สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เครื่องตรวจจับสัญญาณอนุกรม
Serial Data Logger

ผู้จัดทำ

นาย นิรันดร์ แก้วไชค้วน 43015872

นาย อัครพร ภู่กลาง 43015903

อาจารย์ที่ปรึกษา

ผ.ศ. ไพศาล สิทธิโยภาสกุล

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

วิศวกรรมสารสนเทศ (เทคนิคอุตสาหกรรม)

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2545

เลขหมู่.....
เลขทะเบียน 50290
วัน,เดือน,ปี 28 เม.ย. 2547

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	เครื่องจับสัญญาณอนุกรม
	SERIAL DATA LOGGER
นักศึกษา	นายนิรันดร์ แก้วไชค้วน รหัสประจำตัว 43015872
	นายอัครพร ภู่กลาง รหัสประจำตัว 43015903
อาจารย์ผู้ควบคุมปริญญานิพนธ์	ผศ.ไพศาล สิทธิโยภาสกุล
ระดับการศึกษา	ปริญญาอุตสาหกรรมศาสตร์บัณฑิต
	สาขาวิชาเทคโนโลยีโทรคมนาคม
ภาควิชา	วิศวกรรมสารสนเทศ
ปีการศึกษา	2545

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณลาดกระบังอนุมัติ
ให้รับปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรอุตสาหกรรมศาสตร์บัณฑิต

.....

(ผศ.ไพศาล สิทธิโยภาสกุล)
อาจารย์ผู้ควบคุมปริญญานิพนธ์

ลิขสิทธิ์ของคณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	เครื่องจับสัญญาณอนุกรม
	SERIAL DATA LOGGER
นักศึกษา	นายนิรันดร์ แก้วไชค้วน รหัสประจำตัว 43015872
	นายอัครพร ภู่กลาง รหัสประจำตัว 43015903
อาจารย์ผู้ควบคุมปริญญานิพนธ์	ผศ.ไพศาล สิทธิโยภาสกุล
ระดับการศึกษา	ปริญญาอุตสาหกรรมศาสตร์บัณฑิต
	สาขาวิชาเทคโนโลยีโทรคมนาคม
ภาควิชา	วิศวกรรมสารสนเทศ
ปีการศึกษา	2545

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณลาดกระบังอนุมัติให้รับปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรอุตสาหกรรมศาสตร์บัณฑิต

คณะกรรมการการสอบปริญญานิพนธ์

.....ประธานกรรมการ
( )

.....กรรมการ
( )

.....กรรมการ
( )

.....กรรมการ
( )

.....กรรมการ
( )

ลิขสิทธิ์ของคณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรื่อง เครื่องจับสัญญาณอนุกรม

โดย 1. นายนิรันด์ แก้วไชค้วน
2. นายอัครพร ภู่กลาง

อาจารย์ที่ปรึกษา
ผ.ศ. ไพศาล สิทธิโยภาสกุล

ปีการศึกษา 2545

บทคัดย่อ

วิทยานิพนธ์ฉบับนี้ ได้จัดทำขึ้นเพื่อเป็นแนวทางในการจัดทำเครื่องจับสัญญาณอนุกรม ตัวเครื่องจะทำการจับข้อมูลที่บัตรสมาร์ทการ์ด และเครื่องใช้งานทำการติดต่อรับส่งกัน โดยใช้ไมโครคอนโทรลเลอร์ประมวลผลแล้วนำไปจัดเก็บไว้ใน หน่วยความจำ เพื่อสามารถนำข้อมูลที่จับมาได้ แสดงออกทางจอแสดงผลของเครื่อง หรือผ่านทางพอร์ตอนุกรมเพื่อแสดงออกทางหน้าจอคอมพิวเตอร์ในภายหลัง ทำให้มีการใช้งานที่สะดวกขึ้นกว่าในปัจจุบัน ที่ใช้เครื่องคอมพิวเตอร์เป็นตัวประมวลผลในการจับสัญญาณและจัดเก็บทำให้เป็นการไม่สะดวกในการใช้งาน

Abstract

By MR.NIRUN KEAWSIKOUN
MR.UKARAPORN BHUKLANG

ADVISER Asso.Prof PAISAN SITIYOPASSAKOL
YEAR 2545

This paper had purposed to be a guide for how to make The Serial Data Logger. This equipment will get the data from smart card and communication unit by use microcontroller from processing, the data were kept in memory unit (EEprom) and displayed on LCD or serial port form display again on computer monitor later.

There fore , it will make more convenience then today this we use computer to processing and to keep the signal data .

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

การจัดทำปริญญานิพนธ์นี้สำเร็จลุล่วงได้ด้วยดี ดังนั้นขอขอบพระคุณผู้ร่วมงาน และ ผ.ศ. ไพศาล สิทธิโยภาสกุล ซึ่งเป็นอาจารย์ที่ปรึกษา ที่ช่วยเหลือทางด้านเครื่องมือ อุปกรณ์ต่างๆ และคำแนะนำในการดำเนินงานพร้อมทั้งเป็นที่ปรึกษาตลอดเวลาที่ทำปริญญานิพนธ์ชิ้นนี้

นิรันดร์ แก้วไชค้วน

อัครพร ภู่กลาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	
บทที่ 1 บทนำ	1
ความเป็นมาของโครงงาน	1
วัตถุประสงค์	2
ผลที่คาดว่าจะได้รับ	2
บทที่ 2 ทฤษฎี	3
SmartCard	3
PIC (Peripheral Interface Control)	18
พอร์ตสื่อสารข้อมูลอนุกรม RS 232-C	33
LCD	42
Sram 62256	49
บทที่ 3 การออกแบบและการทำงาน	54
ออกแบบเครื่องจับสัญญาณอนุกรม	54
สร้างเครื่องจับสัญญาณอนุกรม	56
การทำงานของโปรแกรมเครื่องจับสัญญาณอนุกรม	69
บทที่ 4 การทดลอง	76
การทดสอบความถูกต้องของเครื่อง	76
เครื่องต้นแบบ SERIAL DATA LOGGER	81
บทที่ 5 สรุป	91
ผลของโครงงาน	91
บรรณณานุกรรม	
ภาคผนวก	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

รูปที่ 2.1 ส่วนประกอบภายนอกของบัตรสมาร์ทการ์ด หน้า 5

รูปที่ 2.2 แผ่นหน้าสัมผัสของบัตรสมาร์ทการ์ด 5

รูปที่ 2.3 ขนาดความกว้าง x ยาวx หนา ของสมาร์ตการ์ดตามมาตรฐาน ISO786 7

รูปที่ 2.4 ตำแหน่งหน้าสัมผัสที่ผิวหน้าของสมาร์ตการ์ด ตามมาตรฐาน ISO786 8

รูปที่ 2.5 ตำแหน่งขาสำหรับการอินเตอร์เฟสกับสมาร์ตการ์ด 9

รูปที่ 2.6 รูปสัญญาณสำหรับการอินเตอร์เฟสแบบ Asynchronous 14

รูปที่ 2.7 รูปสัญญาณสำหรับการอินเตอร์เฟสแบบ Synchronous 14

รูปที่ 2.8 รูปสัญญาณที่ขาต่าง ๆ ของ สมาร์ตการ์ด แบบ MCU ในระหว่างการรีเซต 15

รูปที่ 2.9 รูปสัญญาณที่ขาต่าง ๆ ของ สมาร์ตการ์ด แบบ Memory card
ในระหว่างขั้นตอนการรีเซต 16

รูปที่ 2.10 PIC 16F877 19

รูปที่ 2.11 PIC16F874 AND PIC16F877 BLOCK DIAGRAM 22

รูปที่ 2.12 PIC 16F877 PROGRAM MEMORY MAP AND STACK 25

รูปที่ 2.13 PIC 16F877 REGISTER FILE MAP 26

รูปที่ 2.14 แสดงตำแหน่งขาของ DB-9 33

รูปที่ 2.15 TTL/CMOS Serial Logic Waveform 35

รูปที่ 2.16 RS-232 Logic Waveform 35

รูปที่ 2.17 MAX-232 36

รูปที่ 2.18 ไดอะแกรมโครงสร้างและการทำงานของตัวควบคุมภายในโมดูล LCD 43

รูปที่ 2.19 การจัดขาของโมดูล LCD ขนาด 16 ตัวอักษร 2 บรรทัด ซึ่งมีด้วยกัน 2 แบบ 45

รูปที่ 2.20 แสดงตำแหน่งขาของ SRAM 62256 50

รูปที่ 2.21 ไดอะแกรมเวลาของการอ่านข้อมูลจากแรม 51

รูปที่ 2.22 ไดอะแกรมเวลาของการเขียนหน่วยความจำ 53

รูปที่ 3.1 ส่วนประกอบภายในของเครื่องจับสัญญาณอนุกรม 54

รูปที่ 3.2 วงจรเครื่องจับสัญญาณอนุกรม 56

รูปที่ 3.3 รูปจำลองการต่อของคอนแท็คจำลองกับคอนแท็คของบัตรจริง 57

รูปที่ 3.4 แสดงการต่อใช้งาน LCD 60

รูปที่ 3.5 การเชื่อมต่อระหว่างหน่วยความจำกับซีพียู 62

รูปที่ 3.6 Function Diagram IC Latch 63

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ 2.1 คุณลักษณะทางไฟฟ้าของสัญญาณที่ใช้ในการสื่อสารกับมาร์ทการ์ด 10

ตารางที่ 2.2 คุณลักษณะทางไฟฟ้าของ VPP ภายใต้การทำงานปกติ 11

ตารางที่ 2.3 คุณลักษณะทางไฟฟ้าของสัญญาณที่ขา CLK ภายใต้การทำงานปกติ 12

ตารางที่ 2.4 คุณลักษณะทางไฟฟ้าของสัญญาณที่ขา RST ภายใต้การทำงานปกติ 12

ตารางที่ 2.5 ระดับแรงดันไฟเลี้ยงและกระแสที่ป้อนให้กับสมาร์ตการ์ด
ภายใต้การทำงานปกติ 13

ตารางที่ 2.6 ตัวอย่างของข้อมูล ATR ของสมาร์ตการ์ดจากผู้ผลิตรายหนึ่ง 17

ตารางที่ 2.7 เปรียบเทียบ PIC 16F87X 21

ตารางที่ 2.8-1 PIC 16F877 PINOUT DISCRIPTION 23

ตารางที่ 2.8-2 PIC 16F877 PINOUT DISCRIPTION 24

ตารางที่ 2.9 SPECIAL FUNCTION REGISTER SUMMARY(BANK 0) 27

ตารางที่ 2.10 SPECIAL FUNCTION REGISTER SUMMARY(BANK 1) 28

ตารางที่ 2.11 SPECIAL FUNCTION REGISTER SUMMARY(BANK 2-3) 29

ตารางที่ 2.12 คำสั่ง ของ pic 16F877 32

ตารางที่ 2.13 แสดงรายละเอียดการทำงาน RS 232 34

ตารางที่ 2.14 อัตราและช่วงเวลาของแต่ละบิตข้อมูลในการสื่อสารข้อมูลอนุกรม 34

ตารางที่ 2.15 REGISTER ของ PIC ที่เกี่ยวกับ SERIAL COMMUNICATION 37

ตารางที่ 2.16 PIR1 register บิตที่เกี่ยวข้องก็คือ RCIF และ TXIF 37

ตารางที่ 2.17 RCSTA register เป็น register ที่เกี่ยวข้องกับการรับข้อมูล 38

ตารางที่ 2.18 TXSTA register เป็น register ที่เกี่ยวข้องกับการส่งข้อมูล 39

ตารางที่ 2.19 PIE1 บิตที่เกี่ยวข้องคือ RCIE,TXIE 40

ตารางที่ 2.20 SPBRG register คือ register ที่กำหนดค่าที่ baud rate 41

ตารางที่ 2.21 แสดงความสัมพันธ์ของการทำงานระหว่างขา RS , R/W และ E
ของโมดูล LCD แบบอักขระ 44

ตารางที่ 2.22 คำสั่งเลือกโหมดการป้อนข้อมูล 45

ตารางที่ 2.23 คำสั่งควบคุมการแสดงผล 46

ตารางที่ 2.24 คำสั่งควบคุมการเลื่อนเคอร์เซอร์และตัวอักษร 46

ตารางที่ 2.25 การเลื่อนเคอร์เซอร์และตัวอักษรขึ้นอยู่กับการกำหนดบิต S/C และ R/L 47

ตารางที่ 2.26 คำสั่งกำหนดฟังก์ชั่นการทำงาน 47

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.27 แสดงFUNCTION การทำงานของ SRAM62256 51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาของโครงงาน

ปัจจุบันได้เริ่มมีการใช้บัตรสมาร์ทการ์ดกันอย่างแพร่หลายมาขึ้น ในง่านด้านต่างๆมากมายอย่างกว้างขวาง เช่น ในด้านการจ่ายเงิน การรักษาความปลอดภัย อาทิบัตรผ่านประตูเข้าออก การชำระค่าบริการต่างๆ อาทิการใช้ร่วมกับอินเตอร์เน็ต การใช้ร่วมกับโทรศัพท์ เป็นต้น

เนื่องจากมีข้อดีต่างๆมากมาย ดังนี้

1. พิสูจน์แล้วว่ามีความไว้วางใจได้ดีกว่าบัตรที่ใช้แถบแม่เหล็ก
2. สามารถเก็บสะสมข้อมูลได้มากกว่าบัตรที่ใช้แถบแม่เหล็กเป็นร้อย ๆ เท่า
3. ลดโอกาสที่จะเข้าไปยุ่งเกี่ยวและป้องกันการปลอมแปลงด้วยระบบป้องกันที่ซับซ้อน
4. สามารถเปลี่ยนมือและนำกลับมาใช้ใหม่ได้
5. ทำหน้าที่ต่าง ๆ ได้มากมาย
6. สามารถนำไปใช้ในงานต่าง ๆ ได้อย่างกว้างขวาง เช่น การขนส่ง ธนาคาร และการรักษาสุขภาพ เป็นต้น
7. สามารถประยุกต์ใช้กับอุปกรณ์อิเล็กทรอนิกส์แบบพกพาต่าง ๆ ได้ เช่น เครื่องโทรศัพท์และเครื่องคอมพิวเตอร์กระเป๋าหิ้ว
8. ทำงานด้วยเทคโนโลยีเซมิคอนดัคเตอร์ที่มีการพัฒนาอย่างรวดเร็ว

โดยการติดต่อสื่อสารกับอุปกรณ์ต่างๆ ของสมาร์ทการ์ดนั้น จะเป็นไปตามมาตรฐาน ISO 7816 ส่วนโค๊ดในการติดต่อต่างๆจะขึ้นอยู่กับความเหมาะในการนำไปใช้งาน

อุปกรณ์ที่จะนำมาประกอบการตรวจสอบดูแล และแก้ไข การทำงานของบัตรสมาร์ทการ์ดก็เป็นส่วนหนึ่งที่จะช่วยให้การดูแลและบริการเป็นไปได้อย่างสะดวก เนื่องจากในการทำงานนั้นหากการติดต่อระหว่างอุปกรณ์และบัตรสามร์ทการ์ดมีการขัดข้องเกิดขึ้น เราไม่สามารถเห็นได้ว่าเกิดการบกพร่องจากส่วนได้ ว่าเกิดจากตัวสมาร์ทการ์ด หรือ อุปกรณ์ที่ใช้งานร่วม ที่มีการตอบโต้หรือส่งสัญญาณที่ไม่เป็นไปตามต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จึงได้เกิดแนวคิดที่จะจัดทำ อุปกรณ์สำหรับตรวจสอบการส่งสัญญาณขึ้นเรียกว่าเครื่องตรวจจับสัญญาณแบบอนุกรม เพื่อให้สามารถตรวจสอบดูการส่งสัญญาณตอบโต้ระหว่งสมาร์ทการ์ดและอุปกรณ์ต่างๆที่เรานำมาใช้ร่วมกันว่ามีการส่งสัญญาณที่ถูกต้องเป็นไปตามความต้องการหรือไม่ หรือมีสาเหตุบกพร่องมาจากส่วนใด คล้ายกับการที่ระบบไฟฟ้ามีมิเตอร์ เป็นตัวตรวจสอบวงจร

1.2 วัตถุประสงค์

1. เพื่อศึกษาระบบการรับส่งสัญญาณการติดต่อระหว่างสมาร์ทการ์ดและอุปกรณ์ต่างๆ
2. เพื่อความสะดวกในการจัดเก็บและแสดงสัญญาณสมาร์ทการ์ทที่ต้องการนำมาศึกษา
3. เพื่อให้ได้อุปกรณ์การใช้งานที่เหมาะสมกับงาน

1.3 ผลที่คาดว่าจะได้รับ

1. สามารถจับสัญญาณการติดต่อระหว่างบัตรสมาร์ทการ์ดและตัวเครื่องใช้งานร่วมได้อย่างถูกต้อง
2. สามารถนำได้ใช้ในการศึกษาทดลองการทำงานของสมาร์ทการ์ดได้
3. ได้รับความรู้ในกระบวนการทำงานของสมาร์ทการ์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎี

2.1SmartCard

2.1.1สมาร์ทการ์ดคืออะไร

สมาร์ทการ์ด เป็นบัตรชนิดหนึ่งที่มีขนาดพอ ๆ กับบัตรเครดิตหรือบัตรเอทีเอ็มพลาสติก ที่มีการฝังชิพคอมพิวเตอร์ (Computer Chip) ไว้ภายในบัตร โดยที่ตัวชิพดังกล่าวนี้ ภายในบรรจุข้อมูลต่าง ๆ ไว้ในรูปแบบทางอิเล็กทรอนิกส์ด้วยกรรมวิธีที่มีการรักษาความปลอดภัยอย่างดีเยี่ยม

2.1.2สมาร์ทการ์ดสามารถแบ่งออกได้เป็น 2 แบบใหญ่ ๆ คือ

1. สมาร์ทการ์ดแบบมีการสัมผัส (Contact smart cards) ซึ่งการใช้งานจำเป็นต้องมีการสอดใส่เข้าไปในเครื่องอ่านสมาร์ทการ์ด (smart card reader)

2. สมาร์ทการ์ดแบบไม่มีการสัมผัส (Contactless smart cards) ซึ่งการใช้งานต้องการเพียงให้วางอยู่ใกล้ ๆ กับสายอากาศเท่านั้น

สมาร์ทการ์ดแบบมีการสัมผัสเป็นบัตรที่มีการผนึกชิพทองขนาดเล็กเส้นผ่าศูนย์กลางประมาณครึ่งนิ้ว เอาไว้ที่ด้านหน้าบัตร แทนการใช้แถบแม่เหล็ก (Magnetic stripe) ที่เคยพบเห็นใช้กันมากที่สุดในบัตรเครดิตหรือบัตรเอทีเอ็ม เมื่อผู้ใช้สอดใส่บัตรเข้าไปในเครื่องอ่านบัตรสมาร์ทการ์ดแล้ว มันจะสัมผัสกับหัวต่อหรือคอนเน็กเตอร์ทางไฟฟ้า ซึ่งจะทำการส่งถ่ายข้อมูลเข้าและออกจากชิพ

สมาร์ทการ์ดแบบไม่มีการสัมผัส เป็นบัตรที่มองดูรูปร่างภายนอกแล้วคล้ายกับบัตรเครดิตพลาสติกแบบหนึ่ง ที่ภายในมีการผนึกชิพคอมพิวเตอร์และขดลวดสายอากาศไว้ภายใน ซึ่งใช้ในการติดต่อกับเครื่องรับ/เครื่องส่งที่อยู่ในระยะไกล (Remote receiver/transmitter) โดยทั่ว ๆ ไปเรา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มักใช้บัตรแบบนี้เมื่อต้องมีการดำเนินการทางด้านรายการ (Transactions) อย่างรวดเร็ว ตัวอย่างเช่นที่ใช้กับการจัดเก็บเงินค่าผ่านทางด่วน

นอกจากบัตรสมาร์ทการ์ดทั้งสองแบบดังกล่าวแล้ว ปัจจุบันยังมีการผลิตสมาร์ทการ์ดแบบผสมหรือที่เรียกว่า คอมบิการ์ด (Combi Card) ออกมาใช้งานอีกด้วย โดยบัตรแบบนี้เป็นบัตรใบเดียวแต่ทำหน้าที่เป็นทั้งสมาร์ทการ์ดแบบมีการสัมผัส และสมาร์ทการ์ดแบบไม่มีการสัมผัสเพื่อเพิ่มความสะดวกและประโยชน์ในการใช้งานมากขึ้น

2.1.3สมาร์ทการ์ดมีข้อดีหลายประการที่ควรกล่าวถึง คือ

1. พิสูจน์แล้วว่ามีความไว้วางใจได้ดีกว่าบัตรที่ใช้แถบแม่เหล็ก
2. สามารถเก็บสะสมข้อมูลได้มากกว่าบัตรที่ใช้แถบแม่เหล็กเป็นร้อย ๆ เท่า
3. ลดโอกาสที่จะเข้าไปยุ่งเกี่ยวและป้องกันการปลอมแปลงด้วยระบบป้องกันที่ซับซ้อน
4. สามารถเปลี่ยนมือและนำกลับมาใช้ใหม่ได้
5. ทำหน้าที่ต่าง ๆ ได้มากมาย
6. สามารถนำไปใช้ในงานต่าง ๆ ได้อย่างกว้างขวาง เช่น การขนส่ง ธนาคาร และการรักษาสุขภาพ เป็นต้น
7. สามารถประยุกต์ใช้กับอุปกรณ์อิเล็กทรอนิกส์แบบพกพาต่าง ๆ ได้ เช่น เครื่องโทรศัพท์และเครื่องคอมพิวเตอร์กระเป๋าหิ้ว
8. ทำงานด้วยเทคโนโลยีเซมิคอนดัคเตอร์ที่มีการพัฒนาอย่างรวดเร็ว

2.1.4บัตรพลาสติค (Plastic Card)

ขนาดของบัตรพลาสติคกำหนดโดยมาตรฐานระหว่างประเทศ คือ ISO 7810 โดยมาตรฐานนี้ยังได้กำหนดถึงคุณลักษณะทางกายภาพของพลาสติคที่นำมาใช้ทำบัตรด้วย เช่น ความคลาดเคลื่อนของอุณหภูมิ และความยืดหยุ่นตัวในการใช้งาน ตำแหน่งของหน้าสัมผัสทางไฟฟ้าและการทำงานของมัน ตลอดจนกำหนดว่าการติดต่อระหว่างวงจรร่วม (Integrated Circuit) หรือ IC กับโลกภายนอกเป็นอย่างไรอีกด้วย

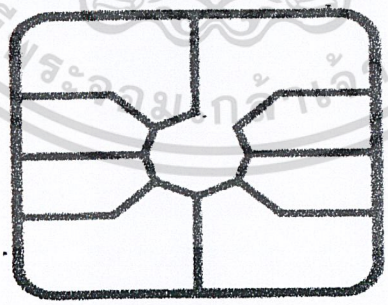
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีพลาสติกอยู่หลายชนิดที่นำมาใช้ผลิตสมาร์ทการ์ด แต่ที่นิยมใช้กันมากคือ พีวีซี (PVC - Polyvinyl Chloride) และเอบีเอส (ABS - Acrylonitrile Butadiene Styrene) อย่างไรก็ดี การใช้พีวีซีมีข้อดีคือสามารถพิมพ์ลายนูนได้ แต่ไม่สามารถนำกลับมาใช้ใหม่ได้ ส่วนเอบีเอสไม่สามารถพิมพ์นูนได้แต่นำกลับมาใช้งานใหม่ได้

2.1.5ลักษณะทางกายภาพของสมาร์ทการ์ด

1. ส่วนประกอบทั่วไปของสมาทการ์ด รูปภายนอกของบัตร สมาร์ทการ์ดมีความแตกต่างจากบัตรเครดิตโดยทั่วๆ ไปโดยที่บัตร สมาร์ทการ์ดมีแผ่นโลหะสีทองขนาดเหรียญกลมอยู่ใกล้ขอบของบัตรสมาร์ทการ์ดแผ่นโลหะนี้จะมีพื้นที่แล่งออกเป็น 8 ส่วน (บางชนิดมี 6 ส่วน) และจะทำงานด้วยวงจรไฟฟ้าแผงเล็กๆ ซึ่งสื่อสารกับส่วนภายนอก ดังรูปที่ 2.1

รูปที่2.1 ส่วนประกอบภายนอกของบัตรสมาร์ทการ์ด



รูปที่2.2 แผ่นหน้าสัมผัสของบัตรสมาร์ทการ์ด

ส่วนต่างๆของสมาร์ทการ์ดภายในแผงวงจรไฟฟ้า(Chip) ประกอบด้วย

1.1 ที่หน่วยความจำ (Memory Area) หน่วยความจำ(Memory) เป็นพื้นที่ที่ใหญ่ที่สุดในแผงวงจรไฟฟ้า (Chip) ประกอบด้วยหน่วยความจำหลายรูปแบบ ซึ่งบัตรสมาร์ทการ์ดต้องเก็บรักษาข้อมูล รหัส และข้อมูลข่าวสารที่ไม่เปลี่ยนแปลงไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 หน่วยประมวลผลกลาง (CPU) และระบบปฏิบัติการ (Operating System) แผงวงจรไฟฟ้าของบัตร สมาร์ทการ์ดเกือบจะเป็นระบบคอมพิวเตอร์ที่สมบูรณ์ โดยไม่มีคีย์บอร์ด หน่วยแสดงผล และแบตเตอร์รี่ ไว้ภายในจุดศูนย์รวมของแผงวงจรไฟฟ้าของบัตรสมาร์ทการ์ด คือ ไมโครโปรเซสเซอร์ หน่วยประมวลผลกลาง ซึ่งหน่วยประมวลผลกลางสามารถทำตามคำสั่งที่ได้กำหนดไว้ตามความสามารถของบัตร สมาร์ทการ์ด การบัตรจุคำสั่งนี้จะบรรจุลงอย่างถาวรลงไปในบัตร สมาร์ทการ์ดโดยโรงงานที่ผลิตบัตร สมาร์ทการ์ดจะใช้สถาปัตยกรรมบางอย่างในบัตร สมาร์ทการ์ดเช่น การนำโปรเซสเซอร์แบบ RISC (Reduced Instruction Set Computer) สำหรับบัตร สมาร์ทการ์ดซึ่งมีความสามรถในการทำงานที่สูงขึ้น

1.2.1 วงจรปกป้องข้อมูล (Protection Circuit)

1.2.2 วงจรรีเซ็ต(Reset Circuit)

1.2.3 วงจรกำเนิดสัญญาณนาฬิกา(Clock Circuit)

1.2.4 วงจรรับส่งข้อมูล(Imput/Output Circuit)

ขนาดของพื้นที่ของแผงวงจรไฟฟ้าภายในบัตร สมาร์ทการ์ดขึ้นอยู่กับประเภทของบัตร ถ้าเป็น Memory Card จะไม่มีหน่วยประมวลผลกลาง (CPU) ในขณะที่บัตรชนิดอื่นๆ อาจไม่มีวงจรรีเซต(Reset Circuit) หรือวงจรกำเนิดสัญญาณนาฬิกา (Clock Circuit)และวงจรที่ทำหน้าที่ในส่วนอื่นๆ

วงจรอื่นๆ ซึ่งอยู่ภายในบัตร สมาร์ทการ์ดยังมีวงจรไฟฟ้าอื่นๆ เช่นมี Protection Circuit เพื่อหลีกลเยงความเสียหายจากไฟฟ้าที่อาจทำลายส่วนประกอบของบัตรได้ นอกจากนี้บัตรสมาร์ทการ์ดยังมีวงจรรีเซต วงจรรับส่งข้อมูล และวงจรที่ทำหน้าที่ในส่วนอื่นๆ

2. หน้าสัมผัสของบัตร สมาร์ทการ์ด(Smart Card Contract) หน้าสัมผัสของบัตรทำหน้าที่เชื่อมต่อระหว่างแผงวงจรไฟฟ้าภาย สมาร์ทการ์ดกับ Interface Device ภายนอกสมาร์ทการ์ดสามรถแบ่งออกเป็น 2 ชนิดคือ

- Memory Cards จะมีรูปแบบการรับและส่ง ข้อมูลแบบ Synchronous
- Microprocesser Card จะมีรูปแบบการรับและส่งข้อมูลแบบ Asynchronous

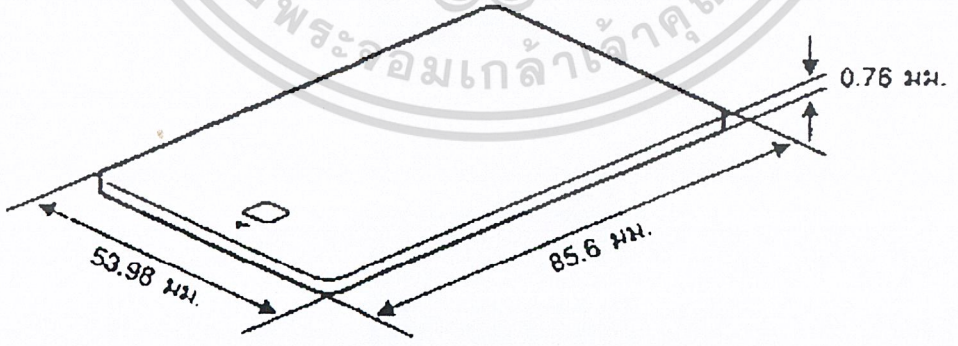
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.6 พื้นฐานในการอินเตอร์เฟสสมาร์ตการ์ดแบบสัมผัส

มาตรฐานและวิธีการอินเตอร์เฟสกับสมาร์ตการ์ดแบบสัมผัส โดยจะกล่าวอ้างอิงกับ มาตรฐาน ISO7816 ซึ่งครอบคลุมการอินเตอร์เฟสในระดับล่าง (Low level) เพื่อควบคุมการทำงานของสมาร์ตการ์ด เนื่องจากสมาร์ตการ์ดที่มีใช้กันในปัจจุบันไม่ว่าจะเป็น บัตรเครดิต , บัตรเอทีเอ็ม , บัตรโทรศัพท์ ส่วนใหญ่เป็นการ์ดซึ่งมาตรฐานนี่แทบทั้งสิ้น ซึ่งจะกล่าวถึงหลักการพื้นฐานของการอินเตอร์เฟสกับสมาร์ทการ์ดโดยจะครอบคลุมเกี่ยวกับคุณลักษณะทางกายภาพและทางไฟฟ้าของสมาร์ทการ์ด , รูปแบบและวิธีการอินเตอร์เฟสอย่างง่าย ๆ รวมทั้งข้อมูลพื้นฐานต่าง ๆ ที่เกี่ยวข้อง

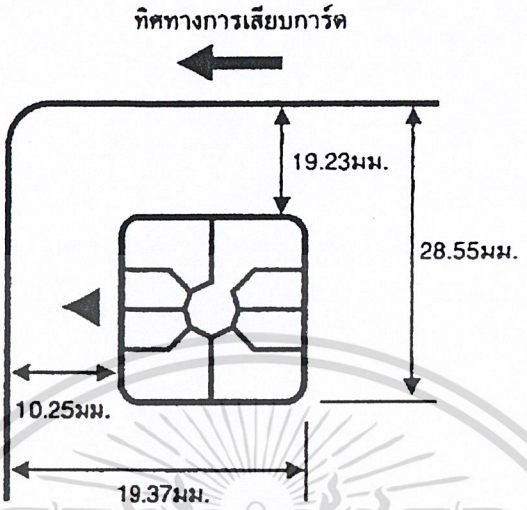
2.1.6.1 ISO7816 มาตรฐานสมาร์ตการ์ดแบบสัมผัส

มาตรฐาน ISO786 เป็นมาตรฐานของสมาร์ตการ์ดแบบสัมผัส (Contact Smart card) ที่กำหนดขึ้นโดยองค์กรมาตรฐานสากล ISO เนื้อหาของมาตรฐานที่ว่านี้โดยหลัก ๆ ประกอบด้วย 3 ส่วนด้วยกัน คือ ISO786-1 , ISO786-2 และ ISO786-3 ประเด็นที่สำคัญของทั้ง 3 ส่วน กล่าวโดยสรุปได้ว่าเป็นข้อกำหนดต่าง ๆ ที่เกี่ยวข้องกับการ์ด ได้แก่ คุณลักษณะทางกายภาพของสมาร์ตการ์ด , ขนาดสัดส่วน (กว้างx ยาวx หนา) และตำแหน่งหน้าสัมผัสบนสมาร์ตการ์ด รวมทั้งคุณสมบัติทางไฟฟ้าของสัญญาณต่าง ๆ และรูปแบบของโปรโตคอลในการสื่อสาร



รูปที่ 2.3 ขนาดความกว้าง x ยาวx หนา ของสมาร์ตการ์ดตามมาตรฐาน ISO786

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่2.4 ตำแหน่งหน้าสัมผัสที่ผิวหน้าของสมาร์ตการ์ด ตามมาตรฐาน ISO7816

รูปที่2.3 และ2.4 เป็นขนาดสัดส่วนและตำแหน่งหน้าสัมผัสของสมาร์ตการ์ด ตามที่มาตรฐาน ISO7816 กำหนดไว้ ซึ่งเป็นข้อมูลที่ผู้ผลิตสมาร์ตการ์ดและเครื่องอ่านใช้ในการผลิต จึงเป็นเหตุผลว่าทำไมผลิตภัณฑ์จากผู้ผลิตต่างรายจึงสามารถใช้งานร่วมกันได้โดยไม่เกิดปัญหา การอินเตอร์เฟสกับสมาร์ตการ์ดจะทำด้วยการเชื่อมต่อเขากับขาสัมผัส (Contact pin) ในรูปที่ว่านี้มีอยู่ทั้งหมด 8 ขา ด้วยกัน แต่ละขาต่างมีหน้าที่และประโยชน์ใช้งานที่ต่างกัน โดยขาเหล่านี้จะไปเกี่ยวข้องกับสัญญาณหลักซึ่งใช้ในการอินเตอร์เฟสสมาร์ตการ์ด 5 สัญญาณ คือ

- **I/O** : หรือสัญญาณอินพุต / เอาต์พุตข้อมูลแบบอนุกรม (Serial data)
- **VPP** : สัญญาณโปรแกรมข้อมูลสำหรับหน่วยความจำ Non-volatile ของการ์ด
- **CLK** : สัญญาณนาฬิกาอ้างอิงการเข้าถึงข้อมูล
- **RST** : สัญญาณรีเซตการทำงานการ์ด
- **VCC** และ **GND** : ไฟเลี้ยงและกราวด์ที่ป้อนให้การ์ด

สัญญาณหลักในการอินเตอร์เฟสทั้ง 5 สัญญาณนี้จะทำงานคู่ขาสัมผัสบนสมาร์ตการ์ดที่มีขาชื่อเดียวกัน โดยแต่ละตำแหน่งสัญญาณจะถูกกำหนดช่วงค่าที่เหมาะสมเอาไว้โดยมาตรฐาน ISO7816 ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.6.2 หน้าที่และตำแหน่งขาของสมาร์ตการ์ด

หน้าที่การทำงานของขาสัมผัสแต่ละขา โดยในรูปที่ 2.5 เป็นตำแหน่งของขาสัมผัสทั้ง 8 ขา ซึ่งเป็นจุดที่เครื่องอ่านใช้อินเตอร์เฟสกับวงจรภายในการ์ด เนื่องจากขาที่ใช้งานเป็นหลักจะมีอยู่เพียง 6 ขา โดยที่ขา RFU (Reserved for future use) มักไม่ถูกงาน (อาจมีการใช้งานบ้างกับการ์ดของบางผู้ผลิตเท่านั้น) ขาสัมผัสที่จะถูกกล่าวถึงจึงเป็น 6 ขาที่สำคัญเท่านั้นในส่วนของคุณลักษณะทางไฟฟ้าของสัญญาณที่ขาต่าง ๆ (ซึ่งคือช่วงค่าทางไฟฟ้าของแต่ละสัญญาณที่มาตรฐาน ISO7816 กำหนดไว้) จะถูกกล่าวถึงด้วยในรูปของตารางซึ่งใช้สัญญาณอ้างอิงที่กำหนด ในการอ่านค่าจากแต่ละตาราง

รูปที่ 2.5 ตำแหน่งขาสำหรับการอินเตอร์เฟสกับสมาร์ตการ์ด

สัญลักษณ์ของแรงดัน, กระแสไฟฟ้า, คาบเวลาและค่าความจุไฟฟ้า
(ประกอบตารางที่ 2.5)

Vih : แรงดันอินพุตค่าสูง

Vil : แรงดันอินพุตค่าต่ำ

Vcc : ระดับแรงดันไฟเลี้ยงที่ใช้งาน

Vpp : แรงดันที่ใช้โปรแกรมสมาร์ตการ์ดที่ภายในเป็น PROM

Voh : แรงดันเอาต์พุตค่าสูง

Vol : แรงดันเอาต์พุตค่าต่ำ

tr : ช่วงระยะเวลาขาขึ้น (คิดที่ช่วง 10 % ถึง 90 % ของแอมปลิจูด)

tf : ช่วงระยะเวลาขาขึ้น (คิดที่ช่วง 10 % ถึง 90 % ของแอมปลิจูด)

Iih : กระแสไฟฟ้าอินพุตค่าสูง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Iil : กระแสไฟฟ้าอินพุตค่าต่ำ

Icc : ระดับกระแสไฟจากแหล่งจ่ายไฟ

Ipp : กระแสที่ใช้โปรแกรมสมาร์ตการ์ดที่ภายในเป็น PROM

Ioh : กระแสไฟฟ้าที่เอาต์พุตค่าสูง

Iol : กระแสไฟฟ้าที่เอาต์พุตค่าต่ำ

Cin : ค่าความจุไฟฟ้าที่อินพุต

Cout : ค่าความจุไฟฟ้าที่เอาต์พุต

ขา I/O

ทำงานเป็นอินพุตหรือเอาต์พุตเพื่อรับหรือส่งข้อมูลอนุกรม ด้วยรูปแบบ Half duplex (ผลัดกันส่งข้อมูลบนสายนำสัญญาณเดียว) โดยที่ขาI/O นี้จะมีสถานะที่เป็นได้อยู่ 2 สถานะด้วยกัน คือ สถานะลอจิกสูง (High) เมื่อการ์ดรับหรือส่งบิตข้อมูลที่เป็นลอจิก "1" สถานะลอจิกต่ำ (Low) เมื่อการ์ดรับหรือส่งบิตข้อมูลที่เป็นลอจิก "0" โดยคุณลักษณะทางไฟฟ้าของสัญญาณที่ขา I/O ควรจะมีค่าเป็นตามตารางที่ 2.1

สัญลักษณ์	สภาวะเงื่อนไข	ค่าต่ำสุด	ค่าสูงสุด	หน่วย
Vih (1)	lih(max) = ±500μA	2.0	VCC	V
	lih(max) = ±50μA	0.7x VCC	VCC (3)	V
Vil	lil(max) = 1mA	0	0.8	V
Voh (2)	lol(max) = ± 100μA	2.4	VCC	V
	lol(max) = ± 20μA	3.8	VCC	V
Vol	lol(max) =1mA	0	0.4	V
tr, tf	Cin = 30pF ; Cout = 30pF	-	1	μs

(1) สำหรับเครื่องอ่านการ์ดบนสภาวะเงื่อนไขต่างกัน 3 กรณี

(2) ในที่นี้กำหนดให้มีการต่อต้านทานพูลอัพ 20 กิโลโอหม์ อยู่ด้วย

(3) ระดับแรงดันของสัญญาณที่ขา I/O ควรมีค่าอยู่ระหว่าง 0.3V และ VCC+0.3V

ตารางที่ 2.1 คุณลักษณะทางไฟฟ้าของสัญญาณที่ใช้ในการสื่อสารกับมาร์ทการ์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขา VPP

ใช้ป้อนแรงดันค่าสูงเพื่อเขียนหรือลบข้อมูลในหน่วยความจำ ขานี้จะมีสถานะเป็นได้ 2 สถานะด้วยกัน คือ สถานะสว่าง (Idle state) และสถานะทำงาน (Active state) โดยคุณลักษณะทางไฟฟ้าของสัญญาณที่ขา VPP ควรมีค่าเป็นดังตารางที่ 2.2 จะเห็นได้ว่า เมื่อต้องการจะเขียนหรือลบข้อมูลในหน่วยความจำ จะต้องใช้ระดับของแรงดันและกระแสไฟฟ้าที่สูงขึ้นกว่าปกติ ซึ่งในทางปฏิบัติแล้วค่าจะมากหรือน้อยเพียงใดก็ขึ้นอยู่กับว่าผู้ผลิตการ์ดชนิดนั้นกำหนดค่าไว้ว่าเป็นเท่าไรด้วย

สัญลักษณ์	สภาวะเงื่อนไข	ค่าต่ำสุด	ค่าสูงสุด	หน่วย
Vpp	สถานะว่าง (ยังไม่ทำการโปรแกรมการ์ด)	0.95xVCC	1.05xVCC	V
Ipp		-	20	mA
Vpp	สถานะทำงาน (เริ่มการโปรแกรมการ์ด)	0.975xP	1.025xP	V
Ipp		-	I	mA

ในที่นี้กำหนดให้ค่า P และ I ในการอินเตอร์เฟสการ์ด มีค่าเท่ากับ 5 และ 50 ตามลำดับ

ตารางที่ 2.2 คุณลักษณะทางไฟฟ้าของ VPP ภายใต้การทำงานปกติ

ขา CLK

ใช้รับสัญญาณนาฬิกาที่ส่งมาจากเครื่องอ่าน เพื่อใช้เป็นสัญญาณนาฬิกาให้ระบบของ CPU/MPU card (โดยทั่วไปจะใช้ความถี่ค่า 3.579 เมกะเฮิรตซ์) หรือเพื่อเป็นสัญญาณนาฬิกาอ้างอิงในการเข้าถึงข้อมูลสำหรับอินเตอร์เฟสกับ Memory card (หมายถึงการที่ขา CLK จะถูกใช้เป็นสัญญาณอ้างอิงในการอ่านหรือเขียนข้อมูลแต่ละบิตให้ขา I/O โดยเครื่องอ่านจะต้องสร้างสัญญาณนาฬิกาขึ้น 1 ลูก เมื่อต้องการการรับหรือส่งข้อมูล 1 บิต) คุณลักษณะทางไฟฟ้าของสัญญาณที่ขา CLK ควรจะเป็นตามตารางที่ 2.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญลักษณ์	สภาวะเงื่อนไข	ค่าต่ำสุด	ค่าสูงสุด	หน่วย
Vih [1]	lih(max) = ±200μA	2.4	VCC [2]	V
	lih(max) = ±20μA	0.7x VCC	VCC [2]	V
	lih(max) = ±10μA	VCC-0.7	VCC [2]	V
Vil	lil(max) = ±200μA	0[2]	0.5	V
tr, tf	Cin = 30pF	-	9% ของคาบเวลา (แต่ว่าไม่เกิน 5uS)	V

(1)สำหรับเครื่องอ่านการ์ดบนสภาวะเงื่อนไขที่ต่างกัน 3 กรณี

(2)ระดับแรงดันของสัญญาณที่ขา CLK ควรมีค่าอยู่ระหว่าง 0.3 V และ VCC+0.3V

ตารางที่ 2.3 คุณลักษณะทางไฟฟ้าของสัญญาณที่ขา CLK ภายใต้การทำงานปกติ

ขา RST

ใช้ในการรีเซตสมาร์ตการ์ดให้เริ่มต้นทำงานใหม่ คุณลักษณะทางไฟฟ้าของสัญญาณที่ขา **RST** ควรจะเป็นไปตามตารางที่ 2.4

สัญลักษณ์	สภาวะเงื่อนไข	ค่าต่ำสุด	ค่าสูงสุด	หน่วย
Vih [1]	**lih(max) = ±200μA**	4	VCC [2]	V
	Lih(max) = ±10μA	VCC-0.7	VCC [2]	V
Vil	lil(max) = ±200μA	0[2]	0.6	V

(1)สำหรับเครื่องอ่านการ์ดบนสภาวะเงื่อนไขที่ต่างกัน 2 กรณี

(2)ระดับแรงดันของสัญญาณที่ขา RST ควรมีค่าอยู่ระหว่าง 0.3 V และ VCC+0.3V

ตารางที่ 2.4 คุณลักษณะทางไฟฟ้าของสัญญาณที่ขา RST ภายใต้การทำงานปกติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขา VCC และ ขา GND

ใช้รับไฟเลี้ยงและกราวด์ที่ป้อนมาจากเครื่องอ่าน โดยระดับแรงดันไฟเลี้ยงและกระแสก็ควรจะมีค่าเป็นตามตารางที่ 2.5

สัญลักษณ์	ค่าต่ำสุด	ค่าสูงสุด	หน่วย
VCC	4.75	5.25	V
ICC	-	200	mA

ตารางที่ 2.5 ระดับแรงดันไฟเลี้ยงและกระแสที่ป้อนให้กับสมาร์ตการ์ดภายใต้การทำงานปกติ

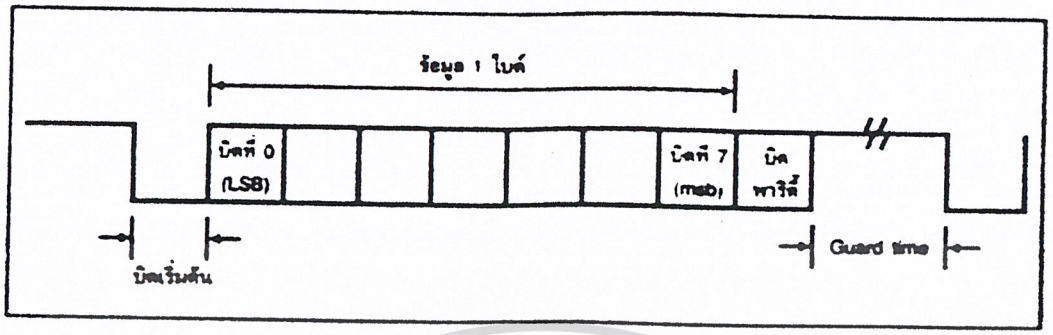
2.1.6.3 รูปแบบการอินเตอร์เฟสของสมาร์ตการ์ดแบบสัมผัส

จากที่ทราบว่าสมาร์ตการ์ดแบบสัมผัสโดยหลัก ๆ มีอยู่ 2 ชนิด คือ MPU/CPU card และ Memory card โดยสมาร์ตการ์ดทั้ง 2 ชนิดนี้จะมีวิธีการอินเตอร์เฟสที่แตกต่างกัน คือ MPU/CPU card จะใช้อินเตอร์เฟสแบบ Asynchronous ส่วน Memory card จะใช้อินเตอร์เฟสแบบ Synchronous

สำหรับกรณีของการอินเตอร์เฟสแบบ **Asynchronous** เป็นการสื่อสารข้อมูลในรูปของตัวอักขระแอสกี (ASCII) โดยผ่านสายข้อมูลไปที่ตำแหน่งขา I/O ของสมาร์ตการ์ดด้วยรูป Asynchronous half duplex mode ตัวอักขระแต่ละตัวมีขนาด 8 บิต การอินเตอร์เฟสแบบนี้ทำงาน

