สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบควบคุมประตูเพื่อรักษาความปลอดภัยด้วยไอบัททอน

Door Control Security System By using Ibutton

โดย

นายพงษ์เทพ ศิษย์สุวรรณ์

นายเสกสรรค์ กอเซ็มมูซอ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เลขหมู่.....
เลขทะเบียน.....62172
วัน,เดือน,ปี.....31 ก.ค. 2549

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

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DOOR CONTROL SECURITY SYSTEM BY USING IBUTTON

BY

MR. PONGTAP SITSUWAN

MR. SAKESON KORSEMMUSOR

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENT FOR THE DEGREE OF

BACHELOR IN DERPARTMENT OF INFORMATION ENGINEERING

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2004

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	ระบบควบคุมประตูเพื่อรักษาความปลอดภัยด้วยไอบัททอน
ชื่อนักศึกษา	นายพงษ์เทพ ศิษย์สุวรรณ์ รหัสประจำตัว 45015857
	นายเสกสรรค์ กอเซ็มมูซอ รหัสประจำตัว 45015876
อาจารย์ที่ปรึกษา	ผศ.อุทัย ศรีธีระวิโรจน์
	อาจารย์เกล็ดดาว สัตย์เจริญ
ระดับการศึกษา	ปริญญาตรีวิศวกรรมศาสตรบัณฑิต
ภาควิชา	วิศวกรรมสารสนเทศ
ปีการศึกษา	2547

ปริญญานิพนธ์ฉบับนี้ได้รับการอนุมัติเป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

.....

(ผศ.อุทัย ศรีธีระวิโรจน์)

อาจารย์ผู้ควบคุมวิทยานิพนธ์

.....

(อาจารย์เกล็ดดาว สัตย์เจริญ)

อาจารย์ผู้ควบคุมวิทยานิพนธ์

ลิขสิทธิ์ของคณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อปริญญานิพนธ์ ระบบควบคุมประตูเพื่อรักษาความปลอดภัยด้วยไอบัททอน
นักศึกษา นายพงษ์เทพ ศิษย์สุวรรณ์ รหัสประจำตัว 45015857
นายเสกสรรค์ กอเซ็มมูซอ รหัสประจำตัว 45015876
อาจารย์ที่ปรึกษา ผศ.อุทัย ศรีธีระวิโรจน์
อาจารย์ เกล็ดดาว สัตย์เจริญ
ระดับการศึกษา ปริญญาตรี วิศวกรรมศาสตรบัณฑิต
ภาควิชา วิศวกรรมสารสนเทศ
ปีการศึกษา 2547

บทคัดย่อ

โครงงานนี้มีวัตถุประสงค์เพื่อพัฒนาระบบการเข้า-ออกห้องที่ต้องการความปลอดภัยสูง โดยการใช้อุปกรณ์ iButton เป็นกุญแจอิเล็กทรอนิกส์ และใช้ไมโครคอนโทรลเลอร์ตระกูลPIC 18F458 ในการอ่านค่ารหัสประจำตัวของตัวอุปกรณ์ iButton ควบคู่กับการใส่รหัสลับตัวเลข4 หลัก เพื่อผ่านเข้าห้อง และในส่วนของประตูเข้า-ออก ก็จะใช้ไมโครคอนโทรลเลอร์ในการควบคุมการจำลองการทำงาน โดยสามารถทำการบันทึกเวลาเข้าออกของพนักงาน สามารถเพิ่มผู้ใช้งานได้ และสามารถระงับการใช้งานของรหัส iButton ได้โดยใช้โปรแกรมภาษา Visual Basic 6.0 โดยสามารถทำการแสดงเป็นรายงานการใช้งานทั้งหมด คือแสดงเวลาเข้า-ออก จำนวนคนภายในห้อง แสดงรายชื่อบุคคล และสามารถทำการค้นหาได้ตามวันและเวลาที่ต้องการ ผ่านคอมพิวเตอร์ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Door Control Security System By Using Ibutton
Student	Mr. Pongtap Sitsuwan ID.45015857
	Mr.Sakeson Korsemmusor ID.45015876
Advisor	Asst.Prof.Uthai Sritheeravirojana
	Miss. Kleddao Satcharoen
Graduate Level	Bachelor Degree of Information Engineering
Department	Information Engineering
Academic Year	2004

Abstract

This project is present to "Door control security system by using iButton" with use PIC 18F458 to controller in the door in- out. Only authorized people could go through the door by insert their iButton and input the correct password. (4 digit number) The incoming-out going is time recorded into the database with the provided interfarence, an administrator could grant access for a new user or revoke access from same user to not allow them to come -in the room.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สามารถสำเร็จขึ้นได้ด้วยดีนั้น ขอขอบพระคุณอาจารย์ที่ปรึกษา ผศ.อุทัย ศรีธีระวิโรจน์ และ อาจารย์เกล็ดดาว สัตย์เจริญ ตลอดจนอาจารย์ทุกท่านในสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังที่ให้คำปรึกษาต่างๆ รวมถึงแนะนำแนวทางการแก้ปัญหาในปริญญานิพนธ์ฉบับนี้

สุดท้ายนี้ขอขอบพระคุณ บิดา-มารดา ซึ่งเป็นผู้ให้กำเนิดและเป็นผู้ที่คอยให้กำลังใจ รวมถึงให้การสนับสนุนในทุก ๆ ด้านตลอดมา และขอขอบคุณ พี่-น้องและเพื่อน ๆ ทุกคนที่คอยเป็นกำลังใจเสมอ

นายพงษ์เทพ ศิษย์สุวรรณ์

นายเสกสรรค์ กอเซ็มมูซอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญรูปภาพ	ช
สารบัญตาราง	ญ
บทที่1 บทนำและความเป็นมา	1
1.1 แนวคิดและที่มาของปัญหา	1
1.2 วัตถุประสงค์ของโครงงาน	2
1.3 ขอบเขตของโครงงาน	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ	2
บทที่ 2 ทฤษฎีที่ใช้ในการออกแบบ	3
2.1 ไอบัททอนและหลักการ	3
2.1.1 คุณสมบัติร่วมของอุปกรณ์ iButton	3
2.1.2 โครงสร้างของอุปกรณ์ iButton	4
2.1.3 รูปร่างตัวถังของอุปกรณ์ iButton	5
2.1.4 คุณสมบัติทางเทคนิคของ DS1990A	7
2.1.5 โครงสร้างหน่วยความจำรอมของ DS1990A	7
2.1.6 การต่อใช้งาน DS1990A	8
2.1.7 จังหวะและคำสั่งในการทำงานของ DS1990A	8
2.2 โครงสร้างและสถาปัตยกรรมของ PIC 18F458	9
2.2.1 คุณสมบัติของ PIC 18F458	9
2.2.2 หน่วยความจำโปรแกรม	18
2.2.3 หน่วยความจำข้อมูล (Data Memory)	19
2.2.4 รีจิสเตอร์	19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

เรื่อง	หน้า
2.3 ระบบสื่อสารข้อมูลอนุกรมแบบหนึ่งสาย (1- Wire™ Serial Bus)	19
2.3.1 คุณสมบัติทางเทคนิคของระบบบัสหนึ่งสาย	20
2.3.2 คุณสมบัติของช่องเวลา (Timeslot)	21
2.3.3 ช่องเวลาการรีเซตและตอบสนอง	22
2.3.4 ช่องเวลาการอ่านข้อมูลของอุปกรณ์ตัวหลัก	22
2.3.5 ช่องเวลาการเขียนข้อมูลของอุปกรณ์ตัวหลัก	24
2.3.6 รูปแบบของการสื่อสารข้อมูลแบบหนึ่งสาย	25
2.4 ทฤษฎีการติดต่อสื่อสารระหว่างคอมพิวเตอร์และอุปกรณ์ภายนอกผ่านพอร์ตอนุกรม	26
2.4.1 การสื่อสารแบบอนุกรม	26
2.5 มาตรฐานพอร์ตอนุกรม RS-232	29
2.6 หัวต่อสำหรับพอร์ต RS-232 และการเชื่อมต่อ	29
2.6.1 รายละเอียดหน้าที่การทำงานในแต่ละขาของพอร์ตอนุกรม RS-232	31
2.7 อุปกรณ์ที่ทำหน้าที่รับและส่งข้อมูลแบบอะซิงโครนัส	32
2.7.1 ชนิดของอุปกรณ์ที่ทำหน้าที่รับและส่งข้อมูลแบบอะซิงโครนัส	32
2.7.2 วงจรภายในและรีจิสเตอร์ของพอร์ตอนุกรม RS-232	33
2.7.3 ลักษณะสัญญาณอินพุตและเอาต์พุตของพอร์ตอนุกรม RS–232	33
2.8 ทฤษฎีทั่วไปเกี่ยวกับโปรแกรมวิชวลเบสิก	34
2.8.1 สาเหตุที่ต้องใช้โปรแกรมวิชวลเบสิก	35
2.8.2 เริ่มต้นกับโปรแกรมวิชวลเบสิก	35
2.9 ขั้นตอนในการพัฒนาโปรแกรมวิชวลเบสิก	36
2.9.1 คำสั่งสำหรับเขียนโปรแกรม	37
2.9.2 การสนใจเหตุการณ์ที่เกิดขึ้นกับโปรแกรมวิชวลเบสิก	38
2.9.3 กระบวนการและฟังก์ชั่นในโปรแกรมวิชวลเบสิก	38

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

เรื่อง	หน้า
บทที่ 3 การออกแบบและการสร้าง	41
3.1ขั้นตอนการดำเนินงาน	41
3.2 ขั้นตอนการออกแบบฐานข้อมูล	45
3.3 การออกแบบโปรแกรมไมโครคอนโทรลเลอร์	48
3.4 การต่อจอแสดงผลLCD(Liquid Crystal Display)	49
3.5 การเชื่อมต่อกับ Key Switch แบบ 4x3	53
3.6 การเชื่อมต่อกับมอเตอร์ไฟฟ้ากระแสตรง (DC Motor)	56
3.7 การเชื่อมต่อกับตัวอุปกรณ์ ไอบัททอน	59
3.8 การทำงานเชื่อมต่อกับคอมพิวเตอร์	61
บทที่ 4 ผลการทดลอง	62
4.1อุปกรณ์ที่ใช้ในการทดสอบ	62
4.2 ขั้นตอนในการทดสอบ	62
บทที่ 5 สรุปผลและข้อเสนอแนะ	71
5.1สรุปผลการดำเนินงาน	71
5.2 ปัญหาและแนวทางการแก้ไข	71
5.3 ข้อเสนอแนะและแนวทางพัฒนา	72
ภาคผนวก	73
บรรณานุกรม	84

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

รูปที่ **หน้า**

บทที่ 2

รูปที่ 2.1 โครงสร้างการทำงานของอุปกรณ์ iButton 4

รูปที่ 2.2 วงจรเชื่อมต่อภายใน iButton 4

รูปที่ 2.3 ลักษณะตัวถังของอุปกรณ์ iBUTTON 5

รูปที่ 2.4 ข้อมูลสำคัญของอุปกรณ์ iButton ที่พิมพ์อยู่บนตัวถัง 6

รูปที่ 2.5 แสดงอุปกรณ์ช่วยในการเชื่อมต่ออุปกรณ์ iButton 6

รูปที่ 2.6 โครงสร้างข้อมูลของหน่วยความจำรอมภายใน DS1990A 7

รูปที่ 2.7 การต่อใช้งาน DS1990A กับขาพอร์ตของไมโครโทรลเลอร์ 8

รูปที่2.8 สถาปัตยกรรมภายในของไมโครคอนโทรลเลอร์ PIC 18F458 11

รูปที่2.9 แสดงขาสัญญาณของ PIC 18F458 11

รูปที่2.10 การวางพื้นที่ของหน่วยความจำโปรแกรมของ PIC 18F458 18

รูปที่ 2.11 การเชื่อมต่อบนระบบบัสหนึ่งสาย 21

รูปที่ 2.12 ช่องเวลาการรีเซตและการตอบรับของอุปกรณ์บนระบบบัสหนึ่งสาย 23

รูปที่ 2.13 ช่องเวลาการอ่านข้อมูลของอุปกรณ์ตัวหลักซึ่งตรงกับช่องเวลาการเขียนข้อมูลของอุปกรณ์ตัวรับ 23

รูปที่ 2.14 ช่องเวลาการเขียนข้อมูล “1” ของอุปกรณ์ตัวหลักซึ่งตรงกับช่องเวลาการอ่านข้อมูลของอุปกรณ์ตัวรับ 24

รูปที่ 2.15 ช่องเวลาการเขียนข้อมูล “0” ของอุปกรณ์ตัวหลัก 25

รูปที่ 2.16 รูปแบบอย่างง่ายที่สุดของข้อมูลอนุกรมแบบอะซิงโครนัส 28

รูปที่ 2.17 การจัดขาของคอนเน็คเตอร์พอร์ทอนุกรมตามมาตรฐานRS-232ทั้งแบบ DB-9 และ DB-25 30

รูปที่ 2.18 การต่ออุปกรณ์ภายนอกกับพอร์ทอนุกรมของคอมพิวเตอร์ในลักษณะต่างๆ 30

รูปที่ 2.19 หน้าจอภาพของโปรแกรมวิชวลเบสิก 37

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ(ต่อ)

รูปที่ **หน้า**

บทที่ 3

รูปที่ 3.1 แผนผังการดำเนินงาน 42
รูปที่ 3.2 บล๊อคไดอะแกรมการทำงานของระบบ 44
รูปที่ 3.3 ตารางที่ใช้ในการออกแบบ 45
รูปที่ 3.4 ฟิลด์ในตาราง Department 46
รูปที่ 3.5 ฟิลด์ในตารางEmployee 47
รูปที่ 3.6 ฟิลด์ในตาราง iBUTTON 47
รูปที่ 3.7 ฟิลด์ในตาราง Initial 47
รูปที่ 3.8 ฟิลด์ในตาราง Login 48
รูปที่ 3.9บล็อกไดอะแกรมอุปกรณ์อินพุต และ เอาต์พุต 49
รูปที่ 3.10 จอแสดงผล LCD 49
รูปที่ 3.11 การเชื่อมต่อ LCD แบบ 4 บิต Data 50
รูปที่ 3.12 การวางตำแหน่งขาสัญญาณของจอแสดงผลแอลซีดี 51
รูปที่ 3.13 ลักษณะของคีย์สวิทช์แบบเมทริกซ์ขนาด 4x3 53
รูปที่ 3.14 วงจรการทดลองต่อคีย์สวิทช์ 54
รูปที่ 3.15 การต่อวงจรของชุดควบคุมมอเตอร์ 56
รูปที่ 3.16 ประตูจำลองที่ใช้ในการเข้า 57
รูปที่ 3.17 ประตูจำลองที่ใช้ในการออก 57
รูปที่ 3.18 อุปกรณ์ภายในทั้งหมด 58
รูปที่ 3.19 หน้าจอ LCD แสดงว่าพร้อมการใช้งาน 58
รูปที่ 3.20 อุปกรณ์IBUTTON และชุดรับรหัส IBUTTON 59
รูปที่ 3.21 ตัวอย่างการต่อ DS1990A เข้ากับขาพอร์ตของไมโครคอนโทรลเลอร์ 59

บทที่4

รูปที่ 4.1 หน้าต่างข้อมูลทั่วไปของพนักงาน 64
รูปที่ 4.2 หน้าต่างรายละเอียดของพนักงาน 65
รูปที่4.3 หน้าต่างข้อมูลทั้งหมดของพนักงาน 66
รูปที่ 4.4 รหัสของ iBUTTON และรหัสบัตรประชาชน 67

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ(ต่อ)

รูปที่	หน้า
รูปที่ 4.5 หน้าต่างประเภทของงาน	68
รูปที่ 4.6 เป็นหน้าต่างที่ใช้ติดต่อกับฮาร์ดแวร์	69
รูปที่ 4.7 เป็นหน้าต่างที่ใช้แสดงรายงานการเข้า / ออกประตู	70

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตาราง	หน้า
ตารางที่ 2.1 หน้าที่การทำงานของขาสัญญาณในแต่ละขา	12
ตารางที่2.2 สรุปหน้าที่การทำงานของขาสัญญาณในแต่ละขา	17
ตารางที่ 3.1 ชื่อและหน้าที่ของขาสัญญาณต่างๆของ LCD	51
ตารางที่ 3.2 สรุปสภาวะการกดสวิทช์ต่างๆ	54
ตารางที่ 3.3 ค่าต่างๆจากการกดคีย์	55
ตารางที่ 3.4 ค่าที่ใช้ในการแสกนคีย์	55
ตารางที่ 3.5 ข้อมูลในการกดสวิทช์ในตำแหน่งต่างๆ	56

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำและความเป็นมา

1.1 แนวคิดและที่มาของปัญหา

ปัจจุบันระบบการทำงาน ตลอดจนสำนักงานและองค์กรต่างย่อมมีอุปกรณ์ที่มีความสำคัญและมีมูลค่าสูง ดังนั้นระบบความปลอดภัยของการรักษาอุปกรณ์ต่างๆ จึงมีความจำเป็นต่อองค์กร หรือสำนักงานนั้นๆ และสืบเนื่องมาจากผู้จัดทำเห็นความสำคัญของการจัดการระบบฐานข้อมูลกับระบบความปลอดภัยของการรักษาอุปกรณ์ต่างๆ จึงได้มีการพัฒนาโดยการนำเอาอุปกรณ์อิเล็กทรอนิกส์ ซึ่งก็คือ iButton มาจัดการเกี่ยวกับการเข้าออกห้อง เพื่อให้เกิดความรัดกุมในการเข้าออก กว่าอุปกรณ์ชนิดอื่น เช่นการใช้การ์ดแม่เหล็ก(Magnet Card) หรือ หัวอ่านรหัสแท่ง(Barcode Reader) โดยอุปกรณ์ทั้ง 2 ชนิดนี้มีข้อเสีย คือ สามารถที่จะทำการปลอมรหัสขึ้นมาให้ซ้ำกันได้ และตัวอุปกรณ์นี้จะมีความผิดพลาดของการอ่านรหัสค่อนข้างสูง เนื่องจากความผิดพลาดของระยะโฟกัสของแสงเลเซอร์ที่ตกกระทบลงบนแถบรหัส การรูดบัตรเร็วเกินไปหรือช้าเกินไปก็จะไม่สามารถอ่านข้อมูลที่ถูกต้องจากบัตรได้ เพราะข้อจำกัดของโปรแกรมที่ใช้อ่านรหัส และข้อจำกัดทางด้านการตอบสนองความถี่ของวงจรขยายสัญญาณจากหัวอ่านบาร์โค้ด และตัวอุปกรณ์เองยังมีความเปราะบางเกิดการชำรุดเสียหายได้ง่าย และเนื่องด้วยคุณสมบัติของตัวอุปกรณ์ iButton ทุกตัวจะมีเลขรหัสประจำตัว ซึ่งใช้เทคโนโลยีเลเซอร์ในการระบุรหัสประจำตัว ทำให้อุปกรณ์ iButton แต่ละตัวจะมีเลขรหัสประจำตัวที่แตกต่างกัน ซึ่งการอ่านรหัสของตัวอุปกรณ์จะทำการอ่านรหัสจากหน่วยความจำรอมภายในตัวอุปกรณ์ ซึ่งโครงสร้างของหน่วยความจำรอมอยู่ภายในตัวอุปกรณ์

iButton จะแบ่งเป็น 3 ส่วนคือ รหัสตรวจสอบด้วยส่วนซ้ำซ้อนแบบวนขนาด 8 บิต หรือ 1 ไบต์จะอยู่ในแอดเดรสสูง ถัดมาคือรหัสประจำตัวขนาด 48 บิตหรือ 6 ไบต์ข้อมูลตรงส่วนนี้เองที่ทำให้อุปกรณ์ iButton เหมาะสมอย่างยิ่งในการนำไปใช้เป็นกุญแจอิเล็กทรอนิกส์ เนื่องจากจะมีรหัสที่แตกต่างกันมากถึง 2^{48} ซึ่งเท่ากับ 2.81×10^{14} ถ้ามีการผลิตจำนวนปีละ1,000 ล้านตัว จะต้องใช้เวลามากกว่า 200 ปี จึงจะใช้เลขรหัสครบหมดทุกตัว ดังนั้นจึงทำให้มั่นใจได้ว่า โอกาสที่จะเกิดเลขรหัสซ้ำนั้นแทบไม่มีเลย ข้อมูลส่วนสุดท้ายคือรหัสตระกูลของอุปกรณ์ มีขนาด 8 บิตหรือ 1 ไบต์ และการเชื่อมต่อเพื่ออ่านหรือเขียนข้อมูล ทำได้โดยการสัมผัสโดยตรงกับอุปกรณ์ หน้าสัมผัสทางไฟฟ้าทั้งหมดทำจากสแตนเลสสตีล จึงทนทานต่อการสัมผัสเชื่อมต่อและไม่เป็นสนิม และใช้เวลาในการเข้าถึงและอ่านข้อมูลน้อยกว่า 5 มิลลิวินาที ดังนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์ iButton จึงมีความเหมาะสมในการนำมาพัฒนาการจัดการเกี่ยวกับการเข้าออกห้อง เพื่อให้เกิดการรัดกุมมากขึ้น

1.2 วัตถุประสงค์ของโครงงาน

1.2.1 เพื่อพัฒนาระบบการจัดการเกี่ยวกับการเข้าออกห้อง เพื่อให้เกิดการรัดกุมขึ้น

1.2.2 กำหนดสิทธิการเข้าออกผ่านคอมพิวเตอร์และตรวจสอบเวลาเข้าออกของพนักงาน

1.2.3 ศึกษาไมโครคอนโทรเลอร์ตระกูล PIC

1.2.4 เพื่อป้องกันการเข้าออกห้องแบบผิดข้อกำหนด

1.3 ขอบเขตของโครงงาน

1.3.1 ใช้ไมโครคอนโทรลเลอร์ ตระกูล PIC ในการควบคุมการทำงานของประตู

1.3.2 ใช้อุปกรณ์ iButton เป็นกุญแจอิเล็กทรอนิกส์ในการเข้าออกห้อง

1.3.3 สามารถกำหนดสิทธิและบันทึกเวลาในการเข้าออกจากห้องได้ โดยใช้คอมพิวเตอร์

1.3.4 สามารถแสดงจำนวนของพนักงานที่อยู่ภายในห้องผ่านคอมพิวเตอร์ได้

1.3.5 สามารถเพิ่มและระงับผู้ใช้งานผ่านคอมพิวเตอร์ได้

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1. สามารถควบคุมสิทธิในการเข้าออกจากห้องได้โดยผ่านคอมพิวเตอร์
2. สามารถตรวจสอบเวลาการเข้า หรือออกจากห้องของพนักงานได้
3. สามารถนำเอาความรู้ด้านไมโครคอนโทรเลอร์ตระกูล PIC ไปใช้ในงานอุตสาหกรรมได้
4. สามารถแสดงจำนวนของพนักงานที่ปฏิบัติงานอยู่ภายในห้องได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีที่ใช้ในการออกแบบ

2.1 ไอบัททอน และหลักการ

อุปกรณ์ iButton จัดได้ว่าเป็นอุปกรณ์อิเล็กทรอนิกส์ที่มีความแตกต่างไปจากอุปกรณ์สารกึ่งตัวนำที่รู้จักกันโดยทั่วไป เนื่องจากอุปกรณ์ iButton มีรูปร่างตัวถังคล้ายกับกระดุม แต่จะมีจุดสัมผัสทางไฟฟ้า 2 จุดคือ สัญญาณและกราวด์ ซึ่งจะคล้ายกับแบตเตอรี่ลิเธียมหรือที่เรียกกันโดยทั่วไปว่า "ถ่านกระดุม" ลักษณะตัวถังแบบนี้เรียกว่า ไมโครแคน (micro can)

อุปกรณ์ iButton ได้รับการออกแบบและผลิตโดยดัลลัส เซมิคอนดักเตอร์ ผู้นำในการผลิตอุปกรณ์ที่ใช้การติดต่อผ่านระบบบัส 1 สายนั่นเอง และกลุ่มของอุปกรณ์ iButton ก็ใช้รูปแบบการติดต่อสื่อสารผ่านระบบบัส 1 สายเช่นเดียวกัน เทคโนโลยีที่ใช้ในการผลิตอุปกรณ์ iButton คือ เทคโนโลยี CMOS ทำให้อุปกรณ์ iButton ทั้งหมดใช้พลังงานไฟฟ้าในการทำงานต่ำมาก

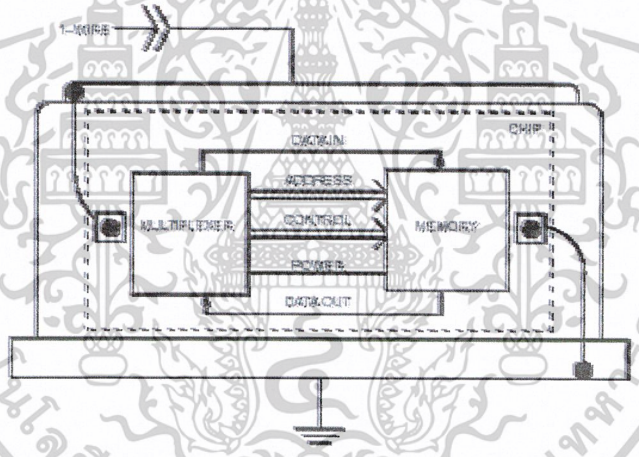
2.1.1 คุณสมบัติร่วมของอุปกรณ์ iButton

- มีรูปร่างตัวถังกลมแบน ขนาดเล็ก เส้นผ่านศูนย์กลาง 16 มิลลิเมตร
- ขั้วต่อทางไฟฟ้าจะแบ่งออกเป็น 2 ส่วนคือ สัญญาณ และกราวด์ โดยหน้าสัมผัสสัญญาณจะอยู่ด้านบนส่วนกราวด์จะอยู่ด้านล่าง รวมถึงพื้นที่ที่เป็นขอบทั้งหมด
- การเชื่อมต่อเพื่ออ่านหรือเขียนข้อมูล ทำได้โดยการสัมผัสโดยตรงกับอุปกรณ์
- หน้าสัมผัสทางไฟฟ้าทั้งหมดทำจากแสตนเลสสตีล จึงทนทานต่อการสัมผัสเชื่อมต่อและไม่เป็นสนิม
- อุปกรณ์ทุกตัวจะมีเลขรหัสประจำตัว ซึ่งใช้เทคโนโลยีเลเซอร์ในการระบุรหัสประจำตัว ทำให้อุปกรณ์ iButton แต่ละตัวจะมีเลขรหัสประจำตัวที่แตกต่างกัน จึงสามารถนำไปใช้งานในด้านการรักษาข้อมูลและความปลอดภัย โดยไม่ต้องมีอุปกรณ์อื่นเพิ่มเติม

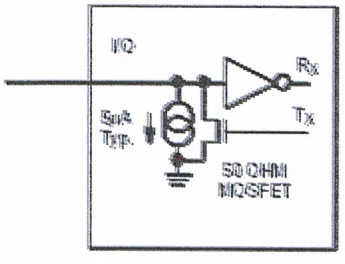
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- รหัสประจำตัวมีขนาด 64 บิต ประกอบด้วย
    1. รหัสของกลุ่มอุปกรณ์หรือตระกูล 8 บิต
    2. รหัสประจำตัว 48 บิต
    3. รหัสตรวจสอบส่วนซ้ำซ้อนแบบวน(Cyclic Redundancy Check) หรือ CRC 8บิต
- ความเร็วในการส่งผ่านข้อมูล 16.3 กิโลบิตต่อวินาที
- สามารถต่อพ่วงกันได้ไม่จำกัดจำนวนบนสายสัญญาณเดียวกัน
- ใช้การเชื่อมต่อผ่านระบบบัส 1 สาย

2.1.2 โครงสร้างของอุปกรณ์ iButton



รูปที่ 2.1 โครงสร้างการทำงานของอุปกรณ์ iButton



รูปที่ 2.2 วงจรเชื่อมต่อภายใน iButton

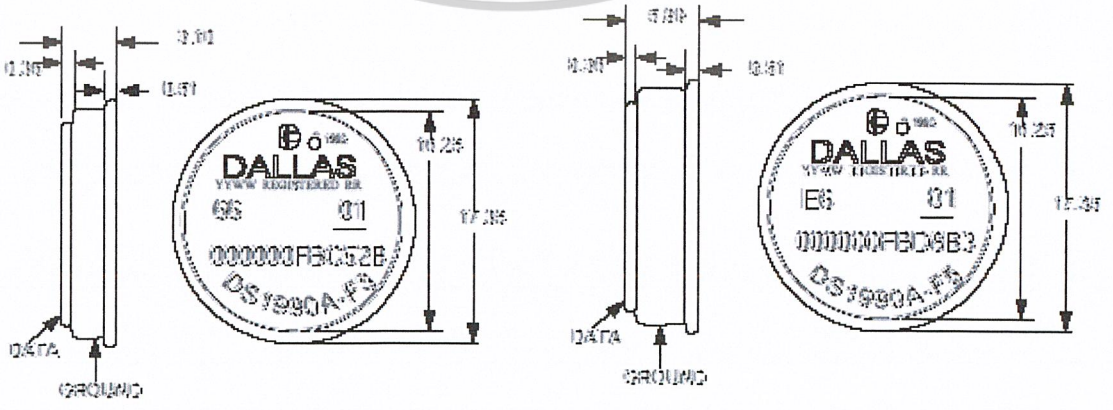
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในรูปที่ 2.1 แสดงโครงสร้างภายในของอุปกรณ์ iButton จะเห็นได้อย่างชัดเจนว่า iButton มีหน้าสัมผัสสำหรับต่อใช้งาน 2 แห่งคือ หน้าสัมผัสสัญญาณซึ่งอยู่ด้านบน และหน้าสัมผัสกราวด์ ซึ่งอยู่ด้านล่างและรอบๆตัว ภายในตัวถังจะบรรจุชิปที่รวมเอาวงจรสำคัญหลัก 2 ส่วน คือ ส่วนวงจรมัลติเพล็กเซอร์ (multiplexer) และ ส่วนของหน่วยความจำ (memory) โดยมัลติเพล็กเซอร์จะเป็นส่วนที่ควบคุมจังหวะการทำงานทั้งหมด สัญญาณที่ส่งมาจากมาสเตอร์บนระบบบัส 1 สายจะถูกส่งผ่านหน้าสัมผัสสัญญาณเข้ามายังวงจรมัลติเพล็กเซอร์จากนั้นมัลติเพล็กเซอร์จะส่งสัญญาณเพื่อเข้าถึงหน่วยความจำภายในของอุปกรณ์เพื่อตรวจสอบว่าสัญญาณที่รับเข้ามานั้นต้องการติดต่อกับตัวอุปกรณ์หรือไม่ โดยสัญญาณที่ส่วนมัลติเพล็กเซอร์ส่งไปยังหน่วยความจำประกอบด้วย สัญญาณแอดเดรส สัญญาณควบคุม และที่สำคัญคือไฟเลี้ยง ถ้าหากการตรวจสอบถูกต้อง อุปกรณ์มาสเตอร์ก็จะอ่านและเขียนข้อมูลกับอุปกรณ์ iButton นี้ได้

ดังนั้นหัวใจสำคัญของอุปกรณ์ iButton คือ ส่วนมัลติเพล็กเซอร์นั่นเอง เพราะต้องทำหน้าที่ทั้งหมดไม่ว่าจะเป็นการจัดเตรียมไฟเลี้ยง ในกรณีที่ติดต่อแบบ Parasite ซึ่งใช้ไฟเลี้ยงจากสายสัญญาณ และการสลับสัญญาณเข้าออกเพื่อเขียนและอ่านข้อมูลกับอุปกรณ์มาสเตอร์

