สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบควบคุมการเปิด-ปิดประตูโดยใช้ไมโครคอนโทรลเลอร์

Automatic Door Control System Using Microcontroller

T042639

โดย

นางสาวนริสา ดิษผึ้ง

นายบุญชัย ตันติวุฒิพงศ์

เลขหมู่.....
เลขทะเบียน..... 42639
วัน, เดือน, ปี..... 5 มิ.ย. 2545

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

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมอิเล็กทรอนิกส์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบควบคุมการเปิด-ปิดประตูโดยใช้ไมโครคอนโทรลเลอร์

Automatic Door Control System Using Microcontroller

โดย

นางสาวนริสา ดิษผึ้ง เลขประจำตัว 40010369

นายบุญชัย ตันติวุฒิพงศ์ เลขประจำตัว 40010412

อาจารย์ที่ปรึกษา

ผศ.พลผดุง ผดุงกุล

ปริญญานิพนธ์สำหรับปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบการควบคุมเปิด-ปิดประตูโดยใช้ไมโครคอนโทรลเลอร์

Automatic Door Control System Using Microcontroller

นางสาวนริสา ดิษผึ้ง รหัสประจำตัว 40010369

นายบุญชัย ตันติวุฒเพงศ์ รหัสประจำตัว 40010412

โครงงานนี้ได้รับการตรวจแล้ว พร้อมที่จะทำการสอบได้

(ผศ.พลผดุง ผดุงกุล)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายงาน ปีการศึกษา 2543

ภาควิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบการควบคุมเปิด-ปิดประตูโดยใช้ไมโครคอนโทรลเลอร์

Automatic Door Control System Using Microcontroller

ผู้จัดทำ

1. นางสาวนริสา ดิษผึ้ง รหัสประจำตัว 40010369
2. นายบุญชัย ตันติวุฒิพงศ์ รหัสประจำตัว 40010412

ลงชื่อ ..... อาจารย์ที่ปรึกษา

(ผศ.พลผดุง ผดุงกุล)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบควบคุมการเปิด-ปิดประตูโดยใช้ไมโครคอนโทรลเลอร์

Automatic Door Control System Using Microcontroller

นางสาวนริสา ดิษผึ้ง
นายบุญชัย ตันติวุฒิพงศ์
ผศ.พลผดุง ผดุงกุล (อาจารย์ที่ปรึกษา)
ภาคเรียนที่ 2 ปีการศึกษา 2543

บทคัดย่อ

ระบบควบคุมการเปิด-ปิดประตูอัตโนมัติโดยใช้ไมโครคอนโทรลเลอร์ เป็นการนำไมโครคอนโทรเลอร์มาทำการควบคุมระบบการทำงานทั้งหมดโดยอาศัยไมโครคอนโทรลเลอร์จำนวน 2 ตัว ตัวหนึ่งทำหน้าที่เป็นตัวแม่เป็นส่วนที่จะทำการแสดงผลของเวลาในการเปิด-ปิดของประตูของแต่ละหน่วยย่อย ส่วนไมโครคอนโทรลเลอร์อีกตัวหนึ่งจะเป็นตัวลูกทำหน้าที่ในการขับมอเตอร์โดยอาศัยสัญญาณอินฟราเรดเป็นตัวตรวจสอบ และทำการบันทึกเวลาที่มีการเปิด-ปิดประตูลงในหน่วยความจำ ซึ่งโครงงานนี้จะช่วยอำนวยความสะดวกในการเปิด-ปิดประตูโดยอัตโนมัติและเวลาในการเปิด-ปิด หวังว่ารายงานฉบับนี้คงจะเป็นประโยชน์สำหรับผู้ที่สนใจ และหากรายงานฉบับนี้มีข้อผิดพลาดประการใดก็ขออภัยไว้ ณ ที่นี้ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบควบคุมการเปิด-ปิดประตูโดยใช้ไมโครคอนโทรลเลอร์

Automatic Door Control System Using Microcontroller

Miss Narisa Ditpueng

Mr. Boonchai Tuntiwuttipong

Asst.Prof Polphadung Phadungkul (Adviser)

2nd Semester , Education year 2000

Abstract

Automatic Door Control Sysem Using Microcontroller use microcontroller to control all system by using 2 microcontrollers. One , Central System Control , use for representation time of opened-closed door in each local system control. Another is Local Control System use to control the drive motor circuit and infrared signal to detect. It also to record time of opened-closed door in memory (RAM). This project be useful for open-close door automatically. Hope that this project to be helpful for someone who interesting to research. If this report have a mistake , we apologize in this time.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทที่ 1 บทนำ	1
บทที่ 2 หลักการและทฤษฎีที่ใช้ในการออกแบบ	3
2.1 คุณสมบัติของ LED (Light-Emitting Diode) อินฟราเรด (infrered)	3
2.2 หลักในการออกแบบตัวส่ง	4
2.2.1 Radient Intensity	4
2.2.2 Inverse Square Law	4
2.3 วงจรพื้นฐานในการขับ LED อินฟราเรด	5
2.4 การใช้งานไอซีไทม์เมอร์ 555 (555 timer)	6
2.4.1 ลักษณะโครงสร้างภายนอก	6
2.4.2 การนำเอาไอซี 555 ไปใช้งาน	7
2.5 ค่าผิดพลาดของเวลา	10
2.6 คุณสมบัติของไอซี SL 486 ปรีแอมป์อินฟราเรดรีโมตคอนโทรล	12
2.6.1 ลักษณะการทำงาน	13
2.6.2 การนำไปใช้งาน	15
2.7 วงจรเปรียบเทียบและชมิททริกเกอร์ (The comparator and Schmitt Trigger)	18
2.7.1วงจรเปรียบเทียบ (Comparator)	18
2.7.2 ชมิททริกเกอร์ (The Schmitt Trigger)	19
2.8 มูเล่ (Pulley)	22
บทที่ 3 พอร์ตสื่อสารข้อมูลแบบอนุกรมใน MCS-51	24
3.1 พอร์ตสื่อสารข้อมูลแบบอนุกรม	24
3.2 การสื่อสารระหว่างไมโครคอนโทรลเลอร์หลายตัว	26
3.3 รีจิสเตอร์ใช้งานเฉพาะ SCON	28
3.4 อัตราเร็วในการรับและส่งข้อมูล	30
3.4.1 เมื่อใช้ไทม์เมอร์ 1 เป็นตัวกำหนด baud rate	30
3.4.2 การใช้ไทม์เมอร์ 2 ในการกำหนด baud rate	32
3.5 การทำงานพอร์ตสื่อสารข้อมูลแบบอนุกรมโหมด 2 และ 3	35
3.6 การสร้างระบบเวลาจริง (Real Time Clock) ให้แก่ MCS-51	38
3.6.1 การสร้างฐานเวลาจริงให้แก่ MCS-51 โดยใช้ RTC (Real Time Clock)	38
3.6.2 โครงสร้างของ command byte	41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6.2.1 Burst Mode 41
3.6.2.2 Write Protect Command 42
3.6.2.3 Reset and Clock control 42
3.6.2.4 Data Input 42
3.6.2.5 Data Output 43
3.6.2.6 Clock / Calender 43
3.6.2.7 Clock Half Flag 43
3.6.2.8 AM-PM /12-24 mode 43
3.6.2.9 Write Protect 43
3.6.2.10 Clock/Calender Burst Mode command byte 43
3.6.2.11 Ram burst Mode command byte 43
บทที่ 4 การออกแบบวงจรที่ใช้งาน 45
4.1 หน่วยกลาง 45
4.2 หน่วยย่อย 46
4.2.1 การออกแบบภาคส่งสัญญาณอินฟราเรด 47
4.2.2 การออกแบบภาครับสัญญาณอินฟราเรด 49
4.2.2.1 วงจร SL486 49
4.2.2.2 วงจรเปรียบเทียบ 50
4.2.3 ไมโครคอนโทรลเลอร์ที่ใช้ควบคุมการขับมอเตอร์ 50
4.2.4 วงจรควบคุมการทำงานของมอเตอร์ 52
4.2.5 ไมโครสวิตช์ 54
4.2.6 ระบบสร้างฐานเวลาให้กับระบบดคยใช้ RTC 54
4.2.7 การออกแบบโปรโตคอล 54
4.2.7.1 HEADER PORTION 55
4.2.7.2 DATA PORTION 55
4.2.8 มาตรฐานการสื่อสารข้อมูล EIA RS-485 56
4.2.8.1 คุณลักษณะเฉพาะของตัวส่ง RS-485 57
4.2.8.2 คุณลักษณะเฉพาะของตัวรับ RS-485 57
4.2.8.3 คุณลักษณะเฉพาะของคู่สายสัญญาณ RS-485 57
4.2.8.4 คุณสมบัติของคู่ตัวรับ-ส่ง RS-485 57
4.2.9 การรับส่งข้อมูลอนุกรมของไมโครคอนโทรลเลอร์ 58
บทที่ 5 สรุปและวิจารณ์ผลการทดลอง 66

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
รูปที่ 1.1 แสดงโครงสร้างโครงข่ายของระบบควบคุมการเปิด-ปิดประตูอัตโนมัติ	2
รูปที่ 2.1 การเปล่งแสงของ LED แบบ GaAs และ AlGaAs	3
รูปที่ 2.2 แสดงขอบเขตของการแผ่รังสีอินฟราเรด	4
รูปที่ 2.3 แสดงขอบเขตของการแผ่รังสีอินฟราเรด	5
รูปที่ 2.4 วงจรการขับ LED อินฟราเรด	5
รูปที่ 2.5 โครงสร้างของไอซี 555 ไทม์เมอร์	6
รูปที่ 2.6 ขาของไอซีไทม์เมอร์ 555	7
รูปที่ 2.7 ไอซี 555 ต่อเป็นวงจรโมโนสเตเบิล	8
รูปที่ 2.8 ไอซี 555 เป็นวงจรอะสเตเบิล	9
รูปที่ 2.9 วงจรที่ปรับปรุงแล้วให้ Duty cycle น้อยกว่า 50 %	10
รูปที่ 2.10 ลักษณะการจัดขาของ SL 486	12
รูปที่ 2.11 โครงสร้างภายในของ SL 486	13
รูปที่ 2.12 การนำไปประยุกต์ ของ SL 486 ด้วยอุปกรณ์น้อยชิ้น	14
รูปที่ 2.13 การนำไปใช้งานของ SL 486 อย่างเต็มประสิทธิภาพ	14
รูปที่ 2.14 การนำ SL486 ไปใช้งานร่วมกับ ML920 Series	15
รูปที่ 2.15 การนำ SL486 ไปใช้งานร่วมกับ ML1910	15
รูปที่ 2.16 การนำ SL486 ไปใช้งานร่วมกับไมโครโปรเซสเซอร์	17
รูปที่ 2.17 (ก) comparator	19
รูปที่ 2.17 (ข) comparator transfer characteristic	19
รูปที่ 2.18 (ก) zero crossing detector	20
รูปที่ 2.18 (ข) input and output	20
รูปที่ 2.19 แสดง characteristic เมื่อ (ก) V1 เพิ่ม	20
รูปที่ 2.19 แสดง characteristic เมื่อ (ข) V1 ลด	20
รูปที่ 2.19 แสดง characteristic เมื่อ (ค) ผลทั้งหมด	20
รูปที่ 2.20 สัญญาณซึ่งมีสัญญาณรบกวนกับสัญญาณชมิททริกเกอร์	22
รูปที่ 2.21 พื้นฐานการทดความเร็วของมูเล่	23
รูปที่ 3.1 แสดงข้อมูลที่รับและส่งในการทำงานของพอร์สื่อสารข้อมูลแบบอนุกรมโหมด0	25
รูปที่ 3.2 แสดงข้อมูลที่รับและส่งในการทำงานของพอร์สื่อสารข้อมูลแบบอนุกรมโหมด1	25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.3แสดงข้อมูลที่รับและส่งในการทำงานของพอร์สื่อสารข้อมูลแบบอนุกรมโหมด2 และ3 26
รูปที่ 3.4 แสดงรีจิสเตอร์ใช้งานเฉพาะ SCON 28
รูปที่ 3.5 ไทม์เมอร์ 2 ในการเป็นตัวกำหนด baud rate 33
รูปที่ 3.6 แสดงวงจรการเชื่อมต่อพอร์ตสื่อสารข้อมูลแบบอนุกรมของ MCS-51 37
รูปที่ 3.7 แสดงลักษณะรูปร่างทั้งสองแบบของไอซีเบอร์ DS1202 39
รูปที่ 3.8 โครงสร้างภายในของ RTC DS1202 40
รูปที่ 3.9 แสดงโครงสร้างของ command byte 41
รูปที่ 4.1 โครงสร้างทั่วไปของระบบการสื่อสาร 45
รูปที่ 4.2 แสดงการใช้ไอซี 555 เป็นวงจรอะสเตเบิล 47
รูปที่ 4.3 แสดงวงจรภาคส่งที่สมบูรณ์ 48
รูปที่ 4.4 แสดงวงจรของ SL 486 49
รูปที่ 4.5 วงจรเปรียบเทียบ (Comparator) 50
รูปที่ 4.6 แสดงพัลส์ที่ออกจากไมโครคอนโทรลเลอร์เข้ามอเตอร์เปิด-ปิดประตูเร็ว 51
รูปที่ 4.7 แสดงพัลส์ที่ออกจากไมโครคอนโทรลเลอร์เข้ามอเตอร์เปิด-ปิดประตูช้า 51
รูปที่ 4.8 แสดงสัญญาณที่ออกมาจากขาคอลเลคเตอร์ของ TIP127 53
รูปที่ 4.9 แสดงสัญญาณที่ออกมาจากไอซีไทม์เมอร์ 555 ขา 3 53
รูปที่ 4.10 วงจรRTC 54
รูปที่ 4.11 ส่วนประกอบ 1 เฟรมของโปรโตคอล 54
รูปที่ 4.12 ส่วนประกอบของ HEADER PORTION 55
รูปที่ 4.13 ส่วนประกอบของ DATA PORTION 55
รูปที่ 4.14 ส่วนประกอบของ DATA PORTION 56
รูปที่ 4.15 แสดงการเชื่อมต่อการรับส่งข้อมูลแบบอนุกรม 58
รูปที่ 4.16 แสดงพารามิเตอร์ SCON ของหน่วยย่อยและหน่วยกลางเมื่อเริ่มต้นส่งข้อมูล 59
รูปที่ 4.17 แสดงพารามิเตอร์ SCON ของหน่วยกลางและหน่วยย่อยเมื่อส่งข้อความตอบรับ 60
รูปที่ 4.18 แสดงพารามิเตอร์ SCON ของหน่วยกลางและหน่วยย่อยเมื่อหน่วยกลาง
ส่งข้อความ ACK 60
รูปที่ 4.19 แสดงโปรโตคอลในการติดต่อสื่อสารระหว่างหน่วยกลางและหน่วยย่อย 62
รูปที่ 4.20 โฟลว์ชาร์ทแสดงการทำงานของหน่วยกลาง 63
รูปที่ 4.21โฟลว์ชาร์ทแสดงการทำงานของหน่วยย่อย 64
รูปที่ 4.22 โฟลว์ชาร์ทแสดงการอินเทอร์รัพท์ของหน่วยย่อยและหน่วยกลาง 65

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
ตารางที่ 2.1 แสดงคุณสมบัติของไอซี 555	11
ตารางที่ 2.2 แสดงคุณสมบัติทางไฟฟ้าของ SL 486	18
ตารางที่ 3.1 ค่าที่ต้องนำไปไว้ในรีจิสเตอร์ของไทม์เมอร์ 1 เมื่อใช้ baud rate ค่าต่างๆ	31
ตารางที่ 3.2 แสดงค่าที่ต้องโหลดไปไว้ในรีจิสเตอร์ RCAP2L และ RCAP2H ของไทม์เมอร์ 2 เพื่อให้ได้ค่า baud rate มาตรฐาน	34
ตารางที่ 3.3 แสดงข้อมูลการทำงานของรีจิสเตอร์และ RAM ใน DS1202	44
ตารางที่ 4.1 กระแสที่ใช้ในการขับมอเตอร์ที่ความเร็ว 2 ระดับ	52

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

เทคโนโลยีในปัจจุบันนับวันยิ่งพัฒนาไปมากยิ่งขึ้นดังนั้นจึงมีสิ่งที่จะอำนวยความสะดวกให้กับมนุษย์มากขึ้น เมื่อเราดูตามห้างสรรพสินค้าต่างๆก็จะพบว่าจะมีระบบอำนวยความสะดวกมากมายทั้งในส่วนของลิฟท์ บันไดเลื่อน และส่วนของประตูเปิด-ปิดอัตโนมัติตรงส่วนเข้าออก ซึ่งในโครงงานชิ้นนี้ได้ทำการออกแบบระบบควบคุมการเปิด-ปิดประตูอัตโนมัติซึ่งทำการควบคุมด้วยส่วนสำคัญ 2 หน่วยคือ หน่วยย่อยซึ่งเป็นส่วนที่ทำหน้าที่ในการควบคุมการทำงานในส่วนของระบบเปิด-ปิดของประตูอัตโนมัติโดยอาศัยการตรวจจับด้วยรังสีอินฟราเรด และทำการบันทึกเวลาที่มีการเปิด-ปิดประตูลงในหน่วยความจำอีกทีหนึ่งรวมถึงการตั้งเวลาในการเปิด-ปิดอีกด้วย และอีกหน่วยหนึ่งคือ หน่วยกลางทำหน้าที่ในการตรวจสอบตรวจสอบเวลาที่มีการเปิด-ปิดประตูของแต่ละหน่วยย่อยได้ โดยอาศัยการสื่อสารผ่านพอร์ตอนุกรม

1.1 หน่วยย่อย

ประกอบไปด้วยส่วนประกอบที่สำคัญ 3ส่วนใหญ่ๆคือ

1.1.1 ส่วนภาครับส่งอินฟราเรด ประกอบไปด้วยวงจรที่ทำหน้าที่ดังนี้คือ

- Transmitter ทำหน้าที่ส่งสัญญาณรังสีอินฟราเรด โดยการใช้ไอซีไทม์ 555 ต่อเป็นวงจรอะสเตเบิล (Astable) เป็นอุปกรณ์กำเนิดสัญญาณพัลส์ไปขับ LED
- Receiver ทำหน้าที่รับสัญญาณอินฟราเรดที่จะสะท้อนจากวัตถุเข้ามาจากการที่สัญญาณอินฟราเรดจากตัวส่งไปกระทบกับวัตถุแล้วสะท้อนมาเข้าตัวรับ
- Comparator เป็นส่วนที่ทำการเปรียบเทียบแรงดันโดยมีแรงดันอ้างอิงไว้เปรียบเทียบกับแรงดันที่มาจากตัวภาครับสัญญาณอินฟราเรดซึ่งจะให้เอาต์พุตเป็นสัญญาณระดับ 0 และ 5 โวลต์ซึ่งก็คือระดับของสัญญาณดิจิตอลนั่นเอง

1.1.2 Microcontroller ประกอบไปด้วยส่วนที่ทำหน้าที่ดังนี้

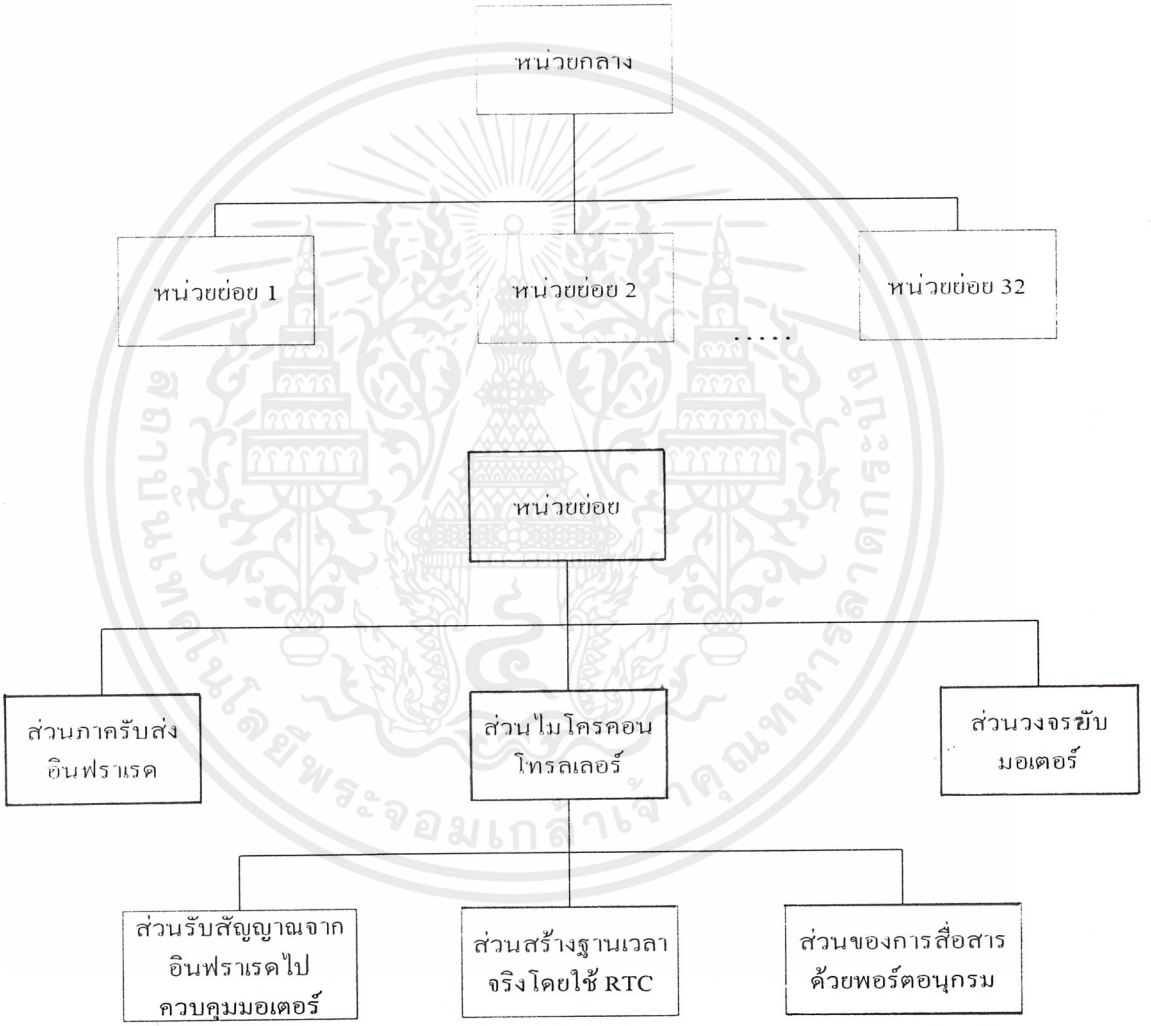
- ส่วนที่จะรับสัญญาณต่อมาจากส่วน Comparator เพื่อจะไปทำการควบคุมการเปิด-ปิดประตูด้วยวงจรขับ DC Motor
- ส่วนควบคุมเวลาโดยมีการสร้างฐานเวลาให้กับระบบโดยใช้ RTC โดยเมื่อประตูมีการเปิด-ปิด จะมีการแสดงเวลาดังกล่าวออกหน้าจอ LCD และทำการเก็บเวลาลงสู่หน่วยความจำในแต่ละส่วนย่อย และสามารถตั้งเวลาเปิด-ปิดประตูได้อีกด้วย
- ส่วนของพอร์ตสื่อสารอนุกรม ทำหน้าที่ในการเชื่อมต่อระบบกับหน่วยกลางเพื่อทำการส่งค่าเวลาที่มีการเปิด-ปิดประตูของตัวมันเองไปให้กับหน่วยกลางเมื่อหน่วยกลางต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.1.3 DC Motor เป็นส่วนที่ไปหมุนมูเล่เพื่อควบคุมการเปิด-ปิดของประตูโดยรับสัญญาณมาจากส่วนของไมโครคอนโทรลเลอร์

1.2 หน่วยกลาง

หน่วยกลางเป็นส่วนที่ทำหน้าที่ในการตรวจสอบเวลาที่มีการเปิด-ปิดประตูของแต่ละหน่วยย่อยโดยจะทำการส่งข้อความให้กับหน่วยย่อยที่ต้องการติดต่อด้วย โดยระบุตำแหน่งตัวลูกที่ต้องการติดต่อด้วยซึ่งทั้งหมดจะสื่อสารผ่านระบบที่เชื่อมต่อกันตามมาตรฐาน RS-485 ดังรูปที่ 1.1



รูปที่ 1.1 แสดงโครงสร้างโครงข่ายของระบบควบคุมการเปิด-ปิด ประตูอัตโนมัติ

ดังนั้นในโครงงานชิ้นนี้ได้ใช้หลักการเบื้องต้นที่ได้กล่าวไปข้างต้นมาทำการสร้างระบบควบคุมการเปิด-ปิดประตู ซึ่งในส่วนบทต่อๆไปจะได้กล่าวถึงในเนื้อหารายละเอียดตลอดจนหลักการออกแบบในแต่ละส่วนของวงจร การออกแบบในส่วนของทางกล รวมทั้งการเขียนโปรแกรมควบคุมการทำงานของวงจรซึ่งผู้สนใจสามารถจะศึกษาในรายละเอียดได้ต่อไป

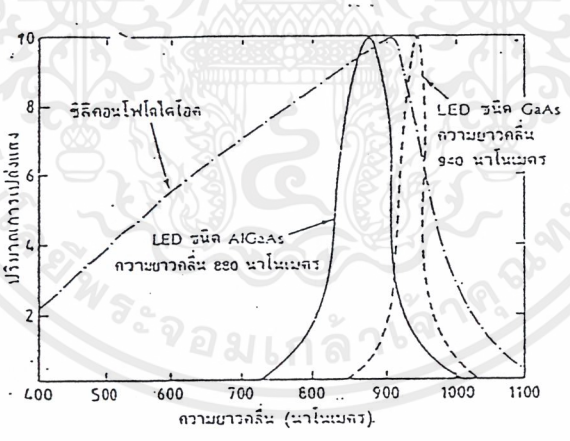
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
หลักการและทฤษฎีที่ใช้ในการออกแบบ

2.1 คุณสมบัติของ LED (Light-Emitting Diode) อินฟราเรด (Infrared)

แรงดันตกคร่อมที่รอยต่อ P-N (P-N junction) ของไดโอดต้องมีค่ามากกว่าแรงดันเทรสโฮลด์ (Threshold) จึงจะสามารถทำให้ไดโอดนำกระแสได้ สำหรับซิลิกอนไดโอด (Silicon Diode) แรงดันทำงานมีค่าประมาณ 0.6 โวลต์ (Volt) ส่วน LED ที่ให้แสงในย่านที่ตามองเห็นได้ ถ้าทำจากสาร GaP ซึ่งให้แสงสีเขียวจะมีค่าแรงดันประมาณ 2.1 ถึง 2.8 โวลต์ ถ้าเป็น LED ที่ทำจาก AlGaAs ให้แสงสีแดงมีแรงดันทำงาน 1.75 ถึง 2.5 โวลต์ ส่วน LED ที่ให้แสงใกล้ย่านอินฟราเรด ทำจากสาร GaAs มีแรงดันทำงาน 1.5 โวลต์โดยให้แสงที่มีความยาวคลื่น 940 นาโมเมตร (nanometer) และถ้าทำจาก AlGaAs จะให้แสงความยาวคลื่น 880 นาโมเมตร ที่แรงดัน 1.75 โวลต์

พลังงานที่ได้จากการเปล่งแสงของ LED หาได้จากกระแสไบอัสตรง (Forward Bias) ของไดโอดและต้องระมัดระวังไม่ให้กระแสส่วนนี้มีค่าสูงจนเกิดความร้อน อันจะทำอันตรายต่อชิ้นไดโอด สิ่งที่สำคัญที่สุดของ LED อินฟราเรดกำลังงานสูงคือชิ้นสาร AlGaAs ที่ให้ความยาวคลื่น 880 นาโนเมตร และสาร GaAs ซิลิกอนไดโอดที่ให้แสงความยาวคลื่น 940 นาโนเมตร



รูปที่ 2.1 การเปล่งแสงของ LED แบบ GaAs และ AlGaAs

ดังที่แสดงไว้ในรูปที่ 2.1 คือ กราฟสเปคตรัม (Spectrum) ที่เปรียบเทียบความยาวคลื่นกับการเปล่งแสงของ LED ทั้งสองชนิด ซิลิกอน LED ที่ทำจาก GaAs ให้กำลังงานประมาณ 5 มิลลิวัตต์ (mW) ที่กระแสไบอัสตรง 100 มิลลิแอมป์ (mA)

ข้อเสียของ LED ที่ให้ความยาวคลื่น 940 นาโนเมตร คือ สามารถถูกดูดกลืนโดยละอองน้ำ จึงไม่เหมาะกับการสื่อสารด้วยแสงภายนอก เพราะจุดอ่อนเรื่องการถูกดูดกลืนด้วยไอน้ำในอากาศนั่นเอง LED ชนิดซิลิกอนที่ทำจาก GaAs มักจะใช้เป็นแหล่งกำเนิดแสงย่านอินฟราเรด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 หลักในการออกแบบตัวส่ง

สิ่งที่ต้องคำนึงถึงในการออกแบบตัวส่งนั้นเราต้องใช้หลักการดังต่อไปนี้

2.2.1 Radient Intensity พิจารณาสมการ และรูปที่ 2.2

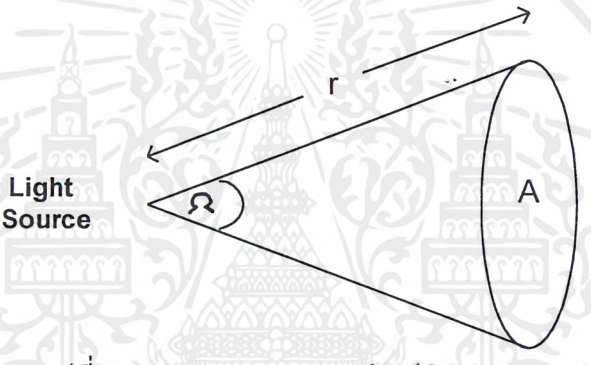
$$I_e = \frac{d\Phi}{d\Omega} = \frac{Watt}{Steradian} \quad (2.1)$$

$$\Phi_e = \int Ied\Omega \quad (2.2)$$

ถ้า I_e = Constant

$$\Phi_e = I_e \Omega \quad (2.3)$$

$$\Omega \text{ (Solid Angle)} = \frac{A}{r^2} \quad (2.4)$$



รูปที่ 2.2 แสดงขอบเขตการแผ่รังสีอินฟราเรด

ในกรณีนี้เราจะพิจารณาว่าตัวส่งเป็นเสมือนแหล่งกำเนิดแบบจุด(Point Source) ซึ่งจากสมการจะพบว่าค่า Intensity ของตัวส่งที่ปล่อยออกมาจะแปรผันตรงกับค่า Power ที่ตัวส่งได้รับ แต่แปรผกผันกับค่า Solid Angle ของตัวส่ง โดยถ้าพื้นที่ A มีค่าน้อย จะทำให้มุม Solid Angle ที่ของตัวส่งแคบ ค่า Intensity ก็จะสูง นั่นก็คือ ถ้าให้พลังงานแก่ตัวส่งที่เท่ากัน ตัวส่งที่ปล่อยพลังงานออกมาด้วยมุมแคบจะมีค่า Intensity ของพลังงานที่ปล่อยออกมา มากกว่าตัวส่งที่ปล่อยพลังงานออกมาด้วยมุมที่กว้าง

2.2.2 Inverse Square Law พิจารณาสมาการ และรูปที่ 2.3

$$\Omega = \frac{A}{r^2}\Omega_0 \quad (2.5)$$

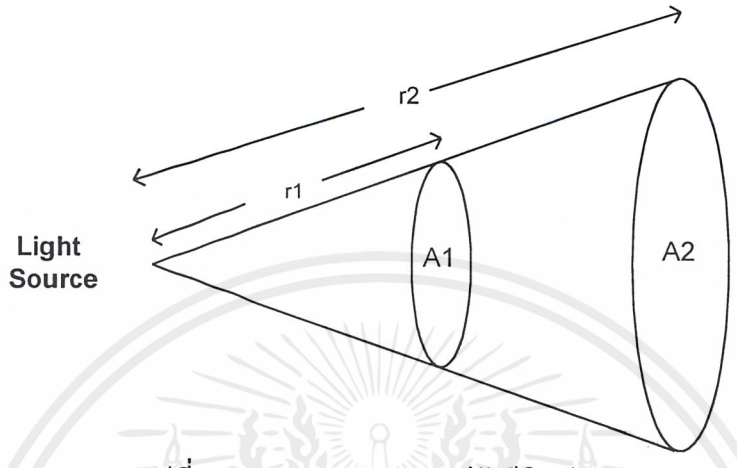
$$\Omega_0 = 1 \text{ Steradian} = \frac{1m^2}{(1m)^2} \quad (2.6)$$

$$E_1 = \frac{d\Phi_e}{dA_{e1}} \quad \text{และ} \quad E_2 = \frac{d\Phi e}{dA_{e2}} \quad (2.7)$$

เนื่องจากค่า Total Radiant Flux (Φ_e) ในแต่ละบริเวณมีค่าเท่าเดิม ดังนั้นจะได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\frac{E_2}{E_1} = \frac{A_{e1}}{A_{e2}} = \frac{r_1^2}{r_2^2} \quad (2.8)$$

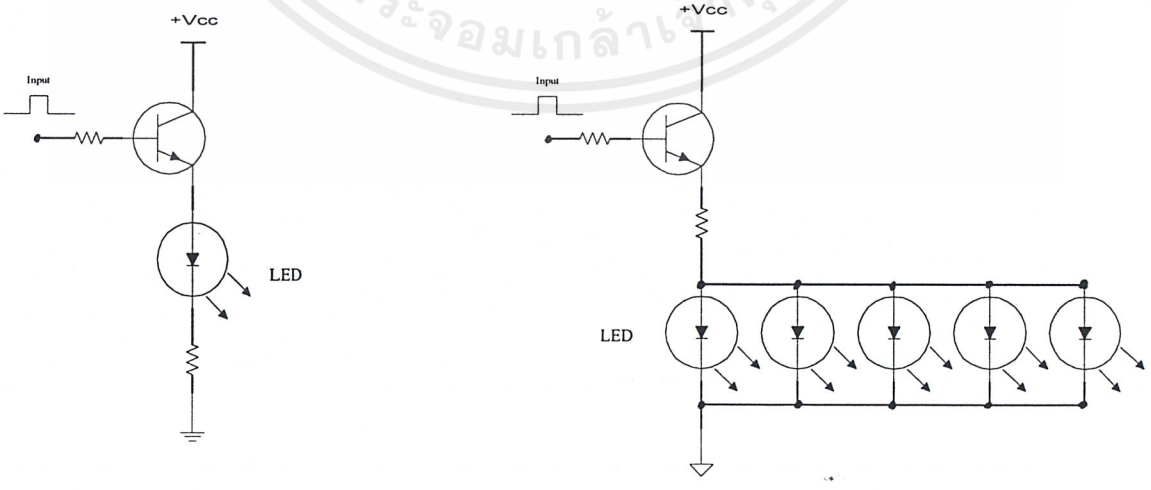


รูปที่ 2.3 แสดงขอบเขตการแผ่รังสีอินฟราเรด

จากรูปค่า Total Radiant Flux (หรือค่าพลังงานที่ตัวส่งปล่อยออกมา)ในแต่ละ Area คือ A1,A2,A3 มีค่าเท่ากัน แต่ค่าIntensity จะต่างกันเป็นสัดส่วนกับระยะทางยกกำลังสอง (r^2) โดยยิ่งไกลจากแหล่งกำเนิดแสงออกไปค่า Intensity จะยิ่งลดลงไปเรื่อยๆผกผันกับระยะทางยกกำลังสอง (r^2)

2.3 วงจรพื้นฐานในการขับ LED อินฟราเรด

เมื่อได้พิจารณาคุณสมบัติของ LED ผ่านมาแล้วในส่วนนี้เราจะมาศึกษาถึงวิธีการในการขับ LED อินฟราเรดให้สามารถส่งสัญญาณได้อย่างมีประสิทธิภาพ โดยทั่วไปจะมีวงจรพื้นฐานในการขับ LED อินฟราเรดอยู่ 2 วงจรก็คือ วงจรขับ LED แบบอนุกรมและแบบขนานดังรูปที่ 2.4



