สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบควบคุมการจอดรถอัตโนมัติ

AUTOMATIC CARPARKING CONTROL

โดย

นายประเวก มโนรมย์ภัทรสาร

นายพิศุทธิพงศ์ เมฆแดง

นางสาวมณีนุช ลี้ภัยรัตน์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2546

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เลขหมู่ .....
เลขทะเบียน 54991
วันเดือนปี - 4 มิ.ย. 2548

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

ระบบควบคุมการจอดรถอัตโนมัติ

AUTOMATIC CARPARKING CONTROL

โดย

นายประเวก มโนรมย์ภัทรสาร 43010248

นายพิศุทธิพงศ์ เมฆแดง 43010308

นางสาวมณีนุช ลี้ภัยรัตน์ 43010330

อาจารย์ที่ปรึกษา

อาจารย์สุรพล บุญจันทร์

อาจารย์พิสิฐ บุญศรีเมือง

อาจารย์ธเนศ พัฒนธาดาพงษ์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2546

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2546

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง **ระบบควบคุมการจอดรถอัตโนมัติ**

AUTOMATIC CARPARKING CONTROL

ผู้จัดทำ

นายประเวก มโนรมย์ภัทรสาร	**43010248**
นายพิศุทธิพงศ์ เมฆแดง	**43010308**
นางสาวมณีนุช ลี้ภัยรัตน์	**43010330**

..... อาจารย์ที่ปรึกษา

(อ. สุรพล บุญจันทร์)

..... อาจารย์ที่ปรึกษา

(อ. พิสิฐ บุญศรีเมือง)

..... อาจารย์ที่ปรึกษา

(อ. ธเนศ พัฒนธาดาพงษ์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบควบคุมการจอดรถอัตโนมัติ

AUTOMATIC CARPARKING CONTROL

โดย นายประเวก มโนรมย์ภัทรสาร 43010248

นายพิสุทธิพงศ์ เมฆแดง 43010308

นางสาวมณีนุช ลี้ภัยรัตน์ 43010330

อาจารย์ที่ปรึกษา อ. สุรพล บุญจันทร์

อ. พิสิฐ บุญศรีเมือง

อ. ธเนศ พัฒนธาดาพงษ์

บทคัดย่อ

โครงงานนี้เป็นโครงงานที่ช่วยลดปัญหาความวุ่นวายในการหาที่จอดรถลงได้ เพราะสามารถตรวจสอบได้ว่าบริเวณนั้นมีที่ว่างให้เข้าไปจอดได้หรือไม่ โดยจะมีจอแสดงผลเป็นตัวแจ้งให้ทราบ นอกจากนี้เมื่อทำการจอดรถแล้ว ระบบเซนเซอร์ที่ติดตั้งไว้ จะทำงานแล้วส่งผลไปยังไมโครคอนโทรลเลอร์ ซึ่งจะทำหน้าที่เป็นตัวจับเวลาการจอดรถจนกระทั่งรถที่จอดออกไป เพื่อนำเวลาที่จับได้มาคำนวณหาค่าจอดรถแล้วทำการแสดงผล เพื่อทำการคิดค่าบริการต่อไป

Abstract

This project is a built for reduce problem in parking because this project can check the space available for parking and display on screen. Besides that, when have parked already, sensor system will be like a time for calculate and display parking fee from parking time.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

		หน้า
บทที่1	บทนำ	1
บทที่2	ทฤษฎีและหลักการ	2
	2.1 ไมโครคอนโทรลเลอร์ MCS-51	2
	2.1.1 หน่วยความจำภายใน	3
	2.1.2 รีจิสเตอร์ภายในไมโครคอนโทรลเลอร์ MCS–51	4
	2.1.3 ความเร็วของไมโครคอนโทรลเลอร์ MCS–51	5
	2.1.4 สายต่างๆ ของบัสและพอร์ต	5
	2.1.5 วงจรนับ/วงจรตั้งเวลา	6
	2.1.6 พอร์ตชนิดอนุกรมอยู่ภายใน	7
	2.1.7 การขัดจังหวะและชุดคำสั่ง	7
	2.1.8 AT89C51/52	7
	2.2 ทฤษฎีการขับโมดูลแสดงผลแบบผลึกเหลว	8
	2.3 โครงสร้างภายในของตัวควบคุมโมดูลแอลซีดี	8
	2.4 คำสั่งควบคุมโมดูลแอลซีดี	11
	2.4.1 คำสั่งเคลียร์ตัวแสดงผล	11
	2.4.2 คำสั่งรีเทิร์นโฮม (return home)	11
	2.4.3 คำสั่งเลือกโหมดการป้อนข้อมูล (Entry mode Set)	11
	2.4.4 คำสั่งควบคุมการเลื่อนเคอร์เซอร์และข้อมูลตัวอักษร	12
	2.4.5 กำหนดฟังก์ชันการทำงาน	12
	2.4.6 คำสั่งเลือกแอดเดรสของซีจีแรม(CGRAM)	12
	2.4.7 คำสั่งเลือดแอดเดรสของดีดีแรม(DDRAM)	13
	2.4.8 การเขียนคำสั่งและข้อมูลให้แก่โมดูลแอลซีดี	13
	2.4.9 จังหวะการทำงานของแอลซีดีโมดูล	14
	2.5 ทฤษฎี DS1307 ไอซีสร้างฐานเวลาหรือเรียลไทม์คล็อก (real time clock)	15
	2.5.1 การทำงานของ DS1307	15
	2.5.2 การจัดสรรหน่วยความจำใน DS1307	16
	2.5.3 รีจีสเตอร์ควบคุม	17
	2.5.4 โหมดการทำงานของ DS1307	18
	2.5.5 โหมดการเขียนข้อมูล	18
	2.5.6 โหมดการอ่านข้อมูล	18
	2.5.7 การเชื่อมต่อ DS1307 กับไมโครคอนโทรลเลอร์ MCS-51	19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่3 การคำนวณและการสร้าง 20
3.1 โครงสร้างทางฮาร์ดแวร์ 20
3.1.1 ส่วนของวงจรเซนเซอร์ 20
3.1.2 ส่วนของไมโครคอนโทรลเลอร์ 21
3.2 โครงสร้างทางซอฟท์แวร์ 23
3.2.1 โฟว์ชาร์ทแสดงการทำงานของโปรแกรม 23
3.2.2 โปรแกรมที่ใช้ในการทำงาน 27
3.2.3 คำอธิบายโปแกรมย่อยของการใช้งานโมดูลแอลซีดี 76
3.2.4 คำอธิบายโปรแกรมย่อยของการติดต่อบนระบบบัส I^2C 76
บทที่4 การทดลองและผลการทดลอง 78
4.1 ขั้นตอนการทดลอง 78
4.2 ผลการทดลอง 78
4.2.1 ผลการทดลองของวงจรเซนเซอร์ทั้งด้านส่งและด้านรับ 78
4.2.2 ผลการทดลองในส่วนของซอฟท์แวร์ 79
บทที่5 บทวิจารณ์และบทสรุป 83
5.1 สรุปผลการทดลอง 83
5.1.1 ปัญหาของโครงงาน 83
5.1.2 ข้อจำกัดของโครงงาน 83

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

หน้า

รูปที่ 1.1 แสดงบล็อกไดอะแกรมของโครงงาน 1
รูปที่ 2.1 โครงสร้างภายในพื้นฐานของไมโครคอนโทรลเลอร์ตระกูล MCS-51 เบอร์ 8051 2
รูปที่ 2.2 การจัดวางขาต่างๆ ของไมโครคอนโทรลเลอร์ MCS–51 5
รูปที่ 2.3 โครงสร้างภายในหน่วยความจำของไมโครคอนโทรลเลอร์ MCS–51 6
รูปที่ 2.4 ไดอะแกรมการทำงานของโมดูลแอลซีดี แบบอักขระ 9
รูปที่ 2.5 จอแสดงผล แอลซีดี 10
รูปที่ 2.6 วงจรสำหรับทดลองการแสดงข้อความบนโมดูลแอลซีดี 16 ตัวอักษร 1 บรรทัด 14
รูปที่ 2.7 การจัดขาของ IC DS1307 15
รูปที่ 2.8 วงจรภายใน DS1307 16
รูปที่ 2.9 แสดงการจัดสรรหน่วยความจำ 17
รูปที่ 2.10 รูปแบบข้อมูลสำหรับติดต่อกับ DS1307 ในโหมดการเขียนข้อมูล 18
รูปที่ 2.11 รูปแบบข้อมูลสำหรับติดต่อกับ DS1307 ในโหมดการอ่านข้อมูล 19
รูปที่ 2.12 แสดงการเชื่อมต่อไมโครคอนโทรลเลอร์ MCS-51 กับไอซีเรียลไทม์คล็อก DS1307 19
รูปที่ 3.1 แสดงบล็อกไดอะแกรมของโครงงาน 20
รูปที่ 3.2 แสดงวงจรอินฟราเรดทางด้านส่ง 21
รูปที่ 3.3 แสดงวงจรอินฟราเรดทางด้านรับ 21
รูปที่ 3.4 แสดงรูปการต่อวงจรไมโครคอนโทรลเลอร์ เพื่อควบคุมวงจรต่างๆ 22
รูปที่ 3.5 แสดงโฟว์ชาร์ทโปรแกรมหลักการควบคุมการทำงานของไมโครคอนโทรลเลอร์ตัวที่หนึ่ง 23
รูปที่ 3.6 แสดงโฟว์ชาร์ทโปรแกรมหลักการควบคุมการทำงานของไมโครคอนโทรลเลอร์ตัวที่สอง 24
รูปที่ 3.7 แสดงโฟว์ชาร์ท ของโปรแกรมย่อยในการเขียนค่าเวลาลงในไอซี 1307 25
รูปที่ 3.8(ก) แสดงโฟว์ชาร์ท ของโปรแกรมย่อยในการอ่านค่าเวลาจากไอซี 1307 26
รูปที่ 3.8(ข) แสดงโฟว์ชาร์ท ของโปรแกรมย่อยในการแสดงผลออกจอแอลซีดี 26
รูปที่ 4.1 แสดงสัญญาณที่ขา 3 ของไอซี 555 78
รูปที่ 4.2 แสดงสัญญาณทางด้านรับของวงจรเซนเซอร์ 78
รูปที่ 4.3 แสดงที่จอดรถขณะยังไม่มีรถจอด 79
รูปที่ 4.4 แสดงสถานะที่จอดรถว่างทางแอลอีดี 79
รูปที่ 4.5 แสดงที่จอดรถขณะที่มีรถเข้ามาจอดช่องที่ 1,2 และ 3 80
รูปที่ 4.6 แสดงสถานะการจอดรถทางแอลอีดี 80
รูปที่ 4.7 แสดงสถานะที่จอดรถเต็มทางแอลอีดี 81
รูปที่ 4.8 การแสดงตำแหน่งและราคาค่าจอดรถออกทางจอแอลซีดี 81
รูปที่ 4.9 วงจรรวมของระบบควบคุมการจอดรถอัตโนมัติ 82

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่ 2.1 จำนวนหน่วยความจำภายในไมโครคอนโทรลเลอร์ MCS–51 4

ตารางที่ 2.2 แสดงความสัมพันธ์ในการทำงานของขา RS,R/W และ E ของโมดูลแอลซีดีแบบอักขระ 10

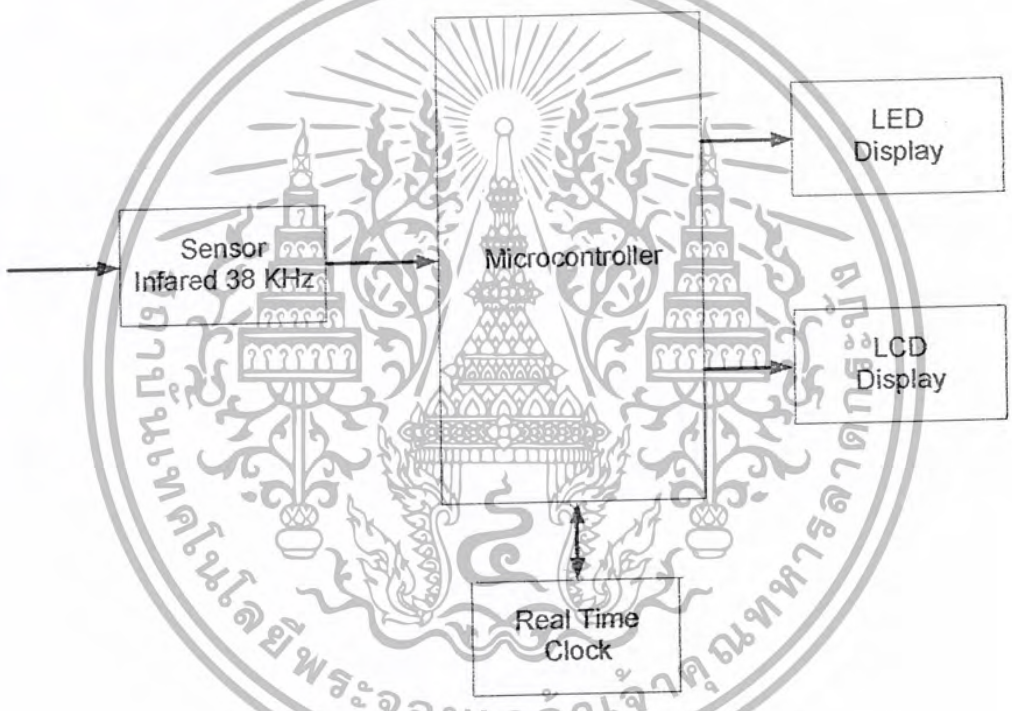
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

ปัจจุบันนี้การหาที่จอดรถในเขตที่มีประชากรคับคั่งนั้นไม่ใช่เรื่องง่ายแล้ว โดยเฉพาะในชั่วโมงเร่งด่วน ซึ่งจะทำให้เสียเวลาอันมีค่าไปโดยเปล่าประโยชน์

โครงงานนี้สามารถนำมาใช้เพื่อลดปัญหาเหล่านี้ลงได้ โดยระบบจะทำการนับรถทุกคันที่เข้ามาจากเซนเซอร์ที่ตั้งไว้ที่ทางเข้าเมื่อ มีรถเข้ามาครบตามจำนวนที่จอดรถที่มี ระบบจะแสดงสัญญาณไฟสีแดงขึ้นที่ทางเข้า เพื่อไม่ให้รถที่เหลือเข้ามา เพราะจะเป็นการเสียเวลา อีกทั้งระบบยังมีเซนเซอร์ บริเวณที่จอดรถอีกเพื่อช่วยในการคิดค่าจอดรถอีกด้วย

โครงงานนี้เป็นการนำไมโครคอนโทรลเลอร์มาประยุกต์ใช้ในการทำงานของระบบที่จอดรถอัตโนมัติ โดยที่มีขบวนการดังบล็อกไดอะแกรมข้างล่างนี้



รูปที่ 1.1 แสดงบล็อกไดอะแกรมของโครงงาน

เมื่อมีการตรวจจับสัญญาณได้ว่ามีรถเข้ามาจอดก็จะทำการส่งสัญญาณไปยัง ส่วนของการควบคุมและประมวลผล เพื่อที่จะให้ส่งสัญญาณแสดงผลออกทางจอแสดงผล แอลอีดี(LED) จากนั้นก็จะทำการนำเวลาจริงจากส่วนของฐานเวลาจริง(Real Time Clock) มาเก็บไว้

และเมื่อมีการตรวจจับสัญญาณได้ ว่ารถที่จอดได้ออกไปแล้วก็จะทำการส่งสัญญาณไปยัง ส่วนของการควบคุมและประมวลผล เพื่อที่จะให้ส่งสัญญาณแสดงผลออกทางจอแสดงผลแอลอีดี จากนั้นก็จะทำการนำเวลาจริงจากส่วนของฐานเวลาจริง(Real Time Clock) มาทำการลบกับค่าที่เก็บไว้ในตอนที่รถมาจอด แล้วจึงทำการแสดงผลออกมาทางจอแสดงผลแอลซีดี(LCD)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

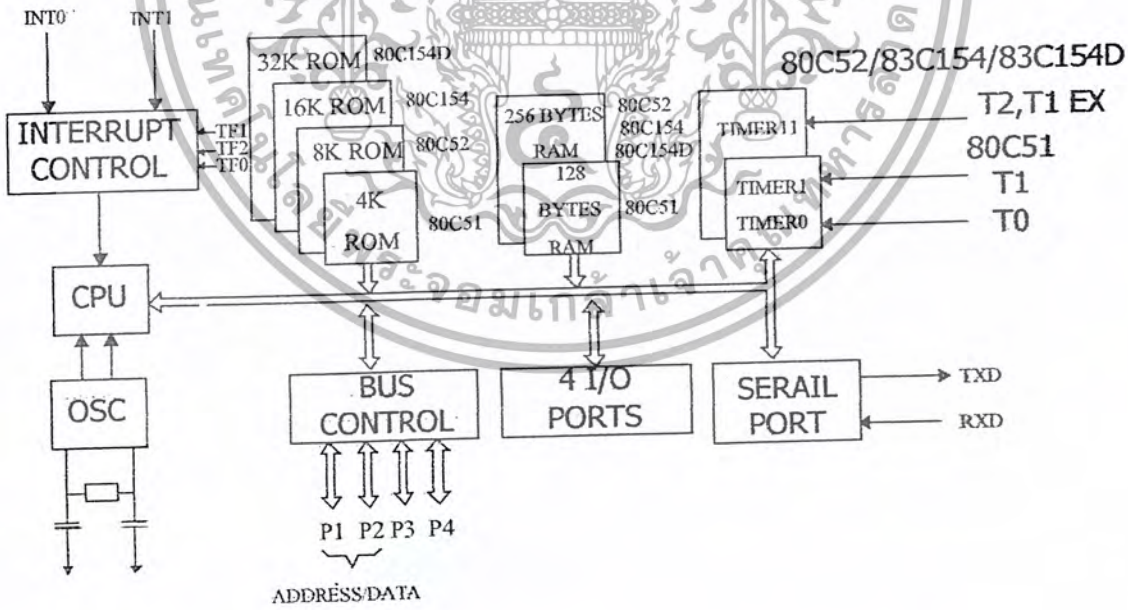
บทที่ 2
ทฤษฎีและหลักการ

2.1 ไมโครคอนโทรลเลอร์ MCS-51

ไมโครคอนโทรลเลอร์ในตระกูล MCS-51 เป็นไมโครคอนโทรลเลอร์ขนาด 8 บิต ที่มีอุปกรณ์สนับสนุนประกอบอยู่ภายใน ได้แก่ หน่วยความจำสำหรับเก็บข้อมูล หน่วยความจำสำหรับเก็บโปรแกรม ตัวตั้งเวลา/ตัวนับ อุปกรณ์รับ-ส่งข้อมูลแบบอนุกรม ซึ่งทำให้การใช้งานง่ายและมีประสิทธิภาพมากขึ้น โดยไม่ต้องมีการเชื่อมต่ออุปกรณ์ภายนอกมาเพิ่มเติมมากเหมือนกับ ไมโครโปรเซสเซอร์ทั่วไปนอกจากนี้ หากต้องการใช้งานไมโครคอนโทรลเลอร์ร่วมกับอุปกรณ์อื่นเพิ่มเติม เช่น ไมโครโปรเซสเซอร์ 8255 หรือ หน่วยความจำภายนอก เป็นต้น

โครงสร้างภายในพื้นฐานของไมโครคอนโทรลเลอร์ตระกูล MCS-51 เบอร์ 8051 ประกอบด้วย ส่วนต่างๆดังต่อไปนี้

- หน่วยความจำสำหรับเก็บข้อมูลจำนวน 128 ไบต์
- หน่วยความจำภายในสำหรับเก็บโปรแกรมขนาด 4 กิโลไบต์
- อุปกรณ์ควบคุมการขัดจังหวะ (Interrupt)
- ตัวตั้งเวลาและตัวนับขนาด 16 บิต 2 ชุด
- พอร์ตควบคุมการสื่อสารอนุกรมดูเพลกซ์เต็มซึ่งสามารถรับส่งข้อมูลได้พร้อมกัน
- พอร์ตขนานสำหรับติดต่ออุปกรณ์ภายนอกได้ 4 พอร์ตๆละ 8 บิต
- วงจรผลิตสัญญาณนาฬิกาภายใน



รูปที่ 2.1 โครงสร้างภายในพื้นฐานของไมโครคอนโทรลเลอร์ตระกูล MCS-51 เบอร์ 8051

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในปัจจุบันมีไมโครคอนโทรลเลอร์ MCS–51 ซึ่งเป็นไมโครคอมพิวเตอร์แบบชิปเดี่ยว ไม่ต้องต่ออุปกรณ์ภายนอกสามารถทำงานได้มีความสะดวกในการใช้งานและเขียนโปรแกรมควบคุมด้วยภาษาเบสิกได้ โดยไม่ต้องศึกษาการทำงานของวงจรเหมือนกับภาษาแอสเซมบลี นอกจากนั้นเป็นอุปกรณ์ที่ออกแบบมาสนองความต้องการของผู้ใช้คือมีอินพุทและเอาท์พุทภายในตัวเอง พอร์ต(port)ของอินพุท(input)และการเชื่อมต่อเอาท์พุท(output)จากบัฟเฟอร์(buffer)และสายควบคุมอื่นๆ ที่ใช้สำหรับแยกข้อมูลกับตำแหน่งที่อยู่และชุดคำสั่งเพิ่มขึ้นเป็นพิเศษและเพื่อจัดการข้อมูลด้วยวงจรตั้งเวลากับวงจรนับด้วย ปกติวงจรนับจะสามารถทำงานเป็นวงจรตั้งเวลาได้ด้วย จึงเรียกควบคู่กันไปคือวงจรตั้งเวลา/วงจรนับ

ไมโครคอนโทรลเลอร์ MCS–51 ประกอบด้วยคุณสมบัติดังนี้

- หน่วยประมวลผลกลาง 8 บิตที่ควบคุมได้ง่าย
- เพิ่มการทำงานลอจิกครั้งละ 1 บิตได้
- สายอินพุทและเอาท์พุทมีจำนวน 32 เส้นใช้เลือกตำแหน่งแยกต่างหากจากกันได้
- มีแรมบรรจุไว้ภายในขนาด 128 ไบต์หรือ 256 ไบต์
- วงจรตั้งเวลา/วงจรนับมีขนาด 2, 3 หรือ 16 บิต
- กำหนดรับส่งข้อมูลอนุกรมได้สองทิศทาง (Universal Synchronous Asynchronous Receiver Transmitter ; UART)
- การขัดจังหวะแบ่งออกเป็น 2 ระดับจาก 5 หรือ 6 แหล่ง
- มีสัญญาณนาฬิกาอยู่ภายในตัว
- มีหน่วยความจำข้อมูลภายใน4หรือ8กิโลไบต์(หน่วยความจำแบบอีอีพรอมเบอร์8051และเบอร์8052)
- มีตำแหน่งของหน่วยความจำสำหรับเก็บโปรแกรมจำนวนทั้งหมด 64 กิโลไบต์
- คำสั่งมีทั้งหมด 111 คำสั่ง (64 รอบ)
- ทำงานด้วยเลขฐานสิบและเลขฐานสิบหก
- ตัวแปลภาษาเบสิกมีขนาด 8 กิโลไบต์ (8052–AH–BASIC)
- ควบคุมหน่วยความจำอีอีพรอมภายในตัวด้วยภาษาเบสิก (8052–AH–BASIC) โดยไม่ต้องส่งให้โรงงานโปรแกรมให้ เพราะสามารถจะเขียนและทดสอบโปรแกรมด้วยหน่วยความจุภายนอกแทน ไม่มีรอมภายใน

2.1.1 หน่วยความจำภายใน

ถ้าต้องการป้องกันการอ่านจากภายนอก ก็สามารถเลือกไมโครคอนโทรลเลอร์ 8051 และ 8052 มีหน่วยความจำแบบอีอีพรอมภายในที่จัดค่าเวลาการอ่านข้อมูล เพื่อกันการลอกเลียนแบบโปรแกรม โครงสร้างภายในหน่วยความจำของไมโครคอนโทรลเลอร์ MCS–51 หน่วยความจำนี้สามารถแบ่งได้ 2 กลุ่ม คือหน่วยความจำสำหรับเก็บโปรแกรมและหน่วยความจำสำหรับเก็บข้อมูล หน่วยความจำกลุ่มแรกมีความจุต่ำกว่า 4 หรือ 6 กิโลไบต์บรรจุอยู่ในรอมซึ่งส่วนไมโครคอนโทรลเลอร์ MCS–51ที่ไม่มีรอมภายในและอาจใช้หน่วยความจำภายนอกซึ่งอาจเป็นรอม,แรมหรืออีพรอม ไมโครคอนโทรลเลอร์ MCS–51 จะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อ่านหน่วยความจำสำหรับเก็บข้อมูล จะใช้เป็นที่เก็บตัวแปร การคำนวณหาผลลัพธ์ได้ทันที การจัดการกับข้อมูลที่มีขนาด 16 บิต หรือ ตารางที่2.1 ใช้ค้นหาค่าต่างๆและหน้าที่อื่นที่คล้ายกัน หน่วยความจำสำหรับเก็บข้อมูลโดยใช้ร่วมกับหน่วยความจำภายนอกได้ถึง 64 กิโลไบต์ ซึ่งเลือกใช้รอมหรือแรมก็ได้และยังมีหน่วยความจำข้อมูลภายในให้ใช้อีก 128 หรือ 256ไบต์ ขึ้นอยู่กับเบอร์ของไอซี

เบอร์	หน่วยความจำโปรแกรมภายใน	หน่วยความจำโปรแกรมภายใน	ตัวจับเวลา/ตัวนับ	การขัดจังหวะการทำงาน
8052AH	ROM 8 กิโลไบต์	RAM 256 ไบต์	3/16 บิต	6
8051	ROM 4 กิโลไบต์	RAM 128 ไบต์	2/16 บิต	5
8051	ROM 4 กิโลไบต์	RAM 128 ไบต์	2/16 บิต	5
8032AH	-	RAM 256 ไบต์	3/16 บิต	6
8031AH	-	RAM 128 ไบต์	2/16 บิต	5
8031	-	RAM 128 ไบต์	2/16 บิต	5
8751H	ROM 4 กิโลไบต์	RAM 128 ไบต์	2/16 บิต	5
8752H	ROM 4 กิโลไบต์	RAM 256 ไบต์	3/16 บิต	6

ตารางที่ 2.1 จำนวนหน่วยความจำภายในไมโครคอนโทรลเลอร์ MCS–51

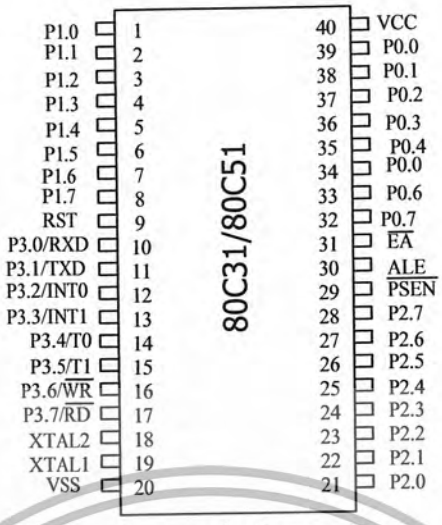
โครงสร้างภายในหน่วยความจำของไมโครคอนโทรลเลอร์ MCS–51 มีดังนี้

Internal = หน่วยความจำภายใน
External = หน่วยความจำภายนอก
Program Counter = วงจรนับลำดับโปรแกรม
Program Memory = หน่วยความจำสำหรับเก็บโปรแกรม
Internal Data Ram = แรมภายในที่เก็บข้อมูล
Special Function Register = รีจิสเตอร์หน้าที่พิเศษ
Internal Data Memory = หน่วยความจำภายในสำหรับเก็บข้อมูล
External Data Memory = หน่วยความจำภายนอกสำหรับเก็บข้อมูล

2.1.2 รีจิสเตอร์ภายในไมโครคอนโทรลเลอร์ MCS–51

ไมโครคอนโทรลเลอร์ MCS–51 จะมีรีจิสเตอร์ซึ่งอำนวยความสะดวกในการใช้งานตามคำสั่งต่างๆ และมีแอคคิวมูเลเตอร์ รีจิสเตอร์ B ซึ่งใช้ในการคูณและหาร รีจิสเตอร์สถานะ เป็นตัวชี้ตำแหน่งข้อมูล 2*8 บิตและ 1*16 บิต พอร์ตหมายเลขศูนย์ถึงพอร์ตหมายเลขสามและมีรีจิสเตอร์เป็นตัวชี้ข้อมูล ซึ่งใช้ส่งและรับข้อมูลแบบอนุกรม มีรีจิสเตอร์ 16 บิต ที่เป็นวงจรตั้งเวลาและวงจรนับรีจิสเตอร์ซึ่งจะจองไว้สำหรับใช้สำหรับตัวที่ 3 เป็นรีจิสเตอร์คำสั่งสำหรับหน้าที่พิเศษ เช่น การขัดจังหวะ เรียลไทม์คล็อก (Real Time Clock : RTC) เป็นต้น และอินพุท, เอาท์พุทแบบอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 การจัดวางขาต่างๆ ของไมโครคอนโทรลเลอร์ MCS–51

2.1.3 ความเร็วของไมโครคอนโทรลเลอร์ MCS–51

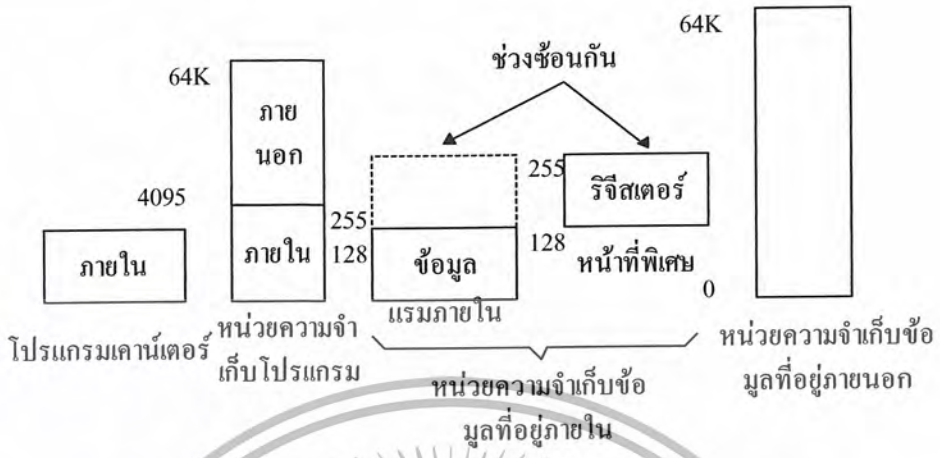
เมื่อเทียบกับไมโครคอนโทรลเลอร์ 6502 และไมโครโปรเซสเซอร์ Z80 คำสั่งทั้งหมดของไมโครคอนโทรลเลอร์ MCS–51 จะทำงานด้วยรอบคำสั่งเดียวกับภาษาเครื่อง คือภายในระยะเวลาของสัญญาณนาฬิกา 12 ลูกและเมื่อใช้ความถี่ 12 เมกะเฮิรตซ์ ดังนั้นแล้วหนึ่งรอบคำสั่งของภาษาเครื่องจะเท่ากับ 1 ไมโครวินาที ซึ่งความเร็วในการประมวลผลของไมโครคอนโทรลเลอร์ MCS–51 เท่ากับไอซี 6502 หน่วยประมวลผลกลางซึ่งใช้ความถี่ 2 เมกะเฮิรตซ์ หรือจะเท่ากับไมโครโปรเซสเซอร์ Z80 มีหน่วยประมวลผลกลางที่จะใช้ความถี่ 8 บิต ต่อหนึ่งครั้ง เพื่อให้ความสะดวกในการเขียนโปรแกรมโดยไม่ต้องลำบากในการปิดกั้น (Mask) บิตที่ไม่ต้องการ

2.1.4 สายต่างๆ ของบัสและพอร์ต

โครงสร้างภายในของไมโครคอนโทรลเลอร์เบอร์ MCS–51 โดยให้มีบัสสองทิศทาง 4 เส้น และพอร์ตขนาด 8 บิตตามทฤษฎี แต่ที่จริงแล้วนั้นจะใช้ได้ก็ต่อเมื่อมีการใช้หน่วยความจำ และภายในตัวรอมหรือแรม เมื่อไม่ใช้หน่วยความจำภายใน พอร์ต 0 และ 2 ใช้เป็นบัสของข้อมูลและตำแหน่ง ดังนั้น พอร์ต 2 พอร์ต ใช้งานเป็นอินพุทและเอาท์พุท ซึ่งพอร์ต 2 เป็นสายสัญญาณ ตำแหน่ง A15...A8 ส่วน พอร์ต 0 ทำหน้าที่เป็นสายสัญญาณตำแหน่ง A7...A0 ออกจากบิตข้อมูล D7...D0 เอาท์พุทของขา RD และ WR มาจากสายสัญญาณเอาท์พุทของพอร์ต P3 โดยโปรแกรมภายในใช้สัญญาณ RD และ WR เพื่อการเขียนข้อมูลและอ่านข้อมูลกับหน่วยความจำภายนอก และจะไม่ทำงานเมื่อมีภาษาเครื่องเก็บอยู่ในหน่วยความจำภายในและถ้าหากขา PSEN เป็นขารับสัญญาณสำหรับเปิดให้มีการอ่านหน่วยความจำภายนอก ถ้าสังเกตทุกๆรอบคำสั่งระหว่างการทำงานด้วยโปรแกรมในรอมหรือหน่วยความจำแบบอีอีพรอม สัญญาณ PSEN ทำงานถึงสองครั้งเหมือนสัญญาณ ALE เพราะว่ามีการอ่านข้อมูลจำนวน 2 ไบต์ และในแต่ละรอบคำสั่งขา PSEN นี้ ซึ่งจะไม่ทำงานเมื่อมีภาษาเครื่องเก็บอยู่ในหน่วยความจำภายใน ถ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน่วยความจำภายนอกไม่มีข้อมูลบรรจุอยู่ขา PSEN ก็จะไม่ทำงานเช่นกันและส่วนไมโครคอนโทรลเลอร์เบอร์ 8052AH จะไม่ใช้สัญญาณขา PSEN เลย



รูปที่ 2.3 โครงสร้างภายในหน่วยความจำของไมโครคอนโทรลเลอร์ MCS–51

ขา EA เป็นขาอินพุทที่จะใช้ร่วมกับตำแหน่งสัญญาณภายนอก โดยจะมีค่าเป็นลอจิก “0” และเมื่อไมโครโปรเซสเซอร์ทำการอ่านคำสั่งจากหน่วยความจำภายนอกซึ่งโดยปกติแล้วไมโครคอนโทรลเลอร์จะอ่านหน่วยความจำภายในน้อยกว่าอ่านจากหน่วยความจำภายนอก ขา EA ยังเป็นขาอินพุทสำหรับป้อนแรงดันไฟฟ้า 21 โวลต์ เพื่อจะเขียนโปรแกรมให้กับหน่วยความจำแบบอีอีพรอม และสำหรับกรณีที่ใช้ไมโครคอนโทรลเลอร์ 8051 หรือไมโครคอนโทรลเลอร์ 8052

2.1.5 วงจรนับ/วงจรตั้งเวลา

จากตารางที่ 2.1 ไมโครคอนโทรลเลอร์เบอร์ 8052 มีวงจรนับและเป็นวงจรตั้งเวลาชนิด 16 บิต มากกว่าไมโครคอนโทรลเลอร์เบอร์ 8051 อยู่หนึ่งตัว ในการทำงานของวงจรนับและวงจรตั้งเวลาเป็นดังนี้ เมื่อทำงานเป็นวงจรตั้งเวลา รีจิสเตอร์วงจรนับจะเพิ่มขึ้นหนึ่งเมื่อมีสัญญาณป้อนให้ทางอินพุท T0, T1 หรือ T2 มีเฉพาะไมโครคอนโทรลเลอร์ 8052 เป็นขอบสัญญาณขาลง อัตราการนับสัญญาณสูงสุดคือ 1/24 ของความเร็วสัญญาณนาฬิกาของไมโครคอนโทรลเลอร์ วงจรนับและวงจรตั้งเวลา 0 และ 1 มีวิธีโปรแกรมให้ทำงานได้ต่างกันถึง 4 แบบ ซึ่งรวมทั้งการทำงานเป็น 8 บิต หรือ 16 บิต และการบรรจุค่าพรีเซตหนึ่งค่าได้เองอย่างอัตโนมัติ วงจรนับและวงจรตั้งเวลาที่ 1 ซึ่งเลือกโปรแกรมให้ทำหน้าที่เป็นตัวกำเนิดสัญญาณของอัตราการส่งบิตออกไปยังพอร์ตอนุกรม สำหรับใช้เชื่อมต่อ วงจรนับและวงจรตั้งเวลาที่ 2 เฉพาะไมโครคอนโทรลเลอร์ 8052 เท่านั้นและมีการทำงานย่อยๆ อีก 3 ชนิด ดังนี้

1. วงจรนับ 16 บิตที่สามารถโหลดค่ากลับคืนเองอย่างอัตโนมัติ
2. วงจรนับที่จองไว้ชนิด 16 บิต
3. วงจรกำเนิดสัญญาณของการส่งบิต เพื่อใช้ในการเชื่อมโยงข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. วงจรกำเนิดสัญญาณของการส่งบิต เพื่อใช้ในการเชื่อมโยงข้อมูล

2.1.6 พอร์ตชนิดอนุกรมอยู่ภายใน

ไมโครคอนโทรลเลอร์ทั้งหมดในตระกูล MCS–51 เป็นไมโครคอนโทรลเลอร์ ที่มีการเชื่อมโยงข้อมูลชนิดสองทาง ทำให้รับและส่งข้อมูลพร้อมกัน ตัวรับส่งข้อมูลชนิดอะซิงโครนัส (Asynchronous Receiver) มีบัฟเฟอร์สำหรับข้อมูลเป็นพิเศษเพื่อเพิ่มความเร็วในการสื่อสารพอร์ตชนิดอนุกรมนั้นสามารถเลือกโปรแกรม เพื่อเลือกใช้การทำงานแบบใดแบบหนึ่งใน 4 แบบด้วยการใช้โปรแกรมควบคุมอัตราการส่งข้อมูลและรูปแบบของข้อมูล อัตราการส่งข้อมูลที่เลือกใช้ได้สูงถึง 19,200 บิต/วินาที ด้วยความเร็วของสัญญาณนาฬิกา 1 เมกะเฮิรตซ์ สำหรับใช้ในระบบเครือข่าย (Networks) และระบบการสื่อสารของไมโครคอนโทรลเลอร์ หลายตัวร่วมกันจะเลือกความเร็วของสัญญาณนาฬิกาด้วยวงจรนับและวงจรตั้งเวลา

2.1.7 การขัดจังหวะและชุดคำสั่ง

ไมโครคอนโทรลเลอร์ 8051 รับการขัดจังหวะได้ 5 แหล่ง ส่วนไมโครคอนโทรลเลอร์8052 รับการขัดจังหวะจากอุปกรณ์อื่นๆ ได้ถึง 6 แหล่ง คือ ขา INT0 และขา INT1 ซึ่งกำหนดให้ใช้ระดับพัลส์หรือขอบขาพัลส์ก็ได้ วงจรนับและวงจรตั้งเวลาที่ 0 และ 1 ซึ่งสำหรับการใช้งานไมโครคอนโทรลเลอร์ 8051 จะเพิ่มวงจรตั้งเวลาหรือ วงจรนับตัวที่ 2 และตัวสุดท้ายจากพอร์ตอนุกรมสามารถกำหนดลำดับความสำคัญของการขัดจังหวะได้ 2 ระดับไม่ต้องอาศัยวงจรภายนอกเข้ามาช่วยแต่ละแหล่งการขัดจังหวะ 5 หรือ 6 แหล่งนั้น สามารถกำหนดให้เป็นเวคเตอร์เฉพาะ ตัวชี้ตำแหน่ง ดังนั้นเมื่อมีขัดจังหวะเข้ามาตัวไมโครโปรเซสเซอร์จะกระโดดไปที่ส่วนของโปรแกรมที่ทำงานตามวัตถุประสงค์ของการขัดจังหวะนั้น และหลังจากเก็บข้อมูลต่างของโปรแกรมการนับลงในสแตก

2.1.8 AT89C51/52

AT89C51/52 พลิกรูปแบบการใช้งานไมโครคอนโทรลเลอร์ขนาด 8 บิต (Microcontroller Unit:MCU) ที่สามารถใช้เครื่องคอมพิวเตอร์ PC ทำการลบและทำการเขียนโปรแกรมได้โดยตรง ไม่ต้องถอด MCU ออกจากการ์ดหรือแผงวงจรในลักษณะที่เรียกว่า in system programming หรือจะใช้เครื่องโปรแกรม (Universal programmer) โดยตรงก็ได้โดยไม่ต้องใช้การลบด้วยแสงอัตราไวโอเลต (UV eraser) ทำให้ใช้เวลาลบเพียงเสี้ยววินาที และนั่นคือ AT89C51/52 ประดิษฐ์กรรมอุปกรณ์ ไมโครคอนโทรลเลอร์ที่มีหน่วยความจำแบบแฟลชอยู่ภายในตัวขนาดตั้งแต่ 1 กิโลไบต์ถึง 8 กิโลไบต์ ที่สามารถโปรแกรมทับลงไปได้อีกนับพันครั้งเหมือนกับการโปรแกรมในลักษณะเดียวกันกับ อีพรอม(EPROM)

