สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

อุปกรณ์บัตรบันทึกข้อมูลใช้สำหรับรู้จำเพื่อควบคุมการเข้าออกประตู

Electronic data storage card equipment for user identification and entry control

T046572

โดย

นายวัชระ พิทักษ์ศฤงคาร

นางสาวสมจิต นันทิสันติกุล

นายอนุชิต หมื่นชนะสงคราม

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2544

เลขหมู่.....
เลขทะเบียน 46572
วัน, เดือน, ปี 4 เม.ย. 2546

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์บัตรบันทึกข้อมูลใช้สำหรับรู้จำผู้ใช้เพื่อควบคุมการเข้าออกประตู

Electronic data storage card equipment for user identification and entry control

โดย

นายวัชระ พิทักษ์ศฤงคาร 42015030

นางสาวสมจิต นันทิสันติกุล 42015037

นายอนุชิต หมื่นชนะสงคราม 42015042

อาจารย์ที่ปรึกษา

ผศ.ดร.สุทธิชัย นพนาคีพงษ์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2544

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2544

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง **อุปกรณ์บัตรบันทึกข้อมูลใช้สำหรับรู้จำผู้ใช้เพื่อควบคุมการเข้าออกประตู**

Electronic data storage card equipment for user identification and entry control

ผู้จัดทำ

1. นายวัชระ พิทักษ์ศฤงการ 42015030

2. นางสาวสมจิต นันทิสันติกุล 42015037

3. นายอนุชิต หมื่นชนะสงคราม 42015042

..... อาจารย์ที่ปรึกษา

(ผศ.ดร.สุทธิชัย นพนาคีพงษ์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบรักษาความปลอดภัยโดยการใช้บัตรบันทึกข้อมูล

Security System By Using The Data Storage Card

โดย นายวัชระ พิทักษ์ศฤงคาร 42015030
นางสาวสมจิต นันทิสันติกุล 42015037
นายอนุชิต หมื่นชนะสงคราม 42015042

อาจารย์ที่ปรึกษา ดร. สุทธิชัย นพนาคีพงษ์

บทคัดย่อ

โครงงานนี้เป็นการประยุกต์ใช้งานบัตรบันทึกข้อมูล โดยทำการบันทึกข้อมูลของบุคลากรภายในสำนักงานลงในบัตรโดยอาศัยไมโครคอนโทรลเลอร์เป็นตัวควบคุมการอ่านและเขียนข้อมูลและตรวจสอบ ตลอดจนทำหน้าที่ติดต่อสื่อสารข้อมูลแบบอนุกรมตามมาตรฐานการสื่อสาร RS-232C เพื่อเชื่อมต่อกับคอมพิวเตอร์ในการทำระบบฐานข้อมูลบันทึกรายชื่อบุคลากร เวลาเข้าออกสำนักงานเพื่อเพิ่มความสะดวกในการลงเวลาการทำงาน โดยไม่ต้องใช้เครื่องตอกบัตรและยังสามารถตรวจสอบข้อมูลดังกล่าวได้หากเกิดการสูญหายของข้อมูลที่สำคัญของบริษัท

Abstract

This project is the application of the data storage card to store officers' affair into the card. The microcontroller MCS-51 is utilized to control and check in reading and writing of the card. It is also used to communicate in the serial data communication RS-232C to a computer to make the database system such as name, time in and time out. The further comfortableness of this project is more useful than using the timecards. Beneficially, it can check officers' affair even though the documents are stolen.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทที่ 1 บทนำ	1
บทที่ 2 ทฤษฎีและหลักการ	2
2.1 ชนิดของการ์ด	2
2.1.1 การ์ดที่มีรอยบาก	2
2.1.2 การ์ดแถบแม่เหล็ก	3
2.1.3 สมาร์ตการ์ด	5
2.1.3.1 การ์ดบันทึกข้อมูล	5
2.1.3.2 การ์ดประมวลผล	7
2.1.3.3 สมาร์ตการ์ดที่ไม่มีหน้าสัมผัส	8
2.1.4 ออปติคอลเมมโมรีการ์ด	10
2.2 รูปแบบของบัตร	11
2.3 ส่วนของตัวบัตร	15
2.4 ชิปโมดูล	16
2.5 คุณสมบัติทางไฟฟ้าของสมาร์ตการ์ด	17
2.5.1 การติดต่อกับบัตร	17
2.5.2 แรงดันที่จ่ายให้บัตร	18
2.5.3 แหล่งจ่ายกระแส	19
2.5.4 สัญญาณนาฬิกาจากภายนอก	21
2.5.5 การส่งถ่ายข้อมูล	21
2.6 ชนิดของหน้าสัมผัสของสมาร์ตการ์ด	22
2.7 ระดับชั้นในการส่งทางกายภาพ	25
2.8 การส่งผ่านข้อมูลแบบซิงโครนัส	26
2.9 โปรโตคอลของบัตรโทรศัพท์	27
2.10 สถาปัตยกรรมของไมโครคอนโทรลเลอร์แบบชิพเดี่ยวตระกูล MCS-51	29
2.10.1 การเชื่อมต่ออุปกรณ์แบบอนุกรม	30
2.10.2 โครงสร้างภายในของตัวควบคุมโมดูล	32
2.11 ซอฟแวร์ที่ใช้ในการติดต่อสื่อสารผ่านพอร์ทอนุกรม	38
2.12 การจัดการโครงสร้างฐานข้อมูล	44
2.12.1 อ็อปเจ็คทางด้านฐานข้อมูล	44
2.12.2 โครงสร้างฐานข้อมูลแบบเจ็ท	47
2.12.3 อ็อปเจ็คข้อมูลและการจัดการกับข้อมูล	48
2.12.4 อ็อปเจ็คข้อมูล	48

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.12.5 การเคลื่อนตำแหน่งตัวชี้ตำแหน่งข้อมูล 50
2.12.6 การอ่านและกำหนดค่าของฟิลด์ในเรคคอร์ด 52
บทที่ 3 การคำนวณและการสร้าง 57
3.1 ส่วนที่ทำการประมวลผล 58
3.1.1 ส่วนประมวลผลกลาง 58
3.1.2 ส่วนรับค่าจากคีย์บอร์ด 58
3.1.3 ส่วนแสดงผล 60
3.1.4 ส่วนที่ทำการติดต่อกับคอมพิวเตอร์ 60
3.2 ส่วนที่ติดต่อกับสมาร์ตการ์ด 61
3.2.1 ไทม์มิ่งไดอะแกรมของการทำงานของบัตรสมาร์ตการ์ด 62
3.3 แผนผังแสดงการทำงานของโปรแกรม 63
บทที่ 4 การทดลองและผลการทดลอง 84
4.1 ส่วนของการแสดงผลทาง LCD โมดูล 84
4.2 ส่วนของการเขียนข้อมูลลงในบัตร 88
4.3 ผลจากการอ่านข้อมูลจากบัตร TOT Card จำนวน 74 ใบ 89
4.4 ส่วนที่ทำการติดต่อกับคอมพิวเตอร์ 108
4.5 ส่วนแสดงผลทางคอมพิวเตอร์ 103
บทที่ 5 บทวิจารณ์และบทสรุป 110
5.1 สรุปผลการทดลอง 110
5.2 วิจารณ์ผลการทดลอง 110
5.3 แนวทางการพัฒนา 110
กิตติกรรมประกาศ
หนังสืออ้างอิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

หน้า

รูปที่ 2.1 แสดงตำแหน่งของรอยนูน 3
รูปที่ 2.2 แสดงตำแหน่งของแถบแม่เหล็ก 4
รูปที่ 2.3 แสดงตำแหน่งของแต่ละแถบแม่เหล็ก 4
รูปที่ 2.4 แสดงลำดับชั้นของการ์ดที่มีชิปบรรจุอยู่ 5
รูปที่ 2.5 โครงสร้างทั่วไปของการ์ดบันทึกข้อมูลที่มีหน้าสัมผัส 6
รูปที่ 2.6 โครงสร้างทั่วไปของการ์ดบันทึกข้อมูลและส่วนเชื่อมต่อที่ไม่มีหน้าสัมผัส 6
รูปที่ 2.7 โครงสร้างทั่วไปของการ์ดประมวลผลที่มีหน้าสัมผัส 7
รูปที่ 2.8 โครงสร้างทั่วไปของการ์ดประมวลผลที่มีตัวประมวลผลร่วมและส่วนต่อที่ไม่มีหน้าสัมผัส 8
รูปที่ 2.9 โครงสร้างทั่วไปของการ์ดประมวลผลที่มีตัวประมวลผลร่วมและส่วนเชื่อมต่อทั้งแบบมีหน้าสัมผัสและไม่มีหน้าสัมผัส 9
รูปที่ 2.10 โครงสร้างทั่วไปที่มีการรวมกันของการ์ดบันทึกข้อมูลแบบไม่มีหน้าสัมผัสและส่วนเชื่อมต่อทั้งสองแบบ 10
รูปที่ 2.11 แสดงตำแหน่งของพื้นที่ที่เก็บข้อมูลของบัตร 11
รูปที่ 2.12 ขนาดของบัตร รูปแบบ ID-1 12
รูปที่ 2.13 แสดงความกว้าง ยาว หนาและรัศมีมุมของ ID-1 13
รูปที่ 2.14 แสดงความกว้าง ยาว หนาและรัศมีมุมของ ID-000 13
รูปที่ 2.15 แสดงความกว้าง ยาว หนาและรัศมีมุมของ ID-00 14
รูปที่ 2.16 ขนาดที่สัมพันธ์กันระหว่างรูปแบบ ID-1, ID-00 และ ID-000 14
รูปที่ 2.17 ไดอะแกรมการแบ่งประเภทสำหรับส่วนต่าง ๆ ของบัตร 15
รูปที่ 2.18 การแบ่งชนิดต่าง ๆ ของชิปโมดูล 16
รูปที่ 2.19 การกำหนดหน้าที่ทางไฟฟ้าและหมายเลขของหน้าสัมผัสของสมาร์ตการ์ดตามมาตรฐาน ISO7816-2 17
รูปที่ 2.20 ไดอะแกรมในการทำงานของขั้นตอนในการเลือกแหล่งจ่ายแรงดันให้แก่บัตร 19
รูปที่ 2.21 แสดงการกินกระแสของไมโครคอนโทรลเลอร์ ต่อความถี่สัญญาณนาฬิกาและอุณหภูมิ 20
รูปที่ 2.22 วงจรของช่องทางสื่อสารอินพุท/เอาท์พุท ระหว่างปลายสายและสมาร์ตการ์ด 22
รูปที่ 2.23 ตำแหน่งของหน้าสัมผัสที่สัมพันธ์กับโครงร่างของตัวบัตร 23
รูปที่ 2.24 ขนาดของหน้าสัมผัสต่ำสุดที่ระบุไว้ตาม ISO7816-2 24
รูปที่ 2.25 รูปแบบการเรียงที่เป็นไปได้ระหว่างชิปส่วนที่ถูกตอกให้นูนและแถบแม่เหล็กตามมาตรฐาน ISO7816-2 25
รูปที่ 2.26 รูปแบบของทรานสมิทชั่นโปรโคคอลที่บัตรสมาร์ทการ์ดใช้ 25
รูปที่ 2.27 การรีเซ็ตตัวชี้ตำแหน่งแอดเดรสไปที่ศูนย์ 28

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.28 แสดงการเพิ่มตำแหน่งแอดเดรสและการอ่านข้อมูลจากแอดเดรส 28
รูปที่ 2.29 แสดงการเขียนข้อมูลลงในหน่วยความจำ EEPROM 29
รูปที่ 2.30 แสดงบล๊อคไดอะแกรมการทำงานภายในของโมดูลแบบตัวอักษร 32
รูปที่ 2.31 แสดงการจำขาของโมดูล LCD 33
รูปที่ 2.32 อ็อปเจ็คโปรแกรมย่อยในฐานข้อมูล 46
รูปที่ 3.1 บล๊อคไดอะแกรมของระบบรักษาความปลอดภัยโดยใช้บัตรบันทึกข้อมูล 57
รูปที่ 3.2 แสดงการต่อวงจรของไมโครคอนโทรลเลอร์ 58
รูปที่ 3.3 แสดงการต่อคีย์บอร์ด 59
รูปที่ 3.4 แสดงการต่อ LCD โมดูล 60
รูปที่ 3.5 แสดงการต่อ IC MAX 232 61
รูปที่ 3.6 แสดงการเชื่อมต่อหน้าสัมผัสของบัตร 61
รูปที่ 3.7 ไทม์มิ่งไดอะแกรมของการทำงานของบัตรสมาร์ตการ์ดแบบซิงโครนัส 62
รูปที่ 3.8 แสดงโฟล์วชาร์ทการอ่านข้อมูลจากบัตร 64
รูปที่ 3.9 แสดงโฟล์วชาร์ทการเขียนข้อมูลลงในบัตร 66
รูปที่ 3.10 แสดงโฟล์วชาร์ทในการส่งข้อมูล ไปยัง LCD 68
รูปที่ 3.11 แสดงโฟล์วชาร์ทการเขียนข้อมูลลงใน IC DS1307 71
รูปที่ 3.12 แสดงโฟล์วชาร์ทของการอ่านข้อมูลจาก IC DS1307 73
รูปที่ 3.13 แสดงโฟล์วชาร์ทของการส่งข้อมูล 8 บิทไปยัง IC DS1307 74
รูปที่ 3.14 แสดงโฟล์วชาร์ทของการรับข้อมูล 8 บิทจาก IC DS1307 75
รูปที่ 3.15 แสดงโฟล์วชาร์ทการทำงานหลักของโปรแกรม 76
รูปที่ 3.16 แสดงโฟล์วชาร์ทของการตรวจสอบการเข้า-ออก ของพนักงาน 77
รูปที่ 3.17 แสดงโฟล์วชาร์ทของ event การค้นหาข้อมูล 78
รูปที่ 3.18 แสดงโฟล์วชาร์ทของ event การสร้างบัตร 80
รูปที่ 3.19 แสดงโฟล์วชาร์ทของ event การคำนวณเงินเดือน 81
รูปที่ 3.20 แสดงเครื่องต้นแบบของเครื่องอ่านและเขียนบัตรสมาร์ตการ์ด 82
รูปที่ 3.21 แสดงส่วนหน้าของเครื่องต้นแบบ 82
รูปที่ 3.22 แสดงส่วนหลังของเครื่องต้นแบบ 83
รูปที่ 4.1 แสดงผลการทำงานของ LCD โมดูล 84
รูปที่ 4.2 แสดงผลที่ได้จากการกดคีย์บอร์ด 84
รูปที่ 4.3 แสดงผลการทดลองการวัดสัญญาณนาฬิกาและสัญญาณข้อมูล 85
รูปที่ 4.4 แสดงผลการทดลองการวัดสัญญาณนาฬิกาและสัญญาณรีเซ็ท 86
รูปที่ 4.5 แสดงผลการทดลองการวัดสัญญาณรีเซ็ทและสัญญาณข้อมูล 87
รูปที่ 4.6 แสดงผลการทดลองการวัดสัญญาณนาฬิกาขณะทำการเขียนข้อมูลลงในบัตร 88
รูปที่ 4.7 แสดงผลการทดลองการส่งสัญญาณเข้าคอมพิวเตอร์ 92

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.8 แสดงผลที่คอมพิวเตอร์รับข้อมูลเข้ามาได้ 92
รูปที่ 4.9 แสดงผลการทำงานของหน้าจอหลัก 93
รูปที่ 4.10 แสดงผลการคำนวณเงินเดือนของพนักงานในแต่ละเดือน 94
รูปที่ 4.11 แสดงผลการค้นหารายชื่อของพนักงานโดยบุคคลนั้นยังไม่มีบัตร 95
รูปที่ 4.12 แสดงผลการค้นหารายชื่อของพนักงานโดยบุคคลนั้นมีบัตรแล้ว 96
รูปที่ 4.13 แสดงการเพิ่มรายชื่อของพนักงาน 97
รูปที่ 4.14 แสดงการสร้างหรือกำหนดบัตรบันทึกข้อมูลให้แก่พนักงานที่ยังไม่มีบัตร 98
รูปที่ 4.15 การสร้างหรือกำหนดบัตรบันทึกข้อมูลให้แก่พนักงานที่มีบัตรแล้ว 99

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่ 2.1 แสดงลักษณะการทำงานของแต่ละแถบ 4

ตารางที่ 2.2 การออกแบบและหน้าที่ของหน้าสัมผัสตามมาตรฐาน ISO7816 18

ตารางที่ 2.3 ผลสรุปของทรานสมิทชั่นโปรโตคอล 25

ตารางที่ 2.4 แสดงความสามารถของขา RS, R / $\overline{W}$ และ E 33

ตารางที่ 2.5 แสดงค่าคงที่ใช้ในการระบุถึงประเภทอ็อปเจ็คข้อมูล 50

ตารางที่ 2.6 วิธีการเลื่อนตำแหน่งตัวชี้ข้อมูล 51

ตารางที่ 2.7 ฟังก์ชันการทำงานสำหรับการค้นหาข้อมูล 53

ตารางที่ 3.1 แสดงค่าความจริงของ IC Key Encoder 59

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

ในปัจจุบันเทคโนโลยีของวงจรรวม (Integrated Circuit) ได้พัฒนาไปมาก ถึงขนาดมีการย่อขนาดของวงจรรวมลงมาอยู่บนบัตร ที่เรียกกันว่าสมาร์ตการ์ด (Smart Card) จึงได้รับความนิยมที่จะใช้การ์ดที่บรรจุชิปไอซีแทนบัตรแถบแม่เหล็กในงานด้านต่างๆ มากขึ้นเรื่อย ๆ เนื่องจากความสามารถที่เหนือกว่าบัตรแถบแม่เหล็ก (Magnetic Strip Card) ในหลาย ๆ ด้าน เช่น ความสามารถในการประมวลผลข้อมูลได้ในตัวเอง ไม่มีผลกระทบของสนามแม่เหล็ก ขนาดของหน่วยความจำ สมาร์ตการ์ดจึงได้ถูกนำมาใช้งานอย่างแพร่หลายไม่ว่าจะใช้ในเชิงธุรกิจ การเก็บข้อมูล การรักษาความปลอดภัยและการเก็บคุณสมบัติของสิ่งต่าง ๆ เป็นต้น

ในโครงงานนี้จะใช้สมาร์ตการ์ด มาใช้งานเป็นบัตรประจำตัวพนักงานเพื่อใช้ในการเข้า-ออก ภายในองค์กรหรือบริษัท โดยเมื่อนำบัตรเข้าสู่ระบบ (Login) ระบบจะทำการตรวจสอบว่าเป็นใครและสอบถามรหัส หลังจากที่ใส่รหัสผ่านระบบจะทำการบันทึกว่าใครเข้าสู่ระบบ ณ เวลานั้นลงสู่สมาร์ตการ์ดและระบบ เพื่อเอาไว้ตรวจสอบหากเกิดปัญหาใดในองค์กรนั้น ๆ และเมื่อออกจากระบบ (Logout) ระบบก็จะทำการปลดสมาชิกออกจากการตรวจสอบเวลาการทำงาน และจดจำไว้ว่าออกจากระบบครั้งล่าสุดเมื่อใด ซึ่งข้อมูลต่าง ๆ จะเป็นประโยชน์อย่างมากเพราะสามารถนำไป จัดการกับการทำงานต่าง ๆ ได้ เช่น ตรวจสอบเวลาการทำงานเพื่อทำการคิดเงินเดือน สามารถทราบได้ว่าใครมาทำงาน-กลับเมื่อใด ซึ่งจะทำให้สามารถตรวจสอบได้ง่ายหากเกิดปัญหาขึ้นกับองค์กรหรือบริษัทนั้น นอกจากนี้ยังนำข้อมูลเหล่านี้ไปวิเคราะห์หรือใช้งานอย่างอื่นแล้วแต่ความประสงค์ขององค์กรหรือบริษัทนั้น

นี่เป็นเพียงส่วนหนึ่งของการนำสมาร์ตการ์ดมาใช้งาน ซึ่งสามารถนำไปใช้งานได้ต่าง ๆ แล้วแต่จุดประสงค์และความต้องการของงานนั้น ๆ แต่สิ่งสำคัญคือจะต้องมี เครื่องอ่านและเขียนสมาร์ตการ์ด ซึ่งโครงงานนี้ได้ทำการศึกษาและทดลองจัดทำขึ้นจึงจะสามารถทำงานได้เต็มที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการ

2.1 ชนิดของการ์ด (Types of Card)

สมาร์ตการ์ดเป็นการ์ดตัวใหม่ของการ์ดที่ใช้รูปแบบ ID-1 ที่กำหนดโดยมาตรฐาน ISO 7810 (ลักษณะทางกายภาพ) ซึ่งมาตรฐานนี้จะระบุคุณสมบัติทางฟิสิกส์ของการ์ด เช่น ความต้านทานอุณหภูมิ มิติของการ์ดที่แตกต่างกัน 3 รูปแบบ (ID-1, ID-2 และ ID-3) มาตรฐานของสมาร์ตการ์ดจะใช้หลักของการ์ด ID-1 ที่ปัจจุบันส่วนใหญ่ใช้สำหรับธุรกิจการเงิน

ในบทนี้จะกล่าวถึงการเปลี่ยนแปลงชนิดของการ์ดในรูปแบบ ID-1 การรวมฟังก์ชันพิเศษที่น่าสนใจ โดยเฉพาะการ์ดที่ใช้ในระบบปัจจุบัน เช่น การ์ดแถบแม่เหล็กจะถูกแทนที่ด้วยสมาร์ตการ์ด ในกรณีเช่นนี้ แสดงให้เห็นว่าเป็นไปไม่ได้ที่จะแทนที่โครงสร้างต่ำกว่าที่ใช้ในปัจจุบัน

ทางออกกว้าง ๆ ของปัญหานี้เกิดจากผลของการ์ดที่มีทั้งแถบแม่เหล็กและชิป ที่ถูกใช้ในระหว่างช่วงเวลาหนึ่ง เช่น การ์ดสามารถใช้ได้ทั้งอุปกรณ์ต่อพ่วงทั้งเก่าและใหม่ โดยปกติฟังก์ชันใหม่อาจจะใช้กับชิปได้อย่างเดียวไม่สามารถใช้ประโยชน์กับอุปกรณ์ต่อพ่วงแถบแม่เหล็กได้

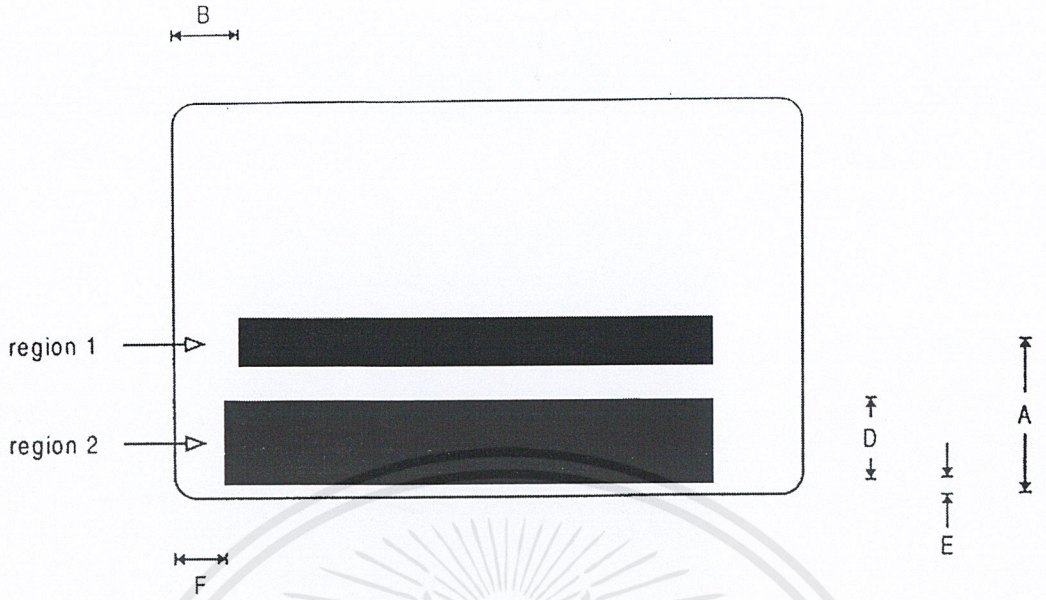
2.1.1 การ์ดที่มีรอยนูน (Embossed Card)

การสร้างรอยนูน เป็นเทคนิคเก่าสำหรับใช้กับเครื่องอ่านเครื่องหมายเพื่อบอกลักษณะของการ์ด ลักษณะของรอยนูนบนการ์ดสามารถเปลี่ยนจากการใช้กระดาษธรรมดาเป็นอุปกรณ์ที่มีราคาแพงได้ การอ่านของรอยนูนเป็นแบบตรงไปตรงมา ลักษณะและตำแหน่งของรอยนูนถูกกำหนดโดยมาตรฐาน ISO 7811 (เทคนิคการบันทึก) ซึ่งมาตรฐานนี้จะแบ่งเป็น 5 ส่วน โดยเกี่ยวข้องกับแถบแม่เหล็กพร้อมกับรอยนูน

ISO 7811 ส่วนที่ 1 กำหนดลักษณะของรอยนูน เช่น รูปแบบของมัน ขนาด ความสูงของรอยนูน ส่วนที่ 3 จะกำหนดลักษณะตำแหน่งที่แน่นอนบนการ์ด และจะกำหนดแยกเป็น 2 แถว ดังแสดงในรูปที่ 2.1 แถวที่ 1 ใช้สำหรับตัวเลขของการ์ดที่จะบอกจำนวนที่ใช้การ์ดพร้อมกับผู้ถือการ์ด แถบที่ 2 ใช้สำหรับเพิ่มข้อมูลเกี่ยวกับผู้ถือการ์ด เช่น ชื่อและที่อยู่ เป็นต้น

แม้จะดูเผิน ๆ เหมือนว่าลักษณะการพิมพ์รอยนูนจะดูโบราณ แต่มันก็ทำให้เครดิตการ์ดแพร่หลายไปทั่วโลก แม้แต่ในประเทศที่กำลังพัฒนาก็ตาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 แสดงตำแหน่งของรอยนูน

2.1.2 การ์ดแถบแม่เหล็ก (Magnetic Strip Card)

ข้อเสียเดิมของการ์ดที่มีรอยนูนคือ การที่มันถูกประดิษฐ์ขึ้นโดยต้องใช้กระดาษจำนวนมาก ซึ่งเป็นการสิ้นเปลืองในการทำ การแก้ไขของปัญหานี้ก็คือทำการเข้ารหัสดิจิตอลข้อมูลบนการ์ดลงบนแถบแม่เหล็กที่อยู่ด้านหลังของการ์ดแทน

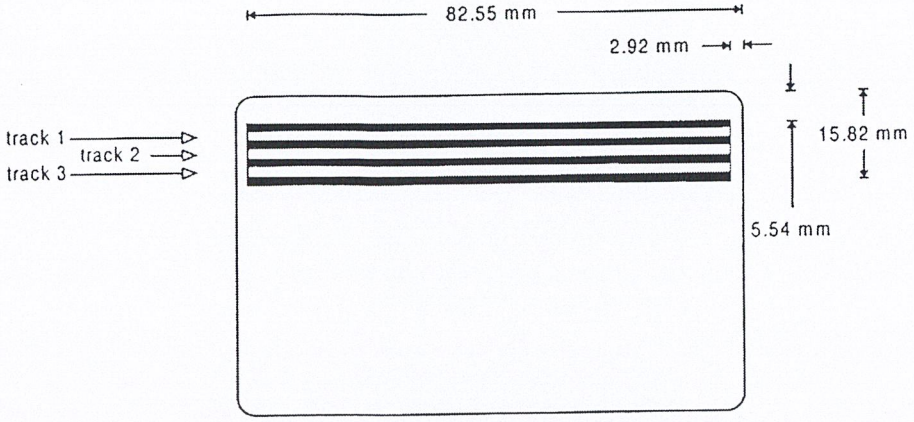
แถบแม่เหล็กถูกอ่านโดยการดึงข้อมูลด้วยหัวอ่าน โดยใช้ทั้งคนและเครื่องจักร ข้อมูลจะถูกอ่านและเก็บในรูปของสัญญาณไฟฟ้า ซึ่งการประมวลผลไม่ต้องใช้กระดาษมากมาย

ส่วน 2, 4 และ 5 ของมาตรฐาน ISO 7811 จะระบุคุณสมบัติของแถบแม่เหล็ก เทคนิคการเข้าโค้ดและตำแหน่งของแถบแม่เหล็ก แถบแม่เหล็กจะแบ่งเป็น 3 แทร็ก แทร็กที่ 1 และ 2 จะเป็นแทร็กที่ใช้สำหรับการอ่านข้อมูลเพียงอย่างเดียว ส่วนแทร็กที่ 3 จะใช้สำหรับการเขียนด้วย

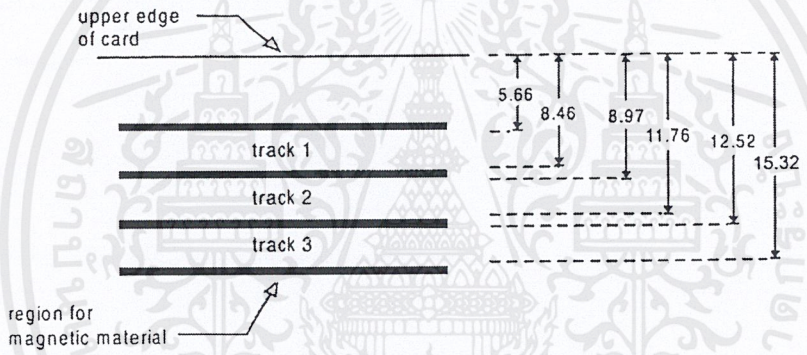
แม้ว่าความจุในการเก็บข้อมูลของแถบแม่เหล็กจะมีประมาณ 1000 บิต ซึ่งก็ไม่มากนัก แต่ก็ยังมากกว่าการ์ดแบบมีรอยนูน อีกทั้งยังสามารถอ่านและเขียนได้ในแทร็กที่ 3

แต่ปัญหาหลักของเทคนิคแถบแม่เหล็กคือ ข้อมูลที่เก็บไว้จะถูกแก้ไขได้ง่าย การแก้ไขการ์ดแบบมีรอยนูนนั้นจะต้องใช้ความชำนาญและยังสามารถตรวจพบได้ง่ายจากการมอง แต่การเปลี่ยนข้อมูลบนแถบแม่เหล็กนั้นง่ายเพราะอุปกรณ์อ่านเขียนที่เป็นมาตรฐาน และยังยากที่จะรู้ว่าข้อมูลถูกแก้ไข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 แสดงตำแหน่งของแถบแม่เหล็ก



รูปที่ 2.3 แสดงตำแหน่งของแต่ละแถบ

Standard feature of the three tracks on a magnetic-strip card, as specified in ISO 7811

Feature	Track 1	Track 2	Track 3
Amount of data	79 characters max	40 characters max	107 characters max
Data coding	6-bit alphanumeric	4-bit BCD	4-bit BCD
Data density	210 bpi (8.3 bit/mm)	75 bpi (3bit/mm)	210 bpi (8.3 bit/mm)
Writing	Not allowed	Not allowed	Allowed

ตารางที่ 2.1 แสดงลักษณะการทำงานของแต่ละแถบ

การ์ดแถบแม่เหล็กถูกพัฒนาขึ้นเพื่อป้องกันการปลอมและแก้ไขข้อมูล ตัวอย่างการ์ดในเยอรมันจะไม่เห็นและไม่สามารถแก้ไขโค้ดในตัวการ์ดได้ นี่ก็เป็นวิธีการหนึ่งที่ต้องใช้ตัวเซนเซอร์พิเศษ ซึ่งราคาก็จะแพงขึ้นตามไปด้วย ด้วยเหตุผลนี้จึงไม่เป็นที่นิยมทั่วไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.3 สมาร์ตการ์ด (Smart Card)

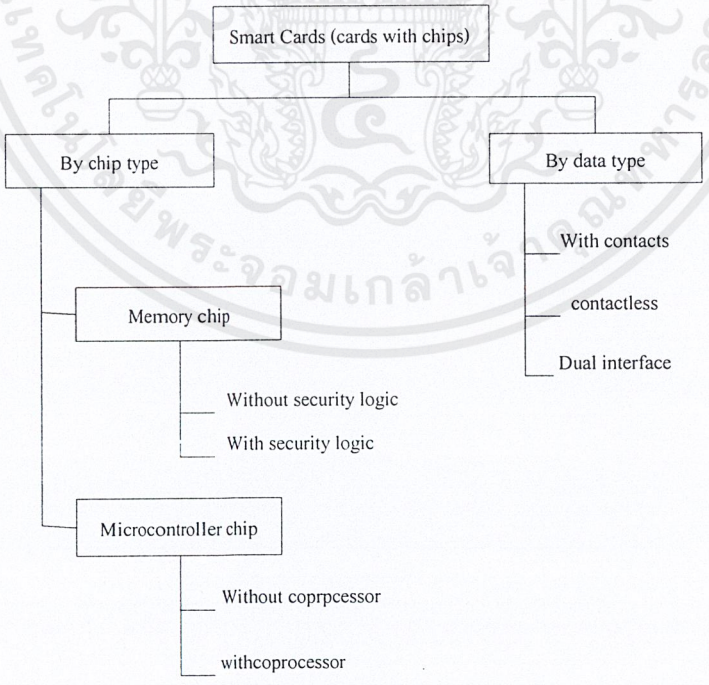
สมาร์ตการ์ดเป็นผลิตภัณฑ์ใหม่ที่สุดและฉลาดที่สุดของการ์ดที่ใช้รูปแบบ ID-1 ลักษณะเด่นของมันคือการฝังวงจร ซึ่งเป็นส่วนที่ใช้ติดต่อสื่อสาร การเก็บข้อมูล การประมวลผลไว้ในการ์ด การโอนถ่ายข้อมูลสามารถทำได้โดยผ่านหน้าสัมผัสบนผิวของการ์ด หรือผ่านสนามแม่เหล็กไฟฟ้าโดยไม่ต้องใช้หน้าสัมผัสก็ได้

ข้อดีที่สำคัญเมื่อเปรียบเทียบกับการ์ดแถบแม่เหล็กก็คือ จะมีความจุในการเก็บข้อมูลที่มากกว่า ข้อดีอีกประการหนึ่งของสมาร์ตการ์ดก็คือ การ์ดสามารถป้องกันข้อมูลได้ การเข้าถึงข้อมูลจะต้องผ่านการอินเตอร์เฟสข้อมูลแบบอนุกรมเท่านั้น โดยถูกควบคุมจากระบบปฏิบัติการและระบบความปลอดภัย การเขียนข้อมูลลงในการ์ดจะไม่สามารถเขียนจากภายนอกได้ ข้อมูลจะถูกประมวลผลภายในเท่านั้นโดยหน่วยประมวลผลของชิป ความสามารถในการเขียน การลบ การอ่านจะถูกจำกัดและถูกระบุเงื่อนไขจากทั้งซอฟแวร์และฮาร์ดแวร์ ซึ่งกลไกในการรักษาความปลอดภัยจะยอมให้ปฏิบัติได้ สมาร์ตการ์ดยังมีอายุการใช้งานที่สูงกว่าเมื่อเทียบกับการ์ดแถบแม่เหล็กที่มีอายุการใช้งานเพียง 1-2 ปี

สมาร์ตการ์ดถูกแบ่งออกเป็น 2 กลุ่ม ตามความแตกต่างทั้งราคาและการใช้งาน โดยจะแบ่งเป็นการ์ดบันทึกข้อมูลและการ์ดประมวลผล

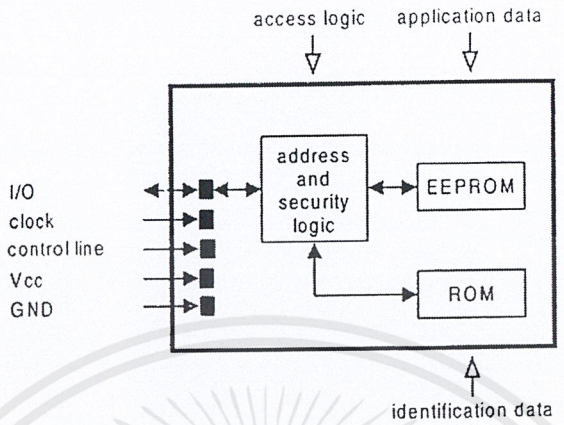
2.1.3.1 การ์ดบันทึกข้อมูล (Memory Card)

จากรูปที่ 2.5 และ 2.6 แสดงให้เห็นถึงบล็อคไดอะแกรมของการ์ดบันทึกข้อมูล



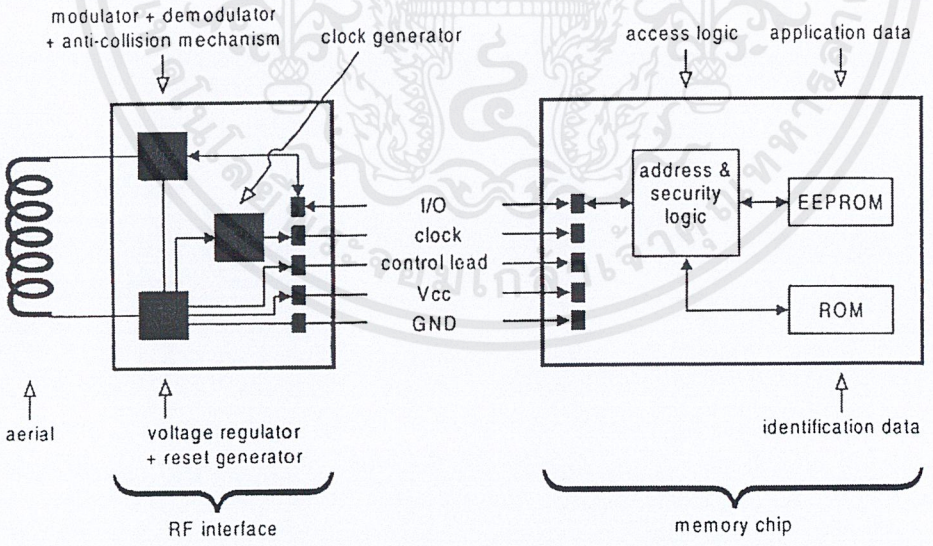
รูปที่ 2.4 แสดงลำดับชั้นของการ์ดที่มีชิปบรรจุอยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 โครงสร้างทั่วไปของการ์ดบันทึกข้อมูลที่มีหน้าสัมผัส

ข้อมูลจะถูกเก็บไว้ในหน่วยความจำ EEPROM การเข้าถึงข้อมูลถูกควบคุมโดยหน่วยรักษาความปลอดภัยพื้นฐาน คือการป้องกันการเขียนและลบของหน่วยความจำ ข้อมูลจะถูกโอนย้ายผ่าน I/O port สมาร์ตการ์ดบางส่วนจะใช้ บัส I²C ที่ใช้ปกติกับการเข้าถึงหน่วยความจำแบบอนุกรม



รูปที่ 2.6 โครงสร้างทั่วไปของการ์ดบันทึกข้อมูลและส่วนเชื่อมต่อที่ไม่มีหน้าสัมผัส

หน้าที่ของการ์ดบันทึกข้อมูลส่วนมากจะเหมาะสำหรับการใช้ที่เฉพาะเจาะจง การใช้งานที่จำกัดก็ทำให้มันมีราคาที่ไม่แพงเกินไป ซึ่งส่วนใหญ่จะใช้เป็นการ์ดโทรศัพท์และการ์ดรับรองสุขภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

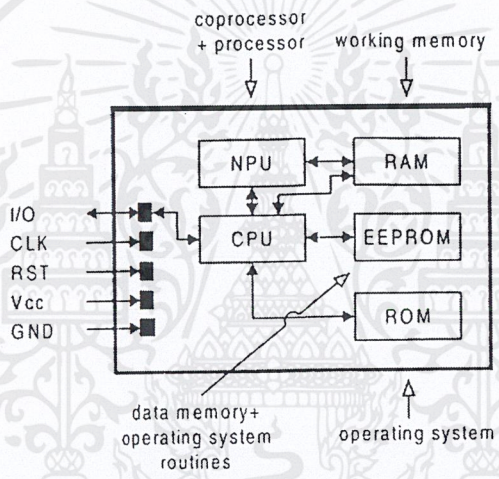
2.1.3.2 การ์ดประมวลผล (Microprocessor Card)

ชิปในไมโครโปรเซสเซอร์การ์ดจะเป็นตัวประมวลผลจะประกอบไปด้วยบล็อคที่ทำหน้าที่ต่าง ๆ มี ROM, EEPROM, RAM และ I/O port ดังรูปที่ 2.7 แสดง ให้เห็นถึงโครงสร้างของส่วนประกอบต่าง ๆ

ใน ROM จะบรรจุไปด้วยระบบปฏิบัติการของชิป ที่จะทำงานเมื่อชิปถูกกระทำ ข้อจำกัดของ ROM คือมันจะไม่สามารถเปลี่ยนแปลงข้อมูลได้

EEPROM เป็นหน่วยความจำที่ไม่สามารถเปลี่ยนแปลงได้ง่าย ข้อมูลและโปรแกรมจะถูกเขียนและอ่านจาก EEPROM ภายใต้การควบคุมจากระบบปฏิบัติการ

RAM เป็นหน่วยความจำที่สามารถเปลี่ยนแปลงได้ง่าย และข้อมูลทั้งหมดที่เก็บไว้จะหายไปเมื่อตัดแหล่งจ่ายไฟออกจากชิป



รูปที่ 2.7 โครงสร้างทั่วไปของการ์ดประมวลผลที่มีหน้าสัมผัส

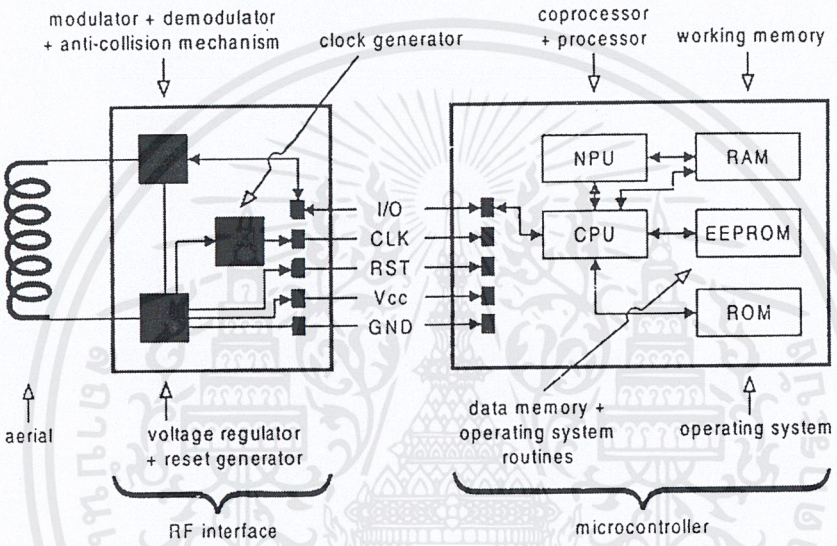
การ์ดประมวลผลมีประโยชน์มากมาย สมาร์ตการ์ดในปัจจุบันสามารถใช้งานได้หลากหลายบนการ์ดใบเดียวกัน ROM จะถูกบรรจุโครงสร้างพื้นฐานของระบบปฏิบัติการ ส่วนโปรแกรมที่ใช้เฉพาะจะถูกบรรจุไว้ใน EEPROM หลังจากที่การ์ดถูกผลิตออกมา ฮาร์ดแวร์และซอฟแวร์พิเศษจะถูกตรวจสอบให้แน่ใจว่าจะสามารถป้องกันข้อมูลได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.3.3 สมาร์ตการ์ดที่ไม่มีหน้าสัมผัส (Contactless Smart Card)

สมาร์ตการ์ดจะใช้หน้าสัมผัสทั้งหมด 8 ส่วน ตามที่ระบุตามมาตรฐาน ISO 7816 ส่วนที่ 1 ชนิดของหน้าสัมผัสของสมาร์ตการ์ดจะแน่นอนและเที่ยงตรง

ปัญหาทางเทคนิคจะถูกแก้ไขโดยสมาร์ตการ์ดที่ไม่มีหน้าสัมผัส โดยการเพิ่มเทคนิคที่เป็นข้อดีเข้าไป เช่น หน้าสัมผัสไม่จำเป็นต้องใส่เข้าไปในเครื่องอ่านการ์ด โดยระบบจะสามารถทำงานได้ที่ระยะห่าง 1 เมตร นี่เป็นข้อดีสำหรับระบบควบคุมการเข้าออก ในส่วนของประตูที่ต้องถูกปิดเปิด จากแต่ก่อนที่ประตูส่วนบุคคลจะต้องตรวจเช็ค โดยการ์ดจะต้องถูกนำออกจากกระเป๋า แล้วใส่เข้าไปในหัวอ่าน

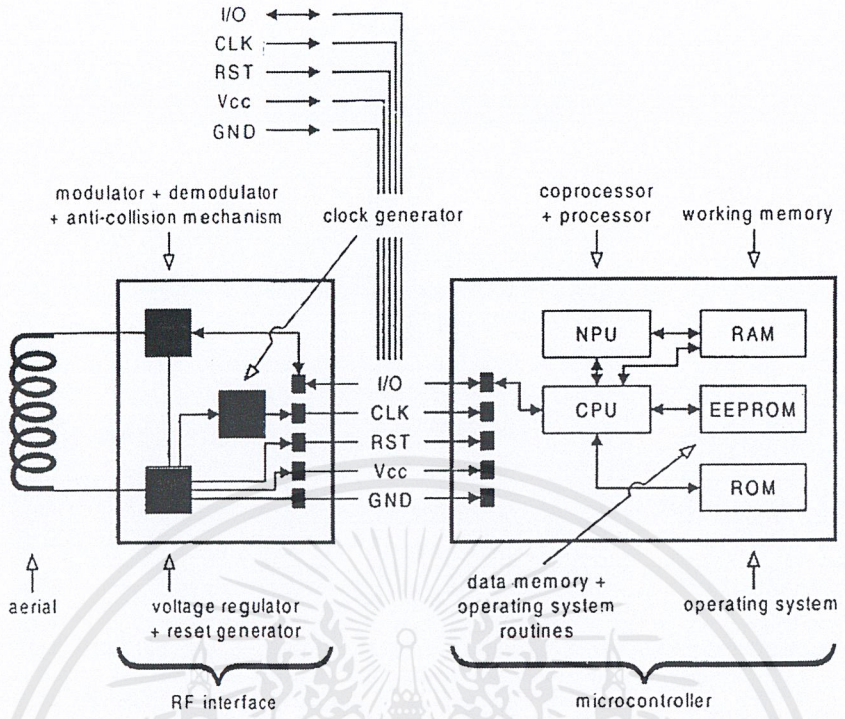


รูปที่ 2.8 โครงสร้างทั่วไปของการ์ดประมวลผลที่มีตัวประมวลผลร่วมและส่วนต่อที่ไม่มีหน้าสัมผัส

