สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การประยุกต์ใช้งานสมาร์ตการ์ดในระบบอำนวยความสะดวกสถานที่จอดรถ

Smart card application for parking system

โดย

นาย เอกพจ สันติเวฒน์รัตน์

นาย เอกภาพ ศรีบุศยกุล

เลขหมู่.....
เลขทะเบียน.....71957
วัน,เดือน,ปี.....- 6 มิ.ย. 2550

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

ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การประยุกต์ใช้งานสมาร์ตการ์ดในระบบอำนวยความสะดวกสถานที่จอดรถ

Smart card application for parking system

โดย

นาย เอกพล สันติเวชน์รัตน์ รหัส 46010992

นาย เอกภาพ ศรีบุศยกุล รหัส 46010994

อาจารย์ที่ปรึกษา

ผศ. พลผดุง ผดุงกุล

ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงงานเรื่อง การประยุกต์ใช้งานสมาร์ตการ์ดในระบบอำนวยความสะดวกสถานที่จอดรถ

Smart card application for parking system

จัดทำโดย นายเอกพล สันติเวสน์รัตน์ รหัส 46010992

นายเอกภาพ ศรีบุศยกุล รหัส 46010994

อาจารย์ที่ปรึกษา ผศ. พลผดุง ผดุงกุล

ปริญญานิพนธ์ฉบับนี้ได้ผ่านการตรวจสอบโดยอาจารย์ที่ปรึกษาแล้ว

ลงชื่อ.....

(ผศ. พลผดุง ผดุงกุล)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การประยุกต์ใช้งานสมาร์ตการ์ดในระบบอำนวยความสะดวกสถานที่จอดรถ

นาย เอกพล สันติเวสน์รัตน์ รหัส 46010992
นาย เอกภาพ ศรีบุศยกุล รหัส 46010994
ผ.ศ. พลผดุง ผดุงกุล อาจารย์ที่ปรึกษา
ภาคการศึกษาที่ 2 ปีการศึกษา 2549

บทคัดย่อ

โครงงานนี้ได้ใช้บัตรสมาร์ตการ์ดมาทำเป็นบัตรของระบบอำนวยความสะดวกสถานที่จอดรถ โดยเป็นบัตรสมาร์ตการ์ดแบบหน่วยความจำชนิด SLE 4442 ในส่วนของเครื่องอ่านบัตรใช้ไมโครคอนโทรลเลอร์ mcs51 โดยเราสามารถเขียน และอ่านข้อมูลลงภายในบัตรได้ และควบคุมการส่งข้อมูล จากเครื่องอ่านสมาร์ตการ์ดมายังเครื่องคอมพิวเตอร์ ด้วยการสื่อสารแบบอนุกรม ตามมาตรฐาน RS232 โดยโครงงานนี้มีการต่อเป็นเน็ตเวิร์คโดยใช้ไอซี MAX3082 ตามมาตรฐาน RS485 ข้อมูลที่ได้จะมาวิเคราะห์ในฐานข้อมูลที่มีอยู่โดยใช้โปรแกรม Microsoft visual basic 6.0 และโปรแกรม Microsoft Access ในการออกแบบ และสร้างโปรแกรม ที่ทำการวิเคราะห์ฐานข้อมูลดังกล่าว รวมทั้งการแสดงรายละเอียดผลการวิเคราะห์ผ่านทางคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Smart card application for parking system

Mr.Eakphon Santivesrat 46010992

Mr.Eakapap Sributsayakul 46010994

Asst. Prof. Polpadung Padungkul

Academic year 2006

Abstract

This project utilizes discarded smart card to make facility for parking system. Smart card is SLE4442 memory smart card. Smart card reader was used Microcontroller MCS51 to read and write data in card and control data transfer from smartcard reader to computer by using RS232,serial port standard in this project connect in network by using IC MAX 3082 with RS485 standard. The data analysis is using Microsoft visual basic 6.0 and Microsoft Access and show the analysis result through computer interface.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี ก็ด้วยความเมตตาของ ผศ.พลผดุง ผดุงกุล ที่ท่านได้ให้คำปรึกษาอย่างเป็นกันเอง เอื้ออำนวยเรื่องอุปกรณ์การทดลอง และห้องทำงานอันสะดวกสบาย

และโดยส่วนตัวของพวกกระผม ก็ขอขอบคุณครอบครัวของพวกผมที่ได้ให้ การสนับสนุนกำลังทรัพย์โดยตลอดมาในการทำโครงงานที่ไม่จำกัด อีกทั้งยังคอยเป็นกำลังใจตลอดเวลาในการทำงานจนปริญญานิพนธ์ ฉบับนี้ สำเร็จลุล่วงไปได้ด้วยดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อ	I
Abstract	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูป	VIII
สารบัญตาราง	XII
บทที่ 1 บทนำ	1
บทที่ 2 สมาร์ตการ์ด	3
2.1 ความหมายของสมาร์ตการ์ด	3
2.2 ประวัติความเป็นมาของสมาร์ตการ์ด	3
2.3 ข้อดีของสมาร์ตการ์ด	4
2.4 ส่วนประกอบและโครงสร้างของสมาร์ตการ์ด	5
2.5 องค์ประกอบในการใช้งานสมาร์ตการ์ด	7
2.5.1 ตัวบัตรและตัวชิป	7
2.5.2 สมาร์ตรีเดอร์	8
2.5.3 ซอฟแวร์	8
2.5.4 Black office	8
2.6 ประเภทของสมาร์ตการ์ด	9
2.6.1 Memory Card	9
2.6.2 Optical Memory Card	10
2.6.3 Processor Card	10
2.6.4 Contact less Smart Card	11
2.6.5 Com-Bi Card	12
2.6.6 Hybrid Smart Card	13
2.7 รูปแบบของสมาร์ตการ์ดที่นำมาใช้ในโครงงาน	14
2.8 รูปแบบชิบสมาร์ตการ์ SLE4442	17
2.9 คุณสมบัติทั่วไปของ SLE4442	18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10 รูปแบบการสื่อสารข้อมูลของสมาร์ตการ์ด SLE4442 20
บทที่ 3 การสื่อสารแบบอนุกรม 26
3.1 การสื่อสารแบบอนุกรม 26
3.2 การสื่อสารข้อมูลแบบอะซิงโครนัส 26
3.3 มาตรฐานพอร์ตอนุกรมแบบ RS-232 29
3.3.1 คอนเน็คเตอร์สำหรับพอร์ต RS-232 และการเชื่อมต่อ 29
3.3.2 UART (Universal Asynchronous Receiver Transmitter) 32
3.3.3 ชนิดของ UART 33
3.4 วงจรภายในและรีจิสเตอร์ของพอร์ตอนุกรม RS-232 33
3.5 ลักษณะสัญญาณอินพุตและเอาต์พุตของพอร์ตRS-232 41
3.6 แอดเดรสของพอร์ตอนุกรม 42
บทที่ 4 มาตรฐานการสื่อสารแบบ RS 485 44
4.1 ตัวส่งแบบไม่สมดุลย์ (Unbalanced line driver) 44
4.2 ตัวส่งแบบสมดุลย์ (Balance line driver) 44
4.3 ตัวรับแบบสมดุลย์ (Balanced line receiver) 45
4.4 การส่งข้อมูลตามมาตรฐาน RS-485 45
4.5 การควบคุม tri-state ของอุปกรณ์ RS-485 46
4.6 การควบคุมการส่งข้อมูลของอุปกรณ์ RS-485 46
4.7 การต่อขั้วปลาย (Termination) 47
4.8 การไบอัสในระบบเครือข่าย RS-485 (Biasing an RS-485 Network) 48
บทที่ 5 ไมโครคอนโทรลเลอร์ MCS-51 49
5.1 ไมโครคอนโทรลเลอร์ MCS-51 49
5.2 โครงสร้างภายในของไมโครคอนโทรลเลอร์ MCS-51 49
5.3 โครงสร้างหน่วยความจำภายในไมโครคอนโทรลเลอร์ MCS-51 51
5.4 หน่วยความจำโปรแกรม 51
5.5 หน่วยความจำข้อมูล 52
5.6 รีจิสเตอร์ที่เกี่ยวกับไมโครคอนโทรลเลอร์MCS-51 54
5.6.1 รีจิสเตอร์หน้าที่พิเศษ 54
5.6.2แอกคิวมูเลเตอร์ (ACCUMULATOR) 55

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.7ตัวดำเนินการในภาษาซี 55
5.8 ประโยคควบคุมในภาษาซี 57
5.9 การทำซ้ำ 57
5.10 อาร์เรย์ พอยน์เตอร์ และสตรักเจอร์ 58
5.11 ไทเมอร์/เคาน์เตอร์ของไมโครคอนโทรลเลอร์ MCS-51 58
5.12 พอร์ตอนุกรมของไมโครคอนโทรลเลอร์ MCS-51 59
บทที่ 6 Data Base & Visual Basic 60
6.1 Data Base 60
6.1.1 ระบบฐานข้อมูล (Data Base System) 60
6.1.2 ระบบแฟ้มข้อมูล (File System) 61
6.1.3 องค์ประกอบของระบบฐานข้อมูล 63
6.1.4 ประโยชน์ของฐานข้อมูล 66
6.1.5 ภาษาทางด้านฐานข้อมูล 66
6.2 Visual Basic 67
บทที่ 7 การออกแบบและหลักการทำงานของวงจร 70
7.1 องค์ประกอบของระบบโดยรวม 70
7.1.1 องค์ประกอบของระบบโดยรวมของเครื่องอ่าน เขียน สมาร์ตการ์ต 71
7.1.2โครงสร้างของเครื่องโปรแกรมสมาร์ตการ์ด 71
7.1.3 หลักการทำงานเครื่องโปรแกรมสมาร์ตการ์ด SLE 4442 72
7.1.4 หลักการทำงานของเครื่องโปรแกรมสมาร์ตการ์ด (P89V51RD2BN) 73
7.2 องค์ประกอบของระบบโดยรวมของเครื่องอ่าน เขียน สมาร์ตการ์ด 74
7.2.1 ชุดวงจรเซ็นเซอร์ 83
7.3 หลักการทำงานโครงงาน 86
บทที่ 8 การทดลองและผลการทดลอง 93
8.1 การทดลอง 93
8.2 การทดลองวัดรูปสัญญาณ ณ ตำแหน่งขาต่างๆในสมาร์ตการ์ด 93
8.3 การทดลองการสั่งการของเครื่องอ่านเขียนสมาร์ตการ์ตผ่าน Hyperterminal 96
8.4 การทดลองการทำงานของระบบผ่านทาง Visual Basic 6.0 99
8.5 การทดลองวัดรูปสัญญาณ ณ ตำแหน่งขาต่างๆในชุด Sensor 102

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 9 วิเคราะห์และสรุปผลการทดลอง 106

บรรณานุกรม 107

ภาคผนวก 108

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

หน้า

รูปที่ 2.1 ส่วนประกอบของสมาร์ตการ์ต 5
รูปที่ 2.2 การแบ่งชนิดของบัตรตามรูปร่างที่นำไปใช้งาน และขนาด 6
รูปที่ 2.3 ส่วนประกอบของสมาร์ตการ์ดโมดูลในสายการผลิตสมาร์ตการ์ด 6
รูปที่ 2.4 ตัวอย่างสมาร์ตการ์ดโมดูล 7
รูปที่ 2.5 การแบ่งสมาร์ตการ์ดตามชนิดของหน่วยความจำและประเภทของหน้าสัมผัส 9
รูปที่ 2.6 บล็อกไดอะแกรมโครงสร้างในชิปสมาร์ตการ์ดชนิด Memory (Synchronous Card) 9
รูปที่ 2.7 บล็อกไดอะแกรมโครงสร้างภายในชิปสมาร์ตการ์ดชนิด Processor (Asynchronous Card) 11
รูปที่ 2.8 สมาร์ตการ์ด ชนิด Memory แบบ Contact less 12
รูปที่ 2.9 สมาร์ตการ์ดชนิดProcessor แบบ Contact less 12
รูปที่ 2.10 โครงสร้างภายในของสมาร์ตการ์ดชนิด Contact แบบ Com-bi card 13
รูปที่ 2.11 โครงสร้างภายในของสมาร์ตการ์ดชนิด Hybrid Smart Card 13
รูปที่ 2.12 บัตร Smart Card SLE 4442 14
รูปที่ 2.13 วิธีทดสอบการบิดงอสมาร์ตการ์ด 15
รูปที่ 2.14 ตำแหน่งหน้าสัมผัสของชิปสมาร์ตการ์ด 16
รูปที่ 2.15 หน้าที่การทำงานของแต่ละหน้าสัมผัส 17
รูปที่ 2.16 หน้าที่การทำงานของแต่ละหน้าสัมผัสสมาร์ตการ์ด SLE 4442 17
รูปที่ 2.17 แสดงหน่วยความจำภายในของ SLE 4442 18
รูปที่ 2.18 บล็อกไดอะแกรมโครงสร้างภายในของ SLE4442 19
รูปที่ 2.19 สัญญาณรีเซต และ ATR 20
รูปที่ 2.20 การส่งคำสั่งไปยังการ์ด (IFD : Interface Device) 21
รูปที่ 2.21 สัญญาณของคำสั่ง Read Main Memory 22
รูปที่ 2.22 สัญญาณคำสั่ง Read Protection Memory 22
รูปที่ 2.23 การลบข้อมูลแล้วเขียนข้อมูลซ้ำ 22
รูปที่ 2.24 การลบหรือเขียนข้อมูล (อย่างใดอย่างหนึ่ง) 23
รูปที่ 2.5 สัญญาณคำสั่ง Read Security Memory 23

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.26 สัญญาณคำสั่ง Compare Verification Data 24
รูปที่ 2.27 Flow Chart ของกระบวนการ PSC 25
รูปที่ 3.1 รูปแบบอย่างง่ายที่สุดของข้อมูลอนุกรม 26
รูปที่ 3.2 รูปแบบอย่างง่ายที่สุดของข้อมูลอนุกรมแบบอะซิงโครนัส 27
รูปที่ 3.3 คอนเนคเตอร์อนุกรม 30
รูปที่ 3.4 การต่ออุปกรณ์ภายนอกกับพอร์ตอนุกรมของคอมพิวเตอร์ในลักษณะต่างๆ 32
รูปที่ 3.5 ไดอะแกรมการทำงานภายในของพอร์ตอนุกรมของเครื่องคอมพิวเตอร์ 34
รูปที่ 3.6 ไดอะแกรมแสดงโครงสร้างทางฮาร์ดแวร์ของพอร์ตอนุกรม 42
รูปที่ 4.1 Balance Differential Output Line Driver 44
รูปที่ 4.2 Balanced Differential Input Line Receiver 45
รูปที่ 4.3 ระบบเครือข่าย two-wire multidrop 46
รูปที่ 4.4 การต่อขั้วปลายแบบ Parallel termination และ AC Coupled Termination 48
รูปที่ 4.5 การต่อความต้านทานไบอัส 48
รูปที่ 5.1 แสดงโครงสร้างพื้นฐานของไมโครคอนโทรลเลอร์ MCS-51 50
รูปที่ 5.2 แสดงรูปร่างและการจัดวางขาต่างๆ ของไมโครคอนโทรลเลอร์ MCS-51 51
รูปที่ 5.3 แสดงการจัดหน่วยความจำข้อมูล 53
รูปที่ 5.4 แสดงการต่อกับหน่วยความจำข้อมูลภายนอกไอซี 54
รูปที่ 6.1 ระบบแฟ้มข้อมูล 60
รูปที่ 6.2 ตัวอย่างการจัดเก็บข้อมูลของผู้ป่วย 63
รูปที่ 6.3 ตัวอย่างการใช้ข้อมูลร่วมกันของแต่ละฝ่าย 64
รูปที่ 6.4 การเขียนโปรแกรมแบบ Visual Programming 68
รูปที่ 6.5 หน้าต่างของโปรแกรมเมื่อทำการสั่งรัน 68
รูปที่ 7.1 โครงสร้างของระบบทั้งหมด 70
รูปที่ 7.2 แสดงโครงสร้างของเครื่องโปรแกรมสมาร์ตการ์ด 71
รูปที่ 7.3 สัญญาณรีเซต และ ATR 73
รูปที่ 7.4 สัญญาณของคำสั่ง Write Memory 73
รูปที่ 7.5 สัญญาณของคำสั่ง Read Memory 74
รูปที่ 7.6 Flow Chart การทำงาน เมื่อเปิดโปรแกรม Hyper Terminal 75
รูปที่ 7.7 Flowchart Reset and Answer to reset บัตร SLE 4442 76

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 7.8 Flow chart โปรแกรม READ บัตรSLE 4442 77
รูปที่ 7.9 Flow chart การตรวจสอบ Programmable Security Code(PSC) 78
รูปที่ 7.10 Flow chart การ Update Main Memory (Write) บัตร SLE 4442 79
รูปที่ 7.11 วงจรเครื่องโปรแกรมสมาร์ตการ์ด SLE 4442 80
รูปที่ 7.12 วงจร Master 81
รูปที่ 7.13 วงจร NE555N ชุดที่ 1 82
รูปที่ 7.14 วงจร NE555N ชุดที่ 2 82
รูปที่ 7.15 ชุดตัวขับ อินฟาเรด (BD 139) 83
รูปที่ 7.16 Block Diagram ของ IR Receiver Modules 83
รูปที่ 7.17 ไอซี ของ Multiplexers และ Demultiplexers 84
รูปที่ 7.18 วงจรตัวรับอินฟาเรด 85
รูปที่ 7.19 วงจรตัวส่งอินฟาเรด 86
รูปที่ 7.20 Flow chart การทำงานของตัวแม่ 88
รูปที่ 7.21 Flow chart การทำงานของเครื่องอ่านสมาร์ตการ์ด 89
รูปที่ 7.22 Flow chart การทำงานของเครื่องตรวจสอบสิ่งกีดขวาง 90
รูปที่ 7.23 Flow chart การทำงานเมื่อทำการเสียบบัตรเมื่อเข้าจอด 92
รูปที่ 7.24 Flow chart การทำงานเมื่อทำการเสียบบัตรเมื่อต้องการนำรถออกจากที่จอด 93
รูปที่ 8.1สัญญาณที่ ATR 94
รูปที่ 8.2สัญญาณตอนเริ่มคำสั่ง 95
รูปที่ 8.3สัญญาณตอนหยุดคำสั่ง 95
รูปที่ 8.4 สัญญาณขณะRead 96
รูปที่ 8.5 สัญญาณช่วง Write 96
รูปที่ 8.6 ผลการทดลองหน้าจอแรกของโปรแกรม เพื่อเตรียมรับบัตรสมาร์ตการ์ด 97
รูปที่ 8.7 ผลการทดลองโปรแกรมเมื่อมีการใส่บัตรสมาร์ตการ์ด 97
รูปที่ 8.8 ผลการทดลองเมื่อต้องการ reset smart card 98
รูปที่ 8.9 ผลการทดลองเมื่อเลือกรายการอ่านข้อมูล 98
รูปที่ 8.10 ผลการทดลองเมื่อเลือกรายการเขียนข้อมูล 99
รูปที่ 8.11 รูปของระบบขณะเสียบบัตรเพื่อนำรถเข้าจอด 100
รูปที่ 8.12 รูปของระบบขณะเสียบบัตรไม่ถูกต้องที่ทางเข้า 100

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 8.13 รูปของระบบขณะเสียบบัตรเพื่อนำรถออกจากที่จอด 101
รูปที่ 8.14 รูปของระบบขณะเสียบบัตรเพื่อนำรถออกจากที่จอดไม่ถูกต้อง 101
รูปที่ 8.15 รูปของระบบเติมเงิน 102
รูปที่ 8.16 รูปของระบบลงทะเบียน 102
รูปที่ 8.17 สัญญาณความถี่ 1kHzจากวงจร555ตัวแรก 103
รูปที่ 8.18 สัญญาณ Pulse ความถี่ 38kHzจากวงจร555ตัวที่สอง 103
รูปที่ 8.19 สัญญาณ Pulseที่เกิดจากการ Modulate จากวงจร555ทั้งสอง 104
รูปที่ 8.20 รูปสัญญาณขณะไม่มีรถจอด 104
รูปที่ 8.21 รูปสัญญาณขณะมีรถจอด 105

สารบัญตาราง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

ตารางที่ 2.1 การเปรียบเทียบการใช้การ์ดชนิดต่างๆของ France Telecom	4
ตารางที่ 2.2 โครงสร้างและความหมายของชุดคำสั่งที่ SLE4442 รองรับ	21
ตารางที่ 3.1 แสดงบิตพาริตี้ของข้อมูล	28
ตารางที่ 3.2 หน้าที่การทำงานของขาอนุกรม	30
ตารางที่ 3.3 ฟังก์ชั่นการทำงานของรีจิสเตอร์ตำแหน่ง 01H	35
ตารางที่ 3.4 ฟังก์ชั่นการทำงานของรีจีสเตอร์ 02H	36
ตารางที่ 3.5 ฟังก์ชั่นการทำงานของรีจีสเตอร์ 03H	37
ตารางที่ 3.6 ฟังก์ชั่นการทำงานของรีจีสเตอร์ 04H	39
ตารางที่ 3.7 ฟังก์ชั่นการทำงานของรีจีสเตอร์ 05H	39
ตารางที่ 3.7 ฟังก์ชั่นการทำงานของรีจีสเตอร์ 06H	40
ตารางที่ 3.8 ข้อมูลในแอดเดส 0000:0411H ที่ใช้แจ้งจำนวนพอร์ตอนุกรม	42
ตารางที่ 6.1 ตัวอย่างโครงสร้างข้อมูลในแฟ้มลูกค้า	62

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1 บทนำ

1.1 ความเป็นมาของโครงงาน

เนื่องจากในปัจจุบัน การพัฒนาในด้านเทคโนโลยีการเก็บข้อมูล และการเชื่อมต่อข้อมูลมีความก้าวหน้าไปอย่างมาก และเป็นที่น่าสนใจในการจะนำมาประยุกต์ใช้ในชีวิตประจำวันให้มากขึ้น โดยเฉพาะอย่างยิ่งเทคโนโลยีการเก็บข้อมูลแบบหนึ่งที่กำลังเป็นที่สนใจเป็นอย่างมากในปัจจุบัน คือ สมาร์ตการ์ด ซึ่งเริ่มจะมีการใช้มากขึ้นในปัจจุบัน จากเหตุผลนี้จึงเป็นที่น่าสนใจในการที่จะศึกษาและนำมาประยุกต์ใช้สมาร์ตการ์ดให้มีบทบาทในชีวิตประจำวันมากขึ้น โดยในโครงงานนี้ได้มุ่งเน้นการประยุกต์ไปในงานด้านระบบความปลอดภัยและในอำนวยความสะดวกให้แก่ผู้ใช้ ซึ่งทั้ง2ด้านนี้เป็นก็ถือมีความสำคัญกับชีวิตประจำวันอย่างมาก สำหรับสมาร์ตการ์ดที่ใช้ในโครงงานนี้ใช้สมาร์ตการ์ดแบบ SLE4442

1.2 วัตถุประสงค์ของโครงงาน

1. เพื่อศึกษาถึงโครงสร้างและการทำงานของบัตรสมาร์ตการ์ด อันจะเป็นเทคโนโลยีที่สำคัญในอนาคต
2. ศึกษาการสื่อสารแบบอนุกรมมาตรฐาน RS-232 เพื่อส่งข้อมูลที่ได้จากบัตรสมาร์ตการ์ดสู่คอมพิวเตอร์เพื่อวิเคราะห์ผลต่อไป
3. ศึกษาการเขียนโปรแกรมประยุกต์สำหรับงานด้านการจัดการฐานข้อมูลและการแสดงผล

1.3 ขอบเขตของโครงงาน

ทำการสร้างเครื่องอ่านข้อมูลจากสมาร์ตการ์ด และวงจรที่ช่วยจัดการการส่งข้อมูลที่ได้จากเครื่องอ่านข้อมูลแต่ละเครื่องเข้าสู่คอมพิวเตอร์ ในส่วนของโปรแกรมประยุกต์ที่จัดการด้านฐานข้อมูลและแสดงผลจะสามารถแสดงข้อมูลต่างๆของการคิดเงินภายในบัตรของสมาร์ตการ์ด ในการอำนวยความสะดวกของสถานที่จอดรถยนต์

1.4 วิธีการดำเนินงาน

1. ศึกษาการเชื่อมโยงข้อมูลระหว่างเครื่องอ่านข้อมูลบัตรสมาร์ตการ์ด กับคอมพิวเตอร์
2. เขียนโปรแกรมประยุกต์ที่มีความสามารถในการรับข้อมูลจากเครื่องอ่านบัตร สมาร์ตการ์ด SLE 4442 พร้อมทั้งนำข้อมูลที่ได้มาใช้ในการค้นหาข้อมูลที่สอดคล้องกันในฐานข้อมูลที่มีอยู่ และแสดงผลข้อมูลที่สอดคล้องในฐานข้อมูลบนจอคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 ประโยชน์ที่ได้รับ

1. เพิ่มประสิทธิภาพความปลอดภัย
2. สร้างความสะดวกสบายแก่ผู้ที่ใช้สถานที่จอดรถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2 สมาร์ตการ์ด (Smart card)

2.1 ความหมายของสมาร์ตการ์ด

Smart Card หมายถึง การ์ดที่มีหน่วยความจำ หรือ มี Microprocessor ฝังอยู่ในการ์ดอาจจะเป็น Chip หน่วยความจำชนิดที่ถูกโปรแกรมเรียบร้อยแล้วมาจากโรงงาน แต่ถ้าเป็นแบบ Microprocessor นั้นก็จะสามารถเพิ่มข้อมูล หรือลบข้อมูล หรือไม่เช่นนั้นก็จะสามารถปรับปรุงเปลี่ยนแปลง ข้อมูลบนตัวการ์ดนี้ อีกประเภทคือ การ์ดที่มีหน่วยความจำดงที หรือที่เรียกว่า memory-chip card เช่น การ์ดโทรศัพท์ เป็นต้น ลักษณะตัวการ์ด Smart card นั้นจะเป็นแผ่นพลาสติก ขนาดเท่ากับ บัตรเครดิต หรือ ขนาดใกล้เคียงกับนามบัตร ซึ่งเป็นขนาดมาตรฐานทั่วโลก ภายในการ์ดนี้ก็จะมีเนื้อที่ส่วนที่เป็นหน่วยความจำอัดอยู่บนการ์ด ซึ่งในส่วนนี้นี่เองที่จะเป็นส่วนบรรจุข้อมูลอยู่ภายใน หากจะเพิ่มเติมข้อมูล หรือ อ่านข้อมูลจากบัตรก็จะต้องมีเครื่องอ่านบัตร ที่เรียกว่าSmartCardReader

Smart Card นั้นจะไม่เหมือนกับการ์ดชนิดใช้แถบแม่เหล็ก เพราะ Smart Card นั้นจะมีข้อมูลที่จำเป็น และข้อมูลข่าวสารอื่นๆอยู่บนการ์ด ดังนั้นการอ่านข้อมูลจึงไม่ต้องย้อนกลับไปค้นข้อมูลจากศูนย์ข้อมูลอันเป็นการเสียเวลาเช่นเดียวกับการ์ดแถบแม่เหล็ก(เช่น บัตรเอทีเอ็ม) นี่ก็จะเป็นข้อดีของ Smart Card

2.2 ประวัติความเป็นมาของสมาร์ตการ์ด

Smart Card นั้นประดิษฐ์ขึ้น ในปี ค.ศ. 1974 จวบกระทั่งทุกวันนี้ มี Smart Card ใช้อยู่ทั่วโลกราว 1000 ล้านใบ(ชิ้น) จากหลายๆผู้ผลิต 95 เปอร์เซ็นต์ ใช้ในประเทศในแถบยุโรป, อเมริกาใต้, และประเทศในแถบเอเชีย คาดว่าสิ้นปีนี้อาจจะใช้งานถึง 3000 ล้านใบ(ชิ้น) กว่า 15 เปอร์เซ็นต์ก็จะใช้งานในประเทศสหรัฐอเมริกาและคานาดา ในจำนวนนี้ประมาณว่า จำนวน 900 ล้านใบ(ชิ้น) จะใช้ในกิจการทางด้านการให้บริการการเงิน การ์ดธนาคาร หรือ การ์ด ATM, การ์ดด้านรักษาความปลอดภัย, การ์ดที่เกี่ยวกับโทรศัพท์ไร้สาย(มือถือ), การ์ดที่เกี่ยวกับการสื่อสารดาวเทียม, และการ์ดที่เกี่ยวกับการให้บริการเคเบิลทีวี TV Set-Top Boxes เป็นต้น

ปัจจุบันบ้านเราก็นำเอาสมาร์ตการ์ดมาใช้เต็มรูปแบบแล้ว ยกตัวอย่างที่เห็นได้ชัด ๆ ก็คือ บัตรประชาชนที่ปัจจุบันก็นำสมาร์ตการ์ดมาเก็บข้อมูลต่าง ๆ เกี่ยวกับเจ้าของบัตร อาทิเช่น กรุ๊ปเลือด ประวัติ การทำความดี หรือ มีประวัติเสียส่วนใด เช่น อุบัติเหตุ ก่อคดีวิวาท เป็นโจทย์ เป็นจำเลย คดีแพ่ง คดีอาญา จ่ายภาษี จ่ายประกันสังคม รวมไปถึงการเจ็บป่วยโดยผู้ป่วยไม้ได้สติเมื่อผู้ป่วยมาถึงโรงพยาบาลเครื่องอ่านบัตรก็รู้ทันทีว่าแพ้ยาอะไร แพทย์ก็ทำการรักษาอย่างถูกต้องทันต่อการช่วยชีวิตทันทีทันควัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการทดสอบ Smart card เทียบกับบัตรชนิดอื่น ๆ ได้ผลดังตารางดังนี้

เทคโนโลยี	**Magnetic Card**	**Optical Card**	**Smart Card**
อัตราการบกพร่อง	2 %	2%	0.03 %
จำนวนครั้งที่เข้าไปยุ่งเกี่ยว	400	400	100
จำนวนครั้งที่บกพร่องต่อปี	1000	800	100
จำนวนเครื่องที่สามารถบำรุงรักษาโดยใช้ช่างเทคนิค 1 คน	20	15	100
ค่าใช้จ่ายอะไหล่	400	500	100

ตารางที่ 2.1 การเปรียบเทียบการใช้งานการ์ดชนิดต่างๆของ France Telecom

โดยผลการเปรียบเทียบการบกพร่อง จำนวนเครื่องที่สามารถบำรุงรักษาโดยใช้ช่างเทคนิค 1 คน และ ค่าอะไหล่ Smart card กับบัตรชนิดอื่น ๆ พบว่า การใช้ Smart card มีประสิทธิภาพดีที่สุด

2.3 ข้อดีของสมาร์ตการ์ด

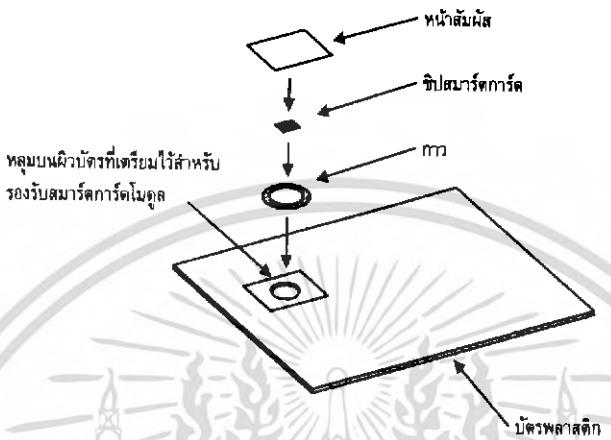
จากลักษณะของ Smart card พบว่ามีข้อดี ดังนี้

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 ส่วนประกอบและโครงสร้างของสมาร์ตการ์ด

สมาร์ตการ์ดประกอบด้วยบัตรพลาสติก กาวหรือวัสดุที่ใช้เชื่อมต่อ และหน้าสัมผัสที่บรรจุชิป สมาร์ตการ์ดเรียบร้อยแล้ว ซึ่งส่วนประกอบต่างๆแสดงดังรูป

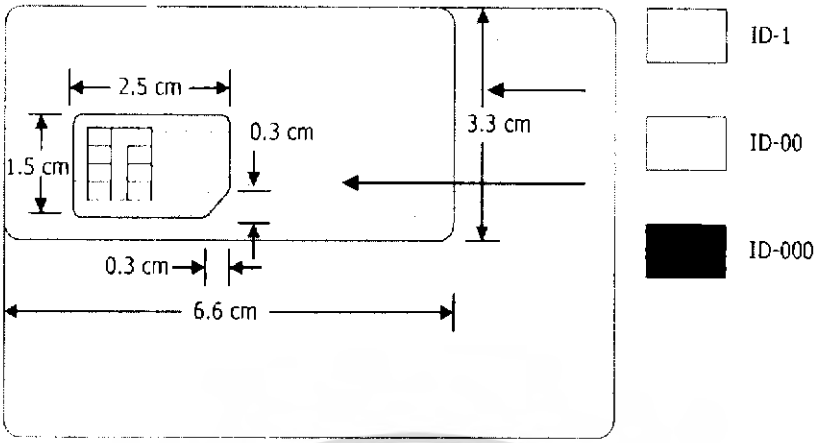


รูปที่ 2.1 ส่วนประกอบของสมาร์ตการ์ด

2.4.1 ตัวบัตรพลาสติก (Plastic Card)

ขนาดของบัตรพลาสติกที่นำมาทำสมาร์ตการ์ดกำหนดโดยมาตรฐานระหว่างประเทศ คือ ISO 7810 โดยมาตรฐานนี้ยังได้กำหนดถึงคุณลักษณะทางกายภาพของพลาสติกที่นำมาใช้ทำบัตรด้วย เช่น ความคลาดเคลื่อนของอุณหภูมิ และความยืดหยุ่นตัวในการใช้งาน ตำแหน่งของหน้าสัมผัสทางไฟฟ้าและการทำงานของมัน ตลอดจนกำหนดว่าการติดต่อระหว่างวงจรร่วม (Integrated Circuit) หรือ IC กับโลกภายนอกเป็นอย่างไรอีกด้วย มีพลาสติกอยู่4ชนิดที่นำมาใช้ผลิตสมาร์ตการ์ดได้แก่ PVC (Polyethylene Terephthalate), ABS (AcrylonitriIe Butadiene Styrene), PC (Polycarbonate), และ PET (Polyethylene Terephthalate) แต่ที่นิยมใช้กันมากในประเทศไทยคือ พีวีซี (PVC - Polyvinyl Chloride) และเอบีเอส (ABS - Acrylonitrile Butadiene Styrene) อย่างไรก็ดี การใช้พีวีซีมีข้อดีคือสามารถพิมพ์ลายนูนได้ แต่ไม่สามารถนำกลับมาใช้ใหม่ไม่ย่อยสลายในธรรมชาติได้ ส่วนเอบีเอสไม่สามารถพิมพ์นูนได้แต่นำกลับมาใช้งานใหม่ได้

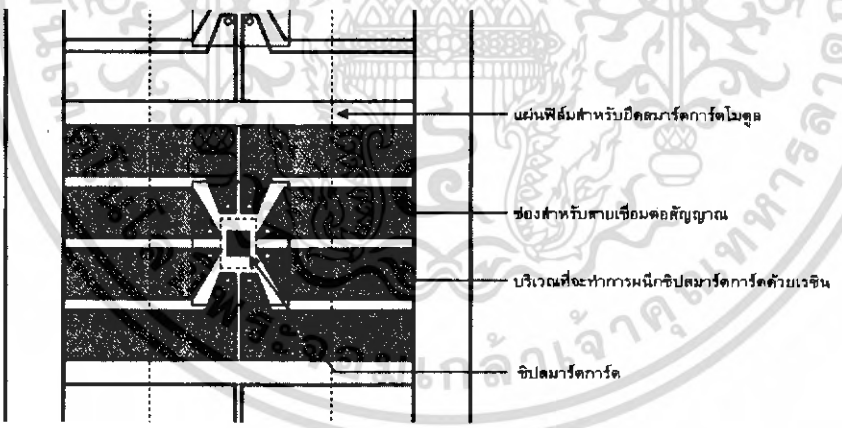
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 การแบ่งชนิดของบัตรตามรูปร่างที่นำไปใช้งาน และขนาด

2.4.2 หน้าสัมผัสและชิปสมาร์ตการ์ด (Smart Card Module)

สมาร์ตการ์ดโมดูลหรือหน้าสัมผัสและชิปสมาร์ตการ์ดคือ ส่วนที่แสดงความเป็นตัวตนของสมาร์ตการ์ดที่สุด สมาร์ตการ์ดบางชนิดเมื่อหยิบขึ้นมาเราอาจไม่อาจทราบได้เลยว่ามันคือ สมาร์ตการ์ดที่มีการฝังชิปไว้ในบัตร โดยส่วนที่จะแสดงภาพลักษณ์ที่ชัดเจนของสมาร์ตการ์ดคือสมาร์ตการ์ดโมดูล

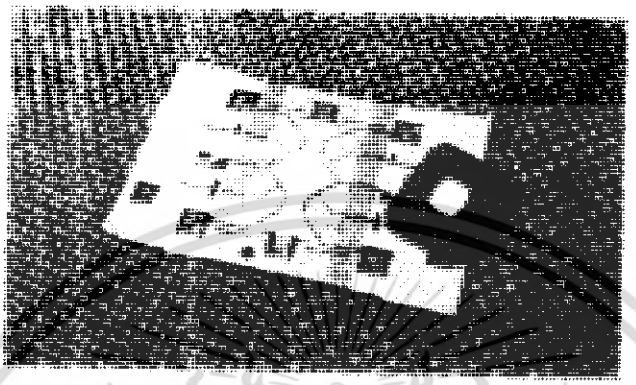


รูปที่ 2.3 ส่วนประกอบของสมาร์ตการ์ดโมดูลในขายการผลิตสมาร์ตการ์ด

ในการผลิตสมาร์ตการ์ดโมดูล ส่วนที่เป็นหน้าสัมผัสของสมาร์ตการ์ดประกอบด้วยโลหะหลายชนิดประกอบกัน แต่ละส่วนจะถูกยึดด้วยฟิล์มบางๆทางด้านหลังของหน้าสัมผัสเพื่อให้คงรูปอยู่ได้ แถบฟิล์มตัวนี้จะมีการเจาะช่องเล็กๆสำหรับการเชื่อมต่อสายนำสัญญาณกับสมาร์ตชิปกับหน้าสัมผัส หลังจากที่วางชิปสมาร์ตการ์ดลงในตำแหน่งที่ต้องการ และทำการเชื่อมต่อสายนำสัญญาณจากชิป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สมาร์ตการ์ดเข้ากับหน้าสัมผัสเรียบร้อยแล้ว ขั้นสุดท้ายจะเป็นการผนึกชิปเพื่อป้องกันตัวชิป และสายนำสัญญาณต่างๆจากสิ่งแวดล้อมภายนอกขั้นสุดท้ายจะเป็นการนำหน้าสัมผัสและชิปใส่ลงในบัตรพลาสติกและทดสอบการทำงานของชิปขั้นสุดท้าย



รูปที่2.4ตัวอย่างสมาร์ตการ์ดโมดูล

2.5 องค์ประกอบในการใช้งานสมาร์ตการ์ด

2.5.1 ตัวบัตรและตัวชิป

บัตรและชิปสมาร์ตการ์ด เป็นส่วนแรกที่จะกล่าวถึง เพราะสมาร์ตการ์ดมีหลากหลายรูปแบบ หลากหลายการใช้งาน โดยหลักการแล้วสมาร์ตการ์ดเป็นเพียงบัตรฝังชิป IC ที่สามารถเก็บข้อมูลได้เท่านั้น ผู้ออกแบบระบบมีหน้าที่นำสมาร์ตการ์ดมาใช้งานอย่างชาญฉลาดเหมาะสมตามประเภทงาน และ บริหารข้อมูลภายในสมาร์ตการ์ดให้เกิดความปลอดภัยสูงสุด

สมาร์ตการ์ดที่นำมาใช้งานมีตั้งแต่ราคาใบละไม่กี่ร้อยบาท ถึงใบละหลายพันบาน โดยในปัจจุบันเราสามารถเห็นการใช้งานสมาร์ตการ์ดในหลายรูปแบบเช่น บัตรโทรศัพท์ ซิมการ์ดในโทรศัพท์มือถือ , บัตรเข้าออกที่อยู่อาศัย (คอนโดมิเนียมบางแห่ง) , บัตรประชาชน , บัตรนักศึกษา , บัตรพนักงาน , บัตรเติมน้ำมันแบบเครดิต (Fleet Card) , บัตรแทนเงินสด , ซิมการ์ดในโทรศัพท์มือถือ ซึ่งมีการกำหนดเป็นมาตรฐาน GSM โดยผู้ผลิตสมาร์ตการ์ดต้องผลิตสมาร์ตการ์ดที่มีโครงสร้างที่มีโครงสร้างข้อมูลภายในตามที่มาตรฐาน GSM กำหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.2 สมาร์ตการ์ดรีดเดอร์

สมาร์ตการ์ดรีดเดอร์จะประกอบด้วยขาสำหรับเชื่อมสัญญาณกับหน้าสัมผัสบนชิปสมาร์ตการ์ด (Card Contact) หรือเป็นเสาอากาศรับส่งคลื่นวิทยุสำหรับสมาร์ตการ์ดแบบไม่มีหน้าสัมผัส (Contact less) และหน่วยประมวลผลพร้อมหน่วยความจำสำหรับติดต่อสื่อสารกับชิปสมาร์ตการ์ดโดยตรง การสร้างสมาร์ตการ์ดรีดเดอร์ขึ้นใช้เองสามารถทำโดยการนำไมโครโปรเซสเซอร์หรือไมโครคอนโทลเลอร์ มาประยุกต์ใช้ในการเชื่อมต่อกับสมาร์ตการ์ด

2.5.3 ซอฟต์แวร์

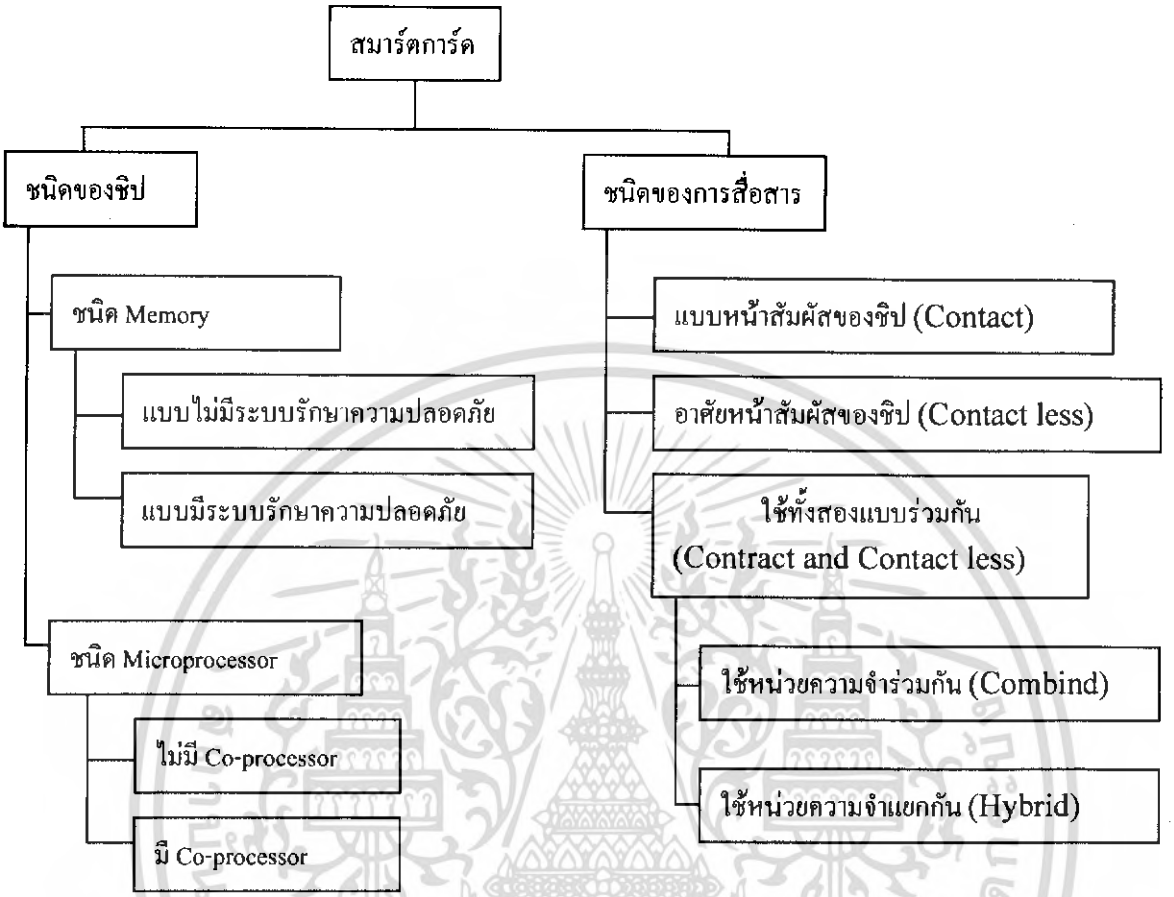
ในการใช้งานสมาร์ตการ์ดนอกจากตัวบัตรสมาร์ตการ์ด สมาร์ตการ์ดรีดเดอร์แล้วยังมีส่วนประกอบอีกส่วนที่สำคัญคือ ซอฟแวร์สำหรับจัดการข้อมูลในสมาร์ตการ์ด และซอฟแวร์สำหรับบริหารงานด้านบัตร หรืออาจเรียกได้ว่าระบบ Front-End (เหมือนกับระบบในบัตรเครดิต) ซึ่งระบบ Front – End ของสมาร์ตการ์ดจะแตกต่างจากระบบบัตรแถบแม่เหล็ก เนื่องจากสมาร์ตการ์ดไม่จำเป็นต้องมีการติดต่อสื่อสารกับ Front-End ทุกครั้งที่ทำรายการเหมือนในระบบบัตรเครดิต ทำให้ระบบ Front-End ของสมาร์ตการ์ดมีเวลามากพอในการบริหารงานด้านอื่นๆหากต้องการติดต่อสื่อสารกับระบบ Front-End ของสมาร์ตการ์ดจำเป็นต้องใช้สมาร์ตการ์ดรีดเดอร์ที่มีส่วนสำหรับการติดต่อสื่อสารไม่ว่าจะเป็น MODEM , Ethernet , Local Area Network ,ระบบสื่อสารด้วยเครื่องวิทยุ, ระบบสื่อสารอนุกรม RS-485/422 สำหรับการสื่อสารในบริเวณพื้นที่ให้บริการที่ไม่กว้างใหญ่นัก เพื่อใช้รับ-ส่งข้อมูล ระหว่าง Front-End เมื่อจำเป็น