ไมโครคอนโทรลเลอร์ขนาด 8 บิต (MCU) โดยทั่วไปมักมีหน่วยความจำภายในตัวเป็นแบบแมสค์รอม(MASK ROM) , วันไทม์โปรแกรมมิ่ง(one time programming: OTP) หรือ EPROM ซึ่งค่อนข้างยุ่งยากในการพัฒนาผลิตภัณฑ์ และพัฒนาโปรแกรมคำสั่งต่างๆ ที่ต้องการโดยเฉพาะในแบบ วันไทม์โปรแกรมมิ่ง(OTP) ซึ่งเป็นแบบโปรแกรมได้เพียงครั้งเดียวถ้ามีการเขียนโปรแกรมผิดพลาดเพียงนิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เดียวก็ต้องทิ้ง ไมโครคอนโทรลเลอร์ขนาด 8 บิต (MCU) ตัวนั้นไปเลย เป็นเพราะเป็นแบบโปรแกรมได้เพียงครั้งเดียว ถ้าเป็นแบบแมสค์รอม (MASK ROM) ถึงแม้ว่าราคาถูกกว่า แต่มีความยุ่งยากมากกว่า เนื่องจากผู้ผลิต ไมโครคอนโทรลเลอร์ขนาด 8 บิต (MCU) จะต้องเป็นผู้โปรแกรมให้ เป็นเหตุทำให้ต้องสั่งผลิตจำนวนคราวละมากๆ และค่อนข้างจะเสียเวลา ทำให้การวางแผนที่จะส่งผลิตภัณฑ์สู่ตลาดต้องใช้เวลา ส่วนแบบ อีพรอม(EPROM) นั้นมีราคาแพงและต้องใช้วิธีการลบด้วยแสงอัลตราไวโอเลตด้วยเครื่องยูวีอิเรเซอร์ (UV eraser) ในกรณีที่จะต้องแก้ไขหรือโปรแกรมข้อมูลใหม่และจำนวนครั้งในการลบก็จำกัด ดังนั้นการเลือกใช้ MCU ที่ผ่านมามักเกิดมีปัญหาในการเลือกเบอร์หรือเลือกชนิดที่ไว้ใจได้มาตลอด

ไมโครคอนโทรลเลอร์ตระกูล AT89C51 สามารถใช้งานร่วมและแทนกันได้กับไมโครคอนโทรลเลอร์ ในตระกูล MCS –51 ได้แก่เบอร์ 80C31 , 80C51 , 80C52 หรือ 87C51/52 เป็นต้น ซึ่ง AT89C51 เหมือนตระกูลเหล่านี้ของ INTEL ทั้งในด้านชุดคำสั่งและการจัดเรียงขา นั่นคือเราสามารถนำ AT89C51 ของ ATMEL มาใช้งานแทนไมโครคอนโทรลเลอร์ของ INTEL ได้เลย

2.2 ทฤษฎีการขับโมดูลแสดงผลแบบผลึกเหลว

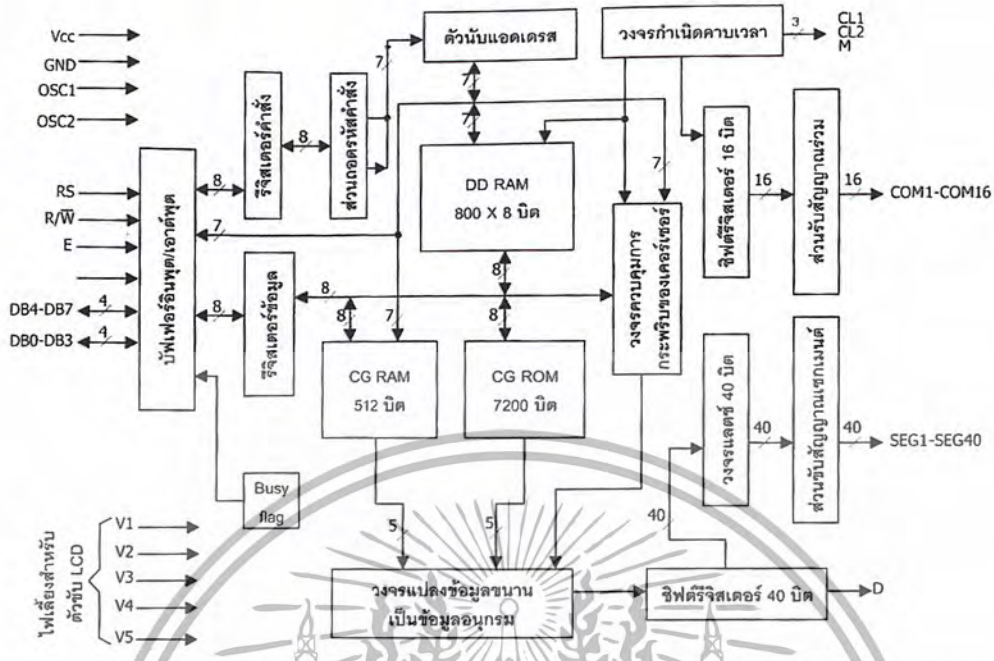
ตัวแสดงผล (display) ภายในเป็นผลึกเหลวสามารถแสดงผลให้เห็นโดยอาศัยแสงจากภายนอก ดังนั้นจึงต้องมีมุมในการมองข้อมูลที่แสดงผลบนจอแอลซีดี ตัวคอนโทรลเลอร์ (controller) เป็นตัวรับข้อมูลจากอุปกรณ์ภายนอกมาควบคุมการทำงานของโมดูลแอลซีดี เช่น ลบจอภาพ แสดงตัวอักษร หรือเลื่อนที่เคอร์เซอร์ เป็นต้น ตัวควบคุมนี้ใช้ชิปควบคุมโดยเฉพาะที่นิยม คือเบอร์ HD44780 จะใช้ควบคุมแอลซีดีแบบอักขระ

ตัวขับ (driver) เป็นตัวรับสัญญาณจากตัวควบคุมมาขับให้ตัวแสดงผลข้อมูลตามที่กำหนดชิปที่ใช้ทำหน้าที่นี้ได้แก่เบอร์ HD44100H และ MSM5259 เป็นต้น

2.3 โครงสร้างภายในของตัวควบคุมโมดูลแอลซีดี

ในการใช้งานโมดูล(module)แอลซีดีจำเป็นต้องทำความเข้าใจเกี่ยวกับโครงสร้างและคำสั่งที่ใช้ในการควบคุมให้ดีเสียก่อน ในที่นี้ขอยกตัวอย่างโมดูลแอลซีดี แบบอักขระเพราะสามารถเข้าใจได้ง่ายและจะเป็นบล็อกไดอะแกรมภายในชิปควบคุมแอลซีดีเบอร์ HD44780 ซึ่งใช้ในโมดูลแอลซีดี แบบอักขระประกอบด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 ไดอะแกรมการทำงานของโมดูลแอลซีดี แบบอักขระ

บัฟเฟอร์อินเอาท์พุท เป็นส่วนที่ใช้ในการติดต่อรับส่งข้อมูลกับอุปกรณ์ภายนอกเพื่อที่ถ่ายทอดข้อมูลเข้าออกภายในตัวควบคุม

รีจีสเตอร์คำสั่ง เป็นรีจีสเตอร์ใช้รับข้อมูลคำสั่งจากอุปกรณ์ภายนอก เพื่อนำไปควบคุมแสดงผล

รีจีสเตอร์ข้อมูล เป็นรีจีสเตอร์รับข้อมูลจากอุปกรณ์ภายนอกเพื่อถ่ายทอดต่อไปยังหน่วยความจำที่ทำหน้าที่เก็บข้อมูลแสดงผล หรือนำข้อมูลไปสร้างตัวอักษรเพิ่มเติมในแรมเก็บตัวอักษร

แรมเก็บข้อมูลแสดงผล เป็นหน่วยความจำแรมที่ทำหน้าที่เก็บข้อมูลที่มาจากรีจีสเตอร์ DR ตัวควบคุมนี้จะนำข้อมูลใน ดีดีแรม นี้ไปเปิดตาราง (Look up-lable) ของตัวอักษรที่เก็บไว้ในหน่วยความจำรอมและแรมเก็บตัวอักษร เพื่อนำไปแสดงผลที่ตัวแสดงผล

รอมเก็บตัวอักษร เป็นหน่วยความจำรอมที่ใช้สำหรับเก็บข้อมูลตัวอักษรหรือสัญลักษณ์ที่อ่านออกไปแสดงที่ตัวแสดงผลได้ มีขนาด 7200 บิต โดยจะถูกอ่านด้วยค่าของข้อมูลใน ดีดีแรม

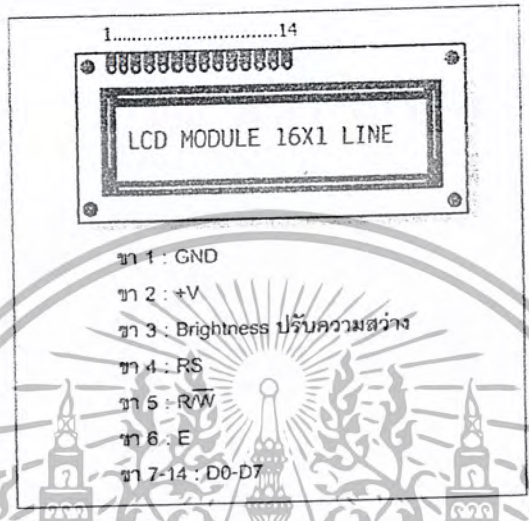
แรมเก็บตัวอักษร เป็นความจำแรมที่ใช้เก็บอักษรที่มีการสร้างเพิ่มเติมขึ้นใหม่ ในกรณีที่ตัวอักษรใน CGROM ไม่เพียงพอ มีขนาด 512 บิต การเขียนการอ่านค่าไปใช้นั้นทำได้เช่นเดียวกับ CGROM คือ เขียนข้อมูลลงใน ดีดีแรม แล้วตัวควบคุมจะมาอ่านค่าจาก CGROM เอง

แฟลก บีซี่(BUSY) เป็นส่วนที่ทำหน้าที่แจ้งสถานะการทำงานของตัวควบคุมให้อุปกรณ์ภายนอกทราบว่า ตัวควบคุมพร้อมที่จะรับข้อมูลหรือคำสั่งหรือไม่ ดังนั้น ก่อนการส่งข้อมูลหรือคำสั่งมายังตัวควบคุมต้องตรวจสอบสถานะของแฟลกบีซี่(BUSY) นี้เสียก่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โมดูลแอลซีดี ขนาด 16 ตัวอักษร 1 บรรทัด (LCD*1)

สำหรับโมดูลแอลซีดีที่ยกมาในการเรียนรู้ในการทดลอง เป็นขนาด 16 ตัวอักษร 1 บรรทัด เนื่องจากราคาถูก หาง่ายและเป็นโมดูลแอลซีดีที่มีโครงสร้างเป็นมาตรฐานมีผู้ผลิตหลายรายและมีการระบุเบอร์แตกต่างกันไปตามผู้ผลิต อาทิ LM020L ของฮิตาชิ ของครอปเท็กซ์ แต่อย่างไรก็ตามคอนโทรลเลอร์ที่ใช้คือ HD44750 ของฮิตาชิโมดูลแอลซีดี ขนาด 16*1 มีขาต่อใช้งานทั้งสิ้น 14 ขา มีรายละเอียดการทำงานของแต่ละขาดังนี้



รูปที่ 2.5 จอแสดงผล แอลซีดี

Vss 1/ ต่อกราวด์

Vdd 2/ ต่อไฟเลี้ยง+5V

Vo 3/เป็นขาอินพุทรับแรงดันเพื่อปรับความเข้มของการแสดงผล

RS 4/เป็นขาอินพุทใช้ในการแยกชนิดของข้อมูลที่ทำการประมวลผลในขณะนั้นว่าเป็นคำสั่งสำหรับ รีจีสเตอร์ IR หรือเป็นข้อมูลสำหรับรีจีสเตอร์ DR โดยถ้าขานี้เป็น 0 ข้อมูลที่ส่งมาจะเป็นคำสั่ง แต่ถ้าขานี้เป็น 1 ข้อมูลที่ส่งมาจะเป็นข้อมูลสำหรับการแสดงผล

R/W 5/เป็นขาที่ใช้เลือกการอ่านหรือเขียนข้อมูลกับแอลซีดี ถ้าเป็น 0 เป็นการกำหนดให้เขียนข้อมูล แต่ถ้าเป็น 1 จะเป็นการอ่านข้อมูล

E 6/เป็นขาอีนาเบิลแอลซีดีให้ทำงาน

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

ตารางที่ 2.2 แสดงความสัมพันธ์ในการทำงานของขา RS,R/W และ E ของโมดูลแอลซีดีแบบอักขระ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 คำสั่งควบคุมโมดูลแอลซีดี

ในการเขียนคำสั่งลงในตัวควบคุม แน่นอนว่าต้องกำหนดให้ขา RS และ R/W เป็น 0 แล้วเขียนคำสั่งตามไป คำสั่งควบคุมโมดูลแอลซีดี ของชิปควบคุม HD44780 ที่สำคัญ มี 10 คำสั่ง

2.4.1 คำสั่งเคลียร์ตัวแสดงผล

มีข้อมูลคำสั่งเป็น 01H เป็นคำสั่งที่ใช้เขียนข้อมูลช่องว่าง หรือ SPACE เข้าไปใน DDRAM ทั้งหมด เมื่อควบคุมเอ็กซีคิวคำสั่งนี้ จะทำการกำหนดแอดเดรสของ DDRAM เป็น 0 เคอร์เซอร์จะกลับไปอยู่ที่ตำแหน่งซ้ายมือสุดของจอแสดงผล แล้วเซตบิต I/D ให้เป็น 1

2.4.2 คำสั่ง return home

ต้องกำหนดให้บิต 1 ของข้อมูลเป็น 1 เป็นคำสั่งให้เคอร์เซอร์เคลื่อนที่กลับไปยังตำแหน่งซ้ายสุดของจอแสดงผล แต่ข้อมูลบนจอแสดงผลไม่เปลี่ยนแปลงนั่นคือ ข้อมูลคำสั่งของคำสั่งนี้จะเป็น 02H หรือ 03H ก็ได้

2.4.3 คำสั่งเลือกโหมดการป้อนข้อมูล (Entry mode Set)

มีรายละเอียดของรูปแบบข้อมูลคำสั่งดังนี้

บิต7	บิต6	บิต5	บิต4	บิต3	บิต2	บิต1	บิต0
0	0	0	0	0	1	I/D	S

บิต S เป็นบิตที่ใช้ในการกำหนดลักษณะของการแสดงผล เมื่อมีการป้อนข้อมูลถ้าหากบิต S เป็น 1 เมื่อเกิดข้อมูลใหม่บนจอแสดงผล ตัวเคอร์เซอร์จะอยู่กับที่ แต่ตัวอักษรข้อมูลเดิมจะถูกดันไปทางซ้ายแต่ถ้าหากบิตนี้เป็น 0 เมื่อเกิดข้อมูลใหม่ตัวเคอร์เซอร์จะเลื่อนไปทางขวามือ

บิต I/D เป็นบิตที่ใช้ในการกำหนดว่า เมื่อเขียนหรืออ่านข้อมูลแล้ว ทำให้แอดเดรสของ DDRAM เพิ่มขึ้นหรือลดลง 1 แอดเดรสของ DDRAM จะเพิ่มขึ้น แต่ถ้าเป็น 0 แอดเดรสจะลดลง

ดังนั้นข้อมูลคำสั่งที่เกิดขึ้นสำหรับคำสั่งนี้ได้แก่ 04H-07H และที่ใช้บ่อยคือ 06H หมายถึงกำหนดให้เมื่อเกิดข้อมูลใหม่ เคอร์เซอร์จะเลื่อนไปทางขวามือและแอดเดรสของ DDRAM เพิ่มขึ้น

บิต7	บิต6	บิต5	บิต4	บิต3	บิต2	บิต1	บิต0
0	0	0	0	1	D	C	B

บิต D ใช้ควบคุมการเปิดปิดจอ ถ้าบิตนี้เป็น 1 จะเปิดจอแสดงผล ถ้าเป็น 0 จะเป็นการปิดจอแสดงผล

บิต C ใช้ควบคุมการแสดงตัวเคอร์เซอร์บนจอแสดงผล ถ้าต้องการให้มีเคอร์เซอร์แสดงผลบนจอแสลงผลต้องกำหนดให้บิตนี้เป็น 1 ถ้ากำหนดให้เป็น 0 จะเป็นการปิดเคอร์เซอร์หรือไม่แสดงเคอร์เซอร์

บิต B ใช้ควบคุมการกระพริบของเคอร์เซอร์ ถ้าบิตนี้เป็น 1 เคอร์เซอร์นี้จะกระพริบ

ดังนั้นจะมีข้อมูลคำสั่งตั้งแต่ 08H-0FH ที่ใช้บ่อยคือ 0CH เป็นการสั่งให้เปิดจอแสดงผลแต่ไม่แสดงเคอร์เซอร์ และ 0FH เป็นการสั่งให้เปิดจอแสดงผล แสดงเคอร์เซอร์ และสั่งให้เคอร์เซอร์กระพริบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.4 คำสั่งควบคุมการเลื่อนเคอร์เซอร์และข้อมูลตัวอักษร

มีรายละเอียดของรูปแบบข้อมูลคำสั่งดังนี้

บิต7	บิต6	บิต5	บิต4	บิต3	บิต2	บิต1	บิต0
0	0	0	1	S/C	R/L	*	*

การควบคุมการเลื่อนเคอร์เซอร์และตัวอักษรบนจอแสดงผลขึ้นอยู่กับการกำหนดบิต S/C และ R/L ซึ่งสามารถสรุปได้ดังนี้

S/C	R/L	ลักษณะการเลื่อน	ข้อมูลคำสั่ง
0	0	เลื่อนเคอร์เซอร์ไปทางซ้าย	10H-13H
0	1	เลื่อนเคอร์เซอร์ไปทางขวา	14H-17H
1	0	เลื่อนตัวอักษรใหม่ไปทางซ้าย	18H-1BH
1	1	เลื่อนตัวอักษรใหม่ไปทางขวา	1CH-1FH

2.4.5 กำหนดฟังก์ชันการทำงาน

มีรายละเอียดของรูปแบบข้อมูลคำสั่งดังนี้

บิต7	บิต6	บิต5	บิต4	บิต3	บิต2	บิต1	บิต0
0	0	1	DL	N	F	*	*

บิต DL ใช้กำหนดจำนวนบิตที่ใช้ติดต่อส่งผ่านข้อมูล ถ้าบิตนี้เป็น 0 จะเป็นการติดต่อแบบ 4 บิต แต่ถ้าเป็น 1 จะเป็นแบบ 8 บิต

บิต N ใช้กำหนดจำนวนบรรทัดของการแสดงผลถ้าเป็น 0 จะแสดงผล 1 บรรทัด ถ้าเป็น 1 จะแสดงผล 2 บรรทัด ในกรณีที่จอแสดงผลสามารถแสดงผลได้มากกว่า 2 บรรทัด และต้องการแสดงให้มากกว่า 2 บรรทัดก็กำหนดบิต N ให้เป็น 1

บิต F ใช้เลือกความละเอียดของตัวอักษรให้แสดงผล ถ้าบิตนี้เป็น 0 จะเป็นการแสดงผลแบบ 5*7 จุด และถ้าเป็น 1 จะแสดงผลเป็นแบบ 5*10 จุด

ข้อมูลคำสั่งที่ใช้บ่อยคือ 38H เป็นการกำหนดให้โมดูลแอลซีดี ทำงานในแบบ 8 บิต แสดงผล 2 บรรทัด และเลือกความละเอียดเป็น 5*7 จุด

จุดที่น่าสังเกตคือ โมดูลแอลซีดีแบบ 16 ตัวอักษร 1 บรรทัด แม้จะมีบรรทัดการแสดงผลเพียง 1 บรรทัด แต่จะต้องกำหนด N ให้เป็น 1 เนื่องจากแอดเดรสของ DDRAM แบ่งเป็น 2 ช่องคือ 00H และ 40H

2.4.6 คำสั่งเลือกแอดเดรสของ CGRAM

เมื่อต้องการกำหนดแอดเดรสของ CGRAM ต้องกำหนดให้บิต 7 เป็น 0 บิต 6 เป็น 1 ส่วนอีก 6บิตที่เหลือจะแสดงด้วยค่าแอดเดรสของ CGRAM จะต้องทำการกำหนดแอดเดรสด้วยคำสั่งนี้ก่อนที่จะอ่านหรือเขียนข้อมูลให้ CGRAM โดยแอดเดรสของ CGRAM อยู่ระหว่าง 00-3H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.7 คำสั่งเลือดแอดเดรสของ DDRAM

ใช้ในการเลือกแอดเดรสของ DDRAM ก่อนที่จะทำการอ่านหรือเขียนข้อมูล โดยบิต 7 ต้องเป็น 1 และข้อมูลอีก 7 บิตที่เหลือจะเป็นค่าแอดเดรสของ DDRAM ซึ่งแอดเดรสของ DDRAM จะอยู่ระหว่าง 8CH-0FFH ทั้งนี้จำนวนแอดเดรสยังขึ้นกับการกำหนดสถานะที่บิต N ด้วย หากบิต N เป็น 0 แอดเดรสของ DDRAM จะอยู่ระหว่าง 80H-0CH และถ้าบิตN เป็น 1 แอดเดรสของ DDRAM จะมี 2 ช่วงคือ 8CH-87H และ 0C0H-0C7H

มีรายละเอียดของรูปแบบข้อมูลคำสั่งดังนี้

บิต7	บิต6	บิต5	บิต4	บิต3	บิต2	บิต1	บิต0
BF	A	A	A	A	A	A	A

เป็นคำสั่งที่ใช้อ่านแฟลก BUSY(BF)โดยแฟลกนี้จะบอกสถานะของตัวควบคุมแอลซีดีว่าพร้อมที่จะรับข้อมูลอยู่หรือไม่ ถ้าหากบิต BF เป็น 0 แสดงว่าตัวควบคุมแอลซีดี พร้อมรับข้อมูลหรือคำสั่ง แต่ถ้าเป็น 1 แสดงว่า ขณะนี้ ตัวควบคุมแอลซีดี ยังอยู่ในกระบวนการทำงานภายในหรือกำลังประมวลผลข้อมูลอยู่ ยังไม่พร้อมรับข้อมูลหรือคำสั่ง

เมื่อต้องการอ่านแฟลกต้องกำหนดให้ขาR/W เป็น 1 ด้วย แต่สัญญาณที่RS ยังต้องเป็น 0 อยู่เพราะข้อมูลนั้นเป็นข้อมูลคำสั่ง

นอกจากนี้ ยังใช้เป็นคำสั่งอ่านข้อมูลแอดเดรสของ CGRAM และ DDRAM ด้วย โดยบิต0-บิต6 เป็นค่าข้อมูลของแอดเดรสที่ต้องการอ่าน

2.4.8 การเขียนคำสั่งและข้อมูลให้แก่โมดูลแอลซีดี

ในการเขียนข้อมูลเพื่อควบคุมให้โมดูลแอลซีดีแสดงผลตามที่ผู้ใช้งานต้องการนั้นจะต้องส่งคำสั่ง (instruction) แล้วกำหนดโหมดการทำงานให้แก่โมดูลของแอลซีดีก่อนหลังจากนั้นจึงค่อยส่งข้อมูล (data) ที่ต้องการแสดงผลเนื่องจากบัสข้อมูลของโมดูลแอลซีดี มี 8 เส้น คือ D0-D7 และใช้เป็นทางผ่านของทั้งคำสั่งและข้อมูล ดังนั้นในการส่งคำสั่งและข้อมูลจึงต้องอาศัยการกำหนดสัญญาณลอจิกที่ขา RS ถ้าหากที่ขา RS ได้ลอจิก 0 หมายความว่าข้อมูลที่ป้อนให้กับ โมดูลแอลซีดี ขณะนี้เป็นคำสั่งในทางตรงข้ามหากขา RS ได้รับลอจิก 1 ข้อมูลที่ป้อนให้ขณะนั้นเป็นข้อมูลที่ใช้ในการแสดงผล

เมื่อต้องการอ่านหรือเขียนข้อมูลใน CGRAM และ DDRAM เริ่มต้นต้องกำหนดแอดเดรสที่ต้องการอ่านหรือเขียนก่อน โดยใช้คำสั่งเลือกแอดเดรส จากนั้นกำหนดให้ขา RS เป็น 1 เพื่อแจ้งให้ตัวควบคุมภายในโมดูลแอลซีดี ทราบว่าข้อมูลที่ปรากฏต่อไปนี้เป็นข้อมูลปกติไม่ใช่คำสั่ง

ในกรณีที่ต้องการอ่านหรือเขียนข้อมูลต้องกำหนดให้ขา R/W เป็น 1 ข้อมูล ขนาด 8 บิต หรือ 4 บิต ก็จะปรากฏบนบัสข้อมูล โดยข้อมูลที่อ่านออกมาได้จะเป็นข้อมูลจากแอดเดรสของ CGRAM หรือ DDRAM ตามที่ต้องการ

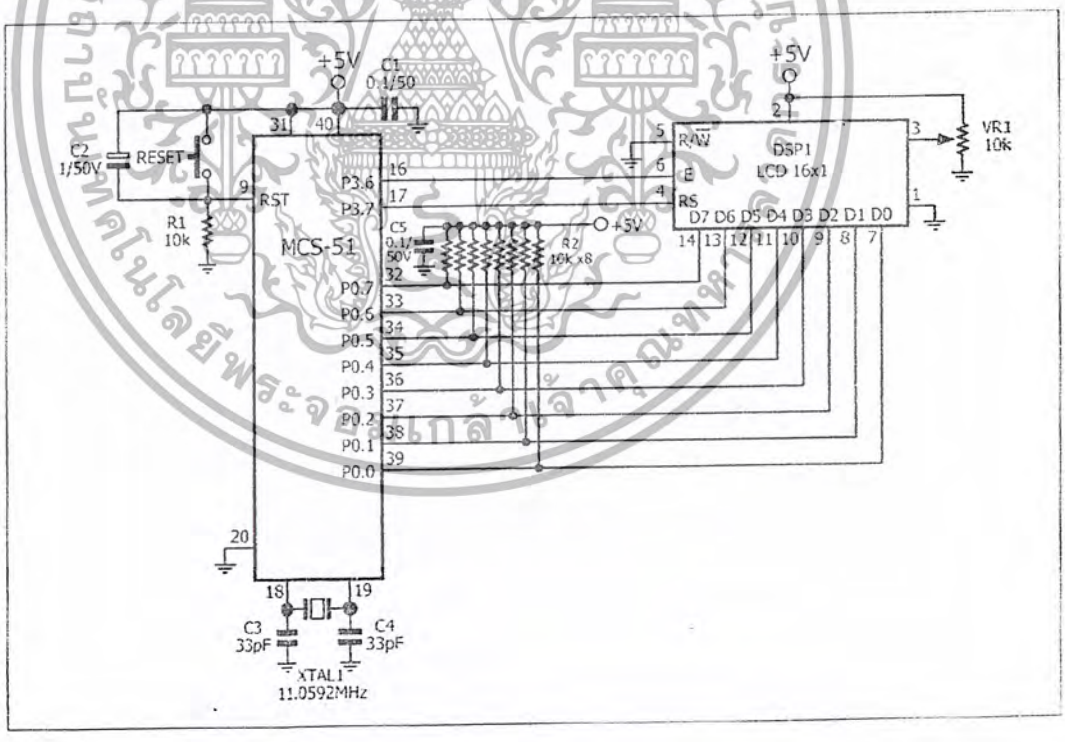
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.9 จังหวะการทำงานของแอลซีดีโมดูล

ในการติดต่อกับโมดูลแอลซีดี จะต้องมีการหน่วงเวลาหลังจากที่ทำการส่งรหัสคำสั่ง หรือข้อมูล เนื่องจากต้องรอให้คอนโทรลเลอร์ภายในแอลซีดีโมดูลแปลความหมายรหัสคำสั่งและทำงานตามคำสั่งให้เรียบร้อยเสียก่อน จากนั้นจึงจะรับข้อมูลหรือดำเนินการต่อไป

ดังนั้นในการใช้งานโมดูลแอลซีดีผู้เขียนโปรแกรมต้องมีโปรแกรมเพื่อหน่วงเวลารอให้โมดูลแอลซีดีพร้อมทำงานด้วย โดยเมื่อเริ่มจ่ายไฟให้แก่โมดูลแอลซีดี ต้องรอประมาณ 10 มิลลิวินาที เพื่อให้โมดูลแอลซีดี ทำการเตรียมความพร้อมหรืออินนิเชียล หลังจากนั้นก็กำหนดลอจิกลอจิกให้แก่ขา RS ของโมดูลแอลซีดี แล้วต้องหน่วงเวลาอีกประมาณ 2 มิลลิวินาที เพื่อให้คอนโทรลเลอร์ในแอลซีดีโมดูล แปลความหมายของขาโลจิกที่ขา RSว่าข้อมูลต่อไปที่จะได้รับนั้นเป็นรหัสคำสั่งหรือเป็นข้อมูลที่ต้องการจะไปแสดงผล จากนั้นจะเป็นการส่งข้อมูลมารอที่บัสข้อมูล D0-D7 ขั้นตอนต่อไปนี้จะเป็นการส่งสัญญาณพัลส์ไปที่ขา E เพื่ออีนาเบิลโมดูลแอลซีดี ให้รับข้อมูลจากบัสข้อมูลเข้าไป โดยพัลส์ที่ป้อนเข้าที่ขา E ของโมดูลแอลซีดี ต้องเป็นพัลส์ขอบขาขึ้น จากนั้นทำการหน่วงเวลา 2 มิลลิวินาที

ทั้งหมดที่กล่าวมาคือขั้นตอนและจังหวะการทำงาน 1 รอบของโมดูลแอลซีดี จะเห็นได้ว่ามีโปรแกรมย่อยที่สำคัญอยู่ 3 โปรแกรมย่อยคือ โปรแกรม อินนิเชียลแอลซีดี โปรแกรมหน่วงเวลาและโปรแกรมย่อยการส่งพัลส์เพื่ออีนาเบิลโมดูลแอลซีดี



รูปที่2.6 วงจรสำหรับทดลองการแสดงข้อความบนโมดูลแอลซีดี 16 ตัวอักษร 1 บรรทัด

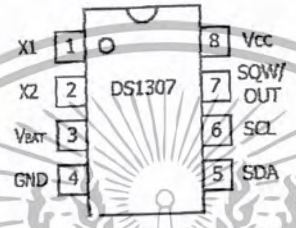
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 ทฤษฎี DS1307 ไอซีสร้างฐานเวลาหรือเรียลไทม์คล็อก (real time clock)

ผู้ผลิตคือ เดลลัสเซมิคอนดักเตอร์ ที่สร้างฐานเวลาจริงให้แก่ระบบไมโครคอนโทรลเลอร์โดย DS1307 จะให้ข้อมูลเกี่ยวกับเวลาทั้งหมด ไม่ว่าจะเป็นค่าของเวลาที่ละเอียดถึงหลักวินาที นาที ชั่วโมง วัน ที่ วันในสัปดาห์ เดือน ปี โดยสามารถปรับเดือนปีให้ตรงตามปฏิทินได้อย่างถูกต้อง

คุณสมบัติทางเทคนิคมีดังนี้

- เป็นไอซีเรียลไทม์คล็อกให้ข้อมูลตั้งแต่วินาทีจนถึงปี
- มีหน่วยความจำนอนโวทาแรม 56 ไบต์อยู่ภายในสามารถเก็บข้อมูลทั่วไป
- มีวงจรตรวจจับไฟเลี้ยงต่ำหรือหายไปอย่างอัตโนมัติ สามารถรักษาข้อมูลไว้ได้แม้ไม่มีไฟเลี้ยง



รูปที่ 2.7 การจัดขาของ IC DS1307

รายละเอียดขาต่อต่อการใช้งานของ DS1307

Vcc.GND ขา 8, 4 ต่อไฟเลี้ยง 5V

VBAT ขา 3 ใช้ต่อกับแบตเตอรี่ 3 V เพื่อรักษาการทำงานของวงจรสร้างฐานเวลา DS1307 ให้คงอยู่ต่อไป แม้ว่าจะไม่มีไฟเลี้ยงให้แก่ DS1307 ชนิดของแบตเตอรี่ที่เหมาะสมคือแบตเตอรี่แบบลิเทียม ซึ่งมีความจุ 40 mAhr หรือมากกว่า จะสามารถรักษาข้อมูลได้นาน 10 ปีที่อุณหภูมิ 25 องศาเซลเซียส

SDA , SCL ขา 5, 6 เป็นขาสำหรับเชื่อมต่อกับไมโครคอนโทรลเลอร์บนระบบบัส I^2C

SQW/OUT ขา 7 ที่ขานี้จะมีสัญญาณรูปสี่เหลี่ยมส่งออกมา โดยสามารถเลือกความถี่ได้ 1 Hz, 4.096 kHz, 8.192 kHz, และ 32 kHz ในการใช้งานต้องต่อตัวต้านทาน 1 k พูลอัพที่ขานี้ด้วย

X1, X2 ขา1, 2 ใช้ต่อกับคริสตอลความถี่มาตรฐาน 32.768 kHz เพื่อใช้เป็นฐานเวลา เพื่อใช้ในการสร้างเวลาจริงในการใช้งานต้องต่อคริสตอลเข้ากับขาทั้งสองนี้และที่แต่ละขาต้องต่อตัวเก็บประจุค่าต่ำๆประมาณ 15 pF คร่อมกับขากราวด์ด้วย

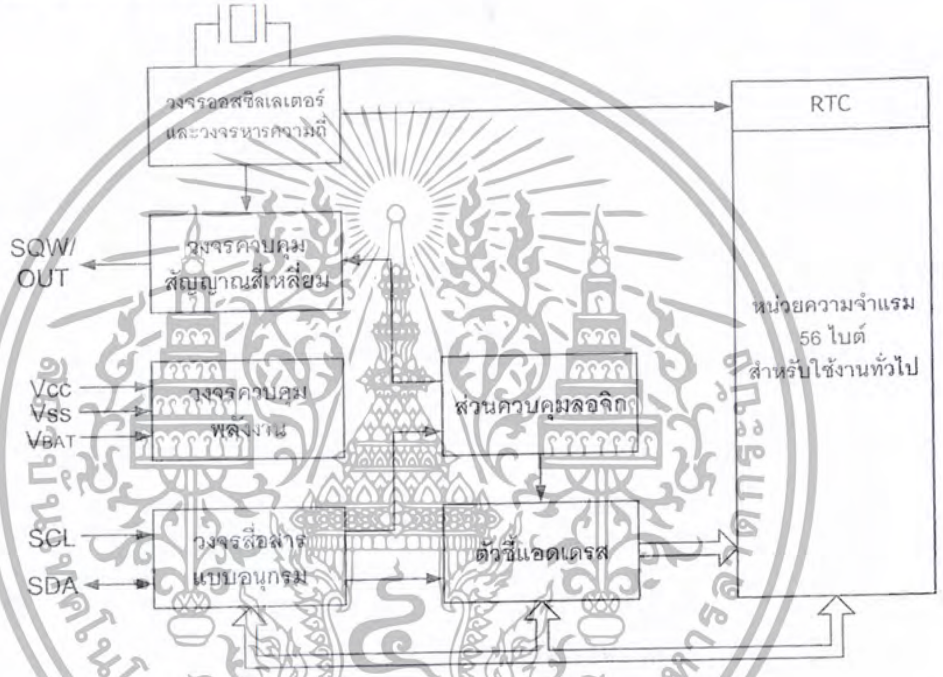
2.5.1 การทำงานของ DS1307

ไอซี DS1307 จัดการเชื่อมต่อในแบบบัส I^2C โดยจะทำงานเป็นอุปกรณ์สเลฟอยู่เสมอ ดังนั้น การติดต่อเพื่อใช้งานจึงต้องกำหนดรูปแบบตามที่กำหนดไว้ในการติดต่อแบบ I^2C ในรูปแสดงส่วนประกอบที่สำคัญและไดอะแกรมการทำงานของ DS1307 วงจรออสซิลเลเตอร์ถือเป็นหัวใจหลักของไอซี เนื่องจากเป็นจุดเริ่มต้นของการสร้างข้อมูลเวลาจริง ในขณะที่ DS1307 ทำงานที่ขาSQW/OUT จะมีสัญลักษณ์พัลส์สี่เหลี่ยมส่งออกมาตลอดเวลาที่มีการอีนาเบิลวงจรกำเนิดสัญญาณพัลส์ที่รีจิสเตอร์ควบคุม ค่าความถี่ของสัญญาณนี้สามารถเลือกได้ พร้อมกันนั้นก็จะมีการเก็บค่าของเวลาไว้ในหน่วยความจำนอนโวลาไทแรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งมีขนาดรวม 64 ไบต์ แต่จัดสรรให้ใช้เก็บข้อมูลเวลา 8 ไบต์และเป็นหน่วยความจำสำหรับเก็บข้อมูลทั่วไปอีก 56 ไบต์

วงจรควบคุมพลังงานไฟฟ้าจะคอยตรวจสอบสภาวะของไฟเลี้ยงไอซี ถ้าหากไฟเลี้ยงนั้นต่ำกว่า 1.25 × VBAT ก็จะควบคุมให้ DS1307 หยุดการทำงานรีเซตค่าตัวนับแอดเดรสภายในทำให้ไม่สามารถติดต่อกับ DS1307 ได้ดังนั้นในการใช้การใช้งาน DS1307 ต้องระวังอย่าให้ไฟเลี้ยงต่ำกว่า1.25 × VBAT หรือประมาณ 3.75 โวลต์ ในกรณีที่ใช้ VBAT เท่ากับ 3 โวลต์ ถ้าหากไฟเลี้ยงมีค่าต่ำกว่า VBAT ไอซี DS1307จะเข้าสู่โหมดสำรองข้อมูลกระแสต่ำทันที จะไม่มีการส่งสัญญาณพัลส์ออกมาที่ขา SQW/OUT แต่วงจรสร้างฐานเวลายังคงทำงานเพื่อให้ค่าของเวลาเดินไปอย่างไม่ผิดพลาด เมื่อไฟเลี้ยงปรากฏขึ้นอีกครั้ง DS1307 ก็สามารถให้ค่าของเวลาที่เป็นจริงได้



รูปที่ 2.8 วงจรภายใน DS1307

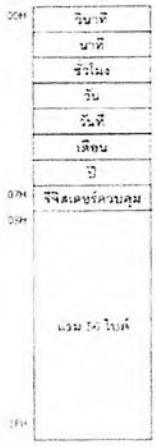
วงจรสื่อสารอนุกรมภายใน DS1307 ได้รับการกำหนดให้ทำงานตามรูปแบบของบัส I^2C เป็นช่องทางการสื่อสารระหว่าง DS1307 กับอุปกรณ์มาสเตอร์ ผู้ใช้งานสามารถเข้าถึงหน่วยความจำที่ใช้เก็บค่าเวลาและหน่วยความจำใช้งานทั่วไปได้โดยการเขียนข้อมูลตามรูปแบบที่กำหนดไว้ในระบบบัส I^2C

2.5.2 การจัดสรรหน่วยความจำใน DS1307

ในรูป 2.9ก แสดงการจัดสรรพื้นที่ของหน่วยความจำภายใน DS1307 พื้นที่ 7 ไบต์แรกตั้งแต่แอดเดรส 00H-06H เป็นพื้นที่ของรีจีสเตอร์ค่าเวลาใช้ในการเก็บข้อมูลเกี่ยวกับเวลา ไบต์ต่อมาที่แอดเดรส 07H เป็นพื้นของรีจีสเตอร์ควบคุมการทำงานของ DS1307 ในรูป ข แสดงรายละเอียดของ รีจีสเตอร์ค่าเวลาและรีจีสเตอร์ควบคุมของ DS1307

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



(ก)

	บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0	ค่าของข้อมูล
CH	ข้อมูลวินาที (หลักสิบ)				ข้อมูลวินาที (หลักหน่วย)				00-59
x	ข้อมูลนาที (หลักสิบ)				ข้อมูลนาที (หลักหน่วย)				00-59
x				ข้อมูลชั่วโมง (หลักสิบ)	ข้อมูลชั่วโมง (หลักหน่วย)				01-12 00-23
x	x	x	x		x	ข้อมูลวันในสัปดาห์			1-7
x	x	ข้อมูลวันที่ (หลักสิบ)			ข้อมูลวันที่ (หลักหน่วย)				01-28/29 01-30 01-31
x	x	x	ข้อมูลเดือน (หลักสิบ)		ข้อมูลเดือน (หลักหน่วย)				01-12
ข้อมูลปี (หลักสิบ)					ข้อมูลปี (หลักหน่วย)				00-99
OUT	x	x	SQWE		x	x	RS1	RS0	