รูปที่ 2.4 วงจรการขับ LED อินฟราเรด (ก) แบบอนุกรม (ข) แบบขนาน

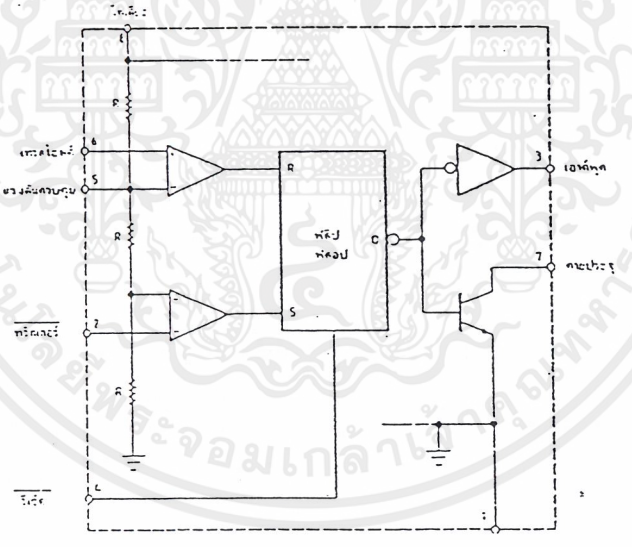
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การจะเลือกใช้งานก็ขึ้นอยู่กับว่า ต้องการออกแบบให้วงจรเป็นอย่างไรซึ่งแต่ละวงจรก็จะมีข้อดีและข้อเสียแตกต่างกันเช่นในรูปที่ 2.4(ก) ก็พบว่ามีข้อดีที่ดึงกระแสไม่มากนักเมื่อเทียบกับรูป (ข) เพราะ LED หลายตัวใช้กระแสตัวเดียวกันในการไดร์ฟ แต่มีข้อเสียตรงที่หาก LED ตัวใดตัวหนึ่งขาดวงจรนั้นก็จะไม่ทำงานทันทีและถ้าจะต่อ LED หลายตัวก็ต้องใช้ไฟเลี้ยงสูงขึ้นเพราะต้องมีศักดาไปตกคร่อม LED แต่ละตัว ส่วนในรูปที่ 2.4(ข) เป็นการต่อแบบขนานจึงมีข้อดีตรงใช้ไฟเลี้ยงไม่สูงมากนักเพราะ LED ที่ต่อขนานหลายๆตัวจะมีศักดาตกคร่อมเท่ากัน และหาก LED ตัวใดตัวหนึ่งขาดวงจรก็ยังคงใช้งานได้อยู่แต่มีข้อเสียตรงที่วงจรดึงกระแสค่อนข้างมาก

2.4 การใช้งานไอซีไทเมอร์ 555 (555 timer)

2.4.1 ลักษณะโครงสร้างภายนอก

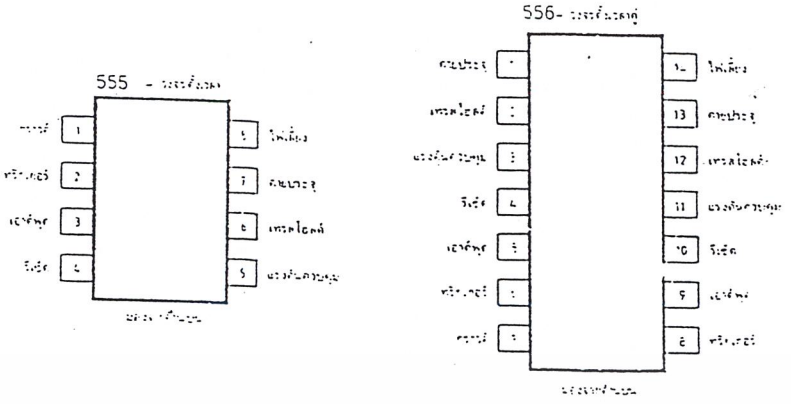
รูปที่ 2.5 แสดงโครงสร้างภายในของตัวไอซีไทเมอร์ 555 หรือ ไอซี 555 จะเห็นว่าประกอบด้วยวงจรภายใน ทั้งที่เป็นดิจิตอล (Digital) (วงจรไบสเตเบิล (Bi-stable)) และวงจรอนาลอก (Analog) (วงจรเปรียบเทียบ) นอกจากนั้นยังมีอุปกรณ์อื่นๆ ร่วมด้วยอีกมากมาย



รูปที่ 2.5 โครงสร้างของไอซี 555 ไทเมอร์

เมื่อเราพิจารณาโครงสร้างของไอซีจะเห็นว่ามีความสามารถจ่ายกระแสได้สูงและสามารถใช้ได้กับไฟเลี้ยงในช่วงกว้างๆ รายละเอียดของการจัดขาของตัวไอซี 555 และ ไอซี 556 ได้แสดงไว้ในรูปที่ 2.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 ขาของไอซีไทเมอร์ 555

ลักษณะภายนอกของไอซี 555 จะมีขาทั้งหมด 8 ขา ดังนี้คือ

- ขากราวด์ (Ground) เป็นจุดต่อกราวด์ของวงจรภายในและภายนอก

- ขาจุดชนวน (Trigger) ทำหน้าที่จุดชนวนให้ไอซี 555 อยู่ในสภาวะ "ON" เมื่อมีสัญญาณจุดชนวนแรงดันลบที่มีขนาดน้อยกว่า 1 ใน 3 ของ V_{CC}

- ขาเอาต์พุต (Output) เป็นเอาต์พุตของไอซี 555 ที่ต่อไปยังโหลด (Load) ภายนอก มี 2 สภาวะคือ สภาวะ "ON" (1) และสภาวะ "OFF" (0)

- ขารีเซ็ต (Reset) ทำหน้าที่เปลี่ยนสภาวะปัจจุบันของไอซี 555 ให้เป็นสภาวะ "OFF" เมื่อมีสัญญาณเข้าที่ขารีเซ็ต

- ขาแรงดันควบคุม (Control Voltage) จะต่อกับแหล่งจ่ายแรงดันภายนอก เพื่อควบคุมไอซี 555 ให้ทำงานที่ค่าแรงดันมากกว่า 1 ใน 3 ของ V_{CC} และ 2 ใน 3 ของ V_{CC}

- ขาแรงดันขีดเริ่มเปลี่ยน (Threshold Voltage) ถ้าแรงดันที่ขานี้มีค่ามากกว่า 2 ใน 3 ของ V_{CC} จะทำให้ไอซี 555 อยู่ในสภาวะ "OFF" ถ้าแรงดันที่ขานี้มีค่าน้อยกว่า 2 ใน 3 ชอง V_{CC} จะทำให้ไอซี 555 อยู่ในสภาวะ "ON"

- ขาคายประจุ (Discharge) มักจะต่อกับตัวเก็บประจุภายนอก เพื่อให้ตัวเก็บประจุคายประจุผ่านทรานซิสเตอร์ (Transistor) Q_1 ขณะที่ Q_1 อยู่ในสภาวะ "ON"

2.4.2 การนำเอาไอซี 555 ไปใช้งาน

ไอซี 555 ช่วงเวลาของพัลส์ (Pulse) จะถูกกำหนดด้วยเวลาที่ใช้ในการประจุตัวเก็บประจุจาก 0 โวลต์ ถึง 2/3 ของแรงดันไฟเลี้ยง (แรงดันเทรสโฮลด์) โดยทั่วไปแล้วแรงดันประจุ C (V_C) จะถูกประจุผ่าน R จากแหล่งจ่ายไฟ V_{CC} จนมีแรงดันเท่ากับ 2/3 V_{CC}

เมื่อ

$$V_C(t) = V_{CC}\left(1 - e^{-\tau / Rc}\right) \qquad (2.9)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่

$$\tau = (-\log e^{1/3}) R_C$$

$$\tau = 1.1 R_C \quad (2.10)$$

แรงดันที่ถูกประจุนี้จะเป็นตัวกำหนดช่วงเวลาเดี่ยว (monotime) และพัลส์ที่มากระตุ้นก็จะสั้นกว่าช่วงเวลาเดี่ยวถ้าพัลส์ที่มากระตุ้นยิ่งกว้างจะส่งผลให้ช่วงเวลาเดี่ยวกว้างขึ้นด้วยแต่สามารถแก้ไขได้โดยการใช้วิธี การส่งผ่านแบบสัญญาณไฟสลับ (โดยการเพิ่ม R_2, C_3 และให้ $R_2.C_3 < R_1.C_1$) เป็นวงจรโมโนสเตเบิล (mono-stable) ตามรูปที่ 2.7

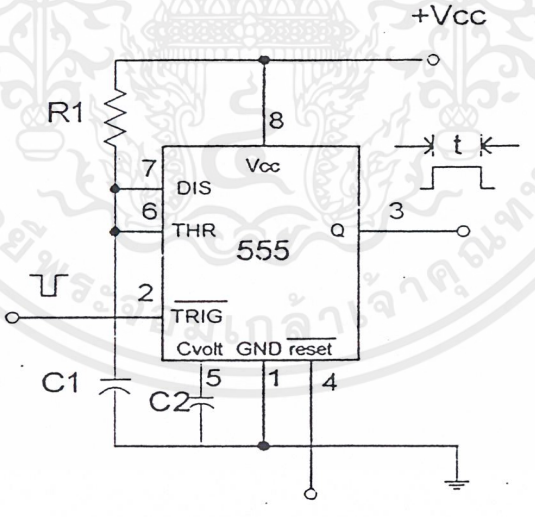
และถ้าต้องการเปลี่ยน วงจรโมโนสเตเบิลไปเป็นวงจรอะสเตเบิล ก็เพียงแต่ทำให้ตัวไอซีสามารถทำการกระตุ้นตัวเองได้ ตัวเก็บประจุ C_1 จะถูกประจุโดย R_1 และ R_2 จนมีแรงดันเท่ากับ 2/3 V_{CC} ในช่วงเวลา t_1

$$t_1 = (-\log^{1/3})(R_1+R_2)C - (-\log e^{2/3})(R_1+R_2)C$$

$$= 0.693 (R_1+R_2) C \quad (2.11)$$

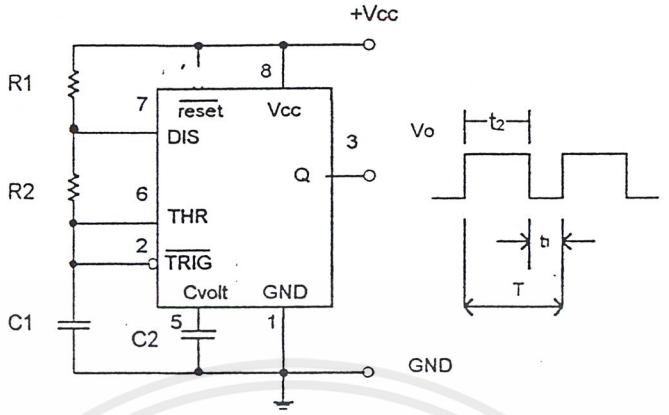
และเมื่อถึงช่วงเวลาของการคายประจุ การคายประจุของ C จะผ่าน R_2 เท่านั้น ดังนั้น ช่วงเวลาของการคายประจุจะเท่ากับ

$$t_2 = 0.693 R_2C \quad (2.12)$$



รูปที่ 2.7 ไอซี 555 ต่อเป็นวงจรโมโนสเตเบิล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 ไอซี 555 เป็นวงจรอะสเตเบิล

ซึ่งก็หมายความว่า แรงดันบนตัวเก็บประจุจะเปลี่ยนไปมาระหว่าง 1/3 V_{CC} ถึง 2/3 V_{CC} ช่วงเวลาทั้งหมด คำนวณได้จาก

$$T = t_1 + t_2$$
$$= 0.693 (R_1 + 2R_2) C \quad (2.13)$$

ค่าความดันถี่เอาต์พุต (f_0) เท่ากับ

$$f_0 = 1/T$$
$$= 1.44 (R_1 + 2R_2) C \quad (2.14)$$

อย่าลืมว่า เมื่อเราจ่ายไฟเลี้ยงให้แก่วงจรครั้งแรก หรือเมื่อให้อินพุตที่รีเซ็ตเป็นระดับสูง C จะเริ่มประจุจาก 0 โวลต์ ดังนั้นช่วงเวลาของพัลส์ลูกแรกจะเท่ากับ $1.1R_1C_1$

จุดเด่นอีกประการหนึ่งของไอซี 555 เมื่อนำมาทำเป็นวงจรโมโนสเตเบิล หรือ อะสเตเบิล ก็คือ ช่วงเวลาของพัลส์จะไม่ขึ้นอยู่กับแรงดันของแหล่งจ่ายไฟเลี้ยง V_{CC} ซึ่งเมื่อแรงดันของแหล่งจ่ายไฟลดลง จะทำให้แรงดันทริกเกอร์ (Trigger) และแรงดันเทรสโฮลด์ลดลงตามมาด้วย ดังนั้นช่วงเวลาของพัลส์ที่เอาต์พุตก็จะไม่เปลี่ยนไปตามแรงดันของแหล่งจ่ายไฟเลี้ยง

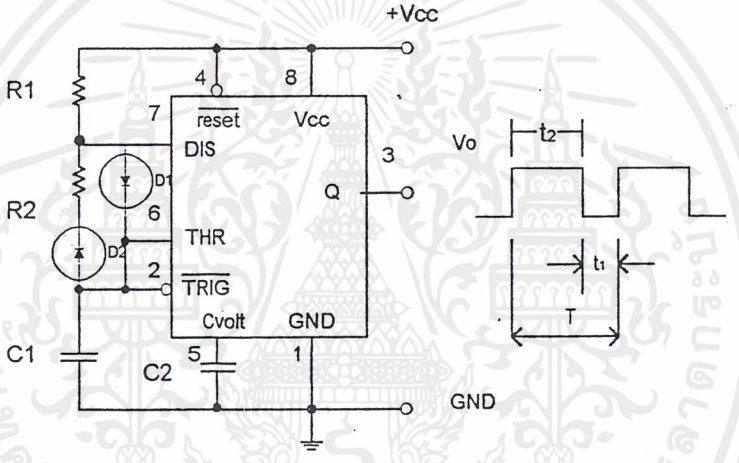
สำหรับวงจรอสะเตเบิล ตัวไอซีเองไม่สามารถจ่ายเอาต์พุตที่มีค่าดิวตี้ไซเคิล(duty cycle) ต่ำกว่า 0.5 ได้ เนื่องจากค่าความต้านทานในการประจุเท่ากับ $R_1 + R_2$ จะมีค่ามากกว่าความต้านทานของการคายประจุ R_2 อย่างแน่นอน ซึ่งสามารถแก้ไขได้ดังนี้

- เพิ่ม D_1 เข้าไปในวงจร ในขณะที่ทำการประจุ C_1 ก็จะไม่มีกระแสไหลผ่าน R_2 แต่ในทางกลับกัน เมื่อ C_1คายประจุ จะคายประจุผ่าน R_2 ทำให้กระแสที่ทำการประจุจะต่ำกว่ากระแสที่ใช้ในการคายประจุ ถ้าเราต้องการให้ R_1 เป็นตัวกำหนดกระแสประจุร่วมกับ C_1 เท่านั้น ก็เพียงแต่เพิ่ม D_2 เข้าไปในวงจร แต่วิธีการเพิ่มไดโอดนี้จะทำให้ผลของการคายประจุและเก็บประจุของ C_1 เปลี่ยน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แปลงไปแบบไม่เป็นเชิงเส้น (non-linear) เพราะว่าจะมีแรงดันตกคร่อมที่ตัวไดโอด ดังนั้น ระดับของแรงดันของแหล่งจ่ายไฟก็จะมีผลต่อพัลส์ที่เอาต์พุต โดยปกติ ที่ขา 5 (อินพุตของแรงดันควบคุม) ของไอซี 555 แบบไบโพลาร์ (Bi-Polar) จะต้องต่อตัวเก็บประจุค่าประมาณ 10 นาโนฟารัด (nanofarad) ลงกราวด์ เพื่อป้องกันสัญญาณรบกวน แต่สำหรับไอซีรุ่นซีมอส (CMOS) เราไม่จำเป็นต้องต่อตัวเก็บประจุลงกราวด์

- นำเอาต์พุตจาก ขา 3 ของไอซี 555 มาผ่าน PNP ทรานซิสเตอร์ (BD 140) เพื่อกลับเฟสเอาต์พุตที่ออกมาจากขา 3 ของไอซี 555 ทำให้เราสามารถปรับค่าดิวตี้ไซเคิลให้ต่ำกว่า 0.5 ได้ การที่เราปรับค่า duty cycle อยู่ระหว่าง 5% ถึง 10% แต่ไม่ควรให้ต่ำกว่า 5% เพราะ power ที่ออกมาจะน้อยเกินไป



รูปที่ 2.9 วงจรที่ปรับปรุงแล้ว ให้ Duty cycle น้อยกว่า 50%

2.5 ค่าผิดพลาดของเวลา

เมื่อเรามาพิจารณาถึงความแม่นยำของช่วงเวลาที่ไอซี 555 สร้างขึ้น จะเห็นว่าขึ้นอยู่กับตัวแปรมากมาย ทั้งนี้ อยู่ในตัวไอซีเองและอยู่นอกตัวไอซี การสร้างแรงดันอ้างอิงภายในตัวไอซีเพื่อรวมกับแรงดันอินพุตออฟเซ็ต (offset) ของวงจรเปรียบเทียบแบบทริกเกอร์และแบบเทรสโฮลด์ จะทำให้ช่วงเวลาที่ถูกสร้างขึ้น มีความผิดพลาดประมาณ 2% ช่วงเวลาของการตอบสนอง (reaction times) และช่วงเวลาในการกลับคืนสู่สภาวะปกติ (recovery times) ภายในตัวไอซี ก็เป็นสาเหตุหนึ่งที่ทำให้เกิดความคลาดเคลื่อนนี้ด้วย

รูปที่ 2.8 แสดงถึงการทำงานของไอซี 555 เมื่อใช้เป็นวงจรอะสเตเบิล ที่ให้เอาต์พุตความถี่สูง และแสดงให้เห็นถึงวงจรอะสเตเบิลที่กำหนดค่าดิวตี้ไซเคิลประมาณ 0.6 จะได้ความถี่เอาต์พุตประมาณ 29 kHz แต่จากการคำนวณแล้วควรจะได้ความถี่ที่เอาต์พุตประมาณ 25 kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนรูปที่ 2.9 เราใช้วงจรเดิม แต่กำหนดค่าดิวตี้ไซเคิลสูงกว่าเดิม แต่ยังคงให้ค่าความต้านทานรวมเท่าเดิมคือ $R_1 + 2R_2$

ดังนั้น ความถี่เอาต์พุตที่ได้จากการคำนวณก็ยังคงมีค่าเท่าเดิม อย่างไรก็ตาม C_1ยังคงคายประจุที่ต่ำกว่าแรงดันทริกเกอร์ (คือแรงดันเทรสโฮลด์ ซึ่งแสดงด้วยเส้นในแนวนอน) จากผลบางส่วนนี้เอง ทำให้แรงดันตกคร่อม C_1 ลดลงอย่างรวดเร็วและทำให้การทำงานร่วมของวงจรเปรียบเทียบแบบทริกเกอร์ และทรานซิสเตอร์ คายประจุช้าลง ทำให้ความถี่ที่ได้รับต่ำกว่าที่ได้จากการคำนวณ คือ ประมาณ 20 kHz

ที่ความถี่ต่ำองค์ประกอบต่างๆ เช่น กระแสไฟรั่วไหลของตัวเก็บประจุที่กำหนดเวลา การคายประจุของทรานซิสเตอร์ และกระแสอินพุตของวงจรเปรียบเทียบแบบเทรสโฮลด์ จะมีผลต่อการทำงานเพิ่มขึ้น โดยทั่วไปแล้ว การใช้งานที่ความถี่ต่ำ ค่าความต้านทานของการประจุและคายประจุจะเพิ่มขึ้น ทำให้กระแสประจุลดลง และการเปลี่ยนแปลงของกระแสรั่วไหลก็จะมีผลมากขึ้น ดังนั้นควรจำว่า ถ้าใช้ตัวเก็บประจุแบบอิเล็กทรอไลต์ (Electrolyte) จะให้กระแสรั่วไหลสูง ทำให้ความจุของC_1 คลาดเคลื่อนไปจากเดิม และช่วงเวลาที่ได้ก็จะผิดพลาดเพิ่มขึ้น

ตารางที่ 2.1 แสดงคุณสมบัติของไอซี 555

คุณสมบัติ		ไอซี 555			ไอซี 7555			TLC 555			หน่วย
		ต่ำสุด	ปกติ	สูงสุด	ต่ำสุด	ปกติ	สูงสุด	ต่ำสุด	ปกติ	สูงสุด	
แรงดันไฟเลี้ยง		4.5		1.8	2		18	2		18	V
กระแส	2V		-	-	-		-			0.25	MA
	5V		3	5	0.08		0.4		0.17	0.35	MA
	10V		10	12	0.12		0.6		0.36	0.60	MA
เอาท์พุท	Isink		200		8		80	100			MA
	Isource		200		1		20	10			MA
กระแสเทรสโฮลด์			100	250			10	0.01			NA
กระแสคายประจุขณะอยู่ในสภาวะเปิด			20	100			10	0.1			NA
ค่าผิดพลาดของเวลาของการต่อโมโนสเตเบิล			1	3	2				1	3	%

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเลื่อนของอุณหภูมิ			500	250						Ppm
แรงดันไฟเลี้ยงที่เลื่อนไป			0.5		0.3	1		0.1	0.5	%/V
ช่วงเวลาเพิ่มขึ้น		100	300	75			20			NS
ช่วงเวลาลด		100	300	75			75			NS
ความถี่สูงสุด			0.5	1			2			MHz

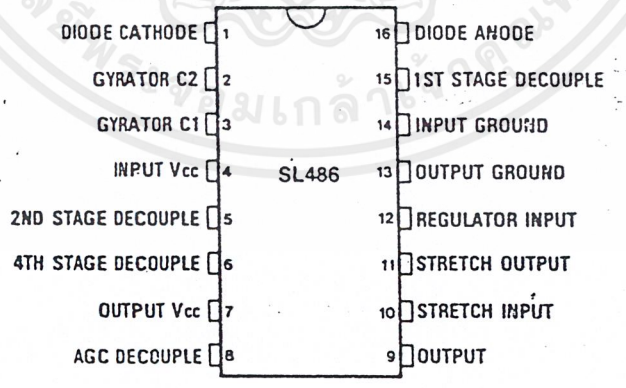
หมายเหตุ บันทึกที่อุณหภูมิ 25 องศาเซลเซียส

2.6 คุณสมบัติของไอซี SL 486 ปรีแอมป์อินฟราเรดรีโมตคอนโทรล

SL 486 ได้ออกแบบมาเพื่อใช้ในภาคปรีของอินฟราเรดรีโมตคอนโทรล โดยออกแบบให้ใช้ร่วมกับไดโอดอินฟราเรด ซึ่งอาศัยสัญญาณดิจิตอลทางด้านอินพุตไปควบคุมวงจรภาครับ วงจรภายใน SL 486 ประกอบด้วยวงจรพื้นฐาน 2 ส่วน คือวงจรกำหนดสัญญาณเอาต์พุต และวงจรควบคุมแรงดัน ซึ่งมีคุณสมบัติต่อไปนี้

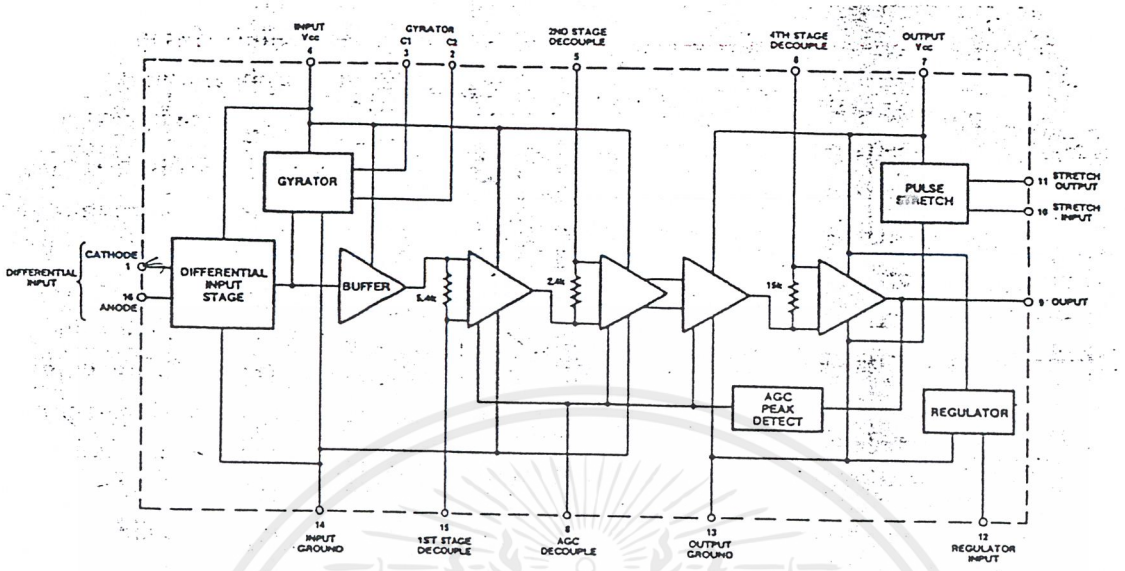
- มีอัตราการปรับสัญญาณอัตโนมัติ ที่รวดเร็วแม้ในสภาวะแวดล้อมที่มีสัญญาณรบกวน
- แยกความแตกต่างระหว่างอินพุต และลดสัญญาณรบกวน เพื่อให้มีความเที่ยงตรงสูง
- รักษาระดับการทำงานของวงจรต่อสภาพแวดล้อมที่มีการเปลี่ยนแปลงของแสง
- สัญญาณรบกวนทางด้านเอาต์พุตต่ำ

การจัดขาและโครงสร้างภายใน ดังรูปที่ 2.10 และ 2.11



รูปที่ 2.10 ลักษณะการจัดขาของ SL 486

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.11 โครงสร้างภายในของ SL 486

2.6.1 ลักษณะการทำงาน

ดูจากรูปที่ 2.12 และ 2.13 ไจเรเตอร์ C_1 (ที่ขา 3) ที่สภาพแวดล้อมต่อการทำงานของ SL 486 จะมีการจำกัดแสงจากภายนอก ซึ่งมีผลต่อไดโอดที่กระแสสูงสุด 200 ไมโครแอมป์ (ดูในรูปที่ 2.12) ค่าของ C_1 68 ไมโครฟารัด แสดงในรูปที่ 2.13 สาเหตุที่ขา 3 ไม่ต่อกับวงจรเพราะเป็นผลในการนำไปใช้งาน เมื่อคุณลักษณะของอุปกรณ์มีการลดลงต่อสภาพแวดล้อม

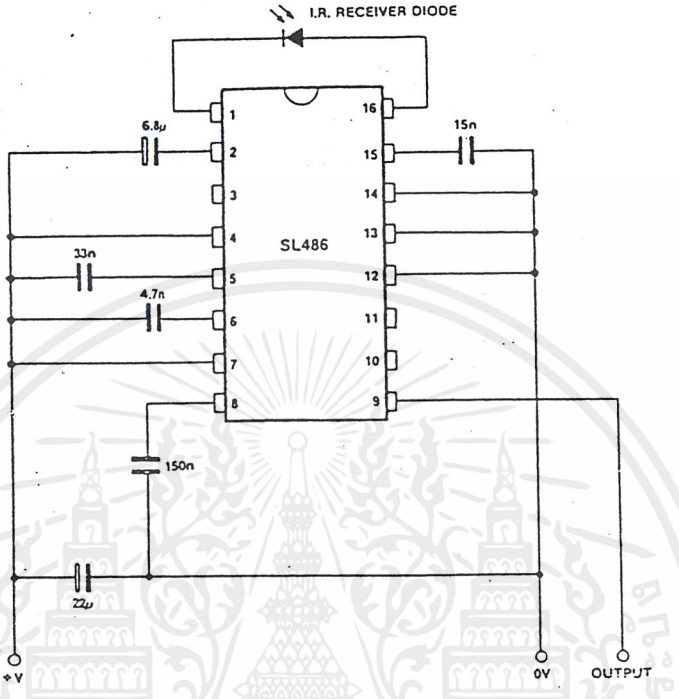
ตัวเก็บประจุ 68 ไมโครฟารัดสามารถสลับเปลี่ยนที่โดยใช้ตัวต้านทานแทนได้ เป็นลักษณะของการลดลงของอัตราการขยายในสภาวะแสงในระดับ 200 ไมโครแอมป์ ค่าในการเลือกตัวต้านทานประมาณ 10 กิโลโอห์ม และ 200 กิโลโอห์ม

ความทนทานต่อการรบกวนสัญญาณเอาต์พุตที่มีความเกี่ยวข้องกับระบบภาครับ ตลอดจนวัตถุประสงค์ที่จะกำหนดค่าของสัญญาณเอาต์พุต รวมถึง C_8 (ในรูปที่ 2.13) สาเหตุที่สัญญาณพัลส์จากขา 9 นำไปใช้รวมกับอินพุต เพราะว่ามีสัญญาณรบกวนรวมมากับพัลส์ จากขา 9 มีสัญญาณต่ำจึงไม่สามารถเห็นได้เมื่อเอาต์พุตออกที่ขา 11 ลักษณะการพัฒนาให้สามารถใช้การช่วยเหลือด้านอินพุต โดยรวมถึงฟิลเตอร์ทางด้านของเอาต์พุต (C_{10} ในรูปที่ 2.13) ค่าที่สามารถปรับ (100 พิโกฟารัด) เพื่อลดสัญญาณรบกวนพัลส์

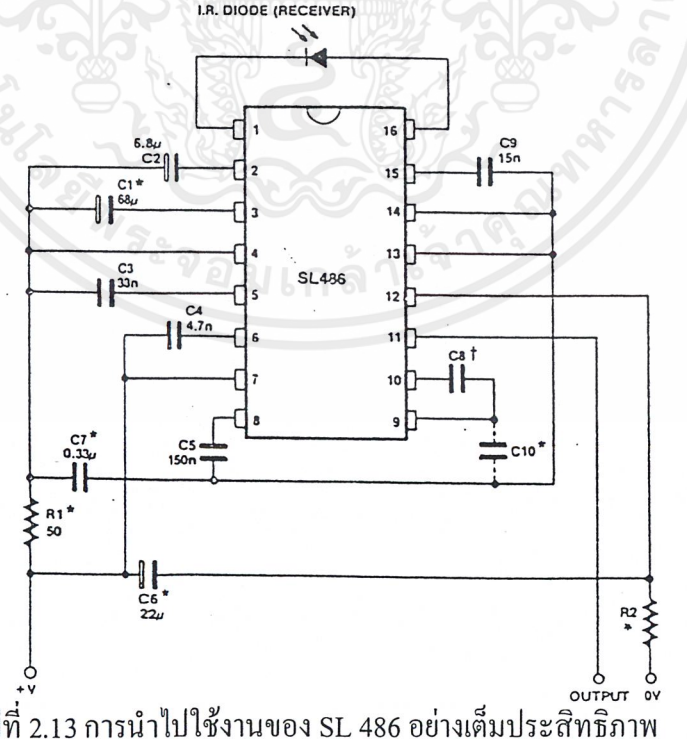
สกรีนนิ่ง ใช้สำหรับสกรีนอุปกรณ์ เพื่อกำหนดความทนทานต่อสัญญาณรบกวนภายนอก ระบบสกรีนนิ่ง ใช้วิธีป้องกันความไวในวงจรส่วนหน้าของอุปกรณ์ โดยกำหนดให้ไดโอดที่ขา 1 และ 16 ,C_2 (ขา 2) เป็นขั้นตอนแรกในการดีคัปเปิล (ขา 15) การสกรีนนิ่งอาจจะพบในการนำไปใช้งาน ในส่วนที่เหลือของวงจรไม่มีการป้องกัน การนำไปประยุกต์ใช้งานเมื่อมีสัญญาณรบกวน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในระดับปกติที่จะมีการสกรีนน้อยที่ขา 1 และ 16 ซึ่งมีไดโอดต่ออยู่ ซึ่งในบางกรณีสกรีนไม่มีความจำเป็นขึ้นอยู่กับสัญญาณรบกวน ซึ่งการดีคับปลิ้งและการฟิลเตอร์ มีความจำต่อการถอดรหัสในการรับ



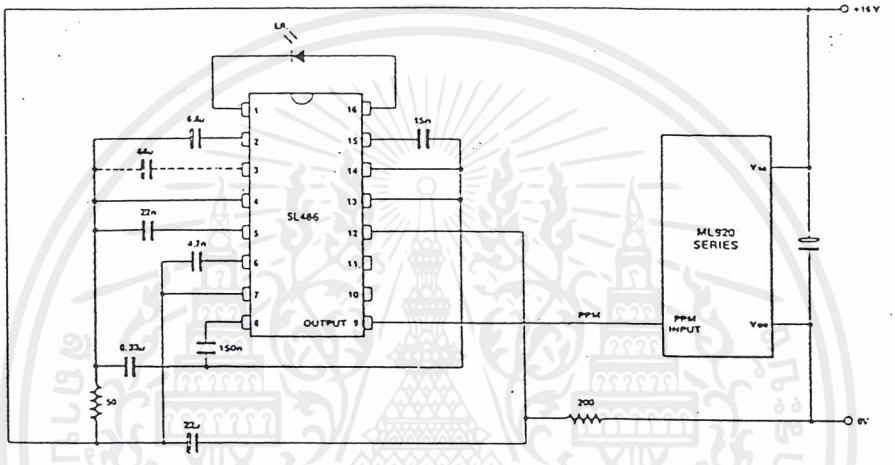
รูปที่ 2.12 การนำไปประยุกต์ของ SL 486 ด้วยอุปกรณ์น้อยชิ้น



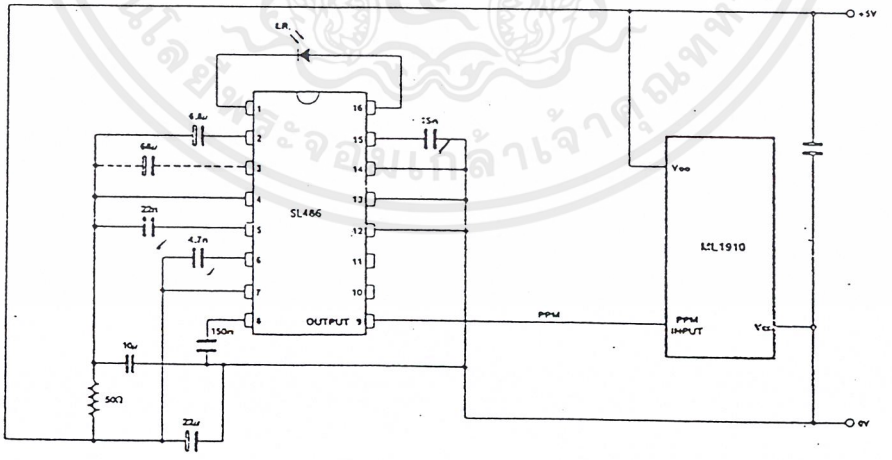
รูปที่ 2.13 การนำไปใช้งานของ SL 486 อย่างเต็มประสิทธิภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดีคัปปลิ้ง การปรับแต่งดีคัปปลิ้งสำหรับใช้งานโดยปราศจากเรกูเลต ดังในรูปที่ 2.13 และ 2.11 ลักษณะการนำเรกูเลเตอร์มาใช้งานในความถี่สูง แหล่งจ่ายประกอบด้วย R_2 โดยการเลือกค่าที่ขา 12 ของ R_2 โดยใช้ –9.0 ถึง –18 โวลต์ ตัวอย่างเช่น การนำไปใช้รวมกับ ML 920 SERIES ดังแสดงในรูปที่ 2.14 โดยใช้ซัพพลาย 16 โวลต์ โดยการนำค่า R_2 ประมาณ 200 โอห์ม ในการเรกูเลตจะมีจุดที่มีอิมพีแดนซ์ต่ำระหว่างขา 12 และ 13 และ C_7 มีค่าเป็น low อิมพีแดนซ์ระหว่างขา 4 และ 12 ที่มีความถี่สูง และลักษณะคุณสมบัติทางไฟฟ้า ดังแสดงในตารางที่ 2.2



