สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบตรวจสอบการเข้าออกสำนักงาน

OFFICE ACCESS CHECKING SYSTEM

T 0 5 0 0 8 1

โดย

นายชินวัชร์ ดำพัฒน์

นายพชรพร ศิระพลานนท์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2545

เลขหมู่.....
เลขทะเบียน 50081
วัน,เดือน,ปี 21 มิ.ย. 2547

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบตรวจสอบการเข้าออกสำนักงาน

OFFICE ACCESS CHECKING SYSTEM

โดย

นายชินวัชร์ ดำพัฒน์ 42015054

นายพชรพร ศิระพลานนท์ 43015024

อาจารย์ที่ปรึกษา

รศ.สมยศ จุณณะปิยะ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2545

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2545

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง **ระบบตรวจสอบการเข้าออกสำนักงาน**

OFFICE ACCESS CHECKING SYSTEM

ผู้จัดทำ

1. นายชินวัชร์ ดำพัฒน์ 42015054
2. นายพชรพร ศิระพลานนท์ 43015024

..... อาจารย์ที่ปรึกษา

(รศ.สมยศ จุณณะปิยะ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบตรวจสอบการเข้าออกสำนักงาน

OFFICE ACCESS CHECKING SYSTEM

โดย นายชินวัชร์ ดำพัฒน์ 42015054
นายพชรพร ศิระพลานนท์ 43015024

อาจารย์ที่ปรึกษา รศ.สมยศ จุณณะปิยะ

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้ เป็นการออกแบบ และการสร้างระบบตรวจสอบเวลาเข้า - ออกสำนักงาน โดยใช้บัตรแม่เหล็ก ซึ่งประกอบด้วยส่วนประมวลผล 3 ส่วน ได้แก่ ส่วนกลาง ส่วนย่อย และคอมพิวเตอร์ส่วนบุคคล โดยส่วนกลางทำหน้าที่ควบคุม และติดต่อรับส่งข้อมูลอนุกรมกับส่วนย่อย ผ่านโครงข่าย RS-485 และติดต่อรับส่งข้อมูลอนุกรมกับคอมพิวเตอร์ส่วนบุคคล ผ่านโครงข่าย RS-232-C สำหรับหน่วยย่อยแต่ละหน่วย ทำหน้าที่อ่านบัตรแม่เหล็กที่นำมารูด และคอมพิวเตอร์ส่วนบุคคลจะทำการบันทึกภาพ และตรวจสอบข้อมูล หากข้อมูลถูกต้อง รายละเอียดการมาทำงานของเจ้าของบัตรจะถูกส่งไปแสดงผลและทำการปลดล็อคประตู

ABSTRACT

This thesis presents a design of the Access Checking System using magnetic card. This system comprises of center, terminal processor unit and a personal computer. The center module is used to control, communicate with terminal via RS-485 and communicate with a personal computer via RS-232-C. While data from a magnetic card is read at a terminal by a magnetic card reader. personal computer saves image file in memory and the assigned code of the system will be unlocked the door.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทที่ 1 บทนำ	1
บทที่ 2 ทฤษฎีและหลักการ	3
2.1 หลักการทั่วไปของบัตรแม่เหล็ก	3
2.1.1 การบันทึกข้อมูลของวงจรแม่เหล็ก	4
2.1.2 การบันทึกข้อมูลลงบนบัตรแม่เหล็ก	5
2.1.3 การอ่านข้อมูลจากบัตรแม่เหล็ก	9
2.1.4 มาตรฐานบัตร ATM มาตรฐาน ISO 7811/2 – 1985 (E)	15
2.2 จอแสดงผลแบบผลึกเหลว (Liquid Crystal Display : LCD)	18
2.2.1 คุณสมบัติต่าง ๆ ของ HD44780	20
2.3 การรับส่งข้อมูล (Data Transmission)	21
2.3.1 การสื่อสารข้อมูลแบบอนุกรม	23
2.4 RS-232 มาตรฐานเพื่อการสื่อสาร	24
2.5 Real Time Clock : RTC (DS 1307)	31
2.5.1 คุณสมบัติ	31
2.5.2 รายละเอียดของขาต่อใช้งาน	32
2.5.3 การทำงาน	32
2.5.4 RTC and Address Map	33
2.5.5 DS 1307 Timekeeping Register	33
2.5.6 การเขียนข้อมูลลง DS 1307 (Data Write)	34
2.5.7 การอ่านข้อมูลจาก DS 1307 (Data Read)	35
2.5.8 I^2C	35
บทที่ 3 การคำนวณและการสร้าง	37
3.1 แนวความคิดในการสร้างระบบตรวจสอบการเข้าออกสำนักงาน	37
3.2 ส่วนประกอบของระบบ	37
3.3 การทำงานของระบบ	38
3.3.1 การทำงานของส่วนประมวลผล	38
3.3.2 การตั้งเวลาจริง และการอ่านค่าเวลาจริง	40
บทที่ 4 การทดลองและผลการทดลอง	42
4.1 การทดลองเรื่องการแสดงผลของค่าเวลาจริง และข้อมูลของบัตรแม่เหล็กผ่านจอแสดงผล LCD	42
4.2 การทดลองเรื่องโปรแกรมส่วนฐานข้อมูลในการบันทึกเวลาเข้าออกสำนักงาน	45

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

เรื่อง	หน้า
4.3 การทดลองเรื่องการตรวจสอบข้อมูลของผู้ดูแลและควบคุมระบบ	48
4.4 การทดลองเรื่องการทำงานของกล้องดิจิตอล	52
บทที่ 5 บทวิจารณ์และบทสรุป	54
ภาคผนวก	
กิตติกรรมประกาศ	
หนังสืออ้างอิง	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

รูป	หน้า
รูปที่ 2.1 B – H Curve	4
รูปที่ 2.2 การเปลี่ยนแปลงเองจากขนาดของ H	5
รูปที่ 2.3 การบันทึกข้อมูล	6
รูปที่ 2.4 แสดงรูปแบบของการบันทึกโดยใช้สัญญาณความถี่ 2 ความถี่ที่มีเฟสตรงกัน	7
รูปที่ 2.5 แสดงรูปวงจรพื้นฐานสำหรับการบันทึกข้อมูลบนบัตรแม่เหล็ก	8
รูปที่ 2.6 แสดงรูปสัญญาณการทำงานของวงจรพื้นฐาน สำหรับการบันทึกข้อมูลบนบัตรแม่เหล็ก	8
รูปที่ 2.7 กราฟแสดงคุณสมบัติการอิ่มตัวของแถบแม่เหล็ก บนบัตรแม่เหล็กมาตรฐาน	9
รูปที่ 2.8 แสดงสัญญาณแรงดันไฟฟ้าจากหัวอ่าน และถูกขยายด้วยอัตราขยายประมาณ 100 เท่า	9
รูปที่ 2.9 แสดงส่วนประกอบของวงจรพื้นฐานสร้างสัญญาณรูปแบบ ISO จากบัตรแม่เหล็ก	10
รูปที่ 2.10 แสดงวงจรขยายแรงดันไฟฟ้าสำหรับอ่านข้อมูลจากบัตรแม่เหล็ก	11
รูปที่ 2.11 แสดงการเปรียบเทียบสัญญาณและลักษณะสัญญาณที่ต้องการ	11
รูปที่ 2.12 แสดงการเลื่อนเฟสและวงจรเปรียบเทียบแรงดันไฟฟ้า	12
รูปที่ 2.13 แสดงรูปวงจรดิฟเฟอร์เรนทิเอเตอร์และวงจรเปรียบเทียบแรงดันไฟฟ้า	12
รูปที่ 2.14 รูปแบบของข้อมูลที่อยู่ในแทร็กที่ 2	15
รูปที่ 2.15 ตำแหน่งแทร็กที่ 2 ของบัตรแม่เหล็ก	16
รูปที่ 2.16 Timing Diagram	16
รูปที่ 2.17 ข้อมูลที่อยู่ในแทร็กที่ 2 แสดงลักษณะของข้อมูลที่ประกอบด้วย Start Character และ Character	17
รูปที่ 2.18 แสดงโครงสร้างของ Liquid Crystal Display	19
รูปที่ 2.19 แสดง LCD Module	19
รูปที่ 2.20 ตัวอย่าง LCD Module ที่ใช้ Chip H0550 และ H0551	20
รูปที่ 2.21 การกำหนดเส้นทางสัญญาณได้	21
รูปที่ 2.22 การกำหนดเส้นทางสัญญาณไม่ได้	21
รูปที่ 2.23 แบบทิศทางเดียว	22
รูปที่ 2.24 แบบกึ่งทางคู่	22
รูปที่ 2.25 แบบทางคู่	23
รูปที่ 2.26 คอนเน็กเตอร์แบบ DB-25 และ DB-9	30
รูปที่ 2.27 แสดงถึงส่วนประกอบหลักของ DS1307	32
รูปที่ 2.28 แสดงตำแหน่งของนาฬิกาและ RAM ของ DS 1307	33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ (ต่อ)

รูป	หน้า
รูปที่ 2.29 แสดง Diagram การเขียนข้อมูลลง DS 1307	34
รูปที่ 2.30 แสดง Diagram การอ่านข้อมูลออกจาก DS 1307	35
รูปที่ 2.31 แสดง Timing Diagram ของ Bit Transfer	35
รูปที่ 2.32 แสดง Timing Diagram ของ Start และ Stop Condition	36
รูปที่ 2.33 แสดง Timing Diagram ของ Acknowledge	36
รูปที่ 3.1 บล็อกไดอะแกรมแสดงระบบตรวจสอบการเข้าออกสำนักงาน	37
รูปที่ 3.2 โฟลชาร์ตการทำงานของส่วนประมวลผล	38
รูปที่ 3.3 แสดงโฟลชาร์ตการตั้งค่าเวลาจริง และการอ่านค่าเวลาจริง	40
รูปที่ 4.1 แสดงการเชื่อมต่อบอร์ด SPI-RD 2 กับคอมพิวเตอร์	42
รูปที่ 4.2 แสดงผลข้อมูลของบัตรแม่เหล็กทางพอร์ตอนุกรม	43
รูปที่ 4.3 แสดงสัญญาณข้อมูลและสัญญาณนาฬิกาของ DS1307	43
รูปที่ 4.4 แสดงหน้าจอ เมื่อทดลองรันโปรแกรม MAIN.VBP	46
รูปที่ 4.5 แสดงข้อมูลเมื่อมีพนักงานรูดบัตรเข้าทำงาน	46
รูปที่ 4.6 แสดงข้อมูลเมื่อรูดบัตรพนักงานอีกคนหนึ่ง	47
รูปที่ 4.7 แสดงข้อมูลเมื่อพนักงานรูดบัตรออกจากงาน	47
รูปที่ 4.8 ค้นหาข้อมูลของพนักงานโดยใช้รหัสพนักงาน	49
รูปที่ 4.9 ค้นหาข้อมูลของพนักงาน โดยใช้ชื่อของพนักงาน	49
รูปที่ 4.10 แสดงการแก้ไขข้อมูล	50
รูปที่ 4.11 แสดงการเพิ่มข้อมูลของพนักงาน	50
รูปที่ 4.12 แสดงการลบข้อมูลของพนักงาน	51
รูปที่ 4.13 แสดงการถ่ายภาพของกล้องดิจิตอล	53

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตาราง	หน้า
ตารางที่ 2.1 ตัวอย่างการใช้งานบัตรแม่เหล็ก	3
ตารางที่ 2.2 ชนิดและการใช้งานบัตรแม่เหล็ก	3
ตารางที่ 2.3 แสดงข้อมูลของรหัส BCD สำหรับเครื่องอ่านแถบแม่เหล็ก	15
ตารางที่ 2.4 รหัสสำหรับแทร็กที่ 2	18
ตารางที่ 2.5 แสดงรายการคุณลักษณะเฉพาะสำหรับการอินเตอร์เฟสแบบ RS-232	25
ตารางที่ 2.6 แสดงรายละเอียดสำหรับการอินเตอร์เฟส RS-232 โดยใช้คอนเน็กเตอร์ แบบ DB-25	26
ตารางที่ 2.7 แสดงข่าวสัญญาณที่ต่อจาก DTE ไปยัง DCE โดยใช้คอนเน็กเตอร์ แบบ DB-25	28
ตารางที่ 2.8 การต่อคอนเน็กเตอร์แบบ DB-9 ตามมาตรฐาน RS-232	29
ตารางที่ 2.9 แสดงตำแหน่งขาสัญญาณของพอร์ตอนุกรม	30
ตารางที่ 2.10 แสดงตำแหน่งมาตรฐานของพอร์ตอนุกรม	31
ตารางที่ 2.11 รายละเอียดของขาสัญญาณ	32

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1. วัตถุประสงค์

1.1 เพื่อศึกษาการทำงานของเครื่องอ่านแถบแม่เหล็ก

1.2 เพื่อศึกษาการทำงานของไมโครคอนโทรลเลอร์

1.3 เพื่อศึกษาการเขียนโปรแกรม ด้วยโปรแกรม Visual Basic

1.4 สามารถนำเครื่องอ่านแถบแม่เหล็กมาใช้งานร่วมกับไมโครคอนโทรลเลอร์ และโปรแกรม Visual Basic ได้

2. ความเป็นมาของโครงงาน

ในปัจจุบันวิวัฒนาการทางเทคโนโลยีได้พัฒนาไปอย่างรวดเร็ว เทคโนโลยีใหม่ๆ ได้ถูกนำมาใช้ในชีวิตประจำวันมากขึ้น แต่บัตรแถบแม่เหล็ก ซึ่งเป็นเทคโนโลยีที่ใช้กันมาหลายปีแล้ว ก็ยังคงถูกนำมาใช้งานอย่างต่อเนื่อง ไม่ว่าจะเป็น บัตรเอทีเอ็ม (ATM) บัตรเครดิต บัตรเช็คเวลา เข้า—ออก พนักงาน หรือบัตรประจำตัวประชาชน ก็ใช้บัตรแถบแม่เหล็กทั้งหมด

ดังนั้น การที่จะนำบัตรแถบแม่เหล็กมาใช้งาน จึงควรที่จะศึกษาการทำงานของเครื่องอ่านบัตรแถบแม่เหล็กให้เข้าใจเสียก่อน อีกทั้งยังต้องศึกษาการทำงานของไมโครคอนโทรลเลอร์ โปรแกรม Visual Basic โปรแกรม Microsoft Access และกล้องวีดีโอดิจิตอล เพื่อที่จะสามารถนำมาประยุกต์ใช้งานร่วมกันในโครงงาน "ระบบตรวจสอบการเข้าออกสำนักงาน" โดยใช้บัตรแถบแม่เหล็ก

3. ขอบเขตของโครงงาน

3.1 ส่วนของบัตร ใช้บัตรแถบแม่เหล็ก

3.2 ส่วนของการอ่านข้อมูล ใช้เครื่องอ่านแถบแม่เหล็ก

3.3 ส่วนควบคุม ใช้ไมโครคอนโทรลเลอร์

3.4 ส่วนแสดงผล ใช้จอแอลซีดี (LCD) และมอนิเตอร์ ของคอมพิวเตอร์

3.5 ส่วนของฐานข้อมูลใช้โปรแกรม Microsoft Access

4. ขั้นตอนและวิธีการดำเนินการ

4.1 ศึกษาข้อมูลในส่วนที่เกี่ยวข้องกับโครงงาน

4.2 ศึกษาการทำงานของเครื่องอ่านบัตรแถบแม่เหล็ก

4.3 ศึกษาการเขียนโปรแกรม Visual Basic

4.4 ศึกษาการทำงานของโปรแกรม Microsoft Access

4.5 จัดเตรียมอุปกรณ์ที่จะต้องใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.6 ศึกษาการทำงานของชุดควบคุม ซึ่งใช้บอร์ด SPI-RD2

4.7 ออกแบบการทำงานของชุดควบคุม

4.8 ออกแบบการทำงานของโปรแกรม Visual Basic ให้ทำงานร่วมกับ CPU (P89C51RD2)

4.9 ทดลองผลการทำงานของชุดควบคุม และโปรแกรม Visual Basic

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีหรือหลักการ

2.1 หลักการทั่วไปของบัตรแม่เหล็ก

บัตรแม่เหล็กที่ใช้กันอยู่ในปัจจุบันมีอยู่ด้วยกันมากมายหลายชนิด ในยุคแรก ๆ บัตรแม่เหล็กจะมีเนื้อวัสดุเป็นพลาสติกพีวีซี (PVC) แข็ง และแบบที่มีเนื้อวัสดุเป็นกระดาษ แต่ในปัจจุบันได้พัฒนาเนื้อวัสดุของบัตรแม่เหล็กมาเป็น Polyethylene Therephthalate หรือเรียกย่อๆ ว่า PET

ชนิด	เนื้อวัสดุ	ความหนาของบัตร	การใช้งาน
บัตรพลาสติก (Plastic Card)	- PVC ชนิดแข็ง - PVCA ชนิดแข็ง	0.76 ม.ม.	- บัตรเอทีเอ็ม (ATM) - บัตรเครดิต - บัตรประจำตัว (ID)
PET Card	- Polyethylene Therephthalate (PET)	0.2 — 0.4 ม.ม.	- บัตรโทรศัพท์ - บัตรช็อปปิ้งต่างๆ (prepaid card)
บัตรกระดาษ (Composite paper Card)	- กระดาษอย่างดี - กระดาษเคลือบพลาสติก - กระดาษบันทึกข้อมูล	0.2 — 0.76 ม.ม.	- ตั๋วรถไฟ - ตั๋วทางด่วน - ตั๋วเครื่องบิน - บัตรโรงแรม

ตารางที่ 2.1 ตัวอย่างการใช้งานบัตรแม่เหล็ก

การใช้งาน	ชนิดของบัตร
การเงิน การธนาคาร	- บัตรเอทีเอ็ม - บัตรเครดิต - บัตรเติมน้ำมัน - บัตรหลักทรัพย์ - บัตรช็อปปิ้ง
การคมนาคม	- ตั๋วรถไฟแม่เหล็ก - บัตรผ่านทางด่วน - ตั๋วเครื่องบิน - ตั๋วจอดรถ
การสื่อสาร	- บัตรโทรศัพท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

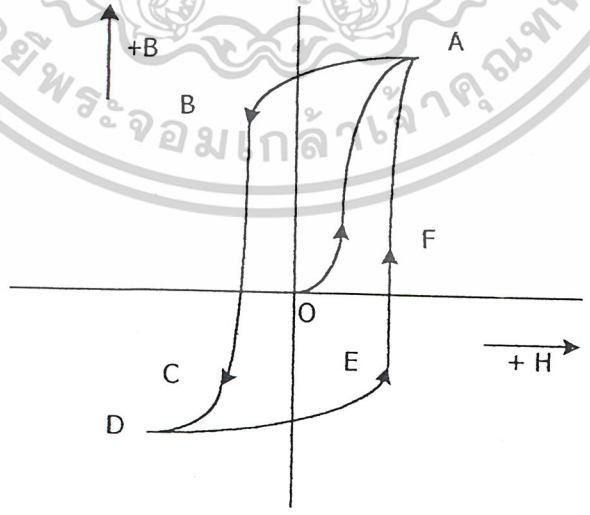
การใช้งาน	ชนิดของบัตร
การสำนักงาน	- บัตรประจำตัวพนักงาน - บัตรประจำตัวนักเรียน - บัตรตรวจโรค - บัตรโรงแรม
การรักษาความปลอดภัย	- บัตร ปิด - เปิด ล็อค

ตารางที่ 2.2 ชนิดและการใช้งานบัตรแม่เหล็ก

2.1.1 คุณลักษณะของวงจรแม่เหล็ก

คุณลักษณะของวงจรแม่เหล็กสามารถแสดงได้ด้วยกราฟดัง รูปที่ 2.2 โดยการพล็อตความสัมพันธ์ระหว่าง "ฟลั๊กซ์แม่เหล็ก (Flux density ; B)" และ "สนามแม่เหล็ก (Magnetizing force; H)" ทั้งนี้ความสัมพันธ์ของ B-H เป็นความสัมพันธ์ระหว่างแรงดันกับกระแสฟลั๊กซ์แม่เหล็ก หมายถึงจำนวนเส้นแรงแม่เหล็กต่อตารางนิ้วที่ผ่านพื้นที่หน้าตัดของส่วนแกน ส่วนสนามแม่เหล็ก (Magnetizing force) นั้นเท่ากับจำนวนแอมแปร์-รอบ (ampere-turn) ต่อนิ้วของความยาวแกนในการทำงาน หากสนามแม่เหล็ก (H) มีค่าเพิ่มขึ้นทีละน้อยจากศูนย์ ฟลั๊กซ์แม่เหล็ก (H) จะค่อย ๆ เพิ่มขึ้นเป็นสัดส่วนด้วย (0 – a – b) จนกระทั่งถึงจุดอิ่มตัวที่ b จากนั้นถ้าแรงดันตกคร่อมขดลวดตกลงเป็นศูนย์ฟลั๊กซ์แม่เหล็ก B จะไม่กลับไปที่ศูนย์ แต่มันจะลดลงตามเส้นโค้งจาก b – c และถึงจุด c เมื่อ H กลายเป็นศูนย์โดยแมกนีจูดหรือขนาด 0 – c

เราอาจจะกล่าวได้ว่า Magnetic substance จะมี Residual magnetism ภายหลังจากการป้อนสนามแม่เหล็ก (Magnetizing force) ให้กับมัน



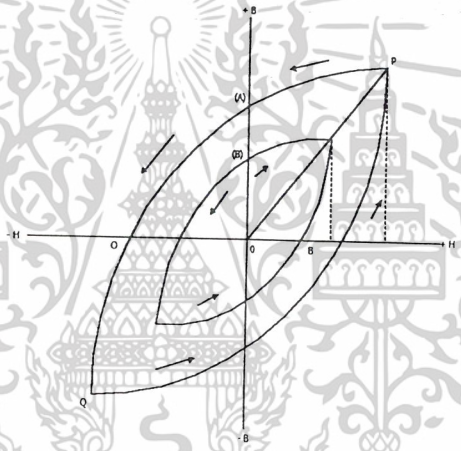
รูปที่ 2.1 B – H Curve

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าหากลองป้อนกระแสหรือ H ในทิศทางตรงกันข้ามกับขดลวด ค่า B จะเป็นไปตามโค้งจาก c – d – e ที่จุด e เป็นจุดที่มีการอิ่มตัวตรงข้ามกันของวัสดุแม่เหล็ก จะเห็นได้ว่าหากเราป้อนแรงดันไฟฟ้ากระแสสลับเข้าที่ขดลวด ค่า B จะเดินทางเกิดเป็นวงรอบ หรือลูป (Loop) ขึ้น คือจาก 0 – a – b – c – d – e – f – g – b เราเรียกลูปดังกล่าวว่า **"ฮีสเตอร์รีซีส"**

- ระยะระหว่างจุดตัดแกน 0 – b เรียกว่า "magnetization curve" ของวัสดุ
- Closed loop เรียกว่า "hysteresis loop"
- ค่าของ Flus density (H) จาก 0 – c เรียกว่า "retentivity ของ magnetic substance"
- ค่าของ Magnetizing curve (H) จาก 0 – d เรียกว่า "coereive force"

จากรูปที่ 2.1 นั้นหากปริมาณหรือขนาดของ H (ซึ่งป้อนเข้าไปครั้งแรก) หยุดที่จุดที่มีค่าต่ำ ๆ จุดหนึ่งแล้วกลับจะมีผลทำให้เกิดเป็นลูปเล็ก ๆ (เล็กกว่าครั้งแรก) ขึ้นดังแสดงในรูปที่ 2.2



รูปที่ 2.2 การเปลี่ยนแปลงเนื่องจากขนาดของ H

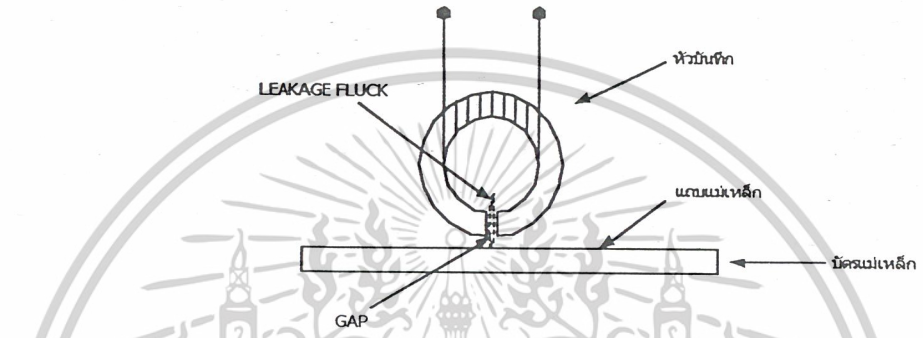
ในกรณีนี้หากป้อนแรงแม่เหล็กเข้าที่ตำแหน่ง α แล้วแรงแม่เหล็ก (A) จะคงอยู่ แม้เอาแรงแม่เหล็กออกแล้วก็ตาม และหากป้อนสนามแม่เหล็ก β เข้าไป (B) ก็จะคงอยู่ด้วยเช่นกัน อันนี้จึงเป็นการอธิบายถึงหลักการจำ (Memory) ของการบันทึกข้อมูลด้วย Hyteresis Ohenomenon "B-H Characteristic" อักนัยหนึ่งก็คือ ฟังก์ชันความจำ (Memory Function) ของการบันทึกนั้นก็คือ เปลี่ยนแปลงรูปคลื่นของข้อมูลที่จะบันทึกให้อยู่ในรูปของฟลั๊กซ์แม่เหล็ก ในกรณีนี้จะกล่าวในเรื่องเครื่องบันทึกเทปทั่วๆ ไป ซึ่งนำมาเป็นแนวทางสำหรับเครื่องอ่านและบันทึกแถบแม่เหล็กเท่านั้น

2.1.2 การบันทึกข้อมูลลงบนบัตรแม่เหล็ก

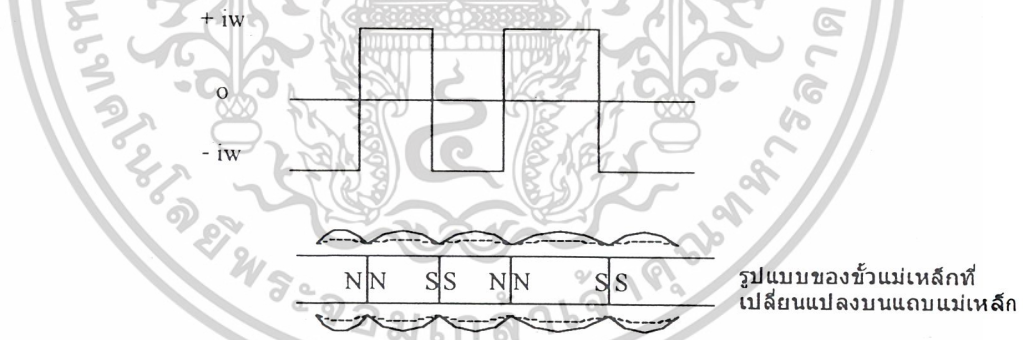
การบันทึกข้อมูลลงบนบัตรแม่เหล็กจะใช้วิธีการบันทึกแบบดิจิตอล ในลักษณะเช่นเดียวกับที่ใช้ในแผ่นฟล็อปปี้ดิสก์ หรือ เทปแม่เหล็กสำหรับเครื่องคอมพิวเตอร์ทั่ว ๆ ไป การบันทึกข้อมูลลงบนแถบแม่เหล็กนั้น จะต้องป้อนกระแสพัลส์ซึ่งมีทั้งด้าน + และ ด้าน – พร้อมทั้งมีขนาดเพียงพอเข้าที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขดลวดของหัวบันทึกซึ่งกดอยู่บนแถบแม่เหล็กที่เคลื่อนด้วยความเร็วคงที่ ดังแสดงในรูปที่ 2.3 (a) แถบแม่เหล็กจะถูกเปลี่ยนให้มีรูปแบบของขั้วแม่เหล็กตาม Leakage Flux จาก Gap ของหัวบันทึกดังในรูปที่ 2.3 (b) แถบแม่เหล็กจะเกิดเป็นแม่เหล็กถาวรขนาดเล็กเรียงตัวกันตามขั้ว + หรือ — ของพัลส์และความกว้างของพัลส์สัญญาณที่บันทึกเนื่องจากกระแสพัลส์ที่ใช้ในการบันทึกมีขนาดเพียงพอที่หัวบันทึกจะทำให้แถบ แม่เหล็กมีสนามแม่เหล็กอิ่มตัวได้ดังนั้น เมื่อทำการบันทึกข้อมูล ตัวข้อมูลที่เคยมีอยู่จะถูกเขียนทับและหายไปเหลือเพียงข้อมูลใหม่เท่านั้น



(a) การบันทึกข้อมูลลงบนบัตรแม่เหล็ก

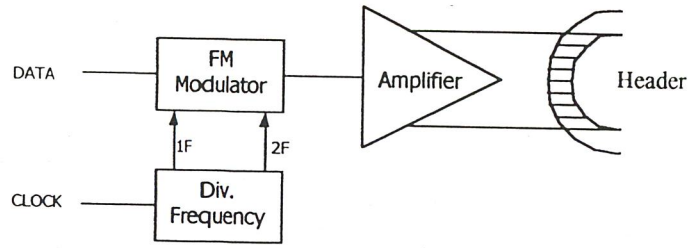


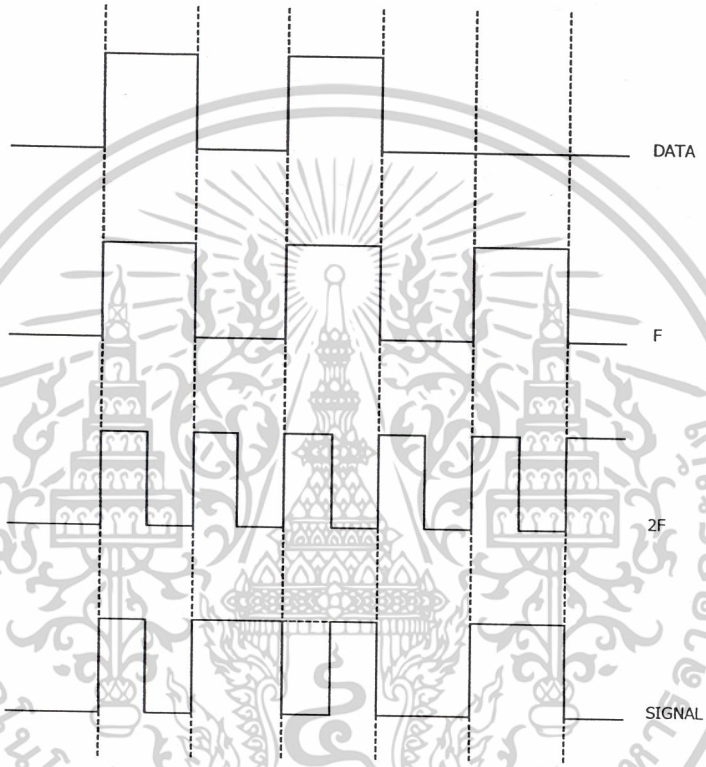
(b) รูปแบบของขั้วแม่เหล็กที่บันทึก

รูปที่ 2.3 การบันทึกข้อมูล

รูปแบบการบันทึกข้อมูลลงบัตรแม่เหล็กส่วนใหญ่ จะใช้รูปแบบ F2F หมายถึง Two Frequency Coherent Encoding และ FM หมายถึง Frequency Modulation ซึ่งมีเฟส (Phase) ตรงกันมาใช้ในการบันทึกข้อมูล ดังแสดงไว้ในรูปที่ 2.4 ซึ่งการบันทึกข้อมูลในลักษณะเช่นนี้ จะบันทึกข้อมูล (Data) และสัญญาณนาฬิกา (Clock) เข้าไว้ในแทร็ค (Track) เดียวกัน ซึ่งคิดค้นโดย Aiken (1954)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



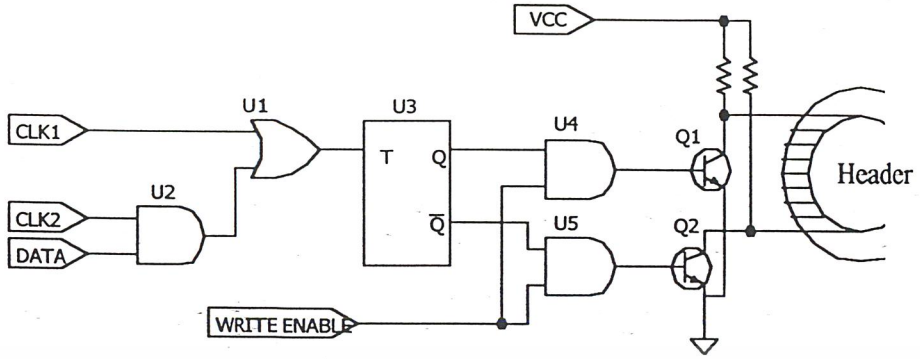


รูปที่ 2.4 แสดงรูปแบบของการบันทึกโดยใช้สัญญาณความถี่สองความถี่ที่มีเฟสตรงกัน

จากรูปที่ 2.4 สัญญาณบันทึกจะประกอบด้วยบิตของข้อมูลที่บันทึก และบิตของสัญญาณนาฬิกา (2F) สำหรับการเปลี่ยนแปลงของฟลั๊กซ์แม่เหล็กของสัญญาณบันทึก จะเกิดขึ้นในระหว่างที่สัญญาณข้อมูลมีสภาวะ "1" ทางตรรก (Logic) หรือระหว่างสัญญาณบิตของข้อมูลติดกัน และจะไม่มีการเปลี่ยนแปลงของฟลั๊กซ์แม่เหล็กในระหว่างที่สัญญาณข้อมูลมีสภาวะ "0" ทางตรรก สำหรับการบันทึกบิตข้อมูลของแต่ละตัวเลขหรือตัวอักษรลงบนบัตรแม่เหล็ก จะเป็นการบันทึกข้อมูลที่นัยสำคัญต่ำสุด (b1) ก่อน และบิตภาวะเสมอมูล (p) เป็นบิตสุดท้ายที่ถูกบันทึก ส่วนค่าความหนาแน่นในการบันทึกข้อมูลในแทร็คที่สอง ISO ได้กำหนดให้บันทึกด้วยความหนาแน่น 75 $\pm$ 20 บิตต่อนิ้ว

