สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เครื่องรับส่งสัญญาณด้วยรีโมทคอนโทรล

IR REMOTE CONTROL TRANSMITTER AND RECEIVER

T046433

โดย

นายวินัย พรัมมะณี

นายอภิฉัตร จินตนากุล

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาอุตสาหกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2544

เลขหมู่.....
เลขทะเบียน 46433
วัน, เดือน, ปี 1 เม.ย. 2546

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	เครื่องรับส่งสัญญาณด้วยรีโมทคอนโทรล
TITLE	**IR REMOTE CONTROL TRANSMITTER AND RECEIVER**
นักศึกษา	นายวินัย พรัมมะณี รหัสประจำตัว 43015744
	นายอภิฉัตร จินตนากุล รหัสประจำตัว 43015758
อาจารย์ผู้ควบคุมปริญญานิพนธ์	ผศ.ไพศาล สิทธิโยภาสกุล
ระดับการศึกษา	ปริญญาอุตสาหกรรมศาสตรบัณฑิต
ภาควิชา	วิศวกรรมสารสนเทศ
ปีการศึกษา	2544

ปริญญานิพนธ์ฉบับนี้ได้รับการอนุมัติให้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรอุตสาหกรรมศาสตรบัณฑิต คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

.....

(ผศ.ไพศาล สิทธิโยภาสกุล)

อาจารย์ผู้ควบคุมปริญญานิพนธ์

ลิขสิทธิ์ของคณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	เครื่องรับส่งสัญญาณด้วยรีโมทคอนโทรล
นักศึกษา	นายวินัย พรัมมะณี รหัสประจำตัว 43015744
	นายอภิฉัตร จินตนากุล รหัสประจำตัว 43015758
อาจารย์ผู้ควบคุมปริญญานิพนธ์	ผศ.ไพศาล สิทธิโยภาสกุล
ระดับการศึกษา	ปริญญาอุตสาหกรรมศาสตรบัณฑิต
ภาควิชา	วิศวกรรมสารสนเทศ
ปีการศึกษา	2544

บทคัดย่อ

ปริญญานิพนธ์นี้เป็นการนำเสนอ การออกแบบเครื่องรับส่งสัญญาณด้วยรีโมทคอนโทรล โดยเป็นการนำตัวไมโครคอนโทรลเลอร์เบอร์ AT89C52 เป็นหัวใจหลัก และแสดงผลผ่านจอ LCD สามารถรับข้อมูลได้ทางคีย์เมตริกซ์ขนาด 5X3 โดยตัวไมโครคอนโทรลเลอร์นี้จะเป็นตัวประมวลผล ซึ่งมีการทำงานอยู่ 3 ฟังก์ชั่น คือฟังก์ชั่นการอ่าน ฟังก์ชั่นการส่ง และการแสดงผล ฟังก์ชั่นการอ่านจะแสดงสัญญาณเป็นช่วง Ton และ Toff ของรูปคลื่นซึ่งมอดูเลตด้วยระบบ PWM และ PCM ในฟังก์ชั่นการส่ง จะสามารถควบคุมด้วยคีย์แพด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PROJECT TITLE	**IR REMOTE CONTROL TRANSMITTER AND RECEIVER**
STUDENT	**Mr. Winai Prummanee No. 43015744**
	Mr. Apichat Jintanakul No. 43015758
ADVISOR	**Asst.Prof. Paisan Sithiyopasakul**
COURSE	**Bachelor of Industrial Technology in Electronics**
DEPARTMENT	**Information Engineering**
YEAR	**2001**

ABSTRACT

This thesis presents to design an infrared remote control transmitter and receiver. By using the Microcontroller is AT89C52, with display by LCD (2 line×20 bit) and receiver input data by key matrix (5×3). The Microcontroller is being used for processing, with having 3 functions as following read function, send function and display function. Read function will show signal mean Ton and Toff of pulse, which modulate with PWM and PCM. In send function able to control by keypad.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ขอขอบพระคุณ ผศ.ไพศาล สิทธิโยภาสกุล ที่ให้คำแนะนำ คำปรึกษาเพื่อเป็นแนวทางในการแก้ไข และดูแลตรวจสอบจนโครงงานนี้ สำเร็จลุล่วงลงได้ด้วยดี ขอขอบคุณ คุณเมาลี กลิ่นหอม ที่ให้คำแนะนำในส่วนของฮาร์ดแวร์ และผู้ที่เกี่ยวข้องกับโครงงานนี้ทุกท่านที่มิได้เอ่ยนามในที่นี้ สำหรับเครื่องมือ อุปกรณ์ต่างๆ และกำลังใจที่มีให้กับคณะผู้จัดทำ ขอบพระคุณอย่างยิ่งและลืมมิได้ บิดา มารดา ของคณะผู้จัดทำที่ให้โอกาสต่างๆ ทั้งการเลี้ยงดู การศึกษา จนทำให้คณะผู้จัดทำประสบความสำเร็จทางการศึกษาในวันนี้

คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูป	VI
สารบัญตาราง	IX
บทที่ 1 บทนำ	1
1.1 ขอบเขตโครงงาน	1
1.2 เนื้อหาแต่ละบท	1
บทที่ 2 ทฤษฎีทั่วไป	3
2.1 ไมโครคอนโทรลเลอร์ในตระกูล MCS-51	3
2.1.1 โครงสร้างของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช	3
2.1.2 การจัดขาของไมโครคอนโทรลเลอร์ MCS-51	4
2.1.3 โครงสร้างและการทำงานของพอร์ต	9
2.1.4 การทำงานของไมโครคอนโทรลเลอร์ MCS-51	12
2.1.5 การจัดหน่วยความจำของไมโครคอนโทรลเลอร์ MCS-51	15
2.2 ทฤษฎีเบื้องต้นสเกนกีย์	26
2.3 การใช้งาน DOT MATRIX LCD MODULE	28
2.3.1 การต่อเข้ากับระบบของไมโครคอนโทรลเลอร์ MCS-51	30
2.3.2 รายละเอียดของแต่ละคำสั่ง	32
2.3.3 การอ่านและเขียนข้อมูลกับ DDRAM/ CGRAM	35
2.3.4 การเขียนโปรแกรมควบคุม	35
2.4 ทฤษฎีอินฟราเรดรีโมทคอนโทรล	36
2.4.1 สัญญาณมาตรฐานรีโมทคอนโทรล	38
2.4.2 รหัสสัญญาณ รีโมทคอนโทรลของเครื่องใช้ไฟฟ้าในปัจจุบัน	43

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	หน้า
บทที่ 3 การออกแบบวงจร	49
3.1 การออกแบบซอฟแวร์	49
3.1.1ส่วนประมวลผล	49
3.1.2 ภาคจ่ายไฟ	50
3.1.3 ส่วนของวงจรรับสัญญาณอินฟราเรดรีโมทคอนโทรล	51
3.1.4 ส่วนของวงจรส่งสัญญาณอินฟราเรดรีโมทคอนโทรล	51
3.1.5 ส่วนแสดงผล	53
3.1.6 ส่วนของคีย์เมตริกซ์	54
3.2 การออกแบบซอฟแวร์ควบคุมวงจร	54
3.2.1 การออกแบบซอฟแวร์ส่วนแสดงผล	54
3.2.2 การออกแบบซอฟแวร์ส่วนควบคุมของคีย์แพด	55
3.2.3 การออกแบบซอฟแวร์ส่วนรับสัญญาณอินฟราเรด	55
3.2.4 การออกแบบซอฟแวร์ส่วนทดสอบค่าสัญญาณอินฟราเรด	55
บทที่ 4 การทดลองและผลลัพธ์ที่ได้	65
4.1 คุณสมบัติเครื่องรับส่งสัญญาณด้วยรีโมทคอนโทรล	65
4.2 โครงสร้างส่วนต่างๆของโครงงาน	65
4.3 การใช้งานเครื่องรับส่งสัญญาณด้วยรีโมทคอนโทรล	66
บทที่ 5 สรุปและวิจารณ์ผลการทดลอง	72
บรรณานุกรม	74
ภาคผนวก	75
ภาคผนวก ก รูปวงจรรวมของโครงงาน และแผนภาพลายวงจรของโครงงาน	
ภาคผนวก ข โปรแกรมเครื่องรับส่งสัญญาณด้วยรีโมทคอนโทรล	
ภาคผนวก ค Data Sheet ของตัวรับสัญญาณอินฟราเรด เบอร์ IRM8601	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

หน้า

รูปที่ 2.1 โครงสร้างพื้นฐานไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช AT89CXX 5

รูปที่ 2.2 โครงสร้างพื้นฐานไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช AT89SXX 5

รูปที่ 2.3 รายละเอียดโครงสร้างหลักของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช 7

รูปที่ 2.4 การจัดขามาตรฐานของไมโครคอนโทรลเลอร์ MCS-51 ในอนุกรม AT89C5X 8

รูปที่ 2.5 วงจรภายในของพอร์ตทุกพอร์ตในไมโครคอนโทรลเลอร์ 11

รูปที่ 2.6 วงจรพูลอัปภายในพอร์ตไมโครคอนโทรลเลอร์ 12

รูปที่ 2.7 ไซเกิลการทำงานของไมโครคอนโทรลเลอร์ 13

รูปที่ 2.8 ไดอะแกรมเวลาแสดงการติดต่อและเข้าถึงหน่วยความจำโปรแกรมภายนอกของ
ไมโครคอนโทรลเลอร์ 14

รูปที่ 2.9 การจัดสรรหน่วยความจำโปรแกรมของไมโครคอนโทรลเลอร์ 16

รูปที่ 2.10 การเชื่อมต่อหน่วยความจำโปรแกรมภายนอกของไมโครคอนโทรลเลอร์ 18

รูปที่ 2.11 การเชื่อมต่อหน่วยความจำข้อมูลภายนอกของไมโครคอนโทรลเลอร์ 19

รูปที่ 2.12 การจัดสรรพื้นที่ของหน่วยความจำข้อมูลภายในส่วนล่างของ
ไมโครคอนโทรลเลอร์ 20

รูปที่ 2.13 โครงสร้างของหน่วยความจำข้อมูลภายในส่วนบนของไมโครคอนโทรลเลอร์ 21

รูปที่ 2.14 การจัดสรรพื้นที่ของรีจิสเตอร์ฟังก์ชั่นพิเศษ (SFR) 21

รูปที่ 2.15 วงจรของสวิทช์แบบเมตริกซ์หรือคีย์แพด 26

รูปที่ 2.16 วงจรเชื่อมต่อคีย์แพดเข้ากับไมโครคอนโทรลเลอร์ 27

รูปที่ 2.17 การต่อ LCD MODULE เข้ากับไมโครคอนโทรลเลอร์ 30

รูปที่ 2.18 การทำงานของ LCM 31

รูปที่ 2.19 ลักษณะทางลอจิกในระบบสแกนพัลซ์ 37

รูปที่ 2.20 ดาต้าสตรีม 38

รูปที่ 2.21 ตัวอย่างสัญญาณรีโมทคอนโทรล 39

รูปที่ 2.22 เครื่องส่งรีโมทคอนโทรลที่ใช้เบอร์ไอซีเบอร์ MN 603303 40

รูปที่ 2.23 สแกนพัลซ์ที่ออกมาจากคีย์เอาต์พุตขาต่างๆ 41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

	หน้า
รูปที่ 2.24 สแกนพัลซ์ของเบอร์ MN 6030B	42
รูปที่ 3.1 ขาต่อใช้งานของไมโครคอนโทรลเลอร์	49
รูปที่ 3.2 วงจรภาคจ่ายไฟ	51
รูปที่ 3.3 ส่วนรับสัญญาณอินฟราเรด	51
รูปที่ 3.4 วงจรควบคุมการส่งสัญญาณอินฟราเรดรีโมทคอนโทรล	53
รูปที่ 3.5 การต่อวงจรเพื่อควบคุมจอแอลซีดี	53
รูปที่ 3.6 วงจรของคีย์เมตริกซ์	54
รูปที่ 3.7 โฟลวชาร์ตโปรแกรมหลัก กำหนดการทำงานของเครื่องรับส่งสัญญาณด้วยรีโมทคอนโทรล	56
รูปที่ 3.8 โฟลวชาร์ต โปรแกรมย่อยแสดงรหัสสัญญาณอินฟราเรดที่รับเข้ามา	57
รูปที่ 3.9 โฟลวชาร์ต โปรแกรมย่อยส่วนรับอินฟราเรด	58
รูปที่ 3.10 โฟลวชาร์ต โปรแกรมย่อยการส่งสัญญาณที่รับเข้ามาเพื่อทดสอบ	60
รูปที่ 3.11 โฟลวชาร์ต โปรแกรมย่อยสุ่มเก็บสัญญาณด้วยฐานเวลา 100 uS	61
รูปที่ 3.12 โฟลวชาร์ต โปรแกรมย่อยแสดงผลข้อความ	61
รูปที่ 3.13 โฟลวชาร์ต โปรแกรมย่อยการอ่านค่าคีย์แพด	62
รูปที่ 3.14 โฟลวชาร์ต โปรแกรมย่อยการส่งสัญญาณของ SONY และ JVC	63
รูปที่ 3.15 โฟลวชาร์ต โปรแกรมย่อยในการสร้างพัลซ์	64
รูปที่ 4.1 โครงสร้างส่วนต่างๆของโครงงาน	65
รูปที่ 4.2 หน้าจอเมื่อเริ่มเปิดเครื่อง	66
รูปที่ 4.3 หน้าจอเมื่อเครื่องพร้อมที่จะทำงานตามฟังก์ชั่น	67
รูปที่ 4.4 ตัวอย่างของค่าที่อ่านได้	67
รูปที่ 4.5 หน้าจอเมื่อเครื่องพร้อมรับสัญญาณอินฟราเรดจากรีโมทคอนโทรล	67
รูปที่ 4.6 หน้าจอเมื่อเครื่องรับสัญญาณเรียบร้อยแล้ว	68
รูปที่ 4.7 หน้าจอเมื่ออ่านค่าของสัญญาณที่รับมาจากรีโมทคอนโทรล	68

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

หน้า

รูปที่ 4.8 หน้าจอการเลือกยี่ห้อของเครื่องใช้ไฟฟ้า 69

รูปที่ 4.9 หน้าจอการใช้คอนโทรลคีย์ 69

รูปที่ 4.10 สัญญาณรีโมทคอนโทรลต้นแบบยี่ห้อ SONY กับสัญญาณที่เลียนแบบขึ้นมา 69

รูปที่ 4.11 สัญญาณรีโมทคอนโทรลต้นแบบยี่ห้อ JVC กับสัญญาณที่เลียนแบบขึ้นมา 70

รูปที่ 4.12 ภาพถ่ายภายในของเครื่องรับส่งสัญญาณด้วยรีโมทคอนโทรล 70

รูปที่ 4.13 ภาพถ่ายด้านบนของเครื่องรับส่งสัญญาณด้วยรีโมทคอนโทรล 71

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้า
ตารางที่ 2.1 รายละเอียดโดยสรุปบางส่วนของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช	6
ตารางที่ 2.2 หน้าที่พิเศษของพอร์ต 1 ในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช	10
ตารางที่ 2.3 การเลือกแบงก์ของหน่วยความจำส่วนล่างเพื่อติดต่อกับรีจิสเตอร์แบงก์ R0-R7	23
ตารางที่ 2.4 ขาสัญญาณของ LCD Module	29
ตารางที่ 2.5 ชุดคำสั่งควบคุมและการแสดงข้อความ	31
ตารางที่ 2.6 ลักษณะทางเทคนิคของ SONY	44
ตารางที่ 2.7 ลักษณะทางเทคนิคของ NEC	46

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

ปริญญานิพนธ์นี้มีชื่อว่า "เครื่องรับส่งสัญญาณด้วยรีโมทคอนโทรล" เนื่องจากปัจจุบันมีการใช้อินฟราเรดรีโมทคอนโทรลอย่างแพร่หลาย เพราะสามารถควบคุมได้จากระยะไกล ทำให้เกิดความสะดวกแก่ผู้ใช้งานอย่างมาก หากต้องการออกแบบอุปกรณ์ที่ควบคุมด้วยอินฟราเรดรีโมทคอนโทรลที่ใช้กับเครื่องใช้ไฟฟ้าทั่วไป จำเป็นต้องรู้รหัสที่ใช้ในการควบคุม เพื่อสามารถออกแบบโปรแกรมที่ใช้ควบคุมส่วนต่างๆได้ เครื่องรับส่งสัญญาณด้วยรีโมทคอนโทรลนี้ จะสามารถทราบรหัสที่ส่งมาจากรีโมทคอนโทรลได้ โดยเครื่องจะแสดงรหัสเป็นเวลาของพัลซ์แต่ละลูกอยู่ในหน่วยไมโครวินาที โดยโครงงานนี้มีส่วนรับสัญญาณอินฟราเรด ซึ่งใช้ IRM8601 เป็นตัวรับ และด้านส่งสัญญาณอินฟราเรด จะใช้ไอซี 555 เป็นตัวสร้างความถี่พาหะแล้วมอดูเลตกับพัลซ์ที่สร้างขึ้นจากตัวไมโครคอนโทรลเลอร์ เพื่อใช้ส่งค่าที่อ่านได้นั้นออกมาทดสอบควบคุมเครื่องใช้ไฟฟ้าต่าง ๆ ควบคู่กับรีโมทคอนโทรลที่ใช้ควบคุมเครื่องใช้ไฟฟ้านั้นอยู่ โดยเครื่องสามารถอ่านสัญญาณอินฟราเรดจากรีโมทคอนโทรล และแสดงผลที่จอแอลซีดี (LCD) ซึ่งควบคุมผ่านไมโครคอนโทรลเลอร์ในตระกูล MCS-51 ของบริษัท ATMEL โดยใช้เบอร์ 89C52 มีหน่วยความจำ 8 Kbyte แบบ Flash Memory สามารถโปรแกรมและลบได้ 1000 ครั้ง

1.1 ขอบเขตโครงงาน

1. สามารถโปรแกรมไมโครคอนโทรลเลอร์ควบคุมการทำงานของโครงงาน
2. สามารถอ่านสัญญาณอินฟราเรดรีโมทคอนโทรลได้
3. สามารถควบคุมส่วนแสดงผลให้แสดงผลตามค่าที่อ่านได้
4. สามารถควบคุมการส่งสัญญาณรีโมทคอนโทรลด้วยคีย์แพด

