สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ไมโครเทอร์มินอลสำหรับบันทึกข้อมูลการตรวจสอบมิเตอร์น้ำ

นายพรรษิษฐ์ ไตรรัตน์
นายพิเชษฐ จิตวิริยะวศิน

โครงงานพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต
ภาควิชาฟิสิกส์ประยุกต์
คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร ลาดกระบัง
พ.ศ. ๒๕๔๑

เลขหมู่.....
เลขทะเบียน 36694
วัน, เดือน, ปี 23 ส.ค. 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Micro Terminal for Water Meter Testing

Mr. PHANSIT TRIRAT
Mr. PICHET JITVIRIYAVASIN

A Special Project Submitted in Partial Fulfillment of the
Reguirement for the Bechelor Degree of Science
Department of Applied Physics
Faculty of Science
King Mongkut's Institute of Technology Ladkrabang
1998

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงงานพิเศษ	ไมโครเทอร์มินอลสำหรับรับข้อมูลการตรวจสอบมิเตอร์น้ำ
โดย	นายพรรษิษฐ์ ไตรรัตน์
	นายพิเชษฐ จิตวิริยะวศิน
ภาควิชา	ฟิสิกส์ประยุกต์
อาจารย์ที่ปรึกษา	ผศ.วิชิต ศิริโชติ

ภาควิชาฟิสิกส์ประยุกต์ คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
อนุมัติให้นับโครงงานพิเศษฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต

(รศ.สุรพล รักวิจัย) หัวหน้าภาควิชาฟิสิกส์ประยุกต์

คณะกรรมการโครงงานพิเศษ

(ผศ.วิชิต ศิริโชติ) ประธานกรรมการ

(รศ.สุรพล รักวิจัย) กรรมการ

(ผศ.ดร.ปรีชา เทียนสมประสงค์) กรรมการ

ภูมินทร์ จินดาจิธาวัฒน์

(อ.ภูมินทร์ จินดาจิธาวัฒน์) กรรมการ

ลิขสิทธิ์ของภาควิชาฟิสิกส์ประยุกต์ คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร ลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงงานพิเศษ	ไมโครเทอร์มินอลสำหรับรับข้อมูลการตรวจสอบมิเตอร์น้ำ
โดย	นายพรรษิษฐ์ ไตรรัตน์
	นายพิเชษฐ จิตวิริยะวศิน
ภาควิชา	ฟิสิกส์ประยุกต์
อาจารย์ที่ปรึกษา	ผศ.วิชิต ศิริโชติ
ปีการศึกษา	2541

บทคัดย่อ

ไมโครเทอร์มินอลเป็นอุปกรณ์ที่ออกแบบและพัฒนาขึ้นเพื่อบันทึกข้อมูลจากการทดสอบมิเตอร์น้ำ ไมโครเทอร์มินอลประกอบด้วย 2 ส่วนคือ 1) ส่วนรับข้อมูล และ 2) ส่วนแสดงผลผ่านหรือไม่ผ่าน ส่วนรับข้อมูลประกอบด้วยปุ่มรับข้อมูลแบบเมตริกขนาด 4X4 จอแสดงผลแบบผลึกเหลวขนาด 16 อักษร 2 บรรทัด หน่วยความจำแบบ Serial EEPROM 93C66 สำหรับเก็บข้อมูลการทดสอบ ส่วนแสดงผลประกอบด้วย LED แบบ 2 สี คือ สีเขียวสำหรับมิเตอร์ที่ผ่านการทดสอบ และสีแดงสำหรับมิเตอร์ที่ไม่ผ่านการทดสอบ การทดสอบเริ่มโดยการใส่ค่าเริ่มต้นสำหรับมิเตอร์แต่ละตัวที่ต่ออนุกรมกันอยู่ หลังจากที่ปล่อยให้น้ำไหลผ่านปริมาณหนึ่ง เช่น 100 ลิตร จะทำการบันทึกค่าที่มิเตอร์แต่ละตัววัดปริมาณน้ำได้ ส่วนรับข้อมูลจะทำการคำนวณค่าผิดพลาดของตัวมิเตอร์น้ำ และส่งผลไปยังส่วนแสดงผลเพื่อแสดงผลการทดสอบของมิเตอร์แต่ละตัว โดยข้อมูลการทดสอบสามารถส่งไปยังเครื่องคอมพิวเตอร์เพื่อสร้างรายงาน เมื่อเปรียบเทียบผลจากการคำนวณพบว่า อุปกรณ์นี้สามารถแสดงค่าความผิดพลาดได้มากกว่าแบบธรรมดา 1 ตำแหน่งทศนิยม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Special Project Title	Micro Terminal for Water Meter Testing
Name	Mr. Phansit Trirat
	Mr. Pichet Jitviriyavasin
Special Project Advisors	Asst. Prof. Wichit Sirichote
Department	Applied Physics
Academic	1998

Abstract

A device used for recording digital data for water meter testing has been designed and developed. The device namely, Micro Terminal, consists of two units: 1) a data entry unit and 2) pass/failed displaying board. The data entry unit has a 4x4 input keypad, a 16x2 line LCD display, a nonvolatile serial EEPROM, 93C66 for saving tested data. The pass/failed displaying board has a 12x2 bicolor LED with green for pass and red if that meter has failed. Testing was performed by entering initial count for each series connected water meter. After supplying an amount of water, say 100 litters, the count of each meter was then be entered. The program in data entry unit will calculate an error, send the result to pass/failed displaying board. In addition, the tested data can be sent to PC for making a hard copy and officially report also. Comparing to manual calculation, our device has shown 1-digit higher accuracy.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิติกรรมประกาศ

โครงงานพิเศษนี้สำเร็จลุล่วงได้ด้วยดี เนื่องด้วยความอนุเคราะห์จากบุคคลดังต่อไปนี้

ผู้ช่วยศาสตราจารย์วิชิต ศิริโชติ ผู้ที่ให้ความรู้ และแนวทางแก้ปัญหาในทุกๆอย่าง

คุณไพศาล โกมลโรจนาภรณ์ ผู้จัดการทั่วไปบริษัทพีบีไปท์ ที่ให้โอกาสพัฒนางานชิ้นนี้

พี่ๆที่แผนกฟิสิกส์ และวิศวกรรม สำนักงานพัฒนาเทคโนโลยี บริษัทพรีเมียร์โกลเบิลคอร์ปอเรชั่น จำกัด

สุดท้ายนี้ คณะผู้จัดทำโครงงานขอขอบคุณบุคคลทุกๆท่านที่ให้ความช่วยเหลือที่มิได้กล่าวถึงมา ณ ที่นี้ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อปัญหาพิเศษภาษาไทย	ก
บทคัดย่อปัญหาพิเศษภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญตาราง	ฉ
สารบัญภาพ	ช
บทที่ 1 บทนำ	1
1.1 ที่มาของปัญหา	1
1.2 วัตถุประสงค์ในการทำโครงงานพิเศษ	1
1.3 ขอบเขตการดำเนินงาน	2
1.4 ประโยชน์ที่ได้รับ	2
บทที่ 2 ทฤษฎี	3
2.1 ไมโครคอนโทรลเลอร์ MCS-51	3
- หน้าที่แต่ละขาของไมโครคอนโทรลเลอร์	5
2.2 หลักการของ Serial E^2PROM	7
- การใช้งาน Serial E^2PROM	8
- การใช้งานในแต่ละคำสั่ง	9
2.3 การใช้งาน LCD โมดูล	10
- ชุดคำสั่งควบคุมการแสดงข้อความ	11
- แนวทางการเขียนโปรแกรมควบคุม	12
2.4 การเชื่อมต่อแบบมาตรฐาน RS-232	16
บทที่ 3 ขั้นตอนการดำเนินการ	19
3.1 หลักการของการตรวจสอบมิเตอร์น้ำ	19
3.2 ส่วนรับข้อมูล	22
3.3 ส่วนแสดงผล	26
3.3.1 ส่วนถอดรหัสข้อมูล	30
3.3.2 ส่วนแลทช์ข้อมูล และบอร์ดแสดงผล	32
3.3.3 ส่วนจ่ายไฟ	32

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 Current loop Converter 33
3.5 ส่วนสร้างรายงาน 33
บทที่ 4 ผลการทดลอง 37
4.1 ทดสอบการส่งข้อมูลแบบ SPI 37
4.2 ทดสอบการส่งข้อมูลแบบ Current loop 38
4.3 ทดสอบเพื่อเปรียบเทียบความถูกต้องในระบบเก่า และระบบใหม่ 39
บทที่ 5 สรุปผล และแนวทางการพัฒนา 58
5.1 สรุปผล 58
5.2 แนวทางการพัฒนา 58
ภาคผนวก ก. โปรแกรมควบคุมส่วนรับข้อมูล
ภาคผนวก ข. โปรแกรมควบคุมส่วนแสดงผล
ภาคผนวก ค. รายละเอียดของอุปกรณ์ต่างๆ
เอกสารอ้างอิง
ประวัติผู้จัดทำโครงงานพิเศษ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้า
ตาราง 2.2.1 รูปแบบของข้อมูลสำหรับควบคุมการทำงาน EEPROM	8
ตาราง 2.4.1 ฟังก์ชันการใช้งานขาของ DB25 และ DB9	17
ตาราง 3.1.1 แสดงพารามิเตอร์ของมิเตอร์ในกลุ่มที่หนึ่ง	20
ตาราง 3.1.2 แสดงพารามิเตอร์ของมิเตอร์ในกลุ่มที่สอง	20
ตาราง 3.1.3 แสดงพารามิเตอร์ของมิเตอร์ในกลุ่มที่สาม	20
ตาราง 3.1.4 แสดงจำนวนมิเตอร์สูงสุดของมิเตอร์ในแต่ละขนาด	21
ตาราง 3.3.1 ลักษณะของชุดข้อมูล 8 บิทแรกที่ส่งทาง SPI	30
ตาราง 4.3.1 ผลการเปรียบเทียบระบบเก่า และระบบที่ได้รับการพัฒนา	39
- ผลการทดสอบในระบบใหม่	
ตาราง 4.3.2 รายงานผลการทดสอบมิเตอร์รุ่น PXS ขนาด 15 มิล.	40
ตาราง 4.3.3 รายงานผลการทดสอบมิเตอร์รุ่น PXS ขนาด 20 มิล.	41
ตาราง 4.3.4 รายงานผลการทดสอบมิเตอร์รุ่น PXS ขนาด 25 มิล.	42
ตาราง 4.3.5 รายงานผลการทดสอบมิเตอร์รุ่น TDA(MWA) ขนาด 15 มิล.	43
ตาราง 4.3.6 รายงานผลการทดสอบมิเตอร์รุ่น TDA(MWA) ขนาด 20 มิล.	44
ตาราง 4.3.7 รายงานผลการทดสอบมิเตอร์รุ่น TDA(MWA) ขนาด 25 มิล.	45
ตาราง 4.3.8 รายงานผลการทดสอบมิเตอร์รุ่น PXA(UHM) ขนาด 15 มิล.	46
ตาราง 4.3.9 รายงานผลการทดสอบมิเตอร์รุ่น PXA(UHM) ขนาด 20 มิล.	47
ตาราง 4.3.10 รายงานผลการทดสอบมิเตอร์รุ่น PXA(UHM) ขนาด 25 มิล.	48
- ผลการทดสอบในระบบเก่า	
ตาราง 4.3.11 รายงานผลการทดสอบมิเตอร์รุ่น PXS ขนาด 15 มิล.	49
ตาราง 4.3.12 รายงานผลการทดสอบมิเตอร์รุ่น PXS ขนาด 20 มิล.	50
ตาราง 4.3.13 รายงานผลการทดสอบมิเตอร์รุ่น PXS ขนาด 25 มิล.	51
ตาราง 4.3.14 รายงานผลการทดสอบมิเตอร์รุ่น TDA(MWA) ขนาด 15 มิล.	52
ตาราง 4.3.15 รายงานผลการทดสอบมิเตอร์รุ่น TDA(MWA) ขนาด 20 มิล.	53
ตาราง 4.3.16 รายงานผลการทดสอบมิเตอร์รุ่น TDA(MWA) ขนาด 25 มิล.	54
ตาราง 4.3.17 รายงานผลการทดสอบมิเตอร์รุ่น PXA(UHM) ขนาด 15 มิล.	55
ตาราง 4.3.18 รายงานผลการทดสอบมิเตอร์รุ่น PXA(UHM) ขนาด 20 มิล.	56
ตาราง 4.3.19 รายงานผลการทดสอบมิเตอร์รุ่น PXA(UHM) ขนาด 25 มิล.	57

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

	หน้า
ภาพที่ 2.1.1 ลักษณะของไมโครคอนโทรลเลอร์ 89c51	5
ภาพที่ 2.2.1 แสดงบล็อกไดอะแกรม และลักษณะของชิพ EEPROM	7
ภาพที่ 3.1.1 แสดงมิเตอร์ที่ทำการทดสอบขนาดต่างๆ	19
ภาพที่ 3.2.1 ลักษณะของส่วนรับข้อมูล	22
ภาพที่ 3.2.2 แสดงองค์ประกอบหลักของวงจรในส่วนรับข้อมูล	23
ภาพที่ 3.2.3 แสดงโฟลชาร์ทของโปรแกรมตัวรับข้อมูล	24
ภาพที่ 3.3.1 ลักษณะของส่วนแสดงผล	26
ภาพที่ 3.3.2 แสดงขั้นตอนการทำงานในส่วนแสดงผล	27
ภาพที่ 3.3.3 แสดงขั้นตอนการรับข้อมูลแบบ SPI	28
ภาพที่ 3.3.4 แสดงการส่งข้อมูลแบบ SPI	29
ภาพที่ 3.3.5 แสดงลักษณะวงจรแลชท์ และแสดงบอร์ดแสดงผล	31
ภาพที่ 3.3.6 วงจรจ่ายไฟแบบมีทรานซิสเตอร์ช่วยจ่ายกระแส	32
ภาพที่ 3.4.1 แสดงวงจร Current loop	33
ภาพที่ 3.5.1	34
ภาพที่ 3.5.2	34
ภาพที่ 3.5.3	34
ภาพที่ 3.5.4	35
ภาพที่ 3.5.5	35
ภาพที่ 3.5.6	36
ภาพที่ 3.5.7	36
ภาพที่ 4.2.1 แสดงการทดสอบวงจร Current loop	38
ภาพที่ 4.2.2 ผลการทดสอบที่วัดได้จาก Oscilloscope	38

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ที่มาของปัญหา

ในปัจจุบันชุมชนได้มีการขยายตัวมากขึ้น ทั้งในเมืองหลวงและชนบท ฉะนั้นความต้องการทางสาธารณูปโภคขั้นพื้นฐานเช่น น้ำ จึงมีความต้องการมากขึ้น โดยจะสังเกตได้จากความต้องการใช้น้ำที่นับวันจะเพิ่มมากขึ้นจนไม่สามารถผลิตได้ทัน แทบทุกครัวเรือนไม่ว่าจะเป็นในเมืองหลวง ชนบท หรือทุกพื้นที่ ที่การประปาสามารถให้บริการได้ถึงนั้น จำเป็นต้องติดตั้งเครื่องมือชนิดหนึ่งซึ่งจะทำหน้าที่ในการตรวจวัดปริมาณน้ำที่แต่ละครัวเรือนได้บริโภคไปในแต่ละเดือน เครื่องมือดังกล่าวก็คือมิเตอร์น้ำซึ่งจะติดตั้งไว้เพื่อบันทึก ปริมาณน้ำที่แต่ละครัวเรือนได้บริโภคไปในแต่ละเดือนนั้นจะถูกคำนวณเป็นค่าใช้จ่าย ซึ่งประชาชนจะต้องจ่ายค่าบริการดังกล่าวตามสัดส่วนที่ตนได้บริโภคไป ดังนั้นการตรวจวัดคุณภาพของมิเตอร์น้ำในขั้นตอนการผลิตจึงมีความสำคัญมากเพื่อให้มิเตอร์น้ำที่ออกสู่ท้องตลาดมีความถูกต้อง แม่นยำที่สุด

กระบวนการตรวจสอบคุณภาพของมิเตอร์น้ำในปัจจุบัน จะตรวจสอบโดยอาศัยการวัดปริมาณน้ำที่อ่านได้จากตัวมิเตอร์ เปรียบเทียบกับปริมาณน้ำหนึ่งหน่วยปริมาตร ซึ่งจะมีอุปกรณ์มาตรฐานในการตรวจเทียบ ความผิดพลาดที่เกิดขึ้นจะถูกคำนวณให้อยู่ในรูปของเปอร์เซ็นต์ความคลาดเคลื่อน ซึ่งจะใช้เป็นข้อมูลในการแก้ไขความผิดพลาดดังกล่าว แต่ถึงกระนั้นในกระบวนการตรวจวัดคุณภาพก็ยังมีความบกพร่องอยู่ เช่นการบันทึกข้อมูลด้วยการจดบันทึก การคำนวณด้วยมือ ซึ่งวิธีการเหล่านี้ถูกใช้มาตั้งแต่อดีตจนถึงปัจจุบัน โครงงานนี้ได้เล็งเห็นความสำคัญของการตรวจสอบคุณภาพของมิเตอร์น้ำ ซึ่งถูกผลิตจากโรงงานผลิต จึงได้ทำการพัฒนาระบบการบันทึกและคำนวณข้อมูลด้วยคอมพิวเตอร์ ซึ่งจะให้ความน่าเชื่อถือสูงกว่าและยังทำให้สามารถทำงานได้รวดเร็วขึ้นอีกด้วย

1.2 วัตถุประสงค์ของโครงงาน

- เพื่อศึกษาและพัฒนาระบบการตรวจวัดคุณภาพมิเตอร์น้ำให้มีความน่าเชื่อถือมากขึ้น
- เพื่อศึกษาการเชื่อมต่อระบบไมโครคอนโทรเลอร์สองระบบเข้าด้วยกัน
- เพื่อศึกษาการใช้งานหน่วยความจำแบบอนุกรม
- เพื่อเรียนรู้หลักในการออกแบบระบบให้สามารถใช้งานได้ง่าย และใช้งานได้จริง
- เพื่อเรียนรู้ลักษณะการทำงานที่เป็นระบบ และรู้จักวิเคราะห์ปัญหา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตการดำเนินงาน

ในส่วนแรกเป็นการศึกษาระบบการตรวจวัดคุณภาพมิเตอร์น้ำในปัจจุบัน และรวบรวมข้อมูลทั้งความต้องการของระบบ และความถูกต้องของระบบ จากนั้นจึงทำการวิเคราะห์ปัญหาที่จะนำไปสู่ความผิดพลาดที่จะเกิดขึ้น แล้วจึงสรุปและแก้ไขข้อบกพร่องดังกล่าว

ในส่วนที่สองจะเป็นการออกแบบระบบซึ่งได้รับการแก้ไขแล้ว ซึ่งจากการวิเคราะห์ปัญหาทั้งหมดแล้วนั้นพอสรุปได้ว่า การตรวจวัดโดยอาศัยการเปรียบเทียบเปอร์เซ็นต์ความคลาดเคลื่อนเป็นวิธีการตรวจสอบคุณภาพที่ให้ผลที่น่าพอใจอยู่แล้ว แต่การจดบันทึก และการคำนวณด้วยมือนั้นเป็นจุดบกพร่องซึ่งควรจะได้รับการแก้ไข

ส่วนถัดมาเป็นการออกแบบอุปกรณ์ที่จะทำหน้าที่ในการจดบันทึก คำนวณผล และแสดงผล ซึ่งจะอยู่ในรูปของอุปกรณ์ทางดิจิตอลซึ่งน่าจะให้ความน่าเชื่อถือได้สูงกว่า

ส่วนสุดท้ายเป็นการประกอบวงจร และทดสอบการใช้งานจริง และสรุปผลความพึงพอใจในระบบการบันทึก และคำนวณแบบใหม่

1.4 ประโยชน์ที่คาดว่าจะได้รับ

- ช่วยเพิ่มความน่าเชื่อถือของระบบตรวจสอบมิเตอร์น้ำ
- ลดเวลาในที่ใช้ในขั้นตอนการทดสอบมิเตอร์น้ำลง
- รู้จักการประยุกต์ใช้งานระบบไมโครคอนโทรเลอร์ เพื่อให้เกิดประโยนช์สูงสุด
- รู้จักการค้นคว้า และวิเคราะห์ข้อมูล
- รู้จักวิธีการทำงานอย่างมีระบบ และสามารถแก้ไขปัญหาอย่างมีหลักการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่2

ทฤษฎี

2.1 ไมโครคอนโทรลเลอร์ MCS-51

ไมโครคอนโทรลเลอร์ตระกูล MCS51 เป็นไมโครคอนโทรลเลอร์แบบชิพเดี่ยว (Single Chip Microcontroller) คือไมโครโปรเซสเซอร์ที่รวมเอาหน่วยความจำ ส่วนควบคุม และส่วนอินพุต/เอาท์พุท อยู่ในชิพตัวเดียวกัน ลักษณะทั่วไปของ MCS-51 ประกอบด้วย

- ใช้เทคโนโลยี HMOS และ CMOS ในการสร้าง และทำงานด้วยแหล่งจ่ายไฟ 5 V เพียงแหล่งเดียว
- มีหน่วยความจำสำหรับเก็บโปรแกรมควบคุมการทำงานอยู่ภายในชิพจำนวน 4 กิโลไบต์ (เบอร์ 8031,8032 จะไม่มีหน่วยความจำในส่วนนี้ ส่วนเบอร์8052 มีหน่วยความจำส่วนนี้ 8 กิโลไบต์)
- สามารถใช้หน่วยความจำโปรแกรมและหน่วยความจำข้อมูลที่อยู่ภายนอกชิพได้อย่างละ 64 กิโลไบต์แยกออกจากกัน
- คำสั่งส่วนใหญ่ใช้เวลาทำงานเพียง 1 ไมโครวินาที เมื่อใช้คริสตอลความถี่ 12 เมกะเฮิรตซ์
- มีพอร์ตที่สามารถรับส่งข้อมูลได้ทั้ง 2 ทิศทางจำนวน 4 พอร์ตๆละ 8 บิต หรือสามารถใช้งานแยกออกจากกัน ทำให้สามารถมีช่องใช้งานขนาด 1 บิตรวมทั้งสิ้น 32 ช่อง
- พอร์ตอนุกรมสามารถทำงานแบบ Full Duplex ได้ที่ความเร็วสูง
- มีอินเทอร์รัพ 5 แหล่ง และมี 6 แหล่งสำหรับ 8032/8052 พร้อมกับมีการจัดระดับความสำคัญได้ 2 ระดับ
- มีรีจิสเตอร์สำหรับใช้งานเป็น counter หรือ timer เพื่อนับจำนวนสัญญาณนาฬิกาภายในชิพ หรือนับการเปลี่ยนสถานะของสัญญาณภายนอกขนาด 16บิตจำนวน 2 ตัวเพื่อใช้นับจำนวนพัลส์
- หน่วยความจำสำหรับเก็บข้อมูลภายใน บางส่วนสามารถเข้าถึงข้อมูลได้ทั้งระบบไบต์และบิต เพื่อให้การออกแบบโปรแกรมและการควบคุมระบบทำได้ง่ายขึ้น
- มีคำสั่ง คูณ และ หาร เลข 8 บิตในตัวเอง โดยใช้เวลาเพียง 4 ไมโครวินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

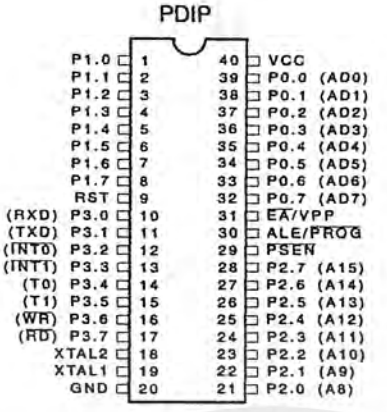
ไมโครคอนโทรลเลอร์ ตระกูล MCS-51 ที่เป็นเบอร์พื้นฐานในตระกูลนี้คือ 8031 ,8051 และเบอร์ 8751 ซึ่งมีขาจำนวน 40 ขาเท่ากัน ใช้เวลาการทำงานในแต่ละคำสั่งเท่ากัน ใช้แรงดันไฟฟ้าเท่ากัน แต่สิ่งที่ต่างกันคือ ขนาดของหน่วยความจำโปรแกรมภายในชิพ ซึ่งลักษณะทั่วไปของแต่ละเบอร์สามารถอธิบายได้ดังนี้

1. เบอร์ 8031 ไม่มีหน่วยความจำสำหรับเก็บโปรแกรมภายในชิพ แต่สามารถใช้หน่วยความจำภายนอกเพื่อเก็บโปรแกรมได้ถึง 64 กิโลไบต์ ซึ่งอาจใช้เป็น ROM (Read Only Memory) ,PROM (Programable Read Only Memory) , EPROM (Eraseable Programable Read Only Memory) เหมาะสำหรับกรณีที่โปรแกรมมีขนาดเล็กกว่า 4 กิโลไบต์ หรือมากกว่า 4 กิโลไบต์มาก
2. เบอร์ 8051 หลังจากทดสอบโปรแกรมจนไม่พบข้อผิดพลาดแล้ว จะเป็นช่วงของการผลิตจริง ซึ่งจะต้อง พิจารณาต้นทุนเป็นอันดับแรก ในการผลิตจริงจะใช้เบอร์นี้ ซึ่งมีหน่วยความจำเก็บโปรแกรมภายในแบบ ROM ขนาด 4 กิโลไบต์แทน เพราะราคาต่ำกว่ามาก แต่มีข้อจำกัดตรงที่ไม่สามารถแก้ไขโปรแกรมที่บันทึกลงไปแล้ว
3. เบอร์ 8751 มีหน่วยความจำสำหรับเก็บโปรแกรมภายในแบบ EPROM ขนาด 4 กิโลไบต์ สามารถใช้รังสีอัลตราไวโอเลตในการลบโปรแกรมเก่าที่มีอยู่ และบันทึกโปรแกรมใหม่ลงได้ทันที ทั้งนี้เพื่อความสะดวกในการแก้ไขหรือปรับปรุงโปรแกรม เหมาะสำหรับใช้ในการพัฒนาเบื้องต้น ซึ่งจำเป็นต้องทดสอบเพื่อหาข้อผิดพลาดของโปรแกรมและแก้ไขให้เรียบร้อยก่อนทำการผลิตจริง ซึ่งการแก้ไขสามารถทำได้ในจำนวนครั้งที่จำกัด

ปัจจุบันนี้ได้มีชิพไมโครคอนโทรลเลอร์ที่ผลิตโดยบริษัทต่างๆมากมาย เช่น ชิพเบอร์ 89C51 ของบริษัท ATMEL ซึ่งหน่วยความจำภายในใช้เทคโนโลยี Flash Memory สามารถลบและบันทึกได้ด้วยไฟฟ้า เหมาะสำหรับงานที่อยู่ในขั้นตอนการพัฒนา เพราะใช้งานง่าย สะดวก รวดเร็ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตำแหน่งขาของ MCS-51



ภาพ 2.1.1 ลักษณะของไมโครคอนโทรลเลอร์ 89C51

หน้าที่แต่ละขาของไมโครคอนโทรลเลอร์

GND ต่อลงกราวด์

Vcc ต่อแหล่งจ่ายไฟฟ้ากระแสตรงแรงดัน 5 โวลต์

Port0 มีขนาด 8 บิต เป็นแบบ Open Drain Bidirectional สามารถใช้งานเป็นพอร์ตอินพุต/เอาท์พุตทั่วไปได้ เมื่อต้องการใช้เป็นอินพุตจะต้องส่งค่า 1 ไปยังแต่ละบิตเพื่อให้เป็น High Impedance นอกจากนั้นยังใช้ติดต่อหน่วยความจำโปรแกรมและข้อมูลภายนอกได้อีกด้วย โดยทำหน้าที่ส่งแอดเดรสไบต์ต่ำ (A0-A7) ร่วมกับการรับส่งข้อมูล (D0-D7)จากหน่วยความจำภายนอก ในระหว่างการเขียนหรืออ่านข้อมูล โดยมีวงจรพูลอัพภายใน

Port1 มีขนาด 8 บิต สามารถใช้งานเป็นพอร์ตอินพุต/เอาท์พุตทั่วไปได้ เมื่อต้องการใช้เป็นอินพุตจะต้องส่งค่า 1 ไปยังแต่ละบิตเพื่อให้เป็น High Impedance โดยมีวงจรพูลอัพภายใน

Port2 มีขนาด 8 บิต เป็นแบบ Open Drain Bidirectional สามารถใช้งานเป็นพอร์ตอินพุต/เอาท์พุตทั่วไปได้ เมื่อต้องการใช้เป็นอินพุตจะต้องส่งค่า 1 ไปยังแต่ละบิตเพื่อให้เป็น High Impedance นอกจากนั้นยังใช้ติดต่อหน่วยความจำโปรแกรมและข้อมูลภายนอกได้อีกด้วย โดยทำหน้าที่ส่งแอดเดรสไบต์สูง (A8-A15) โดยมีวงจรพูลอัพภายใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Port3 มีขนาด 8 บิต สามารถใช้งานเป็นพอร์ตอินพุต/เอาท์พุตทั่วไปได้ เมื่อต้องการใช้เป็นอินพุตจะต้องส่งค่า 1 ไปยังแต่ละบิตเพื่อให้เป็น High Impedance โดยมีวงจรพูลอัพภายใน นอกจากนี้ยังใช้งานในหน้าที่พิเศษต่างๆดังนี้

- P3.0 (RxD) ใช้รับข้อมูลแบบอนุกรมจากภายนอก
- P3.1 (TxD) ใช้ส่งข้อมูลแบบอนุกรมสู่ภายนอก
- P3.2 (INT0) ใช้เป็นอินพุทสำหรับสัญญาณอินเทอร์รัพท์ภายนอกที่ 0
- P3.3 (INT1) ใช้เป็นอินพุทสำหรับสัญญาณอินเทอร์รัพท์ภายนอกที่ 1
- P3.4 (T0) ใช้เป็นอินพุทสัญญาณภายนอกให้เคาน์เตอร์ของไทเมอร์ 0
- P3.5 (T1) ใช้เป็นอินพุทสัญญาณภายนอกให้เคาน์เตอร์ของไทเมอร์ 1
- P3.6 (WR) ใช้เป็นสัญญาณควบคุมการเขียนข้อมูลหน่วยความจำข้อมูลภายนอก
- P3.7 (RD) ใช้เป็นสัญญาณควบคุมการอ่านข้อมูลหน่วยความจำข้อมูลภายนอก

RST ใช้สำหรับรีเซตชิพ โดยจะต้องให้ขานี้เป็น high อย่างน้อย 2 แมชชีนไซเคิล จึงจะเกิดการรีเซต โดยต้องต่อตัวต้านทานพูลดาวน์ไว้ (ประมาณ 10 K) ตลอดการทำงานเพื่อรักษาสถานะขานี้ให้เป็น low อยู่เสมอ และต่อพูตัวเก็บประจุประมาณ 10 ไมโครฟารัดคร่อมไว้กับขา Vcc เพื่อให้เกิดการรีเซตเองเมื่อเริ่มจ่ายไฟ

XTAL1 ใช้ต่อคริสตอลภายนอก ทำหน้าที่เป็นอินพุทของวงจรออสซิลเลเตอร์

XTAL2 ใช้ต่อคริสตอลภายนอก ทำหน้าที่เป็นเอาท์พุทของวงจรออสซิลเลเตอร์

PSEN ใช้ส่งสัญญาณสโตรบ เพื่ออ่านคำสั่งจากโปรแกรมที่เก็บไว้ในหน่วยความจำภายนอกชิพ เมื่อชิพทำงานด้วยโปรแกรมจากหน่วยความจำโปรแกรมภายนอก ขานี้จะส่งสัญญาณสโตรบ 2 ครั้งในแต่ละแมชชีนไซเคิล แต่เมื่อชิพทำงานด้วยโปรแกรมจากหน่วยความจำโปรแกรมภายใน จะไม่มีสัญญาณออกมาจากขานี้

ALE/PROG เป็นขาสำหรับส่งสัญญาณออกไปภายนอก เพื่อเพื่อควบคุมการแลทช์ค่าแอดเดรสไบต์ต่ำจากพอร์ต 0 ในระหว่างการติดต่อกับหน่วยความจำเก็บโปรแกรมหรือข้อมูลภายนอก ปกติเมื่อไม่มีการติดต่อหน่วยความจำโปรแกรมภายนอก ขานี้จะส่งสัญญาณพัลส์ออกมาด้วยความถี่ 1/6 ของความถี่ออสซิลเลเตอร์ตลอดเวลา ดังนั้นเราสามารถนำความถี่จากขานี้ไปใช้งานอย่างอื่นได้ โดยขานี้จะเอคทีพเมื่อมีการใช้คำสั่ง movc หรือ movx

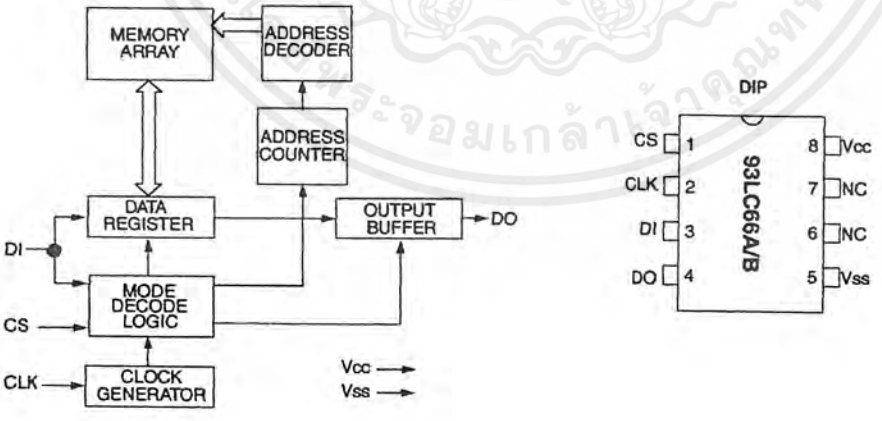
EA/Vpp เป็นขาสำหรับเลือกให้ MCS-51 ทำงานจากโปรแกรมที่อยู่ภายในชิพหรือภายนอกชิพ หากเป็น 0 หมายถึงให้ใช้โปรแกรมจากหน่วยความจำที่เก็บโปรแกรมภายนอก หากเป็น 1 หมายถึงให้ใช้โปรแกรมจากหน่วยความจำโปรแกรมภายในชิพ ซึ่งหากเป็น MCS-51 ที่มีหน่วยความจำโปรแกรมภายในชิพ ก็สามารถเลือกให้ทำงานได้ตามต้องการ แต่หากเป็น MCS-51 ที่ไม่มีหน่วยความจำโปรแกรมภายในจะต้องต่อขานี้ลงกราวน์เสมอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2หลักการของ Serial EEPROM

Serial EEPROM เป็นหน่วยความจำชนิดหนึ่งที่ได้รับการพัฒนา เพื่อประโยชน์ในการเพิ่มขนาดความจุของหน่วยความจำสำหรับระบบไมโครคอนโทรเลอร์ที่มีทรัพยากรระบบอยู่น้อย ด้วยขนาดที่เล็กกระทัดรัด และ ใช้สายสัญญาณเพียงไม่กี่เส้น แต่กลับมีความจุได้ตั้งแต่ 64 ไบท์ ถึง 8 กิโลไบท์ และยังมีแนวโน้มว่าจะพัฒนาให้มีความจุได้มากยิ่งขึ้นในขณะที่ยังคงจำนวนขาสัญญาณให้เท่าเดิมอยู่ ทำให้ Serail EEPROM เป็นที่นิยมในปัจจุบัน อีกทั้งยังใช้เทคโนโลยี CMOS ซึ่งกินกระแสไฟต่ำ(ไม่กี่มิลลิแอมป์ในขณะโปรแกรม และ ไม่กี่ไมโครแอมป์ในขณะปรกติ) และด้วยคุณสมบัติของ EEPROM ที่สามารถโปรแกรมและลบได้ด้วยไฟฟ้า ทั้งยังรักษาข้อมูลอยู่ได้แม้ว่าจะไม่มีไฟเลี้ยงแล้วก็ตาม ทำให้สามารถพัฒนาระบบไมโครคอนโทรเลอร์แบบประหยัดพลังงานได้ ด้วยคุณสมบัติดังกล่าวมาแล้วทั้งหมดสามารถสรุปได้ดังนี้

- มีขนาด 8 ขา ในลักษณะบรรจุภัณฑ์แบบ DIP และ PLCC
- ความจุหน่วยความจำตั่งแต่ 64 ไบท์ ถึง 8 กิโลไบท์
- กินกระแสไฟต่ำ
- สามารถโปรแกรม และ ลบด้วยไฟฟ้าได้
- ลบได้ 1,000,000 ครั้ง ช่วงอายุการเก็บข้อมูล 10 ปี(ตามที่ data sheet ของบริษัทผู้ผลิตได้กล่าวอ้าง)
- ใช้ไฟเลี่ยง +5.0V ซีกเดียว
- ใช้มาตรฐานการสื่อสารแบบ Microwire หรือ I2C หรือ SPI (ตามแต่จะเลือกใช้)



ภาพที่ 2.2.1 แสดงบล็อกไดอะแกรมและลักษณะของชิพ EEPROM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งาน Serial EEPROM

ตามที่ได้กล่าวไว้แล้วข้างต้นว่า Serial EEPROM มีมาตรฐานการเชื่อมต่ออยู่สามแบบ ทั้งนี้ขึ้นอยู่กับความต้องการและความสะดวกของผู้ใช้เองว่าต้องการจะเลือกใช้งานในชนิดใด

สำหรับโครงงานนี้ได้เลือกใช้แบบ Microwire ซึ่งอยู่ในตระกูล 93cXX เชื่อมต่อกับระบบด้วยขา 4 ขา คือ ขาสัญญาณนาฬิกา(SK), ขาข้อมูลเข้า(DI), ขาข้อมูลออก(DO), และขาเลือกไอซี (CS) หลักการของระบบ Microwire นั้นจะเป็นการสื่อสารแบบอนุกรมโดยมีขาสัญญาณนาฬิกา (SK) ซึ่งกำเนิดจากตัวประมวลผลหลัก(Master) เพื่อบอกจังหวะของการรับส่งข้อมูลผ่านขาข้อมูลเข้า/ออก(DI/DO) โดยระบบการสื่อสารทั้งหมดจะถูกควบคุมด้วยขาสัญญาณเลือกการทำงาน(CS) ทั้งนี้ข้อมูลที่ทำการสื่อสารกับตัว EEPROM จะต้องตรงตามมาตรฐานของ Microwire ซึ่งจะเป็นตัวบอกให้ EEPROM ทราบว่าจะทำการอ่าน หรือเขียนข้อมูลกับตัว EEPROM มาตรฐาน Microwire ดังกล่าวได้สรุปไว้ดังตาราง

ตาราง 2.2.1 รูปแบบของข้อมูลสำหรับควบคุมการทำงาน EEPROM

Instruction	SB	Opcode	Address									Data In	Data Out	CLK.
ERASE	1	11	A8	A7	A6	A5	A4	A3	A2	A1	A0	-	RDY/BSY	12
ERAL	1	00	1	0	X	X	X	X	X	X	X	-	RDY/BSY	12
EWDS	1	00	0	0	X	X	X	X	X	X	X	-	HIGH-Z	12
EWEN	1	00	1	1	X	X	X	X	X	X	X	-	HIGH-Z	12
READ	1	10	A8	A7	A6	A5	A4	A3	A2	A1	A0	-	D7-D0	20
WRITE	1	01	A8	A7	A6	A5	A4	A3	A2	A1	A0	D7-D0	RDY/BSY	20
WRAL	1	00	0	1	X	X	X	X	X	X	X	D7-D0	RDY/BSY	20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งานในแต่ละคำสั่ง

ทั้งคำสั่ง ข้อมูล และ ตำแหน่งจะถูกส่งเข้าไปในตัว EEPROM ที่ขา 'DI' โดยอิงกับสัญญาณขอบขาขึ้นของขา 'CLK' ปรกติแล้วขา 'DO' จะเป็น High impedance ยกเว้นในขณะที่ทำการอ่านข้อมูลจากตัว EEPROM หรือในขณะที่ทำการตรวจสอบสถานะ READY/BUSY สถานะ READY/BUSY สามารถตรวจสอบหลังจากทำการ ERASE/WRITE โดยทำการพูลลิ่งขา 'DO' ถ้าเป็นศูนย์ แสดงว่าการโปรแกรมกำลังอยู่ในกระบวนการ ในขณะที่เป็นหนึ่งแสดงถึงอุปกรณ์อยู่ในสถานะพร้อมทำงาน