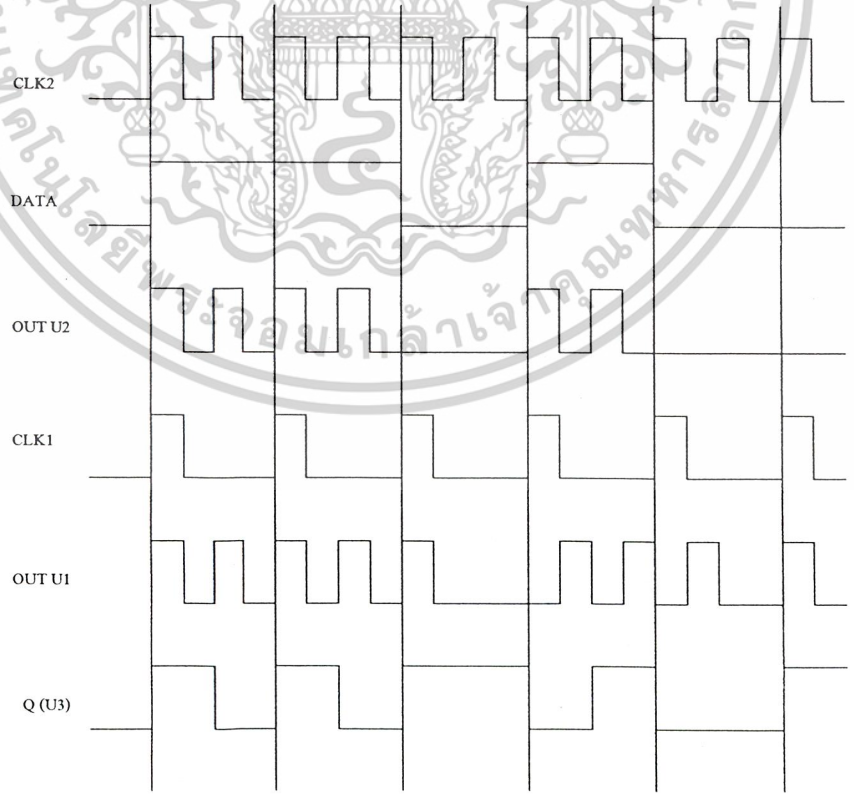
ส่วนวงจรพื้นฐานสำหรับการบันทึกข้อมูลลงบนบัตรแม่เหล็กนั้นได้แสดงไว้ในรูปที่ 2.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 แสดงรูปวงจรพื้นฐาน สำหรับการบันทึกข้อมูลบนบัตรแม่เหล็ก

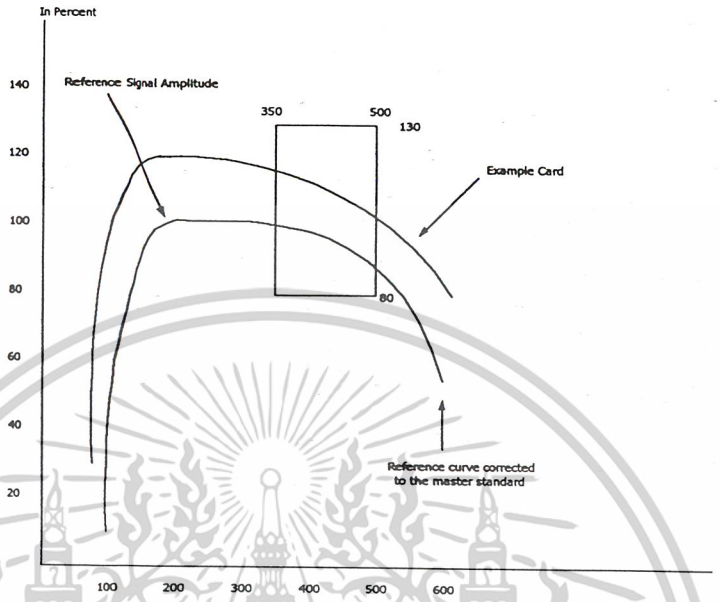
จากรูปที่ 2.5 ไอซี U1, U2, U3, U4 และ U5 จัดรูปวงจรมอดูเลต (Modulation Circuit) สัญญาณข้อมูล และสัญญาณนาฬิกา ให้อยู่ในรูปแบบสัญญาณเอฟเอ็ม และป้อนสัญญาณดังกล่าว ให้วงจรจ่ายกระแสไฟฟ้า ให้กับขดลวดที่พันรอบแกนหัวบันทึก เมื่อทำการบันทึกข้อมูลสัญญาณ Write Enable มีตรรกเป็น "1" พร้อมกับทำให้ทรานซิสเตอร์ (Transistor) Q1 และ Q2 ทำงาน ซึ่งทรานซิสเตอร์ทั้งคู่จะสลับการทำงานเมื่อป้อนกระแสพัลส์ด้านบวกและด้านลบให้กับขดลวดตามสัญญาณบันทึก โดยรูปสัญญาณของวงจรพื้นฐานนี้แสดงดังรูปที่ 2.6



รูปที่ 2.6 แสดงรูปสัญญาณการทำงาน ของวงจรพื้นฐานสำหรับการบันทึกข้อมูลบนบัตรแม่เหล็ก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

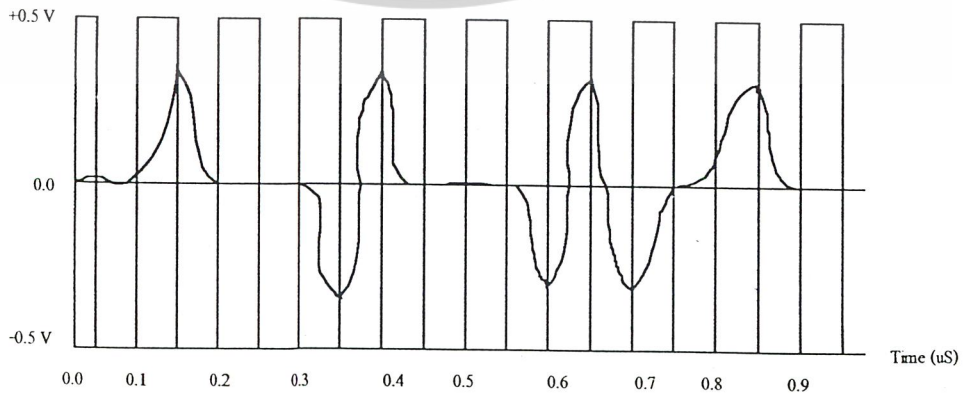
สำหรับค่าของกระแสพัลส์ที่ใช้ในการบันทึกข้อมูล สามารถหาได้จากกราฟแสดงคุณสมบัติการอิ่มตัวของแม่เหล็กบนบัตรแม่เหล็ก ดังแสดงในรูปที่ 2.7



รูปที่ 2.7 กราฟแสดงคุณสมบัติการอิ่มตัว ของแถบแม่เหล็กบนบัตรแม่เหล็กมาตรฐาน

2.1.3 การอ่านข้อมูลจากบัตรแม่เหล็ก

การอ่านข้อมูลจากบัตรแม่เหล็ก สามารถทำได้โดยให้แถบแม่เหล็กสัมผัสกับหัวอ่าน ซึ่งเคลื่อนที่ด้วยความเร็วคงที่ ฟลั๊กซ์แม่เหล็กที่เกิดขึ้นจะผ่านจากช่องว่างของแกนหัวอ่าน ไปยังขดลวดที่พันรอบแกนอยู่ ซึ่งการเปลี่ยนแปลงของฟลั๊กซ์แม่เหล็กตามข้อมูลที่บันทึกจะทำให้เกิดแรงดันไฟฟ้าที่ขดลวดหัวอ่านตามข้อมูลนั้น ดังแสดงในรูปที่ 2.8 ซึ่งจุดสูงสุดของแรงดันที่อ่านได้นั้น จะตรงกับจุดที่สนามแม่เหล็ก บนแถบแม่เหล็กกลับทิศทางพอดี ดังนั้นถ้าขยายแรงดันนี้ขึ้น และตรวจหาจุดสูงสุดด้วยวิธี ดิฟเฟอเรนเชียล (Differential) แล้วเปลี่ยนเป็นสัญญาณพัลส์ก็จะได้ข้อมูลที่บันทึกอยู่ในบัตรแม่เหล็ก

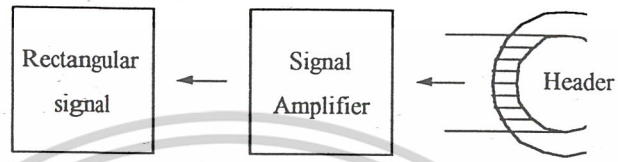


รูปที่ 2.8 แสดงสัญญาณแรงดันไฟฟ้าจากหัวอ่าน และถูกขยายด้วยอัตราขยายประมาณ 100 เท่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.8 ตำแหน่งสูงสุดของแรงดันไฟฟ้าที่อ่านได้จะตรงกับตำแหน่งที่สนามแม่เหล็กบนบัตรแม่เหล็กมีการกลับทิศทาง ทำให้สามารถนำตำแหน่งนี้มาสร้างสัญญาณพัลส์ เพื่ออ่านข้อมูลลงบนบัตรแม่เหล็กได้

วงจรพื้นฐาน สำหรับสร้างสัญญาณรูปแบบ ISO จากบัตรแม่เหล็กนั้น แสดงดังในรูปที่ 2.9 โดยมีส่วนประกอบดังนี้



รูปที่ 2.9 แสดงส่วนประกอบของวงจรพื้นฐานสร้างสัญญาณรูปแบบ ISO จากบัตรแม่เหล็ก

1.) **วงจรขยายสัญญาณแรงดันไฟฟ้า**

สัญญาณแรงดันไฟฟ้าจากหัวอ่านข้อมูลมีค่าค่อนข้างต่ำ (5 — 15 มิลลิวัตต์) จึงจำเป็นต้องขยาย

สัญญาณเพื่อความสะดวกต่อการนำไปสร้างสัญญาณรูปเหลี่ยม โดยใช้วงจรขยายสัญญาณรวม มีค่าประมาณ 100 เท่า (ไม่ให้สัญญาณอิ่มตัว) ดังแสดงในรูปที่ 2.10

ก) วงจรขยายสัญญาณความแตกต่าง (Differential Amplifier)

เป็นวงจรขยายสัญญาณชุดแรกที่รับสัญญาณข้อมูลจากขั้วทั้งสอง ของหัวอ่านข้อมูลซึ่งมีค่าความต้านทานต่ำ และใช้ IC A_1 ทำหน้าที่ขยายสัญญาณความแตกต่าง มีอัตราขยาย

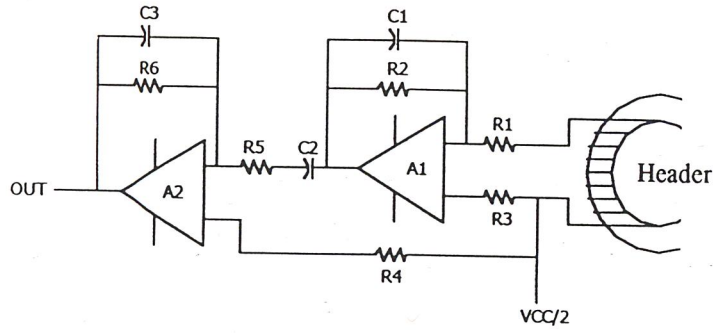
$$AV = -R_2/R_1$$

ข) วงจรขยายสัญญาณแบบกลับเฟส (Inverting Amplifier)

เพื่อให้ได้สัญญาณข้อมูลที่มีค่าแรงดันไฟฟ้าสูงพอ ที่จะป้อนให้กับวงจรสร้างสัญญาณรูปเหลี่ยมจึงนำข้อมูลสัญญาณมาผ่านวงจรแบบกลับเฟส ซึ่งอัตราขยายมีค่าเท่ากับ

$$AV = -R_6/R_5$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



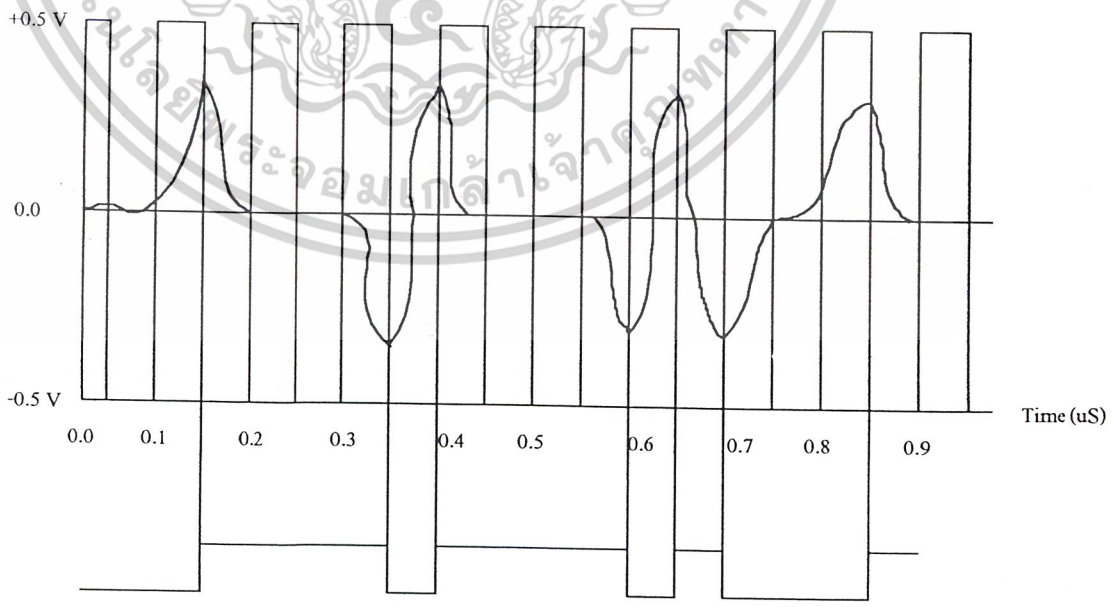
รูปที่ 2.10 แสดงวงจรขยายแรงดันไฟฟ้าสำหรับอ่านข้อมูลจากบัตรแม่เหล็ก

2.) วงจรสร้างสัญญาณรูปเหลี่ยม (รูปแบบ ISO)

จากสัญญาณข้อมูลที่ขยายแล้ว (จากรูปที่ 2.8) ทำให้สามารถพิจารณารูปแบบการสร้างสัญญาณรูปเหลี่ยมได้หลายรูปแบบ โดยเฉพาะการนำตำแหน่งจุดยอดของสัญญาณข้อมูลทั้งด้านบนและด้านล่างมาใช้ในการสร้างสัญญาณรูปเหลี่ยม จะทำให้ได้สัญญาณพัลส์ที่เหมือนกับสัญญาณที่ใช้ในการบันทึกข้อมูลบนบัตรแม่เหล็ก สำหรับการสร้างสัญญาณรูปเหลี่ยมสามารถสร้างได้ 2 วิธี

ก) การเลื่อนเฟสและการเปรียบเทียบแรงดันไฟฟ้า

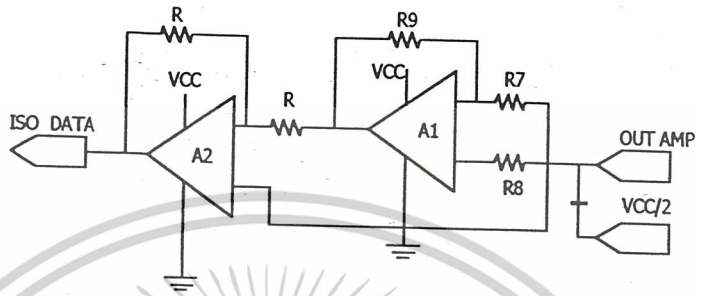
ใช้หลักการหน่วงสัญญาณข้อมูลในช่วงเวลาสั้นมาก ๆ แล้วนำสัญญาณข้อมูลจริง (เส้นทึบ) เปรียบเทียบกับสัญญาณข้อมูลที่ถูกหน่วง (เส้นประ) ซึ่งสัญญาณเอาท์พุตจากการเปรียบเทียบนี้แสดงไว้ในรูปที่ 2.11



รูปที่ 2.11 แสดงการเปรียบเทียบสัญญาณ และลักษณะสัญญาณที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

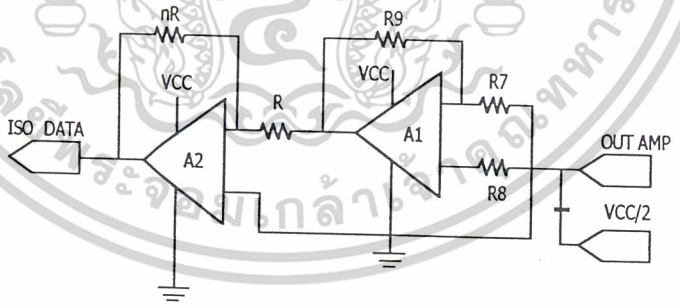
จากรูปที่ 2.11 วงจรที่ใช้งานจะใช้ออปแอมป์ (op amp) สองตัว โดยที่ออปแอมป์ตัวแรกจัดรูปวงจรเป็นวงจรเลื่อนเฟส เพื่อให้ได้สัญญาณข้อมูลที่ถูกหน่วง (เอาท์พุตที่เป็นเส้นประ) และออปแอมป์ตัวที่สอง จัดรูปวงจรเป็นวงจรเปรียบเทียบแรงดันระหว่างสัญญาณข้อมูลกับสัญญาณที่ถูกหน่วง ดังแสดงในรูปที่ 2.12



รูปที่ 2.12 แสดงวงจรเลื่อนเฟส และวงจรเปรียบเทียบสัญญาณแรงดันไฟฟ้า

ข) วงจรดิฟเฟอร์เรนทิเอเตอร์และวงจรเปรียบเทียบแรงดันไฟฟ้า

แนวทางที่สอง จะใช้วงจรดิฟเฟอร์เรนทิเอเตอร์ซึ่งมีไดโอด D1 และ D2 ทำตัวเหมือนตัวต้านทางป้อนกลับด้านลบของวงจร ทำงานร่วมกับวงจรเปรียบเทียบแรงดันไฟฟ้า ดังแสดงในรูปที่ 2.13



รูปที่ 2.13 แสดงรูปวงจรดิฟเฟอร์เรนทิเอเตอร์ และวงจรเปรียบเทียบแรงดันไฟฟ้า

จากรูปที่ 2.13 เมื่อพิจารณาสัญญาณข้อมูลจากหัวอ่านข้อมูล ซึ่งมีลักษณะของสัญญาณคล้ายรูปคลื่นสามเหลี่ยม เมื่อป้อนสัญญาณข้อมูลให้กับวงจรดิฟเฟอร์เรนทิเอเตอร์ สัญญาณเอาท์พุตที่ได้ จะเป็นสัดส่วนกับค่าอนุพันธ์ของสัญญาณเอาท์พุต เมื่อสัญญาณข้อมูลที่เป็นรูปสามเหลี่ยมที่มีค่า สโลบ (Slope)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นค่าบวกและค่าลบที่แกว่งบนแรงดันอ้างอิง (Vcc/2) สัญญาณเอาท์พุตของวงจรดิฟเฟอร์เรนทิเอเตอร์จะเป็นรูปคลื่นสี่เหลี่ยม มีค่าแรงดันเอาท์พุต ตามสมการที่ 2.1

$$Vo = -RCk_n \qquad (2.1)$$

เมื่อค่า k_n เป็นค่าสโลบของสัญญาณอินพุต (โวลต์ต่อวินาที) และใช้วงจรเปรียบเทียบสัญญาณแรงดันไฟฟ้าเป็นส่วนสร้างสัญญาณรูปเหลี่ยมให้ดียิ่งขึ้น

สำหรับวงจรเปรียบเทียบแรงดันไฟฟ้าของทั้งสองแนวทาง จะใช้ตัวต้านทาน R และ nR จัดรูปวงจรแบบป้อนกลับทางบวก เพื่อป้องกันให้วงจรเกิดการออสซิลเลท โดยมีค่าแรงดันขีดเริ่มเปลี่ยนด้านสูง (Upper Threshold Voltage : V_{UT}), ค่าแรงดันขีดเริ่มเปลี่ยนด้านต่ำ (Lower Threshold Voltage : V_{LT}) และค่าศักดาไฟฟ้าฮิสเตอริซิส (VT) ตามสมการที่ 2.2,2.3 และ 2.4 ตามลำดับ

$$V_{UT} = Vref.[1 + (1/n)] + (Vsat/n) \qquad (2.2)$$

$$V_{LT} = Vref.[1 + (1/n)] - (Vsat/n) \qquad (2.3)$$

$$V_T = Vref.[Vsat - Vsat]/n \qquad (2.4)$$

การใช้เครื่องอ่านแถบแม่เหล็กเพื่อให้อ่านข้อมูลจากบัตรแถบแม่เหล็กนั้น จะใช้วิธีการนำแผ่นบัตรส่วนที่เป็นแถบแม่เหล็กไปเคลื่อนที่ผ่านหัวอ่านของเครื่องอ่านไปตามทิศทางและตำแหน่งที่กำหนดไว้เท่านั้น โดยสังเกตตำแหน่งของหัวอ่านกับส่วนที่เป็นแถบแม่เหล็กของบัตรต้องอยู่ในแนวเดียวกัน และเนื่องจากเครื่องอ่านแถบแม่เหล็กสามารถอ่านข้อมูลจากบัตรแถบแม่เหล็กได้เฉพาะข้อมูลของ Track 2 เท่านั้น ดังนั้นจึงขอกล่าวถึงเฉพาะวิธีการอ่านข้อมูลจาก Track 2 เพียงอย่างเดียว โดย Format ของข้อมูลในแถบแม่เหล็กส่วนที่ 2 (Track 2) ซึ่งใช้บันทึกสัญญาณด้วยความหนาแน่นของสนามแม่เหล็ก 75 BPI ซึ่งจะบรรจุจำนวนรหัสของข้อมูลต่างๆ ใน Track นี้ ได้สูงสุดไม่เกิน 40 ตัวอักษร โดยนับรวมรหัสควบคุมและรหัสตรวจสอบด้วย ซึ่งมีรายละเอียดดังนี้ คือ

SS	PAN	FS	Addition Data	ES	LRC

- SS = รหัสข้อมูลแสดงการเริ่มต้นของข้อมูลในบัตร (Start Sentinel) มีรหัสข้อมูลเป็น 0BH (;)
- PAN = ข้อมูลแสดงหมายเลขของบัตร (มีจำนวนข้อมูลในส่วนนี้สูงสุดไม่เกิน 19 หลัก)
- FS = รหัสข้อมูลแสดงการแบ่งแยกข้อมูล (Field Separator) มีรหัสข้อมูลเป็น 0DH (=)
- Additional Data = เป็นข้อมูลเพิ่มเติมอื่นๆ ของบัตร เช่น เดือน/ปี ที่ออกบัตรและหมดอายุ
- ES = รหัสข้อมูลแสดงการสิ้นสุดของข้อมูลในบัตร (End Sentinel) มีรหัสข้อมูลเป็น 0FH (?)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- LRC = ข้อมูลตรวจสอบความผิดพลาด (Longitudinal Redundancy Check)

0000000000	SS	Data,Data,Data,. . .,Data	ES	LRC	0000000000

จากภาพซึ่งใช้แสดง ลักษณะการจัดเรียงของข้อมูลที่เก็บไว้ในแถบแม่เหล็กในส่วนของ Track 2 ซึ่งจัดเรียงลำดับความสำคัญจากซ้ายไปขวาแบบอนุกรมทีละบิตโดยสังเกตเห็นว่ามีข้อมูลซึ่งเป็นลอจิกศูนย์หรือที่เรียกว่า "Clocking Bit" นำหน้าและปิดท้ายข้อมูลจริงอยู่ ซึ่งลักษณะการจัดเก็บข้อมูลของ Track 2 จะเก็บข้อมูลด้วยการเข้ารหัสแบบ "Modulo 5" ซึ่งในแต่ละชุดข้อมูลจะประกอบด้วยข้อมูล 5 บิต โดย 4 บิตแรก (D0..D3) เป็นรหัสข้อมูลแบบ BCD ส่วนบิตที่ 5 (D4) เป็นพาริตี้บิตแบบคี่ (ODD) เพื่อใช้ตรวจสอบความถูกต้องของข้อมูลในแต่ละไบท์ (5-Bit) ที่อ่านได้ ซึ่งหากค่าของพาริตี้ผิดพลาดแสดงว่าการอ่านข้อมูลนั้นล้มเหลว โดยเริ่มทำการอ่านข้อมูลเมื่อสัญญาณ Present เริ่มเป็นศูนย์ก่อน และทำการอ่านข้อมูลในทุก ๆ ขอบขาลงของสัญญาณนาฬิกาเสมอ ซึ่งข้อมูลในส่วนก่อนเริ่มต้นและหลังจากสิ้นสุดของการอ่านนี้จะมีค่าเป็นศูนย์ (สัญญาณ Data = logic "1") นำหน้าและปิดท้ายข้อมูลจริงอยู่ ซึ่งเรียกว่า "Clocking Bit" ซึ่งข้อมูลในส่วนนี้เราไม่ต้องสนใจ แต่ให้ตรวจสอบ และรอจนกว่าจะเริ่มเป็นข้อมูล Start บิต (0BH หรือ 11010 ในที่นี้เรียงลำดับความสำคัญจากซ้ายไปขวา ซึ่งเริ่มต้น ของรหัส Start หรือ 0BH ต้องเริ่มด้วย 1 เป็นบิตแรกเสมอ) ดังนั้น ในการอ่านเราต้องรอจนกว่าจะพบสัญญาณข้อมูลมีค่าเป็น "1" (สัญญาณ Data = Logic "0" เพราะกลับสภาวะกันอยู่) จึงเริ่มเก็บ ข้อมูลชุดละ 5 บิต ไปเรื่อยๆ จนถึงรหัสจบ (0FH) ซึ่งเมื่อพบรหัสจบแล้วจะมีข้อมูลตาม มาอีก 1 ไบท์ ซึ่งเป็นข้อมูลสำหรับตรวจสอบความผิดพลาด ของการอ่านข้อมูลทั้งหมดใน Track 2 เรียกว่า "LRC" สามารถหาได้จากการนำเอาข้อมูลในแต่ละไบท์ (ไม่คิดพาริตี้บิต) ตั้งแต่เริ่มต้นจนสิ้นสุดมาทำการ XOR กัน โดยครั้งแรกให้นำไบท์เริ่มต้น (0BH) ทำการ XOR กับศูนย์แล้วนำผลลัพธ์ที่ได้ไป XOR กับไบท์ถัดไปเรื่อยๆ ตามลำดับ ซึ่งหากผลลัพธ์สุดท้ายที่ได้ไม่เท่ากับค่าของ "LRC" ที่อ่านมาได้แสดงว่าการอ่านข้อมูลทั้งหมดล้มเหลว

สำหรับข้อมูลของรหัส BCD ที่ใช้สำหรับเครื่องอ่านแถบแม่เหล็กนี้จะเป็นข้อมูลชุดละ 5 บิต โดยเป็นข้อมูลจริง 4 บิต และเป็นรหัสตรวจสอบพาริตี้อีก 1 บิต ซึ่งมีจำนวนทั้งหมด 16 อักขระ ดังนี้ คือ

Parity	D3	D2	D1	D0	Character	Function
1	0	0	0	0	0 (0H)	Data
0	0	0	0	1	1 (1H)	Data
1	0	0	1	0	2 (2H)	Data
0	0	0	1	1	3 (3H)	Data
1	0	1	0	0	4 (4H)	Data
0	0	1	0	1	5 (5H)	Data
1	0	1	1	0	6 (6H)	Data
0	0	1	1	1	7 (7H)	Data

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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	< (DH)	Control
0	1	1	0	1	= (DH)	Field Separator
0	1	1	1	0	> (FH)	Control
1	1	1	1	1	? (FH)	End Sentinel

ตารางที่ 2.3 แสดงข้อมูลของรหัส BCD สำหรับเครื่องอ่านแถบแม่เหล็ก

2.14 มาตรฐานบัตร ATM มาตรฐาน ISO 7811/2 — 1985 (E)

เป็นมาตรฐานของบัตรเอทีเอ็ม หรือบัตรเครดิต ที่ใช้กันอยู่ทั่วไปนั้นจะมีทั้งหมด 2 แทร็ค โดยแทร็คที่ 2 จะเป็นแทร็คที่ใช้งานกันมากที่สุดซึ่งเป็นข้อมูลประจำตัวของผู้ถือบัตรจะมีข้อมูลอยู่ได้สูงสุด 40 Char/Track มีรูปแบบดังนี้

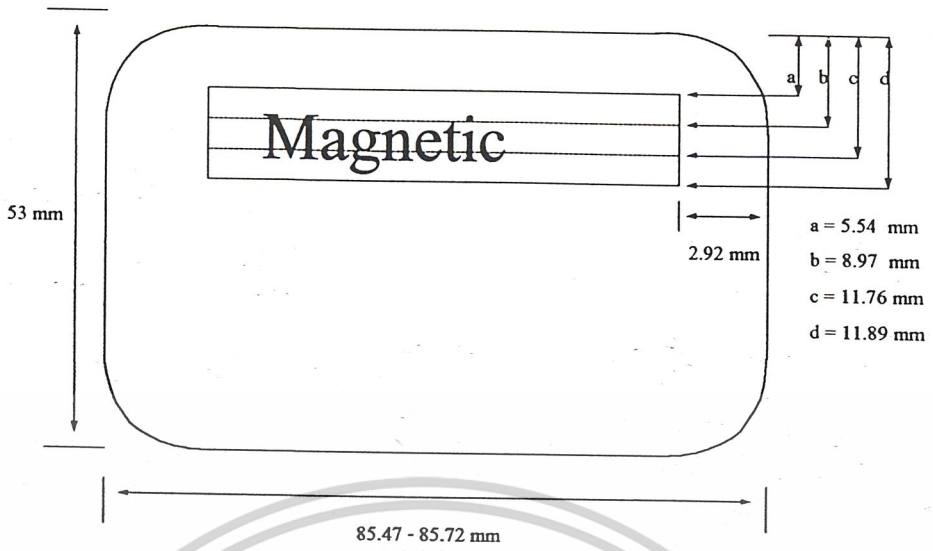
Sync	Sync	start	Data	Sep	Data	Stop	LRC	Sync	Sync
		Bh	0-9h	Dh	0-9h	Fh			

รูปที่ 2.14 รูปแบบของข้อมูลที่อยู่ในแทร็คที่ 2

- Sync เป็นข้อมูลที่ใช้สำหรับเปรียบเทียบข้อมูลที่จะใช้งานและเป็นการแสดงการเริ่มต้นข้อมูล
- Start จะเป็นข้อมูล "Bh" โดยจะใช้ข้อมูล 4 ตัว Parity 1 ตัว
- Data เป็นข้อมูลของบัตร โดยจะมีข้อมูล 4 ตัว Parity 1 ตัว มีค่าตั้งแต่ 0 - 9 ความยาวจะแล้วแต่การใช้งานของแต่ละธนาคาร
- Sep เป็นตัวเชื่อมหรือกั้นระหว่าง Data จะเป็นข้อมูล "Dh"
- Stop เป็นตัวหยุดข้อมูลหรือให้รู้ว่าข้อมูลหมดแล้วจะเป็นข้อมูล "Fh"
- LRC เป็นตัวตรวจสอบข้อมูลในแนวตั้ง