2.5.4 Back Office

เป็นซอฟแวร์สำหรับควบคุมดูแลระบบทั้งหมด ประกอบด้วยซอฟต์แวร์สำหรับป้อนข้อมูลเกี่ยวกับบัตร และผู้ถือบัตรเพื่อออกบัตรใหม่ (Card Issuer), ซอฟต์แวร์สำหรับออกรายงานต่างๆ (Report) และซอฟแวร์ส่วนสุดท้ายคือสำหรับให้บริการผู้ถือบัตร เช่น ซอฟแวร์สำหรับเติมเงินลงในชิป (สมาร์ตการ์ดที่ใช้แทนเงินสด), ซอฟแวร์สำหรับเติม-แลกแต้มในบัตรสะสมแต้ม (Royalty Card) ปกติแล้วซอฟแวร์ในส่วนของ Back-End และ BackOffice ที่กล่าวมาต้องทำร่วมกับสมาร์ตการ์ดรีดเดอรเสมอ เพราะเพียงสมาร์ตการ์ดไม่สามารถทำรายการใดๆได้เอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

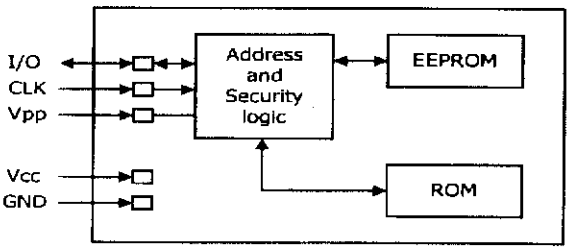
2.6ประเภทของสมาร์ตการ์ด



รูปที่ 2.5 การแบ่งสมาร์ตการ์ดตามชนิดของหน่วยความจำและประเภทของหน้าสัมผัส

2.6.1 Memory Card (Synchronous Card)

สมาร์ตการ์ด แบบ Memory หรืออีกชื่อหนึ่งคือ Synchronous Card เนื่องจากสมาร์ตการ์ดชนิดนี้มีการรับ-ส่งข้อมูลตามสัญญาณนาฬิกาที่ป้อนให้แก่ชิป (ข้อมูลแต่ละบิตที่ส่งให้แก่ชิปต้องสัมพันธ์กับสัญญาณนาฬิกา สมาร์ตการ์ดชนิดนี้มีโครงสร้างที่ประกอบไปด้วย วงจรสำหรับติดต่อสื่อสารภายนอก , หน่วยความจำข้อมูล , และหน่วยความจำสำหรับเก็บชุดคำสั่งของสมาร์ตการ์ด ดังรูปที่ 2.6



รูปที่ 2.6 บล็อกไดอะแกรมโครงสร้างในชิปสมาร์ตการ์ดชนิด Memory (Synchronous Card)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สมาร์ตการ์ดที่เป็นพื้นฐานในปัจจุบัน ก็คือสมาร์ตการ์ดชนิด Free Access Memory สมาร์ตการ์ดชนิดนี้เปิดโอกาสให้อ่านหรือเขียนข้อมูลในแอดเดรสใดๆ ก็ได้ตามชื่อของสมาร์ตการ์ดชนิดนี้ ไม่มีการป้องกันข้อมูลใดๆภายในการ์ดชนิดนี้เป็นการ์ดที่มีความปลอดภัยต่ำสุด แต่การอ่านข้อมูลก็จะไม่สามารถอ่านได้ง่ายนัก เนื่องจากเมื่อมีการออกแบบหน่วยความจำให้มีการสลับตำแหน่งของบิตข้อมูล โดยมีวงจรควบคุมการสลับตำแหน่งของบิตเป็นส่วนป้องกันข้อมูลอีกต่อหนึ่ง ดังนั้นการอ่านข้อมูลแบบธรรมดา จะไม่ได้ข้อมูลที่ถูกต้องหากไม่ได้ติดต่อกับวงจรควบคุมการสลับตำแหน่งของบิตโดยตรง

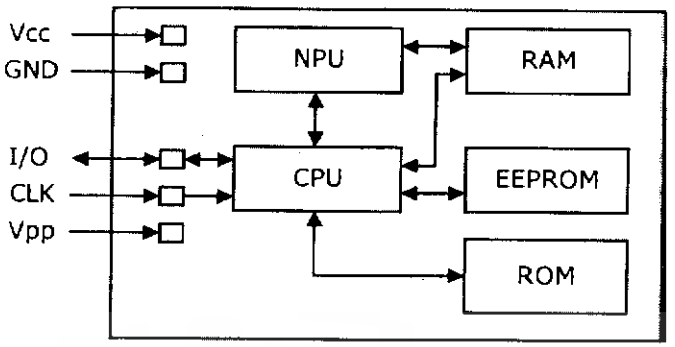
2.6.2 การ์ดแบบออพทิคอลเมมโมรี (Optical Memory Cards)

เป็น Smart Card แบบที่ใช้ระบบส่งข้อมูลทางแสง (Optical) ในส่วนที่มีข้อมูลนั้นก็จะมีลักษณะเหมือนแผ่นพลาสติก หรือ เศษของแผ่น CD อัดอยู่บนการ์ดนี้ ซึ่งในขณะนี้ก็มีการนำมาใช้บ้างแล้ว Optical Memory Card นี้ สามารถเก็บข้อมูลได้ถึง 4MB แต่ลง หรือ บรรจุข้อมูลได้ครั้งเดียว เปลี่ยนแปลง หรือ โยกย้ายข้อมูลออกจากการ์ดนี้ไม่ได้ ดังนั้นการ์ด Optical Memory Card นี้ก็มักจะถูกนำไปใช้ในวงการ การเก็บรักษาข้อมูล เช่น ข้อมูลทางการแพทย์ ข้อมูลเกี่ยวกับบุคคล ข้อมูลการท่องเที่ยว เป็นต้น ปัจจุบัน ตัวการ์ดเองก็ไม่มี Microprocessor อยู่ในตัวการ์ด แต่ในอนาคตก็อาจจะมีการปรับปรุงให้ทันสมัยขึ้นมากกว่าเวลานี้ อีกประการหนึ่งก็คือ เครื่องอ่านข้อมูลของบัตร Optical Memory Card นี้ ราคายังค่อนข้างแพง และ ที่สำคัญก็ยังไม่มีมาตรฐานมาบังคับใช้

2.6.3 Processor Card (Asynchronous Card)

สมาร์ตการ์ดชนิดนี้ เป็นสมาร์ตการ์ดที่ได้รับการพัฒนาปรับปรุงจากสมาร์ตการ์ดชนิด Memory ด้วยการใส่เทคโนโลยีไมโครโปรเซสเซอร์เข้าไปในชิป เพื่อให้ชิปสามารถประมวลผลข้อมูลและเพิ่มวามปลอดภัยให้แก่ข้อมูลให้สูงขึ้น การที่ใส่ไมโครโปรเซสเซอร์ลงในชิป ทำให้จำเป็นต้องมีการเพิ่มส่วนของหน่วยความจำสำหรับจัดการเก็บระบบปฏิบัติการของไมโครโปรเซสเซอร์ และหน่วยความจำชั่วคราสำหรับการประมวลผลนอกจากนี้การใส่ชิปประมวลผลทางคณิตศาสตร์ลงในชิปสมาร์ตการ์ด เพื่อช่วยในการประมวลผลข้อมูลด้วยอัลกอริทึมสำหรับเข้า-ถอดรหัส ทำให้สมาร์ตการ์ดชนิด Processor มีความเร็วสูงกว่าสมาร์ตการ์ดชนิด Memory หลายเท่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



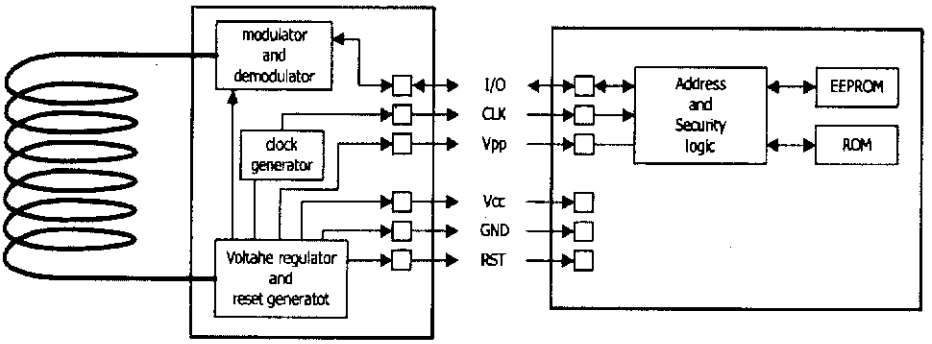
รูปที่ 2.7 บล็อกไดอะแกรมโครงสร้างภายในชิปสมาร์ตการ์ดชนิด Processor (Asynchronous Card)

ในการรับส่งข้อมูลของสมาร์ตการ์ดชนิดนี้ จะใช้หน้าสัมผัสชนิดเดียวกันกับสมาร์ตการ์ดชนิด Memory โดยสัญญาณนาฬิกาที่ป้อน จะถูกใช้เป็นสัญญาณนาฬิกาให้แก่โปรเซสเซอร์ภายในสมาร์ตการ์ด ข้อมูลที่รับ-ส่งจึงไม่จำเป็นต้องสัมพันธ์กับสัญญาณนาฬิกาที่ป้อนให้แก่ชิปเพียงกำหนดอัตราการรับ- ส่งข้อมูลเป็น 9600 บิต/วินาที ก็จะสามารถติดต่อกับโปรเซสเซอร์ของชิปได้แล้ว แต่การเข้าถึงข้อมูลจะไม่สามารถทำได้เหมือนกับการ์ดชนิด Memory การเข้าถึงข้อมูลต้องกระทำผ่านโปรเซสเซอร์ของสมาร์ตการ์ดเท่านั้น ไม่ว่าจะเป็นการอ่านหรือการเขียนข้อมูลก็ตาม เพราะหน่วยความจำจะอยู่ภายในความควบคุมของโปรเซสเซอร์เพียงอย่างเดียว ข้อดีอีกอย่างที่ไม่สามารถติดต่อกับหน่วยความจำในชิปโดยตรงก็คือ การลอบเข้าถึงข้อมูลโดยไม่ได้รับอนุญาตแทบเป็นไปไม่ได้ ยกเว้นมีความบกพร่องในการกำหนดเงื่อนไขในการเข้าถึงข้อมูลที่เป็นความลับ

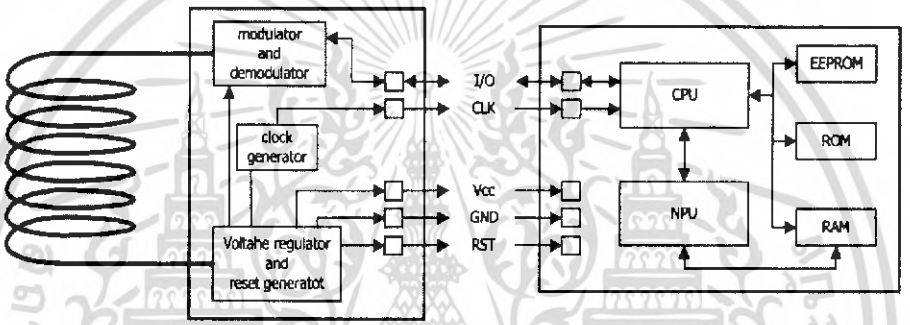
2.6.4 Contact less Smart Card

สมาร์ตการ์ดชนิด Contact less ใช้เทคโนโลยีการสื่อสารผ่านคลื่นวิทยุ โดยการส่งคลื่นวิทยุความถี่ 13.56 เมกะเฮิรตซ์ ที่ได้รับการมอดดูเลทข้อมูลแล้วส่งให้กับชิปสมาร์ตการ์ด ทางด้านชิปสมาร์ตการ์ดจะใช้ขดลวด เป็นเสารับส่งสัญญาณ โดยเสารับส่งสัญญาณนี้จะเป็นขดลวดขนาดเล็กที่ถูกฝังอยู่ในตัวบัตร ภายนอกบัตรชนิดนี้แทบดูไม่ออกว่าเป็นบัตรสมาร์ตการ์ด จากรูป 2.9 จะเห็นว่าส่วนที่เพิ่มเข้ามาเป็นส่วนที่รับสัญญาณคลื่นวิทยุมาแบ่งเป็นสองส่วน โดยส่วนแรกจะถูกแปลงเป็นกระแสไฟฟ้าสำหรับป้อนชิป และวงจรสร้างสัญญาณนาฬิกาให้สามารถทำงานได้ อีกส่วนหนึ่งจะถูก Demodulate เอาข้อมูลออกจากคลื่นวิทยุ และส่งให้แก่ชิปสมาร์ตการ์ดอีกต่อหนึ่ง ส่วนการส่งข้อมูลกลับก็จะใช้กระแสไฟฟ้าที่ได้จากคลื่นวิทยุมาใช้ในการ Modulate ข้อมูลและส่งกลับไปยังเสารับ- ส่งสัญญาณภายในบัตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 สมาร์ตการ์ด ชนิด Memory แบบ Contact less



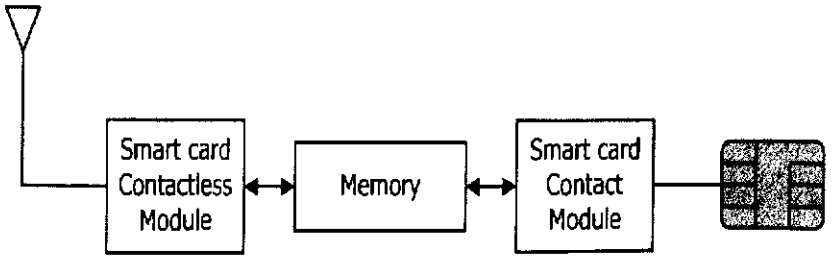
รูปที่ 2.9 สมาร์ตการ์ดชนิดProcessor แบบ Contact less

สมาร์ตการ์ดชนิด Contact less ถูกออกแบบให้ใช้กระแสไฟฟ้าต่ำเพราะกระแสไฟฟ้าที่ได้จากคลื่นวิทยุนั้นมีปริมาณเพียงเล็กน้อย ไม่เพียงพอที่จะทำให้สมาร์ตการ์ดแบบธรรมดาสามารถทำงานได้ สมาร์ตการ์ดชนิดนี้รุ่นแรกๆจะไม่สามารถทำคำสั่งที่ซับซ้อนมากๆเช่นคำสั่งในการเข้ารหัสข้อมูล หรือคำสั่งที่ต้องใช้เวลาในการประมวลผลมากๆและระยะในการรับ-ส่งสัญญาณก็ไม่มากนัก แต่ปัจจุบันสมาร์ตการ์ด Contact less สามารถทำการเข้ารหัสที่ยุ่งยากได้แล้วด้วยการเพิ่มวงจรสำหรับเข้ารหัสภายในชิป

2.6.5 Com-Bi Card

สมาร์ตการ์ดชนิดนี้เป็นการรวมเอาสมาร์ตการ์ดแบบ Contact และสมาร์ตการ์ดชนิด Contact less เข้าด้วยกัน โดยใช้หน่วยความจำข้อมูลร่วมกันเพื่อให้การทำรายการที่จำเป็นต้องอยู่ภายใต้การควบคุมอยู่ และสามารถใช้งานทั่วไปได้อย่างสะดวกสบาย (Speed Pass) ผ่านทางคลื่นวิทยุ

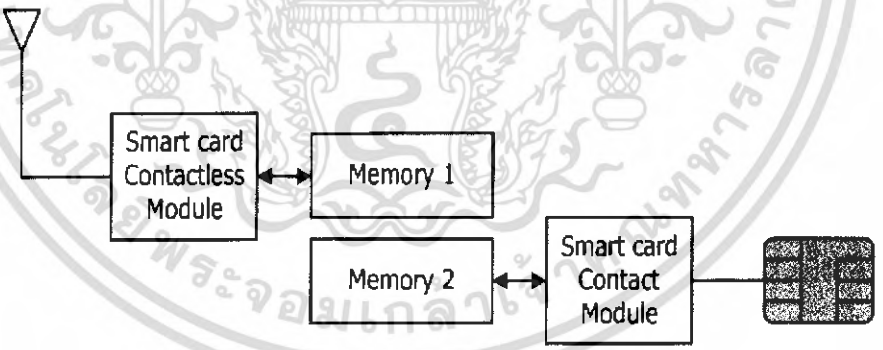
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10โครงสร้างภายในของสมาร์ตการ์ดชนิด Contact แบบ Com-bi card

2.6.6 Hybrid Smart Card

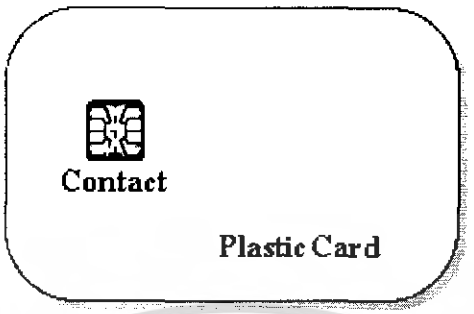
สมาร์ตการ์ดชนิดนี้มีลักษณะโครงสร้างภายในคล้ายกับประเภท Com-Bi Card แต่จะแตกต่างกันในเรื่องของหน่วยความจำข้อมูล โดยหน่วยความจำข้อมูลจะถูกแยกจากกันอย่างสิ้นเชิงระหว่าง Contact และ Contact less เพื่อความสะดวกในการแยกใช้งาน ซึ่งปัจจุบัน Hybrid Card จะมีความหมายรวมถึงบัตรที่มีคุณสมบัติในการใช้งานตั้งแต่สองอย่างขึ้นไป บัตรสมาร์ตการ์ดที่มีทั้งชิปสมาร์ตการ์ดและแถบแม่เหล็ก, บัตรสมาร์ตการ์ด ที่เป็นทั้ง Contact และ Contact less



รูปที่ 2.11 โครงสร้างภายในของสมาร์ตการ์ดชนิด Hybrid Smart Card

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 รูปแบบของสมาร์ตการ์ดที่นำมาใช้ในโครงงาน



รูป2.12 บัตร Smart Card SLE4442

ในโครงงานนี้ได้ใช้บัตร Smart Card SLE4442 ซึ่งเป็นการ์ดที่มีระบบป้องกันความปลอดภัยข้อมูลหรือ Security Memory Card โดยสมาร์ตการ์ดประเภทนี้จะสามารถอ่านข้อมูลที่มีอยู่ภายในบัตรได้อย่างอิสระ แต่การเปลี่ยนแปลงข้อมูลที่มีอยู่ภายในบัตรนั้นจะถูกออกแบบให้ซับซ้อนยิ่งขึ้นโดยไม่สามารถแก้ไขข้อมูลภายในบัตรหากไม่มีรหัส PSC (Program Security Code) โดยจุดนี้เองเป็นส่วนที่ทำให้ Security Memory Card แตกต่างจาก Free Access Memory Card อย่างชัดเจนเนื่องจากข้อมูลภายในบัตรสมาร์ตการ์ดได้รับการปกป้องและมีความน่าเชื่อถือ รูปแบบการสื่อสารข้อมูลของการ์ดชนิดนี้เป็นการสื่อสารข้อมูลแบบซิงโครนัส ตามมาตรฐาน ISO7816 ซึ่งรูปแบบคำสั่งที่ใช้ในการติดต่อและควบคุมการ์ดจะแตกต่างกันไปในผู้ผลิตการ์ดแต่ละราย

2.7.1 การจัดการหน่วยความจำภายใน

ในสมาร์ตการ์ดชนิด Memory มีการแบ่งวิธีการจัดการหน่วยความจำเป็นสองแบบ คือ bitwise และ bytewise การจัดการหน่วยความจำแบบ bitwise เป็นการจัดการหน่วยความจำที่ใช้สมาร์ตการ์ดรุ่นแรกๆ การจัดการหน่วยความจำแบบนี้มักใช้บอกขนาดของหน่วยความจำของสมาร์ตการ์ดเป็นหน่วยบิต เช่น 1 กิโลบิต (128 ไบต์), 8กิโลบิต (1 กิโลไบต์) สาเหตุที่ bitwise จัดการหน่วยความจำข้อมูลเป็นบิตเนื่องมาจากข้อมูลที่ใช้รับ – ส่งในสมาร์ตการ์ดนี้ทำกันได้ในระดับบิตเท่านั้น หมายความว่าการับส่งข้อมูลไม่จำเป็นต้องทำให้ครบทั้ง 8 บิตหรือ 1 ไบต์ การจัดการหน่วยความจำแบบนี้ สามารถอ่านข้อมูลที่บิตใดก็ได้ ซึ่งมีใช้ในสมาร์ตการ์ดที่มีหน่วยความจำข้อมูลไม่มากนัก

สำหรับการจัดการหน่วยความจำแบบ bytewise เป็นการจัดการหน่วยความจำที่อ้างถึงข้อมูลขนาด 8 บิตหรือ 1 ไบต์เต็ม การรับ – ส่งข้อมูลกับชิปต้องทำการรับ – ส่งข้อมูลทั้ง 8 บิต จนครบจึงจะทำการรับ – ส่งข้อมูลเสร็จสมบูรณ์(หากทำไม่เสร็จสมบูรณ์ ชิปสมาร์ตการ์ดจะยกเลิกการรับส่งข้อมูลครั้งนั้นๆ) นอกจากนี้การอ้างอิงหน่วยความจำยังมีความแตกต่างกัน เช่น บางผู้ผลิตกำหนดให้หน่วยความจำเป็นแอดเดรสที่ต่อเนื่องกันตั้งแต่แอดเดรสที่ 0 ถึง แอดเดรสสุดท้ายบางผู้ผลิตแบ่งหน่วยความจำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ออกเป็นเพจ (Page) แต่ละเพจมีขนาดแตกต่างกันตามแต่รุ่นที่ผลิต ทำให้การอ้างถึงข้อมูลใดๆ ในหน่วยความจำของสมาร์ตการ์ดแต่ละแบบไม่เหมือนกันตามแต่ผู้ผลิตจะออกแบบ

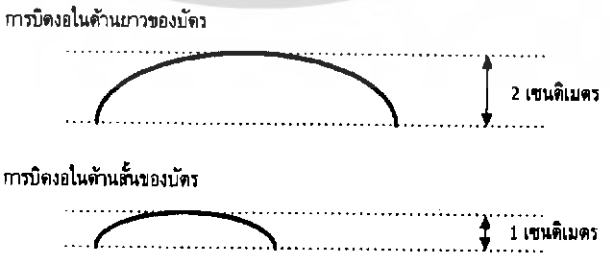
2.7.2 มาตรฐานของสมาร์ตการ์ดที่เกี่ยวข้อง

เพื่อให้เกิดความเข้าใจกันได้ของสมาร์ตการ์ด จึงมีการกำหนดมาตรฐานของสมาร์ตการ์ดคือ ISO7816 เป็นข้อกำหนดในเรื่องของคุณสมบัติของบัตรพลาสติกที่จะนำมาใช้ทำเป็นสมาร์ตการ์ด โดยมีหัวข้อย่อยแบ่งเป็น ISO7816-1, ISO7816-2, ISO7816-3, ISO7816-4, ISO7816-5, ISO7816-6 ในที่นี้จะกล่าวในรายละเอียดของ3มาตรฐานแรกเท่านั้นเนื่องจากมีความสำคัญต่อการใช้งานในโครงงาน

2.7.2.1 มาตรฐานISO7816-1

-เป็นมาตรฐานที่กำหนดด้วยเรื่องของคุณสมบัติทางกายภาพเบื้องต้นของสมาร์ตการ์ด ประกอบด้วย

- ความคงทนต่อแสงและรังสีต่างๆ
- ขนาดความหนาของชิปสมาร์ตการ์ด
- ความทนต่อแรงกดของหน้าสัมผัส (ทนต่อแรงกด 1.5 นิวตันได้โดยไม่เสียหาย)
- ค่าความต้านทานของหน้าสัมผัส (ไม่เกิน 0.5 โอห์ม ที่กระแส 0.5 ไมโครแอมป์ – 300 มิลลิแอมป์)
- ความทนต่อสนามแม่เหล็ก
- ความทนต่อไฟฟ้าสถิต (1500 โวลต์ ประจุ 100 พิโกฟารัด ที่ 1500 โอห์ม)
- ความทนทานต่อการบิดงอ เป็นจำนวน 30 ครั้งต่อนาที โดยที่บัตรและชิปต้องไม่เกิดความเสียหาย



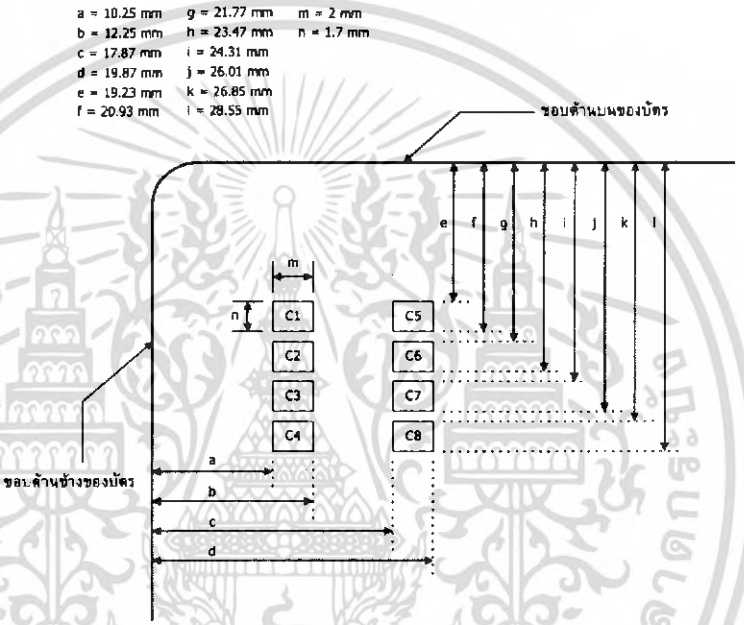
รูปที่ 2.13 วิธีทดสอบการบิดงอสมาร์ตการ์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.2.2 มาตรฐาน ISO7816-2

เป็นมาตรฐานที่กำหนดขนาดของหน้าสัมผัสและตำแหน่งของหน้าสัมผัสชิปสมาร์ตการ์ดบนบัตร ประกอบด้วย

- ขนาดของหน้าสัมผัสชิปสมาร์ตการ์ด
- ตำแหน่งของหน้าสัมผัสบนบัตร ดังรูป



รูปที่ 2.14 ตำแหน่งหน้าสัมผัสของชิปสมาร์ตการ์ด

2.7.2.3 มาตรฐาน ISO7816-3

มาตรฐานที่กำหนดคุณสมบัติทางไฟฟ้าและ Protocol ที่ใช้ในการสื่อสารกับชิปสมาร์ตการ์ด จะเป็นการบรรยายเกี่ยวกับหน้าสัมผัสมีดังนี้

Vcc แรงดันไฟบวกของแหล่งจ่ายไฟฟ้าที่ป้อนให้แก่ชิป

Vpp แรงดันไฟฟ้าสำหรับการเขียนข้อมูลลงในชิปสมาร์ตการ์ด

GND กราวด์ของแหล่งจ่ายกระแสไฟฟ้าที่ป้อนให้แก่ชิป

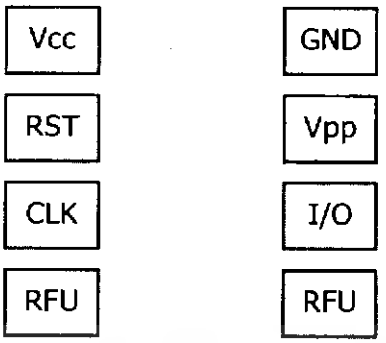
RST แรงดันไฟฟ้าสำหรับรีเซตชิปสมาร์ตการ์ด

I/O Input – Output สำหรับการรับส่งข้อมูลแบบอนุกรม

CLK สัญญาณนาฬิกาสำหรับกำหนดจังหวะการรับ-ส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

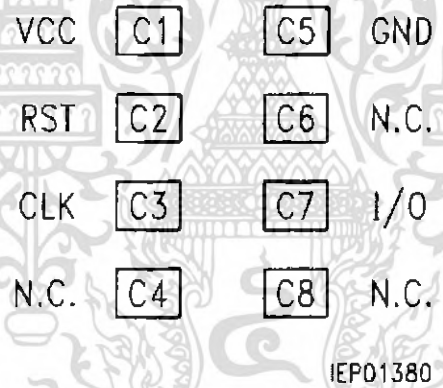
สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



รูปที่ 2.15 หน้าที่การทำงานของแต่ละหน้าสัมผัส

2.8 รูปแบบชิปของสมาร์ตการ์ด SLE4442

ในบัตร SLE4442 การ์ด จะมีหน้าสัมผัสอยู่ 8 หน้าสัมผัส ดังนี้



รูปที่ 2.16 หน้าที่การทำงานของแต่ละหน้าสัมผัสสมาร์ตการ์ด SLE4442

Vcc แรงดันไฟบวกของแหล่งจ่ายไฟฟ้าที่ป้อนให้แก่ชิป

GND กราวด์ของแหล่งจ่ายกระแสไฟฟ้าที่ป้อนให้แก่ชิป

RST แรงดันไฟฟ้าสำหรับรีเซตชิปสมาร์ตการ์ด

I/O Input – Output สำหรับการรับส่งข้อมูลแบบอนุกรม

CLK สัญญาณนาฬิกาสำหรับกำหนดจังหวะการรับ-ส่งข้อมูล

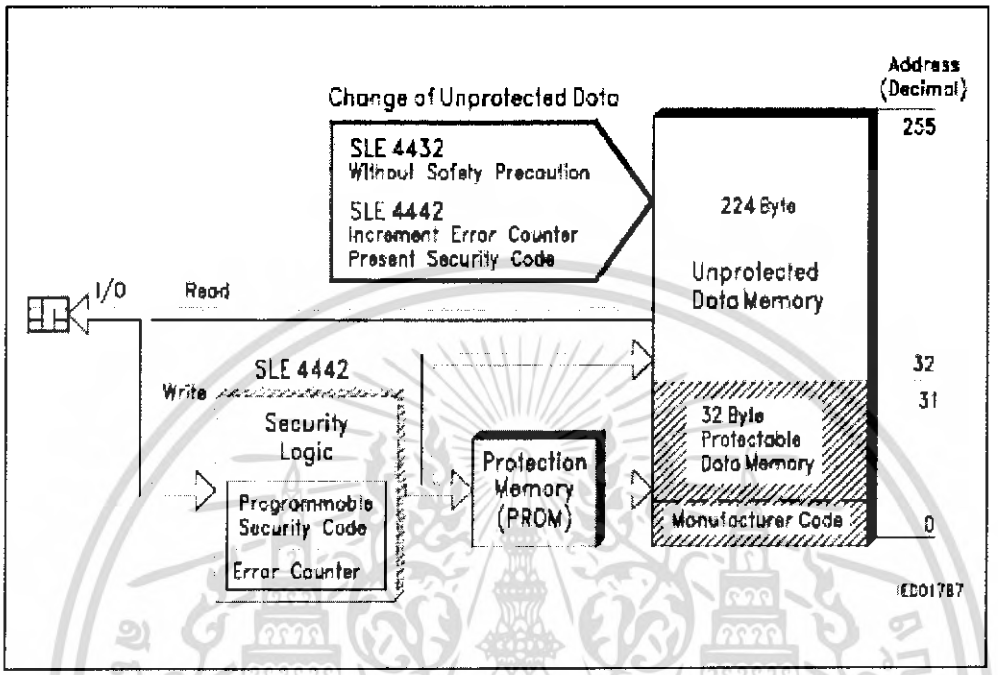
N.C. รองรับไว้เพื่อการใช้งานในอนาคต(Not Connect)

71957

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9 คุณสมบัติทั่วไปของ SLE 4442

2.9.1ใช้หน่วยความจำ EEPROM 8บิต ความจุข้อมูล 256 ไบต์ ดังรูป



รูปที่ 2.17 แสดงหน่วยความจำภายในของ SLE 4442

2.9.2 ใช้รูปแบบของ ATR (Answer To Reset) ตามมาตรฐาน ISO7816 – 3

2.9.3 อินเตอร์เฟสแบบซิงโครนัส(Synchronous) ตามมาตรฐาน ISO7816

2.9.4 ป้องกันการเขียนข้อมูลด้วยรหัสผ่าน PSC (Program Security Code)

2.9.5 การลบและเขียนข้อมูลในแต่ละไบต์ใช้เวลาเพียง 2.5มิลลิวินาที

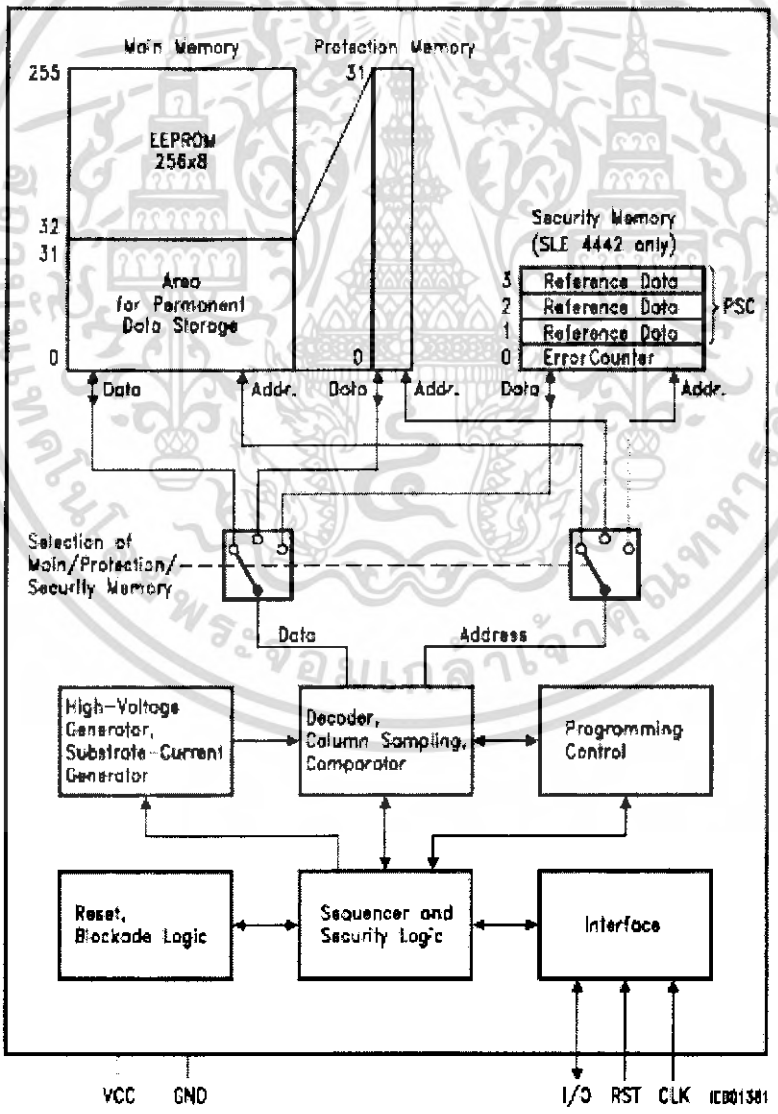
2.9.6 มีฟังก์ชันป้องกันข้อมูลในพื้นที่หน่วยความจำ 32ไบต์แรก โดยสามารถจะกำหนดให้ข้อมูลที่เขียนลงไปยังพื้นที่ช่วงดังกล่าวถูกเขียนลงไปอย่างถาวรได้

จะเห็นว่าหน่วยความจำขนาด 256ไบต์ ที่อยู่ภายใน SLE4442 จะถูกแบ่งออกเป็น 2ส่วนด้วยกัน ได้แก่ข้อมูลในช่วง 32ไบต์แรก ซึ่งเป็นพื้นที่ที่มีระบบป้องกันการเขียนข้อมูลทับ และหน่วยความจำส่วนถัดมาซึ่งเป็น EEPORM ที่สามารถทั้งเขียนและอ่านได้ กลไกในการป้องกันข้อมูลของ SLE4442 มาจากส่วนที่เป็น security memory ที่ได้รับการปกป้องโดยข้อมูลสำคัญ 2ส่วน คือ

Reference Data (PSC) เป็นข้อมูลขนาด 3ไบต์ที่เก็บค่าของรหัสผ่านสำหรับการเข้าไปแก้ไขข้อมูลในหน่วยความจำเอาไว้ (ข้อมูลในส่วนของ PSC ไม่สามารถอ่านออกมาได้ หากไม่มี การป้อนรหัสที่ถูกต้องก่อน) รหัส PSC จะถูกกำหนดเป็นค่าหนึ่งมาโดยผู้ผลิตก่อนซึ่งสามารถ จะปรับเปลี่ยนเองได้ภายหลังเมื่อใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Error Counter Byte เป็นข้อมูลที่บอกถึงจำนวนครั้งที่พยายามจะป้อนรหัส PSC แต่ไม่ถูกต้อง ซึ่งจะถูกกำหนดเอาไว้ตายตัวว่าสามารถป้อนรหัสผิดได้ไม่เกิน 3ครั้ง หากเกินกว่านั้นการ์ดจะล็อกตัวเองอย่างถาวรทันที และไม่มีทางปลดล็อกได้ แม้จะป้อนรหัส PSC ที่ถูกต้องในครั้งต่อไปแล้วก็ตาม การเขียนข้อมูลยังหน่วยความจำก็จะไม่สามารถทำได้อีกต่อไป แต่ยังคงอ่านข้อมูลออกมาได้ตามปกติ การป้อนรหัส PSC ผิดแต่ละครั้ง Error Counter จะถูกจะถูกเปลี่ยนข้อมูลจาก 1 เป็น 0 ทันที หากค่า Error Counter เป็น 0 ทั้ง 3 bit ก็แสดงว่าการ์ดได้ถูกล็อกไปเรียบร้อยแล้ว (ในกรณีที่ป้อนรหัสผิดมาแล้ว 2ครั้ง แต่ป้อนรหัสถูกในครั้งที่ 3 ค่าของ Error Counter จะถูกรีเซตกลับเป็น 1ทั้งหมดเหมือนอย่างตอนแรกเริ่ม)



รูปที่ 2.18 บล็อกไดอะแกรมโครงสร้างภายในของ SLE4442

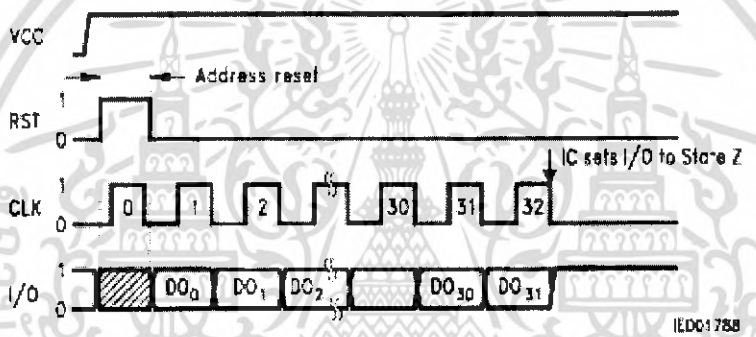
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10 รูปแบบการสื่อสารข้อมูลของสมาร์ตการ์ด SLE4442

รูปแบบการสื่อสารข้อมูลของสมาร์ตการ์ดเบอร์ SLE4442 เป็นการรับส่งข้อมูลระหว่างเครื่อง อ่านสมาร์ตการ์ดแบบ 2ทิศทาง (เป็นข้อมูลบนสาย I/O จะถูกอ่านค่าที่ขอบขาขึ้นของ สัญญาณนาฬิกา) รูปแบบการสื่อสารที่ว่านี้ประกอบด้วย 4โหมดการทำงาน ได้แก่

- การรีเซตและการตอบกลับด้วย ATR (Answer To Reset)
- โหมดการส่งคำสั่ง (Command Mode)
- โหมดการอ่านข้อมูล (Out – going Data Mode)
- โหมดการดำเนินการ (Processing Mode)

2.10.1 การรีเซตและการตอบกลับ



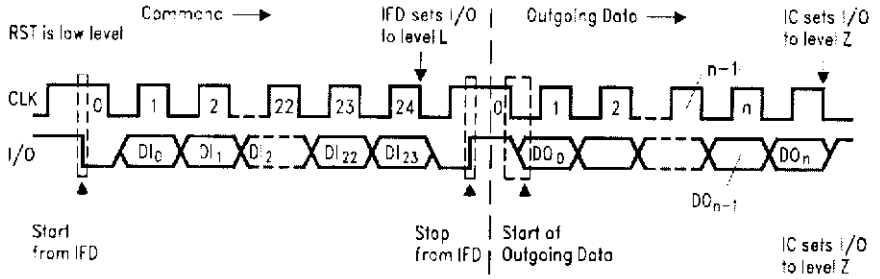
รูปที่ 2.19 สัญญาณรีเซต และ ATR

การอินเตอร์เฟสเข้ากับ Security Memory Card ทั่วๆไปทั้ง SLE4442 จะสอดคล้องกับมาตรฐานในการอินเตอร์เฟสแบบซิงโครนัสในมาตรฐาน ISO7816 โดยเมื่อรีเซตการทำงานของการ์ดจะทำให้การ์ดตอบกลับด้วยข้อมูล ATR สำหรับข้อมูลที่ตอบกลับมาจาก SLE4442 จะประกอบด้วยข้อมูล 4ไบต์

2.10.2 การทำงานของโหมดการส่งคำสั่ง

การส่งคำสั่งไปยังสมาร์ตการ์ดหรือการทำงานในโหมดการส่งคำสั่ง ก็คือกระบวนการต่อเนื่องหลังจากการ์ดถูกรีเซตเรียบร้อยแล้ว โดยการ์ดจะรอรับคำสั่งที่ส่งมาจากเครื่องอ่านซึ่งมีรูปแบบเป็นข้อมูลความยาว 3ไบต์ โครงสร้างของข้อมูลดังกล่าวประกอบด้วยคำสั่ง (Command,Control), แอดเดรส (Address), ข้อมูล (Data)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.20 การส่งคำสั่งไปยังการ์ด (IFD : Interface Device)

รูปสัญญาณที่เกิดขึ้นระหว่างการทำงานของโหมดการส่งคำสั่งจะเป็นดังรูป จะเห็นได้ว่าการส่งข้อมูลแต่ละครั้ง จะต้องมีการส่งสภาวะเริ่มต้นและสภาวะสิ้นสุดกำกับไปกับตัวข้อมูลด้วย ในที่นี้สภาวะเริ่มต้นก็คือการเปลี่ยนแปลงระดับลอจิกค่าสูงเป็นค่าต่ำที่ขา I/O ในขณะที่ลอจิกที่ขา CLK เป็นค่าสูง, ส่วนสภาวะสิ้นสุดก็คือการเปลี่ยนระดับลอจิกค่าต่ำเป็นค่าสูงที่ขา I/O ในขณะที่ระดับลอจิกที่ขา CLK เป็นค่าสูง

โดยคำสั่งทั้งหมดที่การ์ด SLE4442 รองรับถูกแสดงไว้ดังรูป

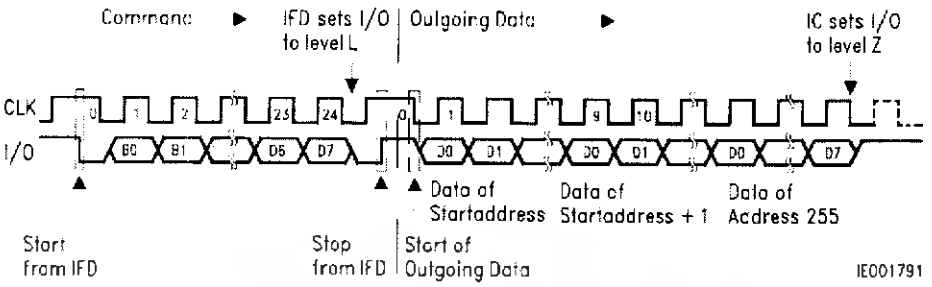
Byte 1 Control								Byte 2 Address	Byte 3 Data	Operation	Mode
B7	B6	B5	B4	B3	B2	B1	B0	A7-A0	D7-D0		
0	0	1	1	0	0	0	0	address	no effect	READ MAIN MEMORY	outgoing data
0	0	1	1	1	0	0	0	address	input data	UPDATE MAIN MEMORY	processing
0	0	1	1	0	1	0	0	no effect	no effect	READ PROTECTION MEMORY	outgoing data
0	0	1	1	1	1	0	0	address	input data	WRITE PROTECTION MEMORY	processing
0	0	1	1	0	0	0	1	no effect	no effect	READ SECURITY MEMORY	outgoing data
0	0	1	1	1	0	0	1	address	input data	UPDATE SECURITY MEMORY	processing
0	0	1	1	0	0	1	1	address	input data	COMPARE VERIFICATION DATA	processing

ตารางที่2.2 โครงสร้างและความหมายของชุดคำสั่งที่ SLE4442 รองรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10.3 ความหมายและวิธีการทำงานของคำสั่ง

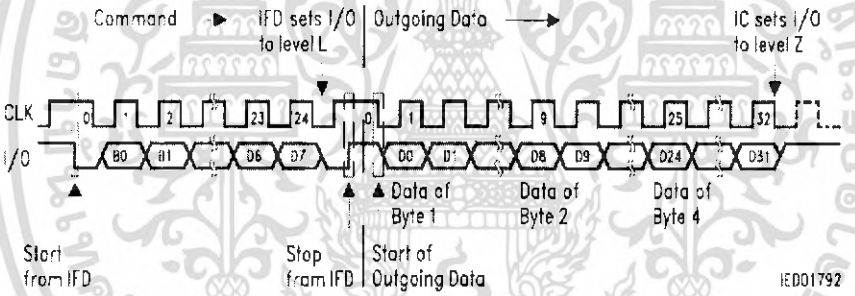
- Read Main memory



รูปที่ 2.21 สัญญาณของคำสั่ง Read Main Memory

เป็นคำสั่งที่ใช้ในการอ่านข้อมูลทั้งหมดออกมาจากหน่วยความจำของการ์ด ทั้งจากพื้นที่ส่วนที่ได้รับการป้องกัน (หน่วยความจำ 32ไบต์แรก) และส่วนที่ไม่ได้รับการป้องกัน (หน่วยความจำ 224ไบต์หลัง) โดยจะเป็นการอ่านค่าโดยเริ่มต้น จากแอดเดรสที่ส่งไปจนถึงแอดเดรสสุดท้าย (0ffh) ของพื้นที่หน่วยความจำ

- Read Protection Memory



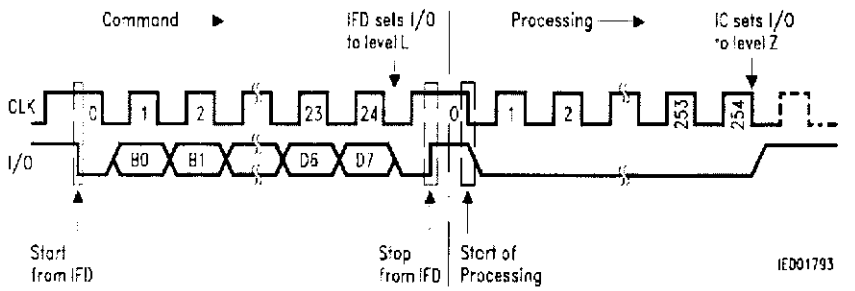
รูปที่ 2.22 สัญญาณคำสั่ง Read Protection Memory