1.2 เนื้อหาของแต่ละบท

บทที่ 1 เป็นการกล่าวถึงขอบเขตของโครงงาน และเนื้อหาในแต่ละบท

บทที่ 2 เป็นทฤษฎีไมโครคอนโทรลเลอร์เกี่ยวกับโครงสร้างทั่วไปของไมโครคอนโทรลเลอร์เบอร์ที่ใช้ในโครงงานนี้ ทฤษฎีการสแกนคีย์เบื้องต้น การใช้งาน DOT MAMTRIX LCD MODULE ซึ่งใช้เป็นส่วนแสดงผลโดยควบคุมการทำงานด้วยไมโครคอนโทรลเลอร์ และทฤษฎีอินฟราเรดรีโมทคอนโทรล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3 กล่าวถึงการออกแบบวงจรทางด้านฮาร์ดแวร์ (HARDWARE) และทางด้านซอฟแวร์ (SOFTWARE) โดยด้านฮาร์ดแวร์อธิบายถึงการออกแบบวงจรในส่วนต่างๆ และหน้าที่ของแต่ละส่วนนั้นๆ ส่วนด้านซอฟแวร์อธิบายการออกแบบโปรแกรมหลัก และโปรแกรมย่อยที่ใช้ควบคุมฮาร์ดแวร์ในส่วนต่างๆ

บทที่ 4 การทดลองและผลลัพธ์ที่ได้ กล่าวถึงผลการทดลอง การใช้งานของเครื่องรับส่งสัญญาณด้วยรีโมทคอนโทรล

บทที่ 5 สรุปและวิจารณ์ผลการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีทั่วไป

2.1 ไมโครคอนโทรลเลอร์ MCS-51

2.1.1โครงสร้างของไมโครคอนโทรลเลอร์ MCS-51 แบบเฟลช

ไมโครคอนโทรลเลอร์ตระกูล MCS-51ที่ใช้จะพูดถึงไมโครคอนโทรเลอร์ตระกูล MCS-51 ซึ่งมีหน่วยความจำแบบแฟลช (flash memory) ของ Atmel corporation มีเบอร์ขึ้นต้นด้วย AT89 เหตุผลที่ใช้ไมโครคอนโทรลเลอร์แบบนี้ในการเรียนรู้เพื่อใช้งานไมโครคอนโทรลเลอร์ตระกูล MCS-51 มีด้วยกันหลายประการดังนี้

1.หน่วยความจำโปรแกรมภายในตัวไมโครคอนโทรลเลอร์เป็นแบบแฟลช ทำให้สามารถลบและเขียนใหม่ได้นับพันครั้งจึงสามารถใช้งานในรูปแบบของไมโครคอนโทรลเลอร์ชิปเดี่ยวจึงไม่มีหน่วยความจำภายนอกส่งผลให้สามารถใช้งานพอร์ตอินพุตเอาต์พุตของไมโครคอนโทรลเลอร์ได้อย่างเต็มประสิทธิภาพ

2.ต้นทุนและเวลาในการพัฒนาระบบไมโครคอนโทรลเลอร์ลดลงอย่างมากเนื่องจากไม่ต้องใช้เครื่องมือพัฒนาจำพวกอีมูเลเตอร์และเครื่องโปรแกรมอีพรอม

3.บริษัทผู้ผลิตได้ทำการผลิตไมโครลคอนโทรลเลอร์ตระกูลนี้ออกมาหลายเบอร์ และมีความสามารถแตกต่างกันไป ทำให้มีทางเลือกในการใช้งานสูง

4.ด้วยการใช้หน่วยความจำภายในตัวไมโครคอนโทรลเลอร์ทำให้สามารถทำการป้องกันการคัดลอกข้อมูลของหน่วยความจำโปรแกรมได้อย่างดี

5.ในบางเบอร์ของไมโครคอนโทรลเลอร์ที่ผลิตโดย Atmel สามารถทำการโปรแกรมข้อมูลในหน่วยความจำโปรแกรมได้โดยที่ไม่ต้องถอดตัวไมโครลคอนโทรลเลอร์ออกมาทำการโปรแกรมใหม่หรือเรียกว่า การโปรแกรมในวงจรหรือ ในระบบ (In-system programming) ทำให้การพัฒนาหรือการซ่อมบำรุงตลอดจนการปรับปรุงหรืออัปเกรดข้อมูลในหน่วยความจำโปรแกรมทำได้อย่างสะดวก ภายใต้งบประมาณที่ไม่สูงมากนัก

6.ชุดคำสั่งและสถาปัตยกรรมพื้นฐานเหมือนกับไมโครคอนโทรลเลอร์ MCS-51 ของผู้ผลิตอื่น ไม่ว่าจะเป็นอินเทล ซีเมนต์ หรือ ดัสลัส

คุณสมบัติของไมโครคอนโทรลเลอร์ ตระกูล MCS-51 อนุกรม AT89xx

- เป็นไมโครคอนโทรลเลอร์ที่ใช้ซีพียูขนาด 8 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ภายในหน่วยความจำโปรแกรมเป็นแบบแฟลชสามารถลบและเขียนใหม่ได้พันครั้ง
- หน่วยความจำพื้นฐานเป็นหน่วยความจำแบบแรม บางเบอร์มีหน่วยความจำแบบอีพรอมเพิ่มเติม
- ขาพอร์ตเป็นแบบสองทิศทาง สามารถใช้งานได้ทั้งอินพุตและเอาต์พุต
- ไทเมอร์/เคาน์เตอร์ขนาด 16 บิตอย่างน้อย 2 ตัว
- สามารถรองรับแหล่งกำเนิดอินเตอร์รัปต์ได้ 6 ประเภท
- สามารถขยายหน่วยความจำภายนอกเพิ่มเติมได้สูงสุด 64 กิโลไบต์
- มีวงจรกำเนิดสัญญาณนาฬิกาอยู่ภายในชิป

ในรูปที่ 2.1 เป็นโครงสร้างพื้นฐานของไมโครคอนโทรลเลอร์ MCS-51 ในอนุกรม AT89Cxx จะเห็นได้ว่าโครงสร้างของ AT89Cxx จะเหมือนกับไมโครคอนโทรลเลอร์ตระกูล MCS-51 พื้นฐาน หากแตกต่างกันเฉพาะหน่วยความจำโปรแกรมแบบแฟรชที่เพิ่มเติมเข้ามา หากเป็นไมโครคอนโทรลเลอร์ในอนุกรม 87xx หน่วยความจำโปรแกรมภายในจะเป็นแบบอีพรอมและบางเบอร์โปรแกรมได้เพียงครั้งเดียว

สำหรับในรูปที่ 2.2 เป็นโครงสร้างพื้นฐานของอนุกรม AT89Sxx จะเห็นได้ว่ามีส่วนประกอบเพิ่มเติมแตกต่างจาก AT89Cxx อยู่หลายส่วน อาทิ วงจรเชื่อมต่ออนุกรมแบบ SPI ซึ่งในไมโครลคอนโทรลร์อนุกรมนี้ใช้ในการเขียนข้อมูลลงในหน่วยความจำโปรแกรมโดยไม่ต้องถอดตัวชิปออกไปจากระบบหรือเรียกว่าการโปรแกรมภานในวงจร ไทเมอร์/เคาน์เตอร์ขนาด 16 บิตที่เพิ่มเติมเข้ามาอีกหนึ่งตัวเป็นไทเมอร์ 2 และวงจรวอตช์ด็อกใช้ในการตรวจสอบการทำงานผิดพลาดของซีพียู

ในตารางที่ 2.1 แสดงรายละเอียดบางส่วนของไมโครคอนโทรลเลอร์ตระกูล MCS-51 แต่ละเบอร์ที่ Atmel ผลิตขึ้น และมีใช้งานอยู่ในปัจจุบัน

2.1.2 การจัดขาของไมโครคอนโทรลเลอร์ MCS-51

ไมโครคอนโทรลเลอร์ตระกูล MCS-51 ทุกเบอร์จะมีสถาปัตยกรรมและขาใช้งานพื้นฐานเหมือนกัน ดังแสดงในรูปที่ 2.3 และ 2.4 โดยมีรายละเอียดขั้นต้น ดังนี้

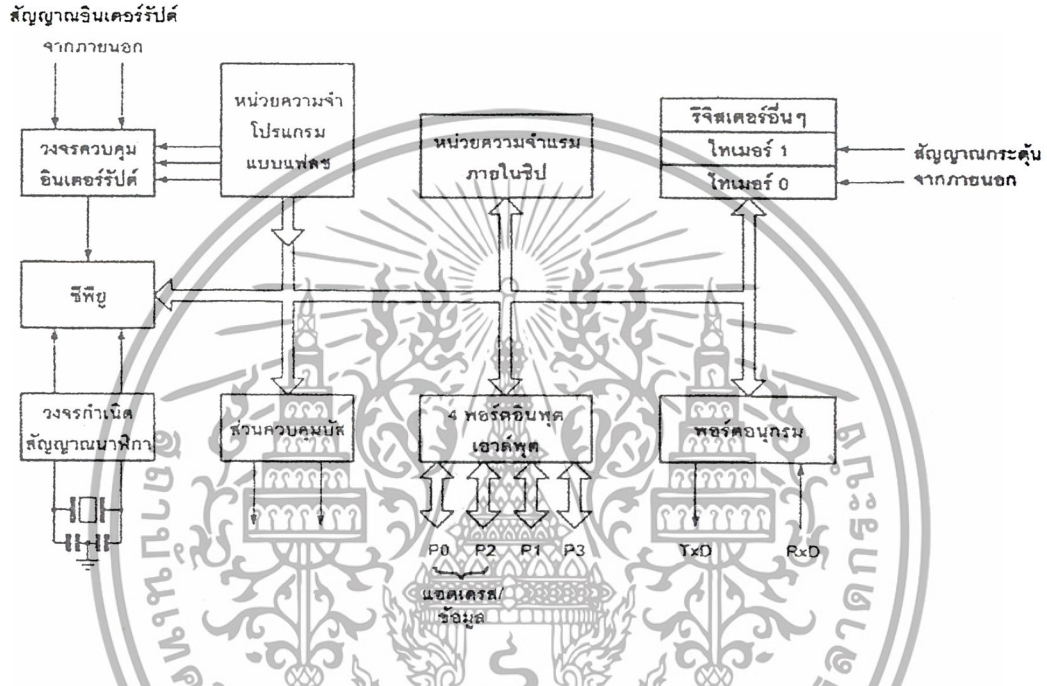
ขา Vcc ใช้ต่อไฟเลี้ยง +5V

ขา GND เป็นขากราวด์ สำหรับต่อกับกราวด์ของระบบ

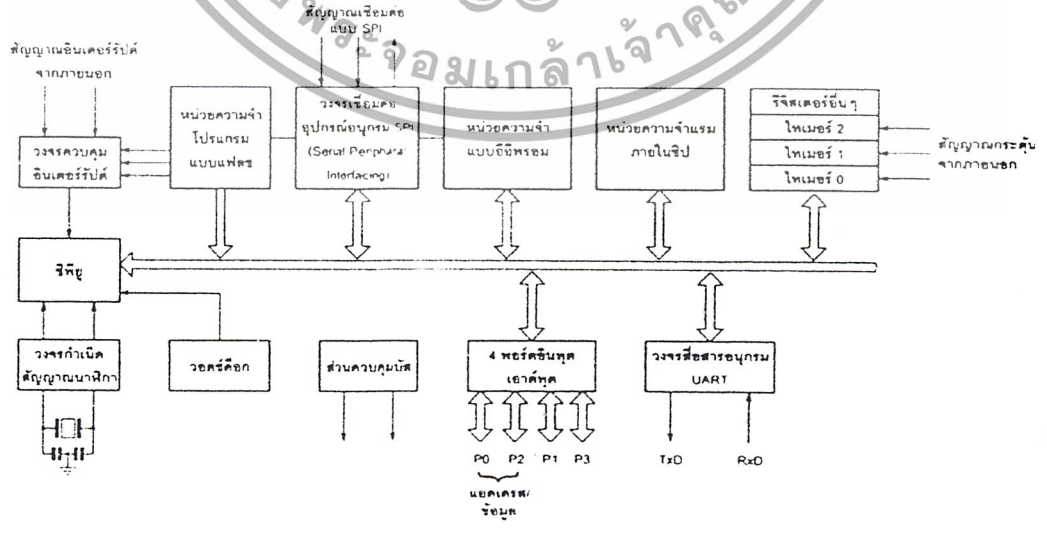
ขาพอร์ต 0 (P0.0-P0.7) มี 8 ขา แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุตและเอาต์พุต สำหรับใช้งานทั่วไป ถ้าหากต้องการกำหนดให้ขาพอร์ต 0 ขาใดขาหนึ่งเป็นอินพุตสามารถทำการได้โดยการเขียนข้อมูล “1” ไปยังแต่ละบิตของพอร์ตที่ต้องการติดต่อด้วย ส่งผลทำให้ขาพอร์ตนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีสถานะปล่อยลอย(float) จึงมีอินพุตอิมพีแดนซ์สูงสามารถใช้งานเป็นขาพอร์ตอินพุตได้ นอกจากขาพอร์ตนี้ยังถูกใช้งานในการติดต่อกับขาแอดเดรสไบต์ต่ำของหน่วยความจำติดต่อภายนอก (A0-A7) และขาข้อมูล (D0-D7) โดยใช้กระบวนการมัลติเพล็กเข้าช่วย เพื่อสลับการทำงานเป็นได้ทั้งขาติดต่อแอดเดรสและขาข้อมูล



รูปที่ 2.1 โครงสร้างพื้นฐานของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชในอนุกรม AT89Cxx



รูปที่ 2.2 โครงสร้างพื้นฐานของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชในอนุกรม AT89Sxx

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เบอร์ของไมโครคอนโทรลเลอร์	หน่วยความจำโปรแกรม	หน่วยความจำข้อมูล	จำนวนไทเมอร์/เคาน์เตอร์ 16 บิต
AT89C1051	แบบแฟลช ขนาด 1 กิโลไบต์	แรม 64 ไบต์	1
AT89C2051	แบบแฟลช ขนาด 2 กิโลไบต์	แรม 128 ไบต์	2
AT89C51	แบบแฟลช ขนาด 4 กิโลไบต์	แรม 128 ไบต์	2
AT89C52	แบบแฟลช ขนาด 8 กิโลไบต์	แรม 256 ไบต์	3
AT89C55	แบบแฟลช ขนาด 20 กิโลไบต์	แรม 256 ไบต์	3
AT89S8252	แบบแฟลช ขนาด 8 กิโลไบต์	แรม 256 ไบต์ อีอีพรอม 2 กิโลไบต์	3
AT89S53	แบบแฟลช ขนาด 12 กิโลไบต์	แรม 256 ไบต์	3

ตารางที่ 2.1 รายละเอียดโดยสรุปบางส่วนของไมโครคอนโทรลเลอร์MCS-51แบบแฟลช Atmel

ขาพอร์ต 1 (P1.0-P1.7) มี 8 ขา แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุตและเอาต์พุต สำหรับใช้งานทั่วไป ถ้าหากต้องการกำหนดให้ขาพอร์ต 0 ขาใดขาหนึ่งเป็นอินพุตสามารถทำการได้โดยการเขียนข้อมูล "1" ไปยังแต่ละบิตของพอร์ตที่ต้องการติดต่อด้วย นอกจากนั้นในอนุกรม AT89Sxx จะใช้ขา P1.0 เป็นขาอินพุตสำหรับนับค่าของไทเมอร์ 2 และ พอร์ต P1.1 เป็นขาอินพุททริกเกอร์ของไทเมอร์ 2 ในขณะที่ขา P1.4-P1.7 เป็นขาเชื่อมต่อแบบ SPI เพื่อทำการโปรแกรมข้อมูลในระบบ

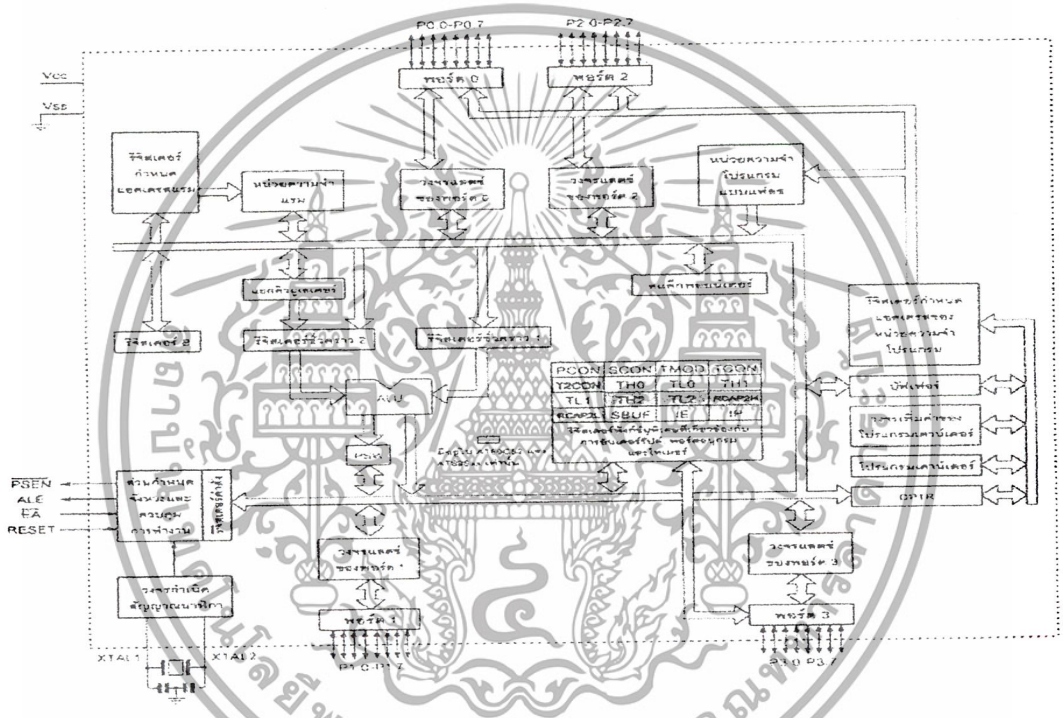
ขาพอร์ต 2 (P2.0-P2.7) มี 8 ขา แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุตและเอาต์พุต สำหรับใช้งานทั่วไป ถ้าหากต้องการกำหนดให้ขาพอร์ตใดเป็นอินพุตสามารถทำการได้โดยการเขียนข้อมูล "1" ไปยังแต่ละบิตของพอร์ตที่ต้องการติดต่อด้วย ส่งผลทำให้ขาพอร์ตนั้นมีสถานะปล่อยลอย(float) จึงมีอินพุตอิมพีแดนซ์สูงสามารถใช้งานเป็นขาพอร์ตอินพุตได้ นอกจากขาพอร์ตนี้ยังถูกใช้งานในการติดต่อกับขาแอดเดรสไบต์สูงของหน่วยความจำภายนอก (A8-A15)