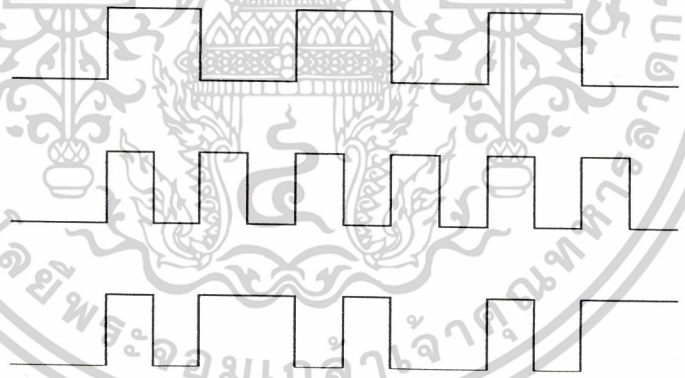
จากรูปที่ 2.15 บริเวณแทร็คที่สอง จะอยู่ระหว่างเส้นขนาน (b) ค่า 8.97 มิลลิเมตร (0.353 นิ้ว) (ค่าสูงสุด) หรือ 8.46 มิลลิเมตร (0.333 นิ้ว) (ค่าต่ำสุด) กับเส้นขนาน c ค่า 11.76 มิลลิเมตร (4.463 นิ้ว)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.15 ตำแหน่งแทร็คที่สองของบัตรแม่เหล็ก

โดยในการบันทึกข้อมูลในบัตรแม่เหล็กนั้นจะถูกบันทึกแบบ ฟรีเควนซี่ มอดูเลชั่น (FM) โดยค่าที่เป็น "1" จะมีความถี่สูงเป็น 2 เท่าของค่าที่เป็น "0" สามารถดูได้จาก ไทม์มิ่งไดอะแกรม (Timming Diagram) ดังรูปที่ 2.16



รูปที่ 2.16 Timming Diagram

ซึ่งบัตรแม่เหล็กที่ใช้ในโครงงานนี้เป็นไปตามมาตรฐานซึ่งบันทึกข้อมูล และสัญญาณนาฬิกา (Synchonous clock) อยู่ในแทร็คเดียวกัน หัวอ่านบัตรแม่เหล็กใช้เป็นหัวเทปแบบโมโน บัตรแม่เหล็กมีขนาด ความกว้าง 2.125 นิ้ว ความยาว 3.375 นิ้ว และความหนา 0.03 นิ้ว

โดยสัญญาณนาฬิกาที่ได้นั้นจะไปทำการอ้างอิงเพื่ออ่านข้อมูลจากตารางที่ 2.4 จะแสดงให้เห็นถึงรหัสข้อมูล (Code Character) ซึ่งเป็นรูปแบบ (Format) ของข้อมูลที่บันทึกลงบัตร จะเห็นว่ารหัสของข้อมูลจะเป็นรหัสไบนารี (Binary code) ขนาด 4 บิตและพาริตี้ อีก 1 บิต ซึ่งเป็นพาริตี้คี่ (Odd parity) ดังนั้นข้อมูล 1 ตัว (1 Character) จึงประกอบด้วยรหัสเลขฐานสอง ขนาด 5 บิต รหัสเหล่านี้ถูกนำมาเรียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ต่อกันด้วย Start , Character , Separater , End sentinel และข้อมูลประกอบกันเป็นบล็อกของข้อมูลที่ถูกบันทึกบนบัตรแม่เหล็ก ดังรูปที่ 2.17

ดังนั้นข้อมูลในแทร็คที่ 2 จะเป็นดังรูปที่ 2.14 ซึ่งมี sync เข้ามาก่อน 2 พัลส์ แล้วก็ตามด้วย start ซึ่งมีข้อมูลเป็น Bh แล้วก็ตามด้วย ข้อมูลมีตั้งแต่ 0—9h และจะมีช่องต่อระหว่าง ข้อมูลแต่ละตัว คือข้อมูล Dh จนถึง ข้อมูลตัวสุดท้ายก็จะเป็น stop คือข้อมูล Fh แล้วก็จะมีการเช็คข้อมูลทางแนวตั้งแล้วเป็นซิงค์ (Sync) เพื่อให้รู้ว่าข้อมูลที่ทำการอ่านนั้นหมดแล้ว

0B	01	01	13	04	15	16	07	08	19	1F

รูปที่ 2.17 ข้อมูลที่อยู่ในแทร็คที่ 2 แสดงลักษณะของข้อมูลที่ประกอบด้วย Start character และ character

โดยในแทร็คที่ 2 นั้นข้อมูลที่จะบันทึกลงบนบัตรจริงๆ แล้วจะมีเป็นตัวเลขตั้งแต่ 0-9h โดยข้อมูลจะมีทั้งหมด 4 บิต และมีพาริตี้รวมอยู่ด้วย 1 บิต เมื่อรวมแล้วข้อมูล 1 ตัว จะมีทั้งหมด 5 บิต ต่อ 1 ส่วน ข้อมูลตั้งแต่ ah — fh จะเป็นข้อมูลที่ทำการเช็คซึ่งสามารถเขียนเป็นรายละเอียดได้ดังตารางที่ 2.4

ซึ่งข้อมูลเหล่านี้จะถูกประมวลผล (Process) อีกครั้งด้วยซอฟต์แวร์เพื่อจะแยกเอาข้อมูลที่ถูกต้องอีกครั้งหนึ่ง ซึ่งจะได้กล่าวถึงวิธีการดึงข้อมูลซึ่งเป็นหมายเลขประจำตัวผู้ถือบัตรออกจากบล็อค (Block) ของข้อมูลอีกครั้งหนึ่งในส่วนของโปรแกรม (Program) และแผนผัง (Flowchart)

PARITY	B4	B3	B2	B1	ROW	CHAR
1	0	0	0	0	0	0
0	0	0	0	1	1	1
0	0	0	1	0	2	2
1	0	0	1	1	3	3
0	0	1	0	0	4	4
1	0	1	0	1	5	5
1	0	2	2	0	6	6
0	0	1	1	1	7	7
0	1	0	0	0	8	8
1	1	0	0	1	9	9
1	1	0	1	0	10	A
0	1	0	1	1	11	B
1	1	1	0	0	12	C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
50081

0	1	1	0	1	13	D
0	1	1	1	0	14	E
1	1	1	1	1	15	F

ตารางที่ 2.4 รหัสสำหรับแทร็คที่ 2

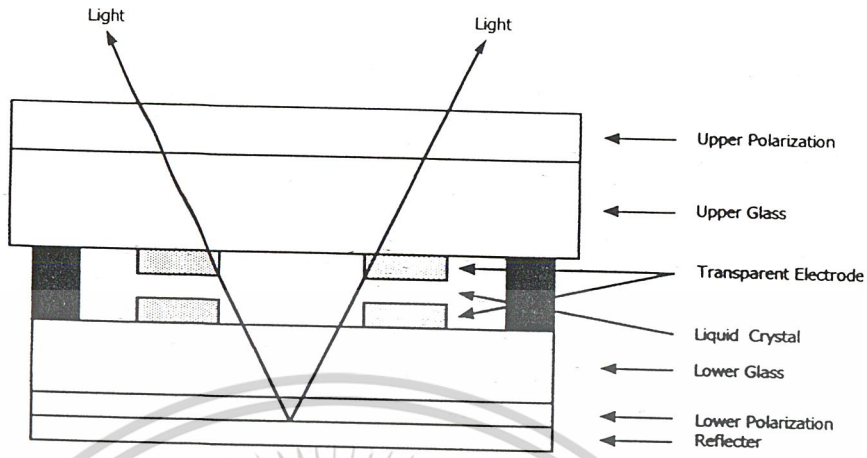
จะเห็นว่ารหัสของข้อมูลจะประกอบด้วยรหัสเลขฐาน 2 ขนาด 5 บิต ซึ่งเป็นรหัสของข้อมูลขนาด 4 บิต และพาริตี้คี่อีก 1 บิต รวมเป็น 5 บิต สำหรับเป็นรูปแบบ 1 ตัวอักษร

2.2 จอแสดงผลแบบผลึกของเหลว (Liquid Crystal Display : LCD)

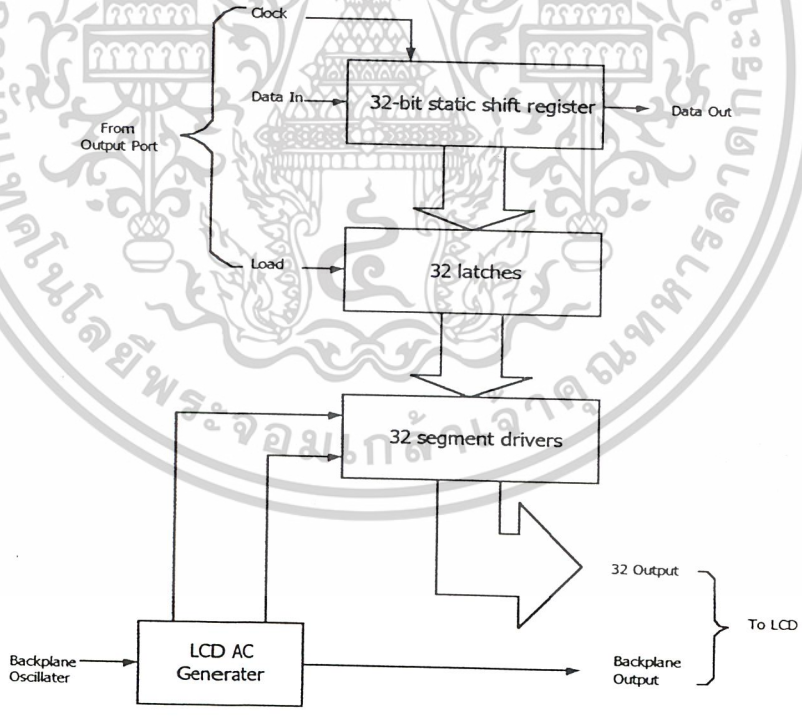
จอแสดงผลแบบผลึกของเหลว (Liquid Crystal Display : LCD) เป็นตัวแสดงผลที่ใช้พลังงานอยู่ในระดับไมโครวัตต์หรือน้อยกว่าเป็นพันเท่าเมื่อเทียบกับพลังงานที่ใช้ใน Light — Emitting Diode Display (LED) และยังมีคุณสมบัติเด่นๆ อีกคือ ใช้ไฟเลี้ยงต่ำอยู่ที่ 2 — 3 โวลต์ ใช้กระแสขับต่ำ ยกตัวอย่างการนำ LCD ไปใช้งาน เช่น เป็นตัวแสดงผลในเครื่องคิดเลข, นาฬิกา และในเครื่องมือวัดต่าง ๆ LCD สามารถแบ่งรูปแบบได้เป็น seven-segment, sixteen-segment และแบบ Dot Matrix ซึ่งใช้สำหรับแสดงเป็น Graphic แต่ถึงอย่างไร LCD ก็ยังมีข้อเสียอยู่บ้าง LCD ไม่สามารถกำหนดแสงในตัวเองหมือนตัวแสดงผล LED แต่มันจะสะท้อน หรือดูดแสง ดังนั้นเราจะไม่สามารถมองเห็นการแสดงผลของ LCD ได้ในที่มืด ซึ่งปัญหาข้อนี้แก้ไขได้โดยเพิ่มความเข้มของการแสดงผลของ LCD แต่ก็จะเป็นการสิ้นเปลืองพลังงานเป็นอย่างมาก อย่างไรก็ตามที่ LCD ไม่มีแสงในตัวเองนั้นยังมีประโยชน์อยู่คือ เราจะสามารถมองเห็นการแสดงผลของ LCD ในที่ซึ่งสว่างมาก ๆ หรือจากแสงอาทิตย์โดยตรง ส่วนข้อเสียอีกประการหนึ่งของ LCD คือ มีข้อจำกัดในเรื่องอุณหภูมิโดยที่อุณหภูมิต่ำๆ การแสดงผลของ LCD จะเปลี่ยนแปลงช้ามาก และที่อุณหภูมิสูงๆ การแสดงผลจะไม่เป็นไปตาม function ที่เรากำหนดเนื่องจากสารเคมีในคริสตอลเหลวมีการเปลี่ยนแปลง LCD โดยทั่วๆ ไปแล้วจะตอบสนองในช่วงอุณหภูมิ 0 ถึง 55 องศาเซลเซียส แต่มี LCD บางชนิดที่มีช่วงตอบสนองต่ออุณหภูมิอยู่ในช่วง -40 ถึง 85 องศาเซลเซียส ในการผลิต LCD เนื่องจาก LCD มีหลายชนิดด้วยกัน แต่จะอธิบายที่เป็นแบบธรรมดา เรียกว่า Twisted Nematic Field Effect (TNFE) ส่วนชนิดอื่น ๆ ก็มีคือ dichroic,supertwisted และ dynamic scattering display ในการผลิต LCD หัวใจสำคัญ ก็คือ คริสตอลเหลวภายใน

จากรูปที่ 2.18 แสดงโครงสร้างพื้นฐานของ LCD จะประกอบด้วยแผ่นแก้ว 2 แผ่นเป็นตัวหลักของโครงสร้าง ระหว่างแผ่นของแก้วจะมีชั้นบาง ๆ ของคริสตอลเหลว ด้านในผิวของแก้วแต่ละชิ้นจะเป็นช่อง และเป็นชั้นตัวนำของออกไซด์ และมีชั้น polarize ประกบติดอยู่กับชั้นแก้วด้านนอก เพื่อบังคับทิศทางแสงและจะเป็นตัวสะท้อนให้กลับไปยังตัว display ลักษณะการทำงานของ LCD โดยมาตรฐานแล้วใช้ CMOS XOR gates เป็นตัวขับ LCD แต่จะสามารถขับได้เฉพาะ 4 seg-ment และเอาท์พุตที่ออกมาจะแปลงให้อยู่ในแบบรหัส BCD แล้วส่งต่อไปยัง seven-segment

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

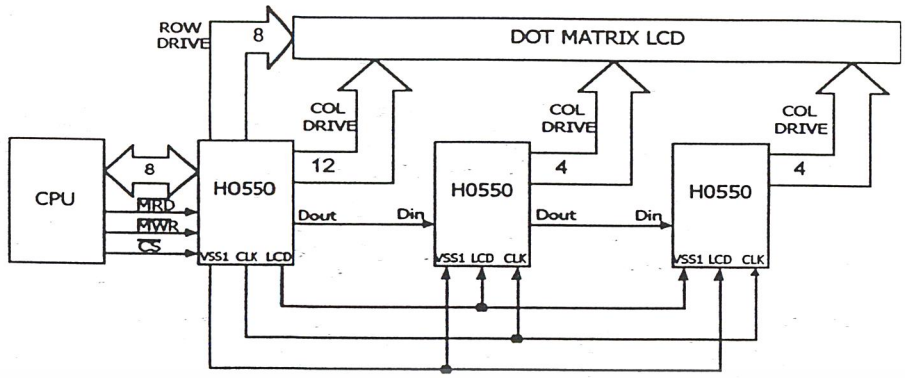


รูปที่ 2.18 แสดงโครงสร้างของ Liquid Crystal Display



รูปที่ 2.19 แสดง LCD Module

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.20 ตัวอย่าง LCD MODULE ที่ใช้ chip H0550 และ H0551

จากรูปที่ 2.19 เป็นบล็อกไดอะแกรมแสดง LCD Module เพื่อสะดวกในการใช้งาน ได้มีการพัฒนาหรือมีการนำ LCD มาทำเป็น module โดยการควบคุมบางอย่างจะกำหนดอยู่ในตัวชิพอยู่แล้ว อย่างเช่น การควบคุมแถวและหลัก จากที่เห็นในรูป IC H0550 จะเป็นตัวติดต่อกับไมโครโปรเซสเซอร์ และเป็นตัวควบคุมแถว และหลักของจุดแสดงผล ส่วน H0551 ที่ต่ออยู่นั้นจะเป็นตัวที่ใช้เพิ่มจำนวนหลักของตัวแสดงผล ดังนั้นในรูปตัวแสดงผลจะมี 8 แถว 80 หลัก หรือถ้าคิดเป็นตัวอักษรใน 1 แถว จะมีตัวอักษรอยู่ 16 ตัวอักษร ส่วนสัญญาณควบคุมที่มาจาก CPU จะประกอบไปด้วยสัญญาณ Read, สัญญาณ Write, Data bus และสัญญาณ Chipselect

LCD MODULE สามารถแบ่งออกเป็น แบบ Dot Matrix และ แบบ Graphic โดย LCD MODULE แบบ dot matrix จะแสดงผลเป็นตัวอักษรขนาด 5x7 และ 5x10 dot และจะมีจำนวนตัวอักษรและบรรทัดที่แตกต่างกันไปตามแต่ละรุ่น ในการติดต่อ LCD MODULE กับระบบไมโครคอมพิวเตอร์และระบบไมโครโปรเซสเซอร์สามารถติดต่อได้ 2 แบบคือ เป็นแบบ 4 บิต และแบบ 8 บิต ส่วนการอ้างอิงก็จะแบ่งเป็น 2 ลักษณะเช่นกันคือ Memory Map และ I/O Port

HD44780 เป็น ชิพ controller และ driver liquid crystal display แบบ dot matrix ที่สามารถแสดงผลได้ทั้งตัวเลข ตัวอักษร สัญลักษณ์ และสามารถต่อใช้งานร่วมกับไมโครคอมพิวเตอร์และไมโครโปรเซสเซอร์ขนาด 4 บิต และ 8 บิต ได้ ฟังก์ชั่นการควบคุม dot matrix จะอยู่ในตัว chip โดย chip HD44780 ควบคุมการแสดงผลได้ถึง 80 ตัวอักษร

2.2.1 คุณสมบัติต่าง ๆ ของ HD44780

1. เป็น dot matrix ขนาด 5x7 และ 5x10
2. สามารถใช้งานร่วมกับไมโครคอมพิวเตอร์และไมโครโปรเซสเซอร์ขนาด 4 บิต และ 8 บิต
3. แสดงผลได้สูงสุด 80 ตัวอักษร
4. ตัวอักษรที่กำเนิดจาก ROM
   - ขนาด 5x7 dot 160 ตัวอักษร
   - ขนาด 5x10 dot 32 ตัวอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. ตัวอักษรที่กำเนิดจาก RAM

   - ขนาด 5x7 dot 8 ตัวอักษร
   - ขนาด 5x10 dot 4 ตัวอักษร

6. คริสตอลเหลวภายใน ควบคุมการแสดงผล

   - 16 สัญญาณควบคุม
   - สัญญาณ Driver 40 Segment สามารถเพิ่มความยาวของตัวแสดงผลโดยต่อ chip HD44100 จะควบคุมได้ถึง 360 Segment

7. มีฟังก์ชั่นคำสั่งต่างๆ

   - เคลียร์ข้อมูล Cursor Home, Display ON/OFF, Cursor ON/OFF, ตัวอักษรกระพริบ, เลื่อน Cursor และตัวอักษรเลื่อน

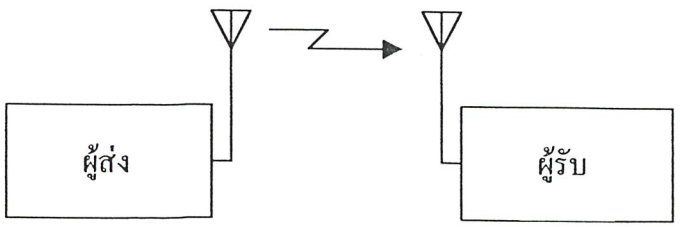
2.3 การรับส่งข้อมูล (DATA TRANSMISSION)

การรับ – ส่ง ข้อมูลเพื่อโอนถ่ายหรือแลกเปลี่ยนข้อมูลกันระหว่างผู้ส่งผู้รับ จะสำเร็จขึ้นได้ ต้องประกอบด้วยปัจจัย 2 ประการ คือ คุณภาพของที่รับส่งกันและคุณลักษณะของสายสื่อสาร สำหรับส่งผ่านข้อมูล

การส่งสัญญาณข้อมูล หมายถึง การส่ง (นำ) ข้อมูลหรือข่าวสารจากเครื่องส่งหรือผู้ส่งผ่านทางสื่อหรือตัวกลางไปยังเครื่องรับหรือผู้รับ ข้อมูลหรือข่าวสารที่ถูกส่งออกไปอาจจะอยู่ในรูปของสัญญาณเสียง สัญญาณคลื่นแม่เหล็กไฟฟ้าหรือแสงก็ได้ โดยที่สื่อหรือตัวกลางของสัญญาณสามารถแบ่งออกได้เป็น 2 แบบ คือ

1. แบบสามารถกำหนดเส้นทางสัญญาณได้ ได้แก่ สายเกลียวคู่ สายโคแอคเชียล และสายไฟเบอร์ออปติค
2. แบบไม่สามารถกำหนดเส้นทางสัญญาณได้ ได้แก่ ใช้บรรยากาศ สุญญากาศ และน้ำ

รูปที่ 2.21 การกำหนดเส้นทางสัญญาณได้



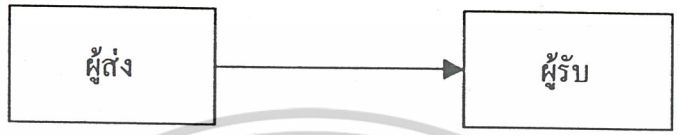
รูปที่ 2.22 การกำหนดเส้นทางสัญญาณไม่ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบของการส่งสัญญาณข้อมูล

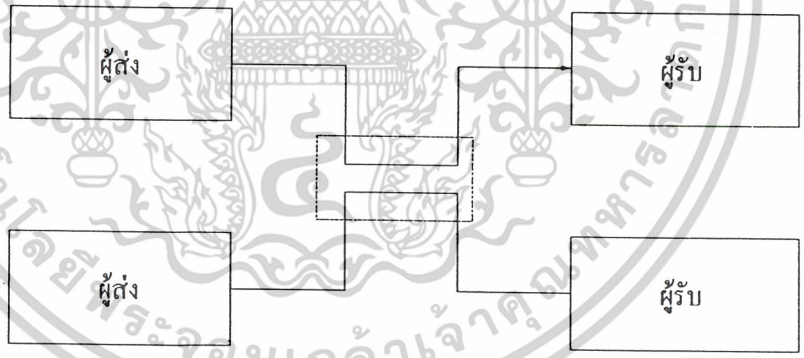
เราสามารถจัดแบ่งรูปแบบการส่งสัญญาณข้อมูลได้เป็น 4 รูปแบบดังนี้

1. แบบทิศทางเดียว หรือซิมเพล็กซ์ (One-way หรือ Simplex) ในการส่งสัญญาณข้อมูลแบบ simplex ข้อมูลจะถูกส่งไปในทิศทางเดียวเท่านั้น เช่น การกระจายเสียงของสถานีวิทยุ หรือการแพร่ภาพทางโทรทัศน์



รูปที่ 2.23 แบบทิศทางเดียว

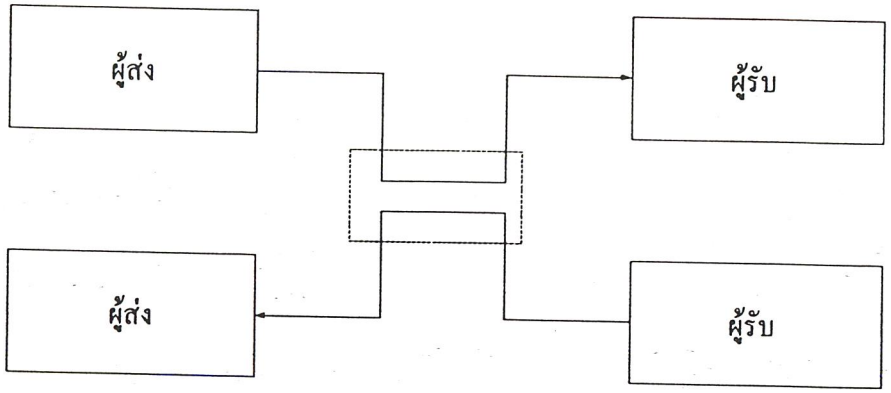
2. แบบกึ่งทางคู่ หรือแบบครึ่งดูเพล็กซ์ (Either-Way of Two Ways หรือ Half Duplex) การสื่อสารแบบครึ่งดูเพล็กซ์ เราสามารถส่งข้อมูลสวนทางกันได้แต่ต้องสลับกันส่ง จะทำในเวลาเดียวกันไม่ได้ เช่น วิทยุสื่อสารของตำรวจ ซึ่งต้องอาศัยการสับสวิตช์ เพื่อแสดงการเป็นผู้ส่งสัญญาณ และให้อีกด้านหนึ่งเป็นผู้รับสัญญาณ คือ ต้องผลัดกันพูด



รูปที่ 2.24 แบบกึ่งทางคู่

3. แบบทางคู่ หรือแบบดูเพล็กซ์เต็ม (Both-Way หรือ Full Duplex) ในแบบนี้เราสามารถส่งข้อมูลได้ในเวลาเดียวกันทั้งสองทาง เช่น การพูดทางโทรศัพท์ สามารถพูดพร้อมกันได้ ประโยชน์การใช้งานของการส่งสัญญาณดูเพล็กซ์เต็มย่อมได้ประโยชน์ใช้สอยได้ดีกว่า รวมทั้งลดเวลาส่งสัญญาณ อย่างไรก็ตามค่าใช้จ่ายในการติดตั้งและอุปกรณ์ของระบบการส่งสัญญาณแบบดูเพล็กซ์เต็ม ย่อมแพงกว่าและยุ่งยากกว่าเช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.25 แบบทางคู่

การสื่อสารระหว่างเครื่องอ่านบัตรแม่เหล็กกับเครื่องคอมพิวเตอร์

การส่งข้อมูลจากเครื่องอ่านบัตรแม่เหล็กไปยังคอมพิวเตอร์ สามารถส่งข้อมูลผ่านพอร์ตต่าง ๆ ของคอมพิวเตอร์ แบ่งออกได้ดังนี้

1. การสื่อสารข้อมูลแบบขนาน
2. การสื่อสารข้อมูลแบบอนุกรม

เนื่องจาก โครงงานชิ้นนี้เกี่ยวเนื่องในส่วนการสื่อสารข้อมูลแบบอนุกรม จึงจะขอกล่าวถึงเฉพาะเรื่องการสื่อสารแบบอนุกรมเท่านั้น

2.3.1 การสื่อสารข้อมูลแบบอนุกรม

การติดต่อสื่อสารผ่านพอร์ตอนุกรมนั้น มีข้อยุ่งยากมากว่าการใช้พอร์ตขนาน สำหรับอุปกรณ์ทั่ว ๆ ไปแล้ว จะต้องการข้อมูลในการประมวลผลแบบขนาน ดังนั้นจึงเพิ่มอุปกรณ์ที่ทำหน้าที่ในการแปลงข้อมูลแบบอนุกรมมาเป็นข้อมูลแบบขนาน และซอฟต์แวร์ที่ใช้ควบคุมการทำงานก็จะยุ่งยากกว่าการควบคุมผ่านพอร์ตมาตรฐานแบบขนาน แต่การสื่อสารโดยใช้พอร์ตอนุกรมนั้นก็มีข้อดีอยู่ไม่น้อยเช่นกัน อาทิเช่น

1. การติดต่อสื่อสารผ่านพอร์ตอนุกรมสามารถใช้สายได้ยาวกว่าการติดต่อสื่อสารผ่านพอร์ตขนาน โดยที่พอร์ตอนุกรมจะใช้ระดับแรงดันในช่วง -3 ถึง -25 V แทนลอจิก “0” และใช้แรงดันในช่วง +3 ถึง +25 V แทนลอจิก “1” ดังนั้นจะเป็นว่าช่วงการสวิงแรงดันของพอร์ตอนุกรมจะมีค่าประมาณ 50 V ส่วนพอร์ตขนานจะมีช่วงสวิง 5 V เท่านั้น ซึ่งจะเห็นได้อย่างชัดเจนว่าหากมีการสูญเสียในสายแล้ว การสื่อสารผ่านพอร์ตอนุกรมจะสามารถส่งข้อมูลไปได้ไกลกว่าอย่างแน่นอน
2. ใช้จำนวนสายน้อยกว่าการส่งข้อมูลแบบขนาน ในกรณีที่อุปกรณ์อยู่ห่างจากคอมพิวเตอร์มาก ๆ ย่อมจะสะดวกและประหยัดกว่าหากจะเดินสายเพียง 3 เส้น ซึ่งเป็นลักษณะโครงสร้างของนัลโมเด็ม (Null Modem) เมื่อเทียบกับการเดินสายจำนวน 19 หรือ 25 เส้น ในการใช้พอร์ตขนาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ปัจจุบันการใช้อุปกรณ์ที่ใช้แสงอินฟาเรด ได้รับความนิยมมากขึ้น ซึ่งจะเห็นได้จากอุปกรณ์ประเภทสมุดบันทึกอิเล็กทรอนิกส์ คอมพิวเตอร์แบบโน้ตบุ๊ค ฯลฯ จะมีการติดต่อสื่อสารโดยใช้อินฟาเรดร่วมอยู่ด้วย และแน่นอนว่าการใช้อินฟาเรดก็จะต้องใช้การติดต่อสื่อสารแบบอนุกรม เนื่องจากความไม่สะดวกอย่างยิ่งในการส่งข้อมูลแบบขนานด้วยอินฟาเรด

2.4 RS-232 มาตรฐานเพื่อการสื่อสาร

มาตรฐาน RS-232 ประกาศและได้กำหนดการอินเตอร์เฟสระหว่างอุปกรณ์เทอร์มินัลของข้อมูล (Data Terminal Equipment , DTE) กับอุปกรณ์ส่วนปลายของวงจรข้อมูล (Data Circuit Terminating Equipment , DCT) อุปกรณ์ DTE ตามปกติแล้วจะเป็นวงจรเทอร์มินัลที่พูดไม่ได้ (dumb terminal) เป็นอุปกรณ์ที่มีความฉลาด เช่น ไมโครคอน โทรลเลอร์ หรือเครื่องคอมพิวเตอร์ส่วนบุคคลที่ความฉลาดในการสร้างบิตข้อมูลเป็นแบบอนุกรมได้ อุปกรณ์ DCE จะรับบิตข้อมูลที่อุปกรณ์ DTE ส่งออกมา ผ่านทางการอินเตอร์เฟสแบบ RS-232 และแปลงให้อยู่ในรูปที่เหมาะสมสำหรับถ่ายทอดผ่านตัวกลางการสื่อสารจากระยะไกล เช่นผ่านทางสายโทรศัพท์

ถ้าคิดตามมาตรฐาน RS-232 ต่อไปจะพบว่าในทางกายภาพแล้วขั้วต่อพอร์ตของอุปกรณ์ DTE จะเป็นขั้วต่อตัวผู้และขั้วต่อพอร์ตของอุปกรณ์ DCE จะเป็นขั้วต่อตัวเมีย พอร์ตอนุกรมของคอมพิวเตอร์ส่วนบุคคลปกติแล้วจะเป็นอุปกรณ์ DTE และพอร์ตของโมเด็มก็มักจะมีโครงสร้างเป็นอุปกรณ์ DCE

พอร์ตอนุกรมของเครื่องพีซีจะเป็นตัวผู้ไม่ว่าจะเป็นขั้วต่อแบบ 9 ขา หรือ 25 ขา ส่วนข้อต่อของโมเด็มส่วนมากแล้วจะเป็นขั้วต่อแบบตัวเมีย 25 ขา แม้ว่าขั้วต่อแบบ 9 ขา จะไม่ได้มาตรฐาน RS-232 ของ EIA แต่ปัจจุบันนี้มีการใช้อยู่ทั่วไปในการอินเตอร์เฟสแบบ RS-232 การอินเตอร์เฟสที่ใช้ขั้วต่อแบบ 9 ขา พบจำหน่ายครั้งแรกในเครื่องพีซีรุ่น AT ในตอนต้นศตวรรษ 1980

มาตรฐาน RS-232 ยังกำหนดคุณลักษณะสำหรับการสื่อสารผ่านทางสายส่งสัญญาณเสียงด้วยอัตราความเร็วสูงถึง 9600 bps ซึ่งช่วยให้สามารถมีการสื่อสารแบบอะซิงโครนัสจากจุดหนึ่งไปยังอีกจุดหนึ่งผ่านทางเครือข่ายโทรศัพท์สาธารณะในขณะนั้นได้

RS-232 กำหนดวงจร 21 วงจร ในการอินเตอร์เฟสนี้ ตารางที่ 2.5 คือสรุปคำบรรยายหน้าที่ของขาในมาตรฐาน RS-232 A , B และ C

ในระหว่างการส่งข้อมูลสภาวะมีข้อมูล (mark condition) บ่งชี้สถานะด้วย “1” ในระบบเลขฐานสอง และสภาวะช่องว่าง (space condition) บ่งชี้ด้วยสถานะ “0” ในระบบเลขฐาน 2

สำหรับวงจรกำหนดจังหวะเวลาและควบคุมการแลกเปลี่ยนข้อมูล การทำงานของมันจะเป็น “on” เมื่อแรงดันไฟฟ้าบวกมากกว่า +3 โวลต์และจะเป็นแบบ “off” เมื่อแรงดันไฟฟ้าเป็นลบมากกว่า -3 โวลต์ เมื่อเทียบกับกราวด์การทำงานนี้จะไม่สามารถกำหนดได้ถ้าแรงดันไฟฟ้าอยู่ในช่วงของการเปลี่ยนแปลงระหว่าง -3 ถึง +3 โวลต์ เมื่อเทียบกราวด์

คำว่า “mark” และ “space” จะใช้อยู่ทั่วไปในเอกสารกำหนดคุณลักษณะเพื่อบรรยายสภาวะของสายนำข้อมูลหรือสายสัญญาณควบคุมในระบบ RS-232