เป็นคำสั่งที่ใช้ในการอ่านข้อมูลทั้งหมดออกมา จากหน่วยความจำ 32ไบต์ แรก

- Update Main Memory

เป็นคำสั่งที่ใช้ในการเขียนข้อมูลยังแอดเดรสใดๆ ของหน่วยความจำทั้งหมด 256ไบต์ ในกรณีที่ใช้คำสั่งในการเขียนข้อมูลยังหน่วยความจำ 32ไบต์แรก ข้อมูลจะยังคงแก้ไขเปลี่ยนแปลงได้ภายหลัง สำหรับการเขียนข้อมูลจะประกอบด้วย 3เงื่อนไข คือ

การลบข้อมูลแล้วเขียนข้อมูลซ้ำ

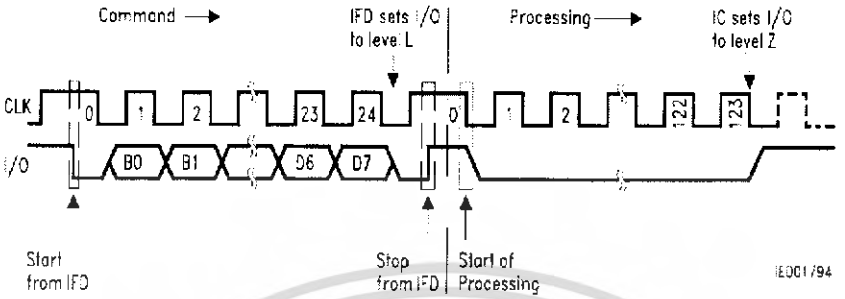


รูปที่ 2.23 การลบข้อมูลแล้วเขียนข้อมูลซ้ำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นการลบข้อมูลที่แอดเดรสของหน่วยความจำที่กำหนดให้เป็น 0FFH แล้วทำการเขียนข้อมูลซ้ำลงแอดเดรสเดิม กระบวนการนี้จะต้องใช้สัญญาณนาฬิกา 255ลูก

การลบหรือเขียนข้อมูล (อย่างใดอย่างหนึ่ง)



รูปที่ 2.24 การลบหรือเขียนข้อมูล (อย่างใดอย่างหนึ่ง)

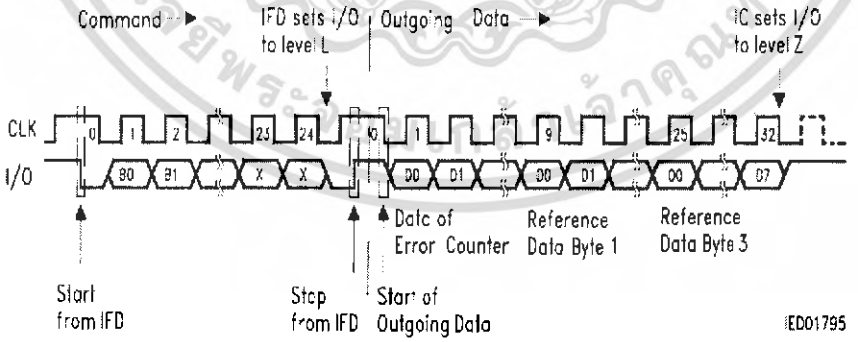
-การเขียนข้อมูลที่แอดเดรสของหน่วยความจำที่กำหนดให้โดยไม่ต้องลบข้อมูลออก สำหรับแอดเดรสดังกล่าวจะต้องเป็นที่ว่าง (มีข้อมูลเป็น 0FFH) อยู่ก่อนหน้านี้แล้วเท่านั้น กระบวนการนี้จะใช้สัญญาณนาฬิกา 124ลูก

-การลบข้อมูลที่แอดเดรสของหน่วยความจำที่กำหนด (มีค่าข้อมูลเป็น 0FFH) โดยไม่มีการเขียนข้อมูลต่อ สำหรับกระบวนการนี่จะใช้สัญญาณนาฬิกา 124ลูก เช่นกัน

- Write Protection Memory

เป็นการเขียนข้อมูลลงยังแอดเดรสหน่วยความจำใดๆ ใน 32ไบต์แรก คำสั่งนี้มีเงื่อนไขว่าข้อมูลที่เขียนลงไปจะถูกเขียนลงยังแอดเดรสของหน่วยความจำที่กำหนดอย่างถาวร ไม่สามารถแก้ไขเปลี่ยนแปลงอะไรได้อีก สำหรับรูปสัญญาณของกระบวนการนี้ อ้างอิงได้จากรูปสัญญาณของคำสั่ง Update Main Memory

- Read Security Memory



รูปที่ 2.25 สัญญาณคำสั่ง Read Security Memory

เป็นการอ่านค่าของ Error Counter เพื่อตรวจดูว่าการ์ดใบนั้นๆได้ถูกล็อกไปหรือยัง โดยค่าภายในบิต D1,D2 และD0 ของ Error Counter จะเป็นส่วนที่บอกถึงสถานะของการ์ดใบนั้น หากค่าของบิต D1,D2 และD0 เป็น 0 ทั้งหมดก็แสดงว่าการ์ดได้ถูกล็อกไปแล้ว ซึ่งไม่สามารถแก้ไขอะไรได้และจะ

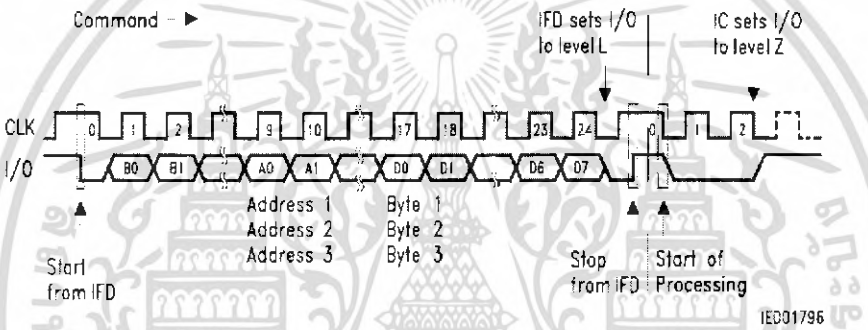
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไม่สามารถเขียนข้อมูลยังการ์ดนั้นได้อีกต่อไป (แต่ว่าการอ่านข้อมูลการ์ดจะยังคงทำได้อย่างเดิมตามปกติ)

- Update Security Memory

เป็นการเข้าไปแก้ไขข้อมูลของรหัส PSC ภายในการ์ด หรือพูดง่ายๆ ก็คือการเข้าไปเปลี่ยนรหัสของการ์ดนั่นเอง คำสั่งจะถูกกระทำต่อเมื่อมีการส่งรหัส PSC ที่ถูกต้อง ไปยังการ์ดเสียก่อน โดยในกรณีที่ป้อนรหัสผิด ค่าของบิต D1,D2 และD0 ใน Error Counter จะค่อยๆถูกเปลี่ยนจาก 1 เป็น 0ไล่ไปทีละบิตตามจำนวนครั้งที่ป้อนรหัสผิด หากทั้งหมดกลายเป็น 0 การ์ดก็จะถูกล็อก สัญญาณของกระบวนการนี้จะเหมือนกับรูปสัญญาณของคำสั่ง Update Main Memory

- Compare Verification Data

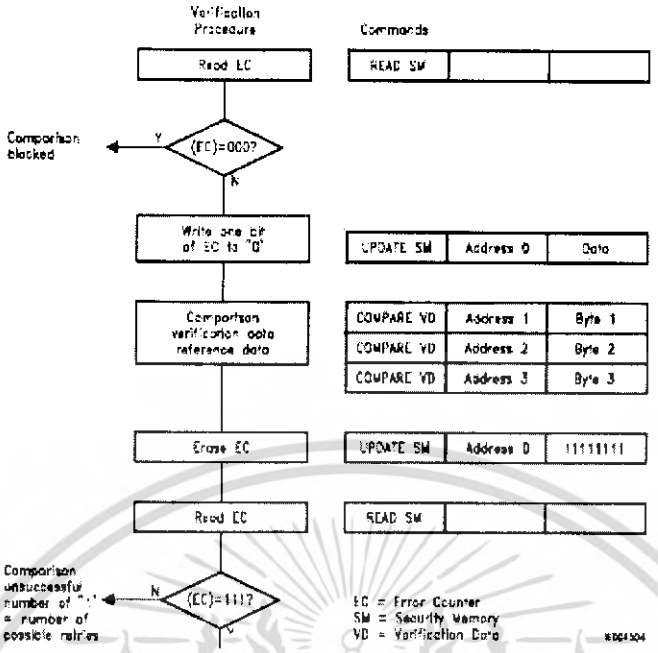


รูปที่ 2.26 สัญญาณคำสั่ง Compare Verification Data

เป็นการสั่งให้การ์ดทำการเปรียบเทียบรหัสผ่านที่เราได้ส่งไปกับ Reference data ในการเปรียบเทียบที่ว่านี้ ข้อมูลที่การ์ดส่งกลับมาคือค่าของ Error Counter ที่บอกว่ารหัสที่เราป้อนไปนั้นถูกต้องหรือไม่ และยังเหลือโอกาสพลาดอีกกี่ครั้งเท่านั้น (เราไม่สามารถจะเข้าไปอ่านค่ารหัส PSC ของการ์ดออกมาได้)

ในการใช้งานคำสั่ง PSC นั้น มีคำสั่งที่เกี่ยวข้องอยู่ 3คำสั่ง คือ Read Security Memory, Read Security Memory, Compare Verification Data โดยกระบวนการ PSC สามารถเขียนแสดงเป็น Flow Chart ได้ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.27 Flow Chart ของกระบวนการ PSC

2.10.4 โหมดการอ่านข้อมูล (Outgoing Data Mode)

โหมดการทำงานนี้จะเกิดขึ้นหลังจากที่มีการสั่งคำสั่งในกลุ่มของการอ่านข้อมูล (เช่น Read Main Memory, Read Protection Memory, Read Security Memory) ไปยังการ์ดเพื่อขออ่านข้อมูลจากพื้นที่ใดๆ ในหน่วยความจำหลังจากได้รับคำสั่งดังกล่าว สมาร์ตการ์ดจะส่งข้อมูลที่ถูกร้องขอกลับมายังเครื่องอ่านซึ่งก็เท่ากับว่าเครื่องอ่านจะสามารถอ่านข้อมูลที่ต้องการออกมาได้สำเร็จจากโหมดการทำงานนี้

2.10.5 โหมดดำเนินการ (Processing Mode)

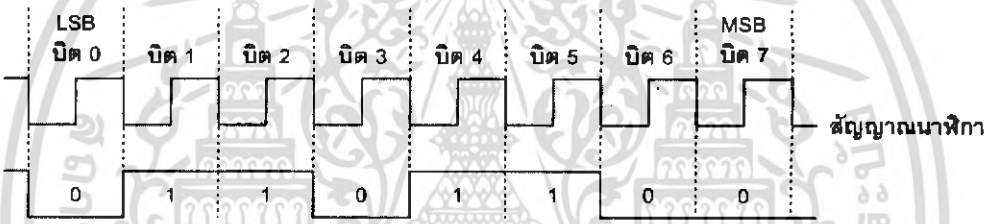
โหมดการดำเนินการจะเกิดขึ้นหลังจากที่มีการส่งคำสั่ง ในกลุ่มของการขอเขียนหรือลบข้อมูลออกจากพื้นที่ใดๆ ในหน่วยความจำ (เช่น Update Main Memory, Write Protection Memory, Update Security Memory และ Compare Verification Data) โดยหลังจากได้รับคำสั่งดังกล่าวสมาร์ตการ์ดจะเริ่มดำเนินกระบวนการตามที่ได้รับการสั่งมาในระหว่างโหมดการทำงานนี้จะสังเกตว่าสัญญาณจากขา I/O จะไม่ถูกนำมาใช้ร่วมในการทำงานเลย (เนื่องจากมีสถานะเป็นลอจิกต่ำตลอดเวลาทั้งช่วง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3 การสื่อสารแบบอนุกรม

3.1 การสื่อสารแบบอนุกรม

การสื่อสารแบบอนุกรมนั้นจะแบ่งออกได้เป็น 2 แบบคือการสื่อสารอนุกรมแบบซิงโครนัสและการสื่อสารอนุกรมแบบอะซิงโครนัส การสื่อสารแบบซิงโครนัสจะมีสัญญาณนาฬิการ่วมอยู่กับการรับและส่งสัญญาณด้วย ตัวอย่างการส่งข้อมูลแบบซิงโครนัสก็คือคีย์บอร์ดของคอมพิวเตอร์ ซึ่งสายเส้นหนึ่งจะเป็นสายของสัญญาณนาฬิกา ส่วนสายอีกเส้นจะเป็นสายของข้อมูล ดังนั้นการติดต่อกันแบบซิงโครนัสนี้ต้องใช้สายในการเชื่อมต่ออย่างน้อยที่สุด 3 เส้น คือ สัญญาณนาฬิกา , ข้อมูล และ กราวด์ รูปที่ 3.1 แสดงให้เห็นถึงไทมิ่งค์ไดอะแกรมของการส่งข้อมูล แบบ ซิงโครนัส



รูปที่ 3.1 รูปแบบอย่างง่ายที่สุดของข้อมูลอนุกรม

3.2 การสื่อสารข้อมูลแบบอะซิงโครนัส

การสื่อสารข้อมูลแบบอะซิงโครนัส คือ การรับและส่งข้อมูลไปในสายโดยไม่จำเป็นต้องมีสัญญาณนาฬิการ่วมด้วยเหมือนแบบการรับส่งข้อมูลแบบซิงโครนัส แต่จะใช้การกำหนดค่าสัญญาณนาฬิกาทั้งภาครับและภาคส่งให้มีค่าเท่ากัน ซึ่งเรียกสัญญาณนาฬิกาที่ใช้ในการกำหนดค่าให้ภาครับและภาคส่งนี้ว่า อัตราการถ่ายทอดข้อมูล หรือ บอดเรต (Baudrate) มีหน่วยเป็น บิตต่อวินาที (bit per second : bps)

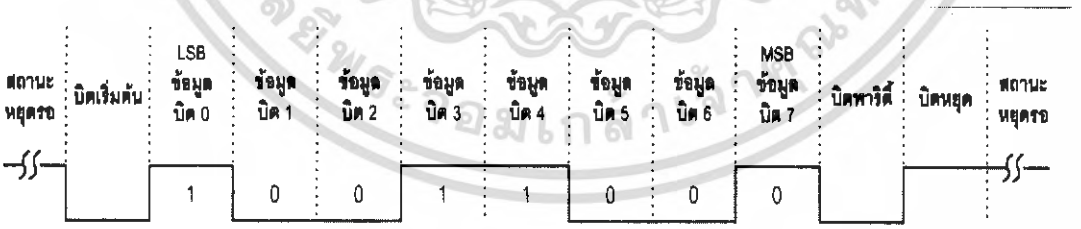
รูปแบบของข้อมูลที่ใช้ในการรับส่งแบบอะซิงโครนัสประกอบด้วย 4 ส่วนด้วยกัน คือ

1. บิตเริ่มต้น (Start Bit) ซึ่งจะมีขนาด 1 บิต
2. บิตข้อมูลแบบอนุกรมจะมีขนาด 5 , 6 , 7 หรือ 8 บิต
3. บิตตรวจสอบพาริตี้ (Parity Bit) จะมีขนาด 1 บิต หรือไม่มี
4. บิตปิดท้าย (Stop Bit) จะมีขนาด 1 , 1.5 หรือ 2 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.2 แสดงรูปแบบของข้อมูลอนุกรมแบบอะซิงโครนัส เมื่อไม่มีข้อมูลที่จะส่ง ขา Data จะมีสถานะลอจิก "1" ซึ่งจะเรียกสถานะนี้ว่าสถานะหยุดรอ (waiting stage) การเริ่มต้นส่งข้อมูลจะเริ่มจากการให้ขา Data มีลอจิก "0" ด้วยช่วงระยะเวลา 1 บิต เรียกบิตนี้ว่า บิตเริ่มต้น จากนั้นบิดข้อมูลจะถูกส่งออกไป โดยเริ่มจากบิตที่มีนัยสำคัญต่ำสุด (LSB) ก่อน ซึ่งข้อมูลในไบต์ที่จะส่งอาจจะมีจำนวนบิต 5 , 6 , 7 หรือ 8 บิตก็ได้ จากนั้นตามด้วย บิตพาริตี้ ซึ่งใช้เพื่อตรวจสอบความผิดพลาดที่เกิดขึ้นจากการส่งข้อมูล บิดสุดท้ายที่ส่งคือ บิตปิดท้าย ซึ่งจะให้ขาดาต้ามีสถานะลอจิก "1" อีกครั้งด้วยระยะเวลาอย่างน้อย 1 บิต , 1.5 บิต หรือ 2 บิต เพื่อเป็นการแสดงว่าสิ้นสุดข้อมูลแล้ว

อุปกรณ์พิเศษที่ได้รับการออกแบบมาสำหรับการรับและส่งข้อมูลแบบอะซิงโครนัสเรียกว่า Universal Asynchronous Receiver/Transmitter หรือ UART อัตราความเร็วในการรับและส่งข้อมูลของการรับส่งข้อมูลแบบอะซิงโครนัส คือ บอดเรต ซึ่งก็คือค่าจำนวนบิตต่อวินาทีที่ใช้ในการรับและส่งข้อมูล บอดเรตมาตรฐานที่ใช้สำหรับพอร์ตอนุกรม RS-232 ได้แก่ 110 , 150 , 300 , 600 , 1200 , 2400 , 4800 , 9600 และ 19200 บิตต่อวินาที และมีค่าเพิ่มมากขึ้นตามเทคโนโลยีของคอมพิวเตอร์ซึ่งการรับส่งแบบอนุกรมโดยไม่ผ่านโมเด็มอาจจะสามารถกำหนดค่าบอดเรตได้สูงถึง 115200 บิตต่อวินาที เนื่องจากบอดเรตคือจำนวนบิตของข้อมูลที่สามารถถ่ายทอดได้ภายใน 1 วินาที ยกตัวอย่าง ข้อมูลอนุกรมถูกส่งในลักษณะ 8 บิต ไม่มีการตรวจสอบพาริตี้ มีบิตเริ่มต้น 1 บิต และบิตเปิดปิดท้าย 1 บิต ความยาวของข้อมูลที่รับส่งนี้เท่ากับ 10 บิต ถ้าใช้บอดเรตในการส่งข้อมูลเท่ากับ 9600 บิตต่อวินาที ก็สามารถรับส่งข้อมูลได้ด้วยอัตราเร็ว 960 ไบต์ต่อวินาที และถ้ามีการใช้พาริตี้ความเร็วในการรับส่งจะเหลือ 872 ไบต์แต่วินาที



รูป 3.2 รูปแบบอย่างง่ายที่สุดของข้อมูลอนุกรมแบบอะซิงโครนัส

การตรวจสอบพาริตี้สามารถกำหนดให้เป็นแบบคี่(odd) , แบบคู่ (even) หรือไม่มีการตรวจสอบพาริตี้ก็ได้ การตรวจสอบพาริตี้เป็นการตรวจสอบจำนวนรวมของบิตที่เป็นลอจิกสูง ภายในข้อมูลที่ส่งไป 1 ไบต์ว่ามีจำนวนรวมเป็นเลขคู่หรือเลขคี่โดยต้องรวมบิตพาริตี้เข้าไปด้วย ยกตัวอย่างข้อมูลที่จะทำการส่งมีขนาด 8 บิต และมีค่าเท่ากับ 99 ฐานสิบหก หรือ 10011001 ฐานสอง จะเห็นว่า

ไม่วากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลในไบต์นี้มีจำนวนลอจิก "1" จำนวน 4 ตัว ซึ่งเป็นเลขคู่ ดังนั้นถ้ากำหนดค่าพาริตี้เป็นคู่ ค่าในบิตพาริตี้ จะต้องมีลอจิกเป็น "0" แต่ถ้าพาริตี้เป็นคี่ ค่าที่บิตพาริตี้จะต้องเป็น "1" เพื่อให้ข้อมูล 1 ไบต์ รวมทั้งบิตพาริตี้มีจำนวนบิตที่ลอจิก "1" รวมกันเป็นเลขคี่ ในตาราง 3.1 แสดงตัวอย่างของบิตพาริตี้ในการรับส่งข้อมูลอนุกรม

บิตพาริตี้ถูกสร้างขึ้นจากภาคส่งข้อมูลของ UART โดยภาครับจะต้องกำหนดคุณสมบัติการตรวจสอบพาริตี้ให้ตรงกันว่า จะตรวจสอบพาริตี้คี่หรือคู่ จากนั้นภาครับของ UART จะตรวจสอบค่าพาริตี้ที่เกิดขึ้นว่าเป็นคู่หรือ เป็นคี่ โดยการนับจำนวนลอจิก "1" ทั้งหมดรวมทั้งบิตพาริตี้ด้วย ถ้ากำหนดพาริตี้ไว้เป็นคู่แต่อ่านค่าตัวเลขในการนับออกมาได้ตัวเลขเป็นคี่ ทางภาครับจะแจ้งข้อผิดพลาดให้ผู้ใช้ทราบ นับเป็นการตรวจสอบความผิดพลาดของการถ่ายทอดข้อมูลที่ง่ายที่สุด แต่จะเชื่อถือได้เมื่อมีบิตข้อมูลที่ทำการส่งผิดพลาดเพียงบิตเดียวเท่านั้น ถ้าข้อมูลที่ส่งมีบิตที่ผิดพลาดมากกว่า 1 บิต การตรวจสอบด้วยวิธีนี้จะไม่ได้ผล สำหรับการตั้งพาริตี้บิดเป็น NONE นั้นทั้งภาครับและภาคส่ง จะไม่มีการตรวจสอบพาริตี้

ข้อมูล	บิตพาริตี้คู่	บิดพาริตี้คี่
00000000	0	1
00000001	1	0
00000010	1	0
00000011	0	1
00000100	1	0
11111110	1	0
11111111	0	1

ตารางที่ 3.1 แสดงบิตพาริตี้ของข้อมูล

คอมพิวเตอร์ในรุ่น AT เกือบทั้งหมดจะใช้ UART เบอร์ 16450 และ 16550 ส่วนคอมพิวเตอร์ในรุ่นXT ใช้ UART เบอร์ 8250 UART ชิปเหล่านี้มีระดับแรงดันเป็นแบบทีทีแอล (0 และ +5 V) แต่เพื่อให้แรงดันเป็นไปตามมาตรฐาน RS-232 และเพื่อให้การรับส่งข้อมูลสามารถทำได้ที่ระยะทางไกลมากขึ้น ระดับแรงดันทีทีแอลจะถูกแปลงเป็นระดับแรงดันที่สูงขึ้นโดยลอจิก "0" มีระดับแรงดัน +3V ถึง +12 V ในขณะที่ลอจิก "1" มีระดับแรงดัน -3 V ถึง -12 V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 มาตรฐานพอร์ตอนุกรมแบบ RS-232

มาตรฐานการเชื่อมต่อแบบอนุกรม RS-232 เป็นมาตรฐานอุตสาหกรรมที่ออกแบบมาเพื่อใช้ในการส่งข้อมูลแบบอะซิงโครนัส 2 ทิศทาง โดยมาตรฐาน RS-232 ในอดีตนั้นถูกออกแบบมาเพื่อการส่งผ่านข้อมูลจากคอมพิวเตอร์ไปยังโมเด็มเพียงอย่างเดียว เพื่อที่จะนำข้อมูลจากโมเด็มนี้สื่อสารผ่านสายโทรศัพท์ไปยังคอมพิวเตอร์อีกชุดซึ่งอยู่ห่างไกลกัน โดยคณะกรรมการที่เรียกว่า สมาคมอุตสาหกรรมอิเล็กทรอนิกส์ (Electronic Industries Association : EIA) ได้วางมาตรฐานที่มีชื่อเรียกกันว่า EIA RS-232 มาตรฐานนี้ในช่วงแรกจะใช้คอนเน็กเตอร์เป็นแบบ DB-25 โดยกำหนดความยาวสูงสุดของสายสัญญาณไว้ที่ 50 ฟุต มีระดับสัญญาณตั้งแต่ -3 ถึง -I2 V แสดงว่ามีข้อมูล (Mark) และ +3 ถึง +12 V แสดงว่าเป็นช่องว่าง(Space)

มาตรฐานRS-232 ได้กำหนดรูปแบบของอุปกรณ์เชื่อมต่อข้อมูล (Data Terminal Equipment: DTE) กับวงจรข้อมูลปลายทาง (Data Circuit Terminating: DCE) ไว้ว่า อุปกรณ์ DTE จะต้องเป็นอุปกรณ์ที่มีการประมวลผลในตัว เช่น ไมโครคอนโทรลเลอร์ หรือ ไมโครคอมพิวเตอร์ ซึ่งมีความสามารถในการสร้างข้อมูลแบบอนุกรมได้ ส่วนอุปกรณ์ DCE จะทำหน้าที่เป็นเพียงตัวรับข้อมูลที่ส่งมาจาก DTE เท่านั้น โดยการรับส่งข้อมูลระหว่างอุปกรณ์ทั้งสองจะกระทำผ่านมาตรฐาน RS-232

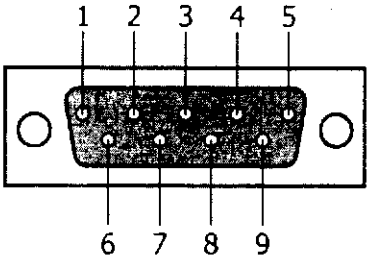
ข้อแตกต่างของอุปกรณ์ DTE และอุปกรณ์ DCE อย่างหนึ่งที่เห็นได้ชัดคือ คอนเน็คเตอร์ของ DTE จะเป็นตัวผู้ ส่วนคอนเน็คเตอร์ของ DCE จะเป็นตัวเมีย ซึ่งพอร์ตอนุกรมของคอมพิวเตอร์ที่ใช้กันอยู่ทั่วไปจะเป็นแบบ DTE ส่วนคอนเน็คเตอร์ที่อยู่ที่โมเด็มจะเป็นแบบ DCE

สำหรับการใช้งานบนคอมพิวเตอร์ พอร์ตอนุกรม RS-232 มักถูกใช้เชื่อมต่อกับโมเด็มหรือเมาส์ โดยสามารถรับส่งข้อมูลได้ที่ความยาวของสายสัญญาณสูงสุด ถึง 20 เมตร

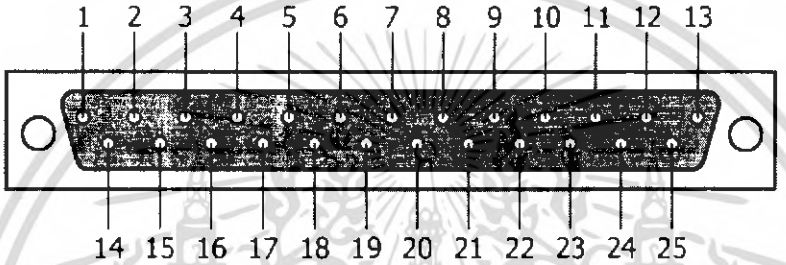
3.3.1คอนเน็คเตอร์สำหรับพอร์ต RS-232 และการเชื่อมต่อ

มาตรฐานการเชื่อมต่อแบบ RS-232 จะใช้คอนเน็คเตอร์แบบ DB-25 ตัวผู้หรือ DB-9 ตัวผู้ซึ่งคอนเน็คเตอร์แบบ DB-25 จะมีขาต่อใช้งานเพียง 9 เส้นเช่นเดียวกับคอนเน็คเตอร์แบบ DB-9 เนื่องจากขาอื่นๆ ที่เคยใช้งานในอดีต ปัจจุบันมีการใช้งานไม่มากนัก จึงถูกยกเลิกไป โดยแสดงรูปร่างและตำแหน่งขาในรูปที่ 3.3 และตาราง 3.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 ก



รูปที่ 3.3 ข

รูปที่ 3.3 คอนเนคเตอร์อนุกรม

รูป3.3 ก คอนเนคเตอร์อนุกรม 9 ขาหรือแบบ DB-9 (มองจากด้านหลังคอมพิวเตอร์)

รูป3.3 ข คอนเนคเตอร์อนุกรม 25 ขาหรือแบบ DB-25 (มองจากด้านหลังคอมพิวเตอร์

คอนเน็กเตอร์ DB-9	คอนเน็กเตอร์ DB-25	ชื่อของสายสัญญาณ	ชนิดของสายสัญญาณ
1	8	Data Carrier Detect : DCD	อินพุต
2	3	Received Data : RxD	อินพุต
3	2	Transmitted Data : TxD	เอาต์พุต
4	20	Data Terminal Ready : DTR	เอาต์พุต
5	7	Signal Ground : GND	-
6	6	Data Set Ready : DSR	อินพุต
7	4	Request To Send : RTS	เอาต์พุต
8	5	Clear To Send : CTS	อินพุต
9	22	Ring Indicator : RI	อินพุต

ตารางที่ 3.2 หน้าที่การทำงานของขาอนุกรม

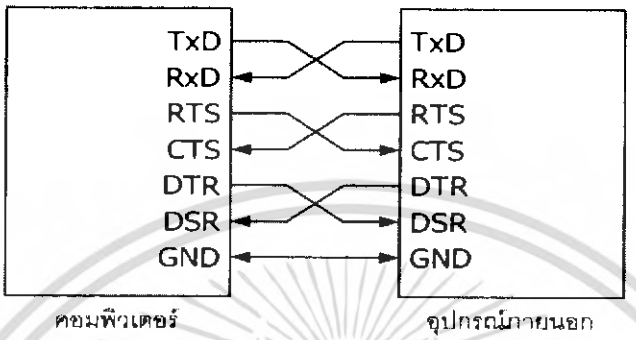
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับการเชื่อมต่อคอมพิวเตอร์กับอุปกรณ์ภายนอกแสดงดังในรูปที่ 3.4 ลูกศรในรูปแสดงถึงทิศทางของข้อมูล ในรูปที่ 3.4 ก เป็นการเชื่อมต่อแบบ Null modem หรือการเชื่อมต่อโดยไม่ต้องผ่านโมเด็ม โดยมีการตรวจสอบหรือแฮนด์เช็คเต็มรูปแบบ ส่วนในรูปที่ 3.4 ข เป็นการเชื่อมต่อแบบ Null modem ในลักษณะที่ใช้สายสัญญาณเพียง 3 เส้น โดยเส้นหนึ่งสำหรับส่งข้อมูล อีกเส้นสำหรับข้อมูลและเส้นสุดท้ายเป็นกราวด์ สำหรับรายละเอียดหน้าที่การทำงานในแต่ละขาของพอร์ตอนุกรม RS-232 มีดังนี้

- Data Carrier Detect : DCD หรืออาจเรียกว่า Carrier Detect :CD ขานี้จะแอกทีฟเมื่อมีการส่งสัญญาณพาห์จากอุปกรณ์สื่อสารข้อมูล เช่น โมเด็ม สำหรับการใช้งานปกติ ขานี้จะไม่ได้ถูกใช้งานมากนัก
- Receive Data :RD หรือ RxD ขานี้ใช้เพื่อรับสัญญาณอนุกรมเข้ามายังคอมพิวเตอร์ โดยนำข้อมูลที่อ่านได้เก็บไว้ในรีจิสเตอร์ บัฟเฟอร์
- Transmitted Data :TD หรือ TxD ใช้ส่งข้อมูลออกจากคอมพิวเตอร์ โดยนำข้อมูลที่เก็บอยู่ในบัฟเฟอร์สำหรับส่งข้อมูลส่งออกไป
- Data Terminal Ready : DTR เป็นขาสัญญาณที่ส่งออกมาจากคอมพิวเตอร์เพื่อให้อุปกรณ์ปลายทางรับรู้ว่า ต้องการติดต่อด้วย โดยขา DTR นี้ต้องเชื่อมต่อกับขา DSR ของอุปกรณ์ปลายทาง และขา DTR ของอุปกรณ์ปลายทางต้องเชื่อมต่อกับขา DSR ของคอมพิวเตอร์ ถ้าใช้การเชื่อมต่อเป็นแบบ Null modem ซึ่งใช้สายในการเชื่อมต่อเพียง 3 เส้น จะต้องต่อขา DTR และ DSR ของตัวเองเข้าด้วยกันและต้องต่อกับขา DCD ด้วยในกรณีที่โปรแกรมสื่อสารที่ใช้มีการตรวจจับสัญญาณพาห์
- Signal Ground : GND กราวด์ระบบ
- Data Set Ready :DSR ขานี้จะใช้คู่กับขาDTR เพื่อตรวจสอบการเชื่อมต่อกันระหว่างคอมพิวเตอร์กับอุปกรณ์ปลายทาง ซึ่งขาDSR นี้จะเป็นขาสำหรับรับข้อมูลจากภายนอกซึ่งถูกส่งมาจากขาDTR
- Request to Send :RTS เป็นขาสำหรับส่งสัญญาณร้องขอให้ทางอุปกรณ์ปลายทางส่งข้อมูลกลับมายังคอมพิวเตอร์ โดยขาที่รับสัญญาณRTS ก็คือขาCTS ในกรณีที่ใช้การเชื่อมต่อแบบ Null modem 3 สาย จะต้องเชื่อมต่อขา RTS และ CTS ของตัวมันเองเข้าด้วยกัน เพื่อจะให้การรับและส่งข้อมูลสามารถเกิดขึ้นได้ตลอดเวลา
- Clear To Send :CTS ขานี้จะคอยรับสัญญาณจากขาRTS เมื่อรับสัญญาณได้ ข้อมูลที่ขาTxD จะถูกส่งออกไป ดังนั้นขานี้จึงถูกใช้เพื่อตรวจสอบอุปกรณ์ต่อพ่วงว่าพร้อมที่จะรับข้อมูลหรือไม่

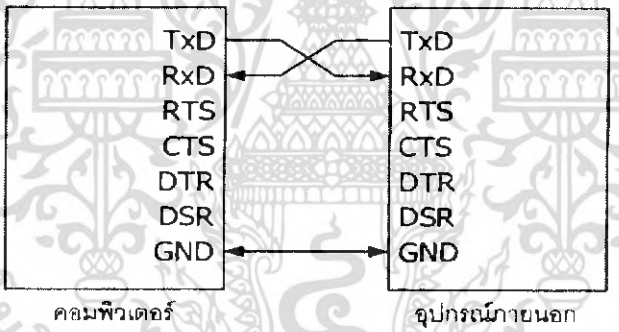
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Ring Indicator :RI ใช้แสดงสถานะสัญญาณเรียกจากสายโทรศัพท์ ปกติในการสื่อสารโดยทั่วไปสายนี้จะไม่ถูกใช้งาน จะใช้งานก็ต่อเมื่อมีการเชื่อมต่อกับโมเด็มและโปรแกรมมีการตรวจสอบสัญญาณนี้เท่านั้น



การต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์
แบบ Null modem

รูปที่ 3.4 ก



การต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์
แบบ RS-232 โดยใช้สายสัญญาณเพียง 3 เส้น

รูปที่ 3.4 ข

รูปที่ 3.4 การต่ออุปกรณ์ภายนอกกับพอร์ตอนุกรมของคอมพิวเตอร์ในลักษณะต่างๆ

3.3.2 UART (Universal Asynchronous Receiver Transmitter)

เป็นอุปกรณ์ที่ทำหน้าที่รับและส่งข้อมูลแบบอะซิงโครนัสนั่นเอง สำหรับการสื่อสารอนุกรมบนคอมพิวเตอร์แล้ว UART ถือว่าเป็นหัวใจสำคัญของการสื่อสารอนุกรม

หน้าที่หลักของUART คือทำหน้าที่แปลงข้อมูลที่อยู่ในรูปแบบขนานจากคอมพิวเตอร์ให้อยู่ในรูปแบบอนุกรมแบบอะซิงโครนัส แล้วส่งออกไป และทำหน้าที่แปลงสัญญาณอนุกรมแบบอะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซิงโครนัสที่ป้อนเข้ามายัง UART ให้เป็นแบบขนานก่อนที่จะส่งเข้าสู่คอมพิวเตอร์ ซึ่งนอกจาก UART จะส่งข้อมูลไปยังคอมพิวเตอร์แล้ว ยังแจ้งข้อมูลอื่นๆ ให้คอมพิวเตอร์รับทราบด้วย เช่น อัตราเร็วในการรับส่งข้อมูล (บอดเรต) , รูปแบบการส่งข้อมูล , ความผิดพลาดที่เกิดขึ้นระหว่างการถ่ายทอดข้อมูล (ผิดพลาดจากพาริตี้ , เฟรมข้อมูล , โอเวอร์รัน) เป็นต้น

ภายใน UART จะมีส่วนของวงจรสร้างบอดเรตแบบโปรแกรมได้ (Programmable buadrate generator) โดยการกำหนดค่าตัวหารให้กับสัญญาณนาฬิกา UART โดยตัวหารนี้มีขนาด 16 บิต ดังนั้นจึงสามารถกำหนดตัวหารอยู่ในช่วง 1 - 65,535 UART สามารถรับส่งข้อมูลได้ทั้งแบบฮาล์ฟดูเพล็กซ์ (half duplex) และฟูลดูเพล็กซ์ (full duplex) โดยการส่งแบบฮาล์ฟดูเพล็กซ์เป็นการส่งแบบทิศทางเดียว ส่วนการส่งแบบฟลูดูเพล็กซ์นั้นสามารถรับและส่งข้อมูลได้ในคราวเดียวกัน

3.3.3ชนิดของ UART

ในเครื่องคอมพิวเตอร์ทั่วไปมีUART ที่ใช้งานกันอยู่ 2 เบอร์ คือ 8250 ซึ่งเป็นUART มาตรฐานที่มีใช้กันมายาวนาน UART เบอร์นี้จะมีบัฟเฟอร์สำหรับรับและส่งข้อมูลตัวแหน่งเดียวกัน ทำให้การรับและส่งข้อมูลถูกจำกัดความเร็วอยู่ที่ 57.6 กิโลบิตต่อวินาทีเท่านั้น แต่UART เบอร์นี้ก็ถือว่าเป็นต้นแบบของ UART ที่ใช้ในคอมพิวเตอร์ โดยคอมพิวเตอร์ทุกๆรุ่นจะต้องสนับสนุนการทำงานตามรูปแบบของUART เบอร์นี้

UART อีกเบอร์หนึ่งคือ 16450 มีความสามารถรับส่งข้อมูลได้ที่ความเร็ว 115,200 บิตต่อวินาที และเพิ่มรีจิสเตอร์สำหรับพักข้อมูลสำหรับ UART นอกจากนั้นยังเพิ่มส่วนของชิฟต์รีจิสเตอร์แบบ FIFO (First IN First Out) ขนาด 16 ไบต์เข้าไปทำให้สามารถสนับสนุนความเร็วในการรับส่งข้อมูลที่ 256 กิโลบิต่อวินาทีได้ โดยคอมพิวเตอร์ในปัจจุบันใช้ UART เบอร์นี้หรือใหม่กว่า เช่น เบอร์ TL16C750 ซึ่งมีรีจิสเตอร์แบบFIFO ขนาด 64 ไบต์ ทำงานได้ที่ระดับแรงดัน +5V และ +3V มีโหมดประหยัดพลังงาน สามารถรับส่งข้อมูลได้ที่ความเร็ว 1 เมกะบิตต่อวินาทีเมื่อใช้สัญญาณนาฬิกา 16 MHz

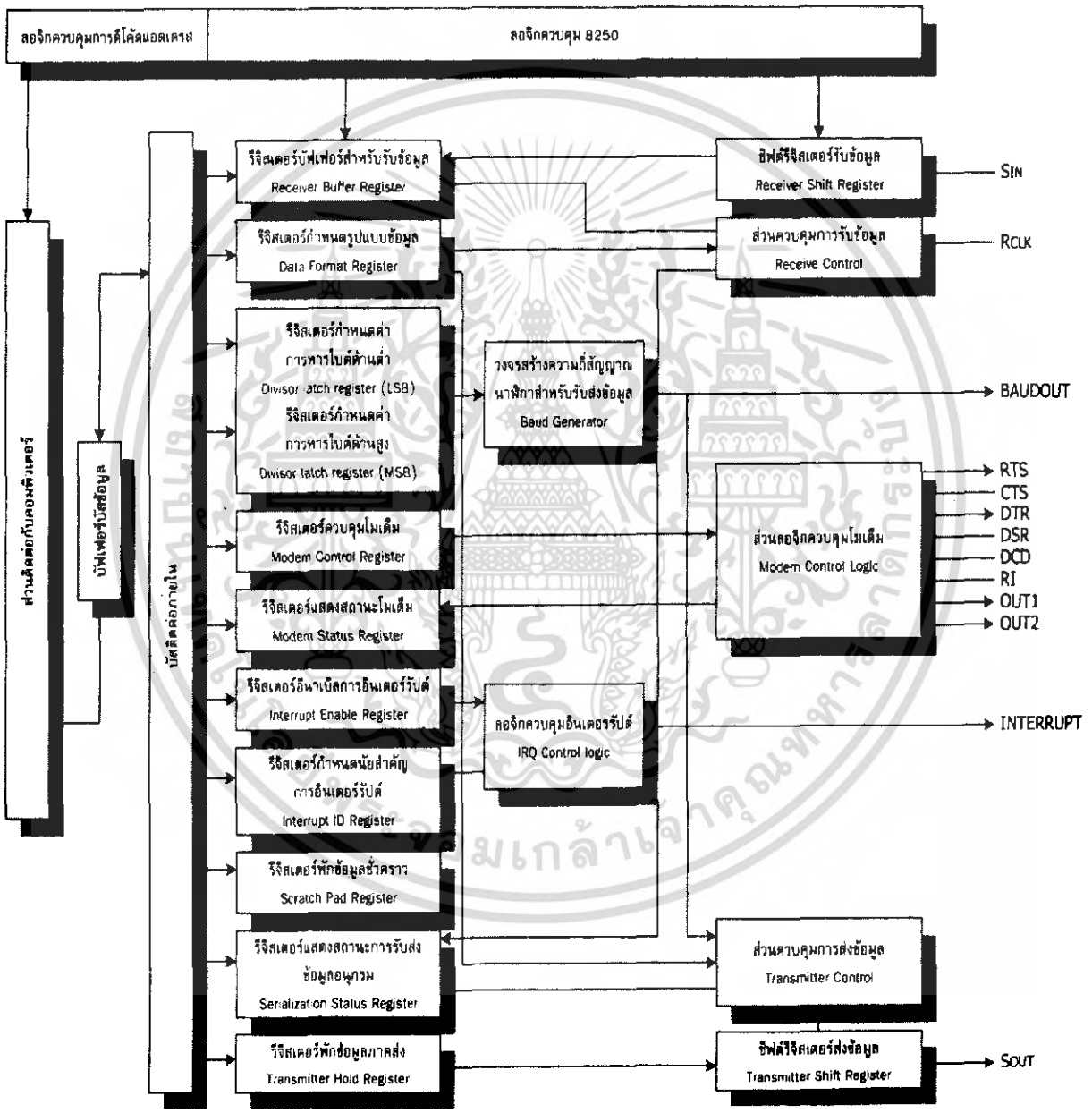
อย่างไรก็ตาม ความเร็วในการส่งข้อมูลที่มากมายของ UART เบอร์ใหม่ ๆ ก็ไม่ได้ช่วยให้การรับส่งข้อมูลของคอมพิวเตอร์เร็วขึ้น เนื่องจากว่าคอมพิวเตอร์ยังใช้ความถี่ของสัญญาณนาฬิกาในการแปลงข้อมูลเพียง 1.8432 MHZ เท่านั้น

3.4วงจรภายในและรีจิสเตอร์ของพอร์ตอนุกรม RS-232

เครื่องคอมพิวเตอร์โดยทั่วไปสามารถต่อพอร์ตอนุกรมสูงสุดได้ 4 พอร์ด มีชื่อเรียกเป็น COM1,COM2,COM3 และ COM4 ซึ่งพอร์ตอนุกรมแต่ละตัวต่างก็ใช้งาน UART ภายในคอมพิวเตอร์ในการติดต่อกับอุปกรณ์ภายนอกเช่นเดียวกัน ในรูปที่ 3.5 แสดงผังการทำงานภายในของพอร์ตอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประกอบด้วยรีจิสเตอร์ 8 บิต 8 ตัว ที่ใช้งานร่วมกับ UART แอดเดรสของรีจิสเตอร์ภายในพอร์ตอนุกรม สามารถคำนวณได้จากค่ารีจิสเตอร์พื้นฐานของพอร์ตอนุกรม ยกตัวอย่าง พอร์ตอนุกรม COM1 มีแอดเดรสอยู่ที่ 3F8H ตำแหน่งของรีจิสเตอร์ต่างๆ จะเป็นตำแหน่งที่บวกเข้าไปกับค่า 3F8H โดยรีจิสเตอร์ที่ใช้งานกับพอร์ตอนุกรมมีดังนี้



รูปที่ 3.5 ไดอะแกรมการทำงานภายในของพอร์ตอนุกรมของเครื่องคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

00H เป็นรีจิสเตอร์บัฟเฟอร์สำหรับเก็บข้อมูลที่รับเข้าหรือเตรียมข้อมูลก่อนที่จะส่งออกไป

01H รีจิสเตอร์เอ็นเนเบิลการอินเตอร์รัปต์ ใช้เซตโหมดการอินเตอร์รัปต์ของพอร์ตอนุกรม

02H รีจิสเตอร์แสดงโหมดการอินเตอร์รัปต์ ใช้เพื่อตรวจสอบโหมดของการอินเตอร์รัปต์

03H รีจิสเตอร์กำหนดรูปแบบขอข้อมูล

04H รีจิสเตอร์ควบคุมโมเด็ม ใช้ตรวจสอบบิตสำหรับติดต่อกับโมเด็ม เช่น RTS หรือ DTR

05H รีจิสเตอร์แสดงสถานการณ์รับและการส่งข้อมูลแบบอนุกรม

06H รีจิสเตอร์แสดงสถานะของโมเด็ม ซึ่งจะแสดงสถานะของขา DCD ,RI,DSR และ CTS

07H รีจิสเตอร์สำหรับการเก็บข้อมูลชั่วคราว

3.4.1 รีจิสเตอร์ตำแหน่ง 00H (รีจิสเตอร์บัฟเฟอร์)

เป็นรีจิสเตอร์เก็บข้อมูลที่รับเข้ามาและส่งออก โดยการติดต่อกับรีจิสเตอร์นี้เพื่อเก็บข้อมูลจะต้องกำหนดให้บิตDLAB ในรีจิสเตอร์กำหนดรูปแบบข้อมูล(03H) มีสถานะเป็น “0” ซึ่งการเขียนข้อมูลมายังแอดเดรสนี้เป็นการส่งข้อมูลไปยังรีจิสเตอร์ส่งข้อมูลและข้อมูลจะถูกส่งออกไปแบบอนุกรม สำหรับการรับข้อมูล เมื่อรับเข้ามาแล้ว จะส่งต่อไปยังรีจิสเตอร์เก็บข้อมูล หลังจากอ่านค่าจากรีจิสเตอร์นี้ออกไป รีจิสเตอร์นี้จะถูกเคลียร์ และเตรียมพร้อมสำหรับรับข้อมูลในไบต์ต่อไป

3.4.2 รีจิสเตอร์ตำแหน่ง 01H (รีจิสเตอร์เอ็นเนเบิลการอินเตอร์รัปต์)