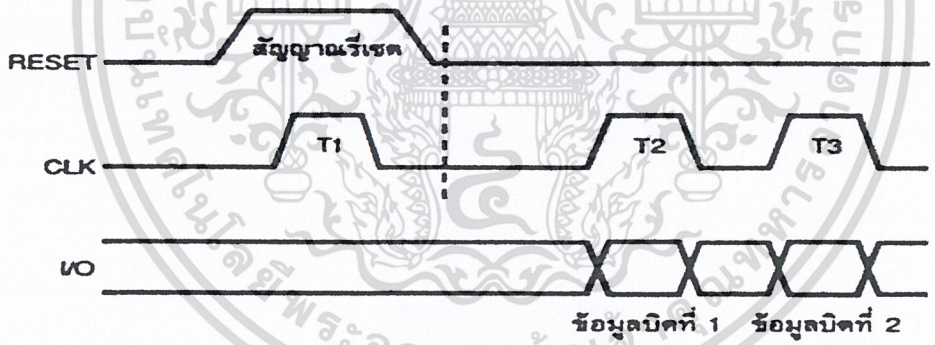
โดยอาศัยสัญญาณนาฬิกาที่มาจากภายนอก ซึ่งโดยมากจะใช้ค่าความถี่ 3.579 เมกะเฮิรตซ์เป็นสัญญาณนาฬิกาของระบบและใช้ป้อนให้วงจรหาร (หาร 372) เพื่อเป็นค่า Baud rate ที่ 9,600 บิตต่อวินาที รูปสัญญาณในการอินเตอร์เฟสมีลักษณะเป็นดังรูปที่ จะเห็นว่าเฟรมข้อมูลแต่ละไบต์นั้นจะประกอบด้วยบิตเริ่มต้น (Start bit) , ข้อมูลขนาด 8 บิต , บิตพาริตี้ (Parity bit) , บิตสิ้นสุด (Stop bit) และ Guard time (การทิ้งช่วงเวลาระหว่างข้อมูลแต่ละชุด) เช่นเดียวกับเฟรมของข้อมูลในการอินเตอร์เฟสแบบ Asynchronous ทั่วไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 รูปสัญญาณสำหรับการอินเตอร์เฟสแบบ Asynchronous

ส่วนในกรณีของการอินเตอร์เฟสแบบ **Synchronous** เป็นการสื่อสารข้อมูลแบบอนุกรม โดยการรับส่งข้อมูลจะทำกันทีละบิต ผ่านสายข้อมูลไปที่ตำแหน่งขา I/O ของสมาร์ตการ์ดด้วยรูป Synchronous half duplex mode ขั้นตอนไม่ว่าจะเป็นการอ่านหรือการเขียนข้อมูลจะถูกทำไปทีละบิตโดยอ้างอิงกับสัญญาณนาฬิกาที่การ์ดได้รับที่ขา CLK เป็นหลัก



รูปที่ 2.7 รูปสัญญาณสำหรับการอินเตอร์เฟสแบบ Synchronous

ขั้นตอนในการอินเตอร์เฟสอย่างง่าย ๆ

ในการอินเตอร์เฟสสมาร์ตการ์ดอย่างง่าย ๆ สามารถทำได้ดังนี้

- ต่อสมาร์ตการ์ดเข้ากับเครื่องอ่านการ์ด และจัดระดับแรงดันที่ขาสัมผัสต่าง ๆ เพื่อสร้างสถานะเริ่มต้นที่เหมาะสมให้กับสมาร์ตการ์ด
- ป้อนสัญญาณรีเซตเข้าไปที่ขา RST

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

• รอรับข้อมูลจาก ATR (Answer To Reset) ที่การ์ดจะตอบกลับมายังเครื่องอ่านที่ขา I/O โดยข้อมูล ATR ที่ได้รับ จะมีขนาดความยาวต่างแต่งกันไปแต่ละบริษัทผู้ผลิต

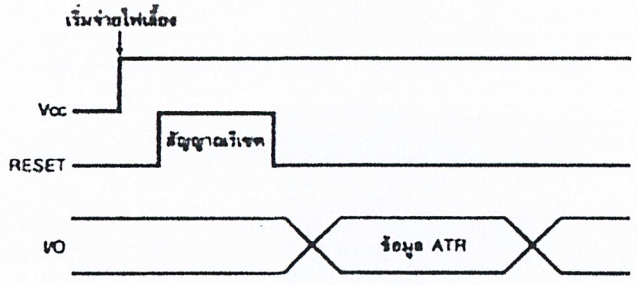
• หลังจากได้รับข้อมูล ATR หรือ ATR byte เรียบร้อยแล้วจะเข้าสู่กระบวนการตีข้อมูล โดยการตีความที่ว่านี้เป็นการตีความข้อมูลแต่ละ ไบต์จากเอกสารของทางบริษัทผู้ผลิต

รีเซตการทำงานของสมาร์ตการ์ด

สมาร์ตการ์ดไม่ควรได้รับการรีเซต จนกว่าขาสัมผัสทั้งหมดจะถูกต่อครบเสียก่อน เพื่อหลีกเลี่ยงความเสียหายที่อาจเกิดกับภายในการ์ด ในการรีเซตการ์ดจะต้องจัดสถานะเริ่มต้นของขาต่าง ๆ ดังนี้

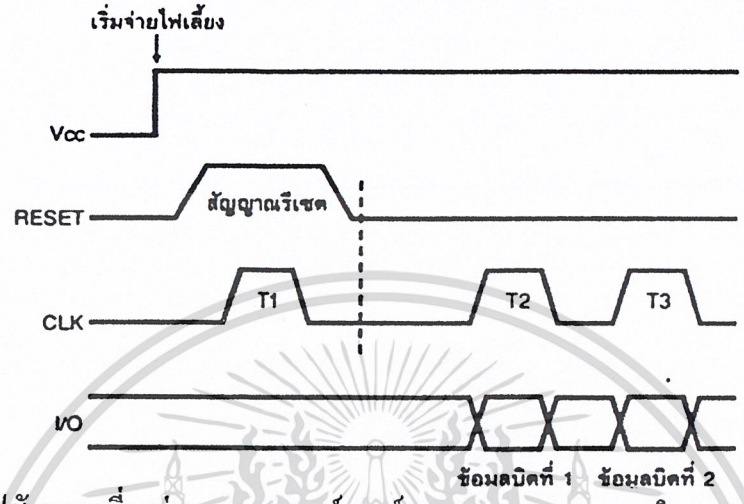
- เริ่มต้นด้วยการเซตให้ขา RST มีสถานะลอจิกต่ำ
- ป้อนแรงดันไฟเลี้ยงมายัง ขา VCC
- ป้อนสัญญาณนาฬิกาความถี่ 3.579 เมกะเฮิรตซ์ ให้กับขา CLK
- เซตให้ขา RST ในลักษณะของการท้อกเกิ้ล
- รอรับข้อมูล ATR ที่ตอบกลับมาจากการ์ด

หลังจากผ่านขั้นตอนข้างต้นแล้ว การ์ดจะส่งข้อมูล ATR ตอบกลับมายังเครื่องอ่าน เพื่อบอกรายละเอียดเกี่ยวกับตัวการ์ดเอง การตอบกลับจะมีกระบวนการที่ต่างกันไปตามชนิดของสมาร์ตการ์ด สำหรับ MPU/CPU card การตอบกลับจะทำโดยใช้การส่งข้อมูลแบบ Asynchronous โดยข้อมูลจะค่อย ๆ ถูกส่งกลับมายังเครื่องอ่านทีละ ไบต์จนครบจำนวน ส่วน Memory card การตอบกลับจะทำโดยใช้การส่งข้อมูลแบบ Synchronous ซึ่งข้อมูลจะค่อย ๆ ถูกอ่านออกมาทีละบิต (โดยอ้างอิงกับสัญญาณ CLK) การรีเซตจะทำให้สถานะของส่วนการทำงานภายในการ์ดทั้งหมดกลับมาอยู่ที่ค่าตั้งต้น สำหรับรูปสัญญาณที่ขาต่าง ๆ ของสมาร์ตการ์ดในขณะทำการรีเซตจะมีลักษณะดังรูปที่ 2.8 และรูปที่ 2.9



รูปที่ 2.8 รูปสัญญาณที่ขาต่าง ๆ ของ สมาร์ตการ์ด แบบ MCU ในระหว่างการรีเซต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 รูปสัญญาณที่ขาต่าง ๆ ของ สมาร์ตการ์ด แบบ Memory card ในระหว่างขั้นตอนการรีเซต

สิ้นสุดขั้นตอนการรีเซต

เมื่อต้องการสิ้นสุดการสื่อสารข้อมูลกันระหว่างเครื่องอ่านและการ์ด เพราะได้รับข้อมูลทั้งหมดมาเรียบร้อยแล้ว หรือต้องการหยุดเนื่องจากพบว่าการ์ดที่ป้อนเข้าผิดประเภทหรือการ์ดถูกดึงออกจากซ็อกเก็ต ในสถานการณ์ดังกล่าวสถานะที่ขาต่าง ๆ ของหน้าสัมผัสสมาร์ตการ์ดจะต้องได้รับการจัดการเพื่อยุติการทำงานของการ์ดด้วย โดยมีวิธีการดังนี้

- เซตให้ขา RST มีสถานะเป็นลอจิกต่ำ
- เซตให้ขา CLK มีสถานะเป็นลอจิกต่ำ
- หยุดการป้อนแรงดันที่หน้าสัมผัส VCC

ATR (Answer To Reset)

ข้อมูล ATR เป็นข้อมูลขนาด 16-17 ไบต์ ที่สมาร์ตการ์ดจะตอบกลับมาหลังจากที่ได้รับการรีเซต โดยข้อมูล ATR จะเป็นส่วนที่บรรจุรายละเอียดเกี่ยวกับสมาร์ตการ์ดใบนั้นๆเอาไว้ ไม่ว่าจะเป็นข้อมูลของผู้ผลิต, ชนิดของการ์ด, พารามิเตอร์ในการอินเตอร์เฟสที่การ์ดต้องการ,รูปแบบโปรโตรคอลที่ใช้งาน เป็นต้น อย่างไรก็ตามการได้ข้อมูล ATR มาจากการรีเซต คงจะไม่สามารถเอามาใช้ประโยชน์ได้ถ้าไม่สามารถตีความข้อมูลดังกล่าวออกมา ซึ่งการตีความก็ต้องอาศัยการอ้างอิงจากเอกสารของผู้ผลิตการ์ดนั้นๆ ว่านิยามความหมายข้อมูลเอาไว้อย่างไร ในตา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

รางที่ 2.6 เป็นตัวอย่างของข้อมูล ATR จาก เอกสารของผู้ผลิตรายหนึ่ง จะเห็นได้ว่าข้อมูล ATR แต่ละไบต์ล้วนมีความหมายซึ่งเราสามรถนำมาใช้ประโยชน์ได้

ATR Byte	ข้อมูล (เลขฐาน 16)	ความหมาย
TS	3B	Synchronization byte
TO	1F	Number of remaining byte
TA1	11	Clock frequency factor F and D
T1	00	Format character
T2	67	Pre-issuing data tag
T3	42	Chip type
T4	41	Operating system version
T5	XX	ข้อมูลเฉพาะของบริษัทผู้ผลิต
T6	XX	
T7	XX	
T8	XX	
T9	XX	
T10	52	Card issuer's data tag
T11	66	Buffer size
T12	XX	Error counter
T13	8X	Set Purse card type code
T14	90	Normal status
T15	00	Normal status

ตารางที่ 2.6 ตัวอย่างของข้อมูล ATR ของสมาร์ตการ์ดจากผู้ผลิตรายหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
50290

2.2.0 PIC (Peripheral Interface Control)

PIC คือ microcontroller อีกตระกูลหนึ่ง ย่อมาจากคำว่า (**Peripheral Interface Control)**ซึ่ง concept ของ microcontroller ตระกูลนี้ก็คือ พยายามรวมเอาทุกอย่างเอาไว้ในตัวของมันไม่ว่าจะเป็น PROGRAM MEMROY, RAM, EEPROM, SERIAL, I2C, PWM, A/D ฯลฯ โดยไม่จำเป็นต้องต่ออุปกรณ์เสริมจากภายนอก ในตัวของ PIC จะมีฟังก์ชันที่ใช้ในการประมวลผลรวมทั้งหน่วยความจำ ซึ่งทำให้มันเหมือนกัน CPU ตัวนึงเลยทีเดียว

2.2.1ความเร็วของ PIC

ภาคของความถี่สัญญาณนาฬิกา ปัจจุบันสามารถทำสัญญาณนาฬิกาได้ที่ 20 MHz ซึ่งทำให้หนึ่งคำสั่งของ PIC ใช้เวลาเพียง 0.25 uSec

2.2.2หน่วยความจำของ PIC

ในอดีตหน่วยความจำของ PIC จะค่อนข้างน้อย คืออยู่ระหว่าง 512 words ถึง 4K words แต่ในปัจจุบัน บริษัท microchip ซึ่งเป็นเจ้าของ PIC ได้พัฒนาจนทำให้ memory ของ PIC มีขนาด

เป็นหลายสิบกิโลไบต์ และมีทีท่าว่าจะขยายได้ใหญ่ขึ้นเรื่อยๆ ในเรื่องของการนับขนาดของหน่วยความจำของ PIC จะนับไม่เหมือนปกติ โดยที่ หนึ่งคำสั่งของ PIC จะมีขนาด 14 bits ดังนั้นเราจะเรียกว่า 1 word ของ PIC จะมีขนาด 14 bits เช่น PIC16F84A ระบุว่ามีหน่วยความจำ 1 K (ซึ่งหมายถึง 1 Kword ถ้าคำนวณให้เป็นแบบ 1 byte = 8 bit จะได้ว่า 1 x 1,024 x 14 = 14,336 bits ดังนั้นก็คือ 14,336/(8 x 1,024) = 1.75K bytes นั่นเอง

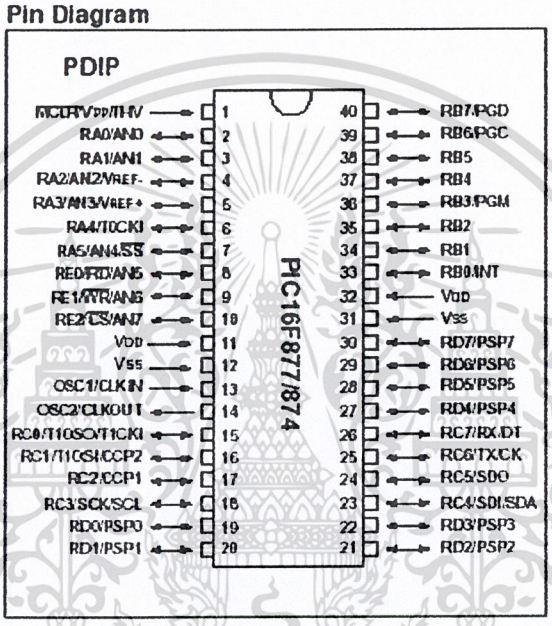
2.2.3 สถาปัตยกรรมของ PIC

ตอนนี้มี 3 สายหลักๆ สมัยก่อนมีแค่สอง คือขึ้นต้นด้วย 16xxx,17xxx และใหม่ล่าสุดคือ 18xxx ถ้าพูดถึง คุณสมบัติที่เหนือกว่าเรียงจากน้อยสุดไปมากสุดก็คือ 16 -> 17 -> 18 คำสั่ง assembly ของ 17 และมี 18 จะมีมากกว่า 16 ทำให้เขียนโปรแกรมได้ง่ายกว่า ราคาก็จะสูงกว่าด้วย แต่ที่เป็นที่นิยมก็คือตระกูล 16xxx

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PIC16F876 คือ เป็นรุ่นที่ได้รับการพัฒนา มาจาก PIC16F84A. มีหน่วยความจำโปรแกรม 8K Words, RAM Memory 368 bytes และ EEPROM 256 bytes. มันมีขาของ timers, Analog Digital converter, Universal communication port....

ทำงานเป็นแบบ Flash memory ดังนั้นทำการ Rewrite ดังนั้นทำการโปรแกรมได้ง่ายโดยใช้เครื่องโปรแกรม PIC



รูปที่ 2.10 PIC 16F877

Pic16f876 specification hardware ของ PIC16F876

RA0-5 : Input/Output port A

RB0-7 : Input/Output port B

RC0-7 : Input/Output port C

RD0-7 : Input/Output port D

RE0-2 : Input/Output port E

AN0-7 : Analog input port

RX : USART Asynchronous Receive

TX : USART Asynchronous Transmit

SCK : Synchronous serial clock input

SCL : Output for both SPI and I^2C modes

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DT : Synchronous Data

CK : Synchronous Clock

SDO : SPI Data Out (SPI mode)

SDI : SPI Data In (SPI mode)

SDA : Data I/O (I^2C mode)

CCP1,2 : Capture1 In/Compare1 Out/PWM1 Out

OSC1/CLKIN : Oscillator In/Ecternal Clock In

OSC2/CLKOUT : Oscillator Out/Clock Out MCLR : Master Clear (Active low Reset)

Vpp : Programming voltage input

THV : High voltage test mode control

VREF+/- : Reference voltage

SS : Slave select for the synchronous serial port

T0CKI : Clock input to Timer0

T1OSO : Timer1 oscillator output

T1OSI :Timer1 oscillator input

T1CKI : Clock input to Timer1

PGD : Serial programming data

PGC : Serial programming clock

PGM : Low voltage programinng input

INT : External interrupt

VDD : Positive supply for logic and I/O pins

Vss : Ground reference for logic and I/O pins

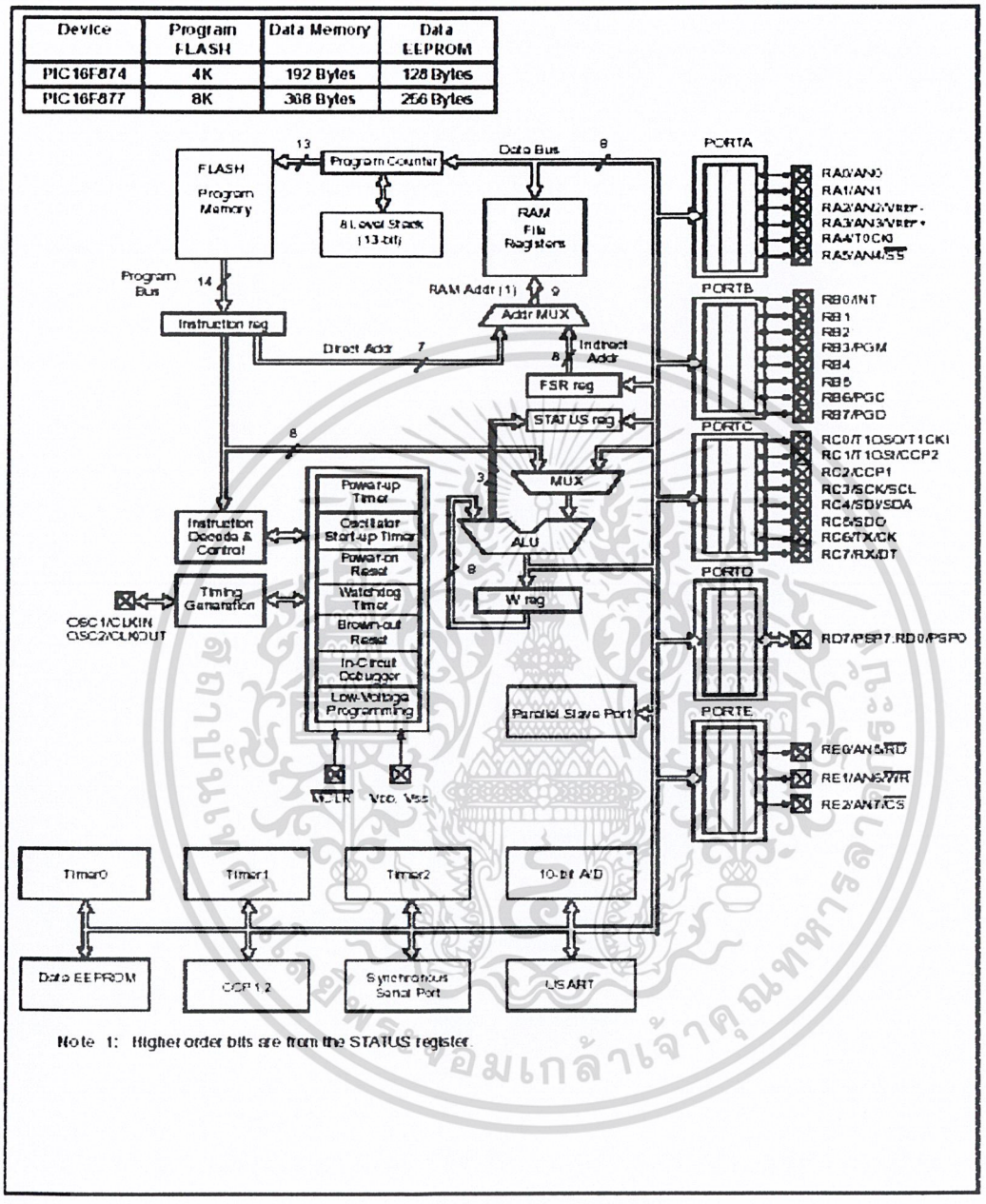
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Key Features PICmicro™ Mid-Range Reference Manual (DS33023)	PIC16F873	PIC16F874	PIC16F876	PIC16F877
Operating Frequency	DC - 20 MHz	DC - 20 MHz	DC - 20 MHz	DC - 20 MHz
Resets (and Delays)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)
FLASH Program Memory (14-bit words)	4K	4K	8K	8K
Data Memory (bytes)	192	192	368	368
EEPROM Data Memory	128	128	256	256
Interrupts	13	14	13	14
I/O Ports	Ports A,B,C	Ports A,B,C,D,E	Ports A,B,C	Ports A,B,C,D,E
Timers	3	3	3	3
Capture/Compare/PWM modules	2	2	2	2
Serial Communications	MSSP, USART	MSSP, USART	MSSP, USART	MSSP, USART
Parallel Communications	—	PSP	—	PSP
10-bit Analog-to-Digital Module	5 input channels	8 input channels	5 input channels	8 input channels
Instruction Set	35 Instructions	35 Instructions	35 Instructions	35 Instructions

ตารางที่ 2.7 เปรียบเทียบ PIC 16F87X

ตารางที่ 2.7 เป็นตารางเปรียบเทียบคุณสมบัติและลักษณะของ PIC แต่ละตัว เพื่อสะดวกในการเลือกใช้ให้เหมาะสมกับงานที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.11 PIC16F874 AND PIC16F877 BLOCK DIAGRAM

ในรูปที่ 2.11 จะเป็นการแสดงโมดูลภายในของ PIC 16f877 เพื่อประกอบการเลือกใช้งาน เนื่องจากในงานบางอย่างที่มีโมดูลอยู่แล้วสามารถหยิบมาใช้ได้เลย ทำให้ง่ายในการเขียนโปรแกรมและการออกแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Name	DIP Pin#	PLCC Pin#	QFP Pin#	I/O/P Type	Buffer Type	Description
OSC1/CLKIN	13	14	30	I	ST/CMOS(4)	Oscillator crystal input/external clock source input.
OSC2/CLKOUT	14	15	31	O	—	Oscillator crystal output. Connects to crystal or resonator in crystal oscillator mode. In RC mode, OSC2 pin outputs CLKOUT which has 1/4 the frequency of OSC1, and denotes the instruction cycle rate.
$\overline{MCLR}$/VPP/THV	1	2	18	I/P	ST	Master clear (reset) input or programming voltage input or high voltage test mode control. This pin is an active low reset to the device.
						PORTA is a bi-directional I/O port.
RA0/AN0	2	3	19	I/O	TTL	RA0 can also be analog input0
RA1/AN1	3	4	20	I/O	TTL	RA1 can also be analog input1
RA2/AN2/VREF-	4	5	21	I/O	TTL	RA2 can also be analog input2 or negative analog reference voltage
RA3/AN3/VREF+	5	6	22	I/O	TTL	RA3 can also be analog input3 or positive analog reference voltage
RA4/T0CKI	6	7	23	I/O	ST	RA4 can also be the clock input to the Timer0 timer/counter. Output is open drain type.
RA5/$\overline{SS}$/AN4	7	8	24	I/O	TTL	RA5 can also be analog input4 or the slave select for the synchronous serial port.
						PORTB is a bi-directional I/O port. PORTB can be software programmed for internal weak pull-up on all inputs.
RB0/INT	33	36	8	I/O	TTL/ST(1)	RB0 can also be the external interrupt pin.
RB1	34	37	9	I/O	TTL	
RB2	35	38	10	I/O	TTL	
RB3/PGM	36	39	11	I/O	TTL	RB3 can also be the low voltage programming input
RB4	37	41	14	I/O	TTL	Interrupt on change pin.
RB5	38	42	15	I/O	TTL	Interrupt on change pin.
RB6/PGC	39	43	16	I/O	TTL/ST(2)	Interrupt on change pin or In-Circuit Debugger pin. Serial programming clock.
RB7/PGD	40	44	17	I/O	TTL/ST(2)	Interrupt on change pin or In-Circuit Debugger pin. Serial programming data.
						PORTC is a bi-directional I/O port.
RC0/T1OSO/T1CKI	15	16	32	I/O	ST	RC0 can also be the Timer1 oscillator output or a Timer1 clock input.
RC1/T1OSI/CCP2	16	18	35	I/O	ST	RC1 can also be the Timer1 oscillator input or Capture2 input/Compare2 output/PWM2 output.
RC2/CCP1	17	19	36	I/O	ST	RC2 can also be the Capture1 input/Compare1 output/PWM1 output.
RC3/SCK/SCL	18	20	37	I/O	ST	RC3 can also be the synchronous serial clock input/output for both SPI and I^2C modes.
RC4/SDI/SDA	23	25	42	I/O	ST	RC4 can also be the SPI Data In (SPI mode) or data I/O (I^2C mode).
RC5/SDO	24	26	43	I/O	ST	RC5 can also be the SPI Data Out (SPI mode).
RC6/TX/CK	25	27	44	I/O	ST	RC6 can also be the USART Asynchronous Transmit or Synchronous Clock.
RC7/RX/DT	26	29	1	I/O	ST	RC7 can also be the USART Asynchronous Receive or Synchronous Data

Legend: I = input O = output I/O = input/output P = power
— = Not used TTL = TTL input ST = Schmitt Trigger input

Note 1: This buffer is a Schmitt Trigger input when configured as an external interrupt.
2: This buffer is a Schmitt Trigger input when used in serial programming mode.
3: This buffer is a Schmitt Trigger input when configured as general purpose I/O and a TTL input when used in the Parallel Slave Port mode (for interfacing to a microprocessor bus).
4: This buffer is a Schmitt Trigger input when configured in RC oscillator mode and a CMOS input otherwise.

ตารางที่ 2.8-1 PIC 16F877 PINOUT DISCRIPTION

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Name	DIP Pin#	PLCC Pin#	QFP Pin#	I/O/P Type	Buffer Type	Description
						PORTD is a bi-directional I/O port or parallel slave port when interfacing to a microprocessor bus.
RD0/PSP0	19	21	38	I/O	ST/TTL(3)	
RD1/PSP1	20	22	39	I/O	ST/TTL(3)	
RD2/PSP2	21	23	40	I/O	ST/TTL(3)	
RD3/PSP3	22	24	41	I/O	ST/TTL(3)	
RD4/PSP4	27	30	2	I/O	ST/TTL(3)	
RD5/PSP5	28	31	3	I/O	ST/TTL(3)	
RD6/PSP6	29	32	4	I/O	ST/TTL(3)	
RD7/PSP7	30	33	5	I/O	ST/TTL(3)	
						PORTE is a bi-directional I/O port.
RE0/RD/AN5	8	9	25	I/O	ST/TTL(3)	RE0 can also be read control for the parallel slave port, or analog input5.
RE1/WR/AN6	9	10	26	I/O	ST/TTL(3)	RE1 can also be write control for the parallel slave port, or analog input6.
RE2/CS/AN7	10	11	27	I/O	ST/TTL(3)	RE2 can also be select control for the parallel slave port, or analog input7.
VSS	12,31	13,34	6,29	P	—	Ground reference for logic and I/O pins.
VDD	11,32	12,35	7,28	P	—	Positive supply for logic and I/O pins.
NC	—	1,17,28, 40	12,13, 33,34		—	These pins are not internally connected. These pins should be left unconnected.

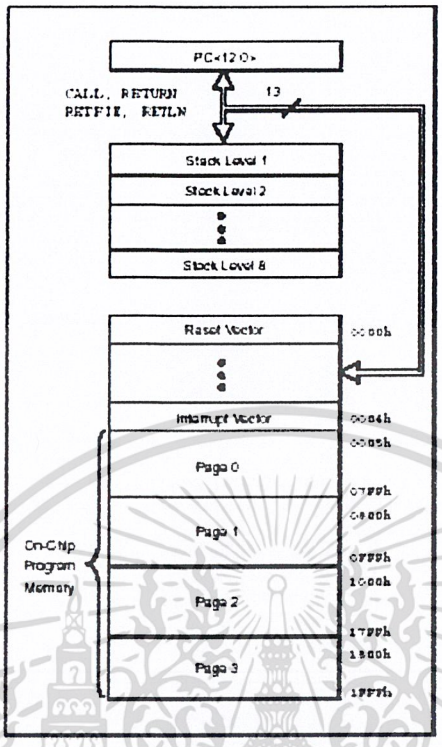
Legend: I = input O = output I/O = input/output P = power
— = Not used TTL = TTL input ST = Schmitt Trigger input

Note 1: This buffer is a Schmitt Trigger input when configured as an external interrupt.
2: This buffer is a Schmitt Trigger input when used in serial programming mode.
3: This buffer is a Schmitt Trigger input when configured as general purpose I/O and a TTL input when used in the Parallel Slave Port mode (for interfacing to a microprocessor bus).
4: This buffer is a Schmitt Trigger input when configured in RC oscillator mode and a CMOS input otherwise.

ตารางที่ 2.8-2 PIC 16F877 PINOUT DISCRIPTION

ตารางที่ 28 จะแสดงถึง ตำแหน่งขา คุณสมบัติและรายละเอียดของขานั้นๆ ว่า มีความสามารถนำไปต่อใช้งานได้อย่างไรบ้าง ซึ่งในบางขาจะมีการต่ออยู่กับโมดูลที่ใช้เฉพาะงาน ซึ่งสามารถที่จะเลือกเซตใช้งานได้

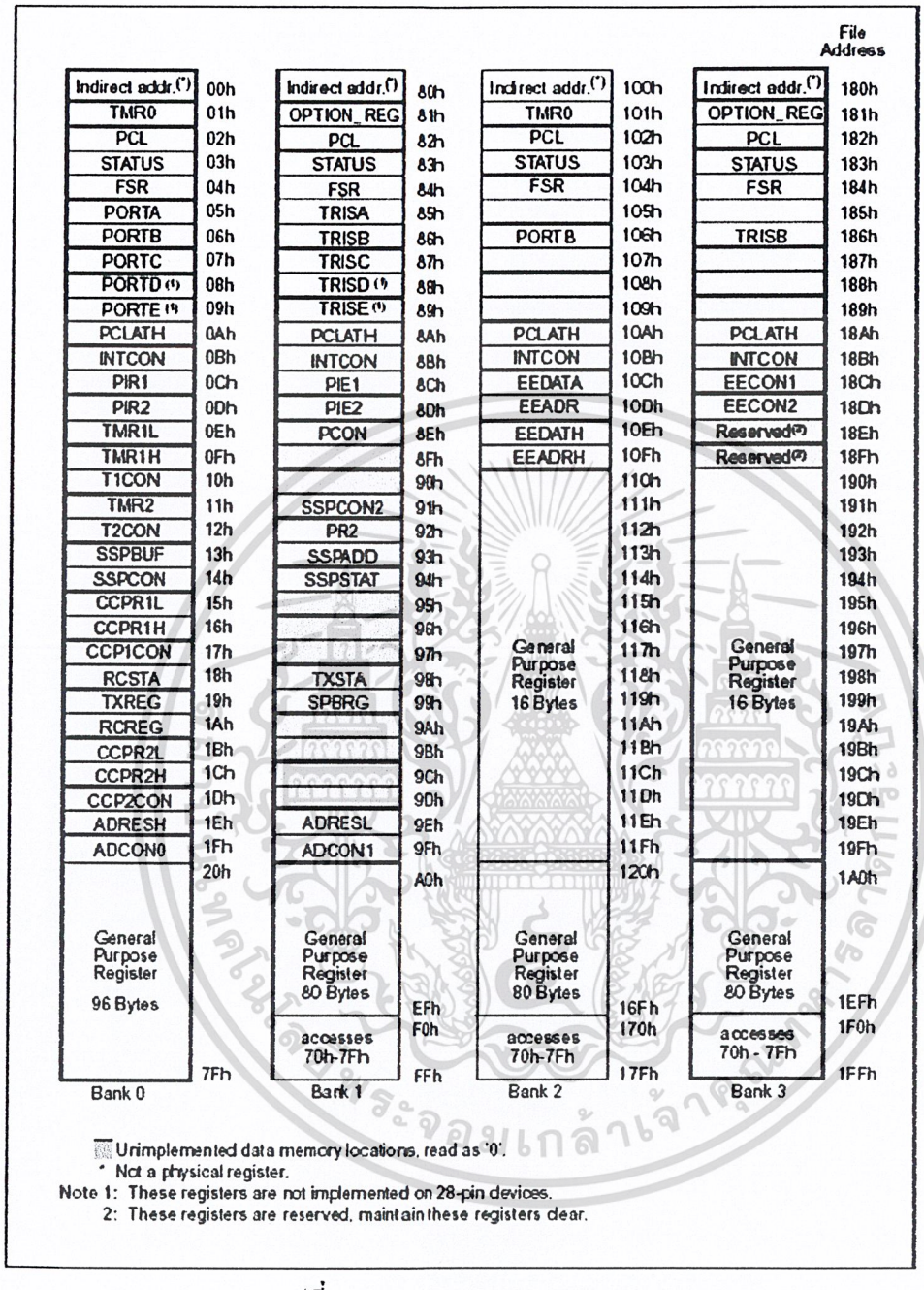
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.12 PIC 16F877 PROGRAM MEMORY MAP AND STACK

ในรูปที่ 2.12 จะแสดงถึง พื้นที่หน่วยความจำในส่วนต่างๆ ทั้ง 4 Page ซึ่งเราจะต้องทราบเพื่อสามารถอ้างถึงตำแหน่ง Page ต่างๆ ได้ว่ามีตำแหน่งอยู่ในช่วงใด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 PIC 16F877 REGISTER FILE MAP

รูปที่ 2.13 จะทำให้เราทราบถึงตำแหน่ง ของรีจีสเตอร์ต่างๆ ว่าแต่ละตัวอยู่ที่ตำแหน่ง และ เพจใด เพื่อสามรถอ้างถึงตำแหน่งได้เมื่อต้องการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets (2)
Bank 0											
00h(4)	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
01h	TMR0	Timer0 module's register								xxxx xxxx	uuuu uuuu
02h(4)	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
03h(4)	STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	000q quuu
04h(4)	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
05h	PORTA	—	—	PORTA Data Latch when written: PORTA pins when read						--0x 0000	--0u 0000
06h	PORTB	PORTB Data Latch when written: PORTB pins when read								xxxx xxxx	uuuu uuuu
07h	PORTC	PORTC Data Latch when written: PORTC pins when read								xxxx xxxx	uuuu uuuu
08h(5)	PORTD	PORTD Data Latch when written: PORTD pins when read								xxxx xxxx	uuuu uuuu
09h(5)	PORTE	—	—	—	—	—	RE2	RE1	RE0	---- -xxx	---- -uuu
0Ah(1,4)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
0Bh(4)	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	PSPIF(3)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
0Dh	PIR2	—	(6)	—	EEIF	BCLIF	—	—	CCP2IF	-r-0 0--0	-r-0 0--0
0Eh	TMR1L	Holding register for the Least Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu
0Fh	TMR1H	Holding register for the Most Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu
10h	T1CON	—	—	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	TMR1CS	TMR1ON	--00 0000	--uu uuuu
11h	TMR2	Timer2 module's register								0000 0000	0000 0000
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000
13h	SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	uuuu uuuu
14h	SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
15h	CCPR1L	Capture/Compare/PWM Register1 (LSB)								xxxx xxxx	uuuu uuuu
16h	CCPR1H	Capture/Compare/PWM Register1 (MSB)								xxxx xxxx	uuuu uuuu
17h	CCP1CON	—	—	CCP1X	CCP1Y	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	--00 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
19h	TXREG	USART Transmit Data Register								0000 0000	0000 0000
1Ah	RCREG	USART Receive Data Register								0000 0000	0000 0000
1Bh	CCPR2L	Capture/Compare/PWM Register2 (LSB)								xxxx xxxx	uuuu uuuu
1Ch	CCPR2H	Capture/Compare/PWM Register2 (MSB)								xxxx xxxx	uuuu uuuu
1Dh	CCP2CON	—	—	CCP2X	CCP2Y	CCP2M3	CCP2M2	CCP2M1	CCP2M0	--00 0000	--00 0000
1Eh	ADRESH	A/D Result Register High Byte								xxxx xxxx	uuuu uuuu
1Fh	ADCON0	ADCS1	ADCS0	CHS2	CHS1	CHS0	$\overline{GO}$/DONE	—	ADON	0000 00-0	0000 00-0

Legend: x = unknown, u = unchanged, q = value depends on condition, - = unimplemented read as '0', r = reserved. Shaded locations are unimplemented, read as '0'.

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<12:8> whose contents are transferred to the upper byte of the program counter.

2: Other (non power-up) resets include external reset through MCLR and Watchdog Timer Reset.

3: Bits PSPIE and PSPIF are reserved on the 28-pin devices; always maintain these bits clear.

4: These registers can be addressed from any bank.

5: PORTD, PORTE, TRISD and TRISE are not physically implemented on the 28-pin devices, read as '0'.

6: PIR2<6> and PIE2<6> are reserved on these devices; always maintain these bits clear.

ตารางที่ 2.9 SPECIAL FUNCTION REGISTER SUMMARY(BANK 0)

ตารางที่ 2.9 นี้จะแสดงถึงบิตที่อยู่ภายใน SPECIAL FUNCTION REGISTER ที่อยู่ในแบงค์ 0 เพื่อนำไปประกอบการเซตค่าของ SPECIAL FUNCTION REGISTER เพื่อใช้ในงานต่างๆ ในการเขียนโปรแกรมซึ่งตารางนี้มีความสำคัญมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets (2)
Bank 1											
80h(4)	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
81h	OPTION_REG	$\overline{RBPU}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
82h(4)	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
83h(4)	STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	000q quuu
84h(4)	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
85h	TRISA	—	—	PORTA Data Direction Register						--11 1111	--11 1111
86h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
87h	TRISC	PORTC Data Direction Register								1111 1111	1111 1111
88h(5)	TRISD	PORTD Data Direction Register								1111 1111	1111 1111
89h(5)	TRISE	IBF	OBF	IBOV	PSPMODE	—	PORTE Data Direction Bits			0000 -111	0000 -111
8Ah(1,4)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
8Bh(4)	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
8Ch	PIE1	PSPIE(3)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
8Dh	PIE2	—	(6)	—	EEIE	BCLIE	—	—	CCP2IE	-r-0 0--0	-r-0 0--0
8Eh	PCON	—	—	—	—	—	—	$\overline{POR}$	BOR	---- --qq	---- --uu
8Fh	—	Unimplemented								—	—
90h	—	Unimplemented								—	—
91h	SSPCON2	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	0000 0000	0000 0000
92h	PR2	Timer2 Period Register								1111 1111	1111 1111
93h	SSPADD	Synchronous Serial Port (I²C mode) Address Register								0000 0000	0000 0000
94h	SSPSTAT	SMP	CKE	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF	0000 0000	0000 0000
95h	—	Unimplemented								—	—
96h	—	Unimplemented								—	—
97h	—	Unimplemented								—	—
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000
9Ah	—	Unimplemented								—	—
9Bh	—	Unimplemented								—	—
9Ch	—	Unimplemented								—	—
9Dh	—	Unimplemented								—	—
9Eh	ADRESL	A/D Result Register Low Byte								xxxx xxxx	uuuu uuuu
9Fh	ADCON1	ADFM	—	—	—	PCFG3	PCFG2	PCFG1	PCFG0	0--- 0000	0--- 0000

Legend: x = unknown, u = unchanged, q = value depends on condition, - = unimplemented read as '0', r = reserved. Shaded locations are unimplemented, read as '0'.

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<12:8> whose contents are transferred to the upper byte of the program counter.

2: Other (non power-up) resets include external reset through MCLR and Watchdog Timer Reset.

3: Bits PSPIE and PSPIF are reserved on the 28-pin devices; always maintain these bits clear.

4: These registers can be addressed from any bank.

5: PORTD, PORTE, TRISD, and TRISE are not physically implemented on the 28-pin devices, read as '0'.

6: PIR2<6> and PIE2<6> are reserved on these devices; always maintain these bits clear.

ตารางที่ 2.10 SPECIAL FUNCTION REGISTER SUMMARY(BANK 1)

ตารางที่ 2.10 นี้จะแสดงถึงบิตที่อยู่ภายใน SPECIAL FUNCTION REGISTER ที่อยู่ในแบงค์ 1 เพื่อนำไปประกอบการเซตค่าของ SPECIAL FUNCTION REGISTER เพื่อใช้ในงานต่างๆ ในการเขียนโปรแกรมซึ่งตารางนี้มีความสำคัญมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other resets (2)
Bank 2											
100h(4)	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
101h	TMR0	Timer0 module's register								xxxx xxxx	uuuu uuuu
102h(4)	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
103h(4)	STATUS	IRP	RP1	RP0	TO	PD	Z	DC	C	0001 1xxx	000q quuu
104h(4)	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
105h	—	Unimplemented								—	—
106h	PORTB	PORTB Data Latch when written: PORTB pins when read								xxxx xxxx	uuuu uuuu
107h	—	Unimplemented								—	—
108h	—	Unimplemented								—	—
109h	—	Unimplemented								—	—
10Ah(1,4)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
10Bh(4)	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
10Ch	EEDATA	EEPROM data register								xxxx xxxx	uuuu uuuu
10Dh	EEADR	EEPROM address register								xxxx xxxx	uuuu uuuu
10Eh	EEDATH	—	—	EEPROM data register high byte						xxxx xxxx	uuuu uuuu
10Fh	EEADRH	—	—	—	EEPROM address register high byte					xxxx xxxx	uuuu uuuu
Bank 3											
180h(4)	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
181h	OPTION_REG	RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
182h(4)	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
183h(4)	STATUS	IRP	RP1	RP0	TO	PD	Z	DC	C	0001 1xxx	000q quuu
184h(4)	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
185h	—	Unimplemented								—	—
186h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
187h	—	Unimplemented								—	—
188h	—	Unimplemented								—	—
189h	—	Unimplemented								—	—
18Ah(1,4)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
18Bh(4)	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
18Ch	EECON1	EEPGD	—	—	—	WRERR	WREN	WR	RD	x--- x000	x--- u000
18Dh	EECON2	EEPROM control register2 (not a physical register)								---- ----	---- ----
18Eh	—	Reserved maintain clear								0000 0000	0000 0000
18Fh	—	Reserved maintain clear								0000 0000	0000 0000

Legend: x = unknown, u = unchanged, q = value depends on condition, - = unimplemented read as '0', r = reserved. Shaded locations are unimplemented, read as '0'.

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<12:8> whose contents are transferred to the upper byte of the program counter.
2: Other (non power-up) resets include external reset through MCLR and Watchdog Timer Reset.
3: Bits PSPIE and PSPIF are reserved on the 28-pin devices; always maintain these bits clear.
4: These registers can be addressed from any bank.
5: PORTD, PORTE, TRISD, and TRISE are not physically implemented on the 28-pin devices, read as '0'.
6: PIR2<6> and PIE2<6> are reserved on these devices; always maintain these bits clear.