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หมายเลขขาสัญญาณ	ชื่อของสายสัญญาณ	ทิศทางของสัญญาณ
1	Positive Ground	N.A.
2	Transmitted Data	TO DCE
3	Received Data	TO DCE
4	Request To Send	TO DCE
5	Clear To Send	TO DCE
6	Data Set Ready	TO DCE
7	Signal Ground	N.A.
8	Received Line Signal Detector (RS-232) Data Carrier Detect (RS-232)	TO DCE
11	Select Standby	TO DCE
12	Secondary Receive Line Signal Detector	TO DCE
13	Secondary Clear To Send	TO DCE
14	Secondary Transmitted Data	TO DCE
14	New Sync	TO DCE
15	Transmitted Signal Element Timing	TO DTE
16	Secondary Received Data	TO DTE
17	Receiver Signal Element Timing	TO DTE
18	Test	TO DCE
19	Secondary Request To Send	TO DCE
20	Data Terminal Ready	TO DCE
21	Signal quality Detector	TO DTE
22	Ring/Calling Indicator	TO DTE
23	Data Signal Rate Selector	TO DCE
23	Data Signal Rate Selector	TO DTE
24	Transmitter Signal Element Timing	TO DCE

ตารางที่ 2.5 แสดงรายการคุณลักษณะเฉพาะสำหรับการอินเตอร์เฟสแบบ RS-232

ปัจจุบันนี้อุปกรณ์ส่วนใหญ่ที่มีวางจำหน่ายอยู่จะเป็นไปตามมาตรฐาน RS-232 หรือ RS-232D (มาตรฐาน CCITT V.24 และ V.28 ก็ยังคงมีใช้อยู่ทั่วไปอย่างกว้างขวาง) วงจรของ RS-232 ส่วนมากไม่ได้ใช้ในการติดต่อสื่อสารระหว่างเครื่องเทอร์มินัล 2 เครื่อง หรือคอมพิวเตอร์ 2 เครื่องโดยตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำบรรยายลักษณะวงจร RS-232

สิ่งแรกที่คุณอาจสังเกตเห็นก็คือ ตารางที่ 2.7 จะแสดงสายสัญญาณเพียง 11 เส้น จาก 25 เส้น ที่เป็นไปได้ของระบบ RS-232 ที่ต้องการใช้ในการสื่อสารระหว่าง DTE ไปยัง DCE ให้สมบูรณ์ ส่วนมากแล้วสามารถจะทิ้งสายวงจรตัวตรวจจับอัตราสัญญาณข้อมูล (Data Signal Rate Detector) และสายวงจรกราวด์ (Protective Ground) ออกไปได้ ทำให้เหลือสายสัญญาณที่ต้องต่อเพียง 9 เส้น

RS-232 เป็นข้อกำหนดอินเตอร์เฟสมาตรฐานและสามารถใช้ เพื่อจุดประสงค์อื่น ๆ ต่าง ๆ กันไป เช่น การสื่อสารแบบซิงโครนัส (Synchronous Communication) และรูปแบบการสื่อสารสามารถทำให้มีการสนทนากันจาก DTE ไปยัง DCE โดยสัญญาณเพียง 3 เส้น จากจำนวน 11 เส้น ถ้าอุปกรณ์ DTE และ DCE ใช้ซอฟต์แวร์ที่เขียนขึ้นตามความต้องการของลูกค้า (Custom — Write — Software) ก็จะใช้เพียงสาย TD , RD และสายกราวด์สัญญาณเท่านั้นในการย้ายข้อมูลไปตามสายตัวนำ 3 เส้นนี้

หมายเลขขาสัญญาณ	ชื่อของขาสัญญาณ
1	Protective Ground
2	Transmitted Data
3	Received Data
4	Request To Send
5	Clear To Send
6	Data Set Ready
7	Signal Common
8	Received line Signal Detect
9	Reserved For Testing
10	Reserved For Testing
11	Unassigned
12	Secondary Received Line Signal Detector
13	Secondary Clear To Send
14	Secondary Transmitted Data
15	Transmission signal Element Timing
16	Secondary Received Data
17	Transmission Signal Element
18	Unassigned
19	Secondary Request to Send
20	Data Terminal Ready
21	Signal quality Detector

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

22	Ring Indicator
23	Data Signal Rate
24	Transmitter Signal Element Timing
25	Unassigned

ตารางที่ 2.6 แสดงรายละเอียดสำหรับการอินเตอร์เฟส RS-232 โดยใช้คอนเน็กเตอร์แบบ DB-25

ข้อกำหนดของขาสัญญาณ 11 ขา

1. ขา 1 (Protective Ground Circuit, AA) ขานี้จะต่อเข้ากับตัวถังของอุปกรณ์และสามารถต่อเข้ากับกราวด์ภายนอกถ้าอุปกรณ์อื่นๆ ต้องใช้ขานี้
2. ขา 2 (Transmitted Data Circuit, TD) เป็นขาสัญญาณข้อมูลที่ออกมาจากอุปกรณ์ DTE กระแสบิตข้อมูลอนุกรมจากขานี้ คือข้อมูลที่จะถูกถอดรหัสโดยอุปกรณ์ DCE
3. ขา 3 (Received Data Circuit BB, RD) สัญญาณที่ขานี้จะถูกสร้างจากอุปกรณ์ DCE กระแสบิตอนุกรมนี้จะกำเนิดขึ้นที่อุปกรณ์ DTE ปลายทางและเป็นผลผลิตของวงจรนับข้อมูลของอุปกรณ์ DCE สัญญาณนี้มักจะเป็นข้อมูลดิจิตอลที่ถูกสร้างขึ้นโดยอุปกรณ์ DCE ที่มีความฉลาดหรือจากวงจรถอดสัญญาณ (demodulation) ของโมเด็ม
4. ขา 4 (Repeat to Send Circuit CA, RTS) สัญญาณนี้จะเตรียมพร้อมอุปกรณ์ DCE สำหรับทำการส่งข้อมูลเมื่อสัญญาณ RTS นี้อยู่ในสภาวะ "ON" จะทำให้อุปกรณ์ DCE อยู่ในโหมดส่งข้อมูล (Transmit Mode) ในขณะที่สัญญาณนี้อยู่ในสภาวะ "OFF" จะทำให้อุปกรณ์ DCE อยู่ใน โหมดรับข้อมูล (Receive Mode) อุปกรณ์ DCE ควรจะตอบสนองต่อสัญญาณ RS ON โดยการทำให้สัญญาณ Clear to Send (CTS) อยู่ในสภาวะ "ON" ด้วย เมื่อสัญญาณ RTS อยู่ในสภาวะ "OFF" สัญญาณ ไม่ควรจะ "ON" ขึ้นอีก จนกว่าสัญญาณ TRS อยู่ในสภาวะ "OFF" เสียก่อน สัญญาณนี้จะถูกใช้ร่วมกับสัญญาณ DTR, DSR และ DCD ขาสัญญาณ RTS จะถูกใช้อย่างมากในการควบคุมการไหลของข้อมูล
5. ขา 5 (Clear to Send Circuit CB, CTS) สัญญาณนี้จะตอบรับกลับไปยังอุปกรณ์ DTE เมื่อได้รับสัญญาณ RTS และข้อมูลสามารถส่งออกไปได้ ข้อมูลจะถูกส่งออกไปตามตัวกลางที่ใช้สื่อสารได้ก็ต่อเมื่อสัญญาณ CTS นี้อยู่ในสภาวะ "ON" เท่านั้น สัญญาณนี้จะใช้ร่วมกับขา DTR,DSR และ ขาสัญญาณ CTS นี้จะใช้ร่วมกับขา RTS สำหรับควบคุมการไหลของข้อมูล
6. ขา 6 (Data Set Ready Circuit CC,DSR) สัญญาณจะบอกต่ออุปกรณ์ DTE ว่าอุปกรณ์ DCE ได้ต่อกับตัวกลางสื่อสารที่ถูกต้องแล้วและในบางกรณีจะบ่งชี้ว่าอุปกรณ์ DCE กำลังอยู่ในโหมด dialing หรือกำลังติดต่ออุปกรณ์ DCE อีกตัวหนึ่งอยู่ เมื่อสัญญาณ DSR นี้อยู่ในสภาวะ "OFF" อุปกรณ์ DTE ก็ควรจะถูกกำหนดให้ไม่สนใจสัญญาณอื่นๆ ทั้งหมดจากอุปกรณ์ DCE ถ้าสัญญาณนี้ถูกทำให้อยู่ในสภาวะ "OFF" ก่อนอุปกรณ์ DTR แล้วอุปกรณ์ DTE ก็จะสรุปว่าการสื่อสารกันนั้นสิ้นสุดลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. ขา 7 (Signal Common Circuit, AB) ขานี้จะให้สัญญาณอ้างอิงของกราวด์ร่วมกัน สำหรับวงจรการแลกเปลี่ยนข้อมูลทั้งหมด ยกเว้นวงจร AA หรือ Protective ground ข้อกำหนด RS-232 จะอนุญาตให้วงจรนี้ถูกทำให้อยู่ในสภาวะ "OFF" ก่อนอุปกรณ์ DTR แล้วอุปกรณ์ DTE ก็จะสรุปว่าการสื่อสารกันนั้นสิ้นสุดลง
8. ขา 8 (Data Carrier Detect CF, DCD) ขานี้ยังรู้จักกันในนามของ Received Line Signal Detect (RLSD) หรือขา Carrier Detect (CD) สัญญาณนี้จะแอคทีฟเมื่อเกิดสัญญาณพาหะที่เหมาะสมระหว่างอุปกรณ์ DCE ที่สถานีที่อยู่ระยะไกล เมื่อสัญญาณนี้อยู่ในสภาวะ "OFF" สัญญาณที่ขา RD ควรจะถูกทำให้ต่างอยู่ในสภาวะ "Mark" (สถานะ "1" ในเลขฐานสอง)
9. ขา 20 (Data Terminal Ready Circuit CD, DTR) สัญญาณ DTR ถูกควบคุมในการสวิทช์อุปกรณ์ DCE เข้ากับตัวกลางในการสื่อสารสัญญาณ DTR ON บ่งชี้ว่าอุปกรณ์ DCE ที่กำลังต่อเชื่อมกันอยู่ ก็ยังคงต่อเชื่อมกันและถ้าไม่มีการต่อเชื่อมกันก็สามารถทำการต่อเชื่อมกันครั้งใหม่ได้ ปกติแล้วสัญญาณ DTR จะอยู่ในสภาวะ "OFF" เพื่อกระตุ้นให้เกิดสภาวะ ON HOOK (วางสาย) (Hang Up) อุปกรณ์ DCE โดยปกติแล้วจะตอบสนองต่อการกระตุ้นจากสัญญาณ DTR โดยการทำให้สัญญาณ DSR แอคทีฟ
10. ขา 22 (Ring Indicator Circuit CE, RI) สภาวะ "ON" ของสัญญาณนี้ จะบ่งชี้ได้ว่าได้รับสัญญาณเรียกสายโทรศัพท์จากตัวกลางในการสื่อสาร (สายโทรศัพท์) ปกติแล้วจะขึ้นอยู่กับโปรแกรมควบคุมในการที่จะทำให้เกิดสัญญาณนี้ขึ้นหรือไม่
11. ขา 23 (Data Signal Rate Detector Circuit CH/CI, DSRD) วงจร CH เป็นส่วนประกอบของ DTE และวงจร CI เป็นส่วนประกอบของ DCE สัญญาณที่ขานี้ถูกใช้ในการเลือกค่าอัตราการส่งสัญญาณข้อมูลค่าใดค่าหนึ่งในสองค่า ในกรณีที่ใช้โมเด็มที่มีอัตราการส่งข้อมูลได้ 2 ค่า (Dual – Rate Modem) ถ้าสัญญาณที่ขานี้เป็น "ON" ก็จะเป็นการเลือกอัตราการส่งข้อมูลที่มีค่าสูงที่สุดใน 2 ค่านั้น

หมายเลขขาสัญญาณ	ชื่อสายสัญญาณ
1	Data Carrier Detect
2	Received Data
3	Transmitted Data
4	Data Terminal Ready
5	Signal Common
6	Data Set Ready
7	Request To Send
8	Clear Td Send
9	Ring Indicator

ตารางที่ 2.7 การต่อแบบคอนเน็กเตอร์แบบ DB-9 ตามมาตรฐาน RS-232

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หมายเลขขาสัญญาณ	ชื่อของสายสัญญาณ
1	Protective Ground
2	Transmitted Data
3	Received Data
4	Request To Send
5	Clear To Send
6	Data Set Ready
7	Signal Common
8	Data Carrier Detect
20	Data Terminal Ready
22	Ring Indicator
23	Data Signal Rate Detector

ตารางที่ 2.8 แสดงขาสัญญาณที่ต่อจาก DTE ไปยัง DCE โดยใช้คอนเน็กเตอร์แบบ DB-25

ขั้นตอนการติดต่อระหว่างอุปกรณ์ DTE และ DCE

1. เมื่อจ่ายกำลังงานให้อุปกรณ์ DTE และอุปกรณ์นี้ส่งสัญญาณ DTR ออกมา
2. อุปกรณ์ DCE ถูกเปิดขึ้น และรับรู้ถึงสัญญาณ DTR ที่ส่งมาจากอุปกรณ์ DTE
3. อุปกรณ์ DCE ส่งสัญญาณ DSR ออกมา และโมเด็มก็กระทำกระบวนการ OFF HOOK
4. ถ้าสายสัญญาณโทรศัพท์อยู่ในสภาพดี และปลายทางอีกด้านหนึ่ง ก็พร้อมจะรับข้อมูลอยู่แล้ว โดยจะตรวจจับพบสัญญาณพาหะ แล้วอุปกรณ์ DCE จะส่งสัญญาณ DCD ออกมา
5. อุปกรณ์ DTE ยกระดับสัญญาณ RTS ขึ้นสูง
6. อุปกรณ์ DCE จะตอบสนองด้วยการส่งสัญญาณ CTS ออกมา
7. การติดต่อสื่อสารก็เริ่มขึ้น โปรแกรมควบคุมจะทำการส่งหรือรับข้อมูล ส่วนลำดับขั้นในการตอบรับก็จะเป็นในทำนองนี้

ลักษณะทางฮาร์ดแวร์

อุปกรณ์ที่ใช้สื่อสารแบบอนุกรมสามารถแยกออกได้เป็น 2 ประเภท คือ DCE (Data Communication Equipment) อุปกรณ์เหล่านี้ ได้แก่ โมเด็ม ฯลฯ และ DTE (Data Terminal Equipment) ซึ่งก็คือคอมพิวเตอร์นั่นเอง

ข้อกำหนดทางไฟฟ้าของพอร์ตอนุกรมได้ถูกกำหนดเป็นมาตรฐานโดย EIA (Electronics Industry Association) หรือ RS-232 ซึ่งประกอบไปด้วยสิ่งต่างๆ เหล่านี้

1. ช่วงไม่มีข้อมูล (space) หรือลอจิก “0” ต้องมีแรงดันอยู่ในช่วง -3 ถึง -25 V
2. ช่วงข้อมูล (Mark) หรือลอจิก “1” ต้องมีแรงดันอยู่ในช่วง +3 ถึง +25 V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

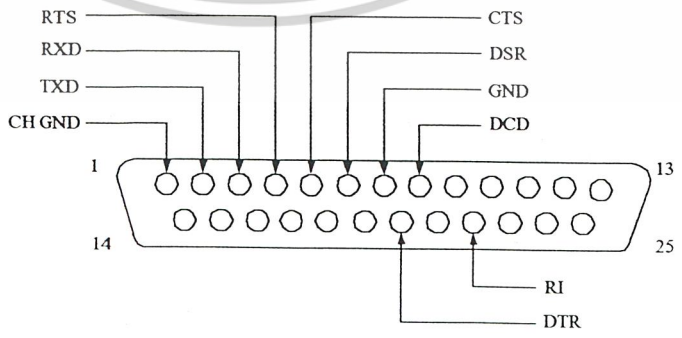
3. แรงดันอยู่ในช่วง -3 ถึง +3 V ไม่มีการนิยามไว้
4. แรงดันในขณะเปิดวงจรต้องมีค่าไม่เกิน 25 V
5. กระแสขณะช็อตวงจรมีค่าไม่เกิน 500 mA

ข้อมูลเหล่านี้ยังไม่ใช่ข้อกำหนดที่ครอบคลุมมาตรฐานของ RS-232 ทั้งหมด มาตรฐานของ RS-232 นั้น นอกจากจะมีคุณสมบัติที่ได้กล่าวมาแล้วยังจะต้องประกอบด้วยค่าคาปาซิแตนซ์ของสาย อัตราบอดสูงสุด ฯลฯ ด้วย

พอร์ตอนุกรมนี้จะมีอยู่ด้วยกัน 2 ขนาด คือ คอนเน็กเตอร์แบบ D — type ตัวผู้ขนาด 25 ขา รูปที่ 2.26 (ก) และคอนเน็กเตอร์ แบบ D — type ตัวผู้เช่นกันขนาด 9 ขา รูปที่ 2.26 (ข) ซึ่งคอนเน็กเตอร์ทั้ง 2 แบบนี้จะติดอยู่ที่ด้านหลังของคอมพิวเตอร์ ตารางที่ 2.9 แสดงตำแหน่งขาสัญญาณของพอร์ตอนุกรม

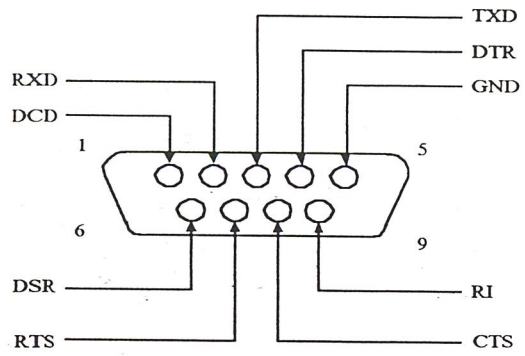
D-type 25 Pin No.	D-type 9 Pin No.	Abbreviation	Full Name
Pin 2	Pin 3	TD	Transmit Data
Pin 3	Pin 2	RD	Receive Data
Pin 4	Pin 7	RTS	Request to Send
Pin 5	Pin 8	CTS	Clear to Send
Pin 6	Pin 6	DSR	Data Set Ready
Pin 7	Pin 5	SG	Signal Ground
Pin 8	Pin 1	CD	Carrier Detect
Pin 20	Pin 4	DTR	Data Terminal Ready
Pin 22	Pin 9	RI	Ring Indicator

ตารางที่ 2.9 แสดงตำแหน่งขาสัญญาณของพอร์ตอนุกรม



(ก) คอนเน็กเตอร์แบบ DB-25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ข) คอนเน็กเตอร์แบบ DB-9

รูปที่ 2.26 คอนเน็กเตอร์แบบ DB-25 และแบบ DB-9

Name	Address	IRQ
COM 1	3F8	4
COM 2	2F8	3
COM 3	3E8	4
COM 4	2E8	3

ตารางที่ 2.10 ตำแหน่งมาตรฐานของพอร์ตอนุกรม

2.5 Real Time Clock : RTC (DS1307)

2.5.1 คุณสมบัติ

- เป็นนาฬิกาที่สามารถให้ข้อมูลออกมาเป็นวินาที นาที ชั่วโมง วันที่ของเดือน เดือน วันของสัปดาห์ ปี ซึ่งชดเชยค่าแล้วสามารถใช้งานได้ถึงปี 2100
- มี RAM แบบ Nonvolatile จำนวน 56 ไบต์
- สามารถตั้งโปรแกรมให้ส่งสัญญาณรูปสี่เหลี่ยมออกที่เอาท์พุตได้
- สามารถตรวจสอบระบบไฟและสลับไปใช้แบตเตอรี่ได้โดยอัตโนมัติ
- เมื่อพลังงานจากแบตเตอรี่จะใช้ไฟน้อยกว่า 500 nA ที่ 25 °C
- สามารถเลือกรุ่นที่ใช้งานในอุตสาหกรรมได้โดยสามารถใช้อุณหภูมิในช่วง -40 ถึง +85 °C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

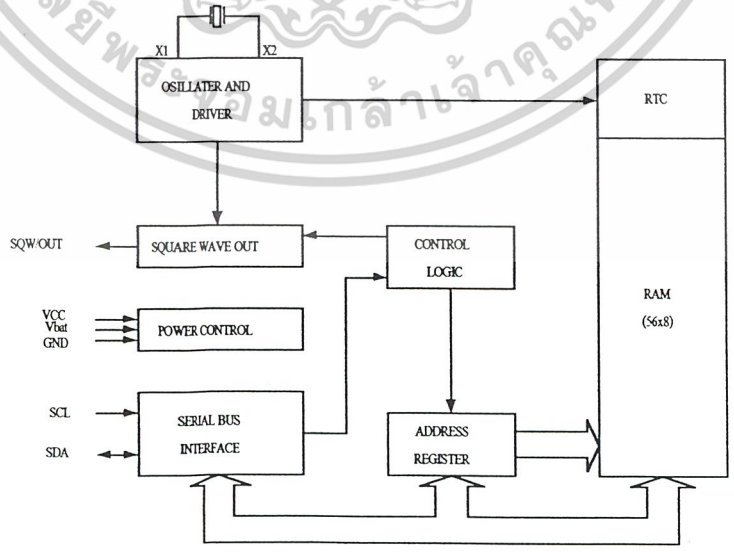
2.5.2 รายละเอียดของขาต่อใช้งาน

ตำแหน่ง	สัญลักษณ์	รายละเอียด
1	X1	ขาต่อ Xtal 32.768 KHz
2	X2	ขาต่อ Xtal 32.768 KHz
3	VBAT	ขาต่อแบตเตอรี่ 3 V
4	GND	ขาต่อกราวด์
5	SDA	สัญญาณ Data แบบอนุกรม
6	SCL	สัญลักษณ์ Clock แบบอนุกรม
7	SQW/OUT	สัญญาณเอาท์พุตเป็นแบบสี่เหลี่ยมหรือเป็นลอจิก
8	Vcc	ต่อเข้ากับแหล่งจ่ายไฟ

ตารางที่ 2.11 รายละเอียดของขาสัญญาณ

2.5.3 การทำงาน

DS1307 เป็นอุปกรณ์ที่ต่ออยู่ในระบบบัสแบบ I^2C โดยทำตัวเป็น Slave การเข้าถึงข้อมูลภายใน สามารถทำได้โดยการส่งเงื่อนไข Start เลขประจำตัว และตำแหน่งของรีจิสเตอร์ตามลำดับ ลงบนบัสแบบ I^2C ที่มี DS1307 ต่อรวมอยู่ โดยรีจิสเตอร์ถูกเข้าถึงต่อไปเรื่อย ๆ จนกระทั่งเกิดเงื่อนไข Stop ในระบบบัสเมื่อแรงดันที่ขา Vcc ตกลงต่ำกว่าแรงดันที่ขา Vbat จะสลับตัวเองเข้าสู่โหมดประหยัดพลังงาน และใช้พลังงานจากแบตเตอรี่ แต่ในทางกลับกัน DS1307 จะกลับไปทำงานในโหมดปกติเมื่อแรงดันที่ขา Vcc สูงกว่าแรงดันที่ขา Vbat +0.2 V โดยไดอะแกรมในรูปที่ 2.27 แสดงถึงส่วนประกอบหลักของ DS1307

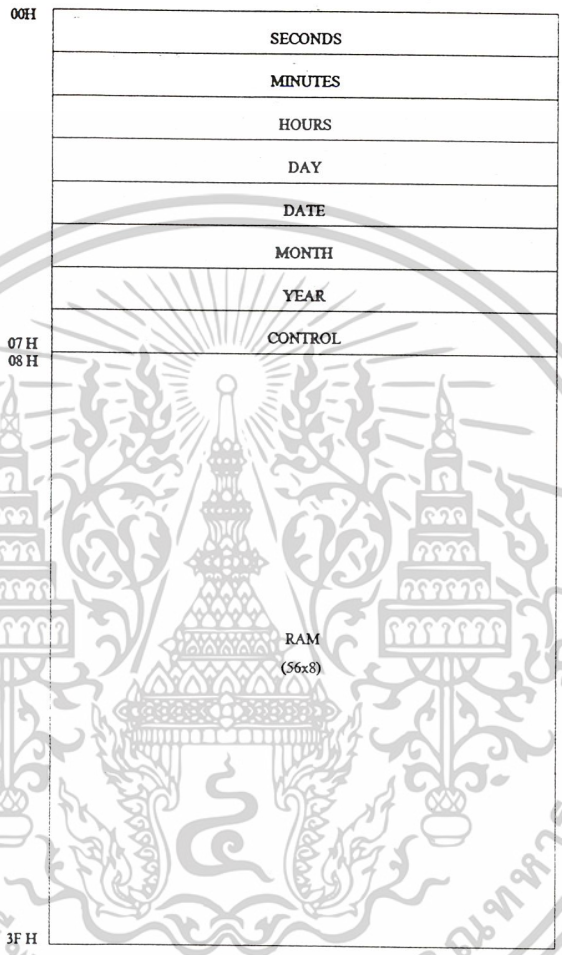


รูปที่ 2.27 แสดงส่วนประกอบหลักของ DS1307

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.4 RTC and Address Map

ตำแหน่งของนาฬิกาและ RAM ของ DS1307 แสดงในรูปที่ 2.28 โดยตำแหน่งของนาฬิกาอยู่ที่ 00H ถึง 07H โดยตำแหน่งของ RAM อยู่ที่ตำแหน่ง 08H ถึง 3FH เมื่อมีการเข้าถึงหน่วยความจำแบบ Multibyte ถ้าการเข้าถึงหน่วยความจำถึง 3FH และจะกลับไปที่ 00H ซึ่งเป็นตำแหน่งเริ่มต้น



รูปที่ 2.28 แสดงตำแหน่งของนาฬิกาและ RAM ของ DS1307

2.5.5 DS1307 Timekeeping Register

- Control Register รีจิสเตอร์ควบคุมเป็นส่วนที่ใช้ควบคุมขาสัญญาณ SQW/OUT ถ้าบิต 4 หรือบิต SQWE มีลอจิก 1 จะทำให้ขา SQW/OUT สร้างสัญญาณพัลส์ออกมาตามความถี่ที่กำหนดในบิต RS0 และ RS1 แต่ถ้าบิต 4 มีลอจิก 0 จะทำให้ขามีลอจิกตามบิต 7 หรือบิต OUT

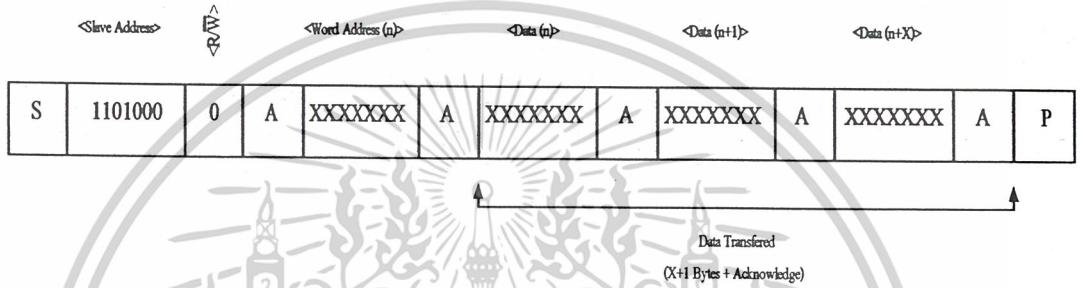
- Squarewave Output Frequency การกำหนดค่าความถี่ที่ต้องการให้ออกที่ขา SQWE มีลอจิก 1 สามารถกำหนดได้จากบิต RS0 และ RS1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.6 การเขียนข้อมูลลง DS1307 (Data Write)

การเขียนข้อมูลลงใน DS1307 จะถูกนำมาใช้เมื่อต้องการตั้งเวลา การกำหนดให้สัญญาณ Pulse ออกที่ขา SQW/OUT หรือแม้กระทั่งการเขียนเข้าไปเก็บไว้ในหน่วยความนำส่วน RAM ที่อยู่ภายใน DS1307

แนวทางการเขียนข้อมูลเข้าใน DS1307 นั้นจะใช้หลักการของการสื่อสารข้อมูลแบบ I^2C คือ การเขียนจะต้องเริ่มต้นจากที่ไมโครคอนโทรลเลอร์ส่งเงื่อนไข Stat ไปให้กับ DS1307 แล้วจึงส่งข้อมูลต่อไปอีก 1 ไบต์ โดยข้อมูลไบต์นี้ใช้เก็บ Address และบิต R/W



รูปที่ 2.29 แสดง Diagram การเขียนข้อมูลลง DS1307

DS1307 ถูกกำหนดให้มีตำแหน่ง Address อยู่ที่ 110100B (ขนาด 7 บิต) ซึ่งในการเขียนข้อมูลจะต้องกำหนดให้บิต R/W เป็น 0 เพราะฉะนั้นเมื่อรวม Address ขนาด 7 บิต เข้ากับบิต R/W จะได้ข้อมูลไบต์ที่จะต้องส่งเป็น 11010000B

หลังจากที่ส่งไบต์ Address ไปให้กับ DS1307 แล้วถ้า DS1307 รับข้อมูลได้ถูกต้อง ตัว DS1307 จะส่งบิต ACK ออกมาให้กับไมโครคอนโทรลเลอร์ซึ่งบิต ACK ที่ DS1307 ส่งออกมาให้กับไมโครคอนโทรลเลอร์จะเป็น 0 แต่ถ้าไมโครคอนโทรลเลอร์ตรวจสอบบิตนี้แล้วเป็น 1 แสดงว่า DS1307 ไม่ได้รับตำแหน่ง Address ที่ถูกต้อง ต้องกลับไปเริ่มขบวนการ Start ใหม่

หลังจากที่ไมโครคอนโทรลเลอร์ได้รับบิต ACK ตอบกลับออกมาจาก DS1307 แล้ว ตัวไมโครคอนโทรลเลอร์จะต้องส่งข้อมูลไปอีก 1 ไบต์ โดยข้อมูลไบต์นี้จะเป็นตำแหน่งหน่วยความจำภายใน DS1307 ซึ่งอยู่ที่ตำแหน่ง 0-3F การส่งข้อมูลไบต์นี้จะเหมือนกับการส่งข้อมูลในไบต์แรก คือ เมื่อ DS1307 ได้รับข้อมูลถูกต้องแล้วตัว DS1307 จะส่งบิต ACK กลับออกมา

หลังจากที่ DS1307 ได้รับตำแหน่งของหน่วยความจำที่ต้องการเขียนเรียบร้อยแล้ว ไมโครคอนโทรลเลอร์สามารถส่งข้อมูลที่ต้องการเขียนไปยังตำแหน่งดังกล่าวได้เลย ซึ่งเมื่อส่งข้อมูลได้ออกไป 1 ไบต์ตัว DS1307 จะเพิ่มค่าตำแหน่ง 1 ค่าและส่งบิต ACK กลับออกมา

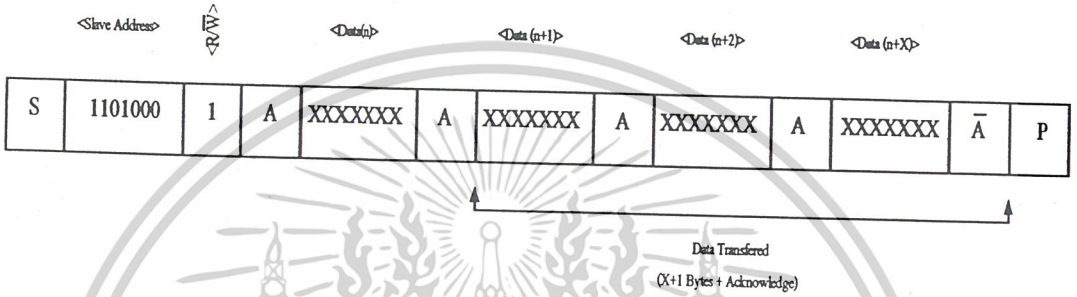
ถ้าต้องการเขียนข้อมูลในตำแหน่งที่เรียงกันไปสามารถทำได้โดยการส่งข้อมูลต่อไปเรื่อย ๆ จนครบแล้วจึงส่งเงื่อนไข STOP เพื่อหยุดการทำงานแต่ถ้าหากต้องการเปลี่ยนตำแหน่งที่จะติดต่อหรือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไม่ทราบว่าตัวชี้ตำแหน่งของ DS1307 ชี้อยู่ที่ตำแหน่งใดก็สามารถทำได้โดยการส่งตำแหน่งใหม่ให้กับ DS1307

2.5.7 การอ่านข้อมูลจาก DS1307 (Data Read)

การอ่านข้อมูลจาก DS1307 จะมีลักษณะคล้ายกับการเขียนข้อมูลให้กับ DS1307 โดยจะต้องให้บิต R/W เป็น 1 เพราะฉะนั้นข้อมูลไบต์ Address จะมีค่าเป็น 11010001B

