สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เครื่องบันทึกข้อมูลบัตรแม่เหล็ก

MAGNETIC CARD DATA RECORDER



เลขหมู่.....
เลขทะเบียน 42295
วัน, เดือน, ปี 16 พ.ค. 2545
b.....
i.....

โดย

นายวัชรพงษ์ เขื่อนขันธ์

นางสาววัชรา อนุรักษ์

ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องบันทึกข้อมูลบัตรแม่เหล็ก

MAGNETIC CARD DATA RECORDER

โดย

นายวัชรพงษ์ เขื่อนขันธ์ 40010711

นางสาววัชรา อนุรักษ์ 40010712

อาจารย์ที่ปรึกษา

อ.ชินภัทร นันทจิวากรชัย

ปริญญานิพนธ์สำหรับปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องบันทึกข้อมูลบัตรแม่เหล็ก

MAGNETIC CARD DATA RECORDER

นายวัชรพงษ์ เขื่อนขันธ์ รหัสประจำตัว 40010711

นางสาววัชรา อนุรักษ์ รหัสประจำตัว 40010712

โครงงานฉบับนี้ได้รับการตรวจสอบแล้ว พร้อมที่จะทำการสอบได้

ลงชื่อ.....อาจารย์ที่ปรึกษา

(อ.ชินภัทร นันทจิวากรชัย)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายงานปีการศึกษา 2543

ภาควิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เครื่องบันทึกข้อมูลบัตรแม่เหล็ก (MAGNETIC CARD DATA RECORDER)

ผู้จัดทำ

1. นายวัชรพงษ์ เขื่อนขันธ์ รหัสประจำตัว 40010711
2. นาสาววัชรา อนุรักษ์ รหัสประจำตัว 40010712

.....อาจารย์ที่ปรึกษา

(อ.ชินภัทร นันทจิวากรชัย)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องบันทึกข้อมูลบัตรแม่เหล็ก

วัชรพงษ์ เขื่อนขันธ์
วัชรา อนุรักษ์
อ.ชินภัทร นันทจิวากรชัย อาจารย์ที่ปรึกษา
ปีการศึกษา 2543

บทคัดย่อ

เครื่องบันทึกข้อมูลบัตรแม่เหล็ก เป็นอุปกรณ์ที่ใช้ในการบันทึกข้อมูลจากบัตรแม่เหล็ก โดยการบันทึกข้อมูลที่มีอยู่บนบัตรและวันเวลาลงยังแผ่นฟล็อปปีดิสก์ โดยมีจุดมุ่งหมายเพื่อความสะดวกในการใช้งาน สามารถเคลื่อนย้ายได้โดยไม่จำเป็นต้องต่อพ่วงกับเครื่องคอมพิวเตอร์ และแผ่นฟล็อปปีดิสก์ก็เป็นอุปกรณ์ที่หาและพกพาได้สะดวก และเครื่องบันทึกข้อมูลนี้ยังสามารถนำเอาไปประยุกต์ใช้กับการบันทึกข้อมูลอื่น ๆ ได้

ปริญญานิพนธ์ฉบับนี้ได้กล่าวถึงทฤษฎีเกี่ยวกับการบันทึกข้อมูลลงยังดิสก์ การอ่านข้อมูลจากบัตรแม่เหล็ก และการใช้งานอุปกรณ์ต่าง ๆ ซึ่งควบคุมการทำงานด้วยไมโครคอนโทรลเลอร์ การทำงานและการทดสอบเครื่องบันทึกข้อมูลบัตรแม่เหล็ก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MAGNETIC CARD DATA RECORDER

Watcharapong Khuenkhan

Watchara Anurak

Mr.Chinnapat Nantajiwakornchai (Advisor)

2000

ABSTRACT

Magnetic Card Data Logger is a device that is used to save data from a magnetic card to a floppy disk. Stored data consists of data from the magnetic card and time when the card is drawn. The objectives of this thesis is to design the recorder that are comfortable to use, portable and unnecessary connecting to the computer. Otherwise, this Magnetic Card Data Logger can apply to save other data.

This thesis consists of the theories about saving data to the floppy disk, reading data from the magnetic card , using other devices that are controlled by microcontroller, operating and testing the Magnetic Card Data Recorder.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อ	**I**
ABSTRACT	**II**
สารบัญ	**III**
สารบัญรูปภาพ	**VI**
สารบัญตาราง	**VIII**
บทที่ 1 บทนำ	1
1.1 วัตถุประสงค์	**1**
1.2 ส่วนประกอบ	**2**
1.3 ขอบเขตการทำงาน	**3**
บทที่ 2 การบันทึกข้อมูลลงแผ่นฟล็อปปีดิสก์	**4**
2.1 โครงสร้างของดิสก์เก็ตต์	**4**
2.1.1 แทร็กและเซกเตอร์	4
2.1.2 การเก็บข้อมูลในแผ่นฟล็อปปีดิสก์	5
2.1.3 การมอดูเลชันของข้อมูล	6
2.2 การจัดข้อมูลในดิสก์	**7**
2.2.1 ไดเรคทอรี	7
2.2.2 ตารางการจัดไฟล์ (File Allocation table – FAT)	11
2.3 ฟล็อปปีดิสก์คอนโทรลเลอร์	**15**
2.3.1 องค์ประกอบ	15
2.3.2 คำอธิบายเกี่ยวกับรีจิสเตอร์	17
2.3.3 รีจิสเตอร์สถานะส่วนผลลัพธ์	20
2.3.4 ส่วนซอฟท์แวร์อินเตอร์เฟสโปรเซสเซอร์	23
2.3.5 อธิบายคำสั่งที่จำเป็นในการใช้งาน	24
2.4 ความรู้เบื้องต้นเกี่ยวกับบัตรแม่เหล็ก	**28**
2.4.1 มาตรฐานของบัตรแม่เหล็ก	28
2.4.2 ชุดรหัสข้อมูลในแทร็กที่ 2 ของบัตรแม่เหล็ก	29

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

2.4.3 รูปแบบของข้อมูลที่บันทึกในแทร็กที่ 2 บนบัตรแม่เหล็ก 30
2.5 เครื่องอ่านบัตรแม่เหล็ก 30
2.5.1 การอ่านข้อมูลจากบัตรแม่เหล็ก 32
2.5.2 การตรวจสอบความผิดพลาด 34
บทที่ 3 ความรู้พื้นฐานเกี่ยวกับไมโครคอนโทรลเลอร์และการสร้างฐานเวลา 38
3.1 สถาปัตยกรรมของไมโครคอนโทรลเลอร์แบบชิพเดี่ยวตระกูล MCS-51 38
3.1.1 โครงสร้างของ AT89S8252 39
3.1.2 การจัดการหน่วยความจำ 41
3.1.3 การจัดวางขาและหน้าที่ต่างๆ 45
3.2 การสร้างฐานเวลาให้แก่ MCS-51 49
3.2.1 คุณสมบัติของชิพ RTC เบอร์ DS12887A 49
3.2.2 แผนผังแสดงตำแหน่งต่างๆ ใน RTC 50
3.2.3 รีจิสเตอร์ควบคุม 51
บทที่ 4 การออกแบบและการทำงาน 56
4.1 ส่วนประมวลผล หน่วยความจำข้อมูล และส่วนอ่านบัตรแม่เหล็ก 56
4.2 ส่วนวงจรควบคุมดิสก์ 57
4.3 ส่วนแสดงผล 58
4.4 ส่วนวงจรสร้างฐานเวลา 59
4.5 ส่วนคีย์บอร์ดรับคำสั่งการทำงาน 59
4.6 โปรแกรมแสดงผลบนจอคอมพิวเตอร์ 60
4.7 แผนผังแสดงการทำงาน 61
บทที่ 5 การทดสอบการทำงาน 66
5.1 การทดสอบอ่านข้อมูลจากบัตรแม่เหล็ก 66
5.2 การทดสอบการเขียนข้อมูลจากบัตรแม่เหล็กลงดิสก์ 67
บทที่ 6 บทสรุป 69
6.1 สรุปและวิจารณ์ 69

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

6.2 ปัญหาและอุปสรรคในการทำงาน 69
6.3 แนวทางแก้ไขปัญหาและปรับปรุง 69

ภาคผนวก
กิตติกรรมประกาศ
บรรณานุกรม



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

หน้า

รูปที่ 1.1 แผนผังแสดงส่วนประกอบของเครื่องบันทึกข้อมูล 2

รูปที่ 2.1 การจัดแทร็กและเซกเตอร์ภายในดิสเก็ตต์ 4

รูปที่ 2.2 หัวอ่านบันทึกดิสก์ ในขณะที่ทำการบันทึกข้อมูล 5

รูปที่ 2.3 File Allocation Table (FAT) 12

รูปที่ 2.4 แสดงขนาดมาตรฐานของบัตรแม่เหล็ก 28

รูปที่ 2.5 รูปแบบของข้อมูลที่บันทึกในแทร็กที่ 2 บนบัตรแม่เหล็ก 30

รูปที่ 2.6 แสดงความสัมพันธ์ของสัญญาณ Data,Clock และ Card Present ของเครื่องอ่านบัตรแม่เหล็ก 31

รูปที่ 2.7 เครื่องอ่านบัตรแม่เหล็ก 32

รูปที่ 2.8 แสดงหลักการสร้างพาริตี้บิต 35

รูปที่ 2.9 แสดงวงจรการสร้างพาริตี้บิต 36

รูปที่ 3.1 ไดอะแกรมโครงสร้างของ AT89S8252 40

รูปที่ 3.2 การจัดวางขาของ AT89S8252 45

รูปที่ 3.3 วงจรออสซิลเลเตอร์ภายใน AT89S8252 49

รูปที่ 3.4 การจัดวางขาของ DS12887A 50

รูปที่ 3.5 แผนผังแสดงตำแหน่งต่างๆใน RTC 51

รูปที่ 4.1 แสดงส่วนประมวลผล หน่วยความจำข้อมูล และส่วนหัวอ่านบัตรแม่เหล็ก 56

รูปที่ 4.2 แสดงส่วนวงจรควบคุมดิสก์ 57

รูปที่ 4.3 แสดงส่วนแสดงผลของเครื่องบันทึกข้อมูล 59

รูปที่ 4.4 ส่วนวงจรสร้างฐานเวลา 59

รูปที่ 4.5 ส่วนคีย์บอร์ดรับคำสั่งการทำงาน 60

รูปที่ 4.6 โปรแกรมแสดงผลบนจอคอมพิวเตอร์ 60

รูปที่ 4.7 แผนผังแสดงการทำงานของเครื่องบันทึกข้อมูลบัตรแม่เหล็ก 61

รูปที่ 4.8 แผนผังแสดงการเก็บข้อมูลลงยังหน่วยความจำ 62

รูปที่ 4.9 แผนผังแสดงการควบคุมการทำงานของส่วนฐานเวลา 63

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

รูปที่ 4.10 แผนผังแสดงการบันทึกข้อมูลลงดิสก์ 64
รูปที่ 4.11 แผนผังแสดงการตรวจสอบรหัสผ่าน 65
รูปที่ 5.1 แสดงข้อมูลจากการรูดบัตรที่บันทึกลงยังแผ่นดิสก์ 67
รูปที่ 5.2 แสดงข้อมูลของเจ้าของบัตรแม่เหล็ก และวันเวลาที่ทำการรูดบัตร 68

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่ 2.1 จำนวนแทร็กและเซกเตอร์ในดิสก์เกตต์แบบต่าง ๆ 14

ตารางที่ 2.2 ผังแสดงแอดเดรสของหน่วยความจำของ DP8473 17

ตารางที่ 2.3 แสดงค่าสำหรับการเลือกในรีจิสเตอร์ควบคุมไดรฟ์ 19

ตารางที่ 2.4 อัตราของการรับส่งข้อมูลและค่าของการชดเชยโปรแกรม 19

ตารางที่ 2.5 สรุป FDC Register 22

ตารางที่ 2.6 แสดงค่าเวลาสูงสุดที่ใช้ใน DMA Mode หรือ Interrupt Mode 24

ตารางที่ 2.7 แสดงการเลือกขนาดของเซกเตอร์ 25

ตารางที่ 2.8 แสดงค่าเวลาการเลื่อนหัวอ่าน การเริ่มและการหยุดมอเตอร์ที่ อัตราการรับส่งข้อมูล 500 kb/s แบบ MFM 26

ตารางที่ 2.9 แสดงรหัสข้อมูลแต่ละตัวของแทร็กที่ 2 29

ตารางที่ 2.10 แสดงสายสัญญาณของเครื่องอ่านบัตรแม่เหล็ก 32

ตารางที่ 2.11 แสดงข้อมูลของรหัส BCD สำหรับเครื่องอ่านบัตรแม่เหล็ก 34

ตารางที่ 2.12 แสดงการตรวจซ้ำในแนวตั้งและแนวนอน 37

ตารางที่ 3.1 แผนผัง SFR และค่ารีจิสเตอร์เมื่อเกิดการรีเซต 42

ตารางที่ 3.2 วอทช์ด็อกและรีจิสเตอร์ควบคุมหน่วยความจำ 43

ตารางที่ 3.3 รีจิสเตอร์ควบคุม SPI 43

ตารางที่ 3.4 รีจิสเตอร์แสดงสภาวะของ SPI 44

ตารางที่ 3.5 แสดงโหมดข้อมูลของ RTC 51

ตารางที่ 4.1 แสดงแอดเดรสของรีจิสเตอร์ภายในตัวไอซี DP8473 58

ตารางที่ 5.1 ผลการทดสอบความถูกต้องของการอ่านข้อมูลบนบัตรแม่เหล็ก หลายๆ ครั้ง 66

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

ระบบการจัดเก็บข้อมูลนั้น มีความสำคัญสำหรับงานทุกๆประเภทที่ต้องนำข้อมูลที่ได้นี้ไปทำการวิเคราะห์ผลทางสถิติ หรือในการดำเนินการต่างๆ ดังนั้นระบบที่มีความถูกต้องและแม่นยำจึงเป็นสิ่งที่มีความสำคัญมาก

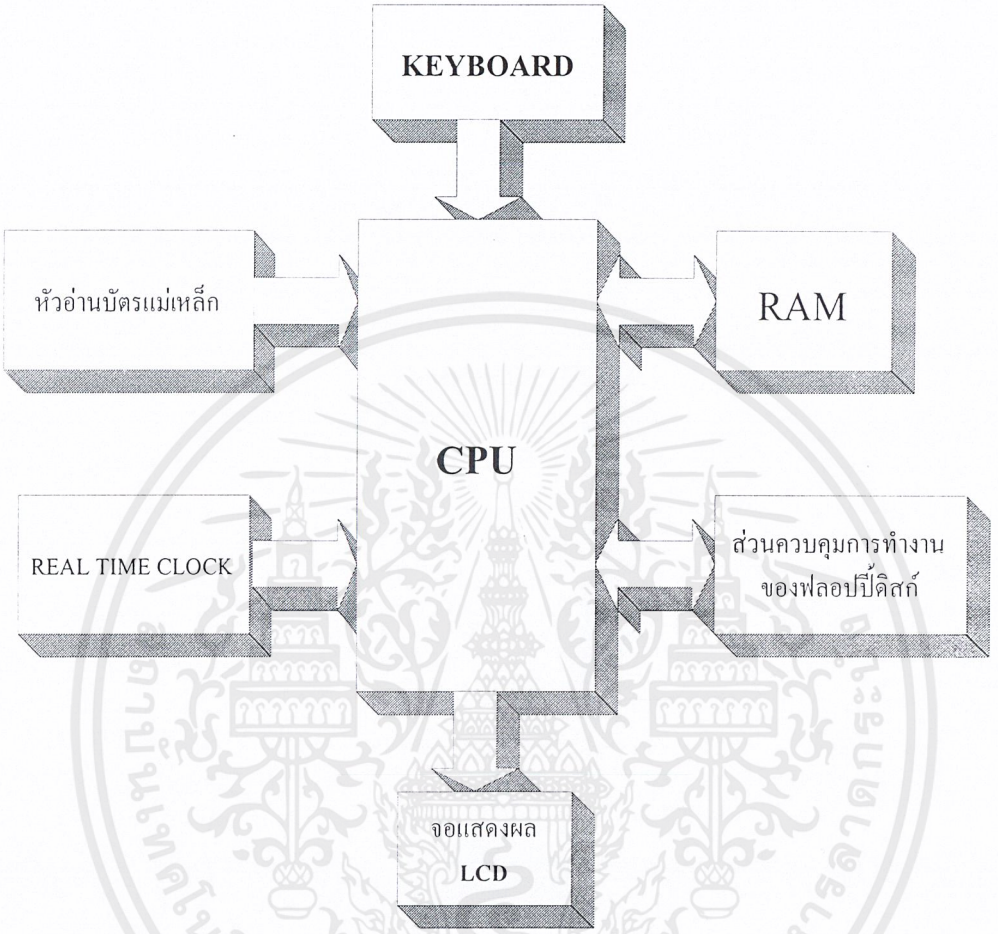
1.1 วัตถุประสงค์

สำหรับเครื่องบันทึกข้อมูลบัตรแม่เหล็กนี้ เป็นการออกแบบการจัดเก็บข้อมูลอีกแบบหนึ่งเพื่อที่จะนำไปใช้งานได้ถูกต้อง โดยมีวัตถุประสงค์ดังนี้

1. เพื่อใช้บันทึกข้อมูลจากการรูดบัตร สำหรับหน่วยงานต่างๆได้โดยบันทึกข้อมูลที่อยู่บนบัตรและวันเวลาขณะนั้น
2. สามารถบันทึกข้อมูลลงยังแผ่นฟล็อปปีดิสก์ได้โดยตรง ทำให้สะดวกกับการใช้งาน เนื่องจากหาซื้อและใช้งานได้ง่าย พกพาได้สะดวก
3. สามารถเคลื่อนย้ายเครื่องมือนี้ไปติดตั้งที่ไหนก็ได้โดยไม่ต้องต่อพ่วงกับเครื่องคอมพิวเตอร์ ทำให้เหมาะกับการเคลื่อนย้ายบ่อย ๆ ตามต้องการ
4. สามารถนำไปประยุกต์ใช้กับการบันทึกข้อมูลอื่น ๆ ได้ เช่น การบันทึกสภาพอากาศที่เวลาต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 ส่วนประกอบ



รูปที่ 1.1 แผนผังแสดงส่วนประกอบของเครื่องบันทึกข้อมูล

จากแผนผัง

1. ไมโครคอนโทรลเลอร์ ใช้ในการควบคุมการทำงานทั้งหมด คือ ควบคุมการส่งข้อมูลจาก Real Time Clock , เครื่องอ่านบัตรแม่เหล็ก , RAM , ควบคุมการอ่าน-เขียนฟล็อปปีดิสก์ , การทำงานโดยรวม และการแสดงผลทางจอ LCD
2. หัวอ่านบัตรแม่เหล็ก ใช้อ่านข้อมูลจากบัตรแม่เหล็ก แล้วส่งข้อมูลที่ได้นั้นมายังไมโครคอนโทรลเลอร์
3. Real Time clock เป็นไอซีที่ให้ระบบฐานเวลาที่ถูกต้องแก่ไมโครคอนโทรลเลอร์ โดยจะบอกถึงเวลา วัน เดือน ปี
4. คีย์บอร์ด ใช้กดเลือกและป้อนค่าต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. RAM ใช้ในการเก็บข้อมูลที่ได้จากการรูดบัตรก่อนที่จะเก็บข้อมูลนั้นลงยังฟล็อปปีดิสก์
6. วงจรควบคุมการทำงานของฟล็อปปีดิสก์ ใช้ในการควบคุมการเขียนข้อมูลบนแผ่นฟล็อปปีดิสก์ โดยจะรับเอาคำสั่งการควบคุมและข้อมูลจากไมโครคอนโทรลเลอร์เพื่อบันทึกข้อมูลจาก RAM ลงยังแผ่นดิสก์
7. จอแสดงผล LCD ใช้แสดงข้อความเพื่อให้ผู้ใช้ได้ทราบสถานะการทำงานของเครื่องและแสดงข้อมูลวัน เวลาขณะนั้น

1.3 ขอบเขตการทำงาน

ในภาคเรียนที่ผ่านมาได้ทำการศึกษาหลักการเก็บข้อมูลลงบนแผ่นดิสเกตต์ การใช้งานวงจรควบคุมการทำงานการอ่านและเขียนดิสเกตต์โดยที่ควบคุมการทำงานทั้งหมด ด้วยไมโครคอนโทรลเลอร์ในตระกูล MCS-51 แล้วแสดงผลที่ได้ทางจอ LCD

สำหรับในภาคเรียนนี้ได้ทำต่อจากภาคเรียนที่แล้ว โดยพัฒนาให้อุปกรณ์สามารถรับข้อมูลจากบัตรแม่เหล็กมาเก็บไว้ในหน่วยความจำ แล้วทำการบันทึกลงยังแผ่นฟล็อปปีดิสก์ตามต้องการ และยังมีการเขียนโปรแกรมเพื่อเปรียบเทียบข้อมูลบนบัตรกับฐานข้อมูล เมื่อนำแผ่นฟล็อปปีดิสก์ไปทำการอ่านข้อมูลที่เครื่องคอมพิวเตอร์ จะแสดงออกมาที่หน้าจอคอมพิวเตอร์ว่าผู้ที่รูดบัตรเป็นใคร ได้ทำการรูดบัตรที่วัน เวลาใด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
การบันทึกข้อมูลลงแผ่นฟล็อปปีดิสก์

2.1 โครงสร้างของดิสเกตต์

ในระบบไมโครคอมพิวเตอร์นิยมเก็บข้อมูลโดยใช้ดิสก์ไดรฟ์ 2 ชนิด ได้แก่ ฟล็อปปีดิสก์และฮาร์ดดิสก์ ความแตกต่างระหว่างดิสเกตต์ทั้งสองคือ เนื้อสารที่ใช้สร้าง ฟล็อปปีดิสก์สร้างขึ้นจากแผ่นจานไมลาร์ ซึ่งได้เคลือบด้วยสารแม่เหล็กหรือเหล็กออกไซด์ เนื่องจากโครงสร้างของตัวฟล็อปปีดิสก์เป็นไมลาร์จึงมีความอ่อนตัว เราจึงเรียกชื่อว่า ฟล็อปปีดิสก์ หรือบางทีก็เรียกว่าดิสเกตต์ ส่วนฮาร์ดดิสก์ เป็นแผ่นโลหะที่เคลือบด้วยเหล็กออกไซด์

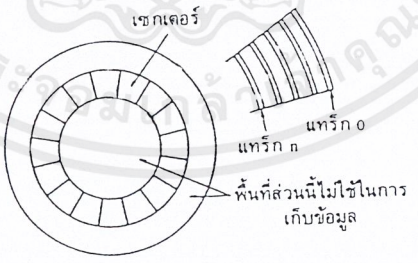
โครงสร้างของฟล็อปปีดิสก์ที่ใช้งานทั่วไปนั้นจะประกอบด้วยส่วนสำคัญ 3 ส่วน ดังนี้

ตัวแผ่นดิสก์ (magnetic disk) มีลักษณะเป็นแผ่นกลมทำด้วยสารโพลีเอสเตอร์ แล้วเคลือบด้วยสารแม่เหล็กบางๆ

ซองหุ้ม (jacket) เป็นซองสี่เหลี่ยมที่บรรจุดิสก์เอาไว้ ทำจากสารพีวีซี โดยมีหน้าที่เป็นส่วนป้องกันไม่ให้ฝุ่นละอองหรือไขมันไปทำลายผิวดิสก์ และป้องกันประจุไฟฟ้าสถิต

แผ่นทำความสะอาด (liner) เป็นแผ่นที่ทอจากไฟเบอร์ มีความหนาประมาณ 0.2 มิลลิเมตร คั่นระหว่างซองหุ้มกับตัวแผ่นดิสก์ ทำหน้าที่รักษาความสะอาดของดิสเกตต์

2.1.1 แทร็กและเซกเตอร์



รูปที่ 2.1 การจัดแทร็กและเซกเตอร์ภายในดิสเกตต์

การอ่านเขียนข้อมูลของดิสเกตต์ ทำในขณะที่ดิสเกตต์หมุนอยู่ ดังนั้นกลุ่มข้อมูลที่บรรจุอยู่ในแนวแถบกลมเรียกว่า แทร็ก (track) ดังแสดงในรูปที่ 2.1 โดยระหว่างแทร็กที่ใช้เก็บข้อมูลจะมีแทร็กที่ไม่ใช้ในการเก็บข้อมูลคั่นไว้ เพื่อลดความผิดพลาด ในการอ่านเขียนข้อมูล จำนวนแทร็กต่อความยาวตัดขวางก็เป็นการบอกความจุของดิสเกตต์วิธีหนึ่ง เช่นดิสเกตต์ขนาด 5¼ นิ้ว ที่ใช้ในไม

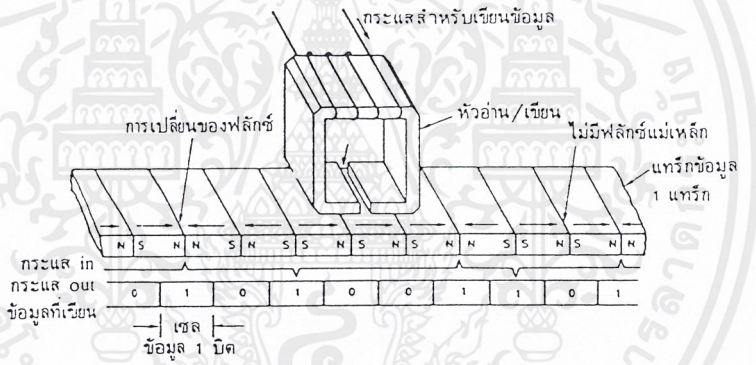
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครคอมพิวเตอร์ 16 บิต มีความจุ 48 แทร็กต่อนิ้วและมีแทร็กข้อมูล 40 แทร็ก และเนื่องจากใน 1 แทร็กสามารถเก็บข้อมูลได้มาก ดังนั้นในแต่ละแทร็กจะทำการแบ่งออกเป็นหลายเซกเตอร์ (sector) เพราะสะดวกในการค้นหาข้อมูล โดยดิสเกตต์ขนาด 5¼ นิ้ว จะมีจำนวน 8 เซกเตอร์ต่อ 1 แทร็ก โดยใน 1 เซกเตอร์มีข้อมูล 512ไบต์ ความจุของดิสเกตต์หาได้จาก

ความจุ = จำนวนหน้า × จำนวนเซกเตอร์ต่อแทร็ก × จำนวนแทร็ก × จำนวนไบต์ต่อเซกเตอร์

ดังนั้นดิสเกตต์ขนาด 3.5 นิ้วที่ใช้ในปัจจุบัน ซึ่งมี 2 หน้า หน้าละ 80 แทร็ก แทร็กละ 18 เซกเตอร์ โดยใน 1 เซกเตอร์มีข้อมูล 512 ไบต์ จะมีความจุเท่ากับ 2 × 18 × 80 × 512 = 1,474,560 ไบต์ = 1.44 เมกกะไบต์

2.1.2 การเก็บข้อมูลในแผ่นฟล็อปปีดิสก์



รูปที่ 2.2 หัวอ่านบันทึกดิสก์ ในขณะทำการบันทึกข้อมูล

การเก็บข้อมูลในฟล็อปปีดิสก์ทำได้โดยการเหนี่ยวนำสารแม่เหล็กให้เกิดขั้วแม่เหล็กบนแผ่นโดยหัวอ่านดิสก์ วิธีการอ่านเขียนฟล็อปปีดิสก์นั้น หัวอ่านจะถูกกด (loaded) จนสัมผัสกับดิสเกตต์ ซึ่งหัวอ่านเขียนจะประกอบด้วยขดลวดพันบนแกนเฟอไรต์ ซึ่งจะมีช่องอากาศ (air gap) อยู่ในช่วงที่สัมผัสกับดิสเกตต์ดังรูปที่ 2.2 การเขียนข้อมูลลงบนดิสเกตต์ทำได้โดยการเหนี่ยวนำสารแม่เหล็กบนดิสเกตต์ให้เกิดทิศทางของฟลักซ์ที่แตกต่างกันโดยวิธีการเขียนลอจิก “1” จะเขียนโดยกลับทิศทางของกระแสที่ไหลผ่านหัวอ่านเขียน เพื่อทำให้ฟลักซ์แม่เหล็กบนแผ่นมีทิศทางตรงข้ามจากเดิม ในขณะที่ลอจิก “0” จะไม่มีการกลับทิศของฟลักซ์แม่เหล็ก ในทางกลับกัน เมื่อทำการอ่านข้อมูลจากดิสเกตต์ เมื่อหัวอ่านอ่านพบข้อมูล ลอจิก “1” จะทำให้เกิดกระแสไหลตามกฎของเลนส์ เมื่อผ่านวงจรอิเล็กทรอนิกส์ เราก็สามารถอ่านค่าข้อมูลที่ถูกต้องออกมาได้ ข้อมูลที่บรรจุบนดิสเกตต์จำเป็นต้องมีการตรวจจับว่าเซกเตอร์แรกของแทร็กอยู่ที่ตำแหน่งใด ดังนั้นบนดิสเกตต์จะมี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การสร้างรูอินเด็กซ์ขึ้น ขณะที่ดิสเกตต์หมุนมาถึงตำแหน่งรูอินเด็กซ์ แสดงว่าเป็นตำแหน่งของเซกเตอร์แรกของทุกแทร็ก โดยการตรวจจับจะกระทำโดยตัวดิสก์ไดรฟ์

2.1.3 การมอดูเลชันของข้อมูล

การเก็บข้อมูลมีความจำเป็นที่จะต้องทำการมอดูเลชัน เพื่อลดความผิดพลาดในการอ่านเขียนข้อมูล เนื่องจากความเร็วของดิสเกตต์ไม่คงที่ และเพื่อที่จะเพิ่มความจุในการเก็บข้อมูลของดิสเกตต์ ซึ่งหลักการของการมอดูเลชันที่ใช้ในฟล็อปปีดิสก์ ทำโดยการผสมสัญญาณนาฬิกากับข้อมูล

วิธีการมอดูเลชันที่ใช้ในฟล็อปปีดิสก์ มีหลายวิธีดังนี้

FM (frequency modulation) วิธีการนี้ใช้การบันทึกข้อมูลลงบนดิสเกตต์ แบบความหนาแน่นเท่าเดียว (single density) หลักการมอดูเลชัน ทำโดยการใส่ข้อมูลระหว่างสัญญาณนาฬิกา ดังนั้นในการดีมอดูเลชัน ก็กระทำโดยอ่านข้อมูลในช่วงสัญญาณนาฬิกา

การมอดูเลชันแบบ FM จะมีอัตราการบันทึกต่ำ เพราะอัตราการบันทึกข้อมูลขึ้นกับจำนวนครั้งของการเปลี่ยนแปลงของฟลักซ์ต่อวินาที และในขณะที่ทำการส่งลอจิก "1" ข้อมูลที่ถูกมอดูเลชัน จะมีอัตราการเปลี่ยนแปลงของฟลักซ์สูงกว่าขณะทำการส่งลอจิก "0" ดังนั้นสัญญาณนาฬิกาหรืออัตราการส่งข้อมูลจะต้องมีค่าเพียงครึ่งหนึ่งของอัตราการบันทึกข้อมูลของดิสเกตต์

MFM (modified frequency modulation) วิธีการนี้ใช้ในการบันทึกข้อมูลลงบนดิสเกตต์แบบความหนาแน่น 2 เท่า (double density) ซึ่งพัฒนาหลักการมาจากกวิธี FM วิธี MFM จะพยายามกำจัดสัญญาณนาฬิกาที่ไม่จำเป็นออกไป โดยมีหลักการของการมอดูเลชัน ดังนี้

1. เมื่อบันทึกข้อมูล ลอจิก "1" ทำการสร้างพัลส์ ในช่วงตรงกลางของบิตเซตเหมือนวิธี FM แต่ตัดสัญญาณนาฬิกาออก
2. เมื่อบันทึกลอจิก "0" ทำการส่งสัญญาณนาฬิกา ยกเว้นเมื่อข้อมูลก่อนหน้าเป็นลอจิก "1"

วิธีการมอดูเลชันแบบ MFM จะได้อัตราการส่งข้อมูลมีค่าเท่ากับการบันทึกข้อมูลของฟล็อปปีดิสก์

M^2FM (modified modified frequency modulation) เป็นวิธีที่ดัดแปลงจากวิธี MFM เล็กน้อยคือ ในกรณี M^2FM จะส่งสัญญาณนาฬิกา เมื่อข้อมูลที่ส่งเป็น ลอจิก "0" ติดต่อกัน 2 บิตขึ้นไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 การจัดข้อมูลในดิสก์

ในการจัดข้อมูลลงในดิสก์นั้น เราจะแบ่งเก็บเป็นกลุ่มที่มีลักษณะอย่างเดียวกันหรือเกี่ยวข้องกับเรื่องเดียวกัน เรียกว่า ไฟล์ (file) เหมือนกับการที่เราจัดเอกสารเป็นแฟ้มนั่นเอง ข้อมูลที่เก็บในไฟล์จะเป็นได้ทั้งตัวเลขต่าง ๆ โปรแกรม ข้อความ ฯลฯ ไฟล์เหล่านี้จะถูกจัดเก็บลงในดิสก์ตามวิธีการของ DOS ไม่ว่าจะเป็นฮาร์ดดิสก์หรือฟล็อปปีดิสก์ โดยแบ่งข้อมูลในแต่ละไฟล์ออกเป็นส่วนย่อยที่มีขนาดคงที่ เพื่อบันทึกลงในเนื้อที่แต่ละเซกเตอร์ของดิสก์ซึ่งมีความจุข้อมูลคงที่ เช่น 512 ไบต์ หรือ ½ กิโลไบต์

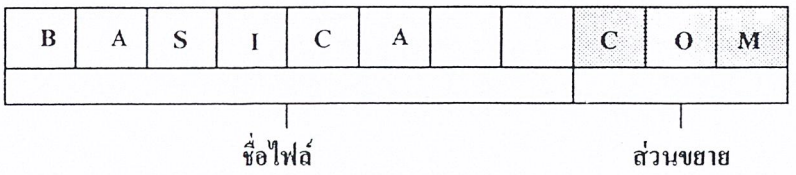
เมื่อจะมีการติดต่อหรือเข้าถึงข้อมูลในดิสก์ ไม่ว่าจะเป็นการอ่านหรือบันทึกลงไปก็ตาม ก็จะต้องมาค้นหาข้อมูลในดิสก์ เพื่อดูว่าไฟล์ที่ระบุนั้นอยู่บนดิสก์แผ่นนั้นหรือไม่ ถ้ามีจะอยู่ตรงไหนบ้าง โดยที่จะต้องมีการเก็บข้อมูลเพิ่มเติมอีกสองส่วน คือจะต้องมีสารบัญหรือไดเรกทอรี (Directory) ที่จะบอกว่าบนแผ่นดิสก์นั้นมีไฟล์อะไรบ้าง และมีตารางที่เรียกว่า ตารางการจัดไฟล์ หรือ File Allocation Table (FAT) ซึ่งจะบอกว่าเซกเตอร์ไหนเก็บไฟล์ไหนและเรียงลำดับกันอย่างไร ทั้งไดเรกทอรีและ FAT นี้จะเก็บเป็นเซกเตอร์แรก ๆ บนแผ่นดิสก์ก่อนที่จะถึงข้อมูลจริง

2.2.1 ไดเรกทอรี (Directory)

ในไดเรกทอรีของดิสก์แต่ละแผ่นจะเก็บข้อมูลของแต่ละไฟล์เรียงเป็นรายการ ๆ ลงมา แต่ละรายการยาว 32 ไบต์ ดังนั้น ใน 1 เซกเตอร์ คือ 512 ไบต์จึงมีที่เก็บไดเรกทอรีได้ 16 รายการหรือ 16 ไฟล์ โดยแต่ละรายการของไดเรกทอรีประกอบด้วยข้อมูลต่าง ๆ เรียงตามลำดับดังนี้

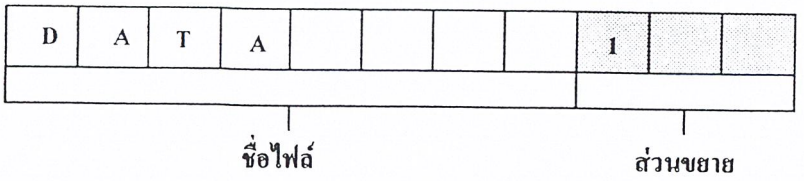
1. ชื่อไฟล์และส่วนขยาย

ชื่อไฟล์ใน DOS ประกอบด้วยสองส่วนคือ ชื่อไฟล์ซึ่งยาว 8 ตัวอักษร และส่วนขยาย (extension) อีก 3 ตัวอักษร รวมเป็น 11 ตัวอักษรหรือ 11 ไบต์ ซึ่งเวลาเขียนโดยทั่วไปเราจะใส่จุดคั่นระหว่างทั้งสองส่วนออกจากกัน เช่น BASICA.COM เป็นต้น โดยจะเก็บชื่อไฟล์ในไดเรกทอรีให้ชิดซ้าย และเติมที่ว่าง (blank) ให้จนเต็มความยาว ดังนั้น ชื่อไฟล์จะเก็บเป็นดังนี้



หรือถ้าส่วนขยายมีความยาวไม่ถึง 3 ตัวอักษร ก็จะถูกเติม blank เข้าไปข้างท้ายเช่นกัน เช่น DATA.1 จะเป็น

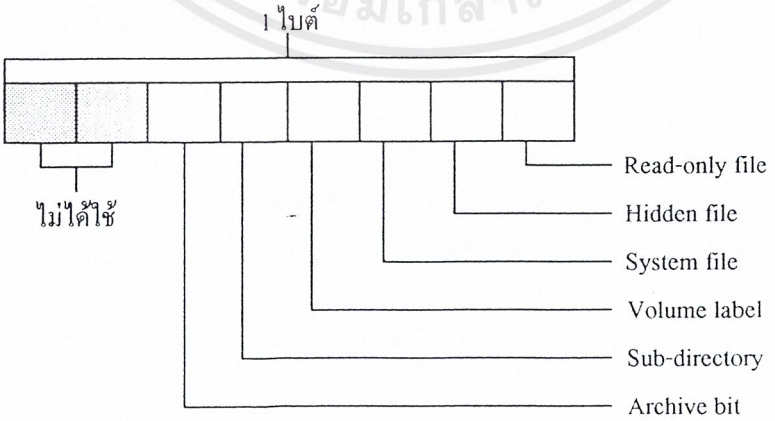
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



แต่ละรายการในไดเรกทอรีจะถูกเติมเข้าไปเมื่อมีการสร้างไฟล์นั้นขึ้นเป็นครั้งแรกบนดิสก์ หรือที่เรียกว่า การ Create file ซึ่งอาจทำโดยโปรแกรมใช้งานต่าง ๆ เช่น อิเล็กทรอนิกส์สเปรดชีต เวิร์ดโปรเซสเซอร์ หรือทำโดยคำสั่งใน DOS เอง เช่น COPY ข้อมูลไปยังไฟล์ที่ไม่มีอยู่ก่อนบนดิสก์ก็จะมีการสร้างไฟล์นั้นขึ้นมา โดยเติมชื่อไฟล์ ส่วนขยาย และข้อมูลอื่น ๆ เช่น แอตทริบิวต์ ซึ่งจะอธิบายต่อไป วันและเวลาที่มีการสร้างหรือแก้ไขไฟล์นั้นเป็นครั้งสุดท้าย เป็นต้น ในทางตรงกันข้าม เมื่อไฟล์ถูกลบออกจากแผ่นดิสก์ ข้อมูลของไฟล์นั้นในไดเรกทอรีจะคงอยู่ เพียงแต่ว่าจะมีการทำเครื่องหมายไว้ที่อักษรตัวแรกของชื่อไฟล์ด้วยเลขฐานสิบหก คือ E5 (รหัส ASCII 229) ซึ่งรหัสนี้ไม่ตรงกับอักษรตัวใด ๆ ที่จะนำมาตั้งเป็นชื่อไฟล์แบบธรรมดาได้ ดังนั้นเมื่อพบเครื่องหมายนี้ ก็จะรู้ว่าไฟล์นั้นได้ถูกลบไปแล้ว ข้อมูลสุดท้ายของไฟล์นั้น ๆ ในไดเรกทอรีจะคงอยู่จนกว่าจะนำที่ตรงนั้นมาใช้ใหม่โดยบันทึกชื่อไฟล์อื่น ๆ ที่ถูกสร้างขึ้นใหม่ภายหลังทับลงไป ซึ่งถ้ายังไม่ถูกทับก็เป็นไปได้ที่เราจะใช้โปรแกรมพิเศษบางตัวเรียกข้อมูลในไดเรกทอรีรายการนั้น ๆ มาดูและแก้ไขกลับเป็นอย่างเก่าเพื่อเรียกไฟล์ที่ถูกลบไปแล้วกลับคืนมา

2. แอตทริบิวต์ (Attribute)

เป็นข้อมูลที่มีความยาว 1 ไบต์ คือ 8 บิต ใช้กำหนดคุณสมบัติของไฟล์นั้น ๆ ว่าจะเป็นไฟล์ธรรมดาหรือจะเป็นไฟล์ที่มีลักษณะพิเศษบางอย่าง คุณสมบัติที่กำหนดโดยแต่ละบิตเหล่านี้จะเกิดขึ้นเมื่อบิตนั้น ๆ ถูกกำหนดค่าเป็น 1 ถ้าเป็น 0 ก็จะไม่มีคุณสมบัตินั้น



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **Read-only file** เป็นไฟล์ที่อ่านข้อมูลออกมาได้เพียงอย่างเดียว แก้ไขหรือบันทึกข้อมูลเพิ่มไม่ได้

- **Hidden file** เป็นไฟล์ที่ถูกซ่อนไว้ เมื่อ DOS ค้นหาชื่อไฟล์ต่าง ๆ ในไดเรกทอรีมันข้ามชื่อไฟล์ที่ถูกกำหนดให้ซ่อนไว้ไป

