สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เครื่องวัดและบันทึกข้อมูลแบบไร้สาย

WIRELESS DATA LOGGER

T 050199

โดย

นายธนาณัติ กล้าหาญ

นายธีระ ทรัพย์เพลง

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมระบบควบคุม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2545

เลขหมู่.....
เลขทะเบียน 50199
วัน,เดือน,ปี 27 เม.ย. 2547

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องวัดและบันทึกข้อมูลแบบไร้สาย

WIRELESS DATA LOGGER

โดย

นายธนาณัติ กล้าหาญ

นายธีระ ทรัพย์เพลง

อาจารย์ที่ปรึกษา

ผศ.วรพงศ์ ตั้งศรีรัตน์

ปริญญานิพนธ์สำหรับปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมระบบควบคุม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2545

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2545

ภาควิชาวิศวกรรมระบบควบคุม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เครื่องวัดและบันทึกข้อมูลแบบไร้สาย

ผู้จัดทำ

1.นายธนาณัติ	กล้าหาญ	รหัสประจำตัว 43015303
2. นายธีระ	ทรัพย์เพลง	รหัสประจำตัว 43015305

.....อาจารย์ที่ปรึกษา

(ผศ.วรพงศ์ ตั้งศรีรัตน์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องวัดและบันทึกข้อมูลแบบไร้สาย

นายธนาณัติ กล้าหาญ
นายธีระ ทรัพย์เพลง
ผศ.วรพงศ์ ตั้งศรีรัตน์ อาจารย์ที่ปรึกษา
ปีการศึกษา 2545

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้เป็นการนำเสนอเครื่องมือสำหรับใช้ในการตรวจวัดและบันทึกข้อมูลแบบไร้สาย หรือ ดาต้าลอกเกอร์แบบไร้สาย (Wireless Data Logger) ซึ่งเป็นอุปกรณ์ที่นิยมใช้กันอย่างแพร่หลายในกระบวนการอุตสาหกรรมต่างๆ เครื่องมือวัดและบันทึกข้อมูลที่นำเสนอนี้สามารถทำการตรวจวัดและบันทึกข้อมูลของอุณหภูมิและสัญญาณทางไฟฟ้าที่เปลี่ยนไป โดยแสดงผลแบบกราฟผ่านทางหน้าจอคอมพิวเตอร์ ตลอดจนสามารถทำการบันทึกผลข้อมูลที่ได้จากการตรวจวัด โดยที่ไม่จำเป็นต้องเปิดเครื่องคอมพิวเตอร์ไว้ตลอดเวลา การควบคุมและประมวลผลจากไมโครคอนโทรลเลอร์MCS- 51 ซึ่งมีการติดต่อสื่อสารข้อมูลกับคอมพิวเตอร์ผ่านทางคลื่นความถี่วิทยุ 433 MHz

Abstract

This project presents the wireless data logger, which is widely used in many industrial process system. The proposed data logger can be measured and record a chang of the process temperature and process variables in the term of electrical signals. By based on the co-operation between the microcontroller MCS-51 and the personal computer via the radio fequency of 433 MHz, this wireless data logger can graphically be display on the computer's monitor. Although the computer is not operated, its operation mode is also achieved.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อ	
สารบัญรูปภาพ	
สารบัญตาราง	
บทที่ 1 บทนำ	1
1.1 ความสำคัญและความเป็นมาของโครงงาน	1
1.2 วัตถุประสงค์	1
1.3 ขอบเขตโครงงาน	2
บทที่ 2 ทฤษฎีและหลักการเบื้องต้น	3
2.1 ความรู้เบื้องต้นเกี่ยวกับ I^2C	3
2.2 คุณสมบัติโดยทั่วไปของบัส I^2C	4
2.3 หลักการทำงานของบัส I^2C	5
2.4 การทำงานบนบัส I^2C	7
2.5 อุปกรณ์ที่ใช้การเชื่อมต่อแบบ I^2C	9
2.6 การเขียนโปรแกรมติดต่อบัส I^2C	11
2.7 รีจิสเตอร์ควบคุมการทำงานของพอร์ตอนุกรม SCON	12
2.8 การกำหนดอัตรารับส่ง	16
2.9 คุณสมบัติของไมโครคอนโทรลเลอร์	17
2.10 โครงสร้างภายในของ MCS-51	18
2.11 พอร์ตของ 8051	21
2.12 วงจรสร้างสัญญาณนาฬิกาของ 8051	26
2.13 ผังสัญญาณเวลาซีพียู	27
2.14 การแบ่งประเภทของหน่วยความจำ	30
บทที่ 3 การออกแบบและการทำงานของเครื่องวัดและบันทึกข้อมูลแบบไร้สาย	32
3.1 หลักการทำงานโดยรวม	32
3.2 บล็อคไดอะแกรมส่วนรับข้อมูลและสั่งงาน	32
3.3 บล็อคไดอะแกรมของส่วนเครื่องมือวัด	33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4 การทดลองและผลการทดลอง 40

4.1 โปรแกรมควบคุมการทำงาน 41

4.2 การใช้งานโปรแกรมการทำงาน 42

บทที่ 5 บทวิจารณ์และสรุป 47

5.1 บทวิจารณ์และสรุป 47

5.2 สรุปผลการทำงาน 47

ภาคผนวก

ภาคผนวก ก โปรแกรมควบคุมเครื่องวัดและบันทึกข้อมูลแบบไร้สาย

ภาคผนวก ข คู่มือไอซี (Data sheet)

- DS 1307 ข-1
- 24LC 256 ข-5
- DS 1620 ข-9
- LTC 1298 ข-13
- BIM 2 – 433 ข-17
- 74 C 14 ข-25
- 4093 B ข-26

ภาคผนวก ค คู่มือการใช้งานเครื่องวัดและบันทึกข้อมูลแบบไร้สาย

กิตติกรรมประกาศ

เอกสารอ้างอิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

หน้า

รูปที่ 2.1 ผังแสดงการเชื่อมต่อของอุปกรณ์ต่างๆบนระบบบัส I^2C 3

รูปที่ 2.2.1 การต่อตัวต้านทานพูลอัพบนสายสัญญาณในระบบบัส I^2C 4

รูปที่ 2.2.2 การต่อตัวต้านทาน Rs เพื่อลดสัญญาณรบกวนขนาดใหญ่ที่อาจเข้ามาในบัส I^2C 5

รูปที่ 2.3 ไดอะแกรมเวลาแสดงสถานะต่างๆในบัส I^2C 7

รูปที่ 2.4 รูปแบบของข้อมูลกำหนดแอดเอรสที่ใช้ในการอ้างถึงแบบ 7 บิต 8

รูปที่ 2.5 วงจรตัวอย่างการต่อ ไมโครคอนโทรลเลอร์ MCS-51 กับอุปกรณ์ระบบบัส 10

รูปที่ 2.10.1 8051 บล็อกไดอะแกรมของ MCS-51 19

รูปที่2.10.2 ตำแหน่งของรีจิสแตอร์ต่าง ๆ 20

รูปที่ 2.10.3 การจัดวางขาของ MCS-51 20

รูปที่2.11.1 โครงสร้างของพอร์ท 0 (บิต) 21

รูปที่ 2.11.2 โครงสร้างของพอร์ท 1 (บิต) 22

รูปที่ 2.11.3 โครงสร้างของพอร์ท 2 (บิต) 23

รูปที่2.11.4 โครงสร้างของพอร์ท 3 (บิต) 24

รูปที่2.11.5 การต่อขารีเซ็ทให้กับ 8051 25

รูปที่2.12 วงจรสร้างสัญญาณนาฬิกาของ 8051 26

รูปที่ 2.13.1 ผังเวลาการทำงานของคำสั่ง 27

รูปที่ 213.2 แสดงผังเวลาการติดต่อกับหน่วยความจำภายนอก 29

รูปที่ 2.14 (ก) ผังเก็บความจำสำหรับโปรแกรมสำหรับเบอร์ 8051 30

(ข) ผังเก็บความจำสำหรับโปรแกรมสำหรับเบอร์ 8052 30

(ค) ผังหน่วยความจำสำหรับ Data Memory เบอร์ 8051 30

รูปที่ 3.2 บล็อคไดอะแกรมของส่วนรับข้อมูลและสั่งงาน 33

รูปที่ 3.3.1 บล็อคไดอะแกรมของส่วนตรวจวัด 35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.3.2	วงจรส่วนตรวจวัด	36
รูปที่ 3.3.3	การวางอุปกรณ์ของส่วนตรวจวัด	37
รูปที่ 3.3.4	วงจร Transceiver	38
รูปที่ 3.3.5	วงจรรับข้อมูลและสั่งงาน	39
รูปที่ 4.2	หน้าต่างการทำงาน	42
รูปที่ 4.2.1	SET MODE	43
รูปที่ 4.2.2	EEPROM MODE	44
รูปที่ 4.2.3	MENU MODE	44
รูปที่4.3.4	MEASURE MODE	45
รูปที่ 4.2.5	RECORD MODE	45
รูปที่ 4.2.6	CALL MODE	46

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความสำคัญและความเป็นมาของโครงงาน

ในปัจจุบันนี้ไม่ว่าจะเป็นงานด้านอุตสาหกรรมหรือแม้แต่ในชีวิตประจำวันของมนุษย์จะมีการสร้างเครื่องอำนวยความสะดวกขึ้น เพื่อตอบสนองความต้องการของมนุษย์ ดาต้าลอกเกอร์ (Data Logger) หรือเครื่องมือสำหรับตรวจวัดและบันทึกผลข้อมูล ก็เป็นส่วนหนึ่งในจำนวนสิ่งอำนวยความสะดวกทั้งหลาย ซึ่งสามารถดูผลการเปลี่ยนแปลงต่างๆของการวัด และสามารถสั่งงานจากคอมพิวเตอร์ได้เลยทันที โดยเราไม่ต้องเดินสายระหว่างตัวเครื่องมือวัดกับเครื่องคอมพิวเตอร์ให้ยุ่งยาก เพราะเราจะใช้การติดต่อทางความถี่แทน จึงทำให้สะดวกและง่ายต่อการติดตั้ง โดยในที่นี้เราใช้ในการวัดอุณหภูมิและการเปลี่ยนแปลงสัญญาณทางไฟฟ้าและสามารถแสดงออกมาเป็นกราฟ ให้ดูผลการเปลี่ยนแปลงต่างๆได้ชัดเจนขึ้น เพราะในบางครั้งจำเป็นต้องเก็บข้อมูลของอุณหภูมิและข้อมูลจากตัววัดสัญญาณต่างๆในเวลาเดียวกัน เพื่อนำข้อมูลเหล่านั้นไปวิเคราะห์หาความสัมพันธ์กันต่อไป

1.2 วัตถุประสงค์

1. เพื่อศึกษาโครงสร้างการทำงานตลอดจนคุณลักษณะต่าง ๆ ของ MCS-51
2. เพื่อศึกษาและออกแบบการอินเตอร์เฟสผ่านทางพอร์ตอนุกรม RS232 ของคอมพิวเตอร์
3. เพื่อศึกษาโปรแกรมควบคุมการทำงานของเครื่องตัวรับและส่งข้อมูล
4. สามารถรับส่งควบคุมโหมดการวัดได้โดยผ่านความถี่
5. สามารถแสดงผลการวัดทางอุณหภูมิและค่าสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตโครงงาน

1. สามารถวัดและบันทึกข้อมูลของค่าสัญญาณจากตัวตรวจวัดสัญญาณต่างๆ ซึ่งทำงานแบบเวลาจริง (Real-time) โดยระบุวันที่และเวลาขณะที่ทำการวัดด้วย
2. การติดต่อรับส่งข้อมูลแบบไร้สายโดยผ่านทางคลื่นความถี่ UHF 433 MHz
3. สามารถควบคุมและแสดงผลข้อมูลผ่านทางคอมพิวเตอร์ และสามารถเรียกดูข้อมูลที่บันทึกไว้โดยใช้โปรแกรมเทอร์มินอลทั่วไปได้
4. การควบคุมโดยเครื่องคอมพิวเตอร์จะผ่านทางพอร์ทเครื่องพิมพ์
5. สามารถเก็บข้อมูลและนำไปดาวน์โหลดดูได้ภายหลัง ตลอดจนสามารถนำเอาข้อมูลที่ได้ไปแสดงผลพล็อตกราฟโดยใช้โปรแกรม Microsoft Excell ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการเบื้องต้น

2.1 ความรู้เบื้องต้นเกี่ยวกับ I^2C

I^2Cย่อมาจาก Inter-IC Communication หมายถึงการติดต่อสื่อสารระหว่างไอซีโดยบัส I^2C ได้รับการพัฒนาขึ้นโดยฟิลิปส์ด้วยจุดมุ่งหมายหลักคือ ต้องการให้ไอซีหรือโมดูลสามารถติดต่อสั่งงานและควบคุมภายใต้สายสัญญาณ 2 เส้นเส้นหนึ่งสายข้อมูลอีกเส้นหนึ่งสายสัญญาณนาฬิกาที่ใช้ในการกำหนดจังหวะการทำงาน การต่อร่วมกันของอุปกรณ์บนบัส I^2C ทำได้ง่ายมากเพียงต่อสายข้อมูลและสายสัญญาณนาฬิกาของอุปกรณ์แต่ละตัวขนานหรือพ่วงกันไปส่วนการกำหนดแอดเดรสหรือตำแหน่งสำหรับติดต่ออุปกรณ์แต่ละตัว จะใช้รหัสข้อมูลและการกำหนดสภาวะลอจิกที่ขาแอดเดรสของอุปกรณแต่ละตัว สายข้อมูลบนบัส I^2C มีชื่อเรียกอย่างเป็นทางการว่าสายข้อมูลอนุกรมหรือ SDA (Serial Data Line) ส่วนสายสัญญาณนาฬิกามีชื่อเรียกว่าสายสัญญาณนาฬิกาอนุกรมหรือ SCL (Serial Clock Line) ในการอธิบายต่อไปนี้จะเรียกสายสัญญาณทั้งสองว่า สาย SDL และ สาย SCL

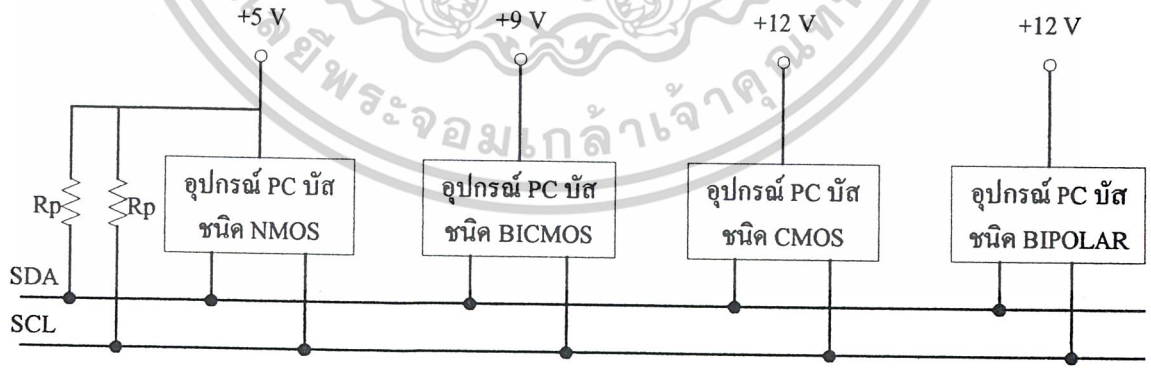
รูปที่ 2.1 ผังแสดงการเชื่อมต่อของอุปกรณ์ต่างๆบนระบบบัส I^2C

ในรูปที่ 2.1 แสดงผังการเชื่อมต่ออุปกรณ์ต่างๆบนบัส I^2C จะเห็นได้ว่าอุปกรณที่ทำการเชื่อมต่อบนบัส I^2C มีหลากหลาย ไม่ว่าจะเป็นไอซีขยายพอร์ตอินพุตเอาท์พุต (I\O Expander) ,ไอซีแปลงสัญญาณอนาล็อกเป็นดิจิตอล (ADC) และแปลงสัญญาณดิจิตอลเป็นอนาล็อก (DAC) ,ไอซีรีลไทม์ล็อก (RTC)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 คุณสมบัติโดยทั่วไปของบัส I^2C

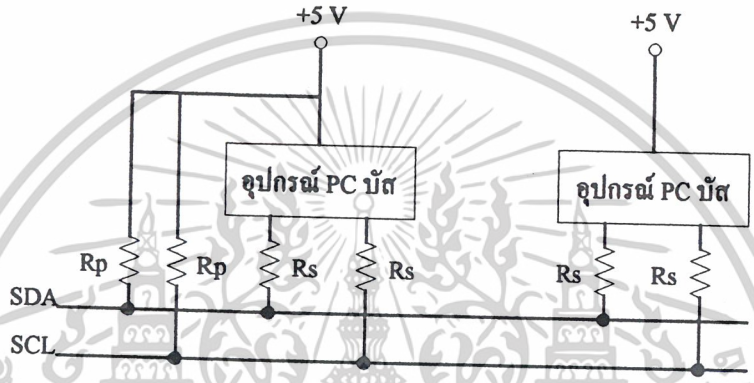
สาย SDA และ SCL เป็นสายสัญญาณ 2 ทิศทาง (bi-directional line) ต้องมีการต่อตัวต้านทานพูลอัพกับแรงดัน +5 V ไว้ตลอดเวลาเพื่อให้สายมีสถานะลอจิกสูงในขณะที่ไม่มีการติดต่อใช้งานทั้งยังช่วยในการป้องกันสัญญาณรบกวนที่อาจมีเข้ามาในสายสัญญาณทั้งสองวงจรเอาท์พุตของอุปกรณ์ที่ต่ออยู่บนบัส I^2C ต้องมีลักษณะเป็นวงจรเดรนเปิด (open-drain) หรือคอลเล็กเตอร์เปิด (open-collector) อัตราการถ่ายทอดข้อมูลบนบัส I^2C สูงถึง 100 กิโลบิตต่อวินาทีในโหมดปกติ (standard mode) และสูงถึง 400 กิโลบิตต่อวินาทีในโหมดความเร็วสูง (fast mode) อุปกรณ์ที่ต่อร่วมอยู่บนบัส I^2Cจะต้องมีค่าความจุไฟฟ้ารวมที่เกิดขึ้นระหว่างสาย SDA และ SCL ไม่เกิน 400pF การเข้าถึงอุปกรณ์บนบัส I^2C ใช้ข้อมูลสำหรับการเข้าถึง 2 ค่าคือ 7 บิต (7-bit addressing) หรือ 10 บิต (10-bit addressing) ข้อดีอีกประการหนึ่งของบัส I^2C คือสามารถเชื่อมต่ออุปกรณ์ที่ใช้ไฟเลี้ยงไม่เท่ากันให้สามารถติดต่อสื่อสารกันได้โดยอุปกรณ์บนบัส I^2C ตัวหนึ่งอาจใช้ไฟเลี้ยง +5V ในขณะที่อีกตัวหนึ่งใช้ไฟเลี้ยง +12V การต่อร่วมกันบน I^2C สามารถกระทำได้ในลักษณะเดียวกับกรณีที่อุปกรณ์ทั้งสองใช้ไฟเลี้ยงต่างกัน กล่าวคือให้ต่อสาย SDA และ SCL ของอุปกรณ์แต่ละตัวเข้าด้วยกัน และต้องต่อตัวต้านทานพูลอัพ (Rp) เข้ากับแรงดัน +5V ไว้เสมอ ดังแสดงในรูปที่ 2.2.1



รูปที่ 2.2.1 การต่อตัวต้านทานพูลอัพบนสายสัญญาณในระบบบัส I^2C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในกรณีที่อาจจะมีแรงดันไฟกระชากขนาดใหญ่ ปะปนเข้ามาในบัส I^2C ที่ขา SDA และ SCL ของอุปกรณ์แต่ละตัวต้องต่อตัวต้านทานอนุกรมกับขา SDA และ SCL เรียกว่า Rsก่อนต่อเข้าสู่บัส I^2C ดังแสดงในรูปที่ 2.2.2



รูปที่ 2.2.2 การต่อตัวต้านทาน Rs เพื่อลดสัญญาณรบกวนขนาดใหญ่ที่อาจเข้ามาในบัส I^2C

2.3 หลักการทำงานของบัส I^2C

บัส I^2C ประกอบด้วยสายสัญญาณ 2 เส้น ดังที่ได้กล่าวมาแล้วคือ SDA และ SCL อุปกรณ์ที่ต่อพ่วงบนบัสสมารถมีได้มากมาย ดังนั้นจึงมีการกำหนดรูปแบบของการติดต่อบนบัสหรือเรียกว่าโปรโต-คอล (protocol) เพื่อให้ผู้ใช้งานทราบว่า ขณะนี้มีอุปกรณ์ใดติดต่อกันอยู่ และอุปกรณ์ตัวใดเป็นตัวรับหรือตัวส่ง ต่อไปนี้จะขออธิบายลักษณะ หน้าที่และนิยามของอุปกรณ์ที่ต่ออยู่บนบัสฯ อุปกรณ์ที่เป็นผู้สร้างข้อมูลหรือส่งข้อมูล เรียกว่าตัวส่ง (transmitter) อุปกรณ์ที่เป็นผู้รับข้อมูลเรียกว่าตัวรับ (receiver) ในอุปกรณ์บนบัส I^2Cสามารถเป็นได้ทั้งตังรับและตัวส่งบางอุปกรณ์ทำหน้าที่เป็นตังรับอย่างเดียว จะไม่มีอุปกรณ์ใดบนบัส I^2C ที่ทำหน้าที่เป็นตัวส่งเพียงอย่างเดียวอุปกรณ์ที่ทำหน้าควบคุมจังหวะการติดต่อบนบัส I^2C เรียกว่า มาสเตอร์ (master) อุปกรณ์ที่ถูกควบคุมหรืออุปกรณ์ที่ต่อพ่วงเข้าไปบนบัส I^2C เรียกว่า สเลฟ (slave)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อกำหนดที่สำคัญ 2 ปะการสำคัญของการติดต่อบนบัส I^2C คือ

(1) การถ่ายทอดข้อมูลจะเกิดขึ้นได้เมื่อบัสว่างเท่านั้น

(2) ในระหว่างการถ่ายทอดข้อมูล เมื่อใดก็ตามที่สาย SCL มีสถานะเป็นลอจิกสูง สายข้อมูลจะรักษาข้อมูลไว้ อย่าให้เกิดการเปลี่ยนแปลงขึ้นเด็ดขาด มิฉะนั้น สัญญาณที่เกิดขึ้นจะได้รับการแปลความหมายเป็นสัญญาณควบคุมแทน

สภาวะที่เกิดขึ้นบนบัส I^2C มีด้วยกัน 5 สภาวะ ดังนี้

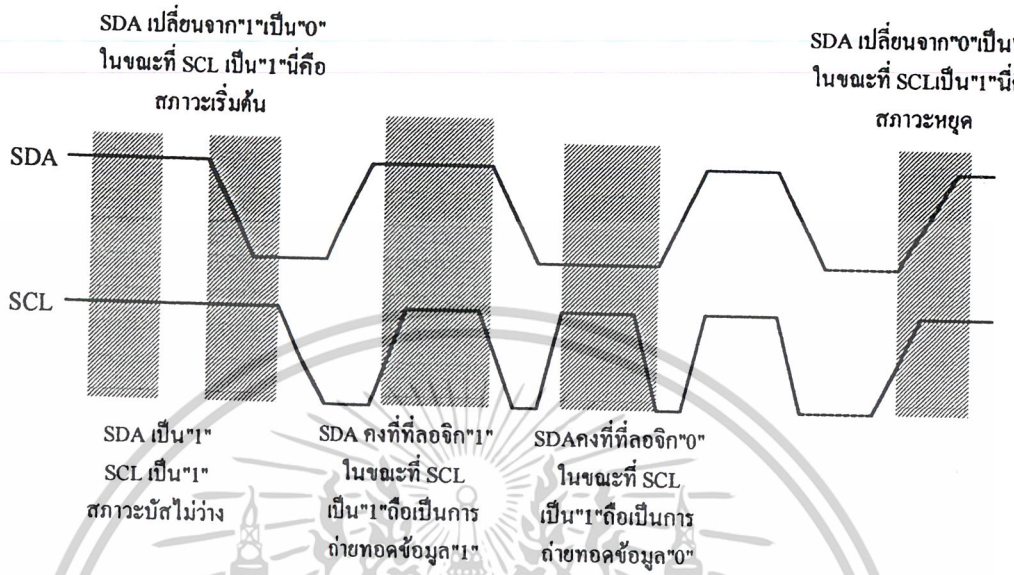
(1) บัสว่าง (bus not busy) สภาวะนี้เกิดขึ้นเมื่อสถานะลอจิกบนสาย SDA และ SCL เป็นลอจิกสูงทั้งคู่ นั่นหมายความว่า การถ่ายทอดข้อมูลสามารถเริ่มต้นขึ้นได้

(2) เริ่มต้นการถ่ายทอดข้อมูล (start data transfer) เกิดขึ้นเมื่อสาย SDA มีการเปลี่ยนแปลงระดับลอจิกจากสูงไปต่ำ ในขณะที่สาย SCL มีสถานะลอจิกสูง เรียกสภาวะที่เกิดขึ้นนี้ว่า สภาวะเริ่มต้น (START)

(3) หยุดการถ่ายทอดข้อมูล (stop data transfer) เกิดขึ้นเมื่อสาย SDA มีการแปลงเปลี่ยนระดับลอจิกจากต่ำไปสูง ในขณะที่สาย SCL มีสถานะลอจิกสูง เรียกสภาวะที่เกิดขึ้นว่า สภาวะหยุด (STOP)

(4) ข้อมูลคำรงอยู่บนบัส (data valid) สภาวะนี้เกิดขึ้นถัดจากสภาพวะเริ่มต้น โดยสถานะลอจิกที่เกิดขึ้นบนสาย SDA ก็คือข้อมูลที่ทำการถ่ายทอด เมื่อสาย SCL เป็นลอจิกสูงสถานะที่สาย SDA ต้องคงที่ เพื่อให้อุปกรณ์รับรู้ข้อมูลในจังหวะนั้นว่า เป็น “0”หรือ “1” ข้อมูลอาจเกิดการเปลี่ยนแปลงได้ในขณะที่สาย SCL เป็นลอจิกต่ำ แต่เมื่อใดก็ตามที่ต้องการให้เกิดการถ่ายทอดข้อมูลอย่างสมบูรณ์ สถานะลอจิกที่ขา SDA ต้องคงที่ตลอดช่วงเวลาที่สาย SCL มีสถานะลอจิกสูง หากเกิดการเปลี่ยนแปลงสถานะลอจิกในขณะที่สาย SCL มีลอจิกสูงอยู่นั้น อุปกรณ์มาสเตอร์ที่ทำการควบคุมการถ่ายทอดข้อมูลจะแปลความหมายเป็นสภาวะหยุดหรือสภาวะเริ่มต้นก็ได้ทำให้ข้อมูลที่ทำการถ่ายทอดอาจผิดพลาดไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 ไดอะแกรมเวลาแสดงสถานะต่างๆในบัส I^2C

(5) รับรู้ข้อมูล (acknowledge) เกิดขึ้นหลังจากที่การถ่ายทอดข้อมูลจากตัวส่งมายังตัวรับเกิดขึ้นอย่างสมบูรณ์ โดยตัวส่งจะทำการส่งข้อมูลมา 1 บิตเรียกว่าบิตรับรู้ (acknowledge bit) มีสถานะเป็นลอจิกสูง หลังจากส่งข้อมูลมาครบถ้วน ส่วนอุปกรณ์มาสเตอร์จะทำการส่งสัญญาณรับรู้พิเศษซึ่งสัมพันธ์กับสัญญาณนาฬิกาเพื่อตอบสนองบิตรับรู้ที่ส่งมาจากตัวส่ง ทางด้านตัวรับจะส่งบิตรับรู้ที่มีสถานะลอจิกต่ำลงบนบัส อุปกรณ์สเลฟที่ถูกอ้างถึงในการติดต่อหรือกำลังติดต่ออยู่ในขณะนั้นก็จะกำเนิดบิตรับรู้เพื่อตอบสนองให้ทราบว่าได้รับข้อมูลในแต่ละบิตเรียบร้อยแล้วในรูปที่ 2.3 เป็นไดอะแกรมเวลาที่แสดงถึงการเกิดสภาวะต่างๆบนบัส I^2C ไม่ว่าจะเป็นสภาวะบัสว่าง เริ่มต้น ถ่ายทอดข้อมูล รับรู้ และหยุดการถ่ายทอดข้อมูล

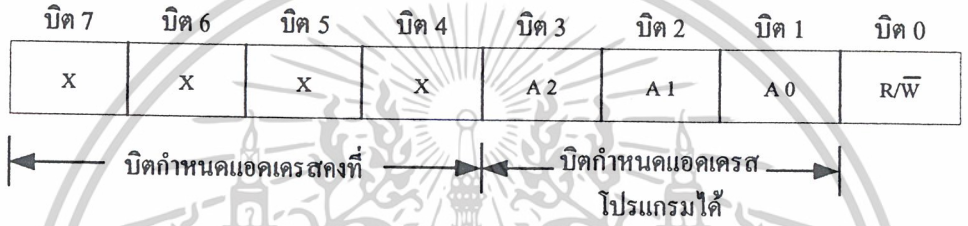
2.4 การทำงานบนบัส I^2C

ก่อนที่จะเริ่มต้นการถ่ายทอดข้อมูลระหว่างต่างๆที่ต่ออยู่บนบัส ต้องมีการอ้างถึงเสียก่อนโดยการอ้างถึงอุปกรณ์บนบัส I^2C นั้นจะใช้การอ้างถึงแบบ 7 บิตหรือ 10 บิต ในกรณีที่มีอุปกรณ์ต่ออยู่บน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บัสไม่มาก ใช้การอ้างถึงแบบ 7 บิตก็เพียงพอ แต่ถ้ามีอุปกรณ์ต่ออยู่บนบัสมากกว่า 127 - แอดเดรส จำเป็นต้องใช้การอ้างอิงถึงแบบ 10 บิต หลังจากที่ติดต่ออุปกรณ์แต่ละตัวได้เรียบร้อยแล้วก็จะเริ่มต้นการถ่ายทอดข้อมูลกันต่อไป

ดังนั้นหัวใจสำคัญอันดับแรกของการทำงานบนบัส I^2C คือการอ้างถึงอุปกรณ์แต่ละตัว ซึ่งในที่นี้จะอธิบายรายละเอียดของการอ้างถึงทั้ง 2 รูปแบบ



รูปที่ 2.4 รูปแบบของข้อมูลกำหนดแอดเอรสที่ใช้ในการอ้างถึงแบบ 7 บิต

2.4.1 การอ้างถึงแบบ 7 บิต (7-bit addressing)

ข้อมูลไบต์แรกที่เกิดขึ้นหลังจากสภาวะเริ่มต้นคือ ข้อมูลที่ใช้ในการอ้างถึงอุปกรณ์ที่ต้างการติดต่อหรือข้อมูลกำหนดแอดเดรส โดยมีรูปแบบแสดงในรูปที่ 2.5 ใน 7 บิตบนรวมทั้งบิต MSB ด้วยจะเป็นข้อมูลแอดเดรสของอุปกรณ์สเลฟที่ต้องการติดต่อ โดยแบ่งเป็น บิตกำหนดแอดเดรสคงที่ (fixed address bit) จำนวน 4 บิต ซึ่งข้อมูลนี้อุปกรณ์แต่ละตัวจะถูกกำหนดมาจากผู้ผลิต ไม่สามารถเปลี่ยนแปลงแก้ไขได้ ถัดมาอีก 3 บิตเป็นบิตกำหนดแอดเดรสที่สามารถโปรแกรมได้ (programmable address bit) โดยผู้ใช้งานต้องกำหนดสถานะลอจิกให้กับขา A0-A2 ของอุปกรณ์ที่มีการเชื่อมต่อแบบบัส I^2C ส่วนในแบบ LSB เป็นที่ใช้กำหนดการอ่านหรือเขียนข้อมูลกับอปกรณ์สเลฟตัวนั้นๆ หากบิต LSB เป็น "0" หมายถึงต้องการเขียนข้อมูลไปยังอุปกรณ์นั้น ถ้าเป็น "1" จะเป็นการอ่านข้อมูลจากอุปกรณ์สเลฟ ข้อมูลในไบต์ต่อมาคือ ข้อมูลควบคุม (control byte) ในอุปกรณ์แต่ละตัวมีการกำหนดข้อมูลควบคุมแตกต่างกันไป ยกตัวอย่าง ไอซีขยายพอร์ตมีข้อมูลควบคุมที่ใช้กำหนดว่า บิตใดเป็นอินพุต บิตใดเป็นเอาท์พุต ในขณะที่ไอซี ADC/DAC ต้องการข้อมูลควบคุมเพื่อกำหนดให้ทำงานเป็นวงจร ADC หรือ DAC เป็นต้น ข้อมูลในไบต์ต่อมาคือ ข้อมูลที่ทำการถ่ายทอดจริง (data) หลังจากที่มีการถ่ยทอดข้อมูลในแต่ละไบต์ อุปกรณ์สเลฟที่ได้รับการติดต่อต้องส่งสัญญาณรับรู้ตอบกลับมาทุกครั้ง เพื่อให้กระบวนการถ่ายทอดข้อมูลสามารถดำเนินต่อไปได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2 การอ้างถึงแบบ 10 บิต

