เครื่องบันทึกเวลาแบบ RFID

RFID TIME RECORDER

สุธินี พันธุ์ไพโรจน์

อรอนงค์ เพชรวิสัย

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมการวัดคุม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2556

เครื่องบันทึกเวลาแบบ RFID
RFID TIME RECORDER

สุธินี พันธุ์ไพโรจน์
อรอนงค์ เพชรวิสัย

ปริญญานี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมการวัดคุม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2556

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RFID TIME RECORDER

SUTHINEE PHANPHAIROT

ORN-ANONG PECHWISAI

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

BACHELOR OF ENGINEERING IN INSTRUMENTATION ENGINEERING

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

ACADEMIC YEAR 2013

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2556
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ใบรับรองปริญญานิพนธ์

.....

หัวข้อปริญญานิพนธ์	เครื่องบันทึกเวลาแบบ RFID RFID TIME RECORDER			
นักศึกษาผู้จัดทำ	สุธินี	พันธุ์ไพโรจน์	**รหัสนักศึกษา**	53011750
	อรอนงค์	เพชรวิสัย	**รหัสนักศึกษา**	53011894
ปริญญา	วิศวกรรมศาสตรบัณฑิต			
สาขาวิชา	วิศวกรรมการวัดคุม			
ปีการศึกษา	2556			

อาจารย์ผู้ควบคุมปริญญานิพนธ์	**ลายมือชื่อ**
อาจารย์สุธรรม สัทธรรมสกุล	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	เครื่องบันทึกเวลาแบบ RFID RFID TIME RECORDER	
นักศึกษาผู้จัดทำ	นางสาวสุธินี พันธุ์ไพโรจน์	รหัสนักศึกษา 53011750
	นางสาวอรอนงค์ เพชรวิสัย	รหัสนักศึกษา 53011894
อาจารย์ที่ปรึกษา	อาจารย์สุธรรม สัทธรรมสกุล	
ปีการศึกษา	2556	

บทคัดย่อ

โครงงานนี้ นำเสนอเครื่องบันทึกเวลาแบบ RFID ที่ใช้ไมโครคอนโทรเลอร์ตระกูล PIC เป็นตัวควบคุม โดยใช้บัตรประจำตัวนักศึกษาแบบ RFID มาใช้ยืนยันสิทธิ์ในการเข้าร่วมกิจกรรม ผ่านหน้าจอแสดงผลแบบแอลซีดี พร้อมทั้งจัดเก็บข้อมูล รหัสนักศึกษา วัน เวลา ที่ได้ลงใน SD Card สำหรับข้อมูลที่เก็บบันทึกใน SD Card สามารถนำข้อมูลมาใช้บนเครื่องคอมพิวเตอร์ได้

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	RFID TIME RECORDER	
Authors	Miss SUTHINEE	PHANPHAIROT
	Miss ORN-ANONG	PECHWISAI
Thesis Advisor	Mr. SUTHAM	SATTHAMSAKUL
Year	2013	

ABSTRACT

This project presents the RFID Time Recorder with PIC's microcontroller for controlling. This system uses the RFID student card for authentication of activity via LCD display. Student ID and date will be saved into the SD card. Data store in the SD card can be used on the computer.

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยดี เนื่องจากได้รับความอนุเคราะห์เครื่องมืออุปกรณ์ที่ใช้ในการดำเนินงาน ตลอดจนอบรมสั่งสอน แนะนำ และให้คำปรึกษาแนวทางในการดำเนินงานจาก อาจารย์สุธรรม สัทธรรมสกุล อาจารย์ผู้ควบคุมปริญญานิพนธ์ คณะผู้จัดทำซาบซึ้งในความกรุณาของท่านเป็นอย่างยิ่ง และขอกราบขอบพระคุณเป็นอย่างสูง

ขอกราบขอบพระคุณอาจารย์และเจ้าหน้าที่ภาควิชาวิศวกรรมการวัดคุม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุกท่าน ที่ได้ให้คำแนะนำและความช่วยเหลืออันเป็นประโยชน์ต่อการทำปริญญานิพนธ์ฉบับนี้

ขอกราบขอบพระคุณคุณพ่อคุณแม่ของคณะผู้จัดทำทุกท่าน ที่ท่านได้ให้ความอนุเคราะห์ในด้านค่าใช้จ่าย และสนับสนุนเป็นแรงบันดาลใจ ในการทำปริญญานิพนธ์ฉบับนี้ ประโยชน์ และคุณค่ารวมทั้งความดีทั้งปวง อันเกิดจากปริญญานิพนธ์ฉบับนี้ ขอมอบแด่บุพการี และผู้มีพระคุณทุกท่าน

คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....I
บทคัดย่อภาษาอังกฤษ.....II
กิตติกรรมประกาศ.....III
สารบัญ.....IV
สารบัญตาราง.....VIII
สารบัญภาพ.....IX

บทที่ 1 บทนำ.....1
1.1 ไมโครคอนโทรลเลอร์ตระกูล PIC.....1
1.2 วัตถุประสงค์ของปริญญานิพนธ์.....1
1.3 ขอบเขตของปริญญานิพนธ์.....1
1.4 ขั้นตอนการศึกษา.....1

บทที่ 2 ทฤษฎีและหลักการที่เกี่ยวข้อง.....2
2.1 ไมโครคอนโทรลเลอร์.....2
2.1.1 โครงสร้างโดยทั่วไปของไมโครคอนโทรลเลอร์.....2
2.1.1.1 หน่วยประมวลผลกลางหรือซีพียู.....3
2.1.1.2 หน่วยความจำ (Memory).....3
2.1.13 ส่วนติดต่อกับอุปกรณ์ภายนอกหรือพอร์ต (Port).....3
2.1.1.4 ช่องทางเดินของสัญญาณหรือบัส (BUS).....3
2.1.1.5 วงจรกำเนิดสัญญาณนาฬิกา (Clock).....3
2.1.2 ไมโครคอนโทรลเลอร์ตระกูลPIC.....3
2.1.2.1 ความเร็วของไมโครคอนโทรลเลอร์ตระกูล PIC.....4
2.1.2.2 หน่วยความจำของไมโครคอนโทรลเลอร์ตระกูล PIC.....4
2.1.2.3 สถาปัตยกรรมของไมโครคอนโทรลเลอร์ตระกูล PIC.....4
2.1.2.4 สรุปแนวคิดของไมโครคอนโทรลเลอร์ตระกูล PIC.....4
2.2 จอแสดงผลแบบผลึกเหลว (LCD: LIQUID CRYSTAL DISPLAY).....4
2.2.1 โครงสร้างภายในของตัวควบคุมโมดูลแอลซีดี.....5
2.3 RFID (Radio Frequency Identification).....6
2.3.1 องค์ประกอบของระบบ RFID.....6
2.3.1.1 ทรานสปอนเดอร์หรือแท็ก (Transponder/Tag).....6
2.3.1.2 เครื่องสำหรับอ่านหรือเขียนข้อมูล/.....8
ภายในแท็กด้วยคลื่นความถี่วิทยุ (Interrogator / Reader)
2.3.2 หลักการทำงานเบื้องต้นของระบบ RFID.....9
2.3.3 ความถี่ของคลื่นพาหะ.....9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

2.3.4 ความถี่ของคลื่นพาหะที่ใช้ในการสื่อสารข้อมูล....10
2.3.5 ระยะการรับส่งข้อมูลและกำลังส่ง....11
2.3.6 อัตราการรับส่งข้อมูล และ Bandwidth....12
2.3.7 ข้อเปรียบเทียบ RFID และบาร์โค้ด (Barcode)....12
2.4 รูปแบบการสื่อสารแบบ USART....13
2.4.1 รีจิสเตอร์ควบคุมการทำงานของโมดูล USART....13
2.4.1.1 รีจิสเตอร์ TXSTA....13
2.4.1.2 รีจิสเตอร์ RCSTA....14
2.4.2 โหมดการทำงานของโมดูล USART....16
2.4.2.1 การทำงานในโหมดอะซิงโครนัส....16
2.4.2.2 การส่งข้อมูลในโหมดอะซิงโครนัส....16
2.4.2.3 การรับข้อมูลในโหมดอะซิงโครนัส....17
2.5 Real Time Clock....19
2.6 I^2C (Inter Integrate Circuit Bus)....21
2.6.1 การเชื่อมต่ออุปกรณ์แบบ I^2C BUS....22
2.6.2 การเขียน-อ่านข้อมูลกับอุปกรณ์แบบ I^2C BUS....22
2.6.2.1 สถานะบัสว่าง....23
2.6.2.2 การกำหนดสถานะเริ่มต้นและสิ้นสุดของ I^2C BUS (START and STOP Conditions)....23
2.6.2.3 รหัสควบคุมของ I^2C BUS (Control Byte)....23
2.6.2.4 ช่วงเวลารับ-ส่งบิตข้อมูลของ I^2C BUS....24
2.7 Secure Digital Card (SD Card)....24
2.7.1 ประวัติของ SD Card....24
2.7.2 คุณสมบัติของ SD Card....25

บทที่ 3 การดำเนินงาน....27
3.1 บทนำ....27
3.2 รายละเอียดขององค์ประกอบต่างๆ....27
3.2.1 RFID MODULE....27
3.2.1.1 การนำ RFID MODULE ไปต่อใช้งานกับ PIC18F458....28
3.2.1.2 รูปแบบการติดต่อสื่อสาร (COMMUNICATION PROTOCAL)....29
3.2.2 ไมโครคอนโทรลเลอร์ (MICROCONTROLLER)....31
3.2.3 DS1307....32
3.2.3.1 การรับส่งข้อมูลแบบ I^2C....33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

3.2.3.2 สถานะของการรับส่งข้อมูลแบบ I^2C 33
3.2.4 SD CARD....34
3.2.5 โมดูลแอลซีดีขนาด 16 ตัวอักษร 2 บรรทัด (LCD 16x2)....35
3.3 รายละเอียดในการออกแบบ....36
3.3.1 การออกแบบส่วนฮาร์ดแวร์....36
3.3.1.1 วงจรของ RFID MODULE....36
3.3.1.2 วงจรของ DS1307....37
3.3.1.3 วงจรของ SD Card....37
3.3.1.4 วงจรของแอลซีดีโมดูล....37
3.3.2 การออกแบบส่วนของซอฟต์แวร์....38
3.3.2.1 การทำงานของเครื่องบันทึกเวลาแบบ RFID....38
3.3.2.2 การตั้งค่าวันและเวลา....39

บทที่ 4 การทดลองและผลการทดลอง....41
4.1 บทนำ....41
4.2 การทดลองและผลการทดลองของการอ่าน ID ของบัตรนักศึกษา/ วัน และเวลา แสดงผ่านจอแอลซีดี....41
4.2.1 การทดลองการอ่าน ID ของบัตรนักศึกษา วัน และเวลา แสดงผ่านจอแอลซีดี....41
4.2.2 ผลการทดลองการอ่าน ID ของบัตรนักศึกษา วัน และ เวลา แสดงผ่านจอแอลซีดี....41
4.3 การทดลองและผลการทดลองเก็บไฟล์ใน SD card....42
4.3.1 การทดลองเก็บไฟล์ใน SD card....42
4.3.2 ผลการทดลองเก็บไฟล์ใน SD card....42
4.4 การทดลองและผลการทดลองของการตั้งค่าวันและเวลา....44
4.4.1 การทดลองการตั้งค่าวันและเวลา....44
4.4.2 ผลการทดลองการตั้งค่าวันและเวลา....44

บทที่ 5 สรุปผล ปัญหาและข้อเสนอแนะ....48
5.1 สรุปผล....48
5.2 ปัญหาและแนวทางแก้ไข....48
5.3 ข้อเสนอแนะ....48

บรรณานุกรม....49

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

ภาคผนวก ก.....50
ภาคผนวก ข.....59

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ หน้า

2.1 ย่านความถี่ต่างๆ ของระบบ RFID และการใช้งาน....11
2.2 แสดงบิตต่างๆ ในรีจิสเตอร์ TXSTA....13
2.3 แสดงบิตต่างๆ ในรีจิสเตอร์ RCSTA....15
2.4 การควบคุมความถี่ออสซิลเลเตอร์ด้วยการเซตบิต RS1, RS....21
2.5 ตารางแสดงรายละเอียดโครงสร้างขาของ SD Card....26
3.1 แสดงรายละเอียดขาของ RFID MODULE....29
3.2 Command Overview....30
3.3 Status Overview....30
3.4 รูปแบบการส่งข้อมูล SL025M to Host....31
3.5 รูปแบบการส่งข้อมูล SL025M to Host....31

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
2.1 โครงสร้างโดยทั่วไปของไมโครคอนโทรลเลอร์	2
2.2 ไดอะแกรมการทำงานของโมดูลแอลซีดีแบบอักขระ	6
2.3 RFID แท็กในรูปแบบต่างๆ	7
2.4 รูปตัวอย่าง Active Tag ที่มีแบตเตอรี่ Lithuim 2 ก้อนอยู่ภายใน	7
2.5 บล็อกไดอะแกรมของ Passive tag	8
2.6 โครงสร้างภายในเครื่องอ่าน	9
2.7 ความถี่ย่านที่ระบบ RFID ถูกใช้งาน	10
2.8 RFID และบาร์โค้ด	12
2.9 ไดอะแกรมส่งข้อมูลในโหมดอะซิงโครนัสของโมดูล USART ใน PIC18F458	17
2.10 ไดอะแกรมรับข้อมูลในโหมดอะซิงโครนัสของโมดูล USART ใน PIC18F458	18
2.11 ไอซี RTC DS1307	19
2.12 การเชื่อมต่อ DS1307 เข้ากับไมโครคอนโทรลเลอร์ด้วยระบบบัสแบบ I^2C	19
2.13 การเขียนข้อมูลอุปกรณ์สลาฟผ่านบัส I^2C	20
2.14 การอ่านข้อมูลจากอุปกรณ์สลาฟผ่านบัส I^2C	20
2.15 รีจิสเตอร์ภายในฐานเวลา DS1307	21
2.16 ลักษณะการเชื่อมต่ออุปกรณ์แบบ I^2C BUS	22
2.17 รูปแบบการเขียนอ่านข้อมูลแบบ I^2C BUS	22
2.18 การกำหนดสถานะเริ่มต้นและสิ้นสุดของ I^2C BUS	23
2.19 รหัสควบคุม I^2C BUS	23
2.20 ช่วงเวลารับ-ส่งบิตข้อมูลของ I^2C BUS	24
2.21 Secure Digital Cards (SD Card)	24
2.22 SD Card ของ Panasonic ขนาด 64 MB	25
2.23 โครงสร้างขาของ SD Card	26
3.1 โครงสร้างการทำงานของเครื่องบันทึกเวลาแบบ RFID	27
3.2 RFID MODULE ของยี่ห้อ StrongLink รุ่น SL025M	28
3.3 แสดงลักษณะของ RFID MODULE ของยี่ห้อ StrongLink รุ่น SL025M	29
3.4 การจัดขาของ PIC18F458	32
3.5 ตำแหน่งขาไอซี RTC DS1307	33
3.6 การรับ-ส่งข้อมูลผ่านบัส I^2C	34
3.7 SD Card	34
3.8 แอลซีดีขนาด 16x2	35
3.9 การจัดขาของแอลซีดี	35
3.10 วงจรรวมของเครื่องบันทึกเวลาแบบ RFID	36
3.11 วงจรของ RFID MODULE	36
3.12 วงจรของ DS1307	37

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่		หน้า
3.13	วงจรของ SD Card	37
3.14	วงจรของแอลซีดีโมดูล	37
3.15	แผนผังการทำงานของเครื่องบันทึกเวลาแบบ RFID	39
3.16	แผนผังการตั้งค่าวันและเวลา	40
4.1	เครื่องบันทึกเวลาแบบ RFID แสดงข้อความเพื่อรอรับการแท็กบัตร RFID	41
4.2	เครื่องบันทึกเวลาแบบ RFID อ่านข้อมูลจากบัตร RFID และแสดง ID ของบัตรพร้อมเวลาที่ทำการแตะบัตร	42
4.3	ไฟล์ที่จัดเก็บใน SD card	43
4.4	ข้อมูลในไฟล์ที่ถูกจัดเก็บในแต่ละวัน	43
4.5	กดปุ่ม SET และปุ่ม ENTER พร้อมกัน	44
4.6	จอแอลซีดีแสดงการตั้งค่าวัน	44
4.7	แสดงการกดปุ่ม SET เพื่อตั้งค่าวันที่	45
4.8	แสดงการกดปุ่ม ENTER เพื่อเลื่อนไปยังการตั้งค่าเดือน	45
4.9	จอแอลซีดีแสดงการตั้งค่าเวลา	46
4.10	แสดงการกดปุ่ม SET เพื่อตั้งค่าชั่วโมง	46
4.11	แสดงการกดปุ่ม ENTER เพื่อเลื่อนไปยังการตั้งค่านาที	47
4.12	แสดงการกดปุ่ม ENTER เพื่อกลับสู่หน้าจอปกติ	47

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความสำคัญของปริญญานิพนธ์

ในการทำกิจกรรมของนักศึกษาในปัจจุบัน มีการเก็บข้อมูลโดยการให้นักศึกษาเซ็นชื่อเพื่อยืนยันสิทธิ์ในการเข้าร่วมกิจกรรม ซึ่งวิธีการดังกล่าวนี้มีปัญหาในขั้นตอนของการรวบรวมข้อมูล เนื่องจากปัจจัยหลายๆประการ อาทิเช่น เอกสารที่ใช้ลงทะเบียนสูญหาย ข้อมูลที่ได้ไม่ชัดเจน และต้องใช้เวลาในการนำข้อมูลมาจัดทำเป็นเอกสาร เป็นต้น จึงเป็นที่มาของการทำเครื่องบันทึกเวลาแบบ RFID โดยนำคุณสมบัติเด่นของ RFID คือเป็นเทคโนโลยีที่ใช้ในการระบุข้อมูลที่แสดงถึงเอกลักษณ์ของวัตถุ หรือบุคคลด้วยคลื่นความถี่วิทยุ ซึ่งสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังได้ใช้บัตรนักศึกษาที่เป็นแบบ RFID อยู่แล้ว จึงสามารถที่จะนำมาใช้ร่วมกับเครื่องบันทึกเวลาแบบ RFID เพื่อเก็บข้อมูลในการทำกิจกรรมของนักศึกษาทั้งภายในและภายนอกสถาบันได้ทันที และแก้ปัญหาต่างๆที่กล่าวมาได้ทั้งหมด สำหรับข้อมูลที่ได้จะจัดเก็บลงใน SD card ซึ่งสามารถนำข้อมูลมาใช้บนเครื่องคอมพิวเตอร์ได้ โดยไม่ต้องเสียเวลาในการจัดทำเอกสารซ้ำอีก

1.2 วัตถุประสงค์ของปริญญานิพนธ์

1. เพื่อศึกษาการทำงานของระบบ RFID (Radio Frequency Identify)
2. เพื่อศึกษาการเขียนโปรแกรมควบคุมไมโครคอนโทรลเลอร์
3. เพื่อใช้บันทึกวัน เวลา การทำกิจกรรมของนักศึกษา

1.3 ขอบเขตของปริญญานิพนธ์

1. สามารถออกแบบเพื่อสร้างเครื่องบันทึกวัน เวลา ในการทำกิจกรรมของนักศึกษาโดยใช้บัตรนักศึกษาแบบ RFID
2. สามารถเก็บบันทึก วัน เวลา ลงในหน่วยความจำสำรองแบบ SD card ได้
3. รูปแบบไฟล์ที่ถูกจัดเก็บใน SD card สามารถนำมาใช้กับเครื่องคอมพิวเตอร์ได้

1.4 ขั้นตอนการศึกษา

1. ศึกษาหลักการอ่านและเขียนข้อมูลลงบัตร RFID
2. ศึกษาการเขียนโปรแกรมภาษา C
3. ศึกษาการเขียนโปรแกรมติดต่อระหว่าง PIC และเครื่องอ่าน RFID
4. ศึกษาการเขียนโปรแกรมติดต่อระหว่าง PIC และ SD Card
5. ศึกษาการเขียนโปรแกรมติดต่อระหว่าง PIC และ Real Time Clock
6. ศึกษาการเขียนโปรแกรมเพื่อนำข้อมูลที่ได้ไปแสดงผลในส่วนของจอแอลซีดี
7. ศึกษาการเขียนโปรแกรมเพื่อนำข้อมูลที่ได้ไปเก็บลงใน SD Card ในรูปแบบไฟล์ .csv

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการที่เกี่ยวข้อง

2.1 ไมโครคอนโทรลเลอร์

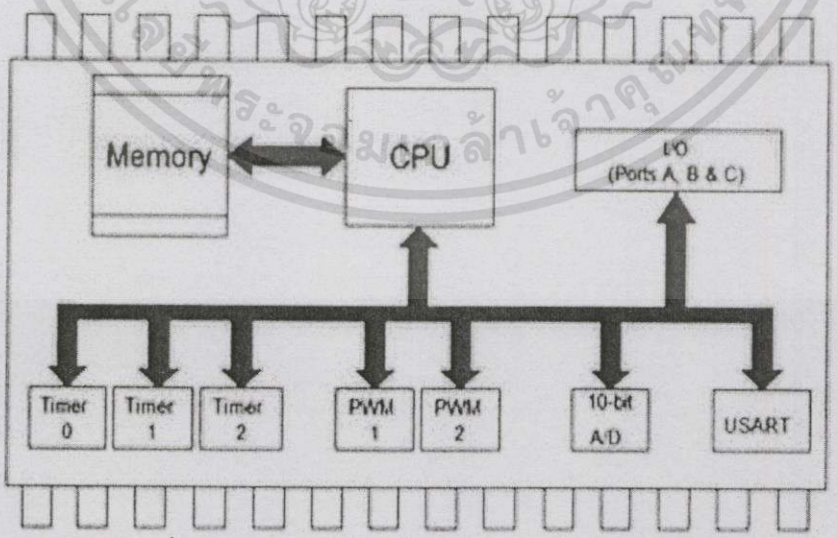
ไมโครคอนโทรลเลอร์ คือ อุปกรณ์ประเภทสารกึ่งตัวนำที่รวบรวมฟังก์ชั่นการทำงานต่างๆ ไว้ภายในตัวของมันเอง ภายในประกอบด้วยหน่วยรับข้อมูลและโปรแกรม หน่วยประมวลผล หน่วยความจำหน่วยแสดงผลซึ่งมีความสมบูรณ์ในตัวของมันเอง ทำให้มีขนาดเล็กและสามารถเขียนโปรแกรมควบคุมการทำงานของอุปกรณ์ต่างๆ ที่เชื่อมต่อกับตัวมัน ง่ายต่อการนำไปประยุกต์ใช้งาน

ความแตกต่างของไมโครคอนโทรลเลอร์กับไมโครคอมพิวเตอร์ คือ ไมโครคอนโทรลเลอร์นั้นมีความสมบูรณ์ภายในตัวของมันเองคือมีส่วนประกอบต่างๆ ครบถ้วน ส่วนไมโครคอมพิวเตอร์นั้นต้องทำงานร่วมกับอุปกรณ์ข้างเคียงที่เชื่อมต่อจากภายนอก เช่น แป้นพิมพ์ เครื่องอ่านเขียนแผ่นบันทึก หน่วยความจำ I/O ฯลฯ

ข้อแตกต่างระหว่างไมโครโปรเซสเซอร์กับไมโครคอนโทรลเลอร์ ไมโครโปรเซสเซอร์ที่ใช้อยู่ในปัจจุบันจะไม่มีหน่วยความจำ RAM, ROM และ I/O Port อยู่ในตัว ทำให้ต้องต่อหน่วยความจำโปรแกรมภายนอกเพิ่มและต้องใช้ ICs ขยายพอร์ตเพิ่มเติม ข้อดีคือสามารถเพิ่มหน่วยความจำได้ตลอด ส่วนไมโครคอนโทรลเลอร์จะมีวงจรพื้นฐานประกอบอยู่ภายในชิพ เช่น หน่วยความจำ RAM, ROM และ I/O Port เป็นต้น ดังนั้นในระบบไมโครคอนโทรลเลอร์จึงมีขนาดที่เล็กกว่าและมีราคาที่ต่ำกว่าระบบของไมโครโปรเซสเซอร์

2.1.1 โครงสร้างโดยทั่วไปของไมโครคอนโทรลเลอร์

โครงสร้างโดยทั่วไปของไมโครคอนโทรลเลอร์นั้น สามารถแบ่งออกมาได้เป็น 5 ส่วน ดังรูปที่ 2.1



รูปที่ 2.1 โครงสร้างโดยทั่วไปของไมโครคอนโทรเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.1.1 หน่วยประมวลผลกลางหรือซีพียู (CPU: Central Processing Unit)

เป็นวงจรอิเลคทรอนิคส์ที่ทำงาน หรือ ประมวลผลตามชุดของคำสั่งของเครื่องจากซอฟต์แวร์ เปรียบเสมือนเป็นสมองของคอมพิวเตอร์ในการทำหน้าที่ตัดสินใจหรือคำนวณจากคำสั่งที่ได้รับมา เช่น การเปรียบเทียบการกระทำการทางคณิตศาสตร์ฯลฯ โดยมีกระบวนการพื้นฐานคือ

- อ่านชุดคำสั่ง (fetch)
- ตีความชุดคำสั่ง (decode)
- ประมวลผลชุดคำสั่ง (execute)
- อ่านข้อมูลจากหน่วยความจำ (memory)
- เขียนข้อมูล/ส่งผลการประมวลกลับ (write back)

2.1.1.2 หน่วยความจำ (Memory)