ขาพอร์ต 3 (P3.0-P3.7) มี 8 ขา แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุตและเอาต์พุต สำหรับใช้งานทั่วไป ถ้าหากต้องการกำหนดให้ขาพอร์ตใดเป็นอินพุตสามารถทำการได้โดยการเขียนข้อมูล "1" ไปยังแต่ละบิตของพอร์ตที่ต้องการติดต่อด้วย ส่งผลทำให้ขาพอร์ตนั้นมีสถานะปล่อยลอย (float) จึงมีอินพุตอิมพีแดนซ์สูงสามารถใช้งานเป็นขาพอร์ตอินพุตได้ นอกจากนั้นขาพอร์ต 3 ยังเป็นขาที่มีหน้าที่ใช้งานพิเศษ ดังมีรายละเอียดดังต่อไปนี้

- P3.0 ใช้เป็นขาอินพุตสำหรับข้อมูลจากการสื่อสารแบบอนุกรม หรือขา RxD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- P3.1 ใช้เป็นขาอินพุตสำหรับข้อมูลจากการสื่อสารแบบอนุกรม หรือขา TxD
- P3.2 ใช้เป็นขาอินพุตรับสัญญาณอินเตอร์รัปต์จากภายนอกช่อง 0 หรือขา INT0
- P3.3 ใช้เป็นขาอินพุตรับสัญญาณอินเตอร์รัปต์จากภายนอกช่อง 1 หรือขา INT1
- P3.4 ใช้เป็นขาอินพุตรับสัญญาณไทเมอร์จากภายนอกช่อง 0 หรือขา T0
- P3.5 ใช้เป็นขาอินพุตรับสัญญาณอินเตอร์รัปต์จากภายนอกช่อง 1 หรือขา T1
- P3.6 ใช้เป็นขาสัญญาณ WR ในกรณีใช้กับหน่วยความจำภายนอก
- P3.7 ใช้เป็นขาสัญญาณ RD ในกรณีใช้กับหน่วยความจำภายนอก

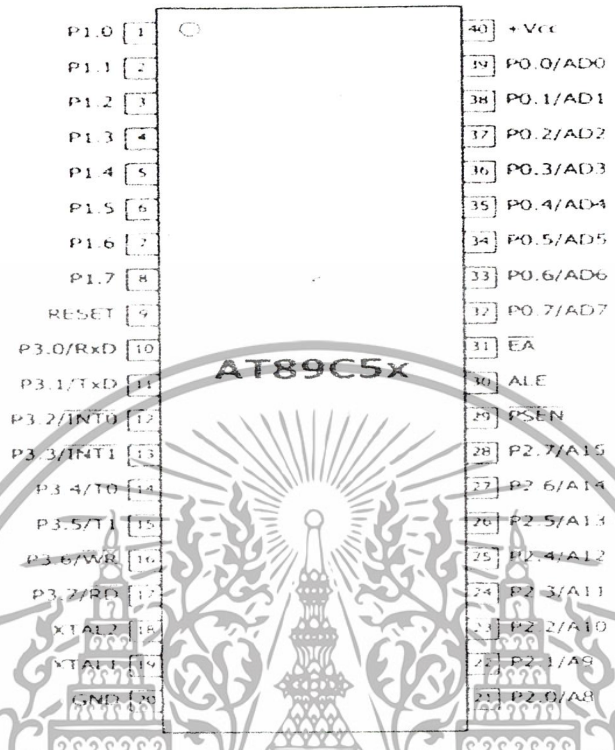


รูปที่ 2.3 รายละเอียดโครงสร้างหลักของไมโครคอนโทรลเลอร์ MCS-51แบบแฟลชของ Atmel

ขารีเซต (Reset) ใช้ในการรีเซตการทำงานของไมโครคอนโทรลเลอร์ โดยในการป้อนสัญญาณเพื่อรีเซตสถานะที่ขานี้ต้องอยู่ในระดับรีเซตอย่างน้อย 2 แมชีนไซเกิล โดยที่วงจรกำเนิดสัญญาณนาฬิกายังคงทำงานต่อเนื่องไปอย่างเป็นปกติ

ขา ALE /PROG (Address Latch Enable/Program pulse input) เป็นขาที่ใช้ในการควบคุมการแลตซ์ของขาพอร์ต 0 เมื่อมีการใช้งานหน่วยความจำภายนอก นอกจากนี้ขานี้ยังใช้เป็นขาสำหรับรับพัลซ์ของการโปรแกรมสำหรับโปรแกรมข้อมูลลงในไมโครคอนโทรลเลอร์ MCS-51 ในรุ่นที่มีหน่วยความจำโปรกรมเป็นแบบอีพรอม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 การจัดขามาตรฐานของไมโครคอนโทรลเลอร์ MCS-51 ในอนุกรม AT89C5x

ขา PSEN (Program Store Enable) ขานี้ใช้ในการส่งสัญญาณเพื่อขอร้องกับหน่วยความจำโปรแกรมภายนอก เมื่อไมโครคอนโทรลเลอร์ต้องการอ่านข้อมูลจากหน่วยความจำโปรแกรมที่อยู่ภายนอกตัวไมโครคอนโทรลเลอร์ต้องการอ่านข้อมูลจากหน่วยความจำโปแกรมภายนอก เนื่องจากตัวไมโครคอนโทรลเลอร์จะส่งสัญญาณออกมาที่ขานี้ 2 ครั้งในแต่ละแมชีนไซเกิล แต่ถ้าหากติดต่อกับหน่วยความจำข้อมูลภายนอก ขานี้จะไม่มีการส่งสัญญาณใดๆออกมา

ขา EA/Vpp (External Access enable/Programming voltage input) ใช้สำหรับเลือกการติดต่อหน่วยความจำโปรแกรมจากภายนอกหรือภายในตัวไมโครคอนโทรลเลอร์ ถ้าหากขานี้เป็น “0” เป็นการลเอกให้ไมโครคอนโทรลเลอร์ติดต่อกับหน่วยความจำโปรแกรมภายนอกแต่ถ้าหากขานี้เป็น “1” เป็นการเลือกให้ตัวไมโครคอนโทรลเลอร์ติดต่อกับหน่วยความจำโปรแกรมภายในของตัวไมโครคอนโทรลเลอร์ นอกจากนี้แล้วที่ขานี้ยังใช้เป็นขาอินพุตสำหรับรับแรงดันไฟสูงสำหรับโปรแกรมหน่วยความจำภายในไมโครคอนโทรลเลอร์ สำหรับไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชต้องการแรงดันสำหรับการโปรแกรม +12 V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขา XTAL1 และ XTAL2 เป็นสำหรับต่อคริสตอลเพื่อสร้างสัญญาณนาฬิกาในการกำหนดจังหวะการทำงานของไมโครคอนโทรลเลอร์

2.1.3 โครงสร้างและทำงานของพอร์ต

ไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชมีพอร์ตให้ใช้งานทั้งสิ้น 4 พอร์ตคือ พอร์ต 0 ถึงพอร์ต 3 แต่ละพอร์ตมีขนาด 8 บิต เป็นพอร์ตแบบ 2 ทิศทาง กล่าวคือ สามารถเป็นได้ทั้งอินพุตสำหรับรับข้อมูลเข้า และเอาท์พุตสำหรับส่งข้อมูลออกทุกพอร์ตไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเบอร์ใด ดังสรุปได้ในตารางที่ 2.2

ในรูปที่ 2.5 แสดงวงจรภายในของแต่ละพอร์ตของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช โดยในรูปที่ 2.5 (ก)เป็นวงจรของพอร์ต 0 วงจรแลตช์ของแต่ละบิตในแต่ละพอร์ตก็คือวงจรดีฟลิฟลอปนั่นเอง การอ่านค่าสถานะของพอร์ตและสถานะของวงจรแลตช์สามารถกระทำได้อย่างอิสระด้วยสัญญาณที่แยกจากกัน นั่นคือสัญญาณอ่านข้อมูลจากขาพอร์ต และสัญญาณอ่านข้อมูลจากวงจรแลตช์ ส่วนการเขียนข้อมูลมายังพอร์ตต้องส่งสัญญาณมายังขา CLK ของดีฟลิฟลอปในขณะที่ข้อมูลจะผ่านมาทางขาบัสข้อมูลภายในเข้าสู่ขาDของดีฟลิฟลอปที่พอร์ตนี้มีวงจรมัลติเพล็กซ์สำหรับกำหนดลักษณะการทำงานของพอร์ตว่า ต้องการใช้งานเป็นขาพอร์ตอินพุตเอาท์พุตปกติหรือใช้การติดต่อกับหน่วยความจำภายนอกไมโครคอนโทรลเลอร์

เนื่องจากที่ขาพอร์ต 0 ไม่มีวงจรพูลอัปภายใน หากมีการนำพอร์ต 0 ไปใช้งานเป็นอินพุตจะต้องต่อตัวต้านทานพูลอัปภายนอกเข้าที่ขาพอร์ต 0 ทุกขาด้วย

ในรูปที่ 2.5 (ข) เป็นวงจรของพอร์ต 1 ซึ่งมีลักษณะโดยทั่วไปคล้ายกับพอร์ต 0 หากแต่ไม่มีวงจร มัลติเพล็กซ์ เนื่องจากพอร์ตนี้จะ ไม่ใช้ในการติดต่อกับหน่วยความจำภายนอก แต่จะมีวงจรพูลอัปภายในที่แต่ละบิตของพอร์ตนี้แทน สำหรับรายละเอียดของวงจรพูลอัป แสดงในรูปที่ 2.6

ในรูปที่ 2.5(ค) เป็นวงจรภายในของพอร์ต 2 จะคล้ายกับพอร์ต 0 มาก ต่างเพียงมีวงจรพูลอัปเพิ่มเข้ามาส่วนในรูปที่ 2.5 (ง) เป็นวงจรภายในของพอร์ต 3 จะเห็นว่าคล้ายกับพอร์ต 1 มีการเพิ่มเติมวงจรบัฟเฟอร์ และวงจรอินพุตเอาท์พุตเมื่อทำงานในฟังก์ชั่นพิเศษเข้ามา เนื่องจากพอร์ต 3 สามารถนำไปใช้งานในหน้าที่พิเศษได้ทุกขา

(1) การใช้งานเป็นพอร์ตอินพุต

เนื่องจากพอร์ตของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชสามารถเป็นได้ทั้งอินพุตและเอาต์พุต ดังนั้นจึงมีความจำเป็นอย่างยิ่งต้องทำความเข้าใจถึงการกำหนดลักษณะการทำงานให้แก่พอร์ตของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการกำหนดให้เป็นพอร์ตอินพุต ต้องเริ่มต้นด้วยการเขียนข้อมูล “1” มาที่แต่ละบิตของพอร์ตที่ต้องการใช้งานเป็นอินพุต เพื่อหยุดการทำงานของเฟตที่ใช้ในการขับสัญญาณเอาต์พุตของบิตนั้นๆ ทำให้ขาสัญญาณของพอร์ตเชื่อมต่อเข้ากับวงจรพูลอัปภายในโดยตรงส่งผลให้ขาพอร์ตนั้นมีลอจิกเป็น “1” สามารถรับสัญญาณลอจิก “0” จากอุปกรณ์ภายนอกได้ง่าย สัญญาณข้อมูลจากอุปกรณ์ภายนอกจะถูกส่งเข้ามาแล้วเก็บไว้ในวงจรบัฟเฟอร์ภายในพอร์ต แล้วรอให้ซีพียูมาอ่านเข้าไป เมื่อเป็นเช่นนี้อุปกรณ์ภายนอกที่เชื่อมต่อกับอินพุตพอร์ตของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชควรกำหนดให้ทำงานในสภาวะลอจิก “0” จะดีและสะดวกที่สุด (ซึ่งในปัจจุบันอุปกรณ์อินพุตที่เชื่อมต่อไมโครคอนโทรลเลอร์แทบทั้งหมดทำงานที่ลอจิก “0” แล้ว)

ขา	เบอร์ของไมโครคอนโทรลเลอร์	หน้าที่พิเศษ
P1.0	AT89C52/AT89Sxx	ขา T2 เป็นขาอินพุตนับค่าของไทเมอร์/เคาน์เตอร์ 2 และเป็นขา
P1.1	AT89C52/AT89Sxx	และควบคุมทิศทางของสัญญาณ
P1.4	AT89Sxx	ขา SS (Slave Select) เป็นขาเลือกการติดต่อในกรณีที่ไมโครคอนโทรลเลอร์เป็นอุปกรณ์สเลฟ ในระบบการติดต่อแบบ SPI
P1.5	AT89Sxx	ขา MOSI (Master data output, Slave data input) ใช้ในการติดต่อกับพอร์ต SPI
P1.6	AT89Sxx	ขา MISO (Master data input, Slave data output) ใช้ในการติดต่อกับพอร์ต SPI
P1.7	AT89Sxx	ขา SCK (Master clock output) เป็นขาสัญญาณนาฬิกาของการติดต่อกับพอร์ต SPI

ตารางที่ 2.2 หน้าที่พิเศษของพอร์ต 1 ในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชของ Atmel การใช้งานเป็นพอร์ตเอาต์พุต

โดยปกติแล้วขาพอร์ตจะกำหนดให้มีลักษณะเป็นเอาต์พุตอยู่แล้ว ดังนั้นจึงสามารถส่งข้อมูลออกไปได้อย่างง่ายดายและตรงไปตรงมา กล่าวคือเมื่อต้องการส่งข้อมูล “0” ออกไปทางเอาต์พุตก็ให้เขียนข้อมูล “0” ไปยังวงจรแลตช์ ซึ่งก็จะส่งต่อไปขับเฟต ทำให้เฟตทำงาน ที่ขาพอร์ตที่กำหนดให้ทำงานก็จะเกิดลอจิก “0” ขึ้น ในทางตรงข้ามหากต้องการส่งข้อมูล “1” ออกไปก็ให้เขียนข้อมูล “1” ไปยังวงจรแลตช์วงจรก็จะหยุดทำงาน ทำให้ที่ขาพอร์ตเชื่อมต่อกับวงจรพูลอัปภายในเกิด

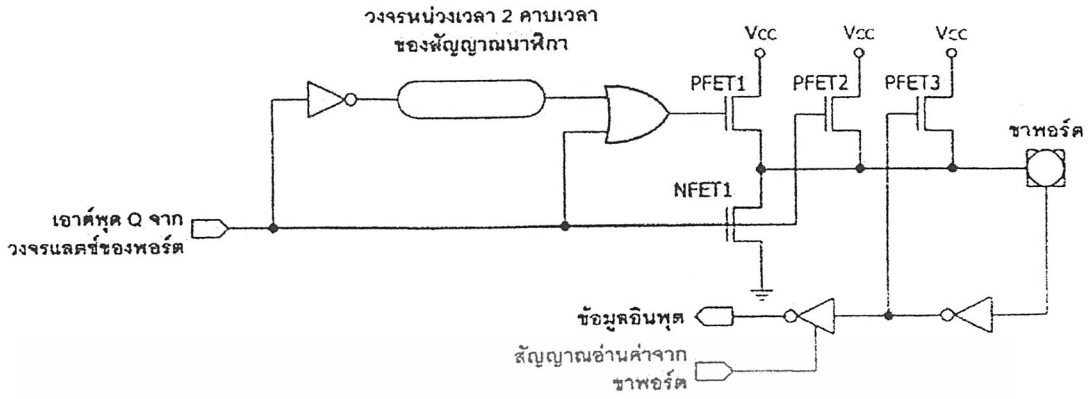
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นลอจิก “1” ที่ขาพอร์ตนั้น ซึ่งจะคล้ายกับการกำหนดให้เป็นขาอินพุตมาก เพียงแต่แตกต่างกันที่กระบวนการในการเคลื่อนย้ายข้อมูล โดยถ้าเป็นอินพุตจะมีสัญญาณมาอ่านข้อมูลที่บัฟเฟอร์ แต่ถ้าเป็นเอาต์พุตจะไม่มีการอ่านข้อมูลที่บัฟเฟอร์แต่อย่างใด เว้นแต่กรณีที่ต้องการตรวจสอบข้อมูลที่ส่งออกมาทางเอาต์พุต

เมื่อใช้งานเป็นพอร์ตเอาต์พุต แต่ละขา (หรือละบิต) ของแต่ละพอร์ตมีความสามารถในการจ่ายกระแสหรือที่เรียกว่า กระแสซอร์ส (source current) ได้สูงสุด 10 mA และทุกขารวมกันในแต่ละพอร์ต (ทั้ง 8 บิต) สูงสุด 26 mA สำหรับพอร์ต 0 และ 15 mA สำหรับพอร์ต 1-3 ในกรณีที่ใช้งานทุกพอร์ตเอาต์พุตจะสามารถจ่ายกระแสได้รวมกันสูงสุด 71 mA ดังนั้นในการใช้งานเป็นพอร์ตเอาต์พุตเพื่อไม่ให้เกิดปัญหาเกี่ยวกับความสามารถในการจ่ายกระแสควรต่อวงจรบัฟเฟอร์นี้ทางเอาต์พุตเพื่อช่วยในการขับกระแสอีกทางหนึ่ง

รูปที่ 2.5 วงจรภายในของพอร์ตทุกพอร์ตในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



วงจรพูลอัปประกอบด้วยเฟตชนิดพีแชนเนล 3 ตัวคือ PFET1-PFET3 โดย NFET1 จะทำงานเมื่อได้รับลอจิก "1" จากขา Q และหยุดทำงานเมื่อได้รับลอจิก "0" วงจรพูลอัปจะเริ่มต้นทำงานเมื่อ NFET1 ได้รับลอจิก "1" PFET1 จะทำงานนานประมาณ 2 คาบเวลาของสัญญาณนาฬิกาภายใน หลังจากที่เกิดการเปลี่ยนแปลงจากลอจิก "0" เป็นลอจิก "1" ในขณะที่ PFET1 ทำงาน จะทำให้ PFET3 ทำงานตามไปด้วย ทำให้เกิดการพูลอัปขาพอร์ต

รูปที่ 2.6 วงจรพูลอัปภายในพอร์ตไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

(2) การอ่านค่าลอจิกจากพอร์ต

ในไมโครคอนโทรลเลอร์ MCS-51แบบแฟลชสามารถอ่านค่าลอจิกจากพอร์ตได้ 2 ลักษณะ คือ อ่านจากขาพอร์ตโดยตรง และอ่านจากวงจรแลตช์ของแต่ละพอร์ต