แต่อย่างไรก็ตาม เทคโนโลยีหน้าสัมผัสนี้ก็ยังใช้ประโยชน์ในระบบที่การ์ดต้องใส่เข้าในหัวอ่าน ซึ่งมันก็ไม่สำคัญว่าการ์ดจะต้องใส่เข้าในเครื่องอ่านการ์ด มันเป็นการเทียบกับบัตรแถบแม่เหล็ก หรือการ์ดกับหน้าสัมผัส

การเปลี่ยนแปลง ที่น่าสนใจในการใช้การ์ดที่ไม่มีหน้าสัมผัสเกี่ยวกับการใช้อุปกรณ์ต่อพ่วงกับหน้าสัมผัสเป็นการ์ดที่ไม่ต้องใส่ในสล๊อตแต่จะมีตำแหน่งที่ถูกกำหนดไว้ง่าย ๆ บนหน้าสัมผัสของหัวอ่าน ซึ่งเป็นการใช้ที่ง่ายขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

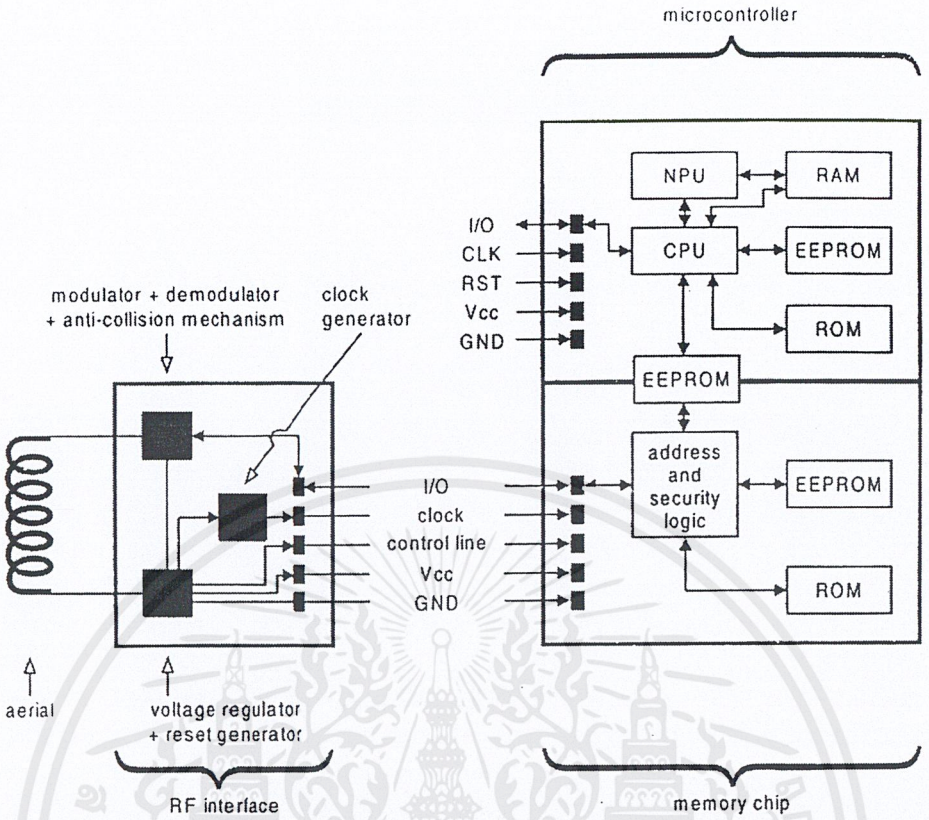


รูปที่ 2.9 โครงสร้างทั่วไปของการ์ดประมวลผลที่มีตัวประมวลผลร่วมและส่วนเชื่อมต่อทั้งแบบมีหน้าสัมผัสและไม่มีหน้าสัมผัส

เทคโนโลยีที่คิดขึ้นเพื่อให้สามารถใช้ประโยชน์ได้มากมาย ในขณะที่ราคาก็ไม่แตกต่างเกินไปนัก หากเปรียบเทียบกับการ์ดชนิดที่มีหน้าสัมผัส ในปัจจุบันการ์ดที่ไม่มีหน้าสัมผัสถูกใช้อย่างมากมายในระบบการขนส่งในรูปแบบของตั๋วอิเล็กทรอนิกส์ ซึ่งระบบในปัจจุบันจะใช้การ์ดทำงานเพียงหน้าที่เดียว

ความต้องการในการเพิ่มหน้าที่ของตั๋วอิเล็กทรอนิกส์มีมากขึ้น ด้วยเหตุผลนี้ในอนาคตจะมีการใช้งานการ์ดในหลากหลายหน้าที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 โครงสร้างทั่วไปที่มีการรวมกันของการ์ดบันทึกข้อมูลแบบไม่มีหน้าสัมผัสและการ์ดประมวลผลแบบมีหน้าสัมผัสและส่วนเชื่อมต่อของทั้งสองแบบ

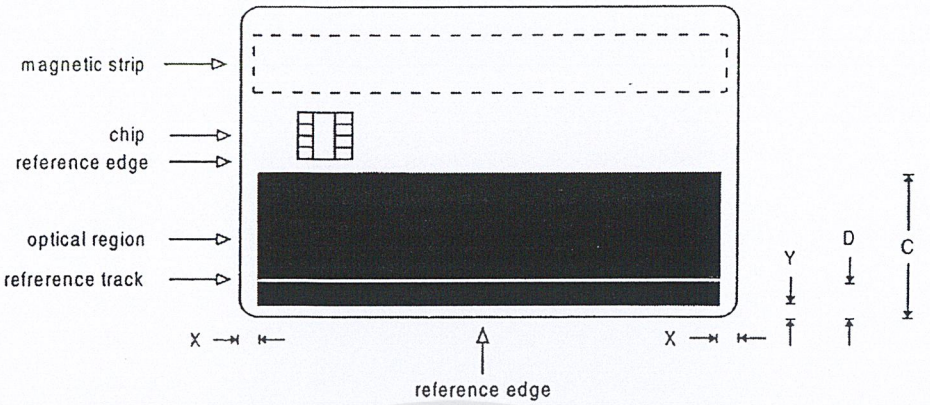
2.1.4 Optical Memory Card

มันน่าจะเป็นไปได้ที่จะสามารถเก็บข้อมูลจำนวนมากหลาย ๆ ครั้ง ลงบนสมาร์ตการ์ด ซึ่งเทคโนโลยีในปัจจุบันสามารถทำการเขียนได้เพียงครั้งเดียวและยังไม่สามารถลบได้ มาตรฐาน IS/IEC 11 693 และ 11 694 กำหนดคุณลักษณะทางฟิสิกส์ของออฟติคอล เมมโมรี่การ์ด และเทคนิคการบันทึกข้อมูลแบบลิเนียร์

การรวมความจุการบันทึกข้อมูลไว้สูง ๆ เป็นความคิดที่ฉลาดของสมาร์ตการ์ด ตัวอย่างเช่น ข้อมูลสามารถเขียนจากออฟติคอลเมมโมรี่ ส่วนกุญแจก็จะถูกเก็บอย่างปลอดภัยในชิปหน่วยความจำที่เป็นความลับ รูปที่ 2.11 แสดงให้เห็นถึงชนิดของหน้าสัมผัสที่สามารถมองเห็นได้ของสมาร์ตการ์ดและแถบแม่เหล็ก

Optical card ถูกใช้เป็นตัวอย่างทางการแพทย์ในการบันทึกข้อมูลของคนไข้ ต้องใช้หน่วยความจำที่มีขนาดใหญ่ เพื่อจะเก็บภาพเอ็กซ์เรย์ไว้ในการ์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.11 แสดงตำแหน่งของพื้นที่ที่เก็บข้อมูลของบัตร

2.2 รูปแบบของบัตร

สมาร์ตการ์ด โดยทั่วไปแล้วมีขนาด 85.6x54 มิลลิเมตร ซึ่ง ID-1 ได้ถูกออกแบบมาให้มีรูปแบบคล้ายกับรูปแบบนี้มากที่สุด โดยขนาดของบัตรถูกกำหนดขึ้นตามมาตรฐานสากล ISO 7810 มาตรฐานนี้มีขึ้นในปี 1985 ดังนั้นจึงไม่จำเป็นที่จะต้องทำตาม สมาร์ตการ์ดที่เรารู้จักในทุกวันนี้ จะมีอักษรย่อ "ID" ซึ่งก็คือ identification card มีลักษณะเป็นบัตรพลาสติกที่มีแถบแม่เหล็กนูน โดยออกแบบขึ้นมาสำหรับระบุข้อมูลเฉพาะบุคคล ซึ่งเวลานี้ยังไม่มีใครคิดถึงการฝังชิปลงไปในบัตร 2-3 ปีต่อมาจึงได้มีการแสดงชิปและตำแหน่งหน้าสัมผัสของบัตรซึ่งกำหนดขึ้นมาตามมาตรฐานอื่นอีก

ปัจจุบันนี้ บัตรมีความหลากหลายมากตามวัตถุประสงค์ที่จะนำไปใช้งาน และมีย่านขนาดของบัตรที่กว้าง จึงเป็นการยากที่จะบอกได้ว่า บัตรใบไหนเป็น ID-1 สมาร์ตการ์ดอย่างแท้จริง และยังมีความหนาของบัตรอีกที่เป็นตัวระบุถึงลักษณะที่ดีที่สุดของบัตร บัตรที่มีความหนา 0.76 มิลลิเมตร และบรรจุไมโครคอนโทรลเลอร์จะถูกพิจารณาเป็นสมาร์ตการ์ดตามมาตรฐาน ISO

รูปแบบ ID-1 มีข้อดีในการจัดการที่ง่าย รูปแบบของบัตรนี้ได้ถูกกำหนดขึ้นเพื่อว่าไม่ให้ใหญ่มากเกินไปที่จะเก็บบัตรไว้ในกระเป๋าเงิน แต่ต้องไม่เล็กจนกระทั่งเกิดการสูญหายได้ง่าย ยิ่งกว่านั้น ความยืด-หยุ่นของบัตรยังทำให้เกิดการยุ่งยากน้อยกว่าที่ควรจะเป็น

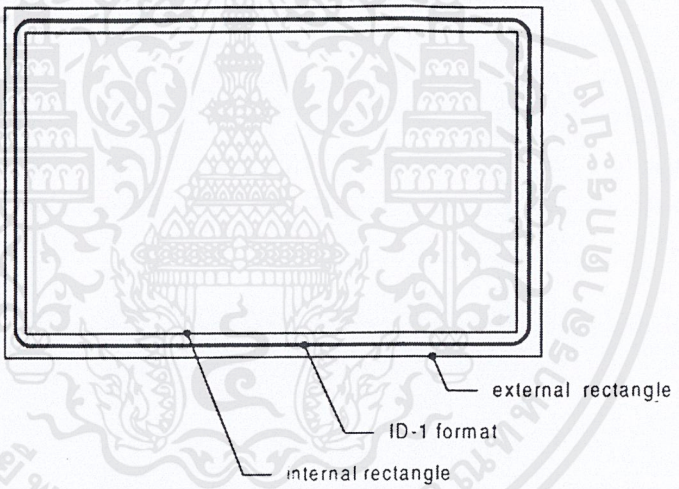
แต่รูปแบบนี้จะไม่พบความต้องการของผู้ซื้อ แสดงให้เห็นถึงความทันสมัยน้อย เซลโฟนบางรุ่นหนักเพียง 200 กรัม และไม่ใหญ่มากเกินกว่าแพ็คเก็ตที่บรรจุมัน ดังนั้นจึงจำเป็นที่จะต้องกำหนดรูปแบบเล็กกว่าเพื่อเพิ่มให้กับ ID-1 format ซึ่งควรจะนำเอาความสนใจของอุปกรณ์ปลายทางที่เล็กเข้าไปในการทำบัญชีด้วย บัตรชนิดนี้เล็กมากเนื่องจากถูกใส่เข้าไปในอุปกรณ์เพียงครั้งแรกเท่านั้นและยังคงอยู่ที่นั้นหากอยู่ในสภาพที่ดี จึงได้เกิด ID-000 format ขึ้น สำหรับเงื่อนไขดังกล่าวและมีชื่อเรียกว่า plug-in card ปัจจุบัน ID-000 ถูกใช้กับระบบโทรศัพท์เคลื่อนที่ GSM เท่านั้น ซึ่งมีที่เก็บบัตรเล็กมาก ๆ และไม่ต้องการบัตรที่ต้องมีการเปลี่ยนบ่อย ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อย่างไรก็ตาม ความจริงแล้ว บัตรที่มีรูปแบบ ID-000 มีความลำบากที่จะจัดการ ทั้งในการผลิตและโดยผู้ใช้ ทำให้มีการพัฒนารูปแบบเพิ่มเติมขึ้น นั่นคือ ID-00 หรือมินิการ์ดขนาดของมันอยู่ประมาณกึ่งกลางระหว่าง ID-1 และ ID-000 บัตรชนิดนี้สะดวกที่จะจัดการและถูกกว่าที่จะผลิตอีกด้วย เนื่องจากสำหรับตัวอย่างแล้วง่ายกว่าที่จะวางบัตรลงไป อย่างไรก็ตาม การกำหนดรูปแบบ ID-00 ค่อนข้างใหม่อยู่และยังไม่ได้ก่อตั้งขึ้นอย่างเป็นทางการแห่งประเทศและแห่งนานาชาติ

รูปแบบได้กำหนดขึ้นในมาตรฐานที่สัมพันธ์กัน ในวิธีที่ง่ายต่อการวัดขนาดของบัตร ดังนั้นส่วนสูงและน้ำหนักของ ID-1 card ต้องเป็นค่าที่เหมาะสม ระหว่างสี่เหลี่ยมซ้อนกัน 2 รูป (ไม่คิดมุมที่โค้ง) ดังรูป 2.12 โดยมีขนาดตามนี้

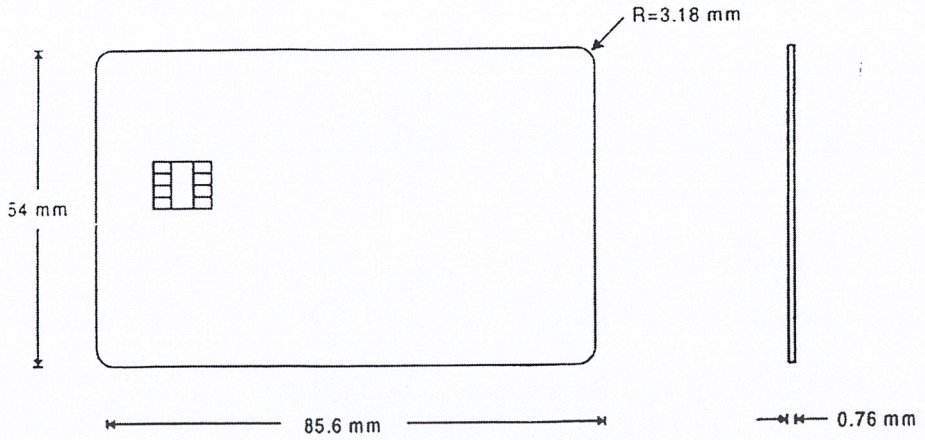
สี่เหลี่ยมรูปนอก	ความกว้าง	85.72 มิลลิเมตร (3.375 นิ้ว)
	ความสูง	54.03 มิลลิเมตร (2.127 นิ้ว)
สี่เหลี่ยมรูปใน	ความกว้าง	85.46 มิลลิเมตร (3.365 นิ้ว)
	ความสูง	53.92 มิลลิเมตร (2.123 นิ้ว)



รูปที่ 2.12 ขนาดของบัตร รูปแบบ ID-1

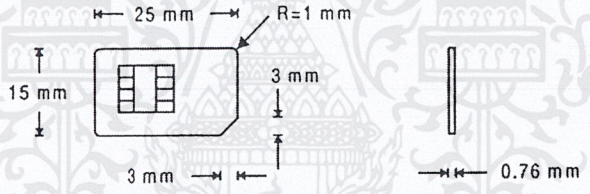
ความหนาต้องเท่ากับ 0.76 มิลลิเมตร (0.03 นิ้ว) โดยมีความผิดพลาด ±0.08 มิลลิเมตร (±0.003 นิ้ว) รัศมีของมุมและความหนาของบัตร กำหนดขึ้นอย่างง่าย จากการกำหนดดังกล่าว ขนาดของบัตร ID-1 สามารถแสดงได้ดังรูปที่ 2.13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 แสดงความกว้าง ยาว หนา และรัศมีมุมของ ID-1

รูปแบบ ID-000 ถูกกำหนดโดยใช้สี่เหลี่ยม 2 รูป ที่เหมือนกัน เนื่องจากรูปแบบนี้กำเนิดขึ้นในยุโรป (ตามระบบโทรศัพท์เคลื่อนที่ GSM) ขนาดพื้นฐานเป็นเมตร มุมล่างขวามือของปลั๊กอินการ์ดถูกหักออกจากมุม 45 องศา ดังรูป 2.14 เพื่อให้สะดวกในการใส่บัตรให้ถูกต้อง ลงไปในตัวอ่านบัตร



รูปที่ 2.14 แสดงความกว้าง ยาว หนา และรัศมีมุมของ ID-000

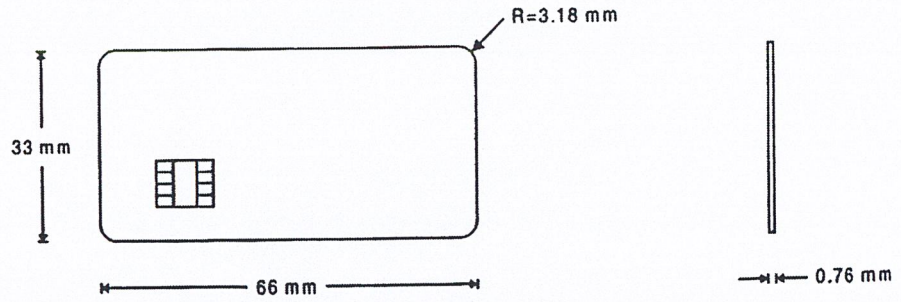
มิติของสี่เหลี่ยม 2 รูป สำหรับรูปแบบ ID-000 เป็น

สี่เหลี่ยมรูปนอก	ความกว้าง	25.10 มิลลิเมตร
	ความสูง	15.10 มิลลิเมตร
สี่เหลี่ยมรูปใน	ความกว้าง	24.90 มิลลิเมตร
	ความสูง	14.90 มิลลิเมตร

รูปแบบ ID-00 มีหน่วยการจัดเป็นเมตร ค่าสูงสุดและต่ำสุดของมิติของมันถูกกำหนดโดยสี่-เหลี่ยมที่ซ้อนกัน 2 รูป ที่มีขนาดดังนี้

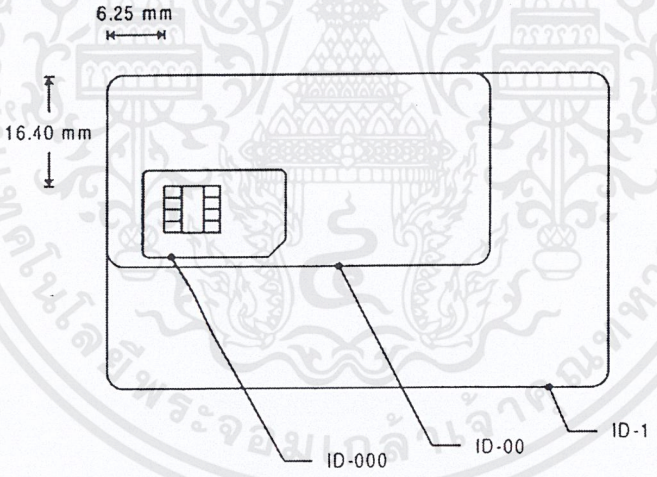
สี่เหลี่ยมภายนอก	ความกว้าง	66.10 มิลลิเมตร
	ความสูง	33.10 มิลลิเมตร
สี่เหลี่ยมภายใน	ความกว้าง	65.90 มิลลิเมตร
	ความสูง	32.90 มิลลิเมตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.15 แสดงความกว้าง ยาว หนาและรัศมีมุมของ ID-00

ขนาดที่สัมพันธ์กันระหว่าง ID-1, ID-00 และ ID-000 แสดงในรูป 2.16 บัตรที่มีรูปแบบที่เล็กกว่าสามารถถูกผลิตขึ้นจากรูปแบบที่ใหญ่กว่า โดยการปั๊มรูปแบบที่เล็กกว่าลงไปในตัวของบัตรที่รูปแบบใหญ่กว่า ซึ่งมีความสำคัญอย่างยิ่งสำหรับผู้ผลิตบัตร เนื่องจากจะทำให้ขบวนการผลิตนั้นเหมาะสมและผลิตขึ้นอย่างประหยัดมากยิ่งขึ้น



รูปที่ 2.16 ขนาดที่สัมพันธ์กันระหว่าง ID-1, ID-00 และ ID-000

ตัวอย่าง สามารถจินตนาการสำหรับผู้ผลิตที่จะผลิตบัตรเปล่าเพียงรูปแบบเดียว (นิยม ID-1) ได้ ซึ่งบัตรถูกฝังโมดูลต่าง ๆ ลงไป และระบุข้อมูลเฉพาะบุคคลลงในบัตรอย่างเต็มที่

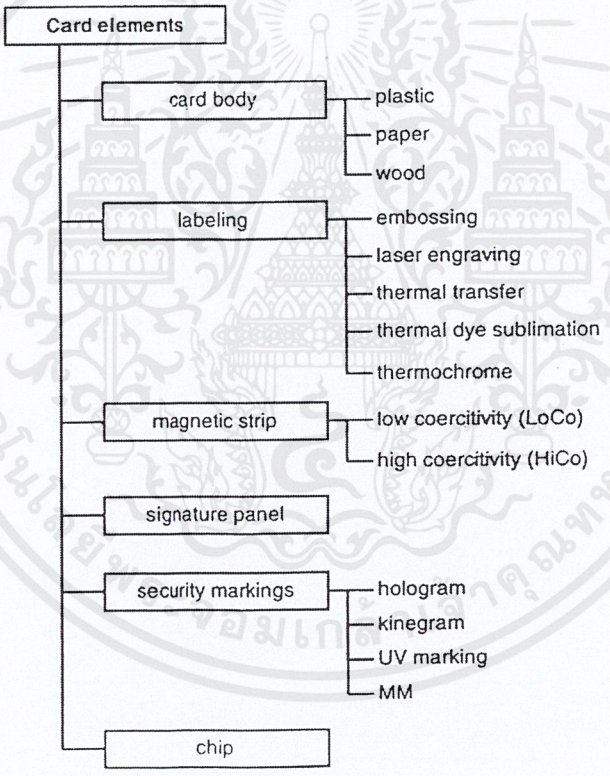
การขึ้นอยู่กับการประยุกต์ใช้งานเฉพาะของบัตรที่ผลิตด้วยบัตร สามารถถูกปรับให้เข้ากับรูปแบบที่ต้องการในขั้นตอนการผลิตขั้นต่อไปได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 ส่วนของตัวบัตร

วัสดุ การสร้าง และกระบวนการผลิตของตัวบัตรต้องพิจารณาที่หน้าที่ของส่วนต่าง ๆของบัตร รวมทั้งความเค้น ที่จะเกิดขึ้นระหว่างการใช้งาน ส่วนต่าง ๆ โดยทั่วไปมีดังนี้

- แถบแม่เหล็ก
- แถบลายเซ็นต์
- รอยนูนบนบัตร
- ภาพสามมิติ
- รอยประทับของข้อมูลส่วนตัว เช่น รูปภาพ ลายนิ้วมือ
- ชิปที่มีหน้าสัมผัส
- ส่วนรักษาความปลอดภัย



รูปที่ 2.17 ไดอะแกรมการแบ่งประเภทสำหรับส่วนต่าง ๆ ของบัตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความต้องการต่ำสุดที่สัมพันธ์กับความแข็งแรงของบัตร เป็นไปตามมาตรฐาน ISO 7810, 7813 และ 7816 ส่วนที่ 1

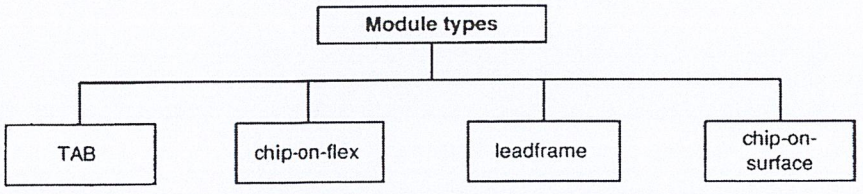
- การแผ่รังสีอัลตราไวโอเลต
- การแผ่รังสีเอ็กซเรย์
- รายละเอียดส่วนผิวหน้าของบัตร
- ความแข็งแรงทางกลของบัตรและหน้าสัมผัส
- ความไวต่อสนามแม่เหล็กไฟฟ้า
- การคายประจุของไฟฟ้าสถิตย์
- ความต้านทานทางอุณหภูมิ

มาตรฐาน ISO/IEC 10373 ระบุรายละเอียดวิธีการทดสอบของความต้องการดังกล่าว ซึ่งอนุญาตให้ผู้ใช้และผู้ผลิตการ์ด เพื่อทดสอบคุณสมบัติของบัตร สำหรับสมาร์ตการ์ดแล้ว การโค้งและการบิดงอ เป็นสิ่งที่สำคัญ เพราะว่าเป็นชิปที่เปราะและแตกหักง่ายเช่นเดียวกันกับแก้ว ลักษณะโครงสร้างพิเศษของชิปมีไว้เพื่อป้องกันชิปจากการบิดงอ การโค้งของบัตร

2.4 ชิปโมดูล

ส่วนประกอบที่สำคัญที่สุดของสมาร์ตการ์ดก็คือ ชิป เนื่องจากเป็นส่วนประกอบที่เปราะบางแตกหักง่าย จึงไม่สามารถนำมาวางบนพื้นผิวของบัตรได้ง่ายเหมือนกับบัตรแถบแม่เหล็ก จึงจำเป็นต้องมีตัวห่อหุ้มเพื่อป้องกันชิปจากความยุ่งยากในการใช้งาน ส่วนที่ห่อหุ้มนี้จะเรียกว่า ชิปโมดูล นอกจากนี้แล้ว สมาร์ตการ์ดยังมีหน้าสัมผัสทั้งชนิด 6 และ 8 ขาที่ต่อออกมาจากตัวชิปซึ่งใช้จ่ายไฟให้แก่ชิปและใช้สื่อสารข้อมูลกับภายนอก การแบ่งพื้นผิวของโมดูลก็เพื่อต่อสัญญาณไฟฟ้าเหล่านี้กับภายนอกชิป โดยที่ชิปโมดูลควรที่จะมีราคาไม่แพงเท่าที่คิด

ความหลากหลายในการออกแบบโมดูลได้คิดขึ้น เนื่องจากการเริ่มต้นของการพัฒนาบัตรสมาร์ตการ์ดก็เพื่อให้ได้ความต้องการทางด้านเทคนิค 2 อย่าง ดังกล่าว คือ การป้องกันการแตกหักของอุปกรณ์เซมิคอนดัคเตอร์ภายในชิป และการกำหนดพื้นผิวหน้าสัมผัส ซึ่งความสัมพันธ์ที่มากที่สุดดังกล่าวแสดงดังรูปที่ 2.18



รูปที่ 2.18 การแบ่งชนิดต่าง ๆ ของชิปโมดูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

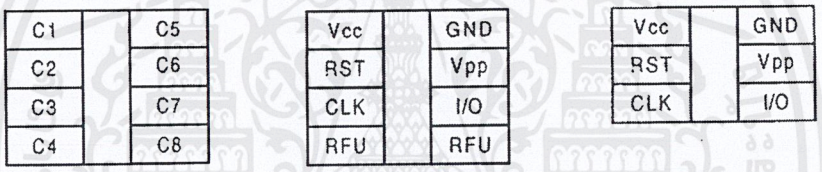
สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

2.5 คุณสมบัติทางไฟฟ้าของสมาร์ตการ์ด

คุณสมบัติทางไฟฟ้าของสมาร์ตการ์ดขึ้นอยู่กับไมโครคอนโทรลเลอร์ที่อยู่ภายในชิปเท่านั้นเนื่องจากมันเป็นอุปกรณ์เพียงชิ้นเดียวของบัตรที่เป็นวงจรทางไฟฟ้า เทคโนโลยีเริ่มแรกของสมาร์ตการ์ดนั้นจะมีปัจจัยหลักอยู่ในส่วนของไมโครคอนโทรลเลอร์ ทั้งหน้าที่และฟังก์ชันการทำงาน

2.5.1 การติดต่อกับบัตร

สมาร์ตการ์ดส่วนใหญ่จะมีหน้าสัมผัส 8 ขา อยู่ด้านหน้าบัตร รูปแบบดังกล่าวนี้ จะใช้ในการอินเตอร์เฟสทางสัญญาณไฟฟ้าระหว่างปลายสายกับไมโครคอนโทรลเลอร์ที่อยู่ในบัตร สัญญาณไฟฟ้าทั้งหมดก็จะผ่านหน้าสัมผัสเหล่านี้ อย่างไรก็ตาม ตามมาตรฐาน ISO/IEC 7816-2 นั้น หน้าสัมผัส C4 และ C8 ถูกสำรองไว้เพื่อใช้ในอนาคต ดังนั้นเราจึงจะยังไม่ใช้ 2 ขานี้ หนึ่งในสองขานี้จะใช้สำหรับอินเตอร์เฟสกับอินพุท/เอาท์พุทชุดที่ 2 เพื่อจะสามารถรองรับการสื่อสารข้อมูลแบบฟูลดูเพล็กซ์ได้ เนื่องจากว่าปัจจุบัน 2 ขานี้ยังไม่ถูกใช้งาน โมดูลของสมาร์ตการ์ดจึงมีเพียง 6 ขาเท่านั้น ซึ่งทำให้สามารถลดต้นทุนการผลิตได้บ้างเล็กน้อย อย่างไรก็ตาม หน้าที่การใช้งานของขาทั้ง 6 ขาก็ยังคงใช้งานเหมือนกับ 8 ขา



รูปที่ 2.19 การกำหนดหน้าที่ทางไฟฟ้า และหมายเลขของหน้าสัมผัสของสมาร์ตการ์ดตามมาตรฐาน ISO 7816-2

หมายเลขของหน้าสัมผัสจะเรียงจากบนซ้ายลงสู่ด้านล่างทางขวามือ ซึ่งถูกออกแบบและระบุคุณสมบัติทางไฟฟ้าดังรูป 2.19 ตามมาตรฐาน ISO

2-3 ปี ที่ผ่านมายังจำเป็นที่จะต้องใช้แรงดันจากภายนอกเพื่อทำการโปรแกรมและลบข้อมูลใน EEPROM อยู่ เนื่องจากว่าไมโครคอนโทรลเลอร์ที่ใช้นั้นไม่มีชาร์จปั๊มนั่นเอง หน้าสัมผัส C6 ถูกใช้ในหน้าที่นี้ อย่างไรก็ตาม เทคโนโลยีในปัจจุบันนี้ แรงดันในการโปรแกรมและลบข้อมูลจะถูกสร้างขึ้นโดยตรงภายในชิปโดยการใช้ชาร์จปั๊ม ดังนั้นหน้าสัมผัสนี้จึงไม่ถูกใช้งานอีกต่อไป แต่กระนั้นขานี้ก็ไม่สามารถใช้ในหน้าที่อื่นได้ เพราะเป็นการขัดแย้งกับมาตรฐาน ISO ดังนั้น สมาร์ตการ์ดทุก ๆ ใบ ก็จะมีหน้าสัมผัสหนึ่งหน้าสัมผัสที่ไม่ได้ใช้งาน แต่ก็ยังคงดำรงไว้อยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
46572

Contact	Designation	Function
C1	Vcc	Supply voltage
C2	RST	Reset input
C3	CLK	Clock input
C4	RFU	Reserved for future applications, presently not used
C5	GND	Ground (earth)
C6	Vpp	Programming voltage (generally no longer used)
C7	I/C	Serial communications input/output
C8	RFU	Reserved for future applications, presently not used

ตารางที่ 2.2 การออกแบบและหน้าที่ของหน้าสัมผัสตามมาตรฐาน ISO 7816-2

2.5.2 แรงดันที่จ่ายให้กับบัตร

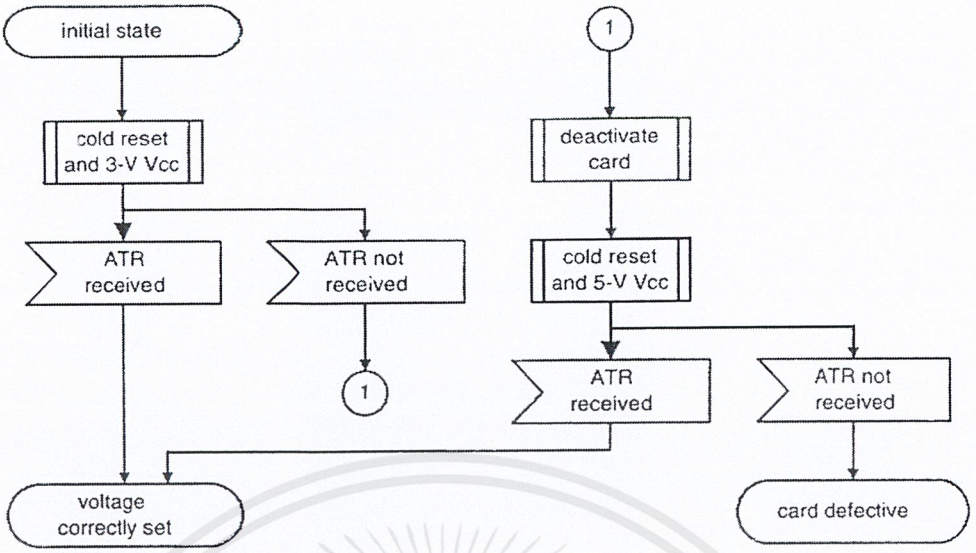
ไฟเลี้ยงของบัตรคือ 5 โวลต์ โดยมีความผิดพลาดได้สูงสุด ±10 % ซึ่งแรงดันค่านี้มีกฎตาม TTL ซึ่งเป็นมาตรฐานของการ์ดที่ใช้ตามท้องตลาดและใช้กันอยู่ในปัจจุบัน

สำหรับโทรศัพท์เคลื่อนที่แล้ว การลดน้ำหนักอุปกรณ์ในการตอบสนอง ความกดดันทางการตลาด จึงทำให้ต้องเปลี่ยนแรงดันแบตเตอรี่จาก 6 โวลต์หรือ 4.5 โวลต์ไปเป็น 3 โวลต์ เนื่องจากว่าส่วนประกอบของโทรศัพท์เคลื่อนที่ทั้งหมดนั้น ปัจจุบันนี้สามารถใช้ได้กับแรงดัน 3 โวลต์ แรงดันสมาร์ตการ์ดจึงเป็นอุปกรณ์เพียงตัวเดียวที่ยังคงใช้ไฟเลี้ยง 5 โวลต์อยู่ ด้วยเหตุนี้ โทรศัพท์เคลื่อนที่จึงต้องมีโวลต์เตจคอนเวอร์เตอร์ในการจัดการกับไฟเลี้ยงให้แก่บัตร ซึ่งเป็นการเพิ่มความซับซ้อนยิ่งขึ้น และดังนั้นไม่จำเป็นที่จะต้องเพิ่มราคาขึ้น ในการพัฒนาต่อไปในอนาคต สมาร์ตการ์ดจะมีย่านแรงดันอยู่ระหว่าง 3-5 โวลต์ ±10 % ซึ่งเป็นผลให้ย่านแรงดันที่ใช้งานจริงอยู่ระหว่าง 2.7-5.5 โวลต์

ในทางทฤษฎีแล้วเป็นไปได้ที่จะพัฒนาไปเป็นบัตรประเภทไมโครคอนโทรลเลอร์ ที่ใช้แรงดันเป็น 3 โวลต์ แต่มีข้อเสียตรงที่ว่าไม่สามารถใช้ได้กับบัตรเก่าที่ใช้ไฟ 5 โวลต์ เพราะเมื่อนำบัตรที่ต้องการไฟ 3 โวลต์เป็นไฟเลี้ยงไปใช้กับเครื่องอ่านที่จ่ายไฟ 5 โวลต์ จะทำให้ชิปภายในบัตร 3 โวลต์ถูกทำลายได้ ด้วยเหตุนี้ทำให้สูญเสียความยืดหยุ่นของบัตรไป เพราะผู้ใช้ต้องใช้บัตรให้ตรงตามประเภทแรงดันที่ใช้เท่านั้น

ตามกฎแล้ว การขยายย่านของแรงดันดังกล่าวไม่เป็นปัญหาสำหรับตัวประมวลผล หรือส่วนประกอบทางหน่วยความจำ แต่ในการสร้าง EEPROM เข้าไปในไมโครคอนโทรลเลอร์ ที่มีตัวปั๊มแรงดันภายในนั้นเกิดอุปสรรคอย่างมาก สำหรับบัตรที่ใช้ไฟ 3 โวลต์ แต่เมื่อเร็ว ๆ นี้สามารถสร้างได้แล้ว โดยชิปนั้นสามารถทำงานได้ที่แหล่งจ่ายแรงดันอยู่ที่ 2.7-5.5 โวลต์ ซึ่งย่านแรงดันในการทำงานเหล่านี้กำลังจะกลายเป็นเกณฑ์บังคับสำหรับไมโครคอนโทรลเลอร์ทุกตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.20 ไดอะแกรมในการทำงาน ของขั้นตอนในการเลือกแหล่งจ่ายแรงดันให้แก่บัตร ในรูปนี้เป็นการเลือกแรงดันที่จำเป็นที่ต่ำที่สุด

2.5.3 แหล่งจ่ายกระแส

ไมโครคอนโทรลเลอร์ของบัตรจะได้รับแหล่งจ่ายแรงดัน ผ่านทางหน้าสัมผัส C1 ตามข้อกำหนดของ GSM 11.11 นั้น กระแสอาจไม่มากเกินกว่า 10 มิลลิแอมป์ ถึงแม้ว่ามาตรฐาน ISO ระบุค่ากระแสไว้ที่ 200 มิลลิแอมป์ ซึ่งเป็นค่าเก่า และจะเปลี่ยนค่าไปตามเทคโนโลยีสมัยใหม่ต่อไป

สำหรับค่าปลดปล่อยพลังงาน (power dissipation) ของสมาร์ตการ์ด มีค่า 50 มิลลิวัตต์ เมื่อแหล่งจ่ายแรงดัน 5 โวลต์ และการกินกระแส 10 มิลลิแอมป์ เนื่องจากมีค่าต่ำ จึงไม่มีความจำเป็นจะต้องมีการกระตุ้น (possible self-warming) ชิปในการใช้งาน ถึงแม้ว่าค่ากำลังงานค่านี้ได้ปลดปล่อยออกทางพื้นที่ผิว 25 ตารางมิลลิเมตร

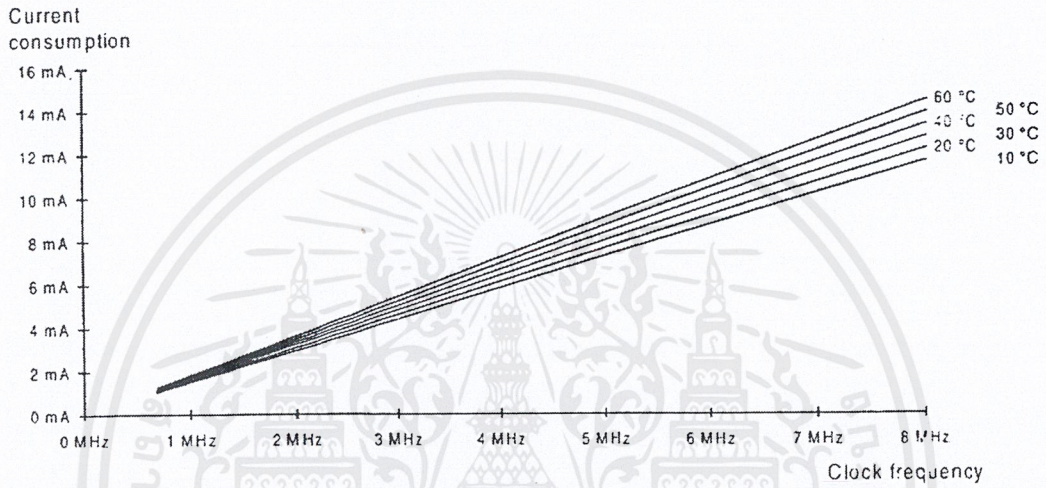
สำหรับการกินกระแสของไมโครคอนโทรลเลอร์นั้นเป็นสัดส่วนโดยตรงกับความถี่ของสัญญาณนาฬิกาที่ใช้ ซึ่งหมายความว่าการกินกระแสสูงสุดหรือกระแสเป็นไปตามหน้าที่การทำงานของความถี่สัญญาณนาฬิกา นอกจากนี้กระแสยังขึ้นอยู่กับอุณหภูมิด้วยแต่ไม่มีผลกระทบมากนัก

การพิจารณา รายละเอียดที่สำคัญอีกอย่างซึ่งแหล่งจ่ายกระแสได้ทำให้เกิดปัญหาสำหรับผู้ผลิตเครื่องอ่านบัตรหลายรายจึงละเลยมันไป ไมโครคอนโทรลเลอร์ทุก ๆ ตัว ในการใช้งานนั้นล้วนเป็นเทคโนโลยีทางด้าน CMOS ภายใต้สภาวะที่แน่นอนนั้น กระแสในการลัดวงจร (short-circuit) จะสูงซึ่งสามารถเกิดขึ้นในเวลาอันสั้น ในระหว่างขบวนการสวิตซ์ของทรานซิสเตอร์เหล่านี้ จะสร้างกระแส (current spikes) ที่สูงกว่ากระแสในการทำงานปกติหลายเท่า ด้วยคาบเวลาในย่านนาโนวินาที (nanosecond range) กระแสเหล่านี้สามารถเกิดขึ้นเมื่อตัวปั๊มแรงดันที่ EEPROM ทำงาน สมมติถ้าเครื่องอ่านบัตรไม่สามารถจ่ายกระแสสูงเหล่านี้ได้ในระหว่างช่วงสั้น ๆ เหล่านี้ แหล่งจ่ายแรงดันก็จะตกลงต่ำกว่าค่าที่ยอมรับได้ ทำให้เกิดความผิดพลาดในการเขียนข้อมูลใน EEPROM หรือ สับไก (trigger) ตัวตรวจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จับระดับแรงดันต่ำในชิป

สำหรับเหตุผลเหล่านี้ การอ้างอิงกับกระแสเหล่านี้ เดี๋ยวนี้สามารถพบได้ในมาตรฐานทุก ๆ มาตรฐานและข้อกำหนดในทางปฏิบัติ แหล่งจ่ายกระแสถูกต้องการให้สามารถควบคุมกระแสด้วยคาบเวลาสูงสุด 400 นาโนวินาที และแอมปลิจูดสูงสุด 200 มิลลิแอมป์ ซึ่งการสมมติกระแสให้เป็นรูปสามเหลี่ยมนั้นตรงกับการสะสมประจุ 40 นาโนแอมป์ต่อวินาที ปัญหานี้สามารถแก้ไขได้โดยการต่อตัวเก็บประจุชนิดเซรามิกค่า 100 นาโนฟารัด ระหว่างกราวด์กับแหล่งจ่ายแรงดันโดยให้ใกล้กับ หน้าสัมผัสของบัตรมาก ๆ



รูปที่ 2.21 แสดงการกินกระแสของไมโครคอนโทรลเลอร์ ต่อความถี่สัญญาณนาฬิกาและอุณหภูมิ

ในอนาคต ข้อกำหนดของ GSM ได้คาดหวังที่จะลดการกินกระแสสูงสุดที่สามารถยอมรับได้เป็น 1 มิลลิแอมป์ เพื่อเพิ่มอายุการใช้งานของแบตเตอรี่ที่ทำงานในโทรศัพท์เซลลูล่าร์ (cellular telephone) ให้มากเท่าที่จะทำได้ ด้วยการก้าวหน้าอย่างรวดเร็วทางด้านเทคโนโลยีสารกึ่งตัวนำ การลดการกินกระแสสูงสุดให้เหลือ 1 มิลลิแอมป์ จึงทำให้มีความยุ่งยากน้อย จึงมีเพียงกระแสที่ใช้ในการเขียนข้อมูลลงใน EEPROM เท่านั้นที่ยังเป็นปัญหาอยู่ แต่ด้วยการเพิ่มเซลล์ใน EEPROM และการลดผลที่เกิดขึ้นในการสะสมประจุในเซลล์ ก็จะได้ค่ากระแส 1 มิลลิแอมป์ ภายในอีก 2-3 ปีนี้

ไมโครคอนโทรลเลอร์ทุก ๆ ตัวส่วนใหญ่จะมีโหมดพิเศษในการประหยัดพลังงาน (power-saving mode) ซึ่งในทุก ๆ ส่วนของชิป ยกเว้นส่วนของการขัดจังหวะอินพุท/เอาท์พุท จะไม่ทำงาน การกินกระแสของไมโครคอนโทรลเลอร์ตกลงอย่างคงที่ เมื่ออยู่ในสภาวะนี้ ซึ่งเรียกว่าโหมดหลับ เนื่องจากส่วนต่างๆ ส่วนใหญ่ของชิปไม่ได้ถูกต่อจากแหล่งจ่ายแรงดัน ในทางทฤษฎีแล้วเพียงส่วนตรรกะในการขัดจังหวะของส่วนเชื่อมต่ออินพุท/เอาท์พุทและแรมเท่านั้นที่จำเป็นต้องได้รับกำลังงาน และส่วนประกอบอื่นที่เหลือไม่สามารถทำงานได้ ข้อมูลต่างๆ ใน RAM ต้องได้รับการรักษาในโหมดการทำงานนี้เพื่อเก็บค่าพารามิเตอร์สถานะการทำงานค่าปัจจุบัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในทางปฏิบัติแล้วตัวประมวลผลได้รับกำลังงานเสมอด้วยแต่ ROM และ EEPROM นั้นไม่ถูกทำงานโหมดการทำงานนี้เป็นโหมดที่น่าสนใจเป็นพิเศษสำหรับแบตเตอรี่ที่โทรศัพท์เซลลูล่าร์ใช้อยู่ดังนั้น GSM 11.11 จึงกำหนดกระแสสูงสุดที่ยอมรับได้ในโหมดหลับ เป็น 200 ไมโครแอมป์ด้วย ความถี่สัญญาณนาฬิกา 1 เมกกะเฮิรตซ์ ที่อุณหภูมิ 25 องศาเซลเซียส ส่วนการจะเกิดการกินกระแสสูงสุดเป็น 10 ไมโครแอมป์นั้นได้ถูกวางแผนไว้สำหรับอนาคต