- **System file** เป็นไฟล์ของระบบซึ่งจะถูกข้ามไปในการค้นหาชื่อไฟล์ในไดเรกทอรีเช่นเดียวกับ Hidden file

แอตทริบิวต์ทั้ง 3 อย่างนี้ใช้ร่วมกันได้ คือ ไฟล์หนึ่ง ๆ ที่อาจเป็นได้ทั้ง Read-only,Hidden และ System file พร้อม ๆ กันก็ได้ เช่นในกรณีของไฟล์ IO.SYS และ MSDOS.SYS เป็นไฟล์ที่เก็บ DOS เอง เป็นต้น ส่วนแอตทริบิวต์อีก 3 อย่างที่เหลือนั้นเป็นคุณสมบัติที่เป็นคนละเรื่องกันออกไป คือ

- **Volume Label** แสดงว่ารายการนั้นในไดเรกทอรีไม่ได้เก็บชื่อไฟล์และส่วนขยาย แต่เก็บชื่อฮาร์ดดิสก์หรือแผ่นดิสเกตต์นั้น ซึ่งชื่อแผ่นนี้จะยาวได้สูงสุด 11 ตัวอักษร เท่ากับชื่อไฟล์และส่วนขยายรวมกันนั่นเอง

- **Sub-directory** ไดเรกทอรีย่อยนี้จะเก็บรายชื่อไฟล์ได้อีกเช่นเดียวกับไดเรกทอรีราก (Root Directory) แต่ไฟล์เหล่านั้นจะไม่ปรากฏในชื่อไดเรกทอรีราก มีแต่ชื่อไดเรกทอรีย่อยเพียงชื่อเดียวไปปรากฏอยู่แทนเพียง 1 รายการต่อ 1 ไดเรกทอรีย่อยเท่านั้น ดังนั้น รายการที่เป็นชื่อของไดเรกทอรีย่อยนี้จะถูกดอสข้ามไปในรายการปกติ นอกจากในคำสั่ง DIR เมื่อดูไดเรกทอรีจึงจะเห็นชื่อของไดเรกทอรีย่อยพร้อมสัญลักษณ์ <DIR> กำกับอยู่ ส่วนในกรณีที่ต้องการอ้างถึงหรือทำงานกับชื่อไฟล์ในไดเรกทอรีย่อย จะต้องใช้คำสั่งย้ายไดเรกทอรี (CHDIR) ก่อน หรือไม่ก็ระบุชื่อไดเรกทอรีย่อยนั้นกำกับชื่อไฟล์ไปด้วยเลย

- **Archive bit** เป็นบิตที่ใช้ในการทำสำเนาข้อมูลจากฮาร์ดดิสก์มายังฟล็อปปีดิสก์ และการเรียกกลับคืนที่ทำตรงข้ามกัน บิตนี้ถูกใช้เป็นที่ทำเครื่องหมายบอกว่า ตั้งแต่การทำการแบ็คอัพครั้งสุดท้ายแล้ว ไฟล์นั้นถูกแก้ไขไปบ้างหรือเปล่า จะได้รู้ว่าต้องทำการแบ็คอัพไฟล์นี้ใหม่หรือไม่ในครั้งถัดไป สำหรับบิตนี้จะสามารถใช้ร่วมกับแอตทริบิวต์อื่น ๆ ได้ด้วย

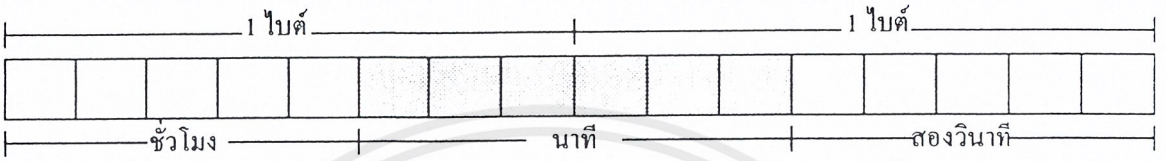
3. Reserved

ส่วนนี้คอสเตรียมเผื่อไว้สำหรับการใช้งานภายในโปรแกรมเอง หรือเผื่อไว้สำหรับการเพิ่มข้อมูลที่เก็บในแต่ละรายการของไดเรกทอรีในอนาคต ยาว 10 ไบต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

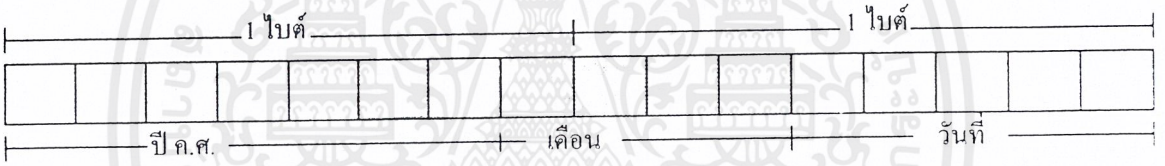
4. เวลา

เก็บเวลาที่สร้างหรือแก้ไขข้อมูลในไฟล์นั้นครั้งสุดท้ายตามนาฬิกาภายในเครื่อง รูปแบบของเวลาในไดเรกทอรีเก็บรายละเอียดถึงทุก 2 วินาที แต่เมื่อดูจากคำสั่ง DIR จะเห็นแค่รายละเอียดถึงระดับนาทีเท่านั้น เวลานี้จะเก็บในเนื้อที่ 2 ไบต์หรือ 16 บิต และเริ่มมีใช้ตั้งแต่ดอสเวอร์ชัน 1.1 เป็นต้นมา โดยเก็บข้อมูลเป็นเลขฐานสองตามรูปแบบดังนี้



5. วันที่

เก็บวันที่ที่สร้างหรือแก้ไขไฟล์นั้นเป็นครั้งสุดท้าย จากนาฬิกาภายในเครื่องเช่นเดียวกับเวลา โดยเก็บข้อมูลในเนื้อที่ 2 ไบต์หรือ 16 บิต เป็นเลขฐานสองตามรูปแบบดังนี้



ปี ค.ศ. จะเก็บตัวเลขตั้งแต่ 0 ถึง 119 เพื่อนำไปบวกด้วย 1980 ดังนั้น ดอสจะเก็บวันที่ได้ตั้งแต่ปี ค.ศ. 1980 ถึง 2099

6. คลัสเตอร์แรกของไฟล์

ใช้เนื้อที่ 2 ไบต์เก็บเลขที่ของคลัสเตอร์ในรูปแบบเลขฐานสอง บอกว่าไฟล์ที่มีชื่อในรายการนั้นเริ่มเก็บข้อมูลส่วนแรกที่คลัสเตอร์ไหน โดยคลัสเตอร์ (Cluster) หรือกลุ่ม เป็นหน่วยที่ดอสใช้ในการจัดที่ให้แต่ละไฟล์บนดิสก์ ประกอบด้วย 1 ถึง 4 เซกเตอร์หรือมากกว่านั้น แล้วแต่ชนิดของดิสก์ การอ้างถึงตำแหน่งที่เก็บข้อมูลก็อ้างถึงเลขที่คลัสเตอร์แทน แต่ละคลัสเตอร์จะมีจำนวนเซกเตอร์คงที่ (แต่ถ้าเป็นดิสก์คนละแบบอาจไม่เท่ากัน) ดังนั้น เวลาอ่านหรือเขียนข้อมูลจึงต้องนำเลขที่คลัสเตอร์ไปแปลงเป็นเลขที่เซกเตอร์อีกทีหนึ่ง การที่ต้องรวมหลายเซกเตอร์เป็น 1 คลัสเตอร์ก็เพื่อไม่ให้มีจำนวนชิ้นย่อย ๆ ของข้อมูลที่ไม่ต่อเนื่องกันบนดิสก์มากจนเกินไปจนยุ่งยากในการจัด File Allocation Table ซึ่งจะเป็นตัวบอกว่าข้อมูลที่เหลือของแต่ละไฟล์ว่าอยู่ที่ใดบ้าง โดยบอกเป็นเลขที่ของคลัสเตอร์เหมือนกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. ขนาดของไฟล์

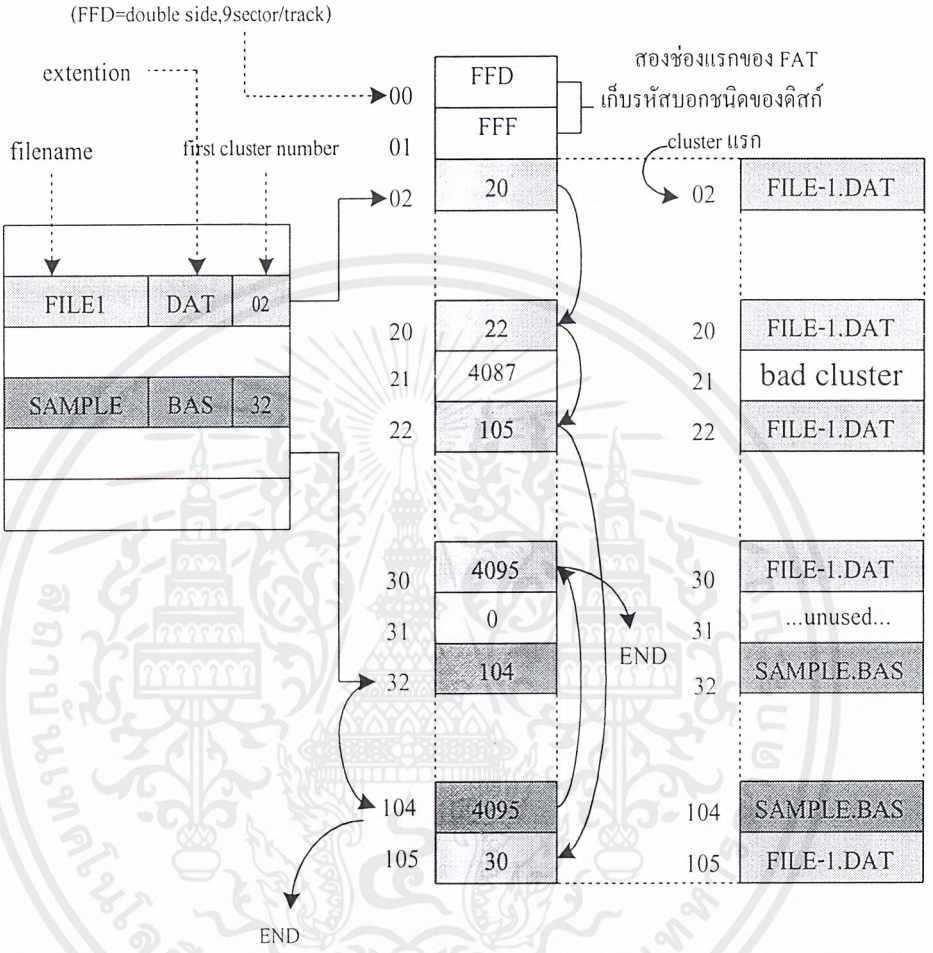
เก็บขนาดของไฟล์เป็นไบต์ในรูปแบบเลขฐานสอง 32 บิต ตัวเลขบอกขนาดนี้จะเก็บโดยดอสซึ่งจะเชื่อถือได้แน่นอนก็เฉพาะเมื่อเป็นไฟล์ที่เก็บโปรแกรมภาษาเครื่องที่มีส่วนขยายเป็น .COM หรือไฟล์ที่เกิดจากการบันทึกข้อมูลแต่ละเรคคอร์ดที่มีขนาดเท่า ๆ กันเท่านั้น เพราะทั้งสองแบบนี้จะทำให้ดอสติดตามขนาดของไฟล์จริง ๆ ได้อย่างถูกต้องทุกไบต์หรือทุกตัวอักษร แต่ในกรณีของไฟล์ข้อความ (Text file) เช่น จดหมายหรือเอกสารที่สร้างโดยโปรแกรมเวิร์ดโปรเซสเซอร์ ซึ่งอาจจะอ่านหรือบันทึกข้อมูลไม่ใช่ทีละไบต์ (เช่น ทีละ 128 หรือ 256 ไบต์) ซึ่งเวลาบันทึกอาจจะเกินท้ายข้อมูลจริงไปบ้าง เพราะไฟล์แบบนี้จะมีรหัสบอกตำแหน่งสุดท้ายของไฟล์ (end-of-file marker) กำกับอยู่ตรงตำแหน่งสุดท้ายอยู่แล้ว เวลาอ่านข้อมูลกลับไปจึงไม่มีปัญหา แต่จะไม่สามารถรู้ได้ว่า end-of-file marker นั้นอยู่ตรงไหน มันจะรู้จักแต่ว่าในการบันทึกข้อมูลนั้นได้บันทึกไปถึงตำแหน่งไหน และถือว่าไฟล์สิ้นสุดแค่นั้น ดังนั้น ตัวเลขบอกขนาดไฟล์จึงอาจเกินจาก end-of-file ไปบ้าง

2.2.2 ตารางการจัดไฟล์ (File Allocation Table-FAT)

แต่ละรายการในไดเรกทอรี จะมีข้อมูลตัวหนึ่งที่บอกว่าคลัสเตอร์แรกของไฟล์นั้นๆ อยู่ที่ไหน แต่คลัสเตอร์อื่นๆ นอกจากนั้นจะต้องดูจาก FAT ซึ่งมีลักษณะเป็นตาราง แต่ละช่องของตารางจะจับคู่กับแต่ละคลัสเตอร์บนดิสก์ได้พอดี เช่น ตาราง FAT ช่องที่ 10 จะคู่กับ คลัสเตอร์ที่ 10, ช่องที่ 100 คู่กับคลัสเตอร์ที่ 100 ทีนี้การจะเก็บข้อมูลว่าคลัสเตอร์ไหนเก็บข้อมูลของไฟล์เดียวกัน เราก็โยงแต่ละช่องของ FAT เข้าด้วยกัน ตัวเลขที่เก็บในแต่ละช่องของ FAT จะบอกว่า คลัสเตอร์ถัดไปที่เก็บข้อมูลของไฟล์นั้น เลขที่เท่าไหร่ ดังตัวอย่างต่อไปนี้

ไฟล์ FILE-1.DAT เก็บในคลัสเตอร์ที่ 2 , 20, 22, 105 และ 30 เรียงตามลำดับ ช่องที่ 20 ของ FAT (คู่กับคลัสเตอร์ที่ 20) ก็จะเก็บข้อมูล 22 ช่องที่ 22 ก็จะเก็บข้อมูล 105 ช่องที่ 105 เก็บข้อมูล 30 ส่วนที่ช่องที่ 30 เป็นคลัสเตอร์สุดท้าย ไม่รู้จะชี้ไปไหนต่อก็เก็บข้อมูล 4095 (หรือ FFF ฐานสิบหก ซึ่งเป็นเลขที่คลัสเตอร์สูงสุดที่จะเป็นไปได้ เพราะเลขที่คลัสเตอร์ใช้เนื้อที่เก็บ 12 บิต ดังนั้นเลขสูงสุดที่เก็บได้คือ $2^{12} - 1 = 4,095$ ซึ่งไม่มีดิสเกตต์แผ่นใดมีจำนวนคลัสเตอร์มากถึงเพียงนั้น) แสดงว่าเป็นคลัสเตอร์สุดท้าย แต่การที่จะบอกว่าคลัสเตอร์เหล่านี้เป็นของไฟล์ FILE-1.DAT ก็จะต้องเก็บค่า 2 ด้วย ดังรูปที่ 2.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 File Allocation Table (FAT) ของดิสก์แบบสองหน้า 9 เซกเตอร์/แทร็ก แสดงข้อมูลของไฟล์ FILE1.DAT ซึ่งใช้ที่ในคลัสเตอร์ที่ 30 ยังไม่หมด ส่วนไฟล์ SAMPLE.BAS กินที่เต็มสองคลัสเตอร์ ขณะที่คลัสเตอร์ 31 ว่างอยู่และคลัสเตอร์ 21 เสีย

เมื่อ DOS อ่านข้อมูลจากไฟล์ในดิสก์ มันจะเริ่มจากการค้นหาชื่อไฟล์นั้นในไดเรกทอรี และดูว่าคลัสเตอร์แรกอยู่ที่ไหน แล้วค่อยมาดูที่ FAT ว่าคลัสเตอร์ใดอยู่ถัดไปบ้างไล่ไปเรื่อยๆ จนกว่าจะจบไฟล์ เลขที่คลัสเตอร์ที่ได้ในแต่ละครั้งจะถูกแปลงเป็นเลขที่เซกเตอร์ เพื่อสั่งให้ดิสก์ไดรฟ์อ่านข้อมูลออกมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการใช้งานดิสก์ สิ่งที่อาจเกิดขึ้นได้คือมีบางบริเวณบนแผ่นที่มีความผิดปกติ ทำให้ใช้บันทึกข้อมูลไม่ได้ ซึ่งจะตรวจพบในระหว่างการใช้คำสั่ง FORMAT เพื่อจัดเตรียมพื้นที่บนดิสก์นั้น ซึ่งถ้าพบว่าเซกเตอร์ใดผิดปกติ จะมีการกำหนดให้คลัสเตอร์ทั้งคลัสเตอร์ที่มีเซกเตอร์เสียนั้น (ถึงแม้จะเสียเพียง 1 ใน 2 เซกเตอร์ก็ตาม) เป็นคลัสเตอร์ที่ใช้ไม่ได้และจัดการใส่ค่า 4087 (FF7 ฐานสิบหก) ในช่องของ FAT ที่คู่กับคลัสเตอร์นั้น ส่วนคลัสเตอร์อื่นๆ ที่ใช้การได้จะถูกใส่ค่าใน FAT ที่คู่กับคลัสเตอร์เป็น 0 เพื่อแสดงว่าเป็นคลัสเตอร์ว่างที่ยังไม่ใช้ นอกจากนี้ค่าตั้งแต่ 4088 ถึง 4095 ก็ใช้เป็นตัวบอก end-of-file คือบอกว่าเป็นคลัสเตอร์สุดท้ายได้เช่นกัน

แต่ละช่องของ FAT จะใช้ที่ 12 บิต หรือหนึ่งไบต์ครึ่งดังที่กล่าวมาแล้ว สองช่องของ FAT จะจับคู่กันใช้ที่ 3 ไบต์ ดังนั้นเนื้อที่ 1 เซกเตอร์บนดิสก์ จึงพอสำหรับเก็บ FAT ได้ 512 * 2/3 หรือ 340 ช่อง สำหรับดิสเกตต์ตามฟอร์แมตของ DOS 1.0 มี 313 คลัสเตอร์ และฟอร์แมตของ DOS 1.1 มี 314 คลัสเตอร์ ซึ่งสามารถเก็บลงใน 1 เซกเตอร์ของ FAT ได้ ดังนั้น FAT จึงมีขนาดเพียง 1 เซกเตอร์ ส่วนในกรณีของ DOS 2.0 นั้น ขนาดของ FAT ได้ขยายขึ้นเป็น 2 เซกเตอร์ เพื่อให้พอกับจำนวนคลัสเตอร์ที่เพิ่มขึ้นตามจำนวนเซกเตอร์ต่อแทร็กที่เพิ่มขึ้นจาก 8 เป็น 9 และใน DOS 3.0 ขึ้นไปนั้น FAT สำหรับดิสเกตต์แบบ high capacity ซึ่งมี 15 เซกเตอร์ต่อแทร็ก FAT ได้เพิ่มขึ้นเป็น 7 เซกเตอร์ ส่วนในฮาร์ดดิสก์นั้น ถ้ามีความจุข้อมูลเกินกว่า 16 เมกกะไบต์ขึ้นไป แต่ละช่องของ FAT จะใช้ที่ 16 บิต หรือ 2 ไบต์เพื่อให้มีจำนวนคลัสเตอร์ได้มากที่สุดถึง 216 หรือ 65,536 คลัสเตอร์ เพราะ FAT 12 บิตบนฮาร์ดดิสก์จะจัด 1 คลัสเตอร์ = 8 เซกเตอร์ = 4 กิโลไบต์ ดังนั้น 212 หรือ 4,096 คลัสเตอร์จึงเก็บข้อมูลได้ไม่เกิน 4,096 × 4 = 16,384 กิโลไบต์ หรือ 16 เมกกะไบต์ เท่านั้น

เนื่องจาก FAT เป็นข้อมูลที่มีความสำคัญมากในการทำงานกับดิสก์ DOS จึงเก็บ FAT ไว้ถึง 2 ชุดบนดิสก์แต่ละแผ่น เผื่อว่า FAT เสียไปชุดหนึ่งก็ยังสามารถนำข้อมูลจากอีกชุดหน่งมาใช้แก้ไขให้กลับดีได้ ตอนต้นของ FAT จะมีช่องพิเศษสองช่อง ช่องแรกจะเก็บรหัสที่จะบอกว่าดิสก์แผ่นนั้นเป็นแบบใด เพื่อว่าเมื่อ DOS อ่าน FAT เข้าไปจะได้รู้ว่าภายในดิสก์นั้นมีการจัดข้อมูลแบบใด จะได้อ่านหรือบันทึกข้อมูลได้ถูกต้อง

ช่องถัดมาของ FAT จะเก็บข้อมูล FFF ฐานสิบหก หรือ 4095 ในฐานสิบ ส่วนช่องที่ใช้บอกคลัสเตอร์ที่ใช้เก็บข้อมูลจริงนั้นจะเริ่มตั้งแต่ช่อง 02 ดังนั้นการอ้างถึงเลขที่คลัสเตอร์จึงเริ่มตั้งแต่ 02 เป็นต้นไปด้วย

เมื่อพิจารณาการจัดเนื้อที่บนดิสก์ทั้งหมดแล้ว จะเห็นได้ว่าข้อมูลจะแบ่งเป็น 4 กลุ่มใหญ่ๆ คือ ไดเรกทอรี, FAT และข้อมูลจริง นอกจากนี้ยังมี Boot record ซึ่งเป็นโปรแกรมสั้นๆ กินเนื้อที่ไม่เกิน 1 เซกเตอร์ (512 ไบต์) ซึ่งจะต้องถูกจัดที่ให้โดยเฉพาะเป็นเซกเตอร์แรกของดิสก์ทุกแผ่นใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบ DOS อีกด้วย (Boot record นี้จะเก็บโปรแกรมที่จะเป็นตัวอ่าน DOS เข้าไปในเครื่องและยังเก็บข้อมูลเกี่ยวกับโครงสร้างของดิสก์โดยละเอียดอีกด้วยว่า ในดิสก์นั้นมีกี่เซกเตอร์, ขนาดกี่ไบต์, ในคลัสเตอร์หนึ่งมีกี่เซกเตอร์, จำนวนไฟล์สูงสุดในไดเรกทอรีราก ฯลฯ ในส่วนที่เรียกว่า BIOS Parameter Block หรือ BPB ดังนั้น DOS จึงรู้จักลักษณะของดิสก์ได้จากข้อมูลบน BPB ใน boot record นี้เอง) ข้อมูลบนดิสก์ทุกแผ่นในระบบ DOS จะเป็นดังในตาราง ซึ่งจำนวนเซกเตอร์ทั้งหมดเมื่อรวมกันจะต้องได้เต็มตามความจุของดิสก์แต่ละแบบ

ตารางที่ 2.1 จำนวนแทร็กและเซกเตอร์ในดิสเกตต์แบบต่างๆ

ชนิดของดิสเกตต์	160 KB	320 KB	360 KB	1.2 MB	720 KB	1.44 MB	2.88 MB
เวอร์ชันของ DOS	1.0	1.1	2.0	3.0	3.2	3.3	5.0
ขนาดดิสก์ (นิ้ว)	5.25	5.25	5.25	5.25	3.5	3.5	3.5
sector/track	8	8	9	15	9	18	36
จำนวน track	40	40	40	80	80	80	80
sector ทั้งหมด	320	640	720	2,400	1,440	2,880	5,760
Boot record	1	1	1	1	1	1	1
FAT** (sector/ชุด)	1	2	2	7	3	9	n a
ไดเรกทอรี (เซกเตอร์)	4	7	7	14	7	14	n a
ข้อมูลจริง (เซกเตอร์)	313	630	708	2,371	1,426	2,847	n a
(คลัสเตอร์)	313	315	354	2,371	713	2,847	n a
เซกเตอร์/คลัสเตอร์	1	2	2	1	2	1	n a

*ดิสก์หน้าเดียว

**ดิสก์เกตต์ทุกแบบที่ใช้ FAT แบบ 12 บิต

การจัดข้อมูลในดิสก์แบบนี้ใช้ได้ทั้งกับฟล็อปปีดิสก์ทุกแบบ ตลอดจนถึงฮาร์ดดิสก์หรือฟิกซ์ดิสก์ หรือแม้แต่เวอร์ชวลดิสก์ จะผิดกันก็แต่ในรายละเอียด เช่น จำนวนไฟล์สูงสุดที่มีได้และจำนวนเซกเตอร์ที่ใช้เก็บไดเรกทอรีราก จำนวนช่องและขนาดรวมทั้งหมดของ FAT เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่หลักการนั้นเหมือนกัน สำหรับในฮาร์ดดิสก์นั้นจะมี Master Boot Record เพิ่มเข้ามาด้วย สำหรับเก็บข้อมูลการแบ่งพาร์ติชันของฮาร์ดดิสก์นั้น ๆ ว่าเริ่มตรงไหน (หรือที่เราเรียกว่าเป็น " ตารางการแบ่งพาร์ติชัน " หรือ partition table นั่นเอง) จากนั้น เมื่อเข้าสู่แต่ละพาร์ติชันแล้วจึงจะมี Boot record , FAT , ไดเรกทอรี และข้อมูลจริงตามลำดับข้างต้นอีกทีหนึ่ง

2.3 ฟล็อปปีดิสก์คอนโทรลเลอร์

สำหรับโครงงานนี้ได้ใช้ฟล็อปปีดิสก์คอนโทรลเลอร์มาใช้ในการควบคุมการทำงานของดิสก์ไดรฟ์เบอร์ DP 8473 ซึ่งมีคุณลักษณะดังนี้

- สามารถใช้งานร่วมกับ uPD765A และ IBM-BIOS ได้
- ใช้คริสตอลออสซิลเลเตอร์ ขนาด 24 เมกะเฮิร์ต
- สามารถใช้งานแบบ DMA ได้
- มีบัฟเฟอร์ต่อกับบัสไมโครโปรเซสเซอร์ ขนาด 12 มิลลิแอมป์
- มีบัฟเฟอร์ส่วนเชื่อมต่อกับฟล็อปปีดิสก์ไดรฟ์ขนาด 48 มิลลิแอมป์
- ใช้กับการถอดรหัสแอดเดรส A0 – A2 ของ IBM ได้
- มีรีจิสเตอร์แสดงอัตราการรับส่งข้อมูลและรีจิสเตอร์ควบคุมไดรฟ์
- มีการปรับเฟสล็อคลูปและดีเลย์ด้วยตัวเอง
- สามารถเลือกใช้ฟิลเตอร์หนึ่งในสามแบบได้โดยอัตโนมัติ
- มีรูปแบบวิธีการอ่านข้อมูลที่ดี
- มี 2 ขาที่สามารถโปรแกรมโหมดการชดเชยได้
- สามารถตั้งค่าอัตราการส่งข้อมูลได้สูงถึง 1 เมกะบิตต่อวินาทีได้
- การค้นหาแบบอัตโนมัติได้สูงจนถึง 4000 แทร็ก
- ใช้ได้กับรูปแบบของ IBM หรือ ISO
- เป็นแบบ CMOS กินกำลังไฟต่ำ และมีโหมดประหยัดพลังงาน

2.3.1.องค์ประกอบ

1. 765A compatible micro-engine

การทำงานหลัก ๆ จะเป็นแบบเดียวกับ μPD765A ซึ่งจะประกอบด้วย sequencer, program ROM ,disk/misc register ซึ่งจะใช้สัญญาณนาฬิกา 4MHz,4.8 MHz หรือ 8 MHz แล้วแต่จะเลือกในรีจิสเตอร์แสดงอัตราของข้อมูลซึ่งขึ้นอยู่กับการใช้งาน คอนโทรลเลอร์ตัวนี้จะประกอบไปด้วย ส่วนที่ควบคุมการทำงานทั้งหมด รวมถึงการส่งข้อมูลกับ CPU การควบคุมไดรฟ์ การอ่าน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เขียนข้อมูลจากดิสก์ และอ่านอัลกอริทึมจากส่วนแยกข้อมูล(Data Separator) การติดต่อกับคำสั่งและสถานะจะใช้รีจิสเตอร์สถานะหลัก (Main Status Register)

2. ส่วนแยกข้อมูล (Data Separator)

ภายในจะประกอบด้วย อนาลอกเฟสล็อคลูป และวงจรที่เกี่ยวข้อง โดยเฟสล็อคลูปจะสร้างข้อมูลดิบจากการอ่านดิสก์ไดรฟ์ สัญญาณที่ได้จะถูกนำไปแยกสัญญาณนาฬิกาและพัลส์ข้อมูลออกจากกัน โดยข้อมูลนี้ก็จะนำมาเรียบเรียงใหม่เป็นไบต์และส่งไปยังไมโครโปรเซสเซอร์โดยคอนโทรลเลอร์

เฟสล็อคลูปจะประกอบไปด้วย ส่วน Phase comparator,filter,Voltage control oscillator (VCO) และ Programmable divider

ส่วนเปรียบเทียบเฟสจะทำการเปรียบเทียบส่วน divider's output กับส่วนข้อมูลดิบจากดิสก์ ซึ่งจะแปลงความแตกต่างนี้ไปอยู่ในรูปของกระแสซึ่งจะประจุและคายประจุฟิลเตอร์ที่ต่อภายนอก ผลจากการเปลี่ยนแรงดันที่ฟิลเตอร์จะเปลี่ยนความถี่ที่ VCO และหารเอาท์พุท PLLจะทำการล็อคเมื่อความถี่ของส่วนหารเท่ากับค่าความถี่เฉลี่ยของข้อมูลจากดิสก์

3. ส่วนชดเชยการเขียน (Write Precompensation)

ชิปนี้มีรีจิสเตอร์แบบ single fixed 3-bit shift register ซึ่งเอาท์พุทจะมัลติเพล็กซ์ไปในส่วนเอาท์พุทที่จะเขียน โดยจะถูกเลือกจากส่วนอัลกอริทึมชดเชย ซึ่งค่าของมันจะเลือกได้จากขา pump/psen

4. DMA Enable Logic

เป็นค่า logic ที่จะไป disable DMA lines และ Interrupt output ภายใต้การควบคุมโดย DMA Enable bit ในรีจิสเตอร์ควบคุมไดรฟ์ ถ้าเป็น 0 แล้ว ขา INT และ DRQ จะอยู่ในภาวะ Tri-state และ DAK จะ disable

5. Drive Output Buffers/Input Recievers

ส่วนเอาท์พุทสามารถขับเอาท์พุทที่มีความต้านทาน 150Ω $\pm$ 10% ซึ่งสามารถที่จะนำมาต่อกับฟลอปปี้

6. Bus Interface – Address Decode

วงจรถอดรหัสแอดเดรสจะใช้ A0-A2 ซึ่งมีค่าดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง_พระจอมเกล้าลาดกระบัง

ตารางที่ 2.2 ผังแสดงแอดเดรสของหน่วยความจำของ DP8473

A0	A1	A2	R/W	รีจิสเตอร์
0	0	0	x	-
0	0	1	x	-
0	1	0	W	รีจิสเตอร์ควบคุมไดรฟ์ (Drive control register)
0	1	1	x	-
1	0	0	R	รีจิสเตอร์สถานะหลัก (Main status register)
1	0	1	R/W	รีจิสเตอร์ข้อมูล (Data register)
1	1	0	x	-
1	1	1	W	รีจิสเตอร์อัตราของข้อมูล (Data rate register)
1	1	1	R	บิตแสดงการเปลี่ยนดิสก์ (Disk change bit)

7. รีจิสเตอร์ควบคุมไดรฟ์ (Drive Control Register)

ทั้ง 8 บิตจะเขียนเฉพาะเมื่อได้เลือกรีจิสเตอร์ควบคุมไดรฟ์เพื่อ ควบคุมมอเตอร์ ,DMA และการรีเซต

8. Reset Logic

ขารีเซตจะ active high และส่งไปยังรีจิสเตอร์ควบคุมไดรฟ์และรีจิสเตอร์อัตราของข้อมูลโดยตรง ถ้ามีการรีเซตทางฮาร์ดแวร์ รีจิสเตอร์ควบคุมไดรฟ์จะรีเซตทั้งหมดเป็น 0 และรีจิสเตอร์แสดงอัตราของข้อมูลจะเซตที่ 250 kb/s และคอนโทรลเลอร์จะยังคงรีเซตจนกว่าจะมีการเซตการทำงานที่รีจิสเตอร์ควบคุมไดรฟ์ ส่วนถ้าหากมีการรีเซตทางซอฟท์แวร์ก็จะไม่ส่งผลให้มีการรีเซตรีจิสเตอร์อัตราของข้อมูลและรีจิสเตอร์ควบคุมไดรฟ์

9. รีจิสเตอร์อัตราของข้อมูล (Data Rate Register) และสัญญาณนาฬิกา (Logic Clock)

เป็นรีจิสเตอร์ 2 บิตที่ควบคุมอัตราการส่งข้อมูล รีจิสเตอร์จะป้อนค่าที่เลือกอัตราของข้อมูลโดยโปรแกรมค่าตัวหารคริสตอลหรือสัญญาณนาฬิกา คือ 3,5 หรือ 6 โดยที่สัญญาณนาฬิกาเป็น 4,4.8 และ 8 MHz ตามลำดับ

10. Crystal Oscillator

DP 8473 จะใช้สัญญาณนาฬิกาด้วยอัตรา 24 MHz

2.3.2. คำอธิบายเกี่ยวกับรีจิสเตอร์

1. รีจิสเตอร์สถานะหลัก (Main Status Register)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

42295

จะบอกถึงสถานะปัจจุบันของคอนโทรลเลอร์ โดยปกติจะใช้อ่านเท่านั้น

D7 Request for Master: จะชี้ว่ารีจิสเตอร์ข้อมูลพร้อมที่จะรับส่งข้อมูลกับไมโครโปรเซสเซอร์หรือยัง บิตนี้จะเคลียร์เมื่อมีการรับส่ง และจะเซตเมื่อคอนโทรลเลอร์พร้อมที่จะมีการรับส่งไบต์ต่อไป

D6 Data Direction : จะบอกว่าคอนโทรลเลอร์ต้องการที่จะเขียนลงรีจิสเตอร์ข้อมูล (เป็น 0) หรือต้องการอ่าน ลงรีจิสเตอร์ข้อมูล (เป็น 1)

D5 Non-DMA Execution : จะเซตเฉพาะเมื่ออยู่ระหว่างกระบวนการที่ไม่ใช่ใน DMA mode

D4 Command in Progress: จะเซตหลังจากได้เขียนไบต์แรกของส่วนคำสั่ง และเคลียร์หลังจากอ่านไบต์สุดท้ายในส่วนของผลลัพธ์ ถ้าเป็นคำสั่งที่ไม่ต้องการผลก็จะเคลียร์หลังจากได้เขียนไบต์สุดท้ายของส่วนคำสั่ง

D3 Drive 3 Seeking: จะเซตหลังจากไบต์สุดท้ายของคำสั่งที่ใช้ค้นหาหรือชดเชยไดรฟ์ 3 และเคลียร์หลังจากอ่านไบต์แรกในส่วนของผลของคำสั่งรับการอินเตอร์รัพท์จากไดรฟ์นี้

D2 Drive 2 Seeking: เช่นเดียวกับ D3 แต่เป็นไดรฟ์ 2 แทน

D1 Drive 1 Seeking : เช่นเดียวกับ D3 แต่เป็นไดรฟ์ 1 แทน

D0 Drive 0 Seeking: เช่นเดียวกับ D3 แต่เป็นไดรฟ์ 0 แทน

2. รีจิสเตอร์ข้อมูล (Data Register)

รีจิสเตอร์นี้เป็นได้ทั้งอ่านและเขียน เป็นส่วนที่ใช้งานในทุก ๆ คำสั่งที่มีการส่งผ่านระหว่าง CPU และ DP 8473 โดยไมโครโปรเซสเซอร์นั้นจะโหลดคำสั่งของคอนโทรลเลอร์ไปเก็บไว้ที่รีจิสเตอร์นี้ ซึ่งจะขึ้นอยู่กับStatus Register Request for Master และบิต Data Direction ส่วนของผลลัพธ์จะส่งจากรีจิสเตอร์สถานะหลักและข้อมูลไปยังไมโครโปรเซสเซอร์ในรูปแบบเดียวกัน

3. รีจิสเตอร์ควบคุมไดรฟ์ (Drive Control Register)

รีจิสเตอร์นี้ใช้เฉพาะอ่านเท่านั้น

D7 Motor Enable 3 : ควบคุมมอเตอร์ของไดรฟ์ที่ 3 โดยถ้าเป็น 0 แล้ว เอาท์พุทจะเป็น High

D6 Motor Enable 2 : เป็นไดรฟ์ที่ 2

D5 Motor Enable 1 : เป็นไดรฟ์ที่ 1

D4 Motor Enable 0 : เป็นไดรฟ์ที่ 0

D3 DMA Enable : ถ้าเป็น 1 จะเป็นการ Enable DRQ, DAK, INT

D2 Reset Controller : ถ้าเป็น 0 จะเป็นการรีเซตคอนโทรลเลอร์ ซึ่งไม่มีผลกับรีจิสเตอร์คอนโทรลไดรฟ์หรือรีจิสเตอร์อัตราของข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

D1-D0 Drive Select : เข้ารหัสเลือกไดรฟ์ทั้ง 4

ตารางที่ 2.3 แสดงค่าสำหรับการเลือกในรีจิสเตอร์ควบคุมไดรฟ์

D7	D6	D5	D4	D1	D0	Function
X	X	X	1	0	0	ไดรฟ์ 0
X	X	1	X	0	1	ไดรฟ์ 1
X	1	X	X	1	0	ไดรฟ์ 2
1	X	X	X	1	1	ไดรฟ์ 3

4. รีจิสเตอร์แสดงอัตราของข้อมูล (Data Rate Register)

D7 – D2 : ไม่ใช้งาน

D1 – D0 Data Rate Register : ตั้งค่าอัตราของข้อมูลและการชดเชยการเขียนตามตาราง

ตารางที่ 2.4 อัตราของการรับส่งข้อมูลและค่าของการชดเชยโปรแกรม

D1	D0	DRVTY P Pin	Data Rate MFM (kb/s)	Normal Precomp. (ns)	Alternative Precomp. (ns)	FGND Pin Enabled	RPM/LC Pin Level
0	0	x	500	125	125	FGND500	H
0	1	0	250	125	250	FGND250	L
0	1	1	300	208	208	FGND250	L
1	0	0	250	125	250	FGND250	L
1	0	1	250	125	250	FGND250	L
1	1	0	1000	83	83	None	H
1	1	1	1000	83	83	None	L

5. Disk Changed Register

รีจิสเตอร์นี้ใช้อ่านเท่านั้น

D7 Disk Changed : บิตนี้จะแลตช์คอมพลีเมนต์ของขา Disk Changed Input เอาไว้ ถ้าที่ DSKCHG pin เป็น 0 แล้วที่บิตนี้จะเป็น 1

D6 – D0 : จะใช้เฉพาะสำหรับ Harddisk controller

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นเมื่อมีการอ่านรีจิสเตอร์นี้ บิต 0-6 ก็จะเป็น 1 Tri-state

2.3.3.รีจิสเตอร์สถานะส่วนผลลัพธ์ (Result Phase Status Registers)

1. Status Register 0 (ST0)

D7 – D6 Interrupt Code

00 : ทำงานตามคำสั่งเสร็จเรียบร้อยแล้ว

01 : กำลังทำงานตามคำสั่ง แต่ยังไม่เสร็จสิ้น

10 : คำสั่งผิดพลาด

11: พร้อมที่จะเปลี่ยนสถานะในระหว่าง Polling Mode

D5 Seek End : การค้นหาหรือการชดเชยเสร็จแล้ว บิตนี้จะเซต

D4 Equipment Check : หลังจากคำสั่งชดเชย พบสัญญาณแทร็กที่ 0 ผิดพลาด (ใช้ในการตอบรับการอินเทอร์รัพท์)

D3 Not used : 0

D2 Head Address (ในส่วน Execution Phase)

D1,D0 Drive Select (ในส่วน Execution Phase)

00 : Drive 0

01 : Drive 1

10 : Drive 2

11 : Drive 3

2. Status Register 1 (ST1)

D7 End of Track : คอนโทรลเลอร์ส่งไบต์สุดท้ายของเซกเตอร์สุดท้าย โดยขา TC ไม่ active เซกเตอร์สุดท้ายเป็น End of Track sector number ของส่วนที่โปรแกรมในส่วนคำสั่ง

D6 ไม่ใช้งาน : 0

D5 CRC Error : ถ้าบิตนี้เซตและบิตที่ 5 ของ ST2 เคลียร์ จะมี CRC Error เกิดขึ้นในส่วนแอดเดรสของเซกเตอร์ที่ถูกต้อง แต่ถ้าบิตที่ 5 ของ ST2 เซต จะมี CRC Error เกิดขึ้นในส่วนของข้อมูล

D4 Over Run : คอนโทรลเลอร์ไม่ได้รับการตอบสนองโดยไมโครโปรเซสเซอร์เร็วพอในระหว่างการส่งข้อมูลในการทำงาน

D3 ไม่ใช้งาน : 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

D2 No Data : ซึ่งจะมีปัญหาที่เป็นไปได้ 3 อย่าง

1. คอนโทรลเลอร์ค้นหาเซกเตอร์ที่ระบุในส่วนของคำสั่งไม่เจอในกระบวนการอ่าน เขียน หรือการสแกน
2. คอนโทรลเลอร์ไม่สามารถอ่านค่าแอดเดรสโดยที่ไม่มี CRC Error เกิดขึ้นในระหว่างคำสั่ง Read ID Command
3. คอนโทรลเลอร์ไม่สามารถหาเซกเตอร์ที่เริ่มต้นในคำสั่ง Read A Track command ได้

D1 Not Writable : Write Protect Active

D0 Missing Address Mark : ถ้าบิต 0 ของ ST2 เคลียร์ ดังนั้น คอนโทรลเลอร์ไม่สามารถค้นหา Address Field Address Mark ภายหลังการหมุนดิสก์ แต่ถ้าบิต 0 ของ ST2 เซต จะเป็นการค้นหา Data Field Address Mark ไม่เจอ

3. Status Register 2 (ST2)

D7 : ไม่ใช้งาน โดยปกติจะเป็น 0

D6 Control Mark : คอนโทรลเลอร์พยายามอ่านเซกเตอร์ที่มี Address Mark ของข้อมูลที่ถูกลบในระหว่างการอ่านข้อมูลหรือคำสั่งสแกน หรือถ้าอ่าน Deleted Data Command แล้วจะตรวจพบ Address Mark ที่เป็นแบบปกติ

D5 CRC Error in Data Field : คอนโทรเลอร์พบความผิดพลาดในส่วน Data Field ซึ่งบิตที่ 5 ของ ST1 ก็จะเซตด้วย

D4 Wrong Track : จะเซตเมื่อหาเซ็กเตอร์ที่ต้องการไม่พบ และหมายเลขของแทร็กที่ตำแหน่งแทร็กปัจจุบันต่างจากที่เก็บไว้ใน Track Register

D3 Scan Equal Hit : กรณี Equal จะเกิดในระหว่างการสแกน

D2 Scan Not Satisfied : คอนโทรลเลอร์หาเซกเตอร์บนแทร็กที่ต้องการไม่พบ

D1 Bad Track : จะเซตเมื่อหาเซกเตอร์ที่ต้องการไม่พบ

D0 Missing Address Mark in Data Field : คอนโทรลเลอร์หา Data Field Address Mark ระหว่างการอ่านหรือสแกนไม่พบ บิต 0 ของ ST1 ก็จะเซตด้วย

4. Status Register 3 (ST3)

D7 : ไม่ใช้งาน : 0

D6 Write Protect Status

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

D5 : ไม่ใช้งาน :1

D4 Track 0 Status

D3 : ไม่ใช้งาน : 0

D2 Head Select Status

D1 ,D0 Drive Selected

00 : Drive 0

01 : Drive 1

10 : Drive 2

11 : Drive 3

ตารางที่ 2.5 สรุป FDC Register

Bits Register	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
รีจิสเตอร์ควบคุมไดรฟ์ (W)	มอเตอร์ 3 Enable	มอเตอร์ 2 Enable	มอเตอร์ 1 Enable	มอเตอร์ 0 Enable	DMA Enable	Software Reset	Drive Select 1	Drive Select 0
รีจิสเตอร์สถานะหลัก (R)	Request for Master	Data Direction	Non-DMA Execution	Connand in Progress	Drive 3 Seeking	Drive2 Seeking	Drive 1 Seeking	Drive 0 Seeking
รีจิสเตอร์ข้อมูล (R/W)	Data 7	Data 6	Data 5	Data 4	Data 3	Data 2	Data 1	Data 0
รีจิสเตอร์อัตราของข้อมูล (W)	X	X	X	X	X	X	Data Rate Bit 1	Data Rate Bit2
Disk Changed Register (R)	DSKCHG Pin Inverse	Z	Z	Z	Z	Z	Z	Z
ST0	Interrupt Code	Interrupt Code	Seek End	No Track 0	0	Head Address	Drive 1 (Exec.)	Drive 2 (Exec.)
ST1	End of Track	0	CRC Error	Data Overrun	0	No Data	Write Protect	Missing Address Mark
ST2	0	Control Mark	CRC Error in Data	Wrong Track	Scan Equal Hit	Scan Not Satisfied	Bad Track	Missing Address Mark in

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

			Field					Data Field
ST3	0	Write Protect Status	1	Track 0 Status	0	Head Select Status	Drive 1 (Exec.)	Drive 2 (Exec.)