ตารางที่ 2.11 SPECIAL FUNCTION REGISTER SUMMARY(BANK 2-3)

ตารางที่ 2.11 นี้จะแสดงถึงบิตที่อยู่ภายใน SPECIAL FUNCTION REGISTER ที่อยู่ในแบงค์ 2-3 เพื่อนำใบประกอบการเซตค่าของ SPECIAL FUNCTION REGISTER เพื่อใช้ในงานต่างๆ ในการเขียนโปรแกรมซึ่งตารางนี้มีความสำคัญมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.4 คุณสมบัติทางเทคนิค

สามารถแบ่งออกเป็น 3 ส่วนประกอบคือ **หน่วยประมวลผลกลาง (Central Prosessing Unit),ส่วนเพอริเฟอรัล(Peripheral)** และ **คุณสมบัติพิเศษอื่นๆ**

2.2.4.1คุณสมบัติทางเทคนิคของหน่วยประมวลผลกลางภายใน PIC 16F877

1. หน่วยประมวลผลกลางเป็นแบบ High-Performance RISC
2. มีคำสั่งเพียง 35 คำสั่งขนาด 14 บิต
3. ทุกคำสั่งใช้การปะมวลผลเพียง 1 ไซเกิลของสัญญาณนาฬิกา หรือประมาณ 250 นาโนวินาทีที่สัญญาณนาฬิกาความถี่ 4 MHz ยกเว้นชุดคำสั่งกระโดดจะใช้เวลา 2 ไซเกิลของสัญญาณนาฬิกา
4. ประมวลผลข้อมูลขนาด 8 บิต
5. มีรีจีสเตอร์ฟังก์ชั่นพิเศษ 32 ตัว
6. มีสแต็ก 8 ระดับ
7. มีโหมดการอ้างอิงแอดเดรส 3 โหมด คือ แบบโดยตรง แบบโดยอ้อม และ แบบสัมพันธ์
8. หน่วยความจำข้อมูล (data memory) เป็นแบบอีอีพรอมสามารถลบและเขียนใหม่ได้ประมาณล้านครั้งและเก็บข้อมูลได้ 40 ปี
9. ขนาดหน่วยความจำโปแกรมซึ่งเป็นแบบเฟลชมีขนาด 8 กิโลเวิร์ด (1 เวิร์ด ของ PIC 16F877 มีขนาด 14 บิต) , หน่วยความจำอีอีพรอมภายใน 256 ไบต์ และหน่วยความจำแรม 368 ไบต์

2.2.4.2 คุณสมบัติทางเทคนิคของเฟอริเฟอรัลใน PIC 16F877

1. มีขาอินพุตเอาท์พุต 22 ขา สามารถกำหนดเป็นอินพุตเอาท์พุตได้อย่างอิสระ
2. กระแสซิงก์/ซอร์สของแต่ละขาอินพุตเอาท์พุตสูงพอที่จะขับ LED ได้โดยตรง
3. กระแสซิงก์สูงสุด 25 mA ต่อขา
4. กระแสเซอร์สูงสุด 20 mA ต่อขา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. มีไทเมอร์/เคาน์เตอร์ขนาด 8 บิตคือ TMRO,TMR1และTMR2 พร้อมกับปรีสเกลเลอร์ขนาด 8 บิตที่สามารถโปแกรมได้

2.2.4.3 คุณสมบัติอื่น

1. มีเพาเวอร์ออนรีเซตในตัว
2. มีเพาเวอร์อัปไทเมอร์ในตัว
3. มีออสซิลเลเตอร์สตาร์ตอัปไทเมอร์
4. มีวอตช์ด็อกไทเมอร์ พร้อมกับวงจรออสซิลเลเตอร์ RC ภายในเพื่อช่วยให้การทำงานของไมโครคอนโทรลเลอร์มีเสถียรภายยิ่งขึ้น
5. ป้องกันการคอกลอกข้อมูลในหน่วยความจำโปรแกรม
6. มีโหมดประหยัดพลังงานหรือโหมดสลีป
7. สามรถเลือกวงจรออสซิลเลเตอร์ที่ใช้กำหนดการทำงานได้
8. การเขียนข้อมูลเข้าสู่หน่วยความจำในไมโครคอลโทรลเลอร์เป็นแบบอนุกรมผ่านขาใช้งานเพียง 2 ขา
9. มีขาสำหรับแปลงอนาล็อกเป็นดิจิตอล
10. มีขา TX สำหรับส่งข้อมูล และขา RX สำหรับรับข้อมูล
11. ย่านไฟเลี้ยง 2.0-5.5V
12. ปริมาณการใช้กระแสไฟฟ้า

    < 2mA ที่ไฟเลี้ยง 5 V สัญญาณนาฬิกาความถี่ 4 MHz

    20 uA ที่ไฟเลี้ยง 3 V สัญญาณนาฬิกาความถี่ 32 kHz

    < 1 uA ที่ไฟเลี้ยง 3 V ขณะสแตนด์บาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Mnemonic, Operands		Description	Cycles	14-Bit Opcode MSb			LSb	Status Affected	Notes
BYTE-ORIENTED FILE REGISTER OPERATIONS									
ADDWF	f, d	Add W and f	1	00	0111	dfff	ffff	C,DC,Z	1,2
ANDWF	f, d	AND W with f	1	00	0101	dfff	ffff	Z	1,2
CLRF	f	Clear f	1	00	0001	lfff	ffff	Z	2
CLRW	-	Clear W	1	00	0001	0xxx	xxxx	Z	
COMF	f, d	Complement f	1	00	1001	dfff	ffff	Z	1,2
DECF	f, d	Decrement f	1	00	0011	dfff	ffff	Z	1,2
DECFSZ	f, d	Decrement f, Skip if 0	1(2)	00	1011	dfff	ffff		1,2,3
INCF	f, d	Increment f	1	00	1010	dfff	ffff	Z	1,2
INCFSZ	f, d	Increment f, Skip if 0	1(2)	00	1111	dfff	ffff		1,2,3
IORWF	f, d	Inclusive OR W with f	1	00	0100	dfff	ffff	Z	1,2
MOVF	f, d	Move f	1	00	1000	dfff	ffff	Z	1,2
MOVWF	f	Move W to f	1	00	0000	lfff	ffff		
NOP	-	No Operation	1	00	0000	0xx0	0000		
RLF	f, d	Rotate Left f through Carry	1	00	1101	dfff	ffff	C	1,2
RRF	f, d	Rotate Right f through Carry	1	00	1100	dfff	ffff	C	1,2
SUBWF	f, d	Subtract W from f	1	00	0010	dfff	ffff	C,DC,Z	1,2
SWAPF	f, d	Swap nibbles in f	1	00	1110	dfff	ffff		1,2
XORWF	f, d	Exclusive OR W with f	1	00	0110	dfff	ffff	Z	1,2
BIT-ORIENTED FILE REGISTER OPERATIONS									
BCF	f, b	Bit Clear f	1	01	00bb	bfff	ffff		1,2
BSF	f, b	Bit Set f	1	01	01bb	bfff	ffff		1,2
BTFSC	f, b	Bit Test f, Skip if Clear	1 (2)	01	10bb	bfff	ffff		3
BTFSS	f, b	Bit Test f, Skip if Set	1 (2)	01	11bb	bfff	ffff		3
LITERAL AND CONTROL OPERATIONS									
ADDLW	k	Add literal and W	1	11	111x	kkkk	kkkk	C,DC,Z	
ANDLW	k	AND literal with W	1	11	1001	kkkk	kkkk	Z	
CALL	k	Call subroutine	2	10	0kkk	kkkk	kkkk		
CLRWDT	-	Clear Watchdog Timer	1	00	0000	0110	0100	$\overline{TO}$,$\overline{PD}$	
GOTO	k	Go to address	2	10	1kkk	kkkk	kkkk		
IORLW	k	Inclusive OR literal with W	1	11	1000	kkkk	kkkk	Z	
MOVLW	k	Move literal to W	1	11	00xx	kkkk	kkkk		
RETFIE	-	Return from interrupt	2	00	0000	0000	1001		
RETLW	k	Return with literal in W	2	11	01xx	kkkk	kkkk		
RETURN	-	Return from Subroutine	2	00	0000	0000	1000		
SLEEP	-	Go into standby mode	1	00	0000	0110	0011	$\overline{TO}$,$\overline{PD}$	
SUBLW	k	Subtract W from literal	1	11	110x	kkkk	kkkk	C,DC,Z	
XORLW	k	Exclusive OR literal with W	1	11	1010	kkkk	kkkk	Z	

Note 1: When an I/O register is modified as a function of itself (e.g., MOVF PORTB, 1), the value used will be that value present on the pins themselves. For example, if the data latch is '1' for a pin configured as input and is driven low by an external device, the data will be written back with a '0'.

2: If this instruction is executed on the TMR0 register (and, where applicable, d = 1), the prescaler will be cleared if assigned to the Timer0 Module.

3: If Program Counter (PC) is modified or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a NOP.

Note: Additional information on the mid-range instruction set is available in the PICmicro™ Mid-Range MCU Family Reference Manual (DS33023).

ตารางที่ 2.12 คำสั่ง ของ pic 16F877

ตาราง คำสั่งจะมีคำสั่งทั้งหมดอยู่ 35 คำสั่ง โดยสามารถดูไซเกิลการทำงานของคำสั่งแต่ละคำสั่ง หรือผลที่คำสั่งแต่ละคำสั่งทำให้เกิด ได้จากตารางที่ 2.12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.0 พอร์ตสื่อสารข้อมูลอนุกรม RS 232- C

(Female at the cable side) (Male at the computer side)

รูปที่ 2.14 แสดงตำแหน่งขาของ DB-9

2.3.1 คุณสมบัติของ RS-232 C

อัตราการรับส่งข้อมูล	: 0-20,000 บิตต่อวินาที
ระดับแรงดันเอาท์พุตสูงสุดในสภาวะไม่มีโหลด	: -25 โวลท์ (ลอจิก "1")
	+25 โวลท์(ลอจิก "0")
ระดับแรงดันเอาท์พุตสำหรับโหลด3-7 กิโลโอห์ม	: ลอจิก"1" –15 โวลท์(7 กิโล
	โอห์ม)
	- 5 โวลท์(3 กิโลโอห์ม)ลอ
	จิก"0" " +15 โวลท์(7 กิโล
	โอห์ม)
	+5 โวลท์(3กิโลโอห์ม)
กระแสเอาท์พุตเมื่อลัดวงจร	: สูงสุด 500 มิลลิแอมป์
เอาท์พุตอิมพิแดนซ์เมื่อไม่จ่ายไฟเลี้ยง	: ต่ำสุด 300 โอห์ม
สลูว์เรตทางเอาท์พุตสูงสุด	: 300 โวลท์/ไมโครวินาที
ความต้านทานอินพุตของภาครับ	: สูงสุด 7 กิโลโอห์ม
	ต่ำสุด 3 กิโลโอห์ม
ค่าความจุอินพุตของภาครับ	: สูงสุด 2500 พิโกฟารัด
ย่านแรงดันอินพุตของภาครับ	: -25 โวลท์ ถึง +25 โวลท์

2.3.2 การจัดขาสัญญาณของ RS-232 C

9 PIN D-SUB MALE at the Computer.

Pin	Name	Dir	Description
1	CD	←	Carrier Detect

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2	RXD	←	Receive Data
3	TXD	→	Transmit Data
4	DTR	→	Data Terminal Ready
5	GND	—	System Ground
6	DSR	←	Data Set Ready
7	RTS	→	Request to Send
8	CTS	←	Clear to Send
9	RI	←	Ring Indicator

ตาราง2.13 แสดงรายละเอียดการทำงาน RS 232

2.3.3 รูปแบบของข้อมูลอนุกรมและอัตราบอดในการสื่อสารข้อมูลอนุกรม

อัตราบอด (band rate) คือความเร็วในการรับส่งข้อมูลอนุกรม มีหน่วยเป็นบิตต่อวินาที

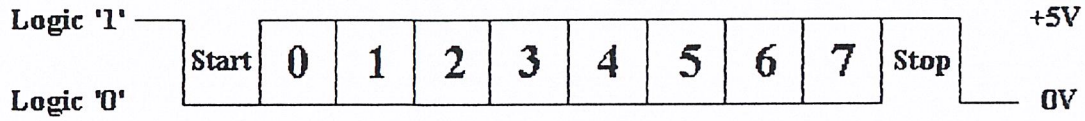
อัตราบอด	ช่วงเวลาของแต่ละบิต
110	9.091 ms
150	6.67 ms
300	3.33 ms
600	1.67 ms
1200	833 µs
2400	417 µs
4800	208 µs
9600	104 µs
19200	52.08 µs

ตารางที่ 2.14 อัตราและช่วงเวลาของแต่ละบิตข้อมูลในการสื่อสารข้อมูลอนุกรม

RS-232 Waveforms232

การสื่อสารโดย RS-232 เป็นการสื่อสารแบบ asynchronous หมายความว่าสัญญาณ clock ที่ใช้ควบคุมจังหวะไม่ได้ส่งไปพร้อมกับ Data แต่จะใช้ start bit เป็นตัว sync.ในแต่ละ word ของการสื่อสารและใช้สัญญาณ clock ภายในของแต่ละด้านเป็นตัวให้จังหวะเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



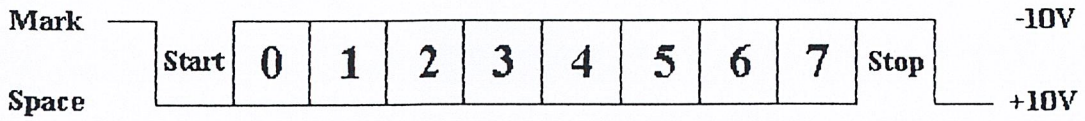
รูปที่ 2.15 TTL/CMOS Serial Logic Waveform

รูปที่ 2.15 แสดงลักษณะของสัญญาณจาก UART เมื่อใช้ format แบบ 8N1 คือ 8 data bits ไม่มี parity bit และมี 1 stop bit ขณะที่ idle จะอยู่ในสถานะ "Mark" หรือ logic "1" การส่งจะเริ่มจากการส่ง start bit คือ logic "0" และตามด้วย LSB bit จนหมด data bits และถ้ามี parity bit ก็จะส่งที่จุดนี้แล้วลงท้ายด้วย stop bit ซึ่งมีค่าเป็น logic "1" ในรูปได้แสดง bit ที่ต่อถัดจาก stop bit ซึ่งมีค่าเป็น logic "0" หมายความว่าเป็น start bit ของ การส่ง word ถัดไป แต่ถ้ายังไม่มีการส่ง word ถัดไป ก็ต้องอยู่ในสถานะของ logic "1" ซึ่งเป็นสถานะของ idle และถ้าสายอยู่ในสถานะของ logic "0" นานกว่าเวลาของการส่ง 1 full word ระบบจะถือว่าเป็นสัญญาณ "Break" เพื่อหยุดการสื่อสาร ดังนั้นต้องไม่ลืมที่จะสั่งในสายกลับสู่สถานะ idle เมื่อสิ้นสุดการส่ง

การรับ-ส่งข้อมูลในลักษณะนี้เรียกว่าแบบ frame คือมีกรอบปิดล้อมข้อมูลไว้ด้วย start bit และ stop bit

RS-232 Level Converters

สัญญาณ RS-232 มีค่าแรงไฟต่างจากที่ใช้ใน UART ดังแสดงใน Figure 5. ดังนั้นจึงต้องมี converter เพื่อแปลงระดับสัญญาณให้เหมาะสมก่อนที่จะเชื่อมต่อกับ serial port หรือ RS-232 port ของ computer

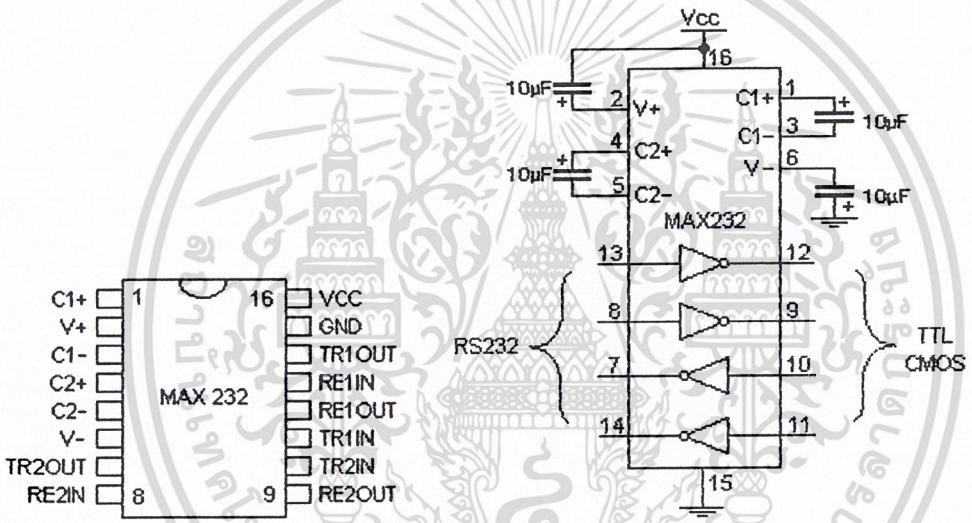


รูปที่ 2.16 RS-232 Logic Waveform.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับสัญญาณ RS-232 นั้น logic "0" จะมีค่า +3 V ถึง +25 V และ logic "1" จะมีค่า –3 V ถึง –25 V ส่วนค่าระหว่าง –3 V ถึง +3 V เป็นค่า undefined ระดับสัญญาณนี้ใช้กับทุกสัญญาณ ไม่ใช่เฉพาะสัญญาณรับ-ส่งข้อมูลเท่านั้นแต่ยังรวมถึงสัญญาณควบคุมต่าง ๆ เช่น DTR, RTS, CTS, DCD, DSR เป็นต้น

IC ที่ใช้จะเป็นเบอร์ MAX -232 ซึ่งมีวงจร charge pump สามารถสร้างไฟ +10 V และ –10 V จากไฟ +5 V ได้ พร้อมทั้งมี 2 Tx และ 2 Rx อยู่ใน package เดียวกัน และรองรับ baud rate ได้ถึง 120 Kbps จึงสะดวกมากเพราะใช้ IC เพียงตัวเดียว รูปข้างล่างคือ MAX –232



รูปที่ 2.17 MAX-232

ส่วนการที่เราจะนำข้อมูลมาใช้งานก็ต้องแปลงเป็น parallel ก่อนซึ่งเป็นหน้าที่ของ UART ซึ่งปัจจุบัน microprocessor และ microcontroller มักจะมี serial communication interface (SCI) อยู่ในตัว แต่อาจจะมีงานบางอย่างที่ไม่ได้ใช้ microcontroller และต้องการ process ข้อมูลกับ serial comm. เช่น ต่อ ADC เข้ากับ UART หรือต่อ LCD display เข้ากับ Serial comm. ก็ต้องใช้ UART ช่วย เช่นเบอร์ 8250 หรือ 16550A หรือเบอร์อื่น ๆ ที่ได้กล่าวมาแล้ว แต่มี UART อีกพวกหนึ่งที่แยก Tx bus กับ Rx bus ออกจากกัน ทำให้มีความคล่องตัวมากขึ้น ซึ่งจะกล่าวในหัวข้อต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.4 รูปแบบการใช้งาน UART

ความเร็วในการส่ง

เราใช้ความเร็วในการส่งขนาด 9600 bps (bps ย่อมาจาก bit per second) พูดกันง่ายๆ ก็คือ ส่งข้อมูลขนาด 9600 bit ใน 1 วินาที ซึ่งก็หมายความได้อีกอย่างหนึ่งว่าสัญญาณขนาด 1 bit มีความกว้าง (1/9600) = 0.104 msec

REGISTER ของ PIC ที่เกี่ยวกับ SERIAL COMMUNICATION

เนื่องจากเราจะทำการส่งแบบ Asynchronous ดังนั้นเราจะพิจารณา register ต่างๆ ในกรณีแบบ Asynchronous

TABLE 10-7 REGISTERS ASSOCIATED WITH ASYNCHRONOUS RECEPTION

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other Resets
0Ch	PIR1	PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
1Ah	RCREG	USART Receive Register								0000 0000	0000 0000
8Ch	PIE1	PSPIE(1)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
99h	SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

Legend: x = unknown, - = unimplemented locations read as '0'. Shaded cells are not used for Asynchronous Reception.
Note 1: Bits PSPIE and PSPIF are reserved on the 28-pin devices, always maintain these bits clear.

ตารางที่ 2.15 REGISTER ของ PIC ที่เกี่ยวกับ SERIAL COMMUNICATION

PIR1 register บิตที่เกี่ยวข้องก็คือ RCIF และ TXIF

R/W-0	R/W-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF
bit7							bit0

ตารางที่ 2.16 PIR1 register บิตที่เกี่ยวข้องก็คือ RCIF และ TXIF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อ RCIF = 1 หมายถึง buffer ที่ทำหน้าที่รับข้อมูล serial นั้นเต็ม

เมื่อ RCIF = 0 หมายถึง buffer ที่ทำหน้าที่รับข้อมูล serial นั้นว่าง

เมื่อ TXIF = 1 หมายถึง buffer ที่ทำหน้าที่ส่งข้อมูล serial นั้นเต็ม

เมื่อ TXIF = 0 หมายถึง buffer ที่ทำหน้าที่ส่งข้อมูล serial นั้นว่าง

RCSTA register เป็น register ที่เกี่ยวข้องกับการรับข้อมูล

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R-0	R-0	R-x
SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D
bit7							bit0

ตารางที่ 2.17 RCSTA register เป็น register ที่เกี่ยวข้องกับการรับข้อมูล

Bit7 SPEN

= 1 หมายถึง Serial port enable (อนุญาตให้ใช้งาน serial port ได้)

= 0 หมายถึง Serial port disble

Bit6 RX9

= 1 หมายถึง ขนาดข้อมูล 1 bytes ที่รับมีขนาด 9 bits

= 0 หมายถึง ขนาดข้อมูล 1 bytes ที่รับมีขนาด 8 bits

Bit5 SREN

= เราไม่สนใจ bit นี้ในกรณีที่ใช้แบบ Asynchronous

Bit4 CREN

= 1 หมายถึง กำหนดให้มีการรับข้อมูลที่เข้ามาอย่างต่อเนื่อง

= 0 หมายถึง ไม่ให้มีการรับข้อมูลที่เข้ามาอย่างต่อเนื่อง

Bit3 ADDEN

= เนื่องจากเราส่งข้อมูลแบบ 8 bit ดังนั้นเราจะไม่สนใจ บิตนี้

Bit2 FERR

= 1 หมายถึง มี Frame error เกิดขึ้นกับข้อมูลที่ได้รับ

= 0 หมายถึง ไม่มี Frame error เกิดขึ้นกับข้อมูลที่ได้รับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Bit1 OERR

= 1 หมายถึง มี Overrun error เกิดขึ้นกับข้อมูลที่ได้รับ

= 0 หมายถึง ไม่มี Overrun error เกิดขึ้นกับข้อมูลที่ได้รับ

Bit0 RX9D

= เป็นบิตที่ 9 ของข้อมูลในกรณีที่ข้อมูลเป็นแบบ 9 bits

RCREG register คือข้อมูลขนาด 1 byte ที่รับได้จากภายนอกในขณะนั้น

TXSTA register เป็น register ที่เกี่ยวข้องกับการส่งข้อมูล

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R-1	R/W-0
CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D
bit7							bit0

ตารางที่ 2.18 TXSTA register เป็น register ที่เกี่ยวข้องกับการส่งข้อมูล

Bit7 CSRC

= เราไม่สนใจ bit นี้ในกรณีที่ใช้แบบ Asynchronous

Bit6 TX9

= 1 หมายถึง กำหนดให้เป็นการส่งแบบ 9 bit

= 0 หมายถึง กำหนดให้เป็นการส่งแบบ 8 bit

Bit5 TXEN

= 1 หมายถึง อนุญาตให้มีการส่งข้อมูล

= 0 หมายถึง ไม่อนุญาตให้มีการส่งข้อมูล

Bit4 SYNC

= 1 หมายถึง เป็นการส่งแบบ Synchronous mode

= 0 หมายถึง เป็นการส่งแบบ Asynchronous mode

Bit3 ไม่มีการใช้งาน

Bit2 BRGH

ใน Asynchronous mode หมายถึง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

= 1 หมายถึง เป็นแบบ High speed

= 0หมายถึง เป็นแบบ Low speed

Bit1 TRMT หมายถึง bit ที่แสดงสถานะของ Transmitt Shift register

= 1 หมายถึง เป็น TSR ว่าง

= 0หมายถึง เป็น TSR เต็ม

Bit0 TX9D เป็นข้อมูล bit ที่ 9 ถ้าจะทำการส่งแบบ 9 bit

PIE1 บิตที่เกี่ยวข้องคือ RCIE,TXIE

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PSPIE(1)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE
bit7							bit0

ตารางที่ 2.19 PIE1 บิตที่เกี่ยวข้องคือ RCIE,TXIE

Bit5 RCIE

= 1 หมายถึงกำหนดให้มีการ interrupt เกิดขึ้นได้เมื่อทำการรับข้อมูลทาง serial เสร็จ 1 byte

= 0 หมายถึงไม่ให้มีการ interrupt เกิดขึ้นได้เมื่อทำการรับข้อมูลทาง serial เสร็จ 1 byte

Bit4 TXIE

= 1 หมายถึงกำหนดให้มีการ interrupt เกิดขึ้นได้เมื่อทำการส่งข้อมูลทาง serial เสร็จ 1 byte

= 0 หมายถึงไม่ให้มีการ interrupt เกิดขึ้นได้เมื่อทำการส่งข้อมูลทาง serial เสร็จ 1 byte

SPBRG register คือ register ที่กำหนดค่าที่ baud rate

การคำนวณหาค่า Baud rate

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TABLE 10-1 BAUD RATE FORMULA

SYNC	BRGH = 0 (Low Speed)	BRGH = 1 (High Speed)
0	(Asynchronous) Baud Rate = Fosc/(64(X+1))	Baud Rate= Fosc/(16(X+1))
1	(Synchronous) Baud Rate = Fosc/(4(X+1))	NA

X = value in SPBRG (0 to 255)

ตารางที่ 2.20 SPBRG register คือ register ที่กำหนดค่าที่ baud rate

สูตรที่ใช้คำนวณหา ค่า Baud rate ขึ้นอยู่บิต 2 บิต บิตคือ **SYNC** กับ **BRGH** ส่วนว่าเลือกบิตให้สูตรคำนวณหา Baud Rate เป็นอย่างไร ก็ดู ส่วนที่ผมจะใช้คือ SYNC=0,BRGH=0 ดั้งนั้นจะใช้สูตร

Baud Rate = Fosc/(64(X+1))

หรือจะดูเทียบตารางก็ได้ เพราะใน Datasheet จะมีตารางมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.0 LCD

2.4.1รายละเอียดเกี่ยวกับโมดูล LCD

ในโมดูล LCD จะมีส่วนประกอบหลักๆ 3 ส่วน ดังนี้

ตัวแสดงผล (Display) ภายในเป็นผลึกเหลวที่สามารถแสดงผลให้เห็นโดยอาศัยแสงจากภายนอก ดังนั้นจึงต้องมีมุมในการมองข้อมูลที่แสดงผลบนจอ LCD

ตัวควบคุม (Controller) เป็นตัวรับข้อมูลจากอุปกรณ์ภายนอกมาควบคุมการทำงานของของโมดูล LCD เช่น ลบจอภาพ แสดงตัวอักษร หรือเลื่อนเคอร์เซอร์ เป็นต้น ตัวควบคุมนี้ใช้ชิปควบคุมโดยเฉพาะ ชิปที่นิยมใช้คือ เบอร์ HD44780 และ HD61830 โดย HD44780 จะใช้ควบคุม LCD แบบอักขระ ส่วน HD61830 ใช้ควบคุม LCD แบบกราฟิก

ตัวขับ (Driver) เป็นตัวรับสัญญาณจากตัวควบคุม มาขับให้ตัวแสดงผลแสดงข้อมูลตามที่กำหนด ชิปที่ใช้ทำหน้าที่ เป็นตัวขับนี้ได้แก่ เบอร์ HD44100H และ MSM5259 เป็นต้น

2.4.2 โครงสร้างภายในของตัวควบคุมโมดูล LCD

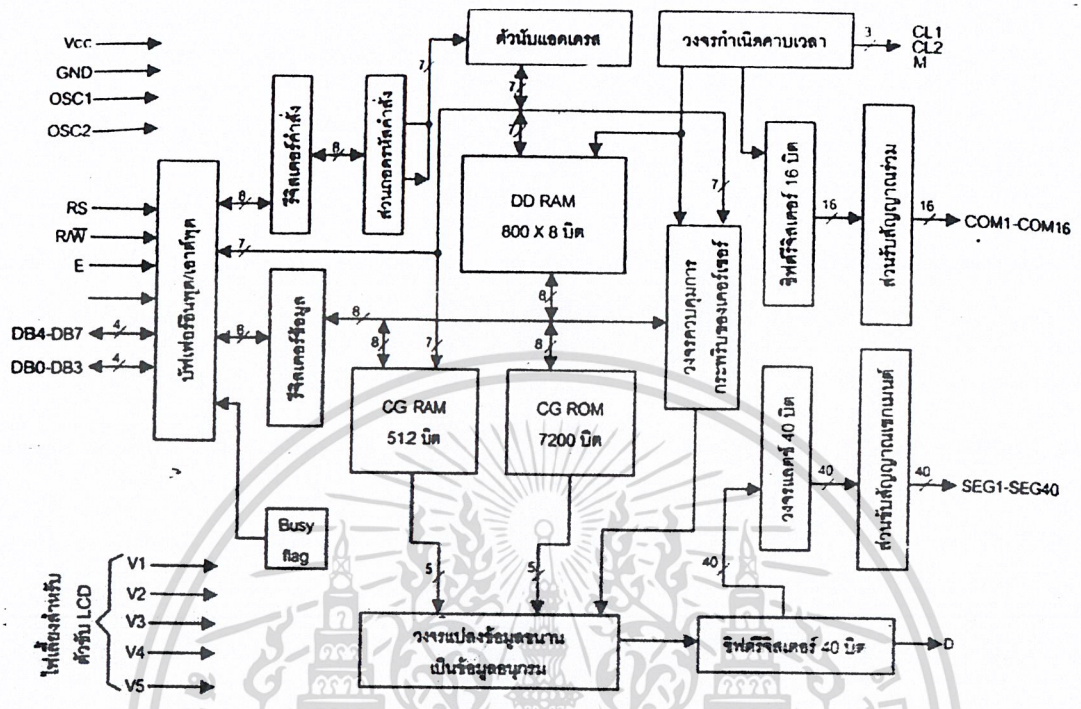
ในการใช้งานโมดูล LCD จำเป็นต้องทำความเข้าใจเกี่ยวกับโครงสร้างและคำสั่งที่ใช้ในการควบคุมให้ดีเสียก่อน ตัวอย่างเช่น โมดูล LCD แบบอักขระ เพราะสามารถเข้าใจได้ง่ายให้ดู รูป 2.13 บล็อกไดอะแกรมภายในของชิปควบคุม LCD เบอร์HD44780 ใช้ในโมดูล LCD แบบอักขระ ประกอบด้วย

บัฟเฟอร์อินพุตเอาท์พุต เป็นส่วนที่ใช้ในการติดต่อรับส่งข้อมูลกับอุปกรณ์ภายนอก เพื่อจะถ่ายทอดข้อมูลเข้าออกภายในตัวควบคุม

รีจิสเตอร์คำสั่ง (Instruction Register : IR) เป็นรีจิสเตอร์ใช้รับข้อมูลคำสั่งจากอุปกรณ์ภายนอกเพื่อนำไปควบคุมการแสดงผล

รีจิสเตอร์ข้อมูล (Data Register : DR) เป็นรีจิสเตอร์ใช้รับข้อมูลจากอุปกรณ์ภายนอกเพื่อถ่ายทอดต่อไปยังหน่วยความจำที่ทำหน้าที่เก็บข้อมูลแสดงผล หรือนำข้อมูลไปสร้างตัวอักษรเพิ่มเติมใน RAM เก็บตัวอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.18 ไดอะแกรมโครงสร้างและการทำงานของตัวควบคุมภายในโมดูล LCD

แรมเก็บข้อมูลแสดงผล (Display Data RAM : DDRAM) เป็นหน่วยความจำแรมทำหน้าที่เก็บข้อมูลที่มาจากรีจิสเตอร์ DR ตัวควบคุมจะนำข้อมูลใน DDRAM นี้ไปเปิดตารางของตัวอักษรที่เก็บไว้ในหน่วยความจำรอมและแรมเก็บตัวอักษร เพื่อนำไปแสดงที่ตัวแสดงผล

รอมเก็บตัวอักษร (Character Generator ROM : CGROM) เป็นหน่วยความจำรอมที่ใช้เก็บข้อมูลตัวอักษรหรือสัญลักษณ์ที่สามารถอ่านออกไปแสดงที่ตัวแสดงผลได้ มีขนาด 7,200 บิต โดยจะถูกอ่านด้วยค่าของข้อมูลใน DDRAM

แรมเก็บตัวอักษร (Character Generator RAM : CGRAM) เป็นหน่วยความจำแรมที่ใช้เก็บตัวอักษรที่มีการสร้างเพิ่มเติมขึ้นใหม่ ในกรณีที่ตัวอักษรใน CGROM ไม่เพียงพอ มีขนาด 512 บิต การเขียนและอ่านค่าไปใช้นั้นทำได้เช่นเดียวกับ CGROM คือ เขียนข้อมูลลงใน DDRAM แล้วตัวควบคุมจะอ่านค่าจาก CGRAM เอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แฟลก BUSY เป็นส่วนที่ทำหน้าที่แจ้งสถานะการทำงานของตัวควบคุมให้อุปกรณ์ภายนอกทราบว่า ตัวควบคุมพร้อมที่จะรับข้อมูล หรือคำสั่งหรือไม่ ดังนั้นก่อนการส่งข้อมูล หรือคำสั่งมายังตัวควบคุม ต้องตรวจสอบสถานะ ของ แฟลก BUSY นี้เสียก่อน

2.4.3 โมดูล LCD ขนาด 16 ตัวอักษร 2 บรรทัด

สำหรับโมดูล LCD ที่ยกมาใช้ในการเรียนรู้ในการทดลอง เป็นขนาด 16 ตัวอักษร 2 บรรทัด มีขาต่อที่ใช้งานทั้งสิ้น 14 ขา มีการจัดขาดังในรูปที่ รายละเอียดการทำงานของแต่ละขามีดังนี้

VSS (ขา 1) : ต่อกราวด์

VDD (ขา 2) : ต่อไฟเลี้ยง + 5 โวลต์

VO (ขา 3) : เป็นขาอินพุตรับแรงดันเพื่อปรับความเข้มของการแสดงผล

RS (ขา4) : เป็นขาอินพุตใช้ในการแยกชนิดของขข้อมูลที่ทำการประมวลผลในขณะนั้นว่าปืนคำสั่งหรือรีจิสเตอร์ IR หรือเป็นข้อมูลสำหรับรีจิสเตอร์ DR โดยที่ขานี้เป็น " 0 " ข้อมูลที่ส่งมาจะเป็นคำสั่ง แต่ถ้าขามีค่าเป็น " 1 "ข้อมูลที่ส่งมาจะเป็นข้อมูลสำหรับการแสดงผล

R/W (ขา 5) : เป็นขาที่ใช้เลือกการอ่านหรือเขียนข้อมูลกับ LCD ถ้าเป็น " 0 " เป็นการกำหนดให้เขียนข้อมูล แต่ถ้าขามีเป็น " 1 " จะเป็นการอ่านข้อมูล

E (ขา 6) : เป็นขาเอ็นเอเบิล LCD ให้ทำงาน

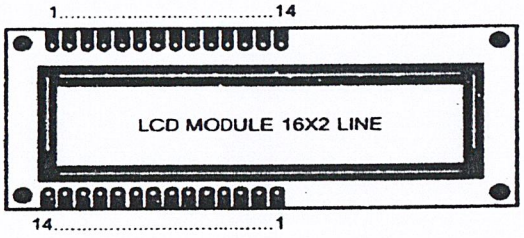
D0-D7 (ขา7-14) : เป็นขาข้อมูลระหว่าง LCD กับอุปกรณ์ภายนอกขนาด 8 บิต

อนึ่งขา RS , R/W และ E จะใช้งานร่วมกันโดยมีความสัมพันธ์ดังในตารางที่ 2.15

RS	R/$\overline{W}$	E	การทำงาน
0	0	↓	เขียนคำสั่ง
0	1	⊓	อ่านสถานะของ LCD
1	0	↓	เขียนข้อมูล
1	1	⊓	อ่านข้อมูล

ตาราง ที่ 2.21 แสดงความสัมพันธ์ของการทำงานระหว่างขา RS , R/W และ E ของโมดูล LCD แบบอักขระ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ขา 1 : GND

ขา 2 : +V

ขา 3: Brightness ปรับความสว่าง

ขา 4 : RS

ขา 5 : R/W

ขา 6 : E

ขา 7-14 : D0-D7

รูปที่ 2.19 การจัดขาของโมดูล LCD ขนาด 16 ตัวอักษร 2 บรรทัด ซึ่งมีด้วยกัน 2 แบบ

2.4.4 คำสั่งควบคุมโมดูล LCD

ในการเขียนคำสั่งลงในตัวควบคุมแน่นอนว่าจะต้องกำหนดให้ขา RS และ R/W เป็น "0" แล้วเขียนคำสั่งตามไป

คำสั่งเคลียร์จอแสดงผล มีข้อมูลคำสั่งเป็น $1 เป็นคำสั่งเขียนข้อมูลช่องว่างหรือ space เข้าไปในหน่วยความจำข้อมูลสำหรับแสดงผลหรือ DDRAM ภายในโมดูล LCD เมื่อเคลียร์จอแสดงผลแล้ว จะกำหนดให้เคอร์เซอร์กลับไปอยู่ที่ตำแหน่งซ้ายมีสุดของจอแสดงผล

คำสั่ง Return Home มีค่าของข้อมูลเท่ากับ $2 หรือ $3 ก็ได้(แนะนำให้ใช้ $2) เป็นการกำหนดให้เคอร์เซอร์ไปแสดงยังตำแหน่งซ้ายสุด ของจอแสดงผล โดยข้อมูลที่แสดงอยู่บนจอแสดงผลจะไม่มีการเปลี่ยนแปลง

คำสั่งเลือกโหมดการป้อนข้อมูล (Entry Mode Set) มีรูปแบบคำสั่งดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
0	0	0	0	0	1	I/D	S

ตาราง ที่ 2.22 คำสั่งเลือกโหมดการป้อนข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บิต I/D ใช้ในการกำหนดว่า เมื่อเขียนหรืออ่านข้อมูลแล้ว แอดเดรสของหน่วยความจำแสดงผล (DDRAM) เพิ่มหรือลดลง

"0" แอดเดรส ลดลง 1 แอดเดรส

"1" แอดเดรส เพิ่มขึ้น 1 แอดเดรส

บิต S ใช้กำหนดลักษณะการแสดงผล

"0" เมื่อเกิดตัวอักษรใหม่ เคอร์เซอร์เลื่อนไปทางด้านขวามือ

"1" เมื่อเกิดตัวอักษรใหม่ เคอร์เซอร์อยู่ที่เดิม แต่ตัวอักษรเลื่อนไปทางซ้าย

คำสั่งควบคุมการแสดงผล มีรูปคำสั่งดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
0	0	0	0	1	D	C	B

ตาราง ที่ 2.23 คำสั่งควบคุมการแสดงผล

บิต D ใช้ควบคุมการเปิดปิดจอแสดงผล ถ้าเป็น "0" เป็นการปิดจอแสดงผล ถ้าเป็น "1" เป็นการเปิดจอแสดงผล

บิต C ใช้ควบคุมตัวเคอร์เซอร์บนจอแสดงผล ถ้าเป็น "0" เป็นการปิดตัวเคอร์เซอร์หรือไม่แสดงเคอร์เซอร์ ถ้าเป็น "1" เป็นการแสดงตัวเคอร์เซอร์

บิต B ใช้ควบคุมการกระพริบของเคอร์เซอร์ ถ้าต้องการให้เคอร์เซอร์กระพริบ ต้องกำหนดให้เป็น "1"

คำสั่งควบคุมการเลื่อนเคอร์เซอร์และตัวอักษร มีรูปดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
0	0	0	1	S/C	R/L	*	*

ตาราง ที่ 2.24 คำสั่งควบคุมการเลื่อนเคอร์เซอร์และตัวอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเลื่อนเคอร์เซอร์และตัวอักษรขึ้นอยู่กับการกำหนดบิต S/C และ R/L ซึ่งสามารถสรุปได้ดังนี้

S/C	R/L	ลักษณะการเลื่อน	ข้อมูลคำสั่ง
0	0	เลื่อนเคอร์เซอร์ไปทางซ้าย	$10 - $13
0	1	เลื่อนเคอร์เซอร์ไปทางขวา	$14 - $17
1	0	เลื่อนตัวอักษรใหม่ไปทางซ้าย	$18 - $1B
1	1	เลื่อนตัวอักษรใหม่ไปทางขวา	$1C - $1F

ตาราง ที่ 2.25 การเลื่อนเคอร์เซอร์และตัวอักษรขึ้นอยู่กับการกำหนดบิต S/C และ R/L

คำสั่งกำหนดฟังก์ชั่นการทำงาน เป็นคำสั่งที่มีความสำคัญมากที่สุดอีกคำสั่งหนึ่ง มีรูปแบบของข้อมูลคำสั่งดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
0	0	1	DL	N	F	*	*

ตาราง ที่ 2.26 คำสั่งกำหนดฟังก์ชั่นการทำงาน

บิต DL ใช้กำหนดจำนวนบิตในการติดต่อกับโมดูล LCD

"0" กำหนดให้ทำงานในโหมด 4 บิต

"1" กำหนดให้ทำงานในโหมด 8 บิต

บิต N ใช้กำหนดจำนวนบรรทัดที่ต้องการให้แสดงผล

"0" แสดงผล 1 บรรทัด

"1" แสดงผล 2 บรรทัด หรือมากกว่า

แต่สำหรับโมดูล LCD ขนาด 16 ตัวอักษร 1 บรรทัด ที่มีจำหน่ายในประเทศไทยต้องกำหนดให้บิตนี้เป็น "1" เนื่องจากหน่วยความจำสำหรับเก็บข้อมูลเพื่อการแสดงผลไม่ได้อยู่ต่อเนื่องกัน กล่าวคือ แบ่งเป็น 2 ช่วงคือ ที่แอดเดรส $40-$47 โดยที่แอดเดรส $40 จะเป็นแอดเดรส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เริ่มต้นของหน่วยความจำสำหรับการแสดงผลในบรรทัดที่ 2 ของโมดูล LCD ที่มีมากกว่า 1 บรรทัด จึงทำให้เมื่อต้องการติดต่อกับโมดูล LCD 1 บรรทัดจึงจำเป็นต้องกำหนดให้เป็น "1"

บิต F ใช้เลือกความละเอียดของตัวอักษรในการแสดงผล

"0" แสดงผลแบบ 5 x 7 จุด

"1" แสดงผลแบบ 5 x 10 จุด

2.4.5 การเขียนคำสั่งและข้อมูลให้แก่โมดูล LCD

ในการเขียนข้อมูลเพื่อควบคุมให้โมดูล LCD แสดงผลตามที่ผู้ใช้งานต้องการ ต้องส่งคำสั่ง (Instruction) แล้วกำหนดโหมดการทำงานให้แก่โมดูล LCD ก่อน จากนั้นจึงค่อยส่งข้อมูลที่ต้องการแสดงผล เนื่องจากบัสข้อมูลของโมดูล LCD มี 8 เส้น คือ D0-D7 และใช้เป็นทางผ่านของทั้งคำสั่งและข้อมูล ดังนั้นในการส่งคำสั่งและข้อมูลจึงต้องอาศัยการกำหนดสัญญาณลอจิกที่ขา RS ถ้าหากที่ขา RS ได้ลอจิก "0" หมายความว่า ข้อมูลที่ป้อนให้แก่โมดูล LCD ขณะนั้นเป็นคำสั่ง ในทางตรงข้าม หากขา RS ได้รับลอจิก "1" ข้อมูลที่ป้อนให้ขณะนั้นเป็นข้อมูลที่ใช้แสดงผล

เมื่อต้องการเขียนหรืออ่านข้อมูลใน CGRAM และ DDRAM เริ่มต้นต้องกำหนดแอดเดรสที่ต้องการอ่านหรือเขียนก่อน โดยใช้คำสั่งหรือเลือกแอดเดรส จากนั้นกำหนดให้ขา RS เป็น "1" เพื่อแจ้งให้ตัวควบคุมภายในโมดูล LCD ทราบว่าข้อมูลที่ปรากฏต่อไปนี้เป็นข้อมูลปกติไม่ใช่คำสั่ง

ในกรณีที่ต้องการอ่านข้อมูลต้องกำหนดให้ขา R/W เป็น"1" ข้อมูลขนาด 8 บิต(หรือ 4 บิต)
ก็จะปรากฏบนบัสข้อมูล โดยข้อมูลที่อ่านออกมาได้จะเป็นข้อมูลจากแอดเดรสของ CGRAM หรือ DDRAM ตามที่ต้องการ

ในกรณีที่ต้องการเขียนข้อมูล เมื่อกำหนดแอดเดรสและป้อนลอจิก " 1 " ให้ขา RS แล้วแล้วต้องกำหนดให้ขา R/W เป็น "0" ข้อมูลที่อยู่บนบัสข้อมูลจะถูกเขียนลงในรีจิสเตอร์ DR จากนั้นจึงถ่ายทอดลงใน DDRAM ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.6 จังหวะการทำงานของโมดูล LCD

ในการติดต่อกับโมดูล LCD จะต้องมีการหน่วงเวลาหลังจากที่ทำการส่งระหัสคำสั่งหรือข้อมูลเนื่องจากต้องรอให้ คอลโทรลเลอร์ภายใน โมดูล LCD แปลความหมายของรหัสคำสั่งและทำงานตามคำสั่งให้เรียบร้อยก่อน จากนั้นจะรับข้อมูลหรือดำเนินการต่อไป

ดังนั้นในการใช้งาน โมดูล LCD ต้องมีโปรแกรมหน่วงเวลารอให้โมดูล LCD พร้อมทำงานด้วย โดยเมื่อเริ่มจ่ายไฟให้โมดูล LCD ต้องรอประมาณ 5-15 มิลลิวินาที เพื่อให้โมดูล LCD ทำการเตรียมความพร้อมหรืออินิเชียล หลังจากนั้นก็จะกำหนดลอจิกให้แก่ขา RS ของโมดูล LCD แล้วต้องหน่วงเวลาประมาณ 40 – 125 ไมโครวินาที เพื่อคอลโทรลเลอร์ในโมดูล LCD แปลความหมายของลอจิกที่ขา RS ว่า ข้อมูลต่อไปที่จะได้รับนั้นเป็นรหัสคำสั่งหรือเป็นข้อมูลที่ต้องการแสดงผล

2.5.0 SRAM 62256

แรม ชนิดนี้ หน่วยความจำแต่ละเซลล์จะใช้ ลักษณะของวงจร ฟลิบฟลอบ เป็นพื้นฐาน ดังนั้นเมื่อไม่มีการเขียนข้อมูลเข้าไปใหม่ ข้อมูลนั้นจะคงที่ตลอดไปเท่าที่มีไฟเลี้ยงอยู่

2.5.1 ศัพท์ที่ควรทราบเกี่ยวกับหน่วยความจำ

Access time ระยะเวลาจากที่กำหนดสัญญาณระบุตำแหน่งให้กับหน่วยความจำ ไปจนถึงเวลาที่ข้อมูลปรากฏออกมาที่เอาต์พุตของไอซีหน่วยความจำ ระยะเวลานี้รวมถึงเวลาที่เสียไปตั้งแต่การส่งสัญญาณเลือกตัวไอซี (Chip Select) การถอดรหัสของตำแหน่ง และการทำงานของวงจร Output Enable

Address Hole Time คือเวลาที่ต้องการนับตั้งแต่จุดสิ้นสุดของสัญญาณ WRITE จนถึงเวลาที่เอาสัญญาณในการกำหนดตำแหน่งออกไปได้โดยข้อมูลไม่เสียหาย

Address Setup Time คือระยะเวลาที่ต้องการเพื่อให้สัญญาณในการกำหนดตำแหน่งให้อยู่ในสภาวะคงที่ก่อนที่จะส่งสัญญาณการเขียนให้แก่หน่วยความจำ เพื่อทำการเขียนข้อมูล

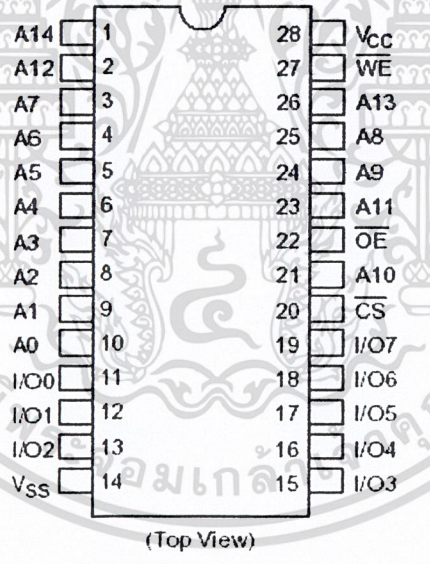
Ship Enable (CE) คือสัญญาณที่ใช้กำหนดให้ไอซีหน่วยความจำนี้ทำงานหรือไม่ ซึ่งเมื่อสัญญาณนี้แอกตีฟ (ซึ่งปกติ แอกตีฟที่ระดับ 0) ไอซีหน่วยความจำนี้จะอยู่ในสภาวะที่ทำงานได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Output Enabel (OE) คือสัญญาณที่ป้อนให้แก่ตัวไอซี ซึ่งเมื่ออยู่ในสวาวะแอกตีฟ จะทำให้เอาต์พุตของไอซีหน่วยความจำนี้ต่อกับบัสข้อมูลของระบบ แต่เมื่อสัญญาณนี้ไม่แอกตีฟเอาต์พุตของไอซีจะอยู่ในสภาวะอิมพีแดนซ์สูง

2.5.2 ลักษณะของหน่วยความจำ

หน่วยความจำแรมจะมีลักษณะภายนอกคล้ายๆ กับหน่วยความจำแบบรอม แต่แตกต่างกันที่แรมจะมีสายสัญญาณเพิ่มมาอีก 1 เส้น คือสัญญาณควบคุมการอ่านเขียน ดังแสดงตำแหน่งของขาในรูปที่ 2.17 และมีการกำหนดสภาวะให้ขาควบคุมเพื่อทำงานต่างๆในตารางที่ 2.21