เป็นรีจิสเตอร์สำหรับการเอ็นเนเบิลการอินเตอร์รัปต์ ซึ่งเป็นการกำหนดให้UART สร้างสัญญาณอินเตอร์รัปต์ขึ้นมา ฟังก์ชันการทำงานในแต่ละบิตของรีจิสเตอร์นี้มีดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
0	0	0	0	SINP	ERBK	TBE	RxRD

ตารางที่ 3.3 ฟังก์ชั่นการทำงานของรีจิสเตอร์ตำแหน่ง 01H

บิต 4-7 บิตเหล่านี้ไม่ถูกใช้งาน กำหนดให้เท่ากับ“0”

SINP เอ็นเอเบิลการอินเตอร์รัปต์เนื่องจากเกิดจากเปลี่ยนสถานะที่ขาอินพุต CTS,DSR,DCD หรือขา RI

“1” เอ็นเอเบิลการอินเตอร์รัปต์

“0” ดิสเอเบิล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ERBK เอ็นเอเบิลการอินเตอร์รัปต์เนื่องจากเกิดความผิดพลาดขึ้นด้วยสาเหตุจากพาริตี้ , โอเวอร์รัน , เฟรมข้อมูล หรือการเบรกข้อมูล

"1" เอ็นเอเบิลการอินเตอร์รัปต์

"0" ดิสเอเบิล

TBE เอ็นเอเบิลการอินเตอร์รัปต์เมื่อรีจิสเตอร์บัฟเฟอร์สำหรับส่งข้อมูลว่าง

"1" เอ็นเอเบิลการอินเตอร์รัปต์

"0" ดิสเอเบิล

RxRD เอ็นเอเบิลการอินเตอร์รัปต์เนื่องจากรีจิสเตอร์บัฟเฟอร์ได้รับข้อมูลแล้ว

"1" เอ็นเอเบิลการอินเตอร์รัปต์

"0" ดิสเอเบิล

3.4.3 รีจิสเตอร์ตำแหน่ง 02H (รีจิสเตอร์แสดงโหมดและสถานะการอินเตอร์รัปต์)

มีรายละเอียดของแต่ละบิตดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
0	0	0	0	0	ID1	ID0	PND

ตารางที่ 3.4 ฟังก์ชั่นการทำงานของรีจีสเตอร์ 02H

บิต 3-7 ไม่ได้ใช้งาน อ่านค่าได้เท่ากับ "0"

ID1,ID0 ใช้งานร่วมกันเพื่อแจ้งสาเหตุของการเกิดอินเตอร์รัปต์

"00" เกิดการอินเตอร์รัปต์เนื่องจากการเปลี่ยนแปลงของขาอินพุตขึ้น การอินเตอร์รัปต์แบบนี้มีนัยสำคัญเป็นอันดับ 4

"01" เกิดการอินเตอร์รัปต์เนื่องจากรีจิสเตอร์บัฟเฟอร์ส่งข้อมูลว่างขึ้น การอินเตอร์รัปต์แบบนี้มีนัยสำคัญเป็นอันดับ 3

"10" เกิดการอินเตอร์รัปต์เนื่องจากข้อมูลถูกเก็บลงในรีจิสเตอร์บัฟเฟอร์สำหรับรับข้อมูลเรียบร้อยแล้ว การอินเตอร์รัปต์แบบนี้มีนัยสำคัญเป็นอันดับ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

"11" เกิดการอินเตอร์รัปต์เนื่องจากความผิดพลาดในการถ่ายทอดข้อมูล หรือเกิดการเบรก (Break :เกิดการหยุดถ่ายทอดข้อมูลกระทันหัน) การอินเตอร์รัปต์แบบนี้มีนัยสำคัญเป็นอันดับ 1 หรือมีนัยสำคัญสูงสุด

PND ใช้แสดงสถานะของการเกิดอินเตอร์รัปต์

"1" แสดงว่าไม่มีการอินเตอร์รัปต์

"0" แสดงว่ามีการอินเตอร์รัปต์เกิดขึ้น

เมื่อมีการสร้างสัญญาณอินเตอร์รัปต์ขึ้น จะต้องมีการเคลียร์ค่าก่อนที่จะให้เกิดอินเตอร์รัปต์ครั้งต่อไป โดยสามารถทำได้ดังนี้ คือ

- ถ้าเกิดอินเตอร์รัปต์เนื่องจากการเปลี่ยนแปลงของขาอินพุตจะต้องอ่านค่าจากรีจิสเตอร์แสดงสถานะของโมเด็ม (รีจิสเตอร์ตำแหน่ง 06H) เพื่อเคลียร์ค่าการอินเตอร์รัปต์
- ถ้าเกิดการอินเตอร์รัปต์เนื่องจากบัฟเฟอร์ส่งข้อมูลว่าง จะต้องเขียนข้อมูลไปยังรีจิสเตอร์บัฟเฟอร์ข้อมูล (รีจิสเตอร์ตำแหน่ง 00H) หรืออ่านค่ารีจิสเตอร์แสดงสถานะอินเตอร์รัปต์ (รีจิสเตอร์ตำแหน่ง 02H)เพื่อเคลียร์ค่าการอินเตอร์รัปต์
- ถ้าเกิดอินเตอร์รัปต์เนื่องจากการเก็บข้อมูลลงในรีจิสเตอร์บัฟเฟอร์สำหรับข้อมูลเรียบร้อย จะต้องเคลียร์ค่าอินเตอร์รัปต์โดยการอ่านข้อมูลจากรีจิสเตอร์บัฟเฟอร์
- ถ้าเกิดอินเตอร์รัปต์เนื่องจากความผิดพลาดในการรับส่งข้อมูลหรือเกิดการเบรก จะต้องเคลียร์ค่าอินเตอร์รัปต์โดยการอ่านค่ารีจิสเตอร์แสดงสถานการณ์รับและส่งข้อมูลแบบอนุกรม

3.4.4 รีจิสเตอร์ตำแหน่ง 03H (รีจิสเตอร์กำหนดรูปแบบของข้อมูล)

มีรายละเอียดหน้าที่ของแต่ละบิตดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิด 0
DLAB	BRK	PAR2	PAR1	PAR0	STOP	DAB1	DAB0

ตารางที่ 3.5 ฟังก์ชั่นการทำงานของรีจีสเตอร์ 03H

DLAB ใช้ในการกำหนดหน้าที่การทำงานของรีจิสเตอร์บัฟเฟอร์ (00H)

"1" เป็นการเข้าสู่โหมดการหารค่าบอดเรต

"0" เป็นการเข้าถึงรีจิสเตอร์บัฟเฟอร์(รีจิสเตอร์ตำแหน่ง 00H) และ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รีจิสเตอร์สำหรับเอ็นเอเบิลการอินเตอร์รัปต์ (รีจิสเตอร์ตำแหน่ง01H) เมื่อบิต DLAB เป็น “1” รีจิสเตอร์บัฟเฟอร์ (00H) และรีจิสเตอร์เอ็นเอเบิลการอินเตอร์รัปต์ (01H) จะใช้สำหรับโหลดค่าการหารความถี่สำหรับกำหนดค่าบอดเรต โดยรีจิสเตอร์ 00H เก็บค่าตัวหารไบต์ต่ำ ส่วนรีจิสเตอร์ 01H ใช้เก็บค่าตัวหารไบต์สูง การหาค่าบอดเรตสามารถเขียนเป็นสมการได้ดังนี้

บอดเรต = 115200/ ค่าตัวหาร16 บิต

ค่าตัวเลข 115200 มาจากความถี่คริสตอลในวงจร UART ภายในเครื่องคอมพิวเตอร์ โดยคริสตอลที่ใช้มีความถี่ 1.8432 MHz วงจรภายใน UART จะหารค่าความถี่นี้ด้วย 16 ทำให้ได้ค่าความถี่ 115200 Hz ออกมา

ค่าตัวหาร16บิต = ข้อมูลในรีจิสเตอร์00H + (256 * ข้อมูลในรีจิสเตอร์01H)

ถ้าต้องการบอดเรตเท่ากับ 9600 ค่าตัวหารที่ใช้จะต้องมีค่าเท่ากับ 12 ซึ่งค่านี้จะถูกเขียนลงในรีจิสเตอร์ 00H และเขียนค่า 0 ลงไปในรีจิสเตอร์ 01H ค่าตัวหารที่ทำให้เกิดค่าบอดเรตสูงสุดที่ 115200 บิตต่อวินาที คือ ค่า 0001 นั่นคือ รีจิสเตอร์ 00H มีค่าเท่ากับ 1 และรีจิสเตอร์ 01H มีค่าเท่ากับ 0

BRK ใช้ควบคุมการหยุดถ่ายทอดข้อมูล

“1” สามารถหยุดหรือเบรกได้

“0” ไม่มีการหยุดหรือเบรกได้

PAR2,PAR1,PAR0 ใช้เพื่อกำหนดบิตพาริตี้ได้

“000” ไม่ใช้บิตพาริตี้

“001” กำหนดพาริตี้คี่

“011” กำหนดพาริตี้คู่

“101” มาร์ก (Mark)

“111” ช่องว่าง(Space)

STOP ใช้กำหนดจำนวนบิตปิดท้าย

“1” มีบิตปิดท้าย 2 บิต

“0” มีบิตปิดท้าย 1 บิต

DAB1,DAB0 ใช้ร่วมกันในการกำหนดจำนวนบิตของข้อมูลที่ต้องการถ่ายทอด

“00” จำนวนบิตข้อมูลเท่ากับ 5 บิต

“01” จำนวนบิตข้อมูลเท่ากับ 6 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

"10" จำนวนบิตข้อมูลเท่ากับ 7 บิต

"11" จำนวนบิตข้อมูลเท่ากับ 8 บิต

3.4.5 รีจิสเตอร์ตำแหน่ง 04H (รีจิสเตอร์ควบคุมโมเด็ม)

มีรายละเอียดหน้าที่ของแต่ละบิตดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
0	0	0	LOOP	OUT2	OUT1	RTS	DTR

ตารางที่ 3.6 ฟังก์ชั่นการทำงานของรีจิสเตอร์ 04H

บิต 5-7 ไม่มีการใช้งาน อ่านค่าได้เท่ากับ 0

LOOP "1" เอ็นเอเบิลการส่งค่ากลับ

"0" ดิสเอเบิล

OUT1,OUT2 "1" เอ็นเอเบิลการใช้งานภายใน

"0" ดิสเอเบิล

RTS ใช้ควบคุมการทำงานของขาRTS(Ready TO Send)

"1" เอ็นเอเบิล

"0" ดิสเอเบิล

DTR ใช้ควบคุมการทำงานของขาDTR (Data Terminal Ready)

"1" เอ็นเอเบิล

"0" ดิสเอเบิล

3.4.6 รีจิสเตอร์ตำแหน่ง 05H (รีจิสเตอร์แสดงสถานะรอรับและส่งข้อมูลอนุกรมของUART)

ใช้งานร่วมกับรีจิสเตอร์แสดงโหมดและสถานะของการอินเตอร์รัปต์ (รีจิสเตอร์ตำแหน่ง02H) เพื่อแสดงสาเหตุของการเกิดอินเตอร์รัปต์ มีรายละเอียดหน้าที่ของแต่ละบิตดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
0	TXE	TBE	BREK	FRME	PARE	OVFE	RxRD

ตารางที่ 3.7 ฟังก์ชั่นการทำงานของรีจิสเตอร์ 05H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TXE(Transmitter Empty)

"1" แสดงว่ารีจิสเตอร์บัฟเฟอร์สำหรับส่งข้อมูลว่าง

"0" แสดงว่ายังคงมีข้อมูล 1 ไบต์เก็บอยู่ในรีจิสเตอร์บัฟเฟอร์สำหรับส่งข้อมูล

TBE(Transmitter Buffer Empty)

"1" รีจิสเตอร์บัฟเฟอร์สำหรับส่งข้อมูลว่าง

"0" ยังคงมีข้อมูล 1 ไบต์เก็บอยู่ในรีจิสเตอร์บัฟเฟอร์สำหรับส่งข้อมูล

BREK(Break)

"1" UART ตรวจพบการเบรก

"0" ไม่มีการเบรก

FRME (Frame Error)

"1" UART ตรวจพบความผิดพลาดด้านเฟรมข้อมูล

"0" ไม่พบความผิดพลาดทางพาริตี้

OVRE(Overrun Error)

"1" UART ตรวจพบความผิดพลาดแบบโอเวอร์รัน

"0" ไม่พบความผิดพลาดแบบโอเวอร์รัน

RxRD(Received Data Ready)

"1" มีการรับข้อมูลเข้ามาเก็บไว้ในบัฟเฟอร์

"0" ไม่มีข้อมูล

3.4.7 รีจิสเตอร์ตำแหน่ง 06H (รีจิสเตอร์แสดงสถานะของโมเด็ม)

ใช้เพื่อกำหนดสถานะสัญญาณอินพุต ของพอร์ตอนุกรม RS-232 ซึ่งได้แก่ สัญญาณ DCD,DSR,CTS และ RI สำหรับการเชื่อมต่อใช้งานแบบอเนกประสงค์ ดังมีรายละเอียดหน้าที่ของแต่ละบิตต่อไปนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
DCD	RI	DSR	CTS	DDCD	DRI	DDSR	DCTS

ตารางที่ 3.7 ฟังก์ชั่นการทำงานของรีจิสเตอร์ 06H

DCD ใช้แสดงสภาวะของขา DCD

"1" แสดงว่าที่ขา DCD เป็นลอจิก "1"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

"0" แสดงว่าที่ขา DCD เป็นลอจิก "0"

RI ใช้แสดงสภาวะของขา RI

"1" แสดงว่าที่ขา RI เป็นลอจิก "1"

"0" แสดงว่าที่ขา RI เป็นลอจิก "0"

DSR ใช้แสดงสภาวะของขา DSR

"1" แสดงว่าที่ขา DSR เป็นลอจิก "1"

"0" แสดงว่าที่ขา DSR เป็นลอจิก "0"

DCTS(Delta Clear To Send) ใช้แจ้งการเปลี่ยนแปลงที่เกิดขึ้นของบิต CTS

"1" แสดงว่าบิตCTS เกิดการเปลี่ยนแปลงเมื่อเทียบจากการอ่านค่าครั้งที่แล้ว

"0" แสดงว่าบิตCTS ไม่เกิดการเปลี่ยนแปลงเมื่อเทียบจากการอ่านค่าครั้งที่แล้ว

DRI (Delta Ring Indicator)ใช้แจ้งการเปลี่ยนแปลงที่เกิดขึ้นของบิต RI

"1" แสดงว่าบิตRI เกิดการเปลี่ยนแปลงเมื่อเทียบจากการอ่านค่าครั้งที่แล้ว

"0" แสดงว่าบิตRIไม่ เกิดการเปลี่ยนแปลงเมื่อเทียบจากการอ่านค่าครั้งที่แล้ว

DDCD(Delta Data Carrier Detect)ใช้แจ้งการเปลี่ยนแปลงที่เกิดขึ้นในบิตDDCD

"1" แสดงว่าบิตDDCD เกิดการเปลี่ยนแปลงเมื่อเทียบจากการอ่านค่าครั้งที่แล้ว

"0" แสดงว่าไม่ เกิดการเปลี่ยนแปลงเมื่อเทียบจากการอ่านค่าครั้งที่แล้ว

DCTS(Delta Clear to send) ใช้แสดงสภาวะของ CTS

"1" แสดงว่าที่ขา CTS มีลอจิกเป็น "1"

"0" แสดงว่าที่ขา CTS มีลอจิกเป็น "0"

3.4.8 รีจิสเตอร์ตำแหน่ง07H (รีจิสเตอร์สำหรับเก็บข้อมูลชั่วคราว)

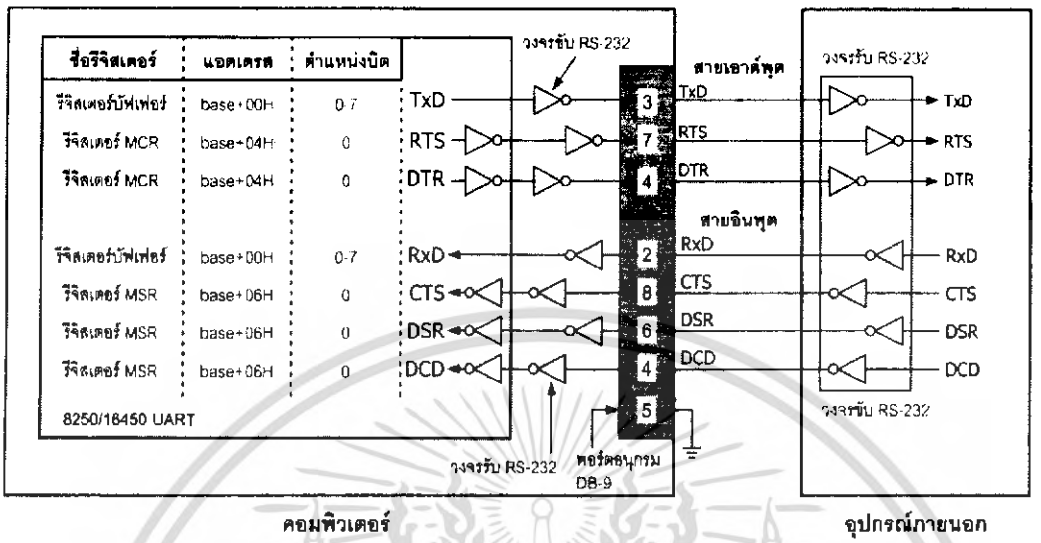
ทำหน้าที่เป็นหน่วยความจำแรมขนาด 1 ไบต์ การอ่านและเขียนข้อมูลที่รีจิสเตอร์ตัวนี้ไม่ส่งผลใดๆ ต่อการใช้งานUART

3.5 ลักษณะสัญญาณอินพุตและเอาต์พุตของพอร์ตRS-232

สัญญาณเอาต์พุตที่ใช้ควบคุม (RTS และ DTS) และสัญญาณแสดงสถานะอินพุต(CTS,CSR และ DCD) ของพอร์ตอนุกรม RS-232 จะถูกกลับสถานะภายในตัว UART ส่วนสัญญาณข้อมูลทั้งภาคส่งและรับจะไม่ถูกกลับสถานะ UART จะให้ระดับสัญญาณเอาต์พุตออกมาเป็นแบบทีทีแอลเท่านั้น ดังนั้นเมื่อสัญญาณถูกส่งออกมาจากUART จึงต้องส่งเข้าสู่วงจรขับเพื่อปรับแรงดันให้ได้ระดับแรงดันเป็นไปตามมาตรฐานRS-232 ก่อนส่งออกไปจากคอมพิวเตอร์สำหรับอุปกรณ์ต่อเชื่อมปลายทางก็จะต้องมีวงจรขับในลักษณะนี้เช่นเดียวกัน เพื่อให้ได้ระดับสัญญาณในระดับเดียวกัน แต่วงจรขับที่ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทั้งภายในคอมพิวเตอร์และอุปกรณ์ต่อเชื่อมปลายทางนั้นจะถูกกลับสถานะดังแสดงเป็นบล็อกไดอะแกรมในรูปที่ 3.6



รูปที่ 3.6 ไดอะแกรมแสดงโครงสร้างทางฮาร์ดแวร์ของพอร์ตอนุกรม

3.6 แอดเดรสของพอร์ตอนุกรม

แอดเดรสพื้นฐานของพอร์ตอนุกรมมี 4 ตำแหน่งดังนี้คือ

COM1 : 3F8H

COM2 : 2F8H

COM3 : 3E8H

COM4 : 2E8H

บิต 3	บิต 2	บิต1	จำนวนพอร์ต
0	0	0	ไม่มีพอร์ตอนุกรม
0	0	1	มีพอร์ตอนุกรม 1 พอร์ต
0	1	0	มีพอร์ตอนุกรม 1 พอร์ต
0	1	1	มีพอร์ตอนุกรม 2 พอร์ต
1	0	0	มีพอร์ตอนุกรม

ตาราง 3.8 ข้อมูลในแอดเดส 0000:0411H ที่ใช้แจ้งจำนวนพอร์ตอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเริ่มเปิดเครื่องเพื่อใช้งานคอมพิวเตอร์ ไบออสภายในคอมพิวเตอร์จะทำการตรวจสอบแอดเดรสของพอร์ตอนุกรมทั้งหมด ถ้าไบออสตรวจพบแอดเดรสของพอร์ตอนุกรมไบออสจะนำแอดเดรสที่ตรวจพบไปเก็บไว้ในหน่วยความจำขนาด 2ไบต์ สำหรับพอร์ตอนุกรม COM1จะเก็บไว้ที่แอดเดรส 0000:0400H และ 0000:0401H ส่วนตำแหน่งอื่นๆมีรายละเอียดดังนี้

COM2 = 0000:0402H – 0000:0403H

COM3 = 0000:0404H – 0000:0405H

COM4 = 0000:0406H – 0000:0407H

นอกจากนี้ที่หน่วยความจำแอดเดรส 0000:0411H ยังใช้สำหรับแสดงจำนวนของพอร์ตอนุกรมที่มีใช้อยู่ในเครื่องคอมพิวเตอร์อีกด้วย โดยรายละเอียดดังแสดงในตารางที่ 3.8

บทที่ 4 มาตรฐานการสื่อสารแบบ RS 485

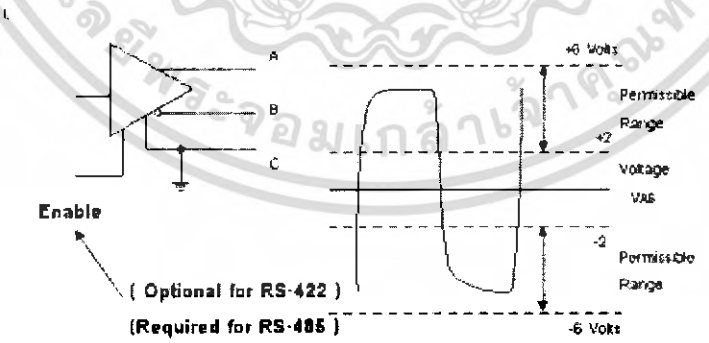
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1 ตัวส่งแบบไม่สมดุลย์ (Unbalanced line driver)

ระดับสัญญาณที่ส่งในระบบไม่สมดุลย์ RS-232 เป็นระบบที่วัดระดับแรงดันสัญญาณเทียบกับสายกราวด์ ยกตัวอย่างเช่น ในการส่งข้อมูล (Transmission Data : TD) จากอุปกรณ์ (Data Terminal Equipment : DTE) ผ่านหัวต่อแบบ DB-25 จะส่งสัญญาณข้อมูลทางขา 2 โดยการวัดระดับแรงดันของสัญญาณจะวัดเทียบกับสายกราวด์ (Signal ground) ที่ขา 7 ถ้าสายส่งข้อมูลอยู่ในสถานะ idle ระดับแรงดันจะมีค่าเป็นลบ และเมื่อทำการส่งข้อมูลระดับแรงดันจะเปลี่ยนแปลงในช่วงค่าบวกและค่าลบ โดยมีระดับแรงดันอยู่ในช่วง ±5 V ถึง ±15 V ดังที่ได้กล่าวแล้วในบทที่ผ่านมา

4.2 ตัวส่งแบบสมดุลย์ (Balance line driver)

ระบบสมดุลย์จะส่งสัญญาณผ่านสาย 2 เส้น รูปที่4.1 แสดงรูปแบบและระดับแรงดัน ตัวส่งจะส่งแรงดันช่วง 2-6 V ที่เอาท์พุทระหว่าง A และ B ระบบจะมีการเชื่อมต่อสายกราวด์ 1 เส้น สายกราวด์ในระบบนี้ไม่ได้ใช้ในการส่งสัญญาณหรือหาสถานะลอจิกของข้อมูล แต่สายกราวด์มีความสำคัญคือใช้เป็นจุดอ้างอิงของระบบ การวัดสัญญาณข้อมูลจะถูกวัดเทียบกับสายกราวด์ ในการส่งข้อมูลตัวส่งจะได้รับสัญญาณหนึ่งเรียกว่าสัญญาณควบคุม หรือ Enable signal ซึ่งเชื่อมต่อระหว่างตัวส่งและเทอร์มินอล A และ B ที่เอาท์พุท ตัวส่งจะส่งสัญญาณควบคุมซึ่งเปรียบเสมือนเป็นเอาท์พุทที่ 3 นอกจากการส่งสถานะลอจิก 1 หรือ 0 ให้กับเทอร์มินอล ถ้าสัญญาณควบคุมอยู่ในสถานะ OFF จะหมายถึงตัวส่งตัวนั้นไม่ได้ต่ออยู่กับสายสัญญาณและไม่สามารถส่งสัญญาณข้อมูลได้ หรืออยู่ในสถานะ disable หรือ tri-state

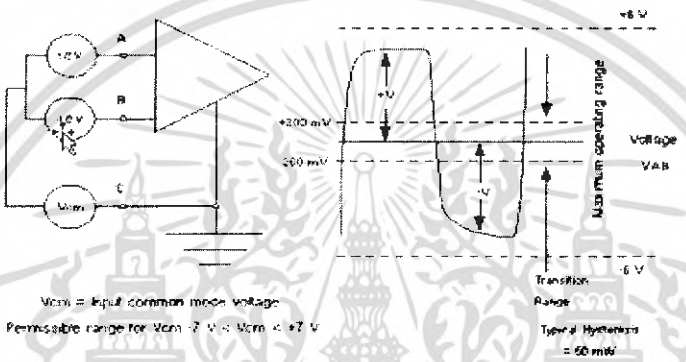


รูปที่ 4.1 Balance Differential Output Line Driver

4.3 ตัวรับแบบสมดุลย์ (Balanced line receiver)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวรับในระบบสมดุลจะถูกเชื่อมต่อด้วยสายส่งสัญญาณข้อมูลและสายกราวด์ โดยจะรับสถานะของสัญญาณได้โดยวัดความแตกต่างของระดับแรงดันที่สายอินพุทของตัวรับ หรือ วัดระหว่างเทอร์มินอล A และ B (Vab) ดังแสดงในรูปที่4.2 ถ้าแรงดันมีขนาดมากกว่า +200 mV จะถูกกำหนดให้มีสถานะเป็นลอจิกค่าหนึ่ง และถ้ามีขนาดน้อยกว่า -200 mV จะถูกกำหนดให้มีสถานะตรงข้าม เนื่องจากอาจเกิดการลดทอนสัญญาณ (Attenuate)ขึ้นในสายส่งที่ตัวรับจึงถูกออกแบบให้สามารถรับความแตกต่างระดับแรงดันของสัญญาณได้ในช่วงที่กว้างกว่าตัวส่งคือ ±200 mV ถึง
±6 V ในขณะที่แรงดันที่ตัวส่งอยู่ในช่วง ±2 V ถึง ±6 V

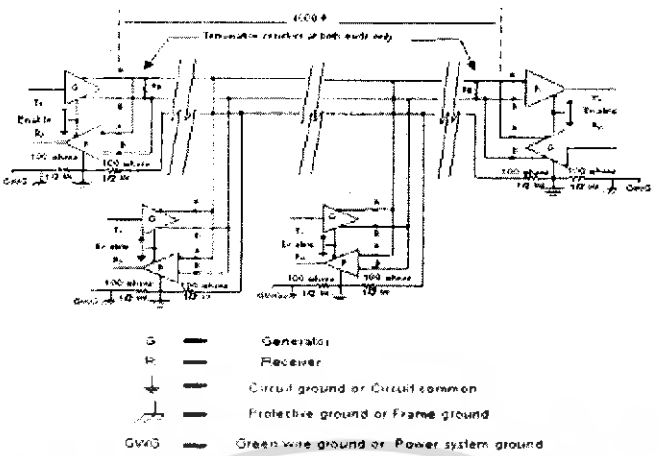


รูปที่ 4.2 Balanced Differential Input Line Receiver

4.4 การส่งข้อมูลตามมาตรฐาน RS-485

RS-485 เป็นการส่งข้อมูลในระบบสมดุลย์ สายสัญญาณ 1 คู่สายสามารถติดต่ออุปกรณ์ได้ถึง 32ตัว สำหรับในระบบ RS-485 สามารถทนแรงดันระหว่างสายสัญญาณและสายกราวด์ หรือ Common Mode Voltage หรือ Vcm ได้ในช่วง -7 V ถึง +12 V รูปที่4.3 แสดงระบบเครือข่ายที่เรียกการต่อในลักษณะนี้ว่า two-wire multidrop จากรูปสังเกตุได้ว่ามีการต่อความต้านทานขั้วที่โหนดปลายทั้งสองด้านของสายส่ง แต่ไม่มีการต่อขั้วปลายที่โหนดที่อยู่ระหว่างสายส่ง การต่อขั้วปลาย (termination) มักใช้กับระบบที่มีอัตราการส่งข้อมูลสูงและระยะทางยาว รายละเอียดของการต่อขั้วปลายจะได้กล่าวถึงต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 ระบบเครือข่าย two-wire multidrop

4.5 การควบคุม tri-state ของอุปกรณ์ RS-485

ในระบบ RS-485 เป็นการส่งผ่านข้อมูลผ่านสายส่งชุดเดียวกันโดยผลัดกันส่ง เมื่ออุปกรณ์ตัวใดตัวหนึ่งต้องการส่งข้อมูลจะทำการเชื่อมต่อตัวส่งเข้ากับสายส่ง และจะตัดตัวส่งออกจากสายส่งเมื่อส่งข้อมูลเสร็จ โดยส่วนมากมักใช้อุปกรณ์ที่เปลี่ยน RS-232 เป็น RS-485 หรือการ์ดอนุกรม RS-485 ต่อเข้ากับระบบ เพื่อใช้เป็นตัวให้สัญญาณควบคุมการส่ง หรือเรียกว่า RTS : Request To Send โดยต่อจาก Asynchronous serial port ไปยังขา Enable ของตัวส่งผ่านสาย RTS และอาจกำหนดการทำงานของขา Enable โดยถ้าได้รับสถานะ High หรือลอจิก 1 ให้ตัวส่งต่อเข้ากับสายส่งและทำการส่งข้อมูลได้ ถ้าได้รับสถานะ Low หรือลอจิก 0 ให้ตัวส่งตัดการต่อออกจากสายส่ง หรือเรียกว่า tri-state และยอมให้ตัวส่งอื่นที่ได้รับลอจิก 1 ต่อเข้ากับสายส่งและส่งข้อมูลผ่านสายส่งได้ ดังนั้นในขณะที่มีตัวส่งตัวเดียวได้รับลอจิก 1 และต่อเข้ากับสายส่ง ตัวส่งตัวอื่นๆที่เหลือจะต้องได้รับลอจิก 0

4.6 การควบคุมการส่งข้อมูลของอุปกรณ์ RS-485

ผลิตภัณฑ์ตัวเปลี่ยน RS-232 เป็น RS-485 (RS-232 to RS-485 Converter) และการ์ดอนุกรม RS-485 (RS-485 serial cards) มีวงจรพิเศษในการใช้สัญญาณข้อมูลเป็นตัว enable ให้กับตัวส่ง (RS-485 driver) ในการส่งข้อมูลนั้นช่วงเวลาคงที่เวลาหนึ่งหลังการส่งข้อมูลบิทสุดท้ายและก่อนการ disable ของตัวส่ง ถ้าระยะเวลานั้นสั้นเกินไปหรือตัวส่งอาจถูก disableก่อนที่ตัวส่งจะส่งข้อมูลเสร็จ อาจทำให้ข้อมูลในส่วนท้ายสูญหายได้ และถ้ามีขนาดยาวเกินไปจะทำให้ระบบเปลี่ยนสายสัญญาณจากส่งเป็นรับก่อนที่โหนดพร้อมจะรับข้อมูลหรือตัวรับเชื่อมต่อเข้ากับสายส่งและรับข้อมูลช้าไม่ทันต่อสัญญาณที่ส่งมาถึง ทำให้ไม่ได้รับข้อมูลในส่วนแรก ดังนั้นควรกำหนดช่วงเวลาดังกล่าวให้เหมาะสม โดยส่วนมากมักเท่ากับความยาวหนึ่งตัวอักษรที่อัตราการส่งข้อมูลนั้นๆ

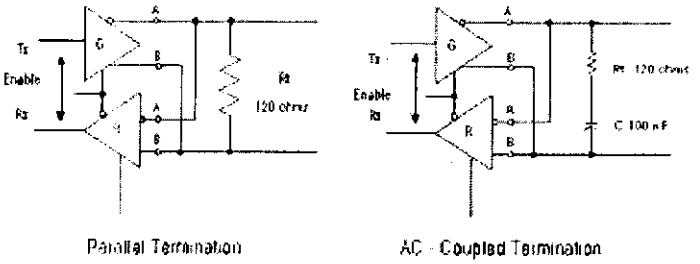
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.7 การต่อขั้วปลาย (Termination)

ระบบควรมีการต่อขั้วปลายหรือการต่ออิมพีแดนซ์(Rt)เข้าระหว่างเทอร์มินอล เพื่อทำให้เกิดอิมพีแดนซ์สมดุลย์กันระหว่างอิมพีแดนซ์ของโหลดกับอิมพีแดนซ์ของสายส่ง ถ้าอิมพีแดนซ์ไม่สมดุลย์จะทำให้โหลดไม่ได้รับสัญญาณที่สมบูรณ์เนื่องจากสัญญาณบางส่วนเกิดการสะท้อนกลับภายในสายส่ง ถ้าอิมพีแดนซ์ของตัวกำเนิด (source) , อิมพีแดนซ์ของสายส่ง (transmission line) และอิมพีแดนซ์ของโหลดมีค่าเท่ากันจะไม่เกิดการสะท้อนกลับในระบบ แต่การต่อขั้วปลายนั้นก็ถือว่าเป็นการเพิ่มโหลดให้กับตัวส่ง ทำให้การติดตั้งซับซ้อนมากขึ้น, จำเป็นต้องเปลี่ยนแปลงการไบอัส และการปรับปรุงหรือแก้ไขระบบจะทำได้ยากขึ้น การตัดสินใจว่าควรทำการต่อขั้วปลายหรือไม่ขึ้นอยู่กับความยาวของสายเคเบิ้ลและอัตราข้อมูลในระบบ หรืออีกทางหนึ่งอาจดูจากPropagation delay ถ้ามีค่าน้อยกว่าความยาว 1 บิท ไม่จำเป็นต้องมีการต่อขั้วปลายเนื่องจากสัญญาณจะสะท้อนกลับไปกลับมาจนมีขนาดน้อยลงและหายไปในที่สุด (damp out) จึงไม่มีผลต่อการรับข้อมูล การรับของ "UART" จะเอาค่าที่อยู่ตรงกลางของบิทซึ่งถือว่าเป็นค่าที่มีความถูกต้องที่สุด การหาค่า Propagation delay คำนวณได้จากผลคูณระหว่างความยาวสายเคเบิ้ลกับ Propagation velocity ของสายเคเบิ้ล โดยจะถูกกำหนดจากผู้ผลิตซึ่งมีค่าอยู่ในช่วง 66%-75% ของความเร็วแสง

การต่อขั้วปลายมีหลายวิธีแต่ เช่น การต่อแบบ Parallel termination ความต้านทานจะต่อขนานเข้ากับขั้ว A และ B ของตัวรับ ถ้าเป็นระบบที่ไม่มีการต่อตัวทวนสัญญาณ หรือ Repeater ควรต่อขั้วปลายเพียง 2 แห่งคือที่ปลายทั้งสองด้านของสายส่ง สำหรับการเลือกค่าความต้านทานขั้ว หรือ Rt นั้น ขึ้นอยู่กับค่าอิมพีแดนซ์ภายในสายส่งซึ่งขึ้นอยู่กับการผลิตของผู้ผลิตไม่ได้ขึ้นอยู่กับความยาวของสาย ซึ่งโดยมากแล้วผู้ผลิตมักกำหนดการผลิตสายเคเบิ้ลให้มีค่าความต้านทานอยู่ที่ 120โอมห์ อย่างไรก็ตามสามารถเลือกค่าความต้านทานได้ตั้งแต่ 90โอมห์ขึ้นไป ข้อเสียของการต่อขั้วปลายแบบขนานคือ เปรียบเสมือนการเพิ่มโหลด DC ให้กับระบบและอาจทำให้ตัวเปลี่ยน RS-232 เป็น RS-485 (RS-232 to RS-485 Converter) เกิดการ overload นอกจากนี้ยังมีการต่อขั้วปลายอีกวิธีหนึ่งเรียกว่า AC Coupled Termination ทำได้โดยการต่อตัวเก็บประจุขนาดเล็กๆอนุกรมเข้ากับความต้านทานขั้ว (Rt) เพื่อกำจัดผลกระทบจาก DC loading การเลือกค่าตัวเก็บประจุขึ้นอยู่กับคุณสมบัติของระบบ รูปที่4.4 แสดงการต่อขั้วปลายแบบ Parallel termination และ AC Coupled Terminationในระบบ RS-485 แบบ 2 สาย สำหรับในระบบแบบ 4 สายความต้านทานขั้ว หรือ Rt จะถูกต่อคร่อมที่ตัวรับ

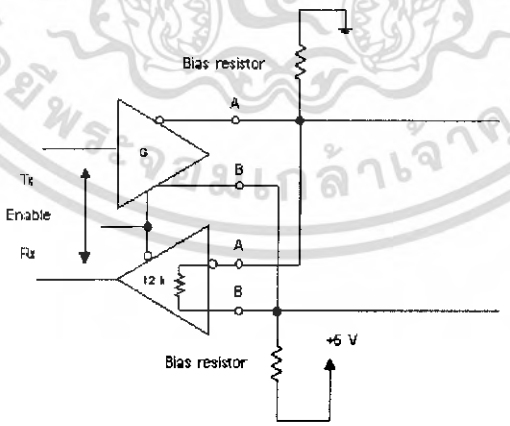
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 การต่อขั้วปลายแบบ Parallel termination และ AC Coupled Termination

4.8 การไบอัสในระบบเครือข่าย RS-485 (Biasing an RS-485 Network)

เมื่อทุกโหนดอยู่ในโหมดของการรอรับข้อมูล หรืออยู่ในสถานะ idle (idle state) จะไม่มีตัวส่งใดทำงาน (active) หรือเรียกว่าอยู่ในสถานะ tri-state สายส่งจะอยู่ในลักษณะเรียกว่า Unknown ถ้าระดับแรงดันของอินพุทที่ ตัวรับ (A และ B)มีค่าอยู่ในช่วง +- 200 mV. ระดับลอจิกที่เอาท์พุทของตัวรับจะคงค่าของบิทสุดท้ายที่รับเพื่อรักษาแรงดัน idle ให้เหมาะสมจำเป็นต้องมีการต่อความต้านทานที่เรียกว่าความต้านทานไบอัส เข้าระหว่างขั้ว B กับ 5V. หรือเรียกว่า Pull up resister และใส่ความต้านระหว่างขั้ว A กับกราวด์หรือเรียกว่า Pull down resister รูปที่ 4.5 แสดงการต่อความต้านทานไบอัสเข้ากับตัวรับในระบบแบบ 2 สาย สำหรับระบบแบบ4 สายจะต่อความต้านทานเข้ากับตัวรับ ค่าความต้านทานไบอัสขึ้นอยู่กับการต่อขั้วปลายและจำนวนโหนดในระบบ การสร้างกระแสไบอัสให้มีค่าเพียงพอที่จะรักษาแรงดันระหว่างสายส่งAและBให้มีค่ามากกว่า 200 mV. ดูได้จากตัวอย่างที่ 2 และ 3 ซึ่งแสดงการคำนวณค่าความต้านทานไบอัส



รูปที่ 4.5 การต่อความต้านทานไบอัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5 ไมโครคอนโทรลเลอร์ MCS-51

5.1 ไมโครคอนโทรลเลอร์ MCS-51

ไมโครคอนโทรลเลอร์ MCS-51 เป็นไมโครคอนโทรลเลอร์ขนาด 8 บิตที่มีอุปกรณ์สนับสนุนประกอบอยู่ภายในหลายอย่างได้แก่ หน่วยความจำสำหรับเก็บข้อมูล หน่วยความจำสำหรับเก็บโปรแกรม ตัวตั้งเวลา/ตัวนับ อุปกรณ์รับส่งข้อมูลแบบอนุกรม เนื่องจากโครงสร้างของไมโครคอนโทรลเลอร์มีอุปกรณ์สนับสนุนประกอบอยู่ภายในนี้เอง ทำให้การใช้งานง่ายขึ้นและมีประสิทธิภาพมากขึ้นโดยไม่ต้องมีการเชื่อมต่ออุปกรณ์ภายนอกเพิ่มเติมมากเหมือนกับ ตัวไมโครโปรเซสเซอร์ทั่วไป นอกจากนี้หากเราต้องการใช้งานไมโครคอนโทรลเลอร์ร่วมกับ อุปกรณ์อื่นเพิ่มเติมเช่น ไอซี 8255 หรือหน่วยความจำภายนอก เรายังสามารถนำมาเชื่อมต่อเพิ่มเติมเข้ากับไมโครคอนโทรลเลอร์ได้อีกด้วย

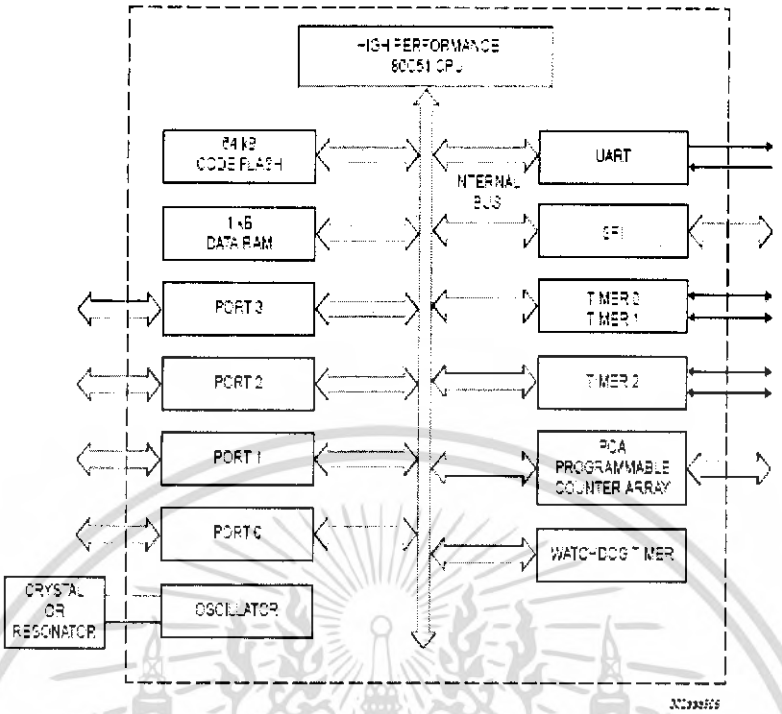
5.2 โครงสร้างภายในของไมโครคอนโทรลเลอร์ MCS-51

โครงสร้างภายในพื้นฐานของไมโครคอนโทรลเลอร์ MCS-51 แสดงในรูปที่ 5.1 ประกอบด้วยอุปกรณ์ต่างๆ ดังนี้

- หน่วยประมวลผลกลางขนาด 8 บิต
- หน่วยประมวลผลสำหรับข้อมูลแบบบิต (BOOLEAN PROCESSOR)
- หน่วยความจำโปรแกรมภายในขนาด 64 กิโลไบต์ แบบ แฟลช
- หน่วยความจำแบบ แรม ภายในจำนวน 1 กิโลไบต์
- พอร์ตอินพุต/เอาต์พุตแบบขนานจำนวน 32 เส้น ซึ่งสามารถแยกทำงานได้อย่างอิสระ
- วงจรนับ/จับเวลาขนาด 16 บิต จำนวน 3วงจร
- วงจรสื่อสารแบบอนุกรมแบบดูเพล็กเต็ม(FULL DUPLEX)
- วงจรควบคุมการอินเตอร์รัปต์จากแหล่งกำเนิดสัญญาณ 6 ประเภท พร้อมการกำหนด ลำดับ

วงจรผลิตสัญญาณนาฬิกาภายในซึ่งโครงสร้างการทำงานทั้งหมดของไมโครคอนโทรลเลอร์จะอาศัยหลักการทำงานที่เกี่ยวข้องกัน โดยอาศัยหลักการทำงานที่เป็นไป ตามโครงสร้างเสมอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



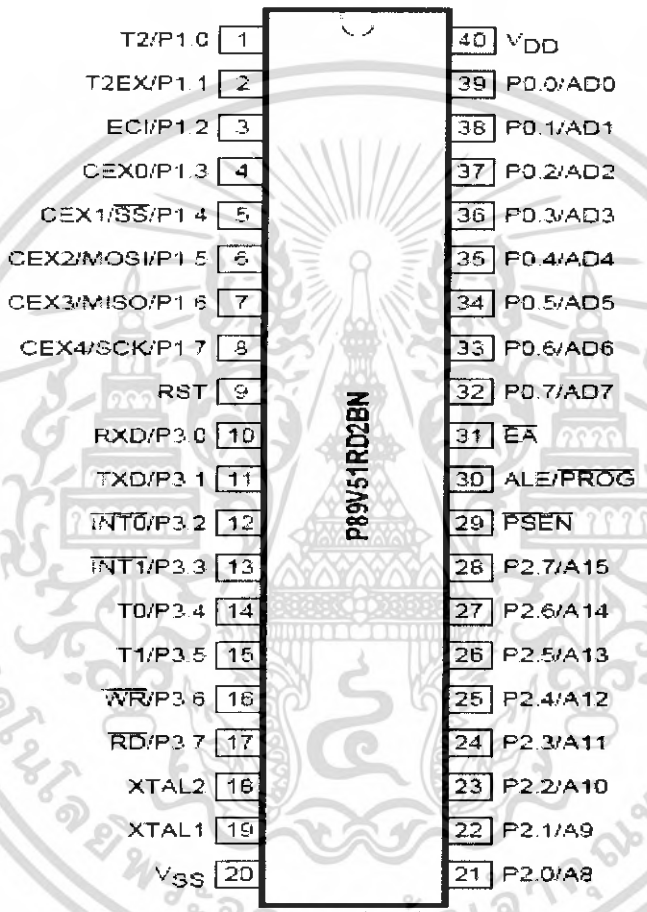
รูปที่ 5.1 แสดงโครงสร้างพื้นฐานของไมโครคอนโทรลเลอร์ MCS-51