ในการอ้างถึงแบบนี้ยังคงใช้รูปแบบข้อมูลอนุกรมที่เหมือนกันแบบ 7 บิต หากแต่จะมีข้อมูลอนุกรมที่เหมือนกันแบบ 7 บิตหากแต่ยังมีข้อมูลเพิ่มเติมขึ้นมาเล็กน้อย โดยในข้อมูลไบต์แรกหลังจากเกิดสภาวะเริ่มต้น ต้องกำหนดให้ 5 บิตบนมีข้อมูลเป็น 11110 ส่วนอีก 2 บิตถัดมาเป็นบิตแอดเดรสของอุปกรณ์ที่ต้องการติดต่อ ในบิต LSB ของข้อมูลไบต์แรกยังคงเป็นการกำหนดว่า ต้องการอ่านหรือเขียนข้อมูลกับอุปกรณ์สเลฟตัวที่ต้องการติดต่อด้วย ข้อมูลไบต์ต่อมาเป็นข้อมูลแอดเดรสในไบต์ที่ 2 ของอุปกรณ์ที่ต้องการติดต่อด้วย ข้อมูลไบต์ถัดไปจึงเป็นข้อมูลควบคุม ข้อมูลหลังจากนั้นก้จะเป็นข้อมูลจริงที่ใช้ในการติดต่อ เช่นเดียวกับการอ้างถึงแบบ 7 บิต หลังจากการถ่ายทอดข้อมูลครบทุกไบต์ ต้องมีสภาวะรับรู้เกิดขึ้น เพื่อให้กระบวนการถ่ายทอดข้อมูลสามารถดำเนินต่อไปได้

2.5 อุปกรณ์ที่ใช้การเชื่อมต่อแบบ บัส I^2C

ในปัจจุบันบัสI^2Cได้รับความนิยมเพิ่มมากขึ้นเรื่อยๆด้วยข้อดีที่ชัดเจนคือใช้สายสัญญาณเพียง 2 เส้นเท่านั้น และการขยายระบบไมโครคอลโทรลเลอร์ ที่มีจำนวนอินพุตเอาท์พุตและหน่วยความจำจำกัดสามารถทำได้ง่ายขึ้นด้วยระบบบัส I^2C เมื่อเป็นเช่นนี้จึงมีอุปกรณ์เพอริเฟอรัลที่ใช้ในการเชื่อมต่อแบบบัส I^2C มากมายจากหลายผู้ผลิตออกมาให้ได้ใช้งาน ดังมีตัวอย่างต่อไปนี้

ไอซีขยายพอร์ตอินพุตเอาท์พุต (I/O expander) : PCF8574, PCF8582, PCF8584

ไอซีหน่วยความจำอีพรอมอนุกรม (serial EEPROM) : 24Cxx, PCF8570, PCF72/73, PCF8582

ไอซี ADC/DAC : PCF8591

ไอซีไทม์คล็อก (Real-time clock;RTC) : PCF8583, PCF8593, PCF8598, 41T56c

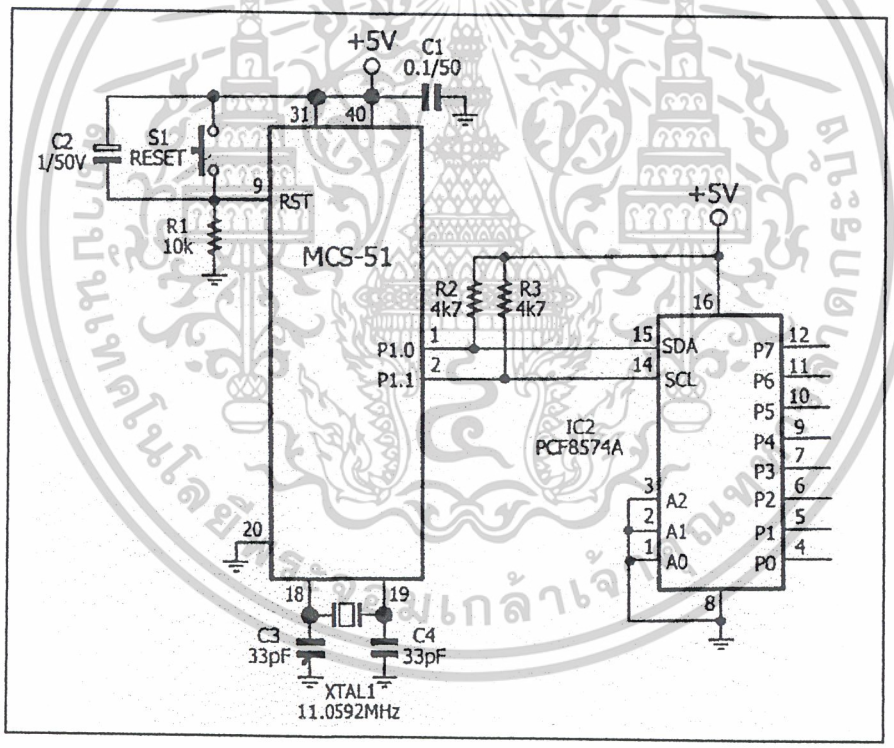
ไอซีขับ LCD โมดูล (LCD driver) : PCF8466, PCF8576, PCF8577/78, PCF8579, SAA1064

ไอซีกำเนิดสัญญาณ DTMF (DTMF generator) : PCD3311/12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.1 การต่ออุปกรณ์ระบบบัส I^2C กับไมโครคอนโทรลเลอร์ MCS-51

สามารถทำได้ เพียงใช้ขาพอร์ต 2 ขา โดยกำหนดให้ขาหนึ่งเป็น SDA อีกขาหนึ่งเป็น SCL และต่อตัวต้านทานค่าประมาณ 4.7k พูลอัพที่ขาพอร์ตทั้งสองขา เพียงเท่านี้ก็สามารถติดต่อกับอุปกรณ์รับบัส I^2C ได้แล้ว



รูปที่ 2.5 วงจรตัวอย่างการต่อไมโครคอนโทรลเลอร์ MCS-51 กับอุปกรณ์ระบบบัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในรูป 2.5 เป็นวงจรตัวอย่างการต่อไมโครคอนโทรลเลอร์ MCS-51 เข้ากับระบบบัส I^2C จากวงจรจะใช้ขาพอร์ต P1.0 เป็นขา SDA และP1.1 เป็นขา SCL อุปกรณ์ที่ทำการติดต่อด้วยคือไอซีขยายพอร์ตอินพุทเอาท์พุทเบอร์ PCF8574

2.6 การเขียนโปรแกรมติดต่อบัส I^2C

เริ่มต้นด้วยการสร้างสภาวะมาตรฐานของบัส I^2C อันประกอบด้วย สภาวะเริ่มต้น สภาวะสิ้นสุดการส่งข้อมูล สภาวะหยุด และสัญญาณนาฬิกาบนขา SCL การสร้างสภาวะหยุด

1. เมื่อต้องการหยุดส่งข้อมูลจะต้องส่งสภาวะหยุดออกไป โดยในตอนแรกต้องกำหนดให้ขา SCL และ SDA เป็นลอจิก "0" ทั้งคู่ก่อน
2. กำหนดให้ขา SCL มีลอจิกเป็น "1" โดย SDA ยังคงมีลอจิกเป็น "0"
3. จากนั้นทำให้ขา SDA มีลอจิกเป็น "1" ซึ่งจะทำให้ระบบบัสกลับเข้าสู่บัสว่างอีกครั้ง พร้อมที่จะรับหรือส่งข้อมูลต่อไป

<u>การส่งข้อมูลลอจิก "0" และลอจิก "1"</u>

หลังจากที่ทำการส่งบิตเริ่มต้นแล้ว ลำดับต่อไปคือ จะต้องส่งข้อมูลควบคุมซึ่งจะเป็นขบวนของลอจิก "0" และลอจิก "1" สำหรับการส่งข้อมูลลอจิก "0" ต้องดำเนินการตามขั้นตอนดังนี้

1. ทำให้ขา SDA เป็น "0" สำหรับการส่งข้อมูลลอจิก "0"
2. ทำให้ขา SCL เป็น "1" สำหรับการป้อนสัญญาณนาฬิกา ในขณะที่ขา SDA ยังคงเป็น "0" อยู่
3. จากนั้นทำให้ขา SCL กลับมามีสถานะเป็นลอจิก "0"เหมือนเดิม

<u>ในขณะที่การส่งข้อมูลลอจิก "1" มีขั้นตอนดังนี้</u>

1. ทำให้ขา SDA มีลอจิกเป็น "1" สำหรับการส่งข้อมูลลอจิก "1"
2. ทำให้ขา SCL เป็น "1" สำหรับการส่งสัญญาณนาฬิกา ในขณะที่ขา SDA ยังคงเป็น "1"อยู่
3. จากนั้นทำให้ขา SCL เป็น "1" กลับมามีสถานะเป็นลอจิก "0" เหมือนเดิม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 รีจิสเตอร์ควบคุมการทำงานของพอร์ตอนุกรม SCON (Serial Port Control Register)

รีจิสเตอร์ควบคุมการทำงานของพอร์ตอนุกรม SCON เป็นรีจิสเตอร์เฉพาะที่ทำหน้าที่ควบคุมโหมดการทำงานของพอร์ตอนุกรม และเป็นที่เก็บข้อมูลบิตที่ 9 ของการรับส่งข้อมูล (บิต TB8 และ RB8) และมีแฟลกของการร้องขออินเตอร์รัพของพอร์ตอนุกรมรวมอยู่ด้วย การควบคุมการทำงานเราจะกำหนดบิตต่างๆในรีจิสเตอร์ตัวนี้ด้วยคำสั่งการโอนย้ายข้อมูลหรือใช้คำสั่งการเซตหรือเคลียร์บิตก็ได้ เนื่องจากรีจิสเตอร์ SCON อ้างตำแหน่งแบบบิตได้

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
SM0	SM1	SM2	REN	TB8	RB8	T1	R1

SCON : Serial Port Control Register

ความหมายของบิตต่างๆในรีจิสเตอร์ SCON เป็นดังนี้

SM0 และ SM1 : เป็นบิตกำหนดโหมดการทำงานของพอร์ตอนุกรมซึ่งมี 4 โหมดดังนี้

SM0	SM1	โหมด	การทำงาน	อัตรารับส่ง
0	0	0	shift register	$f_{osc}/12$
0	1	1	8 bit UART	Variable
1	0	2	9 bit UART	$f_{osc}/32$ หรือ $f_{osc}/64$
1	1	3	9 bit UART	Variable

SM2 : เป็นบิตควบคุมให้ทำงานในลักษณะการเชื่อมต่อไมโครคอนโทรลเลอร์หลายตัวเข้าด้วยกัน สำหรับการใช้งานในโหมด 2 หรือโหมด 3 เป็นดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SM2 = 1 จะทำให้มีแฟลกอินเตอร์รัพต์ทางด้านรับ (RI) ไม่ถูกเซตเมื่อรับข้อมูลเข้ามาแล้วมีค่าบิตที่ 9 เป็น 0 (อยู่ในบิต RB8) สำหรับการทำงานในโหมด 1 ถ้าเซต SM2 = 1 แฟลกอินเตอร์รัพต์ทางด้านรับ (แฟลกRI) จะไม่ถูกเซตหากข้อมูลที่รับเข้ามาไม่มี STOP BIT การใช้งานในโหมด 0 ต้องกำหนดให้ SM2 = 0

REN : เซตหรือรีเซตด้วยซอฟต์แวร์เป็นตัวควบคุมการรับข้อมูลของพอร์ตอนุกรมดังนี้

1 = ให้มีการรับข้อมูล

0 = ไม่ให้มีการรับข้อมูล

TB8 : เป็นบิตข้อมูลบิตที่ 9 ที่ต้องการส่งในโหมด 2 และ 3 สามารถเซตหรือเคลียร์ด้วยซอฟต์แวร์

RB8 : เป็นบิตเก็บข้อมูลที่รับเข้ามาบิตที่ 9 ในโหมด 2 และ 3 สำหรับการทำงานในโหมด 1 หากกำหนดให้บิต SM2 = 0 บิต RB8 จะเป็นค่าของ STOP BITที่รับเข้ามา สำหรับโหมด 0 ไม่มีการใช้ RB8

T1 : แฟลกของการอินเตอร์รัพต์ด้านส่งข้อมูล แฟลกนี้จะถูกเซตด้วยฮาร์ดแวร์เมื่อจบการส่งข้อมูลบิตที่ 8 ในโหมด 0 หรือเมื่อเริ่มต้นส่ง STOP BIT ในโหมด 1,2 หรือ 3 เราต้องเคลียร์แฟลกนี้ด้วยซอฟต์แวร์เมื่อจบโปรแกรมตอบสนองการอินเตอร์รัพต์ของการส่งข้อมูลแล้ว

R1 : แฟลกอินเตอร์รัพต์ด้านรับข้อมูล ถูกเซตด้วยฮาร์ดแวร์ เมื่อข้อมูลบิตที่ 8 ในโหมด 0 ถูกรับเข้ามาหรือเมื่อ STOP BIT ถูกรับเข้ามาในครึ่งเวลาแรกในโหมด 1,2 หรือ 3 เราต้องเคลียร์แฟลกนี้ด้วยซอฟต์แวร์ เมื่อจบโปรแกรมตอบสนองการอินเตอร์รัพต์ของการส่งข้อมูลแล้ว การเลือกโหมดการทำงานของพอร์ตอนุกรม สามารถกำหนดโหมดต่างๆได้ดังรูปต่อไปนี้

MODE	รีจิสเตอร์ SCON	บิต SM2 Variable
0	10 H	Single Mode Environment (SM2 = 0)
1	50 H	
2	90 H	
3	D0 H	
0	NA	Multiprocessor Environment (SM2 = 1)
1	70 H	
2	B0 H	
3	F0 H	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การส่งและรับข้อมูลของพอร์ตอนุกรมจะมีรีจิสเตอร์ที่ทำหน้าที่รับและส่งข้อมูลอยู่ 1 ตัวคือ รีจิสเตอร์ SBUF การส่งข้อมูลออกไปยังพอร์ตอนุกรมของไมโครคอนโทรลเลอร์ทำโดยการใส่ข้อมูลลงในรีจิสเตอร์ SBUF การอ่านข้อมูลจากภายนอกที่รับเข้ามาทางพอร์ตอนุกรมจะอ่านจากรีจิสเตอร์ SBUF เช่นกัน วงจรด้านรับจะมีบัฟเฟอร์ขนาด 1 ไบต์ที่ทำหน้าที่เก็บข้อมูลที่รับเข้ามาประกอบอยู่ภายใน (การมีบัฟเฟอร์รับข้อมูลทำให้ด้านรับสามารถรับข้อมูลไบต์ที่ 2 เข้ามาได้ทันทีหลังจากข้อมูลไบต์แรกเข้ามาแล้วแม้ยังไม่ถูกอ่านออกไปแต่ถ้าข้อมูลไบต์แรกยังไม่ถูกอ่านก่อนที่ข้อมูลไบต์ที่ 2 จะเข้ามาครบข้อมูลไบต์ที่ 2 จะถูกยกเลิก)

<u>โหมด 1</u>

ใช้การรับส่งข้อมูลแบบ 10 บิต ซึ่งข้อมูลอนุกรม 10 บิต เข้ามาทางขา RXD และข้อมูลขนาด 10 บิต ส่งออกแบบอนุกรมทางขา TXD โดยข้อมูล 10 บิตประกอบด้วย 1 Strat bit (ค่า 0), 8 บิตข้อมูล (การรับ/ส่งจะเริ่มจากบิตต่ำก่อน) และ 1 Stop bit (ค่า 1) ด้านรับข้อมูลจะนำค่า Stop bit ที่รับเข้ามาไปในบิต RB8 ที่อยู่ในรีจิสเตอร์ SCON อัตราการรับส่งข้อมูลในโหมดนี้สามารถกำหนดได้ตามต้องการ

<u>การรับส่งข้อมูลในโหมด 1</u>

การส่งข้อมูลจำนวน 10 บิต จะส่งออกไปที่ขา TXD และรับข้อมูลเข้ามาทางขา RXD ข้อมูล 10 บิต ประกอบด้วย 1 Start bit (ค่าลอจิก 0) ข้อมูลจำนวน 8 บิต (LSB ส่งหรือรับก่อน) และ 1 Stop bit (ค่าลอจิก 1)

ในกรณีของการรับข้อมูล Stop bit จะถูกนำเข้าไปเก็บในบิต RB8 ที่อยู่ในรีจิสเตอร์ SCON ความเร็วในการรับส่งข้อมูลจะกำหนดโดยโอเวอร์โฟลวของ Timer1 หรือ Timer2 หรือกำหนดจากทั้ง 2 ตัว (ตัวหนึ่งสำหรับด้านส่ง ตัวหนึ่งสำหรับด้านรับ)

การส่งข้อมูลจะเริ่มต้นหลังจากกำหนดค่าลงในรีจิสเตอร์ข้อมูล SBUF ด้านส่งนะมีสัญญาณ Write to SBUF ควบคุมให้มีการใส่ค่า 1 ลงในตำแหน่งของบิตที่ 9 ของ SHIFT REGISTER ด้านส่งและเริ่มทำการส่งข้อมูลของ TX CONTROL UNIT การส่งข้อมูลออกจะเริ่มต้นในช่วงเวลา S1P1 ของแมชชีนไซเกิลและบิตต่อไปตามจังหวะของตัวกำหนดอัตราส่งที่มาจากโอเวอร์โฟลวของ Timer1 หรือ Timer2 ที่หารด้วย 16

การส่งข้อมูลจะเริ่มด้วยการส่งสัญญาณ SEND ให้แอคทีฟ เมื่อเริ่มต้นการส่งข้อมูลบิตแรกออกไปที่ TXD และบิตของข้อมูลจะถูกส่งตามออกไปโดยการทำงานของ SHIFT REGISTER ซึ่งจะทำการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เลื่อนข้อมูลไป 1 ตำแหน่งตามจังหวะของอัตราส่งที่เข้ามายัง TX CLOCK เมื่อบิตข้อมูลทั้งหมดถูกเลื่อนออกไปทางขวา จะมี 01 เข้ามาทางด้านซ้าย จนกระทั่งบิตข้อมูลสูงสุด (MSB) ออกไปอยู่ที่เอาต์พุตของ SHIFT REGISTER จะทำให้บิตที่ 1 ที่ได้โหลดเข้าไปตอนเริ่มต้นในตำแหน่งที่ 9 ซึ่งอยู่ทางซ้ายมือของ MSB และมีบิตต่างๆที่อยู่ทางซ้ายของบิตที่ 9 เป็น 0 ทั้งหมด เมื่อเกิดเงื่อนไขดังกล่าวจะทำให้ TX CONTROL UNIT ทำการเลื่อนข้อมูลออกไปทางขวาอีก 1 ครั้ง เป็นครั้งสุดท้าย แล้วเลิกส่งสัญญาณ SEND และเซตแฟลก TI = 1 เวลาการส่งข้อมูลจะจบในช่วง CLOCK ลูกที่ 10 ของสัญญาณการส่งข้อมูล หลังจากมีสัญญาณ Write to SBUF

การรับรับข้อมูลจะเริ่มต้นเมื่อสัญญาณที่ขา RXD เปลี่ยนจาก 1 ไป 0 โดยซีพียูจะมีการตรวจสอบด้วยความเร็ว 16 เท่า ของอัตราความเร็วในการรับข้อมูล เมื่อซีพียูตรวจสอบได้ว่ามีการส่งข้อมูลเข้ามา จะทำการรีเซตวงจรหาร 16 ทันทีเพื่อเริ่มต้นใหม่ให้สัมพันธ์กับข้อมูลที่เข้ามาและค่า 1FFH จะถูกกำหนดให้กับ SHIFT REGISTER ด้านรับ

การรับรับข้อมูลจะทำการตรวจสอบบิตข้อมูลที่ขา RXD 3 ครั้งในช่วงบิต RCLK ที่ 7, 8 และ 9 เข้ามาในวงจรหาร 16 โดยจะใช้ค่าที่อ่านได้จาก 2 มน 3 ครั้งเป็นข้อมูล การทำเช่นนี้ก็เพื่อกำจัดสัญญาณรบกวนออก หาก ตรวจสอบว่าในช่วงบิตที่ 0 ของ RCLK ข้อมูลที่ RXD ไม่เป็น 0 วงจรด้านรับจะรีเซตกลับไปรอการเปลี่ยนค่าที่ขา RXD จาก 1 ไป 0 ใหม่ การทำเช่นนี้เพื่อป้องกันการผิดพลาดของ Start bit หากค่า Start bit ถูกต้อง จะทำการเลื่อนข้อมูลเข้าไปใน SHIFT REGISTER และรอรับบิตต่อๆไป โดยบิตข้อมูลจะเข้ามาทางขวาของ SHIFT REGISTER และค่า 1 ที่กำหนดไปตอนเริ่มจะถูกเลื่อนไปทางซ้ายตามลำดับ เมื่อ Start bit ที่เข้ามาถูกเลื่อนไปทางซ้ายมือสุดของ SHIFT REGISTER จะทำให้ RX CONTROL BOX ทำการเลื่อนอีก 1 ครั้ง เป็นครั้งสุดท้าย และนำค่าใน SHIFT REGISTER ไปเก็บในรี-จิสเตอร์ SBUF และ RB8 และการเซตแฟลก RI จะเกิดขึ้นหลังจากเลื่อนข้อมูลครั้งสุดท้ายแล้วก็ต่อเมื่อเราได้กำหนด

1) RI = 0 (จะต้องเคลียร์ RI หลังจากทำโปรแกรมอ่านข้อมูลไปแล้ว) และ

2) SM2 = 0 หรือได้รับ STOP BIT = 1

หากเงื่อนไขทั้ง 2 ไม่จริง ข้อมูลที่รับเข้ามาจะถูกยกเลิก ถ้าเงื่อนไขทั้ง 2 เป็นจริง STOP BITจะถูกนำไปเก็บใน SBUF และ RB8 ถูกเซต วงจรรับข้อมูลจะกลับไปตรวจสอบค่าที่ขา RXD ต่อ เพื่อรับข้อมูลไบต์ใหม่ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8 การกำหนดอัตรารับส่ง (Baud Rate Generator)

อัตรารับส่งข้อมูลเป็นความเร็วในการส่งข้อมูลในหน่วย บิต/วินาที ซึ่งเราสามารถกำหนดอัตรารับส่งที่แตกต่างกันได้ในโหมด 1 และโหมด 3 สำหรับอัตรารับส่งในโหมด 0 จะคงที่ส่วนในโหมด 2 เลือกได้ 2 อัตราความเร็ว

อัตรารับส่งข้อมูลในโหมด 0 จะใช้สัญญาณนาฬิกาของระบบไมโครคอนโทรลเลอร์เป็นฐาน คือ

อัตรารับส่งโหมด 0 = ความถี่ใช้สัญญาณนาฬิกาของไมโครคอนโทรลเลอร์/12

อัตรารับส่งโหมด 2 เลือกได้ 2 อัตราความเร็ว ขึ้นอยู่กับบิต SMOD ที่อยู่ในรีจิสเตอร์ PCON เป็นตัวเลือก

ถ้าบิต SMOD = 0 (เป็นค่าหลังจากรีเซต)

อัตรารับส่งโหมด 2 = 1/64 เท่าของสัญญาณนาฬิกา

ถ้าบิต SMOD = 1

อัตรารับส่งโหมด 2 = 1/32 เท่าของสัญญาณนาฬิกา

สำหรับอัตราการรับและส่งข้อมูลในโหมด 1 และโหมด 3 จะกำหนดได้ตามต้องการ โดยใช้อัตราการเกิดโอเวอร์โฟลวของ Timer1 หรือ Timer2 เป็นตัวกำหนด (8052 มี Timer2 เพิ่มขึ้นมา) อัตราการรับและส่งข้อมูลสามารถกำหนดได้จาก Timer2 หรือจากทั้ง 2 ตัว (ตัวหนึ่งสำหรับส่งอีกตัวสำหรับรับข้อมูล) ความเร็วในการรับและส่งข้อมูลอาจกำหนดไม่เท่ากันได้

<u>การใช้ Timer1 สร้างอัตรารับส่ง</u>

เมื่อใช้ Timer1 เป็นตัวสร้างอัตราการรับส่งข้อมูลของพอร์ตอนุกรมในโหมด 1 หรือโหมด 3 ต้องควบคุมไม่ให้ Timer1 ส่งสัญญาณอินเตอร์รัพต์อัตราการรับและส่งข้อมูลจะมาจากอัตราการเกิดโอเวอร์โฟลวของ Timer1 และค่าของบิต SMOD ที่อยู่ในรีจิสเตอร์ PCON (รีจิสเตอร์ PCON ไม่สามารถอ้างตำแหน่งแบบบิต การเซตค่าให้ใช้คำสั่งทางลอจิกของการ OR เช่น ORL PCON,#01111111B เท่ากับการเคลียร์บิต 7 ของ PCON) เราคำนวณอัตรารับส่งข้อมูลได้ดังนี้

$$\text{อัตรารับส่งในโหมด 1 และ 3} = \frac{2^{s\,\text{mod}}}{32} * (\text{Timer1 Overflow rate})$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เมื่อใช้ Timer1 สร้างอัตราการรับและส่งข้อมูลจะต้องทำให้ไม่มีการร้องขออินเตอร์รัพต์ที่เกิดจาก Timer1 (โดยการควบคุมที่รีจิสเตอร์ IE) เราจะใช้ Timer1 ทำงานในลักษณะของตัวจับเวลาหรือตัวนับอย่างใดอย่างหนึ่งในโหมดต่างๆทั้ง 3 โหมด สำหรับการใช้งานทั่วไปเราจะกำหนดให้ Timer1 ทำงานในลักษณะของตัวจับเวลาในโหมด Auto-relode (มีค่าการควบคุมใน TMOD = 0010XXXX) ซึ่งจะได้อัตราการรับส่งข้อมูลดังนี้

$$\text{อัตราการรับส่งของโหมด 1 และโหมด 3} = \frac{2^{s\,\mathrm{mod}}}{32} * \frac{Oscillator}{12 * [256 - (TH)]}$$

หากเราต้องการให้ Timer1 มีอัตรารับส่งต่ำมากๆเราทำได้โดยใช้ให้ Timer1 ทำงานในลักษณะของตัวจับเวลาแบบ 16 บิต (มีค่าของการควบคุมใน TMOD = 0001XXXX) และให้มีการอินเตอร์รัพต์จาก Timer1 ทำการกำหนดค่าเริ่มต้นใหม่ (Relode) ให้กับตัวจับเวลา ซึ่งเป็นการทำงานที่เป็นแบบ 16 บิต Software Relode

2.9 คุณสมบัติของไมโครคอนโทรลเลอร์

คุณสมบัติทั่วไปของไมโครคอนโทรลเลอร์ มีรายละเอียดสำคัญดังต่อไปนี้

- แหล่งจ่ายไฟ + 5 V
- มีหน่วยความจำโปรแกรม (Program Memory) ขนาด 4 กิโลไบต์สำหรับเบอร์ 8051 8031 และ 8032 ไม่มีหน่วยความจำชุดนี้ ส่วน 8032 มีหน่วยความจำถึง 8 กิโลไบต์
- มีหน่วยความจำสำหรับเก็บข้อมูล (Data Memory) ขนาด 128 ไบต์ สำหรับ 8052 มีถึง 256 ไบต์
- หน่วยความจำสำหรับโปรแกรมและดาต้า (Program Memory และ Data Memory แยกจากกันอย่างละ 64 กิโลไบต์)
- คำสั่งที่ใช้เวลาน้อยที่สุดประมาณ 1 μs เมื่อทำงานที่ความถี่ 12 MHz
- มี Timer/Counter ขนาด 16 บิต 2 ชุด (สำหรับ 8052 มี 3 ชุด) ทำงานได้ 4 โหมด
- รับอินเตอร์รัพท์ได้ 6 แหล่ง 5 เวกเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

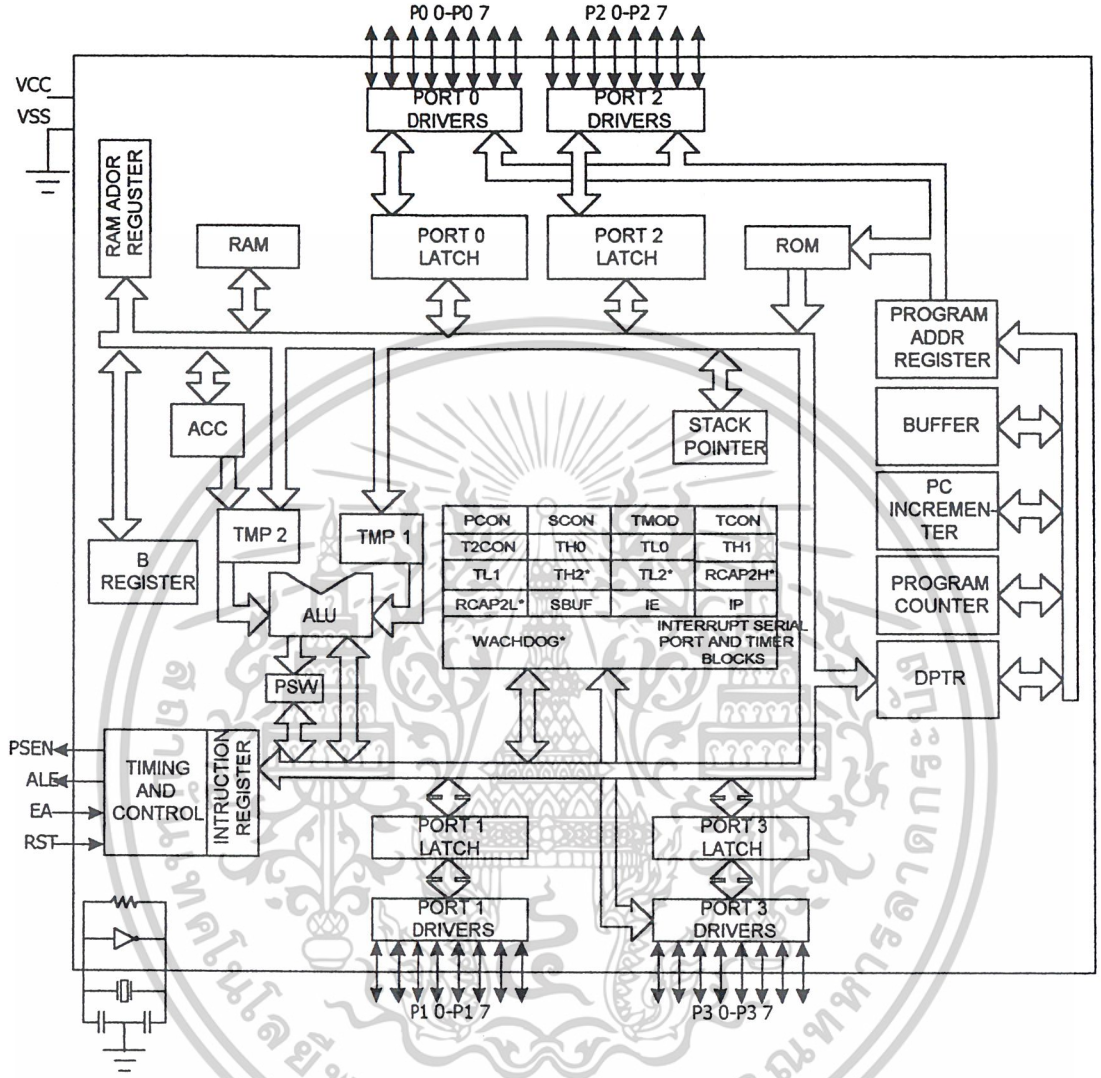
50199

- มีพอร์ทรับส่งข้อมูลอนุกรม (UART) 2 พอร์ท ทั้งรับและส่งในเวลาเดียวกันได้ (Full Duplex) เลือกรูปแบบการส่งข้อมูลได้ 4 โหมด
- มีคำสั่งในการทำ AND, OR หรือ COMPLEMENT ได้ทั้งแบบ 8 บิตและ 1 บิต