รูปที่ 2.14 การนำ SL486 ไปใช้งานร่วมกับ ML920 Series



รูปที่ 2.13 การนำ SL486 ไปใช้งานร่วมกับ ML 1910

2.6.2 การนำไปใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดูตัวอย่างจากรูปที่ 2.13 ไดโอดที่ต่อระหว่างขา 1 และ 16 เป็นตัวรับแสงอินฟราเรดอยู่ในส่วนอินพุตของวงจร ซึ่งสัญญาณจากขาทั้ง 2 จะถูกเปรียบเทียบกัน ทำให้อุปกรณ์มีความเที่ยงตรงมากขึ้น เป็นจุดสำคัญในการลดความไวต่อสภาวะแวดล้อมที่มีสัญญาณรบกวน

ไจเรเตอร์ C_2 และ C_1 ที่ต่อระหว่างขา 2 และ 3 จะเป็นตัวดีคัปปลิ้งสัญญาณ ซึ่งไจเรเตอร์จะกำเนิดสัญญาณในช่วง โรลส์-ออฟ (rolls-off) และมีการฟีดแบ็คกันในลูป (loop) เพื่อให้เกิดความสมดุลกับสัญญาณจากไดโอดอินฟราเรด ค่าของ C_2 และ C_1 ถูกเลือกให้ผลิตความถี่ในช่วงที่ต่ำกว่าความถี่คัตออฟ (cut-off) 2 กิโลเฮิรตซ์ ซึ่งวงจรไจเรเตอร์นี้มีอัตราการลดสัญญาณที่มีความถี่ 100 เฮิรตซ์ ลงได้ถึง 20 dB นอกจากนี้ไจเรเตอร์จะสร้างสัญญาณฟีดแบ็คให้การทำงาน 2 ส่วน คือ ส่วนแรกจะทำหน้าที่เมื่อไดโอดมีกระแสต่ำกว่า 200 ไมโครแอมป์ และจะดีคัปปลิ้งโดย C_2 สำหรับกระแสของไดโอดจะอยู่ในช่วงระหว่าง 200 ไมโครแอมป์ และ 1.5 มิลลิแอมป์ ในส่วนที่ 2 จะเป็นตัวควบคุมการทำงานของลูปและดีคัปปลิ้ง โดย C_1

การดีคัปปลิ้งในขั้นที่ 1 , 2 และ 4 ของตัวอย่างในรูปที่ 2.13 จะต่อตัวเก็บประจุค่า 15 , 33 และ4.7 นาโนฟารัด ที่ขา 15 , 5 และ 6 ตามลำดับ ซึ่งจะทำหน้าที่ดีคัปปลิ้งสัญญาณที่ขานอนอินเวอร์ติ้ง เป็นการขยายความแตกต่างของสัญญาณ โดยกำหนดค่าของตัวเก็บประจุในแต่ละชั้นนี้จะเลือกผลิตความถี่ที่ 2 กิโลเฮิรตซ์ ซึ่งเป็นความถี่ที่ต่ำกว่าช่วงโรลส์-ออฟ

เอาต์พุตจากวงจรขยายความแตกต่างขั้นที่ 4 จะถูกส่งเข้าไปยังวงจร AGC เพื่อตรวจสอบยอดระดับแรงดัน และควบคุมอัตราขยายให้คงที่ก่อนส่งออก และป้อนกลับไปยังวงจรภาคต้นๆ ซึ่งในส่วนนี้จะเหมือนเป็นแหล่งจ่ายกระแสคงที่ที่จำกัดเพียง 10 มิลลิแอมป์ บริเวณขา 8 และจะต้องต่อตัวเก็บประจุดีคัปปลิ้ง (C_1) ค่าประมาณ 150 นาโนฟารัด ให้ด้วย

วงจร AGC มีอัตราการทำงานที่รวดเร็วมากเมื่อภาคอินพุตสามารถตรวจจับพัลส์อินฟราเรดได้ อัตราการขยายของวงจรจะลดลงโดยทันที ทำให้สัญญาณรบกวนระดับต่ำที่ปะปนมาถูกกำจัดออกไป และไม่ไปปรากฏทางด้านเอาต์พุต คงเหลือไว้แต่สัญญาณอินฟราเรดที่เป็นข้อมูลเท่านั้น แต่หากจำเป็นต้องไปใช้ในสถานที่ที่มีการรบกวนสูงจริงๆ วงจร AGC ก็ยังมีการหน่วงเวลาและกำจัดสัญญาณรบกวนออกไก่อนที่จะส่งออกไปยังเอาต์พุต หรือวงจรภาคอื่นๆ ต่อไป

สัญญาณพัลส์อินฟราเรดจากอินพุต จะทำให้ระดับแรงดันเอาต์พุต (ขา 9) ของไอซีมีระดับต่ำลง โดยที่อินพุตที่รับเข้ามาจะถูกขยายอย่างเป็นเชิงเส้น และให้เอาต์พุตที่มีค่าแรงดันสวิงได้ตั้งแต่ระดับกราวด์จนถึง V_{cc}

ปกติการส่งสัญญาณอินฟราเรดในระบบ PPM (pulse position modulation) นี้จะมีคาบเวลาของพัลส์แคบมากอยู่ในช่วง 15 ไมโครวินาที ซึ่งหากจะวิเคราะห์ถอดรหัสกันจริงๆ คงต้องอาศัยระบบไมโครโปรเซสเซอร์เข้ามาช่วยยึดคาบเวลาดังกล่าว แต่โดยตัวไอซีเองแล้ว ก็สามารถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

กระทำการยืดคาบเวลาของพัลส์อินพุตดังกล่าวได้ โดยการต่อตัวเก็บประจุเข้าระหว่างขา 9 และ 10 และเปลี่ยนเอาต์พุตจากขา 9 ไปเป็นขา 11 แทน

ความกว้างของพัลส์กำหนดได้โดยค่าของตัวเก็บประจุที่คัปปลิ้ง(C_8ซึ่งแสดงในรูปที่ 2.12) หาได้โดยใช้สูตร

$$T_p = -R \times C_8 \ln \{1.5 / (V_4 - V_{13})\} \qquad (2.15)$$

เมื่อให้

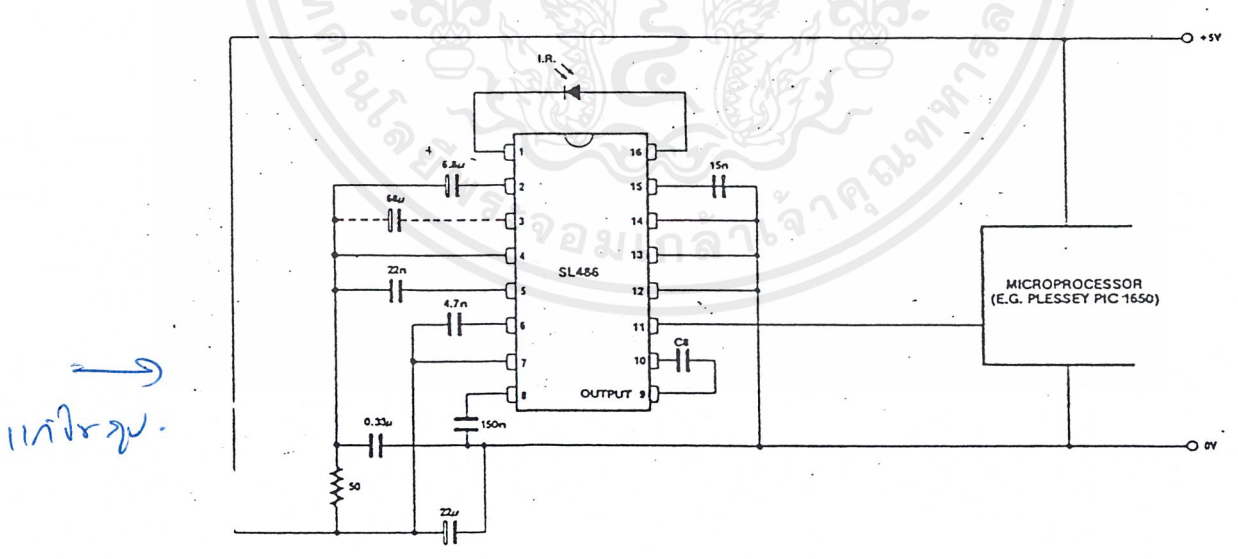
T_p = ความกว้างของพัลส์มีหน่วยเป็นมิลลิวินาที (ms)

R_X = 200 กิโลโอห์ม (ดูจากตารางที่ 2.2)

C_8 = ตัวเก็บประจุคัปปลิ้ง

$(V_4 - V_{13})$ = แรงดันระหว่างขา 13 และ 14

ไอซีตัวนี้สามารถทำได้สบายๆ ในช่วงแรงดันตั้งแต่ 4.5 – 9 โวลต์ โดยจะต่อระหว่างขา 13 , 14 ซึ่งเป็นกราวด์ และ ขา 4 , 7 ซึ่งเป็น V_{CC} ดังรูปที่ 2.10 สำหรับการใช้งานในระดับแรงดันที่เกินกว่า 9 โวลต์ (ไม่เกิน 18 โวลต์) สามารถทำได้โดยอาศัยชิพเรกูเลเตอร์ภายใน โดยจะต้องต่อขา 7 เข้ากับขา 12 ซึ่งจะทำให้ขาเอาต์พุตกราวด์ (ขา 12) มีระดับแรงดันคงที่ประมาณ 6.4 โวลต์ และชิพเรกูเลเตอร์ภายในนี้จะทำงานได้ผลก็ต่อเมื่อมีระดับแรงดันไฟเลี้ยงที่เกินกว่า 9 โวลต์เท่านั้น และหากไม่จำเป็นต้องใช้ชิพเรกูเลเตอร์ภายในก็ควรลัดวงจรขา 12 เข้ากับขา 13



รูปที่ 2.16 การนำ SL486 ไปใช้งานร่วมกับไมโครโปรเซสเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

42639

ตารางที่ 2.2 แสดงคุณสมบัติทางไฟฟ้าของ SL 486

Characteristic	Pin	Value			Unit	Conditions
		Min.	Typ.	Max.		
Supply current (See Note 1)	4,7		6.5	9.0	mA	Vcc = 5.0V, Ioooe = 1.0μA } Pins 13 & 14
	4	3.5•3xIo	4.2•3xIo	5•3xIo	mA	Vcc = 4.5V, Ioooe ≤ 1.5mA } ground
	4,7		8.5	10	mA	Vcc = 18V, Ioooe = 1.0μA Pin 12 ground
Low voltage supply (external)	4,7(+ve), 13,14(-ve)	4.5		9.5	V	Input and output Vcc commoned, input and output ground commoned
High voltage supply (external)	4,7(+ve), 12(-ve)	8.4		18.0	V	Input and output Vcc commoned, input and output ground at internal regulated voltage
Internal regulated voltage	13(wrt 7)	5.9	6.2	6.5	-V	V Pin 7(+) to V Pin 12(-) = +16V
Voltage between input and output Vcc	4,7			1.5	V	At room temperature
				1.1	V	At 70° C
Minimum sensitivity of differential input	1,16	9.0		2.3	nA	Ioooe = 1.0μA
		74.0		18.5	nA	Ioooe = 100μA
		168.0		42.0	nA	Ioooe = 0.5mA
Common mode rejection	1,16	-	35.0		dB	
Maximum signal input	1,16	3.0	4.0		mA(peak)	
AGC range			68.0		dB	
Output and stretch output pull-up resistance (internal)	9,11		55.0		kΩ	At 25°C
Stretch output pulse width (T_s)	11		2.4		ms	Capacitance Pin 9 to Pin 10 = 10nF; $T_s \simeq -R_x C \ln \left\{\frac{1.5}{V_{cc}}\right\}$,
T co-efficient on Rx			0.7		%/°C	Where Rx = 200kΩ ± 25% (internal resistance)
Output low	9			Output ground +0.35	V	0.2mA Sink, max.
Output high	9	Output Vcc -0.5			V	5μA Source
Stretch output low	11			Output ground +0.5	V	1.6mA Sink, max.
Stretch output high	11	Output Vcc -0.1			V	Output open circuit 5μA Source
Supply rejection, input Vcc	4		1.5		V(peak)	Ripple amplitude at 100Hz, Pin 12 ground
			0.8		V(peak)	Ripple amplitude at 100Hz, Pins 13 & 14 ground

2.7 วงจรเปรียบเทียบและชมิททริกเกอร์ (The comparatop and Schmitt Trigger)

2.7.1 วงจรเปรียบเทียบ (Comparators)

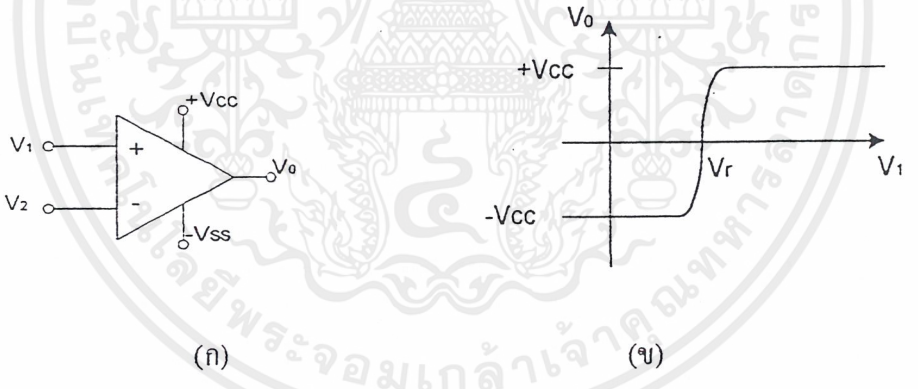
การป้อนกลับบางชนิดมักจะเกี่ยวเนื่องกับการใช้ออปแอมป์(Operation Amplifier) เสมอ แต่มีการประยุกต์ใช้ออปแอป์แบบ non-linear บางชนิดที่ไม่ต้องใช้ feedback พิจารณาตามรูปที่ 2.17 เป็นการใช้ op-amp โดยไม่มีการป้อนกลับ (Feedback) ขณะที่ supply voltage เป็น + V_{CC} และ- V_{CC} เป็นจุดบ่งชี้เราสมมติว่า ค่าอัตราการขยาย(Gain) ของ op-amp คือ A มีค่ามากๆ (ค่า gain โดยปกติของ op-amp 741 คือ A = 200,000) สมมติว่าหนึ่งในอินพุตคือ สัญญาณกลับของอินพุต v_2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถูกกำหนดให้เป็นแรงดันอ้างอิง $V_r > 0$ V นั่นคือ แสดงให้เห็นว่า $v_2 = V_r$ แม้ว่าเอาต์พุตของ op-amp จะเป็น $v_0 = A(v_1 - v_2) = A(v_1 - V_r)$ เนื่องจากค่าอัตราขยายของ op-amp มีค่ามาก ดังนั้นถ้า v_1 มากกว่า V_r เล็กน้อย เอาต์พุตของ op-amp จะเกิดการ saturate และเราจะได้ว่า $v_0 = + V_{CC}$ (ในทางอุดมคติ)

ในทำนองเดียวกัน ถ้า $v_1 < V_r$ แล้ว $v_0 = - V_{CC}$ ผลเหล่านี้สรุปได้จาก transfer characteristic อุปกรณ์ที่ไม่เป็นเชิงเส้น (non linear) เช่นนี้ เป็นตัวอย่างหนึ่งของอุปกรณ์เปรียบเทียบ เช่น ในรูปที่ 2.17(ก) แรงดันอินพุต v_1 จะถูกเปรียบเทียบกับแรงดันอ้างอิง $v_2 = V_r$ และถ้า $v_1 > V_r$ แล้ว $v_0 = + V_{CC}$ ในขณะที่ถ้า $v_1 < V_r$ แล้ว $v_0 = - V_{CC}$

สำหรับในกรณีที่แรงดันอ้างอิงไม่เป็นศูนย์ ($V_r \neq 0$V) comparator นี้จะถูกเรียกว่า Level Detector และถ้าแรงดันอ้างอิงมีค่าเป็นศูนย์ ($V_r = 0$V) ก็จะเรียกว่า Zero-crossing Detector รูปที่ 2.17(ก) แสดงวงจร Zero-crossing detector รูปที่ 2.17(ข) แสดงแรงดันอินพุต sinusoidal v_1 และให้แรงดันเอาต์พุตเป็น v_0ดังนั้นนี้จึงเป็นวงจรตัวอย่างที่ใช้ในการเปลี่ยน sine wave ไปเป็น square wave



(ก) (ข)

รูปที่ 2.17 (ก) comparator (ข) comparator transfer characteristic

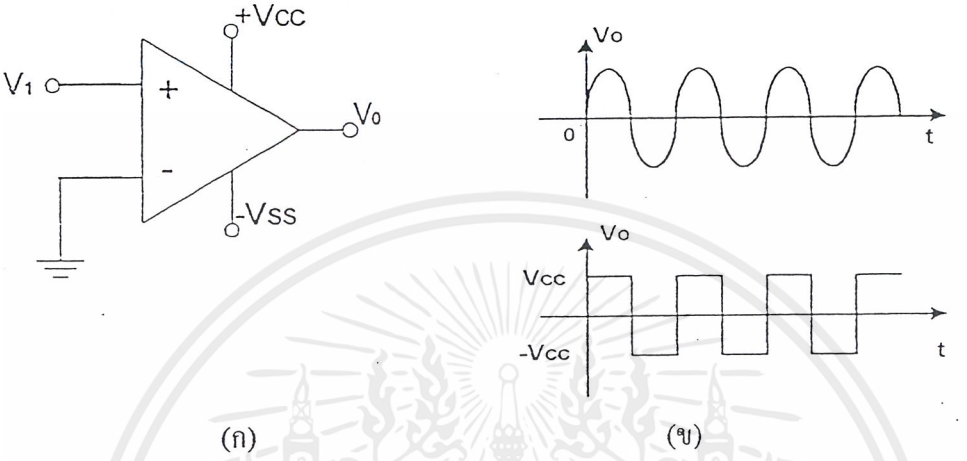
แม้ว่า op-amp ที่มีค่า gain สูง และใช้งานได้ตามจุดประสงค์ทั่วไปนี้ จะสามารถใช้งานเป็นcomparatorได้แต่ก็มีop-ampบางชนิดที่ถูกออกแบบมาโดยเฉพาะสำหรับการใช้งานเป็น comparator โดยเฉพาะ โดยอุปกรณ์เหล่านี้จะมีความเร็วที่สูงกว่า op-amp ธรรมดาทั่วๆ ไป

2.7.2 ชมิตทริกเกอร์ (The schmitt Trigger)

เราสามารถนำ positive feedback มาใช้กับ op-amp ในลักษณะ compartor ได้ดังรูปที่ 2.18 และถูกเรียกว่าวงจร schmit trigger โดยมีค่า loop gain เป็น 1 หรือมากกว่า โดยมีการป้อนกลับที่ขา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บวก (regenerative) ค่าแรงดันเอาต์พุตที่ได้จะ saturate ที่ค่า + V_{CC} หรือ - V_{CC} ซึ่งสองค่านี้ไม่ได้ขึ้นอยู่กับค่าของอินพุตเท่านั้น แต่ขึ้นอยู่กับค่าเอาต์พุตด้วย



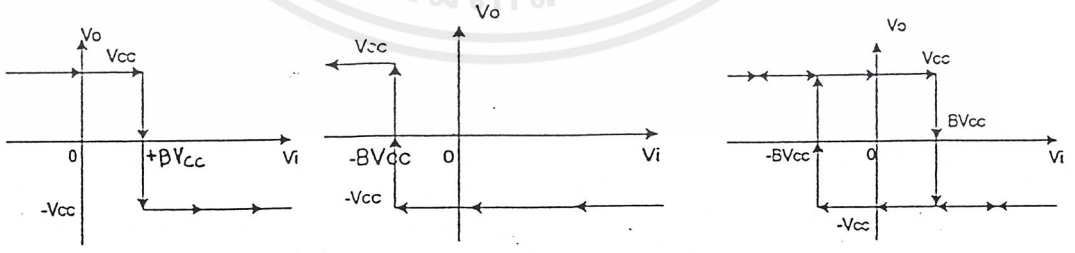
(ก) (ข)

รูปที่ 2.18 (ก) zero crossing detector (ข) input and output

สมมติในตอนแรกว่า แรงดันเอาต์พุตคือ $v_0 = +V_{cc}$ โดยการแบ่งแรงดัน จะทำให้

$$v_2 = \{R_1 / (R_1+R_2)\} v_0 \qquad (2.16)$$

โดย $B = R_1/(R_1+R_2)$ ถ้า $v_1 < BV_{CC}$ แล้ว อินพุตโวลต์เทจ (Voltage) ของ op-amp คือ $v_2 - v_1 = BV_{CC} - v_1 > 0V$ และแรงดันเอาต์พุตยังคงเป็น + V_{CC} ในทางกลับกัน ถ้า $v_1 > BV_{CC}$ แล้ว อินพุตของ op-amp คือ $v_2 - v_1 = BV_{CC} - v_1 < 0V$ และค่าแรงดันเอาต์พุตจะเป็น - V_{CC} Transfer characteristic ของ Schmitt trigger เมื่อแรงดันอินพุต v_1 เพิ่มขึ้นเป็นดังรูปที่ 2.19(ก)



(ก) (ข) (ค)

รูปที่ 2.19 แสดง characteristic เมื่อ (ก) V1 เพิ่ม (ข) V1 ลด (ค) ผลทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อสมมติว่า แรงดันเอาต์พุต คือ $v_0 = -V_{CC}$ แล้ว

$$v_2 = \{R_1/(R_1+R_2)\} v_0 = -BV_{cc} \quad (2.17)$$

ถ้า $v_1 > -BV_{CC}$ แล้ว แรงดันอินพุตของ op-amp คือ $v_2 - v_1 = -BV_{CC} - v_1 < 0V$ และเอาต์พุตยังคงเป็น $-V_{CC}$ ในทางกลับกัน ถ้า $v_1 < -BV_{CC}$ แล้ว อินพุตของ op-amp จะเป็น $v_2 - v_1 = -BV_{CC} - v_1 > 0$ และเอาต์พุตจะกลายเป็น $+V_{CC}$ Transfer characteristic ของ Schmitt trigger สำหรับเมื่อ v_1 ลดลง เป็นดังรูปที่ 2.19 (ข)

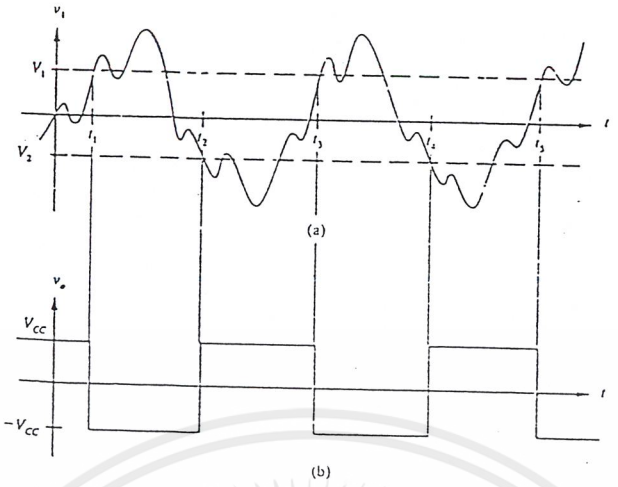
เมื่อรวมเอา Schmitt-trigger Transfer characterictics เข้าด้วยกันเราจะได้รูปTransfer characteristic ที่สมบูรณ์ดังรูป 2.19 (ค) สังเกตว่าแรงดันเอาต์พุตของ $+V_{CC}$ จะไม่เปลี่ยนเป็น $-V_{CC}$ จนกระทั่งแรงดันอินพุตเพิ่มขึ้นเกิน BV_{CC} และเอาต์พุตของ $-V_{CC}$ จะไม่เปลี่ยนเป็น $+V_{CC}$ จนกระทั่งอินพุตลดลงต่ำกว่า-BV_{CC} ปรากฏการณ์นี้เรียกว่า hysteresis ความกว้างของ hysteresis นี้ เป็นความแตกต่างระหว่างค่า threshold voltages สองค่า คือ BV_{CC} และ $-BV_{CC}$ ดังนั้น ในกรณีนี้ความกว้างของ hysteresis คือ $2BV_{CC}$

Schmitt trigger เป็นวงจรประเภท bistable ถ้า stable state หนึ่ง มีเอาต์พุตเป็น $v_0 = +v_{CC}$ อีก stable state หนึ่งก็คือ $v_0 = -v_{CC}$ วงจรจะคงอยู่ที่สองสถานะนี้ จนกระทั่งมันทริกเกอร์ไปสู่ stable state อื่น โดยต้องมีแรงดันอินพุตที่เหมาะสม $v_1 > BV_{CC}$ สำหรับ v_0 เปลี่ยนจาก $+V_{CC}$ ถึง $-V_{CC}$ และ $v_1 < -BV_{CC}$ สำหรับ v_0 เปลี่ยนจาก $+V_{CC}$ ถึง $-V_{CC}$

วงจร Schmitt trigger สามารถเปลี่ยน sinusoid ไปเป็นsquare waveได้โดยเฉพาะพิจารณาตามรูป เมื่อแรงดันอินพุตคือ v_1 เมื่อสัญญาณ sinusoid ถูกรบกวนโดยสัญญาณรบกวน (noise) ถ้าแรงดันนี้ถูกป้อนเข้าอินพุตของ Zero crossing detector เอาพุตจะเป็น $+V_{CC}$ ที่เวลา $t = 0s$ อย่างไรก็ตาม เนื่องจากผลของสัญญาณรบกวน ทำให้ในช่วงเวลา $0 < t < t_1$ เอาต์พุตจะกลายเป็น $-V_{CC}$ อีกครั้งเหตุการณ์อย่างนี้สามารถที่จะเกิดขึ้นอีกในเวลาต่อมา

วงจร schmitt trigger เรียกได้อีกอย่างหนึ่งว่า ทำหน้าที่ลดทอนสัญญาณรบกวนในวงจรที่มี hysteresis สามารถแก้ปัญหานี้ได้โดยดังรูปที่ 2.20 เมื่อแรงดัน v_1 ถูกป้อนเข้ายังวงจร schmitt trigger ทำให้มี upper threshold ของ v_1 และ lower threshold ของ v_2 ถ้าเอาต์พุตของ schmitt trigger เป็น $+V_{CC}$ เอาต์พุตจะไม่เปลี่ยนเป็น $-V_{CC}$ จนกว่า v_1 เกินกว่า V_1 ดังที่เห็น ปรากฏการณ์นี้เกิดขึ้นเมื่อเวลา $t = t_1$ อย่างไรก็ตาม ถ้าเอาต์พุตคือ $-V_{CC}$ มันจะไม่เปลี่ยนกลับมาเป็น $+V_{CC}$ จนกระทั่ง v_1 มีค่าลดลงจนน้อยกว่า V_2 ดังนั้นแม้ว่า v_1 จะค่อยๆ ลดจนต่ำกว่า V_1 ภายหลังที่เวลา $t = t_1$ นั้น ไม่มีทางที่เป็นสาเหตุให้เอาต์พุตเปลี่ยนไปจาก $-V_{CC}$ เอาต์พุตจะไม่มีการเปลี่ยนเป็น $+V_{CC}$ จนกว่าเวลา $t = t_2$ ก็คือช่วงเวลาที่ค่า V1 ลดลงต่ำกว่า V2 นั่นเอง เอาต์พุตของวงจรชมิททริกเกอร์จะเป็นไปตามรูปที่ 2.20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.20 สัญญาณซึ่งมีสัญญาณรบกวนกับสัญญาจากชมิททริกเกอร์

2.8 มูเล่ (Pulley)

การนำมูเล่มาใช้งานมีข้อดีคือ

- ต้นกำลังงานกับปลายแหล่งกำลังงานสามารถมีระยะห่างไกลกันได้ (คือระยะ L)
- สามารถเปลี่ยนแปลงความเร็วได้ง่ายไม่ซับซ้อน โดยการเปลี่ยนขนาดของมูเล่ (เปลี่ยนค่า R)

จากรูปจะได้ความสัมพันธ์ดังนี้

$$\frac{\omega_1}{\omega_2} = \frac{r_2}{r_1} = \frac{N_1}{N_2} \qquad (2.18)$$

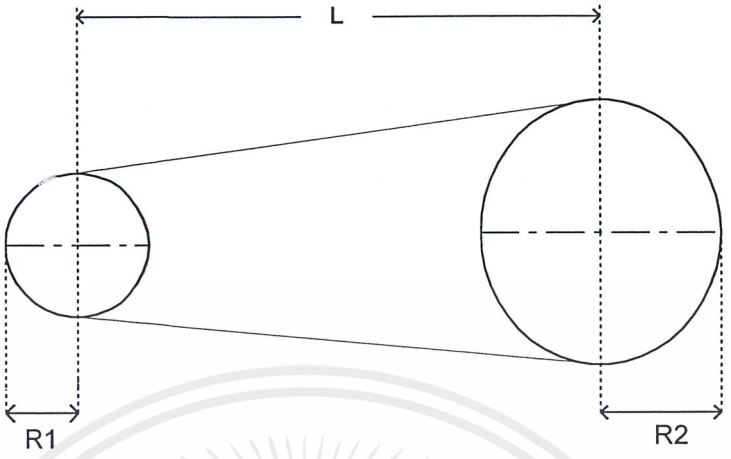
ω คือ ความเร็วเชิงมุม (Rad/sec)

r คือ รัศมีของมูเล่

N คือ จำนวนรอบการหมุนของมูเล่

เนื่องจากตัว DC Motor เองเมื่อมีไฟเลี้ยงจ่ายเข้าไป มันจะหมุนด้วยความเร็วที่ยังช้าอยู่ไม่เพียงพอที่จะทำให้ประตูเปิด-ปิดเร็วได้ เราจึงนำหลักการของมูเล่มาช่วยเพิ่มความเร็วในการเปิดปิดประตู โดยนำมูเล่ตัวใหญ่ซึ่งติดอยู่ที่ DC Motor มาหมุนมูเล่ตัวเล็กที่ติดอยู่ที่รางประตู ก็จะช่วยเพิ่มความเร็วในการเปิด-ปิดประตูได้ จากสมการจะพบว่า ยิ่งมูเล่สองตัว (คือ ตัวใหญ่ไปหมุนตัวเล็ก) ยิ่งรัศมีต่างกันมาก ก็จะยิ่งช่วยเพิ่มความเร็วในการหมุนของมูเล่ตัวเล็กให้มากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.21 พื้นฐานการทดความเร็วของมูเล่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
พอร์ตสื่อสารข้อมูลแบบอนุกรม ใน MCS-51

3.1 พอร์ตสื่อสารข้อมูลแบบอนุกรม

MCS-51มีพอร์ตสำหรับการสื่อสารข้อมูลแบบอนุกรมที่สามารถรับและส่งข้อมูลแบบอนุกรมได้ โดยที่ผู้ใช้ไม่จำเป็นต้องต่อชิปทำหน้าที่รับส่งข้อมูลแบบอนุกรมโดยเฉพาะเพิ่มแต่อย่างใดเลย การนำ MCS-51 ไปประยุกต์ใช้งานที่ต้องมีการติดต่อสื่อสารข้อมูลแบบอนุกรมกับวงจรภายนอกอื่นๆจึงทำได้สะดวกและมีความคล่องตัวสูงมาก

พอร์ตสื่อสารข้อมูลแบบอนุกรมที่มีใน MCS-51 สามารถทำงานได้ในแบบ full duplex หมายความว่า MCS-51 สามารถรับส่งข้อมูลได้พร้อมๆกัน โดยในการรับข้อมูลจะมีการบัฟเฟอร์ข้อมูลให้ด้วย จึงทำให้ MCS-51 สามารถกำหนดการรับส่งข้อมูลไบต์ที่สองซึ่งถูกส่งตามเข้ามาก่อนที่ไบต์แรกที่ได้รับเข้ามาจะถูกอ่านจากรีจิสเตอร์ใช้งานเฉพาะที่ใช้สำหรับรับข้อมูล (receiver register) เพื่อนำไปเก็บไว้ในหน่วยความจำต่อไป (หากไบต์แรกยังไม่ถูกอ่านเมื่อได้รับไบต์ที่สองเรียบร้อยแล้วข้อมูลจะหายไปหนึ่งไบต์)

พอร์ตสื่อสารข้อมูลแบบอนุกรมใน MCS-51 ประกอบด้วยรีจิสเตอร์ขนาด 8 บิตจำนวนสองตัวแต่ละตัวมีชื่อเรียกตามหน้าที่ดังนี้คือ

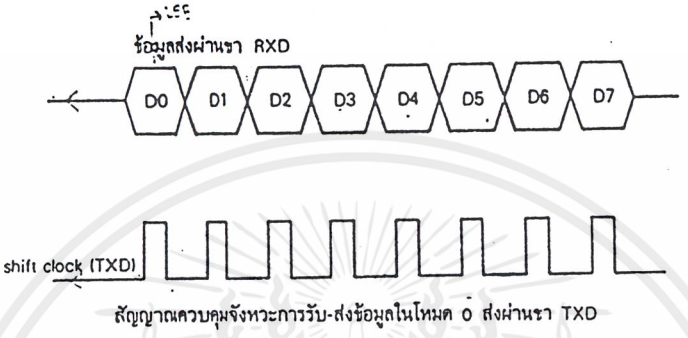
- รีจิสเตอร์สำหรับรับข้อมูลใช้รับข้อมูลที่ส่งเข้ามาจากภายนอก
- รีจิสเตอร์สำหรับส่งข้อมูล (transmit register) ใช้ส่งข้อมูลจาก MCS-51 ออกไปภายนอก

รีจิสเตอร์ทั้งสองมีตำแหน่งเดียวกันในรีจิสเตอร์ใช้งานเฉพาะ คือ ตรงกับตำแหน่งของรีจิสเตอร์ใช้งานเฉพาะ SBUF (ตำแหน่ง 99H) ในหน่วยความจำสำหรับเก็บข้อมูลภายในชิปที่ใช้เป็นรีจิสเตอร์ใช้งานเฉพาะ การเข้าถึงข้อมูลที่อยู่ในรีจิสเตอร์แต่ละตัว MCS-51 จะทราบเองว่าผู้ใช้ต้องการติดต่อกับรีจิสเตอร์ตัวใดโดยตรวจสอบจากรหัสคำสั่ง ทั้งนี้เพราะในการเขียนข้อมูลไปไว้ในรีจิสเตอร์ใช้งานเฉพาะ SBUF หมายถึงการโหลดข้อมูลไปที่รีจิสเตอร์สำหรับส่งข้อมูลเพื่อส่งข้อมูลออกไปภายนอก ส่วนการอ่านข้อมูลจากรีจิสเตอร์ใช้งานเฉพาะ SBUF จะหมายถึงนำค่าที่รับเข้ามาได้จากภายนอกที่เก็บไว้ในรีจิสเตอร์สำหรับรับข้อมูลมาใช้งาน ผู้ใช้สามารถกำหนดการทำงานที่แตกต่างกันได้ถึง 4 ประเภท โดยสามารถกำหนดได้จากค่าของบิตในรีจิสเตอร์ใช้งานเฉพาะ SCON ดังรูปที่ 3.1 การใช้งานที่แตกต่างกันทั้ง 4 ประเภทนี้มีจุดประสงค์เพื่อความคล่องตัวในการรับหรือส่งข้อมูลแบบอนุกรมแต่ละประเภทดังนี้

โหมด 0 การทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมในโหมด 0 ขา RXD จะใช้สำหรับรับและส่งข้อมูล ส่วนขา TXD มีไว้เพื่อสร้างสัญญาณ shift clock เพื่อกำหนดจังหวะในการรับและส่งข้อมูล (ข้อมูลจะถูกรับหรือส่งตามจังหวะของสัญญาณ shift clock) ในโหมดนี้การรับส่ง

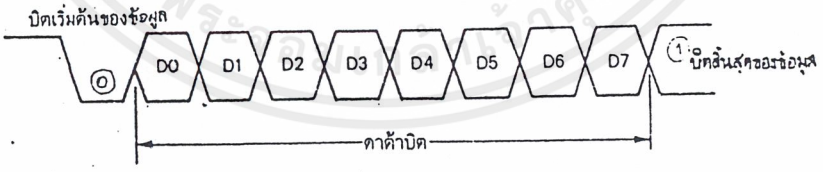
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลจะเป็นแบบ 8 บิต (บิตข้อมูล 8 บิต) โดยเริ่มรับและส่งบิตต่ำสุดก่อน อัตราการรับส่งข้อมูลในโหมด 0 ถูกกำหนดไว้ที่ 1/12 ของความถี่ออสซิเลเตอร์ที่ใช้ การทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมในโหมด 0 จะไม่มีบิตเริ่มต้นของข้อมูล (start bit) และบิตสิ้นสุดของข้อมูล (stop bit) เพราะจังหวะการรับและส่งข้อมูลถูกกำหนดจากสัญญาณ shift clock แล้ว