โดยมากแล้วไมโครคอนโทรลเลอร์ตระกูลนี้มักจะมีรูปร่างของไอซีเป็นแบบขนาด 40 ขา ดังแสดงในรูปที่ 5.2 ซึ่งแต่ละขาสัญญาณจะมีหน้าที่ที่ระบุชัดเจนตามสัญลักษณ์ชื่อย่อ ที่กำกับในแต่ละขา อย่างไรก็ตามจะมีบางขาสัญญาณที่อาจจะมีหน้าที่ได้มากกว่าหนึ่งอย่าง ซึ่งจะไม่สามารถใช้งานในเวลาเดียวกันได้ ตัวอย่างเช่นขาสัญญาณบิต 0 ของพอร์ต 3 (ใช้ตัวย่อเป็น P3.0) อาจจะใช้เป็นขาสัญญาณเอาต์พุต หรืออินพุตตามปกติ ภายในไมโครคอนโทรลเลอร์ MCS-51 ซึ่งประกอบด้วยหน่วยการทำงานต่างๆ ภายในไอซี MCS-51 จำนวนมาก โดยแต่ละบล็อกซึ่งเป็นวงจรควบคุมรีจิสเตอร์ (REGISTER) หรือหน่วยความจำภายในของไอซี MCS-51 จะถูกเชื่อมต่อเข้าด้วยกันผ่านทางเส้นสัญญาณที่เรียกว่าบัสข้อมูลภายใน รีจิสเตอร์และหน่วยความจำเหล่านี้จะถูกนำไปใช้ระหว่างการประมวลผลคำสั่ง หน้าที่ของโปรแกรมที่ผู้ใช้สร้างขึ้นมาก็เป็นการควบคุมการรับหรือส่งข้อมูลระหว่างรีจิสเตอร์เหล่านี้ ซึ่งอาจจะมีการดำเนินการร่วมกับหน่วยการดำเนินงานประมวลผลทางคณิตศาสตร์และลอจิก(Arithmatic and Logic Unit (ALU)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 โครงสร้างหน่วยความจำภายในไมโครคอนโทรลเลอร์ MCS-51

ไมโครคอนโทรลเลอร์ MCS-51 แยกการจัดการหน่วยความจำออกเป็นสองส่วนอย่างชัดเจน คือ หน่วยความจำโปรแกรม (PROGRAM MEMORY) และหน่วยความจำข้อมูล (DATA MEMORY) หน่วยความจำทั้งสองนี้ มีหน้าที่แตกต่างไปจากกัน และใช้วิธีการอ้างแอดเดรสสัญญาณการติดต่อแยกออกจากกัน



รูปที่ 5.2 แสดงรูปร่างและการจัดวางขาต่างๆ ของไมโครคอนโทรลเลอร์ MCS-51

5.4 หน่วยความจำโปรแกรม

หน่วยความจำโปรแกรมของไมโครคอนโทรลเลอร์ MCS-51 เป็นบริเวณหน่วยความจำสำหรับเก็บข้อมูลและคำสั่งใช้งานต่างๆ ซึ่งแม้ว่าจะไม่มีการจ่ายกระแสไฟฟ้าให้กับระบบข้อมูลเหล่านี้ก็ยังคงอยู่ไม่สูญหายโครงสร้างของหน่วยความจำโปรแกรม มีลักษณะเช่นเดียวกับหน่วยความจำที่บรรจุอยู่ใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

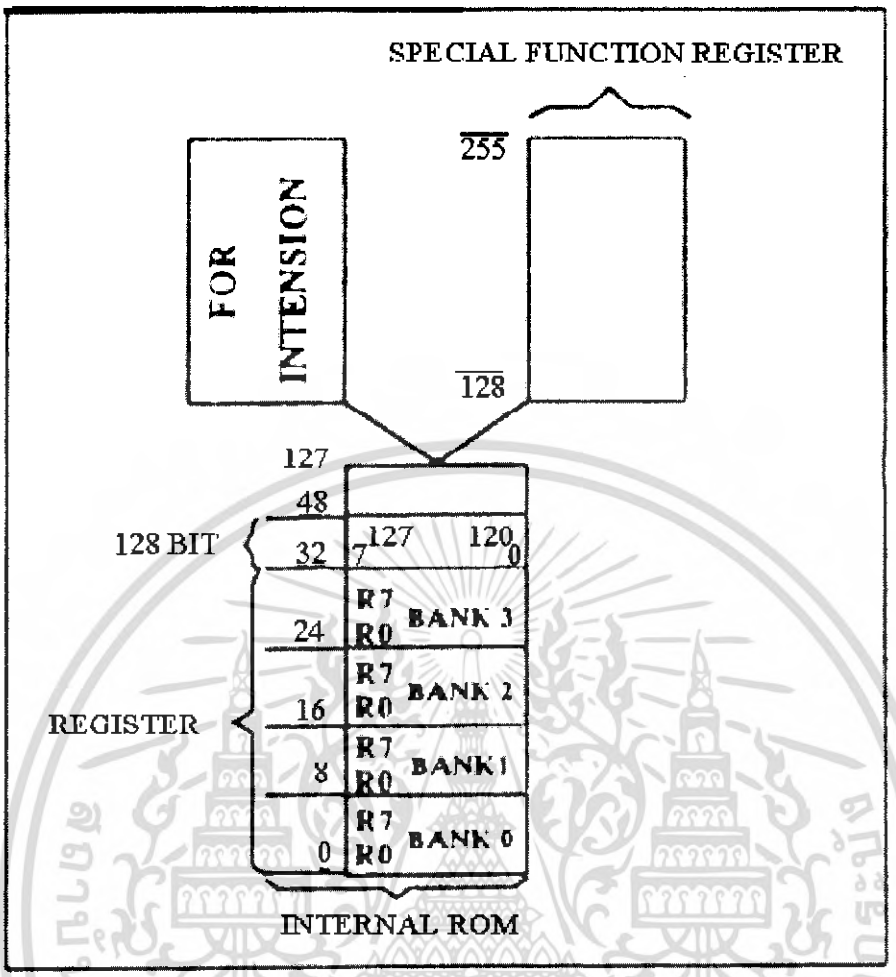
ไมโครคอนโทรลเลอร์ MCS-51 ของหน่วยความจำ ประเภทต่างๆ เช่น หน่วยความจำแบบรอม (READ ONLY MEMORY) หรือ อีพรอม (ERASABLE PROGRAMABLE READ ONLY MEMORY) ใน ไมโครคอนโทรลเลอร์ MCS-51 สามารถอ่านข้อมูลหน่วยความจำโปรแกรมนี้ได้สูงสุดไม่เกิน 64 กิโลไบต์ และแยกประเภทของหน่วยความจำโปรแกรมเป็น 2 ลักษณะ ตามตำแหน่งของหน่วยความจำนั้น คือ หน่วยความจำโปรแกรมภายใน (INTERNAL PROGRAM MEMORY) ซึ่งเป็นหน่วยความจำรอม หรือ อีพรอม ที่อยู่ภายในตัวไอซีของไมโครคอนโทรลเลอร์เอง และหน่วยความจำโปรแกรมภายนอก (EXTERNAL PROGRAM MEMORY) ซึ่งเป็นการใช้ไอซีหน่วยความจำมาทำหน้าที่เป็นหน่วยความจำโปรแกรมของระบบ

ไมโครคอนโทรลเลอร์เบอร์ต่างๆ ของตระกูล 8051 นี้สามารถขยายให้ใช้งาน ในหน่วยความจำภายนอกได้ทั้งสิ้น โดยกรณีที่มีหน่วยความจำโปรแกรมภายในอยู่แล้ว การอ้างตำแหน่งแอดเดรสที่มีทั้งในหน่วยความจำโปรแกรมภายในและภายนอกนั้นจะต้องทำการควบคุมระดับลอจิกของสัญญาณ ในขณะนั้นด้วย ขนาดหน่วยความจำโปรแกรมภายในของไมโครคอนโทรลเลอร์เบอร์ต่างๆ ภายในตระกูล 8051 จะแตกต่างกันออกไป เพื่อความเหมาะสมกับการนำไปใช้งานลักษณะต่างๆ

5.5 หน่วยความจำข้อมูล

หน่วยความจำข้อมูล (DATA MEMORY) ซึ่งโดยพื้นฐานแล้วเป็นหน่วยความจำแรมสามารถเขียนหรืออ่านข้อมูลได้ (READ OR WRITE MEMORY) ใช้สำหรับเก็บข้อมูลหรือตัวแปร ที่เกิดขึ้นในขณะที่กำลังประมวลผลโปรแกรมไว้เป็นการชั่วคราว ซึ่งโดยพื้นฐานแล้วหน่วยความจำข้อมูลจัดเป็นหน่วยความจำแรมแบบสแตติกดังนั้นเมื่อไม่มีการจ่ายไฟฟ้าให้กับระบบก็จะมีผลทำให้ข้อมูลที่จัดเก็บไว้ภายในหน่วยความจำนี้สูญไป พื้นที่ของหน่วยความจำข้อมูลของไมโครคอนโทรลเลอร์ MCS-51 มีได้สูงสุดไม่เกิน 64 กิโลไบต์ และแยกประเภทออกเป็นสองลักษณะตามตำแหน่งที่ตั้งของหน่วยความจำนั้น ตามลักษณะของหน่วยความจำโปรแกรมภายในซึ่งก็เป็นแรมที่อยู่ภายในตัวไอซีในตระกูลของไมโครคอนโทรลเลอร์ และหน่วยความจำข้อมูลภายนอกซึ่งเป็นการใช้ไอซีหน่วยความจำแรมมาเพิ่มเติมเข้าไปในวงจรลักษณะเดียวกับ การนำไอซีอีพรอมมาใช้งานเป็นหน่วยความจำโปรแกรมนั่นเอง

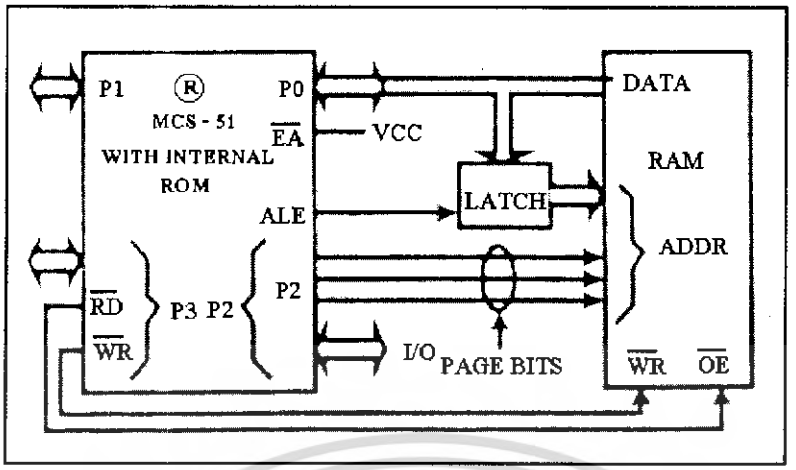
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.3 แสดงการจัดหน่วยความจำข้อมูล

โดยที่หน่วยความจำสำหรับเก็บข้อมูลของไมโครคอนโทรลเลอร์ MCS-51 นี้สามารถแบ่งออกเป็น 2 ส่วนคือ ในส่วนที่เป็นหน่วยความจำสำหรับเก็บข้อมูลภายในไอซี และหน่วยความจำสำหรับเก็บข้อมูลภายนอกไมโครคอนโทรลเลอร์ MCS-51 ทุกๆ เบอร์จะมีหน่วยความจำเก็บข้อมูลทั่วๆ ไปภายในไอซีอย่างน้อยคือ 128 ไบต์ ไปจนถึง 256 ไบต์ทั้งนี้ ขึ้นกับเบอร์ของไอซี หน่วยความจำสำหรับเก็บ ข้อมูลภายในไอซีในบริเวณ 128 ไบต์เรียกว่า LOWER 128 และในบริเวณ 128 ไบต์หลัง ที่มีเพิ่มในบางเบอร์มีชื่อเรียกว่า UPPER

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.4 แสดงการต่อกับหน่วยความจำข้อมูลภายนอกไอซี

5.6รีจิสเตอร์ที่เกี่ยวกับไมโครคอนโทรลเลอร์MCS-51

รีจิสเตอร์ในกลุ่มนี้จะเป็นรีจิสเตอร์ขนาด 16บิตที่ใช้งานเพื่อเก็บข้อมูลของตัวแอดเดรสเป็นสำคัญโดยค่าที่อยู่ภายในแอดเดรสนี้จะนำไปเป็นค่าของข้อมูลที่ส่งออกไปทางบัสแอดเดรส ในส่วนของไมโครคอนโทรลเลอร์ เพื่อบอกตำแหน่งที่ต้องการติดต่อ รีจิสเตอร์ที่จัดในกลุ่มนี้ประกอบด้วยรีจิสเตอร์ใช้งานทั่วไป (GENERAL-PURPOSE REGISTERS) รีจิสเตอร์ในกลุ่มนี้จัดเป็นพื้นที่หน่วยความจำที่ใช้ในการสนับสนุนในการประมวลผล การทำงานจากหน่วยประมวลผลทางคณิตศาสตร์และลอจิก (ALU) เพื่อให้สามารถจัดการข้อมูลให้เร็วที่สุด นอกจากนี้โปรแกรมที่ไม่ได้ใช้คำสั่งเหล่านี้ก็ยังใช้เป็นการเก็บข้อมูลตัวแปรภายในโปรแกรม จะเห็นได้ว่าชื่อของรีจิสเตอร์ไม่ว่าจะอยู่ในรีจิสเตอร์แบงก์ใด ก็จะมีชื่อว่า R0 ถึง R7 เหมือนกันทั้งสิ้น ดังนั้นในการใช้งานผู้ใช้จะต้องให้ความระมัดระวังว่า ต้องการรีจิสเตอร์นั้นๆ จากแบงก์ใดๆ ซึ่งการกำหนดเลือกแต่ละกลุ่มของรีจิสเตอร์นี้ก็ทำได้ง่าย เพียงการกำหนดค่าของบิตที่อยู่ภายในแฟลก (PSW) เท่านั้นอย่างไรก็ตามโดยทั่วไปก็มักจะมีการใช้งานรีจิสเตอร์ R0 ถึง R7 เฉพาะในแบงก์ 0 เท่านั้น ดังนั้นพื้นที่ของแบงก์อื่นๆ ที่เหลือก็สามารถนำมาใช้ในลักษณะของหน่วยความจำแรม

5.6.1 รีจิสเตอร์หน้าที่พิเศษ

เป็นรีจิสเตอร์หน้าที่พิเศษ (SFR) เป็นรีจิสเตอร์สำหรับ การควบคุมหน้าที่ และการทำงานของอุปกรณ์หรือพอร์ตของไมโครคอนโทรลเลอร์ MCS-51 ทั้งหมด ตำแหน่งของรีจิสเตอร์เหล่านี้จะจัดอยู่ในบริเวณแอดเดรส 80H - FFH การใช้งานรีจิสเตอร์หน้าที่พิเศษเหล่านี้สามารถทำได้ทั้งการระบุชื่อของรีจิสเตอร์ หรือตำแหน่งแอดเดรส ที่เป็นของรีจิสเตอร์นั้นก็ได้ การจัดพื้นที่หน่วยความจำสำหรับรีจิสเตอร์หน้าที่พิเศษเหล่านี้ โดยมีข้อสังเกตว่ารีจิสเตอร์ที่อยู่ในตำแหน่งแอดเดรสที่มีจำนวนเป็นทวีคูณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของค่า 8 จะสามารถอ้างถึงในระดับบิตได้ด้วย (นั่นคือแอดเดรส 80H 88H 90H A0H A8H B0H B8H D0H E0H และ F0H)

5.6.2แอกคิวมูเลเตอร์ (ACCUMULATOR)

หรือ ACC เป็นรีจิสเตอร์ขนาด 8 บิต ทำหน้าที่ในการเก็บข้อมูลที่จะส่งให้กับหน่วยทำงานภายในหน่วยประมวลผลกลาง และเก็บผลลัพธ์ที่ได้จากการทำงานเท่านั้น การทำงานของรีจิสเตอร์นั้นมีลักษณะเช่นเดียวกับตัวแอกคิวมูเลเตอร์ของโปรเซสเซอร์ทั่วไป การใช้งานในโปรแกรมซึ่งใช้เรียกเป็นรีจิสเตอร์ A

5.7ตัวดำเนินการในภาษาซี

ตัวนำเนินการจะเป็นตัวที่ใช้กระทำกับตัวแปร ค่าคงที่ต่าง ๆ ให้รวมเป็นค่าเดียวกันโดยอาจกระทำทางคณิตศาสตร์หรือกระทำทางลอจิกก็ได้ ในการเขียนโปรแกรมด้วยภาษาซีนั้น ตัวดำเนินการจะแบ่งออกเป็นสองกลุ่มใหญ่ๆ คือ ตัวดำเนินการที่กระทำกับ ตัวถูกกระทำตัวเดียว (single operand operators) ตัวดำเนินการที่กระทำกับตัวถูกกระทำสองตัว (two operands operators)

5.7.1 ตัวดำเนินการที่กระทำกับตัวถูกกระทำตัวเดียว

ตัวดำเนินการประเภทนี้จะกระทำกับตัวถูกกระทำเพียงตัวเดียว ประกอบด้วยตัวดำเนินการต่าง ๆ ดังนี้

-	ลบ (negate)
~	กลับค่าลอจิกของบิตข้อมูล (bit wise complement)
!	กลับค่าทางลอจิก (logical complement)
++	เพิ่มค่าขึ้นหนึ่งค่า (increment)
--	ลดค่าลงหนึ่งค่า (decrement)
*	ตัวดำเนินการทางพอยน์เตอร์
&	ตัวตำแหน่งหน่วยความจำของตัวแปร

5.7.2 ตัวดำเนินการที่กระทำกับตัวถูกกระทำสองตัว

ตัวดำเนินการประเภทนี้จะกระทำกับตัวถูกกระทำสองตัว ถ้าหากมีตัวถูกกระทำหลาย ๆ ตัว สามารถนำมาเขียนรวมกันเป็นประโยคได้ ซึ่งประกอบไปด้วยตัวดำเนินการที่ใช้กำหนดค่า ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดำเนินการทดสอบค่า ซึ่งจะทำให้ผลลัพธ์เป็นค่าทางลอจิก (จริง , เท็จ) ตัวดำเนินการทางคณิตศาสตร์ และตัวดำเนินการทางลอจิก ดังต่อไปนี้

=	กำหนดค่าในประโยค (assignment)
+	บวก
-	ลบ
*	คูณ
/	หาร (division)
%	หารแบบบวก (modulo)
&&	การแอนด์ (logical AND)
||	การออร์ (logical OR)
&	การแอนด์แบบบิตต่อบิต (bit wise AND)
|	การออร์แบบบิตต่อบิต (bit wise OR)
^	การเอ็กคลูซีฟออร์แบบบิตต่อบิต (bit wise exclusive OR)
<<	เลื่อนบิตไปทางซ้าย
>>	เลื่อนบิตไปทางขวา
==	ทดสอบว่าเท่ากันหรือไม่
!=	ทดสอบว่าไม่เท่ากันหรือไม่
>	ทดสอบว่ามากกว่าหรือไม่
<	ทดสอบว่าน้อยกว่าหรือไม่
>=	ทดสอบว่ามากกว่าหรือเท่ากับหรือไม่
<=	ทดสอบว่าน้อยกว่าหรือเท่ากับหรือไม่

นอกจากนี้ตัวดำเนินการบางประเภทสามารถนำมารวมกันเป็น compound operators ได้ ตัวอย่างเช่น ถ้ามีประโยค y = y * 2; อาจเขียนตัวดำเนินการรวมได้เป็น y *= 2; รูปแบบของตัวดำเนินการที่รวมกันได้ เป็นดังนี้

+=	บวกและให้เท่ากับ
-=	ลบและให้เท่ากับ
*=	คูณและให้เท่ากับ
/=	หารและให้เท่ากับ
%=	หารแบบบวกและให้เท่ากับ
&=	ทำการแอนด์และให้เท่ากับ
|=	ทำการออร์และให้เท่ากับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

^= ทำการเอ็กคลูซีฟออร์และให้เท่ากับ

<<= เลื่อนบิตไปทางซ้ายและให้เท่ากับ

>>= เลื่อนบิตไปทางขวาและให้เท่ากับ

5.8 ประโยคควบคุมในภาษาซี

การทำงานของโปรแกรมนั้นจะทำคำสั่งแต่ละคำสั่งเรียงลำดับกันไป และเราสามารถให้ใช้โปรแกรมตัดสินใจในการเลือกทำได้ หรือให้ทำงานใดงานหนึ่งซ้ำ ๆ ตามเงื่อนไขที่กำหนดได้โดยใช้คำสั่งควบคุมในภาษาซีนั้นจะมีประโยคคำสั่งควบคุมที่ใช้ในการเลือกทำและทำงานซ้ำ ๆ ดังนี้

5.8.1 ประโยค IF / ELSE

ประโยคคำสั่งนี้จะใช้ควบคุมทิศทางการทำงานของโปรแกรม โดยจะถูกแปลออกมาเป็นคำสั่งในภาษาแอสเซมบลีดังนี้ jz, jnz, jb, jnb, jc, และ jnc โดยมีรูปแบบประโยคดังนี้

ประโยคนี้ใช้ในการทดสอบว่าจะทำสเตตเมนต์ที่ตามมาหรือไม่ ถ้าค่าใน expression เป็นจริงหรือมีค่าไม่เป็นศูนย์จะทำสเตตเมนต์ที่ตามมา ถ้าเป็นเท็จหรือมีค่าเป็นศูนย์จะไม่ทำ และ สเตตเมนต์ที่จะทำงานนั้นอาจเป็นประโยคคำสั่งประโยคเดียว หรือเป็นสเตตเมนต์ซ้อนก็ได้ (ต้องมีปีกกาคุม)

5.9 การทำซ้ำ

คำสั่งให้โปรแกรมทำงานซ้ำถือว่าเป็นประโยคคำสั่งควบคุมอย่างหนึ่ง การทำซ้ำหรือที่เรียกว่าการทำลูปนั้นจะมีประโยคคำสั่งอยู่สามประเภทคือ for, white และ do – white ซึ่งแต่ละแบบจะต่างกันตรงเงื่อนไขของการทำซ้ำ

5.9.1 ประโยค for

ประโยคคำสั่งนี้จะใช้ในกรณีที่มีจำนวนรอบของการกระทำซ้ำที่แน่นอน โดยมีรูปแบบดังนี้

```
For (initialization ; condition ; increment)
        Statement ;
```

โดยที่ initialization เป็นค่ากำหนดเริ่มต้นให้กับตัวแปรของการทำลูป condition เป็นเงื่อนไขที่ใช้ทดสอบการทำซ้ำครั้งต่อไป ซึ่งจะเป็นการกระทำลอจิก increment เป็นการเพิ่มค่าให้ตัวแปรในการทำซ้ำแต่ละครั้ง สำหรับ statement จะเป็นสเตตเมนต์ของคำสั่งที่จะทำซ้ำ ซึ่งอาจเป็น สเตตเมนต์รวม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.9.2 ประโยค while

การทำซ้ำแบบนี้จะตรวจสอบเงื่อนไขก่อนการทำซ้ำ ถ้าเงื่อนไขเป็นจริงจะทำสเตตเมนต์ที่กำหนด และทดสอบเงื่อนไขใหม่ ถ้าเงื่อนไขเป็นเท็จจะออกจากการทำซ้ำทันที โดยมีรูปแบบดังนี้

```
While (expression)
{
statement ;
}
```

ในส่วนของ expression นั้นสามารถตรวจสอบค่าคงที่ได้ด้วย ถ้าค่าไม่เท่ากับศูนย์จะทำซ้ำ ถ้าค่าเท่ากับศูนย์จะไม่เท่าซ้ำ

5.9.3 ประโยค do – while

การทำซ้ำประเภทนี้ จะตรวจสอบเงื่อนไขภายหลังการทำสเตตเมนต์แต่ละครั้ง ถ้าหากเงื่อนไขเป็นเท็จจะออกจากการทำซ้ำทันที มีรูปแบบดังนี้

```
do{
statement ;
}
While (condition)
```

5.10 อาร์เรย์ พอยน์เตอร์ และสตรักเจอร์

ในการเขียนโปรแกรมด้วยภาษาซีถ้าหากต้องการใช้งานตัวแปรหลายตัว เราสามารถประกาศชื่อตัวแปรออกมาหลายตัวได้ เช่น x1, x2, x3, ..., xn แต่ถ้าหากตัวแปรทุกตัวใช้เก็บข้อมูลประเภทเดียวกัน เราสามารถประกาศเป็นตัวแปรอาร์เรย์ (array) ได้

5.11 ไทเมอร์/ เคาน์เตอร์ของไมโครคอนโทรลเลอร์ MCS-51

ไทเมอร์ / เคาน์เตอร์ เป็นอีกส่วนหนึ่งที่สำคัญของไมโครคอนโทรลเลอร์ เนื่องจากในการทำงานของไมโครคอนโทรลเลอร์จะต้องมีการเก็บ และตรวจสอบค่าของเวลาและจำนวนสัญญาณนาฬิกาอยู่ตลอดเวลา ทั้งนี้เพื่อประโยชน์ในการสร้างฐานเวลา สร้างสัญญาณพัลส์ เปรียบเทียบค่าเวลาหรือ ค่าการนับ รวมไปถึงการกำหนดอัตราเร็วในการสื่อสารข้อมูลของพอร์ตอนุกรมด้วย

ในการใช้งานไทเมอร์ / เคาน์เตอร์นั้น จะต้องมีการกำหนดหรือควบคุมการทำงานของไทเมอร์ / เคาน์เตอร์ โดยใช้รีจิสเตอร์ที่สำคัญ 2 ตัว คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TCON (Timer / Counter Control Register) : มีแอดเดรสอยู่ที่ 88H เป็นรีจิสเตอร์ที่ใช้ควบคุมการเปิด / ปิดไทเมอร์แต่ละตัว , และแสดงถึงการเกิดโอเวอร์โฟรวที่เกิดจากไทเมอร์ / เคาน์เตอร์ อีกทั้งเป็นตัวกำหนดลักษณะการเกิดอินเตอร์รัปต์ของสัญญาณภายนอกด้วย

TMOD (Timer / Counter Mode Control Register) : มีแอดเดรสอยู่ที่ 89H เป็นรีจิสเตอร์ที่ใช้กำหนดการทำงานว่าเป็นไทเมอร์ หรือ เคาน์เตอร์ อีกทั้งเป็นตัวกำหนดโหมดการทำงานของไทเมอร์ / เคาน์เตอร์ ซึ่งมีอยู่ 4 โหมดด้วยกัน คือ

โหมด 0 : ไทเมอร์ / เคาน์เตอร์ 13 บิต

โหมด 1 : ไทเมอร์ / เคาน์เตอร์ 16 บิต

โหมด 2 : ไทเมอร์ / เคาน์เตอร์ 8 บิด แบบตั้งค่าอัตโนมัติ

โหมด 3 : ไทเมอร์ / เคาน์เตอร์แยกส่วน

5.12 พอร์ตอนุกรมของไมโครคอนโทรลเลอร์ MCS-51

ไมโครคอนโทรลเลอร์ตระกูล MCS-51 มีวงจรสื่อสารอนุกรมแบบ ฟลูดูเพล็กซ์ 1 ชุด โดยใช้ขาสัญญาณของพอร์ต 3 คือ ขาP3.0 เป็นขารับข้อมูลหรือ RxD และขา P3.1 เป็นขาส่งข้อมูลหรือ TxD โดยวงจรสื่อสารข้อมูลแบบอนุกรมของ MCS-51 เป็นแบบอะซิงโครนัส ปกติแล้ว พอร์ตอนุกรมจะใช้ติดต่อสื่อสารกับพอร์ตอนุกรมของคอมพิวเตอร์ โดยใช้มาตรฐาน RS-232 แต่ในปัจจุบัน สามารถติดต่อกับ RS-422 หรือ RS-485 ได้แล้ว โดยใช้ IC พิเศษ ทำหน้าที่ในการแปลงสัญญาณการสื่อสาร

รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของพอร์ตอนุกรมในMCS -51 ได้แก่ SBUF (Serial data buffer register) ซึ่งเป็นบัฟเฟอร์สำหรับ ส่งข้อมูล และรับข้อมูล และรีจิสเตอร์อีกตัวที่สำคัญคือ SCON (Serial Port Control) ซึ่งใช้ในการกำหนดโหมดการทำงานของพอร์ตอนุกรม ซึ่งมี 4 โหมดด้วยกัน คือ

1. โหมด 0 เป็นการกำหนดให้พอร์ดอนุกรมทำงานในลักษณะชิฟรีจิสเตอร์
2. โหมด 1 เป็นการกำหนดให้UART ขนาด 8 บิต สามารถเลือกอัตรา บอร์ดได้
3. โหมด 2 UART ขนาด 8 บิต สามารถเลือกอัตรา บอร์ดได้
4. โหมด 3 เป็นตัวกำหนดให้เป็น UART ขนาด9 บิต สามารถเลือกอัตราบอดได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

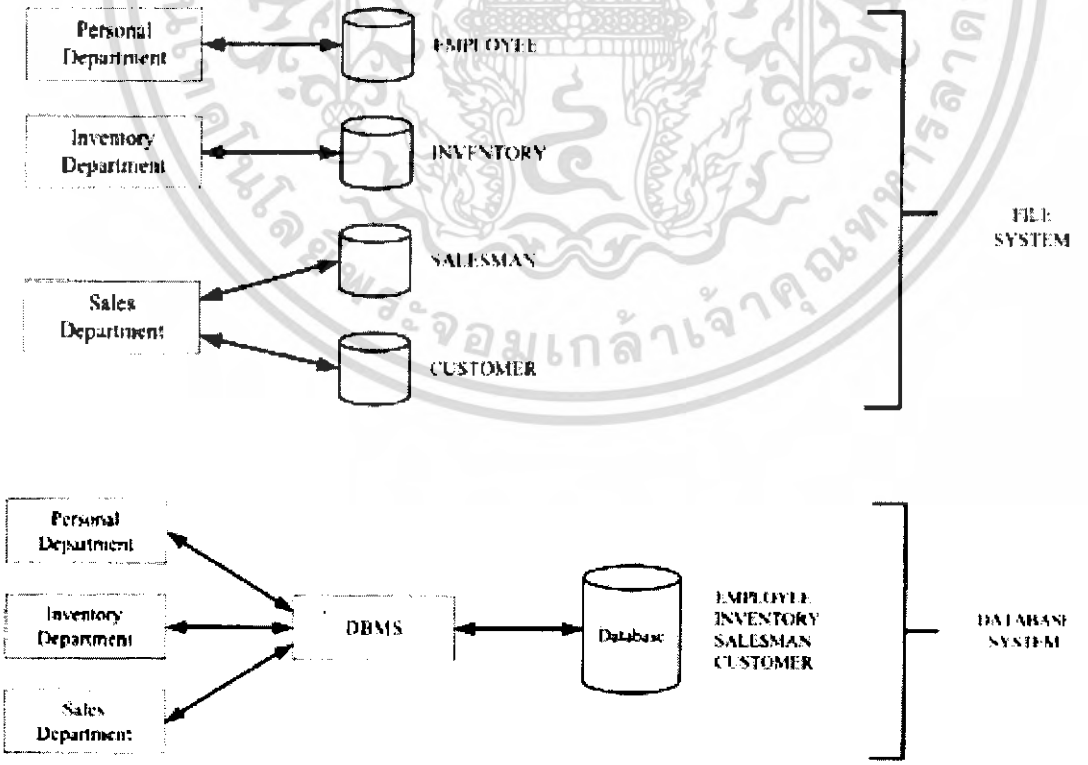
บทที่ 6 Data Base & Visual Basic

6.1 Data Base

6.1.1 ระบบฐานข้อมูล (Data Base System)

ในชีวิตประจำวันปัจจุบัน ไม่ว่าจะดำเนินงานใดๆมนุษย์จะต้องเกี่ยวข้องกับข้อมูลอย่างใดอย่างหนึ่งเสมอ เช่น การติดต่อราชการที่ต้องใช้ข้อมูลจากบัตรประชาชน การติดต่อกับธนาคารที่ต้องใช้ข้อมูลจากสมุดเงินฝาก เป็นต้น โดยเฉพาะอย่างยิ่ง เมื่อเทคโนโลยีของโลกพัฒนาขึ้น จนกระทั่งปัจจุบันที่มีคอมพิวเตอร์แทนที่การจัดเก็บข้อมูลในกระดาษ เนื่องจากปริมาณข้อมูลมีมากขึ้น ประกอบกับความต้องการใช้ข้อมูลมีมากขึ้น รวมทั้งข้อมูลได้เปลี่ยนไปเป็นปัจจัยที่มีผลต่อการแข่งขันทางธุรกิจ การจัดเก็บข้อมูลจึงได้เปลี่ยนไป และเกิดคำว่า “ฐานข้อมูล”

Data Base System คืออะไร จากปัญหาต่างๆที่เกิดขึ้นใน ได้ก่อให้เกิดการจัดเก็บข้อมูลรูปแบบใหม่ขึ้น ที่เรียกว่า “ฐานข้อมูล” การจัดเก็บข้อมูลในฐานข้อมูลนี้จะแตกต่างจากการเก็บข้อมูลในระบบแฟ้มข้อมูล เนื่องจากฐานข้อมูลเป็นการเอาข้อมูลต่างๆที่มีความสัมพันธ์กัน ซึ่งแต่เดิมจัดเก็บอยู่ในแต่ละแฟ้มข้อมูลมาจัดเก็บไว้ในที่เดียวกัน เช่น ข้อมูลพนักงาน สินค้าคงคลัง พนักงานขาย และลูกค้า ซึ่งแต่เดิมจัดเก็บอยู่ในรูปของแฟ้มของฝ่ายต่างๆ ได้ถูกนำมาจัดเก็บรวมกันไว้ในฐานข้อมูลเดียวกัน ซึ่งเป็นฐานข้อมูลรวมของบริษัท ส่งผลให้แต่ละฝ่ายใช้ข้อมูลร่วมกันและสามารถแก้ไขปัญหาต่างๆที่เกิดขึ้นในระบบแฟ้มข้อมูลได้ดังรูป



รูปที่ 6.1 ระบบแฟ้มข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลต่างๆที่ถูกจัดเก็บเป็นบานข้อมูล นอกจากจะเป็นข้อมูลที่สัมพันธ์กันแล้ว ยังต้องเป็นข้อมูลที่ได้รับการสนับสนุนการดำเนินงานอย่างน้อยอย่างใดอย่างหนึ่งขององค์กร ดังนั้นอาจกล่าวได้ว่าแต่ละฐานข้อมูลเทียบได้กับระบบแฟ้มข้อมูล 1 ระบบ และจะเรียกฐานข้อมูลที่จัดทำขึ้นเพื่อสนับสนุนการดำเนินงานอย่างใดอย่างหนึ่งนั้นว่า "ระบบฐานข้อมูล" (Data Base System) เช่นระบบฐานข้อมูลเงินเดือน ซึ่งเป็นฐานข้อมูลที่จัดเก็บข้อมูลต่างๆที่สนับสนุนการจัดทำสำมะโนประชากร เป็นต้น

6.1.2 ระบบแฟ้มข้อมูล (File System)

ในอดีตองค์กรต่างๆ มักจัดเก็บเอกสารต่างๆไว้ในแฟ้มเอกสารต่างๆ ซึ่งมีความเกี่ยวข้องทางข้อมูลน้อยหรืออาจไม่มีเลย ขึ้นอยู่กับความต้องการในการใช้ข้อมูลนั้นๆ เช่น ประวัติการรักษาพยาบาลที่โดยทั่วไปมักจะแยกเก็บเอกสารเฉพาะคนไข้แต่ละคน หรือประวัติพนักงาน ที่อาจแยกเก็บเอกสารตามฝ่ายที่สังกัด ซึ่งข้อมูลพนักงานของแต่ละคนในแต่ละแฟ้มเอกสารจะมีความเกี่ยวข้องกันตามฝ่ายที่สังกัด เช่นแฟ้มเอกสารประวัติพนักงานฝ่ายธุรการ แฟ้มเอกสารพนักงานฝ่ายการเงิน เป็นต้น แต่ต่อมาเมื่อองค์กรมีขนาดใหญ่ขึ้น จากเดิมที่สามารถค้นหาเอกสารจากแฟ้มข้อมูลเพียงแฟ้มเดียว ก็เริ่มต้องค้นหาเอกสารจากแฟ้มหลายแฟ้มจำนวนมากขึ้น ส่งผลให้การค้นเอกสารเป็นงานที่ต้องใช้เวลามาก และมีความยากลำบากมากขึ้น การจัดเก็บเอกสารในคอมพิวเตอร์ จึงถูกริเริ่มมาใช้ในองค์กรแทนการจัดเก็บในรูปแบบเดิม แต่การจัดเก็บเอกสารในคอมพิวเตอร์ยุคแรกๆ ยังคงไม่ค่อยมีประสิทธิภาพมากนัก เนื่องจากยังมีรูปแบบการจัดเก็บเอกสารคล้ายกับการจัดเก็บแบบเดิมอยู่ โดยเป็นเพียงการนำเอกสารต่างๆในแต่ละแฟ้มเอกสารมาจัดเก็บในรูปของแฟ้มข้อมูลแทน และด้วยเหตุนี้จึงต้องอาศัยผู้ชำนาญ เช่น โปรแกรมเมอร์ หรือนักวิเคราะห์เข้ามาช่วยกำหนดโครงสร้างของแฟ้มข้อมูล เพื่อที่จะสารถนำแฟ้มข้อมูลนั้นไปจัดเก็บข้อมูลและสามารถนำไปประมวลผลได้ตามความต้องการ

จากบทบาทของคอมพิวเตอร์ที่เข้ามามีอิทธิพลในการดำเนินการขององค์กร ส่งผลให้การจัดเก็บข้อมูลในแฟ้มข้อมูล มีการใช้แพร่หลายมากยิ่งขึ้น จากเดิมที่เพียง 2 หรือ 3 แฟ้มข้อมูล ได้เพิ่มขึ้นเป็นจำนวน 10 หรือ 20 แฟ้มข้อมูล ดังนั้นจึงต้องมีการเข้ามาควบคุมด้านโครงสร้าง และการใช้งานแฟ้มข้อมูลต่างๆ ให้มีความเหมาะสมต่อการใช้งานมากยิ่งขึ้น และรวบแฟ้มข้อมูลเหล่านี้เข้าเป็นระบบที่เรียกว่า "ระบบแฟ้มข้อมูล" (File System)

การใช้งานระบบแฟ้มข้อมูล จะต้องอาศัยโปรแกรมเมอร์พัฒนาโปรแกรม เพื่ออ่านข้อมูลต่างๆขึ้นมาประมวลผล เพื่อให้ได้ผลลัพธ์ตามที่ผู้ใช้ต้องการ ภาษาคอมพิวเตอร์ที่ใช้พัฒนาโปรแกรมโดยทั่วไปจะได้แก่ ภาษาคอมพิวเตอร์ในยุคที่ 3 (Third Generation Language : 3GL) เช่น ภาษา COBAL, FORTRAN, BASIC เป็นต้น แต่ภาษาคอมพิวเตอร์เหล่านี้มีข้อจำกัดในการเรียกใช้ข้อมูลจากแฟ้มข้อมูล เนื่องจากภาษาคอมพิวเตอร์เหล่านี้ จะต้องอ้างถึงแฟ้มข้อมูลตามโครงสร้างทางกายภาพของข้อมูลภายในแฟ้มข้อมูลนั้น เช่น เมื่อต้องการ อ้างถึงแฟ้มข้อมูลลูกค้า ซึ่งมีโครงสร้างดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CUSTOMER

CUST_NO	CUST_NAME	CUST_ADDRESS	CUST_PHONE
C0001	ABC Industry	111 Moo 4 Samutprakarn	552-000
C0002	Thai Steel CO., Ltd	100 Petchkasem Ratchaburi	321-180
C0003	ITT Computer CO., Ltd	90 III Bldg.Bangkok	433-0015
C0004	KKK (Thailand) CO., Ltd	18 Sathorn Bldg. Bangkok	236-8897

ตารางที่ 6.1 ตัวอย่างโครงสร้างข้อมูลในแฟ้มลูกค้า

ในภาษา COBAL จะต้องมีการนิยาม (Declare) โครงสร้างของ Record เช่น ประเภทของข้อมูล ขนาดของแต่ละ Field ไว้ในส่วนของ Data Division ตามโครงสร้างทางกายภาพของ Record ในแฟ้มลูกค้าก่อน จึงจะสมารถอ้างถึง Field ต่างๆ ในโปรแกรมได้ดังตัวอย่าง

```
01 CUSTOMER.
    02 CUST_NO          PIC X(5).
    02 CUST_NAME        PIC X(50).
    02 CUST_ADDRESS     PIC X(50).
    02 CUST_PHONE       PIC X(8).
```

ด้วยเหตุนี้จึงส่งผลให้การพัฒนาในแต่ละโปรแกรมขึ้นใช้งานกับแฟ้มข้อมูลต่างๆมีความซับซ้อน และต้องใช้เวลาค่อนข้างมาก รวมทั้งการทำให้แต่ละโปรแกรมถูกผูกติดอยู่กับแฟ้มข้อมูลต่างๆ ดังนั้นเมื่อต้องมีการเปลี่ยนแปลงโครงสร้างของแฟ้มข้อมูลใดข้อมูลหนึ่ง จึงต้องแก้ไขโปรแกรมต่างๆ ที่มีการเรียกใช้แฟ้มข้อมูลนั้นตามไปด้วย ส่งผลให้ค่าใช้จ่ายในการบำรุงรักษาค่อนข้างสูง โดยเฉพาะอย่างยิ่งในกรณีที่ต้องมีการแก้ไขโปรแกรมจำนวนมาก

ในยุคเริ่มต้นของการใช้ระบบแฟ้มข้อมูล แต่ละหน่วยงานในองค์กร จะมีการสร้างระบบแฟ้มข้อมูลขึ้นใช้งานภายในหน่วยงานของตนเอง เช่น ระบบแฟ้มข้อมูลการขายของฝ่ายการตลาด ระบบแฟ้มข้อมูลพนักงานของฝ่ายการพนักงาน ระบบแฟ้มข้อมูลสินค้าคงคลังของฝ่ายคลังสินค้า เป็นต้น ซึ่งแต่ละหน่วยงานนั้น จะมีการพัฒนาโปรแกรมที่ใช้ข้อมูลจากระบบแฟ้มข้อมูลการขายในการออกใบสั่งซื้อ ใบกำกับสินค้า รายงานแสดงยอดการขายในแต่ละเดือน หรือฝ่ายคลังสินค้าที่มีการพัฒนา

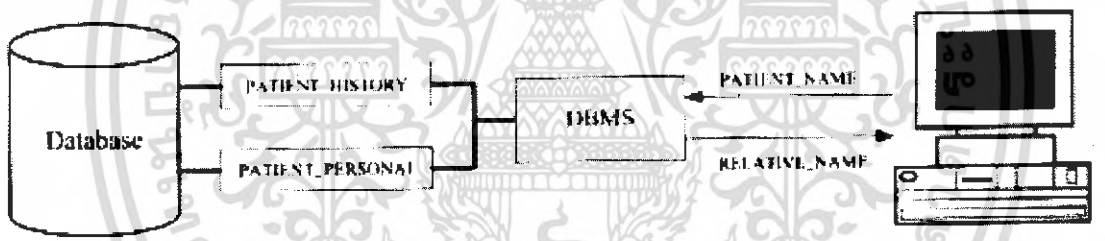
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมที่ใช้ข้อมูลจากระบบแฟ้มข้อมูลสินค้าคงคลังในการพิมพ์รายงานแสดงยอดคงคลังของสินค้าต่างๆ

6.1.3 องค์ประกอบของระบบฐานข้อมูล

1.ข้อมูล (Data)

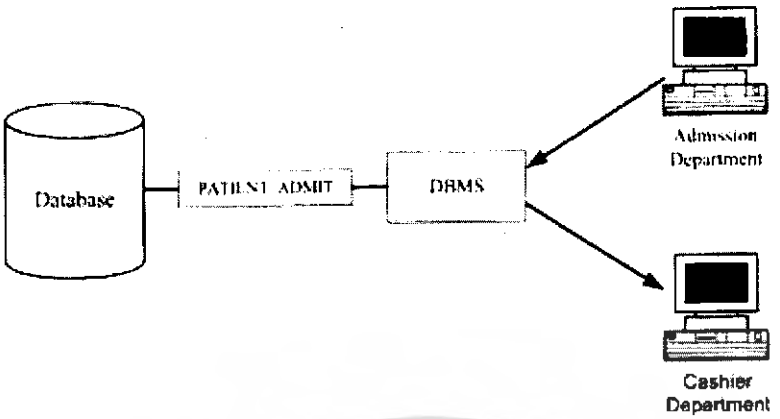
ข้อมูลที่จัดเก็บอยู่ในฐานข้อมูล ไม่ว่าจะเป็นเครื่องคอมพิวเตอร์ส่วนบุคคล ไปจนถึงเครื่องคอมพิวเตอร์ขนาดใหญ่อย่างเช่นเครื่อง Mainframe ข้อมูลในแต่ละส่วนจะต้องสามารถนำมาใช้ร่วมกันได้ เช่นเมื่อแพทย์รักษาผู้ป่วย แพทย์จะอาศัยประวัติจากการรักษาพยาบาลของผู้ป่วย (PATIENT_HISTORY) มาประกอบการรักษา แต่กรณีที่ต้องการติดต่อญาติผู้ป่วย ซึ่งข้อมูลในส่วนนี้ไม่ปรากฏอยู่ในประวัติการรักษาพยาบาล ทางพยาบาลสามารถนำชื่อผู้ป่วย (Field "PATIENT_NAME") ไปค้นหาชื่อญาติ (Field "RELATIVE_NAME") ในทะเบียนผู้ป่วย (PATIENT_PERSONAL) ได้โดยไม่ต้องเก็บชื่อญาติของผู้ป่วยในประวัติการรักษาพยาบาลแต่อย่างใด ดังรูป



รูปที่ 6.2 ตัวอย่างการจัดเก็บข้อมูลของผู้ป่วย

นอกจากคุณลักษณะนี้แล้ว ในเครื่องคอมพิวเตอร์ขนาดใหญ่ที่มีผู้ใช้จำนวนมาก ข้อมูลในฐานข้อมูลจะต้องสามารถถูกใช้ร่วมกัน จากผู้ใช้หลายๆคนได้ เช่น ข้อมูลในการจองห้องพักของผู้ป่วย (PATIENT_ADMIN) จะต้องสามารถนำไปใช้ในการออกใบเสร็จรับเงินเพื่อเก็บค่ารักษาพยาบาลโดยฝ่ายการเงินได้ในขณะเดียวกัน ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.3ตัวอย่างการใช้ข้อมูลร่วมกันของแต่ละฝ่าย

2.ฮาร์ดแวร์ (Hard Ware)

อุปกรณ์ทางคอมพิวเตอร์ที่มีความเกี่ยวข้องกับระบบฐานข้อมูลจะประกอบด้วย 2 ส่วนหลักๆ ดังนี้

* หน่วยความจำสำรอง (Secondary Storage) เนื่องจากเป็นอุปกรณ์ทางคอมพิวเตอร์ ที่ใช้จัดเก็บข้อมูลของฐานข้อมูล ดังนั้นสิ่งที่ต้องคำนึงถึงสำหรับอุปกรณ์ในส่วนนี้คือ ความจุของหน่วยความจำสำรองที่นำมาใช้ จัดเก็บฐานข้อมูลนั้น

* หน่วยประมวลผลและหน่วยความจำหลัก เนื่องจากเป็นอุปกรณ์ที่จะต้องทำงานร่วมกันเพื่อนำข้อมูลจากฐานข้อมูลขึ้นมาประมวลผลตามคำสั่งที่กำหนด ดังนั้นสิ่งที่ต้องคำนึงถึงอุปกรณ์ส่วนนี้ คือ ความเร็วของหน่วยประมวลผล และขนาดของหน่วยความจำหลักของเครื่องคอมพิวเตอร์ที่นำมาใช้ประมวลผลร่วมกับฐานข้อมูลนั้น