ส่วนสำคัญอีกส่วนหนึ่งของอุปกรณ์ iButton คือ วงจรเชื่อมต่อ จริงอยู่ว่า iButton จะใช้สายสัญญาณเพียงเส้นเดียวในการติดต่อกับไมโครคอนโทรลเลอร์หรืออุปกรณ์มาสเตอร์ แต่โดยแท้จริงแล้วภายในวงจรเชื่อมต่อมีการจัดวงจรเป็น 2 สายสัญญาณคือ สายสัญญาณเข้า (Rx) และสายสัญญาณออก (Tx) ดังแสดงในรูปที่ 2.2 ดังนั้นในการรับส่งสัญญาณจริงๆของอุปกรณ์ iButton จะเหมือนกับอุปกรณ์ในระบบบัส 1 สายตัวอื่นๆ คือ รับส่งข้อมูลแบบฮาล์ฟดูเพล็กซ์ (half duplex) คือ สามารถรับส่งได้ 2 ทิศทาง แต่ต้องไม่พร้อมกัน

2.1.3 รูปร่างตัวถังของอุปกรณ์ iButton



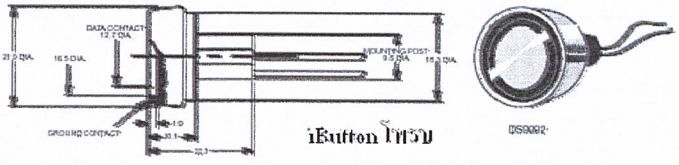
รูปที่ 2.3 ลักษณะตัวถังของอุปกรณ์ iButton

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในรูปที่ 2.3 รูปร่างตัวถังภายนอกของอุปกรณ์ iButton ซึ่งมีด้วยกัน 2 แบบ คือ **F3** และ **F5** โดยทั้งสองแบบจะเหมือนกันคือ มีลักษณะคล้ายกระดุมขนาดเส้นผ่านศูนย์กลาง 16 มิลลิเมตร ตัวถังทำจากสแตนเลสสตีล ความแตกต่างของ F3 กับ F5 คือ ความหนา โดยตัวถังแบบ F3 มีความหนา 3.1 มิลลิเมตร ส่วน F5 มีความหนา 5.89 มิลลิเมตร บนตัวถังของอุปกรณ์ iButton ที่ด้านบนหรือด้านหน้าสัมผัสสัญญาณมีการพิมพ์ข้อมูลสำคัญของอุปกรณ์ iButton ตัวนั้นๆอยู่ด้วย ดังในรูปที่ 2.4

รูปที่ 2.4 ข้อมูลสำคัญของอุปกรณ์ iButton ที่พิมพ์อยู่บนตัวถังด้านหน้าสัมผัสสัญญาณ

จะเห็นว่า พื้นที่ส่วนใหญ่ของตัวถังของอุปกรณ์ iButton จะเป็นกราวด์ โดยส่วนของสัญญาณจะอยู่ทางด้านบน ดังนั้นในการใช้งานจำเป็นอย่างยิ่งที่จะต้องมีอุปกรณ์ช่วย



รูปที่ 2.5 อุปกรณ์ช่วยในการเชื่อมต่ออุปกรณ์ iButton

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.4 คุณสมบัติทางเทคนิคของ DS1990A

- บรรจุเลขรหัสประจำตัวขนาด 48 บิต
- มีรหัสตรวจสอบด้วยส่วนซ้ำซ้อนแบบวน (Cyclic Redundancy Check) หรือ CRC ขนาด 8 บิต
- ใช้เวลาในการเข้าถึงและอ่านข้อมูลน้อยกว่า 5 มิลลิวินาที อุณหภูมิใช้งาน -40 ถึง +85 องศาเซลเซียส
- แรงดันของการพูลอัปต่ำสุด 2.8 V
- ค่าเวลาของไทม์สล็อตต่ำสุด 60 ไมโครวินาที สูงสุด120 ไมโครวินาที
- มีตัวถัง 2 แบบให้เลือกใช้งานทั้งแบบ F3 และ F5

2.1.5 โครงสร้างหน่วยความจำรอมของ DS1990A

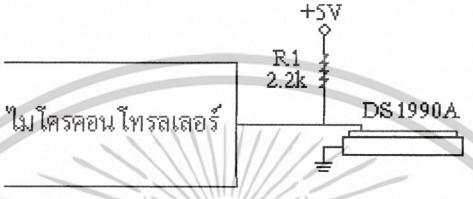
รูปที่ 2.6 โครงสร้างข้อมูลของหน่วยความจำรอมภายใน DS1990A

ในรูปที่ 2.6 โครงสร้างของหน่วยความจำรอมภายใน DS1990A แบ่งเป็น 3 ส่วน คือ รหัสตรวจสอบด้วยส่วนซ้ำซ้อนแบบวน (Cyclic Redundancy Check) หรือ CRC ขนาด 8 บิต หรือ 1 ไบต์จะอยู่ในแอดเดรสสูง ถัดมาคือรหัสประจำตัวขนาด 48 บิต หรือ 6 ไบต์ ข้อมูลตรงส่วนนี้เองที่ทำให้ DS1990A เหมาะสมอย่างยิ่งในการนำไปใช้เป็นกุญแจอิเล็กทรอนิกส์ เนื่องจากจะมีเลขรหัสที่แตกต่างกันมากถึง 2^{48} ซึ่งเท่ากับ 2.81 x 10^{14} ถ้ามีการผลิต DS1990A จำนวนปีละ 1,000 ล้านตัว จะต้องใช้เวลามากกว่า 200 ปี จึงจะใช้เลขรหัสครบหมดทุกตัว ดังนั้นจึงทำให้มั่นใจได้ว่า โอกาสที่จะเกิดเลขรหัสซ้ำนั้นแทบไม่มีเลย ข้อมูลส่วนสุดท้ายคือรหัสตระกูลของอุปกรณ์ มีขนาด 8 บิต หรือ 1 ไบต์ สำหรับ DS1990A มีรหัสตระกูลอุปกรณ์เท่ากับ $01

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.6 การต่อใช้งาน DS1990A

อุปกรณ์ iButton เป็นอุปกรณ์ที่ใช้ในการสื่อสารข้อมูลผ่านระบบบัส 1 สาย ดังนั้นการเชื่อมต่อกับไมโครคอนโทรลเลอร์เพื่อใช้งานจึงสามารถใช้วิธีการเดียวกันได้ โดยทำหน้าสัมผัสสัญญาณเข้ากับขาพอร์ตของไมโครคอนโทรลเลอร์ โดยต้องมีตัวต้านทานต่อพูลอัปไว้ด้วยและต่อหน้าสัมผัสกราวด์เข้ากับกราวด์ของไมโครคอนโทรลเลอร์ ดังในรูปที่ 2.7



รูปที่ 2.7 การต่อใช้งาน DS1990A กับขาพอร์ตของไมโครโทรลเลอร์

อย่างไรก็ตามในทางปฏิบัติจริงการใช้งาน DS1990A และอุปกรณ์ iButton ทุกตัวต้องพึ่งอุปกรณ์เชื่อมต่อด้วย (ดูรูปที่ 2.5)

2.1.7 จังหวะและคำสั่งในการทำงานของ DS1990A

ในการใช้งาน DS1990A จะมีจังหวะการทำงานแบ่งเป็น 3 ช่วง ดังนี้

- เตรียมความพร้อมหรืออินิเชียล (initialization) อุปกรณ์ที่ใช้การเชื่อมต่อผ่านระบบบัส 1 สายทุกตัวจะต้องมีขั้นตอนนี้คือ การอินิเชียลอุปกรณ์ให้พร้อมทำงานและสามารถติดต่อกับอุปกรณ์มาสเตอร์ ซึ่งในที่นี้คือไมโครคอนโทรลเลอร์ โดยไมโครคอนโทรลเลอร์จะส่งสัญญาณพัลส์รีเซตมาเพื่อร้องขอการติดต่อกับอุปกรณ์ หากติดต่อกับอุปกรณ์ iButton ได้ อุปกรณ์ตัวนั้นจะส่งพัลส์ตอบสนองกลับไป

- ส่งคำสั่งติดตั้งต่อหน่วยความจำรอม (ROM Function Command) หลังจากติดต่อกับ DS1990A ได้แล้ว ไมโครคอนโทรลเลอร์จะส่งคำสั่งเพื่อเข้าถึงหน่วยความจำรอมภายใน DS1990A ซึ่งมีด้วยกันเพียง 2 คำสั่ง (แต่ในอุปกรณ์ iButton เบอร์อื่นๆจะมีมากกว่านี้) โดยในแต่ละคำสั่งมีขนาด 8 บิต หรือ 1 ไบต์ อันประกอบด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. คำสั่งอ่านหน่วยความจำรอม (Read ROM $33 หรือ $0F) เป็นคำสั่งอ่านรหัสตระกูล,รหัสประจำตัว 48 บิต และรหัส CRC 8 บิต โดยคำสั่งนี้สามารถใช้ได้ในกรณีที่มี DS1990A เพียงตัวเดียวบนระบบบัสเท่านั้น

2. คำสั่งค้นหาหน่วยความจำรอม (Search ROM : $F0) คำสั่งนี้จะใช้เมื่อมีการต่อ DS1990A มากกว่า 1 ตัวบนระบบบัส โดยมีการทำงาน 3 ขั้นตอนดังนี้

- อ่านข้อมูลแต่ละบิต
- อ่านข้อมูลแบบกลับลอจิก
- เขียนข้อมูลที่ต้องการ

เมื่อกระทำครบทั้งสามขั้นตอนเรียบร้อยแล้ว ไมโครคอนโทรเลอร์จะทราบว่า มี DS1990A ต่ออยู่บนบัสทั้งหมดกี่ตัว แต่ละตัวมีเลขรหัสประจำตัวเป็นเท่าใด

3. อ่านข้อมูล

2.2 โครงสร้างและสถาปัตยกรรมของ PIC 18F458

2.2.1 คุณสมบัติของ PIC 18F458 สามารถสรุปคร่าวๆ ได้ดังนี้

- 35 Instruction คำสั่ง
- ในการปฏิบัติงานคำสั่งต่างๆจะใช้ Cycle เดียวและ 2 Cycle ในคำสั่งที่เป็นการกระโดด
- ความถี่สูงสุดที่ทำงานได้คือ 40MHz
- การทำงานจะเป็นลักษณะ Pipeline ทำให้มีการทำงานที่เร็วขึ้น
- หน่วยความจำโปรแกรม FLASH Program Memory มีขนาด 32k
- หน่วยความจำข้อมูล (RAM) 1536 Bytes
- หน่วยความจำข้อมูล (EEPROM) 256 Bytes
- สามารถตอบสนองการอินเตอร์รัพได้ถึง 14 แหล่ง
- STACK 8 ระดับ
- เพาเวอร์ออนรีเซต(POR), เพาเวอร์อัปไทเมอร์(PWRT) และ Oscillator Start-Up Timer
- Watchdog Timer
- สามารถเลือกการป้องกันข้อมูลได้ (Code Protection)
- โหมดประหยัดพลังงาน (Sleep Mode)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เลือกโหมดของสัญญาณนาฬิกาได้หลายโหมด
- สามารถโปรแกรมโดยใช้แรงดัน +5V ได้
- ฟังก์ชันการโปรแกรมแบบ ICSP (In-Circuit Serial Programming)
- ทำงานที่ไฟเลี้ยง 2.0 ถึง 5.5V
- กระแสทั้งซิงก์และซอร์สของพอร์ตคือ 25mA
- Timer/Counter จำนวน 3 ตัว คือ Timer0,Timer1 และ Timer2
- โมดูล Capture/Compare/PWM จำนวน 2 ชุด
- Analog to Digital Converter ความละเอียด 10 บิต 8 แชนเนลภายในตัว
- มีโมดูลการสื่อสาร USART
- มีโมดูลตรวจจับระดับแรงดันไฟเลี้ยง Brown-out reset(BOR)

มีพอร์ต I/O 5 พอร์ตประกอบด้วย A,B,C,D และ E แต่ละพอร์ทจะมีจำนวนบิตไม่เท่ากัน ซึ่งรวมแล้ว จะมี I/O จำนวน 33บิต

PORTA = RA5-RA0 จำนวน 6 บิต

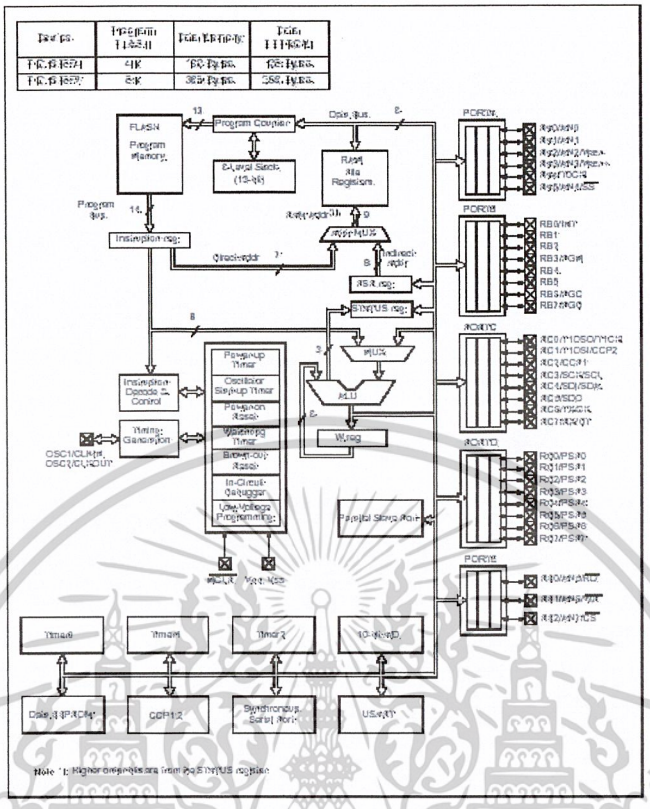
PORTB = RB7-RB0 จำนวน 8 บิต

PORTC = RC7-RC0 จำนวน 8 บิต

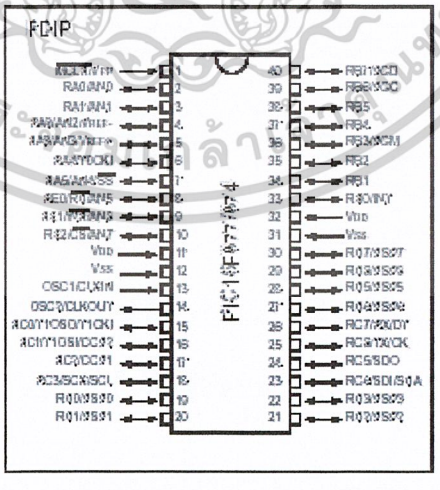
PORTD = RD7-RD0 จำนวน 8 บิต

PORTE = RE2-RE0 จำนวน 3 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 สถาปัตยกรรมภายในของไมโครคอนโทรลเลอร์ PIC 18F458



รูปที่2.9 ขาสัญญาณของ PIC 18F458

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขาสัญญาณของ PIC เบอร์ 18F458 นี้จะมีทั้งหมด 40 ขาจะประกอบไปด้วยขาที่ทำหน้าที่ต่างๆโดยจะมีขาสัญญาณ I/O Ports ทั้งหมดจำนวน 33 ขา โดยสามารถนำไปใช้เป็นอินพุต/เอาต์พุตได้ทั้งหมดทุกขายกเว้นขา RA4 ซึ่งโครงสร้างภายในเป็นแบบ Open Drain ดังนั้นหากต้องการนำไปใช้เป็นขาสัญญาณเอาต์พุตจะต้องต่อตัวต้านทานพูลอัพไว้ด้วยส่วนขาที่เหลือสามารถใช้งานได้ตามปกติ นอกจากขาสัญญาณ I/O แล้วยังประกอบไปด้วยขาสัญญาณอื่นๆอีกคือ ขาไฟเลี้ยง, กราวด์, ขารีเซ็ท และขาออสซิเลเตอร์ ซึ่งสามารถสรุปหน้าที่การทำงานของขาสัญญาณในแต่ละขาได้ดังตารางดังต่อไปนี้

ตารางที่ 2.1 หน้าที่การทำงานของขาสัญญาณในแต่ละขา

ขาสัญญาณ	DIP PIN	I/O/P Type	Buffer Type	Description
OSC1/CLKIN	9	I	ST/CMOS	ขาสัญญาณอินพุตสัญญาณนาฬิกาของ CPU
OSC2/CLKOUT	14	O	-	เป็นขาเอาต์พุตของสัญญาณนาฬิกา (1/4 ของ CLKIN) ใช้ต่อร่วมกับขาสัญญาณ OSC1 เพื่อกำเนิดสัญญาณนาฬิกาในกรณีที่ใช้คริสตอลแบบเรโซเนเตอร์หรือวงจร RC ภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 (ต่อ)

MCLR/VPP	1	I/P	ST	ขาสัญญาณรีเซ็ตแอคทีฟ 0 ขารับแรงดันสำหรับการโปรแกรม
				ขาสัญญาณ อินพุต/เอาต์พุต PORTA
RA0/AN0	2	I/O	TTL	- ขาสัญญาณ อินพุต/เอาต์พุต, ขาสัญญาณอินพุตสัญญาณ
RA1/AN1	3	I/O	TTL	อนาลอก AN0
				- ขาสัญญาณ อินพุต/เอาต์พุต,
RA2/AN2/VREF-	4	I/O	TTL	ขาสัญญาณอินพุตสัญญาณ อนาลอก AN1
				- ขาสัญญาณ อินพุต/เอาต์พุต,
RA3/AN3/VREF+	5	I/O	TTL	ขาสัญญาณอินพุตสัญญาณ อนาลอก AN2 และขาสัญญาณ แรงดันอ้างอิง ลบ(-)
RA4/TOCKI	6	I/O	TTL	- ขาสัญญาณ อินพุต/เอาต์พุต,
				ขาสัญญาณอินพุตสัญญาณ
RA5/SS/AN4	7	I/O	TTL	อนาลอก AN3 และขาสัญญาณ แรงดันอ้างอิง บวก(+)
				- ขาสัญญาณอินพุต/เอาต์พุต, ขาสัญญาณอินพุตสัญญาณนาฬิกาของ TIMER0
				- ขาสัญญาณอินพุต/เอาต์พุต, ขาสัญญาณอินพุตสัญญาณ อนาลอก AN4 และขาสัญญาณ Slave Select ในโหมดการสื่อสารแบบ Synchronous Serial Port

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่2.1 (ต่อ)

				ขาสัญญาณ อินพุต/เอาต์พุต PORTB
RB0/INT	33	I/O	TTL/ST	- ขาสัญญาณ อินพุต/เอาต์พุต, ขาอินพุตสัญญาณอินเตอร์รัพท์ภายนอก
RB1	34	I/O	TTL	
RB2	35	I/O	TTL	- ขาสัญญาณ อินพุต/เอาต์พุต
RB3/PGM	36	I/O	TTL	- ขาสัญญาณ อินพุต/เอาต์พุต
RB4	37	I/O	TTL	- ขาสัญญาณ อินพุต/เอาต์พุต, ขาสัญญาณการโปรแกรมแบบแรงดันต่ำ
RB5	38	I/O	TTL	- ขาสัญญาณ อินพุต/เอาต์พุต, ขาสัญญาณอินเตอร์รัพท์จากการเปลี่ยนแปลงสถานะของขาสัญญาณ
RB6/PGC	39	I/O	TTL/ST[2]	- ขาสัญญาณ อินพุต/เอาต์พุต, ขาสัญญาณอินเตอร์รัพท์จากการเปลี่ยนแปลงสถานะของขาสัญญาณ
RB7/PGD	40	I/O	TTL/ST[2]	- ขาสัญญาณ อินพุต/เอาต์พุต, ขาสัญญาณอินเตอร์รัพท์จากการเปลี่ยนแปลงสถานะของขาสัญญาณ และเป็นขาสัญญาณนาฬิกาในโหมดการโปรแกรม - ขาสัญญาณ อินพุต/เอาต์พุต, ขาสัญญาณอินเตอร์รัพท์จากการเปลี่ยนแปลงสถานะของขาสัญญาณ และเป็นขาสัญญาณนาฬิกาในโหมดการโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่2.1 (ต่อ)

				ขาสัญญาณ อินพุต/เอาต์พุต PORTC
RC0/T1OSO/T1CKI	33	I/O	ST	- ขาสัญญาณ อินพุต/เอาต์พุต, ขาสัญญาณออสซิเลเตอร์เอาต์พุต และขาอินพุตสัญญาณนาฬิกาของ TIMER1
RC1/T1OSI/CCP2	16	I/O	ST	
RC2/CCP1	17	I/O	ST	- ขาสัญญาณ อินพุต/เอาต์พุต, ขาสัญญาณอินพุตออสซิเลเตอร์ TIMER1 และเป็นขาสัญญาณ Capture2 Input/Compare2 Output/PWM2 Output
RC3/SCK/SCL	18	I/O	ST	- ขาสัญญาณ อินพุต/เอาต์พุต, และเป็นขาสัญญาณ Capture1 Input/Compare1 Output/PWM1 Output
RC4/SDI/SDA	23	I/O	ST	- ขาสัญญาณ อินพุต/เอาต์พุต, ขาสัญญาณอินพุตสัญญาณนาฬิกาในการสื่อสาร Synchronous, ขาสัญญาณนาฬิกาในโหมด I^2C และ SPI
RC5/SDO	24	I/O	ST	
RC6/TX/CK	25	I/O	ST	- ขาสัญญาณ อินพุต/เอาต์พุต, ขาอินพุตสัญญาณข้อมูลในโหมด SPI, ขาอินพุต/เอาต์พุตสัญญาณข้อมูลในโหมด I^2C
RC7/RX/DT	26	I/O	ST	- ขาสัญญาณ อินพุต/เอาต์พุต, ขาเอาต์พุตสัญญาณข้อมูลในโหมด SPI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่2.1 (ต่อ)

				ขาสัญญาณ อินพุต/เอาต์พุต PORTD สามารถใช้เป็น Slave Port กรณีติดต่อกับระบบบัสของไมโครโปรเซสเซอร์
RD0/PSP0	19	I/O	ST/TTL[3]	- ขาสัญญาณ อินพุต/เอาต์พุต
RD1/PSP1	20	I/O	ST/TTL[3]	- ขาสัญญาณ อินพุต/เอาต์พุต
RD2/PSP2	21	I/O	ST/TTL[3]	- ขาสัญญาณ อินพุต/เอาต์พุต
RD3/PSP3	22	I/O	ST/TTL[3]	- ขาสัญญาณ อินพุต/เอาต์พุต
RD4/PSP4	27	I/O	ST/TTL[3]	- ขาสัญญาณ อินพุต/เอาต์พุต
RD5/PSP5	28	I/O	ST/TTL[3]	- ขาสัญญาณ อินพุต/เอาต์พุต
RD6/PSP6	29	I/O	ST/TTL[3]	- ขาสัญญาณ อินพุต/เอาต์พุต
RD7/PSP7	30	I/O	ST/TTL[3]	- ขาสัญญาณ อินพุต/เอาต์พุต
				ขาสัญญาณ อินพุต/เอาต์พุต PORTE
RE/RD/AN5	8	I/O	ST/TTL[3]	- ขาสัญญาณ อินพุต/เอาต์พุต, ขาสัญญาณควบคุมการอ่านในโหมด Parallel Slave Port และเป็นขาสัญญาณอินพุตสัญญาณอนาลอก AN5
RE1/WR/AN6	9	I/O	ST/TTL[3]	- ขาสัญญาณ อินพุต/เอาต์พุต, ขาสัญญาณควบคุมการเขียนในโหมด Parallel Slave Port และเป็นขาสัญญาณอินพุตสัญญาณอนาลอก AN6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ตารางที่2.1(ต่อ)

RE2/CS/AN7	10	I/O	ST/TTL[3]	- ขาสัญญาณ อินพุต/เอาต์พุต, ขาสัญญาณควบคุมคอนโทรล Select ในการติดต่อแบบ Parallel Slave Port และเป็นขาสัญญาณอินพุตสัญญาณอนาลอก AN7
Vss VDD	12,31 11,32	P P	- -	ขาสัญญาณกราวด์ ขาแรงดันไฟเลี้ยง

ตารางที่2.2 ตารางสรุปหน้าที่การทำงานของขาสัญญาณในแต่ละขา

I = อินพุต	O = เอาต์พุต	I/O = อินพุต/เอาต์พุต	P = Power
- = ไม่ได้ใช้	TTL = ระดับสัญญาณ TTL	ST = วงจรชมิททริกเกอร์อินพุต	

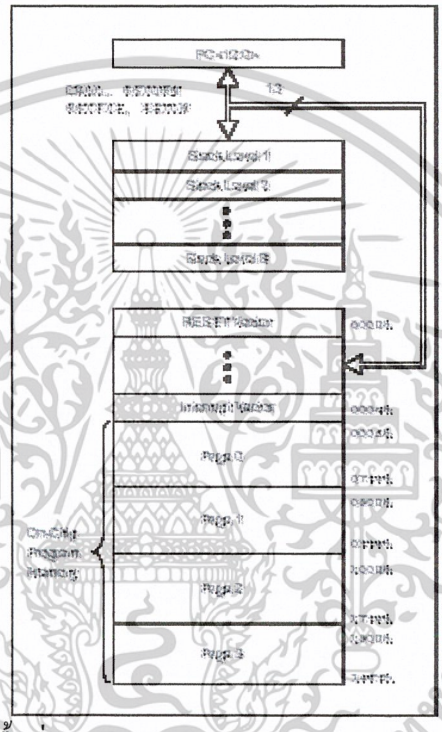
Note

1. Buffer จะเป็นแบบ ชมิททริกเกอร์ เมื่อมีการใช้งานอินเตอร์รัพท์ภายนอก
2. Buffer จะเป็นแบบ ชมิททริกเกอร์ เมื่อทำงานในโหมดการโปรแกรม
3. Buffer จะเป็นแบบ ชมิททริกเกอร์ เมื่อใช้งานทั่วๆไปและจะเป็นแบบ TTL เมื่ออยู่ในโหมด Parallel Slave Port ในการติดต่อกับระบบบัสของไมโครโปรเซสเซอร์
4. Buffer จะเป็นแบบ ชมิททริกเกอร์ เมื่อใช้งานในโหมด RC oscillator และจะเป็นแบบ CMOS เมื่อใช้งานอื่นๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
62172

2.2.2 หน่วยความจำโปรแกรม

หน่วยความจำโปรแกรม (Program Memory) เป็นพื้นที่สำหรับใช้ในการเก็บซอร์สโค้ดโปรแกรม โดยโครงสร้างเป็นแบบ Flash memory จึงสามารถทำการเขียนและลบได้หลายครั้งทำให้สะดวกต่อการทดลองพัฒนาโปรแกรม โดยจะมี Program Counter ขนาด 13 บิต ซึ่งสามารถอ้างถึงตำแหน่งข้อมูลได้ถึง 8 กิโลเวิร์ด ตั้งแต่แอสเดรส 0000h ถึง 1FFFh ดังรูป



รูปที่2.10 การวางพื้นที่ของหน่วยความจำโปรแกรมของ PIC 18F458

ซึ่งหน่วยความจำโปรแกรม ของ PIC 18F458 นี้จะมีขนาด 32K จะแบ่งออกเป็น 4 Page จำนวน Page ละ 2 Kwords ดังจะเห็นได้จากรูป จะมี reset เวกเตอร์อยู่ที่ตำแหน่ง 0000h และอินเตอร์รัพท์เวกเตอร์อยู่ที่ 0004h จะเห็นว่าจะมีแอดเดรสเวกเตอร์ของการอินเตอร์รัพท์เพียงตำแหน่งเดียวแต่ไมโครคอนโทรลเลอร์ PIC 18F458 นั้นมีแหล่งกำเนิดสัญญาณอินเตอร์รัพท์ถึง 14 แหล่ง ซึ่งเมื่อเกิดการอินเตอร์รัพท์จากแหล่งใดก็ตาม CPU จะกระโดดไปทำงานในตำแหน่งแอดเดรสตัวเดียวนั่นก็คือ แอดเดรส 0004h ดังนั้นเราจึงไม่สามรถจัดลำดับความสำคัญของการอินเตอร์รัพท์ได้ การตรวจสอบแหล่งที่มาของสัญญาณอินเตอร์รัพท์ จะใช้วิธีการตรวจสอบบิตสถานะของการอินเตอร์รัพท์ต่างๆ ส่วนหน่วยความจำ สแต็ก (Stack) จะมีระดับความลึก 8 ระดับ และไม่สามารถเข้าถึงได้โดยตรงจากการเขียนโปรแกรม ไม่มีคำสั่งการ PUSH-POP เหมือนกับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CPU ตระกูลอื่นๆ ค่าของ STACK จะมีการเปลี่ยนแปลงอัตโนมัติ เมื่อมีการใช้คำสั่งที่เป็นการกระโดด, การเรียกโปรแกรมย่อย หรือ เมื่อมีการอินเตอร์รัพท์เกิดขึ้นเท่านั้น

2.2.3 หน่วยความจำข้อมูล (Data Memory)

หน่วยความจำข้อมูลนี้ จะประกอบไปด้วยพื้นที่ใช้งานทั่วไป (General Purpose Register) ขนาด 368 Bytes และพื้นที่ของรีจิสเตอร์ฟังก์ชั่นพิเศษ (Special Function Register) ซึ่งพื้นที่ของหน่วยความจำเหล่านี้จะถูกแบ่งออก 4 แบงก์ การเข้าถึงข้อมูลในแต่ละส่วนเหล่านี้จะต้องกำหนดแบงก์ข้อมูลที่ต้องการเข้าถึง โดยการกำหนดค่าในรีจิสเตอร์ที่ทำหน้าที่เลือกแบงก์ คือ RP0 และ RP1 โดยจะอยู่ในรีจีสเตอร์ STATUS บิตที่ 5 และ 6 ตามลำดับ

2.2.4 รีจิสเตอร์

เราจะแยกรีจิสเตอร์ออกเป็น 2 ส่วนคือ รีจิสเตอร์ไฟล์ หรือ รีจิสเตอร์ใช้งานทั่วไป (General Purpose Register) และรีจิสเตอร์ฟังก์ชั่นพิเศษ (Special Function Register)

2.2.4.1 รีจิสเตอร์ใช้งานทั่วไป (General Purpose Register) เป็นหน่วยความจำใช้งานทั่วไป โดยโครงสร้างจะเป็นหน่วยความจำชนิด Static Ram สามารถเปลี่ยนแปลงข้อมูลได้ตลอดเวลา และข้อมูลต่างๆจะหายไปเมื่อไม่มีไฟเลี้ยงจ่ายให้กับ CPU หน่วยความจำส่วนนี้ของ CPU PIC 18F458 จะมีขนาด 368 Byte ข้อมูลจะเป็นแบบ 8 บิต เหมาะสำหรับการใช้เก็บข้อมูล หรือใช้เป็นตัวแปรต่างๆในการเขียนโปรแกรมโดยจะกระจายอยู่ในแบงก์ต่างๆทั้ง 4 แบงก์

2.2.4.2 รีจิสเตอร์ฟังก์ชั่นพิเศษ (Special Function Register) เป็นรีจิสเตอร์ที่ใช้ในการควบคุมการทำงานในฟังก์ชั่นต่างๆของ CPU โดยจะจัดวางลักษณะเดียวกันกับรีจิสเตอร์ใช้งานทั่วไป คือ จะอยู่ภายในแต่ละแบงก์ โดยเราจะแบ่งออกเป็นรีจิสเตอร์หลักและรีจิสเตอร์ที่ใช้งานในฟังก์ชั่นต่างๆ