รูปที่ 3.1 แสดงข้อมูลที่รับและส่งในการทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมโหมด 0

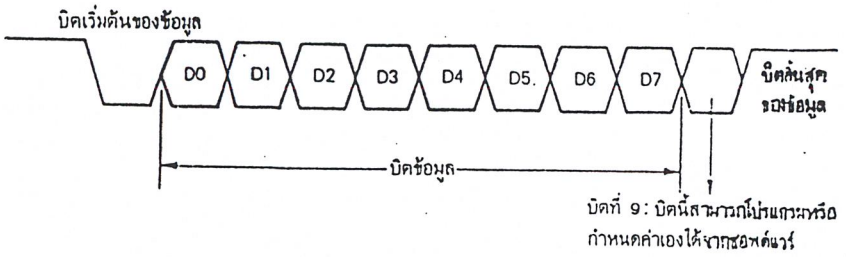
โหมด 1 การทำงานในโหมดนี้มีการรับและส่งข้อมูลครั้งละ 10 บิต ข้อมูลจะส่งออกไปภายนอกผ่านทางขา TXD และรับข้อมูลเข้ามาทางขา RXD ข้อมูลทั้ง 10 บิตประกอบด้วยบิตเริ่มต้นของข้อมูล 1 บิต (มีค่าเป็น 0 เสมอ) บิตข้อมูล 8 บิต (รับและส่งบิตต่ำสุดก่อน) และบิตสิ้นสุดของข้อมูลอีก 1 บิต (มีค่าเป็น 1 เสมอ) ในขณะทำการรับข้อมูล ค่าในบิตสิ้นสุดของข้อมูลที่รับได้จะไปอยู่ในบิต RB8 ของรีจิสเตอร์ใช้งานเฉพาะ SCON อัตราเร็วในการรับหรือส่งข้อมูลของพอร์ตสื่อสารข้อมูลแบบอนุกรมในโหมดนี้สามารถเปลี่ยนแปลงได้



รูปที่ 3.2 แสดงข้อมูลที่รับและส่งในการทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมโหมด 1

โหมด 2 จะมีการรับและส่งข้อมูลครั้งละ 11 บิต ข้อมูลจะถูกส่งออกภายนอกผ่านทางขา TXD และรับเข้ามาผ่านทางขา RXD ข้อมูลที่รับและส่งทั้ง 11 บิต ประกอบด้วยบิตเริ่มต้นของข้อมูล 1 บิต (มีค่าเป็น 0 เสมอ) บิตข้อมูล 8 บิต (รับและส่งบิตต่ำสุดก่อน) ตามด้วยบิตที่ 9 (ต่อจากบิตข้อมูลบิตสุดท้าย) ซึ่งเป็นบิตที่สามารถกำหนดให้มีค่าเป็น 0 หรือ 1 ก็ได้ และบิตสุดท้ายคือบิตสิ้นสุดของข้อมูล (มีค่าเป็น 1 เสมอ) ดังนั้นจำนวนบิตที่รับส่งทั้งหมด 11 บิต จะประกอบด้วยบิตต่างๆดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 แสดงข้อมูลที่รับและส่งในการทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมโหมด 2 และ 3

ในขณะทำการส่งข้อมูล บิตที่ 9 จะได้จากค่าในบิต TB8 ของรีจิสเตอร์ใช้งานเฉพาะ SCON ดังแสดงในรูปที่ 3.2 บิตนี้สามารถถูกกำหนดให้มีค่าเป็น 0 หรือ 1 อย่างไรก็ได้ ส่วนใหญ่ในการใช้งานจริงมักจะใช้บิตนี้สำหรับตรวจสอบความถูกต้องของข้อมูลที่รับส่ง (parity bit) โดยจะนำบิต P (parity) ในรีจิสเตอร์ PSW ไปไว้ในบิต TB8 ส่วนในขณะรับข้อมูลบิตที่ 9 จะไปปรากฏอยู่ในบิต RB8 ของรีจิสเตอร์ SCON โดยไม่สนใจบิตสิ้นสุดของข้อมูลค่าอัตราเร็วในการรับส่งข้อมูลในโหมดนี้ถูกกำหนดไว้ที่ 1/32 หรือ 1/64 ของความถี่ออสซิเลเตอร์ที่ใช้

โหมด 3 การทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมแบบสุดท้าย คือการทำงานในโหมด 3 ในการทำงานโหมดนี้ข้อมูลจำนวน 11 บิตถูกส่งผ่านขา TXD และถูกรับเข้ามาทางขา RXD ข้อมูลทั้ง 11 บิตประกอบด้วยบิตเริ่มต้นของข้อมูล 1 บิต (เป็น 0 เสมอ) บิตข้อมูล 8 บิต (รับและส่งบิตต่ำสุดก่อน) ตามด้วยบิตที่ 9 และบิตสุดท้ายคือบิตสิ้นสุดของข้อมูล (เป็น 1 เสมอ) อัตราเร็วในการรับส่งข้อมูลสามารถเปลี่ยนแปลงได้ ดังนั้น จะเห็นว่ารูปแบบการรับส่งข้อมูลในโหมด 3 จะเหมือนกับโหมด 2 ทุกอย่างแต่ในโหมดนี้สามารถกำหนดค่าอัตราเร็วในการรับหรือส่งข้อมูลได้ตามความต้องการของผู้ใช้

การทำงานของพอร์ตสื่อสารข้อมูลอนุกรมทั้ง 4 โหมดที่กล่าวมานี้การส่งข้อมูลจะเริ่มทันทีเมื่อมีคำสั่งใดๆที่ใช้รีจิสเตอร์ใช้งานเฉพาะ SBUF เป็นรีจิสเตอร์ปลายทาง (destination regiser) เช่น MOV SBUF , A

ส่วนในการรับข้อมูลจะเริ่มต้นโดยมีเงื่อนไขดังนี้

-ในโหมด 0 เริ่มเมื่อมีค่าในบิต RI = 0 และบิต REN = 1

-ในโหมดอื่นการรับข้อมูลเริ่มเมื่อMCS-51 ได้รับบิตเริ่มต้นของข้อมูลเข้ามาโดยที่บิต REN ในขณะนั้นต้องมีค่า 1

3.2 การสื่อสารระหว่างไมโครคอนโทรลเลอร์หลายตัว

การทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมในโหมด 2 และ 3 ของ MCS-51 จะมีรูปแบบการใช้งานพิเศษนอกเหนือจากการรับส่งข้อมูลธรรมดาตามที่ได้กล่าวมาแล้ว นั่นคือการใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พอร์ตสื่อสารข้อมูลแบบอนุกรมในการติดต่อสื่อสารระหว่างซีพียูด้วยกันเอง(multiprocessor communication) ดังมีรายละเอียดดังต่อไปนี้

ข้อมูลที่ใช้รับหรือส่งในการใช้พอร์ตสื่อสารข้อมูลแบบอนุกรมโหมด 2 หรือโหมด 3 เพื่อติดต่อระหว่างซีพียูด้วยกันจะมีจำนวนทั้งสิ้น 11 บิต การกำหนดให้พอร์ตสื่อสารข้อมูลแบบอนุกรมใช้ติดต่อสื่อสารระหว่างซีพียูด้วยกันเองสามารถกำหนดได้จากบิต SM2 ในรีจิสเตอร์ใช้งานเฉพาะ SCON ดังรูปที่ 3.4

บิตที่ 9 ที่รับเข้ามาในการทำงานของพอร์ตสื่อสารข้อมูลอนุกรมทั้งสองโหมดจะถูกนำไปไว้ในบิต RB8 ของรีจิสเตอร์ใช้งานเฉพาะ SCON ตามด้วยบิตสิ้นสุดของข้อมูลเหมือนการรับส่งข้อมูลทั่วไปที่กล่าวมาแล้ว ถ้าบิต SM2 ถูกเซต (เลือกให้พอร์ตสื่อสารข้อมูลแบบอนุกรมใช้ติดต่อระหว่างซีพียูด้วยกันเอง) และบิตสิ้นสุดของข้อมูลถูกรับเข้ามาแล้ว หากบิต RB8 (บิตที่ 9 ที่รับเข้ามา) มีค่าเป็น 1 จะส่งผลไปกระตุ้นให้วงจรส่วนควบคุมการอินเตอร์รัพท์ของพอร์ตสื่อสารข้อมูลแบบอนุกรมเริ่มทำงานเพื่ออินเตอร์รัพท์ซีพียูต่อไป หากบิต RB8 มีค่าเป็น 0 วงจรส่วนควบคุมการอินเตอร์รัพท์ของพอร์ตสื่อสารข้อมูลแบบอนุกรมจะไม่ทำงานแต่อย่างใด รายละเอียดการทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมที่ใช้ในการติดต่อระหว่างซีพียูด้วยกันเองมีดังนี้

เมื่อไมโครคอนโทรลเลอร์ตัวแม่หรือตัวหลัก (mastr processor) ต้องการส่งข้อมูลจำนวนหนึ่งไปยังไมโครคอนโทรลเลอร์ตัวลูก (slave) ตัวใดตัวหนึ่งจากที่มีหลายตัวในระบบ ขั้นแรกไมโครคอนโทรลเลอร์ตัวแม่จะต้องส่งข้อมูลขนาด 1 ไบต์ ที่มีชื่อเฉพาะเรียกว่า " แอดเดรสไบต์ " (address byte) ค่าแอดเดรสไบต์นี้จะเป็นค่าที่ระบุหมายเลขประจำตัวหรือตำแหน่งของไมโครคอนโทรลเลอร์ตัวลูกในระบบที่เป็นเป้าหมายของไมโครคอนโทรลเลอร์ตัวแม่ต้องการติดต่อด้วย ค่าแอดเดรสไบต์ที่ไมโครคอนโทรลเลอร์ตัวแม่ส่งไปมีข้อแตกต่างจากข้อมูลทั่วไปที่รับส่งกันจริงระหว่างไมโครคอนโทรลเลอร์ที่มีชื่อเรียกเฉพาะว่า " ดาต้าไบต์ " (data byte) ดังนี้

- แอดเดรสไบต์ : บิตที่ 9 จะเป็น 1
- ดาต้าไบต์ : บิตที่ 9 จะเป็น 0

หากในไมโครโปรเซสเซอร์ตัวลูกมีการเซ็ทบิต SM2 = 1 แล้ว (ใช้พอร์ตสื่อสารข้อมูลแบบอนุกรมใน MCS-51 ติดต่อระหว่างซีพียูด้วยกันเอง) ถ้าข้อมูลที่รับเข้ามาเป็นดาต้าไบต์จะไม่สามารถอินเตอร์รัพท์ซีพียูได้ แต่หากข้อมูลที่ได้รับเป็นแอดเดรสไบต์ไมโครคอนโทรลเลอร์ตัวลูกทุกตัวจะถูกอินเตอร์รัพท์ การทำเช่นนี้เพื่อที่ไมโครคอนโทรลเลอร์ตัวลูกทุกตัวที่เชื่อมต่อกับตัวแม่จะสามารถตรวจสอบแอดเดรสไบต์ที่ได้รับเข้ามาว่ามีค่าตรงกับหมายเลขตำแหน่งของตัวเองหรือไม่ หากไมโครคอนโทรลเลอร์ตัวลูกตัวใดมีค่าหมายเลขตำแหน่งของตัวเองตรงกับแอดเดรสไบต์ที่ได้รับเข้ามาก็จะเคลียร์บิต SM2 และเตรียมรับดาต้าไบต์ซึ่งจะตามเข้ามาภายหลังจากที่ได้รับแอดเดรสไบต์เรียบ

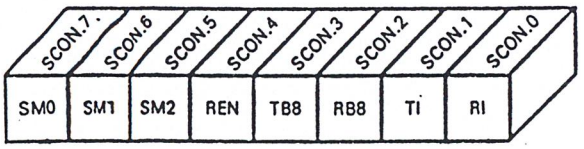
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ร้อยแล้ว สำหรับไมโครคอนโทรลเลอร์ตัวลูกอื่นๆที่ตรวจสอบแอดเดรสไบต์ดูแล้ว และปรากฏว่าไม่ตรงกับหมายเลขตำแหน่งของตัวเองจะยังคงปล่อยให้บิต SM2 ถูกเซ็ตต่อไป และกลับไปทำงานเดิมที่ค้างอยู่ก่อนได้รับการอินเตอร์รัพท์ต่อโดยไม่สนใจดาต้าไบต์ซึ่งตามเข้ามาหลังแอดเดรสไบต์ นั้นคือหากไมโครคอนโทรลเลอร์ตัวลูกตัวใดตรวจสอบพบว่าแอดเดรสไบต์ที่ได้รับเข้ามาไม่ตรงกับค่าตำแหน่งของตัวเอง มันก็จะไม่สนใจดาต้าไบต์ที่ส่งเข้ามาตามหลังแอดเดรสไบต์เลย แต่ในขณะที่ไมโครคอนโทรลเลอร์ตัวแม่ส่งแอดเดรสไบต์มายังไมโครคอนโทรลเลอร์ตัวลูก ไมโครคอนโทรลเลอร์ตัวลูกจะถูกอินเตอร์รัพท์เพื่อตรวจสอบค่าแอดเดรสไบต์ดูว่ามีค่าตรงกับตำแหน่งของตัวเองหรือไม่ทุกครั้ง

บิต SM2 จะไม่มีผลในการทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรม์โหมด 0 แต่การทำงานในโหมด 1 บิต SM2 สามารถถูกใช้เพื่อตรวจสอบบิตสิ้นสุดของข้อมูล (validity of the stop bit) โดยในการรับข้อมูลของโหมด 1 ถ้าบิต SM2 = 1 จะเป็นการตรวจสอบบิตสิ้นสุดของข้อมูลโดยหากบิตสิ้นสุดของข้อมูลมีค่าไม่เป็น 1 การอินเตอร์รัพท์ของพอร์ตสื่อสารข้อมูลแบบอนุกรมจะไม่เกิดขึ้น

3.3 รีจิสเตอร์ใช้งานเฉพาะ SCON (Serial Port Control Registor)

แต่ละบิตในรีจิสเตอร์ใช้งานเฉพาะ SCON จะใช้สำหรับควบคุมและตรวจสอบการทำงานช่องพอร์ตสื่อสารอนุกรมใน MCS-51 ดังนั้นก่อนการใช้งานพอร์ตสื่อสารข้อมูลแบบอนุกรม ผู้เขียนโปรแกรมจำเป็นต้องทราบความหมายของบิตต่างๆในรีจิสเตอร์ตัวนี้ แต่ละบิตในรีจิสเตอร์ใช้งานเฉพาะ SCON มีความหมายแสดงดังรูป 3.4 รีจิสเตอร์ใช้งานเฉพาะ SCON เข้าถึงข้อมูลได้ในระดับบิต รีจิสเตอร์ตัวนี้ไม่เพียงแต่ใช้ควบคุมการทำงานของพอร์ตสื่อสารแบบอนุกรม์ในโหมดต่างๆเท่านั้น บางบิตของรีจิสเตอร์นี้ยังใช้เป็นที่เก็บข้อมูลบิตที่ 9 สำหรับการรับและการส่งข้อมูลในโหมด 2 และ 3 (บิตTB8 และ RB8) และนอกจากนี้รีจิสเตอร์ใช้งานเฉพาะ SCON ยังมีบิตที่ควบคุมการอินเตอร์รัพท์ของพอร์ตสื่อสารข้อมูลแบบอนุกรม (serial port interrupt) นั่นคือบิต TI และ RI รวมอยู่ด้วย



บิต	ชื่อบิต	
SCON.7	SM0	บิตเลือกการทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมในโหมดต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บิต	ชื่อบิต	
SCON.6	SM1	บิตเลือกการทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมในโหมดต่างๆ 0 0 โหมด 0 : ทำงานเป็น shift register อัตราเร็วในการรับส่งข้อมูลเท่ากับ1/12 ของความถี่ออสซิลเลเตอร์ 0 1 โหมด 1 : 8 Bit UART อัตราเร็วในการรับส่งข้อมูลกำหนดเองได้ 1 0 โหมด 2 : 9 Bit UART อัตราเร็วในการรับหรือส่งข้อมูล = 1/32 หรือ 1/64 ของความถี่ออสซิลเลเตอร์ 1 1 โหมด 3 : 9 Bit UART อัตราเร็วในการรับหรือส่งข้อมูล กำหนดเองได้
SCON.5	SM2	บิตเลือกการใช้งานพอร์ตสื่อสารข้อมูลแบบอนุกรมในโหมด 2 และ 3 เพื่อใช้ติดต่อระหว่างซีพียูด้วยกันเอง 1 : ใช้พอร์ตสื่อสารข้อมูลแบบอนุกรมในการติดต่อระหว่างซีพียูด้วยกันเอง เมื่อข้อมูลบิตที่ 9 ที่ได้รับมีค่าเป็น 0 (ดาต้าไบต์) บิต RI จะไม่ถูกเซตแต่หากข้อมูลบิตที่ 9 มีค่าเป็น 1 (แอดเดรสไบต์) บิต RI จะถูกเซต 0 : ใช้พอร์ตสื่อสารข้อมูลแบบอนุกรมโหมด 1 และ โหมด 3 ตามปกติ ในการทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมโหมด 1 หากบิต SM2 ถูกเซต บิต RI จะไม่ถูกเซตจนกว่าบิตสิ้นสุดของข้อมูลจะถูกรับเข้ามา ในการทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมโหมด 0 บิตนี้ควรถูกเคลียร์ให้เป็น0
SCON.4	REN	บิตควบคุมการอนุญาตให้มีการรับข้อมูล ดังนี้ 1 : อนุญาตให้มีการรับข้อมูลจากภายนอกได้ 2 : ไม่อนุญาตให้มีการรับข้อมูลจากภายนอก
SCON.3	TB8	บิตข้อมูลบิตที่ 9 ซึ่งจะถูกส่งออกไปในการทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมโหมด 2 และ 3 การเซตหรือเคลียร์กระทำด้วยคำสั่งในโปรแกรมเท่านั้น
SCON.2	RB8	บิตข้อมูลบิตที่ 9 ทีได้รับเข้ามาจากภายนอกในการทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมโหมด 2 และ 3 ส่วนในการทำงานโหมด 1 ถ้าบิต SM2 = 0 บิตนี้จะเป็นบิตสิ้นสุดของข้อมูลที่รับเข้ามาได้และไม่ถูกกำหนดการใช้งานในโหมด 0
SCON.1	T1	บิตบอกสถานะสัญญาณอินเตอร์รัพท์ที่เกิดจากการส่งข้อมูล ถูกเซตโดยฮาร์ดแวร์เมื่อข้อมูลบิตที่ 8 ถูกส่งออกไปแล้วในการทำงานโหมด 0 ส่วนในการทำงานโหมดอื่นๆจะถูกเซตโดยฮาร์ดแวร์เมื่อเริ่มส่งบิตสิ้นสุดของข้อมูลออกไป และจะต้องถูกเคลียร์โดยคำสั่งในโปรแกรมเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บิต	ชื่อบิต	
SCON.0	RI	บิตบอกสถานะสัญญาณอินเตอร์รัพท์ที่เกิดจากการรับข้อมูล ถูกเซตโดยฮาร์ดแวร์เมื่อได้รับข้อมูลบิตที่ 8 เรียบร้อยแล้วในการทำงานโหมด 0หรือที่จุดครึ่งทางของช่วงรับบิตสิ้นสุดของข้อมูลในการทำงานโหมดอื่น (มีข้อยกเว้นในกรณีใช้พอร์ตสื่อสารข้อมูลแบบอนุกรมติดต่อระหว่างซีพียูด้วยกันเอง) และจะต้องถูกเคลียร์โดยคำสั่งในโปรแกรมเท่านั้น

รูปที่ 3.4 แสดงรีจิสเตอร์ใช้งานเฉพาะ SCON

3.4 อัตราเร็วในการรับและส่งข้อมูล

baud rate หมายความถึงอัตราเร็วในการรับหรือส่งข้อมูลโดยใน MCS-51 ค่าอัตราเร็วในการรับและส่งข้อมูลมีค่าเท่าใดก็ขึ้นอยู่กับการทำงานในแต่ละโหมดของพอร์ตสื่อสารข้อมูลแบบอนุกรมดังนี้

$$\text{baud rate} = \text{ความถี่ออสซิเลเตอร์ที่ใช้} / 12 \qquad (3.1)$$

หากใช้ครอสตอลความถี่ 12 เมกะเฮิรตซ์ ค่า baud rate ของพอร์ตสื่อสารข้อมูลแบบอนุกรมในโหมด 0 จะมีค่าสูงถึง 1 เมกะเฮิรตซ์

ในโหมด 2 ค่า baud rate ขึ้นอยู่กับค่าของบิต SMOD ที่อยู่ในรีจิสเตอร์ใช้งานเฉพาะ PCON โดย

บิต SMOD = 0 ค่า baud rate จะเป็น 1/64 ของความถี่ออสซิเลเตอร์ที่ใช้

บิต SMOD = 1 ค่า baud rate จะเป็น 1/32 ของความถี่ออสซิเลเตอร์ที่ใช้

หลังจากการรีเซต MCS-51 ค่าในบิต SMOD จะเป็น 0 เสมอและเราสามารถเขียนสูตรสำหรับคำนวณ baud rate ได้ดังสมการนี้

$$\text{baud rate โหมด 2} = [2^{(SMOD)} \times (\text{ความถี่ออสซิเลเตอร์})] / 64 \qquad (3.2)$$

หากใช้คริสตอลความถี่ 12 เมกะเฮริตซ์baud rateสูงสุดในการทำงานของโหมดนี้คือ 375K

- baud rate ในโหมด 1 และ 3 จะถูกกำหนดโดยอัตราการเกิด overflow ของไทม์เมอร์ 1 (timer 1 overflow rate) แต่ถ้าเป็น 8052 ซึ่งมีรีจิสเตอร์สำหรับใช้เป็นไทม์เมอร์หรือเคาน์เตอร์เพิ่มให้อีก 1 ตัว จะสามารถใช้ไทม์เมอร์ 2 ที่มีเพิ่มมานี้เป็นตัวกำหนด baud rate ได้ทำให้มีรีจิสเตอร์สำหรับใช้เป็นไทม์เมอร์หรือเคาน์เตอร์ที่สามารถนำมากำหนด baud rate รวมจำนวน 2 ตัว(ไทม์เมอร์ 1 และไทม์เมอร์ 2) โดยอาจใช้ตัวใดตัวหนึ่งในการกำหนด baud rate สำหรับการรับข้อมูล ส่วนอีกตัวหนึ่งกำหนด baud rate สำหรับหารส่งข้อมูล ทำให้การรับและการส่งข้อมูลมีค่า baud rate ต่างกันได้ รายละเอียดของการใช้ไทม์เมอร์ 1 หรือไทม์เมอร์ 2 เป็นตัวกำหนดค่า baud rate มีดังนี้

3.4.1 เมื่อใช้ไทม์เมอร์ 1 เป็นตัวกำหนด baud rate

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อไทม์เมอร์ 1 ถูกใช้เป็นตัวกำหนด baud rate สำหรับการทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมในโหมด 1 และ 3 ค่าของ baud rate ที่ได้จะถูกกำหนดด้วยอัตราการเกิด overflow ของไทม์เมอร์และขึ้นอยู่กับบิต SMOD ในรีจิสเตอร์ PCON ซึ่งเขียนเป็นสมการที่ใช้คำนวณหา baud rate ได้ดังนี้

baud rate โหมด 1 , 3 = $[2^{(SMOD)}$ x (อัตราการเกิด overflow ของไทม์เมอร์ 1)] / 32 (3.3)

เนื่องจากเมื่อเกิด overflow ในไทม์เมอร์ตัวใดจะทำให้เกิดสัญญาณอินเตอร์รัพท์เพื่อบอกให้ซีพียูทราบ ดังนั้นเมื่อเรานำไทม์เมอร์ 1 มาเป็นตัวกำหนด baud rate จึงควรห้ามการเกิดอินเตอร์รัพท์ขึ้นในระหว่างการรับหรือส่งข้อมูล และเนื่องจากตัวไทม์เมอร์ 1 เองยังสามารถถูกกำหนดให้ทำงานเป็นไทม์เมอร์หรือเคาน์เตอร์อย่างใดอย่างหนึ่ง ซึ่งมีโหมดการทำงานย่อยลงไปอี 4 โหมด ดังนั้นการใช้งานพอร์ตสื่อสารข้อมูลแบบอนุกรมจึงต้องทำความเข้าใจในเรื่องไทม์เมอร์และเคาน์เตอร์ให้ละเอียด

การใช้งานพอร์ตสื่อสารข้อมูลแบบอนุกรมที่บ่อยที่สุดนั้นไทม์เมอร์ 1 จะถูกกำหนดให้ทำงานเป็นไทม์เมอร์ในโหมด 2 (Auto-Reload) ในกรณีนี้ baud rate จะถูกกำหนดโดยสมการดังนี้

baud rate โหมด 1,3 = [2(SMOD) x (ความถี่ออสซิเลเตอร์ที่ใช้)] / [32 x12 x (256-THI)] (3.4)

ดังนั้นค่าที่ต้องโหลดไปไว้ยังรีจิสเตอร์ THI เพื่อให้ได้ค่า baud rat จะสามารถคำนวณได้ดังนี้

H1 = 256 - [{ 2(SMOD) x ความถี่ออสซเตอร์ที่ใช้} / (384 x baud rate)] (3.5)

เราสามารถที่จะสร้าง baud rate ค่าต่ำๆด้วยไทม์เมอร์ 1 ได้โดยปล่อยให้ไทม์เมอร์ 1 อินเตอร์รัพท์ซีพียูได้ และกำหนดการทำงานให้เป็นไทม์เมอร์ขนาด 16 บิต (โหมด 1) และใช้ไทม์เมอร์ 1 อินเตอร์รัพท์ซีพียูเพื่อโหลดค่าใหม่เองด้วยซอฟต์แวร์ขณะเกิด overflow เนื่องจากในการทำงานโหมด 1 ของไทม์เมอร์ 1 ไม่สามารถโหลดค่าใหม่เองได้ด้วยฮาร์ดแวร์ได้ (ไม่สามารถทำงานแบบ Auto-Reloaded) ตารางที่ 3.1 เป็นค่า baud rate ค่าต่างๆที่ใช้กันมากและบอกว่าสามารถใช้ได้จากไทม์เมอร์ 1 อย่างไร

ตารางที่ 3.1 ค่าที่ต้องนำไปไว้ในรีจิสเตอร์ของไทม์เมอร์ 1 เมื่อใช้ baud rate ค่าต่างๆ

Baud rate	ความถี่ของคริสตอล	บิต SMOD	ไทม์เมอร์ 1		
			C/T	โหมด	ค่าที่ใช้โหลด
Mode 0 :Max 1MHz	12MHz	X	X	X	X
Mode 2 :Max 375K	12MHz	1	X	X	X
Modes1,3 :62.5K	12MHz	1	0	2	FFH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

12.9K	11.059MHz	1	0	2	FDH
9.6K	11.059MHz	0	0	2	FDH
4.8K	11.059MHz	0	0	2	FAH
2.4K	11.059MHz	0	0	2	F4H
1.2K	11.059MHz	0	0	2	E8H
137.5K	11.059MHz	0	0	2	1DH
110K	6MHz	0	0	2	72H
110K	12MHz	0	0	1	FEEBH

จากตารางที่ 3.1 จะเห็นว่าในการทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมโหมด 0 จะมีความเร็วในการส่งมากที่สุดเมื่อเปรียบเทียบกับโหมดอื่นที่ความถี่คริสตอลค่าเดียวกัน และจะเห็นว่าหากเลือกใช้คริสตอลความถี่ 11.059 เมกะเฮิรตซ์ จะสามารถตั้งค่า baud rate ในโหมด 1 และ 3 เป็นค่ามาตรฐานที่ใช้กันทั่วไปได้เช่น 1200 , 2400 , 4800 , 9600 , 19200 จึงเป็นเหตุผลสำคัญที่ในระบบควบคุมส่วนใหญ่เลือกใช้คริสตอลความถี่ 11.059 เมกะเฮิรตซ์ มากกว่า 12 เมกะเฮิรตซ์

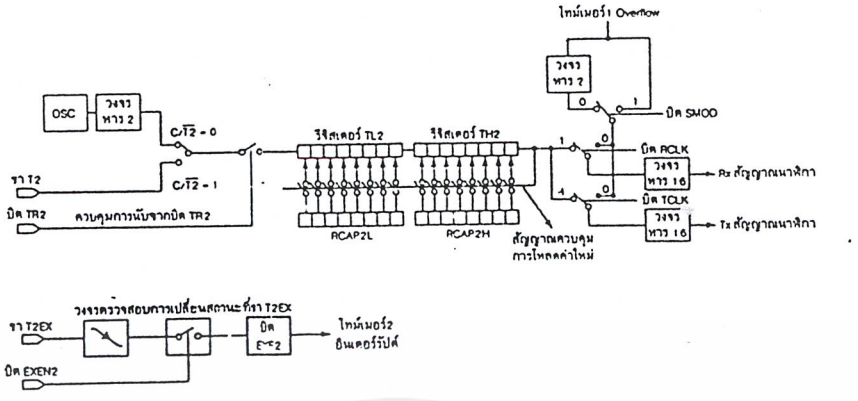
ในตารางที่ 3.1 นอกจากจะแสดงค่า baud rate ค่าต่างๆเปรียบเทียบให้เห็นแล้วตารางนี้ยังแสดงค่าที่ต้องโหลดไปไว้ในรีจิสเตอร์ใช้งานเฉพาะ THI ที่ค่า baud rate มาตรฐานต่างๆให้ทราบอีกด้วย ผู้เขียนโปรแกรมสามารถนำค่านี้ไปใช้ได้เลย แต่หากต้องการค่า baud rate อื่นๆที่นอกเหนือไปจากตารางนี้ จะต้องคำนวณค่าที่ต้องโหลดให้รีจิสเตอร์ใช้งานเฉพาะ THI จากสมการที่แสดงไปแล้ว

3.4.2 การใช้ไทม์เมอร์ 2 ในการกำหนด baud rate

สำหรับ MCS-51 เบอร์ที่มีไทมเมอร์ 2 เพิ่มให้ผู้ใช้อีก 1 ตัวเช่นเบอร์ 8052 ผู้ใช้สามารถนำไทม์เมอร์ 2 ที่มีเพิ่มขึ้นนี้มาเป็นตัวกำหนด baud rate โดยการเซตบิต TCLK และ/หรือบิต RCLK ในรีจิสเตอร์ใช้งานเฉพาะ T2CON บิต TCLK , RCLK จะเป็นสวิตช์เลือกระหว่างการใช้อัตราการเกิด overflow ของไทม์เมอร์ 2 หรือ ไทม์เมอร์ 1 เป็นตัวกำเนิด baud rate ดังนั้นในกรณีที่ใช้ 8052 ค่า baud rate สำหรับการส่งและการรับสามารถที่จะต่างกันได้ในเวลาเดียวกัน(ให้บิต RCLK มีค่าเป็น 1 ส่วน TCLK มีค่าเป็น 0 หรือกลับกัน) การกำหนดบิต RCLK และ/หรือ TCLK ให้เป็น 1 จะเป็นการกำหนดให้ไทม์เมอร์ 2 ถูกใช้เป็นตัวกำหนด baud rate ให้แก่พอร์ตสื่อสารอนุกรมดังรูปที่ 3.5

การใช้ไทม์เมอร์ 2 เป็นตัวกำหนด baud rate จะมีการทำงานเหมือนโหมด Auto-Reloaded ตรงที่การเกิด overflow ในรีจิสเตอร์ TH2 ทำให้รีจิสเตอร์ที่ประกอบขึ้นเป็นไทม์เมอร์ 2 (รีจิสเตอร์ใช้งานเฉพาะ TL2 และ TH2) ถูกโหลดด้วยค่าขนาด 16 บิตจากรีจิสเตอร์ใช้งานเฉพาะ RCAP2L และ RCAP2H ซึ่งต้องได้รับการตั้งค่าไว้ล่วงหน้าด้วยซอฟต์แวร์เรียบร้อยแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 ไทม์เมอร์ 2 ในการเป็นตัวกำหนด baud rate

เมื่อไทม์เมอร์ 2 ทำงานในโหมดนี้ค่า baud rate ของการรับหรือการส่งข้อมูลในโหมด 1 และ 3 จะถูกกำหนดโดยอัตราการเกิด overflow ของรีจิสเตอร์ไทม์เมอร์ 2 ตามสมการดังนี้

baud rate โหมด 1 ,3 = [อัตราการเกิด overflow ของไทมเมอร์ 2] / 16 (3.6)

เนื่องจากไทม์เมอร์ 2 สามารถถูกกำหนดการใช้งานเป็นไทม์เมอร์หรือเคาน์เตอร์อย่างใดอย่างหนึ่งในการใช้งานทั่วไปที่นำไทม์เมอร์ 2 มาใช้เป็นตัวกำหนด baud rate มักจะกำหนดให้ทำงานเป็นไทม์เมอร์ซึ่งมีรายละเอียดการทำงานแตกต่างจากการใช้ไทม์เมอร์ 2 เป็นไทมเมอร์ในโหมดอื่นเล็กน้อย กล่าวคือ ปกติเมื่อใช้ไทม์เมอร์ 2 เป็นไทม์เมอร์ในโหมดอื่นมันจะถูกเพิ่มค่าทุกแมชชีนไซเคิล แต่เมื่อใช้งานเป็นไทม์เมอร์ในโหมดที่ใช้สำหรับกำหนดค่า baud rate สำหรับพอร์ตสื่อสารข้อมูลแบบอนุกรมไทม์เมอร์ 2 จะถูกเพิ่มค่าทุกๆสเตทในแต่ละแมชชีนไซเคิล (ใน 1 แมชชีนไซเคิลมี 6 สเตท) เท่ากับความถี่ ½ ของความถี่ออสซิลเลเตอร์ดังนั้นสมการสำหรับหาค่า baud rate ของพอร์ตสื่อสารอนุกรมเมื่อใช้ไทม์เมอร์ 2 คือ

baud rate โหมด 1,3 = [ความถี่ออสซิลเลเตอร์ที่ใช้] / [32 x (65536-(RCAP2H ,RCAP2L)] (3.7)

RCAP2L , RCAP2H ในสมการคือค่าต้องโหลดไว้ในรีจิสเตอร์ใช้งานเฉพาะ RCAP2L และ RCAP2H ตามลำดับ ค่านี้ถูกกำหนดให้เป็นจำนวนเต็มที่ไม่คิดเครื่องหมายขนาด 16 บิต

ค่าที่ต้องโหลดไปไว้ในรีจิสเตอร์ RCAP2L และ RCAP2H เพื่อให้ได้ค่า baud rate ตามที่ต้องการสามารถคำนวณได้จากสูตรต่อไปนี้

(RCAP2L,RCAP2H) = 65536 – [(ความถี่ออสซิลเลเตอร์ที่ใช้) / (32 x baud rate)] (3.8)