2.5.4 สัญญาณนาฬิกาจากภายนอก

ตัวประมวลผลของสมาร์ตการ์ดนั้นไม่มีตัวกำเนิดสัญญาณนาฬิกาภายใน ดังนั้นจึงจำเป็นต้องมีสัญญาณนาฬิกาจากภายนอกจ่ายให้แก่บัตร ซึ่งสัญญาณนาฬิกานี้จะเป็นตัวอ้างอิงกับอัตราในการส่งถ่ายข้อมูล โดยค่าดิวตี้ไซเคิล (duty factor) ของสัญญาณนาฬิกาต้องอยู่ในย่าน 40–60 เปอร์เซ็นต์ ตามข้อกำหนดของ GSM 11.11

สัญญาณนาฬิกาที่จ่ายให้ แก่หน้าสัมผัสไม่จำเป็นต้องเหมือนกันกับสัญญาณนาฬิกาภายในที่ได้เตรียมให้แก่ตัวประมวลผล ไมโครคอนโทรลเลอร์บางชนิด จะมีตัวหารความถี่ที่แทร็กอยู่ระหว่างสัญญาณนาฬิกาภายนอกและสัญญาณนาฬิกาภายใน ซึ่งมีค่าตัวหารเป็น 2 เสมอ เพื่อที่ว่าสัญญาณนาฬิกาจากภายนอกมีความเร็วเป็น 2 เท่าของสัญญาณนาฬิกาภายใน ซึ่งเหล่านี้เป็นตัวกำหนดคุณลักษณะของส่วนฮาร์ดแวร์ภายในและเป็นส่วนที่อนุญาติให้ตัวกำเนิดสัญญาณที่แสดงในปลายสายแล้วนั้นให้ถูกใช้งานเป็นแหล่งจ่ายสัญญาณนาฬิกาสำหรับชิป

ไมโครคอนโทรลเลอร์ส่วนใหญ่จะอนุญาติให้สัญญาณนาฬิกาอยู่ในสภาวะเปิดวงจร (OFF) เมื่อซีพียูอยู่ในโหมดหลับ ในกรณีนี้การเปลี่ยนสถานะเป็นเปิดวงจร นั้นหมายความว่าการคงสภาวะสัญญาณนาฬิกาไว้ที่ระดับที่กำหนดค่าหนึ่ง ซึ่งขึ้นอยู่กับข้อกำหนดทางผู้ผลิตสารกึ่งตัวนำ ซึ่งสภาวะ “OFF” นั้นอาจเป็นสภาวะสูง (High) หรือไม่ก็เป็นสภาวะต่ำ (Low)

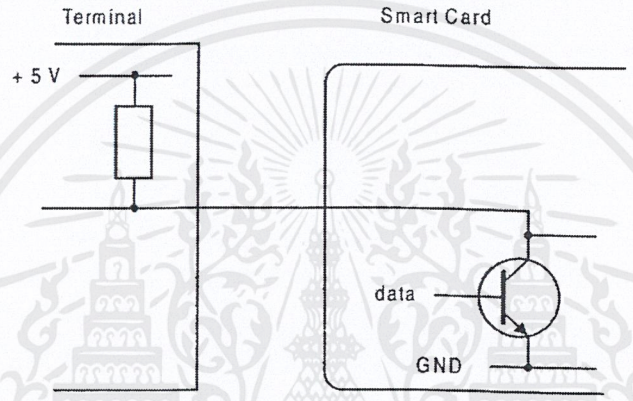
สมาร์ตการ์ดจะมีค่ากระแสเพียง 2-3 ไมโครแอมป์เท่านั้นจากขาสัญญาณนาฬิกา ดังนั้นการเปลี่ยนสภาวะเป็นปิด สัญญาณนาฬิกาในครั้งแรกอาจผิดเพี้ยนไปเล็กน้อยจากปกติได้บ้าง แต่ปริมาณของกำลังงานที่ได้เก็บไว้ภายในปลายสายนั้นก็เป็นสิ่งสำคัญ ดังนั้นจึงคุ้มค่าในการประยุกต์ใช้งานอย่างแน่นอน

2.5.5 การส่งถ่ายข้อมูล

สมมติว่ามีความผิดพลาดเกิดขึ้นระหว่างการส่งถ่ายข้อมูล ซึ่งปลายสายและบัตรได้พยายามส่งข้อมูลที่เวลาเดียวกัน ก็จะเกิดการชนกันของข้อมูลขึ้นบนสายสัญญาณอินพุท/เอาท์พุท ซึ่งเป็นส่วนที่เกิดปัญหาขึ้นในการประยุกต์ใช้งาน โดยผลของกระแสในสายสัญญาณอินพุท/เอาท์พุท อาจมากพอที่จะทำลายส่วนประกอบต่าง ๆ ของชุดเชื่อมต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อป้องกันความเสียหายให้แก่สารกึ่งตัวนำในกรณีเหล่านี้ สายสัญญาณอินพุท/เอาท์พุทที่ปลายสายต้องต่อตัวต้านทานดึงขึ้น (pull-up resistor) 20 กิโลโอห์ม กับแหล่งจ่ายแรงดัน +5 โวลต์ เพื่อหลีกเลี่ยงปัญหาของทั้ง 2 ฝ่าย จึงพยายามที่จะขับสายสัญญาณข้อมูลไปเป็น 2 ระดับที่แตกต่างกัน ในกรณีที่เกิดการผิดพลาดเกิดขึ้น สมมติถ้าสายสัญญาณอินพุท/เอาท์พุท จำเป็นที่จะต้องถูกตั้งค่าที่ระดับ +5 โวลต์ ระหว่างการติดต่อสื่อสาร ฝ่ายที่ส่งข้อมูลมาสัมพันธ์กับค่านี้จะเปลี่ยนเอาท์พุทของมันไปเป็นสถานะความต้านทานสูง (ระดับสภาวะที่ 3) และสายสัญญาณก็จะถูกยกขึ้นเป็น +5 โวลต์ โดยตัวต้านทานดึงขึ้นตัวเดียว

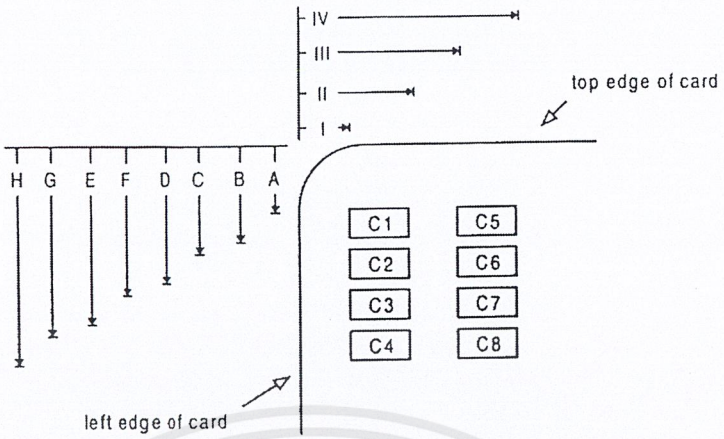


รูปที่ 2.22 วงจรของช่องทางสื่อสารอินพุท/เอาท์พุท ระหว่างปลายสายและสมาร์ตการ์ด

2.6 ชนิดของหน้าสัมผัสของสมาร์ตการ์ด

ความแตกต่างระหว่างสมาร์ตการ์ดกับบัตรชนิดอื่นก็คือ ไมโครคอนโทรลเลอร์ที่อยู่ในตัวของมัน หากหน้าสัมผัสถูกใช้สำหรับหน้าที่ในการจ่ายไฟ และส่งถ่ายข้อมูล ดังนั้นจึงต้องการติดต่อกันด้วยสัญญาณทางไฟฟ้า หน้าสัมผัสเหล่านี้ประกอบไปด้วยหน้าสัมผัสจำนวน 6 หรือ 8 ขา ที่เป็นแผ่นทอง สามารถพบได้บนสมาร์ตการ์ดตามมาตรฐานทุก ๆ ใบ ตำแหน่งของหน้าสัมผัสเหล่านี้ที่อยู่บนตัวบัตรและขนาดของมันได้ถูกระบุรายละเอียดไว้โดยมาตรฐาน ISO 7816-2 ในปี ค.ศ. 1988

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



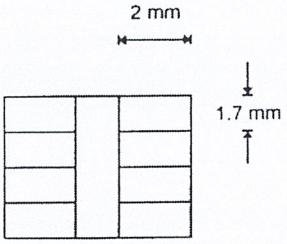
รูปที่ 2.23 ตำแหน่งของหน้าสัมผัสที่สัมพันธ์กับโครงร่างตัวของบัตร

I	10.25 mm maximum	A	19.23 mm minimum
II	12.25 mm minimum	B	20.93 mm minimum
III	17.87 mm maximum	C	21.77 mm maximum
IV	79.87 mm minimum	D	23.47 mm minimum
		E	24.31 mm maximum
		F	26.01 mm minimum
		G	26.85 mm maximum
		H	28.55 mm minimum

ในประเทศฝรั่งเศสนั้นมาตรฐานที่ใช้กันในประเทศ ได้ใช้กันมาก่อนมาตรฐาน ISO 7816-2 ซึ่งกำหนดโดย AFNOR มาตรฐานนี้ได้ระบุตำแหน่งของหน้าสัมผัสซึ่งอยู่สูงกว่ามาตรฐาน ISO เล็กน้อย ตำแหน่งเหล่านี้ถูกรวมไว้ในมาตรฐาน ISO ด้วยในฐานะที่เป็นตำแหน่งหน้าสัมผัสที่เกี่ยวกับหัวต่อ แต่มาตรฐานนั้นแนะนำว่าตำแหน่งเหล่านี้ไม่ถูกใช้งานในอนาคต แต่เนื่องจากยังมีการ์ดหลายใบในประเทศฝรั่งเศสที่ใช้ตำแหน่งเกี่ยวกับหัวต่อนี้อยู่ บัตรเหล่านี้จึงยังไม่หายไปอย่างรวดเร็ว

ตำแหน่งที่แท้จริงในส่วนของหน้าสัมผัสที่อยู่มุมด้านซ้ายบนของตัวบัตร ซึ่งแสดงได้ดังรูป 2.24 โดยขนาดของหน้าสัมผัสต่าง ๆ เท่ากับ 1.7 มิลลิเมตร x 2 มิลลิเมตร (ความสูง x ความกว้าง) ขนาดสูงสุดของหน้าสัมผัสต่าง ๆ ไม่ได้ระบุเอาไว้ แต่แน่นอนว่าหน้าสัมผัสแต่ละอันต้องถูกจำกัด โดยสัญญาณทางไฟฟ้าของแต่ละหน้าสัมผัสจะแยกออกจากกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

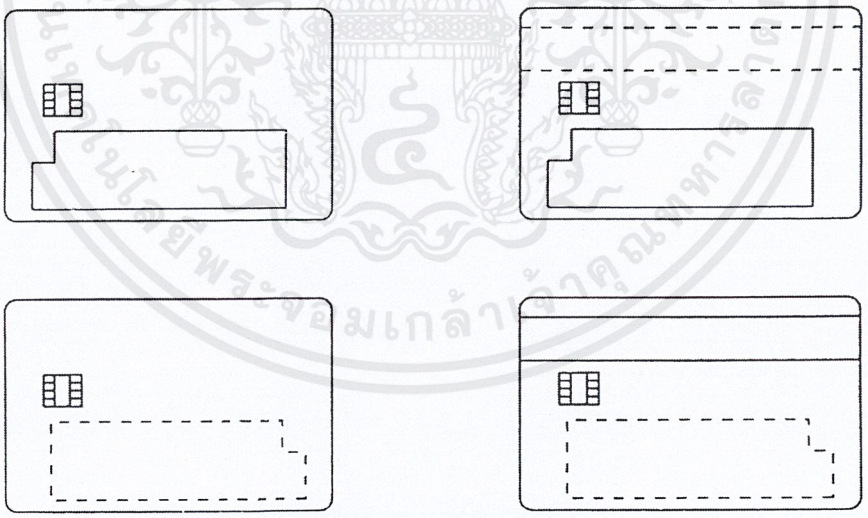


รูปที่ 2.24 ขนาดของหน้าสัมผัสต่ำสุดที่ระบุไว้ตาม ISO-7816-2

ตำแหน่งของ โมดูลภายในตัวบัตรได้ระบุไว้มาตรฐานแล้ว ส่วนตำแหน่งของพื้นที่แถบแม่เหล็กและพื้นที่ที่สำรองไว้สำหรับการตอกบัตรให้นูน จะถูกระบุรายละเอียดไว้อย่างแท้จริง (ดูมาตรฐาน ISO 7816) โดยทั้ง 3 ส่วนนี้อาจถูกแสดงไว้บนบัตรเพียงใบเดียว แต่ในกรณีนี้ได้แก่

(a) ถ้ามีเพียงส่วนของชิป และส่วนที่ตอกให้นูนขึ้น ทั้ง 2 ส่วนนี้ก็จะถูกวางตำแหน่งไว้บนด้านเดียวกันหรือบนด้านตรงข้ามของบัตร

(b) ถ้ามีส่วนของแถบแม่เหล็กอยู่ด้วย แถบแม่เหล็กและส่วนที่ถูกตอกให้นูนต้องถูกวางไว้บนด้านตรงข้ามกันของบัตร

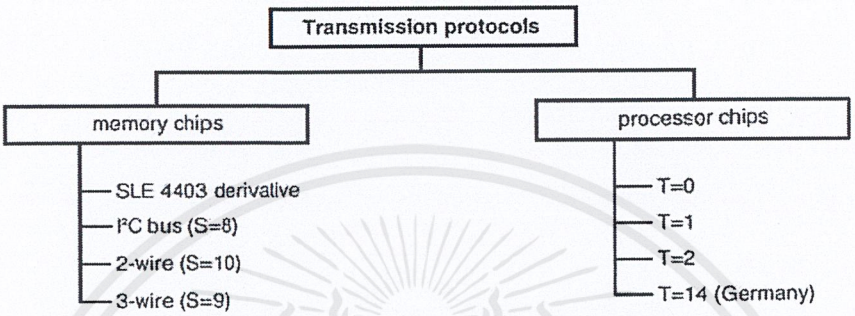


รูปที่ 2.25 รูปแบบการเรียงที่เป็นไปได้ระหว่างชิป ส่วนที่ถูกตอกให้นูนและแถบแม่เหล็กตามมาตรฐาน ISO 7816-2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 Data Transmission Protocols

หลังจากที่สมาร์ตการ์ดได้ส่ง ATR และ PTS อาจมีการแทนที่ลง มันจะรอรับคำสั่งแรกจากปลายทาง กระบวนการต่อมาจะสอดคล้องกับหลักของมาสเตอร์สลาฟเสมอ ซึ่งปลายทางก็คือมาสเตอร์และการ์ดเป็นสลาฟ ในความเป็นจริงปลายทางจะส่งคำสั่งไปให้การ์ด และปฏิบัติตามคำสั่งและส่งค่าตอบกลับไป (back and front) มีผลต่อคำสั่งและผลตอบสนองที่ไม่เคยเปลี่ยนแปลง



รูปที่ 2.26 รูปแบบของทรานสมิทชั่นโปรโตคอลที่บัตรสมาร์ตการ์ดใช้

นี่เป็นโครงสร้างของวิธีการต่าง ๆในการติดต่อสื่อสารของสมาร์ตการ์ด มันเป็นจำนวนตัวเลขของกระบวนการต่าง ๆ ของการติดต่อสื่อสารแบบรีซิงโครไนซ์ซิง ถ้าเกิดสิ่งรบกวน วิธีการที่แน่นอนของการร่วมกันของผลตอบสนอง และกระบวนการที่ใช้ในเหตุการณ์ของการส่งผิดพลาด จะอธิบายได้ในรูปแบบของทรานสมิทชั่นโปรโตคอล

ทรานสมิทชั่นโปรโตคอล แบบพื้นฐานมีอยู่ทั้งหมด 15 โปรโตคอล โดยเราจะใช้สัญลักษณ์ของโปรโตคอล ว่า "T =" (แทนทรานสมิทชั่นโปรโตคอล) บวกกับเลขลำดับของมัน โดยสรุปได้ในตาราง 2.3

Transmission protocol	Meaning
T=0	Asynchronous, half-duplex, byte oriented, covered by ISO/IEC 7816-3
T=1	Asynchronous, half-duplex, block oriented, covered by ISO/IEC 7816-3 and 1
T=2	Asynchronous, full- duplex, block oriented, covered by ISO/IEC 7816-4
T=3	Full-duplex, not yet covered
T=4	Asynchronous, half-duplex, byte oriented, extension of T=0, not yet covered
T=5 to T=13	Reserved for future functions, not yet covered
T=14	Reserved for future functions, no ISO standard
T=15	Reserved for future functions, not yet covered

ตารางที่ 2.3 ผลสรุปของทรานสมิทชั่นโปรโตคอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สองโปรโตคอลที่ใช้กันคือ T = 0 เป็นโปรโตคอลตัวแรก ซึ่งสร้างขึ้นโดยมาตรฐานสากลในปี 1989 (ISO/IEC 7816-3) อีกตัวคือ T = 1 ซึ่งเริ่มใช้ในปี 1992 เป็นส่วนประกอบหนึ่งของมาตรฐานสากล T = 2 เป็น โปรโตคอลในการส่งข้อมูลแบบสองทิศทาง ซึ่งพัฒนามาจาก T = 1 โดยจะเตรียมตัวและมีในมาตรฐานสากลในอีก 2-3 ปี

ในเยอรมันโปรโตคอลที่ 3 เรียกว่า T = 14 ซึ่งใช้ในโทรศัพท์แบบใช้บัตร มันเป็นการกำหนดเองของบริษัทเยอรมัน เทเลคอม (German Telecom)

กลุ่มของข้อมูลที่ถูกส่งโดยทรานสมิทชั่นโปรโตคอล เรียกว่า TPDUs (Transmission Protocol Data Units) มันเป็นวิธีการที่ขึ้นอยู่กับโปรโตคอลที่เก็บข้อมูลส่งมาจากการ์ด ในการใช้งานจริง ข้อมูลจะถูกเก็บไว้ในที่เก็บชนิดนี้

ในส่วนเพิ่มเติมเกี่ยวกับทรานสมิทชั่นโปรโตคอลของสมาร์ตการ์ด เป็นสิ่งเพิ่มเติมของซิงโครนัสโปรโตคอลพื้นฐานของเมมโมรีการ์ดโดยใช้ส่งไปในโทรศัพท์ใช้การ์ด บัตรประกันสุขภาพ อย่างไรก็ตาม มันไม่มีกลไกของการแก้ไขข้อผิดพลาด

2.8 Synchronous Data Transmission

การส่งสัญญาณซิงโครนัส ไม่ได้ใช้ไมโครคอนโทรลเลอร์ของสมาร์ตการ์ด ในเมื่อมันเป็นการติดต่อกับปลายทางแบบซิงโครนัส อย่างไรก็ตามเป็นกระบวนการมาตรฐานของเมมโมรีการ์ด ซึ่งใช้กับตัวเลขจำนวนมาก คือ การจ่ายเงินทางอิเล็คทริคของการ์ดโทรศัพท์

ในเมมโมรีการ์ดการส่งข้อมูลแบบซิงโครนัส เป็นการเชื่อมต่อกับตัวชิป และเป็นการออกแบบที่น่าจะเป็นไปได้ มันไม่ได้เป็นการแยกเลเยอร์ในทรานสมิทชั่นโปรโตคอล ไม่เหมือนกับลอจิกคอลแอดเดรสซิง ดังการใช้งานในด้านปลายทางต้องเป็นแบบ directly access memory address ใน ตัวชิปโปรโตคอล สามารถเก็บข้อมูลลงในชิปแบบฟิสิกส์คอลแอดเดรส และเมื่ออ่านหรือเขียน นี่หมายความว่ากระบวนการของการส่งข้อมูลจริง ๆ นั้นจะต้องเชื่อมต่อไปยังฟังก์ชันของเมมโมรีแอดเดรสซิงของเมมโมรีการ์ด

นี่ไม่เป็นกระบวนการของการดีเท็คติ้งหรือคอลเร็คติ้งเออร์เรอร์ (correcting error) ของการส่งถ่ายข้อมูล ถึงแม้ว่ามันจะเป็นที่กล่าวว่าเกิดเหตุผิดพลาดระหว่างการ์ดและเทอร์มินัลไม่บ่อยนั้น ถ้าทางปลายทางใช้การ detect transfer error มันจะต้องอ่านพื้นที่ของหน่วยความจำในการ์ดใหม่ การจำกัดการบริหารของข้อมูลที่ได้ส่งถ่ายระหว่างการ์ดและเทอร์มินัลที่อัตราสูงและใช้เฉพาะลอจิกฮาร์ดแวร์ ในปริมาณน้อย

เนื่องด้วยการส่งถ่ายข้อมูลแบบซิงโครนัส ใช้สร้างเฉพาะส่งถ่ายข้อมูลแบบพื้นฐาน (หมายถึงจำนวนลอจิกมีน้อย) ผลลัพธ์ที่แทบจะไม่สามารถหลีกเลี่ยงได้คือ ขึ้นอยู่กับฮาร์ดแวร์ที่หมายถึง ซิงโครนัสทรานสมิทชั่นโปรโตคอลไม่เป็นแบบแผน และบางครั้งเปลี่ยนแปลงจากชิปถึงชิป มีแต่ ATR เท่านั้นที่เป็นมาตรฐาน เทอร์มินัลมีการติดต่อกับเมมโมรีการ์ดประเภทต่าง ๆ ดังนั้นการรวมกันของวิธีการหลาย ๆ วิธีการ ที่แตกต่างกันของซิงโครนัสทรานสมิทชั่นโปรโตคอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในกรณีของเมมโมรีการ์ดอย่างง่ายแทบจะไม่ทำการเออร์เรอร์ดีเทคชั่นของข่าวสาร ซึ่งหมายถึงไม่มีการเติมพาริตี้บิต (parity bit) ในการส่ง ความน่าจะเป็นขั้นต่ำของอัตราการส่งถ่ายข้อมูลที่เหมาะสมกับอัตราสัญญาณนาฬิกาต่ำ ซึ่งขอบเขตอยู่ระหว่าง 10 กิโลเฮิรตซ์ ถึง 100 กิโลเฮิรตซ์ ในเมื่อ 1 บิตถูกส่งไปสำหรับแต่ละรอบสัญญาณนาฬิกา อัตราสัญญาณนาฬิกาที่เป็น 20 กิโลเฮิรตซ์ ผลิตอัตราการส่งข้อมูลได้เป็น 20 กิโลบิตต่อวินาที อัตราที่ได้ก็ยังคงต่ำ เนื่องจากการเพิ่มแอดเดรส ของข่าวสารที่ต้องส่งสำหรับการ์ด

ในความเหมาะสมที่จะอธิบาย การส่งถ่ายข้อมูลแบบซิงโครนัสของเมมโมรีการ์ด ใน ลักษณะที่สามารถเข้าใจได้ สิ่งแรกที่เราต้องการเพื่อจะอธิบายถึงลักษณะพื้นฐานของเมมโมรีการ์ด ในรูปแบบที่ง่ายที่สุดคือ การ์ดจะมีหน่วยความจำโดยแบ่งออกเป็น 2 ส่วน ซึ่งคือส่วนของคงที่ คือ ROM และที่สามารถเขียนและลบได้คือ EEPROM ทั้งสองส่วนนี้อ้างข้อมูลแบบบิตได้ และสามารถอ่านได้ ในกรณีของ EEPROM นั้น สามารถเขียนและลบได้ด้วย

2.9 The Telephone Chip Protocol

นี่เป็นการแสดงให้เห็นถึงการส่งถ่ายข้อมูลของบัตรโทรศัพท์ที่บรรจุชิป Infineon SLE4403 อยู่ หน่วยความจำในไอซีตัวนี้เป็นแบบ bit – oriented ซึ่งหมายความว่าการทำงานทั้งหมดนั้นจะทำงานเป็นแบบแต่ละบิต ชิปตัวอื่น ๆ ก็จะมีโปรโตคอลแตกต่างไปจากที่อธิบายอยู่นี้ อย่างไรก็ตาม พื้นฐานเบื้องต้นของการส่งถ่ายข้อมูลของการ์ดแบบซิงโครนัสก็ทำเหมือนกันทั้งหมด

ในการส่งถ่ายข้อมูลนั้นจะใช้อยู่ 3 ส่วนสำคัญ ส่วนของการส่งข้อมูลแบบ 2 ทิศทาง (bi-directional lead) จะใช้การแลกเปลี่ยนข้อมูลระหว่างการ์ดและเทอร์มินัลเป็นแบบ single-bit ส่วนของสัญญาณนาฬิกา (clock lead)ในการต่อจะถูกสร้างจากเทอร์มินัลส่งไปสู่การ์ด สัญญาณนาฬิกานี้กำหนดโดยการอ้างอิงถึง การส่งถ่ายสัญญาณซิงโครนัส ส่วนที่ 3 ที่ต้องการสำหรับการส่งถ่ายข้อมูล ส่วนของการควบคุม (control lead)

ในหลักการการควบคุมเมมโมรีการ์ดอย่างสมบูรณ์นั้นต้องการชิปวงจรลอจิกทางไฟฟ้า ทำการถอดรหัสสำหรับ 4 ฟังก์ชัน มีการอ่าน การเขียน การลบหน่วยความจำและการเพิ่มตัวชี้ตำแหน่งของหน่วยความจำ มีตัวชี้ที่ชี้หน่วยความจำโกลบอล ซึ่งขอบเขตของหน่วยความจำเป็น address bit by bit ถ้าพอยท์เตอร์ ถึงขอบเขตบนสุดของหน่วยความจำมันจะย้อนกลับมาที่ศูนย์ ซึ่งการออกแบบ bit-oriented ชิปนั้นเมื่อมันชี้ไปที่บิตแรกในหน่วยความจำหนึ่งในฟังก์ชันของการส่งถ่ายข้อมูลแบบซิงโครนัส จะทำการตั้งค่าพอยท์เตอร์นี้ใหม่ ที่ค่าเริ่มต้น (initial value) ซึ่งปกติจะมีค่าเป็นศูนย์

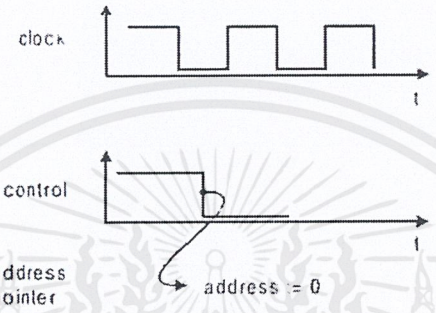
ฟังก์ชันต่อไปคือ การอ่านข้อมูลจากหน่วยความจำ อีก 2 ฟังก์ชันที่เหลือคือ การเขียนและการลบ EEPROM การลบ EEPROM นั่นคือ ยอมให้เขียนข้อมูลใหม่นั่นเอง ดังเช่นในบัตรโทรศัพท์

การรีเซ็ทตำแหน่งแอดเดรสนั้น ตัวชี้ตำแหน่งจะตั้งค่าใหม่ไปที่ค่าเริ่มต้น คือศูนย์นั่นเอง โดย power-up logic ของการ์ด ถ้าส่วนของสัญญาณนาฬิกาและส่วนควบคุมมีค่าระดับแรงดันสูง (high level) อย่างไรก็ตาม คอนโทรลพัลส์จะใช้สำหรับคาบเวลาที่ค่อนข้างยาวกว่าสัญญาณนาฬิกา เป็นความเหมาะ-

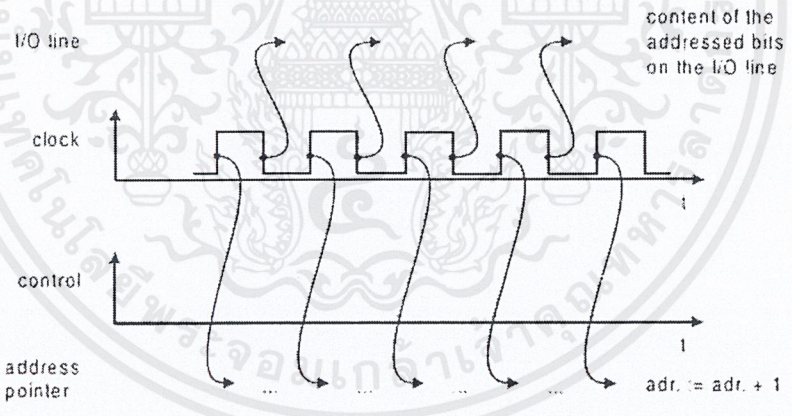
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สมที่คาดการณ์ว่า ค่าแอดเดรสจะถูกเพิ่มขึ้นอย่างฉับพลัน ตัวชี้ตำแหน่งจะตั้งค่าตัวเองใหม่ไปที่ค่าเริ่มต้นหลังทุก ๆ ลำดับการกระตุ้น ในเมื่อมันจะเป็นอีกอย่างหนึ่งคือ จะชี้ไปที่ส่วนที่ไม่ได้ถูกกำหนด

การเพิ่มค่าตัวชี้ตำแหน่ง และ การอ่านข้อมูล ถ้าสัญญาณควบคุมมีระดับแรงดันต่ำ (low level) และสัญญาณนาฬิกาเป็นขอบขาขึ้น ลอจิกภายในของการ์ดจะเพิ่มค่าตัวชี้ตำแหน่งไป 1 ค่า ในขอบขาลงของสัญญาณนาฬิกา ค่าที่ถูกชี้ในตัวชี้จะถูกแทนที่ลงไปในส่วนของข้อมูล ถ้าตัวชี้มาถึงค่าสูงสุดซึ่งขึ้นอยู่กับขนาดของหน่วยความจำ มันจะย้อนกลับมาที่ศูนย์และเริ่มจากจุดเริ่มต้น



รูปที่ 2.27 การรีเซ็ตตัวชี้ตำแหน่งแอดเดรสไปที่ศูนย์



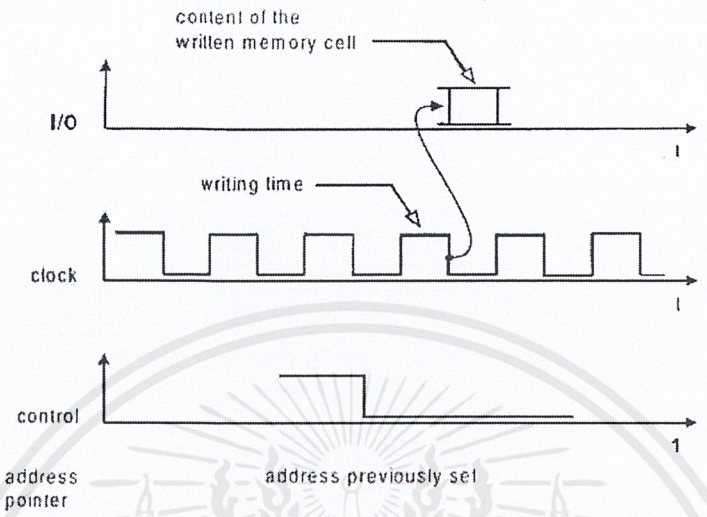
รูปที่ 2.28 แสดงการเพิ่มตำแหน่งแอดเดรสและการอ่านข้อมูลจากแอดเดรส

การเขียนกำหนดตำแหน่ง ถ้าตัวชี้ตำแหน่งอยู่ในช่วงของการเขียนข้อมูลลงสู่ EEPROM ข้อมูลในส่วนของข้อมูลจะถูกเขียนลงสู่ EEPROM โดยการให้ระดับแรงดันสูง กับส่วนของการควบคุมและส่วนของสัญญาณนาฬิกามีค่าแรงดันต่ำ ความยาวของรอบในการเขียนข้อมูล (write cycle) ถ้าบิตถูกเขียนลงไปอย่างถูกต้องแล้ว ข้อมูลที่บรรจุในเซลล์ของหน่วยความจำที่เขียนลงไป จะไปปรากฏที่ข้อมูลเอาท์พุท

การลบไบท์ข้อมูลส่วนของ EEPROM ในต้นแบบของบัตรโทรศัพท์จะประกอบด้วย 8 ส่วน ถ้าไบท์มีการลบในการลบล้างนี้เพื่อที่จะนำไปสู่ที่ต่อไป นี่เป็นการกระทำของวงจรลอจิกทางไฟฟ้า การลบข้อมูลในไบท์ของหน่วยความจำค่อนข้างยุ่งยาก มีการทำงานดังนี้ ถ้าบิตในไบท์ถูกเขียนลงไปทั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2 แถว ชิปจะทำการลบข้อมูลที่ LSB โดยอัตโนมัติ นี่ทำให้มั่นใจได้ว่ามันจะถูกนำไปสู่ที่ต่อไปที่สูงกว่าที่มีอยู่ ในขณะที่ที่ตำแหน่งต่ำกว่าถูกลบไปโดยปราศจากจังหวะสำหรับการหลอกลวง



รูปที่2.29 แสดงการเขียนข้อมูลลงในหน่วยความจำ EEPROM

2.10 สถาปัตยกรรมของไมโครคอนโทรลเลอร์แบบชิปเดี่ยว ตระกูล MCS-51

ไมโครคอนโทรลเลอร์แบบชิปเดี่ยว (Single Chip microcontroller) คือไมโครคอมพิวเตอร์แบบที่มีขนาดเล็กโดยบรรจุไว้ในแผงวงจรรวม (Single Circuit) ชิปเดี่ยวเหมาะสำหรับงานควบคุมอุปกรณ์อื่น ๆ แบบอัตโนมัติ เพราะผู้ใช้สามารถเขียนโปรแกรมควบคุมการทำงานได้ตามต้องการไมโครคอนโทรลเลอร์แบบชิปเดี่ยวตระกูล 51 หรือ MCS-51 อันได้แก่ เบอร์ 8051, 8052 และอื่น ๆ

MCS-51 มีข้อดีดังนี้

- สามารถนำเอาข้อมูลมา AND, OR หรือทำ Complement ทั้งแบบทีละ 8 บิต และ 1 บิต
- สามารถใช้กับหน่วยความจำสำหรับโปรแกรม (Program Memory) ซึ่งเป็นหน่วยความจำที่ใช้สำหรับเก็บชุดคำสั่งที่จะให้ MCS-51 ทำงานได้สูงสุด 64 กิโลไบท์ (Kilobyte) (64 x 1024 ไบท์) ทำให้เขียนโปรแกรมควบคุมการทำงานได้มาก
- สามารถต่อกับหน่วยความจำสำหรับข้อมูล (Data Memory) ซึ่งเป็นหน่วยเก็บความจำสำหรับเก็บข้อมูลในระหว่างการทำงานของโปรแกรมได้สูงสุด 64 กิโลไบท์
- ใน 8051 และ 8751 มีหน่วยความจำสำหรับโปรแกรมจำนวน 4 กิโลไบท์ (ใน 8052 และ 8752 มีหน่วยความจำสำหรับโปรแกรมจำนวน 8 กิโลไบท์) อยู่ภายในวงจรรวมทำให้ไม่ต้องต่อหน่วยความจำสำหรับโปรแกรมอยู่ภายนอก ระบบรวมทั้งหมดจึงมีขนาดเล็กและสัญญาณรบกวนจากภายนอกจะทำให้ MCS-51 ทำงานผิดพลาดได้ยาก
- มีพอร์ทแบบขนาน (Parallel Port) สำหรับข้อมูลเข้าและออกจำนวน 32 บิต ที่ข้อมูลแต่ละบิตเป็นอิสระต่อกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- มีวงจร Timer/Counter ขนาด 16 บิต 2 ชุด (8052 มี 3 ชุด) ที่ทำงานในโหมดต่าง ๆ ได้ถึง 4 โหมด
- มี Universal Asynchronous Receiver Transmitter (UATR) สำหรับรับ-ส่งข้อมูลอนุกรม (Serial) แบบ Full Duplex ที่สามารถเลือกรูปแบบการรับ-ส่งข้อมูลได้ 4 แบบ
- มีแหล่งกำเนิดสัญญาณขอขัดจังหวะการทำงานของโปรแกรม (Interrupt Request signal) 6 แหล่ง ซึ่งสามารถทำการกระโดดไปทำงานตอบสนองการขัดจังหวะ (Interrupt Service Routine ได้ต่าง ๆ กัน 5 ตำแหน่ง
- สามารถเลือกการทำงานให้อยู่ในโหมดของ Idle และ Power Down ซึ่งประหยัดการใช้กำลังไฟในการทำงาน

ในโครงงานนี้ ได้ใช้ไมโครคอนโทรลเลอร์เบอร์ AT89S8252 ของบริษัท ATMEL ซึ่งเป็นไมโครคอนโทรลเลอร์ในตระกูลเดียวกันกับ MCS-51 เช่นกัน แต่มีส่วนที่แตกต่างจาก 8051 ดังนี้

- 8 กิโลไบท์ สำหรับการโปรแกรมหน่วยความจำแบบแฟลซ
- SPI การเชื่อมต่อแบบอนุกรม สำหรับการดาวน์โหลด (Down load) โปรแกรม
- อายุการใช้งาน สามารถเขียนและลบได้ถึง 1,000 ครั้ง
- 2 กิโลไบท์ สำหรับ EEPROM
- ช่วงการทำงานที่ 4.0 ถึง 6 โวลต์
- ช่วงทำงานเต็มที่ 0-24 เมกกะเฮิรตซ์
- สามารถโปรแกรม Memory Lock ได้ 3 ระดับ
- 256 x 8 bit ใน internal RAM
- สามารถโปรแกรมผ่านช่องสัญญาณอนุกรมแบบ UATR ได้
- สามารถโปรแกรม ทางด้านอินพุท/เอาท์พุท (I/O) ได้ 32 เส้น
- Timer/counter 16 bit มี 3 ตัว
- มีอินเตอร์รัพท์ 9 แหล่ง
- มี SPI เชื่อมต่อแบบอนุกรม
- กินกำลังไฟต่ำในโหมด Idle และ กำลังไฟฟ้าตก
- อินเตอร์รัพท์กลับคืนมาจากกำลังไฟฟ้าตกได้
- สามารถโปรแกรมเป็น watchdog time ได้
- มีพอยท์เตอร์ข้อมูล 2 ชุด

2.10.1 การเชื่อมต่ออุปกรณ์แบบอนุกรม

เป็นลักษณะการโอนย้ายข้อมูลแบบซิงโครนัสความเร็วสูง ระหว่าง AT89S8252 กับ อุปกรณ์เพอริเฟอรัล โดย SPI ใน AT89S8252 มีคุณลักษณะดังนี้

- ฟูลดูเพล็กซ์ , สาย 3 เส้น ในการโอนย้ายข้อมูลซิงโครนัส
- การทำงานแบบมาสเตอร์หรือสลาฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 1.5 เมกกะเฮิรตซ์ ความถี่บิตสูงสุด
- LSB ก่อนหรือ MSB ก่อน ในการโอนย้ายข้อมูล
- การโปรแกรมมี 4 bit rate
- สิ้นสุดการส่งด้วย Interrupt flag
- ป้องกันการเขียนซ้ำด้วย Write Collision Flag

การเชื่อมต่อระหว่าง Master และ Slave CPUS

ขา SCK จะเป็นเอาท์พุทในโหมด Master และจะเป็น Clock input ในโหมด Slave การเขียนลง SPI data register ของ Master เริ่มต้นด้วย SPI ผลิต Clock และข้อมูลที่เขียนจะเลื่อนไปยังขา MOSI ของ CPU หลังจากเลื่อนไป 1 byte SPI Clock จะเซ็ทเป็นสิ้นสุดการส่งด้วย SPIE ถ้า SPIE และ ES = "1" จะต้องการการอินเตอร์รัพต์

SSI (Slave Select Input) จะถูกเซ็ทเป็น "0" ไปเลือก SPI Device เป็น Slave แต่เมื่อให้ SSI = "1" SPI Port จะไม่กระทำใด ๆ และขา MOSI สามารถใช้เป็นอินพุทได้

มีการรวม SCK เฟสและขั้ว (phase and polarity) 4 แบบใช้ตรงกับข้อมูลอนุกรม ซึ่งถูกควบคุมด้วยบิต CPHA และ CPOL

Program Memory Lock Bit

มีการล็อกบิต 3 แบบ สามารถใช้กับแบบไม่โปรแกรม (U) หรือใช้โปรแกรม (P) โดยต้องเป็นไปตามตาราง

เมื่อล็อกบิต 1 (LB1) เป็นโปรแกรมระดับลอจิกที่ขา EA จะเป็นการแลทซ์ระหว่างรีเซ็ท ถ้าอุปกรณ์ไม่รีเซ็ท การแลทซ์จะกำหนดให้เป็นค่าสุ่ม และถือว่าค่านั้นจนกระทั่งจะรีเซ็ท ค่ารีแลทซ์ของ EA จะต้องได้รับกระแสที่ขานั้น

	LB1	LB2	LB3	
1	U	U	U	หน่วยความจำภายในจะไม่ถูกล็อก
2	P	U	U	คำสั่ง MOVC ที่ execute จากโปรแกรมภายนอกจะถูก disable จากการ fetching code จากภายใน EA จะสุ่มค่าและแลทซ์ในการเซ็ทของการ Program Flash Memory เป็น Disable
3	P	P	U	เหมือน mode2 แต่การขนานหรืออนุกรมจะถูกตรวจสอบให้ disable
	P	P	P	เหมือน mode3 แต่ภายนอกจะถูก execute ทั้งหมดเป็น disable จากโปรแกรม Flash และ EEPROM

ภายในชิป AT89S8252 จะมี Flash code และ EEPROM data memory การเขียนโปรแกรมต้องมีแรงดันสูง 12 โวลต์ สำหรับโหมดการโปรแกรมแบบขนาน และแรงดันต่ำ 5 โวลต์ สำหรับโหมดแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อนุกรมสำหรับพื้นที่ของ code memory เท่ากับ 0000H – 1FFFFH และพื้นที่ของ data memory เท่ากับ 2000H – 27FFH สำหรับการโปรแกรมแบบขนาน

ขั้นตอนการโปรแกรมแบบขนาน

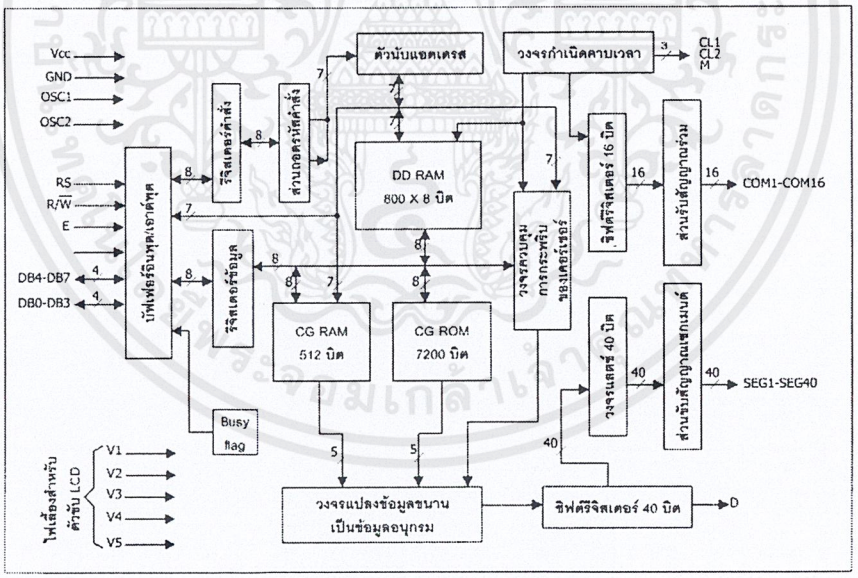
1. ป้อนหรือจ่ายกำลังไฟฟ้าระหว่าง Vcc กับ GND Set RST เป็น “H”การจ่าย 3 เมกกะเฮิรตซ์ – 24 เมกกะเฮิรตซ์ ของสัญญาณนาฬิกา ต้องรออย่างน้อย 10 มิลลิวินาที
2. เซ็ท PSEN = “L” ALE เป็น “H”

2.10.2 โครงสร้างภายในของตัวควบคุมโมดูล LCD

ในการใช้งานโมดูล LCD จำเป็นต้องทำความเข้าใจเกี่ยวกับโครงสร้างและคำสั่งที่ใช้ในการควบคุมให้ดีเสีย ก่อนในที่นี้ ขอยกตัวอย่างโมดูล LCD แบบอักขระเพราะสามารถเข้าใจได้ง่าย รูปที่ 2.30 เป็นบล็อกไดอะแกรมภายในของชิปควบคุม LCD เบอร์ HD44780 ซึ่งประกอบด้วย

บัฟเฟอร์อินพุทเอาท์พุทเป็นส่วนที่ใช้ในการติดต่อรับส่งข้อมูลกับอุปกรณ์ภายนอก เพื่อที่จะถ่ายทอดข้อมูลเข้าออกภายในตัวควบคุม

รีจิสเตอร์คำสั่ง (Instruction Register: IR) เป็นรีจิสเตอร์ใช้รับข้อมูลคำสั่งจากอุปกรณ์ภายนอก เพื่อนำไปควบคุมการแสดงผล



รูปที่ 2.30 แสดงบล็อกไดอะแกรมการทำงานภายในของโมดูลแบบตัวอักษร

รีจิสเตอร์ข้อมูล (Data Register: DR) เป็นรีจิสเตอร์ใช้รับข้อมูลจากอุปกรณ์ภายนอกเพื่อถ่ายทอดต่อไปยังแรมเก็บข้อมูลแสดงผล หรือนำข้อมูลนั้นไปเพื่อสร้างตัวอักษรเพิ่มเติมในแรมเก็บตัวอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

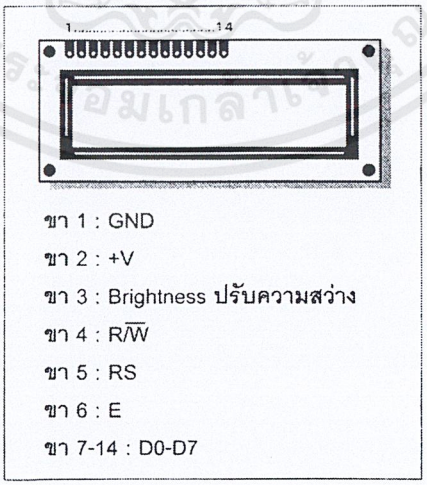
แรมเก็บข้อมูลแสดงผล (Display Data RAM: DDRAM) เป็นหน่วยความจำแรม ทำหน้าที่เก็บข้อมูลที่มาจากรีจิสเตอร์ DR ตัวควบคุมจะนำข้อมูลใน DDRAM นี้ไปเปิดตาราง (Look up-table) ของตัวอักษรที่เก็บไว้ในหน่วยความจำรอมและแรมเก็บตัวอักษร เพื่อนำไปแสดงที่ตัวแสดงผล