2.3.4.ส่วนซอฟท์แวร์อินเตอร์เฟสโปรเซสเซสเซอร์ (Processor Software Interface)

1. Command Sequence

1.1 **Command Phase** ไมโครโปรเซสเซอร์จะเขียนชุดคำสั่งลงไปในรีจิสเตอร์ข้อมูล และจะเริ่มส่วน Execution Phase ในทันทีที่เขียนคำสั่งไบต์สุดท้ายของ Command Phase ก่อนที่จะปฏิบัติตามคำสั่งจะมีการเซตรีจิสเตอร์ควบคุมไดรฟ์และรีจิสเตอร์อัตราของข้อมูล

1.2 **Execution Phase** ดิสก์คอนโทรลเลอร์จะทำตามคำสั่งที่ต้องการ บางคำสั่งจะต้องการให้ไมโครโปรเซสเซอร์อ่านหรือเขียนข้อมูลลงรีจิสเตอร์ข้อมูลในช่วงนี้

1.3 **Result Phase** ไมโครโปรเซสเซอร์จะอ่านชุดของไบต์ต่าง ๆ จากรีจิสเตอร์ข้อมูล ซึ่งจะบอกว่าการทำงานตามคำสั่งถูกต้องหรือไม่ และคำสั่งใหม่จะเริ่มด้วยการเขียนส่วนของชุดคำสั่งภายหลังจากการอ่านในส่วนผลลัพธ์นี้ไบต์สุดท้ายแล้ว ถ้าคำสั่งต่อไปต้องการเลือกไดรฟ์อื่นหรือเปลี่ยนอัตราของข้อมูลแล้ว จะต้องมีการเปลี่ยนแปลงในรีจิสเตอร์ที่เกี่ยวข้อง

ในระหว่างช่วงของ Command Phase และ Result Phase จะมีการส่งข้อมูลเข้าและออกจากรีจิสเตอร์ข้อมูล โดยรีจิสเตอร์สถานะหลักจะตรวจสอบได้โดยการใช้ซอฟท์แวร์ เมื่อมีการส่งข้อมูลเกิดขึ้น บิต 6 จะเคลียร์และบิต 7 จะเซตก่อนที่จะสามารถเขียนลงรีจิสเตอร์ข้อมูลในระหว่าง Command Phase และจะเซตทั้งคู่ก่อนที่จะสามารถอ่านจากรีจิสเตอร์ข้อมูลในส่วน Result Phase ได้

ถ้ามีการส่งข้อมูลในช่วง Execution Phase มีอยู่ 3 วิธีการที่จะใช้คือ ใช้ DMA mode ถ้าระบบมี DMA controller ซึ่งจะยอมให้ไมโครโปรเซสเซอร์ทำงานอย่างอื่นในช่วงการรับส่งข้อมูลได้ ถ้าไม่ใช้ DMA mode ก็อาจจะใช้การอินเตอร์รัพต์สำหรับแต่ละไบต์ที่มีการรับส่งในช่วง Execution Phase หรืออาจใช้การ polling เพื่อบอกว่าต้องการรับส่งข้อมูล

DMA Mode

ถ้าเลือกใช้โหมดนี้ แล้วสัญญาณ DMA Request จะถูกสร้างขึ้นในช่วงการทำงาน ขณะที่พร้อมที่จะรับส่งแต่ละไบต์ จะต้องมีการเลือกการทำงานในส่วนคำสั่ง Specify command และต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เลือกการทำงานใน DMA Mode ในรีจิสเตอร์ควบคุมไดรฟ์ โดย DMA controller จะตอบสนองด้วยสัญญาณ DMA Acknowledge และ Write/Read Strobe สัญญาณ DMA Request จะเคลียร์หลังจากการส่งไบต์สุดท้าย ซึ่งจะมีการสร้างการอินเตอร์รัพต์เพื่อบอกจุดเริ่มต้นของ Result Phase ในระหว่างกระบวนการ DMA ขา CS จะต้องเป็น High และ TC จะแสดงการจบการทำงาน

Interrupt Mode

การอินเตอร์รัพต์จะสร้างขึ้นในช่วงการทำงาน เมื่อพร้อมรับส่งข้อมูล บิต 5 และ 7 ของรีจิสเตอร์สถานะหลักจะเซต การอินเตอร์รัพต์จะเคลียร์เมื่อมีการส่งข้อมูลไปหรือรับจากรีจิสเตอร์ข้อมูล โดยในการรับส่งข้อมูลจะต้องเป็นไปตามเวลา ถ้าเกินเวลาจะมีการแสดงผล Overrun Error เมื่อถึงคำสั่งสุดท้ายของเซกเตอร์นั้น

ตารางที่ 2.6 แสดงค่าเวลาสูงสุดที่ใช้ใน DMA Mode หรือ Interrupt Mode

Data Rate	Time to Service (μs)
125	62
250	30
500	14
1000	6

การอินเตอร์รัพต์จะสร้างขึ้นหลังจากไบต์สุดท้ายได้ทำการส่งเรียบร้อยแล้ว ซึ่งจะบอกถึงการเริ่มต้นส่วนของผลลัพธ์ บิต 7 และ 6 ของรีจิสเตอร์สถานะหลักจะเซตและบิต 5 จะเคลียร์ ซึ่งจะเคลียร์การอินเตอร์รัพต์ได้โดยการอ่านไบต์แรกของ Result Phase

Software Polling

วิธีนี้สามารถใช้วนตรวจที่รีจิสเตอร์สถานะหลักในช่วงของ Execution Phase เพื่อบอกว่าพร้อมมีการรับส่งข้อมูล ในโหมดนี้ บิต 7 ของรีจิสเตอร์สถานะหลักจะบอกสถานะที่ขาอินเตอร์รัพต์ การเคลื่อนย้ายข้อมูลจะเหมือนกับการอินเตอร์รัพต์

2.3.5. อธิบายคำสั่งที่จำเป็นในการใช้งาน (Command Description)

1. Read Data

Op-code ของการอ่านข้อมูลจะเขียนลงรีจิสเตอรด์ข้อมูลตามด้วย 8 ไบต์ตามตาราง โดยหลังจากไบต์สุดท้าย คอนโทรลเลอร์จะเริ่มมองหาหัวอ่านที่ถูกต้อง เมื่อพบแล้วจะส่งข้อมูลไปยังไมโครโปรเซสเซอร์ หลังจากเสร็จแต่ละเซกเตอร์ก็จะทำการเพิ่มหมายเลขของเซกเตอร์ทีละหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แล้วหาเซกเตอร์ต่อไป ถ้าได้เซตเป็นแบบ MT (Multi-track) เอาไว้ สามารถอ่านได้ทั้ง 2 ด้านของแทร็กนั้น โดยเริ่มจากที่ Side 0 และจะอ่านจนจบแทร็กแล้วก็เริ่มอ่านที่เซกเตอร์ 1 ของ Side 1

ใน DMA Mode คำสั่งอ่านข้อมูลจะทำการอ่านจนกระทั่ง ขา TC เซต คือ DMA controller จะโปรแกรมเพื่อการส่งข้อมูลที่ถูกต้อง TC นี้จะทำการควบคุมด้วยไมโครโปรเซสเซอร์

จำนวนไบต์ของข้อมูลต่อเซกเตอร์เป็นไปตามตาราง ซึ่งถ้าได้เซตเป็น 0 ดังนั้น ตัวแปรค่าความยาวของข้อมูลจะบอกถึงจำนวนไบต์ที่คอนโทรเลอร์ส่งไปยังไมโครโปรเซสเซอร์ โดยถ้าน้อยกว่า 128 ไบต์ คอนโทรลเลอร์ก็ยังจะอ่านจนครบ 128 ไบต์แล้วเช็ค CRC เฉพาะจำนวนไบต์ที่กำหนดโดยตัวแปรค่าความยาวของข้อมูลที่ส่งไปยังไมโครโปรเซสเซอร์ ความยาวของข้อมูลนี้จะไม่เซตเป็น 0 โดยถ้าพารามิเตอร์แสดงจำนวนไบต์ต่อเซกเตอร์ไม่เป็น 0 ดังนั้น ความยาวของข้อมูลไม่มีความหมายและจะเซตเป็น FFH

ตารางที่ 2.7 แสดงการเลือกขนาดของเซกเตอร์

Bytes/Sector Code	Number of Bytes in Data Field
0	128
1	256
2	512
3	1024
4	2048
5	4096
6	8192

ถ้าได้เลือกที่ Implied Seek Mode โดยจากทั้งคำสั่งโหมดและ IPS bit ในคำสั่งนี้ การค้นหาก็จะหาตำแหน่งแทร็กที่มีหมายเลขระบุเอาไว้ในส่วนของคำสั่ง คอนโทรลเลอร์จะรอตามเวลาตั้งหัวอ่าน

หลังจากกรณีเหล่านี้ คอนโทรลเลอร์จะค้นหาเซกเตอร์ด้วยการเปรียบเทียบหมายเลขแทร็กหัวอ่าน และจำนวนไบต์ต่อเซกเตอร์ตามที่ระบุในคำสั่งกับไบต์ที่อ่านได้จากดิสก์ในส่วน Address Field ถ้าค้นหาเซกเตอร์ที่ถูกต้องเจอแล้ว แต่พบว่ามี CRC Error ในส่วน Address Field บิตที่ 5 ของ ST1(CRC Error) จะเซต หรือถ้าหาเซกเตอร์นั้นไม่พบ บิต 2 ของ ST1 (No Data) ก็จะเซต ถ้าที่บาง Address Field Track number เป็น FFH บิตที่1 ของ ST2 (Bad Track) ก็จะเซต หรือถ้าที่พบมีค่าต่างจากในคำสั่ง บิตที่ 4 ของ ST2 (Wrong Track) ก็จะเซต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากการหาเซกเตอร์ที่ถูกต้องพบแล้ว คอนโทรลเลอร์จะอ่านใน Data Field ถ้าพบ Deleted Data Mark และ SK bit เซต เซกเตอร์จะไม่ถูกอ่าน บิตที่ 6 ของ ST2 (Control Mark) จะเซตและเริ่มค้นหาเซกเตอร์ต่อไป ถ้า Deleted Data Mark นั้นพบและบิต SK ไม่เซต เซกเตอร์ก็จะถูกอ่าน บิตที่ 6 ของ ST2 เซต และการอ่านจะสิ้นสุดตามปกติ แต่ถ้ามี CRC Error ตรวจพบใน Data Field บิตที่ 5 จะเซตทั้งใน ST1 และ ST2 และจะตรวจพบความผิดปกติ

ถ้าไม่มีปัญหาเกิดขึ้นในระหว่างขั้นตอนการอ่าน ก็จะดำเนินการต่อไปยังเซกเตอร์ถัดไปตามลำดับ จนกว่าขา TC จะเซต หรือว่ามี Error เกิดขึ้น

ถ้าเรายังไม่ได้ใส่ดิสก์ไดรฟ์ คอนโทรลเลอร์ก็จะบอกโดยการส่งไบต์ไปยังรีจิสเตอร์ข้อมูลในส่วน Result Phase และเมื่อช่วง Execution Phase สิ้นสุดลงก็จะมีสัญญาณการอินเตอร์รัพต์ขึ้น ค่าที่ได้ในช่วงของผลลัพธ์นี้จะเป็นไปตามตาราง โดยถ้าเกิด Error ขึ้น ไบต์ของผลลัพธ์ก็จะแสดงเซกเตอร์ที่กำลังอ่านและพบ Error

2. Write Data

จะคล้ายกับการอ่านข้อมูล แต่การส่งข้อมูลจะส่งจากไมโครโปรเซสเซอร์ไปยังดิสก์ ถ้าตรวจพบสัญญาณ Write Protect บิตที่ 1 ของ ST1 (Not Writable) จะเซต

3. Recalibrate

คำสั่งนี้จะคล้ายกับคำสั่งการค้นหา โดยจะเลื่อนไปยังแทร็ก 0 โดยจะสร้างพัลซ์ขึ้นจนกระทั่งพบสัญญาณแทร็ก 0 ถูกต้อง ถ้าใช้พัลซ์ทั้งหมด 77 พัลซ์แล้วยังไม่พบก็จะแสดง Error แต่ถ้าหากเลือกใช้โหมดการขยายพิสัยแทร็กก็จะไม่เกิด Error จนกว่าจะใช้ 3917 พัลซ์

ตารางที่ 2.8 แสดงค่าเวลาการเลื่อนหัวอ่าน การเริ่มและหยุดมอเตอร์ ที่อัตราการรับส่งข้อมูล 500 kb/s แบบ MFM

Timer	Mode 1		Mode 2		Unit
	Value	Range	Value	Range	
Step Rate	(16 - N1)	1 – 16	(16 – N1)	1 – 16	ms
Motor Off	N2 x 16	0 – 240	N2 x 512	0 – 7680	ms
Motor On	N3 x 2	0 – 254	N3 x 32	0 – 4064	ms

4. Specify

คำสั่งนี้จะเซตค่าเริ่มต้นสำหรับ timer ภายในแต่ละตัวตามตาราง การเลือกใช้ DMA หรือ Non-DMA Operation จะใช้ Non-DMA bit เมื่อบิตนี้เป็น 1 คือการเลือกโหมด Non-DMA แต่ถ้าเป็น 0 จะเป็น DMA mode คำสั่งนี้จะไม่เกิดการอินเตอร์รัพต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. Mode

ใช้เลือกคุณสมบัติของคอนโทรลเลอร์ มีคำสั่งตามตารางคำสั่ง

1. **Motor Timer**
   - TMR = 0 เป็นการเลือก Timer สำหรับการเปิดปิดมอเตอร์ในโหมดที่ 1
   - TMR = 1 เป็นการเลือก Timer สำหรับการเปิดปิดมอเตอร์ในโหมดที่ 2
2. **Low Power(LW PR)**
   - 00 ไม่ใช้โหมดของ Low Power Mode
   - 01 ไปยังโหมดนี้เมื่อครบ 500 ms หลังจากที่หมดเวลาของ Head Unload
   - 10 ไปยังโหมดนี้ทันที
   - 11 ไม่ใช้งาน
3. **Index Address Format (IAF)**
   - IAF = 0 คอนโทรลเลอร์จะฟอร์แมตแทร็กด้วย Index Address Field Included (IBM Format)
   - IAF = 1 จะฟอร์แมตโดยไม่มีส่วน Index Address Mark Field (ISO Format)
4. **Implied Seek (IPS)**
   - IPS = 0 ไม่ใช้การค้นหาแบบ Implied Seek
   - IPS = 1 จะทำการค้นหาแบบอัตโนมัติ
5. **Extended Track Range (ETR)**
   - ETR = 0 เป็นรูปแบบสำหรับระบบ IBM System 34 (double density) หรือระบบ 3740 (single density)
   - ETR = 1 จะเป็นรูปแบบเดียวกัน แต่ว่ามีหมายเลขแทร็กทั้งหมด 12 บิต โดยที่ 4 บิตบนจะอยู่ในไบต์ของหมายเลขหัวอ่าน
6. **Scan Wildcard (WLD)**
   - WLD = 0 ค่า FFH ในส่วนที่ได้จากไมโครโปรเซสเซอร์หรือจากดิสก์ในคำสั่งสแกนจะถูกตีความเป็นค่าที่เท่ากันเสมอ เรียกว่าเป็น Wildcard character
   - WLD = 1 ค่า FFH นั้นไม่ถูกตีความเป็น Wildcard character
7. **Head Settle** เวลาที่หัวอ่านใช้หลังการค้นหาแบบ Implied Seek โดยที่

$$\text{เวลา} = N \times (4 \text{ ms})$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. **PUMP Pulse Output (PU)** เมื่อได้เลือกให้มีสัญญาณบอกว่าเมื่อไรที่ Data Separator's charge pump นั้นมีเฟสที่ถูกต้อง โดยจะเป็นชุดของพัลส์ และสัญญาณนี้จะเป็นเอาท์พุทที่ขา PUMP/PREN เมื่อให้บิตนี้เซต

9. **Read Gate (RG)** จะเหมือนกับเอาท์พุทของ PU โดยเมื่อบิตนี้เซตจะเป็นการเลือกให้ที่ขา DSKCHG เป็นสัญญาณ Read Gate Signal ให้กับส่วนแยกข้อมูล

6. Invalid Command

ถ้าคอนโทรลเลอร์ได้รับคำสั่งที่ไม่ถูกต้อง คอนโทรลเลอร์จะตอบกลับด้วย ST0 ในส่วนของ Result Phase โดยจะไม่ทำให้เกิดการอินเตอร์รัพต์ และในกรณีที่บิตที่ 6 และ 7 ในรีจิสเตอร์สถานะหลักเซตเป็น 1 บอกโปรเซสเซอร์ว่าคอนโทรลเลอร์ในส่วนของ Result Phase และใน ST0 จะต้องถูกอ่าน เมื่ออ่านค่า ST0 แล้วมันจะเป็น 80H ที่บอกว่าได้รับคำสั่งผิดพลาด

2.4 ความรู้เบื้องต้นเกี่ยวกับบัตรแม่เหล็ก

บัตรแม่เหล็กที่ใช้ในโครงงานนี้จะใช้มาตรฐานเดียวกันกับมาตรฐานของบัตรเอทีเอ็มที่ใช้ในธนาคารทั่วๆไป โดยรูปแบบของข้อมูลที่บันทึกบนแถบแม่เหล็กและคุณลักษณะต่างๆของบัตรแม่เหล็กจะเป็นไปตามมาตรฐานสากล ISO 7810-7813 เพื่อที่จะสามารถพัฒนาระบบต่อไปได้

2.4.1 มาตรฐานของบัตรแม่เหล็ก



รูปที่ 2.4 แสดงขนาดมาตรฐานของบัตรแม่เหล็ก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.4 ตามมาตรฐานของ ISO เราสามารถพิจารณาแทร็กต่างๆที่ใช้งานได้เป็น 3 แทรก โดยมีรายละเอียดในการใช้งานแต่ละแทร็กดังนี้

แทร็กที่ 1 เป็นแทร็กที่สามารถทำการอ่านข้อมูลได้อย่างเดียว (Read only) โดยข้อมูลประกอบด้วยตัวอักษรและตัวเลข อยู่ในบริเวณเส้นขนาน a และ b

แทร็กที่ 2 เป็นแทร็กที่สามารถอ่านข้อมูลได้อย่างเดียวเช่นกัน แต่ข้อมูลจะเป็นตัวเลขเพียงอย่างเดียว อยู่ในช่วงเส้นขนาน b และ c เป็นแทร็กที่ใช้งานทั่วไป

แทร็กที่ 3 เป็นแทร็กที่สามารถทำการอ่านข้อมูลและเขียนข้อมูลลงไปได้ โดยข้อมูลที่อ่านและเขียนเป็นตัวเลข อยู่ในช่วงเส้นขนาน c และ d

2.4.2 ชุดรหัสข้อมูลในแทร็กที่ 2 ของบัตรแม่เหล็ก

ในการใช้งานโดยทั่วๆไปของบัตรแม่เหล็ก จะใช้งานในส่วนของแทร็กที่ 2 ถ้าเป็นบัตรเอทีเอ็มของธนาคารต่างๆ ในส่วนของแทร็กนี้จะเป็นการเก็บข้อมูลเกี่ยวกับข้อมูลของธนาคารและเขที่บัญชี ดังนั้นแต่ละธนาคารจึงสามารถใช้บริการร่วมกันได้

ข้อมูลที่บันทึกในแทร็กที่ 2 ของบัตรแม่เหล็กจะเป็นตัวเลขเพียงอย่างเดียว โดยที่ตัวเลขหนึ่งตัวจะประกอบด้วยบิตข้อมูลแบบ BCD 4 บิต และบิตพาริตี้อีก 1 บิต ซึ่งใช้ในการตรวจสอบข้อมูลแต่ละตัวเลขโดยตรวจสอบแบบพาริตี้คี่ ตามมาตรฐาน ISO ระบุไว้ว่าจำนวนข้อมูลสูงสุดในแทร็กที่ 2 มีได้ไม่เกิน 40 ตัว(รวมสัญลักษณ์การเริ่มต้นและสิ้นสุด) ในส่วนรหัสของข้อมูลแต่ละตัวสำหรับแทร็กที่ 2 แสดงไว้ในตารางที่ 2.9

ตารางที่ 2.9 แสดงรหัสข้อมูลแต่ละตัวของแทร็กที่ 2

PARITY	B4	B3	B2	B1	CHAR
1	0	0	0	0	0
0	0	0	0	1	1
0	0	0	1	0	2
1	0	0	1	1	3
0	0	1	0	0	4
1	0	1	0	1	5
1	0	1	1	0	6
0	0	1	1	1	7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

0	1	0	0	0	8
1	1	0	0	1	9
1	1	0	1	0	A
0	1	0	1	1	B
1	1	1	0	0	C
0	1	1	0	1	D
0	1	1	1	0	E
1	1	1	1	1	F

2.4.3 รูปแบบของข้อมูลที่บันทึกในแทร็กที่ 2 บนบัตรแม่เหล็ก

Sync	Start	Data	Sep	Data	Stop	LRC	Sync
	Bh	0-9h	Dh	0-9h	Fh		

รูปที่ 2.5 รูปแบบของข้อมูลที่บันทึกในแทร็กที่ 2 บนบัตรแม่เหล็ก

จากรูปที่ 2.5 เป็นรูปแบบการของข้อมูลที่บันทึกในบัตรแม่เหล็กที่ใช้กันโดยทั่วไป โดยที่

- Sync (Synchronization character) เป็นการแสดงการเริ่มต้นและปิดท้ายข้อมูล ใช้รหัส 00H
- Start (Start bit) เป็นการแสดงการเริ่มต้นในการอ่านหรือบันทึกข้อมูล ใช้รหัส 0BH
- Data เป็นส่วนของข้อมูลที่เป็นตัวเลข 0-9
- Sep (Separate) เป็นตัวเชื่อมหรือกั้นระหว่างข้อมูล ใช้รหัส 0DH
- Stop (Stop bit) เป็นการแสดงการสิ้นสุดของข้อมูล ใช้รหัส 0FH
- LRC (Longtitude Redundancy Check) เป็นการตรวจสอบชุดข้อมูลในแนวนอน

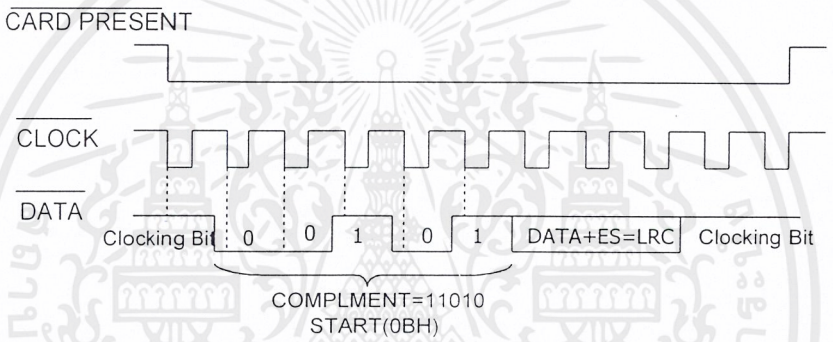
2.5 เครื่องอ่านบัตรแม่เหล็ก

บัตรแม่เหล็ก(Magnetic card) เป็นการ์ดแถบแม่เหล็กซึ่งบนตัวบัตรจะบันทึกข้อมูลและรายละเอียดต่างๆของบัตรไว้ในรูปของเส้นแรงแม่เหล็ก โดยแถบแม่เหล็กบนการ์ด ซึ่งเรียกว่า แทร็ก (Track)นั้นจะถูกแบ่งออกเป็น 3 ส่วน ซึ่งแต่ละส่วนจะใช้เก็บข้อมูลซึ่งมีความหนาแน่นและลักษณะของข้อมูลที่มีความแตกต่างกัน ดังนี้คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แถบแม่เหล็กที่	ความหนาแน่น	การเข้ารหัส	จำนวนตัวอักษร	ลักษณะของข้อมูลที่เก็บ
TRACK1	210 BPI	ALPHA	79	ชื่อเจ้าของบัตรและหมายเลขบัตร
TRACK2	75 BPI	BCD	40	หมายเลขบัตรและวันหมดอายุ
TRACK3	210 BPI	BCD	107	หมายเลขบัตรและรหัสพิเศษ

หมายเหตุ ความหนาแน่นของการบันทึกข้อมูลมีหน่วยเป็น BPI(Byte per inch)



รูปที่ 2.6 แสดงความสัมพันธ์ของสัญญาณ Data,Clock และ Card Presernt ของเครื่องอ่านบัตรแม่เหล็ก

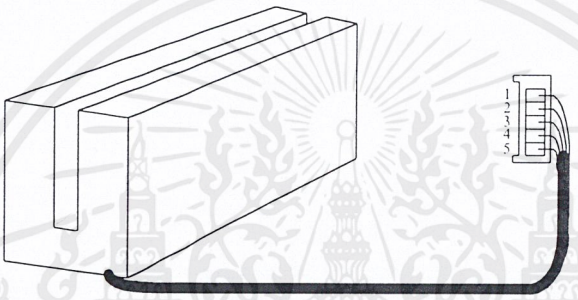
โดยเครื่องอ่านบัตรแม่เหล็กนี้จะอ่านได้เฉพาะข้อมูลที่บันทึกด้วยความละเอียด 75 BPI คือ เฉพาะในส่วนของแทร็กที่ 2 เท่านั้น โดยมีสัญญาณที่ใช้เชื่อมต่อเพื่ออ่านข้อมูลจากเครื่องอ่านแถบแม่เหล็กทั้งหมด 3 สัญญาณ โดยเป็นสัญญาณเอาต์พุตส่งออกมาจากเครื่องอ่านบัตรแม่เหล็กทั้งหมด คือ

1. สัญญาณข้อมูล (Data) เป็นสัญญาณของข้อมูลที่อ่านได้จากบัตรแม่เหล็ก โดยในการอ่านสัญญาณข้อมูลนี้ต้องอ่านแบบอนุกรมทีละบิตโดยต้องพิจารณาให้สอดคล้องกับสัญญาณนาฬิกา (Clock)ด้วย สัญญาณข้อมูลที่อ่านได้นี้จะกลับสภาวะกับสัญญาณจริงอยู่ ดังนั้นเมื่ออ่านสัญญาณจากเครื่องอ่านบัตรแม่เหล็กได้แล้วต้องทำการกลับสภาวะของสัญญาณ(Complement)ก่อนจึงจะได้สัญญาณข้อมูลที่ถูกต้อง ซึ่งสัญญาณข้อมูลที่อ่านได้นั้นจะเริ่มจากบิตที่มีนัยสำคัญต่ำสุด(LSB)ก่อนเป็นอันดับแรก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. สัญญาณนาฬิกา (Clock) ใช้เป็นสัญญาณอ้างอิงเพื่ออ่านข้อมูล โดยการอ่านข้อมูลต้องอ่านในขณะที่สัญญาณนาฬิกาเป็นขอบขาลง(Falling edge)เสมอ

3. สัญญาณ CARD PRESENT เป็นสัญญาณเอาต์พุตส่งออกจากเครื่องอ่านแถบแม่เหล็กทำงานที่ลอจิก 0 เมื่อสัญญาณนี้เป็นลอจิก 0 จะบ่งบอกให้รู้ว่าเครื่องอ่านแถบแม่เหล็กเริ่มต้นทำงาน คือมีการรูดบัตรแม่เหล็กผ่านหัวอ่านสัญญาณ เมื่อสัญญาณนี้หมดลง คือกลับเป็นลอจิก 1 อีกครั้งหนึ่ง แสดงว่าข้อมูลจากการอ่านในครั้งนั้นถูกส่งออกไปหมดเรียบร้อยแล้ว เราอาจใช้ประโยชน์จากสัญญาณนี้ส่งเป็นสัญญาณอินเตอร์รัพต์ให้กับซีพียูเพื่อทำการอ่านข้อมูล



รูปที่ 2.7 เครื่องอ่านบัตรแม่เหล็ก

ตารางที่ 2.10 แสดงสายสัญญาณของเครื่องอ่านบัตรแม่เหล็ก

สัญลักษณ์	การใช้งาน	I/O	สาย
GND	SIGNAL GROUND	-	เขียว
V_{cc}	5V SUPPLY	IN	เหลือง
TD2	TRACK 2 DATA	OUT	น้ำตาล
TC2	TRACK 2 CLOCK	OUT	แดง
CP	CARD PRESENT	OUT	ส้ม

2.5.1 การอ่านข้อมูลจากบัตรแม่เหล็ก

การใช้เครื่องอ่านบัตรแม่เหล็กอ่านข้อมูลจากบัตรแม่เหล็กนั้นสามารถอ่านข้อมูลได้เฉพาะในแทร็กที่ 2 เท่านั้น ดังนั้นจึงขอกล่าวถึงเฉพาะวิธีการอ่านข้อมูลจากแทร็กที่ 2 เพียงอย่างเดียว โดยรูปแบบของข้อมูลในแทร็กที่ 2 ซึ่งใช้บันทึกสัญญาณด้วยความหนาแน่นของสนามแม่เหล็ก 75 BPI จะบรรจุจำนวนรหัสของข้อมูลได้สูงสุดไม่เกิน 40 ตัวอักษร โดยนับรวมรหัสควบคุมและรหัสตรวจสอบด้วย ซึ่งมีรายละเอียดดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SS	PAN	FS	ADDITION DATA	ES	LRC

SS : รหัสข้อมูลแสดงการเริ่มต้นของข้อมูล(Start Sentinel)ในบัตรมีรหัสข้อมูลเป็น 0BH(;)

PAN : ข้อมูลแสดงหมายเลขของบัตร (มีจำนวนข้อมูลในส่วนนี้สูงสุดไม่เกิน 19 หลัก)

FS : รหัสข้อมูลแสดงการแบ่งแยกข้อมูล (Field Seperator)มีรหัสข้อมูลเป็น 0DH(=)

ADDITION DATA : เป็นข้อมูลเพิ่มเติมอื่นๆของบัตร เช่น เดือน ปี ที่ออกบัตรและหมดอายุ

ES : รหัสข้อมูลแสดงการสิ้นสุดของข้อมูลในบัตร (End Sentinel) มีรหัสข้อมูลเป็น 0FH(?)

LRC : ข้อมูลตรวจสอบความผิดพลาด (Longitudinal Redundancy Check)

0000000000 DATA,DATA,.....,DATA ES LRC 000000000000

จากภาพแสดงลักษณะการจัดเรียงของข้อมูลที่เก็บไว้ในแทร็กที่ 2 ของบัตรแม่เหล็ก ซึ่งเรียงลำดับความสำคัญจากซ้ายไปขวา โดยจะสังเกตเห็นว่ามีข้อมูลซึ่งเป็นลอจิก 0 หรือที่เรียกว่า "Clocking Bit" นำหน้าและปิดท้ายข้อมูลจริงอยู่ ซึ่งลักษณะการจัดเก็บข้อมูลของแทร็ก 2 จะเก็บข้อมูลด้วยการเข้ารหัสแบบ "มอดูโล 5" โดยในแต่ละชุดข้อมูลจะประกอบด้วยข้อมูล 5 บิต โดย 4 บิตแรก (D0-D3) เป็นรหัสข้อมูลแบบ BCD ส่วนบิตที่ 5 (D4) เป็นพาริตี้บิตแบบคี่ (ODD) เพื่อใช้ตรวจสอบความถูกต้องของข้อมูลในแต่ละไบต์ (5 บิต) ที่อ่านได้ หากค่าของบิตพาริตี้ผิดพลาดแสดงว่าการอ่านข้อมูลนั้นล้มเหลว โดยเริ่มทำการอ่านเมื่อสัญญาณ ENABLE เริ่มเป็นศูนย์ และทำการอ่านข้อมูลในทุกๆขอบขาลงของสัญญาณนาฬิกา ซึ่งข้อมูลในส่วนก่อนเริ่มต้นและหลังจากการสิ้นสุดของการอ่านนี้จะมีค่าเป็นศูนย์(สัญญาณ DATA=1) ก็คือ "Clocking Bit" นั่นเอง โดยข้อมูลในส่วนนี้เราไม่ต้องสนใจ แต่ให้ตรวจสอบและรอจนกว่าข้อมูลจะเป็นข้อมูลเริ่มต้น (START : 0BH) หรือ 11010 ในที่นี้เรียงลำดับความสำคัญจากซ้ายไปขวา ซึ่งบิตเริ่มต้นของรหัส START หรือ 0BH ต้องเริ่มด้วย 1 เป็นบิตแรกเสมอ ดังนั้นในการอ่านเราต้องรอจนกว่าจะพบสัญญาณข้อมูลเป็นลอจิก 1 (สัญญาณ DATA=0 เพราะกลับสภาวะกันอยู่) จึงเริ่มทำการเก็บข้อมูลชุดละ 5 บิตไปเรื่อยๆจนถึงรหัสจบ(0FH) ซึ่งเมื่อพบรหัสจบแล้วจะมีข้อมูลตามมาอีก 1 ไบต์ ซึ่งเป็นข้อมูลสำหรับตรวจสอบความผิดพลาดของการอ่านข้อมูลในแทร็ก 2 เรียกว่า "LRC" ค่าของ LRC สามารถหาได้จากการนำเอาข้อมูลในแต่ละไบต์ (พาริตี้บิตไม่ผิดพลาด) ตั้งแต่เริ่มต้นจนสิ้นสุดมาทำการ XOR กัน โดยครั้งแรกให้นำไบต์เริ่มต้น (OBH) ทำการ XOR กับศูนย์ แล้วนำผลลัพธ์ที่ได้ไป XOR กับไบต์ถัดไปเรื่อยๆตามลำดับ ซึ่งหากผลลัพธ์สุดท้ายที่ได้ไม่เท่ากับค่าของ LRC ที่อ่านมาได้แสดงว่าการอ่านข้อมูลทั้งหมดล้มเหลว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับข้อมูลของรหัส BCD ที่ใช้สำหรับเครื่องอ่านบัตรแม่เหล็กนี้ จะเป็นข้อมูลชุดละ 5 บิต เป็นข้อมูลจริง 4 บิต และเป็นบิตตรวจสอบพาริตี้อีก 1 บิต ซึ่งมีจำนวนทั้งหมด 16 อักขระ โดยแบ่งได้ดังนี้

- ข้อมูลรหัสตัวเลข 10 อักขระ (0-9)
- ข้อมูลบอกสถานะเริ่มต้น แบ่งชุดข้อมูล และสิ้นสุดข้อมูล 3 อักขระ
- รหัสควบคุม 3 อักขระ

ตารางที่ 2.11 แสดงข้อมูลของรหัส BCD สำหรับเครื่องอ่านบัตรแม่เหล็ก

PARITY	D3	D2	D1	D0	CHARACTER	FUNCTION
1	0	0	0	0	0 (0H)	DATA
0	0	0	0	1	1 (1H)	DATA
0	0	0	1	0	2 (2H)	DATA
1	0	0	1	1	3 (3H)	DATA
0	0	1	0	0	4 (4H)	DATA
1	0	1	0	1	5 (5H)	DATA
1	0	1	1	0	6 (6H)	DATA
0	0	1	1	1	7 (7H)	DATA
0	1	0	0	0	8 (8H)	DATA
1	1	0	0	1	9 (9H)	DATA
1	1	0	1	0	: (AH)	CONTROL
0	1	0	1	1	; (BH)	START SENTINEL
1	1	1	0	0	< (CH)	CONTROL
0	1	1	0	1	= (DH)	FIELD SEPARATOR
0	1	1	1	0	> (EH)	CONTROL
1	1	1	1	1	? (FH)	END SENTINEL

2.5.2 การตรวจสอบความผิดพลาด (Error detection)

ในระบบการสื่อสารข้อมูลในทางอุดมคติจะไม่ให้เกิดความผิดพลาดในการรับ-ส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่ในความเป็นจริง ทางฝ่ายรับข้อมูลจะเกิดความผิดพลาดขึ้น เนื่องจากข้อมูลที่ได้รับไม่ตรงกับข้อมูลที่ส่งมา ดังนั้นทางด้านฝ่ายรับข้อมูลจะต้องมีวิธีการตรวจสอบข้อมูล โดยในระบบการสื่อสารข้อมูลจะมีวิธีการตรวจสอบ 3 วิธี

- ตรวจสอบบิตพาริตี้ (Parity bit checking)
- การตรวจซ้ำในแนวตั้งและแนวนอน(Vertical and Longitude Redundancy Checking)
- การตรวจซ้ำแบบวงกลม (Cyclic Redundancy Checking)

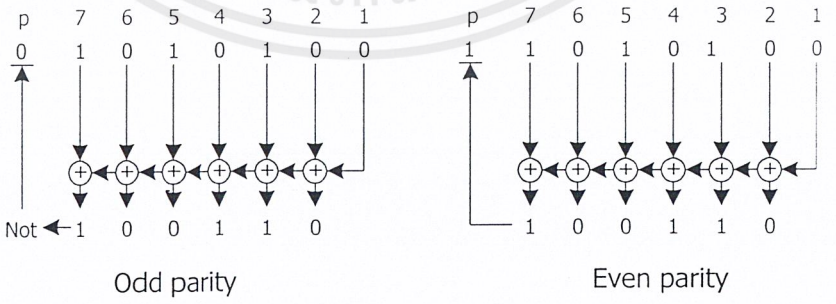
การตรวจสอบแบบพาริตี้บิต

การตรวจสอบแบบพาริตี้บิตเป็นวิธีที่ง่ายที่สุดในการตรวจสอบความผิดพลาดของข้อมูล โดยพาริตี้บิต 1 บิต จะรวมกับตัวอักษรแต่ละตัว จึงเป็นการเพิ่มบิตตัวอักษรอีก 1 บิต ดังนั้นบิตที่เพิ่มขึ้นเรียกว่า พาริตี้บิต (Parity bit)

1. พาริตี้คู่ (Even parity) คือจำนวน "1" รวมทั้งพาริตี้บิต เป็นจำนวนคู่
2. พาริตี้คี่ (Odd parity) คือจำนวน "1" รวมทั้งพาริตี้บิต เป็นจำนวนคี่

เพื่อความเข้าใจได้แสดงตัวอย่างการหาพาริตี้บิต ของข้อมูลตัวอักษร T ซึ่งในรหัสแอสกี้มีค่าเท่ากับ 54H (1010100)