2.3ระบบสื่อสารข้อมูลอนุกรมแบบหนึ่งสาย (1- Wire™ Serial Bus)

ผู้คิดค้นระบบการสื่อสารข้อมูลแบบนี้ คือ ดัลลัสเซมิคอนดักเตอร์ ดังนั้นในบางครั้งจึงเรียกระบบสื่อสารข้อมูลแบบนี้ว่าระบบสื่อสารข้อมูลดัลลัสหนึ่งสาย (The Dallas 1-Wire Bus) ระบบสื่อสารข้อมูลแบบนี้เป็นระบบที่มีความชาญฉลาด และใช้จำนวนสายสัญญาณเพียง 1 สายเท่านั้น โดยไม่ต้องมีสายสัญญาณนาฬิกามาควบคุมจังหวะการถ่ายทอดข้อมูล เหมือนกับระบบสื่อสารข้อมูลอนุกรมในแบบอื่นๆเนื่องจากสายข้อมูลนั้นจะทำหน้าที่เสมือนหนึ่งเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

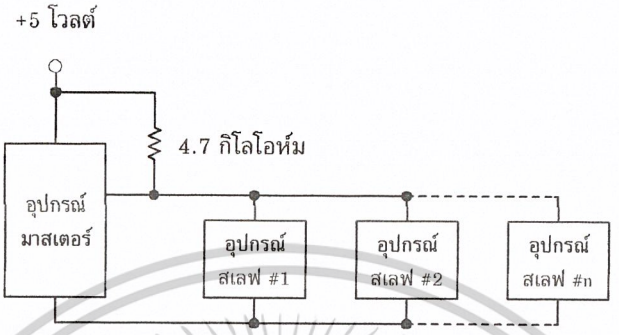
สายสัญญาณนาฬิกาในตัว ส่วนค่าของข้อมูลจะพิจารณาจากลักษณะของรูปสัญญาณที่ปรากฏบนสายสัญญาณในแต่ละช่องของเวลา (Time-Slot) โดยที่คาบเวลาต่ำสุดและสูงสุดของสถานะต่างๆที่ใช้ในการสื่อสารข้อมูลในแต่ละช่องของเวลามีการกำหนดขอบเขตไว้อย่างชัดเจน การถ่ายทอดข้อมูลจะเกิดขึ้นในแต่ละช่องของเวลานั้นรูปแบบการถ่ายทอดข้อมูลจะเป็นแบบอะซิงโครนัสในระดับบิต ไม่มีการกำหนดความยาวของข้อมูลเป็นระดับไบต์ ระบบสื่อสารนี้เหมาะที่จะใช้ในการสื่อสารข้อมูลระหว่างไอซีบนแผงวงจรเดียวกัน หรือสร้างเป็นโครงข่ายสื่อสารแบบสายคู่ตีเกลียวก็ได้

2.3.1 คุณสมบัติทางเทคนิคของระบบบัสหนึ่งสาย

สายสัญญาณบนระบบบัสแบบหนึ่งสายนี้จะเป็นสายสัญญาณแบบสองทิศทางแต่ข้อมูลจะสามารถเดินทางได้ในทิศทางเดียวภายในช่วงเวลาหนึ่งๆนั่นคือมีลักษณะคล้ายกับระบบสื่อสารแบบครึ่งดูเพลกซ์ (Half Duplex) ตัวอย่างที่สามารถเห็นได้ชัดเจนคือการใช้งานวิทยุสื่อสารหรือวิทยุสมัครเล่น อุปกรณ์บนระบบบัสต้องมีการระบุชัดเจนว่าตัวใดเป็นอุปกรณ์ตัวหลักตัวใดเป็นอุปกรณ์ตัวรับโดยส่วนใหญ่อุปกรณ์ตัวหลักคือ ไมโครคอนโทรลเลอร์ ส่วนอุปกรณ์ตัวรับได้แก่ ไอซีตรวจจับอุณหภูมิหรือไอซีหน่วยความจำแรม เป็นต้น อุปกรณ์ตัวหลักจะเป็นตัวจัดเตรียมความพร้อมของสายสัญญาณและควบคุมการถ่ายทอดข้อมูลบนสายสัญญาณนั้น ข้อมูลทั้งหมด ไม่ว่าจะเป็นข้อมูลควบคุมหรือข้อมูลใช้งานจะถูกส่งบนสายสัญญาณที่มีอยู่เพียงเส้นเดียวทั้งหมด ในระหว่างการทำงานอุปกรณ์ตัวหลักและตัวรับสามารถที่จะเป็นได้ทั้งตัวส่งและตัวรับ ขึ้นอยู่กับเงื่อนไขของการทำงานในขณะนั้น ยกตัวอย่างเช่น ถ้าหากมีการเขียนข้อมูลจากอุปกรณ์ตัวรับ ตัวส่งคืออุปกรณ์ตัวหลัก ตัวรับคืออุปกรณ์สเลฟ ในทางตรงข้ามหากเป็นการอ่านข้อมูลจากอุปกรณ์ตัวรับ ตัวส่งจะกลายเป็นอุปกรณ์ตัวรับและตัวรับคืออุปกรณ์ตัวหลัก ในระบบบัสหนึ่งระบบต้องมีอุปกรณ์ตัวหลักเพียงตัวเดียวเท่านั้น

สายสัญญาณของระบบบัสนี้ ต้องกำหนดสภาวะปกติไว้ที่ลอจิกสูง สามารถทำได้โดยการต่อตัวต้านทานค่าประมาณ 4.7 กิโลโอห์ม ต่อร่วมกับกับแหล่งจ่ายแรงดันไฟฟ้า +5 โวลต์ ดังนั้นอุปกรณ์ที่นำเข้ามาต่อบนระบบบัสนี้จึงต้องออกแบบให้ภาคเอาต์พุตที่ต้องต่อกับ สายสัญญาณมีลักษณะเป็นคอลเล็กเตอร์หรือเดรนเปิดในรูปที่ 2.4 แสดงแผนภาพของการสื่อสารข้อมูลอนุกรมแบบหนึ่งสายเบื้องต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.11 การเชื่อมต่อบนระบบบัสหนึ่งสาย

2.3.2 คุณสมบัติของช่องเวลา (Timeslot)

อุปกรณ์ตัวหลักจะเป็นอุปกรณ์เพียงตัวเดียวบนระบบบัสหนึ่งสายนี้ ที่สามารถจะทำการกำหนดสภาพเริ่มต้นสายสัญญาณได้ โดยอุปกรณ์ตัวหลักจะกำเนิดจุดเริ่มต้นของช่องเวลาด้วยการทำให้สายสัญญาณเป็นลอจิกต่ำในช่วงเวลาหนึ่งจากนั้นก็จะทำให้กลับมาเป็นลอจิกสูง ถ้าอุปกรณ์ตัวรับต้องการส่งข้อมูลมายังอุปกรณ์ตัวหลัก อุปกรณ์ตัวรับจะเป็นตัวควบคุมสภาวะของสายสัญญาณต่อไปจนเสร็จสิ้นกระบวนการแต่ถ้าหากอุปกรณ์ตัวหลักต้องการส่งข้อมูลก็จะสามารถดำเนินการต่อไปได้เลย

ฟังก์ชั่นของช่องเวลาที่กำหนดโดยอุปกรณ์ตัวหลักมีด้วยกัน 4 ฟังก์ชั่นคือช่องเวลาของการรีเซต (Reset) การอ่านข้อมูล (Read Data) การเขียนข้อมูล “1” (Write One) และ การเขียนข้อมูล “0” (Write Zero) ส่วนช่องเวลารีเซตใช้ในการเริ่มต้นติดต่อกับอุปกรณ์ตัวรับ ในขณะที่ช่องเวลาการอ่านจะใช้สำหรับอ่านข้อมูลที่ส่งมาจากอุปกรณ์ตัวรับและในส่วนช่องเวลา การเขียนข้อมูล “1” และ “0” สำหรับเขียนข้อมูลไปยังอุปกรณ์ตัวรับผ่านสายสัญญาณของระบบ

ทางด้านอุปกรณ์ตัวรับมีฟังก์ชั่นของช่องเวลาอยู่ทั้งสิ้น 3 ฟังก์ชั่น คือ ช่องเวลาของการตอบสนอง (Presence) การเขียนข้อมูล “1” (Write One) และการเขียนข้อมูล “0” (Write Zero) ซึ่งช่องเวลาของการตอบสนองจะใช้สำหรับตอบสนองการติดต่อจากอุปกรณ์ตัวหลักโดย อุปกรณ์ตัวรับตัวที่ถูกเลือกจากอุปกรณ์ตัวหลักจะต้องส่งสัญญาณตอบสนองลงบนสายสัญญาณเพื่อแจ้งให้อุปกรณ์ตัวหลักทราบว่าขณะนี้สามารถติดต่อกันได้แล้วส่วนช่องเวลาการเขียนข้อมูล “1” และ “0”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับส่งข้อมูลไปยังอุปกรณ์ตัวหลักผ่านสายสัญญาณของระบบ ซึ่งจะสัมพันธ์กับช่องเวลาการอ่านข้อมูลของอุปกรณ์ตัวหลัก

การแยกแยะฟังก์ชั่นของแต่ละช่องเวลาจะใช้ความยาวของคาบเวลาและลักษณะของรูปสัญญาณเป็นตัวกำหนดและทุกครั้งที่มีการเปลี่ยนแปลงฟังก์ชั่นต้องทำให้สายสัญญาณอยู่ในสภาวะว่างเสมอซึ่งก็คือการทำให้สายสัญญาณเป็นลอจิกสูงอย่างน้อยเป็นเวลา 1 ไมโครวินาที

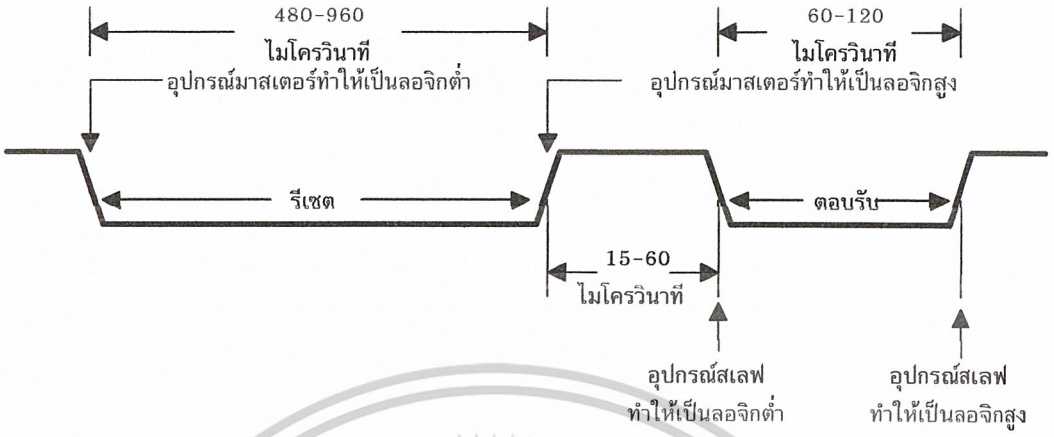
2.3.3 ช่องเวลาการรีเซตและตอบสนอง

อุปกรณ์ตัวหลักทำให้เกิดการรีเซตบนสายสัญญาณเพื่อแจ้งแก่อุปกรณ์ตัวรับโดย การทำให้สายสัญญาณเปลี่ยนเป็นลอจิกต่ำต่อเนื่องอย่างน้อย 480 ไมโครวินาทีและจะต้องทำให้สายสัญญาณกลับมาเป็นลอจิกสูงภายใน 480 ไมโครวินาที จากนั้นถ้าหากมีอุปกรณ์ตัวรับต่ออยู่บนสายสัญญาณ จะมีการตอบสนองสัญญาณรีเซตนั้นด้วยสัญญาณตอบสนอง (Presence) โดย การทำให้สายสัญญาณเปลี่ยนเป็นลอจิกต่ำต่อเนื่องนานประมาณ 60-240 ไมโครวินาที หลังจาก สัญญาณรีเซตปรากฏประมาณ 15–60 ไมโครวินาที ในรูปที่ 2.11 แสดงช่องเวลาของการรีเซตและการตอบสนอง

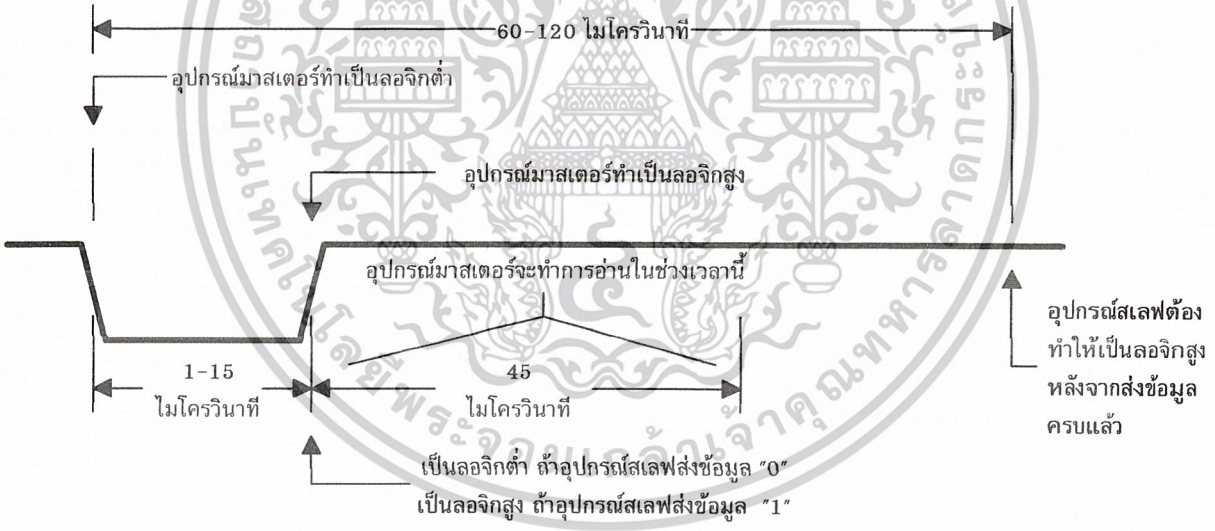
2.3.4 ช่องเวลาการอ่านข้อมูลของอุปกรณ์ตัวหลักและการเขียนข้อมูลของอุปกรณ์ตัวรับ

เมื่อต้องการที่จะอ่านข้อมูลจากอุปกรณ์ตัวรับ อุปกรณ์ตัวหลักจะทำให้สายสัญญาณ เป็นลอจิกต่ำประมาณ 1-15 ไมโครวินาที จากนั้นต้องทำให้สถานะของสายกลับมาเป็นลอจิกสูง อุปกรณ์ตัวรับจะต้องส่งข้อมูลมาให้อุปกรณ์ตัวหลักโดยถ้าข้อมูลเป็น “0” อุปกรณ์ตัวรับจะทำให้สายสัญญาณเป็นลอจิกต่ำนานประมาณ 45 ไมโครวินาทีแล้วทำให้สายสัญญาณกลับมาสู่สภาวะลอจิกสูงอีกครั้ง แต่ถ้าเป็นข้อมูล “1” อุปกรณ์ตัวรับจะทำให้สายสัญญาณเป็นลอจิกสูงต่อเนื่องอีก 45 ไมโครวินาที รวมระยะเวลาทั้งหมดในช่องเวลานี้ประมาณ 60-120 ไมโครวินาที นั่นคือในช่องเวลานี้จะต้องใช้เวลารวมไม่เกิน 120 ไมโครวินาที ในขณะที่อุปกรณ์ตัวหลักจะใช้เวลา ในการอ่านข้อมูลอยู่ระหว่าง 15 และ 60 ไมโครวินาที หลังจากเริ่มต้นช่องเวลานี้ในรูปที่ 2.12 สัญญาณของช่องเวลาการอ่านข้อมูลของอุปกรณ์ตัวหลักซึ่งก็จะมีลักษณะเหมือนกับการเขียนข้อมูลของอุปกรณ์ตัวรับและช่องเวลาทั้งสองจะเกิดขึ้นในช่วงเวลาเดียวกัน กล่าวคือเมื่ออุปกรณ์ตัวหลักอ่านอุปกรณ์ตัวรับก็ต้องทำการเขียนข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.12 ช่องเวลาการรีเซตและการตอบรับของอุปกรณ์บนระบบบัสหนึ่งสาย



รูปที่ 2.13 ช่องเวลาการอ่านข้อมูลของอุปกรณ์ตัวหลักซึ่งตรงกับช่องเวลาการเขียนข้อมูลของอุปกรณ์ตัวรับ

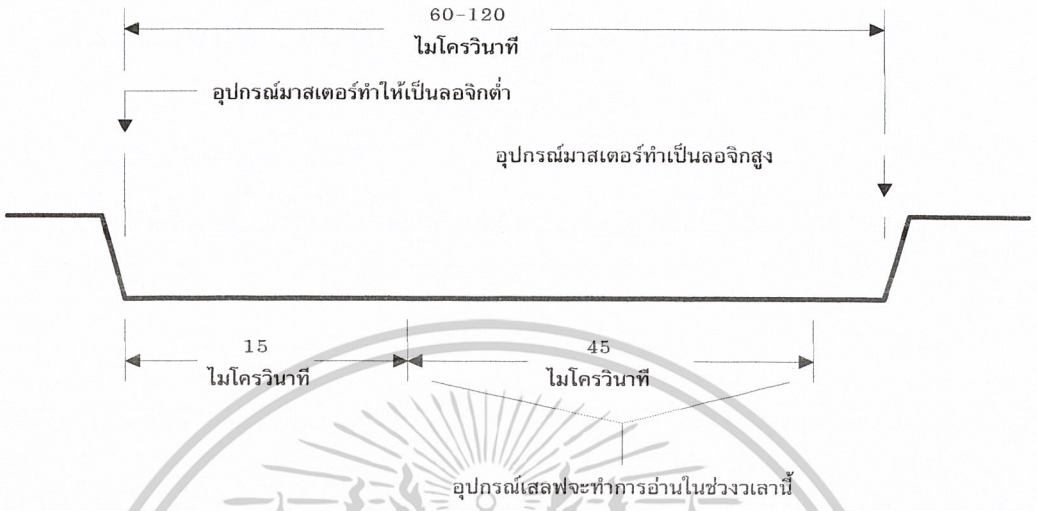
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.5 ช่องเวลาการเขียนข้อมูลของอุปกรณ์ตัวหลัก

เมื่ออุปกรณ์ตัวหลักต้องการเขียนข้อมูล อุปกรณ์ตัวหลักก็จะต้องทำให้สายสัญญาณ เป็นลอจิกต่ำประมาณ 1- 15 ไมโครวินาที จากนั้นต้องทำให้สถานะของสายกลับมาเป็นลอจิกสูงแล้วดำเนินการเขียนข้อมูลได้ทันที ถ้าข้อมูลที่ต้องการเขียนไปยังอุปกรณ์ตัวรับเป็น "0" อุปกรณ์ตัวหลักจะทำให้สายสัญญาณเป็นลอจิกต่ำนานประมาณ 45 ไมโครวินาทีแล้วทำให้สายสัญญาณกลับมาสู่สภาวะลอจิกสูงอีกครั้งดังรูปที่ 2.13 แต่ถ้าต้องการเขียนข้อมูล "1" อุปกรณ์ตัวหลักจะ ทำให้สายสัญญาณเป็นลอจิกสูงต่อเนื่องไปอีก 45 ไมโครวินาที รวมเวลาทั้งหมดในช่องเวลานี้ประมาณ 60-120 ไมโครวินาที ในรูปที่ 2.14 แสดงรูปสัญญาณของช่องเวลาการเขียนข้อมูลของอุปกรณ์ตัวหลักซึ่งก็จะมีลักษณะเหมือนกับการอ่านข้อมูลของอุปกรณ์ตัวรับและช่องเวลาทั้งสองจะเกิดขึ้นในช่วงเวลาเดียวกันกล่าวคือเมื่ออุปกรณ์ตัวหลักเขียนอุปกรณ์ตัวรับก็ต้องทำการอ่านข้อมูล

รูปที่ 2.14 ช่องเวลาการเขียนข้อมูล "1" ของอุปกรณ์ตัวหลักซึ่งตรงกับช่องเวลาการอ่านข้อมูลของอุปกรณ์ตัวรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.15 ช่องเวลาการเขียนข้อมูล “0” ของอุปกรณ์ตัวหลัก

2.3.6รูปแบบของการสื่อสารข้อมูลแบบหนึ่งสาย(1-Wirecommunication Protocol)

ในการติดต่อสื่อสารข้อมูลในระบบบัสหนึ่งสาย อุปกรณ์ตัวหลักสามารถติดต่อกับอุปกรณ์ตัวรับได้ครั้งละ 1 ตัวเท่านั้น ดังนั้นอุปกรณ์ตัวรับแต่ละตัวต้องมีข้อมูลกำหนดตำแหน่งเฉพาะตัวโดยจะเก็บไว้ในหน่วยความจำรอมภายในอุปกรณ์ตัวรับตัวนั้นๆ โดยปกติอุปกรณ์ตัวรับในระบบบัสหนึ่งสายของดัลลัสนี้จะมีหน่วยความจำขนาด 64 บิตหรือ 8 ไบต์ สำหรับเก็บข้อมูลต่างๆที่สำคัญของอุปกรณ์แต่ละตัว ซึ่งประกอบด้วย

1. รหัสของตระกูล จำนวน 8 บิต
2. เลขหมายประจำตัว (Serial Number) จำนวน 48 บิต
3. รหัสตรวจสอบความผิดพลาด (CRC : Cyclical Redundancy Check) จำนวน 8 บิต

ผู้ใช้งานสามารถที่จะอ่านข้อมูลประจำตัวของอุปกรณ์ตัวรับได้ ด้วยการใช้คำสั่งอ่านหน่วยความจำรอม (Read ROM) ในกรณีที่บนสายสัญญาณมีอุปกรณ์ตัวรับอยู่เพียงตัวเดียว ไม่จำเป็นต้องอ้างตำแหน่งในการติดต่อ

รูปแบบการติดต่อบนระบบบัสหนึ่งสายจะเริ่มต้นขึ้น เมื่ออุปกรณ์ตัวหลักทำการรีเซตและกำหนดตำแหน่งของอุปกรณ์ที่ทำการติดต่อ ถ้าหากมีอุปกรณ์ตัวรับเพียงตัวเดียวสามารถข้ามขั้นตอนการติดต่อกับหน่วยความจำรอมในอุปกรณ์สเลฟได้ จะเรียกวิธีการดังกล่าวว่าการไม่ติดต่อหน่วยความจำรอมหรือ สคิป รอม (Skip ROM) จากนั้นรอการตอบรับจากอุปกรณ์ตัวรับเมื่อการตอบรับสมบูรณ์ก็สามารถเริ่มต้นขั้นตอนการอ่านหรือเขียนข้อมูลได้ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 ทฤษฎีการติดต่อสื่อสารระหว่างคอมพิวเตอร์และอุปกรณ์ภายนอกผ่านพอร์ตอนุกรม

การที่จะเคลื่อนย้ายข้อมูลจากคอมพิวเตอร์ไปยังอุปกรณ์ต่อพ่วงอื่นๆหรือคอมพิวเตอร์ด้วยกัน มีช่องทางการติดต่อการรับส่งข้อมูลแบบขนานและการรับส่งข้อมูล แบบอนุกรม การรับส่งข้อมูลแบบขนานจะเป็นการรับหรือส่งข้อมูลครั้งละ 4 บิตหรือ 8 บิต ในเวลาเดียวกัน ซึ่งจะทำให้การรับและส่งข้อมูลทำได้ที่ความเร็วสูง หมายความว่าจำนวนของสาย ที่ใช้ในการส่งต้องมีมากเท่ากับจำนวนบิตของข้อมูลที่จะส่งก็ได้ ซึ่งเป็นปัญหาในเรื่องราคาของสายที่ใช้ในการเชื่อมต่อแบบขนานมักจะมีราคาแพง

ในขณะที่การรับส่งข้อมูลแบบอนุกรมเป็นการส่งข้อมูลครั้งละ 1 บิต แต่ก็สามารถ รับส่งข้อมูลได้คราวละหลายๆบิตได้ หากแต่จะต้องมีการตกลงกันระหว่างตัวส่งและตัวรับว่าจะรับส่งข้อมูลคราวละกี่บิต ตัวรับจะต้องรอข้อมูลมาให้ครบทุกบิตเสียก่อนจึงทำการประมวลผล การส่งข้อมูลแบบอนุกรมจะใช้จำนวนสายที่น้อยกว่ามาก อย่างน้อยที่สุดใช้เพียง 2-3 เส้นเท่านั้น แต่อัตราเร็วในการรับส่งข้อมูลยังช้ากว่าแบบขนาน อย่างไรก็ตามในการรับส่งข้อมูลแบบอนุกรมสามารถใช้สายสัญญาณที่มีความยาวมากกว่าแบบขนาน ทำให้ระยะทางในการสื่อสารข้อมูลแบบอนุกรมสามารถทำได้มากกว่า

2.4.1 การสื่อสารแบบอนุกรม

การสื่อสารแบบอนุกรมแบ่งออกได้เป็น 2 แบบคือ การสื่อสารอนุกรมแบบซิงโครนัส(Synchronous) และการสื่อสารอนุกรมแบบอะซิงโครนัส (Asynchronous)

2.4.1.1 การสื่อสารแบบซิงโครนัส

การสื่อสารข้อมูลแบบสัมพันธ์มีสัญญาณนาฬิการ่วมอยู่กับการรับและส่งสัญญาณ ตัวอย่างการส่งข้อมูลแบบซิงโครนัสคือแป้นพิมพ์ (Key Board) ของคอมพิวเตอร์ ซึ่งสายเส้นหนึ่งจะเป็นสายของสัญญาณนาฬิกา ส่วนสายอีกเส้นหนึ่งจะเป็นสายของข้อมูลดังนั้นการติดต่อแบบซิงโครนัสนี้จะต้องใช้สายในการเชื่อมต่ออย่างน้อยที่สุดสามเส้นคือ สัญญาณนาฬิกา ข้อมูลและกราวด์ (Ground)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.1.2 การสื่อสารข้อมูลแบบอะซิงโครนัส

การสื่อสารข้อมูลแบบอะซิงโครนัสคือการรับและส่งข้อมูลไปในสายโดยไม่จำเป็นต้องมีสัญญาณนาฬิการ่วมด้วยเหมือนกับการรับส่งข้อมูลแบบซิงโครนัส แต่จะใช้การกำหนดค่าสัญญาณนาฬิกาทั้งภาครับและภาคส่งให้มีค่าเท่ากัน ซึ่งเรียกสัญญาณนาฬิกาที่ใช้ในการกำหนดค่าให้ภาครับและภาคส่งนี้ว่า อัตราการถ่ายทอดข้อมูล หรือ บอดเรต (Baud Rate) มีหน่วยเป็นบิตต่อวินาที (Bit Per Second : bps)

รูปแบบของข้อมูลที่ใช้ในการรับส่งแบบอะซิงโครนัสประกอบด้วย 4 ส่วนคือ

บิตเริ่มต้น (Start Bit) ซึ่งจะมีขนาด 1 บิต

บิตข้อมูลแบบอนุกรมจะมีขนาด 5,6,7 หรือ 8 บิต

บิตตรวจสอบพาริตี้ (Parity Bit) จะมีขนาด 1 บิตหรือไม่มี

บิตปิดท้าย (Stop Bit) จะมีขนาด 1,1.5 หรือ 2 บิต

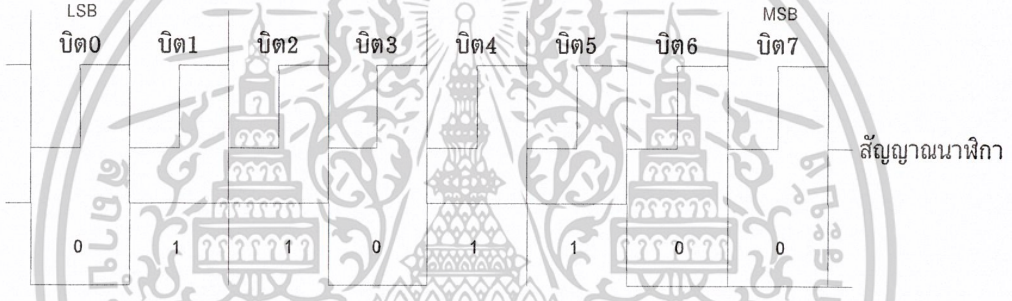
รูปที่ 2.14 รูปแบบของข้อมูลอนุกรมแบบอะซิงโครนัสซึ่งเมื่อไม่มีข้อมูลจะส่ง ลอจิก 0 ด้วยช่วงระยะเวลา 1 บิต ซึ่งจะเรียกบิตนี้ว่าบิตเริ่มต้นจากนั้นบิตข้อมูลจะถูกส่งออกไป โดย เริ่มจากบิตที่มีนัยสำคัญต่ำที่สุด (Lower Side Band : LSB) ก่อนซึ่งข้อมูลในไบต์ที่จะส่ง อาจจะมีจำนวนบิต 5 , 6 , 7 หรือ 8 บิตก็ได้จากนั้นจะตามด้วยพาริตี้บิตซึ่งใช้เพื่อตรวจสอบ ความผิดพลาดที่เกิดขึ้นจากการส่งข้อมูลบิตสุดท้ายที่จะส่งคือบิตปิดท้ายจะทำให้ขารับส่งข้อมูล มีสถานะลอจิก 1 อีกครั้งด้วยระยะเวลาอย่างน้อย 1 บิต 1.5 บิตหรือ 2 บิต เพื่อเป็นการแสดงว่าสิ้นสุดข้อมูลแล้ว

อุปกรณ์พิเศษที่ได้รับการออกแบบมาสำหรับการรับและส่งข้อมูลแบบอะซิงโครนัส เรียกว่า Universal Asynchronous Receiver/Transmitter หรืออุปกรณ์ที่ทำหน้าที่รับและส่งข้อมูลแบบอะซิงโครนัส อัตราความเร็วในการรับและส่งข้อมูลแบบอะซิงโครนัสคือ ค่าบอดเรต ซึ่งก็คือค่าจำนวนบิตต่อวินาทีที่ใช้ในการรับและส่งข้อมูลบอดเรตมาตรฐานที่ใช้สำหรับพอร์ตอนุกรม RS-232 ได้แก่อัตรา 110 , 150 , 300 , 600 , 1200 , 2400 , 4800 , 9600 และ 19200 บิตต่อวินาที และมีค่าเพิ่มมากขึ้นตามเทคโนโลยีของคอมพิวเตอร์ซึ่งการรับส่งแบบอนุกรมโดยที่ไม่ผ่านโมเดมอาจจะสามารถกำหนดค่าบอดเรตได้สูงถึง 115,200 บิตต่อวินาที เนื่องจากบอดเรตคือจำนวนบิตของข้อมูลที่สามารถถ่ายทอดได้ภายใน 1 วินาที ยกตัวอย่างข้อมูลอนุกรมถูกส่ง ในลักษณะ 8 บิต ไม่มีการตรวจสอบพาริตี้ (Parity) มีบิตเริ่มต้น 1 บิต และบิตปิดท้าย 1 บิต ความยาวของข้อมูลที่รับส่งนี้เท่ากับ 10 บิต ถ้าใช้บอดเรตในการส่งข้อมูลเท่ากับ 9600 บิต ต่อวินาที ก็จะสามารถรับส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ได้ด้วยความเร็ว 960 ไบต์ต่อวินาที และถ้ามีการใช้พาริตี้ความเร็วในการรับส่งข้อมูลจะเหลือเป็น 872 ไบต์ต่อวินาที

