37254902
12	14	1.68	1.607843137
14	14	1.96	1.823529412
16	14	2.24	2.294117647
18	14	2.52	2.529411765
20	14	2.8	2.745098039

ตารางที่ 5.7 แสดงค่า Pmeasure เปรียบเทียบกับ Preal โดยกำหนด Vin2 = 14 โวลท์

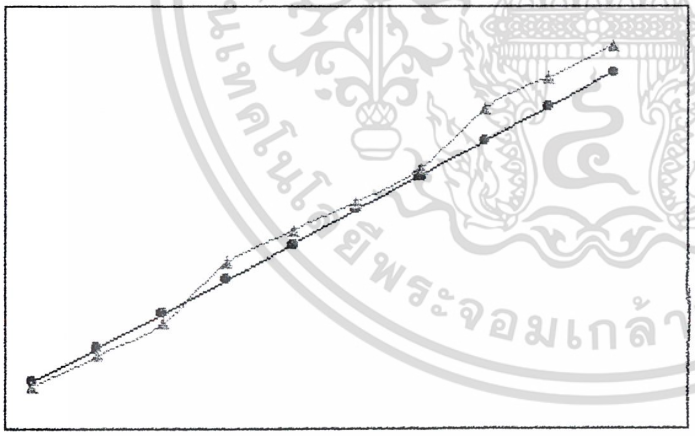


รูปที่ 5.7 กราฟแสดงการเปรียบเทียบค่า Pmeasure กับ Preal โดยกำหนด Vin2 = 14 โวลท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Vin1	Vin2	Preal	Pmeasure
2	16	0.32	0.274509804
4	16	0.64	0.568627451
6	16	0.96	0.862745098
8	16	1.28	1.431372549
10	16	1.6	1.725490196
12	16	1.92	2
14	16	2.24	2.294117647
16	16	2.56	2.862745098
18	16	2.88	3.156862745
20	16	3.2	3.450980392

ตารางที่ 5.8 แสดงค่า Pmeasure เปรียบเทียบกับ Preal โดยกำหนด Vin2 = 16 โวลท์

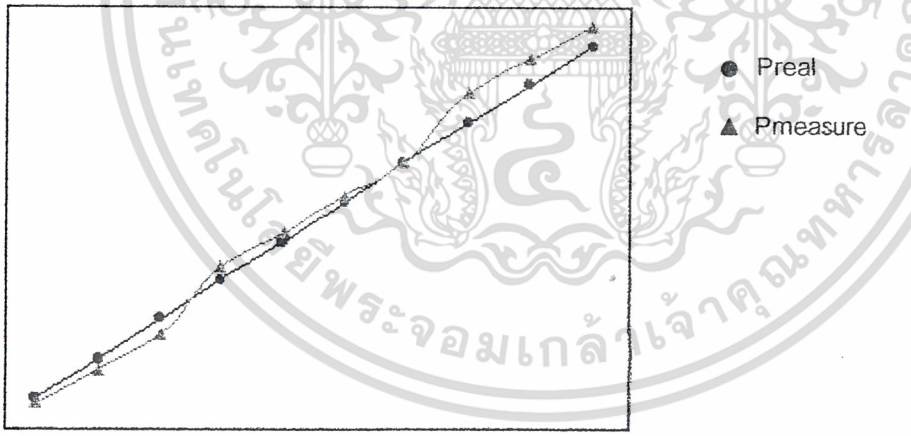


รูปที่ 5.8 กราฟแสดงการเปรียบเทียบค่า Pmeasure กับ Preal โดยกำหนด Vin2 = 16 โวลท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Vin1	Vin2	Preal	Pmeasure
2	18	0.36	0.31372549
4	18	0.72	0.62745098
6	18	1.08	0.941176471
8	18	1.44	1.568627451
10	18	1.8	1.882352941
12	18	2.16	2.215686275
14	18	2.52	2.529411765
16	18	2.88	3.156862745
18	18	3.24	3.470588235
20	18	3.6	3.784313725

ตารางที่ 5.9 แสดงค่า Pmeasure เปรียบเทียบกับ Preal โดยกำหนด Vin2 = 18 โวลท์

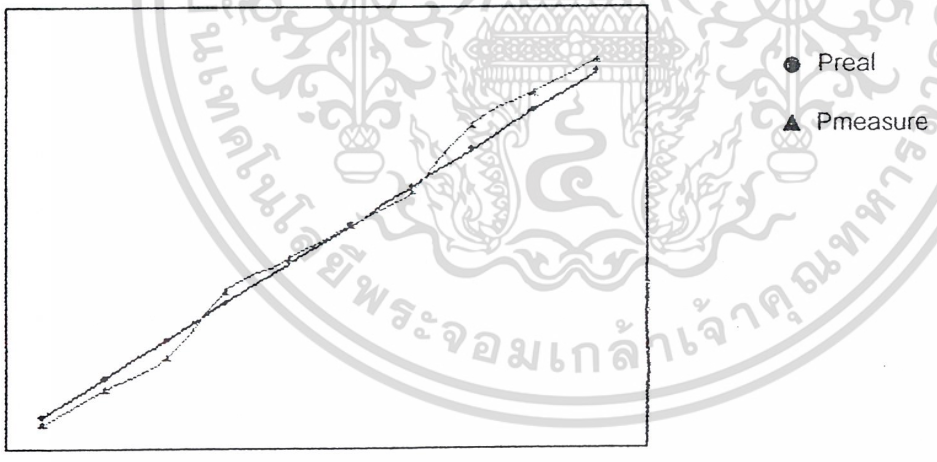


รูปที่ 5.9 กราฟแสดงการเปรียบเทียบค่า Pmeasure กับ Preal โดยกำหนด Vin2 = 18 โวลท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Vin1	Vin2	Preal	Pmeasure
2	20	0.4	0.333333333
4	20	0.8	0.68627451
6	20	1.2	1.019607843
8	20	1.6	1.725490196
10	20	2	2.058823529
12	20	2.4	2.411764706
14	20	2.8	2.745098039
16	20	3.2	3.450980392
18	20	3.6	3.784313725
20	20	4	4.137254902

ตารางที่ 5.10 แสดงค่า Pmeasure เปรียบเทียบกับ Preal โดยกำหนด Vin2 = 20 โวลท์



รูปที่ 5.10 กราฟแสดงการเปรียบเทียบค่า Pmeasure กับ Preal โดยกำหนด Vin2 = 20 โวลท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6
บทสรุปและวิจารณ์

6.1 บทสรุป

ผลการทดลองที่ได้มาก็คือ กำลังไฟฟ้าที่ได้ต่อการใช้เครื่องใช้ไฟฟ้าชนิดนั้นๆ ซึ่งอุปกรณ์ที่ผลิตขึ้นมานี้สามารถที่จะอำนวยประโยชน์ ในการที่เราสามารถทราบค่ากำลังไฟฟ้าที่เราต้องสูญเสียไปให้กับเครื่องใช้ไฟฟ้าชนิดนั้นๆ

การดำเนินงานที่ผ่านมานั้นเริ่มตั้งแต่การคิดโครงงานเพื่อสร้างอุปกรณ์ที่แปลกใหม่ และอยู่บนพื้นฐานของความต้องการของทุกๆคน จากนั้นจึงศึกษาความเป็นไปได้และสิ่งที่จะนำมาเป็นส่วนประกอบในการสร้าง ซึ่งการดำเนินการในส่วนแรกนี้จำเป็นต้องมีการศึกษาอย่างละเอียดและรอบคอบ เพื่อให้การดำเนินงานเป็นไปอย่างมีประสิทธิภาพและเกิดข้อผิดพลาดน้อยที่สุด

จากขั้นตอนการคิดค้นโครงงานเราจะนำแนวความคิดมาสร้างและพัฒนาขึ้น ซึ่งในส่วนนี้จำเป็นต้องอาศัยการแก้ปัญหาที่เกิดขึ้นในการดำเนินงาน เพื่อให้ได้ผลลัพธ์เป็นที่น่าพอใจ

สุดท้ายแล้วเราจึงได้ผลสรุปออกมาตามที่ต้องการ

6.2 บทวิจารณ์

ในผลการทดลองที่ได้มานี้จะทำการเปรียบเทียบค่าที่วัดได้กับค่าที่คำนวณ ออกมาเป็นรูปกราฟ ซึ่งจะเห็นได้ว่ามีแนวโน้มใกล้เคียงกัน การทดลองนี้เป็นการชี้ให้เห็นว่าอุปกรณ์นี้สามารถนำไปวัดค่ากำลังงานได้

ในการทำโครงงานนี้จะเห็นว่า มีส่วนประกอบมากมาย ทั้งในส่วนของการลดขนาดสัญญาณ การแปลงสัญญาณ และการแสดงผล ทุกๆองค์ประกอบนั้นจะถูกนำมาประกอบเข้าเป็นวงจรเดียว ซึ่งสิ่งที่สำคัญที่สำคัญที่สุดคือส่วนของอินพุท ซึ่งจำเป็นต้องมีความถูกต้องมากที่สุด เนื่องจากโครงงานนี้ทำงานด้วยพื้นฐานของการวัด และอีกส่วนที่สำคัญคือโปรแกรมที่ใช้ในควบคุมการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

LC 7985

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Ordering number: EN 3255C

CMOS LSI

LC7985NA, LC7985ND

LCD Controller/Driver

Overview

The LC7985 series devices are low-power CMOS ICs that incorporate dot-matrix character generator, display controller and driver functions in a single device, making them ideal for use in portable equipment containing LCD displays.

The LC7985 series feature 5 × 7-pixel and 5 × 10-pixel character fonts including either eight or four user-defined characters, single-line and two-line display modes, built-in drivers for displays up to eight characters in size, and easy expansion to control displays of up to 80 characters by adding LC7930N display drivers.

The LC7985 series interface directly to both 4-bit and 8-bit microcontrollers. The instruction set includes display clear, cursor home, display ON/OFF, character blink, and cursor and display shift instructions. The built-in reset circuit automatically initializes the devices at power-ON.

The LC7985 series operate from a 5V supply and are available in 80-pin QIPs.

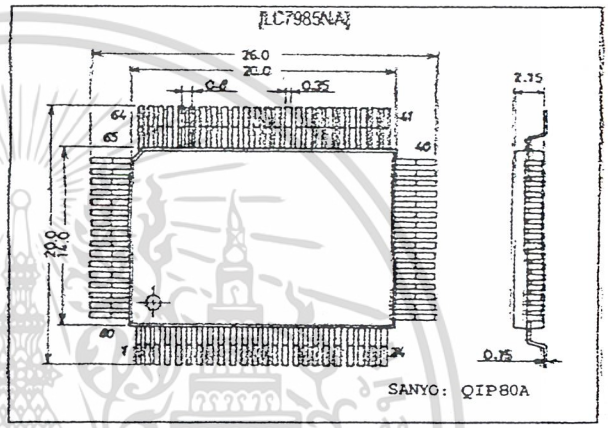
Features

- Controller and driver for dot-matrix LCD displays
- 5 × 7-pixel and 5 × 10-pixel character fonts
- 160, 5 × 7-pixel characters and 32, 5 × 10-pixel characters in character generator ROM
- Eight, 5 × 7-pixel characters or four, 5 × 10-pixel characters in character generator RAM
- 80-character display data RAM
- Built-in drivers for 1-line × 8-character and 2-line × 8-character displays
- Easy expansion to 1-line × 80-character or 2-line × 40-character displays
- 4-bit or 8-bit microcontroller interface
- 11 microcontroller instructions
- Built-in reset circuit
- Built-in oscillator
- 5V supply
- 80-pin QIP

Package Dimensions

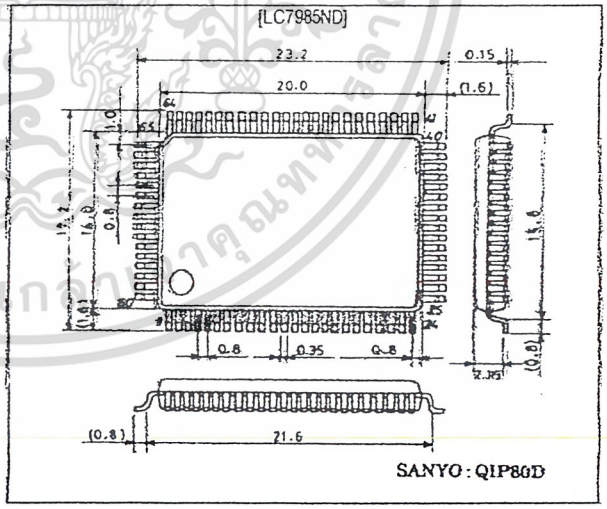
unit: mm

3044B - QFP80A



unit: mm

3177 - QFP80D

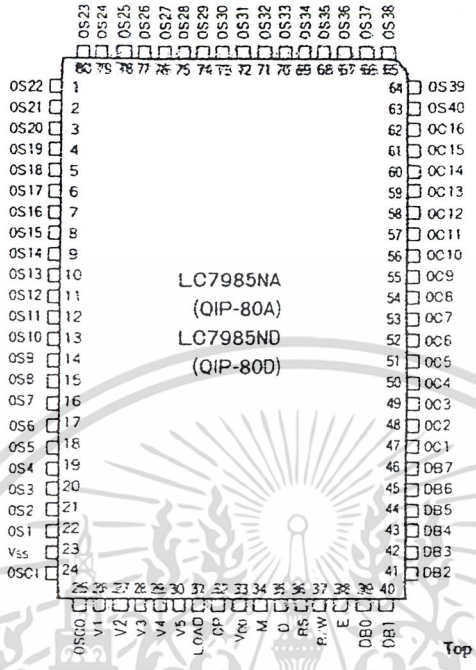


SANYO Electric Co., Ltd. Semiconductor Business Headquarters
TOKYO OFFICE Tokyo Bldg., 1-10, 1 Chome, Ueno, Taito-ku, TOKYO, 110 JAPAN

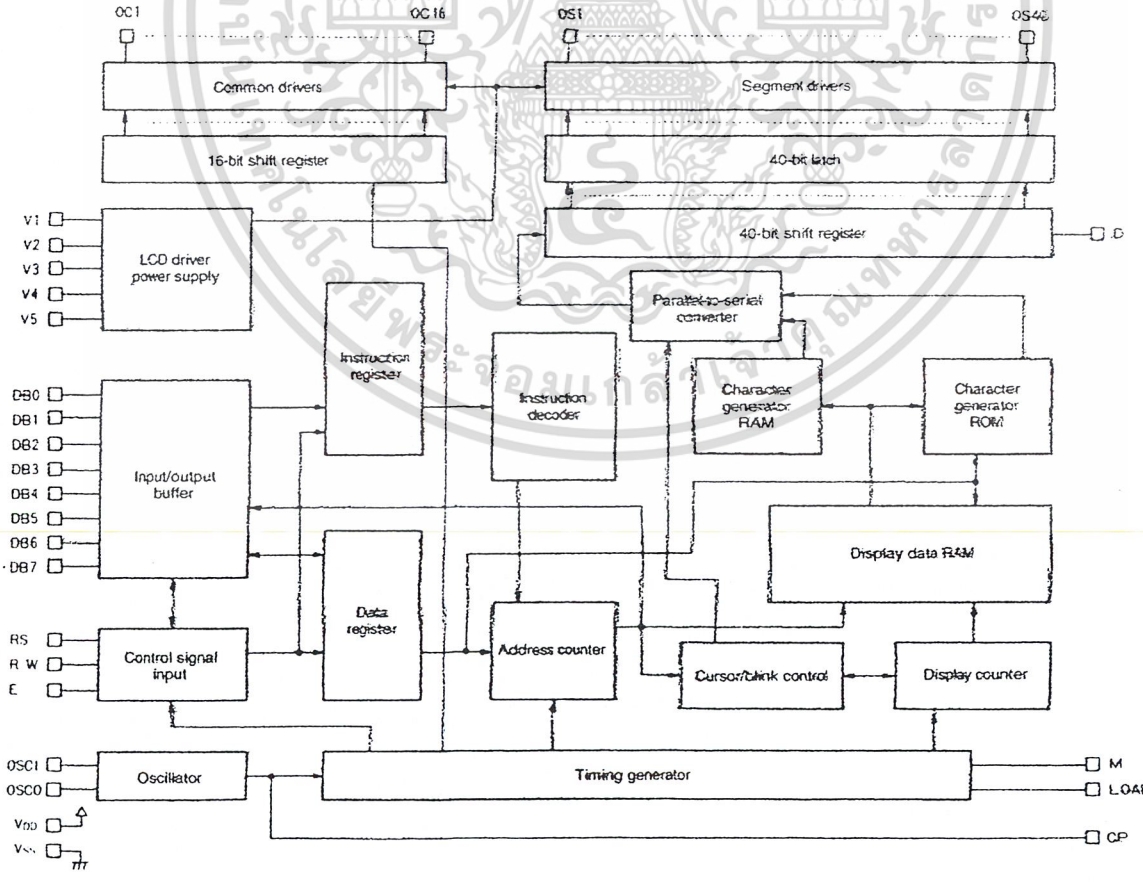
70197HA (ID) / N062JN No. 3255—1/30

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Assignment



Block Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Specifications

Absolute Maximum Ratings at Ta = 25 ± 2°C, V_{SS} = 0V

Parameter	Symbol	Ratings	Unit
Supply voltage range	V_{DD}	−0.3 to +7.0	V
LCD drive supply voltage range*1	V_1 to V_5	V_{DD} − 13.5 to V_{DD} + 0.3	V
Input voltage range	V_I	−0.3 to V_{DD} + 0.3	V
Operating temperature range	Topr	−20 to +75	°C
Storage temperature range	Tstg	−55 to +125	°C

Note: *1. V_{DD} must obey the relationship : $V_{DD} \geq V_1 \geq V_2 \geq V_3 \geq V_4 \geq V_5$

Allowable Operating Ranges at Ta = −20 to +75°C

Parameter	Symbol	Conditions	Ratings			Unit
			min	typ	max	
Supply voltage range	V_{DD}		4.5		5.5	V
Supply voltage*1	V_{D5}	$V_{D5} = V_{DD} - V_5$	1.5			V
	V_{D1}	$V_{D1} = V_{DD} - V_1$			$V_{D5} \times 0.25$	V
Input high level voltage	V_{IH1}	except OSCi	2.2		V_{DD}	V
	V_{IH2}	OSCi only	V_{DD} - 1.0		V_{DD}	V
Input low level voltage	V_{IH1}	except OSCi			0.6	V
	V_{IH2}	OSCi only			1.0	V

Note: *1. These voltages guarantee correct operation of the LC7985NA and LC7985ND. They do not guarantee correct operation of the LCD panel. V_{LCD} must also be observed.

Electrical Characteristics at Ta = −20 to +75°C, V_{SS} = 0V, V_{DD} = 5V ± 10%, unless otherwise noted

Parameter	Symbol	Conditions	Ratings			Unit
			min	typ	max	
Output high-level voltage	V_{OH1}	I_{OH} = −0.205mA Input / Output pins	2.4	–	–	V
	V_{OH2}	I_{OH} = −0.04mA Output pins	0.9V_{DD}	–	–	V
Output low-level voltage	V_{OL1}	I_{OL} = 1.2mA Input / Output pins	–	–	0.4	V
	V_{OL2}	I_{OL} = 0.04mA Output pins	–	–	0.1V_{DD}	V
Driver fall voltage*1	V_{COM}	I_d = 0.05mA All common pins	–	–	2.9	V
	V_{SEG}	I_d = 0.05mA All segment pins	–	–	3.8	V
Leakage current	I_L	V_I = V_{SS} to V_{DD}	–	–	1	µA
Pull-up current*2	I_P	V_{DD} = 5V	50	125	250	µA
Current drain	I_{DD1}	Ceramic resonator oscillator, V_{DD} = 5V, f_{OSC} = 250kHz, no output load	–	0.55	0.8	mA
	I_{DD2}	Feedback resistor oscillator, V_{DD} = 5V, f_{OSC} = 270kHz, no output load	–	0.35	0.6	
External clock*3 Frequency	f_{CP}		125	250	350	kHz
Duty cycle	DUTY		45	50	55	%
Rise time	t_R		–	–	0.2	µs
Fall time	t_F		–	–	0.2	µs

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Parameter	Symbol	Conditions	Ratings			Unit
			min	typ	max	
Internal oscillator frequency	f_{OSC1}	Ceramic filter oscillator	245	250	255	kHz
	f_{OSC2}	Feedback resistor oscillator, R_f = 91kΩ ± 3%	190	270	350	
LCD display voltage	V_{LCD1}	1/5 bias, $V_{LCD} = V_{DD} - V_5$	4.6	–	11	V
	V_{LCD2}	1/4 bias, $V_{LCD} = V_{DD} - V_5$	3.0	–	11	

Note: *1. V_{COM} is the voltage from VDD, V1, V4 and V5 to the LCD common drive pins OC1 to OC16.
V_{SEG} is the voltage from VDD, V2, V3 and V5 to the LCD segment drive pins OC1 to OC40.
Note: *2. Applied pins are RS, R/W, and DB0 to DB7.
Note: *3. External clock

Switching Characteristics at Ta = -20 to +75°C, VDD = 5V ± 10%, VSS = 0V

Parameter	Symbol	Conditions	Ratings			Unit
			min	typ	max	
E cycle time	t_{ECYC}		1000	–	–	ns
E high-level pulsewidth	t_{EW}		450	–	–	ns
E rise time	t_{ER}		–	–	25	ns
E fall time	t_{EF}		–	–	25	ns
RS and R/W to E setup time	t_{SU}		140	–	–	ns
E to RS and R/W address hold time	t_{AH}		10	–	–	ns
DB0 to DB7 to E data setup time	t_{DSU}		195	–	–	ns
Write cycle E to DB0 to DB7 data hold time	t_{DHW}		10	–	–	ns
Read cycle E to data valid delay time	t_{DD}	See measurement circuit.	–	–	320	ns
Read cycle E to DB0 to DB7 data hold time	t_{DHR}		20	–	–	ns
CP low-level pulsewidth	t_{WL}		800	–	–	ns
CP high-level pulsewidth	t_{WH}		800	–	–	ns
CP to LOAD setup time	t_{CSU}		500	–	–	ns
D to CP data setup time	t_{DSU}		300	–	–	ns
CP to D data hold time	t_{DH}		300	–	–	ns
LOAD to M delay time	t_{DM}		–1000	–	1000	ns

Reset characteristics at Ta = -20 to +75°C

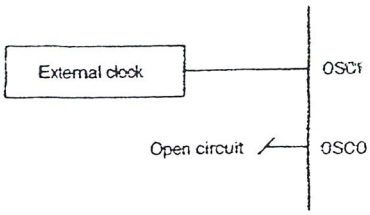
Parameter	Symbol	Conditions	Ratings			Unit
			min	typ	max	
V_{DD} rise time	t_{DDR}		0.1	–	10	ms
V_{DD} off time	t_{DDOFF}		1	–	–	ms

Clock Generator

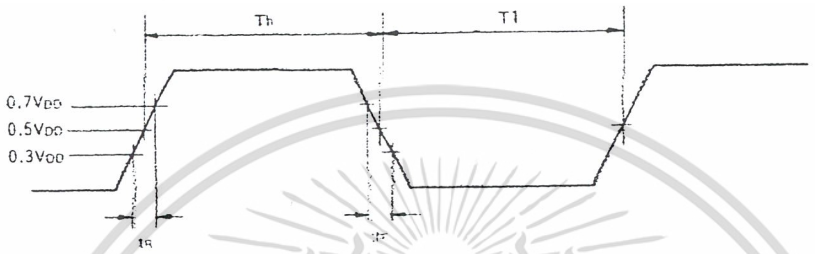
The internal oscillator that generates the clock for the internal circuit requires an external filter, a feedback resistor or an external clock input as shown in the following sections.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

External clock

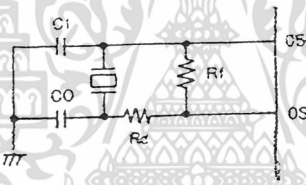


The input duty cycle should be between 45 and 55% as shown in the following figure.



Note. Duty $= \frac{T_h}{T_h + T_l} \times 100\%$

Ceramic filter

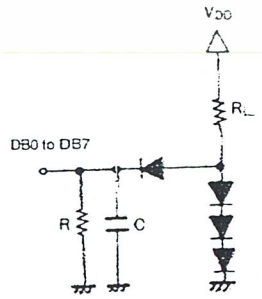


Note. Rf = 1MΩ ± 10%, CI = CO = 680pF ± 10%, Rd = 3.3kΩ ± 5%

Feedback resistor

Note. The resistor should be mounted as close as possible to OSCI and OSCO.