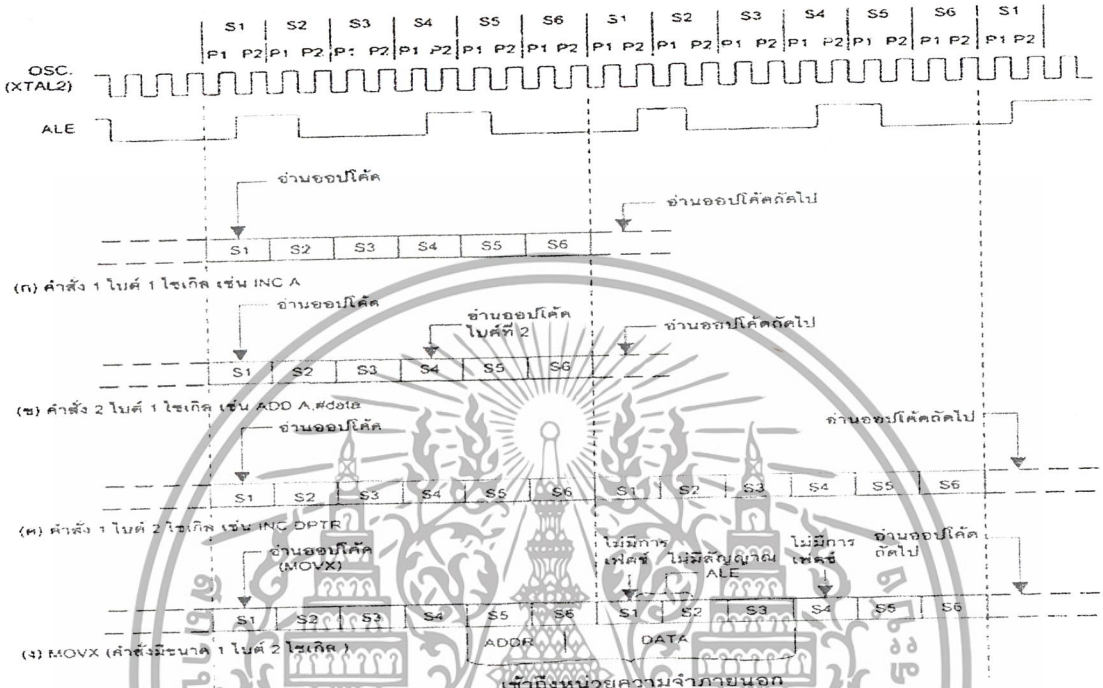
ในกรณีที่พอร์ตต่อกับขาเบสของทรานซิสเตอร์ชนิดที่เป็น NPN และขาอิมิตเตอร์ของทรานซิสเตอร์ตัวนั้นต่อลงกราวด์หากมีการส่งข้อมูล "1" ไปยังทรานซิสเตอร์ทำให้ทรานซิสเตอร์ทำงานสถานะลอจิกที่ขาพอร์ตจะเป็น "0" เนื่องจากเมื่อทรานซิสเตอร์ทำงาน จะเสมือนว่าขาพอร์ตนั้นถูกต่อลงกราวด์ ทำให้หากอ่านค่าลอจิกที่ขาพอร์ตจะได้ผลตรงข้ามกับที่ส่งออกมา แต่หากทำงานอ่านค่าลอจิกที่วงจรแลตช์ จะได้ค่าที่ตรงกับค่าที่ต้องการส่งจริงดังนั้น ในการอ่านค่าลอจิกจากพอร์ตจึงต้องเลือกวิธีการให้เหมาะสมกับอุปกรณ์ที่นำมาต่อด้วย

2.1.4 การทำงานของไมโครคอนโทรลเลอร์ MCS-51

ในการใช้งานไมโครคอนโทรลเลอร์ MCS-51 จะต้องทำความเข้าใจถึงจังหวะการทำงานของซีพียูและลำดับขั้นตอนการประมวณผลคำสั่ง ในการประมวณผลคำสั่งของซีพียูจะมีขั้นตอนหลักๆ 2 ขั้นตอนคือ กระบวนการเฟตช์ (fetch) เป็นการเรียกคำออกจากหน่วยความจำโปรแกรมแล้วทำการแปลงรหัสคำสั่งนั้นเป็นภาษาเครื่องเพื่อเตรียมการประมวณผลขั้นตอนต่อมาคือ กระบวนการเอ็กซิคิวต์ (execute) เป็นการกระทำตามคำสั่งที่กำหนดหรือตามที่เฟตช์ขึ้นมาโดย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กระบวนการก่อนหน้านี้ เมื่อทำการเอ็กซิคิวต์คำสั่งเรียบร้อยแล้วก็จะไปเริ่มกระบวนการเฟตช์คำสั่งใหม่ต่อไป



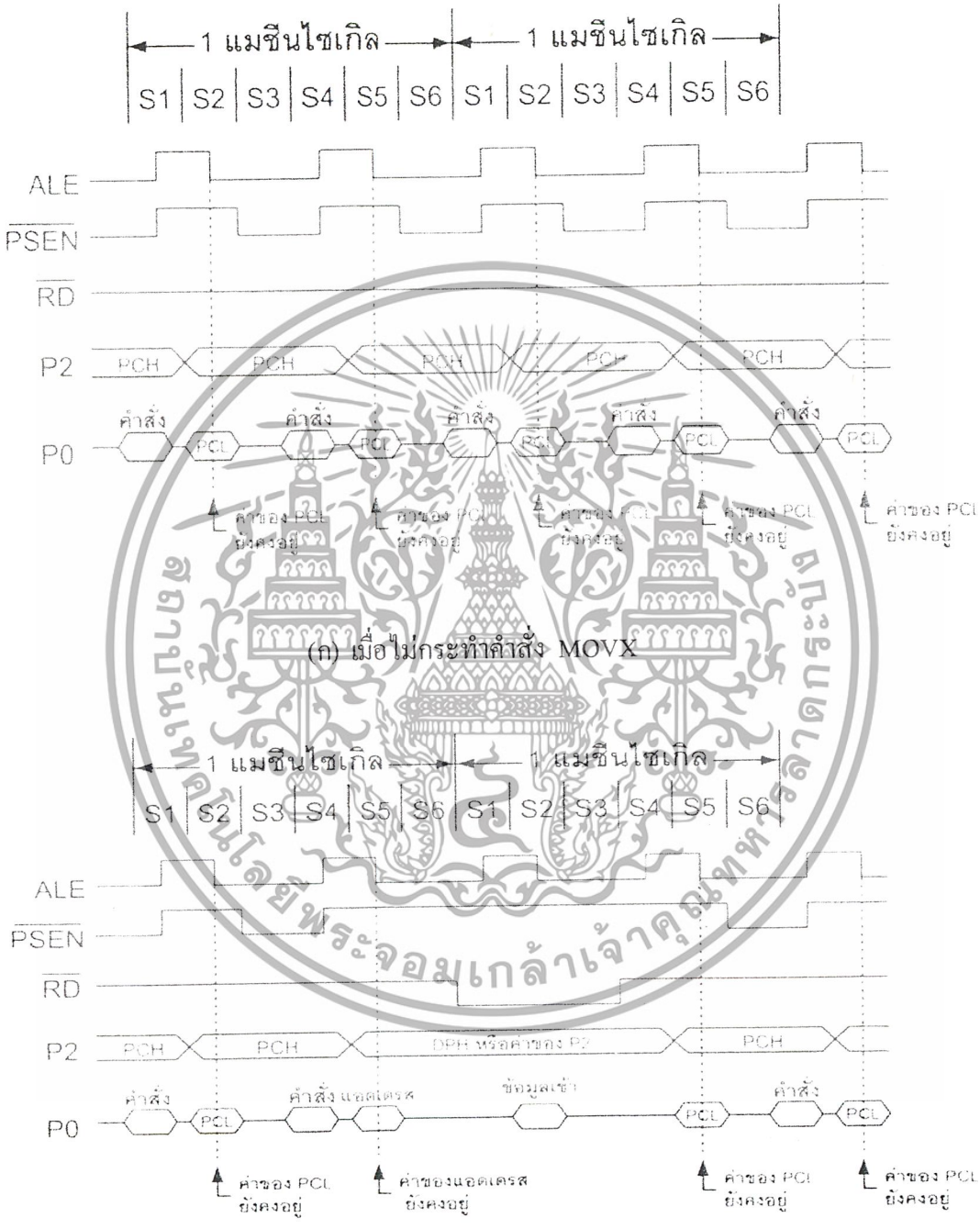
รูปที่ 2.7 ไซเกิลการทำงานของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

เมื่อเริ่มจ่ายไฟให้แก่ไมโครคอนโทรลเลอร์จะเกิดการรีเซตในลักษณะที่เรียกว่าเพาเวอร์ออนรีเซต (power on reset) ซีพียูเริ่มต้นการทำงานที่แอดเดรส 0000H ของหน่วยความจำโปรแกรม จังหวะการทำงานของซีพียูจะเป็นไปตามรูปแบบ โดยได้รับการกำหนดมาจากรอบการทำงานหรือแมชีนไซเกิล ในรูปที่ 2.7 เป็นไดอะแกรมเวลาแสดงจังหวะการทำงานของ ไมโครคอนโทรลเลอร์ MCS-51 โดยในหนึ่งรอบการทำงานหรือแมชีนไซเกิลจะแบ่งย่อยออกเป็น 6 สเตต (state) กำหนดชื่อเป็น S1-S6 ในแต่ละสเตตมีค่าเวลาเท่ากับ 2 คาบเวลาของสัญญาณนาฬิกา ถ้าสัญญาณนาฬิกามีความถี่ 12 MHz จะมีคาบเวลาเท่ากับ 1 ms คาบเวลาทั้งสองภายในหนึ่งสเตตจะเรียกว่าเฟส 1 (phase 1) และเฟส 2 (phase)

ในรูปที่ 2.7(ก) และ (ข) จะเป็นการเอ็กซิคิวต์คำสั่งที่ใช้เวลา 1 ไซเกิลเริ่มต้นที่สเตต 1 จะเป็นการอ่านค่าออปโค้ด อันเป็นกระบวนการแลตช์ค่าของออปโค้ดส่งไปให้รีจิสเตอร์คำสั่ง (Instruction Register:IR) การเฟตช์ครั้งที่ 2 จะเกิดขึ้นที่สเตต 4 ภายในแมชีนไซเกิลเดียวกันในกรณี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่เป็นคำสั่งไบต์เดียว การเฟตช์ครั้งที่ 2 ภายในแมชีนไซเกิลเดียวกันจะถูกตัดทิ้งไป ในคำสั่งที่มีใช้เวลา 1 ไซเกิล จะสิ้นสุดการทำงานลงในสเตต 6 ของแมชีนไซเกิลเดียวกัน



(ก) เมื่อไม่กระทำคำสั่ง MOVX

(ข) เมื่อกระทำคำสั่ง MOVX

รูปที่ 2.8 ไดอะแกรมเวลาแสดงการติดต่อและเข้าถึงหน่วยความจำโปรแกรมภายนอกของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในกรณีที่คำสั่งใช้เวลา 2 ไซเกิล การทำงานของคำสั่งนั้นจะสิ้นสุดลงในสเตต 6 ของแมชีนไซเกิลที่ 2 ดังในไดอะแกรมรูปที่ 2.7 (ค) สำหรับในการกระทำคำสั่ง MOVX ซึ่งเป็นคำสั่งขนาด 1 ไบต์ 2 ไซเกิล จะไม่มีการเฟตซ์เกิดขึ้นในไซเกิลที่สองของคำสั่ง MOVX นี้ เนื่องจากซีพียูจะไปทำการติดต่อกับหน่วยความจำภายนอกดังแสดงในไดอะแกรมรูปที่ 2.7 (ง) และเห็นได้ว่าเวลาในการเอ็กซิคิวต์จะไม่ได้ขึ้นอยู่กับว่าทำการติดต่อหน่วยความจำโปรแกรมภายในหรือภายนอก

ในรูปที่ 2.8 สัญญาณและไดอะแกรมเวลาของการเข้าถึงหน่วยความจำโปรแกรมภายนอก โดยในรูปที่ 2.8 (ก) เป็นไดอะแกรมเวลาในขณะที่ยังไม่มีการกระทำคำสั่ง MOVX สัญญาณที่ขา ALE และPSEN เกิดการแอกตีฟ 2 ครั้งภายในหนึ่งแมชีนไซเกิลในทุกครั้งที่ ALE เกิดการแอกตีฟที่พอร์ต 0 (P0) จะมีค่าของรีจิสเตอร์ PC ในไบต์ต่ำออกมา ในขณะที่พอร์ต 2 (P2) ก็มีค่าของ PC ในไบต์สูงเพื่อชี้ไปยังแอดเดรสต่อไปที่ต้องไปดำเนินการ สำหรับขา PSEN ก็จะเกิดการแอกตีฟเมื่อมีการติดต่อกับหน่วยความจำโปรแกรมภายนอก ในกรณีที่กระทำคำสั่ง MOVX เพื่อเข้าถึงหน่วยความจำข้อมูลภายนอก ที่ขา PSEN จะไม่เกิดการแอกตีฟ 2 ครั้งภายในหนึ่งแมชีนไซเกิลเนื่องจากบัสแอดเดรสและบัสข้อมูลจะใช้ในการติดต่อหน่วยความจำข้อมูลภายนอกแทน แต่สำหรับสัญญาณ ALE ยังคงแอกตีฟตามจังหวะการทำงานเหมือนเดิม

จากไดอะแกรมเวลาสามารถสรุปได้ว่า ในการทำงาน 1 รอบหรือ 1 แมชีนไซเกิล ซีพียูในไมโครคอนโทรลเลอร์ MCS-51 จะใช้เวลา 12 คาบเวลาของสัญญาณนาฬิกา นั่นคือ เวลาในการทำงาน 1 ไซเกิลมีค่าเท่ากับ 1ms หรือมีความเร็วในการทำงานภายใน 1MHz ในกรณีที่ใช้ความถี่สัญญาณนาฬิกา 12MHz ดังนั้น ถ้าต้องการทราบความเร็วของการทำงานภายในของตัวไมโครคอนโทรลเลอร์ MCS-51 สามารถหาได้จาก ค่าความถี่สัญญาณนาฬิกาหารด้วย 12 และถ้าต้องการหาค่าเวลาของ 1 รอบการทำงานหรือ 1 แมชีนไซเกิล สามารถทำได้โดยการหาส่วนกลับของความเร็ว ในการทำงานภายในของไมโครคอนโทรลเลอร์ MCS-51 สามารถสรุปเป็นสูตรทางคณิตศาสตร์ได้ดังนี้

ความเร็วในการทำงานภายในของไมโครคอนโทรลเลอร์เท่ากับ
ความถี่ของสัญญาณนาฬิกา (ค่าของคริสตอลที่ต่ออยู่ที่ขา XTAL1 และ XTAL2)/12
เวลา 1 แมชีนไซเกิล = 1/ความเร็วในการทำงานภายในนของไมโครคอนโทรลเลอร์

2.1.5 การจัดหน่วยความจำของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

ในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชมีหน่วยความจำภายในหลักๆอยู่ 2 ส่วน คือหน่วยความจำโปรแกรมและหน่วยความจำข้อมูล ซึ่งก็มีขนาดและการจัดสรรแตกต่างกันไปแต่ละเบอร์ในบทนี้จะกล่าวถึงรายละเอียดของการจัดสรรหน่วยความจำภายในไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

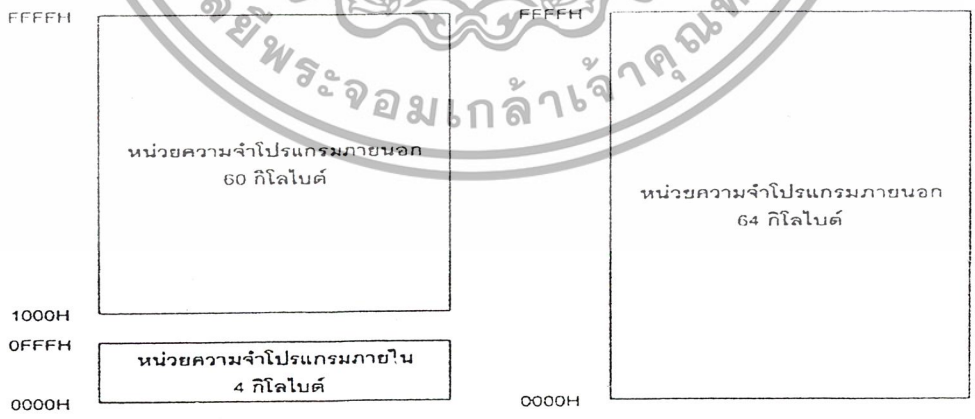
MCS-51แบบแฟลช การเชื่อมต่อกับหน่วยความจำภายนอกและข้อมูลเบื้องต้นของรีจีสเตอร์ฟังก์ชั่นพิเศษที่ใช้ควบคุมการทำงานของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

(1) หน่วยความจำโปรแกรม (Program memory)

ในรูปที่ 2.9 แสดงการจัดหน่วยความจำโปรแกรมของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชในเบอร์ต่างๆที่นิยมใช้ อันประกอบด้วยเบอร์ AT89C51 และ AT89C52 จะเห็นได้ว่าทั้งสองเบอร์สามารถติดต่อหน่วยความจำโปรแกรมได้สูงสุด 64 กิโลไบต์ โดยสามารถเลือกใช้หน่วยความจำโปรแกรมภายในอย่างเดียวหรือร่วมกับภายนอกหรือเลือกใช้หน่วยความจำภายนอกอย่างเดียวก็ได้ ดังในรูปที่ 2.9 (ก) โดยภายใน AT89C51จะมีหน่วยความจำโปรแกรมภายใน 4 กิโลไบต์ ในขณะที่ AT89C52 จะมีขนาด 8 กิโลไบต์

ในกรณีที่ใช้หน่วยความจำภายในภายนอกร่วมกัน หากใช้ AT89C51 ก็จะสามารถติดต่อกับหน่วยความจำภายนอกได้ 60 กิโลไบต์ และถ้าใช้เบอร์ AT89C52 จะสามารถติดต่อกับหน่วยความจำโปรแกรมภายนอกได้ 56 กิโลไบต์

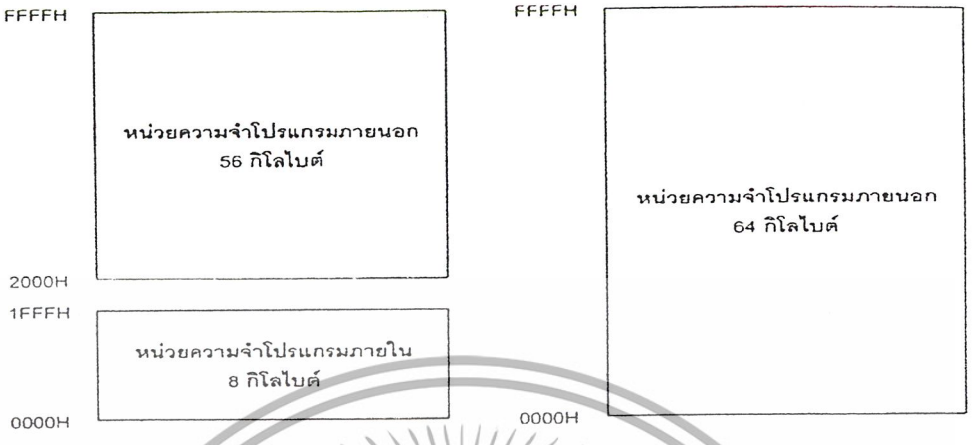
หน่วยความจำที่ใช้เก็บข้อมูลของโปรแกรมควบคุมการทำงานของไมโครคอนโทรลเลอร์ หรือที่เรียกว่า โปรแกรมมอนิเตอร์ (monitor program) หากใช้หน่วยความจำภายนอกมักจะบรรจุอยู่ในหน่วยความจำชนิดอีพรอม (EPROM : Erasable Programmable Read-only Memory) ซึ่งสามารถทำการอ่านได้เพียงอย่างเดียว