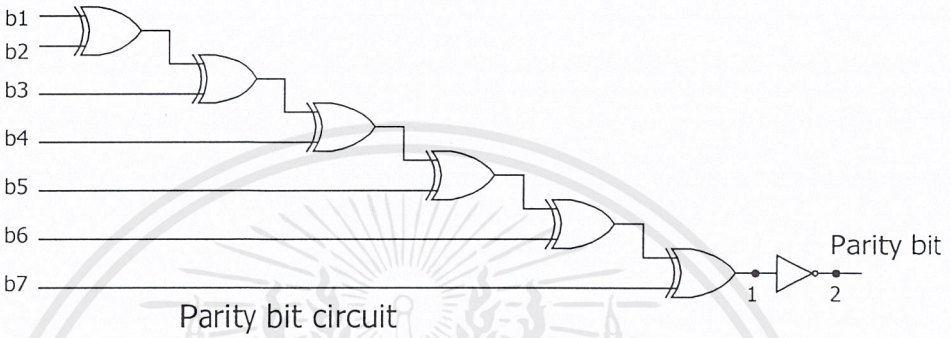
- เมื่อเป็นพาริตี้คี่		- เมื่อเป็นพาริตี้คู่
P 7 6 5 4 3 2 1	ตำแหน่งบิต	P 7 6 5 4 3 2 1
0 1 0 1 0 1 0 0	บิตข้อมูล	1 1 0 1 0 1 0 0



รูปที่ 2.8 แสดงหลักการสร้างพาริตี้บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะเห็นได้ว่าบิตพาริตี้จะเกิดขึ้นจากการเปรียบเทียบบิตข้อมูลแต่ละบิตโดยการใช้ เอ็กคลูซีฟออร์ (Exclusive or) ในการเปรียบเทียบข้อมูลแต่ละบิต ซึ่งในการเอ็กคลูซีฟกันจะได้เอาต์พุตเท่ากับ “0” เมื่ออินพุทเหมือนกัน ดังแสดงไว้ในรูปที่ 2.8 และในรูปที่ 2.9



รูปที่ 2.9 แสดงวงจรการสร้างพาริตี้บิต

จากรูปที่ 2.9 เป็นการสร้างพาริตี้บิตโดยใช้ฮาร์ดแวร์ (Hardware) ณ. จุดเอาต์พุต 1 จะได้พาริตี้คู่ ส่วนเอาต์พุต 2 จะได้พาริตี้คี่ โดยพาริตี้บิตจะใช้ทั้งฝ่ายรับ-ส่งข้อมูล เพื่อตรวจสอบความผิดพลาดข้อมูลแต่ละตัวอักษร

ข้อดีของการตรวจสอบแบบพาริตี้บิต คือ เป็นวิธีการง่าย (Simplicity) แต่จะเกิดข้อเสียเมื่อจำนวนบิตเป็นจำนวนคู่ ถ้าด้านรับเกิดความผิดพลาด การตรวจสอบพาริตี้จะไม่สามารถตรวจสอบได้ถูกต้อง เช่น ถ้าเกิด 2 บิต เกิดการเปลี่ยนแปลงค่าพาริตี้บิตจะยังคงเดิม

การตรวจซ้ำในแนวตั้งและแนวนอน

การตรวจซ้ำในแนวตั้ง(Vertical Redundancy Checking :VCR) เป็นการตรวจสอบความผิดพลาดข้อมูลโดยใช้พาริตี้ในการกำหนด ถ้าการส่งเกิดความผิดพลาดในตัวอักขระ ดังนั้นการตรวจซ้ำในแนวตั้งบางครั้ง เรียกว่า พาริตี้ตัวอักขระ (Character parity)

การตรวจซ้ำในแนวนอน(Horizontal or Longitude Redundancy Checking :HRC or LRC) เป็นการตรวจสอบความผิดพลาดข้อมูลโดยใช้พาริตี้ในการกำหนด ถ้าการส่งเกิดความผิดพลาดในข้อความ ดังนั้น LRC บางครั้งจึงเรียกว่า พาริตี้ข้อความ (Massage parity) จากตารางที่ 2.12 ค่าของการตรวจซ้ำในแนวนอน LRC จะได้เท่ากับ 2FH หรือ (00101111) ส่วนค่าการตรวจซ้ำในแนวตั้ง VCR แต่ละตัวอักษรจะคิดในแนวตั้ง ส่วนบิต LRC จะคำนวณใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แนวนอนในกลุ่มของตัวอักขระที่ทำเป็นข้อความ (เช่น THE CAT) เรียกว่ากลุ่มของข้อมูล (Block of data) ดังนั้นลำดับบิตของ LRC เรียกว่า การตรวจสอบกลุ่มของตัวอักขระ (Block Check Character : BCC) หรือกลุ่มการตรวจสอบลำดับกลุ่ม (Block Check Sequence :BCS)

ตารางที่ 2.12 แสดงการตรวจซ้ำในแนวตั้งและแนวนอน

Character		T	H	E	SP	C	A	T	LRC (→)
Hex		54	48	45	20	43	41	54	2F
LSB	B0	0	0	1	0	1	1	0	1
	B1	0	0	0	0	1	0	0	1
	B2	1	0	1	0	0	0	1	1
	B3	0	1	0	0	0	0	0	1
	B4	1	0	0	0	0	0	1	0
	B5	0	0	0	1	0	0	0	1
MSB	B6	1	1	1	0	1	1	1	0
VRC (↓)	B7	0	1	0	0	0	1	0	0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
ความรู้พื้นฐานเกี่ยวกับไมโครคอนโทรลเลอร์และการสร้างฐานเวลา

3.1 สถาปัตยกรรมของไมโครคอนโทรลเลอร์แบบชิพเดี่ยวตระกูล MCS-51

ไมโครคอนโทรลเลอร์แบบชิพเดี่ยว(Single Chip Microcontroller) คือ ไมโครคอมพิวเตอร์แบบที่มีขนาดเล็กโดยบรรจุไว้ในแผงวงจรรวม (Integrated Circuit) เพียงชิพเดียว เหมาะสำหรับงานควบคุมอุปกรณ์อื่นๆแบบอัตโนมัติ เพราะผู้ใช้สามารถเขียนโปรแกรมควบคุมการทำงานได้ตามต้องการ ไมโครคอนโทรลเลอร์แบบชิพเดี่ยวตระกูล 51 หรือ MCS-51 ได้แก่ เบอร์ 8051,8052 และอื่นๆ โดย MCS-51 มีข้อดีดังนี้

- สามารถนำเอาข้อมูลมา AND,OR หรือทำ Complement ทั้งแบบทีละ 8 บิตและ 1 บิต
- สามารถใช้กับหน่วยความจำโปรแกรม (Program Memory) ซึ่งเป็นหน่วยความจำที่ใช้สำหรับเก็บชุดคำสั่งที่จะให้ MCS-51 ทำงานได้สูงสุด 64 กิโลไบต์(64 × 1024 ไบต์)ทำให้เขียนโปรแกรมควบคุมการทำงานได้มาก
- สามารถต่อกับหน่วยความจำข้อมูล(Data Memory) ซึ่งเป็นหน่วยความจำสำหรับเก็บข้อมูลในระหว่างการทำงานของโปรแกรมได้สูงสุด 64 กิโลไบต์
- ใน 8051 และ 8751 มีหน่วยความจำโปรแกรมจำนวน 4 กิโลไบต์ (ใน 8052 และ 8752 มีหน่วยความจำโปรแกรมจำนวน 8 กิโลไบต์) อยู่ภายในวงจรรวมทำให้ไม่ต้องต่อหน่วยความจำโปรแกรมภายนอก ระบบรวมทั้งหมดจึงมีขนาดเล็กและสัญญาณรบกวนจากภายนอกจะทำให้ MCS-51 ทำงานผิดพลาดได้ยาก
- มีพอร์ตแบบขนาน (Parallel Port) สำหรับข้อมูลเข้าและออกจำนวน 32 บิต โดยข้อมูลแต่ละบิตเป็นอิสระต่อกัน
- มีวงจร Timer/Counter ขนาด 16 บิต 2 ชุด (8052 มี 3 ชุด) ที่ทำงานในโหมดต่างๆได้ถึง 4 โหมด
- มี Universal Asynchronous Receivevr Transmitter (UART) สำหรับรับ-ส่งข้อมูลอนุกรม(Serial) แบบ Full Duplex ที่สามารถเลือกรูปแบบการรับส่งข้อมูลได้ 4 แบบ
- มีแหล่งกำเนิดสัญญาณขอขัดจังหวะการทำงานของโปรแกรม (Interrupt Request Signal) 6 แหล่ง ซึ่งสามารถกระโดดไปทำงานตอบสนองการขัดจังหวะ (Interrupt Service Routine) ได้ต่างๆ กัน 5 ตำแหน่ง
- สามารถเลือกการทำงานให้อยู่ในโหมดของ Idle และ Power Down ซึ่งจะประหยัดการใช้กำลังไฟในการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในโครงงานนี้ได้ใช้ไมโครคอนโทรลเลอร์เบอร์ AT89S8252 ของบริษัท ATMEL ซึ่งเป็นไมโครคอนโทรลเลอร์ในตระกูลเดียวกันกับ MCS-51 เช่นกัน แต่มีส่วนที่แตกต่างจาก 8051 ดังนี้

- 8 กิโลไบต์ สำหรับหน่วยความจำแบบแฟลช(Flash Memory)
- SPI การเชื่อมต่อแบบอนุกรม สำหรับการดาวน์โหลด(Down load) โปรแกรม
- อายุการใช้งาน : สามารถเขียนและลบได้ถึง 1,000 ครั้ง
- 2 กิโลไบต์ สำหรับ EEPROM
- อายุการใช้งาน : สามารถเขียนหรือลบได้ถึง 1,000 ครั้ง
- ช่วงการทำงานที่ 4.0- 6 โวลต์
- ช่วงทำงานเต็มที่ 0-24 MHz
- สามารถโปรแกรม Memory Lock ได้ 3 ระดับ
- 256 × 8 บิต ใน Ram ภายใน (Internal RAM)
- สามารถโปรแกรมทางด้านอินพุต/เอาต์พุต(I/O) ได้ 32 เส้น
- Timer/Counter 16 บิต มี 3 ตัว
- มีอินเตอร์รัพต์ 9 แหล่ง
- สามารถโปรแกรมผ่านช่องสัญญาณอนุกรมแบบ UART ได้
- มี SPI เชื่อมต่อแบบอนุกรม
- กินกำลังไฟต่ำในโหมด Idle และ Power Down
- อินเตอร์รัพต์กลับคืนมาจากกำลังไฟฟ้าตกได้
- สามารถโปรแกรมเป็น watchdog timer ได้
- มีพอยน์เตอร์ข้อมูล 2 ชุด

3.1.1 โครงสร้างของ AT89S8252

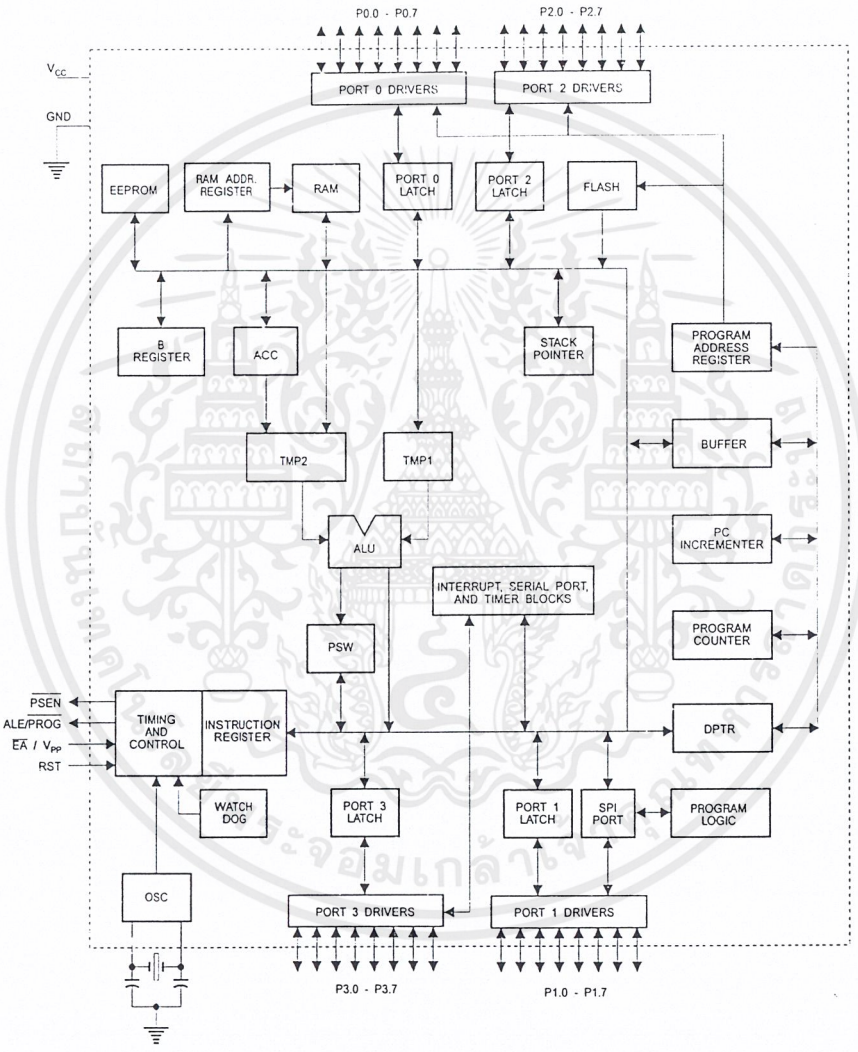
ภายใน AT89S8252 จะประกอบด้วยเกทต่าง ๆ กัน เช่น AND,OR,NOR Gate ซึ่งเกทเหล่านี้จะถูกนำเอามาออกแบบให้มีหน้าที่การทำงานต่าง ๆ เช่น วงจรถอดรหัสคำสั่ง,วงจรสร้างสัญญาณนาฬิกา โครงสร้างภายในของ AT89S8252 จะประกอบด้วยส่วนต่าง ๆ ดังรูปที่ 3.1

- CPU หรือตัวประมวลผล ส่วนนี้จะมีวงจรที่ทำหน้าที่สร้างสัญญาณควบคุมในการติดต่อกับส่วนอื่น ๆ เรียกว่า วงจรควบคุม สัญญาณที่สร้างขึ้นได้แก่ สัญญาณสำหรับการติดต่อกับหน่วยความจำ , อุปกรณ์รับข้อมูลออกจากตัว AT89S8252 ซึ่งส่วนควบคุมการอินเทอรรัพท์ และส่วนควบคุมบัสก็เป็นส่วนหนึ่งของวงจรควบคุมการสร้างสัญญาณควบคุม จากส่วน CPU นี้จะทำการสร้างสัญญาณโดยการถอดรหัสจากคำสั่ง ตามที่มีการกำหนดไว้ และสัญญาณที่สร้างขึ้นมาจะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อ้างอิงกับสัญญาณนาฬิกาที่สร้างจากวงจรออสซิลเลเตอร์ เพื่อให้ทุก ๆ ส่วนในวงจรทำงานประสานกันอย่างถูกต้อง

ใน CPU นี้ยังประกอบด้วยส่วนย่อยอีกส่วนที่เรียกว่าส่วนประมวลผล ซึ่งจะทำหน้าที่ประมวลผลข้อมูล เช่น การบวก ลบ คูณ หรือหารข้อมูล แล้วนำเอาผลลัพธ์ที่ได้ไปเก็บไว้ในรีจิสเตอร์หรือหน่วยความจำที่ต้องการ



รูปที่ 3.1 ไดอะแกรมโครงสร้างของ AT89S8252

- หน่วยความจำ (Memory) มีไว้สำหรับจดจำข้อมูล การเอาข้อมูลไปเก็บในหน่วยความจำเรียกว่า การเขียนข้อมูล และการนำเอาข้อมูลออกจากหน่วยความจำเรียกว่า การอ่านข้อมูล
- อุปกรณ์อินพุทและเอาท์พุท เป็นส่วนที่จะใช้ส่งข้อมูลเข้าหรือออก ทำให้AT89S8252

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ติดต่อกับภายนอกได้ ดังไดอะแกรม อุปกรณ์อินพุทและเอาท์พุทได้แก่ 4 I/O Port,Timer 0,Timer 1 ,Serial Port การทำงานของแต่ละส่วนมีดังนี้

4 I/O Port เป็นที่สำหรับรับส่งข้อมูล ซึ่งเป็นสัญญาณดิจิตอลเข้าหรือออกจากตัวไมโครคอนโทรลเลอร์ โดยมีอยู่ 4 พอร์ต แต่ละพอร์ตสามารถรับส่งข้อมูลได้ 8 บิต คือพอร์ต 0 ถึง พอร์ต 3 โดยที่พอร์ต 0 และพอร์ต 2 จะใช้สำหรับการส่งค่าตำแหน่ง (Address) ของหน่วยความจำที่ต้องการติดต่อและพอร์ต 0 จะใช้รับส่งข้อมูลเมื่อติดต่อกับหน่วยความจำได้ด้วย แต่ต้องใช้การควบคุมการถอดรหัสมาจากแต่ละคำสั่งที่ให้คอมพิวเตอร์ทำงานนั่นเอง และสัญญาณทั้งหมดจะอ้างอิงกับสัญญาณนาฬิกา

- Timer 0 และ Timer 1 เป็นวงจรนับที่สามารถกำหนดให้ทำการนับจำนวนไซเคิลของสัญญาณที่ต่อจากภายนอกของ 8051 หรือจำนวนไซเคิลของสัญญาณนาฬิกาภายใน ก็ได้ค่าจากการนับ จะอ่านค่าหรือตั้งค่าเริ่มต้นของการนับได้โดย CPU

- Serial Port หรือพอร์ตอนุกรม CPU จะอ่านและเขียนข้อมูลกับ Serial Port เป็นแบบ 8บิต แต่ข้อมูลจะถูกส่งออกจากขา TXD และในการรับข้อมูลเข้าก็จะรับเข้ามาทีละบิตทางขา RXD แล้วจัดเรียงใหม่เป็น 8 บิตเพื่อให้ CPU อ่านไปใช้งาน

3.1.2 การจัดการหน่วยความจำ

หน่วยความจำของ 8051 แบ่งออกไว้เป็น 2 แบบตามลักษณะของการใช้งานคือ

1. **Program Memory** เป็นหน่วยความจำที่ใช้เก็บคำสั่งในรูปรหัสภาษาเครื่อง(Machine Language)ซึ่งถ้าต้องการให้ทำงานก็จะอ่านข้อมูลที่เก็บไว้ในหน่วยความจำประเภทนี้เข้าไปถอดรหัสแล้วสร้างสัญญาณควบคุมส่วนอื่น ๆ ตามการทำงานของแต่ละคำสั่งนั้น หน่วยความจำแบบนี้จะต้องเป็นแบบ Read Only Memory (ROM) และผู้ใช้ต้องเขียนข้อมูลในแต่ละตำแหน่งของหน่วยความจำเป็นรหัสภาษาเครื่องตามลำดับการทำงานที่ต้องการ (หน่วยความจำแบบ ROM เป็นแบบ Non volatite ซึ่งเมื่อปิดไฟแล้วข้อมูลก็ไม่มีการสูญหาย) การเขียนข้อมูลลงไปบน ROM จะต้องใช้เครื่องมือพิเศษ จำนวนตำแหน่งสูงสุดของหน่วยความจำแบบนี้ที่จะใช้งานได้คือ 65536 ตำแหน่ง (Address) จะเขียนเป็นเลขฐาน 16 ได้ตั้งแต่ 0000H ถึง FFFFH ผู้ใช้จะเลือกได้ว่าเป็นตำแหน่งของ ROM ที่อยู่ภายในหรือภายนอก ถ้าต้องการให้ทำงานตามคำสั่งที่เก็บไว้ใน ROM ภายในก็ให้ป้อนสัญญาณสภาวะลอจิก High (1) เข้าที่ขา EA แต่ถ้าต้องการให้ทำงานในโปรแกรมที่เก็บไว้ใน ROM ภายนอกก็ให้ต่อลอจิก Low (0) เข้าที่ขา EA

2. **Data Memory** เป็นหน่วยความจำที่จะใช้สำหรับพัก , เก็บข้อมูล แล้วเรียกมาใช้ใหม่ใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระหว่างการทำงานของการอ่านหรือเขียนข้อมูลจากหน่วยความจำจะกระทำโดยคำสั่งที่เก็บไว้ใน Program Memory หน่วยความจำแบบนี้เป็นประเภท Random Access Memory (RAM) ถ้ามีไฟเลี้ยงอยู่ข้อมูลที่เก็บไว้จะไม่สูญหาย แต่ถ้าปิดเครื่องหรือไม่จ่ายไฟให้แก่ RAM แล้ว ข้อมูลใน RAM ก็จะสูญหายไป การสูญหายของข้อมูลไม่ได้หมายความว่าไม่มีอะไรอยู่เลย แต่เป็นการที่มีข้อมูลใหม่ ซึ่งไม่ใช่ข้อมูลที่เก็บไว้เดิมเข้ามาอยู่แทนที่ เช่นเดิมเก็บข้อมูล 18H ไว้ตำแหน่ง 1900H เมื่อปิดไฟแล้วเปิดใหม่ ข้อมูลที่ตำแหน่ง 1900H จะไม่ใช่ 18H อายเป็นค่าอะไรก็ได้ ซึ่งเรียกการเกิดลักษณะแบบนี้ว่า ข้อมูลสูญหายไป

ตารางที่ 3.1 แผนผัง SFR และค่ารีจิสเตอร์เมื่อเกิดการรีเซต

Table 1. AT89S8252 SFR Map and Reset Values

0F8H								0FFH
0F0H	B 00000000							0F7H
0E8H								0EFH
0E0H	ACC 00000000							0E7H
0D8H								0DFH
0D0H	PSW 00000000					SPCR 000001XX		0D7H
0C8H	T2CON 00000000	T2MOD XXXXXX00	RCAP2L 00000000	RCAP2H 00000000	TL2 00000000	TH2 00000000		0CFH
0C0H								0C7H
0B8H	IP XX000000							0BFH
0B0H	P3 11111111							0B7H
0A8H	IE 0X000000		SPSR 00XXXXXX					0AFH
0A0H	P2 11111111							0A7H
98H	SCON 00000000	SBUF XXXXXXXX						9FH
90H	P1 11111111						WMCON 00000010	97H
88H	TCON 00000000	TMOD 00000000	TL0 00000000	TL1 00000000	TH0 00000000	TH1 00000000		8FH
80H	P0 11111111	SP 00000111	DP0L 00000000	DP0H 00000000	DP1L 00000000	DP1H 00000000	SPDR XXXXXXXX	PCON 0XXX0000	87H

จากตารางที่ 3.1 เป็นการแสดงแผนผังของรีจิสเตอร์พิเศษ (SFR : Special Function Register) และค่าของรีจิสเตอร์เมื่อการรีเซ็ท โดยใน AT89S82582 จะมีรีจิสเตอร์ที่พิ่มขึ้นมาคือ

- Timer2 Register เป็นส่วนควบคุมและบอกสถานะบิต ประกอบไปด้วย T2CON(0C814) T2MOD(009H) คู่รีจิสเตอร์ RCAP2H(0CAH) และ RCAP2L(0CBH) ใช้ในการจับและโหลดรีจิสเตอร์ซ้ำสำหรับ Timer2 ในโหมด 16 บิต auto-reload

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Watchdog and Memory control Register (WMCON))ประกอบไปด้วยบิตควบคุมสำหรับ timer อยู่ในแอดเดรส 96H

ตารางที่ 3.2 วอทช์ด็อกและรีจิสเตอร์ควบคุมหน่วยความจำ

WMCON(96H)

PS2	PS1	PS0	EEMWE	EEMEN	DPS	WDTRST	WDTEN
7	6	5	4	3	2	1	0

- PS0,PS1,PS2 บิตฟรีสเกลเลอร์สำหรับ watchdog timer เมื่อทั้ง 3 บิตเป็น "0" watchdog timer มีคาบเวลา 16 ms
- EEMWE Set bit นี้เป็น "1" ก่อนเริ่มการเขียนลงบน EEPROM ด้วยคำสั่ง MOVX
  "0" เมื่อใช้ซอฟท์แวร์ที่เขียนลง EEPROM เรียบร้อยแล้ว
- EEMEN เป็น "1" เป็นการเข้าถึง EEPROM ภายในชิพชุดคำสั่ง MOVX พร้อมด้วย DPTR จะเข้าถึงภายในชิพ
  เป็น "0" ชุดคำสั่ง MOVX พร้อมด้วย DPTR จะเข้าถึงหน่วยความจำข้อมูลภายนอก
- DPS (Data Pointer Register Select) DPS = 0 เลือก DP0 DPS = 1 เลือก DP1
- WDTRST บิตนี้ set เป็น "1" โดยใช้ซอฟแวร์ พัลซ์จะถูกผลิต ไม่รีเซต watchdog timer
- RDY/BSY/ บิต WDTRST จะถูกรีเซตอัตโนมัติ
  RDY/BSY = "1" แสดงว่า EEPROM พร้อมโปรแกรม ขณะโปรแกรมทำงาน บิต
  RDY/BSY = "0" และเป็น "1" อัตโนมัติ เมื่อโปรแกรมเสร็จสิ้น
- WDTEN (Watchdog Timer Enable bit) ถ้ามีค่า = "1" เป็นการ Enable Watchdog Timer
- SPI (Serial Peripheral Interface) register ประกอบไปด้วย SPCR และ SPSR เป็นรีจิสเตอร์ควบคุม

ตารางที่ 3.3 รีจิสเตอร์ควบคุม SPI

SPCR(0D5H)

SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0
7	6	5	4	3	2	1	0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- SPIE (SPI interrupt Enable) เชื่อมต่อกับ ES bitโดย SPIE = " 1" SPI interrupt จะ enable

- SPE (SPI Enable) SPI = " 1" enable ช่องสัญญาณ SPI และต่อกับ SS,MOSI,MISOและ SCK

- DORD (Data Order) = " 1" เลือก LSB ก่อนการส่งข้อมูล

- MSTR (Master/Slave Select) MSTR = " 1" เลือก Master SPI Mode

- CPOL (Clock Polarity) CPOL= " 1" เป็น high เมื่อ Idle

- CPHA (Clock Phade) เป็นบิตควบคุม Clock และข้อมูลระหว่าง Master กับ Slave

SPR0	SPR0	SPR1	SCK=FOSC
	0	0	4
SPR1	0	1	16
	1	0	64
	1	1	128

- SPSR - SPI Status Register

ตารางที่ 3.4 รีจิสเตอร์แสดงสภาวะของ SPI

SPSR(0AAH)

SPIF	WCOL	-	-	-	-	-	-
7	6	5	4	3	2	1	0

- SPIF (SPI Interrupt Flag) เมื่อการโอนย้ายข้อมูลแบบอนุกรมสมบูรณ์ SPIF = "1" และ ES = "1"

- WCOL (Write Collision Flag) บิตนี้เป็น "1" ถ้า SPI ข้อมูลรีจิสเตอร์เขียนระฟหว่างการโอนย้ายข้อมูล ผลลัพธ์การอ่าน SPDR register จะไม่ถูกต้อง

- SPDR - SPI Data Register = 86H

Data Memory – EEPROM and RAM

ประกอบไปด้วย 2Kไบต์ ของ EEPROM ภายในชิพสำหรับเก็บข้อมูล และ 256 ไบต์ ของ RAM ช่วง 128 ไบต์ ขึ้นไปของ RAM จะเป็นพื้นที่ขนานไปกับ SFR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อคำสั่งที่เข้าถึงตำแหน่งภายในเหนือ 7FH แอดเดรสโหมดจะใช้ชุดคำสั่งเฉพาะให้ CPU เข้าถึงช่างเหนือ 128 ไบต์ ขึ้นไปของ RAM หรือ SFR ชุกคำสั่งที่ใช้ direct addressing เข้าถึงพื้นที่ SFR

ภายในชิพ EEPROM หน่วยความจำข้อมูลจะถูกเซตค่าที่บิต EEMEN ในรีจิสเตอร์ WMCON ที่ SFR แอดเดรสที่ 96H EEPROM แอดเดรสในช่วง 000H ถึง 77FH ชุดคำสั่ง MOVX จะใช่เข้าถึง EEPROM การเข้าถึงหน่วยความจำข้อมูลภายนอกชิพ ด้วยชุดคำสั่ง MOVX บิต EEMEN ต้องเซตค่าเป็น"0"

บิต EEMWE ใน WMCON ถ้าเซตให้เป็น "1" จะทำให้ EEPROM สามารถถูกเขียนได้ การใช้ซอฟแวร์ต้องรีเซต EEMWE เป็น "0" ไซเคิลการเขียนข้อมูลลง EEPROM ในการโปรแกรมแบบโหมดอนุกรมจะมีเวลาของตัวเองโดยประมาณ 2.5 ms การเขียนลง EEPROM สามารถดูได้จาก การอ่านบิต RDY/BSY/= "1" หมายความว่า EEPROM มีไซเคิลการเขียนสมบูรณ์แล้ว

3.1.3 การจัดวางขาและหน้าที่ต่างๆ



รูปที่ 3.2 การจัดวางขาของ AT89S8252

ไมโครคอนโทรลเลอร์ที่บรรจุอยู่ในวงจรรวมแบบ Dual Inline Package(DIP) ซึ่งแต่ละข้างของ 8051 มีขาอยู่ข้างละ 20 ขารวมทั้งหมด 40 ขานั้นจะใช้งานต่างๆ กันดังนี้ คือ

- V_{CC} ขา 40 เป็นขาที่ต้องป้อนไฟเลี้ยง +5 โวลต์เข้าไปเพื่อให้วงจรรวมทำงานได้ ระดับโวลเตจของลอจิก 0 และ 1 ของขา 8051 จึงต่อเข้ากับอุปกรณ์ลอจิกแบบ TTL ได้โดยตรง
- V_{SS} ขา 20 เป็นขาที่ต้องต่อกับกราวนด์ของแหล่งจ่ายไฟ การต่ออุปกรณ์ทั้งหมดจะต้องมีกราวนด์ของอุปกรณ์ต่อเข้าด้วยกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **Port 0** เป็นพอร์ตขนานขนาด 8 บิต อยู่ที่ขา 39 ถึง 32 เริ่มจากบิต 0 ถึงบิต 7 ตามลำดับ ดังในรูปที่ 3.2 แต่ละขาจะเขียนว่า P0.0,P0.1,...,P0.7 ซึ่ง P0.0 หมายถึงบิต 7 ของพอร์ต 0 ซึ่งเป็นบิตที่มีนัยสำคัญสูงสุด(Most Significant) และ P0.0 คือบิต 0 ของพอร์ต 0 เป็นบิตที่มีนัยสำคัญต่ำสุด(Least Significant) และพอรร์ต 0 นี้ใช้ได้ทั้งการรับ-ส่งตำแหน่งและข้อมูลกับหน่วยความจำหรือใช้เป็นพอร์ตรับส่งข้อมูลก็ได้ ข้อมูลที่ส่งออกทางพอร์ต 0 จะถูก Latch ไว้ที่ขาของพอร์ต โครงสร้างแต่ละบิตของพอร์ต 0 เป็นแบบ Open Drain Bidirectional พอร์ต 0 จะใช้งานได้หลายอย่างดังนี้
  1. ใช้สำหรับส่งค่าตำแหน่งหน่วยความจำภายนอกที่ต้องการติดต่อด้วย ตำแหน่งหน่วยความจำสูงสุดที่ติดต่อได้คือ 64 กิโลไบต์ มีค่าตำแหน่งหน่วยความจำ 16 บิตของเลขฐานสอง ค่าตำแหน่งหน่วยความจำ 8 บิตล่างจะถูกส่งออกไปทางพอร์ต 0 และ 8 บิตบนจะส่งออกไปทางพอร์ต 2
  2. ใช้รับ-ส่งข้อมูลกับหน่วยความจำข้อมูล หรือใช้รับข้อมูลจาก หน่วยความจำโปรแกรม
  3. ใช้รับ-ส่งข้อมูลผ่านทางพอร์ตโดยตรง ในกรณีที่ไม่มีการใช้หน่วยความจำของหน่วยความจำโปรแกรมหรือหน่วยความจำข้อมูลภายนอก
- **Port 1** เป็นพอร์ตขนานขนาด 8 บิต ที่ขา P1.0 ถึง P1.7 (ขา 1-8) P1.0 หมายถึง บิต 0 ขิงพอร์ต 1 ซึ่งเป็นบิตนัยสำคัญต่ำสุด และบิต P1.7 หมายถึงบิตที่ 7 ของพอร์ต 1 ซึ่งเป็นบิตนัยสำคัญสูงสุด

  P1.0 T2 (เป็นพอร์ตอินพุตรับการนับจากภายนอกไปยัง Timer/Counter2) เอาต์พุตของสัญญาณนาฬิกา

  P1.1 T2EX(ทำหนาที่จับ/กระตุ้นการโหลดซ้ำและควบคุมทิศทางให้ Timer/Counter2)

  P1.4 SS(Slave port ในการเลือกอินพุต)

  P1.5 MOSI (เอาต์พุตของข้อมูลตัวหลัก (Master) รับมาจากอินพุตของข้อมูลตัวรอง (Slave) สำหรับช่อง SPI)

  P1.6 MISO (อินพุตของข้อมูลหลัก ไปยังเอาต์พุตที่ข้อมูลตัวรอง)

  P1.7 SCK (สัญญาณนาฬิกาของเอาต์พุตที่ตัวหลัก โดยรับอินพุตมาจากขาสัญญาณนาฬิกาตัวรอง)

  P1.4 ถึง P1.7 ใช้สำหรับ SPI Channel

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **Port 2** เป็นพอร์ตขนานขนาด 8 บิต คือขา P2.0 ถึง P2.7 (บิต 0 ถึงบิต 7 ของพอร์ต 2) ลักษณะโครงสร้างจะเหมือนกับ Port แตกต่างกันที่ใน Port 2 นั้นภาค Driver จะใช้งานเพียง 2 ลักษณะ คือ
  1. ใช้ส่งค่าตำแหน่งหน่วยความจำภายนอกที่ต้องการติดต่อ ค่าตำแหน่งนี้เป็น 8 บิต บนของค่าตำแหน่ง
  2. ใช้เป็นพอร์ทรับและส่งข้อมูลกับอุปกรณ์ภายนอก
- **Port3** คือขา P3.0 ถึง P3.7 หรือขา 10-17 ตามลำดับ แต่ละบิตของพอร์ต 3 จะมีการทำงานอื่นๆ ดังนี้

  P3.0 RXD(Serail Input Port) เป็นขาที่ใช้รับข้อมูลแบบอนุกรม

  P3.1 TXD(Serail Output Port) เป็นขาที่ใช้ส่งข้อมูลแบบอนุกรม

  P3.2 INT0(External Interrupt) ใช้รับสัญญาณขัดจังหวะจากภายนอก

  P3.3 INT1(Internal Interrupt) ใช้รับสัญญาณขัดจังหวะจากภายใน

  P3.4 T0 (Timer/Counter0 External Input) ขารับสัญญาณเข้าไปยังวงจร Timer/Counter 0 ที่ทำหน้าที่นับจำนวนไซเคิลของสัญญาณนาฬิกาก็ได้

  P3.5 T1 (Timer/Counter1 External Input) ขารับสัญญาณเข้าไปยัง Timer Counter1 ซึ่งมีการทำงานเหมือนกับ T0

  P3.6 WR (External Data Memory Write Strobe) ขาสัญญาณควบคุมการเขียนข้อมูลไปยังหน่วยความจำสำหรับข้อมูลภายนอก 8051

  P3.7 RD (Extrnal Data Memory Read Strobe) ขาสัญญ่ณควบคุมการอ่านข้อมูลจากหน่วยความจำสำหรับข้อมูลภายนอก
- **RST** ขารีเซตนี้จะใช้ทำการรีเซตการทำงาน ที่ขารีเซตภายในจะมีตัวต้านทานต่อระหว่างขานี้กัยกราวนด์ ถ้าป้อนสัญญาณที่มีสภาวะลอจิก 1 เข้าไปที่ขานี้จะเป็นการรีเซตการทำงาน ดังนั้นจึงสามารถต่อตัวเก็บประจุภายนอกระหว่างขารีเซตกับไฟเลี้ยง +5 โวลต์ เพื่อให้เกิดการรีเซตเมื่อเริ่มป้อนไฟเลี้ยง ซึ่งเรียกว่า Power on reset การรีเซตจะทำให้ค่าในรีจิสเตอร์ต่างๆ เปลี่ยนไปเป็นค่าหนึ่งดังในตารางที่ 3.1

  เมื่อสัญญาณที่ขารีเซตกลับเป็น 0 ก็จะออกจากการรีเซต จะเริ่มทำงานจากคำสั่งที่อยู่ในหน่วยความจำโปรแกรมตำแหน่ง 0000H เพราะค่าของรีจิสเตอร์ PC (Program Counter) ซึ่งชี้ตำแหน่งโปรแกรมที่จะทำงานถูกเปลี่ยนให้เป็น 0000H ดังนั้นจะต้องเขียนโปรแกรมมาเก็บไว้ที่ตำแหน่ง 0000H ในเครื่องไมโครคอมพิวเตอร์แบบบอร์ดเดี่ยว (Single Board Microcomputer)จะมีโปรแกรมที่เขียนเก็บไว้เริ่มจากตำแหน่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

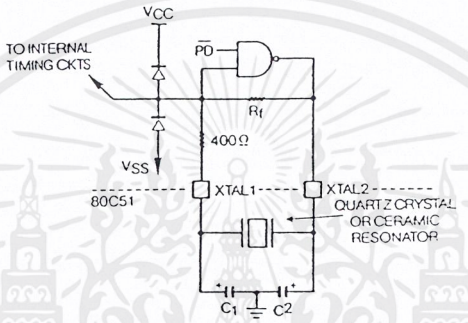
0000H นี้ เรียกว่า มอนิเตอร์โปรแกรม(Monitor Program)ที่จะคอยรับการกดแป้นพิพ์(Keyboard)และแสดงผลทางตัวแสดงผล(Display)แบบ 7 Segment

- **ALE** (Address Latch Enable) ขานี้จะส่งสัญญาณที่มีความถี่ 1/6 เท่าของสัญญาณนาฬิกาจากออสซิลเลเตอร์ สัญญาณนี้จะส่งออกมาตลอดเวลา ยกเว้นบางครั้งของการติดต่อกับหน่วยความจำสำหรับข้อมูลภายนอก 8051 สัญญาณนี้จะใช้บอกอุปกรณ์ภายนอก 8051 ว่าขณะนี้สัญญาณนี้แอคทีฟ(เป็นลอจิก 1)จะมีการส่งข้อมูลที่เป็น 8 บิตล่างของตำแหน่งหน่วยความจำภายนอก 8051 ที่ต้องการติดต่อออกไปทางพอร์ต 0 อุปกรณ์ภายนอกจะใช้สัญญาณนี้ในการ Latch ข้อมูลไว้เพราะพอร์ต 0 จะส่งค่าตำแหน่งหน่วยความจำออกมาเพียงชั่วขณะเท่านั้น ซึ่งในเวลาต่อมาพอร์ต 0 จะใช้รับ-ส่งข้อมูลกับหน่วยความจำภายนอก สัญญาณ ALE จะสามารถต่อเข้ากับอุปกรณ์ TTL ชนิด LS ได้ถึง 8 อินพุต
- **PSEN** (Program Store Enable) เป็นขาที่ 29 ขานี้ปกติจะให้ลอจิก 1 แต่จะส่งลอจิก 0 เมื่อต้องการอ่านคำสั่ง(Fetch Instruction)ที่จะนำไปทำงานมาจากหน่วยความจำสำหรับโปรแกรมภายนอก 8051 ในกรณีที่อ่านคำสั่งซึ่งเก็บอยู่ในหน่วยความจำสำหรับโปรแกรมภายใน 8051 แล้วสัญญาณนี้จะไม่เปลี่ยนลอจิกเป็น 0 ขา PSEN นี้สามารถต่อไปยังขาอินพุตของ TTL ชนิด LS ได้ถึง 8 อินพุต
- **EA** (External Access) ขา 31 ขานี้เป็นขาอินพุตที่ต่อไปยังวงจร Timing and Control เพื่อควบคุมการสร้างสัญญาณ PSEN ถ้าป้อนสัญญาณลอจิก 0 เข้าไปที่ขา EA นี้แสดงว่าโปรแกรมในตำแหน่ง 0000H ถึง 0FFFH ที่ต้องการให้ทำงานถูกเก็บไว้ภายนอก จะต้องสร้างสัญญาณออกไปยังภายนอก เพื่อทำการ Fetch คำสั่งเข้ามาทำงาน แต่ถ้าสัญญาณที่ป้อนให้ขา EA เป็นลอจิก 1 หมายความว่าโปรแกรมในตำแหน่ง 0000H ถึง 0FFFH ถูกเก็บไว้ภายใน 8051 การทำงานในตำแหน่งหน่วยความจำช่วงนี้จะอ่านคำสั่งต่างๆ จาก ROM
- **XTAL1** ขาที่ 19 ขานี้จะต่อเข้ากับขาของ Inverting Amplifier(วงจรขยายแบบป้อนกลับเฟสสัญญาณ)ที่ประกอบเป็นวงจรออสซิลเลเตอร์ จะเห็นวงจรภายในของออสซิลเตอร์ NAND gate จะทำหน้าที่เป็นวงจรขยายาแบบกลับเฟสของสัญญาณที่จะควบคุมให้มีการออสซิลเลตหรือไม่ก็ขึ้นกับสัญญาณ PD ซึ่งต่อมาจากบิต PD ของรีจิสเตอร์ PCON ถ้าต้องการใช้สัญญาณนาฬิกา(Clock Signal)จากภายนอกมาเป็นสัญญาณนาฬิกาควบคุมการทำงานก็ให้ป้อนสัญญาณเข้ามาที่จุดนี้ แต่ถ้าต้องการใช้วงจรออส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซิลเลเตอร์ภายในก็ให้ต่อคริสตัลหรือเซรามิกเรโซเนเตอร์ดังรูปที่ 3.3 โดยตัวเก็บประจุในวงจรควรมีค่าประมาณ 20 pF