3.ซอฟท์แวร์ (Software)

ในการติดต่อข้อมูลกับฐานข้อมูลของผู้ใช้ จะต้องกระทำผ่านทางโปรแกรมที่มีชื่อว่า Data Base Management System DBMS

หน้าที่หลักของโปรแกรม DBMS คือทำให้การเรียกใช้ข้อมูลจากฐานข้อมูล เป็นอิสระจากส่วนของ Hardware หรือกล่าวอีกนัยหนึ่ง โปรแกรม DBMS จะมีหน้าที่จัดการและควบคุมความถูกต้อง ความซ้ำซ้อนและความสัมพันธ์ต่างๆภายในฐานข้อมูลแทนโปรแกรมเมอร์ ส่งผลให้ผู้ใช้สามารถเรียกใช้ข้อมูลจากฐานข้อมูลได้โดยที่ไม่จำเป็นต้องทราบโครงสร้างทางกายภาพของข้อมูลในระดับที่ลึกเช่นเดียวกับโปรแกรมเมอร์ เนื่องจากโปรแกรม DBMS นี้ จะมีส่วนของ Query Language ซึ่งเป็นภาษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ประกอบด้วยคำสั่งต่างๆที่ใช้ในการจัดการและเรียกใช้ข้อมูลจากฐานข้อมูล ซึ่งสามารถนำไปใช้ร่วมกับภาษาคอมพิวเตอร์อื่นๆได้เพื่อพัฒนาเป็นโปรแกรมที่สามารถเรียกใช้ข้อมูลมาเพื่อประมวลผล สำหรับรายละเอียดของโปรแกรม DBMS จะกล่าวถึงต่อไป

4.ผู้ใช้ระบบฐานข้อมูล (User)

ผู้ที่เรียกใช้ข้อมูลจากฐานข้อมูลมาใช้งานสามารถแบ่งออกเป็น 3 กลุ่มได้ดังนี้

*Application Programmer ได้แก่ผู้ที่ทำหน้าที่พัฒนาโปรแกรม เพื่อเรียกใช้ข้อมูลจากระบบฐานข้อมูลมาประมวลผล โดยที่โปรแกรมที่พัฒนาขึ้นส่วนใหญ่ มักจะใช้ร่วมกับคำสั่งในกลุ่ม Data Manipulation Language : DML ของ Query Language เพื่อเรียกใช้ข้อมูลจากฐานข้อมูล สำหรับรายละเอียดของคำสั่งจะกล่าวต่อไป

*End User ได้แก่ ผู้ที่นำข้อมูลจากฐานข้อมูลไปใช้งาน ซึ่งแบ่งออกเป็น 2 กลุ่มดังนี้

Naive User ได้แก่ ผู้ใช้ที่เรียกใช้ข้อมูลจากฐานข้อมูลโดยอาศัยโปรแกรมที่พัฒนาขึ้น

Sophisticated User ได้แก่ ผู้ที่เรียกใช้ข้อมูลจากฐานข้อมูลโดยคำสั่งของ Query Language ซึ่งโดยทั่วไปผลิตภัณฑ์ทางด้านฐานข้อมูลที่จำหน่ายอยู่ในตลาดจะมีส่วนที่ยอมให้ผู้ใช้ ได้ใช้ประโยคคำสั่งของ Query Language เพื่อเรียกใช้ข้อมูลจากฐานข้อมูลได้โดยตรง สำหรับประโยคคำสั่งเหล่านี้จะถูก Query Processor ของโปรแกรม DBMS แปลงให้อยู่ในรูปของคำสั่งในกลุ่ม Data Manipulation Language

* Data Base Administrator :DBA ได้แก่ผู้บริหารที่ควบคุมและตัดสินใจในการกำหนดโครงสร้างฐานข้อมูล ชนิดของข้อมูล วิธีจัดเก็บข้อมูล รูปแบบในการเรียกใช้ข้อมูล ความปลอดภัยของข้อมูล และกฎระเบียบที่ใช้กำหนดความถูกต้องของข้อมูลในฐานข้อมูล โดยอาศัยคำสั่งในกลุ่ม Data Definition Language : DDL ซึ่งเป็นอีกส่วนหนึ่งของ Query Language เป็นตัวกำหนด

หน้าที่ของ DBMS

1. ทำหน้าที่แปลงคำสั่งที่ใช้จัดการกับข้อมูลภายในฐานข้อมูล ให้อยู่ในรูปแบบที่ฐานข้อมูลเข้าใจ
2. ทำหน้าที่ในการนำคำสั่งต่างๆ ซึ่งได้รับการแปลแล้ว มาสั่งให้ฐานข้อมูลทำงาน เช่น การเรียกใช้ข้อมูล (Retrive) การจัดเก็บข้อมูล (Update) การลบข้อมูล (Delete) การเพิ่มข้อมูล (Add) เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ทำหน้าที่ป้องกันความเสียหายที่จะเกิดขึ้นกับข้อมูลภายในฐานข้อมูล โดยจะคอยตรวจสอบว่าคำสั่งใดสามารถทำงานได้ และคำสั่งใดที่ไม่สามารถทำงานได้
4. ทำหน้าที่รักษาความสัมพันธ์ของข้อมูลภายในฐานข้อมูลให้มีความถูกต้องอยู่เสมอ
5. ทำหน้าที่เก็บรายละเอียดต่างๆที่เกี่ยวข้องกับข้อมูลภายในฐานข้อมูลไว้ใน Data Dictionary ซึ่งรายละเอียดเหล่านี้มักถูกเรียกว่า ข้อมูลของข้อมูล (Metadata)
6. ทำหน้าที่ควบคุมให้ฐานข้อมูลทำงานอย่างถูกต้องและมีประสิทธิภาพ

6.1.4 ประโยชน์ของฐานข้อมูล

1.สามารถลดความซ้ำซ้อนของข้อมูล โดยไม่จำเป็นต้องจัดเก็บข้อมูลที่ซ้ำๆกันไว้ในระบบแฟ้มข้อมูลของแต่ละหน่วยงานเหมือนเช่นเดิม แต่สามารถนำข้อมูลมาใช้ร่วมกันในคุณลักษณะ Integrated แทน

2.สามารถหลีเลี่ยงความขัดแย้งของข้อมูล เนื่องจากไม่จำเป็นต้องเก็บข้อมูลที่ซ้ำซ้อนกันในหลายแฟ้มข้อมูลดังนั้นในการแก้ไขข้อมูลในแต่ละชุดจะไม่ก่อให้เกิดค่าที่แตกต่างกันได้

3. แต่ละหน่วยงานในองค์กรสามารถใช้ข้อมูลร่วมกันได้

4. สามารถกำหนดให้ข้อ มูลมีรูปแบบที่เป็นมาตรฐานเดียวกันได้ เพื่อให้ผู้ใช้ข้อมูลในฐานข้อมูลชุดเดียวกันสามารถเข้าใจและสื่อสารถึงความหมายเดียวกัน

5. สามารถกำหนดความปลอดภัยให้กับข้อมูลได้ โดยกำหนดความสามารถในการเรียกใช้ข้อมูลของผู้ใช้แต่ละคนให้แตกต่างตามความรับผิดชอบ

6. สามารถรักษาความถูกต้องของข้อมูลได้ โดยระบุเกณฑ์ในการกำหนดความผิดพลาดที่อาจเกิดขึ้นจากการป้อนข้อมูลผิด

7. สามารถตอบสนองความต้องการใช้ข้อมูลได้หลายรูปแบบ

8. ทำให้ข้อมูลเป็นอิสระจากโปรแกรมที่เราใช้งานข้อมูลนั้น ซึ่งส่งผลให้ผู้พัฒนาโปรแกรมสามารถแก้ไขโครงสร้างข้อมูล โดยไม่กระทบต่อโปรแกรมที่เรียกใช้ข้อมูลนั้น

6.1.5 ภาษาทางด้านฐานข้อมูล

ผลิตภัณฑ์ทางด้านฐานข้อมูลที่มีโครงสร้างในแบบRelational ซึ่งมีจำหน่ายอยู่ในท้องตลาด สิ่งที่จำเป็นสำหรับผลิตภัณฑ์เหล่านี้ ได้แก่ การมีภาษาทางด้านฐานข้อมูลหรือที่เรียกว่า Query Language เช่นภาษา Structured Query Language (SQL) ภาษา Query – by – Example (QBE) และภาษา Quel เป็นต้นภาษาเหล่านี้ได้ถูกพัฒนาขึ้นจากแนวคิดที่ต่างกัน เช่น ภาษา QBE ซึ่งถูกพัฒนาขึ้นจากแนวคิด Relational Calculus ส่วนภาษา Quel ถูกพัฒนาขึ้นจากแนวคิดของ Tuple

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Relational Calculus ในขณะที่ภาษา SQL ถูกพัฒนาขึ้นจากแนวคิดของ Relational Calculus และ Relational Algebra เป็นหลัก แต่อย่างไรก็ตามภาษาที่ได้รับความนิยมมากที่สุดได้แก่ ภาษา Structured Query Language

Struetured Query Language เป็นภาษาทางด้านฐานข้อมูล ที่นิยมมากที่สุดภาษาหนึ่ง โดยมักถูกเรียกย่อๆว่า SQL เริ่มต้นพัฒนาครั้งแรกโดย San Jose Research Laboratory ของบริษัท IBM โดยแรกเริ่มเรียกว่า Sequel และได้ถูกนำมาใช้เป็นต้นแบบของภาษา SQL ของผลิตภัณฑ์ทางด้านฐานข้อมูลจำนวนมาก แต่อย่างไรก็ตามภาษา SQL ยังมีความแตกต่างกันในรายละเอียดทางด้านการใช้งาน คำสั่งต่างๆของภาษา SQL สามารถแบ่งตามลักษณะการใช้งานออกได้เป็น 3 กลุ่ม ดังนี้

1. Data Definition Language : DDL เป็นกลุ่มคำสั่งที่ใช้สร้างฐานข้อมูล หรือใช้สำหรับกำหนดโครงสร้างให้กับ Relation ภายในบานข้อมูล เช่น การเพิ่ม เปลี่ยนแปลง ลบ เป็นต้น
2. Data Manipulation : DML เป็นกลุ่มคำสั่งที่พัฒนาขึ้นตามแนวคิดของ Relational Algebra และ Record Relation Calculus โดยประกอบด้วยคำสั่งสำหรับเพิ่ม ลบ หรือเปลี่ยนแปลงข้อมูลในฐานข้อมูล
3. Data Query Language เป็นกลุ่มคำสั่ง DML ประเภทหนึ่งที่ใช้ในการเลือกข้อมูลจาก Relation ขึ้นมาแสดงผลตามรูปแบบที่ต้องการ

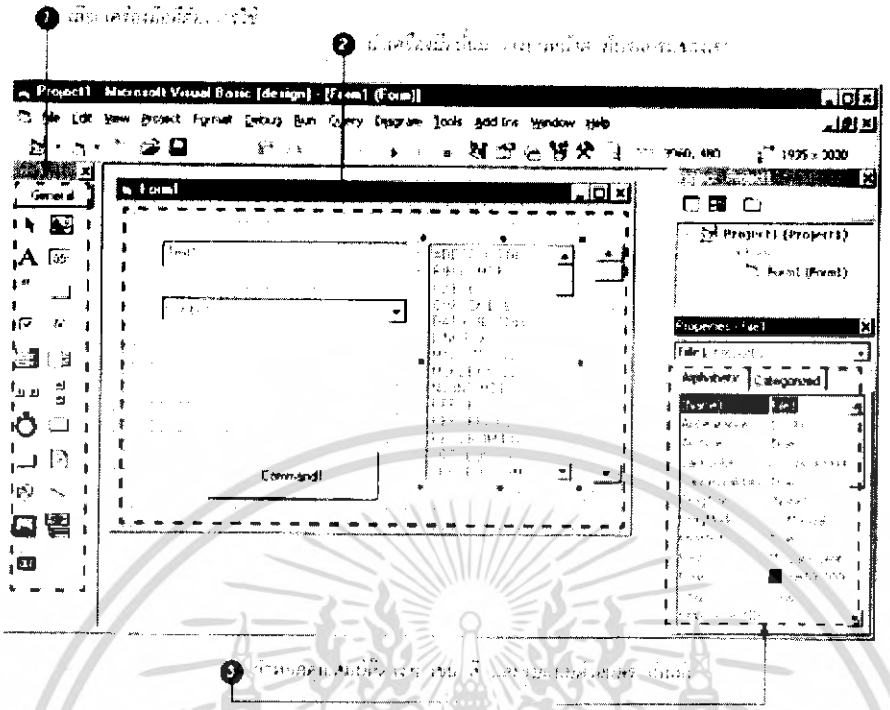
6.2 Visual Basic

ก่อนทำความรู้จักกับ VB6 เราจะต้องทราบความหมายของการเขียนโปรแกรมก่อน ซึ่งการเขียนโปรแกรมก็คือการสั่งให้คอมพิวเตอร์ทำงานตามที่เราต้องการ เช่น โปรแกรมฝึกพิมพ์ดีดเป็นโปรแกรม ที่สั่งให้โปรแกรมตอบโต้กับการกดแป้นคีย์บอร์ด เพื่อสอนให้ผู้ใช้พิมพ์ดีด เป็นต้น

สำหรับ VB6 เป็นเครื่องมือที่ใช้สร้างโปรแกรมบนระบบปฏิบัติการ Windows ที่ใช้งานง่าย โดยในการสร้างโปรแกรมใน VB6 นั้นจะเป็นการเลือกเครื่องมือต่างๆมาออกแบบหน้าจอของโปรแกรมที่เราจะสร้างขึ้น ซึ่งเราเรียกการเขียนโปรแกรมลักษณะนี้ว่า Visual Programming การเขียนโปรแกรมแบบนี้ เราไม่จำเป็นต้องเขียนคำสั่งต่างๆมากนักก็สามารถสร้างโปรแกรมได้อย่างรวดเร็ว

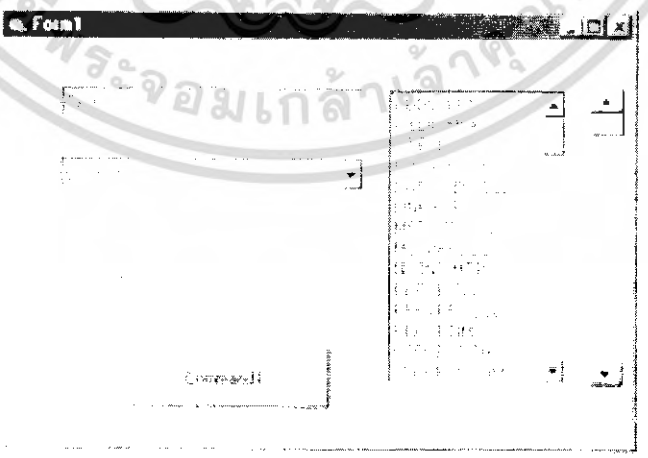
การเขียนโปรแกรมแบบ Visual Programming นั้นมีลักษณะดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.4 การเขียนโปรแกรมแบบ Visual Programming

1. เลือกเครื่องมือที่ต้องการใช้งาน
2. นำเครื่องมือนั้นมาวางบนหน้าต่างโปรแกรมของเรา
3. กำหนดคุณสมบัติต่างๆเช่น สี และรูปแบบตัวอักษร เป็นต้น
4. เมื่อสั่งรันโปรแกรม หน้าต่างของโปรแกรมจะมีลักษณะคล้ายกับที่ออกแบบไว้



รูปที่ 6.5 หน้าต่างของโปรแกรมเมื่อทำการสั่งรัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากที่เราได้ออกแบบหน้าจอโปรแกรมตามวิธีดังกล่าวมาแล้วเราจำเป็นต้องเขียนโปรแกรมควบคุมการทำงานด้วยโดยใช้ภาษา Basic (ย่อมาจาก Beginners All-Purpose Symbolic Instruction Code) ซึ่งเป็นภาษาที่ใช้งานง่ายเหมาะสำหรับผู้เริ่มต้นศึกษาการเขียนโปรแกรมบน Windows

นอกจากนี้โปรแกรมอื่นๆของไมโครซอฟท์เช่น Microsoft Excel , Microsoft Access เป็นต้นก็ใช้ภาษา Visual Basic เป็นส่วนหนึ่งของโปรแกรมด้วยจึงสามารถใช้ความรู้ที่มีอยู่เกี่ยวกับ VB6 ในการเขียนโปรแกรมต่างๆด้วยเครื่องมือเหล่านั้นได้ซึ่งจะช่วยให้การลงทุนศึกษา VB6 นั้นคุ้มค่า

สำหรับ VB6 นั้นเป็นเครื่องมือที่สร้างโปรแกรมต่างๆได้หลากหลายดังต่อไปนี้

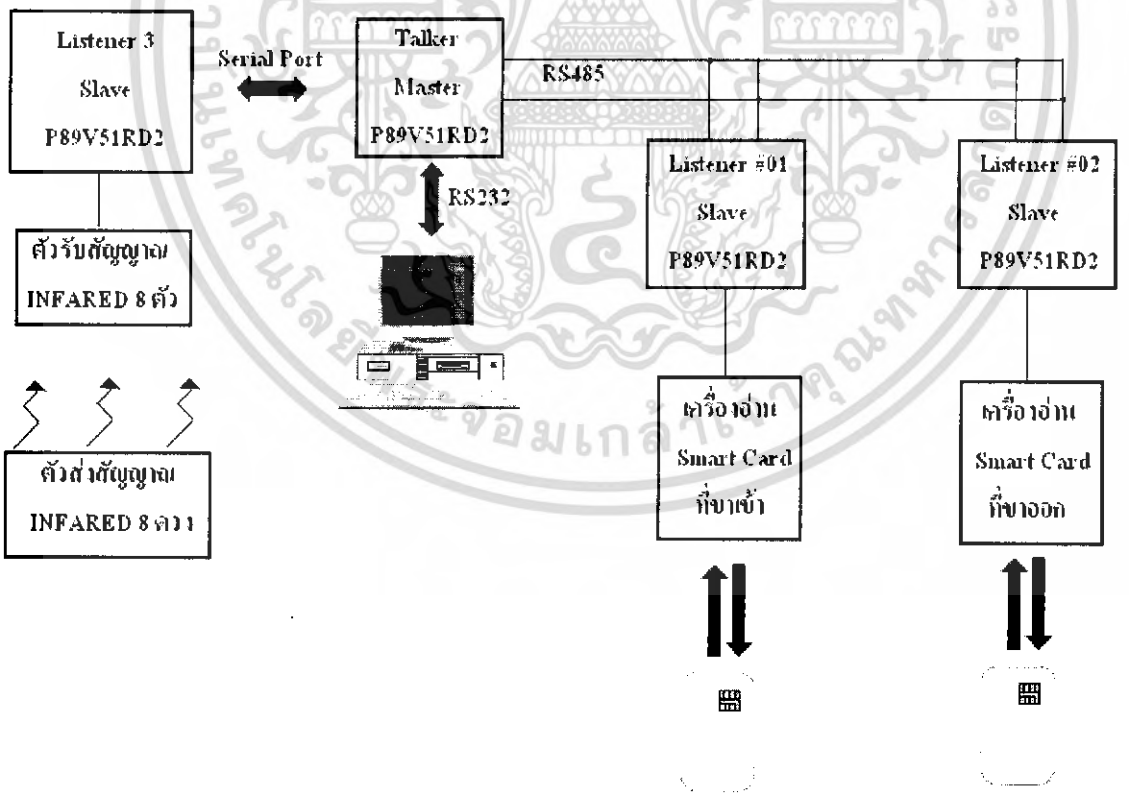
1. โปรแกรมทั่วไปที่รันบนระบบปฏิบัติการ Windows โดยเราสามารถสร้างโปรแกรมทางด้านกราฟฟิก โปรแกรมจัดการไฟล์ โปรแกรมคำนวณพื้นฐาน ให้ตรงกับความต้องการของเราได้
2. โปรแกรมทางด้านบานข้อมูล VB6 นั้นช่วยให้การสร้างโปรแกรมฐานข้อมูลเป็นเรื่องง่าย เนื่องจากมีเครื่องมือต่างๆเกี่ยวกับฐานข้อมูลอย่างครบถ้วน เช่นเครื่องมือในการติดต่อกับฐานข้อมูลทั้ง Microsoft Access หรือฐานข้อมูลบนระบบ Client-Server เช่น Microsoft SQL Server โดยการติดต่อกับฐานข้อมูลนั้นเพียงแต่เรากำหนดตำแหน่งของฐานข้อมูลพร้อมทั้งข้อมูลที่จำเป็นในการติดต่อกับฐานข้อมูลเท่านั้น เราก็สามารถติดต่อกับฐานข้อมูลนั้นได้ทันที
3. คอมโพเนนท์ทางด้าน Active X ซึ่งได้แก่ Active X Component , Active X Control และ Active X Document ซึ่งเป็นเครื่องมือที่ช่วยให้เราสามารถนำส่วนของโปรแกรมที่เราได้สร้างแล้วไปใช้ในโปรแกรมอื่นๆได้เช่น Microsoft Excel เป็นต้น
4. โปรแกรมที่รันบนอินเตอร์เน็ตหรืออินทราเน็ตผ่านทาง Web Browser ด้วยความสามารถของ VB6 ช่วยให้เราสามารถสร้างโปรแกรมที่รันบนอินเตอร์เน็ตได้อย่างง่ายดาย โดยที่ไม่ต้องเรียนรู้การเขียนคำสั่งด้วยภาษา HTML หรือภาษา Script ที่ใช้งานบนอินเตอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7 การออกแบบและหลักการทำงานของวงจร

7.1 องค์ประกอบของระบบโดยรวม

การประยุกต์ใช้งานเครื่องตรวจสอบสิ่งกีดขวางร่วมกับสมาร์ตการ์ด ในโครงงานนี้ได้ถูกออกแบบมาเพื่อใช้ในงานด้านการอำนวยความสะดวก นั่นคือ ในส่วนของการอำนวยความสะดวกนั้น จะเป็นการนำเครื่องตรวจสอบสิ่งกีดขวางมาใช้ในการตรวจสอบว่าที่ ณ ตำแหน่งนั้น ๆ ที่เราต้องการที่จะจอดมีรถจอดอยู่หรือไม่ จะได้ขับรถไปจอดในตำแหน่งที่ต้องการได้โดยไม่ต้องเสียเวลาในการวนหาที่จอดรถ และใช้ประโยชน์ของสมาร์ตการ์ดในการควบคุมการเข้าออกที่จอดรถ โดยจะมีการตรวจสอบรหัสของบัตรขณะที่มีการเข้าออกผ่านทางเข้า ออกและสามารถบันทึกเวลาในขณะเข้าออก รวมไปถึงการคำนวณยอดเงินล่วงหน้าก่อนเข้าจอด ว่ารถที่จะนำเข้าไปจอดนั้นจำนวนเงินที่มีในฐานข้อมูลสามารถจอดได้นานกี่ชั่วโมง จากนั้นเมื่อนำรถออกจากที่จอดรถก็จะทำการหักลบเงินที่มีในฐานข้อมูลตามเวลาที่จอดไป จากที่กล่าวมาข้างต้นทั้งหมดนี้คือระบบโดยรวมที่ได้ออกแบบไว้ สามารถเขียนเป็นภาพโครงสร้างโดยรวมอย่างง่ายๆได้ดังรูปที่ 7.1



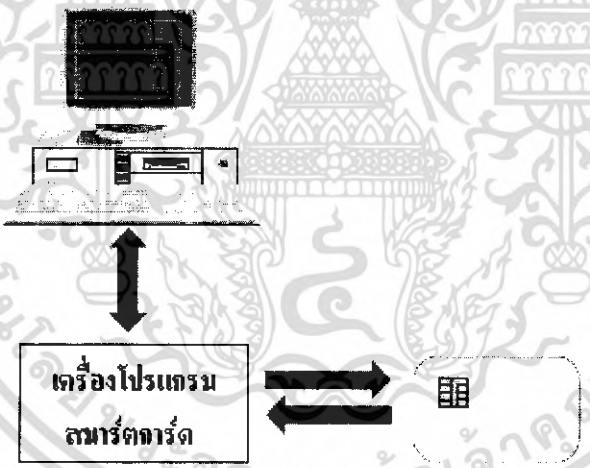
รูปที่ 7.1 โครงสร้างของระบบทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากข้างต้น สิ่งที่สำคัญที่สุดสำหรับการสร้างระบบนี้มี 3 ส่วนคือ ส่วนของเครื่องตรวจสอบการเคลื่อนไหวใช้โมดุล THOP34836 ที่ทำหน้าที่เป็นโมดูลตัวรับ และ Infared เป็นตัวส่ง ส่วนที่สองคือส่วนของสมาร์ตการ์ด โดยทั้งสองส่วนนั้นจะต่อผ่าน P89V51RD2 ที่ทำหน้าที่เป็น Slave และส่วนที่สามคือ บอร์ดควบคุมที่คอยทำหน้าที่ติดต่อระหว่างคอมพิวเตอร์ กับ เครื่องตรวจสอบการเคลื่อนไหว และ เครื่องตรวจสอบการเคลื่อนไหว ผ่านทางพอร์ท RS232 เพื่อแสดงผลผ่านหน้าจอคอมพิวเตอร์ โดยหัวใจสำคัญที่ควบคุมการทำงานคือ P89V51RD2

7.1.1 องค์ประกอบของระบบโดยรวมของเครื่องอ่าน เขียน สมาร์ตการ์ด

โครงสร้างการโปรแกรม สมาร์ตการ์ด จะประกอบด้วยส่วนสำคัญสองส่วนคือ คอมพิวเตอร์และส่วนโปรแกรมที่ใช้ไมโครคอนโทรเลอร์เบอร์ P89V51RD2BN ทำหน้าที่ติดต่อกับ สมาร์ตการ์ด และสามารถติดต่อสื่อสารข้อมูลกับคอมพิวเตอร์ได้ทางพอร์ตอนุกรม RS-232 โครงสร้างแสดงดังรูป



รูปที่ 7.2 แสดงโครงสร้างของเครื่องโปรแกรมสมาร์ตการ์ด

7.1.2โครงสร้างของเครื่องโปรแกรมสมาร์ตการ์ด

1. คอมพิวเตอร์ มีหน้าที่ในการรับและบันทึกข้อมูลที่เป็น HEX File แล้วทำการแปลงค่าข้อมูลให้เป็นตัวเลขฐานสิบ และยังทำ การรับส่งข้อมูลกับเครื่องโปรแกรม (P89V51RD2BN) ทางพอร์ดอนุกรม เราสามารถเขียนข้อมูลในสมาร์ตการ์ดโดยผ่านทาง Hyperterminal

2. เครื่องโปรแกรม (P89V51RD2BN) รับคำสั่งที่ส่งมาจากเครื่องคอมพิวเตอร์ซึ่งมีรูปแบบเป็นข้อมูลความยาว 3ไบต์ โครงสร้างของข้อมูลดังกล่าวประกอบด้วยคำสั่ง (Command), แอดเดรส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Address), ข้อมูล (Data) แล้วก็จะทำการส่งข้อมูลให้คอมพิวเตอร์ทำการแสดงผลทาง Hyperterminal โดยส่วนนี้มีความสำคำอย่างมากเนื่องจากเป็นส่วนที่เชื่อมต่อระหว่าง สมาร์ดการ์ด กับคอมพิวเตอร์

7.1.3 หลักการทำงานเครื่องโปรแกรมสมาร์ตการ์ด SLE 4442

ในรูปที่ 7.3 คือวงจรที่สมบูรณ์ของเครื่องโปรแกรมสมาร์ตการ์ด SLE 4442 ซึ่งการทำงานของวงจรนี้ถูกควบคุมโดยไมโครคอนโทรลเลอร์ตระกูล MCS51 เบอร์ P89V51RD2BN หน้าที่หลักโดยทั่วไปของไมโครคอนโทรลเลอร์ตัวนี้คือ ควบคุมการรับส่งข้อมูลกับเครื่องคอมพิวเตอร์ และสามารถจัดการทางด้านข้อมูลในกรณีที่มีการจัดเก็บ หรือบันทึกข้อมูลบางอย่างไว้เป็นฐานข้อมูล กลไกในการที่ P89V51RD2BN จะสื่อสารกับสมาร์ตการ์ดซึ่งในที่นี้คือบัตรสมาร์ตการ์ด SLE 4442 นั้นเกิดจากรูปแบบสัญญาณที่พอร์ต P0.1 – P0.4 (ขาที่ 39-36 ของ IC ตามลำดับ) ซึ่งเป็นไปตามที่ได้กล่าวมาแล้วในบทที่แล้ว

สำหรับการสื่อสารข้อมูลกับสมาร์ตการ์ดนั้น จะมี P89V51RD2BN ควบคุมการป้อนสัญญาณต่าง ๆ ให้กับสมาร์ตการ์ด โดย แต่ละ Port ของ P89V51RD2BN จะถูกนำไปใช้ในหน้าที่ต่าง ๆ กันดังนี้

PORT 0.0 จะทำหน้าที่เป็นขาสัญญาณ อินพุท/เอาท์พุท(I/O) ให้กับสมาร์ตการ์ด

PORT 0.1 จะทำหน้าที่เป็นขาสัญญาณ รีเซต(RST) ให้กับสมาร์ตการ์ด

PORT 0.2 จะทำหน้าที่เป็นสัญญาณ นาฬิกา(CLK) ให้กับสมาร์ตการ์ด

PORT 0.3 ใช้ในการตรวจสอบว่าบัตรถูกเสียบเข้ามาขังช่องรับแล้วหรือไม่ จากสถานะของสวิตช์ที่ถูกซ่อนอยู่ภายในซ็อกเกตรับบัตรสมาร์ตการ์ด

การสื่อสารระหว่าง P89V51RD2BN กับพอร์ตอนุกรม RS-232 ของเครื่องคอมพิวเตอร์ในแง่ของกระบวนการใช้จะถูกจัดการโดยโปรแกรมที่เขียนและบันทึกไว้ใน P89V51RD2BN

ส่วนในแง่ของระดับสัญญาณเครื่องโปรแกรม (P89V51RD2BN) มีหน้าที่สื่อสารกับสมาร์ตการ์ด และคอมพิวเตอร์ซึ่งการสื่อสารกับคอมพิวเตอร์เป็นการสื่อสารแบบ อะซิงโครนัสจึงสามารถใช้พอร์ตอนุกรมของคอมพิวเตอร์ ทำการสื่อสารกับเครื่องโปรแกรมสมาร์ตการ์ด ได้โดยมีไอซี MAX 232 ทำการแปลงระดับ TTL ไปเป็นระดับ RS-232 และจาก RS-232 ไปเป็น TTL (การแปลงให้ระดับแรงดันจากที่ระดับ 0 และ 5 โวลต์ ไปเป็น +12 และ -12 โวลต์) เพื่อช่วยให้คอมพิวเตอร์สามารถเข้าใจกับข้อมูลที่ P89V51RD2BN ส่งไปได้

หน้าที่อีกอย่างของ P89V51RD2BN ก็คือการควบคุมอุปกรณ์ภายนอก อย่างเช่น หลอดแอลอีดี (LED) จะถูกขับโดยตรงจาก P89V51RD2BN โดยมีการพูลอัพกระแสด้วยความต้านทาน 10K (R pack) และจำกัดกระแสที่จ่ายให้แอลอีดีด้วยความต้านทาน 330 Ω

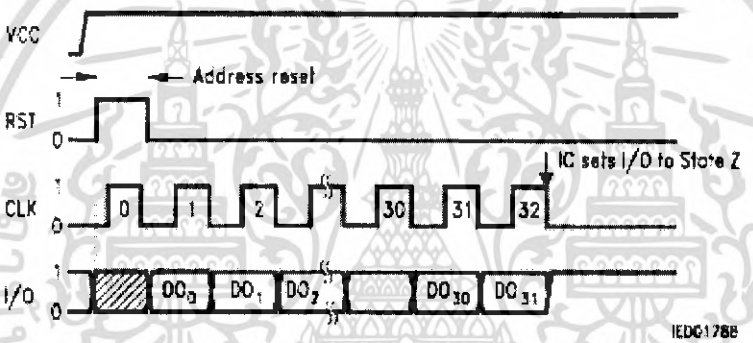
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับในส่วนของแหล่งจ่ายไฟของทั้งวงจรนั้น ได้ใช้วงจรเรกูเลตแรงดันให้มีค่าเท่ากับ 5 V ซึ่งในที่นี้ได้ใช้ไอซีเบอร์ LM2940CT มาช่วยในการทำงานในส่วนนี้

7.1.4 หลักการทำงานของเครื่องโปรแกรมสมาร์ตการ์ด (P89V51RD2BN)

การทำงานของเครื่องโปรแกรมสมาร์ตการ์ดหลังจากจ่ายไฟแล้ว เริ่มจากการกำหนดค่าเริ่มต้นของพอร์ด (Initial Port) และ รีจิสเตอร์ รอจนคอมพิวเตอร์ส่งคำสั่งที่จะทำการ อ่าน/ เขียน สมาร์ตการ์ด จากนั้นเตรียมพร้อมเข้าสู่โหมดโปรแกรม (Initial programming mode) จากวิธีที่กล่าวมาในบทที่ 2 เมื่อสมาร์ตการ์ดพร้อมที่จะอินเตอร์เฟสแล้วให้ส่งสัญญาณขอข้อมูลจากคอมพิวเตอร์

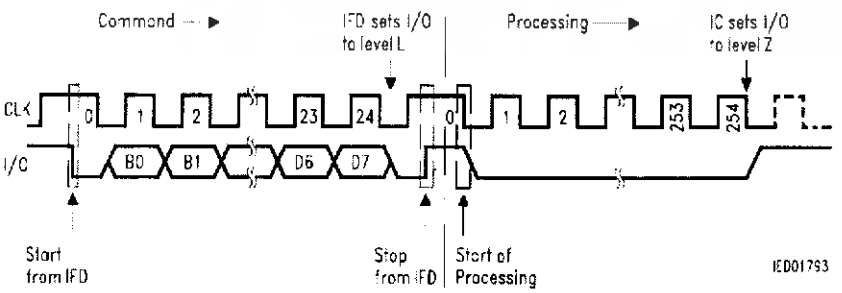
การ Reset and Answer to reset



รูปที่ 7.3 สัญญาณรีเซต และ ATR

ในส่วนของการออกแบบ เราต้องป้อนสัญญาณ Clock ลูกแรกให้อยู่ในสัญญาณ Reset ลูกแรก โดยเมื่อหมดสัญญาณ Reset ข้อมูล ATR จะเริ่มออกมาอย่างต่อเนื่อง โดยมีข้อแม้ว่าเราต้องป้อนสัญญาณ Clock อย่างต่อเนื่อง อีก 32 ลูกจึงจะได้ข้อมูล ATR ของการ์ดออกมาครบ 4ไบต์

การเขียนข้อมูล

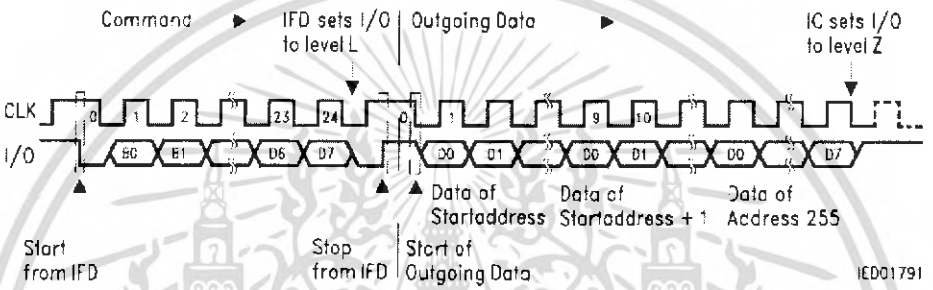


รูปที่ 7.4 สัญญาณของคำสั่ง Read Memory

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะประกอบด้วย รหัสPSC(Programmable security code) , ชุดคำสั่ง(Command) , แอดเดรสเริ่มต้น(Address) , ข้อมูล(Data) โดยเมื่อมีการ Check error counter ในบัตรแล้วว่าบัตรยังไม่ถูก Lock ก็จะนำ PSC ที่เราป้อนไป Check กับPSC ที่อยู่ในบัตร(Address1,2,3) เมื่อ PSC ตรงกับในบัตรก็จะเริ่มทำการเขียน โดยมีขั้นตอนดังนี้ คือเมื่อเครื่องโปรแกรมได้รับ คำสั่ง(Command) , แอดเดรสเริ่มต้น (Address) , ข้อมูล(Data) เครื่องโปรแกรมก็จะนำข้อมูลป้อนเข้าสู่บัตรเริ่มตั้งแต่แอดเดรสเริ่มต้น (Address) ที่เรากำหนดจนครบ Data ที่เราต้องการจะเขียน

การอ่านข้อมูล

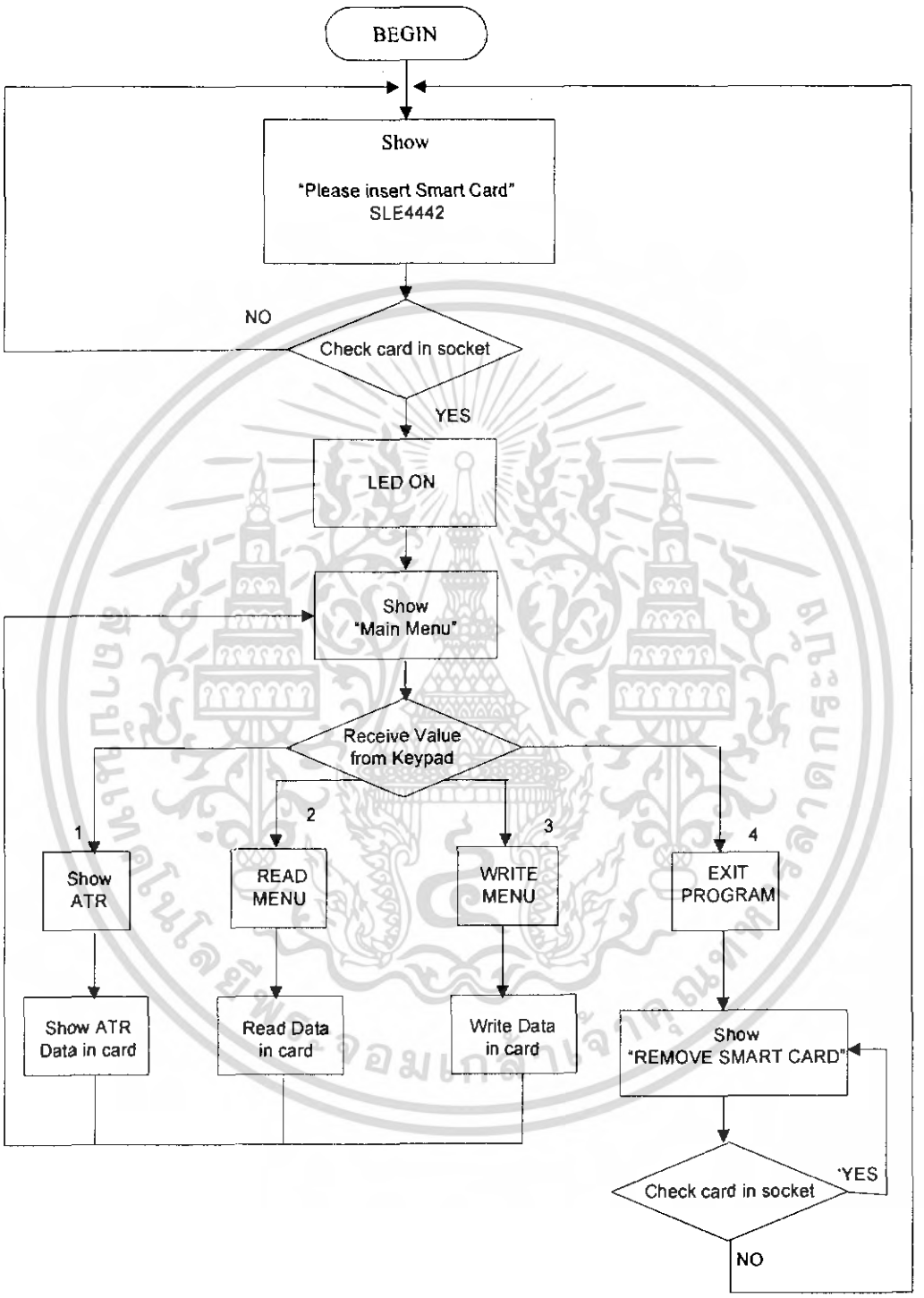


รูปที่ 7.5 สัญญาณของคำสั่ง Read Memory

จะประกอบด้วย ชุดคำสั่ง(Command) , แอดเดรสเริ่มต้น(Address) , จำนวน ไบต์ ที่ต้องการอ่าน โดยมีขั้นตอนดังนี้ คือเมื่อเครื่องโปรแกรมได้รับ start condition , คำสั่ง(Command) , แอดเดรสเริ่มต้น(Address) , จำนวน Byte ที่เราต้องการอ่าน , stop condition เครื่องโปรแกรมก็จะเริ่มนำข้อมูลในบัตรเริ่มตั้งแต่แอดเดรสเริ่มต้น(Address) โดยจะทำการอ่านทีละบิต จนครบ 8 บิต(8บิต=1ไบต์) ก็จะนำข้อมูลที่ได้ไปเก็บในรีจีสเตอร์ ที่กำหนด ทำเช่นนี้ไปเรื่อย ๆ จนครบจำนวน ไบต์ ที่เราต้องการอ่าน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

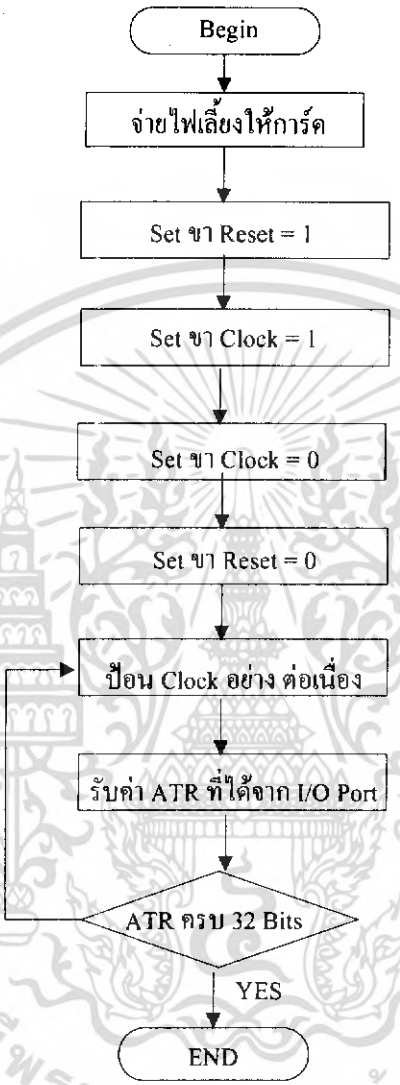
Chart การทำงาน เมื่อเปิดโปรแกรม Hyper Terminal



รูปที่ 7.6 Flow Chart การทำงาน เมื่อเปิดโปรแกรม Hyper Terminal

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

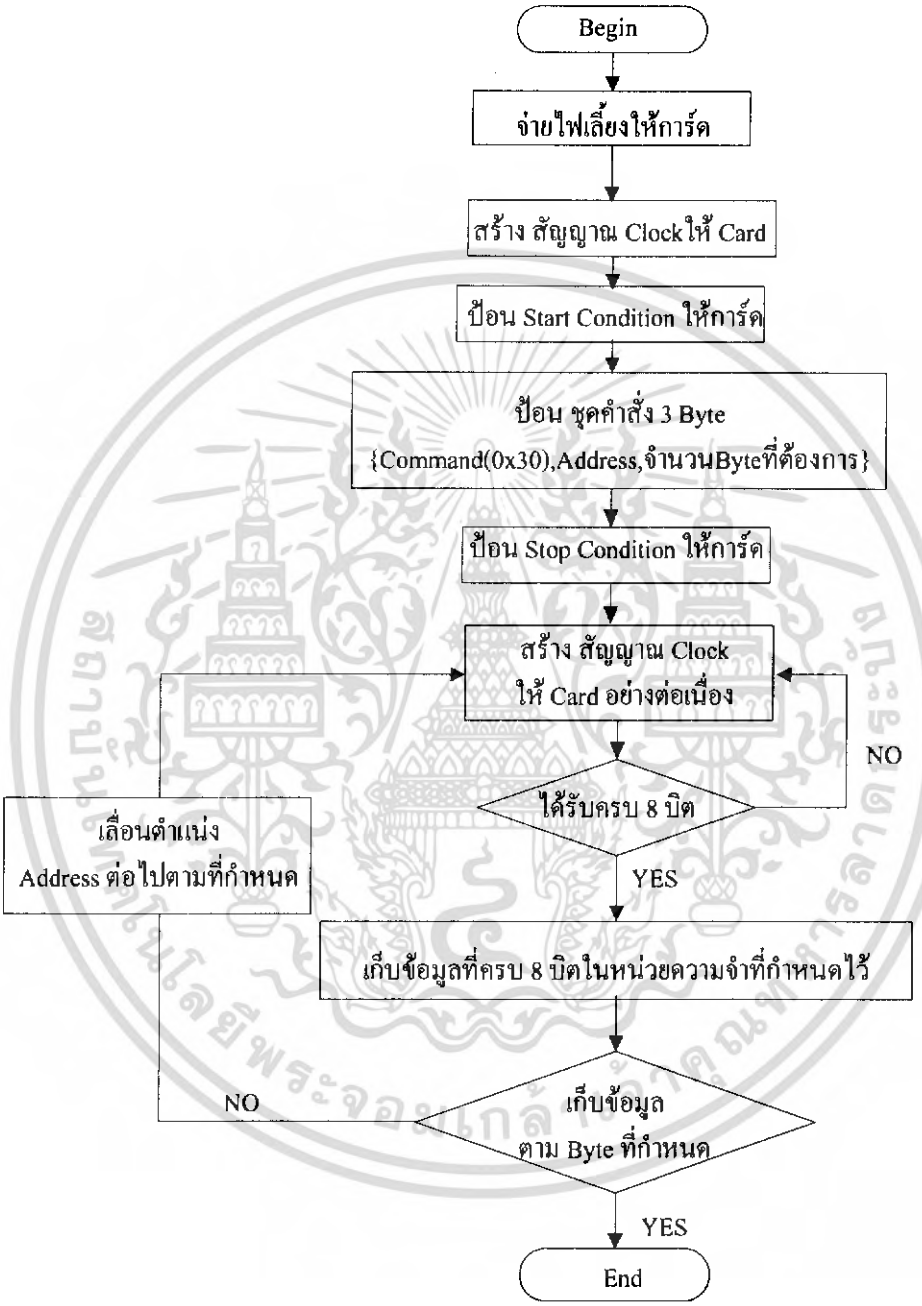
Flow chart การ Reset and Answer to reset บัตร SLE 4442