การตรวจสอบพาริตี้สามารถกำหนดให้เป็นแบบคี่ (Odd) แบบคู่ (Even) หรือไม่มี การตรวจสอบพาริตี้ก็ได้ การตรวจสอบพาริตี้เป็นการตรวจสอบจำนวนรวมของบิตที่เป็นลอจิก 1 ภายในข้อมูลที่ส่งไป 1 ไบต์ ว่ามีจำนวนรวมเป็นเลขคู่หรือเลขคี่โดยต้องรวมบิตพาริตี้เข้าไปด้วย ยกตัวอย่างข้อมูลที่จะทำการส่งมีขนาด 8 บิตและมีค่าเท่ากับ 99 ฐานสิบหกหรือ 10011001 ฐานสอง จะเห็นว่าข้อมูลในไบต์นี้มีจำนวนลอจิก 1 จำนวน 4 ตัวซึ่งเป็นเลขคู่ ดังนั้นถ้ากำหนดค่า พาริตี้เป็นคู่ค่าในบิตพาริตี้จะต้องมีลอจิกเป็น 0 แต่ถ้าพาริตี้เป็นคี่ ค่าที่บิตพาริตี้จะต้องเป็น 1 เพื่อให้ข้อมูล 1 ไบต์รวมทั้งบิตพาริมีจำนวนบิตที่เป็นลอจิก 1 มีจำนวนรวมกันเป็นเลขคี่



รูปที่ 2.16 รูปแบบอย่างง่ายที่สุดของข้อมูลอนุกรมแบบอะซิงโครนัส

บิตพาริตี้ของข้อมูลถูกสร้างขึ้นจากภาคส่งข้อมูลของอุปกรณ์ที่ทำหน้าที่รับและส่ง ข้อมูลแบบอะซิงโครนัสซึ่งทางภาครับจะต้องทำการกำหนดคุณสมบัติการตรวจสอบพาริตี้ ให้ตรงกันว่าจะตรวจสอบพาริตี้คี่หรือว่าพาริตี้คู่ จากนั้นภาครับของอุปกรณ์ที่ทำหน้าที่รับและ ส่งข้อมูลแบบอะซิงโครนัสจะทำการตรวจสอบค่าพาริตี้ที่เกิดขึ้นว่าเป็นคู่หรือเป็นคี่ โดยการนับจำนวน 1 ทั้งหมดรวมทั้งบิตพาริตี้ด้วย ถ้ากำหนดพาริตี้ไว้เป็นคู่แต่อ่านค่าตัวเลขในการนับออก มาได้ตัวเลขเป็นคี่ทางภาครับจะแสดงข้อผิดพลาดออกมาให้ผู้ใช้ได้ทราบ นับเป็นการตรวจสอบ ความผิดพลาดที่เกิดขึ้นในการถ่ายทอดข้อมูลที่ง่ายที่สุด แต่จะเชื่อถือได้ก็ต่อเมื่อมีบิตข้อมูล ที่ทำการส่งผิดพลาดเพียงบิตเดียวเท่านั้น ถ้าข้อมูลที่ทำการส่งมีบิตที่ผิดพลาดมากกว่า 1 บิต การตรวจสอบด้วยวิธีนี้จะไม่ได้ผลสำหรับการตั้งพาริตี้บิตเป็น NONE นั้นทั้งภาครับและภาคส่ง จะไม่มีการตรวจสอบพาริตี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 มาตรฐานพอร์ตอนุกรม RS-232

มาตรฐานการเชื่อมต่อแบบอนุกรม RS-232 เป็นมาตรฐานอุตสาหกรรมที่ออกแบบมาเพื่อใช้ในการส่งข้อมูลอนุกรมแบบอะซิงโครนัส 2 ทิศทาง โดยมาตรฐาน RS-232 ในอดีตนั้นถูกออกแบบมาเพื่อการส่งผ่านข้อมูลจากคอมพิวเตอร์ไปยังโมเดม (Modulator-Demodulator) เพียงอย่างเดียว เพื่อที่จะนำข้อมูลจากโมเดมนี้สื่อสารโดยผ่านสายโทรศัพท์ไปยังคอมพิวเตอร์ อีกชุดซึ่งอยู่ห่างไกลกัน โดยคณะกรรมการที่เรียกว่า สมาคมอุตสาหกรรมอิเล็กทรอนิกส์ (Electronic Industries Association : EIA) ได้วางมาตรฐานที่มีชื่อเรียกว่า EIA RS-232 มาตรฐานนี้ในช่วงแรกจะใช้หัวต่อ (Connector) เป็นแบบ DB-25 โดยกำหนดความยาวสูงสุดของสายสัญญาณไว้ที่ 50 ฟุต มีระดับแรงดันสัญญาณตั้งแต่ –3 โวลต์ ถึง –12 โวลต์ แสดงว่ามีข้อมูล และ +3 โวลต์ ถึง +12 โวลต์ แสดงว่าเป็นช่องไม่มีข้อมูล

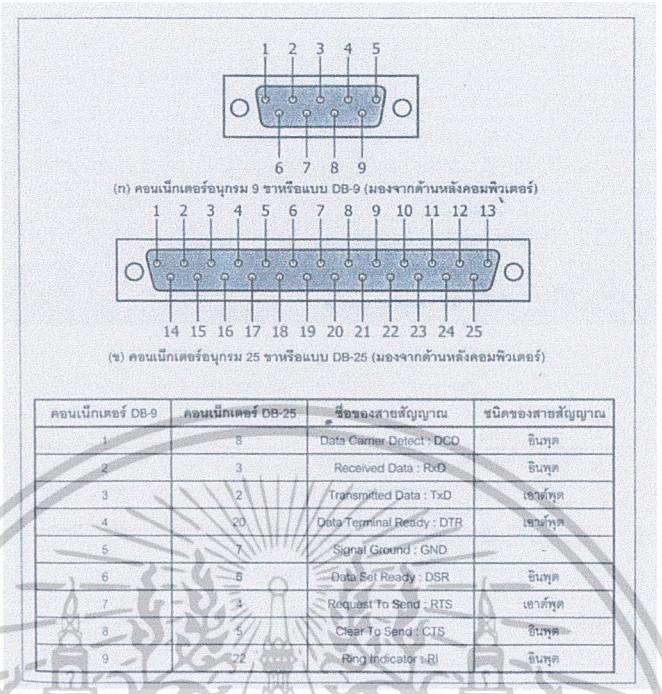
มาตรฐาน RS-232 ได้กำหนดรูปแบบของอุปกรณ์เชื่อมต่อข้อมูล (Data Terminal Equipment : DTE) กับวงจรข้อมูลปลายทาง (Data Circuit Terminating : DCE) ไว้ว่าอุปกรณ์เชื่อมต่อข้อมูล จะต้องเป็นอุปกรณ์ที่มีการประมวลผลในตัวเช่น ไมโครคอนโทรลเลอร์มีความสามารถในการสร้างบิตข้อมูลแบบอนุกรมได้ วงจรข้อมูลปลายทางจะทำหน้าที่เป็นเพียง ตัวรับส่งข้อมูลที่ส่งมาจากอุปกรณ์เชื่อมต่อข้อมูลเท่านั้น โดยการรับส่งข้อมูลระหว่างอุปกรณ์ ทั้งสองจะกระทำผ่านมาตรฐาน RS-232

สำหรับการใช้งานบนคอมพิวเตอร์ พอร์ตอนุกรม RS-232 มักถูกใช้เชื่อมต่อกับ โมเดมหรือเมาส์ (Mouse) โดยสามารถรับส่งข้อมูลได้ที่ความยาวของสายสัญญาณสูงสุดถึง 20เมตร

2.6 หัวต่อสำหรับพอร์ต RS-232 และการเชื่อมต่อ

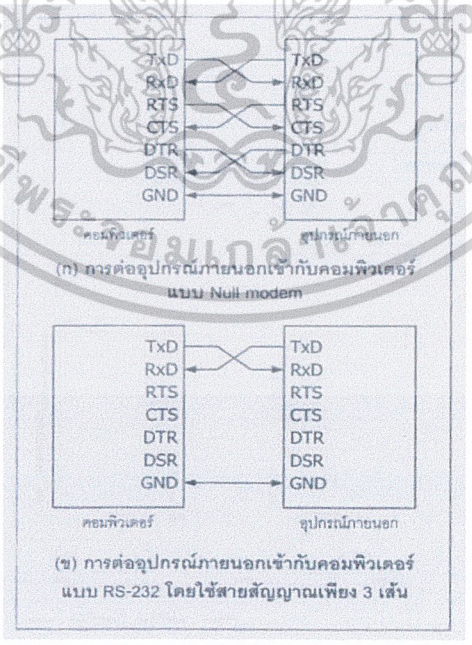
มาตรฐานการเชื่อมต่อแบบ RS-232 จะใช้หัวต่อแบบ DB-25 หรือ DB-9 หัวต่อแบบ DB-25 จะมีขาต่อใช้งานเพียง 9 ขา ลักษณะเดียวกันกับหัวต่อ DB-9 เนื่องจากขาอื่นๆ ที่เคยใช้งานในอดีต ปัจจุบันมีการใช้งานไม่มากนักจึงถูกยกเลิกไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



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

รูปที่ 2.17 การจัดขาของคอนเน็คเตอร์พอร์ทอนุกรมตามมาตรฐานRS-232ทั้งแบบ DB-9 และ DB-25



รูปที่ 2.18 การต่ออุปกรณ์ภายนอกกับพอร์ทอนุกรมของคอมพิวเตอร์ในลักษณะต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับการเชื่อมต่อคอมพิวเตอร์กับอุปกรณ์ภายนอกแสดงดังในรูปที่ 2.16 ลูกศรในรูปแสดงถึงทิศทางของข้อมูล ในรูปที่ 2.16 (ก) เป็นการเชื่อมต่อแบบ Null modem หรือการเชื่อมต่อโดยตรงโดยไม่ต้องผ่านโมเด็ม โดยมีการตรวจสอบ หรือแฮนด์เช็กเต็มรูปแบบ ส่วนในรูปที่ 2.16 (ข) เป็นการเชื่อมต่อแบบ RS-232 ในลักษณะที่ใช้สายสัญญาณเพียง 3 เส้น โดยเส้นหนึ่งสำหรับส่งข้อมูล อีกเส้นสำหรับรับข้อมูล และเส้นสุดท้ายเป็นกราวด์

2.6.1 รายละเอียดหน้าที่การทำงานในแต่ละขาของพอร์ตอนุกรม RS-232

Data Carrier Detect : DCD ขานี้จะถูกกระตุ้นเมื่อมีการส่งสัญญาณพาห์ จากอุปกรณ์สื่อสารข้อมูลเช่น โมเดม เป็นต้น สำหรับการใช้งานปกติ ขานี้จะไม่ได้ถูกใช้งานมากนัก

Receive Data : RD หรือ RxD ขานี้ใช้เพื่อรับสัญญาณอนุกรมเข้ามายังคอมพิวเตอร์ โดยนำข้อมูลที่อ่านได้เก็บไว้ในรีจิสเตอร์ บัฟเฟอร์ (Register Buffer)

Transmitted Data : TD หรือ TxD ขานี้ใช้เพื่อส่งข้อมูลออกจากคอมพิวเตอร์ โดยนำข้อมูลที่เก็บอยู่ในบัฟเฟอร์สำหรับส่งข้อมูลส่งออกไป

Data Terminal Ready : DTR เป็นขาสัญญาณที่ส่งออกจากคอมพิวเตอร์เพื่อให้อุปกรณ์ปลายทางรับรู้ว่า ต้องการติดต่อด้วย โดยขา DTR นี้จะต้องเชื่อมต่อกับขา DTS ของอุปกรณ์ปลายทางและที่ขา DTR ของอุปกรณ์ปลายทางจะต้องทำการเชื่อมต่อกับขา DSR ของคอมพิวเตอร์ถ้าใช้การเชื่อมต่อเป็นแบบไม่ใช้สัญญาณตรวจสอบ ซึ่งใช้สายในการเชื่อมต่อเพียง 3 เส้น จะต้องต่อขา DTR และ DSR ของตัวมันเองเข้าด้วยกันและต้องต่อกับขา DCD ด้วยในกรณีที่โปรแกรมสื่อสารที่ใช้มีการตรวจจับสัญญาณพาห์

Signal Ground : GND ขากราวด์ของระบบ

Data Set Ready : DSR ขานี้จะใช้คู่กับขา DTR เพื่อตรวจสอบการเชื่อมต่อกันระหว่างคอมพิวเตอร์กับอุปกรณ์ปลายทาง ซึ่งขา DSRนี้จะเป็นขาสำหรับรับข้อมูลจากภายนอก ซึ่งถูกส่งมาจากขา DTR

Request To Send : RTS เป็นขาสำหรับส่งสัญญาณร้องขอให้ทางอุปกรณ์ปลายทางส่งข้อมูลกลับมายังคอมพิวเตอร์ โดยขาที่รับสัญญาณ RTS ก็คือขา CTS ในกรณีที่ใช้ลักษณะการเชื่อมต่อแบบ Null modem 3 สาย จะต้องทำการเชื่อมต่อขา RTS และ CTS ของตัวมันเอง เข้าด้วยกันเพื่อจะให้การรับและส่งข้อมูลสามารถเกิดขึ้นได้ตลอดเวลา

Clear To Send : CTS ขานี้จะรับสัญญาณจากขา RTS เมื่อรับสัญญาณได้ ข้อมูลที่ขา TxD จะถูกส่งออกไป ดังนั้นขานี้จึงถูกใช้เพื่อตรวจสอบอุปกรณ์ต่อพ่วงว่าพร้อมที่จะรับข้อมูลหรือไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Ring Indicator : RI ใช้แสดงสถานะสัญญาณเรียกจากชุมสายโทรศัพท์ปกติ ในการสื่อสารโดยทั่วไปสายนี้ถูกใช้งานต่อเมื่อมีการเชื่อมต่อกับโมเดมและโปรแกรม มีการตรวจสอบสัญญาณนี้เท่านั้น

2.7 อุปกรณ์ที่ทำหน้าที่รับและส่งข้อมูลแบบอะซิงโครนัส (Universal Asynchronous Receiver/Transmitter)

สำหรับการสื่อสารอนุกรมบนคอมพิวเตอร์แล้วอุปกรณ์ที่ทำหน้าที่รับและส่งข้อมูลแบบอะซิงโครนัสถือว่าเป็นหัวใจสำคัญของการสื่อสารอนุกรม หน้าที่หลักของอุปกรณ์ที่ทำหน้าที่รับและส่งข้อมูลแบบอะซิงโครนัสคือทำหน้าที่แปลงข้อมูลที่อยู่ในรูปแบบขนานจากคอมพิวเตอร์ให้อยู่ในรูปแบบอนุกรมแบบอะซิงโครนัสแล้วส่งออกไป และทำหน้าที่แปลงสัญญาณอนุกรมแบบอะซิงโครนัสที่ป้อนเข้ามายังอุปกรณ์ที่ทำหน้าที่รับและทำการส่งข้อมูลแบบอะซิงโครนัสให้เป็นแบบขนานเสียก่อนจะส่งเข้าสู่คอมพิวเตอร์นอกจากนี้อุปกรณ์ที่ทำหน้าที่รับและส่งข้อมูลแบบอะซิงโครนัสจะส่งข้อมูลไปยังคอมพิวเตอร์แล้วยังแจ้งข้อมูลความผิดพลาดระหว่าง ทำการถ่ายทอดข้อมูลผิดพลาดจากพาริตี้ เฟรมข้อมูล การทับซ้อนข้อมูลอีกด้วย

ภายในอุปกรณ์ที่ทำหน้าที่รับและส่งข้อมูลแบบอะซิงโครนัส จะมีส่วนของวงจรสร้างบอดเรตแบบโปรแกรมได้ (Programmable Buadrate generator) โดยการกำหนดค่าตัวหารให้กับสัญญาณนาฬิกาของอุปกรณ์ที่ทำหน้าที่รับและส่งข้อมูลแบบอะซิงโครนัส โดยตัวหารนี้มีขนาด 16 บิต ดังนั้นจึงสามารถกำหนดตัวหารอยู่ในช่วง 1-65,535 และยังสามารถรับส่งข้อมูลได้ทั้งแบบ ครึ่งดูเพลกซ์ (Half Duplex) และแบบดูเพลกซ์เต็ม (Full Duplex) การส่งแบบครึ่งดูเพลกซ์ เป็นการส่งแบบทิศทางเดียว ส่วนการส่งแบบดูเพลกซ์เต็มนั้นสามารถรับส่งข้อมูลได้ใน คราวเดียวกัน

2.7.1 ชนิดของอุปกรณ์ที่ทำหน้าที่รับและส่งข้อมูลแบบอะซิงโครนัส

ในเครื่องคอมพิวเตอร์ทั่วไปมีอุปกรณ์ที่ทำหน้าที่รับและส่งข้อมูลแบบอะซิงโครนัสที่ใช้งานกันอยู่ 2 เบอร์คือ ไอซีเบอร์ 8250 ซึ่งเป็นอุปกรณ์มาตรฐานที่ใช้กันมายาวนาน อุปกรณ์นี้มีบัฟเฟอร์สำหรับรับและส่งข้อมูลตำแหน่งเดียวกัน ทำให้การรับและส่งข้อมูลถูกจำกัดความเร็ว อยู่ที่ 5706 กิโลบิตต่อวินาทีเท่านั้น อุปกรณ์เบอร์นี้ก็ถือว่าเป็นต้นแบบของอุปกรณ์ที่ทำหน้าที่ รับและส่งข้อมูลแบบอะซิงโครนัสที่ใช้ในคอมพิวเตอร์

อุปกรณ์ที่ทำหน้าที่รับและส่งข้อมูลแบบอะซิงโครนัสอีกเบอร์หนึ่งคือไอซีเบอร์ 16450 มีความสามารถรับส่งข้อมูลที่ความเร็ว 115,200 บิตต่อวินาทีและเพิ่มรีจิสเตอร์พักข้อมูลสำหรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์ที่ทำหน้าที่รับและส่งข้อมูลแบบอะซิงโครนัส นอกจากนั้นแล้วยังเพิ่มส่วนของชิปรีจิสเตอร์แบบเข้าก่อนออกก่อน (First In First Out) ขนาด 16 ไบต์เข้าไป ทำให้สามารถสนับสนุนความเร็วในการรับส่งข้อมูลที่ 256 กิโลบิตต่อวินาทีได้ โดยคอมพิวเตอร์ในปัจจุบันใช้ อุปกรณ์เบอร์นี้หรือใหม่กว่าเช่นไอซีเบอร์ TL16C750 ซึ่งมีรีจิสเตอร์แบบเข้าก่อนออกก่อนขนาด64 ไบต์ ทำงานได้ที่ระดับแรงดันไฟฟ้าบวก 5 โวลต์ และแรงดันไฟฟ้าบวก 3 โวลต์ มีโหมดประหยัดพลังงาน สามารถรับส่งข้อมูลได้ที่ความเร็ว 1 เมกะบิตต่อวินาทีเมื่อใช้สัญญาณนาฬิกา 16 เมกะเฮิรตซ์

อย่างไรก็ตามความเร็วในการส่งข้อมูลที่มากมายของอุปกรณ์ที่ทำหน้าที่รับและส่ง ข้อมูลแบบอะซิงโครนัสเบอร์ใหม่ๆ ไม่ได้ช่วยให้การรับส่งข้อมูลของคอมพิวเตอร์เร็วขึ้น เนื่องจากว่าคอมพิวเตอร์ที่ใช้งานปัจจุบันยังใช้ความถี่ของสัญญาณนาฬิกาในการแปลงข้อมูลเพียง 1.8432 เมกะเฮิรตซ์ เท่านั้น

2.7.2 วงจรภายในและรีจิสเตอร์ของพอร์ตอนุกรม RS-232

เครื่องคอมพิวเตอร์โดยทั่วไปสามารถต่อพอร์ตอนุกรม RS-232 สูงสุดได้ 4 พอร์ต ซึ่งจะมีชื่อเรียกเป็น COM1 COM2 COM3 และ COM4 ซึ่งพอร์ตอนุกรมแต่ละตัวต่างก็ใช้งานอุปกรณ์ที่ทำหน้าที่รับและส่งข้อมูลแบบอะซิงโครนัส ภายในคอมพิวเตอร์ในการติดต่อกับอุปกรณ์ภายนอกเช่นเดียวกัน

การทำงานภายในของพอร์ตอนุกรม จะประกอบไปด้วย รีจิสเตอร์ขนาด 8 บิต 8 ตัว ที่ใช้งานร่วมกับอุปกรณ์ที่ทำหน้าที่รับและส่งข้อมูลแบบอะซิงโครนัสตำแหน่งที่อยู่ของรีจิสเตอร์ภายในพอร์ตอนุกรมสามารถคำนวณได้จากค่ารีจิสเตอร์พื้นฐานของพอร์ตอนุกรม

2.7.3 ลักษณะสัญญาณอินพุตและเอาต์พุตของพอร์ตอนุกรม RS–232

สัญญาณเอาต์พุตที่ใช้ควบคุมรีจิสเตอร์ RTS และรีจิสเตอร์ DTR และสัญญาณแสดงสถานะอินพุตรีจิสเตอร์ CTS DSR และรีจิสเตอร์ DCD ของพอร์ตอนุกรม RS–232 จะถูกกลับสถานะภายในตัวของอุปกรณ์ที่ใช้ทำหน้าที่รับและส่งข้อมูลแบบอะซิงโครนัสซึ่งส่วนสัญญาณข้อมูลทั้งภาคส่งและรับจะไม่ถูกกลับสถานะ ดังนั้นเมื่อสัญญาณถูกส่งออกมาจาก อุปกรณ์ที่ทำหน้าที่รับและส่งข้อมูลแบบอะซิงโครนัส จึงต้องส่งเข้าสู่วงจรขับเพื่อทำการปรับระดับแรงดันไฟฟ้าให้ได้ ระดับสัญญาณเป็นไปตามมาตรฐาน RS–232 ก่อนส่งออกไปจากคอมพิวเตอร์สำหรับอุปกรณ์ ต่อเชื่อมปลายทางก็จะต้องมีวงจรขับในลักษณะนี้เช่นเดียวกันเพื่อให้ได้ระดับของสัญญาณในระดับเดียวกันในขณะที่วงจรขับที่ใช้ทั้งภายในคอมพิวเตอร์และอุปกรณ์ต่อเชื่อมปลายทางนั้นจะถูกกลับสถานะตำแหน่งที่อยู่ของพอร์ตอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตำแหน่งที่อยู่พื้นฐานของพอร์ตอนุกรมมี 4 ตำแหน่งดังนี้ คือ

COM 1 : 3F8H COM 3 : 3E8H

COM 2 : 2F8H COM 4 : 2E8H

เมื่อเริ่มเปิดใช้งานคอมพิวเตอร์ ไบออสภายในคอมพิวเตอร์จะทำการตรวจสอบตำแหน่งที่อยู่ของพอร์ตอนุกรมทั้งหมด ถ้าหากไบออสตรวจพบตำแหน่งที่อยู่ของพอร์ตอนุกรม ไบออสจะนำตำแหน่งที่อยู่ที่ตรวจพบไปเก็บไว้ในหน่วยความจำขนาด 2 ไบต์ สำหรับพอร์ตอนุกรม COM 1 จะเก็บไว้ที่ตำแหน่งที่อยู่ 0000:0400H และ 0000:0401H พอร์ตอนุกรม
COM 2 เก็บไว้ที่ตำแหน่งที่อยู่ 0000:0402H และ 0000:0403H พอร์ตอนุกรม COM 3 จะ เก็บไว้ที่ตำแหน่ง 0000:0404H และ 0000:0405H พอร์ตอนุกรม COM 4 จะเก็บไว้ที่ตำแหน่ง 0000:0406 และ 0000:0407 ในหน่วยความจำ

2.8ทฤษฎีทั่วไปเกี่ยวกับโปรแกรมวิชวลเบสิก

ในปัจจุบันระบบปฏิบัติการ (Operating System) ในลักษณะของวินโดวส์(Windows) ได้เข้ามาแทนที่ระบบปฏิบัติการในลักษณะเดิมซึ่งส่วนใหญ่ที่นิยมใช้กันอยู่คือ ระบบปฏิบัติการ MS-DOS เนื่องจากรูปแบบของจอภาพที่ใช้ติดต่อระหว่างคอมพิวเตอร์และผู้ใช้อยู่ในรูปแบบของ Graphic User Interface (GDI) ที่ใช้รูปภาพแทนคำสั่งต่างๆแทนซึ่งต่างจากระบบปฏิบัติการ MS-DOS ที่รูปแบบของคำสั่งจะอยู่ในรูปแบบของตัวอักษรและเป็นแบบป้อน ทีละบรรทัดหรือที่เรียกว่า “Command Line” ซึ่งผู้ใช้จะต้องเรียนรู้และจดจำรูปแบบของแต่ละ คำสั่งให้ถูกต้องและแม่นยำ จึงจะใช้งานโปรแกรมนั้นๆได้เป็นอย่างดีและด้วยเหตุนี้ได้ส่งผลต่อการพัฒนาโปรแกรมเช่นเดียวกันเนื่องจากโปรแกรมเมอร์ ซึ่งแต่เดิมพัฒนาโปรแกรมอยู่บน ระบบปฏิบัติการ MS-DOS ต้องเปลี่ยนแปลงรูปแบบและแนวความคิดและหันมาพัฒนาโปรแกรมบนวินโดวส์แทน

การพัฒนาโปรแกรมบนวินโดวส์ในปัจจุบันกระทำได้ง่ายและสะดวกขึ้น เนื่องจากมีการใช้เทคโนโลยีทางด้านเสมือนจริง (Visualize) เข้ามาประกอบในการออกแบบจอภาพ ซึ่งต่างจากในยุคแรกที่การพัฒนาโปรแกรมบนวินโดวส์นั้นค่อนข้างจะทำได้ยาก เนื่องจาก การพัฒนาโปรแกรมหนึ่งๆให้แล้วเสร็จ โปรแกรมเมอร์จะต้องเขียนแผนงาน (Routine) ที่จะ ทำขึ้นเป็นจำนวนมาก ซึ่งโปรแกรมวิชวลเบสิกก็จัดเป็นภาษาหนึ่งที่ได้รับความนิยมและถูกนำมาใช้ในการพัฒนาโปรแกรมเพื่อใช้งานบนวินโดวส์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8.1 สาเหตุที่ต้องใช้โปรแกรมวิชวลเบสิก

โปรแกรมวิชวลเบสิกเป็นภาษาคอมพิวเตอร์ที่ได้รับความนิยมนำมาใช้ในการพัฒนาโปรแกรมบนวินโดวส์ เนื่องจากเป็นภาษาคอมพิวเตอร์ที่ใช้เทคโนโลยีด้านเสมือนจริงซึ่งเพียงแต่เลือกส่วนควบคุมที่เหมาะสม แล้ววาดลงบนฟอร์มก็สามารถสร้างจอภาพที่ใช้สำหรับติดต่อกับผู้ใช้ รวมทั้งการใช้เทคนิคการเขียนโปรแกรมแบบโต้ตอบกับเหตุการณ์ที่เกิดซึ่งเป็นการเขียนโปรแกรมเพื่อกำหนดขั้นตอนการทำงานให้กับส่วนควบคุมต่างๆที่สร้างขึ้นตามเหตุการณ์ต่างๆที่เกิดขึ้น เช่น การเลื่อนเมาส์หรือการรับข้อมูลจากแป้นพิมพ์ ฯลฯ เป็นต้น ประกอบกับภาษาที่ใช้เขียนโปรแกรมเป็นภาษาเบสิก ซึ่งเป็นภาษาคอมพิวเตอร์ที่ผู้ใช้บนคอมพิวเตอร์สะดวกต่อการใช้งานจึงทำให้ผู้ใช้สามารถเรียนรู้ได้ภายในเวลาเพียง 2-3 ชั่วโมง ก็สามารถพัฒนาโปรแกรมบนวินโดวส์ขึ้นเป็นโปรแกรมแรกได้

โปรแกรมวิชวลเบสิกนี้เป็นเครื่องมือที่ใช้ในการพัฒนาโปรแกรมขึ้นใช้งานที่ใช้ได้ตั้งแต่ผู้ใช้ระดับต้น เพื่อใช้สร้างโปรแกรมง่ายๆบนวินโดวส์หรือโปรแกรมเมอร์ระดับกลางที่จะเรียกใช้ฟังก์ชันการทำงานต่างๆของโปรแกรมวิชวลเบสิกได้อย่างมีประสิทธิภาพ ตลอดจนโปรแกรมเมอร์ในระดับมืออาชีพที่จะพัฒนาโปรแกรมในระดับสูง โดยการใช้การเชื่อมต่อข้อมูลกับโปรแกรมอื่น (Object Linking and Embedding : OLE) และการติดต่อกับฐานข้อมูล(Application Programming Interface: API) ของวินโดวส์มาประกอบในการเขียนโปรแกรม

2.8.2 เริ่มต้นกับโปรแกรมวิชวลเบสิก

เมื่อเข้าสู่จอภาพของโปรแกรมวิชวลเบสิกที่ใช้ในการพัฒนาโปรแกรมจะแสดงดังรูปที่ 2.17

2.8.2.1 ฟอร์ม (Form)

เป็นส่วนที่ใช้สำหรับสร้างจอภาพของโปรแกรมโดยจะทำหน้าที่เป็นพื้น (Background) ของจอภาพ ทุกครั้งของการเปิดโปรเจค (Project) ใหม่จะได้ฟอร์มเปล่าซึ่งจะแสดงดังรูปที่ 2.17

2.8.2.2 กล่องเครื่องมือ (Toolbox)

กล่องเครื่องมือเป็นแถบเครื่องมือที่ประกอบไปด้วยไอคอน (Icon) ต่างๆซึ่งเรียกว่าส่วนควบคุมซึ่งจะใช้ร่วมกับฟอร์มเพื่อสร้างจอภาพของโปรแกรมโปรเจคแต่ละส่วนควบคุมจะใช้เป็นเครื่องมือที่ใช้สำหรับสร้างส่วนที่ใช้ติดต่อกับผู้ใช้หรือที่เรียกว่า "User Interface" เช่น ข้อความต่างๆช่องว่างสำหรับรับข้อมูลจากแป้นพิมพ์ ปุ่มต่างๆ ฯลฯ เป็นต้น และจะถูกนำไป ใช้งานด้วยโดยการนำส่วนควบคุมที่ต้องการไปวางลงบนฟอร์ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนควบคุมแต่ละตัวจะมีชื่อและหน้าที่ที่แตกต่างกันไป เมื่อต้องการดูชื่อของ ส่วนควบคุมใด ก็เพียงแต่เลื่อนเมาส์ไปชี้ยังส่วนควบคุมนั้น ชื่อของส่วนควบคุมจะปรากฏขึ้น

2.8.2.3 แถบเครื่องมือ (Toolbar)

แถบเครื่องมือ ทำหน้าที่เป็นผู้ช่วยในการพัฒนาโปรแกรม ซึ่งเมื่อเลื่อนเมาส์ไปชี้ยังไอคอนใดก็จะปรากฏชื่ออยู่ใต้ไอคอนนั้น แต่ละไอคอนจะมีหน้าที่ต่างกันไป

2.8.2.4 โปรแกรมโปรเจค (Project)