1. **เงื่อนไขเริ่มต้น** ตามเงื่อนไขของการสื่อสารในมาตรฐาน Microwire จำเป็นต้องมีสัญญาณเริ่มต้นของการสื่อสาร คล้ายคลึงกับ UART เงื่อนไขเริ่มต้นของ Microwir จะถูกควบคุมโดยขา CS, DI และ SK โดยที่ CS และDI เป็น high ในขณะที่ขา SK เปลี่ยนสถานะจาก low เป็น high ตัว EEPROM จะออกจากโหมด standby และรับรู้ถึงการส่งข้อมูลที่จะเกิดขึ้น ตัว EEPROM จะยังไม่กลับสู่สถานะ standby ถ้าสัญญาณที่ขา CS ยังคงเป็น high

2. **คำสั่งลบ** 'ERASE' คำสั่งลบจะทำให้ข้อมูลในตำแหน่งที่กำหนดเป็น high ทั้งหมด โดยกระบวนการลบจะเริ่มขึ้นเมื่อสัญญาณที่ขา CS เปลี่ยนจาก low เป็น high และ หลังจากกระบวนการลบเสร็จสิ้นแล้ว ควรทำการตรวจเช็คสถานะของไอซีก่อนที่จะทำคำสั่งถัดไป ซึ่งกระบวนการลบจะใช้เวลาประมาณ 250 นาโนวินาที

3. **คำสั่ง ลบทั้งหมด** 'ERASE ALL (ERAL)' คำสั่งลบทั้งหมดจะเหมือนกับคำสั่งลบทุกประการยกเว้น Opcode เท่านั้น ซึ่ง Opcode ดังกล่าวสามารถดูได้จากตารางที่ 1 และเช่นกันที่ต้องทำการตรวจสถานะ READY/BUSY ก่อนทำคำสั่งถัดไป

4. **คำสั่งอนุญาติ หรือ ยกเลิก การลบ/เขียน** 'EWDS/EWEN' Serial EEPROM ในตระกูล 93cXX สามารถทำการป้องกันการเปลี่ยนแปลงของข้อมูลได้ด้วยโปรแกรม โดยสถานะการยินยอมให้มีการเขียน จะอยู่ในสถานะป้องกันการเขียนเมื่อเริ่มจ่ายไฟเลี้ยง และจะเปลี่ยนสถานะเป็นอนุญาติให้มีการเขียนได้เมื่อได้รับคำสั่ง 'EWEN' และในระหว่างการทำงานทุกเมื่อสามารถเปลี่ยนสถานะการเขียนดังกล่าว กลับมาเป็นป้องกันการเขียนได้ด้วยคำสั่ง 'EWDS' หรือเมื่อไม่มีไฟเลี้ยง แต่ไม่ว่าจะอยู่ในสถานะใดก็ตาม คำสั่งในการอ่านข้อมูลจะยังคงกระทำได้อยู่

5. **คำสั่งอ่านข้อมูล** 'READ' คำสั่งอ่านข้อมูลจะทำการอ่านข้อมูลในตำแหน่งที่กำหนดและส่งข้อมูลในตำแหน่งดังกล่าวออกมายังขา DO ซึ่งข้อมูลดังกล่าวจะเป็น 8 บิท หรือ 16 บิท ขึ้นกับการกำหนดโครงสร้างที่ ORG การส่งข้อมูลออกจะกระทำในขณะที่ขา SK เปลี่ยนสถานะเป็น low และจะคงที่เพื่อให้ทำการอ่านเมื่อขา SK เป็น high การอ่านยังสามารถทำการอ่านอย่างต่อเนื่องได้เนื่องจากเมื่อทำการอ่านไปแล้ว 1 word วงจรภายในจะเพิ่มค่าของตำแหน่งโดยอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. **คำสั่งเขียนข้อมูล** 'WRITE' คำสั่งเขียนจะทำการเขียนข้อมูลในหนึ่งตำแหน่งที่กำหนด เมื่อข้อมูลเลื่อนเข้าสู่ขา DI จนครบ 8/16 บิท และเมื่อข้อมูลตัวสุดท้ายเข้าสู่ EEPROM พร้อมกับขา CS ตกลงเป็น low กระบวนการลบ และ เขียนโดยอัตโนมัติจึงจะเกิดขึ้น โดยใช้ช่วงเวลาประมาณ 250 นาโนวินาที

7. **คำสั่งเขียนทั้งหมด** 'WRAL' คำสั่งเขียนทั้งหมดจะคล้ายคลึงกับคำสั่งเขียน แต่ต่างกันที่ คำสั่งเขียนทั้งหมด จะทำการเขียนข้อมูล 1 word ลงบนทุกๆตำแหน่งของ EEPROM ซึ่งผลลัพธ์ที่ได้นั้น ทุกตำแหน่งในหน่วยความจำจะเป็นค่าเดียวกันหมด

2.3 การใช้งาน LCD Module

ในปัจจุบัน LCD ถูกใช้เป็นอุปกรณ์แสดงผลในเครื่องมือเครื่องใช้ทั่วไปอย่างกว้างขวาง เนื่องจาก LCD กินกระแสต่ำ และ สามารถแสดงผลตัวอักษรได้อย่างชัดเจน แต่เดิมนั้น LCD ไม่ค่อยนิยมใช้กันนักเนื่องจากต้องใช้วงจรที่มีระบบการทำงานที่ซับซ้อน และมีราคาค่อนข้างสูง ผู้ผลิตจึงได้คิดค้นวงจรสำเร็จรูปที่ใช้ควบคุม LCD ซึ่งจะควบคุมการทำงานของ LCD โดยอัตโนมัติ และสามารถควบคุมได้โดยตรงจากวงจรภายนอก ซึ่งจะเรียกว่า LCM ดังนั้น LCD ในปัจจุบันจึงใช้งานได้ง่ายขึ้น และด้วยราคาต่อหน่วยที่ถูกลง จึงมีผู้นิยมใช้ LCD แบบโมดูลมากขึ้น

LCM มีอยู่หลายแบบเช่น แบบ ตัวอักษร DOT Matrix และแบบกราฟิก ในที่นี้จะขออธิบายถึงเฉพาะแบบตัวอักษร DOT Matrix เนื่องจากในโครงงานนี้ใช้ LCM แบบ DOT Matrix ก็เพียงพอ รวมทั้งราคาต่อหน่วยที่ถูกกว่าแบบกราฟิก ใช้งานได้ง่ายกว่า และยังใช้ทรัพยากรระบบน้อยกว่าด้วย คุณสมบัติของ LCD แบบ DOT Matrix สามารถสรุปได้ดังนี้

- ตัวอักษรที่แสดงผลเป็นแบบ DOT Matrix ขนาด 5X8 DOT
- มีจำนวนของตัวอักษรต่อบรรทัด และจำนวนบรรทัดที่แตกต่างกันเช่น 16x1, 16x2, 20x1, 20x2, 40x1, 40x2 เป็นต้น
- สามารถต่อกับระบบไมโครคอนโทรเลอร์โดยตรง ทั้งในแบบ 8บิท และ 4บิท
- การใช้งานง่ายด้วยการส่งคำสั่ง และข้อความจากไมโครคอนโทรเลอร์ไปยังตัว LCM ซึ่งข้อความดังกล่าวจะถูกแสดงผลตลอดเวลา ทำให้ไมโครคอนโทรเลอร์ไม่ต้องเสียเวลากับการควบคุมการแสดงผลแบบซ้ำๆ
- มีชุดคำสั่งเพียง11คำสั่ง คือ Clear Display, Return Home, Entry Mode Set, Display On/Off, Cursor & Display Shift, Function Set, Set CGRam Address, Set DDRam Address, Read Busy Flag & Address, Write Data to CGRam or DDRam, Read Data to CGRam or DDRam

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สามารถแสดงตัวอักษรได้ 192 แบบ(ตามมาตรฐาน ASCII) และ ตัวอักษรที่กำหนดขึ้นเองอีก 8 ตัวอักษร
- สามารถใช้ไฟเลี้ยง 5V กินกระแสต่ำ และมีน้ำหนักเบา

การต่อวงจรเข้ากับ LCM

ดังที่ได้กล่าวไว้แล้วว่า LCM สามารถต่อกับอุปกรณ์ควบคุมภายนอกได้สองแบบ คือ แบบ 8บิท และ 4บิท ดังแสดงในภาพ ซึ่งโดยรวมจะมีการต่อวงจรที่คล้ายคลึงกัน เว้นแต่ในส่วนขาข้อมูล ซึ่งในแบบ 4บิทจะใช้ทรัพยากรของระบบน้อยกว่า แบบ 8บิท จึงเหมาะกับระบบที่มีอุปกรณ์ต่อพ่วงหลายอย่าง แต่ไม่ต้องการขยายระบบบัสด้วย 8255 ซึ่งเป็นระบบบัสแบบเก่า และ กินไฟเลี้ยงสูง

ชุดคำสั่งควบคุมการแสดงข้อความ

ชุดคำสั่งในตัว LCM มีด้วยกัน 11 คำสั่ง ดังแสดงในตารางที่ ซึ่งเป็นการกำหนดคุณสมบัติต่างๆในการใช้งานรวมถึงการเขียนข้อมูลที่เป็นข้อความที่ต้องการใช้ปรากฏบนส่วนแสดงผล ก่อนที่เราจะมาทำความเข้าใจในแต่ละคำสั่ง ควรทำความเข้าใจในพื้นฐานเสียก่อนดังนี้

i. การติดต่อข้อมูลกับ LCM สามารถแบ่งได้เป็นสองลักษณะคือ Instruction และ Data โดยความแตกต่างของลักษณะทั้งสองจะถูกกำหนดด้วยขาสัญญาณ 'RS' นั้นคือ ถ้า 'RS' เป็น '0' จะหมายถึงการติดต่อกับ Instruction และถ้าเป็น '1' จะเป็นการติดต่อกับ Data

ii. หลังจากการเขียนข้อมูล หรือคำสั่งให้กับ LCM แล้ว ตัวโมดูลจะใช้เวลาชั่วขณะหนึ่งในการประมวลผลข้อมูล หรือคำสั่งดังกล่าว เราสามารถตรวจความพร้อมในการทำงานได้จาก 'Busy Flag(BF)' ซึ่งในขณะที่ LCM ยังไม่พร้อมที่จะทำงาน จะไม่สามารถเขียนข้อมูลใหม่ หรือ คำสั่งใดๆ ได้

iii. หากใช้ระบบบัสแบบ 4บิท จะต้องทำการเขียนข้อมูลซ้ำสองครั้งในหนึ่งไบท์ คือ เขียน 4 บิทบนก่อนแล้วจึงเขียน 4 บิทล่างตาม ทั้งนี้ต้องทำการกำหนดบิท 'DL' ในคำสั่ง 'Function set' เสียก่อน

iv. DDRam(Display Data Ram) คือหน่วยความจำภายในตัว LCM ที่เป็นที่สำรองข้อมูลที่เป็นข้อความที่ต้องการแสดงผล

v. CGRam(Character Generator Ram) คือหน่วยความจำภายในตัว LCM สำหรับเก็บภาพตัวอักษรที่ผู้ใช้สามารถกำหนดเองได้ 8 ตัวอักษร ซึ่งมีตำแหน่งที่สามารถอ้างได้ทั้งหมด 64ไบท์ (8 ตัวอักษร X 8 แถว)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดในแต่ละคำสั่งมีดังนี้

1. Clear Display : คำสั่ง Clear Display เป็นคำสั่งที่จะทำการเขียนตัวอักษรช่องว่าง(Space) ลงใน DDRam ทั้งหมด และเลื่อนเคอร์เซอร์ ไปยังซ้ายสุดของจอภาพ

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	0	0	0	1

2. Cursor At Home : ทำการเลื่อนตำแหน่งของเคอร์เซอร์ไปยังซ้ายสุดของจอภาพ โดยที่ไม่มีการเปลี่ยนแปลงข้อมูลใน DDRam

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	0	0	1	*

3. Entry Mode Set : เป็นคำสั่งในการกำหนดทิศทางการเลื่อนของเคอร์เซอร์ รวมถึงการสั่งใช้งานหรือยกเลิกคุณสมบัติดังกล่าว

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	0	0	I/D	S

I/D = 0 กำหนดทิศทางของเคอร์เซอร์ และDDRam ให้เป็นแบบลดลงทีละหนึ่ง

I/D = 1 กำหนดทิศทางของเคอร์เซอร์ และDDRam ให้เป็นแบบเพิ่มขึ้นทีละหนึ่ง

S = 0 เมื่อเขียนข้อมูลแล้วตัวเคอร์เซอร์จะถูกเลื่อนไปตามทิศทางที่ค่า I/D กำหนดไว้

S = 0 เมื่อเขียนข้อมูลแล้วตัวเคอร์เซอร์จะอยู่กับที่ แต่ตัวอักษรจะถูกดันไปตามทิศทางที่ค่า I/D กำหนดไว้

4. Display ON/OFF : เป็นคำสั่งในการกำหนดคุณสมบัติของการแสดงผลของหน้าจอ เคอร์เซอร์ และ การกระพริบของเคอร์เซอร์

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	1	D	C	B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

D = 0 กำหนดให้หน้าจอดับ

D = 1 กำหนดให้หน้าจอแสดงผล

C = 0 กำหนดให้ซ้อนเคอร์เซอร์

C = 1 กำหนดให้แสดงเคอร์เซอร์

B = 0 กำหนดให้เคอร์เซอร์ไม่กระพริบ

B = 1 กำหนดให้เคอร์เซอร์กระพริบ

5. Cursor & Display Shift : เป็นคำสั่งในการเลื่อนเคอร์เซอร์พร้อมกับเลื่อนหน้าจอ โดยไม่มีการเปลี่ยนแปลงข้อมูลใน DDRam

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	1	S/C	R/L	*	*

S/C = 0 กำหนดให้เลื่อนเคอร์เซอร์ไปหนึ่งตำแหน่งตามทิศทาง R/L

S/C = 1 กำหนดให้เลื่อนข้อความบนแผงแสดงตามทิศทาง R/L ไปหนึ่งคอลัมป์ (เลื่อนทุกบรรทัด)

R/L = 0 กำหนดทิศทางการเลื่อนไปทางซ้าย

R/L = 1 กำหนดทิศทางการเลื่อนไปทางขวา

6. Function Set : เป็นคำสั่งในการกำหนดขนาดของบัสรับส่งข้อมูล จำนวนแถว และ ขนาดของตัวอักษร

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	1	DL	N	F	*	*

DL = 0 กำหนดขนาดของบัสข้อมูลแบบ 4บิท

DL = 0 กำหนดขนาดของบัสข้อมูลแบบ 8บิท

N = 0 กำหนดจำนวนบรรทัดแบบ 1/8 duty และ 1/11 duty

N = 1 กำหนดจำนวนบรรทัดแบบ 1/16 duty

F = 0 กำหนดขนาดตัวอักษรเป็น 5*7 dots

F = 1 กำหนดขนาดตัวอักษรเป็น 5*11 dots

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

** จะสังเกตว่าการกำหนดค่า DL นี้ สามารถกระทำได้ที่ DB4-DB7 ซึ่งถ้ามีการกำหนดให้เป็นแบบ 4 บิทตั้งแต่แรก หลังจากจ่ายไฟเลี้ยงจะทำให้ LCM อยู่ในโหมดการรับข้อมูลแบบ 4 บิททันที

7. Set CGRam Address: เป็นคำสั่งในการกำหนดตำแหน่งของ DDRam ซึ่งหลังจากที่ได้กำหนดตำแหน่งไว้แล้ว การอ่านและ เขียนข้อมูลจะเป็นไปตามตำแหน่งที่กำหนดทันที

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	1	CGRam Address					

8. Set DDRam Address : เป็นคำสั่งในการกำหนดตำแหน่งของ DDRam ซึ่งหลังจากที่ได้กำหนดตำแหน่งไว้แล้ว การอ่านและ เขียนข้อมูลจะเป็นไปตามตำแหน่งที่กำหนดทันที ตำแหน่งของข้อมูลใน LCM แต่ละรุ่นอาจมีตำแหน่งเริ่มต้นที่ไม่เหมือนกันทั้งนี้ต้องพิจารณาตามคู่มือของ LCM รุ่นนั้นๆ

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	1	DDRam Address						

9. Read Busy Flag & Address : เป็นคำสั่งสำหรับอ่านค่า BF ซึ่งบอกถึงความพร้อมของ LCM ในการรับข้อมูล ถ้า BF = 0 หมายถึงว่าพร้อมที่จะรับข้อมูลต่อไปได้ แต่ถ้า BF = 1 หมายถึงว่ายังไม่พร้อม นอกจากนี้ยังเป็นการอ่านค่าตำแหน่งของ CGRam หรือ DDRam ด้วย

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	1	BF	Address						

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การอ่านและเขียนข้อมูลกับ DDRam และ CGRam

1. การเขียนข้อมูลลงยัง DDRam หรือ CGRam

สำหรับการเขียนข้อมูลลงในหน่วยความจำ DDRam หรือ CGRam โดยเมื่อทำการเขียนแล้ว ตำแหน่งจะถูกเพิ่มขึ้นหรือลดลงโดยอัตโนมัติ ตามที่กำหนดไว้ที่บิท I/D ในคำสั่ง Entry Mode Set และการเขียนจะเป็น DDRam หรือ CGRam ก็ขึ้นกับคำสั่งก่อนหน้ามีการกำหนดตำแหน่งไว้ที่ใด

2. การอ่านข้อมูลจาก DDRam หรือ CGRam

สำหรับการอ่านข้อมูลจากหน่วยความจำ DDRam หรือ CGRam โดยเมื่อทำการอ่านแล้ว ตำแหน่งจะถูกเพิ่มขึ้นหรือลดลงโดยอัตโนมัติ ตามที่กำหนดไว้ที่บิท I/D ในคำสั่ง Entry Mode Set และการอ่านจะเป็น DDRam หรือ CGRam ก็ขึ้นกับคำสั่งก่อนหน้ามีการกำหนดตำแหน่งไว้ที่ใด

แนวทางการเขียนโปรแกรมควบคุม

1. เมื่อจ่ายไฟเลี้ยงให้กับ LCM และระดับแรงดันไฟขึ้นถึง 4.5V วงจรรีเซ็ทภายใน LCM จะทำงาน ซึ่งจะใช้เวลา 10 มิลลิวินาที ทั้งนี้ระบบรีเซ็ทดังกล่าวจะทำกระบวนการดังนี้
   - ทำการเคลียร์จอภาพทั้งหมด
   - กำหนดคุณสมบัติด้วยคำสั่ง 'Function Set' โดยกำหนดให้ DL = 1 (บัสข้อมูลแบบ 8 บิท), N = 0 (แสดงข้อมูล 1 บรรทัด), และ F = 0 (กำหนดขนาดตัวอักษร 5*7 dots)
   - กำหนดคุณสมบัติด้วยคำสั่ง 'Display ON/OFF' โดยกำหนดให้ I/D = 1 (เพิ่ม), S = 0 (ไม่เลื่อน)

* การใช้งาน LCM ต้องรอให้กระบวนการรีเซ็ทภายในทำงานให้เสร็จสิ้นเสียก่อน ซึ่งจะตรวจสอบได้ด้วย BF (Busy Flag) หรืออาจจะใช้การหน่วงเวลาก็ได้

2. การใช้งาน LCM จำเป็นต้องควบคุมด้วยโปรแกรมเป็นส่วนใหญ่ ชุดคำสั่งต่างๆรวมทั้งการอ่านหรือการเขียนข้อมูล จะถูกกำหนดด้วยขาสัญญาณทั้งหมดที่มีอยู่ โดยปรกติโปรแกรมจะต้องกำหนดคุณสมบัติต่างๆที่ต้องการไว้ที่ส่วนต้น และจากนั้นก็จะเป็นการอ่าน และเขียนข้อมูลลงใน DDRam ซึ่งก็คือข้อความที่จะให้แสดงนั้นเอง

การออกแบบตัวอักษรเอง

LCM สามารถแสดงตัวอักษรที่ออกแบบเองได้ โดยแสดงได้เพียง 8 ตัวอักษร ซึ่งมีหลักการดังนี้ ให้ทำการโหลดรูปแบบที่กำหนดไว้แล้ว ลงในหน่วยความจำ CGRam Address ที่อ้างถึง CGRam จะใช้เพียง 6 บิท โดยอ้างได้ทั้งหมด 64 ไบท์ (คือ 8 ตัวอักษร คูณกับ 8 แถว) และ การเรียกใช้ตัวอักษรที่ออกแบบเองนี้ จะใช้รหัส 00-07 (หรือ 08-0F ก็ได้) ขั้นตอนในการโหลดตัวอักษรมี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนี้

- กำหนดตำแหน่งของ CGRam ด้วยคำสั่ง 'Set CGRam' ทั้งนี้ถ้าโหลดอักษรตัวแรกก็จะใช้ตำแหน่งศูนย์

- เขียนข้อมูลที่เป็นแบบของตัวอักษรด้วยคำสั่ง 'Write Data' โดยใช้ 8 ไบท์ ต่อหนึ่งตัวอักษร คือ เรียงแถวจากแถวบนลงล่าง และใช้บิทที่ 0-4 เท่านั้น การโหลดนี้จะทำได้ต่อเนื่องไปเรื่อยๆโดยไม่ต้องกำหนดตำแหน่งใหม่

- ตัวอักษรที่โหลดไปแล้ว สามารถเรียกใช้ได้ตลอดไปในขณะที่ยังมีไฟเลี้ยงอยู่ ซึ่งการโหลดนี้ให้กระทำที่ส่วนต้นของโปรแกรมได้เลย

2.4 การเชื่อมต่อแบบมาตรฐาน RS-232

ในการเชื่อมต่ออุปกรณ์ต่างๆเข้ากับคอมพิวเตอร์นั้น วิธีหนึ่งที่นิยมใช้กันคือ การเชื่อมต่อผ่านทางพอร์ตสื่อสารอนุกรม ซึ่งเป็นไปตามมาตรฐาน RS-232 กำหนดขึ้นโดยสถาบัน Electronics Industrials Association (EIA) ทั้งนี้เพื่อให้มีรูปแบบสัญญาณของอุปกรณ์ที่จะมาเชื่อมต่อสอดคล้องกัน เนื่องจากระดับโวลเตจที่ใช้และการแทนความหมายของระดับลอจิกตามมาตรฐานนี้ แตกต่างจากที่ใช้งานกันในระบบดิจิตอลทั่วไป โดยระดับสัญญาณของ RS-232 เป็นแบบไบโพลาร์ ระดับโวลเตจทางด้านลบช่วง –3 V ถึง –20 V จะแทนค่าลอจิก 1 และโวลเตจทางด้านบวกช่วง +3 V ถึง +20 โวลต์ จะแทนค่าลอจิก 0 ดังนั้นจะเห็นได้ว่ามีความจำเป็นที่จะต้องเพิ่มอุปกรณ์หรือวงจรพิเศษเข้าไป เพื่อเปลี่ยนระดับโวลเตจจากระบบดิจิตอล 0 กับ 5 โวลต์ ไปเป็นมาตรฐาน RS-232 ซึ่งแสดงให้เห็นว่าระดับสัญญาณแบบ TTL จากขาสัญญาณ TxD และ RxD ของไมโครคอนโทรลเลอร์ จะต้องเปลี่ยนเป็นระดับสัญญาณตามมาตรฐาน RS-232 ก่อนที่จะทำการส่งไปในสายสัญญาณ

ชนิดของขั้วต่อที่นิยมใช้กัน จะเป็นแบบ D-subminiature connector 25 ขา หรือ 9 ขา และมีการจัดตำแหน่งของขาสัญญาณตามตาราง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขาที่	สัญญาณของขั้วต่อ 25 ขา	สัญญาณของขั้วต่อ 9 ขา
1	GND	GND
2	Tx (Transmit Data)	Tx (Transmit Data)
3	Rx (Receive Data)	Rx (Receive Data)
4	RTS (Request to Send)	RTS (Request to Send)
5	CTS (Clear to Send)	CTS (Clear to Send)
6	DSR (Data Set Ready)	DSR (Data Set Ready)
7	Ground	Ground
8	DCD (Data Carrier Detect)	DCD (Data Carrier Detect)
9	NC	NC
10	NC	-
11	NC	-
12	NC	-
13	NC	-
14	NC	-
15	NC	-
16	NC	-
17	NC	-
18	NC	-
19	NC	-
20	DTR (Data Terminal Ready)	-
21	NC	-
22	RI (Ring Indicator)	-
23	NC	-
24	NC	-
25	NC	-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
36694

ความหมายของขาต่าง ๆ

- Tx (Transmit Data) เป็นขาส่งสัญญาณออก
- Rx (Receive Data) เป็นขารับสัญญาณเข้า
- RTS (Request to Send) เป็นขาที่บอกว่าตัวเองพร้อมที่จะส่งข้อมูลแล้ว
- CTS (Clear to Send) เป็นขาที่บอกเครื่องอื่นว่าพร้อมจะรับข้อมูลแล้ว
- DSR (Data Set Ready) เป็นขาที่บอกคอมพิวเตอร์ว่าโมเด็มต่อเข้ากับสายโทรศัพท์เรียบร้อย และพร้อมส่งข้อมูลได้
- Ground เป็นขากราวน์ของสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่3

ขั้นตอนการดำเนินการ

3.1 หลักการของการตรวจสอบมิเตอร์น้ำ

รายละเอียดของมิเตอร์น้ำที่ทำการทดสอบ มีดังนี้

- ชนิดของมิเตอร์มีด้วยกัน 6 ชนิดคือ PXS, TDS, TDA(UHM), TDA(PWA), PXA (UHM), TDA(MWA)
- ขนาดของมิเตอร์มีด้วยกัน 3 ขนาดคือ 15 , 20, และ 25 มิลลิเมตร

ภาพที่ 3.1.1 แสดงมิเตอร์ที่ทำการทดสอบขนาดต่างๆ

พารามิเตอร์ในการทดสอบของในแต่ละชนิดและแต่ละขนาดจะแตกต่างกันไปซึ่งจะสามารถแบ่งกลุ่มของพารามิเตอร์ตามชนิดของมิเตอร์ได้ 3 กลุ่มคือ

- กลุ่ม 1 PXS, TDS และ TDA(UHM) มิเตอร์ในกลุ่มนี้จะทำการทดสอบ 3 รอบ โดยในแต่ละรอบจะมีขนาดของอัตราการไหล และปริมาณการไหลแตกต่างกัน ดังตารางที่ 3.1.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.1.1 แสดงพารามิเตอร์ของมิเตอร์ในกลุ่มที่หนึ่ง

ขนาดของมิเตอร์ (มิลลิเมตร)	อัตราการไหล(ลิตร/ชั่วโมง) และ ปริมาณน้ำ(ลิตร)		
	รอบที่หนึ่ง	รอบที่สอง	รอบที่สาม
15	1500/100	120/20	30/20
20	2500/100	200/50	50/20
25	3500/100	280/50	70/20
ความคลาดเคลื่อนที่ยอมรับได้(%)	2	2	5

- กลุ่มที่ 2 TDA(PWA), และ PXA(UHM) มิเตอร์ในกลุ่มนี้จะทำการทดสอบ 3 รอบ ดังนี้

ตารางที่ 3.1.2 แสดงพารามิเตอร์ของมิเตอร์ในกลุ่มที่สอง

ขนาดของมิเตอร์ (มิลลิเมตร)	อัตราการไหล(ลิตร/ชั่วโมง) และ ปริมาณน้ำ(ลิตร)		
	รอบที่หนึ่ง	รอบที่สอง	รอบที่สาม
15	1500/200	120/50	30/20
20	2500/200	200/50	50/20
25	3500/500	280/50	70/20
ความคลาดเคลื่อนที่ยอมรับได้(%)	2	2	5

- กลุ่มที่ 3 TDA(MWA) มิเตอร์ในกลุ่มนี้จะทำการทดสอบ 2 รอบ ดังนี้

ตารางที่ 3.1.3 แสดงพารามิเตอร์ของมิเตอร์ในกลุ่มที่สาม

ขนาดของมิเตอร์ (มิลลิเมตร)	อัตราการไหล(ลิตร/ชั่วโมง) และ ปริมาณน้ำ(ลิตร)	
	รอบที่หนึ่ง	รอบที่สอง
15	1500/200	60/20
20	2400/500	100/50
25	3000/500	150/50
ความคลาดเคลื่อนที่ยอมรับได้(%)	2	5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทั้งนี้ส่วนรับข้อมูลจำเป็นต้องกระทำกระบวนการทดสอบให้ได้ตามพารามิเตอร์ที่มีอยู่ในแต่ละกลุ่มของการทดสอบ เช่น ในกลุ่มที่หนึ่ง และ กลุ่มที่สาม จะมีความแตกต่างที่เห็นได้ชัดคือจำนวนรอบของการทดสอบ ซึ่ง ตัวรับข้อมูลต้องโต้ตอบกับผู้ใช้ให้ถูกต้อง และ จำเป็นต้องใช้ข้อมูลของพารามิเตอร์ในการคำนวณหาความคลาดเคลื่อน เพื่อส่งให้กับส่วนแสดงผลได้ทำการแสดงผลเบื้องต้นเพื่อบอกผู้ทำการทดสอบถึงมิเตอร์ตัวที่มีความผิดพลาด

หลังจากที่ผู้ใช้ได้เลือกชนิด และขนาดของมิเตอร์แล้ว ผู้ใช้ต้องป้อนข้อมูลพื้นฐานของมิเตอร์นั้นคือ เลขประจำตัวมิเตอร์, ค่าเริ่มต้นของมิเตอร์ก่อนการทดสอบ โดยต้องป้อนให้ครบทุกตัวที่อยู่ในระบบการทดสอบ จำนวนมิเตอร์สูงสุดที่จะทำการทดสอบในหนึ่งระบบ จะขึ้นกับขนาดของมิเตอร์ ดังแสดงในตารางที่ 3.1.4 หลังจากทำการป้อนข้อมูลพื้นฐานจนครบทุกตัวแล้ว จึงเริ่มทำการทดสอบในรอบที่หนึ่ง

ตารางที่ 3.1.4 แสดงจำนวนมิเตอร์สูงสุดของมิเตอร์ในแต่ละขนาด

ขนาดของมิเตอร์(mm)	15	20	25
จำนวนมิเตอร์สูงสุด(ตัว)	12	10	8

เมื่อการทดสอบรอบแรกเสร็จสิ้นแล้ว ผู้ใช้จะต้องป้อนข้อมูลที่อ่านได้จากตัวมิเตอร์หลังจากการทดสอบเสร็จสิ้น ตัวรับข้อมูลจะทำการคำนวณความคลาดเคลื่อนของมิเตอร์แต่ละตัวโดยอิงกับชนิด, ขนาด, และพารามิเตอร์อื่นๆของมิเตอร์ในระบบทดสอบ ผลการคำนวณที่ได้จะออกมาในรูปของความคลาดเคลื่อนที่เป็นเปอร์เซ็นต์ และจะส่งข้อมูลที่ได้หลังจากการคำนวณไปยังส่วนแสดงผล ทั้งนี้เปอร์เซ็นต์ที่คำนวนได้จะเป็นเลขจำนวนเต็มซึ่งได้จากการปัดทศนิยมลงทั้งหมด ทำให้ผลการคำนวณไม่มีความละเอียดเพียงพอที่จะแสดงในรายงานสุดท้าย แต่สามารถยอมรับได้ในการแสดงผล ถึงกระนั้นผลที่ได้จากการคำนวณจะไม่มีการบันทึกเก็บไว้ และไม่มีการส่งข้อมูลดังกล่าวไปยังส่วนสร้างรายงาน เนื่องจากส่วนสร้างรายงานจะทำการคำนวณผลลัพธ์อีกครั้ง ซึ่งจะมีความละเอียดในระดับทศนิยมสองหลัก เพื่อใช้ในการทำรายงานสรุปสุดท้าย

ทุกรอบที่ทำการทดสอบเสร็จสิ้น ส่วนแสดงผลจะทำการแสดงผลลัพธ์ของรอบที่ทำการทดสอบนั้น และเมื่อทำการทดสอบจนครบตามจำนวนรอบการทดสอบของมิเตอร์แต่ละชนิดแล้ว ตัวรับข้อมูลจึงจะส่งข้อมูลทั้งหมดในระบบทดสอบนั้นไปยังส่วนสร้างรายงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ส่วนรับข้อมูล

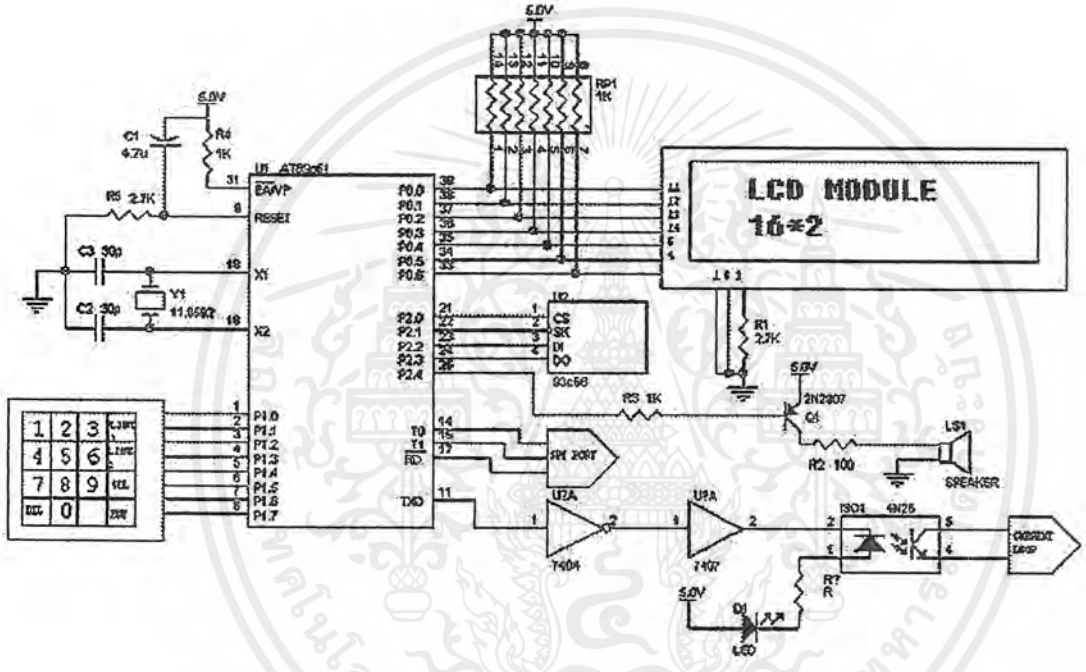
ส่วนรับข้อมูลเป็นส่วนที่ใช้ในการติดต่อกับผู้ใช้เพื่อทำการป้อนข้อมูลที่ได้จากการทดสอบ อีกทั้งยังเป็นตัวประมวลผล และเก็บข้อมูลชั่วคราวก่อนที่จะส่งไปยังส่วนแสดงผล และส่วนสร้างรายงาน และด้วยความที่ส่วนรับข้อมูลเป็นส่วนที่ติดต่อกับผู้ใช้ ฉะนั้นจึงจำเป็นที่จะต้องออกแบบทั้งโปรแกรม และฮาร์ดแวร์ให้ใช้งานได้ง่าย และแม่นยำที่สุด

งานในส่วนของตัวรับข้อมูลจะกระทำตั้งแต่ การให้ผู้ใช้เลือกชนิด และ ขนาดของมิเตอร์ ซึ่งชนิดและขนาดของมิเตอร์ดังรายละเอียดที่ได้กล่าวไว้ ตัวรับข้อมูลมีลักษณะดังภาพที่ 3.2.1 ประกอบด้วยองค์ประกอบหลัก 4 ส่วนดังรูปที่ 3.2.2 นั้นคือตัวประมวลผลหลัก, LCM, คีย์บอร์ด 4x4, และ Serial EEPROM โดยมีการสื่อสารกับส่วนแสดงผล และส่วนสร้างรายงานเป็นแบบ SPI และ current loop ตามลำดับ ซึ่งจะได้อธิบายเกี่ยวกับการสื่อสารในภายหลัง ตัวประมวลผลหลักของส่วนรับข้อมูลจะใช้ไอซีเบอร์ 89c51 ซึ่งมีขนาด 40 ขา (รายละเอียดของตัว 89c51 ได้กล่าวไว้แล้วในบทที่สอง) ตัว LCM จะเชื่อมต่อกับตัวประมวลผลหลักแบบ 4 บิท เพื่อใช้เป็นเอาท์พุทที่จะสื่อสารกับผู้ใช้ คีย์บอร์ดขนาด 4x4 จะต่อแบบเมตริก 4x4 และต่อกับตัวประมวลผลหลักแบบ 8 บิท ซึ่งจะมีปุ่มใช้งานได้ทั้งหมด 16 ปุ่ม เรียกได้ว่าเพียงพอกับความต้องการในการใช้งาน

ภาพที่ 3.2.1 ลักษณะของส่วนรับข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

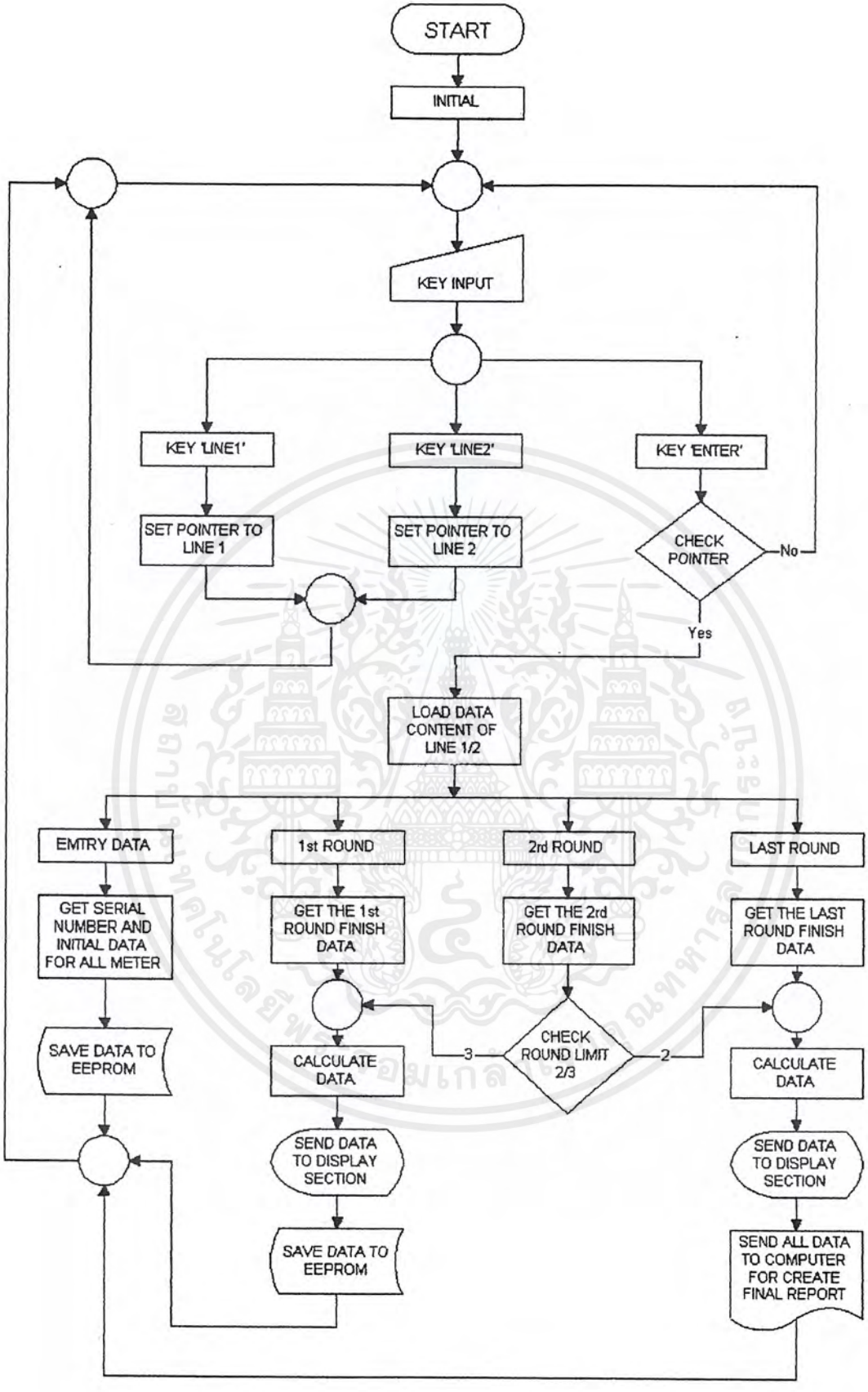
ในการทำการทดสอบจะต้องเก็บข้อมูลประจำตัวของมิเตอร์ที่จะทดสอบ ซึ่งจะประกอบไปด้วยเลขประจำตัว 6 หลัก ปริมาณน้ำเริ่มต้นที่อ่านได้จากมิเตอร์ 6 หลัก ปริมาณสิ้นสุดในการทดสอบสามครั้ง 6x3 หลัก ถ้าคิดประมาณคร่าวๆของข้อมูลมิเตอร์หนึ่งตัว จะต้องใช้หน่วยความจำขนาด 6+6+(6*3) = 30 ไบท์ ซึ่งในการทดสอบหนึ่งครั้งจะมีมิเตอร์ได้สูงสุด 12 ตัว นั้นคือจะต้องใช้ขนาดหน่วยความจำทั้งหมด 30*12 = 360 ไบท์ ในขณะที่ขนาดของแรมในตัวประมวลผลหลักมีเพียง 128 ไบท์ และใช้งานได้เพียง 96 ไบท์ (ตัดแบงค์ 0-3) ส่วนรับข้อมูลจึงจำเป็นที่จะต้องมี Serial EEPROM เพื่อใช้ในการสำรองข้อมูลในขณะทำการทดสอบ เนื่องจากขนาดของแรมในตัวประมวลผลหลักไม่เพียงพอที่จะใช้งาน