รูปที่ 7.7 Flowchart Reset and Answer to reset บัตร SLE 4442

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

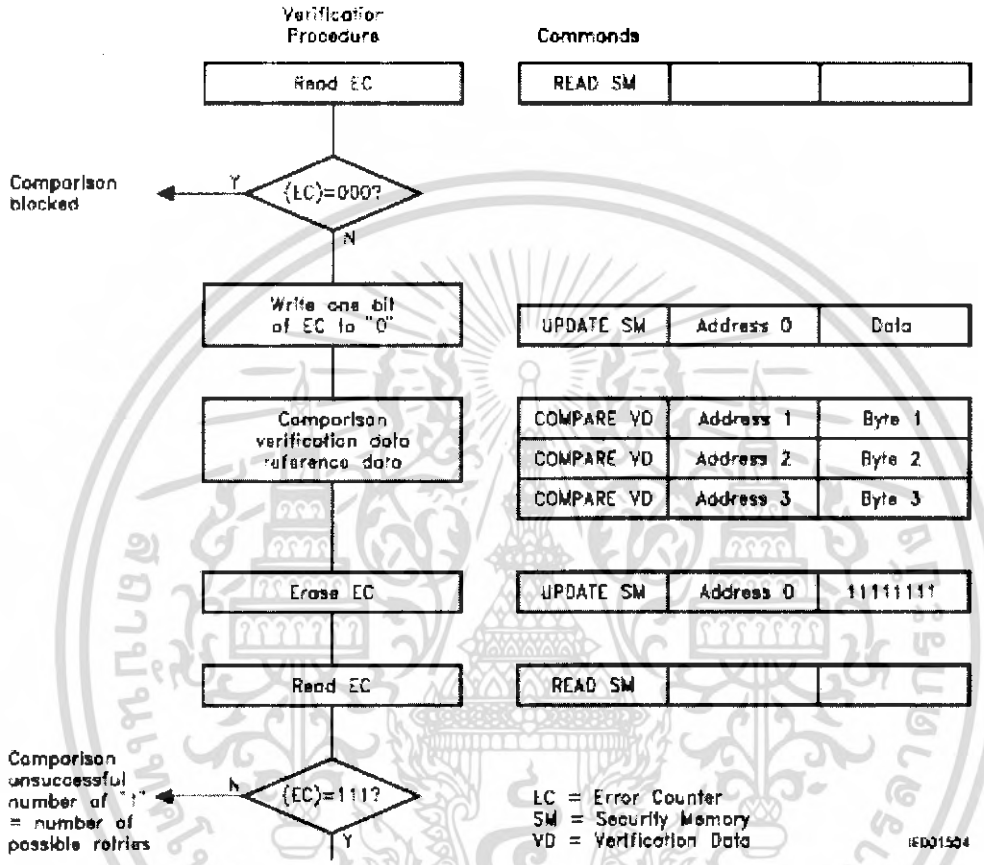
Flow chart การ Read Memoryบัตร SLE 4442



รูปที่ 7.8 Flow chart โปรแกรม READ บัตรSLE 4442

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

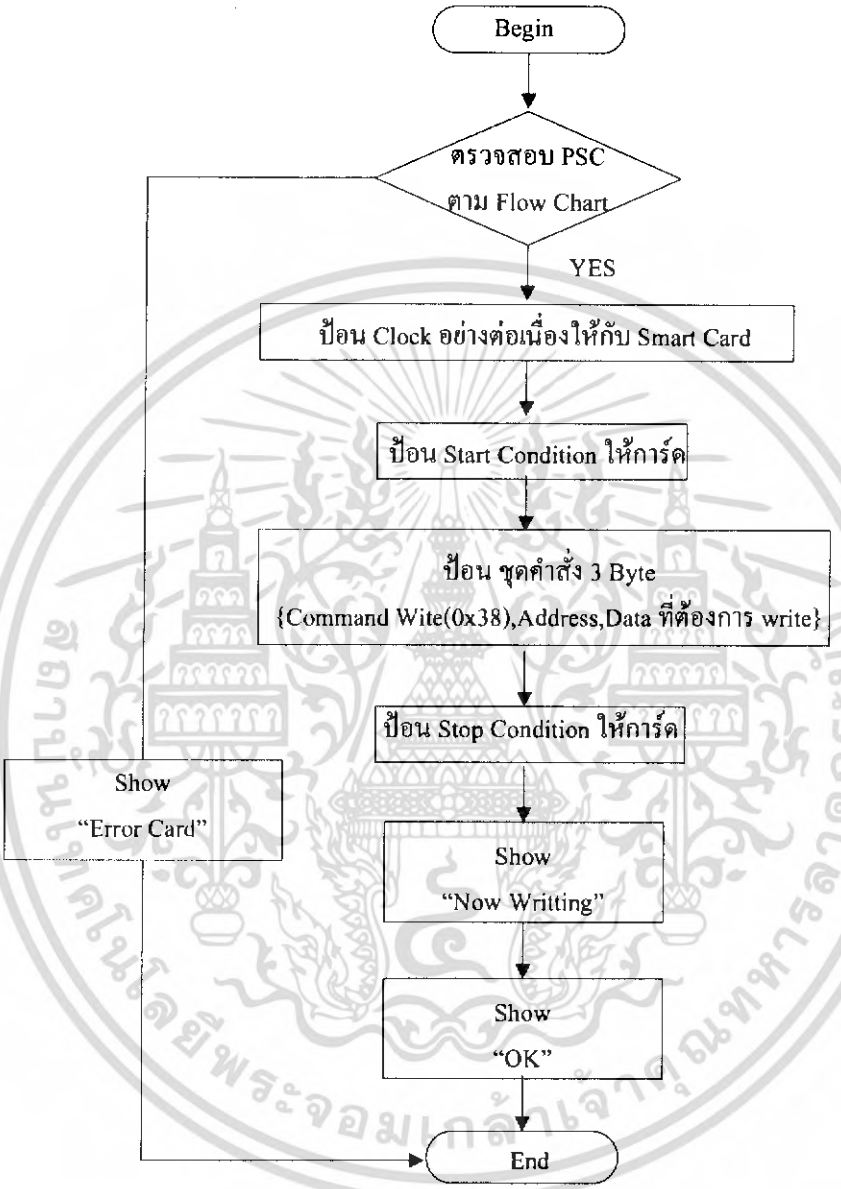
Flow chart การตรวจสอบ Programmable Security Code(PSC)



รูปที่ 7.9 Flow chart การตรวจสอบ Programmable Security Code(PSC)

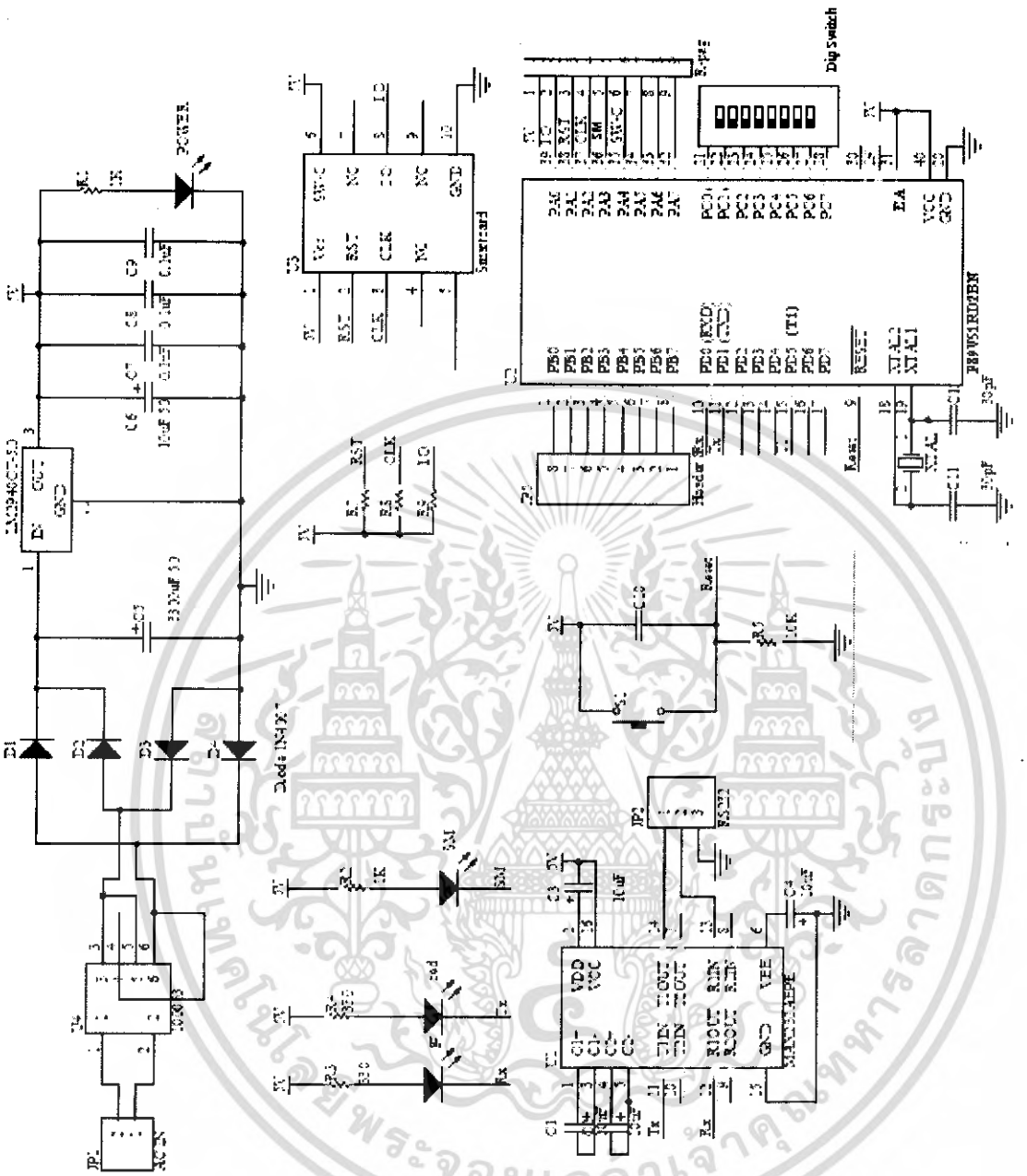
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Flow chart การ Update Main Memory (Write) บัตร SLE 4442



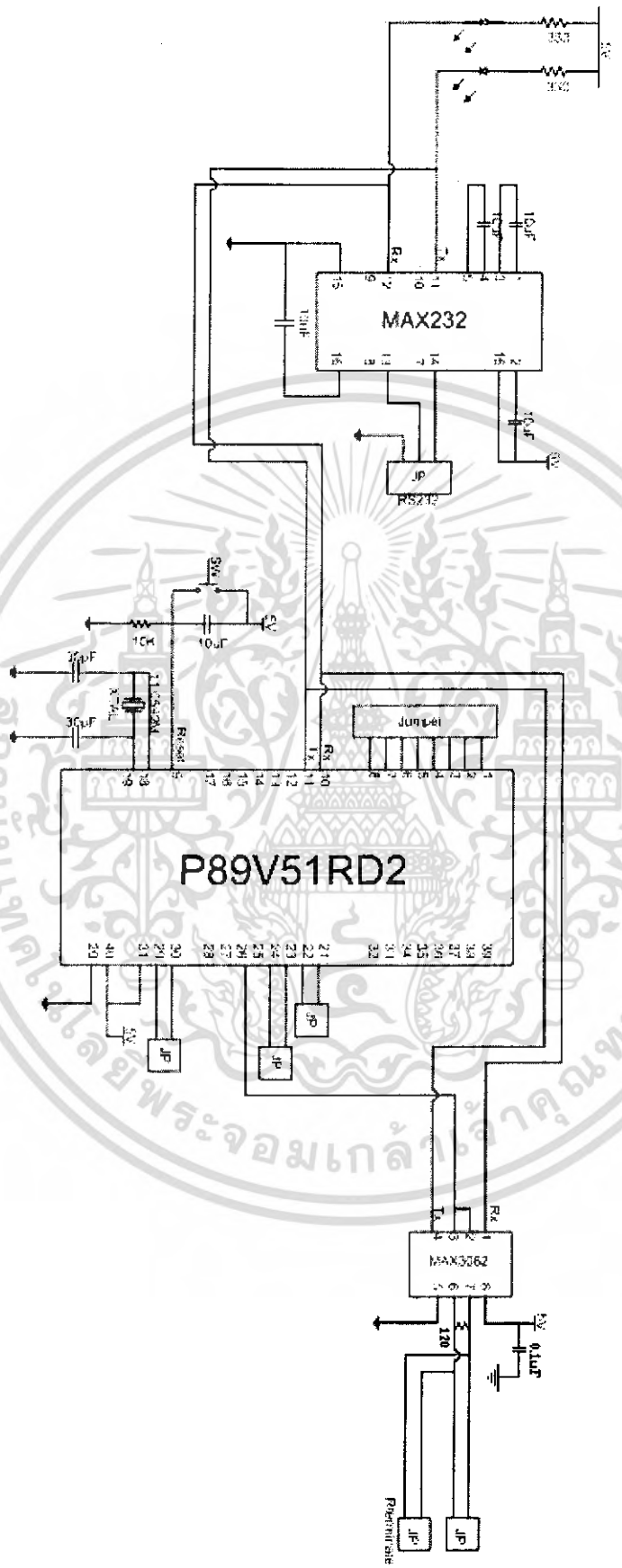
รูปที่ 7.10 Flow chart การ Update Main Memory (Write) บัตร SLE 4442

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.11 วงจรเครื่องโปรแกรมสมาร์ตการ์ด SLE 4442

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

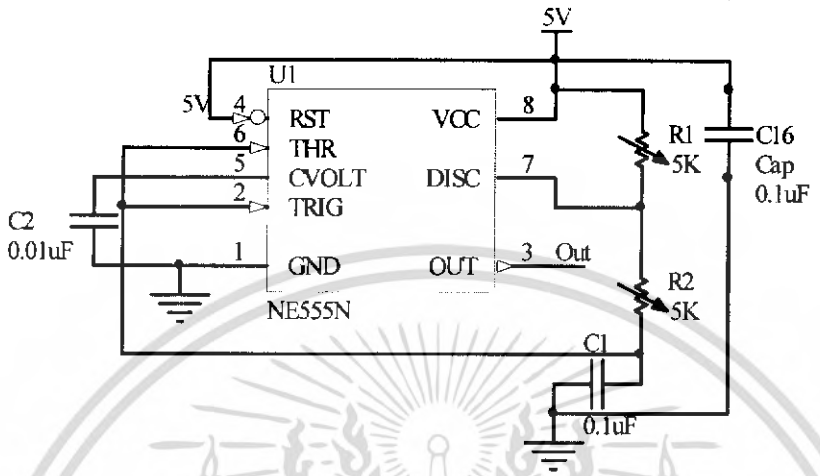


รูปที่ 7.12 วงจร Master

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

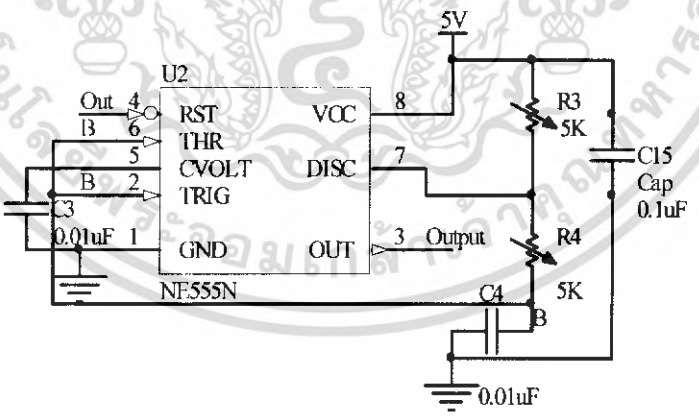
7.2 องค์ประกอบของระบบโดยรวมของเครื่องอ่าน เขียน สมาร์ตการ์ด

7.2.1 ชุดวงจรเซ็นเซอร์



รูปที่ 7.13 วงจร NE555N ชุดที่ 1

วงจร 555 ทำหน้าที่สร้าง Pulse ที่มีความถี่ 1 kHz โดยนำสัญญาณ Pulse ความถี่ 1kHz ที่ได้จากขาที่ 3 ของ 555 ชุดแรก ไปต่อกับ ขาที่ 4 ของ 555 ชุดที่สอง เพื่อทำการ Modulate



รูปที่ 7.14 วงจร NE555N ชุดที่ 2

รวงจร 555 ชุดนี้ทำหน้าที่สร้าง Pulse Carrier ที่มีความถี่ 38 kHz และเมื่อเรานำสัญญาณที่ได้จาก วงจร 555 ชุดแรก มา Modulate กับสัญญาณ ของวงจร 555 ชุดที่ 2 จะได้สัญญาณที่ไปขับอินฟาเรดตัวส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

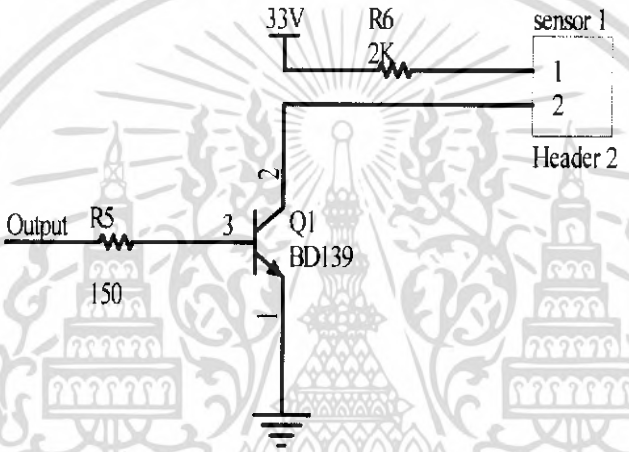
โดยความถี่ทั้งสองที่นำมา Modulate กัน หาได้จากสมการดังนี้

The frequency of oscillation is :

$$f = \frac{1}{T} = \frac{1.44}{(R_A + 2R_B)C}$$

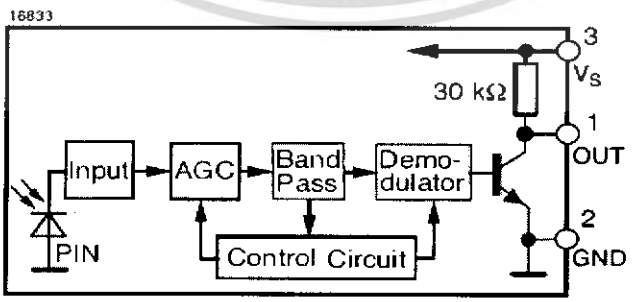
The duty cycle is :

$$D = \frac{R_B}{R_A + 2R_B}$$



รูปที่ 7.15 ชุดตัวขับ อินฟาเรด (BD 139)

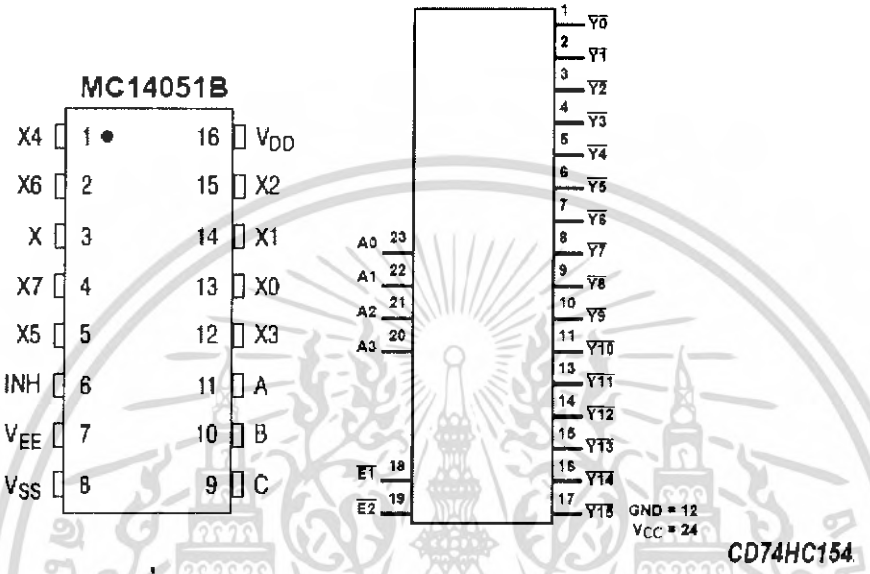
BD 139 เป็น ทรานซิสเตอร์ชนิดNPN ทำหน้าที่ขับกระแสของตัวส่งอินฟาเรดให้ส่งสัญญาณ โดยนำสัญญาณที่ได้จากการ Modulate ของวงจร 555 ทั้ง 2 ชุดมาเพื่อทำการขับกระแสของทรานซิสเตอร์



รูปที่ 7.16 Block Diagram ของ IR Receiver Modules

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

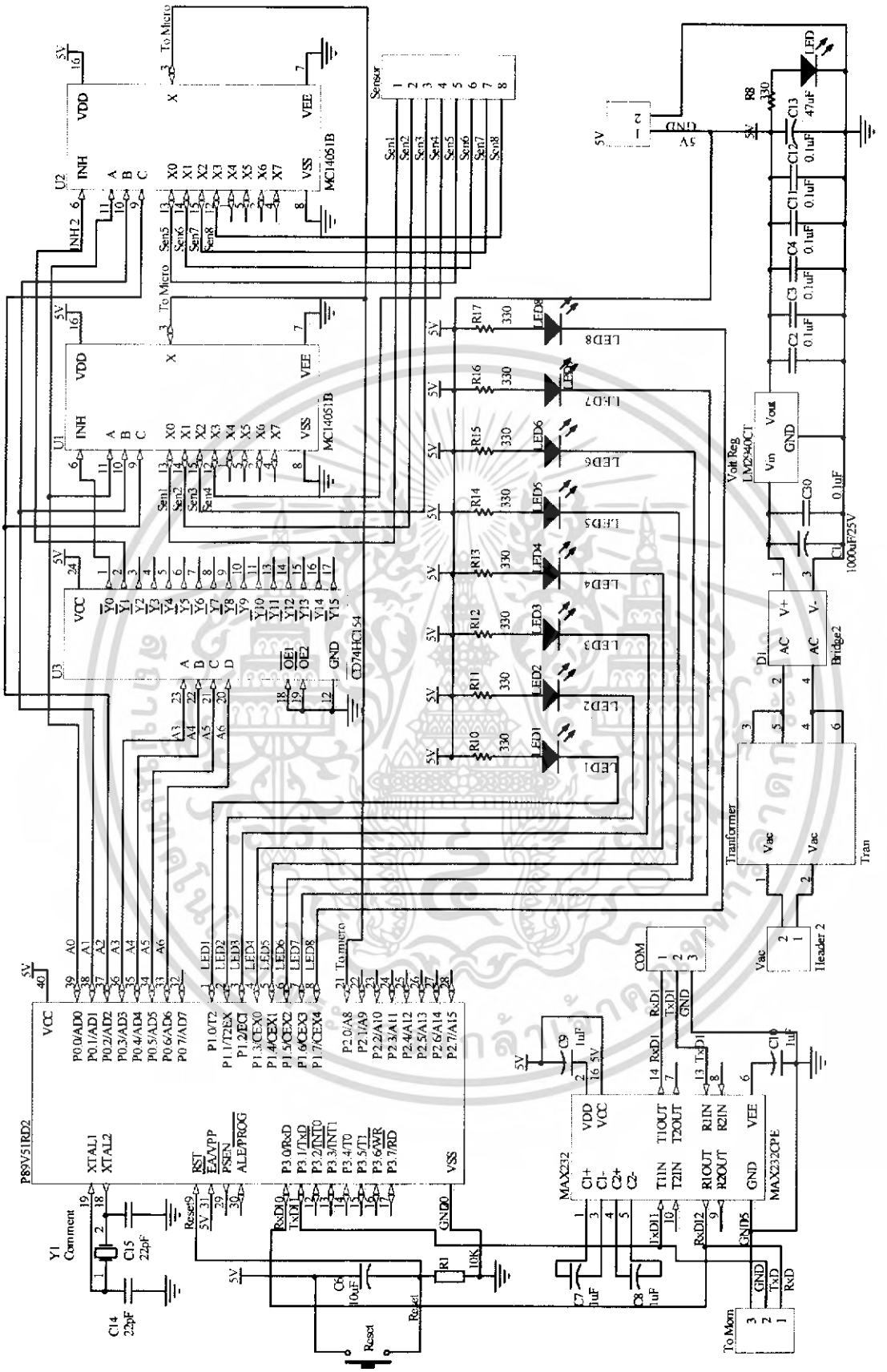
IR Receiver Modules เบอร์ TSOP 34836 เป็นตัวรับสัญญาณอินฟาเรด จากตัวส่ง และ Demodulate สัญญาณความถี่ 1 kHz



รูปที่ 7.17 ไอซี ของ Multiplexers และ Demultiplexers

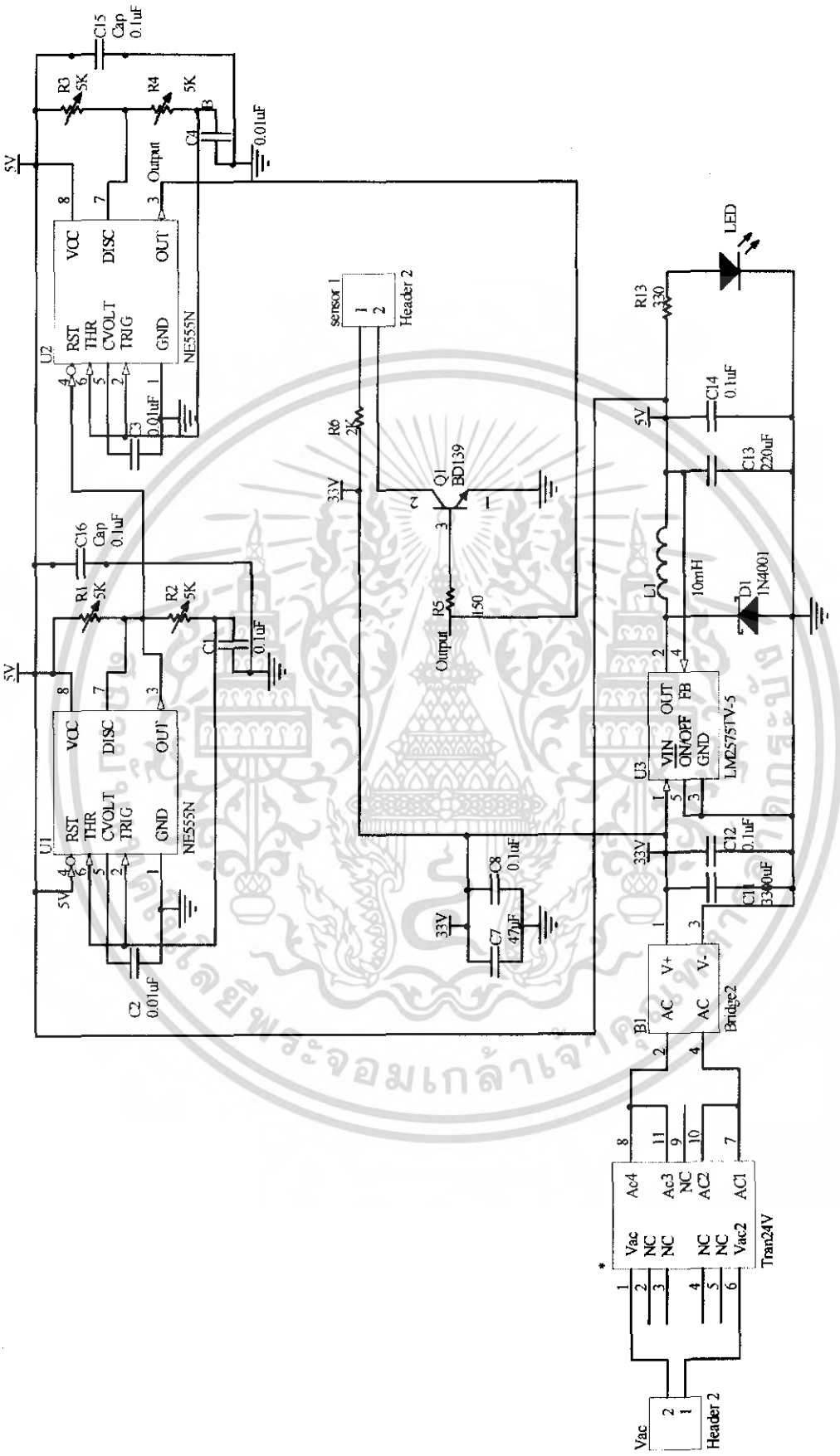
การต่อ Multiplexers และ Demultiplexers เพื่อทำให้สามารถเพิ่มจำนวนเซ็นเซอร์ได้ ในโครงงานนี้สามารถเพิ่มได้ถึง 128 ตัวและ ยังสามารถเพิ่มได้อีกโดยการรับค่าจากเซ็นเซอร์ใช้ Microcontroller ทำหน้าที่ สแกนและรับค่าจากเซ็นเซอร์แล้วส่งค่าที่ได้ให้ Microcontroller ตัวหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.18 วงจรตัวรับอินฟาเรด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.19 วงจรตัวส่งอินฟาเรด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.3 หลักการทำงานโครงงาน

การใช้งานจะประกอบไปด้วย 3 ขั้นตอนสำคัญคือ การลงทะเบียนผู้ใช้ การเข้าใช้งานระบบ และการเติมเงิน

คุณสมบัติของเครื่อง

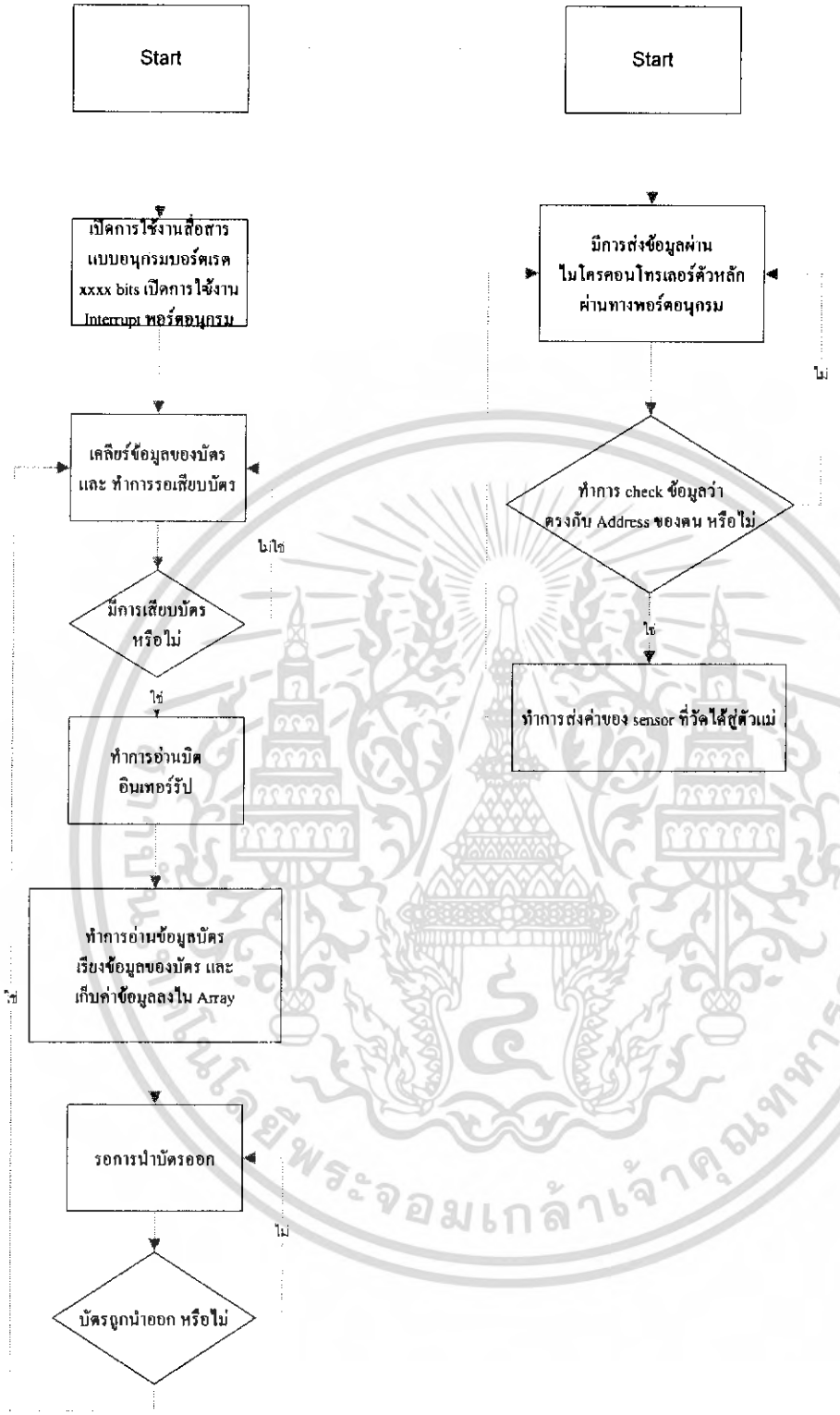
1. รองรับ 128 คัน
2. เก็บฐานข้อมูลและรหัสผ่านในคอมพิวเตอร์
3. บันทึกข้อมูลผู้ใช้ ลงในคอมพิวเตอร์และเรียกดูข้อมูลย้อนหลังได้
   - ID วัน เวลา
4. การเพิ่มหรือลบ ฐานข้อมูลต้องเป็นผู้ดูแลระบบเท่านั้น
   - มี User และ Password เพื่อแก้ไข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โฟลชาร์ตแสดงการทำงานของเครื่องต่าง ๆ ที่ใช้ในระบบ

รูปที่ 7.20 Flow chart การทำงานของตัวแม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

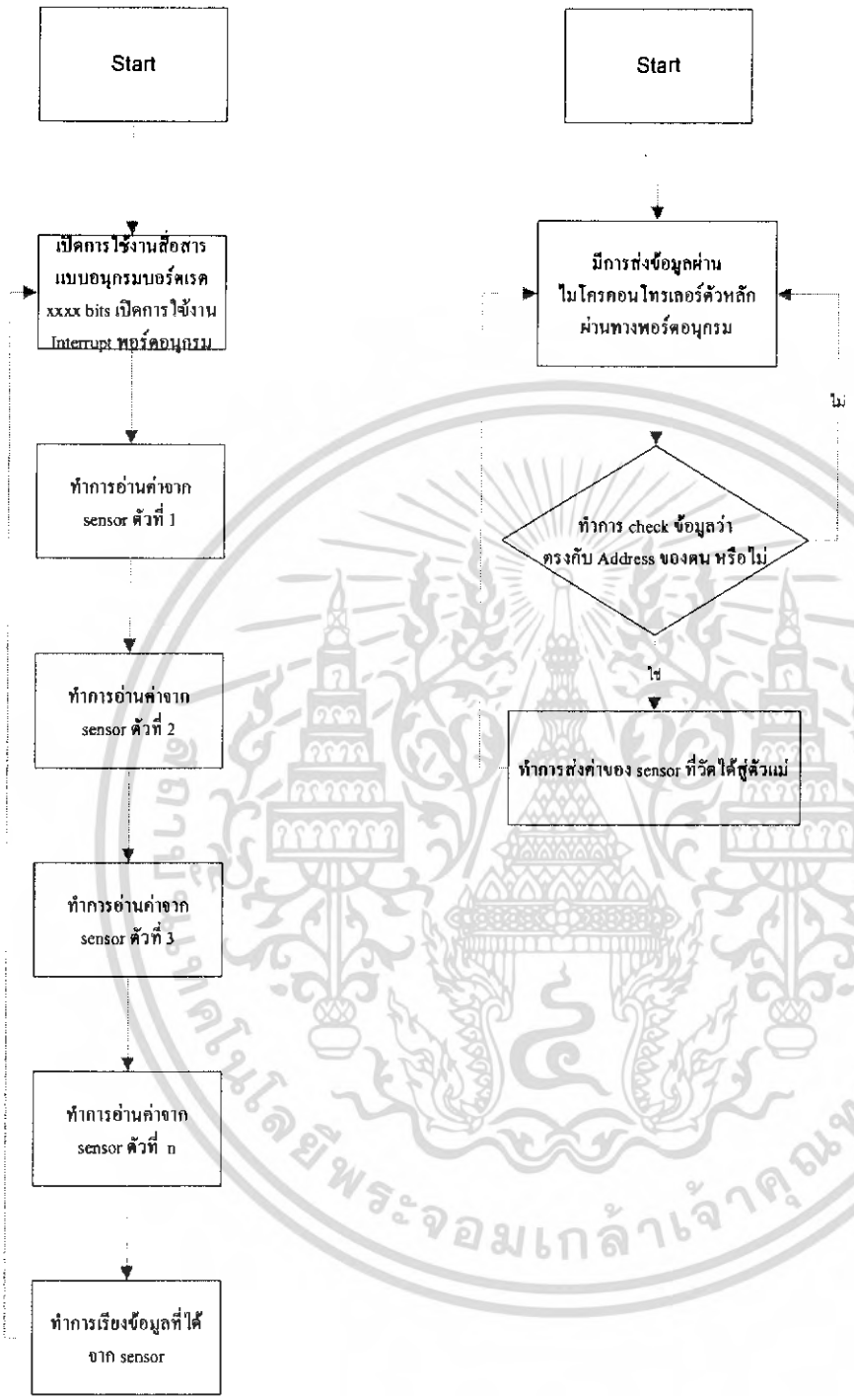


การทำงานของเครื่องอ่านสมาร์ตการ์ดปกติ

การทำงานของเครื่องอ่านสมาร์ตการ์ดขณะมีการInterupt

รูปที่ 7.21 Flow chart การทำงานของเครื่องอ่านสมาร์ตการ์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



การทำงานของเครื่องตรวจสอบสิ่งกีดขวาง

การทำงานของเครื่องตรวจสอบสิ่งกีดขวางขณะมีการInterupt

รูปที่ 7.22 Flow chart การทำงานของเครื่องตรวจสอบสิ่งกีดขวาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การอินเทอร์เฟสระหว่างข้อมูลที่ได้จากบอร์ดตัวมาสเตอร์กับวิชวลเบสิก

หลักจากได้ข้อมูลของตัวลูกทั้งสาม คือ ที่เสียบบัตรขาเข้า ที่เสียบบัตรขาออก เซ็นเซอร์ ตัวแม่ จะทำการส่งข้อมูลที่ได้มาที่คอมพิวเตอร์เพื่อที่คอมพิวเตอร์จะทำการประมวลผลข้อมูลที่ได้ ผ่านทาง วิชวลเบสิก และเก็บข้อมูลลง Data Base ต่อไป โดยข้อมูลที่ได้จะมีทั้งหมด 13 Byte ดังนี้

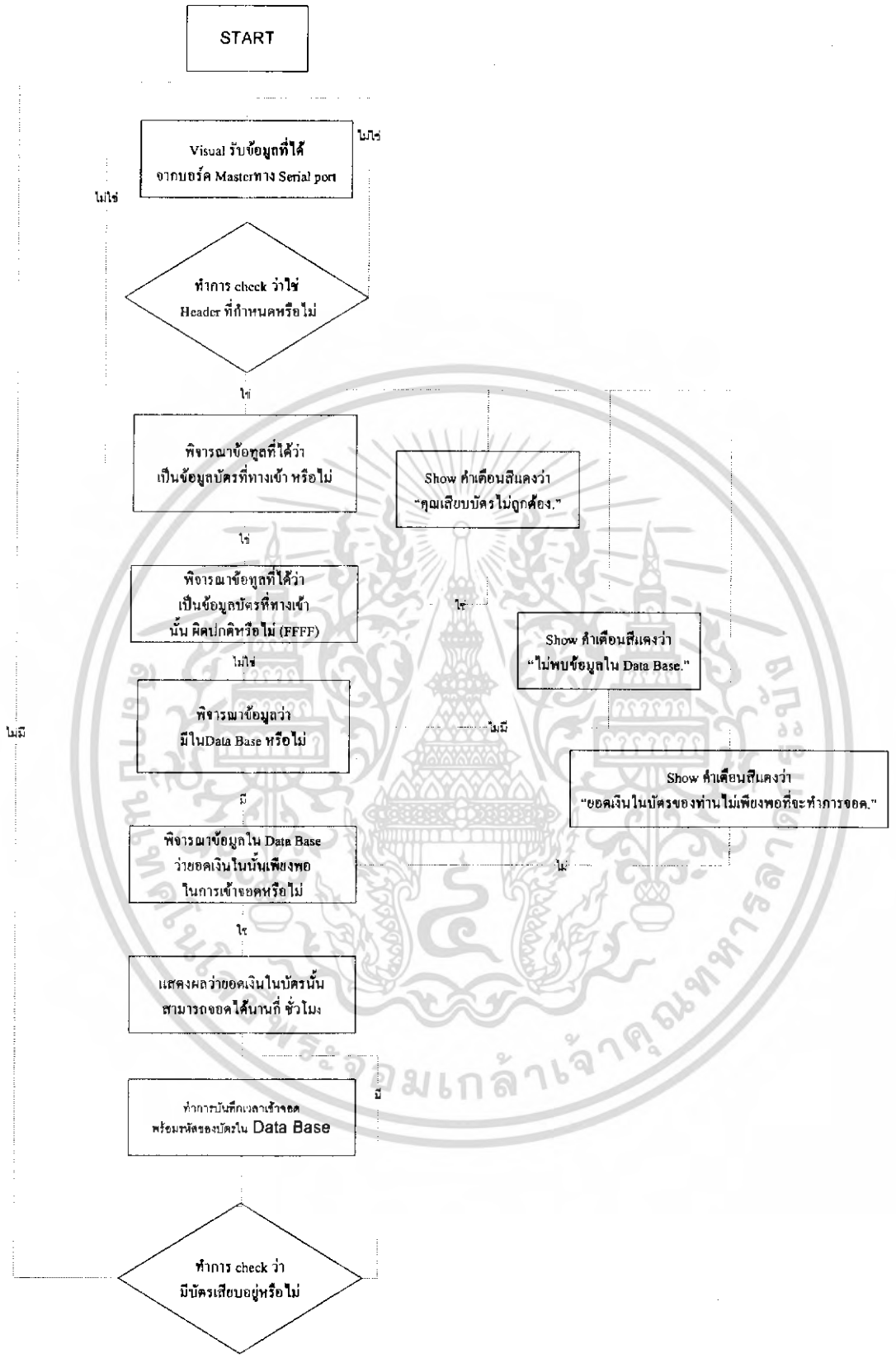
X X X / I I I I / O O O O / S / Z

- X X X แรกเป็น Header ที่ตัว Master ส่งมาเพื่อให้ทราบว่าส่วนที่ตามมาจะเป็นข้อมูล โดยที่ส่วนนี้เราจะไม่นำข้อมูลไปประมวลผล
- I I I I ส่วนต่อมาเป็นส่วน ข้อมูลที่ได้จากที่เสียบบัตรที่ทางเข้า โดยจะเป็นข้อมูลรหัสบัตร จำนวนทั้งหมดทั้งหมด 4 Byte โดยเราจะนำข้อมูลส่วนนี้ไปประมวลผลในส่วนของ การเข้าจอด
- O O O O ส่วนต่อมาเป็นส่วน ข้อมูลที่ได้จากที่เสียบบัตรที่ทางออก โดยจะเป็นข้อมูลรหัสบัตร จำนวนทั้งหมดทั้งหมด 4 Byte โดยเราจะนำข้อมูลส่วนนี้ไปประมวลผลในส่วนของ การนำรถออกจากที่จอด
- S ส่วนต่อมาเป็นส่วนข้อมูลของ Sensor จะเป็นการบอกตำแหน่งว่าขณะนี้ตำแหน่งใดบ้างเป็นที่ว่าง และตำแหน่งใดที่ไม่ว่างโดยข้อมูลที่ได้นั้นต้องนำไปใช้ในการแสดงผลการระบุตำแหน่งที่จอด ข้อมูลที่ได้มีขนาด 1 Byte
- Z ส่วนสุดท้ายเป็นส่วนของ Byte ปิดท้าย โดยที่ตัว Master ส่งมาเพื่อให้ทราบว่าเป็น Byte สิ้นสุดของการส่งข้อมูล โดยที่ส่วนนี้เราจะไม่นำข้อมูลไปประมวลผล

ข้อมูลจำนวน 13 Byte นี้จะถูกควบคุมให้ส่งอย่างต่อเนื่องโดยตัวแม่ เพราะฉะนั้นข้อมูลที่แสดงผลไม่ว่าจะเป็นการแสดงผลของหน้าจอ Sensor การแสดง ID ขณะมีรถเข้า ออก จะเป็นลักษณะของ Real Time

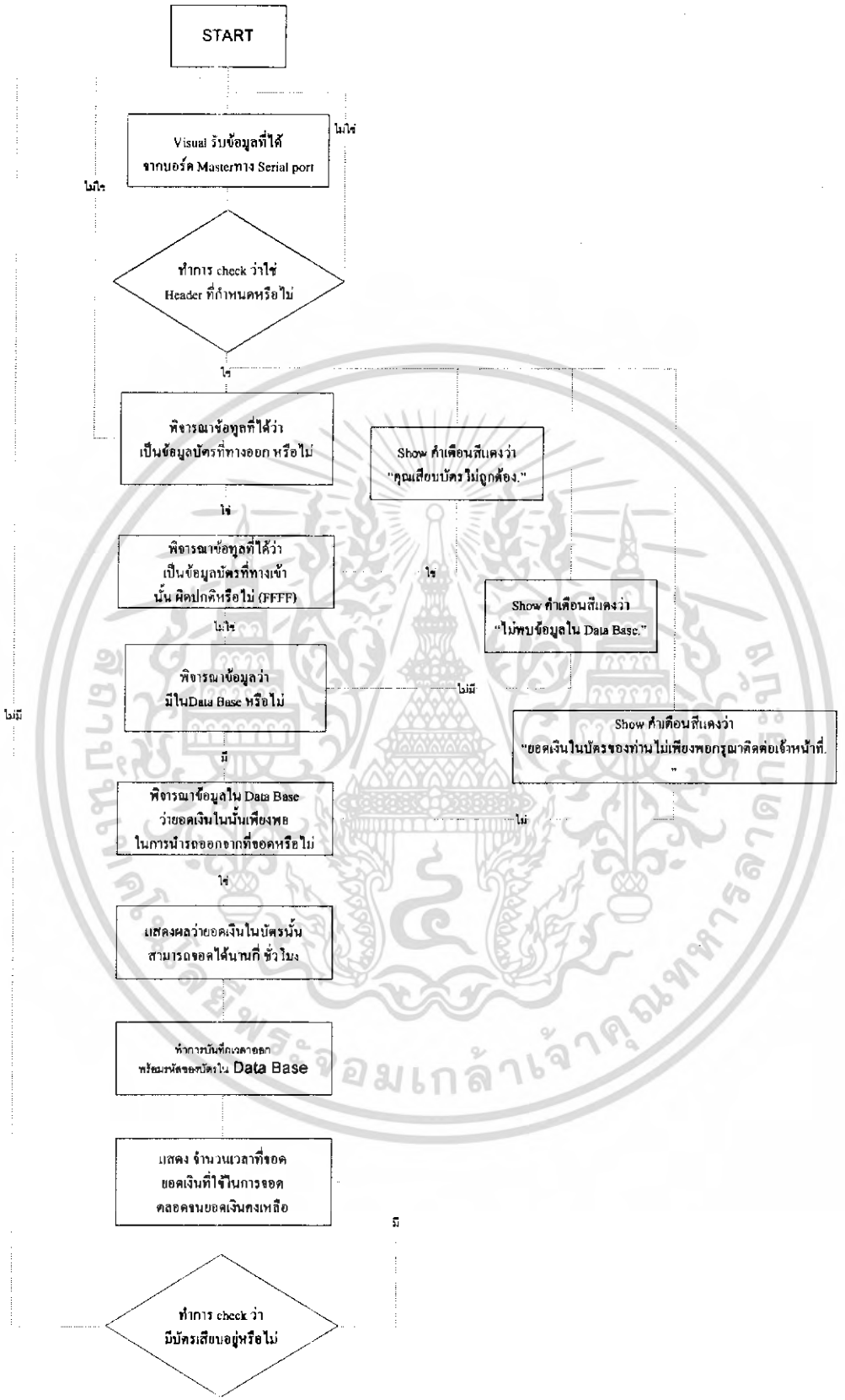
โดยมีหลักการทำงานเป็น Flow chart การทำงานดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.23 Flow chart การทำงานเมื่อทำการเสียบบัตรเมื่อเข้าจอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.24 Flow chart การทำงานเมื่อทำการเสียบบัตรเมื่อต้องการนำรถออกจากที่จอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 8 การทดลองและผลการทดลอง