- **XTAL2** ขาที่ 18 ขานี้เป็นจุดเอาต์พุตของวงจรขยายแบบกลับเฟสสัญญาณที่ประกอบเป็นวงจรออสซิสเลเตอร์(อินพุตคือขา XTAL1) ถ้าจะใช้สัญญาณนาฬิกาที่สร้างมาจากภายนอกมาเป็นสัญญาณนาฬิกาให้ปล่อยขานี้ลอยไว้ แล้วป้อนสัญญาณนาฬิกาจากภายนอกเข้ามาที่ขา XTAL1 ดังรูปที่ 3.3



รูปที่ 3.3 วงจรออสซิลเลเตอร์ภายใน AT89S8252

3.2 การสร้างฐานเวลาให้แก่ MCS-51

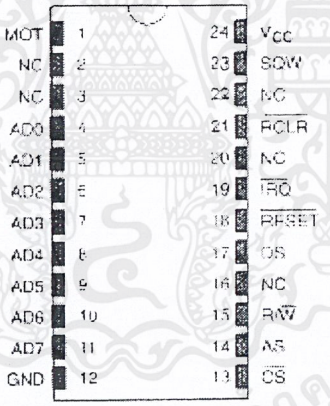
ในที่นี้จะใช้ชิพที่เป็นนาฬิกาที่สามารถส่งข้อมูลเวลาในขณะใดๆ (ชิพ RTC) ให้แก่ไมโครคอนโทรลเลอร์ได้ โดยในโครงงานนี้ได้เลือกใช้ชิพ RTC เบอร์ 12887A ซึ่งมีลิเธียมแบตเตอรี่และคริสตัลออสซิลเลเตอร์อยู่ภายในชิพโมดูล โดยมีอายุการใช้งานของแบตเตอรี่ภายในเป็นเวลา 10 ปี ความเที่ยงตรงของนาฬิกาภายใน RTC คือ คลาดเคลื่อนไม่เกิน 1 นาทีต่อเดือนในขณะที่ไม่มีแหล่งจ่ายไฟให้เลย

3.2.1 คุณสมบัติของชิพ RTC เบอร์ DS12887A

- ทำหน้าที่นับวินาที นาที ชั่วโมง วันในสัปดาห์ วันที่ของเดือน เดือน ปี รวมทั้งคำนวณปีอธิกสุรทินให้เองโดยอัตโนมัติจนถึงปี ค.ศ. 2100
- ข้อมูลสามารถคงอยู่ได้เป็นเวลาถึง 10 ปี แม้ไม่มีไฟเลี้ยง
- มี lithium , quartz , และวงจรสนับสนุนในตัว
- แสดงเวลา วัน เดือน ปี และสัญญาณเตือน เป็นเลขฐานสอง หรือ BCD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- มีทั้งรูปแบบ 12 ชั่วโมง หรือ 24 ชั่วโมง โดยแบบ 12 ชั่วโมงจะมีข้อมูลบอกว่าเป็นเวลากลางวันหรือกลางคืน(AM/PM)
- สามารถเลือกระหว่าง Motorola และ Intel bus timing
- มีหน่วยความจำแบบ RAM 128 ไบต์ โดย
  - 14 ไบต์ สำหรับนาฬิกา และรีจิสเตอร์ควบคุม(Control register)
  - 114 ไบต์ สำหรับ RAM myj;wx
- สามารถส่งสัญญาณอินเตอร์รัพต์ไปยัง CPU ได้
- มีการอินเตอร์รัพท์ 3 แบบ
  - การส่งสัญญาณเตือน Time-of-day โดยสามารถจะเลือกอินเตอร์รัพต์ตั้งแต่วินาทีละครั้งจนถึงวันละครั้ง
  - แบบเป็นคาบ(Periodic rates) เลือกได้ตั้งแต่ 122 ms ถึง 500 ms
  - การอินเตอร์รัพต์หลังจากสิ้นสุดการอัพเดท (update)ข้อมูลวันเวลา



รูปที่ 3.4 การจัดวางขาของ DS12887A

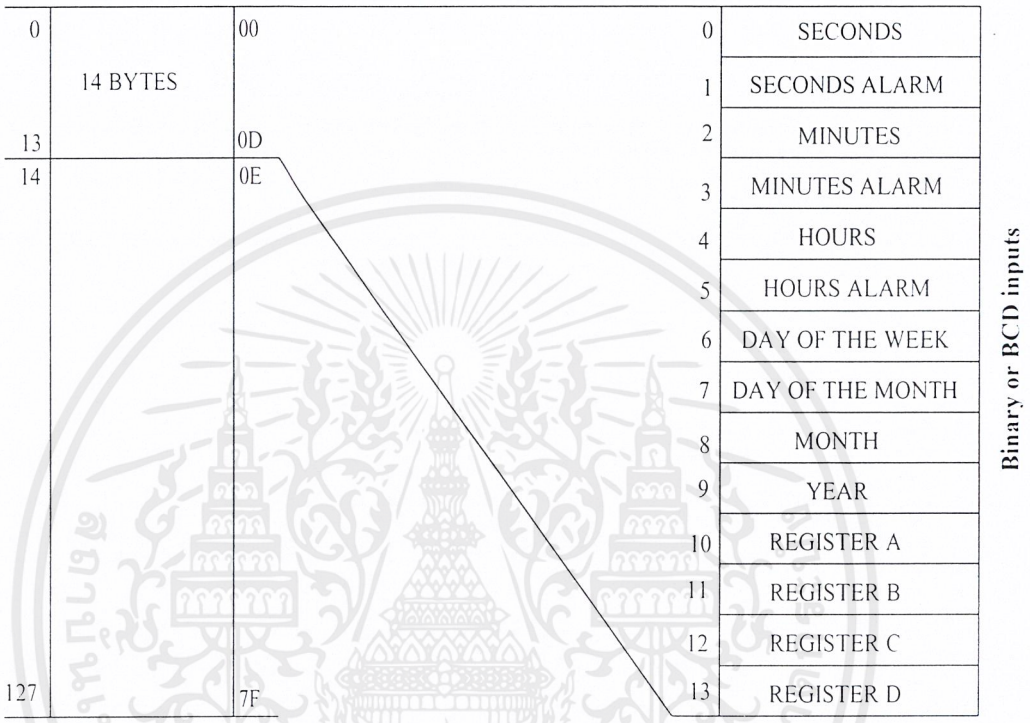
3.2.2 แผนผังแสดงตำแหน่งต่างๆ ใน RTC

แผนผังแสดงตำแหน่งต่างๆ ใน RTC แสดงดังในรูปที่ 3.5 ในแผนผังประกอบด้วย RAM ที่ใช้ได้ 114 ไบต์ และ 10 ไบต์สำหรับเก็บเวลาของ RTC วัน เดือน ปี และข้อมูลสัญญาณเตือน และอีก 4 ไบต์สำหรับการควบคุมและแสดงสถานะ ทั้ง 128 ไบต์สามารถเขียนหรืออ่านได้โดยตรง โดยมีข้อยกเว้นดังนี้

1. รีจิสเตอร์ C และ D อ่านได้อย่างเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. บิต 7 ของรีจิสเตอร์ A อ่านได้อย่างเดียว
3. บิตสูงของไบต์วินาทีอ่านได้อย่างเดียว



รูปที่ 3.5 แผนผังแสดงตำแหน่งต่างๆ ใน RTC

3.2.3 รีจิสเตอร์ควบคุม(Control register)

- รีจิสเตอร์ A

MSB **LSB**

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
UIP	DV2	DV1	DV0	RS3	RS2	RS1	RS

- **UIP**(Update In Progress bit) : เป็นแฟลกสถานะที่สามารถใช้เตือนได้ โดยเมื่อบิต UIP เป็น 1 แสดงว่าจะมีการอัพเดทเกิดขึ้นหลังจากนั้น แต่ถ้าบิต UIP เป็น 0 จะไม่มีการอัพเดทเกิดขึ้นอย่างน้อย 244 us

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **DV2, DV1, DV0** : ทั้ง 3 บิตนี้จะใช้เปิดหรือปิดออสซิลเลเตอร์ โดยรูปแบบ 010 จะเป็นเพียงรูปแบบเดียวที่จะเปิดออสซิลเลเตอร์และอนุญาตให้ RTC ทำการเก็บเวลา
- **RS3, RS2, RS1, RS0** : ทั้ง 4 บิตนี้สามารถใช้งานได้ดังนี้
    1. ทำให้มีการอินเตอร์รัพต์โดยใช้บิต PIE
    2. ทำให้มีเอาต์พุตที่ขา SQW โดยใช้บิต SQWE
    3. อีนาเบิลทั้งคู่ในเวลาเดียวกันและอัตราที่เท่ากัน
    4. ทำให้ไม่มีทั้งคู่

- **รีจิสเตอร์ B**

MSB **LSB**

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
SET	PIE	AIE	UIE	SQWE	DM	24/12	DSE

- **SET** : เมื่อบิต SET เป็น 0 การนับเวลาจะเป็นไปตามปกติโดยจะทำการนับวินาทีละครั้ง แต่เมื่อบิต SET เป็น 1 RTC จะหยุดนับเวลา และสามารถที่จะทำการเริ่มต้นเวลาใหม่ได้
- **PIE** : เป็นบิตที่จะใช้อนุญาตให้บิต PF ในรีจิสเตอร์ C ทำการขับขา IRQ เป็น low เมื่อบิต PIE เป็น1 จะเกิดการอินเตอร์รัพต์แบบเป็นคาบ(Periodic interrupt) โดยการขับขา IRQ เป็น low ซึ่งอัตราการอินเตอร์รัพต์จะถูกกำหนดโดยบิต RS3-RS0 ของรีจิสเตอร์ A ถ้าบิต PIE เป็น 0 จะป้องกัน IRQ เอาต์พุต จากการอินเตอร์รัพต์แบบเป็นคาบ แต่บิต PF จะยังคงเซตที่อัตราคาบ
- **AIE** (Alarm Interrupt Enable bit) : เมื่อบิตนี้ถูกเซตเป็น 1 จะอนุญาตให้บิต AF ในรีจิสเตอร์ C ยืนยันสถานะขา IRQ ไว้ การอินเตอร์รัพต์แบบสัญญาณเตือนจะเกิดเมื่อไบต์เวลาทั้ง 3 ตัวเท่ากับไบต์สัญญาณเตือนทั้ง 3 ตัวซึ่งประกอบด้วยรหัสสัญญาณเตือน “don’t care” เป็นเลขฐานสอง 11XXXXXX เมื่อบิต AIE เป็น 1 บิต AF จะไม่เริ่มต้นสัญญาณ IRQ
- **UIE** (Update Ended Interrupt Enable bit) : เป็นบิตที่ใช้อนุญาตให้บิต UF ในรีจิสเตอร์ C ทำการยืนยันสถานะขา IRQ
- **SQWE** (Square Wave Enable bit) : เมื่อบิตนี้เซตเป็น 1 จะมีสัญญาณคลื่นสี่เหลี่ยมที่มีความถี่ที่ถูกกำหนดโดยบิต RS3-RS0 ถูกขับออกมาที่ขา SQW
- **DM** (Data Mode bit) : เป็นบิตที่ใช้เลือกว่าจะแสดงข้อมูลเวลาเป็นแบบเลขฐานสองหรือแบบ BCD โดยถ้าเป็น 1 ข้อมูลจะเป็นแบบเลขฐานสอง ถ้าเป็น 0 ข้อมูลจะเป็นแบบ BCD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **24/12** ถ้าบิตนี้เป็น 1 ข้อมูลจะเป็นแบบ 24 ชั่วโมง ถ้าเป็น 0 ข้อมูลจะเป็นแบบ 12 ชั่วโมง
- **DSE** (Daylight Saving Enable bit) : ถ้าถูกเซตเป็น 1 จะยอมให้มีการอัพเดทแบบพิเศษ คือ ในวันอาทิตย์แรกของเดือนเมษายนเวลาจะเพิ่มจาก 1:59:59 AM เป็น 3:00:00 AM และในวันอาทิตย์สุดท้ายของเดือนตุลาคมเมื่อเวลาถึง 1:59:59 AM แล้วจะเปลี่ยนเป็น 1:00:00 AM แต่ถ้าบิต DSE นี้เป็น 0 จะไม่มีการอัพเดทเช่นนี้เกิดขึ้น

- รีจิสเตอร์ C

MSB **LSB**

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
IRQF	PF	AF	UF	0	0	0	0

- **IRQF** (Interrupt Request Flag bit) : บิตนี้จะถูกเซตเป็น 1 เมื่อเงื่อนไขต่อไปนี้เป็นจริง

$$PF = PIE = 1$$

$$AF = AIE = 1$$

$$UF = UIE = 1$$

กล่าวคือ $IRQF = (PF \bullet PIE) + (AF \bullet AIE) + (UF \bullet UIE)$

เวลาใดก็ตามที่บิต IRQF เป็น 1 ขา IRQ จะถูกทำให้อยู่ในสถานะ low

- **PF** (Periodic Flag bit) : การที่บิต PF นี้ถูกเซตเป็น 1 จะไม่ขึ้นกับสถานะของบิต PIE เมื่อทั้งบิต PF และบิต PIE เป็น 1 ทั้งคู่ สัญญาณอินเตอร์รัพต์จะแอคทีฟและจะเป็นการเซตบิต IRQF
- **AF** (Alarm Interrupt Flag bit) : ถ้าบิต AF เป็น 1 จะบ่งบอกว่าเวลาขณะนั้นเท่ากับเวลาที่ตั้งเตือนไว้ และถ้าบิต AIE เป็น 1 ด้วยแล้ว ขา IRQ จะเป็น low และบิต IRQF จะเป็น 1
- **UF** (Update Ended Interrupt Flag bit) : จะเซตหลังการอัพเดทแต่ละครั้ง เมื่อบิต UIE และบิต UF เป็น 1 จะทำให้บิต IRQF เป็น 1
- **BIT3 ถึง BIT0** : ทั้ง 3 บิตนี้จะไม่ใช้ โดยจะเป็น 0 อยู่เสมอ

- รีจิสเตอร์ D

MSB **LSB**

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
VRT	0	0	0	0	0	0	0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **VRT** (Valid RAM and Time bit) : แสดงถึงสภาวะของแบตเตอรี่ที่ต่อกับขา V_{BAT} บิตนี้ไม่สามารถเขียนได้และควรจะเป็น 1 เสมอเมื่อทำการอ่าน แต่ถ้าบิตนี้เป็น 0 แสดงว่าแหล่งจ่ายพลังงานภายใน(internal lithium energy source) ถูกใช้จนหมดแล้ว ข้อมูลของ RTC และข้อมูลใน RAM จะมีปัญหา
- **BIT 6 ถึง BIT 0** : จะเป็น 0 เสมอ

3.2.4 ตำแหน่งของเวลา วัน เดือน ปี และ สัญญาณเตือน(alarm)

ข้อมูลเวลา วัน เดือน และปีจะได้มาจากการอ่านไบต์ที่เหมาะสม การกำหนดหรือเริ่มต้นเวลา วัน เดือน ปีและสัญญาณเตือน ทำได้โดยการเขียนลงในแรมไบต์ที่เหมาะสม ไบต์ข้อมูลเวลา วัน เดือน ปี และสัญญาณเตือนจะสามารถเป็นได้ทั้งแบบเลขฐานสอง(Binary)หรือแบบ BCD (Binary-Code Decimal)อย่างใดอย่างหนึ่ง

ตารางที่ 3.5 แสดงโหมดข้อมูลของ RTC

ADDRESS LOCATION	FUNCTION	DECIMAL RANGE	RANGE	
			BINARY DATA MODE	BCD DATA MODE
0	Seconds	0-59	00-3B	00-59
1	Seconds Alarm	0-59	00-3B	00-59
2	Minutes	0-59	00-3B	00-59
3	Minutes Alarm	0-59	00-3B	00-59
4	Hours 12-hr,Mode	1-12	01-0C AM,81-8C PM	01-12 AM,81-92 PM
	Hours 24-hr,Mode	0-23	00-17	00-23
5	Hours Alarm 12-hr,Mode	1-12	01-0C AM,81-8C PM	01-12 AM,81-92 PM
	Hours Alarm 24-hr,Mode	0-23	00-17	00-23
6	Day of the week Sunday=1	1-7	01-07	01-07
7	Day of Month	1-31	01-1F	01-31
8	Month	1-12	01-0C	01-12
9	Year	0-99	00-63	00-99

ก่อนที่จะทำการเขียนค่าลงในรีจิสเตอร์เวลา วัน เดือน ปีและสัญญาณเตือน ควรจะทำการกำหนดให้บิต SET ของรีจิสเตอร์ B เป็น 1 เพื่อที่จะป้องกันไม่ให้เกิดการอัพเดทข้อมูลในขณะที่กำลังเข้าสู่การทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกจากจะเขียนรีจิสเตอร์เวลา วัน เดือน ปีและสัญญาณเตือนทั้งสิบตัวในรูปแบบที่เลือก (เลขฐานสอง หรือ BCD)แล้ว บิต DM(data mode bit) ของรีจิสเตอร์ B จะต้องถูกเซตให้อยู่ในระดับลอจิกที่เหมาะสม ไบต์เวลา วัน เดือน ปีและสัญญาณเตือน จะต้องมีโหมดข้อมูลเหมือนกัน หลังจากที่บิต DM ได้ถูกเขียนเพื่ออนุญาตให้ RTC มีการอัพเดทไบต์เวลา วัน เดือน และปี บิต SET ในรีจิสเตอร์ B ควรจะเป็น 0 ในการเริ่มต้นแต่ละครั้ง RTC จะทำการอัพเดทในโหมดที่เลือกทั้งหมด โดยโหมดข้อมูลจะไม่เปลี่ยนหากไม่มีการเริ่มต้นไบต์ทั้งสิบใหม่ ตารางที่ 3.6 แสดงตำแหน่งของเวลา วัน เดือน ปี และสัญญาณเตือนในรูปแบบเลขฐานสองและแบบ BCD

บิต 24-12 จะไม่เปลี่ยนแปลงจนกว่าจะมีการเริ่มต้นตำแหน่งชั่วโมงใหม่ เมื่อเลือกรูปแบบ 12 ชั่วโมง บิตสูงของไบต์ชั่วโมงจะแสดง PM เมื่อเป็นลอจิก 1 ใน 1 วินาทีจะมีการส่งข้อมูล 11 ไบต์ไปเช็คสถานะสัญญาณเตือน (Alarm condition) ถ้าการอ่านข้อมูลเกิดขึ้นในระหว่างการอัพเดท อาจเกิดปัญหาขึ้นโดย เวลา วัน เดือน ปีที่อ่านได้อาจไม่สัมพันธ์กัน แต่ความเป็นไปได้ที่จะอ่านข้อมูลผิดนั้นมีน้อยมาก เพราะมีหลายวิธีที่สามารถหลีกเลี่ยงการอ่านผิดได้

ส่วนไบต์สัญญาณเตือนทั้ง 3 ไบต์ มีวิธีใช้ได้ 2 วิธี คือ

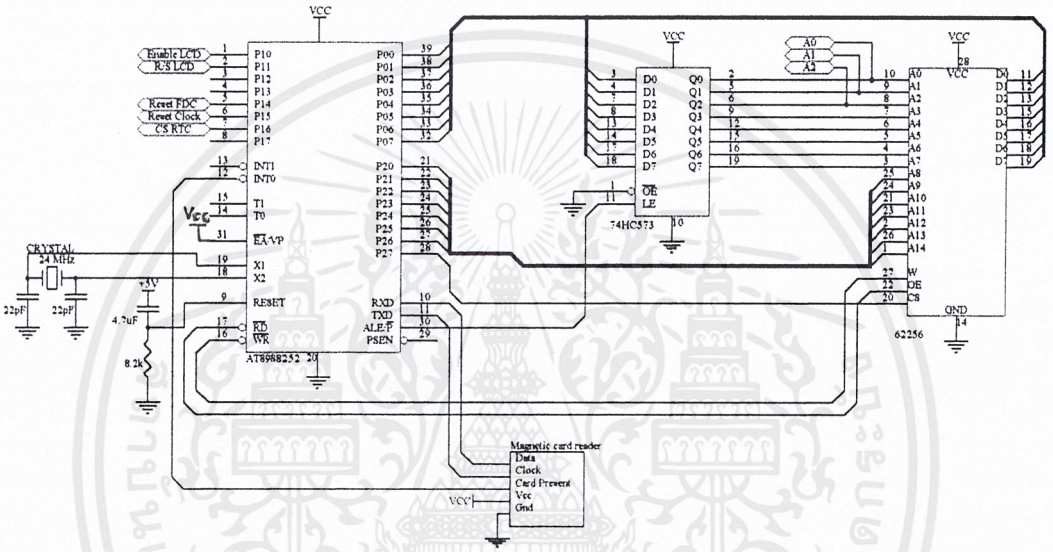
1. เมื่อกำหนดเวลาที่ต้องการให้มีการเตือนไปที่ตำแหน่งไบต์สัญญาณเตือนชั่วโมง นาที และวินาทีแล้ว จะเกิดการอินเตอร์รัพต์แบบสัญญาณเตือนเมื่อถึงเวลาที่ได้กำหนดไว้ในแต่ละวัน
2. ใส่สถานะ “don’t care” ลงไปในไบต์สัญญาณเตือน 3 ไบต์นี้ไบต์ใดไบต์หนึ่งหรือมากว่า 1 ไบต์ โดยรหัส “don’t care” คือเลขฐานสิบหกตั้งแต่ C0-FF เงื่อนไขของวิธีนี้ก็คือ
   - เมื่อใส่ “don’t care” ในไบต์สัญญาณเตือนชั่วโมง จะมีการเตือนทุกชั่วโมง
   - เมื่อใส่ “don’t care” ในไบต์สัญญาณเตือนชั่วโมง และนาที จะมีการเตือนทุกนาที
   - เมื่อใส่ “don’t care” ในไบต์สัญญาณเตือนทั้ง3 ไบต์ จะมีการเตือนทุกวินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การออกแบบและการทำงาน

4.1 ส่วนประมวลผล หน่วยความจำข้อมูล และส่วนอ่านบัตรแม่เหล็ก



รูปที่ 4.1 แสดงส่วนประมวลผล หน่วยความจำข้อมูลและส่วนหัวอ่านบัตรแม่เหล็ก

สำหรับในการทำงาน ได้ใช้ไมโครคอนโทรลเลอร์ตระกูล MCS-51 ซึ่งใช้เบอร์ 89S8252 โดยใช้คริสตอลออสซิลเลเตอร์ความถี่ 24 เมกะเฮิร์ต ทำให้มีความเร็วสูงเพียงพอที่จะใช้ในวงจรและทำงานร่วมกับไอซีฟล็อปปีดิสก์คอนโทรลเลอร์โดยทำการตอบสนองต่อไอซีได้ทัน และสำหรับไมโครคอนโทรลเลอร์เบอร์นี้ยังสามารถโหลดโปรแกรมการทำงานลงยังตัวไมโครคอนโทรลเลอร์และนำไปใช้ทำงานได้โดยตรง ทำให้สะดวกกับการใช้งานและการทดสอบโปรแกรม ซึ่งในการใช้งานโดยทั่ว ๆ ไปก็เหมือนกับเบอร์อื่น ๆ ในตระกูลนี้

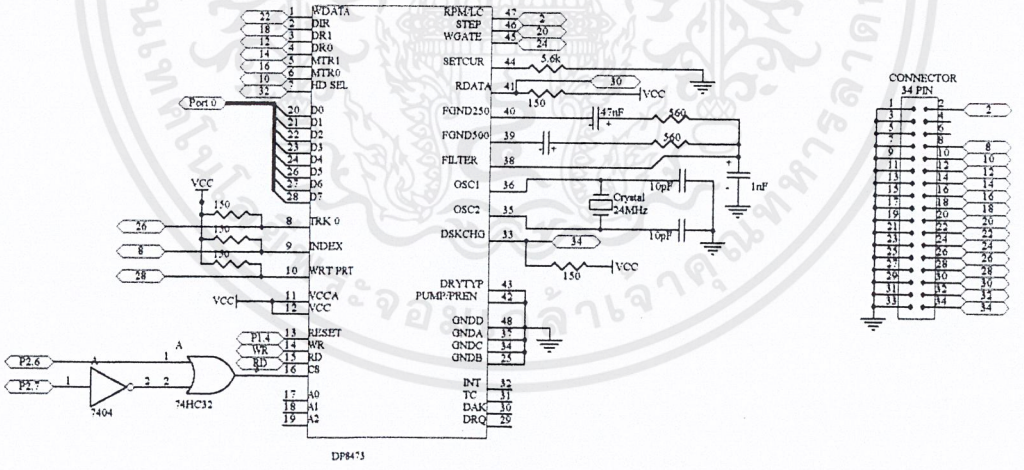
หน่วยความจำที่ใช้เป็นแบบแรม เบอร์ 62256 ซึ่งสามารถเก็บข้อมูลได้ทั้งหมด 32 กิโลไบต์ซึ่งเพียงพอและเผื่อสำหรับการขยายการใช้งานในวงจรที่มีการปรับปรุงต่อไป โดยจะใช้แอดเดรสบัสทั้งหมด 15 เส้น คือ A0 – A14 ดังรูปที่ 4.1 โดยไมโครคอนโทรลเลอร์นั้นใช้พอร์ต 0 ทำหน้าที่เป็นทั้งแอดเดรสบัสและบัสข้อมูล จึงได้ใช้ไอซีเบอร์ 74HC573 ทำหน้าที่แลตช์แอดเด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รสทางด้านไบต์ต่ำเอาไว้ในขณะที่พอร์ต 0 นั้นทำหน้าที่เป็นบัสข้อมูลต่อไป โดยนำสัญญาณจากขา ALE ของไมโครคอนโทรลเลอร์มาใช้ในการแลตช์

สำหรับส่วนรับข้อมูลจากบัตรแม่เหล็ก จะนำเครื่องอ่านบัตรแม่เหล็กมาเชื่อมต่อกับไมโครคอนโทรลเลอร์ดังแสดงในรูปที่ 4.1 โดยที่ขาสัญญาณ Card Present ต่อกับขา INT0 ของไมโครคอนโทรลเลอร์เพื่อใช้เป็นสัญญาณแสดงว่ามีการรูดบัตรผ่านหัวอ่าน โดยในขณะที่ไม่มีการรูดบัตร สัญญาณ Data,Clock,Card Present ทั้ง 3 สัญญาณจะอยู่ในสถานะ High ไมโครคอนโทรลเลอร์จะทำการรับข้อมูลจากบัตรเมื่อมีสัญญาณจากการอินเตอร์รัพต์เข้ามา แล้วทำการอ่านข้อมูลแบบอนุกรมทีละบิตทุกๆ ขอบขาลงของสัญญาณนาฬิกา โดยข้อมูลแต่ละตัวจะมีจำนวน 5 บิต ในการตรวจสอบความผิดพลาดในการอ่านข้อมูล จะตรวจสอบทั้งแบบพาริตี้โดยใช้พาริตี้คี่ และการตรวจสอบแบบ LRC โดยถ้าข้อมูลที่อ่านได้นั้นไม่มีความผิดพลาดก็จะทำการเก็บข้อมูลที่อ่านได้จากบัตรและเวลาที่รูดบัตรลงยังหน่วยความจำต่อไป

4.2 ส่วนวงจรควบคุมดิสก์



รูปที่ 4.2 แสดงส่วนวงจรควบคุมดิสก์

ได้เลือกใช้ไอซีควบคุมดิสก์เบอร์ DP8473 มาใช้ในการติดต่อกับฟล็อปปีดิสก์ โดยมีการควบคุมการทำงานโดยไมโครคอนโทรลเลอร์ ซึ่งจะส่งชุดคำสั่งควบคุมการทำงาน การกำหนดการทำงานเริ่มต้น เลือกโหมดการทำงาน ตั้งค่าต่างๆ โดย DP8473 จะรับข้อมูลจากฟล็อปปี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดิสก์นำเอามาแยกข้อมูลที่ถูกต้องจากในรูปของสัญญาณแบบ Serial ออกเป็นไบต์ที่ถูกต้อง โดยจะมีการชดเชยความผิดพลาดของการอ่านข้อมูลอยู่ในตัวอยู่แล้ว ทั้งยังสามารถเลือกอัตราของการส่งข้อมูลกับฟล็อปปีดิสก์ได้ตามชนิดของฟล็อปปีดิสก์ จึงสะดวกกับการใช้งานมากขึ้น

โดยจากวงจร ได้ติดต่อรับส่งข้อมูลกับไมโครคอนโทรลเลอร์ทางพอร์ต 0 โดยมีการควบคุมและเลือกการทำงานโดยการส่งสัญญาณ Reset,Write,Read หรือ Chip Select จากไมโครคอนโทรลเลอร์ ซึ่งสัญญาณจะต้องเป็นไปตามช่วงเวลาสอดคล้องกับ Timing Diagram ตามที่กำหนดในคู่มือการใช้งานของไอซี และในการเลือกรีจิสเตอร์ภายในตัว DP8473 ต้องใช้แอดเดรสบัสทั้งหมด 3 เส้นในการระบุตามตารางที่ 4.1

ตารางที่ 4.1 แสดงแอดเดรสของรีจิสเตอร์ภายในตัวไอซี DP8473

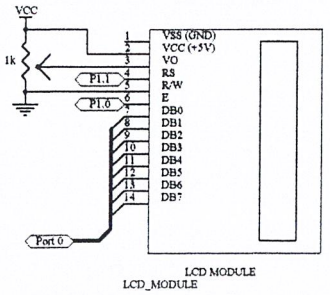
A0	A1	A2	R/W	Register
0	1	0	W	Drive Control Register
1	0	0	R	Main Status Register
1	0	1	R/W	Data Register
1	1	1	W	Data Rate Register
1	1	1	R	Disk Change Bit

ส่วนของฟิลเตอร์ ได้ใช้ตัวต้านทานและตัวเก็บประจุตามที่ได้กำหนดไว้ในคู่มือการใช้งาน คือมีค่าเป็นไปตามรูป ซึ่งจะสามารถนำเอาไปใช้งานได้กับทั้งอัตราการส่งข้อมูลแบบ 250 และ 500 kb/s ด้วย ขึ้นอยู่กับการเลือกใช้งานโดยการตั้งค่าอัตราการส่งข้อมูลได้ในรีจิสเตอร์แสดงอัตราของข้อมูล ทำให้สะดวกกับการใช้งานมากยิ่งขึ้น

4.3 ส่วนแสดงผล

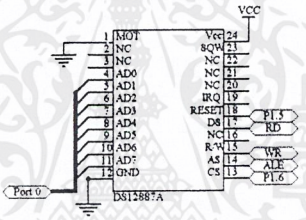
สำหรับส่วนที่ใช้ในการแสดงผลการทำงานและช่วยบอกลำดับขั้นตอนการใช้งานได้ใช้จอแสดงผล LCD แบบ 20 ตัวอักษร 2 บรรทัด ซึ่งในการควบคุมการแสดงผลในระหว่างขั้นตอนการทำงานต่าง ๆก็ทำได้โดยการตั้งค่าเริ่มต้นและส่งข้อมูลให้กับจอ LCD ผ่านทางพอร์ต 0 โดยใช้สัญญาณควบคุมการทำงานจากไมโครคอนโทรลเลอร์ในการควบคุมการเขียนแสดงผลเพื่อแสดงถึงการทำงานและแสดงผลเอาท์พุทที่ได้จากการอ่านเขียนข้อมูลบนดิสก์โดยมีรูปแบบการต่อดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 ส่วนแสดงผลของเครื่องบันทึกข้อมูล

4.4 ส่วนวงจรสร้างฐานเวลา (Real Time Clock)



รูปที่ 4.4 ส่วนวงจรสร้างฐานเวลา

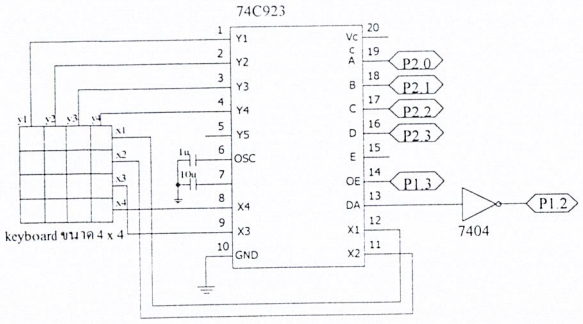
สำหรับวงจรส่วนที่ใช้เป็นฐานเวลาอ้างอิง ได้เลือกใช้ไอซี Real Time Clock เบอร์ DS12887 ซึ่งในการเชื่อมต่อกับไมโครคอนโทรลเลอร์เพื่อรับส่งข้อมูล คำสั่งและค่าแอดเดรสแบบ 8 บิตทางพอร์ท 0 โดยที่ไอซีตัวนี้มีแบตเตอรีภายใน ทำให้ค่าเวลาไม่สูญหายและยังเดินอย่างถูกต้องแม้ว่าจะไม่ได้จ่ายไฟให้กับวงจร และไอซีเบอร์นี้ก็มีความเที่ยงตรงและถูกต้องสูง

เมื่อเริ่มต้นการทำงาน จะต้องมีการตั้งค่าเริ่มต้นให้กับตัว RTC ซึ่งเป็นการเลือกว่าจะให้มีการทำงาน รูปแบบข้อมูลแบบไหน และในการเรียกใช้ข้อมูลตรงส่วนของฐานเวลานี้ ก็ทำได้โดยการเลือกให้ตัว RTC ส่งข้อมูลมายังไมโครคอนโทรลเลอร์ โดยใช้วิธีการแบบ Polling

4.5 ส่วนคีย์บอร์ดรับคำสั่งการทำงาน

สำหรับส่วนนี้ได้ใช้คีย์บอร์ดแบบ Matrix ขนาด 4 × 4 โดยใช้ไอซีเบอร์ 74HC923 ที่แปลงการกดคีย์ไปเป็นข้อมูลแบบไบนารี 4 บิต แล้วส่งข้อมูลที่ได้ให้แก่ไมโครคอนโทรลเลอร์ เพื่อแปลความหมายจากการกดคีย์ต่อไป

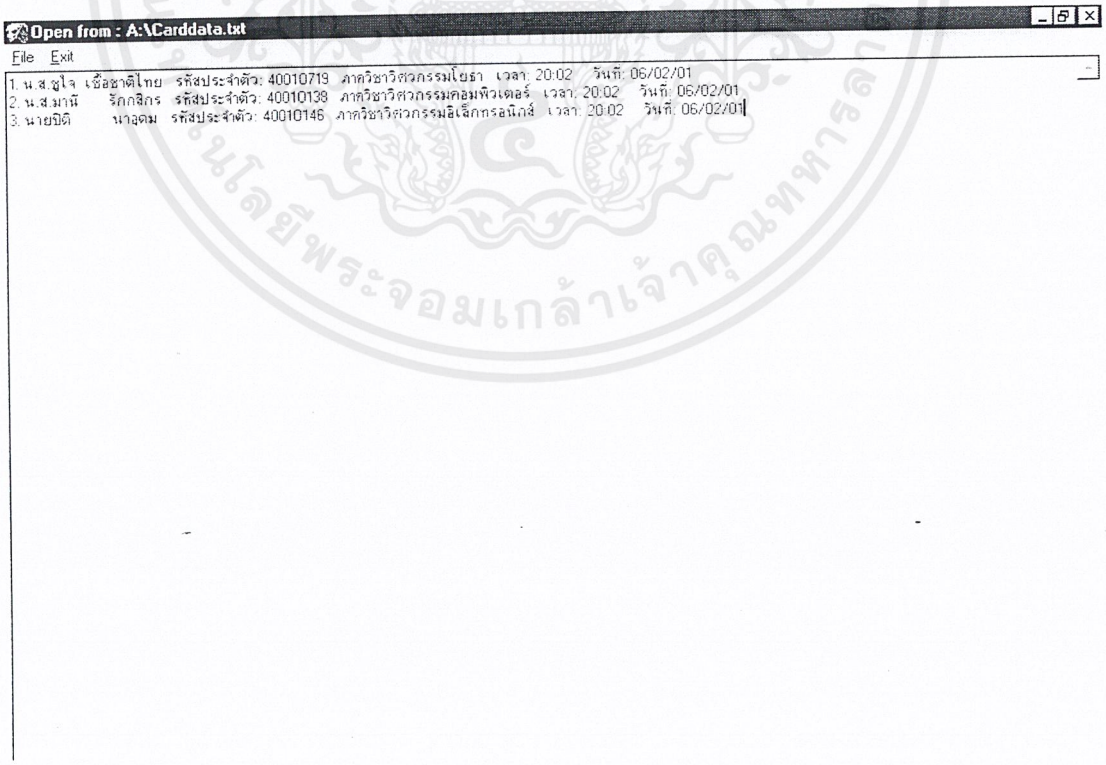
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 ส่วนคีย์บอร์ดรับคำสั่งการทำงาน

4.6 โปรแกรมแสดงผลบนจอคอมพิวเตอร์

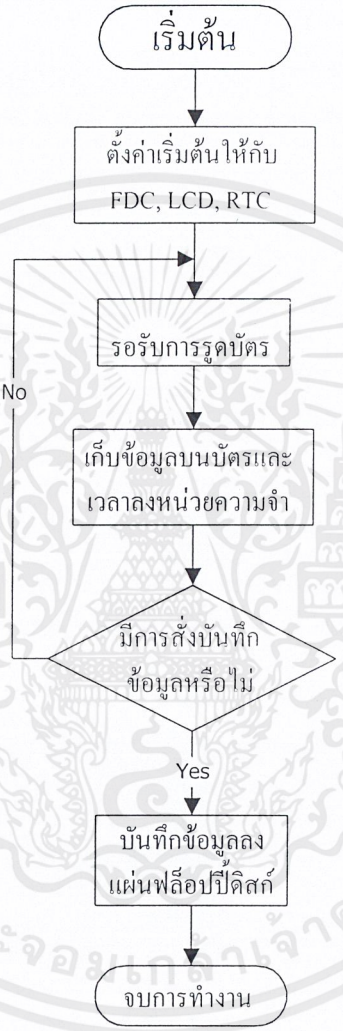
เมื่อทำการบันทึกข้อมูลลงบนดิสก์แล้ว ต้องนำเอาแผ่นดิสก์ที่ได้ไปเปิดในโปรแกรม ซึ่งใช้ภาษา Visual Basic ในการเขียน เพื่อใช้ในการแสดงข้อมูลของเจ้าของบัตร และวันเวลาที่ทำการรูดบัตร โดยที่เครื่องคอมพิวเตอร์ใช้จะต้องมีส่วนของฐานข้อมูลที่เก็บข้อมูลของเจ้าของบัตรไว้ด้วย ซึ่งโปรแกรมที่ใช้ก็จะแสดงรายละเอียดดังรูปที่ 4.6



รูปที่ 4.6 โปรแกรมแสดงผลบนจอคอมพิวเตอร์

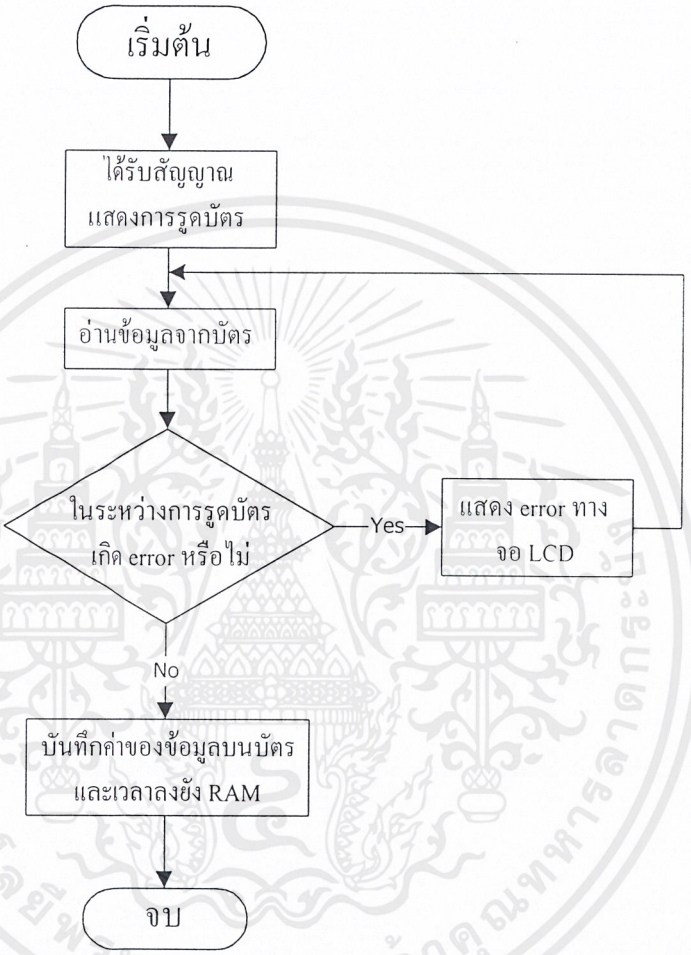
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.7 แผนผังแสดงการทำงาน



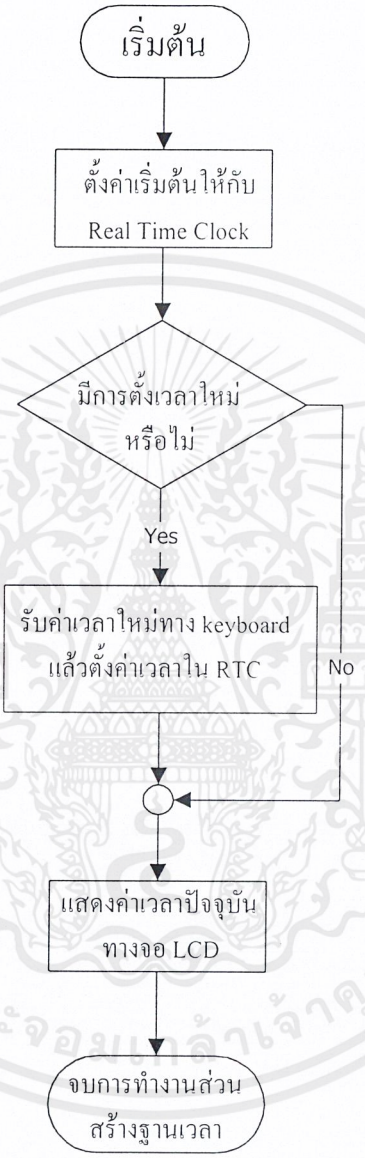
รูปที่ 4.7 แผนผังแสดงการทำงานของเครื่องบันทึกข้อมูลบัตรแม่เหล็ก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



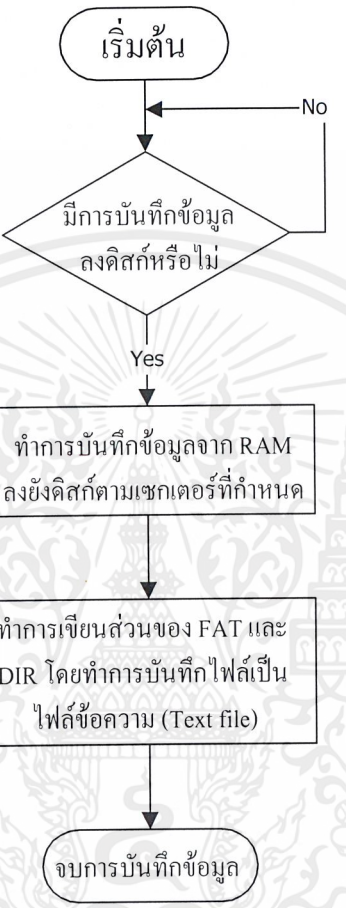
รูปที่ 4.8 แผนผังแสดงการเก็บข้อมูลลงยังหน่วยความจำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



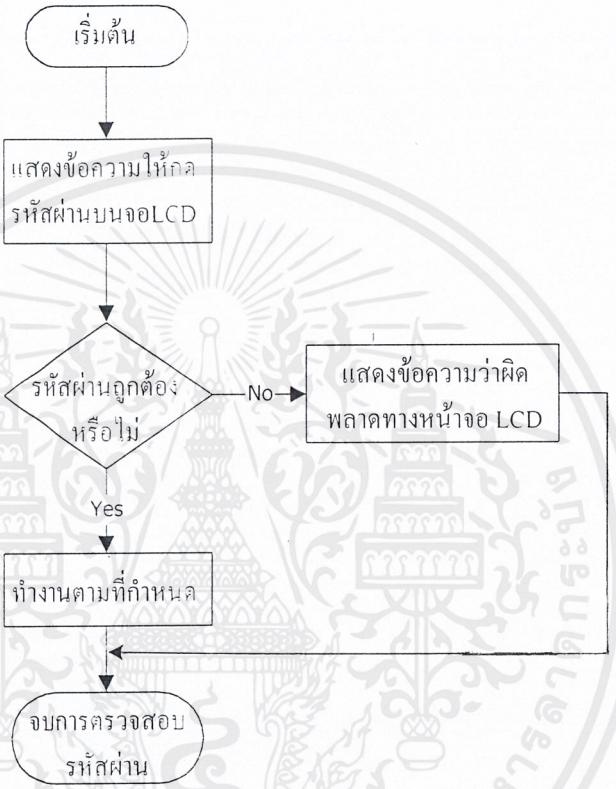
รูปที่ 4.9 แผนผังแสดงการควบคุมการทำงานของส่วนฐานเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.10 แผนผังแสดงการบันทึกข้อมูลลงดิสก์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 แผนผังแสดงการตรวจสอบรหัสผ่าน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

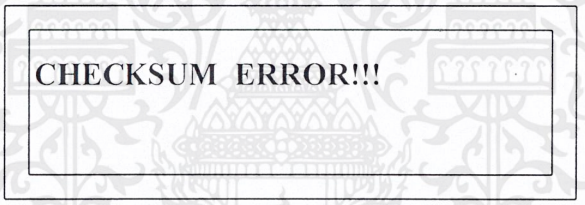
การทดสอบการทำงาน

5.1 การทดสอบอ่านข้อมูลจากบัตรแม่เหล็ก

เมื่อทำการรูดบัตรผ่านหัวอ่านบัตรแม่เหล็ก ถ้าข้อมูลที่อ่านมีพาริตี้ของบิตผิดพลาด บนจอ LCD จะขึ้นข้อความ PARITY ERROR!!!