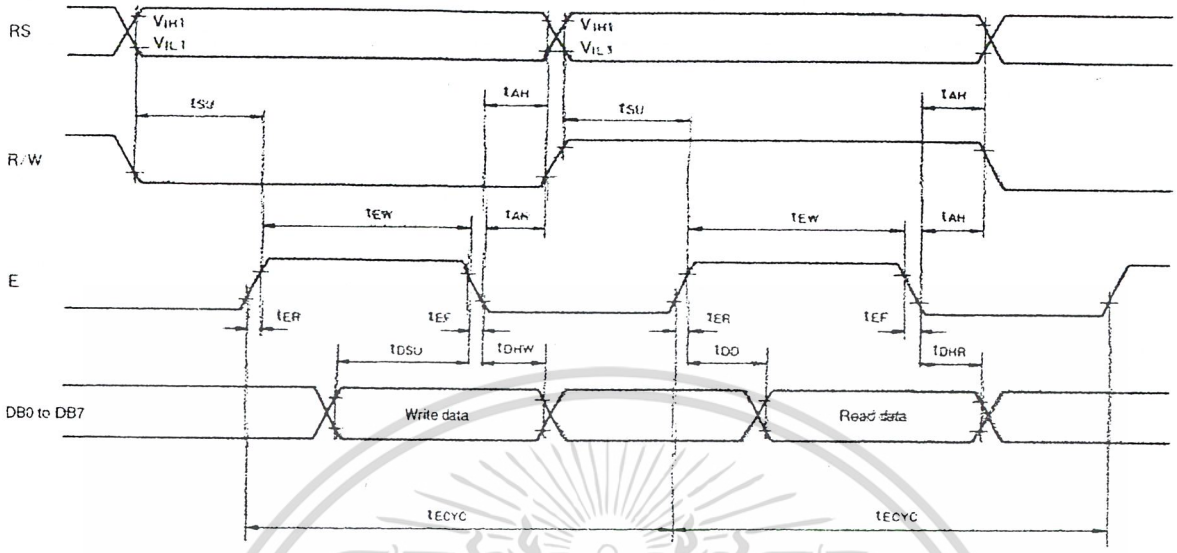
Measurement Circuit



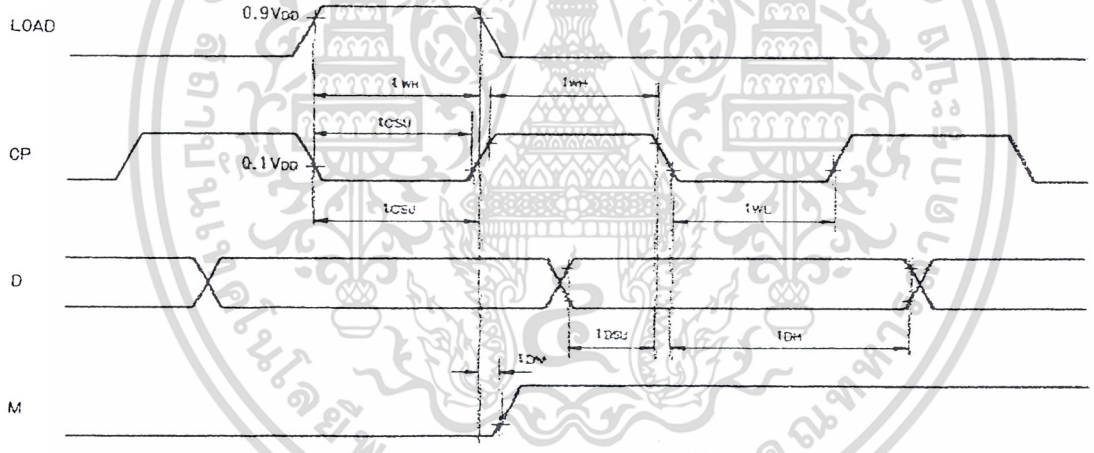
Note. R_L = 2.4kΩ, C = 130pF, R = 11kΩ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

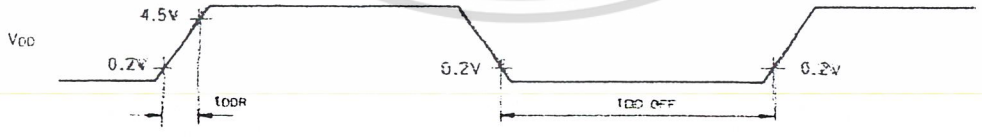
Read/write cycle timing



LC7930N interface timing



Power supply



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Description

Name	Num	I/O	Connect to	Functions
RS	1	I	MPU	Data register or instruction register select input. Data register when "1" and instruction register when "0".
R/W	1	I	MPU	Read or write select input "0" indicates write, "1" ; read
E	1	I	MPU	Execution start input to write or read
DB_4 to DB_7	4	I/O	MPU	4-bit microcontroller interface data bus and 8-bit microcontroller interface high-order four bits data bus connections. Three-state bidirectional. DB_7 can be used as a busyflag.
DB_0 to DB_3	4	I/O	MPU	8-bit microcontroller interface low-order four bits data bus connections. No connection when 4-bit interface size is selected. Three-state bidirectional.
LOAD	1	O	LC7930N	Clock to latch the D serial data output to LC 7930N
CP	1	O	LC7930N	Clock to shift the D serial data
M	1	O	LC7930N	Output to shift the LCD drive signal to alternating current signal
D	1	O	LC7930N	Display expansion serial data output "0" indicates unselected, "1" ; selected
OC_1 to OC_{16}	16	O	LCD	LCD common driver outputs. All common signals unused are unselected wave forms.
OS_1 to OS_{40}	40	O	LCD	LCD segment driver outputs
V_1 to V_5	5		source	Supply voltage for LCD display drive
V_{DD}, V_{SS}	2		source	V_{DD} : +5V, V_{SS} : 0V
OSCI, OSCO	2			Oscillator feedback resistor and ceramic filter connection, and external clock input

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Functional Description

Registers

The LC7985 has two 8-bit registers—instruction register (IR) and data register (DR)—that are selected as shown in the following table.

RS	R/W	Operation
0	0	IR write, instruction execution
0	1	Busy flag (DB7) and address counter (DB0 to DB6) output
1	0	DR write, internal DR to DD RAM or CG RAM data transfer
1	1	DR read, internal DD RAM or CG RAM to DR data transfer

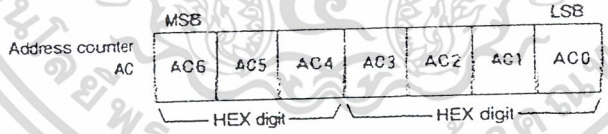
The instruction register is write-only. It contains instruction codes or DD RAM and CG RAM addresses written by the microcontroller.

Busy Flag

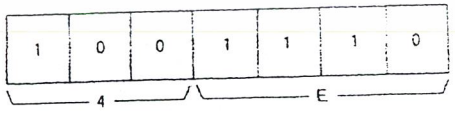
When busy flag is 1, the previous instruction is executing, and when 0, the instruction has completed. The next instruction cannot be received until BF is 0. The microcontroller should, therefore, confirm that BF is 0 before writing the next instruction.

Display Data RAM (DD RAM)

The display data RAM stores 80, 8-bit character codes, and the LC7985 can display a maximum of 80 characters. The address counter contains the location for the next display memory read or write operation as shown in the following figure.



Display data addresses are in hexadecimal. For example, the address counter contents for location 4E are shown in the following figure.



To prevent undesirable effects such as display flicker during DD RAM accesses, the internal memory and the microprocessor interface have separate timing signals.

The data register holds data read from or written to either DD RAM or CG RAM. Data written to the data register by the microcontroller is automatically transferred to the current DD RAM or CG RAM address. Data read from DD RAM or CG RAM is buffered in the data register.

When the microcontroller writes a DD RAM or CG RAM address to the instruction register, the data at that address is copied into the data register. The microcontroller then reads the data in the data register to complete the transfer. Once that data is read, the data from the next DD RAM or CG RAM address is copied into the data register in preparation for the next data read.

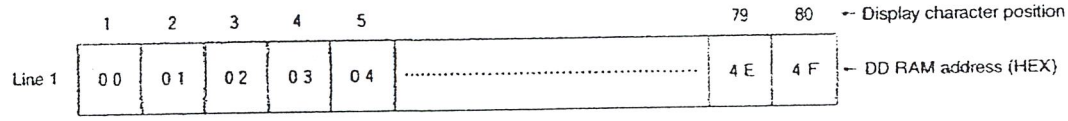
Address Counter

The address counter is used for both the DD RAM and the CG RAM. The address output on DB0 to DB7 is the counter value before the currently executing instruction began.

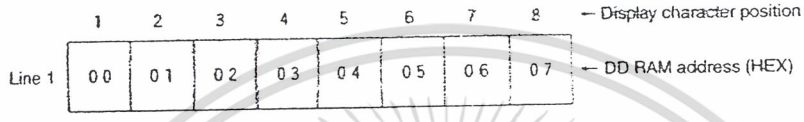
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Single-line display mode (N = 0)

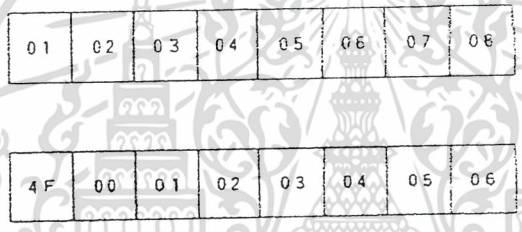
The DD RAM addresses and their corresponding display positions for an 80-character display are shown in the following figure.



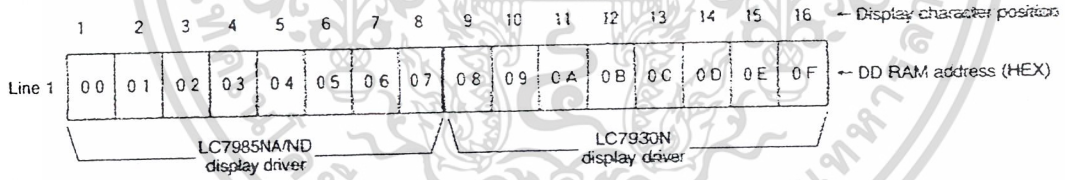
A single LC7985, however, can drive up to eight characters. The display positions and DD RAM addresses for an unshifted 8-character display are shown in the following figure.



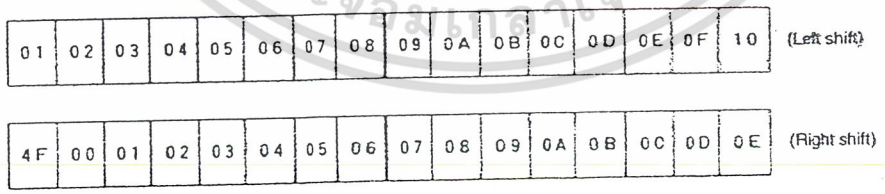
The DD RAM addresses following left and right display shifts are shown in the following figure. Note that the displayed characters wrap around from addresses $4F_H$ to 00_H.



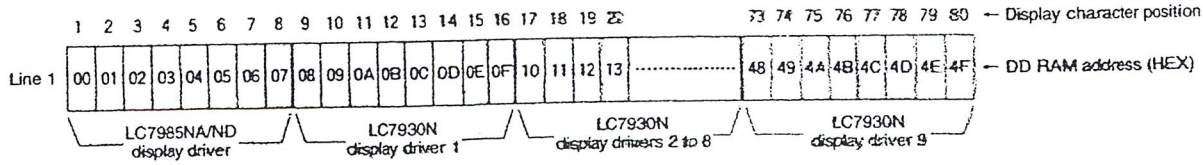
An LC7985 and a single LC7930N can drive a 16-character display. The display positions and DD RAM addresses for an unshifted display are shown in the following figure.



The DD RAM addresses following left and right display shifts are shown in the following figure.



The number of displayed characters can be increased by adding more LC7930Ns. An LC7985 and nine LC7930Ns can drive an 80-character display as shown in the following figure.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Two-line display mode (N = 1)

The DD RAM addresses and their corresponding display positions for a 2-line × 40-character display are shown in the following figure. Note that the address counter automatically increments from 27_H to 40_H.

	1	2	3	4	5	...	39	40	← Display character position
Line 1	00	01	02	03	04	...	26	27	← DD RAM address (HEX)
Line 2	40	41	42	43	44	...	66	67	

A single LC7985, however, can drive up to eight characters per line. The display positions and DD RAM addresses for an unshifted, 2-line × 8-character display are shown in the following figure.

	1	2	3	4	5	6	7	8	← Display character position
Line 1	00	01	02	03	04	05	06	07	← DD RAM address (HEX)
Line 2	40	41	42	43	44	45	46	47	

The display positions following a left or right display shift are shown in the following figure. Note that the display shift is simultaneous for both lines, regardless of which line the cursor is in.

	01	02	03	04	05	06	07	08	(Left shift)
	41	42	43	44	45	46	47	48	
	27	00	01	02	03	04	05	06	(Right shift)
	67	40	41	42	43	44	45	46	

An LC7985 and a single LC7930N can drive a 2-line × 16-character display. The display positions and DD RAM addresses for an unshifted, 2-line × 16-character display are shown in the following figure.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	← Display character position
Line 1	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	← DD RAM address (HEX)
Line 2	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	

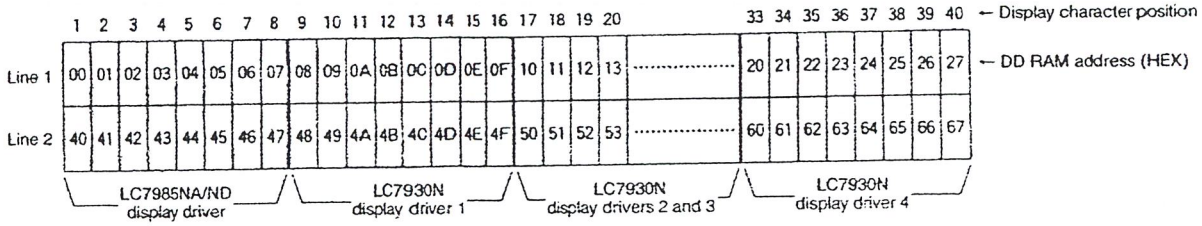
LC7985NA/ND display driver | LC7930N display driver

The DD RAM addresses following left and right display shifts are shown in the following figure.

	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	(Left shift)
	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	50	
	27	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	(Right shift)
	67	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The number of displayed characters can be increased by adding more LC7930Ns. An LC7985 and four LC7930Ns can drive a 2-line × 40-character display as shown in the following figure.



Character Generator ROM (CG ROM)

The character generator ROM contains 160, 5 × 7-pixel bitmaps and 32, 5 × 10-pixel bitmaps as shown in the following figure. The characters are selected by their 8-bit character code.

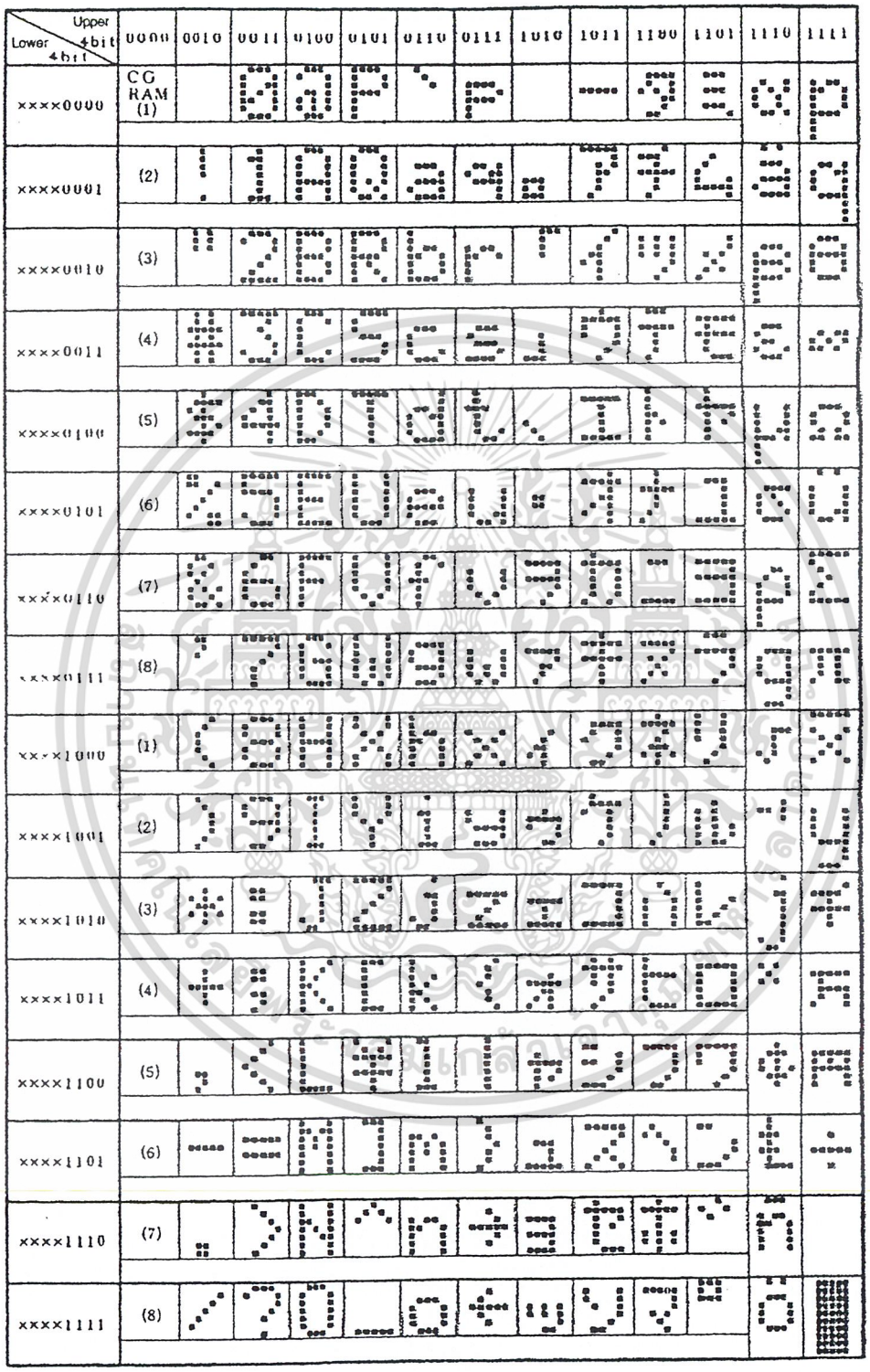
Character Generator RAM (CG RAM)

The character generator RAM stores user-defined bitmaps for either eight, 5 × 7-pixel characters or four, 5 × 10-pixel characters. To display character patterns stored in CG RAM, write the character codes, shown in the leftmost column of the following figure, on DD RAM.

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Character cord and the character bitmap



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5 × 7-pixel characters

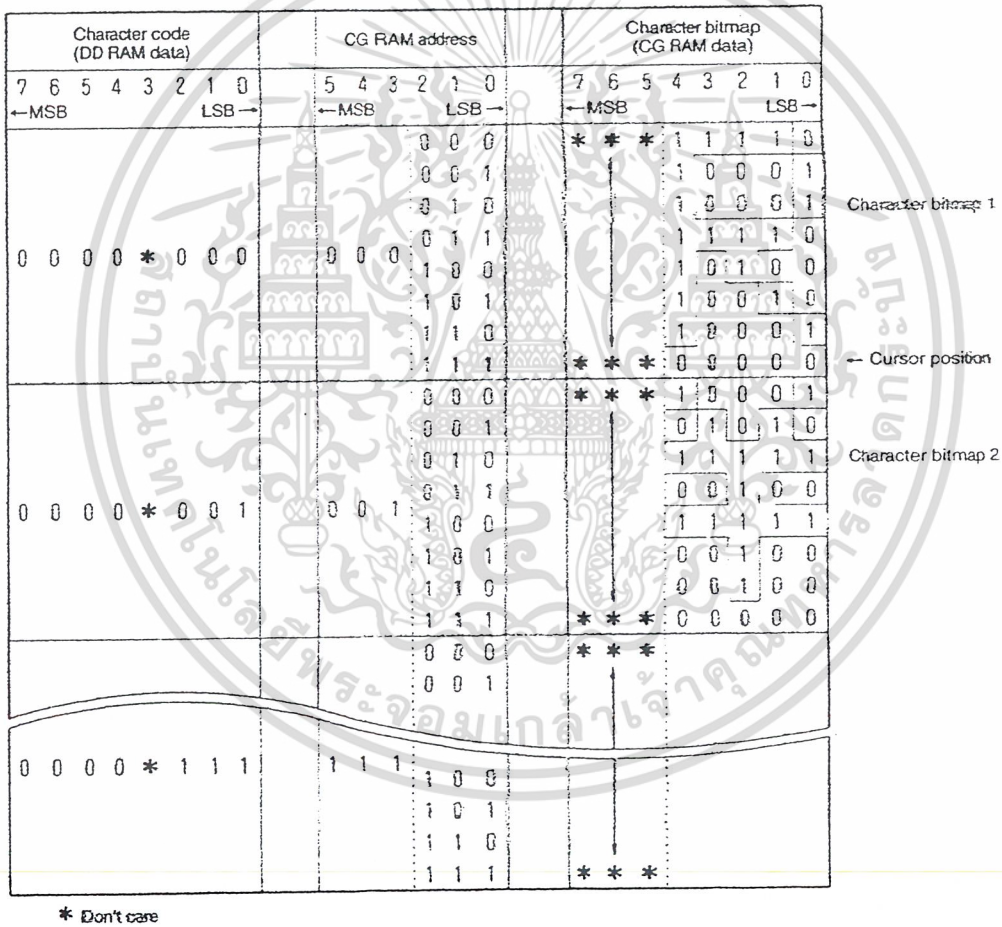
The layout and addressing for 5 × 7-pixel characters is shown in the following figure. Each character occupies eight bytes, where bits 3 to 5 of the CG RAM address correspond to bits 0 to 2 of the character code. Note that bit 3 of the character code is not significant so, for example, codes 00_H and 08_H select the same character.

Bits 0 to 2 of the CG RAM address are the bitmap row address, where row 000 is the topmost displayed row.

The cursor, when displayed, is formed by ORing the bottom row with all 1s. If the cursor is used, row 111 should contain all 0s so the cursor does not obscure the bottom row of the character.

Bits 0 to 4 of the CG RAM data contain the character bitmaps. When a bit is 1, the corresponding pixel is ON, and when 0, the pixel is OFF.

Bits 5 to 7 of the CG RAM data are present in memory, but are not used by the display circuit. These bits can be used as general-purpose RAM.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5 × 10-pixel characters

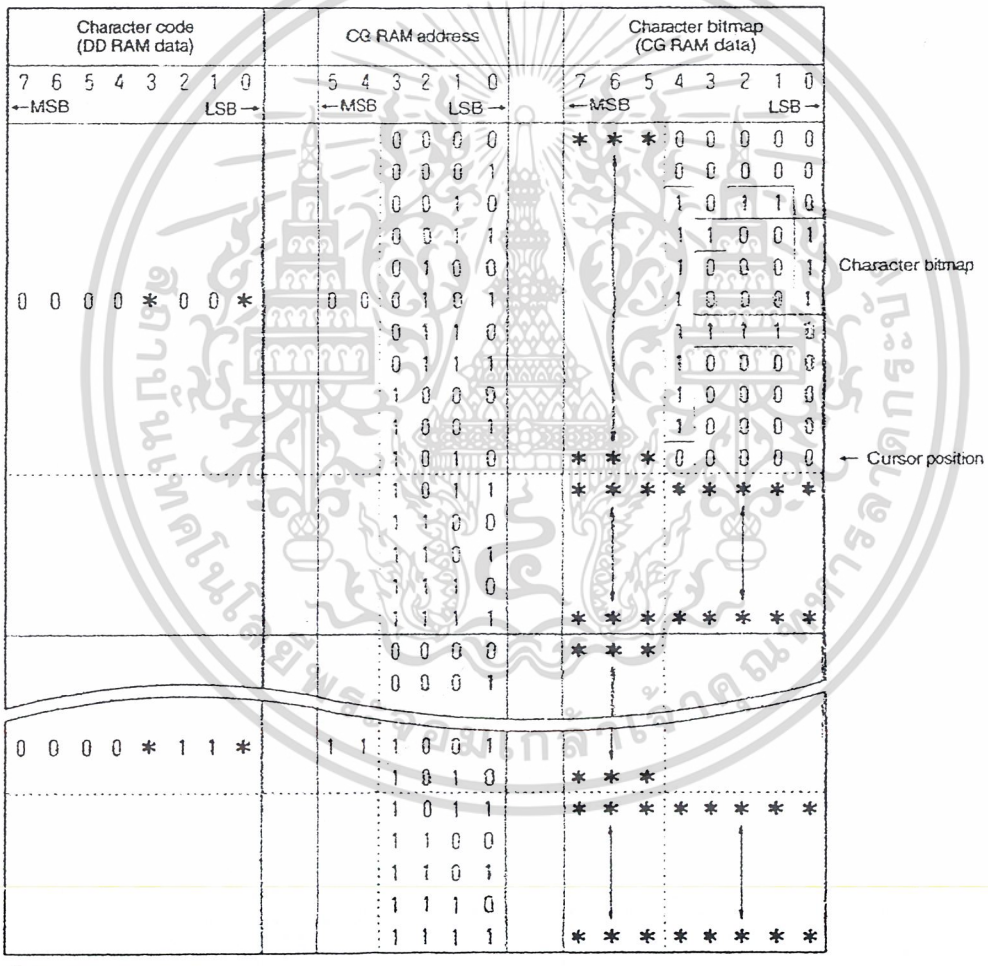
The layout and addressing for 5 × 10-pixel characters is shown in the following figure. Each character occupies eleven bytes, where bits 4 and 5 of the CG RAM address correspond to bits 1 and 2 of the character code. Note that bits 0 and 3 of the character code are not significant so, for example, codes 00_H, 01_H, 08_H and 09_H all select the same character.