ภาพที่ 3.2.2 แสดงองค์ประกอบหลักของวงจรในส่วนรับข้อมูล

ดังจะเห็นได้จากภาพว่า วงจรของส่วนรับข้อมูลนั้นไม่สลับซับซ้อนมากนัก ทั้งนี้เนื่องจากวงจรในส่วนรับข้อมูลจะเป็นส่วนที่สามารถพกพาได้ เพื่อให้สะดวกกับการอ่านค่าข้อมูลจากมิเตอร์น้ำที่ทำการทดสอบ ฉะนั้นส่วนสำคัญของตัวรับข้อมูลจึงอยู่ที่โปรแกรมของมัน ดังภาพที่ 3.2.3 ซึ่งจะแสดงถึงโฟลชาร์ทของโปรแกรมของตัวรับข้อมูล เมื่อเริ่มต้นตัวรับข้อมูลจะทำการตั้งค่าเริ่มต้นและ เข้าสู่เมนูหลักเพื่อให้ผู้ใช้ได้เลือกแถวของมิเตอร์ ในส่วนของการตั้งค่าเริ่มต้นนั้นจะมีการเซ็ทอัพ EEPROM และ LCD โมดูล พร้อมทั้งแสดงข้อความทักทายกับผู้ใช้ เมื่อเข้าสู่เมนูหลักโปรแกรมจะอยู่ในโหมดรอรับคีย์ 'Line1' 'Line2' และ 'Enter' ซึ่งผู้ใช้ต้องทำการเลือกแถวของมิเตอร์ที่จะทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.2.3 แสดงโฟลชาร์ทของโปรแกรมตัวรับข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดสอบเสียก่อน จากที่ทราบมาก่อนแล้วว่า ระบบทดสอบหนึ่งระบบสามารถทำการทดสอบมิเตอร์ได้สองแถวพร้อมกัน และเมื่อผู้ใช้ได้ทำการเลือกแถวของมิเตอร์เรียบร้อยแล้ว โปรมแกรมจะทำการอ่านข้อมูลสถานะเดิมของมิเตอร์ในแถวนั้นๆจาก EEPROM ซึ่งโปรแกรมจะทำการตัดสินใจว่าควรจะทำงานใดต่อไป ในที่นี้สถานะมีอยู่ด้วยกัน 4 สถานะ คือ เริ่มการทดสอบ ทดสอบเสร็จแล้วหนึ่งรอบ ทดสอบเสร็จแล้วสองรอบ และสิ้นการทดสอบรอบสุดท้าย ซึ่งในแต่ละสถานะก็จะมีการทำงานที่คลายคลึง และแตกต่างกันดังนี้

- <u>สถานะเริ่มต้นการทดสอบ</u> จะเกิดขึ้นเมื่อเริ่มต้นการใช้เครื่องครั้งแรก หรือ เมื่อเสร็จสิ้นการทดสอบรอบสุดท้ายและมีการอัพโหลดข้อมูลทั้งหมดเข้าสู่คอมพิเตอร์เรียบร้อยแล้ว การทำงานในสถานะนี้จะให้ผู้ใช้ได้เลือกชนิด และขนาดของมิเตอร์ ซึ่งจะใช้เป็นข้อมูลอ้างอิงกับมิเตอร์ในแถวดังกล่าวไปตลอดจนจบการทดสอบ จากนั้นจึงทำการรับข้อมูลเลขประจำตัว และค่าเริ่มต้นที่อ่านได้จากหน้าปัดของมิเตอร์ ซึ่งจะต้องใส่ให้ครบตามจำนวนมิเตอร์ที่ทำการทดสอบ (จำนวนมิเตอร์ที่ทำการทดสอบในแต่ละแถวจะขึ้นอยู่กับขนาดของมิเตอร์ดังตารางที่ 3.1.4) จากนั้นโปรแกรมจะทำการบันทึกข้อมูลที่ได้รับทั้งหมดลงสู่ EEPROM และกลับสู่เมนูหลัก

- <u>สถานะจบการทดสอบรอบที่หนึ่ง</u> เมื่อออกจากสถานะเริ่มต้นการทดสอบแล้ว ข้อมูลสถานะจะถูกเปลี่ยนไปสู่สถานะจบการทดสอบรอบที่หนึ่ง ซึ่งในสถานะนี้จะมีการทำงานดังนี้ คือ รับค่าที่อ่านได้จากมิเตอร์หลังจากสิ้นสุดการทดสอบรอบแรก แล้วนำข้อมูลดังกล่าวไปคำนวณหาค่าเปอร์เซ็นต์ความคลาดเคลื่อน จากนั้นจึงส่งผลการคำนวณที่ได้ไปยังส่วนแสดงผล เพื่อทำการแสดงผลการทดสอบในรอบดังกล่าว สิ้นสุดด้วยการบันทึกข้อมูลต่างๆกลับสู่ EEPROM เพื่อใช้ในการคำนวณในรอบต่อไป และกลับสู่เมนูหลัก

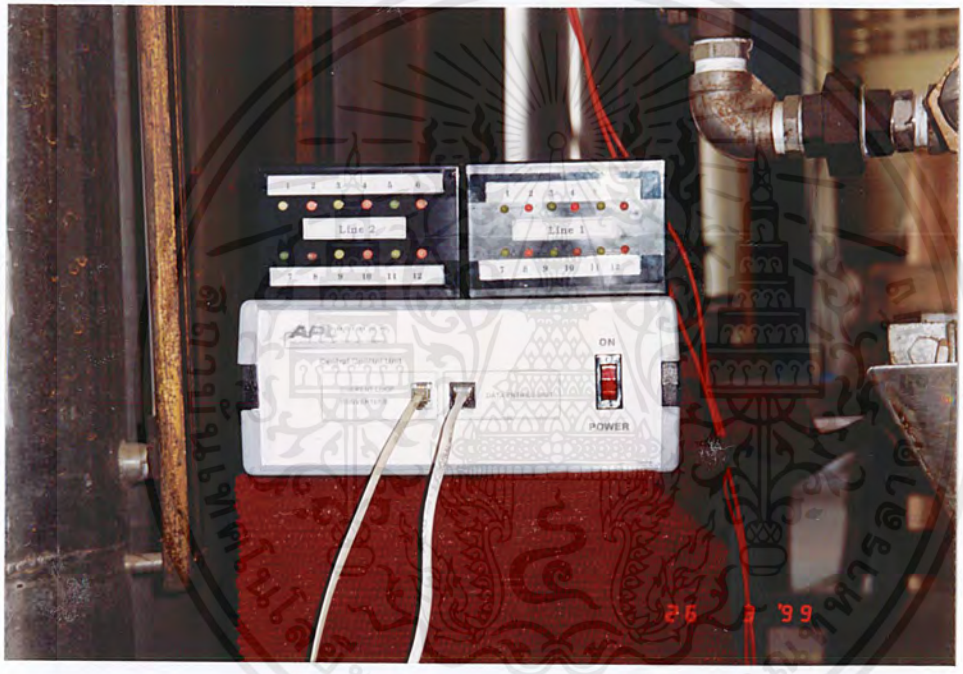
- <u>สถานะจบการทดสอบรอบที่สอง</u> เช่นเดียวกับในสถานะจบการทดสอบรอบที่หนึ่ง แต่ต่างกันตรงที่ ในรอบที่สองจะมีการตรวจสอบชนิดของมิเตอร์ (ว่าอยู่ในกลุ่มใดดังตารางที่ 3.1.1-3) ซึ่งชนิดของมิเตอร์จะเป็นตัวกำหนดจำนวนรอบที่ทำการทดสอบ ถ้าเป็นกลุ่มที่ 1 หรือ 2 ก็จะมีการทดสอบสามรอบ และมีกระบวนการทำงานเหมือนกับในการจบการทดสอบรอบแรก แต่ถ้าเป็นกลุ่มที่ 3 จะมีการทดสอบสองรอบ และมีการทำงานแตกต่างไปคือ เมื่อรับข้อมูล คำนวณ และแสดงผลเสร็จแล้ว จะทำการอัพโหลดข้อมูลไปยังคอมพิวเตอร์เพื่อสร้างรายงานสุดท้าย ก่อนที่จะกลับสู่เมนูหลัก

- <u>สถานะจบการทดสอบรอบสุดท้าย</u> ในมิเตอร์กลุ่มที่ 1 หรือ 2 จะมีจำนวนรอบการทดสอบสามรอบ ซึ่งจะมีแต่มิเตอร์ในกลุ่มนี้เท่านั้นที่จะมีสถานะสุดท้ายนี้ได้ การทำงานของโปรแกรมในสถานะนี้ก็จะคล้ายกับสถานะที่สอง แต่ต่างกันตรงที่ เมื่อข้อมูลถูกส่งไปแสดงผลเสร็จแล้ว โปรแกรมจะทำการส่งข้อมูลไปยังคอมพิวเตอร์เพื่อทำการสร้างรายงานสุดท้ายทันที และจะเปลี่ยนข้อมูลสถานะเป็นข้อมูลเริ่มต้นก่อนที่จะกลับสู่เมนูหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 ส่วนแสดงผล

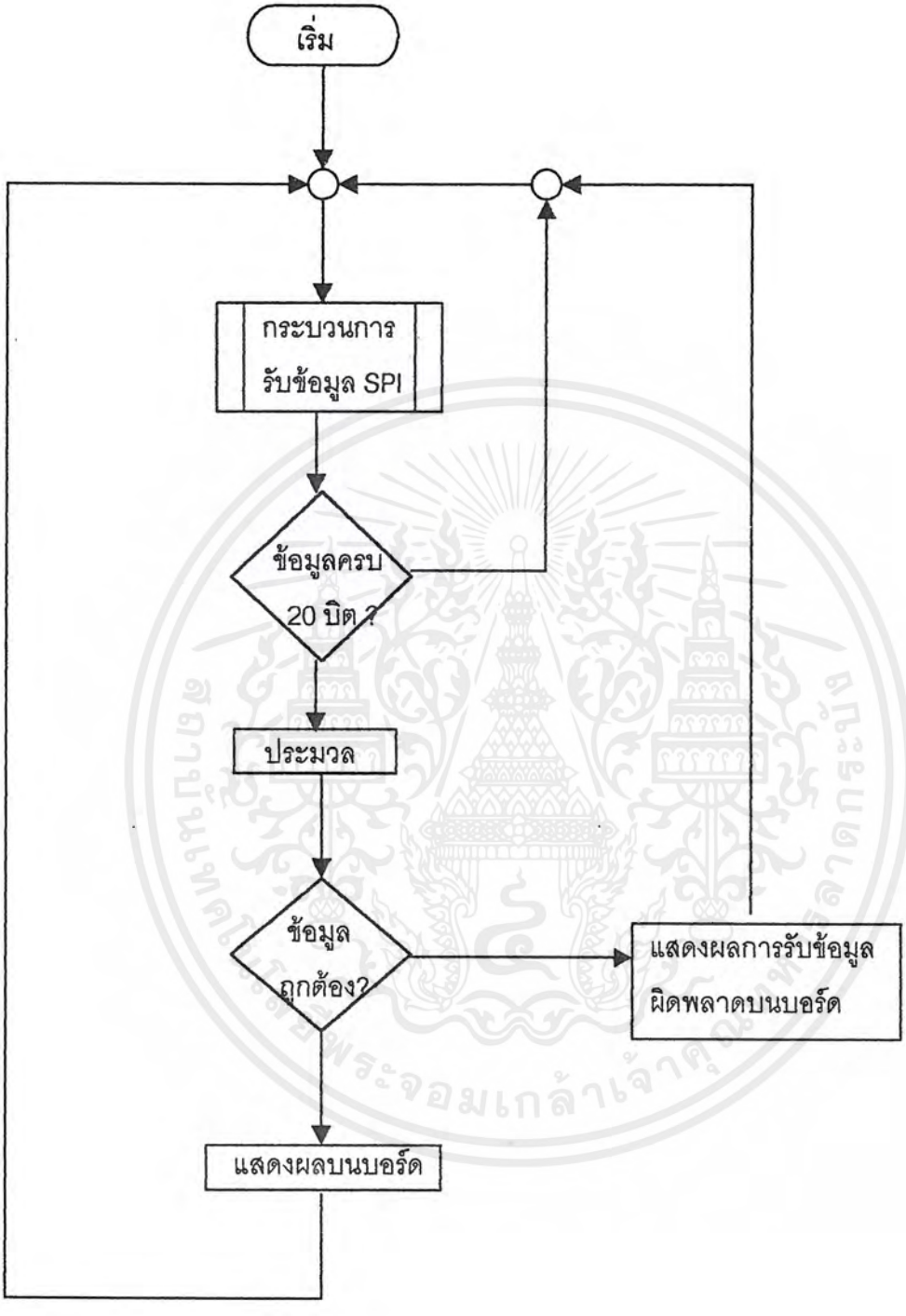
เมื่อผู้บันทึกข้อมูลได้บันทึกข้อมูลเสร็จสิ้นแล้ว ผลการทดสอบจะถูกส่งจากส่วนรับข้อมูลมายังส่วนแสดงผล ใช้สายสัญญาณในการสื่อสาร 3 เส้น คือ SCLK ,DATA และ STRB โดยจะสื่อสารข้อมูลผ่านทางสาย UTP ข้อมูลที่ถูกส่งมาจะถูกแปลความหมายโดยส่วนถอดรหัสข้อมูล จากนั้นจะถูกส่งผ่านวงจรแลทช์ข้อมูลไปแสดงผลยังบอร์ดแสดงผล ทำให้ผู้ทดสอบสามารถทราบได้ทันทีว่ามิเตอร์น้ำตัวใดที่ผ่าน หรือไม่ผ่านการทดสอบ โดยมองจากบอร์ดแสดงผลซึ่งจะแสดงสถานะของมิเตอร์แต่ละตัวด้วยไฟสี ถ้าตำแหน่งใดถูกแทนด้วยไฟสีแดงคือไม่ผ่าน ส่วนไฟสีเขียวคือผ่านการทดสอบ จากนั้นผู้ทำการทดสอบจะต้องทำเครื่องหมายมิเตอร์น้ำที่มีความผิดพลาดเกินค่ายอมรับได้ไว้ ลักษณะของส่วนแสดงผลแสดงดังภาพ 3.3.1



ภาพ 3.3.1 ลักษณะของส่วนแสดงผล

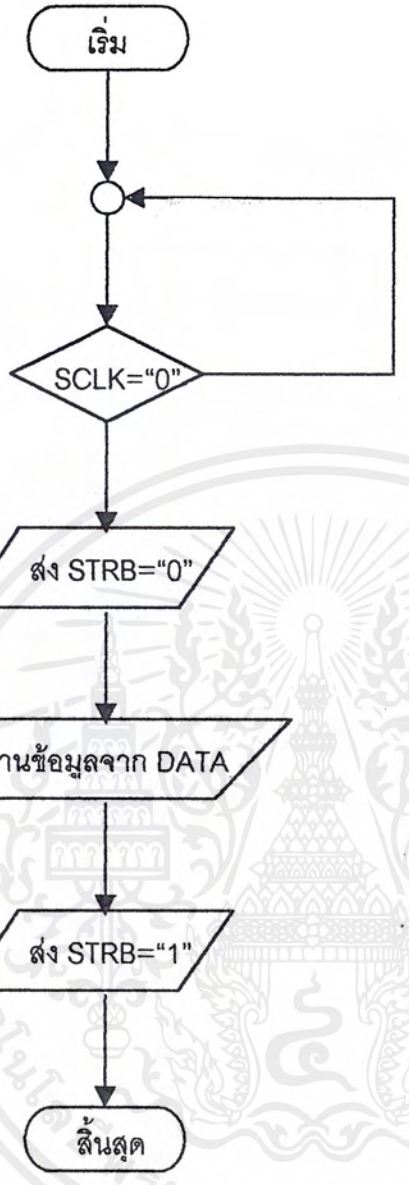
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนการทำงานในส่วนแสดงผล แสดงดังภาพ 3.3.2



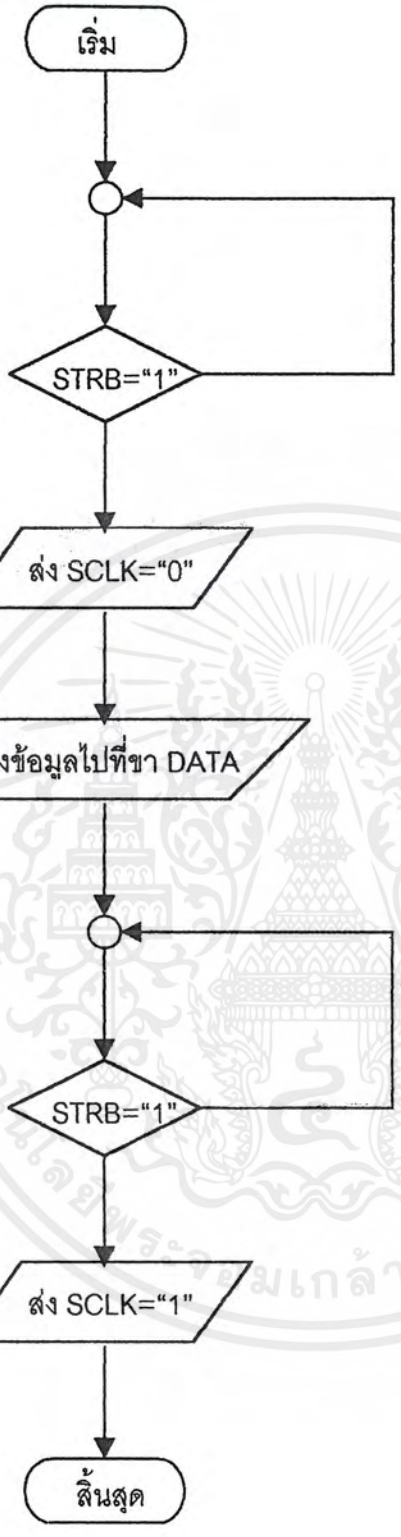
ภาพ 3.3.2 แสดงขั้นตอนการทำงานในส่วนแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพ 3.3.3 แสดงขั้นตอนการรับข้อมูลแบบ SPI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพ 3.3.4 แสดงการส่งข้อมูลแบบ SPI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยส่วนแสดงผลนี้ สามารถแบ่งการทำงานหลักๆได้ 3 ส่วนคือ

3.3.1ส่วนถอดรหัสข้อมูล

จะใช้ไมโครคอนโทรลเลอร์ 89C51 ของ ATMEL ในการควบคุมการทำงาน โดยจะถอดรหัสข้อมูลที่ส่งมาทางสายสัญญาณทั้ง 3 เส้นที่กล่าวถึง โดยส่วนถอดรหัสจะเช็คขา SCLK ว่ามีสถานะเป็น low หรือไม่ หากเป็น high แสดงว่ายังไม่มีการส่งข้อมูล เมื่อตัว Data Entry จะทำการส่งข้อมูลจะส่งข้อมูลทางขา Data และคงสถานะข้อมูลไว้ พร้อมๆกับดึงสัญญาณขา SCLK ให้เป็น low เมื่อส่วนถอดรหัสพบว่า SCLK เป็น low ก็จะทำการอ่านค่าข้อมูลทางขา Data มาเก็บไว้ จากนั้นจะดึงขา STRB ให้เป็น low เพื่อให้ Data Entry ทราบว่ามีการอ่านข้อมูลในบิตนั้นเสร็จแล้ว โดย Data Entry ก็จะดึงสถานะของขา SCLK ให้เป็น high ดังเดิม จากนั้น ส่วนถอดรหัสก็จะทำการเช็คสถานะของขา SCLK ต่อไปว่ามีการส่งข้อมูลมาอีกหรือไม่

ชุดข้อมูลที่ทำการส่งในแต่ละครั้งนั้น มี 20 บิต โดยจำแนกออกได้ดังนี้

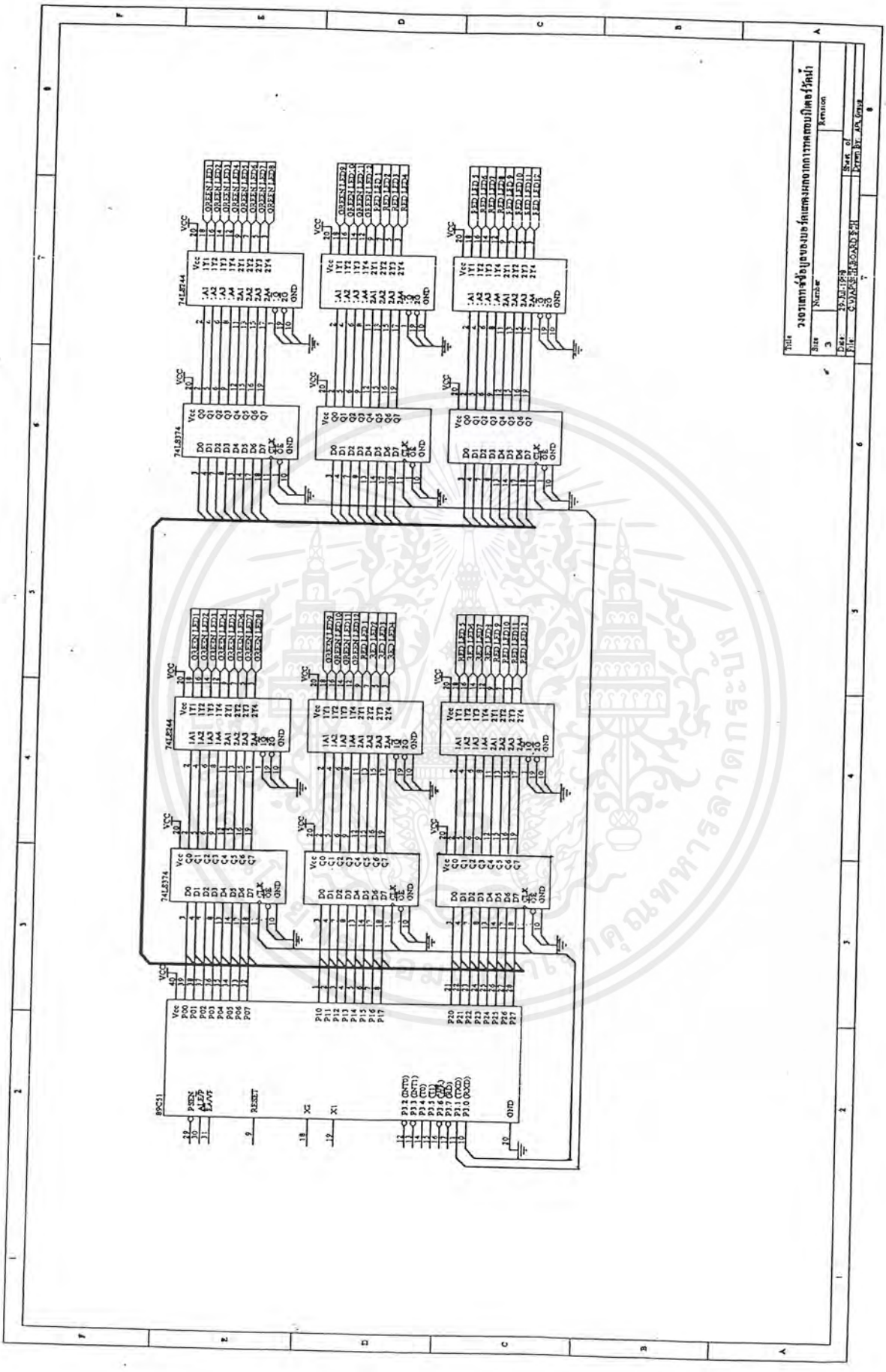
- 8 บิตแรกจะบอกว่าข้อมูลชุดนี้เป็น ผลการทดสอบจาก line1 หรือ line2 และบอกว่าเป็นการทดสอบมิเตอร์จำนวนกี่ตัว ซึ่งค่าที่ส่งมาจะมีรายละเอียดดังตาราง 3.3.1

ตาราง 3.3.1 ลักษณะของชุดข้อมูล 8 บิตแรกที่ส่งทาง SPI

ค่าของข้อมูลใน 8 บิตแรก	จำนวนมิเตอร์ที่ทำทดสอบ	Line ที่ทำการทดสอบ
09H	8	1
0AH	8	2
0BH	10	1
0CH	10	2
0DH	12	1
0EH	12	2

- 12 บิตต่อมาจะแทนสถานะของมิเตอร์ ที่มีจำนวนสูงสุด 12 ตัว โดยบิตที่มีค่า 0 จะแสดงว่ามิเตอร์ตัวนั้นไม่ผ่านการทดสอบ ส่วนบิตที่มีค่า 1 จะแสดงว่ามิเตอร์ตัวนั้นผ่านการทดสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



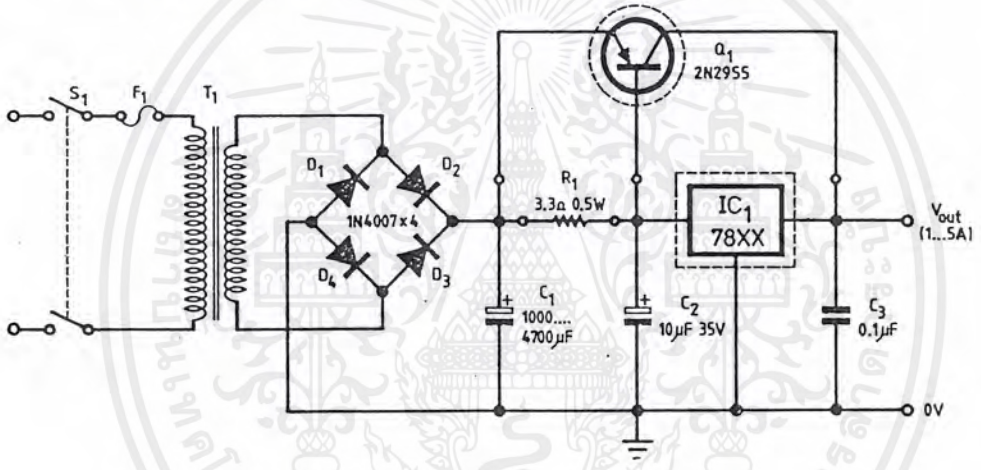
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2ส่วนแลทช์ข้อมูลและบอร์ดแสดงผล

หลังจากที่ข้อมูลได้ถูกถอดรหัสโดยไมโครคอนโทรลเลอร์ในส่วนถอดรหัสข้อมูลแล้ว จะทำการส่งค่าที่ต้องการแสดงผลออกทาง พอร์ต 0 พอร์ต 1 และพอร์ต 2 มายังวงจรแลทช์ข้อมูลโดยไอซีแลทช์ข้อมูล 74LS374 และจะส่งผ่านสถานะข้อมูลไปยังไอซีบัฟเฟอร์ 74LS244 เพื่อขับหลอดแสดงผล LED 2 สี ใน Display Box โดยมีวงจรแลทช์ ข้อมูลดังภาพ 3.3.5

3.3.3ส่วนจ่ายไฟ

ภาคจ่ายไฟที่ติดตั้งอยู่ในส่วนแสดงผลนี้ นอกจากจะทำหน้าที่จ่ายกระแสไฟฟ้าให้กับส่วนถอดรหัสข้อมูล และส่วนรับข้อมูลแล้ว ยังจะต้องจ่ายกระแสให้กับหลอดแสดงผลทั้ง 24 ดวงด้วย โดยจะส่งกระแสไฟฟ้าไปให้กับส่วนรับข้อมูลทางสาย UTP ดังนั้นจึงต้องออกแบบภาคจ่ายไฟให้รองรับการจ่ายกระแสสูงๆได้ ซึ่งจะใช้ทรานซิสเตอร์ร่วมในวงจรเรกูเลเตอร์ดังภาพ



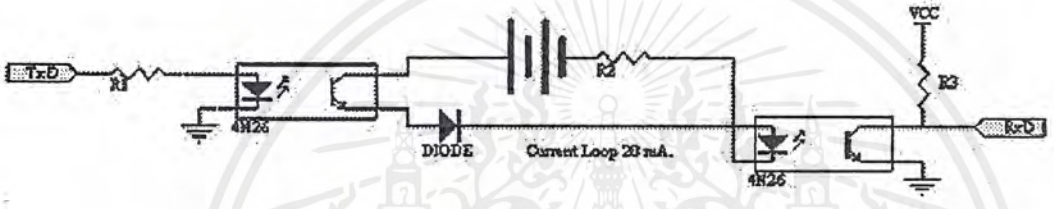
ภาพ 3.3.6 วงจรจ่ายไฟแบบมีทรานซิสเตอร์ช่วยจ่ายกระแส

จากวงจร เมื่อโหลดใช้กระแสเกิน 200 มิลลิแอมป์ ทรานซิสเตอร์ก็จะทำหน้าที่เป็นตัวผ่านกระแสที่เกินนั้นแทนเรกูเลเตอร์ และสามารถจ่ายได้ถึง 5 แอมป์ โดย R1 ทำหน้าที่ป้องกันกระแสเกินให้กับเรกูเลเตอร์เป็นจุดแรก ขณะเดียวกันก็จะทำหน้าที่ป้อนแรงดันไบอัสให้ทรานซิสเตอร์ด้วย ขนาดอัตราทนกำลังของ R1 คำนวณได้จากค่ากระแสที่ผ่านตัวมันสูงสุด โดยจะอยู่ในอัตรา 0.5-5 วัตต์ ส่วนตัวเก็บประจุ C1 นั้นเป็นตัวกำจัดริปเปิลของแรงดันด้านอินพุต หากวงจรมีการใช้กระแสไม่เกิน 1 แอมป์ ใช้ค่าเพียง 1000 ไมโครฟารัด ก็น่าจะพอ แต่หากจำเป็นต้องใช้กระแสเอาท์พุตสูงถึง 5 แอมป์ ก็ควรเพิ่มค่าความจุเป็น 4700 ไมโครฟารัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 Current Loop Converter

เนื่องจากการสื่อสารข้อมูลระหว่างไมโครคอนโทรลเลอร์ในส่วนรับข้อมูลกับเครื่องคอมพิวเตอร์ที่ใช้สำหรับพิมพ์รายงานนั้น จะอยู่ห่างไกลกันมาก ในการทำงานจริงนั้นผู้บันทึกข้อมูลและส่วนรับข้อมูลจะต้องอยู่ที่สถานีทดสอบมิเตอร์ แต่ส่วนพิมพ์รายงานนั้นอาจจะตั้งอยู่ในสำนักงานที่อยู่ห่างออกไปเป็นระยะทาง 200-300 เมตรขึ้นไปก็ได้ เราจะใช้การสื่อสารแบบ RS232 ไม่ได้เพราะจำกัดระยะทางอยู่ไม่กี่ฟุต ดังนั้นจึงต้องใช้วิธี Current Loop เข้าช่วยโดยมีวงจรดังภาพที่ 3.4.1 ซึ่งเป็นการส่งทิศทางเดียว และเป็นแบบที่ใช้งานในโครงงานนี้ โดยเป็นการส่งจากไมโครคอนโทรลเลอร์(ด้านซ้าย)ไปยังคอมพิวเตอร์(ด้านขวา)



ภาพ 3.4.1 แสดงวงจร Current Loop

3.5 ส่วนสร้างรายงาน

เป็นโปรแกรมที่ทำงานบนระบบปฏิบัติการวินโดว์ 95 โดยเขียนด้วยภาษาวิชวลเบสิค ซึ่งโปรแกรมจะมีความสามารถดังนี้

- สามารถแสดงผลทางหน้าจอ และพิมพ์รายงานได้ทันที
- สามารถเปลี่ยนรูปแบบข้อมูลไปเป็นรูปแบบอื่นได้มากมาย เช่น Excel , HTML ,Text ,etc.
- เลือกช่องทางสื่อสารและความเร็วในการรับส่งได้ เพื่อความยืดหยุ่นในการเพิ่มระยะทางสื่อสาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

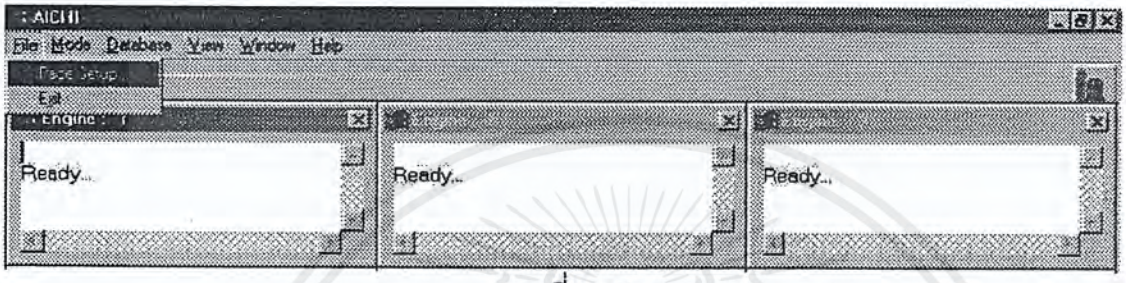
เริ่มต้นใช้งานโปรแกรม

เปิดโปรแกรมชื่อ Aichi โดยกดปุ่ม Start ➔ Programs ➔ Aichi จะปรากฏหน้าต่างของโปรแกรมดังภาพที่ 3.5.1

File

- เลือก Page Setup สำหรับการเซตคุณสมบัติของพรินเตอร์และรูปแบบการพิมพ์

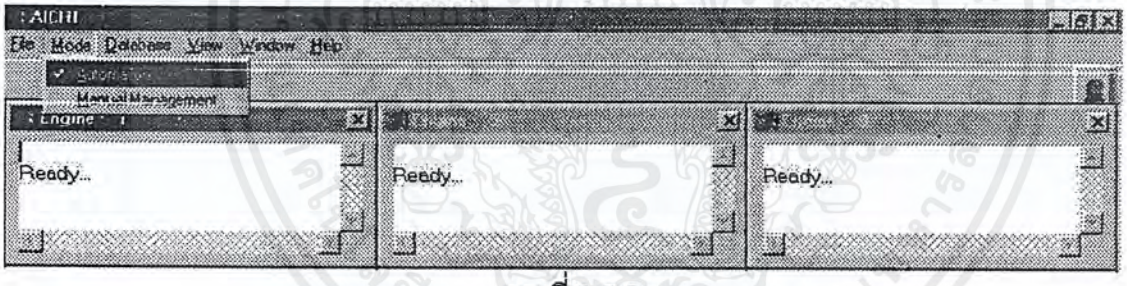
เลือก Exit เพื่อออกจากโปรแกรม



ภาพที่ 3.5.1

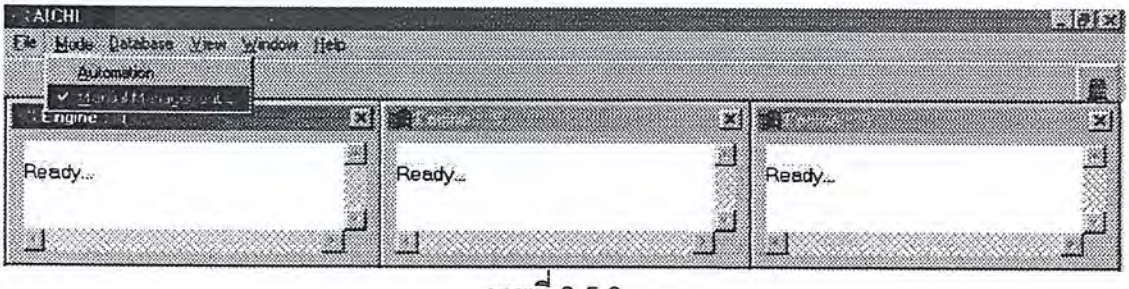
Mode

- เลือก Automatic เมื่อต้องการเริ่มต้นการรับข้อมูล



ภาพที่ 3.5.2

- เลือก Manual เพื่อยินยอมให้มีการเซตค่าค่างๆของตัวโปรแกรมได้

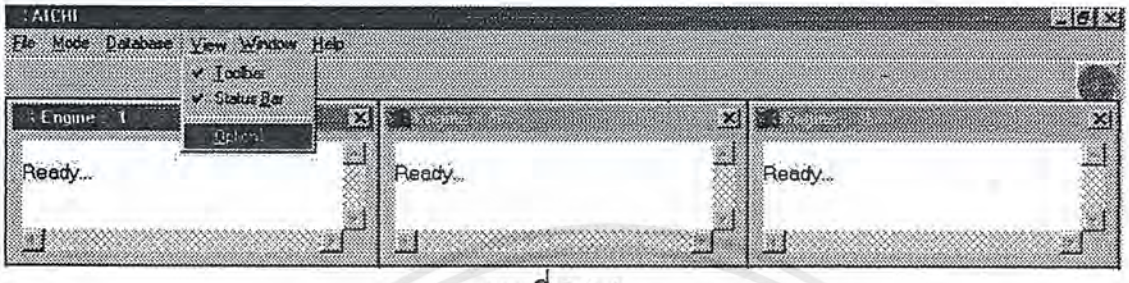


ภาพที่ 3.5.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

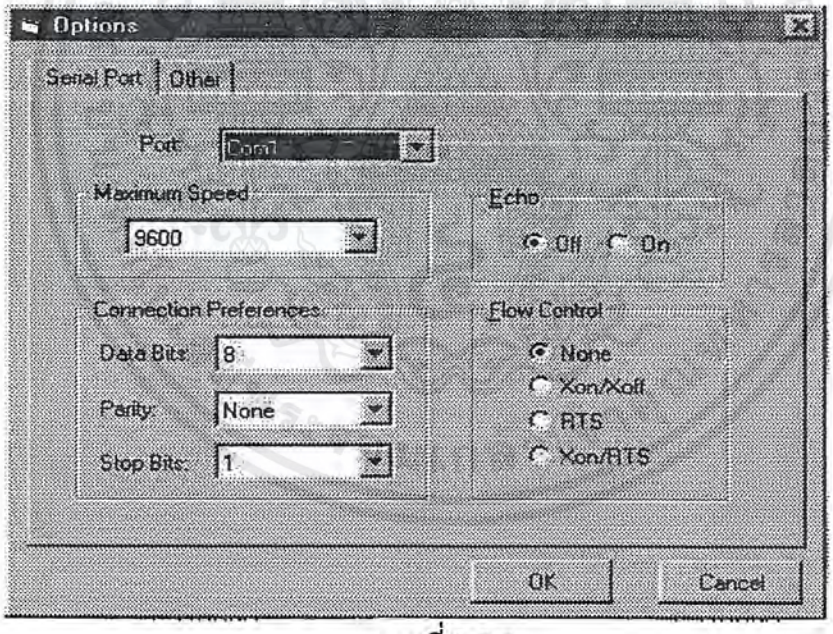
View

- Options ก่อนการใช้งาน จะต้องตั้งค่าต่างๆที่เกี่ยวข้องกับการทำงานของโปรแกรมให้ถูกต้อง ซึ่งจะแบ่งเป็น 2 หมวดคือ การตั้งค่าเกี่ยวกับการสื่อสารข้อมูล และการตั้งค่าที่เกี่ยวกับรูปแบบการแสดงผล โดยเริ่มต้นด้วยการเลือกที่ View ➔ Options... ดังภาพ