สำหรับค่าที่ต้องโหลดไปไว้ในรีจิสเตอร์ RCAP2L และ RCAP2H เมื่อต้องการตั้งค่า baud rate มาตรฐานมีดังแสดงในตารางที่ 3.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไทม์เมอร์ 2 ในการทำงานเป็นตัวกำหนด baud rate มีรายละเอียดดังรูปที่ 3.5 จากรูปนี้จะเห็นได้ว่าไทม์เมอร์ 2 ถูกใช้เป็นตัวกำหนด baud rate ก็ต่อเมื่อบิต RCLK ,TCLK ตัวใดตัวหนึ่งมีค่าเป็น 1 (RCLK+TCLK = 1) ในรีจิสเตอร์ใช้งานเฉพาะ T2CON (บิตทั้งสองจะเป็นตัวเลือกระหว่างการใช้ไทม์เมอร์ 1 หรือ 2 เป็นตัวกำหนด baud rate) และเมื่อเปรียบเทียบกับรูปแสดงการทำงานของไทม์เมอร์ 2 จะสังเกตได้ว่าการเกิด overflow ในรีจิสเตอร์ TH2 จะไม่ทำให้ TF2 ถูกเซตดังนั้นการเกิด overflow ของไทม์เมอร์ 2 เมื่อใช้เป็นตัวกำหนด baud rate จะไม่ทำให้เกิดอินเตอร์รัพท์สำหรับไทม์เมอร์หรือเคาน์เตอร์ผู้ใช้จึงไม่จำเป็นต้องห้ามการเกิดอินเตอร์รัพท์ของไทม์เมอร์ 2 ในระหว่างที่ใช้งานเป็นตัวกำหนด baud rate และจากด้านล่างของรูปที่ 3.5 จะเห็นว่าถ้าบิต EXEN2 ถูกเซตเป็น 1 การเปลี่ยนสถานะของสัญญาณจาก 1 เป็น 0 (1 to 0 transition) ที่ขา T2EX ของ MCS-51 จะมีผลไปเซตบิต EXF2 ทำให้สามารถอินเตอร์รัพท์ซีพียูได้โดยจะไม่มีการโหลดค่าจากในรีจิสเตอร์ RCAP2L และ RCAP2H ไปยังรีจิสเตอร์ของไทม์เมอร์ 2 เลย ทั้งนี้เพื่อให้ระหว่างการใช้ไทม์เมอร์ 2 เป็นตัวกำหนด baud rate ยังสามารถนำขา T2EX มาใช้รับสัญญาณอินเตอร์รัพท์จากภายนอกของไทม์เมอร์ 2 ได้ถ้าต้องการ จากรูปที่ 5.5 เห็นว่าบิต SMOD ไม่มีผลในการเพิ่มค่า baud rate เมื่อใช้ไทม์เมอร์ 2 เป็นตัวกำหนด baud rate

สิ่งหนึ่งที่ควรระลึกไว้เสมอในการใช้งานไทม์เมอร์ 2 เป็นตัวกำหนด baud rate นั่นคือเมื่อไทม์เมอร์ 2 กำลังเป็นไทม์เมอร์ (บิต RCLK + TCLK = 1และบิต TR2 = 1) เพื่อกำหนด baud rate ผู้ใช้ไม่ควรพยายามอ่านหรือเขียนข้อมูลใดๆลงไปยังรีจิสเตอร์ของไทม์เมอร์ 2 ทั้งนี้เพราะภายใต้สภาพการทำงานเช่นนี้รีจิสเตอร์ของไทม์เมอร์ 2 จะถูกเพิ่มค่าทุกๆสเตทในแต่ละแมชชีนไซเคิล ทำให้ข้อมูลที่ได้จากการอ่านหรือเขียนไม่เที่ยงตรง แต่เราอาจเขียนหรืออ่านข้อมูลจากรีจิสเตอร์ RCAP ได้ถึงแม้ว่าการเขียนข้อมูลไปยังรีจิสเตอร์ทั้งสองนี้อาจทำให้เกิดการผิดพลาดได้หากข้อมูลที่เขียนลงไปตรงกับช่วงเวลาการโหลดค่าใหม่จากรีจิสเตอร์นี้ไปยังรีจิสเตอร์ของไทม์เมอร์ 2 ครั้งต่อไปพอดี ดังนั้นในกรณีที่ผู้ใช้ต้องการเปลี่ยนแปลงค่าใดๆในรีจิสเตอร์ของไทม์เมอร์ 2 และรีจิสเตอร์ RCAP ควรจะเคลียร์บิต TR2 เพื่อให้ไทม์เมอร์ 2 หยุดทำงานเสียก่อน จากนั้นจึงสามารถกระทำการใดๆกับรีจิสเตอร์ทั้งสองตัวนี้ได้ตามต้องการ

ตารางที่ 3.2 แสดงค่าที่ต้องโหลดไปไว้ในรีจิสเตอร์ RCAP2L และ RCAP2H ของไทม์เมอร์ 2 เพื่อให้ได้ค่า baud rate มาตรฐาน

Baud rate	ความถี่ออสซิเลเตอร์	ไทม์เมอร์ 2	
		RCAP2H	RCAP2L
375K	12MHz	FF	FF
9.6K	12MHz	FF	D9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.8K	12MHz	FF	B2
2.4K	12MHz	FF	64
1.2K	12MHz	FE	C8
300	12MHz	FB	1E
110	12MHz	F2	AF
300	6MHz	FD	8F
110	6MHz	F9	57

3.5 การทำงานพอร์ตสื่อสารข้อมูลแบบอนุกรมโหมด 2 และ 3

การทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมในโหมด 2 และ 3 มีการรับส่งข้อมูลครั้งละ 11 บิต โดยส่งข้อมูลผ่านขา TXD และรับสัญญาณข้อมูลผ่านทางขา RXD ข้อมูลทั้ง 11 บิตจะประกอบไปด้วยบิตเริ่มต้นของข้อมูล 1 บิต (0) บิตข้อมูล 8 บิต (รับและส่งบิตต่ำสุดก่อน) บิตที่ 9 ซึ่งสามารถกำหนดค่าได้เอง 1 บิต (1 programmable 9th bit) และบิตสิ้นสุดของข้อมูล 1 บิต (1) ดังรูปที่ 3.3

บิตข้อมูลที่มีเพิ่มขึ้นมาในโหมดนี้คือบิตที่9ซึ่งสามารถกำหนดค่าล่วงหน้าได้ด้วยซอฟต์แวร์ในการส่งข้อมูล บิตที่ 9 (บิตTB8 ในรีจิสเตอร์ SCON) สามารถถูกกำหนดให้มีค่าเป็น 0 หรือ 1 ได้ส่วนในการรับข้อมูล บิตที่ 9 จะถูกนำไปเก็บไว้ที่บิต RB8 ในรีจิสเตอร์ใช้งานเฉพาะ SCON

ในการทำงานโหมด 2 ค่า baud rate สามารถกำหนดให้เป็น 1/32 หรือ 1/64 ของความถี่ออสซิลเลเตอร์ที่ใช้ ส่วนในโหมด 3 ค่า baud rate สามารถแปรค่าได้ โดยถูกกำหนดด้วยไทม์เมอร์1 หรือ ไทม์เมอร์ 2 อย่างใดอย่างหนึ่ง การเลือกให้ไทม์เมอร์ 1 หรือ 2 เป็นตัวกำหนด baud rate ควบคุมได้จากบิต RCLK และ TCLK ในรีจิสเตอร์ใช้งานเฉพาะ T2CON

การส่งเริ่มต้นด้วยคำสั่งใดๆที่ใช้รีจิสเตอร์ใช้งานเฉพาะ SBUF เป็นรีจิสเตอร์ปลายทางสัญญาณ write to SBUF จะนำค่าในบิต TB8 (มีค่า 0 หรือ 1 ตามการกำหนดล่วงหน้าของผู้ใช้) ไปไว้ที่ตำแหน่งบิตที่ 9 ของรีจิสเตอร์สำหรับส่งข้อมูล (บิตที่ 9 ในรีจิสเตอร์สำหรับส่งข้อมูลคือ D-flipfloop) และส่งสัญญาณไปบอกวงจรควบคุมการส่งข้อมูล (TX control block) ว่าขณะนี้กำลังต้องการส่งข้อมูลการส่งข้อมูลจะเริ่มต้นที่สแตก 1 เฟส 1 ของแมชชีนไซเคิลถัดจากการเกิด overflow ในเคาน์เตอร์หาร 16 ครั้งถัดไป (ดังนั้นช่วงเวลาในการส่งแต่ละบิตจะซิงโครไนซ์กับเคาน์เตอร์หาร 16 ไม่ใช่กับสัญญาณ write to SBUF)

การส่งข้อมูลเริ่มต้นด้วยการกระตุ้นให้สัญญาณ SEND แอกตีฟ ซึ่งจะเริ่มใส่ค่าบิตเริ่มต้นข่องข้อมูลไปที่ขา TXD เมื่อช่วงเวลาการส่งบิตแรกผ่านไป สัญญาณ DATA จะเริ่มแอกตีฟทำให้ค่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของบิตที่ตำแหน่งเอาต์พุตของรีจิสเตอร์สำหรับส่งข้อมูลถูกส่งไปที่ขา TXD และพัลส์ของสัญญาณ shift ลูกแรกเกิดขึ้น (ดังรูปที่ 3.7) เมื่อเวลาในการส่งผ่านไป 1 บิต พัลส์ของสัญญาณ shift นี้จะไปทำให้ 1 ถูกนำไปไว้ในตำแหน่งบิตที่ 9 (ต่อไปจะกลายเป็นบิตสิ้นสุดของข้อมูล)ของรีจิสเตอร์สำหรับส่งข้อมูล (ตำแหน่งบิตที่ 9 คือ D-flipflop) หลังจากนั้นจะมีเพียงศูนย์ที่ถูกใส่เข้าไปในรีจิสเตอร์สำหรับส่งข้อมูล ดังนั้นขณะที่บิตข้อมูลถูกเลื่อนออกไปทางขวา 0 จะถูกเลื่อนเข้าไปทางซ้าย การส่งจะดำเนินไปเรื่อยๆจนกระทั่งบิต TB8 ซึ่งถูกใส่ไว้ตำแหน่งบิตที่ 9 ของรีจิสเตอร์สำหรับส่งข้อมูลในตอนแรก ถูกเลื่อนไปอยู่ตำแหน่งเอาต์พุตของรีจิสเตอร์สำหรับส่งข้อมูล และทางด้านซ้ายของบิต TB8 นี้จะมีค่า 1 (บิตสิ้นสุดของข้อมูล) จากการโหลดด้วยพัลส์ของสัญญาณ shift ครั้งแรก ส่วนตำแหน่งอื่นที่เหลือทางด้านซ้ายจะมีค่าเป็นศูนย์หมด สภาวะที่มีศูนย์อยู่ที่ด้านซ้ายของบิตสิ้นสุดของข้อมูลนี้จะถูกตรวจพบได้ด้วยวงจรตรวจจับ 0 (zero detector) ซึ่งเมื่อตรวจจับได้ก็จะส่งสัญญาณไปบอกวงจรควบคุมการส่งข้อมูลให้ทำการเลื่อนข้อมูลครั้งสุดท้ายอีก 1 ครั้ง แล้วจึงค่อยให้สัญญาณ SEND หยุดแอกตีฟรวมทั้งเซตบิต T1 เป็นลำดับสุดท้ายเหตุการณ์ที่เกิดขึ้นหลังจากตรวจจับศูนย์ได้ 7 บิตนี้จะเกิดขึ้นขณะที่มีการ overflow ครั้งที่ 11 ของเคาน์เตอร์หาร 16 หลังจากสัญญาณ write to SBUF แอกตีฟ

การรับข้อมูลเริ่มต้นเมื่อมีการตรวจพบการเปลี่ยนสถานะจาก 1 เป็น 0 ที่ขา RXD การตรวจสอบสถานะสัญญาณที่ขา RXD จะกระทำด้วยความถี่ 16 เท่าของค่า baud rate ในขณะนั้นจากเคาน์เตอร์หาร 16 จะถูกรีเซตทันทีและค่า 1FFH จะถูกโหลดไปที่รีจิสเตอร์สำหรับรับข้อมูล

ช่วงเวลาในการตรวจสอบข้อมูลภายนอกที่ขา RXD แต่ละบิตจะถูกแบ่งออกเป็น 16 สเตท ย่อยการตรวจสอบจะกระทำที่สเตทที่ 7,8,9 วงจรตรวจสอบบิตข้อมูลจะตรวจสอบค่าสัญญาณที่ขา RXD ซึ่งค่าที่ถูกรับเข้ามาเป็นข้อมูลจะต้องตรวจพบอย่างน้อย 2 ครั้งจากการตรวจสอบทั้งหมด 3 ครั้ง ทั้งนี้เพื่อป้องกันสัญญาณรบกวนที่อาจเกิดขึ้น แต่ถ้าบิตแรกที่ถูกรับเข้ามาถูกตรวจสอบพบว่าไม่เป็น 0 วงจรควบคุมการรับข้อมูลจะถูกรีเซต และกลับไปเริ่มต้นตรวจการเปลี่ยนสถานะที่ขา RXD ตามเดิมทั้งนี้เพื่อให้แน่ใจว่าบิตแรกที่เข้ามาเป็นบิตเริ่มต้นของข้อมูลแน่นอนนั่นเอง หากบิตแรกตรวจพบว่ามีค่าเป็น 0 นั่นคือบิตนี้เป็นบิตเริ่มต้นของข้อมูลแน่นอน การรับข้อมูลจะเริ่มขึ้น โดยเลื่อนบิตเริ่มต้นของข้อมูลที่รับเข้ามาได้ไปไว้ที่รีจิสเตอร์สำหรับรับข้อมูล

ขณะที่บิตข้อมูลถูกเลื่อนเข้ามาทางขวาในรีจิสเตอร์สำหรับรับข้อมูลจะทำให้ 1FFH ซึ่งถูกนำไปไว้ที่รีจิสเตอร์สำหรับรับข้อมูลในตอนแรกถูกเลื่อนออกไปทางซ้าย เมื่อบิตเริ่มต้นของข้อมูล (0) ถูกเลื่อนไปถึงตำแหน่งซ้ายสุดในรีจิสเตอร์สำหรับรับข้อมูล (ขนาด 9 บิต) จะมีสัญญาณส่งไปบอกวงจรควบคุมการรับข้อมูลทำให้การเลื่อนข้อมูลเข้ามาเป็นครั้งสุดท้ายอีก1ครั้งแล้วจึงส่งสัญญาณเพื่อนำข้อมูลจากรีจิสเตอร์สำหรับรับข้อมูลมาไว้ที่รีจิสเตอร์ใช้งานเฉพาะ SBUF และนำบิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

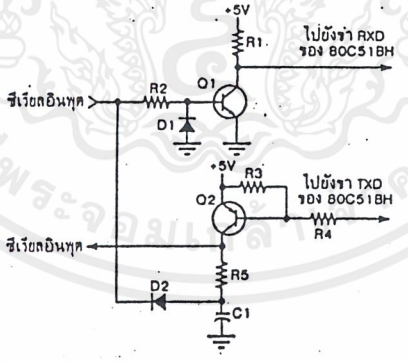
ที่ถูกรับเข้ามาครั้งสุดท้ายไปไว้ในบิต RB8 รวมทั้งเซตบิต RI เหตุการณ์ทั้งหมดนี้เกิดขึ้นก็ต่อเมื่อเป็นไปตามเงื่อนไขต่อไปนี้เมื่อพัลส์ของสัญญาณ shift ลูกสุดท้ายถูกสร้างขึ้น

- บิต RI = 0
- บิต SM2 = 0 หรือบิตที่ 9 ที่รับเข้ามามีค่าเป็น 1

ถ้ามีเงื่อนไขใดไม่ตรงตามนี้ ข้อมูลไบต์ที่ถูกรับเข้ามาทั้งหมดจะหายไปโดยไม่สามารถเรียกคืนกลับมาได้ และบิต RI จะไม่ถูกเซต แต่ถ้าตรงตามเงื่อนไขทั้งสองอย่างนี้บิตที่ 9 ที่รับเข้ามาจะถูกนำไปไว้ในบิต RB8 ของรีจิสเตอร์ SCON ส่วนบิตข้อมูลทั้ง 8 จะถูกนำไปไว้ในรีจิสเตอร์ใช้งานเฉพาะ SBUF ในช่วงเวลาของการรับข้อมูล 1 บิตต่อมา ไม่ว่าเงื่อนไขทั้งสองจะเป็นอย่างไร ระบบการรับข้อมูลก็จะกลับไปตรวจหาการเปลี่ยนสถานะจาก 1 เป็น 0 ที่ขา RXD ต่อ (รอรับบิตเริ่มต้นของข้อมูลไบต์ถัดไป)

สังเกตว่าค่าของบิตสิ้นสุดของข้อมูลที่ถูกรับเข้ามาจะไม่เกี่ยวข้องกับรีจิสเตอร์ SBUF บิต RB8 หรือ RI แต่อย่างใดเลย

ในกรณีที่ใช้พอร์ตสื่อสารข้อมูลแบบอนุกรมเพื่อเชื่อมต่อกับวงจรภายนอกตามมาตรฐาน RS-232 จำเป็นต้องมีวงจรเพิ่มเติมเพื่อช่วยยกระดับแรงดันให้ได้มาตรฐาน ทั้งนี้เพราะการติดต่อสื่อสารแบบอนุกรมตามมาตรฐาน RS-232 ใช้ระดับแรงดัน –12โวลต์ วงจรสำหรับเชื่อมต่อพอร์ตสื่อสารข้อมูลแบบอนุกรมของ MCS-51 ตามมาตรฐาน RS-232 ดังรูปที่ 3.6



รูปที่ 3.6 แสดงวงจรการเชื่อมต่อพอร์ตสื่อสารข้อมูลแบบอนุกรมของ MCS-51 ตามมาตรฐาน RS-232

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6 การสร้างระบบเวลาจริง(Real Time Clock) ให้แก่ MCS-51

ในระบบที่ใช้ไมโครคอนโทรเลอร์เป็นตัวควบคุม ส่วนใหญ่ต้องใช้เวลาเป็นตัวกำหนดการทำงาน เช่น การกำหนดการทำงานให้เครื่องจักรเปิดหรือปิดในเวลาที่กำหนด การบันทึกเวลาการใช้งานเครื่องจักร ดังนั้นหากในระบบควบคุมที่ใช้ไมโครคอนโทรลเลอร์มีวงจรกำเนิดฐานเวลาร่วมอยู่ด้วย จะทำให้การออกแบบระบบและการเขียนโปรแกรมควบคุมสะดวกมากขึ้น

MCS-51 มีรีจิสเตอร์ที่ทำหน้าที่เป็นได้ทั้งไทม์เมอร์หรือเคาน์เตอร์ ซึ่งผู้ใช้สามารถเขียนโปรแกรมให้ทำงานเป็นตัวสร้างฐานเวลาแก่ซีพียูได้ แต่เนื่องจากความไม่สะดวกในการใช้งานจริงที่ต้องการทราบเวลาปัจจุบัน หรือในงานที่ต้องการสัญญาณอินเตอร์รัปต์ในช่วงเวลาที่มากหรือน้อยกว่าที่ไทม์เมอร์สร้างได้ ทั้งนี้เนื่องจากต้องคำนวณคาบเวลาให้ถูกต้อง และต้องมีการตั้งค่าใหม่ทุกครั้งที่เริ่มทำงาน หรือหากการทำงานเกิดผิดพลาดเวลาก็จะผิดตามไปด้วย ดังนั้นในบางครั้งในการออกแบบระบบควบคุม จำเป็นจะต้องมีวงจรซึ่งทำหน้าที่สร้างฐานเวลาให้แก่ MCS-51 โดยทำงานเป็นอิสระแยกจากไมโครคอนโทรลเลอร์วิธีหนึ่งที่จะสร้างฐานเวลาจริงให้แก่ไมโครคอนโทรลเลอร์โดยทำงานอยู่ตลอดเวลาแม้ขณะไม่มีพลังงาน ด้วยการใช้แบตเตอรี่สำรอง วิธีนี้จะใช้ชิปทำหน้าที่สร้างฐานเวลาจริงให้แก่ระบบ (Real Time Clock) ซึ่งทำให้ระบบมีความเที่ยงตรงในการทำงานมากขึ้น และทำให้มีไทม์เมอร์หรือเคาน์เตอร์เหลือไว้ใช้งานอย่างอื่นด้วย

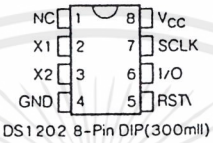
3.6.1 การสร้างฐานเวลาจริงให้แก่ MCS-51 โดยใช้ RTC (Real time Clock)

วิธีนี้จะใช้ชิปRTCเบอร์DS1302 ทำหน้าที่เป็นนาฬิกาที่สามารถส่งข้อมูลเวลาใดๆให้แก่ไมโครคอนโทรลเลอร์ได้ เวลาที่ได้จากวิธีนี้จะเป็นเวลาจริงๆที่เดินอยู่ตลอดเวลาอย่างเที่ยงตรง สามารถนำมาต่อร่วมกับระบบเพื่อบอกเวลาให้กับไมโครคอนโทรลเลอร์ได้สะดวกเพราะใช้จำนวนสายในการติดต่อระหว่างตัวชิปเองกับไมโครคอนโทรลเลอร์เพียง 3 เส้น เนื่องจากชิป RTC ใช้การติดต่อรับส่งข้อมูลแบบอนุกรม คุณสมบัติของชิปRTC มีดังนี้

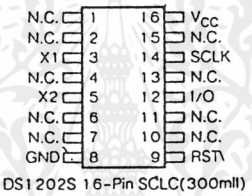
- ทำหน้าที่นับวินาที นาที ชั่วโมง วันที่ของเดือน เดือน ปี รวมทั้งคำนวณปีอธิกสุรทินให้เองโดยอัตโนมัติ
- มีหน่วยความจำขนาด 24 ไบต์สำหรับเก็บข้อมูลทั่วๆไป ส่วนใหญ่ไว้เก็บข้อมูลที่ต้องการสำรองในขณะที่ไม่มีพลังงานจ่ายให้แก่ระบบ ทำให้ไม่จำเป็นต้องสำรองหน่วยความจำทั้งระบบ
- ใช้การติดต่อแบบอนุกรม จึงใช้จำนวนสายในการเชื่อมต่อกับระบบเพียง 3 เส้นเท่านั้น
- ใช้แรงดันไฟฟ้า 2.0 ถึง 5.5 โวลต์ และใช้กระแสเพียง 300 นาโนแอมแปร์ที่แรงดัน 2 โวลต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การโอนย้ายข้อมูลสามารถกระทำได้ทั้งในแบบครั้งละ 1 ไบต์(Single Byte) หรือครั้งละหลายๆไบต์
- ตัวชิปเองมีให้เลือกทั้งแบบ 8 PIN DIP หรือ 16 PIN SOIC เพื่อใช้สำหรับแผ่นวงจรชนิด surface mount
- ระดับสัญญาณ TTL compatible (Vcc = 5 โวลต์)
- ช่วงอุณหภูมิในการใช้งานกว้างมากระหว่าง –40 องศา ±88องศาเซลเซียส



DS1202 8-Pin DIP(300mil)



DS1202S 16-Pin SCLC(300mil)

รูปที่ 3.7 แสดงลักษณะรูปร่างทั้งสองแบบของไอซีเบอร์ DS1202

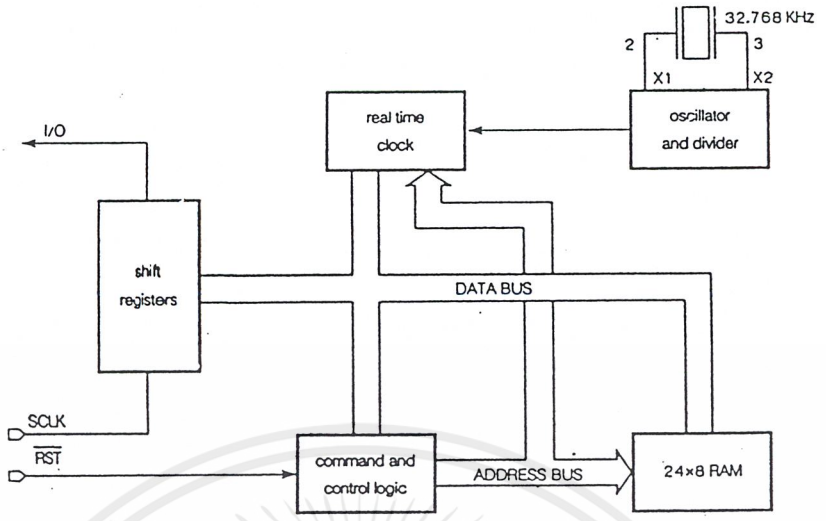
วันที่ในวันสุดท้ายของเดือนจะถูกปรับโดยอัตโนมัติสำหรับเดือนที่มีจำนวนวันน้อยกว่า 31 วัน และมีการคำนวณจำนวนวันของเดือนกุมภาพันธ์ในปีอธิกสุรทินให้เอง ข้อมูลที่ส่งให้แก่ไมโครคอนโทรลเลอร์สามารถเลือกรูปแบบได้ทั้งแบบ 24 ชั่วโมง (0.00-23.59) หรือแบบ 12 ชั่วโมง (0.00-12.00) การเชื่อมต่อ RTC เข้ากับระบบใช้สายจำนวน 3 เส้นเพราะใช้การติดต่อแบบอนุกรมชนิด Synchronous Serial Communication ขาที่ต้องใช้ในการรับส่งข้อมูลทั้ง 3 คือ

- RST (reset)
- I / O (data line)
- SCLK (serial clock)

ชิป RTC ประกอบไปด้วยส่วนที่สำคัญดังนี้คือ

- shift register
- control logic
- oscillator
- real time clock และ
- RAM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8โครงสร้างภายในของ RTC DS1202

ในการรับหรือส่งข้อมูลใดๆให้แก่ RTC ไมโครคอนโทรลเลอร์ที่ต้องการติดต่อด้วยจะต้องส่งข้อมูลที่เป็นคำสั่งควบคุมการติดต่อซึ่งมีขนาด 8 บิตก่อน โดยเริ่มต้นด้วยการให้ขา RST มีสถานะเป็น 1(อยู่ในช่วงการติดต่อ) จากนั้นส่งข้อมูลจำนวน 8 บิตเข้าไปไว้ใน shift register ของ RTC ข้อมูลขนาด 8 บิตจะประกอบด้วยคำสั่งในการควบคุมชิป RTC และตำแหน่งของหน่วยความจำที่ต้องการติดต่อ(address/command byte) ในแต่ละครั้ง การรับข้อมูลจากไมโครโปรเซสเซอร์แต่ละบิตจะกระทำที่ช่วงขอบขาขึ้นของสัญญาณที่ขา SCLK (serial clock)

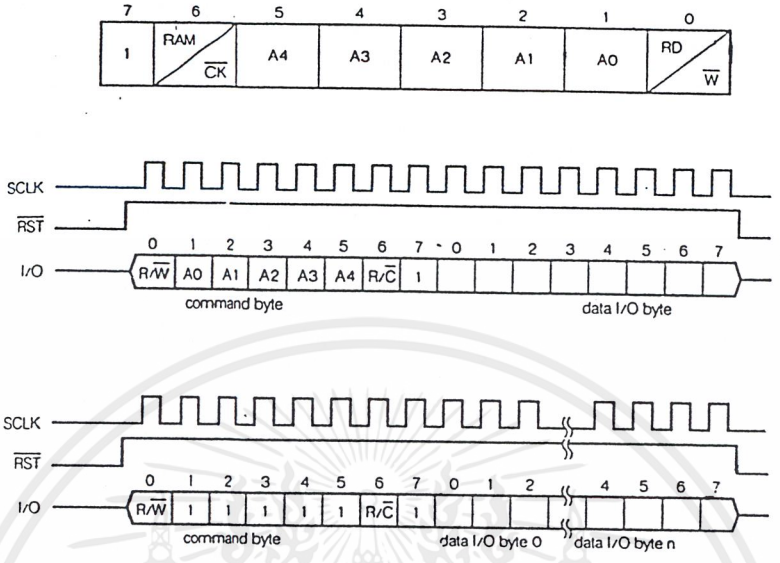
ภายในชิป RTC ประกอบด้วยหน่วยความจำขนาด 24 ไบต์ และรีจิสเตอร์ที่ทำหน้าที่เก็บเวลาของชิปในขณะปัจจุบันจำนวน 8 ตัวและรีจิสเตอร์ทั้ง 8 ตัวนี้สามารถเข้าถึงได้เสมือนเป็นหน่วยความจำตำแหน่งหนึ่ง ดังนั้นต่อไปเราจะมองว่าชิป RTC เบอร์นี้มีหน่วยความจำรวมทั้งสิ้น 32 ตำแหน่ง โดยประกอบขึ้นจากรีจิสเตอร์ 8 ตำแหน่งและหน่วยความจำ 24 ตำแหน่ง

ข้อมูลขนาด 8 บิตแรก (address/command byte) จะระบุตำแหน่งหน่วยความจำที่ต้องการติดต่อ (ทั้งตำแหน่งของหน่วยความจำทั่วไปและตำแหน่งของรีจิสเตอร์สำหรับเก็บเวลา) และบอกว่าเป็นการเขียนหรืออ่านข้อมูลจากหน่วยความจำตำแหน่งนั้นๆรวมทั้งระบุว่าให้การรับส่งข้อมูลเป็นแบบครั้งละ 1 ไบต์หรือครั้งละหลายๆไบต์

หลังจากมีสัญญาณนาฬิกาเกิดขึ้น 8 ครั้ง (สัญญาณ SCLK) ในระหว่างการเขียนข้อมูล 8 บิตแรกเข้าไปใน shift register สัญญาณนาฬิกาต่อไปที่จะเกิดขึ้นเป็นการนำข้อมูลออกจากชิป RTC สำหรับการอ่านข้อมูลหรือนำข้อมูลเข้าไปยังชิป RTC สำหรับการเขียนข้อมูล จำนวนสัญญาณนาฬิกาทั้งหมดที่เกิดขึ้นในการติดต่อครั้งหนึ่งๆจึงเท่ากับ 8+8 ในการส่งข้อมูลครั้งละ 1 ไบต์ single byte mode หรือ 8 บวกมากที่สุด 192 (8x24) สำหรับการส่งข้อมูลครั้งละหลายๆไบต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6.2 โครงสร้างของ **command byte** แสดงในรูปที่3.9



รูปที่ 3.9 แสดงโครงสร้างของ command byte

การรับหรือส่งข้อมูลระหว่างชิป RTC ในตอนเริ่มต้น ไมโครคอนโทรลเลอร์จะต้องส่งข้อมูลเพื่อกำหนดการทำงานให้แก่ RTC เสียก่อน ข้อมูลที่ RTC ได้รับตอนเริ่มต้นนี้จะมีขนาด 1 ไบต์ ซึ่งมีชื่อว่า command byte และเนื่องจากข้อมูลในไบต์นี้จะเป็นตัวกำหนดการทำงานของ RTC ดังนั้นแต่ละบิตในไบต์นี้จะมีความหมายแตกต่างกันไปดังนี้

- MSB (บิต7) ต้องเป็น 1 เสมอ ถ้าเป็น 0 การทำงานต่อจากนี้จะถูกหยุดไว้หมด
- บิต 6 ถ้าเป็น 0 จะระบุว่าต้องการติดต่อกับรีจิสเตอร์สำหรับเก็บเวลา (clock/calendar register) ดังนั้นข้อมูลที่รับส่งกันจะเป็นเวลา หากบิตนี้มีค่าเป็น 1 จะระบุว่าต้องการติดต่อกับหน่วยความจำ
- บิต 1 ถึง 5 เป็นตัวระบุตำแหน่งหน่วยความจำ (ทั้งหน่วยความจำที่ใช้เป็นรีจิสเตอร์สำหรับเก็บเวลาและหน่วยความจำทั่วไป)ที่ต้องการเข้าถึง ไม่ว่าจะเป็นการอ่านหรือเขียนข้อมูล ซึ่งควบคุมด้วยบิต 0 ดังจะได้กล่าวต่อไป
- บิต 0 จะระบุว่าเป็นการเขียนหรืออ่านข้อมูล ถ้าเป็น 0 หมายถึงการเขียนข้อมูลลงไปในชิป หากเป็น 1 หมายถึงการอ่านข้อมูลจากชิป

ในการส่ง command byte ไปยังชิป RTC จะเริ่มต้นด้วยบิต 0 ก่อนเสมอ(LSB first)

3.6.2.1 Burst Mode หมายถึงการรับหรือส่งข้อมูลครั้งละหลายไบต์ในการติดต่อแต่ละครั้ง โดยสามารถกำหนดได้ว่าข้อมูลที่ต้องการติดต่อเป็นรีจิสเตอร์สำหรับเก็บเวลาหรือหน่วยความจำที่ใช้เก็บข้อมูล หากเป็นการรับส่งข้อมูลกับรีจิสเตอร์สำหรับเก็บเวลาก็จะรับส่งกันครั้งละ 8 ไบต์ หาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นการรับส่งข้อมูลกับหน่วยความจำจะรับส่งกันครั้งละ 24 ไบต์ การกำหนดให้รับส่งข้อมูลครั้งละหลายไบต์ กำหนดโดย command byte ดังได้กล่าวมาแล้ว

ในการรับหรือส่งใน burst mode จะเริ่มต้นที่บิต 0 ของหน่วยความจำตำแหน่ง 0 ก่อนเสมอ ไม่ว่าจะเป็นรีจิสเตอร์สำหรับเก็บเวลาหรือหน่วยความจำสำหรับเก็บข้อมูลทั่วไป burst mode นี้มีไว้เพื่อความสะดวกในการรับหรือส่งข้อมูลครั้งละจำนวนมากๆทำให้ไม่ต้องส่ง command byte หลายครั้ง

3.6.2.2 Write Protect Command เวลาที่เดินอยู่ภายในรีจิสเตอร์ที่ทำหน้าที่เก็บเวลาและข้อมูลที่อยู่ในหน่วยความจำทั้ง 24 ตำแหน่ง สามารถป้องกันมิให้เขียนข้อมูลใดๆซ้อนลงไปได้ เพื่อป้องกันเหตุการณ์ที่เกิดขึ้นโดยไม่ได้ตั้งใจ ทั้งนี้โดยการควบคุมจาก write protect register ซึ่งเป็นรีจิสเตอร์ตำแหน่งที่ 7 ของรีจิสเตอร์ที่ทำหน้าที่เก็บเวลาโดยมี write protect bit ซึ่งเป็นบิตที่ 7 ของรีจิสเตอร์ตัวนี้เป็นตัวกำหนดการทำงาน หากwrite protect bit เป็น 0 หมายถึง สามารถเขียนข้อมูลใดๆลงไปยังรีจิสเตอร์ที่เก็บเวลาหรือหน่วยความจำได้ หากบิตนี้เป็น 1 หมายถึงชิป RTC อยู่ในสถานะป้องกันการเขียนข้อมูล ดังนั้นก่อนการเขียนข้อมูลใดๆไปยังรีจิสเตอร์สำหรับเก็บเวลาหรือหน่วยความจำ write protect bit ต้องเป็น 0 เสมอ โดยการให้ RST เป็น 1 (อยู่ในช่วงการติดต่อ) และโหลด write protect command byte (8EH) ตามด้วยข้อมูลที่มีค่า 00H (เคลียร์ให้ write protect bit เป็น 0) หลังจากนั้น RST ต้องกลับมามีสถานะเป็น 0 ก่อนที่คำสั่งอื่นๆจะเริ่มต้นทำงานได้

ส่วนในการบังคับให้ชิป RTC อยู่ในสถานะป้องกันการเขียนข้อมูล ก็ต้องให้ RST เป็น 1 แล้วโหลด command byte ที่มีค่า 8EH ตามด้วยข้อมูล 80H (ให้ write protect bit เป็น 1) ส่วนในการทำงานแบบ burst mode เราไม่สามารถเขียนค่าใดๆเข้าไปใน write protect bit ได้

3.6.2.3 Reset and Clock control การรับหรือส่งข้อมูลทั้งหมดจะต้องเริ่มโดยให้ขา RST มีสถานะเป็น 1 ก่อนเสมอ โดย RST มีหน้าที่หลักอยู่ 2 ประการดังนี้

- RST ใช้ควบคุมการเขียนหรืออ่านข้อมูลใน shift register
- RST ใช้เป็นสัญญาณในการหยุดการทำงานใดๆกับชิป RTCโดยปกติการเขียนข้อมูลเข้าไปใน RTC จะเกิดขึ้นในช่วงขอบขาขึ้นของสัญญาณที่ขา SCLK ส่วนการอ่านข้อมูลจาก RTC จะเกิดขึ้นในช่วงขอบขาลงของสัญญาณที่ขา SCLK โดยในระหว่างการติดต่อ ขา RST ต้องมีสถานะเป็น 1 ตลอดเวลา หากขา RST มีสถานะเป็น 0 หมายถึงยกเลิกการติดต่อหรือสิ้นสุดการติดต่อ