Bits 0 to 3 of the CG RAM address are the bitmap row address where row 000 is the topmost displayed row.

The cursor, when displayed, is formed by ORing the bottom row with all 1s. If the cursor is used, row 1010 should contain all 0s so the cursor does not obscure the bottom row of the character.

Bits 0 to 4 of the CG RAM data contain the character bitmaps. When a bit is 1, the corresponding pixel is ON, and when 0, the pixel is OFF.

Bits 5 to 7 of the CG RAM data are present in memory, but are not used by the display circuit. These bits and the CG RAM bytes, rows 1011 to 1111 that are not used by the display circuit, can be used as general-purpose RAM.



Timing Generator

This circuit generates timing signals both for internal circuit operation and for driving external LC7930Ns. The timing signals for the DD RAM, CG ROM and CG RAM are independent of the microcontroller interface so that memory accesses by the microcontroller do not cause interference with the display drive signals.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Display Drivers

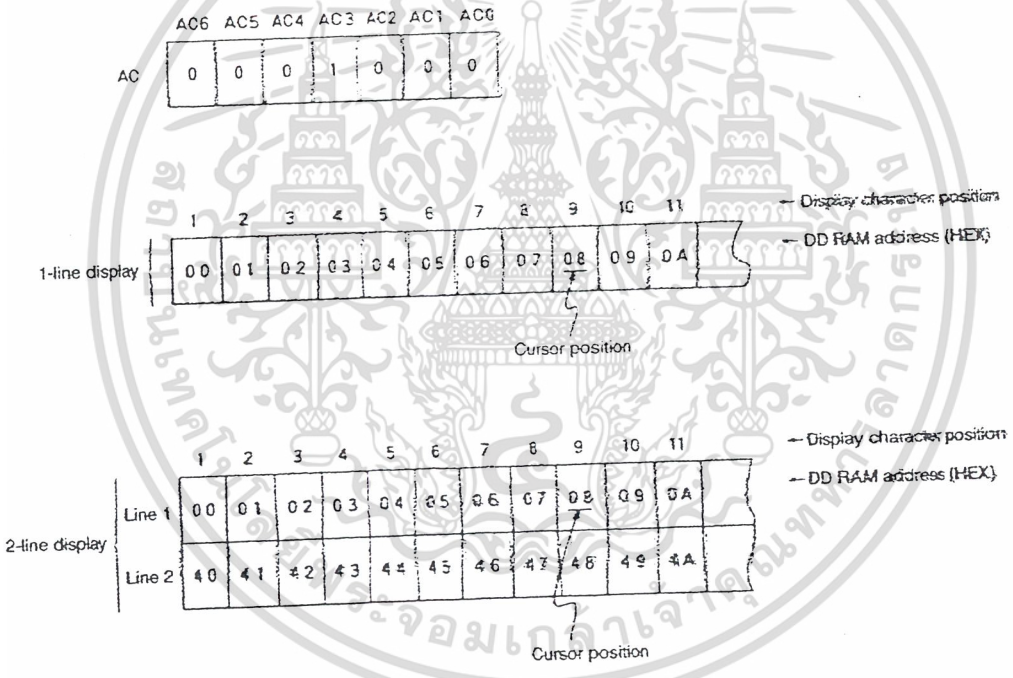
The LC7985 incorporates 16 LCD common driver outputs and 40 LCD segment driver outputs. The character font and the number of display lines determine the number of active common outputs.

The segment drivers function identically to the LC7930N display drivers. The character bitmap data to be displayed is latched in the internal 40-bit shift register before being output on the segment drivers.

The display bitmap data for each pixel-row is generated starting with the right-most character position. The data shifts through the shift register and is output on the shift register serial data output. The shift register latches the last 40 bits in the row so the LC7985 displays the left-most eight characters. External LC7930Ns connect in series to the serial data output and each one latches and displays bitmap data for eight additional characters.

Cursor Display and Blinking

Cursor display and blinking of the character at the cursor position are controlled using the Display ON/OFF instruction. The cursor position is at the character corresponding to the address counter value as shown in the following figure. Note that the cursor and blinking character are also displayed at the address counter value when CG RAM is selected.

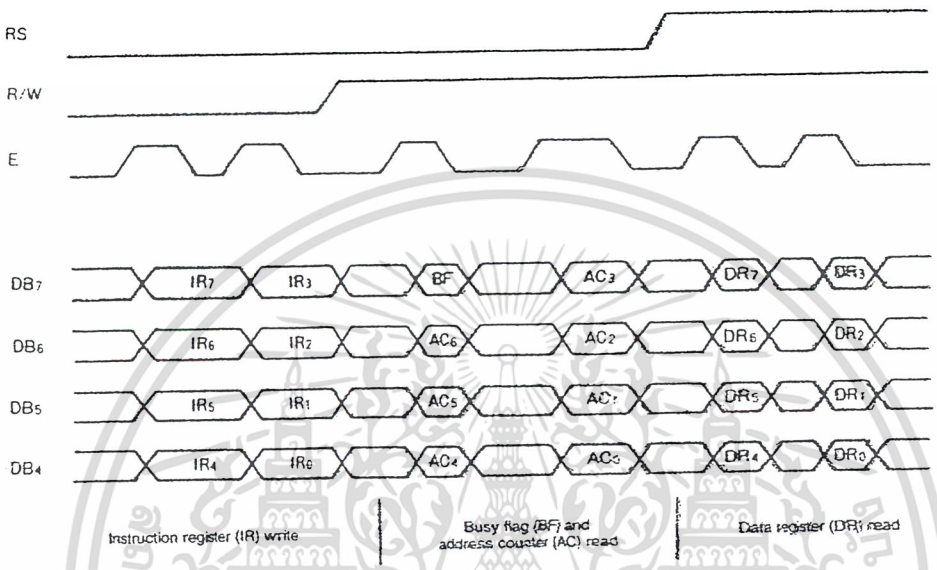


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Microcontroller Interface

The LC7985 interfaces to both 4-bit and 8-bit microcontrollers.

DB0 to DB7 are used for the 4-bit data bus. Two read or write cycles, therefore, are required to transfer each data, status or instruction byte. The high-order four bits—bits DB4 to DB7 in 8-bit interface mode—are transferred first. The low-order four bits are then transferred as shown in the following figure.



Reset Circuit

The internal reset circuit initializes the LC7985 at power-ON. The busy flag remains ON from power-ON until initialization is complete 10ms after V_{DD} reaches 4.5V. Note that if power supply conditions are such that the internal reset circuit does not operate to initialize the device, the LC7985 must be initialized using commands from the microcontroller.

The initialization sequence is as follows.

1. Clear Display
2. Set Function (D/L = 1, N = 0, F = 0)
   Sets 8-bit interface size, 1-line display size and 5 × 7-pixel character font.
3. Cursor/Display Control (D = 0, C = 0, B = 0)
   Sets the display, the cursor and character blinking OFF.
4. Set Entry Mode (I/D = 1, S = 0)
   Sets address counter auto-increment and sets display shift OFF.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Instructions

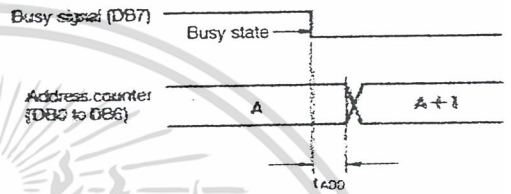
The external microcontroller accesses two register—instruction register and data register—to control the LC7985. So the microcontroller interface is independent of the microcontroller clock frequency, the LC7985 stores the instruction of data internally before executing it.

There are four types of instructions.

- Function set instructions such as display type or interface size set
- Address set instructions
- Data read and write instructions
- Other instructions

The Busy Flag/Address Read instruction is the only instruction that can be executed while the LC7985 is executing a previous instruction. Before transmitting any other instruction, the microcontroller should either check that the busy flag is OFF or else wait longer than the execution time of the previous instruction.

Data read and write instructions are usually the most frequently used instructions. For increased microcontroller efficiency, a display shift and display data write can be executed simultaneously. In addition, the address counter automatically increments or decrements after either a data read or data write instruction, which reduces the operations required by the microcontroller. Note that the increment or decrement occurs after the busy flag turns OFF. The delay until the address counter updates is $t_{ADD} = 1.5/f_{CP}$ or $t_{ADD} = 1.5/f_{OSC}$, and is shown in the following figure.



The instructions are shown in the following table. The instruction code comprises the RS, R/W and DB0 to DB7 signals.

Instruction	Code										Description	Execution time* (max) (f_{CP} or f_{OSC} = 250kHz)
	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0		
Display Clear	0	0	0	0	0	0	0	0	0	1	Clears the display and sets the address counter to DD RAM address 0.	1.64ms
Cursor Home	0	0	0	0	0	0	0	0	1	×	Sets the address counter to DD RAM address 0. Returns a shifted display to the original position. Does not alter the DD RAM data.	1.64ms
Set Entry Mode	0	0	0	0	0	0	0	1	I/D	S	Sets cursor movement and display shift following a data read or write. When I/D is 1, the cursor increments, and when 0, decrements. When S is 1, the display also shifts.	40µs
Display ON/OFF	0	0	0	0	0	0	1	D	C	B	When D is 1, the display is ON, and when 0, OFF. When C is 1, the cursor is ON, and when 0, OFF. When B is 1, blinking of the character at the cursor position is ON, and when 0, OFF.	40µs
Cursor/Display Shift	0	0	0	0	0	1	S/C	R/L	×	×	Moves the cursor or the display without altering the DD RAM data. When S/C is 1, the display shifts, and when 0, the cursor moves. When R/L is 1, the direction is right, and when 0, left.	40µs
Set Function	0	0	0	0	1	DL	N	F	×	×	When DL is 1, the interface size is eight bits, and when 0, four bits. When N is 1, the display size is two lines, and when 0, a single line. When F is 1, the font size is 5 × 10 pixels, and when 0, 5 × 7 pixels.	40µs
Set CG RAM Address	0	0	0	1	CG RAM address						Sets the CG RAM address. Data read and writes after this instruction are to and from CG RAM.	40µs
Set DD RAM Address	0	0	1	DD RAM address							Sets the DD RAM address. Data read and writes after this instruction are to and from DD RAM.	40µs
Busy Flag/Address Read	0	1	BF	Address counter							Used during execution of other instructions, outputs the busy flag state and the address counter value. The address counter is used for both DD RAM and CG RAM.	0µs
Data Write	1	0	Write data								Writes data to DD RAM or CG RAM.	40µs (t_{ADD} = 6µs)
Data Read	1	1	Read data								Reads data from DD RAM or CG RAM.	40µs (t_{ADD} = 6µs)
	I/D =1: increment S =1: accompanied by display shift S/C=1: display shift R/L =1: right shift DL =1: 8-bit N =1: two rows F =1: 5× 10-pixel characters BF =1: internally operating					I/D =0: decrement S/C =0: cursor shift R/L =0: left shift DL =0: 4-bit N =0: a row F =0: 5× 7-pixel characters BF =0: open to instructions					DD RAM : display data RAM CG RAM : character generator RAM A_{CG} : CG RAM address A_{DD} : DD RAM address: corresponding to cursor address AC : address counter used for both DD RAM and CG RAM	

Note: *1.The execution time depends on the operating frequency. For example, if f_{CP} or f_{OSC} = 270kHz, the execution time is 40µs × 250/270 = 37µs.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Display Clear

	RS	R/W	DB7						DB0
Code	0	0	0	0	0	0	0	0	1

Fills the DD RAM with space characters (20_H), returns the display to the unshifted position and sets the address counter to zero, returning the cursor to the top-left display position. The address counter increment/decrement mode is set to increment. The character blinking and display shift modes are not affected.

Note that if a custom character generator ROM is used, the space character must correspond to the 20_H character code for the display to be cleared correctly.

Cursor Home

	RS	R/W	DB7						DB0
Code	0	0	0	0	0	0	0	1	*

* Don't care

Returns the display to the unshifted position and sets the address counter to zero, returning the cursor to the top-left display position. Does not alter the DD RAM data.

Set Entry Mode

	RS	R/W	DB7						DB0
Code	0	0	0	0	0	0	0	I/D	S

Sets the cursor auto-increment direction and the display shift mode and direction. When I/D is 1, the address counter increments when data is read from or written to either the DD RAM or the CG RAM, thereby shifting the cursor right one character position. When I/D is 0, the address counter decrements, shifting the cursor left.

When S is 1, display shift is ON, and the display also shifts one character position to the right or left when data is written to the DD RAM so that the cursor position relative to the display is unchanged. No display shift occurs when data is read from the DD RAM or when data is read from or written to the CG RAM, although the address counter increments or decrements for all read and write operations. When S is 0, display shift is OFF.

Display ON/OFF

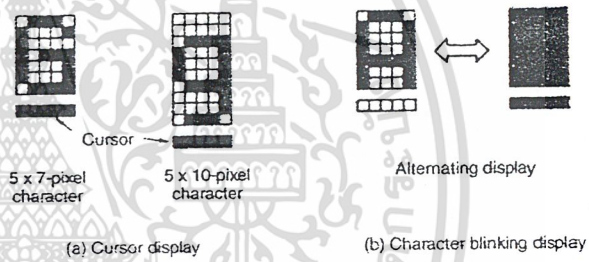
	RS	R/W	DB7						DB0
Code	0	0	0	0	0	0	1	D	C	B

Sets the display, the cursor and character blinking ON or OFF.

When D is 1, the display is ON, and when 0, OFF. Setting the display ON or OFF does not alter the address counter or the DD RAM data.

When C is 1, the cursor is ON, and when 0, OFF. Setting the cursor ON or OFF does not affect the cursor auto-increment and display shift modes.

When B is 1, the cursor and the character at the cursor position blink, alternating between black (all pixels ON) and the displayed character as shown in the following figure. When f_{CP} or f_{OSC} = 250kHz, the blink interval is 409.6ms, and when f_{CP} or f_{OSC} = 270kHz, 379.2ms.



Cursor/Display Shift

	RS	R/W	DB7						DB0
Code	0	0	0	0	0	1	S/C	R/L	*	*

* Don't care

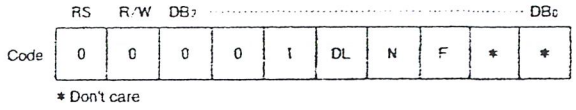
Shifts the cursor or the display either left or right as shown in the following table. A DD RAM write is not required.

When shifting a 2-line display, both rows shift simultaneously, but characters do not move from one row to another. Each time the display shifts, the characters in each row only move within the row.

S/C	R/L	Description
0	0	Decrements the address counter and shifts the cursor left.
0	1	Increments the address counter and shifts the cursor right.
1	0	Shifts the display left. The address counter does not change, and the cursor moves with the display.
1	1	Shifts the display right. The address counter does not change, and the cursor moves with the display.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Set Function



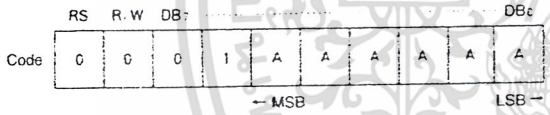
Sets the microcontroller interface bus size and the display mode. When DL is 1, the interface size is eight bits, and when 0, four bits. When the interface size is four bits, two reads or writes of the high-order bits of the data bus, DB4 toB7, are required.

N and F set the display mode as shown in the following table. N sets the number of lines in the display, and F, the font size. Note that a 2-line display cannot use the 5 × 10-pixel font size.

N	F	Display lines	Font size (pixels)	Duty
0	0	1	5×7	1/8
0	1	1	5×10	1/11
1	×	2	5×7	1/16

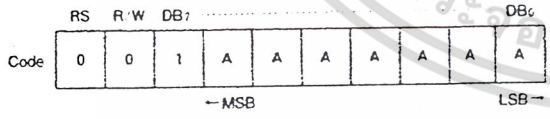
Note that the font size and number of lines cannot be changed once any other instruction is executed following the Set Function instruction.

Set CG RAM Address



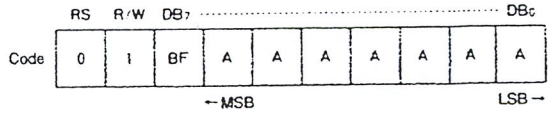
Loads the 6-bit character generator RAM address into the address counter. Data reads and writes after this instruction is executed are to and from the CG RAM.

Set DD RAM Address



Loads the 7-bit display data RAM address into the address counter. Data reads and writes after this instruction is executed are to and from the DD RAM.

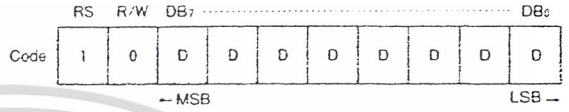
Busy Flag/Address Read



Outputs the busy flag state and the address counter value. The busy flag is used to check if the previous instruction has finished executing. When BF is 1, the previous instruction is executing, and when 0, the instruction has completed. The next instruction cannot be received until BF is 0. The microcontroller should, therefore, confirm that BF is 0 before writing the next instruction.

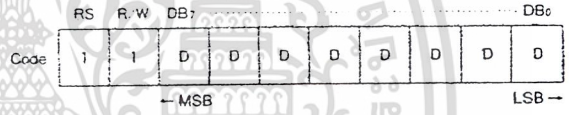
The address counter is used for both the DD RAM and the CG RAM. The address output on DB0 to DB7 is the counter value before the currently executing instruction began.

Data Write



Writes the 8-bit data on DB0 to DB7 to either the DD RAM or the CG RAM, according to whether a Set DD RAM Address or a Set CG RAM Address instruction was executed previously. After writing, the address counter automatically increments or decrements according to the entry mode setting, and the display can also shift.

Data Read



Outputs 8-bit data on DB0 to DB7 from either the DD RAM or the CG RAM, according to whether a Set DD RAM Address or a Set CG RAM Address instruction was executed previously. After the data is read, the address counter automatically increments or decrements according to the entry mode setting, but the display does not shift.

Note that a Set DD RAM Address or Set CG RAM Address instruction should be executed before executing this command. If a Data Read instruction is executed without first executing an address set instruction, the output data will not be valid. If the instruction is repeated, however, the output data will be valid data from the next address. Subsequent Data Read instructions will output valid data.

The output data will not be valid if this command is executed following a Data Write command, even though the address counter has just incremented or decremented.

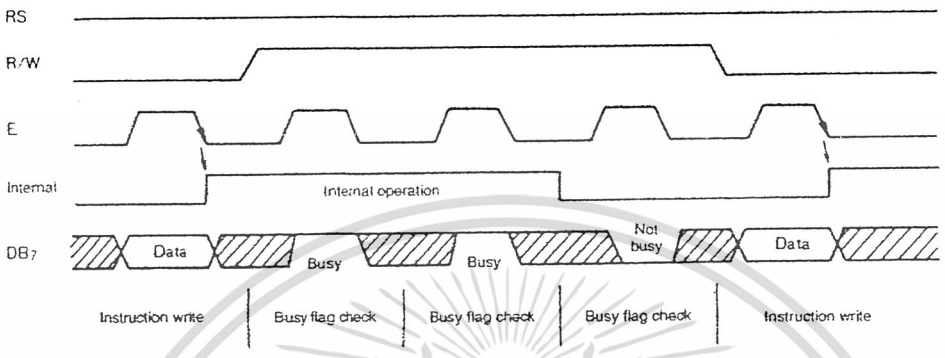
A Cursor/Display Shift instruction has the same effect as a Set DD RAM Address instruction. If a Cursor/Display Shift instruction moves the cursor, an address set instruction does not have to be executed before the Data Read instruction, and the data is read from the DD RAM.

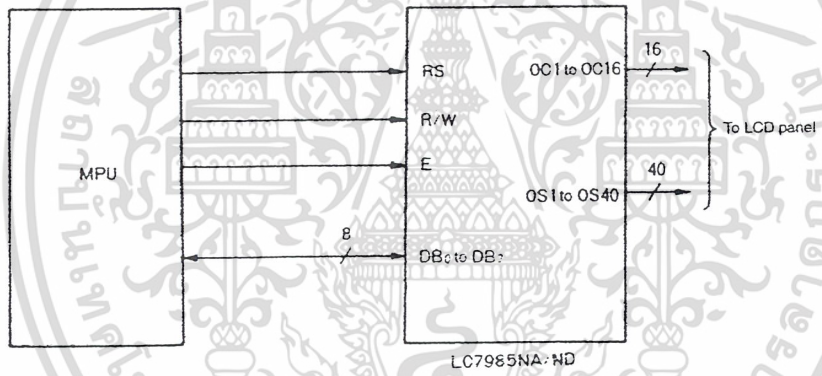
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Microcontroller Interface

8-bit interface

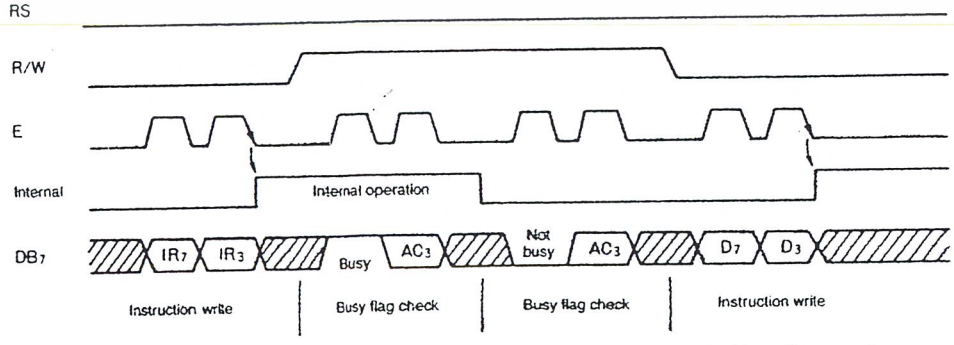
DB0 to DB7 are used for the 8-bit data bus. The timing sequence for instruction write, instruction execution, and busy flag checking is shown in the following figure.





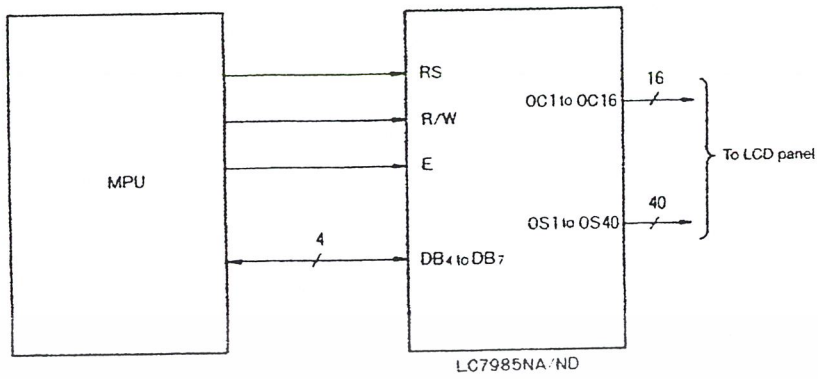
4-bit interface

The timing sequence for instruction write, instruction execution and busy flag checking is shown in the following figure. The busy flag is checked after transferring two 4-bit sets of data. The busy flag and address counter value are output as two 4-bit words. Checking the busy flag, therefore, requires two read cycles so the low-order four bits of the address counter value are flushed from the data buffer.



Note. IR7 and IR3 are the 7th and 3rd bit, respectively, of the instruction. AC3 is the 3rd bit of the address counter.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



LCD Interface

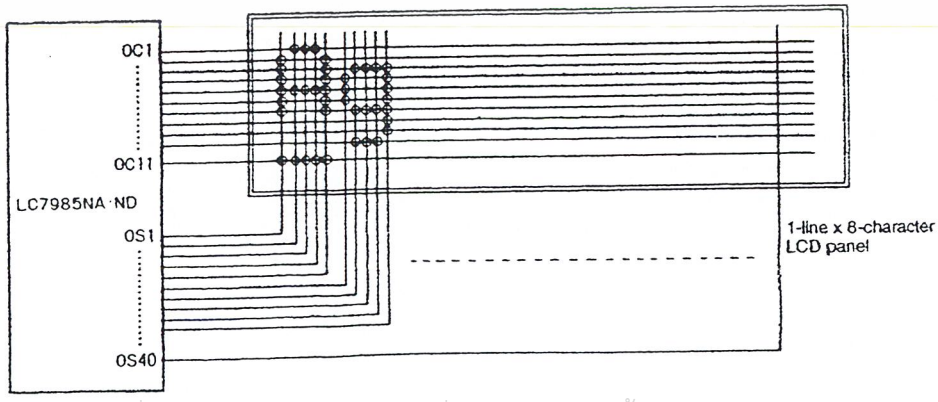
The number of common signals and the duty cycle for each combination of font and display lines are shown in the following table. One common signal is required for each pixel-row in the character, and an additional common signal is required for the cursor row beneath the character.