สามารถแบ่งเป็น 2 ส่วน คือ หน่วยความจำที่มีไว้สำหรับเก็บโปรแกรม (Program Memory) เปรียบเสมือนฮาร์ดดิสก์ของเครื่องคอมพิวเตอร์คือ ข้อมูลใดๆ ที่ถูกเก็บไว้ในนี้จะไม่สูญหายไปแม้ไม่มีไฟเลี้ยง อีกส่วนหนึ่งคือ หน่วยความจำข้อมูล (Data Memory) ใช้เป็นที่พักข้อมูลชั่วคราว ขณะทำงานหากไม่มีไฟเลี้ยงข้อมูลก็จะหายไปคล้ายกับหน่วยความจำแรม (RAM) ในเครื่องคอมพิวเตอร์ทั่วไป แต่สำหรับไมโครคอนโทรลเลอร์สมัยใหม่หน่วยความจำข้อมูลจะมีทั้งที่เป็นหน่วยความจำแรม ซึ่งข้อมูลจะหายไปเมื่อไม่มีไฟเลี้ยงและเป็นอีอีพรอม (EEPROM : Erasable Electrically Read-Only Memory) ซึ่งสามารถเก็บข้อมูลได้แม้ไม่มีไฟเลี้ยง

2.1.1.3 ส่วนติดต่อกับอุปกรณ์ภายนอกหรือพอร์ต (Port)

มี 2 ลักษณะ คือ พอร์ตอินพุต (Input Port) และพอร์ตเอาต์พุต (Output Port) ส่วนนี้จะใช้ในการเชื่อมต่อกับอุปกรณ์ภายนอกซึ่งถือว่าเป็นส่วนที่สำคัญมาก

2.1.1.4 ช่องทางเดินของสัญญาณหรือบัส (BUS)

คือ เส้นทางการแลกเปลี่ยนสัญญาณข้อมูลระหว่างซีพียู หน่วยความจำ และพอร์ตโดยแบ่งเป็นบัสข้อมูล (Data Bus), บัสแอดเดรส (Address Bus) และบัสควบคุม (Control Bus)

2.1.1.5 วงจรกำเนิดสัญญาณนาฬิกา (Clock)

นับเป็นส่วนประกอบที่สำคัญมากอีกส่วนหนึ่ง เนื่องจากการทำงานที่เกิดขึ้นในตัวไมโครคอนโทรลเลอร์จะขึ้นอยู่กับการกำหนดจังหวะ หากสัญญาณนาฬิกามีความถี่สูงจังหวะการทำงานก็จะสามารถทำได้ถี่ขึ้น ส่งผลให้ไมโครคอนโทรลเลอร์ตัวนั้นมีความเร็วในการประมวลผลสูงตามไปด้วย

2.1.2 ไมโครคอนโทรลเลอร์ตระกูล PIC

PIC ย่อมาจากคำว่า Peripheral Interface Controller คือ ไมโครคอนโทรลเลอร์อีกตระกูลหนึ่ง แนวคิดของไมโครคอนโทรลเลอร์ตระกูลนี้ ก็คือ พยายามรวมเอาทุกอย่างเอาไว้ในตัวของมัน ไม่ว่าจะเป็น PROGRAM MEMROY, RAM, EEPROM, SERIAL, I2C, PWM, A/D ฯลฯ โดยไม่จำเป็นต้องต่ออุปกรณ์เสริมจากภายนอก ในตัวของ PIC จะมีฟังก์ชั่นที่ใช้ในการประมวลผลรวมทั้งหน่วยความจำทำให้มันเหมือนกับซีพียูตัวหนึ่งเลยทีเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2.1 ความเร็วของไมโครคอนโทรลเลอร์ตระกูล PIC

ภาคของความถี่ของสัญญาณนาฬิกา ปัจจุบันสามารถทำสัญญาณนาฬิกาได้ที่ 20 MHz ซึ่งทำให้หนึ่งคำสั่งของ PIC ใช้เวลาเพียง 0.25 us แต่อย่างไรก็ตาม มีบริษัทอื่นได้ซื้อลิขสิทธิ์ PIC จากไมโครชิพและได้สร้างชิพที่มีความเร็วได้มากกว่าเดิมขึ้นไปอีก

2.1.2.2 หน่วยความจำของไมโครคอนโทรลเลอร์ตระกูล PIC

ในอดีตหน่วยความจำของ PIC จะมีค่อนข้างน้อย คือ อยู่ระหว่าง 512 words ถึง 4K words แต่ในปัจจุบันบริษัทไมโครชิพซึ่งเป็นเจ้าของ PIC ได้พัฒนาจนทำให้หน่วยความจำของ PIC มีขนาดเป็นหลายสิบกิโลไบต์ และมีแนวโน้มว่าจะขยายได้ใหญ่ขึ้นเรื่อย ๆ ในเรื่องของการนับขนาดของหน่วยความจำของ PIC จะนับไม่เหมือนปกติ โดยที่หนึ่งคำสั่งของ PIC จะมีขนาด 14บิต ดังนั้นจะเรียกว่า 1 word ของ PIC จะมีขนาด 14 บิต เช่น PIC16F877A ระบุได้ว่ามีหน่วยความจำ 2 กิโลไบต์ซึ่งหมายถึง 2 Kword ถ้าคำนวณให้เป็นในรูปแบบ 1 ไบต์ = 8 บิตจะได้ว่า 1 x 1,024 x 14 = 14,336 บิต ดังนั้นก็คือ 14,336/(8 x 1,024) =1.75 กิโลไบต์นั่นเอง

2.1.2.3 สถาปัตยกรรมของไมโครคอนโทรลเลอร์ตระกูล PIC

ตอนนี้มี 3 สายหลัก สมัยก่อนมีแค่ 2 คือ ขึ้นต้นด้วย 16xxx, 17xxx และใหม่ล่าสุดคือ 18xxx ถ้าพูดถึงคุณสมบัติที่เหนือกว่าเรียงจากน้อยสุดไปมากสุดก็คือ 16 -> 17 -> 18 คำสั่งแอสแซมบลีของ 17 และมี 18 จะมีมากกว่า 16 ทำให้เขียนโปรแกรมได้ง่ายกว่าราคาก็จะสูงกว่าด้วย แต่ที่เป็นที่นิยมก็คือตระกูล 16xxx

2.1.2.4 สรุปแนวคิดของไมโครคอนโทรลเลอร์ตระกูล PIC

จะยึดถือการออกแบบที่ว่ารวมทุกอย่างไว้ในชิพตัวเดียว โดยไม่ต้องต่ออุปกรณ์ใดๆ เพิ่มเติม ผลที่ตามมาก็คือแผ่นวงจรจะมีขนาดเล็กและอุปกรณ์ที่ใช้จะไม่มาก บางงานอาจจะใช้แค่ PIC เพียงตัวเดียวโดยไม่ต้องใช้ชิพอื่นมาเพิ่มเติมเลย นี่คือคุณสมบัติพิเศษของไมโครคอนโทรลเลอร์ตระกูล PIC ซึ่งปัจจุบันหลายบริษัทที่ผลิตไมโครคอนโทรลเลอร์ก็เริ่มจะหันมาเลียนแบบแนวทางนี้ แต่ทุกอย่างย่อมมีข้อดีและข้อเสีย เนื่องจากแนวคิดที่จะรวมทุกอย่างไว้ในชิพเดียว ทำให้หน่วยความจำโปรแกรมและหน่วยความจำข้อมูลไม่สามารถขยายโดยใช้กับหน่วยความจำภายนอกได้ PIC จึงเหมาะสำหรับงานเล็กๆ ไม่ใช่งานใหญ่ๆ ที่ต้องใช้การคำนวณและหน่วยความจำมากๆ

2.2 จอแสดงผลแบบผลึกเหลว (LCD: LIQUID CRYSTAL DISPLAY)

ในอุปกรณ์เครื่องมือเครื่องใช้ทางด้านอิเล็กทรอนิกส์มากมาย ในปัจจุบันได้มีการติดตั้งจอแสดงผลแบบผลึกเหลว หรือ LCD (LIQUID CRYSTAL DISPLAY) สำหรับใช้ในการแสดงผลการทำงานเพียงอย่างเดียวหรือแสดงผลในลักษณะของเมนูโต้ตอบเพื่อสั่งงานเครื่องกับผู้ใช้งาน โมดูลจอแสดงผลแบบแอลซีดีจึงถือเป็นอุปกรณ์แสดงผลประเภทหนึ่งที่มีจุดเด่นในหลายประการ เช่น สามารถแสดงเป็นตัวอักขระแบบต่างๆ หรือสัญลักษณ์รูปภาพทำให้สื่อสารเข้าใจง่าย มีน้ำหนักเบาและสามารถออกแบบเป็นรูปแบบต่างๆ ได้ ราคาถูกและกินกำลังไฟในขณะทำงานน้อย ด้วยข้อดีเหล่านี้จึงทำให้จอแสดงผลแบบแอลซีดีถูกนำมาใช้งานกันอย่างแพร่หลาย โดยเฉพาะในเครื่องมือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อิเล็กทรอนิกส์ขนาดเล็ก ซึ่งมีพื้นที่จำกัดและบ่อยครั้งที่ต้องใช้กำลังไฟจากแบตเตอรี่ จอแสดงผลแอลซีดีจึงเป็นตัวเลือกที่ดีที่สุดจากจอแสดงผลในทุกประเภท

ในโมดูลแอลซีดี จะมีส่วนประกอบหลักๆ 3 ส่วนดังนี้

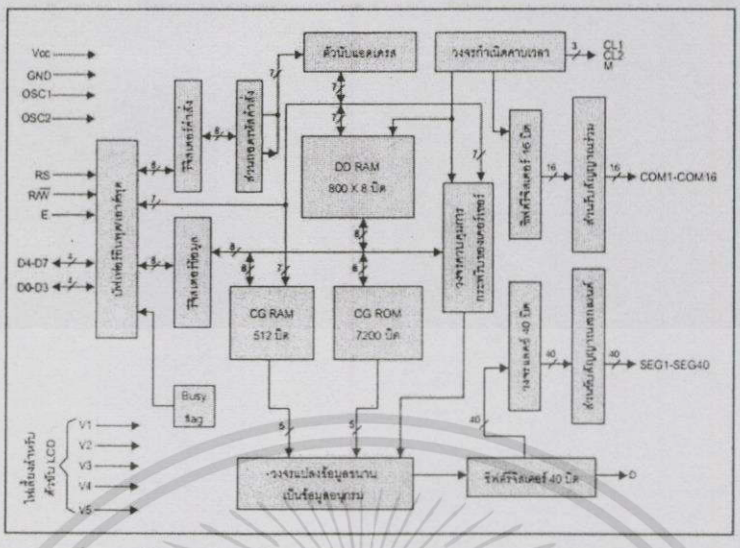
- ตัวแสดงผล (DISPLAY) ภายในเป็นผลึกเหลวที่สามารถแสดงผลให้เห็นโดยอาศัยแสงจากภายนอก ดังนั้นจึงต้องมีมุมในการมองข้อมูลที่แสดงผลบนจอแอลซีดี
- ตัวควบคุม (CONTROLLER) เป็นตัวรับข้อมูลจากอุปกรณ์ภายนอกมาควบคุมการทำงานของโมดูลแอลซีดี เช่น จอลบภาพ แสดงตัวอักษร หรือเลื่อนเคอร์เซอร์ เป็นต้น ตัวควบคุมนี้ใช้ชิพควบคุมโดยเฉพาะ ชิพที่นิยมใช้ก็คือ เบอร์ HD44780 และ HD61830 โดย HD44780 จะใช้ควบคุมแอลซีดีแบบอักขระ ส่วน HD61830 ใช้ควบคุมแอลซีดีแบบกราฟิก
- ตัวขับ (DRIVER) เป็นตัวรับสัญญาณจากตัวควบคุมมาขับให้ตัวแสดงผลแสดงข้อมูลตามที่กำหนด ชิพที่ใช้ทำหน้าที่เป็นตัวขับนี้ได้แก่ เบอร์ HD4410H และ MSM5259 เป็นต้น

2.2.1 โครงสร้างภายในของตัวควบคุมโมดูลแอลซีดี

ในการใช้งานโมดูลแอลซีดีจำเป็นต้องทำความเข้าใจเกี่ยวกับโครงสร้างและคำสั่งที่ใช้ในการควบคุมให้ดีเสียก่อน ในที่นี้จะขอยกตัวอย่างโมดูลแอลซีดีแบบอักขระ เพราะสามารถเข้าใจได้ง่าย โดยรูปที่ 2.2 เป็นบล็อกไดอะแกรมภายในของชิพควบคุมแอลซีดี เบอร์ HD44780 ซึ่งใช้ในโมดูลแอลซีดีแบบอักขระ ประกอบด้วย

- บัฟเฟอร์อินพุตเอาต์พุต เป็นส่วนที่ใช้ในการติดต่อรับส่งข้อมูลกับอุปกรณ์ภายนอก เพื่อที่จะถ่ายทอดข้อมูลเข้าออกภายในตัวควบคุม
- รีจิสเตอร์คำสั่ง (INSTRUCTION REGISTER:IR) เป็นรีจิสเตอร์ใช้รับข้อมูลคำสั่งจากอุปกรณ์ภายนอก เพื่อถ่ายทอดไปยังหน่วยความจำที่ทำหน้าที่เก็บข้อมูลแสดงผล หรือนำข้อมูลไปสร้างตัวอักษรเพิ่มเติมในแรมเก็บตัวอักษร
- แรมเก็บข้อมูลแสดงผล (DISPLAY DATA RAM: DDRAM) เป็นหน่วยความจำแรมทำหน้าที่เก็บข้อมูลที่มาจากรีจิสเตอร์ DR ตัวควบคุมจะนำข้อมูลใน DDRAM นี้ไปเปิดตาราง (LOOK UP- LABLE) ของตัวอักษรที่เก็บไว้ในหน่วยความจำรอมและแรมเก็บตัวอักษร เพื่อนำไปแสดงที่ตัวแสดงผล
- รอมเก็บตัวอักษร (CHARACTER GENERATOR ROM: CGROM) เป็นหน่วยความจำรอมที่ใช้เก็บข้อมูลตัวอักษรหรือสัญลักษณ์ ที่สามารถอ่านออกไปแสดงที่ตัวแสดงผลได้ มีขนาด 7200 บิต โดยจะถูกอ่านด้วยค่าของข้อมูลใน DDRAM
- แรมเก็บตัวอักษร (CHARACTER GENERATOR RAM: CGRAM) เป็นหน่วยความจำแรมที่ใช้เก็บอักษรที่มีการสร้างเพิ่มเติมขึ้นใหม่ ในกรณีที่ตัวอักษรใน CGROM ไม่เพียงพอ มีขนาด 512 บิต การเขียนและการอ่านค่าไปใช้นั้นทำได้เช่นเดียวกับ CGROM คือเขียนข้อมูลลงใน DDRAM แล้วตัวควบคุมจะมาอ่านค่าจาก CGRAM เอง
- แฟลก BUSY เป็นส่วนที่ทำหน้าที่แจ้งสถานะการทำงานของตัวควบคุมให้อุปกรณ์ภายนอกทราบว่า ตัวควบคุมพร้อมที่จะรับข้อมูลหรือคำสั่งหรือไม่ ดังนั้นก่อนการส่งข้อมูลหรือคำสั่งมายังตัวควบคุม ต้องตรวจสอบสถานะของแฟลก BUSY นี้เสียก่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 ไดอะแกรมการทำงานของโมดูลแอลซีดีแบบอักขระ

2.3 RFID (Radio Frequency Identification)

RFID ย่อมาจาก Radio Frequency Identification หรือ ระบบชี้เฉพาะอัตโนมัติ (Automatic Identification) แบบไร้สาย (Wireless) เป็นระบบระบุเอกลักษณ์ของวัตถุด้วยคลื่นความถี่วิทยุ แนวความคิดในการนำคลื่นวิทยุมาใช้เพื่อแสดงตำแหน่งหรือแสดงตนเอง ได้เกิดขึ้นตั้งแต่ราวประมาณปลายสงครามโลกครั้งที่สอง โดยการนำข้อมูลที่ต้องการส่งมาทำการมอดูเลต (Modulation) กับคลื่นวิทยุแล้วส่งออกผ่านทางสายอากาศที่อยู่ในตัวรับข้อมูล ซึ่งตั้งแต่ปี ค.ศ. 1980 เป็นต้นมา RFID ได้ถูกพัฒนาเพื่อวัตถุประสงค์หลัก คือ การนำไปใช้แทนระบบบาร์โค้ด (Barcode) และเนื่องจากอุปกรณ์ RFID ในขณะนั้นไม่สะดวกที่จะนำมาใช้งานเพราะมีขนาดใหญ่ นอกจากนั้นยังมีราคาแพง จึงไม่ได้รับความนิยมมากเพียงพอที่จะนำมาใช้ในเชิงพาณิชยกรรม บริหารและเชิงราชการ ต่อมาจึงได้มีการพัฒนา RFID อย่างต่อเนื่อง จนสามารถลดขนาดให้เป็นแผ่นเล็กๆ (Chip) ได้ดังในปัจจุบัน

2.3.1 องค์ประกอบของระบบ RFID

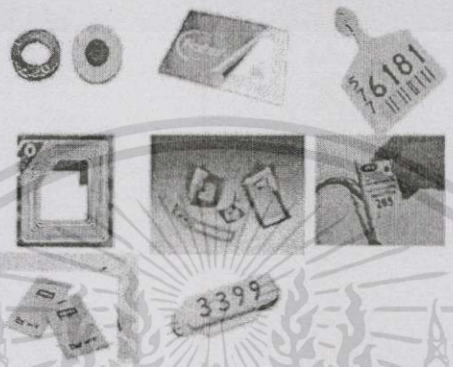
ระบบ RFID จะมีองค์ประกอบหลักอยู่สองส่วน โดยส่วนแรกคือ ทรานสปอนเดอร์ หรือแท็ก (Transponder / Tag) ที่ใช้ติดกับวัตถุต่างๆ ที่ต้องการ โดยแท็กจะบันทึกข้อมูลเกี่ยวกับวัตถุชิ้นนั้นๆไว้ ส่วนที่สองคือ เครื่องสำหรับอ่านหรือเขียนข้อมูลภายในแท็กด้วยคลื่นความถี่วิทยุ (Interrogator / Reader) โดยการทำงานนั้นเครื่องอ่านจะทำหน้าที่จ่ายกำลังงานในรูปคลื่นความถี่วิทยุให้กับตัวบัตร ยังผลให้วงจรอิเล็กทรอนิกส์ภายในสามารถส่งข้อมูลจำเพาะที่แสดงถึง "Identity" กลับมาประมวลผลที่ตัวอ่านได้

2.3.1.1 ทรานสปอนเดอร์หรือแท็ก (Transponder / Tag)

มาจากคำว่า “ทรานสมิตเตอร์” (Transmitter) ผสมกับคำว่า “เรสปอนเดอร์” (Responder) โดยโครงสร้างภายในของแท็กจะประกอบด้วย 2 ส่วนใหญ่ๆ ได้แก่ ขดลวดขนาดเล็กซึ่งทำหน้าที่เป็นสายอากาศ (Antenna) และไมโครชิพ (Microchip) ซึ่งขดลวดขนาดเล็กที่ทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าที่เป็นสายอากาศนั้น จะใช้สำหรับส่งสัญญาณคลื่นความถี่วิทยุและสร้างพลังงานป้อนให้ส่วนของไมโครชิป (Microchip) ที่ทำหน้าที่เก็บข้อมูลต่างๆ ของวัตถุนั้นๆ โดยทั่วไปตัวแท็กอาจจะอยู่ในรูปแบบที่เป็นได้ทั้งกระดาษ แผ่นฟิล์ม พลาสติกที่มีขนาดและรูปร่างแตกต่างกันออกไป ทั้งนี้ขึ้นอยู่กับวัสดุที่จะนำแท็กไปติดและมีได้หลายรูปแบบ เช่น ขนาดเท่าบัตรเครดิต เหรียญกระดุม ฉลากสินค้า แคปซูล เป็นต้น

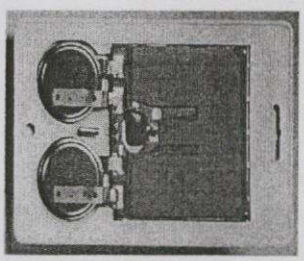


รูปที่ 2.3 RFID แท็กในรูปแบบต่างๆ

แท็กที่มีการใช้งานกันอยู่นั้นจะมีอยู่ 2 ชนิดใหญ่ๆ โดยแต่ละชนิดก็จะมีความแตกต่างกันในแง่ของการใช้งาน ราคา โครงสร้างและหลักการทำงานอยู่ ซึ่งจะสามารถแยกออกเป็นหัวข้อดังนี้

- Active Tag

แท็กชนิดนี้จะต้องอาศัยแหล่งจ่ายไฟจากแบตเตอรี่ภายนอก เพื่อจ่ายพลังงานให้กับวงจรภายในทำงาน เราจะสามารถทั้งอ่านและเขียนข้อมูลลงในแท็กชนิดนี้ได้ และการที่ต้องใช้แบตเตอรี่จึงทำให้แท็กชนิดแอ็กทีฟมีอายุการใช้งานจำกัดตามอายุของแบตเตอรี่ เมื่อแบตเตอรี่หมดก็ต้องนำแท็กไปทิ้ง ไม่สามารถนำกลับมาใช้ใหม่ได้ อย่างไรก็ตามถ้าเราสามารถออกแบบวงจรของแท็กให้กินกระแสไฟน้อยๆ ก็อาจจะมีอายุการใช้งานนานนับสิบปี โดยแท็กชนิดนี้สามารถมีหน่วยความจำภายในขนาดใหญ่ได้ถึง 1 เมกกะไบต์และสามารถอ่านข้อมูลได้ระยะไกลสูงสุดประมาณ 10 เมตร ซึ่งไกลกว่าแท็กชนิดพาสซีฟ อีกทั้งยังมีกำลังส่งสูงและยังสามารถทำงานในบริเวณที่มีสัญญาณรบกวนได้ดีอีกด้วย แม้ว่าแท็กชนิดนี้จะมีข้อดีอยู่ในหลายๆส่วน แต่ก็มีข้อเสียด้วยเช่นกัน คือ มีราคาต่อหน่วยแพงและมีขนาดค่อนข้างใหญ่และมีอายุการใช้งานจำกัดดังที่ได้กล่าวมาแล้วข้างต้น

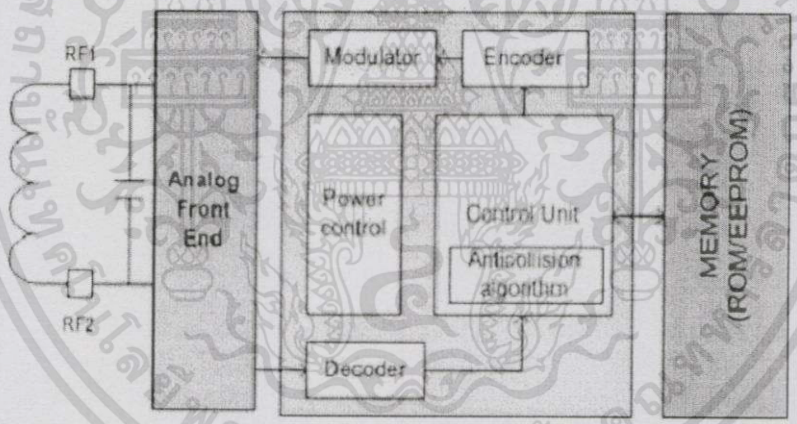


รูปที่ 2.4 รูปตัวอย่าง Active Tag ที่มีแบตเตอรี่ Lithium 2 ก้อนอยู่ภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Passive Tag

แท็กชนิดนี้ ไม่จำเป็นต้องอาศัยแหล่งจ่ายไฟจากภายนอก เพราะภายในแท็กจะมีวงจรกำเนิดไฟฟ้าเหนี่ยวนำขนาดเล็กเป็นแหล่งจ่ายไฟในตัว ทำให้การอ่านข้อมูลนั้นทำได้ไม่ไกลมากนัก ระยะอ่านสูงสุดประมาณ 1 เมตร โดยจะขึ้นอยู่กับความแรงของเครื่องส่งและคลื่นความถี่วิทยุที่ใช้ โดยปกติแล้วแท็กชนิดนี้มักมีหน่วยความจำขนาดเล็ก โดยทั่วไปประมาณ 16 ถึง 1,024 ไบต์เท่านั้น และด้วยความที่แท็กชนิดนี้ไม่มีแบตเตอรี่ภายในจึงทำให้มีน้ำหนักเบากว่าแท็กชนิดแอ็กทีฟ ราคาต่ำกว่าและมีอายุการใช้งานไม่จำกัด แต่ข้อเสียก็คือตัวอ่านข้อมูลจะต้องมีความไวสูงและมักจะมีปัญหาเมื่อนำไปใช้งานในสิ่งแวดล้อมที่มีสัญญาณแม่เหล็กไฟฟ้ารบกวนสูงอีกด้วย แต่ด้วยข้อได้เปรียบในเรื่องราคาและอายุการใช้งาน ทำให้แท็กชนิดพาสซีฟนี้เป็นที่นิยมมากกว่าแท็กชนิดแอ็กทีฟ แผงวงจรไอซีของแท็กชนิดพาสซีฟที่มีการผลิตออกมาจะมีทั้งขนาดและรูปร่างเป็นได้ตั้งแต่แท่งหรือแผ่นขนาดเล็กจนแทบไม่สามารถมองเห็นได้จนไปถึงขนาดใหญ่สะดุดตา ซึ่งต่างก็มีความเหมาะสมตามลักษณะการใช้งานที่แตกต่างกัน โดยทั่วไปโครงสร้างภายในส่วนที่เป็นไอซีของแท็กนั้นจะประกอบด้วยส่วนหลักๆ 3 ส่วน ได้แก่ ส่วนควบคุมการทำงานของภาครับส่งสัญญาณวิทยุ (Analog Front-End) ส่วนควบคุมภาคลอจิก (Digital Control Unit) ส่วนของหน่วยความจำ (Memory) ซึ่งอาจจะเป็นแบบ ROM หรือ EEPROM แสดงดังรูปที่ 2.5