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

แรมเก็บตัวอักษร (Character Generator RAM: CGRAM) เป็นหน่วยความจำแรมที่ใช้เก็บอักษรที่มีการสร้างเพิ่มเติมขึ้นใหม่ ในกรณีที่ตัวอักษรใน CGROM ไม่เพียงพอ มีขนาด 512 บิต การเขียนและอ่านค่าไปใช้นั้นทำได้เช่นเดียวกับ CGROM คือเขียนข้อมูลลงใน CGRAM แล้วตัวควบคุมจะมาอ่านค่าจาก CGRAM เอง

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

ตารางที่ 2.4 แสดงความสามารถของขา RS, R / $\overline{W}$ และ E



รูปที่ 2.31 แสดงการจำขาของโมดูล LCD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แฟลก BUSY เป็นส่วนที่ทำหน้าที่แจ้งสถานะการทำงานของตัวควบคุมให้อุปกรณ์ภายนอกทราบว่า ตัวควบคุมพร้อมที่จะรับข้อมูลหรือคำสั่งหรือไม่ ดังนั้นก่อนการส่งข้อมูลหรือคำสั่งมายังตัวควบคุมต้องตรวจสอบสถานะของแฟลก BUSY นี้เสียก่อน

- โมดูล LCD ขนาด 16 ตัวอักษร 1 บรรทัด

สำหรับโมดูล LCD ขนาด 16 ตัวอักษร 1 บรรทัด มีขาต่อใช้งานทั้งสิ้น 14 ขา มีการจัดขาดังในรูปที่ 2.31 สำหรับรายละเอียดการทำงานของแต่ละขามีดังนี้

V_{SS} (ขา 1): ต่อกราวด์

V_{DD} (ขา 2): ต่อไฟเลี้ยง + 5 V

V_O (ขา 3): เป็นขาอินพุตสำหรับป้อนแรงดัน เพื่อปรับความเข้มของการแสดงผล

RS (ขา 4): เป็นขาอินพุต ใช้เลือกว่าข้อมูลที่ทำการส่งในขณะนั้นเป็นคำสั่งสำหรับรีจิสเตอร์ IR หรือเป็นข้อมูลสำหรับรีจิสเตอร์ DR โดยถ้าขานี้เป็น “0” ข้อมูลที่ส่งมาจะเป็นคำสั่งแต่ถ้าเป็น “1” ข้อมูลที่ส่งมาจะเป็นข้อมูลสำหรับการแสดงผล

R / $\overline{W}$ (ขา 5): เป็นขาที่ใช้เลือกว่าจะอ่านหรือเขียนข้อมูลกับ LCD ถ้าหากเป็น “0” จะเป็นการเขียนข้อมูล แต่ถ้าเป็น “1” จะเป็นการอ่านข้อมูล

E (ขา 6): เป็นขาอีนาเบิล LCD ให้ทำงาน

DB0-DB7 (ขา 7 – 14): เป็นขาที่ใช้เป็นทางผ่านของข้อมูลระหว่าง LCD กับอุปกรณ์ภายนอกขนาด 8 บิต

อนึ่งขา RS, R / $\overline{W}$ และ E จะใช้งานร่วมกัน โดยมีลักษณะความสัมพันธ์ดังในตารางที่ 2.4

- คำสั่งควบคุมโมดูล LCD

ในการเขียนคำสั่งลงในตัวควบคุม แน่นอนว่าต้องกำหนดให้ขา RS และ R / $\overline{W}$ เป็น “0” ทั้งคู่ แล้วเขียนข้อมูลคำสั่งตามไป คำสั่งควบคุมโมดูล LCD ของชิปควบคุม HD44780 ที่สำคัญมี 9 คำสั่งดังนี้

1. คำสั่งเคลียร์ตัวแสดงผล (clear display) มีข้อมูลคำสั่งเป็น 01H เป็นคำสั่งที่ใช้เขียนข้อมูลช่องว่าง หรือ space เข้าไปใน DDRAM ทั้งหมดเมื่อตัวควบคุมเอ็กซิคิวต์คำสั่งดังนี้ จะทำการกำหนดแอดเดรสของ DDRAM เป็น 0 เคอร์เซอร์จะกลับไปอยู่ที่ตำแหน่งซ้ายมือสุดของจอแสดงผลแล้วเซตบิต I/D ให้เป็น “1”

2. คำสั่ง return home ต้องกำหนดให้บิต 1 ของข้อมูลเป็น “1” เป็นคำสั่งให้เคอร์เซอร์เคลื่อนที่กลับไปยังตำแหน่งซ้ายสุดของจอแสดงผล แต่ข้อมูลบนจอแสดงผลไม่เปลี่ยนแปลง นั่นคือข้อมูลคำสั่งของคำสั่งนี้จะเป็น 02H หรือ 03H ก็ได้

3. คำสั่งเลือกโหมดการป้อนข้อมูล (Entry mode Set) คำสั่งมีรูปแบบดังนี้

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บิต S เป็นบิตที่ใช้ในการกำหนดลักษณะของการแสดงผล เมื่อมีการป้อนข้อมูล ถ้าหากบิต S เป็น “1” เมื่อเกิดข้อมูลใหม่บนจอแสดงผล ตัวเคอร์เซอร์จะอยู่กับที่ แต่ตัวอักษรข้อมูลเดิมจะถูกดันไปทางซ้าย แต่ถ้าหากบิตนี้เป็น “0” เมื่อเกิดข้อมูลใหม่ตัวเคอร์เซอร์จะเลื่อนไปทางขวามือ

บิต I/D เป็นบิตที่ใช้ในการกำหนดว่า เมื่อเขียนหรืออ่านข้อมูลแล้ว ทำให้แอดเดรสของ DDRAM เพิ่มของหรือลดลงหนึ่งแอดเดรส โดยถ้าบิตนี้เป็น “1” แอดเดรสของ DDRAM จะเพิ่มขึ้น แต่ถ้าเป็น “0” แอดเดรสจะลดลง

ดังนั้นข้อมูลคำสั่งที่เกิดขึ้นสำหรับคำสั่งนี้ได้แก่ 04H-07H (4 ข้อมูลคำสั่ง) และที่ใช้บ่อยคือ 06H หมายถึงกำหนดให้เมื่อเกิดข้อมูลใหม่ เคอร์เซอร์จะเลื่อนไปทางขวามือ และแอดเดรสของ DDRAM เพิ่มขึ้น

4. คำสั่งควบคุมการแสดงผล มีรูปแบบคำสั่งดังนี้

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

บิต D ใช้ควบคุมการเปิดปิดจอแสดงผล ถ้าบิตนี้เป็น “1” จะเป็นการเปิดจอแสดงผลถ้าเป็น “0” จะเป็นการปิดจอแสดงผล

บิต C ใช้ควบคุมการแสดงตัวเคอร์เซอร์บนจอแสดงผล ถ้าต้องการให้มีเคอร์เซอร์แสดงผลบนจอแสดงผลต้องกำหนดให้บิตนี้เป็น “1” ถ้ากำหนดให้เป็น “0” จะเป็นการปิดเคอร์เซอร์หรือไม่แสดงเคอร์-เซอร์

บิต B ใช้ควบคุมการกระพริบของเคอร์เซอร์ ถ้าบิตนี้เป็น “1” เคอร์เซอร์จะกระพริบ

ดังนั้นจะมีข้อมูลคำสั่งได้ตั้งแต่ 08H –0FH (8 รูปแบบคำสั่ง) ที่ใช้บ่อยคือ 0CH เป็นการสั่งให้เปิดจอแสดงผล แต่ไม่แสดงเคอร์เซอร์ และ 0FH เป็นการสั่งให้เปิดจอแสดงผลแสดงเคอร์เซอร์และ สั่งให้เคอร์เซอร์กระพริบ

5. คำสั่งควบคุมการเลื่อนเคอร์เซอร์และข้อมูลตัวอักษร มีรูปแบบข้อมูลคำสั่ง ดังนี้

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

การควบคุมการเลื่อนเคอร์เซอร์และตัวอักษรบนจอแสดงผล ขึ้นอยู่กับการกำหนดบิต S/C และ R/L ซึ่งสามารถสรุปได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บิต S/C	บิต R/L	ลักษณะการเลื่อน	ข้อมูลคำสั่ง
0	0	เลื่อนเคอร์เซอร์จากตำแหน่งเดิมไปทางซ้าย 1 ตำแหน่ง	10H-13H
0	1	เลื่อนเคอร์เซอร์จากตำแหน่งเดิมไปทางขวา 1 ตำแหน่ง	14H-17H
1	0	เลื่อนตัวอักษรที่เกิดขึ้นใหม่ไปทางซ้าย	18H-1BH
1	1	อักษรที่เกิดขึ้นใหม่ไปทางขวา	1CH-1FH

6. คำสั่งกำหนดฟังก์ชันการทำงาน มีรูปแบบคำสั่ง ดังนี้

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

บิต DL จะใช้กำหนดจำนวนบิตที่ใช้ติดต่อส่งผ่านข้อมูล ถ้าบิตนี้เป็น “0” จะเป็นการติดต่อแบบ 4 บิต แต่ถ้าเป็น “1” จะเป็นแบบ 8 บิต

บิต N ใช้กำหนดจำนวนบรรทัดของการแสดงผล ถ้าเป็น “0” จะแสดงผล 1 บรรทัดถ้าเป็น “1” จะแสดง 2 บรรทัด ในกรณีที่จอแสดงผลสามารถแสดงได้มากกว่า 2 บรรทัดและต้องการให้แสดงผลมากกว่า 2 บรรทัดก็กำหนดบิต N นี้ให้เป็น “1”

บิต F ใช้เลือกข้อความละเอียดของตัวอักษรให้การแสดงผล ถ้าบิตนี้เป็น “0” จะเป็นการแสดงผลแบบ 5 x 7 จุด และถ้าเป็น “1” จะแสดงผลเป็นแบบ 5 x 10 จุด

ข้อมูลคำสั่งที่ใช้บ่อยคือ 38H เป็นการกำหนดให้โมดูล LCD ทำงานในแบบ 4 บิตโดยจะแสดงผล 2 บรรทัดและเลือกความละเอียดเป็น 5 x 7 จุด

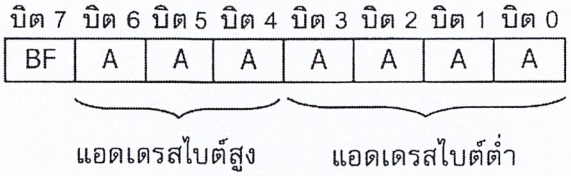
จุดที่น่าสังเกตคือ โมดูล LCD แบบ 16 ตัวอักษร 1 บรรทัด แม้จะมีบรรทัดการแสดงผลเพียงแค่ 1 บรรทัดแต่จะต้องกำหนด N ให้เป็น “1” เนื่องจากแอดเดรสของ DDRAM แบ่งเป็น 2 ช่องคือ 0x00 และ 0x40

7. คำสั่งเลือกแอดเดรสของ CGRAM เมื่อต้องการกำหนดแอดเดรสของ CGRAM ต้องกำหนดให้บิต 7 เป็น “0” บิต 6 เป็น “1” ส่วนอีก 6 บิตที่เหลือจะแทนด้วยค่าของแอดเดรสของ CGRAM จะต้องทำการกำหนดแอดเดรสด้วยคำสั่งนี้ ก่อนที่จะอ่านหรือเขียนข้อมูลให้ CGRAM โดยแอดเดรสของ CGRAM อยู่ระหว่าง 00H-3FH

8. คำสั่งเลือกแอดเดรสของ DDRAM ใช้ในการเลือกแอดเดรสของ DDRAM ก่อนที่จะทำการอ่านหรือเขียนข้อมูล โดยบิต 7 ต้องเป็น “1” และข้อมูลอีก 7 บิตที่เหลือจะเป็นค่าแอดเดรสของ DDRAM ซึ่งแอดเดรสของ DDRAM จะอยู่ระหว่าง 8CH-0FFH ทั้งนี้จำนวนแอดเดรสยังขึ้นกับการกำหนดสถานะที่บิต N ด้วย หากบิต N เป็น “0” แอดเดรสของ DDRAM จะอยู่ระหว่าง 80H-0CFH และถ้าบิต N เป็น “1” แอดเดรสของ DDRAM จะมี 2 ช่วงคือ 80H-87H และ C0H-C7H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9. คำสั่งอ่านแฟลก busy และแอดเดรส มีรูปแบบของบิตข้อมูลคำสั่งดังนี้



เป็นคำสั่งที่ใช้อ่านแฟลก BUSY (BF) โดยแฟลกนี้จะเป็นตัวบอกสถานะของตัวควบคุม LCD ว่าพร้อมที่จะรับข้อมูลอยู่หรือไม่ ถ้าหากบิต BF เป็น “0” แสดงว่าตัวควบคุม LCD พร้อมรับข้อมูลหรือคำสั่ง แต่ถ้าเป็น “1” แสดงว่า ขณะนี้ตัวควบคุม LCD ยังอยู่ในกระบวนการทำงานภายในหรือกำลังประมวลผลข้อมูลอยู่ ยังไม่พร้อมรับข้อมูลหรือคำสั่ง เมื่อต้องการอ่านแฟลกต้องกำหนดให้ขา R / $\overline{W}$ เป็น “1” ด้วย แต่สัญญาณที่ RS ยังต้องเป็น “0” อยู่เพราะข้อมูลนี้เป็นข้อมูลคำสั่ง

นอกจากนี้ ยังใช้เป็นคำสั่งอ่านข้อมูลแอดเดรสของ CGRAM และ DDRAM ด้วย โดยบิต 0 ถึงบิต 6 เป็นค่าข้อมูลของแอดเดรสที่ต้องการอ่าน

- การติดต่อเพื่อเขียนคำสั่งและข้อมูลให้แก่โมดูล LCD

ในการติดต่อกับโมดูล LCD จะต้องมีการหน่วงเวลาหลังจากที่ทำการส่งรหัสคำสั่งหรือข้อมูล เนื่องจากต้องรอให้คอนโทรลเลอร์ภายในโมดูล LCD แปลความหมายของรหัสคำสั่งและทำงานตามคำสั่งให้เรียบร้อยก่อน จากนั้นจึงจะรับข้อมูลหรือดำเนินการต่อไป แต่เนื่องจากการส่งข้อมูลจากพอร์ทอนุกรมไปยังโมดูล LCD นั้นค่อนข้างช้า ผู้ใช้งานจึงสามารถข้ามขั้นตอนการหน่วงเวลาไปได้ แต่สำหรับการใช้งานโมดูล LCD กับไมโครคอนโทรลเลอร์ หรือกับคอมพิวเตอร์ที่มีความเร็วสูงมาก การหน่วงเวลาก็ยังเป็นสิ่งจำเป็น

เมื่อเริ่มต้นใช้งานจะต้องกำหนดลอจิกให้กับ ขา RS ของโมดูล LCD เพื่อให้คอนโทรลเลอร์ในโมดูล LCD แปลความหมายของลอจิกที่ขา RS ว่า ข้อมูลต่อไปที่จะได้รับนั้นเป็นรหัสคำสั่งหรือเป็นข้อมูลที่ต้องการแสดงผล จากนั้นจะเป็นการส่งข้อมูลมารอที่บัสข้อมูล D0-D7 (กรณีทำงานในโหมด 8 บิต) หรือ D4-D7 (กรณีทำงานในโหมด 4 บิต)

ขั้นตอนต่อไปจะเป็นการส่งสัญญาณพัลส์ไปที่ขา E เพื่ออีนาเบิลโมดูล LCD ให้รับข้อมูลจากบัสข้อมูลเข้าไปโดยพัลส์ที่ป้อนเข้าที่ขา E ของโมดูล LCD ต้องเป็นพัลส์ของขาขึ้น

ในการเขียนข้อมูลเพื่อควบคุมให้โมดูล LCD แสดงผลตามที่ผู้ใช้งานต้องการ ต้องส่งคำสั่ง (instruction) แล้วกำหนดโหมดการทำงานให้แก่โมดูล LCD ก่อน จากนั้นจึงค่อยส่งข้อมูล (data) ที่ต้องการแสดงผล เนื่องจากบัสข้อมูลของโมดูล LCD มี 8 เส้น คือ D0-D7 และใช้เป็นทางผ่านของทั้งคำสั่งและข้อมูล ดังนั้นในการส่งรหัสคำสั่งและข้อมูลจึงต้องอาศัยการกำหนดสัญญาณลอจิกที่ขา RS ถ้าหากที่ขา RS ได้ลอจิก “0” หมายความว่า ข้อมูลที่ป้อนให้แก่โมดูล LCD ขณะนั้นเป็นคำสั่ง ในทางตรงข้าม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หากขา RS ได้รับลอจิก “0” หมายความว่า ข้อมูลที่ป้อนให้แก่โมดูล LCD ขณะนั้นเป็นคำสั่ง ในทางตรงข้าม หากขา RS ได้รับลอจิก “1” ข้อมูลที่ป้อนให้ขณะนั้นเป็นข้อมูลที่ใช้ในการแสดงผล

เมื่อต้องการเขียนหรืออ่านข้อมูลใน CGRAM และ DDRAM เริ่มต้นต้องกำหนดแอดเดรสที่ต้องการอ่านหรือเขียนก่อน โดยใช้คำสั่งเลือกแอดเดรส จากนั้นกำหนดให้ขา RS เป็น “1” เพื่อแจ้งให้ตัวควบคุมภายในโมดูล LCD ทราบว่าข้อมูลที่ปรากฏต่อไปนี้เป็นข้อมูลปกติหรือไม่ใช่คำสั่ง ในกรณีที่ต้องการอ่านข้อมูลต้องกำหนดให้ขา R / $\overline{W}$ เป็น “1” ข้อมูลขนาด 8 บิต (หรือ 4 บิต) ก็จะปรากฏบนบัสข้อมูล โดยข้อมูลที่อ่านออกมาได้จะเป็นข้อมูลจากแอดเดรสของ CGRAM หรือ DDRAM ตามที่ต้องการ

ในกรณีที่ต้องการเขียนข้อมูล เมื่อกำหนดแอดเดรสและป้อนลอจิก “1” ให้ขา RS แล้ว แล้วต้องกำหนดให้ขา R / $\overline{W}$ เป็น “0” ข้อมูลที่อยู่บนบัสข้อมูลจะถูกเขียนลงในรีจิสเตอร์ DR จากนั้นจึงถ่ายทอดลงใน DDRAM ต่อไป

2.11 ซอฟแวร์ที่ใช้ในการติดต่อสื่อสารผ่านพอร์ทอนุกรม

คอนโทรล MSComm (Communication) เป็นคอนโทรลตัวหนึ่งที่ใช้ช่วยในการติดต่อกับพอร์ทอนุกรม (Serial Port) ซึ่งผู้อ่านสามารถทำการรับส่งข้อมูลผ่านทางพอร์ทอนุกรมได้ด้วยคอนโทรลนี้ เช่น การติดต่อผ่านโมเด็ม (Modem) หรือติดต่อโดยตรงกับบอร์ดอิเล็กทรอนิกส์ เป็นต้น ซึ่งคอนโทรล MSComm ที่มากับ Visual Basic จะเป็นคอนโทรลที่ทำงานโดยมีการตอบสนองต่อเหตุการณ์แบบ Event-Driven นั่นก็คือคอนโทรลจะทำหน้าที่ตรวจสอบการเกิดขึ้นหรือการร้องขอให้เกิดเหตุการณ์ต่าง ๆ กับพอร์ทอนุกรมโดยอัตโนมัติ และจะมีการแจ้งเตือนให้ผู้อ่านได้ทราบโดยผ่านทางโพรซีเยอร์เหตุการณ์เช่นเดียวกับคอนโทรลทั่วไปนั่นเอง ดังนั้นในการเขียนโค้ดผู้อ่านจึงไม่จำเป็นต้องสร้างโพรซีเยอร์ที่ทำหน้าที่คอยตรวจสอบเหตุการณ์ต่าง ๆ ของพอร์ทอนุกรม ซึ่งจะทำให้ง่ายต่อการใช้งานเป็นอย่างมาก

คอนโทรล MSComm จะมีหน้าที่มาตรฐานหลัก ๆ สำหรับการสื่อสารผ่านพอร์ทอนุกรม 3 ประการดังต่อไปนี้

- หมุนหมายเลขติดต่อกับโทรศัพท์ปลายทางที่กำหนด
- ตรวจสอบการเข้ามาของข้อมูลยังพอร์ทอนุกรมโดยอัตโนมัติ
- ส่งข้อมูลตามที่กำหนดจากโปรแกรมไปยังพอร์ทอนุกรม

ในความเป็นจริงคอนโทรล MSComm ไม่ได้ทำหน้าที่ติดต่อกับพอร์ทอนุกรมโดยตรง แต่มันจะทำหน้าที่เรียกใช้ฟังก์ชันวินโดวส์ API ซึ่งวินโดวส์จะทำการส่งหรือรับข้อมูลผ่านทางพอร์ทอนุกรมโดยอาศัยไดรเวอร์ Comm.drv อีกทอดหนึ่ง ดังนั้นจึงสามารถสรุปสั้น ๆ ได้ว่าทุกครั้งที่ผู้อ่านมีการเรียกใช้คอนโทรล MSComm ก็หมายถึงการเรียกใช้ฟังก์ชันวินโดวส์ API ซึ่งจะถูกตีความอีกทอดหนึ่งโดยไดรเวอร์ Comm.drv จากนั้นก็จะส่งผ่านข้อมูลที่ถูกจัดรูปแบบตามมาตรฐานการสื่อสาร (ทั้งนี้ก็ขึ้นกับอุปกรณ์ที่ต่อเข้ากับพอร์ทอนุกรม) ให้กับดีไวซ์ไดรเวอร์อีกทอดหนึ่งนั่นเองสำหรับฟอร์มหนึ่ง ๆ ผู้อ่านสามารถเพิ่มได้หลาย ๆ คอนโทรล MSComm ทั้งนี้ก็ขึ้นกับความต้องการของผู้อ่านในการติดต่อกับพอร์ทอนุกรมใดบ้าง สำหรับวินโดวส์ 95 และ NT 4.0 ผู้อ่านสามารถที่จะติดตั้งพอร์ทอนุกรมได้มากกว่า 4 พอร์ท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยเมื่อผู้อ่านเพิ่มคอนโทรล MSComm ลงในฟอร์ม ซึ่งจะสนับสนุนคุณสมบัติ และโพซีเยอร์เหตุการณ์ที่สำคัญ ดังต่อไปนี้

- CommEvent

รายงานเหตุการณ์ทุกครั้งที่เกิดข้อผิดพลาดหรือมีการสื่อสาร ซึ่งผู้อ่านสามารถอ่านค่าได้เฉพาะในขณะทำงานเท่านั้น

รูปแบบการใช้งาน

Object.CommEvent

คอนโทรล MSComm จะมีการเรียกโพรซีเยอร์เหตุการณ์ OnComm ทุกครั้งที่เกิดข้อผิดพลาดหรือมีการสื่อสารเกิดขึ้น ซึ่งค่าตัวเลขที่จำนวนเต็ม แสดงถึงข้อผิดพลาดหรือเหตุการณ์ที่มีการสื่อสารดังกล่าว ก็จะถูกจัดเก็บเอาไว้ที่คุณสมบัติ CommEvent เสมอ ดังนั้นถ้าหากผู้อ่านต้องการตรวจสอบข้อผิดพลาดหรือเหตุการณ์ที่มีการสื่อสารภายใน โพรซีเยอร์เหตุการณ์ OnComm ก็ควรจะใช้ค่าตัวเลขจากคุณสมบัติ CommEvent ในการตรวจสอบเสมอ ซึ่งค่าตัวเลขที่รายงานโดยคุณสมบัติ CommEvent มีดังต่อไปนี้

ข้อผิดพลาด (Error)

ค่าคงที่	ค่าตัวเลข	รายละเอียด
ComEventBreak	1001	ได้รับสัญญาณการหยุด (Break Signal)
ComEventCTSTO	1002	สายสัญญาณ Clear To send อยู่ในสถานะ Low (Timeout) ในขณะที่จะพยายามส่งออกตัวอักษร
ComEventFrame	1003	สายสัญญาณ Data Set Ready จะอยู่ในสถานะ Low (Timeout) ในขณะที่พยายามจะส่งออกตัวอักษร
ComEventFrame	1004	เฟรมของข้อมูลไม่ถูกต้อง ซึ่งถูกตรวจพบโดยฮาร์ดแวร์
ComEventOverrun	1006	เกิด Port Overrun หมายถึงมีการรับตัวอักษรตัวใหม่เข้ามาในขณะที่ตัวอักษรก่อนหน้านี้ยังไม่ถูกอ่าน จากฮาร์ดแวร์ ดังนั้นจึงเกิดการสูญหายของตัวอักษร
ComEventCDTO	1007	สายสัญญาณ Carrier Detect อยู่ในสถานะ Low (Timeout) ในขณะที่พยายามจะส่งออกตัวอักษร
ComEventRxOver	1008	Receiver Buffer Overflow หมายถึงขนาดของบัฟเฟอร์ด้านรับเข้าข้อมูล (Receiver Buffer) ไม่เพียงพอ กับขนาดของข้อมูลที่รับเข้ามา
ComEventRxParity	1009	Parity Error ซึ่งถูกตรวจพบโดยฮาร์ดแวร์
ComEventTxFull	1010	Transmit Buffer full หมายถึงบัฟเฟอร์ด้านส่งออกข้อมูลเต็มในขณะที่พยายามจัดเก็บข้อมูลใหม่ลงในบัฟเฟอร์
ComEventDCB	1011	Unexpected Error หมายถึง ข้อผิดพลาดที่ไม่ได้ถูกนิยามเอาไว้ในขณะอ่าน Device Control Block (DCB) จากพอร์ทอนุกรม

เหตุการณ์ (Event)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ComEventSend	1	มีจำนวนตัวอักษรในบัฟเฟอร์ด้านส่งออกข้อมูลน้อยกว่าจำนวนตัวอักษรที่กำหนดในคุณสมบัติ threshold
ComEventReceive	2	การรับเข้าจำนวนตัวอักษรที่ถูกกำหนดในคุณสมบัติ threshold ซึ่งจะเกิดขึ้นอย่างต่อเนื่องจนกว่าผู้อ่านจะใช้คุณสมบัติอินพุทในการอ่านข้อมูลจากบัฟเฟอร์สำหรับรับเข้าข้อมูล
ComEbCTS	3	มีการเปลี่ยนสถานะของสายสัญญาณ Clear To Send
ComEvDSR	4	มีการเปลี่ยนสถานะของสายสัญญาณ Data Set Ready ซึ่งเหตุการณ์นี้จะเกิดขึ้นเมื่อมีการเปลี่ยนสถานะของสายสัญญาณ Data Set Ready จาก 1 เป็น 0 เท่านั้น
ComEvCD	5	มีการเปลี่ยนสถานะของสายสัญญาณ Carrier Detect
ComEvRing	6	มีการตรวจพบการเรียกหมายเลข (สัญญาณเสียงกริ่ง) ซึ่ง UART บางตัวอาจจะไม่สนับสนุนคุณสมบัตินี้
ComEvEOF	7	มีการรับตัวอักษรรหัสจุดสิ้นสุดของไฟล์ (EOF, ASCII26)

- CommPort

รายงานหรือกำหนดหมายเลขของพอร์ทอนุกรมของเครื่องคอมพิวเตอร์ที่ต้องการติดต่อ

รูปแบบการใช้งาน

Object.CommPort [=Value]

Value หมายถึงข้อมูลชนิดเลขจำนวนเต็ม ที่กำหนดหมายเลขของพอร์ทอนุกรม

สำหรับหมายเลขของ พอร์ทอนุกรมสามารถมีค่าได้ตั้งแต่ 1 ถึง 16 (ค่าปกติจะเท่ากับ 1) ซึ่งก่อนที่ผู้อ่านจะเปิดพอร์ทด้วยคุณสมบัติ PortOpen ผู้อ่านต้องกำหนดหมายเลขของพอร์ทอนุกรมให้กับคุณสมบัติ CommPort เสียก่อน โดยถ้าหากหมายเลขของพอร์ทอนุกรมที่กำหนดให้กับคุณสมบัติสามารถแก้ไขได้โดยการกำหนดหมายเลขของพอร์ทอนุกรมที่ถูกต้องเสียใหม่ แล้วจึงทำการเปิดพอร์ทอนุกรมอีกครั้งด้วยคุณสมบัติ PortOpen

- Handshaking

รายงาน หรือ กำหนดการใช้โปรโตคอลการตอบ รับการติดต่อสื่อสารฮาร์ดแวร์ (Hardware Handshaking Protocol)

รูปแบบการใช้งาน

Object.Handshaking [=Value]

Value หมายถึงข้อมูลชนิดเลขจำนวนเต็มที่กำหนดโปรโตคอลการตอบรับการติดต่อสื่อสารดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าคงที่	ค่าตัวเลข	รายละเอียด
comNone	0	ไม่มีการตอบรับการติดต่อสื่อสาร (Default)
comXOnXOff	1	การตอบรับการติดต่อสื่อสารแบบ XON/XOFF
comRTS	2	การตอบรับการติดต่อสื่อสารแบบ RTS/CTS (Request To Send/Clear To Send)
comRTSXOnXOff	3	การตอบรับการติดต่อสื่อสารแบบทั้ง Request To Send และ XON/XOFF

ในด้านการสื่อสารด้วยโมเด็ม Handshaking หมายถึงโปรโตคอลการสื่อสารภายในที่ซอฟต์แวร์ใช้ในการส่งข้อมูลจากพอร์ทอนุกรมของเครื่องคอมพิวเตอร์ไปยังบัฟเฟอร์ด้านรับเข้าข้อมูล โดยทุกครั้งที่มีการรับข้อมูลมายังพอร์ทอนุกรมของเครื่องคอมพิวเตอร์ มันก็จะถูกจัดส่งต่อมายังบัฟเฟอร์ด้านรับเข้าข้อมูลทันที ทั้งนี้เพื่อให้ซอฟต์แวร์สามารถอ่านข้อมูลดังกล่าวได้ต่อไป ซึ่งโปรโตคอลการสื่อสารภายในดังกล่าวจะเป็นการช่วยในการตรวจสอบ เพื่อป้องกันการสูญหายของข้อมูล เมื่อเกิดปัญหาขนาดของบัฟเฟอร์ไม่เพียงพอกับจำนวนข้อมูลที่ส่งมา (Buffer Overrun)

- Input

รายงานพร้อมทั้งทำการลบข้อมูลในบัฟเฟอร์ด้านรับเข้าทิ้ง ซึ่งผู้อ่านสามารถอ่านค่าได้เฉพาะในขณะทำงานเท่านั้น

รูปแบบการใช้งาน

object. Input

ทุกครั้งที่มีการใช้คุณสมบัติ Input ในการอ่านข้อมูลจากบัฟเฟอร์ด้านรับเข้านั้น จำนวนของตัวอักษรที่อ่านได้จะถูกกำหนดลงในคุณสมบัติ InputLen ทันที ซึ่งถ้าหากผู้อ่านกำหนดให้คุณสมบัติ InputLen มีค่าเท่ากับ 0 ก็จะหมายถึงการกำหนดให้คุณสมบัติ Input อ่านข้อมูลทั้งหมดจากบัฟเฟอร์ด้านรับเข้านั่นเอง ซึ่งชนิดของข้อมูลที่อ่านโดยคุณสมบัติ Input จะเป็นข้อมูลแบบข้อความหรือไบนารีก็ขึ้นกับการกำหนดค่าของคุณสมบัติ InputMode ดังจะกล่าวในรายละเอียดต่อไป

- InputLen

รายงานหรือกำหนดจำนวนของ ตัวอักษรที่อ่านโดยคุณสมบัติ Input จากบัฟเฟอร์ด้านรับเข้าซึ่งมีหน่วยเป็น ไบท์

รูปแบบการใช้งาน

oject.InputLen [=Value]

Value หมายถึงข้อมูลชนิดเลขจำนวนเต็มที่กำหนดจำนวนตัวอักษรที่อ่าน โดยคุณสมบัติอินพุท

โดยปกติ ผู้เขียนมักจะกำหนดให้คุณสมบัติ InputLen มีค่าเท่ากับ 0 เพื่อให้คุณสมบัติอินพุท มีการอ่านตัวอักษรทั้งหมดจากบัฟเฟอร์ด้านรับเข้าทั้งหมด แต่ถ้าหากผู้เขียนติดต่อกับบอร์ดควบคุมต่างๆ (Single Control Board) ก็จะกำหนดค่าของคุณสมบัติ InputLen ให้เท่ากับจำนวนของเฟรมข้อมูลที่จะมีการรับ-ส่ง ในแต่ละครั้งซึ่งจะมีค่าคงที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Output

ทำการส่งข้อมูลไปยังบัฟเฟอร์ด้านส่งออก ซึ่งผู้อ่านสามารถกำหนดค่าได้เฉพาะในขณะทำงานเท่านั้น

รูปแบบการทำงาน

object. Output [=Value]

Value หมายถึงข้อมูลชนิดสตริงหรือชนิด Variant ที่ต้องการส่งไปยังบัฟเฟอร์ด้านส่งออกสำหรับชนิดของข้อมูลที่ถูกส่งโดยคุณสมบัติเอาท์พุทจะเป็นข้อมูลแบบข้อความหรือไบนารี่ ก็ขึ้นกับการกำหนดค่าของคุณสมบัติ InputMode ดังที่กล่าวมาแล้วข้างต้น

- PortOpen

กำหนดสถานะการเปิด (Open) หรือปิด (Close) ของพอร์ทอนุกรมของเครื่องคอมพิวเตอร์

รูปแบบการใช้งาน

object. PortOpen [=Boolean]

Boolean หมายถึง ข้อมูลชนิดบูลีนที่กำหนดสถานะของพอร์ทอนุกรมดังต่อไปนี้

True หมายถึง พอร์ทอนุกรมถูกเปิด

False หมายถึง พอร์ทอนุกรมถูกปิด

พอร์ทอนุกรมถูกปิดอัตโนมัติโดยคอนโทรล MSComm เมื่อแอปพลิเคชันสิ้นสุดการทำงาน โดยถ้าหากหมายเลขของพอร์ทอนุกรมที่กำหนดให้เปิดไม่มีการติดตั้งอยู่จริง ก็จะเกิดข้อผิดพลาดหมายเลข 68 (Device Unavailable) ทันที

- Rthreshold

รายงานหรือกำหนดจำนวนตัวอักษรที่จะรับเข้าก่อนที่จะคอนโทรล MSComm จะกำหนดให้คุณสมบัติ CommEvent มีค่าเท่ากับ comEvReceive และมีการเรียกโพรซีเยอร์เหตุการณ์ OnComm

รูปแบบการใช้งาน

Object. Rthreshold [=Value]

Value หมายถึง ข้อมูลชนิดเลขจำนวนเต็ม ที่กำหนดจำนวนตัวอักษรที่จะรับเข้าก่อนที่จะคอนโทรล MSComm ตามรายละเอียดข้างต้น

ถ้าหากคุณสมบัติ Rthreshold มีค่าเท่ากับ 0 (Default) ก็จะเป็นการยกเลิกการเรียกโพรซีเยอร์เหตุการณ์ OnComm เมื่อมีการรับเข้าตัวอักษรมายังบัฟเฟอร์ด้านรับเข้า และในทางกลับกันถ้าหากคุณสมบัติ Rthreshold มีค่าเท่ากับ 1 ก็มีการเรียกโพรซีเยอร์เหตุการณ์ OnComm ทุกครั้งที่มีการรับเข้าตัวอักษรมายังบัฟเฟอร์ด้านรับเข้า

- RTSEnable

กำหนดให้มีการใช้งานสาย Request To Send (RTS) ซึ่งโดยปกติสัญญาณ Request To Send จะถูกส่งจากเครื่องคอมพิวเตอร์ไปยังโมเด็ม เพื่อเป็นการแจ้งขอส่งข้อมูลจากคอมพิวเตอร์

รูปแบบการใช้งาน

Object. RTSEnable [=Boolean]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Boolean หมายถึงข้อมูลชนิดบูลีนที่กำหนดให้มีการใช้งานสาย Request To Send ดังต่อไปนี้

True หมายถึง มีการใช้งานสายสัญญาณ RTS

False หมายถึง ไม่มีการใช้งานสายสัญญาณ RTS

ถ้าหากคุณสมบัติ RTSEnable มีค่าเท่ากับ True สายสัญญาณ RTS จะอยู่ในสถานะ High เมื่อพอร์ทอนุกรมถูกเปิดและอยู่สถานะ Low เมื่อพอร์ทอนุกรมถูกปิด โดยปกติสายสัญญาณ RTS จะถูกใช้ในการตอบรับการติดต่อสื่อสารแบบฮาร์ดแวร์ RTS/CTS เท่านั้น

- Setting

รายงานหรือกำหนดพารามิเตอร์ในการสื่อสารผ่านพอร์ทอนุกรม

รูปแบบการใช้งาน

Object. Setting [=Value]

Value หมายถึงข้อมูลชนิดสตริงที่กำหนดพารามิเตอร์ในการสื่อสารผ่านพอร์ทอนุกรมดังต่อไปนี้

รูปแบบของการกำหนดลำดับของพารามิเตอร์ในการสื่อสารผ่านพอร์ทอนุกรม สำหรับคุณสมบัติ Setting จะต้องเรียงลำดับดังนี้ “BBBB, P, D, S” (โดยปกติทั่ว ๆ ไปจะมีค่าเท่ากับ “9600, N, 8, 1” เพราะถ้าหากลำดับไม่ถูกต้องก็จะทำให้เกิดข้อผิดพลาดหมายเลข 380 (Invalid Property Value) ทันทีซึ่งสัญลักษณ์แต่ละตัวมีความหมายดังนี้

- BBBB

ความเร็วของการส่งถ่ายข้อมูลในหน่วยของ Baud Rate ซึ่งในทางปฏิบัติ 1 Baud Rate อาจจะมีค่าเท่ากับ 1 BPS (Bits Per Second) หรือมากกว่าก็ได้ สำหรับค่าของ Baud Rate ที่คอนโทรล MSComm สามารถรับได้มีค่าดังต่อไปนี้ 110, 300, 600, 1200, 2400, 9600 (Default), 14400, 19200, 28800, 38400 (Reserved), 56000 (Reserved), 128000, 256000 (Reserved)

- P

บิตพาริตี้ (Parity Bit) สำหรับใช้ในการตรวจสอบความถูกต้องของข้อมูล ซึ่งสามารถมีค่าดังต่อไปนี้

พารามิเตอร์	ความหมาย
E	Even
M	Mark
N	None (Default)
O	Odd
S	Space

- D

ขนาดของบิตข้อมูล ซึ่งสามารถมีค่าได้ดังต่อไปนี้ 4, 5, 6, 7 หรือ 8 (Default)

- S

ขนาดของบิตหยุด (Stop Bit) ซึ่งสามารถมีค่าได้ดังต่อไปนี้ 1 (Default), 1.5 หรือ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Sthreshold

รายงานหรือการกำหนดจำนวนตัวอักษร ที่น้อยที่สุด ที่ถูกจัดเก็บในไว้ในบัฟเฟอร์ด้านส่งออกไป ก่อนที่คอนโทรล MSComm จะกำหนดให้คุณสมบัติ CommEvent มีค่าเท่ากับ comEvSend และมีการเรียกเหตุการณ์ OnComm

รูปแบบการใช้งาน

Object. Sthreshold [= Value]

Value หมายถึงข้อมูลชนิดตัวเลขจำนวนเต็มที่กำหนดจำนวนตัวอักษรที่น้อยที่สุดที่ถูกจัดเก็บในบัฟเฟอร์ด้านส่งออกตามรายละเอียดข้างต้น

ถ้าหากคุณสมบัติ Sthreshold มีค่าเท่ากับ 0 (Default) ก็จะเป็นการยกเลิกการเรียกโพรซีเยอร์เหตุการณ์ OnComm เมื่อมีการส่งออกตัวอักษรไปยังบัฟเฟอร์ด้านส่งออก และในทางกลับกันถ้าหากคุณสมบัติ Sthreshold มีค่าเท่ากับ 1 ก็จะมีการเรียกโพรซีเยอร์เหตุการณ์ OnComm เมื่อบัฟเฟอร์ด้านส่งออกว่าง

โดยถ้าหากจำนวนของตัวอักษรในบัฟเฟอร์ด้านส่งออกน้อยกว่าค่าตัวเลขที่กำหนด คุณสมบัติ CommEvent ก็จะมีค่าเท่ากับ comEvSend และพร้อมทั้งเกิดการเรียกโพรซีเยอร์เหตุการณ์ OnComm ทันที เช่น สมมติให้คุณสมบัติ Sthreshold มีค่าเท่ากับ 10 และถ้าหากจำนวนตัวอักษรในบัฟเฟอร์ด้านส่งออกลดลงจาก 10 เป็น 9 ก็จะเกิดเหตุการณ์ OnComm ทันที เป็นต้น

โพรซีเยอร์เหตุการณ์ OnComm

- OnComm

จะเกิดขึ้นเมื่อมีการเปลี่ยนแปลงค่าของคุณสมบัติ CommEvent ซึ่งเป็นการบอกถึงการเกิดข้อผิดพลาดหรือมีการสื่อสารเกิดขึ้นก็ได้

รูปแบบโพรซีเยอร์เหตุการณ์

Private Sub object_OnComm()

โดยปกติเมื่อคอนโทรล MSComm มีการเรียกโพรซีเยอร์เหตุการณ์ OnComm เรามักจะมีการเขียนโค้ดภายในโพรซีเยอร์เหตุการณ์นี้ เพื่อทำการตรวจสอบค่าของคุณสมบัติ CommEvent ทั้งนี้เพื่อตรวจสอบสถานะของการสื่อสารหรือข้อผิดพลาดที่เกิดขึ้นนั่นเอง

2.12 การจัดการโครงสร้างฐานข้อมูล

2.12.1 อ็อปเจ็คทางด้านฐานข้อมูล (Database Related Objects)

ในการทำงานหรือจัดการกับฐานข้อมูลโดยใช้อ็อปเจ็คทางด้านฐานข้อมูลของวิชวลเบสิกมีอ็อปเจ็คต่าง ๆที่เกี่ยวข้องที่ใช้หมายถึงองค์ประกอบต่าง ๆ ภายในฐานข้อมูล ซึ่งจะอธิบายในรายละเอียดเรียงตามลำดับของโครงสร้างฐานข้อมูลได้ดังนี้

1. อ็อปเจ็คระบบจัดการฐานข้อมูล (DBEngine Object) คือ อ็อปเจ็คที่หมายถึงระบบจัดการฐานข้อมูลวิชวลเบสิก หรือที่เรียกว่า JET ซึ่งจะมีเพียงอ็อปเจ็คเดียวเท่านั้น อ็อปเจ็คทางด้านฐานข้อมูลอื่น ๆ จะถูกกำหนดควบคุมผ่านทาง DBEngine ทั้งหมด โดยจะมีคุณลักษณะ (Property) เช่น เวอร์ชันของ JET (Version) ชื่อผู้ใช้ที่จะใช้ Login เข้าระบบจัดการฐานข้อมูล (Default User) เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในอ็อปเจ็คระบบจัดการฐานข้อมูล จะเก็บรายละเอียดหรือประกอบด้วยอ็อปเจ็คพื้นที่ทำงานกับระบบจัดการฐานข้อมูล (Workspace Object) ซึ่งถ้าหากมี Workspace Object มากกว่าหนึ่งอ็อปเจ็คก็หมายความว่าโปรแกรมนั้น มีการติดต่อกับระบบจัดการฐานข้อมูล (JET) มากกว่าหนึ่งการเชื่อมต่อ (Connection)

2. อ็อปเจ็คพื้นที่ทำงานกับระบบจัดการฐานข้อมูล (Workspace Object) โปรแกรมของวิชวลเบสิกที่ต้องการทำงานกับฐานข้อมูล จะต้องทำงานโดยการเรียกใช้บริการจากอ็อปเจ็คระบบจัดการฐานข้อมูล (DBEngine Object) ดังนั้นทุก ๆ โปรแกรมดังกล่าวจะต้องมีการติดต่อกับระบบจัดการฐานข้อมูลซึ่งในที่นี้วิชวลเบสิกจะเรียกว่าเป็นอ็อปเจ็คพื้นที่ทำงานกับระบบจัดการฐานข้อมูล หรือ (Workspace Object) อ็อปเจ็คนี้ ทำหน้าที่เป็นตัวกลาง เชื่อมต่อโปรแกรมวิชวลเบสิกกับอ็อปเจ็คระบบจัดการฐานข้อมูลทุกโปรแกรม ที่ทำงานกับฐานข้อมูลจะต้องมี อ็อปเจ็คพื้นที่ทำงานกับระบบจัดการฐานข้อมูล อย่างน้อยหนึ่งอ็อปเจ็คหรือมากกว่าหนึ่งอ็อปเจ็คก็ได้ ในแต่ละอ็อปเจ็คชนิดนี้จะประกอบด้วยอ็อปเจ็คฐานข้อมูลที่โปรแกรมทำงานด้วยผ่านทางอ็อปเจ็คพื้นที่การทำงานกับระบบจัดการฐานข้อมูลนั้น ๆ

3. อ็อปเจ็คฐานข้อมูล (Database Object) คืออ็อปเจ็คที่หมายถึงตัวฐานข้อมูลเอง โดยจะมีคุณลักษณะ เช่นฐานข้อมูลชื่ออะไร อนุญาตให้แก้ไขข้อมูลได้หรือไม่ วิธีการเรียงลำดับข้อมูลเป็นอย่างไร เป็นต้น ทุกโปรแกรมที่ทำงานกับฐานข้อมูลจะต้องมีการกำหนดอ็อปเจ็คฐานข้อมูล อย่างน้อยเท่ากับจำนวนฐานข้อมูลที่ใช้ในโปรแกรมนั้น

ในแต่ละอ็อปเจ็คฐานข้อมูลจะประกอบด้วยอ็อปเจ็คทางด้านโครงสร้างอื่น ๆ ที่ประกอบกันเป็นฐานข้อมูลนั้นเช่น อ็อปเจ็คข้อกำหนดตาราง (TebleDef Object) อ็อปเจ็คข้อกำหนดความสัมพันธ์ (Relation Object) และอ็อปเจ็คโปรแกรมย่อยในฐานข้อมูล (ซึ่งในวิชวลเบสิกเรียกว่า QueryDef) เป็นต้น