Display lines	Font size	Common signals	Duty
1	5 × 7-pixel + cursor	8	1/8
1	5 × 10-pixel + cursor	11	1/11
2	5 × 7-pixel + cursor	16	1/16

Sample Application Circuits

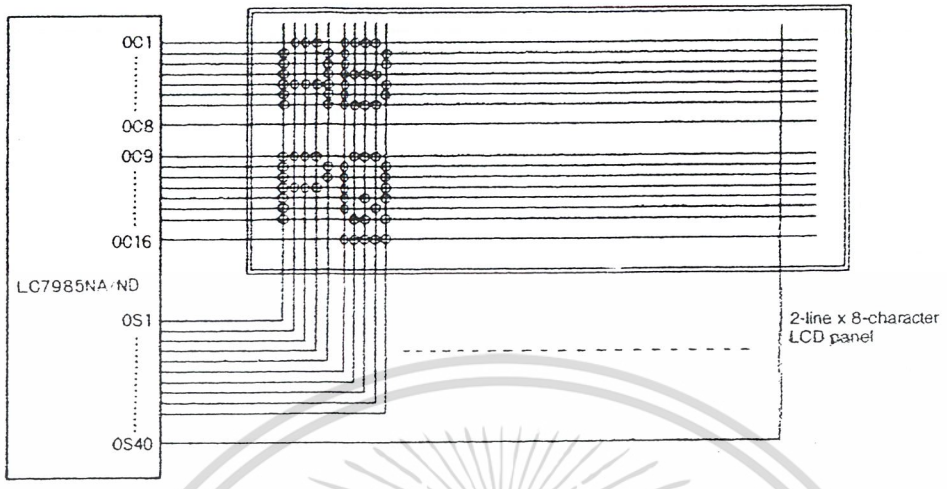
1-line × 8-character, 1/4-bias and 1/8-duty Display with 5 × 7-pixel Font

1-line × 8-character, 1/4-bias and 1/11-duty Display with 5 × 10-pixel Font



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

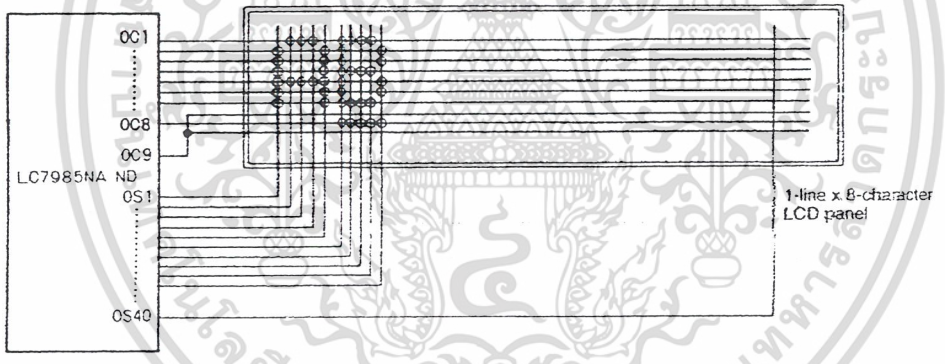
2-line × 8-character, 1/5-bias and 1/16-duty Display with 5 × 7-pixel Font



Connecting Unused Display Rows

Connecting unused LCD panel common pins to an unused LC7985 common output pin as shown in the following figure prevents crosstalk from the active drive signals affecting the display.

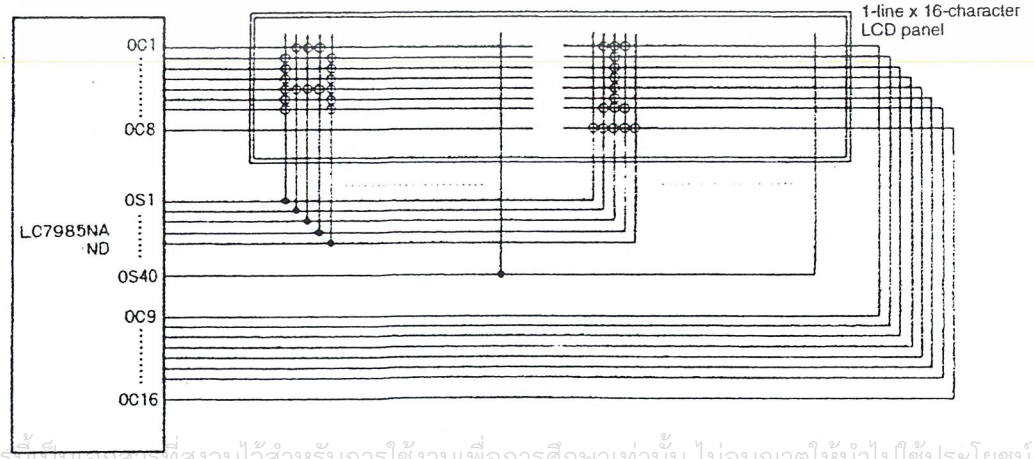
1-line × 8-character, 1/4-bias and 1/8-duty Display with 5 × 7 pixel Font



Alternative Display Connections

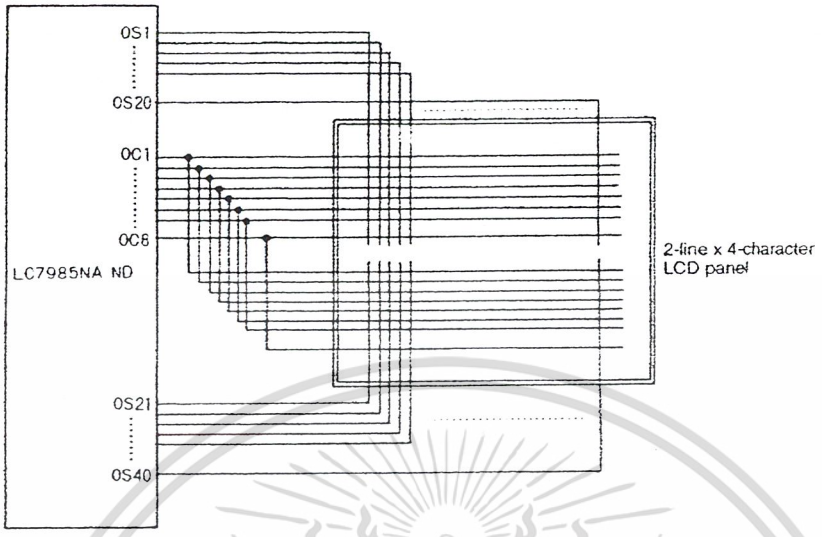
The LC7985 to LC7985 panel connections can be varied to match the LCD panel matrix as shown in the following sections.

1-line × 16-character, 1/8-bias and 1/16-duty Display with 5 × 7 pixel Font



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2-line × 4-character, 1/4-bias and 1/8-duty Display with 5 × 7 pixel Font



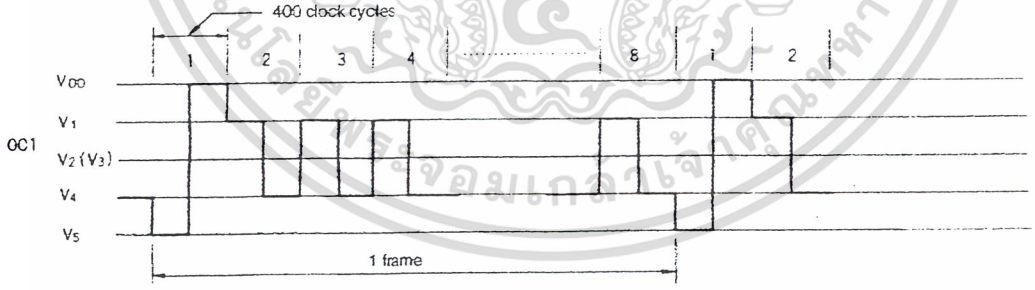
LCD driver power supply

The reference voltage levels required to generate the LCD drive waveforms are shown in the following table.

Voltages V_1 to V_5 are input on pins V1 to V5, respectively. The voltages can be produced using a voltage-divider resistor network. The voltages required depend upon the duty cycle. V_{LCD} is the LCD driver peak voltage, where $V_{LCD} = V_{DD} - V_5$.

Voltage	1/4 bias and 1/8 or 1/11 duty	1/5 bias and 1/16 duty
V_1	$V_{DD} - 0.25V_{LCD}$	$V_{DD} - 0.2V_{LCD}$
V_2	$V_{DD} - 0.5V_{LCD}$	$V_{DD} - 0.4V_{LCD}$
V_3	$V_{DD} - 0.5V_{LCD}$	$V_{DD} - 0.6V_{LCD}$
V_4	$V_{DD} - 0.75V_{LCD}$	$V_{DD} - 0.8V_{LCD}$
V_5	$V_{DD} - V_{LCD}$	$V_{DD} - V_{LCD}$

1/8 duty LCD drive

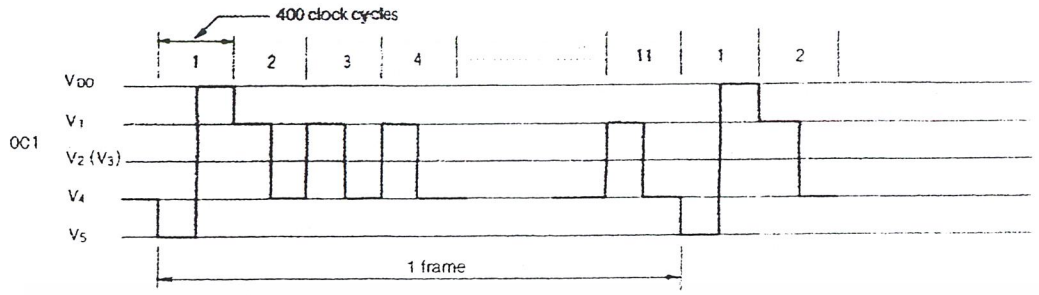


$$1\ \text{frame} = 4\ \mu s \times 400 \times 8 = 12800\ \mu s = 12.8\ ms$$

$$\text{Frame frequency} = \frac{1}{12.8} = 78.1\ Hz$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

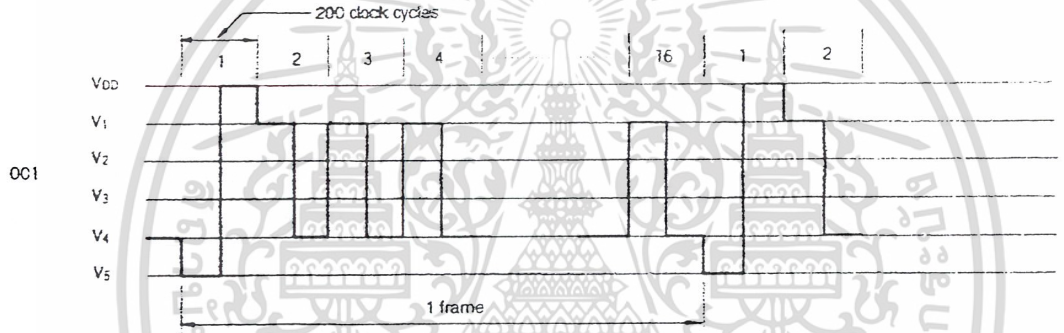
1/11 duty LCD drive



$$1\ \text{frame} = 4\ (\mu s) \times 400 \times 11 = 17600\ (\mu s) = 17.6\ \text{ms}$$

$$\text{Frame frequency} = \frac{1}{17.6\ (\text{ms})} = 56.8\ (\text{Hz})$$

1/16 duty LCD drive



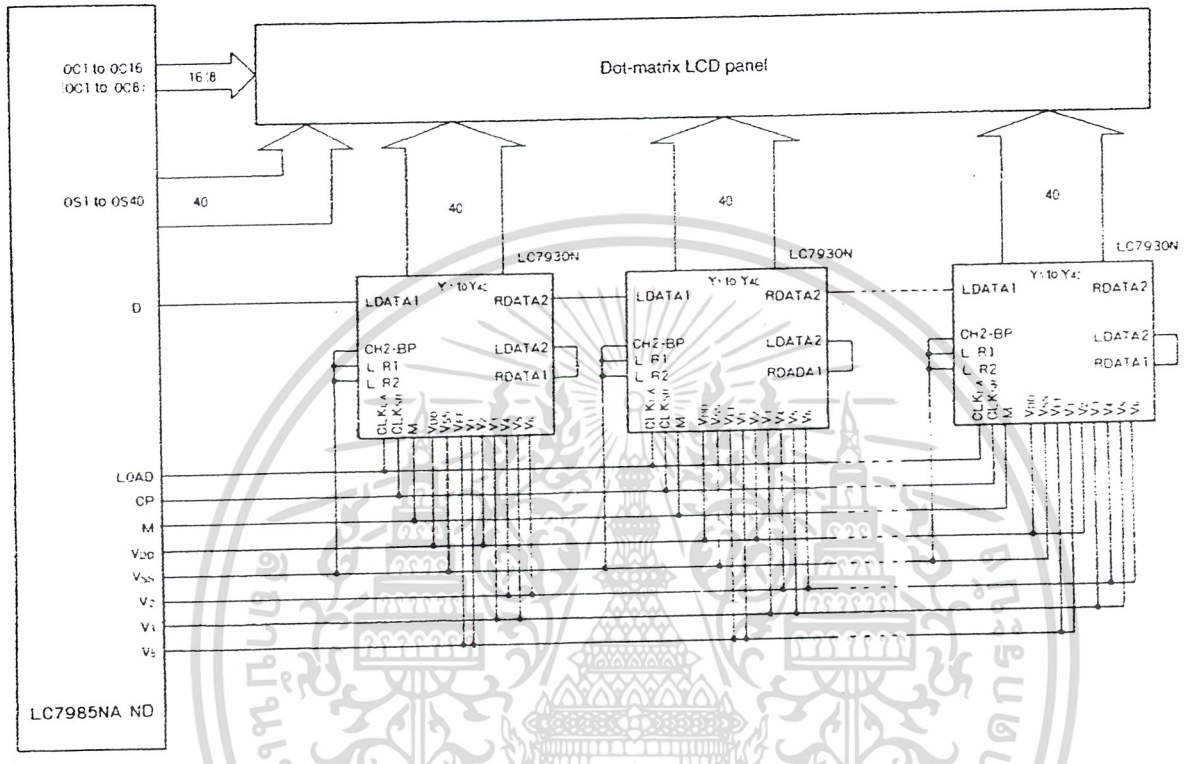
$$1\ \text{frame} = 4\ \mu s \times 200 \times 16 = 12800\ \mu s = 12.8\ \text{ms}$$

$$\text{Frame frequency} = \frac{1}{12.8\ \text{ms}} = 78.1\ \text{Hz}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LC7930N Interface

When using a single-line display, up to nine LC7930Ns, and when using a two-line display, up to four LC7930Ns can interface to the LC7985 using the circuit shown in the following figure. The LC7985 LOAD, CP, M and D outputs connect directly to the LC7930Ns. Take care that the V1 to V5 voltage reference outputs are connected correctly to the LC7930Ns.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Examples

8-bit interface size, 1-line × 8-character display and internal reset circuit

The programming example is shown in the following table. This example assumes that the internal reset circuit initializes the LC7985.

The Set Function instruction that is executed before the display is turned ON determines the operation of the device.

Since the DD RAM stores 80 characters, the display shift function can be used as shown in the example. Note that display shifts only change the display position and do not alter the DD RAM. Using the Cursor Home instruction, therefore, returns the display to its original position.

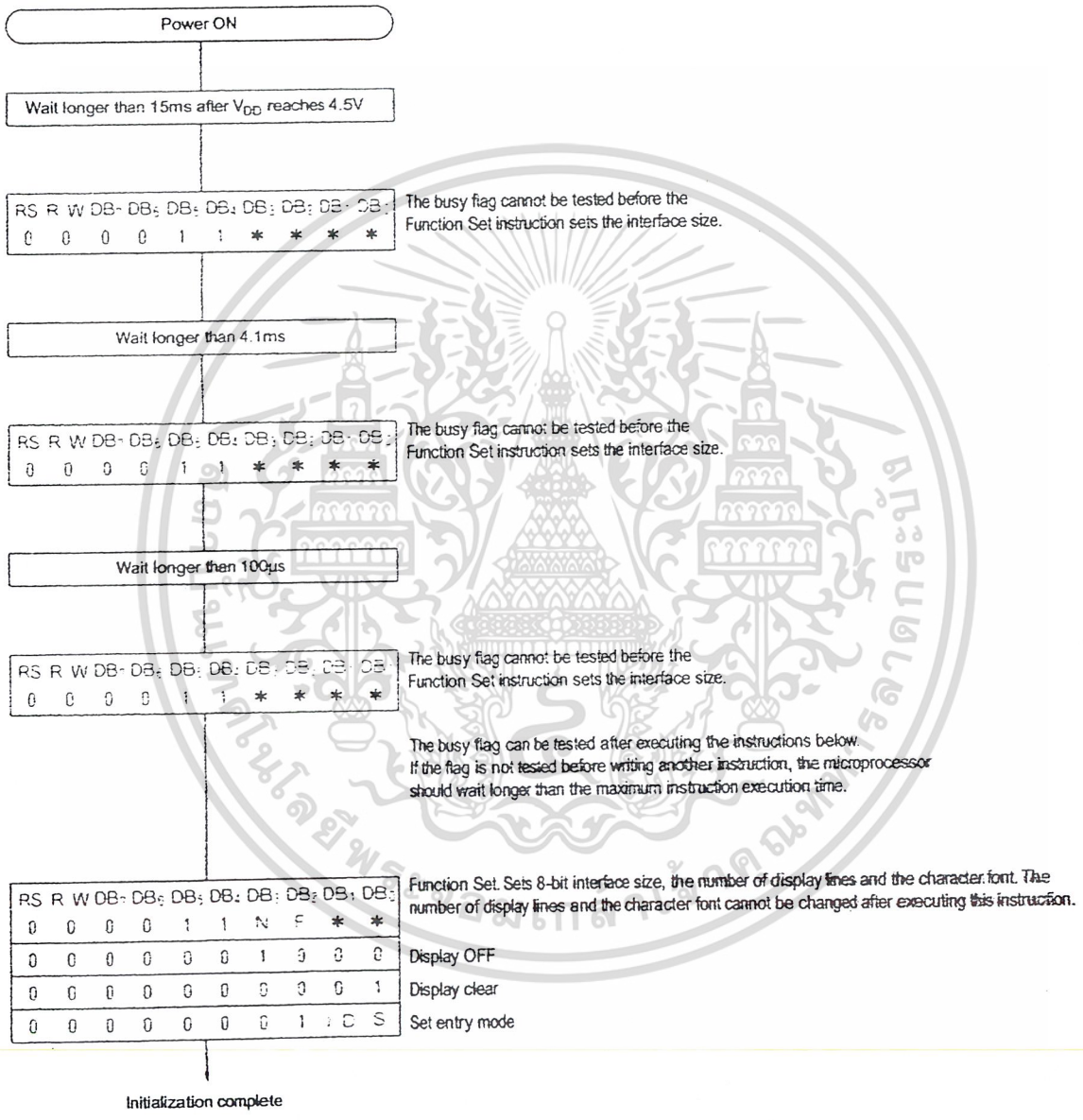
Instruction	Code										Display	Description
	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0		
Power-ON												The internal reset circuit initializes the LC7985. The display is OFF.
Set Function	0	0	0	0	1	1	0	0	×	×		Sets 8-bit interface size, 1-line display size and 5 × 7-pixel character font. The number of display lines and the character font cannot be changed later.
Display ON/OFF	0	0	0	0	0	0	1	1	1	0	_	Turns the display ON and enables the cursor. The display is blank.
Set Entry Mode	0	0	0	0	0	0	0	1	1	0	_	Sets address auto-increment and automatic cursor right shift on writing to DD RAM or CG RAM. The display is not shifted.
Data Write	1	0	0	1	0	0	1	1	0	0	L_	Writes 'L' to DD RAM, since DD RAM was selected when the LC7985 was initialized at power-ON. The cursor position increments and the cursor moves right.
Data Write	1	0	0	1	0	0	0	0	1	1	LC_	Writes 'C'.
Data Write	↓										↓	↓
Data Write	1	0	0	0	1	1	0	1	0	1	LC7985_	Writes '5'.
Set Entry Mode	0	0	0	0	0	0	0	1	1	1	LC7985_	Sets display shift on writing to DD RAM.
Data Write	1	0	0	0	1	0	0	0	0	0	C7985_ _	Writes a space ' '.
Data Write	1	0	0	1	0	0	1	1	0	0	7985 L_	Writes 'L'.
Data Write	↓										↓	↓
Data Write	1	0	0	1	0	0	1	1	1	1	LCD KO_	Writes 'O'.
Cursor/Display Shift	0	0	0	0	0	1	0	0	×	×	LCD KO	Shifts the cursor left.
Cursor/Display Shift	0	0	0	0	0	1	0	0	×	×	LCD KO	Shifts the cursor left.
Data Write	1	0	0	1	0	0	0	0	1	1	CD CO	Writes 'C', the correct character. The display scrolls left.
Cursor/Display Shift	0	0	0	0	0	1	1	1	×	×	LCD CO	Shifts both the display and the cursor right.
Cursor/Display Shift	0	0	0	0	0	1	0	1	×	×	LCD CO_	Shifts the cursor right.
Data Write	1	0	0	1	0	0	1	1	1	0	CD CON_	Writes 'N'.
Data Write	↓										↓	↓
Cursor Home	0	0	0	0	0	0	0	0	1	0	LC7985 L	Sets both the display and the cursor position to 0.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8-bit interface size, 1-line × 8-character display and microcontroller initialization

The initialization sequence for an LC7985 using an 8-bit interface is shown in the following figure.

Note that if power supply conditions are such that the internal reset circuit does not operate to initialize the device, the LC7985 must be initialized using commands from the microcontroller.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8-bit interface size, 2-line × 8-character display and internal reset circuit

The programming example is shown in the following table.

Note that each row uses 40 bytes of DD RAM. When the display is eight characters long, to move the cursor from the first row to the second, the DD RAM address should be reset after the eighth character as shown in the example.

When shifting the display, both rows shift simultaneously but characters do not move from one row to another. Each time the display shifts, the characters in each row only move within the row.

Note that if power supply conditions are such that the internalreset circuit does not operate to initialize the device, the LC7985 must be initialized using commands from the microcontroller.