ถ้าบล็อก(Block)ข้อมูลมีความผิดพลาดจะขึ้นข้อความ CHECKSUM ERROR!!!



ถ้าข้อมูลถูกต้องไม่มีความผิดพลาดทั้งในพาริตี้บิต และในบล็อกของข้อมูล ก็จะปรากฏข้อมูลเป็นตัวเลขขึ้นมา จากการทดลองรูดบัตรแม่เหล็กซ้ำใบเดิมหลายๆครั้ง ปรากฏข้อมูลตัวเลขที่ได้ดังตารางที่ 5.1 ซึ่งจะเห็นว่าข้อมูลตัวเลขบนบัตรใบเดิมที่อ่านได้เหมือนกันทุกครั้ง

ตารางที่ 5.1 ผลการทดสอบความถูกต้องของการอ่านข้อมูลบนบัตรแม่เหล็กหลายๆ ครั้ง

บัตรใบที่	ครั้งที่รูด	ข้อมูลตัวเลขที่อ่านได้
1	1	B0014820037359994D1299D211190088D10000F1
	2	B0014820037359994D1299D211190088D10000F1
	3	B0014820037359994D1299D211190088D10000F1
2	1	B0000021672321831720D12990003613F2
	2	B0000021672321831720D12990003613F2
	3	B0000021672321831720D12990003613F2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

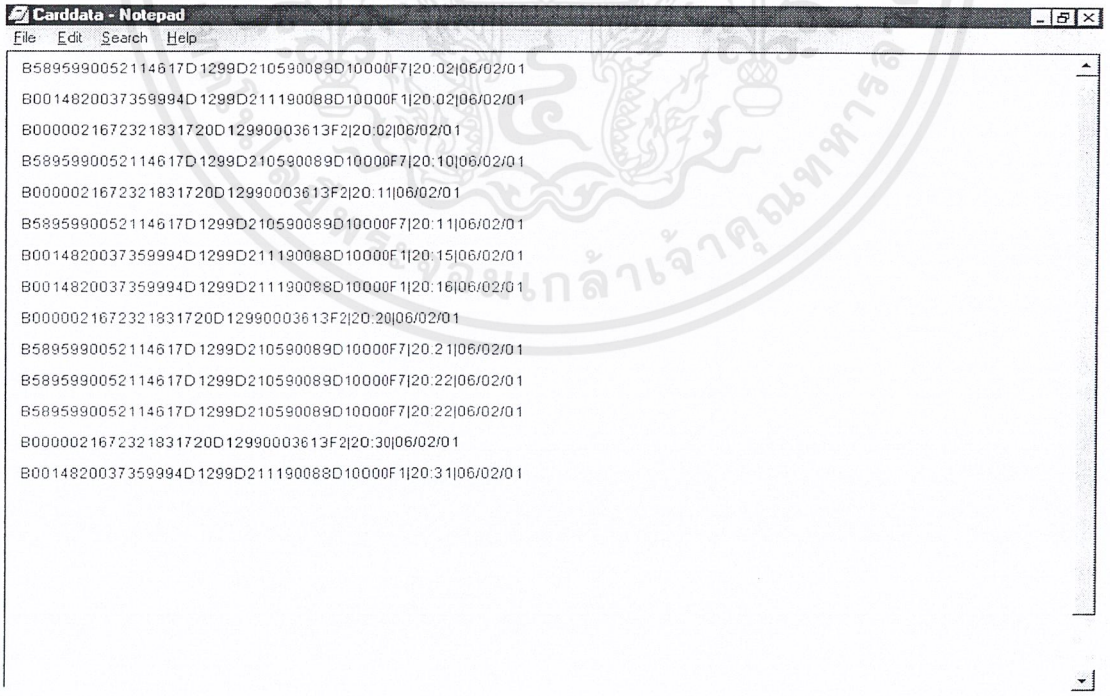
3	1	B5895990052114617D1299D210590089D10000F7
	2	B5895990052114617D1299D210590089D10000F7
	3	B5895990052114617D1299D210590089D10000F7

5.2 การทดสอบการเขียนข้อมูลจากบัตรแม่เหล็กลงดิสก์

โดยได้ทดสอบการรูดบัตรที่เวลาต่าง ๆ และทำการบันทึกเป็นไฟล์ข้อความลงบนดิสก์โดยที่บัตรที่ใช้มี 3 ใบ โดยที่มีข้อมูลสำหรับแต่ละใบที่เก็บเอาไว้ในฐานข้อมูล คือ

Card_Code	Name	StudentID	Major
B0014820037359994D1299D211190088D10000F1	น.ส.มานี รักกสิกร	40010138	วิศวกรรมคอมพิวเตอร์
B0000021672321831720D12990003613F2	นายปิติ นาอุดม	40010146	วิศวกรรมอิเล็กทรอนิกส์
B5895990052114617D1299D210590089D10000F7	น.ส.ชูใจ เชื้อชาติไทย	40010719	วิศวกรรมโยธา

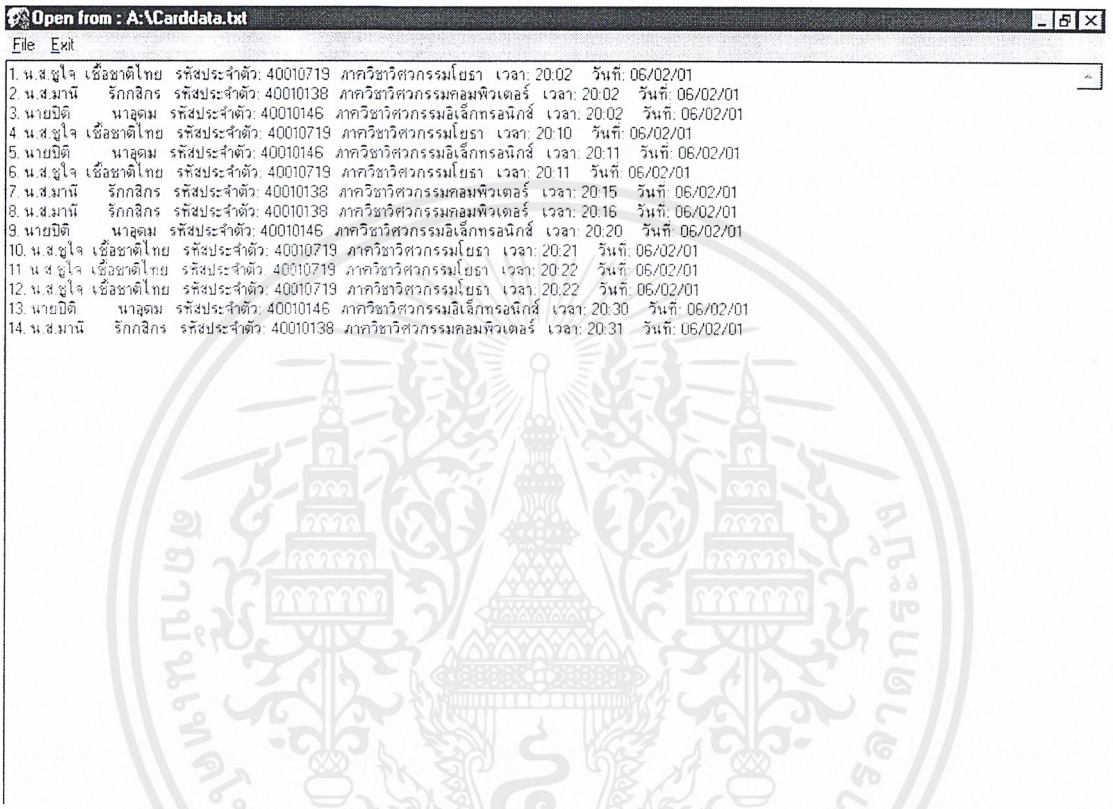
เมื่อทำการทดสอบนั้นก็ได้ทำการรูดบัตรที่เวลาต่าง ๆ หลาย ๆ ครั้งแล้วทำการบันทึกข้อมูลที่เก็บในหน่วยความจำลงไปยังแผ่นดิสก์ ซึ่งเก็บในรูปของไฟล์ข้อความ ซึ่งแสดงได้ดังรูป



รูปที่ 5.1 แสดงข้อมูลจากการรูดบัตรที่บันทึกลงยังแผ่นดิสก์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และเมื่อนำเอาไฟล์ที่ได้ไปเปิดด้วยโปรแกรมแสดงข้อมูลของบัตรที่ทำการรูดบนเครื่องคอมพิวเตอร์ที่เก็บฐานข้อมูลของบัตรแม่เหล็กไว้ จะได้ผลดังรูป



รูปที่ 5.2 แสดงข้อมูลของเจ้าของบัตรแม่เหล็ก และวันเวลาที่ทำการรูดบัตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6
บทสรุป

6.1 สรุปและวิจารณ์

จากการทดสอบการรูดบัตรเพื่อดูข้อมูลบนบัตร โดยเขียนโปรแกรมให้แสดงผลบนจอแสดงผล ในการรูดบัตรที่ข้อมูลถูกต้องและแสดงว่าผ่านนั้น ข้อมูลที่ได้มีความถูกต้องและเหมือนเดิมทุกครั้ง และนำเอาผลที่ได้นี้เก็บเป็นข้อมูลในฐานข้อมูลของบัตรใบนั้น

และในการทำงาน เมื่อมีการรูดบัตร ก็จะมีการนำเอาข้อมูลบัตรที่รูดผ่านไปเก็บลงหน่วยความจำ และเมื่อทำการบันทึกลงดิสก์ ก็จะนำข้อมูลจากหน่วยความจำเก็บลงไปยังแผ่นดิสก์ และเมื่อนำเอาไปเปิดที่เครื่องคอมพิวเตอร์ ไฟล์จากการบันทึกที่ได้ซึ่งชื่อว่า Carddata.txt นั้นพบว่าสามารถเปิดใช้งานได้และแสดงข้อมูลที่ได้ทำการบันทึกด้วยโปรแกรม Notepad แสดงว่า ในการสร้างไฟล์บนแผ่นดิสก์ และการเขียนข้อมูลนั้นเป็นไปอย่างถูกต้องและเป็นไฟล์ที่สมบูรณ์ คือ มีส่วนตารางจัดการไฟล์ ส่วนของไดเรคทอรี และส่วนของข้อมูล ดังนั้น เมื่อนำเอาไฟล์ที่ได้ไปเปิดด้วยโปรแกรมบนคอมพิวเตอร์ที่เขียนขึ้นเพื่อเชื่อมโยงข้อมูลที่บันทึกกับข้อมูลในฐานข้อมูล เพื่อทำการแสดงข้อมูลของเจ้าของบัตร และวันเวลาที่ทำการรูดบัตรได้ แต่ถ้าบัตรใบที่ไม่มีข้อมูลในฐานข้อมูลลงไป โปรแกรมจะไม่สามารถแสดงข้อมูลของบัตรใบนั้นและเวลาได้

ในส่วนของ RAM ซึ่งใช้เบอร์ 62256 ซึ่งสามารถเก็บข้อมูลได้ทั้งหมด 32 กิโลไบต์ แต่ส่วนของข้อมูลที่ทำการบันทึกที่มากที่สุดหนึ่งข้อมูลมีความยาวของส่วนข้อมูลจากบัตร วันเวลาและเครื่องหมายต่าง ๆ ยาวที่สุดคือ 57 ไบต์ ดังนั้นจะสามารถเก็บข้อมูลได้มากที่สุดคือ

$32K / 57 = 574.877 \cong 574$ ครั้ง

ดังนั้นจะสามารถเก็บข้อมูลที่อ่านได้จากบัตรแม่เหล็กลงหน่วยความจำก่อนจะบันทึกลงดิสก์ได้มากที่สุดคือ 574 ข้อมูล

6.2 ปัญหาและอุปสรรคในการทำงาน

1. เนื่องจากยังไม่เคยใช้งานอุปกรณ์หลายอย่าง ในการใช้งานอุปกรณ์แต่ละตัวจึงเสียเวลาในการศึกษาการใช้งานค่อนข้างนานและยังไม่ค่อยเข้าใจวิธีการใช้งานอย่างแท้จริง จึงเกิดปัญหาและข้อผิดพลาดในรายละเอียดของแต่ละส่วน ทำให้การทำงานโดยรวมค่อนข้างช้า
2. ในบางครั้งอาจเกิดข้อผิดพลาดในส่วนที่ทำการบันทึกข้อมูล เช่น ข้อมูลที่ทำการบันทึกนั้นมีการเลื่อนออกไปจากเดิม หรือทำการบันทึกข้อมูลไม่ครบ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3 แนวทางการแก้ไขปรับปรุง

สำหรับแนวทางการแก้ไขปรับปรุงสำหรับอุปกรณ์นี้ก็คือ

1. ทำการศึกษาการทำงานและการใช้งานของอุปกรณ์อย่างละเอียดให้เข้าใจก่อนใช้งาน และหาตัวอย่างการใช้งานที่ถูกต้องมาทำการทดสอบดูก่อนเพื่อการใช้งานที่ถูกต้อง ซึ่งจะทำให้การทำงานจริงนั้นเกิดข้อผิดพลาดน้อยลง และการดำเนินงานเร็วขึ้น ซึ่งจะทำให้มีโอกาสปรับปรุงส่วนอื่น ๆ แทนที่จะต้องมาเสียเวลากับข้อผิดพลาดเพียงบางจุดนาน ๆ

2. สำหรับความผิดพลาดในบางครั้งในระหว่างการทำงานของเครื่องบันทึกข้อมูลบัตรแม่เหล็กนี้ อาจแก้ไขด้วยการปรับปรุงโปรแกรมการทำงานให้ดีขึ้น พยายามลดความผิดพลาดอันอาจจะเกิดขึ้นได้โดยไม่คาดคิด และในการต่อวงจรก็ควรต่ออย่างระมัดระวังให้มั่นใจว่าถูกต้อง ไม่เกิดข้อผิดพลาดขึ้นได้

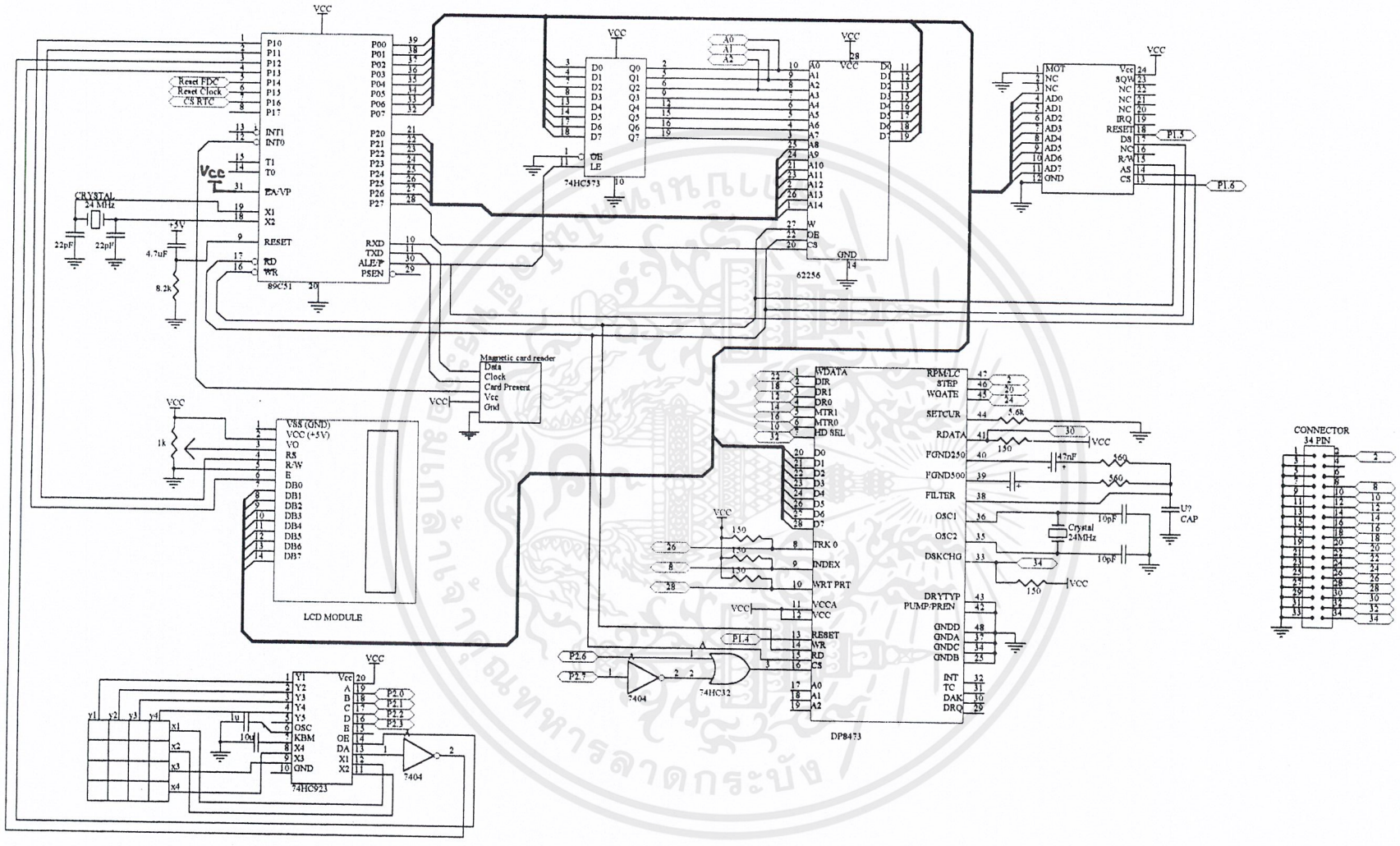
3. อาจพัฒนาขีดความสามารถของเครื่องมือ ทำการเพิ่มเมนูฟังก์ชันการทำงานให้สามารถใช้งานได้อย่างมีประสิทธิภาพและใช้ง่ายขึ้น เช่น ทำให้สามารถทำการเปลี่ยนค่ารหัสผ่านได้ การติดตั้งแบตเตอรีเพื่อทำการสำรองไฟเพื่อไม่ให้ข้อมูลสูญหายในกรณีที่และเลือกใช้อุปกรณ์ที่มีการทำงานที่ถูกต้องเที่ยงตรงสูง ซึ่งจะทำให้ข้อมูลที่ได้มีความเชื่อถือได้

4. เพิ่มเติมส่วนที่จะสามารถนำเอาข้อมูลที่ได้ไปใช้งานจริง เป็นการประยุกต์การใช้งานข้อมูลที่ได้ให้เกิดประโยชน์ตามลักษณะรูปแบบของจุดประสงค์ต่าง ๆ เช่น การนำเอาข้อมูลที่ได้ไปตัดสินว่าการรูดบัตรอยู่ในช่วงเวลาที่กำหนดหรือไม่ ซึ่งอาจนำเอาไปคำนวณเวลาการทำงานที่แท้จริงหรือดูเวลาการเข้าทำงานในหน่วยงานต่าง ๆ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปแสดงวงจรของเครื่องบันทึกข้อมูลบัตรแม่เหล็ก