รูปที่ 2.5 บล็อกไดอะแกรมของ Passive Tag

นอกจากการแบ่งแท็กตามที่กล่าวมาแล้วข้างต้น แท็กยังถูกแบ่งประเภทจากรูปแบบการใช้งานได้เป็น 3 แบบคือ แบบที่สามารถถูกอ่านและเขียนข้อมูลได้อย่างอิสระ (Read-Write) แบบเขียนได้เพียงครั้งเดียวเท่านั้นแต่อ่านได้อย่างอิสระ (WORM หรือ Write-Once Read-Many) และแบบอ่านได้เพียงครั้งเดียว (Read Only)

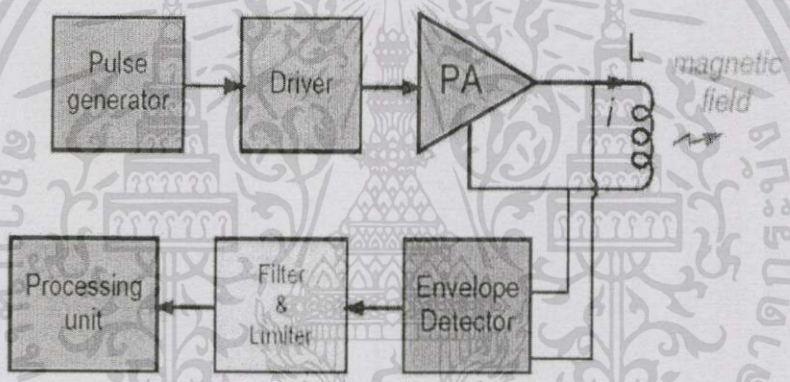
2.3.1.2 เครื่องสำหรับอ่านหรือเขียนข้อมูลภายในแท็กด้วยคลื่นความถี่วิทยุ (Interrogator / Reader)

หน้าที่ของเครื่องอ่าน คือ การเชื่อมต่อหรือเขียนข้อมูลลงในแท็กด้วยสัญญาณความถี่วิทยุ นอกจากนี้ตัวอ่านข้อมูลที่ดีต้องมีความสามารถในการป้องกันการอ่านข้อมูลซ้ำ เช่น ในกรณีที่แท็กถูกวางทิ้งอยู่ในบริเวณสนามแม่เหล็กไฟฟ้าที่ตัวอ่านข้อมูลสร้างขึ้นหรืออยู่ในระยะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การรับส่งก็อาจทำให้ตัวอ่านข้อมูลทำการรับหรืออ่านข้อมูลจากแท็กซ้ำอยู่เรื่อยๆ ไม่สิ้นสุด ดังนั้นตัวอ่านข้อมูลที่ดีต้องมีระบบป้องกันเหตุการณ์เช่นนี้ เรียกว่าระบบ "Hands Down Polling" โดยตัวอ่านข้อมูลจะสั่งให้แท็กหยุดการส่งข้อมูล ในกรณีเกิดเหตุการณ์ดังกล่าวหรืออาจมีบางกรณีที่มีแท็กหลายๆ แท็กอยู่ในบริเวณสนามแม่เหล็กไฟฟ้าพร้อมกันหรือที่เรียกว่า "Batch Reading" ตัวอ่านข้อมูลควรมีความสามารถที่จะจัดลำดับการอ่านแท็กทีละตัวได้ ซึ่งการที่จะชี้เฉพาะระบุตัวแท็กนั้นเป็นระบบอัตโนมัติ (Automatic Identification) แสดงดังรูปที่ 2.6 โดยทั่วไปเครื่องอ่านจะประกอบด้วยส่วนประกอบหลักๆ ดังนี้

- ภาครับและส่งสัญญาณวิทยุ
- ภาคสร้างสัญญาณพาหะ
- ขดลวดที่ทำหน้าที่เป็นสายอากาศ
- วงจรจูนสัญญาณ
- หน่วยประมวลผลข้อมูลและภาคติดต่อกับคอมพิวเตอร์



รูปที่ 2.6 โครงสร้างภายในเครื่องอ่าน

2.3.2 หลักการทำงานเบื้องต้นของระบบ RFID

ตัวเครื่องอ่านจะปล่อยคลื่นแม่เหล็กไฟฟ้าออกมาอย่างต่อเนื่องตลอดเวลา และคอยตรวจจับว่ามีแท็กเข้ามาอยู่ในบริเวณสนามแม่เหล็กไฟฟ้าหรือไม่ หรืออีกนัยหนึ่งก็คือการคอยตรวจจับว่ามีการมอดูเลตสัญญาณเกิดขึ้นหรือไม่ เมื่อมีแท็กเข้ามาอยู่ในบริเวณสนามแม่เหล็กไฟฟ้า แท็กจะได้รับพลังงานไฟฟ้าที่เกิดจากการเหนี่ยวนำของคลื่นไฟฟ้า เพื่อให้แท็กเริ่มทำงานและจะส่งข้อมูลในหน่วยความจำ ที่ผ่านการมอดูเลตกับคลื่นพาหะแล้วส่งออกทางสายอากาศที่อยู่ภายในแท็ก คลื่นพาหะที่ถูกส่งออกมาจากแท็กจะเกิดการเปลี่ยนแปลงทางด้านแอมปลิจูด, ความถี่ หรือเฟส ขึ้นอยู่กับวิธีการมอดูเลต โดยตัวอ่านข้อมูลจะคอยตรวจจับความเปลี่ยนแปลงของคลื่นพาหะแล้วแปลงออกมาเป็นข้อมูลแล้วทำการถอดรหัสและส่งไปยังคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม เพื่อนำข้อมูลไปใช้งาน

2.3.3 ความถี่ของคลื่นพาหะ

ในปัจจุบันคลื่นพาหะที่ใช้งานกันในระบบ RFID จะอยู่ในย่านความที่ ISM (Industrial-Scientific-Medical) ซึ่งเป็นย่านความถี่ที่กำหนดในการใช้งานในเชิงอุตสาหกรรม วิทยาศาสตร์และ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การแพทย์ สามารถใช้งานได้โดยไม่ตรงกับย่านความถี่ที่ใช้งานในการสื่อสารโดยทั่วไป แสดงดังรูปที่ 2.7 โดยคลื่นพาหะที่ใช้กันในระบบ RFID อาจแบ่งออกได้เป็น 3 ย่านหลักๆ ได้แก่

- ย่านความถี่ต่ำ (Low Frequency: LH) ต่ำกว่า 150 KHz
- ย่านความถี่สูง (High Frequency: HF) 13.56 MHz
- ย่านความถี่สูงยิ่ง (Ultra High Frequency: UHF) 433/868/915 MHz

รูปที่ 2.7 ความถี่ย่านที่ระบบ RFID ถูกใช้งาน

ในแง่การใช้งาน 2 ย่านความถี่แรกจะเหมาะสำหรับใช้กับงานที่มีระยะการสื่อสารข้อมูลในระยะใกล้ (LH ระยะอ่านประมาณ 10-20 ซม. และ HF ระยะอ่านประมาณ 1 เมตร) เช่น การตรวจสอบการผ่านเข้าออกพื้นที่การตรวจหาและเก็บประวัติในสัตว์

ส่วนย่านความถี่สูงยิ่งจะถูกใช้กับงานที่มีระยะการสื่อสารข้อมูลในระยะไกล (UHF ระยะอ่านประมาณ 1-10 เมตร) เช่น ระบบเก็บค่าบริการทางด่วน เป็นต้น ซึ่งในปัจจุบันระบบ RFID กำลังถูกวิจัยและพัฒนาในย่านความถี่ไมโครเวฟที่ความถี่ 2.4 GHz และความถี่ 5.8 GHz เพื่อใช้งานที่ต้องการอ่านในระยะไกลกว่า 10 เมตร ในแง่ของราคาและความเร็วในการสื่อสารข้อมูลเมื่อเทียบกันแล้ว RFID ซึ่งใช้คลื่นพาหะย่านความถี่สูง เป็นระบบที่มีความเร็วในการส่งข้อมูลสูงสุดและมีราคาแพงที่สุดด้วยเช่นกัน ส่วน RFID ที่ใช้คลื่นพาหะในอีกสองย่านความถี่จะมีระดับราคาและความเร็วลดหลั่นกันไป

2.3.4 ความถี่ของคลื่นพาหะที่ใช้ในการสื่อสารข้อมูล

ในปัจจุบันได้มีการรวมกลุ่มระหว่างแต่ละประเทศ เพื่อทำการกำหนดมาตรฐานความถี่คลื่นพาหะของระบบ RFID โดยมี 3 กลุ่มใหญ่ๆ คือ กลุ่มประเทศในยุโรปและแอฟริกา (Region 1), กลุ่มประเทศอเมริกาเหนือและอเมริกาใต้ (Region 2) และสุดท้ายคือกลุ่มประเทศตะวันออกไกลและออสเตรเลีย (Region 3) ซึ่งแต่ละกลุ่มประเทศจะกำหนดแนวทางในการเลือกใช้ความถี่ต่างๆ ให้แก่บรรดาประเทศสมาชิก อย่างไรก็ตามความถี่ของคลื่นพาหะที่นิยมใช้งานในย่านความถี่ต่ำ คือ 125 kHz ย่านความถี่ปานกลาง คือ 13.56 MHz และย่านความถี่สูงก็คือ 2.45 GHz ดังที่แสดงไว้ในตารางที่ 2.1 นอกจากนี้รัฐบาลของแต่ละประเทศโดยทั่วไปจะมีการออกกฎหมายเกี่ยวกับระเบียบการใช้งานย่านความถี่ต่างๆ รวมถึงกำลังส่งของระบบ RFID ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 ย่านความถี่ต่างๆของระบบ RFID และการใช้งาน

ย่านความถี่	คุณลักษณะ	การใช้งาน
ย่านความถี่ต่ำ 100-500 kHz ความถี่มาตรฐานที่ใช้งานทั่วไป คือ 125kHz	- ระยะการรับข้อมูลใกล้ - ต้นทุนไม่สูง - ความเร็วในการอ่านข้อมูลต่ำ - ความถี่ในย่านนี้เป็นที่แพร่หลายทั่วโลก	- Access Control - ปศุสัตว์ - ระบบคงคลัง - รถยนต์
ย่านความถี่กลาง 10 -15 MHz ความถี่มาตรฐานที่ใช้งานทั่วไป คือ 13.56 MHz	- ระยะการรับส่งข้อมูลปานกลาง - ราคามีแนวโน้มถูกลงในอนาคต - ความเร็วในการอ่านข้อมูลปานกลาง - ความถี่ในย่านนี้เป็นที่แพร่หลายทั่วโลก	- Access Control - สมาร์ตการ์ด
ย่านความถี่สูง 850 -950 MHz 2.4 -5.8 GHz ความถี่มาตรฐานที่ใช้งานทั่วไป คือ 2.45 GHz	- ระยะการรับส่งข้อมูลไกล (10 เมตร) - ความเร็วในการอ่านข้อมูลสูง - ราคาแพง	- รถไฟ - ระบบเก็บค่าผ่านทาง

2.3.5 ระยะการรับส่งข้อมูลและกำลังส่ง

ระยะการรับส่งข้อมูลในระบบ RFID ขึ้นอยู่กับปัจจัยสำคัญต่างๆ คือ กำลังส่งของตัวอ่านข้อมูล (Reader/Interrogator Power) กำลังส่งของ Tag (Tag Power) และสภาพแวดล้อม ส่วนการออกแบบสายอากาศของตัวอ่านข้อมูล จะเป็นตัวกำหนดลักษณะรูปร่างของคลื่นแม่เหล็กไฟฟ้าที่แผ่กระจายออกมาจากสายอากาศ ดังนั้นระยะการรับส่งข้อมูลบางทีอาจขึ้นอยู่กับมุมของการรับส่งระหว่าง Tag และตัวอ่านข้อมูลด้วยเช่นกัน ทั้งนี้ขึ้นอยู่กับรูปร่างของคลื่นแม่เหล็กไฟฟ้าเป็นสำคัญ ความเข้มของคลื่นแม่เหล็กไฟฟ้าโดยทั่วไปจะลดลงตามระยะทาง โดยแปรผกผันกับระยะทางยกกำลังสอง แต่ในบางสภาพแวดล้อมซึ่งอาจมีการสะท้อนกลับของคลื่นแม่เหล็กไฟฟ้า จากสิ่งต่างๆ รอบตัว เช่น โลหะก็อาจทำให้ความเข้มของคลื่นแม่เหล็กไฟฟ้าลดลงอย่างรวดเร็วโดยอาจแปรผกผันกับระยะทางยกกำลังสี่ ปรากฏการณ์เช่นนี้เราเรียกว่า "Multi-path Attenuation" ซึ่งจะส่งผลให้ระยะการรับส่งข้อมูลสั้นลงหรือแม้กระทั่งความชื้นในอากาศก็อาจมีผลในกรณีที่ความถี่สูงๆ

ดังนั้นการนำระบบ RFID ไปใช้งานควรมีการคำนึงถึงสภาพแวดล้อมเพราะจะมีผลกระทบกับระยะการรับส่งข้อมูล และพยายามติดตั้งระบบให้ห่างไกลจากโลหะซึ่งอาจทำให้เกิดการสะท้อนของคลื่นแม่เหล็กไฟฟ้าได้ กำลังส่งของ Tag ที่จะส่งกลับมายังตัวอ่านข้อมูลนั้น โดยทั่วไปจะมีกำลังที่ต่ำมากเมื่อเทียบกับกำลังส่งของตัวอ่านข้อมูล ดังนั้นความไวในการตรวจจับสัญญาณของตัวอ่านข้อมูลก็เป็นอีกจุดหนึ่งที่ต้องพิจารณา ถึงแม้ในทางเทคนิคเราจะสามารถทำให้ตัวอ่านข้อมูลมีกำลังส่งมากแค่ไหนก็ได้ แต่โดยทั่วไปก็จะถูกจำกัดโดยกฎหมายของแต่ละประเทศเช่นเดียวกับความถี่ ดังนั้นในระบบ RFID โดยทั่วไปจะมีกำลังส่งเพียงระหว่าง 100 -500 mW

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.6 อัตราการรับส่งข้อมูลและ Bandwidth

อัตราการรับส่งข้อมูล (Data Transfer Rate) จะขึ้นอยู่กับความถี่ของคลื่นพาหะ โดยปกติถ้าความถี่ของคลื่นพาหะยิ่งสูงอัตราการรับส่งข้อมูลก็จะยิ่งสูงตามไปด้วย ส่วนการเลือก Bandwidth หรือย่านความถี่นั้นก็จะมีผลต่ออัตราการรับส่งข้อมูลเช่นกัน โดยมีหลักว่า Bandwidth ควรจะมีค่ามากกว่าอัตราการรับส่งข้อมูลที่ต้องการอย่างน้อยสองเท่า เช่น ถ้าใช้ Bandwidth ในช่วง 2.4-2.5 GHz ก็จะสามารถรองรับอัตราการรับส่งข้อมูลได้ถึงประมาณ 2 megabits ต่อวินาทีเป็นต้น แต่การใช้ Bandwidth ที่กว้างเกินไปก็อาจทำให้เกิดปัญหาเกี่ยวกับสัญญาณรบกวนมากหรือทำให้ S/N Ratio ต่ำลงนั่นเอง ดังนั้นการเลือกใช้ Bandwidth ให้ถูกต้องก็เป็นส่วนสำคัญในการพิจารณา

2.3.7 ข้อเปรียบเทียบ RFID และบาร์โค้ด (Barcode)

รูปที่ 2.8 RFID และบาร์โค้ด

1. RFID สามารถทำได้ทั้งอ่านและเขียน โดยจะทำการประมวลผลแล้วสามารถบันทึกค่าข้อมูลบางอย่างกลับลงไปในแท็กที่ตัวสินค้าได้โดยตรง และสามารถเขียนทับข้อมูลได้ จึงทำให้สามารถนำกลับมาใช้ใหม่ได้ ซึ่งจะลดต้นทุนของการผลิตป้ายสินค้า ซึ่งคิดประมาณ 5% ของรายรับของบริษัทในขณะที่ระบบบาร์โค้ดสามารถทำการอ่านแถบรหัสได้อย่างเดียว

2. RFID มีความปลอดภัยสูง เนื่องจากข้อมูลเป็นข้อมูลดิจิตอลในรูปคลื่นแม่เหล็กไฟฟ้า ซึ่งความปลอดภัยของข้อมูลด้วยการเข้ารหัสได้

3. RFID สามารถอ่านค่าได้โดยอัตโนมัติ เมื่อแท็กอยู่ในรัศมีของการอ่านและสามารถอ่านข้อมูลได้พร้อมๆ กันหลายๆ แท็กอย่างรวดเร็ว แต่การใช้งานระบบบาร์โค้ดผู้ใช้งานจะต้องนำเครื่องสแกนไปอ่านที่แถบรหัสโดยตรงและสามารถอ่านได้ทีละแถบเท่านั้น ความเร็วในการอ่านข้อมูลจากแถบ RFID เร็วกว่าการอ่านข้อมูลจากแถบบาร์โค้ดหลายสิบเท่า

4. RFID สามารถขจัดปัญหาที่เกิดขึ้นจากการอ่านข้อมูลซ้ำ ที่อาจเกิดขึ้นจากระบบบาร์โค้ดและค่าเฉลี่ยของความถูกต้องของการอ่านข้อมูล ด้วยเทคโนโลยี RFID นั้นอยู่ที่ประมาณ 99.5 เปอร์เซ็นต์ ขณะที่ความถูกต้องของการอ่านข้อมูลด้วยระบบบาร์โค้ดอยู่ที่ 80 เปอร์เซ็นต์

5. RFID สามารถอ่านได้แม้ไม่เห็นตัวป้ายที่ติดอยู่และป้ายไม่ต้องอยู่ในระนาบเดียวกับเครื่องอ่าน ทำให้ไม่ต้องเคลื่อนย้ายสิ่งของ จึงสามารถลดความเสียหายของแท็ก แถมยังช่วยลดเวลาและข้อผิดพลาดในการทำงาน

6. RFID สามารถอ่านได้จากระยะไกล แม้อยู่ในพื้นที่ๆ ไม่สะดวกหรือในพื้นที่อันตราย ในขณะที่ระบบบาร์โค้ดต้องอยู่ในระยะใกล้และตำแหน่งที่สแกนถึง

7. RFID ทนต่อความเปียกชื้น แรงสั่นสะเทือน การกระทบกระแทก สามารถอ่านได้แม้ในสภาพทัศนวิสัยไม่ดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 รูปแบบการสื่อสารแบบ USART

การแปลงข้อมูลแบบขนานเป็นข้อมูลแบบอนุกรม เริ่มแรกข้อมูลแบบขนานจะถูกเก็บไว้ในชิฟรีจิสเตอร์ (Shift register) จากนั้นจะใช้สัญญาณนาฬิกาในการเลื่อนค่าในรีจิสเตอร์ออกมาทีละบิต โดยบิต LSB จะถูกเลื่อนออกมาก่อน การแปลงข้อมูลแบบอนุกรมไปเป็นข้อมูลแบบขนาน ข้อมูลแบบอนุกรมจะถูกเลื่อนเข้าไปเก็บในชิฟรีจิสเตอร์ (Shift register) หลังจากที่ได้มีการเลื่อนข้อมูลทุกบิตเข้าไปใน shift register แล้วข้อมูลใน shift register ก็จะถูกนำออกมาแบบขนานเพื่อนำไปให้กับไมโครโปรเซสเซอร์

USART (Universal Synchronous Asynchronous Receiver Transmitter) เป็นโมดูลสื่อสารข้อมูลอนุกรมเอนกประสงค์ที่สามารถทำงานเป็นตัวรับ หรือ ตัวส่งข้อมูลในแบบอะซิงโคนัส หรือซิงโคนัสก็ได้ ทั้งยังสามารถกำหนดรูปแบบของข้อมูลได้ เลือกอัตราเร็วในการถ่ายทอดข้อมูลได้ เลือกอัตราเร็วในการถ่ายทอดข้อมูลได้เมื่อกำหนดให้ทำงานแบบฟูลดูเพล็กซ์อะซิงโครนัส (full duplex asynchronous) จะสามารถรับ-ส่งข้อมูล 2 ทิศทางได้ในเวลาเดียวกัน ทั้งนี้เพื่อใช้ติดต่อกับพอร์ตอนุกรมของคอมพิวเตอร์หรืออุปกรณ์เทอร์มินอลแบบอื่น ถ้ากำหนดให้ทำงานเป็นโมดูลสื่อสารข้อมูลอนุกรมแบบซิงโครนัสจะสามารถถ่ายทอดข้อมูลในลักษณะฮาล์ฟดูเพล็กซ์ กล่าวคือ สามารถรับ-ส่งข้อมูล 2 ทิศทางได้ แต่ต้องสลับการรับและส่ง การใช้งานในลักษณะนี้ คือ ติดต่อกับอุปกรณ์ต่อพ่วง อาทิ หน่วยความจำอีอีพรอมอนุกรม ไอซีแปลงสัญญาณอนาลอกเป็นดิจิตอล หรือแปลงสัญญาณดิจิตอลเป็นอนาลอก เป็นต้น

โมดูล USART ใน PIC 18F458 สามารถกำหนดการทำงานได้ 3 โหมดคือ

1. โหมดอะซิงโคนัส สามารถรับส่งข้อมูลในลักษณะฟลูดูเพล็กซ์
2. โหมดซิงโครนัส-มาสเตอร์ สามารถรับส่งข้อมูลในลักษณะฮาล์ฟดูเพล็กซ์
3. โหมดซิงโคนัส-สลาฟ สามารถรับส่งข้อมูลในลักษณะฮาล์ฟดูเพล็กซ์ ซึ่งในที่นี่จะใช้โหมดอะซิงโคนัสซึ่งสามารถรับส่งข้อมูลในลักษณะฟลูดูเพล็กซ์

2.4.1 รีจิสเตอร์ควบคุมการทำงานของโมดูล USART

ในโมดูล USART มีรีจีสเตอร์ควบคุมการทำงาน 3 ตัว คือ TXSTA (Transmit status and control register) ใช้ควบคุมและแสดงสถานการณ์ส่งข้อมูล RCSTA (Receive status and control register) ใช้ควบคุมและแสดงสถานการณ์การรับข้อมูล และ SPBRG (Serial port baud rate generator) ใช้กำหนดอัตราเร็วในการถ่ายทอดข้อมูล รีจีสเตอร์ที่ใช้เก็บข้อมูลมี 2 ตัว TXREG (Transmit data register) ใช้เก็บข้อมูลที่ต้องการส่ง และ RCREG (Receive data register) ใช้เก็บข้อมูลที่รับเข้ามา

2.4.1.1 รีจิสเตอร์ TXSTA

รีจิสเตอร์ TXSTA เป็นรีจิสเตอรที่ใช้ควบคุมและเก็บสถานะการทำงานในภาคการส่งข้อมูลรีจิสเตอร์โดยแสดงบิตต่างๆ ดังตารางที่ 2.2

ตารางที่ 2.2 แสดงบิตต่างๆในริจิสเตอร์ TXSTA

CSRC	TX9	TXEN	SYNC	-	BRGH	TRMT	TX9D

bit 7 bit 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- บิต 7 CSRS (Clock source bit) : บิตเลือกแหล่งกำเนิดสัญญาณนาฬิกา
  ในโหมดอะซิงโครนัส : ไม่มีการใช้งานบิตนี้
  ในโหมดซิงโครนัส:
  "0": ทำงานเป็นอุปกรณ์สลาฟ เลือกแหล่งกำเนิดสัญญาณนาฬิกาภายนอก
  "1": ทำงานเป็นอุปกรณ์มาสเตอร์ ใช้แหล่งกำเนิดสัญญาณนาฬิกาจากส่วนกำเนิดอัตราบอดหรืออัตราเร็ว ในการถ่ายทอดข้อมูลภายในไมโครคอนโทรลเลอร์
- บิต 6 TX9 (9-bit transmit enable bit): บิตเลือกการส่งข้อมูลแบบ 9 บิต
  "0": เลือกการส่งข้อมูลแบบ 8 บิต
  "1": เลือกการส่งข้อมูลแบบ 9 บิต
- บิต 5 TXEN (Transmit enable bit): บิตเลือกการทำงานของตัวส่ง
  "0": ดิสเอเบิลตัวส่งข้อมูล
  "1": เอ็นเอเบิลตัวส่งข้อมูล
- บิต 4 SYNC (USART mode select bit): บิตเลือกโหมดการทำงานของโมดูลUSART
  "0": เลือกโหมดอะซิงโครนัส
  "1": เลือกโหมดซิงโครนัส
- บิต 3 ไม่มีการใช้งาน อ่านค่าเป็น "0"
- บิต 2 BRGH (High baud rate select bit): บิตเลือกโหมดของอัตราเร็วในการถ่ายทอดข้อมูล
  ในโหมดอะซิงโครนัส:
  "0": เลือกอัตราเร็วในการถ่ายทอดข้อมูลต่ำ
  "1": เลือกอัตราเร็วในการถ่ายทอดข้อมูลสูง
  ในโหมดซิงโครนัส: ไม่มีการใช้งานบิตนี้
- บิต 1 TRMT (Transmit shift register status bit: บิต 1): บิตแสดงสถานะรีจิสเตอร์บัฟเฟอร์ของการส่ง
  "0": รีจิสเตอร์บัฟเฟอร์เต็ม หรือไม่ว่าง
  "1": รีจิสเตอร์บัฟเฟอร์ว่าง
- บิต 0 TX9D (9^{th} bit of transmit data) : บิตเก็บข้อมูลบิตที่ 9 ของการส่งข้อมูลแบบ 9 บิต บิตนี้ใช้บรรจุข้อมูลบิตที่ 9 ในกรณีที่เลือกการส่งข้อมูลแบบ 9 บิต โดยการเซตบิต TX9 และยังสามารถใช้เป็นบิตพาริตี้ของการสื่อสารข้อมูลอนุกรมได้

2.4.1.2 รีจิสเตอร์ RCSTA

รีจิสเตอร์ RCSTA เป็นรีจิสเตอร์แสดงสถานะและควบคุมการรับส่งข้อมูล โดยแสดงบิตต่างๆ ดังตารางที่ 2.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.3 แสดงบิตต่างๆในริจิสเตอร์ RCSTA