โดยทั่วไประบบงานหนึ่งๆมักจะประกอบด้วยหลายๆจอภาพ เช่น จอภาพสำหรับป้อนข้อมูล (Data Entry) จอภาพสำหรับค้นหาข้อมูล (Data Inquiry) เป็นต้น ดังนั้นใน การพัฒนาโปรแกรม จึงนิยมจะแยกแต่ละจอภาพออกเป็นโปรแกรมเพื่อความสะดวกต่อการแก้ไขตามหลักของการเขียนโปรแกรมแบบรวม (Modularity) เช่น ภาษาตระกูล XBase ที่แยกออกเป็นไฟล์ประเภท PRG หรือในภาษาเบสิกที่แยกออกเป็นไฟล์ประเภท BAS เป็นต้น แล้วจึงนำ แต่ละโปรแกรมย่อยมาประกอบกันขึ้นเป็นระบบ โดยการแปลไฟล์เหล่านั้นให้ร่วมกันเป็นโปรแกรมสำเร็จรูป (Executed Program ไฟล์ประเภท EXE) เพื่อนำไปใช้งาน

ในโปรแกรมวิชวลเบสิกก็เช่นเดียวกัน แต่ละจอภาพที่พัฒนาขึ้นจะได้แก่ฟอร์มต่างๆ และเมื่อนำมารวมกันก็จะกลายเป็นระบบงานหนึ่งหรือที่เรียกว่าโปรแกรมโปรเจค ดังนั้นโปรแกรมโปรเจค และ ฟอร์ม จึงต้องทำงานร่วมกันจะขาดไฟล์ใดไฟล์หนึ่งไม่ได้

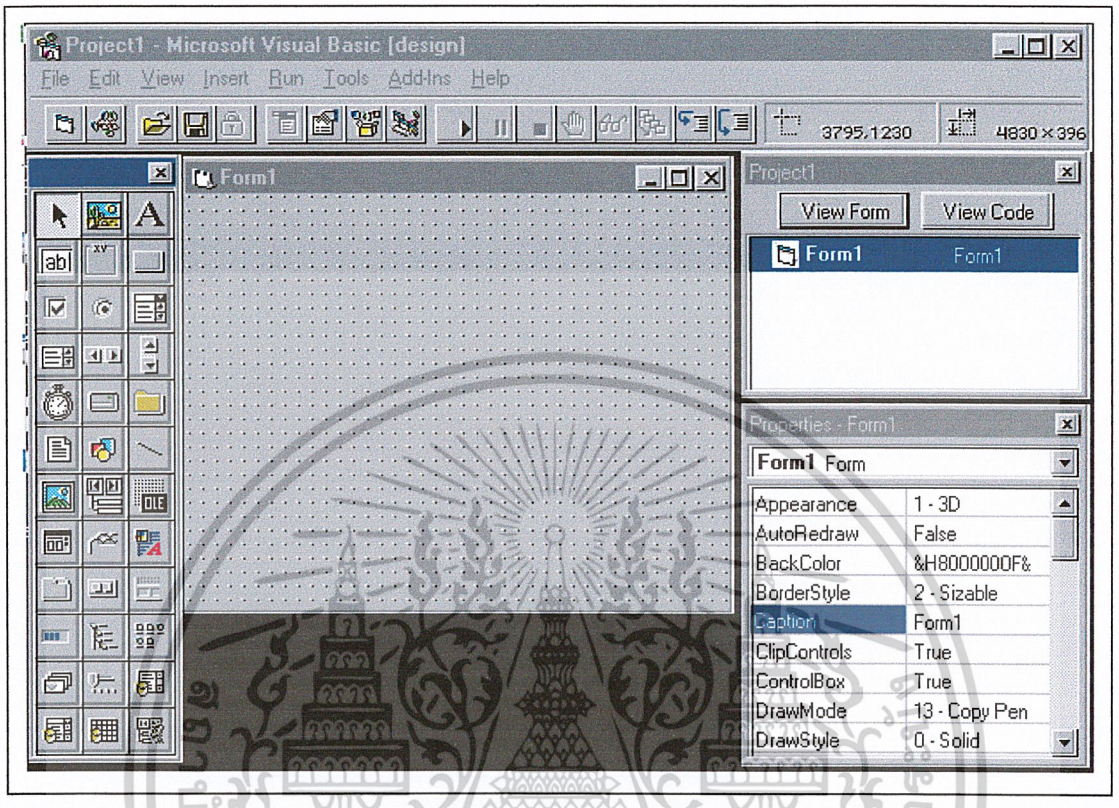
2.9 ขั้นตอนในการพัฒนาโปรแกรมวิชวลเบสิก

ขั้นตอนในการพัฒนาโปรแกรม ประกอบด้วยขั้นตอนหลัก 2 ขั้นตอน ดังนี้

ขั้นตอนที่ 1 สร้างจอภาพของโปรแกรม

ในขั้นตอนนี้ก็จะทำการนำฟอร์มมาออกแบบเพื่อใช้ในการติดต่อกับผู้ใช้หรือที่เรียกว่า การออกแบบ “User Interface” ในการพัฒนาโปรแกรมแบบเดิมขั้นตอนนี้ใช้เวลาและค่าใช้จ่ายสูง เนื่องจากจะต้องเขียน โปรแกรมเพื่อสร้างจอภาพต่างๆจากนั้นต้องแปลโปรแกรมนั้นแล้วเปิดโปรแกรมจึงจะเห็นจอภาพที่จัดทำขึ้น แต่สำหรับโปรแกรมวิชวลเบสิกปัญหาในลักษณะนี้ได้ถูกแก้ไขโดยใช้เทคนิคแบบเสมือนจริง ซึ่งเป็นความสามารถส่วนหนึ่งของโปรแกรมวิชวลเบสิก ขั้นตอนนี้จึงสามารถทำได้อย่างง่ายดายเพียงแต่นำเอาส่วนควบคุมต่างๆในทูลบอกซ์ที่ต้องการ ใช้งานมาวาดไว้บนฟอร์ม ซึ่งทำให้ประหยัดเวลาและสามารถเห็นลักษณะจอภาพที่ออกแบบได้ในขณะนั้นเลย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.19 หน้าจอภาพของโปรแกรมวิชวลเบสิก

ขั้นตอนที่ 2 เขียนโปรแกรม

เมื่อวางส่วนควบคุมต่างๆลงบนฟอร์มเป็นที่เรียบร้อยแล้ว (ส่วนควบคุมต่างๆเมื่อถูกนำมาวาดไว้บนฟอร์มจะเรียกว่าวัตถุ (Object)) ขั้นตอนต่อมาได้แก่ การเขียนโปรแกรมเพื่อกำหนดการทำงานให้กับแต่ละวัตถุภายใต้เหตุการณ์ (Event) ต่างๆที่จะเกิดขึ้นกับจอภาพนั้นๆ

2.9.1 คำสั่งสำหรับเขียนโปรแกรม

จากที่กล่าวมาแล้ว เราได้เรียนรู้ถึงวิธีการวาดวัตถุและการกำหนดคุณสมบัติแล้วแต่วัตถุดังกล่าวยังไม่สามารถทำงานได้เนื่องจาก วัตถุ บนฟอร์ม จะมีเฉพาะส่วนที่เป็นข้อมูลหรือคุณสมบัติ แต่ยังไม่มีในส่วนของรหัสโปรแกรม (Code) จึงยังไม่ครบตามหลักการของการเขียนโปรแกรมในตัววัตถุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9.2 การสนใจเหตุการณ์ที่เกิดขึ้นกับโปรแกรมวิชวลเบสิก

รูปแบบการใช้งานของโปรแกรมในแบบเดิม (Procedural-Oriented Programming)ซึ่งทำงานอยู่บนระบบปฏิบัติการดอส เหตุการณ์ต่างๆที่เกิดขึ้นในโปรแกรม เช่นการรับข้อมูลจากแป้นพิมพ์ การยืนยันการบันทึกข้อมูล ฯลฯ เป็นต้น จะเกิดขึ้นอย่างเป็นลำดับ เพื่อควบคุมให้ผู้ใช้ทำงานตามขั้นตอนที่โปรแกรมกำหนดไว้ เช่น โปรแกรมอาจกำหนดลำดับการทำงานไว้ให้ รับข้อมูลในส่วนที่ 1,2 และ 3 ตามลำดับจนครบแล้วจึงยืนยันเพื่อบันทึกข้อมูลจากนั้นก็ให้กลับไปรับข้อมูลใหม่ จะสังเกตเห็นว่าขั้นตอนการทำงานจะอยู่ในลักษณะเดิมตลอดเวลาอาจมีการใช้ฟังก์ชันคีย์เข้ามาช่วยบ้างเพื่อให้โปรแกรมมีความคล่องตัวขึ้น เช่น ใช้ฟังก์ชันคีย์เพื่อยกเลิกการรับข้อมูลโดยไม่ต้องรอจนครบทุกส่วน เพื่อข้ามไปยังขั้นตอนบันทึกข้อมูลได้เลย เป็นต้นแต่ เหตุการณ์ที่เกิดขึ้นก็ยังคงอยู่ในลักษณะเดิม

นับตั้งแต่ระบบปฏิบัติการวินโดวส์ได้มีการใช้กันอย่างแพร่หลาย รูปแบบของโปรแกรมได้มีการเปลี่ยนแปลงไปกล่าวคือ เหตุการณ์ที่เกิดขึ้นในโปรแกรมวินโดวส์ไม่ได้เกิดขึ้นอย่างเป็นลำดับเหมือนเช่นเดิม แต่จะเกิดตามที่ผู้ใช้กำหนดแทน เช่น ผู้ใช้อาจเลือกกดปุ่มบันทึกก่อนป้อนข้อมูลหรือเลือกกดปุ่มที่ 2 ก่อนปุ่มที่ 1 ทั้งๆที่ผู้พัฒนาโปรแกรมตั้งใจให้กดปุ่มที่ 1 ก่อน เป็นต้น ด้วยเหตุนี้แนวความคิดในการเขียนโปรแกรมในรูปแบบเดิม จึงไม่สามารถรองรับความสามารถที่เพิ่มขึ้นของระบบปฏิบัติการวินโดวส์ได้ แนวความคิดในการเขียนโปรแกรมในแบบสนใจเหตุการณ์ที่เกิดขึ้นจึงเกิดขึ้น

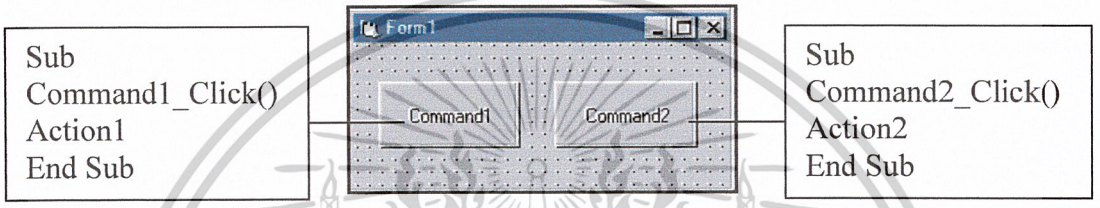
แนวความคิดในการเขียนโปรแกรมแบบสนใจเหตุการณ์ที่เกิดขึ้นจะเปลี่ยนมาสนใจกับเหตุการณ์ที่จะเกิดขึ้นในโปรแกรมมากกว่าการกำหนดขั้นตอนการทำงานของโปรแกรมในแบบเดิม เช่น ถ้ามีการเลื่อนเมาส์เกิดขึ้นจะให้โปรแกรมทำอย่างไรหรือมีการกดปุ่มที่ 1 ขึ้น จะให้โปรแกรมทำอย่างไร เป็นต้น แต่อย่างไรก็ตามก็ยังคงต้องอาศัยแนวความคิดในการเขียนแบบเดิมอยู่บ้าง เนื่องจากการกำหนดการทำงานให้กับแต่ละเหตุการณ์ยังคงต้องกำหนดอย่างเป็นขั้นตอนอยู่ดี การเขียนโปรแกรมแบบสนใจเหตุการณ์ที่เกิดขึ้นในโปรแกรมวิชวลเบสิกจะเป็นการเขียนโปรแกรมให้กับวัตถุต่างๆที่ปรากฏอยู่บนฟอร์มโดยจะพิจารณาว่าแต่ละวัตถุจะมีเหตุการณ์อะไรเกิดขึ้นบ้างแล้วจึงเลือกเขียนโปรแกรมเฉพาะเหตุการณ์นั้น

2.9.3 กระบวนการและฟังก์ชั่นในโปรแกรมวิชวลเบสิก

โปรแกรมวิชวลเบสิกยังคงใช้หลักการแบ่งโปรแกรมออกเป็นส่วนๆที่เรียกว่าโมดูล (ตามหลักการการเขียนโปรแกรมแบบ Modularity) โดยจะแบ่งโปรแกรมออกเป็นกระบวนการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และฟังก์ชั่น แต่เนื่องจากการกำหนดการทำงานของโปรแกรมจะกำหนดตามเหตุการณ์ที่เกิดขึ้น กับวัตถุ ดังนั้นการแบ่งกระบวนการในโปรแกรมวิชวลเบสิกจึงแบ่งตามชื่อวัตถุและเหตุการณ์ เช่น การเขียนโปรแกรมให้กับวัตถุ สมมุติชื่อ “Command1” ซึ่งเป็นปุ่มบนจอภาพภายใต้เหตุการณ์ กดปุ่มก็จะเขียนเป็นกระบวนการหนึ่งกับอีกวัตถุหนึ่ง ซึ่งเป็นวัตถุประเภทเดียวกันแต่คนละชื่อ สมมติชื่อ “Command2” ภายใต้เหตุการณ์กดปุ่มเหมือนกันก็ต้องเขียนเป็นอีกกระบวนการหนึ่ง ถึงแม้จะเป็นเหตุการณ์เดียวกันแต่เกิดกับวัตถุคนละตัวกันดังรูปที่ 2.18



รูปที่ 2.18 กระบวนการและฟังก์ชั่นในโปรแกรมวิชวลเบสิก

ชื่อของกระบวนการในโปรแกรมวิชวลเบสิกจะอยู่ในรูปแบบดังนี้

```
[Private | Private] Sub name [(arglist)]
...
End Sub
```

โดยที่ Private Sub จะเป็นคำเฉพาะที่ใช้บอกว่าเป็นกระบวนการ

name ได้แก่ ชื่อของกระบวนการ

arglist ได้แก่ รายชื่อ Argument ที่ใช้ภายในกระบวนการนั้น

End Sub เป็นคำเฉพาะที่ใช้สำหรับจบการทำงานของแต่ละกระบวนการ

คำสั่ง Public ใช้ในกรณีที่ต้องการให้กระบวนการนั้นสามารถถูกเรียกใช้โดยกระบวนการอื่นที่สร้างขึ้นในฟอร์มอื่น ส่วนคำสั่ง Private ค่าเริ่มต้นที่โปรแกรมวิชวลเบสิก เตรียมไว้ให้ทุกครั้งที่มีการสร้างกระบวนการจะใช้ในกรณีที่ต้องการให้กระบวนการนั้นสามารถ ถูกเรียกใช้ได้เฉพาะกระบวนการที่สร้างขึ้นในฟอร์มเดียวกับกระบวนการนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับชื่อของกระบวนการจะประกอบขึ้นจากชื่อของวัตถุรวมกับชื่อของเหตุการณ์โดยมีขีดคั่นอยู่ ในการดูว่าแต่ละวัตถุมีเหตุการณ์ใดเกิดขึ้นได้บ้างสามารถดูได้ในส่วนรวบรวม การแก้ไขของโปรแกรมวิชวลเบสิก

ส่วนฟังก์ชั่นในโปรแกรมวิชวลเบสิกจะเป็นเครื่องมือที่ช่วยในการเขียนโปรแกรม เช่น ฟังก์ชั่นที่ใช้ในการแปลงตัวเลขไปเป็นข้อความ (String) หรือ ฟังก์ชั่นในการตัดคำเป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบและการสร้าง

3.1 ขั้นตอนการดำเนินงาน

3.1.1 ศึกษาการเขียนโปรแกรมระบบจัดการฐานข้อมูล ด้วยโปรแกรม Microsoft Visual Basic 6.0

3.1.2 ศึกษาการใช้งานโปรแกรม Microsoft Access.

3.1.3 ศึกษาการเขียนโปรแกรมไมโครคอนโทรลเลอร์ PIC 18F458

3.1.4 ศึกษาการเชื่อมต่อไมโครคอนโทรลเลอร์กับอุปกรณ์ภายนอก

3.1.5 สร้างประตูจำลองการทำงานเพื่อแสดงผลการเปิดปิดของประตูอัตโนมัติ

3.1.6 ทำการเขียนโปรแกรมเพื่อสั่งการทำงานของประตูอัตโนมัติ และทดสอบการทำงานของโปรแกรม

3.1.7 เขียนโปรแกรมวิชวลเบสิกระบบการจัดการเกี่ยวกับฐานข้อมูลในการเข้าออกห้อง

3.1.8 ติดตั้งชุดอ่าน iBUTTON และอุปกรณ์ทั้งหมดที่ประตูจำลอง

3.1.9 เก็บบันทึกผลการทดลอง

3.1.10 สรุปผลการทดลอง

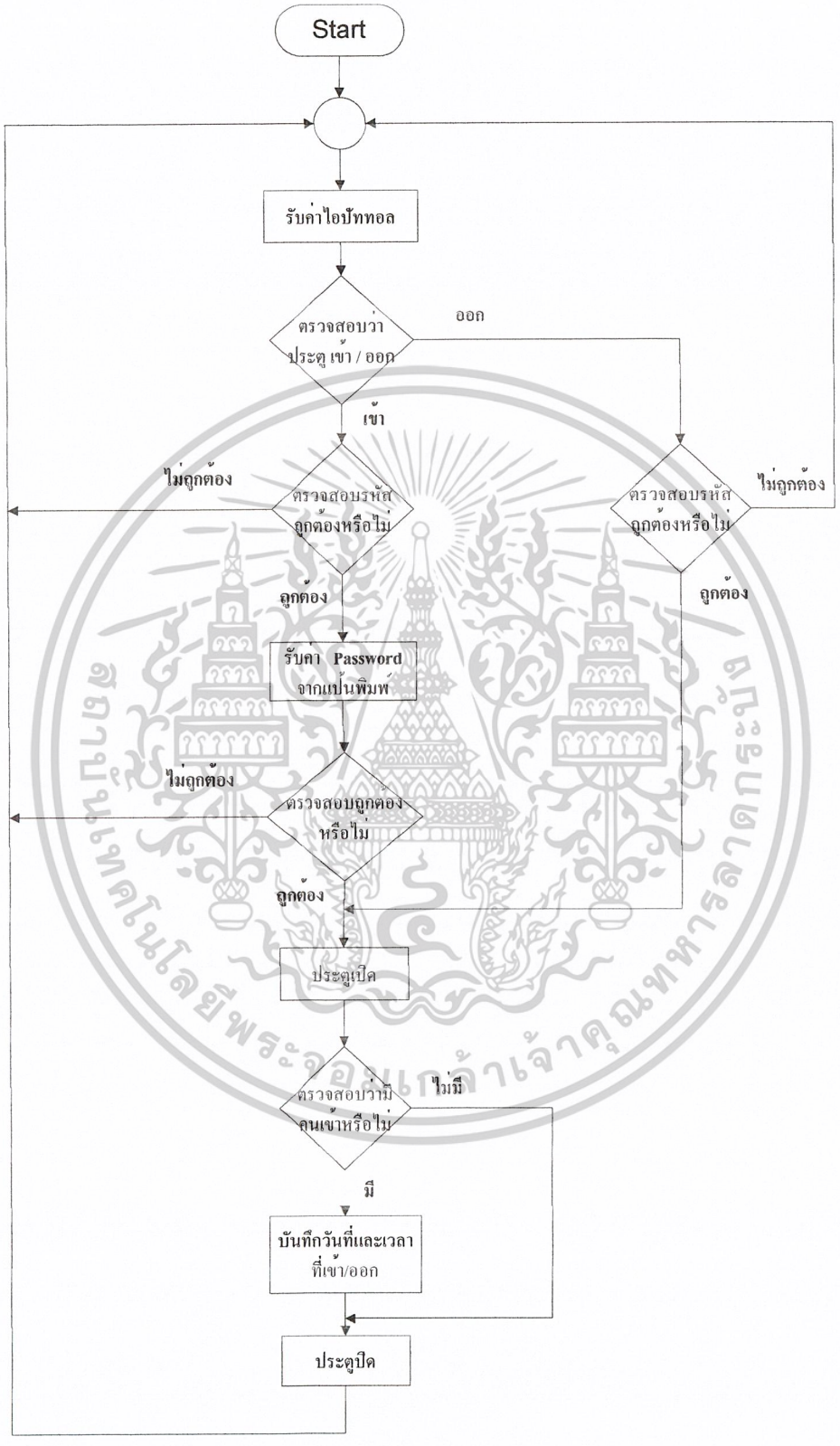
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.1 แผนผังการดำเนินงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.1 แผนผังการดำเนินงาน (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 บล๊อคไดอะแกรมการทำงานของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ขั้นตอนการออกแบบฐานข้อมูล

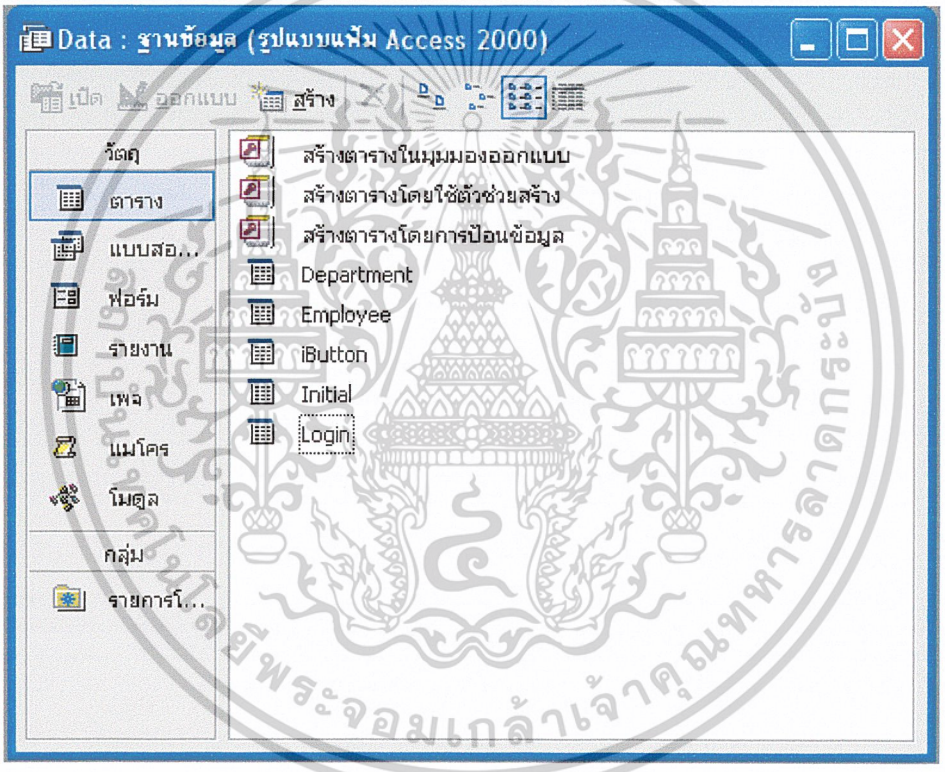
3.2.1. ศึกษาการใช้งานโปรแกรม Microsoft Access

3.2.2. ศึกษาความต้องการการใช้งาน

3.2.3. ออกแบบตารางที่จำเป็น

3.2.4. ออกแบบฟิลด์ในตาราง

จากขั้นตอนการออกแบบฐานข้อมูลดังกล่าว ทำให้สร้างตาราง และข้อมูลที่จะทำการเก็บลงไปในตารางได้ดังนี้

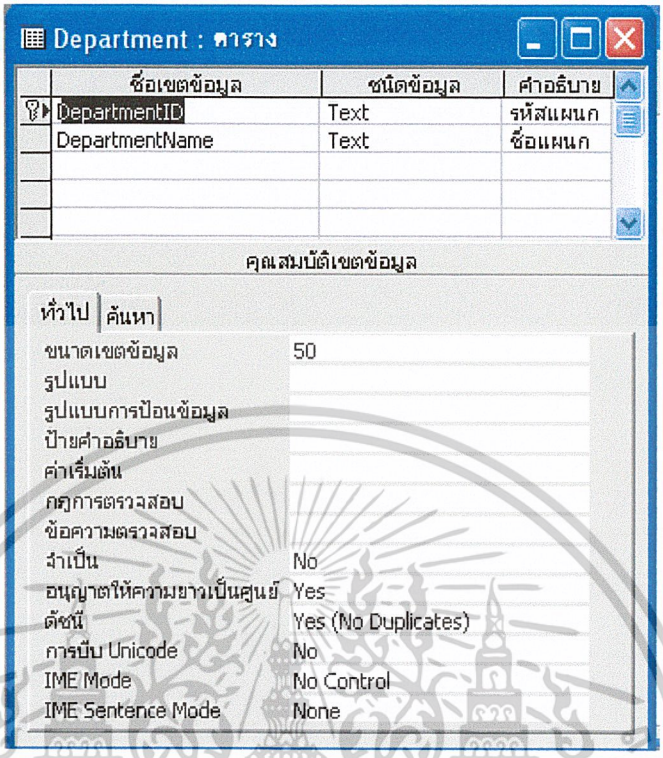


รูปที่ 3.3 ตารางที่ใช้ในการออกแบบ

จากรูปที่ 3.3 จะเห็นได้ว่ามีตารางที่จำเป็นต้องใช้ในโครงงานนี้ จำนวน 5 ตาราง ดังนี้

1. ตาราง Department ใช้ในการเก็บข้อมูลของผู้ใช้โปรแกรม ซึ่งภายในตารางนี้จะประกอบไปด้วย ฟิลด์ DepartmentID และ DepartmentName โดยฟิลด์ DepartmentID จะทำหน้าที่เก็บรหัสแผนกของผู้ใช้โปรแกรม และฟิลด์ DepartmentName จะทำหน้าที่เก็บชื่อแผนกของผู้ใช้โปรแกรม ดังรูปที่ 3.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 ฟิลด์ในตาราง Department

2. ตาราง Employee ใช้ในการเก็บข้อมูลรายละเอียดทั้งหมดของพนักงาน ซึ่งภายในตารางนี้จะประกอบไปด้วย ฟิลด์ SerialIDcode , InitialID , Name , SurName , Birthday , Sex , Address , Telephone , Mobile, DepartmentID, Iswork

ฟิลด์ SerialIDcode จะทำหน้าที่ในการเก็บรหัสบัตรประชาชน

ฟิลด์ InitialID จะทำหน้าที่ในการเก็บข้อมูลคำนำหน้าชื่อ

ฟิลด์ Name จะทำหน้าที่ในการเก็บชื่อของพนักงาน

ฟิลด์ SurName จะทำหน้าที่ในการเก็บนามสกุลของพนักงาน

ฟิลด์ Birthday จะทำหน้าที่ในการเก็บ วัน/เดือน/ปี เกิดของพนักงาน

ฟิลด์ Sex จะทำหน้าที่ในการเก็บเพศ

ฟิลด์ Address จะทำหน้าที่ในการเก็บที่อยู่

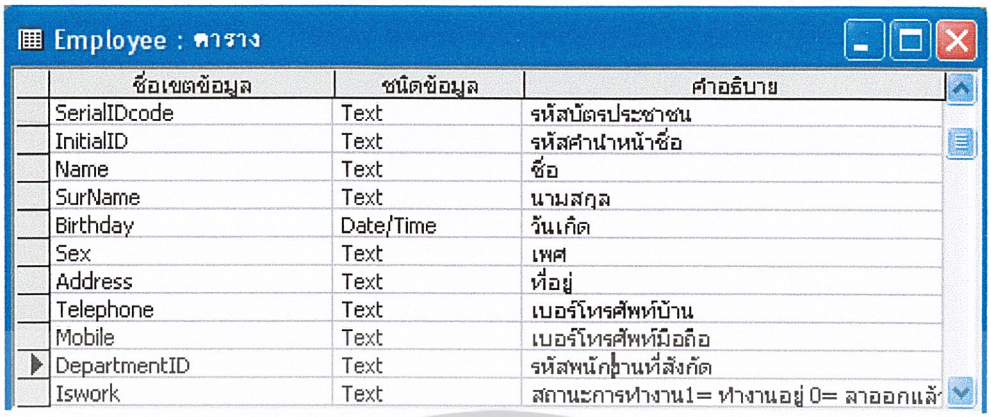
ฟิลด์ Telephone จะทำหน้าที่ในการเก็บเบอร์โทรศัพท์บ้าน

ฟิลด์ Mobile จะทำหน้าที่ในการเก็บเบอร์โทรศัพท์มือถือ

ฟิลด์ DepartmentID จะทำหน้าที่ในการเก็บรหัสพนักงานที่สังกัด

ฟิลด์ Iswork จะทำหน้าที่ในการเก็บ สถานะการทำงาน1= ทำงานอยู่ 0= ลาออกแล้ว

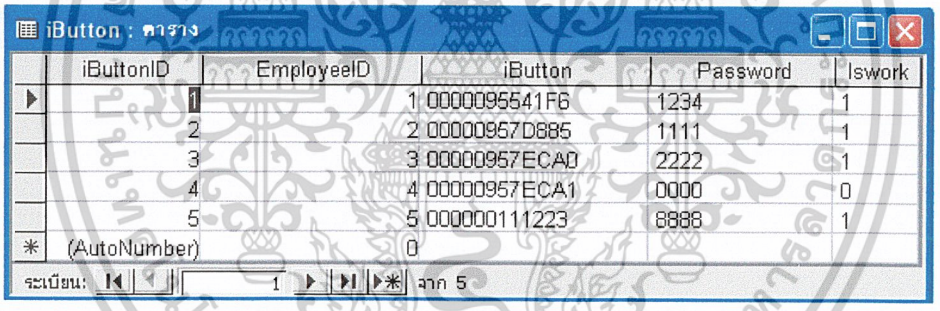
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


Employee : ตาราง

ชื่อเขตข้อมูล	ชนิดข้อมูล	คำอธิบาย
SerialIDcode	Text	รหัสบัตรประชาชน
InitialID	Text	รหัสคำนำหน้าชื่อ
Name	Text	ชื่อ
SurName	Text	นามสกุล
Birthday	Date/Time	วันเกิด
Sex	Text	เพศ
Address	Text	ที่อยู่
Telephone	Text	เบอร์โทรศัพท์บ้าน
Mobile	Text	เบอร์โทรศัพท์มือถือ
DepartmentID	Text	รหัสพนักงานที่สังกัด
Iswork	Text	สถานะการทำงาน1= ทำงานอยู่ 0= ลาออกแล้

รูปที่ 3.5 ฟิลด์ในตารางEmployee

3. ตาราง iBUTTON ในฟิลด์ต่างๆ จะใช้ในการเก็บ ข้อมูลรหัสพนักงาน หมายเลข iBUTTON รหัสผ่าน และสถานะการทำงานของพนักงาน

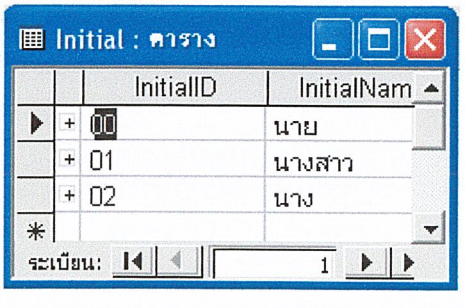

iButton : ตาราง

iButtonID	EmployeeID	iButton	Password	Iswork
1	1	0000095541F6	1234	1
2	2	00000957D885	1111	1
3	3	00000957ECA0	2222	1
4	4	00000957ECA1	0000	0
5	5	000000111223	8888	1
(AutoNumber)		0		

ระเบียน: 1 จาก 5

รูปที่ 3.6 ฟิลด์ในตาราง iBUTTON

4. ตาราง Initial ใช้ในการเก็บข้อมูลคำนำหน้าของพนักงาน เช่น นาย นาง นางสาวที่บุคคลนั้นได้สังกัดอยู่