วิธีการใช้งานเครื่องบันทึกข้อมูลบัตรแม่เหล็ก

1. เสียบปลั๊กเพื่อจ่ายไฟให้กับเครื่อง กดปุ่มใด ๆ บนคีย์บอร์ดก็ได้เพื่อเริ่มการทำงาน
2. การรูดบัตร
   - รูดบัตรไปในทิศทางที่ถูกต้องตามที่กำหนดที่หัวอ่านบัตร แล้วถ้าจอแสดงผล LCD แสดงข้อความว่า "PASS" แสดงว่าการรูดบัตรครั้งนั้นถูกต้อง
   - ถ้าทำการรูดบัตรแล้วเกิดข้อผิดพลาด จะแสดงข้อความว่า "PARITY ERROR!!!" หรือ "CHECKSUM ERROR!!!" ให้ทำการรูดบัตรใหม่ให้ถูกต้องจนกว่าจะแสดงข้อความว่า "PASS"
3. การตั้งค่าเวลาใหม่
   - เมื่อต้องการตั้งค่าเวลาใหม่ ให้กดคีย์บอร์ดปุ่ม 2nd ซึ่งจะปรากฏเมนูว่า "SET TIME" ให้กดปุ่ม ENTER เพื่อตอบตกลง แล้วจอ LCD จะแจ้งให้ผู้ใช้ป้อนค่ารหัสผ่าน ให้ทำการกดรหัสผ่านเป็นเลข 4 หลักแล้วกด ENTER แล้วจึงป้อนเวลา กด ENTER ป้อนวันที่ แล้วกด ENTER อีกครั้ง จะเป็นการตั้งค่าของเวลาที่ถูกต้องใหม่
   - ถ้าหากว่ารหัสผ่านไม่ถูกต้อง จะแสดงข้อความว่า "WRONG!!" และออกจากเมนู
4. การบันทึกข้อมูลลงยังดิสก์
   - กดคีย์บอร์ดที่ปุ่ม 2nd แล้วกดปุ่ม ↓ จะปรากฏเมนูว่า "SAVE DATA" กดปุ่ม ENTER แล้วป้อนรหัสผ่านที่ถูกต้องเช่นเดียวกับการตั้งเวลา ขณะที่กำลังบันทึกข้อมูลจะแสดงข้อความว่า "SAVING" ซึ่งในช่วงนี้จะไม่สามารถรูดบัตรได้ เนื่องจากจะทำให้การบันทึกข้อมูลมีความผิดพลาดได้ รอจนกว่าการบันทึกข้อมูลจะเสร็จสมบูรณ์ ซึ่งจะปรากฏข้อความว่า "COMPLETE" และแสดงวันและเวลาขณะนั้นตามปกติ
5. การลบข้อมูลที่ทำการบันทึกแล้ว
   - เมื่อทำการบันทึกข้อมูลลงยังดิสก์แล้ว ให้ทำการลบข้อมูลเดิมออกจากหน่วยความจำ ทำได้โดยการกดปุ่ม 2nd แล้วกดปุ่ม ↓ ทั้งหมด 2 ครั้ง จะปรากฏเมนูว่า "CLEAR DATA" ให้กดปุ่ม ENTER แล้วทำการป้อนรหัสผ่าน ถ้าถูกต้อง จะแสดงข้อความว่า "COMPLETE" เพื่อแสดงว่าการลบข้อมูลเสร็จเรียบร้อยแล้ว สามารถรูดบัตรในครั้งใหม่ได้ โดยถ้าทำการบันทึกข้อมูลอีก ก็จะไม่มีข้อมูลเก่าที่ได้ทำการลบไปแล้ว
6. การเรียกดูข้อมูลของผู้ถือบัตรและวันเวลาที่ทำการรูดบัตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ให้นำเอาแผ่นดิสก์ที่ได้ทำการบันทึกข้อมูลแล้ว ไปเปิดด้วยโปรแกรมแสดงข้อมูลบนเครื่องคอมพิวเตอร์ที่มีการเชื่อมต่อกับฐานข้อมูลที่มีข้อมูลของเจ้าของบัตรบันทึกไว้ ซึ่งโปรแกรมจะแสดงชื่อ รายละเอียดของผู้ถือบัตร และวันเวลาที่ทำการรูดบัตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมภาษา Visual Basic เชื่อมต่อกับฐานข้อมูลและแสดงข้อมูลทางหน้าจอคอมพิวเตอร์

```
Option Explicit
Dim FileN As String
Private Sub mnExit_Click()
   End
End Sub

Private Sub mnOpenFile_Click()
   Dim FileN As String, Reci As String, st As String
   Dim c As Integer, ru As Integer, n As Integer
   Dim Buff() As String
   Dim Buff2(2) As String
   Dim dat As Variant
   CommonDialog1.DialogTitle = " Load TEXT File"
   CommonDialog1.Filter = "TextForm Files (*.txt)|*.txt|(*.*)|*.*"
   CommonDialog1.FilterIndex = 1
   CommonDialog1.ShowOpen
   Form1.Caption = "Open from : " & CommonDialog1.FileName
   FileN = CommonDialog1.FileName
   If Len(FileN) = 0 Then Exit Sub
   c = 1
   ru = 1
   Open FileN For Input As #2
   Do While Not EOF(2)
      Line Input #2, Reci
'*****************************************
      If Len(Reci) <> 0 Then
         st = Reci
         Buff() = Split(Reci, "|", , vbTextCompare)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        If UBound(Buff()) <> -1 Then
          Data1.Recordset.MoveFirst
          Do While Not Data1.Recordset.EOF
            If Data1.Recordset.Fields("Card_code") = Buff(0) Then
              Buff(0) = Data1.Recordset.Fields("Name")
              Buff2(1) = Data1.Recordset.Fields("StudentID")
              Buff2(2) = Data1.Recordset.Fields("Major")
              dat = dat & c & ". " & Buff(0) & "  รหัสประจำตัว: " & Buff2(1) & _
                  " ภาควิชา" & Buff2(2) & "  เวลา: " & Buff(1) & _
                  "    วันที่: " & Buff(2) & vbCrLf
              c = c + 1
              Text1.Text = dat
              Exit Do
            End If
            Data1.Recordset.MoveNext
            ru = ru + 1
          Loop
        Else
          Text1.Text = "Unknow"
        End If
      Else
        Text1.Text = Buff(n)
      End If
    Loop
    Close #2
  Exit Sub

  ErrCom1:
    Select Case Err.Number
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        Case 9
            Resume Next
        Case 53
            Resume Next
        Case Else
            Text1.Text = Buff(n) & " " & Err.Description
    End Select
End Sub
Private Sub Text1_Change()
    Data1.DatabaseName = App.Path & "\now.mdb"
    Data1.Refresh
    Data1.RecordSource = "dbtable"
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

National Semiconductor

July 1990

DP8473 Floppy Disk Controller PLUS-2™

General Description

This controller is a full featured floppy disk controller that is software compatible with the μPD765A, but also includes many additional hardware and software enhancements. These enhancements include additional logic specifically required for an IBM® PC, PC-XT®, PC-AT®, or PS/2® design.

This controller incorporates a precision analog data separator, that includes a self trimming delay line and VCO. Up to three external filters are switched automatically depending on the data rate selected. This provides optimal performance at the standard PC data rates of 250/300 kb/s, and 500 kb/s. It also enables optimum performance at 1 Mb/s (MFM). These features combine to provide the lowest possible PLL bandwidth, with the greatest lock range, and hence the widest window margin.

This controller includes write precompensation circuitry. A shift register is used to provide a fixed 125 ns early-late precompensation for all tracks at 500k/300k/250 kb/s (83 ns for 1 MB/s), or a precompensation value that scales with

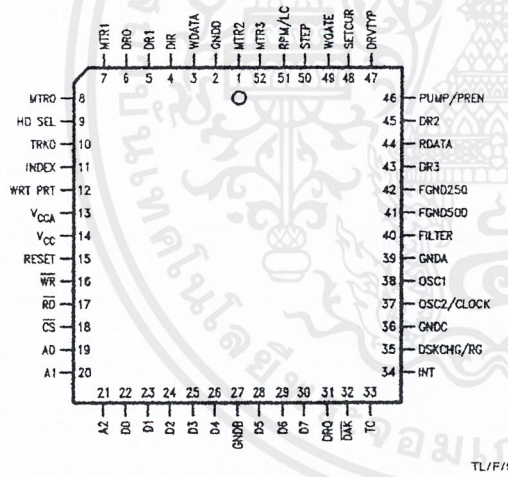
(Continued)

Features

- Fully μPD765A and IBM-BIOS compatible
- Integrates all PCXT®, PCAT®, and most PS/2® Logic
  - On chip 24 MHz Crystal Oscillator
  - DMA enable logic
  - IBM compatible address decode of A0–A2
  - 12 mA μP bus interface buffers
  - 48 mA floppy drive interface buffers
  - Data rate and drive control registers
- Precision analog data separator
  - Self-calibrating PLL and delay line
  - Automatically chooses one of three filters
  - Intelligent read algorithm
- Two pin programmable precompensation modes
- Other enhancements
  - up to 1 Mb/s data rate
  - Implied seek up to 4000 tracks
  - IBM or ISO formatting
- Low power CMOS, with power down mode

Connection Diagrams

Plastic Leaded Chip Carrier

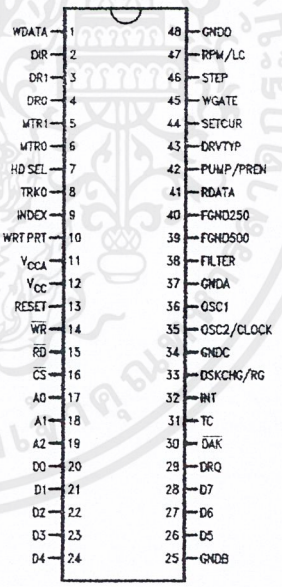


TL/F/9384–1

Top View

Order Number DP8473V
See NS Package Number V52A

Dual-In-Line Package



TL/F/9384–2

Top View

Order Number DP8473N
See NS Package Number N48A

TRI-STATE® is a registered trademark of National Semiconductor Corporation.
PLUS-2™ is a trademark of National Semiconductor Corporation.
IBM®, PCXT®, PCAT® and PS/2® are registered trademarks of International Business Machines Corporation.

©1995 National Semiconductor Corporation TL/F/9384

RRD-B30M105/Printed in U. S. A.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

General Description (Continued)

the data rate, 83 ns/125 ns/208 ns/ 250 ns for data rates of 1.0M/500k/300k/250 kb/s respectively.

Specifically to support the PC-AT and PC-XT design, the Floppy Disk Controller PLUS-2 includes address decode for the A0–A2 address lines, the motor/drive select register, data rate register for selecting 250/300/500 kb/s, Disk Changed status, dual speed spindle motor control, low write current and DMA/interrupt sharing logic. The controller also supports direct connection to the µP bus via internal 12 mA buffers. The controller also can be connected directly to the disk drive via internal open drain high drive outputs, and Schmitt inputs.

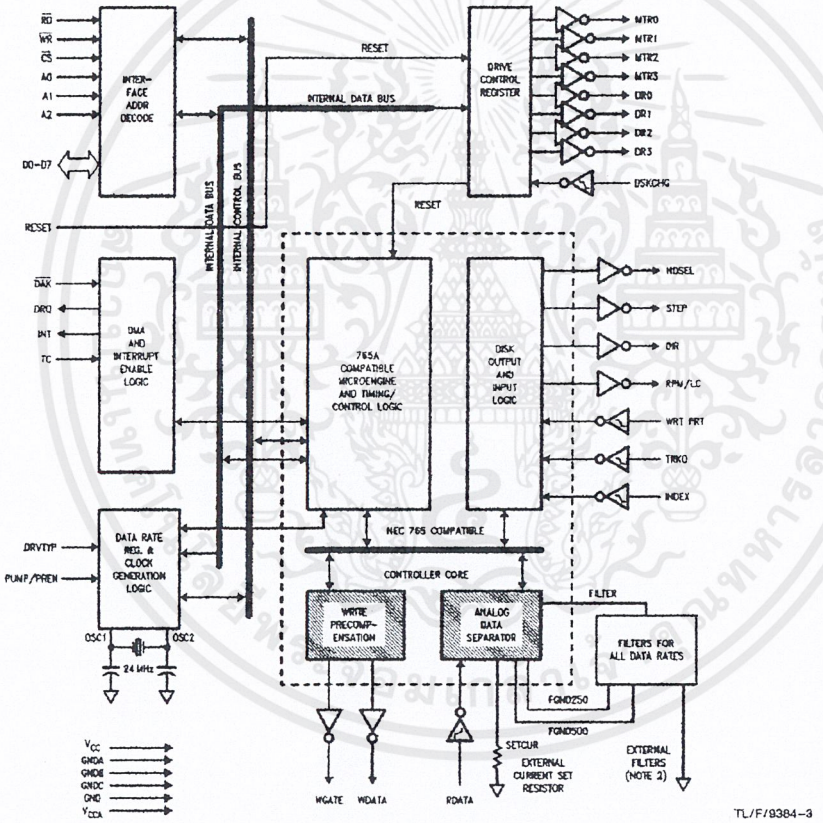
In addition to this logic the DP8473 includes many features to ease design of higher performance drives and future controller upgrades. These include 1.0 Mb/s data rate, extended track range to 4096, implied seeking, working Scan Commands, motor control timing, both standard IBM formats as well as Sony 3.5" (ISO) formats, and other enhancements.

This device is available in a 52 pin Plastic Chip Carrier, and in a 48 pin Dual-In-Line package.

Table of Contents

General Description
Pin Description
Functional Description
Register Description
Result Phase Register Description
Processor Software Interface
Command Description Table
Command Descriptions
DC and AC Characteristics

Block Diagram



Note 1: The MTR2, MTR3, DR2, and DR3 are not available on the 48 pin DIP (DP8473N, J) versions.

Note 2: See *Figure 4* for filter description.

Note 3: Total transistor count is 29,700 (approx).

FIGURE 1. DP8473 Functional Block Diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า - ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Descriptions

Symbol	DP8473 PCC	DP8473 DIP	Function
A0, A1, A2	19–21	17–19	Address lines from the microprocessor. This determines which registers the microprocessor is accessing as shown in Table IV in the Register Description Section. Don't care during DMA transfers.
$\overline{CS}$	18	16	Active low input to enable the $\overline{RD}$ and $\overline{WR}$ inputs. Not required during DMA transfers. This should be held high during DMA transfers.
$\overline{DAK}$	32	30	Active low input to acknowledge the DMA request and enable the $\overline{RD}$ and $\overline{WR}$ inputs. This signal is enabled when D3 of the Drive Control Register is set.
DIR	4	2	This output determines the direction of the head movement (low = step in, high = step out). When in the write or read modes, this output will be high. This is a high drive open drain output.
DRQ	31	29	Active high output to signal the DMA controller that a data transfer is needed. This signal is enabled when D3 of the Drive Control Register is set.
DRV TYP	47	43	This is an input used by the controller to enable the 300 kb/s mode. This enables the use of floppy drives with either dual or single speed spindle motors. For dual speed spindle motors, this pin is tied low. When low, and 300 kb/s data rate is selected in the data rate register, the PLL actually uses 250 kb/s. This pin is tied high for single speed spindle motor drives (standard AT drive). When this pin is high and 300 kb/s is selected 300 kb/s is used. (See also RPM/LC pin).
DR0	6	4	This is an active low drive select line for drive 0 that is controlled by the Drive Control register bits D0, D1. The Drive Select bit is ANDed with the Motor Enable of the same number. This is a high drive open drain output.
DR1	5	3	This is an active low drive select similar to DR0 line except for drive 1.
DR2	45	—	This is the same as DR0 except for drive 2.
DR3	43	—	This is the same as DR0 except for drive 3.
DSKCHG/RG	35	33	This latched Schmitt input signal is inverted and routed to D7 of the data bus and is read when address xx7H is enabled. DSKCHG is a disk drive output that indicates when the drive door has been opened. When the RG bit in the Mode Command is set, this pin functions as a Read Gate signal that when low forces the data separator to lock to the crystal, and when high it locks to data for diagnostic purposes.
D0–D4	22-26	20–24	Bi-directional data lines to the microprocessor. These are the lower 5 bits and have buffered 12 mA outputs.
D5–D7	28–30	26–28	Bi-directional data lines to the microprocessor. These upper 3 bits have buffered 12 mA outputs.
FGND250	42	40	This pin connects the PLL filter for 250k(MFM)/125k(FM) b/s or 300k(MFM)/150k(FM) b/s to ground. This is a low impedance open drain output.
FGND500	41	39	This pin connects the PLL filter for 500k(MFM)/250k(FM) b/s to ground. This is a low impedance open drain output.
FILTER	40	38	This pin is the output of the charge pump and the input to the VCO. One or more filters are attached between this pin and the GNDA, FGND250 and FGND500 pins.
GNDA	39	37	This pin is the analog ground for the data separator, including all primary and secondary PLLs and delay lines.
GNDB	27	25	This pin is the digital ground for the 12 mA microprocessor interface buffers. This includes D0–D7, INT, and DRQ.
GNDC	36	34	This pin is the digital ground for the controller's digital logic, including all internal registers, micro-engine, etc.
GNDD	2	48	This pin is the digital ground for the disk interface output drivers.
HD SEL	9	7	This output determines which disk drive head is active. Low = Head 1, Open (high) = Head 0. This is a high drive open drain output.
INDEX	11	9	This active low Schmitt input signals the beginning of a track.
INT	34	32	Active high output to signal that an operation requires the attention of the microprocessor. The action required depends on the current function of the controller. This signal is enabled when D3 of the Drive Control Register is set.

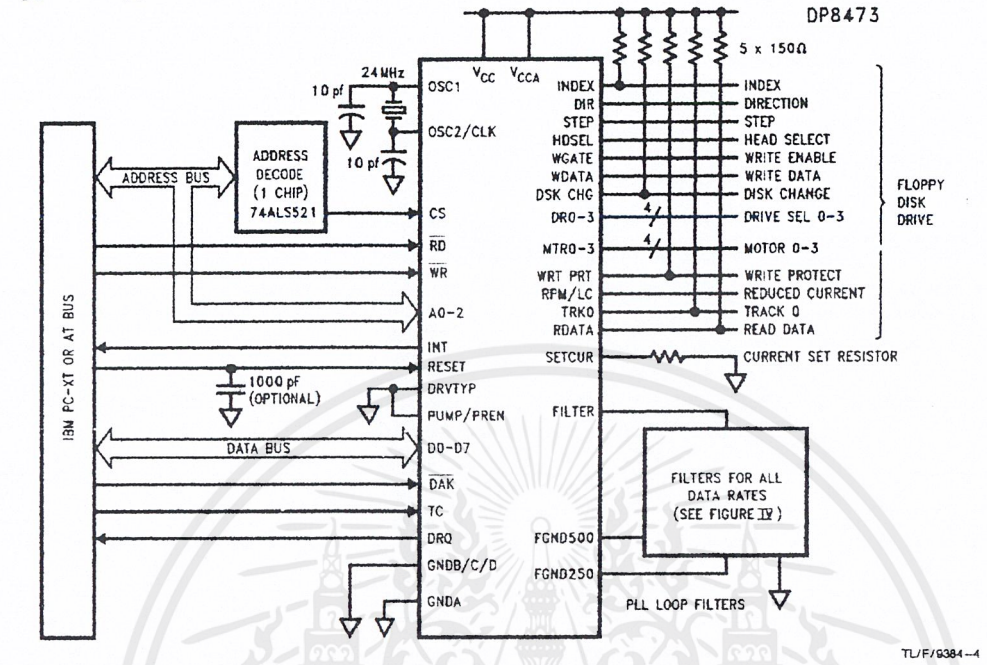
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Descriptions (Continued)

Symbol	DP8473 PCC	DP8473 DIP	Function
MTR0	8	6	This is an active low motor enable line for drive 0, which is controlled by the Drive Control register. This is a high drive open drain output.
MTR1	7	5	This is an active low motor enable line for drive 1. Similar to MTR0.
MTR2	1	—	This is an active low motor enable line for drive 2. Similar to MTR0.
MTR3	52	—	This is an active low motor enable line for drive 3. Similar to MTR0.
OSC1	38	36	One side of an external 24 MHz crystal is attached here. This pin is tied low if an external clock is used.
OSC2/CLOCK	37	35	One side of the external 24 MHz crystal is attached here. If a crystal is not used, a TTL or CMOS compatible clock is connected to this pin.
PUMP/PREN	46	42	When the PU bit is set in Mode Command this pin is an output that indicates when the charge pump is making a correction. Otherwise this pin is an input that sets the precomp mode as shown in Table VI. If pin is configured as PUMP, PREN is assumed high.
$\overline{RD}$	17	15	Active low input to signal a read from the controller to the microprocessor .
RDATA	44	41	The active low raw data read from the disk is connected here. This is a Schmitt input.
RESET	15	13	Active high input that resets the controller to the idle state, and resets all the output lines to the disk drive to their disabled state. The Drive Control register is reset to 00. The Data Rate register is set to 250 kb/s. The Specify command registers are not affected. The Mode Command registers are set to the default values. Reset should be held active during power up. To prevent glitches activating the reset sequence, a small capacitor (1000 pF) should be attached to this pin.
RPM/LC	51	47	This high drive open drain output pin has two functions based on the selection of the DRVTYP pin. 1. When using a dual speed spindle motor floppy drive (DRVTYP pin low), this output is used to select the spindle motor speed, either 300 RPM or 360 RPM. In this mode this output goes low when 250/300 kb/data rate is chosen in the data rate register, and high when 500 kb/s is chosen. 2. When using a single speed spindle motor floppy drive (DRVTYP pin high), this pin indicates when to reduce the write current to the drive. This output is high for high density media (when 500 kb/s is chosen).
SETCUR	48	44	An external resistor connected from this pin to analog ground programs the amount of charge pump current that drives the external filters. The PLL Filter Design section shows how to determine the values.
STEP	50	46	This active low open drain high drive output will produce a pulse at a software programmable rate to move the head during a seek operation.
TC	33	31	Active high input to indicate the termination of a DMA transfer. This signal is enabled when the DMA Acknowledge pin is active.
TRK0	10	8	This active low Schmitt input tells the controller that the head is at track zero of the selected disk drive.
WDATA	3	1	This is the active low open drain write precompensated serial data to be written onto the selected disk drive. This is a high drive open drain output.
WGATE	49	45	This active low open drain high drive output enables the write circuitry of the selected disk drive. This output has been designed to prevent glitches during power up and power down. This prevents writing to the disk when power is cycled.
$\overline{WR}$	16	14	Active low input to signal a write from the microprocessor to the controller.
WRT PRT	12	10	This active low Schmitt input indicates that the disk is write protected. Any command that writes to that disk drive is inhibited when a disk is write protected.
V_{CC}	14	12	This pin is the 5V supply for the digital circuitry.
V_{CCA}	13	11	This pin is the 5V supply for the analog data separator circuitry.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Typical Application



Recommended Plastic Chip Carrier Socket:
AMP P/N 821551-1 or equivalent.

FIGURE 2. DP8473 Typical Application

Functional Description

This section describes the basic architectural features of the DP8473, and many of the enhancements provided. Refer to *Figure 1.*

765A COMPATIBLE MICRO-ENGINE

The core of the DP8473 is a μPD765A compatible microcoded engine. This engine consists of a sequencer, program ROM, and disk/misc registers. This core is clocked by either a 4 MHz, 4.8 MHz or 8 MHz clock selected in the Data Rate Register. Upon this core is added all the glue logic used to implement a PC-XT or AT, or PS/2 floppy controller, as well as the data separator and write precompensation logic.

The controller consists of a microcoded engine that controls the entire operation of the chip including coordination of data transfer with the CPU, controlling the drive controls, and actually performing the algorithms associated with reading and writing data to/from the disk. This includes the read algorithm for the data separator.

Like the μPD765A, this controller takes commands and returns data and status through the Data Register in a byte serial fashion. Handshake for command/status I/O is provided via the Main Status Register. All of the μPD765A commands are supported, as are many other enhanced commands.

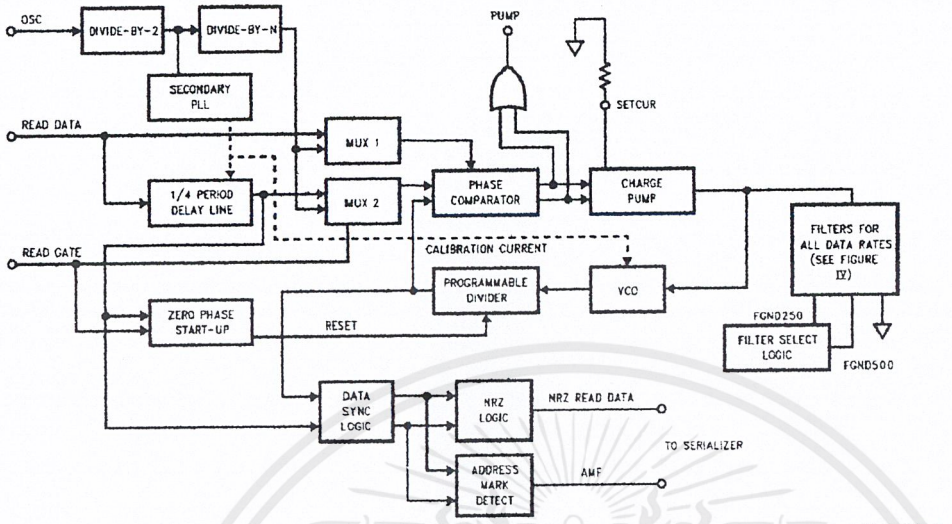
DATA SEPARATOR

The internal data separator consists of an analog PLL and its associated circuitry. The PLL synchronizes the raw data signal read from the disk drive. The synchronized signal is used to separate the encoded clock and data pulses. The data pulses are de-serialized into bytes and then sent to the μP by the controller.

The main PLL consists of four main components, a phase comparator, a filter, a voltage controlled oscillator (VCO), and a programmable divider. The phase comparator detects the difference between the phase of the divider's output and the phase of the raw data being read from the disk. This phase difference is converted to a current which either charges or discharges one of the three external filters. The resulting voltage on the filter changes the frequency of the VCO and the divider output to reduce the phase difference between the input data and the divider's output. The PLL is "locked" when the frequency of the divider is exactly the same as the average frequency of the data read from the disk. A block diagram of the data separator is shown in *Figure 3.*

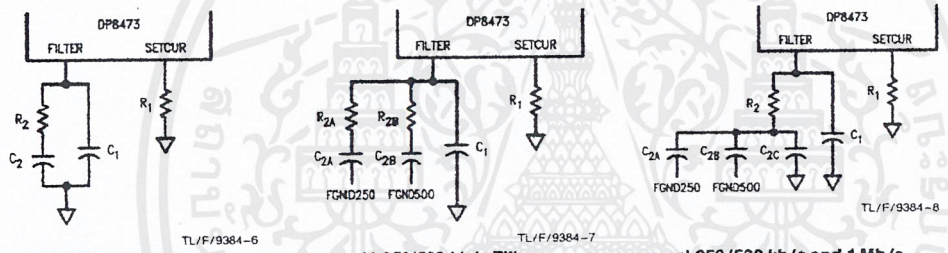
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Functional Description (Continued)



TL/F/9384–5

FIGURE 3. Block Diagram of DP8473's Data Separator



a) Single Data Rate **b) 250/500 kb/s Filter** **c) 250/500 kb/s and 1 Mb/s**

Note: For all filter configurations, 250kb/s and 300 kb/s share the same filter.

FIGURE 4. Typical Configuration for Loop Filters for the DP8473 Showing Component Labels

To ensure optimal performance, the data separator incorporates several additional circuits. The quarter period delay line is used to determine the center of each bit cell. A secondary PLL is used to automatically calibrate the quarter period delay line. The secondary PLL also calibrates the center frequency of the VCO.

To eliminate the logic associated with controlling multiple data rates the DP8473 supports the connection of three filters to the chip via the FGND250, and FGND500 pins (filter ground switches). The controller chooses which filter components to use based on the value loaded in the Data Rate Register. If 500 k(MFM) is being used then the FGND500 is enabled (FGND250 is disabled). If 250 k(MFM) or 300 k(MFM) is being used the FGND250 pin is enabled, and FGND500 is disabled. For 1 Mb/s (MFM) both FGND pins are disabled.

Note for FM encoding: Sometimes, after a reset, the DP8473 will consistently return an error in the Result Phase after an FM read command. If this occurs, simply reset the DP8473 and retry the operation. This may have to be done more than once, as many as five times. Resetting and repeating will prevent soft errors being reported prematurely. This technique is used by MS-DOS.

Figure 4 shows several possible filter configurations. For a filter to cover all data rates (*Figure 4c*), the DP8473 has a 1 Mb/s filter always connected and other capacitor filter components for the other data rates are switched in parallel to this filter. The actual loop filter for 500 kb/s is the parallel combination of the two capacitors, C_{2C} and C_{2B}, attached to the FGND500 pin and to ground. The 250/300 kb/s filter is the parallel combination of the capacitors, C_{2C} and C_{2A}, attached to the FGND250, and ground. If 1 Mb/s need not be supported then the filter configuration of *Figure 4b* can be used. This configuration allows more optimal performance for both 500k and 250/300 kb/s. *Figure 4a* is a simple filter configuration primarily for a single data rate (or multiple data rates with a performance compromise). Table II shows some typical filter values. Other filter configurations and values are possible, these result in good general performance.

While the controller and data separator support both FM and MFM encoding, the filter switch circuitry only supports

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Functional Description (Continued)

the IBM standard MFM data rates. To provide both FM and MFM filters external logic may be necessary.

The controller takes best advantage of the internal data separator by implementing a sophisticated ID search algorithm. This algorithm, shown in *Figure 5*, enhances the PLL's lock characteristics by forcing the PLL to relock to the crystal any time the data separator attempts to lock to a non-preamble pattern. This algorithm ensures that the PLL is not thrown way out of lock by write splices or bad data fields.

TABLE II. Typical Filter Values for the Various Data Rates (Assuming ±6% Capture Range)

Data Rate (MFM b/s)	C_2	R_2	C_1	R_1
Filter Values when Using All 3 Data Rates				
1.0M 500k 250/300k	C_{2C} = 0.012 µF C_{2B} = 0.015 µF C_{2A} = 0.033 µF	560Ω	510 pF	5.6 kΩ
Filter Values when Using 250/300 and 500 kb/s				
500k 250/300k	C_{2B} = 0.027 µF C_{2A} = 0.047 µF	560Ω 560Ω	1000 pF	5.6 kΩ
Filter Using Only One Data Rate				
1.0M 500k 300/250k	C_2 = 0.012 µF C_2 = 0.027 µF C_2 = 0.047 µF	560Ω 560Ω 560Ω	510 pF 1000 pF 2000 pF	5.6 kΩ 5.6 kΩ 5.6 kΩ

(These values are preliminary and thus are subject to change.)

TABLE III. Data Rates (MFM) versus VCO Divide-By Factor

Data Rate	N
1 Mb/s	4
500 kb/s	8
300 kb/s	16
250 kb/s	16

PLL DIAGNOSTIC MODES

In addition, the DP8473 has two diagnostic modes to enable filter optimization, 1) enabling the Charge Pump output signal onto the PUMP/PREN pin, and 2) providing external control of the Read Gate signal to the data separator. Both modes are enabled in the last byte of the Mode Command.

The Pump output signal indicates when the charge pump is making a phase correction, and hence whether the loop is locked or not.

The Read Gate function, when enabled, allows the designer to manually force the data separator to lock to the incoming data or back to the reference clock. This enables easy verification of the lock characteristics of the PLL, by monitoring the FILTER pin, and the Pump signal.

PLL FILTER DESIGN

This section provides information to enable design of the data separator's external filter and charge pump set resistor. This discussion is for a single data rate filter, and can be easily extrapolated to the other filters of *Figure 4*. Table II shows some typical filter component values, but if a custom filter is desired, the following parameters must be considered:

R_1: Charge pump current setting resistor. The current set by this resistor is multiplied by the charge pump gain, K_P which is ~2.5. Thus the charge pump current is:

I_{PUMP} = (2.5) 1.2V/R_1. R_1 should be set to between 3–12 kΩ. This resistor determines the gain of the phase detector, which is $K_D = I_{PUMP}/2\pi$.

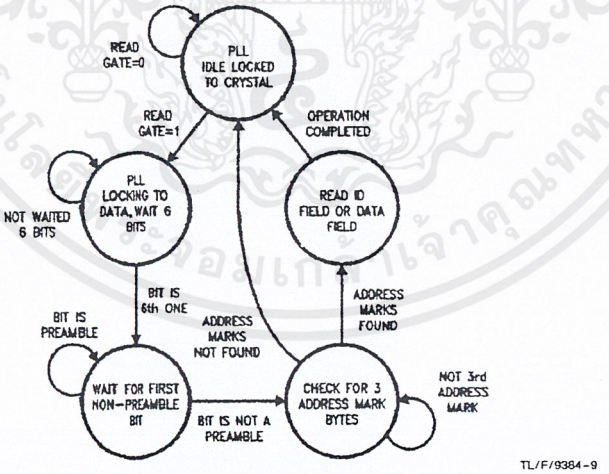


FIGURE 5. Read Algorithm-State Diagram for Data

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Functional Description (Continued)

C_2: Filter capacitor in series with R_2. With pump current this determines loop bandwidth.

R_2: Filter resistor. Determines the PLL damping factor.

C_1: This filter capacitor improves the performance of the PLL by providing additional filtering of bit jitter and noise.

K_{VCO}: The ratio of the change in the frequency of the VCO output due to a voltage change at the VCO input. $K_{VCO} \approx 25$ Mrad/s/V. The VCO is followed by a divider to achieve the desired frequency for each data rate. VCO center frequency is 4 MHz for data rates of 1 Mb/s, 500 kb/s, and 250 kb/s (MFM), and is 4.8 MHz for 300 kb/s (MFM).

K_{PLL}: This is the gain of the internal PLL circuitry, and is the product of $V_{REF} \times K_{VCO} \times K_P$. This value is specified in the Phase Locked Loop Characteristics table.

ω_n: This is the bandwidth of the PLL, and is given by,

$$\omega_n = \sqrt{\frac{K_{PLL}}{2\pi C_2 N R_1}}$$

where N is the number of VCO cycles between two phase comparisons. The value of N for the various data rates are shown in Table III.

ζ: The damping factor is set to 0.7 to 1.2 and is given by,

$$\zeta = \frac{\omega_n R_2 C_2}{2}$$

The trade off, when choosing filter components is between acquisition time while the PLL is locking and jitter immunity while reading data. To select the proper components for a standard floppy disk application the following procedure can be used:

1. Choose FM or MFM, and data rate. Determine N from Table III. Determine preamble length (MFM = 12). The PLL should lock within ½ the preamble time.

2. Determine loop bandwidth (ω_n) required, and set the charge pump resistor R_1.

3. Calculate C_2 using:

$$C_2 = \frac{K_{PLL}}{2\pi R_1 N \omega_n^2}$$

4. Choose R_2 using:

$$R_2 = \frac{2\zeta}{\omega_n C_2}$$

6. Select C_1 to be about $1/20$th of C_2.

The above procedure will yield adequate loop performance. If optimum loop performance is required, or if the nature of the loop performance is very critical, then some additional consideration must be given to choosing ω_n and the damping factor. (For a detailed description on how to choose ω_n and ζ, see: **AN-505 Floppy Disk Data Separator Design Guide for the DP8473**).

WRITE PRECOMPENSATION

The DP8473 incorporates a single fixed 3-bit shift register. This shift register outputs are tapped and multiplexed onto the write data output. The taps are selected by a standard precompensation algorithm. This precompensation value can be selected from the PUMP/PREN pin. When this pin is low 125 ns precomp is used for all data rates except 1 Mb/s which uses 83 ns. When PREN is tied high, the precompensation-value scales with data rate at 250 kb/s its 250 ns, for 300 kb/s its 208 ns, at 500 kb/s its 125 ns, and at 1.0 Mb/s its 83 ns. These values are shown in Table VI.

PC-AT AND PC-XT LOGIC BLOCKS

This section describes the major functional blocks of the PC logic that have been integrated on the controller. Refer back to *Figure 1*, the block diagram.

DMA Enable Logic: This is gating logic that disables the DMA lines and the Interrupt output, under the control of the DMA Enable bit in the Drive control register. When the DMA Enable bit is 0 then the INT, and DRQ are held TRI-STATE, and $\overline{DAK}$ is disabled.

Drive Output Buffers/Input Receivers: The drive interface output pins can drive 150Ω ±10% termination resistors. This enables connection to a standard floppy drive. All drive interface inputs are TTL compatible schmitt trigger inputs with typically 250 mV of hysteresis. *The only functional differences between the 52 pin PLCC and the 48 pin DIP version are that the MTR2 and 3, and DR2 and 3 pins have been removed in order to accommodate the 48 pin package.*

Bus Interface-Address Decode: The address decode circuit allows software access to the controller, Drive Control Register, and Data Rate Register (see Table IV for the memory map) using the same address map as is used in the XT, AT, or PS/2. The decoding is provided for A0–A2, so only a single address decoder connected to the chip select is needed to complete the decode. The bus interface logic includes the 8-bit data bus and DRQ/INT signals. The output drive for these pins is 12 mA.

TABLE IV. Address Memory Map for DP8473

A2	A1	A0	R/W	Register
0	0	0	X	None (Bus TRI-STATE)
0	0	1	X	None (Bus TRI-STATE)
0	1	0	W	Drive Control Register
0	1	1	X	None (Bus TRI-STATE)
1	0	0	R	Main Status Register
1	0	1	R/W	Data Register
1	1	0	X	None (Bus TRI-STATE)
1	1	1	W	Data Rate Register
1	1	1	R	Disk Changed Bit*

*When this location is accessed only bit D7 is driving, all others are held TRI-STATE.

Drive Control Register: This 8-bit write only register controls the drive selects, motor enables, DMA enable, and Reset. See Register Description.

Reset Logic: The reset input pin is active high, and directly feeds the Drive Control Register and the Data Rate Register. After a hardware reset the Drive Control Register is reset to all zeros, and the Data Rate Register is set to 250 kb/s data rate. The controller is held reset until the software sets the Drive Control reset bit, after which the controller may be initialized. A software reset to the controller core can be issued by resetting then setting this bit. A software reset does not reset the Drive Control Register, or the Data Rate Register.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Functional Description (Continued)

Data Rate Register and Clock Logic: This is a two bit register that controls the data rate that the controller uses. See Register Description. This register feeds logic that selects the data rates by programming a prescaler that divides the crystal or clock input by either 3, 5, or 6. This causes either 4 MHz, 4.8 MHz and 8 MHz to be input as the master clock for the controller core. If the Drive Type pin is high and a 300 kb/s data rate is chosen, 4.8 MHz is used to generate 300 kb/s, but when the DRVTYP pin is low and 300 kb/s is selected, 4 MHz is used, and the actual data rate is 250 kb/s. See Table VI.

Low Power Mode Logic: This logic is an enhancement over the standard XT, AT, PS/2 design. In the Low Power Mode the crystal oscillator, controller and all linear circuitry are turned off. When the oscillator is turned off the controller will typically draw about 1 mA. The internal circuitry is disabled while the oscillator is off because the internal circuitry is driven from this clock. The oscillator will turn back on automatically after it detects a read or a write to the Main Status or Data Registers. It may take a few milli-seconds for the oscillator to stabilize and the μP will be prevented from trying to access the Data Register during this time through the normal Main Status Register protocol. (The Request for Master bit in the Main Status Register will be inactive.) There are two ways to go into the low power mode. One is to command the controller to switch to low power immediately. The other method is to set the controller to automatically go into the low power mode 500 ms after the beginning of the idle state (based on a 500 kb/s (MFM) data rate). This would be invisible to the software. The low power mode is programmed through the Mode Command.

The Data Rate Register and the Drive Control Register are unaffected by the power down mode. They will remain active. It is up to the user to ensure that the Motor and Drive select signal are turned off.

TABLE V. Truth Table for Drive Control Register

D7	D6	D5	D4	D1	D0	Function
X	X	X	1	0	0	Drive 0 Selected (DR0 = 0)
X	X	1	X	0	1	Drive 1 Selected (DR1 = 0)
X	1	X	X	1	0	Drive 2 Selected (DR2 = 0)
1	X	X	X	1	1	Drive 3 Selected (DR3 = 0)

Crystal Oscillator: The DP8473 is clocked by a single 24 MHz signal. An on-chip oscillator is provided, to enable the attachment of a crystal, or a clock. If a crystal is used, a 24 MHz fundamental mode, parallel resonant crystal should be used. This crystal should be specified to have less than 40Ω series resistance, and shunt capacitance of less than 7 pF. Low profile and surface mount crystals should be avoided due to their high start-up resistance, which could prohibit the circuit from oscillating.

If an external oscillator circuit is used, it must have a duty cycle of at least 40–60%, and minimum input levels of 2.4V and 0.4V. The controller should be configured so that the clock is input into the OSC2 pin, and OSC1 is tied to ground.

Crystals: NEL Frequency Controls:
NEL-54024-2
NEL-C2800N
SaRonix: SRX 3164

Register Description

This section describes the register bits for all the registers that are directly accessible to the μP. Table IV (previous page) shows the memory map for these registers. Note that in the PC some of the registers are partially decoded, this is not the case here. All registers occupy only their documented addresses.

MAIN STATUS REGISTER (Read Only)

The read only Main Status Register indicates the current status of the disk controller. The Main Status Register is always available to be read. One of its functions is to control the flow of data to and from the Data Register. The Main Status Register indicates when the disk controller is ready to send or receive data. It should be read before each byte is transferred to or from the Data Register except during a DMA transfer. No delay is required when reading this register after a data transfer.

D7 Request for Master: Indicates that the Data Register is ready to send or receive data from the μP. This bit is cleared immediately after a byte transfer and will become set again as soon as the disk controller is ready for the next byte.

D6 Data Direction: Indicates whether the controller is expecting a byte to be written to (0) or read from (1) the Data Register.

D5 Non-DMA Execution: Bit is set only during the Execution Phase of a command if it is in the non-DMA mode. In other words, if this bit is set, the multiple byte data transfer (in the Execution Phase) must be monitored by the μP either through interrupts, or software polling as described in the Processor Software Interface section.

D4 Command in Progress: Bit is set after the first byte of the Command Phase is written. Bit is cleared after the last byte of the Result Phase is read. If there is no result phase in a command, the bit is cleared after the last byte of the Command Phase is written.

D3 Drive 3 Seeking: Set after the last byte of the Command Phase of a Seek or Recalibrate command is issued for drive 3. Cleared after reading the first byte in the Result Phase of the Sense Interrupt Command for this drive.

D2 Drive 2 Seeking: Same as above for drive 2.

D1 Drive 1 Seeking: Same as above for drive 1.

D0 Drive 0 Seeking: Same as above for drive 0.

DATA REGISTER (Read/Write)

This is the location through which all commands, data and status flow between the CPU and the DP8473. During the Command Phase the μP loads the controller's commands into this register based on the Status Register Request for Master and Data Direction bits. The Result Phase transfers the Status Registers and header information to the μP in the same fashion.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Register Description (Continued)

TABLE VI. Data Rate and Precompensation Programming Values

D1	D0**	DRVTYP Pin	Data Rate MFM (kb/s)	Normal Precomp* (ns)	Alternate Precomp* (ns)	FGND Pin Enabled	RPM/LC Pin Level
0	0	X	500	125	125	FGND500	High
0	1	0	250	125	250	FGND250	Low
0	1	1	300	208	208	FGND250	Low
1	0	0	250	125	250	FGND250	Low
1	0	1	250	125	250	FGND250	Low
1	1	0	1000	83	83	None	High
1	1	1	1000	83	83	None	Low

*Normal values when PUMP/PREN pin set low; Alternate values when PUMP/PREN pin set high.

**D0 and D1 are Data Rate Control Bits.

DRIVE CONTROL REGISTER (Write Only)

D7 Motor Enable 3: This controls the Motor for drive 3, MTR3. When 0 the output is high, when 1 the output is low. (Note this signal is not output to a pin on 48 pin DIP version.)

D6 Motor Enable 2: Same function as D7 except for drive 2's motor. (Note this signal is not brought out to a pin on DIP.)

D5 Motor Enable 1: This bit controls the Motor for drive 1's motor. When this bit is 0 the MTR1 output is high.

D4 Motor Enable 0: Same as D5 except for drive 0's motor.

D3 DMA Enable: When set to a 1 this enables the DRQ, DAK, INT pins. A zero disables these signals.

D2 Reset Controller: This bit when set to a 0 resets the controller, and when a 1 enables normal operation. It does not affect the Drive Control or Data Rate Registers which are reset only by a hardware reset.

D1–D0 Drive Select: These two pins are encoded for the four drive selects, and are gated with the motor enable lines, so that only one drive is selected when it's Motor Enable is active. (See Table V.)

DATA RATE REGISTER (Write Only)

D7–D2: Not used.

D1, D0 Data Rate Select: These bits set the data rate and the write precompensation values for the disk controller. After a hardware reset these bits are set to 10 (250 kb/s). They are encoded as shown in Table VI.

DISK CHANGED REGISTER (Read Only)

D7 Disk Changed: This bit is the latched complement of the Disk Changed input pin. If the DSKCHG input is low this bit is high.

D6–D0: These bits are reserved for use by the hard disk controller, thus during a read of this register, these bits are TRI-STATE.

Result Phase Status Registers

The Result Phase of a command contains bytes that hold status information. The format of these bytes are described below. Do not confuse these register bytes with the Main Status Register which is a read only register that is always available. The Result Phase status registers are read from the Data Register only during the Result Phase.

STATUS REGISTER 0 (ST0)

D7–D6 Interrupt Code:

00 = Normal Termination of Command.

01 = Abnormal Termination of Command. Execution of Command was started, but was not successfully completed.

10 = Invalid Command Issue. Command Issued was not recognized as a valid command.

11 = Ready changed state during the polling mode.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Result Phase Status Registers (Continued)

D5 Seek End: Seek or Recalibrate Command completed by the Controller. (Used during Sense Interrupt command.)

D4 Equipment Check: After a Recalibrate Command, Track 0 signal failed to occur. (Used during Sense Interrupt command.)

D3 Not Used: 0

D2 Head Address (at end of Execution Phase).

D1, D0 Drive Select (at end of Execution Phase).

00 = Drive 0 selected. 01 = Drive 1 selected.

10 = Drive 2 selected. 11 = Drive 3 selected.

STATUS REGISTER 1 (ST1)

D7 End of Track: Controller transferred the last byte of the last sector without the TC pin becoming active. The last sector is the End Of Track sector number programmed in the Command Phase.

D6 Not Used: 0

D5 CRC Error: If this bit is set and bit 5 of ST2 is clear, then there was a CRC error in the Address Field of the correct sector. If bit 5 of ST2 is set, then there was a CRC error in the Data Field.

D4 Over Run: Controller was not serviced by the μP soon enough during a data transfer in the Execution Phase.

TABLE VII. Maximum Time Allowed to Service an Interrupt or Acknowledge a DMA Request in Execution Phase

Data Rate	Time to Service
125	62.0 μs
250	30.0 μs
500	14.0 μs
1000	6.0 μs

Time from rising edge of DRQ or INT to trailing edge of $\overline{DAK}$ or $\overline{RD}$ or $\overline{WR}$.

D3 Not Used: 0

D2 No Data: Three possible problems: 1) Controller cannot find the sector specified in the Command Phase during the execution of a Read, Write, or Scan command. An address mark was found however so it is not a blank disk. 2) Controller cannot read any Address Fields without a CRC error during Read ID command. 3) Controller cannot find starting sector during execution of Read A Track command.

D1 Not Writable: Write Protect pin is active when a Write or Format command is issued.

D0 Missing Address Mark: If bit 0 of ST2 is clear then the disk controller cannot detect any Address Field Address Mark after two disk revolutions. If bit 0 of ST2 is set then the disk controller cannot detect the Data Field Address Mark.

STATUS REGISTER 2 (ST2)

D7 Not Used: 0

D6 Control Mark: Controller tried to read a sector which contained a deleted data address mark during execution of Read Data or Scan commands. Or, if a Read Deleted Data command was executed, a regular address mark was detected.

D5 CRC Error in Data Field: Controller detected a CRC error in the Data Field. Bit 5 of ST1 is also set.

D4 Wrong Track: Only set if desired sector not found, and the track number recorded on any sector of the current track is different from that stored in the Track Register.

D3 Scan Equal Hit: "Equal" condition satisfied during any Scan Command.

D2 Scan Not Satisfied: Controller cannot find a sector on the track which meets the desired condition during Scan Command.

D1 Bad Track: Only set if the desired sector is not found, and the track number recorded on any sector on the track is different from that stored in the Track Register and the recorded track number is FF.

D0 Missing Address Mark in Data Field: Controller cannot find the Data Field Address Mark during Read/Scan command. Bit 0 of ST1 is also set.

STATUS REGISTER 3 (ST3)

D7 Not Used: 0

D6 Write Protect Status

D5 Not Used: 1

D4 Track 0 Status

D3 Not Used: 0

D2 Head Select Status

D1, D0 Drive Selected:

00 = Drive 0 selected. 01 = Drive 1 selected.

10 = Drive 2 selected. 11 = Drive 3 selected.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Result Phase Status Registers (Continued)

TABLE VIII. Summary of FDC Registers

Bits Register	7	6	5	4	3	2	1	0
DCR (W)	MTR3 Enable	MTR2 Enable	MTR1 Enable	MTR0 Enable	DMA Enable	Software Reset	Drive Select 1	Drive Select 0
MSR (R)	Request for Master	Data Direction	Non-DMA Execution	Command in Progress	Drive 3 Seeking	Drive 2 Seeking	Drive 1 Seeking	Drive 0 Seeking
DR (R/W)	Data 7	Data 6	Data 5	Data 4	Data 3	Data 2	Data 1	Data 0
DRR (W)	X	X	X	X	X	X	Data Rate Bit 1	Data Rate Bit 0
DKR (R)	DSKCHG Pin Inverse	Z	Z	Z	Z	Z	Z	Z
ST0	Interrupt Code	Interrupt Code	Seek End	No Track 0	0	Head Addr	DRV 1 (Exec.)	DRV 0 (Exec.)
ST1	End of Track	0	CRC Error	Data Overrun	0	No Data	WRT PRT	Missing Address Mark
ST2	0	Control Mark	CRC Error in Data Fld	Wrong Track	Scan Equal Hit	Scan Not Satisfied	Bad Track	Missing Address Mark in Data Fld
ST3	0	Write Protect Status	1	Track 0 Status	0	Head Select Status	DRV 1 (Exec.)	DRV 0 (Exec.)

X = Don't Care
Z = TRI-STATE

Processor Software Interface

Bytes are transferred to and from the disk controller in different ways for the different phases in a command.

COMMAND SEQUENCE

The disk controller can perform various disk transfer, and head movement commands. Most commands involve three separate phases.