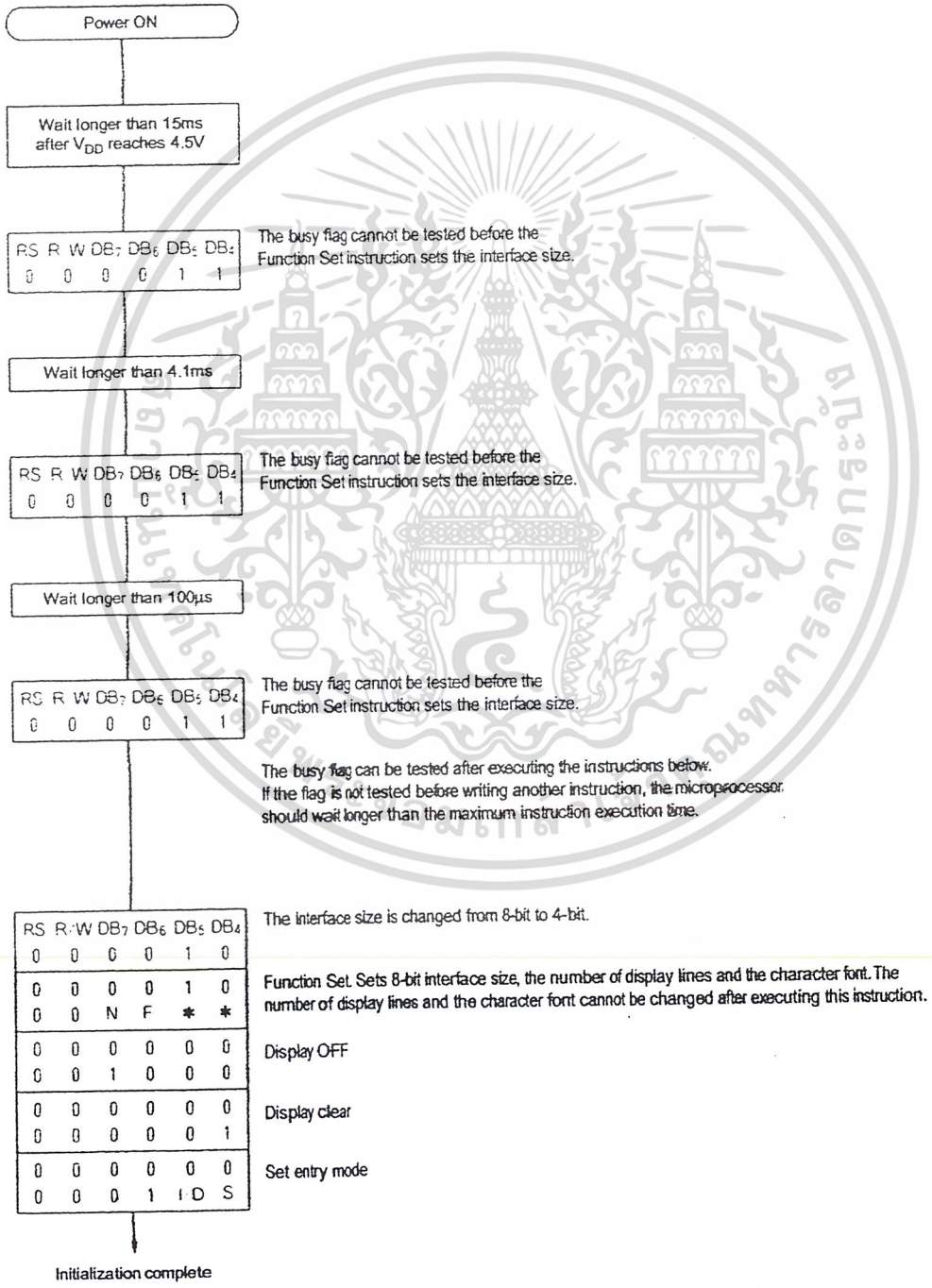
Instruction	Code										Display	Description
	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0		
Power-ON												The internal reset circuit initializes the LC7985. The display is OFF.
Set Function	0	0	0	0	1	1	1	0	×	×		Sets 8-bit interface size, 2-line display size and 5 × 7-pixel character font.
Display ON/OFF	0	0	0	0	0	0	1	1	1	0	_	Turns the display ON and enables the cursor. The display is blank.
Set Entry Mode	0	0	0	0	0	0	0	1	1	0	_	Sets address auto-increment and automatic cursor right shift on writing to DD RAM or CG RAM. The display is not shifted.
Data Write	1	0	0	1	0	0	1	1	0	0	L_	Writes 'L' to DD RAM, since DD RAM was selected when the LC7985 was initialized at power-ON. The cursor position increments and the cursor moves right.
Data Write	↓										↓	↓
Data Write	1	0	0	0	1	1	0	1	0	1	LC7985_	Writes '5'.
Set DD RAM Address	0	0	1	1	0	0	0	0	0	0	LC7985 _	Sets the DD RAM address to the first position in the second row.
Data Write	1	0	0	1	0	0	1	1	0	0	LC7985 L_	Writes 'L'.
Data Write	↓										↓	↓
Data Write	1	0	0	1	0	0	1	1	1	1	LC7985 LCD CO_	Writes 'D'.
Set Entry Mode	0	0	0	0	0	0	0	1	1	1	LC7985 LCD CO_	Sets display shift on writing to DD RAM.
Data Write	1	0	0	1	0	0	1	1	1	0	C7985 CD CON_	Writes 'N.' The display scrolls left. The two lines scroll simultaneously.
Data Write	↓										↓	↓
Cursor Home	0	0	0	0	0	0	0	0	1	0	LC7985 LCD CON	Sets both the display and the cursor position to 0.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4-bit interface size, 1-line × 8-character display and microcontroller initialization

The initialization sequence for an LC7985 using a 4-bit interface is shown in the following figure.

The Function Set instruction is required to set the interface size. With a 4-bit interface size, two write accesses are required for each instruction. Since 8-bit interface size is selected when the LC7985 is initialized at power-ON, the first write access is to an 8-bit interface on the LC7985. DB0 to DB3 are not connected, however, and are not written. The Function Set instruction should therefore be repeated, writing DB4 to DB7 again and then DB0 to DB3, to initialize the device.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- No products described or contained herein are intended for use in surgical implants, life-support systems, aerospace equipment, nuclear power control systems, vehicles, disaster/crime-prevention equipment and the like, the failure of which may directly or indirectly cause injury, death or property loss.
- Anyone purchasing any products described or contained herein for an above-mentioned use shall:
  - ① Accept full responsibility and indemnify and defend SANYO ELECTRIC CO., LTD., its affiliates, subsidiaries and distributors and all their officers and employees, jointly and severally, against any and all claims and litigation and all damages, cost and expenses associated with such use:
  - ② Not impose any responsibility for any fault or negligence which may be cited in any such claim or litigation on SANYO ELECTRIC CO., LTD., its affiliates, subsidiaries and distributors or any of their officers and employees, jointly or severally.
- Information (including circuit diagrams and circuit parameters) herein is for example only; it is not guaranteed for volume production. SANYO believes information herein is accurate and reliable, but no guarantees are made or implied regarding its use or any infringements of intellectual property rights or other rights of third parties.

This catalog provides information as of July, 1997. Specifications and information herein are subject to change without notice.

No. 3255—30/30

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

ADC 0804

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

intersil

ADC0802, ADC0803 ADC0804

8-Bit, Microprocessor-Compatible, A/D Converters

August 1997

Features

- 80C48 and 80C80/85 Bus Compatible - No Interfacing Logic Required
- Conversion Time < 100μs
- Easy Interface to Most Microprocessors
- Will Operate in a "Stand Alone" Mode
- Differential Analog Voltage Inputs
- Works with Bandgap Voltage References
- TTL Compatible Inputs and Outputs
- On-Chip Clock Generator
- 0V to 5V Analog Voltage Input Range (Single +5V Supply)
- No Zero-Adjust Required

Description

The ADC0802 family are CMOS 8-Bit, successive-approximation A/D converters which use a modified potentiometric ladder and are designed to operate with the 8080A control bus via three-state outputs. These converters appear to the processor as memory locations or I/O ports, and hence no interfacing logic is required.

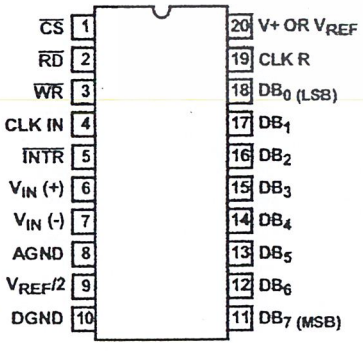
The differential analog voltage input has good common-mode-rejection and permits offsetting the analog zero-input-voltage value. In addition, the voltage reference input can be adjusted to allow encoding any smaller analog voltage span to the full 8 bits of resolution.

Ordering Information

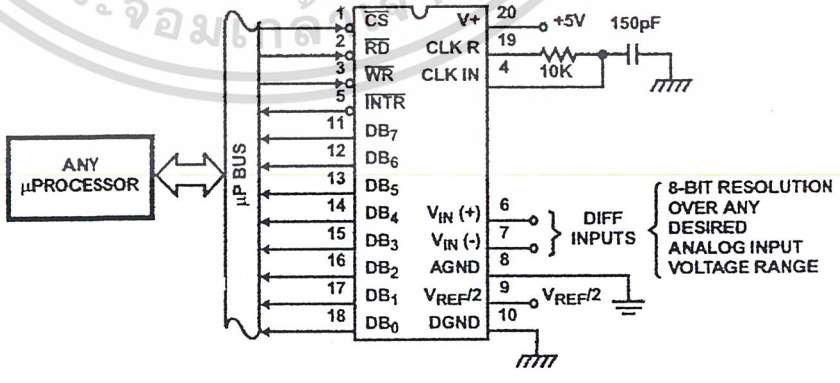
PART NUMBER	ERROR	EXTERNAL CONDITIONS	TEMP. RANGE (°C)	PACKAGE	PKG. NO
ADC0802LCN	$\pm^1/_2$ LSB	$V_{REF}/2 = 2.500V_{DC}$ (No Adjustments)	0 to 70	20 Ld PDIP	E20.3
ADC0802LCD	$\pm^3/_4$ LSB		-40 to 85	20 Ld CERDIP	F20.3
ADC0802LD	±1 LSB		-55 to 125	20 Ld CERDIP	F20.3
ADC0803LCN	$\pm^1/_2$ LSB	$V_{REF}/2$ Adjusted for Correct Full Scale Reading	0 to 70	20 Ld PDIP	E20.3
ADC0803LCD	$\pm^3/_4$ LSB		-40 to 85	20 Ld CERDIP	F20.3
ADC0803LCWM	±1 LSB		-40 to 85	20 Ld SOIC	M20.3
ADC0803LD	±1 LSB		-55 to 125	20 Ld CERDIP	F20.3
ADC0804LCN	±1 LSB	$V_{REF}/2 = 2.500V_{DC}$ (No Adjustments)	0 to 70	20 Ld PDIP	E20.3
ADC0804LCD	±1 LSB		-40 to 85	20 Ld CERDIP	F20.3
ADC0804LCWM	±1 LSB		-40 to 85	20 Ld SOIC	M20.3

Pinout

ADC0802, ADC0803, ADC0804
(PDIP, CERDIP)
TOP VIEW



Typical Application Schematic



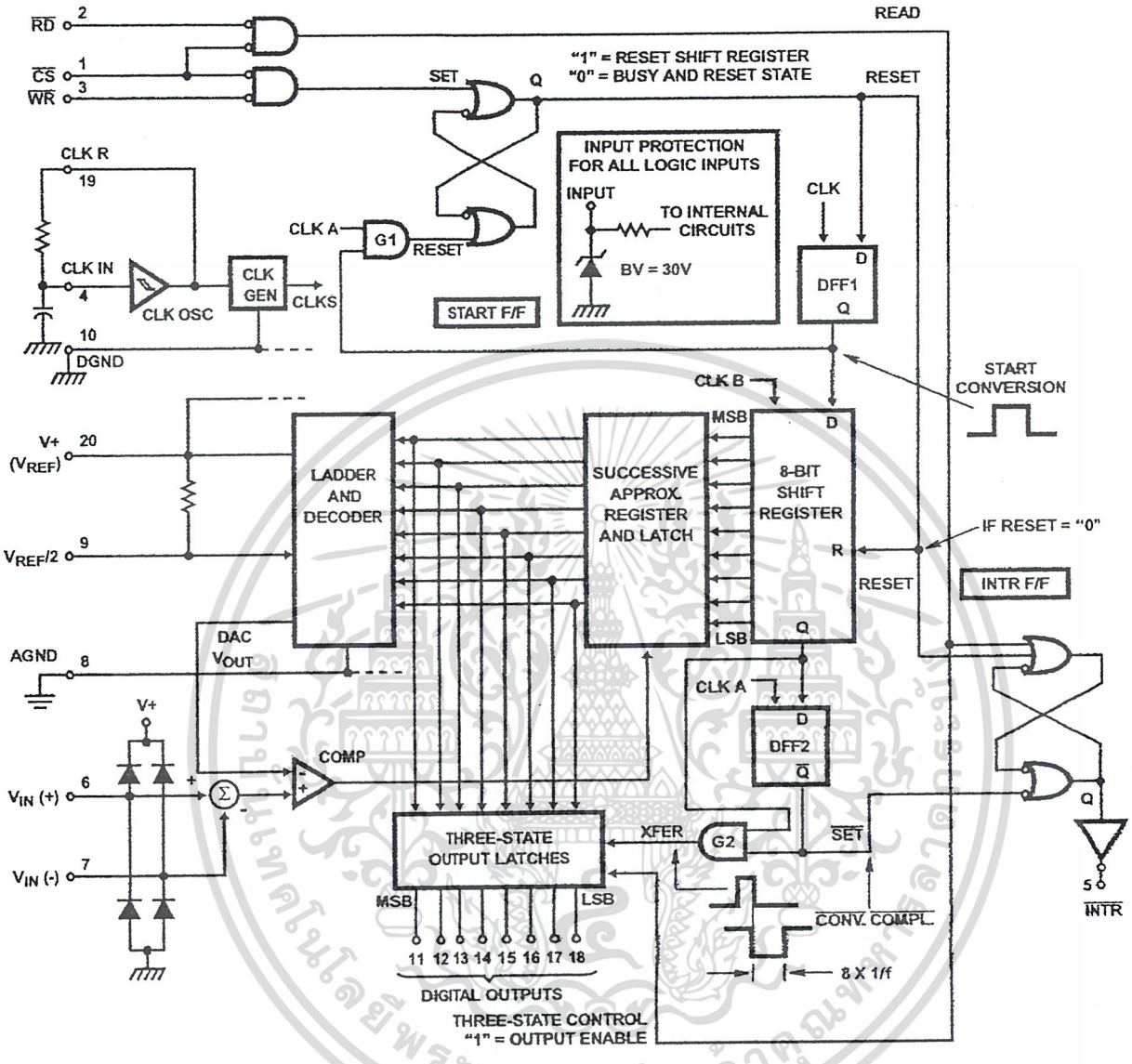
CAUTION: These devices are sensitive to electrostatic discharge; follow proper IC Handling Procedures.

1-888-INTERSIL or 321-724-7143 | Copyright © Intersil Corporation 1999

File Number **3094.1**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Functional Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Absolute Maximum Ratings

Supply Voltage 6.5V
Voltage at Any Input . . . . . -0.3V to (V^+ +0.3V)

Operating Conditions

Temperature Range
ADC0802/03LD. . . . . -55°C to 125°C
ADC0802/03/04LCD . . . . . -40°C to 85°C
ADC0802/03/04LCN . . . . . 0°C to 70°C
ADC0803/04LCWM . . . . . -40°C to 85°C

Thermal Information

Thermal Resistance (Typical, Note 1)	θ_{JA} (°C/W)	θ_{JC} (°C/W)
PDIP Package	125	N/A
CERDIP Package	80	20
SOIC Package	120	N/A

Maximum Junction Temperature
Hermetic Package . . . . . 175°C
Plastic Package . . . . . 150°C
Maximum Storage Temperature Range . . . . . -65°C to 150°C
Maximum Lead Temperature (Soldering, 10s) . . . . . 300°C
(SOIC - Lead Tips Only)

CAUTION: Stresses above those listed in "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress only rating and operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied.

NOTE:

1. θ_{JA} is measured with the component mounted on an evaluation PC board in free air.

Electrical Specifications (Notes 1, 7)

PARAMETER	TEST CONDITIONS	MIN	TYP	MAX	UNITS
CONVERTER SPECIFICATIONS $V+ = 5V$, $T_A = 25°C$ and $f_{CLK} = 640kHz$, Unless Otherwise Specified					
Total Unadjusted Error ADC0802	$V_{REF}/2 = 2.500V$	-	-	$\pm 1/2$	LSB
ADC0803	$V_{REF}/2$ Adjusted for Correct Full Scale Reading	-	-	$\pm 1/2$	LSB
ADC0804	$V_{REF}/2 = 2.500V$	-	-	±1	LSB
$V_{REF}/2$ Input Resistance	Input Resistance at Pin 9	1.0	1.3	-	kΩ
Analog Input Voltage Range	(Note 2)	GND-0.05	-	(V+) + 0.05	V
DC Common-Mode Rejection	Over Analog Input Voltage Range	-	$\pm 1/16$	$\pm 1/8$	LSB
Power Supply Sensitivity	V+ = 5V ±10% Over Allowed Input Voltage Range	-	$\pm 1/16$	$\pm 1/8$	LSB
CONVERTER SPECIFICATIONS $V+ = 5V$, $0°C$ to $70°C$ and $f_{CLK} = 640kHz$, Unless Otherwise Specified					
Total Unadjusted Error ADC0802	$V_{REF}/2 = 2.500V$	-	-	$\pm 1/2$	LSB
ADC0803	$V_{REF}/2$ Adjusted for Correct Full Scale Reading	-	-	$\pm 1/2$	LSB
ADC0804	$V_{REF}/2 = 2.500V$	-	-	±1	LSB
$V_{REF}/2$ Input Resistance	Input Resistance at Pin 9	1.0	1.3	-	kΩ
Analog Input Voltage Range	(Note 2)	GND-0.05	-	(V+) + 0.05	V
DC Common-Mode Rejection	Over Analog Input Voltage Range	-	$\pm 1/8$	$\pm 1/4$	LSB
Power Supply Sensitivity	V+ = 5V ±10% Over Allowed Input Voltage Range	-	$\pm 1/16$	$\pm 1/8$	LSB
CONVERTER SPECIFICATIONS $V+ = 5V$, $-25°C$ to $85°C$ and $f_{CLK} = 640kHz$, Unless Otherwise Specified					
Total Unadjusted Error ADC0802	$V_{REF}/2 = 2.500V$	-	-	$\pm 3/4$	LSB
ADC0803	$V_{REF}/2$ Adjusted for Correct Full Scale Reading	-	-	$\pm 3/4$	LSB
ADC0804	$V_{REF}/2 = 2.500V$	-	-	±1	LSB
$V_{REF}/2$ Input Resistance	Input Resistance at Pin 9	1.0	1.3	-	kΩ
Analog Input Voltage Range	(Note 2)	GND-0.05	-	(V+) + 0.05	V
DC Common-Mode Rejection	Over Analog Input Voltage Range	-	$\pm 1/8$	$\pm 1/4$	LSB
Power Supply Sensitivity	V+ = 5V ±10% Over Allowed Input Voltage Range	-	$\pm 1/16$	$\pm 1/8$	LSB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Electrical Specifications (Notes 1, 7) **(Continued)**

PARAMETER	TEST CONDITIONS	MIN	TYP	MAX	UNITS
CONVERTER SPECIFICATIONS V+ = 5V, -55°C to 125°C and f_{CLK} = 640kHz, Unless Otherwise Specified					
Total Unadjusted Error ADC0802	$V_{REF}/2$ = 2.500V	-	-	±1	LSB
ADC0803	$V_{REF}/2$ Adjusted for Correct Full Scale Reading	-	-	±1	LSB
$V_{REF}/2$ Input Resistance	Input Resistance at Pin 9	1.0	1.3	-	kΩ
Analog Input Voltage Range	(Note 2)	GND-0.05	-	(V+) + 0.05	V
DC Common-Mode Rejection	Over Analog Input Voltage Range	-	$\pm^1/_8$	$\pm^1/_4$	LSB
Power Supply Sensitivity	V+ = 5V±10% Over Allowed Input Voltage Range	-	$\pm^1/_8$	$\pm^1/_4$	LSB
AC TIMING SPECIFICATIONS V+ = 5V, and T_A = 25°C, Unless Otherwise Specified					
Clock Frequency, f_{CLK}	V+ = 6V (Note 3)	100	640	1280	kHz
	V+ = 5V	100	640	800	kHz
Clock Periods per Conversion (Note 4), t_{CONV}		62	-	73	Clocks/Conv
Conversion Rate In Free-Running Mode, CR	$\overline{INTR}$ tied to $\overline{WR}$ with $\overline{CS}$ = 0V, f_{CLK} = 640kHz	-	-	8888	Conv/s
Width of $\overline{WR}$ Input (Start Pulse Width), $t_{W(WR)I}$	$\overline{CS}$ = 0V (Note 5)	100	-	-	ns
Access Time (Delay from Falling Edge of $\overline{RD}$ to Output Data Valid), t_{ACC}	C_L = 100pF (Use Bus Driver IC for Larger C_L)	-	135	200	ns
Three-State Control (Delay from Rising Edge of $\overline{RD}$ to HI-Z State), t_{1H}, t_{0H}	C_L = 10pF, R_L = 10K (See Three-State Test Circuits)	-	125	250	ns
Delay from Falling Edge of $\overline{WR}$ to Reset of $\overline{INTR}$, t_{WI}, t_{RI}		-	300	450	ns
Input Capacitance of Logic Control Inputs, C_{IN}		-	5	-	pF
Three-State Output Capacitance (Data Buffers), C_{OUT}		-	5	-	pF
DC DIGITAL LEVELS AND DC SPECIFICATIONS V+ = 5V, and T_{MIN} to T_{MAX}, Unless Otherwise Specified					
CONTROL INPUTS (Note 6)					
Logic "1" Input Voltage (Except Pin 4 CLK IN), V_{INH}	V+ = 5.25V	2.0	-	V+	V
Logic "0" Input Voltage (Except Pin 4 CLK IN), V_{INL}	V+ = 4.75V	-	-	0.8	V
CLK IN (Pin 4) Positive Going Threshold Voltage, V^+_{CLK}		2.7	3.1	3.5	V
CLK IN (Pin 4) Negative Going Threshold Voltage, V^-_{CLK}		1.5	1.8	2.1	V
CLK IN (Pin 4) Hysteresis, V_H		0.6	1.3	2.0	V
Logic "1" Input Current (All Inputs), I_{INHI}	V_{IN} = 5V	-	0.005	1	µA
Logic "0" Input Current (All Inputs), I_{INLO}	V_{IN} = 0V	-1	-0.005	-	µA
Supply Current (Includes Ladder Current), I+	f_{CLK} = 640kHz, T_A = 25°C and $\overline{CS}$ = HI	-	1.3	2.5	mA
DATA OUTPUTS AND $\overline{INTR}$					
Logic "0" Output Voltage, V_{OL}	I_O = 1.6mA, V+ = 4.75V	-	-	0.4	V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Electrical Specifications (Notes 1, 7) (Continued)

PARAMETER	TEST CONDITIONS	MIN	TYP	MAX	UNITS
Logic "1" Output Voltage, V_{OH}	I_O = -360μA, V+ = 4.75V	2.4	-	-	V
Three-State Disabled Output Leakage (All Data Buffers), I_{LO}	V_{OUT} = 0V	-3	-	-	μA
	V_{OUT} = 5V	-	-	3	μA
Output Short Circuit Current, I_{SOURCE}	V_{OUT} Short to Gnd T_A = 25°C	4.5	6	-	mA
Output Short Circuit Current, I_{SINK}	V_{OUT} Short to V+ T_A = 25°C	9.0	16	-	mA

NOTES:

1. All voltages are measured with respect to GND, unless otherwise specified. The separate AGND point should always be wired to the DGND, being careful to avoid ground loops.
2. For $V_{IN(-)} \geq V_{IN(+)}$ the digital output code will be 0000 0000. Two on-chip diodes are tied to each analog input (see Block Diagram) which will forward conduct for analog input voltages one diode drop below ground or one diode drop greater than the V+ supply. Be careful, during testing at low V+ levels (4.5V), as high level analog inputs (5V) can cause this input diode to conduct - especially at elevated temperatures, and cause errors for analog inputs near full scale. As long as the analog V_{IN} does not exceed the supply voltage by more than 50mV, the output code will be correct. To achieve an absolute 0V to 5V input voltage range will therefore require a minimum supply voltage of 4.950V over temperature variations, initial tolerance and loading.
3. With V+ = 6V, the digital logic interfaces are no longer TTL compatible.
4. With an asynchronous start pulse, up to 8 clock periods may be required before the internal clock phases are proper to start the conversion process.
5. The $\overline{CS}$ input is assumed to bracket the $\overline{WR}$ strobe input so that timing is dependent on the $\overline{WR}$ pulse width. An arbitrarily wide pulse width will hold the converter in a reset mode and the start of conversion is initiated by the low to high transition of the $\overline{WR}$ pulse (see Timing Diagrams).
6. CLK IN (pin 4) is the input of a Schmitt trigger circuit and is therefore specified separately.
7. None of these A/Ds requires a zero-adjust. However, if an all zero code is desired for an analog input other than 0V, or if a narrow full scale span exists (for example: 0.5V to 4V full scale) the $V_{IN(-)}$ input can be adjusted to achieve this. See the Zero Error description in this data sheet.