2.10 โครงสร้างภายในของ MCS-51

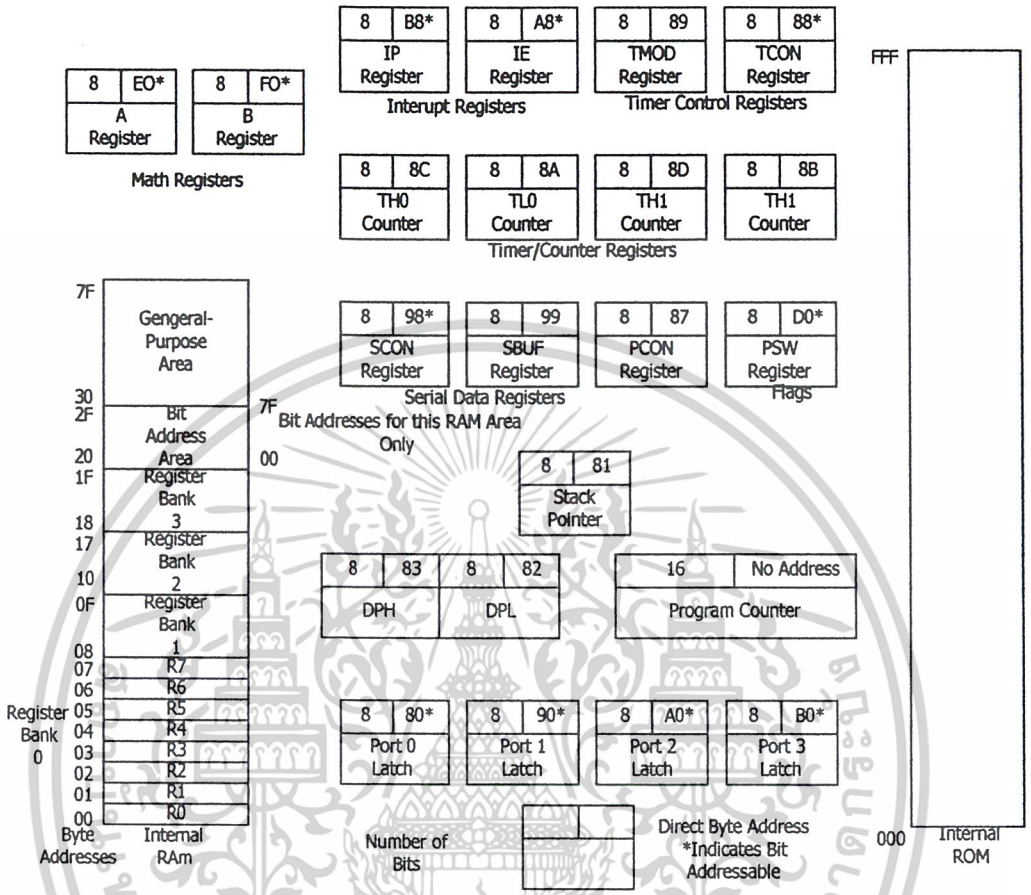
MCS-51 ใช้เทคโนโลยีในการผลิตแบบ NMOS และ CMOS เบอร์ 8032 และ 8052 จะมี ROM BASIC อยู่ภายในจึงสะดวก สำหรับโปรแกรมเมอร์ที่จะเขียนโปรแกรมด้วยภาษาเบสิก โครงสร้างสำหรับ เบอร์ 8051 ดังแสดงในรูปที่ 2.10.1 และ 2.10.2 เขียนโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

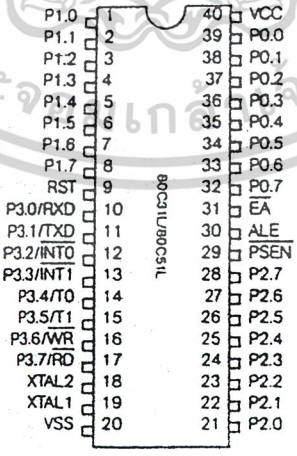


รูปที่ 2.10.1 8051 บล็อกไดอะแกรมของ MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่2.10.2 ตำแหน่งของรีจิสเตอร์ต่าง ๆ



รูปที่ 2.10.3 การจัดวางขาของ MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

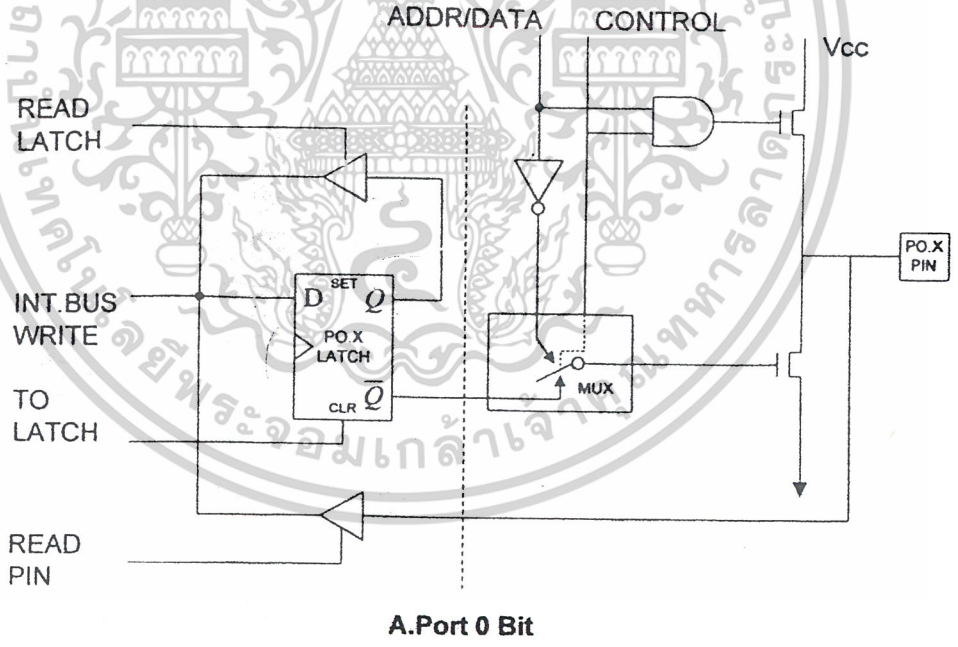
2.11 พอร์ทของ 8051

8051 เป็นไมโครคอนโทรลเลอร์ขนาด 40 ขา ซึ่งมีขาต่าง ๆ ดังนี้

- Vcc (ขา 40) ต่อกับ + 5 V

- Vss (ขา 20) เป็นขา GND

- PORT 0 (ขา 32-39) มีทั้งหมด 8 บิตคือ (P0.0-P0.7) มีโครงสร้างแบบ Open Drain Bi-directional ดังแสดงในรูปที่ 2.11.1

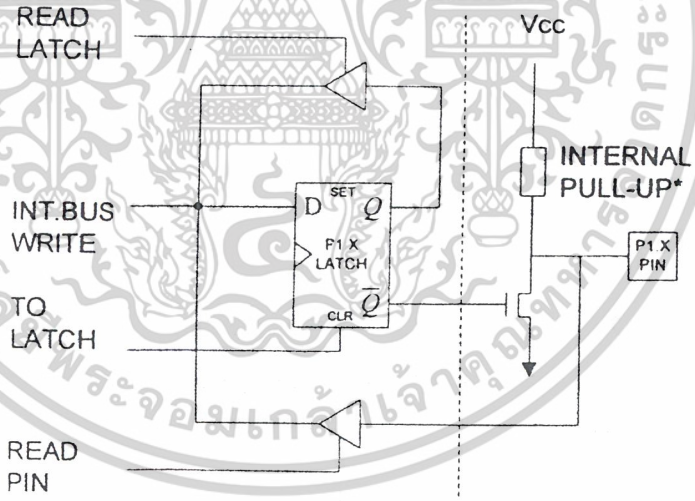


รูปที่2.11.1 โครงสร้าง พอร์ท 0 (บิต)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- พอร์ท 0 (ขา 32-39) มีทั้งหมด 8 บิต คือ (P0.0-P0.7) ใช้งานได้ 2 หน้าที่คือ ส่งแอดเดรสและดาต้าออกให้หน่วยความจำภายนอกเมื่อทำการเขียนข้อมูลลงในหน่วยความจำภายนอกควบคุมด้วยขา Control ดูรูป 2.11(ก) ประกอบและอีกหน้าที่หนึ่งหน้าที่ก็คือ เป็นพอร์ท I/O ถ้าต้องการให้ทำงานเป็นอินพุตพอร์ทต้องส่งลอจิก " 1 " ไปยังพอร์ทนี้ จะมีผลให้ $\overline{Q}$ ของ D-FF เป็น " 0 " ทำให้ FET ตัวล่างมีสถานะ OFF สัญญาณที่ใช้อ่านอินพุทพอร์ท PIN (พอร์ท P0.X PIN) จะใช้สัญญาณ READ LATCH เมื่อถูกกระตุ้นที่ Tri-State Buffer ตัวบน

- พอร์ท 1 (ขา 1-8) มีทั้งหมด 8 บิต คือ (P1.0-P1.7) มีโครงสร้างคล้าย พอร์ท 0 แต่จะใช้ความต้านทานภายในพูลอัพแทน (Internal Pull Up Reggister) มีโครงสร้างดังรูปที่ 2.11.2

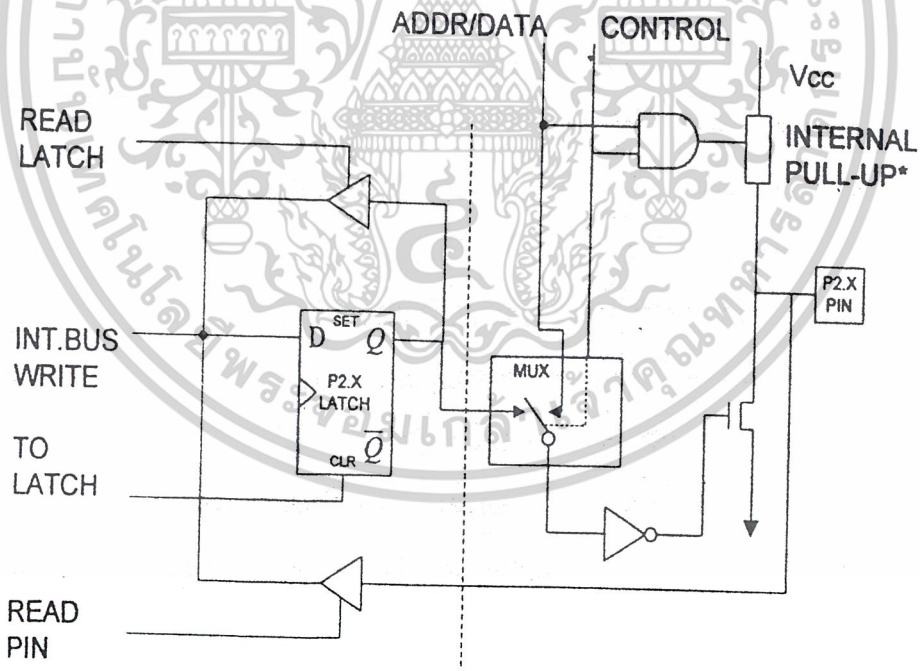


รูปที่ 2.11.2 โครงสร้างของพอร์ท 1 (บิต)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- พอร์ท 2 (ขา 21-28) มีทั้งหมด 8 บิต คือขา (P2.0-P2.7) มีโครงสร้างคล้าย PORT 0 โดยมี FET ตัวล่างตัวเดียวส่วนตัวด้านบนใช้ความต้านทานพูลอัพแทน (Internal Pull up) พอร์ทนี้ทำงาน 2 หน้าที่ คือ สามารถใช้เป็นพอร์ทสำหรับส่งแอดเดรส 8 บิตบน (A8-A15) และเป็น I/O พอร์ทใช้งานทั่วไป เมื่อจะใช้งานเป็นอินพุตพอร์ท ต้องส่งลอจิก " 1 " มาที่พอร์ทนี้ก่อนเพื่อบังคับให้ FET อยู่ในสภาวะ OFF ดังแสดงในรูปที่ 2.11.3

- พอร์ท 3 (ขา 10-17) มีทั้งหมด 8 บิต คือขา (P3.0-P3.7) มีโครงสร้างคล้าย พอร์ท 1 พอร์ทนี้ทำหน้าที่เป็น I/O พอร์ท ถ้าจะให้พอร์ทนี้เป็น I/O PORT ก็ส่งลอจิก " 1 " มาที่พอร์ทนี้และอีกหน้าที่หนึ่ง คือ ส่งสัญญาณควบคุมออกมาและรับสัญญาณเข้าไป สัญญาณต่าง ๆ มีดังนี้



C.Port 2 Bit

รูปที่ 2.11.3 โครงสร้างของพอร์ท 2 (บิต)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

P3.0/RXD (Serial Input Port) เป็นขาที่ใช้รับข้อมูลแบบอนุกรม

P3.1/RXD (Serial Output Port) เป็นขาที่ใช้ส่งข้อมูลแบบอนุกรม

P3.2/$\overline{INT0}$ (External Interrupt) ใช้รับสัญญาณขัดจังหวะจากภายนอก

P3.3/$\overline{INT1}$ (External Interrupt) ใช้รับสัญญาณขัดจังหวะจากภายนอก

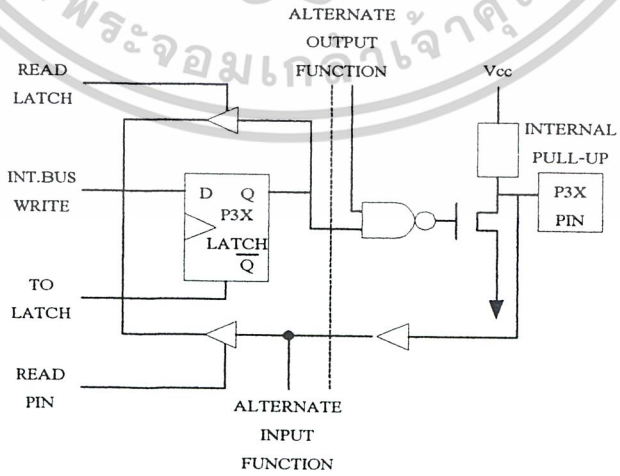
P3.4/T0 (Timer/Counter 0 External Input) ขารับสัญญาณเข้าไปยังวงจร Timer/Counter 0 ที่ทำหน้าที่นับสัญญาณไซเคิลของสัญญาณ T1 นี้หรือสัญญาณนาฬิกาก็ได้

P3.5/T1 (Timer/Counter 1 External Input) ขารับสัญญาณเข้าไปยัง Timer/Counter 1 ซึ่งมีการทำงานเหมือน T0

P3.6/$\overline{WR}$ (External Data Memory Write Strobe) ขาสัญญาณควบคุมการเขียนข้อมูลไปยังหน่วยความจำสำหรับข้อมูลภายนอก 8051

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

โครงสร้างของ พอร์ท 3 แสดงได้ดังรูปที่ 2.11.4



รูปที่2.11.4 โครงสร้างของพอร์ท 3 (บิต)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ALE (ขา 30) เป็นขาส่งสโตรบสำหรับใช้ในการแลตซ์แอดเดรสไบร์ต่ำ (A0-A7) ที่ส่งออกมาจาก (พอร์ท 0) สัญญาณนี้จะแอคทีฟทุก ๆ 2 ครั้ง ใน 1 แมชชีนไซเคิล (1/6 ของสัญญาณนาฬิกา)

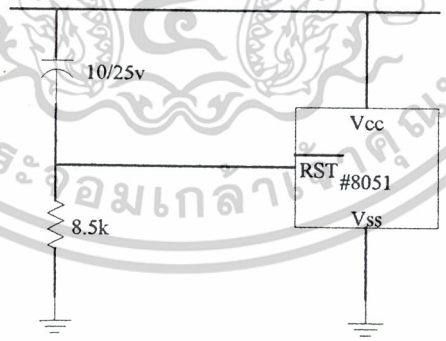
- $\overline{\text{PSEN}}$ (ขา 29) เป็นขาที่ใช้ส่งสโตรบสำหรับอ่านข้อมูลจาก Program Memory ภายนอก (หน่วยความจำประเภท ROM EPROM) สัญญาณนี้จะส่งออกมา 2 ครั้งในแต่ละแมชชีนไซเคิลแต่ถ้าเป็นการอ่าน Internal Program Memory จะไม่มีสัญญาณออกที่ขานี้

- $\overline{\text{EA}}$ (ขา 30) ถ้าป้อนลอจิก " 0 " เข้ามาที่ขานี้ซีพียูจะอ่านค่าจาก Program Memory ภายนอกชิพเท่านั้น แต่ถ้าถูกป้อนด้วยลอจิก " 1 " ก็จะอ่านโปรแกรมภายในชิพ

- RST (ขา 9) เป็นขารีเซ็ตซีพียูจะรีเซ็ตได้ก็ต่อเมื่อ ป้อนลอจิก " 1 " เข้ามาที่ขานี้อย่างน้อย 2 - แมชชีนไซเคิล เมื่อซีพียูถูกรีเซ็ตค่าต่าง ๆ ในรีจิสเตอร์ใด ๆ

- XTAL1 (ขา 19) ใช้ต่อคริสตอลภายนอกโดยเป็นอินพุตเข้าสู่วงจรออสซิเลเตอร์

- XTAL2 (ขา 18) ใช้ต่อคริสตอลภายนอกโดยเป็นอินพุตเข้าสู่วงจรออสซิเลเตอร์

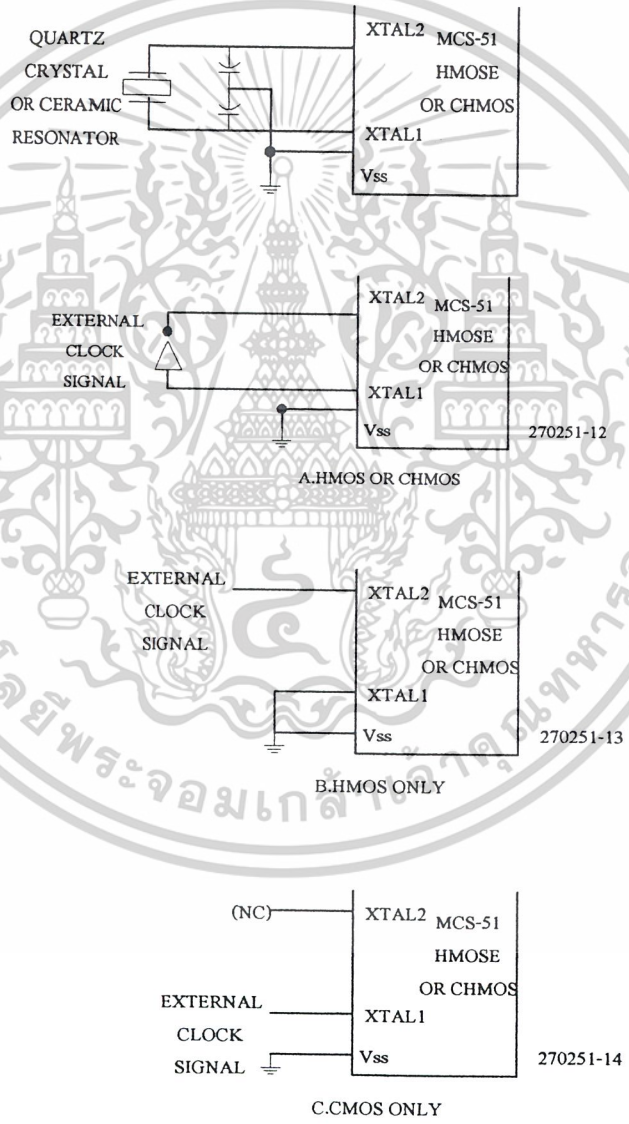


รูปที่ 2.11.5 การต่อขารีเซ็ทให้กับ 8051

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.12 วงจรสร้างสัญญาณนาฬิกาของ 8051

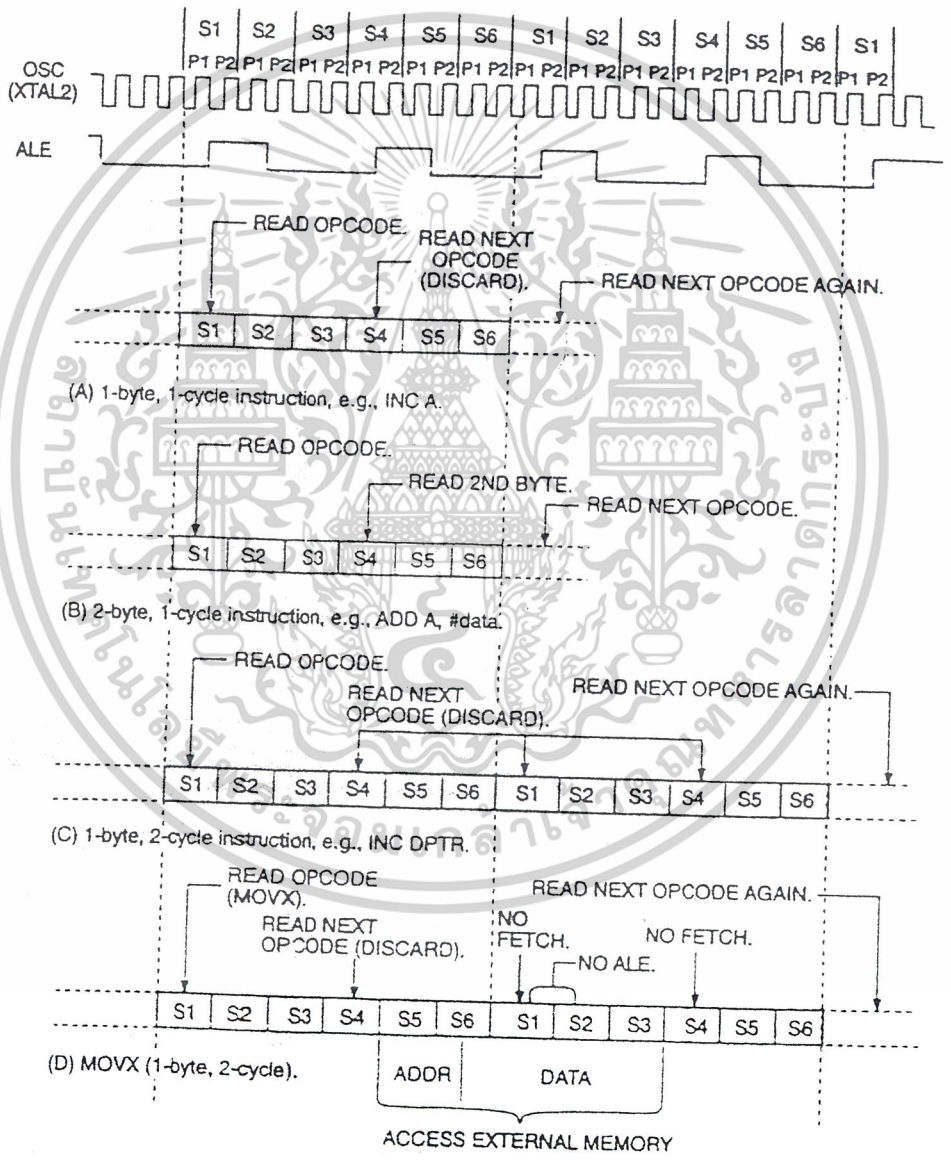
การต่อวงจรสร้างสัญญาณนาฬิกามีอยู่ด้วยกัน 2 รูปแบบ คือแบบภายในและภายนอกแสดงดังได้รูปที่ 2.12



รูปที่ 2.12 วงจรสร้างสัญญาณนาฬิกาของ 8051

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.13 ผังสัญญาณเวลาซีพียู (CPU Timming)



รูปที่ 2.13.1 ผังเวลาการทำงานของคำสั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซีพียูจะ RUN ด้วยความเร็วเท่ากับ 12 MHz ดังนั้น คลือก12 ลูกจะกินเวลาเท่ากับ (1/12)*12 = 1 μS) ว่า 1 แมชีนไซเคิล คือช่วงการทำงานตั้งแต่ S1 จนถึง S6

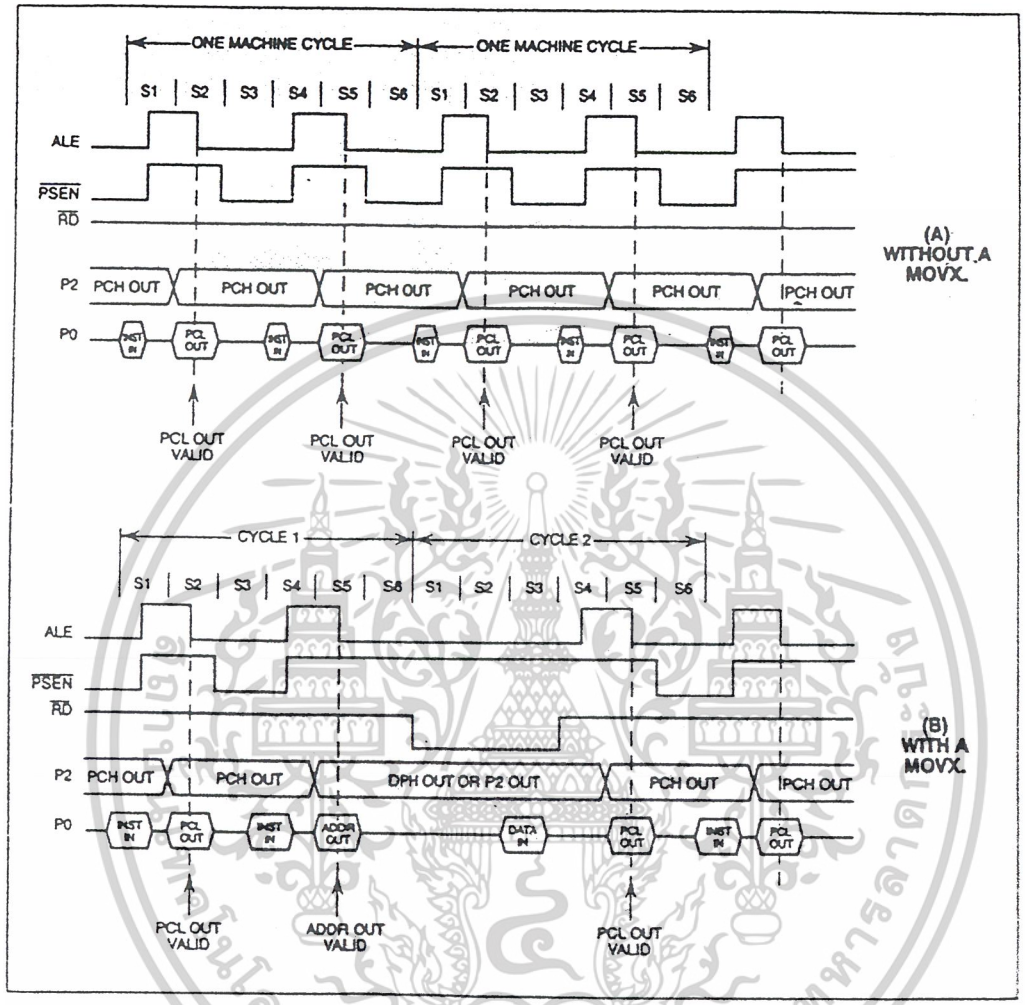
รูปที่ 2.13.1 (A) แสดงการทำงานของคำสั่ง INT A ซึ้งเป็นคำสั่ง 1 ไบต์ทำงาน เสร็จ ภายใน - 1 แมชีนไซเคิล

รูปที่ 2.13.1 (B) แสดงการทำงานของคำสั่ง ADD A.#Data ซึ่งเป็นคำสั่ง 2 ไบต์ทำงานเสร็จใน 1 แมชีนไซเคิล

รูปที่ 2.13.1 (C) แสดงการทำงานของคำสั่ง INC DPTR ซึ่งเป็นคำสั่ง 1 ไบต์ แต่ทำงานเสร็จใน 2 แมชีนไซเคิล

รูปที่ 2.13.1 (D) แสดงการทำงานของคำสั่ง MOVX ซึ่งเป็นคำสั่ง 1 ไบต์ แต่ทำงานเสร็จใน - 2 แมชีนไซเคิล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 213.2 แสดงผังเวลาการติดต่อกับหน่วยความจำภายนอก

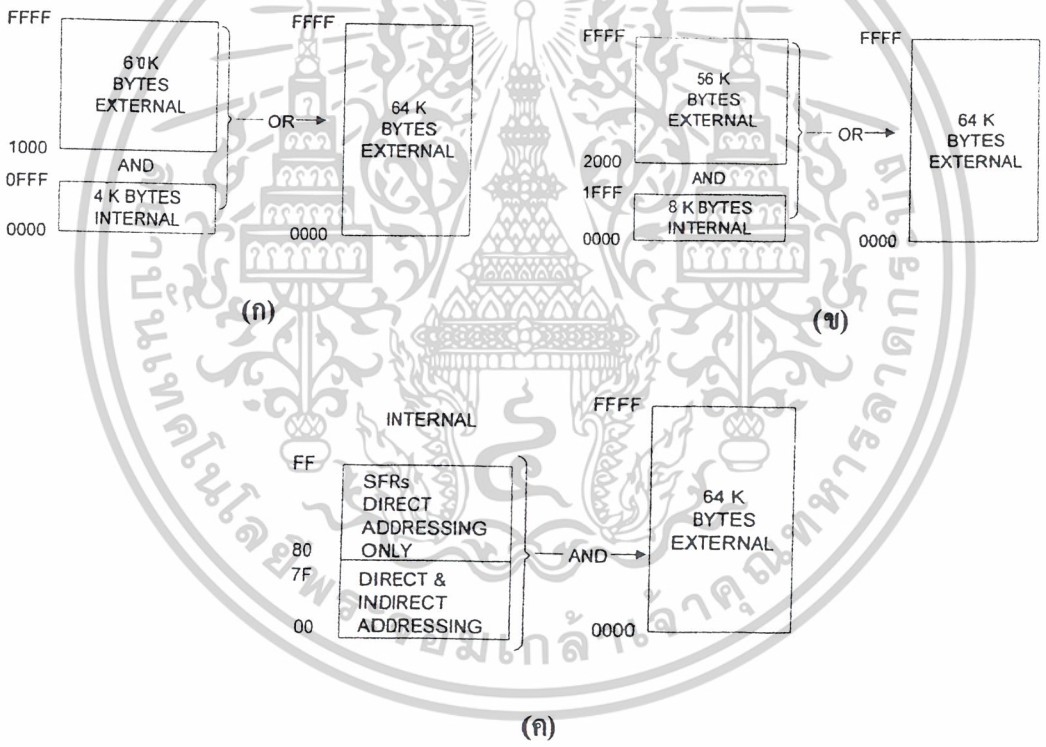
รูปที่ 2.13.2 (A) เป็นผังเวลาของสัญญาณซึ่งเกี่ยวข้องกับเฟิทซ์เมื่อส่วนของ Program Memory อยู่ภายนอก ดังนั้น สัญญาณที่จะนำมาใช้อ่าน OP-Code จาก Program Memory ก็คือ $\overline{PSEN}$ ซึ่งจะ-แอคทีฟ 2 ครั้งใน 1 แมชีนไซเคิล ดังนั้นสัญญาณที่ใช้อ่านข้อมูลจาก Program Memory จะใช้สัญญาณ $\overline{PSEN}$

รูปที่ 2.13.2 (B) เป็นผังเวลาของสัญญาณที่เกี่ยวข้องกับการอ่านข้อมูลจาก Data Memory โดยใช้สัญญาณ $\overline{RD}$ (อาจสรุปได้ง่าย ๆ ว่าการอ่านข้อมูลจาก Program Memory จะใช้สัญญาณ $\overline{PSEN}$ และการอ่านข้อมูลจาก Data Memory จะใช้สัญญาณ $\overline{RD}$ ส่วนสัญญาณ ALE คือสัญญาณที่ใช้ในการ Latch Address A0-A7 นั่นเอง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.14 การแบ่งประเภทของหน่วยความจำ

หน่วยความจำที่ใช้กับ MCS-51 มีอยู่ด้วยกัน 2 ชนิด คือ Program Memory และ Data Memory ซึ่งเป็นหน่วยความจำที่ใช้เก็บโปรแกรมสั่งงานบรรจุอยู่ในชิพ 8051 ส่วนที่เป็น Program Memory ก็คือ ROM ขนาด 4 กิโลไบต์นั่นเอง แต่ถ้าเป็นเบอร์ 8052 ก็คือ ROM ขนาด 8 กิโลไบต์ ดังแสดงในรูปที่-2.14(ก) และ ที่ 2.14(ข)



รูปที่ 2.14 (ก) ผังเก็บความจำสำหรับโปรแกรมสำหรับเบอร์ 8051

(ข) ผังเก็บความจำสำหรับโปรแกรมสำหรับเบอร์ 8052

(ค) ผังหน่วยความจำสำหรับ Data Memory เบอร์ 8051

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Data Memory เป็นหน่วยความจำที่ใช้เก็บข้อมูลหน่วยความจำนี้ สามารถเขียนข้อมูลลงไป และอ่านข้อมูลออกมาได้ ซึ่งเป็นหน่วยความจำภายในชิปมีเพียง 128 ไบต์ สำหรับเบอร์ 8051 และ 256 ไบต์ สำหรับเบอร์ 8052 ส่วนหน่วยความจำภายนอกชิพมี 64 กิโลไบต์ ดังแสดงในรูปที่ 2.14 (ค)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบและการทำงานของเครื่องวัดและบันทึกข้อมูลแบบไร้สาย