รูปที่ 2.20 แสดงตำแหน่งขาของ SRAM 62256

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Function Table

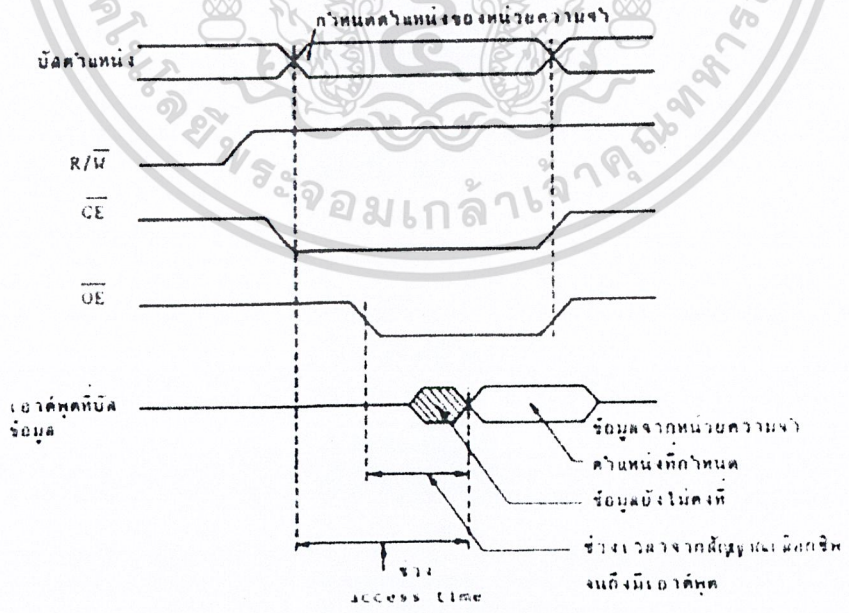
WE	CS	OE	Mode	V_{CC} Current	I/O Pin	Ref. Cycle
X	H	X	Not selected	I_{SB}, I_{SB1}	High-Z	—
H	L	H	Output disable	I_{CC}	High-Z	—
H	L	L	Read	I_{CC}	Dout	Read cycle (1)-(3)
L	L	H	Write	I_{CC}	Din	Write cycle (1)
L	L	L	Write	I_{CC}	Din	Write cycle (2)

Note: X: H or L

ตารางที่ 2.27 แสดงFUNCTION การทำงานของ SRAM62256

2.5.2.1 ลำดับขั้นในการอ่านข้อมูลจากแรม

ในการอ่านข้อมูลจากแรมนั้น ต้องใช้สัญญาณควบคุมการอ่านและเขียนมาร่วมในการควบคุมการทำงานด้วย โดยทำการกำหนดให้สัญญาณควบคุมการอ่านและเขียนนี้แอกตีฟตามลักษณะของการอ่านหรือเขียนข้อมูลดังต้องการ เช่นถ้าให้สัญญาณนี้อยู่ในสภาวะแอกตีฟอ่าน (สมสุติ = 1) การทำงานก็จะเป็นการอ่านข้อมูลจากหน่วยความจำตามตำแหน่งที่ระบุ ซึ่งลำดับขั้นของการอ่านข้อมูลนี้สามารถเขียนเป็นไดอะแกรมเวลาได้ดังรูปที่ 2.18 และมีลำดับขั้นเป็นดังนี้



รูปที่ 2.21 ไดอะแกรมเวลาของการอ่านข้อมูลจากแรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ซีพียูส่งข้อมูลของตำแหน่งที่ต้องการอ่านออกไปบนบัสตำแหน่ง (กำหนดให้ในขณะนี้สัญญาณการอ่านแอกตีฟ)

2.ซีพียูส่งสัญญาณเลือกใช้ชิพและส่งไปยังหน่วยความจำ

3.ซีพียูรอเวลาสักระยะหนึ่ง นับตั้งแต่ส่งข้อมูลเพื่อกำหนดตำแหน่งเข้าไป ซึ่งเวลานี้ก็คือ Access Time ซึ่งอาจมีค่าประมาณ 35 – 200 nS แล้วแต่ชนิดและขนาดของแรม หลังจากช่วง Access Time ผ่านไปแล้ว ข้อมูลจากหน่วยความจำก็จะปรากฏบนบัสข้อมูล ซึ่งซีพียูจะอ่านข้อมูลในช่วงนี้เข้าไป

4.สัญญาณเลือกชิพเลิกแอกตีฟ บัสข้อมูลกลับสู่สภาวะอิมพีแดนซ์สูง

2.5.2.2. ลำดับขั้นในการเขียนข้อมูลลงหน่วยความจำแบบแรม

ในการเขียนข้อมูลลงหน่วยความจำแบบแรม จะมีลำดับขั้นดังนี้

1.ซีพียูส่งข้อมูลของตำแหน่งที่ต้องการให้เขียนไปบนบัสตำแหน่ง และอาจพร้อมทั้งสัญญาณการเลือกใช้ชิพ

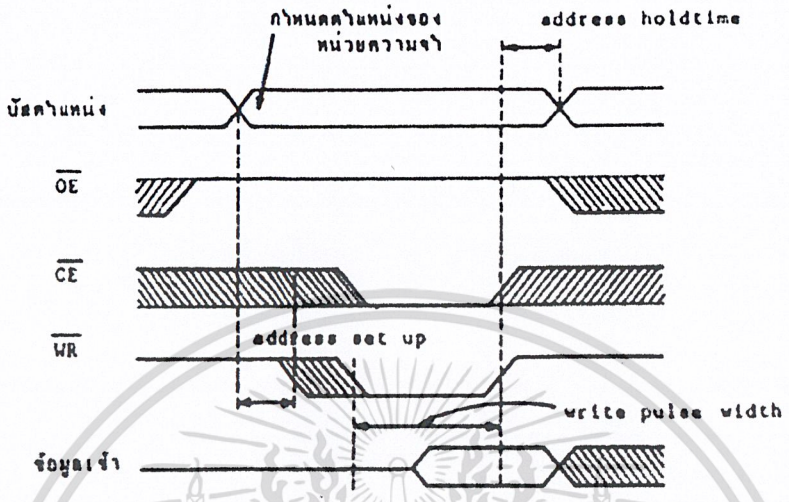
2.ข้อมูลที่จะเป็นข้อมูลอินพุต ถูกส่งเข้ามายังอินพุตของแรม

3.หน่วยความจำต้องการเวลาสักระยะหนึ่ง เพื่อให้ข้อมูลในการกำหนดตำแหน่งอยู่ในสภาวะคงที่ซึ่งช่วงเวลานี้คือ Address setup time ซึ่งจะมีค่าประมาณ 20 nS จากนั้นระบบจึงส่งสัญญาณควบคุมการเขียนให้อยู่ในสภาวะแอกตีฟ

4.หลังจากที่สัญญาณควบคุมการเขียนแอกตีฟชักระยะหนึ่ง หน่วยความจำจะทำการนำข้อมูลจากอินพุตเข้าเขียนเข้าไปยังหน่วยความจำ และเมื่อสัญญาณการเขียนเลิกแอกตีฟ ก็เป็นการสิ้นสุดกระบวนการเขียน

ลำดับขั้นของการเขียนข้อมูลลงแรม สามารถแสดงได้โดยไดอะแกรมเวลาดังรูปที่ 2.22

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.22 ไดอะแกรมเวลาของการเขียนหน่วยความจำ

จากรูปเห็นได้ว่าการกำหนดให้สัญญาณ OE อยู่ในสภาวะไม่แอกตีฟ เพื่อไม่ให้มีสัญญาณเอาต์พุตออกมาที่บัสข้อมูล แต่ถ้าสัญญาณ OE อยู่ในสภาวะแอกตีฟ และขณะนั้นสัญญาณ CE แอกตีฟด้วยก็จะทำให้เอาต์พุตปรากฏออกมาที่บัสขาออก แต่เมื่อมีสัญญาณการเขียนเข้ามา บัสทางออกนี้ก็จะมีสภาวะเป็นอิมพีแดนซ์สูง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

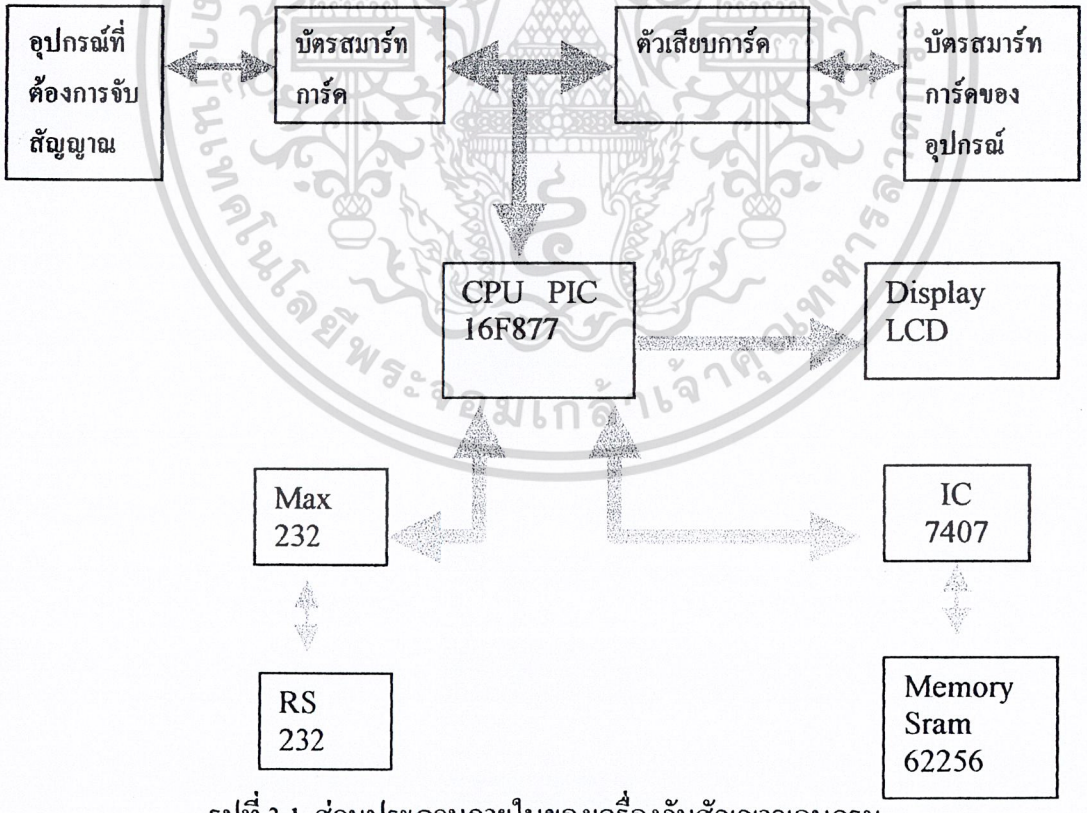
บทที่ 3
การออกแบบและการทำงาน

ในการจัดทำเครื่องจับสัญญาณแบบอนุกรม ให้โครงงานสำเร็จลุล่วงตามขอบเขตที่กำหนดไว้ มีขั้นตอนการดำเนินงานดังต่อไปนี้

1. ออกแบบเครื่องจับสัญญาณอนุกรม
2. สร้างเครื่องจับสัญญาณอนุกรม
3. การเขียนโปรแกรมควบคุมการทำงานของจับสัญญาณอนุกรม

3.1.0 ออกแบบเครื่องจับสัญญาณอนุกรม

แนวความคิดในการออกแบบระบบเครื่องจับสัญญาณอนุกรมนี้ ออกแบบเพื่อให้มีความสะดวกแก่การใช้งาน และสามารถใช้ติดต่อกับคอมพิวเตอร์ ซึ่งรายละเอียดภายในของเครื่องจับสัญญาณ จะประกอบด้วยส่วนต่างๆดังนี้



รูปที่ 3.1 ส่วนประกอบภายในของเครื่องจับสัญญาณอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการออกแบบวงจรเครื่องจับสัญญาณ จะใช้ไมโครคอลโทรลเลอร์เบอร์ 16F877 เป็น CPU โดยใช้โมดูลการรับสัญญาณแบบ Asyncronous มาเป็นตัวรับสัญญาณการติดต่อของสมาร์ตการ์ด ใช้ ไอซี MAX 232 เป็นตัวแปลงระดับแรงดันเพื่อติดต่อกับคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม โดยใช้ขั้วต่อแบบ DB-9 สำหรับโปรแกรมในคอมพิวเตอร์ที่ใช้ติดต่อจะ Down Load มา ส่วน Display จะใช้ตัว LCD เป็นตัวแสดงผล ใช้ ไอซี เบอร์ 62256 เป็นตัวเก็บข้อมูล โดยได้นำ IC เบอร์ 74 LS373 มาเป็นตัว Latch เพื่อช่วยให้ใช้พอร์ตของ CPU ในการติดต่อกับ Sram น้อยลง และใช้ IC 7407 ซึ่งเป็นบัฟเฟอร์มาช่วยในการแสดงการรับพัลส์ ข้อมูลเข้าสูง CPU โดย LEDจะสว่างเมื่อมีพัลส์ค่า 0 เข้ามา

จากแนวความคิดในการออกแบบทำให้ได้วงจรดังรูปที่ 3.2

ขา RA 0 เป็นสวิตช์เลือกให้เครื่องทำการบันทึกข้อมูลลง Memory เมื่อมีค่าเป็น “1”
เป็นสวิตช์เลือกให้เครื่องอ่านข้อมูลจาก Memory เมื่อมีค่าเป็น “0”

ขา RA 4 เป็นสวิตช์เลือกให้แสดงผลออกทางพอร์ตอนุกรม เมื่อมีค่าเป็น “1”
เป็นสวิตช์เลือกให้แสดงผลออกทาง LCD เมื่อมีค่าเป็น “0”

ขา RA2 ต่อกับขา Enable ของ LCD

ขา RA3 ต่อกับขา RS ของ LCD

ขา MCLR รับสัญญาณ RST เพื่อเริ่มต้นการทำงาน

ขา RC6 ส่งข้อมูลไปให้ RS-232 ผ่านAX-232

ขา RC7 รับข้อมูลการติดต่อของสมาร์ตการ์ดกับเครื่อง

ขา RB0-RB7 รับส่งข้อมูลที่ทำการบันทึกหรืออ่านจาก Sram 62256

ขา RD0-RD7 รับส่ง Address ที่ต้องการเขียนหรืออ่านให้ Sram 62256

ขา RC0-RC3 เป็นขาส่งข้อมูลให้กับ LCD

ขา RE0 ส่งคำสั่งให้ขา LE ของ 74LS 373

ขา RE1 ส่งคำสั่งเลือกให้ขา CS ของ Sram 62256

ขา RE2 ส่งคำสั่งเลือกให้ขา R/W ของ Sram 62256

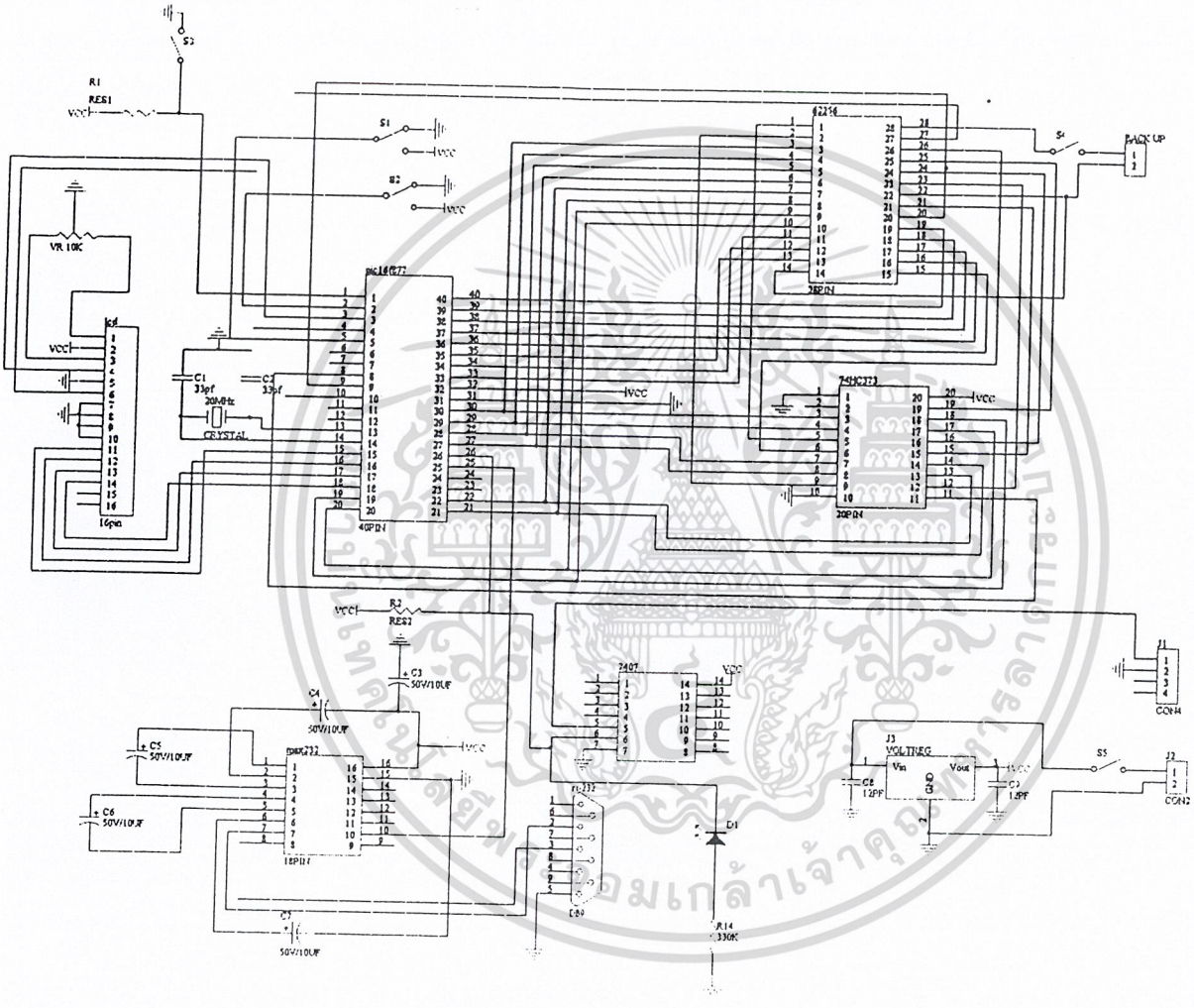
ส่วนขา OE ของ IC 62256 และขา OE ของ 74LS373 ต่อลงกราว์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.0 สร้างเครื่องจับสัญญาณอนุกรม

การสร้างเครื่องจับสัญญาณอนุกรม เพื่อให้ง่ายแก่การอธิบายและง่ายในการทำความเข้าใจ จะขอแยกส่วนประกอบออกเป็นสวนๆ เพื่อทำการอธิบาย ส่วนวงจรของเครื่องสามารถดูได้จาก รูปวงจรเครื่องจับสัญญาณอนุกรม

- วงจรแสดงเป็น Shematic



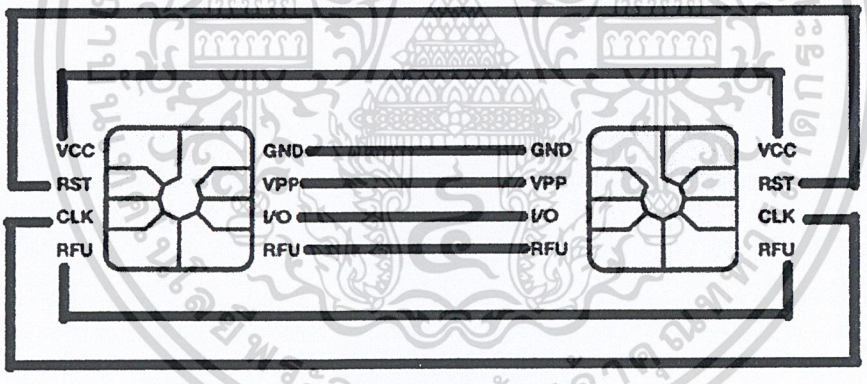
รูปที่ 3.2 วงจรเครื่องจับสัญญาณอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1 รายละเอียดส่วนประกอบต่างๆ

3.2.1.1 ส่วนบัตรสมาร์ตการ์ดจำลอง และ สล็อตสมาร์ตการ์ด

การจะสามารถจับสัญญาณการติดต่อของบัตรสมาร์ตการ์ดกับตัวเครื่องใช้งานได้นั้น จำเป็นที่จะต้องมีจุดสำหรับจั้มสายสัญญาณออกมาเพื่อทำการรับสัญญาณที่จำเป็น จึงไม่สามารถใช้บัตรจริงได้โดยตรง ต้องสร้างบัตรสมาร์ตการ์ดจำลองขึ้นมา โดยบัตรใบนี้จะมีเพียงหน้าคอนแท็คที่ใช้เสียบกับสล็อตในตัวเครื่องที่ต้องการจับสัญญาณเท่านั้น ซึ่งหน้าคอนแท็คนี้จะต้องมีรายละเอียดเหมือนกับบัตรจริงทุกประการตามมาตรฐาน ISO – 7816 ส่วน Memory ที่อยู่ในตัวบัตรนั้นเราจะนำมาจากบัตรที่ใช้งานจริงกับอุปกรณ์นั้นๆ โดยเสียบบัตรจริงเข้ากับสล็อตสมาร์ตการ์ดของเรา แล้วทำการต่อขั้วของสล็อตเข้ากับหน้าคอนแท็คโดยตำแหน่งเชื่อมของขั้วสล็อตกับ หน้าคอนแท็คของบัตรจำลองนั้นเมื่อทำการเสียบบัตรที่ใช้งานจริงเข้ากับสล็อตของเราแล้วหน้าคอนแท็คบัตรจำลองจะต้องตรงกับ ตำแหน่งหน้าคอนแท็คของบัตรจริง ดังรูป



รูป 3.3 จำลองการต่อของคอนแท็คจำลองกับคอนแท็คของบัตรจริง

รายละเอียดหน้าที่การทำงานของขาต่างๆ ของสมาร์ทการ์ด

- **I/O** : หรือสัญญาณอินพุต / เอาต์พุตข้อมูลแบบอนุกรม (Serial data)
- **VPP** : สัญญาณโปรแกรมข้อมูลสำหรับหน่วยความจำ Non-volatile ของการ์ด
- **CLK** : สัญญาณนาฬิกาอ้างอิงการเข้าถึงข้อมูล
- **RST** : สัญญาณรีเซตการทำงานการ์ด
- **VCC** และ **GND** : ไฟเลี้ยงและกราวด์ที่ป้อนให้การ์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการจั๊มสายออกมาเพื่อทำการจับสัญญาณการติดต่อจากสายเชื่อมระหว่าง ตัวบัตรจำลองกับสล็อตสมาร์ตการ์ดนั้น จะใช้เพียง 2 สาย คือ สาย DATA และสาย GND เท่านั้น โดยนำสาย DATA ไปต่อเข้าขา RX ซึ่งใช้สำหรับรับสัญญาณแบบ Syncronous ของตัวไมโครคอลโทรเลอร์ เนื่องจากการรับส่งสัญญาณของบัตรสมาร์ตการ์ดเป็นแบบ Syncronous ส่วนสาย GND ต่อกับ GND ของวงจรเครื่องจับสัญญาณอนุกรม

3.2.1.2 ส่วนจอแสดงผล LCD

การแสดงผลด้วย LCD นี้ใช้เพื่อแสดงข้อมูลอย่างคร่าวๆ ของข้อมูลที่ได้ทำการจัดเก็บไว้ใน Memory และแสดงจำนวนการใช้แอดเดรสของ Memory

LCD ที่ใช้งานนี้จะเป็นแบบ 1 บรรทัด 16 ตัวอักษร ซึ่งจะมีส่วนประกอบหลักๆ ส่วน ดังนี้

ตัวแสดงผล (Display) ภายในเป็นผลึกเหลวที่สามารถแสดงผลให้เห็นโดยอาศัยแสงจากภายนอก ดังนั้นจึงต้องมีมุมในการมองข้อมูลที่แสดงผลบนจอ LCD

ตัวควบคุม (Controller) เป็นตัวรับข้อมูลจากอุปกรณ์ภายนอกมาควบคุมการทำงานของของโมดูล LCD เช่น ลบจอภาพ แสดงตัวอักษร หรือเลื่อนเคอร์เซอร์ เป็นต้น ตัวควบคุมนี้ใช้ชิปควบคุมโดยเฉพาะ ชิปที่ใช้คือ เบอร์ HD44780 โดย HD44780 จะใช้ควบคุม LCD แบบอักขระ

ตัวขับ (Driver) เป็นตัวรับสัญญาณจากตัวควบคุม มาขับให้ตัวแสดงผลแสดงข้อมูลตามที่กำหนด ชิปที่ใช้ทำหน้าที่ เป็นตัวขับนี้ได้แก่ เบอร์ HD44100H และ MSM5259 เป็นต้น

ในการใช้งานโมดูล LCD จำเป็นต้องทำความเข้าใจเกี่ยวกับโครงสร้างและคำสั่งที่ใช้ในการควบคุมให้ดีเสียก่อน LCD เบอร์HD44780 ใช้ในโมดูล LCD แบบอักขระ ประกอบด้วย

บัฟเฟอร์อินพุตเอาท์พุต เป็นส่วนที่ใช้ในการติดต่อรับส่งข้อมูลกับอุปกรณ์ภายนอก เพื่อจะถ่ายทอดข้อมูลเข้าออกภายในตัวควบคุม

รีจิสเตอร์คำสั่ง (Instruction Register : IR) เป็นรีจิสเตอร์ใช้รับข้อมูลคำสั่งจากอุปกรณ์ภายนอกเพื่อนำไปควบคุมการแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รีจิสเตอร์ข้อมูล (Data Register : DR) เป็นรีจิสเตอร์ใช้รับข้อมูลจากอุปกรณ์ภายนอกเพื่อถ่ายทอดต่อไปยังหน่วยความจำที่ทำหน้าที่เก็บข้อมูลแสดงผล หรือนำข้อมูลไปสร้างตัวอักษรเพิ่มเติมใน RAM เก็บตัวอักษร

แรมเก็บข้อมูลแสดงผล (Display Data RAM : DDRAM) เป็นหน่วยความจำแรมทำหน้าที่เก็บข้อมูลที่มาจากรีจิสเตอร์ DR ตัวควบคุมจะนำข้อมูลใน DDRAM นี้ไปเปิดตารางของตัวอักษรที่เก็บไว้ในหน่วยความจำรอมและแรมเก็บตัวอักษร เพื่อนำไปแสดงที่ตัวแสดงผล

รอมเก็บตัวอักษร (Character Generator ROM : CGROM) เป็นหน่วยความจำรอมที่ใช้เก็บข้อมูลตัวอักษรหรือสัญลักษณ์ที่สามารถอ่านออกไปแสดงที่ตัวแสดงผลได้ มีขนาด 7,200 บิต โดยจะถูกอ่านด้วยค่าของข้อมูลใน DDRAM

แรมเก็บตัวอักษร (Character Generator RAM : CGRAM) เป็นหน่วยความจำแรมที่ใช้เก็บตัวอักษรที่มีการสร้างเพิ่มเติมขึ้นใหม่ ในกรณีที่ตัวอักษรใน CGROM ไม่เพียงพอ มีขนาด 512 บิต การเขียนและอ่านค่าไปใช้นั้นทำได้เช่นเดียวกับ CGROM คือ เขียนข้อมูลลงใน DDRAM แล้วตัวควบคุมจะอ่านค่าจาก CGRAM เอง

แฟลก BUSY เป็นส่วนที่ทำหน้าที่แจ้งสถานะการทำงานของตัวควบคุมให้อุปกรณ์ภายนอกทราบว่า ตัวควบคุมพร้อมที่จะรับข้อมูล หรือคำสั่งหรือไม่ ดังนั้นก่อนการส่งข้อมูล หรือคำสั่งมายังตัวควบคุม ต้องตรวจสอบสถานะ ของ แฟลก BUSY นี้เสียก่อน

รายละเอียดการทำงานของแต่ละขาทั้ง 14 ขามีดังนี้

VSS (ขา 1) : ต่อกราวด์

VDD (ขา 2) : ต่อไฟเลี้ยง + 5 โวลต์

VO (ขา 3) : เป็นขาอินพุตรับแรงดันเพื่อปรับความเข้มของการแสดงผล

RS (ขา4) : เป็นขาอินพุตใช้ในการแยกชนิดของขข้อมูลที่ทำการประมวลผลในขณะนั้นว่าปืนคำสั่งหรือรรีจิสเตอร์ IR หรือเป็นข้อมูลสำหรับรีจิสเตอร์ DR โดยที่ขานี้เป็น " 0 " ข้อมูลที่ส่งมาจะเป็นคำสั่ง แต่ถ้าขามีค่าเป็น " 1 "ข้อมูลที่ส่งมาจะเป็นข้อมูลสำหรับการแสดงผล

R/W (ขา 5) : เป็นขาที่ใช้เลือกการอ่านหรือเขียนข้อมูลกับ LCD ถ้าเป็น " 0 " เป็นการกำหนดให้เขียนข้อมูล แต่ถ้าขามีเป็น " 1 " จะเป็นการอ่านข้อมูล

E (ขา 6) : เป็นขาเอ็นเอเบิล LCD ให้ทำงาน

D0-D7 (ขา7-14) : เป็นขาข้อมูลระหว่าง LCD กับอุปกรณ์ภายนอกขนาด 8 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LCD ที่ใช้งานนี้ จะมีรายละเอียดในการต่อใช้งานในวงจรดังนี้

ขาที่ 1 และ ขาที่ 3 จะต่ออยู่กับ ความต้านทานปรับค่าได้ค่า 10 K ส่วนขากลางของ ความต้านทานปรับค่าได้จะต่อลง GND เพื่อใช้ในการปรับค่าความสว่างของตัวอักษร เนื่องจากขา 1 เป็นขา GND และขา 3 เป็นขา ปรับความสว่าง

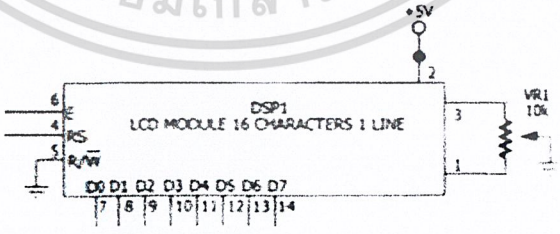
ขาที่ 2 จะทำการต่อกับขั้วไฟบวก

ขาที่ 5 เนื่องจากไม่ได้มีการอ่านค่าจาก LCD แต่จะมีแต่เพียงการเขียนค่าเพื่อแสดงผลออกทาง LCD เท่านั้น ซึ่งการจะเขียนค่าให้ LCD นั้นจะต้องให้ขา 5 ได้รับ GND ดังนั้นในวงจรเราจะต่อให้ขานี้ต่อลง GND ตลอดเวลาเพื่องานแก่การใช้งาน

ขาที่ 4 ซึ่งเป็นขา RS จะต่อเข้าพอร์ตของไมโครคอนโทรลเลอร์ ซึ่งพอร์ตนี้จะต้องเซ็ตค่าให้เป็น Output เพื่อส่งสัญญาณมาควบคุมการทำงานของ LCD โดยในวงจรจะทำการต่อขา RS นี้เข้ากับ พอร์ต RA3 ของตัวไมโครคอนโทรลเลอร์

ขาที่ 6 เป็นขา Enable ของ LCD จะ ต่อเข้าพอร์ตของไมโครคอนโทรลเลอร์ ซึ่งพอร์ตนี้จะต้องเซ็ตค่าให้เป็น Output เพื่อส่งสัญญาณมาควบคุมการทำงานของ LCD โดยในวงจรจะทำการต่อขา Enable นี้เข้ากับ พอร์ต RA2 ของตัวไมโครคอนโทรลเลอร์

ส่วนขาที่ 7-14 นั้นเป็นขาดาต้า แต่ในวงจรเราจะใช้เพียง 4 บิตบนเท่านั้นเพื่อเป็นการประหยัดพอร์ตใช้งานของตัวไมโครคอนโทรลเลอร์ แต่การทำเช่นนี้ ดาต้า ของ LCD ที่ใช้งานจะต้องเป็น 4 บิตบนเท่านั้น ส่วนขา 4 บิตล่างจะทำการต่อลงกราวด์



รูปที่ 3.4 แสดงการต่อใช้งาน LCD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1.3 ส่วนหน่วยความจำ SRAM 62256 และ IC Latch 74LS373

3.2.1.3.1 หน่วยความจำ SRAM 62256

หน่วยความจำแรมนี้ เป็นหน่วยความจำที่มีการทำงานที่ต่างจากรอม คือมันสามารถที่จะทำการอ่านหรือเขียนข้อมูลก็ได้ และในการอ่านหรือเขียนข้อมูลแต่ละคำ หรือแต่ละบิต ณ ที่ตำแหน่งใดๆ ในพื้นที่ของหน่วยความจำนี้จะใช้เวลาเท่าๆกัน แต่ในกรณีของรอมการเขียนข้อมูลเข้าหน่วยความจำ จะใช้เวลามากกว่าการอ่านข้อมูลมาก หน่วยความจำ แบบ แรม จะเป็นชนิด Volatile คือต้องการไฟเลี้ยงตลอดเวลา และเมื่อขาดไฟเลี้ยง ข้อมูลในแรมจะหายไป แรมจะแบ่งออกได้เป็น 2 แบบ คือ

Static RAM หรือเรียกย่อๆว่า SRAM แรม ชนิดนี้ หน่วยความจำของแต่ละเซลล์จะใช้ลักษณะของวงจรฟลิบฟลอบ เป็นพื้นฐาน ดังนั้นเมื่อไม่มีการเขียนข้อมูลเข้าไปใหม่ ข้อมูลนั้นๆจะคงที่ตลอดไปตราบเท่าที่มีไฟเลี้ยงอยู่

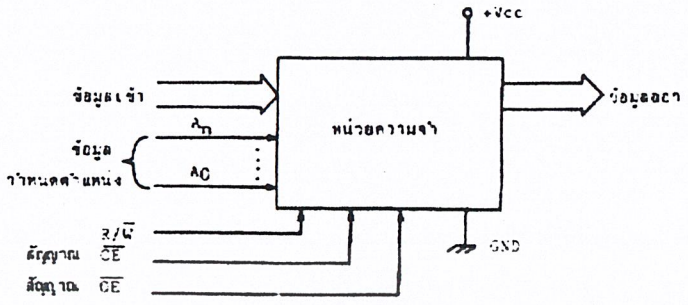
Dynamic RAM หรือ DRAM ลักษณะของ แรมชนิดนี้ใช้การเก็บประจุที่ขาเกตของ MOSFET เพื่อเป็นการเก็บข้อมูล และเมื่อมีการอ่านข้อมูลออกมา ประจุที่เก็บไว้จะถูกคายออกมาและหมดไป ดังนั้น DRAM จึงต้องมีการรีเฟรชทำให้ข้อมูลยังคงอยู่ไม่สูญหายไปไหน

ลักษณะพื้นฐานของการต่อหน่วยความจำกับไมโครโปรเซสเซอร์

ในการเชื่อมต่อกันระหว่างไมโครโปรเซสเซอร์กับหน่วยความจำโดยทั่วไปนั้น จะต้องมีสัญญาณเชื่อมต่อดังต่อไปนี้

1. บัสตำแหน่ง ซีพียูจะใช้ข้อมูลบนบัสนี้สำหรับทำการเลือกตำแหน่งของหน่วยความจำที่จะใช้เป็นต้นทางข้อมูล (เมื่อทำการอ่าน) หรือเป็นปลายทางข้อมูลเ (มื่อทำการเขียน)
2. บัสข้อมูล ใช้สำหรับเป็นทางผ่านของข้อมูลจากซีพียู เพื่อไปยังหน่วยความจำ หรือข้อมูลจากหน่วยความจำ เพื่อมายังซีพียู
3. บัสควบคุม บัสนี้ใช้สำหรับการส่งสัญญาณควบคุมการทำงานในการอ่าน หรือการเขียนข้อมูล เช่นสัญญาณ RD WR เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 การเชื่อมต่อระหว่างหน่วยความจำกับซีพียู

การเลือกใช้หน่วยความจำเพื่อนำมาจัดเก็บข้อมูลที่ได้จากการจับสัญญาณการติดต่อนี้มีความสำคัญมาก คือ ต้องเป็นหน่วยความจำที่มีความจุเพียงพอต่อความต้องการ และ มีความรวดเร็วในการทำกระบวนการจัดเก็บข้อมูลที่รวดเร็วเพียงพอที่จะทำการจัดเก็บข้อมูลได้ทันก่อนที่จะมีข้อมูลชุดใหม่เข้ามา ดังนั้นในส่วนของ Memory นี้จึงใช้ หน่วยความจำที่เป็น SRAM เบอร์ 62256 เพราะมีความเร็วในการทำงานเพียงพอ และมีหน่วยความจุ 32 K ซึ่งเพียงพอในการจัดเก็บข้อมูล

รายละเอียดการทำงานของแต่ละขาทั้ง 28 ขามีดังนี้

VSS (ขา 14) : เป็นขากราวด์

VDD (ขา 28) : เป็นขาไฟบวก 5 โวลท์

WE (ขา 27) : เป็นขาสำหรับเลือกให้เขียนหรืออ่านข้อมูล ถ้าเป็น “1” จะเป็นการสั่งให้อ่านข้อมูลที่เก็บไว้ ถ้าเป็น “0” จะเป็นการสั่งให้เขียนข้อมูล

OE (ขา 22) : เป็นขา OUTPUT ENABLE

CS (ขา 20) : เป็นขา Chip select

E (ขา 6) : เป็นขาเอ็นเอเบิล LCD ให้ทำงาน

A0 - A14 (ขา 1-10,21,23,26) : เป็นขา บัสตำแหน่ง ซีพียูจะใช้ข้อมูลบนบัสนี้สำหรับทำการเลือกตำแหน่งของหน่วยความจำที่จะใช้เป็นต้นทางข้อมูล (เมื่อทำการอ่าน) หรือเป็นปลายทางข้อมูลเ (มื่อทำการเขียน)

I/O1 - I/O7 (ขา 15-19,11,12,13) : เป็นขา บัสข้อมูล ใช้สำหรับเป็นทางผ่านของข้อมูลจากซีพียู เพื่อไปยังหน่วยความจำ หรือข้อมูลจากหน่วยความจำ เพื่อมายังซีพียู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1.3.2 IC Latch 74LS373

เนื่องจาก SRAM ตัวนี้มีการต่อใช้งานเป็นแบบขนาน จึงทำให้ต้องใช้พอร์ต I/O ของ CPU มาใช้งานเป็นจำนวนมาซึ่งจะทำให้ พอร์ตสำหรับใช้งานในส่วนอื่นๆไม่เพียงพอ จึงต้องใช้ IC Latch 74LS373 มาต่อร่วมด้วยเพื่อช่วยลดพอร์ตที่ต้องนำมาต่อใช้งานกับส่วน Memory

โดย IC ตัวนี้จะทำหน้าที่ Latch ข้อมูลเก็บเอาไว้เมื่อ ขา LE (Latch Enable) เป็น "1" และจะทำการส่งออกเมื่อ ขา OE "0"

รายละเอียดการทำงานของแต่ละขา ทั้ง 20 ขา มีดังนี้

VSS (ขา 20) : เป็นขากราวด์

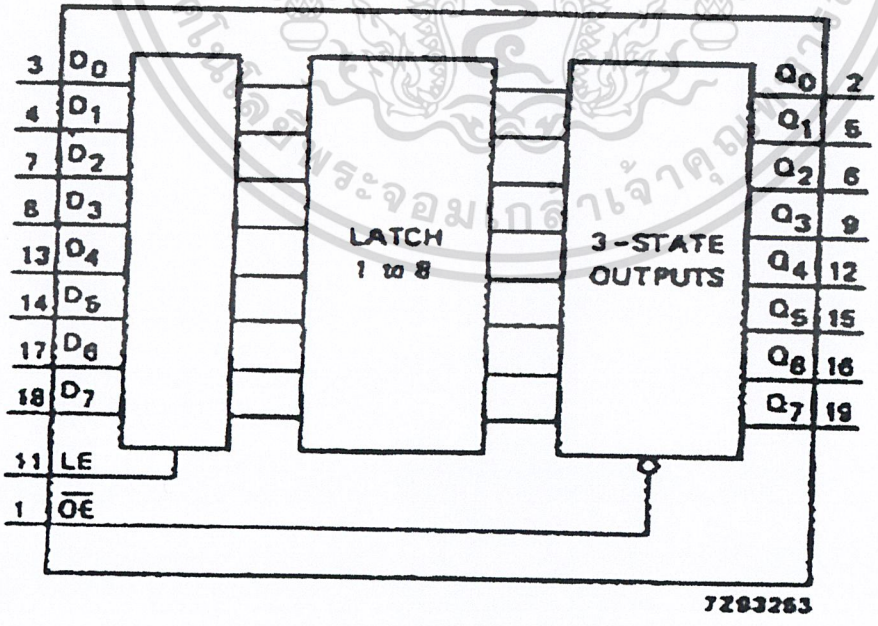
VDD (ขา 10) : เป็นขาไฟบวก 5 โวลท์

LE (ขา 11) : เป็นขาสำหรับเลือกให้ทำการ LATCH ข้อมูล เมื่อเป็น "1"

OE (ขา 1) : เป็นขา OUTPUT ENABLE ทำการส่งออกเมื่อ ขา OE "0"

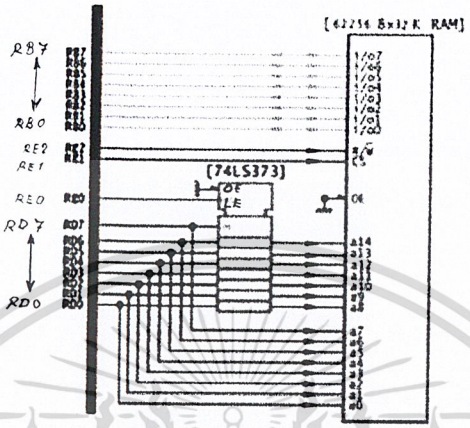
INPUT (ขา 3,4,7,8,13,14,17,18) : เป็นขา อินพุทของข้อมูลที่ต้องการ LATCH

OUTPUT (ขา 2,5,6,9,12,15,16,19) : เป็นขา เอาท์พุท ของข้อมูลที่ทำการ LATCH ไว้



รูปที่ 3.6 Functional diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 แสดงการต่อ ภาค Memory

จากรูปจะเห็นได้ว่าเราสามารถลด พอร์ตของ CPU ที่ต้องนำมาต่อใช้งานได้ถึง 7 พอร์ตด้วยกัน โดยใช้พอร์ต E ทั้ง 3 พอร์ตมาใช้สำหรับควบคุมการทำงานของ IC ทั้ง 2 ตัว โดยขาที่ 1 ของ IC 74 LS 373 ที่เป็นขา OUTPUT Control และ ที่ ขา OE ของ IC RAM ก็เช่นกัน จะต่อลง GND ไว้ตลอดเวลา เนื่องจากดูตารางการควบคุมของ IC ทั้ง 2 ตัวแล้วสามารถต่อลง GND ได้เลยเพราะทุกสถานะที่ต้องการให้ IC ทำงาน ขาทั้งสองนี้จะอยู่ในสภาวะที่เป็น GND ทั้งสิ้น พอร์ต B ของ CPU ต่อเข้าขา I/O ของ SRAM พอร์ต D ของ CPU ต่อเข้าขา อินพุทของ IC Latch ส่วนขา เอาท์พุทของ IC Latch ต่อเข้าขา Address ของ SRAM ซึ่งจะช่วยให้การเขียนโปรแกรมลดความยุ่งยากในการควบคุมการทำงานได้มากละเป็นการประหยัดพอร์ตใช้งานอีกด้วย

3.2.1.4 ส่วนการสื่อสารอนุกรม

การสื่อสารข้อมูลนี้ พอร์ตที่ใช้งานคือ พอร์ต RS-232 แบบ DB 9 ซึ่งเป็นมาตรฐานที่ใช้กันอยู่ในปัจจุบัน จะมีตำแหน่งขาใช้งานดังนี้

ขาที่ 1 Carrier Detect

ขาที่ 2 Receive Data

ขาที่ 3 Transmit Data

ขาที่ 4 Data Terminal Ready

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขาที่ 5 System Ground

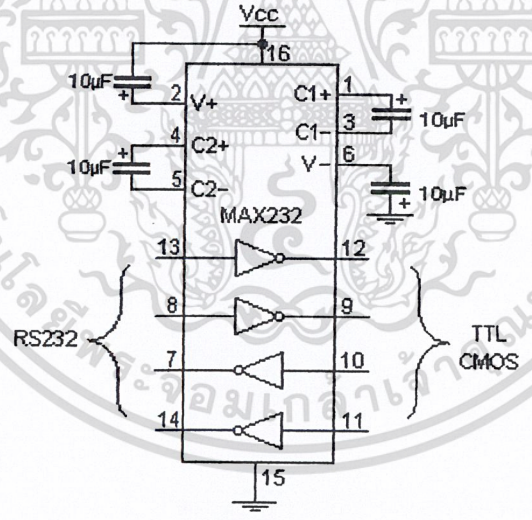
ขาที่ 6 Data Set Ready

ขาที่ 7 Request to Send

ขาที่ 8 Clear to Send

ขาที่ 9 Ring Indicator

โดยพอร์ต RS – 232 นี้ เราจะต่อใช้งานในวงจรเพียง 2 ขา คือขา GND และขา RXD แต่เนื่องจากตัวพอร์ต RS- 232 ไม่สามารถที่จะใช้ต่อกับตัว CPU โดยตรงได้ต้องมีการต่อใช้งานร่วมกับ MAX 232 ซึ่ง MAX 232 เนื่องจากปรกติ UART ทุกตัวจะมีระดับสัญญาณตามมาตรฐาน TTL คือ 0Volt และ +5Volt ดังนั้นจึงต้องมีวงจรที่ทำหน้าที่เป็น RS-232 Converter เพื่อแปลงสัญญาณระหว่าง TTL และ RS- 232 อีกทีหนึ่ง เช่น IC MAX - 232 ซึ่งมีวงจร charge pump โดยใช้ Capaciotor ค่า 10 uF สามารถสร้างไฟ +10 V และ –10 V จากไฟ +5 V ได้ พร้อมทั้งมี 2 Tx และ 2 Rx อยู่ใน package เดียวกัน และรองรับ baud rate ได้ถึง 120 Kbps จึงสะดวกมากเพราะใช้ IC เพียงตัวเดียว