(ก) การจัดสรรหน่วยความจำโปรแกรมของไมโครคอนโทรลเลอร์เบอร์ AT89C51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



(ข) การจัดสรรหน่วยความจำโปรแกรมของไมโครคอนโทรลเลอร์เบอร์ AT89C52

รูปที่ 2.9 การจัดสรรหน่วยความจำโปรแกรมของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

หน่วยความจำโปรแกรมมีแอดเดรสเริ่มต้นที่ 000H เมื่อซีพียูได้รับการรีเซตให้เริ่มต้นการทำงานจะต้องมาเริ่มต้นที่แอดเดรส 0000H นี้เสมอ อย่างไรก็ตามในพื้นที่หน่วยความจำโปรแกรมไม่ว่าจะใช้งานจากภายในหรือภายนอกก็ตาม ต้องมีการสงวนพื้นที่บางตำแหน่งเอาไว้สำหรับการบริการอินเตอร์รัปต์ 6 ประเภท ประเภทละ 8 ไบต์ ประกอบด้วย

- พื้นที่สำหรับการบริการอินเตอร์รัปต์ 0 จากภายนอก กำหนดไว้ที่แอดเดรส 0003H
- พื้นที่สำหรับการบริการอินเตอร์รัปต์จากไทเมอร์ 0 กำหนดไว้ที่แอดเดรส 000BH
- พื้นที่สำหรับการบริการอินเตอร์รัปต์ 1 จากภายนอก กำหนดไว้ที่แอดเดรส 0013H
- พื้นที่สำหรับการบริการอินเตอร์รัปต์จากไทเมอร์ 1 กำหนดไว้ที่แอดเดรส 001BH
- พื้นที่สำหรับการบริการอินเตอร์รัปต์ของการสื่อสารอนุกรม กำหนดไว้ที่แอดเดรส 0023H
- พื้นที่สำหรับการบริการอินเตอร์รัปต์จากไทเมอร์ 2 กำหนดไว้ที่แอดเดรส 002BH

กรณีที่ใช้ไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชที่มีหน่วยความจำโปรแกรมภายในแต่ต้องการติดต่อกับหน่วยความจำโปรแกรมภายนอกด้วย สามารถทำได้โดยต้องกำหนดแอดเดรสของหน่วยความจำโปรแกรมให้ต่อจากแอดเดรสสุดท้ายของหน่วยความจำโปรแกรมภายในของตัวไมโครคอนโทรลเลอร์ ยกตัวอย่างไมโครคอนโทรลเลอร์ AT89C51 มีหน่วยความจำโปรแกรมภายในขนาด 4 กิโลไบต์ มีแอดเดรสอยู่ระหว่าง 0000H-0FFFH เมื่อต่อหน่วยความจำโปรแกรมภายนอกต้องกำหนดให้แอดเดรสอยู่ในช่วง 1000H-FFFFH

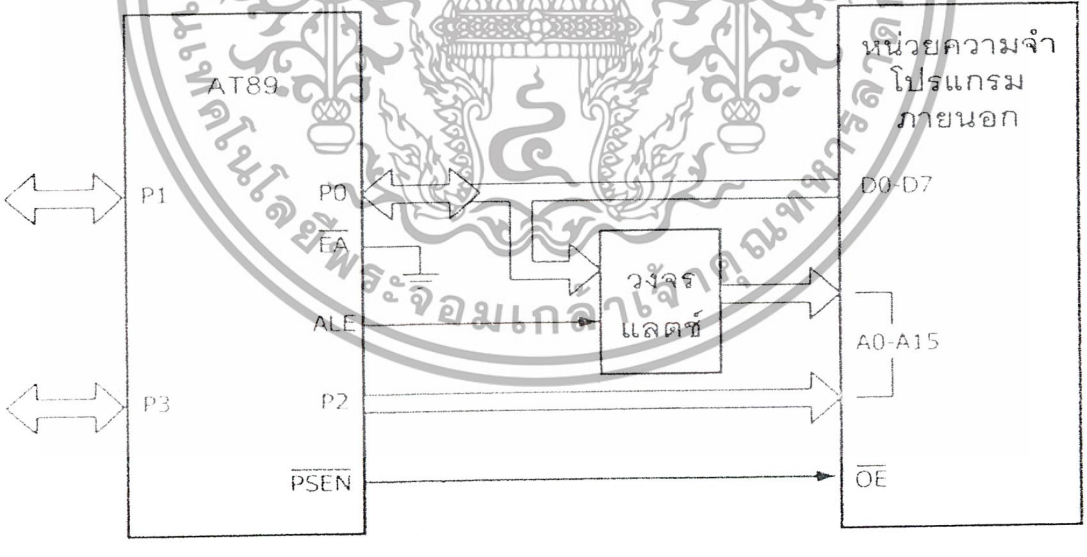
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

46433

การต่อหน่วยความจำโปรแกรมภายนอกแสดงดังในรูปที่ 2.10 ขาพอร์ต P0.0-P0.7 ใช้เป็นขาข้อมูล D0-D7 และขาแอดเดรสไบต์ต่ำ โดยผ่านวงจรแลตช์ ซึ่งปกติใช้ไอซีเบอร์ 74HC573 และใช้สัญญาณ ALE และ PSEN ในการใช้งานขา P0.0-P0.7 เพื่อเป็นขาข้อมูลหรือขาแอดเดรส ในขณะที่ขา P2.0-P2.7 ใช้ในการเชื่อมต่อกับขาแอดเดรสไบต์สูง A8-A15 ดั้งนั้นเมื่อมีการติดต่อกับหน่วยความจำโปรแกรมภายนอกไมโครคอนโทรลเลอร์จะเหลือขาพอร์ตเพียง 16 บิต คือที่ขาพอร์ต P1.0-P1.7 และ P3.0-P3.7

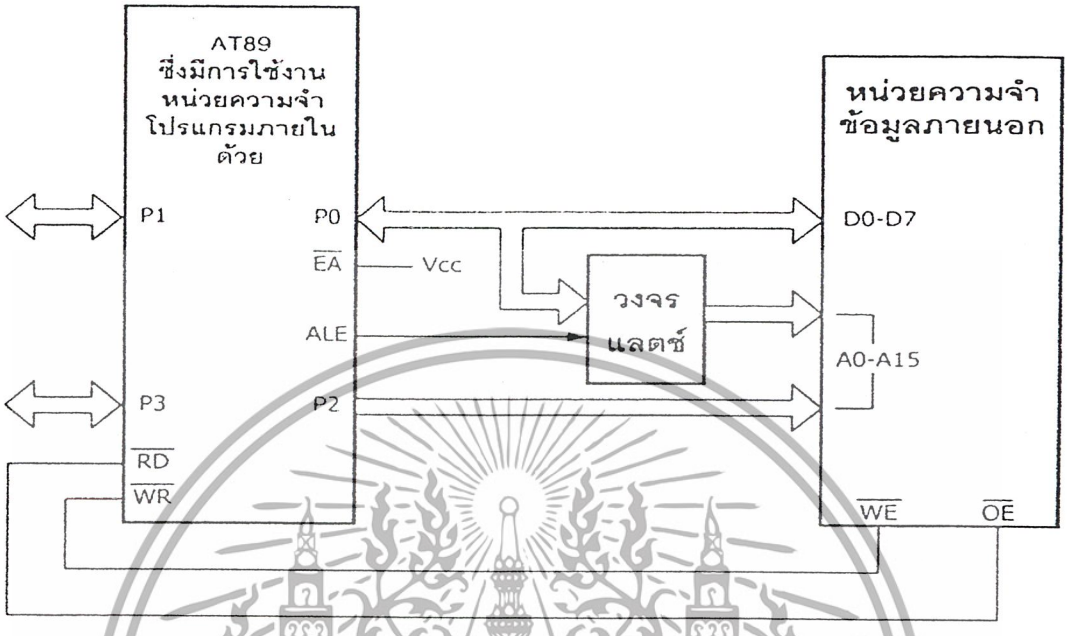
(2) หน่วยความจำข้อมูล (Data memory)

มีด้วยกัน 2 แบบคือ หน่วยความจำข้อมูลภายนอกและภายใน โดยไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชในอนุกรม AT89 สามารถติดต่อกับหน่วยความจำข้อมูลภายนอกได้สูงสุด 64 กิโลไบต์โดยการใช้คำสั่ง MOVX ในการติดต่อกับหน่วยความจำข้อมูลภายนอก การติดต่อกับหน่วยความจำข้อมูลภายนอกของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชแสดงดังรูปที่ 2.11 จะเห็นได้ว่า มีลักษณะคล้ายกับการติดต่อกับหน่วยความจำโปรแกรมภายนอก แตกต่างกันที่มีสัญญาณที่ใช้สำหรับการอ่านและเขียนหน่วยความจำข้อมูลภายนอก นั่นคือ ขา RD และ WR



รูปที่ 2.10 การเชื่อมต่อหน่วยความจำโปรแกรมภายนอกของไมโครคอนโทรลเลอร์ MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.11 การเชื่อมต่อหน่วยความจำข้อมูลภายนอกของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

สำหรับไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชในอนุกรม AT89 ทุกเบอร์จะมีหน่วยความจำข้อมูลภายในเป็นแบบแรม(RAM : Ramdom Access Memory) โดยแต่ละเบอร์จะมีขนาดแตกต่างกันไป ในเบอร์AT89C51 มีหน่วยความจำข้อมูลภายในขนาด 128 ไบต์ ในขณะที่เบอร์ AT89C52 มีขนาด 256 ไบต์ สำหรับการจัดสรรหน่วยความจำข้อมูลภายในแบ่งเป็น 3 ส่วน คือ หน่วยความจำข้อมูลส่วนล่าง (lower), ส่วนบน(upper) และรีจิสเตอร์ฟังก์ชั่นพิเศษ (SFR : Special Function Register) แต่ละส่วนมีขนาด 128ไบต์

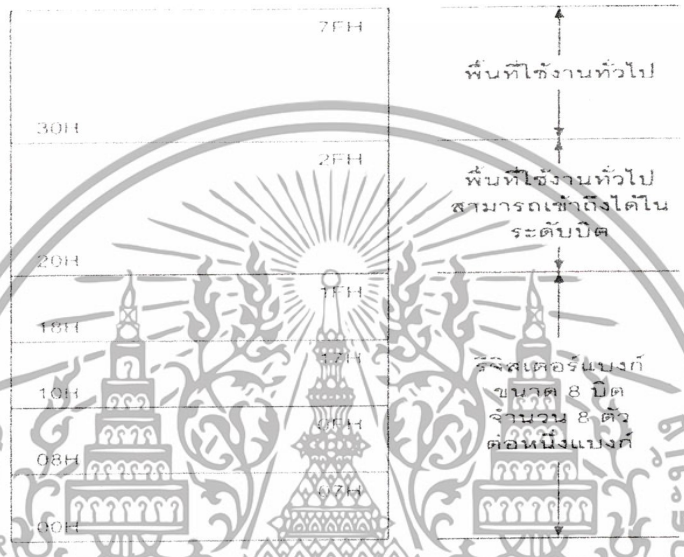
หน่วยความจำส่วนบนและรีจิสเตอร์ฟังก์ชั่นพิเศษมีตำแหน่งทับซ้อนกันแต่จะใช้การติดต่อที่แตกต่างกัน และไมโครคอนโทรลเลอร์ MCS-51 บางเบอร์จะไม่มีหน่วยความจำข้อมูลส่วนบน

ขนาดของหน่วยความจำข้อมูลของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชโดยแท้จริงแล้วมีเพียง 256 ไบต์ แต่ด้วยการจัดการที่เข้าถึงที่แตกต่างกัน จึงดูเสมือนว่าไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชมีหน่วยความจำข้อมูลภายในสูงถึง 384 ไบต์ โดยในหน่วยความจำข้อมูลส่วนล่างขนาด 128 ไบต์ มีแอดเดรสอยู่ที่ 00H-7FH สามารถเข้าถึงได้โดยตรงและโดยอ้อม สำหรับหน่วยความจำข้อมูลส่วนบนมีขนาด 128 ไบต์ เช่นกัน มีแอดเดรสอยู่ที่ 80H-FFH สามารถเข้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถึงแบบโดยอ้อมได้เท่านั้น ในขณะที่รีจิสเตอร์ SFR มีแอดเดรสอยู่ที่ 80H-FFH เช่นเดียวกับหน่วยความจำข้อมูลส่วนบน แต่สำหรับรีจิสเตอร์ SFR ใช้ในการเข้าถึงแบบโดยตรง

ดังนั้นเพื่อความสะดวกและง่ายตลอดจนป้องกันความสับสนในการเขียนโปรแกรมสำหรับผู้เริ่มต้นจึงควรใช้หน่วยความจำข้อมูลภายในเพียง 128 ไบต์จากหน่วยความจำข้อมูลส่วนล่างร่วมกับรีจิสเตอร์ SFR



รูปที่ 2.12 การจัดสรรพื้นที่ของหน่วยความจำข้อมูลภายในส่วนล่างของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

ในรูปที่ 2.12 แสดงการจัดสรรข้อมูลส่วนล่าง หน่วยความจำ 32 ไบต์ต่ำสุดที่แอดเดรส 00H-1FH แบ่งเป็น 4 กลุ่ม เรียกว่า 4 แบงก์ (bank) แต่ละแบงก์มีรีจิสเตอร์ 8 ตัวคือ R0-R7 การติดต่อกับหน่วยความจำในแบงก์ใดให้กำหนดที่รีจิสเตอร์ PSW (Program Status Word register)

หน่วยความจำข้อมูล 16 ไบต์ถัดมาที่แอดเดรส 20H-2FH เป็นพื้นที่สำหรับใช้งานทั่วไป สามารถเข้าถึงได้ในระดับบิต (Bit addressable) และหน่วยความจำข้อมูลที่เหลือ 80 ไบต์จะต้องแบ่งส่วนหนึ่งไว้สำรองพื้นที่ของสแต็ก (Stack : ที่พักชั่วคราวในกรณีที่ซีพียูมีการกระโดดไปทำงานในโปรแกรมย่อย) การเข้าถึงความจำในส่วนนี้ต้องใช้การเข้าถึงในระดับไบต์

ในรูปที่ 2.13 แสดงโครงสร้างของหน่วยความจำข้อมูลส่วนบน ซึ่งจะมีลักษณะที่คล้ายกับหน่วยความจำข้อมูลส่วนล่าง หากในแต่ 80 ไบต์บนไม่จำเป็นต้องสำรองไว้สำหรับสแต็ก และต้องใช้การเข้าถึงในลักษณะโดยอ้อมเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

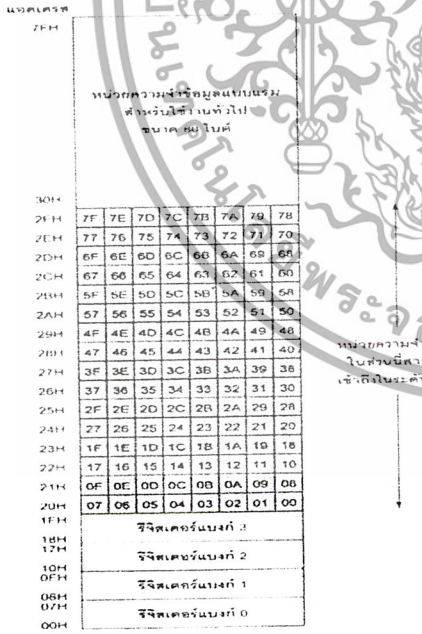
(3) รีจิสเตอร์ฟังก์ชั่นพิเศษ(SFR : Special Function Register)

เป็นรีจิสเตอร์ที่ใช้ควบคุมการทำงานของไมโครคอนโทรลเลอร์ MCS-51 มีด้วยกัน 22 ตัวสำหรับเบอร์ AT89C51 และ 28 ตัวในเบอร์ AT89C52 และอนุกรม AT89S5xx ทั้งนี้เนื่องจากใน AT89C52 และ AT89S5xx มีจำนวนไทเมอร์เคาร์เตอร์มากกว่า AT89C51

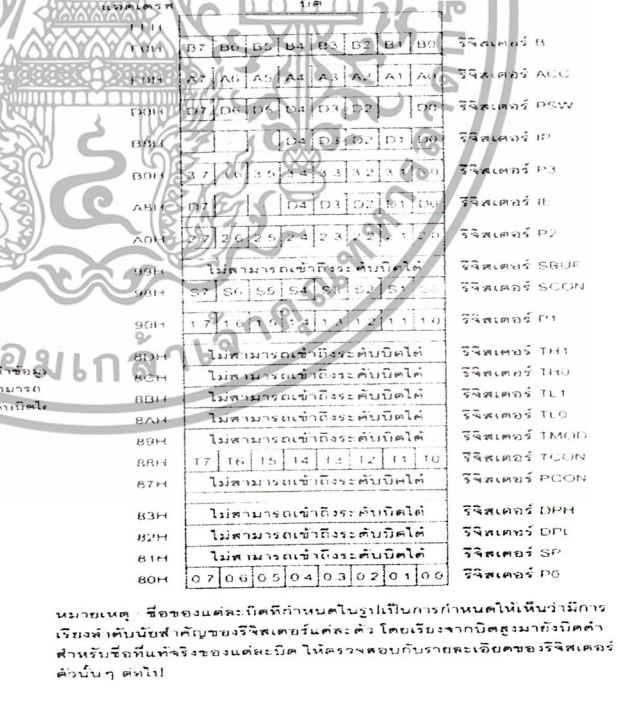
รีจิสเตอร์ SFR มีแอดเดรสอยู่ระหว่าง 80H-FFH ในพื้นที่ของหน่วยความจำข้อมูลส่วนบน สามารถเข้าถึงได้โดยตรง (direct addressing) ในรูปที่ 2.14 แสดงการจัดสรรพื้นที่ของรีจิสเตอร์ SFR แต่ละตัวในหน่วยความจำข้อมูลส่วนบน

(4) รีจิสเตอร์แสดงสถานะของโปรแกรม (Program Status Word : PSW)