3.1 หลักการทำงานโดยรวม

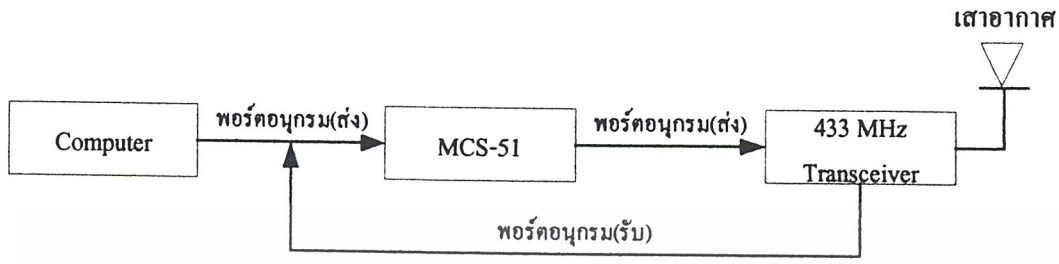
โครงสร้างของบอร์ดนั้น ตัวควบคุมและตัวประมวลผลส่วนกลางจะทำหน้าที่ควบคุมการทำงานของอุปกรณ์ในส่วนต่างๆได้แก่ หน่วยความจำข้อมูล,ตัวตรวจวัดอุณหภูมิ, ตัวตรวจวัดสัญญาณ และตัวสร้างฐานเวลาจริง เมื่อเริ่มทำงานตัวควบคุมและตัวประมวลผลส่วนกลางจะทำการอ่านค่าอุณหภูมิจากตัวตรวจวัดอุณหภูมิและค่าสัญญาณจากตัวตรวจวัดสัญญาณ แปลงข้อมูลให้อยู่ในรูปแบบที่เหมาะสม แล้วเก็บข้อมูลนั้นไว้ในหน่วยความจำข้อมูล อัตราการเก็บข้อมูลจะอิงเวลาจากตัวสร้างฐานเวลาจริงเพื่อให้ได้เวลาที่แน่นอนและตรงกับความเป็นจริง

3.2 บล็อคไดอะแกรมส่วนรับข้อมูลและสั่งงาน

วงจรในส่วนนี้ประกอบด้วย 2 วงจร ในการติดต่อกับคอมพิวเตอร์แบ่งออกเป็น 2 ส่วนคือ ส่วนรับและส่วนส่งข้อมูล การเปลี่ยนโหมดการทำงาน ในส่วนรับเมื่อ Transceiver ได้รับสัญญาณแล้วก็จะแปลงสัญญาณเป็นดิจิตอลติดต่อกับคอมพิวเตอร์โดยตรง โดยไม่ผ่าน MCS-51 ส่วนส่งโหมดคำสั่ง PC จะสั่งงานผ่านขา TXD ของ MCS-51 ส่งผ่านไปยัง Transceiver

พอร์ตอนุกรม RS232 ใช้สำหรับติดต่อกับคอมพิวเตอร์ ผู้ใช้งานสามารถสั่งงานผ่านงานทางคีย์บอร์ดและดูข้อมูลต่างๆได้ทางหน้าจอ โดยใช้โปรแกรมเทอร์มินอลทั่วไป วงจรในส่วนนี้จะใช้-อินเวอร์เตอร์ แต่เนื่องจากสัญญาณที่ได้รับจากพอร์ต RS232 มีแรงดันประมาณ ± 12 โวลต์ ซึ่งเกินกว่าแรงดันที่อินเวอร์เตอร์รับได้ คือ 0-5 โวลต์ จึงมีการต่อ R3,D1 และ D2 ไว้เพื่อควบคุมแรงดันของสัญญาณที่ขาอินพุตของอินเวอร์เตอร์ไม่ให้มากเกินไปจนทำให้เกิดความเสียหายแก่วงจรได้ ส่วนสัญญาณที่ส่งออกไปจากอินเวอร์เตอร์จะถูกคั่นด้วย R4 เพื่อจำกัดกระแสที่ไหลผ่าน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 บล็อคไดอะแกรมของส่วนรับข้อมูลและสั่งงาน

3.3 บล็อคไดอะแกรมของส่วนเครื่องมือวัด

การติดต่อกับอุปกรณ์ต่างๆที่กล่าวมานั้นจะใช้รูปแบบการติดต่อแบบบัสข้อมูลอนุกรมเพื่อลดจำนวนของสายสัญญาณในระบบ เนื่องจากต้องการให้บอร์ดมีขนาดเล็กกระทัดรัด เพื่อความสะดวกในการนำไปใช้งาน อีกทั้งอุปกรณ์ที่ใช้การรับส่งข้อมูลแบบอนุกรมเหล่านี้ นอกจากจะมีข้อดีในเรื่องของขนาดอุปกรณ์ที่เล็กแล้ว ในปัจจุบันยังได้รับการพัฒนาให้มีความสามารถในด้านการประหยัดพลังงาน คือ ใช้กระแสไฟฟ้าต่ำอีกด้วย

วงจรสมบูรณ์ของบอร์ด ตัวควบคุมและตัวประมวลผลส่วนกลางจะใช้ไมโครคอนโทรลเลอร์ MCS51 เบอร์ AT-89C4051 ซึ่งมีหน่วยความจำโปรแกรมภายในขนาด 4 กิโลไบต์ทำงานที่ความถี่ 11.059 เมกกะเฮิรตซ์ ควบคุมอุปกรณ์ต่างๆผ่านทางบัสข้อมูลอนุกรม อุปกรณ์ต่างๆเหล่านี้ได้แก่ หน่วยความจำข้อมูล ใช้หน่วยความจำเบอร์ 24LC256 ซึ่งเป็นหน่วยความจำชนิด EEPROM ขนาด 32 กิโลไบต์ สื่อสารข้อมูลโดยใช้การติดต่อผ่านบัส I^2C

ตัวสร้างฐานเวลาจริง ใช้ตัวสร้างสัญญาณนาฬิกาแบบเรียลไทม์ (RTC) เบอร์ DS1307 นอกจากจะทำหน้าที่สร้างฐานเวลาจริงแล้ว (โดยใช้คริสตอลความถี่ 32.768 กิโลเฮิรต์ในการสร้างสัญญาณนาฬิกา) ภายในยังมีหน่วยความจำ RAM ไว้ให้ใช้งานได้อีก 56 ไบต์ ซึ่งในการทำงานจะใช้เป็นที่เก็บค่าคอนฟิกต่างๆของบอร์ด การสื่อสารข้อมูลจะใช้บัส I^2C เช่นกัน

ตัววัดอุณหภูมิ เบอร์ DS1620 สามารถวัดอุณหภูมิได้ตั้งแต่ –55 ถึง +125 องศาเซลเซียส มีความละเอียด 0.5 องศาเซลเซียส โดยจะแปลงค่าอุณหภูมิที่วัดได้เป็นข้อมูลดิจิตอลขนาด 9 บิต สื่อสารข้อมูลผ่านการติดต่อแบบอนุกรมโดยใช้สายสัญญาณ 3 เส้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัววัดสัญญาณอนาล็อก ใช้ตัวแปลงสัญญาณอนาล็อกเป็นดิจิตอล (ADC) เบอร์ LTC1298 มีความละเอียดในการวัดถึง 12 บิต สื่อสารข้อมูลโดยใช้สายสัญญาณ 3 เส้น (โดยที่ขา DI และ DO ใช้สายสัญญาณร่วมกัน) สามารถวัดสัญญาณแบบขั้วเดียว (Single end) ได้ 2 ช่องสัญญาณคือ 12V,15V

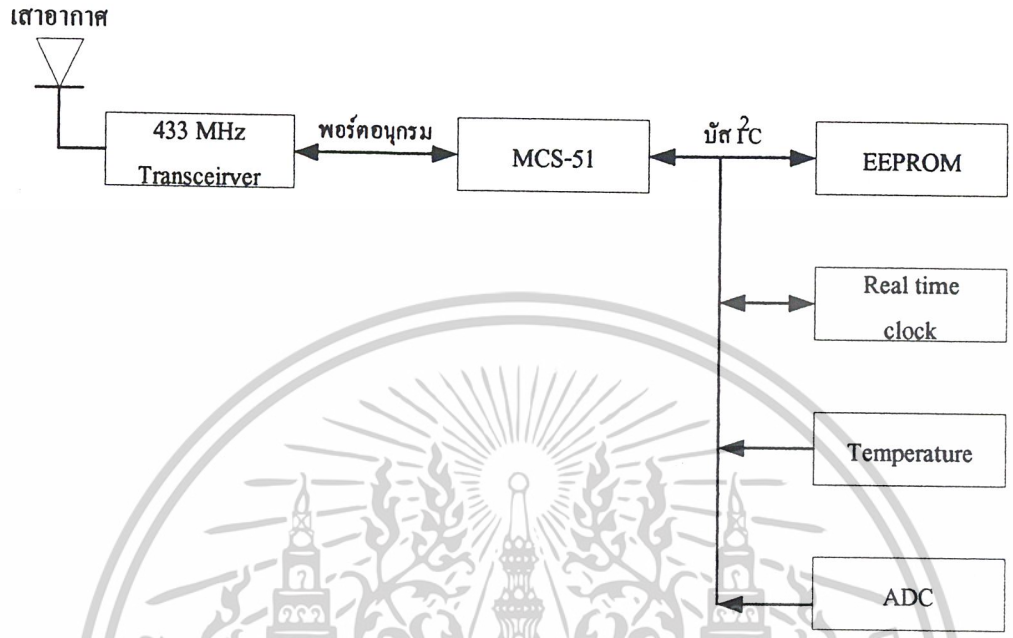
อุปกรณ์เหล่านี้สามารถนำมาต่อเป็นบัสข้อมูลอนุกรมโดยใช้สัญญาณนาฬิกา (SCL และCLK) และสัญญาณข้อมูล (SDA,และDQDI/DO) ร่วมกันได้ โดยต้องต่อความต้านทานพูลอัพ R1,R2 ไว้ เนื่องจากในระบบบัส I^2C ขา SCL และ SDA จะมีลักษณะเปิดเดรน (open-drain) เพื่อให้สามารถต่ออุปกรณ์หลายๆตัวเข้าด้วยกันเป็นบัสได้นั่นเอง ดังนั้นจึงจำเป็นที่จะต้องต่อความต้านทานพูลอัพไว้ที่ 2 ขานี้ด้วยเสมอ

การวัดค่าสัญญาณ ก่อนที่สัญญาณจะถูกส่งไปยังวงจร ADC จะผ่านวงจรกรองแรงดันต่ำ R8-C8 และ R9-C9 เพื่อลดสัญญาณรบกวน ในขณะที่ R11,R12 มีค่าสูงสุดไม่เกิน 5 โวลต์ ซึ่งเป็นแรงดันที่ขาอินพุตของวงจร ADC

ไฟเลี้ยงวงจรทั้งหมด ใช้ไฟ DC 8-12 โวลต์ต่อที่แจ็ค J1 ภายในบอร์ดจะมีไดโอดเรกดิไฟร์ช่วยป้องกันการต่อไฟเลี้ยงผิดขั้ว และมีเรกกูเลเตอร์ IC7 เบอร์ 78L05 เพื่อใช้สร้างแรงดันไฟเลี้ยง 5 โวลต์จ่ายให้กับวงจรทั้งหมด

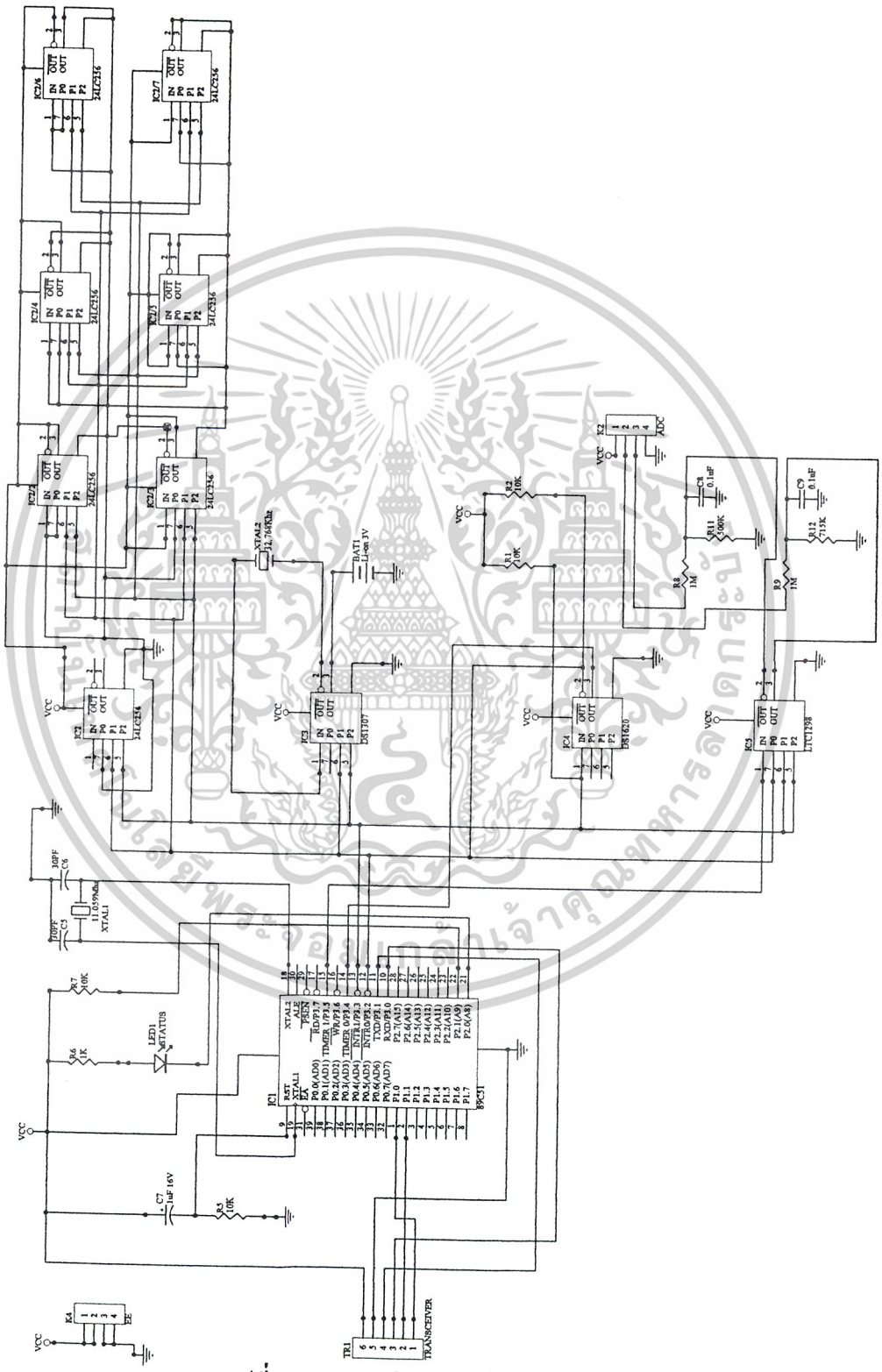
ไฟสำรองจ่ายให้วงจร RTC เนื่องจากวงจร RTC จำเป็นต้องทำงานอยู่ตลอดเวลาเพื่อรักษาค่าเวลาให้ถูกต้อง จึงจำเป็นต้องมีไฟเลี้ยงสำรองจ่ายให้กับวงจร RTC ในขณะที่ไม่ได้จ่ายไฟเลี้ยงตามปกติ (ไม่ได้ใช้งาน) โดยใช้แบตเตอรี่ ลิเธียม-ไอออน 3 โวลต์ ต่อแบ็กอัพไว้ค่าสูงสุดไม่เกิน 5 โวลต์ ซึ่งเป็นแรงดันที่ขาอินพุตของวงจร ADC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



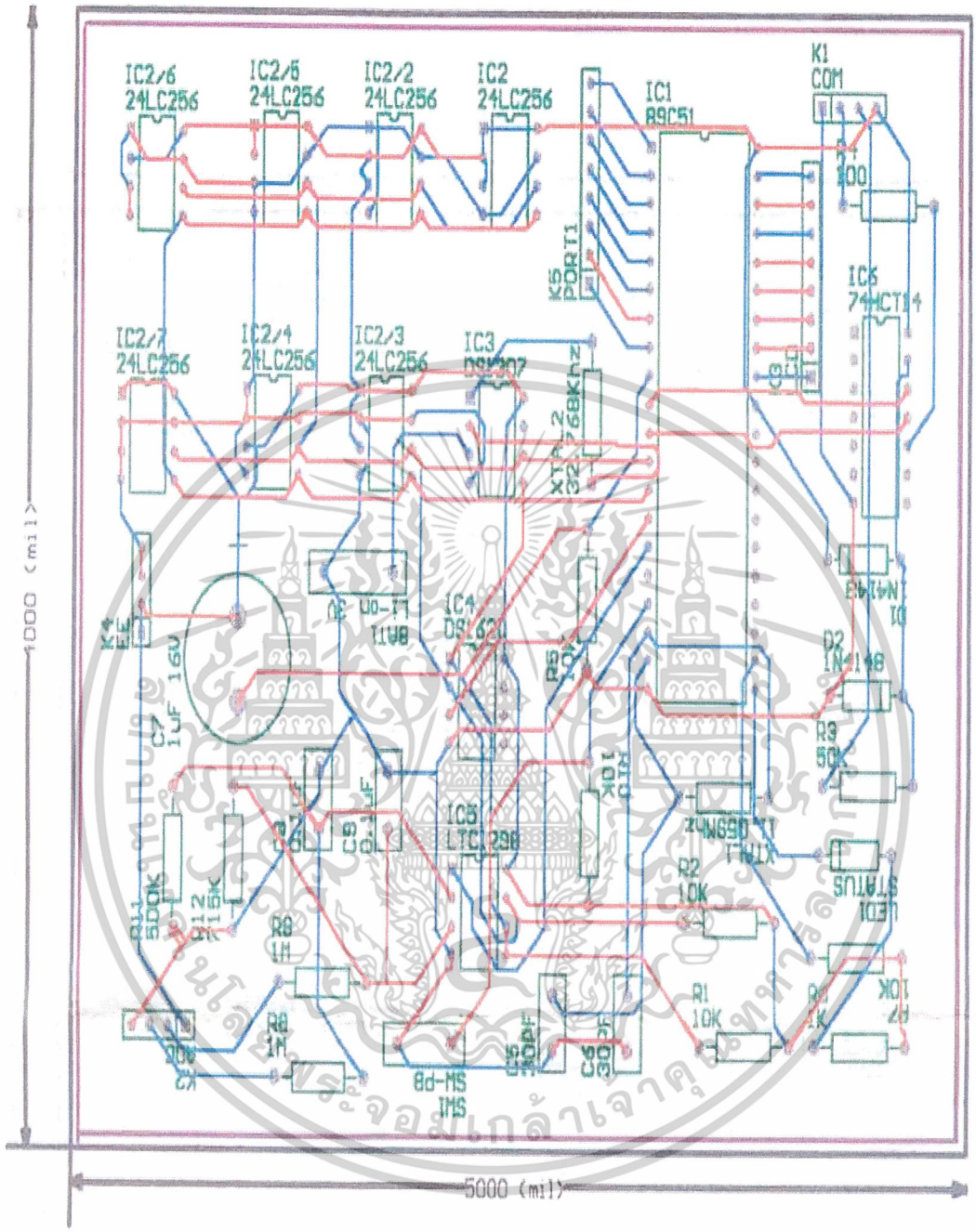
รูปที่ 3.3.1 บล็อคไดอะแกรมของส่วนตรวจวัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



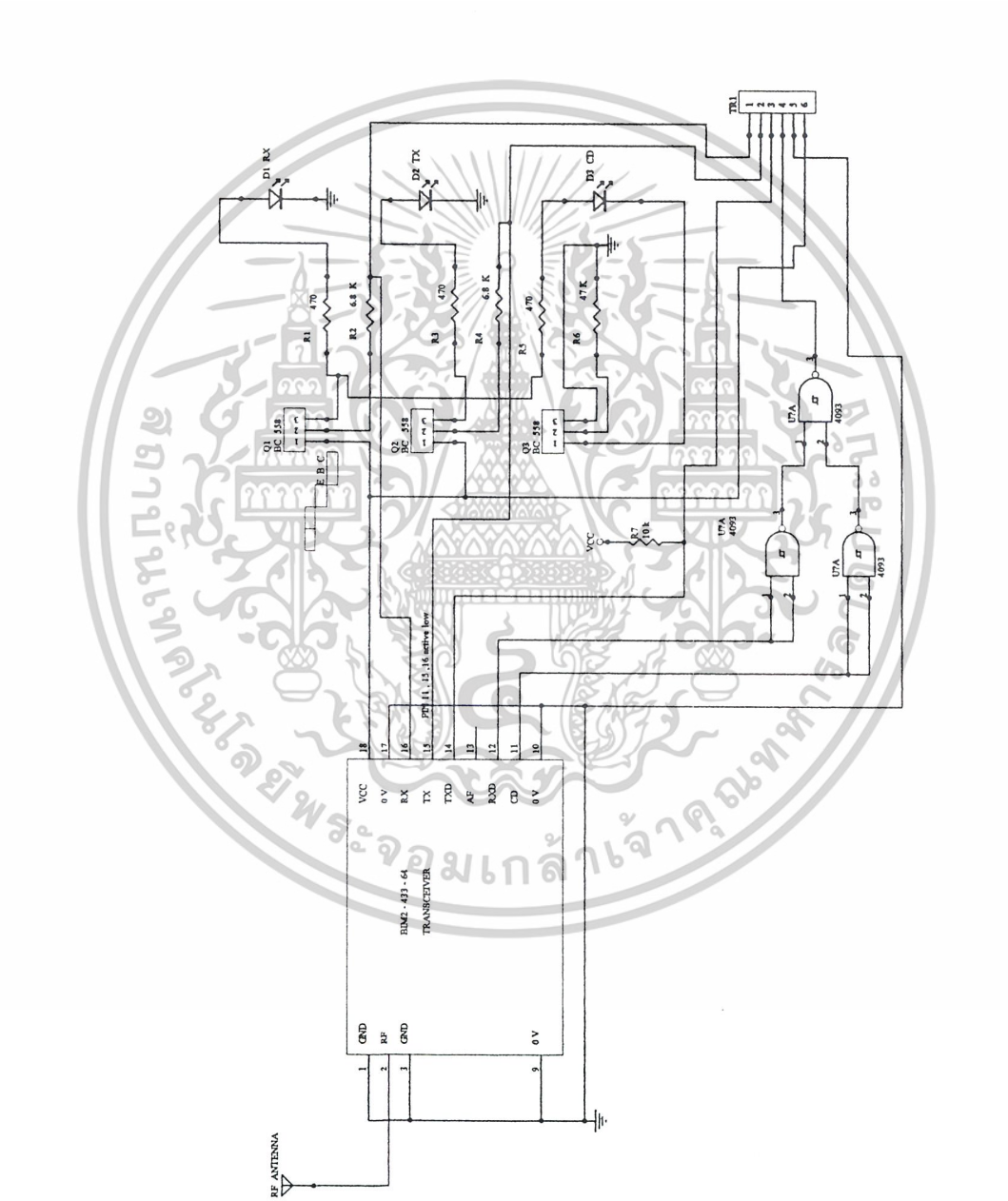
รูปที่ 3.3.2 วงจรส่วนตรวจวัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



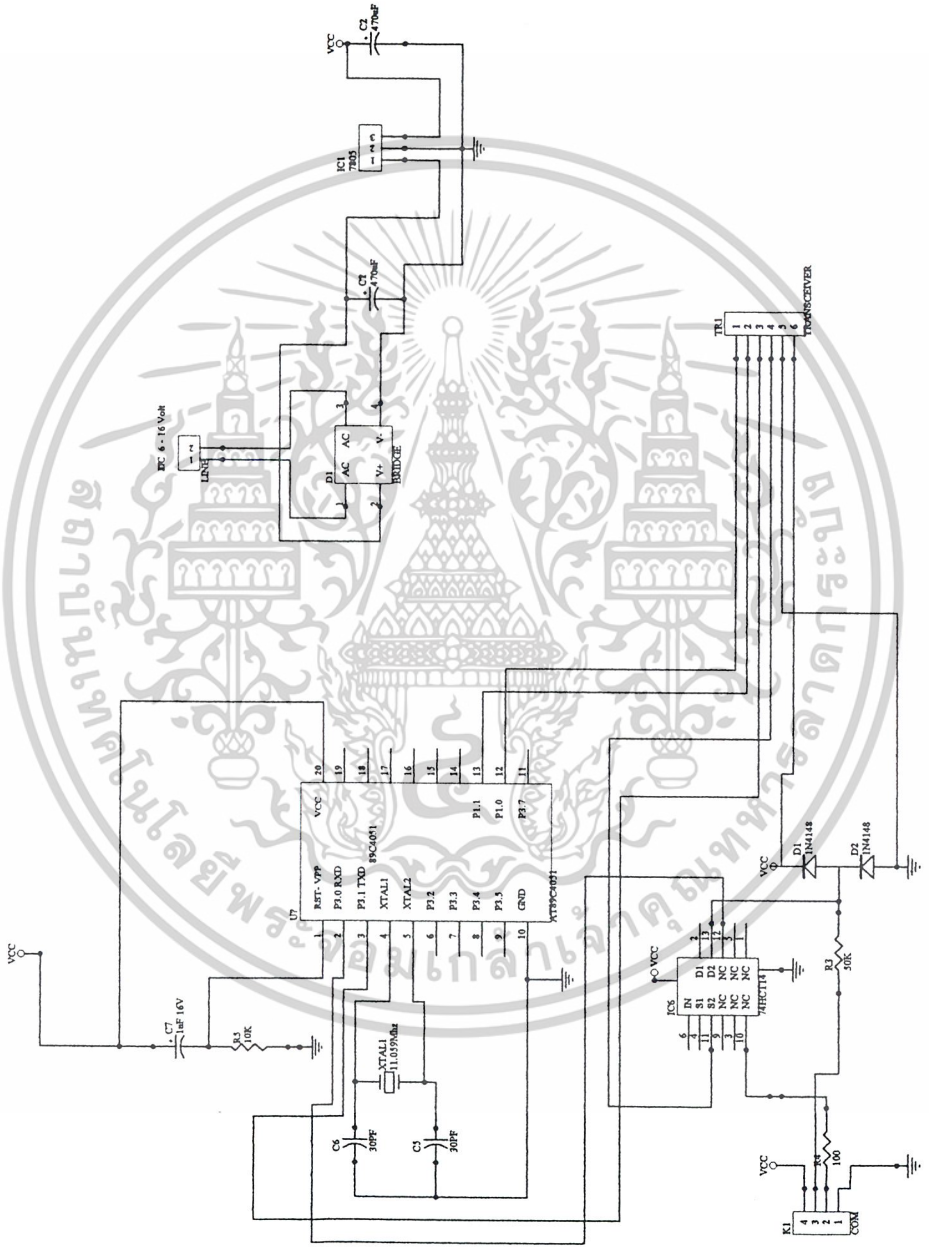
รูปที่ 3.3.3 การวางอุปกรณ์ส่วนตรวจวัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3. 3.4 วงจร Transceiver

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3.5 วงจรรับข้อมูลและสั่งงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

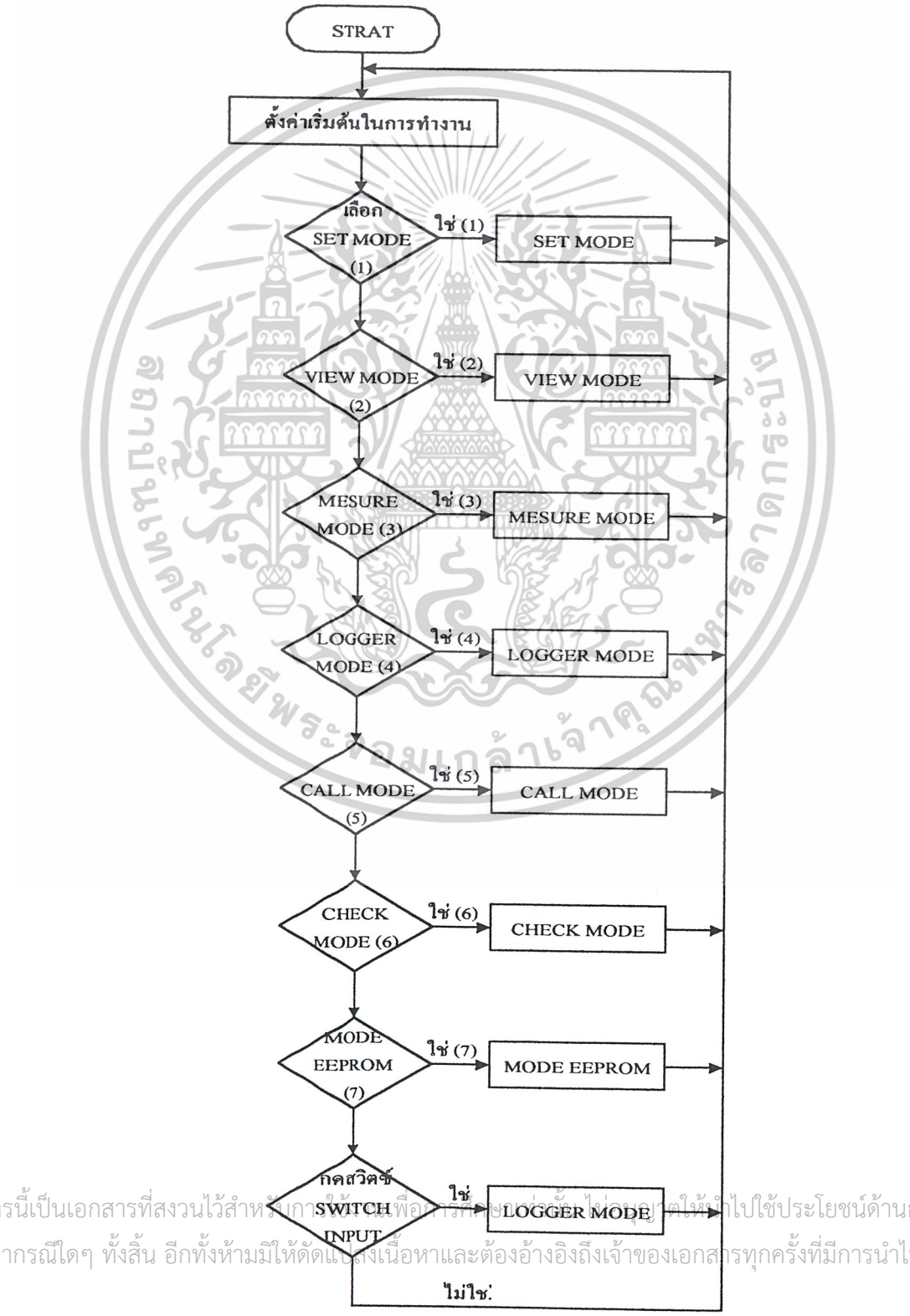
บอร์ดในการวัดในที่นี้ใช้การอ้างตำแหน่งแบบ 7 บิตแบบ I^2C และการติดต่อผ่านพอร์ท RS232 เพื่อลดจำนวนสายและขนาดของบอร์ด ให้มีขนาดเล็กกระทัดรัด เพื่อความสะดวกในการใช้งาน อีกทั้งในปัจจุบันยังได้รับการพัฒนาให้มีความสามารถในด้านการประหยัดพลังงาน คือ ใช้กระแสไฟฟ้าต่ำอีกด้วย

ถ้าทำการเก็บค่าอุณหภูมิและสัญญาณทางไฟฟ้า 1 วินาทีต่อครั้ง จะเก็บได้ 1 EEPROM เท่ากับ 2 ชั่วโมง 17 นาที ความละเอียดในการเก็บได้ 1 ถึง 255 วินาทีต่อครั้ง ความละเอียด 1 วินาที และถ้าต้องการเก็บตลอด 24 ชั่วโมง ในบอร์ดนี้มี EEPROM 7 ตัว ก็จะทำการเก็บที่ความละเอียด 2 วินาที หรือถ้าต้องการเก็บแบบ 1 วินาทีต่อครั้งก็จะต้องเพิ่ม EEPROM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