CSRC	TX9	TXEN	SYNC	-	BRGH	TRMT	TX9D

bit 7 bit 0

- บิต 7 SPEN (Serial port enable bit): บิตเลือกการทำงานของโมดูล USART
  "0": ดิสเอเบิล
  "1": เอ็นเอเบิลเพื่อใช้งานโมดูล USART ทำให้ขา RB7/RxD และ RB6/TxD ใช้งานกับโมดูล USART เพื่อสื่อสารอนุกรม เมื่อบิต 7 และ 6 ของรีจิสเตอร์ TRISC ถูกเซตเป็น "1"
- บิต 6 RX9 (9-bit receive enable bit): บิตเลือกการรับข้อมูลแบบ 9 บิต
  "0": เลือกการรับข้อมูลแบบ 8 บิต
  "1": เลือกการรับข้อมูลแบบ 9 บิต
- บิต 5 SREN (Single receive enable bit): บิตเลือกการรับข้อมูลเพียงครั้งเดียว จะเคลียร์หลังจากรับข้อมูลสมบูรณ์
  ในโหมดอะซิงโครนัส และซิงโครนัส– สลาฟ: ไม่ใช้งานบิตนี้
  ในโหมดซิงโครนัส - มาสเตอร์
  "0": ดิสเอเบิลการรับข้อมูลครั้งเดียว
  "1": เอ็นเอเบิลการรับข้อมูลครั้งเดียว
- บิต 4 CREN (Continuous receive enable bit): บิตเลือกการรับข้อมูลต่อเนื่อง
  "0": ดิสเอเบิลการรับข้อมูลต่อเนื่อง
  "1": เอ็นเอเบิลการรับข้อมูลต่อเนื่อง
- บิต 3 ADDEN (Address detect enable bit): บิตเลือกการตรวจจับแอดเดรส ใช้ในกรณีเลือกการรับข้อมูลแบบ 9 บิตเท่านั้น
  "0": ดิสเอเบิล การตรวจจับแอดเดรส ข้อมูลจะถูกรับทั้งหมด และบิตที่ 9 ใช้เป็นบิตพาริตี้ได้
  "1": เอ็นเอเบิล การตรวจจับแอดเดรสส่งผลให้เกิดการเอ็นเอเบิลอินเตอร์รัปต์ และมีการถ่ายทอดข้อมูลไปยังบัฟเฟอร์เมื่อบิต RSR ถูกเซต
- บิต 2 Free (Framing error bit): บิตแจ้งความผิดพลาดทางเฟรมข้อมูล เซตเป็น "1" เมื่อมีความผิดพลาดทางเฟรมข้อมูลเกิดขึ้น
- บิต 1 OERR (Overrun error bit): บิตแจ้งความผิดพลาด เนื่องจากการชนกันของข้อมูล
  "0": ไม่มีความผิดพลาดเกิดขึ้น
  "1": เกิดความผิดพลาดขึ้น สามารถเคลียร์บิตนี้ได้ด้วยการเคลียร์บิต CREN

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- บิต 0 RX9D (9^{th} bit of received data): บิตเก็บข้อมูลบิตที่ 9 ของการรับข้อมูลแบบ 9 บิตบิตนี้ใช้บรรจุข้อมูลบิตที่ 9 ในกรณีที่เลือกการรับข้อมูลแบบ 9 บิต โดยการเซตบิต RX9 และ ยังสามารถใช้เป็นบิตพาริตี้ของการสื่อสารข้อมูลอนุกรมได้

2.4.2 โหมดการทำงานของโมดูลUSART

2.4.2.1 การทำงานในโหมดอะซิงโคนัส

ในโหมดนี้โมดูล USART จะกำหนดรูปแบบของข้อมูลที่ใช้ในการสื่อสารตามมาตรฐาน NRZ (non-return–to-zero) ซึ่งประกอบด้วยบิตเริ่มต้น 1 บิต, บิตข้อมูล 8-9 บิต และบิตหยุดหรือบิตปิดท้าย 1 บิต โดยปกติจะใช้ขนาดของข้อมูล 8 บิต การรับส่งข้อมูลในโหมดนี้จะเริ่มต้นด้วยบิตนัยสำคัญต่ำสุดหรือ LSB ก่อน ส่วนการตรวจสอบพาริตี้นั้นต้องใช้กระบวนการทางซอฟแวร์โดยบรรจุข้อมูลของบิตพาริตี้ลงในบิตสำหรับบรรจุข้อมูลบิตที่ 9 ของทั้งรีจิสเตอร์ TXSTA และ RCSTA แต่โดยปกติจะเลือกไม่ตรวจสอบหรือกำหนดบิตพาริตี้เป็น none

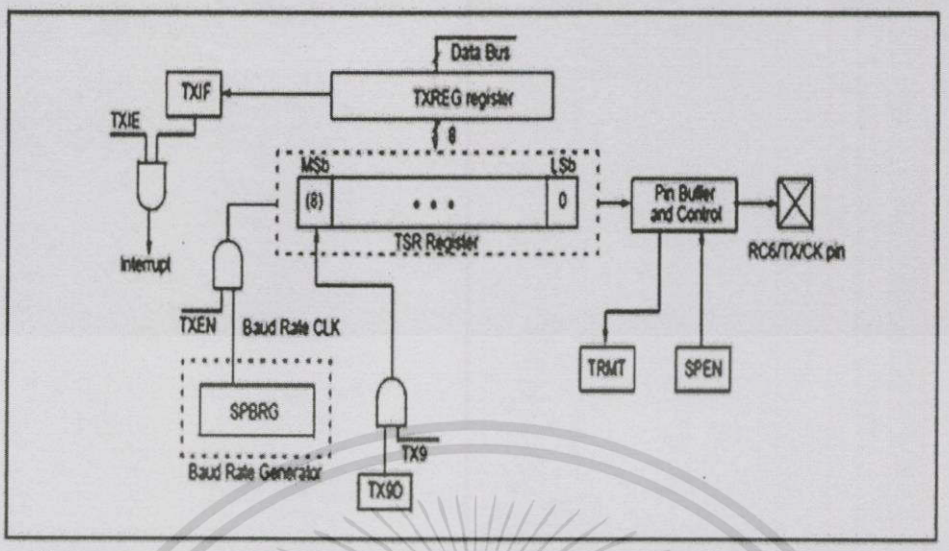
โมดูล USART มีส่วนประกอบ 4 ส่วนที่ใช้ในโหมดอะซิงโคนัส ได้แก่

1. ส่วนกำเนิดอัตราเร็วในการถ่ายทอดข้อมูลหรือบอดเรตเจเนอเรเตอร์
2. วงจรสุ่มสัญญาณ
3. ตัวส่งข้อมูลแบบอะซิงโคนัส
4. ตัวรับข้อมูลแบบอะซิงโคนัส

2.4.2.2 การส่งข้อมูลในโหมดอะซิงโครนัส

มีไดอะแกรมการทำงานแสดงในรูปที่ 2.9 หัวใจสำคัญอยู่ที่ชิฟต์รีจิสเตอร์ตัวส่ง (Transmit shift register: TSR) โดยในชิฟต์รีจิสเตอร์ตัวนี้บรรจุข้อมูลที่ได้มาจากรีจิสเตอร์เก็บข้อมูลสำหรับการส่งหรือ TXREG ข้อมูลต้องการส่งจะถูกเขียนมายัง TXREG จากนั้นจะถ่ายทอดต่อไปยังชิฟต์รีจิสเตอร์ TSR แต่การถ่ายทอดจะเกิดขึ้นก็ต่อเมื่อมีการส่งบิตปิดท้ายของข้อมูลในชุดก่อนหน้าออกไปแล้ว หลังจากที่มีการส่งบิตปิดท้ายออกไป TSR จะได้รับข้อมูลใหม่จากรีจิสเตอร์ TXREG ซึ่งเกิดขึ้นภายในไซเกิลการทำงานเดียวที่รีจิสเตอร์ TXREG จะว่างลง และบิต TXIF (บิต 4 ของรีจิสเตอร์ PIR1) จะเซต หากมีการเอ็นเอเบิลอินเตอร์รัปไว้ ก็จะเกิดการอินเตอร์รัปขึ้น บิต TXIF จะเคลียร์ได้ ก็ต่อเมื่อมีการถ่ายทอดข้อมูลใหม่ลงในรีจิสเตอร์ TXREG ส่วนสถานะของชิฟต์รีจิสเตอร์ TSR จะแสดงผ่านบิต TRMT (บิต 1 ของรีจิสเตอร์ TXSTA) เมื่อ TSR ว่างจะเซตบิตนี้ ดังนั้นผู้ใช้งานสามารถเขียนโปรแกรมตรวจจับการเปลี่ยนแปลงที่บิตนี้เพื่อเริ่มต้นการส่งข้อมูลก็ได้ การเริ่มต้นส่งข้อมูลออกจากโมดูล USART เกิดขึ้นเมื่อเซตบิตTXEN (บิต 5 ของรีจิสเตอร์ TXSTA) แต่การเริ่มต้นในทางปฏิบัติจะเกิดขึ้นหลังจากที่รีจิสเตอร์ TXREG ได้รับการเขียนข้อมูลและบอดเรตเจเนอเรเตอร์สร้างสัญญาณนาฬิกาของการถ่ายทอดข้อมูลขึ้น ตามรูปที่ 2.9 ข้อมูลจาก TXREG จะส่งไปยังชิฟรีจิสเตอร์ TSR แล้วส่งออกไปตามลำดับทางขาพอร์ต RC6/TxD/CK โดยที่ขาพอร์ตยังมีวงจรบัฟเฟอร์และสัญญาณควบคุมอีก 1 เส้น คือ บิต SPEN (บิต 7 ของรีจิสเตอร์ RCSTA) ซึ่งต้องเซตบิตนี้ มิฉะนั้นข้อมูลจะไม่สามารถส่งออกไปได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 ไดอะแกรมการส่งข้อมูลในโหมดอะซิงโครนัสของโมดูล USART ใน PIC18F458

การเลื่อนข้อมูลจะเริ่มต้นด้วยบิตเริ่มต้นต่อด้วยข้อมูลบิต 0 จนถึงข้อมูลบิต 7 หรือ 8 ขึ้นอยู่กับการกำหนดขนาดของข้อมูลว่าต้องการส่งแบบ 8 หรือ 9 บิต และส่งบิตปิดท้ายอีก 1 บิตสุดท้าย เป็นอันครบ 1 ชุดของข้อมูล บิต TRMT จะแจ้งว่าชิฟต์รีจิสเตอร์ไม่ว่างหลังจากที่ส่งข้อมูลบิตสุดท้าย ก่อนส่งบิตปิดท้าย

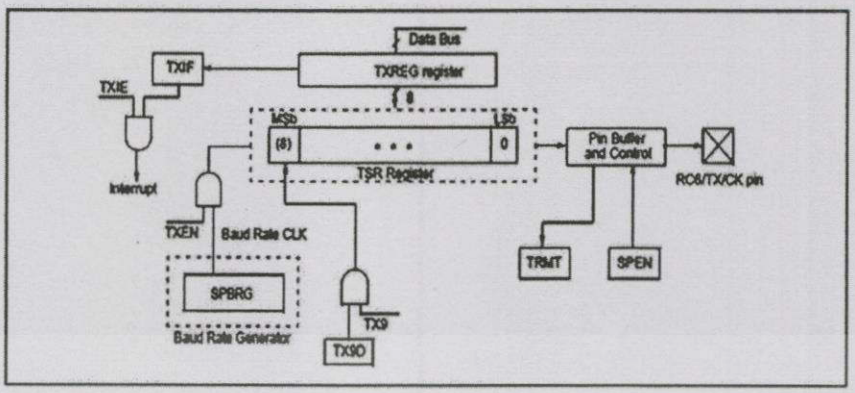
สรุปขั้นตอนการกำหนดให้โมดูล USART ส่งข้อมูลในโหมดอะซิงโครนัสได้ดังนี้

1. กำหนดข้อมูลสร้างบอดเรต
2. เซตบิต SPEN และเคลียร์บิต SYNC เพื่อกำหนดการทำงานเป็นโหมดอะซิงโคนัส
3. ถ้าต้องการให้เกิดอินเตอร์รัปต์ ต้องทำการเซตบิต TXIE
4. เอ็นนาเบิลการส่งข้อมูลโดยเซตบิต TXEN ส่งผลให้แฟลก TXIF ถูกเซตตาม
5. กำหนดข้อมูลลงรีจิสเตอร์ TXREG เพื่อเริ่มต้นส่งข้อมูล

2.4.2.3 การรับข้อมูลในโหมดอะซิงโครนัส

ไดอะแกรมการทำงานแสดงในรูปที่ 2.10 การกำหนดให้ตัวรับข้อมูลแบบอะซิงโครนัสในโมดูล USART ทำงาน ต้องเซตบิต CREN (บิต 4 ของรีจิสเตอร์ RCSTA) ส่วนข้อมูลจะถูกส่งเข้ามาทางขาพอร์ต RC7/RxD/DT ทีละบิต ผ่านวงจรบัฟเฟอร์เข้าสู่ส่วนเรียกคืนข้อมูล (data recovery) ซึ่งภายในเป็นตัวเลื่อนข้อมูลความเร็วสูงทำงานที่ความเร็ว 16 เท่าของบอดเรต ทั้งนี้เพื่อให้โมดูล USART สามารถรับข้อมูลเข้ามาและถ่ายทอดต่อไปยังชิฟต์รีจิสเตอร์ตัวรับหรือ RSR (receive shift register) ได้เร็วขึ้น ส่งผลให้สามารถรับข้อมูลได้อย่างทันทีและครบถ้วน ข้อมูลที่รับเข้ามาจะเริ่มจากบิตเริ่มต้นต่อด้วยข้อมูลบิต 0 จนถึงบิต 7 หรือ 8 ขึ้นอยู่กับการเลือกและสุดท้ายคือบิตปิดท้าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 ไดอะแกรมการรับข้อมูลในโหมดอะซิงโครนัสของโมดูล USART ใน PIC18F458

หัวใจของการรับข้อมูล คือชิฟต์รีจิสเตอร์ RSR หลังจากที่ตรวจจับบิตปิดท้ายของการรับข้อมูลได้ ข้อมูลใน RSR จะส่งไปยังรีจิสเตอร์ข้อมูลตัวรับ RCREG หลังจากที่ถ่ายทอดข้อมูลเสร็จสมบูรณ์บิตแฟลก RCIF (บิต 5 ของรีจิสเตอร์ PIR1) จะถูกเซต หากมีการเอ็นเอเบิลอินเตอร์รัปต์ไว้ก็จะเกิดอินเตอร์รัปต์ขึ้น แฟลก RCREG ถูกอ่านออกไปและว่างลง รีจิสเตอร์ข้อมูลตัวรับ RCREG มีด้วยกัน 2 ชุด จึงมีลักษณะคล้ายกับเป็นที่พักข้อมูล 2 ชุด แบบ FIFO คือ ข้อมูลที่เข้าก่อนจะถูกส่งออกก่อน ทำให้การรับข้อมูลทำได้อย่างรวดเร็ว เพราะสามารถรับข้อมูล 3 ไบต์ได้อย่างต่อเนื่อง โดยในขณะที่กำลังถ่ายทอดข้อมูลทั้ง 2 ไบต์จากรีจิสเตอร์ RCREG ข้อมูลในไบต์ที่สามารถเข้าสู่โมดูล USART ได้ทันที โดยข้อมูลจะถูกเก็บไว้ที่ RSR ก่อน ถ้าสามารถตรวจจับบิตปิดท้ายได้ แต่ในรีจิสเตอร์ RCREG ยังมีข้อมูลอยู่จะเกิดเหตุการณ์ชนกันของข้อมูล เรียกว่า โอเวอร์รัน (overrun error) บิต OERR จะถูกเซตเพื่อแจ้งความผิดพลาดดังกล่าว ส่งผลให้ข้อมูลใน RSR หายไป และการถ่ายทอดข้อมูลจาก RSR ไปยังรีจิสเตอร์ RCREG จะถูกยับยั้งไว้ตราบเท่าที่บิต OERR ยังคงเซตอยู่ บิต OERR สามารถเคลียร์ได้ด้วยกระบวนการทางซอฟต์แวร์ ความผิดพลาดอีกแบบหนึ่งที่อาจเกิดขึ้นได้ คือ ความผิดพลาดทางเฟรมข้อมูล (framing error) จะเกิดขึ้นเมื่อการตรวจจับบิตปิดท้ายพบว่า เป็นลอจิก “0” ซึ่งจะตรงกับสถานะของบิตเริ่มต้นของข้อมูลในชุดหรือเฟรมถัดไป ส่งผลให้ไม่สามารถตีความได้ว่า ได้รับข้อมูลชุดก่อนนี้อย่างสมบูรณ์แล้วหรือไม่ เมื่อการตรวจจับข้อมูลพบว่า เป็นลอจิก“0” การรับข้อมูลของตัวรับก็จะเริ่มต้นต่อไป ส่งผลให้เกิดการอ่านข้อมูลบิตแรกเป็น “0” ซึ่งแท้ที่จริงแล้วมัน คือ บิตเริ่มต้นของข้อมูลชุดต่อไป ทำให้ข้อมูลที่รับเข้ามาเกิดความผิดพลาดได้ เมื่อเกิดผิดพลาดแบบนี้ขึ้น บิต FERR จะถูกเซตเพื่อแจ้งความผิดพลาดดังกล่าว เมื่อมีการอ่านค่าของรีจิสเตอร์ RCREG บิต FERR จะถูกปรับปรุงโดยอัตโนมัติ ถ้าหากการรับข้อมูลครั้งใหม่ถูกต้อง บิต FERR จะกลายเป็น “0” โดยอัตโนมัติ อย่างไรก็ตามหากเกิดความผิดพลาดทางเฟรมข้อมูลขึ้น และเลือกการรับข้อมูลแบบ 9 บิต จำเป็นอย่างยิ่งที่จะต้องทำการอ่านข้อมูลจาก RCSTA ขึ้นมาก่อน เพื่อรักษาข้อมูลบิตที่ 9 เอาไว้ จากนั้นจึงทำการอ่านค่าของรีจิสเตอร์ RCREG เพื่ออ่านข้อมูลใน 8 บิตที่เหลือ พร้อมกับปรับปรุงค่าของบิต FERR ด้วย

สรุปขั้นตอนการกำหนดให้โมดูล USART รับข้อมูลในโหมดอะซิงโครนัสได้ดังนี้

1. ส่วนกำเนิดอัตราเร็วในการถ่ายทอดข้อมูล หรือบอดเรตเจเนอเร-เตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

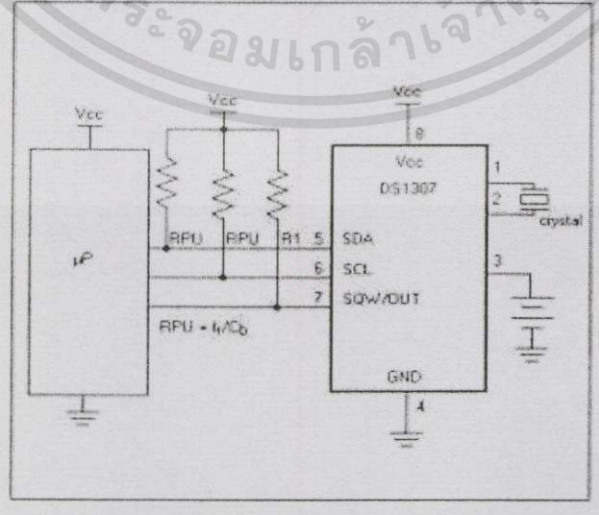
2. เซตบิต SPEN และเคลียร์บิต SYNC เพื่อกำหนดดารทำงาน เป็นโหมดอะซิงโคนัส
3. บิตแฟลก RCIF ถูกเซต เมื่อรับข้อมูลเสร็จสิ้น จะเกิดอินเตอร์รัพต์หากเอ็นเอเบิล
4. อ่านข้อมูล 8 บิต ที่รับมาจากรีจีสเตอร์ RCREG
5. ถ้าเกิดความผิดพลาดขึ้น ให้เคลียร์บิต CREN

2.5 Real Time Clock

ระบบฐานเวลา เป็นสิ่งสำคัญที่สามารถนำไปใช้ในอุปกรณ์อิเล็กทรอนิกส์ได้หลากหลาย ภายในไมโครคอนโทรลเลอร์สามารถทำงานได้ต่อเมื่อมีไฟเลี้ยงเท่านั้น ดังนั้น การใช้งานไทเมอร์ของไมโครคอนโทรลเลอร์สร้างฐานเวลาจริงจึงไม่สามารถรักษาข้อมูลไว้ได้ ไอซี RTC DS1307 แสดงในรูปที่ 2.11

รูปที่ 2.11 ไอซี RTC DS1307

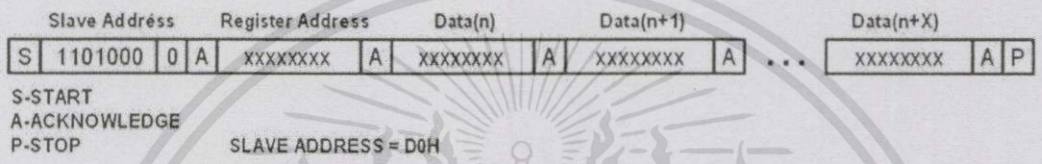
การต่อใช้งานร่วมกับไมโครคอนโทรลเลอร์ด้วยระบบบัส I^2C นั้น สามารถทำได้โดยต่อตัวต้านทาน Pull up ดังแสดงดังรูปที่ 2.12 ในกรณีที่ต้องการต่อร่วมกับอุปกรณ์สลาฟหลายตัว ก็สามารถทำได้โดยต่ออุปกรณ์สลาฟขนานกันไป การติดต่อสื่อสารระหว่างอุปกรณ์มาสเตอร์กับสลาฟแต่ละตัวนั้น จะถูกแยกโดย Address ของอุปกรณ์สลาฟ ซึ่งจะถูกส่งจากอุปกรณ์มาสเตอร์ไปยังอุปกรณ์สลาฟก่อนเริ่มการรับส่งข้อมูล



รูปที่ 2.12 การเชื่อมต่อ DS1307 เข้ากับไมโครคอนโทรลเลอร์ด้วยระบบบัสแบบ I^2C

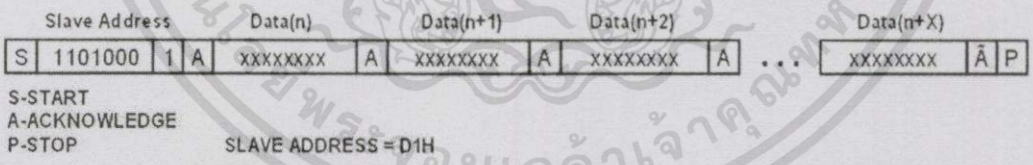
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการรับ-ส่งข้อมูลผ่านบัส I^2C อุปกรณ์มาสเตอร์จะเป็นผู้สร้างสัญญาณ Clock บน SDA และเป็นตัวควบคุมสถานะ Start และ Stop เพื่อควบคุมการรับ-ส่งข้อมูลทั้งหมด การส่งข้อมูลไปยังอุปกรณ์ DS1307 ดังแสดงในรูปที่ 2.13 ไมโครคอนโทรลเลอร์ต้องสร้างสภาวะ Start ก่อน จากนั้นต้องส่ง Address ของ DS1307 ขนาด 7 บิตซึ่งมีค่าเป็น 1101000 และตามด้วยบิตระบุทิศทางของข้อมูล ในกรณีที่เป็นการเขียนข้อมูลลง DS1307 จะต้องเป็น "0" จากนั้นไมโครคอนโทรเลอร์จะต้องส่งตำแหน่ง Address ภายในรีจิสเตอร์ของ DS1307 ที่ต้องการเขียนข้อมูลลง แล้วจึงค่อยเขียนข้อมูลลง โดยในการส่งข้อมูลแต่ละไบต์จะต้องรอบิต Acknowledge จาก DS1307 ทุกไบต์ เมื่อส่งจนครบแล้วถึงจะสร้างสภาวะ Stop เพื่อกลับสู่สถานะว่าง



รูปที่ 2.13 การเขียนข้อมูลอุปกรณ์สลาฟผ่านบัส I^2C

การรับข้อมูลจากอุปกรณ์สลาฟ ดังแสดงในรูปที่ 2.14 เริ่มแรกไมโครคอนโทรลเลอร์ต้องสร้างสภาวะ Start ก่อน จากนั้นต้องส่ง Address ของ DS1307 ขนาด 7 บิต ซึ่งมีค่าเป็น 1101000 และตามด้วยบิตระบุทิศทางของข้อมูล ในกรณีที่เป็นการอ่านข้อมูลลง DS1307 จะต้องเป็น "1" จากนั้นจึงค่อยรับข้อมูลจากอุปกรณ์สลาฟทีละไบต์ โดยตำแหน่งที่อ่านเข้ามาจะขึ้นอยู่กับตำแหน่งรีจิสเตอร์พอยท์เตอร์ ซึ่งเป็นตำแหน่งท้ายสุดที่ได้ทำการเขียนข้อมูลไว้ เมื่ออ่านข้อมูลครบแต่ละไบต์อุปกรณ์มาสเตอร์ต้องส่ง Acknowledge บิตกลับไปให้อุปกรณ์สลาฟด้วย ในกรณีที่เป็นไบต์สุดท้ายอุปกรณ์มาสเตอร์ต้องส่ง "not acknowledge" กลับไป