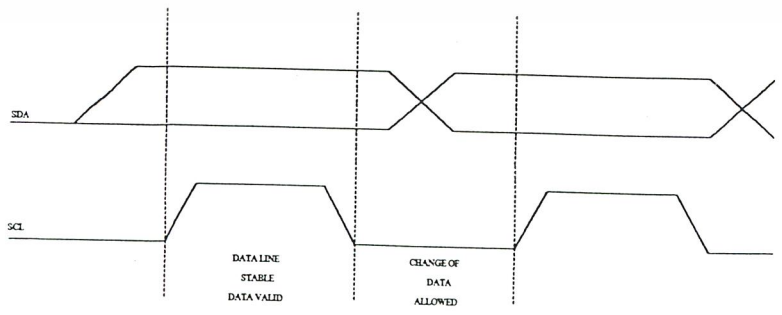


รูปที่ 2.30 แสดง Diagram การอ่านข้อมูลออกจาก DS1307

2.5.8 I^2C

การสื่อสารข้อมูลด้วยระบบ I^2C บัสเป็นอีกมิติหนึ่งของการสื่อสารระหว่างไมโครคอนโทรลเลอร์และอุปกรณ์ประกอบร่วม ระบบ I^2C เป็นระบบที่ถูกพัฒนาจากบริษัท PHILIPS ดังนั้นอุปกรณ์หลาย ๆ ตัวที่มีการสื่อสารแบบ I^2C จึงถูกผลิตออกมาจากบริษัท PHILIPS ระบบ I^2C บัส เป็นการสื่อสารแบบ 2 ทาง โดยใช้สายสัญญาณในการสื่อสารเพียง 2 เส้น โดยสายที่ใช้สื่อสารนี้คือ SDA ซึ่งเป็นสายสัญญาณข้อมูลและ SCL ซึ่งทำให้การใช้งานไมโครคอนโทรลเลอร์มีประสิทธิภาพมาก ในแง่ของความสิ้นเปลืองพอร์ตของไมโครคอนโทรลเลอร์ ทางบริษัทผู้พัฒนาระบบ I^2C ได้ให้คำจำกัดความของการสื่อสารแบบ I^2C ไว้เพื่อให้การศึกษาเกิดความเข้าใจเดียวกัน โดยมีรายละเอียดต่างๆ ดังนี้

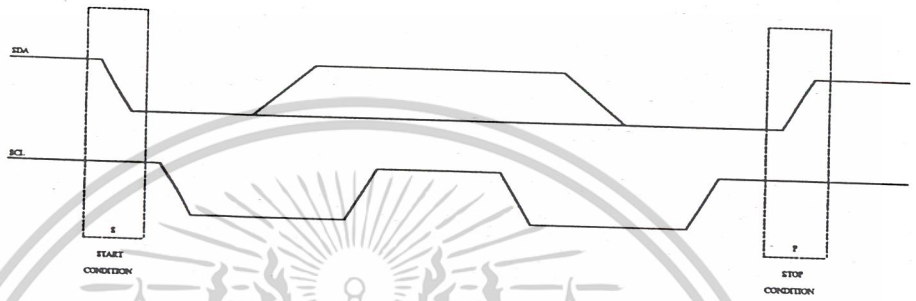
- Bit Transfer ข้อมูล 1 บิตจะถูกส่งออกไปด้วยช่วงเวลา 1 CLOCK โดยข้อมูลที่สาย SDA จะต้องคงที่ในขณะที่ CLOCK เป็นลอจิก 1



รูปที่ 2.31 แสดง Timing Diagram ของ Bit Transfer

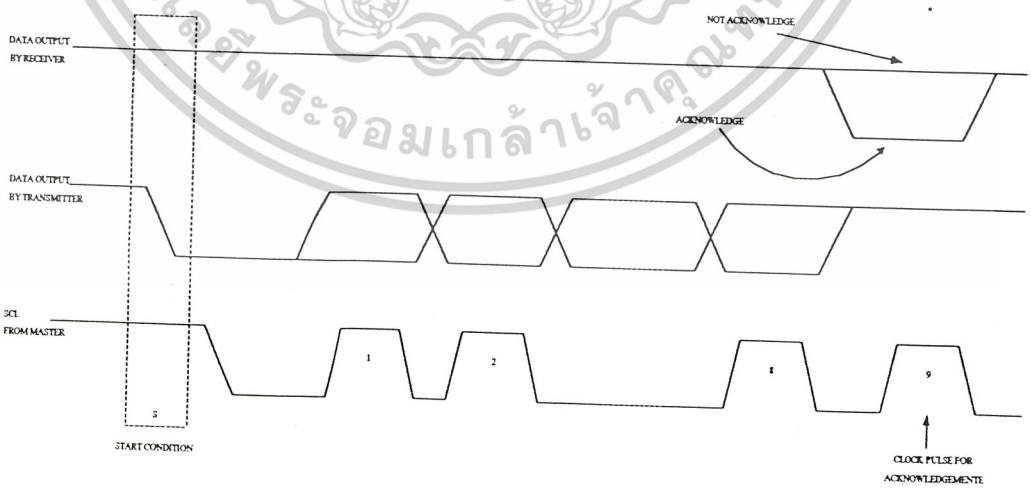
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Start and Stop Conditions ทั้งสายสัญญาณ SDA และสายสัญญาณ SCL ถ้าอยู่ในสถานะไม่ Busy จะเป็นลอจิก 1 การเปลี่ยนแปลงจากลอจิก 1 เป็นลอจิก 0 ของสายสัญญาณ SDA ขณะที่สายสัญญาณ SCL เป็น 1 เรียกว่าการกำหนดเงื่อนไข START แต่การเปลี่ยนแปลงจาก 0 เป็น 1 ของสายสัญญาณ SDA ในขณะที่สายสัญญาณ SCL เป็น 1 เรียกว่าการกำหนดเงื่อนไข STOP



รูปที่ 2.32 แสดง Timing Diagram ของ Start and Stop Conditions

- System configuration อุปกรณ์ที่ส่งข้อมูลเรียกว่า Transmitter ส่วนอุปกรณ์ที่รับข้อมูลเรียกว่า Receiver และอุปกรณ์ที่ใช้ควบคุมทิศทางการสื่อสารข้อมูลเรียกว่า Master และอุปกรณ์ที่ถูกควบคุมจาก Master เรียกว่า Slave
- Acknowledge จำนวนไบต์ของข้อมูลที่ถูกส่งระหว่างตัวรับและตัวส่งมีไม่จำกัดซึ่งเมื่อส่งข้อมูลครบ 1 ไบต์จะต้องส่งบิต ACK ตามออกไป 1 บิต



รูปที่ 2.33 แสดง Timing Diagram ของ Acknowledge

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

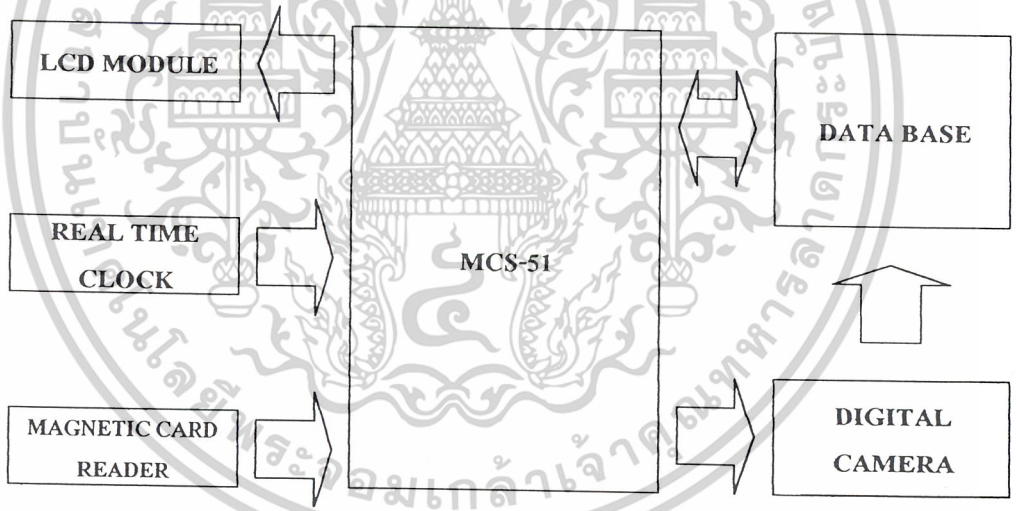
บทที่ 3
การคำนวณและการสร้าง

3.1 แนวความคิดในการสร้างระบบตรวจสอบการเข้าออกสำนักงาน

โครงงานนี้เป็นการสร้างระบบตรวจสอบสำนักงาน ซึ่งการตรวจสอบของระบบจะแบ่งเป็นสองส่วนคือ

- พนักงาน ระบบจะทำการตรวจสอบข้อมูลจากบัตรประจำตัวพนักงานซึ่งเป็นบัตรแถบแม่เหล็ก ถ้าข้อมูลจากบัตรถูกต้อง ระบบจะทำการบันทึกภาพพนักงาน และแจ้งการตรวจสอบบัตรผ่านพนักงานทางจอแสดงผล LCD
- ส่วนผู้มาติดต่อสำนักงาน ในส่วนนี้การปลดล็อคประตูจะเป็นหน้าที่ ของประชาสัมพันธ์ของสำนักงานในการพิจารณาอนุญาตให้เข้าสำนักงาน ซึ่งถ้าประชาสัมพันธ์ปลดล็อค ระบบจะทำการบันทึกสัญญาณภาพและเวลาการเข้ามาติดต่อไว้ในฐานข้อมูล

3.2 ส่วนประกอบของระบบ



รูปที่ 3.1 บล็อกไดอะแกรมแสดงระบบตรวจสอบการเข้าออกสำนักงาน

1. ส่วนรับข้อมูล ข้อมูลที่รับเข้ามาแบ่งออกเป็นสองส่วน คือ ข้อมูลของพนักงาน จากบัตรพนักงาน ซึ่งรับมาจากเครื่องอ่านบัตรแม่เหล็ก และข้อมูลภาพซึ่งระบบจะรับข้อมูลภาพผ่านทางกล้องวิดีโอดิจิตอล ซึ่งข้อมูลภาพจะถูกเก็บในส่วนฐานข้อมูล
2. ส่วนประมวลผล (Processing) ในส่วนการประมวลผลแบ่งออกเป็นสามส่วน คือ
   - ส่วนประมวลผลเวลาจริง ซึ่งรับข้อมูลจากส่วนกำเนิดสัญญาณเวลาจริงมาประมวลผลเพื่อแสดงผลแจ้งพนักงานให้ทราบถึงเวลาเข้าออกสำนักงาน

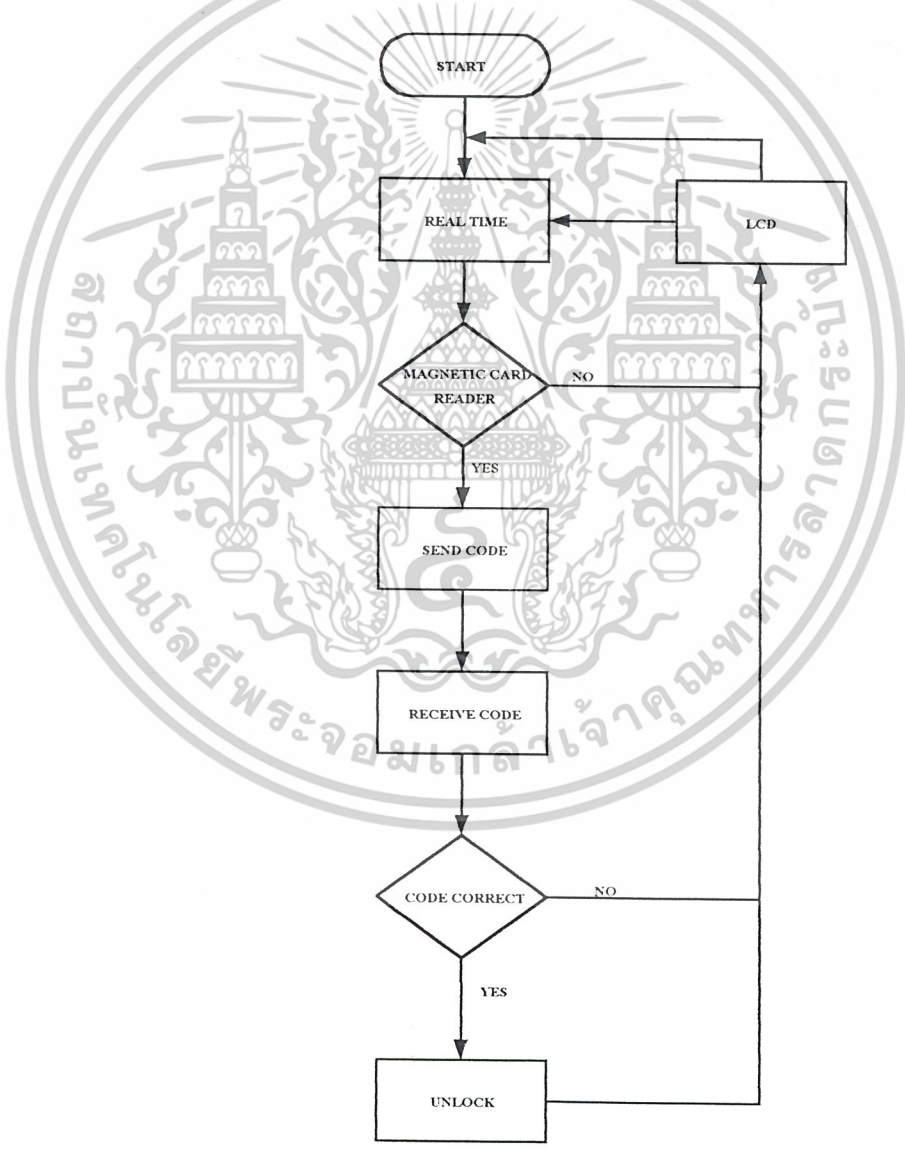
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ส่วนถอดรหัสสัญญาณจากบัตรแม่เหล็ก ตัวประมวลผลจะรับข้อมูลจากเครื่องอ่านบัตรแม่เหล็กมาถอดรหัสข้อมูลส่งให้ฐานข้อมูลเปรียบเทียบและฐานข้อมูลจะส่งผลของการเปรียบเทียบทางจอ LCD เพื่อแสดงผลให้พนักงานทราบ
- ส่วนควบคุมกล้องวีดีโอดิจิตอล ซึ่งในส่วนนี้จำเป็นในส่วนของข้อมูลผู้ที่มาติดต่อ ซึ่งผู้มาติดต่อไม่อยู่ตำแหน่งประจำในการเก็บภาพคือตำแหน่งหน้าเครื่องอ่านบัตร

3. ส่วนฐานข้อมูล (Data Base) เป็นส่วนการเก็บข้อมูลซึ่งประกอบด้วยข้อมูลภาพ และข้อมูลเวลาเข้าออก รวมทั้งทำการเปรียบเทียบข้อมูลจากบัตรแม่เหล็กกับฐานข้อมูล

3.3 การทำงานของระบบ

3.3.1 การทำงานของส่วนประมวลผล



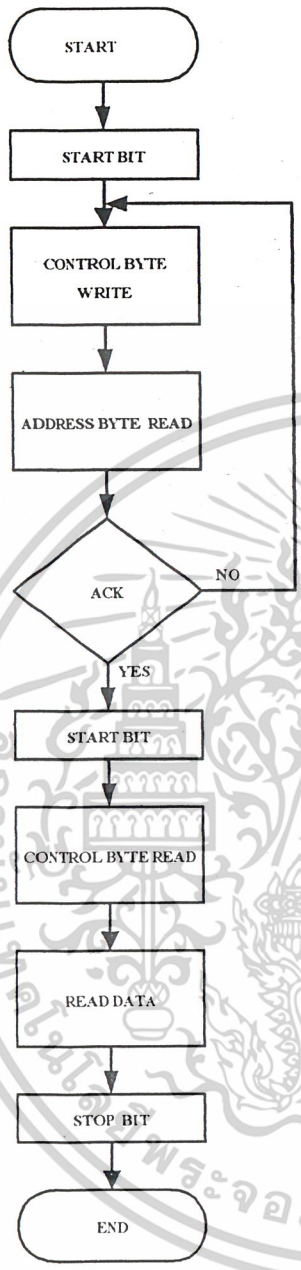
รูปที่ 3.2 โฟลชาร์ตการทำงานของส่วนประมวลผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

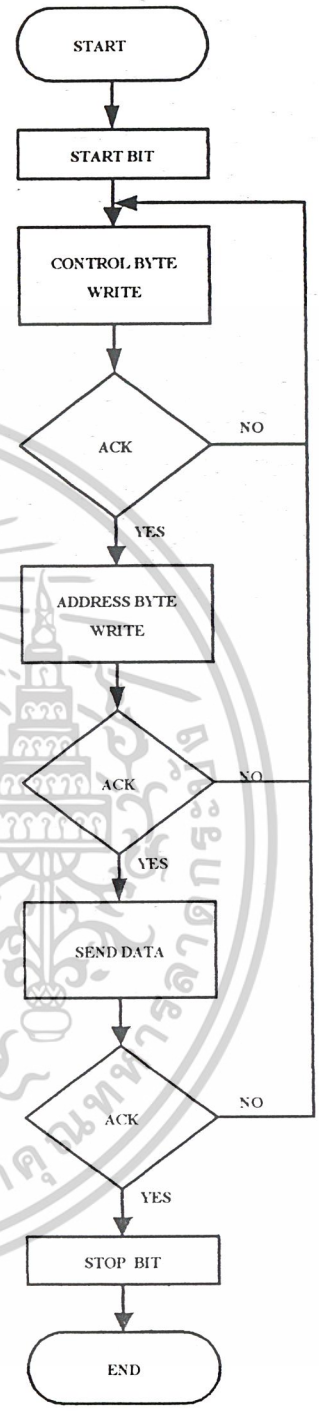
ส่วนการประมวลผลจะใช้ไมโครคอนโทรลเลอร์ตระกูล MCS-51 เบอร์ 89P2051RD2 เป็นตัวประมวลผล ซึ่งการทำงานก็คือเริ่มต้นด้วยโปรแกรมทำการอ่านค่าเวลาจริงจากไอซีเบอร์ DS 1307 จากนั้นทำการตรวจสอบว่ามีการอ่านบัตรแม่เหล็กหรือไม่ หากไม่มีการอ่านบัตรแม่เหล็ก ระบบก็จะทำการแสดงผลเวลาจริงซึ่งแสดงผลเป็น ชั่วโมง นาที วินาที ที่ LCD ถ้ามีการอ่านบัตรแม่เหล็ก ซึ่งในส่วนการอ่านจะมีการตรวจสอบการผิดพลาดของข้อมูล หากข้อมูลผิดพลาดก็จะแสดงผลที่ LCD ว่ามีการผิดพลาดเกิดขึ้นจากการรูดบัตร หากข้อมูลถูกต้องข้อมูลจะถูกส่งไปเปรียบเทียบกับฐานข้อมูลซึ่งหากเปรียบเทียบกับฐานข้อมูลแล้วผิดพลาด คือไม่มีข้อมูลของบัตรแม่เหล็กบัตรนี้อยู่ในฐานข้อมูล ฐานข้อมูลก็จะส่งข้อมูลมาแจ้งให้ส่วนประมวลผลทราบ ซึ่งส่วนประมวลผลก็จะแสดงผลการอ่านบัตรแม่เหล็กที่ไม่มีในฐานข้อมูลที่ LCD หากข้อมูลในบัตรแม่เหล็กตรงกับฐานข้อมูล ฐานข้อมูลก็จะส่งข้อมูลมาบอกส่วนประมวลผลว่าบัตรที่ใช้รูดถูกต้อง และทำการเก็บค่าวันที่ เวลา และรูปที่ถ่ายจากกล้องดิจิตอลลงในฐานข้อมูล แล้วระบบก็จะสั่งให้ ปลดล็อกประตูเพื่อให้พนักงานสามารถเข้ามาทำงานได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 การตั้งเวลาจริง และการอ่านค่าเวลาจริง



(ก) การตั้งค่าเวลา



(ข) การอ่านค่าเวลาจริง

รูปที่ 3.3 แสดงโฟลชาร์ตการตั้งค่าเวลาจริง และการอ่านค่าเวลาจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การตั้งค่าเวลาจริงคือการเขียนข้อมูลลงไอซีเบอร์ DS 1307 ซึ่งการเขียนข้อมูลจะใช้หลักการสื่อสารข้อมูลแบบ I^2C ซึ่งในส่วนนี้สามารถดูได้จากหัวข้อ 2.5.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

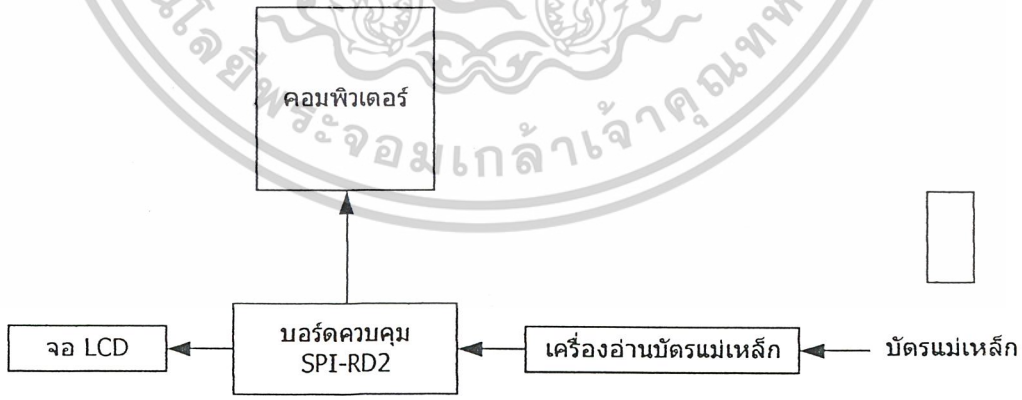
4.1 การทดลองเรื่อง การแสดงผลของค่าเวลาจริง และข้อมูลของบัตรแม่เหล็กผ่านจอแสดงผล LCD

อุปกรณ์การทดลอง

1. แหล่งจ่ายไฟ 5 VDC
2. บอร์ด SPI-RD2
3. จอแสดงผล LCD
4. โปรแกรมควบคุม ACCESS.ASM
5. เครื่องอ่านบัตรแม่เหล็ก
6. บัตรแถบแม่เหล็ก
7. คอมพิวเตอร์

การทดลอง

1. ต่อวงจรดังรูปที่ 4.1
2. เชื่อมต่อบอร์ด SPI-RD2 เข้ากับคอมพิวเตอร์ โดยผ่านพอร์ตสื่อสารอนุกรม (RS-232)



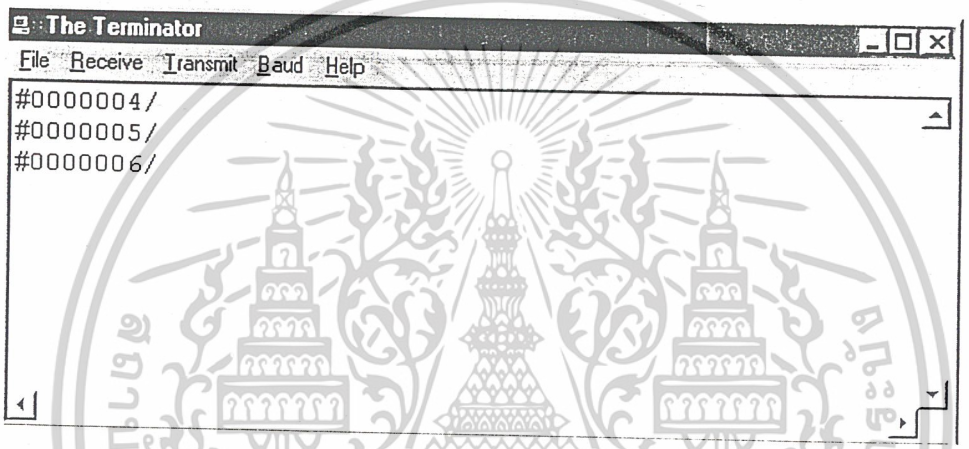
รูปที่ 4.1 แสดงการเชื่อมต่อบอร์ด SPI-RD2 กับคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

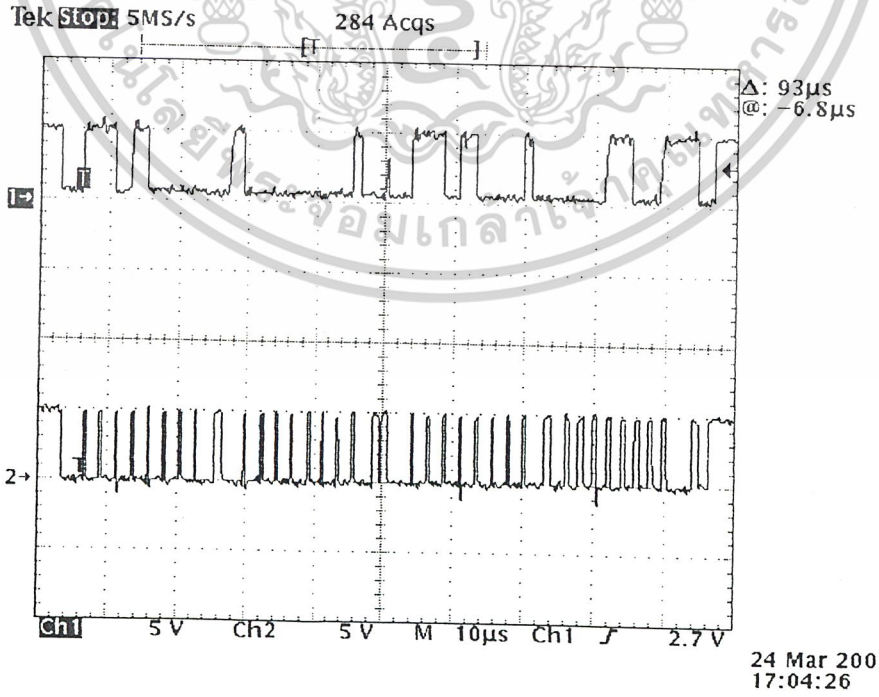
ผลการทดลอง

เมื่อต่อวงจรตามรูปที่ 4.1 ขณะไม่มีการรูดบัตรแม่เหล็ก ผลที่ได้คือที่จอแอลซีดีตำแหน่งตัวอักษรที่หนึ่งถึงหกจะแสดงค่าเวลาจริงเป็นวินาที คือ ชั่วโมง นาที วินาที เมื่อมีการรูดบัตรหากการอ่านข้อมูลถูกต้อง จอแอลซีดีตำแหน่งตัวอักษรที่แปดถึงสิบหกจะแสดงรหัสข้อมูลของบัตรแม่เหล็ก และในส่วนของเวลาจริงก็ยังแสดงผลต่อเนื่อง ถ้าหากข้อมูลจากการอ่านบัตรผิดพลาดที่จอแอลซีดี จะแสดงผลว่าเกิดการผิดพลาดขึ้นคือ ERROR!

การรับสัญญาณเวลาจริงที่สร้างมาจาก IC DS1307 จะประกอบไปด้วยสัญญาณข้อมูล (SDA) และสัญญาณนาฬิกา (SCL) ดังแสดงในรูปที่ 4.3

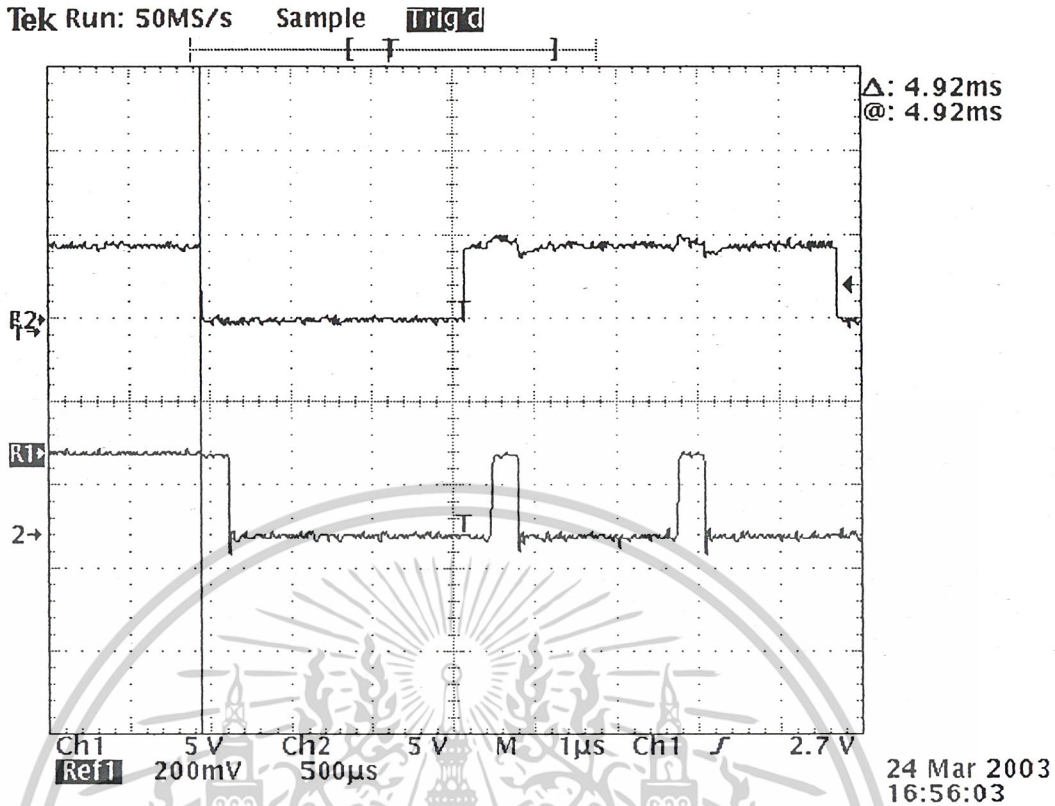


รูปที่ 4.2 แสดงผลข้อมูลของบัตรแม่เหล็กทางพอร์ตอนุกรม

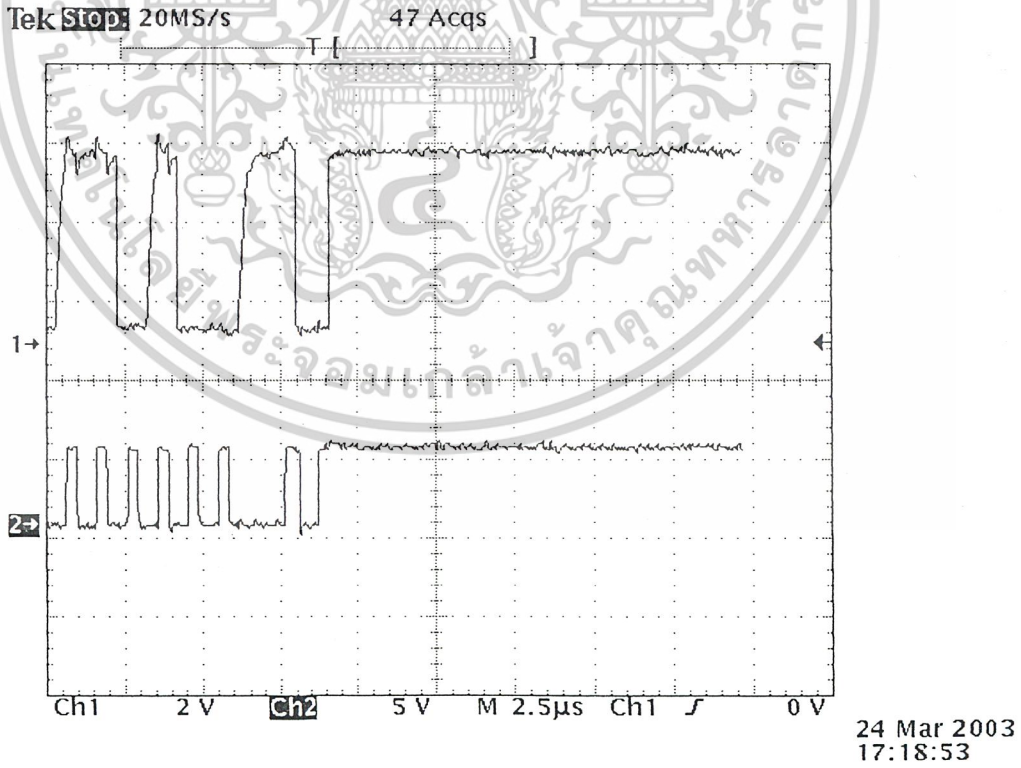


(ก) การแสดงรอบการอ่านข้อมูลจาก DS 1307

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ข) แสดงเงื่อนไข Start Bit ของการใช้งาน DS 1307



(ก) แสดงเงื่อนไข Stop Bit ของการใช้งาน DS1307

รูปที่ 4.3 แสดงสัญญาณข้อมูลและสัญญาณนาฬิกาของ DS1307

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การทดลองเรื่อง โปรแกรมส่วนฐานข้อมูลในการบันทึกเวลาเข้าออกสำนักงาน