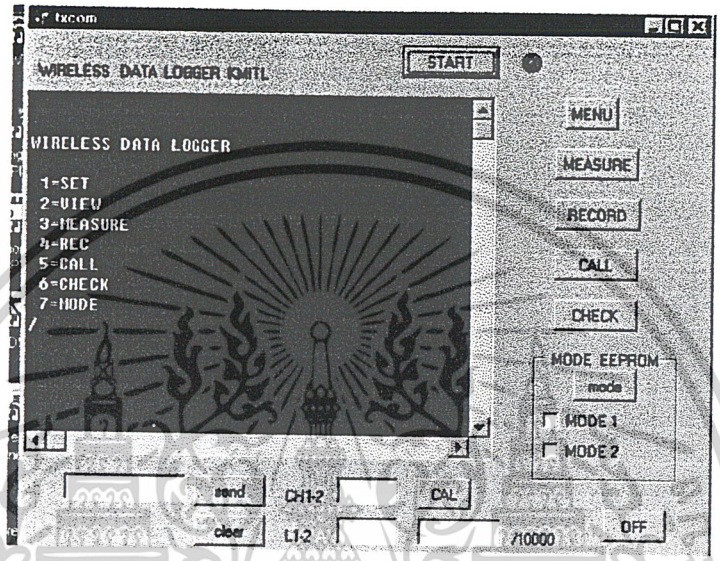
4.1 โปรแกรมควบคุมการทำงาน

โปรแกรมควบคุมการทำงานหลัก



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การใช้งานโปรแกรมการทำงาน



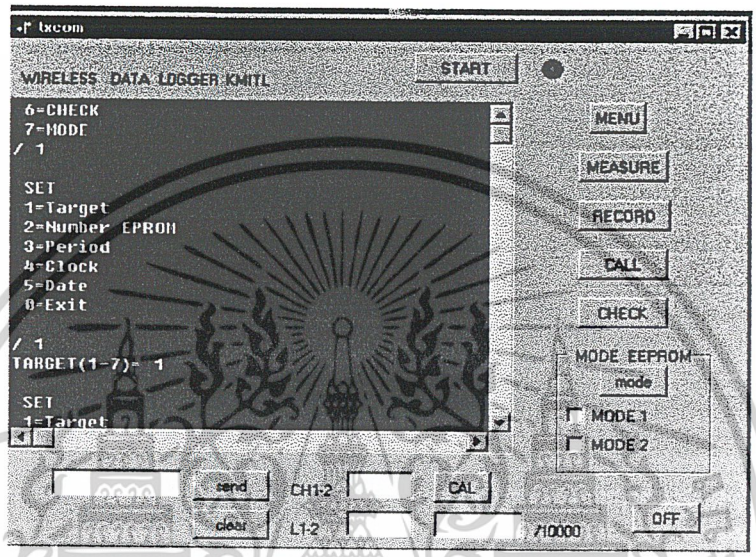
รูปที่ 4.2 หน้าต่างการทำงาน

การทำงาน เมื่อเราเข้าหน้าต่างการทำงานแล้วกด START และ MENU ตัวโปรแกรมก็จะขึ้นโหมดต่างๆทั้ง 7 โหมด คือ

1) SET MODE คือการเข้าไปเซตค่าโหมดต่างๆ
2) VIEW MODE คือการเช็คสภาพค่าต่างๆของ EEPROM ว่าตัวใดทำงานและอยู่ในโหมดใด
3) MEASURE MODE คือการวัดค่าเพียงอย่างเดียว ไม่มีการบันทึก
4) RECORD MODE คือการวัดและบันทึกผล
5) CALL MODE คือการเรียกทั้ง EEPROM โดยกำหนดเป้าหมายแบบเดียวกับการบันทึก ขั้นตอนการเซตโหมด
6) CHECK MODE คือการเช็คค่าที่เก็บครั้งสุดท้ายเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1 การทำงานของปุ่มต่างบนหน้าต่าง



รูปที่ 4.2.1 SET MODE

การเลือกโหมดการทำงาน กด 1 SET MODE เข้าไปในเซตโหมด

1) Target = เป้าหมายที่จะ ใช้(เริ่ม)

2) Number EEPROM = จำนวน EEPROM ที่จะใช้

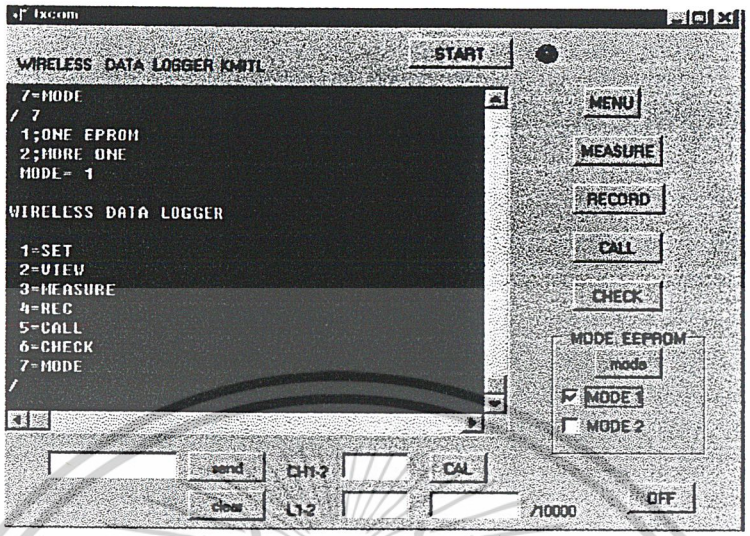
3) Period = เวลาความถี่ที่จะเก็บ

4) Clock = เวลา

5) Date = วันที่

6) Exit = ออก

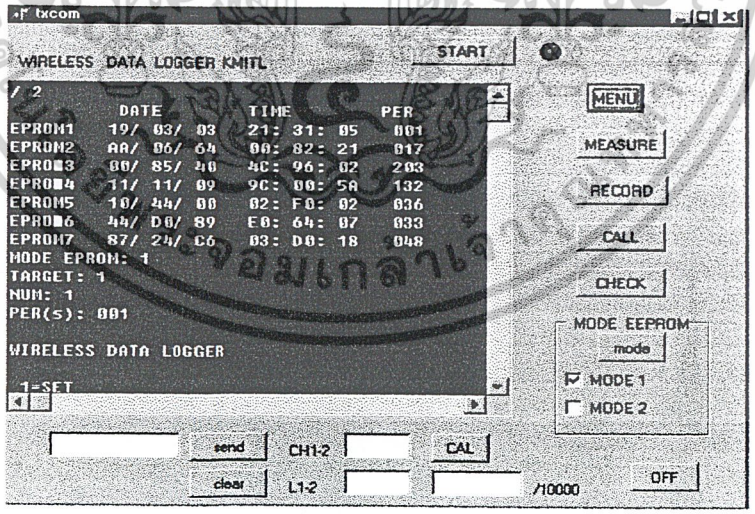
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2.2 EEPROM MODE

- ถ้าต้องการใช้ EEPROM 1 ตัวให้คลิกที่โหมด 1
- ถ้าต้องการใช้ EEPROM มากกว่า 1 ตัวคลิกที่โหมด 2

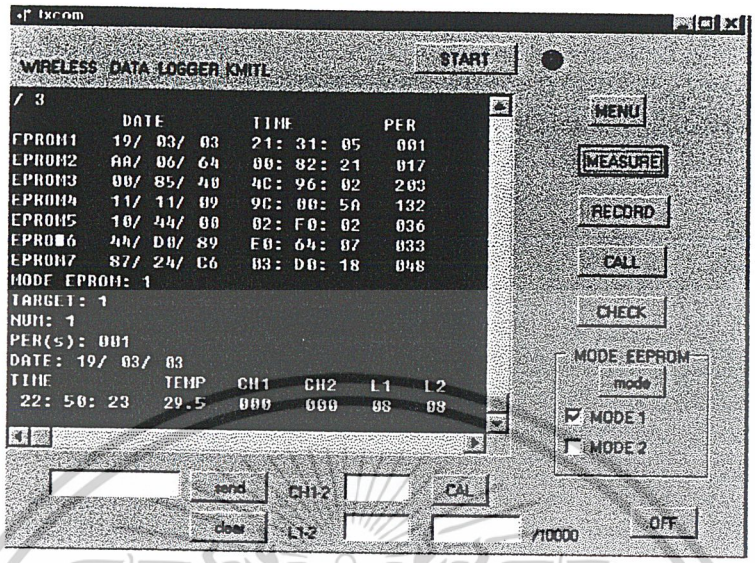
แต่ในที่นี้ขอยกตัวอย่างการใช้ EEPROM 1 ตัว ดังรูปที่ 4.2.2



รูปที่ 4.2.3 MENU MODE

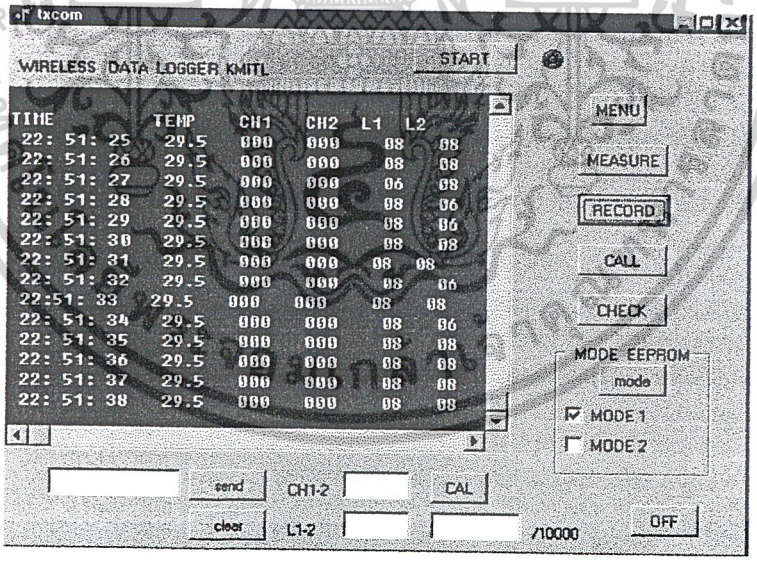
- MENU MODE เพื่อดูรูปแบบการเซตค่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่4.3.4 MEASURE MODE

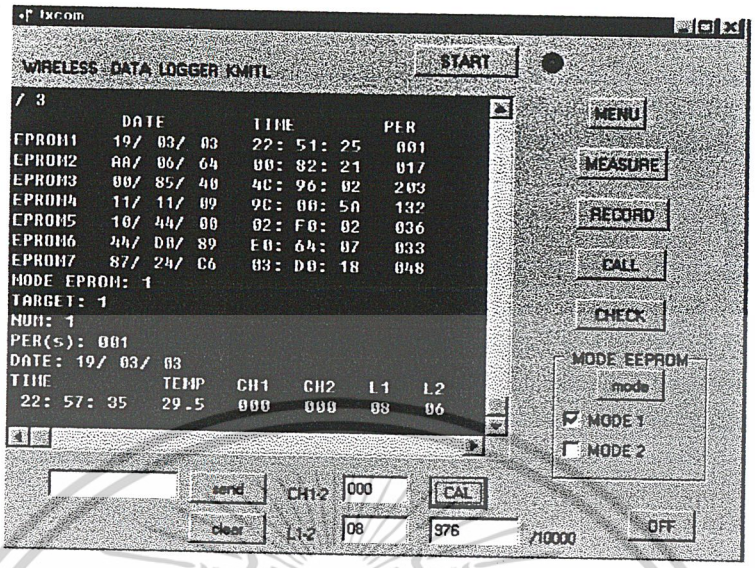
- MEASURE MODE เป็นการวัดอย่างเดียวไม่มีการบันทึกค่า



รูปที่ 4.2.5 RECORD MODE

- RECORD MODE เป็นการวัดและบันทึกผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2.6 CALL MODE

- CALL MODE เป็นการเรียกทั้ง EEPROM โดยเป้าหมายเดียวกับการบันทึกในขั้นตอนการเซตโหมด
- CHECK MODE เป็นการเรียกค่าครั้งสุดท้ายว่ามีการเก็บที่ EEPROM ตัวใดและเวลาเท่าใด
- ในที่นี้ ที่เวลา 22 : 57 :35 เก็บอุณหภูมิอยู่ที่ 29.5 เซลเซียส ค่า CH1 เป็น 000 และค่า L1 เป็น 08 จากนั้นนำค่ามาคำนวณดังรูปที่ 4.2.6 จะได้ค่าเท่ากับ0.0976 โวลต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทวิจารณ์และสรุป

5.1 บทวิจารณ์และสรุป

ในการทำวิทยานิพนธ์นั้น ในครั้งนี้เป็นแนวทางเพื่อแสดงถึงเครื่องวัดและบันทึกข้อมูลแบบไร้สาย โดยใช้ความถี่ 433 MHz มีการรับและส่งค่าตลอดจนเปลี่ยนโหมดการทำงานต่างๆได้ทันทีจากคอมพิวเตอร์

ในการติดต่อใช้การติดต่อแบบพอร์ทอนุกรม RS232 การอ้างตำแหน่งอุปกรณ์ก็จะใช้แบบ I^2C เป็นการอ้างตำแหน่งแบบ 7 บิต การวัดสัญญาณทางไฟฟ้าวัดได้ 2 ช่องสัญญาณ จะเปลี่ยนเป็นสัญญาณดิจิตอลโดยจะวัดเป็น 8 บิตบนและ 4 บิตล่าง โดยเราสามารถนำค่าที่หาได้มาใส่ในหน้าต่างการทำงานแล้วกดคำนวณ เครื่องจะคำนวณออกมาเป็นสัญญาณทางไฟฟ้าที่รับมาได้ ตัววัดอุณหภูมิในบอร์ดสามารถประยุกต์ใช้โดยการต่อออกไปนอกบอร์ดเพื่อวัดร่วมกับค่าสัญญาณทางไฟฟ้าได้ แต่ในที่นี้จะติดไว้กับบอร์ดเพื่อเช็คอุณหภูมิในส่วนบอร์ดติดตั้ง การเก็บค่าถ้ามีช่วงการเก็บอยู่ที่ 1-255 วินาทีต่อครั้ง ความละเอียด 1 วินาที เราจึงสามารถตั้งความละเอียดในการเก็บตามความเหมาะสม นอกจากนี้เรายังสามารถนำค่าที่เก็บบันทึกไว้ไปพล็อตในโปรแกรม Microsoft Excel

5.2 สรุปผลการทำงาน

โครงงานเครื่องวัดและบันทึกข้อมูลแบบไร้สายสามารถปฏิบัติงานได้ตามวัตถุประสงค์ โดยการวัดอุณหภูมิและค่าสัญญาณส่งผ่านมาทางความถี่ ทำให้ผู้ใช้งานไม่จำเป็นต้องเดินสายให้ยุ่งยาก ซึ่งจากแนวความคิดนี้เราสามารถประยุกต์และพัฒนาเพื่อให้เกิดประโยชน์ในงานระบบควบคุมในด้านอื่นๆอีกมากมาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

โปรแกรมควบคุมเครื่องวัดและบันทึกข้อมูลแบบไร้สาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
; Parameter definitions
BAUD  EQU  256-6        ; 256-1 -> baud rate = 57,600 bps
                                    ; 256-3 -> baud rate = 19,200 bps
                                    ; 256-6 -> baud rate = 9,600 bps
                                    ; 256-24 -> baud rate = 2,400 bps
ADDR24      EQU   0AEh   ; device address for 24C256 EEPROM 1-7
ADDR242 EQU 0ACh
ADDR243 EQU 0AAh
ADDR244 EQU 0A8h
ADDR245 EQU 0A6h
ADDR246 EQU 0A4h
ADDR247 EQU 0A2h
ADDR13      EQU   0D0h   ; device address for DS1307 RTC
YEAR  EQU   06h
MONTH       EQU   05h
DATE  EQU   04h
DAY         EQU   03h
HOUR  EQU   02h
MINUTE      EQU   01h
SECOND      EQU   00h
NUMREC      EQU   37h
TARGET      EQU   3Fh

; Register definitions
INDEX       EQU   0          ; index register
ZDATA       EQU   1          ; data transfer register

; I/O pin definitions
SCL             BIT  P3.2    ; serial clock
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SDA             BIT  P3.3    ; serial data
RST             BIT  P3.4    ; reset DS1620
CS              BIT  P3.5    ; chip select LTC1298
SW              BIT  P3.7    ; switch input
LED             BIT  P2.0    ; LED output
MODE        BIT      P2.1    ; mode select

; Memory allocations
STACK       EQU   07h    ; first stack
ADDR_RTC    EQU   30h    ; RTC address register
ADDR_LO     EQU 31h      ; 2-byte EEPROM address register
ADDR_HI     EQU 32h      ;
DATA_LO     EQU 33h
DATA_HI     EQU 34h
COUNT           EQU   35h
BUFF1       EQU   36h
BUFF2    EQU37h
BUFFD       EQU   38h
HOU         EQU 39h
MIN         EQU 3Ah
SEC         EQU 3Bh
PER         EQU 3Ch
NUM         EQU 3Dh
BUFF3    EQU 28h   ;fix eeprom
BUFF4    EQU 29h   ;mode eeprom 1,2
BUFF5    EQU 2Ah   ;eeprom 7
BUFF6    EQU 2Bh   ;num rec
                ORG   0000h          ; power-on reset vector

ON_RESET:
            mov   SP, #STACK
    mov  PCON,#80h
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        mov   TMOD,#20h
        mov   TH1,#BAUD
        mov   TL1,#BAUD
        setb  TR1
        mov   SCON,#52h
        mov   BUFF5,#07h

                setb    CS
                clr     RST
                mov R5,#00H
        mov R4,#00H
        mov 40H,#00H
        mov 42H,#00H
                setb    SDA
                setb    SCL
                clr     LED


MAIN:   setb    LED
        lcall PRINT_SER
        db              0Dh,0Ah,0Dh,0Ah
                db              "WIRELESS DATA LOGGER",0Dh,0Ah
                db              0Dh,0Ah
                db              " 1=SET",0Dh,0Ah
                db              " 2=VIEW",0Dh,0Ah
                db              " 3=MEASURE",0Dh,0Ah
                db              " 4=REC",0Dh,0Ah
                db              " 5=CALL",0Dh,0Ah
                db              " 6=CHECK",0Dh,0Ah
                db              " 7=MODE",0Dh,0Ah
                db              "/:",0

SUB:    mov p1,#02h
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
READY:  jnb  SCON.0,CHK_SW
        lcall   RX1
CHK_KEY1: cjne      A,#31h,CHK_KEY2
    lcall DELAY
        lcall   TX2
        lcall   SET_MODE
        ajmp    MAIN
CHK_KEY2: cjne      A,#32h,CHK_KEY3
    lcall DELAY
        lcall TX2
        lcall   VIEW_MODE
        ajmp    MAIN
CHK_KEY3: cjne      A,#33h,CHK_KEY4
    lcall DELAY
        lcall TX2
        lcall   MEASURE_MODE
        ajmp    MAIN
CHK_KEY4: cjne      A,#34h,CHK_KEY5
    lcall DELAY
        lcall   TX2
        lcall   LOGGER_MODE
        ajmp    MAIN
CHK_KEY5: cjne      A,#35h,CHK_KEY6
    lcall DELAY
        lcall   TX2
        lcall   CALL_MODE
        ajmp    MAIN
CHK_KEY6: cjne      A,#36h,CHK_KEY7
    lcall DELAY
        lcall   TX2
        lcall   CHECK_MODE
        ajmp    MAIN
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CHK_KEY7: cjne      A,#37h,CHK_SW
      lcall DELAY
      lcall     TX2
            lcall   EE_MODE
            ajmp    MAIN

CHK_SW:     jb      SW,READY
            mov     A,#34h
            lcall DELAY
            lcall   TX2
            lcall   LOGGER_MODE
            ajmp    MAIN

SET_MODE:
      lcall PRINT_SER
      db          0Dh,0Ah,0Dh,0Ah
            db          " SET ",0Dh,0Ah
            db          " 1=Target",0Dh,0Ah
            db          " 2=Number EPROM",0Dh,0Ah
            db          " 3=Period",0Dh,0Ah
            db          " 4=Clock",0Dh,0Ah
            db          " 5=Date",0Dh,0Ah
            db          " 0=Exit",0Dh,0Ah
            db          0Dh,0Ah
            db          "/:",0
SET_SLC:  mov p1,#02h
      lcall   DELAY2
      lcall     RX1
CHK_KEY1S:
            cjne    A,#31h,CHK_KEY2S
            lcall   DELAY
            lcall   TX
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                lcall   SET_TARGET
                ajmp    SET_MODE
CHK_KEY2S:
                cjne    A,#32h,CHK_KEY3S
                lcall DELAY
                lcall   TX
                lcall   SET_NUMREC
                ajmp    SET_MODE
CHK_KEY3S:
                cjne    A,#33h,CHK_KEY4S
                lcall DELAY
                lcall   TX
                lcall   SET_PERIOD
                ajmp    SET_MODE
CHK_KEY4S:
                cjne    A,#34h,CHK_KEY5S
                lcall DELAY
                lcall   TX
                lcall   SET_CLOCK
                ajmp    SET_MODE
CHK_KEY5S:
                cjne    A,#35h,CHK_KEY0S
                lcall DELAY
                lcall   TX
                lcall   SET_DATE
                ajmp    SET_MODE

CHK_KEY0S:
                cjne    A,#30h,SET_SLC
                ret

SET_TARGET:
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        lcall   LINE_FEED
        lcall   PRINT_SER
        db              "TARGET(1-7)= ",0
        lcall  RX
        lcall  DELAY
        lcall  TX2
        dec     A
        anl     A,#07h
        inc A
        mov BUFF3,A
        dec A
        mov     ZDATA,A
        mov     ADDR_RTC,#TARGET
        lcall   WRITE_RTC
        ret

SET_NUMREC:
        lcall   LINE_FEED
        lcall   PRINT_SER
        db              "NUM(1-7)= ",0
    lcall  RX
        lcall  DELAY
        lcall   TX2
        dec     A
        anl     A,#07h
        inc A
        mov BUFF6,A
        dec A
        mov     ZDATA,A
        mov     ADDR_RTC,#NUMREC
        lcall   WRITE_RTC
        ret
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SET_PERIOD:
        lcall   LINE_FEED
        lcall   PRINT_SER
        db              "PER(s)(000-255)= ",0
    lcall RX
        lcall DELAY
        lcall   TX2
        anl     A,#0Fh
        mov     B,#100
        mul     AB
        mov     ZDATA,A
        lcall RX
  lcall DELAY
        lcall TX2
        anl     A,#0Fh
        mov     B,#10
        mul     AB
        add     A,ZDATA
        mov     ZDATA,A
        lcall RX
        lcall DELAY
        lcall   TX2
        anl     A,#0Fh
        add     A,ZDATA
        mov     ZDATA,A
        mov     ADDR_RTC,#TARGET
        lcall   READ_RTC
        anl     A,#07h
        mov     B,#8
        mul     AB
        add     A,#14
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        mov     ADDR_RTC,A
        lcall   WRITE_RTC
        ret

SET_CLOCK:
        lcall   LINE_FEED
        lcall   PRINT_SER
        db              "HOUR(00-23)= ",0
        lcall   RX_NUM
        mov     ZDATA,A
        mov     ADDR_RTC,#HOUR
        lcall   WRITE_RTC
        lcall   LINE_FEED
        lcall   PRINT_SER
        db              "MIN(00-59)= ",0
        lcall   RX_NUM
        mov     ZDATA,A
        mov     ADDR_RTC,#MINUTE
        lcall   WRITE_RTC
        lcall   LINE_FEED
        lcall   PRINT_SER
        db              "SEC(00-59)= ",0
        lcall   RX_NUM
        mov     ZDATA,A
        mov     ADDR_RTC,#SECOND
        lcall   WRITE_RTC
        lcall   RTC_CLOCK_ENABLE
        lcall   CONFIG_1620
        ret

SET_DATE:
        lcall   LINE_FEED
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        lcall   PRINT_SER
        db              "DATE(01-31)= ",0
        lcall   RX_NUM
        mov     ZDATA,A
        mov     ADDR_RTC,#DATE
        lcall   WRITE_RTC
        lcall   LINE_FEED
        lcall   PRINT_SER
        db              "MONTH(01-12)= ",0
        lcall   RX_NUM
        mov     ZDATA,A
        mov     ADDR_RTC,#MONTH
        lcall   WRITE_RTC
        lcall   LINE_FEED
        lcall   PRINT_SER
        db              "YEAR(00-99)= ",0
        lcall   RX_NUM
        mov     ZDATA,A
        mov     ADDR_RTC,#YEAR
        lcall   WRITE_RTC
        ret

VIEW_MODE:
        lcall   LINE_FEED
        lcall   PRINT_SER
        db              "      DATE      TIME      PER",0
        mov     COUNT,#7
LPV1:   lcall   LINE_FEED
        lcall   PRINT_SER
        db              "EPROM",0
        mov     A,#7
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
clr     C
subb    A,COUNT
add     A,#31h
lcall   TX
clr     C
subb    A,#31h
mov     B,#8
mul     AB
add     A,#8
mov     ADDR_RTC,a
mov     A,#20h
lcall   TX2
lcall   TX2
lcall   READ_RTC
inc     ADDR_RTC
lcall   TX_BYTE
mov     A,#'/'
lcall   TX2
lcall   READ_RTC
inc     ADDR_RTC
lcall   TX_BYTE
mov     A,#'/'
lcall   TX2
lcall   READ_RTC
inc     ADDR_RTC
lcall   TX_BYTE
mov     A,#20h
lcall   TX2
lcall   TX2
lcall   READ_RTC
inc     ADDR_RTC
lcall   TX_BYTE
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        mov     A,#':'
        lcall   TX2
        lcall   READ_RTC
        inc     ADDR_RTC
        lcall   TX_BYTE
        mov     A,#':'
        lcall   TX2
        lcall   READ_RTC
        inc     ADDR_RTC
        lcall   TX_BYTE
        mov     A,#20h
        lcall   TX2
        lcall   TX2
        lcall   READ_RTC
        lcall   TX_BASE10
        djnz    COUNT,LPV2
    ajmp    LPV3
LPV2: ajmp  LPV1
LPV3: lcall LINE_FEED
        lcall   PRINT_SER
        db          "MODE EPROM: ",0
    mov A,BUFF4
    orl A,#30h
    lcall TX
        lcall   LINE_FEED
        lcall   PRINT_SER
        db          "TARGET: ",0
        mov     ADDR_RTC,#TARGET
        lcall   READ_RTC
        anl     A,#07h
        mov     BUFF1,a
        add     A,#31h
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        lcall   TX
        lcall   LINE_FEED
        lcall   PRINT_SER
        db              "NUM: ",0
        mov     ADDR_RTC,#NUMREC
        lcall   READ_RTC
        anl     A,#07h
        add     A,#31h
        lcall   TX
        lcall   LINE_FEED
        lcall   PRINT_SER
        db              "PER(s): ",0
        mov     A,BUFF1
        mov     B,#8
        mul     AB
        add     A,#14
        mov     ADDR_RTC,a
        lcall   READ_RTC
        mov     PER,a
        lcall   TX_BASE10
        ret


MEASURE_MODE:
    lcall  VIEW_MODE
        lcall   START_1620
        lcall   LINE_FEED
        lcall   PRINT_SER
        db              "DATE: ",0
        mov     ADDR_RTC,#DATE
        lcall   READ_RTC
        lcall   TX_BYTE
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        mov     A,#'/'
        lcall   TX2
        mov     ADDR_RTC,#MONTH
        lcall   READ_RTC
        lcall   TX_BYTE
        mov     A,#'/'
        lcall   TX2
        mov     ADDR_RTC,#YEAR
        lcall   READ_RTC
        lcall   TX_BYTE
    lcall       LINE_FEED
        lcall   PRINT_SER
        db              "TIME     TEMP  CH1  CH2  L1  L2",0
    lcall       LINE_FEED
LPM:  mov A,#0Dh
    lcall  TX2
    mov         ADDR_RTC,#HOUR
        lcall   READ_RTC
        lcall   TX_BYTE
        mov     A,#':'
        lcall   TX2
        mov     ADDR_RTC,#MINUTE
        lcall   READ_RTC
        lcall   TX_BYTE
        mov     A,#':'
        lcall   TX2
        mov     ADDR_RTC,#SECOND
        lcall   READ_RTC
        lcall   TX_BYTE

        mov     A,#20h
        lcall   TX2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        lcall   TX2
        lcall   READ_1620
        mov     ZDATA,a
        clr     C
        subb    A,#20h
        jnc     JUMP1
        cpl     A
        inc     A
        push    ACC
        mov     A,#'-'
        lcall   TX2
        pop     ACC
JUMP1:  clr  C
        rrc     A
        mov     B,#10
        div     AB
        swap    A
        add     A,B
        lcall   TX_BYTE
        mov     A,#'.'
        lcall   TX2
        mov     A,ZDATA
        rrc     A
        mov     A,#30h
        jnc     JUMP2
        add     A,#5
JUMP2:  lcall   TX2

        mov     A,#20h
        lcall   TX2
        lcall   TX2
        clr     C
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        lcall   READ_ADC
        mov     A,DATA_LO
        swap    A
        mov     BUFFD,A
        mov     A,DATA_HI
        lcall   TX_BASE10

        mov     A,#20h
        lcall   TX2
        lcall   TX2
        setb    C
        lcall   READ_ADC
        mov     A,DATA_LO
        add     A,BUFFD
        mov     ZDATA,A
        mov     A,DATA_HI
        lcall  TX_BASE10
        lcall  SHOW_REM
        mov p1,#02h
        clr SCON.0
        lcall WAIT
        jnb     SCON.0,JUMP3
clr SCON.0
ret

JUMP3:  lcall   DELAY
    ajmp        LPM


LOGGER_MODE:
    lcall   VIEW_MODE
        lcall   START_1620
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        mov R5,#02H
        mov R4,#00H
        lcall   LINE_FEED
        lcall   PRINT_SER
        db              "TARGET: ",0
        mov     ADDR_RTC,#TARGET
        lcall   READ_RTC
        anl     A,#07h
        mov     BUFF1,A
        add     A,#31h
        lcall   TX
        lcall   LINE_FEED
        lcall   PRINT_SER
        db              "NUM: ",0
        mov     ADDR_RTC,#NUMREC
        lcall   READ_RTC
        anl     A,#07h
        add     A,#31h
        lcall   TX
        lcall   LINE_FEED
        lcall   PRINT_SER
        db              "PER(s): ",0
        mov     A,BUFF1
        mov     B,#8
        mul     AB
        add     A,#14
        mov     ADDR_RTC,a
        lcall   READ_RTC
        mov     PER,A
        lcall   TX_BASE10
LOGGER: lcall LINE_FEED
        mov     ADDR_RTC,#TARGET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
lcall   READ_RTC
anl     A,#07h
mov     B,#8
mul     AB
add     A,#8
mov     BUFF1,A
lcall   PRINT_SER
db              "DATE: ",0
mov     ADDR_RTC,#DATE
lcall   READ_RTC
mov     ZDATA,A
lcall   TX_BYTE
mov     ADDR_RTC,BUFF1
lcall   WRITE_RTC
inc     BUFF1
mov     A,#'/'
lcall   TX2
mov     ADDR_RTC,#MONTH
lcall   READ_RTC
mov     ZDATA,A
lcall   TX_BYTE
mov     ADDR_RTC,BUFF1
lcall   WRITE_RTC
inc     BUFF1
mov     A,#'/'
lcall   TX2
mov     ADDR_RTC,#YEAR
lcall   READ_RTC
mov     ZDATA,A
lcall   TX_BYTE
mov     ADDR_RTC,BUFF1
lcall   WRITE_RTC
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
inc     BUFF1

mov     ADDR_RTC,#HOUR
lcall   READ_RTC
mov     ZDATA,A
mov     ADDR_RTC,BUFF1
lcall   WRITE_RTC
inc     BUFF1
mov     ADDR_RTC,#MINUTE
lcall   READ_RTC
mov     ZDATA,A
mov     ADDR_RTC,BUFF1
lcall   WRITE_RTC
inc     BUFF1
mov     ADDR_RTC,#SECOND
lcall   READ_RTC
mov     ZDATA,A
mov     ADDR_RTC,BUFF1
lcall   WRITE_RTC
mov     ADDR_RTC,#TARGET
lcall   READ_RTC
anl     A,#07h
rl          A
swap    A
mov     ADDR_HI,A
mov     ADDR_LO,#0
mov     INDEX,#0
mov     ADDR_RTC,#NUMREC
lcall   READ_RTC
inc     A
mov     NUM,#04h
lcall   LINE_FEED
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        lcall   LINE_FEED
                lcall   PRINT_SER
                db              "TIME    TEMP   CH1  CH2 L1  L2",0
LPL:    clr     LED
                lcall   LINE_FEED
                lcall   LINE_FEED
                mov     ADDR_RTC,#HOUR
                lcall   READ_RTC
                lcall   TX_BYTE
                mov     A,#':'
                lcall   TX2
                mov     ADDR_RTC,#MINUTE
                lcall   READ_RTC
                lcall   TX_BYTE
                mov     A,#':'
                lcall   TX2
                mov     ADDR_RTC,#SECOND
                lcall   READ_RTC
                lcall   TX_BYTE
        mov     A,#20h
                lcall   TX2
                lcall   TX2
                lcall READ_1620
                mov     ZDATA,A
WA1:    lcall   WRITE_BYTE
                jc      WA1
                inc     ADDR_LO
                mov     A,ZDATA
                clr     C
                subb    A,#20h
                jnc     SKIP1
                cpl     A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        inc     A
        push    ACC
        mov     A,#'-'
        lcall   TX2
        pop     ACC
SKIP1:  clr  C
        rrc     A
        mov     B,#10
        div     AB
        swap    A
        add     A,B
        lcall   TX_BYTE
        mov     A,#'.'
        lcall   TX2
        mov     A,ZDATA
        rrc     A
        mov     A,#30h
        jnc     SKIP2
        add     A,#5
SKIP2:  lcall   TX2

        mov     A,#20h
        lcall   TX2
        lcall   TX2
        clr     C
        lcall   READ_ADC
        mov     ZDATA,DATA_HI
WA2:    lcall   WRITE_BYTE
        jc              WA2
        inc     ADDR_LO
        mov     A,DATA_LO
        swap    A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        mov     BUFFD,A
        mov     A,DATA_HI
        lcall   TX_BASE10

        mov     A,#20h
        lcall   TX2
        lcall   TX2
        setb    C
        lcall   READ_ADC
        mov     ZDATA,DATA_HI
WA3:    lcall   WRITE_BYTE
        jc              WA3
        inc     ADDR_LO
        mov     A,DATA_LO
        add     A,BUFFD
        mov     ZDATA,a
        mov     A,DATA_HI
        lcall   TX_BASE10
WA4:    lcall   WRITE_BYTE
        jc              WA4
        inc     ADDR_LO
        lcall   SHOW_REM
        clr     A
        cjne    A,ADDR_LO,SKIP3
        inc     ADDR_HI
        inc     INDEX
        mov     A,INDEX
        cjne    A,#20h,SKIP3
        mov     INDEX,#0
        djnz    NUM,SKIP3
        mov R2,BUFF4
        cjne R2,#01h,SPEC
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                lcall   LINE_FEED
                lcall   PRINT_SER
                db              "F!",0
                lcall RX2
                setb    LED
                jnb     SCON.0,$
                clr     SCON.0
        ret
SPEC:   mov NUM,#04h
        dec BUFF5
        djnz BUFF6,SKIP4
        ret