ภาพที่ 3.5.4

สำหรับการตั้งค่าที่เกี่ยวข้องกับการสื่อสารข้อมูลให้เหมือนดังภาพ ยกเว้นการตั้ง PORT : ให้เลือก Com1 (DB9) หรือ Com2 (DB25) ตามการเชื่อมต่อจริง



ภาพที่ 3.5.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

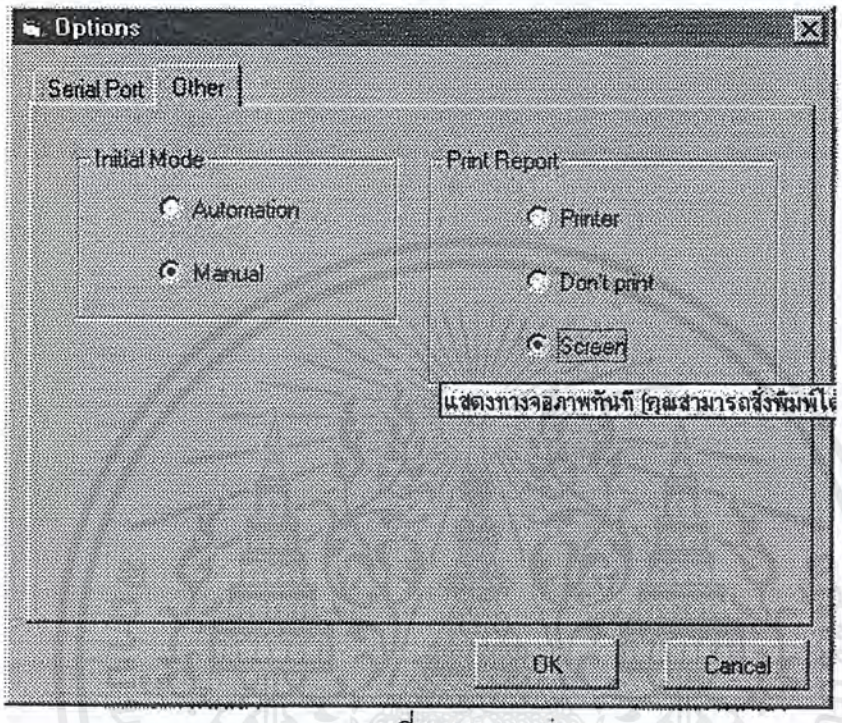
ส่วนหมวดของการตั้งค่าที่เกี่ยวข้องกับการแสดงผล มีรายละเอียดดังนี้คือ

Initial Mode : เป็นการกำหนดโหมดการทำงาน ในตอนเริ่มต้นการใช้โปรแกรม

Print Report : Printer ➔ พิมพ์ออกเครื่องพิมพ์ทันที

Don't print ➔ ไม่ต้องพิมพ์รายงาน

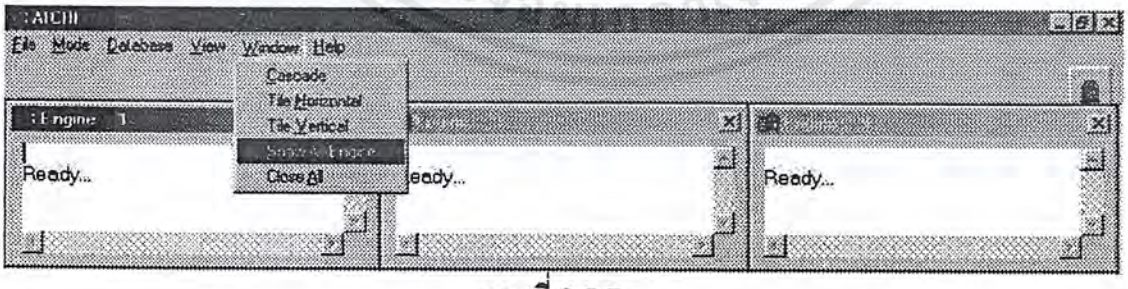
Screen ➔ แสดงทางจอภาพทันที และสามารถสั่งพิมพ์ได้



ภาพที่ 3.5.6

Windows

- เป็นการเลือกรูปแบบในการแสดงผลทางจอภาพ แบบต่างๆ



ภาพที่ 3.5.7

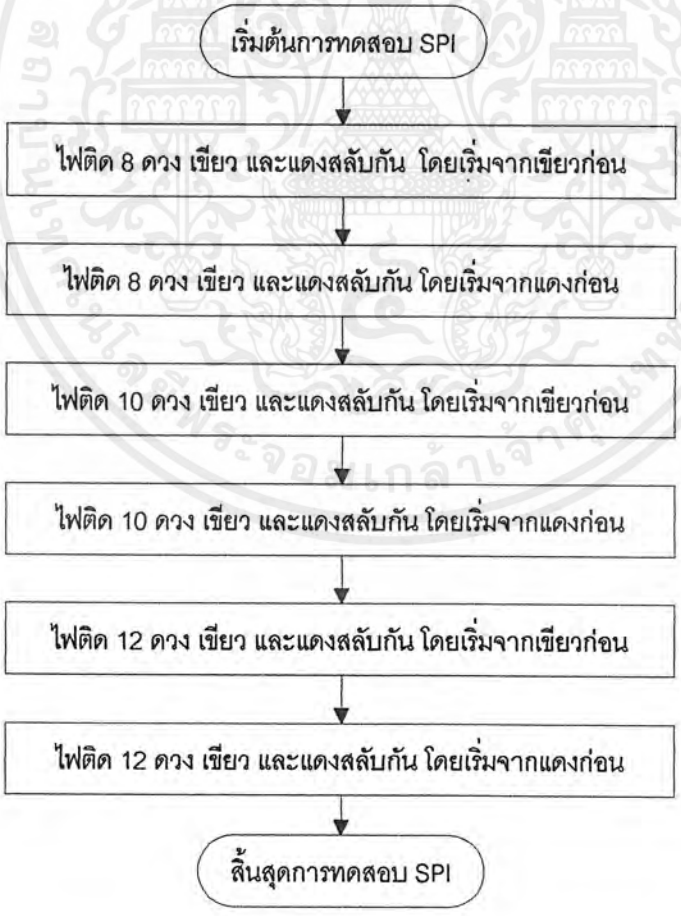
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
ผลการทดลอง

ในการทดลองใช้งานไมโครเทอร์มินอลสำหรับตรวจสอบมิเตอร์น้ำนั้นได้แบ่งการทดสอบย่อยๆออกเป็นสามส่วนคือ ทดสอบการส่งข้อมูลแบบ SPI ทดสอบการส่งข้อมูลแบบ current loop และ ทดสอบเพื่อทำการเปรียบเทียบผลการทดสอบในระบบเก่า และระบบใหม่

4.1 ทดสอบการส่งข้อมูลแบบ SPI

ในการทดสอบการส่งข้อมูลแบบ SPI นั้นจะทำการทดลองส่งข้อมูลระหว่างไมโครคอนโทลเลอร์สองตัว คือ ไมโครคอนโทลเลอร์ที่อยู่กับตัวรับข้อมูล(Data Entry Unit) และ ไมโครคอนโทลเลอร์ที่อยู่กับภาคแสดงผล โดยในการทดสอบนั้นจะทำการส่งข้อมูลจากตัว Data entry ไปยังส่วนแสดงผลเพื่อให้ส่วนแสดงผลทำการแสดงผลข้อมูลดังกล่าว ข้อมูลที่ทำการแสดงผลนั้นจะเป็นข้อมูลที่กำหนดให้ไฟแสดงผลการทดสอบมิเตอร์ทั้งสองแถว ติดสว่างเป็นสีเขียว และแดง สลับกันไป เริ่มต้นจากแปดดวง สิบดวง และสิบสองดวงตามลำดับดังนี้

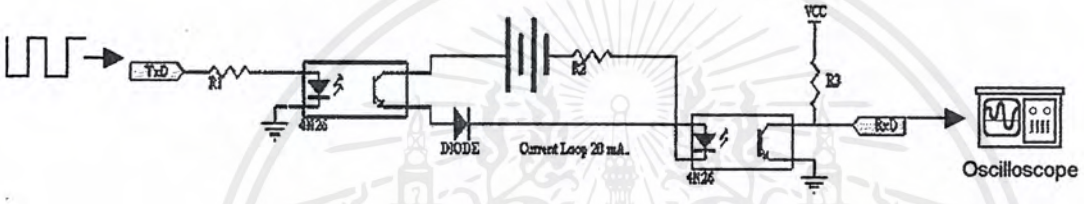


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการทดสอบข้างต้น พบว่าสามารถใช้สายไฟ(ในการรับส่งข้อมูล) ได้ยาวถึง 4 เมตร โดยข้อมูลไม่มีการผิดเพี้ยน ซึ่งสามารถสังเกตุได้จากไฟที่แสดงบนบอร์ดแสดงผล

4.2 ทดสอบการส่งข้อมูลแบบ current loop

ในการทดสอบการส่งข้อมูลแบบ current loop ในจะใช้การทดสอบความเพี้ยนของสัญญาณ ซึ่งอาจเกิดจากความยาวของสายส่งสัญญาณ และตัวววงจร current loop เอง การทดสอบนั้นจะทำการต่อวงจร current loop แล้วป้อนคลื่นสัญญาณรูปสี่เหลี่ยม (square wave) เข้าไปทางด้านหนึ่งของวงจร current loop จากนั้นจับสัญญาณเอาท์พุทที่อยู่อีกด้านหนึ่งของวงจร เพื่อทำการเปรียบเทียบรูปร่างของสัญญาณทั้งสอง ดังรูป



ภาพที่ 4.2.1 แสดงการทดสอบวงจร current loop

ภาพที่ 4.2.2 ผลการทดสอบที่วัดได้จาก Oscilloscope

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการทดลองพบว่าเราสามารถใช้อัตราเร็วในการส่งข้อมูล 19,200 บิทต่อวินาที กับสายสัญญาณยาว 30 เมตรได้ โดยมีความผิดเพี้ยนของสัญญาณน้อยมาก ถึงกระนั้นการใช้ความเร็วในการส่งข้อมูล ที่ต่ำลง(9,600 บิทต่อวินาที) จะช่วยให้มีสเถียรภาพในการส่งขอ้มูลที่สูงกว่า และให้ความถูกต้องของข้อมูลมากกว่า

4.3 ทดสอบเพื่อเปรียบเทียบความถูกต้องในระบบเก่า และระบบใหม่

จุดประสงค์หลักของการทำโครงงานชิ้นนี้คือ การสร้างระบบการตรวจสอบมิเตอร์น้ำระบบใหม่ ให้มีความถูกต้องและน่าเชื่อถือมากกว่าระบบเดิม ดังนั้นก่อนการทำการเปรียบเทียบเราจำเป็นที่จะต้องทราบอย่างคราวๆก่อนว่าระบบเดิมนั้นมีการทำงานเป็นอย่างไร

การตรวจสอบในระบบเดิมนั้น อาศัยผู้ทำการทดสอบในการจดบันทึกข้อมูลทั้งหมดที่อ่านได้จากมิเตอร์ และทำการคำนวณด้วยเครื่องคิดเลข จากนั้นจึงทำการระบุว่ามิเตอร์เครื่องใดมีความความคลาดเคลื่อนเกินกว่ามาตรฐาน หลังจากนั้นจึงให้ผู้ทำการทดสอบบันทึกข้อมูลสุดท้ายลงยังรายงานผลการทดสอบ ซึ่งจะเห็นว่ากระบวนการในระบบเก่าที่กล่าวมาทั้งหมดแล้วนั้นสามารถก่อให้เกิดความคลาดเคลื่อนได้ทุกขั้นตอน ฉะนั้นระบบที่ได้รับการปรับปรุงจึงจำเป็นที่จะต้องแก้ไขความผิดพราดดังกล่าวให้มากที่สุด

ตารางที่ 4.3.1 ผลการเปรียบเทียบระบบเก่า และ ระบบที่ได้รับการพัฒนา

	ระบบเก่า	ระบบใหม่
ความละเอียดของข้อมูล	ทศนิยมหนึ่งหลัก	ทศนิยมสองหลัก
ระบบการคำนวณ	ผู้ทำการทดสอบใช้เครื่องคิดเลขในการคำนวณ	เครื่องบันทึกข้อมูลทำการคำนวณด้วยระบบคอมพิวเตอร์โดยอัตโนมัติ
การทำรายงาน	เขียนด้วยมือหลังจากที่รวบรวมข้อมูลเสร็จ	สร้างรายงานด้วยคอมพิวเตอร์โดยอัตโนมัติ หลังจากที่มีข้อมูลครบ
การจดบันทึก	จดบันทึกด้วยมือ	จดบันทึกด้วยระบบคอมพิวเตอร์
ความสะดวก	ผู้ทำการทดสอบต้องหาตารางจดบันทึกข้อมูลให้ตรงกับชนิดของมิเตอร์	ผู้ทำการทดสอบสามารถเลือกชนิดของมิเตอร์ได้จากเครื่องบันทึกข้อมูล

ผลการทดสอบระบบการตรวจวัดมิเตอร์ทั้งสองระบบ สามารถเปรียบเทียบด้วยรายงานสุดท้ายที่ถูกสร้างขึ้นจากทั้งสองระบบ โดยข้อมูลที่ใช้จะเป็นข้อมูลที่ได้จากการทดสอบมิเตอร์น้ำจริง ซึ่งใช้ทั้งระบบใหม่และระบบเก่าในการจดบันทึก จะได้ผลดังข้อมูลดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

THAI AICHI CO.,LTD.
ACCURACY TEST REPORT

Model : PXS Inspector : ROOT

Size : 15 mm. Date : 29/3/99

Serial No.	Flow rate (Lt./hr.)	Quantity (Lt.)	Reading			Percentage Error		Remark
			Before	After	Difference	+	-	
001561	1,500	100	68.71	168.20	99.49		0.51	
	120	20		187.95	19.75		1.25	
	30	20		208.52	20.57	2.85		
001562	1,500	100	158.11	259.30	101.19	1.19		
	120	20		279.60	20.30	1.50		
	30	20		300.50	20.90	4.50		
001563	1,500	100	141.56	242.33	100.77	0.77		
	120	20		261.23	18.90		5.50	
	30	20		279.73	18.50		7.50	
001564	1,500	100	160.29	257.56	97.27		2.73	
	120	20		277.07	19.51		2.45	
	30	20		298.17	21.10	5.50		
001565	1,500	100	157.18	256.13	98.95		1.05	
	120	20		276.03	19.90		0.50	
	30	20		296.61	20.58	2.90		
001566	1,500	100	152.69	253.90	101.21	1.21		
	120	20		274.03	20.13	0.65		
	30	20		296.45	22.42	12.10		
001567	1,500	100	158.39	257.40	99.01		0.99	
	120	20		277.08	19.68		1.60	
	30	20		296.86	19.78		1.10	
001568	1,500	100	194.40	293.26	98.86		1.14	
	120	20		313.04	19.78		1.10	
	30	20		332.21	19.17		4.15	
001569	1,500	100	164.58	263.30	98.72		1.28	
	120	20		283.09	19.79		1.05	
	30	20		304.24	21.15	5.75		
001570	1,500	100	165.13	265.46	100.33	0.33		
	120	20		285.57	20.11	0.55		
	30	20		305.08	19.51		2.45	
001571	1,500	100	68.71	168.20	99.49		0.51	
	120	20		187.95	19.75		1.25	
	30	20		208.52	20.57	2.85		
001572	1,500	100	158.11	259.30	101.19	1.19		
	120	20		279.60	20.30	1.50		
	30	20		300.50	20.90	4.50		

Certified By : ..... Approved By : .....

Date : ..... Date : .....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

THAI AICHI CO.,LTD.
ACCURACY TEST REPORT

Model : PXS Inspector : ROOT

Size : 25 mm. Date : 29/3/99

Serial No.	Flow rate (Lt./hr.)	Quantity (Lt.)	Reading			Percentage Error		Remarks
			Before	After	Difference	+	-	
001561	3,500	100	145.62	249.32	103.70	3.70		
	280	50		299.21	49.89		0.22	
	70	20		319.00	19.79		1.05	
001562	3,500	100	158.11	259.30	101.19	1.19		
	280	50		309.21	49.91		0.18	
	70	20		329.90	20.69	3.45		
001563	3,500	100	141.56	242.33	100.77	0.77		
	280	50		291.30	48.97		2.06	
	70	20		311.56	20.26	1.30		
001564	3,500	100	160.29	257.56	97.27		2.73	
	280	50		307.76	50.20	0.40		
	70	20		327.98	20.22	1.10		
001565	3,500	100	157.18	256.13	98.95		1.05	
	280	50		306.12	49.99		0.02	
	70	20		324.40	18.28		8.60	
001566	3,500	100	152.69	253.90	101.21	1.21		
	280	50		305.40	51.50	3.00		
	70	20		325.90	20.50	2.50		
001567	3,500	100	158.39	257.40	99.01		0.99	
	280	50		307.39	49.99		0.02	
	70	20		328.48	21.09	5.45		
001568	3,500	100	194.40	293.26	98.86		1.14	
	280	50		343.30	50.04	0.08		
	70	20		364.18	20.88	4.40		

Certified By : Approved By :

Date : Date :

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการสุ่มทดสอบมาตรวัดน้ำ THAI AICHI ขนาด 1/2 นิ้ว

ทดสอบวันที่ 29/3/99

ลำดับที่	หมายเลข	ตัวเลขใน 1,500 ลิตร/ชม. ปริมาณน้ำ 200 ลิตร			% ผิดพลาด	ตัวเลขใน 120 ลิตร/ชม. ปริมาณน้ำ 50 ลิตร			% ผิดพลาด	ตัวเลขใน 30 ลิตร/ชม. ปริมาณน้ำ 20 ลิตร			% ผิดพลาด
		ก่อนทดสอบ	หลังทดสอบ	ได้น้ำ		ก่อนทดสอบ	หลังทดสอบ	ได้น้ำ		ก่อนทดสอบ	หลังทดสอบ	ได้น้ำ	
1	001561	68.71	268.09	199.38	-0.31		318.15	50.06	0.12		338.10	19.95	-0.2
2	001562	58.11	259.30	201.19	0.59		309.27	49.97	-0.06		329.21	19.94	-0.3
3	001563	41.56	242.33	200.77	0.39		291.23	48.90	-2.20		311.73	20.50	2.5
4	001564	60.29	257.56	197.27	-1.36		307.07	49.51	-0.98		327.17	20.10	0.5
5	001565	57.18	256.13	198.95	-0.53		306.03	49.90	-0.20		326.61	20.58	2.9
6	001566	52.69	253.90	201.21	0.61		304.03	50.13	0.26		323.45	19.42	-2.9
7	001567	58.39	257.40	199.01	-0.50		307.08	49.68	-0.64		326.86	19.78	-1.1
8	001568	94.40	293.26	198.86	-0.57		343.04	49.78	-0.44		362.21	19.17	-4.1
9	001569	64.58	263.30	198.72	-0.64		313.09	49.79	-0.42		334.24	21.15	5.7
10	001570	65.13	265.46	200.33	0.16		315.57	50.11	0.22		335.08	19.51	-2.4
11	001571	68.71	268.09	199.38	-0.31		348.15	80.06	60.12		368.10	19.95	-0.2
12	001572	58.11	259.30	201.19	0.59		309.60	50.30	0.60		330.50	20.90	4.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการสุ่มทดสอบมาตรวัดน้ำ THAI AICHI ขนาด 3/4 นิ้ว

ทดสอบวันที่ 29/3/99

ลำดับที่	หมายเลข	ตัวเลขใน 2,500 ลิตร/ชม. ปริมาณน้ำ 200 ลิตร			% ผิดพลาด	ตัวเลขใน 200 ลิตร/ชม. ปริมาณน้ำ 50 ลิตร			% ผิดพลาด	ตัวเลขใน 50 ลิตร/ชม. ปริมาณน้ำ 20 ลิตร			
		ก่อนทดสอบ	หลังทดสอบ	ได้น้ำ		ก่อนทดสอบ	หลังทดสอบ	ได้น้ำ		ก่อนทดสอบ	หลังทดสอบ	ได้น้ำ	
1	001561	68.71	268.09	199.38	-0.31		318.15	50.06	0.12		338.10	19.95	-0
2	001562	58.11	259.30	201.19	0.59		309.27	49.97	-0.06		329.21	19.94	-0
3	001563	41.56	242.33	200.77	0.39		291.23	48.90	-2.20		311.73	20.50	2
4	001564	60.29	257.56	197.27	-1.36		307.07	49.51	-0.98		327.17	20.10	0
5	001565	57.18	256.13	198.95	-0.53		306.03	49.90	-0.20		326.61	20.58	2
6	001566	52.69	253.90	201.21	0.61		304.03	50.13	0.26		323.45	19.42	-2
7	001567	58.39	257.40	199.01	-0.50		307.08	49.68	-0.64		326.86	19.78	-1
8	001568	94.40	293.26	198.86	-0.57		343.04	49.78	-0.44		362.21	19.17	-4
9	001569	64.58	263.30	198.72	-0.64		313.09	49.79	-0.42		334.24	21.15	5
10	001570	65.13	265.46	200.33	0.16		315.57	50.11	0.22		335.08	19.51	-2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการสุ่มทดสอบมาตรวัดน้ำ THAI AICHI ขนาด 1 นิ้ว

ทดสอบวันที่ 29/3/99

ลำดับที่	หมายเลข	ตัวเลขใน 3,500 ลิตร/ชม. ปริมาณน้ำ 500 ลิตร			% ผิดพลาด	ตัวเลขใน 280 ลิตร/ชม. ปริมาณน้ำ 50 ลิตร			% ผิดพลาด	ตัวเลขใน 70 ลิตร/ชม. ปริมาณน้ำ 20 ลิตร			% ผิด
		ก่อนทดสอบ	หลังทดสอบ	ได้น้ำ	พลาด	ก่อนทดสอบ	หลังทดสอบ	ได้น้ำ	พลาด	ก่อนทดสอบ	หลังทดสอบ	ได้น้ำ	พล
1	001561	45.62	549.32	503.70	0.74		599.21	49.89	-0.22		619.00	19.79	-1.
2	001562	58.11	559.30	501.19	0.24		609.21	49.91	-0.18		629.90	20.69	3.
3	001563	41.56	542.33	500.77	0.15		591.30	48.97	-2.06		611.56	20.26	1.
4	001564	60.29	557.56	497.27	-0.55		607.76	50.20	0.40		627.98	20.22	1.
5	001565	57.18	556.13	498.95	-0.21		606.12	49.99	-0.02		624.40	18.28	-8.
6	001566	52.69	553.90	501.21	0.24		605.40	51.50	3.00		625.90	20.50	2.
7	001567	58.39	557.40	499.01	-0.20		607.39	49.99	-0.02		628.48	21.09	5.
8	001568	94.40	593.26	498.86	-0.23		643.30	50.04	0.08		664.18	20.88	4.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ACCURACY TEST AICHI WATER METER

Model : PXA(UHM) Date : 29/3/99 Inspector : ROOT

Serial No.	Flow rate (Lt./hr.)	Quantity (Lt.)	Meter Reading			Percentage Error		Remark
			Before	After	Difference	+	-	
001561	1,500 60	200 20	68.71	268.09 288.15	199.38 20.06	0.30	0.31	
001562	1,500 60	200 20	58.11	259.30 279.27	201.19 19.97	0.59	0.15	
001563	1,500 60	200 20	41.56	242.33 262.23	200.77 19.90	0.39	0.50	
001564	1,500 60	200 20	60.29	257.56 277.07	197.27 19.51		2.45	
001565	1,500 60	200 20	57.18	256.13 276.03	198.95 19.90		0.50	
001566	1,500 60	200 20	52.69	253.90 273.03	201.21 19.13	0.61	4.35	
001567	1,500 60	200 20	58.39	257.40 277.08	199.01 19.68		1.60	
001568	1,500 60	200 20	94.40	293.26 313.04	198.86 19.78		1.10	
001569	1,500 60	200 20	64.58	263.30 283.09	198.72 19.79		1.05	
001570	1,500 60	200 20	65.13	265.46 285.57	200.33 20.11	0.55		
001571	1,500 60	200 20	68.71	268.09 288.15	199.38 20.06	0.30	0.31	
001572	1,500 60	200 20	58.11	259.30 279.60	201.19 20.30	1.50		

Certified by : ..... Date : .....

Approved by : ..... Date : .....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ACCURACY TEST AICHI WATER METER

Model : PXA(UHM) Date : 29/3/99 Inspector : ROOT

Serial No.	Flow rate (Lt./hr.)	Quantity (Lt.)	Meter Reading			Percentage Error		Remark
			Before	After	Difference	+	-	
001561	2,400 100	500 50	45.62	549.32 599.21	503.70 49.89	0.74	0.22	
001562	2,400 100	500 50	58.11	559.30 609.21	501.19 49.91	0.24	0.18	
001563	2,400 100	500 50	41.56	542.33 591.30	500.77 48.97	0.15	2.06	
001564	2,400 100	500 50	60.29	557.56 607.76	497.27 50.20	0.40	0.55	
001565	2,400 100	500 50	57.18	556.13 606.12	498.95 49.99		0.02	
001566	2,400 100	500 50	52.69	553.90 605.40	501.21 51.50	3.00		
001567	2,400 100	500 50	58.39	557.40 607.39	499.01 49.99		0.02	
001568	2,400 100	500 50	94.40	593.26 643.30	498.86 50.04	0.08	0.23	
001569	2,400 100	500 50	64.58	563.30 613.09	498.72 49.79		0.42	
001570	2,400 100	500 50	65.13	565.46 615.57	500.33 50.11	0.22		

Certified by : ..... Date : .....

Approved by : ..... Date : .....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ACCURACY TEST AICHI WATER METER

Model : PXA(UHM) Date : 29/3/99 Inspector : ROOT

Serial No.	Flow rate (Lt./hr.)	Quantity (Lt.)	Meter Reading			Percentage Error		Remark
			Before	After	Difference	+	-	
001561	3,000 150	500 50	45.62	549.32 599.21	503.70 49.89	0.74	0.22	
001562	3,000 150	500 50	58.11	559.30 609.21	501.19 49.91	0.24	0.18	
001563	3,000 150	500 50	41.56	542.33 591.30	500.77 48.97	0.15	2.06	
001564	3,000 150	500 50	60.29	557.56 607.76	497.27 50.20	0.40	0.55	
001565	3,000 150	500 50	57.18	556.13 606.12	498.95 49.99		0.02	
001566	3,000 150	500 50	52.69	553.90 605.40	501.21 51.50	3.00		
001567	3,000 150	500 50	58.39	557.40 607.39	499.01 49.99		0.02	
001568	3,000 150	500 50	94.40	593.26 643.30	498.86 50.04	0.08	0.23	

Certified by : ..... Date : .....

Approved by : ..... Date : .....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ACCURACY TEST REPORT

Water meter Model : PXS Inspect By : BOOT

Size : 15 mm. Date : 29/3/99

Serial No.	Flow rate (lt / ht)	Quantity (lt.)	Reading			Percentage Error		Remarks
			Before	after	Difference	+	-	
01561	1500	100	68.71	168.20	99.49		0.5	
	120	20		187.95	19.75		1.2	
	30	20		208.52	20.57	2.8		
01562	1500	100	158.11	259.30	101.19	1.1		
	120	20		279.60	20.30	1.5		
	30	20		300.50	20.90	4.5		
01563	1500	100	141.56	242.33	100.77	0.7		
	120	20		261.23	18.90		5.5	
	30	20		279.73	18.50		7.5	
01564	1500	100	160.29	257.56	97.27		2.7	
	120	20		277.07	19.51		2.4	
	30	20		298.17	21.10	5.5		
01565	1500	100	157.18	256.13	98.95		1.0	
	120	20		276.03	19.90		0.5	
	30	20		296.61	20.58	2.9		
01566	1500	100	152.69	253.90	101.21	1.2		
	120	20		274.03	20.13	0.6		
	30	20		296.45	22.42	12.1		
01567	1500	100	158.39	257.40	99.01		0.9	
	120	20		277.08	19.68		1.6	
	30	20		296.86	19.78		1.1	
01568	1500	100	194.40	293.26	98.86		1.1	
	120	20		313.04	19.78		1.1	
	30	20		332.21	19.17		4.1	
01569	1500	100	164.58	263.30	98.72		1.2	
	120	20		283.09	19.79		1.0	
	30	20		304.24	21.15	5.7		
01570	1500	100	165.13	265.46	100.33	0.3		
	120	20		285.57	20.11	0.5		
	30	20		305.08	19.51		2.1	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ยกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของลิขสิทธิ์ทุกครั้งที่มีการนำไปใช้

Certified By : ..... Approved By : .....

Date : ..... Date : .....

ACCURACY TEST REPORT

Water meter Model : PXS Inspect By : ROOT

Size : 20 mm. Date : 29/3/99

Serial No.	Flow rate (lt/hr)	Quantity (lt.)	Reading			Percentage Error		Remarks
			Before	after	Difference	+	-	
001561	2500	100	145.62	248.32	102.70	2.7		
	200	50		298.21	49.89		0.2	
	50	20		318.90	20.69	3.4		
001562	2500	100	158.11	259.30	101.19	1.1		
	200	50		309.21	49.91		0.1	
	50	20		329.38	20.17	0.8		
001563	2500	100	141.56	242.33	100.77	0.7		
	200	50		291.30	48.97		2.0	
	50	20		311.56	20.26	1.3		
001564	2500	100	160.29	257.56	97.27		2.7	
	200	50		307.76	50.20	0.4		
	50	20		327.98	20.22	1.1		
001565	2500	100	157.18	256.13	98.95		1.0	
	200	50		306.12	49.99		0.0	
	50	20		324.40	18.28		8.6	
001566	2500	100	152.69	253.90	101.21	1.2		
	200	50		305.40	51.50	3.0		
	50	20		325.90	20.50	2.5		
001567	2500	100	159.39	257.40	98.01		1.9	
	200	50		307.39	49.99		0.0	
	50	20		328.48	21.09	5.4		
001568	2500	100	194.40	293.26	98.86		1.1	
	200	50		343.30	50.04	0.0		
	50	20		364.18	20.88	4.4		
001569	2500	100	164.58	263.30	98.72		1.2	
	200	50		313.50	50.20	0.4		
	50	20		333.29	19.79		1.0	
001570	2500	100	165.63	265.46	99.83		0.1	
	200	50		315.59	50.13	0.2		
	50	20		335.10	19.51		2.1	

Certified By : .....

Date : .....

Approved By : .....

Date : .....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ACCURACY TEST REPORT

Water meter Model : PXS Inspect By : ROOT

Size : 25 mm. Date : 29/3/99

Serial No.	Flow rate (lt / ht)	Quantity (lt.)	Reading			Percentage Error		Remarks
			Before	after	Difference	+	-	
01561	3500	100	145.62	249.32	103.70	3.7		
	280	50		299.21	49.89		0.2	
	70	20		319.08	19.79		1.0	
01562	3500	100	158.11	259.30	101.19	1.1		
	280	50		309.21	49.91		0.1	
	70	20		329.90	20.69	3.4		
01563	3500	100	141.56	242.33	100.77	3.7		
	280	50		291.30	48.97		2.0	
	70	20		311.56	20.26	1.3		
01564	3500	100	160.29	257.56	97.27		2.7	
	280	50		307.70	50.20	0.4		
	70	20		327.98	20.22	1.1		
01565	3500	100	157.18	256.13	98.95		1.0	
	280	50		306.12	49.99		0.0	
	70	20		324.40	18.28		8.6	
01566	3500	100	152.69	253.90	101.21	1.2		
	280	50		305.40	51.50	3.0		
	70	20		325.90	20.50	2.5		
01567	3500	100	158.39	257.40	99.01		0.9	
	280	50		307.39	49.99		0.0	
	70	20		328.48	21.09	5.4		
01568	3500	100	194.40	293.26	98.86		1.1	
	280	50		343.30	50.04	0.0		
	70	20		364.18	20.88	4.4		
	3500	100						
	280	50						
	70	20						
	3500	100						
	280	50						
	70	20						

Certified By : .....

Date : .....

Approved By : .....

Date : .....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการสุ่มทดสอบมาตรวัดน้ำTHAI AICHI ขนาด 1/2"

ทดสอบวันที่

ลำดับที่	หมายเลข D0250	ตัวเลขใน 1500 ลิตร/ชม. ปริมาณน้ำ 200 ลิตร			% ผิดพลาด	ตัวเลขใน 120 ลิตร/ชม. ปริมาณน้ำ 50 ลิตร			% ผิดพลาด	ตัวเลขใน 30 ลิตร/ชม. ปริมาณน้ำ 20 ลิตร			% ผิดพลาด
		ก่อนทดสอบ	หลังทดสอบ	ได้น้ำ		ก่อนทดสอบ	หลังทดสอบ	ได้น้ำ		ก่อนทดสอบ	หลังทดสอบ	ได้น้ำ	
1	C01561	68.71	268.09	199.38	-0.3		318.15	50.06	0.1		338.10	19.95	-0.2
2	001562	58.11	259.30	201.19	0.5		309.27	49.97	-0.0		329.21	19.94	-0.3
3	001563	41.56	242.33	200.77	0.3		291.23	48.90	-2.2		311.73	20.50	2.5
4	001564	60.29	257.56	197.27	-1.3		307.07	49.51	-0.9		327.17	20.10	0.5
5	001565	57.18	256.13	198.95	-0.5		301.03	49.90	-0.2		326.61	20.58	2.9
6	001566	52.69	253.90	201.21	0.6		304.03	50.13	0.2		323.45	19.42	-2.9
7	001567	58.39	257.40	199.01	-0.5		307.08	49.68	-0.6		326.86	19.78	-1.1
8	001568	94.40	293.26	198.86	-0.5		343.04	49.78	-0.4		362.21	19.17	-4.1
9	001569	64.58	263.30	198.72	-0.6		313.09	49.79	-0.4		334.24	21.15	5.7
10	001570	65.13	265.46	200.33	0.1		315.57	50.11	0.2		335.08	19.51	-2.4
11	001571	68.71	268.09	199.38	-0.3		348.15	80.06	60.1		368.10	19.95	-0.2
12	C01572	58.11	259.30	201.19	0.5		309.60	50.30	0.6		330.50	20.90	4.5
13													
14													
15													
16													
17													
18													
19													
20													
21													
22													
23													
24													
25													
26													
27													
28													
29													
30													
31													
32													

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการสุ่มทดสอบมาตรวัดน้ำ AICHITOKEI ขนาด 3/4"

ลำดับที่	หมายเลข ID1250	ตัวเลขใน 2500 ลิตร/ชม. ปริมาณน้ำ 200 ลิตร			% ผิดพลาด	ตัวเลขใน 200 ลิตร/ชม. ปริมาณน้ำ 50 ลิตร			% ผิดพลาด	ตัวเลขใน 50 ลิตร/ชม. ปริมาณน้ำ 20 ลิตร			% ผิดพลา
		ก่อนทดสอบ	หลังทดสอบ	ได้น้ำ		ก่อนทดสอบ	หลังทดสอบ	ได้น้ำ		ก่อนทดสอบ	หลังทดสอบ	ได้น้ำ	
1	CC1561	68.71	268.09	199.38	-0.3		318.15	50.06	0.1		338.10	19.95	-0.2
2	CC1562	58.31	259.30	201.19	0.5		299.27	49.97	-0.0		329.21	19.94	-0.3
3	001563	41.56	242.33	200.77	0.3		291.23	48.90	-2.2		311.73	20.50	2.5
4	001564	60.29	257.56	197.27	-1.3		307.07	49.51	-0.9		327.17	20.10	0.5
5	001565	57.18	256.13	198.95	-0.5		306.03	49.90	-0.2		326.61	20.58	2.9
6	CC1566	52.69	253.90	201.21	0.6		304.03	50.13	0.2		323.45	19.42	-2.9
7	001567	58.39	257.40	199.01	-0.5		307.08	49.68	-0.6		326.56	19.78	-1.1
8	001568	94.40	293.26	198.86	-0.5		343.04	49.78	-0.4		362.21	19.17	-4.1
9	001569	64.58	263.30	198.72	-0.6		313.09	49.79	-0.4		334.24	21.15	5.7
10	CC1570	65.13	265.46	200.33	0.1		315.57	50.11	0.2		335.08	19.51	-2.4
11													
12													
13													
14													
15													
16													
17													
18													
19													
20													
21													
22													
23													
24													
25													
26													
27													
28													
29													
30													

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการสุ่มทดสอบมาตรวัดน้ำ TIAI AICHI ขนาด 1"

ลำดับที่	D2250...	หัวเตยใน 3500 ลิตร/ชม. ปริมาณน้ำ 500 ลิตร			% ผิดพลาด	หัวเตยใน 280 ลิตร/ชม. ปริมาณน้ำ 50 ลิตร			% ผิดพลาด	หัวเตยใน 70 ลิตร/ชม. ปริมาณน้ำ 20 ลิตร			% ผิดพลาด
		ก่อนทดสอบ	หลังทดสอบ	ได้น้ำ		ก่อนทดสอบ	หลังทดสอบ	ได้น้ำ		ก่อนทดสอบ	หลังทดสอบ	ได้น้ำ	
1	001561	45.62	549.32	503.70	0.7		599.21	49.89	-0.2		619.00	19.79	-1.0
2	001562	58.11	559.30	501.19	0.2		609.21	49.91	-0.1		629.90	20.19	1.1
3	001563	41.56	542.33	500.77	0.1		591.30	48.97	-2.0		611.56	20.26	1.3
4	001564	10.29	557.56	497.27	-0.5		607.76	50.20	0.4		627.98	20.22	1.1
5	001565	57.18	556.13	498.95	-0.2		606.12	49.99	-0.0		624.40	18.28	-8.6
6	001566	52.19	553.90	501.21	0.2		605.40	51.50	3.0		625.90	20.50	2.5
7	001567	58.39	557.40	499.01	-0.2		607.39	49.99	-0.0		628.18	21.09	5.4
8	001568	94.40	593.26	498.86	-0.2		643.30	50.04	0.0		664.18	20.88	4.4
9													
10													
11													
12													
13													
14													
15													
16													
17													
18													
19													
20													

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Model PXA (UHM) 2 date 29/3/99 Inspector ROOT

Meter No.	Water flow Lt./hr.	Water Quantity Used (Lt)	Meter Reading Before	Meter Reading After	Meter Reading Difference	Percentage +	Percentage -	Remarks
001561	1500	200	68.71	268.09	199.38	0.3	0.3	
	60	20		288.15	20.06			
001562	1500	200	58.11	259.30	201.19	0.5	0.1	
	60	20		279.27	19.97			
001563	1500	200	41.56	242.33	200.77	0.3	0.5	
	60	20		262.23	19.90			
001564	1500	200	60.29	257.56	197.27		2.4	
	60	20		277.07	19.51			
001565	1500	200	57.18	256.13	198.95		0.5	
	60	20		276.03	19.90			
001566	1500	200	52.69	253.90	201.21	0.6	4.3	
	60	20		273.03	19.13			
001567	1500	200	58.39	257.40	199.01		1.6	
	60	20		277.08	19.68			
001568	1500	200	94.40	293.26	198.86		1.1	
	60	20		313.04	19.78			
001569	1500	200	64.58	263.30	198.72		2.0	
	60	20		283.09	19.79			
001570	1500	200	65.13	265.46	200.33	0.5		
	60	20		285.57	20.11			
001571	1500	200	68.71	268.09	199.38	0.3	0.3	
	60	20		288.15	20.06			
001572	1500	200	58.11	259.30	201.19	1.5		
	60	20		279.60	20.30			
	1500	200						
	60	20						
	1500	200						
	60	20						
	1500	200						

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Model : PXA (UHM) Date : 29/3/99 Inspector : BCOT

Meter Nr.	Water Flow It./hr.	Water Quantity Used (It.)	Meter Reading Before	After	Difference	Percentage Error +	-	Remarks
001561	2,400	500	45.62	549.32	503.70	0.7	0.2	
	100	50		599.21	49.89			
001562	2,400	500	58.11	559.30	501.19	0.2	0.1	
	100	50		609.21	49.91			
001563	2,400	500	41.56	542.33	500.77	0.1	2.0	
	100	50		591.30	48.97			
001564	2,400	500	60.29	557.56	497.27	0.4	0.5	
	100	50		607.76	50.20			
001565	2,400	500	57.18	556.13	498.95		0.0	
	100	50		606.12	49.99			
001566	2,400	500	52.69	553.90	501.21	3.0		
	100	50		605.40	51.50			
001567	2,400	500	58.39	557.40	499.01		0.0	
	100	50		607.39	49.99			
001568	2,400	500	92.40	593.26	498.86	0.0	0.2	
	100	50		643.30	50.04			
001569	2,400	500	64.58	563.30	498.72		0.1	
	100	50		613.09	49.79			
001570	2,400	500	65.13	565.46	500.33	0.2		
	100	50		615.57	50.11			
	2,400	500						
	100	50						
	2,400	500						
	100	50						
	2,400	500						
	100	50						
	2,400	500						
	100	50						
	2,400	500						
	100	50						

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Certified by : ..... Date : .....