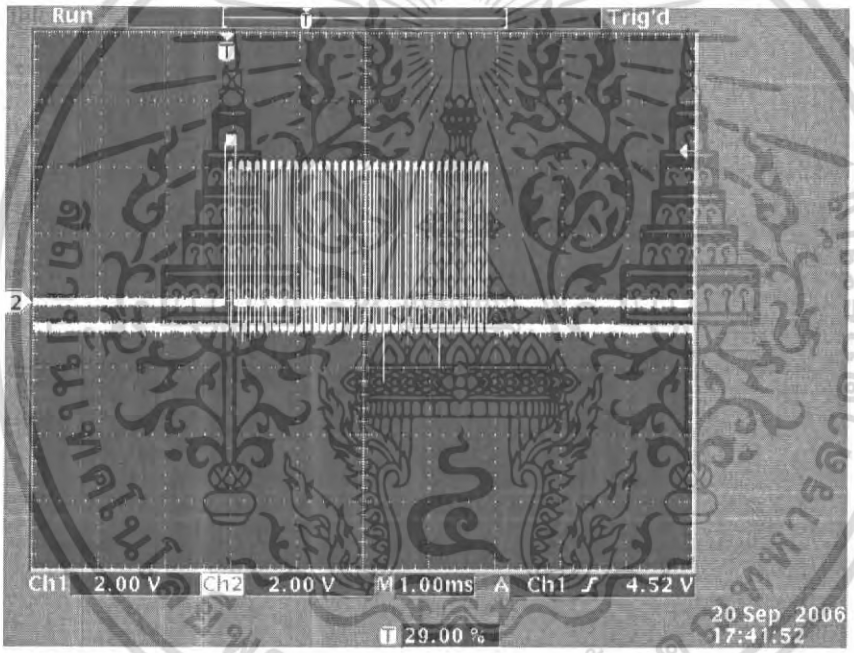
8.1 การทดลอง

- ความเร็วในการรับข้อมูล 9600 บิต/วินาที ผ่านพอร์ตคอม 1
- ทำการทดลองโดยทำการรับข้อมูลจากบัตรสมาร์ตการ์ดโดยการเติมจำนวนเงินลงในบัตร

8.2 การทดลองวัดรูปสัญญาณ ณ ตำแหน่งขาต่างๆในสมาร์ตการ์ด

- ทำการวัดสัญญาณที่ขาต่างๆ ของสมาร์ตการ์ด

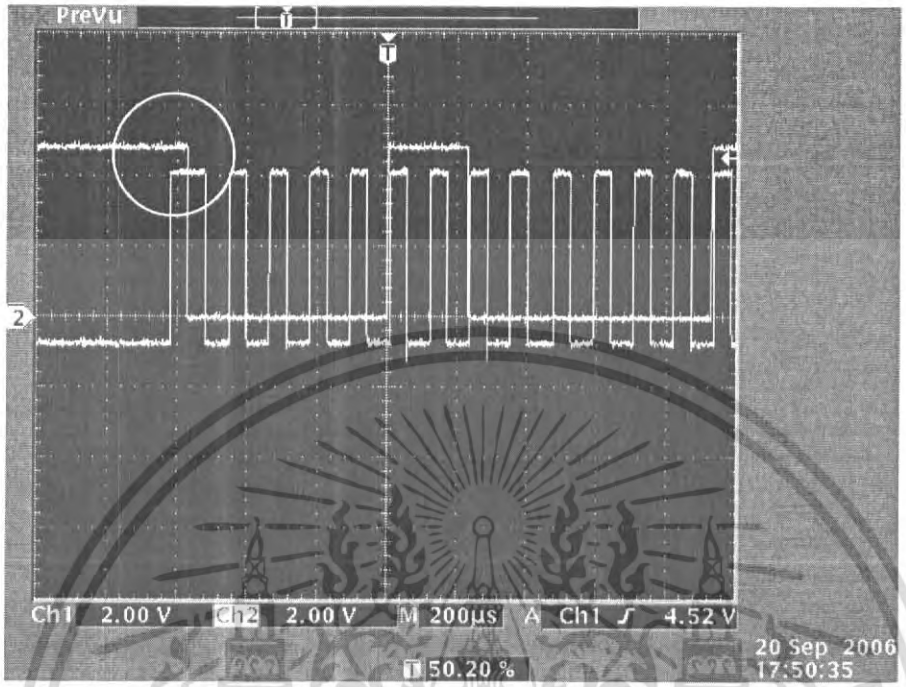
- สัญญาณที่ATR



รูป 8.1สัญญาณที่ ATR

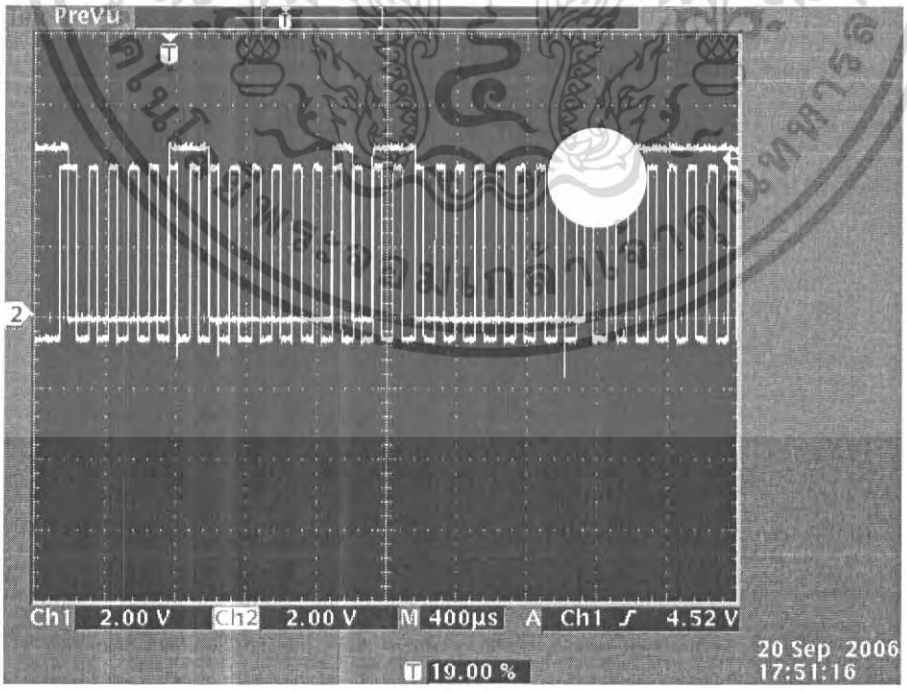
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สัญญาณตอนเริ่มคำสั่ง



รูป 8.2สัญญาณตอนเริ่มคำสั่ง

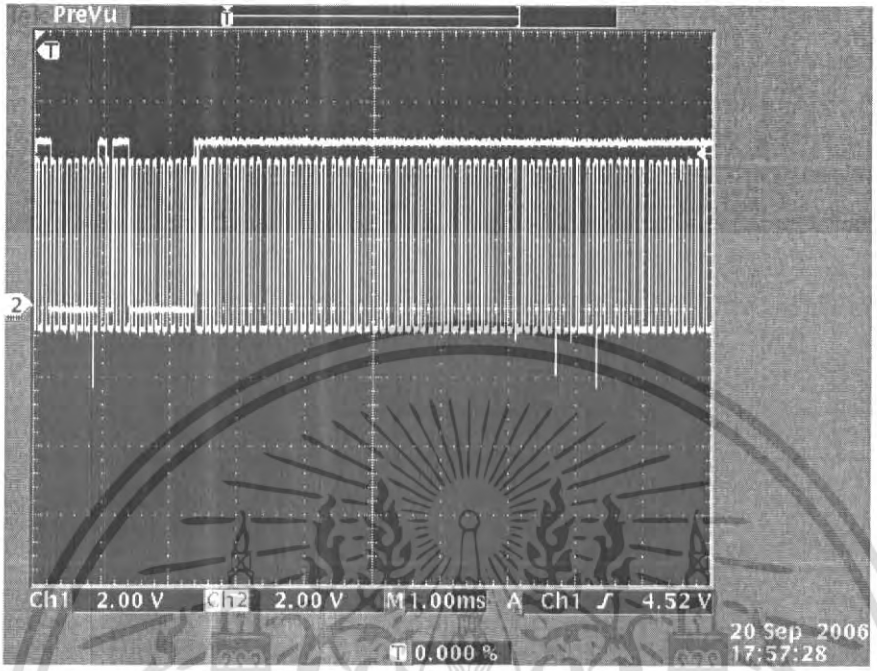
- สัญญาณตอนเริ่มหยุดคำสั่ง



รูป 8.3สัญญาณตอนหยุดคำสั่ง

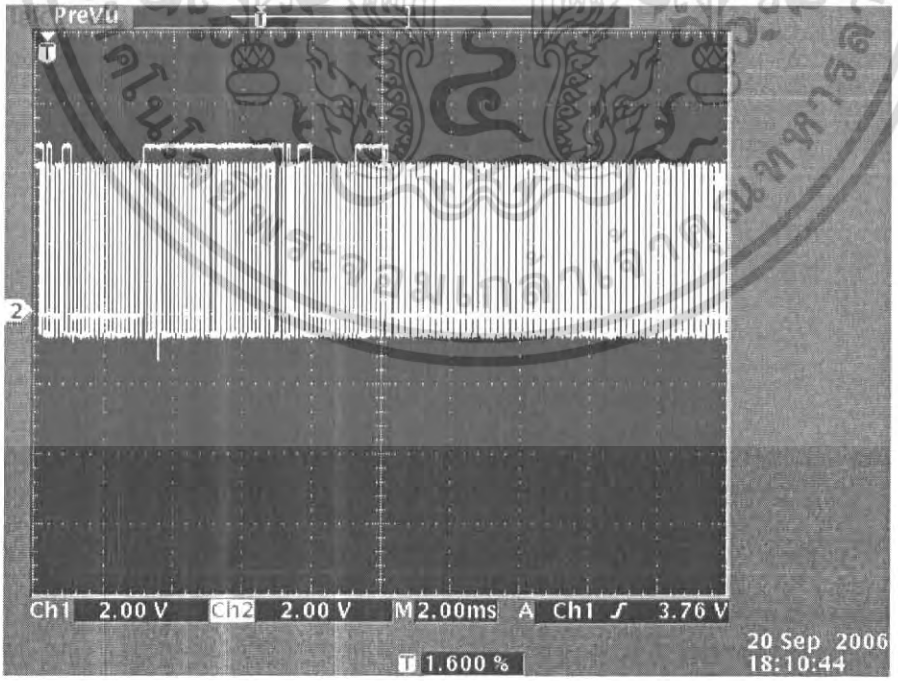
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สัญญาณตอนอ่าน



รูป 8.4 สัญญาณขณะRead

- สัญญาณตอนเขียน



รูป 8.5 สัญญาณช่วง Write

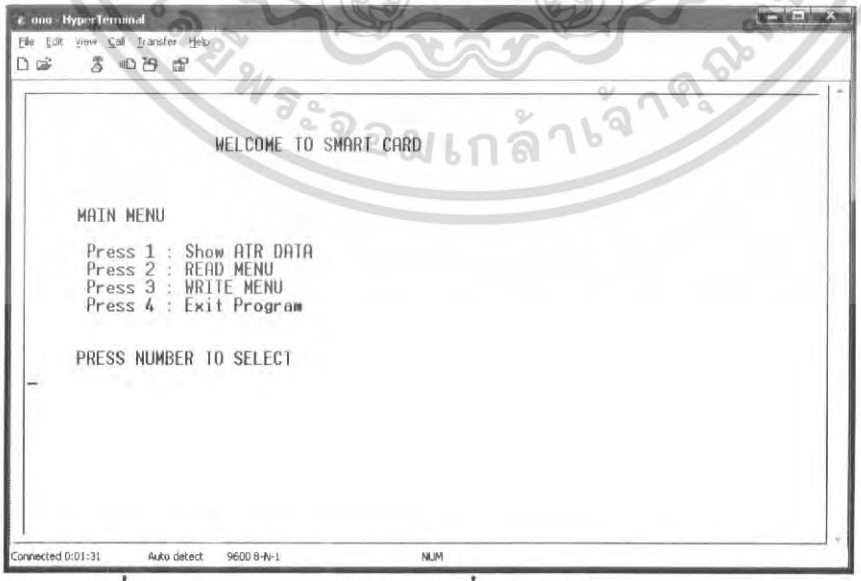
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.3 การทดลองการสั่งการของเครื่องอ่านเขียนสมาร์ตการ์ดผ่าน Hyperterminal

การทดลองกระบวนการติดต่อบัตรสมาร์ตการ์ดเบอร์ SLE4442 โดยจะใช้โปรแกรมไฮเปอร์เทอร์มินอลในการติดต่อ โดยเมื่อเริ่มต้นทำงาน บนหน้าจอคอมพิวเตอร์จะปรากฏหน้าต่างดังรูปที่ 6.6

รูปที่ 8.6 ผลการทดลองหน้าจอแรกของโปรแกรม เพื่อเตรียมรับบัตรสมาร์ตการ์ด

เมื่อใส่บัตรสมาร์ตการ์ดในซอกเกต จะเข้าสู่หน้าจอ Main menu เพื่อเลือก menu ที่จะติดต่อกับบัตรสมาร์ตการ์ด



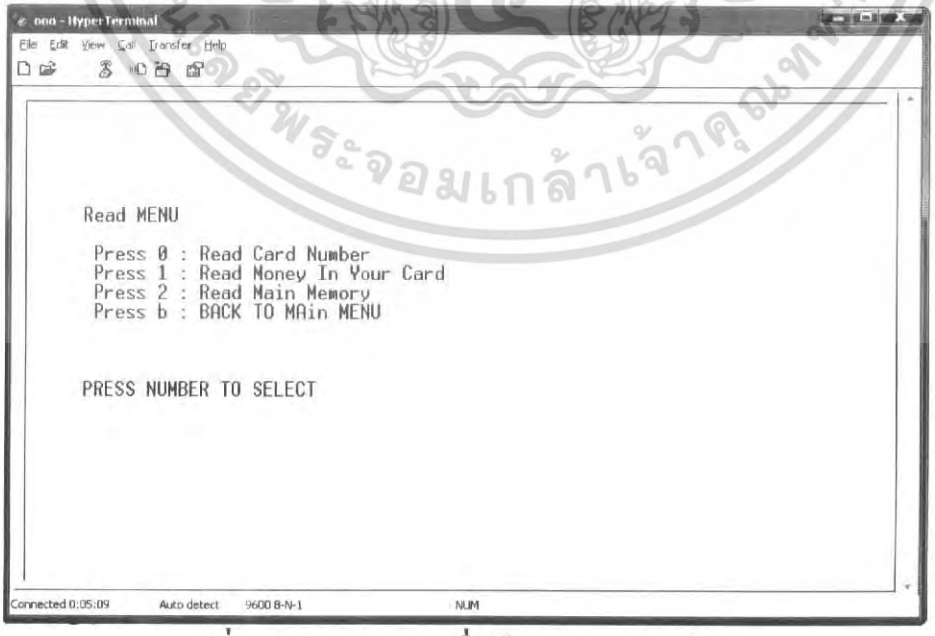
รูปที่ 8.7 ผลการทดลองโปรแกรมเมื่อมีการใส่บัตรสมาร์ตการ์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นทำการทดลองเมื่อกด pad 1 คือการ reset สมาร์ตการ์ดแล้วจะปรากฏหน้าจอแสดง รหัส ATR ดังรูปที่ 8.8

รูปที่ 8.8 ผลการทดลองเมื่อต้องการ reset smart card

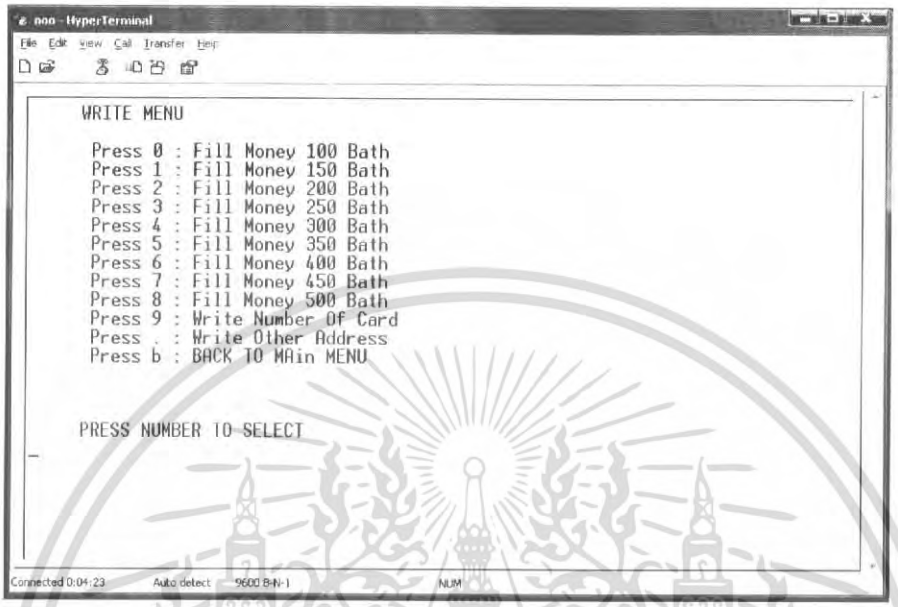
จากนั้นทำการทดลองเมื่อกด pad 2 จากหน้าจอ main menu คือการเข้าสู่โหมดการอ่านข้อมูลสมาร์ตการ์ดแล้วจะปรากฏเมนูย่อย ดังรูปที่ 8.8



รูปที่ 8.9 ผลการทดลองเมื่อเลือกรายการอ่านข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นทำการทดลองเมื่อกด pad 3 จากหน้าจอ main menu คือการเข้าสู่โหมดการเขียนข้อมูลที่เราต้องการลงในสมาร์ตการ์ด จะปรากฏเมนูย่อย ดังรูปที่ 8.8



รูปที่ 8.10 ผลการทดลองเมื่อเลือกรายการเขียนข้อมูล

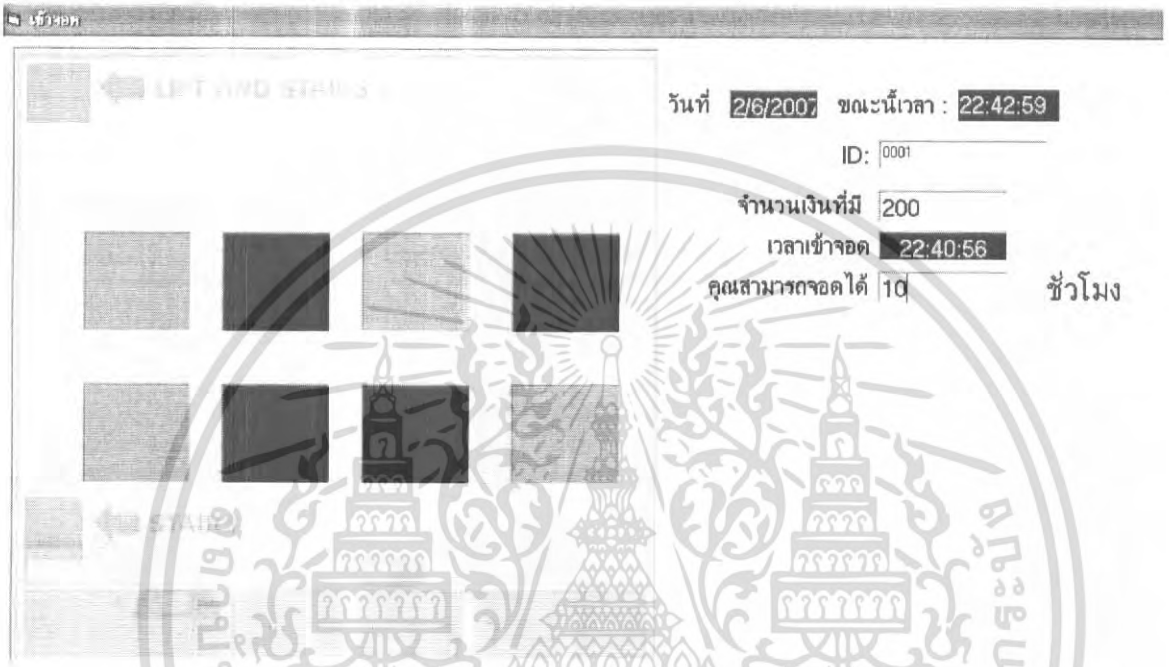
จากการทดลองเมื่อเลือกที่จะติดต่อกับบัตร สมาร์ตการ์ด ไม่ว่าจะเป็นทั้งการเขียนหรือการอ่าน พบว่าสามารถติดต่อกับสมาร์การ์ดได้จริงตามที่ได้ออกแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

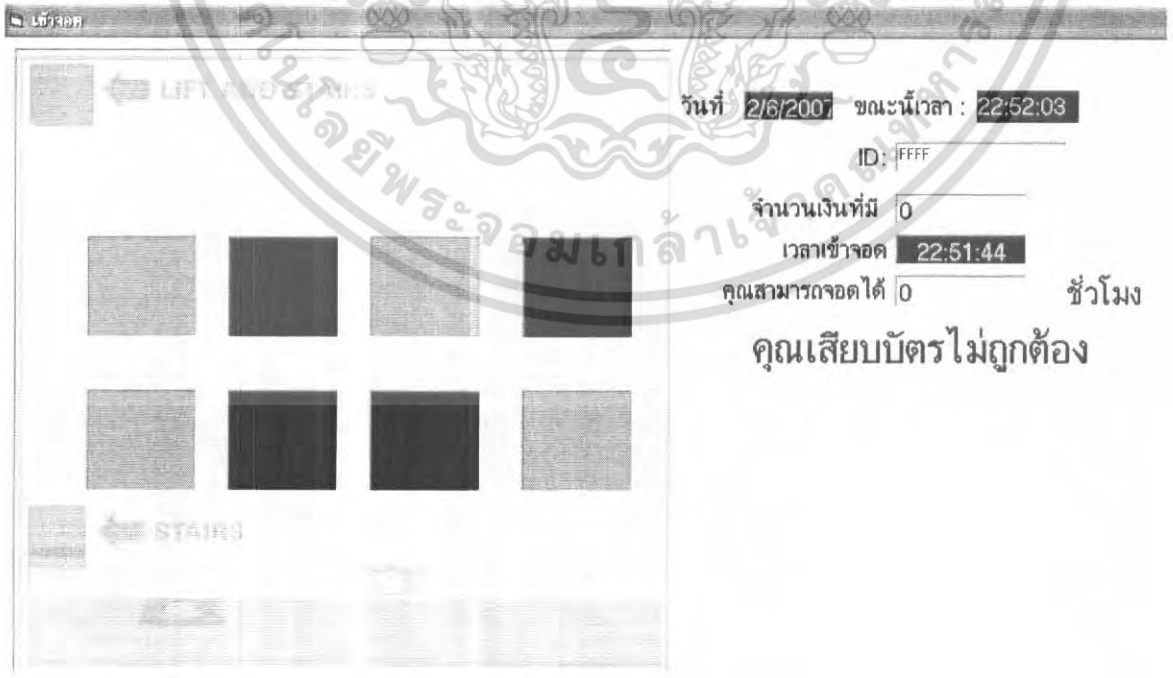
8.4 การทดลองการทำงานของระบบผ่านทาง Visual Basic 6.0

สำหรับวิธีการทดสอบและใช้งาน สามารถทำได้โดยหลังจากลงอุปกรณ์บนแผงวงจรครบถ้วน และประกอบเข้ากับเครื่องตรวจจับวัตถุ และสมาร์ตการ์ด ตรวจดูความเรียบร้อยต่างๆของสายวงจร และการบัดกรีอีกครั้งว่าถูกต้องหรือไม่ หลังจากนั้นจึงทำการจ่ายไฟเลี้ยงให้แก่วงจร

รูปแสดงโปรแกรมที่ โหมดต่างๆ

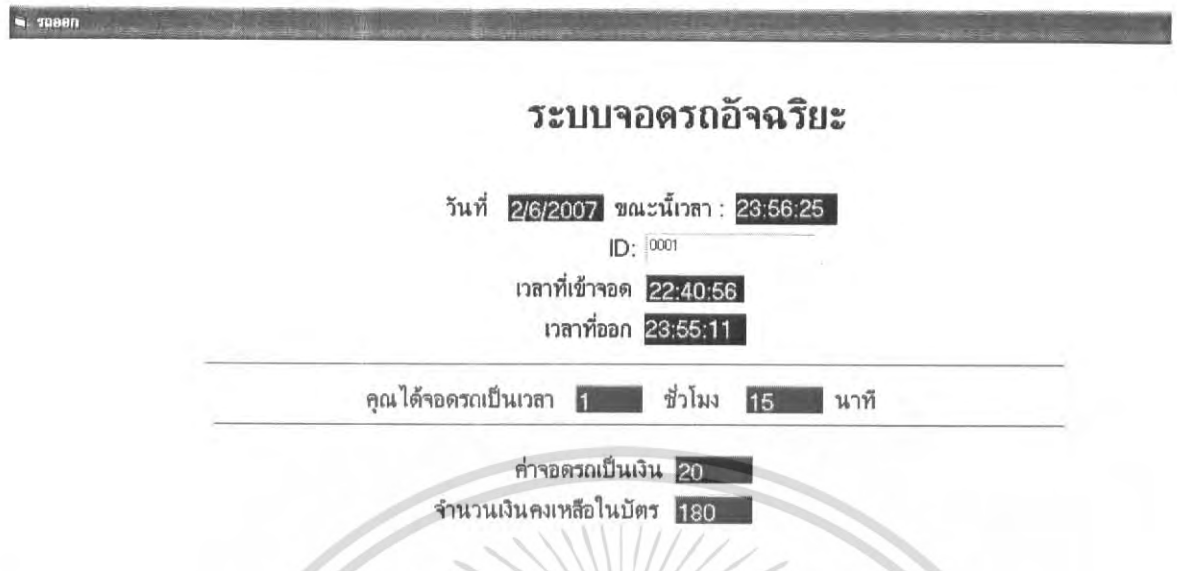


รูปที่ 8.11 รูปของระบบขณะเสียบบัตรเพื่อนำรถเข้าจอด



รูปที่ 8.12 รูปของระบบขณะเสียบบัตรไม่ถูกต้องที่ทางเข้า

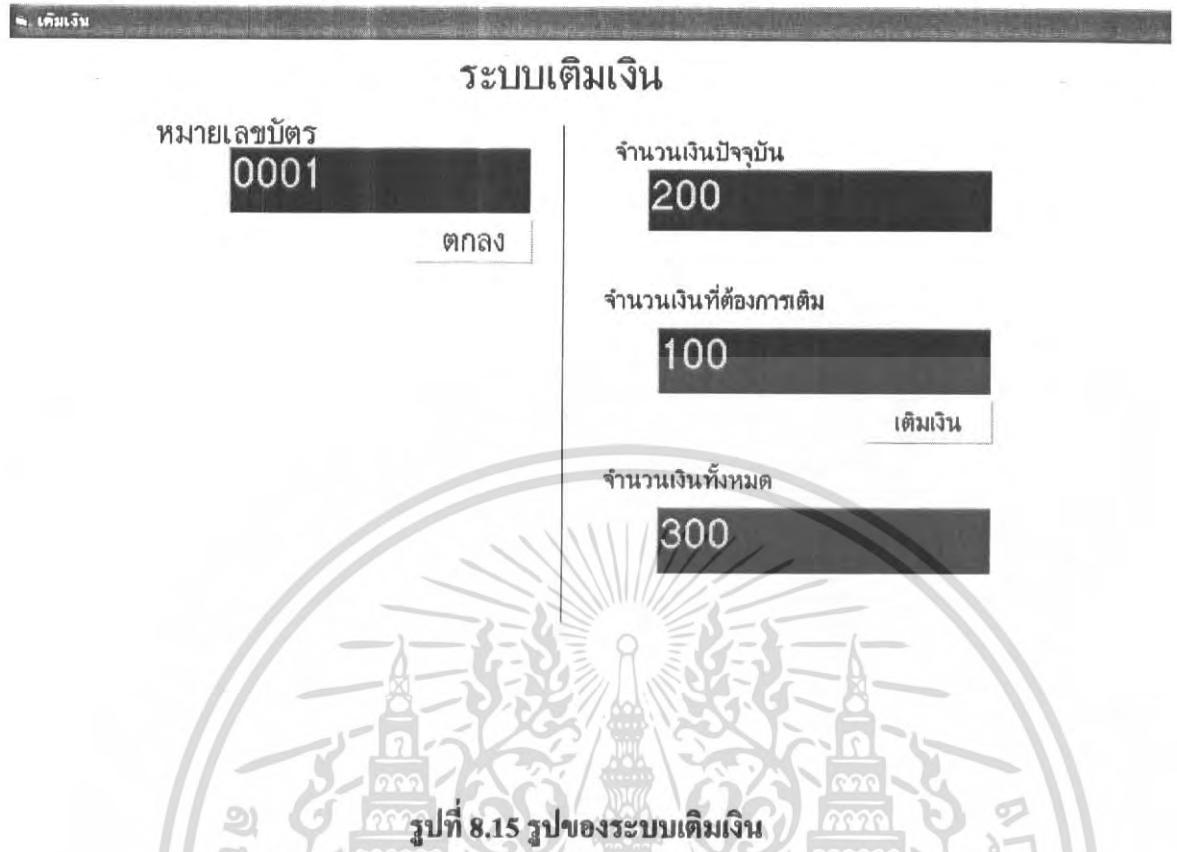
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 8.13 รูปของระบบขณะเสียบบัตรเพื่อนำรถออกจากที่จอด

รูปที่ 8.14 รูปของระบบขณะเสียบบัตรเพื่อนำรถออกจากที่จอดไม่ถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



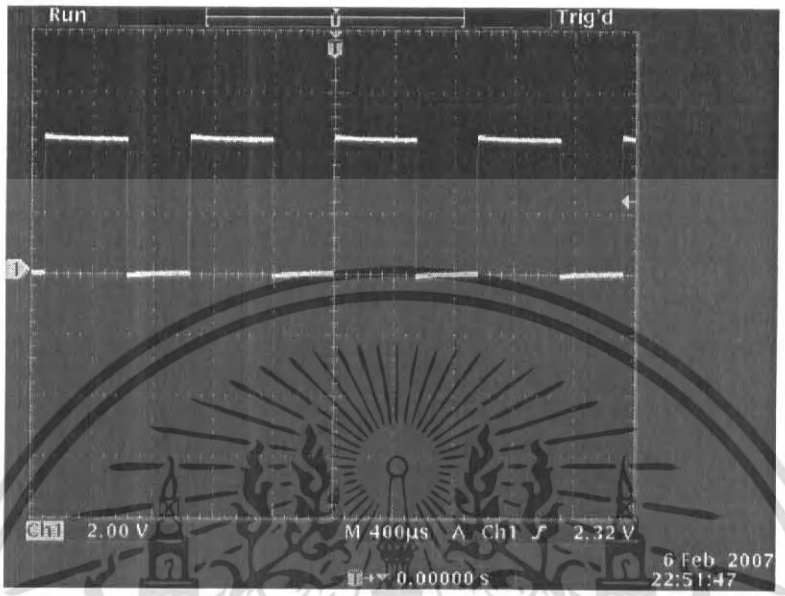
รูปที่ 8.15 รูปของระบบเติมเงิน

รูปที่ 8.16 รูปของระบบลงทะเบียน

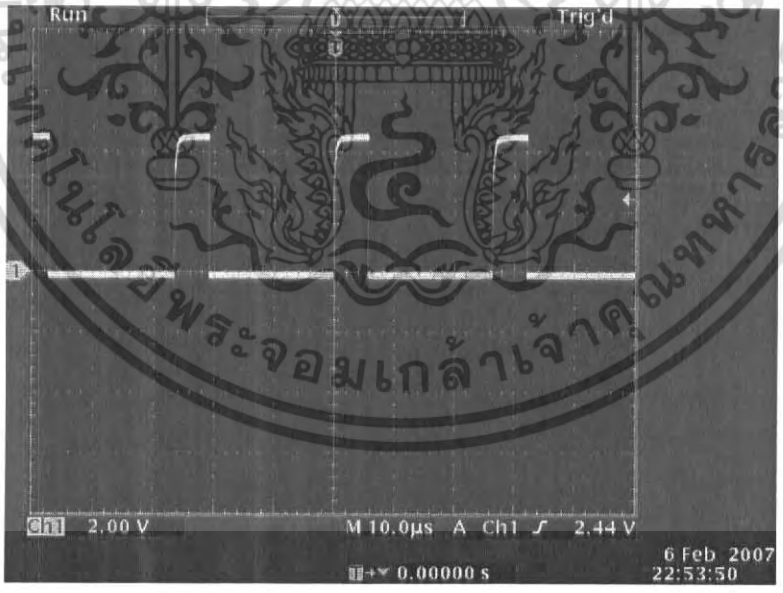
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.5 การทดลองวัดรูปสัญญาณ ณ ตำแหน่งขาต่างๆในชุด Sensor

- ทำการวัดสัญญาณที่ขาต่างๆ ของ sensor

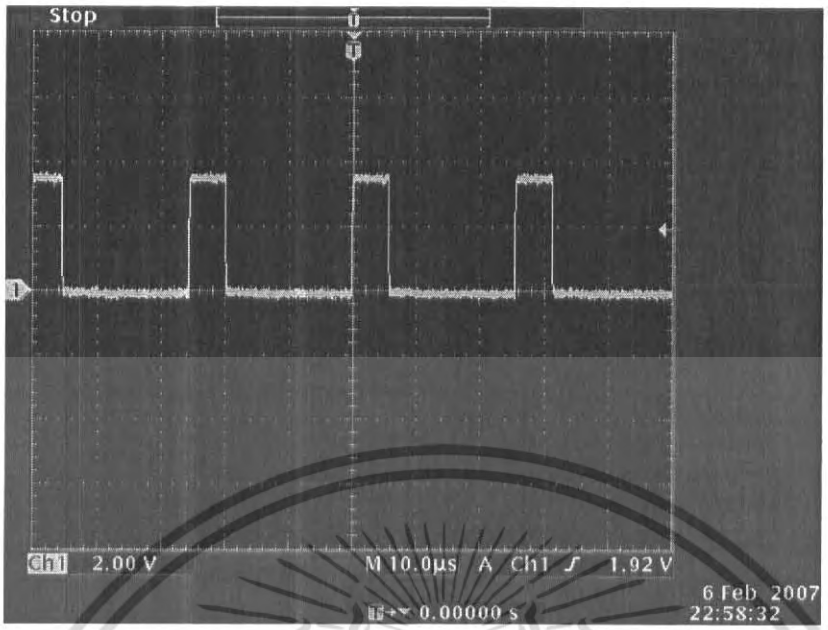


รูปที่ 8.17 สัญญาณความถี่ 1kHzจากวงจร555ตัวแรก

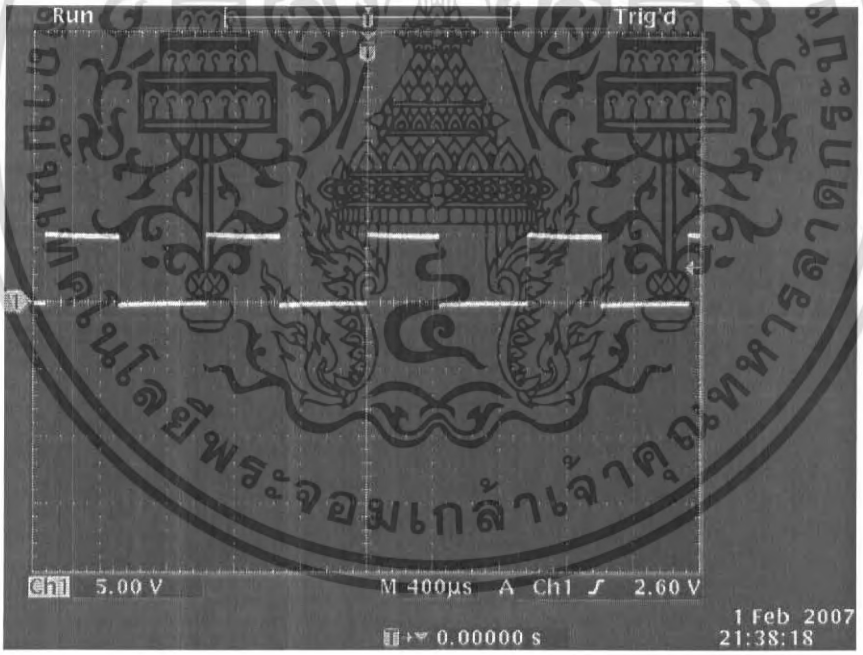


รูปที่ 8.18 สัญญาณ Pulse ความถี่ 38kHzจากวงจร555ตัวที่สอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

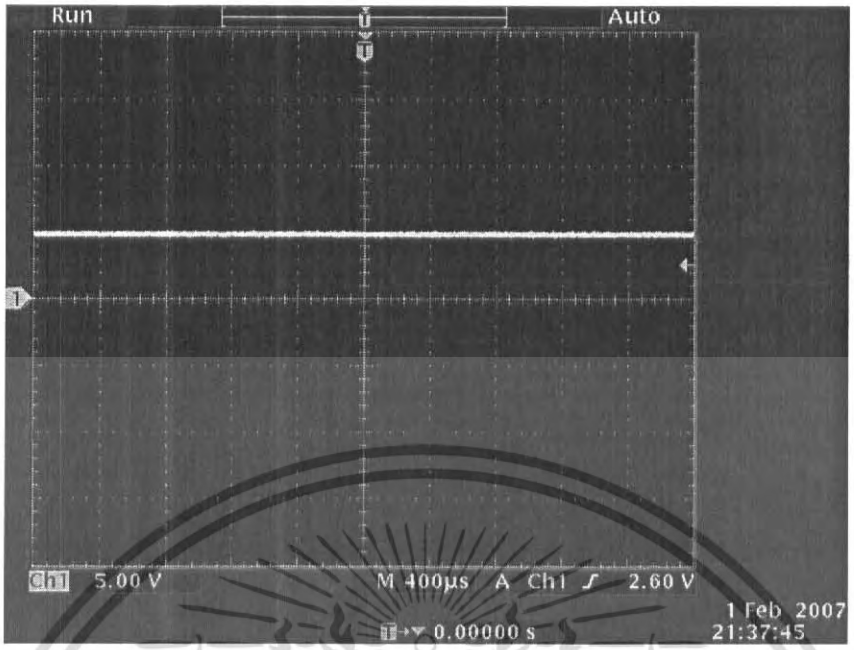


รูปที่ 8.19 สัญญาณ Pulseที่เกิดจากการ Modulate จากวงจร555ทั้งสอง



รูปที่ 8.20 รูปสัญญาณขณะไม่มีรถจอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 8.21 รูปสัญญาณขณะมีรถจอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 9 วิเคราะห์และสรุปผลการทดลอง

จากผลการทดลองที่ได้แสดงในบทที่ 8 การที่จะโปรแกรมลงไปภายในบัตรสมาร์ตการ์ดต้องใช้ MCS-51 เป็นตัวเก็บโปรแกรมเพื่อที่จะนำไปใช้ โดยสมาร์ตการ์ดที่เรานำมาใช้งานสามารถเขียนและอ่านข้อมูลภายในบัตรได้ จากที่เราทำการทดลองวัดสัญญาณเราสามารถจับสัญญาณได้ทั้งตอน ที่เริ่มโปรแกรม , หยุดการทำงานของโปรแกรม , ขณะที่อ่าน และ เขียนข้อมูล และสัญญาณจากขา ATR พบว่าได้สัญญาณเป็นจริง ตาม Timing diagram ที่ Datasheet ระบุ

จากการทดลองเมื่อเราต้องการที่จะกระทำการใด ๆ กับการ์ดไม่ว่าจะเป็นการReset อ่าน เขียน พบว่ากระบวนการต่างๆที่เกิดขึ้นจะเป็นไปตามมาตรฐานของบัตรที่ระบุไว้ในDatasheet โดยแต่ละกระบวนการนั้นจะเน้นช่วงเวลาการจ่ายสัญญาณให้การ์ดให้เป็นไปตาม Timing Diagram ซึ่งการติดต่อกับบัตรสมาร์ตการ์ด แต่ละแบบนั้นจำเป็นต้องศึกษา และทำความเข้าใจการสื่อสารกับบัตรเป็นอย่างดี เพื่อที่จะควบคุมการติดต่อสื่อสารกับบัตรได้อย่างถูกต้อง ไม่เกิดการผิดพลาดขึ้นได้ โดยเฉพาะการป้อนรหัส PSC หากผิดพลาดอาจทำให้บัตรถูก Lock ได้

จากการทดลองพบว่า ปัจจัยที่เป็นตัวกำหนดความเร็วในการสื่อสารคือ สัญญาณนาฬิกา หากเพิ่มความถี่ของสัญญาณนาฬิกา เวลาที่ใช้ในการนำสัญญาณเข้า และออก จากบัตรสมาร์ตการ์ดจะมีค่าลดลง ในทางตรงกันข้ามหากสัญญาณนาฬิกามีค่าลดลง เวลาในการติดต่อสื่อสารก็จะเพิ่มขึ้น

จากการทดลองวัดสัญญาณผ่านเครื่องรับ infared พบว่าปัจจัยที่มีผลในการรับสัญญาณมีหลายปัจจัยด้วยกันเราต้องพิจารณาในหลาย ๆ ด้าน ไม่ว่าจะเป็นด้านกระแสที่จ่ายให้กับตัว infared ควรจะจ่ายกระแสให้สัมพันธ์กับระยะทางที่ใช้ในการส่งถ้าจ่ายกระแสมากเกินไปทำให้ข้อมูลที่ได้เกิดความผิดพลาดได้

จากการทดลองในการใช้ตัว Master ติดต่อกับ Slave แต่ละตัวไม่ว่าจะเป็นเครื่องอ่านบัตรที่ทางเข้าหรือ เครื่องอ่านบัตรที่ทางออกปัจจัยที่มีผลมากคือ การ delay เนื่องจากหากกำหนด delay ในการรับส่งไม่เหมาะสม อาจทำให้ข้อมูลที่ได้เกิดความผิดเพี้ยนได้

เนื่องจากการประมวลผลของข้อมูล เป็นการประมวลผลผ่าน Visual Basic จึงทำให้มีการบิดเบือนของข้อมูลใน Data base น้อยมาก ปัจจัยหลักหากเกิดการผิดพลาดมักเกิดจากข้อมูลที่ส่งมาจากตัว Master หากวางระบบงานทางด้านการส่งผ่านข้อมูลจาก Slave → Master → PC ดีก็จะทำให้ความผิดพลาดของข้อมูลใน Data Base ผิดพลาดน้อยมาก จนแทบไม่มี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1) ฐานข้อมูล Access 2002 XP บัณฑิต จามรภูมิ สำนักพิมพ์ สวัสดีไอที

2) คู่มือเรียน Visual Basic 6 ฉันทวุฒิ พืชผล พิชัย สันติกุลานนท์ พิมพ์ครั้งที่ 5 บริษัทโปรวิชั่น จำกัด

3) Visual Basic 6 ฉบับโปรแกรมเมอร์ กิตติ ภักดีวัฒนะกุล จำลอง ครูอุตสาหะ พิมพ์ครั้งที่ 11 สำนักพิมพ์ เคพีที คอมพ์ แอนด์ คอนซัลท์

4) Visual Basic 6 ฉบับฐานข้อมูล กิตติ ภักดีวัฒนะกุล จำลอง ครูอุตสาหะ พิมพ์ครั้งที่ 11 สำนักพิมพ์ เคพีที คอมพ์ แอนด์ คอนซัลท์

5) การเขียนโปรแกรม สำหรับ Client/Server ด้วย Visual Basic 6 และ ASP.VBScript.Access.SQL Server

6) พ.อ. เจนวิทย์ เหลืองอร่าม ปิยวิทย์ เหลืองอร่าม สำนักพิมพ์ เจ แอนด์ พี ดอท เนท

7) เรียนรู้และปฏิบัติการเชื่อมต่อคอมพิวเตอร์กับอุปกรณ์ภายนอกผ่านพอร์ตอนุกรม โดย อรรถพล บุญยะโภคา วรพจน์ กรแก้ววัฒนกุล ชัยวัฒน์ ลิ้มพรจิตร วิไล Innovation Experiment Co.,Ltd.

8) เทคโนโลยีสมาร์ตการ์ด เลสแซ่ตั้ง บริษัท ซีเอ็ดยูเคชั่น จำกัด(หมาชน)

9) http://www.bangkokbiznews.com/2003/kit/2003kit/0130/14.html

10) http://www.bangkokbiznews.com/2003/kit/2003kit/0130/14.html (wire less smart card)

11) http://library.kmitnb.ac.th/projects/edu/TCT/tct0013t.html (เครื่องอ่านสมาร์ตการ์ด)

12) http://tmcc.nectec.or.th/smartcard/smartcard_main.htm (smarteard news)

13) http://www.cipitc.or.th/court_order_decision/IP_judgement/jm2545/ip_121_2545.html

14) http://www.thaiinternetwork.com/article/view.php?No=123 (wire less smart card)

15)http://web.ku.ac.th/schoolnet/snet1/network/smart_c.htm (บัตรอเนกประสงค์)

16) http://www.atshop.com/MRWINDOW/(tn110201)smartcard.html (Smart Card)

17) http://www.nectec.or.th/bid/hotissue_smartcard.htm (Business smart card)

18) http://securetech-corp.com/smart_ca.html #1ACS Smart Card 8K EEPROM

19) www.slb.com

20) http://www.smartcards.net/infosec/index.html (detail)

20) http://www.scmegastore.com (market)

21) http://www.schlumbergersema.com/caseStudies/infrastructure/deXaBadgeReprint.pdf (manual)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้