Initial : ตาราง

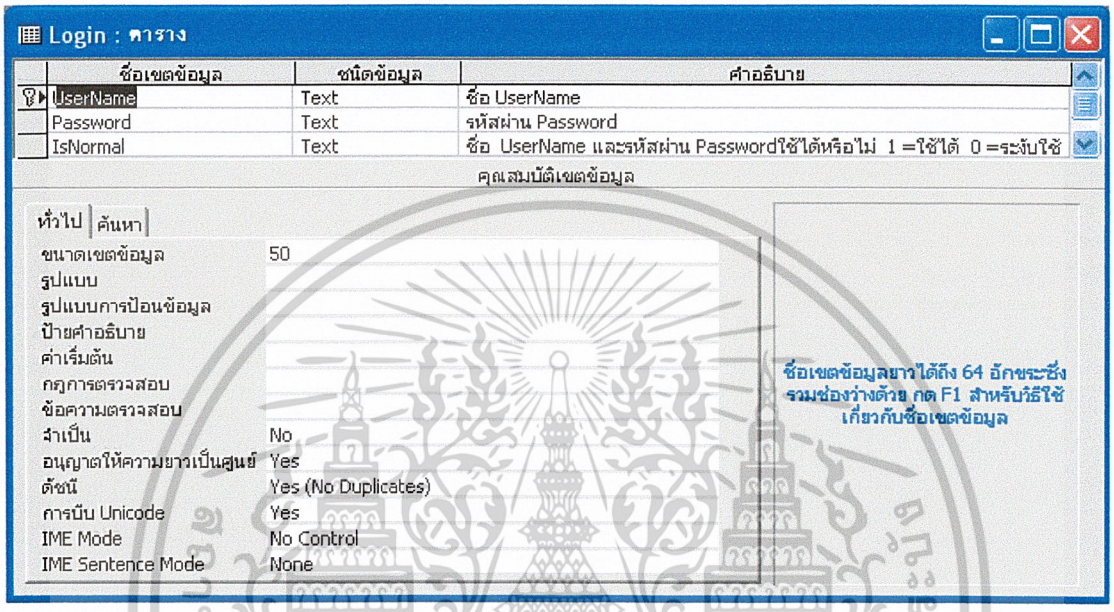
InitialID	InitialNam
00	นาย
01	นางสาว
02	นาง

ระเบียน: 1

รูปที่ 3.7 ฟิลด์ในตาราง Initial

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. ตาราง Login ใช้ในการเก็บข้อมูลของผู้ใช้งาน, รหัสลับ ,รหัสลับผิด และข้อมูลการระงับใช้งาน

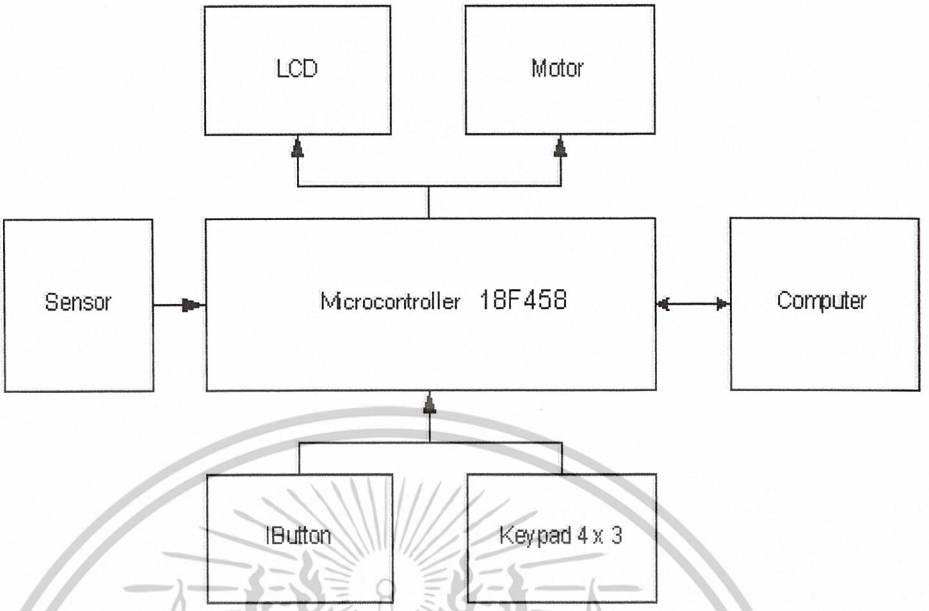


รูปที่ 3.8 ฟิลด์ในตาราง Login

3.3 การออกแบบโปรแกรมไมโครคอนโทรลเลอร์

สำหรับการควบคุมประตูอัตโนมัติ และการเชื่อมต่อเพื่ออ่านข้อมูลของตัวอุปกรณ์ iButton ได้ทำการนำไมโครคอนโทรลเลอร์เข้ามาใช้ในการควบคุมการทำงานของประตูในส่วนฮาร์ดแวร์ต่างๆ โดยจะแบ่งส่วนการทำงานของไมโครคอนโทรลเลอร์ดังบล็อกไดอะแกรมดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



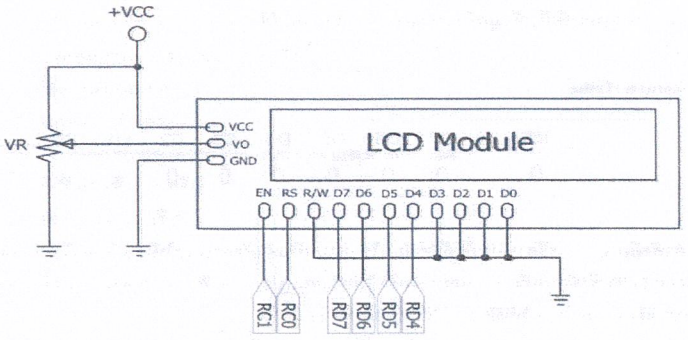
รูปที่ 3.9 บล็อกไดอะแกรมอุปกรณ์อินพุต และ เอาต์พุต

3.4 การต่อจอแสดงผลLCD (Liquid Crystal Display)

รูปที่ 3.10 จอแสดงผล LCD

จากแผนภาพจอแสดงผลผลึกเหลวจะใช้ ขนาด 16 ตัวอักษร จำนวน 2 แถว ซึ่งเป็นจอแสดงผลผลึกเหลวสำเร็จรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.11 การเชื่อมต่อ LCD แบบ 4 บิต Data

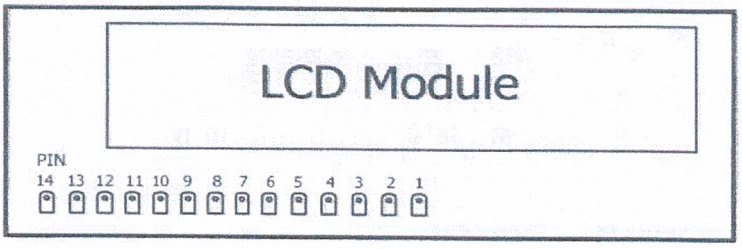
จะแสดงการต่อใช้งานของจอแบบ 4 บิต โดยจอแสดงผลผลึกเหลวจะมีขาทั้งหมด 14 ขา การต่อใช้งานต่อได้ดังรูปที่ จะต่อโดยตรงจากไมโครคอนโทรลเลอร์ PIC 18F458 โดยจะต่อใช้งานจาก pin 4,6,11,12,13,14 ของจอแสดงผลผลึกเหลวเข้ากับ pic ที่ port RC0,RC1,RD4,RD5,RD6และ RD7ตามลำดับ ส่วน PIN ที่1,5,6,7,8,9,10 จะต่อลงกราวด์ โดยขา 2 ของจอLCD ต่อกับแหล่งจ่ายไฟฟ้า +5 โวลต์ ขา 3 จะต้องต่อกับตัวต้านทานปรับค่าได้ มีค่า 10 กิโลโอห์ม เพื่อเป็นการปรับความเข้มของตัวแสดงผล ขา 4 ของจอแสดงผลผลึกเหลวซึ่งเป็นขา RS จะต่อกับพอร์ต RC0 ของไมโครคอนโทรลเลอร์ PIC 18F458 ขา 6 ของจอแสดงผลผลึกเหลว ซึ่งเป็นขา E ต่อกับ พอร์ต RC1 ของไมโครคอนโทรลเลอร์ PIC 18F458 และขา 11,12,13,14 ของจอแสดงผลผลึกเหลวซึ่งเป็นขา DATA จะต่อกับ พอร์ตRD4,RD5,RD6และRD7 ตามลำดับโดยจะ ติดต่อระหว่างจอภาพผลึกเหลวกับไมโครคอนโทรลเลอร์ PIC 18F458 แบบ 4 บิต ซึ่งการใช้งานต้องเขียนโปรแกรมควบคุมการติดต่อแบบ 4 บิต

LCD เป็นอุปกรณ์กำเนิดแสงชนิดหนึ่งที่นิยมนำไปใช้เป็นตัวแสดงผลข้อความหรือตัวเลขมากกว่านำไปใช้เป็นอุปกรณ์กำเนิดแสงสว่าง เนื่องจากมันมีความเข้มของแสงต่ำมาก และกินกำลังไฟฟ้าต่ำมากด้วยเช่นกัน จึงนิยมนำ LCD มาใช้ในเครื่องคิดเลขและนาฬิกาดิจิตอล

LCD เป็นจอแสดงผลที่ได้รับความนิยมอย่างสูง ในปัจจุบัน LCD ถูกนำมาใช้งานแทนที่ 7-SEGMENT เป็นจำนวนมากเนื่องจาก LCD สามารถแสดงตัวอักษรและรายละเอียดได้มากกว่า 7-SEGMENT ทำให้มีการนำไปใช้งานในเครื่องมือต่างๆ มากขึ้น เพราะการสื่อสารระหว่างผู้ใช้กับเครื่องมือต่างๆโดยผ่าน LCD นั้นมีความสะดวก ซึ่งในที่นี้จะกล่าวถึงเฉพาะ LCD ที่นำมาใช้ในโครงงานนี้ ซึ่งเป็นแบบ 16 ตัวอักษร 2 บรรทัด เท่านั้น

โมดูล LCD แบบ 16 ตัวอักษร 2 บรรทัด มีขาต่อใช้งาน 14 ขาดังแสดงรายละเอียดการจัดขาดังรูปที่ ส่วนหน้าที่ของขาที่ใช้งานโมดูล LCD มีรายละเอียดดังตารางที่3.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.12 การวางตำแหน่งขาสัญญาณของจอแสดงผลแอลซีดี

ตารางที่ 3.1 ชื่อและหน้าที่ของขาสัญญาณต่างๆของ LCD

ขาที่	ชื่อขา	หน้าที่ใช้งาน
1	GND	ต่อกับกราวด์ของวงจร
2	+VDD	ต่อกับไฟเลี้ยง +5V
3	Vo	เป็นขาสำหรับป้อนแรงดันเพื่อปรับความสว่างของจอแสดงผลแอลซีดี
4	RS	เป็นขาเลือกการติดต่อกับรีจิสเตอร์คำสั่งหรือรีจิสเตอร์ข้อมูล 0 : จะติดต่อกับรีจิสเตอร์คำสั่ง 1 : จะติดต่อกับรีจิสเตอร์ข้อมูลเพื่อนำข้อมูลไปแสดงผล
5	R/W	เป็นขาเลือกการอ่านหรือเขียนข้อมูลกับโมดูล LCD
6	E	เป็นขาสำหรับป้อนสัญญาณพัลส์เอ็นเอเบิลให้โมดูล LCD ทำงาน
7-14	D0-D7	เป็นขาข้อมูล 8 บิต โดยใช้ขา 7 คือ D0 ไล่ไปจนถึงขา 14 คือ D7

การติดต่อกับโมดูล LCD 16 ตัวอักษร 2 บรรทัด

มีอยู่ด้วยกัน 2 แบบคือ แบบ 8 บิต และแบบ 4 บิต โดยปกติจะมีการใช้แบบ 8 บิต มากกว่าแต่หากมีข้อจำกัดเรื่องจำนวนของพอร์ต ควรเลือกใช้การติดต่อแบบ 4 บิต ซึ่งจะมีขั้นตอนที่เพิ่มขึ้นเพียงเล็กน้อย แต่จะใช้สายสัญญาณเพียง 6 เส้นในขณะที่แบบ 8 บิต จะใช้สายสัญญาณ 10 เส้น

- การติดต่อแบบ 4 บิต

ลำดับขั้นตอนในการเขียนโปรแกรมเพื่อใช้งาน LCDในการเขียนโปรแกรมควบคุมให้จอ LCD แสดงผลทำงานนั้น ขั้นแรกจะต้องทำการกำหนดฟังก์ชั่นการทำงานต่างๆของจอ LCD หรือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรียกว่าการ อินิเชียล LCD(Initial LCD) ซึ่งก็คือการเขียนข้อมูลคำสั่งไปยังรีจิสเตอร์คำสั่งภายในของโมดูล LCD เพื่อเตรียมความพร้อมให้แก่โมดูล LCD ซึ่งในการอินิเชียล LCD ก็คือ การกำหนดให้ LCD มีการทำงานในรูปแบบต่างๆ เช่น กำหนดตำแหน่งของเคอร์เซอร์มาอยู่ที่จุดเริ่มต้นที่ตำแหน่งซ้ายมือสุด เปิดจอแสดงผล เปิด-ปิด เคอร์เซอร์ กำหนดรูปแบบการแสดงผลของตัวอักษร กำหนดจำนวนบรรทัด และการกำหนดโหมดในการติดต่อ โดยในการส่งข้อมูลไปยัง LCD จะแบ่งข้อมูลออกเป็น 2 ชนิด คือ ข้อมูลคำสั่ง(Command) และข้อมูล(Data) ในการแสดงผล ซึ่งการส่งข้อมูลทั้ง 2 ชนิดจะมีลำดับขั้นตอนเหมือนกันแต่จะต่างกันตรง การกำหนดข้อมูลที่บิต RS เพื่อแยกชนิดของข้อมูล

มีลักษณะการต่อวงจรดังรูป แสดงการเชื่อมต่อ LCD แบบ 4 บิต Data จะเห็นว่าขาของข้อมูลที่ใช้มีเพียง 4 เส้น คือ D4-D7 ซึ่งต่อเข้ากับ RD4-RD7 ของ CPU สำหรับขา D0-D3 ของโมดูล LCD ให้ต่อลงกราวด์ ส่วนขา RS ต่อเข้ากับ RC0 และ E ต่อเข้ากับ RC1 จุดที่แตกต่างจากการติดต่อแบบ 8 บิต ในการเขียนโปรแกรมคือ ต้องทำการส่งข้อมูล 2 ครั้ง คือส่ง 4 บิตบนของข้อมูลก่อน จากนั้นจึงส่งข้อมูล 4 บิตล่างตามไป สำหรับการอินนิเชียลนั้นมีสิ่งที่ต้องทำก่อนเสมอ คือ ต้องส่งข้อมูล 03h(0011) ออกไปให้ LCD ที่ขา D7-D4 แล้วทำการส่งสัญญาณ Enable จำนวน 2 ครั้ง เพื่อจัดสถานะการทำงานของ LCD จากนั้นส่งข้อมูลคำสั่ง 02h ออกไปที่ขา D7-D4 แล้ว Enable อีกเช่นกันเพื่อกำหนดให้แอลซีดีทำงานในโหมด 4 บิต เท่านี้แอลซีดีก็พร้อมทำงานในโหมด 4 บิต แล้วส่วนการกำหนดค่าอื่นๆสามารถทำต่อจากนี้ได้เลยแต่ต้องส่งข้อมูลในแบบ 4 บิต

การเขียนข้อมูลคำสั่งไปยังจอแอลซีดี

1. ทำให้ขา RS เป็น “0” เพื่อแจ้งให้โมดูล LCD ทราบว่าข้อมูลที่ขา Data เป็นข้อมูลคำสั่ง
2. ส่งข้อมูลคำสั่ง 4 บิตบน (บิต7 – บิต4) ที่ต้องการไปยังขา Data ทั้ง 4 เส้น
3. ส่งพัลส์ Enable ไปยังขา E
4. ส่งข้อมูลคำสั่ง 4 บิตล่าง (บิต3 – บิต0) ที่ต้องการไปยังขา Data ทั้ง 4 เส้น
5. ส่งพัลส์ Enable ไปยังขา E

การเขียนข้อมูล (Data) เพื่อแสดงผลข้อมูล

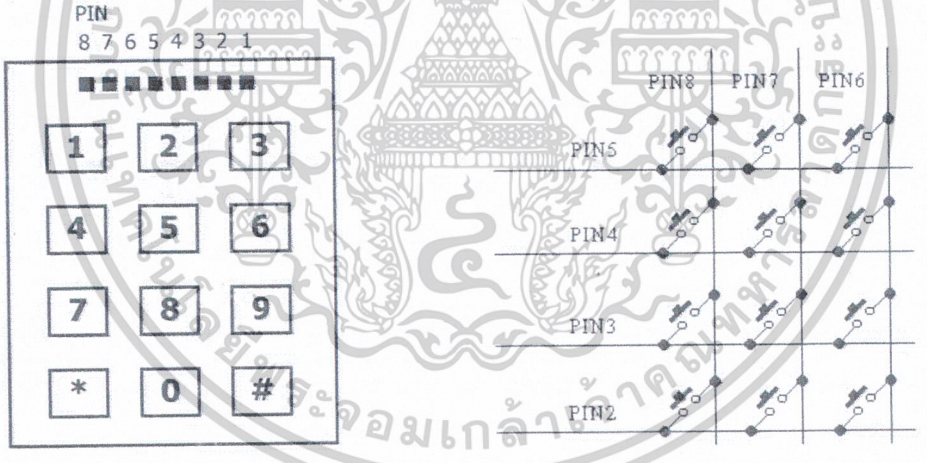
1. ทำให้ขา RS เป็น “1” เพื่อแจ้งให้โมดูล LCD ทราบว่าข้อมูลที่ขา Data เป็นข้อมูลในการแสดงผล
2. ส่งข้อมูล (Data) 4 บิตบน (บิต7 – บิต4) ที่ต้องการไปยังขา Data ทั้ง 4 เส้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ส่งพัลส์ Enable ไปยังขา E
4. ส่งข้อมูล (Data) 4 บิตล่าง (บิต3 – บิต0) ที่ต้องการไปยังขา Data ทั้ง 4 เส้น
5. ส่งพัลส์ Enable ไปยังขา E

3.5 การเชื่อมต่อกับ Key Switch แบบ 4x3

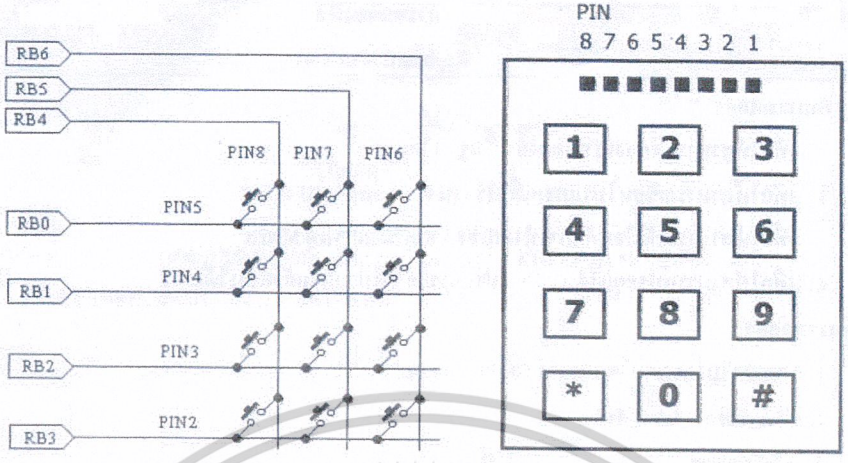
ในการต่อใช้งานกับไมโครคอนโทรลเลอร์ จะทำการต่อสวิตซ์ ดังแสดงในรูปที่ 3.23 การต่อสวิทช์แบบนี้เรียกว่า การต่อสวิทช์แบบเมตริกส์ (Matrix Switch) โดยสวิทช์จะถูกต่ออยู่ในลักษณะแนวแกนตั้งและแนวแกนนอน จะเรียกแนวแกนตั้งว่าหลักหรือคอลัมน์ (Column) ในขณะที่แนวนอนจะเรียกว่าแถวหรือโรว์ (Row) ดังนั้นค่าของสวิทช์จะต้องประกอบด้วย ตำแหน่งในแนวหลักและแถว กระบวนการที่จะทำให้ได้ค่าของสวิทช์ค่อนข้างซับซ้อนอย่างไรก็ตามก็มีข้อดีคือสามารถรองรับการเพิ่มของสวิทช์ได้อย่างสะดวก เพียงเพิ่มเติมจำนวนสวิทช์และแก้ไขซอฟท์แวร์อีกเล็กน้อยทำให้วงจรสวิทช์แบบเมตริกซ์เป็นที่นิยมใช้กันอย่างแพร่หลาย



รูปที่ 3.13 ลักษณะของคีย์สวิทช์แบบเมทริกซ์ขนาด 4x3

การใช้งานคีย์สวิทช์แบบ Matrix ขนาด 4x3 จะทำการต่อวงจรคีย์สวิทช์เข้ากับบอร์ดไมโครคอนโทรลเลอร์กับ PORTB เนื่องจาก PORTB จะมีคุณสมบัติพิเศษคือ มีวงจรพูลอัพ (pull-up) ภายในจึงไม่ต้องต่อตัวต้านทานภายนอกให้เสียเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.14 วงจรการทดลองต่อคีย์สวิทช์

ส่วนโปรแกรมที่รับค่าการกดสวิทช์จาก จากสวิทช์เมตริกซ์แบบ 4x3 ซึ่งจะใช้หลักการสแกนคีย์ตรวจสอบการกดสวิทช์ทีละแถว โดยหลักการคือส่งลอจิก “0” ออกไปทางด้าน ROW ทีละแถวแล้วตรวจสอบบิตที่ต่ออยู่ทางด้านคอลัมน์ (COL1,COL2 และ COL3) หากอ่านบิตทางด้านคอลัมน์ใดเป็น “0” แสดงว่ามีการกดที่ แถวและคอลัมน์นั้น ดังตารางที่ 3.2จะสรุปสภาวะการกดสวิทช์ต่างๆ

ตารางที่ 3.2 สรุปสภาวะการกดสวิทช์ต่างๆ

ROW 1, 2, 3, 4				ด้าน คอลัมน์ COL1, COL2, COL3											
R1	R2	R3	R4	C1	C2	C3	C1	C2	C3	C1	C2	C3	C1	C2	C3
				0	1	1	1	0	1	1	1	0	1	1	1
0	1	1	1	คีย์ 1 ถูกกด			คีย์ 2 ถูกกด			คีย์ 3 ถูกกด			ไม่กดคีย์		
1	0	1	1	คีย์ 4 ถูกกด			คีย์ 5 ถูกกด			คีย์ 6 ถูกกด			ไม่กดคีย์		
1	1	0	1	คีย์ 7 ถูกกด			คีย์ 8 ถูกกด			คีย์ 9 ถูกกด			ไม่กดคีย์		
1	1	1	0	คีย์ * ถูกกด			คีย์ 0 ถูกกด			คีย์ # ถูกกด			ไม่กดคีย์		

ตัวอย่างเช่น ส่งลอจิก “0” ไปที่ขา RB0(ROW1) แล้วเช็คสถานะที่ RB4,RB5 และ RB6 หากขา RB4 เป็น “0” แสดงว่ามีการกดคีย์ 1 หรือหาก RB5 เป็น “0” คีย์ที่กดก็จะเป็น 2 และ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หาก RB6 เป็น “0” แสดงว่ากดคีย์ 3 ซึ่งเราจะใช้หลักการนี้ในการสแกนแต่ละแถวแต่ละคอลัมน์เพื่อหาตำแหน่งการกดคีย์

โดย RB0-RB3 จะทำหน้าที่เป็นอินพุต และ RB4-RB6 จะเป็นเอาท์พุต ในบิตที่เป็นอินพุตนี้โดยปกติแล้วจะต้องต่อตัวต้านทานพูลอัพ (pull-up) ให้ทุกขาแต่ในการทำโครงงานนี้ไม่จำเป็นต้องต่อเนื่องจากได้เลือกใช้งานที่พอร์ต B ซึ่งมีคุณสมบัติพิเศษคือ มีวงจรพูลอัพภายในสามารถเลือกการพูลอัพได้จากการเคลียร์บิต RBPU ซึ่งถ้าเบอร์ 16F877 จะอยู่ที่รีจิสเตอร์ OPTION_REG ในบิตที่7 ส่วน 18F458 อยู่ที่รีจิสเตอร์ INTCON2 ในบิตที่7 เช่นกัน

ตารางที่ 3.3 ค่าต่างๆจากการกดคีย์

ตำแหน่งคีย์	RB3	RB2	RB1	RB0	ค่าข้อมูล
ROW 0	1	1	1	0	0Eh
ROW 1	1	1	0	1	0Dh
ROW 2	1	0	1	0	08h
ROW 3	0	1	1	1	07h

ตารางที่ 3.4 ค่าที่ใช้ในการแสกนคีย์

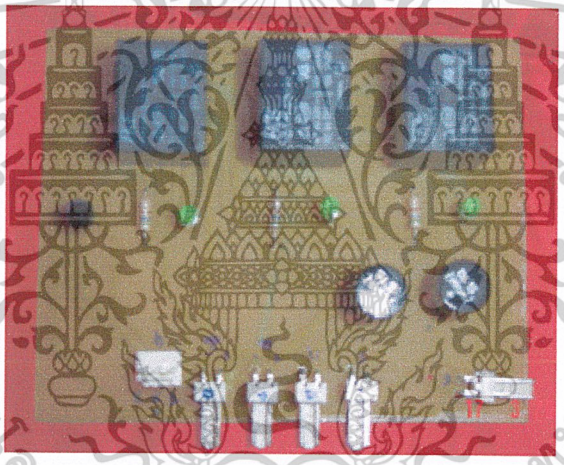
ตำแหน่งคีย์	RB7	RB6	RB5	RB4	ค่าข้อมูล
Column 0	1	1	1	0	E0h
Column 1	1	1	0	1	D0h
Column 2	1	0	1	1	B0h

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.5 ข้อมูลในการกดสวิทซ์ในตำแหน่งต่างๆ

High/Low	Column 0(RB4)	Column 1(RB5)	Column 2(RB6)
ROW 0(RB0)	EEh	DEh	BEh
ROW 1(RB1)	EDh	DDh	BDh
ROW 2(RB2)	EBh	DBh	BBh
ROW 3(RB3)	E7h	D7h	B7h

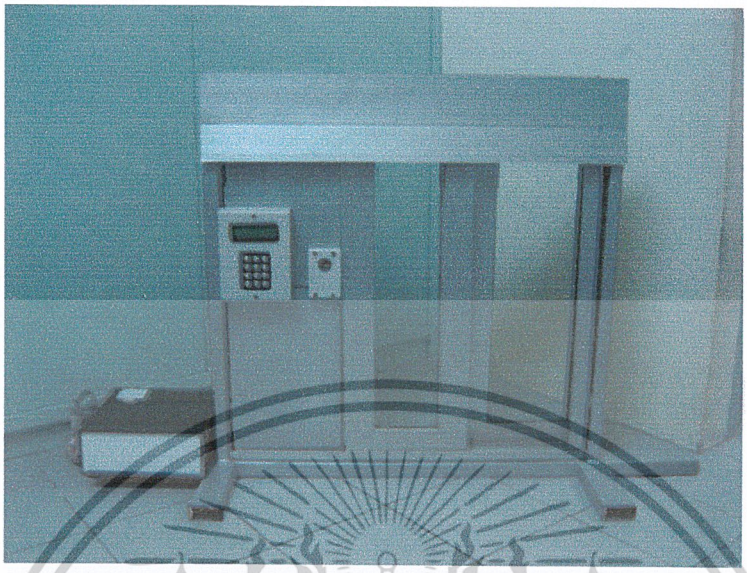
3.6 การเชื่อมต่อกับมอเตอร์ไฟฟ้ากระแสตรง (DC Motor)



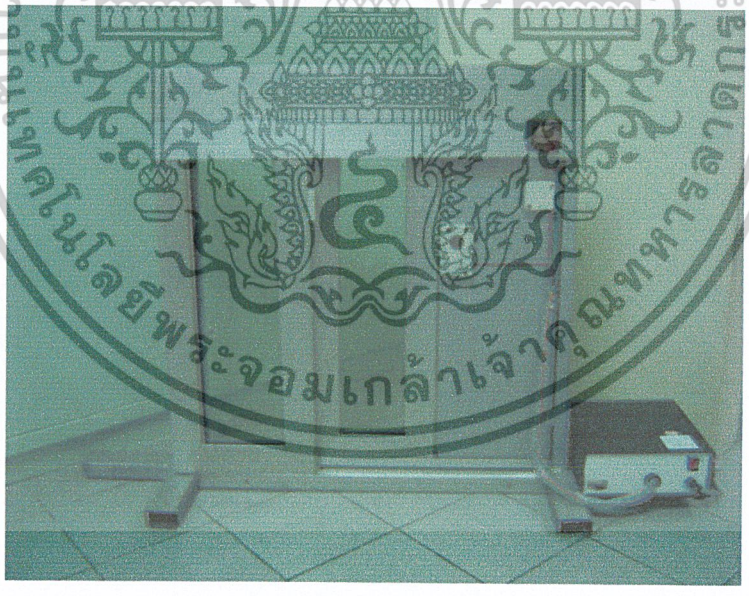
รูปที่ 3.15 การต่อวงจรของชุดควบคุมมอเตอร์

เมื่อวงจรควบคุมมอเตอร์ได้รับสัญญาณ output จาก ไมโครคอลโทรลเลอร์ PIC 18F458 ตัวทรานซิสเตอร์ของวงจรควบคุมมอเตอร์ที่ทำหน้าที่เป็นสวิทซ์ จะทำงาน ทำให้ Relay ตัวที่ 3 ทำงาน ซึ่ง Relay ตัวที่ 3 จะทำหน้าที่เป็นตัวควบคุมการ เปิด/ปิด ประตู และเมื่อ Relay ตัวที่ 3 ทำงาน จะทำให้ Relay ตัวที่1 ทำงานด้วย ซึ่ง Relay ตัวที่ 1 จะทำหน้าที่ในการเปิดประตู และจะมี Limit Switch เป็นตัวตัดวงจรควบคุมการทำงานของมอเตอร์ และเมื่อกรณีที่วงจรควบคุมมอเตอร์หยุดรับสัญญาณ หรือไม่มีสัญญาณที่ส่งมาจากคอลโทรลเลอร์ จะทำให้ Relay ตัวที่ 3 ไม่ทำงาน จะทำให้ Relay ตัวที่ 2 ทำงาน ซึ่งเป็น Relay ที่ใช้ควบคุมการปิดประตู มอเตอร์จะหมุนกลับทิศทาง และจะมี Limit Switch อีกตัวหนึ่งซึ่งจะทำหน้าที่ในการควบคุมการทำงานของมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