Approved by : .....

Model : PXA (OHM) Date : 29/3/99 Inspector : BOOT

Meter Nr.	Water Flow It./hr.	Water Quantity Used (It.)	Meter Reading Before	Meter Reading After	Meter Reading Difference	Percentage Error +	Percentage Error -	Remarks
001561	3,000	500	45.62	549.32	503.70	0.7	0.2	
	150	50		599.21	49.89			
001562	3,000	500	58.11	559.30	501.19	0.2	0.1	
	150	50		609.21	49.91			
001563	3,000	500	41.56	542.33	500.77	0.1	2.0	
	150	50		591.30	48.97			
001564	3,000	500	60.29	557.56	497.27	0.4	0.5	
	150	50		607.76	50.20			
001565	3,000	500	57.18	556.13	498.95		0.0	
	150	50		606.12	49.99			
001566	3,000	500	52.69	553.90	501.21	3.0		
	150	50		605.40	51.50			
001567	3,000	500	58.39	557.40	499.01		0.0	
	150	50		607.39	49.99			
001568	3,000	500	94.40	593.26	498.86	0.0	0.2	
	150	50		643.30	50.04			
	3,000	500						
	150	50						
	3,000	500						
	150	50						
	3,000	500						
	150	50						
	3,000	500						
	150	50						
	3,000	500						
	150	50						
	3,000	500						
	150	50						
	3,000	500						
	150	50						

Certified by :..... Date :.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลและแนวทางการพัฒนา

5.1 สรุปผล

ไมโครเทอร์มินอลสำหรับบันทึกข้อมูลการตรวจสอบมิเตอร์น้ำที่ได้จัดทำเสร็จสมบูรณ์แล้วนั้น สามารถทำการบันทึกข้อมูลของการทดสอบมิเตอร์ได้สองแถวการทดสอบ โดยในการทดสอบนั้นได้จัดเงื่อนไขพารามิเตอร์ในการทดสอบมิเตอร์แต่ละแถวไว้แตกต่างกัน เช่น ใช้มิเตอร์ในกลุ่มที่แตกต่างกันให้ทดสอบพร้อมกัน หรือ ใช้มิเตอร์ในกลุ่มเดียวกันแต่ขนาดต่างกันมาทำการทดสอบพร้อมกัน ซึ่งพบว่าผลการทดสอบสามารถทำการทดสอบมิเตอร์พร้อมกันได้จริง และจากการทดสอบส่งข้อมูลไปยังคอมพิวเตอร์ด้วยวงจร current loop สามารถส่งได้ไกล 30 เมตร ด้วยความเร็วของการส่งข้อมูล 9600 บิทต่อวินาที ในส่วนความถูกต้องของข้อมูลนั้น เมื่อเทียบรายกับระบบเก่า (ระบบคำนวณด้วยมือ)นั้น พบว่า ระบบที่ได้รับการพัฒนานี้ สามารถให้ความละเอียดของข้อมูลได้สูงกว่า และมีความถูกต้อง น่าเชื่อถือมากกว่า นอกจากนั้น ในส่วนของการนำเสนอข้อมูลในรูปของรายงาน ก็มีความรวดเร็วกว่าระบบเดิมมาก และมีรูปแบบที่เหมาะแก่การนำเสนอมากกว่า

5.2 แนวทางการพัฒนา

1.) พัฒนารูปแบบของการทำงานให้เป็นระบบเครือข่าย ซึ่งสามารถป้อนข้อมูลการทดสอบได้จากหลายๆจุดทดสอบพร้อมๆกัน โดยใช้คอมพิวเตอร์เพียงเครื่องเดียวในการจัดการรายงาน

2.) พัฒนาโปรแกรมบนคอมพิวเตอร์ ให้สามารถจัดเก็บข้อมูลผลการทดสอบให้อยู่ในรูปของฐานข้อมูล

3.) ออกแบบส่วนรับข้อมูลให้มีรูปร่างที่กระทัดรัด และสวยงามขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก. โปรแกรมตัวรับข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
; ***********************************************************
; Program: Data Entry Unit for Micro Terminal for Water Meter Testing
; Developed by: Mr. Pichet Jitviriyavasin Department Applied Physics
;                Faculty of Science KMIT'L
; Version: 1.2 complete date 9 Dec 1998
; ***********************************************************
lcd_digit       equ     16
num_digit       equ     6
bause_time      equ     4
NADDR           equ     9
num_size        equ     3

; KEY DEFINE
line1           equ     10
line2           equ     11
select          equ     12
enter           equ     13
del             equ     14
test            equ     15

; Reserve data segment
                org     10h
lcd_buffer1:    ds      lcd_digit       ; reserve lcd buffer
                org     21h
lcd_buffer2:    ds      lcd_digit
sn_buffer:      ds      num_size        ; reserve serial number
int_buffer:     ds      num_size        ; reserve start value buffer
fin_buffer1:    ds      num_size        ; reserve finish value buffer
fin_buffer2:    ds      num_size        ; reserve finish value buffer
fin_buffer3:    ds      num_size        ; reserve finish value buffer

;\\\\\\\\\\\\\ Reserve Byte address \\\\\\\\\\\\\\
process:        ds      1
model:          ds      1
round:          ds      1
max_round:      ds      1
max_index:      ds      1
byte_status:    ds      2
key_buf:        ds      1
addr_l:         ds      1
addr_h:         ds      1
data:           ds      1
dt_counter:     ds      1
index:          ds      1
a1:             ds      3
a2:             ds      3
std:            ds      2
buffer:         ds      3
b_number:       ds      1

;\\\\\\\\\\\\\ Bit define \\\\\\\\\\\\\\\\
RS              bit     P0.5
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
EN              bit     P0.4
CS              BIT     p2.0            ; 93Cxx chip select
SK              BIT     p2.1            ; serial data clock
DI              BIT     p2.2            ; serial data input
DO              BIT     p2.3            ; serial data output
strb            bit     p3.4
sclk            bit     p3.5
pdata           bit     p3.7

;\\\\\\\\\\\\\ Reserve Bit address \\\\\\\\\\\\\\\\
key_flag        bit     0
key_release     bit     1
ans             bit     2
bit_status      bit     3
line_1          bit     4
line_2          bit     5
ZOV             BIT     6
sign            bit     7

; ####### function define #######
start           equ     0
lab1            equ     1
lab2            equ     2
lab3            equ     3
send_pc         equ     4

                org     000h
                ljmp    main
                org     30h
main:           mov     sp,#60h
                setb    strb                ; init ISP display module
                setb    sclk                ;
                setb    pdata               ;
                call    clear_lcd_buff
                call    delay1sec
                call    int_lcd
                call    init_eeprom
                mov     tmod,#00100001b     ; set timer0 to mode1/timer1 to mode2
                mov     scon,#01010010b     ; asynchronous transmission
                mov     th1,#0fdh           ; 9600 bps
                setb    tr1                 ; run baud rate generator
                clr     line_1
                clr     line_2
                clr     ans

main1:          mov     dptr,#title1
                call    put_string1
                mov     dptr,#title2
                call    put_string2
                call    put_to_lcd
                call    delay1sec
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
menu:       mov   dptr,#menu_msg
            call  put_string1
            mov   dptr,#menu_txt
            call  put_string2
            call  put_to_lcd
      mn00: call  key_wait
            mov   key_buf,a
            cjne  a,#line1,mn01
            mov   a,#'1'
            call  select_line
            setb  line_1
            clr   line_2
            mov   addr_h,#0
            setb  ans
            mov   b_number,#1
            jmp   mn00
      mn01: cjne  a,#line2,mn01_1
            mov   a,#'2'
            call  select_line
            clr   line_1
            setb  line_2
            mov   addr_h,#1
            setb  ans
            mov   b_number,#2
            jmp   mn00
      mn01_1:
            cjne  a,#test,mn02
            call  start_test
            jmp   mn00
      mn02: cjne  a,#enter,mn00
            jnb   ans,mn00
            clr   ans
            mov   r0,#process
            mov   r7,#5
            mov   addr_l,#0
            call  read_eeprom
            mov   a,process

            cjne  a,#start,mn04
      mn03: call  select_model
            call  get_start
            call  sound
            mov   process,#lab1
            mov   round,#0
            mov   addr_l,#0
            mov   r7,#5
            mov   r0,#process
            call  write_eeprom
            mov   a,process
            jmp   menu
      mn04: cjne  a,#lab1,mn05
            mov   r2,#fin_buffer1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            mov   r3,#6
            call  get_fin
            call  sound
            inc   round
            call  calc_proc
            call  data2board
            mov   process,#lab2
            mov   addr_l,#0
            mov   r0,#process
            mov   r7,#5
            call  write_eeprom
            jmp   menu
mn05:       cjne  a,#lab2,mn08
            mov   r2,#fin_buffer2
            mov   r3,#9
            call  get_fin
            call  sound
            inc   round
            call  calc_proc
            call  data2board
            mov   a,round
            cjne  a,max_round,mn06
            call  send_data
            mov   process,#start
            jmp   mn07
mn06:       mov   process,#lab3
mn07:       mov   addr_l,#0
            mov   r0,#process
            mov   r7,#5
            call  write_eeprom
            jmp   menu
mn08:       cjne  a,#lab3,mn09
            mov   r2,#fin_buffer3
            mov   r3,#12
            call  get_fin
            call  sound
            inc   round
            call  calc_proc
            call  data2board
            call  send_data
            mov   process,#start
            mov   addr_l,#0
            mov   r0,#process
            mov   r7,#5
            call  write_eeprom
mn09:       jmp   menu

select_line:  push  acc
              mov   dptr,#seline_msg
              call  put_string1
              pop   acc
              mov   r0,#lcd_buffer1+13
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                mov     @r0,a
                mov     dptr,#press_msg
                call    put_string2
                call    put_to_lcd
                ret

;::::::::::::::::::::: SCAN KEY RUTINE ::::::::::::::::::::::::::::::;
scan_key:       mov     p1,#0ffh
                mov     r7,#4
                mov     r6,#01111111b
        sk00:   mov     a,r6
                rl      a
                anl     p1,a
                mov     r6,a
                mov     b,p1
                mov     a,#00001111b
                orl     a,b
                cpl     a
                jz      sk01
                mov     key_buf,b
                setb    key_flag
                clr     key_release
                ret
        sk01:   orl     p1,#0ffh
                djnz    r7,sk00
                setb    key_release
                ret

decode:         mov     a,key_buf
                cjne    a,#7dh,de00
                mov     a,#0
        de00:   cjne    a,#0eeh,de01
                mov     a,#1
        de01:   cjne    a,#0edh,de02
                mov     a,#2
        de02:   cjne    a,#0ebh,de03
                mov     a,#3
        de03:   cjne    a,#0deh,de04
                mov     a,#4
        de04:   cjne    a,#0ddh,de05
                mov     a,#5
        de05:   cjne    a,#0dbh,de06
                mov     a,#6
        de06:   cjne    a,#0beh,de07
                mov     a,#7
        de07:   cjne    a,#0bdh,de08
                mov     a,#8
        de08:   cjne    a,#0bbh,de09
                mov     a,#9
        de09:   cjne    a,#0e7h,de10
                mov     a,#10
        de10:   cjne    a,#0d7h,de11
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                mov     a,#11
        de11:   cjne    a,#0b7h,de12
                mov     a,#12
        de12:   cjne    a,#77h,de13
                mov     a,#13
        de13:   cjne    a,#7eh,de14
                mov     a,#14
        de14:   cjne    a,#7bh,end_de
                mov     a,#15
end_de:         ret

debause:        mov     r7,#bause_time
debause00:      mov     r6,#32h
debause01:      mov     r5,#0
                djnz    r5,$
                djnz    r6,debause01
                djnz    r7,debause00
                ret

key_wait:       call    scan_key
                jnb     key_release,key_wait
                call    debause
                clr     key_flag
key_wait1:      call    scan_key
                jnb     key_flag,key_wait1
                call    sound
                call    sound
                clr     key_flag
                call    decode
                ret

sound:          mov     r6,#2
sound1:         mov     r5,#00
loop:           mov     r7,#100
                djnz    r7,$
                cpl     p2.4
                djnz    r5,loop
                djnz    r6,sound1
                setb    p2.4
                ret

; $$$$$$$$$$$ FUNCTION GET DATA $$$$$$$$$$$$
get_data:       call    key_wait
                mov     key_buf,a
                cjne    a,#0ah,$+3
                jc      gd00
                ret
        gd00:   cjne    @r0,#space,get_data
                mov     r7,dt_counter
        gd01:   inc     r0
                mov     a,@r0
                dec     r0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            mov   @r0,a
            inc   r0
            djnz  r7,gd01
            mov   a,key_buf
            call  htoa
            mov   @r0,a
            call  put_to_lcd
            ret

select_model: call  clear_lcd_buff
            mov   dptr,#model_msg
            call  put_string1
            mov   model,#1
            mov   lcd_buffer2,#':'
      sm00: mov   r0,#lcd_buffer2+1
            call  put_model
            call  put_to_lcd
            call  key_wait
            mov   key_buf,a

            cjne  a,#select,sm02
            mov   a,model
            cjne  a,#6,$+3
            jc    sm01
            mov   model,#1
            jmp   sm00
      sm01: inc   model
            jmp   sm00

      sm02: cjne  a,#enter,sm00
            mov   a,model
            swap  a
            mov   std,a
            cjne  a,#50h,$+3
            jc    sm03
            mov   max_round,#2
            sjmp  sm04
      sm03: mov   max_round,#3

      sm04: mov   dptr,#model_msg2
            call  put_string1
            mov   r0,#lcd_buffer1+8
            call  put_model
            mov   dptr,#size_msg
            call  put_string2
            mov   dptr,#M_size
            mov   model,#0
      sm05: mov   a,model
            movc  a,@a+dptr
            mov   r0,#lcd_buffer2+12
            call  htoa
            mov   @r0,a
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                mov     a,b
                dec     r0
                mov     @r0,a
                call    put_to_lcd
                call    key_wait
                mov     key_buf,a

                cjne    a,#select,sm07
                mov     a,model
                cjne    a,#2,$+3
                jc      sm06
                mov     model,#0
                sjmp    sm05
        sm06:   inc     model
                sjmp    sm05

        sm07:   cjne    a,#enter,sm05
                mov     a,model
                mov     dptr,#N_units
                movc    a,@a+dptr
                mov     max_index,a
                mov     r0,#std
                xchd    a,@r0
                mov     model,std
                mov     a,model
                ret
N_units:        db      0ch,0ah,8h
M_size:         db      15h,20h,25h

put_model:      mov     dptr,#model_txt
                mov     r6,model
                jmp     pm01
        pm00:   clr     a
                movc    a,@a+dptr
                inc     dptr
                jnz     pm00
        pm01:   djnz    r6,pm00
        pm02:   clr     a
                movc    a,@a+dptr
                jz      pm03
                mov     @r0,a
                inc     dptr
                inc     r0
                jmp     pm02
        pm03:   ret
model_txt:      db      'PXS    ',00h           ; max round = 3
                db      'TDS    ',00h           ;     .
                db      'TDA(UHM)',00h          ;     .
                db      'TDA(MWA)',00h          ;     .
                db      'TDA(PWA)',00h          ; max round = 2
                db      'PXA    ',00h           ;     .
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
; $$$$$$$$$$ FUNCTION GET SERIAL NUMBER $$$$$$$$$$
get_start:      mov     dptr,#line_msg1
                call    put_string1
                call    put_line
                mov     dptr,#sn_msg
                call    put_string2
                call    put_to_lcd
                mov     index,#0
                mov     dptr,#sn
                call    put_string2
        gs00:   mov     r1,#sn_buffer
                call    get_val
                mov     dptr,#line_msg2
                call    put_string1
                call    put_line
                mov     r0,#lcd_buffer1+(lcd_digit-num_digit)
                mov     r1,#lcd_buffer2+(lcd_digit-num_digit)
                mov     r7,#num_digit
        gs01:   mov     a,@r1
                mov     @r0,a
                inc     r0
                inc     r1
                djnz    r7,gs01
                mov     dptr,#init
                call    put_string2
                call    put_to_lcd
                mov     r1,#int_buffer
                call    get_val
                mov     a,index
                call    find_addr
                mov     r0,#sn_buffer
                mov     r7,#num_size+num_size
                call    write_eeprom
                inc     index
                mov     a,index
                cjne    a,max_index,gs02
                ret
        gs02:   mov     dptr,#line_msg1
                call    put_string1
                call    put_line
                mov     dptr,#sn
                call    put_string2
                mov     r0,#sn_buffer+2
                mov     r7,#3
                setb    c
        gs03:   mov     a,@r0
                addc    a,#0
                da      a
                mov     @r0,a
                dec     r0
                djnz    r7,gs03
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                mov     r0,#lcd_buffer2+(lcd_digit-num_digit)
                mov     r1,#sn_buffer
                mov     r6,#num_size
        gs04:   mov     a,@r1
                call    htoa
                mov     r7,#2
        gs05:   push    acc
                mov     a,b
                mov     @r0,a
                inc     r0
                pop     b
                djnz    r7,gs05
                inc     r1
                djnz    r6,gs04
                call    put_to_lcd
                jmp     gs00

put_line:       mov     r0,#lcd_buffer1+4
                jnb     line_1,pl00
                mov     a,#'1'
                jmp     pl01
        pl00:   mov     a,#'2'
        pl01:   mov     @r0,a
                ret

; $$$$$$$$$$ FUNCTION GET FINISH VALUE $$$$$$$$$$$
get_fin:        mov     index,#0
                mov     dptr,#line_msg2
                call    put_string1
                call    put_line
        gf00:   mov     a,index
                call    find_addr
                mov     r0,#sn_buffer
                mov     r7,#num_size
                call    read_eeprom
                mov     r0,#lcd_buffer1+(lcd_digit-num_digit)
                mov     r1,#sn_buffer
                mov     r6,#num_size
        gf01:   mov     a,@r1
                call    htoa
                mov     r7,#2
        gf02:   push    acc
                mov     a,b
                mov     @r0,a
                inc     r0
                pop     b
                djnz    r7,gf02
                inc     r1
                djnz    r6,gf01

                mov     dptr,#fin
                call    put_string2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                call    put_to_lcd
                mov     r1,02
                call    get_val
                mov     a,index
                call    find_addr
                mov     a,r3
                add     a,addr_l
                mov     addr_l,a
                mov     r0,02
                mov     r7,#num_size
                call    write_eeprom
                inc     index
                mov     a,index
                cjne    a,max_index,gf00
                ret

; $$$$$$$$$$$ FUNCTION GET INITIAL VALUE $$$$$$$$$$$$
get_val:        mov     r0,#lcd_buffer2+(lcd_digit - num_digit)
                mov     dt_counter,#num_digit-1
                call    get_data
                jc      get_val
                mov     a,key_buf

                cjne    a,#del,gi01
                mov     r7,#num_digit-1
                mov     r0,#lcd_buffer2+lcd_digit - 1
                call    del_digit
                jmp     get_val
        gi01:   cjne    a,#enter,get_val
                mov     a,lcd_buffer2+lcd_digit - 1
                cjne    a,#space,gi01_1
                jmp     get_val
        gi01_1:mov      r7,#num_size
                mov     r0,#lcd_buffer2+(lcd_digit - num_digit)
        gi02:   mov     a,@r0
                orl     a,#30h
                mov     b,a
                inc     r0
                mov     a,@r0
                orl     a,#30h
                call    atoh
                mov     @r1,a
                inc     r0
                inc     r1
                djnz    r7,gi02
                ret

;|||||||||||||||| FUNCTION DELETE LAST DIGIT ||||||||||||||||
del_digit:      dec     r0
                mov     a,@r0
                inc     r0
                mov     @r0,a
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                dec     r0
                djnz    r7,del_digit
                mov     a,#space
                mov     @r0,a
                call    put_to_lcd
                ret

find_content:   mov     a,round
                mov     b,#num_size
                mul     ab
                mov     b,a
                add     a,#sn_buffer
                mov     r0,a
                mov     a,b
                ret

; ########## FUNCTION CALCULATE PROCESS ###########
calc_proc:      call    open_table
                mov     index,#0
        cp00:   mov     a,index
                call    find_addr
                call    find_content
                add     a,addr_l
                mov     addr_l,a
                mov     r7,#num_size * 2
                call    read_eeprom

                call    find_content
                call    dtoh
                mov     r1,#a1
                call    push_buff
                call    dtoh
                mov     r1,#a2
                call    push_buff
                call    calc
                mov     a,round
                cjne    a,#3,$+3
                mov     a,a1
                jnc     cp03
                cjne    a,#2,$+3
        cp01:   jnc     cp02
                setb    bit_status
                jmp     cp04
        cp02:   clr     bit_status
                jmp     cp04
        cp03:   cjne    a,#5,$+3
                jmp     cp01
        cp04:   mov     r0,#byte_status
                call    rr16bit
                inc     index
                mov     a,index
                cjne    a,max_index,cp00
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            mov   r0,#byte_status
cp05:       clr   bit_status
            call  rr16bit
            inc   index
            mov   a,index
            cjne  a,#16,cp05
            ret

push_buff:  mov   a,r0
            add   a,#num_size
            mov   r0,a
            mov   @r1,dpl
            inc   r1
            mov   @r1,dph
            inc   r1
            dec   r0
            mov   a,@r0
            mov   @r1,a
            inc   r0
            ret

open_table: mov   a,model
            swap  a
            anl   a,#0fh
            cjne  a,#4,$+3
            jnc   ot00
            mov   dptr,#volume1
            jmp   ot02
ot00:       cjne  a,#5,$+3
            jnc   ot01
            mov   dptr,#volume2
            jmp   ot02
ot01:       mov   dptr,#volume3
ot02:       mov   a,max_index
            cjne  a,#12,ot03
            mov   r6,#1
            jmp   ot07
ot03:       cjne  a,#10,ot04
            mov   r6,#2
            jmp   ot07
ot04:       mov   r6,#3
            jmp   ot07
ot05:       mov   r7,max_round
ot06:       inc   dptr
            inc   dptr
            djnz  r7,ot06
ot07:       djnz  r6,ot05
            mov   r7,round
            jmp   ot09
ot08:       inc   dptr
            inc   dptr
ot09:       djnz  r7,ot08
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                mov     r1,#std
                mov     r7,#2
        ot10:   clr     a
                movc    a,@a+dptr
                mov     @r1,a
                inc     r1
                inc     dptr
                djnz    r7,ot10
                ret
volume1:        db
64h,00h,14h,00h,14h,00h,64h,00h,32h,00h,14h,00h,64h,00h,32h,00h,14h,00h
volume2:        db
0c8h,00h,32h,00h,14h,00h,0c8h,00h,32h,00h,14h,00h,0f4h,01h,32h,00h,14h,00h
volume3:        db      0c8h,C0h,14h,00h,0f4h,01h,32h,00h,0f4h,01h,32h,00h

rr16bit:        inc     r0
                mov     a,@r0
                mov     c,bit_status
                rrc     a
                mov     @r0,a
                dec     r0
                mov     a,@r0
                rrc     a
                mov     @r0,a
                ret

;------------------------ Service Rutines ------------------------
data2board:     mov     r7,#00h
                mov     a,b_number
                add     a,max_index
        one:    call    send_process        ;send address of board to latch
                cjne    r7,#08h,one
                mov     r7,#00h
                mov     a,byte_status
        two:    call    send_process        ;send pass/fail data 8 bits (led1-8)
                cjne    r7,#08h,two
                mov     r7,#00h
                mov     a,byte_status+1
        three:  call    send_process        ;send pass/fail data 4 bits (led9-12)
                cjne    r7,#04h,three
                ret

send_process:   rrc     a
                call    send_clk
                inc     r7
                ret

send_clk:       jb      strb,$
                mov     pdata,c
                clr     sclk
                jnb     strb,$
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                setb    sclk
                ret

start_test:     push    max_index
                push    b_number

                mov     max_index,#0
                mov     b_number,#9
                mov     byte_status,#10101010b
                mov     byte_status+1,#10101010b

        ss00:   call    data2board
                call    delay
                call    swap_data
                call    data2board
                call    delay
                call    swap_data
                inc     b_number
                mov     a,b_number
                cjne    a,#0fh,ss00

                pop     b_number
                pop     max_index
                ret

delay:          mov     r1,#00h
delay1:         mov     r2,#30h
delay2:         mov     r3,#10h
                djnz    r3,$
                djnz    r2,delay2
                djnz    r1,delay1
                ret

swap_data:      mov     a,byte_status
                rr      a
                mov     byte_status,a
                mov     a,byte_status+1
                rr      a
                mov     byte_status+1,a
                ret

;*************************************************8
send_data:      mov     dptr,#send_msg1
                call    put_string1
                mov     dptr,#press_msg
                call    put_string2
                call    put_to_lcd
        sd00:   call    key_wait

                cjne    a,#enter,sd00
                mov     a,model
                call    send_word
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                mov     dptr,#new_line
                call    send_string
                clr     a
                mov     index,a
        sd01:   call    find_addr
                mov     r0,#sn_buffer
                mov     r7,#num_size * 5
                call    read_eeprom
                mov     r0,#sn_buffer
                mov     r6,#3
        sd02:   mov     r7,#num_size
                call    send_pack
                djnz    r6,sd02
                mov     r7,#num_size
                mov     a,max_round
                cjne    a,#3,sd03
                call    send_pack
                mov     r7,#num_size
        sd03:   call    send_pack
                mov     dptr,#new_line
                call    send_string
                inc     index
                mov     a,index
                cjne    a,max_index,sd01
                ret

send_word:      call    htoa
                push    acc
                mov     a,b
                call    send
                pop     acc
                call    send
                ret

send_pack:      mov     a,@r0
                inc     r0
                call    send_word
                djnz    r7,send_pack
                ret

;send_string send string constant to terminal
send_string:    clr     a
                movc    a,@a+dptr
                cjne    a,#eos,send_string1
                ret
send_string1:   push    dpl
                push    dph
                lcall   send
                pop     dph
                pop     dpl
                inc     dptr
                sjmp    send_string
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
send:       jnb   ti,send
            clr   ti
            mov   sbuf,a
            ret

delay1sec:  mov   r7,#32h
delay1sec1: mov   r6,#64h
delay1sec2: mov   r5,#64h
            djnz  r5,$
            djnz  r6,delay1sec2
            djnz  r7,delay1sec1
            ret

;::::::::::::::::::::::: LCD DISPLAY RUTINE ::::::::::::::::::::::::::
int_lcd:    mov   a,#00101010b      ;function set
            call  lcdwi
            mov   a,#00101001b
            call  lcdwi
            mov   a,#00001110b      ;display on/off
            call  lcdwi
            mov   a,#01h            ;clear
            call  lcdwi
            ret
lcdwi:      push  acc
            mov   r6,#2
      li00: swap  a
            call  send_4bit
            clr   rs
            clr   en
            nop
            setb  en
            nop
            clr   en
            djnz  r6,li00
            pop   acc
            call  wait_lcd
            ret

lcdwd:      push  acc
            mov   r6,#2
      ld00: swap  a
            call  send_4bit
            setb  rs
            clr   en
            nop
            setb  en
            nop
            clr   en
            djnz  r6,ld00
            pop   acc
            call  wait_lcd
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ret

wait_lcd:       mov     r6,#1
                clr     a
wait_lcd1:      dec     a
                jnz     wait_lcd1
                djnz    r6,wait_lcd1
                ret

;:::::::::::: RUTINE FOR DISPLAY LCD FROM LCD BUFFER ::::::::::::
put_to_lcd:     mov     a,#1
                call    lcdwi
                mov     a,#00h
                mov     r0,#lcd_buffer1
                call    ptl00
                call    wait_lcd
                mov     a,#0c0h
                mov     r0,#lcd_buffer2
                call    ptl00
                ret

        ptl00:  call    lcdwi
                mov     r7,#16
        ptl01:  mov     a,@r0
                call    lcdwd
                inc     r0
                djnz    r7,ptl01
                ret

put_string1:    mov     r0,#lcd_buffer1
        ps100:  clr     a
                movc    a,@a+dptr
                cjne    a,#eos,ps101
                ret
        ps101:  mov     @r0,a
                inc     r0
                inc     dptr
                jmp     ps100

put_string2:    mov     r0,#lcd_buffer2
     ps200:     clr     a
                movc    a,@a+dptr
                cjne    a,#eos,ps201
                ret
     ps201:     mov     @r0,a
                inc     r0
                inc     dptr
                jmp     ps200

;::::::::::::::::: Service Rutine for LCD display ::::::::::::::::::
send_4bit:      push    acc
                anl     a,#0fh
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                anl     p0,#0f0h
                orl     p0,a
                pop     acc
                ret

clear_lcd_buff:
                mov     r0,#lcd_buffer1
                mov     r6,#16
                mov     a,#space
        clb00:  mov     @r0,a
                inc     r0
                djnz    r6,clb00

                mov     r0,#lcd_buffer2
                mov     r6,#16
        clb01:  mov     @r0,a
                inc     r0
                djnz    r6,clb01
                ret

; ********** htoa sub **********
htoa:           push    acc
                swap    a
                call    htoas
                mov     b,a
                pop     acc
                call    htoas
                ret
        htoas:  anl     a,#0fh
                cjne    a,#0ah,$+3
                jnc     htoas1
                orl     a,#30h
                ret
        htoas1: subb    a,#9
                orl     a,#40h
                ret

; ********** atoh sub **********
atoh:           push    acc
                mov     a,b
                call    atohs
                swap    a
                mov     b,a
                pop     acc
                call    atohs
                orl     a,b
                ret
         atohs: cjne    a,#'a',$+3
                jc      atohs1
                add     a,#9
        atohs1: anl     a,#0fh
                ret
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
; ********** htod sub **********
htod:       clr   a           ;clear output
            mov   r1,a
            mov   r2,a
            mov   r3,a
            mov   r4,#16      ;shift 16 bit
      htod1: mov  a,dpl
            rlc   a
            mov   dpl,a
            mov   a,dph
            rlc   a
            mov   dph,a
            mov   r5,#3       ;add decimal
            mov   r0,#3       ;index to r3
      htod2: mov  a,@r0
            addc  a,acc
            da    a
            mov   @r0,a
            dec   r0
            djnz  r5,htod2
            djnz  r4,htod1
            ret

; ********** dtoh sub **********
dtoh:       mov   r4,#16
      dtoh1: push 00h
            mov   r5,#3       ;shift & sub
            clr   c
      dtoh2: mov  a,@r0
            rrc   a
            push  psw         ;-[
            jnb   acc.7,dtoh3
            clr   c
            subb  a,#30h
      dtoh3: jnb  acc.3,dtoh4
            clr   c
            subb  a,#03h
      dtoh4: mov  @r0,a
            inc   r0
            pop   psw         ;-]
            djnz  r5,dtoh2
            mov   a,dph
            rrc   a
            mov   dph,a
            mov   a,dpl
            rrc   a
            mov   dpl,a
            pop   00h
            djnz  r4,dtoh1
            ret
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
; ############ function calculate percent #############
calc:       clr   c
            call  subb24
            jnc   ca00
            ret
      ca00: mov   a1,std
            mov   a1+1,std+1
            mov   a,#100
            call  mul_a_16
            call  subb24
            jnc   ca01
            mov   a1,#0
            mov   a1+1,#0
            mov   a1+2,#0
            call  subb24
            call  cpl24
            setb  sign
            jmp   ca02
      ca01: mov   a1,a2
            mov   a1+1,a2+1
            mov   a1+2,a2+2
            clr   sign
      ca02: mov   a2,std
            mov   a2+1,std+1
            call  div16
            ret

cpl24:      mov   r0,#a1
            mov   r1,#a2
            mov   r7,#3
    c24_00: mov   a,@r1
            cpl   a
            mov   @r0,a
            inc   r0
            inc   r1
            djnz  r7,c24_00
            ret

subb24:     mov   r0,#a1
            mov   r1,#a2
            mov   r7,#3
            clr   c
    s24_00: mov   a,@r1
            subb  a,@r0
            mov   @r1,a
            inc   r0
            inc   r1
            djnz  r7,s24_00
            mov   zov,c
            ret

sub16:      mov   a,a1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            clr     c
            subb    a,a2
            mov     buffer,a
            mov     a,a1+1
            subb    a,a2+1
            mov     buffer+1,a
            mov     zov,c
            ret

div16:      mov     a,a2+1
            orl     a,a2
            jnz     div_ok
            setb    c
            ret
div_ok:     mov     r1,a1+1
            mov     r0,a1
            mov     a1,#0
            mov     a1+1,#0
            mov     r3,#0
            mov     r2,#0
            mov     r7,#16
div_loop:   clr     c
            mov     a,r0
            rlc     a
            mov     r0,a
            mov     a,r1
            rlc     a
            mov     r1,a
            mov     a,a1
            rlc     a
            mov     a1,a
            mov     a,a1+1
            rlc     a
            mov     a1+1,a
            call    sub16
            mov     c,zov
            cpl     c
            jnc     div_1
            mov     a1+1,buffer+1
            mov     a1,buffer
div_1:      mov     a,r2
            rlc     a
            mov     r2,a
            mov     a,r3
            rlc     a
            mov     r3,a
            djnz    r7,div_loop
            mov     a2+1,a1+1
            mov     a2,a1
            mov     a1+1,r3
            mov     a1,r2
            clr     c
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                mov     zov,c
                ret

mul_a_16:       mov     psw,#0
                push    acc
                mov     b,a1
                mul     ab
                mov     a1,a
                pop     acc
                push    b
                mov     b,a1+1
                mul     ab
                pop     0
                add     a,r0
                mov     a1+1,a
                clr     a
                addc    a,b
                mov     b,a
                mov     a1+2,#0
                ret

init_eeprom:    clr     cs              ; low for eeprom
                clr     sk              ; low
                setb    di              ; high
                setb    do              ; high
                call    ewen            ; enable write for eeprom
                call    eral
                mov     data,#0
                call    wral
                ret

; !!!!!!!!!!! FUNCTION FIND ADDRESS FOR EEPROM !!!!!!!!!!
find_addr:      mov     b,#15
                mul     ab
                add     a,#5
                mov     addr_l,a
                ret

; !!!!!!!!!!!!!!!! FUNCTION WRITE DATA TO EEPROM !!!!!!!!!!!!!!!
;       write data in content of R0 to address of ADDR
write_eeprom:   mov     a,@r0
                mov     data,a
                call    write
                inc     addr_l
                inc     r0
                djnz    r7,write_eeprom
                ret

; !!!!!!!!!!!!!!!!! FUNCTION READ DATA TO EEPROM !!!!!!!!!!!!!!!!!!!!
read_eeprom:    call    read
                mov     a,data
                mov     @r0,a
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                inc     addr_l
                inc     r0
                djnz    r7,read_eeprom
                ret

; Read data from addressed byte or word.
read:           setb    CS              ; raise CS
                mov     dpl, #110b
                mov     b, #3
                call    outdata
                mov     dpl,addr_l      ; low byte
                mov     dph,addr_h      ; high byte (may be undefined)
                mov     b, #NADDR       ; bit count
                call    outdata
                call    indata          ; get return data
                clr     CS              ; drop CS
                ret

; Write data into the addressed byte or word.
write:          setb    CS              ; raise CS
                mov     dpl, #101b
                mov     b, #3
                call    outdata
                mov     dpl,addr_l      ; low byte
                mov     dph,addr_h      ; high byte (may be undefined)
                mov     b, #NADDR       ; bit count
                call    outdata
                mov     dpl,data
                mov     b,#8
                call    outdata
                clr     CS              ; drop CS
                call    status          ; returns CY
                ret

WRAL:           Setb    CS              ; raise CS
                mov     dptr, #(10001b SHL (NADDR-2))
                mov     b,NADDR+3
                call    outdata
                mov     dpl,data
                mov     b,#8h
        ee72:   call    outdata
                clr     CS              ; drop CS
                call    status          ; returns CY
                ret

; Erase all addresses.
eral:           setb    CS              ; raise CS
                mov     dptr, #(10010b SHL (NADDR-2))
                mov     b, #(NADDR+3)
                call    outdata
                clr     CS              ; drop CS
                call    status          ; returns CY
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ret
; Enable erase/write.
ewen:           setb    CS              ; raise CS
                mov     dptr, #(10011b SHL (NADDR-2))
                mov     b, #(NADDR+3)
                call    outdata
                clr     CS              ; drop CS
                ret
; Read device status.
status:         push    b
                setb    DO              ; float pin
                setb    CS              ; select device
                mov     b, #220         ; 220 * 50 uS = 11 mS

        ee40:   push    b               ; 2 uS
                mov     b, #22          ; 2 uS
                djnz    b, $            ; 2 uS * 22
                pop     b               ; 2 uS
                jb      DO, ee41        ; jump if DO is high (READY)
                djnz    b, ee40         ; next pass
                setb    c               ; flag error
                jmp     ee42            ; return
        ee41:
                clr     c               ; clear error flag
        ee42:
                clr     CS              ; deselect device
                pop     b
                ret

; Fetch data from the AT93Cxx. The AT93Cxx family presents data
indata:         setb    DO              ; float pin
                clr     SK              ; drop clock
                call    shin
                mov     data,a          ; save low byte
                ret

; Shift in eight bits from the AT93Cxx, most significant bit first,
shin:           push    b
                clr     SK              ; drop clock
                mov     b, #8           ; init loop / delay min one uS
        ee30:
                setb    SK              ; raise clock
                nop                     ; delay min one uS
                mov     c, DO           ; read serial data output
                rlc     a               ; shift in bit / delay min one uS
                clr     SK              ; drop clock
                djnz    b, ee30         ; next bit / delay min one uS
                pop     b
                ret

; Massage data and move it out to the AT93Cxx.
outdata:        push    b
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                mov     a, b            ; get bit count
                clr     c               ; compare count to eight
                subb    a, #8           ;
                jc      ee6             ; jump if count less than eight
                jz      ee5             ; jump if count equals eight
                mov     b, a            ; save adjusted bit count
                clr     c               ; compare adjusted count to eight
                subb    a, #8           ;
                jc      ee2             ; jump if count less than eight
                jnz     ee9             ; jump if count greater than eight
                mov     a, dph
                jmp     ee4
        ee2:
                push    b               ; save count
                mov     a, dph          ; get data
        ee3:
                rr      a               ; shift data
                djnz    b, ee3          ;
                pop     b               ; restore count
        ee4:
                call    shout
                mov     b, #8
        ee5:
                mov     a, dpl
                jmp     ee8
        ee6:
                push    b               ; save count
                mov     a, dpl          ; get data
        ee7:
                rr      a               ; shift data
                djnz    b, ee7          ;
                pop     b               ; restore count
        ee8:
                call    shout
        ee9:
                pop     b
                ret

; Shift out a byte to the AT93Cxx, most significant bit first.
shout:          clr     SK              ; drop clock
                rlc     a               ; move bit into CY
                mov     DI, c           ; output bit
                nop                     ; delay min 400 nS
                setb    SK              ; raise clock
                djnz    b, shout        ; next bit / delay min one uS
                clr     SK              ; drop clock
                ret