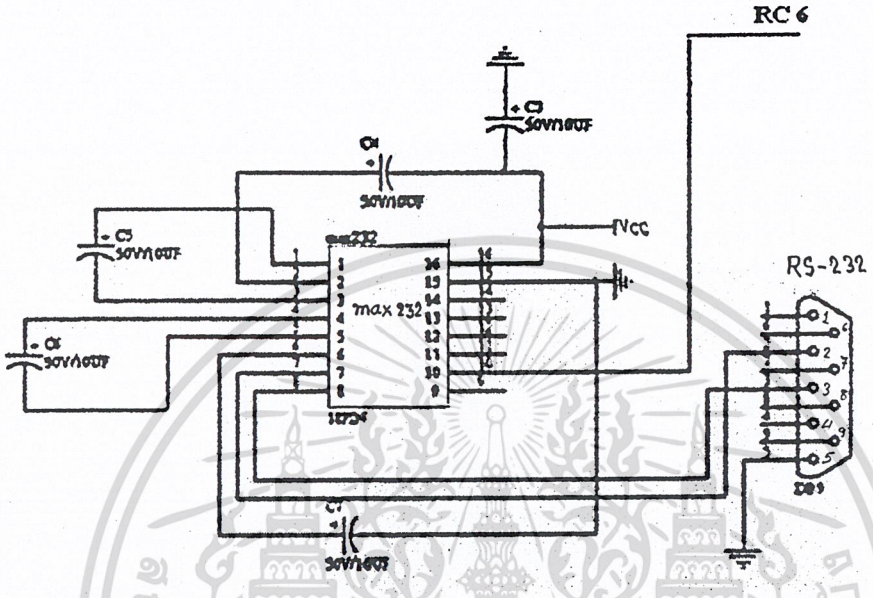


รูปที่ 3.8 แสดงการต่อ MAX – 232 เพื่อใช้งาน

จากรูปด้านล่าง เป็นการต่อใช้งานจริงในวงจร C3 นำมาต่อคร่อมขาไฟกับ GND เพื่อทำหน้าที่กรองกระแสให้เรียบ ส่วน C4,C5,C6,C7 เป็นวงจร charge pump สามารถสร้างไฟ +10 V และ –10 V จากไฟ +5 V ขาที่ 2 ของ RS-232 เข้าที่ขา 7 ของ MAX 232 ขาที่ 3 ของ RS-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

232 เข้าที่ขา 8 ของ MAX 232 ขาที่ 2 ของ RS-232 ต่อลง GND ขาที่ 10 ของ MAX 232 ต่อเข้าพอร์ต TX ของ CPU เพื่อรับข้อมูล



รูปที่ 3.9 แสดงการต่อใช้ RS – 232 ร่วมกับ MAX 232

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1.5 ส่วนของ CPU

ไมโครคอนโทรลเลอร์ที่ใช้เป็น Pic 16F877 มี 40 มีพอร์ตใช้งาน 5 พอร์ต โดยรายละเอียดต่างต่างของขาสามารถดูได้จากบทที่ 2 เรื่อง PIC 16F877 ซึ่งมีขีดความสามารถเพียงพอต่อความต้องการ คือ มีโมดูลการส่งข้อมูลแบบซิงโครนัสในตัวเอง มีโมดูลสำหรับใช้กับพอร์ต RS – 232 และมีจำนวนพอร์ตการใช้งานที่เพียงพอแก่ความต้องการ เป็นต้น โดยในการเลือกใช้จำเป็นต้องเป็น Pic 16F877 ที่สามารถใช้กับ สัญญาณความถี่ 20 MHz ด้วย ทำให้ให้มีความเร็วในการกระทำคำสั่งต่างๆ ได้ทันที่จะกลับมารอรับข้อมูลที่เข้ามาใหม่ได้ทัน ซึงการใช้สัญญาณนาฬิกา 20 MHz นี้ต้องเป็นซีพียู 16F877 ที่สามารถใช้กับความถี่ 20 MHz ได้ด้วย

เพื่อความแน่นอนของสัญญาณที่ได้รับจากการจับสัญญาณ ในพอร์ตที่ทำหน้าที่รับสัญญาณจะต้องทำการต่อ PULL UP ไว้ด้วย

สำหรับการควบคุมการทำงานจะมีสวิตช์อยู่ 3 ตัวด้วยกันทำหน้าที่ต่างๆ

สวิตช์ Reset จะเป็นสวิตช์แบบ 2 ขั้ว จะเชื่อมต่อระหว่าง GND และพอร์ต MCLR (ขาที่ 1) และที่ขา MCLR จะต่ออยู่กับไฟ 5 โวลท์ โดยมี ตัวต้านทานต่อขึ้นอยู่ การทำงานในปกตินั้นพอร์ตนี้จะได้รับสัญญาณ 5 โวลท์ แต่เมื่อเราทำการกดสวิตช์ จะได้รับสัญญาณ 0 โวลท์ ซึ่งเป็นการสั่ง Reset

ส่วนสวิตช์อีก 2 ตัวจะเป็นสวิตช์แบบ 3 ขั้ว ขั้วกลางจะต่ออยู่กับ พอร์ตของไมโครคอลโทรลเลอร์ ซึ่งทำการเซตเป็น input แล้ว ส่วนอีก 2 ขาที่เหลือ ขาหนึ่งจะต่อลง GND อีกขาหนึ่งจะต่อกับไฟ 5โวลท์

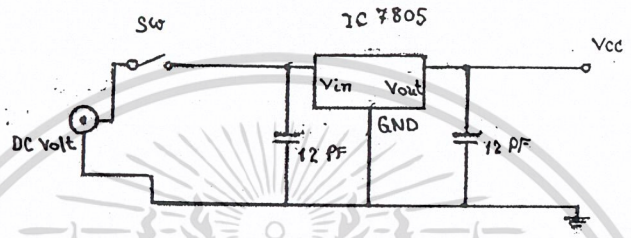
สวิตช์ที่ต่อกับขา ที่ 2 ของไมโครคอนโทรลเลอร์นั้น จะเป็นสวิตช์ที่ใช้เลือกว่าจะทำการจับสัญญาณหรือ ทำการแสดงค่าที่ทำการจับมาได้

สวิตช์ที่ต่อกับขา ที่ 5 ของไมโครคอนโทรลเลอร์นั้น จะเป็นสวิตช์ที่ใช้เลือกว่าจะแสดงค่าที่ LCD หรือ จะ แสดงค่าที่เก็บไว้ออกทาง RS – 232 พร้อมกับจะแสดงการใช้พื้นที่หน่วยความจำออกทาง LCD พร้อมกันด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1.6 ส่วนวงจรไฟเลี้ยง

หลังจากที่เราป้อนไฟให้วงจร โดยใช้จากแหล่งจ่ายแบบ DC ขนาด 6-9 Voltเข้าสู่บอร์ดเริ่มต้นเราจะมีสวิตช์ SW1 ไว้สำหรับเปิด-ปิด การจ่ายไฟให้กับวงจรทั้งหมด คอนเดนเซอร์ C1 ทำหน้าที่กรองกระแสให้เรียบ ต่อมาเป็น ไอซี Regulate ขนาด 5 Volt ที่จะทำ หน้าที่ปรับ และรักษาระดับแรงไฟ DC 5 VOLT ให้คงที่ คอนเดนเซอร์ C2 ทำหน้าที่กรองกระแส



รูปที่ 3.10 วงจรแสดงไฟเลี้ยง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.0 การทำงานของโปรแกรมเครื่องจับสัญญาณอนุกรม

การทำงานของโปรแกรมเริ่มต้นจากการตั้งค่าเริ่มต้นสำหรับพารามิเตอร์ต่างๆ เซ็ตพอร์ต การทำงาน การสื่อสารข้อมูลอนุกรม

การสื่อสารข้อมูลอนุกรม (RS-232) ใช้อัตราเร็วในการรับส่งข้อมูลเท่ากับ 9,600 บิตต่อวินาที โดยมีขนาดของข้อมูลเป็น 8 บิตไม่มีพาริตี้บิต และมีบิตหยุด 1 บิต

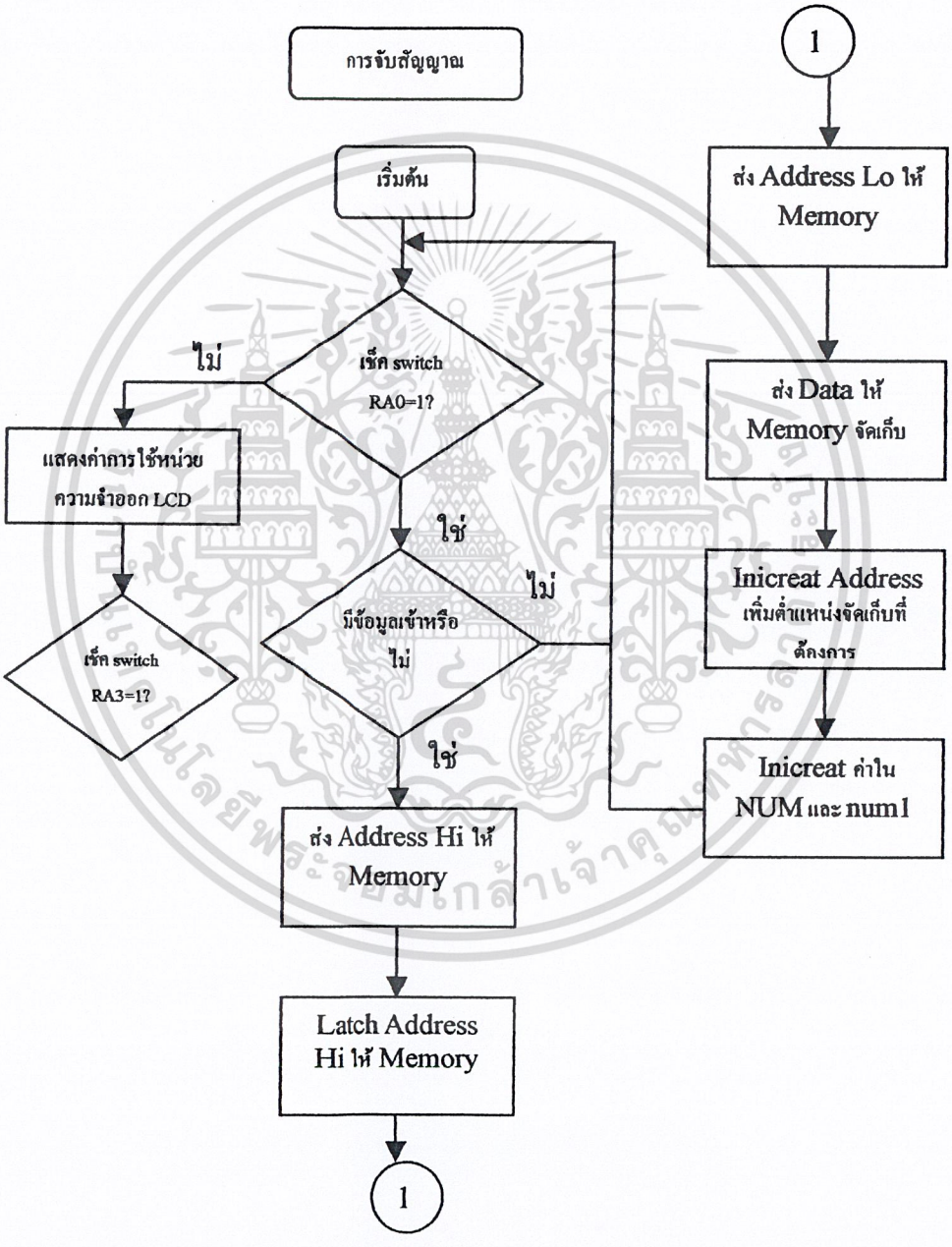
จากนั้นการทำงานของโปรแกรมจะแบ่งออกได้เป็นส่วนใหญ่ๆ 3 ส่วนด้วยกัน

3.3.1 ส่วนของการจับสัญญาณการติดต่อ

โปรแกรมจะทำการเช็คว่าต้องการให้ทำการจับสัญญาณหรือแสดงค่าที่จัดเก็บไว้ โดยเช็คที่พอร์ต RA1 มีค่าเป็น “1” หรือ “0” หากพอร์ตที่เช็คได้มีค่าเป็น “0” จะไปทำการเช็คพอร์ต RA3 ต่อเพื่อไปเลือกทำงานในส่วนของการแสดงผลที่ LCD หรือ RS-232 หากเป็น “1” จะทำการจับสัญญาณการติดต่อ โปรแกรมจะไปทำงานที่ Label READ_DATA จากนั้นโปรแกรมจะทำการตรวจสอบว่ามีข้อมูลถูกส่งเข้ามาหรือไม่โดยเช็คได้ที่บิต RCIF ของรีจีสเตอร์ PIR1 คือมีข้อมูลเข้ามาบิตนี้จะถูกเซตเป็น “1” โดยอัตโนมัติ หากมีข้อมูลเข้ามาข้อมูลที่เข้ามาข้อมูลจะถูกเก็บอยู่ที่รีจีสเตอร์ RCREG ของโมดูล UART โปรแกรมจะทำการย้ายข้อมูลที่อยู่ภายในรีจีสเตอร์ RCREG ไปไว้ใน รีจีสเตอร์ W (Work Register) แล้วทำการย้ายข้อมูลที่อยู่ภายในรีจีสเตอร์ W (Work Register) ไปไว้ใน General Register ชื่อ CUR ที่เรากำหนดขึ้น จากนั้นส่งข้อมูลของตำแหน่งที่ต้องการให้เขียนไปบนบัสตำแหน่ง พร้อมทั้งสัญญาณการเลือกใช้ชิพ จากนั้นโปรแกรมจะทำการย้ายข้อมูลที่อยู่ภายในรีจีสเตอร์ CUR ไปไว้ใน รีจีสเตอร์ PORTB เพื่อส่งข้อมูลออกทาง PORTB เพื่อข้อมูลที่จะเป็นข้อมูลอินพุต ถูกส่งเข้ามายังอินพุตของแรม หน่วยความจำต้องการเวลาสักระยะหนึ่ง เพื่อให้ข้อมูลในการกำหนดตำแหน่งอยู่ในสภาวะคงที่ซึ่งช่วงเวลานี้คือ Address setup time ซึ่งจะมีค่าประมาณ 20 nS จากนั้นระบบจึงส่งสัญญาณควบคุมการเขียนให้อยู่ในสภาวะแอกตีฟ หลังจากที่สัญญาณควบคุมการเขียนแอกตีฟชักระยะหนึ่ง หน่วยความจำจะทำการนำข้อมูลจากอินพุตเข้าเขียนเข้าไปยังหน่วยความจำ และเมื่อสัญญาณการเขียนเลิกแอกตีฟ ก็เป็นการสิ้นสุดกระบวนการเขียน จากนั้น ทำการ Inicreat Address เพิ่มตำแหน่งจัดเก็บที่ต้องการและ Inicreat ค่าใน NUM และ num1 หากค่าในรีจีสเตอร์ข้างต้นมีค่าถึง 256 เมื่อไหร่โปรแกรมจะเว็ตเป็น 0 แล้วไปทำการ Inicreat ค่าใน MAKE และ MAKE1ขึ้น1ค่า เพื่อสามรถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รู้จำนวนข้อมูลที่จัดเก็บไป จากนั้นโปรแกรมจะกลับไปรอข้อมูลเพื่อนำข้อมูลมาบันทึกในตำแหน่งต่อไปของหน่วยความจำ



รูปที่ 3.11 ขั้นตอนการจับสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2.0 ส่วนของการแสดงผล

เมื่อโปรแกรมเช็คพอร์ต RA2 แล้วได้ค่าเป็น “0” โปรแกรมจะทำการเซ็ต LCD โดยมีรายละเอียดดังนี้ การติดต่อเป็นแบบ 4 บิต (เนื่องจากมีจำนวนอินพุต - เอาท์พุต อยู่จำกัด) แสดง 2บรรทัด 5 X 7 เคอร์เซอร์ปิดและไม่กระพริบ เมื่อเปิดเข้าไปจะเพิ่ม Address ขึ้น 1 เลื่อนไปทางขวา

ที่ Address 0x80 ของ LCD จะกำหนดให้แสดงคำว่า “ USEBYTE ” ส่วนที่ตำแหน่ง 0xC0 จะแสดงจำนวนตำแหน่งที่หน่วยความจำถูกใช้ไป โดยค่าที่แสดงจะแสดงออกเป็นเลขฐาน 16 โดย ค่า 2 ตัวแรกจะเป็นค่าที่ได้จากรีจีสเตอร์ MAKE ซึ่งแสดงค่า 8 บิตบน โปรแกรมจะทำการส่งคำสั่งเขียน LCD จากนั้นทำการย้ายข้อมูลที่อยู่ภายในรีจีสเตอร์ MAKE ให้ไปอยู่ในอยู่ในรีจีสเตอร์ W แล้วทำการแปลงเลขฐาน 16 เป็นรหัส ascii แล้วทำการเก็บไว้ในรีจีสเตอร์ W จากนั้นโปรแกรมส่งคำสั่ง LCD DATA เพื่อแสดงค่าในรีจีสเตอร์ MAKE ออก LCD ค่า 2 ตัวหลังจะเป็นค่าที่ได้จากรีจีสเตอร์ NUM ซึ่งแสดงค่า 8 บิตล่าง โปรแกรมจะทำการส่งคำสั่งเขียน LCD จากนั้นทำการย้ายข้อมูลที่อยู่ภายในรีจีสเตอร์ NUM ให้ไปอยู่ในอยู่ในรีจีสเตอร์ W แล้วทำการแปลงเลขฐาน 16 เป็นรหัส ascii แล้วทำการเก็บไว้ในรีจีสเตอร์ W จากนั้นโปรแกรมส่งคำสั่ง LCD DATA เพื่อแสดงค่าในรีจีสเตอร์ NUM ออก LCD เป็นการแสดงจำนวนพื้นที่ในหน่วยความจำที่ถูกใช้ไป จากนั้นโปรแกรมจะทำการเช็คพอร์ต RA3 ถ้าเป็น “1” จะทำในส่วนของการติดต่อกับคอมพิวเตอร์ถ้าเป็น “0” จะทำในส่วนของการติดต่อกับ LCD

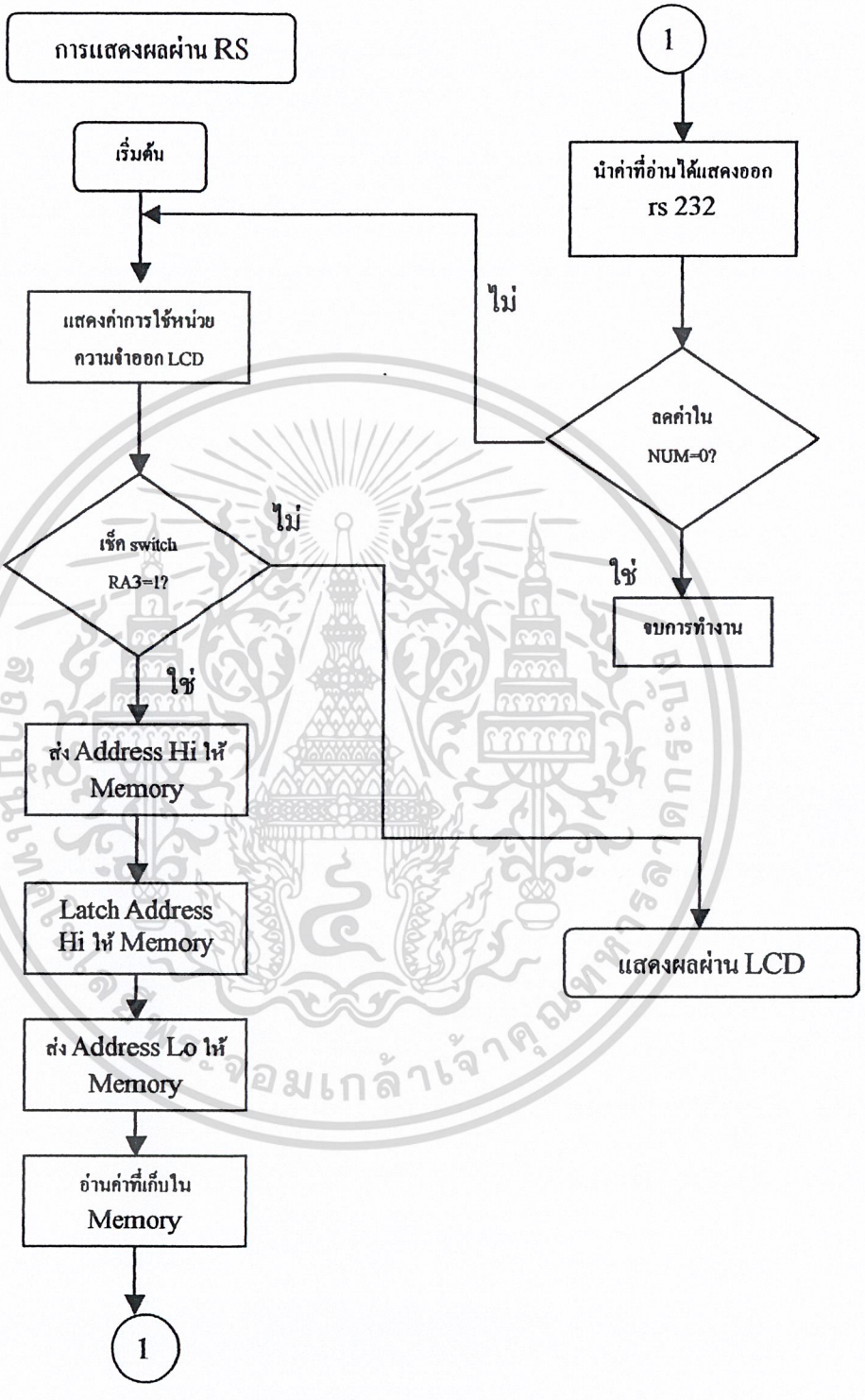
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2.1 ส่วนของการติดต่อกับคอมพิวเตอร์

เมื่อโปรแกรมจะทำการเช็คพอร์ต RA3 ถ้าเป็น "1" จะทำในส่วนของการติดต่อกับคอมพิวเตอร์โปรแกรมส่งข้อมูลของตำแหน่งที่ต้องการอ่านออกไปบนบัสตำแหน่ง (กำหนดให้ในขณะนี้สัญญาณการอ่านแอกตีฟ) โปรแกรมส่งสัญญาณเลือกใช้ชิพและส่งไปยังหน่วยความจำรอเวลาสักระยะหนึ่ง นับตั้งแต่ส่งข้อมูลเพื่อกำหนดตำแหน่งเข้าไป ซึ่งเวลานี้ก็คือ Access Time ซึ่งอาจมีค่าประมาณ 35 – 200 nS แล้วแต่ชนิดและขนาดของแรม หลังจากช่วง Access Time ผ่านไปแล้ว ข้อมูลจากหน่วยความจำก็จะปรากฏบนบัสข้อมูล ที่ PORTB ดังนั้นข้อมูลที่อ่านได้จะถูกเก็บไว้ในรีจีสเตอร์ PORTB จากนั้นทำการย้ายข้อมูลที่อยู่ภายในรีจีสเตอร์ PORTB ให้ไปอยู่ในอยู่ในรีจีสเตอร์ W

โปรแกรมสัญญาณเลือกชิพเลิกแอกตีฟ บัสข้อมูลกลับสู่สภาวะอิมพีแดนซ์สูง จากนั้นโปรแกรมเช็คว่าทีบิต TXIF ของรีจีสเตอร์ PIR1 มีค่าเป็น "1" หรือไม่หากเป็น "1" แสดงว่าไม่มีการส่งข้อมูลอื่นอยู่ หาก PIR1 มีค่าเป็น "0" แสดงว่ามีการส่งข้อมูลอื่นอยู่ก็จะต้องรอไปก่อน เมื่อ PIR1 มีค่าเป็น "1" แล้วทำการแปลงค่าที่อ่านมาได้ซึ่งเก็บอยู่ที่รีจีสเตอร์ W ให้เป็นค่า ascii โดยทำการแปลง 4 บิตบนก่อน แล้วส่งค่าที่ได้ให้รีจีสเตอร์ TXREG เพื่อส่งข้อมูลไปแสดงที่คอมพิวเตอร์ จากนั้นโปรแกรมเช็ค บิต TRMT ของรีจีสเตอร์ TXSTA ว่าเป็น "1" หรือยังซึ่งหากเป็น "1" แสดงว่าส่งข้อมูลเรียบร้อยแล้วจึงทำการส่ง 4 บิตล่างต่อไป โดยทำการแปลงค่า 4 บิตที่อ่านมาได้ซึ่งเก็บอยู่ที่รีจีสเตอร์ W ให้เป็นค่า ascii แล้วส่งค่าที่ได้ให้รีจีสเตอร์ TXREG เพื่อส่งข้อมูลไปแสดงที่คอมพิวเตอร์ จากนั้นโปรแกรมเช็ค บิต TRMT ของรีจีสเตอร์ TXSTA ว่าเป็น "1" หรือยังซึ่งหากเป็น "1" แสดงว่าส่งข้อมูลเรียบร้อยแล้ว ทำการลดค่าใน NUM และเพิ่ม Address Lo แล้วย้อนกลับทำการอ่านข้อมูลเช่นนี้ต่อไปเรื่อยๆต่อไปจนกว่าจะหมด โดยถ้า NUM เป็น "0" โปรแกรมจะทำการเซ็ตค่าให้ NUM เป็น 0xff และลดค่าใน MAKE ลง "1" ไปเรื่อยๆ จนกว่าค่าใน MAKE และ NUM จะเป็น "0" ทั้งคู่แสดงว่าได้อ่านข้อมูลหมดแล้ว ซึ่งการลดค่านี้จะต้องทำควบคู่ไปกับการเพิ่ม Address Lo และ Address Hi โดยเมื่อ Address Lo ถึง 0xff แล้วจะทำการเซ็ตเป็น "0x00" ใหม่แล้วทำการเพิ่มค่าใน Address Hi ขึ้น "1" เพื่ออ่านข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



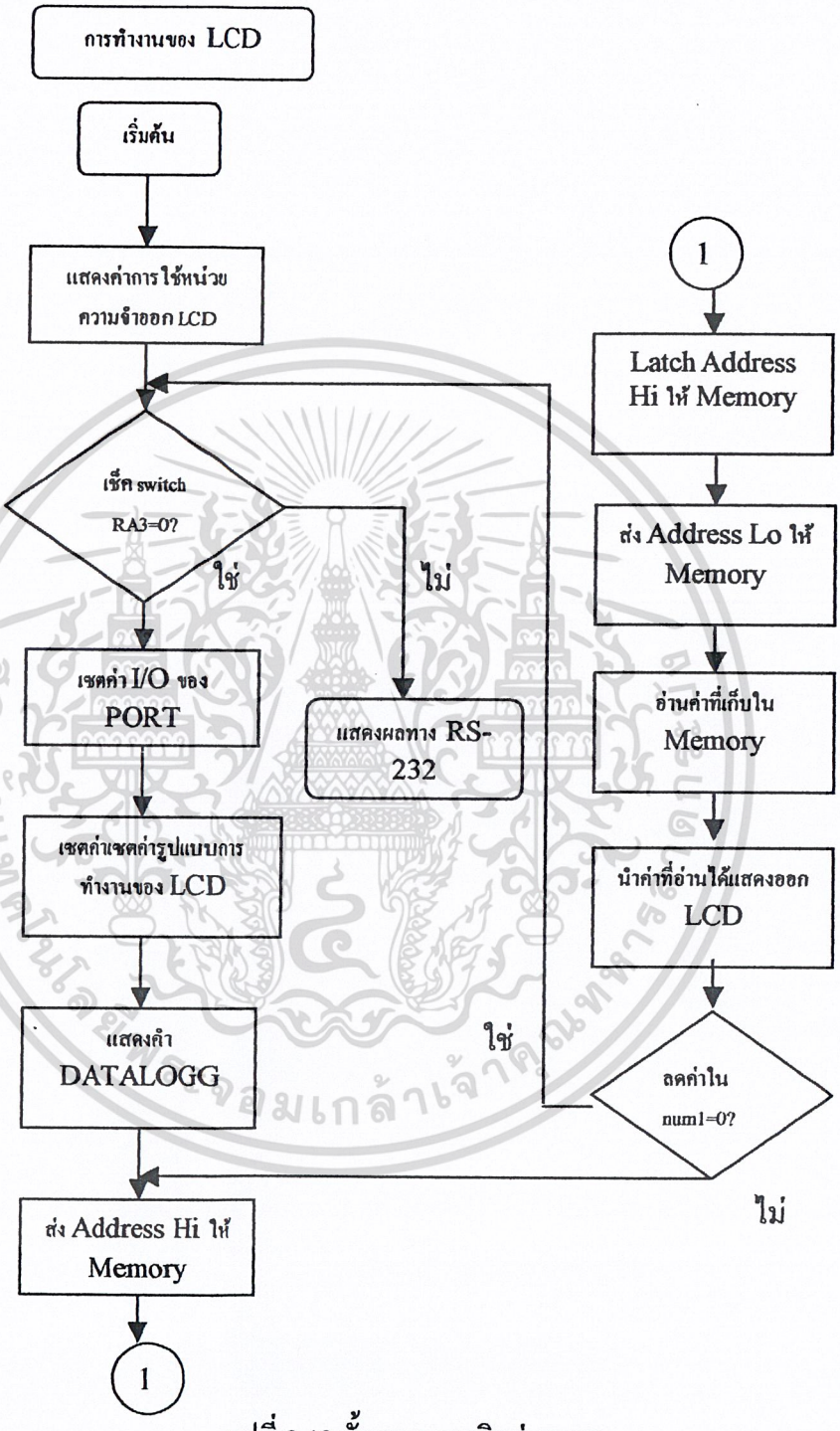
รูปที่ 3.12 ขั้นตอนการส่งข้อมูลให้คอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2.2 ส่วนของการติดต่อกับ LCD

เมื่อโปรแกรมจะทำการเช็คพอร์ต RA3 ถ้าเป็น "0" จะทำในส่วนของการติดต่อกับ LCD โปรแกรมทำการเซ็ต LCD ที่ Address 0x80 ของ LCD จะกำหนดให้แสดงคำว่า " DATALOGG " ส่วนที่ตำแหน่ง 0xC0 จะแสดงข้อมูลที่ถูกจัดเก็บไว้ โปรแกรมส่งข้อมูลของตำแหน่งที่ต้องการอ่านออกไปบนบัสตำแหน่ง (กำหนดให้ในขณะนี้สัญญาณการอ่านแอกตีฟ) โปรแกรมส่งสัญญาณเลือกใช้ชิพและส่งไปยังหน่วยความจำ รอเวลาสักระยะหนึ่ง นับตั้งแต่ส่งข้อมูลเพื่อกำหนดตำแหน่งเข้าไป ซึ่งเวลานี้ก็คือ Access Time ซึ่งอาจมีค่าประมาณ 35 – 200 nS แล้วแต่ชนิดและขนาดของแรม หลังจากช่วง Access Time ผ่านไปแล้ว ข้อมูลจากหน่วยความจำก็จะปรากฏบนบัสข้อมูล ที่ PORTB ดังนั้นข้อมูลที่อ่านได้จะถูกเก็บไว้ในรีจีสเตอร์ PORTB จากนั้นทำการย้ายข้อมูลที่อยู่ภายในรีจีสเตอร์ PORTB ให้ไปอยู่ในอยู่ในรีจีสเตอร์ W โปรแกรมสัญญาณเลือกชิพเลิกแอกตีฟ บัสข้อมูลกลับสู่สภาวะอิมพีแดนซ์สูง จากนั้นทำการย้ายข้อมูลที่อยู่ภายในรีจีสเตอร์ W ให้ไปอยู่ในอยู่ในรีจีสเตอร์ DATAS จากนั้นทำการย้ายข้อมูลที่อยู่ภายในรีจีสเตอร์ DATAS ให้ไปอยู่ในอยู่ในรีจีสเตอร์ W ทำการแปลงค่า 4 บิตที่บนอ่านมาได้ซึ่งเก็บอยู่ที่รีจีสเตอร์ W ให้เป็นค่า ascii แล้วส่งคำสั่ง LCD Command ให้ LCD เพื่อส่งข้อมูลไปแสดงที่ LCD โดยส่งคำสั่ง LCD DATA จากนั้นทำการแปลงค่า 4 บิตที่บนอ่านมาได้ซึ่งเก็บอยู่ที่รีจีสเตอร์ W ให้เป็นค่า ascii แล้วส่งคำสั่ง LCD Command ให้ LCD เพื่อส่งข้อมูลไปแสดงที่ LCD โดยส่งคำสั่ง LCD DATA ทำการลดค่าใน num1 และเพิ่ม Address Lo แล้วย้อนกลับทำการอ่านข้อมูลเช่นนี้ต่อไปเรื่อยๆต่อไปจนกว่าจะหมด โดยถ้า num1 เป็น "0" โปรแกรมจะทำการเซ็ตค่าให้ num1 เป็น 0xff และลดค่าใน MAKE1 ลง "1" ไปเรื่อยๆ จนกว่าค่าใน MAKE1 และ num1 จะเป็น "0" ทั้งคู่แสดงว่าได้อ่านข้อมูลหมดแล้ว ซึ่งการลดค่านี้จะต้องทำควบคู่ไปกับการเพิ่ม Address Lo และ Address Hi โดยเมื่อ Address Lo ถึง 0xff แล้วจะทำการเซ็ตเป็น "0x00" ใหม่แล้วทำการเพิ่มค่าใน Address Hi ขึ้น "1" เพื่ออ่านข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.13 ขั้นตอนการติดต่อ LCD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดลอง

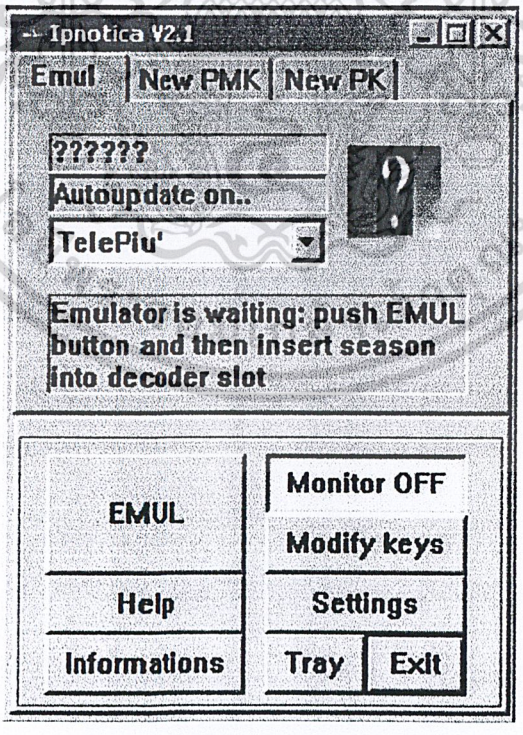
4.1 การทดสอบความถูกต้องของเครื่อง

การทดลองที่ทำในโปรเจ็คที่ 2 นี้จะเป็นการทดสอบความถูกต้องแม่นยำในการทำงานของโปรแกรม และ ตัวเครื่องที่ได้ทำการออกแบบ

4.1.1 โปรแกรม Ipnotica

การทดสอบความถูกต้องของการออกแบบ เครื่องจับสัญญาณอนุกรม จะใช้เครื่อง EMULATOR SMARTCARD ของ RECEIVER ในการทดสอบ

โปรแกรมการทำงานที่ใช้ประกอบกับเครื่องในการทดสอบคือโปรแกรม Ipnotica ซึ่งสามารถจะแสดงผลให้สามารถเปรียบเทียบความถูกต้องของโปรแกรมที่จับมาได้

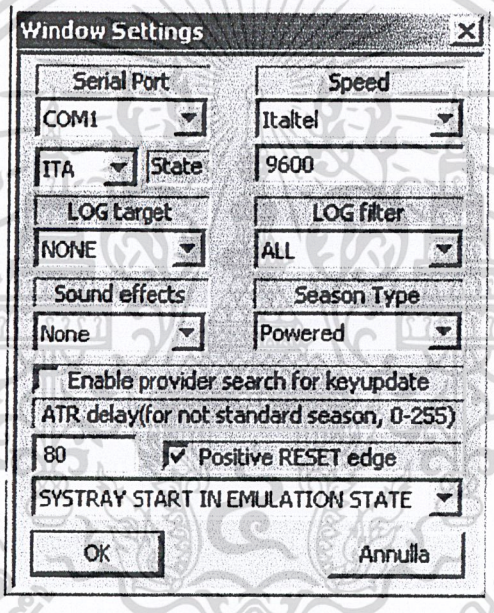


รูปที่ 4.1 แสดงหน้าจอหลักของโปรแกรม Ipnotica

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนการใช้งาน

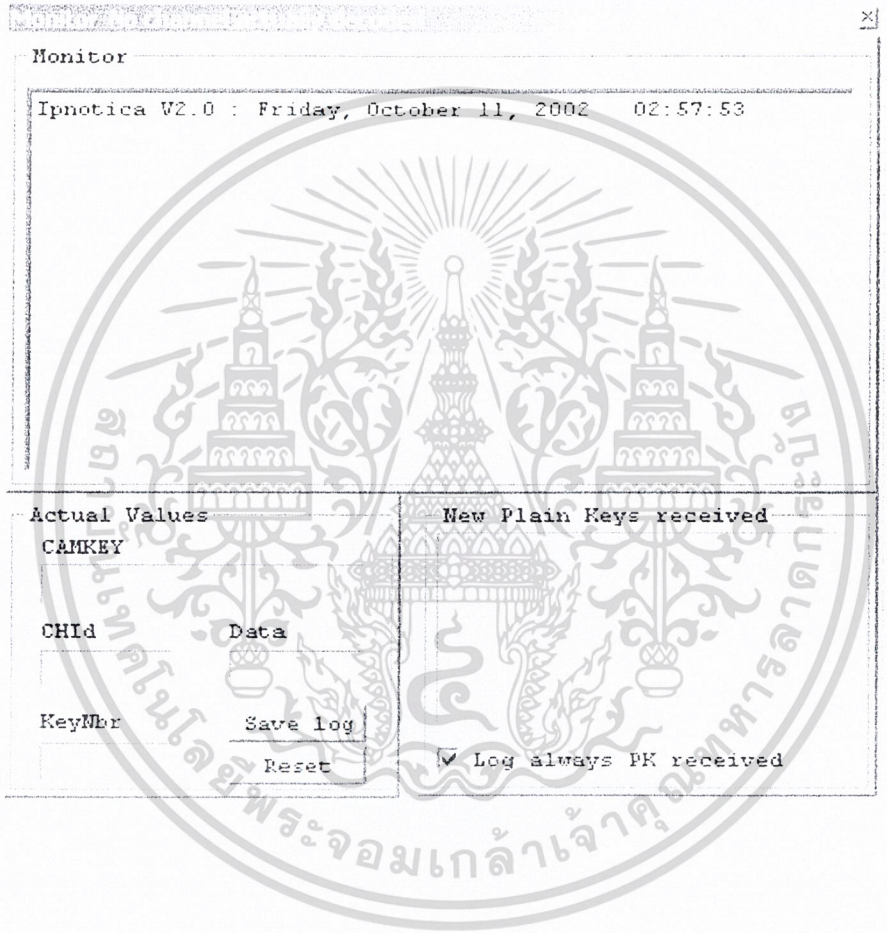
1. เมื่อทำการรันโปรแกรมแล้วจะต้องกดปุ่ม Settings เพื่อทำการเซตการทำงานต่างๆ เช่น พอร์ตที่ใช้ต่อกับการ์ดอีมูเลเตอร์ , Buab Rate ที่ใช้ในการติดต่อ ในการทดลองนี้ตั้งค่าที่ 9600 bps ดังรูปที่ 4.2



รูปที่ 4.2 แสดงหน้าการ Settings

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

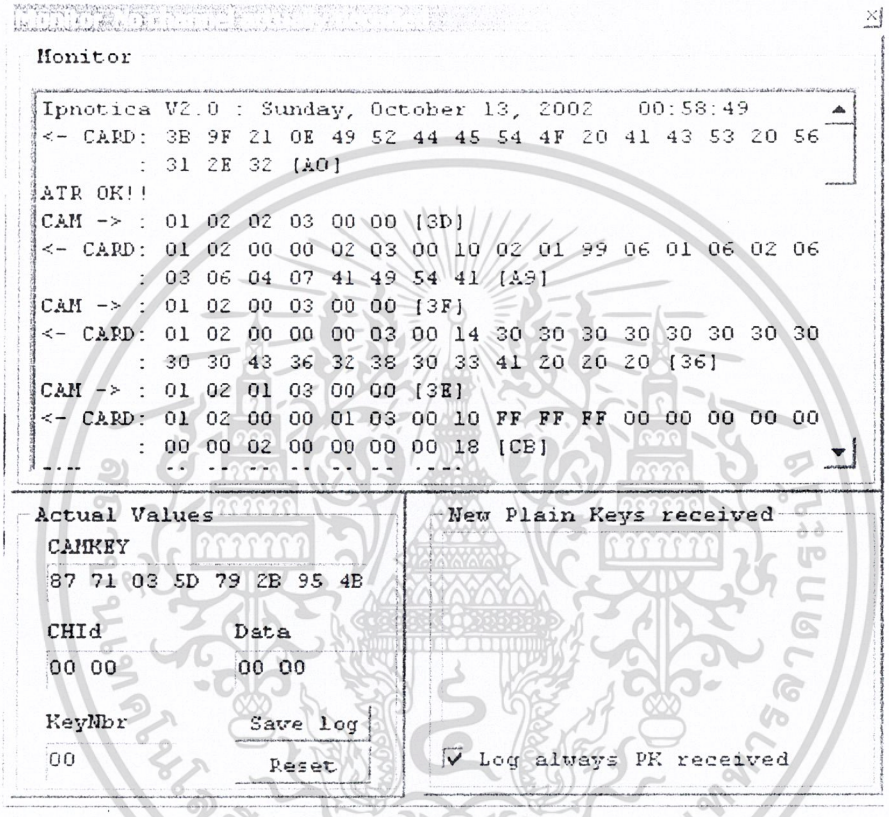
2. กดปุ่ม Monitor เพื่อเปิดหน้าจอแสดงผล เพื่อแสดงข้อมูลที่อีมูเลเตอร์มาได้ เพื่อนำมาเปรียบเทียบกับการจับด้วยเครื่องที่จัดทำขึ้น



รูปที่ 4.3 หน้าจอการแสดงผลของโปรแกรม Ipnotica

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. กดปุ่ม EMUL เพื่อเริ่มต้นจับสัญญาณ เมื่อเสียบการ์ดเข้ากับตัวเครื่อง Receiver แล้วจะแสดงผลดังรูป 4.4



รูปที่ 4.4 การแสดงข้อมูลที่เกิดจากการติดต่อระหว่างบัตรกับเครื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2 ขั้นตอนการทดสอบความถูกต้อง

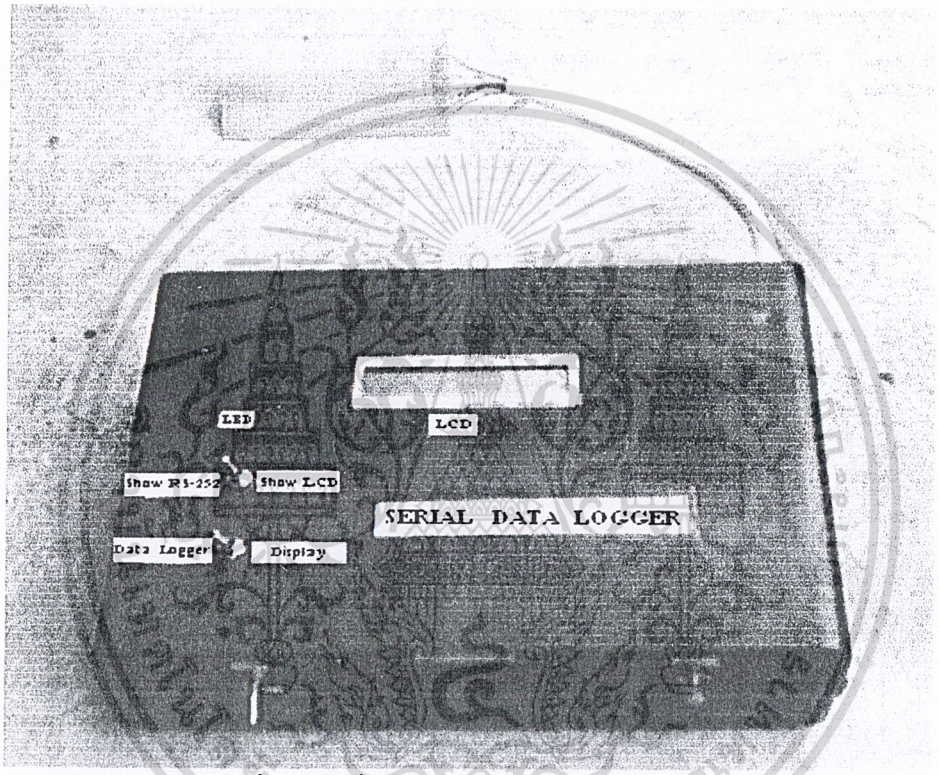
สามารถทำการทดสอบความถูกต้องในการทำงานของเครื่องได้โดย นำการ์ดมาเสียบเข้าที่ Slot ของตัวเครื่องจับแล้วนำหน้าคอนแทคของเครื่องจับไปเสียบเข้าตัว Receiver แล้วแสดงข้อมูลที่จัดเก็บไว้แสดงออกทางพอร์ตอนุกรมด้วยโปรแกรมสำหรับแสดงผล จากนั้นทำการเปรียบเทียบกับข้อมูลที่แสดงออกที่โปรแกรม Ipnotica ว่าตรงกันหรือไม่ (เปรียบเทียบรูปที่ 4.4 กับ 4.5 ว่าตรงกันหรือไม่)

รูปที่ 4.5 การแสดงข้อมูลที่จัดเก็บไว้ออกทางพอร์ตอนุกรม

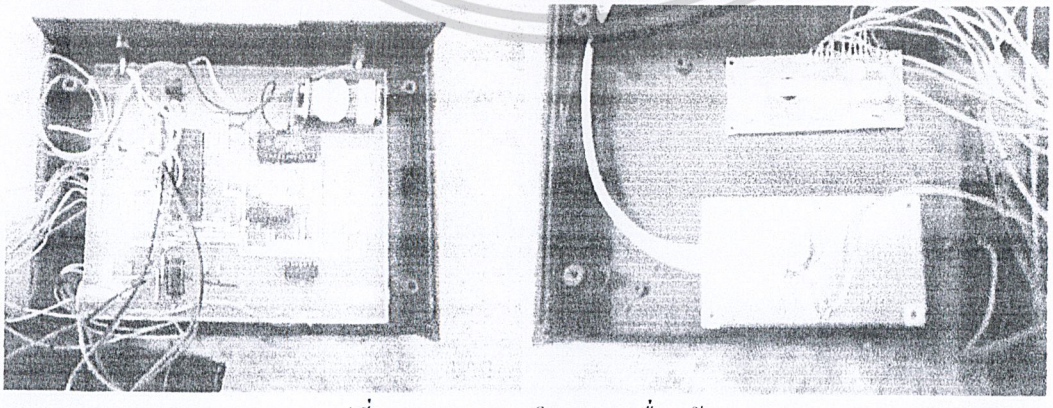
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 เครื่องต้นแบบ SERIAL DATA LOGGER

ตัวเครื่องต้นแบบที่จัดสร้างเสร็จเรียบร้อยแล้วจะมี ลักษณะภายนอกและวงจรภายใน ดังรูปที่ 4.6 และรูปที่ 4.7 ตามลำดับ