(ข)

รูปที่ 2.9 แสดงการจัดสรรหน่วยความจำ

ด้วยการจัดสรรพื้นที่แบบนี้ ทำให้ผู้ใช้งานสามารถเรียกข้อมูลเวลาออกมาได้ตามที่ต้องการโดยไม่จำเป็นต้องอ่านออกมาทั้งหมดก็ได้ ค่าของเวลาทั้งหมด จะอยู่ในรูปของเลขฐาน 10 สำหรับการแสดงเวลาในรูปของชั่วโมงสามารถเลือกได้ว่าต้องการแบบ 12 ชั่วโมงหรือ 24 ชั่วโมง โดยกำหนดบิตที่ 6 ของแอดเดรส 02H และเมื่อเลือกแบบ 12 ชั่วโมงที่บิต 5 ในแอดเดรสเดียวกันจะใช้ในการแสดงค่า AM/PM โดยถ้าบิตนี้เป็น 1 หมายถึงค่าชั่วโมงในขณะนี้เป็นช่วงเวลาหลังเที่ยงวัน ในกรณีที่เป็นแบบ 24 ชั่วโมงบิตนี้จะใช้ในการแสดงค่า 2 ของหลักสิบในหน่วยชั่วโมง

2.5.3 รีจีสเตอร์ควบคุม

มีแอดเดรสอยู่ที่ 07H มีรายละเอียดดังนี้

OUT (Output control) ใช้ควบคุมระดับลอจิกที่ขา SQW/OUT ในกรณีที่ดิสเอเบิลการกำเนิดสัญญาณสี่เหลี่ยม โดยถ้าบิตนี้เป็น 1 ที่ขา SQW/OUT ก็จะเป็น 1 ถ้าบิตนี้เป็น 0 ที่ขา SQW/OUT ก็ จะเป็น 0 เช่นเดียวกัน

SQWE (Square Wave Enable) ใช้ในการอีนาเบิลวงจรกำเนิดสัญญาณสี่เหลี่ยมที่ขา SQW/OUT ถ้าต้องการให้มีสัญญาณสี่เหลี่ยมออกมาต้องกำหนดให้บิตนี้เป็น 1

RS1, RS0 ใช้ในการเลือกความถี่ของสัญญาณสี่เหลี่ยมที่ออกจากขา SQW/OUT ดังมีรายละเอียดดังนี้

RS1	RS0	ค่าความถี่สัญญาณ
0	0	1Hz
0	1	4.096kHz
1	0	80192kHz
1	1	32.768kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

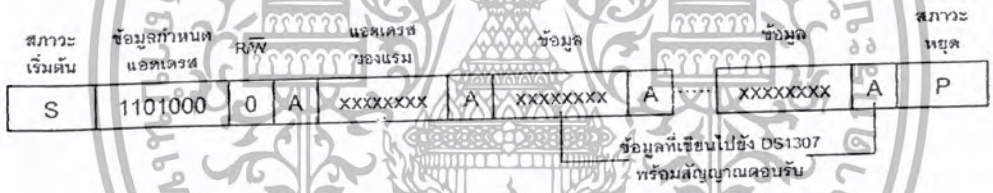
54991

2.5.4 โหมดการทำงานของ DS1307

มีด้วยกัน 2 โหมด คือโหมดเขียนข้อมูลและโหมดอ่านข้อมูล ในการใช้งาน DS1307 ตามปกติจะใช้งานเฉพาะโหมดอ่านข้อมูลเท่านั้น เนื่องจากไมโครคอนโทรลเลอร์จะติดต่อกับ DS1307 เพื่ออ่านข้อมูลของเวลาไปใช้งาน ในโหมดการเขียนข้อมูลจะถูกใช้งานก็ต่อเมื่อต้องการตั้งค่าเวลาใหม่และต้องการเขียนข้อมูลลงในหน่วยความจำใช้งานทั่วไป อย่างไรก็ตามเมื่อเริ่มต้นติดต่อกับ DS1307 จำเป็นอย่างยิ่งต้องเข้าสู่โหมดการเขียนข้อมูลก่อนเพื่อกำหนดแอดเดรสที่ต้องการอ่านข้อมูลจากนั้นจึงเปลี่ยนโหมดการทำงานมาเป็นโหมดการอ่านข้อมูลต่อไป

2.5.5 โหมดการเขียนข้อมูล

เริ่มต้นเมื่อไมโครคอนโทรลเลอร์ทำการกำหนด สภาวะเริ่มต้น จากนั้นส่งข้อมูลกำหนดแอดเดรส 1101000 ตามด้วยข้อมูลเลือกการเขียน นั่นคือค่า 0 จากนั้นจะรอการตอบรับจาก DS1307 ขั้นตอนต่อคือส่งข้อมูลเพื่อเลือกแอดเดรสที่ต้องการเขียนจากนั้นรอการตอบรับจาก DS1307 เมื่อมีการตอบรับมาเรียบร้อย ก็เริ่มทยอยเขียนข้อมูลลงไปครั้งละแอดเดรสหลังจากเขียนข้อมูลในแต่ละแอดเดรส จะต้องรอการตอบรับจาก DS1307 ทุกครั้งจึงสามารถเขียนข้อมูลต่อไปได้ เมื่อเขียนเรียบร้อยแล้วให้ส่งสภาวะหยุดเป็นอันสิ้นสุดกระบวนการเขียนข้อมูล

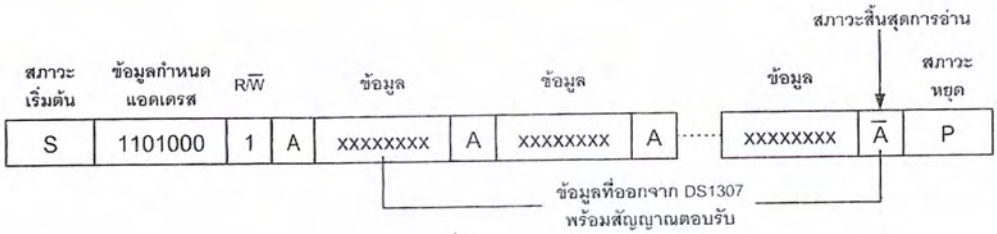


รูปที่2.10 รูปแบบข้อมูลสำหรับติดต่อกับ DS1307 ในโหมดการเขียนข้อมูล

2.5.6 โหมดการอ่านข้อมูล

เริ่มต้นทำงานเหมือนกับโหมดเขียนข้อมูลคือ ไมโครคอนโทรลเลอร์กำหนดสภาวะเริ่มต้นแล้วส่งข้อมูลกำหนดแอดเดรสตามด้วยข้อมูลเลือกการอ่านซึ่งเท่ากับ 1 จากนั้นรอการตอบรับจาก DS1307 เมื่อตอนเรียบร้อยแล้ว DS1307 จะทยอยส่งข้อมูลออกมาให้คอนโทรลเลอร์คราวละ 1 แอดเดรส หรือ 1 ไบต์ โดยแอดเดรสที่เลือกอ่านข้อมูลจะต้องมีการกำหนดมาก่อนล่วงหน้าด้วย โหมดการเขียนข้อมูล วิธีการง่ายๆคือเข้าโหมดการเขียนข้อมูลก่อน เมื่อถึงจังหวะที่ต้องเขียนข้อมูล ให้ทำการสร้างสภาวะเริ่มต้นและส่งข้อมูลแอดเดรสใหม่อีกครั้ง ตามด้วยเลือกโหมดการอ่านข้อมูล ข้อมูลที่ออกมาจาก DS1307 ก็จะเป็นข้อมูลจากแอดเดรสที่กำหนดไว้ก่อนหน้านี้

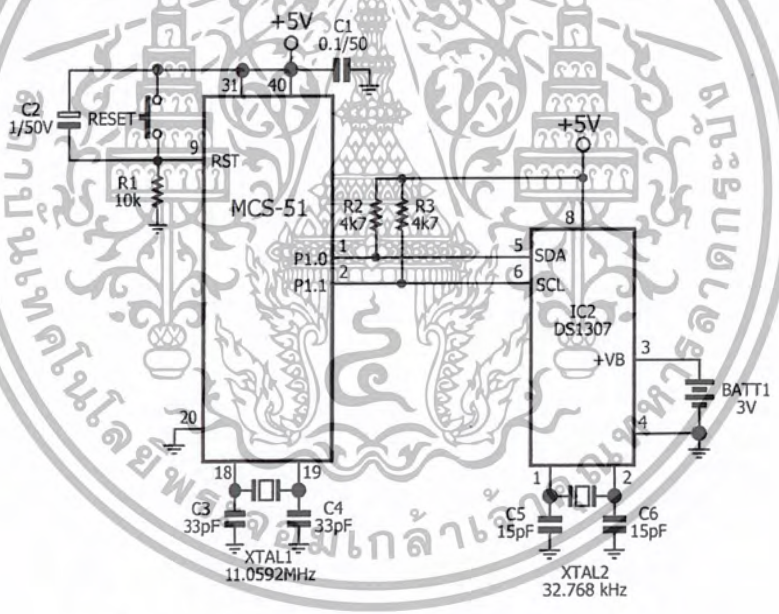
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่2.11 รูปแบบข้อมูลสำหรับติดต่อกับ DS1307 ในโหมดการอ่านข้อมูล

2.5.7 การเชื่อมต่อ DS1307 กับไมโครคอนโทรลเลอร์ MCS-51

จากรูป2.12 จะเห็นได้ว่ามีลักษณะการต่อเหมือนกับอุปกรณ์ระบบบัส I^2C ตัวอื่นๆทุกประการ และสามารถที่จะต่อไอซีทั้งหมดร่วมกันบนสาย SDA และ SCL ได้ ซึ่งเป็นการย้ำให้เห็นถึงความสามารถพิเศษของระบบบัส I^2C ที่ผู้ใช้งานสามารถเชื่อมต่ออุปกรณ์ที่มีความต่างกันในหน้าที่การทำงานบนสายสัญญาณเดียวกันได้ ถึงการทดลองนี้ผู้ใช้งานสามารถเชื่อมต่ออุปกรณ์ระบบบัสได้คือ I^2C ได้ถึง 3 ตัว 3 ลักษณะการทำงาน โดยใช้สายเพียง 2 เส้น

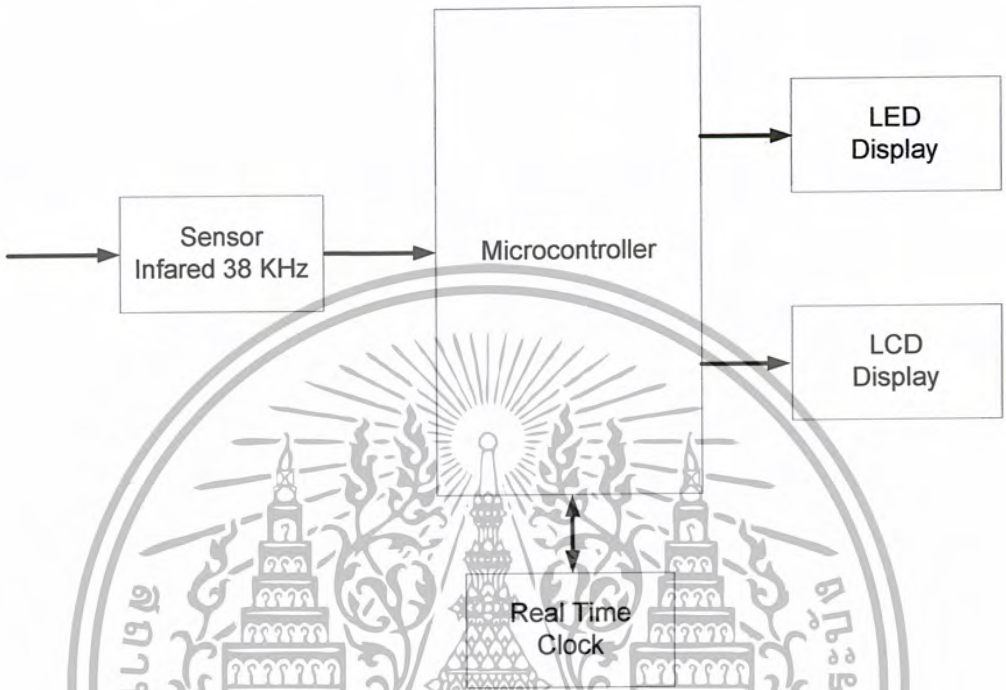


รูปที่ 2.12 แสดงการเชื่อมต่อไมโครคอนโทรลเลอร์ MCS-51
กับไอซีเรียลไทม์คล็อก DS1307

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การคำนวณและการสร้าง

โครงงานนี้เป็นการนำไมโครคอนโทรลเลอร์มาประยุกต์ใช้ในการทำงานของระบบที่จอดรถอัตโนมัติ โดยที่มีขบวนการดังบล็อกไดอะแกรมข้างล่างนี้



รูปที่ 3.1 แสดงบล็อกไดอะแกรมของโครงงาน

3.1 โครงสร้างทางฮาร์ดแวร์(Hardware)

3.1.1 ส่วนของวงจรเซนเซอร์

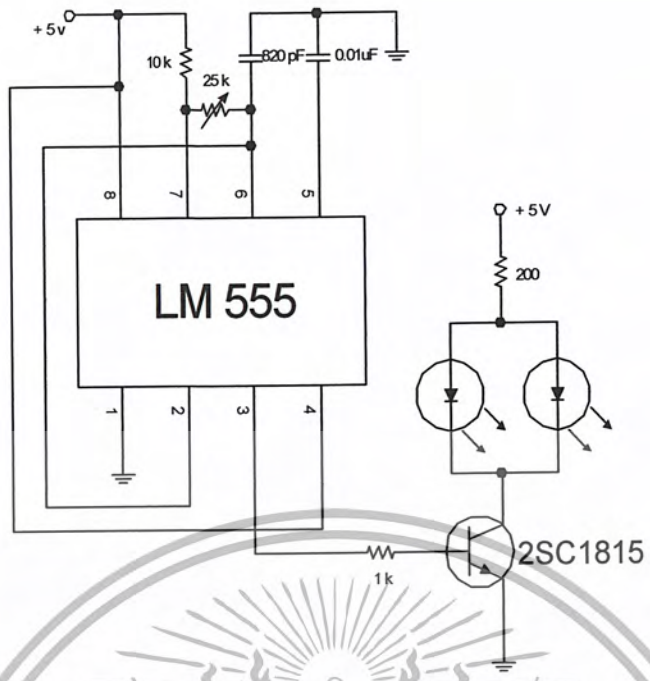
จากลักษณะการทำงานของวงจรตัวส่งสัญญาณอินฟราเรด จะเห็นได้ว่าช่วงความกว้างพัลส์ที่ได้ทางเอาท์พุทจะขึ้นอยู่กับการเก็บประจุ และการคายประจุของ Ca ซึ่งจากดาต้าชีท(Datasheet) จะได้ว่า

ช่วงการเก็บประจุ : t1 = 0.693*(Ra+Rb)*Ca

ช่วงการคายประจุ : t2 = 0.693*Ra*Ca

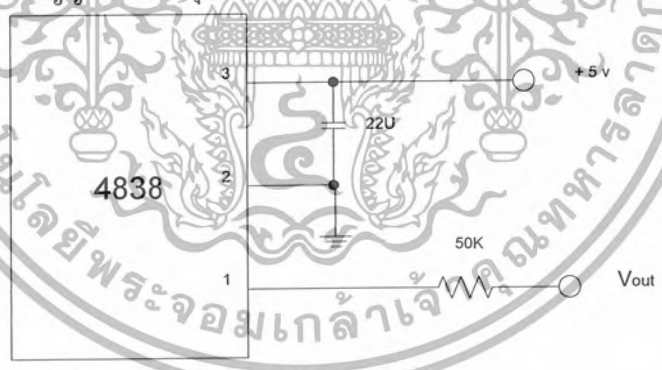
ความถี่ของสัญญาณเอาท์พุท : f = 1/(t1+t2)วงจรอินฟราเรดทางด้านส่งจะสร้างพัลส์โดยใช้ IC555 ซึ่งสร้างความถี่ที่สร้างขึ้นสามารถกำหนดได้จากค่า Ra , Rb และ Ca จากสูตร โดยในวงจรที่ใช้ต้องการสร้างพัลส์ความถี่ประมาณ 38 kHz เพื่อให้ตรงกับความถี่ของวงจรทางด้านรับสัญญาณอินฟราเรด โดยสัญญาณพัลส์ที่สร้างได้จะถูกส่งไปยังวงจรไดรว์ โดยใช้เพาเวอร์ทรานซิสเตอร์เบอร์ 2SC1815

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 แสดงวงจรอินฟราเรดทางด้านส่ง

ส่วนวงจรทางด้านรับสัญญาณอินฟราเรดจะใช้โฟโต้ไดโอด(Photo Diode) เบอร์ 4838 ซึ่งสามารถรับสัญญาณอินฟราเรดที่ความถี่ 38 kHz ซึ่งภายในโฟโต้ไดโอดเบอร์ 4838 จะประกอบด้วยส่วนรับสัญญาณอินฟราเรด และภาคขยายสัญญาณที่เอาท์พุท



รูปที่ 3.3 แสดงวงจรอินฟราเรดทางด้านรับ

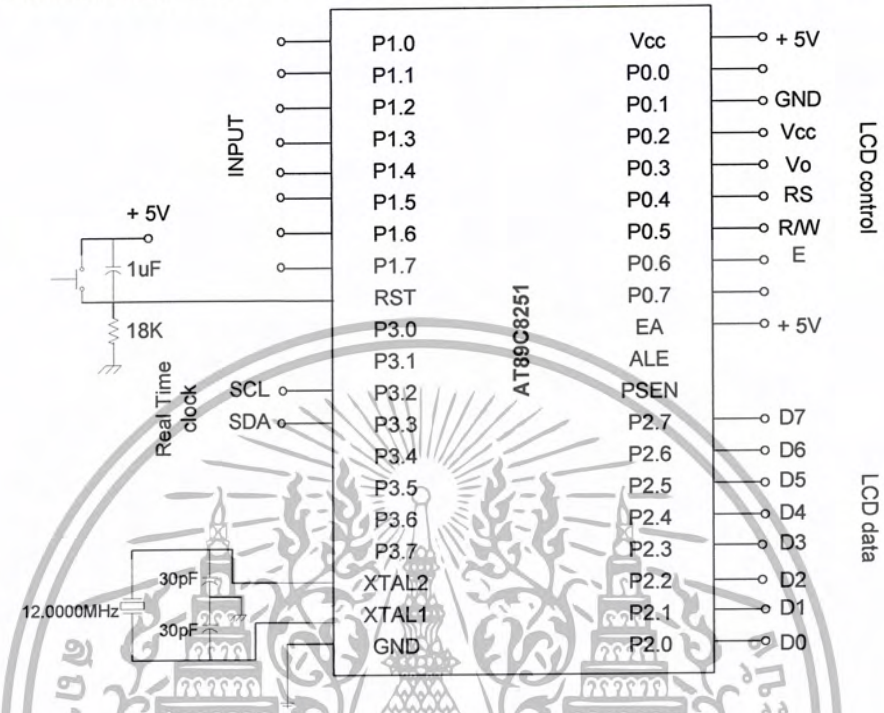
3.1.2 ส่วนของไมโครคอนโทรลเลอร์

AT89C8251 เป็นไมโครคอนโทรลเลอร์ขนาด 8 บิต และเป็นไอซีประเภท CMOS ทำให้มีประสิทธิภาพสูง กินไฟต่ำ มีหน่วยความจำแบบแฟลช บรรจุภายในตัวขนาด 8 กิโลไบต์ ทั้งยังทำงานร่วมและทดแทนไมโครคอนโทรลเลอร์ ตระกูล MCS-51 ได้ทั้งชุดคำสั่งและการจัดเรียงขา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การที่มีหน่วยความจำแบบแฟลช ทำให้เครื่องคอมพิวเตอร์สามารถ ทำการลบและเขียนโปรแกรมได้โดยตรง โดยไม่ต้องถอด ไมโครคอนโทรลเลอร์ ออกจากแผงวงจร ในลักษณะที่เรียกว่าอินซิสเต็มโปรแกรมมิ่ง (In System Programming) ซึ่งเหมาะต่อการพัฒนาโปรแกรม

ในโครงงานนี้ทำการต่อวงจรไมโครคอนโทรลเลอร์ เพื่อควบคุมการทำงานของวงจรต่างๆ ดังรูป

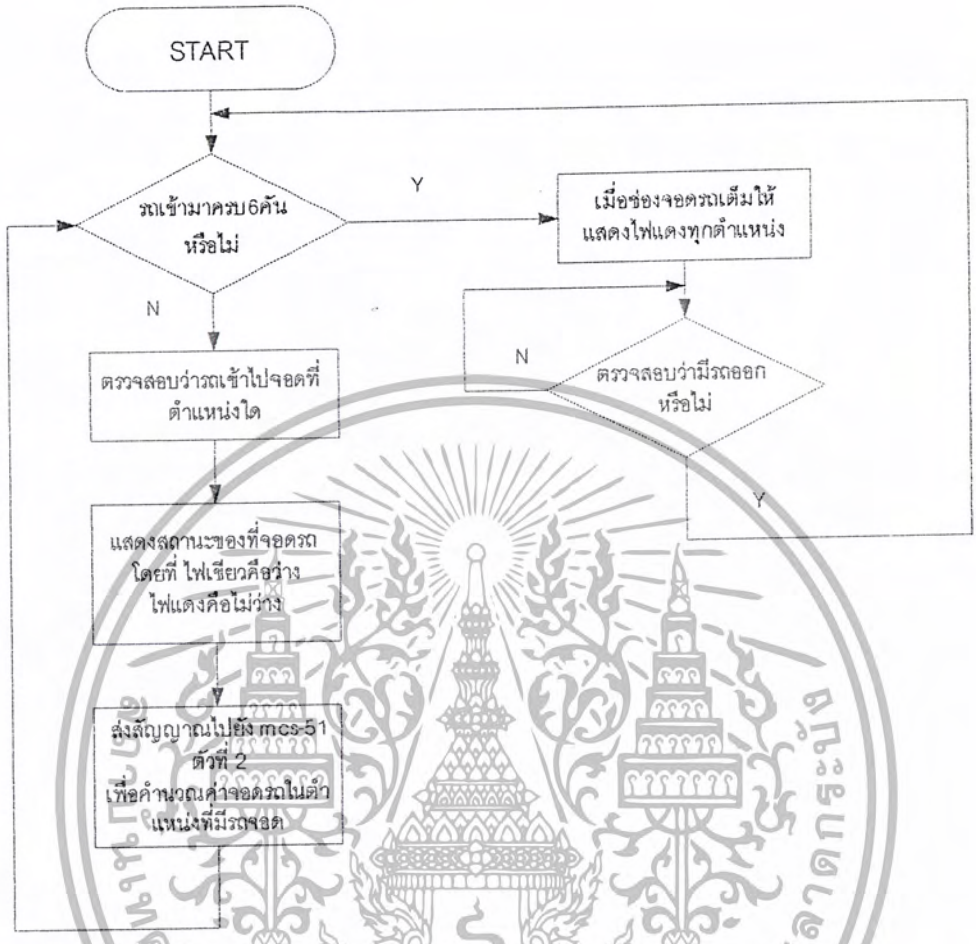


รูปที่ 3.4 แสดงรูปการต่อวงจรไมโครคอนโทรลเลอร์ เพื่อควบคุมวงจรต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 โครงสร้างทางซอฟท์แวร์

3.2.1 โฟว์ชาร์ทแสดงการทำงานของโปรแกรม

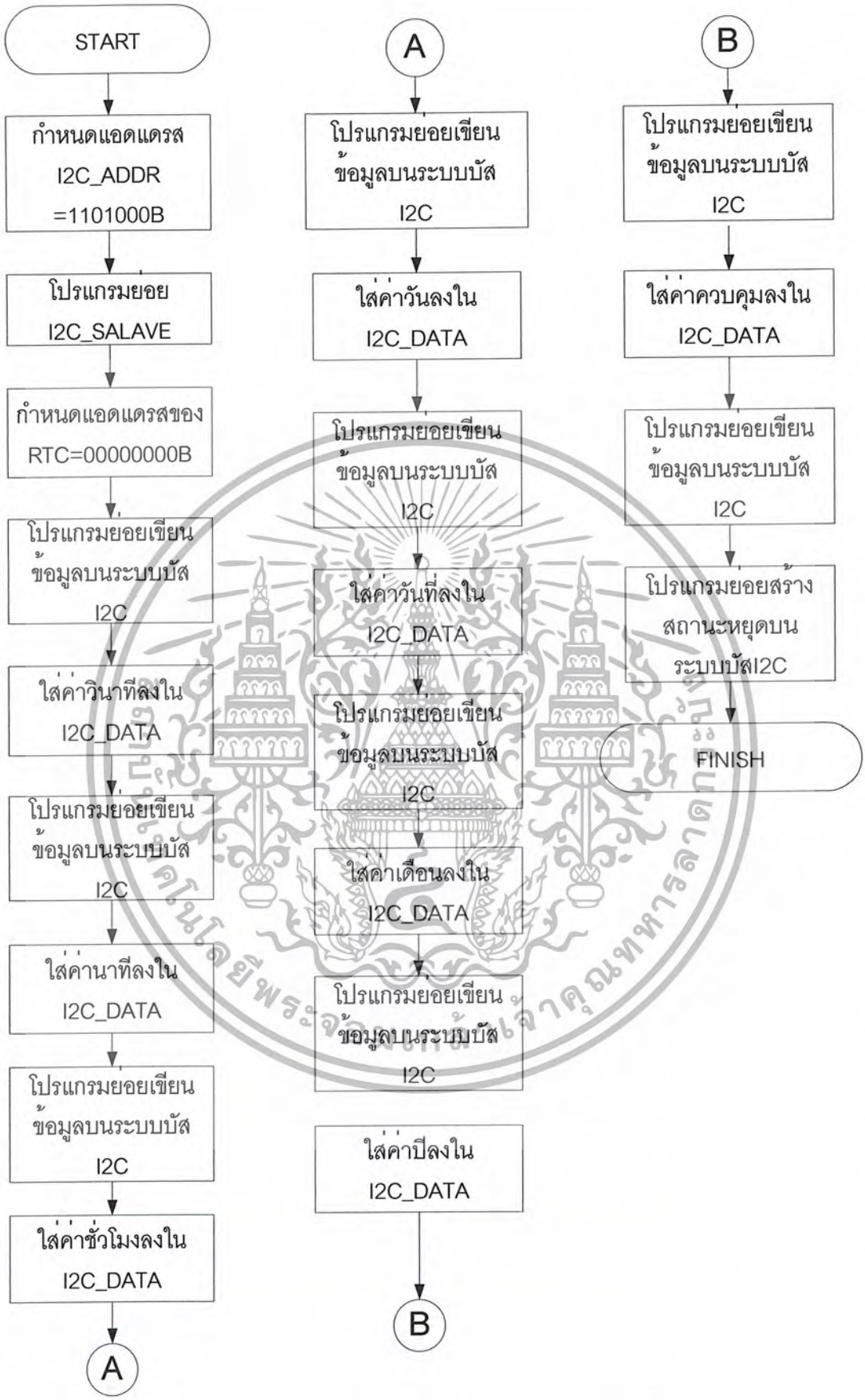


รูปที่3.5 แสดงโฟว์ชาร์ท โปรแกรมหลักการควบคุมการทำงานของไมโครคอนโทรลเลอร์ตัวที่หนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

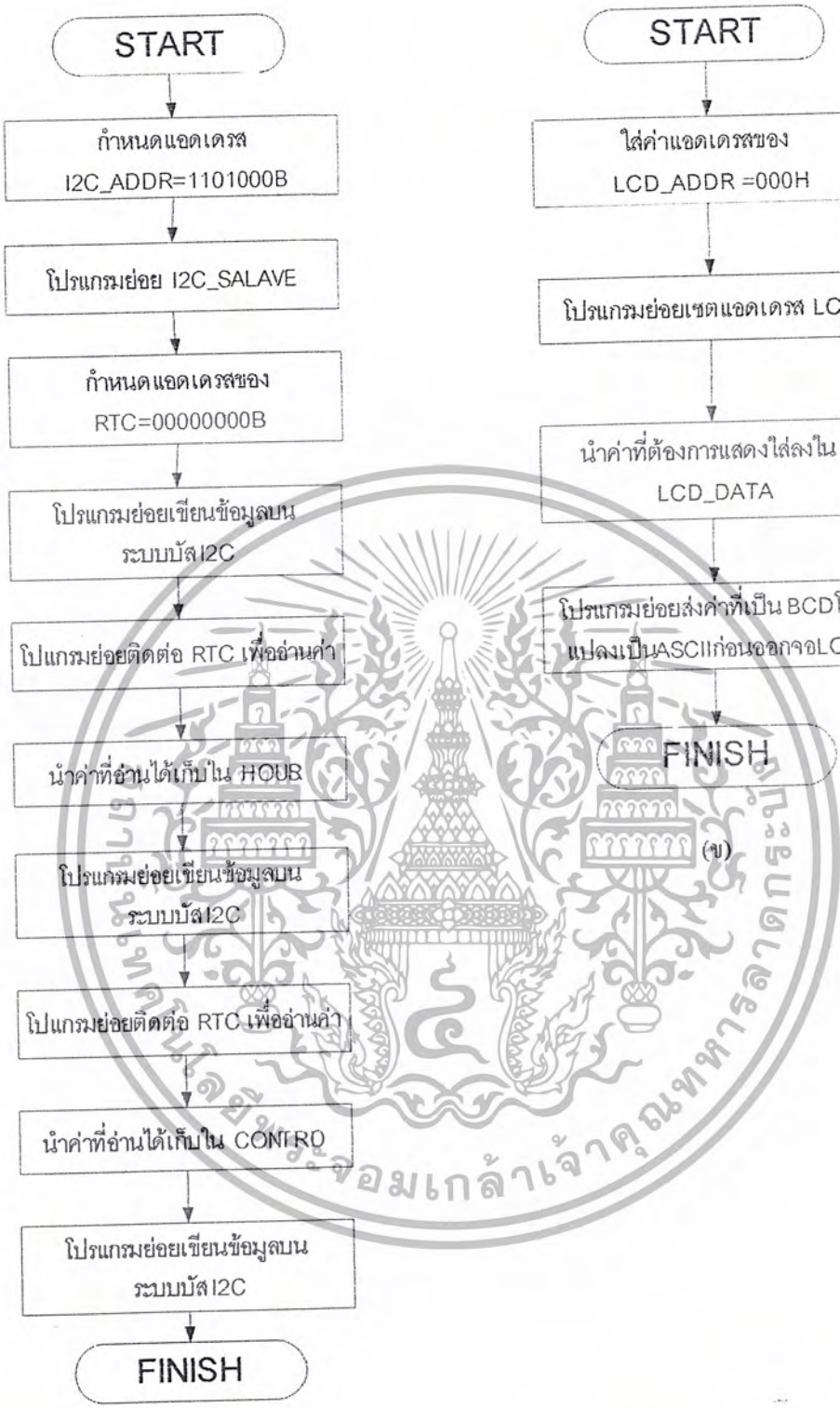
รูปที่3.6 แสดงโฟว์ชาร์ทโปรแกรมหลักในการควบคุมการทำงานของไมโครคอนโทรลเลอร์ตัวที่สอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่3.7 แสดงโฟว์ชาร์ท ของโปรแกรมย่อยในการเขียนค่าเวลาลงในไอซี 1307

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่3.8(ก) แสดงโฟว์ชาร์ท ของโปรแกรมย่อยในการอ่านค่าเวลาจากไอซี 1307