;::::::::::::::::::::::: String Constant :::::::::::::::::::::::
title1:         db      'APL',39d,'s data entry ',eos
title2:         db      'units version1.2',eos
menu_msg:       db      ' Meter',39d,'s test  ',eos
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
menu_txt:     db    ' --- READY ---  ',eos
seline_msg:   db    'Select : LINE   ',eos
press_msg:    db    'Press ENTER key ',eos
model_msg:    db    'Select model ?  ',eos
model_msg2:   db    'Model :         ',eos
size_msg:     db    'Size(mm) :      ',eos
line_msg1:    db    'LINE            ',eos
line_msg2:    db    'LINE : No.      ',eos
sn_msg:       db    'Serial Number ? ',eos
sn:           db    'Sn Number-      ',eos
init:         db    'Init val :      ',eos
fin:          db    'Fin. val :      ',eos
send_msg1:    db    'Ready to upload ',eos
new_line:     db    cr,lf,eos

eos           equ   10h
space         equ   20h
null          equ   00h
block         equ   0ffh
comma         equ   44
cr            equ   0DH
lf            equ   0AH

              end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข. โปรแกรมส่วนแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
; ******************************************************
; file thesis3.html is created  13/03/99.
; it recieve data for b_number ,data1 ,data2 from spi interface.
; and it use only 1 rutine for display data,call decode_data.
; decode data is content both decode_data and latch board and set timer.
; and it will check ,if b_number is not match anychoise,1,a,b,c,d,e
; it will do error_b_number that on all red led both board without timer.
; ******************************************************

grn1        equ   p0.0
grn2        equ   p0.1
grn3        equ   p0.2
grn4        equ   p0.3
grn5        equ   p0.4
grn6        equ   p0.5
grn7        equ   p0.6
grn8        equ   p0.7
grn9        equ   p1.0
grn10       equ   p1.1
grn11       equ   p1.2
grn12       equ   p1.3
red1        equ   p1.4
red2        equ   p1.5
red3        equ   p1.6
red4        equ   p1.7
red5        equ   p2.0
red6        equ   p2.1
red7        equ   p2.2
red8        equ   p2.3
red9        equ   p2.4
red10       equ   p2.5
red11       equ   p2.6
red12       equ   p2.7
platch1     equ   p3.0
platch2     equ   p3.1
strb        equ   p3.4
sclk        equ   p3.5
pdata       equ   p3.7

;---- p3.2, 3.3, 3.6 is empty now ---------

;---------- define byte ----------
b_number    equ   30h    ;store address of board to latch 01-04
b1_sec1     equ   31h
b1_sec2     equ   32h
b2_sec1     equ   33h
b2_sec2     equ   34h
data1       equ   20h    ;first  byte of bit addressable 00h-07h
data2       equ   21h    ;second byte of bit addressable 08h-0fh
                         ;but only use 4 bits 08h-0bh
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;---------- define constant value ----------
count_l     equ   0feh    ;value for timer
count_h     equ   0fh     ;value for timer

            org   000h
            jmp   main
            org   00bh
            jmp   serv_t0         ;interrupt vector for timer board1
            org   01bh
            jmp   serv_t1         ;interrupt vector for timer board2
            org   050h
main:       mov   sp,#60h
            setb  strb
            setb  sclk
            setb  pdata
            call  int_interrupt
            call  test_board
            call  off_green
            call  off_red
            call  latch_b1
            call  latch_b2
            call  delay
            call  delay

main1:      call  data2board
            call  decode_data
            jmp   main1

data2board: mov   r7,#00h
      one:  call  recieve_process
            cjne  r7,#08h,one
            mov   b_number,a
            mov   r7,#00h
      two:  call  recieve_process
            cjne  r7,#08h,two
            mov   data1,a
            mov   r7,#00h
    three:  call  recieve_process
            cjne  r7,#04h,three
            mov   data2,a
            ret

recieve_process:
            call  recieve_clk
            rrc   a
            inc   r7
            ret

recieve_clk: clr  strb
            jb    sclk,$
            mov   c,pdata
            setb  strb
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                jnb     sclk,$
                ret

;--------------- service rutines for decode data ---------
decode_data:    nop
chk_led1:       jb      00h,led1_pass
led1_fail:      clr     grn1
                setb    red1
                jmp     chk_led2
led1_pass:      clr     red1
                setb    grn1

chk_led2:       jb      01h,led2_pass
led2_fail:      clr     grn2
                setb    red2
                jmp     chk_led3
led2_pass:      clr     red2
                setb    grn2

chk_led3:       jb      02h,led3_pass
led3_fail:      clr     grn3
                setb    red3
                jmp     chk_led4
led3_pass:      clr     red3
                setb    grn3

chk_led4:       jb      03h,led4_pass
led4_fail:      clr     grn4
                setb    red4
                jmp     chk_led5
led4_pass:      clr     red4
                setb    grn4

chk_led5:       jb      04h,led5_pass
led5_fail:      clr     grn5
                setb    red5
                jmp     chk_led6
led5_pass:      clr     red5
                setb    grn5

chk_led6:       jb      05h,led6_pass
led6_fail:      clr     grn6
                setb    red6
                jmp     chk_led7
led6_pass:      clr     red6
                setb    grn6

chk_led7:       jb      06h,led7_pass
led7_fail:      clr     grn7
                setb    red7
                jmp     chk_led8
led7_pass:      clr     red7
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                setb    grn7

chk_led8:       jb      07h,led8_pass
led8_fail:      clr     grn8
                setb    red8
                jmp     chk_end8
led8_pass:      clr     red8
                setb    grn8
chk_end8:       mov     a,b_number
                cjne    a,#09h,chk_end8_again
                call    end8
                call    start_board1
                ret
chk_end8_again:
                cjne    a,#0ah,chk_led9
                call    end8
                call    start_board2
                ret

chk_led9:       jb      0ch,led9_pass
led9_fail:      clr     grn9
                setb    red9
                jmp     chk_led10
led9_pass:      clr     red9
                setb    grn9

chk_led10:      jb      0dh,led10_pass
led10_fail:     clr     grn10
                setb    red10
                jmp     chk_end10
led10_pass:     clr     red10
                setb    grn10
chk_end10:      mov     a,b_number
                cjne    a,#0bh,chk_end10_again
                call    end10
                call    start_board1
                ret
chk_end10_again:
                cjne    a,#0ch,chk_led11
                call    end10
                call    start_board2
                ret

chk_led11:      jb      0eh,led11_pass
led11_fail:     clr     grn11
                setb    red11
                jmp     chk_led12
led11_pass:     clr     red11
                setb    grn11

chk_led12:      jb      0fh,led12_pass
led12_fail:     clr     grn12
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                setb    red12
                jmp     chk_end12
led12_pass:     clr     red12
                setb    grn12
chk_end12:      mov     a,b_number
                cjne    a,#0dh,chk_end12_again
                call    start_board1
                ret
chk_end12_again:
                cjne    a,#0eh,error_b_number
                call    start_board2
                ret

error_b_number:
                call    on_all
                call    latch_b1        ;don't use time0
                call    latch_b2        ;don't use timer1
                call    delay
                call    delay
                call    test_board
                call    off_green
                call    off_red
                call    latch_b1
                call    latch_b2
                setb    strb
                setb    sclk
                setb    pdata
                clr     c
                mov     b_number,#0ffh
                mov     data1,#0ffh
                mov     data2,#0ffh
                ret

end8:           clr     grn9
                clr     red9
                clr     grn10
                clr     red10
end10:          clr     grn11
                clr     red11
                clr     grn12
                clr     red12
                ret

;------------ service rutine ------------
start_board1:   call    latch_b1
                clr     tr0
                mov     th0,#count_h
                mov     tl0,#count_l
                mov     b1_sec1,#00h
                mov     b1_sec2,#00h
                setb    tr0
                ret
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
start_board2:  call   latch_b2
               clr    tr1
               mov    th1,#count_h
               mov    tl1,#count_l
               mov    b2_sec1,#00h
               mov    b2_sec2,#00h
               setb   tr1
               ret

latch_b1:      clr    platch1
               setb   platch1
               clr    platch1
               ret

latch_b2:      clr    platch2
               setb   platch2
               clr    platch2
               ret

on_all:        mov    p0,#0ffh
               setb   grn9          ;led9
               setb   grn10         ;led10
               setb   grn11         ;led11
               setb   grn12         ;led12
               setb   red1          ;led1
               setb   red2          ;led2
               setb   red3          ;led3
               setb   red4          ;led4
               mov    p2,#0ffh      ;led5-12
               ret

on_green:      mov    p0,#0ffh      ;led 1-8
               setb   grn9          ;led9
               setb   grn10         ;led10
               setb   grn11         ;led11
               setb   grn12         ;led12
               call   off_red
               ret

on_red:        setb   red1          ;led1
               setb   red2          ;led2
               setb   red3          ;led3
               setb   red4          ;led4
               mov    p2,#0ffh      ;led5-12
               call   off_green
               ret

off_green:     mov    p0,#00h
               clr    grn9
               clr    grn10
               clr    grn11
               clr    grn12
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ret

off_red:        clr     red1
                clr     red2
                clr     red3
                clr     red4
                mov     p2,#00h
                ret

test_board:     call    on_green
                call    latch_b1        ;board 1 is green
                call    on_red
                call    latch_b2        ;board 2 is red
                call    delay
                call    on_red
                call    latch_b1        ;board 1 is red
                call    on_green
                call    latch_b2        ;board 2 is green
                call    delay
                ret

delay:          mov     r1,#00h
delay1:         mov     r2,#30h
delay2:         mov     r3,#10h
                djnz    r3,$
                djnz    r2,delay2
                djnz    r1,delay1
                ret


;------------ initial rutine ------------
int_interrupt:  setb    ea                      ;enable all interrupt.
                setb    et0                     ;enable interrupt timer0
                setb    et1                     ;enable interrupt timer1
                mov     tmod,#00010001b         ; timer0=8bit ,timer1=8bit autoreload
                ret

;--------------- interrupt service rutine ------------
serv_t0:        push    acc                     ;service rutine for timer
                mov     th0,#count_h            ;reload value of timer0 high
                mov     tl0,#count_l            ;reload value of timer0 low
                inc     b1_sec1                 ;incress value every interrupt
                mov     a,b1_sec1
chk_b1_sec1:    cjne    a,#0fh,exit_t0          ;check for 1 second
                mov     b1_sec1,#00h
                inc     b1_sec2
                mov     a,b1_sec2
chk_b1_sec2:    cjne    a,#120d,exit_t0         ;check limit time for board1
                mov     b1_sec2,#00h            ;clear buffer for board1
                mov     b1_sec1,#00h            ;clear buffer for board1
                clr     tr0
                call    on_green
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                call    on_red
                call    delay
                call    off_green       ;off all leds of board1
                call    off_red         ;off all leds of board1
                call    latch_b1        ;off all leds of board1
exit_t0:        pop     acc
                reti

serv_t1:        push    acc             ;service rutine for timer
                mov     th1,#count_h    ;reload value of timer1 high
                mov     tl1,#count_l    ;reload value of timer1 low
                inc     b2_sec1         ;incress value every interrupt
                mov     a,b2_sec1
chk_b2_sec1:    cjne    a,#0fh,exit_t1  ;check for 1 second
                mov     b2_sec1,#00h
                inc     b2_sec2
                mov     a,b2_sec2
chk_b2_sec2:    cjne    a,#120d,exit_t1 ;check limit time for board2
                clr     tr1
                mov     b2_sec2,#00h    ;clear buffer for board1
                mov     b2_sec1,#00h    ;clear buffer for board1
                call    off_green       ;off all leds of board1
                call    off_red         ;off all leds of board1
                call    latch_b2        ;off all leds of board1
exit_t1:        pop     acc
                reti

                end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค. รายละเอียดของอุปกรณ์ต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Features

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

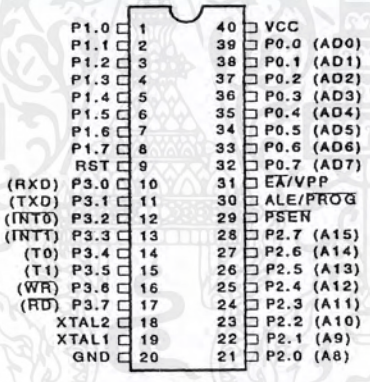
Description

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

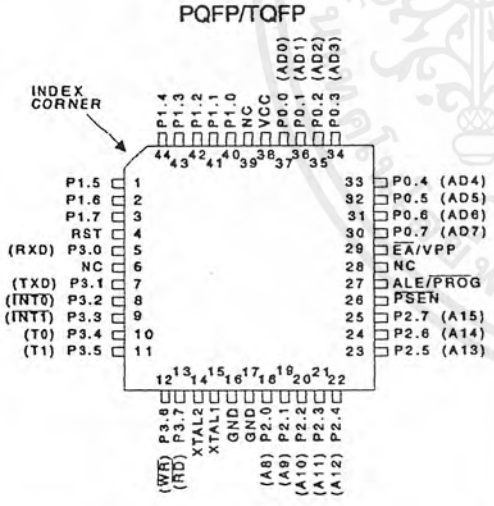
(continued)

Pin Configurations

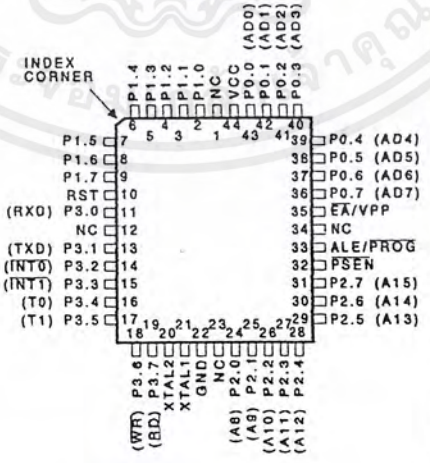
PDIP



PQFP/TQFP



PLCC



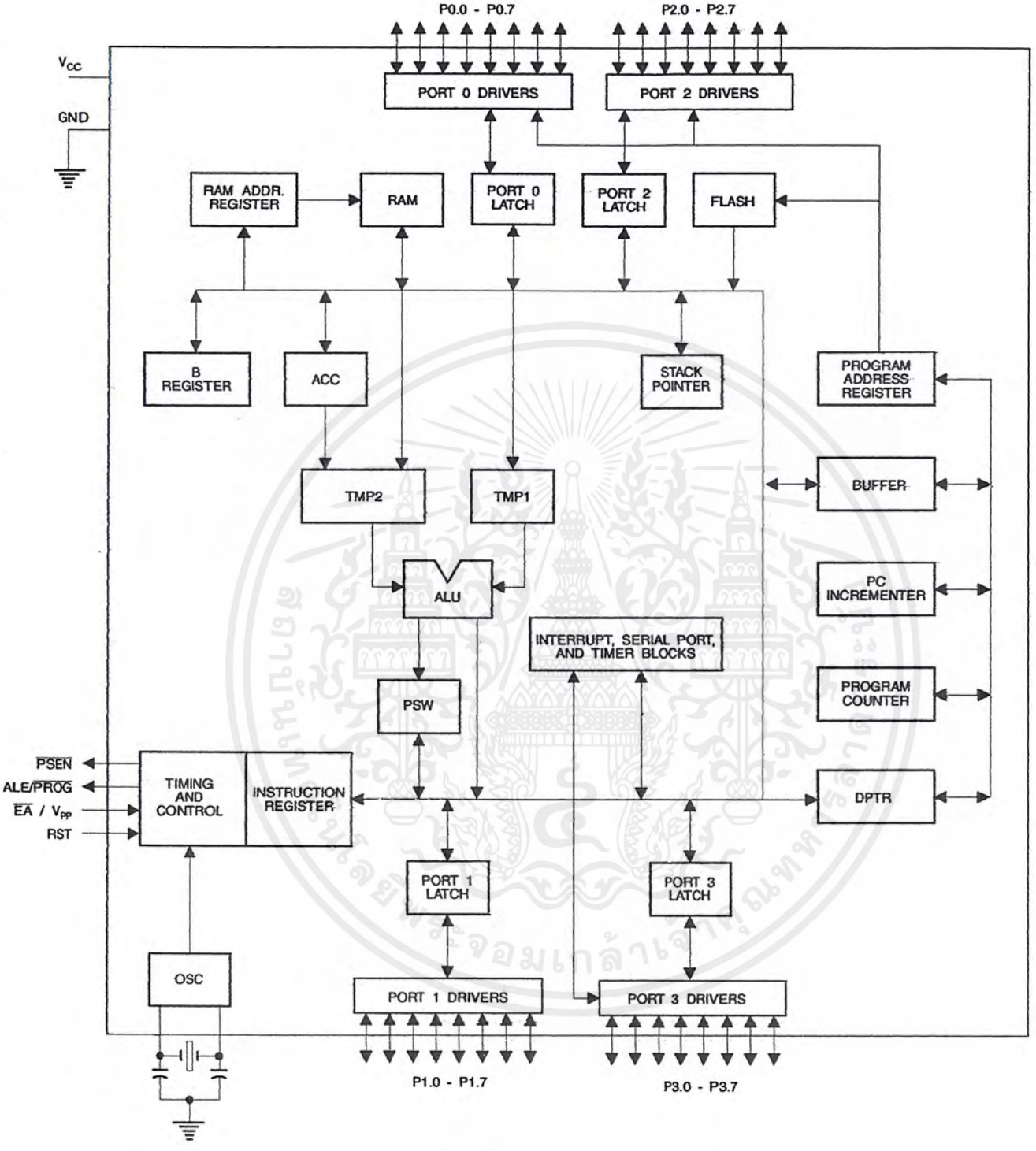
8-Bit Microcontroller with 4K Bytes Flash

AT89C51

0265F-A–12/97

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Block Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The AT89C51 provides the following standard features: 4K bytes of Flash, 128 bytes of RAM, 32 I/O lines, two 16-bit timer/counters, a five vector two-level interrupt architecture, a full duplex serial port, on-chip oscillator and clock circuitry. In addition, the AT89C51 is designed with static logic for operation down to zero frequency and supports two software selectable power saving modes. The Idle Mode stops the CPU while allowing the RAM, timer/counters, serial port and interrupt system to continue functioning. The Power Down Mode saves the RAM contents but freezes the oscillator disabling all other chip functions until the next hardware reset.

Pin Description

V_{CC}
Supply voltage.

GND
Ground.

Port 0
Port 0 is an 8-bit open drain bidirectional I/O port. As an output port each pin can sink eight TTL inputs. When 1s are written to port 0 pins, the pins can be used as high-impedance inputs.

Port 0 may also be configured to be the multiplexed low-order address/data bus during accesses to external program and data memory. In this mode P0 has internal pullups.

Port 0 also receives the code bytes during Flash programming, and outputs the code bytes during program verification. External pullups are required during program verification.

Port 1
Port 1 is an 8-bit bidirectional I/O port with internal pullups. The Port 1 output buffers can sink/source four TTL inputs. When 1s are written to Port 1 pins they are pulled high by the internal pullups and can be used as inputs. As inputs, Port 1 pins that are externally being pulled low will source current (I_{IL}) because of the internal pullups.

Port 1 also receives the low-order address bytes during Flash programming and verification.

Port 2
Port 2 is an 8-bit bidirectional I/O port with internal pullups. The Port 2 output buffers can sink/source four TTL inputs. When 1s are written to Port 2 pins they are pulled high by the internal pullups and can be used as inputs. As inputs, Port 2 pins that are externally being pulled low will source current (I_{IL}) because of the internal pullups.

Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @ DPTR). In this application it uses strong internal pullups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOVX @ RI), Port 2 emits the contents of the P2 Special Function Register.

Port 2 also receives the high-order address bits and some control signals during Flash programming and verification.

Port 3
Port 3 is an 8-bit bidirectional I/O port with internal pullups. The Port 3 output buffers can sink/source four TTL inputs. When 1s are written to Port 3 pins they are pulled high by the internal pullups and can be used as inputs. As inputs, Port 3 pins that are externally being pulled low will source current (I_{IL}) because of the pullups.

Port 3 also serves the functions of various special features of the AT89C51 as listed below:

Port Pin	Alternate Functions
P3.0	RXD (serial input port)
P3.1	TXD (serial output port)
P3.2	$\overline{INT0}$ (external interrupt 0)
P3.3	$\overline{INT1}$ (external interrupt 1)
P3.4	T0 (timer 0 external input)
P3.5	T1 (timer 1 external input)
P3.6	$\overline{WR}$ (external data memory write strobe)
P3.7	$\overline{RD}$ (external data memory read strobe)

Port 3 also receives some control signals for Flash programming and verification.

RST
Reset input. A high on this pin for two machine cycles while the oscillator is running resets the device.

ALE/$\overline{PROG}$
Address Latch Enable output pulse for latching the low byte of the address during accesses to external memory. This pin is also the program pulse input ($\overline{PROG}$) during Flash programming.

In normal operation ALE is emitted at a constant rate of 1/6 the oscillator frequency, and may be used for external timing or clocking purposes. Note, however, that one ALE pulse is skipped during each access to external Data Memory.

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

$\overline{PSEN}$
Program Store Enable is the read strobe to external program memory.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

$\overline{EA}$/V_{PP}

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

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

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

XTAL1

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

XTAL2

Output from the inverting oscillator amplifier.

Oscillator Characteristics

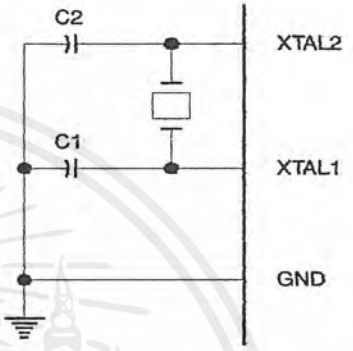
XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier which can be configured for use as an on-chip oscillator, as shown in Figure 1. Either a quartz crystal or ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected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

Idle Mode

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

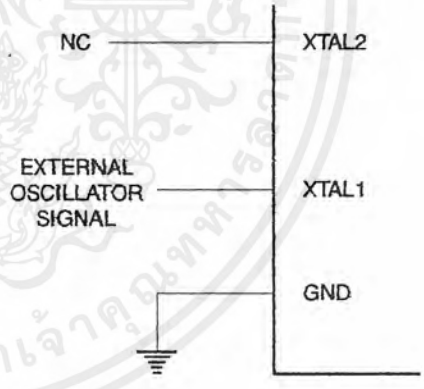
It should be noted that when idle is terminated by a hard ware reset, the device normally resumes program execution, from where it left off, up to two machine cycles before the internal reset algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to the port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

Figure 1. Oscillator Connections



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

Figure 2. External Clock Drive Configuration



Status of External Pins During Idle and Power Down Modes

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Power Down Mode

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

Program Memory Lock Bits

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

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

Lock Bit Protection Modes

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

Programming the Flash

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

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

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

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

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

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

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

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

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

Reading the Signature Bytes: The signature bytes are read by the same procedure as a normal verification of locations 030H,

031H, and 032H, except that P3.6 and P3.7 must be pulled to a logic low. The values returned are as follows.

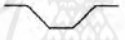
(030H) = 1EH indicates manufactured by Atmel
(031H) = 51H indicates 89C51
(032H) = FFH indicates 12V programming
(032H) = 05H indicates 5V programming

Programming Interface

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

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

Flash Programming Modes

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

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 3. Programming the Flash

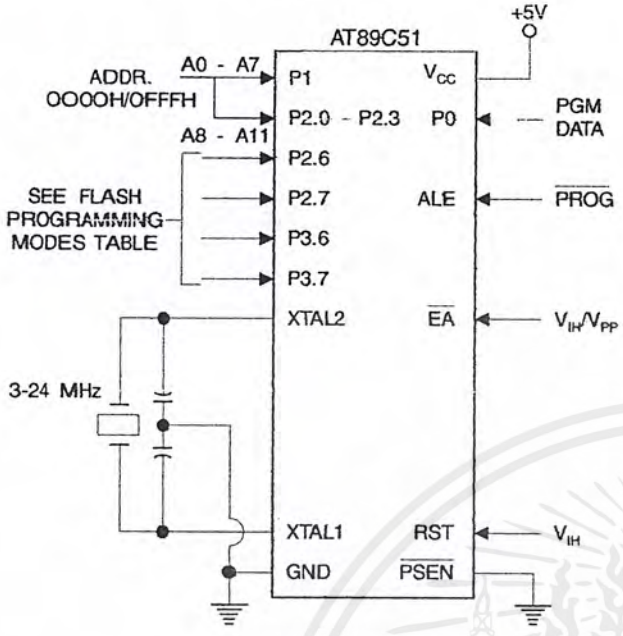
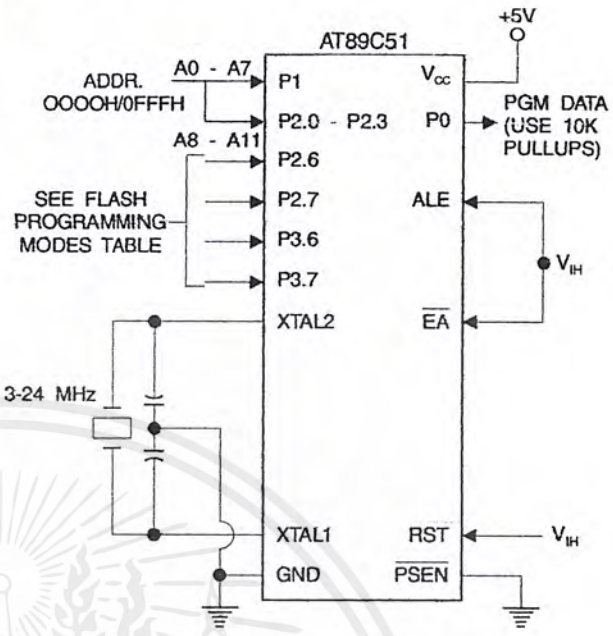


Figure 4. Verifying the Flash



Flash Programming and Verification Characteristics

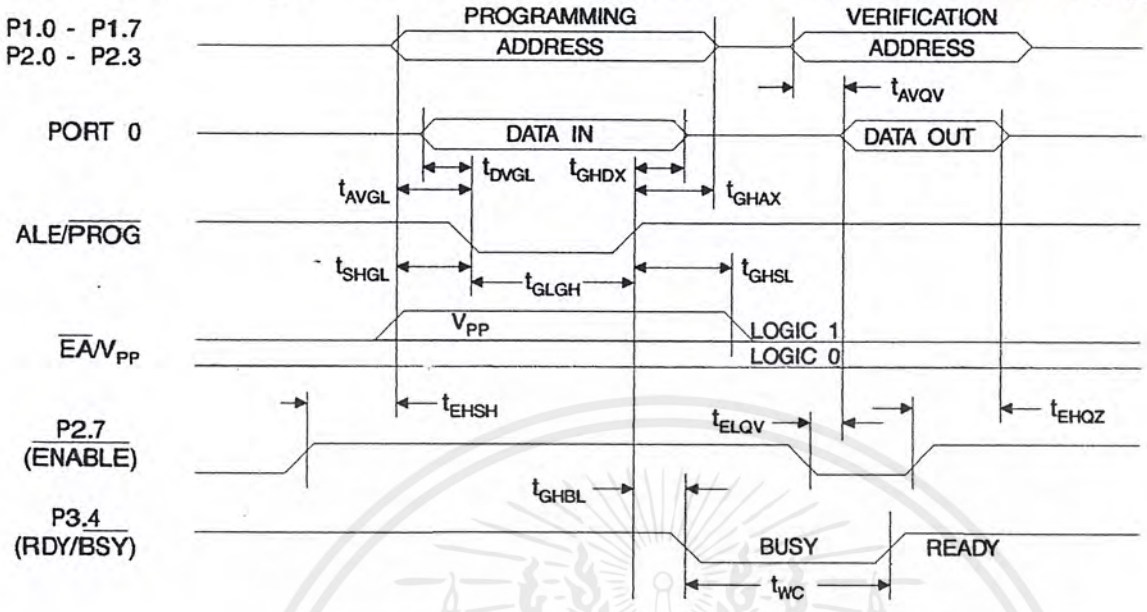
$T_A = 0°C$ to $70°C$, $V_{CC} = 5.0 \pm 10\%$

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

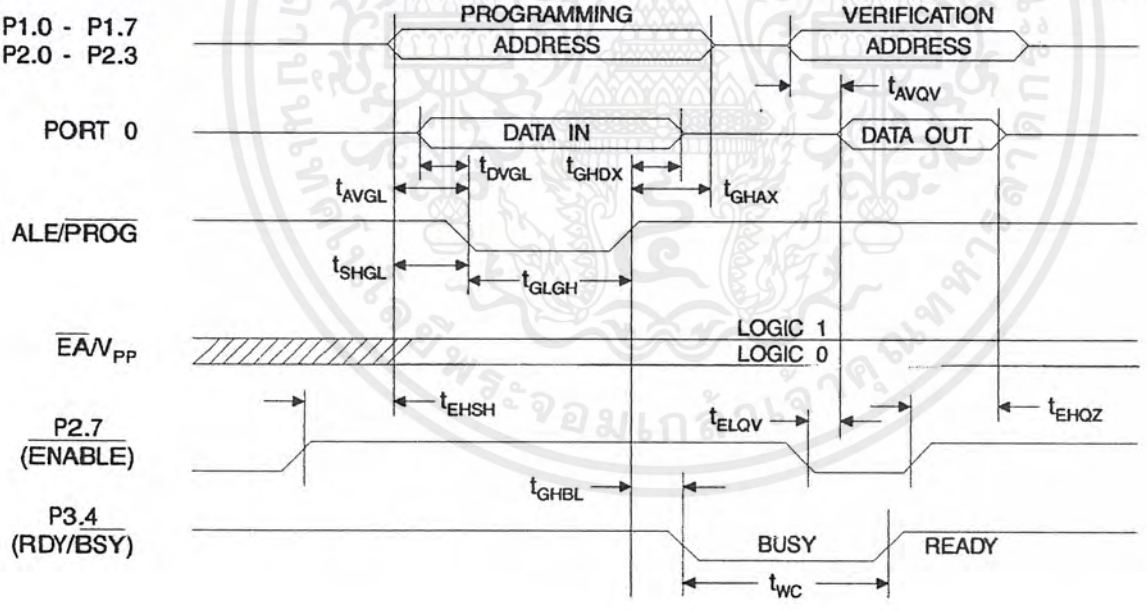
Note: 1. Only used in 12-volt programming mode.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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



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



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Absolute Maximum Ratings*

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

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

DC Characteristics

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

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

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

2. Minimum V_{CC} for Power Down is 2V.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AC Characteristics

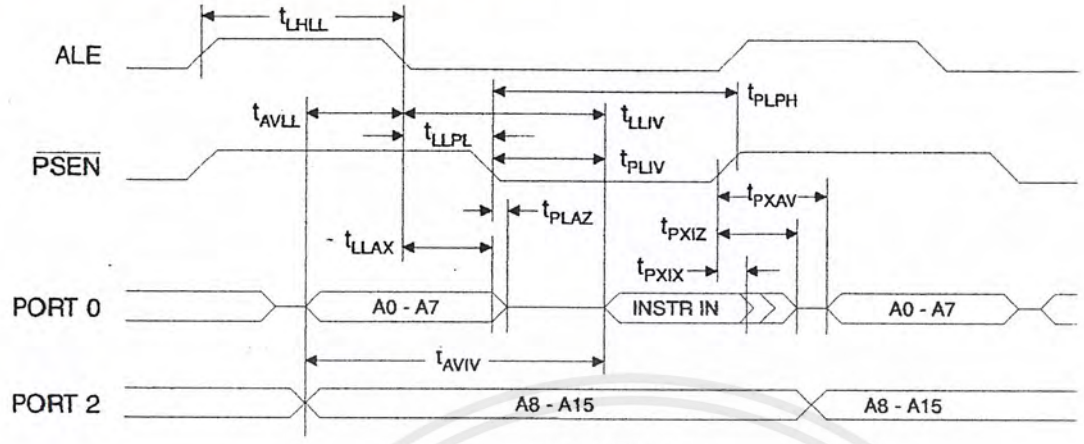
(Under Operating Conditions; Load Capacitance for Port 0, ALE/$\overline{PROG}$, and $\overline{PSEN}$ = 100 pF; Load Capacitance for all other outputs = 80 pF)

External Program and Data Memory Characteristics

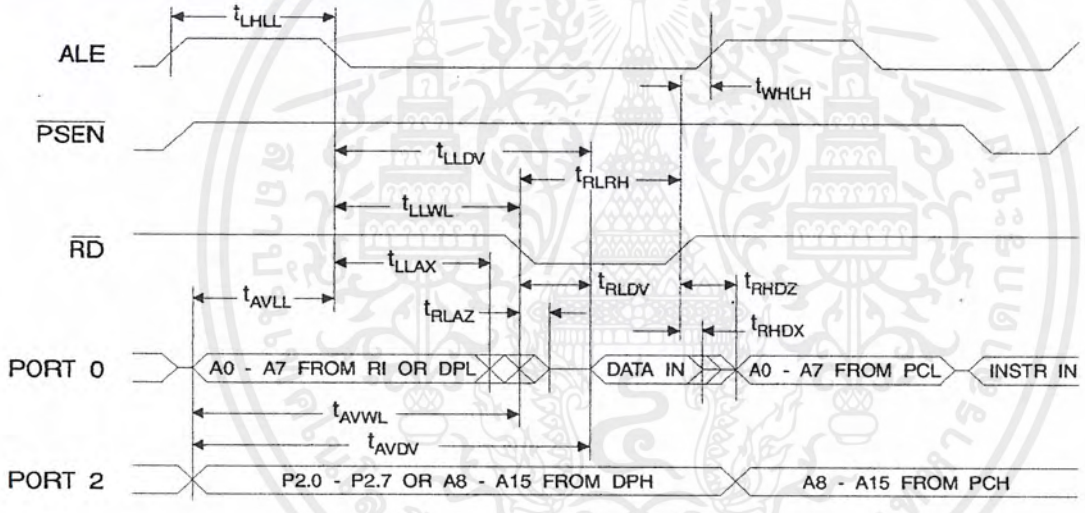
Symbol	Parameter	12 MHz Oscillator		16 to 24 MHz Oscillator		Units
		Min	Max	Min	Max	
$1/t_{CLCL}$	Oscillator Frequency			0	24	MHz
t_{LHLL}	ALE Pulse Width	127		$2t_{CLCL}$-40		ns
t_{AVLL}	Address Valid to ALE Low	43		t_{CLCL}-13		ns
t_{LLAX}	Address Hold After ALE Low	48		t_{CLCL}-20		ns
t_{LLIV}	ALE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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

External Program Memory Read Cycle

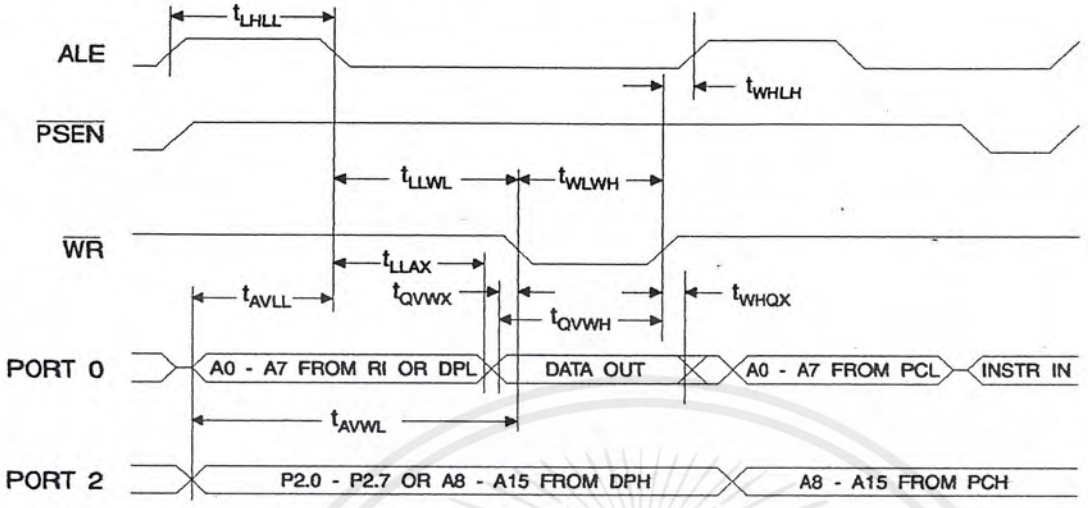


External Data Memory Read Cycle

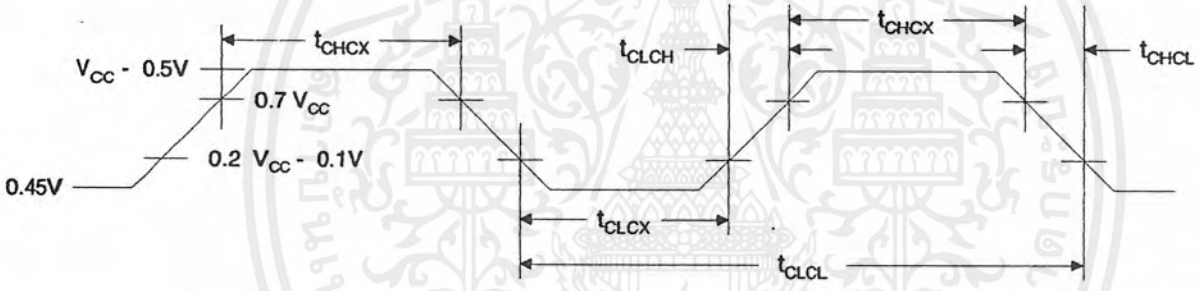


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

External Data Memory Write Cycle



External Clock Drive Waveforms



External Clock Drive

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

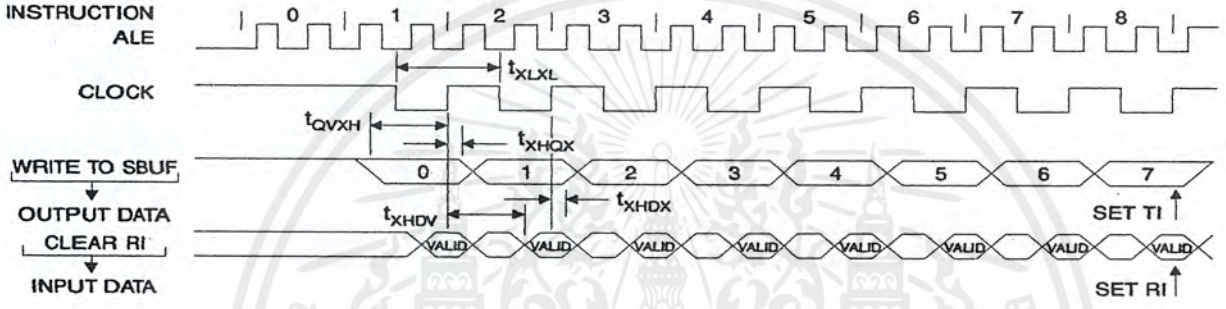
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Serial Port Timing: Shift Register Mode Test Conditions

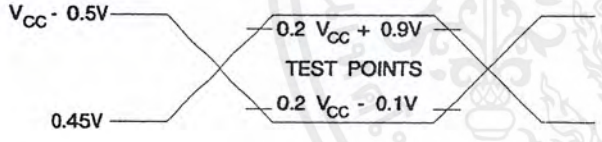
(V_{CC} = 5.0 V ± 20%; Load Capacitance = 80 pF)

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

Shift Register Mode Timing Waveforms

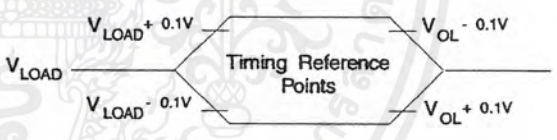


AC Testing Input/Output Waveforms[1]



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

Float Waveforms[1]



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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Ordering Information

Speed (MHz)	Power Supply	Ordering Code	Package	Operation Range
12	5V ± 20%	AT89C51-12AC	44A	Commercial
		AT89C51-12JC	44J	(0°C to 70°C)
		AT89C51-12PC	40P6	
		AT89C51-12QC	44Q	
		AT89C51-12AI	44A	Industrial
		AT89C51-12JI	44J	(-40°C to 85°C)
		AT89C51-12PI	40P6	
		AT89C51-12QI	44Q	
		AT89C51-12AA	44A	Automotive
		AT89C51-12JA	44J	(-40°C to 105°C)
		AT89C51-12PA	40P6	
		AT89C51-12QA	44Q	
16	5V ± 20%	AT89C51-16AC	44A	Commercial
		AT89C51-16JC	44J	(0°C to 70°C)
		AT89C51-16PC	40P6	
		AT89C51-16QC	44Q	
		AT89C51-16AI	44A	Industrial
		AT89C51-16JI	44J	(-40°C to 85°C)
		AT89C51-16PI	40P6	
		AT89C51-16QI	44Q	
		AT89C51-16AA	44A	Automotive
		AT89C51-16JA	44J	(-40°C to 105°C)
		AT89C51-16PA	40P6	
		AT89C51-16QA	44Q	
20	5V ± 20%	AT89C51-20AC	44A	Commercial
		AT89C51-20JC	44J	(0°C to 70°C)
		AT89C51-20PC	40P6	
		AT89C51-20QC	44Q	
		AT89C51-20AI	44A	Industrial
		AT89C51-20JI	44J	(-40°C to 85°C)
		AT89C51-20PI	40P6	
		AT89C51-20QI	44Q	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Ordering Information

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

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MICROCHIP

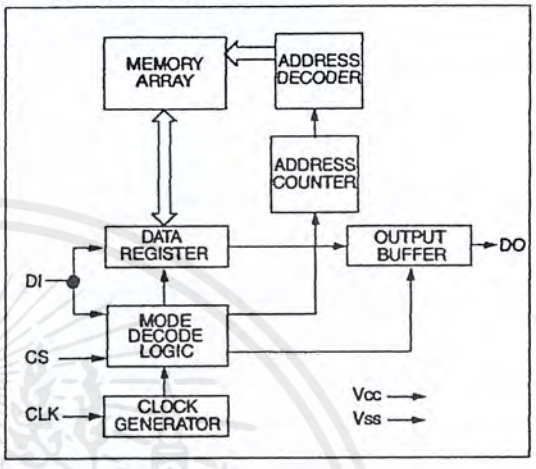
93LC66A/B

4K 2.5V Microwire® Serial EEPROM

FEATURES

- Single supply with operation down to 2.5V
- Low power CMOS technology
  - 1 mA active current (typical)
  - 1 µA standby current (maximum)
- 512 x 8 bit organization (93LC66A)
- 256 x 16 bit organization (93LC66B)
- Self-timed ERASE and WRITE cycles (including auto-erase)
- Automatic ERAL before WRAL
- Power on/off data protection circuitry
- Industry standard 3-wire serial interface
- Device status signal during ERASE/WRITE cycles
- Sequential READ function
- 1,000,000 E/W cycles guaranteed
- Data retention > 200 years
- 8-pin PDIP/SOIC and 8-pin TSSOP packages
- Available for the following temperature ranges:
  - Commercial (C): 0°C to +70°C
  - Industrial (I): -40°C to +85°C

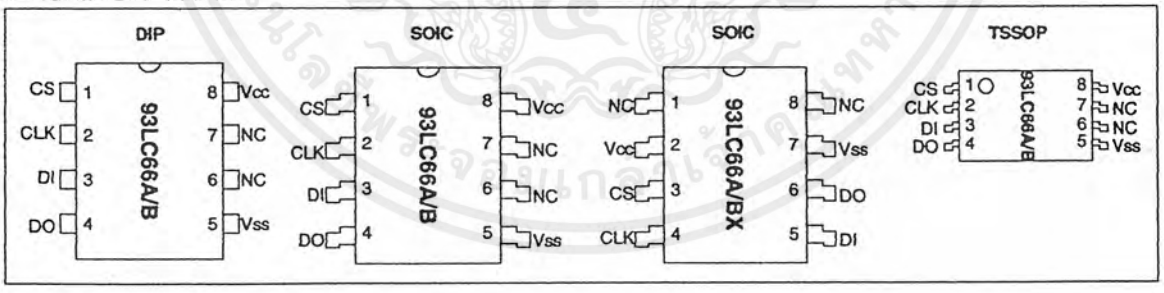
BLOCK DIAGRAM



DESCRIPTION

The Microchip Technology Inc. 93LC66A/B are 4K-bit, low voltage serial Electrically Erasable PROMs. The device memory is configured as x8 (93LC66A) or x16 bits (93LC66B). Advanced CMOS technology makes these devices ideal for low power nonvolatile memory applications. The 93LC66A/B is available in standard 8-pin DIP, surface mount SOIC, and TSSOP packages. The 93LC66AX/BX are only offered in a 150-mil SOIC package.