รูปที่ 4.6 เครื่อง SERIAL DATA LOGGER



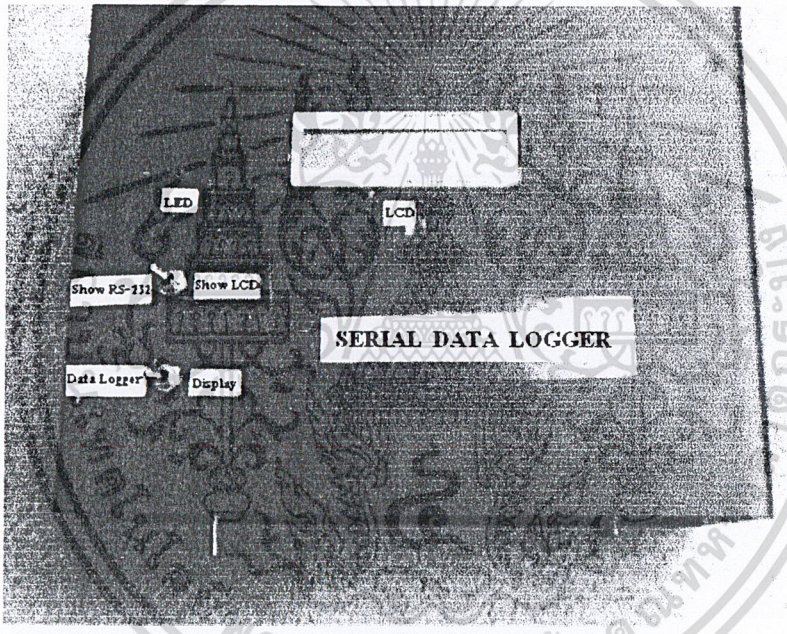
รูปที่ 4.7 วงจรภายในของเครื่องต้นแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1 รายละเอียดของ ตัวเครื่องต้นแบบ

รายละเอียดการใช้งานของสวิตช์และปุ่มต่างๆ แต่ละปุ่ม รวมทั้งตำแหน่ง ส่วนรวมทั้งส่วนที่ใช้ในการแสดงผล จะแยกอธิบาย ตามตำแหน่งที่จัดว่าง เป็น 3 ส่วนใหญ่ๆ คือ ส่วนที่อยู่ด้านบน ด้านหน้า และด้านหลังตามลำดับ เพื่อง่ายต่อการทำความเข้าใจ

4.2.1.1 ด้านบนของตัวเครื่อง



รูปที่ 4.8 ภาพด้านบนของเครื่องต้นแบบ

ด้านบนของเครื่องจะมีรายละเอียด ดังนี้

1. ปุ่ม Data Logger / Display เป็นปุ่มที่ใช้เพื่อทำการเลือกว่าต้องการที่จะให้เครื่องทำงานที่สถานะทำการจับสัญญาณการติดต่อในกรณีที่สวิตช์อยู่ที่ตำแหน่ง Data Logger หรือต้องการที่จะให้เครื่องทำงานที่สถานะการแสดงผลของข้อมูลที่ทำการจัดเก็บไว้ในหน่วยความจำในกรณีที่ตำแหน่งของสวิตช์อยู่ที่ Display (ซึ่งในสถานะที่ทำการแสดงผลนี้จะต้องมีการเลือกสถานะการทำงานของสวิตช์ Show RS-232 / Show LCD ร่วมด้วย ว่าต้องการให้แสดงในรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

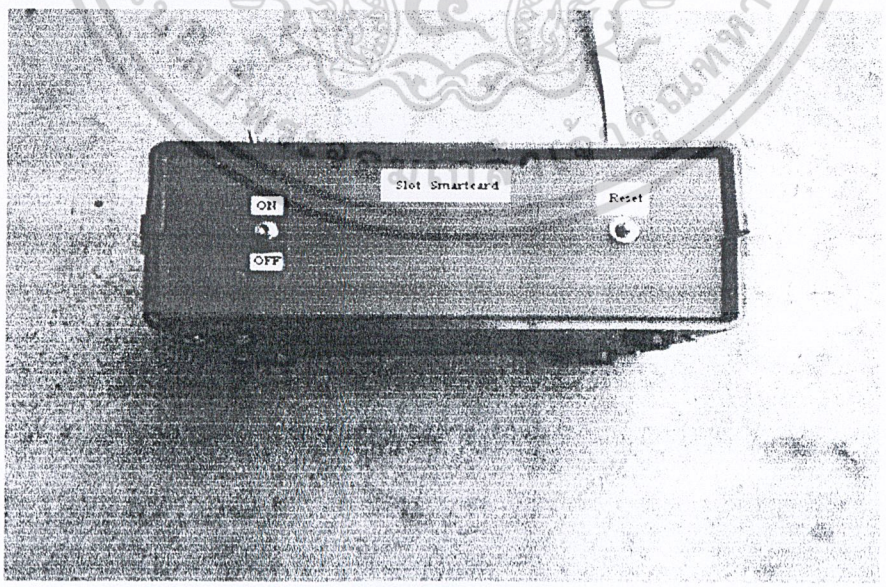
แบบใด) จะแสดงข้อมูลที่ถูกจัดเก็บไว้และบอกถึงตำแหน่งของหน่วยความจำที่ถูกใช้ไปทาง LCD

2. ปุ่ม Show RS-232 / Show LCD เป็นปุ่มที่ใช้สำหรับเลือกสถานะว่าต้องการที่จะให้เครื่องทำการแสดงผลที่จัดเก็บออกมาในรูปแบบใด ในกรณีที่ มีการเลือกสวิตช์ Data Logger / Display ไปที่ตำแหน่ง Display จะแสดงผลออก RS-232 เมื่อเลือกไปที่ Show – RS 232 และแสดงผลออก LCD เมื่อเลือกไปที่ Show LCD

3. ขอแสดงผล LCD จะทำการแสดงผลเมื่อทำการเลือกให้ Show LCD โดยนำข้อมูลที่ทำการจัดเก็บไว้ในหน่วยความจำออกมาแสดงผล โดยจะทำการแสดงผลเป็นเลขฐาน 16 ซึ่งเป็นโค๊ตที่ใช้ในการส่งข้อมูล หากเลือก ปุ่ม Data Logger / Display ไปที่ตำแหน่ง Display พร้อมเลือก สวิตช์ Show RS-232/Show LCD ไปที่ Show RS-232 พร้อมกัน จะทำการแสดงตำแหน่งของหน่วยความจำที่ถูกใช้ไปในการจัดเก็บข้อมูล โดยทำการแสดงออกเป็นเลขฐาน 16

4. LED จะใช้สำหรับแสดงว่าสถานะในขณะนั้นเครื่องทำการตรวจจับสัญญาณอยู่หรือเสร็จสิ้นกระบวนการตรวจจับสัญญาณแล้ว โดยดูได้จาก LED หาก LED ดับแสดงว่าไม่ได้ทำการตรวจจับสัญญาณอยู่ในขณะนั้น แต่หากมีการกระพริบของ LED แสดงว่าขณะนั้น มีการตรวจจับและจัดเก็บสัญญาณการติดต่ออยู่

4.2.1.2 ด้านหน้าของตัวเครื่อง



รูปที่ 4.9 ภาพด้านหน้าของเครื่องต้นแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ด้านหน้าของเครื่องจะมีรายละเอียด ดังนี้

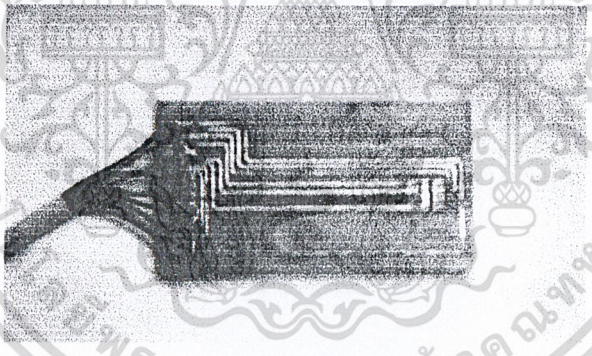
1. สวิตช์ ON / OFF ใช้สำหรับเปิดหรือปิดเครื่อง

2. ปุ่ม RESET ใช้สำหรับ RESET การทำงานของเครื่อง

3. Slot Smart card เป็นช่องที่ใช้สำหรับ เสียบบัตรสมาร์ทการ์ดที่ใช้งานอยู่จริงกับตัวเครื่องที่ต้องการจับสัญญาณการติดต่อระหว่างบัตรกับตัวเครื่อง

4.2.1.3 ด้านหลังของตัวเครื่อง

1. หน้า Contact เป็นส่วนที่ใช้สำหรับเสียบเข้าตัวเครื่องที่ต้องการจับสัญญาณการติดต่อ โดยหน้า Contact นี้ที่ปลายอีกด้านหนึ่งจะต่ออยู่กับบัตรจริงที่ใช้กับเครื่องนั้นๆ ดังนั้นหน้า Contact นี้จึงมีคุณสมบัติเหมือนกับบัตรจริงทุกประการ ส่วนสาเหตุที่ต้องมีหน้า Contact นี้ขึ้นมาเนื่องจากเพื่อต้องการให้มีจุดที่สามารถ ต่อสายเข้าไปเพื่อรับสัญญาณการติดต่อ ของบัตรกับเครื่อง



รูปที่ 4.10 หน้า Contact จำลอง

2. พอร์ต DB-9 แบบตัวเมีย ใช้สำหรับส่งข้อมูลเมื่อทำการเลือกสวิตช์ไปที่ Show Rs-232 ไปให้เครื่องคอมพิวเตอร์แสดงผลออกทางโปรแกรมแสดงผล การใช้งานต้องต่อโดยใช้สาย DB-9 ต่อกับพอร์ต DB-9 ที่ตัวเครื่องไปต่อกับพอร์ต DB–9 ที่เครื่องคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

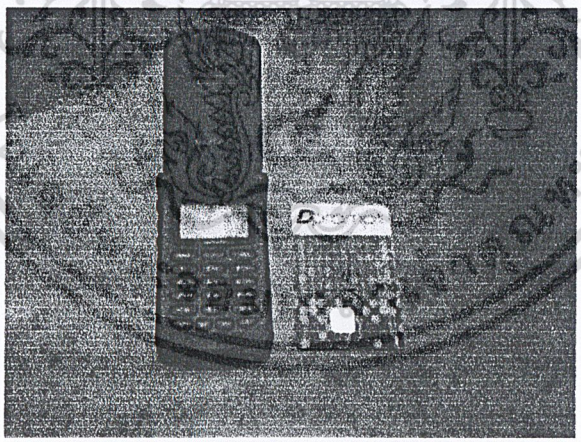
รูปที่ 4.11 สาย พอร์ต DB-9

4.2.2 ขั้นตอนการใช้งาน

การอธิบายขั้นตอนการใช้งานนี้จะแบ่งเป็นส่วนๆ เพื่อง่ายต่อการทำความเข้าใจ โดยจะแบ่งเป็นส่วนของ การจับสัญญาณ และส่วนของการแสดงผล

4.2.2.1 ขั้นตอนการจับสัญญาณ

4.2.2.1.1 ขั้นตอนการจับสัญญาณการจับสัญญาณการติดต่อ กับโทรศัพท์

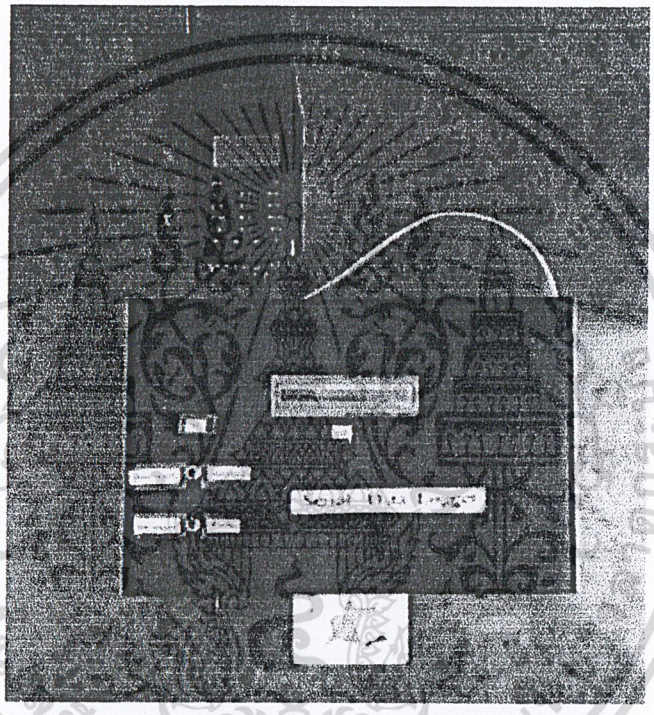


รูปที่ 4.12 โทรศัพท์ที่นำมาใช้จับสัญญาณ

1. เลือกสวิตช์ ON/OFF ไปที่ตำแหน่ง ON เพื่อเปิดเครื่อง
2. เลือกสวิตช์ Data Logger / Show Data ไปที่ตำแหน่ง Data Logger เพื่อเพื่อทำการจับสัญญาณ
3. สวิตช์ Show RS-232 / Show LCD จะอยู่ที่ตำแหน่งใดก็ได้
4. เสียบบัตรสมาร์ทการ์ดเข้าที่ช่องสล็อตสมาร์ทการ์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. กดปุ่ม RESET
6. เสียบหน้า Contact จำลองเข้ากับตัวเครื่องที่ต้องการจับสัญญาณ
7. รอจนไฟ LED หยุดกระพริบแสดงว่าจับสัญญาณเสร็จแล้ว
8. นำไปทำการแสดงผลต่อไป



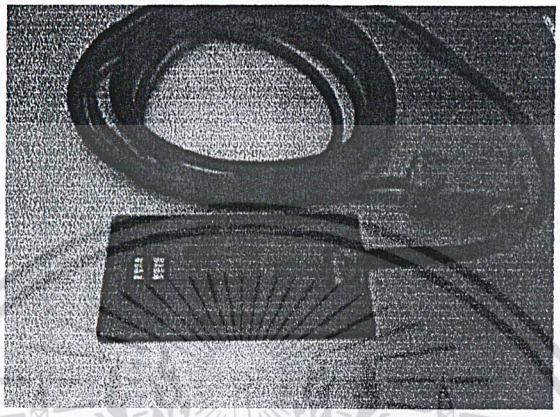
รูปที่ 4.13 แสดงการจับสัญญาณกับโทรศัพท์

4.2.2.1.2 การจับสัญญาณการติดต่อกับ Reciver (ใช้กับการ์ด Emulator เพื่อทดสอบความถูกต้อง)

1. เลือกสวิตช์ ON/OFF ไปที่ตำแหน่ง ON เพื่อเปิดเครื่อง
2. เลือกสวิตช์ Data Logger / Show Data ไปที่ตำแหน่ง Data Logger เพื่อเพื่อทำการจับสัญญาณ
3. สวิตช์ Show RS-232 / Show LCD จะอยู่ที่ตำแหน่งใดก็ได้
4. เปิดโปรแกรมการ์ด EMULATOR ดูรายละเอียดการใช้งานได้ที่ บทที่ 4 ในหัวข้อการใช้งานโปรแกรม Ipnotica

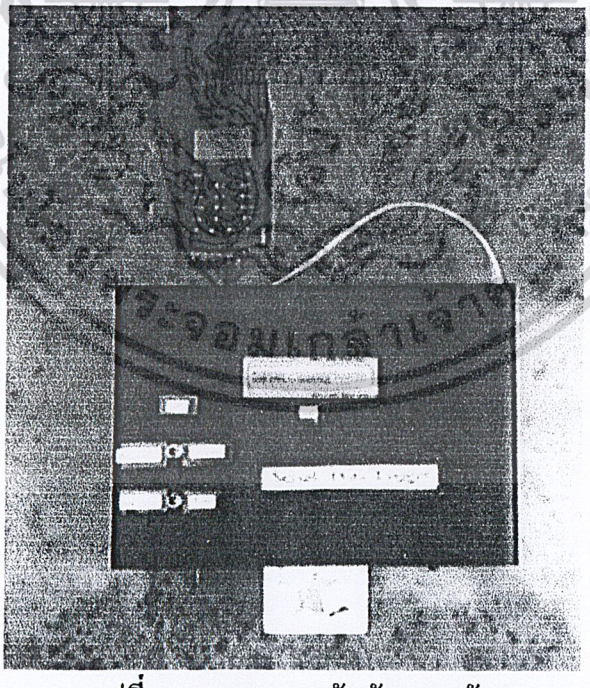
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. เสียบ Card Emulator เข้าที่ช่องสล็อตสมาร์ทการ์ด



รูปที่ 4.14 การ์ด EMLUTOR

6. กดปุ่ม RESET
7. เสียบหน้า Contact จำลองเข้ากับตัวเครื่องที่ต้องการจับสัญญาณ



รูปที่ 4.15 แสดงการจับสัญญาณกับReciver

8. รอจนไฟ LED หยุดกระพริบแสดงว่าจับสัญญาณเสร็จแล้ว

9. นำไปทำการแสดงผลต่อไป

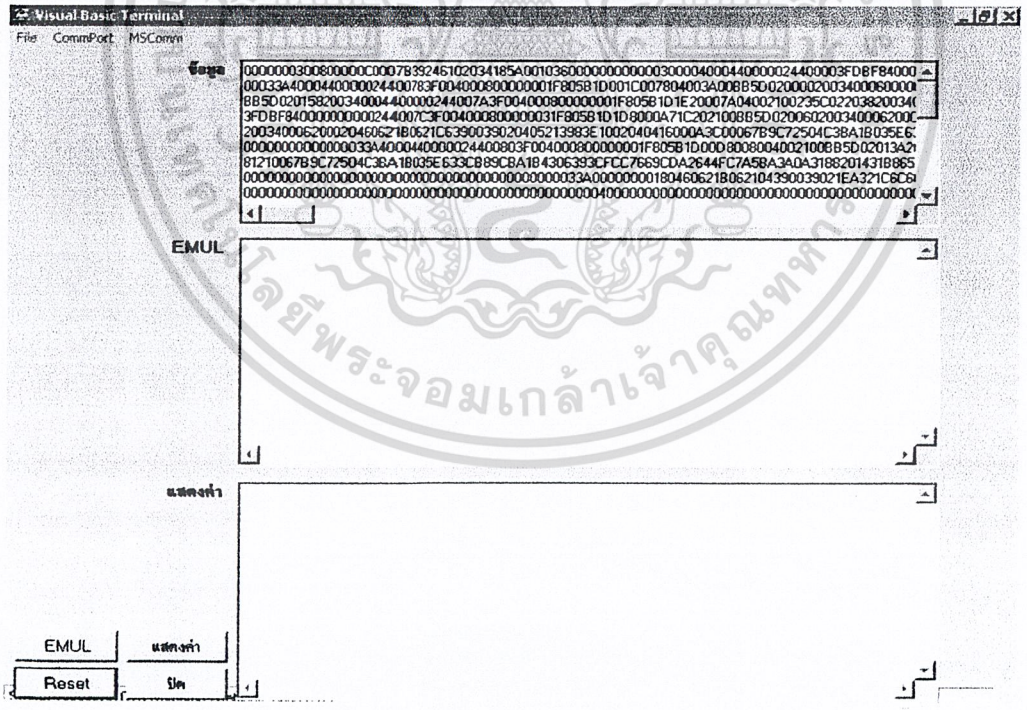
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

* หมายเหตุ หากปิดสวิตช์ไฟเลี้ยง ON/OFF โดย อยู่ที่ตำแหน่ง OFF ข้อมูลที่จัดเก็บไว้จะหายทั้งหมดเพราะใช้ RAM เป็นตัวจัดเก็บ หากต้องการให้ข้อมูลอยู่ต้อง ON ไว้ตลอดเวลา

4.2.2.2 ขั้นตอนการแสดงผล

4.2.2.2.1 การแสดงผลด้วย RS -232

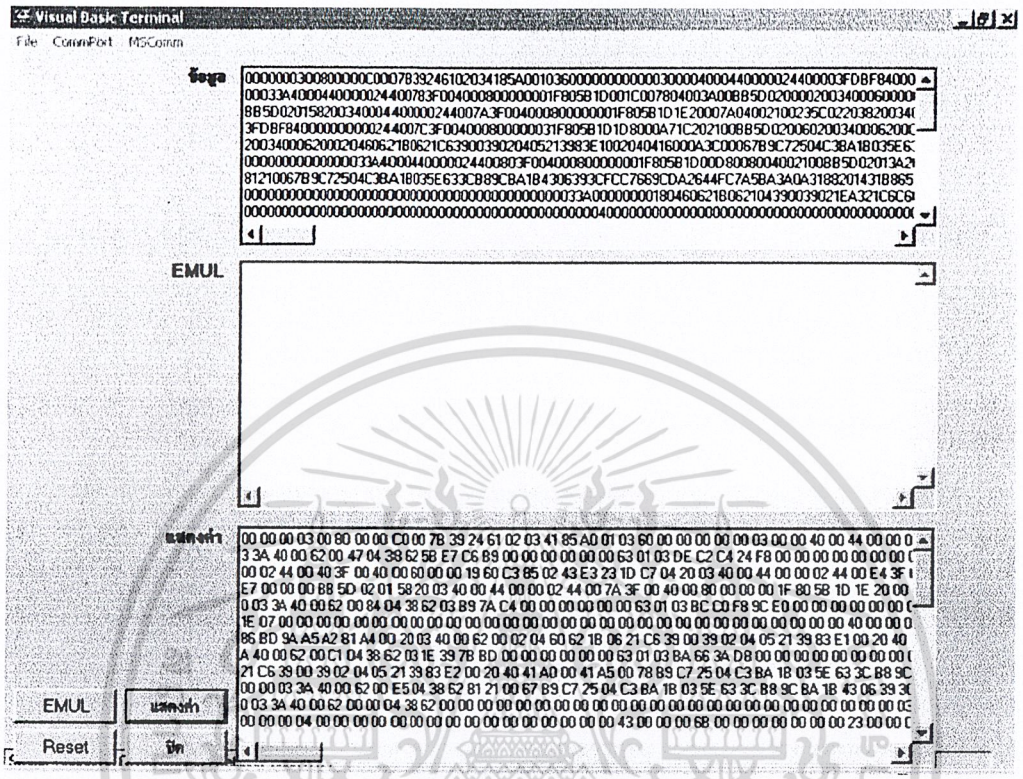
1. ต่อพอร์ตอนุกรม ที่ตัวเครื่องจับสัญญาณกับ พอร์ตอนุกรมที่เครื่องคอมพิวเตอร์
2. เปิดโปรแกรมการแสดงผล เลือกพอร์ตใช้งานของโปรแกรมให้ตรงกับพอร์ตที่ทำการต่อกับเครื่องจับสัญญาณ รายละเอียดการใช้ปุ่มต่างๆ ของโปรแกรมสามารถดูได้จากภาคผนวกโปรแกรมการแสดงผล
3. เลือกปุ่ม Show RS-232 / Show LCD ไปที่ Show RS-232
4. เลือกปุ่ม Data Logger / Display ไปที่ Display
5. ที่จอ LCD จะทำการแสดงค่าการใช้ตำแหน่งในการจัดเก็บข้อมูลลงในหน่วยความจำ
6. ที่โปรแกรมรับข้อมูลจะได้รับข้อมูลเข้ามาที่หน้าจอ



รูปที่ 4.16 แสดงการรับข้อมูลที่หน้าจอแสดงผล

7. กดปุ่มแสดงค่า เพื่อแสดงค่าที่รับมาเป็นแบบฐาน 16

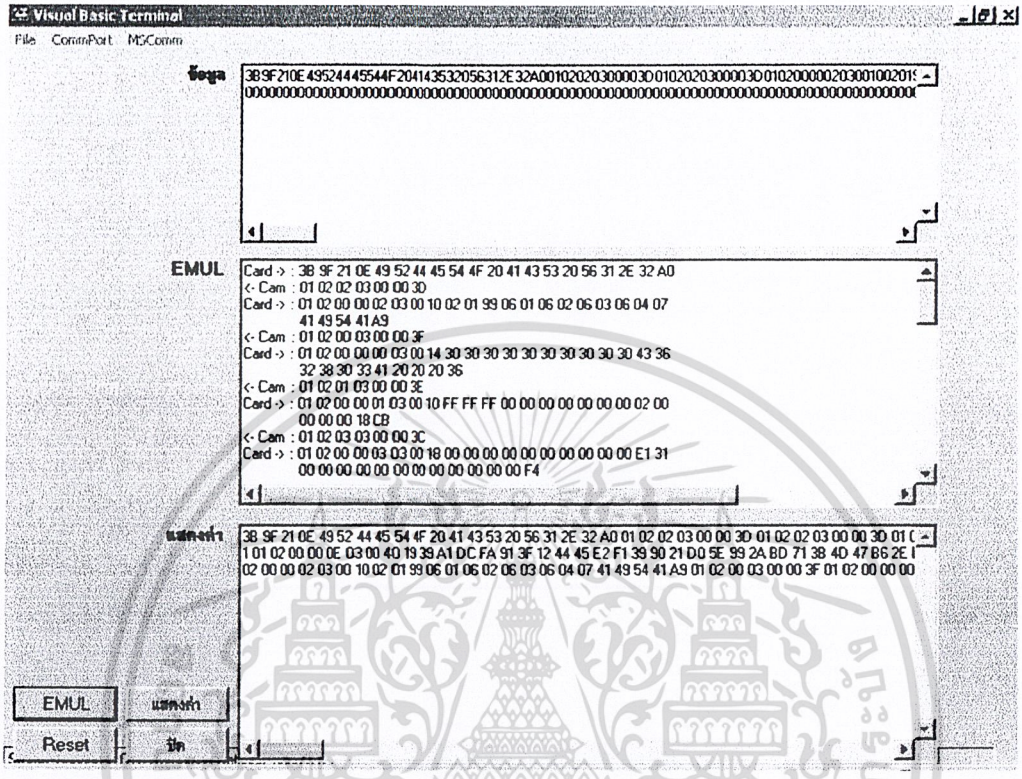
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.17 การแสดงค่าโดยกดปุ่ม แสดงค่า

8. ในกรณีที่ข้อมูลที่จับมาได้เป็นข้อมูลที่มาจากการติดต่อของ Emulator กับ Reciver จะสามารถกดปุ่ม Emulator เพื่อแสดงค่าได้ด้วย แต่ข้อมูลจากโทรศัพท์ไม่สามารถแสดงด้วยปุ่มนี้ได้ เนื่องจากมีการใช้โค๊ตที่ต่างกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.18 การแสดงค่าโดยกดปุ่ม Emulator (เฉพาะกับการ์ด Emulator เท่านั้น)

4.2.2.2.2 การแสดงผลด้วย LCD

1. เลือกสวิตช์ Show RS-232 / Show LCD ไปที่สภาวะ Show
2. LCD หากก่อนหน้านั้นทำการแสดงผลที่ RS-232 มาก็ต้องทำการกดปุ่ม RESET ด้วยหลังจากเลือกมาที่ Show LCD แล้ว
3. เครื่องจะทำการแสดงผลที่ จอ LCD เป็นเลขฐาน 16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุป

5.1.0 ผลของโครงงาน

ผลจากโครงงานที่ได้เป็นที่น่าพอใจ สามารถจับข้อมูลได้ถูกต้องเหมือนกับการจับโดยใช้ Emulator โดยมีความผิดเพี้ยนบางเงื่นน้อย อันเป็นผลเนื่องจากการเสียบการ์ดเข้าเครื่องหากทำการเสียบช้าหรือผิดพลาดจะทำให้มี Byte ข้อมูลเพิ่มขึ้นมาข้างหน้าข้อมูลจริงอันเป็นผลจากการเสียบการ์ดช้าที่ให้ขณะที่การ์ดเคลื่อนตัวเข้าอยู่ในช่วงคาบเกี่ยวของสล็อตมีพัลซ์เข้ามา ทำให้ได้รับเข้ามาโดยไม่ตั้งใจ

ในการแสดงผลที่หน้าจอการแสดงผลจะใช้ทำการแสดงผลกับข้อมูลที่มีความยาวเกินกว่าความจุของตัวแปลแบบ String ของโปรแกรม วิชวลเบสิก จะไม่สามารถทำได้แต่สามรถนำไปใช้แสดงด้วยโปรแกรม Hyper terminal แทนได้

การแสดงผลที่หน้าจอ Emulator สามารถแสดงได้แต่ข้อมูลที่ทำการจัดเก็บมาจากการติดต่อระหว่างการ์ด Emulator กับเครื่อง Receiver เท่านั้นหากต้องการแสดงข้อมูลที่เป็นการติดต่อระหว่างการ์ดทั่วๆ ไปกับเครื่องจะต้องทำการแสดงผลที่หน้าจอแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

ชัยวัฒน์ ลิ้มพรจิตรวิลัย , **คู่มือนักอิเล็กทรอนิก** : กรุงเทพ ,บริษัท ซีเอ็ดยูเคชั่น จำกัด , 2538

ชัยวัฒน์ ลิ้มพรจิตรวิลัย และ กฤษดา ใจเย็น :**เรียนรู้และปฏิบัติการไมโครคอลโทรลเลอร์ PIC 16F84** กรุงเทพ , บริษัท อินโนเวตีฟ เอ็กเพอริเมนต์ จำกัด

DATA SHEET PIC 16F87X , Microchip

ชัยวัฒน์ ลิ้มพรจิตรวิลัย และ กฤษดา ใจเย็น :**เรียนรู้และปฏิบัติการไมโครคอลโทรลเลอร์ PIC 16F628** กรุงเทพ , บริษัท อินโนเวตีฟ เอ็กเพอริเมนต์ จำกัด

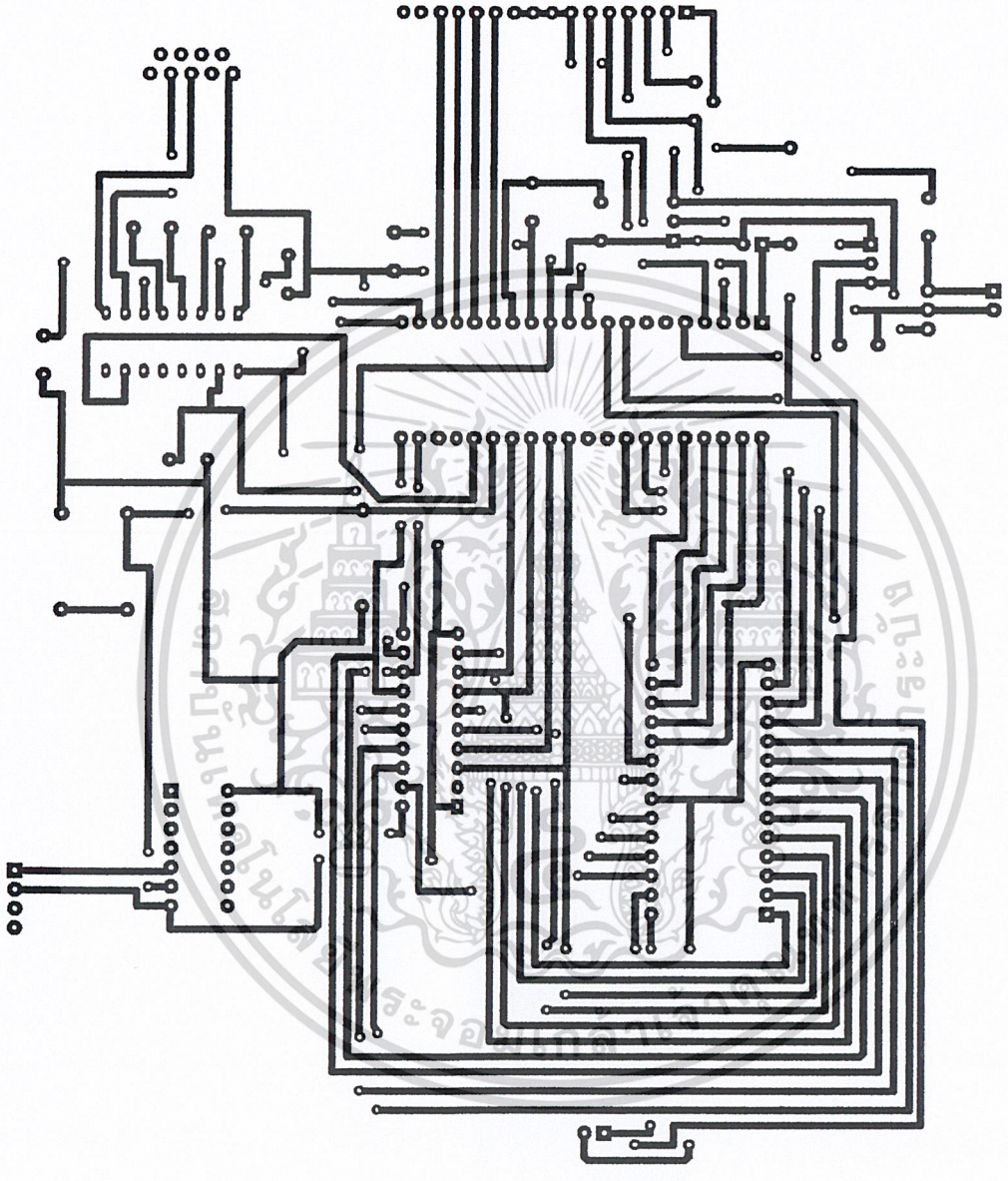
วิบูลย์ ชื่นแขก : **ไมโครโปรเซสเซอร์** กรุงเทพ ,สำนักพิมพ์สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ, 2532

เซมิคอนดักเตอร์ อิเล็กทรอนิกส์ , ฉบับที่ 241 พฤศจิกายน 2545

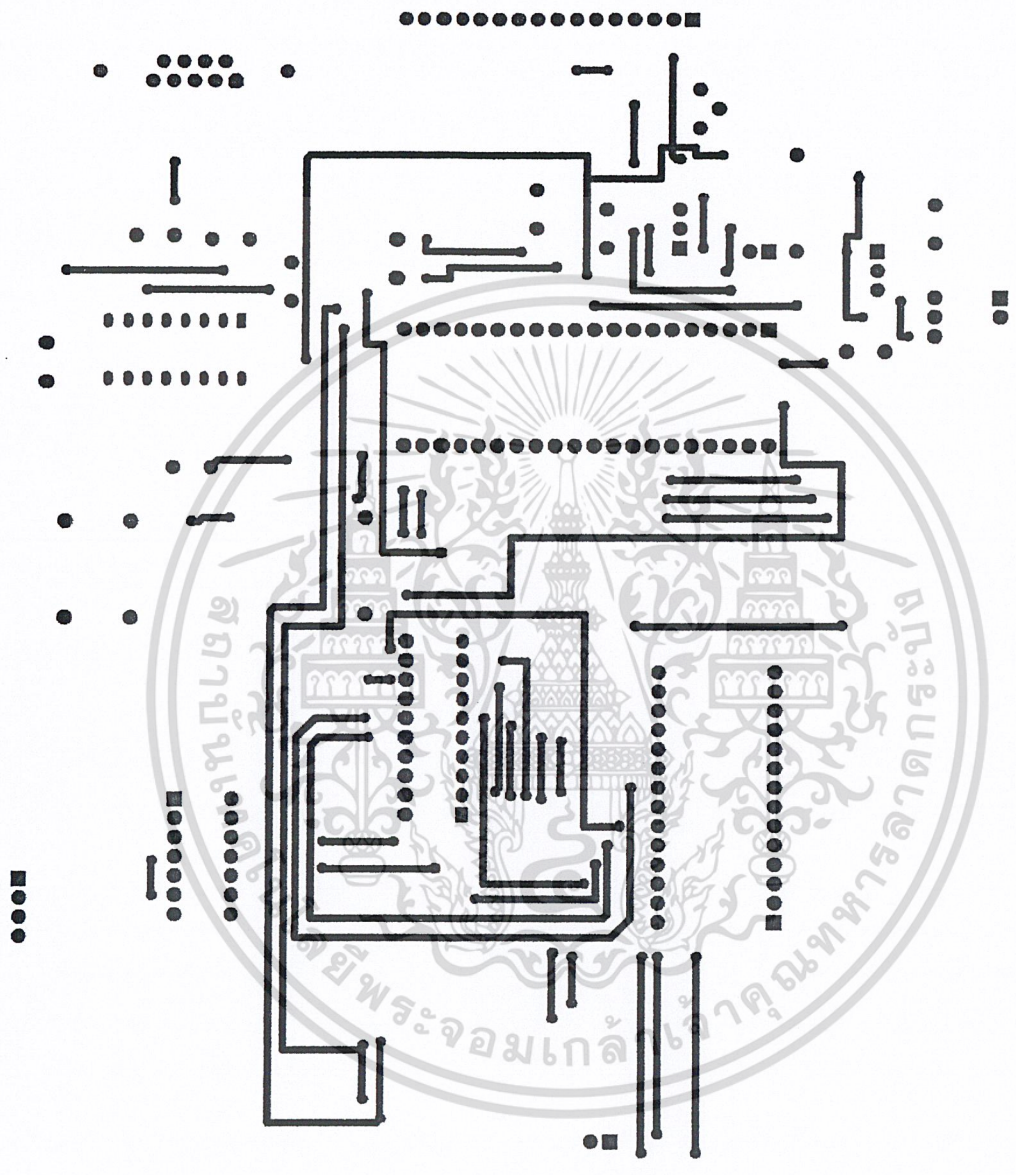
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก วงจร PCB

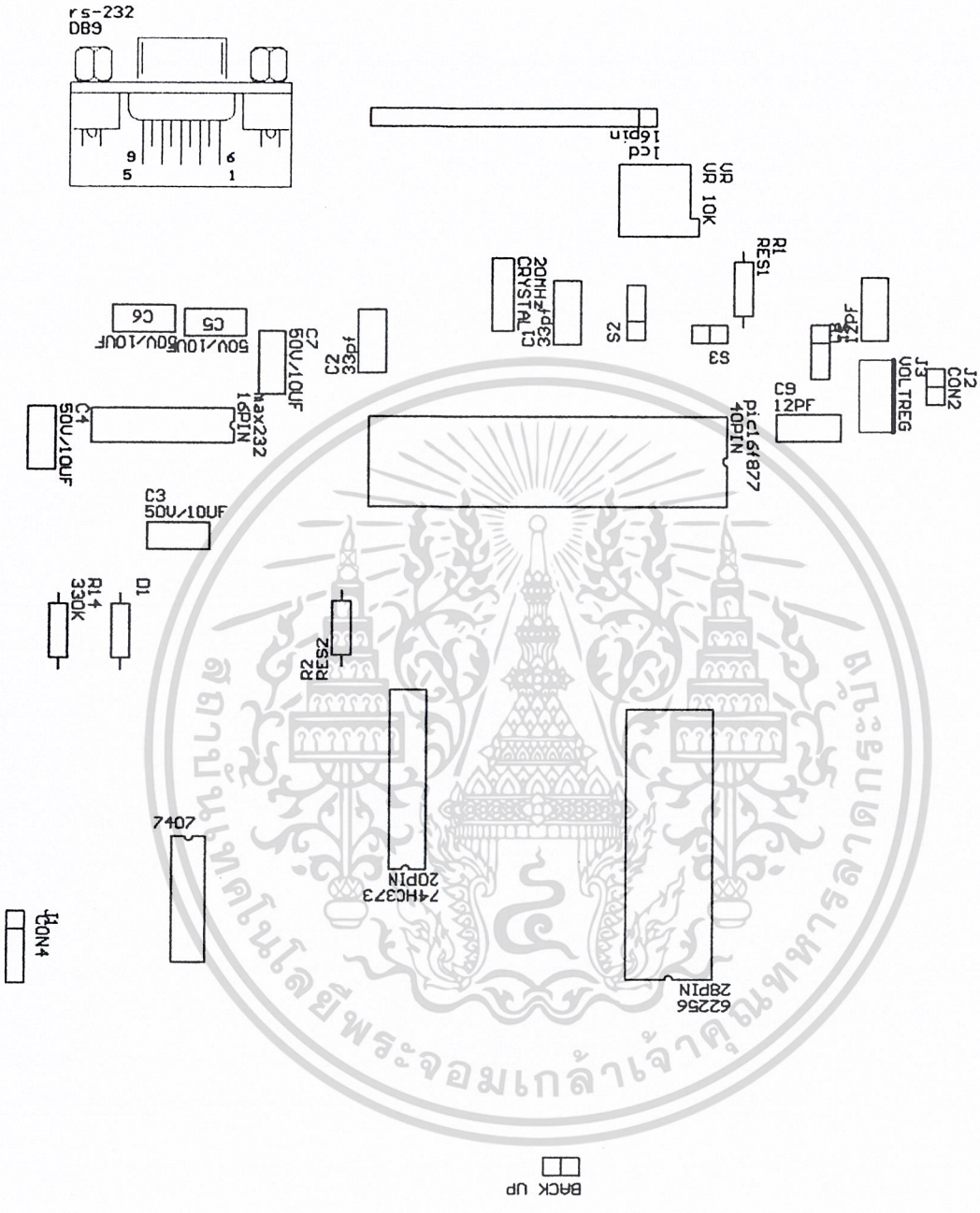
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก DATA SHEET

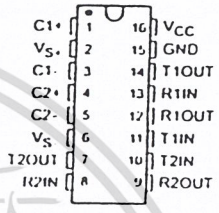
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MAX232, MAX232I
DUAL EIA-232 DRIVER/RECEIVER

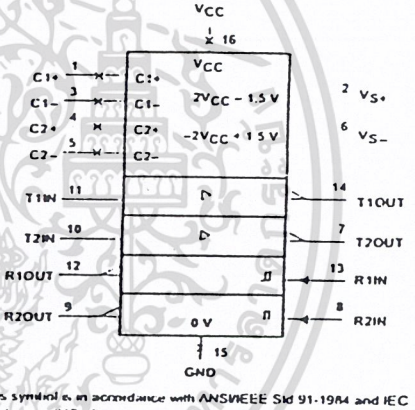
SLLS047G – FEBRUARY 1989 – REVISED AUGUST 1998

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

D, DW, OR N PACKAGE
(TOP VIEW)



logic symbol†



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

description

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

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

AVAILABLE OPTIONS

T_A	PACKAGED DEVICES		
	SMALL OUTLINE (D)	SMALL OUTLINE (DW)	PLASTIC DIP (N)
0°C to 70°C	MAX232D†	MAX232DW†	MAX232N
–40°C to 85°C	MAX232ID†	MAX232IDW†	MAX232IN

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

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

LinASIC and LinBiCMOS are trademarks of Texas Instruments Incorporated.

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

Copyright © 1998, Texas Instruments Incorporated

TEXAS INSTRUMENTS
POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SLLS047G – FEBRUARY 1989 – REVISED AUGUST 1998

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†

Input supply voltage range, V_{CC} (see Note 1) ..... −0.3 V to 6 V
Positive output supply voltage range, V_{S+} ..... V_{CC} − 0.3 V to 15 V
Negative output supply voltage range, V_{S-} ..... −0.3 V to −15 V
Input voltage range, V_I Driver ..... −0.3 V to V_{CC} + 0.3 V
Receiver ..... ±30 V
Output voltage range, V_O T1OUT, T2OUT ..... V_S −0.3 V to V_{S+} + 0.3 V
R1OUT, R2OUT ..... −0.3 V to V_{CC} + 0.3 V
Short-circuit duration: T1OUT, T2OUT ..... Unlimited
Package thermal impedance, θ_{JA} (see Note 2): D package ..... 113°C/W
DW package ..... 105°C/W
N package ..... 78°C/W
Storage temperature range, T_{stg} ..... −65°C to 150°C
Lead temperature 1,6 mm (1/16 inch) from case for 10 seconds ..... 260°C

† Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTES: 1. All voltage values are with respect to network ground terminal.
2. The package thermal impedance is calculated in accordance with JESD 51, except for through-hole packages, which use a trace length of zero.

recommended operating conditions

		MIN	NOM	MAX	UNIT
Supply voltage, V_{CC}		4.5	5	5.5	V
High-level input voltage, V_{IH} (T1IN, T2IN)		2			V
Low-level input voltage, V_{IL} (T1IN, T2IN)				0.8	V
Receiver input voltage, R1IN, R2IN				±30	V
Operating free-air temperature, T_A	MAX232	0		70	°C
	MAX232I	−40		85	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

electrical characteristics over recommended ranges of supply voltage and operating free-air temperature range (unless otherwise noted)

PARAMETER			TEST CONDITIONS		MIN	TYP†	MAX	UNIT
V_{OH}	High-level output voltage	T1OUT, T2OUT	R_L = 3 kΩ to GND		5	7		V
		R1OUT, R2OUT	I_{OH} = – 1 mA		3.5			
V_{OL}	Low-level output voltage‡	T1OUT, T2OUT	R_L = 3 kΩ to GND			–7	–5	V
		R1OUT, R2OUT	I_{OL} = 3.2 mA				0.4	
V_{IT+}	Receiver positive-going input threshold voltage	R1IN, R2IN	V_{CC} = 5 V,	T_A = 25°C		1.7	2.4	V
V_{IT-}	Receiver negative-going input threshold voltage	R1IN, R2IN	V_{CC} = 5 V,	T_A = 25°C	0.8	1.2		V
V_{hys}	Input hysteresis voltage	R1IN, R2IN	V_{CC} = 5 V		0.2	0.5	1	V
r_i	Receiver input resistance	R1IN, R2IN	V_{CC} = 5 V,	T_A = 25°C	3	5	7	kΩ
r_o	Output resistance	T1OUT, T2OUT	V_{CC} = V_+ = V_- = 0,	V_O = ± 2 V	300			Ω
I_{OS}§	Short-circuit output current	T1OUT, T2OUT	V_{CC} = 5.5 V,	V_O = 0		±10		mA
I_{IS}	Short-circuit input current	T1IN, T2IN	V_I = 0				200	µA
I_{CC}	Supply current		V_{CC} = 5.5 V, T_A = 25°C	All outputs open,		8	10	mA

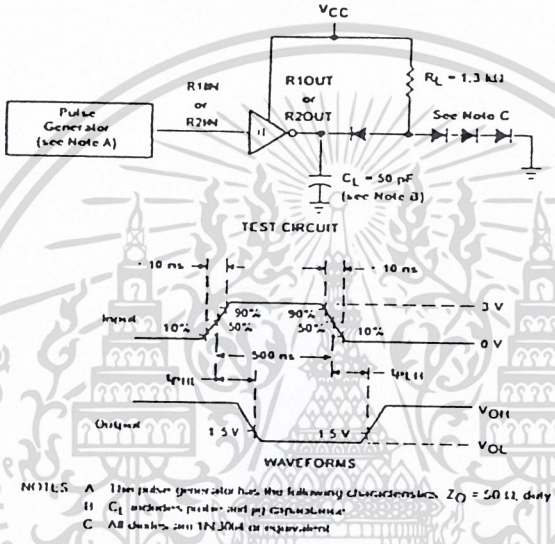
† All typical values are at V_{CC} = 5 V, T_A = 25°C.
‡ The algebraic convention, in which the least positive (most negative) value is designated minimum, is used in this data sheet for logic voltage levels only.
§ Not more than one output should be shorted at a time.

switching characteristics, V_{CC} = 5 V, T_A = 25°C

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
$t_{PLH(R)}$	Receiver propagation delay time, low- to high-level output	See Figure 1		500		ns
$t_{PHL(R)}$	Receiver propagation delay time, high- to low-level output	See Figure 1		500		ns
SR	Driver slew rate	R_L = 3 kΩ to 7 kΩ, See Figure 2			30	V/µs
SR(tr)	Driver transition region slew rate	See Figure 3		3		V/µs

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PARAMETER MEASUREMENT INFORMATION



NOTES: A. The pulse generator has the following characteristics: Z_O = 50 Ω, duty cycle = 50%.
B. C_L includes probe and jig capacitance.
C. All diodes are 1N3064 or equivalent.