3.6.2.4 Data Input ในตอนเริ่มต้นติดต่อระหว่างชิป RTC กับไมโครคอนโทรลเลอร์ไบต์แรกจะต้องเป็น command byte เสมอ หากในcommand byte ระบุว่าเป็นการเขียนข้อมูลไปในชิป ข้อมูลจะถูกรับเข้ามาในช่วงขอบขาขึ้น (rising edge) ของ SCLK เท่านั้น โดยเริ่มต้นด้วยบิต 0 ก่อนเสมอ และหากเป็นคำสั่งให้รับส่งครั้งละ 1 ไบต์ เมื่อข้อมูลได้รับเข้ามาครบแล้ว สัญญาณ SCLK ที่รับได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เกินจะถูกละเลยไป หากเป็นคำสั่งให้รับส่งแบบ burst mode ซึ่งรับส่งครั้งละ 24 ไบต์ก็มีลักษณะเช่นเดียวกันคือ เมื่อรับข้อมูลครบ 24 ไบต์แล้ว สัญญาณ SCLK ที่รับได้เกินจะถูกละเลยเช่นกัน

3.6.2.5 Data Output หลังจากรับ command byte แล้ว หากมีการระบุว่าเป็นการอ่านข้อมูลจากชิป RTC ข้อมูลจะถูกส่งออกจากชิปสู่ภายนอกในขณะช่วงขอบขาลง(falling edge) ของ SCLK หลังจากมีการรับ command byte เรียบร้อยแล้ว นั่นคือบิตแรกที่จะถูกส่งออกจาก RTC จะเกิดขึ้นในช่วงขอบขาลงของสัญญาณนาฬิกาลูกที่ 9 นั่นเอง

3.6.2.6 Clock/Clender คือ รีจิสเตอร์ 8 ตัวโดยข้อมูลในรีจิสเตอร์เหล่านี้จะอยู่ในรูปของรหัส BCD เท่านั้น

3.6.2.7 Clock Half Flag บิต7 ของรีจิสเตอร์ที่เก็บค่าวินาที จะเป็นตัวบอกให้ชิป RTC หยุดการทำงานของวงจรในส่วนออสซิลเลเตอร์เมื่อบิตนี้มีค่าเป็น 1 ซึ่งเป็นผลให้นาฬิกาภายในชิปหยุดการทำงานไปด้วย และจะบังคับให้ชิปอยู่ในสถานะ low power standby mode และเมื่อบิตนี้เป็น 0 อีกครั้งวงจรออสซิลเลอเตอร์จะเริ่มทำงานต่อทันที

3.6.2.8 AM-PM/12-24mode บิต 7 ของรีจิสเตอร์ที่เก็บค่าชั่วโมง ถูกกำหนดให้เป็น12/24 hour mode select bit นั่นคือ เป็นตัวเลือกว่าจะให้รีจิสเตอร์นี้เก็บค่าชั่วโมงแบบ 12 ชั่วโมงหรือ 24 ชั่วโมง โดย

- บิตที่ 7 เป็น 1 จะเป็นการเลือกให้เก็บค่าแบบ 12 ชั่วโมง โดยมีบิต 5 เป็นตัวบอกว่าเป็นช่วงกลางวันหรือกลางคืน (AM/PM indicator) โดย 1 จะหมายถึง PM และ 0 จะหมายถึง AM
- บิตที่ 7 เป็น 0 จะเป็นการเลือกให้เก็บค่าแบบ 24 ชั่วโมง และบิต 5 จะเป็นบิตที่แสดงหลักสิบตัวที่ 2 ของชั่วโมง(20-23)

3.6.2.9 Write Protect บิต 7 ของ write protect register จะเป็น write protect bit โดย 7 บิตแรกถูกบังคับให้เป็น 0 หมด ทำให้อ่านค่าได้เป็น 0 เสมอ

3.2.6.10 Clock/Calender Burst Mode command byte ที่มีค่า 8EH จะเป็นการระบุว่าให้มีการเขียนข้อมูลในรีจิสเตอร์ที่เก็บเวลาในแบบ burst mode หาก command byte มีค่า BFH จะระบุว่าให้มีการอ่านข้อมูลในรีจิสเตอร์นี้ในแบบ burst mode เช่นกัน ซึ่งใน clock/calendar burst mode นี้จะมีการรับหรือส่งข้อมูลคราวละ 8 ไบต์ติดต่อกัน(รับหรือส่งข้อมูลในรีจิสเตอร์ที่เก็บค่าเวลาทั้ง 8 ตัว) โดยจะเป็นการรับหรือส่งขึ้นกับ command byte ดังได้กล่าวมาแล้ว โดยข้อมูลที่รับหรือส่งจะเริ่มต้นด้วยบิต 0 ของรีจิสเตอร์ 0 ก่อนเสมอ

3.6.2.11 Ram burst Mode command byte ที่มีค่า FEH จะเป็นการระบุว่าให้มีการเขียนข้อมูลในหน่วยความจำแบบ burst mode หาก command byte มีค่า FFH จะระบุว่าให้มีการอ่านข้อมูลในหน่วยความจำแบบ burst mode ทำนองเดียวกับใน clock/calendar burst mode จะมีข้อแตกต่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กันก็เพียงจำนวนข้อมูลที่รับหรือส่งเท่านั้น เพราะหน่วยความจำในชิป RTC มีขนาด 24 ไบต์ ดังนั้น ใน RAM burst mode นี้ข้อมูล 24 ไบต์จะรับส่งกัน โดยเริ่มต้นด้วยบิต 0 ของหน่วยความจำตำแหน่ง 0 ก่อนเสมอ

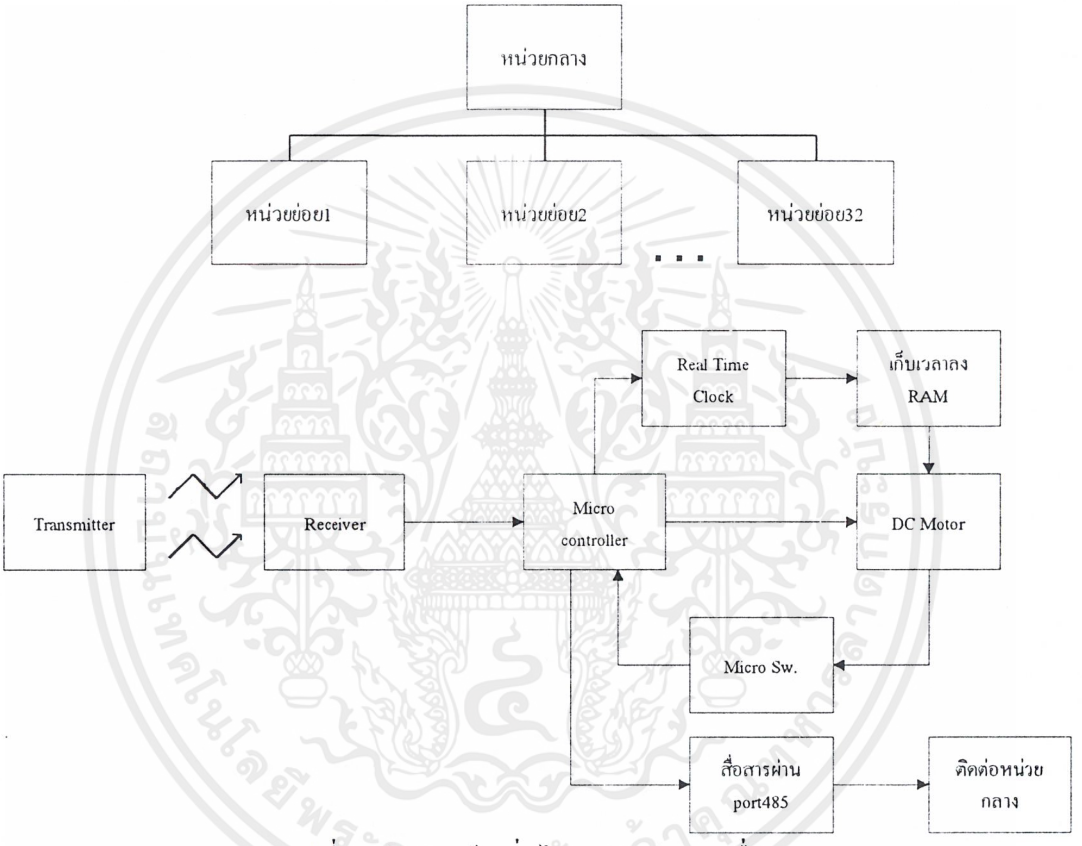
ตารางที่ 3.3 แสดงข้อมูลการทำงานของรีจิสเตอร์และ RAM ใน DS1202

รีจิสเตอร์	ฟังก์ชัน	COMMAND ADDDRSS (HEX)	เขียน= W อ่าน = R	RANGE DATA (BCD)	รีจิสเตอร์ที่กำหนด							
					7	6	5	4	3	2	1	0
0	วินาที	80 81	W R	00-59	CH	10 วินาที			วินาที			
1	นาที	82 83	W R	00-59	0	10 วินาที			นาที			
2	12ชม. 24ชม.	84 85	W W	01-12	12\ 24	0 0	AP ชม. 10ชม.		ชั่วโมง			
3	วัน	86 87	W R	00-23 01-31	0	0	10วัน		วัน			
4	เดือน	88 89	W R	01-12	0	0	0	10 เดือน	เดือน			
5	วัน	8A 8B	W R	01-07	0	0	0	0	วัน			
6	ปี	8C 8D	W R	00-99	10 ปี				ปี			
7	WRITE PROTECT	8E 8F	W R	00-80	WP	เป็น 0 ทั้งหมด						

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การออกแบบวงจรที่ใช้งาน

โครงสร้างของระบบควบคุมการเปิด-ปิดประตูอัตโนมัติ ประกอบด้วยการทำงาน 2 ส่วน คือ หน่วยกลางและหน่วยย่อย ซึ่งทำงานเป็นอิสระต่อกัน และสามารถติดต่อสื่อสารข้อมูล โดยผ่านระบบโครงข่ายที่เชื่อมต่อกันตามมาตรฐาน RS-485 ดังรูปที่4.1ซึ่งมีรายละเอียดดังนี้



รูปที่ 4.1 โครงสร้างทั่วไปของระบบการสื่อสาร

4.1 หน่วยกลาง เมื่อเริ่มต้นการทำงานของหน่วยกลาง จะมีขั้นตอนดังนี้

-ใส่รหัสผ่าน (Password) 4 ตัว จึงจะสามารถเข้ายังระบบของหน่วยกลาง เพื่อติดต่อกับหน่วยย่อยได้

-เมื่อใส่รหัสผ่านได้ถูกต้องแล้วจะมีหมายเลขให้เลือกว่าจะตรวจสอบเวลาเปิดหรือปิดของประตู โดยป้อนหมายเลขที่ให้เลือกผ่านทางแป้นพิมพ์(แป้นพิมพ์จะต่อกับ 8255 และใช้พอร์ทC ในการสแกน)

- จากนั้นก็จะให้เลือกหมายเลขของหน่วยย่อยที่ต้องการติดต่อด้วยโดยเลือกผ่านทางแป้นพิมพ์ ซึ่งหมายเลขที่ป้อนดังกล่าวจะถูกใส่เข้าไปในโปรโตคอลเพื่อสร้างชุดของข้อมูลในการรับส่งข้อมูลระหว่างหน่วยกลางและหน่วยย่อย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยถ้าหมายเลขของหน่วยย่อยที่ป้อนเข้าไปดังกล่าวมีค่าเกินที่ตั้งไว้ในโปรแกรมคือ 32 หน่วยย่อย ก็จะมีข้อความแสดงที่หน้าจอ LCD ว่าใส่ค่า หมายเลข address ของหน่วยย่อยไม่ถูกต้อง และจะกลับไปยังหน้าจอที่ให้เลือกว่าจะตรวจสอบเวลาเปิดหรือปิดประตูอีกครั้ง หรือถ้าในขณะนั้นหน่วยย่อยที่ต้องการติดต่อกำลังทำงานอยู่คือมีการตรวจเจอคนผ่านเข้ามาในบริเวณที่มีการรับส่งอินฟราเรด เก็บเวลาที่ประตูมีการเปิด-ปิดลงหน่วยความจำ และไปควบคุมหมุนมอเตอร์อยู่ หน่วยกลางก็จะไม่สามารถติดต่อได้ ที่หน้าจอ LCD ของหน่วยกลางก็จะมีข้อความแสดงว่า หน่วยย่อยกำลังทำงานอยู่ ให้ติดต่อมาใหม่อีกครั้ง และกลับไปยังหน้าจอที่ให้เลือกว่าจะตรวจสอบเวลาเปิดหรือปิดประตูอีกครั้ง (ในการออกแบบการทำงานนี้ เราได้ให้ความสำคัญกับการทำงานของหน่วยย่อย คือตรวจสอบการรับส่งอินฟราเรด เก็บเวลาที่มีการเปิด-ปิดประตูลงหน่วยความจำ และไปควบคุมการหมุนมอเตอร์ ดังนั้นถ้าหน่วยกลางส่งโปรโตคอลชุดของข้อมูลมาให้หน่วยย่อยในขณะนั้นจะไม่สามารถติดต่อกับหน่วยย่อยที่ต้องการดังกล่าวได้ จะติดต่อได้ก็ต่อเมื่อหน่วยย่อยอยู่ในสภาวะที่ไม่ได้ตรวจเจอคนผ่านเข้ามาในบริเวณรับส่งอินฟราเรด หรือหน่วยย่อยว่างอยู่เท่านั้น) แต่ถ้าในขณะนั้นหน่วยย่อยว่างอยู่ถ้าหน่วยย่อยได้รับการอินเตอร์รัปต์ที่พอร์ทสื่อสารอนุกรม ก็จะทำการตรวจสอบค่าแอดเดรสไบต์ซึ่งเป็นค่าที่ระบุหมายเลขหรือตำแหน่งของไมโครคอนโทรลเลอร์ของหน่วยย่อยที่หน่วยกลางต้องการติดต่อด้วย จากนั้นหน่วยย่อยที่ได้รับโปรโตคอลชุดของข้อมูลดังกล่าวก็จะเกิดการอินเตอร์รัปต์ที่พอร์ทสื่อสารอนุกรมขึ้น แล้วทำการตรวจสอบค่าแอดเดรสไบต์ที่หน่วยกลางส่งมาว่าตรงกับหมายเลขของตำแหน่งตัวเองหรือไม่หากหน่วยย่อยใดมีหมายเลขตำแหน่งของตัวเองตรงกับแอดเดรสไบต์ที่ได้รับเข้ามา ก็จะเคลียร์บิต SM2 และเตรียมรับชุดของข้อมูลที่ตามมาหลังจากได้รับแอดเดรสไบต์เรียบร้อยแล้ว ก็จะเกิดการรับส่งข้อมูลสื่อสารกันระหว่างหน่วยย่อยและหน่วยกลางแล้วหน่วยย่อยก็จะทำการส่งเวลาที่มีการเปิดหรือปิดประตูตามที่หน่วยกลางขอมาไปให้ จากนั้นหน่วยย่อยก็จะกลับไปทำงานตามปกติก่อนที่จะเกิดอินเตอร์รัปต์(คือแสดงเวลาออกหน้าจอ LCD ทำการตรวจสอบการรับส่งอินฟราเรดแล้วไปควบคุมการหมุนมอเตอร์ และบันทึกเวลาที่มีการเปิด-ปิดประตูลงหน่วยความจำต่อไป)ส่วนหน่วยกลางเมื่อได้รับเวลาเปิดหรือปิดประตูที่ต้องการแล้ว ก็จะแสดงเวลาที่ได้รับมาดังกล่าวแสดงออกหน้าจอ LCD จนเสร็จ

4.2 หน่วยย่อย ใช้ควบคุมการเปิด-ปิดประตู 1 ประตูซึ่งมีส่วนสำคัญๆดังนี้

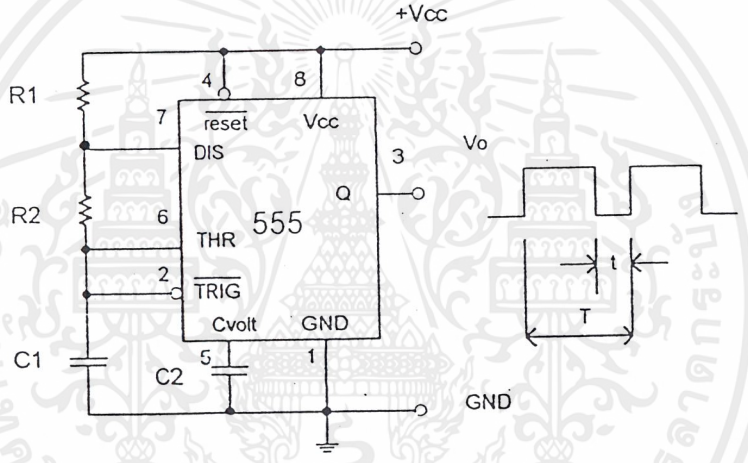
- ภาคส่งสัญญาณอินฟราเรด
- ภาครับสัญญาณอินฟราเรด
- ไมโครคอนโทรลเลอร์ที่ใช้ควบคุมการขับมอเตอร์
- วงจรควบคุมการขับมอเตอร์
- ไมโครสวิตซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ระบบสร้างฐานเวลาให้กับระบบโดยใช้ RTC(Real Time Clock)
- การออกแบบโปรโตคอล
- มาตรฐานการสื่อสารข้อมูล EIA RS485
- การรับส่งข้อมูลอนุกรมของไมโครคอนโทรลเลอร์

4.2.1 การออกแบบภาคส่งสัญญาณอินฟราเรด

เนื่องจาก LED อินฟราเรด สามารถทนกระแสสูงสุดได้ 50 mA ทนกระแสพัลส์สูงสุดได้ 1 A และสามารถทนแรงดันสูงสุดได้ 150 mW เมื่อเราต้องการส่งกระแสให้มากที่สุด โดยไม่เกินกำลังของ LED จึงใช้ไอซีไทเมอร์ 555 เป็นตัวให้กำเนิดสัญญาณพัลส์ เพื่อควบคุมการจ่ายกระแสให้ LED ใช้วงจรอะสเตเบิลเพื่อให้กำเนิดความถี่ตามต้องการ โดยต่อวงจรดังรูป 4.2



รูปที่ 4.2 แสดงการใช้ไอซี 555 เป็นวงจรอะสเตเบิล

โดยสามารถประมาณค่า R_1 และ R_2 ได้จาก

$$t = (-\log e^{2/3})(R_1+R_2)C_1 \quad (4.1)$$

$$= 0.694\,(R_1+R_2)C_1 \quad (4.2)$$

และ

$$T-t = 0.694R_2C \quad (4.3)$$

และสามารถหาช่วงเวลาทั้งหมดได้จาก

$$T = 0.694\,(R_1+R_2)C_1 \quad (4.4)$$

เนื่องจากค่าความถี่เอาต์พุต $f_0 = 1/T$

$$f_0 = 1.44\,(R_1+R_2)C_1 \quad (4.5)$$

และจะได้ค่า Duty Factor : t / T คือ

$$t/T = (R_1+R_2)/(R_1+2R_2) \quad (4.6)$$

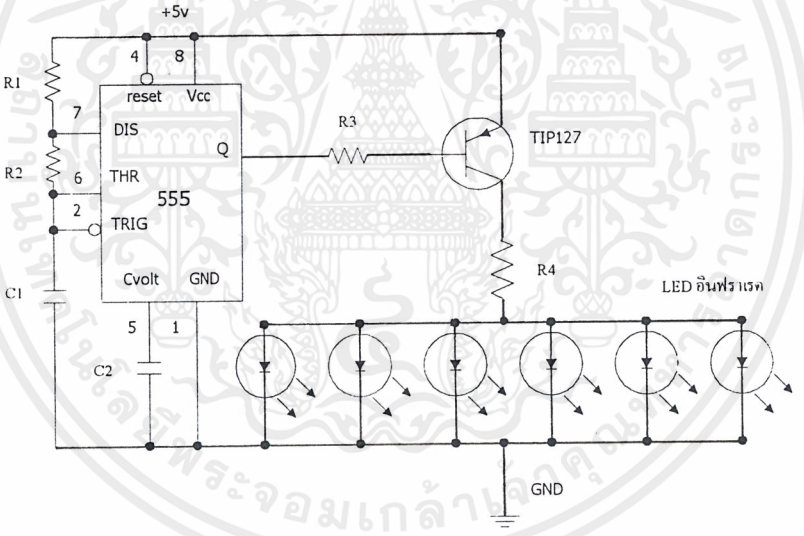
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากเราสามารถกำหนดค่า Duty factor ด้วยวิธีนี้ได้จำกัด ตัวไอซีเองไม่สามารถจ่าย Duty factor ที่มีค่ามากกว่า 0.5 ได้เนื่องจากค่าความต้านทานในการคายประจุ $C(R_1+R_2)$ มากกว่าค่าความต้านทานในการคายประจุ (R_2) อย่างแน่นอน จึงเป็นการง่ายกว่าที่จะปรับค่า (T-t) ให้ได้ค่า Duty factor ที่ต้องการแล้วค่อยนำสัญญาณไปกลับเฟสอีกทีหนึ่ง

ในที่นี้ใช้ค่า Duty factor = 6% และค่า C1 = 10nF จึงคิด Duty Factor = (100-6) = 94% แทนจะได้จากสมการ 4.6

$$t / T = 94\ \%$$

จะได้สัญญาณออกที่ขา 3จะเป็นสัญญาณที่มีขนาด t = 94% ซึ่งนำไปขับ LED อินฟราเรด เนื่องจากไอซีไทม์เมอร์ 555 ไม่สามารถจ่ายกระแสได้เพียงพอจึงใช้ทรานซิสเตอร์แบบ PNP ช่วยขับกระแสและกลับเฟสสัญญาณ ในที่นี้ใช้ทรานซิสเตอร์เบอร์ TIP 127 พิจารณารูปวงจรภาคส่งที่สมบูรณ์ได้ดังรูปที่ 4.3 ซึ่งเมื่อพิจารณาการทำงานได้คือ



รูปที่ 4.3 แสดงวงจรภาคส่งที่สมบูรณ์

เมื่อแรงดันที่ออกจากขา 3 เป็นบวกจะมีค่ามากกว่า V_{BE} เสมอ ดังนั้นเมื่อแรงดันที่ออกจากขา 3 ของไอซีมีค่าเป็น high ทรานซิสเตอร์ TIP 127 จะไม่ทำงานและเมื่อแรงดันที่ออกจากขา 3 ของไอซีมีค่าเป็น low ทรานซิสเตอร์ TIP127 จะทำงานและขับกระแสให้ LED อินฟราเรด โดยสามารถควบคุมค่ากระแสผ่าน LED อินฟราเรดได้จากการควบคุมค่า R_4 นั่นเอง

พิจารณาการคำนวณหาค่า R_3 และ R_4 สามารถคำนวณได้ดังนี้คือ

$$V_{cc} - V_{ce\ sat} - V_{Diode} = I_C R_4 \qquad (4.7)$$

ในที่นี้ได้ใช้ค่า R_4 = 1.25 โอห์ม V_{cc} = 5V , $V_{ce\ sat}$ = 2V , V_{Diode} = 1.7V แทนค่าใน (4.7) จะได้

$$I_c = 1.04\ A$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และเมื่อเราพิจารณาที่ I_B ก็ได้ว่าเนื่องจากทรานซิสเตอร์ TIP 127 จะ ON เมื่อเป็นพัลส์ขาลงดังนั้น

$$V_{cc} - V_{BE} - 0 = I_B R_3 \quad (4.8)$$

ในที่นี้ใช้ค่า $V_{cc} = 5V$, $V_{BE} = 0.7V$, $R_3 = 347$ โอห์ม แทนใน (4.8) จะได้

$$I_B = 12 \text{ mA}$$

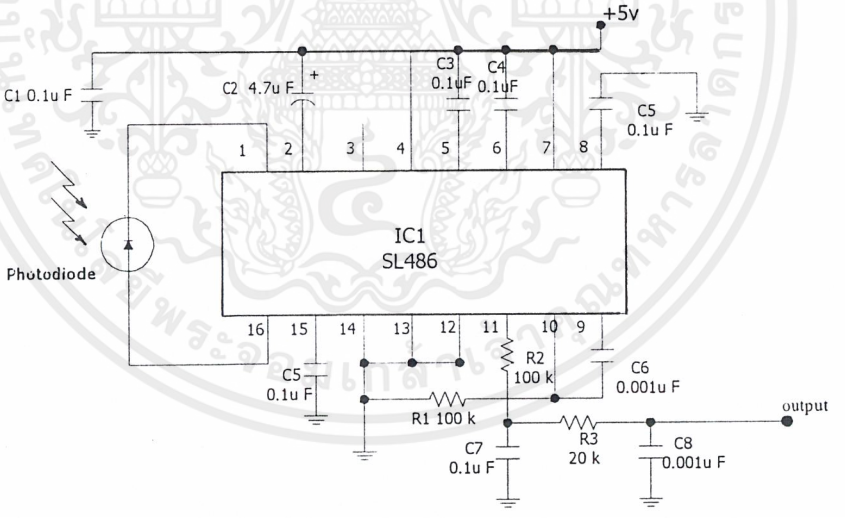
เมื่อได้ค่าดังกล่าวก็จะสามารถสร้างวงจรขับสัญญาณอินฟราเรดได้โดยที่ค่า R1 และ R2 ควรจะใส่เป็นค่า R ปรับค่าได้เพื่อป้องกันความผิดพลาดที่อาจเกิดขึ้นได้จากการทดลองบนโปรโต้บอร์ดเมื่อนำมาลงแผ่นปริ้นท์ จึงควรใส่เป็น R ปรับค่าไว้เพื่อจะได้แก้ไขข้อผิดพลาดของ Duty Factor ได้

4.2.2 การออกแบบภาครับสัญญาณอินฟราเรด

ในส่วนของภาครับสัญญาณอินฟราเรดนั้นจะประกอบไปด้วย 2 ส่วนด้วยกันคือ ส่วนของตัวรับสัญญาณอินฟราเรดโดยใช้ตัวไอซี SL 486 และอีกส่วนหนึ่งเป็นส่วนของวงจรเปรียบเทียบซึ่งสัญญาณที่ได้จากวงจรเปรียบเทียบจะเป็นสัญญาณที่ระดับ 0 และ 5 โวลต์ ซึ่งจะส่งต่อไปยังภาคต่อไปของวงจร พิจารณารายละเอียดได้ดังนี้คือ

4.2.2.1 วงจร SL 486

เป็นส่วนแรกของวงจรภาครับสามารถพิจารณาการต่อรูปวงจรได้ดังรูปที่ 4.4



รูปที่ 4.4 แสดงวงจรของ SL 486

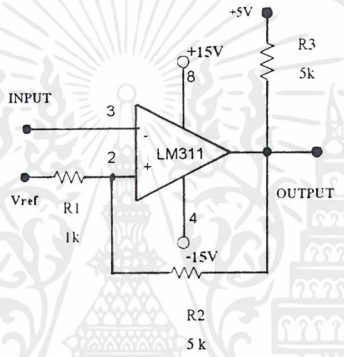
พิจารณา SL 486 เป็นไอซีที่ทำหน้าที่รับสัญญาณความถี่อินฟราเรดจากโฟโต้ไดโอด โดยเฉพาะเมื่อมีแสงจาก LED ตัวส่งมากระทบกับ LED ตัวรับซึ่งต่ออยู่ในลักษณะไบอัสกลับ ทำให้มีสัญญาณมาเข้าที่ IC_1 หลังจากนั้น IC_1 ก็จะทำการตรวจจับสัญญาณและมีการดีมอดูเลต (Demodulate) ออกจากความถี่แสงให้สัญญาณเอาต์พุตออกมาที่ขา 11 ผ่านตัวต้านทาน R_2 , C_7 , R_3 และ C_8 ซึ่งอุปกรณ์เหล่านี้ทำหน้าที่เป็นวงจรกรองความถี่ซึ่งจะส่งสัญญาณเอาต์พุตดังกล่าวไปยังส่วนต่อไป เราจะไม่เอาสัญญาณเอาต์พุตที่ขา 9 โดยตรงเพราะว่ามีขนาดของสัญญาณที่เบามากทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ให้ไม่เพียงพอกับวงจรในภาคต่อไป และที่ขา 9 มีสัญญาณรบกวนมากจึงต้องทำการลดสัญญาณรบกวนนั้นด้วยวงจรภายในตัว IC_1 เองพร้อมกับการควบคุมระดับแรงดันอัตโนมัติและวงจรกรองความถี่ที่ไม่ต้องการออกไปจากสัญญาณเอาต์พุตที่ขา 11 ก็ได้สัญญาณที่มาสามารถส่งต่อไปยังวงจรส่วนอื่นต่อไปได้

4.2.2.2 วงจรเปรียบเทียบ

เมื่อได้สัญญาณจากตัว SL486 แล้วสัญญาณดังกล่าวก็จะมาเข้าที่ส่วนของวงจรเปรียบเทียบซึ่งจะทำการเปรียบเทียบแรงดันของเอาต์พุตที่ SL486 กับแรงดันที่อ้างอิงไว้โดยการต่อวงจรดังรูปที่ 4.5



รูปที่ 4.5 วงจรเปรียบเทียบ (Comparator)

เราสามารถคำนวณได้ตามสมการ (4.9)

$$V_1 - V_2 = \{2R_1 / (R_1+R_2)\} V_{CC} \qquad (4.9)$$

ในที่นี้กำหนดให้ R1 = 1k ดังนั้น

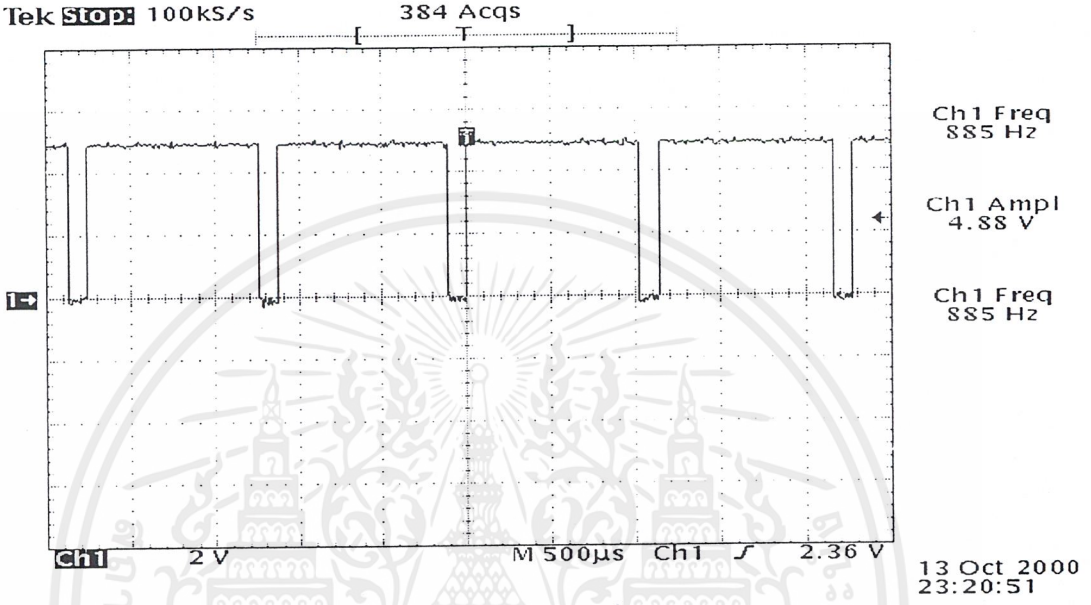
$$5 - 0 = \{2k / (2k + R2)\} 15$$

จะได้ $$R2 = 5k$$

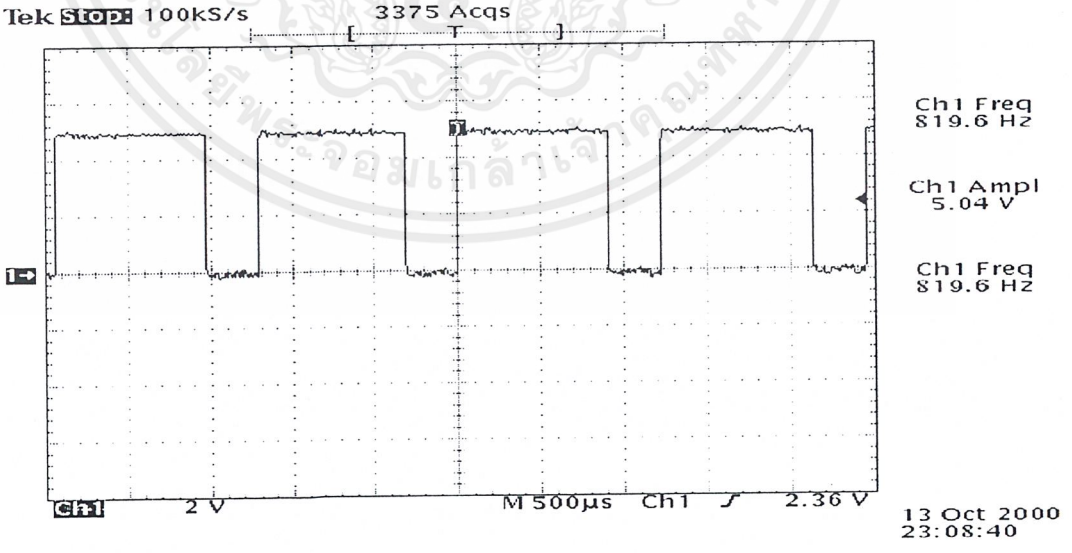
4.2.3 ไมโครคอนโทรลเลอร์ที่ใช้ควบคุมการขับมอเตอร์

ตัวไมโครคอนโทรลเลอร์จะรับสัญญาณมาจากตัววงจรเปรียบเทียบซึ่งสัญญาณที่ได้มาจะเป็นระดับสัญญาณ 0 และ 5 โวลต์ซึ่งก็คือระดับลอจิกของดิจิตอลนั่นเองซึ่งโปรแกรมในการควบคุมการขับมอเตอร์จะทำการขับมอเตอร์ให้เปิด-ปิดประตูตามระดับสัญญาณที่ได้รับมาจากวงจรเปรียบเทียบและจะทำการตรวจสอบอยู่ตลอดเวลา รวมทั้งการควบคุมระดับความเร็วในการเปิด-ปิดประตูด้วยการส่งพัลส์ขนาดต่างๆกันตามรูปที่ 4.6 และ 4.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 แสดงพัลส์ที่ออกมาจากไมโครคอนโทรลเลอร์เพื่อไปป้อนเข้าวงจรขับมอเตอร์ให้ประตูเปิด-ปิด เร็ว



รูปที่ 4.7 แสดงพัลส์ที่ออกมาจากไมโครคอนโทรลเลอร์เพื่อไปป้อนเข้าวงจรขับมอเตอร์ให้ประตูเปิด-ปิด ช้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.4 วงจรควบคุมการทำงานของมอเตอร์

วงจรที่ใช้ในการควบคุมการทำงานของมอเตอร์แสดงในภาคผนวก ซึ่งมีการทำงานดังนี้ คือ ขา 2 ,12 ของแนนด์เกทจะรับอินพุทมาจากไมโครคอนโทรลเลอร์ โดยเมื่อมีคนผ่านเข้ามาในบริเวณที่มีการรับส่งอินฟราเรด ไมโครคอนโทรลเลอร์จะส่งลอจิก 0 (0 V) ไปเข้าขา 12 ของแนนด์เกท และส่งลอจิก 1 (5V) ไปเข้าขา2 ของแนนด์เกท ทำให้ขา 6 ของแนนด์เกทมีศักดาเป็น 5V ก็จะมี I_b ไหลเข้า Q1 ทำให้ Q1 on ก็จะมีการดึงกระแสทำให้ I_bของ Q5 ไหล Q5 ก็จะ on ในขณะเดียวกัน ศักดาที่ขาเบสของ Q4 จะมีค่าเป็น 0 โวลต์ ก็จะไม่มีการดึงกระแส Q4 ก็จะ off กระแสจากไฟเลี้ยงจึงไหลเข้าขาเบสของ Q8 มี I_c ไหลผ่าน Q8 Q8 จึง on โดยขณะนี้ขา 12 ของแนนด์เกทมีศักดา 0 โวลต์ ทำให้ศักดาที่ ขาเบสของ Q2 มีค่าเป็น 0 โวลต์ ไม่มีการดึงกระแส Q4 จึง off และทำให้ Q6 off ในขณะเดียวกัน Q3 จะ on ทำให้ Q7 off นั่นคือในตอนนี้จะมีการดึงกระแสผ่าน Q5 และ Q8 ทำให้ Q5 ,Q8 on มอเตอร์จึงหมุนได้