PACKAGE TYPE



Microwire is a registered trademark of Motorola.

© 1998 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.0 ELECTRICAL CHARACTERISTICS

1.1 Maximum Ratings*

Vcc 7.0V
All inputs and outputs w.r.t. Vss .....-0.6V to Vcc +1.0V
Storage temperature .....-65°C to +150°C
Ambient temp. with power applied.....-65°C to +125°C
Soldering temperature of leads (10 seconds) .....+300°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.

TABLE 1-1 PIN FUNCTION TABLE

Name	Function
CS	Chip Select
CLK	Serial Data Clock
DI	Serial Data Input
DO	Serial Data Output
Vss	Ground
NC	No Connect
Vcc	Power Supply

TABLE 1-2 DC AND AC ELECTRICAL CHARACTERISTICS

All parameters apply over the specified operating ranges unless otherwise noted	Commercial (C): Vcc = +2.5V to +6.0V Tamb = 0°C to +70°C Industrial (I): Vcc = +2.5V to +6.0V Tamb = -40°C to +85°C				
Parameter	**Symbol**	**Min.**	**Max.**	**Units**	**Conditions**
High level input voltage	VIH1	2.0	Vcc +1	V	2.7V ≤ Vcc ≤ 5.5V (Note 2)
	VIH2	0.7 Vcc	Vcc +1	V	Vcc < 2.7V
Low level input voltage	VIL1	-0.3	0.8	V	Vcc > 2.7V (Note 2)
	VIL2	-0.3	0.2 Vcc	V	Vcc < 2.7V
Low level output voltage	VOL1	—	0.4	V	IOL = 2.1 µA; Vcc = 4.5V
	VOL2	—	0.2	V	IOL =100 µA; Vcc = Vcc Min.
High level output voltage	VOH1	2.4	—	V	IOH = -400 µA; Vcc = 4.5V
	VOH2	Vcc-0.2	—	V	IOH = -100 µA; Vcc = Vcc Min.
Input leakage current	ILI	-10	10	µA	VIN = Vss to Vcc
Output leakage current	ILO	-10	10	µA	VOUT = Vss to Vcc
Pin capacitance (all inputs/outputs)	CIN, COUT	—	7	pF	VIN/VOUT = 0 V (Notes 1 & 2) Tamb = +25°C, FCLK = 1 MHz
Operating current	ICC read	—	1 500	mA µA	FCLK = 2 MHz; Vcc = 6.0V FCLK = 1 MHz; Vcc = 3.0V
	ICC write	—	1.5	mA	
Standby current	ICCS	—	1	µA	CS = Vss
Clock frequency	FCLK	—	2 1	MHz MHz	Vcc > 4.5V Vcc < 4.5V
Clock high time	TCKH	250	—	ns	
Clock low time	TCKL	250	—	ns	
Chip select setup time	TCSS	50	—	ns	Relative to CLK
Chip select hold time	TCSH	0	—	ns	Relative to CLK
Chip select low time	TCSL	250	—	ns	
Data input setup time	TDIS	100	—	ns	Relative to CLK
Data input hold time	TDIH	100	—	ns	Relative to CLK
Data output delay time	TPD	—	400	ns	CL = 100 pF
Data output disable time	TCZ	—	100	ns	CL = 100 pF (Note 2)
Status valid time	TSV	—	500	ns	CL = 100 pF
Program cycle time	TWC	—	6	ms	ERASE/WRITE mode
	TEC	—	6	ms	ERAL mode
	TWL	—	15	ms	WRAL mode
Endurance	—	1M	—	cycles	25°C, Vcc = 5.0V, Block Mode (Note 3)

Note 1: This parameter is tested at Tamb = 25°C and Fclk = 1 MHz.
2: This parameter is periodically sampled and not 100% tested.
3: This application is not tested but guaranteed by characterization. For endurance estimates in a specific application, please consult the Total Endurance Model which may be obtained on Microchip's BBS or website.

 © 1997 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.0 PIN DESCRIPTION

2.1 Chip Select (CS)

A high level selects the device; a low level deselects the device and forces it into standby mode. However, a programming cycle which is already in progress will be completed, regardless of the Chip Select (CS) input signal. If CS is brought low during a program cycle, the device will go into standby mode as soon as the programming cycle is completed.

CS must be low for 250 ns minimum (TCSL) between consecutive instructions. If CS is low, the internal control logic is held in a RESET status.

2.2 Serial Clock (CLK)

The Serial Clock (CLK) is used to synchronize the communication between a master device and the 93LC66A/B. Opcode, address, and data bits are clocked in on the positive edge of CLK. Data bits are also clocked out on the positive edge of CLK.

CLK can be stopped anywhere in the transmission sequence (at high or low level) and can be continued anytime with respect to clock high time (TCKH) and clock low time (TCKL). This gives the controlling master freedom in preparing opcode, address, and data.

CLK is a "Don't Care" if CS is low (device deselected). If CS is high, but a START condition has not been detected, any number of clock cycles can be received by the device without changing its status (i.e., waiting for a START condition).

CLK cycles are not required during the self-timed WRITE (i.e., auto ERASE/WRITE) cycle.

After detection of a START condition the specified number of clock cycles (respectively low to high transitions of CLK) must be provided. These clock cycles are required to clock in all required opcode, address, and data bits before an instruction is executed (Table 2-1 and Table 2-2). CLK and DI then become don't care inputs waiting for a new START condition to be detected.

2.3 Data In (DI)

Data In (DI) is used to clock in a START bit, opcode, address, and data synchronously with the CLK input.

2.4 Data Out (DO)

Data Out (DO) is used in the READ mode to output data synchronously with the CLK input (TPD after the positive edge of CLK).

This pin also provides READY/$\overline{\text{BUSY}}$ status information during ERASE and WRITE cycles. READY/$\overline{\text{BUSY}}$ status information is available on the DO pin if CS is brought high after being low for minimum chip select low time (TCSL) and an ERASE or WRITE operation has been initiated.

The status signal is not available on DO, if CS is held low during the entire ERASE or WRITE cycle. In this case, DO is in the HIGH-Z mode. If status is checked after the ERASE/WRITE cycle, the data line will be high to indicate the device is ready.

TABLE 2-1 INSTRUCTION SET FOR 93LC66A

Instruction	SB	Opcode	Address									Data In	Data Out	Req. CLK Cycles
ERASE	1	11	A8	A7	A6	A5	A4	A3	A2	A1	A0	—	(RDY/$\overline{\text{BSY}}$)	12
ERAL	1	00	1	0	X	X	X	X	X	X	X	—	(RDY/$\overline{\text{BSY}}$)	12
EWDS	1	00	0	0	X	X	X	X	X	X	X	—	HIGH-Z	12
EWEN	1	00	1	1	X	X	X	X	X	X	X	—	HIGH-Z	12
READ	1	10	A8	A7	A6	A5	A4	A3	A2	A1	A0	—	D7 - D0	20
WRITE	1	01	A8	A7	A6	A5	A4	A3	A2	A1	A0	D7 - D0	(RDY/$\overline{\text{BSY}}$)	20
WRAL	1	00	0	1	X	X	X	X	X	X	X	D7 - D0	(RDY/$\overline{\text{BSY}}$)	20

TABLE 2-2 INSTRUCTION SET FOR 93LC66B

Instruction	SB	Opcode	Address								Data In	Data Out	Req. CLK Cycles
ERASE	1	11	A7	A6	A5	A4	A3	A2	A1	A0	—	(RDY/$\overline{\text{BSY}}$)	11
ERAL	1	00	1	0	X	X	X	X	X	X	—	(RDY/$\overline{\text{BSY}}$)	11
EWDS	1	00	0	0	X	X	X	X	X	X	—	HIGH-Z	11
EWEN	1	00	1	1	X	X	X	X	X	X	—	HIGH-Z	11
READ	1	10	A7	A6	A5	A4	A3	A2	A1	A0	—	D15 - D0	27
WRITE	1	01	A7	A6	A5	A4	A3	A2	A1	A0	D15 - D0	(RDY/$\overline{\text{BSY}}$)	27
WRAL	1	00	0	1	X	X	X	X	X	X	D15 - D0	(RDY/$\overline{\text{BSY}}$)	27

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.0 FUNCTIONAL DESCRIPTION

Instructions, addresses, and write data are clocked into the DI pin on the rising edge of the clock (CLK). The DO pin is normally held in a HIGH-Z state except when reading data from the device, or when checking the READY/$\overline{\text{BUSY}}$ status during a programming operation. The READY/$\overline{\text{BUSY}}$ status can be verified during an ERASE/WRITE operation by polling the DO pin; DO low indicates that programming is still in progress, while DO high indicates the device is ready. The DO will enter the HIGH-Z state on the falling edge of the CS.

3.1 START Condition

The START bit is detected by the device if CS and DI are both high with respect to the positive edge of CLK for the first time.

Before a START condition is detected, CS, CLK, and DI may change in any combination (except to that of a START condition), without resulting in any device operation (ERASE, ERAL, EWDS, EWEN, READ, WRITE, and WRAL). As soon as CS is high, the device is no longer in the standby mode.

An instruction following a START condition will only be executed if the required amount of opcodes, addresses, and data bits for any particular instruction is clocked in.

After execution of an instruction (i.e., clock in or out of the last required address or data bit) CLK and DI become don't care bits until a new START condition is detected.

3.2 Data In(DI) Data Out (DO)

It is possible to connect the Data In (DI)and Data Out (DO)pins together. However, with this configuration, if A0 is a logic-high level, it is possible for a "bus conflict" to occur during the "dummy zero" that precedes the READ operation. Under such a condition the voltage level seen at Data Out is undefined and will depend upon the relative impedances of Data Out and the signal source driving A0. The higher the current sourcing capability of A0, the higher the voltage at the Data Out pin.

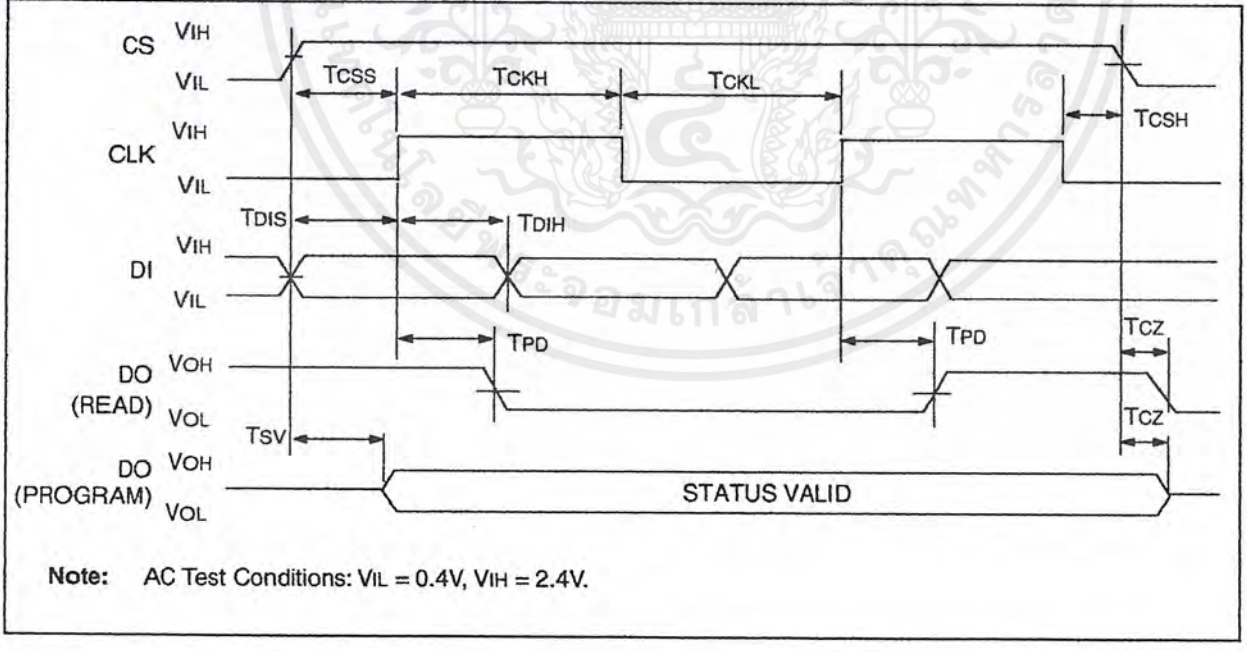
3.3 Data Protection

During power-up, all programming modes of operation are inhibited until Vcc has reached a level greater than 2.2V. During power-down, the source data protection circuitry acts to inhibit all programming modes when Vcc has fallen below 2.2V at nominal conditions.

The ERASE/WRITE Disable (EWDS) and ERASE/WRITE Enable (EWEN) commands give additional protection against accidentally programming during normal operation.

After power-up, the device is automatically in the EWDS mode. Therefore, an EWEN instruction must be performed before any ERASE or WRITE instruction can be executed.

FIGURE 3-1: SYNCHRONOUS DATA TIMING



Note: AC Test Conditions: V_{IL} = 0.4V, V_{IH} = 2.4V.

 © 1997 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 ERASE

The ERASE instruction forces all data bits of the specified address to the logical "1" state. CS is brought low following the loading of the last address bit. This falling edge of the CS pin initiates the self-timed programming cycle.

The DO pin indicates the READY/$\overline{\text{BUSY}}$ status of the device if CS is brought high after a minimum of 250 ns low (TCSL). DO at logical "0" indicates that programming is still in progress. DO at logical "1" indicates that the register at the specified address has been erased and the device is ready for another instruction.

3.5 Erase All (ERAL)

The Erase All (ERAL) instruction will erase the entire memory array to the logical "1" state. The ERAL cycle is identical to the ERASE cycle except for the different opcode. The ERAL cycle is completely self-timed and commences at the falling edge of the CS. Clocking of the CLK pin is not necessary after the device has entered the ERAL cycle.

The DO pin indicates the READY/$\overline{\text{BUSY}}$ status of the device if CS is brought high after a minimum of 250 ns low (TCSL) and before the entire ERAL cycle is complete.

FIGURE 3-2: ERASE TIMING

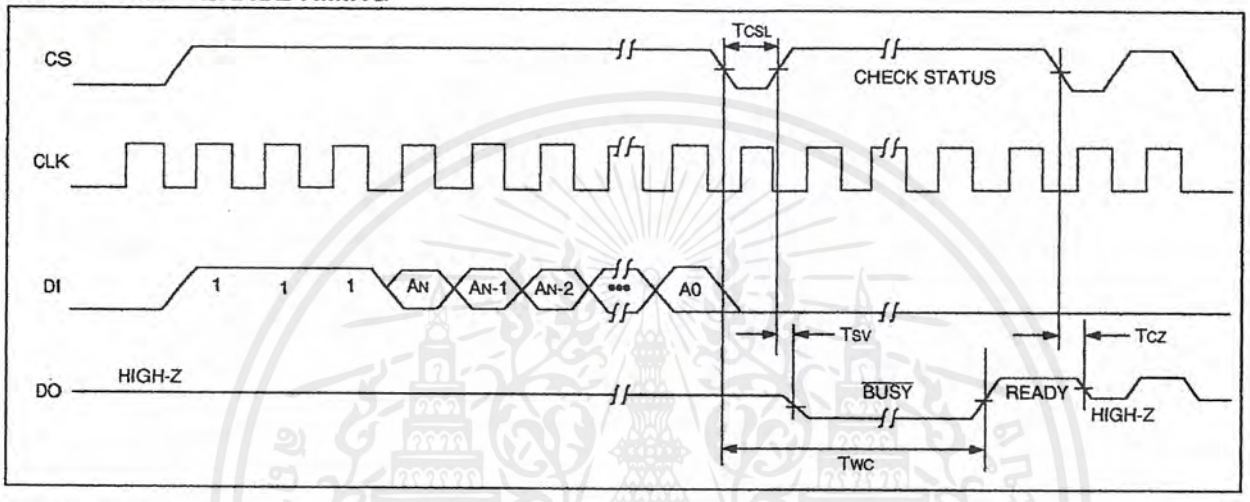
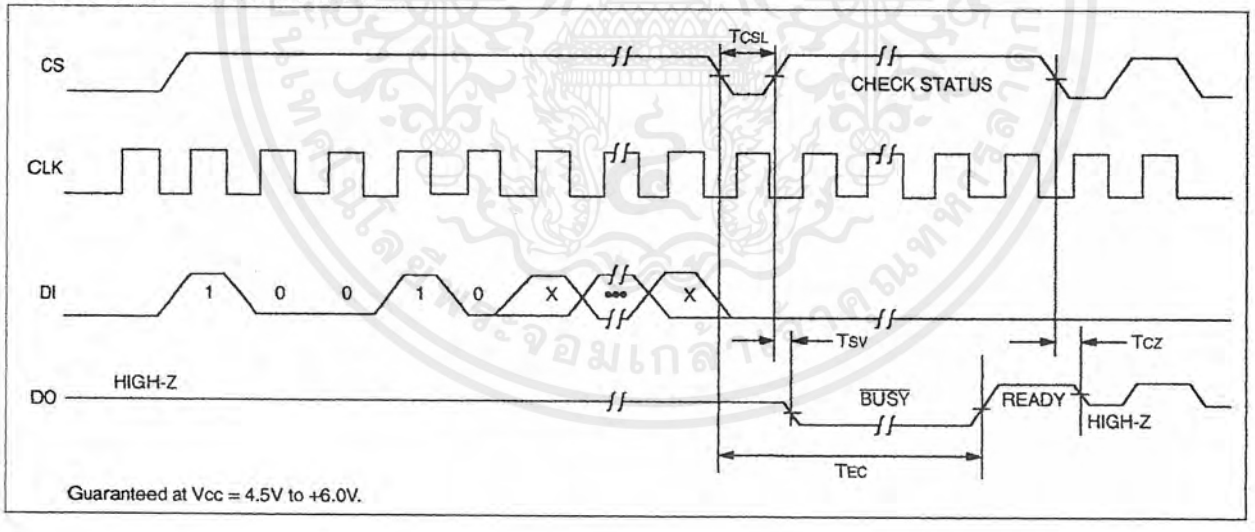


FIGURE 3-3: ERAL TIMING



Guaranteed at Vcc = 4.5V to +6.0V.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6 ERASE/WRITE Disable and Enable (EWDS/EWEN)

The 93LC66A/B powers up in the ERASE/WRITE Disable (EWDS) state. All programming modes must be preceded by an ERASE/WRITE Enable (EWEN) instruction. Once the EWEN instruction is executed, programming remains enabled until an EWDS instruction is executed or Vcc is removed from the device. To protect against accidental data disturbance, the EWDS instruction can be used to disable all ERASE/WRITE functions and should follow all programming operations. Execution of a READ instruction is independent of both the EWDS and EWEN instructions.

3.7 READ

The READ instruction outputs the serial data of the addressed memory location on the DO pin. A dummy zero bit precedes the 8-bit (93LC66A) or 16-bit (93LC66B) output string. The output data bits will toggle on the rising edge of the CLK and are stable after the specified time delay (TPD). Sequential read is possible when CS is held high. The memory data will automatically cycle to the next register and output sequentially.

FIGURE 3-4: EWDS TIMING

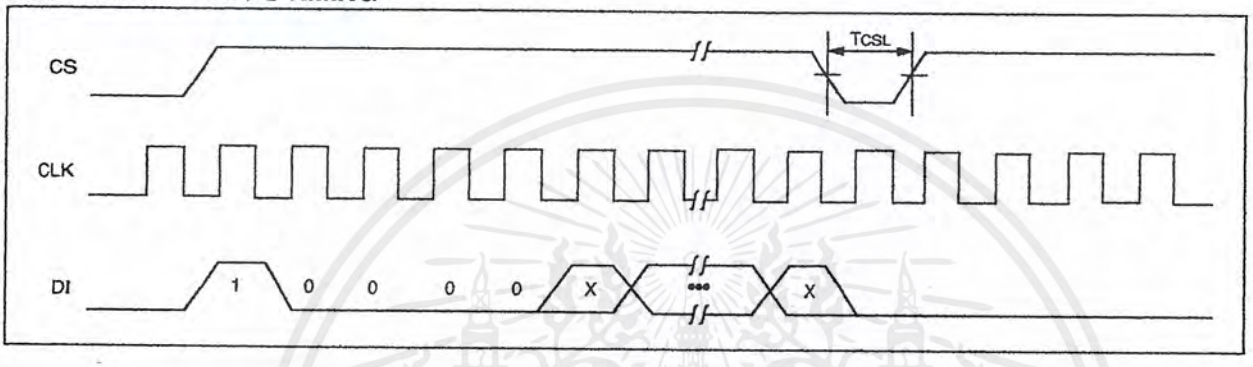


FIGURE 3-5: EWEN TIMING

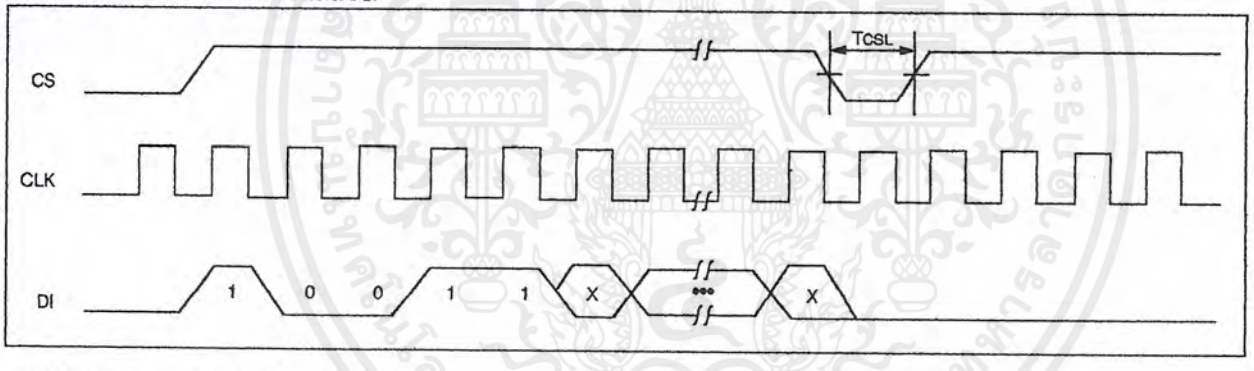
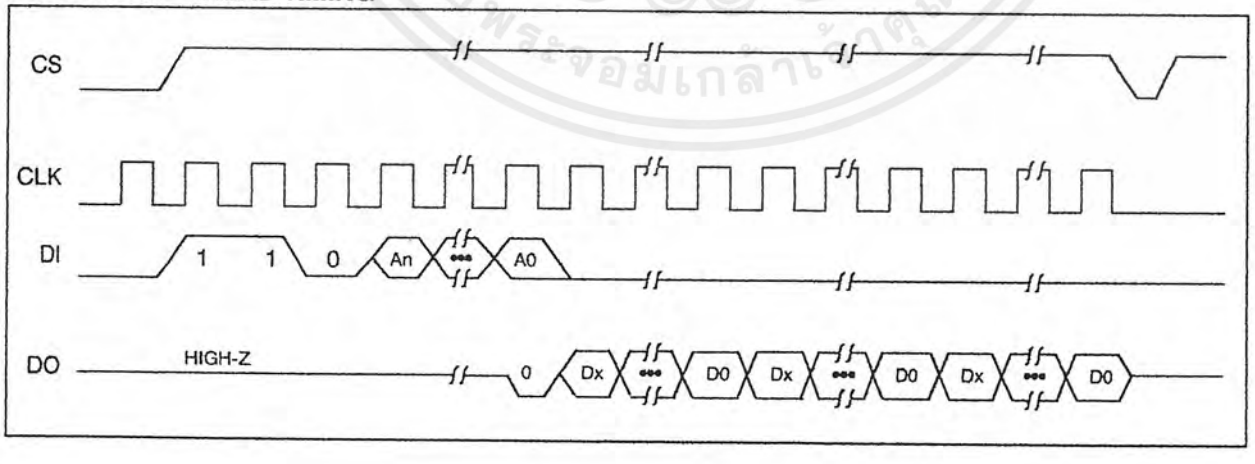


FIGURE 3-6: READ TIMING



 © 1997 Microchip Technology Inc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.8 WRITE

The WRITE instruction is followed by 8 bits (93LC66A) or 16 bits (93LC66B) of data which are written into the specified address. After the last data bit is put on the DI pin, the falling edge of CS initiates the self-timed auto-erase and programming cycle.

The DO pin indicates the READY/$\overline{\text{BUSY}}$ status of the device, if CS is brought high after a minimum of 250 ns low (TCSL) and before the entire write cycle is complete. DO at logical "0" indicates that programming is still in progress. DO at logical "1" indicates that the register at the specified address has been written with the data specified and the device is ready for another instruction.

3.9 Write All (WRAL)

The Write All (WRAL) instruction will write the entire memory array with the data specified in the command. The WRAL cycle is completely self-timed and commences at the falling edge of the CS. Clocking of the CLK pin is not necessary after the device has entered the WRAL cycle. The WRAL command does include an automatic ERAL cycle for the device. Therefore, the WRAL instruction does not require an ERAL instruction, but the chip must be in the EWEN status.

The DO pin indicates the READY/$\overline{\text{BUSY}}$ status of the device if CS is brought high after a minimum of 250 ns low (TCSL).

FIGURE 3-7: WRITE TIMING

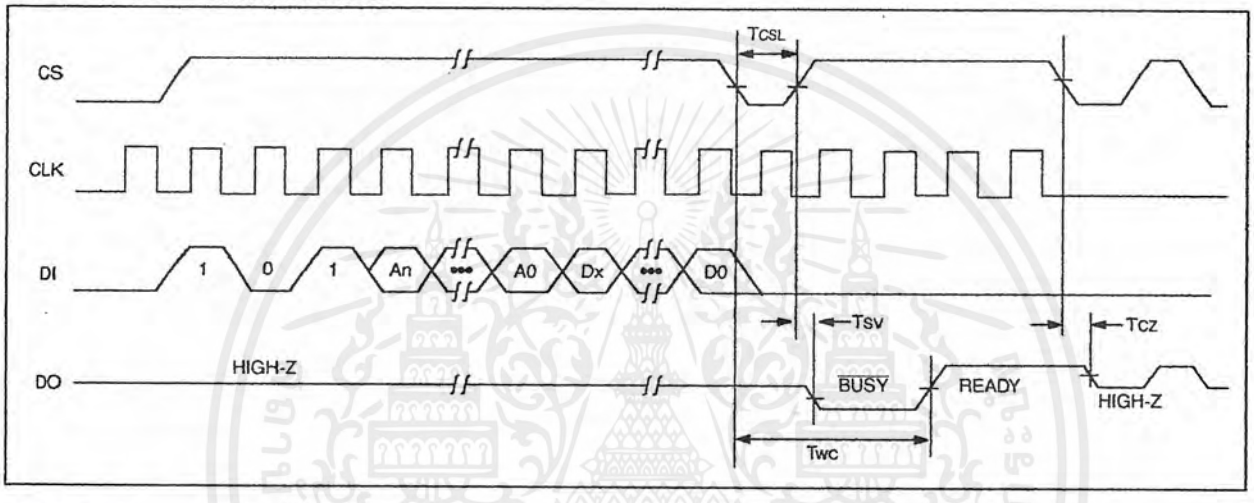
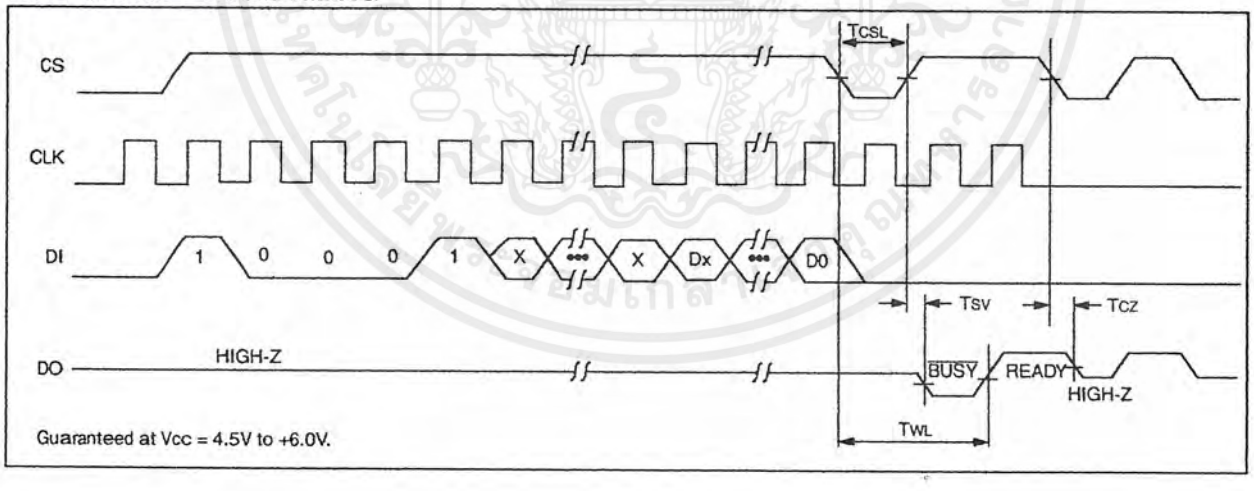


FIGURE 3-8: WRAL TIMING

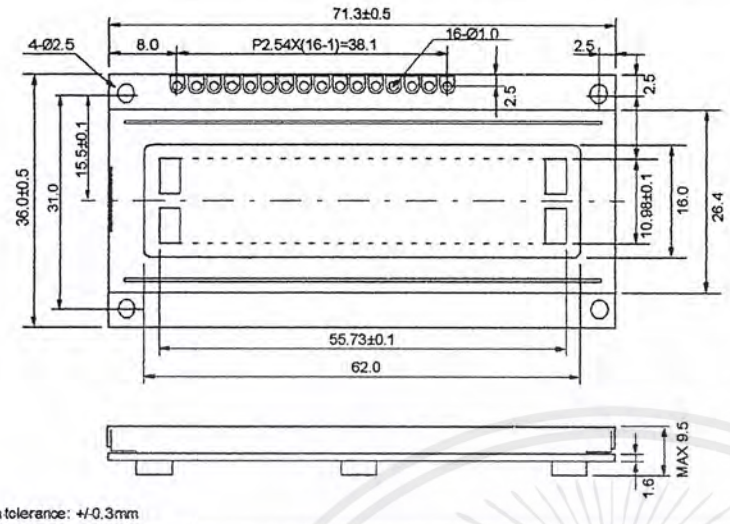


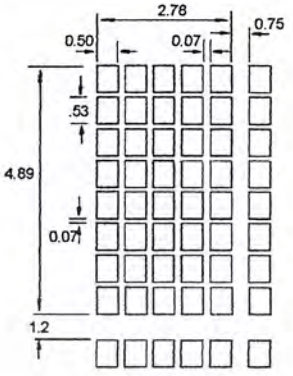
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

HDM16216H-C

Dimensional Drawing

16 Character x 2 Lines, Short Length PCB





Dimension tolerance: +/-0.3mm

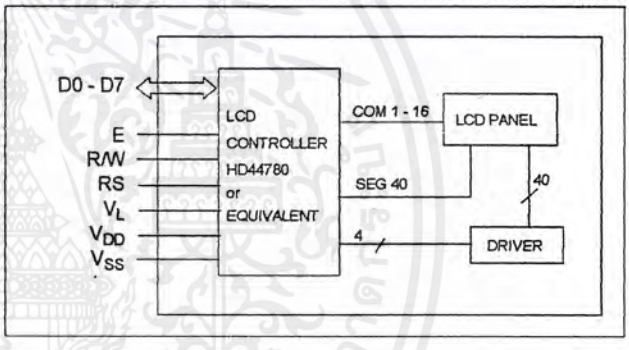
Features

Character Format .....5x7 Dots with Cursor
Backlight.....EL Optional
Options.....Gray STN/Yellow STN, 12 o'Clock/6 o'Clock View
Normal/Extended Temperature
Normal/Negative Displays

Physical Data

Module Size.....71.5W x 36.0H x 9.5T mm
Viewing Area Size.....62.0W x 16.0H mm
Weight.....26.3g

Block Diagram



Absolute Maximum Ratings

PARAMETER	SYMBOL	MIN	MAX	UNIT
SUPPLY VOLTAGE	V_{DD}-V_{SS}	-0.3	6.5	V
SUPPLY VOLTAGE FOR LCD	V_{DD}-V_L	0	13.5	V
INPUT VOLTAGE	V_{IN}	-0.3	V_{DD}	V
OPERATING TEMPERATURE	T_{OP}	0	50	℃
STORAGE TEMPERATURE	T_{STG}	-20	70	℃

Pin Connections

PIN NO.	SYMBOL	LEVEL	FUNCTION
1	V_{SS}	-	0V — Power Supply
2	V_{DD}	-	5V — Power Supply
3	V_L	-	- — Power Supply
4	RS	H/L	H: Data input L: Instruction data input
5	R/W	H/L	H: Data read L: Data write
6	E	H,H→L	Enable signal
7	D0	H/L	Data Bus
8	D1	H/L	Data Bus
9	D2	H/L	Data Bus
10	D3	H/L	Data Bus
11	D4	H/L	Data Bus
12	D5	H/L	Data Bus
13	D6	H/L	Data Bus
14	D7	H/L	Data Bus
15			EL Backlight connections
16			EL Backlight connections

Electrical Characteristics (VDD=5.0±0.25V 25℃)

PARAMETER	SYM	CONDITION	MIN	TYP	MAX	UNIT
INPUT HIGH VOLTAGE	V_{IH}	-	2.2	-	-	V
INPUT LOW VOLTAGE	V_{IL}	-	-	-	.06	V
OUTPUT HIGH VOLTAGE	V_{OH}	I_{OH}=0.2mA	2.4	-	-	V
OUTPUT LOW VOLTAGE	V_{OL}	I_{OL}=1.2mA	-	-	0.4	V
POWER SUPPLY CURRENT	I_{DD}	V_{DD}=5.0V	-	-	2.8	mA
POWER SUPPLY FOR LCD	V_{DD}-V_L	TA=25℃	4.3	4.8	-	V
DRIVE METHOD	1/16 Duty					

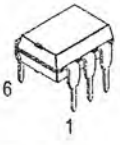
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

UL CSA SETI BABT

M4N26

6-Pin DIP Optoisolators Transistor Output

STYLE 1 PLASTIC



STANDARD THRU HOLE

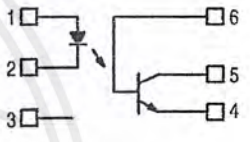
The M4N26 device consists of a gallium arsenide infrared emitting diode optically coupled to a monolithic silicon phototransistor detector.

- Most Economical Optoisolator Choice for Medium Speed, Switching Applications
- Meets or Exceeds All JEDEC Registered Specifications

Applications

- General Purpose Switching Circuits
- Interfacing and coupling systems of different potentials and impedances
- I/O Interfacing
- Solid State Relays

SCHEMATIC



PIN 1. LED ANODE
2. LED CATHODE
3. N.C.
4. EMITTER
5. COLLECTOR
6. BASE

MAXIMUM RATINGS (T_A = 25°C unless otherwise noted)

Rating	Symbol	Value	Unit
INPUT LED			
Reverse Voltage	V_R	3	Volts
Forward Current — Continuous	I_F	60	mA
LED Power Dissipation @ T_A = 25°C with Negligible Power in Output Detector	P_D	100	mW
Derate above 25°C		1.41	mW/°C
OUTPUT TRANSISTOR			
Collector–Emitter Voltage	V_{CEO}	30	Volts
Emitter–Collector Voltage	V_{ECO}	7	Volts
Collector–Base Voltage	V_{CBO}	70	Volts
Collector Current — Continuous	I_C	50	mA
Detector Power Dissipation @ T_A = 25°C with Negligible Power in Input LED	P_D	150	mW
Derate above 25°C		1.76	mW/°C
TOTAL DEVICE			
Isolation Surge Voltage(1) (Peak ac Voltage, 60 Hz, 1 sec Duration)	V_{ISO}	7500	Vac(pk)
Total Device Power Dissipation @ T_A = 25°C	P_D	250	mW
Derate above 25°C		2.94	mW/°C
Ambient Operating Temperature Range(2)	T_A	−55 to +100	°C
Storage Temperature Range(2)	T_{stg}	−55 to +150	°C
Soldering Temperature (10 sec, 1/16″ from case)	T_L	260	°C

1. Isolation surge voltage is an internal device dielectric breakdown rating. For this test, Pins 1 and 2 are common, and Pins 4, 5 and 6 are common.
2. Refer to Quality and Reliability Section in Opto Data Book for information on test conditions.