เป็นรีจิสเตอร์ขนาด 8 บิต สามารถเข้าถึงได้ในระดับบิต จึงสามารถกำหนดค่าในแต่ละบิตของรีจิสเตอร์ตัวนี้ได้อย่างอิสระ มีแอดเดรสที่ D0H ทำหน้าที่เก็บสถานะของการทำงานของโปรแกรมในขณะนั้นจะเรียกสถานะต่างๆของโปรแกรมว่า แฟลก(flag) เมื่อซีพียูกระทำคำสั่งทางคณิตศาสตร์และลอจิกแล้วเกิดการเปลี่ยนสถานะขึ้น ผลของการเปลี่ยนแปลงนั้นจะมาปรากฏที่บิตต่างๆของรีจิสเตอร์ PSW รายละเอียดของแต่ละบิตในรีจิสเตอร์ PSW



รูปที่ 2.13 โครงสร้างของหน่วยความจำข้อมูล ภายในส่วนบนของไมโครคอนโทรลเลอร์ MCS-51



รูปที่ 2.14 การจัดสรรพื้นที่ของรีจิสเตอร์ฟังก์ชั่นพิเศษ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะเห็นได้ว่า นอกจากรีจิสเตอร์ PSW ถูกใช้ในการเก็บสถานะของโปรแกรมแล้วที่บิต RS0 และ RS1ยังใช้ในการเลือกแบงก์ของหน่วยความจำส่วนล่าง ซึ่งเป็นพื้นที่ของรีจิสเตอร์ R0-R7 ด้วย ดังมีรายละเอียดแสดงในตาราง 2.3 โดยปกติแล้วในการใช้งานรีจิสเตอร์ R0-R7 ด้วย มักนิยมใช้แบงก์ 0 เป็นลำดับแรก หากไม่เพียงพอจึงเลือกแบงก์อื่นๆมาใช้ แต่ต้องระมัดระวังในการกำหนดค่าและลำดับการติดต่อให้ดี มิเช่นนั้น อาจทำให้การเขียนโปรแกรมเกิดการสับสน ดังนั้นสำหรับผู้เริ่มต้นใช้งานไมโครคอนโทรลเลอร์ MCS-51จึงควรเลือกใช้รีจิสเตอร์ R0-R7 ในแบงก์ 0 เพียงเบอร์เดียวให้ชำนาญเสียก่อน

การกำหนดค่าของรีจิสเตอร์ PSW เพื่อเลือกใช้งานรีจิสเตอร์ R0-R7 ควรกำหนดไว้ตอนต้นของโปรแกรมเสมอ เพื่อจะได้เขียนโปรแกรมติดต่อกับรีจิสเตอร์ R0-R7 ได้อย่างสะดวกและไม่เกิดความผิดพลาด

(5) แอกคิวมูเลเตอร์(Accumulator : ACC)

มีขนาด 8 บิต มีแอดเดรสอยู่ที่ตำแหน่ง E0H เป็นรีจิสเตอร์ที่ใช้เก็บข้อมูลหรือผลลัพธ์ที่ได้จากการทำงานของไมโครคอนโทรลเลอร์ โดยเฉพาะอย่างยิ่งในการคำนวณทางคณิตศาสตร์และลอจิก ก่อนที่จะส่งข้อมูลหรือผลลัพธ์ที่ได้แก่ซีพียูเพื่อทำการประมวลผลต่อไป อาจเรียกสั้นๆว่า รีจิสเตอร์ A หรือ ACC รีจิสเตอร์นี้สามารถเข้าถึงในระดับบิตได้

รีจิสเตอร์ Bมีขนาด 8 บิต มีแอดเดรสอยู่ที่ตำแหน่ง F0H มีหน้าที่พิเศษคือ หากต้องการคูณหรือการหารทางคณิตศาสตร์ ต้องนำข้อมูลที่ต้องการหารหรือคูณมาเก็บไว้ในรีจิสเตอร์ B แล้วจึงกระทำคำสั่งการคูณหรือหารกับค่าในรีจิสเตอร์ A ต่อไป

ในกรณีที่ไม่ได้มีการคูณหรือการหารข้อมูล สามารถใช้รีจิสเตอร์ B นี้ในการเก็บข้อมูลทั่วไปได้เหมือนกับรีจิสเตอร์ปกติ และสามารถเข้าถึงในระดับบิตได้เช่นเดียวกับรีจิสเตอร์ A

(6) โปรแกรมเคาน์เตอร์ (Program Counter : PC)

มีขนาด 16 บิต มีหน้าที่แจ้งแอดเดรสของหน่วยความจำโปรแกรมในตำแหน่งถัดไปที่ซีพียูจะต้องไปทำงาน รีจิสเตอร์ PC เป็นรีจิสเตอร์ตัวเดียวที่ไม่ได้จัดสรรไว้รวมกับรีจิสเตอร์ SFR ตัวอื่นๆ การเปลี่ยนแปลงค่าของรีจิสเตอร์ PC จะขึ้นอยู่กับผลของการกระทำคำสั่งภายในหน่วยความจำโปรแกรมที่ผู้เขียนโปรแกมกำหนด

รีจิสเตอร์ PC มีความสำคัญมาก โดยเฉพาะอย่างยิ่งในการตรวจสอบการทำงานของโปรแกรมว่า ดำเนินไปตามลำดับขั้นตอนตามที่กำหนดไว้หรือไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บิต7	บิต6	บิต5	บิต4	บิต3	บิต2	บิต1	บิต0
CY	AC	F0	RS1	RS2	OV	-	P

CY : แฟลกทด (Carry flag) เป็น “1” เมื่อมีการกระทำคำสั่งทางคณิตศาสตร์และลอจิก แล้วค่าของแอกคิวมูเลเตอร์เกิน 255 (ฐานสิบ) หรือ FFH

AC : แฟลกทดเสริม (Auxiliary Carry flag) เป็น “1” เมื่อมีการกระทำคำสั่งทางคณิตศาสตร์แล้วทำให้เกิดหารทดข้ามจากบิต 3 มายังบิต 4

F0 : แฟลกใช้งานทั่วไป เมื่อผู้เขียนโปรแกรมกำหนดค่าที่บิตนี้แล้ว ไม่ว่าจะทำคำสั่งใดๆ ที่บิตนี้จะไม่มีการเปลี่ยนแปลง

RS1 : บิตเลือกรีจิสเตอร์แบงก์ (Register Select1) ใช้งานร่วมกับบิต RS0 เพื่อเลือกแบงก์ของรีจิสเตอร์ R0-R7

RS2 : บิตเลือกรีจิสเตอร์แบงก์ (Register Select0) ใช้งานร่วมกับบิต RS1 เพื่อเลือกแบงก์ของรีจิสเตอร์ R0-R7

OV : บิตเกิน (Oerflow) เป็น “1” เมื่อมีการกระทำคำสั่งทางคณิตศาสตร์และลอจิก แล้วทำให้เกิดการทดข้ามบิต 6 มายังบิต 7 ของแอกคิวมูเลเตอร์ หรือแอกคิวมูเลเตอร์มีค่าเกิน 127 (ฐานสิบ) นอกจากนั้นยังมีการใช้แสดงเป็นค่าลบด้วย

- : บิตนี้ผู้ใช้งานสามารถกำหนดได้อย่างอิสระ

P : บิตพาราตี้ (Parity) ใช้ในการตรวจสอบจำนวนค่า “1” ภายในแอกคิวมูเลเตอร์ ภายในแอกคิวมูเตอร์มีจำนวนบิตที่เป็น “1” รวมกันเป็นเลขคู่ บิตนี้จะเป็น “0” ถ้ารวมกันเป็นเลขคี่ บิตนี้จะเป็น “1”

RS1	RS2	แบงก์ของรีจิสเตอร์	ช่วงแอดเดรส
0	0	แบงก์ 0	00H-07H
0	1	แบงก์ 1	08H-0FH
1	0	แบงก์ 2	10H-17H
1	1	แบงก์ 3	18H-0FH

ตาราง 2.3 การเลือกแบงก์ของหน่วยความจำส่วนล่างเพื่อติดต่อกับรีจิสเตอร์แบงก์ R0-R7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(7) สแต็กพอยเตอร์(Stack pointer : SP)

รีจิสเตอร์ตัวชี้สแต็ก มีขนาด 8 บิต มีแอดเดรสอยู่ที่ตำแหน่ง 81H ใช้ในการเก็บค่าตำแหน่งของตัวชี้สแต็กซึ่งสามารถเปลี่ยนแปลงได้เมื่อซีพียูมีการกระโดดไปทำงานที่โปรแกรมย่อย หรือกระโดดโปรแกรมย่อยกลับมายังโปรแกรมหลัก เมื่อมีการรีเซตเกิดขึ้น (รีเซต : การกระทำที่ส่งผลให้ซีพียูต้องเริ่มการทำงานใหม่ตั้งแต่ต้น) ค่าของรีจิสเตอร์ SP จะเท่ากับ 07H ดังนั้นแอดเดรสของพื้นที่ที่สำรองไว้ทำหน้าที่เป็นสแต็กจะเท่ากับ 08H

(8) รีจิสเตอร์ชี้ข้อมูลหรือดาต้าพอยน์เตอร์ (Data Pointer : DPTR)

มีขนาด 16 บิต โดยแบ่งเป็นรีจิสเตอร์ชี้ข้อมูลไบต์สูง (DPH) และรีจิสเตอร์ชี้ข้อมูลไบต์ต่ำ (DPL) แต่ละตัวมีขนาด 8 บิต มีแอดเดรสอยู่ที่ 82H สำหรับ DPL และ 83H สำหรับ DPH รีจิสเตอร์ DPTR นี้ใช้การเก็บค่าแอดเดรสของหน่วยความจำภายนอกที่ไมโครคอนโทรลเลอร์ต้องการติดต่อด้วย

(9) รีจิสเตอร์พอร์ต (Port register)

เป็นรีจิสเตอร์ขนาด 8บิตที่ใช้เก็บข้อมูลของแต่ละพอร์ตของไมโครคอนโทรลเลอร์MCS-51 มี 4 ตัวคือรีจิสเตอร์พอร์ต 0 หรือ P0 มีแอดเดรสอยู่ที่ 80H ,รีจิสเตอร์พอร์ต 1 หรือ P1 มีแอดเดรสอยู่ที่ 90H รีจิสเตอร์พอร์ต 2 หรือ P2 มีแอดเดรสอยู่ที่ A0H,รีจิสเตอร์พอร์ต 3 หรือ P3 มีแอดเดรสอยู่ที่ B0H รีจิสเตอร์ทุกตัวสามารถเข้าถึงในระดับบิตได้ เมื่อต้องการอ่านหรือเขียนข้อมูลออกไปยังพอร์ตของไมโครคอนโทรลเลอร์ จะต้องทำการผ่านรีจิสเตอร์นี้ทุกครั้ง

(10) รีจิสเตอร์บัฟเฟอร์ข้อมูลอนุกรม (Serial Data Buffer : SBUF)

เป็นรีจิสเตอร์ขนาด 8 บิตมีแอดเดรสอยู่ที่ 99H ใช้ในการเก็บข้อมูลที่ส่งออกหรือรับเข้าของวงจรสื่อสารอนุกรมที่มีอยู่ในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช โดยภายในรีจิสเตอร์ SBUF นี้จะแบ่งออกเป็น 2 ส่วน คือรีจิสเตอร์บัฟเฟอร์สำหรับส่งข้อมูล (transmit buffer register) และรีจิสเตอร์บัฟเฟอร์สำหรับรับข้อมูล(recieve buffer register) เมื่อมีการเขียนข้อมูลมายังรีจิสเตอร์ SBUF ข้อมูลนั้นจะถูกส่งต่อไปยังบัฟเฟอร์สำหรับส่งข้อมูลเพื่อส่งออกจากไมโครคอนโทรลเลอร์ผ่านทางขา TxD หรือ P3.1 ในกรณีที่มีการอ่านข้อมูลจากรีจิสเตอร์ SBUF ข้อมูลจะถกส่งผ่านไปยังรีจิสเตอร์บัฟเฟอร์สำหรับรับข้อมูลเพื่อส่งไปยังไมโครคอนโทรลเลอร์ต่อไป สำหรับการรับข้อมูลอนุกรมจากภายนอกนั้นจะผ่านทางขา RxD หรือ P3.0 ของไมโครคอนโทรลเลอร์ MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(11) รีจิสเตอร์ไทเมอร์ (Timer register)

เป็นรีจิสเตอร์ขนาด 16 บิต แบ่งเป็นไบต์สูงและไบต์ต่ำเช่นเดียวกับรีจีสเตอร์ DPTR ซึ่งเป็นรีจิสเตอร์ไทเมอร์ใช้ในการเก็บค่าของตัวนับหรือเคาน์เตอร์(counter)ภายในไมโครคอนโทรลเลอร์ เพื่อใช้ในการสร้างเวลา,จับเวลา หรือนับจำนวนสัญญาณนาฬิกาภายใน บางทีเรียกรีจิสเตอร์ตัวนี้ว่า รีจิสเตอร์ไทเมอร์/เคาน์เตอร์

ในไมโครคอนโทรลเลอร์เบอร์ AT89C51 มีรีจิสเตอร์ไทเมอร์/เคาน์เตอร์ 2 ตัว แบ่งเป็น T0 หรือTimer 0 และ T1 หรือTimer 1 ในรีจิสเตอร์ยังแบ่งเป็นรีจิสเตอร์ไทเมอร์ไบต์ต่ำ(TL) และรีจิสเตอร์ไทเมอร์ไบต์สูง (TH)เหมือนกัน โดยรีจิสเตอร์ TL0 มีแอดเดรสอยู่ที่ 8AH รีจิสเตอร์ TH0 มีแอดเดรสอยู่ที่ 8BH ในขณะที่ TL1 และ TH1 มีแอดเดรสอยู่ที่ 8CH และ 8DH สำหรับในเบอร์ AT89C52 และในอนุกรม AT89Sxx จะมีรีจิสเตอร์ไทเมอร์/เคาน์เตอร์ถึง 3 ตัว โดยมีรีจิสเตอร์ TL2 และ TH2 มีแอดเดรสอยู่ที่ 0CCH และ 0CDH เพิ่มเติมเข้ามา

(12) รีจิสเตอร์แคปเจอร์ (Capture register)

รีจิสเตอร์ขนาด 16 บิต มีเฉพาะในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเบอร์ AT89C52 และในอนุกรม AT89Sxx เท่านั้น เนื่องจากใช้ร่วมกับไทเมอร์/เคาน์เตอร์ 2 (Timer 2) โดยรีจิสเตอร์แคปเจอร์นี้มีชื่อเรียกย่อว่า รีจิสเตอร์ RCAP2 ซึ่งแบ่งออกเป็นไบต์ต่ำคือ RCAP2L มีแอดเดรสอยู่ที่ 0CAH และไบต์สูง คือ RCAP2H มีแอดเดรสอยู่ที่ 0CBH

รีจิสเตอร์แคปเจอร์นี้จะถูกใช้งานเมื่อกำหนดให้ไทเมอร์ 2 ทำงานในโหมดแคปเจอร์ ซึ่งเป็นโหมดที่กำหนดให้ไมโครคอนโทรลเลอร์ทำการตรวจจับการเปลี่ยนแปลงสถานะลอจิกที่ขา T2EX ทั้งนี้เพื่อประโยชน์ในการวัดคาบเวลา ความถี่ และการเปลี่ยนแปลงสัญญาณพัลซ์ที่ขา T2EX

(13) รีจิสเตอร์ควบคุม (Control register)

รีจิสเตอร์ PCON เป็นรีจิสเตอร์ที่เกี่ยวข้องกับการกำหนดอัตราการรับส่งข้อมูลของวงจรสื่อสารอนุกรมและกำหนดการทำงานในโหมดประหยัดพลังงานของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

รีจิสเตอร์ SCON เป็นรีจิสเตอร์ที่ใช้ในการควบคุมการทำงานของวงจรสื่อสารอนุกรมภายในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช

รีจิสเตอร์ TCON และ T2CON เป็นรีจิสเตอร์ที่ใช้ในการควบคุมการทำานของ ไทเมอร์/เคาน์เตอร์ภายในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช โดย T2CON ใช้สำหรับไทเมอร์/

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

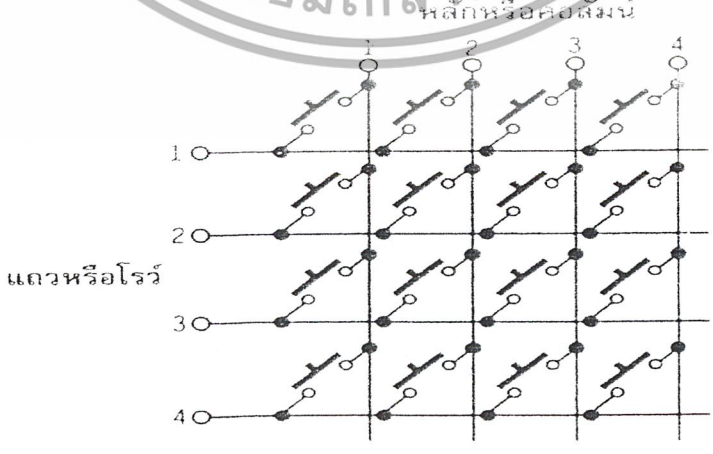
เคาน์เตอร์ 2 ของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเบอร์ AT89C52 และในอนุกรม AT89Sxx

รีจิสเตอร์ TMOD และ T2MOD เป็นรีจิสเตอร์ที่ใช้กำหนดโหมดหรือลักษณะในการทำงานของไทเมอร์/เคาน์เตอร์ ภายในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช โดย T2MOD ใช้สำหรับไทเมอร์/เคาน์เตอร์ 2 ของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเบอร์ AT89C52 และในอนุกรม AT89Sxx

รีจิสเตอร์ IE และ IPเป็นรีจิสเตอร์ที่เกี่ยวข้องกับการตอบสนองการอินเตอร์รัปต์ (interrupt : การขัดจังหวะการทำงานปกติของซีพียู) โดย IE เป็นรีจีสเตอร์สำหรับเอ็นเอเบิลหรือใช้ในการกำหนดลักษณะของการตอบสนองการอินเตอร์รัปต์ ในขณะที่ IP เป็นรีจิสเตอร์สำหรับกำหนดลำดับความสำคัญของการตอบสนองการอินเตอร์รัปต์ว่า จะให้ซีพียูตอบสนองการเกิดอินเตอร์รัปต์ในลักษณะก่อนหรือหลัง