Timing Waveforms

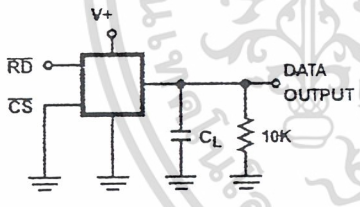


FIGURE 1A. t_{1H}

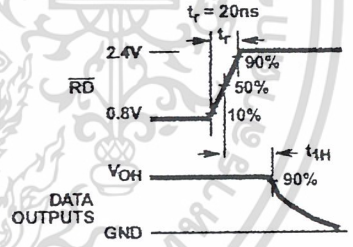


FIGURE 1B. t_{1H}, C_L = 10pF

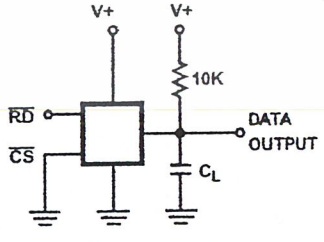


FIGURE 1C. t_{0H}

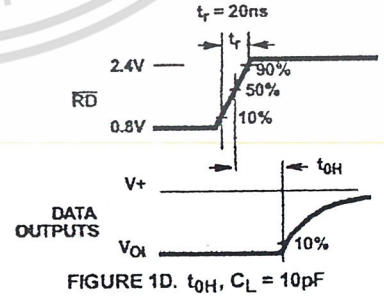


FIGURE 1D. t_{0H}, C_L = 10pF

FIGURE 1. THREE-STATE CIRCUITS AND WAVEFORMS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Typical Performance Curves

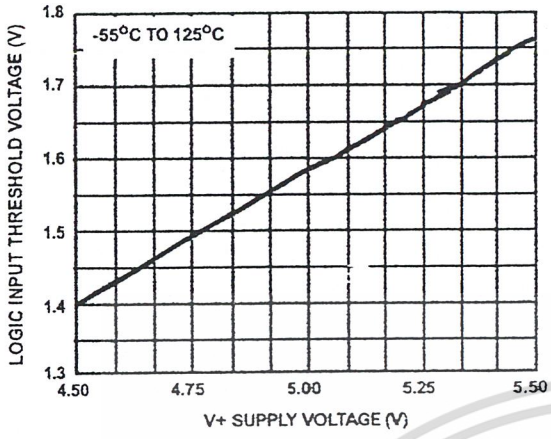


FIGURE 2. LOGIC INPUT THRESHOLD VOLTAGE vs SUPPLY VOLTAGE

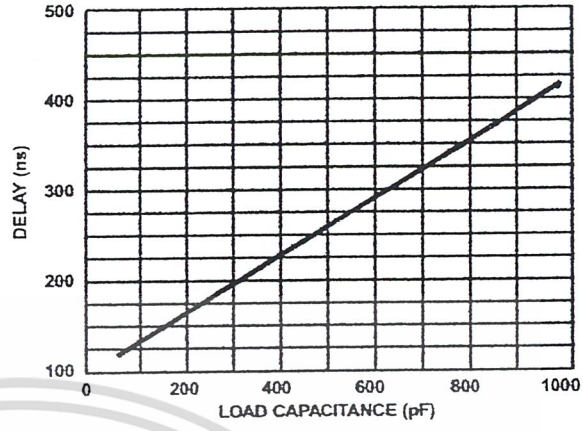


FIGURE 3. DELAY FROM FALLING EDGE OF $\overline{RD}$ TO OUTPUT DATA VALID vs LOAD CAPACITANCE

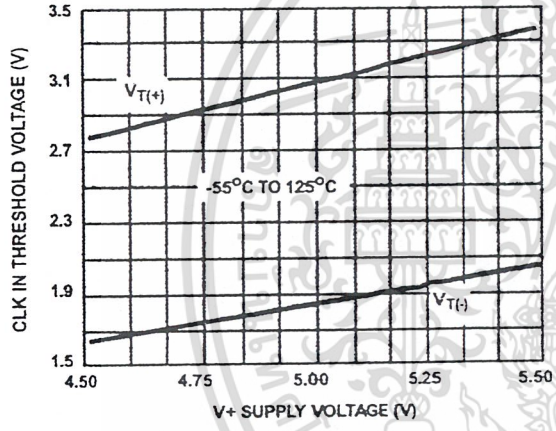


FIGURE 4. CLK IN SCHMITT TRIP LEVELS vs SUPPLY VOLTAGE

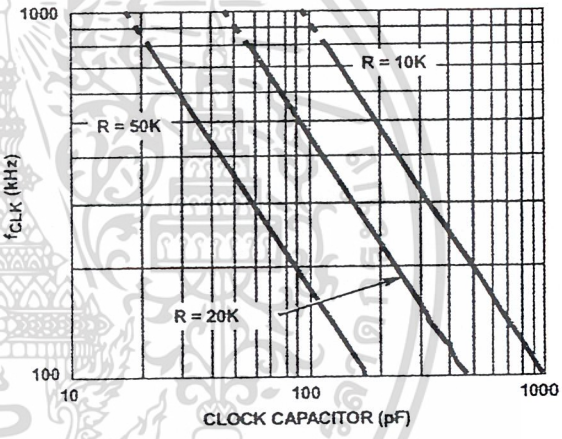


FIGURE 5. f_{CLK} vs CLOCK CAPACITOR

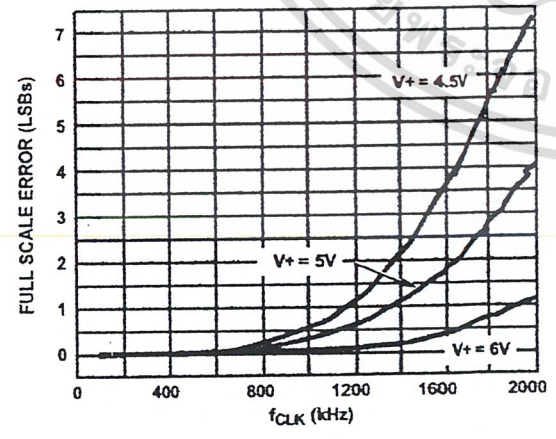


FIGURE 6. FULL SCALE ERROR vs f_{CLK}

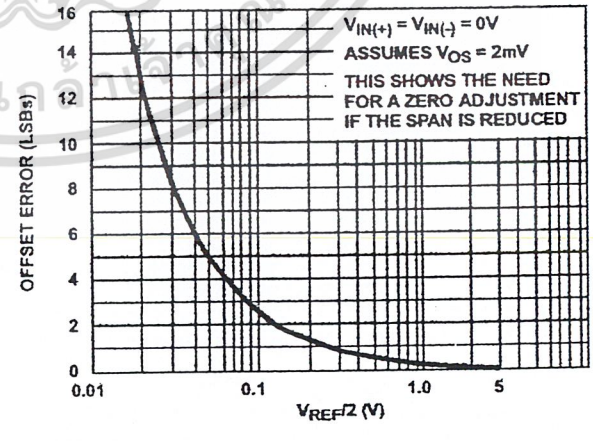


FIGURE 7. EFFECT OF UNADJUSTED OFFSET ERROR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Typical Performance Curves (Continued)

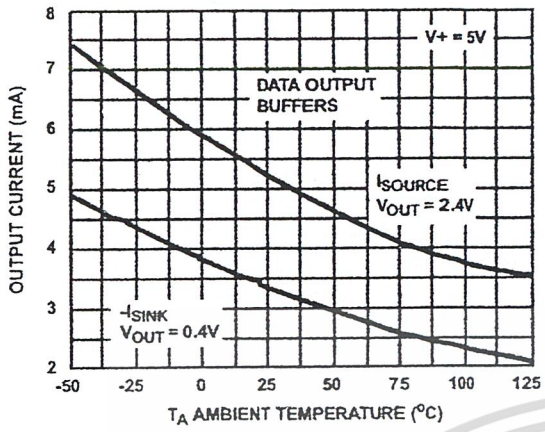


FIGURE 8. OUTPUT CURRENT vs TEMPERATURE

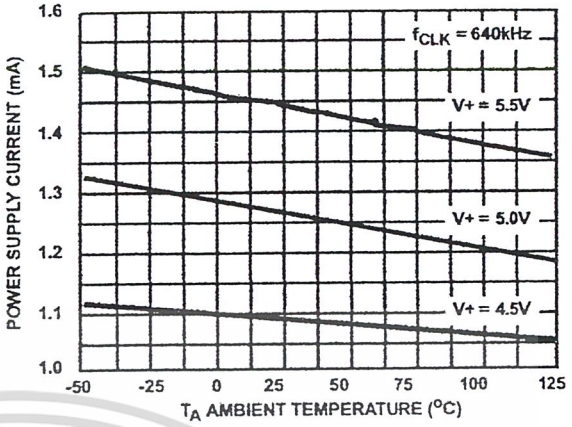


FIGURE 9. POWER SUPPLY CURRENT vs TEMPERATURE

Timing Diagrams

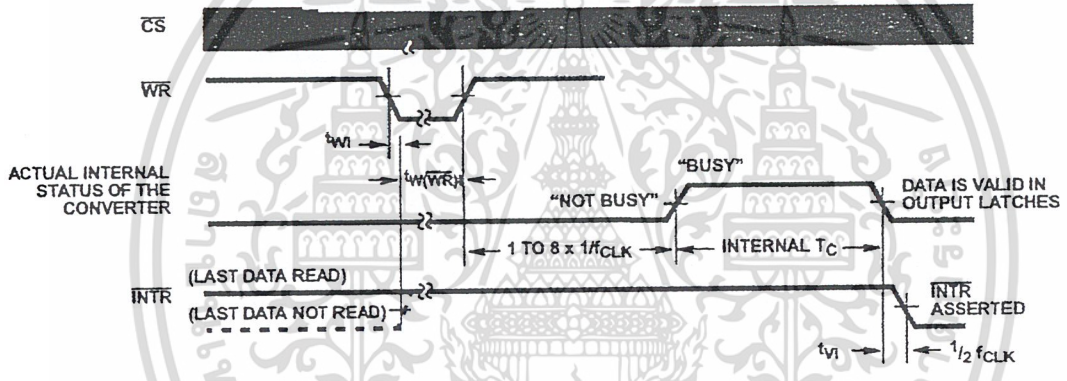


FIGURE 10A. START CONVERSION

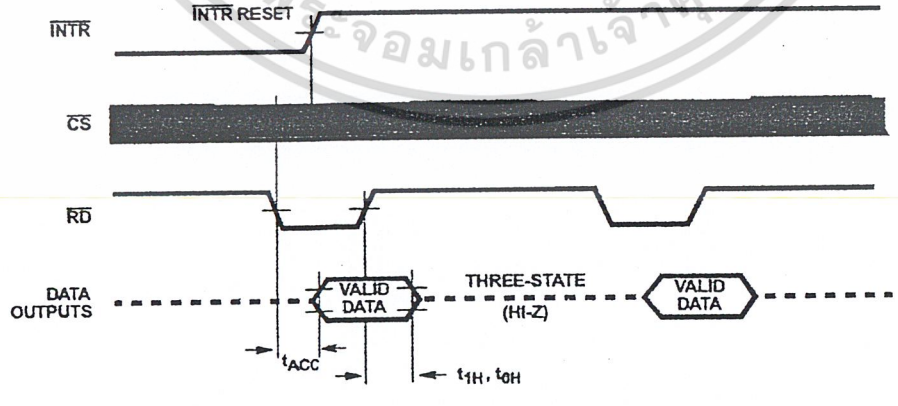


FIGURE 10B. OUTPUT ENABLE AND RESET INTR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

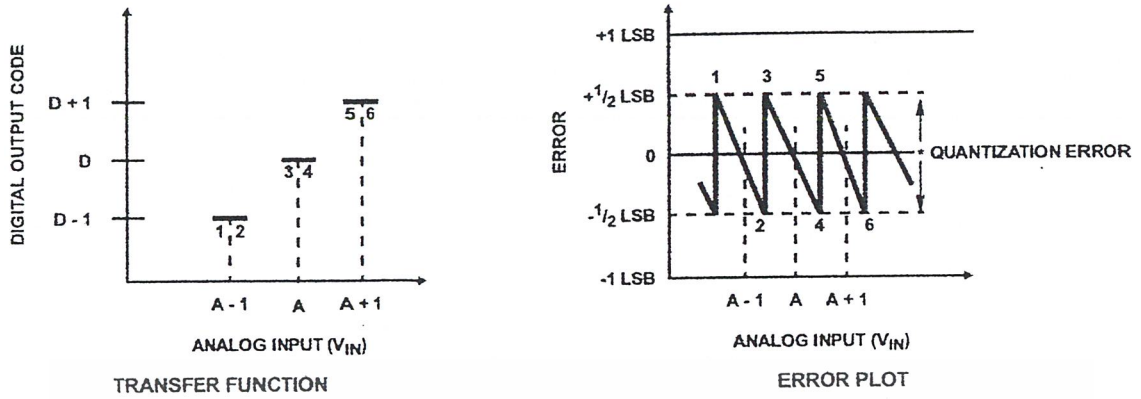


FIGURE 11A. ACCURACY = ±0 LSB; PERFECT A/D

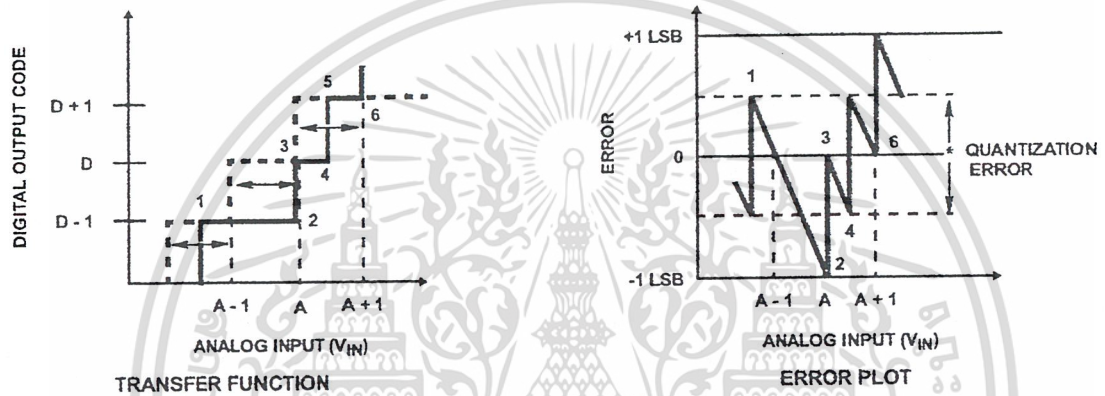


FIGURE 11B. ACCURACY = $\pm^1/_2$ LSB

FIGURE 11. CLARIFYING THE ERROR SPECS OF AN A/D CONVERTER

Understanding A/D Error Specs

A perfect A/D transfer characteristic (staircase wave-form) is shown in Figure 11A. The horizontal scale is analog input voltage and the particular points labeled are in steps of 1 LSB (19.53mV with 2.5V tied to the $V_{REF}/2$ pin). The digital output codes which correspond to these inputs are shown as D-1, D, and D+1. For the perfect A/D, not only will center-value (A - 1, A, A + 1, . . .) analog inputs produce the correct output digital codes, but also each riser (the transitions between adjacent output codes) will be located $\pm^1/_2$ LSB away from each center-value. As shown, the risers are ideal and have no width. Correct digital output codes will be provided for a range of analog input voltages which extend $\pm^1/_2$ LSB from the ideal center-values. Each tread (the range of analog input voltage which provides the same digital output code) is therefore 1 LSB wide.

The error curve of Figure 11B shows the worst case transfer function for the ADC0802. Here the specification guarantees that if we apply an analog input equal to the LSB analog voltage center-value, the A/D will produce the correct digital code.

Next to each transfer function is shown the corresponding error plot. Notice that the error includes the quantization uncertainty of the A/D. For example, the error at point 1 of Figure 11A is $+^1/_2$ LSB because the digital code appeared $^1/_2$ LSB in advance of the center-value of the tread. The error plots always have a constant negative slope and the abrupt upside steps are always 1 LSB in magnitude, unless the device has missing codes.

Detailed Description

The functional diagram of the ADC0802 series of A/D converters operates on the successive approximation principle (see Application Notes AN016 and AN020 for a more detailed description of this principle). Analog switches are closed sequentially by successive-approximation logic until the analog differential input voltage $[V_{IN(+)} - V_{IN(-)}]$ matches a voltage derived from a tapped resistor string across the reference voltage. The most significant bit is tested first and after 8 comparisons (64 clock cycles), an 8-bit binary code (1111 1111 = full scale) is transferred to an output latch.

The normal operation proceeds as follows. On the high-to-low transition of the WR input, the internal SAR latches and the shift-register stages are reset, and the INTR output will be set high. As long as the CS input and WR input remain low, the A/D will remain in a reset state. Conversion will start from 1 to 8 clock periods after at least one of these inputs makes a low-to-high transition. After the requisite number of clock pulses to complete the conversion, the INTR pin will make a high-to-low transition. This can be used to interrupt a processor, or otherwise signal the availability of a new conversion. A RD operation (with CS low) will clear the INTR line high again.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The device may be operated in the free-running mode by connecting INTR to the WR input with CS = 0. To ensure start-up under all possible conditions, an external WR pulse is required during the first power-up cycle. A conversion-in-process can be interrupted by issuing a second start command.

Digital Operation

The converter is started by having CS and WR simultaneously low. This sets the start flip-flop (F/F) and the resulting "1" level resets the 8-bit shift register, resets the Interrupt (INTR) F/F and inputs a "1" to the D flip-flop, DFF1, which is at the input end of the 8-bit shift register. Internal clock signals then transfer this "1" to the Q output of DFF1. The AND gate, G1, combines this "1" output with a clock signal to provide a reset signal to the start F/F. If the set signal is no longer present (either WR or CS is a "1"), the start F/F is reset and the 8-bit shift register then can have the "1" clocked in, which starts the conversion process. If the set signal were to still be present, this reset pulse would have no effect (both outputs of the start F/F would be at a "1" level) and the 8-bit shift register would continue to be held in the reset mode. This allows for asynchronous or wide CS and WR signals.

After the "1" is clocked through the 8-bit shift register (which completes the SAR operation) it appears as the input to DFF2. As soon as this "1" is output from the shift register, the AND gate, G2, causes the new digital word to transfer to the Three-State output latches. When DFF2 is subsequently clocked, the $\overline{Q}$ output makes a high-to-low transition which causes the INTR F/F to set. An inverting buffer then supplies the $\overline{INTR}$ output signal.

When data is to be read, the combination of both $\overline{CS}$ and $\overline{RD}$ being low will cause the INTR F/F to be reset and the three-state output latches will be enabled to provide the 8-bit digital outputs.

Digital Control Inputs

The digital control inputs (CS, RD, and WR) meet standard TTL logic voltage levels. These signals are essentially equivalent to the standard A/D Start and Output Enable control signals, and are active low to allow an easy interface to microprocessor control busses. For non-microprocessor based applications, the CS input (pin 1) can be grounded and the standard A/D Start function obtained by an active low pulse at the WR input (pin 3). The Output Enable function is achieved by an active low pulse at the RD input (pin 2).

Analog Operation

The analog comparisons are performed by a capacitive charge summing circuit. Three capacitors (with precise ratioed values) share a common node with the input to an auto-zeroed comparator. The input capacitor is switched between $V_{IN(+)}$ and $V_{IN(-)}$, while two ratioed reference capacitors are switched between taps on the reference voltage divider string. The net charge corresponds to the weighted difference between the input and the current total value set by the successive approximation register. A correction is made to offset the comparison by $^1/_2$ LSB (see Figure 11A).

Analog Differential Voltage Inputs and Common-Mode Rejection

This A/D gains considerable applications flexibility from the analog differential voltage input. The $V_{IN(-)}$ input (pin 7) can be used to automatically subtract a fixed voltage value from the input reading (tare correction). This is also useful in 4mA - 20mA current loop conversion. In addition, common-mode noise can be reduced by use of the differential input.

The time interval between sampling $V_{IN(+)}$ and $V_{IN(-)}$ is $4^1/_2$ clock periods. The maximum error voltage due to this slight time difference between the input voltage samples is given by:

$$\Delta V_E(\quad) = (\quad)(\quad)\left[\frac{4.5}{f_{CLK}}\right]$$

where:

ΔV_E is the error voltage due to sampling delay,

V_{PEAK} is the peak value of the common-mode voltage,

f_{CM} is the common-mode frequency.

For example, with a 60Hz common-mode frequency, f_{CM}, and a 640kHz A/D clock, f_{CLK}, keeping this error to $^1/_4$ LSB (~5mV) would allow a common-mode voltage, V_{PEAK}, given by:

$$V_{PEAK} = \frac{\left[\Delta V_E(\quad)(\quad)\right]}{(\quad)(\quad)}$$

or

$$V_{PEAK} = \frac{(\quad)(\quad)}{(\quad)(\quad)(\quad)} \cong 1.9V.$$

The allowed range of analog input voltage usually places more severe restrictions on input common-mode voltage levels than this.

An analog input voltage with a reduced span and a relatively large zero offset can be easily handled by making use of the differential input (see Reference Voltage Span Adjust).

Analog Input Current

The internal switching action causes displacement currents to flow at the analog inputs. The voltage on the on-chip capacitance to ground is switched through the analog differential input voltage, resulting in proportional currents entering the $V_{IN(+)}$ input and leaving the $V_{IN(-)}$ input. These current transients occur at the leading edge of the internal clocks. They rapidly decay and do not inherently cause errors as the on-chip comparator is strobed at the end of the clock period.

Input Bypass Capacitors

Bypass capacitors at the inputs will average these charges and cause a DC current to flow through the output resistances of the analog signal sources. This charge pumping action is worse for continuous conversions with the $V_{IN(+)}$ input voltage at full scale. For a 640kHz clock frequency with the $V_{IN(+)}$ input at 5V, this DC current is at a maximum of approximately 5μA. Therefore, **bypass capacitors should not be used at the analog inputs or the $V_{REF}/2$** pin for high resistance sources (>1kΩ). If input bypass capacitors are necessary for noise filtering and high source resistance is desirable to minimize capacitor size, the effects of the voltage drop across this input resistance, due to the average value of the input current, can be compensated by a full scale adjustment while the given source resistor and input bypass capacitor are both in place. This is possible because the average value of the input current is a precise linear function of the differential input voltage at a constant conversion rate.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
112

Input Source Resistance

Large values of source resistance where an input bypass capacitor is not used will not cause errors since the input currents settle out prior to the comparison time. If a low-pass filter is required in the system, use a low-value series resistor (≤1kΩ) for a passive RC section or add an op amp RC active low-pass filter. For low-source-resistance applications (≤1kΩ), a 0.1μF bypass capacitor at the inputs will minimize EMI due to the series lead inductance of a long wire. A 100Ω series resistor can be used to isolate this capacitor (both the R and C are placed outside the feedback loop) from the output of an op amp, if used.

Stray Pickup

The leads to the analog inputs (pins 6 and 7) should be kept as short as possible to minimize stray signal pickup (EMI). Both EMI and undesired digital-clock coupling to these inputs can cause system errors. The source resistance for these inputs should, in general, be kept below 5kΩ. Larger values of source resistance can cause undesired signal pickup. Input bypass capacitors, placed from the analog inputs to ground, will eliminate this pickup but can create analog scale errors as these capacitors will average the transient input switching currents of the A/D (see Analog Input Current). This scale error depends on both a large source resistance and the use of an input bypass capacitor. This error can be compensated by a full scale adjustment of the A/D (see Full Scale Adjustment) with the source resistance and input bypass capacitor in place, and the desired conversion rate.

Reference Voltage Span Adjust

For maximum application flexibility, these A/Ds have been designed to accommodate a 5V, 2.5V or an adjusted voltage reference. This has been achieved in the design of the IC as shown in Figure 12.

Notice that the reference voltage for the IC is either $^1/_2$ of the voltage which is applied to the V+ supply pin, or is equal to the voltage which is externally forced at the $V_{REF}/2$ pin. This allows for a pseudo-ratiometric voltage reference using, for the V+ supply, a 5V reference voltage. Alternatively, a voltage less than 2.5V can be applied to the $V_{REF}/2$ input. The internal gain to the $V_{REF}/2$ input is 2 to allow this factor of 2 reduction in the reference voltage.

Such an adjusted reference voltage can accommodate a reduced span or dynamic voltage range of the analog input voltage. If the analog input voltage were to range from 0.5V to 3.5V, instead of 0V to 5V, the span would be 3V. With 0.5V applied to the $V_{IN(-)}$ pin to absorb the offset, the reference voltage can be made equal to $^1/_2$ of the 3V span or 1.5V. The A/D now will encode the $V_{IN(+)}$ signal from 0.5V to 3.5V with the 0.5V input corresponding to zero and the 3.5V input corresponding to full scale. The full 8 bits of resolution are therefore applied over this reduced analog input voltage range. The requisite connections are shown in Figure 13. For expanded scale inputs, the circuits of Figures 14 and 15 can be used.