4. อ็อปเจ็คข้อกำหนดตาราง คือ อ็อปเจ็คที่หมายถึงโครงสร้างของตารางนั้น (ไม่ได้หมายถึงข้อมูลที่เก็บในตาราง) เช่น ประกอบด้วยฟิลด์อะไรบ้าง หรือมีดรรชนีอะไรบ้าง เป็นต้น ในแต่ละอ็อปเจ็คข้อกำหนดตารางจะประกอบด้วยอ็อปเจ็คที่แสดงถึงองค์ประกอบของตารางนั้น ๆ เช่น อ็อปเจ็คข้อกำหนดฟิลด์ (Field Object) อ็อปเจ็คข้อกำหนดดรรชนี (Index Object)

5. อ็อปเจ็คข้อกำหนดฟิลด์ อ็อปเจ็คนี้เป็นองค์ประกอบหนึ่งของอ็อปเจ็คข้อกำหนดตาราง เพื่อใช้อธิบายถึงคุณลักษณะของฟิลด์ที่ประกอบกันเป็นตารางนั้น

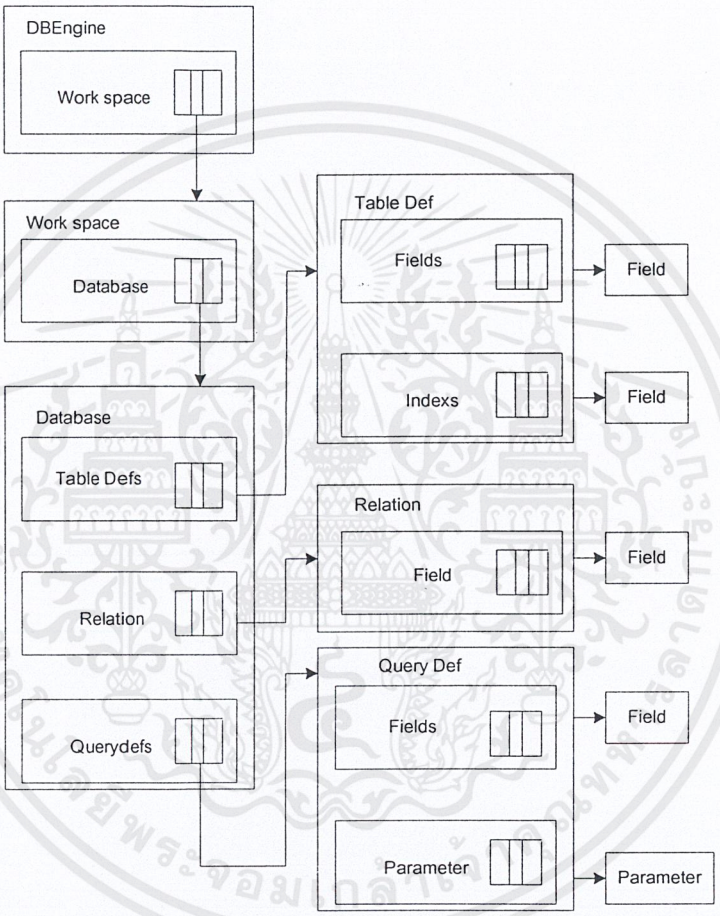
6. อ็อปเจ็คข้อกำหนดดรรชนีตัวอ็อปเจ็คข้อกำหนดดรรชนีก็เช่นเดียวกันกับอ็อปเจ็คข้อกำหนดฟิลด์ คือจะช่วยในการอธิบายให้ทราบว่าตารางนั้นมีดรรชนีในลักษณะใดประกอบอยู่ด้วย

7. อ็อปเจ็คข้อกำหนดความสัมพันธ์ (Relation Object) ฐานข้อมูล โดยทั่วไปแล้วมักจะมีความสัมพันธ์ระหว่างตารางเป็นอีกองค์ประกอบหนึ่งที่สำคัญ อ็อปเจ็คข้อกำหนดความสัมพันธ์นี้จะเป็นส่วนที่ใช้อธิบายความสัมพันธ์ต่าง ๆ ที่มีอยู่ในฐานข้อมูลนั้น ๆ จัดเป็นองค์ประกอบย่อยภายในอ็อปเจ็คฐานข้อมูลเช่นเดียวกับอ็อปเจ็คข้อกำหนดตาราง

8. อ็อปเจ็คโปรแกรมย่อยในฐานข้อมูล (QueryDef Object) โปรแกรมย่อยในฐานข้อมูลหรือที่เรียกว่า Store Procedure ในระบบจัดการฐานข้อมูลโดยทั่วไป แต่สำหรับระบบจัดการฐานข้อมูลเจ็ต (JET) ที่ใช้ในวิชวลเบสิกและ MsAccess จะเรียกว่า QueryDef เป็นชุดคำสั่งทำงานจัดการฐานข้อมูลที่ประกอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ด้วยคำสั่ง SQL ต่าง ๆ โดยจะถูกสร้างและจัดเก็บรวบรวมไว้กับฐานข้อมูล ประโยชน์การใช้โปรแกรมย่อยในฐานข้อมูลจะช่วยในเรื่องของความเร็ว ลดความซับซ้อนของโปรแกรมและช่วยให้โปรแกรมเป็นอิสระจากโครงสร้างของฐานข้อมูลมากขึ้น



รูปที่ 2.32 อ็อปเจ็คโปรแกรมย่อยในฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.12.2 โครงสร้างฐานข้อมูลแบบเจ็ต (JET)

การตรวจสอบถึงโครงสร้างและ องค์ประกอบต่าง ๆ ในฐานข้อมูล จะทำได้โดยการท่องไปในอ็อปเจ็คต่าง ๆ ที่เกี่ยวข้องกับฐานข้อมูล แล้วตรวจสอบค่าของคุณสมบัติในอ็อปเจ็คต่าง ๆ นั้นนั่นเอง โดยมีขั้นตอนต่าง ๆ ดังนี้

1. กำหนดวิธีการติดต่อกับระบบจัดการฐานข้อมูลและการเปิดฐานข้อมูล

การทำงานกับฐานข้อมูลทุก ๆ อย่างในวิชวลเบสิก จะถูกจัดการโดยระบบจัดการฐานข้อมูลหรือ JET ดังนั้นโปรแกรมที่ต้องการทำงานกับฐานข้อมูลจะต้องกำหนดรายละเอียดและวิธีการติดต่อกับระบบจัดการฐานข้อมูลเสียก่อน ซึ่งทำได้โดยการสร้างอ็อปเจ็คพื้นที่ทำงานกับระบบจัดการฐานข้อมูล (Workspace Object) โดยปกติแล้ววิชวลเบสิกจะสร้างอ็อปเจ็คพื้นที่ทำงานกับระบบจัดการฐานข้อมูลขึ้นมาให้โดยอัตโนมัติแล้วหนึ่งอ็อปเจ็ค ที่เป็นมาตรฐานสำหรับการทำงานกับฐานข้อมูลทั่วไป เราสามารถเรียกอ็อปเจ็คพื้นที่ทำงานกับระบบจัดการฐานข้อมูลซึ่งเป็นมาตรฐานนี้ขึ้นมาใช้ได้เลย โดยจะอ้างถึงอ็อป-เจ็คลำดับแรก ในอ็อปเจ็คกลุ่มของอ็อปเจ็คพื้นที่ทำงานกับระบบจัดการฐานข้อมูล ดังในตัวอย่างข้างล่างนี้

```
Dim wksDefault  AS  Workspace
Set  wlsDefault  = DBEngine.Workspaces (0)
```

การเปิด (open) ฐานข้อมูลที่ต้องการ หลักจากที่ได้อ็อปเจ็คพื้นที่ทำงานกับระบบจัดการฐานข้อ-มูลแล้ว ก่อนจะสามารถอ่านโครงสร้างหรือจัดการใด ๆ กับข้อมูลที่อยู่ภายในฐานข้อมูลนั้นได้ ฐานข้อมูลที่ต้องการนั้นจะต้องถูกเปิดก่อน ในลักษณะเช่นเดียวกับการเปิดแฟ้มข้อมูลทั่วไปก่อนที่จะทำการอ่านหรือเขียนแฟ้มข้อมูลนั้น

การเปิดฐานข้อมูลขึ้นมาเพื่อใช้งานสามารถทำได้โดยการเรียกฟังก์ชันการทำงาน open database จากอ็อปเจ็คพื้นที่ทำงานกับฐานข้อมูล แล้วกำหนดค่าที่ได้ให้กับตัวแปรอ็อปเจ็คฐานข้อมูลโดยมีรูปของการเรียกใช้ดังนี้

```
Set db_object=wks_object.OpenDatabase(db_filename[,exclusive[,readonly[, connect]]])
```

- ob_ject ตัวแปรประเภทอ็อปเจ็คฐานข้อมูล
- db_filename ตัวแปรหรือค่าคงที่ประเภทตัวอักษร ที่แสดงชื่อแฟ้มข้อมูลของฐานข้อมูล ซึ่งการระบุชื่อแฟ้มข้อมูลทำในลักษณะเช่นเดียวกับการระบุถึงแฟ้มข้อมูลทั่วไป จะมีหรือไม่ส่วนที่บอกว่าอยู่ที่ดิสก์ไดร์ฟหรือไดเรคทอรีก็ได้
- exclusive ค่าของตัวแปรหรือค่าคงที่มีค่าเป็น True หรือ False เท่านั้น คือ ถ้ามีค่าเป็น True หมายความว่าต้องการจะใช้ฐานข้อมูลนี้เพียงคนเดียว แต่หากมีค่าเป็น False หรือละไว้ไม่กำหนด จะเป็นการใช้ฐานข้อมูลร่วมกับผู้อื่น
- readyonly ค่าของตัวแปรหรือค่าคงที่ที่มีค่าเป็น True หรือ False เท่านั้น เพื่อบอกว่าต้องการจะใช้ฐานข้อมูลนี้ในลักษณะอย่างไร จะเป็นการอ่านข้อมูลอย่างเดียว หรือว่าต้องการแก้ไขข้อมูลด้วย ซึ่งถ้าเป็น False หรือไม่ได้ระบุ จะหมายถึงว่าต้องการจะใช้ฐานข้อมูลในลักษณะที่มีการแก้ไขข้อมูล
- connect เป็นตัวแปรหรือค่าคงที่ประเภทตัวอักษร จะใช้เมื่อต้องการจะเปิดฐานข้อมูลภาย-นอกอื่น ๆ (External Database) ที่ไม่ใช่ Ms-Access เช่น SQL-Server หรือ Oracle เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.12.3 อ็อปเจ็คข้อมูล (Recordset) และการจัดการกับข้อมูล

การใช้วิชวลเบสิกจัดการกับข้อมูลในฐานข้อมูลทำได้ 2 วิธีคือ

วิธีที่ 1 ทำโดยการใช้คอนโทรลข้อมูล (Data Control) ซึ่งวิธีนี้มีข้อดีในเรื่องของความสะดวกและง่ายต่อการใช้งานเป็นอย่างมาก ช่วยให้สร้างโปรแกรมขึ้นมาได้อย่างรวดเร็ว แต่ก็มีข้อเสียคือ ความยืดหยุ่นในการใช้งาน และวิธีการควบคุมการใช้งานทำได้ไม่ค่อยสะดวกนัก ไม่สามารถเรียกใช้ประโยชน์จากโครงสร้างของฐานข้อมูลได้มากนัก ทำให้การจัดการหรือค้นหาข้อมูลไม่มีประสิทธิภาพและยืดหยุ่นอย่างเพียงพอ

วิธีที่ 2 เป็นการจัดการกับข้อมูลโดยอาศัย อ็อปเจ็คข้อมูล ด้วยวิธีการนี้ผู้พัฒนาจะจัดการข้อมูลได้อย่างมีประสิทธิภาพ และมีความยืดหยุ่นต่อการใช้งานในลักษณะต่าง ๆ มากกว่าการใช้คอนโทรลข้อมูล สำหรับวิธีการนี้จะมีเฉพาะในวิชวลเบสิกชุดโปรเฟสชั่นนัลและเอ็นเทอร์ไพรซ์เท่านั้น จะไม่มีในเวอร์ชันมาตรฐาน

2.12.4 อ็อปเจ็คข้อมูล (Recordset Object)

อ็อปเจ็คข้อมูล หมายถึง กลุ่มของข้อมูลในฐานข้อมูลที่สนใจหรือต้องการจะใช้งานอ็อปเจ็คข้อมูล จะแบ่งออกเป็นประเภทที่แตกต่างกันได้ 3 ประเภทหลัก ๆ แต่ละประเภทก็จะเหมาะสมกับการใช้งานในลักษณะที่แตกต่างกัน ซึ่งพอจะสรุปได้ดังนี้

1. อ็อปเจ็คข้อมูลประเภทตาราง (Table Type Recordset Object) อ็อปเจ็คข้อมูลประเภทตารางนี้ จะเก็บข้อมูลทั้งหมดทุก ๆ เรคคอร์ด (Record) และทุก ๆ ฟิลด์ (Field) ของตารางที่สนใจการกำหนดข้อมูลให้กับอ็อปเจ็คข้อมูลประเภทตารางนี้ ทำได้โดยการระบุชื่อตารางที่ต้องการในขณะที่มีการกำหนดสร้างอ็อปเจ็คขึ้นมา

2. อ็อปเจ็คข้อมูลประเภทไดนาเซ็ต (Dynaset Type Recordset Object) เป็นอ็อปเจ็คข้อมูลที่อาจมีข้อมูลทั้งหมดจากตาราง ข้อมูลบางส่วนของตาราง หรือข้อมูลจากตารางหลายตารางรวมกันก็ได้และนำเสนอให้ผู้ใช้เข้าใจเสมือนว่าเป็นอีกตารางหนึ่ง คำว่า ไดนาเซ็ต (Dynaset) ในวิชวลเบสิกจะมีลักษณะเดียวกันกับ View ที่พบในระบบจัดการฐานข้อมูล ขนาดใหญ่ทั่วไป วิธีการกำหนดข้อมูลให้กับอ็อปเจ็คข้อมูลประเภทนี้ ทำได้โดยการใช้คำสั่ง Select ของ SQL ในขณะที่มีการกำหนดโครงสร้างอ็อปเจ็คนี้ขึ้นมา เพื่อเลือกเอาข้อมูลที่มีลักษณะที่ต้องการจากตาราง (Table) ตารางหนึ่ง หรือเลือกข้อมูลจากหลายตารางมารวมกัน (จากการเชื่อม (Join) ตารางต่าง ๆ เข้าด้วยกัน)

3. อ็อปเจ็คข้อมูลประเภทสแน็ปช็อต (Snapshot Type Recordset) สแน็ปช็อตจะมีลักษณะเหมือนกับอ็อปเจ็คข้อมูลประเภทไดนาเซ็ตทุกประการทั้งในเรื่องของการสร้างและการนำไปใช้งานยกเว้นแต่เพียงว่าสามารถอ่านหรือค้นหาข้อมูลที่อยู่ในอ็อปเจ็คข้อมูลประเภทสแน็ปช็อตได้เพียงอย่างเดียวเท่านั้น ไม่สามารถเพิ่มเติม ลบ หรือแก้ไขได้ และข้อมูลในอ็อปเจ็คประเภทนี้เมื่อมีการกำหนดสร้างขึ้นมาแล้วจะไม่มีการเปลี่ยนแปลง ไม่ว่าข้อมูลในฐานข้อมูลได้ถูกเปลี่ยนแปลงโดยโปรแกรมหรือผู้ใช้รายอื่น ๆ แล้วก็ตาม สำหรับวิธีการกำหนดสร้างก็ทำได้ในลักษณะเดียวกันกับการกำหนดสร้างอ็อปเจ็คข้อมูลประเภทไดนาเซ็ต เพียงแต่ระบุประเภทที่แตกต่างกันเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนและวิธีการต่าง ๆ ที่จะใช้งานอ็อปเจ็คข้อมูลมีดังนี้

1. กำหนดตัวแปร

- อ็อปเจ็คข้อมูล จัดประเภทของตัวแปร (Variable) หนึ่ง มีวิธีการกำหนดและการอ้างถึงเหมือนกับตัวแปรชนิดอื่น ๆ ดังในตัวอย่างข้างล่างนี้

Dim recordset_object As Recordset

Recordset_object เป็นชื่อของตัวแปรที่ต้องการกำหนดให้มีประเภทเป็นอ็อปเจ็คข้อมูล

2. กำหนดค่าให้กับตัวแปรอ็อปเจ็คข้อมูล

ก่อนที่จะใช้ข้อมูลที่อยู่ในอ็อปเจ็คข้อมูลได้ จะต้องมีการกำหนดค่าให้กับตัวแปรอ็อปเจ็คข้อมูลนั้นเสียก่อน เพื่อเป็นการบอกว่าอ็อปเจ็คข้อมูลที่ต้องการ จะประกอบไปด้วยข้อมูลหรือเรคคอร์ดใดบ้าง และเป็นอ็อปเจ็คข้อมูลประเภทใด เป็นต้น

สำหรับวิธีการกำหนดค่าให้กับตัวแปรอ็อปเจ็คข้อมูล ทำโดยการเรียกใช้ฟังก์ชันการทำงาน (Method) ชื่อ OpenRecordset บนตัวแปรอ็อปเจ็คฐานข้อมูลที่ได้เปิดไว้แล้ว โดยมีรูปแบบและวิธีการเรียกใช้ ดังต่อไปนี้

Set recordset_object = db_object. OpenRecordset (source[,type[,options[,lockedits]]])

recordset_object เป็นตัวแปรประเภทอ็อปเจ็คข้อมูลที่ต้องการจะกำหนดค่าให้

db_object เป็นตัวแปรประเภทอ็อปเจ็คฐานข้อมูลที่ได้ถูกกำหนดค่าหรือเปิดไว้แล้ว

source เป็นตัวแปรหรือค่าคงที่ที่เป็นตัวอักษร เพื่อใช้ในการระบุถึงแหล่งของข้อมูลสำหรับกำหนดให้อ็อปเจ็คข้อมูลนี้ที่มีค่าเป็นได้ 3 กรณีคือ ดังนี้

1. เป็นชื่อของตาราง ในกรณีที่ต้องการจะให้อ็อปเจ็คข้อมูลนี้ที่จะกำหนดค่านี้หมายถึงข้อมูลทั้งหมดในตารางนั้น ๆ
2. เป็นคำสั่ง Select ของ SQL ในกรณีที่ต้องการจะให้อ็อปเจ็คข้อมูลนี้มีข้อมูลตามผลลัพธ์ที่ได้จากคำสั่ง Select นี้
3. เป็นชื่อของโปรแกรมย่อยในฐานข้อมูล (QueryDef) ในฐานข้อมูลที่มีการกำหนดคำสั่ง Select เก็บไว้ในฐานข้อมูลแล้ว

Type เป็นค่าคงที่ที่ใช้ในการระบุถึงประเภทของอ็อปเจ็คข้อมูลที่ต้องการจะกำหนดขึ้น มีรายละเอียดดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าคงที่	ความหมาย
DbOpenTable	ระบุเมื่อต้องการจะกำหนดสร้างอ็อปเจ็คข้อมูลประเภทตาราง
DbOpenDynaset	ระบุเมื่อต้องการจะกำหนดสร้างอ็อปเจ็คข้อมูลประเภทไดนาเซ็ต
DbOpenSnapshort	ระบุเมื่อต้องการจะกำหนดสร้างอ็อปเจ็คข้อมูลประเภทสแน็ปช็อต
DbOpenDynamic	สำหรับประเภทนี้จะมีลักษณะเช่นเดียวกับอ็อปเจ็คข้อมูลประเภทไดนาเซ็ต แต่แตกต่างตรงที่จำนวนของข้อมูลที่อยู่ในอ็อปเจ็คนี้จะมีการเปลี่ยนแปลงได้เสมอ จะใช้ได้กับอ็อปเจ็คพื้นที่ทำงานกับระบบจัดการฐานข้อมูลโดยตรง (ODBC Direct Workspace) เท่านั้น
DbOpenForwardOnly	อ็อปเจ็คข้อมูลประเภทนี้ จะมีลักษณะเช่นเดียวกับอ็อปเจ็คข้อมูลประเภทสแน็ปช็อต ต่างกันตรงที่อ็อปเจ็คประเภทนี้จะอ่านเรคคอร์ดถัดไปได้เท่านั้น ไม่สามารถจะอ่านข้อมูลในเรคคอร์ดก่อนหน้าเรคคอร์ดปัจจุบันได้

ตารางที่ 2.5 แสดงค่าคงที่ใช้ในการระบุถึงประเภทอ็อปเจ็คข้อมูล

2.12.5 การเคลื่อนตำแหน่งตัวชี้ตำแหน่งข้อมูล (Navigation)

การควบคุมตำแหน่งของตัวชี้ตำแหน่งข้อมูล ทำโดยการเรียกใช้ฟังก์ชันการทำงาน (Method) ของอ็อปเจ็คข้อมูลที่เกี่ยวข้องกับการเลื่อนตำแหน่งตัวชี้ตำแหน่งข้อมูล พร้อมทั้งทำการตรวจสอบค่าคุณ-สมบัติบางตัว ที่เกี่ยวข้องคุณสมบัติของอ็อปเจ็ค ข้อมูลที่เกี่ยวข้องกับตัวชี้ตำแหน่งข้อมูลคุณสมบัติของอ็อปเจ็คข้อมูลที่สำคัญและเกี่ยวข้องกับตัวชี้ตำแหน่งข้อมูล มีเพียง 2 คุณสมบัติที่สำคัญ คือ

1. BOF เป็นค่าของคุณสมบัติ ซึ่งถ้ามีค่าเป็น True หมายความว่า ตอนนี้ตัวชี้ตำแหน่งของข้อมูลของอ็อปเจ็คนี้ อยู่ก่อนหน้าเรคคอร์ดแรกของอ็อปเจ็คข้อมูลนี้ (Begin of file) หมายความว่าไม่ได้ชี้อยู่ที่ตำแหน่งของข้อมูลที่ถูกต้อง

2. EOF มีลักษณะเช่นเดียวกันกับ BOF หากมีค่าเป็น True หมายความว่า ตำแหน่งของตัวชี้ข้อมูลจะอยู่เลยถัดไปจากเรคคอร์ดสุดท้ายในอ็อปเจ็คข้อมูลนี้อยู่แล้ว (End of file)

ฟังก์ชันการทำงานสำหรับการเคลื่อนตำแหน่งตัวชี้ตำแหน่งข้อมูล มีวิธีการเคลื่อนตำแหน่งตัวชี้ข้อมูล มีอยู่ 4 ฟังก์ชัน ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชัน	คำอธิบายการทำงาน	คุณลักษณะที่เกี่ยวข้อง
MoveFirst	เป็นการเลื่อนตัวชี้ตำแหน่งในอ็อปเจ็คข้อมูลนั้นให้ไปชี้ที่ตำแหน่งของเรคคอร์ดแรก	EOF, BOF
MoveNext	เพื่อเลื่อนตัวชี้ตำแหน่งไปยังเรคคอร์ดถัดไปจากตำแหน่งปัจจุบัน	จะมีค่าเป็น False ทั้งคู่
MovePrevious	เพื่อเลื่อนตัวชี้ตำแหน่งไปยังเรคคอร์ดก่อนหน้าตำแหน่งปัจจุบัน	EOF: หากเรคคอร์ดปัจจุบันก่อนการทำงานเป็นเรคคอร์ดสุดท้ายในอ็อปเจ็คข้อมูล ค่าของ EOF นี้จะเป็น True EOF: หากเป็นการเลื่อนถอยหลังไปจากเรคคอร์ดแรกสุด BOF จะเป็น True
MoveLast	เพื่อเลื่อนตัวชี้ตำแหน่งเรคคอร์ดสุดท้ายในอ็อป-เจ็คข้อมูลนั้น	EOF, BOF จะมีค่าเป็น False ทั้งคู่
Move N	เลื่อนไปยังตำแหน่งเรคคอร์ดถัดไปจากตำแหน่งปัจจุบันนี้ไปอีก N เรคคอร์ด หาก N เป็นบวก หมายถึงจะเลื่อนไปทางตำแหน่งสุดท้าย ถ้า N เป็นลบ จะเลื่อนไปทางตำแหน่งแรก	EOF และ BOF จะมีค่าในลักษณะเดียวกับการ MoveNext และ MovePrevious แล้วแต่ค่าของ N

ตารางที่ 2.6 วิธีการเลื่อนตำแหน่งตัวชี้ข้อมูล

วิธีการเรียกใช้ฟังก์ชันการทำงานเคลื่อนตัวชี้ตำแหน่งข้อมูล มีรูปแบบและตัวอย่าง ดังนี้

recordset_object. MoveFirst หรือ

recordset_object. MoveNext หรือ

recordset_object. MovePrevious หรือ

recordset_object. MoveLast หรือ

recordset_object. Move N

หมายเหตุ : หากมีการเรียกฟังก์ชันการทำงาน MoveNext ในขณะที่คุณสมบัติ EOF มีค่าเป็น True อยู่แล้ว (ตำแหน่งตัวชี้ข้อมูลอยู่ถัดจากเรคคอร์ดสุดท้ายในอ็อปเจ็คข้อมูลแล้ว) จะเกิดข้อผิดพลาดในขณะทำงานขึ้น (Run-Time Error) ทำนองเดียวกันหากมีการเรียกฟังก์ชันการทำงาน MovePrevious ในขณะที่คุณสมบัติ BOF มีค่าเป็น True ก็จะเกิดข้อมูลผิดพลาดขณะทำงานได้ เช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.12.6 การอ่านและกำหนดค่าของฟิลด์ในเรคคอร์ด

เมื่อระบุตำแหน่งของเรคคอร์ดที่ต้องการได้แล้วจะสามารถทำการอ่านค่าของฟิลด์ต่าง ๆ ในเรคคอร์ดที่ตำแหน่งนั้นหรือทำการเปลี่ยนแปลงแก้ไขค่าของฟิลด์ในตำแหน่งนั้นให้มีค่าตามที่ต้องการได้ (ในกรณีที่ต้องการจะเปลี่ยนแปลงค่าของข้อมูลในตำแหน่งนั้นของอ็อปเจ็คข้อมูล)

- การอ้างถึงค่าในฟิลด์และการอ่านค่า

วิธีที่ง่ายต่อการเข้าใจ สำหรับการอ้างถึง ค่าในฟิลด์ทำได้ด้วยการอ้างถึงชื่อฟิลด์ที่ต้องการทำได้ตามรูปแบบดังนี้

recordset_object. Fields (field_name). Value

โดยที่

recordset_object เป็นตัวแปรประเภทอ็อปเจ็คข้อมูล

field_name เป็นตัวแปรหรือค่าคงที่ประเภทตัวอักษรที่หมายถึงชื่อฟิลด์ที่ต้องการ

อีกวิธีจะเป็นการระบุถึงลำดับของฟิลด์ในอ็อปเจ็คข้อมูลนั้นแทนที่จะเป็นชื่อของฟิลด์ด้วยรูปแบบ

recordset_object. Fields (field_ordinal_number).Value

โดยที่

recordset_object เป็นตัวแปรประเภทอ็อปเจ็คข้อมูล

field_ordinal_name เป็นตัวแปร หรือค่าคงที่ ที่เป็นตัวเลข ที่ใช้ระบุลำดับของฟิลด์ที่ต้องการในเรคคอร์ดนั้น โดยฟิลด์แรกจะมีลำดับที่เป็น 0

- วิธีกำหนดค่าให้กับฟิลด์

หากมองว่าฟิลด์ในอ็อปเจ็คข้อมูล เป็นเหมือนกับตัวแปรใดๆตัวหนึ่งแล้ววิธีการกำหนดค่าจะสามารถทำได้ในลักษณะเดียวกับตัวแปรอื่น ๆ ต่างเพียงแต่วิธีการอ้างถึงฟิลด์ค่อนข้างจะยาวท่านั้นดังรูปแบบ

recordset_object. Fields (field_name).Value=value

โดยที่

recordset_object เป็นตัวแปรประเภทอ็อปเจ็คข้อมูล

field_ name เป็นตัวแปรหรือค่าคงที่ประเภทตัวอักษรที่หมายถึงชื่อฟิลด์ที่ต้องการ

value เป็นตัวแปรหรือค่าคงที่ที่ต้องการจะกำหนดให้กับฟิลด์

- การจัดการค้นหาข้อมูลที่ต้องการ

วิธีการที่สะดวกกว่านี้ในการหาข้อมูลในอ็อปเจ็คที่ต้องการ คือ การเรียกใช้ฟังก์ชันการทำงานสำหรับการค้นหาข้อมูล ซึ่งมีอยู่ 5 ฟังก์ชัน พอสรุปได้พร้อมทั้งคุณสมบัติที่เกี่ยวข้อง ดังตารางประกอบข้างล่างนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชัน	คำอธิบายการทำงาน	
FindFirst	การค้นหาข้อมูลที่มีลักษณะตามที่ต้องการ โดยเริ่มค้นหา จากเรคคอร์ดแรกสุดไปยังเรคคอร์ดสุดท้ายในอ็อปเจ็คข้อมูล หากพบจะทำการเคลื่อนตัวชี้ตำแหน่งข้อมูลไปยังตำแหน่งของเรคคอร์ดที่มีข้อมูลจะอยู่ที่เดิมก่อนเริ่มค้นหา	NoMath หากพบเรคคอร์ดที่ข้อมูลในลักษณะที่กำหนด ค่าของคุณสมบัติจะมีค่าเป็น False
FindNext	มีวิธีการทำงานเช่นเดียวกันกับ FindFirst แต่จะเริ่มการค้นหาข้อมูลที่ต้องการจากเรคคอร์ดในตำแหน่งถัดต่อจากตัวชี้ตำแหน่งข้อมูลไปยังเรคคอร์ดสุดท้ายในอ็อปเจ็คข้อมูล	เช่นเดียวกันกับ FindFirst
FindPrevious	เหมือนกับฟังก์ชัน FindFirst แต่เป็นการค้นหาในทิศทางย้อนกลับ คือเริ่มจากเรคคอร์ดที่อยู่ในตำแหน่งก่อนหน้าตัวชี้ข้อมูลย้อนหลังไปยังเรคคอร์ดแรกสุดในอ็อปเจ็คข้อมูล	เช่นเดียวกันกับ FindFitst
FindLast	เหมือนกับฟังก์ชัน FindFirst แต่ในทิศทางตรงข้าม คือ เริ่มจากเรคคอร์ดสุดท้ายก่อน	เช่นเดียวกันกับ FindFirst

ตารางที่ 2.7 ฟังก์ชันการทำงานสำหรับการค้นหาข้อมูล

1. การค้นหาด้วยฟังก์ชัน Find ในลักษณะต่าง ๆ

การค้นหาข้อมูลที่ต้องการด้วยฟังก์ชัน Find ต่าง ๆ จะสามารถใช้ได้กับอ็อปเจ็คข้อมูลประเภทไดนาเซ็ตและสแน็ปช็อต ไม่สามารถจะใช้กับอ็อปเจ็คข้อมูลประเภทตารางได้

รูปแบบของคำสั่งที่ใช้กับฟังก์ชันการทำงานค้นหาข้อมูลต่าง ๆ มีดังนี้

โดยที่

recordset_object เป็นตัวแปรประเภทอ็อปเจ็คข้อมูลประเภทไดนาเซ็ตหรือสแน็ปช็อต

criteria เป็นตัวแปรหรือค่าคงที่ประเภทตัวอักษรที่กำหนดเงื่อนไขของการค้นหาข้อมูล

รูปแบบและการกำหนดค่าเงื่อนไขการค้นหา (criteria)

criteria เป็นตัวแปรหรือค่าคงที่ประเภทตัวอักษร (String) เพื่อกำหนดว่าต้องการจะค้นหาข้อมูลในลักษณะอย่างไร วิธีการกำหนดค่าให้กับ criteria จะมีหลักการเหมือนกับการกำหนดค่าในส่วนของ Where ในคำสั่ง Select ของ SQL เพียงแต่ไม่มีคำว่า Where ประกอบด้วยเท่านั้น รูปแบบมีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟิลด์ที่มีค่าเป็นตัวเลข เช่น Integer, Long, Curency และอื่น ๆ

field_name [=,<>,>,<=,>=] value หรือ

field_name In (value 1,value 2, …)

โดยที่

field_name เป็นชื่อของฟิลด์ที่ต้องการจะค้นหาข้อมูลที่อยู่ในฟิลด์นี้

value 1, value 2,… เป็นค่าที่ต้องการจะค้นในฟิลด์ field_name

ฟิลด์ที่มีชนิดเป็นตัวอักษร เช่น Text

field_name[=,<>,>,<=,>=] value หรือ

field_name In (value 1,value 2, …)

field_name Like pattern_string

โดยที่

field_name เป็นชื่อของฟิลด์ที่ต้องการจะค้นหาข้อมูลที่อยู่ในฟิลด์นี้

value 1, value 2, … เป็นค่าที่ต้องการจะค้นหาในฟิลด์ field_name

pattern_string เป็นรูปแบบชุดตัวอักษร (pattern) ที่ต้องการจะค้นหา

2. การค้นหาโดยใช้ฟังก์ชัน Seek

การค้นหาโดยใช้ฟังก์ชันการทำงาน Seek เป็นการค้นหาข้อมูลหรือเรคคอร์ดที่ต้องการ โดยการค้นหาจากดรรชนี การค้นหาโดยการใช้ฟังก์ชันการทำงาน Seek จะเร็วมาก แต่ข้อจำกัดของการใช้ Seek ก็คือจะใช้ได้กับอ็อปเจ็คข้อมูลประเภทตารางเท่านั้น ไม่สามารถจะใช้กับอ็อปเจ็คข้อมูลประเภทไดนาเซ็ตหรือสแน็ปช็อตได้ และมีวิธีการใช้ที่ค่อนข้างยุ่งยากคือ ต้องกำหนดว่าจะให้ค้นหาดรรชนีตัวใดของตารางนั้น ดังนั้นการระบุดรรชนีที่ใช้สำหรับการ Seek ต้องใช้ดรรชนีที่กำหนดจากฟิลด์ที่ต้องการจะค้นหาข้อมูล รูปแบบของคำสั่งที่ใช้กับฟังก์ชันการทำงาน Seek มีดังนี้

recordset_object. Index = index_name

recordset_object. seek comparison_operator, value

โดยที่

recordset_object คือตัวแปรอ็อปเจ็คข้อมูลประเภทตาราง

indexname เป็นตัวแปรหรือค่าคงที่ชนิดตัวอักษร มีค่าเป็นชื่อของดรรชนีที่จะใช้ในการค้นหา

comparison_operator เป็นตัวแปรหรือค่าคงที่ชนิดตัวอักษร ระบุวิธีการเปรียบเทียบ จะกำหนดได้เพียง =,<>,>,<=,>=

comparison_value เป็นตัวแปรมีชนิดตามชนิดของฟิลด์ ที่มีค่าที่ต้องการค้นหา ข้อควรระวังในการค้นหาข้อมูลโดยการใช้ฟังก์ชัน seek มีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. โดยปกติการ seek จะเริ่มค้นหาจากเรคคอร์ดแรกตามดรรชนีที่ใช้ แต่หากเราได้ทำการกำหนด comparison_operator เป็น < หรือ <= จะเป็นการเริ่มหาเรคคอร์ดจากเรคคอร์ดสุดท้ายตามดรรชนีก่อนแล้วจึงย้อนกลับ
2. ผลของการค้นหาโดย seek ว่าพบหรือไม่ จะแสดงที่ค่าของคุณสมบัติ NoMath ของอ็อปเจ็คข้อมูลที่ค้นหา เหมือนกับการค้นหาโดยใช้ฟังก์ชันต่าง ๆ ของ Find

- การเพิ่มข้อมูลใหม่

วิธีการเพิ่มข้อมูลใหม่เข้าไปในอ็อปเจ็คข้อมูลนั้น จะต้องเรียกใช้ฟังก์ชันการทำงานกับอ็อปเจ็คข้อมูลนั้น 2 ฟังก์ชันด้วยกัน คือ AddNew และ Update โดยมีรูปแบบของการเรียกใช้ดังนี้

recordset_object. AddNew เป็นการสั่งให้เพิ่มเรคคอร์ดว่าง ๆ หนึ่งเรคคอร์ดแล้วจึงตามด้วยคำสั่งสำหรับกำหนดค่าให้กับฟิลด์ต่าง ๆ ตามต้องการ

recordset_object. Field("FIELD_NAME") =value

…

…

recordset_object. Update เป็นการสั่งให้แก้ไขค่าในฐานข้อมูลตามที่กำหนด

โดยที่

recordset_object คือตัวแปรอ็อปเจ็คข้อมูล

field_name คือชื่อของฟิลด์ที่ต้องการจะกำหนดค่า

value เป็นตัวแปรหรือค่าคงที่ที่ต้องการจะกำหนดให้กับฟิลด์

- การแก้ไขข้อมูลเดิม

มีลักษณะการใช้งานที่เหมือนกับการเพิ่มข้อมูลใหม่ คือต้องเรียกฟังก์ชันการทำงาน 2 ฟังก์ชันเหมือนกัน แต่จะต้องเคลื่อนตัวชี้ตำแหน่งข้อมูลของอ็อปเจ็คข้อมูลไปยังตำแหน่งของเรคคอร์ดที่ต้องการจะแก้ไขเสียก่อนจะด้วยวิธีการเคลื่อนตัวชี้ตำแหน่ง (การ Move ต่าง ๆ) การค้นหา (การ Find ต่าง ๆ หรือ Seek) หรือการระบุตำแหน่งโดยกำหนดค่าให้กับคุณสมบัติ Bookmark โดยตรงก็ได้

รูปแบบและตัวอย่างที่แสดงจะสมมติว่า ได้เคลื่อนตัวชี้ตำแหน่งมายังตำแหน่งของข้อมูลที่ต้องการแล้ว

recordset_object. Edit ประกาศว่าจะเริ่มแก้ไขข้อมูลที่ตำแหน่งของตัวชี้ กำหนดค่าให้กับฟิลด์ที่ต้องการจะเปลี่ยนค่า

recordset_object. Fields ("FIELD_NAME")=value

…

…

recordset_object. Updata สั่งให้เปลี่ยนข้อมูลในฐานข้อมูลได้แล้ว

- การลบข้อมูล

เมื่อต้องการจะลบเรคคอร์ดหนึ่งออกจากอ็อปเจ็คข้อมูล ให้เรียกฟังก์ชันการทำงาน Delete บน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

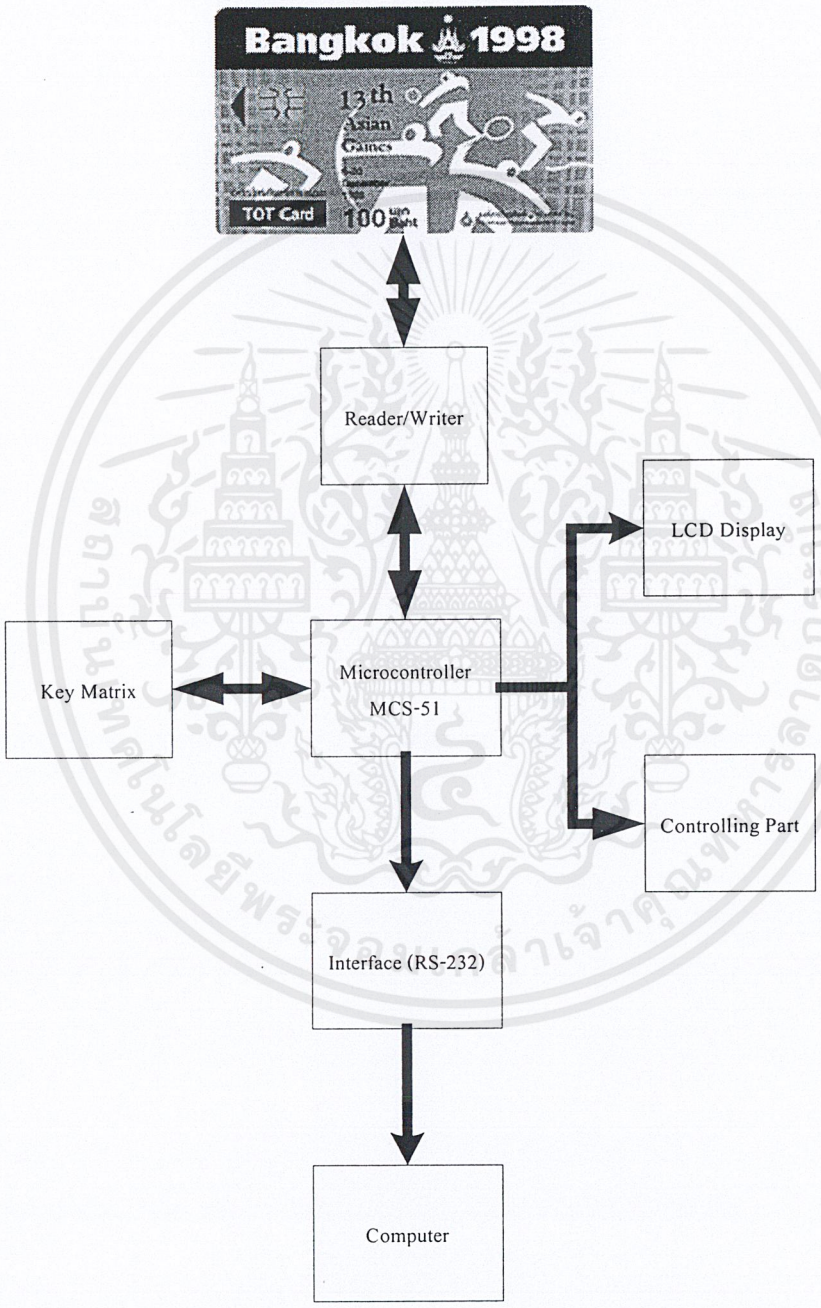
อ็อปเจ็คข้อมูลนั้น โดยที่จะต้องเคลื่อนตัวชี้ตำแหน่งข้อมูลไปยังเรคคอร์ดที่ต้องการจะลบเสียก่อน รูปแบบของฟังก์ชันที่เรียกใช้ มีดังนี้

recordset_object. Delete

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การคำนวณและการสร้าง

การออกแบบการทำงานของโครงงานนี้แบ่งออกเป็น 3 ส่วนใหญ่อันได้แก่ ส่วนที่ทำการประมวลผลส่วนที่ติดต่อกับบัตรสมาร์ตการ์ดและโปรแกรมบนเครื่องคอมพิวเตอร์ ดังแสดงได้ในรูปที่ 3.1



รูปที่ 3.1 บล็อกไดอะแกรมของระบบ

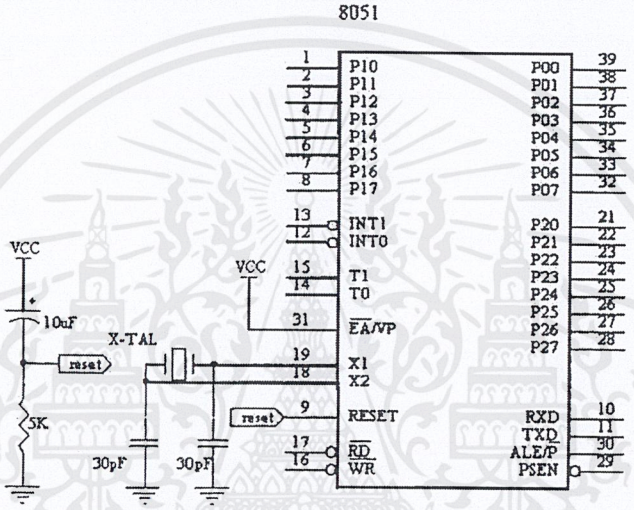
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1 ส่วนที่ทำการประมวลผล

ส่วนประมวลผลนี้ยังแบ่งออกเป็นอีก 3 ส่วน คือ ส่วนรับค่าจากคีย์บอร์ด (Key Matrix) ส่วนแสดงผล (LCD) ส่วนประมวลผลกลาง (Microprocessor)

3.1.1 ส่วนประมวลผลกลาง

ใช้ไมโครคอนโทรลเลอร์ AT 89S8252 ซึ่งเป็นตระกูลเกี่ยวกับ MCS-51 แต่มีคุณสมบัติแตกต่างไปคือมีหน่วยความจำสำหรับเก็บข้อมูลทั่วไป 256 ไบท์คือที่ตำแหน่ง 00H-FFH มีหน่วยความจำสำหรับเก็บโปรแกรมจึงไม่ต้องการต่อ EPROM ภายนอกเพื่อแทนทำให้การต่อขา EA ของซีพียู ต้องต่อ +Vcc ส่วนสัญญาณนาฬิกา ของระบบจะต้องใช้ออสซิลเลเตอร์ ที่มีความถี่ขนาด 11.059 เมกกะเฮิรตซ์



รูปที่ 3.2 แสดงการต่อวงจรของไมโครคอนโทรลเลอร์

3.1.2 ส่วนรับค่าจากคีย์บอร์ด

คีย์บอร์ดที่นิยมและใช้กันอย่างแพร่หลายและนำมาอินเตอร์เฟสร่วมกับไมโครคอนโทรลเลอร์ จะมีอยู่ 2 ชนิด คือ

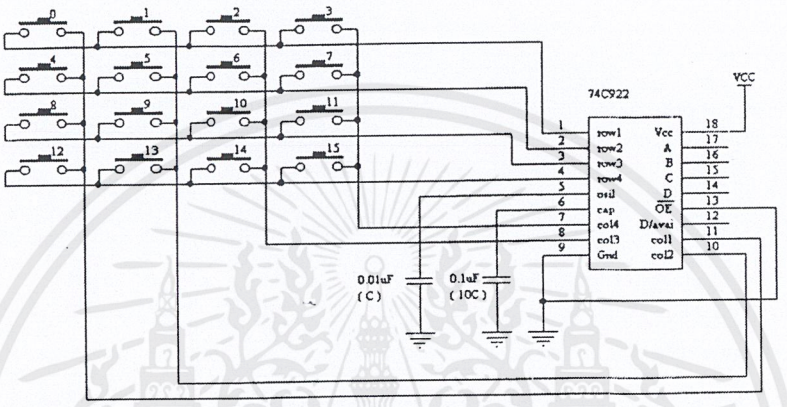
1. สวิตช์ชนิดกดติดปล่อยดับ (Single Pole/Single Throw; SPST)
2. IC Key Encoder

ทั้งสองชนิดดังกล่าวนั้นถูกนำมาใช้เป็นคีย์สวิตช์ หรือคีย์บอร์ดในการใช้งานร่วมกับไมโครคอนโทรลเลอร์

การอินเตอร์เฟสคีย์สวิตช์ ในรูปแบบที่ 2 โดยจะเป็นการประยุกต์ใช้งาน IC Key Encoder ในการใช้งานเป็นคีย์สวิตช์ หรือคีย์บอร์ด ร่วมกับการใช้งานไมโครคอนโทรลเลอร์ ซึ่งการใช้งาน IC Key Encoder นี้จะมีลักษณะฟังก์ชันในรูปแบบของ คีย์บอร์ด คือ ลักษณะของเมทริกซ์สวิตช์ IC Key Encoder ที่ใช้จะเป็นเบอร์ 74C922 และ 74C923 ซึ่งในเบอร์ 74C922 นั้นจะเป็นชนิด 16 ขา Key Encoder ในเบอร์ 74C923 โดยทั้ง 2 เบอร์นั้นจะเป็นไอซี ชนิด CMOS กินพลังงานต่ำและจะต้องมีการใช้งานควบคู่กันกับสวิตช์ชนิด SPST