รูปที่3.8(ข) แสดงโฟว์ชาร์ท ของโปรแกรมย่อยในการแสดงผลออกจอLCD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2โปรแกรมที่ใช้ในการทำงาน

```
;*********************************
; Define
;*********************************
SEN_IN          BIT         P0.0
SEN_11          BIT         P0.1
SEN_12          BIT         P0.2
SEN_13          BIT         P0.3
SEN_14          BIT         P0.4
SEN_15          BIT         P0.5
SEN_16          BIT         P0.6
SEN_OT          BIT         P0.7
SENSOR_IN       BIT         P1.0
SENSOR_11       BIT         P1.1
SENSOR_12       BIT         P1.2
SENSOR_13       BIT         P1.3
SENSOR_14       BIT         P1.4
SENSOR_15       BIT         P1.5
SENSOR_16       BIT         P1.6
SENSOR_OT       BIT         P1.7
LED_R_10        BIT         P2.0
LED_R_11        BIT         P2.1
LED_R_12        BIT         P2.2
LED_R_13        BIT         P2.3
LED_R_14        BIT         P2.4
LED_R_15        BIT         P2.5
LED_R_16        BIT         P2.6
LED_R_17        BIT         P2.7
LED_G_10        BIT         P3.0
LED_G_11        BIT         P3.1
LED_G_12        BIT         P3.2
LED_G_13        BIT         P3.3
LED_G_14        BIT         P3.4
LED_G_15        BIT         P3.5
LED_G_16        BIT         P3.6
LED_G_17        BIT         P3.7

;*********************************
; Main Program
;*********************************
                ORG         0000H

                MOV         A,#00H
                SETB        LED_G_10
                SETB        LED_G_17
                CLR         LED_R_10
                CLR         LED_R_17
                SJMP        MAIN

FULLED:         LCALL       DELAY_100ms
                MOV         P3,#00H
                MOV         P2,#0FFH
                LCALL       DELAY_100ms
                LCALL       DELAY_100ms
A_0:            JNB         SENSOR_OT,A_1
                SETB        SEN_OT
                DEC         A
                LCALL       DELAY_2s
                SJMP        MAIN
A_1:            LR          SEN_OT
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                JMP         A_0
MAIN:           LCALL       DELAY_10ms
EMP_10:         JNB         SENSOR_IN,A_2       ;NO CAR COME IN  (COUNT)
                SETB        SEN_IN
                INC         A
                LCALL       DELAY_2s
                SJMP        D_2
A_2:            CLR         SEN_IN
D_2             JNB         SENSOR_OT,B_2
                SETB        SEN_OT
                DEC         A
                LCALL       DELAY_2s
B_2:            CLR         SEN_OT
                CJNE        A,#06H,EMP_11       ;IF = 6 SHOW INSIDE FULLED

                LJMP        FULLED
EMP_11:         JNB         SENSOR_11,A_3
                SETB        SEN_11
                CLR         LED_G_11
                SETB        LED_R_11
                JNB         SENSOR_IN,B_3       ;NO CAR COME IN  (COUNT)
                SETB        SEN_IN
                INC         A
                LCALL       DELAY_2s
B_3:            CLR         SEN_IN
                JNB         SENSOR_OT,D_3
                SETB        SEN_OT
                DEC         A
                LCALL       DELAY_2s
D_3:            CLR         SEN_OT
                CJNE        A,#06H,EMP_12       ;IF = 6 SHOW INSIDE FULLED

                LJMP        FULLED
A_3:            LJMP        NEMP_10
EMP_12:         JNB         SENSOR_12,A_4
                SETB        SEN_12
                CLR         LED_G_12
                SETB        LED_R_12
                JNB         SENSOR_IN,B_4       ;NO CAR COME IN  (COUNT)
                SETB        SEN_IN
                INC         A
                LCALL       DELAY_2s
B_4:            CLR         SEN_IN
                JNB         SENSOR_OT,D_4
                SETB        SEN_OT
                DEC         A
                LCALL       DELAY_2s
D_4:            CLR         SEN_OT
                CJNE        A,#06H,EMP_13
                LJMP        FULLED
A_4:            LJMP        NEMP_11
EMP_13:         JNB         SENSOR_11,A_5
                SETB        SEN_13
                CLR         LED_G_13
                SETB        LED_R_13
                JNB         SENSOR_IN,B_5       ;NO CAR COME IN  (COUNT)
                SETB        SEN_IN
                INC         A
                LCALL       DELAY_2s
B_5:            CLR         SEN_IN
                JNB         SENSOR_OT,D_5
                SETB        SEN_OT
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                DEC         A
                LCALL       DELAY_2s
D_5:            CLR         SEN_OT
                CJNE        A,#06H,EMP_14
                LJMP        FULLED
A_5:            LJMP        NEMP_12

EMP_14:         JNB         SENSOR_14,A_6
                SETB        SEN_14
                CLR         LED_G_14
                SETB        LED_R_14
                JNB         SENSOR_IN,B_6      ;NO CAR COME IN  (COUNT)
                SETB        SEN_IN
                INC         A
                LCALL       DELAY_2s
B_6:            CLR         SEN_IN
                JNB         SENSOR_OT,D_6
                SETB        SEN_OT
                DEC         A
                LCALL       DELAY_2s
D_6:            CLR         SEN_OT
                CJNE        A,#06H,EMP_15
                LJMP        FULLED
A_6:            LJMP        NEMP_13

EMP_15:         JNB         SENSOR_15,A_7
                SETB        SEN_15
                CLR         LED_G_15
                SETB        LED_R_15
                JNB         SENSOR_IN,B_7      ;NO CAR COME IN  (COUNT)
                SETB        SEN_IN
                INC         A
                LCALL       DELAY_2s
B_7:            CLR         SEN_IN
                JNB         SENSOR_OT,D_7
                SETB        SEN_OT
                DEC         A
                LCALL       DELAY_2s
D_7:            CLR         SEN_OT
                CJNE        A,#06H,EMP_16
                LJMP        FULLED
A_7:            LJMP        NEMP_14
EMP_16:         JNB         SENSOR_16,A_8
                SETB        SEN_16
                CLR         LED_G_16
                SETB        LED_R_16
                JNB         SENSOR_IN,B_8      ;NO CAR COME IN  (COUNT)
                SETB        SEN_IN
                INC         A
                LCALL       DELAY_2s
B_8:            CLR         SEN_IN
                JNB         SENSOR_OT,D_8
                SETB        SEN_OT
                DEC         A
                LCALL       DELAY_2s
D_8:            CLR         SEN_OT
                CJNE        A,#06H,A_M
                LJMP        FULLED
A_8:            LJMP        NEMP_15
A_M:            LJMP        MAIN
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
NEMP_10:        SETB    LED_G_11
                CLR     SEN_11
                CLR     LED_R_11
                JNB     SENSOR_IN,B_9       ;NO CAR COME IN  (COUNT)
                SETB    SEN_IN
                INC     A
                LCALL   DELAY_2s
B_9:            CLR     SEN_IN
                JNB     SENSOR_OT,D_9
                SETB    SEN_OT
                DEC     A
                LCALL   DELAY_2s
D_9:            CLR     SEN_OT
                CJNE    A,#06H,A_9
                LJMP    FULLED
A_9:            LJMP    EMP_10
NEMP_11:        SETB    LED_G_12
                CLR     SEN_12
                CLR     LED_R_12
                JNB     SENSOR_IN,B_10      ;NO CAR COME IN  (COUNT)
                SETB    SEN_IN
                INC     A
                LCALL   DELAY_2s
B_10:           CLR     SEN_IN
                JNB     SENSOR_OT,D_10
                SETB    SEN_OT
                DEC     A
                LCALL   DELAY_2s
D_10:           CLR     SEN_OT
                CJNE    A,#06H,A_10
                LJMP    FULLED
A_10:           LJMP    EMP_13

NEMP_12:        SETB    LED_G_13
                CLR     SEN_13
                CLR     LED_R_13
                JNB     SENSOR_IN,B_11      ;NO CAR COME IN  (COUNT)
                SETB    SEN_IN
                INC     A
                LCALL   DELAY_2s
B_11:           CLR     SEN_IN
                JNB     SENSOR_OT,D_11
                SETB    SEN_OT
                DEC     A
                LCALL   DELAY_2s
D_11:           CLR     SEN_OT
                CJNE    A,#06H,A_11
                LJMP    FULLED
A_11:           LJMP    EMP_14

NEMP_13:        SETB    LED_G_14
                CLR     SEN_14
                CLR     LED_R_14
                JNB     SENSOR_IN,B_12      ;NO CAR COME IN  (COUNT)
                SETB    SEN_IN
                INC     A
                LCALL   DELAY_2s
B_12:           CLR     SEN_IN
                JNB     SENSOR_OT,D_12
                SETB    SEN_OT
                DEC     A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                LCALL   DELAY_2s
D_12:           CLR     SEN_OT
                CJNE    A,#06H,A_12
                LJMP    FULLED
A_12:           LJMP    EMP_15
NEMP_14:        SETB    LED_G_15
                CLR     SEN_15
                CLR     LED_R_15
                JNB     SENSOR_IN,B_13      ;NO CAR COME IN  (COUNT)
                SETB    SEN_IN
                INC     A
                LCALL   DELAY_2s
B_13:           CLR     SEN_IN
                JNB     SENSOR_OT,D_13
                SETB    SEN_OT
                DEC     A
                LCALL   DELAY_2s
D_13:           CLR     SEN_OT
                CJNE    A,#06H,A_13
                LJMP    FULLED
A_13:           LJMP    EMP_16
NEMP_15:        SETB    LED_G_16
                CLR     SEN_16
                CLR     LED_R_16
                JNB     SENSOR_IN,B_14      ;NO CAR COME IN  (COUNT)
                SETB    SEN_IN
                INC     A
                LCALL   DELAY_2s
B_14:           CLR     SEN_IN
                JNB     SENSOR_OT,D_14
                SETB    SEN_OT
                DEC     A
                LCALL   DELAY_2s
D_14:           CLR     SEN_OT
                CJNE    A,#06H,A_14
                LJMP    FULLED
A_14:           LJMP    MAIN

;*********************************
;DELAY
;*********************************
I2C_DELAY:      MOV     R6,#00CH
I2C_DELAY_1:    NOP
                NOP
                DJNZ    R6,I2C_DELAY_1
                RET

RTC_DELAY:      MOV     R7,#48              ;DELAY 72ns
RTC_DELAY_1:    MOV     R6,03H
RTC_DELAY_2:    MOV     R5,00H
RTC_DELAY_3:    DJNZ    R5,RTC_DELAY_3
                DJNZ    R6,RTC_DELAY_2
                DJNZ    R7,RTC_DELAY_1
                RET
RTCDL:          MOV     R5,#4               ;DELAY
                DJNZ    R5,$
                RET

DELAY_50us:     MOV     R7,02H
DELAY_50us_1:   MOV     R6,03H
DELAY_50us_2:   MOV     R5,02H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
DELAY_50us_3:   DJNZ    R5,DELAY_50us_3
                DJNZ    R6,DELAY_50us_2
                DJNZ    R7,DELAY_50us_1
                RET

LCD_DELAY:      MOV     R7,#002
LCD_DELAY_1:    MOV     R6,#0E6H
LCD_DELAY_2:    NOP
                NOP
                DJNZ    R6,LCD_DELAY_2
                DJNZ    R7,LCD_DELAY_1
                RET

LCD_DELAY2:     ACALL   LCD_DELAY
                ACALL   LCD_DELAY
                ACALL   LCD_DELAY
                ACALL   LCD_DELAY
                ACALL   LCD_DELAY
                ACALL   LCD_DELAY
                ACALL   LCD_DELAY
                ACALL   LCD_DELAY
                ACALL   LCD_DELAY
                ACALL   LCD_DELAY
                RET

LCD_DELAY3:     ACALL   LCD_DELAY2
                ACALL   LCD_DELAY2
                ACALL   LCD_DELAY2
                RET

DELAY_10ms:     MOV     R7,010
DELAY_10ms_1:   MOV     R6,0E6H
DELAY_10ms_2:   NOP
                NOP
                DJNZ    R6,DELAY_10ms_2
                DJNZ    R7,DELAY_10ms_1
                RET

DELAY_100ms:    LCALL   DELAY_10ms
                LCALL   DELAY_10ms
                LCALL   DELAY_10ms
                LCALL   DELAY_10ms
                LCALL   DELAY_10ms
                LCALL   DELAY_10ms
                LCALL   DELAY_10ms
                LCALL   DELAY_10ms
                LCALL   DELAY_10ms
                LCALL   DELAY_10ms
                RET

DELAY_1s:       LCALL   DELAY_100ms
                LCALL   DELAY_100ms
                LCALL   DELAY_100ms
                LCALL   DELAY_100ms
                LCALL   DELAY_100ms
                LCALL   DELAY_100ms
                LCALL   DELAY_100ms
                LCALL   DELAY_100ms
                LCALL   DELAY_100ms
                RET
                END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;*********************************
;DEF
;*********************************
LCD_DATA        EQU     020H
I2C_ADDR        EQU     021H
I2C_DATA        EQU     022H
LCD_ADDR        EQU     023H
SECOND          EQU     040H
MINUTE          EQU     041H
HOUR            EQU     042H
DAY             EQU     043H
DATE            EQU     044H
MONTH           EQU     045H
YEAR            EQU     046H
CONTROL         EQU     047H
TIM_01          EQU     048H
TIM_02          EQU     049H
TIM_03          EQU     04AH
TIM_04          EQU     04BH
TIM_05          EQU     04CH
TIM_06          EQU     04DH
TIM_11          EQU     04EH
TIM_12          EQU     04FH
TIM_13          EQU     050H
TIM_14          EQU     051H
TIM_15          EQU     052H
TIM_16          EQU     053H
TIM_21          EQU     054H
TIM_22          EQU     055H
TIM_23          EQU     056H
TIM_24          EQU     057H
TIM_25          EQU     058H
TIM_26          EQU     059H
TIM_31          EQU     05AH
TIM_32          EQU     05BH
TIM_33          EQU     05CH
TIM_34          EQU     05DH
TIM_35          EQU     05EH
TIM_36          EQU     05FH
TIM_41          EQU     060H
TIM_42          EQU     061H
TIM_43          EQU     062H
TIM_44          EQU     063H
TIM_45          EQU     064H
TIM_46          EQU     065H
TIM_51          EQU     066H
TIM_52          EQU     067H
TIM_53          EQU     068H
TIM_54          EQU     069H
TIM_55          EQU     06AH
TIM_56          EQU     06BH
TIM_IN_H        EQU     071H
TIM_IN_M        EQU     072H
TIM_IN_S        EQU     073H
KET_01          EQU     074H
KET_02          EQU     075H
KET_03          EQU     076H
DIF_M1          EQU     077H
DIF_S1          EQU     078H
DIF_H1          EQU     079H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
DIF_M2          EQU     07AH
DIF_S2          EQU     07BH
DIF_H2          EQU     07CH
DIF_M3          EQU     07DH
DIF_S3          EQU     07EH
DIF_H3          EQU     07FH
DIF_M4          EQU     017H
DIF_S4          EQU     018H
DIF_H4          EQU     019H
DIF_M5          EQU     01AH
DIF_S5          EQU     01BH
DIF_H5          EQU     01CH
DIF_M6          EQU     01DH
DIF_S6          EQU     01EH
DIF_H6          EQU     01FH
SEN_IN          BIT     P1.0
SEN_21          BIT     P1.1
SEN_22          BIT     P1.2
SEN_23          BIT     P1.3
SEN_24          BIT     P1.4
SEN_25          BIT     P1.5
SEN_26          BIT     P1.6
SEN_OT          BIT     P1.7
SCL             BIT     P2.6
SDA             BIT     P2.5
EN              BIT     P3.7
RW              BIT     P3.6
RS              BIT     P3.5
RTC_ID          EQU     11010000B
FLAG            EQU     2FH
I2C_ACK         BIT     FLAG.0

;*********************************

                ORG     0000H

                SETB    SDA
                SETB    SCL

MAIN_0:         MOV     TIM_01,#00H
                MOV     TIM_02,#00H
                MOV     TIM_03,#00H
                MOV     TIM_04,#00H
                MOV     TIM_05,#00H
                MOV     TIM_06,#00H
                MOV     TIM_11,#00H
                MOV     TIM_12,#00H
                MOV     TIM_13,#00H
                MOV     TIM_14,#00H
                MOV     TIM_15,#00H
                MOV     TIM_16,#00H
                MOV     TIM_21,#00H
                MOV     TIM_22,#00H
                MOV     TIM_23,#00H
                MOV     TIM_24,#00H
                MOV     TIM_25,#00H
                MOV     TIM_26,#00H
                MOV     TIM_31,#00H
                MOV     TIM_32,#00H
                MOV     TIM_33,#00H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV     TIM_34,#00H
        MOV     TIM_35,#00H
        MOV     TIM_36,#00H
        MOV     TIM_41,#00H
        MOV     TIM_42,#00H
        MOV     TIM_43,#00H
        MOV     TIM_44,#00H
        MOV     TIM_45,#00H
        MOV     TIM_46,#00H
        MOV     TIM_51,#00H
        MOV     TIM_52,#00H
        MOV     TIM_53,#00H
        MOV     TIM_54,#00H
        MOV     TIM_55,#00H
        MOV     TIM_56,#00H
        MOV     TIM_IN_H,#00H
        MOV     TIM_IN_M,#00H
        MOV     TIM_IN_S,#00H
        MOV     R1,#00H
        MOV     R2,#00H
        MOV     R3,#00H
        MOV     R4,#00H
        MOV     R5,#00H
        MOV     R6,#00H
        LJMP    MAIN_1

KT_1:   LCALL   INITIAL             ;INITIAL LCD
        LCALL   RTC_RD              ;READ TIME
        MOV     LCD_DATA,HOUR
        LCALL   CHG                 ;CHANGE TIME FOR OPERATING MATH
        MOV     TIM_IN_H,KET_03
        LCALL   DELAY_1ms

        MOV     LCD_DATA,MINUTE
        LCALL   CHG
        MOV     TIM_IN_M,KET_03
        LCALL   DELAY_1ms

        MOV     LCD_DATA,SECOND
        LCALL   CHG
        MOV     TIM_IN_S,KET_03
        LCALL   DELAY_1ms

        RET

KT_2:   LCALL   INITIAL             ;INITIAL LCD
        LCALL   RTC_RD              ;READ TIME
        MOV     LCD_DATA,HOUR
        LCALL   BCD2LCD
        LCALL   LCD_DELAY3

        MOV     A,#':'              ;SENT  :
        MOV     P0,A
        MOV     P3,#0A0H            ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV     P3,#20H             ;EN,RW,RS,0,0,0,0,0==>00100000


        MOV     LCD_DATA,MINUTE
        LCALL   BCD2LCD
        LCALL   LCD_DELAY3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV     A,#':'          ;SENT  :
        MOV     P0,A
        MOV     P3,#0A0H        ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV     P3,#20H         ;EN,RW,RS,0,0,0,0,0==>00100000

        MOV     LCD_DATA,SECOND
        LCALL   BCD2LCD
        LCALL   LCD_DELAY3

        RET

CHG:    PUSH    ACC
        PUSH    B
        MOV     A,LCD_DATA
        LCALL   LCD_DELAY
        ANL     A,#0F0H
        LCALL   LCD_DELAY
        SWAP    A
        LCALL   LCD_DELAY
        MOV     B,#10
        MUL     AB
        MOV     KET_02,A
        MOV     A,LCD_DATA
        LCALL   LCD_DELAY
        ANL     A,#0FH
        LCALL   LCD_DELAY
        ADD     A,KET_02
        MOV     KET_03,A
        POP     B
        POP     ACC
        RET

MAIN_1: LCALL   KT_2            ;SHOW TIME NOW
        JB      SEN_IN,CH_N1    ;CHECK CAR   0 = NO CAR
        LCALL   DELAY_10ms
        LJMP    CHK_OUT

CH_N1:  JNB     SEN_21,CH_N2    ;CHECK CAR IN CH1
        CJNE    R1,#00H,CH_N2   ;CHECK FOR NO KEEP TIME
                                ;REPETITION
        LCALL   KT_1
        MOV     TIM_01,TIM_IN_H
        MOV     TIM_02,TIM_IN_M
        MOV     TIM_03,TIM_IN_S

        MOV     R1,#01H
        LJMP    CHK_OUT

CH_N2:  JNB     SEN_22,CH_N3    ;CHECK CAR IN CH2
        CJNE    R2,#00H,CH_N3   ;CHECK FOR NO KEEP TIME
                                ;REPETITION
        LCALL   KT_1
        MOV     TIM_11,TIM_IN_H
        MOV     TIM_12,TIM_IN_M
        MOV     TIM_13,TIM_IN_S
        MOV     R2,#01H
        LJMP    CHK_OUT
CH_N3:  JNB     SEN_23,CH_N4    ;CHECK CAR IN CH3
        CJNE    R3,#00H,CH_N4   ;CHECK FOR NO KEEP TIME
                                ;REPETITION
        LCALL   KT_1
        MOV     TIM_21,TIM_IN_H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV     TIM_22,TIM_IN_M
                MOV     TIM_23,TIM_IN_S
                MOV     R3,#01H
                LJMP    CHK_OUT

CH_N4:          JNB     SEN_24,CH_N5        ;CHECK CAR IN CH4
                CJNE    R4,#00H,CH_N5       ;CHECK FOR NO KEEP TIME
                                            ;REPETITION
                LCALL   KT_1
                MOV     TIM_31,TIM_IN_H
                MOV     TIM_32,TIM_IN_M
                MOV     TIM_33,TIM_IN_S

                MOV     R4,#01H
                LJMP    CHK_OUT

CH_N5:          JNB     SEN_25,CH_N6        ;CHECK CAR IN CH5
                CJNE    R5,#00H,CH_N6       ;CHECK FOR NO KEEP TIME
                                            ;REPETITION
                LCALL   KT_1
                MOV     TIM_41,TIM_IN_H
                MOV     TIM_42,TIM_IN_M
                MOV     TIM_43,TIM_IN_S
                MOV     R5,#01H
                LJMP    CHK_OUT

CH_N6:          JNB     SEN_26,J2N1         ;CHECK CAR IN CH6
                CJNE    R6,#00H,J2N1        ;CHECK FOR NO KEEP TIME
                                            ;REPETITION
                LCALL   KT_1
                MOV     TIM_51,TIM_IN_H
                MOV     TIM_52,TIM_IN_M
                MOV     TIM_53,TIM_IN_S

                MOV     R6,#01H
                LJMP    CHK_OUT
J2N1:           LJMP    CH_N1


CHK_OUT:        LCALL   DELAY_10ms          ;CHECK CAR OUT

CH_U1:          JB      SEN_21,TO_CH_U2     ;CAR OUT SEN_21 = 0
                LCALL   DELAY_10ms
                CJNE    R1,#01H,TO_CH_U2    ;CHECK FOR NO KEEP TIME
                                            ;REPETITION
                LCALL   KT_1
                MOV     TIM_04,TIM_IN_H
                MOV     TIM_05,TIM_IN_M
                MOV     TIM_06,TIM_IN_S
                SJMP    DIS_T1

TO_CH_U2:       LJMP    CH_U2

DIS_T1:         LCALL   INITIAL
                MOV     A,TIM_01            ;SHOW TIME CAR IN
                MOV     B,#0AH              ;HH
                DIV     AB
                ADD     A,#30H
                MOV     P0,A
                MOV     P3,#0A0H
                MOV     P3,#20H
                LCALL   LCD_DELAY3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
MOV         A,B
ADD         A,#30H
MOV         P0,A
MOV         P3,#0A0H            ;EN,RW,RS,0,0,0,0,0==>10100000
MOV         P3,#20H             ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL       LCD_DELAY3
MOV         A,#':'              ;SENT  :
MOV         P0,A
MOV         P3,#0A0H            ;EN,RW,RS,0,0,0,0,0==>10100000
MOV         P3,#20H             ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL       LCD_DELAY3

MOV         A,TIM_02            ;SHOW TIME IN
MOV         B,#0AH              ;MINUTE
DIV         AB
ADD         A,#30H
MOV         P0,A
MOV         P3,#0A0H            ;EN,RW,RS,0,0,0,0,0==>10100000
MOV         P3,#20H             ;EN,RW,RS,0,0,0,0,0==>00100000

LCALL       LCD_DELAY3
MOV         A,B
ADD         A,#30H
MOV         P0,A
MOV         P3,#0A0H            ;EN,RW,RS,0,0,0,0,0==>10100000
MOV         P3,#20H             ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL       LCD_DELAY3

MOV         A,#"-"
MOV         P0,A
MOV         P3,#0A0H            ;EN,RW,RS,0,0,0,0,0==>10100000
MOV         P3,#20H             ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL       LCD_DELAY3

MOV         A,TIM_04            ;SHOW TIME CAR OUT
MOV         B,#0AH              ;HH
DIV         AB
ADD         A,#30H
MOV         P0,A
MOV         P3,#0A0H            ;EN,RW,RS,0,0,0,0,0==>10100000
MOV         P3,#20H             ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL       LCD_DELAY3

MOV         A,B
ADD         A,#30H
MOV         P0,A
MOV         P3,#0A0H            ;EN,RW,RS,0,0,0,0,0==>10100000
MOV         P3,#20H             ;EN,RW,RS,0,0,0,0,0==>00100000

LCALL       LCD_DELAY3
MOV         A,#':'              ;SENT  :
MOV         P0,A
MOV         P3,#0A0H            ;EN,RW,RS,0,0,0,0,0==>10100000
MOV         P3,#20H             ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL       LCD_DELAY3

MOV         A,TIM_05            ;SHOW TIME CAR OUT
MOV         B,#0AH              ;MM
DIV         AB
ADD         A,#30H
MOV         P0,A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV         P3,#0A0H            ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV         P3,#20H             ;EN,RW,RS,0,0,0,0,0==>00100000

                LCALL       LCD_DELAY3
                MOV         A,B
                ADD         A,#30H
                MOV         P0,A
                MOV         P3,#0A0H            ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3


                MOV   A,#"-"
                MOV   P0,A
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                LCALL DELAY_600ms
                MOV   A,TIM_04                  ;OPERATE TO FIND DIFF TIME & COST
                SUBB  A,TIM_01                  ;DIFFERENCE OF HOUR
                JC    O1                        ; H2 < H1  IMPOSSIBLE
                CJNE  A,#00H,CH_HR1             ; H2 = H1  COMPUTE DIF MIN IF
                                                ;FLASE JUMP COMPUTE DIF  HOUR
                MOV   A,TIM_05
                SUBB  A,TIM_02                  ; DIFFERENCE OF MIN
                      JC    O1                  ; M2 < M1 IMPOSIBLE
                MOV   DIF_M1,A                  ; M2 >= M1 KEEP DIF MIN
                SJMP  CH_MIN11

CH_HR1:         MOV   DIF_H1,A                  ; KEPP DIF HOUR
CH_MIN11:       MOV   A,TIM_05
                SUBB  A,TIM_02                  ;DIFFERENCE OF MIN2>=MIN1
                JC    CH_MIN12
                MOV   DIF_M1,A
                SJMP  CH_HM1
CH_MIN12:       MOV   A,TIM_05
                ADD   A,#60
                SUBB  A,TIM_02                  ;DIFFERENCE OF MIN2<MIN1
                MOV   DIF_M1,A
                DEC   DIF_H1

CH_HM1          MOV   A,DIF_H1
                CJNE  A,#00H,EX_CH13
                SJMP  EX_CH12

EX_CH12:        MOV   A,DIF_M1
                SUBB  A,#01H
                JC    O1
                SJMP  EX_CH11

EX_CH13:        MOV   A,DIF_M1
                SUBB  A,#01H
                JC    OO1
EX_CH11:        INC   DIF_H1
                MOV   A,DIF_H1
                SJMP  D12

OO1:            MOV   DIF_M1,#00H
D12:            MOV   A,DIF_M1
                MOV   B,#0AH
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                DIV   AB
                ADD   A,#30H
                MOV   P0,A
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                MOV   A,B
                ADD   A,#30H
                MOV   P0,A
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000

                LCALL DELAY_600ms
                MOV   A,DIF_M1
                MOV   R0,#1                     ;INDEX channel 1
                LCALL CAL_0

O1:             MOV   R1,#00H                   ;CLEAR
                MOV   TIM_01,#00H
                MOV   TIM_02,#00H
                MOV   TIM_03,#00H
                MOV   TIM_04,#00H
                MOV   TIM_05,#00H
                MOV   TIM_06,#00H
                MOV   DIF_H1,#00H
                MOV   DIF_M1,#00H

;*********************************
CH_U2:          JB    SEN_22,TO_CH_U3
                LCALL DELAY_10ms
                CJNE  R2,#01H,TO_CH_U3          ;CHECK FOR NO KEEP TIME REPETITION
                LCALL KT_1

                MOV   TIM_14,TIM_IN_H
                MOV   TIM_15,TIM_IN_M
                MOV   TIM_16,TIM_IN_S
                LJMP  DIS_T2

TO_CH_U3:       LJMP  CH_U3
DIS_T2:         LCALL INITIAL
                MOV   A,TIM_11
                MOV   B,#0AH
                DIV   AB
                ADD   A,#30H
                MOV   P0,A
                MOV   P3,#0A0H
                MOV   P3,#20H
                LCALL LCD_DELAY3

                MOV   A,B
                ADD   A,#30H
                MOV   P0,A
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000

                LCALL LCD_DELAY3
                MOV   A,#':'                    ;SENT  :
                MOV   P0,A
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
LCALL LCD_DELAY3

MOV   A,TIM_12
MOV   B,#0AH
DIV   AB
ADD   A,#30H
MOV   P0,A
MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000

LCALL LCD_DELAY3
MOV   A,B
ADD   A,#30H
MOV   P0,A
MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3

MOV   A,#"-"
MOV   P0,A
MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3

MOV   A,TIM_14
MOV   B,#0AH
DIV   AB
ADD   A,#30H
MOV   P0,A
MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3

MOV   A,B
ADD   A,#30H
MOV   P0,A
MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000

LCALL LCD_DELAY3
MOV   A,#':'                    ;SENT :
MOV   P0,A
MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3

MOV   A,TIM_15
MOV   B,#0AH
DIV   AB
ADD   A,#30H
MOV   P0,A
MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000

LCALL LCD_DELAY3
MOV   A,B
ADD   A,#30H
MOV   P0,A
MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV   A,#"-"
                MOV   P0,A