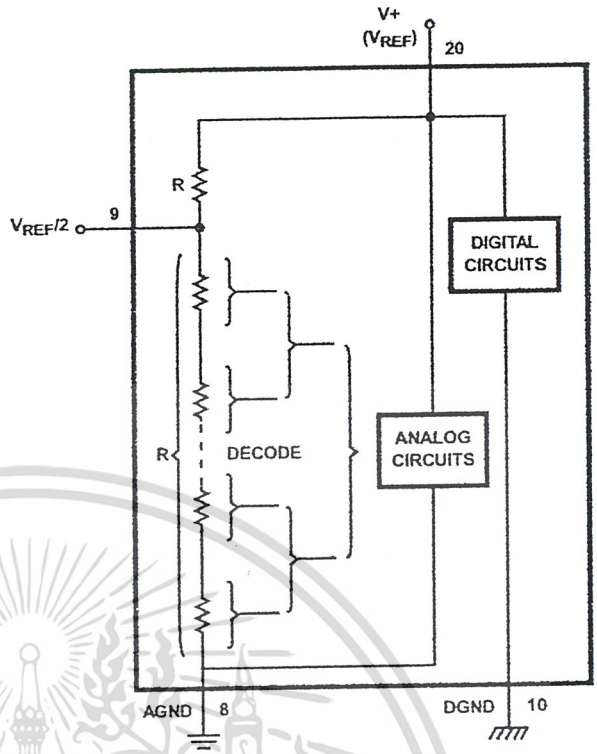


FIGURE 12. THE $V_{REFERENCE}$ DESIGN ON THE IC

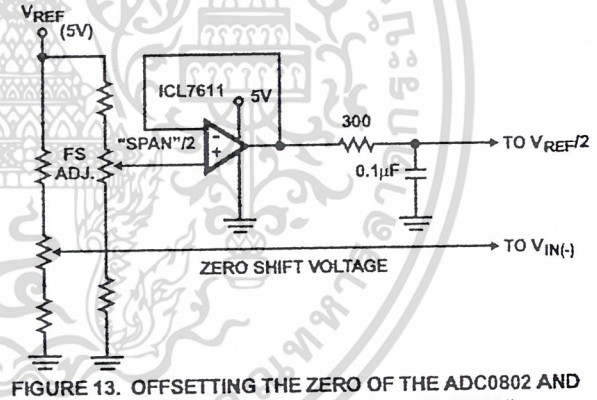


FIGURE 13. OFFSETTING THE ZERO OF THE ADC0802 AND PERFORMING AN INPUT RANGE (SPAN) ADJUSTMENT

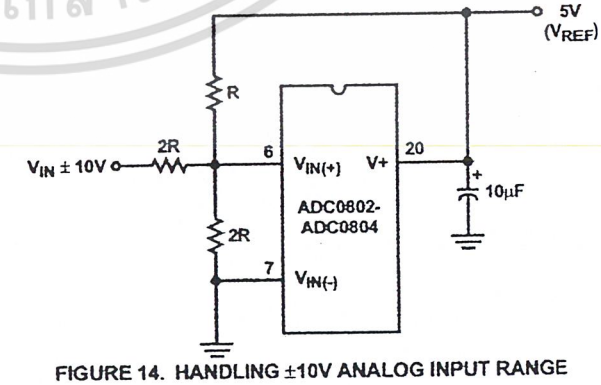


FIGURE 14. HANDLING ±10V ANALOG INPUT RANGE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

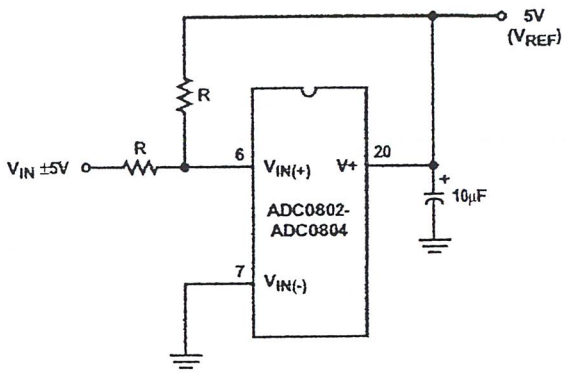


FIGURE 15. HANDLING ±5V ANALOG INPUT RANGE

Reference Accuracy Requirements

The converter can be operated in a pseudo-ratiometric mode or an absolute mode. In ratiometric converter applications, the magnitude of the reference voltage is a factor in both the output of the source transducer and the output of the A/D converter and therefore cancels out in the final digital output code. In absolute conversion applications, both the initial value and the temperature stability of the reference voltage are important accuracy factors in the operation of the A/D converter. For $V_{REF}/2$ voltages of 2.5V nominal value, initial errors of ±10mV will cause conversion errors of ±1 LSB due to the gain of 2 of the $V_{REF}/2$ input. In reduced span applications, the initial value and the stability of the $V_{REF}/2$ input voltage become even more important. For example, if the span is reduced to 2.5V, the analog input LSB voltage value is correspondingly reduced from 20mV (5V span) to 10mV and 1 LSB at the $V_{REF}/2$ input becomes 5mV. As can be seen, this reduces the allowed initial tolerance of the reference voltage and requires correspondingly less absolute change with temperature variations. Note that spans smaller than 2.5V place even tighter requirements on the initial accuracy and stability of the reference source.

In general, the reference voltage will require an initial adjustment. Errors due to an improper value of reference voltage appear as full scale errors in the A/D transfer function. IC voltage regulators may be used for references if the ambient temperature changes are not excessive.

Zero Error

The zero of the A/D does not require adjustment. If the minimum analog input voltage value, $V_{IN(MIN)}$, is not ground, a zero offset can be done. The converter can be made to output 0000 0000 digital code for this minimum input voltage by biasing the A/D $V_{IN(-)}$ input at this $V_{IN(MIN)}$ value (see Applications section). This utilizes the differential mode operation of the A/D.

The zero error of the A/D converter relates to the location of the first riser of the transfer function and can be measured by grounding the $V_{IN(-)}$ input and applying a small magnitude positive voltage to the $V_{IN(+)}$ input. Zero error is the difference between the actual DC input voltage which is necessary to just cause an output digital code transition from 0000 0000 to 0000 0001 and the ideal $^1/_2$ LSB value ($^1/_2$ LSB = 9.8mV for $V_{REF}/2$ = 2.500V).

Full Scale Adjust

The full scale adjustment can be made by applying a differential input voltage which is $1^1/_2$ LSB down from the desired analog full scale voltage range and then adjusting the magnitude of the $V_{REF}/2$ input (pin 9) for a digital output code which is just changing from 1111 1110 to 1111 1111. When offsetting the zero and using a span-adjusted $V_{REF}/2$ voltage, the full scale adjustment is made by inputting V_{MIN} to the $V_{IN(-)}$ input of the A/D and applying a voltage to the $V_{IN(+)}$ input which is given by:

$$V_{IN(+)fSADJ} = V_{MAX} - 1.5\left[\frac{(\text{.....})}{256}\right],$$

where:

V_{MAX} = the high end of the analog input range,

and

V_{MIN} = the low end (the offset zero) of the analog range. (Both are ground referenced.)

Clocking Option

The clock for the A/D can be derived from an external source such as the CPU clock or an external RC network can be added to provide self-clocking. The CLK IN (pin 4) makes use of a Schmitt trigger as shown in Figure 16.

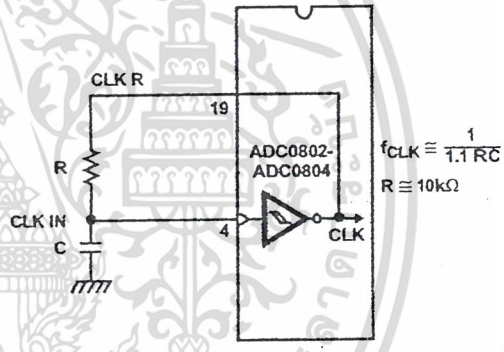


FIGURE 16. SELF-CLOCKING THE A/D

Heavy capacitive or DC loading of the CLK R pin should be avoided as this will disturb normal converter operation. Loads less than 50pF, such as driving up to 7 A/D converter clock inputs from a single CLK R pin of 1 converter, are allowed. For larger clock line loading, a CMOS or low power TTL buffer or PNP input logic should be used to minimize the loading on the CLK R pin (do not use a standard TTL buffer).

Restart During a Conversion

If the A/D is restarted ($\overline{CS}$ and $\overline{WR}$ go low and return high) during a conversion, the converter is reset and a new conversion is started. The output data latch is not updated if the conversion in progress is not completed. The data from the previous conversion remain in this latch.

Continuous Conversions

In this application, the $\overline{CS}$ input is grounded and the $\overline{WR}$ input is tied to the $\overline{INTR}$ output. This $\overline{WR}$ and $\overline{INTR}$ node should be momentarily forced to logic low following a power-up cycle to insure circuit operation. See Figure 17 for details.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

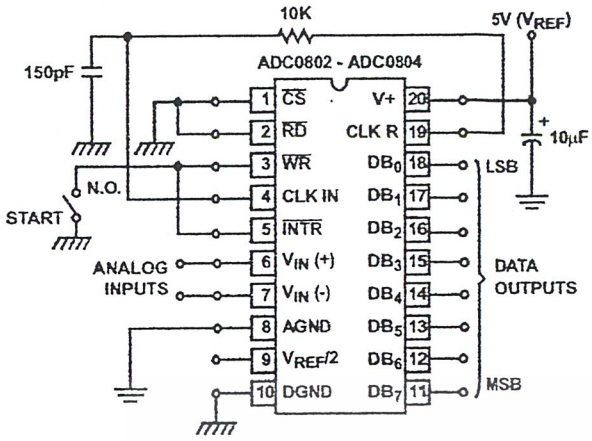


FIGURE 17. FREE-RUNNING CONNECTION

Driving the Data Bus

This CMOS A/D, like MOS microprocessors and memories, will require a bus driver when the total capacitance of the data bus gets large. Other circuitry, which is tied to the data bus, will add to the total capacitive loading, even in three-state (high-impedance mode). Back plane busing also greatly adds to the stray capacitance of the data bus.

There are some alternatives available to the designer to handle this problem. Basically, the capacitive loading of the data bus slows down the response time, even though DC specifications are still met. For systems operating with a relatively slow CPU clock frequency, more time is available in which to establish proper logic levels on the bus and therefore higher capacitive loads can be driven (see Typical Performance Curves).

At higher CPU clock frequencies time can be extended for I/O reads (and/or writes) by inserting wait states (8080) or using clock-extending circuits (6800).

Finally, if time is short and capacitive loading is high, external bus drivers must be used. These can be three-state buffers (low power Schottky is recommended, such as the 74LS240 series) or special higher-drive-current products which are designed as bus drivers. High-current bipolar bus drivers with PNP inputs are recommended.

Power Supplies

Noise spikes on the V+ supply line can cause conversion errors as the comparator will respond to this noise. A low-inductance tantalum filter capacitor should be used close to the converter V+ pin, and values of 1μF or greater are recommended. If an unregulated voltage is available in the system, a separate 5V voltage regulator for the converter (and other analog circuitry) will greatly reduce digital noise on the V+ supply. An ICL7663 can be used to regulate such a supply from an input as low as 5.2V.

Wiring and Hook-Up Precautions

Standard digital wire-wrap sockets are not satisfactory for breadboarding with this A/D converter. Sockets on PC boards can be used. All logic signal wires and leads should be grouped and kept as far away as possible from the analog signal leads. Exposed leads to the analog inputs can cause undesired digital noise and hum pickup; therefore, shielded leads may be necessary in many applications.

A single-point analog ground should be used which is separate from the logic ground points. The power supply bypass capacitor and the self-clocking capacitor (if used) should both be returned to digital ground. Any $V_{REF}/2$ bypass capacitors, analog input filter capacitors, or input signal shielding should be returned to the analog ground point. A test for proper grounding is to measure the zero error of the A/D converter. Zero errors in excess of $^1/_4$ LSB can usually be traced to improper board layout and wiring (see Zero Error for measurement). Further information can be found in Application Note AN018.

Testing the A/D Converter

There are many degrees of complexity associated with testing an A/D converter. One of the simplest tests is to apply a known analog input voltage to the converter and use LEDs to display the resulting digital output code as shown in Figure 18.

For ease of testing, the $V_{REF}/2$ (pin 9) should be supplied with 2.560V and a V+ supply voltage of 5.12V should be used. This provides an LSB value of 20mV.

If a full scale adjustment is to be made, an analog input voltage of 5.090V (5.120 - $1^1/_2$ LSB) should be applied to the $V_{IN(+)}$ pin with the $V_{IN(-)}$ pin grounded. The value of the $V_{REF}/2$ input voltage should be adjusted until the digital output code is just changing from 1111 1110 to 1111 1111. This value of $V_{REF}/2$ should then be used for all the tests.

The digital-output LED display can be decoded by dividing the 8 bits into 2 hex characters, one with the 4 most-significant bits (MS) and one with the 4 least-significant bits (LS). The output is then interpreted as a sum of fractions times the full scale voltage:

$V_{OUT} = \left(\quad\right)(\quad)V.$

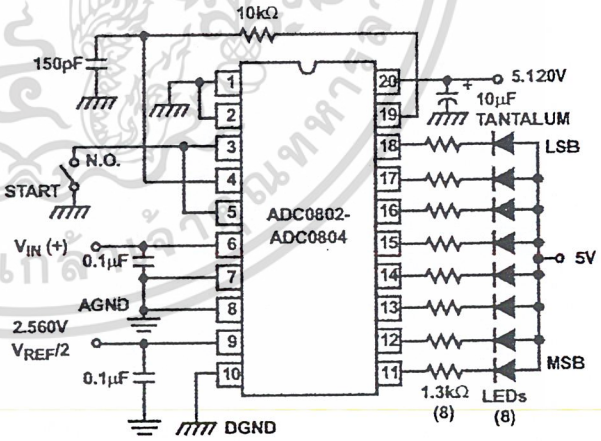


FIGURE 18. BASIC TESTER FOR THE A/D

For example, for an output LED display of 1011 0110, the MS character is hex B (decimal 11) and the LS character is hex (and decimal) 6, so:

$V_{OUT} = \left(\quad\right)(\quad) = 3.64V.$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figures 19 and 20 show more sophisticated test circuits.

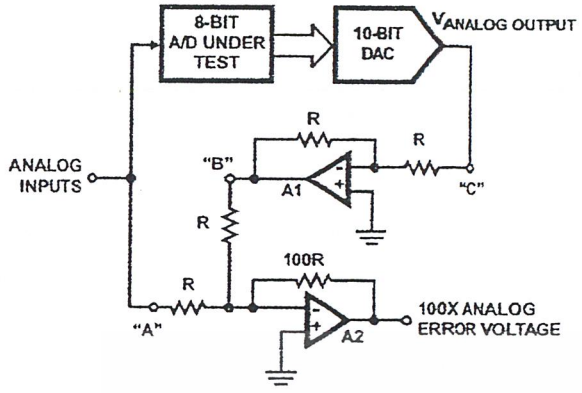


FIGURE 19. A/D TESTER WITH ANALOG ERROR OUTPUT. THIS CIRCUIT CAN BE USED TO GENERATE "ERROR PLOTS" OF FIGURE 11.

FIGURE 20. BASIC "DIGITAL" A/D TESTER

Typical Applications

Interfacing 8080/85 or Z-80 Microprocessors

This converter has been designed to directly interface with 8080/85 or Z-80 Microprocessors. The three-state output capability of the A/D eliminates the need for a peripheral interface device, although address decoding is still required to generate the appropriate $\overline{CS}$ for the converter. The A/D can be mapped into memory space (using standard memory-address decoding for $\overline{CS}$ and the $\overline{MEMR}$ and $\overline{MEMW}$ strobes) or it can be controlled as an I/O device by using the $\overline{I/OR}$ and $\overline{I/OW}$ strobes and decoding the address bits A0 → A7 (or address bits A8 → A15, since they will contain the same 8-bit address information) to obtain the $\overline{CS}$ input. Using the I/O space provides 256 additional addresses and may allow a simpler 8-bit address decoder, but the data can only be input to the accumulator. To make use of the additional memory reference instructions, the A/D should be mapped into memory space. See AN020 for more discussion of memory-mapped vs I/O-mapped interfaces. An example of an A/D in I/O space is shown in Figure 21.

The standard control-bus signals of the 8080 ($\overline{CS}$, $\overline{RD}$ and $\overline{WR}$) can be directly wired to the digital control inputs of the A/D, since the bus timing requirements, to allow both starting the converter, and outputting the data onto the data bus, are met. A bus driver should be used for larger microprocessor systems where the data bus leaves the PC board and/or must drive capacitive loads larger than 100pF.

It is useful to note that in systems where the A/D converter is 1 of 8 or fewer I/O-mapped devices, no address-decoding circuitry is necessary. Each of the 8 address bits (A0 to A7) can be directly used as $\overline{CS}$ inputs, one for each I/O device.

Interfacing the Z-80 and 8085

The Z-80 and 8085 control buses are slightly different from that of the 8080. General $\overline{RD}$ and $\overline{WR}$ strobes are provided and separate memory request, $\overline{MREQ}$, and I/O request, $\overline{IORQ}$, signals have to be combined with the generalized strobes to provide the appropriate signals. An advantage of operating the A/D in I/O space with the Z-80 is that the CPU will automatically insert one wait state (the $\overline{RD}$ and $\overline{WR}$ strobes are extended one clock period) to allow more time for the I/O devices to respond. Logic to map the A/D in I/O space is shown in Figure 22. By using $\overline{MREQ}$ in place of $\overline{IORQ}$, a memory-mapped configuration results.

Additional I/O advantages exist as software DMA routines are available and use can be made of the output data transfer which exists on the upper 8 address lines (A8 to A15) during I/O input instructions. For example, MUX channel selection for the A/D can be accomplished with this operating mode.

The 8085 also provides a generalized $\overline{RD}$ and $\overline{WR}$ strobe, with an IO/$\overline{M}$ line to distinguish I/O and memory requests. The circuit of Figure 22 can again be used, with IO/$\overline{M}$ in place of $\overline{IORQ}$ for a memory-mapped interface, and an extra inverter (or the logic equivalent) to provide $\overline{IO}$/M for an I/O-mapped connection.

Interfacing 6800 Microprocessor Derivatives (6502, etc.)

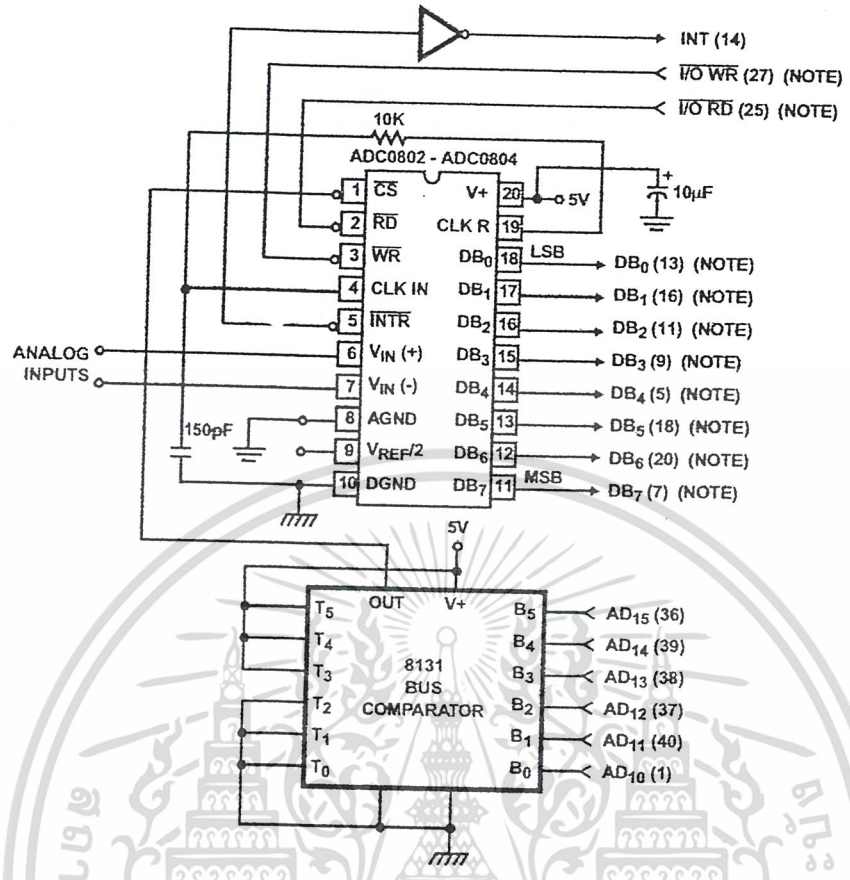
The control bus for the 6800 microprocessor derivatives does not use the RD and WR strobe signals. Instead it employs a single R/W line and additional timing, if needed, can be derived from the $\phi 2$ clock. All I/O devices are memory-mapped in the 6800 system, and a special signal, VMA, indicates that the current address is valid. Figure 23 shows an interface schematic where the A/D is memory-mapped in the 6800 system. For simplicity, the CS decoding is shown using $^1/_2$ DM8092. Note that in many 6800 systems, an already decoded $\overline{4/5}$ line is brought out to the common bus at pin 21. This can be tied directly to the CS pin of the A/D, provided that no other devices are addressed at HEX ADDR: 4XXX or 5XXX.

In Figure 24 the ADC0802 series is interfaced to the MC6800 microprocessor through (the arbitrarily chosen) Port B of the MC6820 or MC6821 Peripheral Interface Adapter (PIA). Here the CS pin of the A/D is grounded since the PIA is already memory-mapped in the MC6800 system and no CS decoding is necessary. Also notice that the A/D output data lines are connected to the microprocessor bus under program control through the PIA and therefore the A/D RD pin can be grounded.

Application Notes

NOTE #	DESCRIPTION	AnswerFAX DOC. #
AN016	"Selecting A/D Converters"	9016
AN018	"Do's and Don'ts of Applying A/D Converters"	9018
AN020	"A Cookbook Approach to High Speed Data Acquisition and Microprocessor Interfacing"	9020
AN030	"The ICL7104 - A Binary Output A/D Converter for Microprocessors"	9030

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



NOTE: Pin numbers for 8228 System Controller: Others are 8080A.

FIGURE 21. ADC0802 TO 8080A CPU INTERFACE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

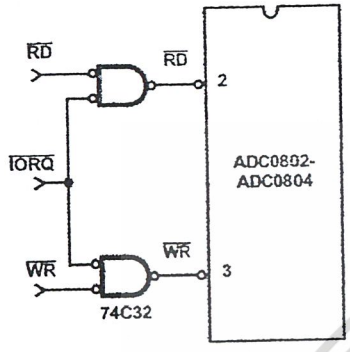
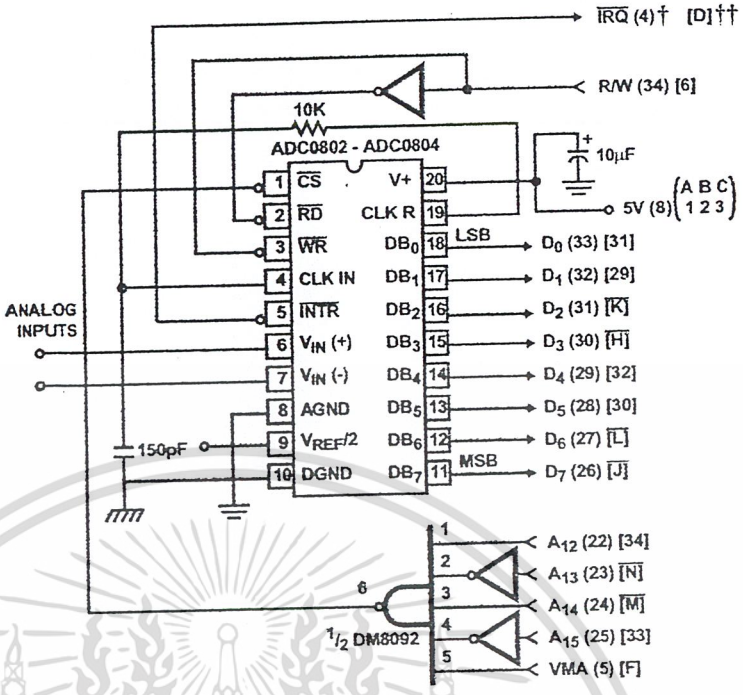


FIGURE 22. MAPPING THE A/D AS AN I/O DEVICE FOR USE WITH THE Z-80 CPU



† Numbers in parentheses refer to MC6800 CPU Pinout.
†† Numbers or letters in brackets refer to standard MC6800 System Common Bus Code.