Command Phase: The μP writes a series of bytes to the Data Register. These bytes indicate the command desired and the particular parameters required for the command. All the bytes must be written in the order specified in the Command Description Table. The Execution Phase starts immediately after the last byte in the Command Phase is written. Prior to performing the Command Phase, the Drive Control and Data Rate Registers should be set.

Execution Phase: The disk controller performs the desired command. Some commands require the μP to read or write data to or from the Data Register during this time. Reading data from a disk is an example of this.

Result Phase: The μP reads a series of bytes from the data register. These bytes indicate whether the command executed properly and other pertinent information. The bytes are read in the order specified in the Command Description Table.

A new command may be initiated by writing the Command Phase bytes after the last bytes required from the Result Phase have been read. If the next command requires selecting a different drive or changing the data rate the Drive Control and Data Rate Registers should be updated. If the command is the last command, then the software should deselect the drive. *(Note as a general rule the operation of the controller core is independent of how the μP updates the Drive Control and Data Rate Registers. The software must ensure that manipulation of these registers is coordinated with the controller operation.)*

During the Command Phase and the Result Phase, bytes are transferred to and from the Data Register. The Main Status Register is monitored by the software to determine when a data transfer can take place. Bit 6 of the Main Status Register must be clear and bit 7 must be set before a byte can be written to the Data Register during the Command Phase. Bits 6 and 7 of the Main Status Register must

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Processor Software Interface (Continued)

both be set before a byte can be read from the Data Register during the Result Phase.

If there is information to be transferred during the Execution Phase, there are three methods that can be used. The DMA mode is used if the system has a DMA controller. This allows the μP to do other things during the Execution Phase data transfer. If DMA is not used, an interrupt can be issued for each byte transferred during the Execution Phase. If interrupts are not used, the Main Status Register can be polled to indicate when a byte transfer is required.

DMA MODE

If the DMA mode is selected, a DMA request will be generated in the Execution Phase when each byte is ready to be transferred. To enable DMA operations during the Execution Phase, the DMA mode bit in the Specify Command must be enabled, and the DMA signals must be enabled in the Drive Control Register. The DMA controller should respond to the DMA request with a DMA acknowledge and a read or write strobe. The DMA request will be cleared by the active edge of the DMA acknowledge. After the last byte is transferred, an interrupt is generated, indicating the beginning of the Result Phase. During DMA operations the Chip Select input must be held high. TC is asserted to terminate an operation. Due to the internal gating TC is only recognized when the $\overline{DAK}$ input is low.

INTERRUPT MODE

If the non-DMA mode is selected, an interrupt will be generated in the Execution Phase when each byte is ready to be transferred. The Main Status Register should be read to verify that the interrupt is for a data transfer. Bits 5 and 7 of the Main Status Register will be set. The interrupt will be cleared when the byte is transferred to or from the Data Register. The μP should transfer the byte within the time allotted by Table VII. If the byte is not transferred within the time allotted, an Overrun Error will be indicated in the Result Phase when the command terminates at the end of the current sector.

An interrupt will also be generated after the last byte is transferred. This indicates the beginning of the Result Phase. Bits 7 and 6 of the Main Status Register will be set and bit 5 will be clear. This interrupt will be cleared by reading the first byte in the Result Phase.

SOFTWARE POLLING

If the non-DMA mode is selected and interrupts are not suitable, the μP can poll the Main Status Register during the Execution Phase to determine when a byte is ready to be transferred. In the non-DMA mode, bit 7 of the Main Status Register reflects the state of the interrupt pin. Otherwise, the data transfer is similar to the Interrupt Mode described above.

Command Description Table

READ DATA

Command Phase

MT	MFM	SK	0	0	1	1	0
IPS	X	X	X	X	HD	DR1	DR0
Track Number							
Drive Head Number							
Sector Number							
Number of Bytes per Sector							
End of Track Sector Number							
Intersector Gap Length							
Data Length							

Note 1

Result Phase

Status Register 0
Status Register 1
Status Register 2
Track Number
Head Number
Sector Number
Bytes/Sector

READ ID

Command Phase

0	MFM	0	0	1	0	1	0
X	X	X	X	X	HD	DR1	DR0

Result Phase

Status Register 0
Status Register 1
Status Register 2
Track Number
Head Number
Sector Number
Bytes/Sector

FORMAT A TRACK

Command Phase

0	MFM	0	0	1	1	0	1
X	X	X	X	X	HD	DR1	DR0
Number of Bytes per Sector							
Number of Sectors per Track							
Format Gap							
Data Pattern							

Result Phase

Status Register 0
Status Register 1
Status Register 2
Track Number
Head Number
Sector Number
Bytes/Sector

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Command Description Table (Continued)

READ DELETED DATA

Command Phase

MT	MFM	SK	0	1	1	0	0
IPS	X	X	X	X	HD	DR1	DR0
Track Number							
Drive Head Number							
Sector Number							
Number of Bytes per Sector							
End of Track Sector Number							
Intersector Gap Length							
Data Length							

Result Phase

Status Register 0
Status Register 1
Status Register 2
Track Number
Head Number
Sector Number
Bytes/Sector

WRITE DATA

Command Phase

MT	MFM	0	0	0	1	0	1
IPS	X	X	X	X	HD	DR1	DR0
Track Number							
Drive Head Number							
Sector Number							
Number of Bytes per Sector							
End of Track Sector Number							
Intersector Gap Length							
Data Length							

Result Phase

Status Register 0
Status Register 1
Status Register 2
Track Number
Head Number
Sector Number
Bytes/Sector

SCAN EQUAL

Command Phase

MT	MFM	SK	1	0	0	0	1
IPS	X	X	X	X	HD	DR1	DR0
Track Number							
Drive Head Number							
Sector Number							
Number of Bytes per Sector							
End of Track Sector Number							
Intersector Gap Length							
Sector Step Size							

Result Phase

Status Register 0
Status Register 1
Status Register 2
Track Number
Head Number
Sector Number
Bytes/Sector

READ A TRACK

Command Phase

0	MFM	SK	0	0	0	1	0
IPS	X	X	X	X	HD	DR1	DR0
Track Number							
Drive Head Number							
Sector Number							
Number of Bytes per Sector							
End of Track Sector Number							
Intersector Gap Length							
Data Length							

Result Phase

Status Register 0
Status Register 1
Status Register 2
Track Number
Head Number
Sector Number
Bytes/Sector

WRITE DELETED DATA

Command Phase

MT	MFM	0	0	1	0	0	1
IPS	X	X	X	X	HD	DR1	DR0
Track Number							
Drive Head Number							
Sector Number							
Number of Bytes per Sector							
End of Track Sector Number							
Intersector Gap Length							
Data Length							

Result Phase

Status Register 0
Status Register 1
Status Register 2
Track Number
Head Number
Sector Number
Bytes/Sector

SCAN LOW OR EQUAL

Command Phase

MT	MFM	SK	1	1	0	0	1
IPS	X	X	X	X	HD	DR1	DR0
Track Number							
Drive Head Number							
Sector Number							
Number of Bytes per Sector							
End of Track Sector Number							
Intersector Gap Length							
Sector Step Size							

Result Phase

Status Register 0
Status Register 1
Status Register 2
Track Number
Head Number
Sector Number
Bytes/Sector

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Command Description Table (Continued)

SCAN HIGH OR EQUAL

Command Phase

MT	MFM	SK	1	1	1	0	1
IPS	X	X	X	X	HD	DR1	DR0
Track Number							
Drive Head Number							
Sector Number							
Number of Bytes per Sector							
End of Track Sector Number							
Intersector Gap Length							
Sector Step Size							

Result Phase

Status Register 0
Status Register 1
Status Register 2
Track Number
Head Number
Sector Number
Bytes/Sector

SEEK

Command Phase

0	0	0	0	1	1	1	1
X	X	X	X	X	X	DR1	DR0
New Track Number							
MSB of Track				0	0	0	0

Note 2

RECALIBRATE

Command Phase

0	0	0	0	0	1	1	1
0	0	0	0	0	0	DR1	DR0

SENSE INTERRUPT

Command Phase

0	0	0	0	1	0	0	0

Result Phase

Status Register 0				
Present Track Number (PTN)				
MSN PTN				0	0	0	0

Note 2

SENSE DRIVE STATUS

Command Phase

0	0	0	0	0	1	0	0
X	X	X	X	X	HD	DR1	DR0

Result Phase

Status Register 3

SPECIFY

Command Phase

0	0	0	0	0	0	1	1
Step Rate Time (N1)				Motor Off Time (N2)			
Motor On Time (N3)							DMA

MODE

Command Phase

0	0	0	0	0	0	0	1
TMR	IAF	IPS	0	LW	PR	1	ETR
0	0	0	0	0	0	0	0
1	1	0	WLD	Head Settle			
0	0	0	0	0	RG	0	PU

Note 3

SET TRACK

Command Phase

0	$\overline{R}$/W	1	0	0	0	0	1
0	0	1	1	0	MSB	DR1	DR0
New Track Number							

Result Phase

Value

Note 3

INVALID COMMAND

Command Phase

Invalid Op Codes

Result Phase

Status Register 0

Note 1: The IPS bit is only enabled if the IPS bit in the mode command is set. Otherwise this bit is a don't care.

Note 2: Shaded byte only written or read if the extended track range mode is enabled in the Mode Command (ET) = 1.

Note 3: These commands are additional enhanced commands.

Note: Mnemonic Definitions

X = DON'T CARE

MFM = Data Encoding Scheme

MSN PTN = Most Significant Nibble Present Track Number

MT = Multi-Track

IPS = Implied Seek (In individual commands this bit is a don't care unless the IPS bit in the mode command is set.)

SK = Skip Sector

HD = Head Number

DRn = Drive to Select (encoded)

TMR = Motor/Head Timer Mode

IAF = Index Address Field

LW PR = Low Power Mode

ETR = Extended Track Range

WLD = Wildcard in Scan

RG = Enables the Read Gate Input on the DSKCHG pin for the Data Separator.

PU = Enables Charge Pump PUMP signal to be output on the PUMP/PREN pin.

MSB = Selects whether the most significant or least significant byte of the track is read. 1 = MSB.

$\overline{R}$/W = Selects whether the track is written or read (Read = 0, Write = 1).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Command Description

READ DATA

The Read Data op-code is written to the data register followed by 8 bytes as specified in the Command Description Table. After the last byte is written, the controller starts looking for the correct sector header. Once the sector is found the controller sends the data to the μP. After one sector is finished, the Sector Number is incremented by one and this new sector is searched for. If MT (Multi-Track) is set, both sides of one track can be read. Starting on side zero, the sectors are read until the sector number specified by End of Track Sector Number is reached. Then, side one is read starting with sector number one.

In DMA mode the Read Data command continues to read until the TC pin is set. This means that the DMA controller should be programmed to transfer the correct number of bytes. TC could be controlled by the μP and be asserted when enough bytes are received. An alternative to these methods of stopping the Read Data command is to program the End of Track Sector Number to be the last sector number that needs to be read. The controller will stop reading the disk with an error indicating that it tried to access a sector number beyond the end of the track.

The Number of Data Bytes per Sector parameter is defined in Table IX. If this is set to zero then the Data Length parameter determines the number of bytes that the controller transfers to the μP. If the data length specified is smaller than 128 the controller still reads the entire 128 byte sector and checks the CRC, though only the number of bytes specified by the Data Length parameter are transferred to the μP. Data Length should not be set to zero. If the Number of Bytes per Sector parameter is not zero, the Data Length parameter has no meaning and should be set to FF (hex).

If the Implied Seek Mode is enabled by both the Mode command and the IPS bit in this command, a Seek will be performed to the track number specified in the Command Phase. The controller will also wait the Head Settle time if the implied seek is enabled.

After all these conditions are met, the controller searches for the specified sector by comparing the track number, head number, sector number, and number bytes/sector given in the Command Phase with the appropriate bytes read off the disk in the Address Fields.

If the correct sector is found, but there is a CRC error in the Address Field, bit 5 of ST1 (CRC Error) is set and an abnormal termination is indicated. If the correct sector is not found, bit 2 of ST1 (No Data) is set and an abnormal termination is indicated. In addition to this, if any Address Field track number is FF, bit 1 of ST2 (Bad Track) is set or if any Address Field track number is different from that specified in the Command Phase, bit 4 of ST2 (Wrong Track) is set.

After finding the correct sector, the controller reads that Data Field. If a Deleted Data Mark is found and the SK bit is set, the sector is not read, bit 6 of ST2 (Control Mark) is set, and the next sector is searched for. If a deleted data mark is found and the SK bit is not set, the sector is read, bit 6 of ST2 (Control Mark) is set, and the read terminates with a normal termination. If a CRC error is detected in the Data Field, bit 5 is set in both ST1 and ST2 (CRC Error) and an abnormal termination is indicated.

If no problems occur in the read command, the read will continue from one sector to the next in logical order (not physical order) until either TC is set or an error occurs.

If a disk has not been inserted into the disk drive, there are many opportunities for the controller to appear to hang up. It does this if it is waiting for a certain number of disk revolutions for something. If this occurs, the controller can be forced to abort the command by writing a byte to the Data register. This will place the controller into the Result Phase.

TABLE IX. Sector Size Selection

Bytes/Sector Code	Number of Bytes in Data Field
0	128
1	256
2	512
3	1024
4	2048
5	4096
6	8192

An interrupt will be generated when the Execution Phase of the Read Data command terminates. The values that will be read back in the Result Phase are shown in Table X. If an error occurs, the result bytes will indicate the sector being read when the error occurred.

READ DELETED DATA

This command is the same as the Read Data command except for its treatment of a Deleted Data Mark. If a Deleted

TABLE X. Result Phase Termination Values with No Error

MT	HD	Last Sector	ID Information at Result Phase			
			Track	Head	Sector	B/S
0	0	< EOT	NC	NC	S+1	NC
0	0	= EOT	T+1	NC	1	NC
0	1	< EOT	NC	NC	S+1	NC
0	1	= EOT	T+1	NC	1	NC
1	0	< EOT	NC	NC	S+1	NC
1	0	= EOT	NC	1	1	NC
1	1	< EOT	NC	NC	S+1	NC
1	1	= EOT	T+1	0	1	NC

EOT = End of Track Sector Number from Command Phase

NC = No Change in Value

S = Sector Number last operated on by controller

T = Track Number programmed in Command Phase

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Command Description (Continued)

Data Mark is read, the sector is read normally. If a Regular Data Mark is found and the SK bit is set, the sector is not read, bit 6 of ST2 (Control Mark) is set, and the next sector is searched for. If a Regular Data Mark is found and the SK bit is not set, the sector is read, bit 6 of ST2 (Control Mark) is set, and the read terminates with a normal termination.

WRITE DATA

The Write Data command is very similar to the Read Data command except that data is transferred from the μP to the disk rather than the other way around. If the controller detects the Write Protect signal, bit 1 of ST1 (Not Writable) is set and an abnormal termination is indicated.

WRITE DELETED DATA

This command is the same as the Write Data Command except a Deleted Data Mark is written at the beginning of the Data Field instead of the normal Data Mark.

READ A TRACK

This command is similar to the Read Data command except for the following. The controller starts at the index hole and reads the sectors in their physical order, not their logical order.

Even though the controller is reading sectors in their physical order, it will still perform a comparison of the header ID bytes with the Data programmed in the Command Phase. The exception to this is the sector number. Internally, this is initialized to a one, and then incremented for each successive sector read. Whether or not the programmed Address Field matches that read from the disk, the sectors are still read in their physical order. If a header ID comparison fails, bit 2 of ST1 (No Data) is set, but the operation will continue. If there is a CRC error in the Address Field or the Data Field, the read will also continue.

The command will terminate when it has read the number of sectors programmed in the EOT parameter.

READ ID

This command will cause the controller to read the first Address Field that it finds. The Result Phase will contain the header bytes that are read. There is no data transfer during the Execution Phase of this command. An interrupt will be generated when the Execution Phase is completed.

FORMAT A TRACK

This command will format one track on the disk. After the index hole is detected, data patterns are written on the disk including all gaps, address marks, Address Fields, and Data Fields. The exact details of the number of bytes for each field is controlled by the parameters given in the Format A Track command, and the IAF (Index Address Field) bit in the Mode command. The Data Field consists of the Fill Byte specified in the command, repeated to fill the entire sector.

To allow for flexible formatting, the μP must supply the four Address Field bytes (track, head, sector, number of bytes) for each sector formatted during the Execution Phase. In other words, as the controller formats each sector, it will request four bytes through either DMA requests or interrupts. This allows for non-sequential sector interleaving. Some typical values for the programmable GAP size are shown in Table XI.

The Format Command terminates when the index hole is detected a second time, at which point an interrupt is generated. Only the first three status bytes in the Result Phase are significant.

TABLE XI. Gap Length for Various Sector Sizes and Disk Types

Format Table for Sector Size and Data Rate

Mode	Sector Size	Sector Code	EOT	Sector Gap	Format Gap
	decimal	hex	hex	hex	hex
FM 125 kb/s	128	00	12	07	09
	128	00	10	10	19
	256	01	08	18	30
	512	02	04	46	87
	1024	03	02	C8	FF
	2048	04	01	C8	FF
MFM 250 kb/s	256	01	12	0A	0C
	256	01	10	20	32
	512	02	08	2A	50
	512	02	09	1B	50
	1024	03	04	80	F0
	2048	04	02	C8	FF
	4096	05	01	C8	FF
FM 250 kb/s	128	00	1A	07	1B
	256	01	0F	0E	2A
	512	02	08	1B	3A
	1024	03	04	47	8A
	2048	04	02	C8	FF
	4096	05	01	C8	FF
MFM 500 kb/s	256	01	1A	0E	36
	512	02	0F	1B	54
	512	02	12	1B	6C
	1024	03	08	35	74
	2048	04	04	99	FF
	4096	05	02	C8	FF
	8192	06	01	C8	FF

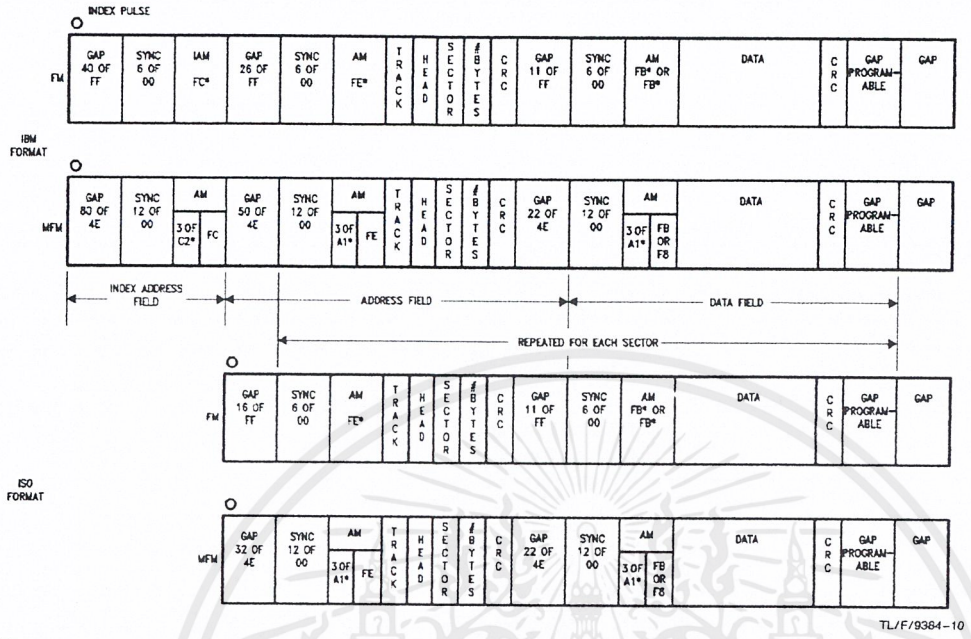
Format Table for PC Compatible Diskette Media

Media Type	Sector Size	Sector Code	EOT	Sector Gap	Format Gap
	decimal	hex	hex	hex	hex
360k	512	02	09	2A	50
1.2M	512	02	0F	1B	54
720k	512	02	09	1B	50
1.44M	512	02	12	1B	6C

Note: Format Gap is the gap length used only for the Format command.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Command Description (Continued)



TL/F/9384–10

Notes:

FE* = Data pattern of FE, Clock pattern of C7
FC* = Data pattern of FC, Clock pattern of D7
FB* = Data pattern of FB, Clock pattern of C7
F8* = Data pattern of F8, Clock pattern of C7
A1* = Data pattern of A1, Clock pattern of 0A
C2* = Data pattern of C2, Clock pattern of 14

All byte counts in decimal.
All byte values in hex.
CRC uses standard polynomial $x^{16} + x^{12} + x^5 + 1$.

FIGURE 6. IBM and ISO Formats Supported by the Format Command

SCAN COMMANDS

The Scan Commands allow data read from the disk to be compared against data sent from the μP. There are three Scan Commands to choose from:

Scan Equal	Disk Data = μP Data
Scan Less Than or Equal	Disk Data ≤ μP Data
Scan Greater Than or Equal	Disk Data ≥ μP Data

Each sector is interpreted with the most significant bytes first. If the Wildcard mode is enabled from the Mode command, an FF(hex) from either the disk or the μP is used as a don't care byte that will always match equal. After each sector is read, if the desired condition has not been met, the next sector is read. The next sector is defined as the current sector number plus the Sector Step Size specified. The Scan command will continue until the scan condition has been met, or the End of Track Sector Number has been reached, or if TC is asserted.

If the SK bit is set, sectors with deleted data marks are ignored. If all sectors read are skipped, the command will terminate with D3 of ST2 set (Scan Equal Hit). The result phase of the command is shown in Table XII.

TABLE XII. Scan Command Termination Values

Command	Status Register 2		Conditions
	D2	D3	
Scan Equal	0	1	Disk = μP
	1	0	Disk ≠ μP
Scan Low or Equal	0	1	Disk = μP
	0	0	Disk < μP
	1	0	Disk > μP
Scan High or Equal	0	1	Disk = μP
	0	0	Disk > μP
	1	0	Disk < μP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Command Description (Continued)

SEEK

There are two ways to move the disk drive head to the desired track number. Method One is to enable the Implied Seek Mode. This way each individual Read or Write command will automatically move the head to the track specified in the command.

Method Two is using the Seek Command. During the Execution Phase of the Seek Command, the track number to seek to is compared with the present track number and a step pulse is produced to move the head one track closer to the desired track number. This is repeated at the rate specified by the Specify Command until the head reaches the correct track. At this point an interrupt is generated and a Sense Interrupt Command is required to clear the interrupt.

During the Execution Phase of the Seek Command the only indication via software that a Seek Command is in progress is bits 0–3 (Drive Busy) of the Main Status Register. Bit 4 of the Main Register (Controller Busy) is not set. While the internal microengine is capable of multiple seeks on 2 or more drives at the same time since the drives are selected via the Drive Control Register in software, software should ensure that only one drive is seeking at one time. No other command except the Sense Interrupt Command should be issued while a Seek Command is in progress.

If the extended track range mode is enabled, a fourth byte should be written in the Command Phase to indicate the four most significant bits of the desired track number. Otherwise, only three bytes should be written.

RECALIBRATE

The Recalibrate Command is very similar to the Seek Command. It is used to step a drive head out to track zero. Step pulses will be produced until the track zero signal from the drive becomes true. If the track zero signal does not go true before 77 step pulses are issued, an error is generated. If the extended track range mode is enabled, an error is not generated until 3917 pulses are issued.

Recalibrations on more than one drive at a time should not be issued for the same reason as explained in the Seek Command. No other command except the Sense Interrupt Command should be issued while a Recalibrate Command is in progress.

SENSE INTERRUPT STATUS

An interrupt is generated by the controller when any of the following conditions occur:

1. Upon entering the Result Phase of:
   a. Read Data Command
   b. Read Deleted Data Command
   c. Write Data Command
   d. Write Deleted Data Command
   e. Read a Track Command
   f. Read ID Command
   g. Format Command
   h. Scan Commands
2. During data transfers in the Execution Phase while in the Non-DMA mode
3. Internal Ready signal changes state (only occurs immediately after a hardware or software reset).
4. Seek or Recalibrate Command termination

An interrupt generated for reasons 1 and 2 above occurs during normal command operations and are easily discernible by the μP. During an execution phase in Non-DMA Mode, bit 5 (Execution Mode) in the Main Status Register is set to 1. Upon entering Result Phase this bit is set to 0. Reasons 1 and 2 do not require the Sense Interrupt Status command. The interrupt is cleared by reading or writing information to the data register.

Interrupts caused by reasons 3 and 4 are identified with the aid of the Sense Interrupt Status Command. This command resets the interrupt when the command byte is written. Use bits 5, 6 and 7 of ST0 to identify the cause of the interrupt as shown in Table XIII.

TABLE XIII. Status Register 0 Termination Codes

Status Register 0			Cause
Interrupt Code		Seek End	
D7	D6	D5	
1	1	0	Internal Ready Went True
0	0	1	Normal Seek Termination
0	1	1	Abnormal Seek Termination

TABLE XIV. Step, Head Load and Unload Timer Definitions (500 kb/s MFM)

Timer	Mode 1		Mode 2		Unit
	Value	Range	Value	Range	
Step Rate	(16 − N1)	1–16	(16 − N1)	1–16	ms
Motor Off	N2 × 16	0–240	N2 × 512	0–7680	ms
Motor On	N3 × 2	0–254	N3 × 32	0–4064	ms

Issuing a Sense Interrupt Status Command without an interrupt pending is treated as an invalid command.

If the extended track range mode is enabled, a third byte should be read in the Result Phase which will indicate the four most significant bits of the Present Track Number. Otherwise, only two bytes should be read.

SPECIFY

The Specify Command sets the initial values for each of the three internal timers. The timer programming values are shown in Table XIV.

The Head Load and Head Unload timers are artifacts of the μPD765A. These timers determine the delay from loading the head until a read or write command is started, and unloading the head sometime after the command was completed. Since the DP8473's head load signal is now the software controlled Motor lines in the Drive Control Register, these timers only provide some delay from the initiation of a command until it is actually started. These times can be extended by setting the TMR bit in the Mode Command.

The Step Rate Time defines the time interval between adjacent step pulses during a Seek, Implied Seek, or Recalibrate Command.

The times stated in the table are affected by the Data Rate. The values in the table are for 500 kb/s MFM (250 kb/s FM) and 1 Mb/s MFM (500 kb/s FM). For a 300 kb/s MFM data rate (150 kb/s FM) these values should be multiplied by 1.6667, and for 250 kb/s MFM (125 kb/s FM) these values should be doubled.

The choice of DMA or Non-DMA operation is made by the NON-DMA bit. When this bit is 1 then Non-DMA mode is selected, and when this bit is 0, the DMA mode is selected.

This command does not generate an interrupt.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Command Description (Continued)

SENSE DRIVE STATUS

This two byte command obtains the status of a disk drive. Status Register 3 is returned in the result phase and contains the drive status. This command does not generate an interrupt.

MODE

This command is used to select the special features of the controller. The bits for the command phase bytes are shown in the command description table, and their function is described below. The defaults after a hardware or software reset are shown by the "bullets" to the left of each item.

- **TMR = 0 (motor TiMeR):** Timers for motor on and motor off are defined for Mode 1. (See Specify Command)
  TMR = 1: Timers for motor on and motor off are defined for Mode 2. (See Specify Command)

LW PR (LoW PoweR)

- 00 Completely disable the low power mode. (default)
  01 Go into low power mode 500 ms after the head unload timer times out.
  10 Go into low power mode now.
  11 Not Used.
- **IAF = 0 (Index Address Format):** The controller will format tracks with the Index Address Field included. (IBM Format)
  IAF = 1: The controller will format tracks without including the Index Address Mark Field. (ISO Format)
- **IPS = 0 (ImPlied Seek):** The implied seek bit in the command is ignored.
  IPS = 1: The implied seek bit in the command is enabled so that if the bit is set in the command, a Seek will be performed automatically.
- **ETR = 0 (Extended Track Range):** Header format is the IBM System 34 (double density) or System 3740 (single density).
  ETR = 1: Header format is the same as above but there are 12 bits of track number. The MSB's of the track number are in the upper four bits of the head number byte.
- **WLD = 0 (scan WiLD card):** An FF(hex) from either the μP or the disk during a Scan Command is interpreted as a wildcard character that will always match true.
  WLD = 1: The Scan commands do not recognize FF(hex) as a wildcard character.

Head Settle: Time allowed for head to settle after an Implied Seek. Time = N × 4 ms, (0 ms–60 ms). (Based on 500 kb/s and 1 Mb/s MFM data rates. Double for 250 kb/s.)

PU (PUMP Pulse Output): When set enables a signal that indicates when the Data Separator's charge pump is making a phase correction. This is a series of pulses. This signal is output on the PUMP/PREN pin when this bit is set.

This is intended as a test mode to aid in evaluation of the Data Separator. (Default mode is off)

RG (Read Gate): Like the PUMP output, when this bit is set it enables a pin (the DSKCHG pin) to act as an external Read Gate signal for the Data Separator. This is intended as a test mode to aid in evaluation of the Data Separator. (Default mode is off)

SET TRACK

This command is used to inspect or change the value of the internal Present Track Register. This could be useful for recovery from disk mis-tracking errors, where the real current track could be read through the Read ID command and then the Set Track Command can set the internal present track register to the correct value.

The first byte of the command contains the command opcode and the R/W bit. If the R/W bit is low, a track register is to be read. In this case, the result phase contains the value in the internal register specified, and the third byte of the command is a dummy byte.

If the R/W bit is high, data is written to a track register. In this case the 3rd byte of the command phase is forced into the specified internal register, and the result phase contains the new byte value written.

The particular track register chosen to operate on is determined by the least significant 3 bits of the second byte of the command. The two LSB's select the drive (DR1, DR0), and the next bit (MSB) determines whether the least significant byte (MSB = 0) or the most significant byte (MSB = 1) of the track register is to be read/written. When not in the extended track range mode, only the LSB track register need be updated. In this instance, the MSB bit is set to 0.

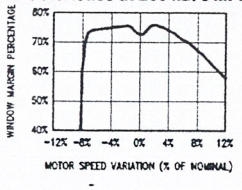
This command does not generate an interrupt.

INVALID COMMAND

If an invalid command (i.e., a command not defined) is received by the controller, the controller will respond with ST0 in the Result Phase. The Controller does not generate an interrupt during this condition. Bits 6 and 7 in the Main Status Register are both set to one's indicating to the processor that the Controller is in the Result Phase and the contents of ST0 must be read. When the system reads ST0 it will find an 80(hex) indicating an invalid command was received.

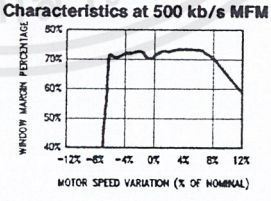
Typical Performance Characteristics

Typical Window Margin Performance Characteristics at 250 kb/s MFM



TL/F/9384–18

Typical Window Margin Performance Characteristics at 500 kb/s MFM



TL/F/9384–19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Absolute Maximum Ratings (Notes 1 and 2)

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

Supply Voltage (V_{CC})	−0.5V to +7V
DC Input Voltage (V_{IN})	−0.5V to V_{CC} + 0.5V
DC Output Voltage (V_{OUT})	−0.5V to V_{CC} + 0.5V
Storage Temperature Range (T_{STG})	−65°C to +165°C
Package Power Dissipation (P_D)	750 mW
Lead Temperature (T_L) (Soldering, 10 seconds)	260°C
$\lvert V_{CC}-V_{CCA}\rvert$	0.6V

Operating Conditions

	Min	Max	Units
Supply Voltage (V_{CC})	4.5	5.5	V
Operating Temperature (T_A)	0	+70	°C
ESD Tolerance: C_{ZAP} = 100 pF, R_{ZAP} = 1.5 kΩ (Note 5)	1500		V

DC Electrical Characteristics V_{CC} = 5V ±10% unless otherwise specified (Note 3)

Symbol	Parameter	Conditions	Min	Max	Units
V_{IH}	High Level Input Voltage	(except OSC2/CLK)	2.0		V
V_{IL}	Low Level Input Voltage	(except OSC2/CLK)		0.8	V
I_{IN}	Input Current (except OSC pins)	V_{IN} = V_{CC} or GND		±1.0	μA
I_{CCA}	Average V_{CCA} Supply Current	V_{IN} = 2.4V or 0.5V, I_O = 0 mA (Note 4)		20.0	mA
	Quiescent V_{CCA} Supply Current in Low Power Mode	V_{IN} = V_{CC} or GND, I_O = 0 mA (Note 4)		400	μA
I_{CC}	Average V_{CC} Supply Current	V_{IN} = 2.4V or 0.5V, I_O = 0 mA (Note 4)		20.0	mA
	Quiescent V_{CC} Supply Current in Low Power Mode	V_{IN} = V_{CC} or GND, I_O = 0 mA (Note 4)		2	mA
OSCILLATOR PINS (OSC2/CLK)					
I_{OSC}	OSC2 Input Current (OSC1 = GND)	V_{IN} = V_{CC} or GND	±1.6		mA
V_{IH}	OSC2 High Level Input Voltage	OSC1 = GND	2.4		V
V_{IL}	OSC2 Low Level Input Voltage	OSC1 = GND		0.4	V
MICROPROCESSOR INTERFACE PINS (D0–D7, INT, DAK, TC, DRQ, RD, WR, CS, A0–A3)					
V_{OH}	High Level Output Voltage	I_{OUT} = −20 μA I_{OUT} = −4.0 mA	V_{CC} − 0.1 3.5		V V
V_{OL}	Low Level Output Voltage	I_{OUT} = 20 μA I_{OUT} = 12 mA		0.1 0.4	V V
I_{OZ}	Output TRI-STATE® Leakage Current	V_{OUT} = V_{CC} or GND		±10.0	μA
DISK DRIVE INTERFACE PINS (MTR0–3, DR0–3, WDATA, WGATE, RDATA, DIR, HDSEL, TRK0, WRTPRT, RPM, STEP, DSKCHG, INDEX)					
V_H	Input Hysteresis		250 Typical		mV
V_{OL}	Low Level Output Voltage	I_{OUT} = 48 mA		0.4	V
I_{LKG}	Output High Leakage Current	V_{OUT} = V_{CC} or GND		±100	μA
V_{IH}	High Level Input Voltage		2.2		V
V_{IL}	Low Level Input Voltage			0.8	V

Note 1: Absolute Maximum Ratings are those values beyond which damage to the device may occur.

Note 2: Unless otherwise specified all voltages are referenced to ground.

Note 3: These DC Electrical Characteristics are measured statidy, and not under dynamic conditions.

Note 4: I_{CC} is measured with a 0.1 μF supply decoupling capacitor to ground.

Note 5: Value based on test complying with NSC SOP5-026 human body model ESD testing using the ETS-910 tester.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Phase Locked Loop Characteristics V_{CC} = 5V ±10%, F_{XTAL} = 24 MHz unless otherwise specified

Symbol	Parameter	Conditions	Typ		Units
V_{REF}	SETCUR Pin Reference Voltage	R_1 = 5.6 kΩ, V_{CC} = 5V	1.1		V
K_{VCO}	VCO Gain (Note 5)	t_{DATA} = 1 μs ±10%	25		Mrad/s/V
R_1	Recommended Pump Resistor Range		3–12		kΩ
$K_{P(UP)}$	Charge Pump Up Current Gain ($I_{REF}/I_{P(UP)}$) (Note 6)	R_1 = 5.6 kΩ	2.50		(none)
$K_{P(DWN)}$	Charge Pump Down Current Gain ($I_{REF}/I_{P(DWN)}$) (Note 6)	R_1 = 5.6 kΩ	2.25		(none)
K_{PLL}	Internal Phase Locked Loop Gain (Note 7)	(R_1 = 5.6 kΩ) Pump Up Pump Down	75 70		Mrad Mrad
T_{SW}	Static Window (Note 8)		**Early**	**Late**	
		(R_1 = 5.6 kΩ) 250 kb/s 500 kb/s 1.0 Mb/s	 1075 530 259	 872 440 234	 ns ns ns
T_{DW}	Dynamic Window Margin	(Note)	70		%

Note: Measurements made with a repeating 'DB6" data pattern with reverse write precompensation, using recommended filter values for the configuration shown in Figure 4c. 25°C, 5.0V, 0% MSV.

Note 5: The VCO gain is measured at the 1.0 Mb/s data rate by forcing the data period over a range from 900 ns to 1100 ns, and measuring the resulting voltage on the filter pin. The best straight line gain is fit to the measured points.

Note 6: This is the current gain of the charge pump, which is defined as the output current divided by the current through R_1.

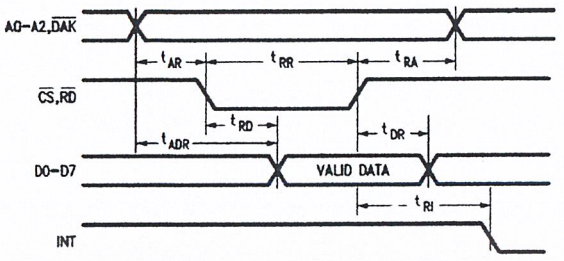
Note 7: This is the product of: $V_{REF} \times K_P \times K_{VCO}$. The total variation in this specification indicates the total loop gain variation contributed by the internal circuitry. The K_{VCO} portion of this specification is measured at the 1.0 Mb/s data rate by forcing the data period over a range of 900 ns to 1100 ns, and measuring the resultant K_{VCO}. K_P is measured by forcing the Filter pin to 2.1V and measuring the ratio of the charge pump current over the input current.

Note 8: The DP8473 is guaranteed to correctly decode a single shifted clock pulse at the end of a long series of non-shifted preamble bits as long as the single shifted pulse is shifted less than the amount specified in T_{SW}. The length of the preamble is long enough for the PLL to lock. The filter components used are those in Table II.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AC Electrical Characteristics

MICROPROCESSOR READ TIMING

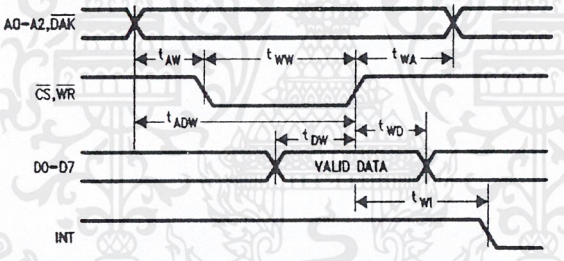


TL/F/9384–11

Symbol	Parameter	Min	Max	Units
t_{AR}	Address Valid prior to Read Strobe	10		ns
t_{RA}	Address Hold from Read Strobe	0		ns
t_{RR}	Read Strobe Width	75		ns
t_{RD}	Read Strobe and Chip Select to Data Valid		75	ns
t_{ADR}	Address Valid to Read Data		85	ns
t_{DR}	Data Hold from Read Strobe to High Impedance (TRI-STATE Note)	5	60	ns
t_{RI}	Clear INT from Read Strobe		65	ns

TRI-STATE Note: This limit includes the RC delay inherent in our test method. This signal will typically turn off within 15 ns, enabling other devices to drive this signal with no contention.

MICROPROCESSOR WRITE TIMING



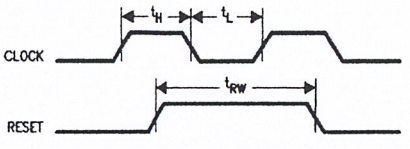
TL/F/9384–12

Symbol	Parameter	Min	Max	Units
t_{AW}	Address Valid to Leading Edge of Write Strobe	10		ns
t_{WA}	Address Hold from Write Strobe	0		ns
t_{WW}	Write Strobe Width	25		ns
t_{ADW}	Address Valid to Trailing Edge of Write Strobe	35		ns
t_{DW}	Data Setup to End of Write Strobe or Chip Select	20		ns
t_{WD}	Data Hold from Write Strobe	12		ns
t_{WI}	Clear INT from Write Strobe		65	ns

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AC Electrical Characteristics (Continued)

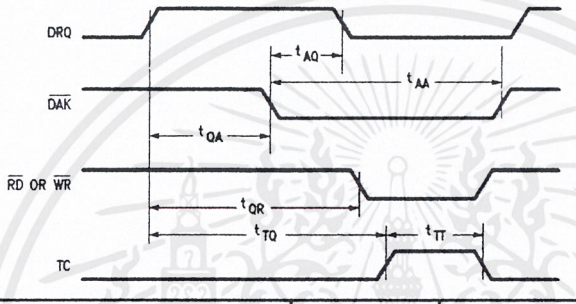
OSC2/CLOCK AND RESET TIMING



TL/F/9384–13

Symbol	Parameter	Min	Max	Units
t_H	Clock High Time	16		ns
t_L	Clock Low Time	16		ns
t_{RW}	Reset Pulse Width	100		ns

DMA TIMING (Note 9)



TL/F/9384–14

Symbol	Parameter	Min	Max	Units
t_{AQ}	End of DRQ from DAK		115	ns
t_{QA}	DAK Assertion from DRQ	10		ns
t_{AA}	DAK Pulse Width	75		ns
t_{QR}	DRQ to Read or Write Strobe	10		ns
t_{TT}	TC Strobe Width	50		ns
t_{TQ}	Time after Last DRQ That TC Must Be Asserted By		(Note 10)	ns

Note 9: DMA Acknowledge is sufficient to acknowledge a data transfer. Read or Write Strobes are neccessary only if data is to be presented to the data bus. If Read/Write Strobes are applied, then they and the Acnowledge must be removed within 1 μs of each other.

Note 10: TC is is the terminal count pin which terminates the data transfer operation. There are several constraints placed on the timing of TC. 1) TC is enabled by $\overline{DAK}$, so TC must be pulsed while $\overline{DAK}$ is low. 2) TC must occur before ((1/data rate x 8) − 1 μs). Data rate is the exact data transfer rate being used.

DRIVE READ TIMING



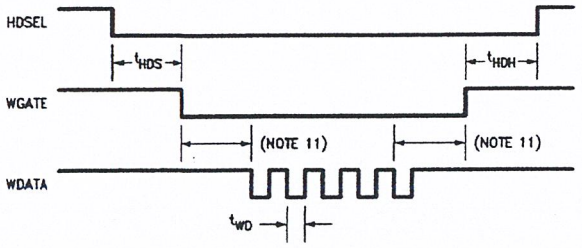
TL/F/9384–15

Symbol	Parameter	Min	Max	Units
t_{RDW}	Read Data Pulse Width	25		ns

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AC Electrical Characteristics (Continued)

DRIVE WRITE TIMING

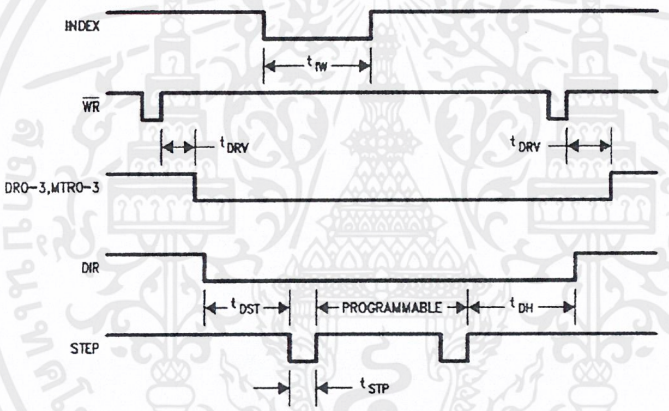


TL/F/9384–16

Symbol	Parameter	Conditions	Min	Max	Units
t_{WD}	Write Data Pulse Width	250 kb/s (MFM)	500		ns
		300 kb/s (MFM)	416		ns
		500 kb/s (MFM)	250		ns
		1000 kb/s(MFM)	225		ns
t_{HDS}	Head Select Setup to Write Gate Assertion		50		μs
t_{HDH}	Head Select Hold from Write Gate		15		μs

Note 11: Whenever WGATE is asserted the WDATA line is active. At the end of each write one dummy byte is written before WGATE is deasserted.

DRIVE TRACK ACCESS TIMING



TL/F/9384–17

Symbol	Parameter	Min	Max	Units
t_{DST}	Direction Setup prior to Step	6		μs
t_{DH}	Direction Hold from End of Step	1 step time		
t_{STP}	Step Pulse Width	8		μs
t_{IW}	Index Pulse Width	100		ns
t_{DRV}	Drive Select or Motor Time from Write Strobe		100	ns

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AC Test Conditions (Notes 11, 12, 13)

Input Pulse Levels	GND to 3V
Input Rise and Fall Times	6 ns
Input and Output Reference Levels	1.3V
TRI-STATE Reference Levels	Active High − 0.5V Active Low + 0.5V

Note 11: C_L = 100 pF, includes jig and scope capacitance.

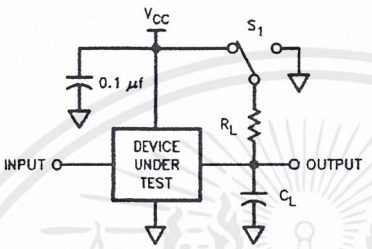
Note 12: S1 = open for push-pull outputs. S1 = V_{CC} for high impedance to active low and active low to high impedance measurements. S1 = GND for high impedance to active high and active high to high impedance measurements. R_L = 1.0 kΩ for μP interface pins.

Note 13: For the Open Drain Drive Interface Pins S1 = V_{CC} and R_L = 150Ω.

Capacitance T_A = 25°C, f = 1 MHz (Note 14)

Symbol	Parameter	Typ	Units
C_{IN}	Input Capacitance	5	pF
C_{OUT}	Output Capacitance	8	pF

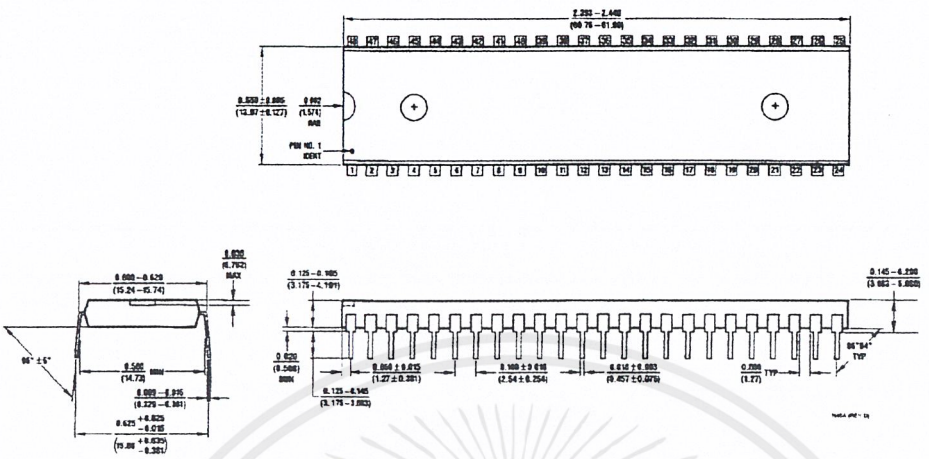
Note 14: This parameter is not 100% tested.



TL/F/9384–20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Physical Dimensions inches (millimeters)

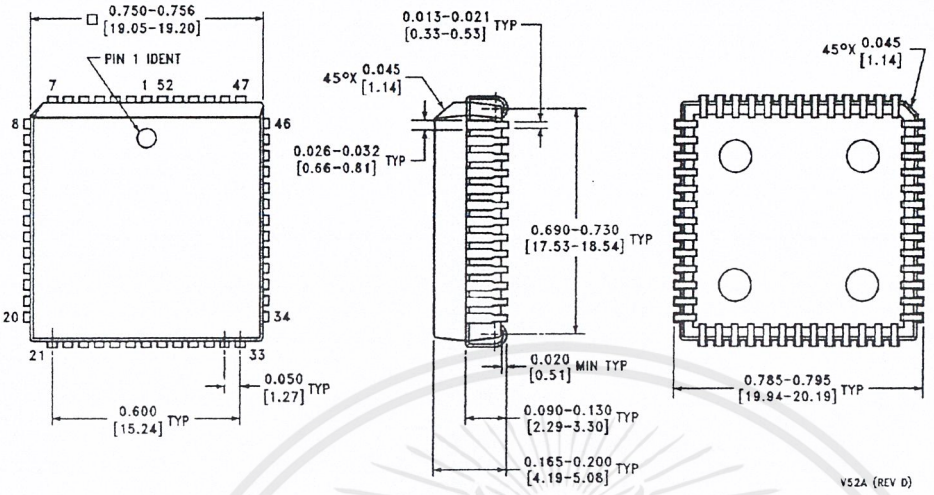


Plastic Dual-In-Line Package (N)
Order Number DP8473N
NS Package Number N48A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Physical Dimensions inches (millimeters) (Continued)

Lit. # 103175



Plastic Leaded Chip Carrier (V)
Order Number DP8473V
NS Package Number V52A

LIFE SUPPORT POLICY

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

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

National Semiconductor Corporation
1111 West Bardin Road
Arlington, TX 76017
Tel: 1(800) 272-9959
Fax: 1(800) 737-7018

National Semiconductor Europe
Fax: (+49) 0-180-530 85 86
Email: cnjwge@tevm2.nsc.com
Deutsch Tel: (+49) 0-180-530 85 85
English Tel: (+49) 0-180-532 78 32
Français Tel: (+49) 0-180-532 93 58
Italiano Tel: (+49) 0-180-534 16 80

National Semiconductor Hong Kong Ltd.
13th Floor, Straight Block,
Ocean Centre, 5 Canton Rd.
Tsimshatsui, Kowloon
Hong Kong
Tel: (852) 2737-1600
Fax: (852) 2736-9960

National Semiconductor Japan Ltd.
Tel: 81-043-299-2309
Fax: 81-043-299-2408

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. ธีรวัฒน์ ประกอบผล , "การประยุกต์ใช้งานไมโครคอนโทรลเลอร์" , สมาคมส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น) , 234 หน้า , 2541
2. วศิน เพิ่มทรัพย์ , " คัมภีร์ DOS 6.22" , บริษัท โปรวิชั่น จำกัด , 588 หน้า , 2538
3. ยืน ภู่วรวรรณ , "เทคโนโลยีฮาร์ดแวร์ IBM PC" , บริษัท ซีเอ็ดยูเคชั่น จำกัด(มหาชน) , 2533
4. สัจจะ จรัสรุ่งรวีวร , "Visual Basic 5.0" , บริษัท ดวงกมลสมัย จำกัด
5. ฉัททวุฒิ พืชผล , "คู่มือเรียน Visual Basic 6.0" , บริษัท โปรวิชั่น จำกัด , 472 หน้า , 2542
6. ISO 7811, "Identification cards-Recording", 1985

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ทางผู้จัดทำโครงงานนี้ ขอขอบพระคุณอาจารย์ชินภัทร นันทจิวากรชัย อาจารย์ที่ปรึกษาเป็นอย่างยิ่ง ที่ได้ให้คำแนะนำและคำปรึกษาในระหว่างขั้นตอนการทำงาน ขอขอบคุณเพื่อน ๆ และพี่ ๆ ที่ได้ให้คำแนะนำ ความช่วยเหลือสำหรับการทำโครงงานนี้ และก็ขอขอบคุณอาจารย์พิชัยที่ช่วยเอื้อเฟื้อให้ใช้ห้อง B411 ในการทำโครงงานนี้ รวมถึงทุก ๆ คนที่ไม่ได้กล่าวถึง ที่มีส่วนช่วยให้โครงงานนี้มีความก้าวหน้ามากขึ้นในทุกส่วนด้วย

วัชรพงษ์ เขื่อนขันธ์

(นายวัชรพงษ์ เขื่อนขันธ์)

วัชรา อนุรักษ์

(นางสาววัชรา อนุรักษ์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้