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                LCALL DELAY_600ms
                MOV   A,TIM_14
                SUBB  A,TIM_11                  ;DIFFERENCE OF HOUR
                JC    O2                        ; H2 < H1  IMPOSSIBLE
                CJNE  A,#00H,CH_HR2             ; H2 = H1  COMPUTE DIF MIN IF
                                                ;FLASE JUMP COMPUTE DIF  HOUR
                MOV   A,TIM_15
                SUBB  A,TIM_12                  ;DIFFERENCE OF MIN
                      JC    O2                  ; M2 < M1 IMPOSIBLE
                MOV   DIF_M2,A                  ; M2 >= M1 KEEP DIF MIN
                SJMP  CH_MIN21

CH_HR2:         MOV   DIF_H2,A                  ; KEPP DIF HOUR
CH_MIN21:       MOV   A,TIM_15
                SUBB  A,TIM_12                  ;DIFFERENCE OF MIN2>=MIN1
                JC    CH_MIN22
                MOV   DIF_M2,A
                SJMP  CH_HM2
CH_MIN22:       MOV   A,TIM_15
                ADD   A,#60
                SUBB  A,TIM_12                  ;DIFFERENCE OF MIN2<MIN1
                MOV   DIF_M2,A
                DEC   DIF_H2

CH_HM2          MOV   A,DIF_H2
                CJNE  A,#00H,EX_CH23
                SJMP  EX_CH22

EX_CH22:        MOV   A,DIF_M2
                SUBB  A,#01H
                JC    O2
                SJMP  EX_CH21
EX_CH23:        MOV   A,DIF_M2
                SUBB  A,#01H
                JC    OO2
EX_CH21:        INC   DIF_H2
                MOV   A,DIF_H2
                SJMP  D22

OO2:            MOV   DIF_M2,#00H
D22:            MOV   A,DIF_M2
                MOV   B,#0AH
                DIV   AB
                ADD   A,#30H
                MOV   P0,A
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                MOV   A,B
                ADD   A,#30H
                MOV   P0,A
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                LCALL DELAY_600ms
                MOV   A,DIF_M2
                MOV   R0,#2                     ;INDEX channel 2
                LCALL CAL_0

O2:             MOV   R2,#00H
                MOV   TIM_11,#00H
                MOV   TIM_12,#00H
                MOV   TIM_13,#00H
                MOV   TIM_14,#00H
                MOV   TIM_15,#00H
                MOV   TIM_16,#00H
                MOV   DIF_H2,#00H
                MOV   DIF_M2,#00H

;=========================  CH_U3 ========================
CH_U3:          JB    SEN_23,TO_CH_U4
                LCALL DELAY_10ms
                CJNE  R3,#01H,TO_CH_U4          ;CHECK FOR NO KEEP TIME REPETITION
                LCALL KT_1
                MOV   TIM_24,TIM_IN_H
                MOV   TIM_25,TIM_IN_M
                MOV   TIM_26,TIM_IN_S
                LJMP  DIS_T3

TO_CH_U4:       LJMP  CH_U4
DIS_T3:         LCALL INITIAL
                MOV   A,TIM_21
                MOV   B,#0AH
                DIV   AB
                ADD   A,#30H
                MOV   P0,A
                MOV   P3,#0A0H
                MOV   P3,#20H
                LCALL LCD_DELAY3

                MOV   A,B
                ADD   A,#30H
                MOV   P0,A
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                MOV   A,#':'                    ;SENT  :
                MOV   P0,A
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                MOV   A,TIM_22
                MOV   B,#0AH
                DIV   AB
                ADD   A,#30H
                MOV   P0,A
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000

                LCALL LCD_DELAY3
                MOV   A,B
                ADD   A,#30H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
MOV    P0,A
MOV    P3,#0A0H                    ;EN,RW,RS,0,0,0,0,0==>10100000
MOV    P3,#20H                     ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3

MOV    A,#"-"
MOV    P0,A
MOV    P3,#0A0H                    ;EN,RW,RS,0,0,0,0,0==>10100000
MOV    P3,#20H                     ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3

MOV    A,TIM_24
MOV    B,#0AH
DIV    AB
ADD    A,#30H
MOV    P0,A
MOV    P3,#0A0H                    ;EN,RW,RS,0,0,0,0,0==>10100000
MOV    P3,#20H                     ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3

MOV    A,B
ADD    A,#30H
MOV    P0,A
MOV    P3,#0A0H                    ;EN,RW,RS,0,0,0,0,0==>10100000
MOV    P3,#20H                     ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3

MOV    A,#':'                      ;SENT :
MOV    P0,A
MOV    P3,#0A0H                    ;EN,RW,RS,0,0,0,0,0==>10100000
MOV    P3,#20H                     ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3

MOV    A,TIM_25
MOV    B,#0AH
DIV    AB
ADD    A,#30H
MOV    P0,A
MOV    P3,#0A0H                    ;EN,RW,RS,0,0,0,0,0==>10100000
MOV    P3,#20H                     ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3

MOV    A,B
ADD    A,#30H
MOV    P0,A
MOV    P3,#0A0H                    ;EN,RW,RS,0,0,0,0,0==>10100000
MOV    P3,#20H                     ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3

MOV    A,#"-"
MOV    P0,A
MOV    P3,#0A0H                    ;EN,RW,RS,0,0,0,0,0==>10100000
MOV    P3,#20H                     ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3

LCALL DELAY_600ms
MOV    A,TIM_24
SUBB   A,TIM_21                    ;DIFFERENCE OF HOUR
JC     O3                          ; H2 < H1  IMPOSSIBLE
CJNE   A,#00H,CH_HR3               ; H2 = H1  COMPUTE DIF MIN IF
                                   ;FLASE JUMP COMPUTE DIF  HOUR
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV   A,TIM_25
                SUBB  A,TIM_22              ;DIFFERENCE OF MIN
                JC    O3                    ; M2 < M1 IMPOSIBLE
                MOV   DIF_M3,A              ; M2 >= M1 KEEP DIF MIN
                SJMP  CH_MIN31

CH_HR3:         MOV   DIF_H3,A              ; KEPP DIF HOUR
CH_MIN31:       MOV   A,TIM_25
                SUBB  A,TIM_22              ;DIFFERENCE OF MIN2>=MIN1
                JC    CH_MIN32
                MOV   DIF_M3,A
                SJMP  CH_HM3
CH_MIN32:       MOV   A,TIM_25
                ADD   A,#60
                SUBB  A,TIM_22              ;DIFFERENCE OF MIN2<MIN1
                MOV   DIF_M3,A
                DEC   DIF_H3
CH_HM3          MOV   A,DIF_H3
                CJNE  A,#00H,EX_CH33
                SJMP  EX_CH32

EX_CH32:        MOV   A,DIF_M3
                SUBB  A,#01H
                JC    O3
                SJMP  EX_CH31
EX_CH33:        MOV   A,DIF_M3
                SUBB  A,#01H
                JC    OO3
EX_CH31:        INC   DIF_H3
                MOV   A,DIF_H3
                SJMP  D32
OO3:            MOV   DIF_M3,#00H
D32:            MOV   A,DIF_M3
                MOV   B,#0AH
                DIV   AB
                ADD   A,#30H
                MOV   P0,A
                MOV   P3,#0A0H              ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H               ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                MOV   A,B
                ADD   A,#30H
                MOV   P0,A
                MOV   P3,#0A0H              ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H               ;EN,RW,RS,0,0,0,0,0==>00100000

                LCALL DELAY_600ms
                MOV   A,DIF_M3
                MOV   R0,#3                 ;INDEX channel 3
                LCALL CAL_0

O3:             MOV   R3,#00H
                MOV   TIM_21,#00H
                MOV   TIM_22,#00H
                MOV   TIM_23,#00H
                MOV   TIM_24,#00H
                MOV   TIM_25,#00H
                MOV   TIM_26,#00H
                MOV   DIF_H3,#00H
                MOV   DIF_M3,#00H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;============================ CH_U4 ========================

CH_U4:      JB    SEN_24,TO_CH_U5
            LCALL DELAY_10ms
            CJNE  R4,#01H,TO_CH_U5          ;CHECK FOR NO KEEP TIME REPETITION
            LCALL KT_1
            MOV   TIM_34,TIM_IN_H
            MOV   TIM_35,TIM_IN_M
            MOV   TIM_36,TIM_IN_S
            LJMP  DIS_T4
TO_CH_U5:   LJMP  CH_U5

DIS_T4:     LCALL INITIAL
            MOV   A,TIM_31
            MOV   B,#0AH
            DIV   AB
            ADD   A,#30H
            MOV   P0,A
            MOV   P3,#0A0H
            MOV   P3,#20H
            LCALL LCD_DELAY3

            MOV   A,B
            ADD   A,#30H
            MOV   P0,A
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,#':'                    ;SENT   :
            MOV   P0,A
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,TIM_32
            MOV   B,#0AH
            DIV   AB
            ADD   A,#30H
            MOV   P0,A
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,B
            ADD   A,#30H
            MOV   P0,A
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,#"-"
            MOV   P0,A
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3


            MOV   A,TIM_34
            MOV   B,#0AH
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        DIV   AB
        ADD   A,#30H
        MOV   P0,A
        MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   A,B
        ADD   A,#30H
        MOV   P0,A
        MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   A,#':'                  ;SENT  :
        MOV   P0,A
        MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   A,TIM_35
        MOV   B,#0AH
        DIV   AB
        ADD   A,#30H
        MOV   P0,A
        MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   A,B
        ADD   A,#30H
        MOV   P0,A
        MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   A,#"-"
        MOV   P0,A
        MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        LCALL DELAY_600ms
        MOV   A,TIM_34
        SUBB  A,TIM_31                ;DIFFERENCE OF HOUR
        JC    O4                      ; H2 < H1  IMPOSSIBLE
        CJNE  A,#00H,CH_HR4           ; H2 = H1  COMPUTE DIF MIN IF
                                      ;FLASE JUMP COMPUTE DIF HOUR
        MOV   A,TIM_35
        SUBB  A,TIM_32                ;DIFFERENCE OF MIN
        JC    O4                      ; M2 < M1 IMPOSIBLE
        MOV   DIF_M4,A                ; M2 >= M1 KEEP DIF MIN
        SJMP  CH_MIN41

CH_HR4: MOV   DIF_H4,A                ; KEPP DIF HOUR
CH_MIN41: MOV A,TIM_35
        SUBB  A,TIM_32                ;DIFFERENCE OF MIN2>=MIN1
        JC    CH_MIN42
        MOV   DIF_M4,A
        SJMP  CH_HM4
CH_MIN42: MOV A,TIM_35
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ADD   A,#60
                SUBB  A,TIM_32                ;DIFFERENCE OF MIN2<MIN1
                MOV   DIF_M4,A
                DEC   DIF_H4
CH_HM4:         MOV   A,DIF_H4
                CJNE  A,#00H,EX_CH43
                SJMP  EX_CH42

EX_CH42:        MOV   A,DIF_M4
                SUBB  A,#01H
                JC    O4
                SJMP  EX_CH41
EX_CH43:        MOV   A,DIF_M4
                SUBB  A,#01H
                JC    OO4

EX_CH41:        INC   DIF_H4
                MOV   A,DIF_H4
                SJMP  D42
OO4:            MOV   DIF_M4,#00H
D42:            MOV   A,DIF_M4
                MOV   B,#0AH
                DIV   AB
                ADD   A,#30H
                MOV   P0,A
                MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                MOV   A,B
                ADD   A,#30H
                MOV   P0,A
                MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000

                LCALL DELAY_600ms
                MOV   A,DIF_M4
                MOV   R0,#4                   ;INDEX channel 4
                LCALL CAL_0

O4:             MOV   R4,#00H
                MOV   TIM_31,#00H
                MOV   TIM_32,#00H
                MOV   TIM_33,#00H
                MOV   TIM_34,#00H
                MOV   TIM_35,#00H
                MOV   TIM_36,#00H
                MOV   DIF_H4,#00H
                MOV   DIF_M4,#00H


;*********************************
CH_U5:          JB    SEN_25,TO_CH_U6
                LCALL DELAY_10ms
                CJNE  R5,#01H,TO_CH_U6        ;CHECK FOR NO KEEP TIME REPETITION
                LCALL KT_1
                MOV   TIM_44,TIM_IN_H
                MOV   TIM_45,TIM_IN_M
                MOV   TIM_46,TIM_IN_S
                LJMP  DIS_T5
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
TO_CH_U6:   LJMP  CH_U6
DIS_T5:     LCALL INITIAL
            MOV   A,TIM_41
            MOV   B,#0AH
            DIV   AB
            ADD   A,#30H
            MOV   P0,A
            MOV   P3,#0A0H
            MOV   P3,#20H
            LCALL LCD_DELAY3

            MOV   A,B
            ADD   A,#30H
            MOV   P0,A
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,#':'                    ;SENT  :
            MOV   P0,A
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,TIM_32
            MOV   B,#0AH
            DIV   AB
            ADD   A,#30H
            MOV   P0,A
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3
            MOV   A,B
            ADD   A,#30H
            MOV   P0,A
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,#"-"
            MOV   P0,A
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,TIM_44
            MOV   B,#0AH
            DIV   AB
            ADD   A,#30H
            MOV   P0,A
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,B
            ADD   A,#30H
            MOV   P0,A
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000

            LCALL LCD_DELAY3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV   A,#':'                    ;SENT  :
                MOV   P0,A
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                MOV   A,TIM_45
                MOV   B,#0AH
                DIV   AB
                ADD   A,#30H
                MOV   P0,A
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                MOV   A,B
                ADD   A,#30H
                MOV   P0,A
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                MOV   A,#"-"
                MOV   P0,A
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                LCALL DELAY_600ms
                MOV   A,TIM_44
                SUBB  A,TIM_41                  ;DIFFERENCE OF HOUR
                JC    O5                        ;H2 < H1  IMPOSSIBLE
                CJNE  A,#00H,CH_HR5             ;H2 = H1  COMPUTE DIF MIN IF
                                                ;FLASE JUMP COMPUTE DIF  HOUR
                MOV   A,TIM_45
                SUBB  A,TIM_42                  ;DIFFERENCE OF MIN
                JC    O5                        ; M2 < M1 IMPOSIBLE
                MOV   DIF_M5,A                  ; M2 >= M1 KEEP DIF MIN
                SJMP  CH_MIN51

CH_HR5:         MOV   DIF_H5,A                  ; KEPP DIF HOUR
CH_MIN51:       MOV   A,TIM_45
                SUBB  A,TIM_42                  ;DIFFERENCE OF MIN2>=MIN1
                JC    CH_MIN52
                MOV   DIF_M5,A
                SJMP  CH_HM5

CH_MIN52:       MOV   A,TIM_45
                ADD   A,#60
                SUBB  A,TIM_42                  ;DIFFERENCE OF MIN2<MIN1
                MOV   DIF_M5,A
                DEC   DIF_H5
CH_HM5          MOV   A,DIF_H5
EX_CH52:        MOV   A,DIF_M5
                SUBB  A,#01H
                JC    O5
                SJMP  EX_CH51
EX_CH53:        MOV   A,DIF_M5
                SUBB  A,#01H
                JC   OO5
EX_CH51:        INC   DIF_H5
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV   A,DIF_H5
                SJMP  D52

OO5:            MOV   DIF_M5,#00H
D52:            MOV   A,DIF_M5
                MOV   B,#0AH
                DIV   AB
                ADD   A,#30H
                MOV   P0,A
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                MOV   A,B
                ADD   A,#30H
                MOV   P0,A
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000

                LCALL DELAY_600ms
                MOV   A,DIF_M5
                MOV   R0,#5                     ;INDEX channel 5
                LCALL CAL_0

O5:             MOV   R5,#00H
                MOV   TIM_41,#00H
                MOV   TIM_42,#00H
                MOV   TIM_43,#00H
                MOV   TIM_44,#00H
                MOV   TIM_45,#00H
                MOV   TIM_46,#00H
                MOV   DIF_H5,#00H
                MOV   DIF_M5,#00H

;*********************************
CH_U6:          JB    SEN_26,TO_CH_U7
                LCALL DELAY_10ms
                CJNE  R6,#01H,TO_CH_U7          ;CHECK FOR NO KEEP TIME REPETITION
                LCALL KT_1
                MOV   TIM_54,TIM_IN_H
                MOV   TIM_55,TIM_IN_M
                MOV   TIM_56,TIM_IN_S
                LJMP  DIS_T6

TO_CH_U7:       LJMP  JTM
DIS_T6:         LCALL INITIAL
                MOV   A,TIM_51
                MOV   B,#0AH
                DIV   AB
                ADD   A,#30H
                MOV   P0,A
                MOV   P3,#0A0H
                MOV   P3,#20H
                LCALL LCD_DELAY3

                MOV   A,B
                ADD   A,#30H
                MOV   P0,A
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
LCALL LCD_DELAY3
MOV   A,#':'                    ;SENT  :
MOV   P0,A
MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3

MOV   A,TIM_32
MOV   B,#0AH
DIV   AB
ADD   A,#30H
MOV   P0,A
MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3

MOV   A,B
ADD   A,#30H
MOV   P0,A
MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3

MOV   A,#"-"
MOV   P0,A
MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3

MOV   A,TIM_54
MOV   B,#0AH
DIV   AB
ADD   A,#30H
MOV   P0,A
MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3

MOV   A,B
ADD   A,#30H
MOV   P0,A
MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3

MOV   A,#':'                    ;SENT  :
MOV   P0,A
MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3

MOV   A,TIM_55
MOV   B,#0AH
DIV   AB
ADD   A,#30H
MOV   P0,A
MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3
MOV   A,B
ADD   A,#30H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV   P0,A
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                MOV   A,#"-"
                MOV   P0,A
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                LCALL DELAY_600ms
                MOV   A,TIM_54
                SUBB  A,TIM_51                  ; DIFFERENCE OF HOUR
                JC    O6                        ; H2 < H1  IMPOSSIBLE
                CJNE  A,#00H,CH_HR6             ; H2 = H1  COMPUTE DIF MIN IF
                                                ; FLASE JUMP COMPUTE DIF  HOUR
                MOV   A,TIM_55
                SUBB  A,TIM_52                  ;DIFFERENCE OF MIN
                JC    O6                        ; M2 < M1 IMPOSIBLE
                MOV   DIF_M6,A                  ; M2 >= M1 KEEP DIF MIN
                SJMP  CH_MIN61

CH_HR6:         MOV   DIF_H6,A                  ; KEPP DIF HOUR
CH_MIN61:       MOV   A,TIM_55
                SUBB  A,TIM_52                  ;DIFFERENCE OF MIN2>=MIN1
                JC    CH_MIN62
                MOV   DIF_M6,A
                SJMP  CH_HM6

CH_MIN62:       MOV   A,TIM_55
                ADD   A,#60
                SUBB  A,TIM_52                  ;DIFFERENCE OF MIN2<MIN1
                MOV   DIF_M6,A
                DEC   DIF_H6

CH_HM6          MOV   A,DIF_H6
EX_CH62:        MOV   A,DIF_M6
                SUBB  A,#01H
                JC    O6
                SJMP  EX_CH61
EX_CH63:        MOV   A,DIF_M6
                SUBB  A,#01H
                JC    OO6

EX_CH61:        INC   DIF_H6
                MOV   A,DIF_H6
                SJMP  D62
OO6:            MOV   DIF_M6,#00H
D62:            MOV   A,DIF_M6
                MOV   B,#0AH
                DIV   AB
                ADD   A,#30H
                MOV   P0,A
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                MOV   A,B
                ADD   A,#30H
                MOV   P0,A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV     P3,#0AOH                ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV     P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000

                LCALL   DELAY_600ms
                MOV     A,DIF_M6
                MOV     R0,#6                   ;INDEX channel 6
                LCALL   CAL_0

O6:             MOV     R6,#00H
                MOV     TIM_51,#00H
                MOV     TIM_52,#00H
                MOV     TIM_53,#00H
                MOV     TIM_54,#00H
                MOV     TIM_55,#00H
                MOV     TIM_56,#00H
                MOV     DIF_H6,#00H
                MOV     DIF_M6,#00H

JTM:            LJMP    MAIN_1

;*****************************************
;COMPARE DIFFERENCE TIME FOR SHOW COST
;*****************************************
CAL_0:          CJNE    A,#0,CAL_1
                LCALL   COST_1
                RET
CAL_1:          LCALL   DELAY_10ms
                CJNE    A,#1,CAL_2
                LCALL   COST_1
                RET
CAL_2:          LCALL   DELAY_10ms
                CJNE    A,#2,CAL_3
                LCALL   COST_2
                RET
CAL_3:          LCALL   DELAY_10ms
                CJNE    A,#3,CAL_4
                LCALL   COST_3
                RET
CAL_4:          LCALL   DELAY_10ms
                CJNE    A,#4,CAL_5
                LCALL   COST_4
                RET
CAL_5:          LCALL   DELAY_10ms
                CJNE    A,#5,CAL_6
                LCALL   COST_5
                RET
CAL_6:          LCALL   DELAY_10ms
                CJNE    A,#6,CAL_7
                LCALL   COST_6
                RET
CAL_7:          LCALL   DELAY_10ms
                CJNE    A,#7,CAL_8
                LCALL   COST_7
                RET
CAL_8:          LCALL   DELAY_10ms
                CJNE    A,#8,CAL_9
                LCALL   COST_8
                RET

CAL_9:          LCALL   DELAY_10ms
                CJNE    A,#9,CAL_10
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            LCALL COST_9
            RET
CAL_10:     LCALL DELAY_10ms
            CJNE  A,#10,CAL_11
            LCALL COST_10
            RET
CAL_11:     LCALL DELAY_10ms
            CJNE  A,#11,CAL_12
            LCALL COST_11
            RET
CAL_12:     LCALL DELAY_10ms
            CJNE  A,#12,CAL_13
CAL_13:     LCALL COST_12
            RET

;*********************************************************************
;SHOW COST
;*********************************************************************
COST_1:     LCALL INITIAL
            PUSH  ACC
            MOV   A,#22H                    ;SENT  "
            MOV   P0,A
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3
            MOV   A,#43H
            MOV   P0,A                      ;WRITE  C
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000

            LCALL LCD_DELAY3
            MOV   A,#48H
            MOV   P0,A                      ;WRITE  H
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,R0
            ADD   A,#30H
            MOV   P0,A                      ;WRITE  CH(NO.)
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,#22H                    ;SENT  "
            MOV   P0,A
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   LCD_DATA,#10H             ;WRITE 10
            LCALL BCD2LCD
            LCALL LCD_DELAY3

            MOV   A,#42H
            MOV   P0,A                      ;WRITE  B
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,#61H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            MOV   P0,A                    ;WRITE  a
            MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,#68H
            MOV   P0,A                    ;WRITE  h
            MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,#74H
            MOV   P0,A                    ;WRITE  t
            MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            POP   ACC
            RET

;*********************************
COST_2:     LCALL INITIAL
            PUSH  ACC
            MOV   A,#22H                  ;SENT  "
            MOV   P0,A
            MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3
            MOV   A,#43H
            MOV   P0,A                    ;WRITE  C
            MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3
            MOV   A,#48H
            MOV   P0,A                    ;WRITE  H
            MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,R0
            ADD   A,#30H
            MOV   P0,A                    ;WRITE  CH(NO.)
            MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,#22H                  ;SENT  "
            MOV   P0,A
            MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   LCD_DATA,#20H           ;WRITE 20
            LCALL BCD2LCD
            LCALL LCD_DELAY3

            MOV   A,#42H
            MOV   P0,A                    ;WRITE  B
            MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV   A,#61H
        MOV   P0,A                      ;WRITE  a
        MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   A,#68H
        MOV   P0,A                      ;WRITE  h
        MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   A,#74H
        MOV   P0,A                      ;WRITE  t
        MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        POP   ACC
        RET

;*************************************
COST_3: LCALL INITIAL
        PUSH  ACC
        MOV   A,#22H                    ;SENT  "
        MOV   P0,A
        MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3
        MOV   A,#43H
        MOV   P0,A                      ;WRITE  C
        MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000

        LCALL LCD_DELAY3
        MOV   A,#48H
        MOV   P0,A                      ;WRITE  H
        MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   A,R0
        ADD   A,#30H
        MOV   P0,A                      ;WRITE  CH(NO.)
        MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   A,#22H                    ;SENT  "
        MOV   P0,A
        MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   LCD_DATA,#30H             ;WRITE 30
        LCALL BCD2LCD
        LCALL LCD_DELAY3

        MOV   A,#42H
        MOV   P0,A                      ;WRITE  B
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                MOV   A,#61H
                MOV   P0,A                      ;WRITE  a
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                MOV   A,#68H
                MOV   P0,A                      ;WRITE  h
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                MOV   A,#74H
                MOV   P0,A                      ;WRITE  t
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                POP   ACC
                RET

;*************************************
COST_4:         LCALL INITIAL
                PUSH  ACC
                MOV   A,#22H                    ;SENT  "
                MOV   P0,A
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3
                MOV   A,#43H
                MOV   P0,A                      ;WRITE  C
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000

                LCALL LCD_DELAY3
                MOV   A,#48H
                MOV   P0,A                      ;WRITE  H
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                MOV   A,R0
                ADD   A,#30H
                MOV   P0,A                      ;WRITE  CH(NO.)
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                MOV   A,#22H                    ;SENT  "
                MOV   P0,A
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                MOV   LCD_DATA,#40H             ;WRITE 40
                LCALL BCD2LCD
                LCALL LCD_DELAY3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV   A,#42H
        MOV   P0,A                      ;WRITE  B
        MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   A,#61H
        MOV   P0,A                      ;WRITE  a
        MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   A,#68H
        MOV   P0,A                      ;WRITE  h
        MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   A,#74H
        MOV   P0,A                      ;WRITE  t
        MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        POP   ACC
        RET

;*********************************
COST_5: LCALL INITIAL
        PUSH  ACC
        MOV   A,#22H                    ;SENT  "
        MOV   P0,A
        MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3
        MOV   A,#43H
        MOV   P0,A                      ;WRITE  C
        MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000

        LCALL LCD_DELAY3
        MOV   A,#48H
        MOV   P0,A                      ;WRITE  H
        MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   A,R0
        ADD   A,#30H
        MOV   P0,A                      ;WRITE  CH(NO.)
        MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   A,#22H                    ;SENT  "
        MOV   P0,A
        MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV   LCD_DATA,#50H          ;WRITE 50
        LCALL BCD2LCD
        LCALL LCD_DELAY3

        MOV   A,#42H
        MOV   P0,A                   ;WRITE  B
        MOV   P3,#0A0H               ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   A,#61H
        MOV   P0,A                   ;WRITE  a
        MOV   P3,#0A0H               ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   A,#68H
        MOV   P0,A                   ;WRITE  h
        MOV   P3,#0A0H               ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   A,#74H
        MOV   P0,A                   ;WRITE  t
        MOV   P3,#0A0H               ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        POP   ACC
        RET

;*********************************
COST_6: LCALL INITIAL
        PUSH  ACC
        MOV   A,#22H                 ;SENT  "
        MOV   P0,A
        MOV   P3,#0A0H               ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3
        MOV   A,#43H
        MOV   P0,A                   ;WRITE  C
        MOV   P3,#0A0H               ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3
        MOV   A,#48H
        MOV   P0,A                   ;WRITE  H
        MOV   P3,#0A0H               ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   A,R0
        ADD   A,#30H
        MOV   P0,A                   ;WRITE  CH(NO.)
        MOV   P3,#0A0H               ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   A,#22H                 ;SENT  "
        MOV   P0,A
        MOV   P3,#0A0H               ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                ;EN,RW,RS,0,0,0,0,0==>00100000
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        LCALL LCD_DELAY3

        MOV   LCD_DATA,#60H          ;WRITE 60
        LCALL BCD2LCD
        LCALL LCD_DELAY3

        MOV   A,#42H
        MOV   P0,A                   ;WRITE  B
        MOV   P3,#0A0H               ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   A,#61H
        MOV   P0,A                   ;WRITE  a
        MOV   P3,#0A0H               ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                ;EN,RW,RS,0,0,0,0,0==>00100000
              LCALL LCD_DELAY3

        MOV   A,#68H
        MOV   P0,A                   ;WRITE  h
        MOV   P3,#0A0H               ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   A,#74H
        MOV   P0,A                   ;WRITE  t
        MOV   P3,#0A0H               ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        POP   ACC
        RET

;*********************************
COST_7: LCALL INITIAL
        PUSH  ACC
        MOV   A,#22H                 ;SENT  "
        MOV   P0,A
        MOV   P3,#0A0H               ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3
        MOV   A,#43H
        MOV   P0,A                   ;WRITE  C
        MOV   P3,#0A0H               ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                ;EN,RW,RS,0,0,0,0,0==>00100000

        LCALL LCD_DELAY3
        MOV   A,#48H
        MOV   P0,A                   ;WRITE  H
        MOV   P3,#0A0H               ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   A,R0
        ADD   A,#30H
        MOV   P0,A                   ;WRITE  CH(NO.)
        MOV   P3,#0A0H               ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   A,#22H                 ;SENT  "
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV   P0,A
        MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   LCD_DATA,#70H             ;WRITE 70
        LCALL BCD2LCD
        LCALL LCD_DELAY3

        MOV   A,#42H
        MOV   P0,A                      ;WRITE  B
        MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   A,#61H
        MOV   P0,A                      ;WRITE  a
        MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   A,#68H
        MOV   P0,A                      ;WRITE  h
        MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   A,#74H
        MOV   P0,A                      ;WRITE  t
        MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        POP   ACC
        RET

;************************************
COST_8: LCALL INITIAL
        PUSH  ACC
        MOV   A,#22H                    ;SENT "
        MOV   P0,A
        MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3
        MOV   A,#43H
        MOV   P0,A                      ;WRITE  C
        MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000

        LCALL LCD_DELAY3
        MOV   A,#48H
        MOV   P0,A                      ;WRITE  H
        MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        MOV   A,R0
        ADD   A,#30H
        MOV   P0,A                      ;WRITE  CH(NO.)
        MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                MOV   A,#22H                    ;SENT  "
                MOV   P0,A
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                MOV   LCD_DATA,#80H             ;WRITE 80
                LCALL BCD2LCD
                LCALL LCD_DELAY3

                MOV   A,#42H
                MOV   P0,A                      ;WRITE  B
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                MOV   A,#61H
                MOV   P0,A                      ;WRITE  a
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                MOV   A,#68H
                MOV   P0,A                      ;WRITE  h
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                MOV   A,#74H
                MOV   P0,A                      ;WRITE  t
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                POP   ACC
                RET

;*********************************
COST_9:         LCALL INITIAL
                PUSH  ACC
                MOV   A,#22H                    ;SENT  "
                MOV   P0,A
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3
                MOV   A,#43H
                MOV   P0,A                      ;WRITE  C
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3
                MOV   A,#48H
                MOV   P0,A                      ;WRITE  H
                MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
                LCALL LCD_DELAY3

                MOV   A,R0
                ADD   A,#30H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            MOV   P0,A                      ;WRITE  CH(NO.)
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,#22H                    ;SENT  "
            MOV   P0,A
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   LCD_DATA,#90H             ;WRITE 90
            LCALL BCD2LCD
            LCALL LCD_DELAY3

            MOV   A,#42H
            MOV   P0,A                      ;WRITE  B
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,#61H
            MOV   P0,A                      ;WRITE  a
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,#68H
            MOV   P0,A                      ;WRITE  h
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,#74H
            MOV   P0,A                      ;WRITE  t
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            POP   ACC
            RET

;************************************
COST_10:    LCALL INITIAL
            PUSH  ACC
            MOV   A,#22H                    ;SENT  "
            MOV   P0,A
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3
            MOV   A,#43H
            MOV   P0,A                      ;WRITE  C
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000

            LCALL LCD_DELAY3
            MOV   A,#48H
            MOV   P0,A                      ;WRITE  H
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            MOV   A,R0
            ADD   A,#30H
            MOV   P0,A                    ;WRITE  CH(NO.)
            MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,#22H                  ;SENT  "
            MOV   P0,A
            MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   LCD_DATA,#10H           ;WRITE 10
            LCALL BCD2LCD
            LCALL LCD_DELAY3

            MOV   A,#30H
            MOV   P0,A                    ;WRITE  0
            MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,#42H
            MOV   P0,A                    ;WRITE  B
            MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,#61H
            MOV   P0,A                    ;WRITE  a
            MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,#68H
            MOV   P0,A                    ;WRITE  h
            MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,#74H
            MOV   P0,A                    ;WRITE  t
            MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            POP   ACC
            RET

;*****************************************
COST_11:    LCALL INITIAL
            PUSH  ACC
            MOV   A,#22H                  ;SENT  "
            MOV   P0,A
            MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3
            MOV   A,#43H
            MOV   P0,A                    ;WRITE  C
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000

LCALL LCD_DELAY3
MOV   A,#48H
MOV   P0,A                    ;WRITE  H
MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3

MOV   A,R0
ADD   A,#30H
MOV   P0,A                    ;WRITE  CH(NO.)
MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3

MOV   A,#22H                  ;SENT  "
MOV   P0,A
MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3

MOV   LCD_DATA,#10H           ;WRITE 11
LCALL BCD2LCD
LCALL LCD_DELAY3

MOV   A,#30H
MOV   P0,A                    ;WRITE  0
MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3

MOV   A,#42H
MOV   P0,A                    ;WRITE  B
MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3

MOV   A,#61H
MOV   P0,A                    ;WRITE  a
MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3

MOV   A,#68H
MOV   P0,A                    ;WRITE  h
MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3

MOV   A,#74H
MOV   P0,A                    ;WRITE  t
MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
LCALL LCD_DELAY3

POP   ACC
RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;*******************************************
COST_12:    LCALL INITIAL
            PUSH  ACC
            MOV   A,#22H                    ;SENT  "
            MOV   P0,A
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3
            MOV   A,#43H
            MOV   P0,A                      ;WRITE  C
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000

            LCALL LCD_DELAY3
            MOV   A,#48H
            MOV   P0,A                      ;WRITE  H
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,R0
            ADD   A,#30H
            MOV   P0,A                       ;WRITE  CH(NO.)
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,#22H                    ;SENT  "
            MOV   P0,A
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   LCD_DATA,#10H             ;WRITE 12
            LCALL BCD2LCD
            LCALL LCD_DELAY3

            MOV   A,#30H
            MOV   P0,A                      ;WRITE  0
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3


            MOV   A,#42H
            MOV   P0,A                      ;WRITE  B
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,#61H
            MOV   P0,A                      ;WRITE  a
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3

            MOV   A,#68H
            MOV   P0,A                      ;WRITE  h
            MOV   P3,#0A0H                  ;EN,RW,RS,0,0,0,0,0==>10100000
            MOV   P3,#20H                   ;EN,RW,RS,0,0,0,0,0==>00100000
            LCALL LCD_DELAY3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV   A,#74H
        MOV   P0,A                    ;WRITE  t
        MOV   P3,#0A0H                ;EN,RW,RS,0,0,0,0,0==>10100000
        MOV   P3,#20H                 ;EN,RW,RS,0,0,0,0,0==>00100000
        LCALL LCD_DELAY3

        POP   ACC
        RET

;***************************************
;I2C RTC WRITE
;***************************************
RTC_WR:         MOV     I2C_ADDR,#RTC_ID    ;11010000          ;***
                LCALL   I2C_SLAVE           ;CONNECT  SLAVE
                MOV     I2C_DATA,#00H       ;SET  ADDR
                LCALL   I2C_DATA_WR
                MOV     SECOND,#40H
                MOV     I2C_DATA,SECOND     ;WR ss
                LCALL   I2C_DATA_WR

                MOV     I2C_ADDR,#RTC_ID    ;11010000          ;***
                LCALL   I2C_SLAVE           ;CONNECT  SLAVE
                MOV     I2C_DATA,#01H       ;SET  ADDR
                LCALL   I2C_DATA_WR
                MOV     MINITE,#57H
                MOV     I2C_DATA,MINITE     ;WR mm
                LCALL   I2C_DATA_WR

                MOV     I2C_ADDR,#RTC_ID    ;11010000          ;***
                LCALL   I2C_SLAVE           ;CONNECT  SLAVE
                MOV     I2C_DATA,#02H       ;SET  ADDR
                LCALL   I2C_DATA_WR
                MOV     HOUR,#23H
                MOV     I2C_DATA,HOUR       ;WR hh
                LCALL   I2C_DATA_WR

                MOV     I2C_ADDR,#RTC_ID    ;11010000          ;***
                LCALL   I2C_SLAVE           ;CONNECT  SLAVE
                MOV     I2C_DATA,#03H       ;SET  ADDR
                LCALL   I2C_DATA_WR
                MOV     DAY,#04H
                MOV     I2C_DATA,DAY        ;WR mon-sun
                LCALL   I2C_DATA_WR

                MOV     I2C_ADDR,#RTC_ID    ;11010000          ;***
                LCALL   I2C_SLAVE           ;CONNECT  SLAVE
                MOV     I2C_DATA,#04H       ;SET  ADDR
                LCALL   I2C_DATA_WR
                MOV     DATE,#29H
                MOV     I2C_DATA,DATE       ;WR DD
                LCALL   I2C_DATA_WR

                MOV     I2C_ADDR,#RTC_ID    ;11010000          ;***
                LCALL   I2C_SLAVE           ;CONNECT  SLAVE
                MOV     I2C_DATA,#05H       ;SET  ADDR
                LCALL   I2C_DATA_WR
                MOV     MONTH,02H
                MOV     I2C_DATA,MONTH      ;WR MM
                LCALL   I2C_DATA_WR
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV         I2C_ADDR,#RTC_ID    ;11010000       ;***
                LCALL       I2C_SLAVE           ;CONNECT  SLAVE
                MOV         I2C_DATA,#06H       ;SET  ADDR
                LCALL       I2C_DATA_WR
                MOV         YEAR,#47H
                MOV         I2C_DATA,YEAR       ;WR YY
                LCALL       I2C_DATA_WR

                MOV         I2C_ADDR,#RTC_ID    ;11010000       ;***
                LCALL       I2C_SLAVE           ;CONNECT  SLAVE
                MOV         I2C_DATA,#07H       ;SET  ADDR
                LCALL       I2C_DATA_WR
                MOV         CONTROL,#03H        ;00000011
                MOV         I2C_DATA,CONTROL    ;WR CON
                LCALL       I2C_DATA_WR

                LCALL       I2C_STOP            ;STOP CONDITION
                RET

;*************************************
;I2C RTC READ
;*************************************
RTC_RD:         MOV         I2C_ADDR,#RTC_ID
                LCALL       I2C_SLAVE
                MOV         I2C_DATA,00H
                LCALL       I2C_DATA_WR

                MOV         I2C_ADDR,#RTC_ID+1
                LCALL       I2C_SLAVE
                LCALL       I2C_DATA_RD
                MOV         SECOND,I2C_DATA
                LCALL       I2C_ACK_BIT

                LCALL       I2C_DATA_RD
                MOV         MINITE,I2C_DATA
                LCALL       I2C_ACK_BIT

                LCALL       I2C_DATA_RD
                MOV         HOUR,I2C_DATA
                LCALL       I2C_ACK_BIT

                LCALL       I2C_DATA_RD
                MOV         DAY,I2C_DATA
                LCALL       I2C_ACK_BIT

                LCALL       I2C_DATA_RD
                MOV         DATE,I2C_DATA
                LCALL       I2C_ACK_BIT

                LCALL       I2C_DATA_RD
                MOV         MONTH,I2C_DATA
                LCALL       I2C_ACK_BIT

                LCALL       I2C_DATA_RD
                MOV         YEAR,I2C_DATA
                LCALL       I2C_ACK_BIT

                LCALL       I2C_DATA_RD
                MOV         CONTROL,I2C_DATA
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                LCALL       I2C_NACK_BIT
                LCALL       I2C_STOP
                RET

;*************************************
;               BCD2LCD
;*************************************
BCD2LCD:        PUSH        ACC
                PUSH        B
                MOV         A,LCD_DATA
                MOV         B,A
                LCALL       LCD_DELAY
                ANL         A,#0F0H
                LCALL       LCD_DELAY
                SWAP        A
                LCALL       LCD_DELAY
                ADD         A,#030H
                MOV         LCD_DATA,A
                MOV         P0,LCD_DATA     ;WRITE  DATA
                MOV         P3,#0A0H        ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV         P3,#20H         ;EN,RW,RS,0,0,0,0,0==>00100000

                LCALL       LCD_DELAY
                MOV         A,B
                LCALL       LCD_DELAY
                ANL         A,#0FH
                LCALL       LCD_DELAY
                ADD         A,#30H
                LCALL       LCD_DELAY
                MOV         LCD_DATA,A
                MOV         P0,LCD_DATA     ;WRITE  DATA
                MOV         P3,#0A0H        ;EN,RW,RS,0,0,0,0,0==>10100000
                MOV         P3,#20H         ;EN,RW,RS,0,0,0,0,0==>00100000

                POP         B
                POP         ACC
                RET

;*************************************
;I2C SLAVE CONNECT
;I/P:  I2C_ADDR
;O/P FLAG: I2C_ACK
;RESERVE: R5
;*************************************
I2C_SLAVE:      PUSH        ACC
                SETB        I2C_ACK
                MOV         A,I2C_ADDR
                LCALL       I2C_START
                MOV         R5,#008

I2C_SLAVE_1:    RLC         A
                MOV         SDA,C
                LCALL       I2C_CLK
                DJNZ        R5,I2C_SLAVE_1

                SETB        SDA
                LCALL       I2C_DELAY
                SETB        SCL
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    LCALL       I2C_DELAY
                    JB          SDA,I2C_SLAVE_2
                    CLR         I2C_ACK
I2C_SLAVE_2:        CLR         SCL
                    POP         ACC
                    RET

;*****************************************
;I2C DATA WRITE
;I/P:  I2C_DATA
;RESERVE :  R5
;*****************************************
I2C_DATA_WR:        PUSH        ACC
                    SETB        I2C_ACK
                    MOV         A,I2C_DATA
                    MOV         R5,#008
I2C_DATA_WR_1:      RLC         A
                    MOV         SDA,C
                    LCALL       I2C_CLK
                    DJNZ        R5,I2C_DATA_WR_1

                    SETB        SDA
                    LCALL       I2C_DELAY
                    SETB        SCL
                    LCALL       I2C_DELAY
                    JB          SDA,I2C_DATA_WR_2
                    CLR         I2C_ACK
I2C_DATA_WR_2:      CLR         SCL
                    POP         ACC
                    RET


;*****************************************
;I2C DATA READ
;O/P: I2C_DATA
;RESERVE  R5
;*****************************************
I2C_DATA_RD:        PUSH        ACC
                    CLR         A
                    MOV         R5,#008
I2C_DATA_RD_1:      LCALL       I2C_DELAY
                    SETB        SCL
                    LCALL       I2C_DELAY
                    MOV         C,SDA
                    RLC         A

                    CLR         SCL
                    DJNZ        R5,I2C_DATA_RD_1
                    MOV         I2C_DATA,A
                    POP         ACC
                    RET

;*****************************************
;I2C START CONDITION
;*****************************************

I2C_START:          JNB         SCL,I2C_START_1
                    CLR         SCL
I2C_START_1:        SETB        SDA
                    SETB        SCL
                    LCALL       I2C_DELAY
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                CLR         SDA
                LCALL       I2C_DELAY
                CLR         SCL
                RET

;****************************************
;I2C STOP CONDITION
;****************************************

I2C_STOP:       JNB         SCL,I2C_STOP_1
                CLR         SCL
I2C_STOP_1:     CLR         SDA
                LCALL       I2C_DELAY
                SETB        SCL
                LCALL       I2C_DELAY
                SETB        SDA
                RET

;****************************************
;I2C ACKNOWLEDGE
;****************************************

I2C_ACK_BIT:    CLR         SDA
                LCALL       I2C_DELAY
                LCALL       I2C_CLK
                SETB        SDA
                RET

;****************************************
;I2C NOT ACK
;****************************************

I2C_NACK_BIT:   SETB        SDA
                LCALL       I2C_DELAY
                LCALL       I2C_CLK
                SETB        SCL
                RET


;****************************************
;INITIAL LCD
;****************************************

INITIAL:        MOV         P3,#00H
                MOV         P0,#00111000B       ;1.FUNCTION  SET
                LCALL       PULSE_EN
                LCALL       LCD_DELAY

                MOV         P3,#00H
                MOV         P0,#00001110B       ;2.DISPLAY  CONTROL
                LCALL       PULSE_EN
                LCALL       LCD_DELAY

                MOV         P3,#00H
                MOV         P0,#01H             ;3.CLEAR
                LCALL       PULSE_EN
                LCALL       LCD_DELAY

                MOV         P3,#00H
                MOV         P0,#00000110B       ;4.ENTRY MODE SET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                LCALL       PULSE_EN
                LCALL       LCD_DELAY

                MOV         P3,#00H
                MOV         P0,#00010100B       ;5.display shift
                LCALL       PULSE_EN
                LCALL       LCD_DELAY

                RET

;***************************************
;PULSE
;***************************************

PULSE_EN:       MOV         P3,#10000000B
                MOV         R7,#200
EN_1:           NOP
                DJNZ        R7,EN_1
                MOV         P3,#00000000B       ;EN,RW,RS,0,0,0,0,0
                MOV         R7,#200
EN_2:           NOP
                DJNZ        R7,EN_2
                RET

;***************************************
;I2C CLOCK
;***************************************

I2C_CLK:        LCALL       I2C_DELAY
                SETB        SCL
                LCALL       I2C_DELAY
                CLR         SCL
                RET

;***************************************
;DELAY
;***************************************

I2C_DELAY:      MOV         R6,#00CH
I2C_DELAY_1:    NOP
                NOP
                DJNZ        R6,I2C_DELAY_1
                RET

RTC_DELAY:      MOV         R7,#48              ;DELAY 72ns
RTC_DELAY_1:    MOV         R6,03H
RTC_DELAY_2:    MOV         R5,00H
RTC_DELAY_3:    DJNZ        R5,RTC_DELAY_3
                DJNZ        R6,RTC_DELAY_2
                DJNZ        R7,RTC_DELAY_1
                RET

RTCDL:          MOV         R5,#4               ;DELAY
                DJNZ        R5,$
                RET
DELAY_50us:     MOV         R7,02H
DELAY_50us_1:   MOV         R6,03H
DELAY_50us_2:   MOV         R5,02H
DELAY_50us_3:   DJNZ        R5,DELAY_50us_3
                DJNZ        R6,DELAY_50us_2
                DJNZ        R7,DELAY_50us_1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                RET

LCD_DELAY:      MOV         R7,#002
LCD_DELAY_1:    MOV         R6,#0E6H
LCD_DELAY_2:    NOP
                NOP
                DJNZ        R6,LCD_DELAY_2
                DJNZ        R7,LCD_DELAY_1
                RET

LCD_DELAY2:     LCALL       LCD_DELAY
                LCALL       LCD_DELAY
                LCALL       LCD_DELAY
                LCALL       LCD_DELAY
                LCALL       LCD_DELAY
                LCALL       LCD_DELAY
                LCALL       LCD_DELAY
                LCALL       LCD_DELAY
                LCALL       LCD_DELAY
                LCALL       LCD_DELAY
                RET

LCD_DELAY3:     LCALL       LCD_DELAY2
                LCALL       LCD_DELAY2
                LCALL       LCD_DELAY2
                RET

DELAY_50ms:     LCALL       DELAY_10ms
                LCALL       DELAY_10ms
                LCALL       DELAY_10ms
                LCALL       DELAY_10ms
                LCALL       DELAY_10ms
                RET

DELAY_10ms:     MOV         R7,#010             ; Do 10 times
DELAY_10ms_1:   MOV         R6,#0E6H            ; Each loop = 1 ms
DELAY_10ms_2:   NOP
                NOP
                DJNZ        R6,DELAY_10ms_2
                DJNZ        R7,DELAY_10ms_1
                RET

DELAY_100ms:    MOV         R7,#100             ; Do 100 times
DELAY_100ms_1:  MOV         R6,#0E6H            ; Each loop = 1 ms
DELAY_100ms_2:  NOP
                NOP
                DJNZ        R6,DELAY_100ms_2
                DJNZ        R7,DELAY_100ms_1
                RET

DELAY_1s:       MOV         R5,#100             ; Do 100 times
DELAY_1s_1:     LCALL       DELAY_10ms
                DJNZ        R5,DELAY_1s_1
                RET

DELAY_30ms:     LCALL       DELAY_10ms
                LCALL       DELAY_10ms
                LCALL       DELAY_10ms
                RET

DELAY_1ms:      MOV         R6,#0E6H            ; Each loop = 1 ms
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
DELAY_1ms_1:        NOP
                    NOP
                    DJNZ        R6,DELAY_1ms_1
                    RET

DELAY_600ms:        MOV    R7,#06H
DELAY_600ms_1:      LCALL DELAY_100ms
                    DJNZ   R7, DELAY_600ms_1
                    RET

                    END
;***************************************
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3 คำอธิบายโปแกรมย่อยของการใช้งานโมดูลแอลซีดี

- **INIT_LCD** เป็นโปรแกรมย่อยสำหรับเตรียมแอลซีดี ให้พร้อมในการทำงาน โดยจะเรียกเพียงครั้งแรกครั้งเดียว หลังจากเริ่มจ่ายไฟให้กับตัวแอลซีดี
- **LCD_CLR** เป็นโปรแกรมย่อยสำหรับเคลียร์ข้อความใน ดีดีแรม ทั้งหมด
- **LCD_HOME** เป็นโปรแกรมย่อยสำหรับเคลียร์ค่าแอดแดรส ดีดีแรม เป็น 00H (ตำแหน่งของเคอร์เซอร์จะอยู่ด้านซ้ายบนสุด)
- **LCD_OFF** เป็นโปรแกรมย่อยสำหรับดับการแสดงหน้าจอแสดงผลบนแอลซีดี โดยที่ข้อมูลยังคงอยู่
- **LCD_CLK** เป็นโปรแกรมย่อยสำหรับป้อนสัญญาณในการส่งข้อมูลไปยังแอลซีดี
- **LCD_ON** เป็นโปรแกรมย่อยสำหรับแสดงข้อมูลหน้าจอแสดงผลบนแอลซีดี
- **SET_ADDR_LCD** เป็นโปรแกรมย่อยสำหรับตั้งค่าแอดแดรส ดีดีแรม(ตำแหน่งของเคอร์เซอร์)โดยรับค่าจากตัวแปร LCD_ADDR
- **WRCHAR_LCD** เป็นโปรแกรมย่อยสำหรับเขียนค่าตัวแปร LCD_DATA ไปยังแอลซีดีในตำแหน่งที่เคอร์เซอร์อยู่

3.2.4 คำอธิบายโปรแกรมย่อยของการติดต่อบนระบบบัส I^2C

- **I^2C_DATA_WR** เป็นโปรแกรมย่อยสำหรับเขียนข้อมูลไปยังบัส I^2C โดยจะรับข้อมูลจากรีจิสเตอร์ I^2C_DATA พร้อมกับตรวจสอบการตอบรับของอุปกรณ์ที่ติดต่อด้วย โดยบิต I^2C_ACK จะเป็น "1" เมื่อไม่มีการตอบรับจากอุปกรณ์และเป็น "0 " เมื่อมีการตอบรับ
- **I^2C_DATA_RD** เป็นโปรแกรมย่อยสำหรับอ่านข้อมูล โดยเก็บข้อมูลที่อ่านได้ในรีจิสเตอร์ I^2C_DATA
- **I^2C_SLAVE** เป็นโปรแกรมย่อยสำหรับส่งค่าแอดแดรสของอุปกรณ์ที่ต้องการติดต่อด้วยไปยังบัส I^2C โดยจะรับค่าแอดเดรสจากรีจิสเตอร์ I^2C_ADDR พร้อมกับตรวจสอบการตอบรับของอุปกรณ์ที่ต่อด้วยโดยบิต I^2C_ACK เป็น "1" เมื่อไม่มีการตอบรับจากอุปกรณ์และเป็น "0" เมื่อมีการตอบรับจากอุปกรณ์
- **I^2C_START** เป็นโปรแกรมย่อยสำหรับสร้างสภาวะเริ่มต้นในการส่งข้อมูลบนบัส I^2C
- **I^2C_STOP** เป็นโปรแกรมย่อยสำหรับสร้างสภาวะสิ้นสุดในการส่งข้อมูลบนบัส I^2C
- **I^2C_CLK** เป็นโปรแกรมย่อยสำหรับสร้างสร้างสัญญาณนาฬิกาของระบบบัส I^2C
- **I^2C_ACK_BIT** เป็นโปรแกรมย่อยสำหรับแจ้งให้อุปกรณ์ที่ติดต่อด้วยให้ส่งข้อมูลชุดต่อไปได้ โดยจะเรียกใช้โปรแกรมย่อยนี้ต่อจากโปรแกรมย่อย I^2C_DATA_RD แล้วเรียกใช้โปรแกรมย่อย I^2C_DATA_RD ต่อหลังอีกเพื่อรับหรืออ่านข้อมูลชุดต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **I^2C_NACK_BIT** เป็นโปรแกรมย่อยสำหรับบอกอุปกรณ์ ที่ติดต่อด้วยให้หยุดส่งข้อมูล จากนั้นจึงเรียกโปรแกรมย่อย I^2C_STOP ได้ โดยเรียกใช้โปรแกรมย่อยนี้ต่อจากโปรแกรมย่อย I^2C_DATA_RD
- **BCD2LCD** เป็นโปรแกรมย่อยสำหรับที่ใช้แสดงค่ารีจิสเตอร์ LCD_DATA ที่มีข้อมูลแบบบีซีดี ให้มาแสดงผลบนแอลซีดี