Figure 1. Receiver Test Circuit and Waveforms for t_{PHL} and t_{PLH} Measurements

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PARAMETER MEASUREMENT INFORMATION

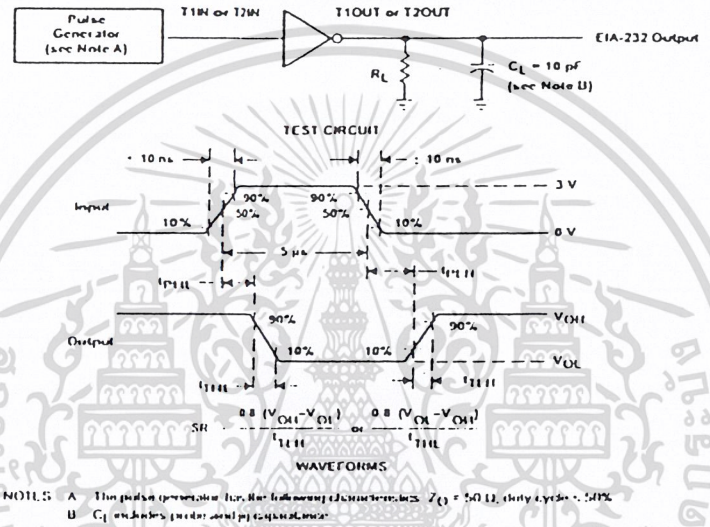


Figure 2. Driver Test Circuit and Waveforms for t_{PHL} and t_{PLH} Measurements (5-µs input)

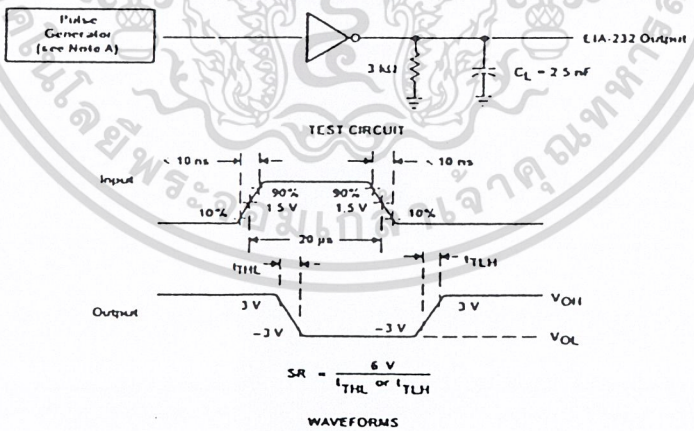


Figure 3. Test Circuit and Waveforms for t_{THL} and t_{TLH} Measurements (20-µs input)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SLLS047G – FEBRUARY 1989 – REVISED AUGUST 1998

APPLICATION INFORMATION

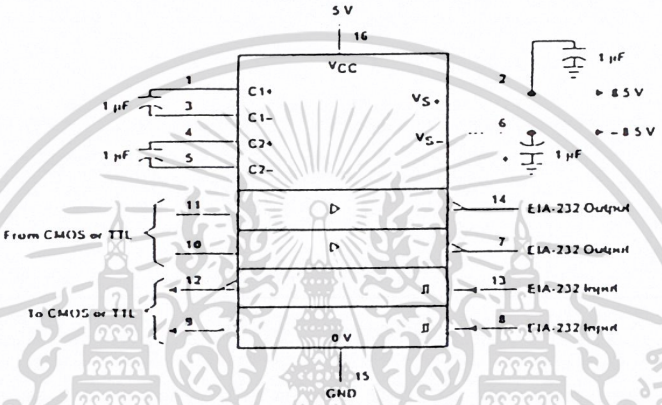
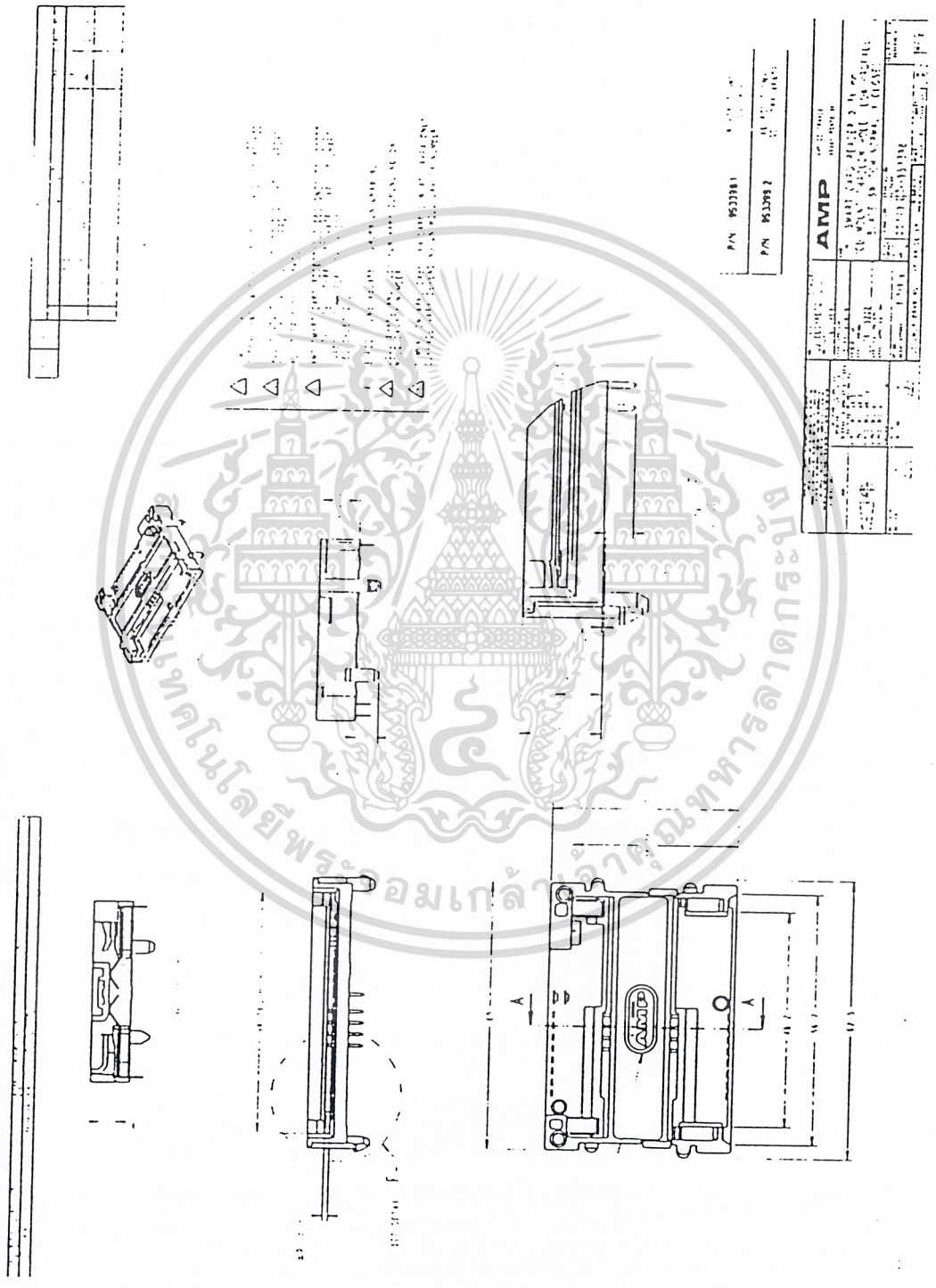
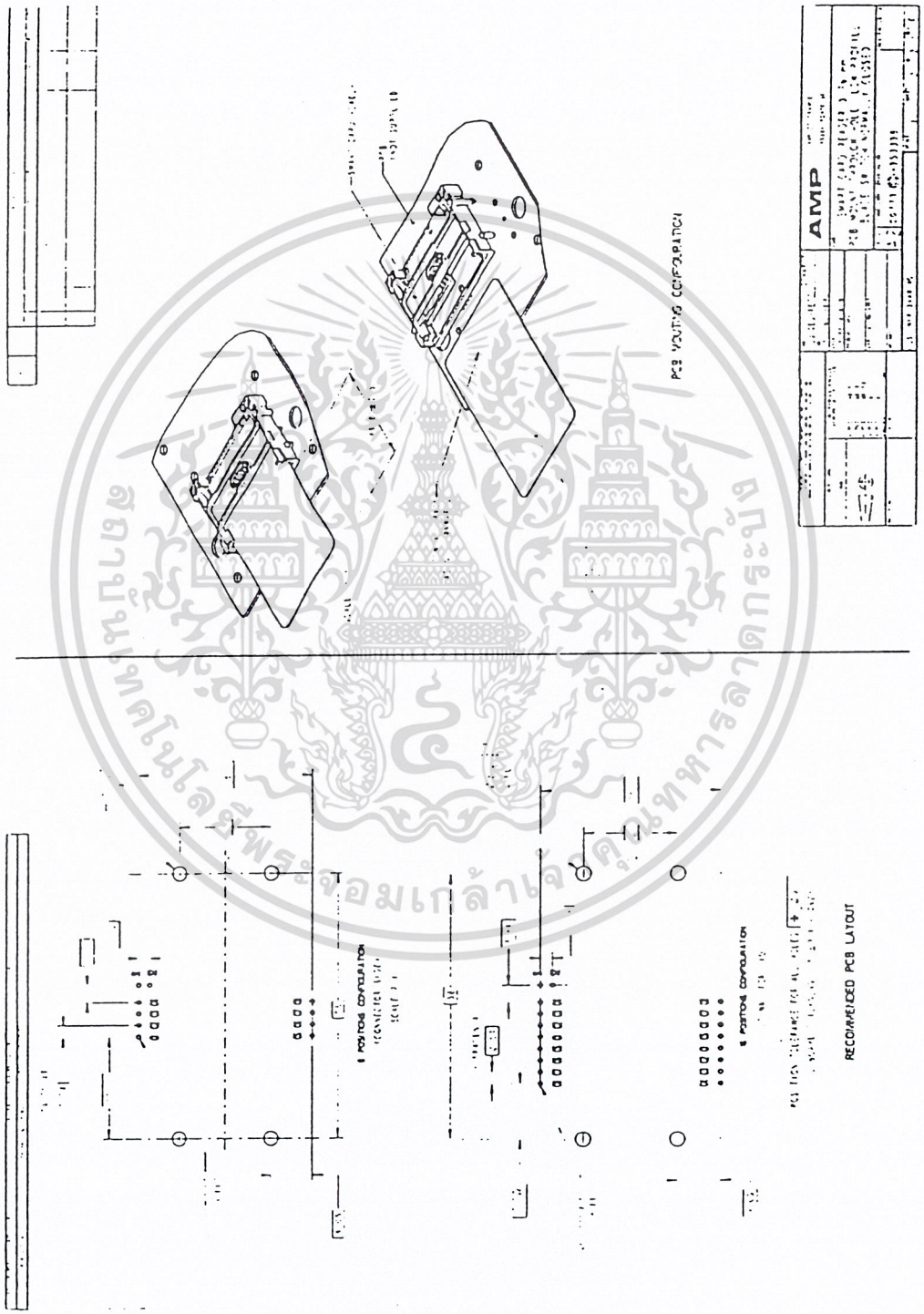


Figure 4. Typical Operating Circuit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5407/7407 Hex Buffer/Driver with Open-Collector High-Voltage Output

	Schottky TTL					High-Speed TTL					Low-Power Schottky TTL					Standard TTL					Low-Power TTL				
	Device Type	Package				Device Type	Package				Device Type	Package				Device Type	Package				Device Type	Package			
		C	P	M	CF		C	P	M	CF		C	P	M	CF		C	P	M	CF		C	P	M	CF
T.I.																SN5407	J			W					
																SN7407	J	N							
FAIRCHILD																FM5407/FM9N07	D			F					
																FC7407/FC9N07	D	P							
MOTOROLA																SN7407		P							
																DM5407	J	N		W					
N.S.C.																DM7407	J	N							
PHILIPS																S5407	F	A		W					
SIGNETICS																N7407	F	A							
SIEMENS																FLH491		J							
FUJITSU																									
HITACHI																HD7407		P							
MITSUBISHI																M7407		P							
NEC																									
TOSHIBA																TD7407		P							

Electrical Characteristics SN5407/SN7407

absolute maximum ratings over operating free-air temperature range

Supply voltage, V_{CC}	7V	Operating free-air temperature range	SN54'	−55°C to 125°C
Input voltage	5.5V		SN74'	0°C to 70°C
Off-state (high-level) voltage applied to open-collector outputs	30 V	Storage temperature range		−65°C to 150°C

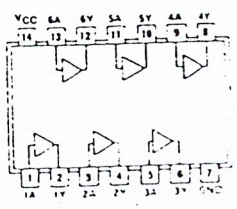
recommended operating conditions

	SN5407			SN7407			UNIT
	MIN	NOM	MAX	MIN	NOM	MAX	
Supply voltage, V_{CC}	4.5	5	5.5	4.75	5	5.25	V
High-level output voltage, V_{OH}			30			30	V
Low-level output current, I_{OL}			30			40	mA
Operating free-air temperature, T_A	−55		125	0		70	°C

electrical characteristics over recommended operating free-air temperature range

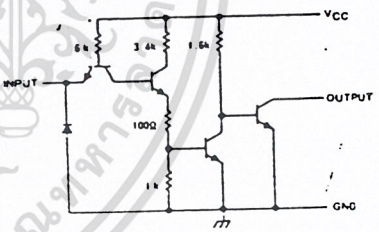
PARAMETER		TEST CONDITIONS†		MIN	TYP‡	MAX	UNIT
V_{IH}	High-level input voltage			2			V
V_{IL}	Low-level input voltage					0.8	V
V_I	Input clamp voltage	V_{CC}=MIN, I_I=−12mA				−1.5	V
I_{OH}	High-level output current	V_{CC}=MIN, V_{IH}=2V, V_{OH}=MAX				250	μA
V_{OL}	Low-level output voltage	V_{CC}=MIN, V_{IL}=V_{IL} max, I_{OL}=16mA				0.4	V
		V_{CC}=MIN, V_{IL}=V_{IL} max, I_{OL}=MAX				0.7	
I_I	Input current at maximum input voltage	V_{CC}=MAX, V_I=5.5V				1	mA
I_{IH}	High-level input current	V_{CC}=MAX, V_{IH}=2.4V				40	μA
I_{IL}	Low-level input current	V_{CC}=MAX, V_{IL}=0.4V				−1.6	mA
I_{CCH}	Supply current	V_{CC}=MAX	Total, outputs high		29	41	mA
I_{CCL}	Supply current		Total, outputs low		21	30	mA
I_{CC}	Supply current	V_{CC}=5V	Average per gate (50% duty cycle)		4.17		mA
t_{PLH}	Propagation delay time, low-to-high-level output	V_{CC}=5V, T_A=25°C, C_L=15pF, R_L=110Ω			6	10	ns
t_{PHL}	Propagation delay time, high-to-low-level output				20	30	ns

Pin Assignment (Top View)



positive logic:
$Y = \overline{A}$

schematic (each gate)



'07 CIRCUIT

Resistor values shown are nominal and in ohms.

† For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating conditions.
‡ All typical values are at V_{CC}=5V, T_A=25°C.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

54373/74373 Octal D-Type Transparent Latches and Edge-Triggered Flip-Flops

	Schottky TTL					High-Speed TTL					Low-Power Schottky TTL					Standard TTL					Low-Power TTL				
	Device Type	Package				Device Type	Package				Device Type	Package				Device Type	Package				Device Type	Package			
		C	P	M	CF		C	P	M	CF		C	P	M	CF		C	P	M	CF		C	P	M	CF
T.I.	SN54S373	J Ⓦ									SN54LS373	J Ⓦ													
	SN74S373	J Ⓝ	N								SN74LS373	J Ⓝ	N												
FAIRCHILD																									
MOTOROLA																									
N.S.C.																									
PHILIPS																									
SIGNETICS																									
SIEMENS																									
FUJITSU																									
HITACHI																									
MITSUBISHI																									
NEC																									
TOSHIBA																									

Electrical Characteristics SN54LS373/SN74LS373

absolute maximum ratings over operating free-air temperature range

Supply voltage, V_{CC}	7V	Operating free-air temperature	SN54LS	−55°C to 125°C
Input voltage	7V		SN74LS	0°C to 70°C
		Storage temperature range		−65°C to 150°C

recommend operating conditions

		SN54LS373			SN74LS373			UNIT
		MIN	NOM	MAX	MIN	NOM	MAX	
Supply voltage, V_{CC}		4.5	5	5.5	4.75	5	5.25	V
High-level output current, I_{OH}				−1			−2.6	mA
High-level output voltage, V_{OH}				5.5			5.5	V
Pulse width, t_w	Clock enable high	15			15			ns
	Clock enable high	15			15			
Setup time, t_{SUTUP}		0↓			0↓			ns
Hold time, t_{hold}		10↓			10↓			ns
Operating free-air temperature, T_A		−55		125	0		70	°C

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER		TEST CONDITIONS‡		MIN	TYP‡	MAX	UNIT
V_{IH}	High-level input voltage			2			V
V_{IL}	Low-level input voltage					0.8	V
V_{IK}	Input clamp voltage	V_{CC} = MIN, I_I = −18mA				−1.5	V
V_{OH}	High-level output voltage	V_{CC} = MIN, V_{IH} = 2V, V_{IL} = V_{IL} max, I_{OH} = MAX		2.4	3.1		V
V_{OL}	Low-level output voltage	V_{CC} = MIN, V_{IH} = 2V, V_{IL} = V_{IL} max	I_{OL} = 24mA		0.35	0.5	V
I_{OZH}	Off-state output current, high-level voltage applied	V_{CC} = MAX, V_{IH} = 2V, V_O = 2.7V				20	μA
I_{OZL}	Off-state output current, low-level voltage applied	V_{CC} = MAX, V_{IH} = 2V, V_O = 0.4V				−20	μA
I_I	Input current at maximum input voltage	V_{CC} = MAX, V_I = 7V				0.1	mA
I_{IH}	High-level input current	V_{CC} = MAX, V_I = 2.7V				20	μA
I_{IL}	Low-level input current	V_{CC} = MAX, V_I = 0.4V				−0.4	mA
I_{OS}	Short-circuit output current §	V_{CC} = MAX		−30		−130	mA
I_{CC}	Supply current	V_{CC} = MAX, Output control at 4.5V	'LS372		24	40	mA

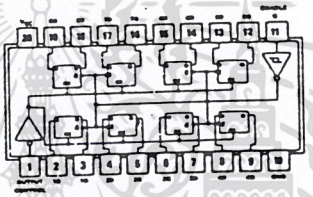
switching characteristics, V_{CC} = 5V, T_A = 25°C

PARAMETER	FROM (INPUT)	TO (OUTPUT)	TEST CONDITIONS	MIN	TYP	MAX	UNIT
f_{max}			C_L = 45pF, R_L = 667Ω, See Notes 2 and 3				MHz
t_{PLH}	Data	Any Q			12	18	ns
t_{PHL}					12	18	
t_{PLH}	Clock or enable	Any Q			20	30	ns
t_{PHL}					18	30	
t_{PZH}	Output Control	Any Q			15	28	ns
t_{PZL}					25	36	
t_{PHZ}	Output Control	Any Q	C_L = 5pF, R_L = 667Ω, See Note 3		12	20	ns
t_{PLZ}					15	25	

† For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating conditions.
‡ All typical values are at V_{CC} = 5V, T_A = 25°C.
§ Not more than one output should be shorted at a time, and duration of the short circuit should not exceed one second.

NOTES 2. Maximum clock frequency is tested with all outputs loaded.
3. See load circuits and waveforms on page 3-11.
f_{max} = maximum clock frequency
t_{PLH} = propagation delay time, low-to-high-level output
t_{PHL} = propagation delay time, high-to-low-level output
t_{PZH} = output enable time to high level
t_{PZL} = output enable time to low level
t_{PHZ} = output disable time from high level
t_{PLZ} = output disable time from low level

Pin Assignments (Top View)



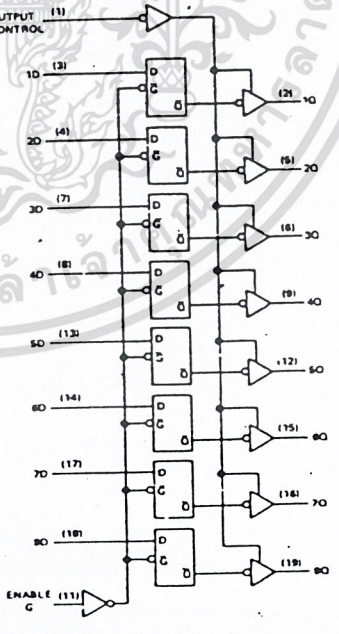
SN54LS373 (J)
SN54S373 (J)

SN74LS373 (J, N)
SN74S373 (J, N)

'LS373, 'S373
FUNCTION TABLE

OUTPUT CONTROL	ENABLE G	D	OUTPUT
L	H	H	H
L	H	L	L
L	L	X	Q_0
H	X	X	Z

'LS373, 'S373
TRANSPARENT LATCHES



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

HM62256B Series

32,768-word × 8-bit High Speed CMOS Static RAM

HITACHI

ADE-203-135D (Z)
Rev. 4.0
Nov. 29, 1995

Description

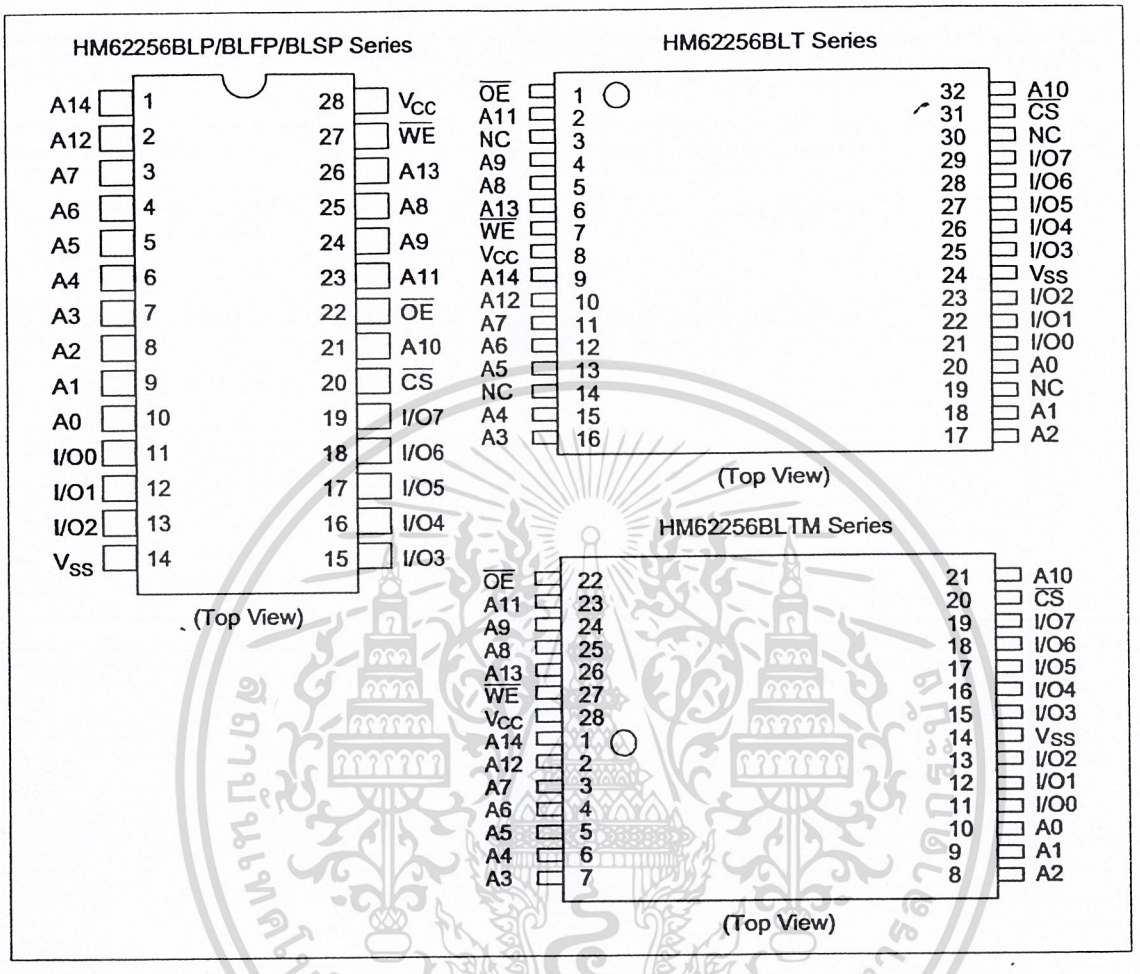
The Hitachi HM62256B is a CMOS static RAM organized 32-kword × 8-bit. It realizes higher performance and low power consumption by employing 0.8 µm Hi-CMOS process technology. The device, packaged in 8 × 14 mm TSOP, 8 × 13.4 mm TSOP with thickness of 1.2 mm, 450-mil SOP (foot print pitch width), 600-mil plastic DIP, or 300-mil plastic DIP, is available for high density mounting. It offers low power standby power dissipation; therefore, it is suitable for battery back-up systems.

Features

- High speed
  Fast access time: 45/55/70/85 ns (max)
- Low power
  Standby: 1.0 µW (typ)
  Operation: 25 mW (typ) (f = 1 MHz)
- Single 5 V supply
- Completely static memory
  No clock or timing strobe required
- Equal access and cycle times
- Common data input and output
  Three state output
- Directly TTL compatible
  All inputs and outputs
- Capability of battery back up operation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Arrangement

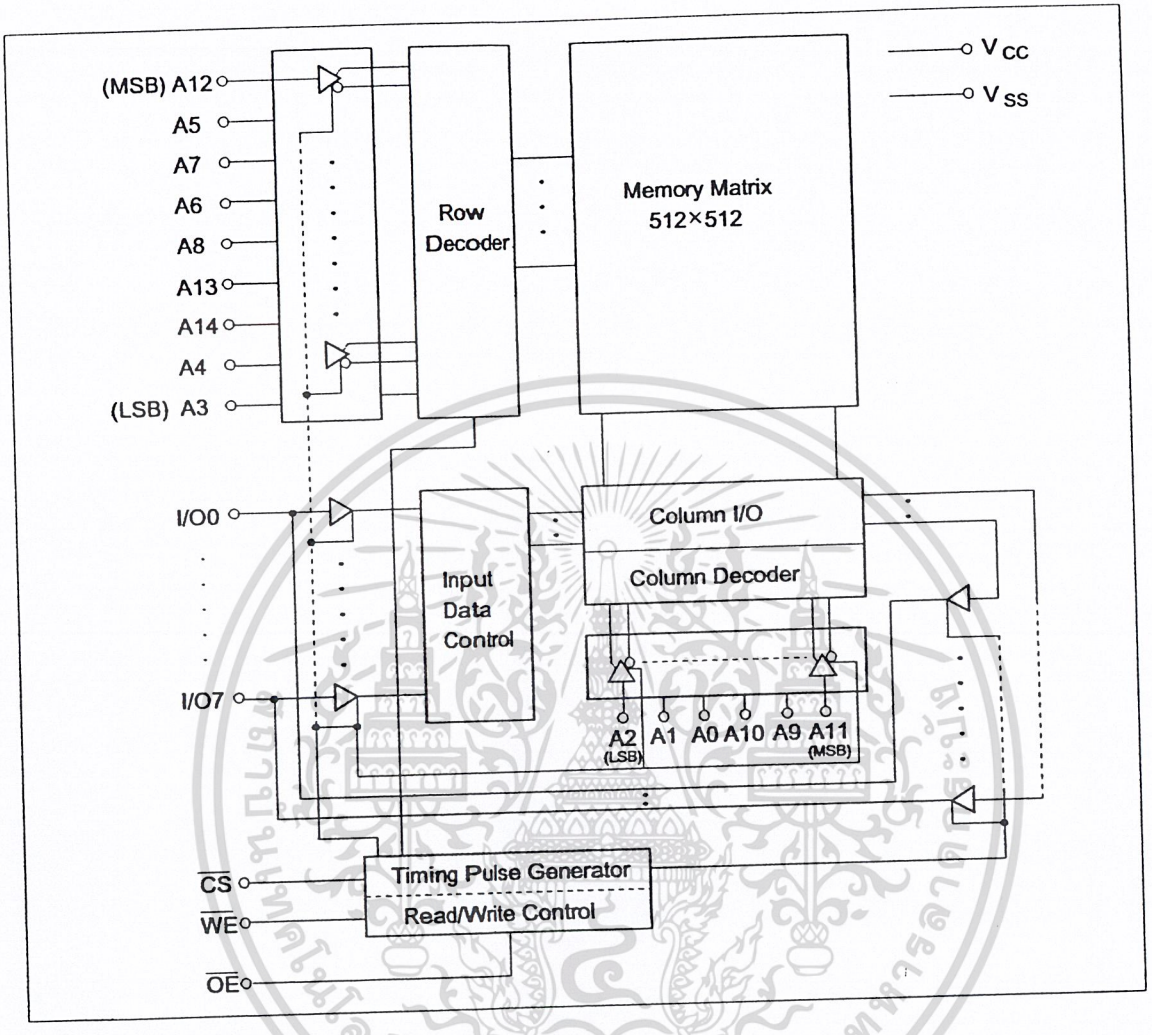


Pin Description

Symbol	Function
A0 – A14	Address
I/O0 – I/O7	Input/output
$\overline{CS}$	Chip select
$\overline{WE}$	Write enable
$\overline{OE}$	Output enable
NC	No connection
V_{cc}	Power supply
V_{ss}	Ground

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Block Diagram



Function Table

$\overline{WE}$	$\overline{CS}$	$\overline{OE}$	Mode	V_{cc} Current	I/O Pin	Ref. Cycle
X	H	X	Not selected	I_{SB}, I_{SB1}	High-Z	—
H	L	H	Output disable	I_{cc}	High-Z	—
H	L	L	Read	I_{cc}	Dout	Read cycle (1)–(3)
L	L	H	Write	I_{cc}	Din	Write cycle (1)
L	L	L	Write	I_{cc}	Din	Write cycle (2)

Note: X: H or L

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DC Characteristics (Ta = 0 to +70°C, V_{CC} = 5 V ±10%, V_{SS} = 0 V)

Parameter		Symbol	Min	Typ*1	Max	Unit	Test Conditions
Input leakage current		$\|I_{LI}\|$	—	—	1	µA	Vin = V_{SS} to V_{CC}
Output leakage current		$\|I_{LO}\|$	—	—	1	µA	$\overline{CS}$ = V_{IH} or $\overline{OE}$ = V_{IH} or $\overline{WE}$ = V_{IL}, $V_{SS} \le V_{I/O} \le V_{CC}$
Operating power supply current		I_{CC}	—	6	15	mA	$\overline{CS}$ = V_{IL}, others = V_{IH}/V_{IL}, $I_{I/O}$ = 0 mA
Average operating power supply current	HM62256B-4	I_{CC1}	—	—	70	mA	min cycle, duty = 100 %, $I_{I/O}$ = 0 mA, $\overline{CS}$ = V_{IL}, others = V_{IH}/V_{IL}
	HM62256B-5	I_{CC1}	—	—	60		
	HM62256B-7	I_{CC1}	—	33	60		
	HM62256B-8	I_{CC1}	—	29	50		
Standby power supply current		I_{CC2}	—	5	15	mA	Cycle time = 1 µs, $I_{I/O}$ = 0 mA, $\overline{CS}$ = V_{IL}, V_{IH} = V_{CC}, V_{IL} = 0
		I_{SB}	—	0.3	2	mA	$\overline{CS}$ = V_{IH}
		I_{SB1}	—	0.2	100	µA	Vin ≥ 0 V, $\overline{CS}$ ≥ V_{CC} – 0.2 V,
			—	0.2*2	50*2		
				0.2*3	10*3		
Output low voltage		V_{OL}	—	—	0.4	V	I_{OL} = 2.1 mA
Output high voltage		V_{OH}	2.4	—	—	V	I_{OH} = –1.0 mA

Notes: 1. Typical values are at V_{CC} = 5.0 V, Ta = +25°C and not guaranteed.
2. This characteristics is guaranteed only for L-SL version.
3. This characteristics is guaranteed only for L-UL version.

Capacitance (Ta = 25°C, f = 1.0 MHz)*1

Parameter	Symbol	Min	Typ	Max	Unit	Test Conditions
Input capacitance*1	Cin	—	—	8	pF	Vin = 0 V
Input/output capacitance*1	$C_{I/O}$	—	—	10	pF	$V_{I/O}$ = 0 V

Note: 1. This parameter is sampled and not 100% tested.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AC Characteristics (Ta = 0 to +70°C, V_{CC} = 5 V ± 10%, unless otherwise noted.)

Test Conditions

- Input pulse levels: 0.8 V to 2.4 V
- Input rise and fall times: 5 ns
- Input and output timing reference level: 1.5 V
- Output load: HM62256B-4: 1 TTL Gate + C_L (30 pF)(Including scope & jig)
  HM62256B-5: 1 TTL Gate + C_L (50 pF)(Including scope & jig)
  HM62256B-7/8: 1 TTL Gate + C_L (100 pF)(Including scope & jig)

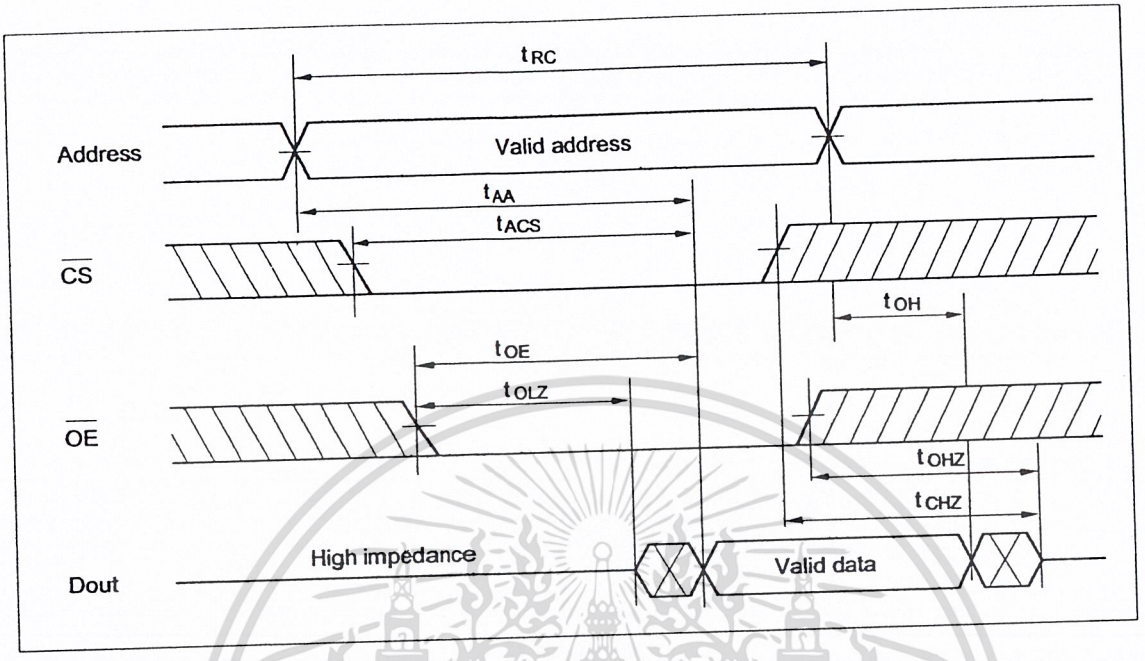
Read Cycle

		HM62256B									
		-4		-5		-7		-8			
Parameter	Symbol	Min	Max	Min	Max	Min	Max	Min	Max	Unit	Notes
Read cycle time	t_{RC}	45	—	55	—	70	—	85	—	ns	
Address access time	t_{AA}	—	45	—	55	—	70	—	85	ns	
Chip select access time	t_{ACS}	—	45	—	55	—	70	—	85	ns	
Output enable to output valid	t_{OE}		30	—	35	—	40	—	45	ns	
Chip selection to output in low-Z	t_{CLZ}	5	—	5	—	10	—	10	—	ns	2
Output enable to output in low-Z	t_{OLZ}	5	—	5	—	5	—	5	—	ns	2
Chip deselection in to output in high-Z	t_{CHZ}	0	20	0	20	0	25	0	30	ns	1, 2
Output disable to output in high-Z	t_{OHZ}	0	20	0	20	0	25	0	30	ns	1, 2
Output hold from address change	t_{OH}	5	—	5	—	5	—	10	—	ns	

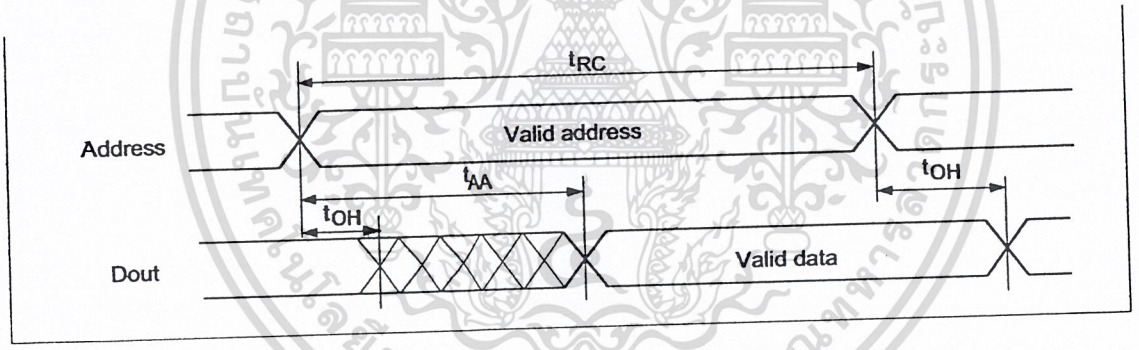
Notes: 1. t_{CHZ} and t_{OHZ} defined as the time at which the outputs achieve the open circuit conditions and are not referred to output voltage levels.
2. This parameter is sampled and not 100% tested.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Read Timing Waveform (1) ($\overline{WE}$=V_{IH})

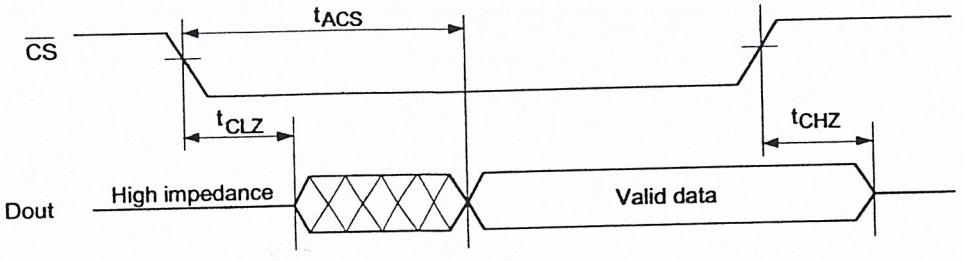


Read Timing Waveform (2) ($\overline{WE}$=V_{IH}, $\overline{CS}$=V_{IL}, $\overline{OE}$=V_{IL})



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Read Timing Waveform (3) ($\overline{WE}$=V_{IH}, $\overline{OE}$=V_{IL})*1



Note: 1. Address must be valid prior to or simultaneously with $\overline{CS}$ going low.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

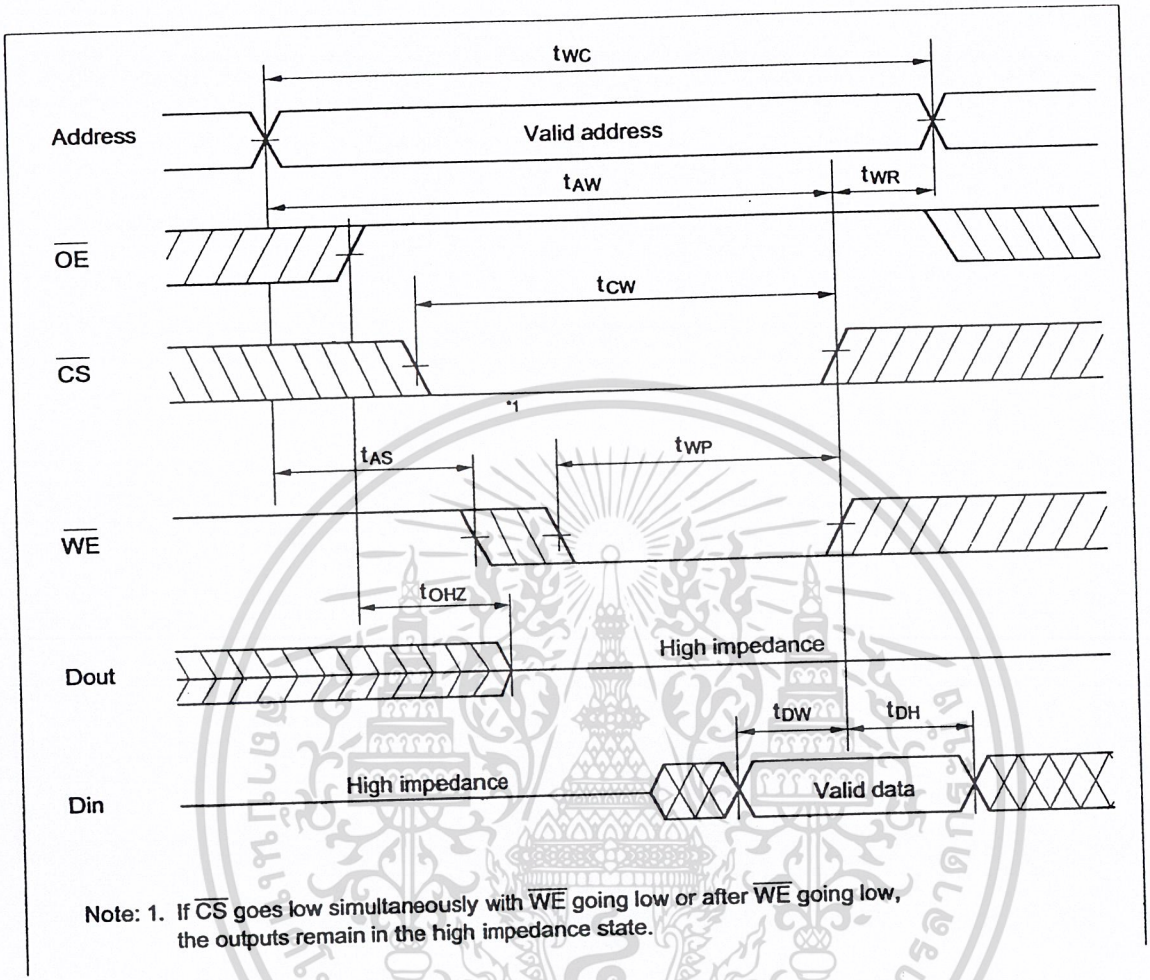
Write Cycle

		HM62256B									
		-4		-5		-7		-8			
Parameter	Symbol	Min	Max	Min	Max	Min	Max	Min	Max	Unit	Notes
Write cycle time	t_{WC}	45	—	55	—	70	—	85	—	ns	
Chip selection to end of write	t_{CW}	35	—	40	—	60	—	75	—	ns	4
Address setup time	t_{AS}	0	—	0	—	0	—	0	—	ns	5
Address valid to end of write	t_{AW}	35	—	40	—	60	—	75	—	ns	
Write pulse width	t_{WP}	30	—	35	—	50	—	55	—	ns	3, 8
Write recovery time	t_{WR}	0	—	0	—	0	—	0	—	ns	6
$\overline{WE}$ to output in high-Z	t_{WHZ}	0	20	0	20	0	25	0	40	ns	1, 2, 7
Data to write time overlap	t_{DW}	20	—	25	—	30	—	35	—	ns	
Data hold from write time	t_{DH}	0	—	0	—	0	—	0	—	ns	
Output active from end of write	t_{OW}	5	—	5	—	5	—	5	—	ns	2
Output disable to output in high-Z	t_{OHZ}	0	20	0	20	0	25	0	40	ns	1, 2, 7

Notes:
1. t_{OHZ} and t_{WHZ} are defined as the time at which the outputs achieve the open circuit conditions and are not referred to output voltage levels.
2. This parameter is sampled and not 100% tested.
3. A write occurs during the overlap (t_{WP}) of a low $\overline{CS}$ and a low $\overline{WE}$. A write begins at the later transition of $\overline{CS}$ going low or $\overline{WE}$ going low. A write ends at the earlier transition of $\overline{CS}$ going high or $\overline{WE}$ going high. t_{WP} is measured from the beginning of write to the end of write.
4. t_{CW} is measured from $\overline{CS}$ going low to the end of write.
5. t_{AS} is measured from the address valid to the beginning of write.
6. t_{WR} is measured from the earlier of $\overline{WE}$ or $\overline{CS}$ going high to the end of write cycle.
7. During this period, I/O pins are in the output state so that the input signals of the opposite phase to the outputs must not be applied.
8. In the write cycle with $\overline{OE}$ low fixed, t_{WP} must satisfy the following equation to avoid a problem of data bus contention, $t_{WP} \geq t_{WHZ}$ max + t_{OW} min.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

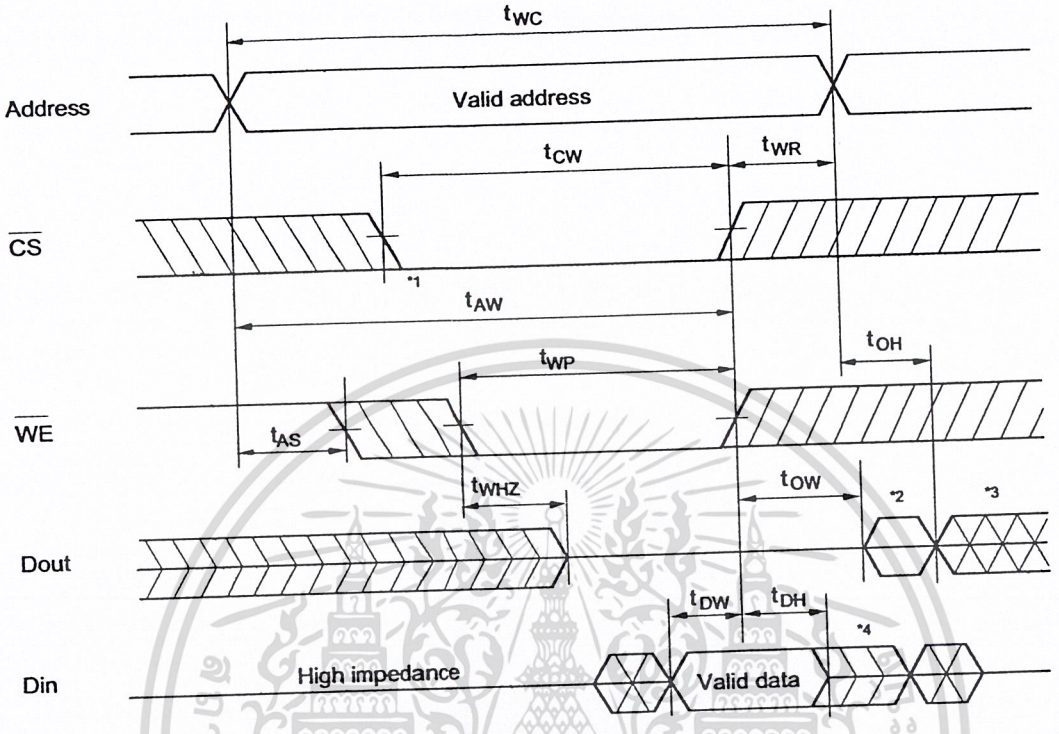
Write Timing Waveform (1) ($\overline{OE}$ Clock)



Note: 1. If $\overline{CS}$ goes low simultaneously with $\overline{WE}$ going low or after $\overline{WE}$ going low, the outputs remain in the high impedance state.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Write Timing Waveform (2) ($\overline{OE}$ Low Fixed) ($\overline{OE}$ = V_{IL})



Notes: 1. If $\overline{CS}$ goes low simultaneously with $\overline{WE}$ going low or after $\overline{WE}$ going low, the outputs remain in the high impedance state.
2. Dout is the same phase of the write data of this write cycle.
3. Dout is the read data of next address.
4. If $\overline{CS}$ is low during this period, I/O pins are in the output state. Therefore, the input signals of the opposite phase to the output must not be applied to them.