รูปที่ 2.14 การอ่านข้อมูลจากอุปกรณ์สลาฟผ่านบัส I^2C

ภายใน DS1307 มีรีจิสเตอร์ภายในใช้เก็บข้อมูลเวลาขนาด 7 ไบต์ 00H-06H ดังแสดงในรูปที่ 2.15 ข้อมูลค่าเวลาและวันที่จะถูกเก็บอยู่ในรูปของเลขฐาน 10 สามารถเลือกได้ว่าให้ทำงานแบบ 12 ชั่วโมง หรือ 24 ชั่วโมง โดยกำหนดที่บิตที่ 6 ที่แอดเดรส 02H โดยถ้าเป็น "1" จะเป็นการทำงานในโหมด 12 ชั่วโมง และเมื่อเลือกแบบ 12 ชั่วโมง ที่บิต 5 ในแอดเดรส 02H นั้นจะใช้แสดงค่า AM/PM โดยถ้าบิตนี้เป็น "1" จะเป็น PM ในกรณีที่แสดงแบบ 24 ชั่วโมง บิตนี้จะใช้ในการแสดงค่าของหลักสิบในของหน่วยชั่วโมงด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	BIT7							BIT0	
00H	CH	10 SECONDS			SECONDS				00-59
	0	10 MINUTES			MINUTES				00-59
	0	12/24	10 HR A/P	10 HR	HOURS				01-12 00-23
	0	0	0	0	0	DAY			1-7
	0	0	10 DATE		DATE				
	0	0	0	10 MONTH	MONTH				01-12
	10 YEAR				YEAR				00-99
07H	OUT	0	0	SQWE	0	0	RS1	RS0	

รูปที่ 2.15 รีจิสเตอร์ภายในไอซีฐานเวลา DS1307

ที่แอดเดรส 07H เป็นรีจิสเตอร์ควบคุมการทำงานของ SQW/OUT โดยมีรายละเอียดดังนี้

- OUT (Out control): ใช้ควบคุมเอาต์พุต
- SQWE (Square Wave Enable): ใช้ควบคุมออสซิลเลเตอร์ภายใน DS1307 โดยถ้าบิตนี้เป็น "1" จะเป็นการเปิดออสซิลเลเตอร์
- RS (Rate Select): ใช้ควบคุมความถี่ของ Square Wave เมื่อเปิดการทำงานของออสซิเลเตอร์ โดยสามารถปรับเปลี่ยนความถี่ได้ 4 ความถี่ด้วยกันดังแสดงในตารางที่ 2.4

ตารางที่ 2.4 การควบคุมความถี่ออสซิลเลเตอร์ด้วยการเซตบิต RS1, RS0

RS1	RS0	SQW OUTPUT FREQUENCY
0	0	1 Hz
0	1	4.096 kHz
1	0	8.192 kHz
1	1	32.768 kHz

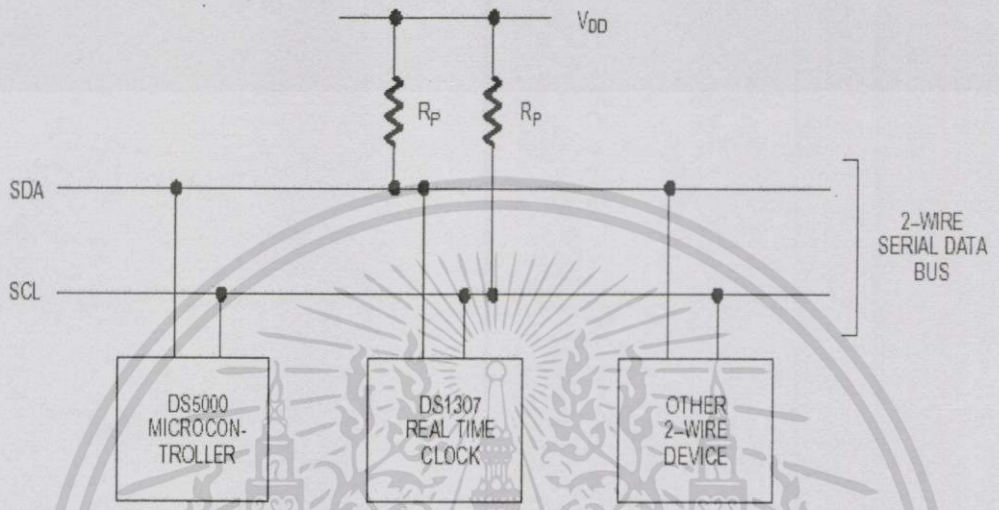
2.6 I^2C (Inter Integrate Circuit Bus)

I^2C= I2C Bus ย่อมาจาก Inter Integrate Circuit Bus (IIC) นิยมเรียกสั้นๆ ว่า I^2C BUS (ไอ-แสควร์-ซี-บัส) เป็นการสื่อสารอนุกรมแบบซิงโครนัส (Synchronous) เพื่อใช้ติดต่อสื่อสารระหว่างไมโครคอนโทรลเลอร์ (MCU) กับอุปกรณ์ภายนอก ซึ่งถูกพัฒนาขึ้นโดยบริษัท Philips Semiconductors โดยใช้สายสัญญาณเพียง 2 เส้นเท่านั้น คือ serial data (SDA) และสาย serial clock (SCL) ซึ่งสามารถเชื่อมต่ออุปกรณ์จำนวนหลายๆ ตัว เข้าด้วยกันได้ ทำให้ MCU ใช้พอร์ตเพียง 2 พอร์ตเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

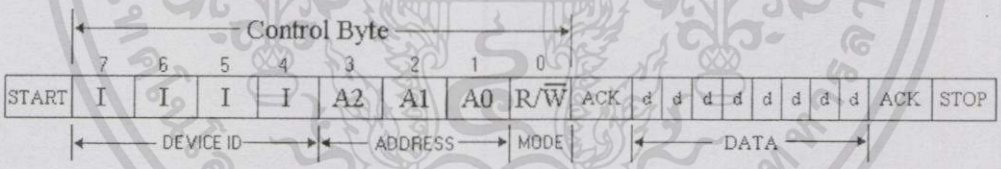
2.6.1 การเชื่อมต่ออุปกรณ์แบบ I^2C BUS

I^2C BUS ใช้สายสัญญาณ 2 เส้น คือ SCL, SDA สำหรับติดกับอุปกรณ์แบบ 2 ทิศทาง โดยที่ขาสัญญาณทั้ง 2 จะต้องต่อกับตัวต้านทานแบบ pull up 2-10K เนื่องจากเอาต์พุตมีลักษณะเป็นแบบ Open Darin หรือเป็นแบบ Open Collector เพื่อให้เอาต์พุตเชื่อมต่อกันได้หลายตัว



รูปที่ 2.16 ลักษณะการเชื่อมต่ออุปกรณ์แบบ I^2C BUS

2.6.2 การเขียน-อ่านข้อมูลกับอุปกรณ์แบบ I^2C BUS



รูปที่ 2.17 รูปแบบการเขียน/อ่านข้อมูลแบบ I^2C BUS

การรับ-ส่งข้อมูลแบบ I^2C BUS ไมโครคอนโทรลเลอร์ จะเริ่มต้นการส่งข้อมูลด้วยการ

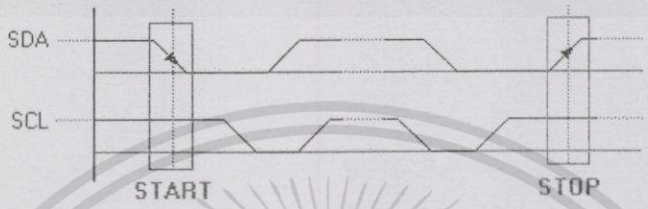
- ส่งสถานะเริ่มต้น (START Conditions) เพื่อแสดงขอการใช้บัส
- แล้วตามด้วยรหัสควบคุม (Control; Byte) ซึ่งประกอบด้วยรหัสประจำตัวอุปกรณ์ Device ID, Device Address และ Mode ในการเขียนหรืออ่านข้อมูล
- เมื่ออุปกรณ์รับทราบว่าไมโครคอนโทรลเลอร์ต้องการจะติดต่อด้วย ก็ต้องส่งสถานะรับรู้ (Acknowledge) หรือแจ้งให้ไมโครคอนโทรลเลอร์รับรู้ว่าข้อมูลที่ได้ส่งมามีความถูกต้อง
- และเมื่อสิ้นสุดการส่งข้อมูลไมโครคอนโทรลเลอร์ จะต้องส่งสถานสิ้นสุด (STOP Conditions) เพื่อบอกกับอุปกรณ์ว่า สิ้นสุดการใช้บัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.2.1 สถานะบัสว่าง

- สัญญาณ SDA และ SCL มีระดับสัญญาณเป็น 1 ทั้งคู่

2.6.2.2 การกำหนดสถานะเริ่มต้นและสิ้นสุดของ I^2C BUS (START and STOP Conditions)

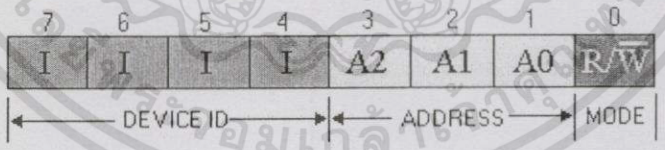


รูปที่ 2.18 การกำหนดสถานะเริ่มต้นและสิ้นสุดของ I^2C BUS

ลักษณะการกำหนดสถานะเริ่มต้นและสถานะสิ้นสุดของ I^2C BUS

- เมื่อต้องการส่งข้อมูล ไมโครคอนโทรลเลอร์จะต้องส่งค่าสถานะเริ่มต้น (START Conditions) คือให้ SDA เปลี่ยนจาก 1 เป็น 0 ในขณะที่ SCL มีค่าเป็น 1
- เมื่อสิ้นสุดการใช้บัส ไมโครคอนโทรลเลอร์จะต้องส่งค่าสถานะสิ้นสุด (STOP Conditions) คือให้ SDA เปลี่ยนจาก 0 เป็น 1 ในขณะที่ SCL มีค่าเป็น 1

2.6.2.3 รหัสควบคุมของ I^2C BUS (Control Byte)



รูปที่ 2.19 รหัสควบคุมของ I^2C BUS

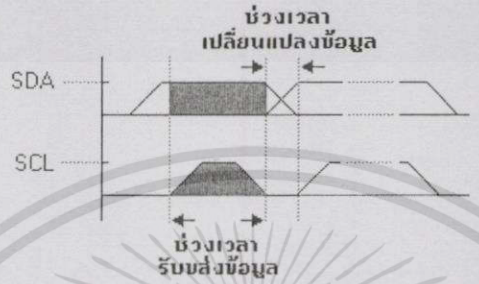
รหัสควบคุมของ I^2C BUS ประกอบด้วยรหัสประจำตัวของอุปกรณ์ (Device ID) ประกอบด้วยบิต 1-7 และบิต 0 เป็นบิตควบคุมการเขียนอ่าน

- รหัสประจำตัวของอุปกรณ์ ประกอบด้วยรหัสประจำตัวจากผู้ผลิต (Product ID) 4 บิต (บิต 4-7) ที่เปลี่ยนแปลงแก้ไขไม่ได้และ Device Address 3 บิต (บิต 1-3) ซึ่งผู้ใช้สามารถกำหนดเองได้ รวมแล้วเป็นรหัส 7 บิต ใช้ระบุตัวอุปกรณ์ที่ต่ออยู่บนบัส จะมีค่าซ้ำกันไม่ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- บิตควบคุมการเขียนอ่าน(Mode) บิต 0 เมื่อไมโครคอนโทรลเลอร์ต้องการเขียนข้อมูลไปยังอุปกรณ์จะกำหนดให้บิตนี้เป็น 0 และเมื่อต้องการอ่านข้อมูลจากอุปกรณ์จะกำหนดให้บิตนี้เป็น 1

2.6.2.4 ช่วงเวลารับส่งบิตข้อมูลของ I^2C BUS

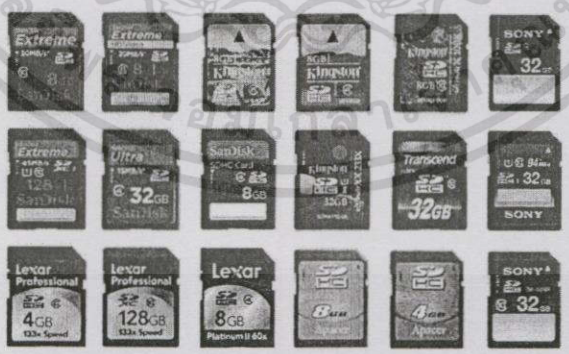


รูปที่ 2.20 ช่วงเวลาการรับส่งบิตข้อมูลของ I^2C BUS

- สภาวะการรับ-ส่งข้อมูล จะกระทำในขณะที่ SCL เป็น 1
- สภาวะการเปลี่ยนแปลงข้อมูล จะกระทำในขณะที่ SCL เป็น 0

2.7 Secure Digital Card (SD Card)

คิดค้นขึ้นโดย 3 บริษัท คือ Toshiba/Panasonic/Sandisk ใช้ในกล้องดิจิตอลหลายยี่ห้อและกล้องรุ่นที่มีขนาดเล็กเป็นพิเศษ นอกจากนี้ยังนิยมใช้ในโทรศัพท์มือถือและอุปกรณ์ PDA เพราะขนาดของการ์ดของการ์ดเล็ก สามารถอ่านเขียนข้อมูลได้เร็ว มีตัวเลือกให้เลือกหลายรุ่น หลายขนาด เป็นตัวเลือกที่น่าสนใจ เพราะสามารถใช้ร่วมกับอุปกรณ์อื่นๆ ได้ง่าย ลักษณะภายนอกของ Secure Digital Card แสดงดังรูปที่ 2.21



รูปที่ 2.21 Secure Digital Cards (SD Card)

2.7.1 ประวัติของSD Card

Secure Digital Card (SD Card) เป็นรูปแบบการ์ดที่เป็นหน่วยความจำแบบ non-volatile โดยเป็นการร่วมกันพัฒนาของ Matsushita SanDisk และ Toshiba Secure Digital Card นั้นนิยมนำมาใช้กับอุปกรณ์พกพาขนาดเล็กต่างๆ หลายประเภท ไม่ว่าจะเป็น digital cameras,

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PDAs, mobile phones, GPS receivers and video game console ความจุมาตรฐานของ Secure Digital (SD) สามารถมีความจุได้ตั้งแต่ 4 MB ถึง 4 GB ยังมีรุ่นที่สามารถมีความจุสูง คือ SDHC สามารถมีความจุได้ถึง 4 GB ถึง 32 GB อย่างไรก็ตามขณะที่ความจุสามารถทำได้ถึง 4 GB ซึ่งออกมาในรุ่นที่เป็น SDHC แต่อุปกรณ์เก่าก็ยังไม่สามารถอ่านรูปแบบของการ์ดแบบ SDHC ได้ ต้องทำการ up firmware เพื่อให้สามารถรองรับหรือใช้การ์ดที่เป็นแบบ SDHC ได้ ในเดือนสิงหาคม ปี 1999 บริษัท Panasonic SanDisk และ Toshiba ได้ร่วมกันพัฒนา Secure Digital Card (SD Card) ดังรูปที่ 2.22 ได้พัฒนาขึ้นมาอยู่บนพื้นฐานของการ์ดหน่วยความจำแบบ MMC Card เท่ากัน นั่นคือมีความยาว 32 มิลลิเมตร และกว้าง 24 มิลลิเมตร แต่จะมีความหนาที่มากกว่า MMC Card อยู่เล็กน้อย นั่นคือมีความหนา 2.1 มิลลิเมตร โดยทางบริษัท Toshiba ได้เพิ่มความสามารถทางด้านฮาร์ดแวร์ที่ทำหน้าที่เข้ารหัสข้อมูลไปใน เทคโนโลยีเดิมของ MMC Card และยังใส่เทคนิคพิเศษที่เรียกว่า DRM (Digital Rights Management) ซึ่งเป็นตัวจัดการเกี่ยวกับเรื่องของลิขสิทธิ์ของข้อมูลมาให้ ซึ่งถือได้ว่าเป็นหน่วยความจำที่เป็น high memory ในช่วงเวลานั้น แต่ยังมี ทาง Sony ได้มีการพัฒนา Card ขึ้นมาเป็นคู่แข่งกับ SD Card คือ Memory Stick ซึ่งได้มีการเปิดตัวก่อน SD Card แต่ไม่ได้รับความนิยม

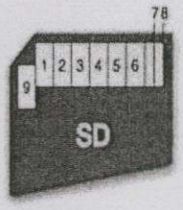
รูปที่ 2.22 SD Card ของ Panasonic ขนาด 64 MB

2.7.2 คุณสมบัติของ SD Card

การบันทึกไฟล์ อ่านไฟล์ ลบไฟล์ และอื่นๆ ทั้งหมดนี้ถ้าทำงานบนระบบปฏิบัติการ Window หรือ Linux ปัจจุบันแค่ใช้ mouse click แต่สำหรับระบบฝังตัวที่ประกอบ MCU นั้น หมายความว่า MCU ต้องทำหน้าที่เขียนไฟล์ อ่านไฟล์และอื่นๆ ทั้งหมด การอ่านหรือเขียนข้อมูลกับ SD Card ทำได้สองลักษณะ คือ ใช้รูปแบบของ FAT หรืออ่านเขียนข้อมูลเป็น Byte นั้นทำให้ Computer ไม่สามารถเข้าใจข้อมูลได้ SD Card ในตอนแรกมีความจุตั้งแต่ 4MB – 4GB แต่มีการพัฒนาอย่างต่อเนื่องทำให้ความจุของ SD Card มากขึ้นเรียกว่า SDHC (Secure Digital High Capacity) ซึ่งมีความจุตั้งแต่ 4GB – 32GB และจะพัฒนาไปเป็น SDXC ซึ่งจะมีความจุตั้งแต่ 32GB – 2TB โดยโครงสร้างขาของ SD Card แสดงดังรูปที่ 2.23 และหน้าที่ของแต่ละขาแสดงดังตารางที่ 2.5

การติดต่อสื่อสารระหว่าง MCU กับ SD Card ใช้โหมด SPI การทำงานในโหมดนี้ใช้สายสัญญาณเพียง 3 เส้น แต่ความเร็วในการรับส่งข้อมูลก็จะช้าลงไปด้วย แต่เมื่อเทียบกับการใช้งานส่วนใหญ่แล้วไม่ต้องการความเร็วมากนัก ความเร็วในการรับส่งข้อมูลจึงไม่เป็นปัญหาอะไร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Pin	SD	SPI
1	CD/DAT3	CS
2	CMD	DI
3	VSS1	VSS1
4	VDD	VDD
5	CLK	SCLK
6	VSS2	VSS2
7	DAT0	DO
8	DAT1	X
9	DAT2	X

รูปที่ 2.23 โครงสร้างขาของ SD Card

ตารางที่ 2.5 ตารางแสดงรายละเอียดโครงสร้างขาของ SD Card

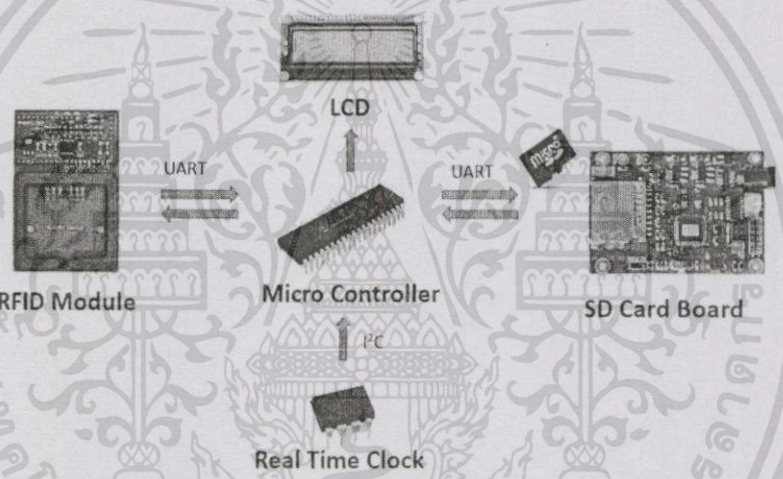
Pin 1	ในโหมดของ SPI ทำหน้าที่เป็นขา CS หรือ Chip Select
Pin 2	ในโหมดของ SPI ทำหน้าที่เป็นขา DI หรือ Data Input
Pin 3	ในโหมดของ SPI ทำหน้าที่เป็นขา GND หรือ Ground
Pin 4	ในโหมดของ SPI ทำหน้าที่เป็นขา POWER หรือ +5V
Pin 5	ในโหมดของ SPI ทำหน้าที่เป็นขา CLK หรือ Clock
Pin 6	ในโหมดของ SPI ทำหน้าที่เป็นขา GND หรือ Ground
Pin 7	ในโหมดของ SPI ทำหน้าที่เป็นขา DATA0 หรือ Data Out
Pin 8	ในโหมดของ SPI ไม่ใช้
Pin 9	ในโหมดของ SPI ไม่ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การดำเนินงาน

3.1 บทนำ

บทนี้จะกล่าวถึงองค์ประกอบของระบบ การทำงานโดยรวมของระบบ และรายละเอียดในการออกแบบทั้งหมด ในการออกแบบระบบของเครื่องบันทึกเวลาแบบ RFID นั้น มีองค์ประกอบโดยรวมซึ่งแสดงได้ ดังรูปที่ 3.1 หลักการทำงานคือ เมื่อมีบัตรนักศึกษาที่เป็นแบบ RFID มาแตะที่เครื่องอ่านแบบ RFID เครื่องอ่านจะทำการอ่านรหัสจากบัตรแล้วนำมาแสดงที่จอแสดงผลแอลซีดี พร้อมกับเวลาที่ทำการแตะบัตร โดยข้อมูลที่ได้จะนำไปเก็บใน SD Card ในรูปแบบไฟล์.csv ซึ่งสามารถนำไปใช้กับเครื่องคอมพิวเตอร์ได้เลย



รูปที่ 3.1 โครงสร้างการทำงานของเครื่องบันทึกเวลาแบบ RFID

3.2 โครงสร้างการทำงานของเครื่องบันทึกเวลาแบบ RFID

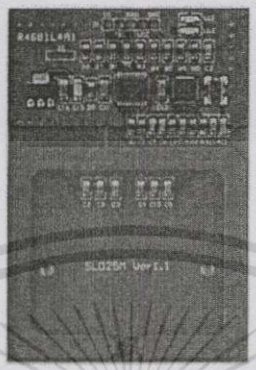
เครื่องบันทึกเวลาแบบ RFID ประกอบไปด้วยส่วนต่างๆ ทั้งหมด 5 ส่วน ดังแสดงตามรูปที่ 3.1 คือ โครงงานนี้ได้ออกแบบให้ส่วนของ RFID MODULE ทำงานโดยการอ่านข้อมูลจากบัตรนักศึกษาที่นำมาแตะบนตัวเครื่อง ซึ่งมีไมโครคอนโทรลเลอร์ทำหน้าที่ในการส่งคำสั่งไปยัง RFID MODULE เพื่อตรวจสอบสัญญาณการแตะบัตรแล้วอ่านข้อมูลภายในบัตรและรับข้อมูลจากบัตร โดยไมโครคอน-โทรลเลอร์จะทำหน้าที่เก็บข้อมูลภายในบัตรและเวลาที่ทำการแตะบัตรจาก Real Time Clock เพื่อนำมาแสดงผลที่จอแอลซีดีพร้อมกับส่งข้อมูลไปเก็บใน SD Card

3.2.1 RFID MODLE

RFID MODLE เป็นโมดูลสำหรับอ่านและเขียนบัตร RFID เป็นอุปกรณ์สำหรับการอ่านหรือเขียนข้อมูลต่างๆ ลงในแท็ก โดยมีมาตรฐานความถี่ที่ใช้งาน คือ ความถี่ 13.56 MHz ในการรับ – ส่งข้อมูลหรือที่เรียกกันโดยทั่วไปว่า Mifare มีรูปแบบการติดต่อสื่อสารรูปแบบไบต์ มีการเชื่อมต่อแบบ UART (Universal Asynchronous Receiver transmitter) กับส่วนของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไมโครคอนโทรลเลอร์ซึ่งมีอัตราการส่งผ่านข้อมูลที่ประมาณ 9,600 - 115,200 บิตต่อวินาที มีระยะการรับ-ส่งสัญญาณที่ระยะสูงสุดไม่เกิน 8 เซนติเมตร ซึ่งในโครงงานนี้เลือกใช้ RFID MODULE ของยี่ห้อ StrongLink รุ่น SL025M โดยมีลักษณะดังรูปที่ 3.2



รูปที่ 3.2 RFID MODULE ของยี่ห้อ StrongLink รุ่น SL025M

รูปแบบการติดต่อสื่อสารเป็นแบบรับส่งทีละไบต์อยู่ในรูปแบบเลขฐานสิบหก ซึ่งมีพารามิเตอร์ต่างๆ เป็นดังนี้

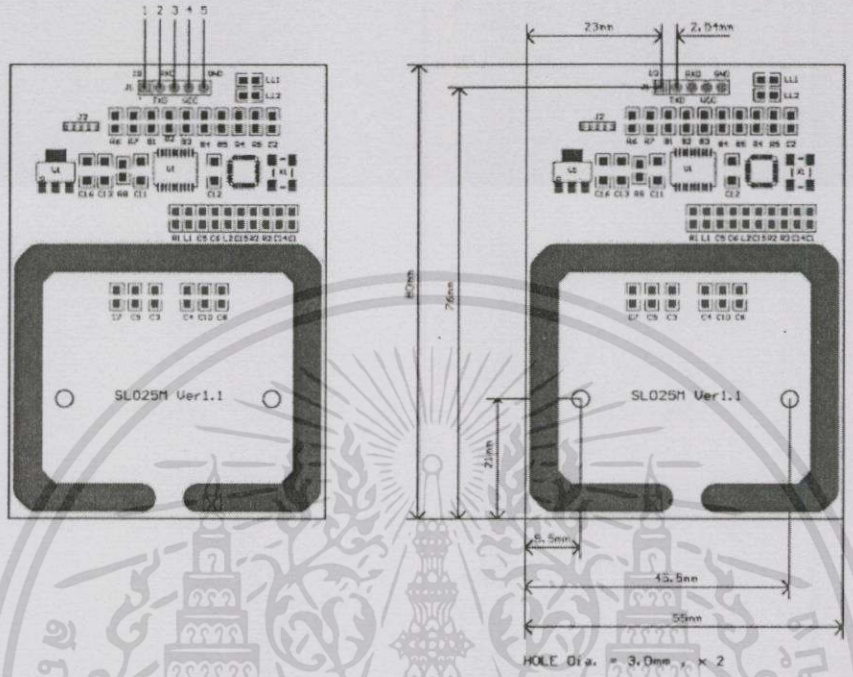
- Baud Rate: 9,600 - 115,200 bps
- Data bits: 8 bits
- Start Bit: 1 bit
- Stop Bit: 1 bit
- Parity: None
- Flow control: None
- Frequency 13.56MHz
- Protocol ISO14443A (Mifare)
- Tag supported Ultralight, Mifare Mini, Mifare 1k, Mifare 4k, and FM11RF08
- Interface UART TTL (TTL Interface, ถ้าผู้ใช้ต้องการต่อกับ PC ต้องเปลี่ยนระดับแรงดันให้เป็น RS232 Level โดยเลือกใช้ IC เช่น MAX232)
- Supply voltage 4.4 -12.0 VDC
- Dimension 86 x 55 mm