- **RTC_RD** เป็นโปรแกรมย่อยสำหรับ ใช้อ่านค่ารีจิสเตอร์จากไอซีเรียลไทม์คล็อก โดยจะทำการอ่านได้ค่าเรียงกันดังนี้ SECONDS,MINUTES,HOURS, DAY,DATE,MONTH,YEAR และ CONTROL ซึ่งสามารถดูรายละเอียดของข้อมูลแต่ละตัวได้ใน ดาต้าชีท ของ ไอซี DS1307 ในโปรแกรมย่อยนี้จะทำการติดต่อกับไอซีเรียลไทม์คล็อก เพื่อทำงานในโหมดเขียนค่าก่อน (เรียก I^2C_SLAVE โดยใช้ ID 11010000B) โดยจะเขียนค่า 00H ลงไป (เรียก I^2C_DATA_WR) ซึ่งเป็นการเขียนค่า เวิร์ดแอดเดรส ภายในตัวไอซีเรียลไทม์คล็อกให้ชี้ที่ตำแหน่งแอดเดรส 00H ซึ่งเป็นแอดเดรสของค่าวินาที จากนั้นให้เรียกโปรแกรมย่อยเพื่อติดต่อกับไอซีเรียลไทม์คล็อกเพื่อทำงานในโหมดการอ่านค่า(เรียก I^2C_SLAVE โดยใช้ID 11010001B)และเรียกโปรแกรมย่อยที่ใช้ในการอ่านค่าออกมา(เรียก I^2C_DATA_RD)เมื่อทำการอ่านค่าเรียบร้อยแล้วเวิร์ดแอดเดรสภายในไอซีเรียลไทม์คล็อกก็จะเพิ่มขึ้นอีก1โดยอัตโนมัติต่อจากนั้นให้ส่งคำสั่งหยุดการรับข้อมูลตัวถัดไป(I^2C_NACK)จึงทำการเก็บค่า I^2C_DATAที่อ่านได้ลงในรีจิสเตอร์SECONDSและให้ทำการอ่านค่าเช่นนี้ไปจนครบ8ครั้ง โดยทำการเก็บค่าตัวถัดมาก็จะได้ค่า MINUTES, HOURS, DAY, DATE, MONTH, YEAR และ CONTROL ตามลำดับ เมื่อทำการอ่านค่าครบแล้ว จึงจะทำการส่งสถานะหยุดการติดต่อ(เรียก I^2C_STOP)แล้วกลับเข้าสู่โปรแกรมหลักต่อไป
- **RTC_WR** เป็นโปรแกรมย่อยสำหรับเขียนค่ารีจิสเตอร์ไปยังไอซีเรียลไทม์คล็อก โดยจะทำการเขียนค่าเรียงกันดังนี้ SECONDS, MINUTES, HOURS, DAY, DATE, MONTH, YEAR, และ CONTROL ในโปรแกรมย่อยนี้จะทำการติดต่อกับไอซีเรียลไทม์คล็อก เพื่อทำงานในโหมดการเขียนค่า(เรียก I^2C_SLAVE โดยใช้ID11010000B) โดยจะเขียนค่า00H ลงไป (เรียก I^2C_DATA_WR) ซึ่งเป็นการเขียนค่าเวิร์ดแอดเดรสภายในตัวไอซีเรียลไทม์คล็อกให้ชี้ที่ตำแหน่งแอดเดรส 00Hซึ่งเป็นแอดเดรสของค่าวินาที จากนั้นจึงทำการเขียนค่า SECONDS, MINUTES, HOURS, DAY, DATE, MONTH, YEAR, และ CONTROL ต่อเนื่องกันตามลำดับ เมื่อทำการเขียนค่าครบแล้ว จึงทำการส่งสถานะหยุดการติดต่อ (เรียก I^2C_STOP)แล้วกลับเข้าสู่โปรแกรมหลักต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

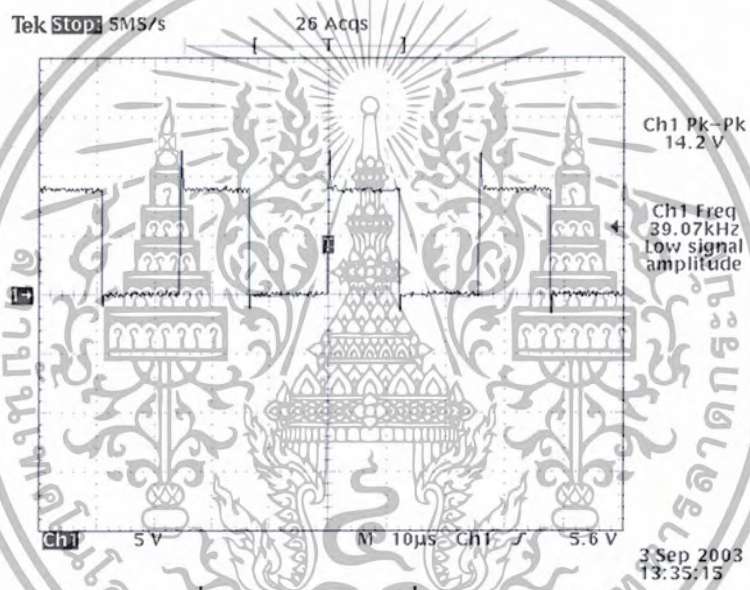
การทดลองและผลการทดลอง

4.1 ขั้นตอนการทดลอง

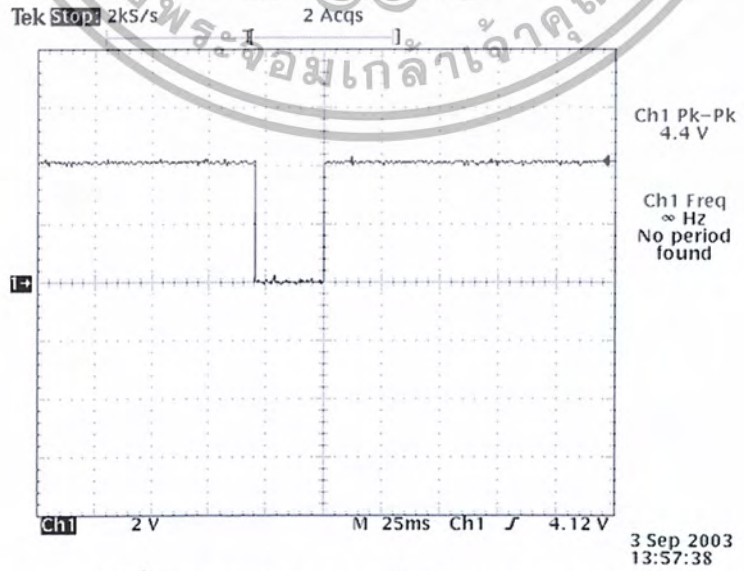
ในโครงงานนี้แบ่งเป็น 2 ส่วน คือส่วนของฮาร์ดแวร์และ ซอฟท์แวร์ โดยทำการทดลองในส่วนของ ฮาร์ดแวร์ ตามวงจรในบทที่ 3 ในส่วนของซอฟท์แวร์นั้น จะได้ทำการทดลองเขียนโปรแกรมตามโฟว์ชาร์ทที่แสดงแล้วในบทที่ 3

4.2 ผลการทดลอง

4.2.1 ผลการทดลองของวงจรเซนเซอร์ทั้งด้านส่งและด้านรับ ดังรูปที่ 4.1 และ 4.2 โดยวัดสัญญาณทางด้านส่งที่ขา 3 ของ ไอซี 555 และทางด้านรับที่ ขา 1 ของโฟโต้ไดโอด เบอร์ 4838



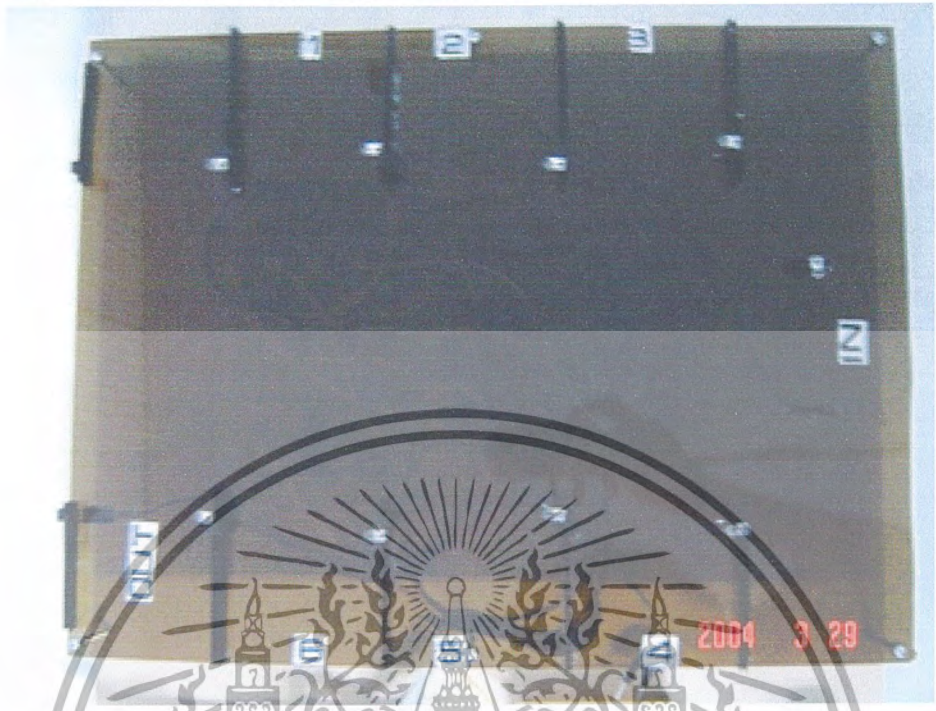
รูปที่ 4.1 แสดงสัญญาณที่ขา 3 ของไอซี 555



รูปที่ 4.2 แสดงสัญญาณทางด้านรับของวงจรเซนเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.2 ผลการทดลองในส่วนของซอฟท์แวร์ ซึ่งจะเป็นผลการทดลองที่แสดงผลทางแอลอีดีและจอแอลซีดี ในกรณีที่จอดรถว่างแอลอีดีก็จะแสดงไฟสีเขียวทั้งหมด ดังรูป

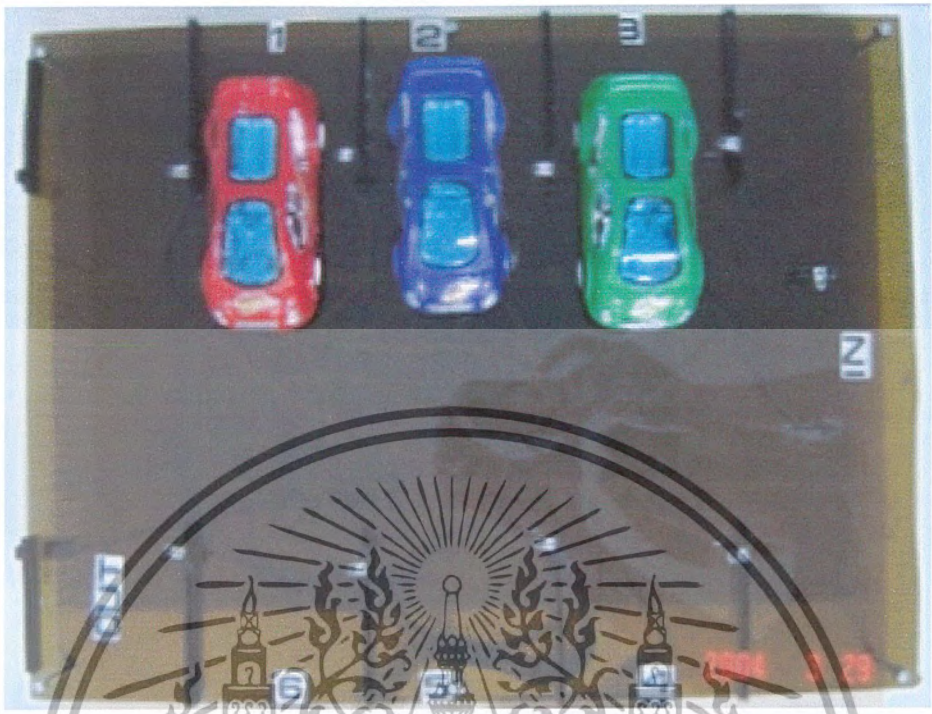


รูปที่ 4.3 แสดงที่จอดรถขณะยังไม่มีรถจอด

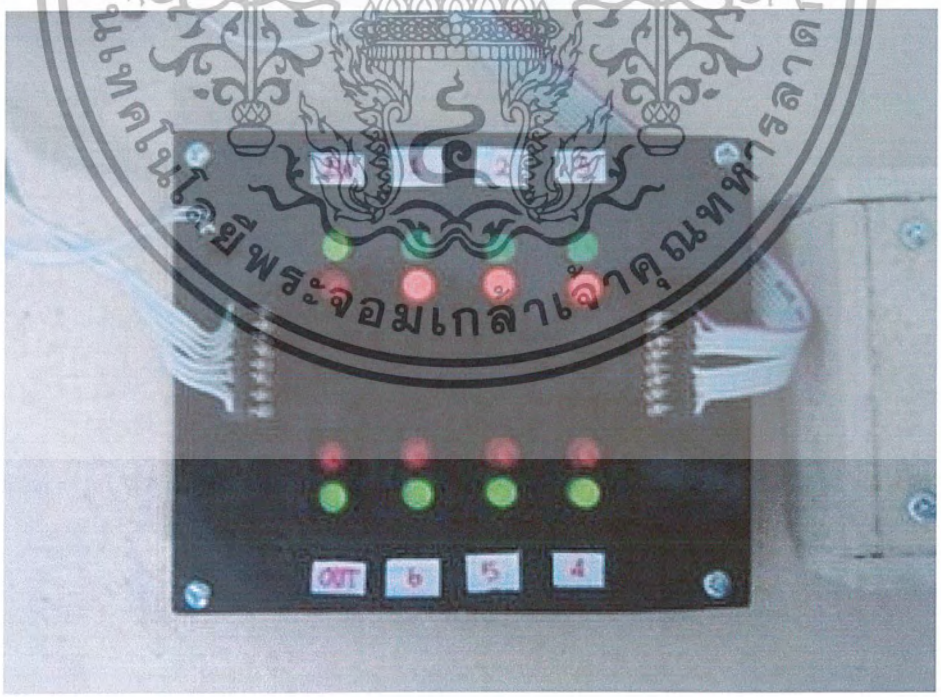
รูปที่4.4 แสดงสถานะที่จอดรถว่างทางแอลอีดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในกรณีที่มีรถมาจอดที่ช่องที่ 1,2 และ 3 แอลอีดีจะแสดงไฟสีแดงช่องจอดรถที่ 1,2 และ 3 ส่วนช่องจอดรถที่เหลือจะแสดงไฟสีเขียวทั้งหมด ดังรูป



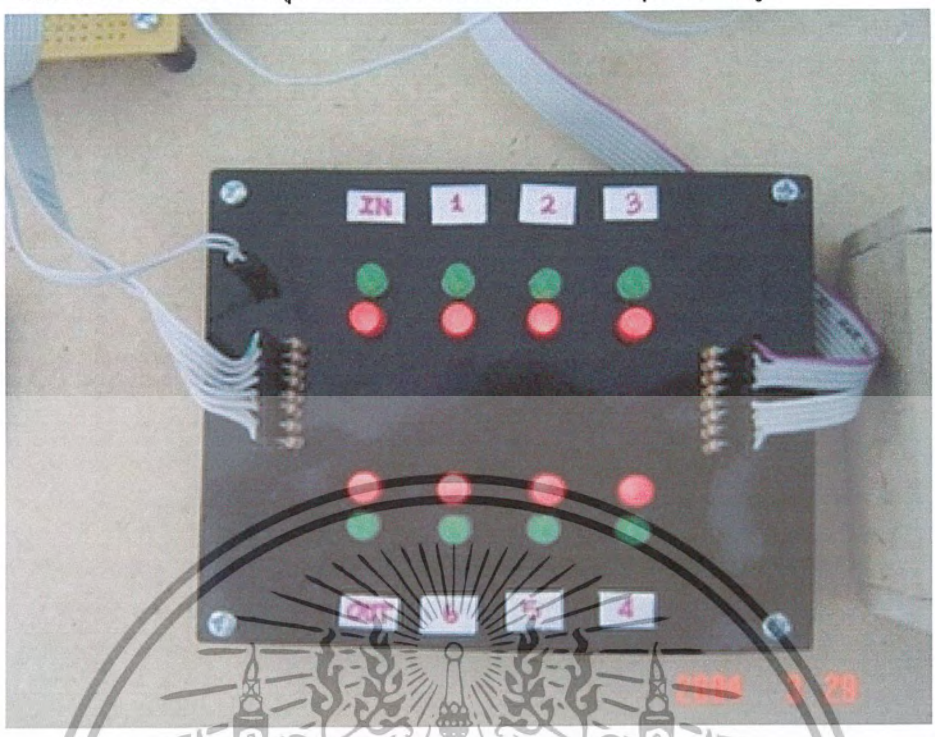
รูปที่4.5 แสดงที่จอดรถขณะที่มีรถเข้ามาจอดช่องที่ 1,2 และ 3



รูปที่4.6 แสดงสถานะการจอดรถทางแอลอีดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในกรณีที่มีรถมาจอดครบทุกช่องแอลอีดีก็จะแสดงไฟสีแดงทุกช่อง ดังรูป

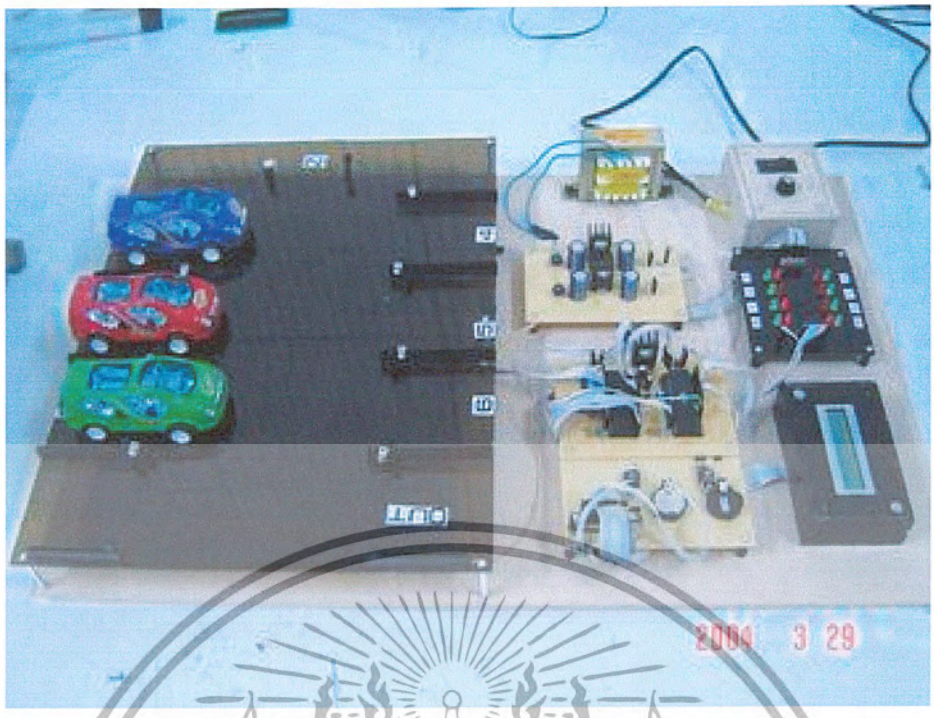


รูปที่4.7 แสดงสถานะที่จอดรถเต็มทางแอลอีดี

เมื่อรถออกจากที่จอดรถจะมีการแสดงราคาการจอดรถทางจอแอลซีดี โดยในที่นี้เราคิดค่าจอดรถชั่วโมงละ10บาท และเราให้รถจอดนาน12 ชั่วโมง คิดค่าบริการการจอดรถได้ 120 บาท และบอกตำแหน่งที่จอดรถคือช่องที่4 ดังรูป

รูปที่4.8 การแสดงตำแหน่งและราคาค่าจอดรถออกทางจอแอลซีดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่4.9 วงจรรวมของระบบควบคุมการจอดรถอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทวิจารณ์ และ บทสรุป

5.1 สรุปผลการทดลอง

โครงงานนี้จำลองระบบการจอดรถแบบห้างสรรพสินค้า ซึ่งในภาคเรียนนี้จะทำในส่วนของเซนเซอร์และส่วนควบคุมประมวลผลต่อจากภาคเรียนที่แล้วจนสมบูรณ์ซึ่งจัดทำสำหรับที่จอดรถทั้งหมด6 คัน ซึ่งจะสามารถตรวจนับจำนวนรถและหาตำแหน่งที่รถจอด แสดงออกทางแอลอีดี แล้วบันทึกระยะเวลาการจอดรถ นำมาคำนวณค่าจอดรถนำไปแสดงผลที่จอแอลซีดี วงจรต่างๆที่ใช้ในโครงงานนี้ เช่น การอ่านค่าเวลาจากไอซี 1307 เรียลไทม์คล็อก , การคำนวณค่าจอดรถ และการแสดงผลออกจอแอลซีดี โดยที่ในแต่ละส่วนนั้นเกิดปัญหาต่างๆ ดังนี้

5.1.1 ปัญหาของโครงงาน

เนื่องจากเราทำการใช้ อินฟราเรดมาใช้เป็นวงจรเซนเซอร์ ดังนั้นในการต่อเข้ากับวงจรควบคุมและประมวลผล ไมโครคอนโทรลเลอร์ ทำให้สัญญาณที่เข้ามีค่าไม่ถึง 3 โวลต์ ทำให้ไม่สามารถตรวจจับความเปลี่ยนแปลงได้ จึงแก้ปัญหาโดยการเพิ่มค่าศักดาไฟฟ้าที่ตัวรับอินฟราเรดจากเดิม +5 โวลต์ เป็น +9 โวลต์ จึงสามารถที่จะนำไปทำการประมวลผลต่อไปได้

5.1.2 ข้อจำกัดของโครงงาน

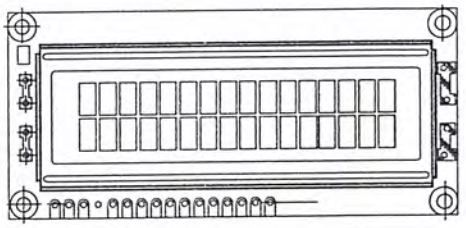
1. เวลาออกจากช่องไปถึงทางออกทำได้ทีละคัน เพราะว่าถ้ามีอีกคันหนึ่งออกไปตอนที่คันแรกยังไม่ถึงทางออกจะมีการคิดเงินผิดพลาด
2. ถ้ารถที่เข้ามากันแรกยังหาที่จอดไม่ได้ แล้วมีรถคันที่ สองเข้ามา เราจะไม่ทราบว่าเวลาไหนเป็นของคันไหน จึงแก้ไขโดยการจัดลำดับให้รถคันแรกไปจอดช่องที่ 1 และรถคันที่สองไปจอดช่องที่ 2 ตามลำดับ
3. ต้องมีรถคันที่ 7 เข้ามาก่อน ถึงจะแสดงว่าที่จอดรถเต็มแล้วดังนั้นคันที่ 7 นั้นจะไม่มีที่จอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

16 x 2 Character LCD



FEATURES

- 5 x 8 dots includes cursor
- Built - in controller (KS 0066 or Equivalent)
- + 5V power supply (Also available for + 3V)
- 1/16 duty cycle
- LED can be driven by pin 1, pin 2, pin 15 or A and K
- N.V. optional for + 3V power supply

MECHANICAL DATA

ITEM	STANDARD VALUE	UNIT
Module Dimension	80.0 x 36.0	mm
Viewing Area	66.0 x 16.0	mm
Mounting Hole	75.0 x 31.0	mm
Character Size	2.95 x 5.55	mm

ABSOLUTE MAXIMUM RATING

ITEM	SYMBOL	STANDARD VALUE			UNIT
		MIN.	TYP.	MAX.	
Power Supply	VDD-VSS	- 0.3	–	7.0	V
Input Voltage	VI	- 0.3	–	VDD	V

NOTE: VSS = 0 Volt, VDD = 5.0 Volt

ELECTRICAL SPECIFICATIONS

ITEM	SYMBOL	CONDITION		STANDARD VALUE			UNIT
				MIN.	TYP.	MAX.	
Input Voltage	VDD	VDD = + 5V		4.7	5.0	5.3	V
Supply Current	IDD	VDD = + 5V		–	1.2	1.4	mA
Recommended LC Driving Voltage for Normal Temp. Version Module	VDD - V0	- 20 °C		–	–	5.2	V
		0°C		–	–	4.2	
		25°C		–	3.8	–	
		50°C		3.5	–	–	
		70°C		3.2	–	–	
LED Forward Voltage	VF	25°C		–	4.2	4.6	V
LED Forward Current	IF	25°C	Array	–	130	260	mA
		25°C	Edge	–	20	40	mA
EL Power Supply Current	IEL	Vel = 110VAC; 400Hz		–	–	5.0	mA

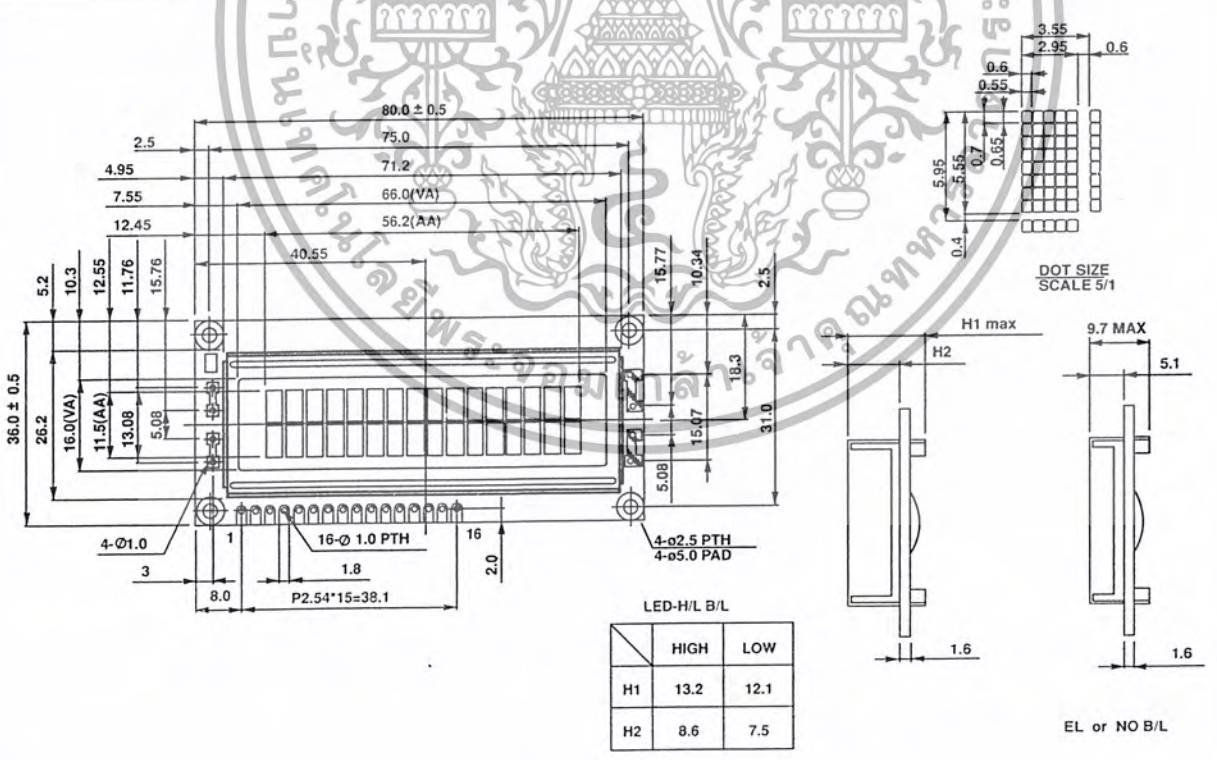
DISPLAY CHARACTER ADDRESS CODE:

Display Position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
DD RAM Address	00	01														0F
DD RAM Address	40	41														4F

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PIN NUMBER	SYMBOL	FUNCTION
1	Vss	GND
2	Vdd	+ 3V or + 5V
3	Vo	Contrast Adjustment
4	RS	H/L Register Select Signal
5	R/$\overline{W}$	H/L Read/Write Signal
6	E	H →L Enable Signal
7	DB0	H/L Data Bus Line
8	DB1	H/L Data Bus Line
9	DB2	H/L Data Bus Line
10	DB3	H/L Data Bus Line
11	DB4	H/L Data Bus Line
12	DB5	H/L Data Bus Line
13	DB6	H/L Data Bus Line
14	DB7	H/L Data Bus Line
15	A/Vee	+ 4.2V for LED (RA = 0Ω)/Negative Voltage Output
16	K	Power Supply for B/L (0V)

DIMENSIONS in millimeters



	HIGH	LOW
H1	13.2	12.1
H2	8.6	7.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

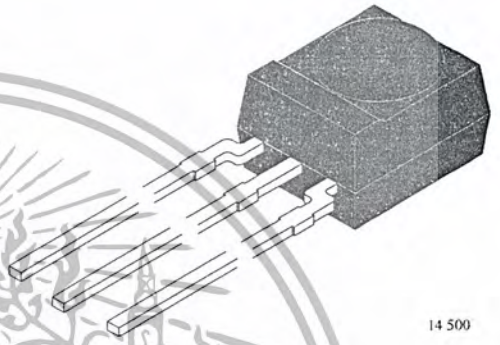
Photo Modules for PCM Remote Control Systems

Available types for different carrier frequencies

Type	fo	Type	fo
TSOP4830	30 kHz	TSOP4833	33 kHz
TSOP4836	36 kHz	TSOP4837	36.7 kHz
TSOP4838	38 kHz	TSOP4840	40 kHz
TSOP4856	56 kHz		

Description

The TSOP48.. – series are miniaturized receivers for infrared remote control systems. PIN diode and preamplifier are assembled on lead frame, the epoxy package is designed as IR filter.
The demodulated output signal can directly be decoded by a microprocessor. TSOP48.. is the standard IR remote control receiver series, supporting all major transmission codes.

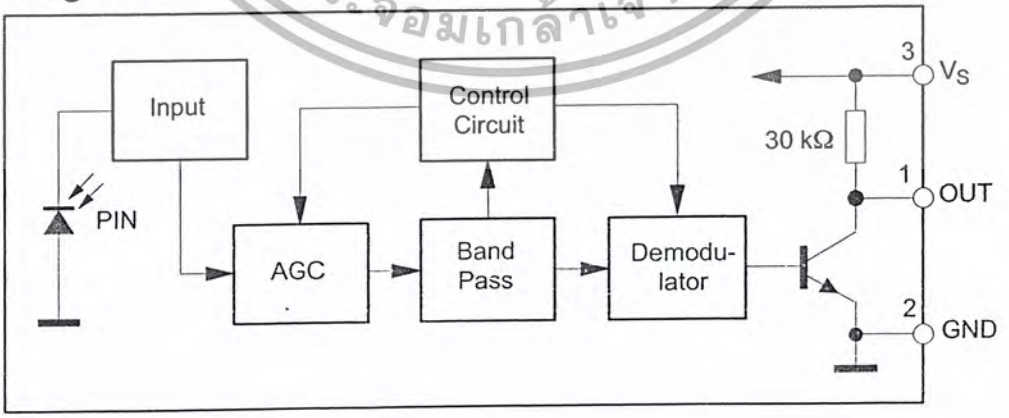


14 500

Features

- Photo detector and preamplifier in one package
- Internal filter for PCM frequency
- Improved shielding against electrical field disturbance
- TTL and CMOS compatibility
- Output active low
- Low power consumption
- High immunity against ambient light
- Continuous data transmission possible (800 bit/s)
- Suitable burst length ≥10 cycles/burst

Block Diagram



9612226

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Absolute Maximum Ratings

T_{amb} = 25°C

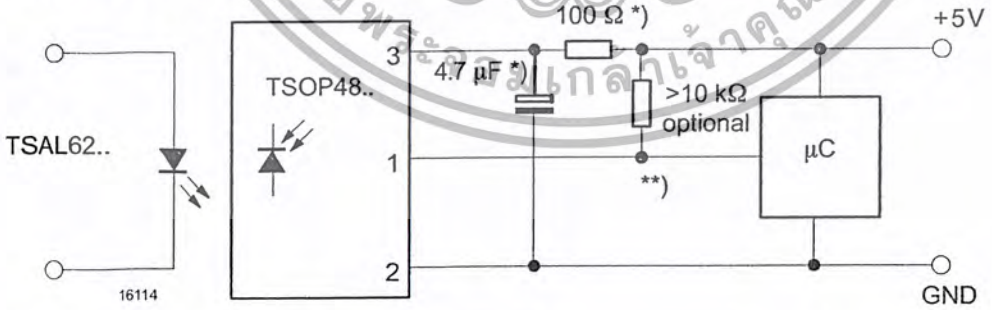
Parameter	Test Conditions	Symbol	Value	Unit
Supply Voltage	(Pin 3)	V_S	–0.3...6.0	V
Supply Current	(Pin 3)	I_S	5	mA
Output Voltage	(Pin 1)	V_O	–0.3...6.0	V
Output Current	(Pin 1)	I_O	5	mA
Junction Temperature		T_j	100	°C
Storage Temperature Range		T_{stg}	–25...+85	°C
Operating Temperature Range		T_{amb}	–25...+85	°C
Power Consumption	(T_{amb} ≦ 85 °C)	P_{tot}	50	mW
Soldering Temperature	t ≦ 10 s, 1 mm from case	T_{sd}	260	°C

Basic Characteristics

T_{amb} = 25°C

Parameter	Test Conditions	Symbol	Min	Typ	Max	Unit
Supply Current (Pin 3)	V_S = 5 V, E_v = 0	I_{SD}	0.8	1.1	1.5	mA
	V_S = 5 V, E_v = 40 klx, sunlight	I_{SH}		1.4		mA
Supply Voltage (Pin 3)		V_S	4.5		5.5	V
Transmission Distance	E_v = 0, test signal see fig.7, IR diode TSAL6200, I_F = 250 mA	d		35		m
Output Voltage Low (Pin 1)	I_{OSL} = 0.5 mA, E_e = 0.7 mW/m²	V_{OSL}			250	mV
Irradiance (30 – 40 kHz)	Pulse width tolerance: $t_{pi} - 5/f_o < t_{po} < t_{pi} + 6/f_o$, test signal see fig.7	$E_{e\,min}$		0.2	0.4	mW/m²
Irradiance (56 kHz)	Pulse width tolerance: $t_{pi} - 5/f_o < t_{po} < t_{pi} + 6/f_o$, test signal see fig.7	$E_{e\,min}$		0.3	0.6	mW/m²
Irradiance	$t_{pi} - 5/f_o < t_{po} < t_{pi} + 6/f_o$	$E_{e\,max}$	30			W/m²
Directivity	Angle of half transmission distance	$\varphi_{1/2}$		±45		deg

Application Circuit



*) recommended to suppress power supply disturbances
**) The output voltage should not be hold continuously at a voltage below 3.3V by the external circuit.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Suitable Data Format

The circuit of the TSOP48.. is designed in that way that unexpected output pulses due to noise or disturbance signals are avoided. A bandpassfilter, an integrator stage and an automatic gain control are used to suppress such disturbances.
The distinguishing mark between data signal and disturbance signal are carrier frequency, burst length and duty cycle.

The data signal should fullfill the following condition:

- Carrier frequency should be close to center frequency of the bandpass (e.g. 38kHz).
- Burst length should be 10 cycles/burst or longer.
- After each burst which is between 10 cycles and 70 cycles a gap time of at least 14 cycles is necessary.
- For each burst which is longer than 1.8ms a corresponding gap time is necessary at some time in the data stream. This gap time should be at least 4 times longer than the burst.
- Up to 800 short bursts per second can be received continuously.

Some examples for suitable data format are:
NEC Code, Toshiba Micom Format, Sharp Code, RC5 Code, RC6 Code, R–2000 Code.

When a disturbance signal is applied to the TSOP48.. it can still receive the data signal. However the sensitivity is reduced to that level that no unexpected pulses will occure.

Some examples for such disturbance signals which are suppressed by the TSOP48.. are:

- DC light (e.g. from tungsten bulb or sunlight)
- Continuous signal at 38kHz or at any other frequency
- Signals from fluorescent lamps with electronic ballast with high or low modulation (see Figure A or Figure B).

Figure A: IR Signal from Fluorescent Lamp with low Modulation

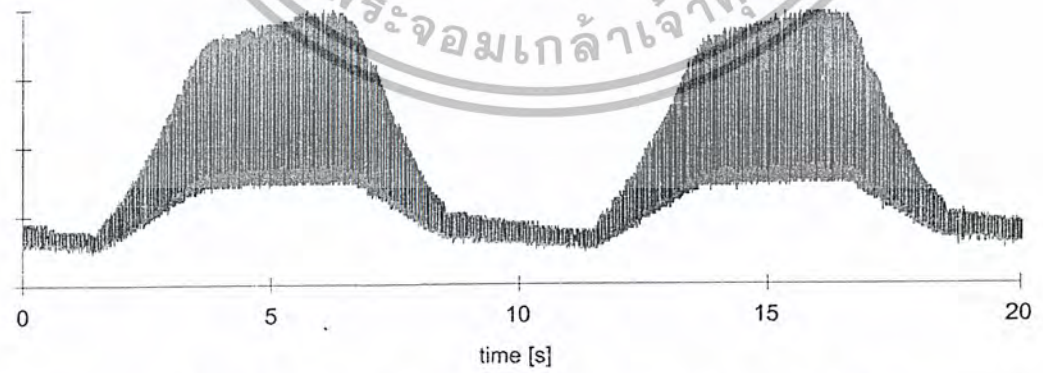


Figure B: IR Signal from Fluorescent Lamp with high Modulation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Typical Characteristics (T_{amb} = 25°C unless otherwise specified)

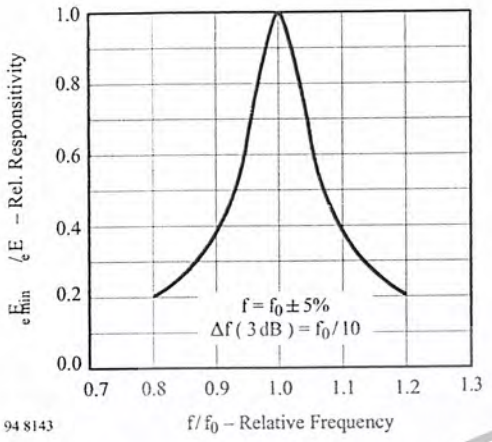


Figure 1. Frequency Dependence of Responsivity

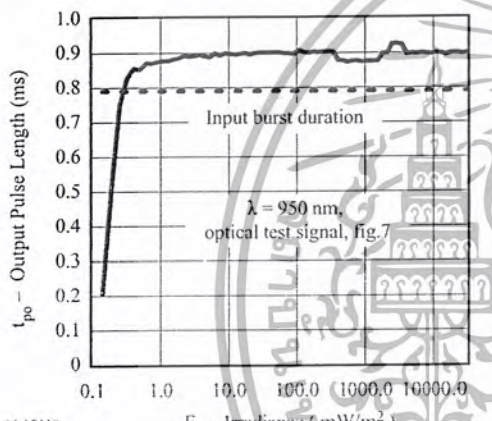


Figure 2. Sensitivity in Dark Ambient

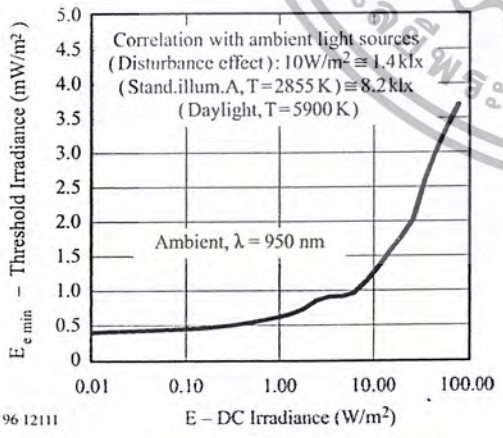


Figure 3. Sensitivity in Bright Ambient

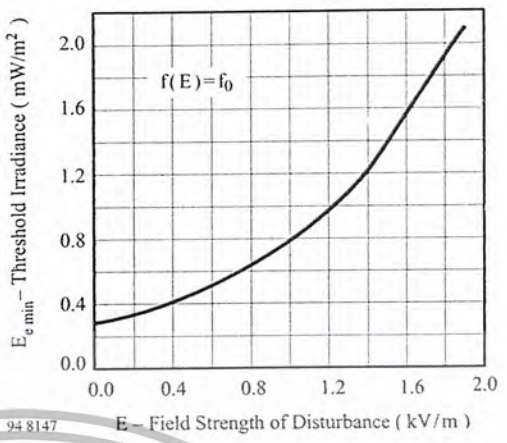


Figure 4. Sensitivity vs. Electric Field Disturbances

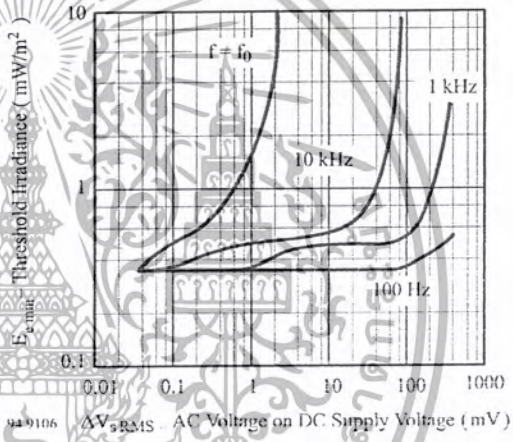


Figure 5. Sensitivity vs. Supply Voltage Disturbances

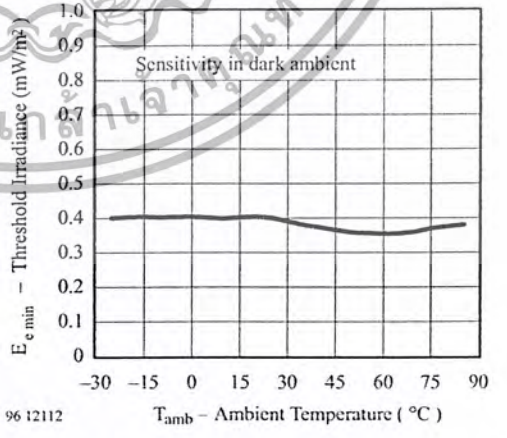
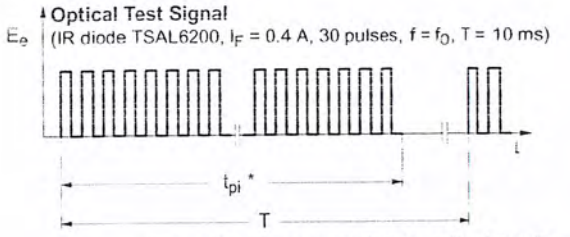


Figure 6. Sensitivity vs. Ambient Temperature

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



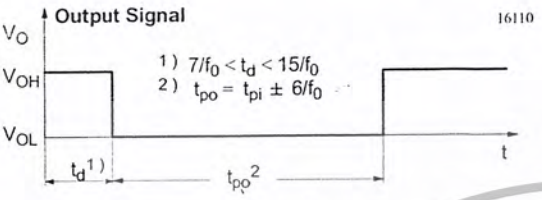


Figure 7.

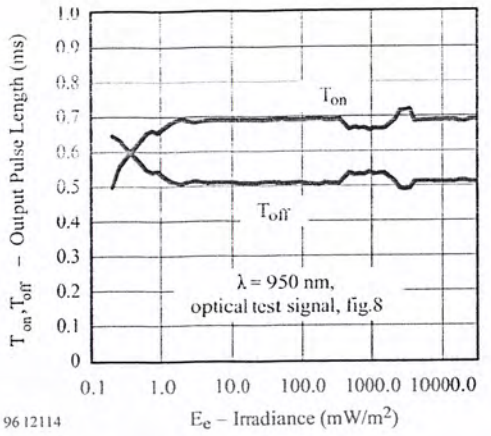


Figure 10. Output Pulse Diagram

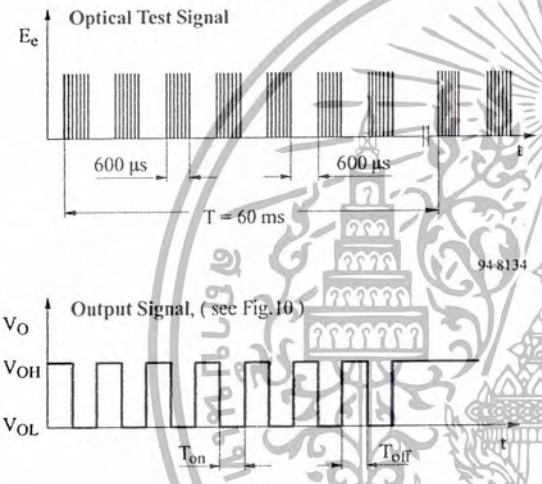


Figure 8. Output Function

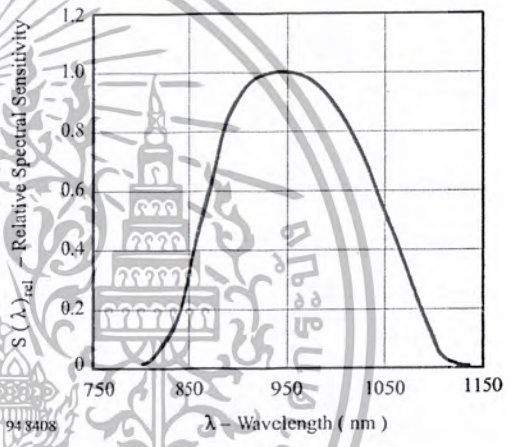


Figure 11. Relative Spectral Sensitivity vs. Wavelength

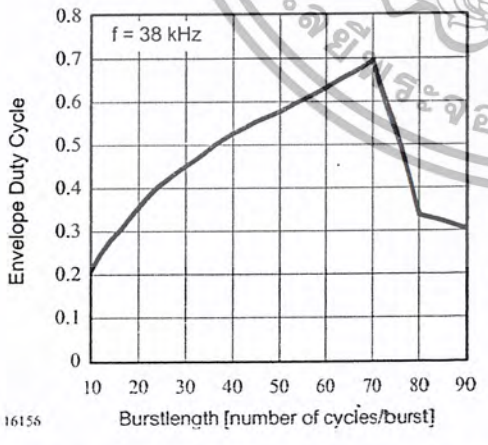


Figure 9. Max. Envelope Duty Cycle vs. Burstlength

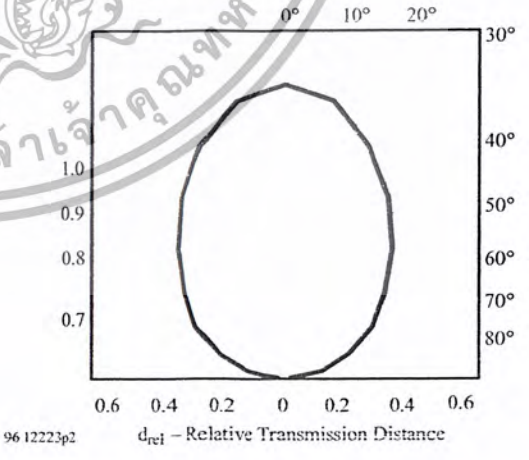
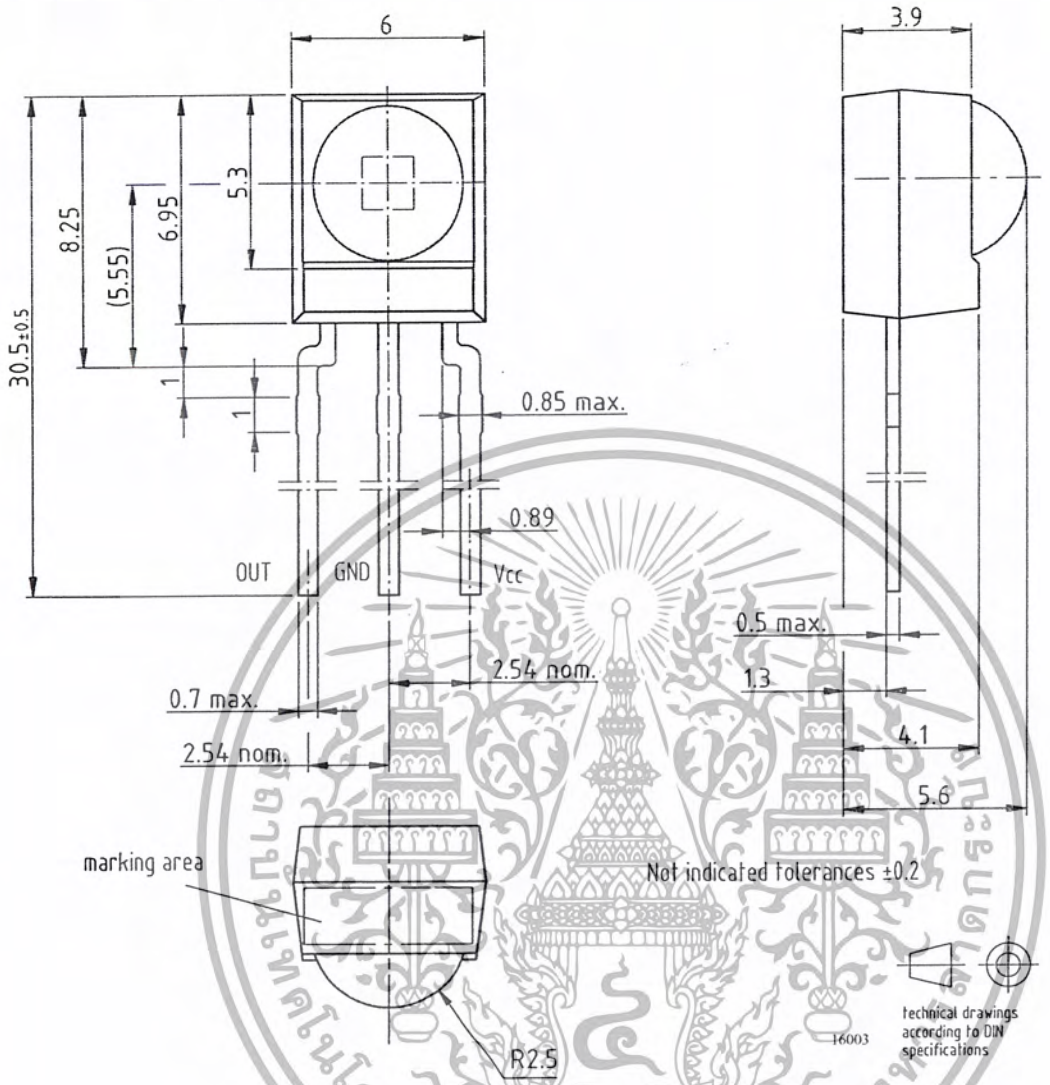


Figure 12. Directivity

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Dimensions in mm



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Ozone Depleting Substances Policy Statement

It is the policy of **Vishay Semiconductor GmbH** to

1. Meet all present and future national and international statutory requirements.

2. Regularly and continuously improve the performance of our products, processes, distribution and operating systems with respect to their impact on the health and safety of our employees and the public, as well as their impact on the environment.

It is particular concern to control or eliminate releases of those substances into the atmosphere which are known as ozone depleting substances (ODSs).

The Montreal Protocol (1987) and its London Amendments (1990) intend to severely restrict the use of ODSs and forbid their use within the next ten years. Various national and international initiatives are pressing for an earlier ban on these substances.

Vishay Semiconductor GmbH has been able to use its policy of continuous improvements to eliminate the use of ODSs listed in the following documents.

1. Annex A, B and list of transitional substances of the Montreal Protocol and the London Amendments respectively

2. Class I and II ozone depleting substances in the Clean Air Act Amendments of 1990 by the Environmental Protection Agency (EPA) in the USA

3. Council Decision 88/540/EEC and 91/690/EEC Annex A, B and C (transitional substances) respectively.

Vishay Semiconductor GmbH can certify that our semiconductors are not manufactured with ozone depleting substances and do not contain such substances.

We reserve the right to make changes to improve technical design and may do so without further notice.
Parameters can vary in different applications. All operating parameters must be validated for each customer application by the customer. Should the buyer use Vishay-Telefunken products for any unintended or unauthorized application, the buyer shall indemnify Vishay-Telefunken against all claims, costs, damages, and expenses, arising out of, directly or indirectly, any claim of personal damage, injury or death associated with such unintended or unauthorized use.

Vishay Semiconductor GmbH, P.O.B. 3535, D-74025 Heilbronn, Germany
Telephone: 49 (0)7131 67 2831, Fax number: 49 (0)7131 67 2423

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DALLAS SEMICONDUCTOR

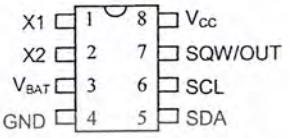
DS1307
64 x 8 Serial Real-Time Clock

www.maxim-ic.com

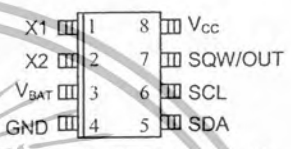
FEATURES

- Real-time clock (RTC) counts seconds, minutes, hours, date of the month, month, day of the week, and year with leap-year compensation valid up to 2100
- 56-byte, battery-backed, nonvolatile (NV) RAM for data storage
- Two-wire serial interface
- Programmable squarewave output signal
- Automatic power-fail detect and switch circuitry
- Consumes less than 500nA in battery backup mode with oscillator running
- Optional industrial temperature range: -40°C to +85°C
- Available in 8-pin DIP or SOIC
- Underwriters Laboratory (UL) recognized

PIN ASSIGNMENT



DS1307 8-Pin DIP (300-mil)



DS1307 8-Pin SOIC (150-mil)

PIN DESCRIPTION

V_{CC}	- Primary Power Supply
X1, X2	- 32.768kHz Crystal Connection
V_{BAT}	- +3V Battery Input
GND	- Ground
SDA	- Serial Data
SCL	- Serial Clock
SQW/OUT	- Square Wave/Output Driver

ORDERING INFORMATION

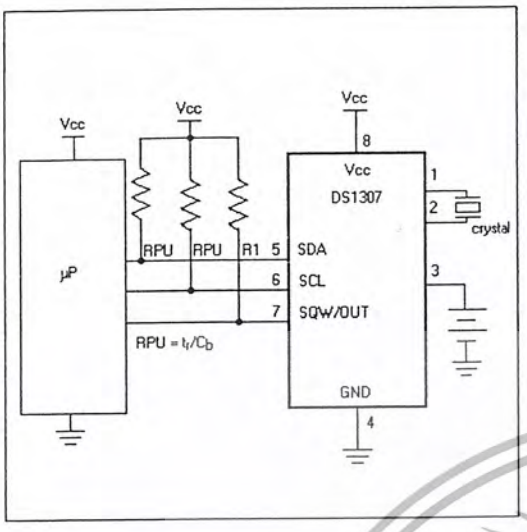
DS1307	8-Pin DIP (300-mil)
DS1307Z	8-Pin SOIC (150-mil)
DS1307N	8-Pin DIP (Industrial)
DS1307ZN	8-Pin SOIC (Industrial)

DESCRIPTION

The DS1307 Serial Real-Time Clock is a low-power, full binary-coded decimal (BCD) clock/calendar plus 56 bytes of NV SRAM. Address and data are transferred serially via a 2-wire, bi-directional bus. The clock/calendar provides seconds, minutes, hours, day, date, month, and year information. The end of the month date is automatically adjusted for months with fewer than 31 days, including corrections for leap year. The clock operates in either the 24-hour or 12-hour format with AM/PM indicator. The DS1307 has a built-in power sense circuit that detects power failures and automatically switches to the battery supply.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

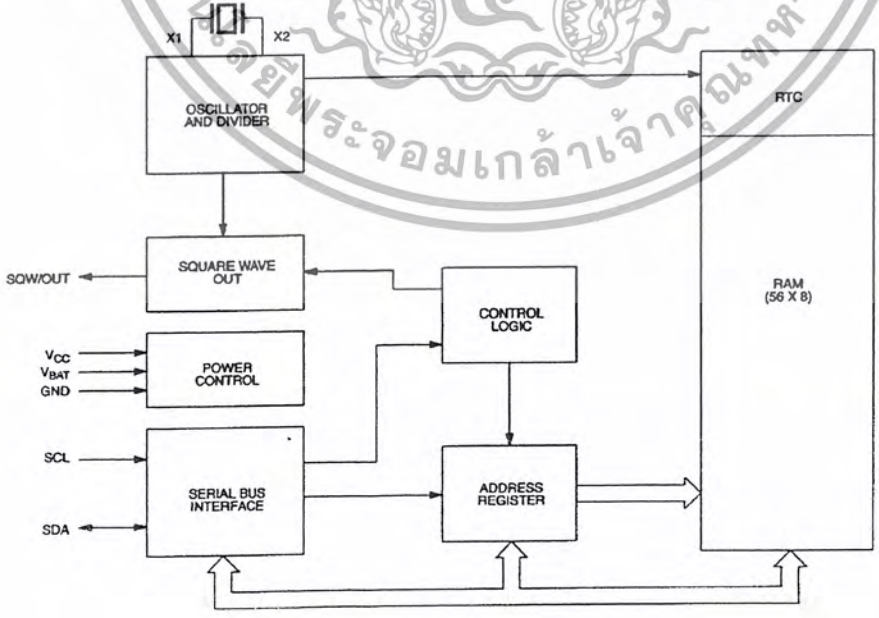
TYPICAL OPERATING CIRCUIT



OPERATION

The DS1307 operates as a slave device on the serial bus. Access is obtained by implementing a START condition and providing a device identification code followed by a register address. Subsequent registers can be accessed sequentially until a STOP condition is executed. When V_{CC} falls below 1.25 x V_{BAT} the device terminates an access in progress and resets the device address counter. Inputs to the device will not be recognized at this time to prevent erroneous data from being written to the device from an out of tolerance system. When V_{CC} falls below V_{BAT} the device switches into a low-current battery backup mode. Upon power-up, the device switches from battery to V_{CC} when V_{CC} is greater than V_{BAT} + 0.2V and recognizes inputs when V_{CC} is greater than 1.25 x V_{BAT}. The block diagram in Figure 1 shows the main elements of the serial RTC.

DS1307 BLOCK DIAGRAM Figure 1



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SIGNAL DESCRIPTIONS

V_{CC}, GND – DC power is provided to the device on these pins. V_{CC} is the +5V input. When 5V is applied within normal limits, the device is fully accessible and data can be written and read. When a 3V battery is connected to the device and V_{CC} is below 1.25 x V_{BAT}, reads and writes are inhibited. However, the timekeeping function continues unaffected by the lower input voltage. As V_{CC} falls below V_{BAT} the RAM and timekeeper are switched over to the external power supply (nominal 3.0V DC) at V_{BAT}.

V_{BAT} – Battery input for any standard 3V lithium cell or other energy source. Battery voltage must be held between 2.0V and 3.5V for proper operation. The nominal write protect trip point voltage at which access to the RTC and user RAM is denied is set by the internal circuitry as 1.25 x V_{BAT} nominal. A lithium battery with 48mAhr or greater will back up the DS1307 for more than 10 years in the absence of power at 25°C. UL recognized to ensure against reverse charging current when used in conjunction with a lithium battery.

See "Conditions of Acceptability" at http://www.maxim-ic.com/TechSupport/QA/ntrl.htm.

SCL (Serial Clock Input) – SCL is used to synchronize data movement on the serial interface.

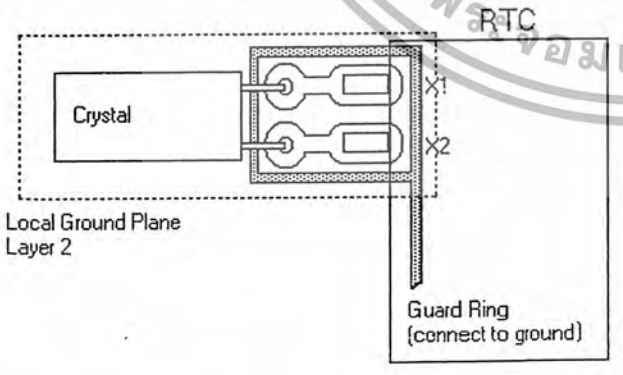
SDA (Serial Data Input/Output) – SDA is the input/output pin for the 2-wire serial interface. The SDA pin is open drain which requires an external pullup resistor.

SQW/OUT (Square Wave/Output Driver) – When enabled, the SQWE bit set to 1, the SQW/OUT pin outputs one of four square wave frequencies (1Hz, 4kHz, 8kHz, 32kHz). The SQW/OUT pin is open drain and requires an external pull-up resistor. SQW/OUT will operate with either Vcc or Vbat applied.

X1, X2 – Connections for a standard 32.768kHz quartz crystal. The internal oscillator circuitry is designed for operation with a crystal having a specified load capacitance (CL) of 12.5pF.

For more information on crystal selection and crystal layout considerations, please consult Application Note 58, "Crystal Considerations with Dallas Real-Time Clocks." The DS1307 can also be driven by an external 32.768kHz oscillator. In this configuration, the X1 pin is connected to the external oscillator signal and the X2 pin is floated.

RECOMMENDED LAYOUT FOR CRYSTAL



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CLOCK ACCURACY

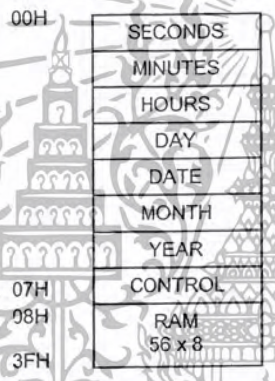
The accuracy of the clock is dependent upon the accuracy of the crystal and the accuracy of the match between the capacitive load of the oscillator circuit and the capacitive load for which the crystal was trimmed. Additional error will be added by crystal frequency drift caused by temperature shifts. External circuit noise coupled into the oscillator circuit may result in the clock running fast. See Application Note 58, "Crystal Considerations with Dallas Real-Time Clocks" for detailed information.

Please review Application Note 95, "Interfacing the DS1307 with a 8051-Compatible Microcontroller" for additional information.

RTC AND RAM ADDRESS MAP

The address map for the RTC and RAM registers of the DS1307 is shown in Figure 2. The RTC registers are located in address locations 00h to 07h. The RAM registers are located in address locations 08h to 3Fh. During a multi-byte access, when the address pointer reaches 3Fh, the end of RAM space, it wraps around to location 00h, the beginning of the clock space.

DS1307 ADDRESS MAP Figure 2



CLOCK AND CALENDAR

The time and calendar information is obtained by reading the appropriate register bytes. The RTC registers are illustrated in Figure 3. The time and calendar are set or initialized by writing the appropriate register bytes. The contents of the time and calendar registers are in the BCD format. Bit 7 of register 0 is the clock halt (CH) bit. When this bit is set to a 1, the oscillator is disabled. When cleared to a 0, the oscillator is enabled.

Please note that the initial power-on state of all registers is not defined. Therefore, it is important to enable the oscillator (CH bit = 0) during initial configuration.

The DS1307 can be run in either 12-hour or 24-hour mode. Bit 6 of the hours register is defined as the 12- or 24-hour mode select bit. When high, the 12-hour mode is selected. In the 12-hour mode, bit 5 is the AM/PM bit with logic high being PM. In the 24-hour mode, bit 5 is the second 10 hour bit (20-23 hours).

On a 2-wire START, the current time is transferred to a second set of registers. The time information is read from these secondary registers, while the clock may continue to run. This eliminates the need to re-read the registers in case of an update of the main registers during a read.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DS1307 TIMEKEEPER REGISTERS Figure 3

	BIT7						BIT0	
00H	CH	10 SECONDS			SECONDS			00–59
	0	10 MINUTES			MINUTES			00–59
	0	12/24	10 HR A/P	10 HR	HOURS			01–12 00–23
	0	0	0	0	0	DAY		1–7
	0	0	10 DATE		DATE			01–28/29 01–30 01–31
	0	0	0	10 MONTH	MONTH			01–12
	10 YEAR				YEAR			00–99
07H	OUT	0	0	SQWE	0	0	RS1	RS0

CONTROL REGISTER

The DS1307 control register is used to control the operation of the SQW/OUT pin.

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
OUT	0	0	SQWE	0	0	RS1	RS0

OUT (Output control): This bit controls the output level of the SQW/OUT pin when the square wave output is disabled. If SQWE = 0, the logic level on the SQW/OUT pin is 1 if OUT = 1 and is 0 if OUT = 0.

SQWE (Square Wave Enable): This bit, when set to a logic 1, will enable the oscillator output. The frequency of the square wave output depends upon the value of the RS0 and RS1 bits. With the square wave output set to 1Hz, the clock registers update on the falling edge of the square wave.

RS (Rate Select): These bits control the frequency of the square wave output when the square wave output has been enabled. Table 1 lists the square wave frequencies that can be selected with the RS bits.

SQUAREWAVE OUTPUT FREQUENCY Table 1

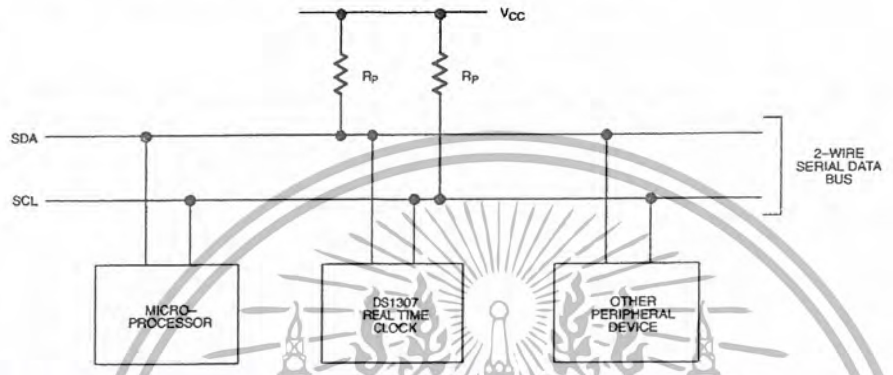
RS1	RS0	SQW OUTPUT FREQUENCY
0	0	1Hz
0	1	4.096kHz
1	0	8.192kHz
1	1	32.768kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2-WIRE SERIAL DATA BUS

The DS1307 supports a bi-directional, 2-wire bus and data transmission protocol. A device that sends data onto the bus is defined as a transmitter and a device receiving data as a receiver. The device that controls the message is called a master. The devices that are controlled by the master are referred to as slaves. The bus must be controlled by a master device that generates the serial clock (SCL), controls the bus access, and generates the START and STOP conditions. The DS1307 operates as a slave on the 2-wire bus. A typical bus configuration using this 2-wire protocol is show in Figure 4.

TYPICAL 2-WIRE BUS CONFIGURATION Figure 4



Figures 5, 6, and 7 detail how data is transferred on the 2-wire bus.

- Data transfer may be initiated only when the bus is not busy.
- During data transfer, the data line must remain stable whenever the clock line is HIGH. Changes in the data line while the clock line is high will be interpreted as control signals.

Accordingly, the following bus conditions have been defined:

Bus not busy: Both data and clock lines remain HIGH.

Start data transfer: A change in the state of the data line, from HIGH to LOW, while the clock is HIGH, defines a START condition.

Stop data transfer: A change in the state of the data line, from LOW to HIGH, while the clock line is HIGH, defines the STOP condition.

Data valid: The state of the data line represents valid data when, after a START condition, the data line is stable for the duration of the HIGH period of the clock signal. The data on the line must be changed during the LOW period of the clock signal. There is one clock pulse per bit of data.

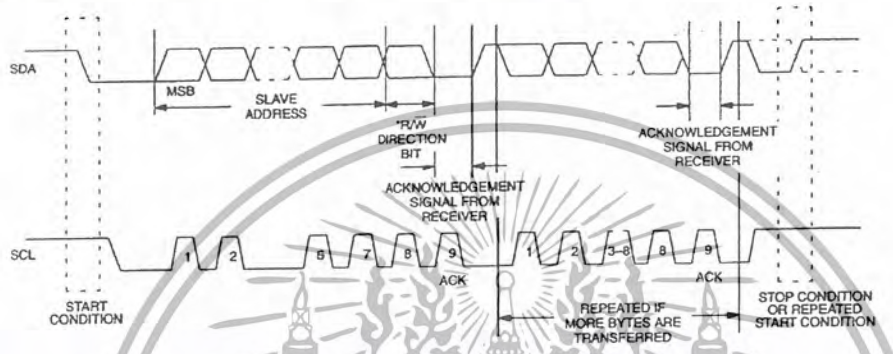
Each data transfer is initiated with a START condition and terminated with a STOP condition. The number of data bytes transferred between START and STOP conditions is not limited, and is determined by the master device. The information is transferred byte-wise and each receiver acknowledges with a ninth bit. Within the 2-wire bus specifications a regular mode (100kHz clock rate) and a fast mode (400kHz clock rate) are defined. The DS1307 operates in the regular mode (100kHz) only.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Acknowledge: Each receiving device, when addressed, is obliged to generate an acknowledge after the reception of each byte. The master device must generate an extra clock pulse which is associated with this acknowledge bit.

A device that acknowledges must pull down the SDA line during the acknowledge clock pulse in such a way that the SDA line is stable LOW during the HIGH period of the acknowledge related clock pulse. Of course, setup and hold times must be taken into account. A master must signal an end of data to the slave by not generating an acknowledge bit on the last byte that has been clocked out of the slave. In this case, the slave must leave the data line HIGH to enable the master to generate the STOP condition.

DATA TRANSFER ON 2-WIRE SERIAL BUS Figure 5



Depending upon the state of the R/$\overline{W}$ bit, two types of data transfer are possible:

1. **Data transfer from a master transmitter to a slave receiver.** The first byte transmitted by the master is the slave address. Next follows a number of data bytes. The slave returns an acknowledge bit after each received byte. Data is transferred with the most significant bit (MSB) first.

2. **Data transfer from a slave transmitter to a master receiver.** The first byte (the slave address) is transmitted by the master. The slave then returns an acknowledge bit. This is followed by the slave transmitting a number of data bytes. The master returns an acknowledge bit after all received bytes other than the last byte. At the end of the last received byte, a "not acknowledge" is returned.

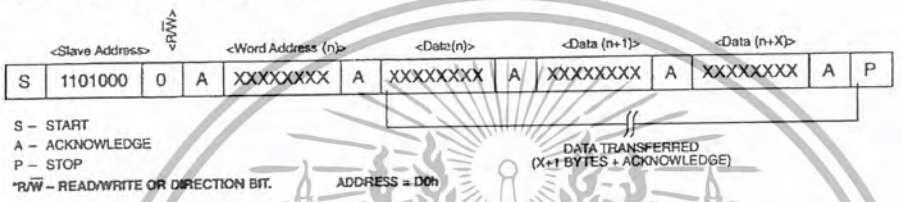
The master device generates all of the serial clock pulses and the START and STOP conditions. A transfer is ended with a STOP condition or with a repeated START condition. Since a repeated START condition is also the beginning of the next serial transfer, the bus will not be released. Data is transferred with the most significant bit (MSB) first.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The DS1307 may operate in the following two modes:

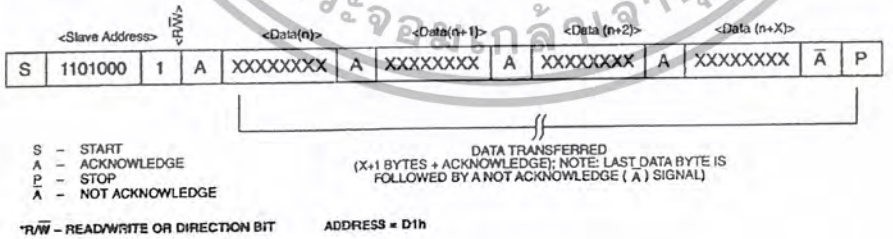