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยจะเห็นว่าสวิตช์ชนิด SPST นั้น เมื่อทำการกดจะมีการกระเด้งของหน้าสัมผัสหรือ ที่เรียกว่าเกิดการ Bounce ถ้ากรณีใช้งานโดยตรง ผลของการ Bounce ก็อาจจะส่งผลกระทบต่อการตรวจสอบสภาวะของตัวไมโครคอนโทรลเลอร์ได้ แต่เมื่อใช้ IC Key Encoder ดังกล่าวแล้วสภาวะที่เอาท์พุทที่ออกจากไอซีจะมีความแน่นอน (Stable) มาก เพราะภายในไอซี นี้จะประกอบไปด้วยวงจรต่างๆ ซึ่งรวมถึงวงจรในการป้องกันการกระเด้งของหน้าสัมผัสของสวิตช์ที่นำมาใช้งาน



รูปที่ 3.3 แสดงการต่อคีย์บอร์ด

(Pins 0 through 11)

Switch Position	0 Y1,X1	1 Y1,X2	2 Y1,X3	3 Y1,X4	4 Y2,X1	5 Y2,X2	6 Y2,X3	7 Y2,X4	8 Y3,X1	9 Y3,X2	10 Y3,X3	11 Y3,X4
D												
A A	0	1	0	1	0	1	0	1	0	1	0	1
T B	0	0	1	1	0	0	1	1	0	0	1	1
A C	0	0	0	0	1	1	1	1	0	0	0	0
O D	0	0	0	0	0	0	0	0	1	1	1	1
U E (Note 1)	0	0	0	0	0	0	0	0	0	0	0	0
T												

(Pins 12 through 19)

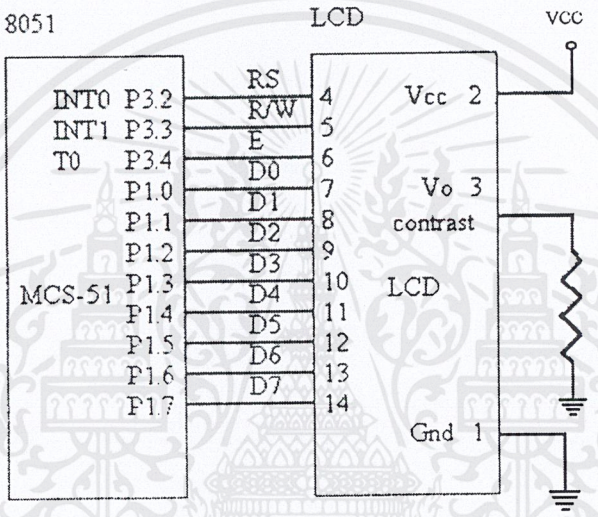
Switch Position	12 Y4,X1	13 Y4,X2	14 Y4,X3	15 Y4,X4	16 Y5(Note 1), X1	17 Y5 (Note 1), X2	18 Y5 (Note 1), X3	19 Y5 (Note 1), X4
D								
A A	0	1	0	1	0	1	0	1
T B	0	0	1	1	0	0	1	1
A C	1	1	1	1	0	0	0	0
C D	1	1	1	1	0	0	0	0
U E (Note 1)	0	0	0	0	1	1	1	1
T								

ตารางที่ 3.1 แสดงค่าความจริงของ IC Key Encoder

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.3 ส่วนแสดงผล

แอลซีดีโมดูลที่ใช้งานจะมีอยู่ 14 ขา โดยขา 1, 2 และ 3 จะไม่พูดถึงเนื่องจากไม่มีความซับซ้อนในการใช้งาน สามารถต่อตามชื่อขาได้เลย จะเหลืออยู่ 11 ขา คือ ขา RS, R/ $\overline{W}$, E และ D0-D7 จากรูปจะเห็นว่า ใช้บิต P3.2 เป็น RS บิต P3.3 เป็น R/ $\overline{W}$ และบิต P3.2 เป็น E (Enable) ให้กับแอลซีดี (หรือจะใช้งานบิทอิสระอื่นๆ ของตัวไมโครคอนโทรลเลอร์ เพื่อมาควบคุมขา RS, R/ $\overline{W}$, และ E ของแอลซีดีก็ได้แต่จากวงจรข้างต้นจะใช้บิต P3.2, P3.3, P3.4 ของพอร์ท 3 ในการควบคุม) และพอร์ท 1 ทั้งหมดเป็น Data Bus 8 บิตให้กับแอลซีดี

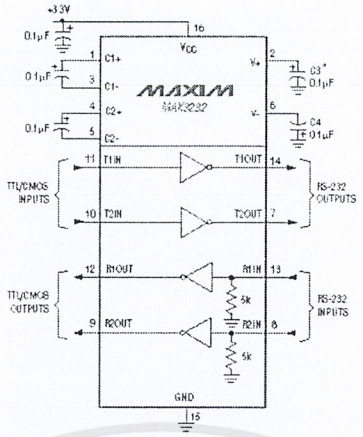


รูปที่ 3.4 แสดงการต่อแอลซีดีโมดูล

3.1.4 ส่วนที่ทำการติดต่อกับคอมพิวเตอร์

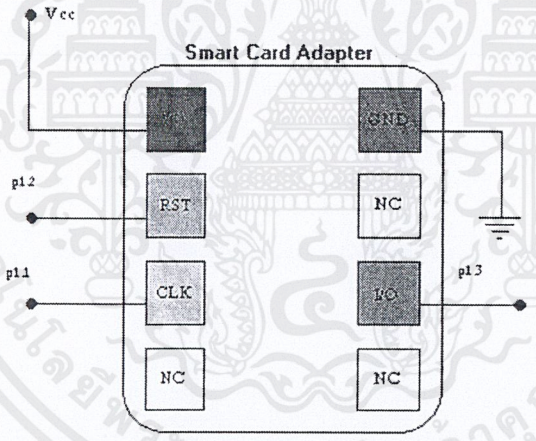
มาตรฐานการสื่อสารข้อมูลของพอร์ทอนุกรม ได้ระบุระดับแรงดันสำหรับการทำงานของพอร์ทอนุกรมไว้ว่า ที่ลอจิก "0" จะมีระดับสัญญาณ +3 โวลต์ ถึง +15 โวลต์ ส่วนลอจิก "1" จะมีระดับสัญญาณ –3 โวลต์ ถึง –15 โวลต์ ระดับสัญญาณนี้ทำให้ไม่สามารถที่จะนำขาเอาท์พุทใดๆ ต่อเข้ากับลอจิกเกจเพื่อใช้งานโดยตรงจะต้องผ่านวงจรเพื่อเปลี่ยนระดับแรงดันเสียก่อน โดยปกติจะใช้ไอซีจำพวก RS-232 transceiver ที่นิยมมากคือ MAX232 หรือ ICL 232 ไอซีในกลุ่มนี้จะทำหน้าที่แปลงระดับแรงดันของ RS-232 ให้อยู่ในระดับทีทีแอล โดยลอจิก "0" ซึ่งเดิมมีระดับสัญญาณ +3 โวลต์ ถึง +15 โวลต์ จะถูกแปลงเป็น 0 โวลต์ ส่วนลอจิก "1" ซึ่งมีระดับสัญญาณ –3 โวลต์ ถึง –15 โวลต์ จะแปลงเป็น +5 โวลต์ ทั้งนี้เพื่อกำหนดให้สามารถเชื่อมต่อกับอุปกรณ์ดิจิตอลอื่นที่ใช้ระดับแรงดันทีทีแอลได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 แสดงการต่อ IC MAX 232

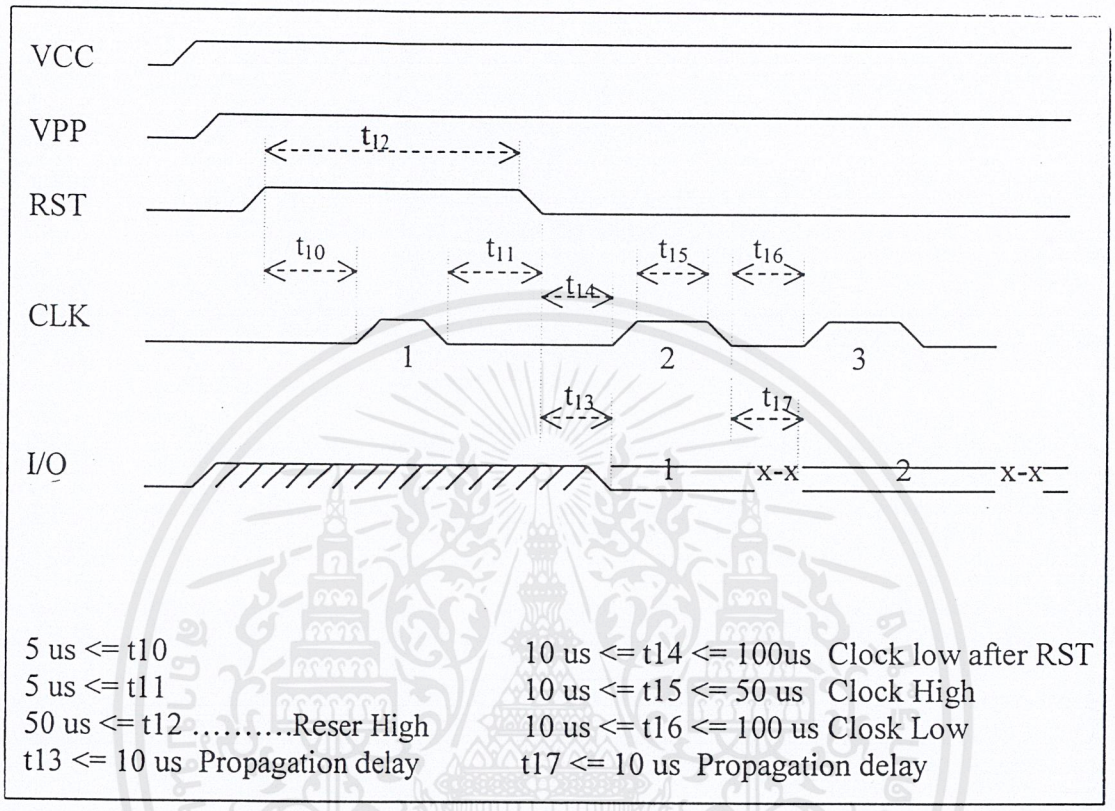
3.2 ส่วนที่ติดต่อกับสมาร์ทการ์ด



รูปที่ 3.6 แสดงการเชื่อมต่อหน้าสัมผัสของบัตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

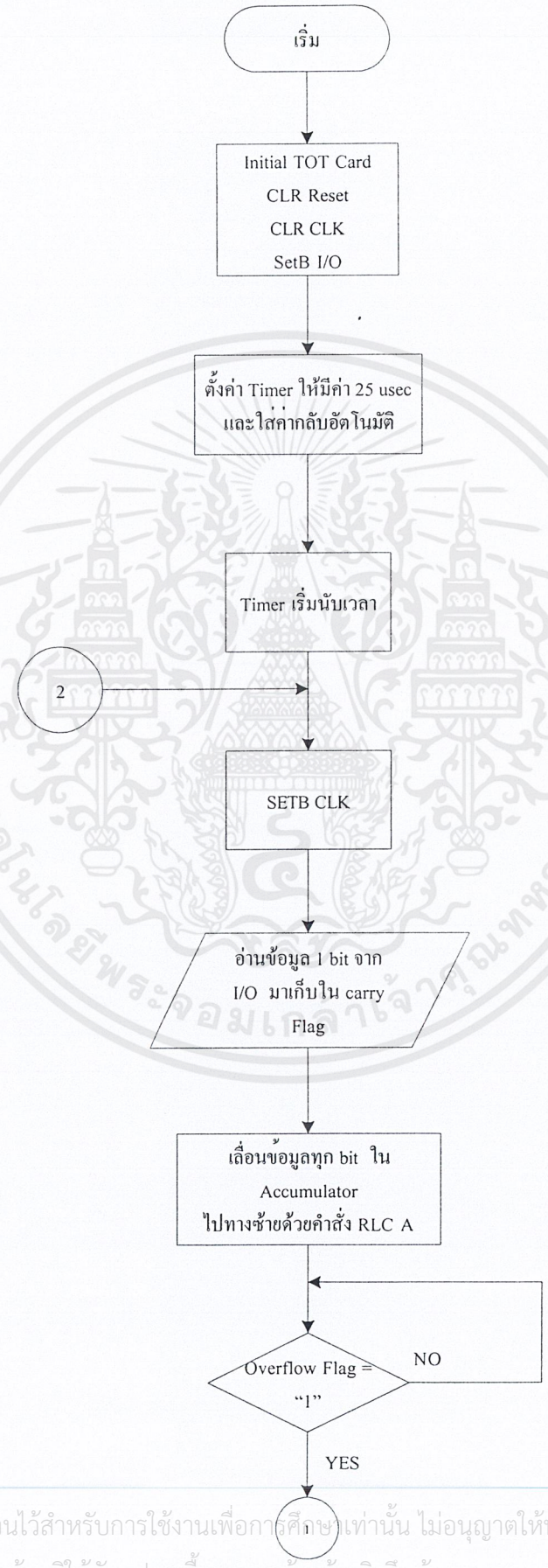
3.2.1 ไทม์มิ่งไดอะแกรมของการทำงานของบัตรสมาร์ตการ์ด



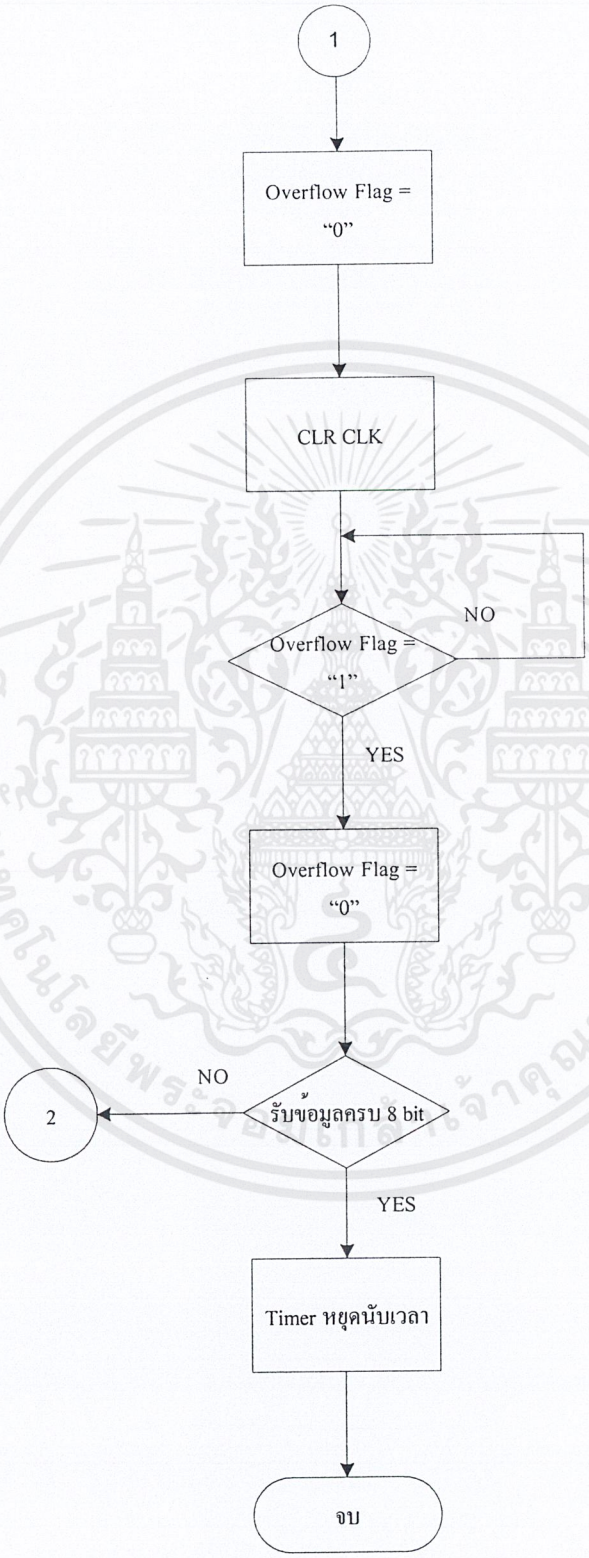
รูปที่ 3.7 ไทม์มิ่งไดอะแกรมของการทำงานของบัตรสมาร์ตการ์ดแบบซิงโครนัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 แผนผังการทำงานของโปรแกรม

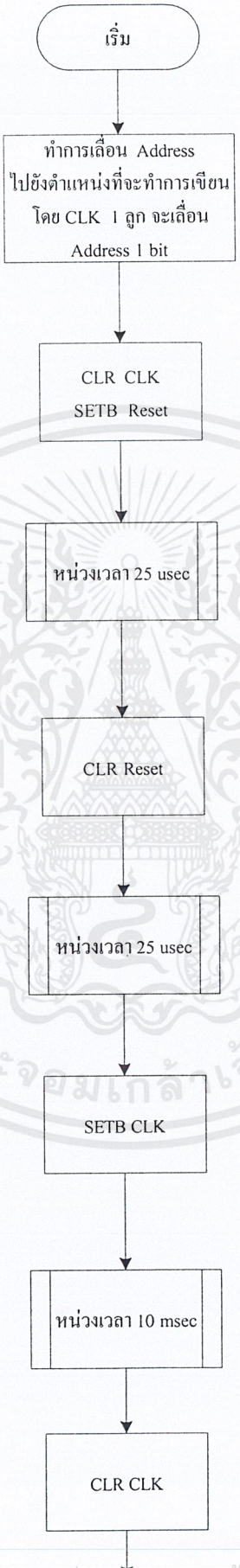


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 แสดง Flow chart การอ่านข้อมูลจากบัตร TOT

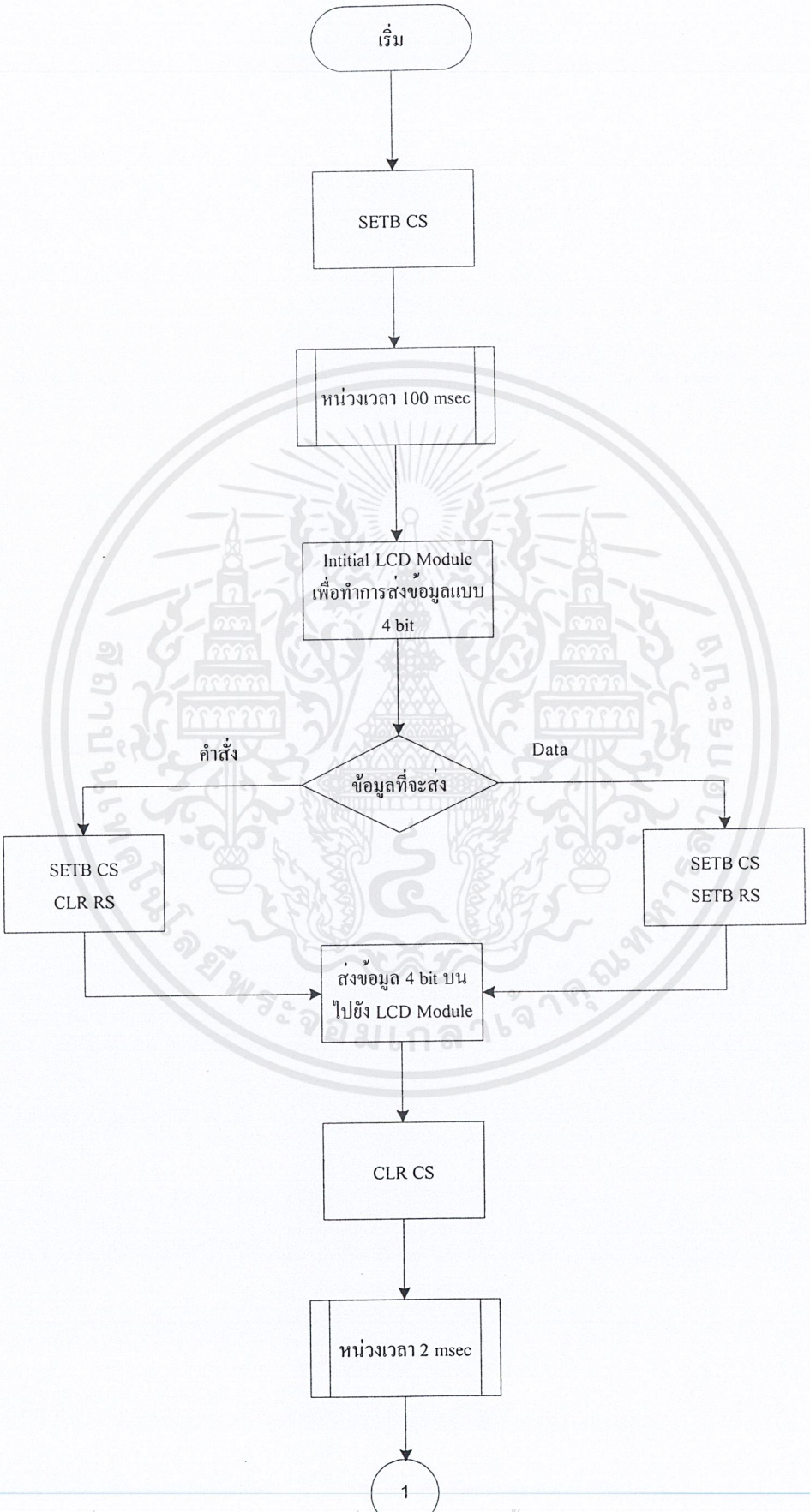
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



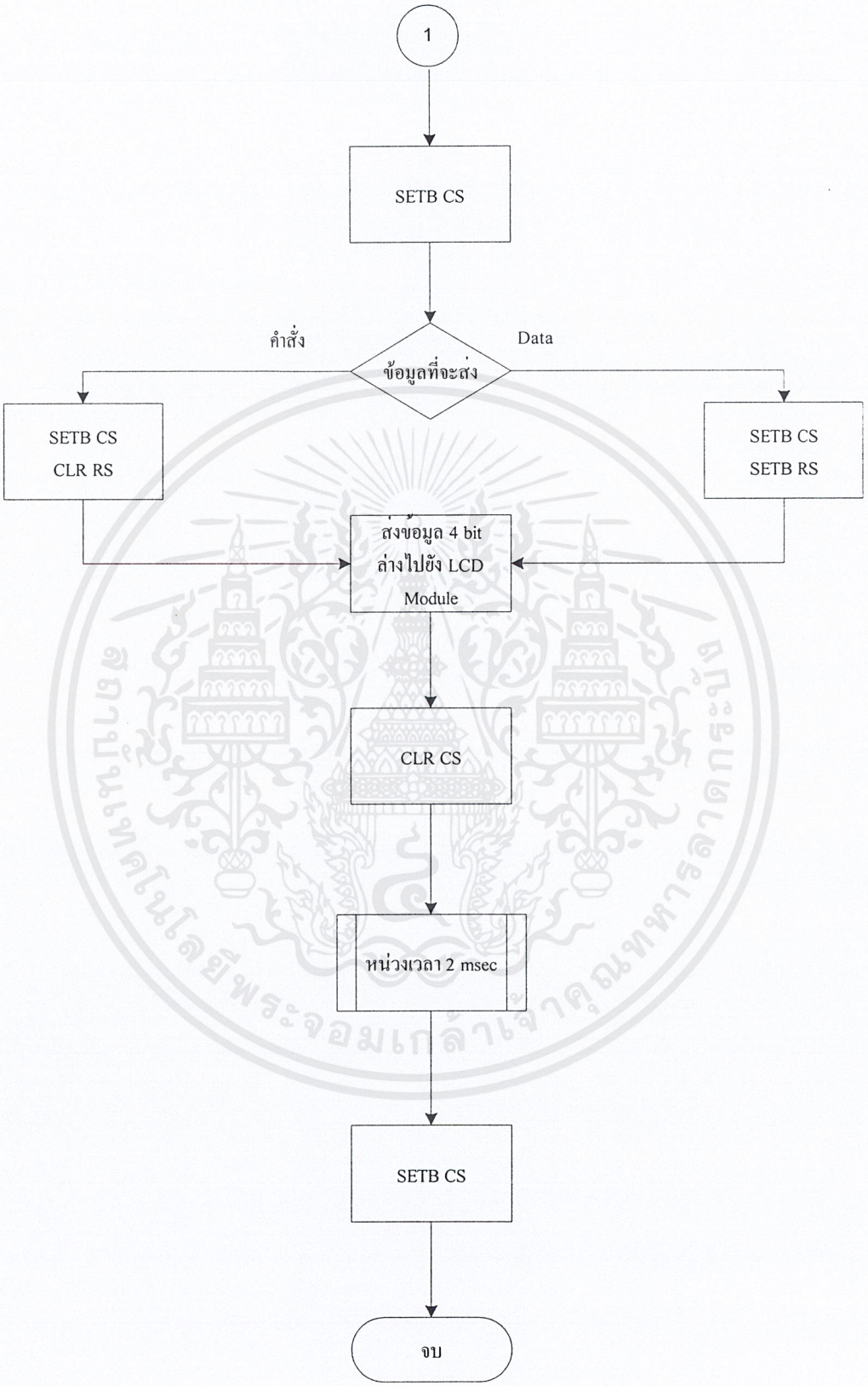
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.9 แสดง Flow chart การเขียนข้อมูลลงในบัตร TOT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

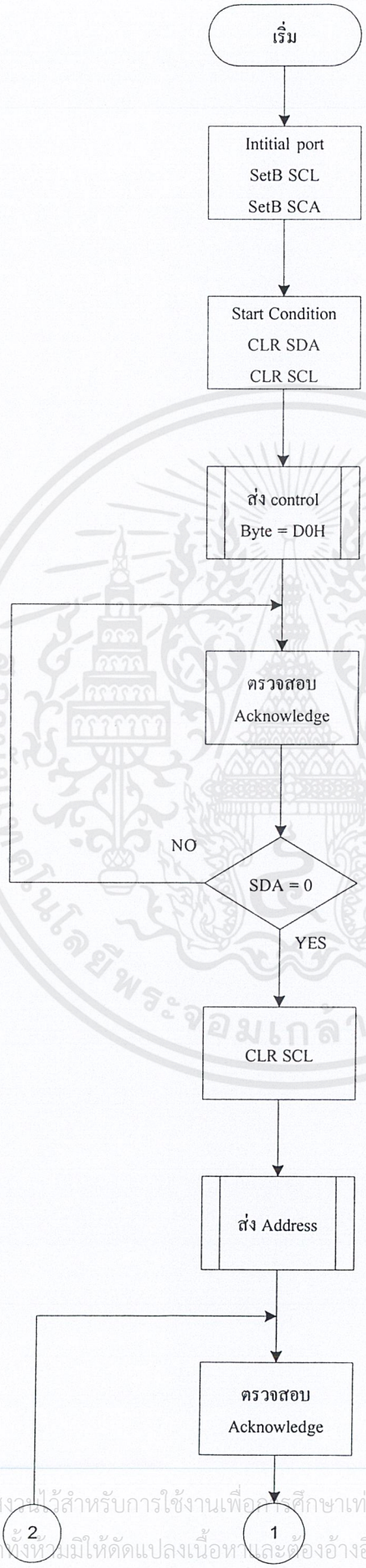


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

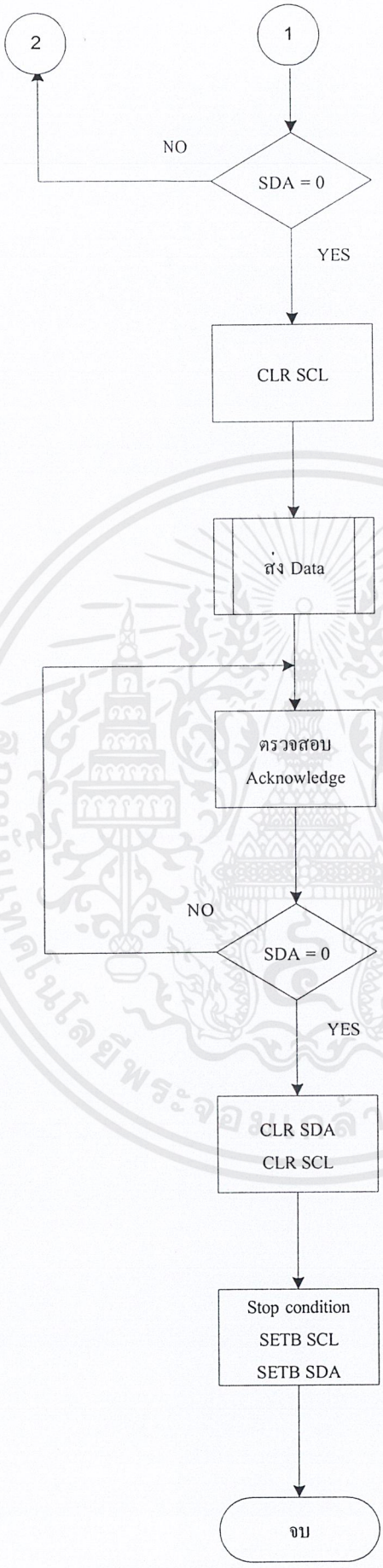


รูปที่ 3.10 แสดง Flow chart ในการส่งข้อมูล 1 Byte ไปยัง LCD Module แบบทีละ 4 bit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

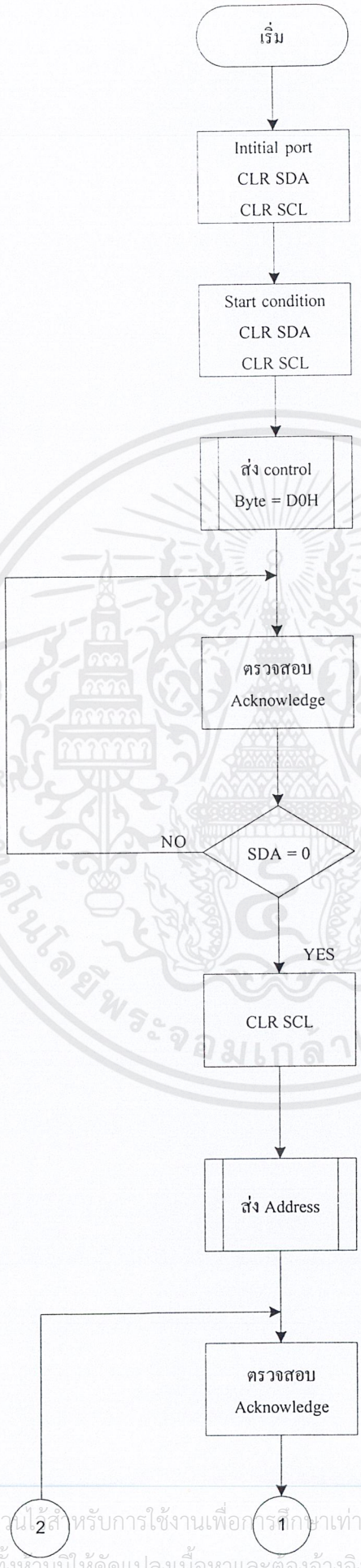


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

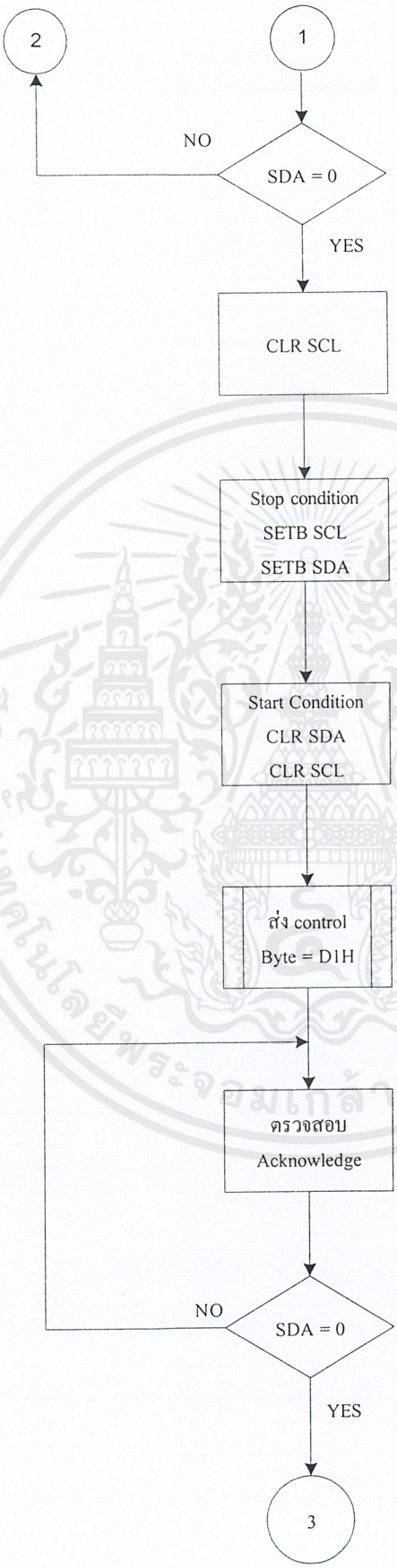


รูปที่ 3.11 แสดง Flow chart การเขียนข้อมูลลงใน IC DS1307

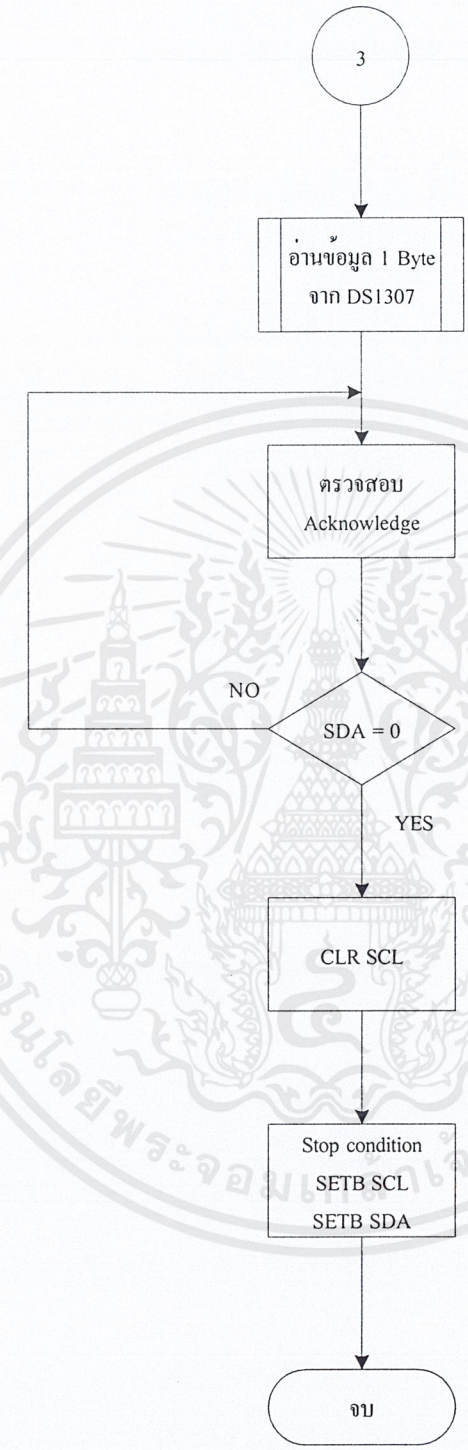
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

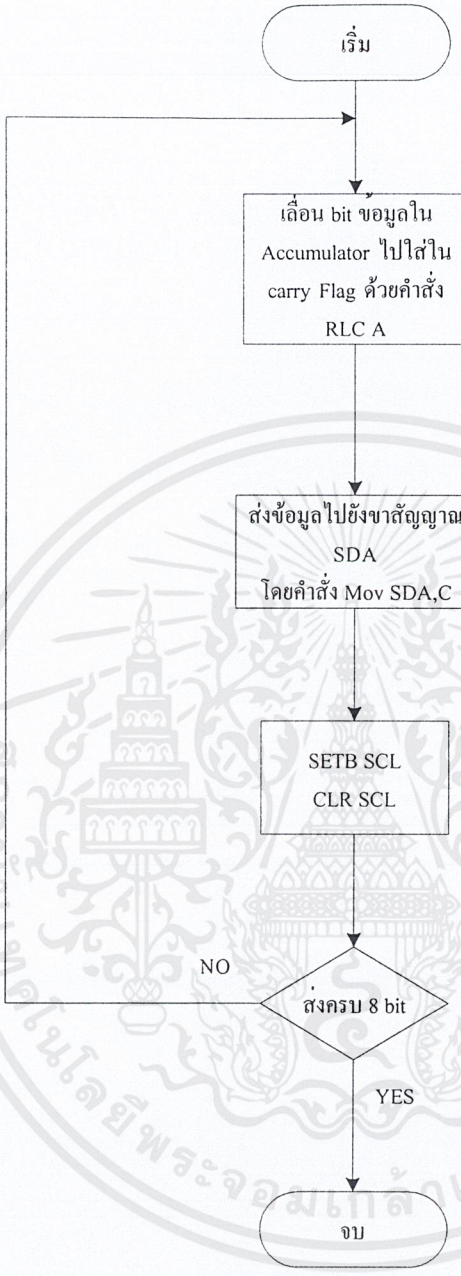


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



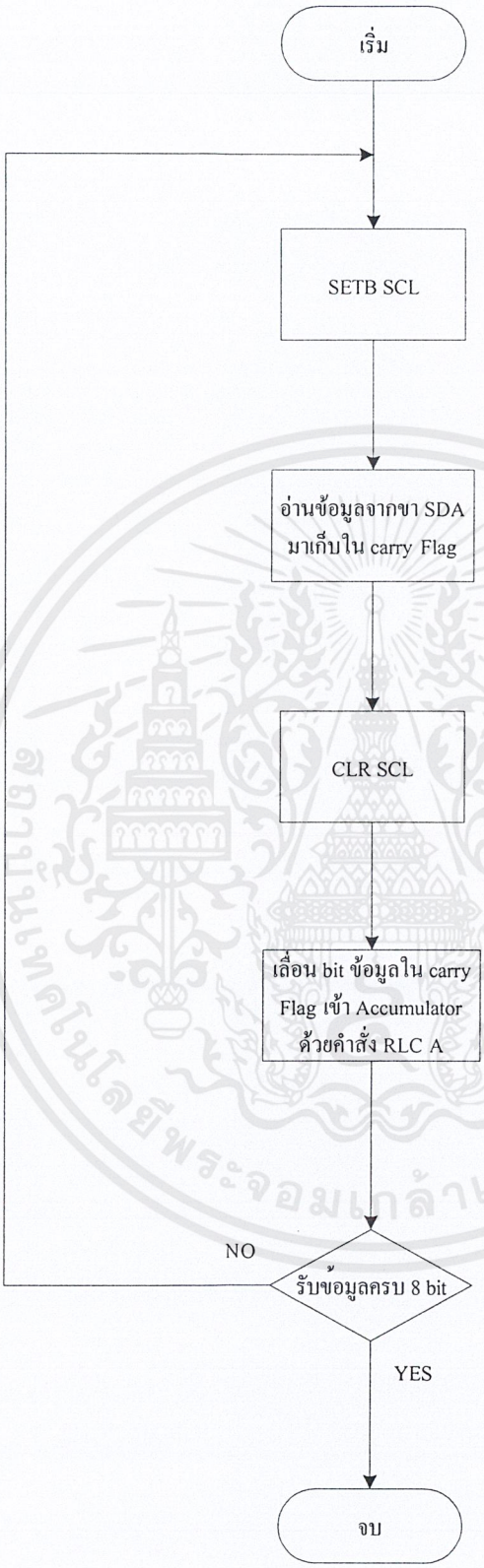
รูปที่ 3.12 แสดง Flow chart ของการอ่านข้อมูลจาก IC DS1307

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.13 แสดง Flow chart ของการส่งข้อมูล 8 bit ไปยัง IC DS1307

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



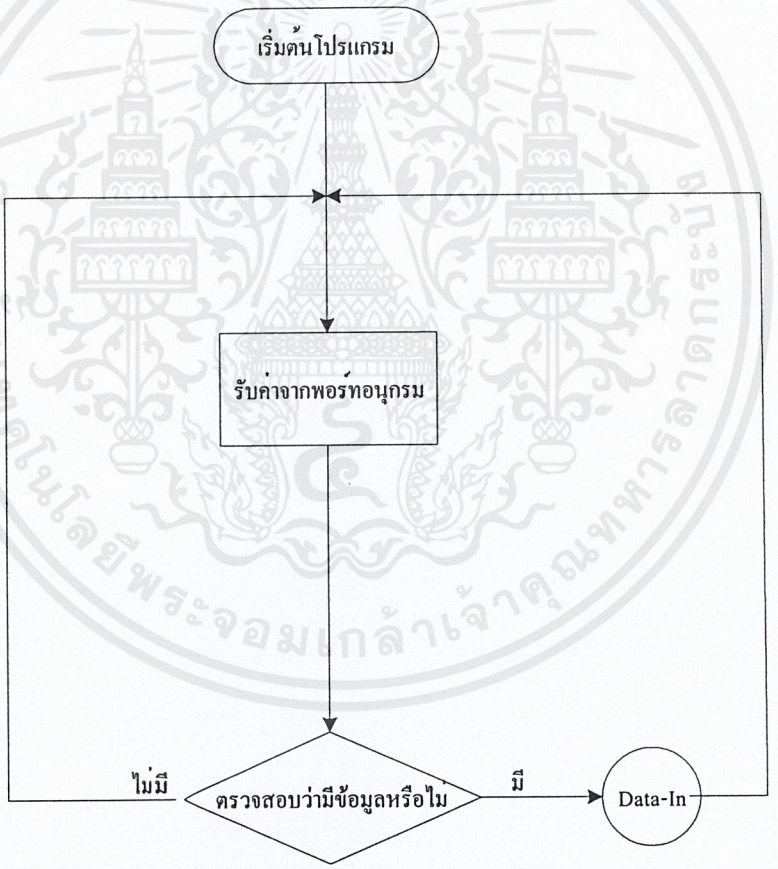
รูปที่ 3.14 แสดง Flow chart ของการรับข้อมูล 8 bit จาก IC DS1307

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 ส่วนติดต่อและแสดงผลทางคอมพิวเตอร์

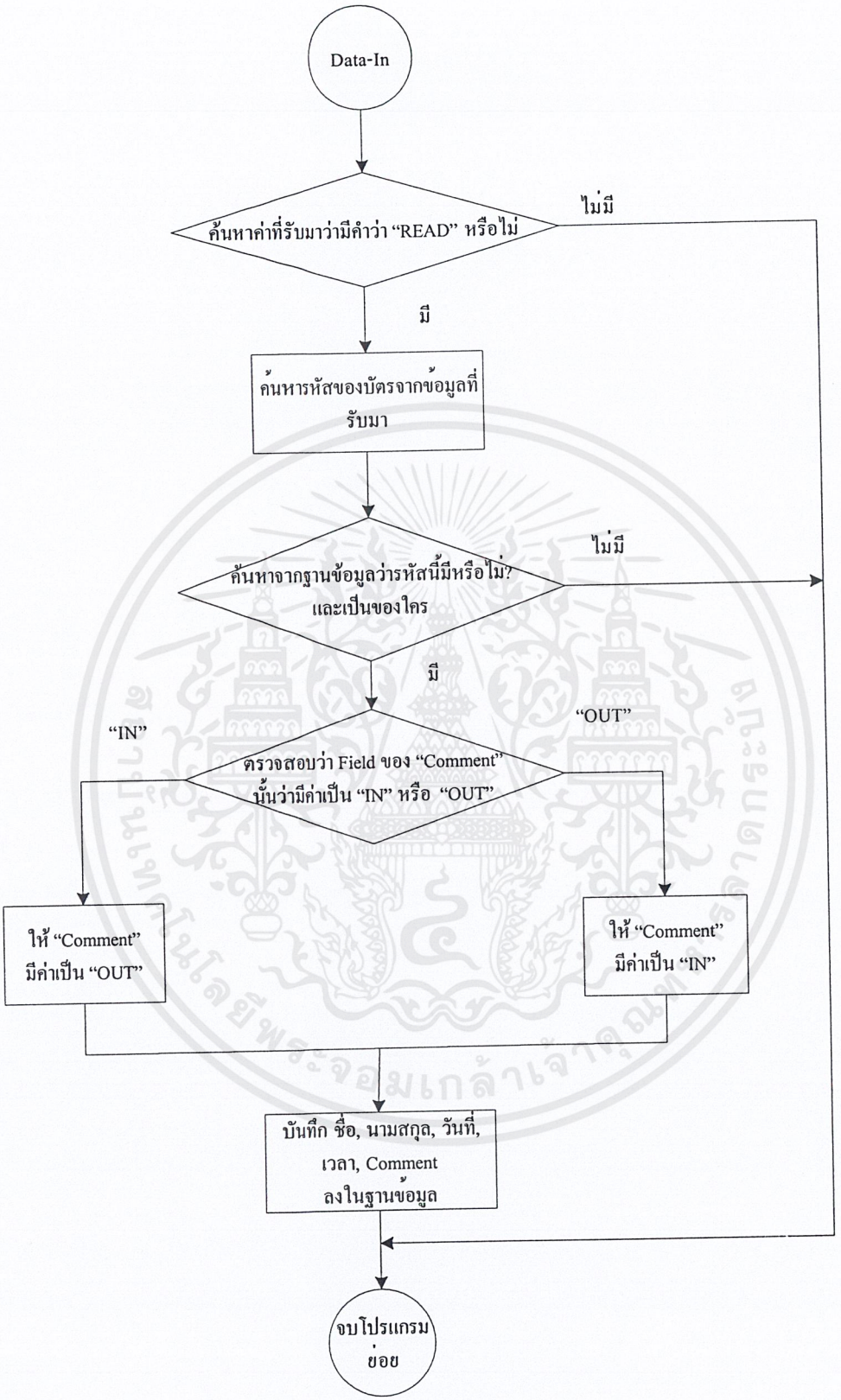
สำหรับการติดต่อสื่อสารข้อมูลระหว่างไมโครคอนโทลเลอร์กับคอมพิวเตอร์จะติดต่อสื่อสารผ่านทางพอร์ทอนุกรม ซึ่งได้ใช้โปรแกรม Visual Basic เป็นโปรแกรมหลักในการติดต่อและแสดงผลผ่านทางจอมอนิเตอร์ เพื่อแสดงข้อมูลต่างๆของผู้ใช้งาน โดยข้อมูลต่างๆนั้นจะถูกเก็บไว้ในลักษณะของฐานข้อมูลโดยโปรแกรมนั้นจะถูกแบ่งออกเป็น 3 ส่วนหลักๆคือ