3.2.1.1 การนำ RFID MODULE ไปต่อใช้งานกับ PIC18F458

การทำงานของ RFID MODULE นั้นจะต้องติดต่อกับไมโครคอนโทรลเลอร์ในรูปแบบ UART (Universal Asynchronous Receiver transmitter) เพื่อทำการรับ - ส่งข้อมูลภายในบัตรและอ่านบัตร RFID ซึ่งไมโครคอนโทรลเลอร์ที่ใช้ในโครงงานนี้เป็นชนิด PIC18F458 โดยในการเชื่อมต่อกับไมโครคอนโทรลเลอร์ จะต้องจ่ายไฟเลี้ยงให้กับไมโครคอนโทรลเลอร์และRFID MODULE ที่ระดับแรงดัน 3.3 – 5V จึงจะทำงานได้ เนื่องจากการเชื่อมต่อแบบ UART นั้น ระดับสัญญาณจะต้องมีระดับแรงดันไฟที่เท่ากัน จึงจะไม่เกิดความผิดพลาดในการรับ-ส่งข้อมูล โดยใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงงานนี้จะใช้ภาษาซีทำการเขียนโปรแกรมในการติดต่อสื่อสารกันระหว่าง RFID MODULE กับไมโครคอนโทรลเลอร์ ซึ่งลักษณะของ RFID MODULE แสดงดังรูปที่ 3.3



รูปที่ 3.3 แสดงลักษณะของ RFID MODULE ของยี่ห้อ StrongLink รุ่น SL025M

ตารางที่ 3.1 แสดงรายละเอียดขาของ RFID MODULE

PIN	SYMBOL	TYPE	DESCRIPTION
1	TagSta	Output	Tag detect signal Low level indicating tag in detection range High level indicating tag out
2	TXD	Output	Serial output port
3	RXD	Input	Serial input port
4	VCC	PWR	Power Supply
5	GND	PWR	Ground

3.2.1.2 รูปแบบการติดต่อสื่อสาร (COMMUNICATION PROTOCOL)

ค่าพารามิเตอร์ต่างๆ ที่ใช้ในชุดคำสั่งเพื่อทำการติดต่อระหว่าง RFID MODULE กับไมโครคอนโทรลเลอร์ PIC18F458 จะแสดงในตารางที่ 3.2 และสถานะระหว่าง RFID MODULE กับไมโครคอนโทรลเลอร์ PIC18F458 ซึ่งแสดงในตารางที่ 3.3 เนื่องจากโครงงานนี้ต้องการอ่าน UID ในบัตร RFID ดังนั้นจึงต้องใช้ชุดคำสั่งในการอ่าน UID (READ UID) คือ เป็นชุดคำสั่งที่ใช้ทำการอ่าน UID จากบัตร RFID ที่นำมาแท็ก Host to SL025 (ไมโครคอนโทรลเลอร์จะส่งชุดคำสั่งไปให้ RFID MODULE) แสดงในตารางที่ 3.4 รวมถึงรูปแบบการส่งข้อมูล SL025 to Host

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(RFID MODULE จะส่งชุดข้อมูลพร้อมทั้ง UID ของบัตร RFID ที่นำมาแท็กกลับมาให้ไมโครคอนโทรลเลอร์) แสดงในตารางที่ 3.5

ตารางที่ 3.2 Command Overview

Command	Description
0x01	Select Mifare card
0x02	Login to a sector
0x03	Read a data block
0x04	Write a data block
0x05	Read a value block
0x06	Initialize a value block
0x07	Write master key (key A)
0x08	Increment Value
0x09	Decrement value
0x0A	Copy value
0x10	Read a data page (UltraLight)
0x11	Write a data page (UltraLight)
0x12	Download Key
0x13	Login sector via stored Key
0x50	Go to Power Down mode

ตารางที่ 3.3 Status Overview

Status	Description
0x00	Operation succeed
0x01	No tag
0x02	Login succeed
0x03	Login fail
0x04	Read fail
0x05	Write fail
0x06	Unable to read after write
0x08	Address overflow
0x09	Download Key fail
0x0D	Not authenticate
0x0E	Not a value block
0xF0	Checksum error
0xF1	Command code error

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.4 รูปแบบการส่งข้อมูล SL025 to Host

Preamble	Len	Command	Status	Data	Checksum

Preamble	1 byte equal to 0xBD
Len	1 byte indicating the number from Command to Checksum
Command	1 byte Command code, see Table 3.2
Status	1 byte Command status, see Table 3.3
Data	Variable length depends on the command type.
Checksum	1 byte XOR of all the bytes from Preamble to Data

ตารางที่ 3.5 รูปแบบการส่งข้อมูล SL025 to Host

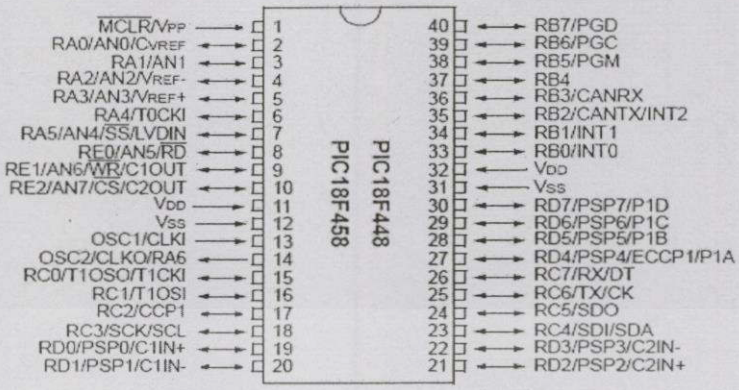
Preamble	Len	Command	Status	Data	Checksum

Preamble	1 byte equal to 0xBD
Len	1 byte indicating the number from Command to Checksum
Command	1 byte Command code, see Table 3.2
Status	1 byte Command status, see Table 3.3
Data	Variable length depends on the command type.
Checksum	1 byte XOR of all the bytes from Preamble to Data

3.2.2 ไมโครคอนโทรลเลอร์ (MICROCONTROLLER)

โครงงานนี้ใช้การเขียนโปรแกรมภาษาซี ในการควบคุมการทำงานระหว่างไมโครคอนโทรลเลอร์กับอุปกรณ์ต่างๆ และใช้คอมไพเลอร์ของ CCS เป็นตัวสร้าง HEX ไฟล์เพื่อทำการเบิร์นลงสู่ไมโครคอนโทรลเลอร์ให้สามารถนำมาใช้งานได้ ส่วนไมโครคอนโทรลเลอร์ที่ใช้ในโครงงานเป็นของบริษัท MICRO CHIP เป็นไมโครคอนโทรลเลอร์ชนิด PIC เบอร์ PIC18F458 ซึ่งแสดงรูปขาของ PIC18F458 ดังรูปที่ 3.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 การจัดการขาของ PIC18F458

ไมโครคอนโทรลเลอร์ PIC18F458 มีคุณสมบัติสำคัญ ดังนี้

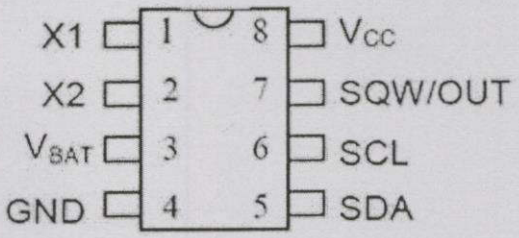
- ทำงานได้สูงสุดที่ 40MHz
- หน่วยความจำโปรแกรมเป็นแบบ Flash มีขนาด 32 กิโลไบต์
- มีแรมขนาด 1536 ไบต์
- มี EEPROM ขนาด 256 ไบต์
- มีระบบ Code Protection
- มี CAN คอนโทรลเลอร์ในตัว
- สามารถเปิดโปรแกรมด้วยไฟ +5VDC ได้
- ทำงานที่ไฟเลี้ยงตั้งแต่ 2VDC ถึง 5.5VDC
- มีระบบ UART สำหรับต่อกับการสื่อสารแบบ RS232
- มี I/O พอร์ตทั้งหมด 5 พอร์ต

จากคุณสมบัติที่สำคัญข้างต้น โครงงานนี้จึงเลือกใช้ไมโครคอนโทรลเลอร์ PIC18F458 เพราะว่ามีคุณสมบัติตามที่โครงงานต้องการ เช่น โครงงานมีการเชื่อมต่อกับ RFID MODULE ที่ไฟเลี้ยง 3.3VDC -5VDC ซึ่งต้องการไมโครคอนโทรลเลอร์ที่ทำงานที่ไฟเลี้ยงเพียง 3.3VDC -5VDC ได้ และโครงงานนี้ต้องการไมโครคอนโทรลเลอร์ที่มีพอร์ต I/O มากกว่า 3 พอร์ต เป็นต้น

3.2.3 DS1307

DS1307 เป็นไอซีฐานเวลาของดัลลัสเซมิคอนดักเตอร์ (Dallas Semiconductor) มีบัสรับ-ส่งข้อมูลแบบ I^2C ซึ่งเป็นแบบ 2 wire สามารถสื่อสารได้ 2 ทิศทาง (bi-direction bus) ฐานเวลาของ DS1307 นั้นสามารถเก็บข้อมูล วินาที, นาที, ชั่วโมง, วัน, วันที่, เดือน และปีได้ ระบบเวลาสามารถทำงานโหมดรูปแบบ 24 ชั่วโมง หรือ 12 ชั่วโมง AM/PM ก็ได้ ภายในมีระบบตรวจจับแหล่งจ่ายไฟ โดยถ้าแหล่งจ่ายไฟหลักถูกตัดไป DS1307 สามารถสวิตช์ไปใช้ไฟจากแบตเตอรี่และทำงานต่อไป โดยที่ยังสามารถรักษาข้อมูลไว้ได้ โครงสร้างมีขาทั้งหมด 8 ขา ดังแสดงในรูปที่ 3.5 และมีรายละเอียดการทำงานของขาต่างๆ ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 ตำแหน่งขาไอซี RTC DS1307

- VCC: ใช้ต่อไฟเลี้ยง +5V
- GND: ใช้ต่อกราวด์
- VBAT: ใช้ต่อกับแบตเตอรี่ 3V เพื่อรักษาการทำงาน ในกรณีที่ไม่มีไฟเลี้ยงจ่าย
- SDA: ขารับ-ส่งข้อมูลด้วยระบบบัส I^2C
- SCL: ขาสัญญาณนาฬิกาสำหรับการรับ-ส่งข้อมูลด้วยระบบบัส I^2C
- SQW/OUT: ขาเอาต์พุตสัญญาณ Square Wave สามารถเลือกความถี่ได้
- X1, X2: ใช้ต่อกับคริสตอลความถี่มาตรฐาน 32.768 kHz เพื่อสร้างฐานเวลาจริงให้กับไอซี

3.2.3.1 การรับ-ส่งข้อมูลแบบ I^2C นั้นมีข้อกำหนดอยู่ 2 ประการด้วยกัน คือ

1. การรับ-ส่งข้อมูลจะเริ่มขึ้นได้เมื่อบัสมีสถานะว่างเท่านั้น
2. ในช่วงที่ทำการรับ-ส่งข้อมูลอยู่ สายสัญญาณ SDA ต้องไม่เปลี่ยนสถานะในช่วงที่ SCL มีสถานะเป็นลอจิก "1" ถ้า SDA มีการเปลี่ยนสถานะในช่วงที่ SCL เป็นลอจิก "1" จะถือว่าเป็นสัญญาณควบคุมการรับ-ส่งข้อมูล

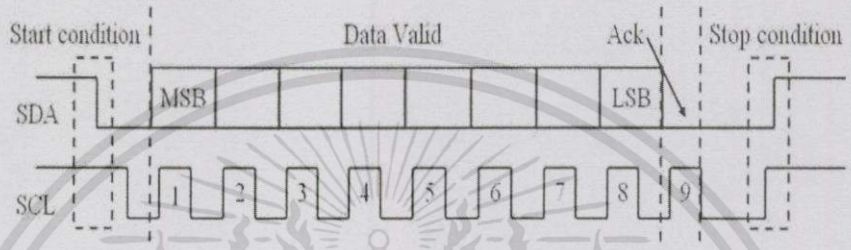
3.2.3.2 สถานะของการรับ-ส่งข้อมูลแบบ I^2C สามารถแบ่งออกได้เป็น 5 สถานะด้วยกันดังแสดงในรูปที่ 3.6 และมีรายละเอียดดังนี้

1. สถานะว่าง (Bus not busy): สัญญาณ SDA และ SCL มีระดับสัญญาณเป็น High
2. เริ่มส่งข้อมูล (Start data transfer): มีการเปลี่ยนระดับสัญญาณของ SDA จาก High เป็น Low ในขณะที่ SCL มีระดับสัญญาณเป็น High ค้างไว้
3. หยุดส่งข้อมูล (Stop data transfer): มีการเปลี่ยนระดับสัญญาณของ SDA จาก Low เป็น High ในขณะที่ SCL มีระดับสัญญาณเป็น High ค้างไว้
4. รับ-ส่งข้อมูล (Data valid): มีการรับ-ส่งข้อมูลผ่านสายสัญญาณ SDA โดยข้อมูลแต่ละบิตจะถูกส่งในช่วงที่ SCL มีระดับเป็น High โดยในช่วงที่ SCL มีสถานะเป็น High อยู่นั้น SDA จะต้องไม่เกิดการเปลี่ยนระดับสัญญาณ
5. SDA จะเปลี่ยนระดับของสัญญาณ ในช่วงที่ SCL มีระดับสัญญาณเป็น Low เท่านั้น ตามมาตรฐานการส่งข้อมูล แบบ I^2C นี้สามารถส่งข้อมูลด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความถี่สัญญาณนาฬิกาสูงสุด 100 kHz ที่โหมดการทำงานธรรมดา และ 400 kHz ที่โหมดการทำงานแบบเร็ว แต่สำหรับ DS1307 สามารถทำงานได้ในโหมดธรรมดาเท่านั้น

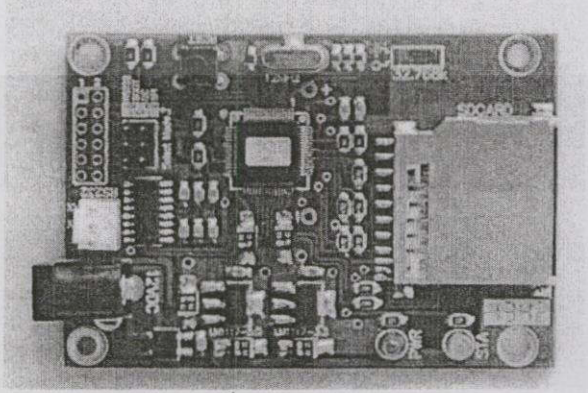
6. ตอบรับ (Acknowledge): เกิดขึ้นหลังจากที่มีการรับ-ส่งข้อมูลครบแล้ว โดยอุปกรณ์มาสเตอร์ต้องสร้างสัญญาณ Clock บน SCL เพิ่มอีกลูก อุปกรณ์ที่เป็นตัวรับข้อมูลจะดึงระดับสัญญาณบน SDA ให้เป็น Low เพื่อให้ตัวส่งรับรู้ว่าตัวรับได้รับข้อมูลครบแล้ว



รูปที่ 3.6 การรับ-ส่งข้อมูลผ่านบัส I^2C

3.2.4 SD CARD

ALFAT-SD คือบอร์ดอ่านเขียน SD-CARD ทำงานด้วย ALFAT SoC processor ที่มีประสิทธิภาพสูงรองรับหน่วยความจำของ SD-CARD ความจุได้ถึง 8 GB รองรับไฟล์ได้ 2 ระบบคือ FAT16 และ FAT32 ประยุกต์ใช้งานได้หลากหลายต่อกับไมโครคอนโทรลเลอร์ (MCU) ได้ 4 รูปแบบ คือ I2C, SPI, UART (TTL), RS232 หรือคอมพิวเตอร์ (Computer) ด้วย RS232 สามารถเลือกโหมดการติดต่อสื่อสารด้วยจั้มเปอร์ในกรณีใช้งานแบบ UART (TTL), RS232 และ Boot loader (สำหรับ Update Firmware) สามารถเลือกด้วยจั้มเปอร์ได้เลย แต่ในกรณีใช้งานแบบ I2C และ SPI ควรจะใช้ไมโครคอนโทรลเลอร์ (MCU) ในการควบคุมขา SPI_SSEL และ SPI_MOSI โดยตรงด้วยขั้ว I/O 12 PIN และมีไฟ LED POWER (สีแดง) และไฟแสดงสถานะการติดต่อสื่อสารด้วย LED สีเขียวคือขณะทำงาน LED จะติดและดับเมื่อการทำงานเสร็จสิ้นสำหรับความเร็วการสื่อสาร (Baud Rate) UART (TTL), RS232 ค่าเริ่มต้นจากโรงงานคือ Baud Rate = 115200 bps Start bit = 1, Data bits = 8, Parity = No, Stop bit = 1

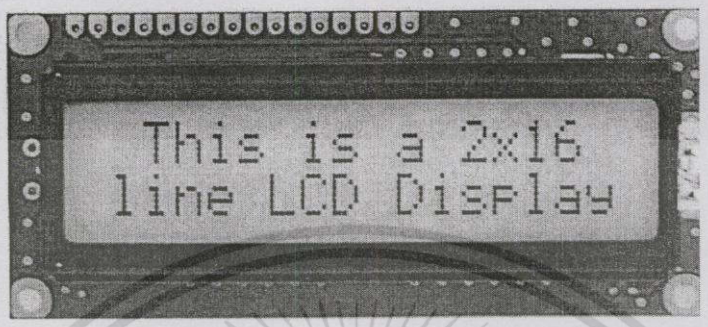


รูปที่ 3.7 SD Card

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.5 โมดูลแอลซีดีขนาด 16 ตัวอักษร 2 บรรทัด (LCD 16x2)

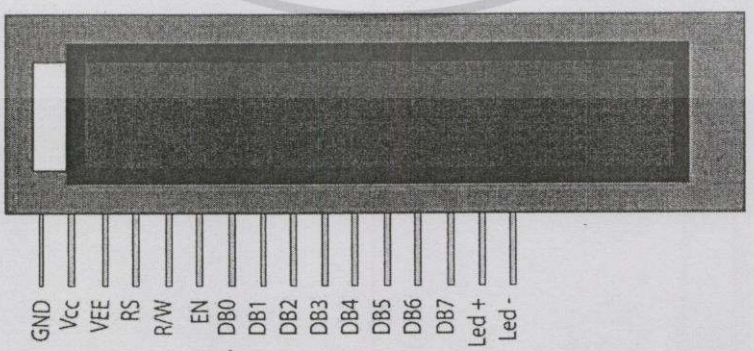
โมดูลแอลซีดีแบบ 16 ตัวอักษร 2 บรรทัดเป็นโมดูลที่มีราคาถูก หาได้ง่าย และเป็นโมดูลแอลซีดีที่มีโครงสร้างเป็นมาตรฐาน



รูปที่ 3.8 แอลซีดีขนาด 16x2

โมดูลแอลซีดีขนาด 16x2 มีขาต่อใช้งานทั้งสิ้น 14 ขา มีการจัดขาดังรูปที่ 3.9 สำหรับรายละเอียดการทำงานของแต่ละขามีดังนี้

- VSS (ขา 1): ต่อกราวด์
- VDD (ขา2): ต่อไฟเลี้ยง + 5โวลต์
- VO (ขา 3): เป็นขาอินพุตรับแรงดันเพื่อปรับความเข้มของการแสดงผล
- RS (ขา4): เป็นขาอินพุตใช้ในการแยกชนิดของข้อมูลที่ทำการประมวลผลในขณะนั้นว่าเป็นคำสั่งสำหรับรีจิสเตอร์ IR หรือเป็นข้อมูลสำหรับรีจิสเตอร์ DR โดยถ้าขานี้เป็น 0 ข้อมูลที่ส่งมาจะเป็นคำสั่ง แต่ถ้าขานี้เป็น 1 ข้อมูลที่ส่งมาจะเป็นข้อมูลสำหรับแสดงผล
- R/W (ขา5): เป็นขาที่ใช้เลือกการอ่านหรือเขียนข้อมูลกับแอลซีดี ถ้าเป็น 0 เป็นการกำหนดให้เขียนข้อมูล แต่ถ้าเป็น 1 จะเป็นการอ่านข้อมูล
- E (ขา6): เป็นขาอีนาเบิลแอลซีดีให้ทำงาน
- D0-D7 (ขา 7-14): เป็นขาที่ใช้เป็นทางผ่านของข้อมูลระหว่างแอลซีดีกับอุปกรณ์ภายนอกขนาด 8 บิต



รูปที่ 3.9 การจัดการขาของแอลซีดี

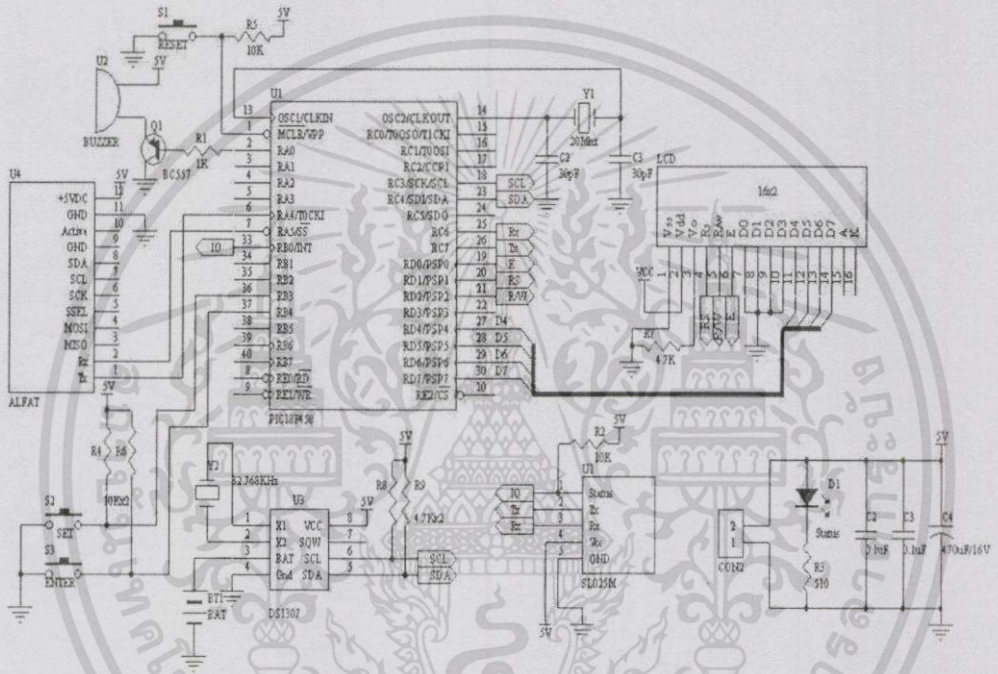
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 รายละเอียดการออกแบบส่วนต่างๆ ของระบบ

รายละเอียดในการออกแบบทั้งหมดแบ่งออกเป็น 2 ส่วนด้วยกัน คือ ส่วนของฮาร์ดแวร์ และซอฟต์แวร์

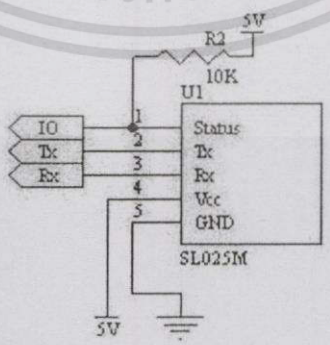
3.3.1 การออกแบบส่วนฮาร์ดแวร์

ในส่วนของฮาร์ดแวร์นั้นสามารถแบ่งออกได้ 4 ส่วน คือ ส่วนของ RFID MODULE, ส่วนของ Real Time Clock, ส่วนของ SD Card และส่วนแอลซีดีโมดูล โดยวงจรรวมของระบบจะเป็นดังรูปที่ 3.10