1. **Slave receiver mode (DS1307 write mode):** Serial data and clock are received through SDA and SCL. After each byte is received an acknowledge bit is transmitted. START and STOP conditions are recognized as the beginning and end of a serial transfer. Address recognition is performed by hardware after reception of the slave address and *direction bit (See Figure 6). The address byte is the first byte received after the start condition is generated by the master. The address byte contains the 7 bit DS1307 address, which is 1101000, followed by the *direction bit (R/$\overline{W}$) which, for a write, is a 0. After receiving and decoding the address byte the device outputs an acknowledge on the SDA line. After the DS1307 acknowledges the slave address + write bit, the master transmits a register address to the DS1307 This will set the register pointer on the DS1307. The master will then begin transmitting each byte of data with the DS1307 acknowledging each byte received. The master will generate a stop condition to terminate the data write.

DATA WRITE – SLAVE RECEIVER MODE Figure 6



2. **Slave transmitter mode (DS1307 read mode):** The first byte is received and handled as in the slave receiver mode. However, in this mode, the *direction bit will indicate that the transfer direction is reversed. Serial data is transmitted on SDA by the DS1307 while the serial clock is input on SCL. START and STOP conditions are recognized as the beginning and end of a serial transfer (See Figure 7). The address byte is the first byte received after the start condition is generated by the master. The address byte contains the 7-bit DS1307 address, which is 1101000, followed by the *direction bit (R/$\overline{W}$) which, for a read, is a 1. After receiving and decoding the address byte the device inputs an acknowledge on the SDA line. The DS1307 then begins to transmit data starting with the register address pointed to by the register pointer. If the register pointer is not written to before the initiation of a read mode the first address that is read is the last one stored in the register pointer. The DS1307 must receive a "not acknowledge" to end a read.

DATA READ – SLAVE TRANSMITTER MODE Figure 7



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ABSOLUTE MAXIMUM RATINGS*

Voltage on Any Pin Relative to Ground	-0.5V to +7.0V
Storage Temperature	-55°C to +125°C
Soldering Temperature	260°C for 10 seconds DIP See JPC/JEDEC Standard J-STD-020A for Surface Mount Devices

* This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operation sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods of time may affect reliability.

Range	Temperature	V_{CC}
Commercial	0°C to +70°C	4.5V to 5.5V V_{CC1}
Industrial	-40°C to +85°C	4.5V to 5.5V V_{CC1}

RECOMMENDED DC OPERATING CONDITIONS

(Over the operating range*)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Supply Voltage	V_{CC}	4.5	5.0	5.5	V	
Logic 1	V_{IH}	2.2		V_{CC} + 0.3	V	
Logic 0	V_{IL}	-0.5		+0.8	V	
V_{BAT} Battery Voltage	V_{BAT}	2.0		3.5	V	

*Unless otherwise specified.

DC ELECTRICAL CHARACTERISTICS

(Over the operating range*)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Input Leakage (SCL)	I_{LI}			1	µA	
I/O Leakage (SDA & SQW/OUT)	I_{LO}			1	µA	
Logic 0 Output (I_{OL} = 5mA)	V_{OL}			0.4	V	
Active Supply Current	I_{CCA}			1.5	mA	7
Standby Current	I_{CCS}			200	µA	1
Battery Current (OSC ON); SQW/OUT OFF	I_{BAT1}		300	500	nA	2
Battery Current (OSC ON); SQW/OUT ON (32kHz)	I_{BAT2}		480	800	nA	
Power-Fail Voltage	V_{PF}	1.216 x V_{BAT}	1.25 x V_{BAT}	1.284 x V_{BAT}	V	8

*Unless otherwise specified.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AC ELECTRICAL CHARACTERISTICS

(Over the operating range*)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
SCL Clock Frequency	f_{SCL}	0		100	kHz	
Bus Free Time Between a STOP and START Condition	t_{BUF}	4.7			µs	
Hold Time (Repeated) START Condition	$t_{HD:STA}$	4.0			µs	3
LOW Period of SCL Clock	t_{LOW}	4.7			µs	
HIGH Period of SCL Clock	t_{HIGH}	4.0			µs	
Set-up Time for a Repeated START Condition	$t_{SU:STA}$	4.7			µs	
Data Hold Time	$t_{HD:DAT}$	0			µs	4,5
Data Set-up Time	$t_{SU:DAT}$	250			ns	
Rise Time of Both SDA and SCL Signals	t_R			1000	ns	
Fall Time of Both SDA and SCL Signals	t_F			300	ns	
Set-up Time for STOP Condition	$t_{SU:STO}$	4.7			µs	
Capacitive Load for each Bus Line	C_B			400	pF	6
I/O Capacitance (T_A = 25°C)	$C_{I/O}$		10		pF	
Crystal Specified Load Capacitance (T_A = 25°C)			12.5		pF	

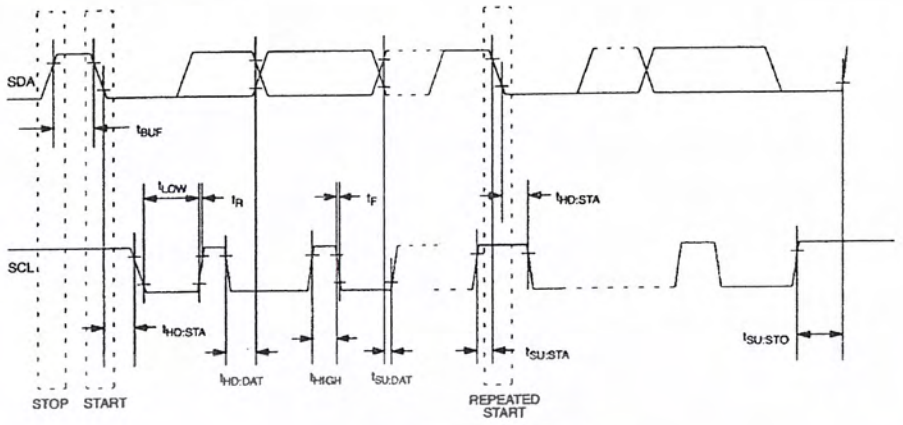
*Unless otherwise specified.

NOTES:

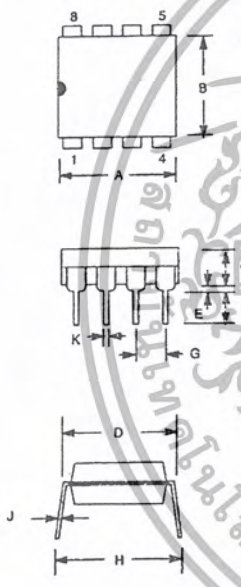
1. I_{CCS} specified with V_{CC} = 5.0V and SDA, SCL = 5.0V.
2. V_{CC} = 0V, V_{BAT} = 3V.
3. After this period, the first clock pulse is generated.
4. A device must internally provide a hold time of at least 300ns for the SDA signal (referred to the V_{IHMIN} of the SCL signal) in order to bridge the undefined region of the falling edge of SCL.
5. The maximum $t_{HD:DAT}$ has only to be met if the device does not stretch the LOW period (t_{LOW}) of the SCL signal.
6. C_B – Total capacitance of one bus line in pF.
7. I_{CCA} – SCL clocking at max frequency = 100kHz.
8. V_{PF} measured at V_{BAT} = 3.0V.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TIMING DIAGRAM Figure 8



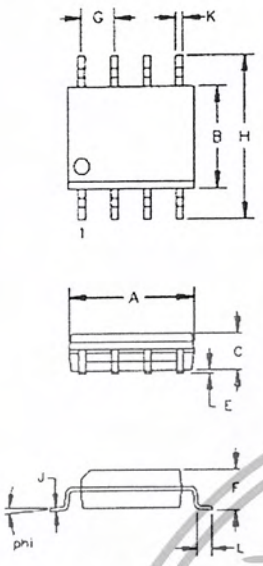
DS1307 64 X 8 SERIAL REAL-TIME CLOCK
8-PIN DIP MECHANICAL DIMENSIONS



PKG	8-PIN	
DIM	MIN	MAX
A IN.	0.360	0.400
MM	9.14	10.16
B IN.	0.240	0.260
MM	6.10	6.60
C IN.	0.120	0.140
MM	3.05	3.56
D IN.	0.300	0.325
MM	7.62	8.26
E IN.	0.015	0.040
MM	0.38	1.02
F IN.	0.120	0.140
MM	3.04	3.56
G IN.	0.090	0.110
MM	2.29	2.79
H IN.	0.320	0.370
MM	8.13	9.40
J IN.	0.008	0.012
MM	0.20	0.30
K IN.	0.015	0.021
MM	0.38	0.53

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DS1307Z 64 X 8 SERIAL REAL-TIME CLOCK
8-PIN SOIC (150-MIL) MECHANICAL DIMENSIONS



PKG	8-PIN (150 MIL)	
DIM	MIN	MAX
A IN.	0.188	0.196
MM	4.78	4.98
B IN.	0.150	0.158
MM	3.81	4.01
C IN.	0.048	0.062
MM	1.22	1.57
E IN.	0.004	0.010
MM	0.10	0.25
F IN.	0.053	0.069
MM	1.35	1.75
G IN.	0.050 BSC	
MM	1.27 BSC	
H IN.	0.230	0.244
MM	5.84	6.20
J IN.	0.007	0.011
MM	0.18	0.28
K IN.	0.012	0.020
MM	0.30	0.51
L IN.	0.016	0.050
MM	0.41	1.27
phi	0°	8°

56-G2008-001

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Features

- Compatible with MCS-51™ Products
- 4K Bytes of In-System Reprogrammable Flash Memory
  - Endurance: 1,000 Write/Erase Cycles
- Fully Static Operation: 0 Hz to 24 MHz
- Three-level Program Memory Lock
- 128 x 8-bit Internal RAM
- 32 Programmable I/O Lines
- Two 16-bit Timer/Counters
- Six Interrupt Sources
- Programmable Serial Channel
- Low-power Idle and Power-down Modes

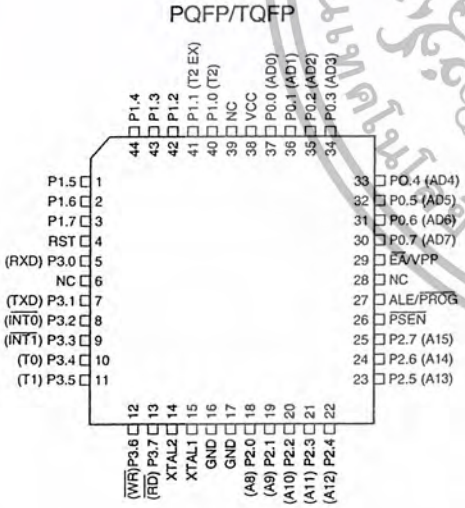
Description

The AT89C51 is a low-power, high-performance CMOS 8-bit microcomputer with 4K bytes of Flash programmable and erasable read only memory (PEROM). The device is manufactured using Atmel's high-density nonvolatile memory technology and is compatible with the industry-standard MCS-51 instruction set and pinout. The on-chip Flash allows the program memory to be reprogrammed in-system or by a conventional nonvolatile memory programmer. By combining a versatile 8-bit CPU with Flash on a monolithic chip, the Atmel AT89C51 is a powerful microcomputer which provides a highly-flexible and cost-effective solution to many embedded control applications.

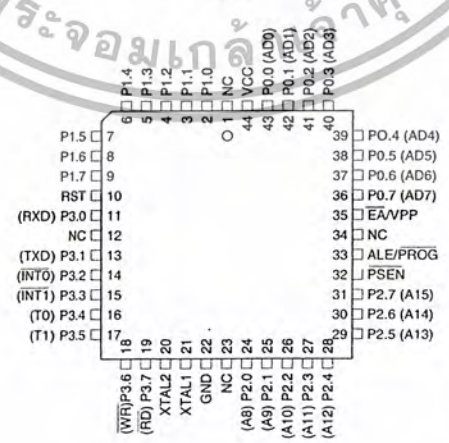
Pin Configurations

PDIP

PQFP/TQFP



PLCC



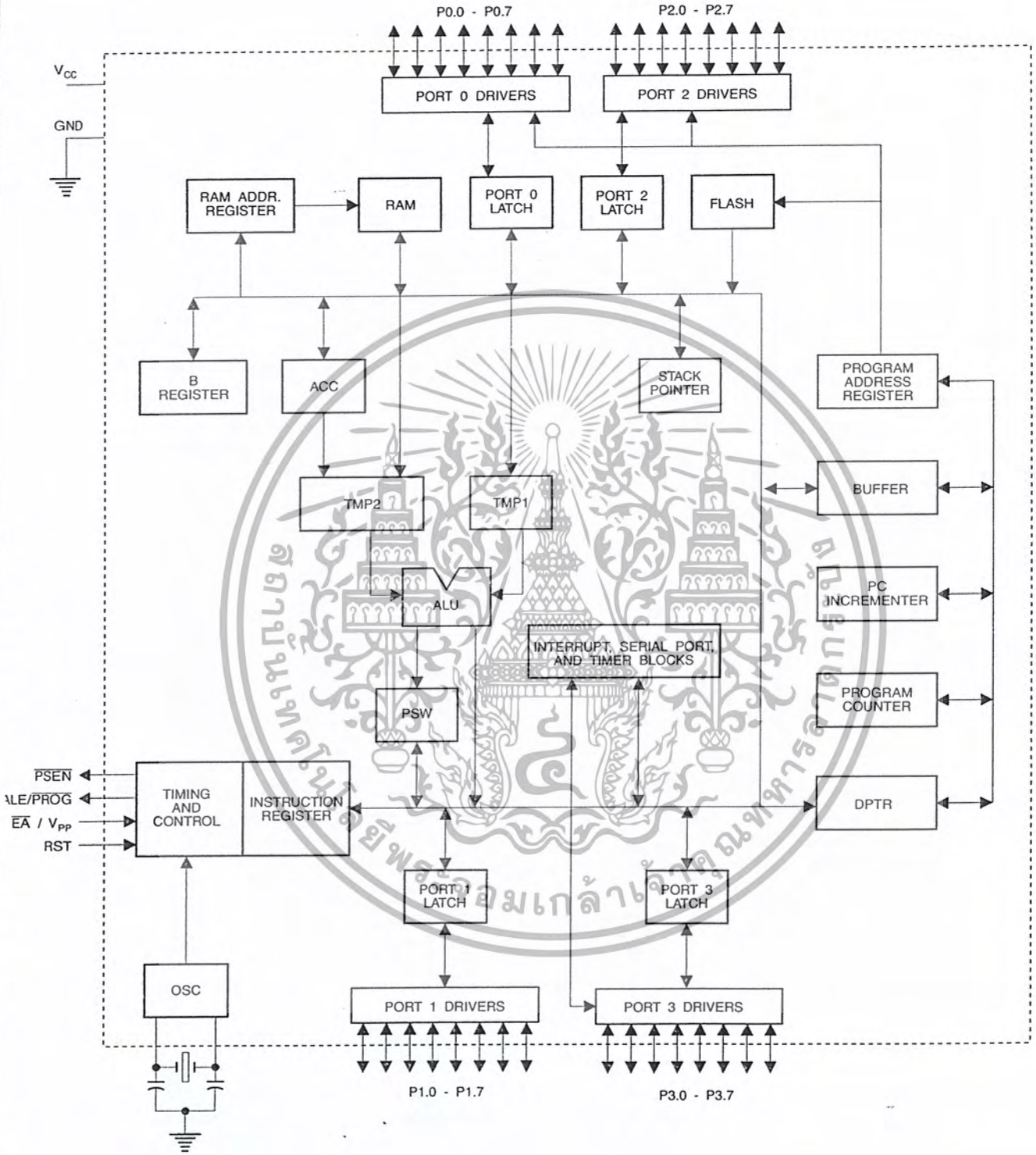
8-bit Microcontroller with 4K Bytes Flash

AT89C51

Rev. 0265G–02/00

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Block Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

pulse is skipped during each access to external Data Memory.

If desired, ALE operation can be disabled by setting bit 0 of SFR location 8EH. With the bit set, ALE is active only during a MOVX or MOVC instruction. Otherwise, the pin is weakly pulled high. Setting the ALE-disable bit has no effect if the microcontroller is in external execution mode.

$\overline{\text{PSEN}}$

Program Store Enable is the read strobe to external program memory.

When the AT89C51 is executing code from external program memory, $\overline{\text{PSEN}}$ is activated twice each machine cycle, except that two $\overline{\text{PSEN}}$ activations are skipped during each access to external data memory.

$\overline{\text{EA}}$/VPP

External Access Enable. $\overline{\text{EA}}$ must be strapped to GND in order to enable the device to fetch code from external program memory locations starting at 0000H up to FFFFH. Note, however, that if lock bit 1 is programmed, $\overline{\text{EA}}$ will be internally latched on reset.

$\overline{\text{EA}}$ should be strapped to V_{CC} for internal program executions.

This pin also receives the 12-volt programming enable voltage (V_{PP}) during Flash programming, for parts that require 12-volt V_{PP}.

XTAL1

Input to the inverting oscillator amplifier and input to the internal clock operating circuit.

XTAL2

Output from the inverting oscillator amplifier.

Oscillator Characteristics

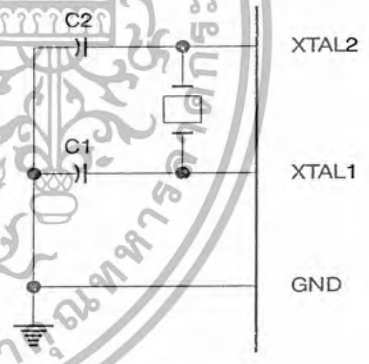
XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier which can be configured for use as an on-chip oscillator, as shown in Figure 1. Either a quartz crystal or ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected while XTAL1 is driven as shown in Figure 2. There are no requirements on the duty cycle of the external clock signal, since the input to the internal clocking circuitry is through a divide-by-two flip-flop, but minimum and maximum voltage high and low time specifications must be observed.

Idle Mode

In idle mode, the CPU puts itself to sleep while all the on-chip peripherals remain active. The mode is invoked by software. The content of the on-chip RAM and all the special functions registers remain unchanged during this mode. The idle mode can be terminated by any enabled interrupt or by a hardware reset.

It should be noted that when idle is terminated by a hard ware reset, the device normally resumes program execution, from where it left off, up to two machine cycles before the internal reset algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to the port pins is not inhibited. To eliminate the possibility of an unexpected write to a port pin when Idle is terminated by reset, the instruction following the one that invokes Idle should not be one that writes to a port pin or to external memory.

Figure 1. Oscillator Connections



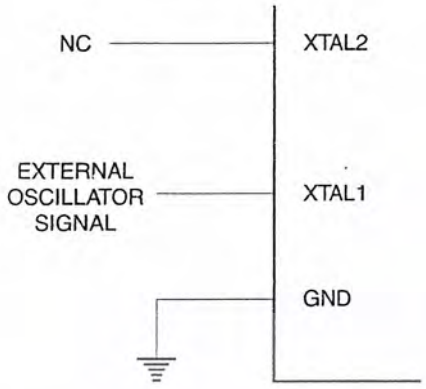
Note: C1, C2 = 30 pF ± 10 pF for Crystals
= 40 pF ± 10 pF for Ceramic Resonators

Status of External Pins During Idle and Power-down Modes

Mode	Program Memory	ALE	$\overline{\text{PSEN}}$	PORT0	PORT1	PORT2	PORT3
Idle	Internal	1	1	Data	Data	Data	Data
Idle	External	1	1	Float	Data	Address	Data
Power-down	Internal	0	0	Data	Data	Data	Data
Power-down	External	0	0	Float	Data	Data	Data

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 2. External Clock Drive Configuration



Power-down Mode

In the power-down mode, the oscillator is stopped, and the instruction that invokes power-down is the last instruction executed. The on-chip RAM and Special Function Registers retain their values until the power-down mode is terminated. The only exit from power-down is a hardware reset. Reset redefines the SFRs but does not change the on-chip RAM. The reset should not be activated before V_{CC} is restored to its normal operating level and must be held active long enough to allow the oscillator to restart and stabilize.

Program Memory Lock Bits

On the chip are three lock bits which can be left unprogrammed (U) or can be programmed (P) to obtain the additional features listed in the table below.

When lock bit 1 is programmed, the logic level at the $\overline{EA}$ pin is sampled and latched during reset. If the device is powered up without a reset, the latch initializes to a random value, and holds that value until reset is activated. It is necessary that the latched value of $\overline{EA}$ be in agreement with the current logic level at that pin in order for the device to function properly.

Lock Bit Protection Modes

Program Lock Bits				
	LB1	LB2	LB3	Protection Type
1	U	U	U	No program lock features
2	P	U	U	MOVC instructions executed from external program memory are disabled from fetching code bytes from internal memory, $\overline{EA}$ is sampled and latched on reset, and further programming of the Flash is disabled
3	P	P	U	Same as mode 2, also verify is disabled
4	P	P	P	Same as mode 3, also external execution is disabled

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Programming the Flash

The AT89C51 is normally shipped with the on-chip Flash memory array in the erased state (that is, contents = FFH) and ready to be programmed. The programming interface accepts either a high-voltage (12-volt) or a low-voltage (V_{CC}) program enable signal. The low-voltage programming mode provides a convenient way to program the AT89C51 inside the user's system, while the high-voltage programming mode is compatible with conventional third-party Flash or EPROM programmers.

The AT89C51 is shipped with either the high-voltage or low-voltage programming mode enabled. The respective top-side marking and device signature codes are listed in the following table.

	V_{PP} = 12V	V_{PP} = 5V
Top-Side Mark	AT89C51 xxxx yyww	AT89C51 xxxx-5 yyww
Signature	(030H) = 1EH (031H) = 51H (032H) =F FH	(030H) = 1EH (031H) = 51H (032H) = 05H

The AT89C51 code memory array is programmed byte-by-byte in either programming mode. *To program any non-blank byte in the on-chip Flash Memory, the entire memory must be erased using the Chip Erase Mode.*

Programming Algorithm: Before programming the AT89C51, the address, data and control signals should be set up according to the Flash programming mode table and Figure 3 and Figure 4. To program the AT89C51, take the following steps.

1. Input the desired memory location on the address lines.
2. Input the appropriate data byte on the data lines.
3. Activate the correct combination of control signals.
4. Raise $\overline{EA}/V_{PP}$ to 12V for the high-voltage programming mode.
5. Pulse ALE/$\overline{PROG}$ once to program a byte in the Flash array or the lock bits. The byte-write cycle is self-timed and typically takes no more than 1.5 ms. Repeat steps 1 through 5, changing the address and data for the entire array or until the end of the object file is reached.

$\overline{Data}$ Polling: The AT89C51 features $\overline{Data}$ Polling to indicate the end of a write cycle. During a write cycle, an attempted read of the last byte written will result in the complement of the written datum on PO.7. Once the write cycle has been completed, true data are valid on all outputs, and the next cycle may begin. $\overline{Data}$ Polling may begin any time after a write cycle has been initiated.

Ready/$\overline{Busy}$: The progress of byte programming can also be monitored by the RDY/$\overline{BSY}$ output signal. P3.4 is pulled low after ALE goes high during programming to indicate BUSY. P3.4 is pulled high again when programming is done to indicate READY.

Program Verify: If lock bits LB1 and LB2 have not been programmed, the programmed code data can be read back via the address and data lines for verification. The lock bits cannot be verified directly. Verification of the lock bits is achieved by observing that their features are enabled.

Chip Erase: The entire Flash array is erased electrically by using the proper combination of control signals and by holding ALE/$\overline{PROG}$ low for 10 ms. The code array is written with all "1"s. The chip erase operation must be executed before the code memory can be re-programmed.

Reading the Signature Bytes: The signature bytes are read by the same procedure as a normal verification of locations 030H, 031H, and 032H, except that P3.6 and P3.7 must be pulled to a logic low. The values returned are as follows.

(030H) = 1EH indicates manufactured by Atmel
(031H) = 51H indicates 89C51
(032H) = FFH indicates 12V programming
(032H) = 05H indicates 5V programming

Programming Interface

Every code byte in the Flash array can be written and the entire array can be erased by using the appropriate combination of control signals. The write operation cycle is self-timed and once initiated, will automatically time itself to completion.

All major programming vendors offer worldwide support for the Atmel microcontroller series. Please contact your local programming vendor for the appropriate software revision.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Flash Programming Modes

Mode		RST	PSEN	ALE/PROG	EA/V_{PP}	P2.6	P2.7	P3.6	P3.7
Write Code Data		H	L		H/12V	L	H	H	H
Read Code Data		H	L	H	H	L	L	H	H
Write Lock	Bit - 1	H	L		H/12V	H	H	H	H
	Bit - 2	H	L		H/12V	H	H	L	L
	Bit - 3	H	L		H/12V	H	L	H	L
Chip Erase		H	L	(1)	H/12V	H	L	L	L
Read Signature Byte		H	L	H	H	L	L	L	L

Note: 1. Chip Erase requires a 10 ms PROG pulse.

Figure 3. Programming the Flash

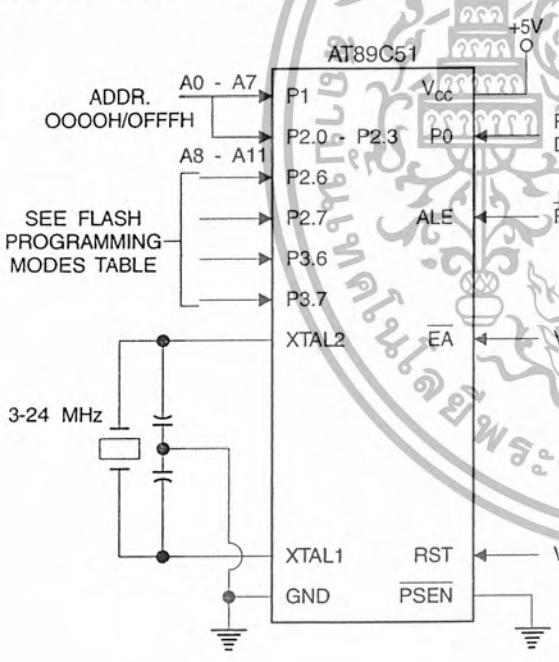
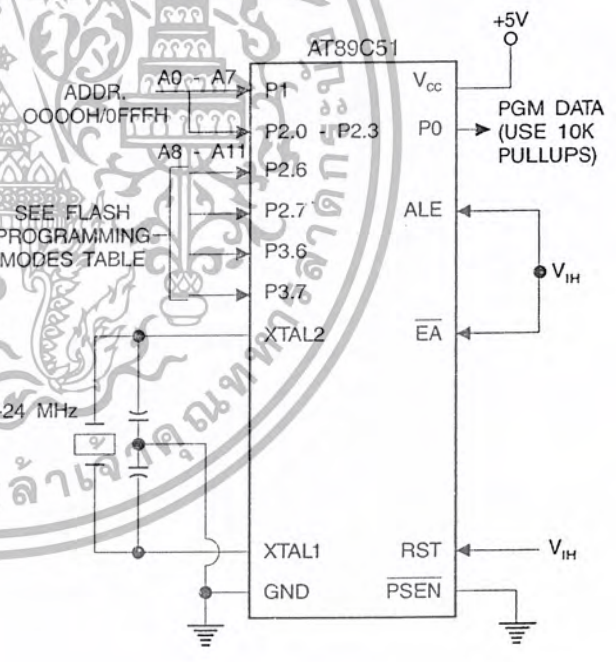
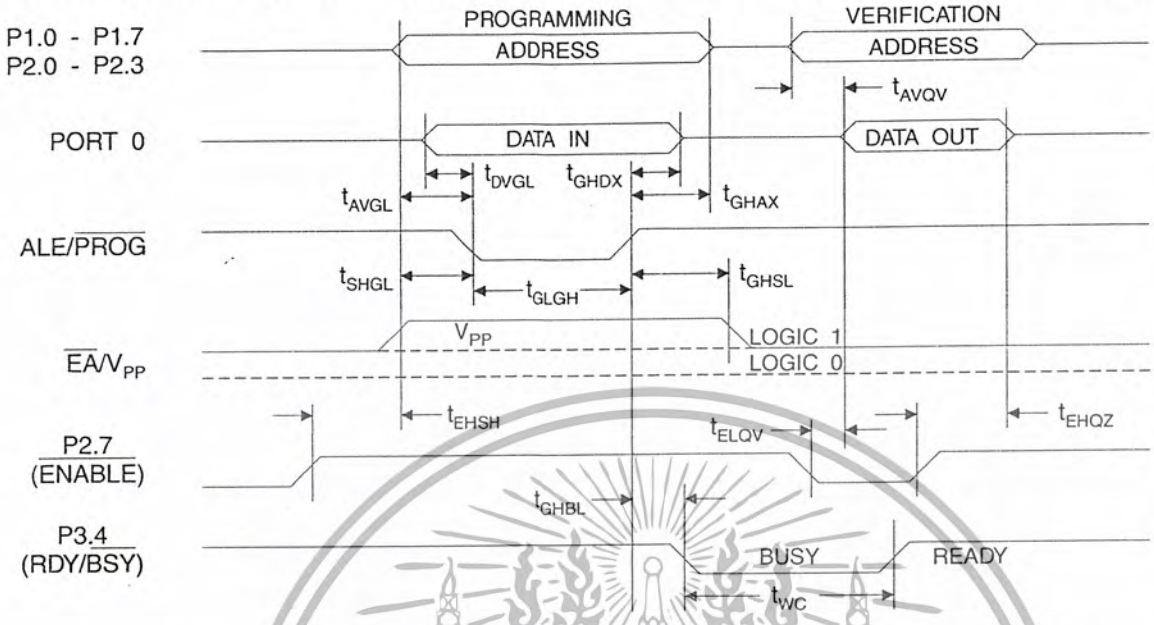


Figure 4. Verifying the Flash

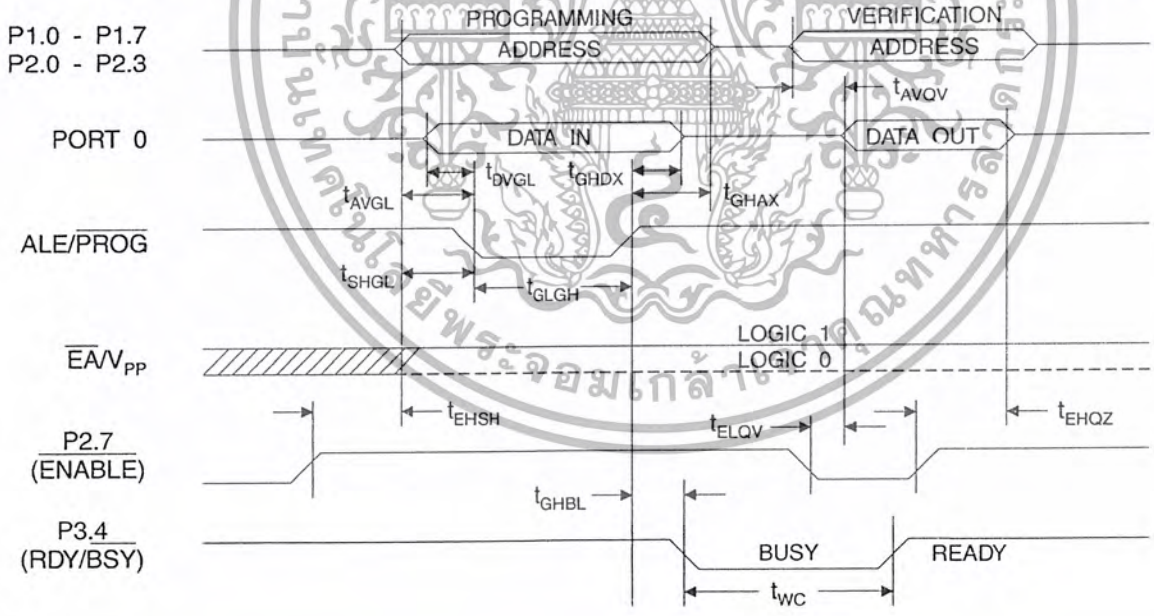


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Flash Programming and Verification Waveforms - High-voltage Mode (V_{PP} = 12V)



Flash Programming and Verification Waveforms - Low-voltage Mode (V_{PP} = 5V)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Flash Programming and Verification Characteristics

T_A = 0°C to 70°C, V_{CC} = 5.0 ± 10%

Symbol	Parameter	Min	Max	Units
V_{PP}[1]	Programming Enable Voltage	11.5	12.5	V
I_{PP}[1]	Programming Enable Current		1.0	mA
$1/t_{CLCL}$	Oscillator Frequency	3	24	MHz
t_{AVGL}	Address Setup to $\overline{PROG}$ Low	$48t_{CLCL}$		
t_{GHAX}	Address Hold After $\overline{PROG}$	$48t_{CLCL}$		
t_{DVGL}	Data Setup to $\overline{PROG}$ Low	$48t_{CLCL}$		
t_{GHDX}	Data Hold After $\overline{PROG}$	$48t_{CLCL}$		
t_{EHSH}	P2.7 ($\overline{ENABLE}$) High to V_{PP}	$48t_{CLCL}$		
t_{SHGL}	V_{PP} Setup to $\overline{PROG}$ Low	10		µs
t_{GHSL}[1]	V_{PP} Hold After $\overline{PROG}$	10		µs
t_{GLGH}	$\overline{PROG}$ Width	1	110	µs
t_{AVQV}	Address to Data Valid		$48t_{CLCL}$	
t_{ELQV}	$\overline{ENABLE}$ Low to Data Valid		$48t_{CLCL}$	
t_{EHQZ}	Data Float After $\overline{ENABLE}$	0	$48t_{CLCL}$	
t_{GHBL}	$\overline{PROG}$ High to $\overline{BUSY}$ Low		1.0	µs
t_{WC}	Byte Write Cycle Time		2.0	ms

Note: 1. Only used in 12-volt programming mode.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Absolute Maximum Ratings*

Operating Temperature	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-1.0V to +7.0V
Maximum Operating Voltage	6.6V
DC Output Current	15.0 mA

*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

DC Characteristics

T_A = -40°C to 85°C, V_{CC} = 5.0V ± 20% (unless otherwise noted)

Symbol	Parameter	Condition	Min	Max	Units
V_{IL}	Input Low-voltage	(Except $\overline{EA}$)	-0.5	0.2 V_{CC} - 0.1	V
V_{IL1}	Input Low-voltage ($\overline{EA}$)		-0.5	0.2 V_{CC} - 0.3	V
V_{IH}	Input High-voltage	(Except XTAL1, RST)	0.2 V_{CC} + 0.9	V_{CC} + 0.5	V
V_{IH1}	Input High-voltage	(XTAL1, RST)	0.7 V_{CC}	V_{CC} + 0.5	V
V_{OL}	Output Low-voltage[1] (Ports 1,2,3)	I_{OL} = 1.6 mA		0.45	V
V_{OL1}	Output Low-voltage[1] (Port 0, ALE, $\overline{PSEN}$)	I_{OL} = 3.2 mA		0.45	V
V_{OH}	Output High-voltage (Ports 1,2,3, ALE, $\overline{PSEN}$)	I_{OH} = -60 µA, V_{CC} = 5V ± 10%	2.4		V
		I_{OH} = -25 µA	0.75 V_{CC}		V
		I_{OH} = -10 µA	0.9 V_{CC}		V
V_{OH1}	Output High-voltage (Port 0 in External Bus Mode)	I_{OH} = -800 µA, V_{CC} = 5V ± 10%	2.4		V
		I_{OH} = -300 µA	0.75 V_{CC}		V
		I_{OH} = -80 µA	0.9 V_{CC}		V
I_{IL}	Logical 0 Input Current (Ports 1,2,3)	V_{IN} = 0.45V		-50	µA
I_{TL}	Logical 1 to 0 Transition Current (Ports 1,2,3)	V_{IN} = 2V, VCC = 5V ± 10%		-650	µA
I_{LI}	Input Leakage Current (Port 0, $\overline{EA}$)	0.45 < V_{IN} < V_{CC}		±10	µA
RRST	Reset Pull-down Resistor		50	300	KΩ
C_{IO}	Pin Capacitance	Test Freq. = 1 MHz, T_A = 25°C		10	pF
I_{CC}	Power Supply Current	Active Mode, 12 MHz		20	mA
		Idle Mode, 12 MHz		5	mA
	Power-down Mode[2]	V_{CC} = 6V		100	µA
		V_{CC} = 3V		40	µA

Notes:
1. Under steady state (non-transient) conditions, I_{OL} must be externally limited as follows:
   Maximum I_{OL} per port pin: 10 mA
   Maximum I_{OL} per 8-bit port: Port 0: 26 mA
   Ports 1, 2, 3: 15 mA
   Maximum total I_{OL} for all output pins: 71 mA
   If I_{OL} exceeds the test condition, V_{OL} may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
2. Minimum V_{CC} for Power-down is 2V.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AC Characteristics

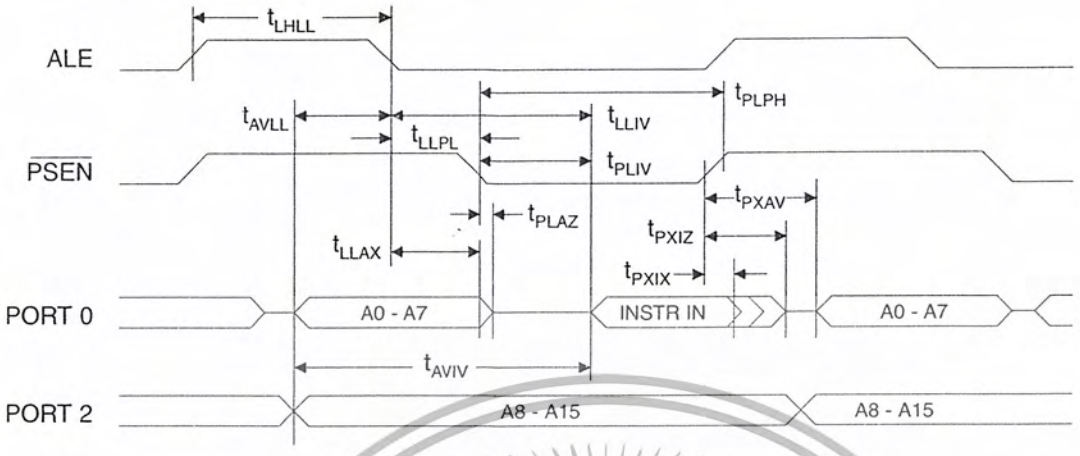
Under operating conditions, load capacitance for Port 0, ALE/$\overline{PROG}$, and $\overline{PSEN}$ = 100 pF; load capacitance for all other outputs = 80 pF.

External Program and Data Memory Characteristics

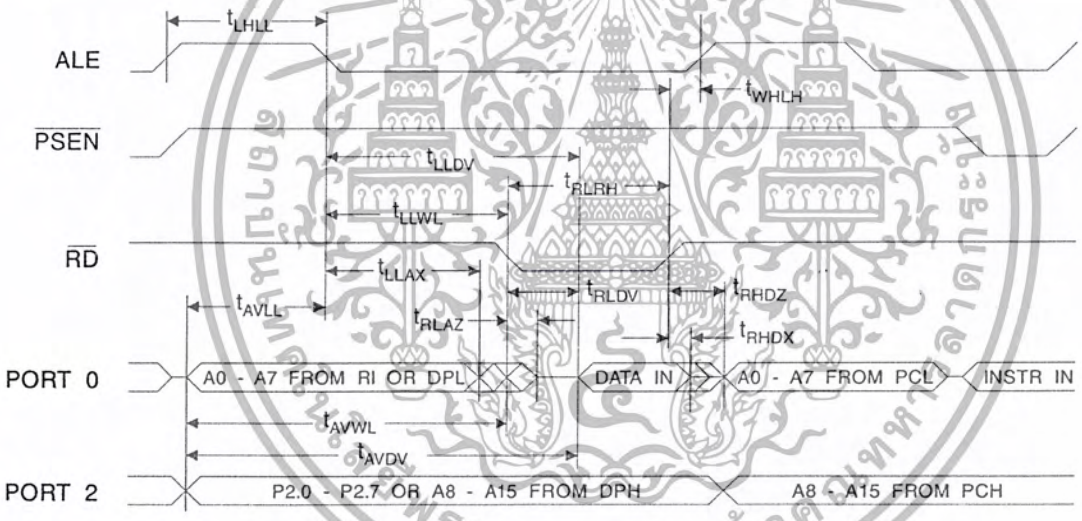
Symbol	Parameter	12 MHz Oscillator		16 to 24 MHz Oscillator		Units
		Min	Max	Min	Max	
$1/t_{CLCL}$	Oscillator Frequency			0	24	MHz
t_{LHLL}	ALE Pulse Width	127		$2t_{CLCL}$-40		ns
t_{AVLL}	Address Valid to ALE Low	43		t_{CLCL}-13		ns
t_{LLAX}	Address Hold After ALE Low	48		t_{CLCL}-20		ns
t_{LLIV}	ALE Low to Valid Instruction In		233		$4t_{CLCL}$-65	ns
t_{LLPL}	ALE Low to $\overline{PSEN}$ Low	43		t_{CLCL}-13		ns
t_{PLPH}	$\overline{PSEN}$ Pulse Width	205		$3t_{CLCL}$-20		ns
t_{PLIV}	$\overline{PSEN}$ Low to Valid Instruction In		145		$3t_{CLCL}$-45	ns
t_{PXIX}	Input Instruction Hold After $\overline{PSEN}$	0		0		ns
t_{PXIZ}	Input Instruction Float After $\overline{PSEN}$		59		t_{CLCL}-10	ns
t_{PXAV}	$\overline{PSEN}$ to Address Valid	75		t_{CLCL}-8		ns
t_{AVIV}	Address to Valid Instruction In		312		$5t_{CLCL}$-55	ns
t_{PLAZ}	$\overline{PSEN}$ Low to Address Float		10		10	ns
t_{RLRH}	$\overline{RD}$ Pulse Width	400		$6t_{CLCL}$-100		ns
t_{WLWH}	$\overline{WR}$ Pulse Width	400		$6t_{CLCL}$-100		ns
t_{RLDV}	$\overline{RD}$ Low to Valid Data In		252		$5t_{CLCL}$-90	ns
t_{RHDX}	Data Hold After $\overline{RD}$	0		0		ns
t_{RHDZ}	Data Float After $\overline{RD}$		97		$2t_{CLCL}$-28	ns
t_{LLDV}	ALE Low to Valid Data In		517		$8t_{CLCL}$-150	ns
t_{AVDV}	Address to Valid Data In		585		$9t_{CLCL}$-165	ns
t_{LLWL}	ALE Low to $\overline{RD}$ or $\overline{WR}$ Low	200	300	$3t_{CLCL}$-50	$3t_{CLCL}$+50	ns
t_{AVWL}	Address to $\overline{RD}$ or $\overline{WR}$ Low	203		$4t_{CLCL}$-75		ns
t_{QVWX}	Data Valid to $\overline{WR}$ Transition	23		t_{CLCL}-20		ns
t_{QVWH}	Data Valid to $\overline{WR}$ High	433		$7t_{CLCL}$-120		ns
t_{WHQX}	Data Hold After $\overline{WR}$	33		t_{CLCL}-20		ns
t_{RLAZ}	$\overline{RD}$ Low to Address Float		0		0	ns
t_{WHLH}	$\overline{RD}$ or $\overline{WR}$ High to ALE High	43	123	t_{CLCL}-20	t_{CLCL}+25	ns

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

External Program Memory Read Cycle

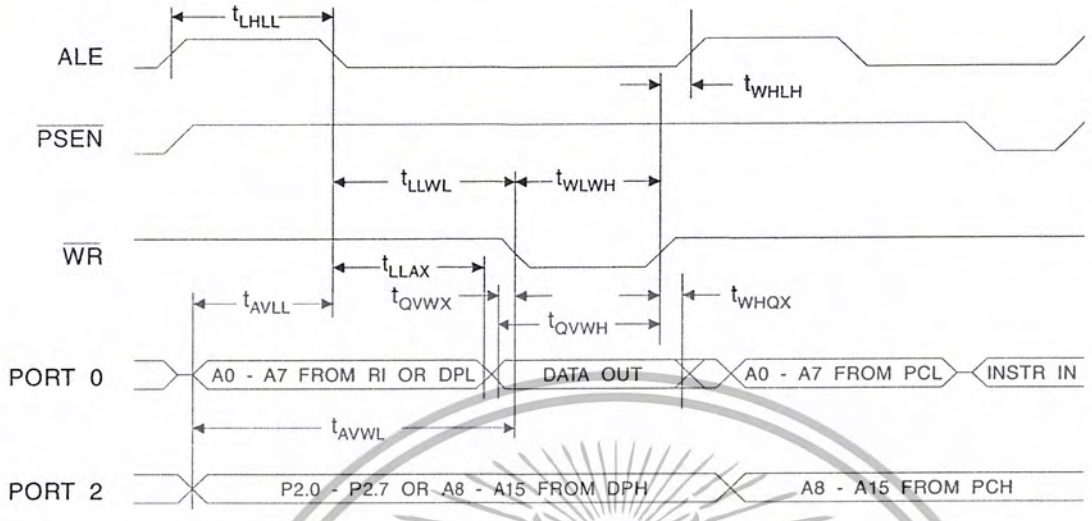


External Data Memory Read Cycle

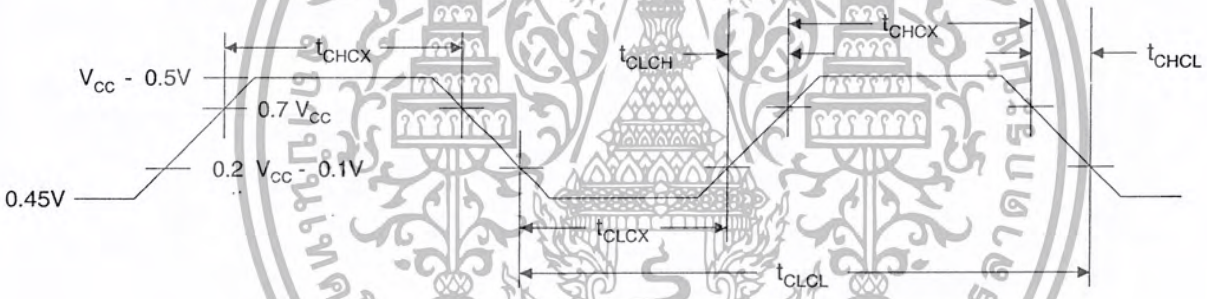


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

External Data Memory Write Cycle



External Clock Drive Waveforms



External Clock Drive

Symbol	Parameter	Min	Max	Units
$1/t_{CLCL}$	Oscillator Frequency	0	24	MHz
t_{CLCL}	Clock Period	41.6		ns
t_{CHCX}	High Time	15		ns
t_{CLCX}	Low Time	15		ns
t_{CLCH}	Rise Time		20	ns
t_{CHCL}	Fall Time		20	ns

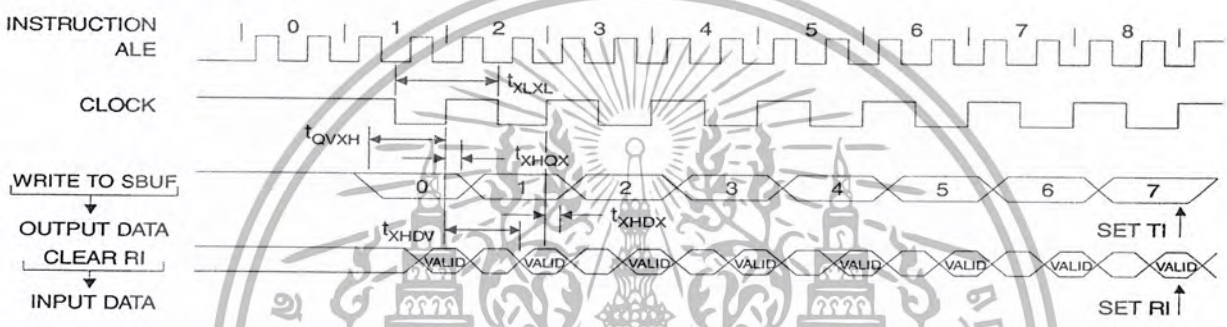
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Serial Port Timing: Shift Register Mode Test Conditions

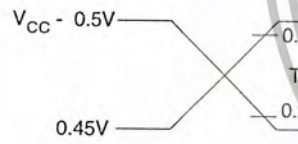
(V_{CC} = 5.0 V ± 20%; Load Capacitance = 80 pF)

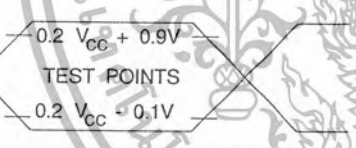
Symbol	Parameter	12 MHz Osc		Variable Oscillator		Units
		Min	Max	Min	Max	
t_{XLXL}	Serial Port Clock Cycle Time	1.0		$12t_{CLCL}$		µs
t_{QVXH}	Output Data Setup to Clock Rising Edge	700		$10t_{CLCL}$-133		ns
t_{XHQX}	Output Data Hold After Clock Rising Edge	50		$2t_{CLCL}$-117		ns
t_{XHDX}	Input Data Hold After Clock Rising Edge	0		0		ns
t_{XHDV}	Clock Rising Edge to Input Data Valid		700		$10t_{CLCL}$-133	ns

Shift Register Mode Timing Waveforms



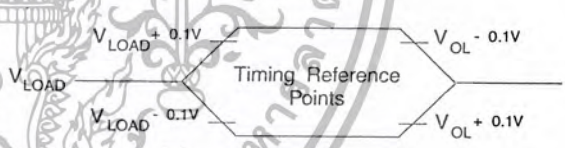
AC Testing Input/Output Waveforms[1]





Note: 1. AC Inputs during testing are driven at V_{CC} - 0.5V for a logic 1 and 0.45V for a logic 0. Timing measurements are made at V_{IH} min. for a logic 1 and V_{IL} max. for a logic 0.

Float Waveforms[1]



Note: 1. For timing purposes, a port pin is no longer floating when a 100 mV change from load voltage occurs. A port pin begins to float when 100 mV change from the loaded V_{OH}/V_{OL} level occurs.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Ordering Information

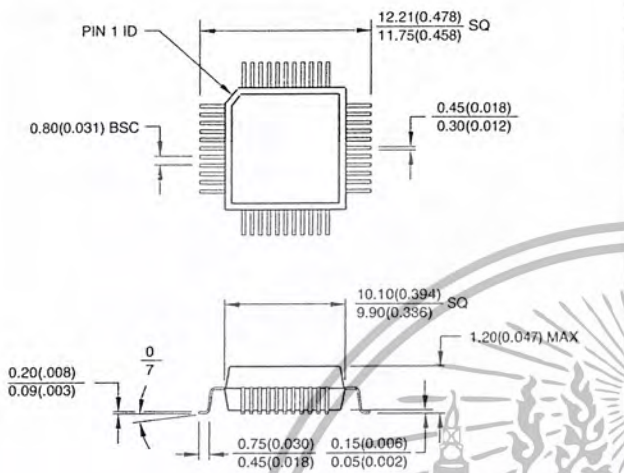
Speed (MHz)	Power Supply	Ordering Code	Package	Operation Range
12	5V ± 20%	AT89C51-12AC	44A	Commercial (0°C to 70°C)
		AT89C51-12JC	44J	
		AT89C51-12PC	40P6	
		AT89C51-12QC	44Q	
		AT89C51-12AI	44A	Industrial (-40°C to 85°C)
		AT89C51-12JI	44J	
		AT89C51-12PI	40P6	
		AT89C51-12QI	44Q	
16	5V ± 20%	AT89C51-16AC	44A	Commercial (0°C to 70°C)
		AT89C51-16JC	44J	
		AT89C51-16PC	40P6	
		AT89C51-16QC	44Q	
		AT89C51-16AI	44A	Industrial (-40°C to 85°C)
		AT89C51-16JI	44J	
		AT89C51-16PI	40P6	
		AT89C51-16QI	44Q	
20	5V ± 20%	AT89C51-20AC	44A	Commercial (0°C to 70°C)
		AT89C51-20JC	44J	
		AT89C51-20PC	40P6	
		AT89C51-20QC	44Q	
		AT89C51-20AI	44A	Industrial (-40°C to 85°C)
		AT89C51-20JI	44J	
		AT89C51-20PI	40P6	
		AT89C51-20QI	44Q	
24	5V ± 20%	AT89C51-24AC	44A	Commercial (0°C to 70°C)
		AT89C51-24JC	44J	
		AT89C51-24PC	40P6	
		AT89C51-24QC	44Q	
		AT89C51-24AI	44A	Industrial (-40°C to 85°C)
		AT89C51-24JI	44J	
		AT89C51-24PI	40P6	
		AT89C51-24QI	44Q	

Package Type	
44A	44-lead, Thin Plastic Gull Wing Quad Flatpack (TQFP)
44J	44-lead, Plastic J-leaded Chip Carrier (PLCC)
40P6	40-lead, 0.600" Wide, Plastic Dual Inline Package (PDIP)
44Q	44-lead, Plastic Gull Wing Quad Flatpack (PQFP)

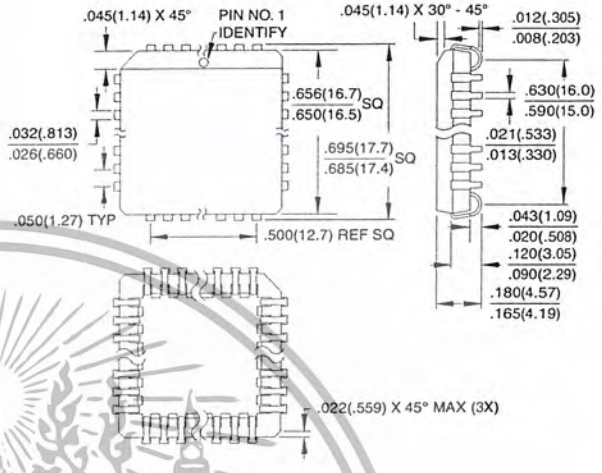
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Packaging Information

44A, 44-lead, Thin (1.0 mm) Plastic Gull Wing Quad Flatpack (TQFP)
Dimensions in Millimeters and (Inches)*
JEDEC STANDARD MS-026 ACB

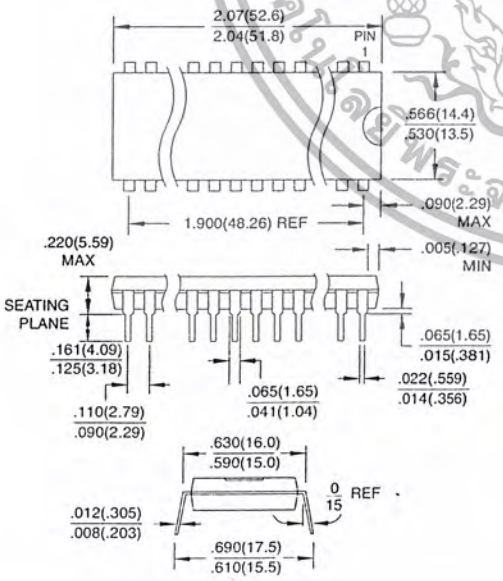


44J, 44-lead, Plastic J-leaded Chip Carrier (PLCC)
Dimensions in Inches and (Millimeters)
JEDEC STANDARD MS-018 AC

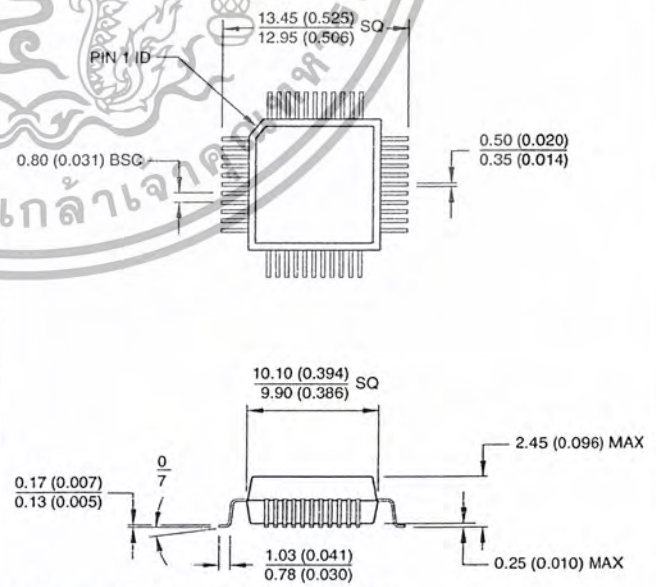


Controlling dimension: millimeters

40P6, 40-lead, 0.600" Wide, Plastic Dual Inline Package (PDIP)
Dimensions in Inches and (Millimeters)



44Q, 44-lead, Plastic Quad Flat Package (PQFP)
Dimensions in Millimeters and (Inches)*
JEDEC STANDARD MS-022 AB



Controlling dimension: millimeters

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Atmel Headquarters

Corporate Headquarters
2325 Orchard Parkway
San Jose, CA 95131
TEL (408) 441-0311
FAX (408) 487-2600

Europe
Atmel U.K., Ltd.
Coliseum Business Centre
Riverside Way
Camberley, Surrey GU15 3YL
England
TEL (44) 1276-686-677
FAX (44) 1276-686-697

Asia
Atmel Asia, Ltd.
Room 1219
Chinachem Golden Plaza
77 Mody Road Tsimhatsui
East Kowloon
Hong Kong
TEL (852) 2721-9778
FAX (852) 2722-1369

Japan
Atmel Japan K.K.
9F, Tonetsu Shinkawa Bldg.
1-24-8 Shinkawa
Chuo-ku, Tokyo 104-0033
Japan
TEL (81) 3-3523-3551
FAX (81) 3-3523-7581

Atmel Operations

Atmel Colorado Springs
1150 E. Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906
TEL (719) 576-3300
FAX (719) 540-1759

Atmel Rousset
Zone Industrielle
13106 Rousset Cedex
France
TEL (33) 4-4253-6000
FAX (33) 4-4253-6001

Fax-on-Demand
North America:
1-(800) 292-8635
International:
1-(408) 441-0732

e-mail
literature@atmel.com

Web Site
http://www.atmel.com

BBS
1-(408) 436-4309

© Atmel Corporation 2000.
Atmel Corporation makes no warranty for the use of its products, other than those expressly contained in the Company's standard warranty which is detailed in Atmel's Terms and Conditions located on the Company's web site. The Company assumes no responsibility for any errors which may appear in this document, reserves the right to change devices or specifications detailed herein at any time without notice, and does not make any commitment to update the information contained herein. No licenses to patents or other intellectual property of Atmel are granted by the Company in connection with the sale of Atmel products, expressly or by implication. Atmel's products are not authorized for use as critical components in life support devices or systems.

Marks bearing ® and/or ™ are registered trademarks and trademarks of Atmel Corporation.

Terms and product names in this document may be trademarks of others.

Printed on recycled paper.

0265G–02/00/xM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

1. วรพจน์ กรแก้ววัฒนกุล และ ชัยวัฒน์ ลิ้มพรจิตรวิไล. เรียนรู้และปฏิบัติการไมโครคอนโทรลเลอร์ MCS-51 Flash Microcontroller :บริษัทอินโนเวติฟ เอ็กเพอริเมนต์ จำกัด
2. รศ.สมยศ จุณณะปิยะ. การประยุกต์ใช้งานไมโครคอนโทรลเลอร์. พิมพ์ครั้งที่5. กรุงเทพมหานคร: สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ,2546

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้