และเมื่อไม่มีคนผ่านเข้ามาในบริเวณที่มีการรับส่งอินฟราเรด ไมโครคอนโทรลเลอร์ก็จะส่งลอจิก 0 (0V) ไปเข้าขา 2 ของแนนด์เกท และส่งลอจิก 1 (5V) ไปเข้าขา 12 ของแนนด์เกท วงจรก็จะทำงานในลักษณะเดียวกับแบบแรก ทำให้มีการดึงกระแสผ่าน Q6 และ Q7 ทำให้ Q6,Q7 on มอเตอร์จึงหมุนกลับทิศไปในอีกทางหนึ่งได้

มีการใส่ไดโอดคร่อมทรานซิสเตอร์ เพราะเมื่อปิดไฟเลี้ยงที่จ่ายให้มอเตอร์ ศักดาที่ตกคร่อม DC Motor จะกลับทิศ เราจึงต้องใส่ไดโอดเข้าไป เพื่อให้เป็นทางผ่านของกระแสที่ไหลย้อนกลับจาก DC Motor เข้าไฟเลี้ยง เพื่อป้องกันไม่ให้ทรานซิสเตอร์เสียหาย รูปวงจรแสดงในท้ายบทนี้

ผลการวัดกระแสที่ใช้ในการขับมอเตอร์ที่ความเร็ว 2 ระดับแสดงดังตารางที่ 4.1

ตารางที่ 4.1 กระแสที่ใช้ในการขับมอเตอร์ที่ความเร็ว 2 ระดับ

ความเร็วในการเปิด-ปิด	Duty Cycle (%)	กระแสที่ใช้ขับมอเตอร์(A)
เร็ว	90	1.467
ช้า	72	1.2

การคำนวณเมื่อนำมูเล่มาใช้ในการออกแบบ

ให้ R_1 แทนรัศมีของมูเล่ตัวใหญ่

R_2 แทนรัศมีของมูเล่ตัวเล็ก

- ถ้าหมุนด้วย speed เร็วมูเล่ตัวใหญ่หมุนด้วยความเร็ว 30.625 รอบ/นาที

ดังนั้นจากสมการ $\frac{R_1}{R_2} = \frac{N_2}{N_1}$ แทนค่า $R_1 = 37mm$,$R_2 = 15mm$, $N_1 = 30.625$รอบ/นาที

ดังนั้น $N_2 = 75.54$ รอบ/นาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

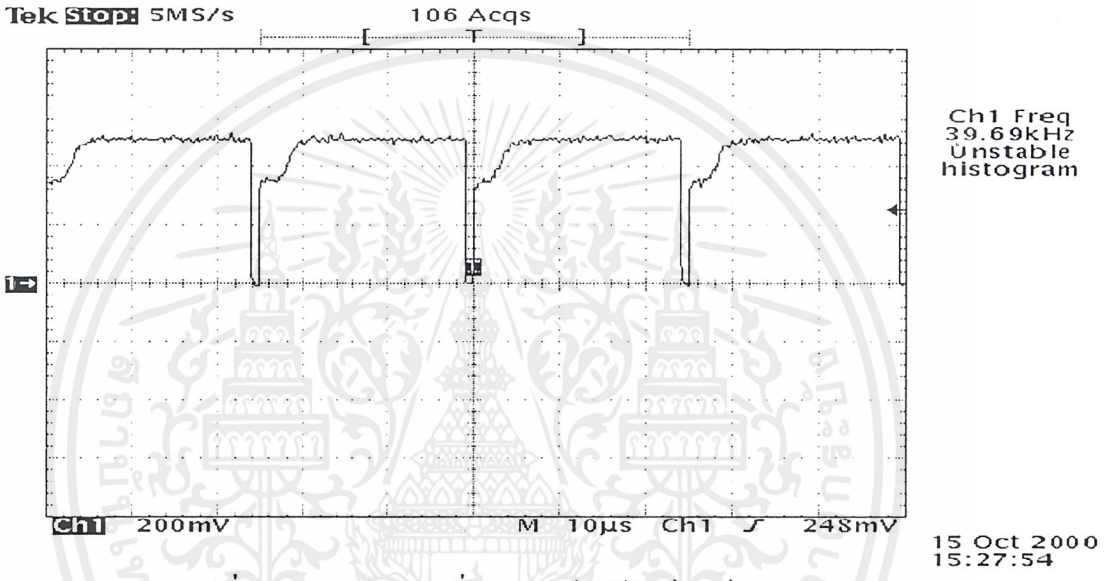
จะได้ระยะทางที่ประตูเคลื่อนที่ได้ใน 1นาที = $2\pi R_2 N_2$ = 2 x 3.1416 x15mm x 75.54 = 7.12 m

- ถ้าหมุนด้วย speed ช้ามูเล่ตัวใหญ่หมุนด้วยความเร็ว 18.75 รอบ/นาที

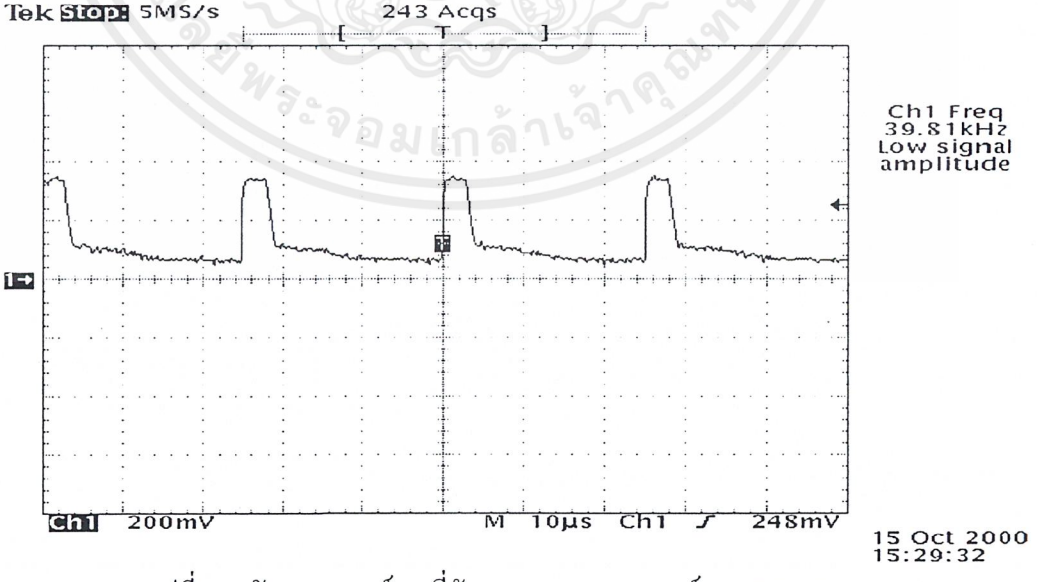
ดังนั้นจากสมการ $\frac{R_1}{R_2} = \frac{N_2}{N_1}$ แทนค่า R_1 = 37mm ,R_2 = 15mm , N_1 = 18.75รอบ/นาที

ดังนั้น N_2 = 46.25 รอบ/นาที

จะได้ระยะทางที่ประตูเคลื่อนที่ได้ใน 1นาที = $2\pi R_2 N_2$ = 2 x 3.1416 x15mm x 46.25 = 4.36 m



รูปที่ 4.8 แสดงสัญญาณที่ออกมาจาไอซีไทม์เมอร์ 555 ขา 3



รูปที่ 4.9 สัญญาณเอาต์พุตที่วัดจากขาคอลเลคเตอร์ของTIP 127

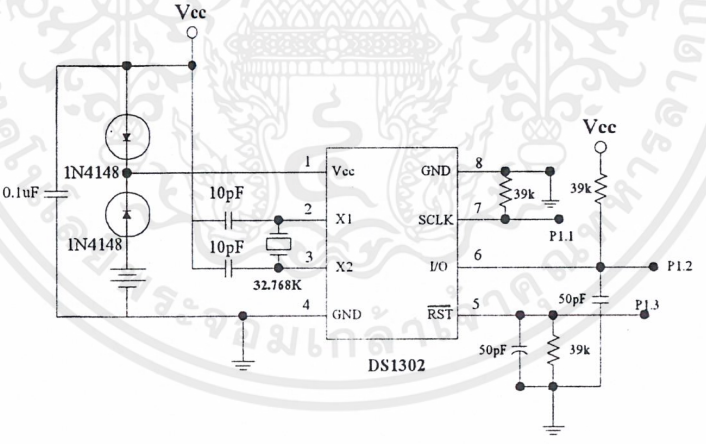
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.5 ไมโครสวิตช์

โดยจะใช้ไมโครสวิตช์ 2 ตัวโดยตัวหนึ่งอยู่ที่ขอบของโครงประตู ช่วยป้องกันความเสียหายที่อาจเกิดขึ้นกับวงจรขับมอเตอร์ โดยเมื่อระบบทำงานผิดพลาด มอเตอร์จะหมุนไม่หยุดไปชนไมโครสวิตช์ที่ขอบประตู แล้วประตูก็จะหยุดหมุนและมีเสียงเตือนดังขึ้นประมาณ 40 วินาที เป็นการบอกว่าระบบเกิดการทำงานผิดพลาดขึ้น

ส่วนอีกตัวหนึ่งติดที่ขอบของบานประตู เพื่อตรวจสอบว่าประตูทำการปิดจนสุดแล้วหรือยังเมื่อไมโครสวิตช์ตัวนี้ทำงานแสดงว่าประตูปิดสนิทแล้ว มอเตอร์ก็จะหยุดหมุนและไปคอยตรวจสอบว่ามีการคนผ่านเข้ามาในบริเวณที่มีการรับส่งอินฟราเรดอีกหรือไม่เพื่อทำการเปิดประตูในครั้งต่อไป

4.2.6 ระบบสร้างฐานเวลาให้กับระบบโดยใช้ RTC(Real Time Clock) ใช้ชิป DS-1302 ทำหน้าที่สร้างระบบเวลาจริงให้กับแต่ละหน่วยย่อย ผู้ควบคุมระบบสามารถปรับแต่งค่าเวลา วัน เดือน ปี ชั่วโมง นาที วินาที โดยผ่านทางแป้นพิมพ์ (แป้นพิมพ์จะต่อกับ 8255 และใช้พอร์ต C ในการสแกน) และเมื่อประตูมีการเปิด-ปิดก็จะทำการบันทึกเวลาดังกล่าวลง RAM รูปวงจรของ RTC แสดงดังรูปที่ 4.10



รูปที่ 4.10 วงจร RTC

4.2.7 การออกแบบโปรโตคอล

โปรโตคอลเพื่อสร้างชุดของข้อมูลในการรับส่งข้อมูลมีรายละเอียดดังนี้

เฟรมข้อมูลประกอบด้วย HEADER PORTION และ DATA PORTION ดังรูปที่ 4.11

SYN.	HEADER	DATA PORTION	SYN.	SYN.	SYN.

รูปที่ 4.11 ส่วนประกอบ 1 เฟรมของโปรโตคอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.7.1 HEADER PORTION

ในส่วนของ HEADER ของโปรโตคอลมีรายละเอียดดังรูปที่ 4.12

SOH	CDAT	ไม่ใช้งาน	ไม่ใช้งาน	ADDRESS	LRC#1
8 BIT	8 BIT	8 BIT	8 BIT	24 BIT	8 BIT

รูปที่ 4.12 ส่วนประกอบของ HEADER PORTION

- SOH : Start of Header [8บิต] เป็นส่วนแสดงการเริ่มต้นข้อความของ HEADER POTRION มีขนาด 8 บิต

- CDAT : Count of Data เป็นส่วนแสดงจำนวนของข้อมูลที่บรรจุอยู่ใน DATA FILD ของ DATA PORTION โดยนับข้อมูลขนาดไบนารีได้สูงสุด 256 ไบต์ต่อ 1 เฟรม

- ADDRESS : Address of node [24 บิต] : เป็นส่วนแสดงค่าตำแหน่งของตัวรับที่ตัวส่งต้องการติดต่อด้วย มีขนาด 24 บิต

เมื่อตัวรับเป็นหน่วยกลาง

ADDRESS = CXX : C = 43H (ASCII)

: XX = 00H (ASCII 2 byte)

เมื่อตัวรับเป็นหน่วยย่อย

หน่วยย่อย = TXX:T = 54H (ASCII)

: XX = 01H-32H(ASCII 2 byte)

- LRC# 1 (Longitudinal Redundacy Check) เป็นส่วนแสดงความผิดพลาดและเป็นรหัสเช็คความผิดพลาดของ HEADER มีขนาด 8 บิต คำนวณจาก SOH , CDAT และADDRESS

4.2.7.2 DATA PORTION

ในส่วนของ DATA PORTION มีรายละเอียดดังรูปที่ 4.13

STX	DATA FIELD	EXT	LRC#2
8 BIT	32-288 BIT	8 BIT	8 BIT

รูปที่ 4.13 ส่วนประกอบของ DATA PORTION

- STX : Strat of text [8บิต] เป็นส่วนแสดงการเริ่มต้นข้อความของส่วน DATA PORTION มีขนาด 8 บิท

- EXT : End of text [8 บิต] เป็นส่วนแสดงการสิ้นสุดข้อความของส่วน DATA PORTION มีขนาด 8 บิท

- LRC#2 : (Longitudinal Redundacy Check) [8บิต] เป็นส่วนแสดงรหัสตรวจสอบของ DATA PORTION มีขนาด 8 บิต คำนวณจาก STX , DATA FIELD และ ETX

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- DATA FIELD : เป็นส่วนแสดงข้อมูลที่ใช้ในการติดต่อ มีขนาดไม่เกิน 256 ตัวอักษรต่อเฟรมข้อความดังรูปที่ 4.14

COMMAND	SEP	SDB	DATA	EDB
8 BIT	8 BIT	8 BIT	0-256 BIT	8 BIT

รูปที่ 4.14 ส่วนประกอบของ DATA PORTION

COMMAND : คำสั่ง

- SEP : Seperator (ASCII = '-' = 2DH) [8บิต] เป็นตัวกั้นระหว่างคำสั่งกับข้อมูล หรือพารามิเตอร์

- SDB : Start of Data Block (ASCII = '(' = 28H) [8 บิต] เป็นส่วนแสดงการเริ่มต้นของข้อมูล หรือพารามิเตอร์

- EDB : End of Data Block (ASCII = ')' = 29H) [8 บิต] เป็นส่วนแสดงการสิ้นสุดของข้อมูล หรือ พารามิเตอร์

สำหรับชุดคำสั่งและรูปแบบการโพลข้อมูลแสดงในภาคผนวก

4.2.8 มาตรฐานการสื่อสารข้อมูล EIA RS485

การสื่อสารข้อมูลอนุกรมระหว่างคอมพิวเตอร์ด้วยกัน หรือ คอมพิวเตอร์กับอุปกรณ์รอบข้าง ด้วยตัวรับและตัวส่งแบบไม่สมดุลย์ อาทิเช่น RS-232-C และ RS-423-A จะประสบปัญหาเกิดข้อมูลผิดพลาดได้ง่ายเมื่อมีการรับส่งข้อมูลด้วยอัตราการรับส่งข้อมูลสูง การใช้คู่สายสัญญาณรับส่งข้อมูลยาวมากๆในการรับส่งข้อมูลที่มีระยะทางที่ไกล รวมทั้งการวางคู่สายสัญญาณรับส่งผ่านบริเวณที่มีสัญญาณรบกวนสูง The Electronics Industries Association (EIA) จึงได้พัฒนามาตรฐานการสื่อสารข้อมูลอนุกรมแบบสมดุลย์ (ตัวรับและตัวส่งไม่ใช้สายสัญญาณกราวน์ร่วมกัน) ขึ้นเพื่อแก้ปัญหาดังกล่าว โดยได้พัฒนาขึ้นสองมาตรฐาน ได้แก่ RS-422-A และ RS-485 ซึ่งมีการประยุกต์ใช้งานแตกต่างกัน ขึ้นอยู่กับความเหมาะสมของระบบงาน สำหรับการเปรียบเทียบมาตรฐานการรับส่งข้อมูลที่มีตัวรับและตัวส่งแบบไม่สมดุลย์ กับ ตัวรับและตัวส่งแบบสมดุลย์ แสดงในภาคผนวก

RS-485 เป็นมาตรฐานการสื่อสารข้อมูลแบบสมดุลย์ที่ได้พัฒนาจากมาตรฐาน RS-422-A เพื่อให้ตัวรับและตัวส่งจำนวนมากคู่สามารถใช้คู่สายสัญญาณรับส่งร่วมกันได้ (multipoint multiple drivers and receivers) ซึ่งในกรณีของ RS-422-A คู่สายสัญญาณรับส่งคู่หนึ่ง จะมีตัวรับได้ไม่เกิน 10 ชุด และมีตัวส่งเพียง 1 ชุด แต่ในกรณีของ RS-485 สามารถใช้ตัวรับ 32 ชุดและตัวส่ง 32 ชุดร่วมกันได้ภายในสายสัญญาณคู่เดียวโดยทั่วๆไป RS-485 มีคุณลักษณะเฉพาะทางไฟฟ้าของตัวรับและตัวส่งคล้ายกับตัวรับและตัวส่งของ RS-422-A และไม่จำกัดรูปแบบของโปรโตคอลที่จะนำมาใช้งานกับระบบที่พัฒนาขึ้น โดยขึ้นอยู่กับผู้พัฒนาระบบเองว่าจะเลือกโปรโตคอลแบบไหนมาใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

งาน นอกจากนี้ตัวรับและตัวส่งมีราคาไม่สูง ทำให้ RS-485 ถูกนำมาประยุกต์ใช้งานในระบบการสื่อสารข้อมูลอนุกรมแบบโครงข่ายอย่างแพร่หลาย

4.2.8.1 คุณลักษณะเฉพาะของตัวส่ง RS-485

- ตัวส่ง 1 ตัว สามารถขับโหลดได้ถึง 32 ชุด (โหลดหนึ่งชุดประกอบด้วยตัวส่ง 1 ตัวและตัวรับ 1ตัว)และค่าความต้านทานรวมที่ต่อคร่อมคู่สายสัญญาณมีค่า 60 โอห์ม หรือมากกว่า

- เอาท์พุทของตัวส่งในสภาวะออฟ มีกระแสรั่วไหลไม่เกิน 100 uA ในช่วงแรงดันไฟฟ้าโหมดร่วมระหว่างค่า –7V ถึง 7V

- เอาท์พุทของตัวส่งให้แรงดันไฟฟ้าเอาท์พุท 1.5V ถึง 5V ในช่วงแรงดันไฟฟ้าโหมดร่วมระหว่างค่า –7V ถึง 12V

-ตัวส่งมีวงจรป้องกันตัวเองที่ส่วนเอาท์พุทในกรณีที่ตัวส่งหลายๆตัวส่งข้อมูลออกมาพร้อมๆกัน

4.2.8.2 คุณลักษณะเฉพาะของตัวรับ RS -485

- ค่าความต้านทานที่อินพุทมีค่าสูง โดยมีค่าไม่น้อยกว่า 12 กิโลโอห์ม

- ตัวรับมีค่าแรงดันไฟฟ้าอินพุทโหมดร่วม ระหว่างาค่า –7V ถึง 12V

-ตัวรับสามารถตอบสนองต่อสัญญาณที่แตกต่างจากสัญญาณโหมดร่วมได้±200mV (น้อยสุด)

4.2.8.3 คุณลักษณะเฉพาะของคู่สายสัญญาณ RS-485

- คู่สายสัญญาณรับส่ง ควรพันสลับกันเป็นเกลียวยาว เพื่อลดทอนสัญญาณรบกวน

4.2.8.4 คุณสมบัติของคู่ตัวรับ-ส่ง (Transceivers) RS-485

คู่ตัวรับ-ส่ง (Transceivers) เป็นอุปกรณ์ที่มีทั้งตัวรับ และตัวส่งอยู่ในชิปเดียวกัน เพื่อให้เกิดความสะดวกใช้การใช้งาน และทำให้ระบบมีขนาดเล็กลง คู่ตัวรับ-ส่ง ของ RS-485 มีอยู่หลายเบอร์ได้แก่ SN75176B , SN75177B , SN75178B , SN75179B คุณลักษณะเฉพาะของคู่ตัวรับ-ส่ง (Transceivers) มีดังนี้

- ตามมาตรฐาน RS-485 , RS-422A , CCITT V.11 และ X.27

- เอาท์พุทของตัวส่งเป็นแบบ 3-State ยกเว้น SN75179B

- เอาท์พุทตัวส่งสามารถขับกระแสได้สูง 60 mA

- Thermal Shutdown Protection

- ค่าความต้านทานอินพุทของตัวรับ 20k (น้อยที่สุด)

- ตัวรับมีค่า input sensitivity ± 200 mV

- คู่ตัวรับมีค่า input hysteresis 50 mV, ใช้ไฟเลี้ยง 5 โวลท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รับ หรือถ้าตัวรับตั้งค่า SM2 = 0 การอินเทอร์รัพท์การรับข้อมูลอนุกรมโดยข้อมูลอนุกรมจากตัวส่งจะเกิดขึ้น โดยตัวส่งต้องตั้งบิต TB8 = 0 ซึ่งค่านี้เมื่อตัวรับ รับมาแล้วจะถูกเอาไว้ในบิต RB8 ของตัวรับ

นำหลักการข้างต้นมาประยุกต์ใช้ในการติดต่อทางอนุกรมแบบมัลติโปรเซสเซอร์โหมด 3 ระหว่างหน่วยกลางและหน่วยย่อย ดังตัวอย่างต่อไปนี้

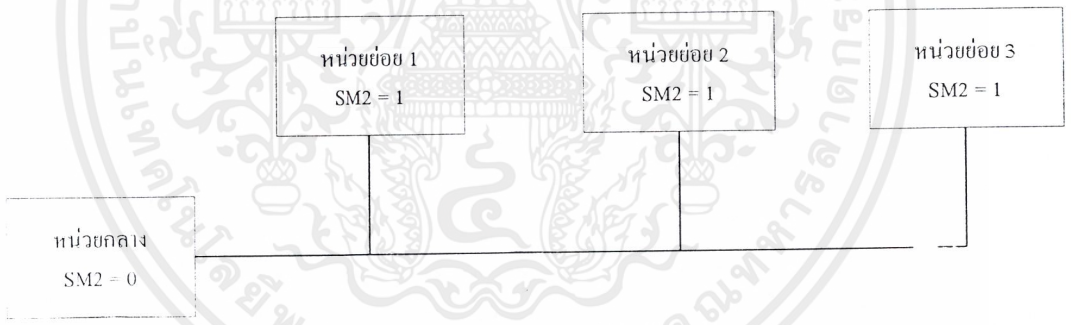
จากตัวอย่างเป็นการติดต่อระหว่างหน่วยกลางกับหน่วยย่อย "T01" ในขั้นแรกหน่วยย่อยทุกตัวจะถูกตั้งค่า SM2 เป็น "1" หน่วยกลางทำการติดต่อโดยส่งข้อมูลไป โดยในแต่ละไบต์กำหนด TB8 เป็น "1" (หน่วยย่อยถูกตั้งค่าเป็น SM2 เป็น "0" ไว้แล้ว)หน่วยย่อยแต่ละตัวจะเก็บไว้ในบิต RB8 เพื่อตรวจสอบกับค่า SM2 ที่ตั้งไว้โดยในสภาวะนี้ หน่วยย่อยแต่ละตัวจะถูกอินเทอร์รัพท์เพื่อตรวจสอบค่าแอดเดรส ดังรูปที่ 4.15

SOH,CDAT(0),Address("T01"),LRC#1,

STX,COMMAND("?"),SEP,SDP,DATA(""),EDB,ETX,LRC#2

---------->

หน่วยย่อย 1



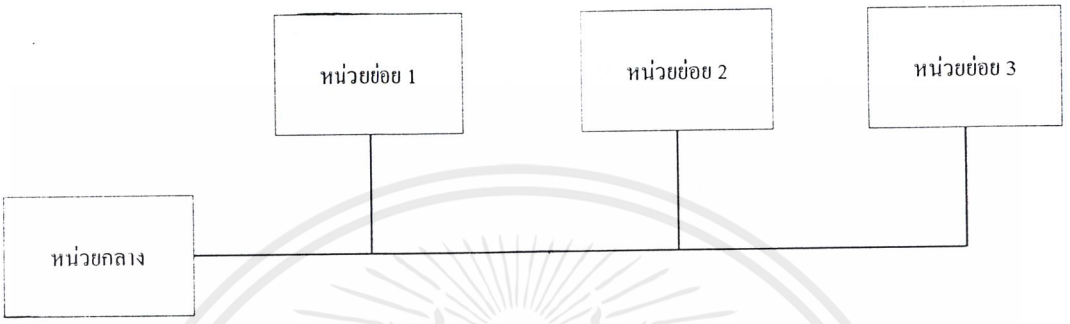
รูปที่ 4.16 แสดงพารามิเตอร์SCON ของหน่วยย่อยและหน่วยกลางเมื่อเริ่มต้นส่งข้อมูล

จากรูปที่ 4.16 หน่วยย่อยทุกตัวที่อยู่ภายในโครงข่ายได้รับข้อความจากหน่วยกลางพร้อมกับอ่านคาตำแหน่งของหน่วยย่อยจากข้อความดังกล่าวมาตรวจสอบซึ่งตำแหน่งที่หน่วยย่อยต้องการติดต่อด้วยคือ หน่วยย่อย1 (T01) สำหรับหน่วยย่อยแต่ละหน่วย จะทำการตรวจสอบค่าแอดเดรสถ้าตรงจะทำการตั้งค่า SM2 ให้เป็น "0" ถ้าค่าแอดเดรสไม่ตรงกับค่าแอดเดรสของหน่วยย่อยนั้นก็จะยังคงค่า SM2 ไว้คือ "1" จากตัวอย่างหน่วยย่อย 1 จะตั้งค่า SM2 เป็น "0" แล้วทำการส่งข้อมูลตอบรับให้หน่วยกลางรับรู้ว่า หน่วยย่อย 1 พร้อมทำการติดต่อกับหน่วยกลางแล้ว ในการส่งข้อมูลของหน่วยย่อย 1 จะส่งโดย TB8 เป็น "0" เพื่อให้หน่วยกลางสามารถรับข้อมูลได้และหน่วยย่อยอื่นที่ในเครือข่ายเดียวกันจะสามารถทำงานอื่นได้โดยไม่ถูกอินเทอร์รัพท์(หน่วยอื่นไม่สามารถรับข้อมูลได้) เพราะหน่วยอื่นๆถูกตั้งค่า SM2 ไว้ที่ "1" ดังรูปที่ 4.17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.9 การรับส่งข้อมูลอนุกรมของไมโครคอนโทรลเลอร์

การใช้งานในระบบการสื่อสารข้อมูล เลือกใช้โหมด 3 และใช้อัตราการรับส่งข้อมูล 9600 บิตต่อวินาที การเชื่อมต่อเป็นโครงข่ายดังรูปที่ 4.15



รูปที่ 4.15 แสดงการเชื่อมต่อการรับส่งข้อมูลแบบอนุกรม

จากรูปที่ 4.15 การทำงานแบบโครงข่ายเริ่มต้นด้วย

- หน่วยกลาง ส่งข้อความให้กับหน่วยย่อยที่ต้องการติดต่อด้วย โดยระบุตำแหน่งของตัวลูกที่ต้องการติดต่อด้วย
- หน่วยย่อยทุกตัวที่อยู่ภายในโครงข่าย ได้รับข้อความจากหน่วยกลาง พร้อมทั้งอ่านค่าตำแหน่งของหน่วยย่อยจากข้อความดังกล่าวมาตรวจสอบ
- ใช้หน่วยย่อยทุกตัว ตรวจสอบค่าตำแหน่งที่หน่วยกลางส่งมา ว่าเป็นตำแหน่งของตัวเองหรือไม่ ถ้าใช่ให้ส่งข้อความตอบรับกลับไปยังหน่วยกลาง ถ้าไม่ใช่ไม่ต้องตอบข้อความใดๆไปยังหน่วยกลาง

จากการทำงานของระบบโครงข่ายที่หน่วยกลางและหน่วยย่อยใช้ SN75176B เป็นชุดรับส่งข้อมูล จะประสบปัญหาในกรณีที่หน่วยกลางต้องการติดต่อด้วย ส่งข้อมูลตอบรับออกมาที่สายสัญญาณรับส่งข้อมูล นอกจากหน่วยกลางจะได้รับข้อความตอบรับแล้ว หน่วยย่อยตำแหน่งอื่นๆที่อยู่ในโครงข่ายก็ได้รับข้อความดังกล่าวด้วย ซึ่งข้อความตอบรับนี้ หน่วยกลางเท่านั้นที่ควรได้รับทำให้หน่วยย่อยตำแหน่งอื่นๆต้องถูกขัดจังหวะในการตรวจสอบข้อมูลที่เป็นการรับ-ส่งระหว่างหน่วยกลางกับหน่วยย่อยที่ต้องการติดต่อนั้น อาจทำให้ระบบทำงานได้อย่างไม่ค่อยมีประสิทธิภาพ อย่างไรก็ตามในส่วนการรับ-ส่งข้อมูลแบบอนุกรมของระบบไมโครคอนโทรลเลอร์ได้มีการแก้ปัญหาในส่วนนี้ โดยอาศัยเทคนิคดังต่อไปนี้

โดยการพิจารณารีจิสเตอร์ SCON จากรายละเอียดของบิต SM2 จะพบว่า

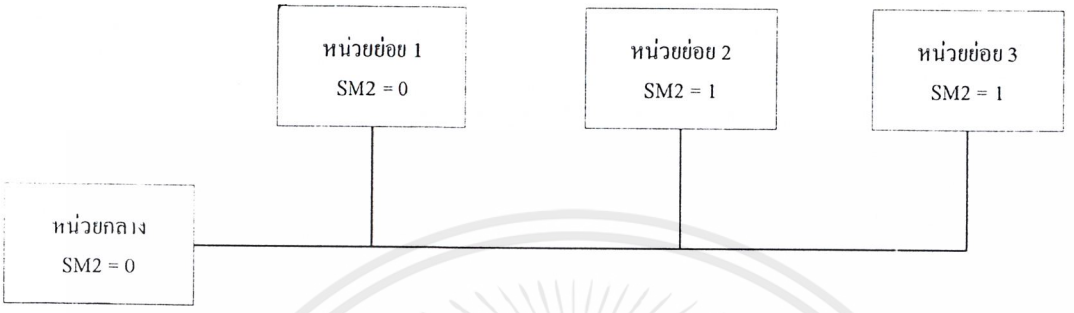
ถ้าตัวรับตั้งค่า SM2 = 1การอินเทอร์รัพท์การรับข้อมูลอนุกรมโดยข้อมูลอนุกรมจากตัวส่งจะเกิดขึ้นโดยตัวส่งต้องตั้งบิต TB8 = 1 ซึ่งค่านี้เมื่อตัวรับ รับมาแล้วจะถูกเก็บไว้ในบิต RB8 ของตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน่วยย่อย 1 ติดต่อเรียบร้อยแล้ว SOH,CDAT(0),Address(“C33”),LRC#1,

STX,COMMAND(“?”),SEP,SDP,DATA(“”),EDB,ETX,LRC#2

<---------------



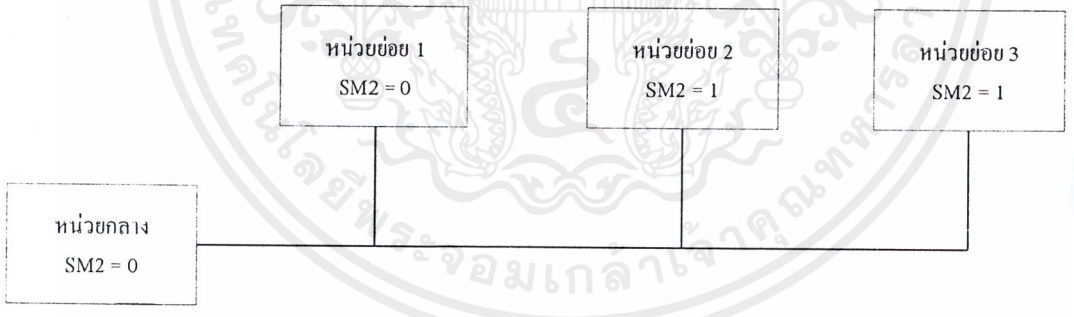
รูปที่ 4.17 แสดงพารามิเตอร์ SCON ของหน่วยกลางและหน่วยย่อยเมื่อส่งข้อความตอบรับ

หน่วยกลางส่งข้อความแสดงความถูกต้องของการตอบรับของข้อความให้กับหน่วยย่อย1 โดยต่อไปนี้กำหนด TB8 เป็น “0” เพื่อให้ติดต่อกับหน่วยย่อย 1 ได้ดังรูปที่ 4.18

SOH,CDAT(0),Address(“T01”),LRC#1,

STX,COMMAND(“A”),SEP,SDP,DATA(“”),EDB,ETX,LRC#2

--------------->



รูปที่ 4.18แสดงพารามิเตอร์ SCON ของหน่วยกลางและหน่วยย่อยเมื่อหน่วยกลางส่งข้อความ ACK จากรูปที่ 4.18 หน่วยกลางส่งข้อความ ACK ให้กับหน่วยย่อย 1 ซึ่งข้อความดังกล่าวมีเพียงหน่วยย่อย 1 ที่สามารถรับได้ ส่วนหน่วยย่อยตำแหน่งอื่นๆ ไม่สามารถรับได้และเมื่อหน่วยกลางส่งข้อความ ACK ให้กับหน่วยย่อย 1 แล้วพารามิเตอร์ SCON ของหน่วยกลางและหน่วยย่อย 1 จะกลับไปมีค่าเริ่มต้นใหม่แสดงดังรูปที่4.16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตรวจสอบเวลาเปิดประตู

SOH,CDAT(0),Address("TO1"),LRC#1,

STX,COMMAND("?"),SEP,SDP,DATA(""),EDB,ETX,LRC#2

---------->

SOH,CDAT(0),Address("C33"),LRC#1,

STX,COMMAND("?"),SEP,SDP,DATA(""),EDB,ETX,LRC#2

<----------

SOH,CDAT(0),Address("TO1"),LRC#1,

STX,COMMAND("A"),SEP,SDP,DATA(""),EDB,ETX,LRC#2

---------->

SOH,CDAT(0),Address("TO1"),LRC#1,

STX,COMMAND("t"),SEP,SDP,DATA("()"),EDB,ETX,LRC#2

---------->

SOH,CDAT(0),Address("C33"),LRC#1,

STX,COMMAND("t"),SEP,SDP,DATA("01:23:09/ 15/02/01@")

,EDB,ETX,LRC#2

<----------

SOH,CDAT(0),Address("TO1"),LRC#1,

STX,COMMAND("A"),SEP,SDP,DATA(""),EDB,ETX,LRC#2

ตัวส่งหน่วยกลาง ตัวรับหน่วยย่อย 1

ตรวจสอบเวลาปิดประตู

SOH,CDAT(0),Address("TO1"),LRC#1,

STX,COMMAND("?"),SEP,SDP,DATA(""),EDB,ETX,LRC#2

---------->

SOH,CDAT(0),Address("C33"),LRC#1,

STX,COMMAND("?"),SEP,SDP,DATA(""),EDB,ETX,LRC#2

<----------

SOH,CDAT(0),Address("TO1"),LRC#1,

STX,COMMAND("A"),SEP,SDP,DATA(""),EDB,ETX,LRC#2

---------->

SOH,CDAT(0),Address("TO1"),LRC#1,

STX,COMMAND("C"),SEP,SDP,DATA("()"),EDB,ETX,LRC#2

---------->

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SOH,CDAT(0),Address("C33"),LRC#1,

STX,COMMAND("C"),SEP,SDP,DATA("01:23:19/ 15/02/01@")