SKIP3:  setb LED
        mov p1,#02h
        clr SCON.0
        lcall WAIT
        jnb SCON.0,SKIP4
                clr     SCON.0
                ret

SKIP4:  inc R5
                cjne R5,#00H,chk
                inc R4
chk:    ljmp    LPL


CALL_MODE: lcall    VIEW_MODE
DUMP:   lcall   LINE_FEED
                lcall   PRINT_SER
                db              "DATE: ",0
                mov     ADDR_RTC,#TARGET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
lcall   READ_RTC
anl     A,#07h
mov     B,#8
mul     AB
add     A,#8
mov     ADDR_RTC,A
lcall   READ_RTC
inc     ADDR_RTC
lcall   TX_BYTE
mov     A,#'/'
lcall   TX2
lcall   READ_RTC
inc     ADDR_RTC
lcall   TX_BYTE
mov     A,#'/'
lcall   TX2
lcall   READ_RTC
lcall   TX_BYTE
inc     ADDR_RTC
lcall   READ_RTC
mov     HOU,A
inc     ADDR_RTC
lcall   READ_RTC
mov     MIN,A
inc     ADDR_RTC
lcall   READ_RTC
mov     SEC,A
mov     ADDR_RTC,#TARGET
lcall   READ_RTC
anl     A,#07h
rl              A
swap    A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        mov     ADDR_HI,A
        mov     ADDR_LO,#0
        mov     INDEX,#0
        mov     ADDR_RTC,#NUMREC
        lcall   READ_RTC
        inc     A
        mov     NUM,#04h
        lcall   LINE_FEED
        lcall   LINE_FEED
        lcall   PRINT_SER
        db              "TIME    TEMP  CH1 CH2  L1  L2",0
LPD:    lcall   LINE_FEED
    lcall   LINE_FEED
    lcall DELAY
        mov     A,HOU
        lcall   TX_BYTE
        mov     A,#':'
        lcall   TX2
        mov     A,MIN
        lcall   TX_BYTE
        mov     A,#':'
        lcall   TX2
        mov     A,SEC
        lcall   TX_BYTE

        mov     A,#20h
        lcall   TX2
    lcall       READ_RANDOM
        mov     ZDATA,A
        inc     ADDR_LO
        clr     C
        subb    A,#20h
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        jnc     SKIP6
        cpl     A
        inc     A
        push    ACC
        mov     A,#'-'
        lcall   TX2
        pop     ACC
SKIP6:  clr  C
        rrc     A
        mov     B,#10
        div     AB
        swap    A
        add     A,B
        lcall   TX_BYTE
        mov     A,#'.'
        lcall   TX2
        mov     A,ZDATA
        rrc     A
        mov     A,#30h
        jnc     SKIP7
        add     A,#5
SKIP7:  lcall   TX2
    mov     A,#20h
        lcall   TX2

        lcall   READ_RANDOM
        inc     ADDR_LO
        lcall   TX_BASE10

        mov     A,#20h
        lcall   TX2
        lcall   TX2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        lcall   READ_RANDOM
        inc     ADDR_LO
        lcall   TX_BASE10
        lcall   READ_RANDOM
        inc     ADDR_LO
        mov     ZDATA,A
        lcall   SHOW_REM
        clr     A
        cjne    A,ADDR_LO,SKIP8
        inc     ADDR_HI
        inc     INDEX
        mov     A,INDEX
        cjne    A,#20h,SKIP8
        mov     INDEX,#0
        djnz    NUM,SKIP8
        mov R2,BUFF4
        cjne R2,#01h,SPET
lcall   LINE_FEED
        lcall   PRINT_SER
        db              "F",0
        lcall RX2
        jnb     SCON.0,$
        clr     SCON.0
        ret

SPET:   mov NUM,#04h
    dec BUFF5
    djnz BUFF6,SKIP9
    lcall       PRINT_SER
                db              "F",0
                lcall  RX2
                jnb     SCON.0,$
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            clr     SCON.0
            ret

SKIP8:  mov p1,#02h
        clr SCON.0
        lcall DELAY3
        lcall DELAY3
        lcall   NEXT_TIME
                jnb     SCON.0,SKIP9
                clr     SCON.0
                ret
SKIP9:  ljmp    LPD

CHECK_MODE: lcall   VIEW_MODE
DUMP1:  cjne R5,#00H,loop2
        mov R5,40H
        mov R4,42H
        ajmp loop3
loop2:  mov 40H,R5
        mov 42H,R4
        cjne R5,#00H,loop3
        lcall   LINE_FEED
                lcall   PRINT_SER
                db              "NO",0
                ljmp MAIN

loop3:  lcall   LINE_FEED
                lcall   PRINT_SER
                db              "DATE: ",0
                mov     ADDR_RTC,#TARGET
                lcall   READ_RTC
                anl     A,#07h
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
mov     B,#8
mul     AB
add     A,#8
mov     ADDR_RTC,A
lcall   READ_RTC
inc     ADDR_RTC
lcall   TX_BYTE
mov     A,#'/'
lcall   TX2
lcall   READ_RTC
inc     ADDR_RTC
lcall   TX_BYTE
mov     A,#'/'
lcall   TX2
lcall   READ_RTC
lcall   TX_BYTE
inc     ADDR_RTC
lcall   READ_RTC
mov     HOU,A
inc     ADDR_RTC
lcall   READ_RTC
mov     MIN,A
inc     ADDR_RTC
lcall   READ_RTC
mov     SEC,A
mov     ADDR_RTC,#TARGET
lcall   READ_RTC
anl     A,#07h
rl              A
swap    A
mov     ADDR_HI,A
mov     ADDR_LO,#0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        mov     INDEX,#0
        mov     ADDR_RTC,#NUMREC
        lcall   READ_RTC
        inc     A
        mov     NUM,#04h

        lcall  LINE_FEED
        lcall   LINE_FEED
        lcall   PRINT_SER
        db              "TIME    TEMP  CH1  CH2  L1  L2 ",0
LPD1:  djnz R5,LPD2
    cjne R4,#00H,loop1
    ljmp MAIN
loop1:  mov R5,#0FFH
    djnz R4,LPD2
LPD2:  lcall DELAY3
    lcall DELAY3
    lcall  LINE_FEED
        mov     A,HOU
        lcall   TX_BYTE
        mov     A,#':'
        lcall   TX2
        mov     A,MIN
        lcall   TX_BYTE
        mov     A,#':'
        lcall   TX2
        mov     A,SEC
        lcall   TX_BYTE

        mov     A,#20h
        lcall   TX2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        lcall   READ_RANDOM
        mov     ZDATA,A
        inc     ADDR_LO
        clr     C
        subb    A,#20h
        jnc     SKIPS6
        cpl     A
        inc     A
        push    ACC
        mov     A,#'-'
        lcall   TX2
        pop     ACC
SKIPS6: clr  C
        rrc     A
        mov     B,#10
        div     AB
        swap    A
        add     A,B
        lcall   TX_BYTE
        mov     A,#'.'
        lcall   TX2
        mov     A,ZDATA
        rrc     A
        mov     A,#30h
        jnc     SKIPS7
        add     A,#5
SKIPS7: lcall   TX2

        mov     A,#20h
        lcall   TX2
        lcall   TX2
        lcall   READ_RANDOM
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
inc     ADDR_LO
lcall   TX_BASE10

mov     A,#20h
lcall   TX2
lcall   TX2
lcall   READ_RANDOM
inc     ADDR_LO
lcall   TX_BASE10
lcall   READ_RANDOM
inc     ADDR_LO
mov     ZDATA,A
lcall   SHOW_REM
clr     A
cjne    A,ADDR_LO,SKIPS8
inc     ADDR_HI
inc     INDEX
mov     A,INDEX
cjne    A,#20h,SKIPS8
mov     INDEX,#0
djnz    NUM,SKIPS8
mov R2,BUFF4
cjne R2,#01h,SPET1
lcall   LINE_FEED
lcall   PRINT_SER
db              "F",0
lcall RX2
jnb     SCON.0,$
clr     SCON.0
ret

SPET1:  mov NUM,#04h
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        dec BUFF5
        djnz BUFF6,SKIPS9
                lcall   PRINT_SER
                db              "F",0
        lcall RX2
            jnb   SCON.0,$
                clr     SCON.0
                ret

SKIPS8:  mov p1,#02h
        clr SCON.0
        lcall DELAY3
        lcall DELAY3
        lcall       NEXT_TIME
                jnb     SCON.0,SKIPS9
                clr     SCON.0
                ret
SKIPS9:  lcall DELAY3
        ljmp        LPD1

EE_MODE: lcall LINE_FEED
        lcall PRINT_SER
        db " 1;ONE EPROM",0Dh,0Ah
        db " 2;MORE ONE",0Dh,0Ah
        db " MODE= ",0
        lcall RX
        lcall DELAY
        lcall TX2
        anl A,#03h
        mov BUFF4,A
        ret
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SHOW_REM: mov    A,#20h
          lcall  TX2
          lcall  TX2
          mov    A,ZDATA
          anl    A,#0F0h
          swap   A
          mov    B,#10
          div    AB
          swap   A
          add    A,B
          lcall  TX_BYTE

          mov    A,#20h
          lcall  TX2
          lcall  TX2
          mov    A,ZDATA
          anl    A,#0Fh
          mov    B,#10
          div    AB
          swap   A
          add    A,B
          lcall  TX_BYTE
          ret

LINE_FEED: mov   A,#0Dh
           lcall TX
           mov   A,#0Ah
           lcall TX
           ret

TX:   mov p1,#01h
      lcall DELAY
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        lcall TX1
        mov p1,#03h
        ret
TX1:    jnb  SCON.1,TX1
        clr  SCON.1
        mov  SBUF,A
        ret
TX2:    mov p1,#01h
                lcall DELAY
                lcall TX1
                lcall DELAY2
                mov p1,#03h
        ret
RX:     mov p1,#02h
path:   lcall       RX1
        mov R7,A
                anl A,#0F0h
                cjne A,#30h,path
                mov A,R7
        mov p1,#03h
        ret
RX1:    jnb  SCON.0,RX1
        clr  SCON.0
        mov  A,SBUF
        ret
RX2:    mov p1,#02h
        clr SCON.0
        lcall DELAY
        ret

NUM2HEX: clr C
                subb    A,#30h
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        jc      N2HX
        subb    A,#10
        jnc     N2HX
        add     A,#10
        clr     C
        ret
N2HX:   setb    C
        ret

RX_NUM: lcall RX
        lcall DELAY
        lcall TX2
        lcall NUM2HEX
        jc   RX_NUM
        swap  A
        mov   B,A
RB2:    lcall RX
        lcall DELAY
        lcall TX2
        lcall NUM2HEX
        jc      RB2
        orl   A,B
        mov     B,A
        mov     A,B
        ret

HEX2CHR: push  ACC
        anl   A,#0Fh
        clr   C
        subb  A,#0Ah
        jnc   HC1
        add   A,#3Ah
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        ajmp  HC2
HC1:    add  A,#41h
HC2:    mov  B,A
        pop  ACC
        swap A
        anl  A,#0Fh
        clr  C
        subb A,#0Ah
        jnc  HC3
        add  A,#3Ah
        ajmp  HC4
HC3:    add  A,#41h
HC4:    ret


TX_BYTE: lcall HEX2CHR
        lcall TX2
        mov  A,B
        lcall TX2
        ret

PRINT_SER: pop       DPH
                pop   DPL
PS1:     clr  A
         movc A,@A+DPTR
         cjne A,#00,PS2
         ajmp  PSX
PS2:     lcall TX
         inc  DPTR
         ajmp  PS1
PSX:     push   DPL
                push  DPH
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ret

TX_BASE10: mov      B,#100
           div      AB
           add      A,#30h
           lcall    TX2
           mov      A,B
           mov      B,#10
           div      AB
           add      A,#30h
           lcall    TX2
           mov      A,B
           add      A,#30h
           lcall    TX2
           ret

NEXT_TIME: mov      A,PER
      mov      B,#60
      div      AB
      mov      BUFF1,A
      mov      BUFF2,B
      mov      A,SEC
               lcall    BCD2HEX
      add      A,BUFF2
      clr      C
      subb     A,#60
      jnc      NT1
      add      A,#60
      lcall    HEX2BCD
      mov      SEC,A
      mov      A,MIN
      ajmp     NT2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
NT1:    lcall   HEX2BCD
                mov   SEC,A
                mov   A,MIN
                inc   A
NT2:    add     A,BUFF1
                lcall   BCD2HEX
                clr   C
                subb  A,#60
                jnc   NT3
                add   A,#60
        lcall           HEX2BCD
                mov   MIN,A
                ret
NT3:    lcall   HEX2BCD
        mov MIN,A
                mov   A,HOU
                lcall   BCD2HEX
                inc   A
                cjne  A,#24,NT4
                mov   HOU,#0
                ret
NT4:    lcall   HEX2BCD
                mov   HOU,A
                ret

BCD2HEX: push           ACC
        anl         A,#0F0h
        swap        A
        mov         B,#10
        mul         AB
        mov         B,A
        pop         ACC
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        anl     A,#0Fh
        add     A,B
                ret

HEX2BCD: mov    B,#10
         div    AB
         swap   A
         add    A,B
         ret

DELAY:  mov  50h,R7
        mov  49h,R6
        mov     R7,#04h
DL1:    mov     R6,#0
         djnz   R6,$
         djnz   R7,DL1
         mov R7,50h
        mov R6,49h
         ret

DELAY2: mov  50h,R7
        mov  49h,R6
        mov     R7,#03h
DL2:    mov     R6,#0D9h
         djnz   R6,$
         djnz   R7,DL2
         mov R7,50h
         mov R6,49h
         ret

DELAY3: mov  50h,R7
        mov  49h,R6
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        mov     R7,#0
DL3:    mov     R6,#0
        djnz    R6,$
        djnz    R7,DL3
        mov R7,50h
    mov R6,49h
        ret

WAIT:   clr     A
        cjne    A,PER,WT1
        ret
WT1:    mov     ADDR_RTC,#SECOND
        lcall   READ_RTC
        mov     BUFF1,A
        mov     COUNT,#0
WT2:    lcall   READ_RTC
        cjne    A,BUFF1,WT3
        ajmp    WT2
WT3:    mov     BUFF1,A
        inc     COUNT
        mov     A,COUNT
        cjne    A,PER,WT2
        ret

WRITE_BYTE: mov R3,BUFF4
    cjne R3,#01h,SPEC1
    mov R3,BUFF3
    cjne R3,#02h,TP0
    mov A,#ADDR242
    ajmp SPEC2
TP0:    cjne R3,#03h,TP1
    mov A,#ADDR243
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        ajmp SPEC2
TP1:    cjne R3,#04h,TP2
        mov A,#ADDR244
        ajmp SPEC2
TP2:    cjne R3,#05h,TP3
        mov A,#ADDR245
        ajmp SPEC2
TP3:    cjne R3,#06h,TP4
        mov A,#ADDR246
        ajmp SPEC2
TP4:    cjne R3,#07h,TP5
        mov A,#ADDR247
        ajmp SPEC2
TP5:    cjne R3,#01h,TP6
        mov     A,#ADDR24
TP6:    ajmp SPEC2

SPEC1:  mov R3,BUFF5
        cjne R3,#07h,TPS0
        mov A,#ADDR24
        ajmp SPEC2
TPS0:   cjne R3,#06h,TPS1
        mov A,#ADDR242
        ajmp SPEC2
TPS1:   cjne R3,#05h,TPS2
        mov A,#ADDR243
        ajmp SPEC2
TPS2:   cjne R3,#04h,TPS3
        mov A,#ADDR244
        ajmp SPEC2
TPS3:   cjne R3,#03h,TPS4
        mov A,#ADDR245
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        ajmp SPEC2
TPS4:   cjne R3,#02h,TPS5
        mov A,#ADDR246
        ajmp SPEC2
TPS5:   cjne R3,#01h,TPS6
        mov A,#ADDR247
        ajmp SPEC2
TPS6:   ljmp MAIN

SPEC2:  lcall   START
                jc              X49

                clr     ACC.0
                lcall   SHOUT
                jc              X48

                mov     A, ADDR_HI
                lcall   SHOUT
                jc              X48

                mov     A, ADDR_LO
                lcall   SHOUT
                jc              X48

                mov     A, ZDATA
                lcall   SHOUT
                jc              X48

                clr     C
X48:    lcall   STOP
X49:    ret
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
READ_CURRENT: lcall START
              jc      X45

              setb ACC.0
              lcall SHOUT
              jc      X44

              lcall SHIN
              lcall NAK
              clr  C
X44:      lcall  STOP
X45:      ret


READ_RANDOM: mov R3,BUFF4
      cjne R3,#01h,SPECS1
      mov R3,BUFF3
      cjne R3,#02h,AP0
      mov A,#ADDR242
      ajmp SPECS2
AP0:     cjne R3,#03h,AP1
      mov A,#ADDR243
      ajmp SPECS2
AP1:     cjne R3,#04h,AP2
      mov A,#ADDR244
      ajmp SPECS2
AP2:     cjne R3,#05h,AP3
      mov A,#ADDR245
      ajmp SPECS2
AP3:     cjne R3,#06h,AP4
      mov A,#ADDR246
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        ajmp SPECS2
AP4:    cjne R3,#07h,AP5
        mov A,#ADDR247
        ajmp SPECS2
AP5:    cjne R3,#01h,AP5
        mov     A,#ADDR24
        ajmp SPECS2

SPECS1:  mov R3,BUFF5
        cjne R3,#07h,APS0
        mov A,#ADDR24
        ajmp SPECS2
APS0:   cjne R3,#06h,APS1
        mov A,#ADDR242
        ajmp SPECS2
APS1:   cjne R3,#05h,APS2
        mov A,#ADDR243
        ajmp SPECS2
APS2:   cjne R3,#04h,APS3
        mov A,#ADDR244
        ajmp SPECS2
APS3:   cjne R3,#03h,APS4
        mov A,#ADDR245
        ajmp SPECS2
APS4:   cjne R3,#02h,APS5
        mov A,#ADDR246
        ajmp SPECS2
APS5:   cjne R3,#01h,APS6
        mov A,#ADDR247
        ajmp SPECS2
APS6:   ljmp MAIN
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
P46:    ajmp X46
P47:    ajmp X47
SPECS2:         lcall   START
                jc              P47

                clr     ACC.0
                lcall   SHOUT
                jc              P46

                mov     A,ADDR_HI
                lcall   SHOUT
                jc              X46

                mov     A, ADDR_LO
                lcall   SHOUT
                jc              X46

        mov R3,BUFF4
        cjne R3,#01h,SPOCS1
        mov R3,BUFF3
        cjne R3,#02h,APP0
        mov A,#ADDR242
        ajmp SPOCS2
APP0:       cjne R3,#03h,APP1
        mov A,#ADDR243
        ajmp SPOCS2
APP1:       cjne R3,#04h,APP2
        mov A,#ADDR244
        ajmp SPOCS2
APP2:       cjne R3,#05h,APP3
        mov A,#ADDR245
        ajmp SPOCS2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
APP3:    cjne R3,#06h,APP4
         mov A,#ADDR246
         ajmp SPOCS2
APP4:    cjne R3,#07h,APP5
         mov A,#ADDR247
         ajmp SPOCS2
APP5:    cjne R3,#01h,APP5
         mov      A,#ADDR24
         ajmp SPOCS2

SPOCS1:  mov R3,BUFF5
         cjne R3,#07h,APSS0
         mov A,#ADDR24
         ajmp SPOCS2
APSS0:   cjne R3,#06h,APSS1
         mov A,#ADDR242
         ajmp SPOCS2
APSS1:   cjne R3,#05h,APSS2
         mov A,#ADDR243
         ajmp SPOCS2
APSS2:   cjne R3,#04h,APSS3
         mov A,#ADDR244
         ajmp SPOCS2
APSS3:   cjne R3,#03h,APSS4
         mov A,#ADDR245
         ajmp SPOCS2
APSS4:   cjne R3,#02h,APSS5
         mov A,#ADDR246
         ajmp SPOCS2
APSS5:   cjne R3,#01h,APSS6
         mov A,#ADDR247
         ajmp SPOCS2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
APSS6:    ljmp MAIN


SPOCS2:         lcall    READ_CURRENT
                ajmp     X47
X46:  lcall     STOP
X47:  ret


START: setb     SDA
                setb     SCL

                jnb      SDA, X40
                jnb      SCL, X40

                nop
                clr      SDA
                nop
                nop
                nop
                nop
                nop
                clr      SCL

                clr      C
                ajmp     X41
X40:  setb      C
X41:  ret


STOP:  clr      SDA
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        nop
        nop
        setb    SCL
        nop
        nop
        nop
        nop
        nop
        nop
        setb    SDA
        ret

SHOUT:  push    B
        mov     B,#8
X42:    rlc     A
        mov     SDA,C
        nop
        setb    SCL
        nop
        nop
        nop
        nop
        clr     SCL
        djnz    B,X42

        setb    SDA
        nop
        nop
        setb    SCL
        nop
        nop
        nop
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        nop
        mov     C, SDA
        clr     SCL

        pop     B
        ret

SHIN:   setb    SDA

        push    B
        mov     B, #8
X43:    nop
        nop
        nop
        setb    SCL
        nop
        nop
        mov     C, SDA
        rlc     A
        clr     SCL
        djnz    B, X43

        pop     B
        ret

ACK:    clr     SDA
        nop
        nop
        setb    SCL
        nop
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        nop
        nop
        nop
        clr     SCL
        ret

NAK:    setb    SDA
        nop
        nop
        setb    SCL
        nop
        nop
        nop
        nop
        clr     SCL
        ret

RTC_CLOCK_ENABLE: mov   ADDR_RTC,#0
                  lcall READ_RTC
                  clr   ACC.7
                  mov   ZDATA,A
                  lcall WRITE_RTC
                  ret

;RTC_CLOCK_DISABLE:
;                 mov   ADDR_RTC,#0
;                 call  READ_RTC
;                 setb  ACC.7
;                 mov   ZDATA,A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;               call    WRITE_RTC
;               ret


WRITE_RTC: mov      A,#ADDR13
                lcall   START
                jc          X59
                clr     ACC.0
                lcall   SHOUT
                jc          X58
                mov     A, ADDR_RTC
                lcall   SHOUT
                jc          X58
                mov     A, ZDATA
                lcall   SHOUT
                jc          X58
                clr     C
X58:    lcall   STOP
X59:    ret


READ_RTC:  mov      A,#ADDR13

                lcall   START
                jc          X55

                clr     ACC.0
                lcall   SHOUT
                jc          X54

                mov     A, ADDR_RTC
                lcall   SHOUT
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                jc          X54

                lcall   STOP

                mov     A,#ADDR13
                lcall   READ_CURRENT
                ajmp    X55
X54:    lcall   STOP
X55:    ret

READ_ADC: setb      CS
                clr     CS
                mov     A,#0dfh
                mov     ACC.5,C
                mov     B,#4
X60:    clr     SCL
                rlc     A
                mov     SDA,C
                setb    SCL
                djnz    B,X60
                setb    SDA
                mov     B,#9
X61:    clr     SCL
                setb    SCL
                mov     C,SDA
                rlc     A
                djnz    B,X61
                mov     DATA_HI,A
                clr     A
                mov     B,#4
X62:    clr     SCL
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        setb    SCL
        mov     C,SDA
        rlc     A
        djnz    B,X62
        anl     A,#0Fh
        mov     DATA_LO,A
        setb    CS
        setb    SCL
        setb    SDA
        ret

CONFIG_1620: clr    RST
        setb    RST
        mov     A,#0ch
        mov     B,#8
X64:    clr     SCL
        rrc     A
        mov     SDA,C
        setb    SCL
        djnz    B,X64
        mov     A,#0ah
        mov     B,#8
X65:    clr     SCL
        rrc     A
        mov     SDA,C
        setb    SCL
        djnz    B,X65
        setb    SCL
        setb    SDA
        clr     RST
        ret
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
START_1620: clr     RST
            setb    RST
            mov     A,#0eeh
            mov     B,#8
X66:  clr   SCL
            rrc     A
            mov     SDA,C
            setb    SCL
            djnz    B,X66
            setb    SCL
            setb    SDA
            clr     RST
            ret

READ_1620: clrRST
            setb    RST
            mov     A,#0aah
            mov     B,#8
X68:  clr   SCL
            rrc     A
            mov     SDA,C
            setb    SCL
            djnz    B,X68
            setb    SDA
            mov     B,#8
X69:  clr   SCL
            mov     C,SDA
            rrc     A
            setb    SCL
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
djnz    B,X69
add     A,#20h
setb    SCL
setb    SDA
clr     RST
ret

END
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

คู่มือไอซี (Data sheet)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

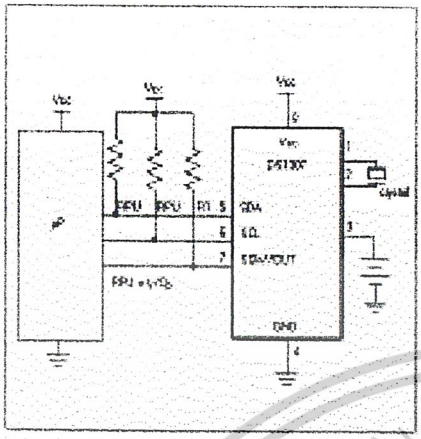
DS1307	8-Pin DIP (300-mil)
DS1307Z	8-Pin SOIC (150-mil)
DS1307N	8-Pin DIP (Industrial)
DS1307ZN	8-Pin SOIC (Industrial)

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 อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

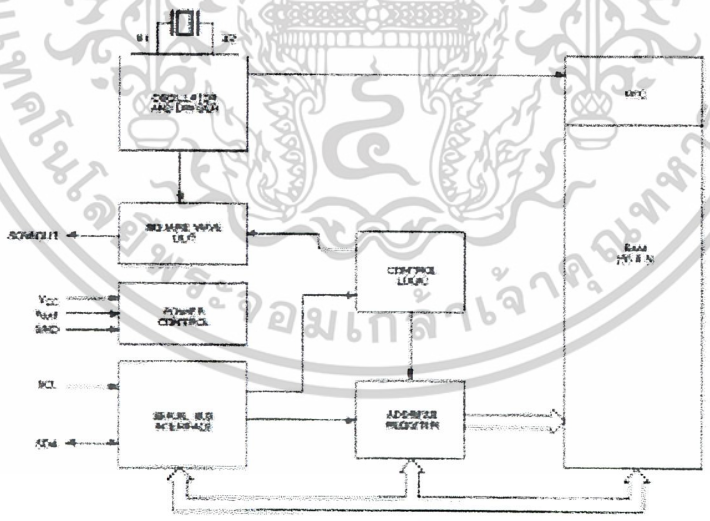
TYPICAL OPERATING CIRCUIT



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 อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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.

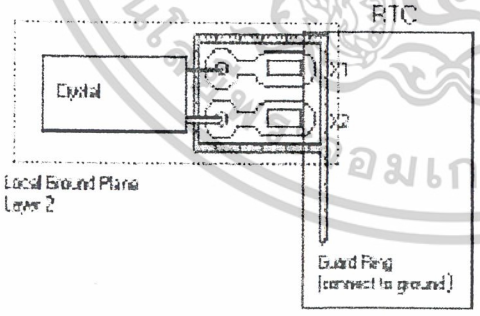
SDA (Serial Data Input/Output) – SDA is the input/output pin for the 2-wire serial interface. The SDA pin is open drain which requires an external pullup resistor.

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 อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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 อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MICROCHIP

24AA256/24LC256/24FC256

256K $I^2C^{™}$ CMOS Serial EEPROM

Features

- Low power CMOS technology
  - Maximum write current 3 mA at 5.5 V
  - Maximum read current 400 µA at 5.5 V
  - Standby current 100 nA typical at 5.5 V
- 2-wire serial interface bus, I²C compatible
- Cascadable for up to eight devices
- Self-timed ERASE/WRITE cycle
- 64-byte page-write mode available
- 5 ms max write-cycle time
- Hardware write protect for entire array
- Output slope control to eliminate ground bounce
- Schmitt trigger inputs for noise suppression
- 1,000,000 erase/write cycles
- Electrostatic discharge protection > 4000 V
- Data retention > 200 years
- 8-pin PDIP, SOIC, TSSOP, MSOP, and DFN packages
- 14-lead TSSOP package
- Temperature ranges:
  - Industrial (I): -40°C to +85°C
  - Automotive (E): -40°C to +125°C

Device Selection Table

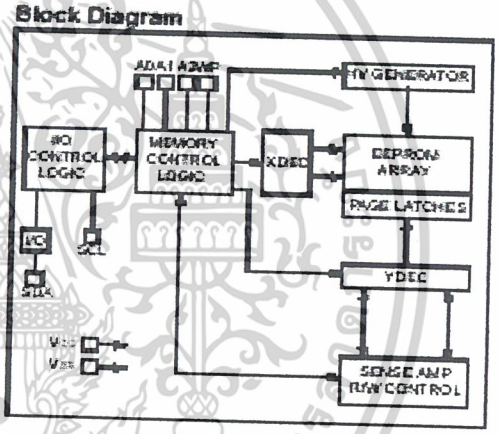
Part Number	Vcc Range	Max. Clock Frequency	Temp. Ranges
24AA256	1.8-5.5 V	400 kHz(1)	I
24LC256	2.5-5.5 V	400 kHz	I, E
24FC256	2.5-5.5 V	1 MHz	I

Note 1: 100 kHz for Vcc < 2.5 V.

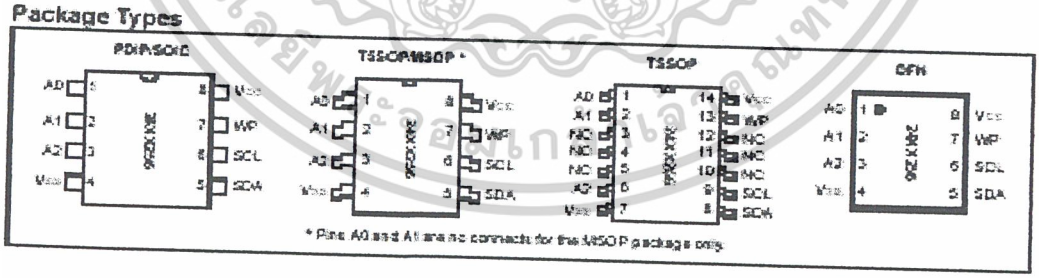
Description

The Microchip Technology Inc. 24AA256/24LC256/24FC256 (24XX256*) is a 32K x 8 (256 Kbit) Serial Electrically Erasable PROM, capable of operation across a broad voltage range (1.8 V to 5.5 V). It has been developed for advanced, low power applications such as personal communications or data acquisition. This device also has a page-write capability of up to 64 bytes of data. This device is capable of both random and sequential reads up to the 256K boundary. Functional address lines allow up to eight devices on the same bus, for up to 2 Mbit address space. This device is available in the standard 8-pin plastic DIP, SOIC, TSSOP, MSOP, DFN and 14-lead TSSOP packages.