1. ส่วนแสดงเวลาการเข้าออกของพนักงาน
2. ส่วนคำนวณเงินเดือนของพนักงาน
3. ส่วนของประวัติของพนักงาน



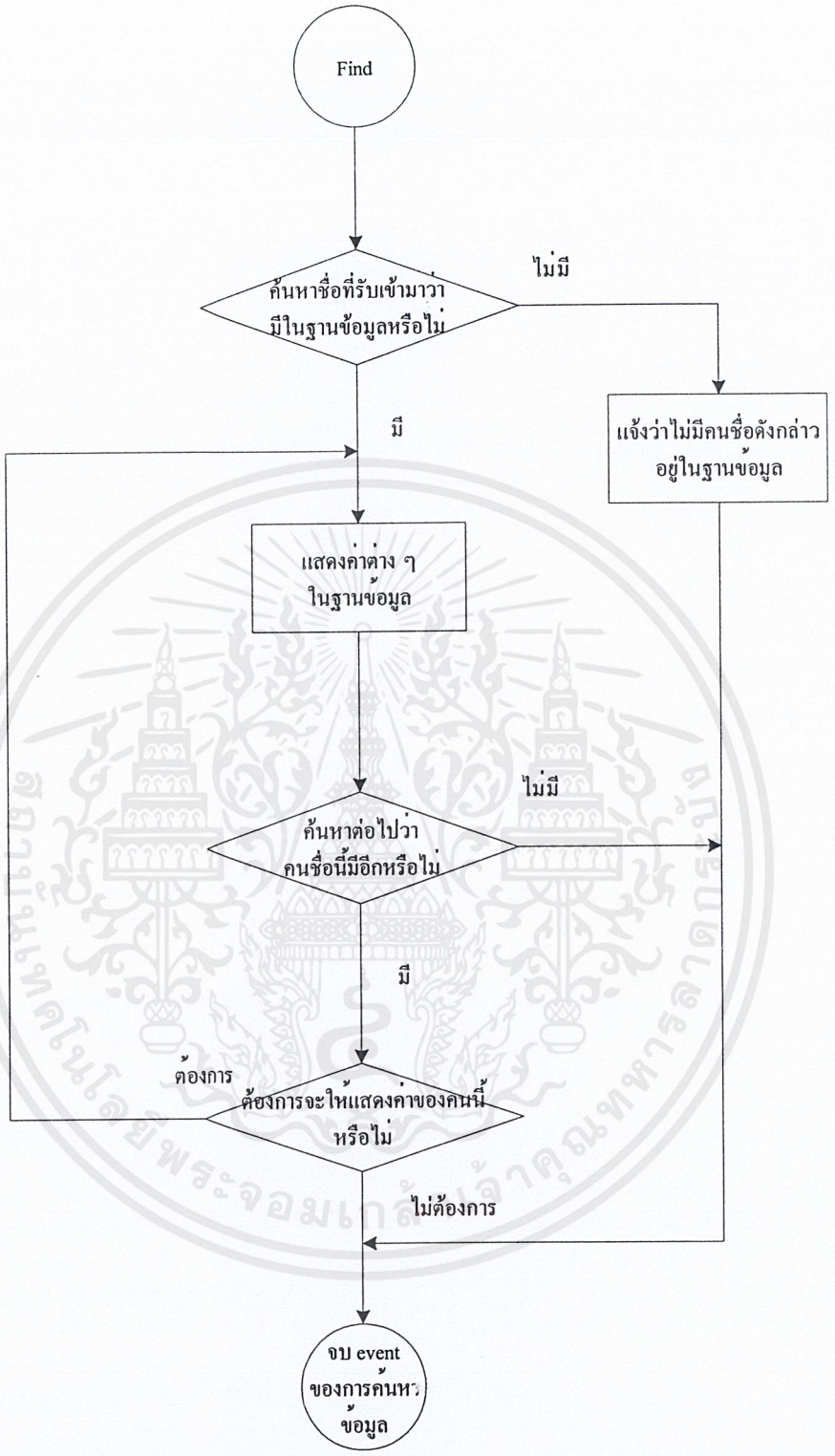
<u>รูปที่ 3.15 แสดงโฟล์วชาร์ทการทำงานหลักของโปรแกรม</u>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



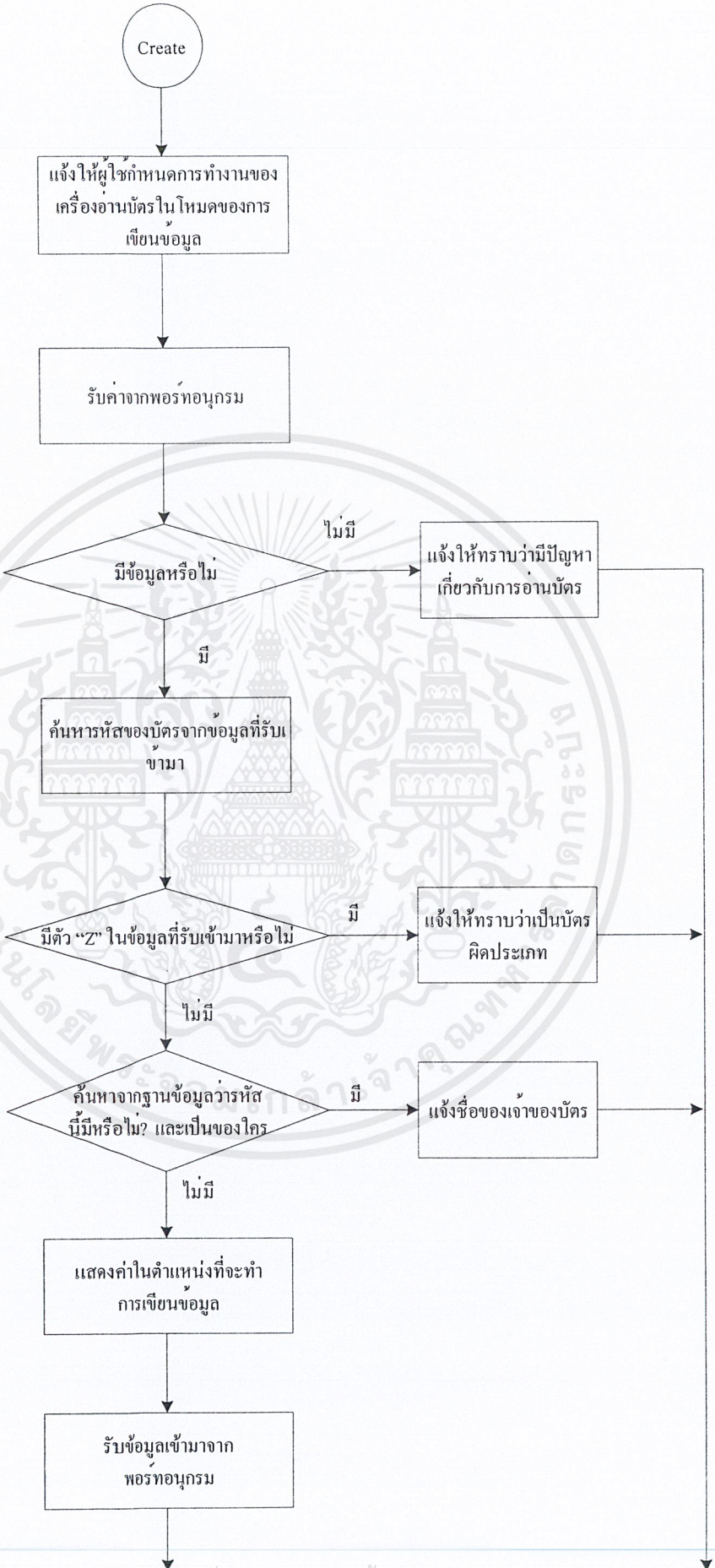
<u>รูปที่ 3.16 โฟลว์ชาร์ทของการตรวจสอบการเข้า-ออก ของพนักงาน</u>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

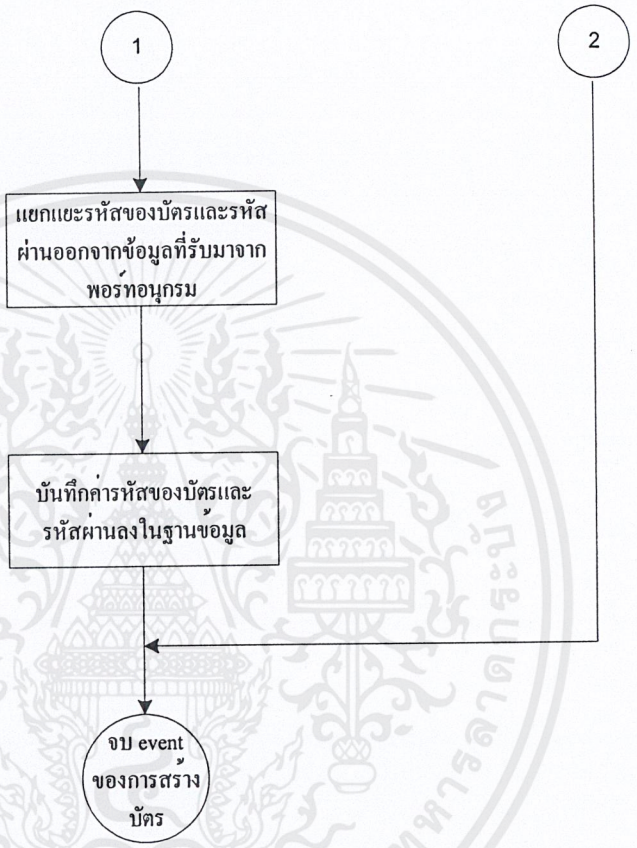


<u>รูปที่ 3.17 แสดงโฟลว์ชาร์ทของ event การค้นหาข้อมูล</u>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

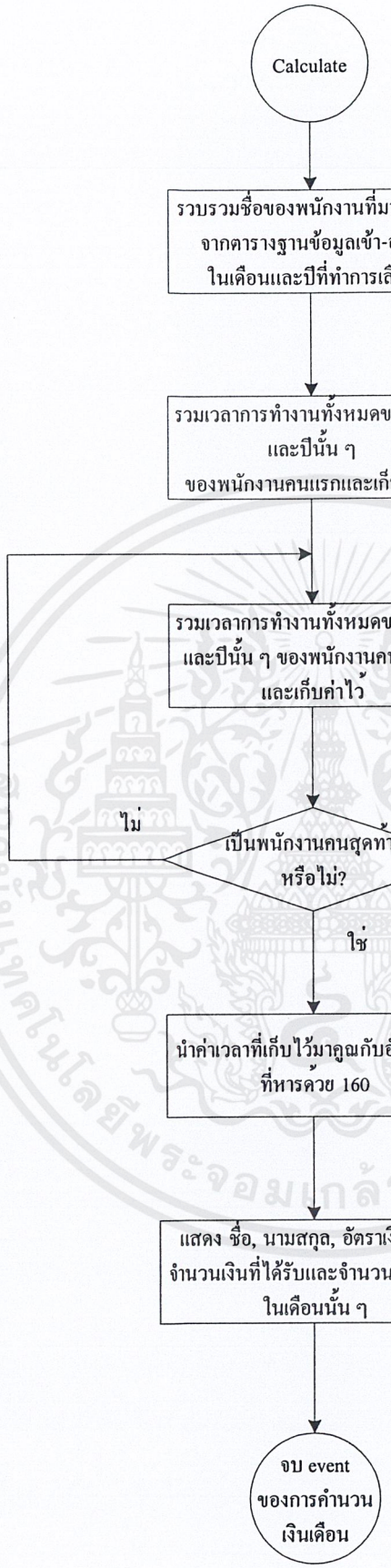


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.18 แสดงโฟลว์ชาร์ทของ event การสร้างบัตร

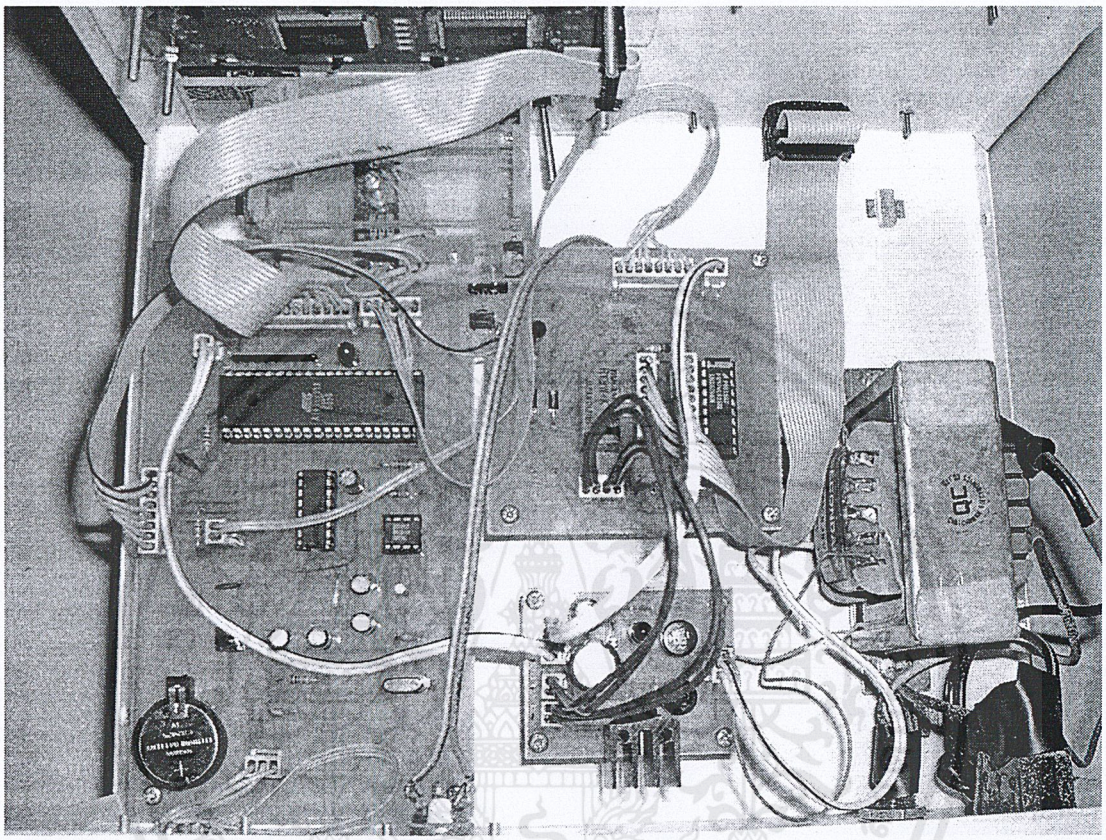
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.19 แสดงโฟลว์ชาร์ทของ event การคำนวณเงินเดือน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

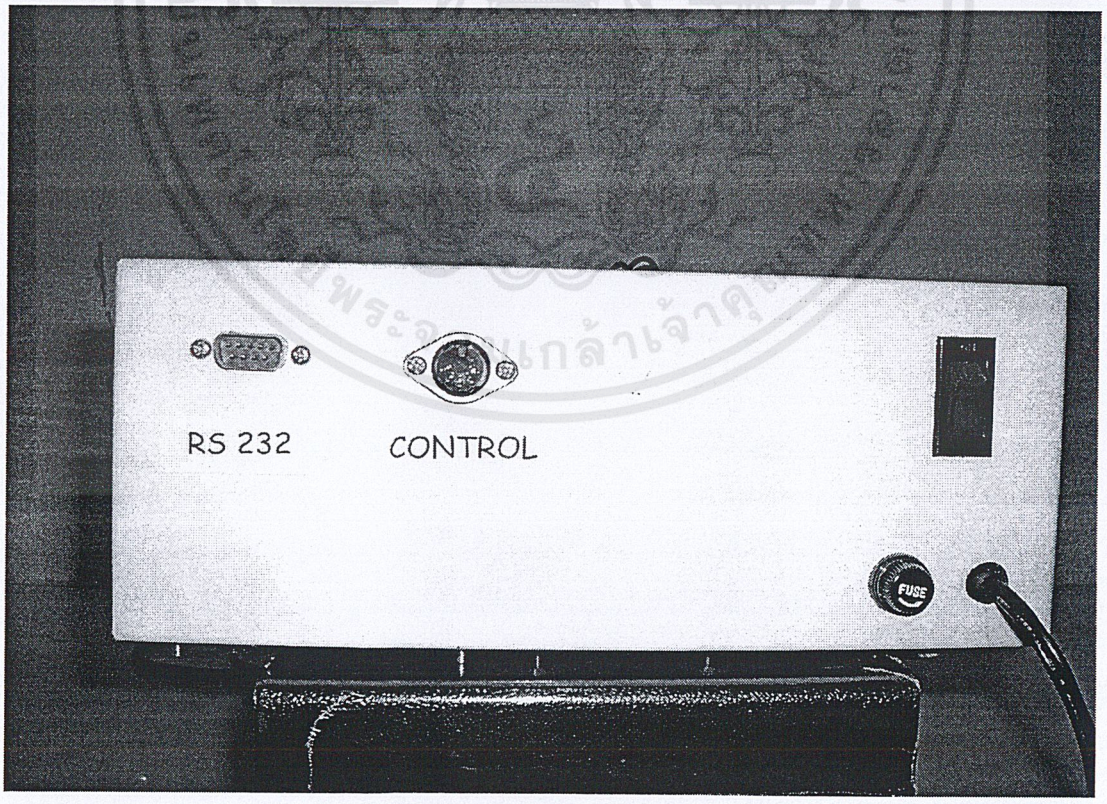
จากการออกแบบในส่วนต่าง ๆได้ทำการประกอบวงจรลงบนแผ่นปริ๊นท์เอนกประสงค์ก่อนเพื่อทดลองทดสอบแก้ไขปรับปรุงวงจรก่อนประกอบทำเครื่องต้นแบบ ซึ่งวงจรและลายวงจรพิมพ์ต่าง ๆ ได้แทรกไว้ในภาคผนวกแล้ว



รูปที่ 3.20 แสดงเครื่องต้นแบบของเครื่องอ่านเขียนบัตรสมาร์ทการ์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.21 แสดงส่วนหน้าของเครื่องต้นแบบ



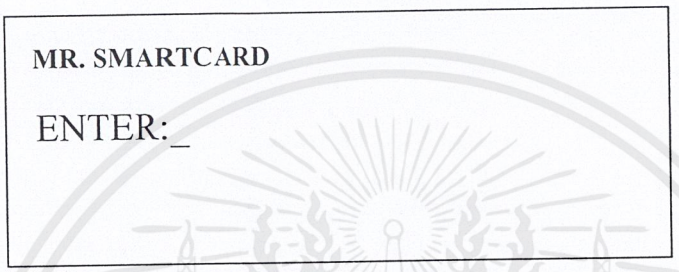
รูปที่ 3.22 แสดงส่วนหลังของเครื่องต้นแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดลองและผลการทดลอง

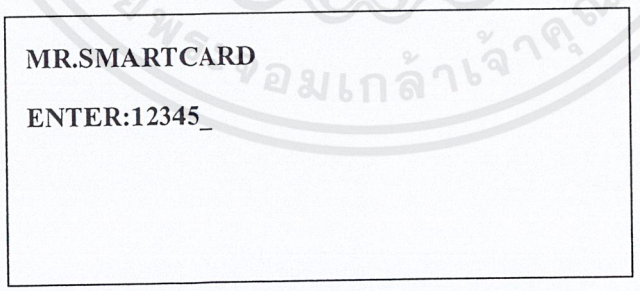
4.1 ส่วนของการแสดงผลทาง LCD โมดูล

4.1.1 เป็นผลของการทดสอบการใช้งานของ LCD โมดูล โดยทำการส่งข้อมูลไปยัง LCD โมดูล เพื่อแสดงผล



รูปที่ 4.1 แสดงผลการทำงานของ LCD โมดูล

4.1.2 เป็นการทดสอบการใช้งานของคีย์บอร์ด โดยจะทำงานร่วมกับ LCD โมดูล คือ ไมโครคอนโทลเลอร์จะรับค่าจากคีย์บอร์ดแล้วค่าที่อ่านได้มาแสดงผลที่ LCD โมดูล



รูปที่ 4.2 แสดงผลที่ได้จากการกดคีย์บอร์ด

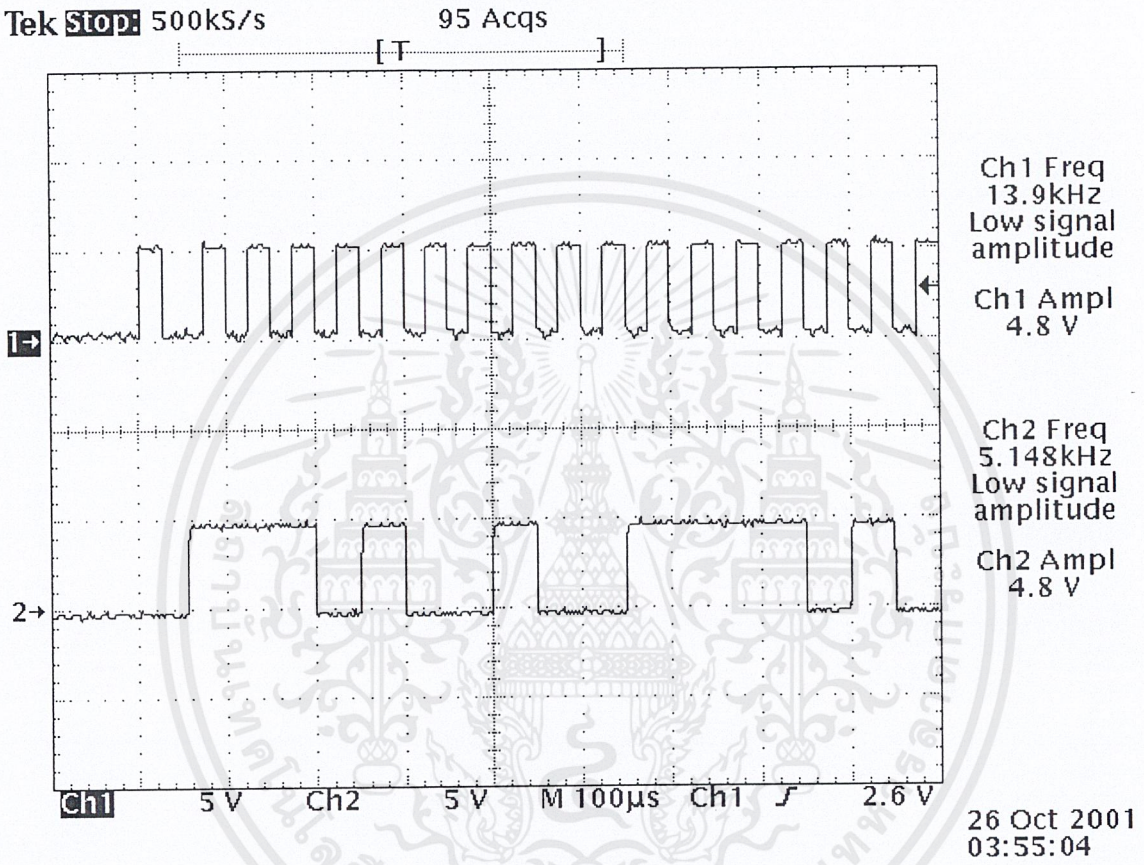
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการทดลองได้ใช้บัตรโทรศัพท์สาธารณะ (TOT Card) ขององค์การโทรศัพท์แห่งประเทศไทย มูลค่า 50, 100 และ 200 บาท

4.1 ส่วนของการอ่านข้อมูลจากบัตร

4.1.1 วัดผลการทดลองของสัญญาณนาฬิกาที่สร้างจากไมโครคอนโทรลเลอร์

4.1.2 วัดผลการทดลองของสัญญาณข้อมูลจากบัตร

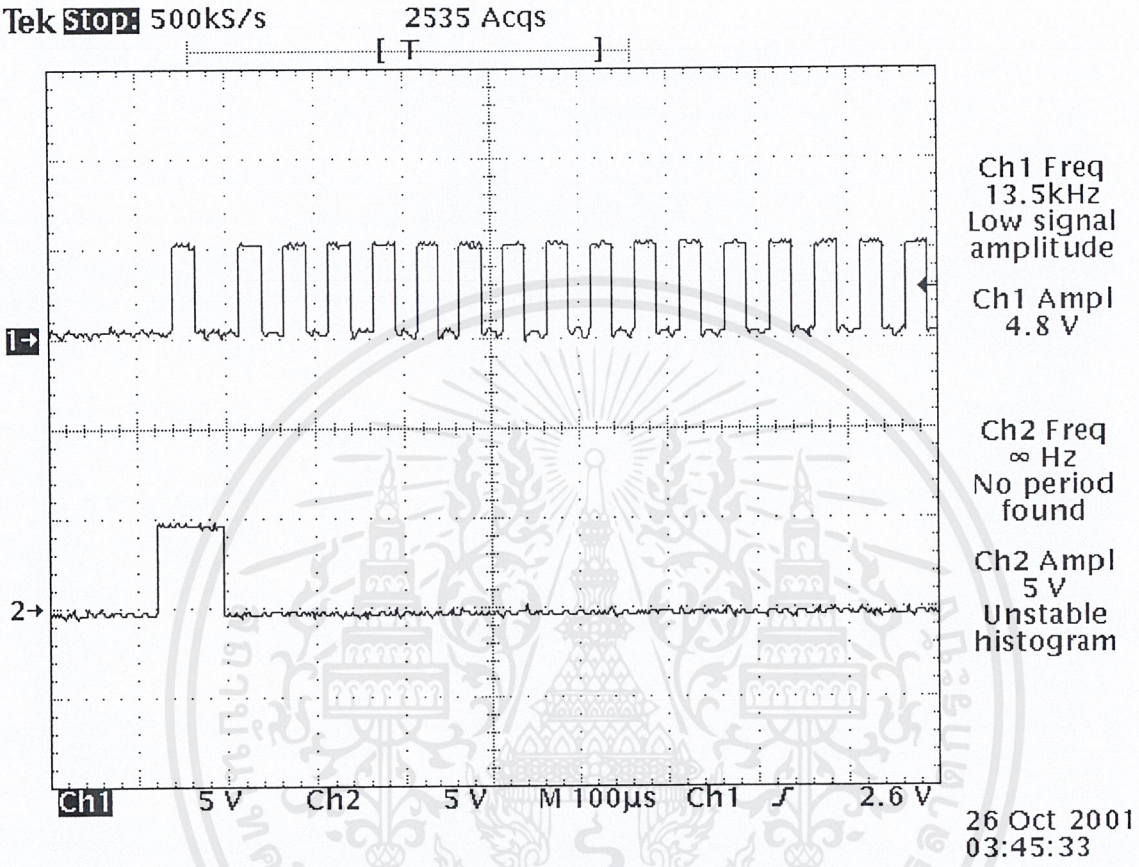


รูปที่ 4.3 แสดงผลการทดลองการวัดสัญญาณนาฬิกาและสัญญาณข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.3 วัดผลการทดลองของสัญญาณนาฬิกาที่สร้างจากไมโครคอนโทรลเลอร์

4.1.4 วัดผลการทดลองของสัญญาณรีเซ็ทที่สร้างจากไมโครคอนโทรลเลอร์

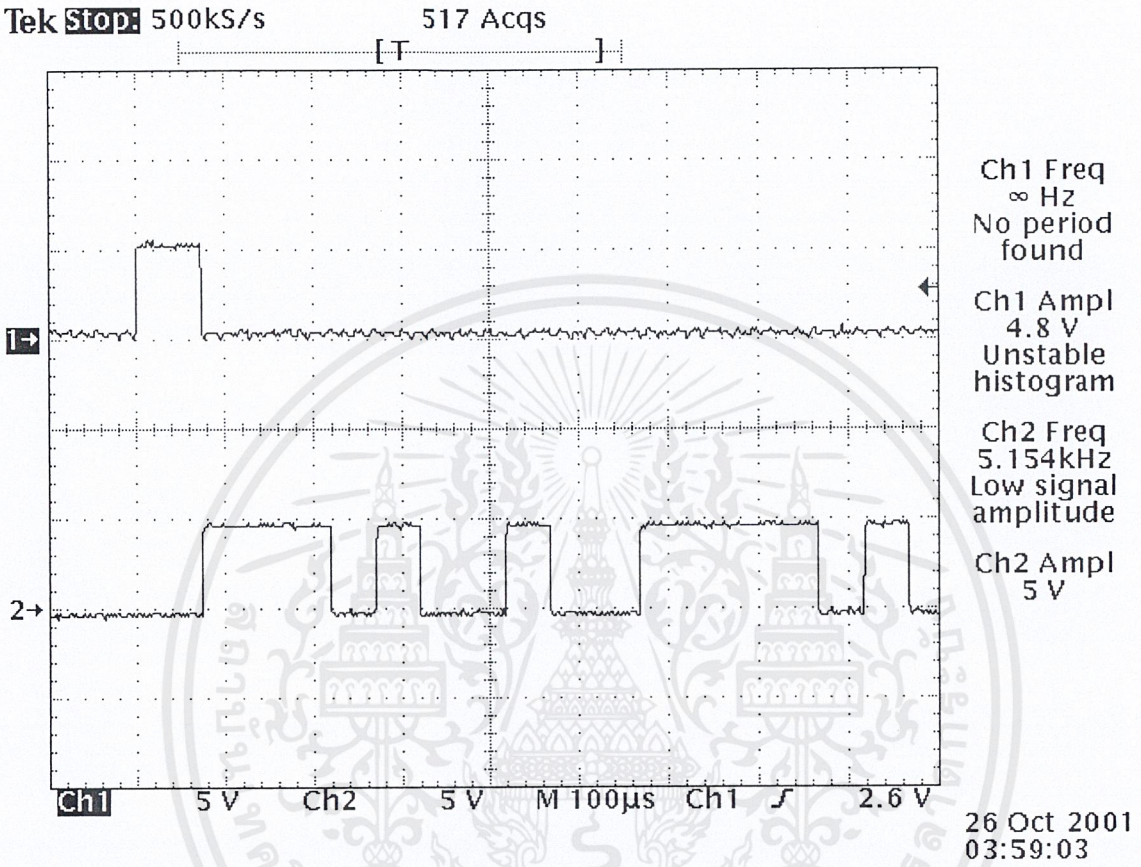


รูปที่ 4.4 แสดงผลการทดลองสัญญาณนาฬิกาและสัญญาณรีเซ็ท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.5 วัดผลการทดลองของสัญญาณรีเซ็ทที่สร้างจากไมโครคอนโทรลเลอร์

4.1.6 วัดผลการทดลองของสัญญาณข้อมูลที่อ่านได้จากบัตร



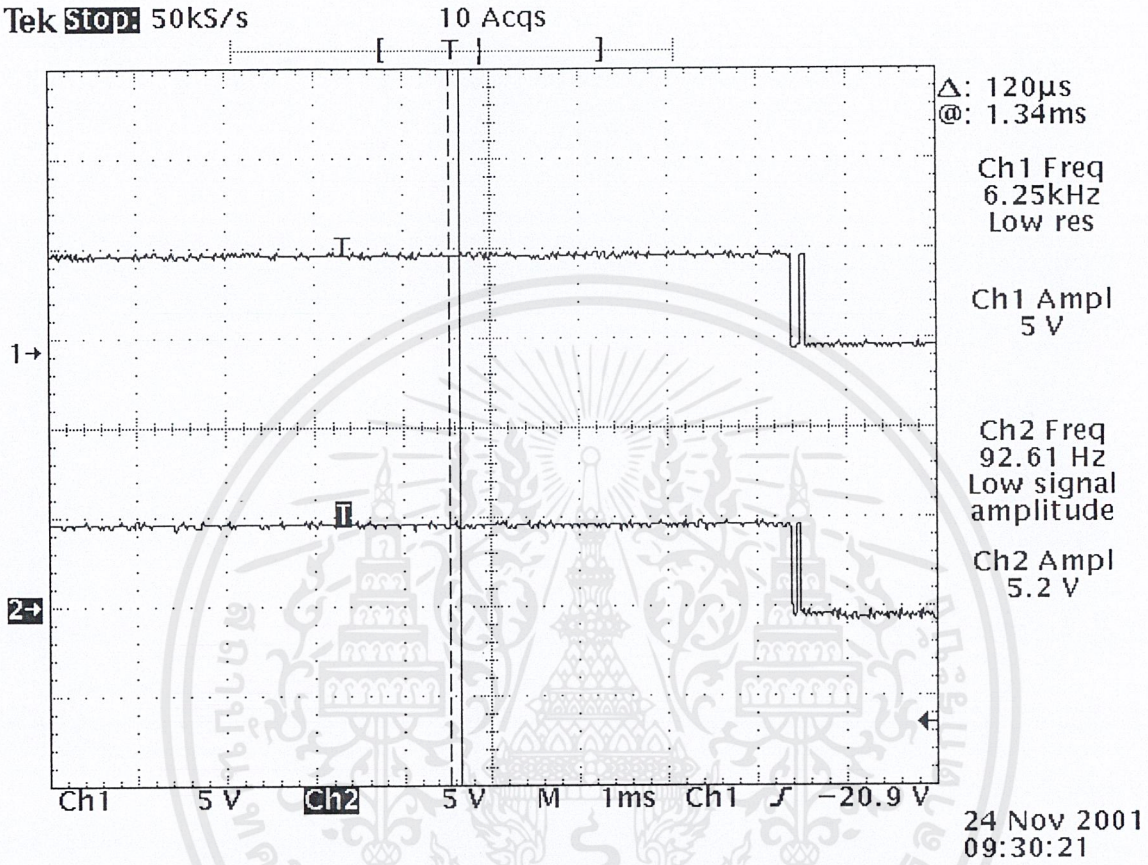
รูปที่ 4.5 แสดงผลการทดลองการวัดสัญญาณรีเซ็ทและสัญญาณข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ส่วนของการเขียนข้อมูลลงในบัตร

4.2.1 วัดผลการทดลองของสัญญาณนาฬิกาที่สร้างขึ้นจากไมโครคอนโทรลเลอร์

4.2.2 วัดผลการทดลองของสัญญาณข้อมูลที่อ่านจากบัตร



รูปที่ 4.6 แสดงการวัดสัญญาณนาฬิกาขณะทำการเขียนข้อมูลลงในบัตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 ผลจากการอ่านข้อมูลจากบัตร TOT Card จำนวน 12ใบ

**********1**********

97 BC 26 12 09 04 23 46 00 00 00 3F 03 FF
00 32
FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF FF
FF FF FF FF FF FF FF FF 54 F1 AA E8 68 11
14 92
FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF FF

**********2**********

97 BC 29 10 15 06 27 86 00 00 00 00 80 FF
00 C8
FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF
FF FF FF FF FF FF FF FF B9 61 2B F5 45 11
14 66
FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF FF

**********3**********

97 BC 27 10 44 02 32 14 00 00 00 00 E0 FF
00 64
FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF FF
FF FF FF FF FF FF FF FF 8A D0 66 5D 8B 11
14 78
FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF FF

**********4**********

97 BC 24 12 13 01 68 43 00 00 00 00 00 FF
00 64
FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF
FF FF FF FF FF FF FF FF 00 10 7D 5F 99 11
14 92
FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF FF

**********5**********

97 BC 29 11 64 00 44 74 00 00 00 00 C0 FF
00 64
FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF FF
FF FF FF FF FF FF FF FF 34 C4 A5 97 F6 11
14 8B
FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF FF

**********6**********

97 BC 32 11 41 01 55 10 00 00 00 00 C0 FF
00 64
FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF FF
FF FF FF FF FF FF FF FF A3 BB AB DB 3E 11
14 88
FF FF FF FF FF FF FF FF FF FF FF FF FF FF
FF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**********7**********

97 BC 28 11 98 02 58 41 00 00 00 00 FC FF 00 64

FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF

FF FF FF FF FF FF FF FF 52 39 D9 3D 33 11 14 91

FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF

**********8**********

97 BC 28 11 59 01 96 14 00 00 00 00 00 FF 00 00

FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF

FF FF FF FF FF FF FF FF 00 00 00 00 00 00 00 00

FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF

**********9**********

97 BC 23 10 89 04 52 87 00 00 00 00 00 FF 00 64

FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF

FF FF FF FF FF FF FF FF 77 01 65 22 3C 11 14 82

FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF

**********10**********

97 BC 27 10 15 02 20 74 00 00 00 00 80 FF 00 C8

FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF

FF FF FF FF FF FF FF FF B3 A1 91 50 AF 11 14 66

FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF

**********11**********

97 BC 27 11 26 02 02 70 00 00 00 00 F0 FF 00 64

FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF

FF FF FF FF FF FF FF FF 1F 89 5C A6 8F 11 14 88

FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF

**********12**********

97 BC 30 10 52 01 17 66 00 00 00 00 00 FF 00 32

FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF

FF FF FF FF FF FF FF FF 31 AC 63 57 36 11 14 7B

FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการอ่านข้อมูลจากบัตร

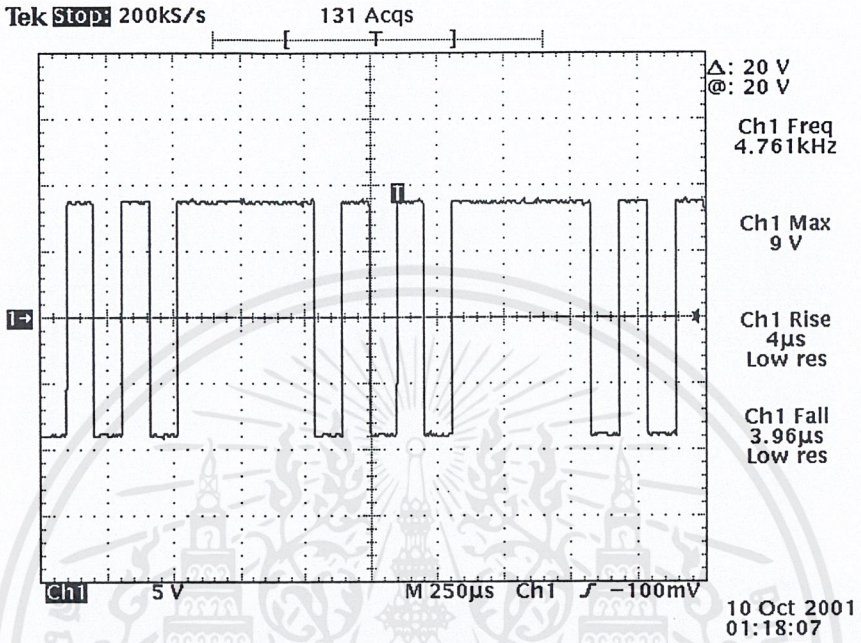
เป็นการแสดงให้รู้ว่า บัตรทุกใบมี serial number และส่วนของ user data ที่ไม่เหมือนกัน โดยเราสามารถนำข้อมูลทั้ง 2 ส่วนมาใช้ประโยชน์ได้

ตัวอย่าง บัตรที่ 1 มี serial number เป็น 12 09 04 23 46 และมี user data เป็น 54 F1 AA E8 68 11 14 92 ซึ่งแตกต่างจากบัตรที่ 2 ที่มี serial number เป็น 10 15 06 27 86 และมี user data เป็น B9 61 2B F5 45 11 14 66

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

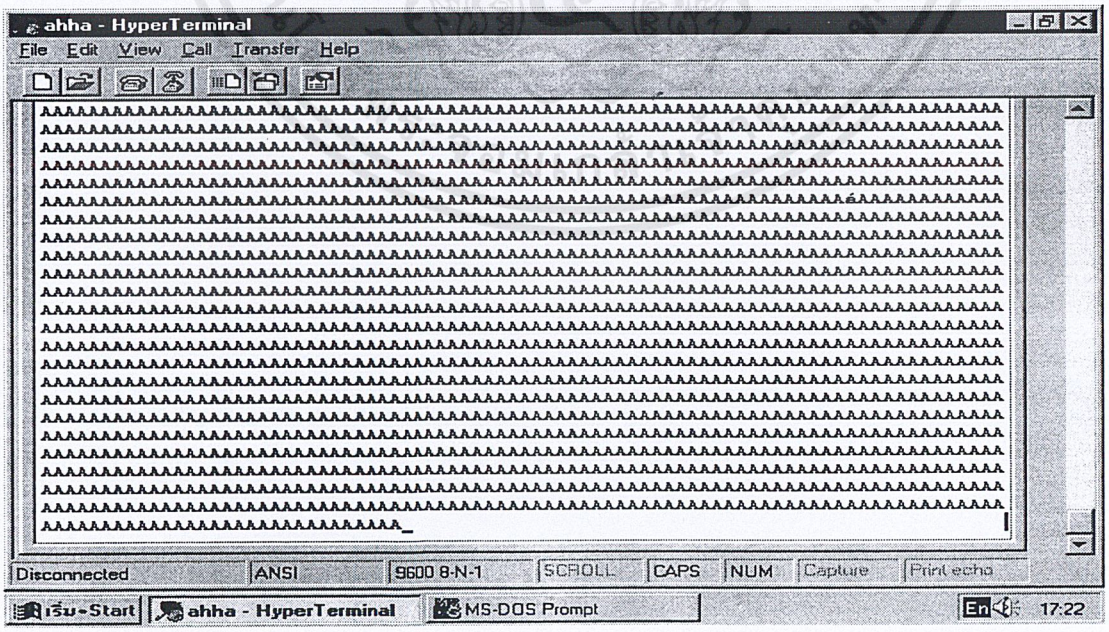
4.4 ส่วนที่ทำการติดต่อกับคอมพิวเตอร์

4.4.1 สัญญาณที่ได้คือสัญญาณจากไอซี MAX 232 เป็นสัญญาณในการส่งข้อมูลเข้าสู่คอมพิวเตอร์ผ่านทางพอร์ทอนุกรม โดยทำการส่งตัว A ไปให้คอมพิวเตอร์



รูปที่ 4.7 แสดงผลการทดลองการส่งสัญญาณเข้าคอมพิวเตอร์

4.4.2 การทดสอบเบื้องต้นว่าคอมพิวเตอร์นั้นสามารถรับข้อมูลที่ส่งมาทางพอร์ตอนุกรมได้ถูกต้อง โดยจะใช้โปรแกรม Hyper Terminal ที่มีในวินโดว์สอยู่แล้วมาใช้งาน และข้อมูลที่ส่งมาคือตัว A

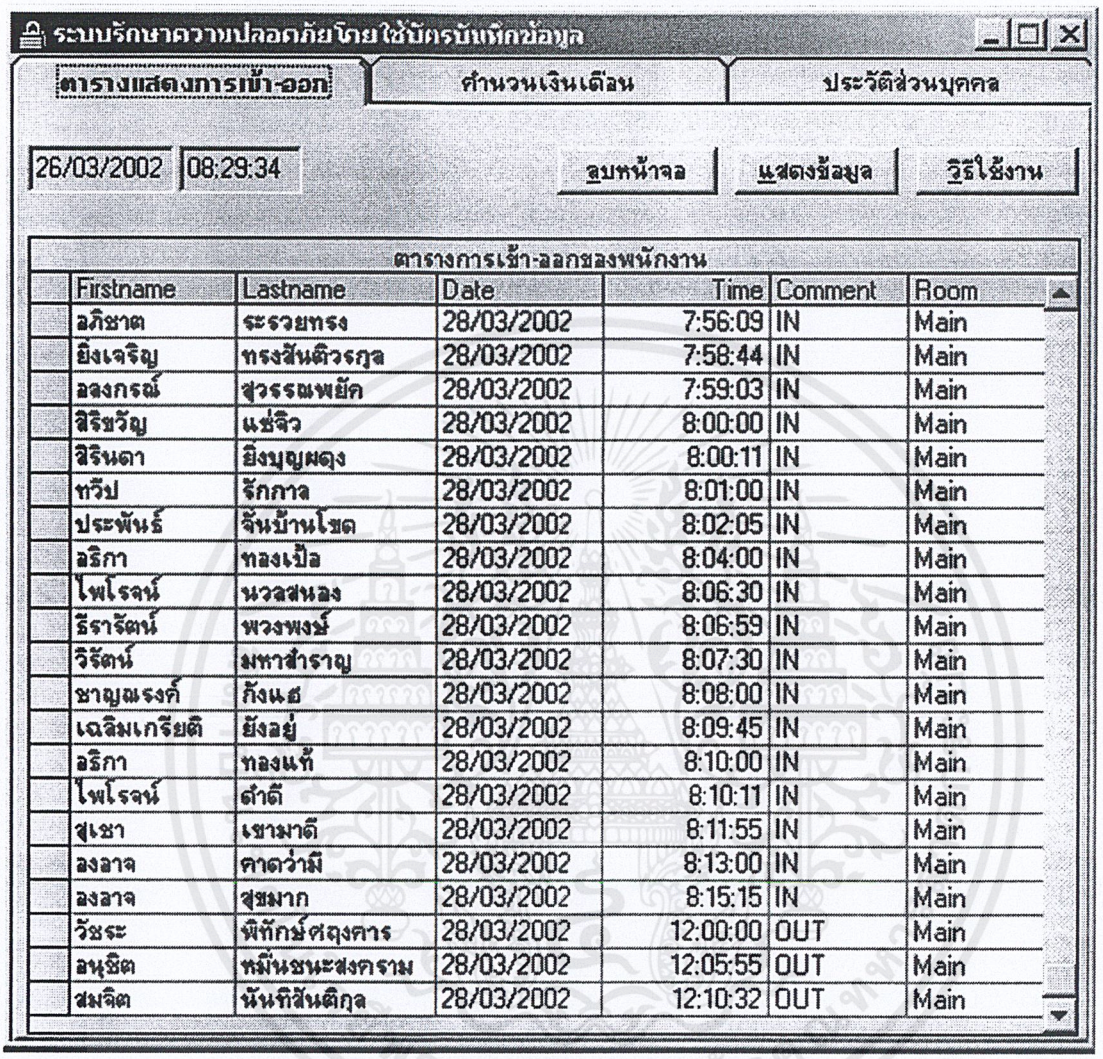


รูปที่ 4.8 แสดงผลที่คอมพิวเตอร์รับข้อมูลเข้ามาได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 ส่วนแสดงผลทางคอมพิวเตอร์