รูปที่ 3.10 วงจรรวมของเครื่องบันทึกเวลาแบบ RFID

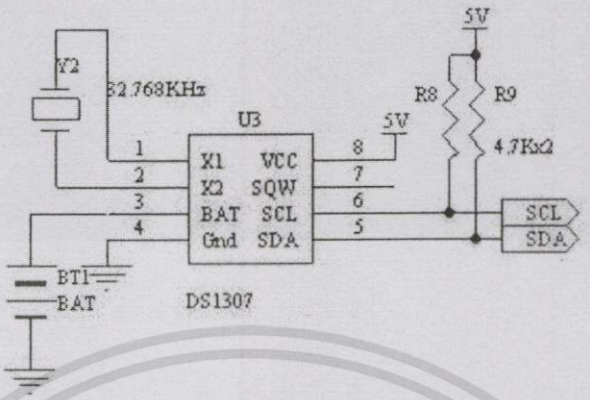
3.3.1.1 วงจรของ RFID MODULE



รูปที่ 3.11 วงจรของ RFID MODULE

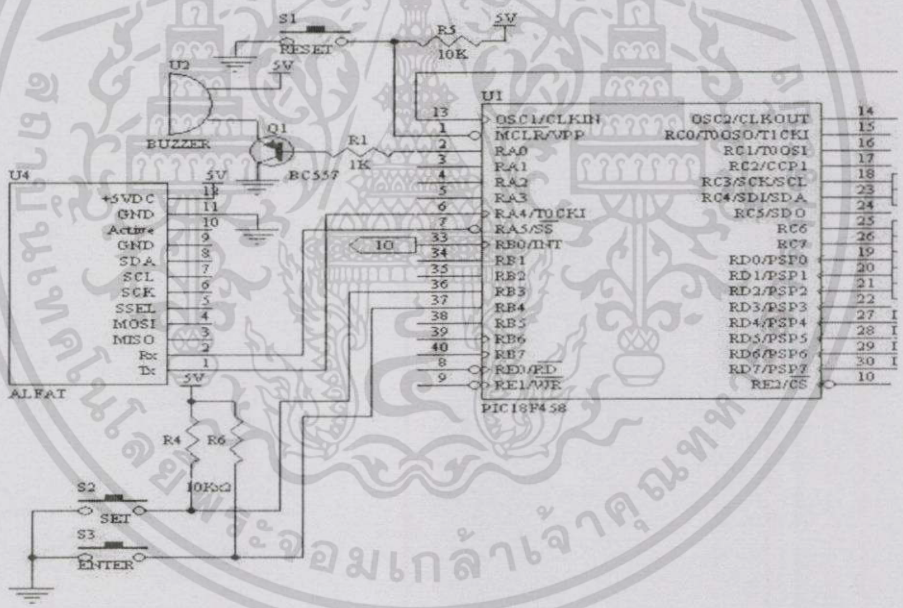
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.1.2 วงจรของ DS1307



รูปที่ 3.12 วงจรของ DS1307

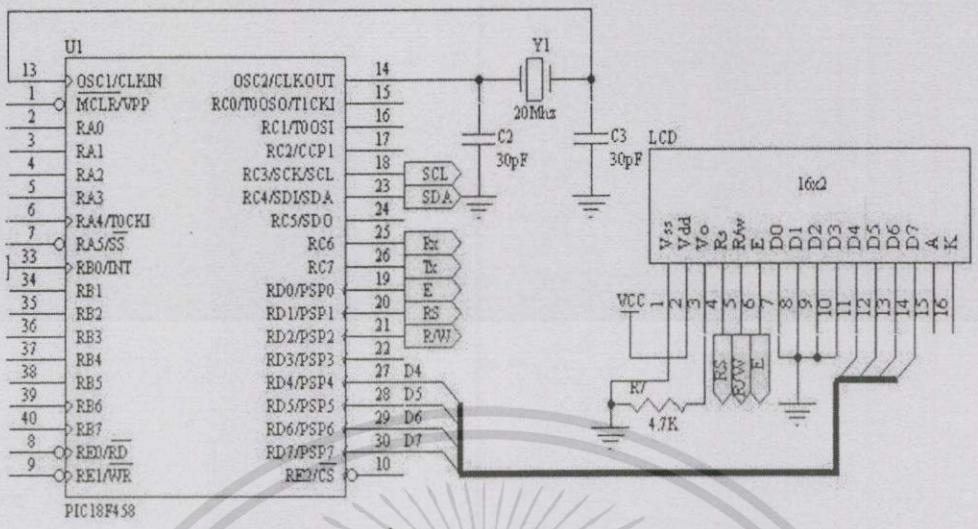
3.3.1.3 วงจรของ SD Card



รูปที่ 3.13 วงจรของ SD Card

3.3.1.4 วงจรของแอลซีดีโมดูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.14 วงจรของแอลซีดีโมดูล

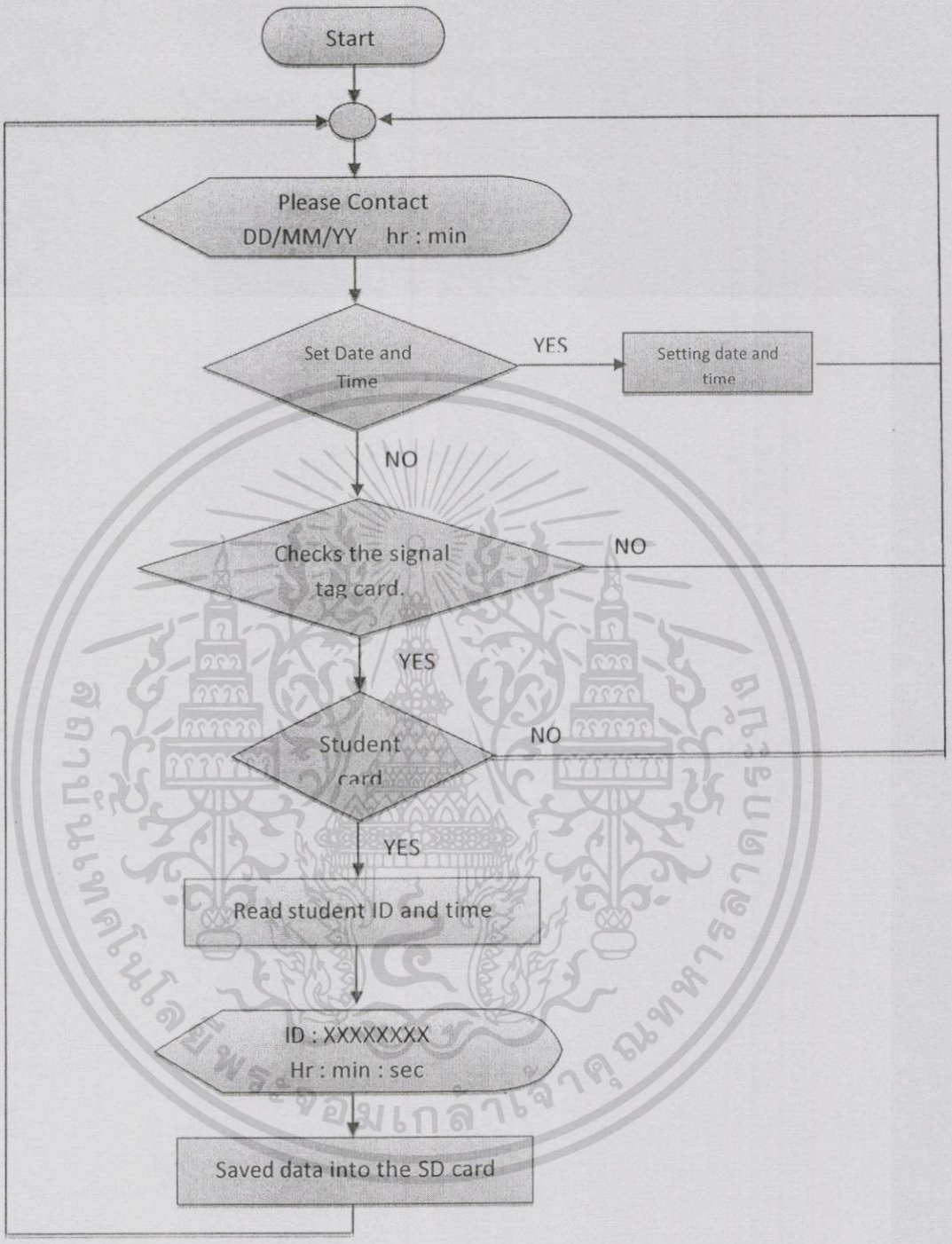
3.3.2 การออกแบบส่วนของซอฟต์แวร์

ในการออกแบบส่วนของซอฟต์แวร์นี้ จะเป็นการออกแบบโปรแกรมการทำงานของระบบ ซึ่งแบ่งออกเป็น 2 ส่วน คือ ส่วนของการทำงานของเครื่องบันทึกเวลาแบบ RFID และส่วนของการตั้งค่าเวลา

3.3.2.1 การทำงานของเครื่องบันทึกเวลาแบบ RFID

เมื่อเปิดเครื่องหน้าจอจะแสดงข้อความ “Please Contact” วัน เดือน ปี และเวลา ณ ขณะนั้น เมื่อไม่มีการตั้งเวลา ก็จะทำการตรวจเช็คว่ามีสัญญาณการแตะบัตรหรือไม่ ถ้าไม่มีจะแสดงข้อความข้างต้น ถ้ามีจอแสดงผลจะโชว์รหัสนักศึกษาและเวลา ณ ขณะนั้น และเก็บข้อมูลที่ได้ลงในเอสดีการ์ด แผนผังการทำงานของเครื่องบันทึกเวลาแบบ RFID แสดงในรูปที่ 3.15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

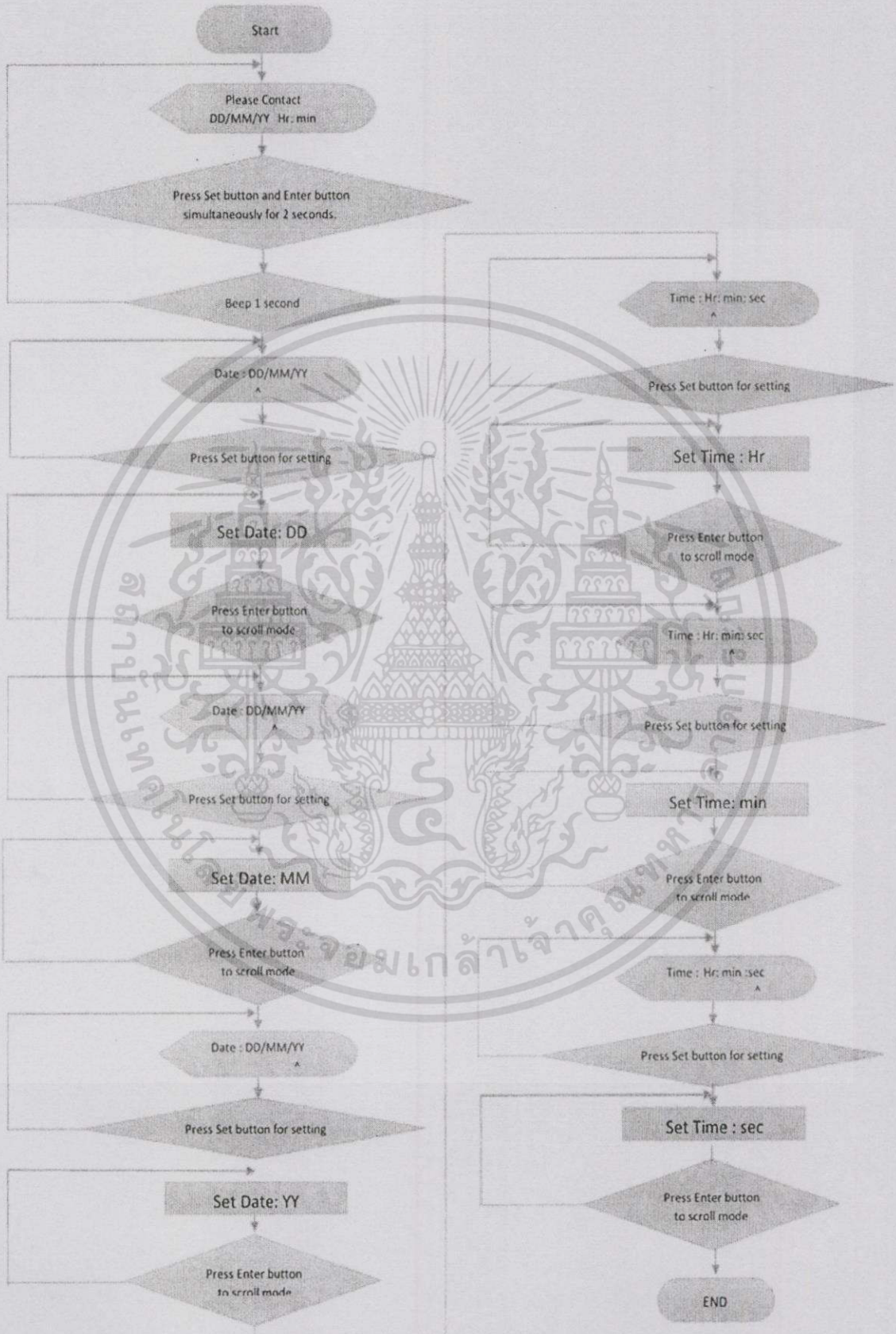


รูปที่ 3.15 แผนผังการทำงานของเครื่องบันทึกเวลาแบบ RFID

3.3.2.2 การตั้งค่าวันและเวลา

กดปุ่ม SET และ ENTER พร้อมกันเป็นเวลา 3 วินาที จนมีเสียงดังขึ้น จะเข้าสู่โหมดการตั้งค่าวัน เดือน ปี ปุ่ม SET ใช้เพิ่มค่าขึ้นทีละหนึ่ง ปุ่ม ENTER ใช้ตกลงเลือกค่านั้น หลังจากตั้งวัน เดือน ปี เสร็จจะเข้าสู่โหมดการตั้งเวลา ชั่วโมง นาที และวินาที แผนผังการตั้งค่าวันและเวลา แสดงตามรูปที่ 3.16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.16 แผนผังการตั้งค่าวันและเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดลองและผลการทดลอง

4.1 บทนำ

ในบทนี้จะกล่าวถึงรายละเอียดการทดลองและผลการทดลองของโครงงาน โดยเครื่องบันทึกเวลาแบบ RFID จะต้องสามารถอ่านข้อมูลจากบัตรนักศึกษา คือ ID ของบัตรและสามารถนำ ID ของบัตรนักศึกษามาแสดงที่หน้าจอแอลซีดีได้ โดยที่ข้อมูล ID ของบัตรนั้นกับเวลา ณ ที่ทำการแตะบัตรกับเครื่องบันทึกเวลาแบบ RFID ถูกจัดเก็บลงใน SD card และสามารถนำไฟล์ที่อยู่ใน SD card มาใช้กับคอมพิวเตอร์ได้เลย

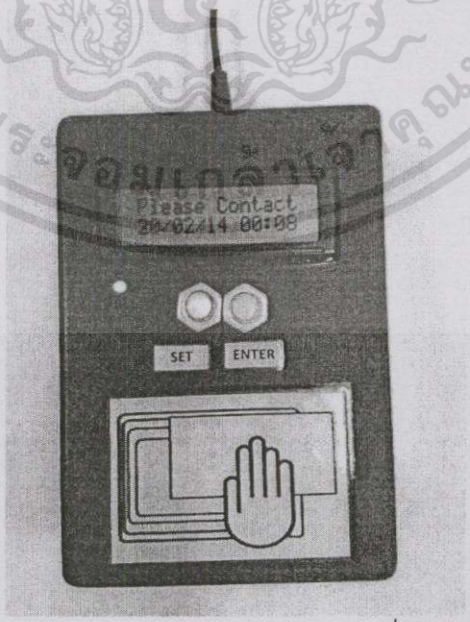
4.2 การทดลองและผลการทดลองของการอ่าน ID ของบัตรนักศึกษา วัน และเวลาแสดงผ่านจอแอลซีดี

4.2.1 การทดลองการอ่าน ID ของบัตรนักศึกษา วัน และเวลา แสดงผ่านจอแอลซีดี

ในการทดลองนี้ เป็นการทดลองนำบัตรนักศึกษามาแตะที่เครื่องบันทึกเวลาแบบ RFID และข้อมูล ID ของบัตรนักศึกษาจะแสดงที่จอแอลซีดี พร้อมเวลา ณ ที่ทำการทาบบัตร ซึ่งบัตรนักศึกษาแต่ละใบนั้นจะมีรหัสไม่ซ้ำกัน

4.2.2 ผลการทดลองการอ่าน ID ของบัตรนักศึกษา วัน และเวลา แสดงผ่านจอแอลซีดี

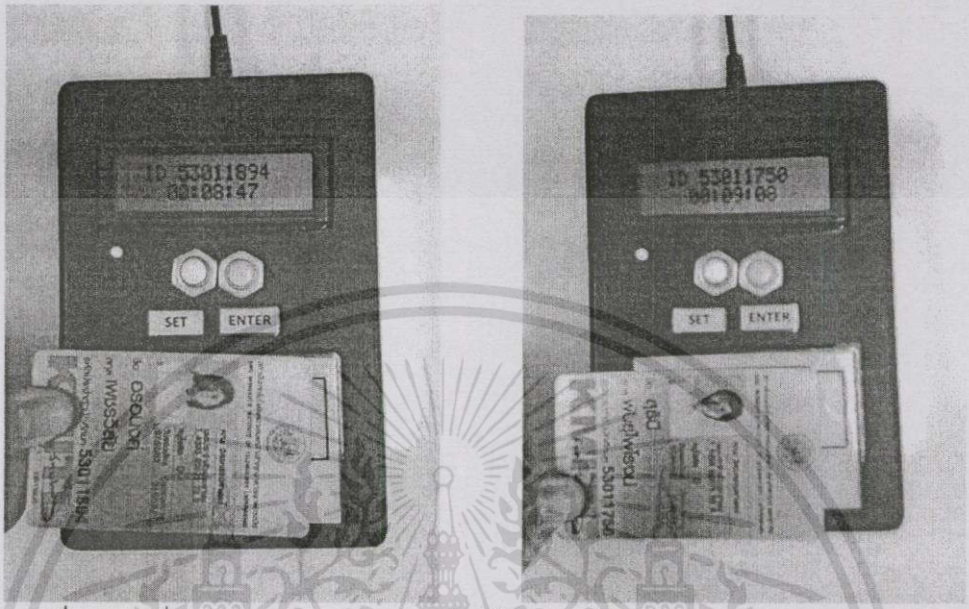
(ก) เมื่อทำการเปิดเครื่องบันทึกเวลาแบบ RFID หน้าจอแอลซีดีจะแสดง "Please Contact , DD/MM/YY and hr : min" เพื่อรอการแตะบัตรนักศึกษา สำหรับการเก็บข้อมูลลงใน SD card ดังรูปที่ 4.1



รูปที่ 4.1 เครื่องบันทึกเวลาแบบ RFID แสดงข้อความเพื่อรอการแตะบัตรนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(ข) เมื่อนำบัตรนักศึกษามาทำการแตะในบริเวณที่แตะการ์ด หน้าจอแอลซีดีจะแสดง ID ของบัตรนักศึกษาและเวลา ณ ที่ทำการแตะบัตร ตามรูปที่ 4.2



รูปที่ 4.2 เครื่องบันทึกเวลาแบบ RFID อ่านข้อมูลจากบัตร RFID และแสดง ID ของบัตร พร้อมเวลาที่ทำการแตะบัตร

4.3 การทดลองและผลการทดลองเก็บไฟล์ใน SD card

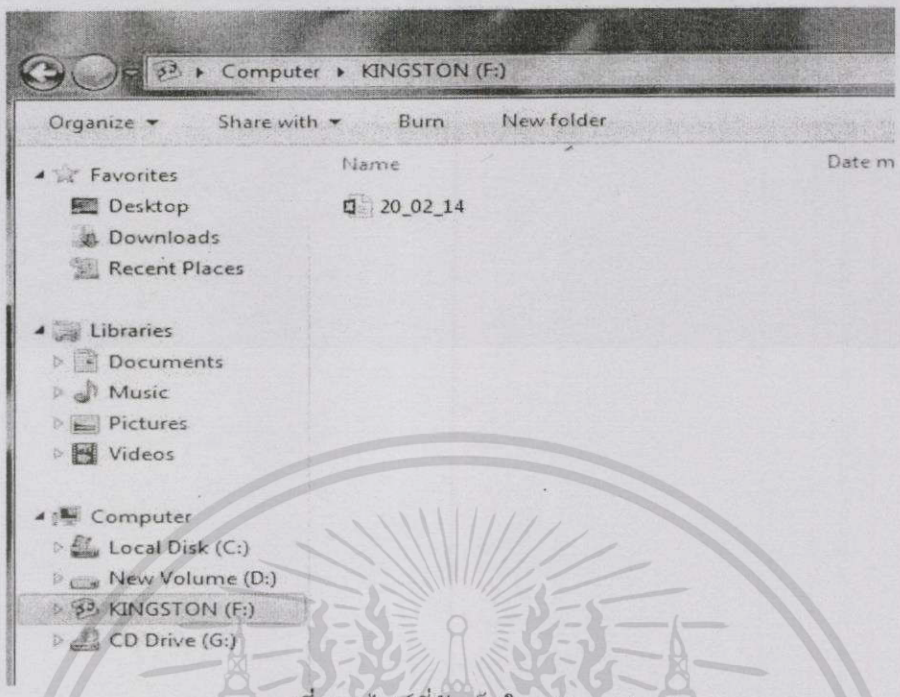
4.3.1 การทดลองเก็บไฟล์ใน SD card

การทดลองนี้เป็นการทดลองเก็บไฟล์ลงใน SD card ในรูปแบบไฟล์ .csv ซึ่งเป็นไฟล์ที่สามารถนำไปใช้กับโปรแกรม Microsoft Excel ได้

4.3.2 ผลการทดลองเก็บไฟล์ใน SD card

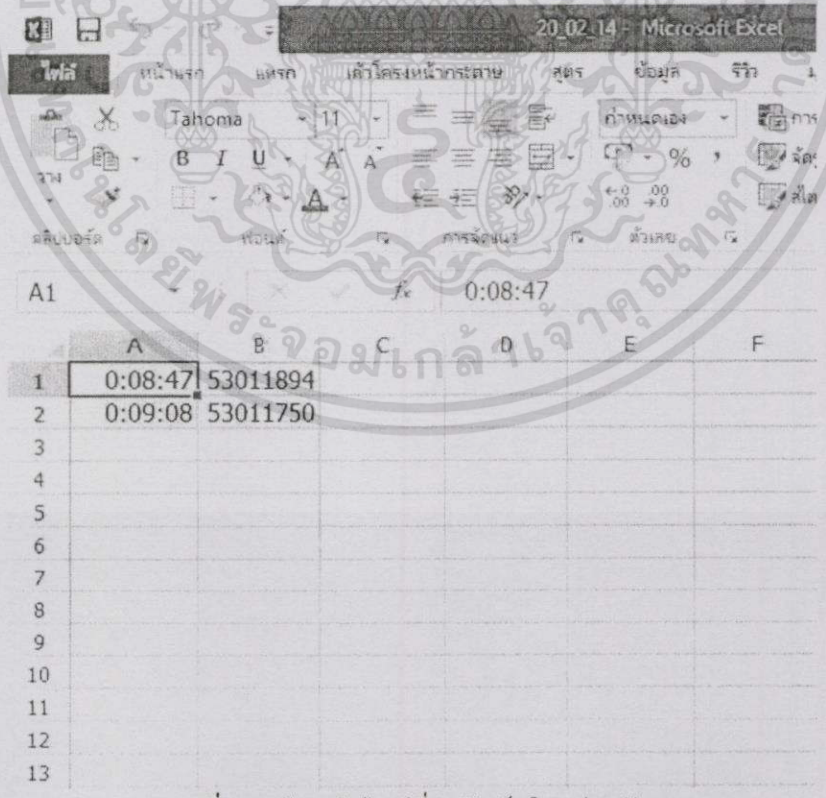
(ก) เมื่อมีบัตรนักศึกษามาทำการแตะบัตรกับเครื่องบันทึกเวลาแบบ RFID นั้น หน้าจอแอลซีดีจะแสดง ID ของบัตรนักศึกษาและเวลา ณ ที่ทำการแตะบัตร ซึ่งข้อมูลที่แสดงบนจอแอลซีดีจะถูกเก็บลงใน SD card ในรูปแบบไฟล์.csv โดยชื่อไฟล์ที่ถูกจัดเก็บใน SD card จะเป็นวันที่ ที่ได้ทำการแตะบัตร ดังรูปที่ 4.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 ไฟล์ที่จัดเก็บใน SD card

(ข) ข้อมูลในไฟล์ที่ถูกจัดเก็บในแต่ละวันจะมีเวลาขณะที่ทำการแตะบัตรและ ID ของบัตรนักศึกษา ดังรูปที่ 4.4



รูปที่ 4.4 ข้อมูลในไฟล์ที่ถูกจัดเก็บในแต่ละวัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 การทดลองและผลการทดลองของการตั้งค่าวันและเวลา

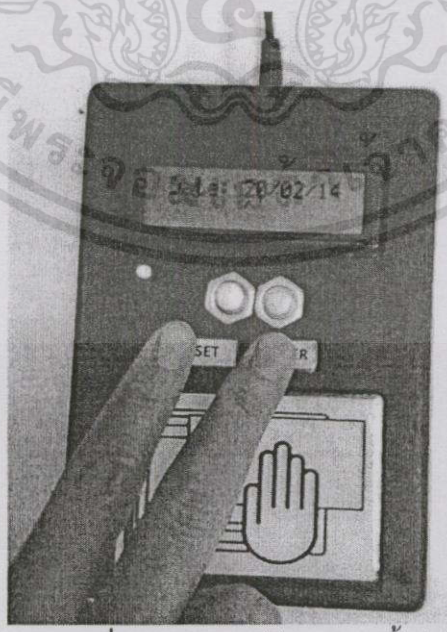
4.4.1 การทดลองการตั้งค่าวันและเวลา

การทดลองนี้เป็นการทดลองของการตั้งค่าวันและเวลาของเครื่องบันทึกเวลาแบบ RFID ซึ่งวันและเวลานี้จะแสดงบนหน้าจอแอลซีดี

4.4.2 ผลการทดลองการตั้งค่าวันและเวลา

(ก) กดปุ่ม SET และปุ่ม ENTER พร้อมกันค้างไว้เป็นเวลา 3 วินาที จะมีเสียงดังขึ้นและหน้าจอแอลซีดีแสดงการตั้งค่าวัน ดังรูปที่ 4.5 และ 4.6 ตามลำดับ