อุปกรณ์การทดลอง

1. คอมพิวเตอร์
2. บัตรแถบแม่เหล็ก
3. เครื่องอ่านบัตรแม่เหล็ก

การทดลอง

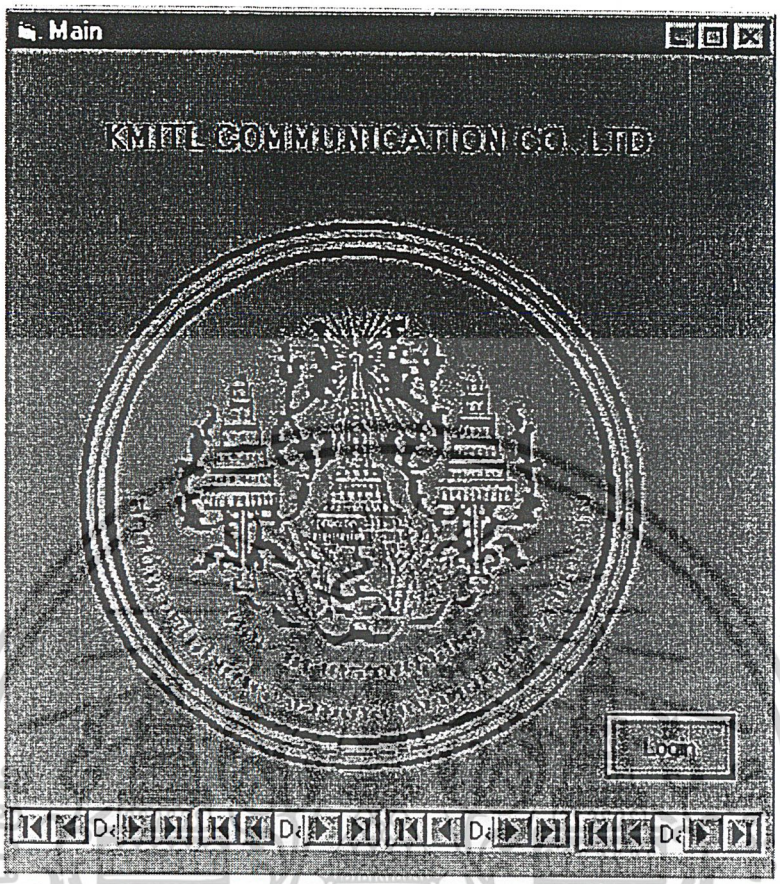
1. RUN โปรแกรม MAIN.VBP
2. รูดบัตรแม่เหล็ก
3. ดูข้อมูลที่เก็บลงในฐานข้อมูล

ผลการทดลอง

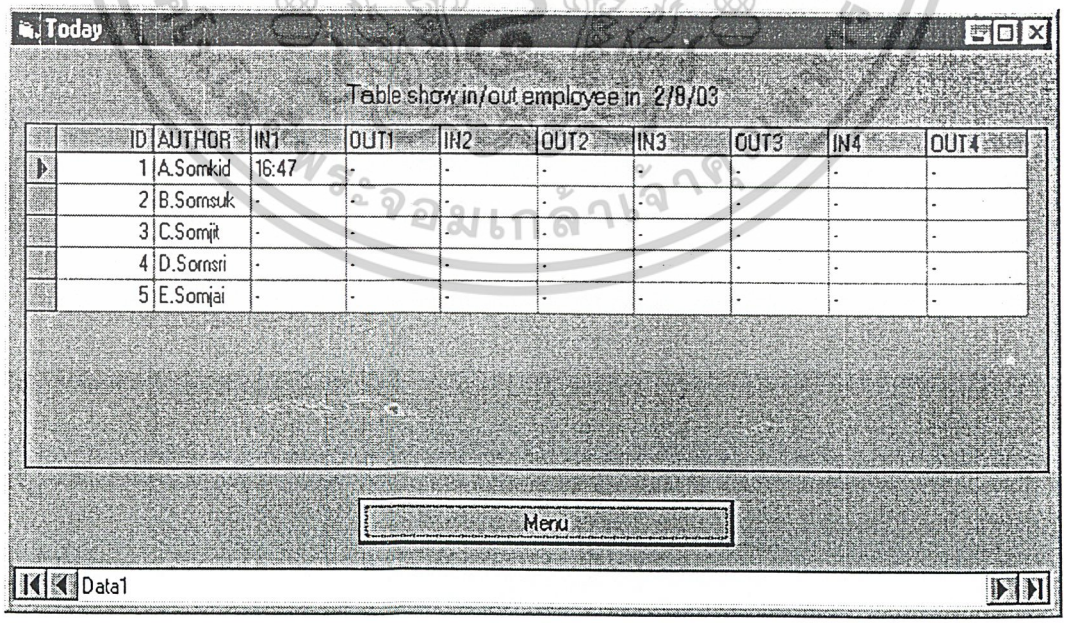
เมื่อทดลองรันโปรแกรม MAIN.VBP จะได้หน้าจอแสดงผลดังรูปที่ 4.4 เมื่อรูดบัตรแม่เหล็ก ซึ่งเป็นบัตรประจำตัวพนักงาน และมีข้อมูลของผู้ถือบัตรอยู่ในฐานข้อมูลจะได้ผลการทดลอง คือมีการบันทึกเวลาการเข้าทำงานของพนักงาน ดังรูปที่ 4.5 และเมื่อรูดบัตรแม่เหล็กของพนักงานอีกคนหนึ่ง ก็จะมีการเก็บค่าเวลาเช่นกัน ดังรูปที่ 4.6

เมื่อทดลองรูดบัตรเป็นครั้งที่สองซึ่งเป็นเวลาออกงานก็จะมีการบันทึกข้อมูลเวลาการออกจากงานของพนักงานแต่ละคนดังรูปที่ 4.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

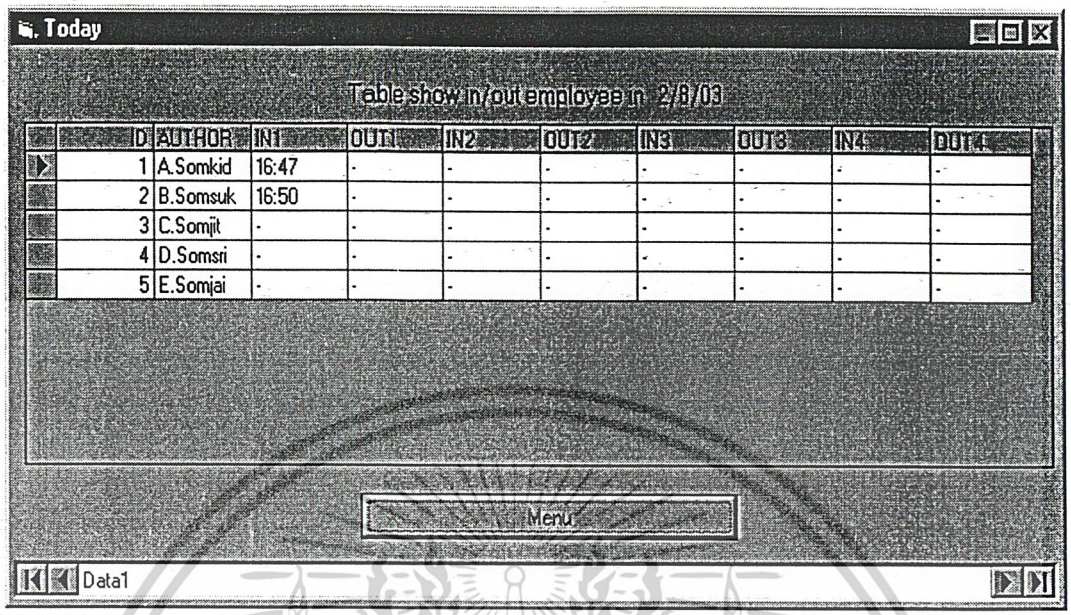


รูปที่ 4.4 แสดงหน้าจอ เมื่อทดลองรันโปรแกรม MAIN.VBP

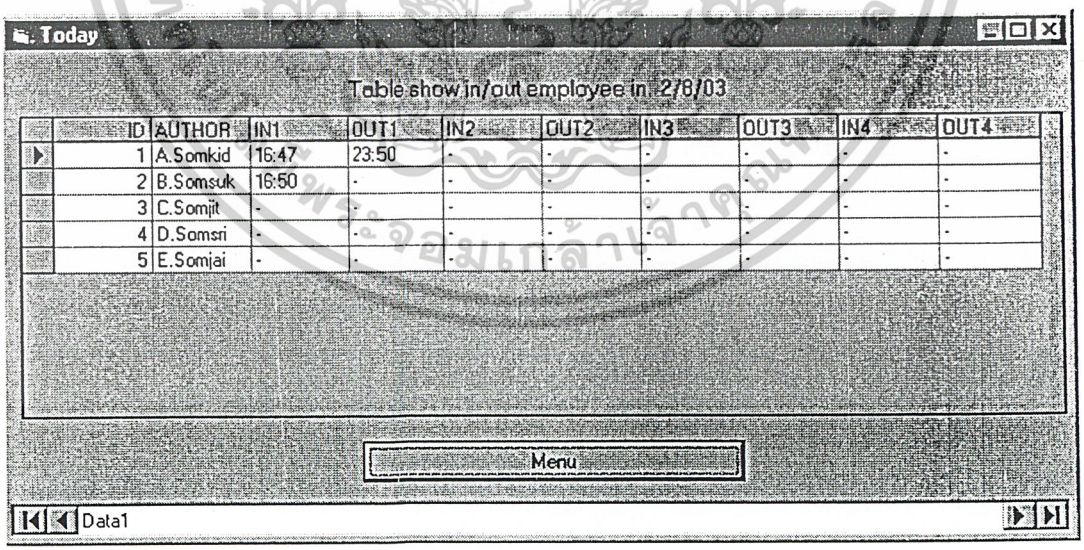


รูปที่ 4.5 แสดงข้อมูลเมื่อมีพนักงานรูดบัตรเข้าทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 แสดงข้อมูลเมื่อรูดบัตรพนักงานอีกคนหนึ่ง



รูปที่ 4.7 แสดงข้อมูลเมื่อพนักงานรูดบัตรออกจากงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การทดลองเรื่อง การตรวจสอบข้อมูลของ ผู้ดูแลและควบคุมระบบ

อุปกรณ์การทดลอง

1. คอมพิวเตอร์
2. บัตรแถบแม่เหล็ก
3. เครื่องอ่านบัตรแม่เหล็ก

การทดลอง

1. RUN โปรแกรม MAIN.VBP
2. คลิก login แล้วใส่ User และ Password
3. เข้าไปที่ Manage
4. ทดลองค้นหาข้อมูลพนักงานจากฐานข้อมูล
5. ทดลองแก้ไขข้อมูลของพนักงาน
6. ทดลองเพิ่มข้อมูลของพนักงาน
7. ทดลองลบข้อมูลของพนักงาน

ผลการทดลอง

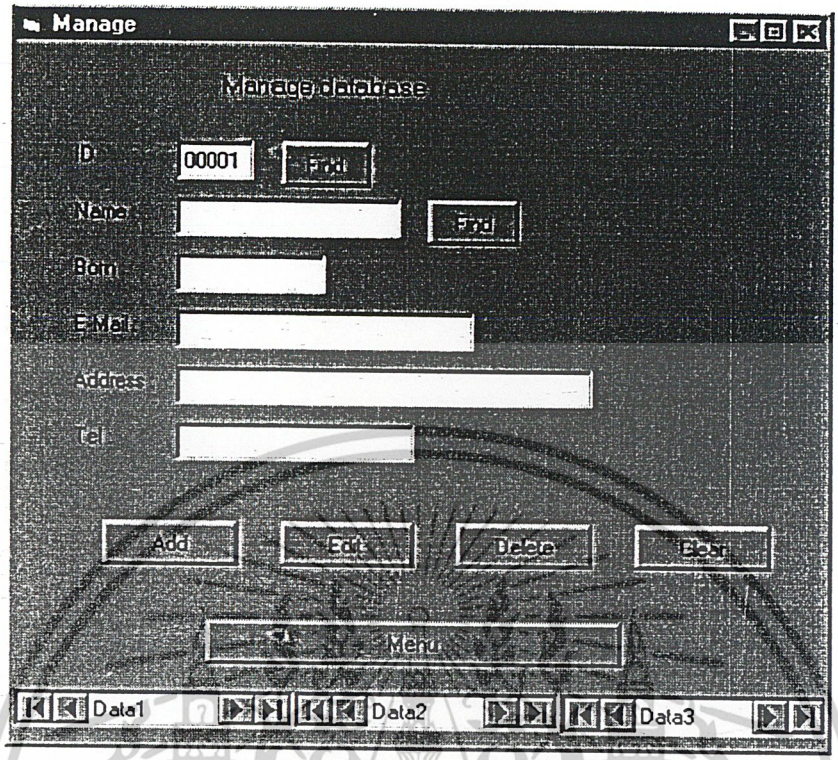
การค้นหาสามารถทำได้ 2 วิธี คือ ค้นหาโดยใช้รหัสของพนักงาน และแบบค้นหาโดยใช้ชื่อของพนักงาน ดังรูปที่ 4.8 และ 4.9

การแก้ไขข้อมูล ขั้นตอนแรกต้องทำการค้นหาข้อมูลก่อน โดยวิธีไหนก็ได้ หลังจากนั้นจึงแก้ไขข้อมูลตามที่ต้องการแล้วคลิกที่ Edit ก็จะสามารถบันทึกข้อมูลที่แก้ไขลงในฐานข้อมูล ดังรูปที่ 4.10

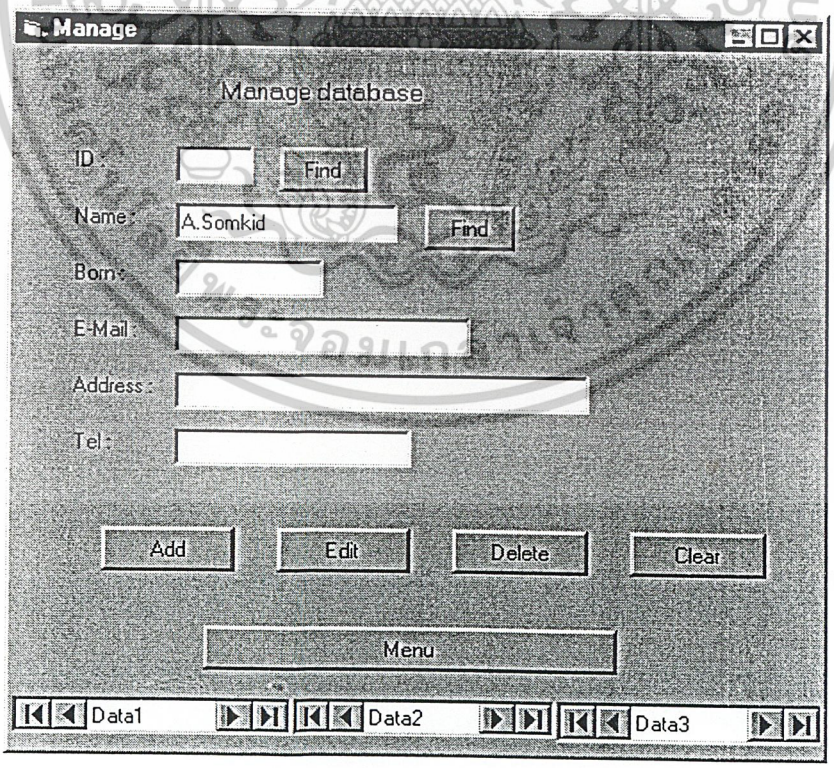
การเพิ่มข้อมูล สามารถทำได้โดยคลิกที่ Add หลังจากนั้นก็กรอกข้อมูล รวมทั้งเลือกรูป ของพนักงานลงไป แล้วก็คลิก Save โปรแกรมก็รันรหัสของพนักงาน และ Save ข้อมูลของพนักงานลงในฐานข้อมูล ดังรูปที่ 4.11

การลบข้อมูล ขั้นตอนแรกต้องทำการค้นหาข้อมูลก่อน แล้วก็คลิก Delete โปรแกรมก็จะทำการลบข้อมูลออกจากฐานข้อมูล ดังรูปที่ 4.12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

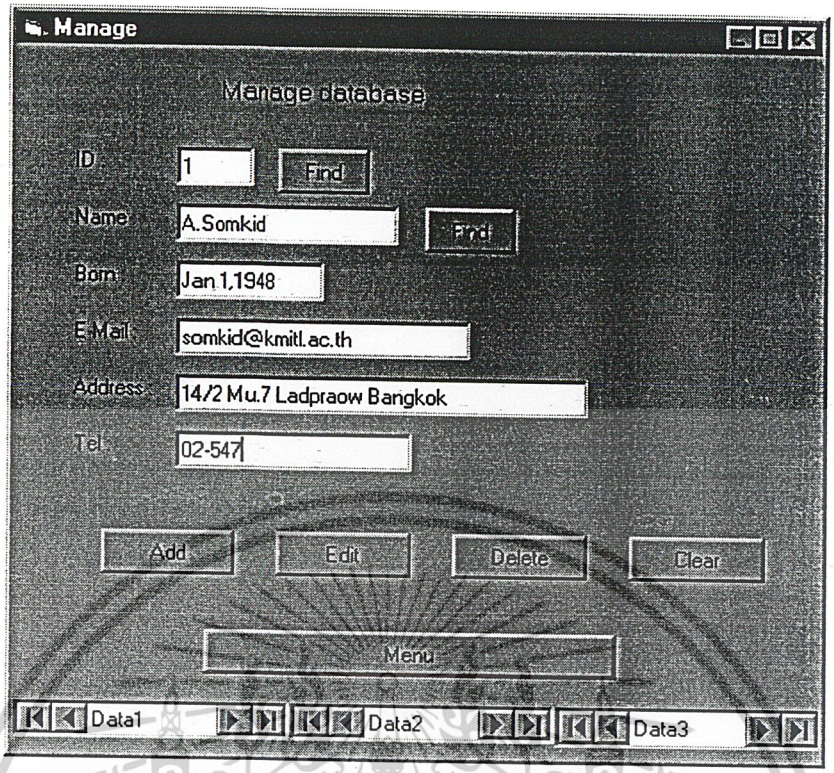


รูปที่ 4.8 ค้นหาข้อมูลของพนักงานโดยใช้รหัสพนักงาน

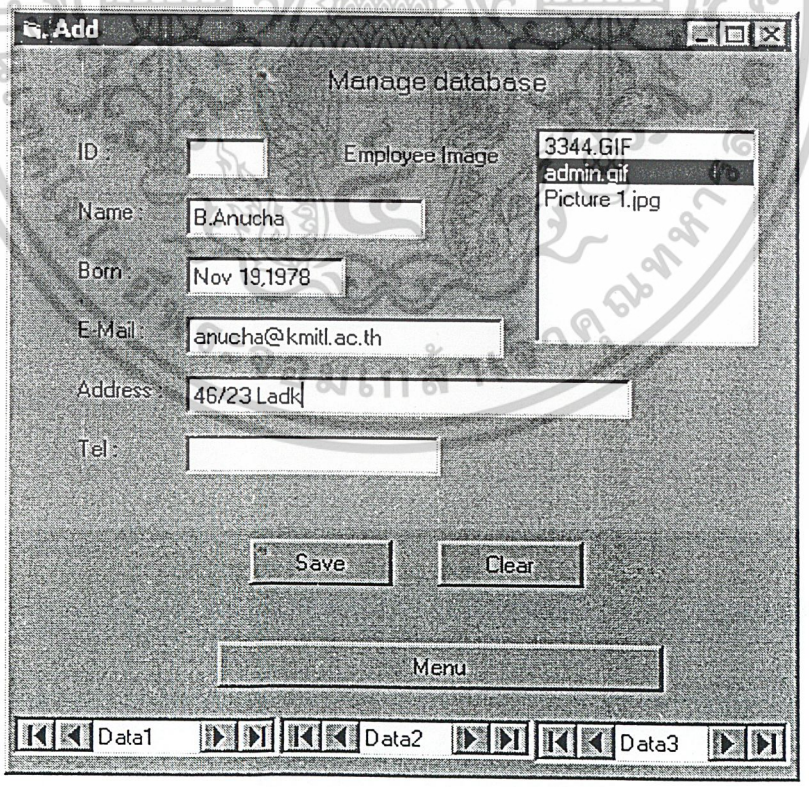


รูปที่ 4.9 ค้นหาข้อมูลของพนักงานโดยใช้ชื่อของพนักงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

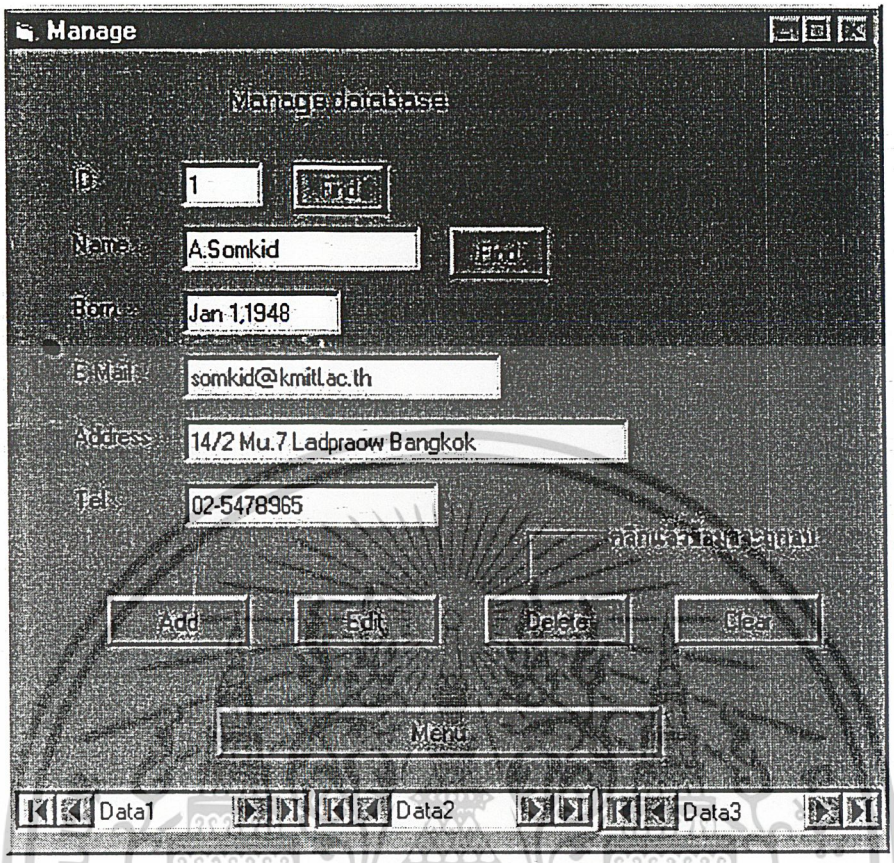


รูปที่ 4.10 แสดงการแก้ไขข้อมูล



รูปที่ 4.11 แสดงการเพิ่มข้อมูลของพนักงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.12 แสดงการลบข้อมูลของพนักงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 การทดลองเรื่อง การทำงานของกล้องดิจิตอล

อุปกรณ์การทดลอง

1. คอมพิวเตอร์
2. บัตรแถบแม่เหล็ก
3. เครื่องอ่านบัตรแม่เหล็ก
4. กล้องดิจิตอล

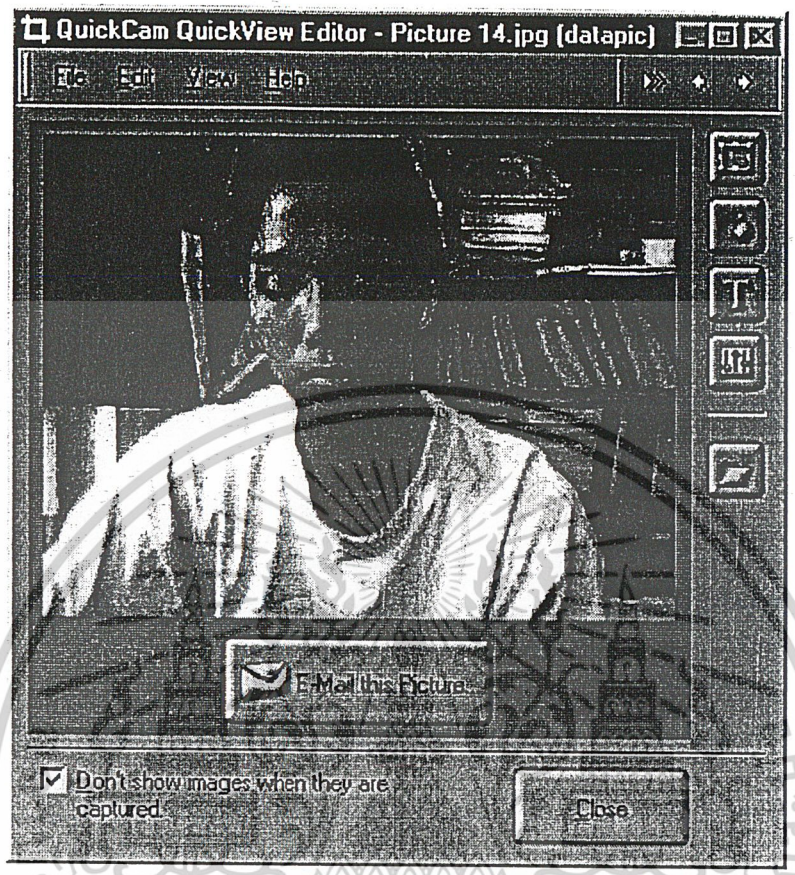
การทดลอง

1. RUN โปรแกรม MAIN.VBP
2. เปิดโปรแกรม Launch Logitech Quick Cam
3. รูดบัตรแถบแม่เหล็ก
4. ตรวจดูในโปรแกรม Launch Logitech Quick Cam ว่ามีรูปที่ถ่ายหรือไม่

ผลการทดลอง

เมื่อรันโปรแกรม MAIN.VBP แล้ว หลังจากที่รูดบัตรแถบแม่เหล็ก โปรแกรมจะสั่งให้กล้องดิจิตอล ถ่ายรูปของผู้ที่รูดบัตร และสามารถเข้าไปตรวจสอบรูปที่ถ่ายมา ได้ที่โปรแกรม Launch Logitech Quick Cam ดังรูปที่ 4.13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.13 แสดงการถ่ายภาพของกล้องดิจิตอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทวิจารณ์และบทสรุป

โครงงานนี้เป็นการศึกษาการทำงานของเครื่องอ่านบัตรแม่เหล็ก และการส่งข้อมูลผ่าน RS-232 ไปยังคอมพิวเตอร์ โดยจะเขียนโปรแกรม Visual Basic รองรับข้อมูลที่ส่งมา แล้วนำไปประมวล เพื่อส่งข้อมูลกลับไปยังไมโครคอนโทรลเลอร์ และเก็บลงในฐานข้อมูล โดยที่ใช้โปรแกรม Microsoft Access เป็น Database

เมื่อมีการรูดบัตรแม่เหล็ก เครื่องอ่านบัตรแม่เหล็กจะอ่านข้อมูลในบัตรแม่เหล็ก โดยมีไมโครคอนโทรลเลอร์เป็นตัวรับค่าจากเครื่องอ่านบัตรแม่เหล็ก แล้วส่งไปให้คอมพิวเตอร์ประมวลผลว่าบัตรแม่เหล็กที่ใช้รูดนั้น เป็นบัตรที่ถูกต้องหรือไม่ ถ้าถูกต้องก็จะเก็บข้อมูลการมาทำงานของพนักงานลงในฐานข้อมูล แล้วแสดงข้อมูลผ่านจอ LCD ว่าข้อมูลถูกเก็บเรียบร้อยแล้ว หากไม่ถูกต้องก็จะไม่มีการเก็บข้อมูลลงในฐานข้อมูล และจะแจ้งผ่านจอ LCD ว่าบัตรที่ใช้รูดไม่ถูกต้อง

จากการทดลองในส่วนของไมโครคอนโทรลเลอร์นั้น ยังมีปัญหาเกิดขึ้นในส่วนของโปรแกรมแสดงผลเวลาจริง (Real Time Clock) ซึ่งโปรแกรมจะต้องแสดงผลเวลาจริงตลอดเวลา แต่จากการทดลองพบว่ายังมีเปอร์เซ็นต์ความผิดพลาดอยู่บ้าง ก็คือบางครั้งเมื่อเปิดสวิตช์ใช้งาน เวลาที่แสดงขึ้นมาบนหน้าจอ LCD อาจจะไม่ตรงกับเวลาจริง แต่จะไม่ส่งผลต่อการเก็บข้อมูลของโปรแกรม Visual Basic เพราะเนื่องจากโปแกรม Visual Basic จะเก็บข้อมูลเวลาเมื่อมีการรูดบัตรจากตัวไบออสของคอมพิวเตอร์เอง ไม่ใช่เก็บค่าจากโปรแกรมแสดงผลเวลาจริง โดยโปรแกรมแสดงผลเวลาจริงมีไว้แสดงผ่านจอ LCD ให้ผู้ที่รูดบัตรทราบว่าขณะนี้เป็นเวลาเท่าไร

ในส่วนของโปรแกรมวิชวลเบสิกนั้น จากการทดลองสามารถใช้งานได้ทุกฟังก์ชันการใช้งานตามจุดประสงค์ที่ต้องการ ก็คือสามารถที่จะเก็บข้อมูลการมาทำงานของพนักงาน และสามารถที่จะตรวจสอบข้อมูลได้ทั้งในวันปัจจุบัน และประวัติข้อมูลการมาทำงาน สามารถที่จะเพิ่มข้อมูล ลบข้อมูล และแก้ไขข้อมูลของพนักงานในฐานข้อมูลได้ และสุดท้ายก็คือสามารถแสดงภาพขณะรูดบัตรของพนักงานและแสดงขึ้นมาให้เห็นได้

ปัญหาของโปรแกรม Visual Basic ที่เกิดขึ้น เกิดในส่วนของการประมวลผลเพื่อตรวจสอบว่าบัตรแม่เหล็กที่รูดนั้นถูกต้องหรือไม่ การทำงานของส่วนนี้ก็คือ เมื่อมีการรูดบัตรแม่เหล็ก โปรแกรมจะเก็บค่า ID ในบัตรมาค้นหาในฐานข้อมูลพนักงานว่ามี ID เบอร์นี้อยู่ในฐานข้อมูลหรือไม่ ถ้าหากว่าไม่มีโดยปกติ ฐานข้อมูลจะไปชี้อยู่ในตำแหน่ง Record ที่1 ซึ่งก็คือ ID เบอร์ 1 นั่นเอง ดังนั้นเมื่อฐานข้อมูลส่งค่า ID ที่มันชี้อยู่กลับมาเปรียบเทียบกับ ID ที่รับเข้ามานั้นก็จะไม่เท่ากัน โปรแกรมก็จะส่งข้อมูลไปบอกไมโครคอนโทรเลอร์ว่าบัตรที่ใช้รูดนั้นไม่ถูกต้อง และโปรแกรมจะไม่เก็บข้อมูลในการรูดครั้งนี้ ลงในฐานข้อมูล แต่ถ้าหากว่า ID ที่รับเข้ามามีอยู่ในฐานข้อมูล เมื่อฐานข้อมูลค้นหาเจอก็จะชี้ตำแหน่งอยู่ที่ Record นั้น และส่งค่า ID ของ Record ที่ชี้อยู่ กลับมาเปรียบเทียบกับ ID ที่รับเข้ามาซึ่งก็จะมีค่าเท่ากัน โปรแกรมก็จะส่ง ข้อมูลไปบอกไมโครคอนโทรลเลอร์ว่าบัตรแม่เหล็กที่ใช้รูดนั้นถูกต้อง และโปรแกรมจะเก็บวันที่ เวลา และรูปขณะที่ทำการรูดบัตรซึ่งถ่ายโดยกล้องดิจิตอล ลงในฐานข้อมูล ปัญหาที่เกิดขึ้นก็

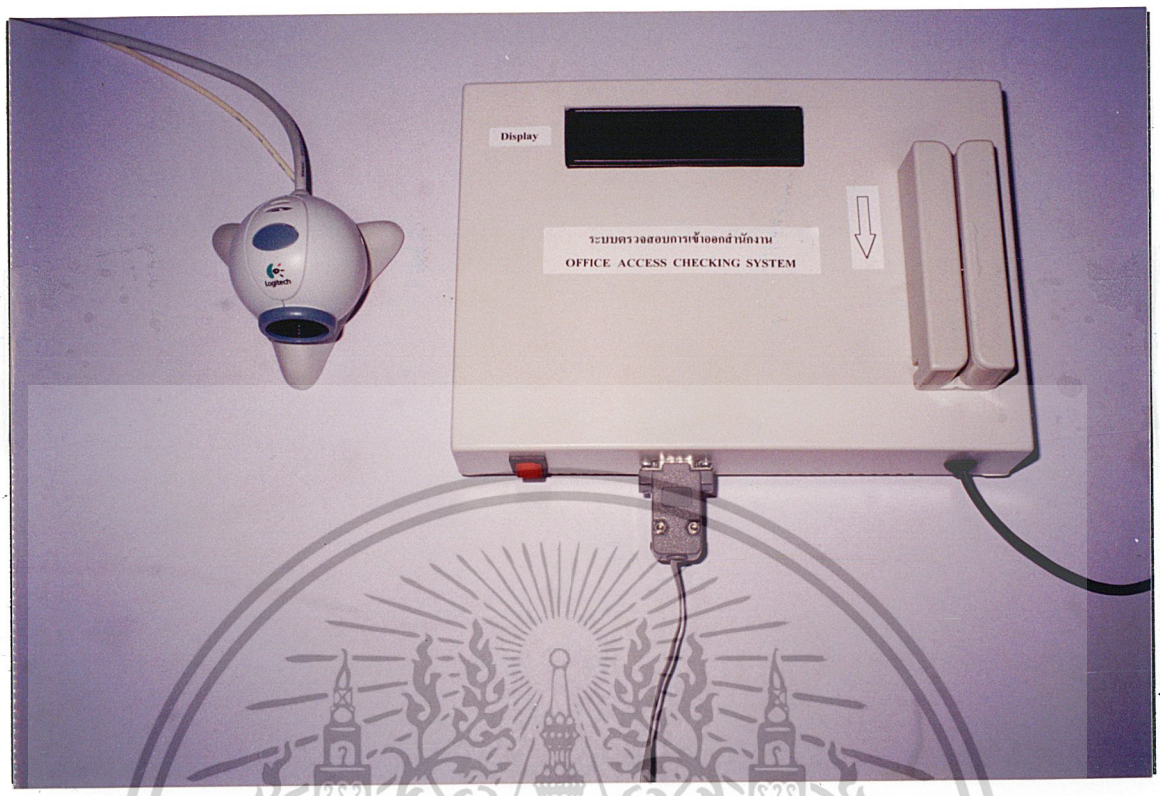
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คือ บางครั้งเมื่อรูดบัตรแม่เหล็กที่ถูกต้อง โปรแกรมกลับส่งข้อมูลกลับมาแสดงที่ LCD ว่าบัตรไม่ถูกต้อง แต่เก็บข้อมูลวันที่ เวลา และรูป ลงในฐานข้อมูล ซึ่งจากการอธิบายข้างต้นจะเห็นว่าเป็นไปไม่ได้ที่รูดบัตรไม่ถูกต้องแล้วโปรแกรมจะเก็บข้อมูลลงในฐานข้อมูล ปัญหาที่เกิดขึ้นนี้ยังหาสาเหตุไม่ได้ และยังไม่สามารถแก้ไขได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