© Motorola, Inc. 1997

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ELECTRICAL CHARACTERISTICS (T_A = 25°C unless otherwise noted)(1)

Characteristic		Symbol	Min	Typ(1)	Max	Unit
INPUT LED						
Forward Voltage (I_F = 10 mA)	T_A = 25°C T_A = −55°C T_A = 100°C	V_F	— — —	1.15 1.3 1.05	1.5 — —	Volts
Reverse Leakage Current (V_R = 3 V)		I_R	—	—	100	µA
Capacitance (V = 0 V, f = 1 MHz)		C_J	—	18	—	pF
OUTPUT TRANSISTOR						
Collector–Emitter Dark Current (V_{CE} = 10 V, T_A = 25°C)		I_{CEO}	—	1	50	nA
(V_{CE} = 10 V, T_A = 100°C)		I_{CEO}	—	1	—	µA
Collector–Base Dark Current (V_{CB} = 10 V)		I_{CBO}	—	0.2	—	nA
Collector–Emitter Breakdown Voltage (I_C = 1 mA)		$V_{(BR)CEO}$	30	45	—	Volts
Collector–Base Breakdown Voltage (I_C = 100 µA)		$V_{(BR)CBO}$	70	100	—	Volts
Emitter–Collector Breakdown Voltage (I_E = 100 µA)		$V_{(BR)ECO}$	7	7.8	—	Volts
Collector–Emitter Capacitance (f = 1 MHz, V_{CE} = 0)		C_{CE}	—	7	—	pF
Collector–Base Capacitance (f = 1 MHz, V_{CB} = 0)		C_{CB}	—	19	—	pF
Emitter–Base Capacitance (f = 1 MHz, V_{EB} = 0)		C_{EB}	—	9	—	pF
COUPLED						
Output Collector Current (I_F = 10 mA, V_{CE} = 10 V)		I_C (CTR)(2)	2 (20)	7 (70)	—	mA (%)
Collector-Emitter Saturation Voltage (I_C = 2 mA, I_F = 50 mA)		$V_{CE(sat)}$	—	0.15	0.5	Volts
Turn-On Time (I_F = 10 mA, V_{CC} = 10 V, R_L = 100 Ω)(3)		t_{on}	—	2.8	—	µs
Turn-Off Time (I_F = 10 mA, V_{CC} = 10 V, R_L = 100 Ω)(3)		t_{off}	—	4.5	—	µs
Rise Time (I_F = 10 mA, V_{CC} = 10 V, R_L = 100 Ω)(3)		t_r	—	2	—	µs
Fall Time (I_F = 10 mA, V_{CC} = 10 V, R_L = 100 Ω)(3)		t_f	—	2	—	µs
Isolation Voltage (f = 60 Hz, t = 1 sec)(4)		V_{ISO}	7500	—	—	Vac(pk)
Isolation Resistance (V = 500 V)(4)		R_{ISO}	10^{11}	—	—	Ω
Isolation Capacitance (V = 0 V, f = 1 MHz)(4)		C_{ISO}	—	0.2	—	pF

1. Always design to the specified minimum/maximum electrical limits (where applicable).
2. Current Transfer Ratio (CTR) = I_C/I_F x 100%.
3. For test circuit setup and waveforms, refer to Figure 14.
4. For this test, Pins 1 and 2 are common, and Pins 4, 5 and 6 are common.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ในเชิงพาณิชย์ ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

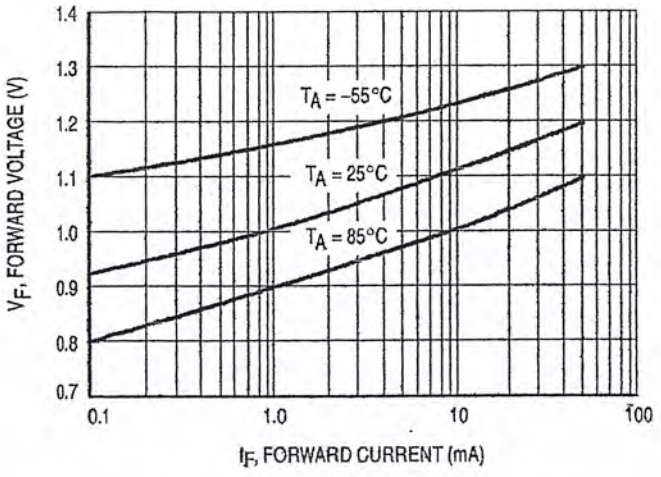


Figure 1. Forward Voltage vs. Forward Current

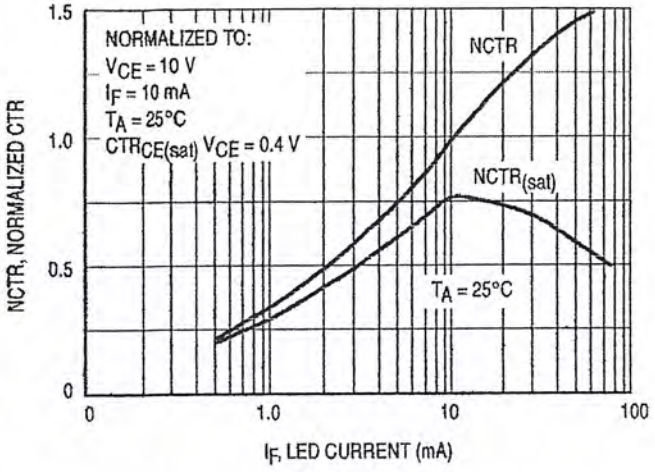


Figure 2. Normalized Non–Saturated and Saturated CTR, T_A = 25°C vs. LED Current

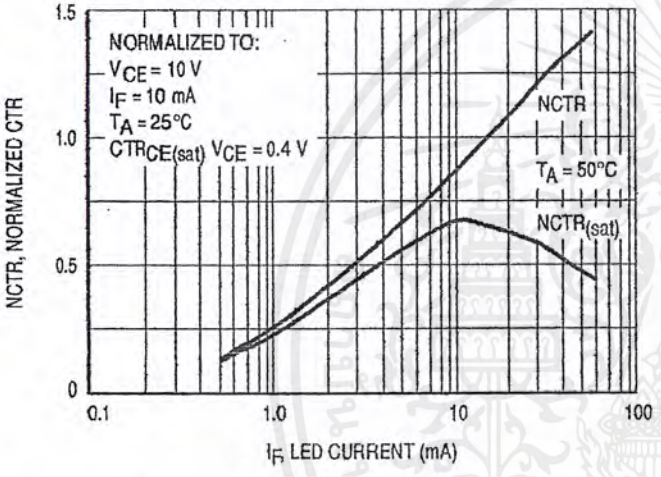


Figure 3. Normalized Non–Saturated and Saturated CTR, T_A = 50°C vs. LED Current

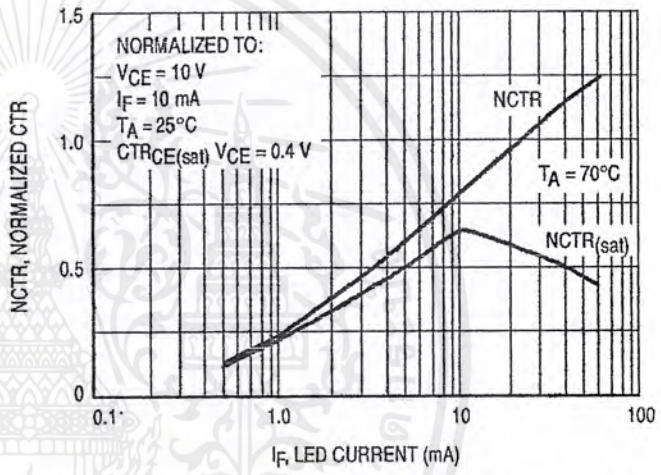


Figure 4. Normalized Non–Saturated and Saturated CTR, T_A = 70°C vs. LED Current

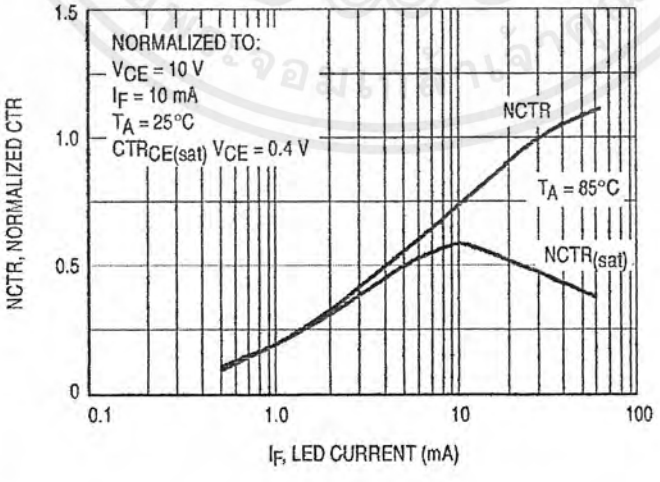


Figure 5. Normalized Non–Saturated and Saturated CTR, T_A = 85°C vs. LED Current

เอกสารนี้จัดทำขึ้นสำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

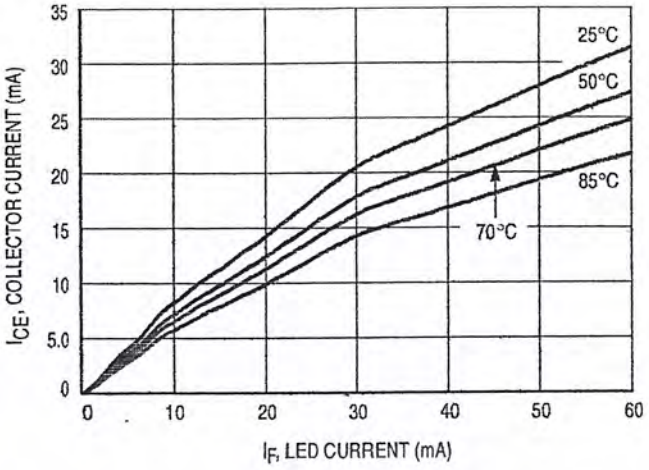


Figure 6. Collector–Emitter Current vs. Temperature and LED Current

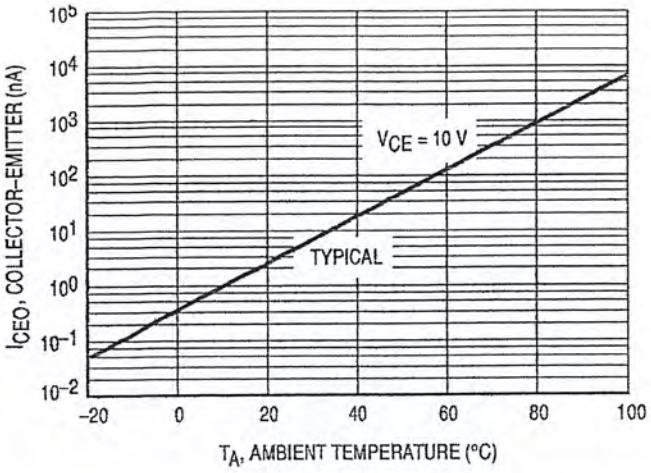


Figure 7. Collector–Emitter Leakage Current vs. Temperature

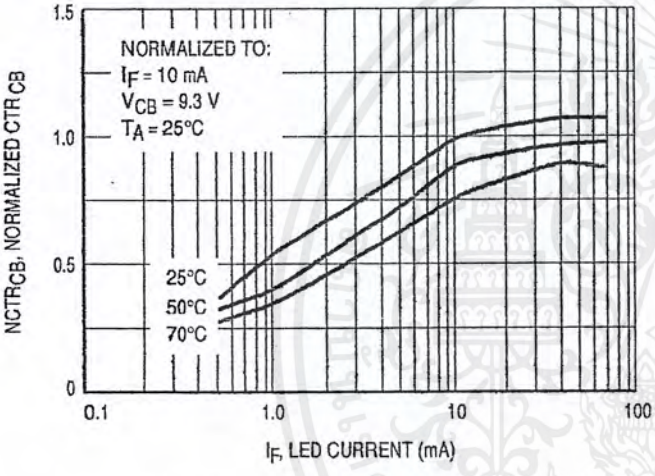


Figure 8. Normalized CTRcb vs. LED Current and Temperature

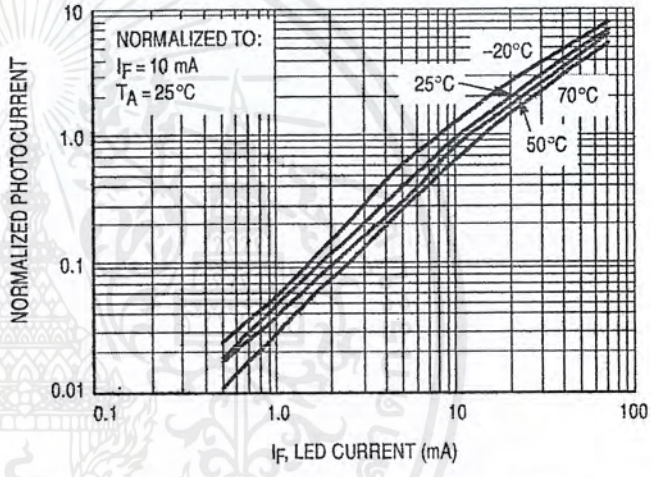


Figure 9. Normalized Photocurrent vs. I_F and Temperature

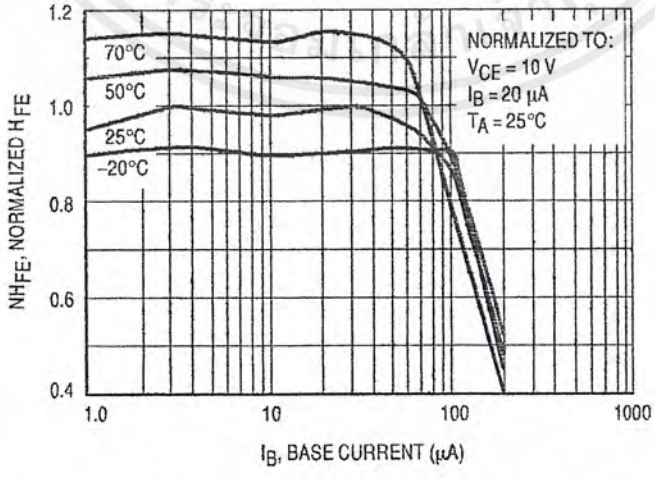


Figure 10. Normalized Non–Saturated H_{FE} vs. Base Current and Temperature

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ในเชิงพาณิชย์ ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

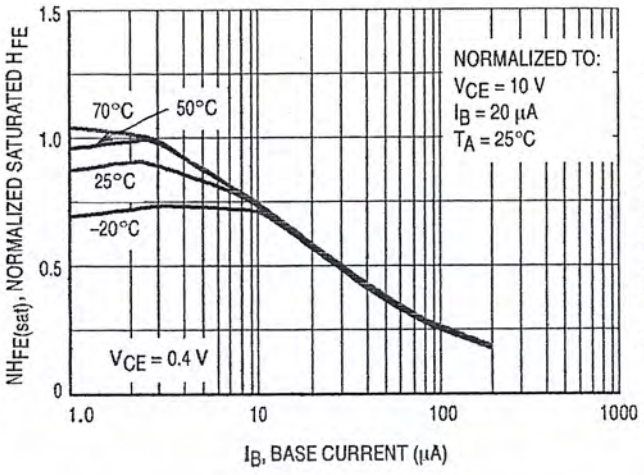


Figure 11. Normalized H_{FE} vs. Base Current and Temperature

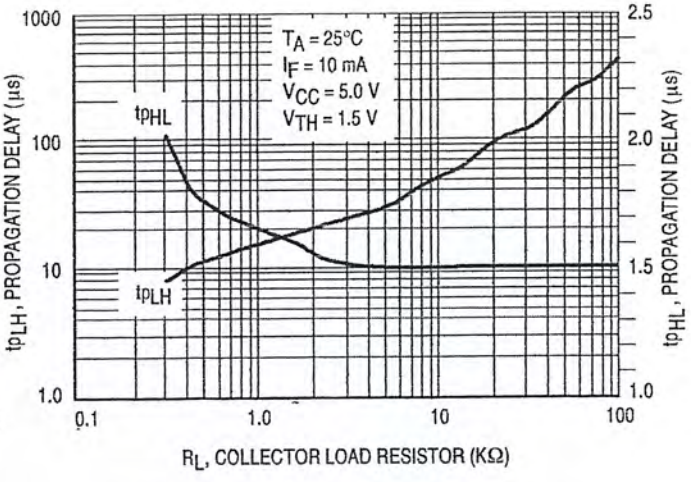


Figure 12. Propagation Delay vs. Collector Load Resistor

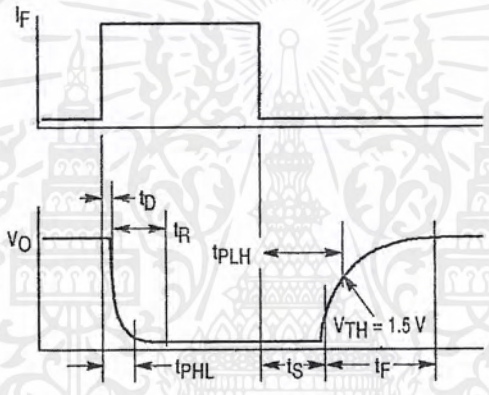


Figure 13. Switching Timing

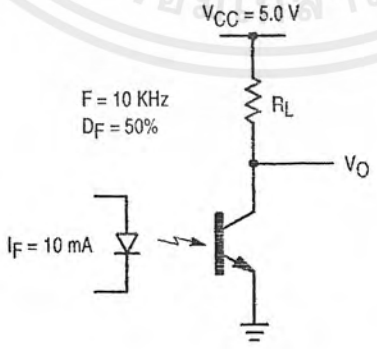
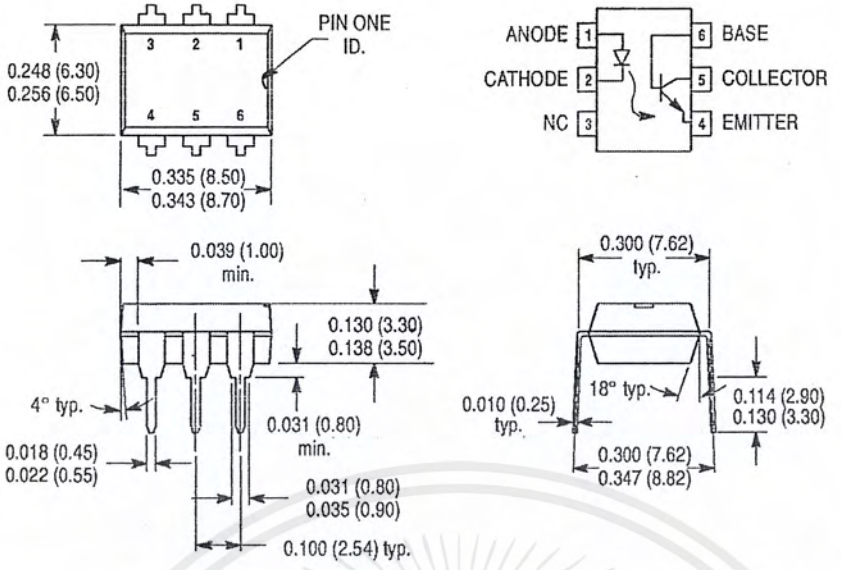


Figure 14. Switching Schematic

สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Package Dimensions in Inches (mm)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ในเชิงพาณิชย์ ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Motorola reserves the right to make changes without further notice to any products herein. Motorola makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Motorola assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters which may be provided in Motorola data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. Motorola does not convey any license under its patent rights nor the rights of others. Motorola products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Motorola product could create a situation where personal injury or death may occur. Should Buyer purchase or use Motorola products for any such unintended or unauthorized application, Buyer shall indemnify and hold Motorola and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Motorola was negligent regarding the design or manufacture of the part. Motorola and Ⓜ are registered trademarks of Motorola, Inc. Motorola, Inc. is an Equal Opportunity/Affirmative Action Employer.

เอกสารนี้จัดทำขึ้นสำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Mfax is a trademark of Motorola, Inc.

How to reach us:
USA/EUROPE/Locations Not Listed: Motorola Literature Distribution;
P.O. Box 5405, Denver, Colorado 80217. 303–675–2140 or 1–800–441–2447

Mfax™: RMFAX0@email.sps.mot.com – TOUCHTONE 602–244–6609
– US & Canada ONLY 1–800–774–1848

INTERNET: http://www.mot.com/SPS/

JAPAN: Nippon Motorola Ltd.; Tatsumi–SPD–JLDC, 6F Seibu–Butsuryu–Center,
3–14–2 Tatsumi Koto–Ku, Tokyo 135, Japan. 81–3–3521–8315

ASIA/PACIFIC: Motorola Semiconductors H.K. Ltd.; 8B Tai Ping Industrial Park,
51 Ting Kok Road, Tai Po, N.T., Hong Kong. 852–26629298

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

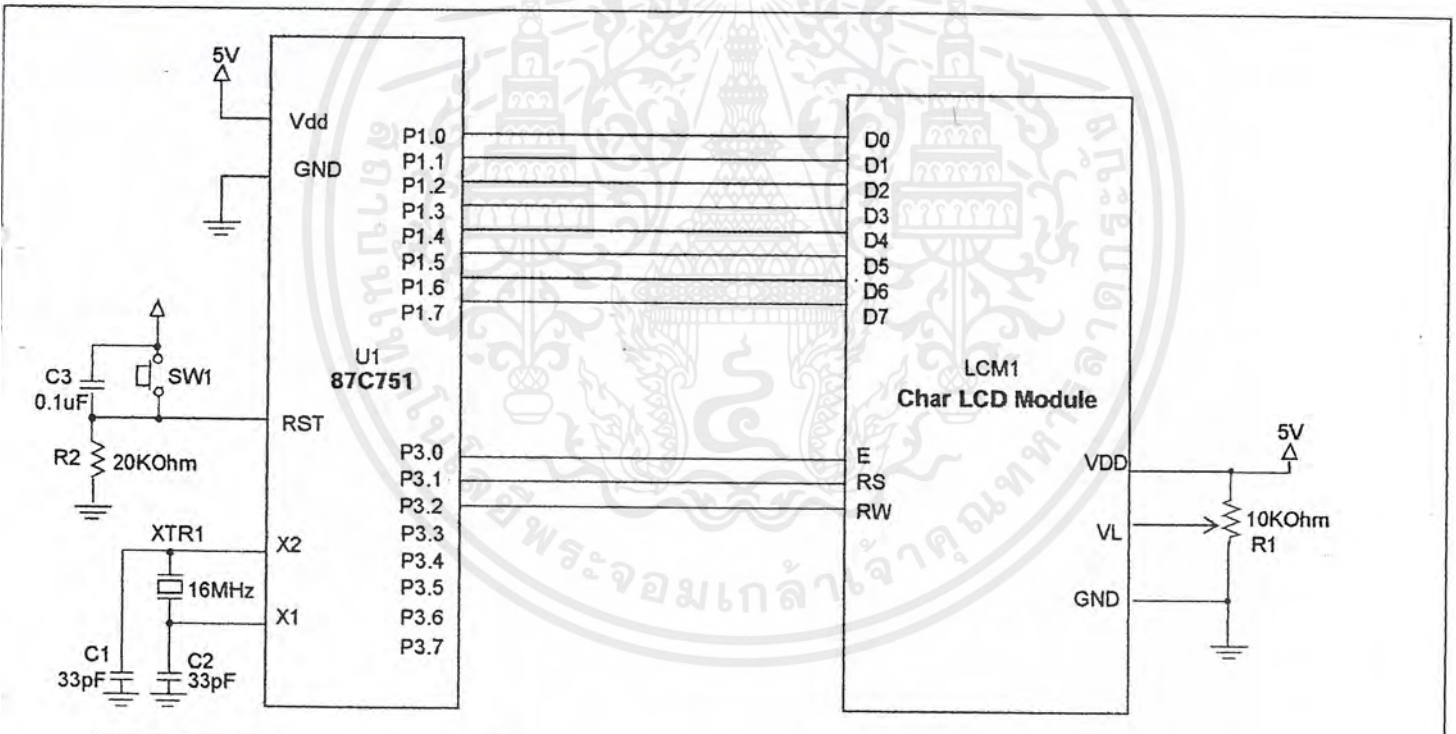
Displaying Characters on an LCD Character Module

I. Introduction:

This application note describes a simple technique to display characters from both the internal character generator and user designed characters on an LCD character module.The controlling microcontroller is a Phillips 87C751, a derivitive of the popular Intel 8051. The LCD module is connected to the microcontroller through its I/O ports. It could also be connected directly to the data bus with the addition of address decoding logic.

The process of displaying character to this module is divided into three steps. First the module must be initialized. This sets up the built-in LCD controller chip. Second, some user designed characters are uploaded to the CGRAM. This allows the displaying of up to 8 custom characters in addition to the 192 character permanently stored in the module. Lastly, a message consisting of a mix of standard ASCII characters and custom designed characters is displayed on the module.

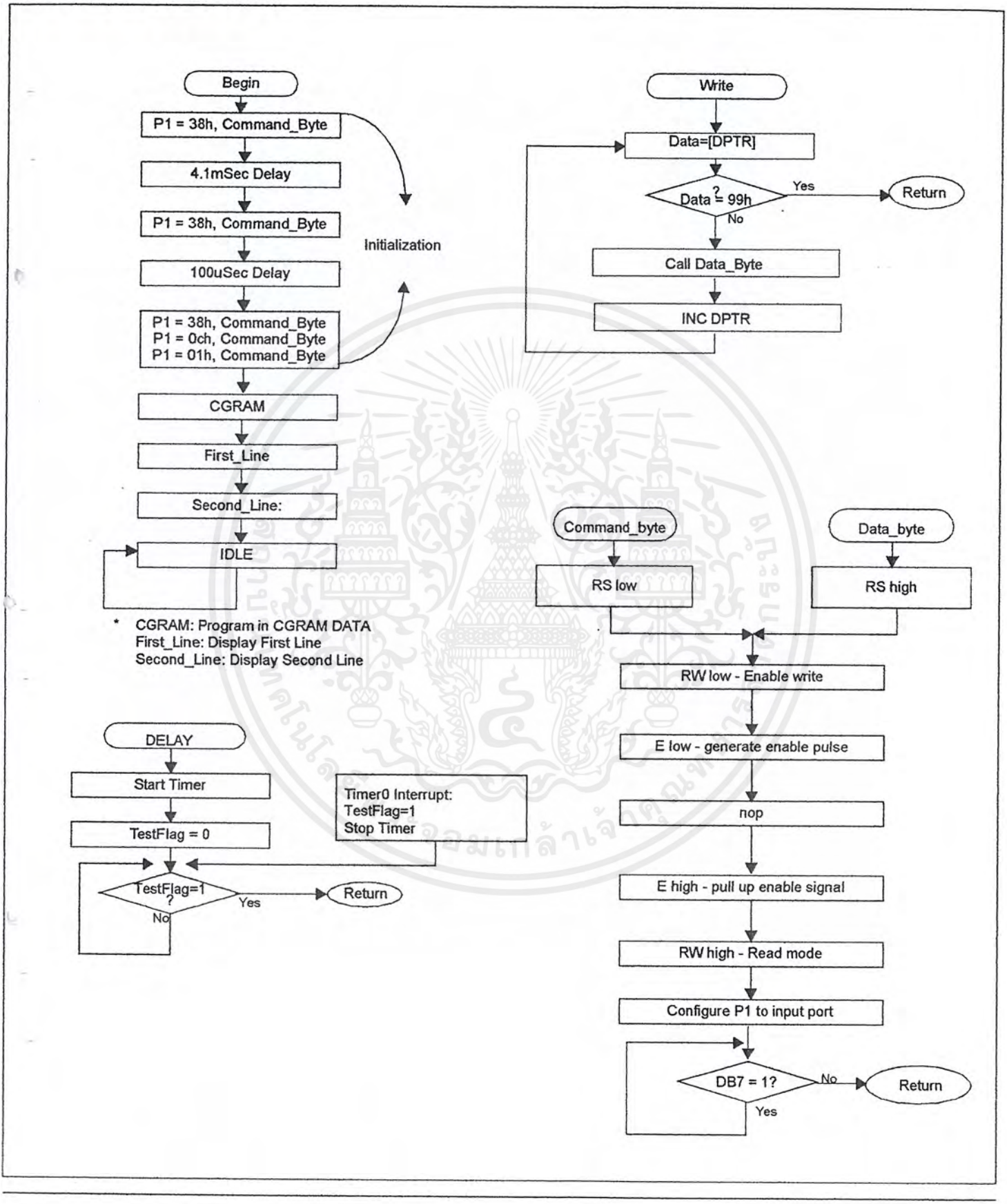
II. Circuit Schematic



Parts List	Part Number	Description
LCM1	HDM16216H-5	Hantronix Alphanumeric LCD Module
U1	87C751	Phillips Microcontroller
XTR1	NMP160	16MHz Crystal
C1,C2	33pF	Capacitor
SW1	–	Push Button Switche (Reset)
R1	10Kohm	Pot
R2	20Kohm	Resistor
C3	0.1 uF	Capacitor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

III. Software Flowchart:



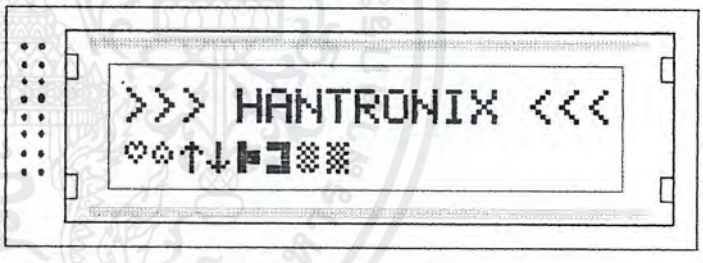
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;*****************************************************
;;      Application Note:
;       ================
;       Displaying Characters on an LCD Character Module
;
;       Description:    Demo software to display "canned"
                        message and custom characters.
;       Controller:     Phillips 87C751
;       LCD controller: HD44780, KS0066, SED1278
;
;*****************************************************
;       Constant Definition
;*****************************************************
EnableT0        equ     082h    ;enable timer0
Disable         equ     000h    ;disable timer
D4100h          equ     00ch    ;timer reload high byte def.
D4100l          equ     003h    ;timer reload = 4.1mSec.
D100h           equ     000h    ;timer reload
D100l           equ     04ch    ;timer reload = 100uSec.
;*************************************
;       Ram Definition
;*************************************
Flags           DATA    020h    ;flag
TstFlag         BIT     Flags.0 ;interrupt flag bit
;*************************************
;       Port Connections
;       ================
;       P1.0 -> D0
;       P1.1 -> D1
;       P1.2 -> D2
;       ...
;       P1.7 -> D7
;       P3.0 -> Enable
;       P3.1 -> RS
;       P3.2 -> RW
;*****************************************************
;       Interrupt Vectors
;       -----------------
        org     000h
        jmp     PowerUp         ; Power up reset vector
        org     003h
        jmp     ExInt0          ; External interrupt 0 vector
        org     00bh
        jmp     Timer0          ; Counter/ Timer 0 int vector
        org     013h
        jmp     ExInt1          ; External int 1 vector
        org     01bh            ;
        jmp     Timer1          ; Timer 1 int vector
        org     023h
        jmp     I2C             ; I2C serial int vector

        org     50h
PowerUp:
;*****   Timer 0 Interrrupt preparation
        clr     TR              ;disable timer
        clr     TF              ;clear overflow
;*************************************
;       LCD Initialization Routine
;*************************************
cinit:  clr     P3.1            ;RS low
        clr     P3.2            ;RW low
        setb    P3.0            ;Enable

        mov     RTL,#D4100l     ;set timer reload value
        mov     RTH,#D4100h     ;dealy time = 4.1mSec
        mov     p1,#38h
        acall   command_byte
        acall   ddelay          ;initial delay 4.1mSec

        mov     RTL,#D100l      ;set timer reload value
        mov     RTH,#D100h      ;dealy time = 100uSec
        mov     p1,#38h         ;function set
        acall   command_byte
        acall   ddelay          ;busy flag not avail. yet

        mov     p1,#38h         ;function set
        acall   command_byte
        mov     p1,#0ch         ;display on
        acall   command_byte
        mov     p1,#01h         ;clear display
        acall   command_byte

        acall   cgram           ;define custom fonts
        acall   first_line      ;display first line
        acall   second_line     ;display second line
sdone:
        setb    IDLE            ;power down mode
        jmp     sdone

;*****************************************************
;Subroutine: WRITE
;=================
;Purpose: To feed in data/command bytes to the LCD module
;Parameters:dptr =  should be set to the beginning of
;                   the data byte address
;                   Data bytes should be finished with 99H
;Alg:    get a new data/command byte
;        while (new data != 99h) {
;                   set port1 with new data
;                   call data_byte
;                   increment data pointer
;                   }
;        return
;*****************************************************
write:
write_loop:
        mov     a,#0
        movc    a,@a+dptr
        cjne    a,#99h,write_cont
        ret
write_cont:
        mov     p1,a
        acall   data_byte
        inc     dptr
        jmp     write_loop

;*****************************************
;       Delay Routine:
;       Delay periond = 4/3uSec * DefRth,l
;*****************************************
ddelay:
        setb    TR              ;start timer
        mov     IE,#EnableT0    ;enable timer
        clr     TstFlag         ;reset flag
dloop:  jnb     TstFlag,dloop
        ret
;*******************************
;       set address to beginning
;       of CG RAM
;*******************************
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
cgram:
        mov     p1,#40h
        acall   command_byte
        mov     dptr,#cgram_data
        acall   write
        ret
;*****************************
;       Set DDRAM to the beginnig of
;       the first line - 00
;*****************************
first_line:
        mov     p1,#080h ;set DDRAM
        acall   command_byte
        mov     dptr,#fline_data
        acall   write
        ret
;*****************************
;       Set DDRAM to the beginning of
;       the second line - 40
;*****************************
second_line:
        mov     p1,#0c0h ;set DDRAM
        acall   command_byte
        mov     dptr,#sline_data
        acall   write
        ret
;*************************************************
;       Feed Command/Data to the LCD module
;*************************************************
command_byte:
        clr     p3.1            ; RS low for a command byte.
        jmp     bdelay
data_byte:
        setb    p3.1            ; RS high for a data byte.
        nop
bdelay:
        clr     p3.2            ; R/w low for a write mode
        clr     p3.0
        nop
        setb    p3.0            ;enable pulse
        nop
;****************** Check Busy Flag
        mov     p1,#0ffh  ;configure port1 to input mode
        setb    p3.2            ;set RW to read
        clr     p3.1            ;set RS to command
        clr     p3.0            ;generate enable pulse
        nop
        setb    p3.0
bloop:  nop
        mov     a,p1
        anl     a,#80h          ;check bit#7 busy flag
        cjne    a,#00h,bloop;keep waiting until busy flag clears
;*************************************
;       check busy flag twice
;*************************************
bwait:
        mov     a,p1
        anl     a,#80h
        cjne    a,#00h,bloop
        clr     p3.2            ;return to write mode
        ret
;*************************************
;       Interrupt Routines
;*************************************
ExInt0:
ExInt1:
Timer1:
I2C:
        reti
;*************************************
Timer0:
        setb    TstFlag
        clr     TR
        mov     IE,#Disable
        reti
;*************************************
;       Data Bytes
;*************************************
FLINE_DATA:
        db      '>>> HANTRONIX <<<'
        db      099h
SLINE_DATA:
        db      00h,01h,02h,03h,04h,05h,06h,07h
        db      099h
CGRAM_DATA:
font1:  db      0ah,15h,11h,11h,0ah,04h,00h,00h
font2:  db      04h,0ah,11h,11h,15h,0ah,00h,00h
font3:  db      04h,0eh,15h,04h,04h,04h,04h,00h
font4:  db      04h,04h,04h,04h,15h,0eh,04h,00h
font5:  db      18h,18h,1fh,1fh,1fh,18h,18h,00h
font6:  db      1fh,1fh,03h,03h,03h,1fh,1fh,00h
font7:  db      0ah,15h,0ah,15h,0ah,15h,0ah,00h
font8:  db      15h,0ah,15h,0ah,15h,0ah,15h,00h
        db      99h
        end
```

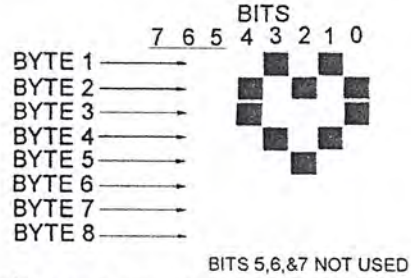
*Display on a 16x2 Character Module



* Display Character Position and DDRAM Address of a 16x2 Character Module.

	Display Position 1	2	3	4		14	15	16
First Line	00	01	02	03	....	0D	0E	0F
Second Line	40	41	42	43	....	4D	4E	4F

DDRAM Address



* Custom characters memory map.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COMMANDS FOR CHARACTER MODULES

Command	Code										Description	Execution Time
	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0		
Clear Display	0	0	0	0	0	0	0	0	0	1	Clears the display and returns the cursor to the home position (address 0).	82μs~1.64ms
Retum Home	0	0	0	0	0	0	0	0	1	*	Returns the cursor to the home position (address 0). Also returns a shifted display to the home position. DD RAM contents remain unchanged.	40μs~1.64ms
Entry Mode Set	0	0	0	0	0	0	0	1	I/D	S	Sets the cursor move direction and enables/disables the display.	40μs
Display ON/OFF Control	0	0	0	0	0	0	1	D	C	B	Turns the display ON/OFF (D), or the cursor ON/OFF (C), and blink of the character at the cursor position (B).	40μs
Cursor & Display Shift	0	0	0	0	0	1	S/C	R/L	*	*	Moves the cursor and shifts the display without changing the DD RAM contents.	40μs
Function Set	0	0	0	0	1	DL	N$	F	*	#	Sets the data width (DL), the number of lines in the display (L), and the character font (F).	40μs
Set CG RAM Address	0	0	0	1	A_{CG}						Sets the CG RAM address. CG RAM data can be read or altered after making this setting.	40μs
Set DD RAM Address	0	0	1	A_{DD}							Sets the DD RAM address. Data may be written or read after making this setting.	40μs
Read Busy Flag & Address	0	1	BF	AC							Reads the BUSY flag (BF) indicating that an internal operation is being performed and reads the address counter contents.	1μs
Write Data to CG or DD RAM	1	0	Write Data								Writes data into DD RAM or CG RAM.	46μs
Read Data from CG or DD RAM	1	1	Read Data								Reads data from DD RAM or CG RAM.	46μs
	I/D = 1: Increment I/D = 0: Decrement S = 1: Accompanies display shift. S/C= 1: Display shift S/C = 0: cursor move R/L= 1: Shift to the right. R/L= 0: Shift to the left. DL = 1: 8 bits DL = 0: 4 bits N = 1: 2 lines N = 0: 1 line F = 1: 5x10 dots F = 0: 5 x 7 dots BF = 1: Busy BF = 0: Can accept data *# Set to 1 on 24x4 modules* *$ With KS0072 is Address Mode.*											DD RAM: Display data RAM CG RAM: Character generator RAM A_{CG}: CG RAM Address A_{DD}: DD RAM Address Corresponds to cursor address. AC: Address counter Used for both DD and CG RAM address.	Execution times are typical. If transfers are timed by software and the busy flag is not used, add 10% to the above times.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้จัดทำโครงงาน

นายพรรษิษฐ์ ไตรรัตน์ เกิดวันที่ 17 สิงหาคม พ.ศ. 2521 สำเร็จการศึกษาระดับมัธยมตอนต้น และมัธยมศึกษาตอนปลายจากโรงเรียนภูเก็ตวิทยาลัย จากนั้นเข้าศึกษาต่อในระดับปริญญาตรี สาขาฟิสิกส์ประยุกต์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร ลาดกระบัง และสำเร็จการศึกษาในปีการศึกษา 2542

นายพิเชษฐ จิตวิริยะวศิน เกิดวันที่ 9 ธันวาคม พ.ศ. 2519 สำเร็จการศึกษาระดับมัธยมตอนต้นจากโรงเรียนมัธยมวัดเบญจมบพิต และระดับมัธยมตอนปลายจากโรงเรียนสามเสนวิทยาลัย จากนั้นเข้าศึกษาต่อในระดับปริญญาตรี สาขาฟิสิกส์ประยุกต์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร ลาดกระบัง และสำเร็จการศึกษาในปีการศึกษา 2542

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้