4.5.1 แสดงผลการทำงานของหน้าจอหลัก

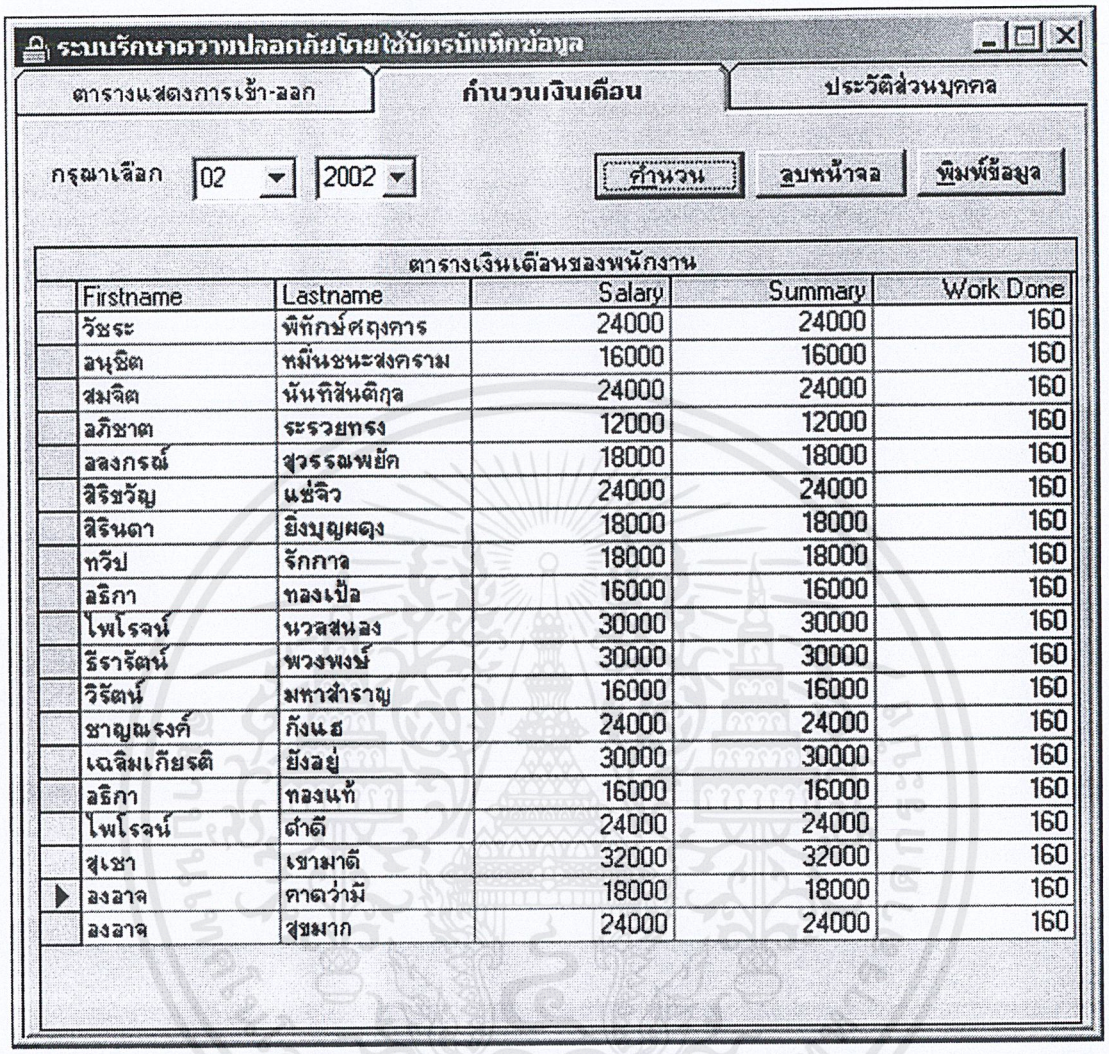


Firstname	Lastname	Date	Time	Comment	Room
อภิชาต	ระรวยทรง	28/03/2002	7:56:09	IN	Main
ยิงเจริญ	ทรงสันติวรกุล	28/03/2002	7:58:44	IN	Main
ลลงกรณ์	สุวรรณเพยัค	28/03/2002	7:59:03	IN	Main
สิริชวัญ	แซ่จิว	28/03/2002	8:00:00	IN	Main
สิรินดา	ยิงบุญผดุง	28/03/2002	8:00:11	IN	Main
ทวีป	รักกาล	28/03/2002	8:01:00	IN	Main
ประพันธ์	จันบ้านโขด	28/03/2002	8:02:05	IN	Main
อธิกา	ทองเปื้อ	28/03/2002	8:04:00	IN	Main
โพโรจน์	นวลสนอง	28/03/2002	8:06:30	IN	Main
ธีราธัตน์	พวงพงษ์	28/03/2002	8:06:59	IN	Main
วิรัตน์	มหาสำราญ	28/03/2002	8:07:30	IN	Main
ชาญณรงค์	กังแธ	28/03/2002	8:08:00	IN	Main
เฉลิมเกรียติ	ยังอยู่	28/03/2002	8:09:45	IN	Main
อธิกา	ทองแท้	28/03/2002	8:10:00	IN	Main
โพโรจน์	ดำดี	28/03/2002	8:10:11	IN	Main
สุเชา	เชามาดี	28/03/2002	8:11:55	IN	Main
ลงอาจ	คาดว่ามี	28/03/2002	8:13:00	IN	Main
ลงอาจ	สุขมาก	28/03/2002	8:15:15	IN	Main
วัชระ	พิทักษ์ศตุงศาร	28/03/2002	12:00:00	OUT	Main
อนุชิต	หมื่นชนะสงคราม	28/03/2002	12:05:55	OUT	Main
สมจิต	นันทิสันติกุล	28/03/2002	12:10:32	OUT	Main

รูปที่ 4.9 แสดงผลการทำงานของหน้าจอหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.2 แสดงผลการคำนวณเงินเดือนของพนักงาน

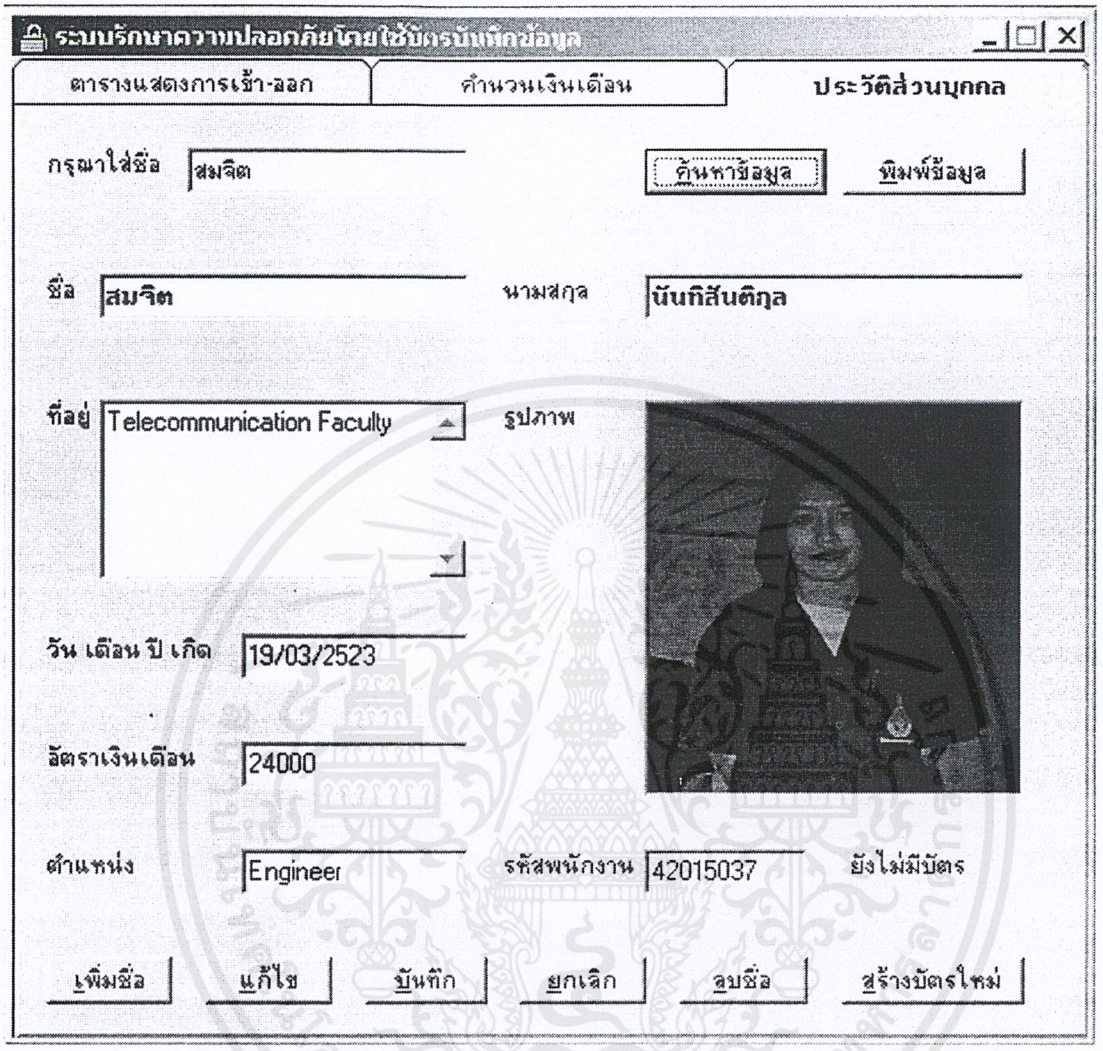


Firstname	Lastname	Salary	Summary	Work Done
วัชระ	พิทักษ์ศฤงคาร	24000	24000	160
อนุชิต	ทมิฬชนะสงคราม	16000	16000	160
สมจิต	นันทิสันติกุล	24000	24000	160
อภิชาต	ระรวยทรง	12000	12000	160
อลงกรณ์	ชุวรรณพยัค	18000	18000	160
สิริชวัญ	แซ่จิว	24000	24000	160
สิรินดา	ยิ่งบุญผดุง	18000	18000	160
ทวีป	รักกาล	18000	18000	160
อธิกา	ทองเปื้อ	16000	16000	160
โพโรจน์	นวลสนอง	30000	30000	160
ธีรารัตน์	พวงพงษ์	30000	30000	160
วิรัตน์	มหาสำราญ	16000	16000	160
ชาญณรงค์	กังแฮ	24000	24000	160
เฉลิมเกียรติ	ยังอยู่	30000	30000	160
อธิกา	ทองแท้	16000	16000	160
โพโรจน์	ดำดี	24000	24000	160
ชูเชา	เขามาดี	32000	32000	160
▶ องอาจ	คาดว่ามี	18000	18000	160
องอาจ	ชูชมาก	24000	24000	160

รูปที่ 4.10 แสดงผลการคำนวณเงินเดือนของพนักงานในแต่ละเดือน

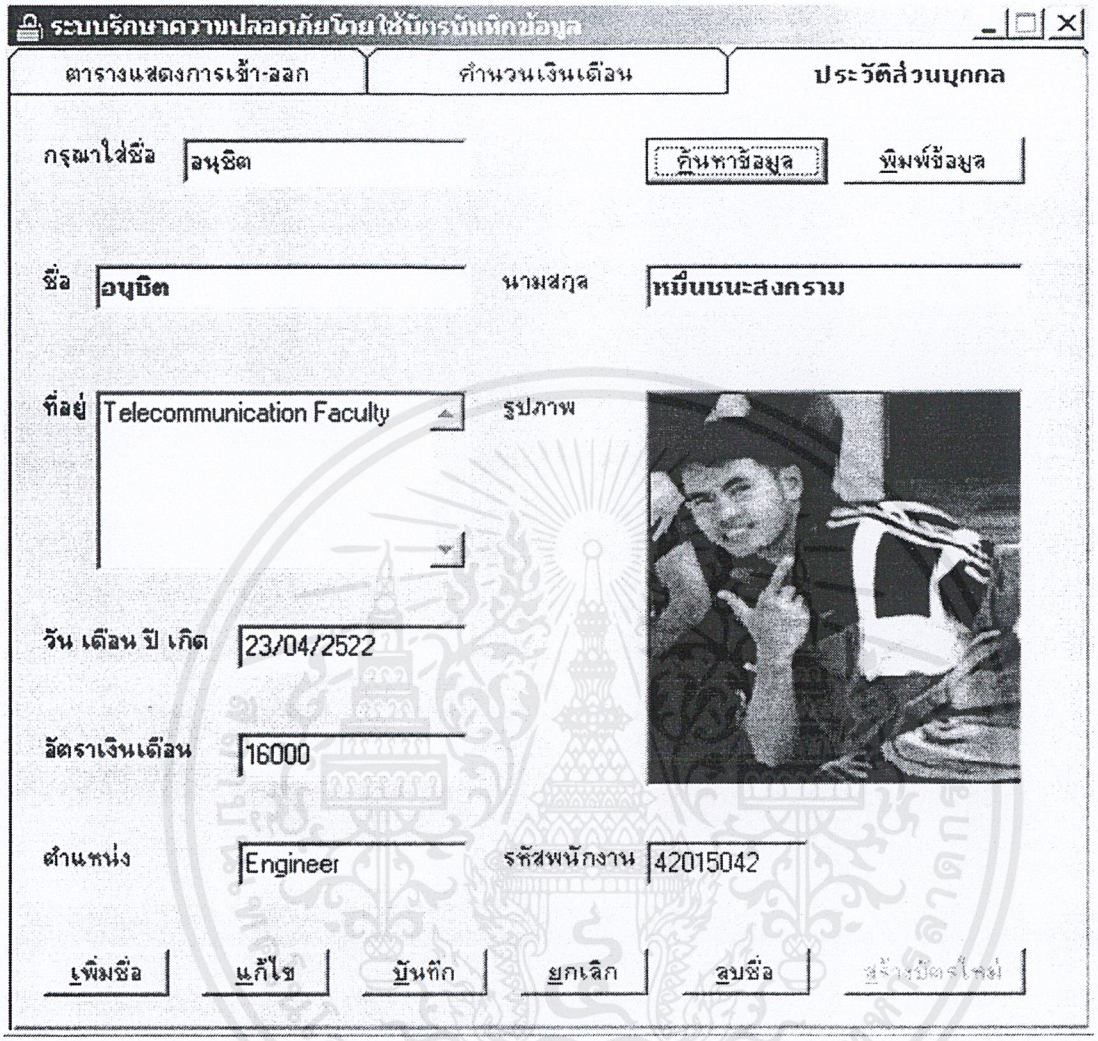
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.3 แสดงผลการการค้นหาพนักงาน



รูปที่ 4.11 แสดงผลการค้นหารายชื่อของพนักงานโดยบุคคลนั้นยังไม่มีบัตร

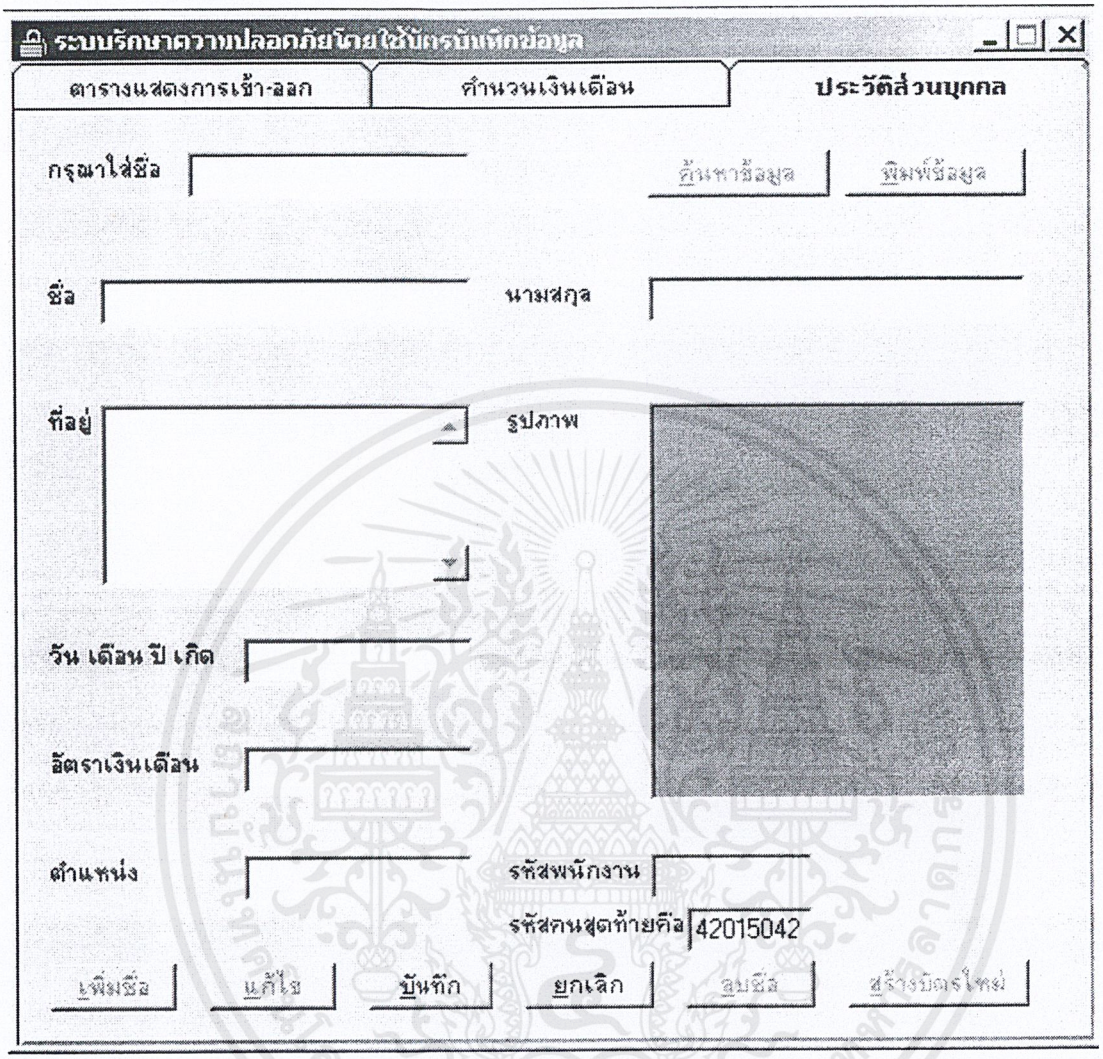
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.12 แสดงผลการค้นหารายชื่อของพนักงานโดยบุคคลนั้นมีบัตรแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

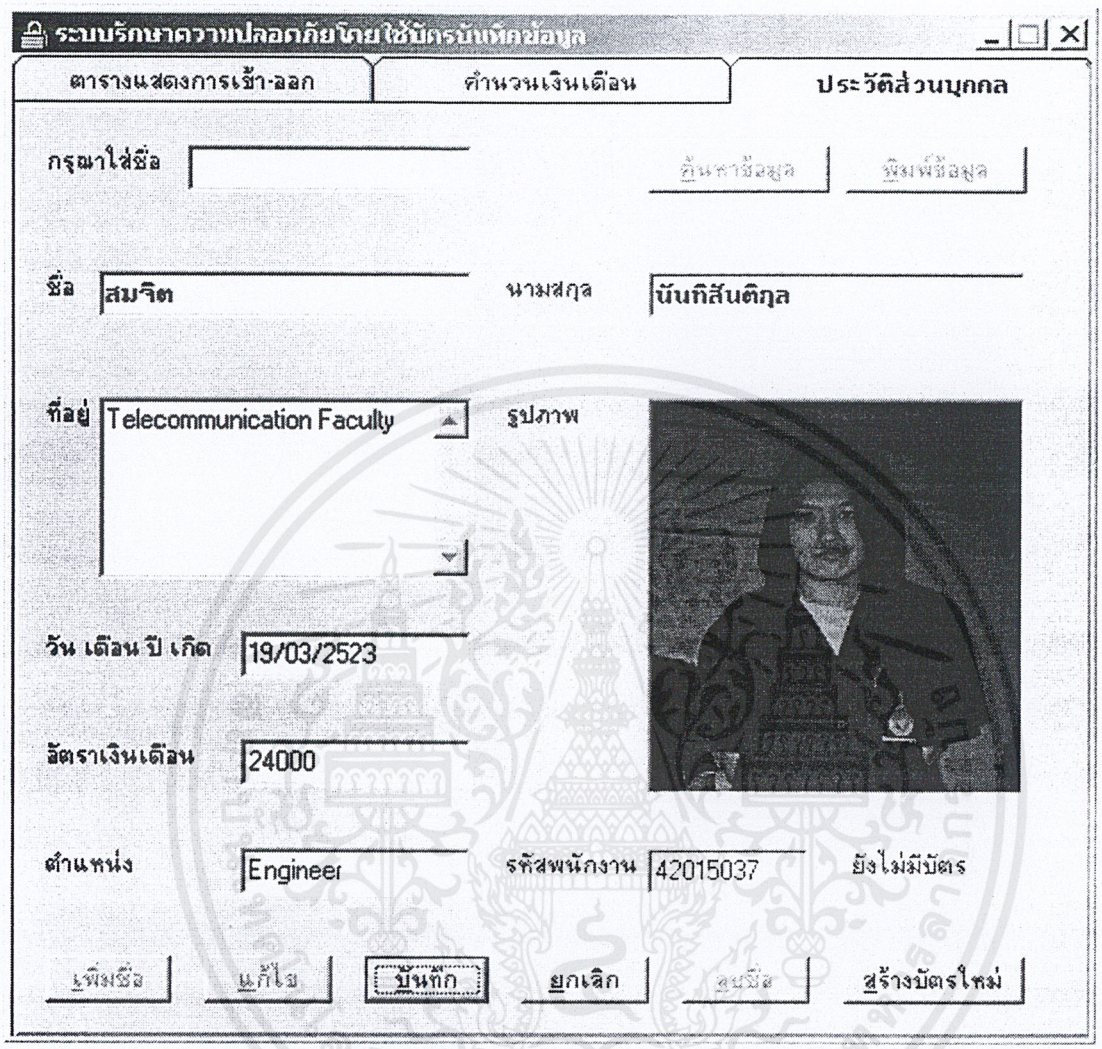
4.5.4 แสดงการเพิ่มชื่อพนักงาน



รูปที่ 4.13 แสดงการเพิ่มรายชื่อของพนักงาน

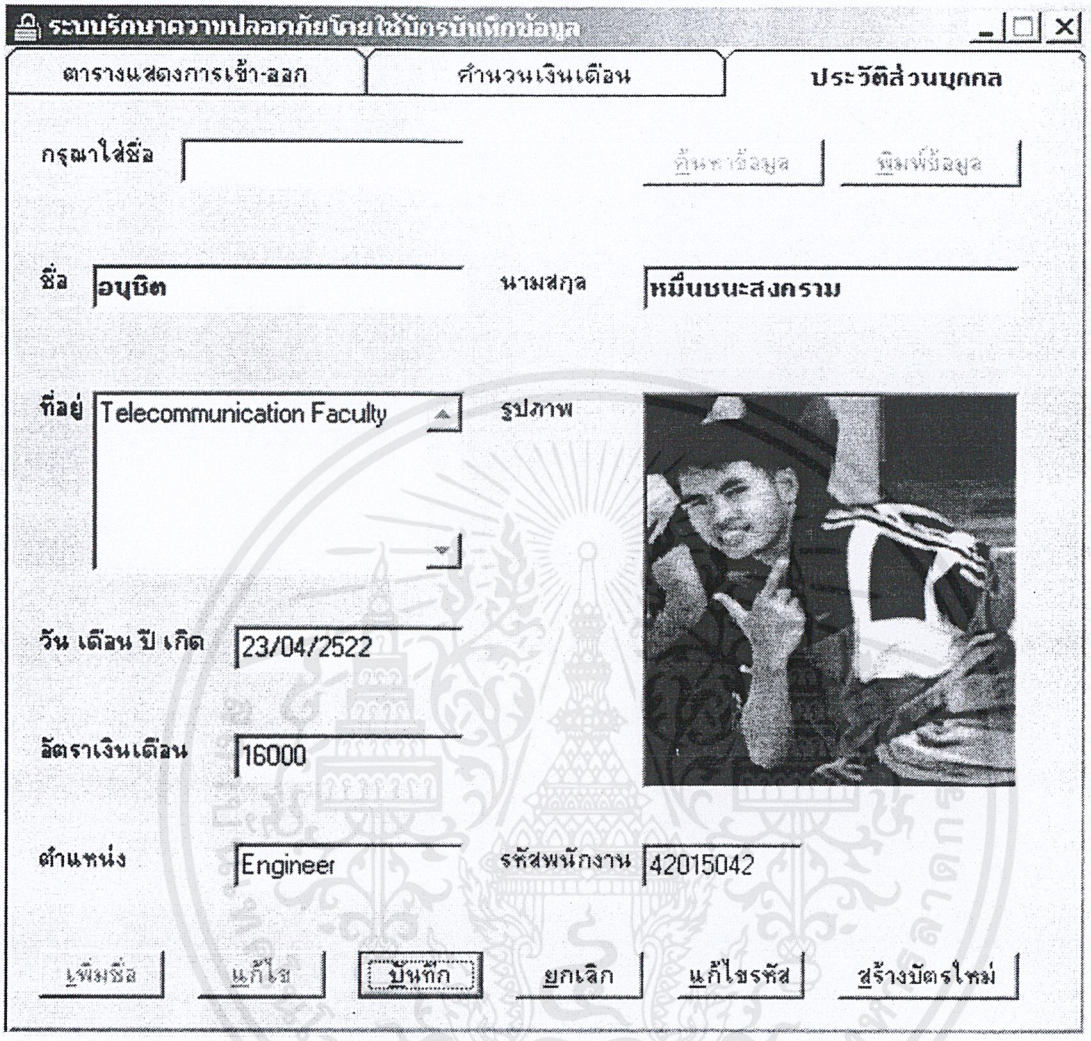
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.5 แสดงผลของการสร้างหรือกำหนดบัตรบันทึกข้อมูลให้แก่พนักงาน



รูปที่ 4.14 แสดงการสร้างหรือกำหนดบัตรบันทึกข้อมูลให้แก่พนักงานที่ยังไม่มีบัตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.15 การสร้างหรือกำหนดบัตรบันทึกข้อมูลให้แก่พนักงานที่มีบัตรแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทวิจารณ์และบทสรุป

5.1 สรุปผลการทดลอง

- ส่วนของการอ่านบัตร

ในส่วนของการอ่านข้อมูลจากบัตรที่ได้ทำการออกแบบและทำการสร้าง เมื่อนำมาทดลองและทดสอบเครื่อง โดยนำบัตรโทรศัพท์สาธารณะ (TOT Card) มาทำการทดสอบ ซึ่งได้ผลการอ่านรหัสข้อมูลได้ถูกต้อง

- ส่วนของการเขียนข้อมูลลงในบัตร

ในส่วนนี้ไม่สามารถเขียนข้อมูลลงในบัตรได้ทุกพื้นที่หน่วยความจำ เนื่องจากบัตรได้ถูกทำการป้องกันไว้โดยผู้ผลิตบัตร การเขียนข้อมูลลงในบัตรสามารถเขียนได้บางพื้นที่ที่มีข้อมูลเก่าอยู่ก่อนแล้ว โดยการเข้าถึงพื้นที่หน่วยความจำจะเป็นแบบบิท ซึ่งสามารถทำการเปลี่ยนบิทจากบิทที่เป็น “1” ให้เป็น “0” ได้เท่านั้น

- ส่วนแสดงผลทางคอมพิวเตอร์

เนื่องจากโปรแกรม Visual Basic เป็นโปรแกรมแบบ Event-Driven คือจะต้องมีเหตุการณ์เกิดขึ้นโปรแกรมถึงจะทำงาน ดังนั้นเราจึงทำการวนลูปรอรับค่าจากพอร์ตอนุกรมตลอดเวลา ซึ่งหากว่ามีข้อมูลเข้าก็จะนำไปตีความหมายต่อไป แต่ในกรณีที่หลังจากคลิกโหมดของการสร้างบัตร (ปุ่มสร้างบัตรใหม่) นั้นจะไม่มีการรับค่าใด ๆ ในกรณีที่ผู้ใช้ ใช้งานเครื่องอ่านบัตรในโหมดของผู้ใช้

5.2 วิจารณ์ผลการทดลอง

โครงงานชิ้นนี้เป็นการสร้างเครื่องอ่านและเขียนบัตรสมาร์ตการ์ด (บัตร TOT นั่นเอง) โดยมีลักษณะคล้ายกับการนำเอาบัตรแถบแม่เหล็กมาเป็นบัตรผ่านก่อนที่จะเข้าภายในอาคารต่าง ๆ ที่ใช้อยู่ในปัจจุบัน จุดเด่นของโครงงานนี้คือระบบจะสามารถบันทึกเวลาที่ทุกคนผ่านเข้า-ออกอาคารในรูปของฐานข้อมูลซึ่งเราสามารถนำไปตรวจสอบหรือนำไปประมวลผลในด้านที่ต้องการได้ และจุดเด่นอีกประการคือเราสามารถนำบัตรโทรศัพท์ที่ใช้ไม่ได้ (มูลค่าของเงินหมดแล้ว) นำมาใช้ประโยชน์อีกได้ (Reuse) เป็นการประหยัดทรัพยากรอย่างหนึ่ง

5.3 แนวทางพัฒนา

แนวทางพัฒนาคือโครงการนี้ยังสามารถใช้ได้แค่เครื่องอ่าน-เขียนบัตรสมาร์ตการ์ดกับคอมพิวเตอร์ได้เครื่องต่อเครื่อง (Stand alone) ควรพัฒนาต่อให้สามารถเชื่อมต่อกันเป็นโครงข่ายโดยมีคอมพิวเตอร์เครื่องเดียวเป็นตัวเก็บฐานข้อมูล หากทำเช่นนี้ได้ก็จะเป็นประโยชน์อย่างมาก เช่น เราสามารถนำเครื่องอ่านบัตรสมาร์ตการ์ดไว้ที่แต่ละห้องภายในอาคาร เราก็จะสามารถทำการกำหนดสิทธิของผู้ใช้ว่าจะเข้าห้องใดได้บ้าง เข้าตอนไหนบ้าง เป็นต้น

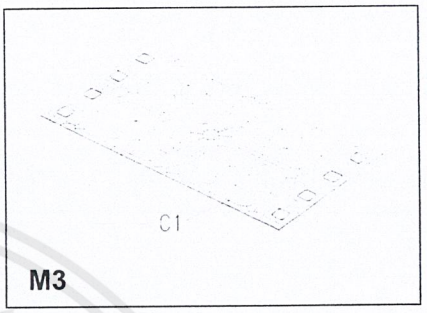
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Intelligent 221–Bit EEPROM Counter for > 20000 Units with Security Logic and High Security Authentication

Features



M3

- **221 bit EEPROM and 16 bit mask-programmable ROM**

  104 bit user memory fully compatible with SLE 4406/06E
  - 64 bit Identification Area consisting of
    - 16 bit Manufacturer code (mask-programmable ROM)
    - SLE 4436:
      8 bit Manufacturer data, card issuer dependent (ROM)
      40 bit for personalization data of card issuer (PROM)
    - SLE 4436E:
      48 bit for personalization data of card issuer (PROM)
  - 40 bit Counter Area including 1 bit for personalization (PROM/EEPROM)

  133 bit additional memory for advanced features
  - 4 bit Counter Backup (anti-tearing flags)
  - 1 bit initiation flag for Authentication Key 2
  - 16 bit Data Area 1 for free user access
  - 48 bit Authentication Key 1
  - either 48 bit Data Area 2 for user defined data
    or 48 bit Authentication Key 2
  - 16 bit Data Area 3 for free user access
- **Counter with up to 33352 count units fully compatible with SLE 4406/06E**
  - Five stage abacus counter
  - Due to testing purposes a maximum of 21064 count units is guaranteed
- **Counter tearing protection**
  - Backup feature activated at choice
- **High security authentication unit**
  - Random number as challenge
  - Individual secret Authentication Key 1
  - Optional individual secret Authentication Key 2
  - Calculation of up to 16 bit response
  - Calculation of a 16 bit response within 30 ms at a clock frequency of 100 kHz
- **Transport Code protection for delivery**
- **EEPROM security cells in sensitive areas**
- **Chip circuitry and chip layout optimised for high security against physical and electrical signal analysis**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Features (cont'd)

- **Ambient temperature –35 ... +80°C**
- **Supply voltage 5 V ± 10 %**
- **Supply current < 5 mA**
- **EEPROM programming time 5 ms**
- **ESD protection typical 4000 V**
- **Endurance minimum 10^5 write/erase cycles / bit[1)]**
- **Data retention for minimum of 10 years[1)]**
- **Contact configuration and Answer-to-Reset (synchronous transmission) in accordance to standard ISO/IEC 7816**

Table 1 Ordering Information

Type	Package[2)]	Access of 3rd byte
SLE 4436 M3	M3	Data of 3rd byte are programmed by Infineon exclusively
SLE 4436 C	C	
SLE 4436E M3	M3	Data of 3rd byte are programmed by the card manufacturer at personalisation
SLE 4436E C	C	

1) Values are temperature dependent
2) Available as a wire-bonded module (M3) for embedding in plastic cards or as a die (C) for customer packaging

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Description

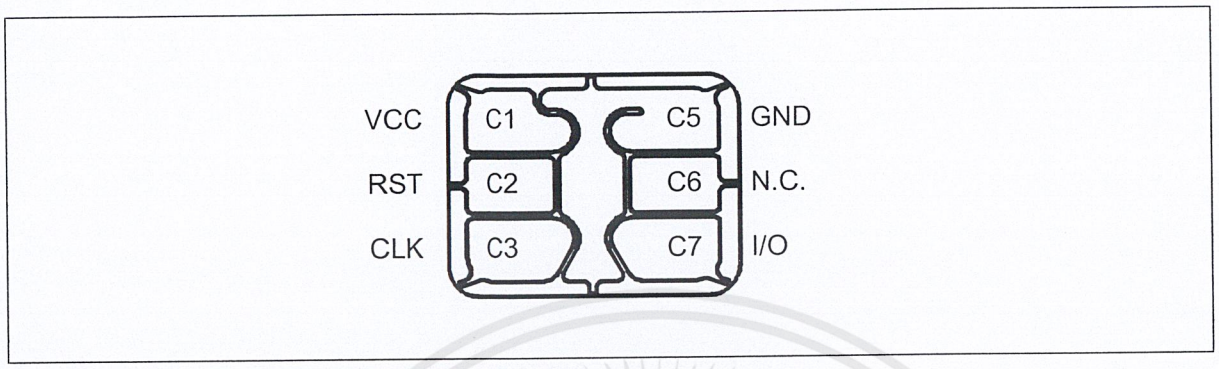


Figure 1 Pin Configuration Wire-bonded Module (top view)

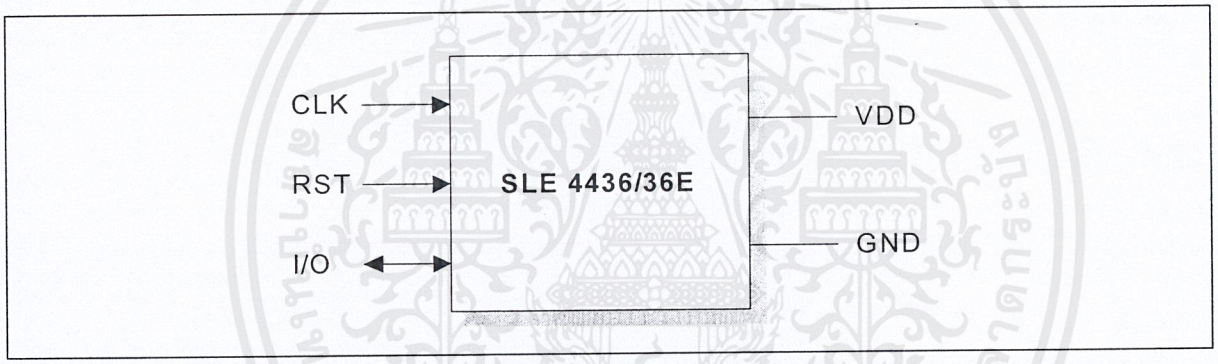


Figure 2 Pad Configuration Die

Table 2 Pin Definitions and Functions

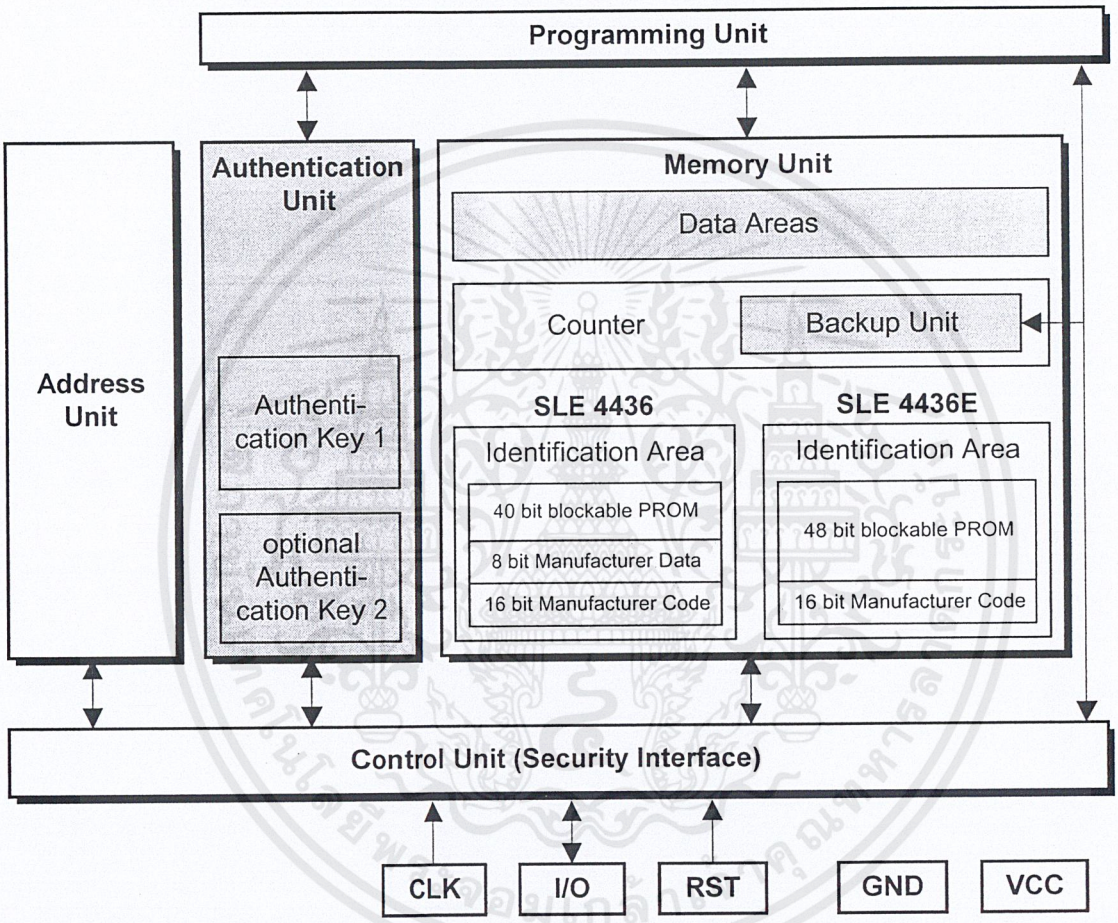
Card Contact	Symbol	Function
C1	VCC	Supply voltage
C2	RST	Control input (Reset Signal)
C3	CLK	Clock input
C5	GND	Ground
C6	N.C.	Not connected
C7	I/O	Bi-directional data line (open drain)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

และการแสวงหากำไร ทั้งนี้ ยังห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

General Description

SLE 4436/36E is designed for applications in prepaid telephone cards. The chip consists of an EEPROM memory of 221 bit, a ROM of 16 bits, a control/security unit and a special computing unit for chip authentication. The shaded blocks in the block diagram (Figure 3) contain the enhanced features of SLE 4436/36E compared to SLE 4406/06E.



Legend
Shaded blocks - Enhanced features of SLE 4436/36E compared to SLE 4406/06E
Blocks marked by cross hatch - Defined by chip manufacturer

Figure 3 Block Diagram

- **Memory Unit**
  Counter, Identification Data (e.g. serial number, expiry date) and Data Areas.
- **Address Unit**
  Setting of the address counter is synchronously with the CLK.
- **Programming Unit**
  The programming voltage for the EEPROM/PROM is generated internally.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

แม้ว่าก็รณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **Backup Unit**
  An associated backup bit indicates an interrupt caused by e.g. tearing a card out of a reader without mechanical locking device during a reloading cycle of a devaluated counter stage.
- **Authentication Unit**
  The secret algorithm offers a challenge & response procedure for individual card authentication; the optional activation of cipher block chaining allows the certification of a counter decreasing procedure.
- **Security Interface**
  Ensures a minimum and a maximum frequency and proper logical voltage levels.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FAIRCHILD SEMICONDUCTOR™

October 1987
Revised April 2001

MM74C922 • MM74C923
16-Key Encoder • 20-Key Encoder

General Description

The MM74C922 and MM74C923 CMOS key encoders provide all the necessary logic to fully encode an array of SPST switches. The keyboard scan can be implemented by either an external clock or external capacitor. These encoders also have on-chip pull-up devices which permit switches with up to 50 kΩ on resistance to be used. No diodes in the switch array are needed to eliminate ghost switches. The internal debounce circuit needs only a single external capacitor and can be defeated by omitting the capacitor. A Data Available output goes to a high level when a valid keyboard entry has been made. The Data Available output returns to a low level when the entered key is released, even if another key is depressed. The Data Available will return high to indicate acceptance of the new key after a normal debounce period; this two-key roll-over is provided between any two switches.

An internal register remembers the last key pressed even after the key is released. The 3-STATE outputs provide for easy expansion and bus operation and are LPTTL compatible.

Features

- 50 kΩ maximum switch on resistance
- On or off chip clock
- On-chip row pull-up devices
- 2 key roll-over
- Keybounce elimination with single capacitor
- Last key register at outputs
- 3-STATE output LPTTL compatible
- Wide supply range: 3V to 15V
- Low power consumption

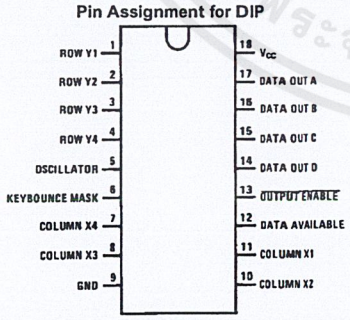
Ordering Code:

Order Number	Package Number	Package Description
MM74C922WM	M20B	20-Lead Small Outline Integrated Circuit (SOIC), JEDEC MS-013, 0.300" Wide
MM74C922N	N18B	18-Lead Plastic Dual-In-Line Package (PDIP), JEDEC MS-001, 0.300" Wide
MM74C923WM	M20B	20-Lead Small Outline Integrated Circuit (SOIC), JEDEC MS-013, 0.300" Wide
MM74C923N	N20A	20-Lead Plastic Dual-In-Line Package (PDIP), JEDEC MS-001, 0.300 Wide

Device also available in Tape and Reel. Specify by appending suffix letter "X" to the ordering code.

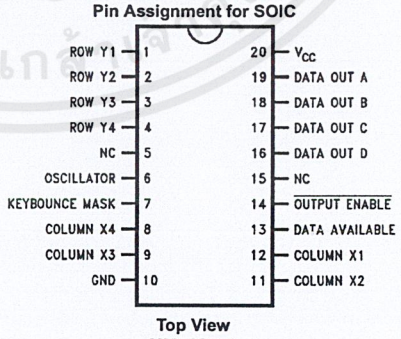
Connection Diagrams

Pin Assignment for DIP



Top View
MM74C922

Pin Assignment for SOIC



Top View
MM74C922

© 2001 Fairchild Semiconductor Corporation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DALLAS SEMICONDUCTOR

DS1307/DS1308
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 with oscillator running
- Optional industrial temperature range –40°C to +85°C (IND) available for DS1307 and DS1308
- DS1307 available in 8–pin DIP or SOIC
- DS1308 available in 36–pin SMD BGA (Ball Grid Array)
- DS1308 accuracy is better than ±2 minute/month at 25°C

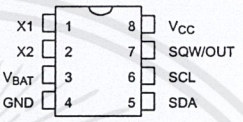
ORDERING INFORMATION

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)
DS1308	36–pin BGA
DS1308N	36–pin BGA(IND)

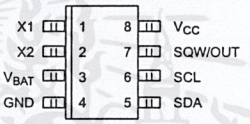
DESCRIPTION

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.

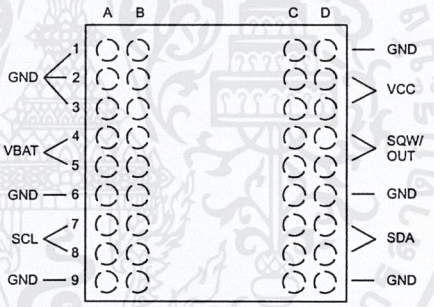
PIN ASSIGNMENT



DS1307 8–PIN DIP (300 MIL)



DS1307Z 8–PIN SOIC (150 MIL)



DS1308
36–PIN SMD BGA
(TOP VIEW)

PIN DESCRIPTION DS1307/DS1308

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

DS1308 PIN IDENTIFIER

V_{CC}	– C2, C3, D2, D3
V_{BAT}	– A4, A5, B4, B5
SDA	– C7, C8, D7, D8
SCL	– A7, A8, B7, B8
SQW/OUT	– C4, C5, D4, D5
GND	– All Remaining Balls

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี ขอขอบพระคุณ ผศ.ดร.สุทธิชัย นพนาคีพงษ์ เป็นอย่างสูงที่ได้ให้คำแนะนำ ความอนุเคราะห์และความไว้วางใจในการให้ใช้อุปกรณ์และเครื่องมือในการสร้างผลงานนี้อย่างดียิ่ง ในการทำปริญญานิพนธ์ฉบับนี้ ตลอดจนคณาจารย์ทุกท่านที่ได้ให้การประสิทธิ์ประสาธน์วิชาการแก่ผู้จัดทำ และที่สำคัญยิ่งขอขอบพระคุณคุณพ่อและคุณแม่ที่ได้เป็นกำลังใจอันสำคัญทำให้ผลงานชิ้นนี้สามารถสำเร็จลุล่วงได้อย่างสมบูรณ์

คณะผู้จัดทำ

1. นายวัชระ พิทักษ์ศฤงคาร
2. นางสาวสมจิต นันทิสันติกุล
3. นายอนุชิต หมื่นชนะสงคราม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนังสืออ้างอิง

1. รศ.สมยศ จุณณะปิยะ, "การใช้งานไมโครคอนโทรลเลอร์ ตระกูล MCS-51", สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, กรุงเทพ, 2541
2. W.Rang, W.Effing, "Smart Card Handbook", Second Edition.,John Wiley & Sons,LTD., 2000
3. สุทธิศักดิ์ พงศ์ธนาพานิชน์, "Visual Basic5.0 Professional การใช้คำสั่งและคอนโทรล ActiveX", บริษัทซีเอ็ดยูเคชั่น จำกัด (มหาชน), 2541

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้