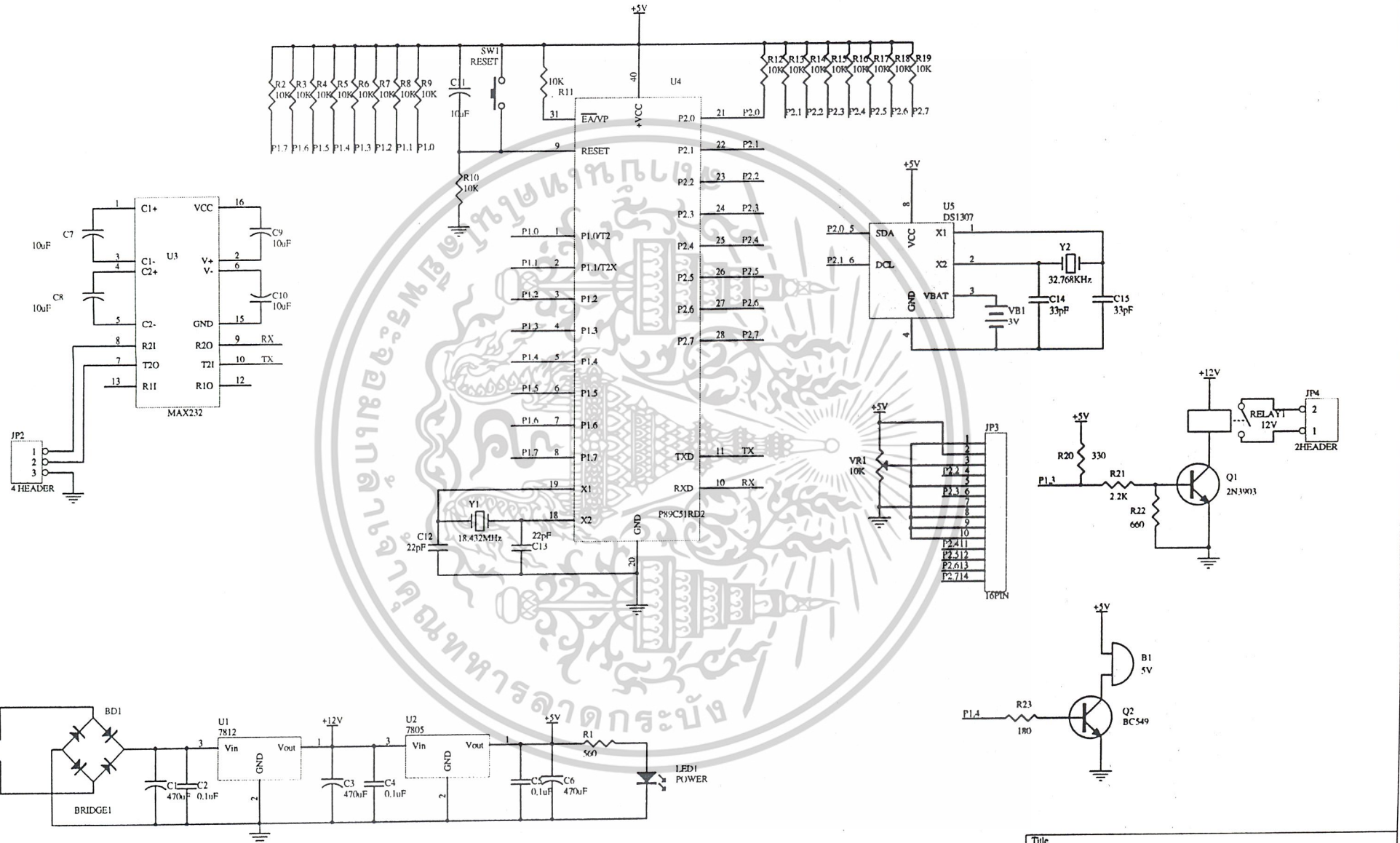
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปโครงงานระบบตรวจสอบการเข้าออกสำนักงาน

รูปบัตรแม่เหล็กที่ใช้เป็นบัตรรูด เข้า/ออก สำนักงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Title	ACCESS CHECKING SYSYTEM BORD
Size	B
Number	R030318
Revision	1.0
Date:	18-Mar-2003
File:	D:\WATCH\RD2.SCH
Sheet of	1/1
Drawn By:	Mr.CHINNAWATCH

DALLAS SEMICONDUCTOR

DS1307
64 X 8 Serial Real Time Clock

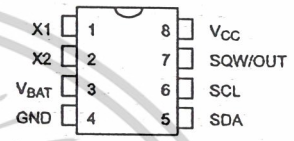
FEATURES

- Real time clock counts seconds, minutes, hours, date of the month, month, day of the week, and year with leap year compensation valid up to 2100
- 56 byte nonvolatile RAM for data storage
- 2–wire serial interface
- Programmable squarewave output signal
- Automatic power fail detect and switch circuitry
- Consumes less than 500 nA in battery backup mode at 25°C
- Optional industrial temperature range –40°C to +85°C (IND)
- Available in 8–pin DIP or SOIC

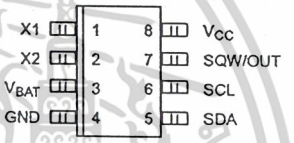
ORDERING INFORMATION

DS1307	Serial Timekeeping Chip; 8–pin DIP
DS1307Z	Serial Timekeeping Chip; 8–pin SOIC (150 mil)
DS1307N	8–pin DIP (IND)
DS1307ZN	8–pin SOIC (IND)

PIN ASSIGNMENT



DS1307 8–PIN DIP (300 MIL)



DS1307Z 8–PIN SOIC (150 MIL)

PIN DESCRIPTION

V_{CC}	– Primary Power Supply
X1, X2	– 32.768 KHz Crystal Connection
V_{BAT}	– +3 Volt Battery Input
GND	– Ground
SDA	– Serial Data
SCL	– Serial Clock
SQW/OUT	– Square wave/Output Driver

DESCRIPTION

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

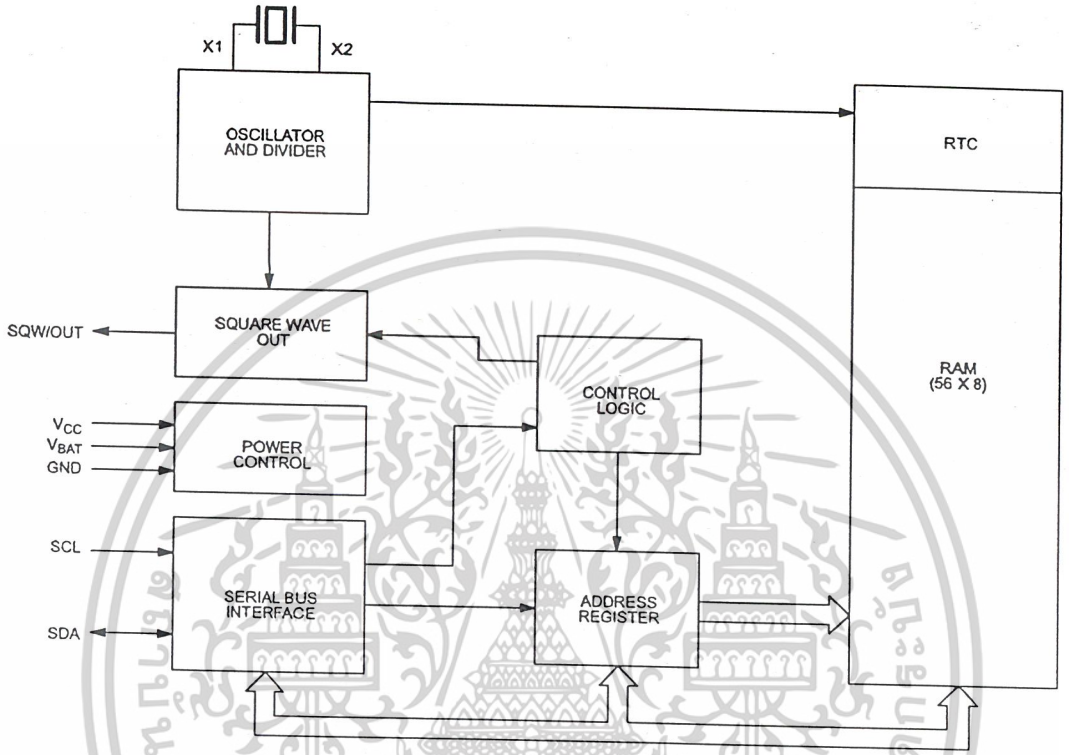
OPERATION

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

©Copyright 1995 by Dallas Semiconductor Corporation.
All Rights Reserved. For important information regarding patents and other intellectual property rights, please refer to Dallas Semiconductor data books.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DS1307 BLOCK DIAGRAM Figure 1



SIGNAL DESCRIPTIONS

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

V_{BAT} – Battery input for any standard 3 volt lithium cell or other energy source. Battery voltage must be held between 2.5 and 3.5 volts for proper operation. The nominal write protect trip point voltage at which access to the real time clock and user RAM is denied is set by the internal circuitry as 1.25 x V_{BAT} nominal. A Lithium battery with 35 mAh or greater will back up the DS1307 for more than 10 years in the absence of power.

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

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

SQW/OUT (Square Wave/ Output Driver) – When enabled, the SQWE bit set to 1, the SQW/OUT pin outputs one of four square wave frequencies (1 Hz, 4 KHz, 8 KHz, 32 KHz). The SQW/OUT pin is open drain which requires an external pull–up resistor.

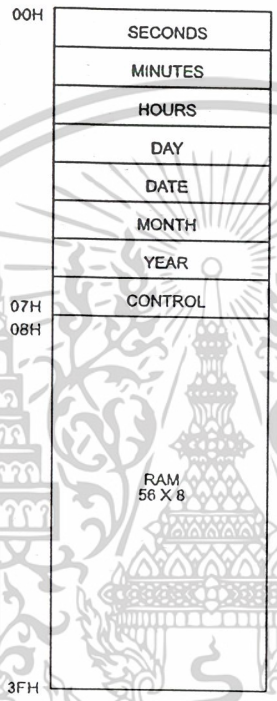
X1, X2 – Connections for a standard 32.768 KHz quartz crystal. The internal oscillator circuitry is designed for operation with a crystal having a specified load capacitance (CL) of 12.5 pF.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RTC AND RAM ADDRESS MAP

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

DS1307 ADDRESS MAP Figure 2



CLOCK AND CALENDAR

The time and calendar information is obtained by reading the appropriate register bytes. The real time clock registers are illustrated in Figure 3. The time and calendar are set or initialized by writing the appropriate register bytes. The contents of the time and calendar registers are in the Binary–Coded Decimal (BCD) format. Bit 7 of Register 0 is the Clock Halt (CH) bit. When this bit is set to a one, the oscillator is disabled. When cleared to a zero, the oscillator is enabled.

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DS1307 TIMEKEEPER REGISTERS Figure 3

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

CONTROL REGISTER

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

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

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

SQWE (Square wave Enable): This bit when set to a logic 1 will enable the oscillator output. The frequency of the square wave output depends on the value of the RS0 and RS1 bits.

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

SQUAREWAVE OUTPUT FREQUENCY Table 1

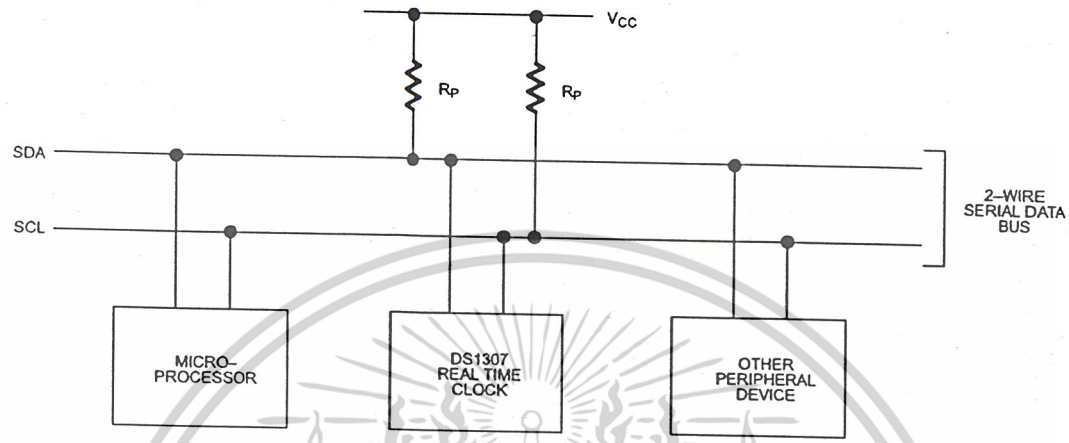
RS1	RS0	SQW OUTPUT FREQUENCY
0	0	1 Hz
0	1	4 KHz
1	0	8 KHz
1	1	32 KHz

2–WIRE SERIAL DATA BUS

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TYPICAL 2–WIRE BUS CONFIGURATION Figure 4



The following bus protocol has been defined (see Figure 5).

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

Accordingly, the following bus conditions have been defined:

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

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

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

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

Each data transfer is initiated with a START condition and terminated with a STOP condition. The number of data bytes transferred between the START and the STOP conditions is not limited, and is determined by the master device. The information is transferred byte–wise and each receiver acknowledges with a ninth bit.

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

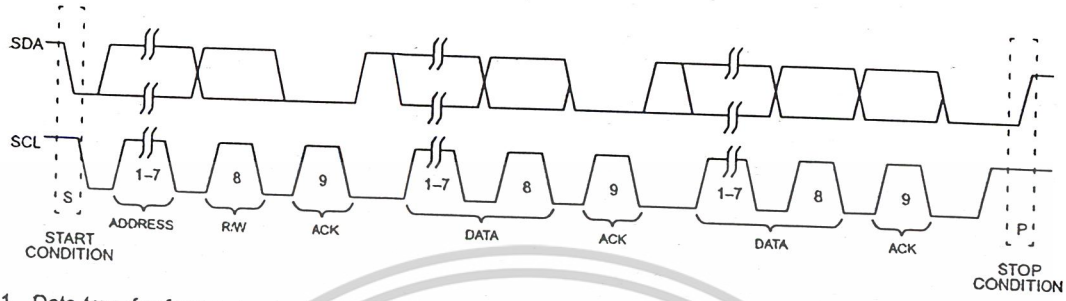
A device that acknowledges must pull down the SDA line during the acknowledge clock pulse in such a way that the SDA line is stable low during the high period of the acknowledge related clock pulse. Of course, setup and hold times must be taken into account. When receiving data from a slave a master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

DATA TRANSFER

Figures 5, 6, and 7 detail how data transfer is accomplished on the 2–wire bus. Depending on the state of the R/$\overline{W}$ bit in the transmission protocols as shown in Figures 6 and 7, two types of data transfer are possible:

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DATA TRANSFER ON 2–WIRE SERIAL BUS Figure 5



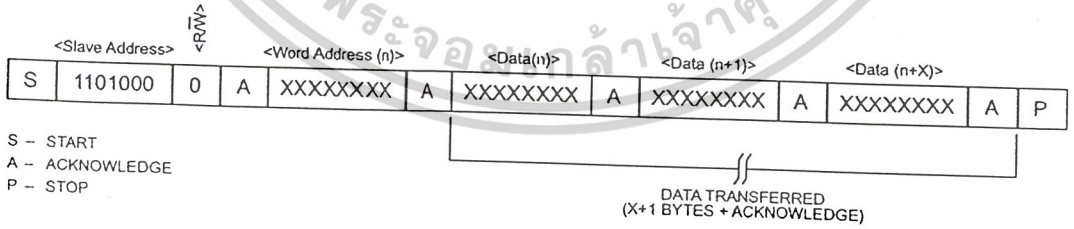
1. Data transfer from a master transmitter to a slave receiver. The first byte transmitted by the master is the slave address. Next follows a number of data bytes. The slave returns an acknowledge bit after each received byte. Data is transfered with the most significant bit (MSB) first.
2. Data transfer from a slave transmitter to a master receiver. The first byte (the slave address) is transmitted by the master. The slave then returns an acknowledge bit. This is followed by the slave transmitting a number of data bytes. The master returns an acknowledge bit after all received bytes other than the last byte. At the end of the last received byte, a 'not acknowledge' is returned.

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

The DS1307 may operate in the following two modes:

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

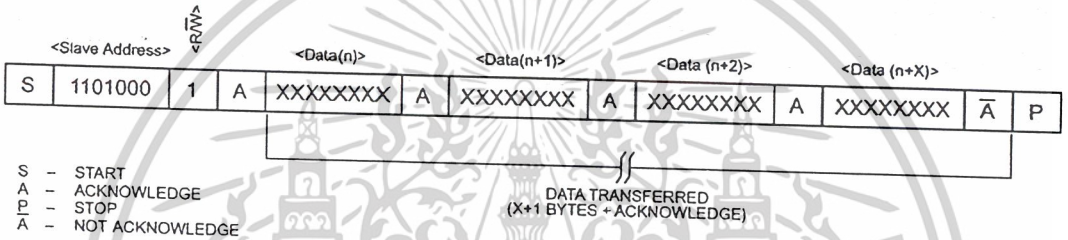
DATA WRITE – SLAVE RECEIVER MODE Figure 6



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

DATA READ – SLAVE TRANSMITTER MODE Figure 7



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ABSOLUTE MAXIMUM RATINGS*

Voltage on Any Pin Relative to Ground	–0.5V to +7.0V
Operating Temperature	0°C to 70°C
Storage Temperature	–55°C to +125°C
Soldering Temperature	260°C for 10 seconds

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

The Dallas Semiconductor DS1307 is built to the highest quality standards and manufactured for long term reliability. All Dallas Semiconductor devices are made using the same quality materials and manufacturing methods. However, standard versions of the DS1307 are not exposed to environmental stresses, such as burn–in, that some industrial applications require. Products which have successfully passed through this series of environmental stresses are marked IND or N, denoting their extended operating temperature and reliability rating. For specific reliability information on this product, please contact the factory at (972) 371–4448.

RECOMMENDED DC OPERATING CONDITIONS

(0°C to 70°C)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Supply Voltage	V_{CC}	4.5	5.0	5.5	V	1
Logic 1	V_{IH}	2.2		V_{CC}+0.3	V	1
Logic 0	V_{IL}	–0.3		+0.8	V	1
V_{BAT} Battery Voltage	V_{BAT}	2.5		3.5	V	1

DC ELECTRICAL CHARACTERISTICS

(0°C to 70°C; V_{CC}=4.5V to 5.5V)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Input Leakage	I_{LI}			1	µA	10
I/O Leakage	I_{LO}			1	µA	11
Logic 0 Output	V_{OL}			0.4	V	2
Active Supply Current	I_{CCA}			1.5	mA	9
Standby Current	I_{CCS}			200	µA	3
Battery Current (OSC ON); SQW/OUT OFF	I_{BAT1}		300	500	nA	4
Battery Current (OSC ON); SQW/OUT ON (32 KHz)	I_{BAT2}		480	800	nA	4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AC ELECTRICAL CHARACTERISTICS

(0°C to 70°C; V_{CC}=4.5V to 5.5V)

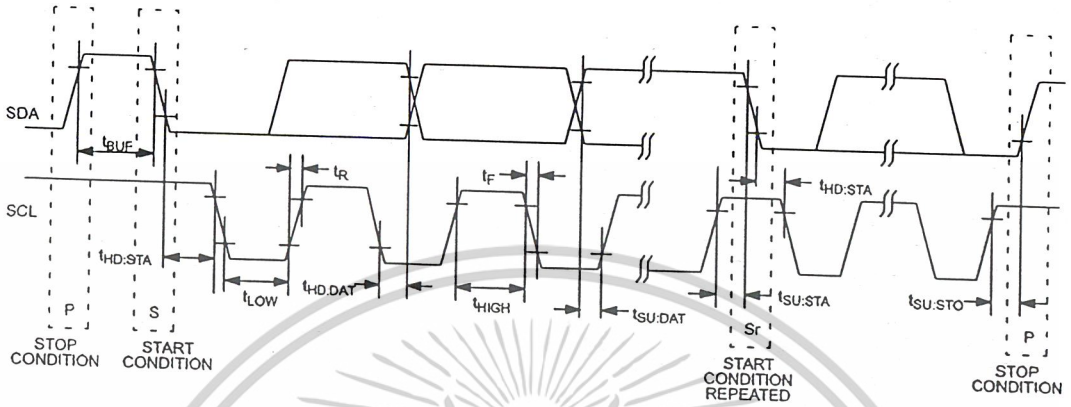
PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
SCL Clock Frequency	f_{SCL}	0		100	KHz	
Bus Free Time Between a STOP and START Condition	t_{BUF}	4.7			µs	
Hold Time (Repeated) START Condition	$t_{HD:STA}$	4.0			µs	5
LOW Period of SCL Clock	t_{LOW}	4.7			µs	
HIGH Period of SCL Clock	t_{HIGH}	4.0			µs	
Set–up Time for a Repeated START Condition	$t_{SU:STA}$	4.7			µs	
Data Hold Time	$t_{HD:DAT}$	0			µs	6, 7
Data Set–up Time	$t_{SU:DAT}$	250			ns	
Rise Time of Both SDA and SCL Signals	t_R			1000	ns	
Fall Time of Both SDA and SCL Signals	t_F			300	ns	
Set–up Time for STOP Condition	$t_{SU:STO}$	4.7			µs	
Capacitive Load for each Bus Line	C_B			400	pF	8
I/O Capacitance	$C_{I/O}$		10		pF	
Crystal Capacitance			12.5		pF	

NOTES:

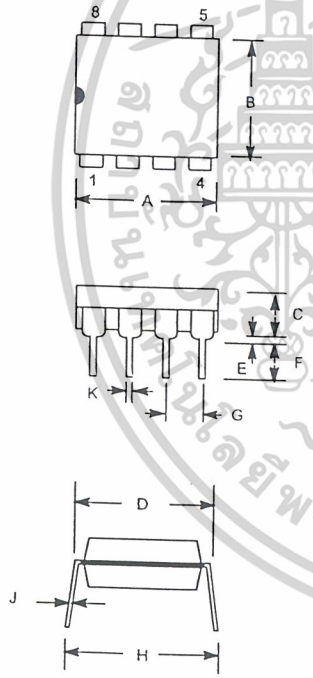
1. All voltages are referenced to ground.
2. Logic zero voltages are specified at a sink current of 5 mA at V_{CC}=4.5V, V_{OL}=GND for capacitive loads.
3. I_{CCS} specified with V_{CC}=5.0V and SDA, SCL=5.0V.
4. V_{CC}=0V, V_{BAT}=3V.
5. After this period, the first clock pulse is generated.
6. A device must internally provide a hold time of at least 300 ns for the SDA signal (referred to the V_{IHMIN} of the SCL signal) in order to bridge the undefined region of the falling edge of SCL.
7. The maximum $t_{HD:DAT}$ has only to be met if the device does not stretch the LOW period (t_{LOW}) of the SCL signal.
8. C_B – total capacitance of one bus line in pF.
9. I_{CCA} – SCL clocking at max frequency = 100 KHz.
10. SCL only.
11. SDA and SQW/OUT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TIMING DIAGRAM



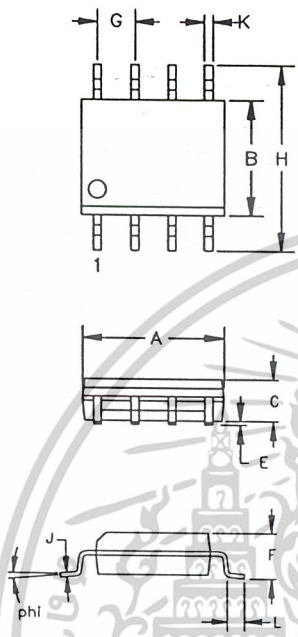
DS1307 64 X 8 SERIAL REAL TIME CLOCK 8–PIN DIP



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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DS1307Z 64 X 8 SERIAL REAL TIME CLOCK 8–PIN SOIC (150 MIL)



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

SG–G2008–001

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;****************************** *
;* Program ACCESS CHECKING SYSTEM *     SEND CODE -- OK
;* RTC+MCU+MAGNETIC READER        *     RECEIVE NAME -- NAME
(LINE 1)
;* HardWare     : CP-SPI/RD2      *     RECEIVE INFO -- INFO
(LINE 2)
;* Display By Dot. LCD 24x2       *
************************
;* CPU Control : MCS51            *      hh:mm:nn <<TELECOM
CO.,LTD
;* Complier    : SXA51            *
;****************************** *
************************

            ORG     20H

SCL     EQU     P2.1            ;***** DS1307 ****
SDA     EQU     P2.0

CONT_BYTE_W     EQU     0D0H            ;**** DS1307 ******
CONT_BYTE_R     EQU     0D1H

CS_LCD      EQU     P2.3        ; E  LCD (PIN INT1)     ****
LCD ****
RS_LCD      EQU     P2.2        ; RS LCD (Pin T0)


DATA_MAG    EQU     P1.0    ; Data From Magnetic Card       **
MAGNETIC READER
PRESENT     EQU     P1.1    ; Enable Magnetic Card          **
CLK_MAG     EQU     P1.2    ; Clock Sync Magnetic Card

FLAG_BUF:   DS      1                   ; Flag Buffer
CHECKSUM:   DS      1
BUFFER:     DS      40                  ; 40 Byte Buffer

ADD_LOW:    DS      1                   ;***** DS1307 *******
   DATA:    DS      1

SEC:        DS      1
MIN:        DS      1
HOUR:       DS      1                   ;***** DS1307 *******
;
START       EQU     FLAG_BUF.0          ; Start Flag Status
STOP        EQU     FLAG_BUF.1          ; Stop Flag Status
ERROR       EQU     FLAG_BUF.2          ; Error Flag Status
PARITY      EQU     FLAG_BUF.3          ; Parity Error Flag
Status

            ORG     0000H
PROGRAM:    MOV     SP,#256-32          ; Stack 32 Byte
            MOV     FLAG_BUF,#0         ; Clear All Flag
            SETB    DATA_MAG            ;
            SETB    PRESENT
            SETB    CLK_MAG
            MOV     SCON,#50H
            MOV     TMOD,#20H
            MOV     TH1,#0F6H
            MOV     A,#00H
            MOV     PCON,A
            CLR     ES
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            CLR     ET1
            SETB    TR1
            CLR         P1.3            ;@@@@@@@@@@@
            LCALL   DELAY               ; DELAY FIRST***
            SETB    CS_LCD
            LCALL   INIT_LCD            ; Initial LCD 4Bit

;*************************************************************
**
;                       REAL TIME
;*************************************************************
**
            SETB        SCL
            SETB        SDA
            MOV         R7,#0FH             ; COUNTER
COMPANY DISPLAY

INIT3:      JNB         PRESENT,DSP_NUM0
            LCALL       GOTO_LCD            ; 00 DDRAM OF LCD
            MOV         ADD_LOW,#02H
            LCALL       READ_BYTE             ; HOUR :   hh
            LCALL       CHANG

            MOV         A,#3AH                  ; :
            LCALL       WR_LCD

            MOV         ADD_LOW,#01H
            LCALL       READ_BYTE               ; MINITE  mm
            LCALL       CHANG

            MOV         A,#3AH                  ;       :
            LCALL       WR_LCD

            MOV         ADD_LOW,#00H
            LCALL       READ_BYTE               ;  SECOND  ss
            LCALL       CHANG

;*****************END REAL TIME
************************************

DSP_NUM0:   JB      PRESENT,TO_SHOW
            LCALL   GET_DATA            ; Get Number of Card
            MOV     R0,#BUFFER
            INC     R0
            JB      ERROR,SUM_ERR
            JB      PARITY,PAR_ERR
            MOV     R4,#08H
            AJMP    DSP_TX

            ;
SUM_ERR:    MOV     DPTR,#CK_ER
            LCALL   GOTO_LCD1
            LCALL   PRINT_LCD                   ; SHOW PARITY
ERROR
            CLR     ERROR
            SETB    PRESENT
            LCALL   DELAY_L
            LCALL   DELAY_L
            LCALL   DELAY_L
            LCALL   DELAY_L
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            MOV     R7,#0FH                 ; BEGINING COUNTER
COMPANY DISPLAY
            AJMP    INIT3
            ;
PAR_ERR:    MOV     DPTR,#P_ER
            LCALL   GOTO_LCD1
            LCALL   PRINT_LCD               ; SHOW CHECKSUM
ERROR
            CLR     PARITY
            SETB    PRESENT
            LCALL   DELAY_L
            LCALL   DELAY_L
            LCALL   DELAY_L
            LCALL   DELAY_L
            MOV     R7,#0FH                 ; BEGINING COUNTER
COMPANY DISPLAY
            AJMP    INIT3

;-------------------------------------------------------
ALWAY:          PUSH    ACC
                MOV     A,#02H
                ANL     A,P1
                CJNE    A,#02H,DSP_NUM0
                POP     ACC
                RET

;-------------------------------------------------------


            ;
  CK_ER:    DB      'CARD ERROR !!   ',00
   P_ER:    DB      'TRACK Error !! ',00
TO_SHOW:     MOV        SP,#09H
             LCALL       SHOW
           ;

 DSP_TX:    MOV         A,#00100011B            ; # START
DATA TO SEND
            LCALL       TX_BYTE
 DSP_NUM1:  MOV      A,@R0
            DJNZ     R4,DSP_NUM2
            MOV         A,#00101111B            ; / STOP DATA
TO SEND
            LCALL       TX_BYTE
            SETB     PRESENT
            LCALL       RX_BYTE
MIROO:      MOV         R7,#0FH
            AJMP       INIT3
  DSP_NUM2: ADD      A,#30H
            LCALL    TX_BYTE
            LCALL       GOTO_LCD1
            MOV         DPTR,#SEND
            LCALL       PRINT_LCD
            INC      R0
            AJMP     DSP_NUM1

SEND:           DB      '   CHECKING     ',00
;;*********************
;* Clear Data Buffer *
;* Register : R0,ACC *
;*********************
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CLR_BUFF:   PUSH    B
            MOV     B,#40
            MOV     R0,#BUFFER          ; Clear Data Buffer
CLR_BUF1:   CLR     A
            MOV     @R0,A
            INC     R0
            DJNZ    B,CLR_BUF1
            POP     B
            RET

;*******************************
;* Read Data From Magnetic Card *
;* Get Number of Card (Modulo5) *
;* Output : BUFFER (BCD 40Byte) *
;* Reg.   : R1,R2,ACC,CHECKSUM  *
;*******************************
;
GET_DATA:   SETB    DATA_MAG            ; Standy Signale
            SETB    PRESENT
            SETB    CLK_MAG
            LCALL   CLR_BUFF            ; Clear Buffer
            MOV     CHECKSUM,#0
            MOV     R1,#BUFFER-1        ; Pointer to Save
Data
            CLR     START               ; Clear Any Flag
            CLR     STOP
            CLR     ERROR
            ;
GET_DAT0:   LCALL   GET_CLK             ; Get Start Sentinel
            MOV     C,DATA_MAG
            CPL     C
            JB      DATA_MAG,GET_DAT0   ; Loop Until Start
Bit
            SETB    START               ; Strat Bit OK
            ;
GET_DAT1:   MOV     R2,#5               ; Modulo 5 Format
Counter
            CLR     A
            CLR     PARITY
            INC     R1                  ; Point to Next Byte
Save
GET_DAT2:   JB      START,GET_DAT3      ; Get 1-Byte Data
            LCALL   GET_CLK
            MOV     C,DATA_MAG
            CPL     C
GET_DAT3:   RRC     A
            CLR     START
            DJNZ    R2,GET_DAT2
            RR      A
            RR      A
            RR      A
            JB      P,GET_DAT4          ; Parity Odd OK
            SETB    PARITY              ; Parity Error
            SJMP    GET_DAT8
            ;
GET_DAT4:   ANL     A,#0FH              ; Ignore Parity
            JB      STOP,GET_DAT5
            PUSH    ACC
            XRL     A,CHECKSUM          ; Checksum Data
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            MOV     CHECKSUM,A
            POP     ACC
GET_DAT5:   MOV     @R1,A               ; Save Data
            JB      STOP,GET_DAT6       ; Stop Operation
            CJNE    A,#0FH,GET_DAT1
            SETB    STOP
            SJMP    GET_DAT1
GET_DAT6:   CJNE    A,CHECKSUM,GET_DAT7
            CLR     ERROR               ; Checksum OK
            SJMP    GET_DAT8
GET_DAT7:   SETB    ERROR               ; Checksum Error
GET_DAT8:   RET