,EDB,ETX,LRC#2

<----------

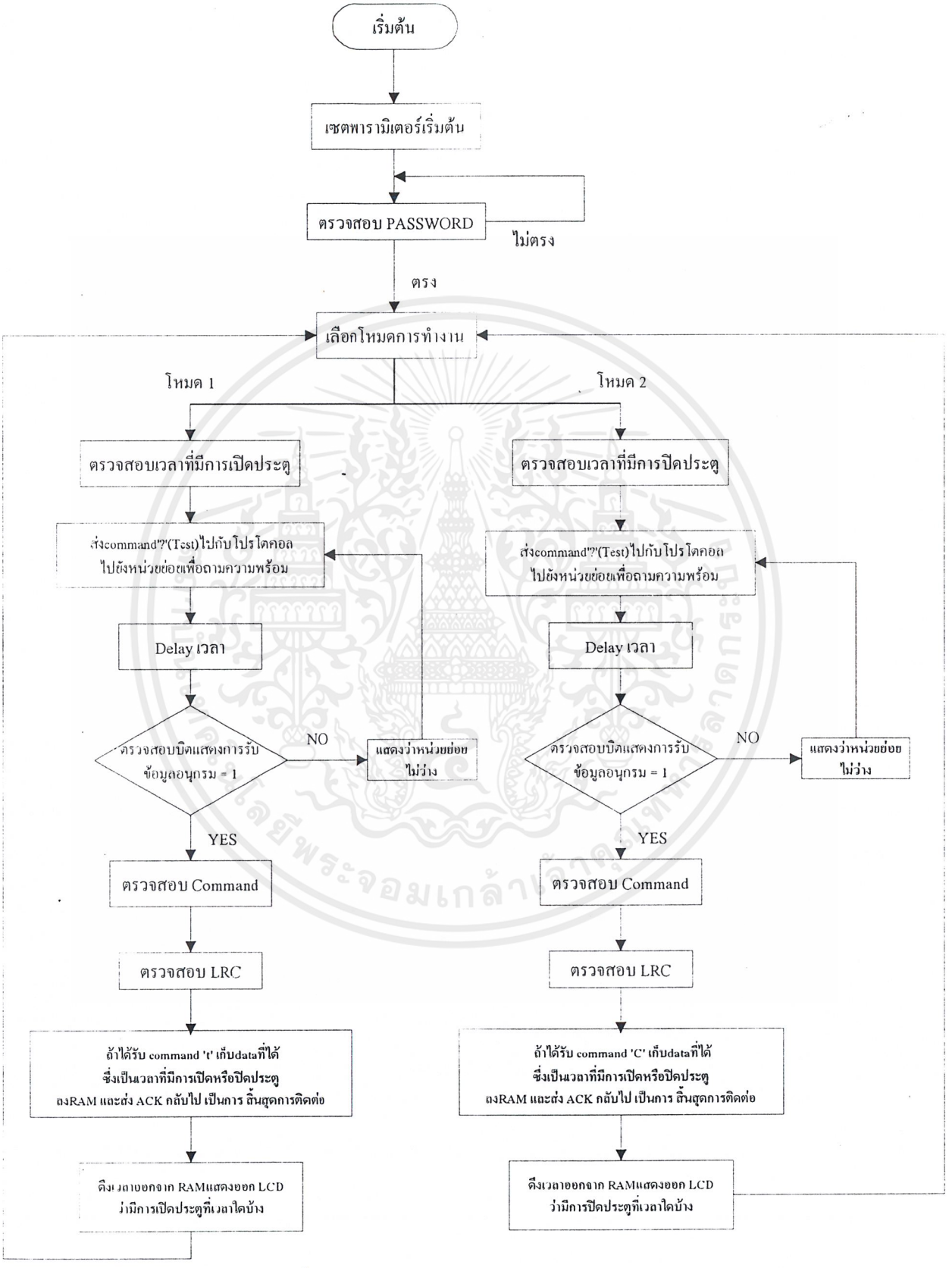
SOH,CDAT(0),Address("TO1"),LRC#1,

STX,COMMAND("A"),SEP,SDP,DATA(""),EDB,ETX,LRC#2

รูปที่ 4.19 แสดงโปรโตคอลในการติดต่อสื่อสารระหว่างหน่วยกลางและหน่วยย่อย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

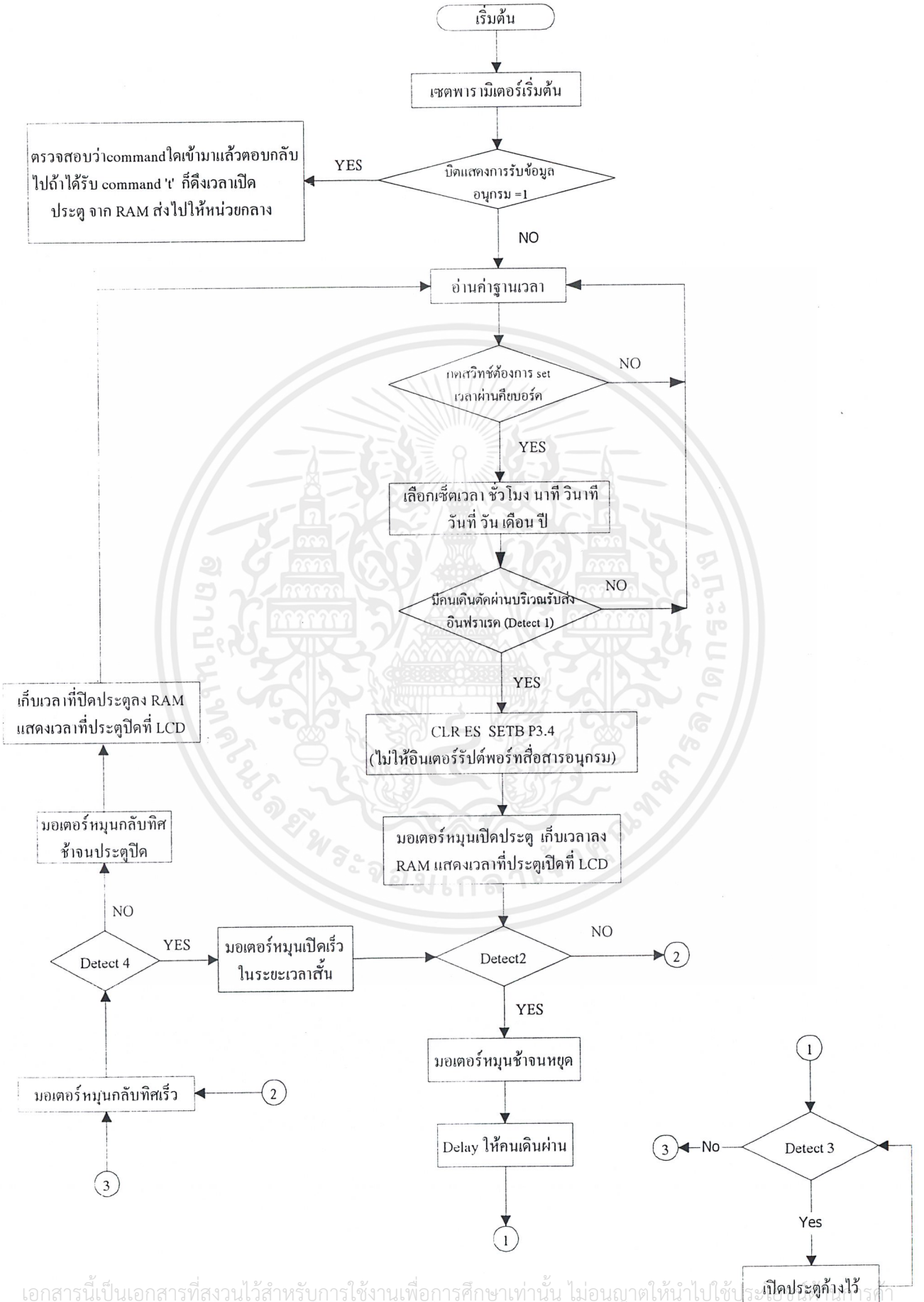
Flowchart แสดงการทำงานของหน่วยกลาง



รูปที่4.20 โฟลว์ชาร์ทแสดงการทำงานของหน่วยกลาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

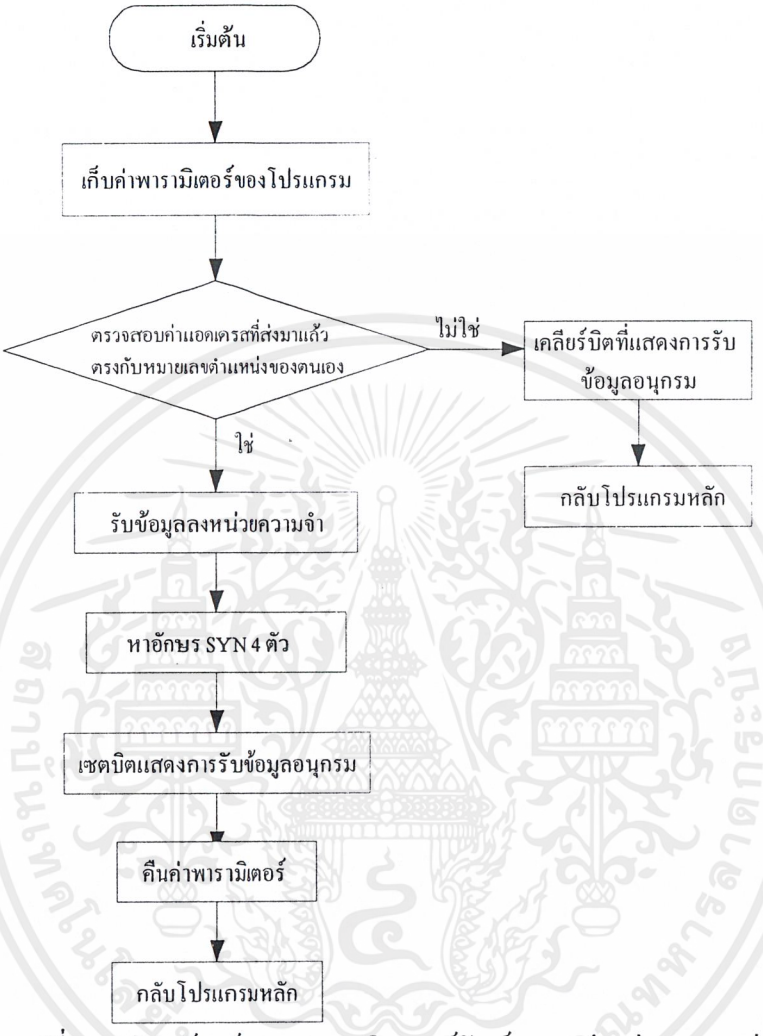
Flowchart แสดงการทำงานของหน่วยย่อย



รูปที่ 4.21โฟลว์ชาร์ทแสดงการทำงานของหน่วยย่อย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Flowchart บริการอินเตอร์รัปต์ของหน่วยย่อยและหน่วยกลาง



รูปที่ 4.22 โฟลว์ชาร์ทแสดงการอินเทอร์รัพท์ของหน่วยย่อยและหน่วยกลาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปและวิจารณ์ผลการทดลอง

5.1 สรุปผลการทดลอง

การทำงานของระบบควบคุมการเปิด-ปิดประตูอัตโนมัติ ภายใต้การควบคุมของไมโครคอนโทรลเลอร์ โดยการทำงานของหน่วยกลางและหน่วยย่อย ได้ใช้ระบบการสื่อสารข้อมูลแบบ RS-485 สามารถทำการรับส่งข้อมูลคือ เวลาที่มีการเปิด-ปิดประตูที่แต่ละหน่วยย่อยได้ระดับหนึ่ง โดยผ่านโปรโตคอลที่สร้างขึ้น ซึ่งมีรายละเอียดดังนี้

5.1.1 หน่วยกลาง เมื่อใส่รหัสผ่าน(Password) ได้ถูกต้องก็จะสามารถเข้าสู่การทำงานของหน่วยกลางได้ จากนั้นก็ทำการเลือกว่าจะตรวจสอบเวลาเปิดหรือปิดประตู จากนั้นก็เลือกแอดเดรสของหน่วยย่อยที่ต้องการติดต่อด้วย ซึ่งหมายเลขดังกล่าวจะถูกใส่เข้าไปเป็นแอดเดรสไบต์ในส่วนของการสื่อสารผ่านพอร์ทอนุกรม
และในโปรโตคอลเพื่อสร้างชุดของข้อมูลในการรับส่งระหว่างหน่วยกลางและหน่วยย่อย

- โดยถ้าหมายเลขของหน่วยย่อยที่ป้อนเข้าไปมีค่าเกิดที่ตั้งไว้ในโปรแกรม คือเกิน 32 หน่วยย่อย ก็จะมีข้อความแสดงที่หน้าจอ LCD ว่าใส่ค่าหมายเลขแอดเดรสของหน่วยย่อยไม่ถูกต้อง และกลับไปยังหน้าจอหลักที่ให้เลือกว่าจะตรวจสอบเวลาเปิดหรือปิดประตูอีกครั้ง

- ถ้าหน่วยกลางส่งโปรโตคอลของชุดข้อมูล มาในขณะที่หน่วยย่อยกำลังทำงานอยู่คือมีการตรวจเจอคนผ่านเข้ามาในบริเวณที่มีการรับส่งอินฟราเรด เก็บเวลาที่มีการเปิด-ปิดประตูลงหน่วยความจำและไปควบคุมการหมุนมอเตอร์อยู่ หน่วยกลางก็จะไม่สามารถติดต่อขอเวลาจากหน่วยย่อยที่ต้องการติดต่อได้ และที่หน้าจอ LCD ของหน่วยกลางก็จะมีข้อความแสดงว่าหน่วยย่อยกำลังทำงานอยู่ ให้ติดต่อมาใหม่อีกครั้ง และกลับไปยังหน้าจอหลักที่ให้เลือกว่าจะตรวจสอบเวลาเปิดหรือปิดประตูอีกครั้ง(ในการออกแบบการทำงานนี้ เราให้ความสำคัญกับงานทำงานของหน่วยย่อย คือ ตรวจสอบการรับส่งอินฟราเรด เก็บเวลาที่มีการเปิด-ปิดประตูลงหน่วยความจำ และไปควบคุมการหมุนมอเตอร์ ดังนั้นถ้าหน่วยย่อยส่งโปรโตคอลชุดของข้อมูลมาให้หน่วยย่อยในขณะนั้น จะไม่สามารถติดต่อกับหน่วยย่อยที่ต้องการได้ จะติดต่อได้ก็ต่อเมื่อหน่วยย่อยอยู่ในสภาวะที่ไม่ได้ตรวจเจอคนผ่านเข้ามาในบริเวณที่มีการรับส่งอินฟราเรด เท่านั้น)

-ถ้าหน่วยย่อยอยู่ในสภาวะที่ไม่ได้ตรวจเจอคนผ่านเข้ามาในบริเวณที่มีการรับส่งอินฟราเรด ก็จะทำการตรวจสอบแอดเดรสไบต์ถ้าตรงกับหมายเลขตำแหน่งของตน ก็จะรับโปรโตคอลของชุดข้อมูล และมีการสื่อสารกันระหว่างหน่วยกลางและหน่วยย่อย หน่วยย่อยส่งเวลาที่มีการเปิดหรือปิดประตูตามที่หน่วยกลางต้องการไปให้ จนเสร็จการสื่อสารหน่วยย่อยก็จะกลับไปทำงานปกติ คือแสดงเวลาออกหน้าจอ LCD ตรวจสอบการรับส่งอินฟราเรด แล้วไปควบคุมการหมุน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มอเตอร์ และบันทึกเวลาที่มีการเปิด-ปิดประตูลงหน่วยความจำต่อไป ส่วยหน่วยกลางเมื่อได้รับเวลาเปิดหรือปิดประตูที่ต้องการ จนเสร็จสิ้นการสื่อสารแล้ว ก็จะแสดงเวลาที่ได้รับมาออกหน้าจอ LCD ได้อย่างถูกต้อง

- ในแต่ละหน่วยย่อยได้มีการตั้งเวลาเปิดและปิดประตูไว้ ผลที่ได้คือเมื่อยังไม่ถึงเวลาเปิดประตูแม้ว่ามีการตรวจเจอคนผ่านเข้ามาในบริเวณที่มีการรับส่งอินฟราเรด มอเตอร์ก็จะไม่หมุนและประตูก็จะไม่มีการเปิดออก จนกว่าจะถึงเวลาที่ตั้งไว้ในโปรแกรม และเมื่อถึงเวลาปิดประตูแล้วก็จะได้ผลเหมือนกันคือ ประตูจะไม่มีการเปิดออก

5.1.2 หน่วยย่อย

- ภาคส่งอินฟราเรด ในที่นี้ได้ออกแบบให้ตัวส่งไดร์ฟกระแสให้ LED หลายๆตัวที่ต่อขนานกัน (การต่อขนานกันมีข้อดีตรงถ้า LED ตัวใดตัวหนึ่งขาด วงจรจะยังคงทำงานได้และใช้ไฟเลี้ยงวงจรไม่ต้องสูงมาก แต่มีข้อเสียคือ วงจรจะใช้กระแสค่อนข้างมาก) ที่ต้องใช้ LED หลายดวงเพราะLED แต่ละดวงปล่อยพลังงานออกมาในมุมที่ไม่กว้างมากนัก และการใช้ LED หลายดวงทำให้ Power ที่ LED ปล่อยออกมาเกิดการซ้อนทับรวมกัน ทำให้มี Intensity ที่สูง ทำให้ตัวรับ Detect รับพลังงานจากตัวส่งได้ดีขึ้น

- ภาครับอินฟราเรด จะต้องมีคุณสมบัติที่ดี เช่น มีความไวสูง มีความเที่ยงตรง มีความสามารถในการลดสัญญาณรบกวน(Noise) ได้ดี ซึ่งในที่นี้ได้นำ SL486 ซึ่งเป็นไอซีที่มีคุณสมบัติดังกล่าวมาใช้ ทำให้การรับอินฟราเรดค่อนข้างมีประสิทธิภาพ โดยตัวรับสามารถตรวจจับอินฟราเรดเมื่อมีคนเดินผ่านเข้ามาได้ในระยะทางประมาณ1.5 เมตร และมุมกว้างประมาณ 120 องศา (ใช้ LED ตัวส่ง 6 ดวง) โดยเอาท์พุทจาก SL486 จะเป็นระดับแรงดัน DC ที่มีการเปลี่ยนแปลงตามพลังงานที่ตัวรับตรวจจับได้ จึงต้องเอาเอาท์พุทจาก SL486 มาเข้า Comparator เพื่อให้ได้เอาท์พุทเป็นระดับแรงดัน 5V, 0V เพื่อไปป้อนเข้าส่วนของไมโครคอนโทรลเลอร์

- ไมโครสวิตซ์ ตัวที่อยู่ที่ขอบของโครงประตูสามารถช่วยป้องกันความเสียหายที่อาจเกิดขึ้นกับวงจรขับมอเตอร์ โดยเมื่อระบบทำงานผิดพลาด มอเตอร์จะหมุนไม่หยุดไปชนไมโครสวิตซ์ที่ขอบประตู แล้วประตูก็จะหยุดหมุนและมีเสียงเตือนดังขึ้นประมาณ 40 วินาที เป็นการบอกว่าระบบเกิดการทำงานผิดพลาดขึ้น

ส่วนไมโครสวิตซ์ที่อยู่ที่ขอบของบานประตู เมื่อประตูปิดบานประตูทั้งสองข้างจะชนกันและไปกดให้ไมโครสวิตซ์ทำงาน มอเตอร์ก็จะหยุดหมุนและไปตรวจสอบการรับส่งอินฟราเรดเพื่อเปิดประตูออกในครั้งต่อไปได้ถูกต้อง

5.2 ปัญหาและอุปสรรคในการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- จากการทดลองการทำงานในการติดต่อสื่อสารระหว่างหน่วยกลางและหน่วยย่อย ผลคือ หน่วยกลางสามารถดึงข้อมูลที่เป็นเวลาที่มีการเปิดหรือปิดประตู ทำงานได้ตามวัตถุประสงค์ที่ตั้งไว้ แต่ถ้าระบบทำงานไปนานๆ แล้วเกิดไฟฟ้าขัดข้อง เช่น ไฟตก จะทำให้หน่วยกลางหรือหน่วยย่อยรีเซ็ตหรือหยุดการทำงานไปเลย

- เนื่องจากการส่งและรับอินฟราเรด จะตั้งตัวส่ง ตัวรับอยู่ด้วยกัน ดังนั้นตัวรับจะรับสัญญาณได้ก็ต่อเมื่อสัญญาณที่ตัวส่งปล่อยออกมา มาสะท้อนโดนวัตถุหรือสิ่งมีชีวิตแล้วสะท้อนเข้ายังตัวรับ ซึ่งติดตั้งอยู่ใกล้ตัวส่ง ซึ่งทำให้ตัวส่งต้องปล่อยพลังงานออกมาที่มีความเข้มมากพอ ตัวรับจึงจะตรวจสอบได้ และในทางปฏิบัติการจัดตำแหน่งของ LED ตัวส่งทั้ง 6 ดวงที่ต่างกัน ก็จะได้มุมและระยะทางที่ตรวจจับอินฟราเรดได้ต่างกันทำให้ค่อนข้างยากในการจัดตำแหน่งของ LED และควบคุมกระแสที่ป้อนให้ LED เพราะจะมีผลต่อระยะทางและมุมที่ตรวจจับอินฟราเรดค่อนข้างมาก

- ในการเขียนโปรแกรมเพื่อทำการติดต่อสื่อสารผ่านพอร์ทอนุกรมระหว่างหน่วยกลางและหน่วยย่อย ได้พบปัญหามากมาย ซึ่งปัญหาดังกล่าวสามารถแก้ไขได้โดยการศึกษา เข้าใจในหลักการของการสื่อสารระหว่างไมโครคอนโทรลเลอร์อย่างถูกต้อง และการฝึกฝนบ่อยๆในการเขียนโปรแกรม

5.3 วิจารณ์โครงงาน

ในส่วนของโปรแกรมที่ไปควบคุมการทำงานของวงจรขับมอเตอร์ สามารถพัฒนาโปรแกรมให้มีประสิทธิภาพมากกว่านี้ได้ เช่น ตรวจจับสัญญาณที่มาจากการรับส่งอินฟราเรด ให้บ่อยขึ้นและเร็วขึ้น

ในส่วนของการเก็บเวลาที่มีการเปิด-ปิดประตูลงหน่วยความจำ ในที่นี้ได้ออกแบบไว้ให้มีการเก็บเวลาดังกล่าวประมาณอย่างละ 50 ครั้ง ซึ่งถ้าต้องการเก็บเวลาให้มากกว่านี้ก็สามารถทำได้โดยขยายหน่วยความจำ RAM เพิ่มก็จะสามารถเก็บเวลาที่มีการเปิด-ปิดประตูได้มากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

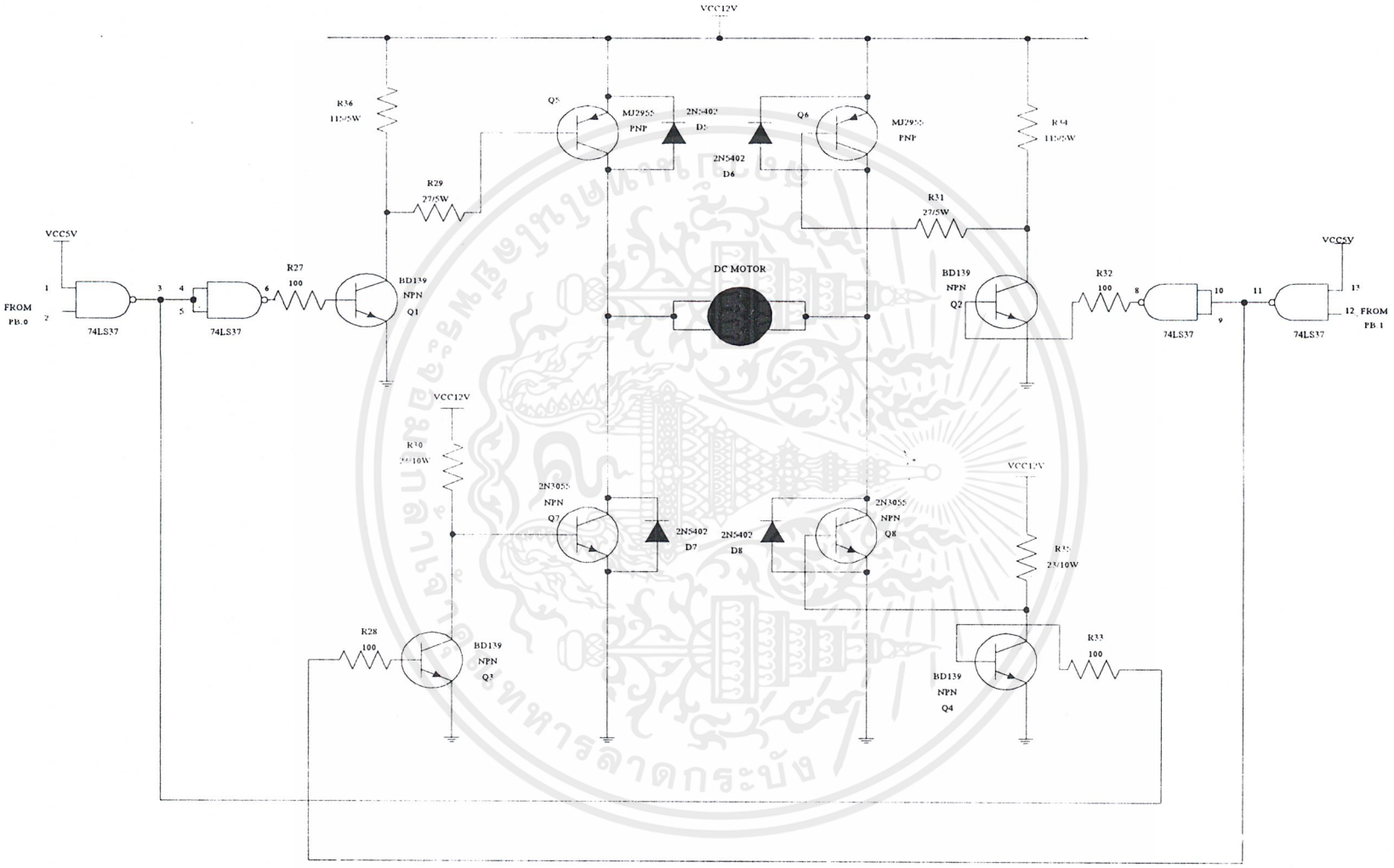
ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางการเปรียบเทียบมาตรฐานการสื่อสารข้อมูลของ EIA

พารามิเตอร์	RS-232-C	RS-423-A	RS-422-A	RS-485
โหมดการทำงาน	Single-ended	Single-ended	Differential	Differential
จำนวนของตัวรับ และตัวส่งที่ยอมรับได้	1 ตัวส่ง 1 ตัวรับ	1 ตัวส่ง 10 ตัวรับ	1 ตัวส่ง 10 ตัวรับ	32 ตัวส่ง 32 ตัวรับ
ความยาวของคู่สาย สัญญาณรับส่งข้อมูล	50 ฟุต	4000 ฟุต	4000 ฟุต	4000 ฟุต
อัตราการส่งข้อมูล สูงสุด (บิตต่อวินาที)	20 k	100k	10M	10M
แรงดันไฟฟ้าโหมดร่วม สูงสุด	±2.5V	±6V	6V -2.5V	12V -7V
Drive output	±5V ต่ำสุด ±15V สูงสุด	±3.6V ต่ำสุด ±6.0V สูงสุด	±2V ต่ำสุด	±1.5V ต่ำสุด
Drive load Ω	3k ถึง 7k	450 ต่ำสุด	100 ต่ำสุด	60 ต่ำสุด
Drive slew rate	30V /us สูงสุด	-	NA	NA
กระแสลิมิต เมื่อเอาท์พุทลัดวงจร	500mA ลัดวงจรกับ VCC หรือ GND	150mA ลัดวงจรกับ GND	150mA ลัดวงจรกับ GND	150mA ลัดวงจรกับ GND 250mA ลัดวงจรกับ 8V หรือ 12V
ค่าความต้านทาน เอาท์พุทของตัวส่ง	NA-power on 300-power off	NA-power on 60k-power off	NA-power on 60k-power off	120k power on , off
ค่าความต้านทานอิน พุทของตัวรับ	3k ถึง 7k	4k	4k	12k
ความไวของตัวรับ	±3V	±200mV	±200mV	±200mV

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



วงจรขับมอเตอร์

เอกสารอ้างอิง

1.ซีเอ็ดยูเคชั่น," รีโมตเครื่องควบคุมไร้สาย" ,ซีเอ็ดยูเคชั่น, 50 หน้า,2538

2.รศ.สมยศ จุณณะปิยะ ," การประยุกต์ใช้งานไมโครคอนโทรลเลอร์ตระกูล MCS-51 ",คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ,2541

3.Leonard S.Bobrow ,"Fundamentals of Electrical Enginerring",OXFORD UNIVERSITY PRESS ,704p,1996

4. "ANT-31PJ VERSION 2.0 PROJECT EMBEDDED CONTROLLER", บริษัทศิลารีเสิร์ช จำกัด

5.ปรเมษฐ์ ประนยานันทน์ , ปิยพงศ์ เผ่าวานิช , " คู่มือการประยุกต์ใช้งานไมโครคอนโทรลเลอร์ MCS-51" , บริษัทซีเอ็ดยูเคชั่น จำกัด (มหาชน) , 2536

6.ETT , "DOT MATRIX LCD MODULE" ,ETT CO.,LTD

7. Somphop Purivigraipong , "Access Control System using Magnetic Card"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงงานระบบควบคุมการเปิด-ปิดประตูโดยใช้ไมโครคอนโทรลเลอร์นี้สำเร็จได้ก็เพราะได้รับความรู้ ความช่วยเหลือและคำแนะนำจาก ผศ.พลผดุง ผดุงกุล (อาจารย์ที่ปรึกษา) ,อาจารย์ประจำภาควิชาอิเล็กทรอนิกส์ทุกท่าน นายอนุทิน มณีอนันต์สุข(พี่เอ) , นายเอกสิทธิ์ ดำสง(พี่เอ็กซ์)ที่ได้ให้ความรู้เรื่องโปรโตคอล และเพื่อนๆ ,รุ่นพี่ที่ได้ให้ความช่วยเหลือ แนะนำอุปกรณ์ต่างๆ ให้ยืมหนังสือ ให้ข้อมูลเพื่อเป็นประโยชน์ในโครงงานนี้

ทางฝ่ายผู้จัดทำ จึงขอขอบคุณทุกๆท่านมา ณ ที่นี้

นริสา ดิษผึ้ง

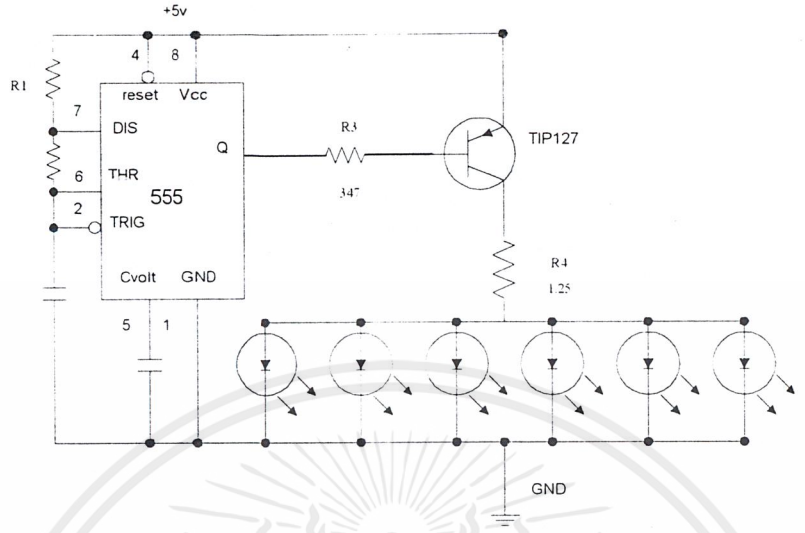
(นางสาวนริสา ดิษผึ้ง)

บุญชัย ตันติวุฒิพงศ์

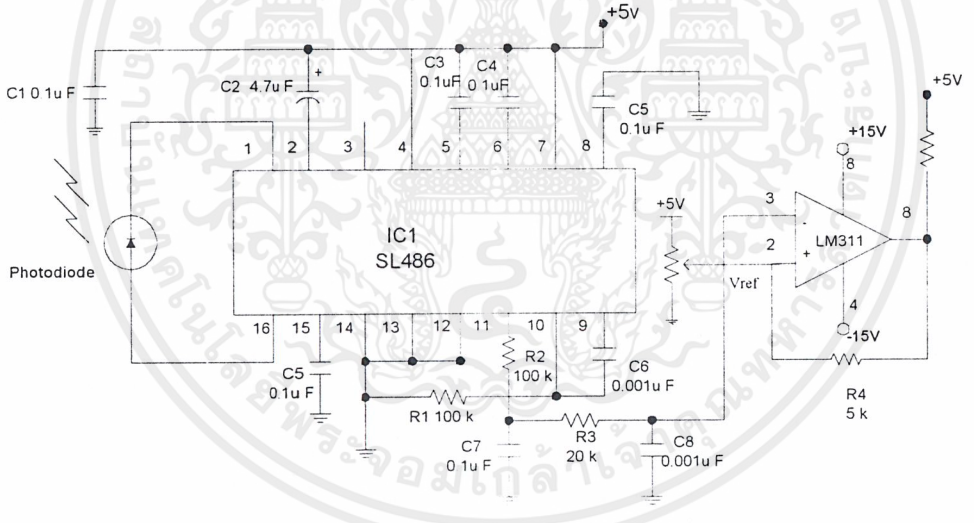
(นายบุญชัย ตันติวุฒิพงศ์)

ผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

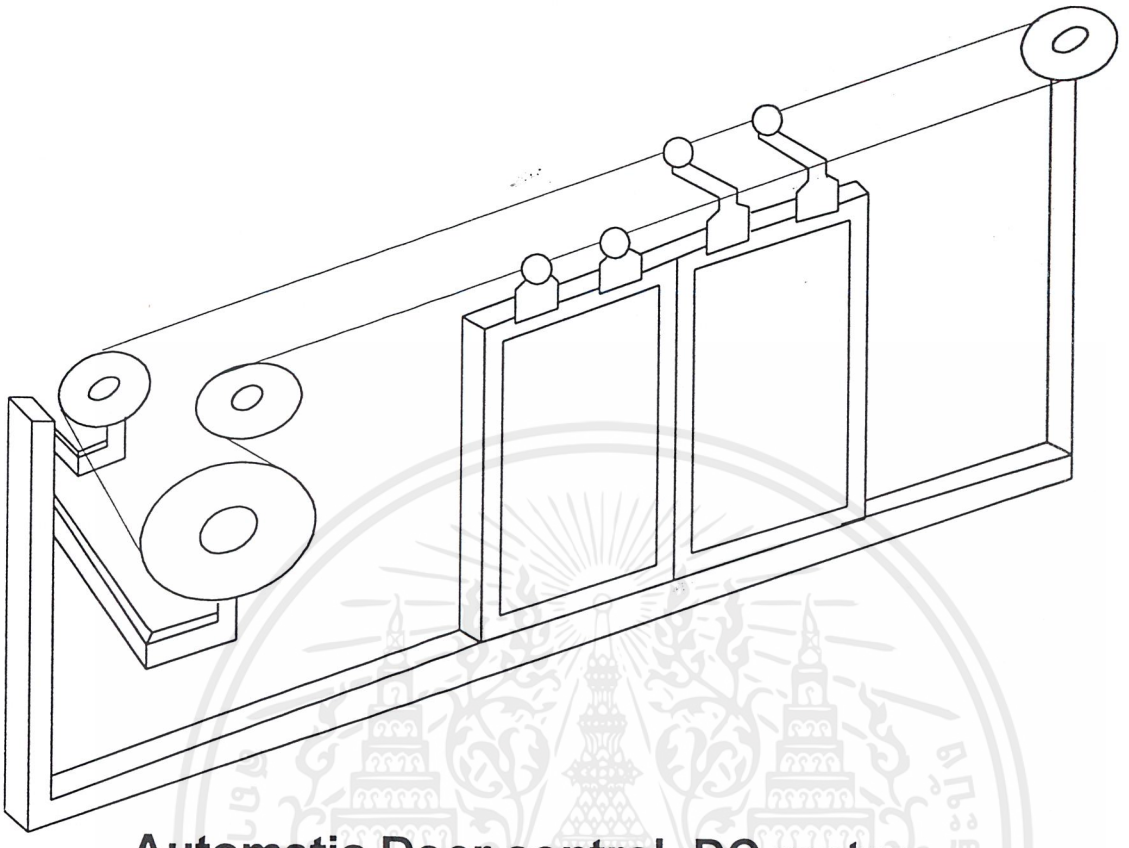


รูปแสดงวงจรส่วนของภาคส่งสัญญาณอินฟราเรด



แสดงวงจรส่วนของภาครับสัญญาณอินฟราเรดและส่วนของ Comparator

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Automatic Door control DC.motor by use microcontroller

รูปแสดงโครงสร้างของประตู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้