Block Diagram



Package Types



* Pins A0 and A1 are no connects for the MSOP package only.

*24XX256 is used in this document as a generic part number for the 24AA256/24LC256/24FC256 devices.

© 2002 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

24AA256/24LC256/24FC256

1.0 ELECTRICAL CHARACTERISTICS

Absolute Maximum Ratings†

Vcc ..... 6.5 V

All inputs and outputs w.r.t. Vss ..... -0.6 V to Vcc +1.0 V

Storage temperature ..... -65°C to +150°C

Ambient temp. with power applied ..... -65°C to +125°C

ESD protection on all pins ..... ≥ 4 kV

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

1.1 24AA256/24LC256/24FC256 DC Electrical Specifications

DC Specifications			Electrical Characteristics: Industrial (I): Vcc = +1.8 V to 5.5 V TAMB = -40°C to +85°C; Automotive (E): Vcc = +2.5 V to 5.5 V TAMB = -40°C to +125°C			
Param. No.	Sym	Characteristic	Min	Max	Units	Conditions
D1	—	A0, A1, A2, SCL, SDA and WP pins	—	—	—	—
D2	VIH	High level input voltage	0.7 Vcc	—	V	—
D3	VIL	Low level input voltage	—	0.3 Vcc	V	Vcc ≥ 2.5 V
				0.2 Vcc	V	Vcc = 2.5 V
D4	VHYS	Hysteresis of Schmitt Trigger inputs (SDA, SCL pins)	0.05 Vcc	—	V	Vcc ≥ 2.5 V **(Note)**
D5	VOL	Low level output voltage	—	0.40	V	IOL = 3.0 mA @ Vcc = 4.5 V IOL = 2.1 mA @ Vcc = 2.5 V
D6	ILI	Input leakage current	—	±10	µA	VIN = Vss or Vcc, WP = Vss VIN = Vss or Vcc, WP = Vcc
D7	ILO	Output leakage current	—	±10	µA	VOUT = Vss or Vcc
D8	CIN, COUT	Pin capacitance (all inputs/outputs)	—	10	pF	Vcc = 5.0 V **(Note)** TAMB = 25°C, fC = 1 MHz
D9	ICC Read	Operating current	—	400	µA	Vcc = 5.5 V, SCL = 400 kHz
	ICC Write		—	3	mA	Vcc = 5.5 V
D10	ICCS	Standby current	—	1	µA	TAMB = -40°C to +85°C SCL = SDA = Vcc = 5.5 V A0, A1, A2, WP = Vss
			—	5	µA	TAMB = -40°C to +125°C SCL = SDA = Vcc = 5.5 V A0, A1, A2, WP = Vss

Note: This parameter is periodically sampled and not 100% tested.

© 2002 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 24AA256/24LC256/24FC256 AC Electrical Specifications

AC Specifications			Electrical Characteristics: Industrial (I): Vcc = +1.8 V to 5.5 V TAMB = -40°C to +85°C Automotive (E): Vcc = +2.5 V to 5.5 V TAMB = -40°C to +125°C			
Param. No.	Sym.	Characteristic	Min.	Max.	Units	Conditions
1	FCLK	Clock frequency	— — —	100 400 1000	kHz	1.8 V ≤ VCC < 2.5 V 2.5 V ≤ VCC ≤ 5.5 V 2.5 V ≤ VCC ≤ 5.5 V 24FC256
2	THIGH	Clock high time	4000 600 500	— — —	ns	1.8 V ≤ VCC < 2.5 V 2.5 V ≤ VCC ≤ 5.5 V 2.5 V ≤ VCC ≤ 5.5 V 24FC256
3	TLOW	Clock low time	4700 1300 500	— — —	ns	1.8 V ≤ VCC < 2.5 V 2.5 V ≤ VCC ≤ 5.5 V 2.5 V ≤ VCC ≤ 5.5 V 24FC256
4	TR	SDA and SCL rise time (Note 1)	— — —	1000 300 300	ns	1.8 V ≤ VCC < 2.5 V 2.5 V ≤ VCC ≤ 5.5 V 2.5 V ≤ VCC ≤ 5.5 V 24FC256
5	TF	SDA and SCL fall time (Note 1)	— —	300 100	ns	All except 24FC256 2.5 V ≤ VCC ≤ 5.5 V 24FC256
6	THD:STA	START condition hold time	4000 600 250	— — —	ns	1.8 V ≤ VCC < 2.5 V 2.5 V ≤ VCC ≤ 5.5 V 2.5 V ≤ VCC ≤ 5.5 V 24FC256
7	TSU:STA	START condition setup time	4700 600 250	— — —	ns	1.8 V ≤ VCC < 2.5 V 2.5 V ≤ VCC ≤ 5.5 V 2.5 V ≤ VCC ≤ 5.5 V 24FC256
8	THD:DAT	Data input hold time	0	—	ns	(Note 2)
9	TSU:DAT	Data input setup time	250 100 100	— — —	ns	1.8 V ≤ VCC < 2.5 V 2.5 V ≤ VCC ≤ 5.5 V 2.5 V ≤ VCC ≤ 5.5 V 24FC256
10	TSU:STO	STOP condition setup time	4000 600 250	— — —	ns	1.8 V ≤ VCC < 2.5 V 2.5 V ≤ VCC ≤ 5.5 V 2.5 V ≤ VCC ≤ 5.5 V 24FC256
11	TSU:WP	WP setup time	4000 600 600	— — —	ns	1.8 V ≤ VCC < 2.5 V 2.5 V ≤ VCC ≤ 5.5 V 2.5 V ≤ VCC ≤ 5.5 V 24FC256
12	THD:WP	WP hold time	4700 1300 1300	— — —	ns	1.8 V ≤ VCC < 2.5 V 2.5 V ≤ VCC ≤ 5.5 V 2.5 V ≤ VCC ≤ 5.5 V 24FC256
13	TAA	Output valid from clock (Note 2)	— — —	3500 900 400	ns	1.8 V ≤ VCC < 2.5 V 2.5 V ≤ VCC ≤ 5.5 V 2.5 V ≤ VCC ≤ 5.5 V 24FC256
14	TBUF	Bus free time: Time the bus must be free before a new transmission can start	4700 1300 500	— — —	ns	1.8 V ≤ VCC < 2.5 V 2.5 V ≤ VCC ≤ 5.5 V 2.5 V ≤ VCC ≤ 5.5 V 24FC256
15	TOF	Output fall time from VIH minimum to VIL maximum Cb ≤ 400 pF	10 + 0.1CB —	250 250	ns	All except 24FC256 (Note 1) 24FC256 (Note 1)
16	TSP	Input filter spike suppression (SDA and SCL pins)	—	50	ns	All except 24FC256 (Notes 1 and 3)
17	TWC	Write cycle time (byte or page)	—	5	ms	—
18	—	Endurance	1,000,000	—	cycles	25°C (Note 4)

Note 1: Not 100% tested. Cb = total capacitance of one bus line in pF.

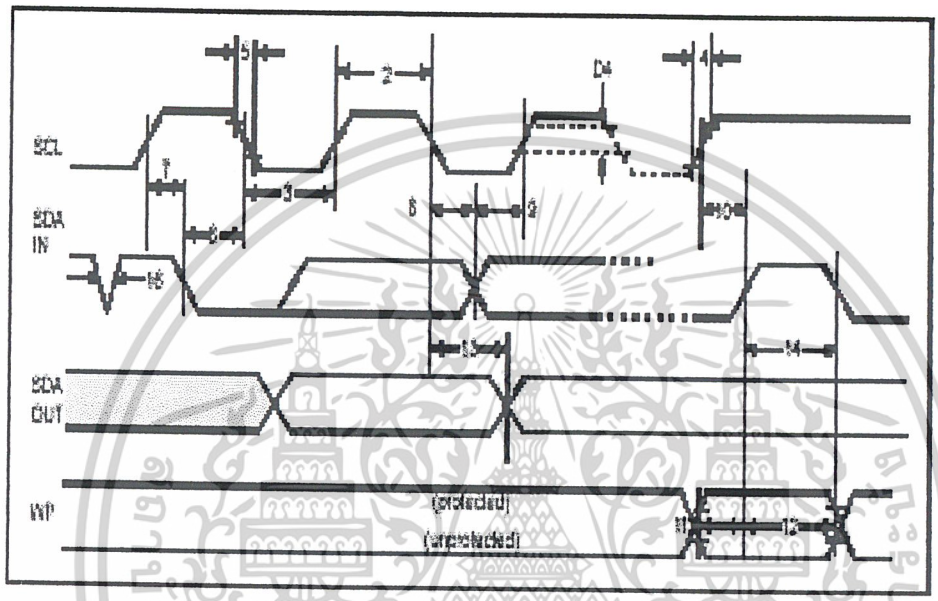
2: As a transmitter, the device must provide an internal minimum delay time to bridge the undefined region (minimum 300 ns) of the falling edge of SCL to avoid unintended generation of START or STOP conditions.

3: The combined TSP and TWC specifications are due to new Schmitt trigger inputs, which provide improved noise spike suppression. This eliminates the need for a TI specification for standard operation.

4: This parameter is not tested but ensured by characterization. For endurance estimates in a specific application, please consult the Total Endurance Model, which can be obtained on Microchip's website: www.microchip.com.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FIGURE 1-1: BUS TIMING DATA



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DALLAS SEMICONDUCTOR

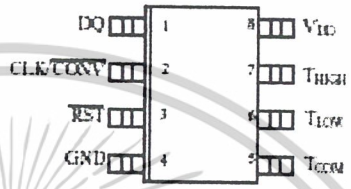
DS1620
Digital Thermometer and Thermostat

www.dalsemi.com

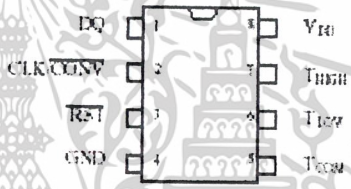
FEATURES

- Requires no external components
- Supply voltage range covers from 2.7V to 5.5V
- Measures temperatures from -55°C to +125°C in 0.5°C increments; Fahrenheit equivalent is -67°F to +257°F in 0.9°F increments
- Temperature is read as a 9-bit value
- Converts temperature to digital word in 1 second (max)
- Thermostatic settings are user-definable and nonvolatile
- Data is read from/written via a 3-wire serial interface (CLK, DQ, $\overline{RST}$)
- Applications include thermostatic controls, industrial systems, consumer products, thermometers, or any thermally sensitive system
- 8-pin DIP or SOIC (208-mil) packages

PIN ASSIGNMENT



DS1620S 8-Pin SOIC (208-mil)
See Mech Drawings Section



DS1620 8-Pin DIP (300-mil)
See Mech Drawings Section

PIN DESCRIPTION

DQ	- 3-Wire Input/Output
CLK/$\overline{CONV}$	- 3-Wire Clock Input and Stand-alone Convert Input
$\overline{RST}$	- 3-Wire Reset Input
GND	- Ground
T_{HIGH}	- High Temperature Trigger
T_{LOW}	- Low Temperature Trigger
T_{COM}	- High/Low Combination Trigger
V_{DD}	- Power Supply Voltage (3V - 5V)

DESCRIPTION

The DS1620 Digital Thermometer and Thermostat provides 9-bit temperature readings which indicate the temperature of the device. With three thermal alarm outputs, the DS1620 can also act as a thermostat. T_{HIGH} is driven high if the DS1620's temperature is greater than or equal to a user-defined temperature TH. T_{LOW} is driven high if the DS1620's temperature is less than or equal to a user-defined temperature TL. T_{COM} is driven high when the temperature exceeds TH and stays high until the temperature falls below that of TL.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

User-defined temperature settings are stored in nonvolatile memory, so parts can be programmed prior to insertion in a system, as well as used in standalone applications without a CPU. Temperature settings and temperature readings are all communicated to/from the DS1620 over a simple 3-wire interface.

OPERATION-MEASURING TEMPERATURE

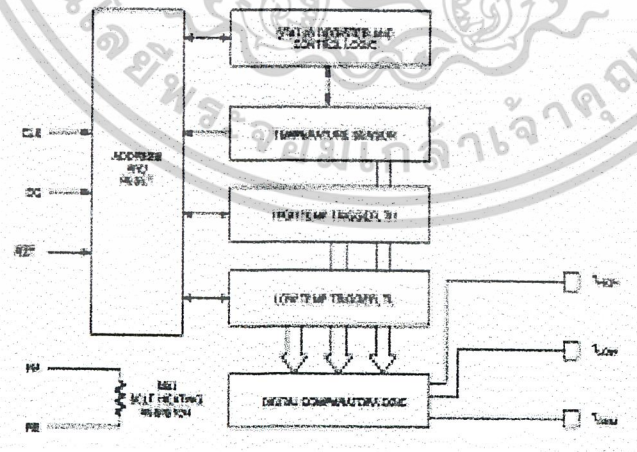
A block diagram of the DS1620 is shown in Figure 1. The DS1620 measures temperatures through the use of an onboard proprietary temperature measurement technique. A block diagram of the temperature measurement circuitry is shown in Figure 2.

The DS1620 measures temperature by counting the number of clock cycles that an oscillator with a low temperature coefficient goes through during a gate period determined by a high temperature coefficient oscillator. The counter is preset with a base count that corresponds to −55°C. If the counter reaches 0 before the gate period is over, the temperature register, which is also preset to the −55°C value, is incremented, indicating that the temperature is higher than −55°C.

At the same time, the counter is then preset with a value determined by the slope accumulator circuitry. This circuitry is needed to compensate for the parabolic behavior of the oscillators over temperature. The counter is then clocked again until it reaches 0. If the gate period is still not finished, then this process repeats.

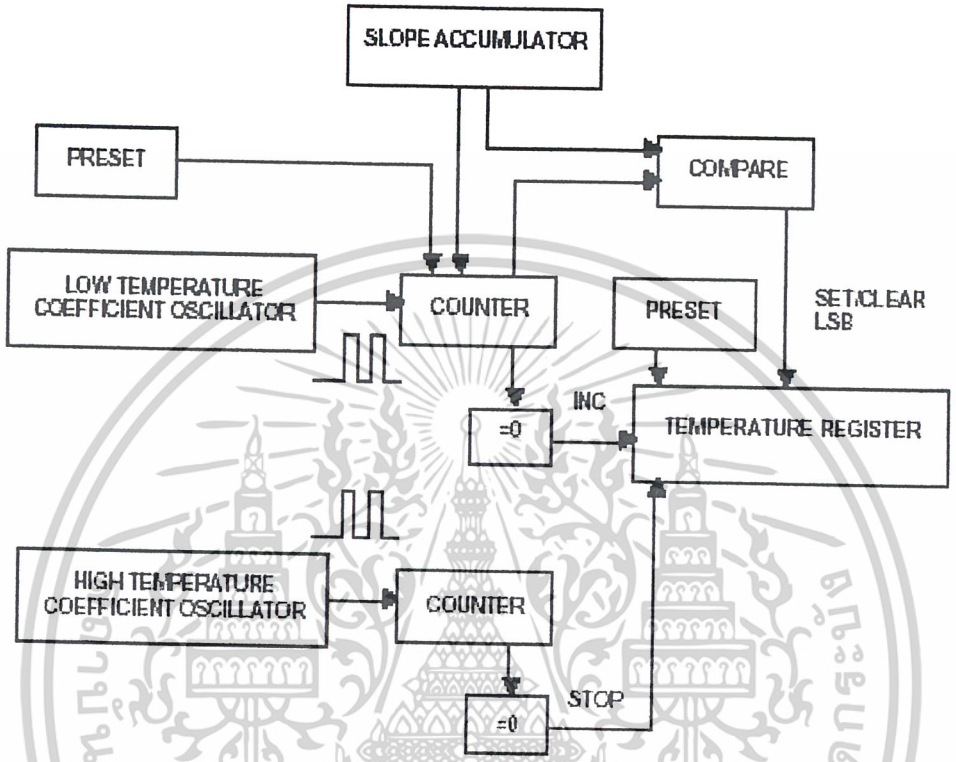
The slope accumulator is used to compensate for the nonlinear behavior of the oscillators over temperature, yielding a high-resolution temperature measurement. This is done by changing the number of counts necessary for the counter to go through for each incremental degree in temperature. To obtain the desired resolution, therefore, both the value of the counter and the number of counts per degree C (the value of the slope accumulator) at a given temperature must be known.

DS1620 FUNCTIONAL BLOCK DIAGRAM Figure 1



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TEMPERATURE MEASURING CIRCUITRY Figure 2



This calculation is done inside the DS1620 to provide 0.5°C resolution. The temperature reading is provided in a 9-bit, two's complement reading by issuing a READ TEMPERATURE command. Table 1 describes the exact relationship of output data to measured temperature. The data is transmitted serially through the 3-wire serial interface, LSB first. The DS1620 can measure temperature over the range of -55°C to +125°C in 0.5°C increments. For Fahrenheit usage, a lookup table or conversion factor must be used.

TEMPERATURE/DATA RELATIONSHIPS Table 1

TEMP	DIGITAL OUTPUT (Binary)	DIGITAL OUTPUT (Hex)
+125°C	0 11111010	00FAh
+25°C	0 00110010	0032h
+½°C	0 00000001	0001h
+0°C	0 00000000	0000h
-½°C	1 11111111	01FFh
-25°C	1 11001110	01CEh
-55°C	1 10010010	0192h

Since data is transmitted over the 3-wire bus LSB first, temperature data can be written to/read from the DS1620 as either a 9-bit word (taking $\overline{RST}$ low after the 9th (MSB) bit), or as two transfers of 8-bit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

words, with the most significant 7 bits being ignored or set to 0, as illustrated in Table 1. After the MSB, the DS1620 will output 0s.

Note that temperature is represented in the DS1620 in terms of a ½°C LSB, yielding the following 9–bit format:

MSB

X	X	X	X	X	X	X	1

LSB

1	1	0	0	1	1	1	0

T = -25°C

Higher resolutions may be obtained by reading the temperature, and truncating the 0.5°C bit (the LSB) from the read value. This value is TEMP_READ. The value left in the counter may then be read by issuing a READ COUNTER command. This value is the count remaining (COUNT_REMAIN) after the gate period has ceased. By loading the value of the slope accumulator into the count register (using the READ SLOPE command), this value may then be read, yielding the number of counts per degree C (COUNT_PER_C) at that temperature. The actual temperature may be then be calculated by the user using the following:

$$\text{TEMPERATURE} = \text{TEMP_READ} - 0.25 + \frac{(\text{COUNT_PER_C} - \text{COUNT_REMAIN})}{\text{COUNT_PER_C}}$$

DETAILED PIN DESCRIPTION Table 2

PIN	SYMBOL	DESCRIPTION
1	DQ	**Data Input/Output pin** for 3-wire communication port.
2	CLK/$\overline{\text{CONV}}$	**Clock input pin** for 3-wire communication port. When the DS1620 is used in a stand-alone application with no 3-wire port, this pin can be used as a convert pin. Temperature conversion will begin on the falling edge of $\overline{\text{CONV}}$.
3	$\overline{\text{RST}}$	**Reset input pin** for 3-wire communication port.
4	GND	**Ground pin.**
5	T_{COM}	**High/Low Combination Trigger.** Goes high when temperature exceeds TH; will reset to low when temperature falls below TL.
6	T_{LOW}	**Low Temperature Trigger.** Goes high when temperature falls below TL.
7	T_{HIGH}	**High Temperature Trigger.** Goes high when temperature exceeds TH.
8	V_{DD}	**Supply Voltage.** 2.7V – 5.5V input power pin.

OPERATION–THERMOSTAT CONTROLS

Three thermally triggered outputs, T_{HIGH}, T_{LOW}, and T_{COM}, are provided to allow the DS1620 to be used as a thermostat, as shown in Figure 3. When the DS1620's temperature meets or exceeds the value stored in the high temperature trip register, the output T_{HIGH} becomes active (high) and remains active until the DS1620's measured temperature becomes less than the stored value in the high temperature register, TH. The T_{HIGH} output can be used to indicate that a high temperature tolerance boundary has been met or exceeded, or it can be used as part of a closed loop system to activate a cooling system and deactivate it when the system temperature returns to tolerance.

The T_{LOW} output functions similarly to the T_{HIGH} output. When the DS1620's measured temperature equals or falls below the value stored in the low temperature register, the T_{LOW} output becomes active. T_{LOW} remains active until the DS1620's temperature becomes greater than the value stored in the low temperature register, TL. The T_{LOW} output can be used to indicate that a low temperature tolerance

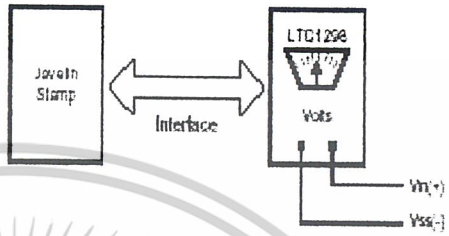
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PARALLAX

599 Menlo Drive, Suite 100
Rocklin, California 95765, USA
Office/Tech Support: (916) 624-8333
Fax: (916) 624-8003

Web Site: www.javelinstamp.com
Home Page: www.parallaxinc.com

General: info@parallaxinc.com
Sales: sales@parallaxinc.com
Technical: javelintech@parallaxinc.com



Contents

Introduction to the LTC1298 12-bit Analog to Digital Converter 1
How the LTC1298 Works 1
Downloads, Parts, and Equipment for the LTC1298 2
LTC1298 Example Circuit 3
Testing the LTC1298 Circuit 4
Program Listing 1.1 – The LTC1298Test 6
Extra Features Built into the LTC1298 7
The LTC1298 Library 7
The AtoD Abstract Library 8
Compare the LTC1298 Voltages 8
Program Listing 1.2 – LTC1298VoltageCompare 9
LTC1298 Demo 10
Published Resources – for More Information 10
Javelin Stamp Discussion Forum – Questions and Answers 11

Introduction to the LTC1298 12-bit Analog to Digital Converter

The LTC1298 is a 2 channel 12-bit analog to digital converter (A/D converter). An A/D converter will read an analog voltage and give you a corresponding digital number. This number can be easily converted into the equivalent voltage measurement.

How the LTC1298 Works

The LTC1298 reads a voltage from 0 to 5 volts and will return a *raw* value which corresponds to an interval from 0 to 4095 (12-bits). The Javelin will take this *raw* value and calculate the voltage level it represents. The

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

highest voltage the chip can read is 4.998 volts. Each increment (bit) is 1.220703125 millivolts in width. There are 4,095 increments from 0 to 5 volts. The LTC1298 measures a voltage, determines which increment it is closest to and gives you this result. Therefore, there is an error of ±1.220703125/2 or 0.6103515625 millivolts.

The LTC1298 has 2 channels, so it can read two separate voltages. If you read each of these voltages and compared them manually there is a possibility of a 1.220703125 millivolt error. It is best when comparing the two channels to allow the LTC1298 to do the comparison itself.

The LTC1298 returns a *raw* value of 0 to 4095. The Javelin will take this *raw* value and calculate the correct voltage measurement.

Downloads, Parts, and Equipment for the LTC1298

This application note (AppNote008-LTC1298.java), the LTC1298 library file (LTC1298.java), the library's javadoc file (LTC1298.pdf), the test program (LTC1298Test.java), the demonstration program (LTC1298Demo.java) which will demonstrate the methods available to you for the LTC1298, and an application example (LTC1298VoltageCompare.java) are all available to you for free download from:

http://www.javelinstamp.com/Applications.htm

You can use the AppNote008-LTC1298.exe, to install the files listed below. These files must be located in specific paths within the Javelin Stamp IDE directory. Although the path to this directory can be different, the default root path is: C:\Program Files\Parallax Inc\Javelin Stamp IDE

The file list below is organized by directory, then by filename; please verify that your file list is organized in the same way.

<root path>\doc\AppNote008-LTC1298.pdf
<root path>\lib\stamp\peripheral\io\ADC\LTC1298.java
<root path>\doc\LTC1298.pdf
<root path>\Projects\examples\peripheral\io\ADC\LTC1298Test.java
<root path>\Projects\examples\peripheral\io\ADC\LTC1298Demo.java
<root path>\Projects\examples\peripheral\io\ADC\LTC1298VoltageCompare.java

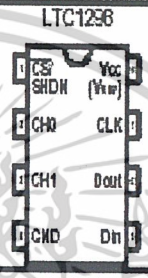
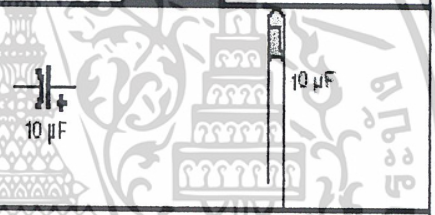
In addition to the files above, the AtoD abstract library class file is also required. The AtoD.java file can be found in AppNote006 and must be installed in the following directory:

<root path>\lib\stamp\peripheral\io\ADC\AtoD.java

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 1.1 lists the parts you will need for this application note.

Table 1.1: Parts List

Quantity	Part Ordering Info and Part Description	Schematic Symbol/ Pin Map
1	Linear Technology's LTC1298 Parallax Part #604-00001	
1	10 µF tantalum Capacitor Parallax Part #202-01060	
1	1 kΩ Resistor Parallax Part #150-01020	
2	100 kΩ Potentiometer Parallax Part #152-01040	

The equipment used to test this example includes a Javelin Stamp, Javelin Stamp Demo Board, 7.5 V, 1000 mA DC power supply, serial cable, and PC with the Javelin Stamp IDE v2.01.

LTC1298 Example Circuit

Figure 1.1 is the circuit that will be used with Program Listing 1.1. Each potentiometer is connected to a separate channel of the LTC1298 chip. By turning a potentiometer you will introduce a voltage (0-5 V) to one of the channels the LTC1298 is measuring. The channels are software selectable; once the channel is selected, the LTC1298 will read the voltage then send a numeric number that represents this voltage value to the Javelin Stamp. The Javelin Stamp will transmit this message to the IDE's *messages from the Javelin window*.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Here's how you connect the LTC1298 to the Javelin Stamp (see Figure 1.1).

- Pin 1, the chip select pin (CS) of the LTC1298 is connected to pin 9 (P4) of the Javelin Stamp.
- Pin 2, the channel 0 pin (CH0) of the LTC1298 is connected to the *wiper* of a potentiometer. The other two leads of the potentiometer are connected to ground (Vss) and +5 V (Vdd).
- Pin 3, the channel 1 pin (CH1) of the LTC1298 is connected to the *wiper* of a potentiometer. The other two leads of the potentiometer are connected to ground (Vss) and +5 V (Vdd).
- Pin 4, the ground pin (GND) of the LTC1298 is connected to ground (Vss).
- Pin 5, the data in pin (Din) is connected to pin 6 (P1) of the Javelin Stamp.
- Pin 6, the data out pin (Dout) of the LTC1298 is connected to a 1 kΩ resistor. The resistor is then connected to pin 5, the data in pin (Din), of the LTC1298.
- Pin 7, the clock pin (CLK) of the LTC1298 is connected to pin 7 (P2) of the Javelin Stamp.
- Pin 8, the power pin (Vcc) of the LTC1298 is connected to +5 V (Vdd). This pin is also connected to the positive lead of a 10 uF tantalum capacitor. The negative lead of this capacitor is connected to ground (Vss).

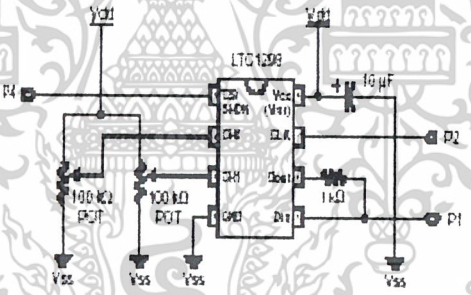


Figure 1.1
Wiring diagram
for the LTC1298

Testing the LTC1298 Circuit

Program Listing 1.1 is a short program that will verify that the circuit in Figure 1.1 is working properly. This program will display the current voltage being read from each potentiometer. When the program is executed it will create an LTC1298 object called `adc`. This object contains the methods from both the LTC1298.java and the AtoD.java libraries.

```
LTC1298 adc = new LTC1298(CPU.pin1,CPU.pin2,CPU.pin4);// Create ADC object
```