;**********************
;* Get Magnetic Clock *
;**********************
GET_CLK:    JB      CLK_MAG,$           ; Wait Falling Clock
            NOP
            NOP
           JNB     CLK_MAG,$           ; Wait Rising Clock
            NOP
            NOP
            RET
;************************************
;*  Print Data Box to DotMatrix LCD *
;*  Usage    : LCALL PRINT_LCD      *
;*           : DB    'xxxx',00      *
;*  Register : ACC                  *
;*  Note     : last byte must be 00 *
;************************************

PRINT_LCD:  CLR     A
            MOVC    A,@A+DPTR
            CJNE    A,#00H,PRINT2
            RET
PRINT2:     LCALL   WR_LCD
            INC     DPTR
            SJMP    PRINT_LCD

;**************************
;*   Write ASCII to LCD   *
;* Input  : ACC  (ASCII)  *
;* Output : Data bus LCD  *
;**************************
;
WR_LCD:     PUSH    B
            MOV     B,A
             ANL     A,#0F0H
            ORL     A,#0CH
            MOV     R2,A
            MOV     A,P2
            ANL     A,#03H
            ORL     A,R2
            MOV     P2,A            ; High byte
            LCALL   EN_LCD
            MOV     A,B             ; Low byte
            SWAP    A
            ANL     A,#0F0H
            ORL     A,#0CH
            MOV     R2,A
            MOV     A,P2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            ANL     A,#03H
            ORL     A,R2
            MOV     P2,A
            LCALL   EN_LCD
            POP     B
            RET

;**************************
;* Write Instruction LCD  *
;* Input  : ACC (Command) *
;* Output : Data bus LCD  *
;**************************
;
WR_INS:     PUSH    B
            MOV     B,A
            ANL     A,#0F0H
            SETB    ACC.3           ;  CS HI
            MOV     R2,A
            MOV     A,P2
            ANL     A,#03H
            ORL     A,R2
            MOV     P2,A
            LCALL   EN_LCD          ; High byte
            MOV     A,B             ; Low byte
            SWAP    A
            ANL     A,#0F0H
            SETB    ACC.3
            MOV     R2,A
            MOV     A,P2
            ANL     A,#03H
            ORL     A,R2
            MOV     P2,A
            LCALL   EN_LCD
            POP     B
            RET

;************************
;* Goto position of LCD *
;* Input : ACC (addr.)  *
;************************
;
GOTO_LCD:   MOV     A,#01H  ;CLEAR DISPLAY * SET DD RAM
ADDRESS 0 ***
            LCALL   WR_INS
            RET
GOTO_LCD1:                  ;SET DD RAM #089H BEGINING
POSITION DATA DISPLAY ***
            MOV     A,#089H
            LCALL   WR_INS
            RET
GOTO_LCD2:                  ;SET DD RAM #040H BEGINING
POSITION DATA DISPLAY ***
            MOV     A,#0C0H ; SECOND LINE
            LCALL   WR_INS
            RET

;********************
;*   Initial LCD    *
;* 4-Bit Interface *
;********************
;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
INIT_LCD:   CLR     RS_LCD
            MOV     A,#33H          ; Set DL=1 3-time
            LCALL   WR_INS
            MOV     A,#32H          ; Clear DL=0 1-time
            LCALL   WR_INS
            MOV     A,#28H          ; Function set
            LCALL   WR_INS          ; DL=0 4Bit,N=0 1Line,F=0
5X7
            MOV     A,#0CH          ; Display on/off Control
            LCALL   WR_INS          ; Entry display,cursor
off,cursor not blink
            MOV     A,#06H          ; Entry mode set
            LCALL   WR_INS          ; I/D=1 Increment,S=0
cursor shift
            MOV     A,#01H          ; Clear display
            LCALL   WR_INS          ; clear display,set DD
RAM address=0
            RET

;**********************
;*  Enable Pin E LCD  *
;* Active Chip select *
;**********************
;
EN_LCD:     CLR     CS_LCD          ; Enable LCD
            LCALL   BUSY            ; Busy delay time
            SETB    CS_LCD          ; Disable LCD
            RET

;**********************
;* Delay time for Busy *
;*    Wait LCD Ready   *
;**********************
 BUSY:      PUSH    07H
            PUSH    06H
            MOV     R6,#0FH
 BUSY1:     MOV     R7,#0FFH
            DJNZ    R7,$
            DJNZ    R6,BUSY1
            POP     06H
            POP     07H
            RET
;****************************
;       DELAY FIRST         *
;****************************
DELAY:      PUSH    07H
            PUSH    06H
  DEL1:     MOV     R6,#0FFH
  DEL2:     MOV     R7,#0FFH
            DJNZ    R7,$
            DJNZ    R6,DEL2
            POP     06H
            POP     07H
            RET
 ;****************************
 ;      TEST PRINT LCD
 ;****************************
   WR_0:    PUSH    ACC
            MOVC    A,@A+DPTR
            CJNE    A,#00H,WR_1
            POP     ACC
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            RET
   WR_1:    LCALL       WR_LCD
            POP         ACC
            INC         ACC
            CJNE        A,#08H,PASS
            LCALL       GOTO_LCD1
  PASS:     LJMP        WR_0
            RET
;*************************************************************
;``              DS1307 LET GO
;*************************************************************
WRITE_BYTE:     CLR     SDA
                CLR     SCL
                MOV     A,#CONT_BYTE_W
                LCALL   LOOP_BYTE
                SETB    SDA
                SETB    SCL
                JB      SDA,WRITE_BYTE
                CLR     SCL
                MOV     A,ADD_LOW
                LCALL   LOOP_BYTE
                SETB    SDA
                SETB    SCL
                JB      SDA,WRITE_BYTE
                CLR     SCL
                MOV     A,DATA
                LCALL   LOOP_BYTE
                SETB    SDA
                SETB    SCL
                JB      SDA,WRITE_BYTE
                CLR     SDA
                CLR     SCL
                SETB    SCL
                SETB    SDA             ;  STOP BIT
                RET

;*************************************************************
;``              DS1307 LET GO
;*************************************************************
READ_BYTE:      CLR     SDA
                CLR     SCL
                MOV     A,#CONT_BYTE_W
                LCALL   LOOP_BYTE
                SETB    SDA
                SETB    SCL
                JB      SDA,READ_BYTE
                CLR     SCL
                MOV     A,ADD_LOW
                LCALL   LOOP_BYTE
                SETB    SDA
                SETB    SCL
                JB      SDA,READ_BYTE
                CLR     SCL

                SETB    SCL
                SETB    SDA
                CLR     SDA
                CLR     SCL
                MOV     A,#CONT_BYTE_R
                LCALL   LOOP_BYTE
                SETB    SDA
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                SETB    SCL
                JB      SDA,READ_BYTE
                CLR     SCL
                LCALL   LOOP_READ
                SETB    SDA
                SETB    SCL
                CLR     SCL

                SETB    SCL
                SETB    SDA             ;  STOP BIT
                RET


;***************LOOP_BYTE**********************
LOOP_BYTE:      PUSH    02H
                MOV     R2,#08H
LOOP_SEND:      RLC     A
                MOV     SDA,C
                SETB    SCL
                CLR     SCL
                DJNZ    R2,LOOP_SEND
                POP     02H
                RET
;****************** READ ************************
LOOP_READ:       PUSH    02H
                MOV     R2,#08H
LOOP_READ1:     SETB    SCL
                MOV     C,SDA
                CLR     SCL
                RLC     A
                DJNZ    R2,LOOP_READ1
                MOV     DATA,A
                POP     02H
                RET
 ;*********************************************

CURSOR:         PUSH    ACC
                MOV     A,00111010B
                LCALL   WR_LCD
                POP     ACC
                RET

CHANG:          PUSH    ACC
                SWAP    A
                ANL     A,#0FH
                LCALL   CHANG1
                LCALL   WR_LCD
                POP     ACC
                ANL     A,#0FH
                LCALL   CHANG1
                LCALL   WR_LCD
                RET
CHANG1:         PUSH    DPH
                PUSH    DPL
                MOV     DPTR,#REAL
                MOVC    A,@A+DPTR
                POP     DPL
                POP     DPH
                RET

TX_BYTE:        CLR     TI
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV     SBUF,A
                JNB     TI,$
                CLR     TI
                RET
RX_BYTE:        LCALL   GOTO_LCD1
                LCALL   CLR_BUFF
                MOV     R1,#BUFFER
AGAIN:          CLR     RI
                JNB     RI,$
                MOV     A,SBUF
                CJNE    A,#00100011B,TELE       ; # CHECK FOR
START DATA
                SJMP    REAM
TELE:           CJNE    A,#00100001B,AGAIN       ; ! CHECK
FOR INCORRECT
                SJMP    NOTT

REAM:           CLR     RI
                JNB     RI,$
                MOV     A,SBUF
                MOV     @R1,A
                CJNE    A,#00100100B,REAM_1     ;  $ CHECK
FOR STOP DATA
                LCALL   HIDOO
REAM_1:         MOV     @R1,A
                INC     R1
                SJMP    REAM

;***** SHOW RECEIVE ON DISPLAY ***********

HIDOO:          SJMP    HIMOD
HIDOO_0:        MOV     R0,#BUFFER
                MOV     A,@R0
CK_DOO:         CJNE    A,#00101111B,HIDOO_1    ; / CHECK
FOR SAPERATE DATA
                LCALL   GOTO_LCD2
KLUB:           INC     R0
                MOV     A,@R0
                CJNE    A,#00100100B,CK_DOO1    ; $ CHECK FOR
STOP DATA                SJMP    PAUSH_X
                SETB    P1.3                    ; ON CAMERA
                SJMP    PAUSH_X
CK_DOO1:        LCALL   WR_LCD
                SJMP    KLUB
HIDOO_1:        LCALL   WR_LCD
                INC     R0
                MOV     A,@R0
                SJMP    CK_DOO

HIMOD:          MOV     R0,#0EH
HIMOD_1:        MOV     A,#20H
                LCALL   WR_LCD
                DJNZ    R0,HIMOD_1
                LCALL   GOTO_LCD1
                SJMP    HIDOO_0

;***** SHOW CARD NOT CORECT **********

NOTT:           LCALL   GOTO_LCD1
                MOV     DPTR,#FAIL0
                LCALL   PRINT_LCD
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                LCALL   GOTO_LCD2
                MOV     DPTR,#FAIL1
                LCALL   PRINT_LCD
                SJMP    PAUSH
                LCALL   MIROO

PAUSH:          MOV     R7,#0FH         ; PAUSH DISPLAY
PAUSH_1:        LCALL   DELAY_L
                DJNZ    R7,PAUSH_1
                RET
PAUSH_X:        MOV     R7,#0FH         ; PAUSH DISPLAY
PAUSH_Y:        LCALL   DELAY_L
                DJNZ    R7,PAUSH_Y
                CLR     P1.3
                LCALL   MIROO

DELAY_L:        PUSH    04H
                MOV     R4,#05H
TOO:            LCALL   DELAY
                DJNZ    R4,TOO
                POP     04H
                RET
                ;
SHOW:           LCALL   ALWAY           ;-------CHECK ------
                CJNE    R7,#0FH,CHECK_0
                MOV     A,#097H
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT
                LCALL   INIT3
CHECK_0:        LCALL   ALWAY           ;-------CHECK ------
                CJNE    R7,#0EH,CHECK_1
                MOV     A,#097H
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_0
                LCALL   INIT3
CHECK_1:        LCALL   ALWAY           ;-------CHECK ------
                CJNE    R7,#0DH,CHECK_2
                MOV     A,#096H
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_1
                LCALL   INIT3
CHECK_2:        LCALL   ALWAY           ;-------CHECK ------
                CJNE    R7,#0CH,CHECK_3
                MOV     A,#095H
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_2
CHECK_3:        LCALL   ALWAY           ;-------CHECK ------
                CJNE    R7,#0BH,CHECK_4
                MOV     A,#094H
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_3
CHECK_4:        CJNE    R7,#0AH,CHECK_5
                MOV     A,#093H
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_4
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CHECK_5:        LCALL   ALWAY           ;-------CHECK ------
                CJNE    R7,#09H,CHECK_6
                MOV     A,#092H
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                AJMP    NEXT_5
CHECK_6:        CJNE    R7,#08H,CHECK_7
                MOV     A,#091H
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_6
CHECK_7:        LCALL   ALWAY           ;-------CHECK ------
                CJNE    R7,#07H,CHECK_8
                MOV     A,#090H
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_7
CHECK_8:        CJNE    R7,#06H,CHECK_9
                MOV     A,#08FH
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_8
CHECK_9:        CJNE    R7,#05H,CHECK_10
                MOV     A,#08EH
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_9
CHECK_10:       LCALL   ALWAY           ;-------CHECK ------
                CJNE    R7,#04H,CHECK_11
                MOV     A,#08DH
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_10
CHECK_11:       CJNE    R7,#03H,CHECK_12
                MOV     A,#08CH
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_11
CHECK_12:       CJNE    R7,#02H,CHECK_13
                MOV     A,#08BH
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_12
CHECK_13:       LCALL   ALWAY           ;-------CHECK ------
                CJNE    R7,#01H,CHECK_14
                MOV     A,#08AH
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_13
CHECK_14:       CJNE    R7,#00H,CHECK_15
                MOV     A,#089H
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_14
CHECK_15:       LCALL   ALWAY           ;-------CHECK ------
                CJNE    R7,#0FFH,CHECK_16
                MOV     A,#089H
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_15
CHECK_16:       CJNE    R7,#0FEH,CHECK_17
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV     A,#089H
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_16
CHECK_17:       LCALL   ALWAY           ;-------CHECK ------
                CJNE    R7,#0FDH,CHECK_18
                MOV     A,#089H
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_17
CHECK_18:       CJNE    R7,#0FCH,CHECK_19
                MOV     A,#089H
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_18
CHECK_19:       CJNE    R7,#0FBH,CHECK_20
                MOV     A,#089H
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_19
CHECK_20:       LCALL   ALWAY           ;-------CHECK ------
                CJNE    R7,#0FAH,CHECK_21
                MOV     A,#089H
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_20
CHECK_21:       CJNE    R7,#0F9H,CHECK_22
                MOV     A,#089H
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_21
CHECK_22:       LCALL   ALWAY           ;-------CHECK ------
                CJNE    R7,#0F8H,CHECK_23
                MOV     A,#089H
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_22
CHECK_23:       CJNE    R7,#0F7H,CHECK_24
                MOV     A,#089H
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_23
CHECK_24:       LCALL   ALWAY           ;-------CHECK ------
                CJNE    R7,#0F6H,CHECK_25
                MOV     A,#089H
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_24
CHECK_25:       CJNE    R7,#0F5H,CHECK_26
                MOV     A,#089H
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_25
CHECK_26:       LCALL   ALWAY           ;-------CHECK ------
                CJNE    R7,#0F4H,CHECK_27
                MOV     A,#089H
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_26
CHECK_27:       CJNE    R7,#0F3H,CHECK_28
                MOV     A,#089H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_27
CHECK_28:       LCALL   ALWAY           ;-------CHECK ------
                CJNE    R7,#0F2H,CHECK_29
                MOV     A,#089H
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_28
CHECK_29:       CJNE    R7,#0F1H,CHECK_30
                MOV     A,#089H
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_29
CHECK_30:       MOV     A,#089H
                LCALL   WR_INS
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   NEXT_30
                ;
NEXT:           DEC     R7
                MOV     A,#20H
                LCALL   WR_LCD
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   DELAY
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   DELAY
                LCALL   ALWAY           ;-------CHECK ------
                LCALL   DELAY
                AJMP    INIT3
NEXT_0:         DEC     R7
                MOV     DPTR,#COM_0
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_1:         DEC     R7
                MOV     DPTR,#COM_1
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_2:         DEC     R7
                MOV     DPTR,#COM_2
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_3:         DEC     R7
                MOV     DPTR,#COM_3
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_4:         DEC     R7
                MOV     DPTR,#COM_4
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_5:         DEC     R7
                MOV     DPTR,#COM_5
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_6:         DEC     R7
                MOV     DPTR,#COM_6
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_7:         DEC     R7
                MOV     DPTR,#COM_7
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_8:         DEC     R7
                MOV     DPTR,#COM_8
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_9:         DEC     R7
                MOV     DPTR,#COM_9
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_10:        DEC     R7
                MOV     DPTR,#COM_10
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_11:        DEC     R7
                MOV     DPTR,#COM_11
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_12:        DEC     R7
                MOV     DPTR,#COM_12
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_13:        DEC     R7
                MOV     DPTR,#COM_13
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_14:        DEC     R7
                MOV     DPTR,#COM_14
                LCALL   PRINT_LCD
                LCALL   STABLE
                MOV     R7,#0FFH
                AJMP    INIT3
NEXT_15:        DEC     R7
                MOV     DPTR,#COM_15
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_16:        DEC     R7
                MOV     DPTR,#COM_16
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_17:        DEC     R7
                MOV     DPTR,#COM_17
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_18:        DEC     R7
                MOV     DPTR,#COM_18
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_19:        DEC     R7
                MOV     DPTR,#COM_19
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_20:        DEC     R7
                MOV     DPTR,#COM_20
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_21:        DEC     R7
                MOV     DPTR,#COM_21
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_22:        DEC     R7
                MOV     DPTR,#COM_22
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_23:        DEC     R7
                MOV     DPTR,#COM_23
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_24:        DEC     R7
                MOV     DPTR,#COM_24
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_25:        DEC     R7
                MOV     DPTR,#COM_25
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_26:        DEC     R7
                MOV     DPTR,#COM_26
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_27:        DEC     R7
                MOV     DPTR,#COM_27
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_28:        DEC     R7
                MOV     DPTR,#COM_28
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_29:        DEC     R7
                MOV     DPTR,#COM_29
                LCALL   PRINT_LCD
                LCALL   STABLE
                AJMP    INIT3
NEXT_30:        DEC     R7
                MOV     DPTR,#COM_30
                LCALL   PRINT_LCD
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                LCALL   STABLE
                MOV     R7,#0FH
                AJMP   INIT3

STABLE:         MOV     R1,#04H
TOY:            LCALL   ALWAY
                LCALL   DELAY
                DJNZ    R1,TOY
                RET

DISPLAY:        PUSH    05H
                MOV     R5,#03H
ONOFF:          LCALL   DELAY_L
                MOV     A,#0CH
                LCALL   WR_INS
                LCALL   DELAY_L
                MOV     A,#08H
                LCALL   WR_INS
                DJNZ    R5,ONOFF
                POP     05H
                RET
                ;
COM_0:          DB      'T',00
COM_1:          DB      'TE',00
COM_2:          DB      'TEL',00
COM_3:          DB      'TELE',00
COM_4:          DB      'TELEC',00
COM_5:          DB      'TELECO',00
COM_6:          DB      'TELECOM',00
COM_7:          DB      'TELECOM ',00
COM_8:          DB      'TELECOM C',00
COM_9:          DB      'TELECOM CO',00
COM_10:         DB      'TELECOM CO.',00
COM_11:         DB      'TELECOM CO.,',00
COM_12:         DB      'TELECOM CO.,L',00
COM_13:         DB      'TELECOM CO.,LT',00
COM_14:         DB      'TELECOM CO.,LTD',00
COM_15:         DB      'ELECOM CO.,LTD',00
COM_16:         DB      'LECOM CO.,LTD',00
COM_17:         DB      'ECOM CO.,LTD',00
COM_18:         DB      'COM CO.,LTD',00
COM_19:         DB      'OM CO.,LTD',00
COM_20:         DB      'M CO.,LTD',00
COM_21:         DB      ' CO.,LTD',00
COM_22:         DB      'CO.,LTD',00
COM_23:         DB      'O.,LTD',00
COM_24:         DB      '.,LTD',00
COM_25:         DB      ',LTD',00
COM_26:         DB      'LTD',00
COM_27:         DB      'TD',00
COM_28:         DB      'D',00
COM_29:         DB      '',00
COM_30:         DB      '',00

FAIL0:           DB      '  NOT ENTRY     ',00
FAIL1:           DB      '   YOUR CARD INCORECT',00
 REAL:          DB
30H,31H,32H,33H,34H,35H,36H,37H,38H,39H
                DB      41H,42H,43H,44H,45H,46H
                END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

***** โปรแกรมในส่วนของ VB *****

```
** Form Main **
Dim firstdate
Dim n As Integer

Private Sub Command1_Click()
Login.Show
Main.Hide
End Sub

Private Sub Form_Load()
firstdate = DateValue(Date)
End Sub

Private Sub Timer1_Timer()      ' Timer ให้โปรแกรมทำงานตามเวลาที่กำหนด
If Not MSComm1.PortOpen Then                    ' เช็ค Port
MSComm1.PortOpen = True
End If
Dim inr
Dim id

MSComm1.InputLen = 0
If MSComm1.InBufferCount Then
inr = MSComm1.Input                             ' อ่านค่าจาก Buffer
id = Mid$(inr, 2, 7)

Dim totalid As Integer                          ' ตวจสอบรหัสพนักงาน
Dim checkcard As Boolean
Data1.Recordset.FindFirst "id = " & id & ""
totalid = Data1.Recordset.id.Value
If id = totalid Then
   checkcard = True

Else
   checkcard = False
End If
If checkcard = True Then

Dim clock
Dim olddate
Dim m As Integer
Dim total As Integer
Dim total1 As Integer
Dim a As Integer

olddate = DateValue(Date)
If id > n Then
   n = id
End If

If firstdate <> olddate Then                    ' เคลียค่าในตาราง Viewtoday
  For m = 1 To n
     Data2.Recordset.FindFirst "ID = " & m & ""
     Data2.Recordset.Edit
     Data2.Recordset.[total] = 0
     Data2.Recordset.[in1] = "-"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      Data2.Recordset.[out1] = "-"
      Data2.Recordset.[in2] = "-"
      Data2.Recordset.[out2] = "-"
      Data2.Recordset.[in3] = "-"
      Data2.Recordset.[out3] = "-"
      Data2.Recordset.[in4] = "-"
      Data2.Recordset.[out4] = "-"
      Data2.Recordset.Update
     Next m
     firstdate = DateValue(Date)
End If
Data2.Recordset.FindFirst "ID = " & id & ""
total = Data2.Recordset.total.Value
total1 = total + a
clock = FormatDateTime(Time, 4)
Data2.Recordset.Edit

If total1 = 0 Then
   Data2.Recordset.[in1] = clock
   a = a + 1
ElseIf total1 = 1 Then
   Data2.Recordset.[out1] = clock
   a = a + 1
ElseIf total1 = 2 Then
   Data2.Recordset.[in2] = clock
   a = a + 1
ElseIf total1 = 3 Then
   Data2.Recordset.[out2] = clock
   a = a + 1
ElseIf total1 = 4 Then
   Data2.Recordset.[in3] = clock
   a = a + 1
ElseIf total1 = 5 Then
   Data2.Recordset.[out3] = clock
   a = a + 1
ElseIf total1 = 6 Then
   Data2.Recordset.[in4] = clock
   a = a + 1
Else
   Data2.Recordset.[out4] = clock
   a = a + 1
End If
  Data2.Recordset.[total] = total + a
  Data2.Recordset.Update

Dim count As Integer
count = Data4.Recordset.count.Value
count = count + 1
Data4.Recordset.Edit
Data4.Recordset.[count] = count
Data4.Recordset.Update

Data3.Recordset.AddNew
Data3.Recordset.[id] = id
Data3.Recordset.[Date] = Date
Data3.Recordset.[Time] = clock
Data3.Recordset.[Picture] = "C:\My
Documents\QuickCam\Album\datapic\Pictures\picture " & count & ".jpg"

'Dim path
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
'path = "C:\My Documents\QuickCam\Album\datapic\Pictures\picture " &
count & ".jpg"
'Image2.Picture = LoadPicture(path)
Data3.Recordset.Update
End If
Debug.Print id
End If
If checkcard = True Then
   MSComm1.Output = "#  COMPLEAT/          THANK YOU$"
End If
If checkcard = False Then
   MSComm1.Output = "!"
End If
End Sub
```

```
** Form Today **

Private Sub Command1_Click()
Menu.Show
Unload Today
End Sub

Private Sub Form_Load()
Label2.Caption = Date
sqlcmd$ = "select ID,AUTHOR,IN1,OUT1,IN2,OUT2,IN3,OUT3,IN4,OUT4 from
viewtoday"
Data1.RecordSource = sqlcmd$
Data1.Refresh
End Sub

** Form Detail **

Function checkcard(receiveid) As Boolean
Dim totalid As Integer
Data3.Recordset.FindFirst "id = " & receiveid & ""
totalid = Data3.Recordset.id.Value
If receiveid = totalid Then
   checkcard = True
Else
   checkcard = False
End If
End Function

Private Sub Command1_Click()
Dim n
Dim pic
n = Text1.Text

If checkcard(n) = True Then
Data1.Recordset.FindFirst "id = " & Text1.Text & ""
Text2.Text = Data1.Recordset.author.Value
pic = Data1.Recordset.Picture.Value
Set Image1.Picture = LoadPicture(pic)
sqlcmd$ = "select Number,Id,[Date],Time from viewtotal where id =" &
n & " order by [Number]"
Data2.RecordSource = sqlcmd$
Data2.Refresh
Else
   If MsgBox("id invalid", 0) = vbOK Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      Unload Detail
      Detail.Show
   End If
End If
End Sub

Private Sub Command2_Click()
Menu.Show
Unload Detail
End Sub

Private Sub DBGrid1_Click()
Dim test
Dim pic
test = DBGrid1.Text
Data4.Recordset.FindFirst "Number = " & test & ""
Debug.Print test
pic = Data4.Recordset.Picture.Value
'Debug.Print pic
Set Image2.Picture = LoadPicture(pic)
End Sub



** Form Manage **

Function checkcard(receiveid) As Boolean
Dim totalid As Integer
Data3.Recordset.FindFirst "id = " & receiveid & ""
totalid = Data3.Recordset.id.Value
If receiveid = totalid Then
   checkcard = True
Else
   checkcard = False
End If
End Function
Private Sub Command1_Click()
Dim n As Integer
Add.Show
Unload Manage
End Sub

Private Sub Command2_Click()
Data1.Recordset.Edit
Data2.Recordset.Edit
Data1.Recordset.[id] = Text1.Text
Data1.Recordset.[author] = Text2.Text
Data2.Recordset.[id] = Text1.Text
Data2.Recordset.[author] = Text2.Text
Data1.Recordset.[born] = Text3.Text
Data1.Recordset.[e-mail] = Text4.Text
Data1.Recordset.[address] = Text5.Text
Data1.Recordset.[telephone] = Text6.Text
Data2.Recordset.Update
Data1.Recordset.Update
Text1.Text = ""
Text2.Text = ""
Text3.Text = ""
Text4.Text = ""
Text5.Text = ""
Text6.Text = ""
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End Sub

Private Sub Command3_Click()
Dim n As Integer
n = Text1.Text
Data1.Recordset.Delete
Data2.Recordset.FindFirst "id = " & Text1.Text & ""
Data2.Recordset.Delete
Text1.Text = ""
Text2.Text = ""
Text3.Text = ""
Text4.Text = ""
Text5.Text = ""
Text6.Text = ""
Data3.Recordset.Edit
n = Data3.Recordset.totalid.Value
n = n - 1
Data3.Recordset.[totalid] = n
Data3.Recordset.Update
End Sub

Private Sub Command4_Click()
Dim n
n = Text1.Text
If checkcard(n) = True Then
Data1.Recordset.FindFirst "id = " & Text1.Text & ""
Data2.Recordset.FindFirst "id = " & Text1.Text & ""
Text2.Text = Data1.Recordset.author.Value
Text3.Text = Data1.Recordset.born.Value
Text4.Text = Data1.Recordset.[e-mail].Value
Text5.Text = Data1.Recordset.address.Value
Text6.Text = Data1.Recordset.telephone.Value
Else
   If MsgBox("id invalid", 0) = vbOK Then
      Unload Manage
      Manage.Show
   End If
End If
End Sub

Private Sub Command5_Click()
Data1.Recordset.FindFirst "author = '" & Text2.Text & "'"
Data2.Recordset.FindFirst "author = '" & Text2.Text & "'"
Text1.Text = Data1.Recordset.id.Value
Text3.Text = Data1.Recordset.born.Value
Text4.Text = Data1.Recordset.[e-mail].Value
Text5.Text = Data1.Recordset.address.Value
Text6.Text = Data1.Recordset.telephone.Value
End Sub

Private Sub Command6_Click()
Text1.Text = ""
Text2.Text = ""
Text3.Text = ""
Text4.Text = ""
Text5.Text = ""
Text6.Text = ""
End Sub

Private Sub Command7_Click()
Menu.Show
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Unload Manage
End Sub


** Form Menu **


Private Sub Command1_Click()
Today.Show
Menu.Hide
End Sub

Private Sub Command2_Click()
Detail.Show
Menu.Hide
End Sub

Private Sub Command3_Click()
Manage.Show
Menu.Hide
End Sub

Private Sub Command4_Click()
Main.Show
Menu.Hide
End Sub

Private Sub Command5_Click()
Change.Show
Menu.Hide
End Sub

** Form Login **

Private Sub Command1_Click()
Dim user1
Dim pass1
Dim user2
Dim pass2
user1 = Data1.Recordset.Login.Value
pass1 = Data1.Recordset.Password.Value
user2 = Text1.Text
pass2 = Text2.Text

If user1 <> user2 Then
  If MsgBox("user invalid", 0) = vbOK Then
    Text1.Text = ""
    Text2.Text = ""
    Main.Show
    Unload Login
  End If
ElseIf pass1 <> pass2 Then
  If MsgBox("password invalid", 0) = vbOK Then
    Text1.Text = ""
    Text2.Text = ""
    Main.Show
    Unload Login
  End If
Else
Menu.Show
Unload Login
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End If
End Sub

Private Sub Command2_Click()
Forget.Show
Unload Login
End Sub
```

** Form Forget **

```
Private Sub Command1_Click()
Dim answer1
Dim answer2
answer1 = Data1.Recordset.answer.Value
answer2 = Text1.Text
If answer1 <> answer2 Then
  If MsgBox("Your answer not math", 0) = vbOK Then
    Text1.Text = ""
    Main.Show
    Unload Forget
  End If
Else
Label5.Caption = Data1.Recordset.Login.Value
Label7.Caption = Data1.Recordset.Password.Value
End If
End Sub

Private Sub Command2_Click()
Login.Show
Unload Forget
End Sub

Private Sub Form_Activate()
quesion = Data1.Recordset.quesion.Value
Label1.Caption = quesion
End Sub
```

** Form Change **

```
Private Sub Command1_Click()
Data1.Recordset.Edit
Data1.Recordset.[Login] = Text1.Text
Data1.Recordset.[Password] = Text2.Text
Data1.Recordset.[quesion] = Text3.Text
Data1.Recordset.[answer] = Text4.Text
Data1.Recordset.Update
Text1.Text = ""
Text2.Text = ""
Text3.Text = ""
Text4.Text = ""
End Sub

Private Sub Command2_Click()
Menu.Show
Unload Change
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ขอขอบคุณ อาจารย์สมยศ จุณณะปิยะ ที่กรุณาให้คำปรึกษาในการทำโครงงานเอื้อเฟื้อสถานที่ อุปกรณ์ หนังสืออ้างอิง และข้อมูลต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนังสืออ้างอิง

1. คู่มือการทำงานของเครื่องอ่านบัตรแม่เหล็ก : บริษัทอีทีที
2. คู่มือการทำงานของ บอร์ด CP-SPI RD2 : บริษัทอีทีที
3. รศ. สมยศ จุณณะปิยะ "การประยุกต์ใช้งานไมโครคอนโทรลเลอร์ MCS-51" กรุงเทพมหานคร : สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
4. ชัยวัฒน์ ลิ้มพรจิตรวิไล วรพจน์ กรแก้ววัฒนกุล "เรียนรู้และปฏิบัติการไมโครคอนโทรลเลอร์ MCS-51" กรุงเทพมหานคร : อินโนเวทีฟ เอ็กเพอริเมนต์, 2543
5. กิตติ ภัคีวัฒนะกุล จำลอง ครูอุตสาหะ "VISUAL BSIC 6 ฉบับโปรแกรมเมอร์" กรุงเทพมหานคร : ดวงกมลสมัย, 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้