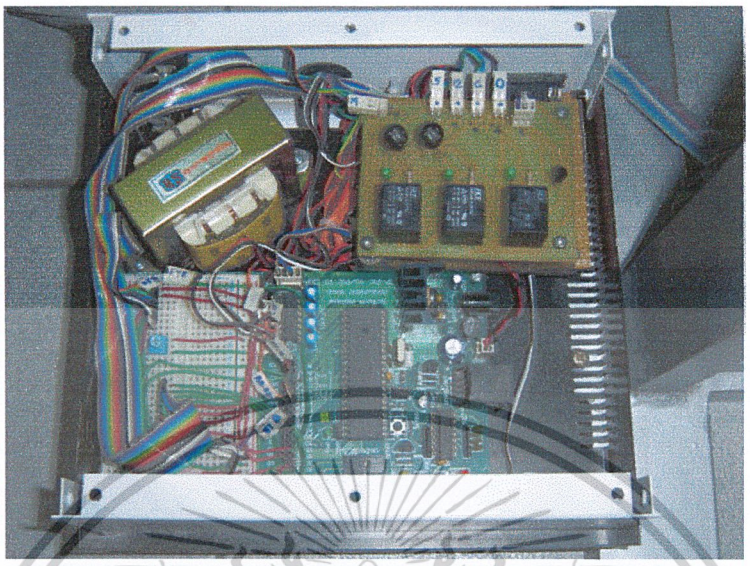


รูปที่ 3.16 ประตูจำลองที่ใช้ในการเข้า



รูปที่ 3.17 ประตูจำลองที่ใช้ในการออก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

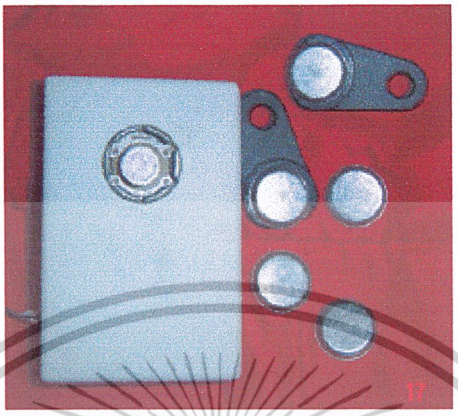


รูปที่ 3.18 อุปกรณ์ภายในทั้งหมด

รูปที่ 3.19 หน้าจอ LCD แสดงว่าพร้อมการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

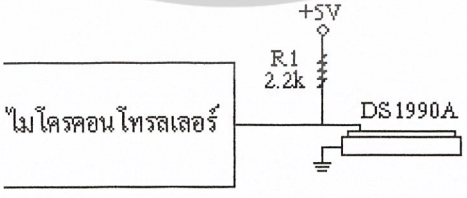
3.7 การเชื่อมต่อกับตัวอุปกรณ์ไอบัททอน



รูปที่ 3.20 อุปกรณ์IBUTTON และชุดรับรหัส IBUTTON

IBUTTON คือ อุปกรณ์อิเล็กทรอที่มีการใส่รหัสโค้ดเข้าไปโดย IBUTTON ที่ผลิตมาจากโรงงานแต่ละตัวจะมีโค้ดรหัสข้อมูลที่ไม่ซ้ำกัน โดยการนำเอาไปใช้งาน เช่น ใช้เป็นกุญแจในการเปิด/ปิด สิ่งต่างๆ ที่เราต้องการเนื่องจากอุปกรณ์เหล่านี้ จะทำการเลียนแบบได้ค่อนข้างยากมาก (ต้องใช้เทคโนโลยีสูง) ทำให้มีความปลอดภัยสูง การติดต่อจะเป็นแบบ 1-wire โดยจะมี 8 –bit family code + 48 bit serial number + 8 bit CRC tester รวมทั้งหมดเป็น 64 บิต

ในโครงงานนี้จะทำการอ่านค่าข้อมูลจากอุปกรณ์ IBUTTON ออกมาแล้วนำเอาข้อมูลที่ได้ส่งออกทาง port อนุกรมด้วยความเร็วอัตราบอร์ด 9600 ไปแสดงที่คอมพิวเตอร์ผ่านโปรแกรม Hyper terminal บน Window การต่อวงจรต้องมีตัวต้านทานต่อพูลอัพไว้ด้วยและต่อหน้าสัมผัสกราวด์เข้ากับกราวด์ของไมโครคอนโทรลเลอร์ ดังที่แสดงไว้



รูปที่ 3.21 ตัวอย่างการต่อ DS1990A เข้ากับขาพอร์ตของไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบการใช้งานคำสั่งของการติดต่ออุปกรณ์ระบบบัส 1 สาย (1-Wire)

OWIN Pin, Mode, [InputData] เป็นคำสั่งรับข้อมูลเข้าจากอุปกรณ์ที่ใช้การเชื่อมต่อผ่านระบบบัส 1 สาย (1-Wire)

Pin เป็นขาของพอร์ตที่ใช้ในการติดต่อกับอุปกรณ์ผ่านระบบบัส 1 สาย มีอัตราเร็วในการถ่ายทอดข้อมูล 20 กิโลบิตต่อวินาที (ไม่รวมสัญญาณตอบสนองต่างๆ) ในระหว่างกระทำคำสั่งนี้ ขาพอร์ตของไมโครคอนโทรลเลอร์จะทำงานสลับกันระหว่างการเป็นเอาท์พุตและอินพุตจนกระทั่งการทำงานเสร็จสิ้นลง ขาพอร์ตนี้จะถูกกำหนดเป็นขาอินพุต ในการต่อวงจรทางฮาร์ดแวร์ต้องต่อตัวต้านทานพูลอัปขาที่ใช้งานด้วย

Mode เป็นตัวแปรหรือค่าคงที่ใช้กำหนดโหมดของการถ่ายทอดข้อมูลดังแสดงรายละเอียดในตาราง

InputData เป็นพื้นที่สำหรับเก็บข้อมูลที่อ่านมาได้จากอุปกรณ์ระบบบัส 1 สายที่ติดต่อด้วย โดยสามารถเก็บไว้ได้ในหลายรูปแบบ ทั้งรหัสแอสกี้, เลขฐานสอง, เลขฐานสิบ, เลขฐานสิบหก หรือรูปแบบของตัวอักษร (String)

OWOUT Pin, Mode, [OutputData] เป็นคำสั่งส่งข้อมูลไปยังอุปกรณ์ที่ใช้การเชื่อมต่อผ่านระบบบัส 1 สาย (1-Wire)

Pin เป็นขาของพอร์ตที่ใช้ในการติดต่อกับอุปกรณ์ผ่านระบบบัส 1 สาย มีอัตราเร็วในการถ่ายทอดข้อมูล 20 กิโลบิตต่อวินาที (ไม่รวมสัญญาณตอบสนองต่างๆ) ในระหว่างกระทำคำสั่งนี้ ขาพอร์ตของไมโครคอนโทรลเลอร์จะทำงานสลับกันระหว่างการเป็นเอาท์พุตและอินพุตจนกระทั่งการทำงานเสร็จสิ้นลง ขาพอร์ตนี้จะถูกกำหนดเป็นขาอินพุต ในการต่อวงจรทางฮาร์ดแวร์ต้องต่อตัวต้านทานพูลอัปขาที่ใช้งานด้วย

Mode เป็นตัวแปรหรือค่าคงที่ใช้กำหนดโหมดของการถ่ายทอดข้อมูลดังแสดงรายละเอียดในตาราง

OutputData เป็นพื้นที่สำหรับเก็บข้อมูลที่ต้องการเขียนหรือส่งไปยังอุปกรณ์ระบบบัส 1 สายที่ติดต่อด้วย โดยรูปแบบของข้อมูลมีได้หลายรูปแบบ ทั้งรหัสแอสกี้, เลขฐานสอง, เลขฐานสิบ, เลขฐานสิบหก หรือรูปแบบของตัวอักษร (String)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Mode bit number	Effect
0	1 = Send reset pulse before data
1	1 = Send reset pulse after data
2	0 = Byte-sized data 1 = Bit-sized data

ตารางที่ 3.6 รายละเอียดในการเลือกโหมดในการติดต่ออุปกรณ์ระบบบัส 1 สายของคำสั่ง OWIN และ OWOUT

3.8 การทำงานเชื่อมต่อกับคอมพิวเตอร์

ซึ่งการติดต่อรับส่งข้อมูลจากไมโครคอนโทรลเลอร์จะส่งผ่าน PORTC.6 และ PORTC.7 ของไมโครคอนโทรเลอร์ซึ่งเป็นการเชื่อมต่อกับพอร์ต RS232 ซึ่งจะเรียกรูปแบบการสื่อสารข้อมูลแบบนี้ว่า การสื่อสารข้อมูลแบบอะซิงโครนัส ซึ่งความหมายก็คือ ไม่ใช้สัญญาณนาฬิกาในการติดต่อข้อมูล ดังนั้นถ้าต้องการส่งหรือรับข้อมูลเพียงอย่างเดียวสามารถใช้สายสัญญาณกับสายกราวด์เพียง 2 เส้นเท่านั้น

มาตรฐาน RS232 เป็นมาตรฐานทางไฟฟ้าสำหรับพอร์ตอนุกรมของคอมพิวเตอร์ ซึ่งค่าลอจิก “1” และลอจิก “0” จะมีแรงดันไม่เท่ากับมาตรฐานทีทีแอล ซึ่งมาตรฐานทีทีแอลนั้นที่ลอจิก “1” จะมีแรงดันเท่ากับ 5V ส่วนลอจิก “0” จะมีแรงดันเท่ากับ 0 V สำหรับมาตรฐาน RS232 ที่ลอจิก “1” จะมีแรงดัน -3V ถึง -12V และที่ลอจิก “0” จะมีแรงดัน +3V ถึง +12V ดังนั้นเมื่อต้องการอ่านค่าจะต้องมีวงจรเพื่อเปลี่ยนระดับแรงดันส่วนนี้ก่อน ซึ่งวงจรแปลงระดับสัญญาณนี้จะต้องอาศัยการเชื่อมต่อผ่าน IC MAX-232 เพื่อแปลงระดับสัญญาณ ซึ่ง IC MAX-232 จะทำการแปลงข้อมูลส่งของไมโครคอนโทรลเลอร์ จากระดับ TTL ไปเป็นระดับของ RS232 และทำการแปลงข้อมูลรับจากคอมพิวเตอร์จากระดับของ RS232 มาเป็นระดับ TTL โดยในตัวของ IC MAX-232 นั้นจะประกอบด้วยขาสัญญาณสำหรับการรับและการส่งข้อมูลอย่างละ 2 ชุด ซึ่งวงจรแปลงระดับสัญญาณนี้บรรจุอยู่ในบอร์ดไมโครคอนโทรลเลอร์แล้ว

สำหรับการสื่อสารข้อมูลแบบอะซิงโครนัส อัตราเร็วในการรับส่งข้อมูลทั้งตัวรับและตัวส่งจะต้องเท่ากันอัตราเร็วในการรับส่งข้อมูลนี้เรียกว่า อัตราบอด หรือ บอดเรต(Baudrate) มีหน่วยเป็น บิตต่อวินาที (bps)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการดำเนินงาน

4.1 อุปกรณ์ที่ใช้ในการทดสอบ

1. เครื่องคอมพิวเตอร์	1 เครื่อง
2. บอร์ดไมโครคอนโทรลเลอร์ตระกูล PIC 18F458	1 บอร์ด
3. Soft Ware โปรแกรมวิชวล เบสิก	1 ชุด
4. ประตูจำลอง	1 ชุด
5. iBUTTON	5 ตัว
6.จอแสดงผล LCD	1 อัน
7.แป้นพิมพ์	1 อัน

4.2 ขั้นตอนในการทดสอบ

จากการทดลองอ่านรหัส iBUTTON กับ socket เพื่อทำการอ่านค่านั้น ค่าที่ได้จากการอ่านนั้นมีความผิดพลาดขึ้นเป็นบางครั้ง ซึ่งค่าที่อ่านผิดพลาดนั้นอาจมาจากสาเหตุดังต่อไปนี้

1. หน้าสัมผัส iBUTTON มีความสกปรก
2. iBUTTON ที่ทำการสัมผัสกับ socket สัมผัสกันไม่พอดี
3. เกิดรอยขีดข่วนที่ iBUTTON
4. ตัวอุปกรณ์ iButton เกิดการเปียกชื้นทำให้หน้าสัมผัสจุดรับสัญญาณเกิดการช็อตกับกราวด์ที่ตัวซ็อกเก็ตรับสัญญาณ

หากรู้วิธีการใช้อุปกรณ์ iBUTTON อย่างถูกต้องแล้วปัญหาจากการอ่านข้อมูลผิดพลาดก็อาจจะลดน้อยลง และควรเก็บรักษาอุปกรณ์ iBUTTON ไว้ให้ปลอดภัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1 การทดสอบการเข้าห้อง

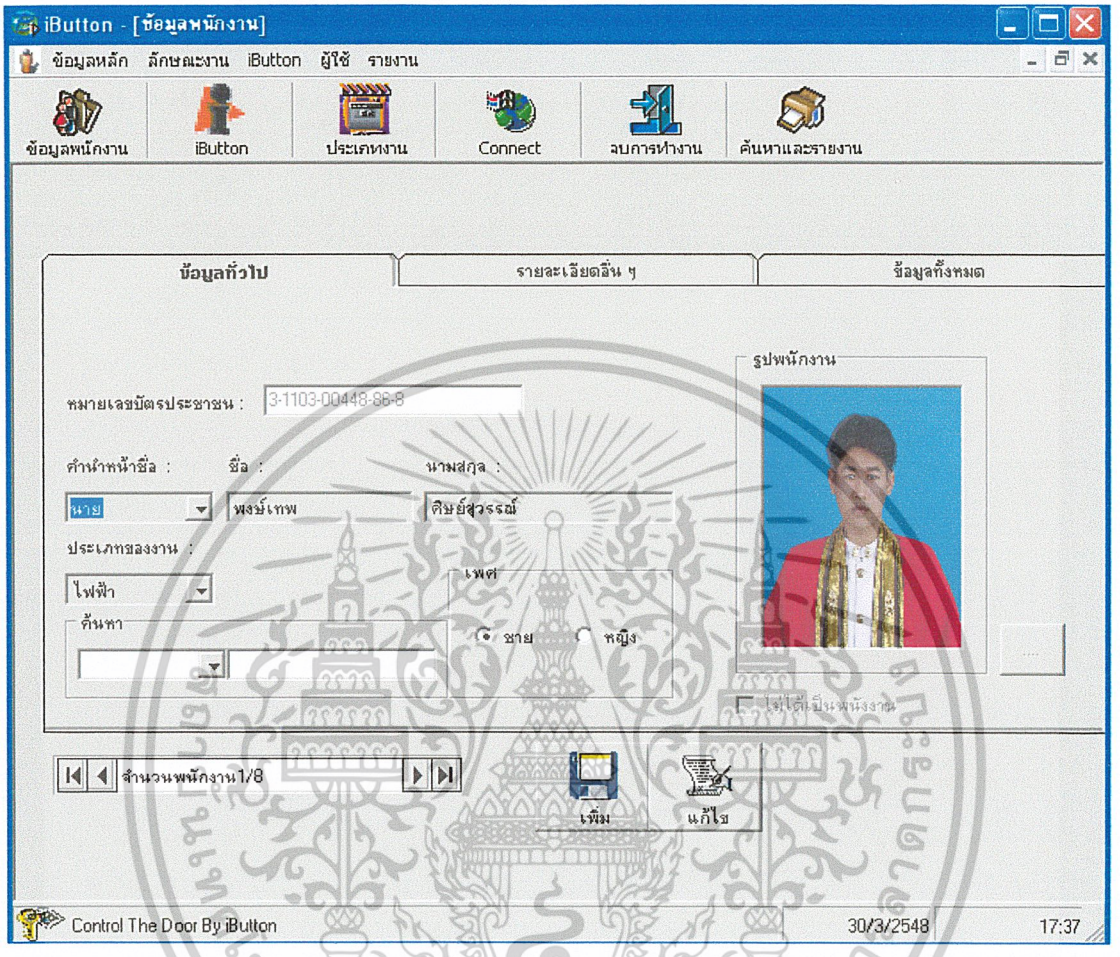
โดยการทดสอบจะทำงานแบบตามลำดับ โดยนำเอาอุปกรณ์ iBUTTON จำนวน 5 ตัว ซึ่งทำการบันทึกรหัสประจำตัวของตัวอุปกรณ์ iBUTTON ทั้ง 5 ตัวนี้ลงในฐานข้อมูลแล้ว ไปแตะที่ socket ที่ทำการอ่านรหัสประจำตัวของตัวอุปกรณ์ iBUTTON ทางด้านประตูเข้าซึ่งจะทำการอ่านรหัสประจำตัวโดยไมโครคอนโทรลเลอร์แล้วส่งค่ารหัสที่อ่านได้ไปให้คอมพิวเตอร์ทำการตรวจเช็คซึ่งถ้าไมโครคอนโทรลเลอร์อ่านรหัสถูกต้อง จะทำการนำรหัสนี้เข้าไปเปรียบเทียบกับฐานข้อมูลที่มีอยู่ ถ้าเป็นรหัส iBUTTON ที่ไม่มีอยู่ในฐานข้อมูลประตูก็จะไม่ทำงาน แต่ถ้าเป็นรหัส iBUTTON ที่มีอยู่ในฐานข้อมูลจริง ก็จะให้ทำการกดรหัสผ่านเพื่อป้องกันการนำ iBUTTON ผู้อื่นมาใช้ และเป็นการยืนยันการเป็นเจ้าของ iBUTTON ไมโครคอลโทรลเลอร์ก็จะนำรหัสผ่านที่ได้ไปทำการตรวจเช็คกับฐานข้อมูลอีกครั้ง ถ้ารหัสผ่านถูกต้องก็จะสั่งให้ประตูเปิดและหน่วงเวลา ซึ่งถ้าไม่มีการเดินผ่านประตูตามเวลาที่กำหนด sensor ก็จะไม่ทำงาน ประตูก็จะทำการปิด และจะไม่มีการบันทึกเวลาเข้าและผู้ใช้ iBUTTON แต่ถ้าประตูเปิดและมีการเดินผ่าน sensor ก็จะมีการบันทึกเวลาเข้าและผู้ใช้ iBUTTON รวมทั้งจำนวนผู้ที่เข้าไปในห้อง

4.2.2 การทดสอบการออกห้อง

ถ้าผู้ที่อยู่ภายในห้องต้องการออกจากห้อง สามารถทำได้โดยนำ iBUTTON แตะที่ socket ที่ทางด้านออก และไมโครคอลโทรลเลอร์ก็จะนำรหัสที่ได้จาก iBUTTON ไปทำการเปรียบเทียบกับฐานข้อมูลการใช้ประตู ถ้าเป็นรหัส iBUTTON ที่ผู้ใช้ใช้ทำการเปิดประตูเข้ามาภายในห้องประตูก็จะทำการเปิดและบันทึกเวลาที่ออก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

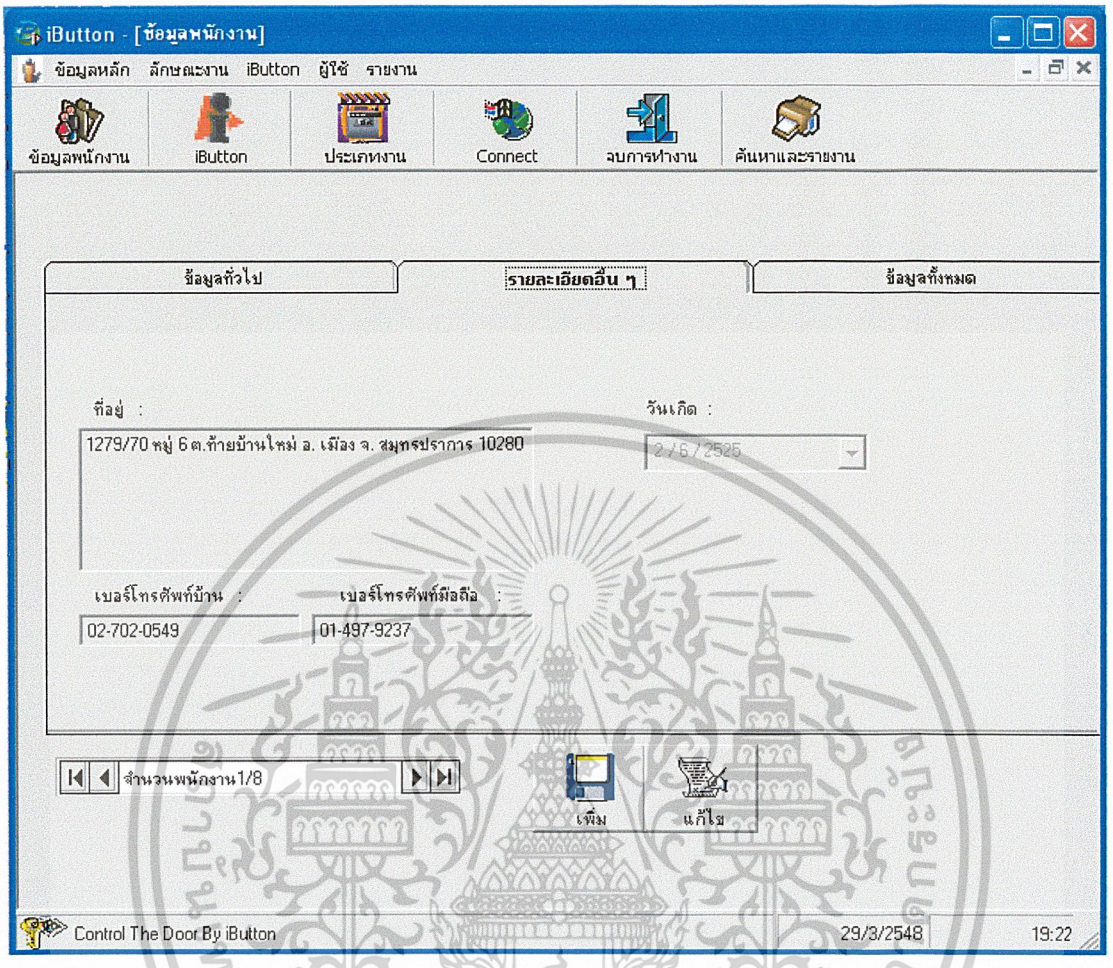
4.2.3 APPLICATION



รูปที่ 4.1 หน้าต่างข้อมูลทั่วไปของพนักงาน

จากรูปที่ 4.1 จะมีรายละเอียดข้อมูลของพนักงานต่างๆ เช่น รหัสบัตรประจำตัว ชื่อ นามสกุล ประเภทของงานที่พนักงานทำ เพศ โดยสามารถทำการ เพิ่ม และแก้ไขข้อมูลได้ โดยแจ้งกับผู้ดูแลระบบ

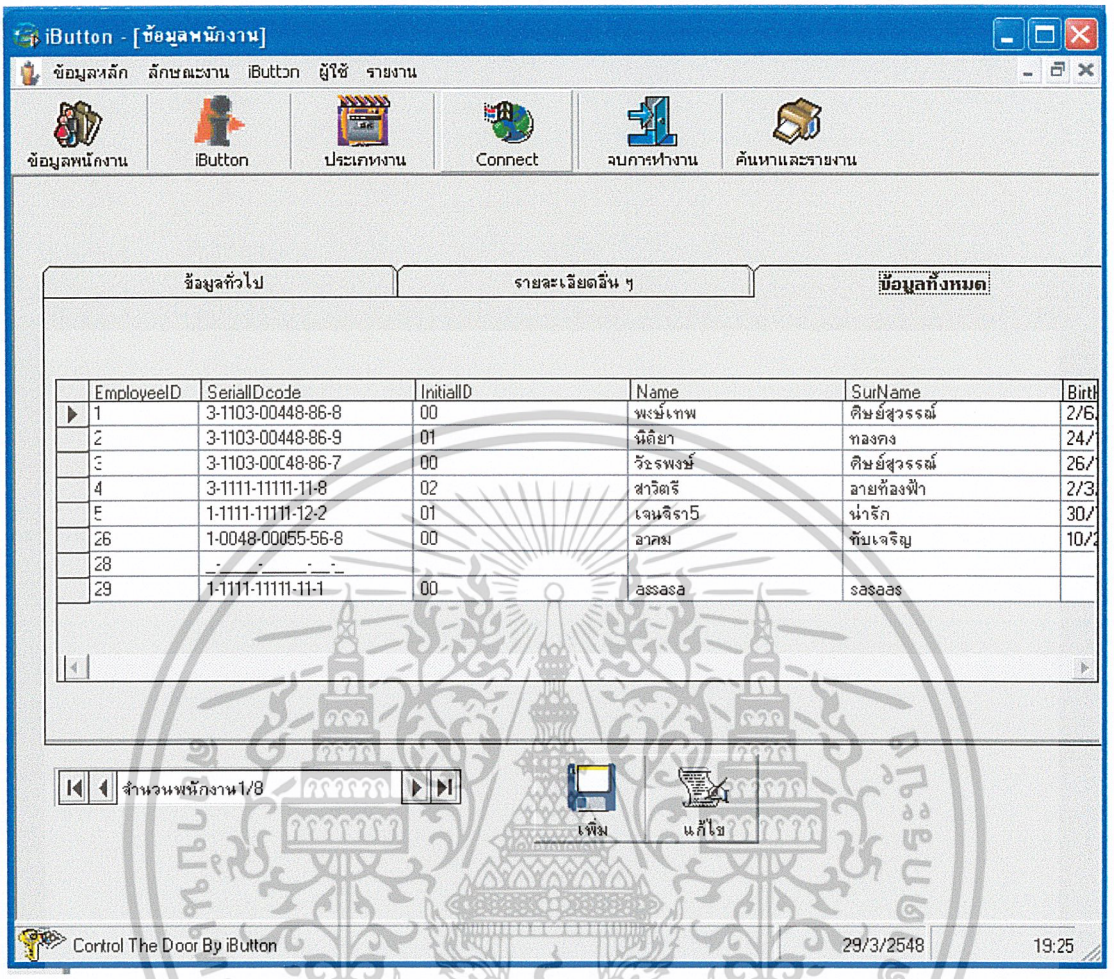
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 หน้าต่างรายละเอียดของพนักงาน

จากรูปที่ 4.2 จะแสดงที่อยู่ของพนักงาน เบอร์ติดต่อ และ วัน/เดือน/ปีเกิดของพนักงาน

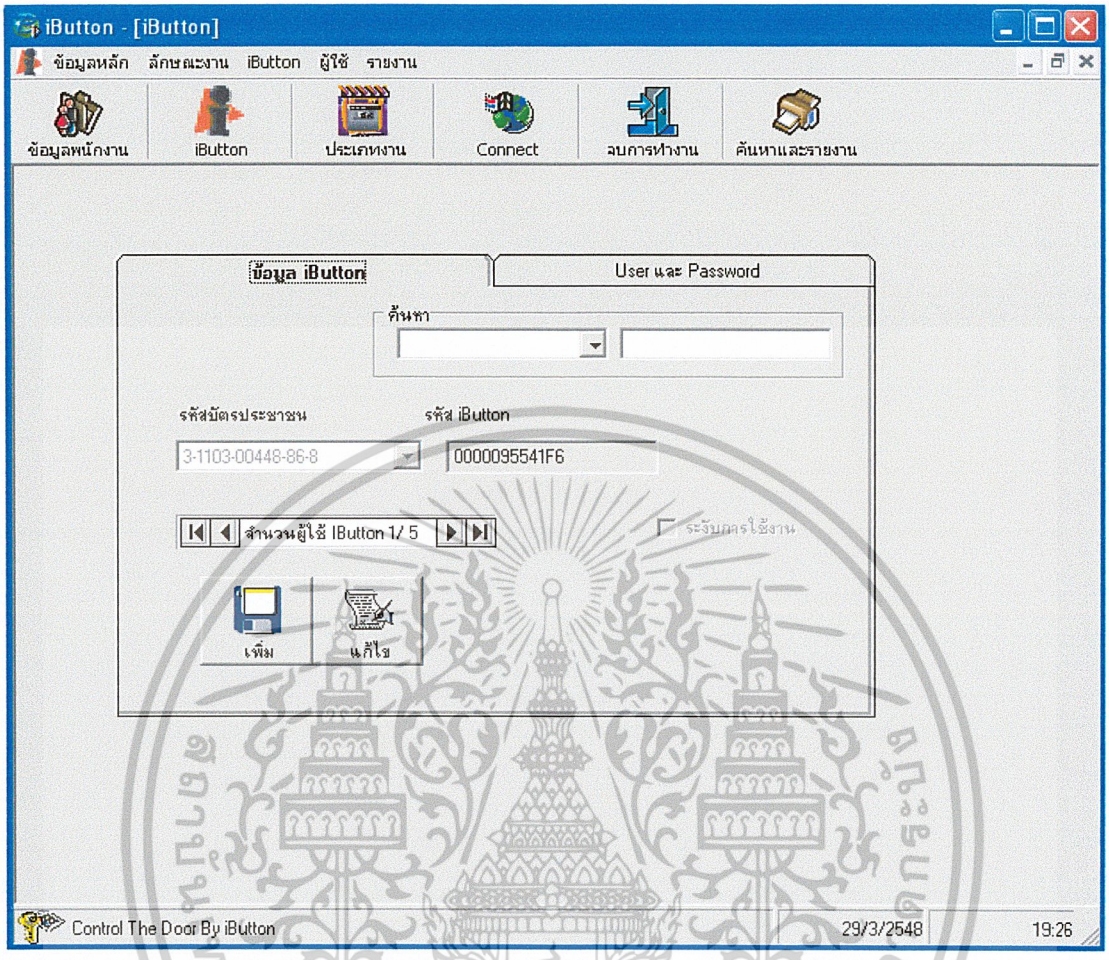
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่4.3 หน้าต่างข้อมูลทั้งหมดของพนักงาน

จากรูปที่ 4.3 จะแสดงรายละเอียดทั้งหมดของพนักงาน ตั้งแต่ ลำดับที่ หมายเลขบัตรประจำตัวประชาชน ชื่อ นามสกุล วัน/เดือน/ปีเกิด

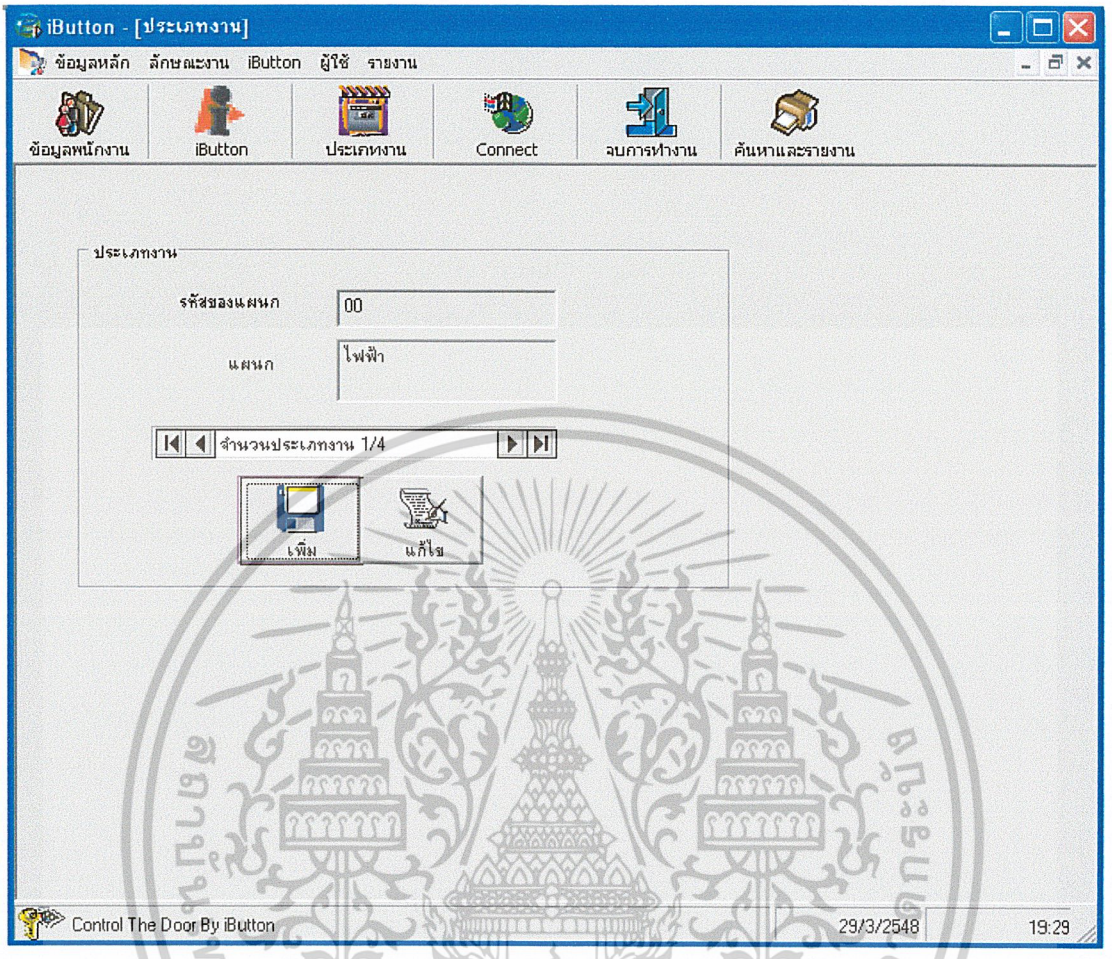
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 รหัสของ iBUTTON และรหัสบัตรประชาชน