Next, the program will clear the Javelin's message window by printing the value `CLS`.

```
System.out.print(CLS);              // Clear the display
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

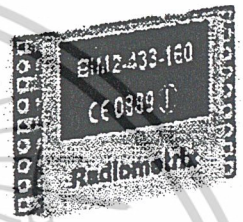
NEW

Radiometrix

Issue 0144-2-1WU | BiM2-433 | 12 February 2002

433MHz high speed FM radio transceiver module

The BiM2 transceiver is an enhanced replacement for our original BiM module. It offers greater transmit power, higher data rates, greatly improved receiver interference rejection and a lower profile. The module is ideal for enabling bi-directional wireless connectivity in battery powered or handheld applications



BiM2 Transceiver

Features

- CE Certified by independent Notified Body according to the R&TTE Directive (1999/5/EC)
- Verified to comply with Radio standard ETSI EN 300 220-3 by UKAS accredited Test Laboratory
- Verified to comply with EMC standard ETSI EN 301 489-3 by UKAS accredited Test Laboratory
- Usable range to 200 metres external, 50 metres in building
- Data rates up to 160kbps
- SAW controlled 10mW FM transmitter
- Double conversion FM superhet receiver
- SAW front end filter and full screening
- Plug in replacement for Radiometrix BiM-433-40
- 3V or 5Volt supply at < 20mA

The BiM2 is a half duplex radio transceiver module for use in high speed bi-directional data transfer applications at ranges up to 200metres. The module operates on the European licence exempt frequency of 433.92MHz. The small footprint of 23 x 33mm and low profile of 4mm together with low power requirements of <20mA @ 3 to 5 Volts enable convenient PCB installation. The high raw data rate capability of 64kbps and fast state change times will support high data throughput of up to 3 kbyte/s in 'streaming' applications or alternatively allows very short air time utilization in multi-node scanning networks.

Applications

- PDA's, organizers and laptops
- Handheld terminals
- EPOS equipment, barcode scanners, belt clip printers
- Data loggers
- Audience response systems
- In Building environmental monitoring and control
- High end security and fire alarms
- Restaurant ordering systems
- Vehicle data up/download

Further information will be available imminently, however, BiM-433-F data sheets contains applications information that is equally applicable to the new BiM2.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AF *pin 13*

This is a buffered and filtered analogue output from the FM demodulator. It has a standing DC bias of 1.2 volts and 400mV p-p base band signal. It is useful as a test point or to drive linear decoders. Load impedance should be >2kΩ and <100pF.

TXD *pin 14*

This DC coupled modulation input will accept either serial digital data (0V to Vcc levels) or High level linear signals. Input impedance is 10kΩ.

TX select *pin 15*

Active low transmit select. 10kΩ internal pull up to Vcc.

RX select *pin 16*

Active low receive select. 10kΩ internal pull up to Vcc.

Pin 15 TX	Pin 16 RX	Function
1	1	power down (<1μA)
1	0	receiver enabled
0	1	transmitter enabled
0	0	self test loop back

Vcc *pin 17*

+ve supply pin. +3.0 to +5.5 volts @ <20mA . The supply must be clean < 20mVp-p ripple. A 2.2μF decoupling capacitor and 10Ω series resistor are used internally to filter the supply.

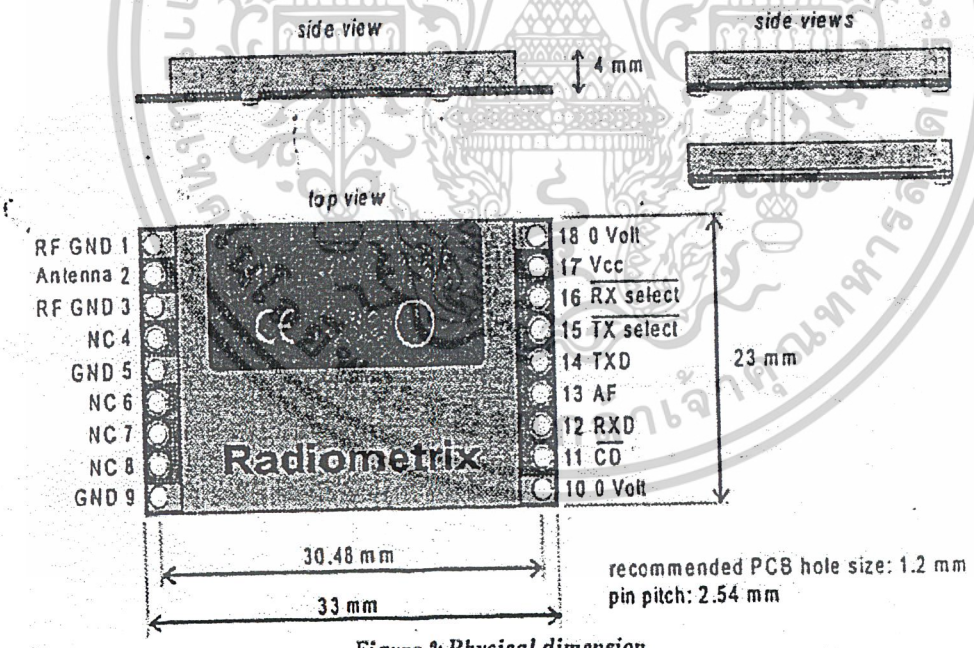


Figure 2: Physical dimension

Weight: 5g (typical ith pins)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Baseband Transfer Performance	pin	min.	typ.	max.	units	notes
Signal applied with supply on						
Signal to valid CI[1], $t_{sig\text{-}CD}$	11	-	0.25	0.5	ms	
Signal to stable data, $t_{sig\text{-}data}$	12	-	3	4	ms	
Signal to stable data, $t_{sig\text{-}data}$	12	-	-	1	ms	1, S version
Signal to stable data, $t_{sig\text{-}data}$	12	-	-	0.5	ms	160kbps
TX power up to full RF	2	-	100	-	µs	

Note 1: from 45% to 55% duty cycle

Antenna requirements

Three types of integral antenna are recommended and approved for use with the module:

A) *Helical* Wire coil, connected directly to pin 2, open circuit at other end. This antenna is very efficient given it's small size (20mm x 4mm dia.). The helical is a high Q antenna, trim the wire length or expand the coil for optimum results. The helical de-tunes badly with proximity to other conductive objects.

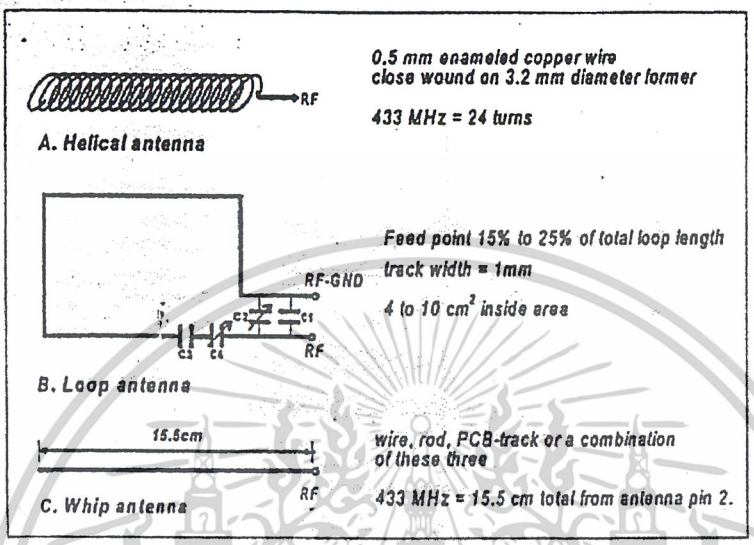
B) *Loop* A loop of PCB track tuned by a fixed or variable capacitor to ground at the 'hot' end and fed from pin 2 at a point 20% from the ground end. Loops have high immunity to proximity de-tuning.

C) *Whip* This is a wire, rod ,PCB track or combination connected directly to pin 2 of the module. Optimum total length is 16cm (1/4 wave @ 433MHz). Keep the open circuit (hot) end well away from metal components to prevent serious de-tuning. Whips are ground plane sensitive and will benefit from internal 1/4 wave earthed radial(s) if the product is small and plastic cased

	A *helical*	B *loop*	C *whip*
Ultimate performance	**	*	***
Easy of design set-up	**	*	***
Size	***	**	*
Immunity proximity effects	**	***	*
Range open ground to similar antenna			200m

The antenna choice and position directly controls the system range. Keep it clear of other metal in the system, particularly the 'hot' end. The best position by far, is sticking out the top of the product. This is often not desirable for practical/ergonomic reasons thus a compromise may need to be reached. If an internal antenna must be used, try to keep it away from other metal components, particularly large ones like transformers, batteries and PCB tracks/earth plane. The space around the antenna is as important as the antenna itself.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



0.5 mm enameled copper wire close wound on 3.2 mm diameter former

433 MHz = 24 turns

Feed point 15% to 25% of total loop length

track width = 1mm

4 to 10 cm² inside area

wire, rod, PCB-track or a combination of these three

433 MHz = 15.5 cm total from antenna pin 2.

Ordering information

The standard BiM2, order code BiM2-433-64 is supplied with pins fitted for operation on 5 volt supplies (4 to 5.5v)

Versions and Variants

BiM2-433-64-3V

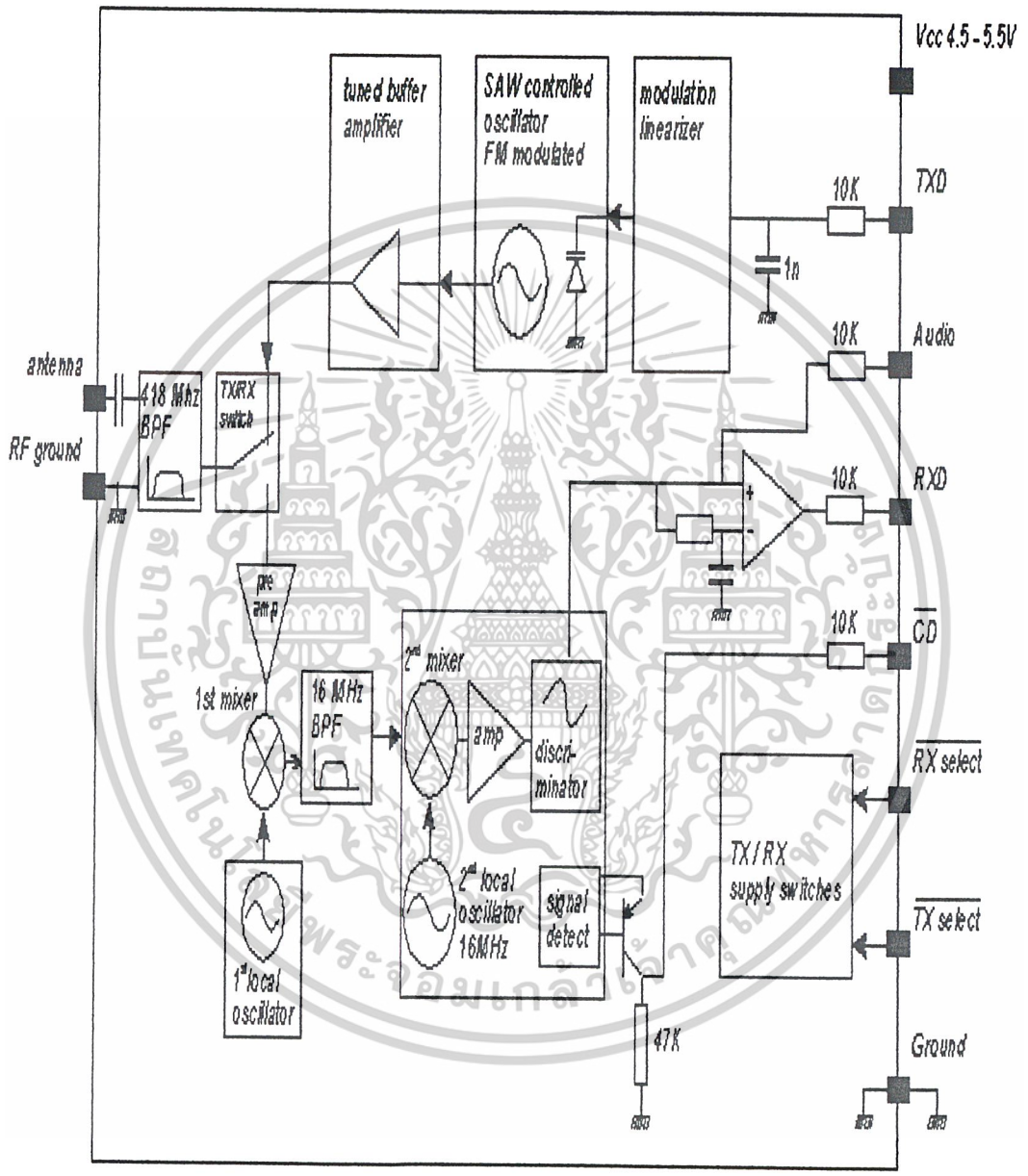
A 3 volt version is available, BiM-433-64-3V for operation at 3.3volts (3.0 to 4.0 V) it is identical to the standard version but has been tested and aligned for operation at 3.3v.

BiM2-433-64-S

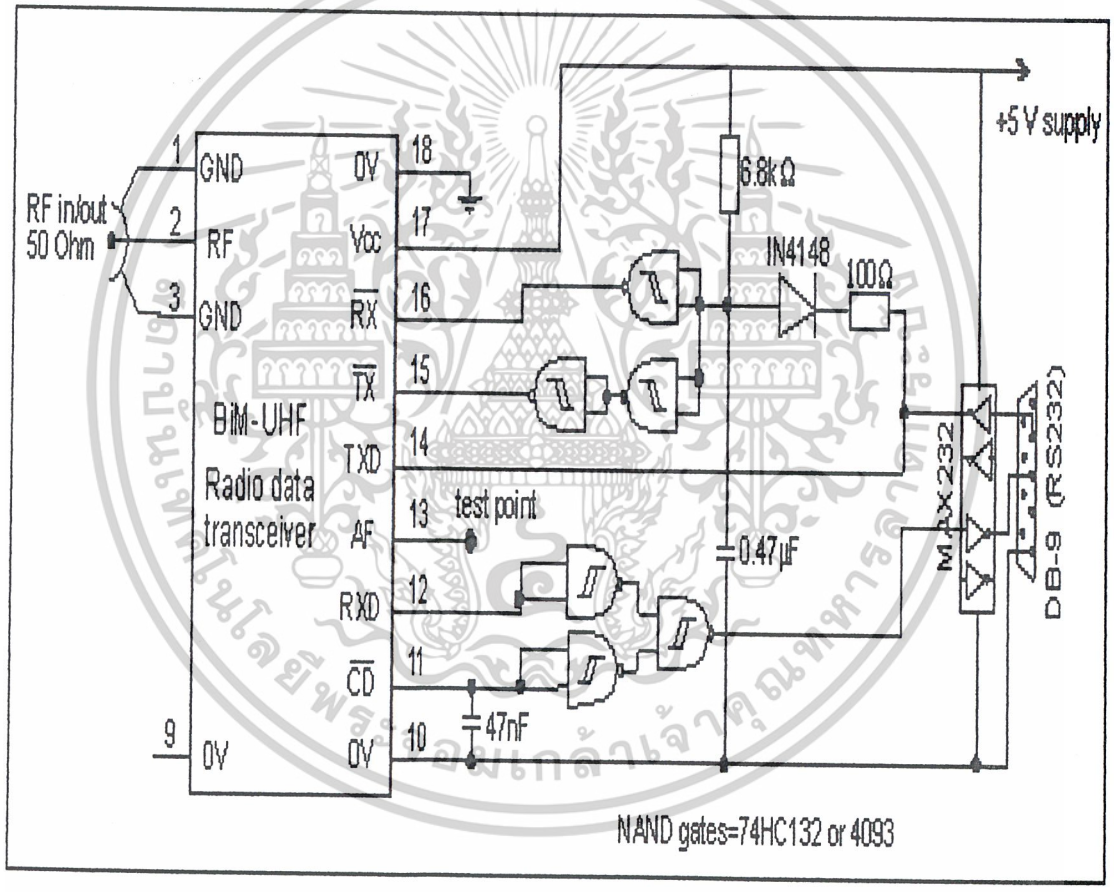
This is intended for RPG or Manchester code only and has fast settling time (maximum 4ms)

Additionally, for volume orders, Radiometrix can supply the BiM2 to the customers' PCB pin requirements or even without any pins.

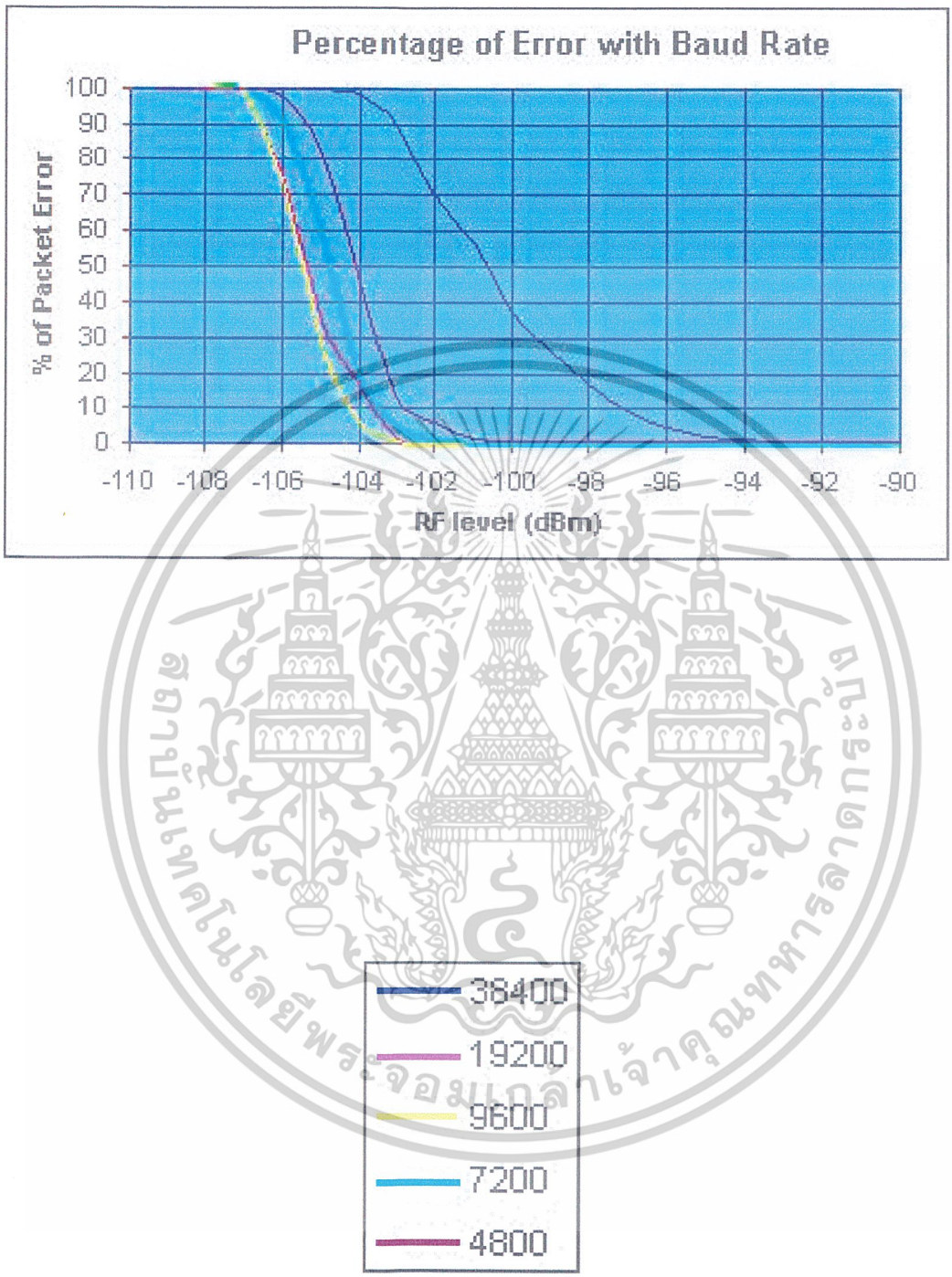
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



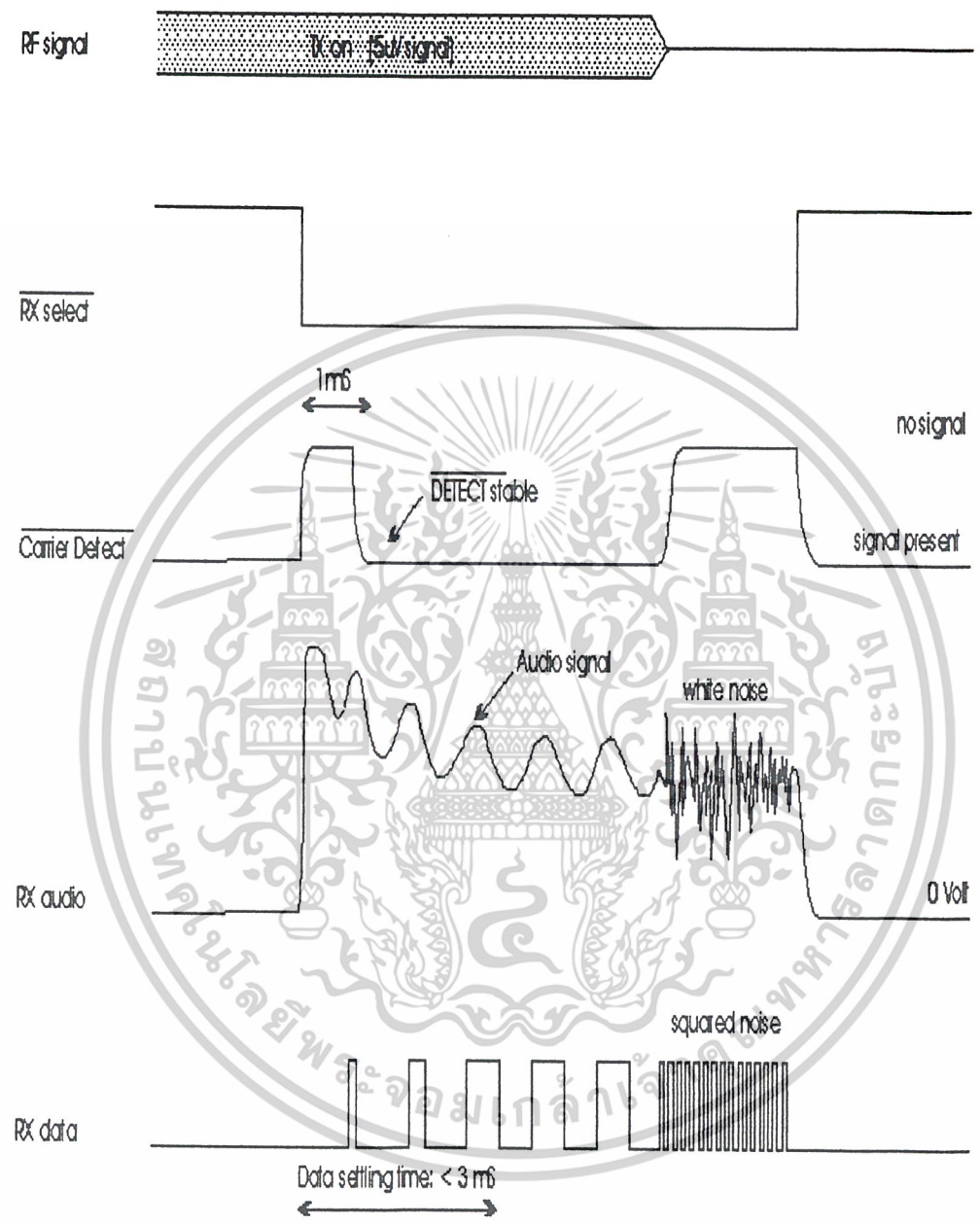
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



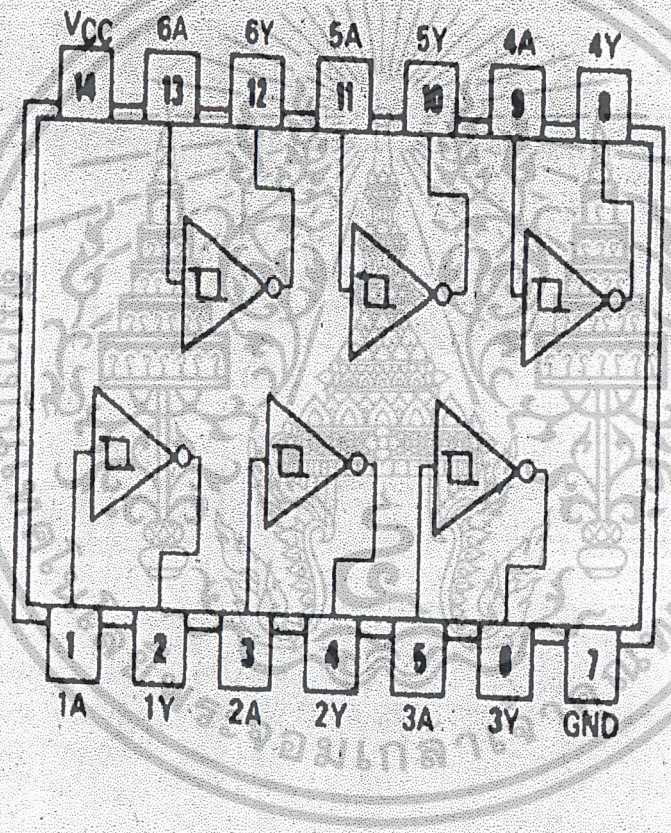
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

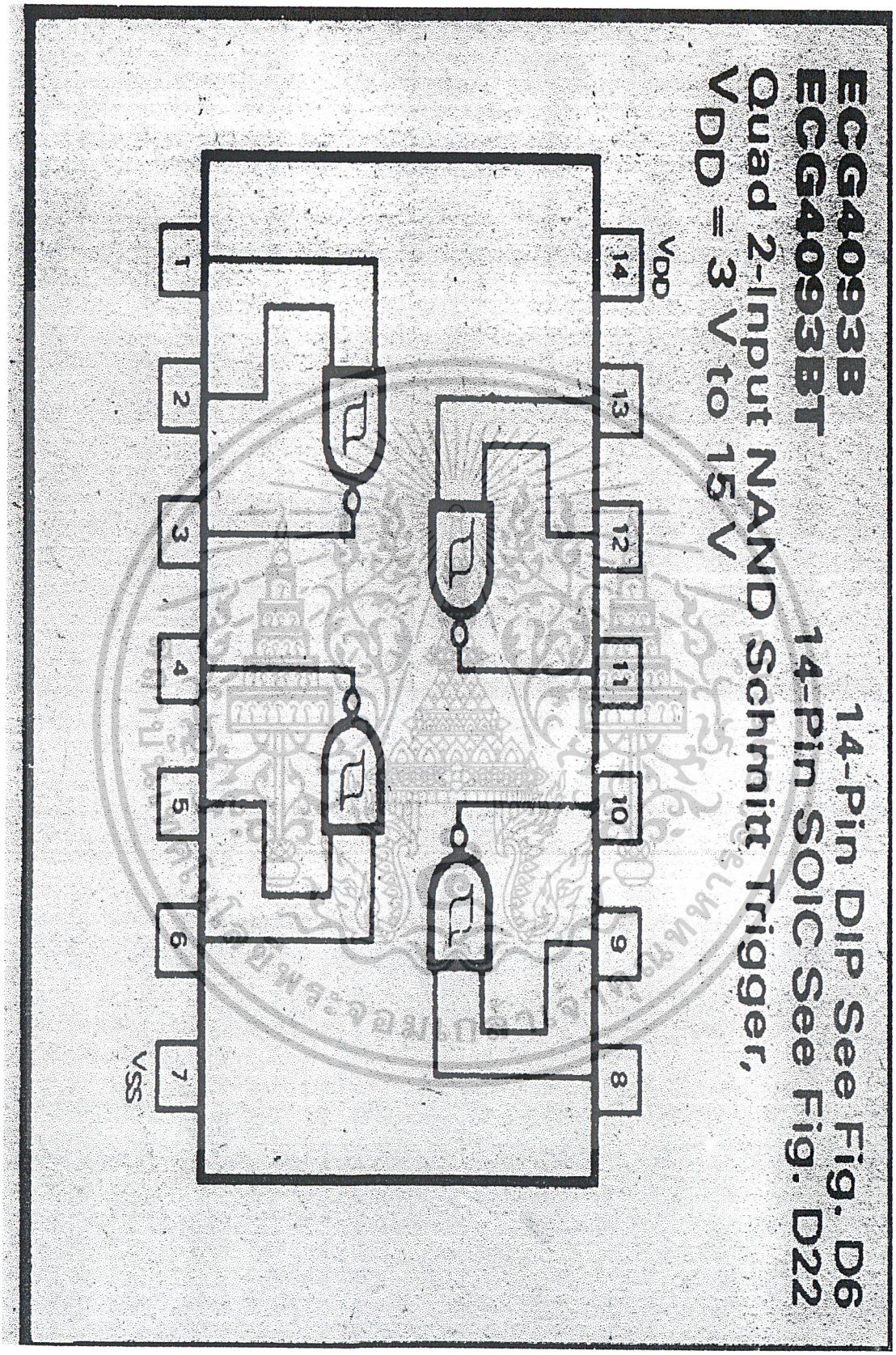
Diag. 16 14-Pin DIP See Fig. D6

ECG7414, ECG74C14, ECG74HC14, ECG74HCT14, ECG74LS14



Hex Schmitt Trigger Inverter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



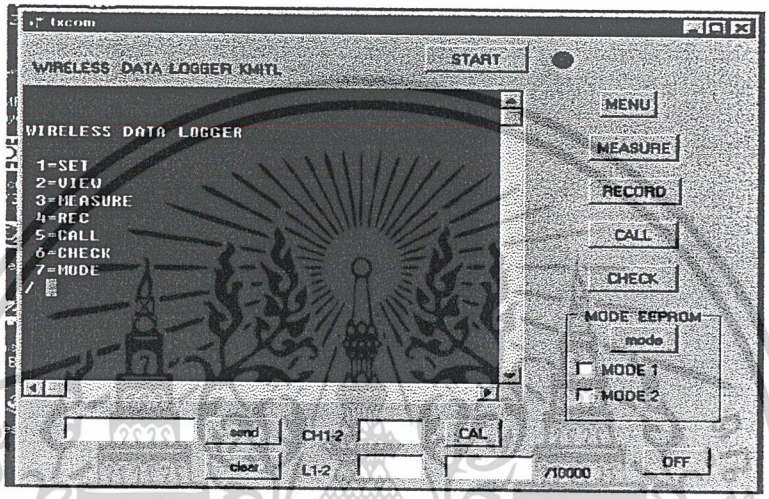
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค
คู่มือการใช้งานเครื่องวัดและบันทึกข้อมูลแบบไร้สาย

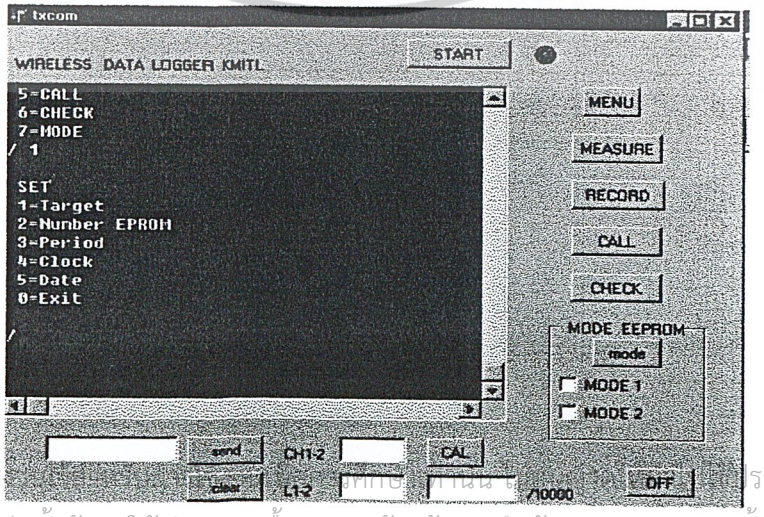
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนการทำงาน

1. ทำการ click file wrpro.exe เพื่อเปิดหน้าต่างการทำงาน
2. ทำการ click ปุ่ม start เพื่อเปิดหน้าต่างการทำงาน (จะมีไฟสีแดงติด)
3. ทำการเปิดเครื่องเพื่อทำการติดต่อแสดงผล
4. ทำการ click ที่หน้าจอสีแดง เพื่อให้เกิด cursor สีฟ้า



5. ทำการ set ค่า กด '1' ตั้งค่าต่างๆ
   - target ตำแหน่งเริ่มต้น eprom (ถ้าใช้มากกว่า 1 ตัว ให้ตั้งที่ '1')
   - num จำนวน eprom ที่จะทำการบันทึก
   - period ตั้งค่าช่วงระยะห่างในการบันทึกแต่ละครั้ง (sec)
   - clock ตั้งเวลา hour (00-24) , min (00-59) และ sec (00-59)
   - date ตั้งวันที่ date (00-31) , month (00-12) และ year (00-99)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. หลังจากนั้นก็ทำงานได้

- MEASURE ทำการวัดแล้วแสดงผล ไม่มีการบันทึก
- RECORD ทำการบันทึกตามค่าset (เลือก mode eprom กด "7" แล้วเลือก mode ก่อนบันทึก 1. เก็บตัวเดียว 2. เก็บมากกว่า 1 ตัว)
- CALL เรียกข้อมูล ใน eprom ตามค่าที่ set
- CHECK เรียกข้อมูลในการบันทึกครั้งสุดท้าย

หมายเหตุ ถ้าจะทำการบันทึกใหม่ต้องทำการ set ใหม่ ถ้าไม่ต้องการให้ทับข้อมูลเดิม

7. ทำการ run บน hyperterminal ได้ โดยใช้การกด หมายเลขบน keybord เป็นการสั่งงาน (baud rate 9600 none parity)

8. send สามารถใช้การใส่หมายเลข แทนการกดปุ่ม

9. clear ทำการลบหน้าจอทั้งหมด

10. ch1-2 ใส่ค่าตัวเลขในช่อง , L1-2 ใส่ค่าตัวเลข กด CAL จะคำนวณค่ามาเป็นค่า volt

โดย ทศนิยม 5 ตำแหน่ง ค่า max 5 volt (ch 1 max 15 volt , ch 2 max 12 volt)

- ch1 นำค่าที่ได้จาก CAL * 3 = max 15 volt
- ch2 นำค่าที่ได้จาก CAL * 2 . 4 = max 12 volt

11. การ print ให้เปิด hyperterminal เลือก transfer , capture text ทำการเลือกfolder ที่ save ตั้งชื่อ แล้ว click start กด call หรือ check ข้อมูลจะถูก save เป็น file .text แล้วนำไป เปิดใน notepad แล้ว print

หมายเหตุ การ setup hyperterminal เปิด setting, control panel, add/remove program windows setup เลือกตามลำดับขั้น click communication, เลือก hyperterminal - กด ok

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงงานเครื่องวัดและบันทึกข้อมูลแบบไร้สายนี้ ได้พบอุปสรรคบ้างเช่นกันแต่สุดท้ายก็สำเร็จไปได้ด้วยดี ทั้งนี้ก็เพราะได้รับคำปรึกษา คำเสนอแนะ และความช่วยเหลือต่าง ๆ จากท่านอาจารย์วรพงศ์ ตั้งศรีรัตน์ ตลอดจนความรู้พื้นฐานต่าง ๆ ที่สำคัญที่นำมาใช้ในการใช้ในการทำโครงงานนี้จากท่านคณาจารย์ทั้งหลายที่ถ่ายทอดผ่านความรู้ในทางวิชาการต่าง ๆ รวมไปถึงเจ้าหน้าที่ทุกท่านที่ช่วยอำนวยความสะดวกทางด้านอุปกรณ์ที่ใช้ในการทำโครงงานนี้ และกำลังใจจากเพื่อน ๆ ทุกคน ทางคณะผู้จัดทำจึงขอขอบพระคุณทุกท่านไว้ ณ ที่นี้ด้วย และจะนำความรู้ที่ได้จากการทำโครงงานชิ้นนี้ไปประยุกต์ใช้งานอื่น ๆ ต่อไป

ธนาณัติ กล้าหาญ

ธีระ ทรัพย์เพลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

1. วรพล ลีลาเกียรติสกุล และคณะ, "ระบบไมโครคอมพิวเตอร์ PC/XT 8088", มหาวิทยาลัยเทคโนโลยีมหานคร, 301 หน้า, 2542
2. บัณฑิต จามรภูติ, "คู่มือการใช้ Protel for Windows", ซีเอ็ด 239 หน้า, 2540
3. ศักดิ์ วาสิกะสิน และ ชนก หงส์น้อย, "ดิจิตอล คอมพิวเตอร์ อิเลคทรอนิกส์", ซีเอ็ด 343 หน้า, 2537
4. มงคล ทองสงคราม, "ดิจิตอลเบื้องต้น", วี.เจ. พริ้นติ้ง 218 หน้า, 2540
5. สักรียา ชิตวงศ์, "วิศวกรรมอิเลคทรอนิกส์", วี.เจ. พริ้นติ้ง 775 หน้า, 2544
6. กิติ ตรีเศษฐ และ คณะ, "วงจรดิจิตอล ภาค", หจก.ภาพพิมพ์, 117 หน้า, 2531
7. ออม จีนประดับ, "ไมโครคอนโทรลเลอร์ MCS-51", สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ, 256 หน้า, 2542

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้