2.2 ทฤษฎีเบื้องต้นสแกนคีย์

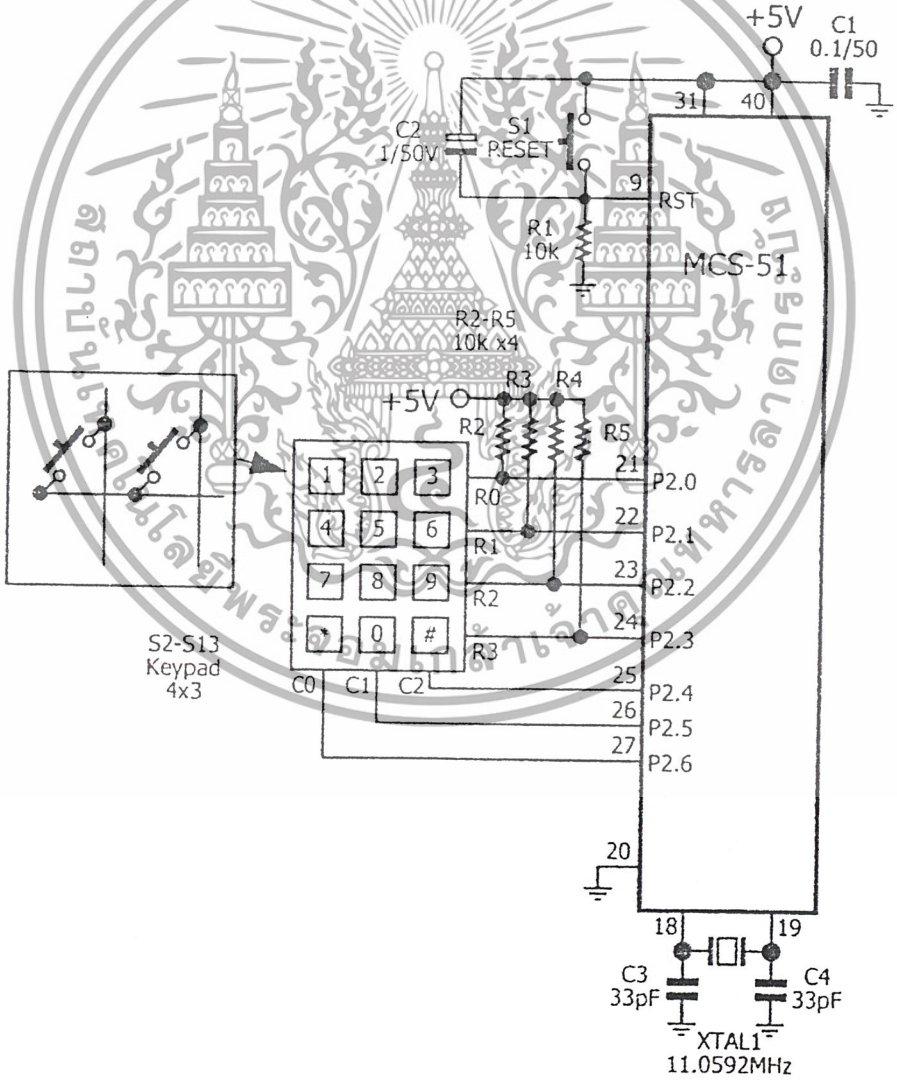
การอ่านค่าหรือรับค่าการกดสวิทช์ซึ่งก็เป็นอีกลักษณะหนึ่งของการใช้งานของตัวไมโครคอนโทรลเลอร์ต้องสามารถรองรับและเชื่อมต่อใช้งานร่วมกับวงจรของสวิทช์มี 2 ลักษณะใหญ่ๆ คือ ต่อเข้ากับไฟเลี้ยงหรือกราวด์โดยตรง เมื่อสวิทช์ตัวใดต่อวงจรสามารถอ่านค่าโดยตรง วงจรในลักษณะนี้ไม่มีความซับซ้อน สามารถอ่านค่าของสวิทช์ได้ง่ายและรวดเร็ว มีข้อเสียคือ ถ้าจำนวนของสวิทช์มีมากๆจำนวนของสายข้อมูลก็จะมีมากตาม ทำให้ระบบหรือวงจรโดยรวมมีขนาดใหญ่และสิ้นเปลือง



รูปที่ 2.15 วงจรของสวิทช์แบบเมตริกซ์หรือคีย์แพด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรของสวิทช์อีกลักษณะหนึ่งคือ การต่อวงจรแบบเมตริกซ์ (matrix switch) ดังในรูปที่ 2.15 สวิทช์จะถูกต่อกันในแนวแกนตั้งและแกนนอน จะเรียกแนวตั้งว่า หลักหรือคอลัมน์(column) ในขณะที่แนวนอนจะเรียกว่า แถวหรือโรว์ (row) ดังนั้นค่าของสวิทช์จะประกอบด้วย ตำแหน่งในแนวหลักและแถว กระบวนการที่จะทำให้ได้มาซึ่งค่าของสวิทช์มีขั้นตอนซับซ้อนพอสมควร แต่วงจรของสวิทช์แบบนี้มีข้อดีคือสามารถรองรับการเพิ่มของสวิทช์ได้อย่างสะดวก เพียงเพิ่มจำนวนสวิทช์ และแก้ไขซอฟต์แวร์อีกเล็กน้อยเท่านั้น ทำให้วงจรสวิทช์แบบเมตริกซ์เป็นที่นิยมใช้มากในระบบควบคุมอัตโนมัติหรือกึ่งอัตโนมัติที่จำนวนสวิทช์มากกว่า 8 ตัว ในการใช้งานทั่วไปจะเรียกสวิทช์แบบเมตริกซ์นี้ว่า คีย์แพด(keypad)



รูปที่ 2.16 วงจรเชื่อมต่อคีย์แพดเข้ากับไมโครคอนโทรลเลอร์ MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเชื่อมต่อคีย์แพดเข้ากับไมโครคอนโทรลเลอร์ MCS-51

ตัวอย่างวงจรในรูปที่ 2.16 จะใช้พอร์ต 2 ไมโครคอนโทรลเลอร์เชื่อมต่อเข้ากับคีย์แพดทั้ง 7 เส้นคือ สายคอลัมน์ 3 เส้นคือ C0-C2 และสายทางโรว์หรือแถวอีก 4 เส้นคือ R0-R3 โดยเฉพาะที่ขาพอร์ต P2.0-P2.3 จะต้องต่อตัวต้านทานพูลอัปไว้เพื่อกำหนดสภาวะเริ่มต้นที่ไม่มีการกดคีย์ โดยที่ไมโครคอนโทรลเลอร์จะทำการส่งข้อมูล "0" ไปยัง P2.6, P2.5 และ P2.4 ตามลำดับ ในทุกครั้งที่มีการส่งข้อมูลไปยังสายคอลัมน์ของคีย์แพดไมโครคอนโทรลเลอร์จะทำการอ่านค่าที่ P2.0-P2.3 เข้ามาด้วย หากไม่มีการกดค่าของ P2.0-P2.3 ก็จะเป็น "1" ทั้งหมด ถ้าหากมีการกดคีย์ค่าของ P2.0-P2.3 จะไม่เป็น 1111 อีกต่อไปเป็นการแจ้งให้ทราบว่ามีการกดคีย์แพดขึ้น จากนั้นไมโครคอนโทรลเลอร์ก็จะทำการค้นหาตำแหน่งต่อไป โดยการค้นหาตำแหน่งสิ่งที่จะได้มาอย่างแรกคือ ค่าตำแหน่งของคีย์นั้น จากนั้นก็จะนำค่าตำแหน่งนั้นไปเปิดตารางข้อมูล เพื่อจะได้หมายเลขของคีย์ที่กดอย่างแท้จริง ยกตัวอย่างจากวงจรในรูปที่ 2.16 หากคีย์ 0 ถูกกด ไมโครคอนโทรลเลอร์จะได้ค่าตำแหน่งของคีย์ 0 เป็น 0BH จากนั้นนำค่า 0BH นี้ไปเปิดตารางก็จะได้ข้อมูลเป็น 0 หมายถึงคีย์ 0 ซึ่งกระบวนการหลังจากนี้จะเป็นหน้าที่ของซอฟต์แวร์

2.3 การใช้งาน DOT MATRIX LCD MODULE

ปัจจุบัน LCD เป็นที่นิยมกันอย่างมาก สำหรับการแสดงผลในเครื่องมือเครื่องใช้ต่างๆทั้งนี้เนื่องจากมีความเหมาะสมในหลายๆ ด้าน เช่นการใช้กระแสต่ำ สามารถแสดงผลเป็นตัวอักษรและตัวเลข หรือแสดงเป็นกราฟฟิก (เฉพาะรุ่น) และในขณะนี้ผู้ผลิต LCD จะผลิต LCD รุ่นที่เป็น module ที่มีตัว LCD และวงจรควบคุมมาให้พร้อม (เรียกว่าLCM) ออกมา ทำให้ผู้ใช้สามารถต่อเข้ากับระบบไมโครคอนโทรลเลอร์ได้ง่ายและสะดวกสำหรับการเขียนโปรแกรม

LCD Module มีอยู่หลายรุ่นและคุณสมบัติแตกต่างกันออกไป แบ่งได้เป็น 2 กลุ่มคือแบบ Dot Matrix และ Graphic โดยแบบ Dot Matrix จะแสดงผลเป็นตัวอักษรขนาด 5x8 Dotและมีจำนวนตัวอักษรและบรรทัดแตกต่างกันออกไปในแต่ละรุ่น ส่วนแบบ Graphic จะแสดงผลในแบบ Bit Map คือสร้างเป็นภาพใดๆก็ได้ตามต้องการ แนวทางในการใช้งานของทั้ง 2 แบบต่างก็มีลักษณะใกล้เคียงกัน การใช้งานโดยทั่วไปมักจะเป็นแบบ Dot Matrix มากกว่าเนื่องจากราคาถูกกว่าและเพียงพอต่องานส่วนใหญ่

คุณสมบัติของ Dot Matrix LCD Module และสามารถสรุปเป็นข้อๆดังนี้

1.ตัวอักษรแสดงด้วย Dot Matrix ขนาด 5x8 Dot

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.สามารถต่อเข้ากับระบบไมโครคอนโทรลเลอร์ได้ 2 ลักษณะคือแบบ Memory Map และแบบผ่าน PORT ซึ่งจะใช้ขาสัญญาณทั้งหมด 14 PIN (ขั้วต่อ 16 PIN)

3.การใช้งานและสะดวก ระบบไมโครคอนโทรลเลอร์เพียงแค่ส่งข้อมูลให้กับ LCD Module เท่านั้น ข้อความก็จะปรากฏบนแผงแสดงและจะค้างไว้ตลอดทำให้ไม่ต้องเสียเวลาของระบบ

4.มีคำสั่งพิเศษสำหรับอำนวยความสะดวกมากมาย เช่น CLEAR DISPLAY, HOME CURSOR ON ,OFF CURSOR, BLANK CHARACTER ฯลฯ

5.สามารถแสดงผลเป็นตัวอักษรภาษาอังกฤษและตัวเลขได้160 ตัวและสัญลักษณ์พิเศษอีก 32 ตัว

6.กินกระแสน้อยและมีน้ำหนักเบา รวมทั้งทำงานได้ด้วยไฟเลี้ยงระดับ 5 โวลท์

ขาสัญญาณของ LCD Module

PIN	SYMBOL	LEVEL	FUNCTION
1	Vss	-	0 V GND
2	Vcc	-	+5 V Power Supply
3	Vee	-	+V For Liquid Crystal Drive
4	RS	H/L	Register Select H:Data Input L:Intruction
5	R/W	H/L	Input H:Data Input L:Data Write
6	E	H/L	Enable Signal (L->H)
7	DB 0	H/L	Data Bus Bit 0
8	DB 1	H/L	Data Bus Bit 1
9	DB 2	H/L	Data Bus Bit 2
10	DB 3	H/L	Data Bus Bit 3
11	DB 4	H/L	Data Bus Bit 4
12	DB 5	H/L	Data Bus Bit 5
13	DB 6	H/L	Data Bus Bit 6
14	DB 7	H/L	Data Bus Bit 7

ตารางที่ 2.4 ขาสัญญาณของ LCD Module

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.1 การต่อเข้ากับระบบไมโครคอนโทรลเลอร์

LCD Module จะเข้ากับระบบไมโครคอนโทรลเลอร์ได้ 2 ลักษณะดังได้กล่าวไปแล้วซึ่งทั้งแบบนี้จะมีข้อดีและข้อเสียแตกต่างกันออกไป โดยแต่ละแบบมีหลักการดังนี้

(1) การต่อแบบ Memory Map

1.สามารถต่อเข้ากับ Chip เบอร์ทั่วๆไปได้เช่น 8051 หรือ Z80 โดยจะทำให้ระบบที่จะใช้ตัวไมโครคอนโทรลเลอร์สามารถมองเห็น LCD Module ในลักษณะของ Memory ได้ทันที

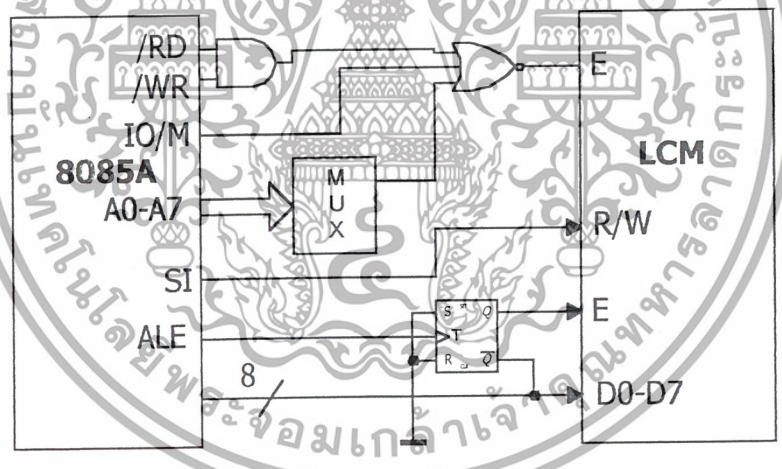
2.ผู้ใช้สามารถเขียนและอ่านข้อมูลจาก LCD Module ได้ ทำให้มองเสมือนว่าเป็นMemory Buffer ไปในตัว

3.เนื่องจากสามารถอ่านข้อมูลกลับได้ จึงทำให้สามารถตรวจสอบ Flag ความพร้อมในขณะที่ LCD Module กำลังทำงานได้

4.ใช้ได้กับบอร์ดที่มี LCD Bus มาให้พร้อมเท่านั้น

5.ทำให้กินพื้นที่ของหน่วยความจำไปส่วนหนึ่งและต้องมีการ Decode ละเอียดพอสมควร

6.การจัดขาสัญญาณจะต้องเป็นไปตามแบบของ Chip



รูปที่ 2.17 การต่อ LCD Module เข้ากับไมโครคอนโทรลเลอร์

(2) การต่อแบบ I/O Port

1.สามารถต่อเข้ากับ I/O Port ใดๆก็ได้โดยใช้สายสัญญาณจำนวน 11 เส้นและใช้โปรแกรมเป็นตัวสร้างสัญญาณขึ้นมา ให้ตรงกับข้อกำหนดของ LCD Module

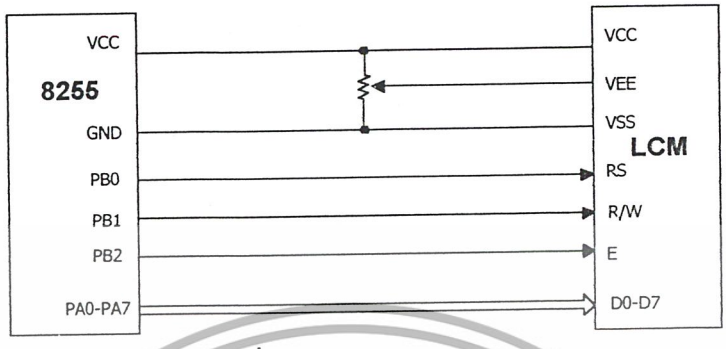
2.ผู้ใช้จะเขียนข้อมูลให้ LCD Module ได้อย่างเดียว ซึ่งผู้ใช้ควรจะกำหนด Memory ส่วนหนึ่งให้เสมือน Buffer ให้กับ LCD Module

3.เนื่องจากไม่สามารถอ่านข้อมูลกลับได้ จึงต้องใช้การหน่วงเวลาของระบบเอง เพื่อรอให้ LCD Module กระทำขบวนการต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.ใช้ได้กับบอร์ดทั่วๆไปที่มี Port

5.ไม่เปลืองส่วนของ Memory ในการใช้งาน



รูปที่ 2.18 การทำงาน LCM

(3) ชุดคำสั่งควบคุมและการแสดงข้อความ

ขาสัญญาณ Vee มีไว้สำหรับกำหนดความเข้มของตัวอักษร โดยถ้าต่อกับGNDจะมีความเข้มสูงสุดแต่ถ้ากับ Vcc จะมีความเข้มต่ำสุด ปกติ LCD รุ่นธรรมดา อาจจะต่อกับGNDไว้ให้เลยไม่ต้องใส่ VR ให้สิ้นเปลือง แต่ถ้ารุ่น SNT (มุมมองกว้าง) ให้ใช้ R 2K ต่อลงกราวด์อีกทีเพื่อให้ความเข้มที่เหมาะสม การเขียนหรืออ่านข้อมูลกับ LCD Module ก็คือ การกำหนดคุณสมบัติต่างๆในการใช้งานของ LCD ตามชุดคำสั่งควบคุมและรวมไปถึงการเขียนข้อมูลที่เป็นข้อความเพื่อให้ปรากฏบนแผงแสดงด้วย โดยมีรายละเอียดดังตารางต่อไปนี้

INSTRUCTION	RS	R/W	DATA BIT								EXETIME (micro Sec)
			7	6	5	4	3	2	1	0	
CLEAR DISPLAY	0	0	0	0	0	0	0	0	0	1	1640
CURSOR AT HOME	0	0	0	0	0	0	0	0	1	*	1640
ENTRY MODE SET	0	0	0	0	0	0	0	1	I/D	S	40
DISPLAY ON/OFF	0	0	0	0	0	0	1	D	C	B	40
DISPLAY SHIFT	0	0	0	0	0	1	S/C	R/L	*	*	40
FUNCTION SET	0	0	0	0	1	DL	N	F	*	*	40
SET CGRAM ADD.	0	0	0	1	CGRAM ADDRESS						40
SET DDRAM ADD.	0	0	1	DDRAM ADDRESS							40
BUSY ADD. READ	0	1	BF	ADDRESS							0
CGRAM,DDRAM WR	1	0	WRITE DATA								40
CGRAM,DDRAM RD	1	1	READ DATA								40

ตารางที่ 2.5 ชุดคำสั่งควบคุมและการแสดงข้อความ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(4) ความเข้าใจพื้นฐาน