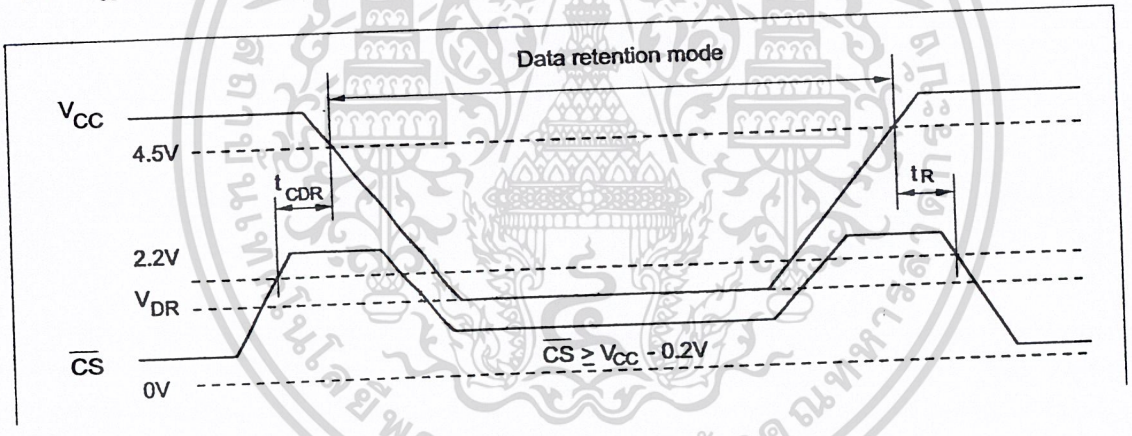
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Low V_{CC} Data Retention Characteristics (Ta = 0 to +70°C)

Parameter	Symbol	Min	Typ*1	Max	Unit	Test Conditions*6
V_{CC} for data retention	V_{DR}	2.0	—	5.5	V	$\overline{CS} \geq V_{CC} - 0.2$ V, Vin ≥ 0 V
Data retention current	I_{CCDR}	—	0.05	30*2	µA	V_{CC} = 3.0 V, Vin ≥ 0 V
		—	0.05	10*3		$\overline{CS} \geq V_{CC} - 0.2$ V,
		—	0.05	3*4		
Chip deselect to data retention time	t_{CDR}	0	—	—	ns	See retention waveform
Operation recovery time	t_R	t_{RC}*5	—	—	ns	

Notes: 1. Typical values are at V_{CC} = 3.0 V, Ta = 25°C and not guaranteed.
2. 10 µA max at Ta = 0 to + 40°C.
3. This characteristics guaranteed for only L-SL version. 3 µA max at Ta = 0 to +40°C.
4. This characteristics guaranteed for only L-UL version. 0.6 µA max at Ta = 0 to +40°C.
5. t_{RC} = read cycle time.
6. $\overline{CS}$ controls address buffer, $\overline{WE}$ buffer, $\overline{OE}$ buffer, and Din buffer. If $\overline{CS}$ controls data retention mode, other input levels (address, $\overline{WE}$, $\overline{OE}$, I/O) can be in the high impedance state.

Low V_{CC} Data Retention Timing Waveform

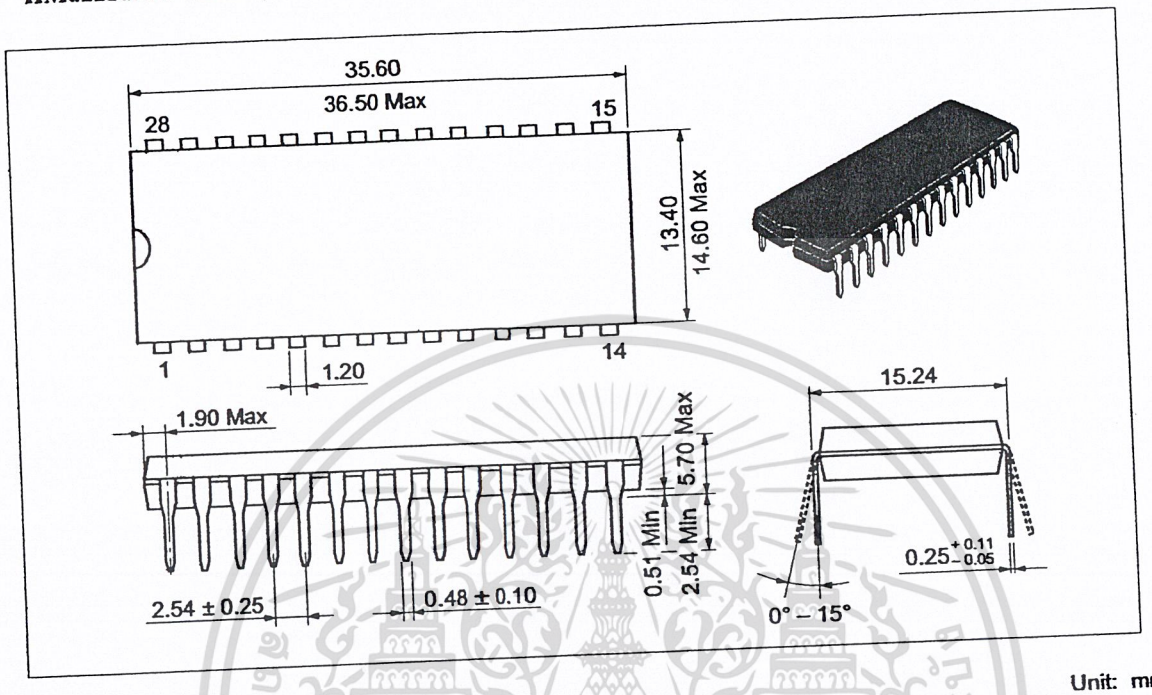


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Package Dimensions

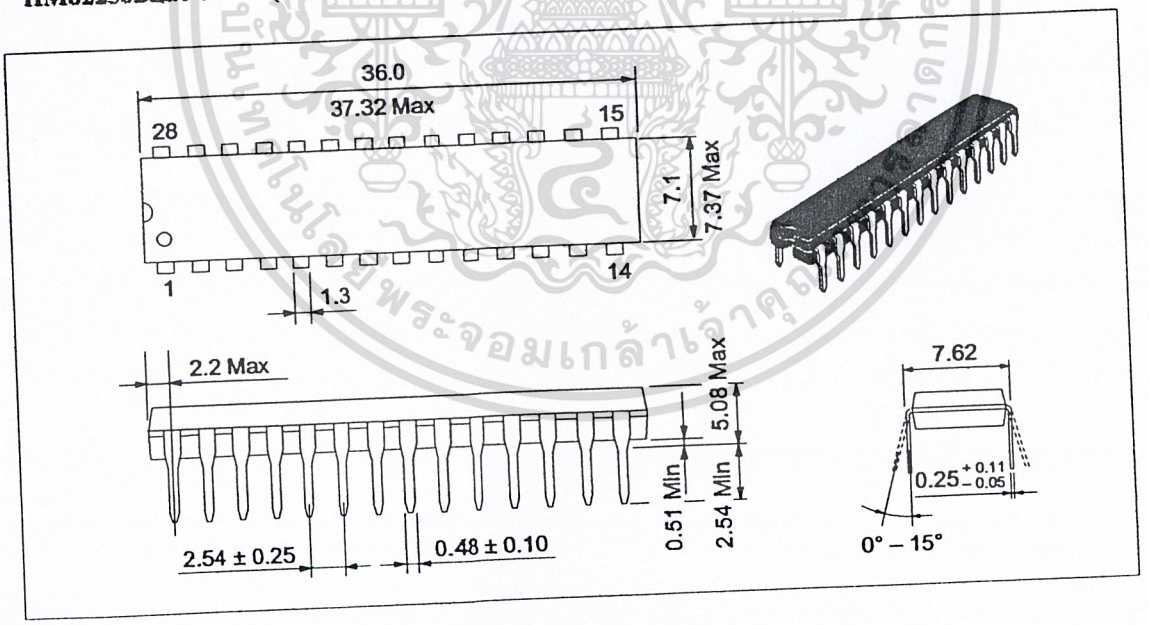
Unit: mm

HM62256BLP Series (DP-28)



Unit: mm

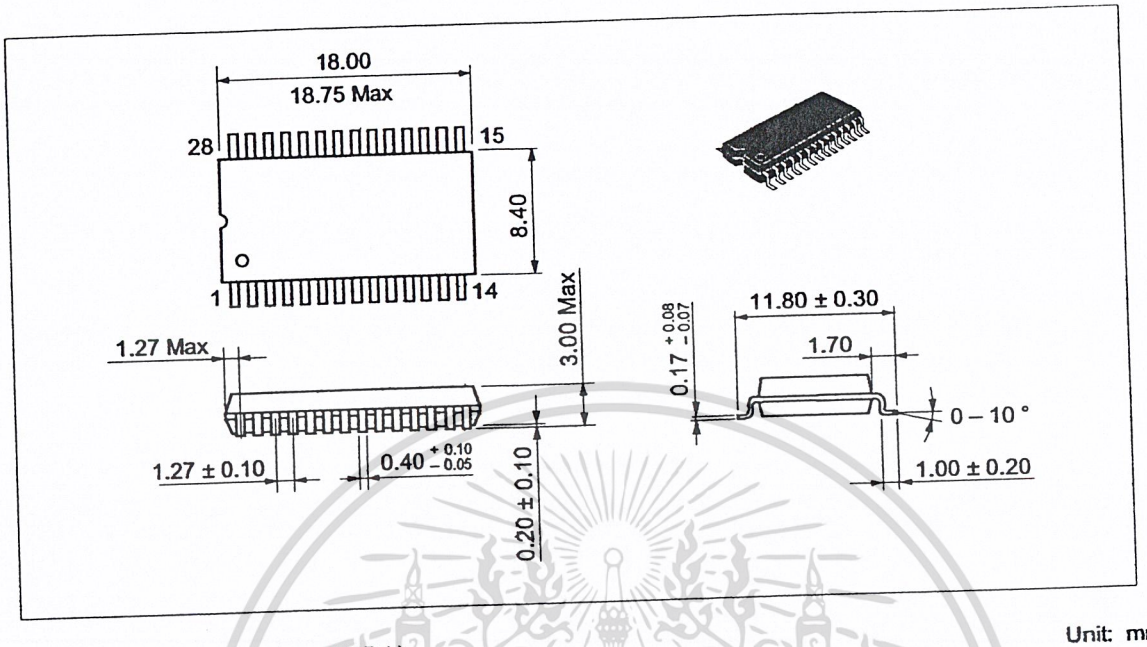
HM62256BLSP Series (DP-28NA)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

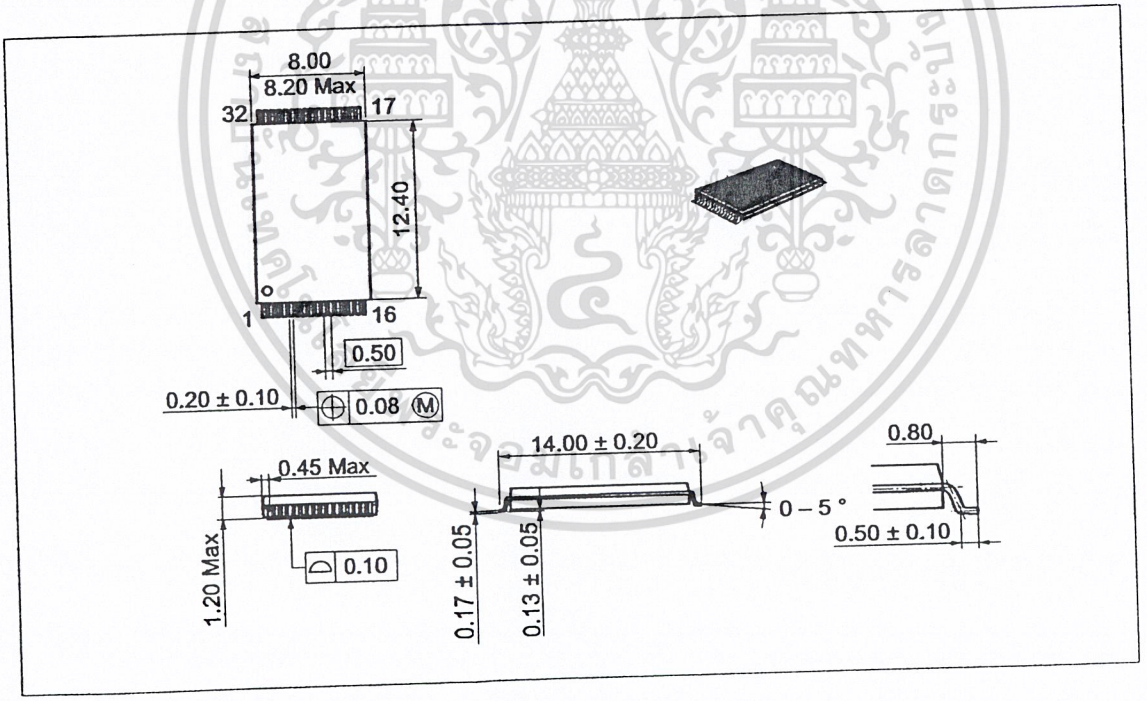
Unit: mm

HM62256BLFP Series (FP-28DA)



Unit: mm

HM62256BLT Series (TFP-32DA)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

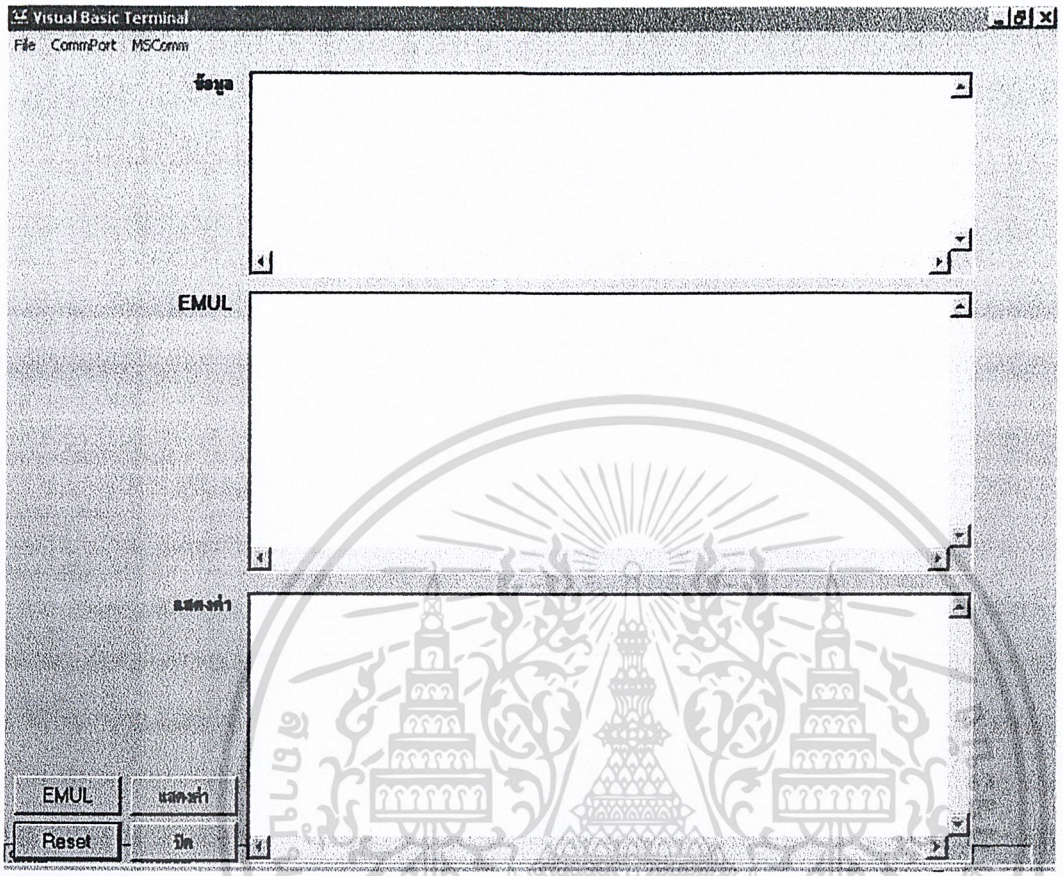
ภาคผนวกโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวกโปรแกรมแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมการแสดงผล



รูป หน้าจอการแสดงผล

การใช้งาน

1. **หน้าจอข้อมูล** จะแสดงข้อมูลที่ทำการจัดเก็บไว้ใน RAM
2. **หน้าจอ Emulator** จะใช้กับการแสดงผลของข้อมูลที่ทำการจัดเก็บมาจากตัว Emulator
3. **หน้าจอ แสดงผล** จะทำการแสดงผลกับข้อมูลที่จัดเก็บมาจากสมาร์ทการ์ดทั่วไป
4. **ปุ่ม Emulator** ใช้สำหรับสั่งให้ทำการแสดงข้อมูลที่หน้าจอ Emulator
5. **ปุ่มแสดงผล** ใช้สำหรับสั่งให้ทำการแสดงข้อมูลที่หน้าจอแสดงผล
6. **ปุ่ม Reset** ใช้สำหรับทำการ Reset หน้าจอ
7. **ปุ่มปิด** ใช้สำหรับปิดโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Source Code ของโปรแกรมแสดงผล (โปรแกรมวิชวลเบสิก)

```
Option Explicit

Dim Ret As Integer     ' Scratch integer.
Dim Temp As String     ' Scratch string.

Private Sub Command1_Click()
   frmBrowser.Show
End Sub

Private Sub btn_exit_Click()
   End
End Sub

Private Sub btn_hex_Click()
   Dim i, j As Integer
   If Len(Trim(receive)) = 0 Then
      MsgBox "no data read"
   Else
      j = 1
      For i = 1 To Len(Trim(receive)) Step 2
         txt_hexshow = txt_hexshow + Mid(receive, i, 2) & " "
         If j = 1020 Then
            j = 1
            txt_hexshow = txt_hexshow + vbNewLine
         Else
            j = j + 1
         End If
      Next i
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    End If
End Sub

Private Sub btn_reset_Click()
    receive.Text = ""
    txt_show.Text = ""
    txt_hexshow.Text = ""
End Sub

Private Sub btn_show_Click()
    Dim st_temp, st_title, st_cut, st_mast As String
    Dim chk_dup, source As String
    Dim i, j, k, l As Integer
    If Len(Trim(receive)) = 0 Then
        MsgBox " no data read"
    Else
        st_temp = Mid(receive, InStr(1, receive, "3B", 1), Len(receive))
        st_cut = "0102"
        i = 1
        j = 1
        If InStr(1, st_temp, st_cut, 1) > 0 Then
            source = Mid(st_temp, 1, InStr(1, st_temp, st_cut, 1) - 1)
        End If
        chk_dup = ""
        Do While InStr(1, st_temp, st_cut, 1) > 0
            If i = 1 Then
                st_title = "Card -> : "
            Else
                st_title = "<- Cam  : "
            End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
If j > 1 Then
   st_title = st_title + "01 02 "
   l = 3
Else
   l = 1
End If
If chk_dup <> source Then
   st_mast = Mid(st_temp, 1, InStr(1, st_temp, st_cut, 1) - 1)
   txt_show.Text = Trim(txt_show.Text) + st_title
   For k = 1 To Len(Trim(st_mast)) Step 2
      txt_show.Text = txt_show.Text + Mid(st_mast, k, 2) + " "
      If l = 20 Then
         l = 1
         txt_show.Text = txt_show.Text + vbNewLine + "          "
      Else
         l = l + 1
      End If
   Next k
   If l < 20 And l > 1 Then
      txt_show.Text = txt_show.Text + vbNewLine
   End If
   st_temp = Mid(st_temp, InStr(1, st_temp, st_cut, 1) + 4, Len(st_temp))
   chk_dup = source
   If InStr(1, st_temp, st_cut, 1) > 0 Then
      source = Mid(st_temp, 1, InStr(1, st_temp, st_cut, 1) - 1)
   End If
   If i = 1 Then
      i = 2
   Else
      i = 1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            End If
        Else
            st_temp = Mid(st_temp, InStr(1, st_temp, st_cut, 1) + 4, Len(st_temp))
            chk_dup = source
            source = Mid(st_temp, 1, InStr(1, st_temp, st_cut, 1) - 1)
        End If
        j = 2
    Loop
    If Len(st_temp) > 0 Then
        If i = 2 Then
            st_title = "<- Cam  : 01 02 "
        Else
            st_title = "Card -> : 01 02 "
        End If
'       txt_show.Text = Trim(txt_show.Text) + st_title + "0102" + st_temp
        st_mast = st_temp
        txt_show.Text = txt_show.Text + st_title
        For k = 1 To Len(Trim(st_mast)) Step 2
            txt_show.Text = txt_show.Text + Mid(st_mast, k, 2) + " "
            If l = 40 Then
                l = l + 1
                txt_show.Text = txt_show.Text + vbNewLine
            Else
                l = l
            End If
        Next k
    End If
  End If
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub Form_Load()
    Dim CommPort As String, Handshaking As String, Settings As String, Echo As String

    On Error Resume Next

    ' Set the default color for the terminal
    txtTerm.SelLength = Len(txtTerm)
    txtTerm.SelText = ""
    txtTerm.ForeColor = vbBlue

    ' Set Title
    App.Title = "Visual Basic Terminal"

    ' Center Form
    frmTerminal.Move (Screen.Width - Width) / 2, (Screen.Height - Height) / 2

    ' Load Registry Settings

    Settings = GetSetting(App.Title, "Properties", "Settings", "") '
frmTerminal.MSComm1.Settings]\
    If Settings <> "" Then
      MSComm1.Settings = Settings
      If Err Then
        MsgBox Error$, 48
        Exit Sub
      End If
    End If

    CommPort = GetSetting(App.Title, "Properties", "CommPort", "") '
frmTerminal.MSComm1.CommPort
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    If CommPort <> "" Then MSComm1.CommPort = CommPort

    Handshaking = GetSetting(App.Title, "Properties", "Handshaking", "")
'frmTerminal.MSComm1.Handshaking
    If Handshaking <> "" Then
      MSComm1.Handshaking = Handshaking
      If Err Then
        MsgBox Error$, 48
        Exit Sub
      End If
    End If

    Echo = GetSetting(App.Title, "Properties", "Echo", "") ' Echo
    On Error GoTo 0

End Sub



Private Sub Form_Unload(Cancel As Integer)
    Dim Counter As Long

    If MSComm1.PortOpen Then
      MSComm1.PortOpen = 0
    End If

    End
End Sub

' Toggle the DTREnabled property.
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub mnuDTREnable_Click()
    ' Toggle DTREnable property
    MSComm1.DTREnable = Not MSComm1.DTREnable
    mnuDTREnable.Checked = MSComm1.DTREnable
End Sub


Private Sub mnuFileExit_Click()
    ' Use Form_Unload since it has code to check for unsent data and an open log file.
    Form_Unload Ret
End Sub


' Display the value of the CDHolding property.
Private Sub mnuHCD_Click()
    If MSComm1.CDHolding Then
        Temp = "True"
    Else
        Temp = "False"
    End If
    MsgBox "CDHolding = " + Temp
End Sub

' Display the value of the CTSHolding property.
Private Sub mnuHCTS_Click()
    If MSComm1.CTSHolding Then
        Temp = "True"
    Else
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        Temp = "False"
    End If
    MsgBox "CTSHolding = " + Temp
End Sub

' Display the value of the DSRHolding property.
Private Sub mnuHDSR_Click()
    If MSComm1.DSRHolding Then
        Temp = "True"
    Else
        Temp = "False"
    End If
    MsgBox "DSRHolding = " + Temp
End Sub

' This procedure sets the InputLen property, which determines how
' many bytes of data are read each time Input is used
' to retreive data from the input buffer.
' Setting InputLen to 0 specifies that
' the entire contents of the buffer should be read.
Private Sub mnuInputLen_Click()
    On Error Resume Next

    Temp = InputBox$("Enter New InputLen:", "InputLen", Str$(MSComm1.InputLen))
    If Len(Temp) Then
        MSComm1.InputLen = Val(Temp)
        If Err Then MsgBox Error$, 48
    End If
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub mnuProperties_Click()
  ' Show the CommPort properties form
  frmProperties.Show vbModal

End Sub

' Toggles the state of the port (open or closed).
Private Sub mnuOpen_Click()
    On Error Resume Next
    Dim OpenFlag

    MSComm1.PortOpen = Not MSComm1.PortOpen
    If Err Then MsgBox Error$, 48

    OpenFlag = MSComm1.PortOpen

    mnuOpen.Checked = OpenFlag

    If MSComm1.PortOpen Then
       sbrStatus.Panels("Settings").Text = "Settings: " & MSComm1.Settings
    Else
       sbrStatus.Panels("Settings").Text = "Settings: "
    End If

End Sub


' This procedure sets the RThreshold property, which determines
' how many bytes can arrive at the receive buffer before the OnComm
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
' event is triggered and the CommEvent property is set to comEvReceive.
Private Sub mnuRThreshold_Click()
    On Error Resume Next

    Temp = InputBox$("Enter New RThreshold:", "RThreshold", Str$(MSComm1.RThreshold))
    If Len(Temp) Then
        MSComm1.RThreshold = Val(Temp)
        If Err Then MsgBox Error$, 48
    End If

End Sub


' The OnComm event is used for trapping communications events and errors.
Private Static Sub MSComm1_OnComm()
    Dim EVMsg$
    Dim ERMsg$

    ' Branch according to the CommEvent property.
    Select Case MSComm1.CommEvent
        ' Event messages.
        Case comEvReceive
            Dim Buffer As Variant
            Buffer = MSComm1.Input
            receive.Text = receive.Text + StrConv(Buffer, vbUnicode)

        Case comEvSend
        Case comEvCTS
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    EVMsg$ = "Change in CTS Detected"
  Case comEvDSR
    EVMsg$ = "Change in DSR Detected"
  Case comEvCD
    EVMsg$ = "Change in CD Detected"
  Case comEvRing
    EVMsg$ = "The Phone is Ringing"
  Case comEvEOF
    EVMsg$ = "End of File Detected"

' Error messages.
  Case comBreak
    ERMsg$ = "Break Received"
  Case comCDTO
    ERMsg$ = "Carrier Detect Timeout"
  Case comCTSTO
    ERMsg$ = "CTS Timeout"
  Case comDCB
    ERMsg$ = "Error retrieving DCB"
  Case comDSRTO
    ERMsg$ = "DSR Timeout"
  Case comFrame
    ERMsg$ = "Framing Error"
  Case comOverrun
    ERMsg$ = "Overrun Error"
  Case comRxOver
    ERMsg$ = "Receive Buffer Overflow"
  Case comRxParity
    ERMsg$ = "Parity Error"
  Case comTxFull
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        ERMsg$ = "Transmit Buffer Full"
    Case Else
        ERMsg$ = "Unknown error or event"
End Select

If Len(EVMsg$) Then
    ' Display event messages in the status bar.
    sbrStatus.Panels("Status").Text = "Status: " & EVMsg$

ElseIf Len(ERMsg$) Then
    ' Display event messages in the status bar.
    sbrStatus.Panels("Status").Text = "Status: " & ERMsg$

    ' Display error messages in an alert message box.
    Ret = MsgBox(ERMsg$, 1, "Click Cancel to quit, OK to ignore.")

    ' If the user clicks Cancel (2)...
    If Ret = 2 Then
        MSComm1.PortOpen = False    ' Close the port and quit.
    End If

End If
End Sub

' This procedure sets the SThreshold property, which determines
' how many characters (at most) have to be waiting
' in the output buffer before the CommEvent property
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
' is set to comEvSend and the OnComm event is triggered.
Private Sub mnuSThreshold_Click()
   On Error Resume Next

   Temp = InputBox$("Enter New SThreshold Value", "SThreshold",
Str$(MSComm1.SThreshold))
   If Len(Temp) Then
      MSComm1.SThreshold = Val(Temp)
      If Err Then MsgBox Error$, 48
   End If
End Sub



Private Sub receive_Change()

End Sub

' Keystrokes trapped here are sent to the MSComm
' control where they are echoed back via the
' OnComm (comEvReceive) event, and displayed
' with the ShowData procedure.
Private Sub txtTerm_KeyPress(KeyAscii As Integer)
   ' If the port is opened...
   If MSComm1.PortOpen Then
      ' Send the keystroke to the port.
      MSComm1.Output = Chr$(KeyAscii)

   txtTerm.SelStart = Len(txtTerm)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    End If

End Sub
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวกโปรแกรมเครื่องจับสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;*****************************************************************
; Program :       Serial data logger
; Filename :      ukaraporn
;
;*****************************************************************
        List    P=16F877
;       RADIX HEX
;*****************************************************************
        Include "P16F877.inc"

CONFIG1 = _CP_OFF & _DEBUG_OFF & _WRT_ENABLE_OFF & _CPD_OFF &
_LVP_OFF
CONFIG2 = _BODEN_OFF & _PWRTE_ON & _WDT_OFF & _HS_OSC
        __Config CONFIG1 & CONFIG2
;*****************************************************************
;-----------------lcd 16*1 define area-----------------------------;
        #define         lcd_out PORTC           ; lcd out IS PORTC
        #define         RS      PORTA,1         ; RS      IS PORTA,1
        #define         E       PORTA,2         ; E       IS PORTA,2

;-------------------------------------------------------------------;
        ORG             0X20
        ADDRESLW    RES     1
        ADDRESHW    RES     1
        ADDRESLR    RES     1
        ADDRESHR    RES     1
        NUM             RES     1
        CUR             RES     1
        ADD             RES     1
        ASC             RES     1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
temp            RES    1
gcount1         RES    1
gcount2         RES    1
gcount3         RES    1
temp_data       RES    1
DATAS           RES    1
ADDRESLRL       RES    1
ADDRESHRL       RES    1
num1            RES    1
MAKE            RES    1
MAKE1           RES    1
SUM             RES    1
RAM             RES    1
RAM1            RES    1
;*********************************************************************
        ORG     0x0000
        BANKSEL TRISB           ; SELECT BANK 1
        movlw   b'11000000'     ; PORTC RC6,RC7 IS INPUT
        movwf   TRISC
        CLRF    TRISB           ; PORTB IS OUTPUT ALL
        CLRF    TRISD           ; PORTD IS OUTPUT ALL
        MOVLW   0X07            ; PORTA AND PORTD IS ANALOG
        MOVWF   ADCON1
        Movlw   0x87            ; Asigna un preescaler de 1:256
        Movwf   OPTION_REG
        MOVLW   B'00001001'     ; PORTA RA0 AND RA3 IS INPUT
        MOVWF   TRISA
        CLRF    TRISE           ; PORTE IS OUTPUT ALL
        MOVLW   .129
        MOVWF   SPBRG           ; baudrate = 9600 bps
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
BSF         TXSTA,BRGH  ; HI SPEED baudrate
BSF         TXSTA,TXEN
BANKSEL     RCSTA       ; Bank 0

movlw       0x00        ; GENERRAL REGESTER ADDRESLW BEGIN
                        ; AT  ZERO

movwf       ADDRESLW
movlw       0x00        ; GENERRAL REGESTER ADDRESHW BEGIN
                        ; AT  ZERO

movwf       ADDRESHW
movlw       .0          ; GENERRAL REGESTER NUM BEGIN AT
                        ;  ZERO
movwf       NUM
movlw       0x00        ; GENERRAL REGESTER ADDRESLR BEGIN
                        ; AT  ZERO

movwf       ADDRESLR
movlw       0x00        ; GENERRAL REGESTER ADDRESHR BEGIN
                        ; AT  ZERO

movwf       ADDRESHR
MOVLW       0X00        ; GENERRAL REGESTER ADD BEGIN AT
                        ; ZERO
MOVWF       ADD
movlw       0x00        ; GENERRAL REGESTER num1 BEGIN AT
                        ;  ZERO
movwf       num1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        movlw   0x00        ; GENERRAL REGESTER ADDRESLRL BEGIN
                            ; AT  ZERO

        movwf   ADDRESLRL
        MOVLW   0X00        ; GENERRAL REGESTER ADDRESHRL BEGIN
                            ; AT  ZERO

        MOVWF   ADDRESHRL
        movlw   .1          ; GENERRAL REGESTER MAKE BEGIN AT
                            ; ONE
        movwf   MAKE
        movlw   0x01        ; GENERRAL REGESTER MAKE1 BEGIN AT
                            ;  ONE
        movwf   MAKE1
        CLRF    PORTC               ; PORTC IS ZERO
        bsf     E                   ; PORTA,1 IS ONE
        bsf     RS                  ; PORTA,2 IS ONE
        BSF     RCSTA,SPEN          ; Serial port enable
        BSF     RCSTA,CREN          ; Begin receive

START
        BTFSS   PORTA,0             ; CHACK PORTA,0  IF PORTA,O = 1
SKIP
        GOTO    $+2
        GOTO    $+3
        CALL    SHOW_RAM
        GOTO    SHOW_DATA

READ_DATA
        BTFSS   PIR1,RCIF           ; If have data then skip
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        GOTO    START           ; No data then waiting
        MOVFW   RCREG           ; Move data to w
        MOVWF   CUR             ; Move data to CUR
        CALL    WRITE_MEM
        GOTO    START

WRITE_MEM
        BSF     PORTE,0         ; ENABLE LATCH
        MOVF    ADDRESHW,W      ; MOVE ADDRESS HI TO W
        MOVWF   PORTD           ; MOVE ADDRESS HI TO MEMORY
        BCF     PORTE,0         ; DISABLE LATCH
        MOVF    ADDRESLW,W      ; MOVE ADDRESS LO TO W
        MOVWF   PORTD           ; MOVE ADDRESS LO TO MEMMORY
        BCF     PORTE,1         ; ENABLE CHIP SELECT
        MOVFW   CUR             ; MOVE  DATA TO W
        MOVWF   PORTB           ; MOVE DATA TO MEMORY
        Nop                     ; DELAY 1 CYCLE
        BCF     PORTE,2         ; ENABLE WRITE
        NOP                     ; DELAY 1 CYCLE
        BSF     PORTE,2         ; DISABLE WRITE
        BSF     PORTE,1         ; DISABLE SHIP SELECT
        bcf     STATUS,Z        ; SET BIT ZERO CHACK IS 0
        INCF    ADDRESLW,F      ; INCREAT ADDRESS LO
        btfss   STATUS,Z        ; SKIP IF BIT ZERO CHACK IS 1
        GOTO    $+2
        GOTO    $+4
        INCF    NUM,F           ; INCREAT NUM
        incf    num1,f          ; INCREAT num1
        GOTO    $+8
        INCF    ADDRESHW,F      ; INCREAT ADDRESS HI
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        INCF    MAKE,F          ; INCREAT MAKE
        INCF    MAKE1,F         ; INCREAT MAKE1
        MOVLW   .0              ; SET NUM IS 0
        MOVWF   NUM
        MOVLW   0X00            ; SET num1 IS 0
        MOVWF   num1
        RETURN

SHOW_DATA
        BTfSS   PORTA,3         ; SKIP IF PORTA,3 IS 1
        goto    main_prog
        BCF     PORTE,2         ; DISABLE READ
        BSF     PORTE,2         ; ENABLE READ
        BSF     PORTE,0         ; ENABLE LATCH
        MOVF    ADDRESHR,W      ; MOVE ADDRESS HI TO W
        MOVWF   PORTD           ; MOVE ADDRESS HI TO MEMORY
        BCF     PORTE,0         ; DISABLE LATCH
        MOVF    ADDRESLR,W      ; MOVE ADDRESS LO TO W
        MOVWF   PORTD           ; MOVE ADDRESS LO TO MEMORY
        BCF     PORTE,1         ; ENABLE SHIP SELECT
        NOP                     ; DELAY 1 CYCLE
        MOVF    PORTB,W         ; MOVE DATA TO W
        BSF     PORTE,1         ; DISABLE SHIP SELECT
        MOVWF   ASC             ; MOVE DATA TO ASC
TX
        BTFSS   PIR1,TXIF       ; SKIP IF TXREG EMPTY
        GOTO    $-1             ; NOT EMPTY THEN WAIT
        SWAPF   ASC,W           ; 4 BIT HI CHANGE ASCII
        ANDLW   0X0F
        CALL    ASCII_TAB
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOVWF   TXREG           ; SAND DATA 4 BIT HI TO RS 232
        BANKSEL TXSTA           ; BANK 1
        BTFSS   TXSTA,TRMT      ; SKIP IF TRMT NOT EMPTY
        GOTO    $-1             ; EMPTY THEN WAIT
        BANKSEL PORTA           ; BANK 0
        BTFSS   PIR1,TXIF       ; SKIP IF TXREG EMPTY
        GOTO    $-1             ; NOT EMPTY THEN WAIT
        MOVF    ASC,W           ; 4 BIT LO CHANGE ASCII
        ANDLW   0X0F            ;
        CALL    ASCII_TAB       ;
        MOVWF   TXREG           ; SAND DATA 4 BIT LO TO RS 232
        BANKSEL TXSTA           ; BANK 1
        BTFSS   TXSTA,TRMT      ; SKIP IF TRMT NOT EMPTY
        GOTO    $-1             ; EMPTY THEN WAIT
        BANKSEL PORTA           ; BANK 0
        bcf     STATUS,Z        ; SET BIT ZERO CHACK IS 0
        DECF    NUM,F           ; DECREAT NUM
        btfss   STATUS,Z        ; SKIP IF BIT ZERO CHACK IS 1
        goto    $+2
        goto    $+6
ADL     bcf     STATUS,Z        ; SET BIT ZERO CHACK IS 0
        INCF    ADDRESLR,F      ; INCREAT ADDRESS LO
        btfss   STATUS,Z        ; SKIP IF BIT ZERO CHACK IS 1
        GOTO    SH
        GOTO    ADH
        bcf     STATUS,Z        ; SET BIT ZERO CHACK IS 0
        DECF    MAKE            ; DECREAT MAKE
        btfss   STATUS,Z        ; SKIP IF BIT ZERO CHACK IS 1
        GOTO    $+2
        GOTO    $+4
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        GOTO        ADL
ADH     INCF        ADDRESHR,F          ; INCREAT ADDRESS HI
SH      GOTO        SHOW_DATA
        GOTO        $
;-------------------------------------------------
ASCII_TAB
        addwf       PCL,F
        dt          "0123456789ABCDEF"
        return

;*************** LCD ***************************************
main_prog
        clrf        ADDRESLRL           ; SET ADDRESLRLIS 0
        clrf        ADDRESHRL           ; SET ADDRESHRL 0
        clrf        DATAS               ; SET DATAS 0
        movf        NUM,w               ; MOVE NUM TO W
        movwf       num1                ; MOVE W TO num1
        movf        MAKE,w              ; MOVE MAKE TO W
        movwf       MAKE1               ; MOVE W TO MAKE1
        BTFSC       PORTA,3             ; SKIP IF SWITCH IS GND
        GOTO        SHOW_DATA
        movlw       B'00101000'         ; 4 BITS, 2 LINES , 5X7 DOT
        call        lcd_command
        movlw       B'00001100'         ; DISPLAY ON/OFF
        call        lcd_command
        movlw       B'00000110'         ; ENTRY MODE
        call        lcd_command
        movlw       B'00000001'         ; CLEAR RAM
        call        lcd_command
loop    movlw       0x80                ; RAM ADDRESS HI OF LCD
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        call    lcd_command
        movlw   "D"
        call    lcd_data
        movlw   "A"
        call    lcd_data
        movlw   "T"
        call    lcd_data
        movlw   "A"
        call    lcd_data
        movlw   "L"
        call    lcd_data
        movlw   "O"
        call    lcd_data
        movlw   "G"
        call    lcd_data
        movlw   "G"
        call    lcd_data

slow    movlw   0x05            ; MOVE 5 TO SUM
        movwf   SUM
        MOVLW   0xC0            ; RAM ADDRESS LO OF LCD
        call    lcd_command
LOOP1   decfsz  SUM             ; DECREAT AND SKIP IF SUM IS 0
        goto    $+2
        goto    slow
        CALL    READ
        swapf   DATAS,w         ; 4 BIT HI OF DATAS CHANGE ASCII
        andlw   0X0f
        call    ASCII_TAB
        call    lcd_data
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        movf    DATAS,w         ; 4 BIT LO OF DATAS CHANGE
ASCII
        andlw   0X0f
        call    ASCII_TAB
        call    lcd_data
        movlw   .255            ; DELAY LCD
        call    delay
        movlw   .255
        call    delay
        movlw   .255
        call    delay
        movlw   .255
        call    delay
        movlw   .255
        call    delay
        movlw   .255
        call    delay
        movlw   .255
        call    delay
        movlw   .255
        call    delay
        movlw   .255
        call    delay
        movlw   .255
        call    delay
        movlw   .255
        call    delay
        movlw   .255
        call    delay


        bcf     STATUS,Z        ; SET BIT ZERO CHACK IS 0
        DECF    num1,F          ; DECERAT num1
        btfss   STATUS,Z        ; SKIP IF BIT ZERO CHACK IS 1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        goto    $+2
        goto    $+6
ADL1    bcf     STATUS,Z        ; SET BIT ZERO CHACK IS 0
        INCF    ADDRESLRL,F     ; INCREAT ADDRESS LO
        btfss   STATUS,Z        ; SKIP IF BIT ZERO CHACK IS 1
        GOTO    SH1
        GOTO    ADH1
        bcf     STATUS,Z        ; SET BIT ZERO CHACK IS 0
        DECF    MAKE1           ; DECREAT MAKE1
        btfss   STATUS,Z        ; SKIP IF BIT ZERO CHACK IS 1
        GOTO    $+2
        GOTO    $+4
        GOTO    ADL1
ADH1    INCF    ADDRESHRL,F     ;INCREAT ADDRESS HI
SH1     goto    LOOP1
        goto    main_prog

;----------------------write data to lcd-----------------;
lcd_data
        bsf     RS              ;SEND 1st HALF DATA TO LCD
        movwf   temp_data       ; MOVE W TO temp _data
        swapf   temp_data,w     ; 4 BIT HI CHANGE TO ASCII
        andlw   0x0F
        movwf   lcd_out         ; SEND TO LCD
        bcf     E
        movlw   .25             ; DELAY 5 ms
        call    delay
        bsf     E
        movf    temp_data,w     ; SEND 2nd HALF DATA TO LCD
        andlw   0x0F            ; 4 BIT LO CHANGE TO ASCII
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        movwf   lcd_out         ; SEND TO LCD
        bcf     E
        movlw   .25             ; DELAY 5 ms
        call    delay
        bsf     E
        return
;----------write command to lcd----------;
lcd_command
        bcf     RS              ;SEND 1st HALF DATA TO LCD
        movwf   temp            ; MOVE W TO temp
        SWAPF   temp,W          ; 4 BIT HI CHANGE TO ASCII
        andlw   0x0F
        movwf   lcd_out         ; SEND TO LCD
        bcf     E
        movlw   .25             ; DELAY 5 ms
        call    delay
        bsf     E
        movf    temp,w          ; SEND 2nd HALF DATA TO LCD
        andlw   0x0F
        movwf   lcd_out         ; SEND TO LCD
        bcf     E
        movlw   .25             ; DELAY 5 ms
        call    delay
        bsf     E
        return
delay                           ; 5 ms
        movwf   gcount1
        movlw   0x0a
        movwf   gcount2
        movlw   0x20
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        movwf   gcount3
        decfsz  gcount3,F
        goto    $-1
        decfsz  gcount2,F
        goto    $-5
        decfsz  gcount1,F
        goto    $-9
        return
READ
        BCF     PORTE,2         ; DISABLE READ
        BSF     PORTE,2         ; ENABLE READ
        BSF     PORTE,0         ; ENABLE LATCH
        MOVF    ADDRESHRL,W     ; MOVE ADDRESS HI TO W
        MOVWF   PORTD           ; MOVE ADDRESS HI TO MEMORY
        BCF     PORTE,0         ; DISABLE LATCH
        MOVF    ADDRESLRL,W     ; MOVE ADDRESS LO TO W
        MOVWF   PORTD           ; MOVE ADDRESS LO TO MEMORY
        BCF     PORTE,1         ; ENABLE SHIP SELECT
        BCF     PORTA,1         ; ENABLE RS
        NOP                     ; DELAY 1 CYCLE
        MOVF    PORTB,W         ; SAND DATA TO LCD
        BSF     PORTA,1         ; DISABLE RS
        BSF     PORTE,1         ; DISABLE SHIP SELECT
        MOVWF   DATAS           ; MOVE W TO DATAS
        RETURN
;************************NUMBER BYTE************************
SHOW_RAM
        movlw   B'00101000'     ; 4 BITS, 2 LINES , 5X7 DOT
        call    lcd_command
        movlw   B'00001100'     ; DISPLAY ON/OFF
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
call        lcd_command
movlw       B'00000110'         ; ENTRY MODE
call        lcd_command
movlw       B'00000001'         ; CLEAR RAM
call        lcd_command
movlw       0x80                ; RAM ADDRESS HI OF LCD
call        lcd_command
movlw       "U"
call        lcd_data
movlw       "S"
call        lcd_data
movlw       "E"
call        lcd_data
movlw       " "
call        lcd_data
movlw       "B"
call        lcd_data
movlw       "Y"
call        lcd_data
movlw       "T"
call        lcd_data
movlw       "E"
call        lcd_data
MOVLW       0xC0                ; RAM ADDRESS LO OF LCD
call        lcd_command
movlw       " "
call        lcd_data
MOVFW       MAKE                ; MOVE MAKE TO W
MOVWF       RAM                 ; MOVE W TO RAM
swapf       RAM,w               ; 4 BIT HI OF DATAS CHANGE ASCII
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        andlw       0X0f
        call        ASCII_TAB
        call        lcd_data            ; SAND DATA TO LCD
        movf        RAM,w               ; 4 BIT LO OF DATAS CHANGE
ASCII
        andlw       0X0f
        call        ASCII_TAB
        call        lcd_data            ; SAND DATA TO LCD
        movlw       " "
        call        lcd_data
        MOVFW       NUM                 ; MOVE NUM TO W
        MOVWF       RAM1                ; MOVE W TO RAM1
        swapf       RAM1,w              ; 4 BIT HI OF DATAS CHANGE ASCII
        andlw       0X0f
        call        ASCII_TAB
        call        lcd_data            ; SAND DATA TO LCD
        movf        RAM1,w              ; 4 BIT HI OF DATAS CHANGE ASCII
        andlw       0X0f
        call        ASCII_TAB
        call        lcd_data            ; SAND DATA TO LCD
        RETURN


        END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้