จากรูปที่ 4.4 จะแสดงหมายเลขบัตรประจำตัวประชาชน และรหัสของอุปกรณ์ iBUTTON ของพนักงานแต่ละบุคคล โดยหน้าต่างนี้จะมีฟังก์ชันระงับการใช้อุปกรณ์ iBUTTON และถ้าจะใช้ iBUTTON ที่ถูกระงับการใช้งาน พนักงานเจ้าของอุปกรณ์ iBUTTON ต้องติดต่อผู้ดูแลระบบ

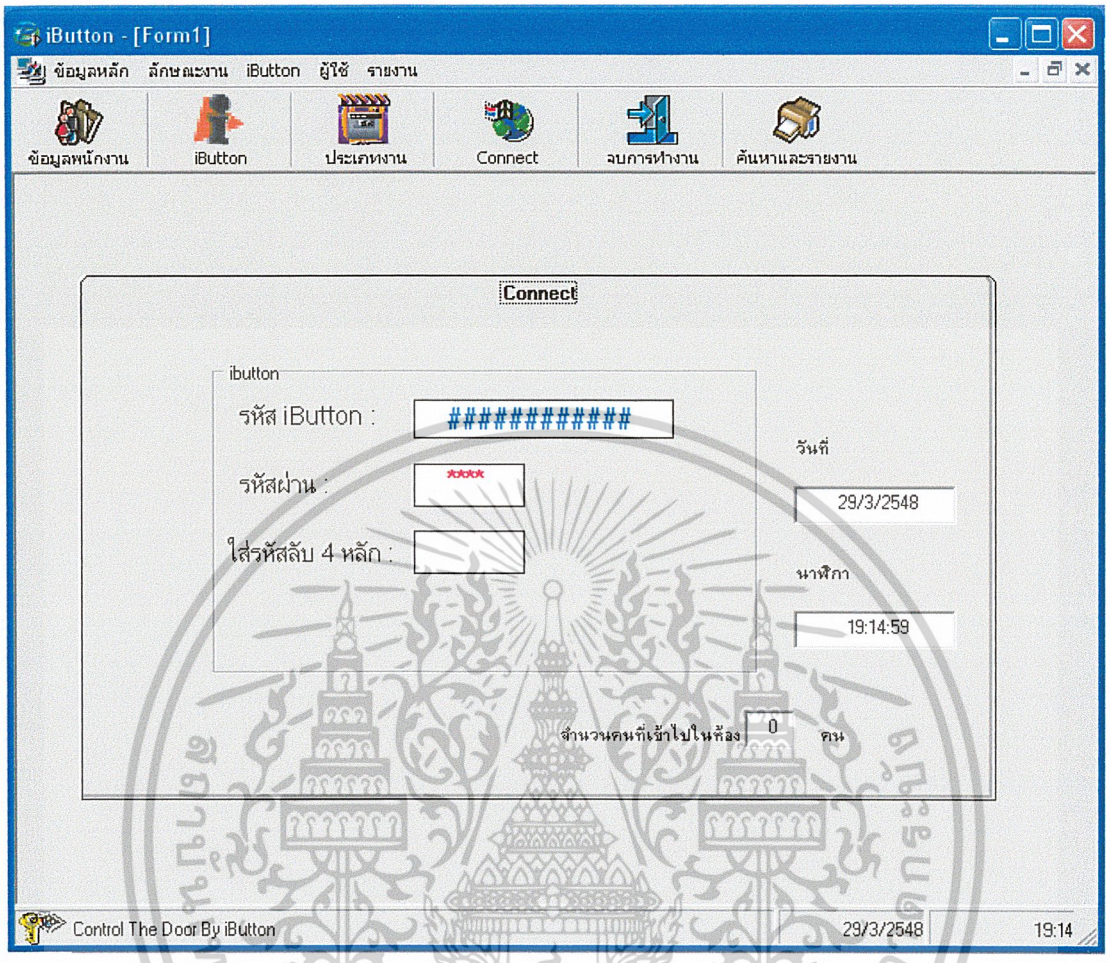
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 หน้าต่างประเภทของงาน

จากรูป 4.5 จะแสดงแผนกที่มีอยู่ทั้งหมด และสามารถเพิ่ม แก้ไข หรือเปลี่ยนแปลง ได้

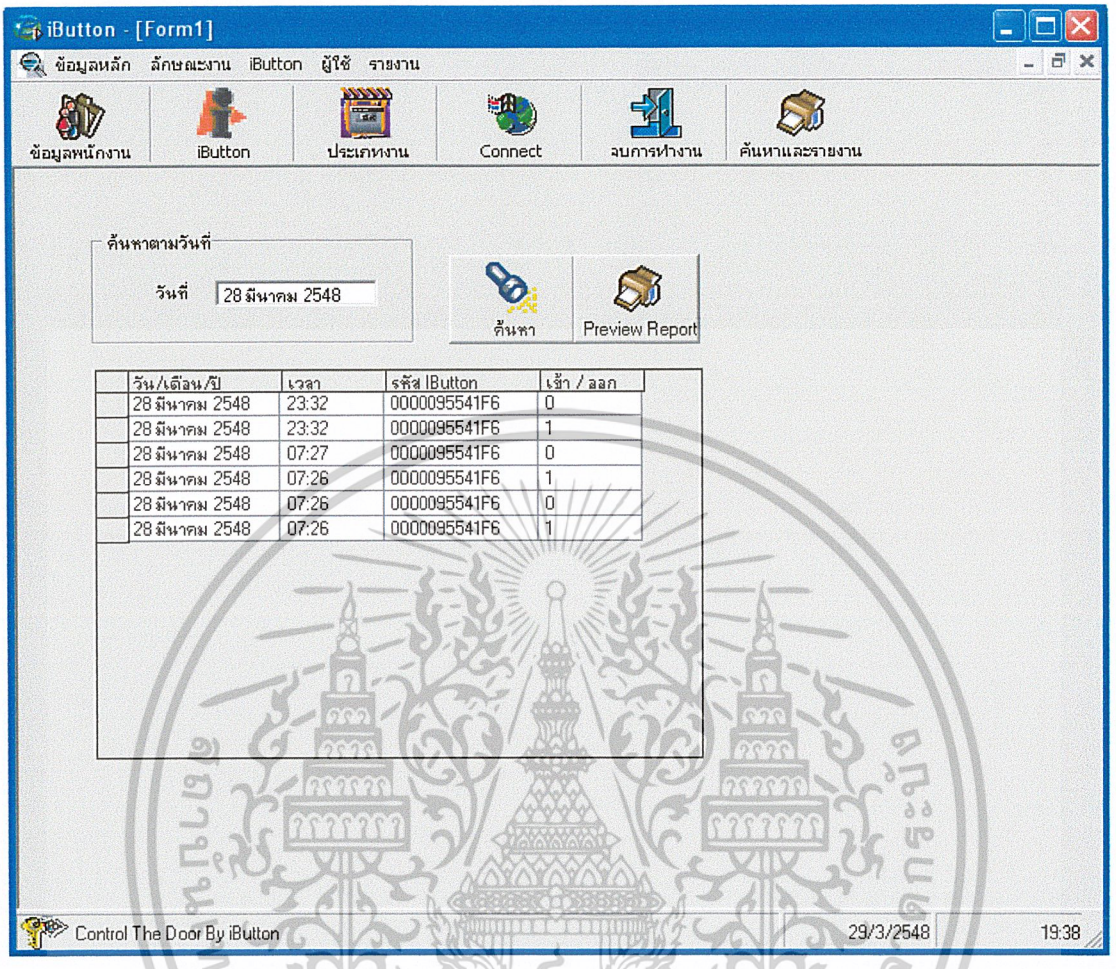
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 เป็นหน้าต่างที่ใช้ติดต่อกับฮาร์ดแวร์

จากรูป 4.6 จะเป็นการแสดงจำนวนคนที่อยู่ในห้องขณะนั้น ๆ ว่ามีคนเข้า / ออกห้องกี่คน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 เป็นหน้าต่างที่ใช้แสดงรายงานการเข้า / ออกประตู

จากรูป 4.7 เป็นการออกแบบรายงานแสดงเวลาและวันที่มีการเข้า / ออก ประตูตามรหัสประจำตัวของ iBUTTON สามารถค้นหาตามวันที่ที่เราต้องการได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลและข้อเสนอแนะ

ระบบรักษาความปลอดภัยโดยใช้ iBUTTON ในส่วนของตัวฮาร์ดแวร์ เกิดปัญหาในขณะทดลองหลายสาเหตุ เช่นเนื่องจากการอ่านรหัส iBUTTON ไม่ถูกต้องจากหลายสาเหตุดังที่กล่าวมาแล้วในบทก่อนหน้านี้ และอีกสาเหตุหนึ่งก็คือ การใส่ขั้วต่อของจอ LCD ผิดขั้ว ทำให้จอเสียหายได้ จึงควรตรวจสอบ pin ของจอ LCD ให้ถูกต้องก่อนที่จะต่อเข้ากับไมโครคอลโทรลเลอร์ และควรกำหนดความถี่ OSC ให้เหมาะสมการใช้งาน ถ้าจอแสดงผลเกิดการเสียหาย แก้ไขโดยการเปลี่ยนจอ LCD ใหม่และเปลี่ยนตัวต้านทานปรับค่าได้ที่ใช้ควบคุมความเข้มของจอภาพด้วย เนื่องจากเกิดการลัดวงจร ในส่วนโปรแกรมวิชวลเบสิกที่ใช้ในการสร้างฐานข้อมูลในเครื่องคอมพิวเตอร์สามารถทำงานได้ตามปกติ

5.1 สรุปผลการดำเนินงาน

5.1.1 ใช้อุปกรณ์ iButton เป็นกุญแจอิเล็กทรอนิกส์ในการเข้าออกห้องได้

5.1.2 สามารถใช้ไมโครคอนโทรลเลอร์ PIC18F458 ติดต่อกับวิชวลเบสิกได้

5.1.3 สามารถใช้โปรแกรมบันทึกเวลาเข้า-ออก จากห้องได้

5.1.4 สามารถใช้โปรแกรมเป็นตัวนับจำนวนพนักงานที่อยู่ในห้องได้

5.1.5 สามารถใช้โปรแกรมที่เขียนขึ้นเพิ่มและระงับผู้ใช้งานได้

5.1.6 มีระบบป้องกันการนำ iBUTTON ผู้อื่นมาใช้โดยการป้อนรหัสผ่าน

5.2 ปัญหาและแนวทางการแก้ไข

5.2.1 การเขียนโปรแกรมควบคุมไมโครคอลโทรลเลอร์ ส่วนของการรับข้อมูลจาก iBUTTON มีความผิดพลาด เนื่องจากการสัมผัสที่ไม่ไม่สนิทพอ หรืออาจมาจากอุปกรณ์ iBUTTON เกิดการสกปรก แก้ไขโดยรักษาอุปกรณ์ iBUTTON ให้สะอาดและสัมผัสกับSOCKET ให้สนิทกัน และขนาดของตัวอุปกรณ์ iButton มีผลต่อการอ่านค่า เนื่องจากตัวถังขนาด F3 มีความบางกว่าตัวถังขนาด F5 ทำให้การแตะที่ SOCKET เพื่อทำการอ่านค่ารหัสเฉพาะของตัวถังขนาด F3 มีค่าความผิดพลาดมากกว่า F5 เนื่องจากเนื้อที่ในการสัมผัสที่SOCKETเพื่อทำการอ่านค่ารหัสประจำตัวของตัวถังขนาด F3 สัมผัสโดนไม่เต็มที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.2 จอแสดงผลไม่แสดงผลตามที่กำหนดไว้ สาเหตุเนื่องจากการต่อ pin ผิดทำให้จอ LCD เกิดการเสียหาย และอีกสาเหตุหนึ่งคือการกำหนดความถี่ OSC ไม่เหมาะสม จึงควรตรวจดูความถี่ให้เหมาะสม

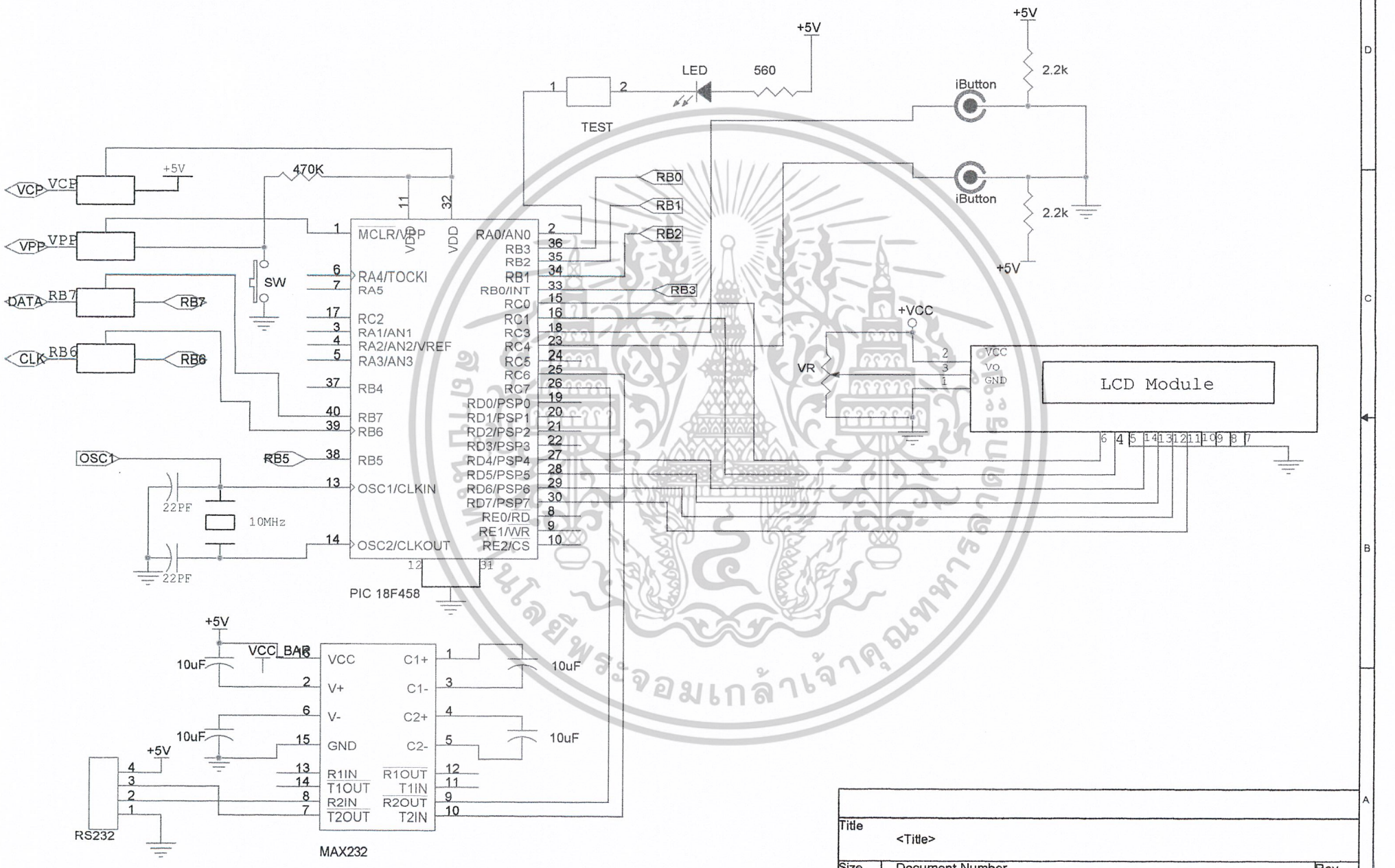
5.3 ข้อเสนอแนะและแนวทางพัฒนา

โครงงานนี้สามารถที่จะพัฒนาไปใช้ในการเป็นกุญแจอิเล็กทรอนิกส์ได้ เช่น อาจเป็นกุญแจ เปิดรถ บ้าน และเนื่องจากเทคโนโลยีที่ได้พัฒนาขึ้นเรื่อยๆ จึงทำให้เราจำเป็นต้องปรับปรุงตามไปด้วย โครงงานนี้สามารถที่จะปรับปรุงได้หลายส่วนด้วยกัน ส่วนของโปรแกรมระบบการจัดการบนคอมพิวเตอร์ สามารถที่จะเพิ่มรูปแบบการทำงานได้หลากหลายรูปแบบเพื่อให้รองรับกับความต้องการของผู้ใช้งาน ส่วนควบคุมอุปกรณ์หรือส่วนของคอนโทรลเลอร์สามารถที่จะเพิ่มการควบคุมได้หลายอย่างเพราะคอนโทรลเลอร์ตัวที่ใช้นั้นมีพอร์ตถึง 5 พอร์ต

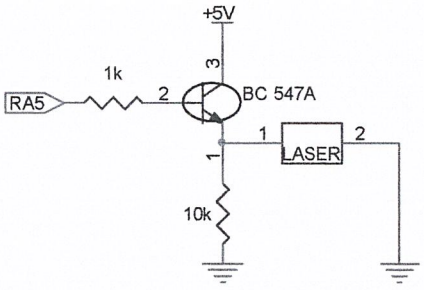
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

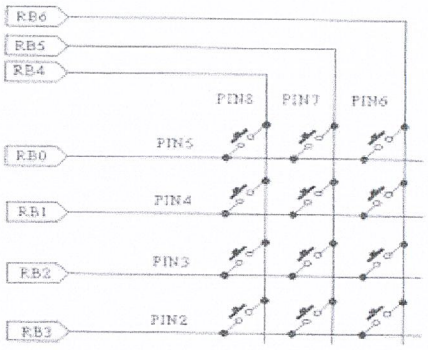
ภาคผนวก

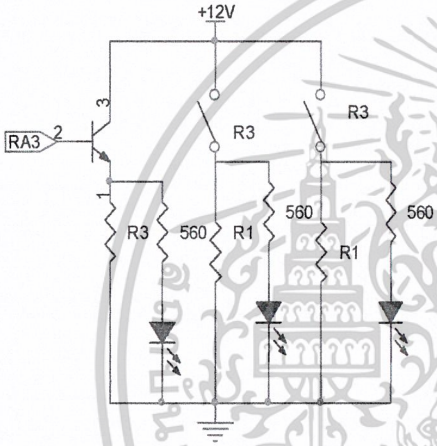
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

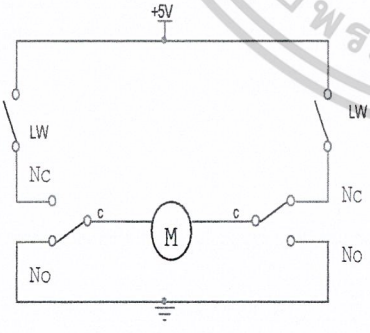


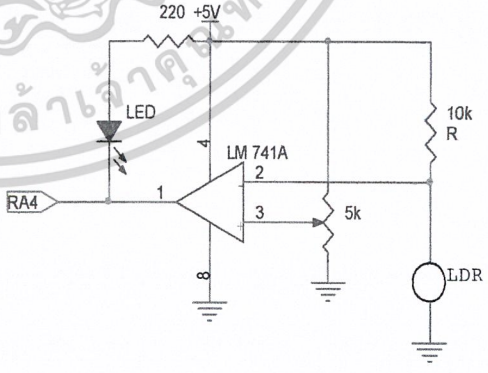
Title	<Title>	
Size A	Document Number <Doc>	Rev <RevCode>
Date: Wednesday, March 30, 2005	Sheet 1 of 1	











เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
INCLUDE "modedefs.bas"
DEFINE  OSC    40                          ' define oscillator 40 MHz

DEFINE  LCD_DREG      PORTD
DEFINE  LCD_DBIT      4
DEFINE  LCD_RSREG     PORTC
DEFINE  LCD_RSBIT     0
DEFINE  LCD_EREG      PORTC
DEFINE  LCD_EBIT      1
DEFINE  LCD_BIT       4

DEFINE SER2_BITS 8                        'Set number of data bits for Serin2 and Serout2
DEFINE SER2_ODD 1                         'Use odd parity instead of even parity

H             VAR   BYTE
INDEX         VAR   BYTE
KEYIN         VAR   BYTE
KEYOUT        VAR   BYTE
SHOW          VAR   BYTE
S             VAR   BYTE
SO            VAR   PORTC.6                ' Define serial output pin
SI            VAR   PORTC.7                ' Define serial input pin
STORE         VAR   WORD
STORER        VAR   BYTE
MODE          CON   0
MODE1         CON   1
MODE2         CON   2
RESPONSE      VAR   BYTE
I             VAR   WORD
J             VAR   BYTE
TRUE          CON   1
FALSE         CON   0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SENSOR_IN    VAR   PORTA.4
SENSOR_OUT   VAR   PORTA.5
MODE0        CON   0
C3           VAR   PORTC.3
C4           VAR   PORTC.4
Z            VAR   BIT
M            VAR   BYTE
N            VAR   BYTE
K            VAR   BYTE
CRC          VAR   BYTE
BVAR         VAR   BYTE
DAT          VAR   BYTE[8]

        LOW    INTCON2.7
        TRISE = %1111
        TRISB = %00001111                    'PORTB as input
        TRISC = %10111100
        TRISA =  $FF
        ADCON0 = $00
        ADCON1 = %10000111

START:      PAUSE 1000
            LCDOut $fe, 1                    ' Clear LCD screen
            LCDOut "    Welcome.    "

LOOP_CHECK:
            GOSUB IBUTTON_IN
            IF RESPONSE = TRUE THEN GOTO IN_DOOR
            GOSUB IBUTTON_OUT
            IF RESPONSE = TRUE THEN GOTO OUT_DOOR

            GOTO LOOP_CHECK
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
IN_DOOR:      SEROUT2  PORTC.6,84,["I@"]                    ' Send I@
              GOSUB SER_OUT
              SERIN  SI,T9600,1000,EXIT_IN_DOOR,STORER      ' Wait receive T
              IF STORER="T" THEN
                    pause  500
                    GOTO INSERT_PASS
              ENDIF
EXIT_IN_DOOR:  RETURN


OUT_DOOR:     SEROUT2  PORTC.6,84,["O@"]                    ' Send "O@"
              GOSUB SER_OUT
              SERIN  SI,T9600,1000,EXIT_OUT_DOOR,STORER     ' Wait receive T
              IF STORER="T" THEN
                    GOTO OPEN_DOOR
              ENDIF
EXIT_OUT_DOOR:  RETURN


INSERT_PASS:
              S=0
              LCDOut $fe, 1
              LCDOUT "INSERT_PASSWORD"
              LCDOut  $fe,$c0


LOOP_SCAN:    FOR H = 0 TO 2
                    LOOKUP H,[$E0,$D0,$B0],KEYOUT
                    PORTB = KEYOUT
                    KEYIN = PORTB & $0F
              IF KEYIN = $0F THEN NEXT_SCAN
              S = S+1
              PAUSE 300
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            KEYIN = KEYIN | KEYOUT
LOOKDOWN  KEYIN,[$EE,$DE,$BE,$ED,$DD,$BD,$EB,$DB,$BB,$E7,$D7,$B7],INDEX
LOOKUP   INDEX,[$01,$02,$03,$04,$05,$06,$07,$08,$09,$0A,$00,$0B],SHOW
            STORER=0
            SEROUT2   SO,84,[HEX SHOW]              ' Show password for computer
            LCDout "*"
            LCDOut $fe,$14
            IF S = 4 THEN GOTO CHK_PASSWORD
NEXT_SCAN:    NEXT
            GOTO LOOP_SCAN

CHK_PASSWORD:
            SERIN  SI,T9600,500,START,["P"],STORER      ' Wait receive "P"
            IF STORER="T" THEN
                LCDOut $fe, 1                           ' Clear LCD screen
                LCDOut "PASSWORD CORRECT"
                pause 2000
                GOTO OPEN_DOOR
            ELSE
                LCDOut $fe, 1                           ' Clear LCD screen
                LCDOut " PASSWORD WRONG "
                PAUSE 3000
            ENDIF
            RETURN

OPEN_DOOR:
            LCDOut $fe, 1                               ' Clear LCD screen
            LCDOut "   OPEN DOOR   "
            HIGH PORTA.3                                ' Control Moter
            GOSUB ON_SENSOR
            PAUSE 3000
            LOW PORTA.3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        LCDOut $fe, 1                   ' Clear LCD screen
        LCDOut "   CLOSE DOOR   "
        RETURN


IBUTTON_IN:
        OWOUT C3,MODE1,[$33]
        OWIN C3,MODE0,[STR DAT\8]
        GOSUB CRC_CHECK
        K=DAT[0]
        FOR M = 0 TO 6
                K=K|DAT[M]
        NEXT M
        IF (CRC=$00) AND (K=$00) THEN EXIT_IBUTTON_IN
        IF (CRC=$14) AND (K=$FF) THEN EXIT_IBUTTON_IN
        IF CRC<>DAT[7] THEN EXIT_IBUTTON_IN
        RESPONSE=TRUE
        RETURN

EXIT_IBUTTON_IN:
        RESPONSE=FALSE
        RETURN


IBUTTON_OUT:
        OWOUT C4,MODE1,[$33]
        OWIN C4,MODE0,[STR DAT\8]
        GOSUB CRC_CHECK
        K=DAT[0]
        FOR M = 0 TO 6
                K=K|DAT[M]
        NEXT M
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        IF (CRC=$00) AND (K=$00) THEN EXIT_IBUTTON_OUT
        IF (CRC=$14) AND (K=$FF) THEN EXIT_IBUTTON_OUT
        IF CRC<>DAT[7] THEN EXIT_IBUTTON_IN
        RESPONSE=TRUE
        RETURN

EXIT_IBUTTON_OUT:
        RESPONSE=FALSE
        RETURN

CRC_CHECK: CRC=00
        FOR N=0 TO 6
            K = DAT[N]
            FOR M=0 TO 7
                Z=CRC.0^K.0
                CRC.3=CRC.3^Z
                CRC.4=CRC.4^Z
                CRC=CRC>>1
                K=K>>1
                CRC.7 = Z
            NEXT M
        NEXT  N
        RETURN

SER_OUT:
        FOR  N=6 TO 1 STEP -1
        SEROUT2  PORTC.6,84,[HEX2 DAT(N)]          ' Send Code iButton
        PAUSE 50
        NEXT  N
        RETURN
```

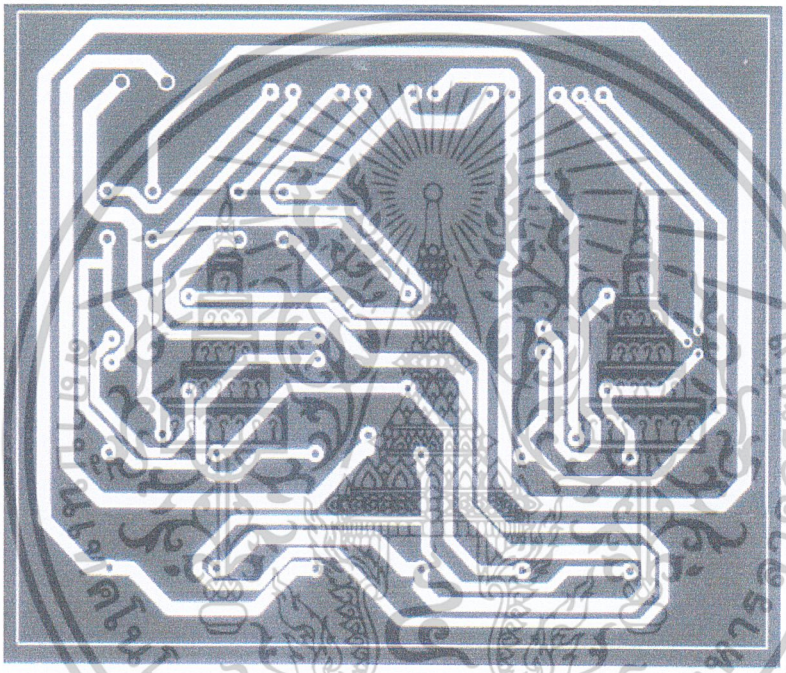
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ON_SENSOR:
        HIGH SENSOR_OUT                 ' High PortA.5
        PAUSE 100
        BVAR=0

        FOR I=1 TO 1000
            BUTTON SENSOR_IN,0,10,10,BVAR,1,COUNTER
            PAUSE 10
        NEXT I
        SEROUT2  PORTC.6,84,["F"]       ' Send "F"
        LOW SENSOR_OUT                  ' Low PortA.5
        RETURN

COUNTER:
        SEROUT2  PORTC.6,84,["T"]       ' Send "T"
        LOW SENSOR_OUT                  ' Low PortA.5
        RETURN
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. อภิชาติ ภู่พลับ. **เริ่มต้นเขียนโปรแกรมติดต่อและควบคุมฮาร์ดแวร์ ด้วย Vistual Basic.** นนทบุรี: อินโฟเพรส, 2546.
2. วัชรินทร์ เคารพ . **คู่มือการทดลอง PIC16F877 และ PIC18F458.** กรุงเทพฯ : อีทีที, 2546
3. ศุภชัย สมพานิช. **สร้างระบบงานฐานข้อมูลด้วย Vistual Basic ฉบับปรับปรุง.** นนทบุรี: อินโฟเพรส, 2546.
4. เจนวิทย์ เหลืองอร่าม และปิยวิทย์ เหลืองอร่าม. **การเขียนโปรแกรม สำหรับ Applications ด้วย Vistual Basic 6.** กรุงเทพฯ : ซีเอ็ดยูเคชั้น, 2543.
5. อรรถพล บุญยะโภคา และวรพจน์ ทรแก้ววัฒนกุลง **เรียนรู้และปฏิบัติการเชื่อมต่อด้วยคอมพิวเตอร์กับอุปกรณ์ภายนอกผ่าน พอร์ตอนุกรม.** กรุงเทพฯ: อิน โนเวตีฟ, 2544
6. กฤษดา ใจเย็น ณัฏฐพล วงศ์สุนทรชัย ชัยวัฒน์ ลิ้มพรจิตรวิไล **เรียนรู้และใช้งาน PICBASIC PRO คอมไพเลอร์** กรุงเทพฯ : อินโนเวตีฟ เอ็กเพอริเมนต์ ,2521
7. วัชรินทร์ เคารพ **เรียนรู้และเข้าใจไมโครคอลโทรลเลอร์ PIC ด้วยภาษาเบสิก** กรุงเทพฯ: อีทีที, 2537

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้