FIGURE 23. ADC0802 TO MC6800 CPU INTERFACE

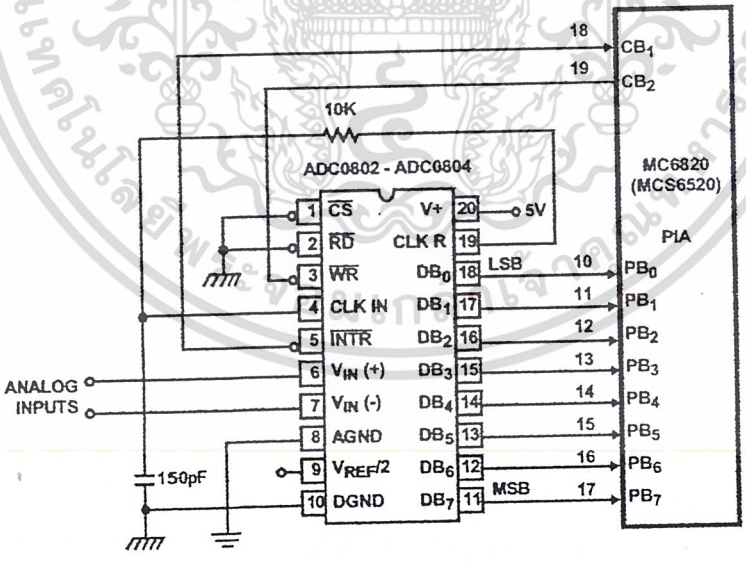


FIGURE 24. ADC0802 TO MC6820 PIA INTERFACE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Die Characteristics

DIE DIMENSIONS:

(101 mils x 93 mils) x 525µm x 25µm

METALLIZATION:

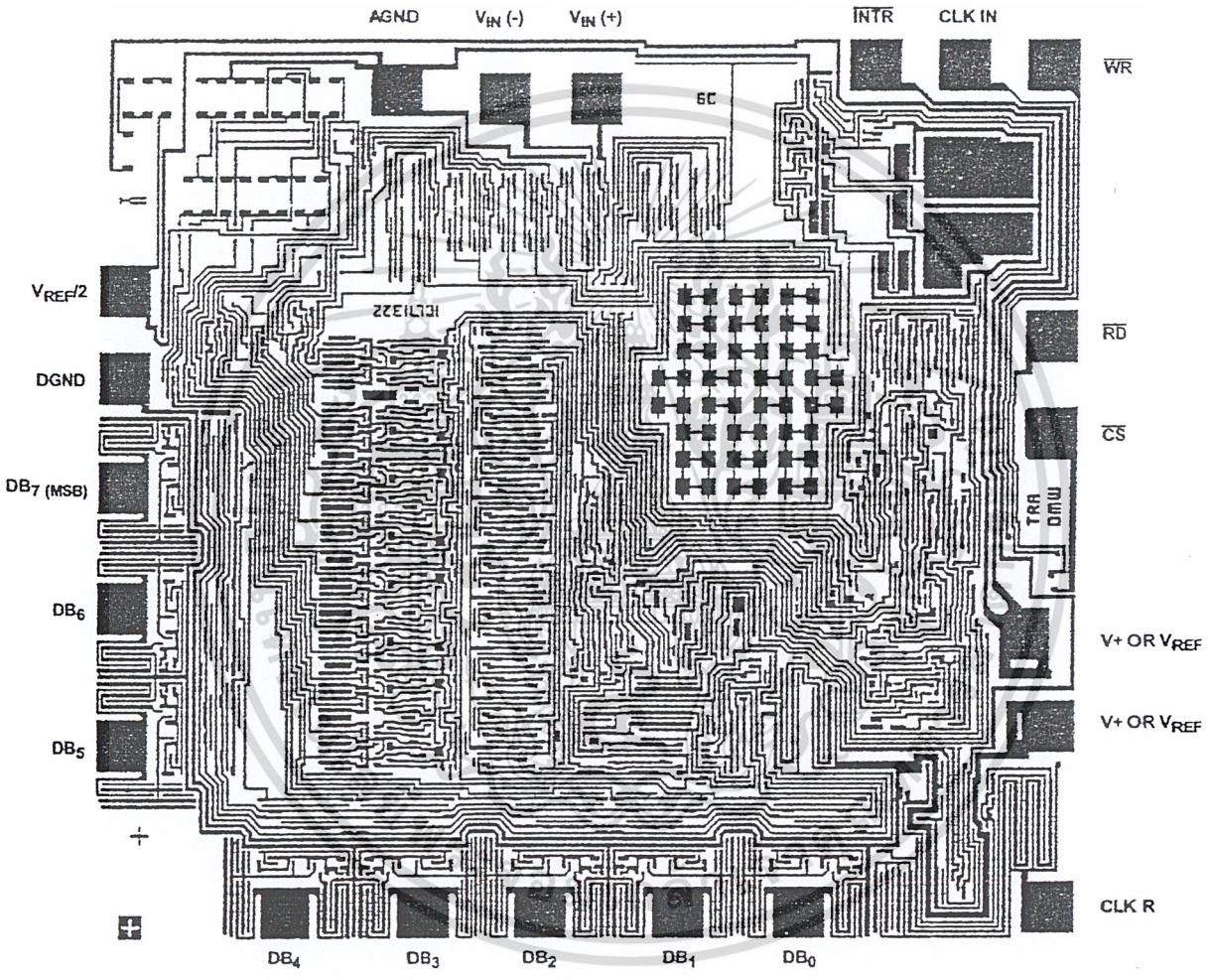
Type: Al
Thickness: 10kÅ ±1kÅ

PASSIVATION:

Type: Nitride over Silox
Nitride Thickness: 8kÅ
Silox Thickness: 7kÅ

Metallization Mask Layout

ADC0802, ADC0803, ADC0804



All Intersil semiconductor products are manufactured, assembled and tested under **ISO9000** quality systems certification.

Intersil products are sold by description only. Intersil Corporation reserves the right to make changes in circuit design and/or specifications at any time without notice. Accordingly, the reader is cautioned to verify that data sheets are current before placing orders. Information furnished by Intersil is believed to be accurate and reliable. However, no responsibility is assumed by Intersil or its subsidiaries for its use; nor for any infringements of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of Intersil or its subsidiaries.

For information regarding Intersil Corporation and its products, see web site **http://www.intersil.com**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค

LM 741

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

National Semiconductor

August 2000

LM741
Operational Amplifier

General Description

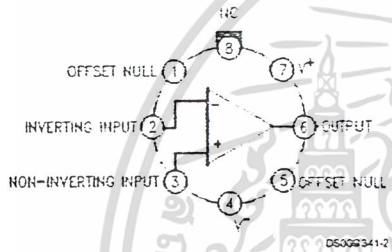
The LM741 series are general purpose operational amplifiers which feature improved performance over industry standards like the LM709. They are direct, plug-in replacements for the 709C, LM201, MC1439 and 748 in most applications.

The amplifiers offer many features which make their application nearly foolproof: overload protection on the input and output, no latch-up when the common mode range is exceeded, as well as freedom from oscillations.

The LM741C is identical to the LM741/LM741A except that the LM741C has their performance guaranteed over a 0°C to +70°C temperature range, instead of −55°C to +125°C.

Connection Diagrams

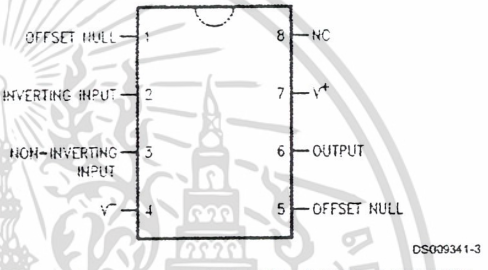
Metal Can Package



Note 1: LM741H is available per JM38510/10101

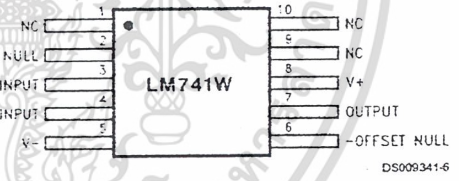
Order Number LM741H, LM741H/883 (Note 1), LM741AH/883 or LM741CH
See NS Package Number H08C

Dual-In-Line or S.O. Package



Order Number LM741J, LM741J/883, LM741CN
See NS Package Number J08A, M08A or N08E

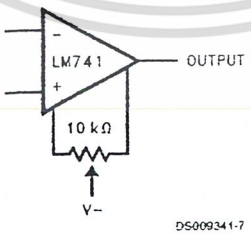
Ceramic Flatpak



Order Number LM741W/883
See NS Package Number W10A

Typical Application

Offset Nulling Circuit



© 2000 National Semiconductor Corporation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Absolute Maximum Ratings (Note 2)

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

(Note 7)

	LM741A	LM741	LM741C
Supply Voltage	±22V	±22V	±18V
Power Dissipation (Note 3)	500 mW	500 mW	500 mW
Differential Input Voltage	±30V	±30V	±30V
Input Voltage (Note 4)	±15V	±15V	±15V
Output Short Circuit Duration	Continuous	Continuous	Continuous
Operating Temperature Range	−55°C to +125°C	−55°C to +125°C	0°C to +70°C
Storage Temperature Range	−65°C to +150°C	−65°C to +150°C	−65°C to +150°C
Junction Temperature	150°C	150°C	100°C
Soldering Information			
N-Package (10 seconds)	260°C	260°C	260°C
J- or H-Package (10 seconds)	300°C	300°C	300°C
M-Package			
Vapor Phase (60 seconds)	215°C	215°C	215°C
Infrared (15 seconds)	215°C	215°C	215°C
See AN-450 "Surface Mounting Methods and Their Effect on Product Reliability" for other methods of soldering surface mount devices.			
ESD Tolerance (Note 8)	400V	400V	400V

Electrical Characteristics (Note 5)

Parameter	Conditions	LM741A			LM741			LM741C			Units
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
Input Offset Voltage	$T_A = 25°C$										
	$R_S \le 10\ k\Omega$					1.0	5.0		2.0	6.0	mV
	$R_S \le 50\Omega$		0.8	3.0							mV
	$T_{AMIN} \le T_A \le T_{AMAX}$										
	$R_S \le 50\Omega$			4.0							mV
	$R_S \le 10\ k\Omega$						6.0			7.5	mV
Average Input Offset Voltage Drift				15							µV/°C
Input Offset Voltage Adjustment Range	$T_A = 25°C, V_S = \pm 20V$	±10				±15			±15		mV
Input Offset Current	$T_A = 25°C$		3.0	30		20	200		20	200	nA
	$T_{AMIN} \le T_A \le T_{AMAX}$			70		85	500			300	nA
Average Input Offset Current Drift				0.5							nA/°C
Input Bias Current	$T_A = 25°C$		30	80		80	500		80	500	nA
	$T_{AMIN} \le T_A \le T_{AMAX}$			0.210			1.5			0.8	µA
Input Resistance	$T_A = 25°C, V_S = \pm 20V$	1.0	6.0		0.3	2.0		0.3	2.0		MΩ
	$T_{AMIN} \le T_A \le T_{AMAX}$, $V_S = \pm 20V$	0.5									MΩ
Input Voltage Range	$T_A = 25°C$							±12	±13		V
	$T_{AMIN} \le T_A \le T_{AMAX}$				±12	±13					V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Electrical Characteristics (Note 5) (Continued)

Parameter	Conditions	LM741A			LM741			LM741C			Units
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
Large Signal Voltage Gain	T_A = 25°C, $R_L \geq 2$ kΩ										
	V_S = ±20V, V_O = ±15V	50									V/mV
	V_S = ±15V, V_O = ±10V				50	200		20	200		V/mV
	$T_{AMIN} \leq T_A \leq T_{AMAX}$,										
	$R_L \geq 2$ kΩ,										
	V_S = ±20V, V_O = ±15V	32									V/mV
	V_S = ±15V, V_O = ±10V				25			15			V/mV
	V_S = ±5V, V_O = ±2V	10									V/mV
Output Voltage Swing	V_S = ±20V										
	$R_L \geq 10$ kΩ	±16									V
	$R_L \geq 2$ kΩ	±15									V
	V_S = ±15V										
	$R_L \geq 10$ kΩ				±12	±14		±12	±14		V
	$R_L \geq 2$ kΩ				±10	±13		±10	±13		V
Output Short Circuit	T_A = 25°C	10	25	35		25			25		mA
Current	$T_{AMIN} \leq T_A \leq T_{AMAX}$	10		40							mA
Common-Mode	$T_{AMIN} \leq T_A \leq T_{AMAX}$										
Rejection Ratio	$R_S \leq 10$ kΩ, V_{CM} = ±12V				70	90		70	90		dB
	$R_S \leq 50$Ω, V_{CM} = ±12V	80	95								dB
Supply Voltage Rejection	$T_{AMIN} \leq T_A \leq T_{AMAX}$,										
Ratio	V_S = ±20V to V_S = ±5V										
	$R_S \leq 50$Ω	86	96								dB
	$R_S \leq 10$ kΩ				77	96		77	96		dB
Transient Response	T_A = 25°C, Unity Gain										
Rise Time			0.25	0.8		0.3			0.3		µs
Overshoot			6.0	20		5			5		%
Bandwidth (Note 6)	T_A = 25°C	0.437	1.5								MHz
Slew Rate	T_A = 25°C, Unity Gain	0.3	0.7			0.5			0.5		V/µs
Supply Current	T_A = 25°C					1.7	2.8		1.7	2.8	mA
Power Consumption	T_A = 25°C										
	V_S = ±20V		80	150							mW
	V_S = ±15V					50	85		50	85	mW
LM741A	V_S = ±20V										
	$T_A = T_{AMIN}$			165							mW
	$T_A = T_{AMAX}$			135							mW
LM741	V_S = ±15V										
	$T_A = T_{AMIN}$					60	100				mW
	$T_A = T_{AMAX}$					45	75				mW

Note 2: "Absolute Maximum Ratings" indicate limits beyond which damage to the device may occur. Operating Ratings indicate conditions for which the device is functional, but do not guarantee specific performance limits.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
123

Electrical Characteristics (Note 5) (Continued)

Note 3: For operation at elevated temperatures, these devices must be derated based on thermal resistance, and T_j max. (listed under "Absolute Maximum Ratings"). $T_j = T_A + (\theta_{jA} P_D)$.

Thermal Resistance	Cerdip (J)	DIP (N)	HO8 (H)	SO-8 (M)
θ_{jA} (Junction to Ambient)	100°C/W	100°C/W	170°C/W	195°C/W
θ_{jC} (Junction to Case)	N/A	N/A	25°C/W	N/A

Note 4: For supply voltages less than ±15V, the absolute maximum input voltage is equal to the supply voltage.

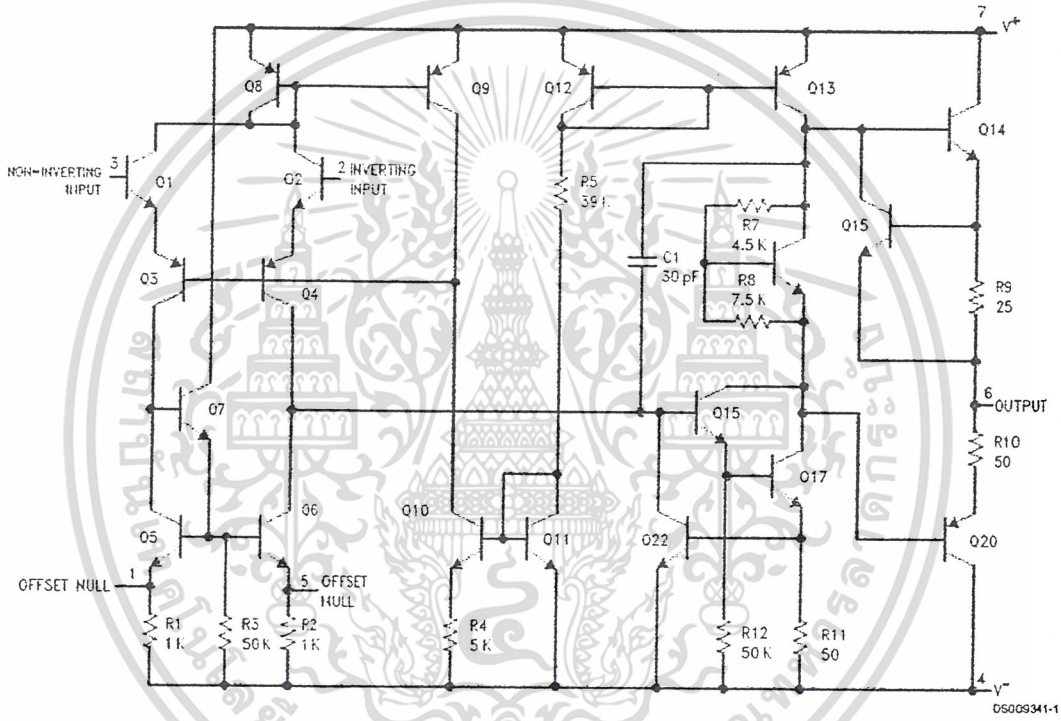
Note 5: Unless otherwise specified, these specifications apply for V_S = ±15V, −55°C ≤ T_A ≤ +125°C (LM741/LM741A). For the LM741C/LM741E, these specifications are limited to 0°C ≤ T_A ≤ +70°C.

Note 6: Calculated value from: BW (MHz) = 0.35/Rise Time(μs).

Note 7: For military specifications see RETS741X for LM741 and RETS741AX for LM741A.

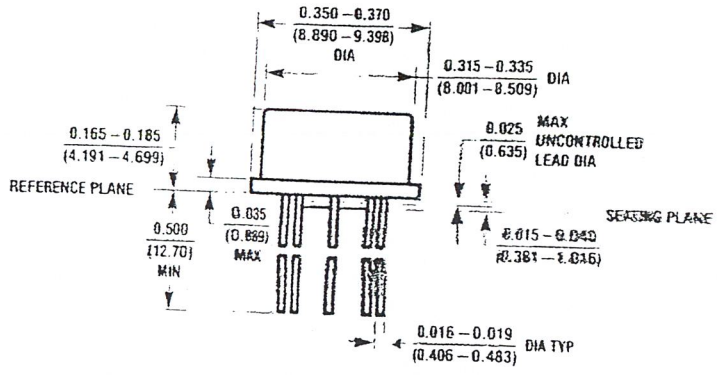
Note 8: Human body model, 1.5 kΩ in series with 100 pF.

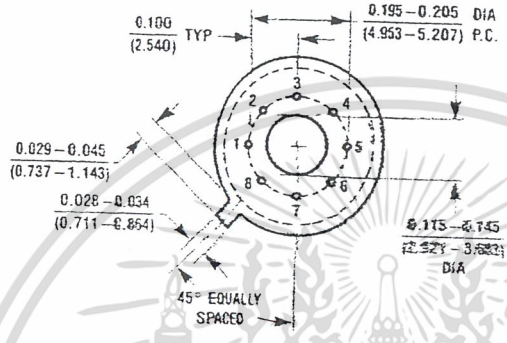
Schematic Diagram



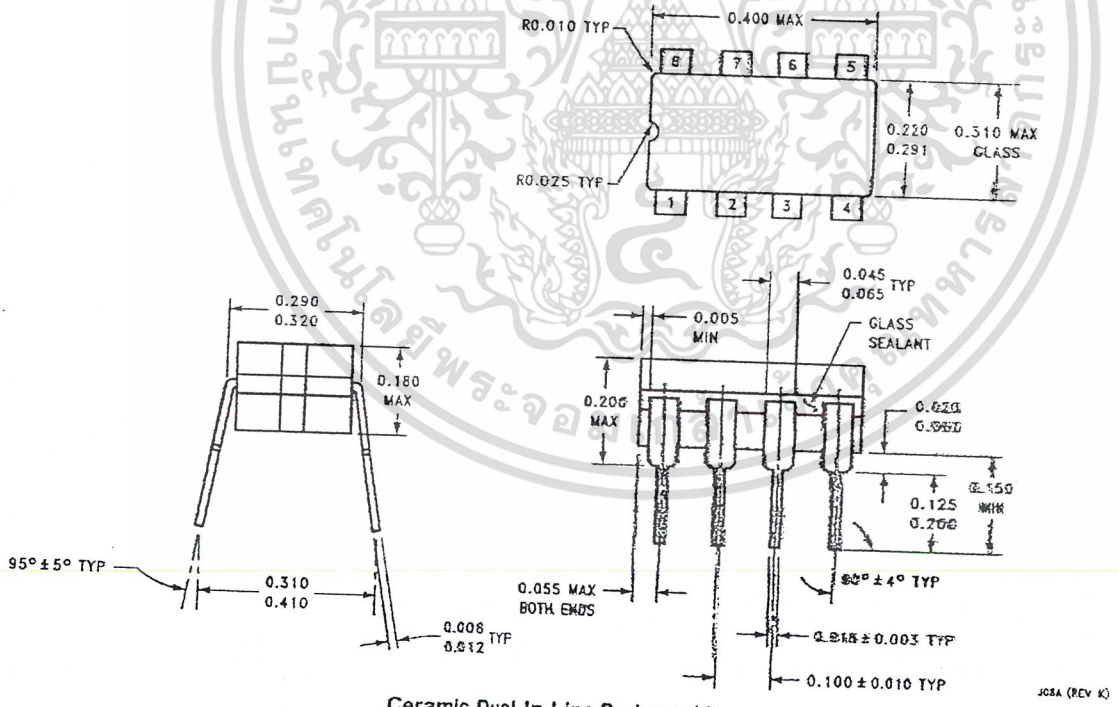
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Physical Dimensions inches (millimeters) unless otherwise noted





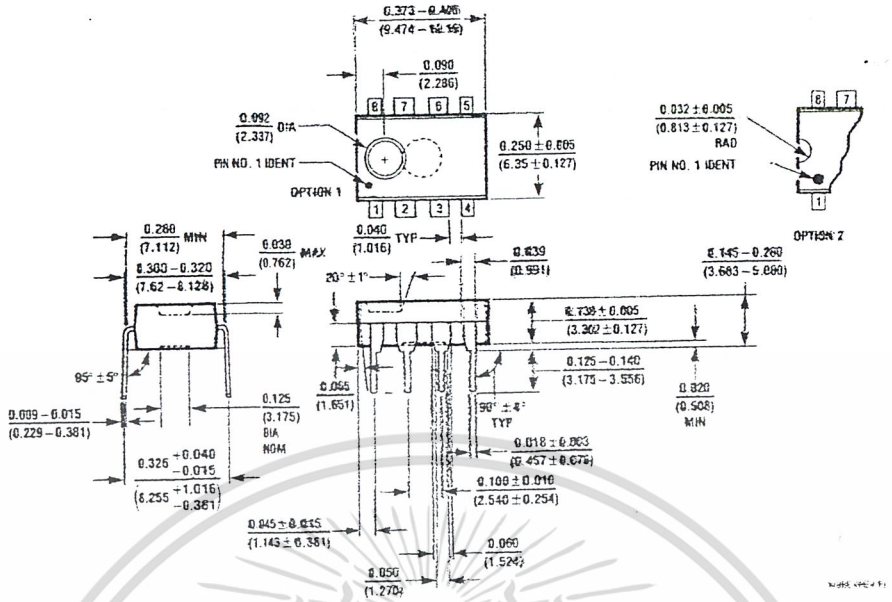
Metal Can Package (H)
Order Number LM741H, LM741H/883, LM741AH/883, LM741AH-MIL or LM741CH
NS Package Number H08C



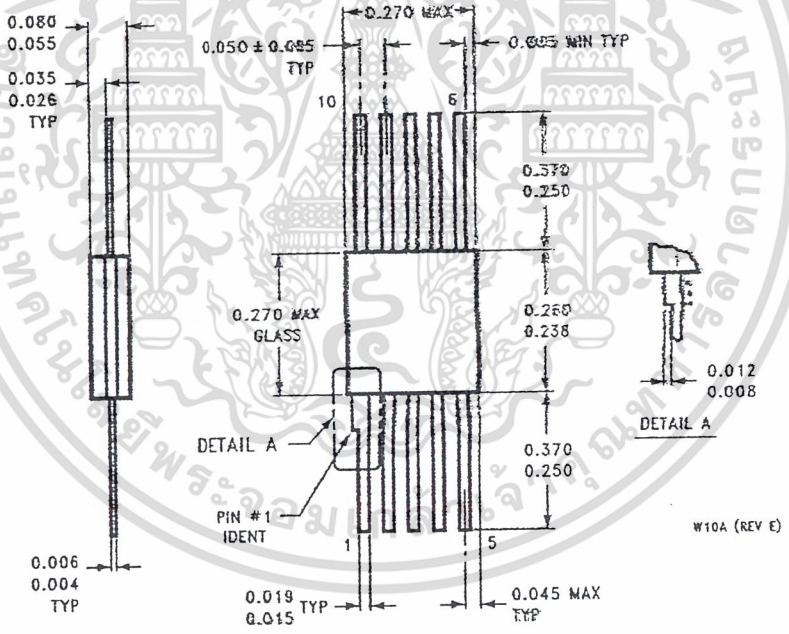
Ceramic Dual-In-Line Package (J)
Order Number LM741J/883
NS Package Number J08A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Physical Dimensions inches (millimeters) unless otherwise noted (Continued)



Dual-In-Line Package (N)
Order Number LM741CN
NS Package Number N08E



10-Lead Ceramic Flatpak (W)
Order Number LM741W/883, LM741WG-MPR or LM741WG/883
NS Package Number W10A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
126

Notes

LIFE SUPPORT POLICY

NATIONAL'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT AND GENERAL COUNSEL OF NATIONAL SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.

2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

National Semiconductor Corporation
Americas
Tel: 1-800-272-9959
Fax: 1-800-737-7018
Email: support@nsc.com
www.national.com

National Semiconductor Europe
Fax: +49 (0) 180-530 85 86
Email: europe.support@nsc.com
Deutsch Tel: +49 (0) 69 9508 6208
English Tel: +44 (0) 870 24 0 2171
Français Tel: +33 (0) 1 41 91 8790

National Semiconductor Asia Pacific Customer Response Group
Tel: 65-2544466
Fax: 65-2504466
Email: ap.support@nsc.com

National Semiconductor Japan Ltd.
Tel: 81-3-5639-7560
Fax: 81-3-5639-7507

National does not assume any responsibility for use of any circuitry described, no circuit patent licenses are implied and National reserves the right at any time without notice to change said circuitry and specifications.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนังสืออ้างอิง

1. David E. Johnson, Johnny R. Johnson, John L. Hilburn, Peter D. Scott : ELECTRIC CIRCUIT ANALYSIS , New Jersy . Pentice-Hall International Edition , 3rd Edition , 1997
2. รศ. สมยศ จุณณะปิยะ : การประยุกต์ใช้งานไมโครคอนโทรลเลอร์ ตระกูล MCS-51 , สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้