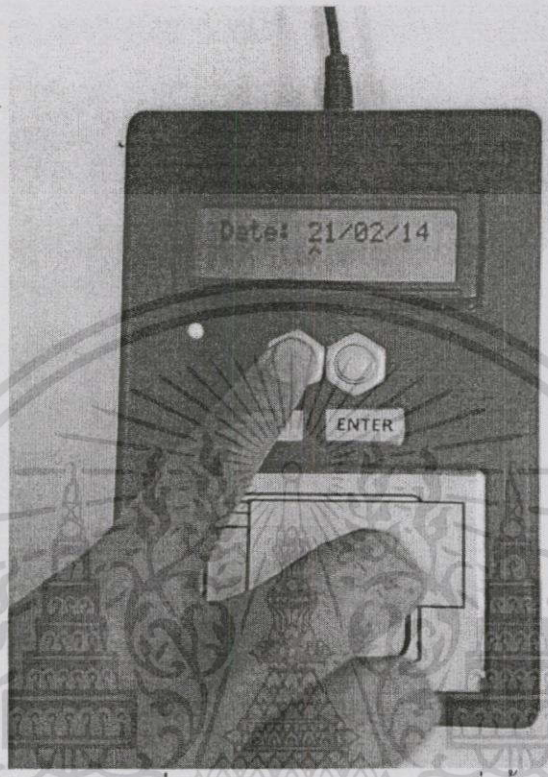
รูปที่ 4.5 กดปุ่ม SET และปุ่ม ENTER พร้อมกัน



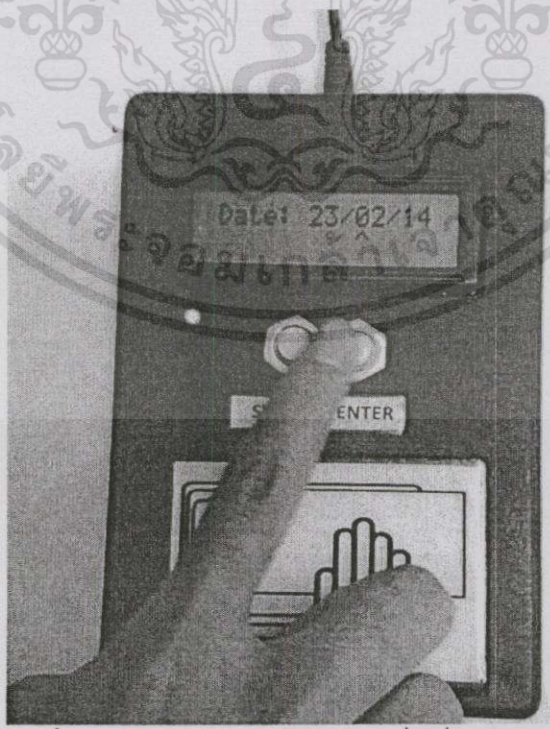
รูปที่ 4.6 จอแอลซีดีแสดงการตั้งค่าวัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(ข) กดปุ่ม SET เพื่อตั้งค่าวันที่และกดปุ่ม ENTER เพื่อเลื่อนไปยังการตั้งค่าเดือน ตามรูปที่ 4.7 และ 4.8 ตามลำดับ



รูปที่ 4.7 แสดงการกดปุ่ม SET เพื่อตั้งค่าวันที่



รูปที่ 4.8 แสดงการกดปุ่ม ENTER เพื่อเลื่อนไปยังการตั้งค่าเดือน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

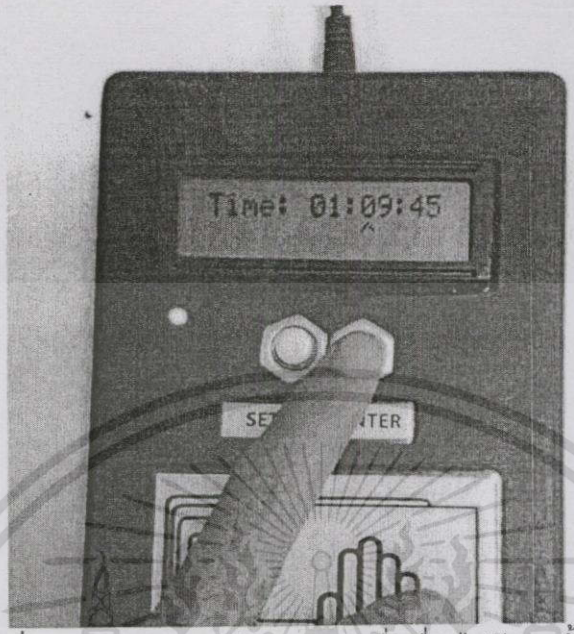
(ค) การตั้งค่าวันที่ เดือนและปี จะทำเหมือนกันในข้อ *(ข)* เมื่อทำการตั้งค่าวันเสร็จแล้ว กดปุ่ม ENTER เพื่อไปยังการตั้งค่าเวลา ดังรูปที่ 4.9

รูปที่ 4.9 จอแอลซีดีแสดงการตั้งค่าเวลา

(ง) กดปุ่ม SET เพื่อตั้งค่าชั่วโมงและกดปุ่ม ENET เพื่อเลื่อนไปยังการตั้งค่านาที ดังรูปที่ 4.10 และ 4.11 ตามลำดับ

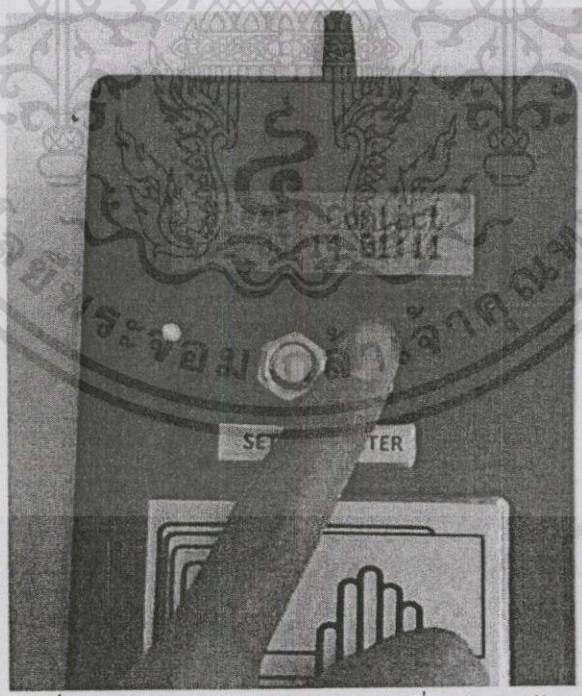
รูปที่ 4.10 แสดงการกดปุ่ม SET เพื่อตั้งค่าชั่วโมง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 แสดงการกดปุ่ม ENTER เพื่อเลื่อนไปยังการตั้งค่านาที

(จ) การตั้งค่าชั่วโมง นาทีและวินาที จะทำเหมือนกันในข้อ *(ง)* และเมื่อทำการตั้งค่าเวลาเสร็จแล้ว กดปุ่ม ENTER เพื่อกลับสู่หน้าจอปกติ ดังรูปที่ 4.12



รูปที่ 4.12 แสดงการกดปุ่ม ENTER เพื่อกลับสู่หน้าจอปกติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผล ปัญหาและข้อเสนอแนะ

5.1 สรุปผล

จากการทดลองของเครื่องบันทึกเวลาแบบ RFID เมื่อนำบัตรนักศึกษามาแตะที่เครื่องบันทึกเวลาแบบ RFID ข้อมูลถูกจัดเก็บใน SD card และสามารถตั้งค่าวัน เวลาได้ ผลที่ได้เป็นไปตามวัตถุประสงค์ กล่าวคือ เมื่อทำการแตะบัตรนักศึกษา เครื่องบันทึกเวลาแบบ RFID สามารถแสดง ID ของบัตรนักศึกษาพร้อมเวลาที่ทำการแตะบัตร และสามารถจักเก็บไฟล์ลงใน SD card ได้ ซึ่งไฟล์ที่จัดเก็บ SD card สามารถนำไปใช้งานกับคอมพิวเตอร์ได้ทันที และสามารถตั้งค่าวันและเวลาให้กับเครื่องบันทึกเวลาแบบ RFID ได้

5.2 ปัญหาและแนวทางแก้ไข

ปัญหาในการทดลองและแนวทางแก้ไขมีดังนี้

- การเขียนโปรแกรมในการสร้างชุดคำสั่งเพื่ออ่านข้อมูลจากบัตรนักศึกษา ควรศึกษา Manual ของ RFID Module อย่างละเอียด เนื่องจากชุดคำสั่งแต่ละคำสั่งมีเงื่อนไขที่แตกต่างออกไป เพราะถ้าส่งชุดคำสั่งผิดข้อมูลที่ได้รับกลับมาก็จะเกิดการผิดพลาด
- การเชื่อมต่อระหว่าง RFID Module กับไมโครคอนโทรลเลอร์ PIC18F458 เป็นการเชื่อมต่อแบบ UART(Universal Asynchronous Receiver Transmitter) ควรใช้แรงดันไฟเลี้ยงของอุปกรณ์ที่จำนำมาเชื่อมต่อให้เท่ากัน เนื่องจาก RFID Module ใช้แหล่งจ่ายแรงดันที่ 5V ดังนั้น PIC18F458 จึงต้องใช้แรงดันที่ 5V การรับ-ส่งข้อมูลจึงจะไม่ผิดพลาด
- มีการเปลี่ยนบอร์ด SD card ของ ET-MINI SPI SD/ MicroSD เป็น ALFAT SD card Board เนื่องจากบอร์ด ET-MINI SPI SD/ MicroSD ได้มีการตัดแปลงวงจรบางส่วนและยากต่อการเขียนโปรแกรม จึงทำให้เกิดความล่าช้าในการดำเนินงาน
- ในการทดลองของเครื่องบันทึกเวลาแบบ RFID นั้นจะต้องทำการเสียบ SD card ให้กับตัวเครื่องก่อนจะจ่ายไฟให้กับเครื่องบันทึกเวลาแบบ RFID เนื่องจากได้เขียนโปรแกรมให้มีการตรวจสอบการเชื่อมต่อของ SD card ก่อน มิเช่นนั้นจะทำให้ข้อมูลไม่ถูกบันทึกลงใน SD card

5.3 ข้อเสนอแนะ

การสื่อสารโดยใช้เทคโนโลยีในรูปแบบของ RFID นั้นยังสามารถนำไปประยุกต์ใช้กับเทคโนโลยีอื่นๆได้อย่างแพร่หลาย ทำให้มีความสะดวก และรวดเร็วต่อการใช้งาน เนื่องจากคุณสมบัติที่สำคัญของ RFID นั้นคือสามารถทนกับสภาพอากาศที่เปียกชื้นได้ มีความปลอดภัยสูง เก็บข้อมูลได้เยอะ อีกทั้งเป็นเทคโนโลยีไร้สายที่ใช้คลื่นความถี่วิทยุในการสื่อสาร จึงเป็นที่นิยมในการนำไปใช้งาน

การจัดเก็บไฟล์ SD card ที่เป็นรูปแบบไฟล์ .csv ซึ่งทำให้สามารถใช้งานกับคอมพิวเตอร์ได้ในโปรแกรม Microsoft Excel ทำให้สะดวกรวดเร็ว และมีความถูกต้องของข้อมูลที่ถูกจัดเก็บ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

ประจิน พลังสันติกุล. 2551. **All about CCS C.** พิมพ์ครั้งที่ 1. กรุงเทพฯ : บริษัทแอพซอฟต์เทค จำกัด.

ผศ.ดร.ปิยะ โควินท์ทวีวัฒน์. 2552. **ระบบบ่งชี้ด้วยคลื่นความถี่วิทยุ**. ปทุมธานี : สำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ.

ณัฐพล สีนาก, ปฏิพัทธ์ แปลกโฉมฉิน และปียาวี อ้นโต. 2555. "ระบบตอบรับความพึงพอใจแบบหลายจุด." ปริญญานิพนธ์วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมการวัดคุม คณะวิศวกรรมศาสตร์, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.

สถาบันส่งเสริมความเป็นเลิศทางเทคโนโลยีอาร์เอฟไอดีแห่งประเทศไทย.[Online]. Available : www.rfid.or.th

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

Code Program

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#include <18F458.h>
#fuses HS,NOWDT,NOPROTECT,NOLVP
#use delay(clock=20000000)
#use rs232(baud=115200, xmit=PIN_C6, rcv=PIN_C7,stream=RFID)
#use rs232(baud=115200, xmit=PIN_A5,rcv=PIN_A4,stream=SD)
#use I2C(sda=PIN_C4, scl=PIN_C3,stream=i2c)

#define ADDR_DS1307      0xD0      // Address DS1307
#define  Buzzer          PIN_A0
#define  Tag_In          PIN_B0
#define  Bt_Enter        PIN_B4
#define  Bt_Set          PIN_B2

#define  OK              0
#define  Error           1
#define  Init            2
#define  Open            3
#define  Close           4
#define  Write           5

#include <lcd.c>
const unsigned char
Login_sector[12]={0xBA,0x0A,0x02,0x01,0xAA,0x49,0x4E,0x4E,0x45,0x54,0x98,0xD9};
const unsigned char Read_Block[5]={0xBA,0x03,0x03,0x04,0xBE};

unsigned char Combuff[40];
unsigned char ID[4];
unsigned char len = 0;
unsigned char ret = 0;

typedef struct {
  BYTE sec;
  BYTE min;
  BYTE hr;
  BYTE day;
  BYTE date;
  BYTE month;
  BYTE year;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}DS1307_RTC;
DS1307_RTC RTC;

void DS1307_Write(unsigned char ctl,unsigned char dat){
    i2c_start(i2c);
    i2c_write(i2c,ADDR_DS1307);
    i2c_write(i2c,ctl);
    i2c_write(i2c,dat);
    i2c_stop(i2c);
}

BYTE DS1307_Read(unsigned char ctl){
    BYTE dat;

    i2c_start(i2c);
    i2c_write(i2c,ADDR_DS1307);
    i2c_write(i2c,ctl);

    i2c_start(i2c);
    i2c_write(i2c,ADDR_DS1307+1);
    dat = i2c_read(i2c,0);
    i2c_stop(i2c,);
    return (dat);
}

void DS1307_WriteDate(void){
    DS1307_Write(0x04,RTC.date);
    DS1307_Write(0x05,RTC.month);
    DS1307_Write(0x06,RTC.year);
}

void DS1307_WriteTime(void){
    DS1307_Write(0x00,RTC.sec);
    DS1307_Write(0x01,RTC.min);
    DS1307_Write(0x02,RTC.hr);
}

void DS1307_ReadDate(void){
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   RTC.date  = DS1307_Read(0x04);
   RTC.month = DS1307_Read(0x05);
   RTC.year  = DS1307_Read(0x06);
}

void DS1307_ReadTime(void){
   RTC.sec = DS1307_Read(0x00);
   RTC.min = DS1307_Read(0x01);
   RTC.hr  = DS1307_Read(0x02);
}

void Beep(int16 time){
output_low(Buzzer);
delay_ms(time);
output_high(Buzzer);
}

unsigned char Count(unsigned char x){
  char y=0x00;

  y=x&0x0F;
  if(y>0x9){
    x=(x&0xF0)+0x10;
  }
  return x;
}

void Set_Date_Time(void){
char index =1;
  if(!input(Bt_Enter)&&!input(Bt_Set)){
    delay_ms(2000);
    if(!input(Bt_Enter)&&!input(Bt_Set)){
    Beep(1000);
    delay_ms(500);

    while(TRUE){
      delay_ms(200);
      if(!input(Bt_Enter))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    index++;

  if(index==1){
    lcd_gotoxy(1,2);
    lcd_putc("      ^       ");
    if(!input(Bt_Set)){
      RTC.date++;
      RTC.date=Count(RTC.date);
      if(RTC.date>0x31)
        RTC.date=0x01;
        }
  }else if(index==2){
    lcd_gotoxy(1,2);
    lcd_putc("          ^    ");
    if(!input(Bt_Set)){
      RTC.month++;
      RTC.month=Count(RTC.month);
      if(RTC.month>0x12)
        RTC.month=0x01;
        }
  }else if(index==3){
    lcd_gotoxy(1,2);
    lcd_putc("              ^ ");
    if(!input(Bt_Set)){
      RTC.year++;
      RTC.year=Count(RTC.year);
      if(RTC.year>0x99)
        RTC.year=0x00;
        }
  }else if(index==4){
    lcd_gotoxy(1,2);
    lcd_putc("      ^       ");
    if(!input(Bt_Set)){
      RTC.hr++;
      RTC.hr=Count(RTC.hr);
      if(RTC.hr>0x23)
        RTC.hr=0x00;
        }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      }else if(index==5){
        lcd_gotoxy(1,2);
        lcd_putc("          ^     ");
        if(!input(Bt_Set)){
          RTC.min++;
          RTC.min=Count(RTC.min);
          if(RTC.min>0x59)
            RTC.min=0x00;
            }
      }else if(index==6){
        lcd_gotoxy(1,2);
        lcd_putc("               ^ ");
        if(!input(Bt_Set)){
          RTC.sec++;
          RTC.sec=Count(RTC.sec);
          if(RTC.sec>0x59)
            RTC.sec=0x00;
            }
      }
      if(index==1||index==2 || index==3){
        lcd_gotoxy(1,1);
        printf(lcd_putc," Date: %2X/%2X/%2X ",RTC.date,RTC.month,RTC.year);
      }else{
        lcd_gotoxy(1,1);
        printf(lcd_putc," Time: %2X:%2X:%2X ",RTC.hr,RTC.min,RTC.sec);
        }

      if(index >=7){
        DS1307_WriteDate();
        DS1307_WriteTime();
        lcd_gotoxy(1,1);
        printf(lcd_putc," Please Contact ");
          return;
        }
      }
    }
  }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void SD_Command(char cmd){
 switch(cmd){
     case Init:
       fprintf(SD,"I M:%C",0x0A);
     break;

     case Close:
       fprintf(SD,"C 1%C",0x0A);
       delay_ms(500);
     break;

     case Open:
       fprintf(SD,"O 1A>M:%C%X_%X_%X.CSV%C",0x5C,RTC.date,RTC.month,RTC.year,0x0A);
       delay_ms(500);
     break;

     case Write:
       fprintf(SD,"W 1>12%C",0x0A);
       delay_ms(1500);

fprintf(SD,"%X:%X:%X,%X%X%X%X%C",RTC.hr,RTC.min,RTC.sec,ID[0],ID[1],ID[2],ID[3],0x0A);
       delay_ms(500);
     break;

     default :
     break;
   }
}

char Read_Card(void){
unsigned char i;
  for(i=0;i<12;i++)
     fprintf(RFID,"%C",Login_Sector[i]);
       delay_ms(10);
       len=0;
       memset(Combuff,0xFF,40);
       for(i=0;i<5;i++)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            fprintf(RFID,"%C",Read_Block[i]);
            delay_ms(50);
    if(Combuff[3]==0x00 ){
       memcpy(ID,&Combuff[7],sizeof(ID));
       memset(Combuff,0xFF,40);
       return Ok;}
         else
            return Error;
}

void Check_Card(void){
if(!input(Tag_In))
    if(Read_Card() == Ok){
       lcd_gotoxy(1,1);
       lcd_putc("                ");
       lcd_gotoxy(0,1);
       printf(lcd_putc,"  ID %2X%2X%2X%2X",ID[0],ID[1],ID[2],ID[3]);
       lcd_gotoxy(1,2);
       printf(lcd_putc,"    %2X:%2X:%2X    ",RTC.hr,RTC.min,RTC.sec);
       Beep(250);
       SD_Command(Open);
       SD_Command(Write);
       SD_Command(Close);
       lcd_gotoxy(1,1);
       printf(lcd_putc," Please Contact ");
    }
}
#INT_RDA
void RS232_isr(){
        ret = getc();
         Combuff[len]=ret;
         len++;
}

void main(){
    output_high(Tag_In);
    output_high(Buzzer);
    len=0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

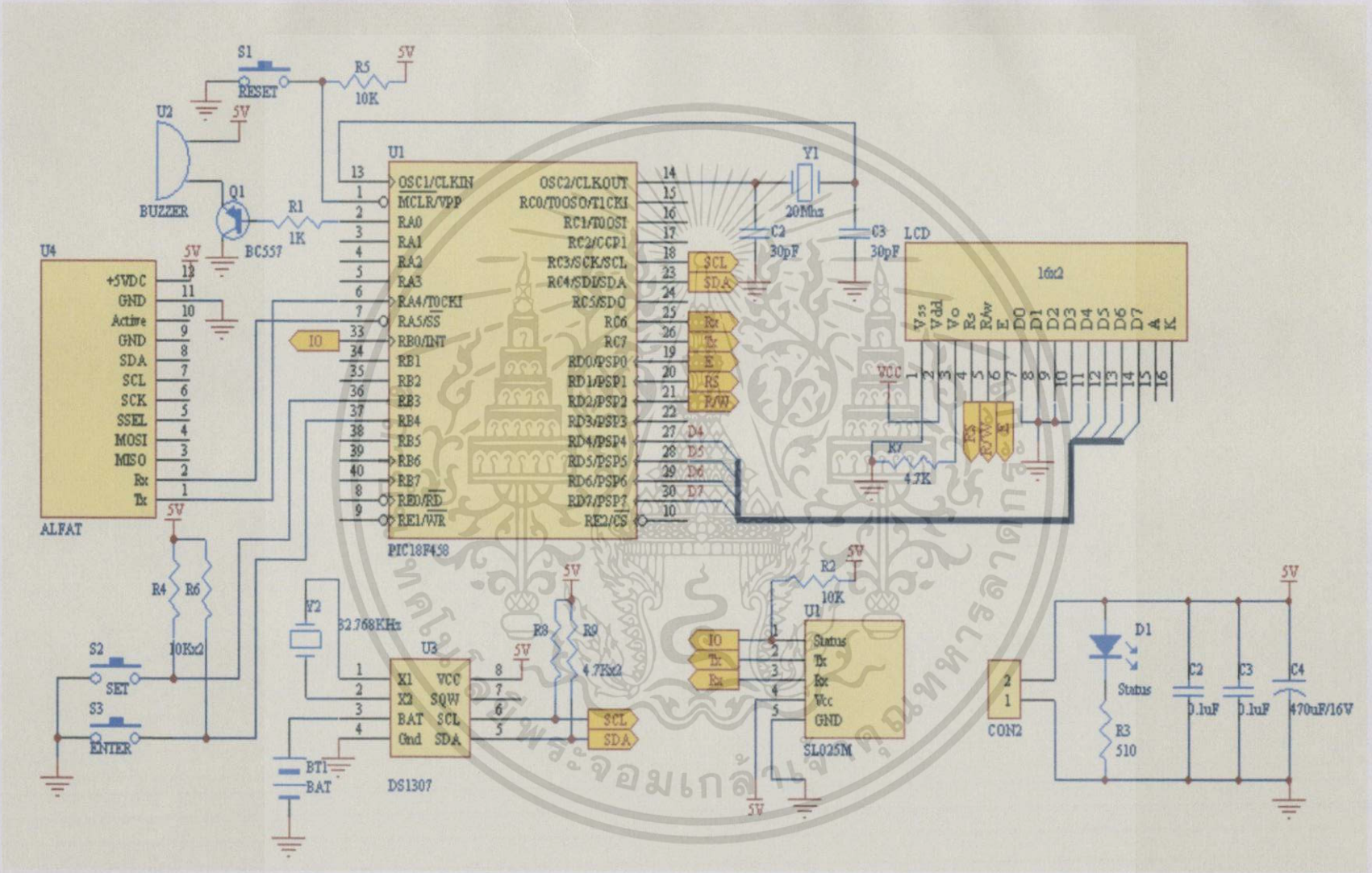
```
   memset(Combuff,0xFF,40);
   enable_interrupts(GLOBAL);
   enable_interrupts(INT_RDA);
   Beep(500);
   lcd_init();
   delay_ms(1500);
   SD_Command(Init);
   lcd_gotoxy(1,1);
   printf(lcd_putc," Please Contact ");

  while(TRUE){
      DS1307_ReadDate();
      DS1307_ReadTime();
      Set_Date_Time();
      lcd_gotoxy(1,2);
      printf(lcd_putc," %2X/%2X/%2X %2X",RTC.date,RTC.month,RTC.year,RTC.hr);
      lcd_putc(":");
      printf(lcd_putc,"%2X",RTC.min);
      delay_ms(200);
      Check_Card();
      delay_ms(200);
      Check_Card();
      lcd_gotoxy(13,2);
      lcd_putc(" ");
      delay_ms(200);
      Check_Card();
      delay_ms(200);
      Check_Card();
   }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

วงจรของเครื่องบันทึกเวลาแบบ RFID

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


S1
RESET
R5
10K
5V
U2
BUZZER
Q1
BC557
R1
1K
U1
OSC1/CLKIN
MCLR/VPP
RA0
RA1
RA2
RA3
RA4/T0CKI
RA5/SS
RB0/INT
RB1
RB2
RB3
RB4
RB5
RB6
RB7
RED/RD
RE1/WR
OSC2/CLKOUT
RC0/T0OSO/T1CKI
RC1/T0OSI
RC2/CCP1
RC3/SCK/SCL
RC4/SDI/SDA
RC5/SDO
RC6
RC7
RD0/PSP0
RD1/PSP1
RD2/PSP2
RD3/PSP3
RD4/PSP4
RD5/PSP5
RD6/PSP6
RD7/PSP7
RE2/CS
PIC18F458
Y1
20Mhz
C2
30pF
C3
30pF
SCL
SDA
Rx
Tx
E
RS
R/W
D4
D5
D6
D7
LCD
16x2
Vss
Vdd
Vo
Rs
R/w
E
D0
D1
D2
D3
D4
D5
D6
D7
A
K
VCC
R7
4.7K
U4
+5VDC
GND
Active
GND
SDA
SCL
SCK
SSEL
MOSI
MISO
Rx
Tx
ALFAT
IO
R4
R6
10Kx2
S2
SET
S3
ENTER
X2
32.768KHz
U3
X1
X2
BAT
Gnd
VCC
SQW
SCL
SDA
DS1307
BT1
BAT
R8
R9
4.7Kx2
R2
10K
U1
IO
Tx
Rx
Status
Tx
Rx
Vcc
GND
SL025M
CON2
D1
Status
R3
510
C2
0.1uF
C3
0.1uF
C4
470uF/16V