1.การเขียนข้อมูลให้กับ LCD Module จะแบ่งเป็น 2 ลักษณะ คือ Instruction และ Data โดยที่จะกำหนดด้วยขาของสัญญาณ RS คือถ้า RS=0 จะหมายถึงส่งสัญญาณควบคุม (Instruction)หรืออ่านค่า Flag สภาพการทำงานของ LCD Module และถ้า RS=1 จะหมายถึงการเขียนหรืออ่านข้อมูล (Data) กับ LCD Module

2.หลักการในการเขียนข้อมูลให้ LCD Module นี้คือเมื่อมีการเขียนข้อมูลไปแล้ว ตัว LCD Module จะใช้เวลาในการทำงานชั่วขณะหนึ่ง ซึ่งระบบไมโครคอนโทรลเลอร์สามารถตรวจสอบได้จาก Busy Flag (BF) และถ้าเรียบร้อยแล้วจึงจะสามารถเขียนข้อมูลอันต่อไปได้ ในกรณีที่มีการต่อวงจรเป็นแบบ I/O Port คือ ไม่สามารถอ่านข้อมูลย้อนกลับได้ ระบบไมโครคอนโทรลเลอร์ก็จะต้องใช้วิธีหน่วงเวลาแทน

3. การเขียนข้อมูลให้กับ LCD Module นี้สามารถทำได้ทั้งแบบ 8 bit และ 4 bit โดยกรณี 4 bit จะใช้สายสัญญาณข้อมูลเพียง 4 เส้นคือ DB4-DB7 (ใช้สำหรับระบบไมโครคอนโทรลเลอร์แบบ 4 bit หรือเพื่อเป็นการประหยัดสาย) การเขียนข้อมูลจะกระทำเหมือนกับ 8 bit เพียงแต่ให้เขียน 2 ครั้งคือ DB4-DB7 ก่อนแล้วตามด้วย DB0-DB3 และจะต้องกำหนดคุณสมบัติตามค่า DL ในคำสั่ง Function Set ด้วย

4.DDRAM (Display Data Dam) คือ หน่วยความจำภายในตัว LCD Module ที่เป็น Buffer ของข้อมูลโดยถ้าเขียนรหัส ASCII ใดๆ ลงไปในหน่วยความจำนี้ ก็จะปรากฏเป็นตัวอักษรที่แผงแสดงทันที

5.CGRAM (Character Generator Ram) คือหน่วยความจำภายในตัว LCD Module สำหรับเก็บภาพตัวอักษรที่ผู้ใช้สร้างได้เอง (8 ตัว) โดยจะอ้าง Address ได้ทั้งหมด 64 byte คือ 8 ตัวอักษรคูณกับ 8 แถว

2.3.2 รายละเอียดของแต่ละคำสั่ง

1.CLEAR DISPLAY

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	0	0	0	1

สำหรับการ Clear Display โดยจะทำการเขียนตัวอักษร Space ลงไปใน DDRAM ทั้งหมด และกำหนดค่า DDRAM Address ให้เป็น 0 พร้อมทั้ง Cursor จะกลับไปที่ตำแหน่งซ้ายบนสุดของจอภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.CURSOR AT HOME

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	0	0	1	*

สำหรับกำหนดค่า DDRAM Adress ให้เป็น 0 พร้อมทั้ง Cursor จะไปที่ตำแหน่งซ้ายบนสุดของจอภาพ โดยที่ข้อมูลใน DDRAM ไม่การเปลี่ยนแปลง

3.ENTRY MODE SET

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	0	1	I/D	S

I/D=0 กำหนดทิศทางของ Cursor และ DDRAM ให้เป็นแบบ Decrement

I/D=1 กำหนดทิศทางของ Cursor และ DDRAM ให้เป็นแบบ Increment

S=0 เมื่อเขียนข้อมูลแล้ว ตัว Cursor จะถูกเลื่อนไปตามทิศทางของค่า I/D

S=1 เมื่อเขียนข้อมูลแล้ว ตัว Cursor จะอยู่กับที่ และตัวอักษรจะถูกดันไปตามทิศทางของค่า I/D

กำหนดค่า I/D และ S นี้ให้กำหนดก่อนการเขียนข้อมูลใน DDRAM และเมื่อกำหนดแล้วจะไม่ต้องคำสั่ง Clear Display อีก

4.DISPLAY ON/OFF

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	1	D	C	B

D=0 กำหนดให้ Off Display

D=1 กำหนดให้ ON Display

C=0 กำหนดให้ Off Cursor

C=1 กำหนดให้ ON Cursor โดย Cursor จะเป็นขีดใต้ตัวอักษร

B=0 กำหนดให้ไม่มีการกระพริบที่ตำแหน่ง Cursor

B=1 กำหนดให้มีการกระพริบที่ตำแหน่ง Cursor (กระพริบเป็นรูป [])

5. DISPLAY SHIFT

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	1	S/C	R/L	*	*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

S/C=0 กำหนดให้เลื่อน Cursor ตามทิศทาง R/L ไป 1 ตำแหน่ง

S/C=1 กำหนดให้เลื่อนข้อความบนแผงแสดงตามทิศทาง R/L ไป 1 Column (เลื่อนทุกบรรทัด)

R/L=0 กำหนดให้มีทิศทางไปทางซ้าย R/L=1 กำหนดให้มีทิศทางไปทางขวา

6.FUNCTION SET

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	1	DL	N	F	*	*

DL=0 กำหนดให้การติดต่อกับ LCD Module เป็นแบบ 4 bit

DL=1 กำหนดให้การติดต่อกับ LCD Module เป็นแบบ 8 bit จะสังเกตว่าการกำหนดค่า DL นี้สามารถกระทำได้ที่ DB4-DB7 ซึ่งถ้ามีกำหนดให้เป็นแบบ 4 bit ตั้งแต่ครั้งแรก หลังจากจ่ายไฟเลี้ยงก็จะทำให้ LCD Module มีการรับข้อมูลแบบ 4 bit ทันที

N=0 กำหนดจำนวนบรรทัดแบบ 1/8 Dutyและ 1/11 Duty

N=1 กำหนดจำนวนบรรทัดแบบ 1/16 Duty

F=0 กำหนดให้มีตัวอักษรเป็นแบบ 5*7 Dots

F=1 กำหนดให้มีตัวอักษรเป็นแบบ 5*10 Dots

(กรณีที่ LCD Module เป็นแบบ 5*7 ก็จะไม่มีผลอะไร)

7.SET CGRAM ADDRESS

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	1	CGRAM ADDRESS					

สำหรับการกำหนด Address ของ CGRAM เมื่อได้ทำการกำหนดไว้แล้วการอ่านและเขียน Data ที่ต่อจากนี้จะเป็นไปตาม Address ที่กำหนดทันที

8. SET DDRAM ADDRESS

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	1	DDRAM ADDRESS						

สำหรับการกำหนด Address ของ DDRAM เมื่อได้ทำการกำหนดไว้แล้วการอ่านและเขียน Data ที่ต่อจากนี้จะเป็นไปตาม Address ที่กำหนดทันที ตำแหน่งของ Address ในแต่ละรุ่นจะมีความแตกต่างกันบ้าง เพราะจำนวนตัวอักษรต่อบรรทัดไม่เท่ากัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9.BUSY FLAG AND ADDRESS READ

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	1	BF	ADDRESS						

สำหรับการอ่านค่า BF (Busy Flag) ซึ่งบอกถึงความพร้อมของ LCD Module ในการรับข้อมูล ถ้า BF=0 หมายความว่าพร้อมที่จะรับข้อมูลต่อไปได้ แต่ถ้า BF=1 หมายความว่ายังไม่พร้อม นอกจากนี้ยังเป็นการอ่านค่า Address ของ CGRAM หรือ DDRAM ด้วย

2.3.3 การอ่านและเขียนข้อมูลกับ DDRAM/ CGRAM

1.WRITE DATA TO DDRAM OR CGRAM

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
1	0	DATA							

สำหรับการเขียนข้อมูลลงหน่วยความจำ DDRAMหรือ GRAMโดยเมื่อทำการเขียนแล้ว Address จะถูกเพิ่มหรือลดลงโดยอัตโนมัติตามที่กำหนดจากค่า I/D ในคำสั่ง Entry Mode Set และการเขียนจะเป็น CGRAM หรือDDRAMทั้งนี้ก็ขึ้นกับว่าก่อนหน้าคำสั่งนี้มีการกำหนด Addressที่ใด

2.READ DATA FROM DDRAM OR CGRAM

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
1	1	DATA							

สำหรับการอ่านข้อมูลจากหน่วยความจำDDRAMหรือGRAMโดยเมื่อทำการอ่านแล้ว Address จะถูกเพิ่มหรือลดลงโดยอัตโนมัติตามที่กำหนดจากค่า I/D ในคำสั่ง Entry Mode Set และการ อ่านจะเป็น CGRAM หรือ DRAM ทั้งนี้ก็ขึ้นกับว่าก่อนหน้าคำสั่งนี้มีการกำหนด Address ที่ใด

2.3.4 การเขียนโปรแกรมควบคุม

1.เมื่อจ่ายไฟเลี้ยงให้กับ LCD Module ครั้งแรก ภายในจะมีการ RESET ระบบโดยอัตโนมัติซึ่งจะใช้เวลา 10 mS. หลังจากแรงไฟขึ้นถึง 4.5V แล้วทั้งนี้ระบบ RESET ดังกล่าวจะกระทำสิ่งต่างๆ ต่อไปดังนี้

- ทำการ Clear จอภาพทั้งหมด (Clear Display)
- กำหนดคุณสมบัติด้วยคำสั่ง Function Set คือ N = 0 (แสดงข้อมูล 1 บรรทัด)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

F=0 (กำหนดตัวอักษรแบบ 5*7 Dots), DL=1 (ติดต่อกับระบบไมโครคอนโทรลเลอร์ในแบบ 8 บิต)

- กำหนดคุณสมบัติด้วยคำสั่ง Display ON/OFF คือ D=0 (ไม่แสดงข้อมูล) C=0 (Cursor OFF), B=0 (Blank OFF)

- กำหนดคุณสมบัติด้วยคำสั่ง Entry Mode Set คือ I/D=1 (Increment), S=0(No Shift)

การใช้งาน LCD Module ต้องรอให้ขบวนการ ReSet ภายในทำงานเรียบร้อยก่อน ซึ่งจะตรวจสอบได้ด้วย BF (Busy Flag) หรืออาจจะใช้การหน่วงเวลาก็ได้

2. การใช้งาน LCD Module จะต้องเกี่ยวข้องกับทางด้านโปรแกรมเป็นส่วนใหญ่ชุดคำสั่งต่างๆรวมทั้งการอ่านหรือเขียนข้อมูลนั้น จะถูกกำหนดด้วยขาสัญญาณทั้งหมดที่มีอยู่ปกติโปรแกรมจะต้องกำหนดคุณสมบัติต่างๆที่ต้องไว้ที่ส่วนต้น จากนั้นจะเป็นการอ่านและเขียนข้อมูลลงใน DDRAM ซึ่งก็คือข้อความที่จะให้แสดงผลนั่นเอง

2.4 ทฤษฎีอินฟราเรดรีโมทคอนโทรล

ในปัจจุบันการดำรงชีวิตของคนเรานั้นต้องการสิ่งอำนวยความสะดวกในรูปแบบของเครื่องใช้ไฟฟ้าที่ทันสมัย ซึ่งในการควบคุมเครื่องใช้ไฟฟ้าเหล่านี้มักนิยมใช้รีโมทคอนโทรลเป็นตัวควบคุม เนื่องจากใช้งานง่าย เครื่องส่งใช้กำลังไฟน้อยและราคาไม่แพง

รีโมทคอนโทรล หมายถึง ระบบควบคุมระยะไกล ที่ทำหน้าที่เสมือนแขนขาของมนุษย์ จึงทำให้เป็นที่นิยมใช้กันอย่างแพร่หลาย และเริ่มเป็นสิ่งจำเป็นควบคู่กับเครื่องใช้ไฟฟ้าในยุคนี้จึงที่จะให้ความสนใจกันอย่างละเอียด การทำงานของระบบดาต้าโค้ด

ตามที่ทราบแล้วว่า ระบบตัวเลขที่ใช้ในรีโมทคอนโทรลเป็นตัวเลขฐาน 2 ซึ่งมีสภาพทางลอจิก “0” และ “1” หรือหากพูดถึงระดับแรงไฟสภาพลอจิก 1 ก็คือสภาพที่วงจรสามารถจ่ายแรงดันไฟออกไปยังอุปกรณ์ในวงจรได้ ส่วนสภาพลอจิก 0 ก็คือสภาพที่แรงดันไฟใกล้เคียงศูนย์โวลท์ นั่นคือระบบสัญญาณพัลซ์นั่นเอง

กล่าวเฉพาะในส่วนของข้อมูลหลายๆบิต การที่จะแยกออกมาได้อย่างชัดเจนว่าอันไหนเป็นลอจิก0 อันไหนเป็นลอจิก 1 จำเป็นต้องใช้ฐานเวลาเข้ามาเป็นตัวแยกในทางปฏิบัติเป็นเรื่องค่อนข้างยากอยู่พอประมาณ เนื่องจากบางครั้งผู้ใช้รีโมทอาจจะกดช้านานแต่ละครั้งอาจจะไม่เท่ากัน บางคนอาจกดแช่ไว้ บางคนอาจกดคีย์แล้วปล่อยเลย

ดังนั้นเราจึงเลือกเอาระบบสแกนพัลซ์ซึ่งเป็นรูปแบบการกวาดข้อมูลมาทดแทน เพื่อแก้ปัญหาเรื่องการที่จะต้องกำหนดฐานเวลา โดยหากเป็นลอจิก 0 เราจะให้พัลซ์ที่ออกมามีพัลซ์ชิดหรือแคบกว่าและหากถ้าเป็นลอจิก 1 เราจะให้ระยะของพัลซ์ทอดยาวออกไป การกำหนดฟังก์ชั่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำได้โดยการใช้หลักการของคีย์แบบเมตริกซ์ (Matrix) เพื่อลดขาใช้งานของไอซีและสายเชื่อมโยงต่างๆให้น้อยลง

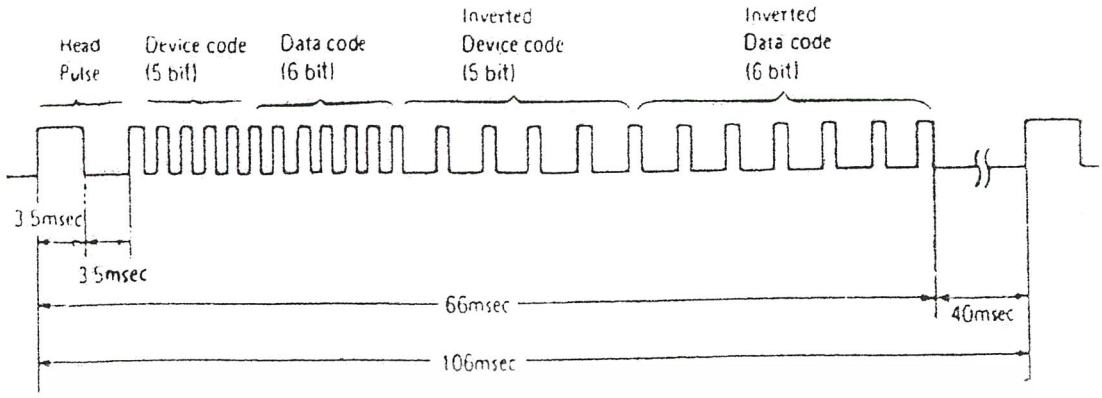
เมื่อเรากำหนดพัลซ์ ในลักษณะที่กล่าวมา มีผลทำให้ฐานเวลาของข้อมูล (DataTime)แต่ละตัวมีความแตกต่างกันออกไป ยกตัวอย่างเช่นในกรณีที่ข้อมูลนั้นๆ มีสภาพลอจิก 0 เป็นจำนวนมากบิตจะมีผลทำให้เวลาของข้อมูลสั้น หากมีสภาพทางลอจิก 1 มากกว่าจะมีผลทำให้เวลาของข้อมูลยาวนานขึ้น ดังนั้นในยุคปัจจุบันที่ใช้มูลหลายๆบิต จะเกิดปัญหาในเรื่องความเหลื่อมของเวลาขึ้น ปัญหานี้มีผลต่อการรับข้อมูลในเครื่องรับเป็นอันมาก รวมไปถึงหากเราจะใส่ข้อมูลอย่างอื่นฝากไปด้วยจะทำได้อย่างลำบาก เราจะพบว่ารีโมทคอนโทรลในปัจจุบันนี้ สามารถผ่านข้อมูลพิเศษเข้าไปได้มากทีเดียว เครื่องรับแยกได้อย่างไรหากเวลาไม่แน่นอนจึงต้องมีการอินเวิร์ต (Inverse) ข้อมูลทั้งหมดให้กลายเป็นตรงข้ามเพื่อรักษาเวลาให้คงที่ และเพื่อไม่ต้องส่งสัญญาณซิงโครไนซ์ไปทำการควบคุมเครื่องรับซึ่งนับเป็นความซับซ้อนยุ่งยาก

สัญญาณแทน logic "1" สัญญาณแทน logic "0"

รูปที่ 2.19 ลักษณะทางลอจิกในระบบสแกนพัลซ์

ยกตัวอย่างเช่นเราส่งข้อมูลหรือดาต้าโค้ดเป็นข้อมูลขนาด 6 บิต เป็นดังนี้ 000001 เราจะพบว่าเมื่อถูกแปรสภาพเป็นพัลซ์แล้ว ลอจิก 0 มีจำนวน 5 บิต มีลอจิก 1 จำนวน 1 บิต จะพบว่าเวลารวมของพัลซ์สั้นมาก และหากเราส่งข้อมูลเป็น 111111 จะพบว่าเวลารวมของพัลซ์จะยาวมาก แต่ถ้าหากเรานำมาอินเวิร์ตจะพบว่าเวลารวมทั้งหมดเท่ากัน นั่นคือข้อมูล 000001 เมื่อนำมาอินเวิร์ตแล้วจะเป็น 111110 ข้อมูลรวมคือ 000001111110 (แยกเป็นลอจิก 0 รวม 6 บิต ลอจิก 1 รวม 6 บิต) พบว่าระยะเวลาของข้อมูลจะเท่ากันโดยอัตโนมัติ ดังนั้นวิธีการอินเวิร์ตข้อมูลคือวิธีการแก้ปัญหาฐานเวลาที่เกิดขึ้นกับเครื่องรับหรือตัวรับนั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.20 คาต้าสตรีม

2.4.1 สัญญาณมาตรฐานรีโมทคอนโทรล

การที่เราจะให้วงจรภาครับของรีโมทคอนโทรลรับรู้และทำการแยกสัญญาณหรือข้อมูลต่างๆ ได้อย่างถูกต้องจะต้องมีข้อมูลตัวอื่นเข้ามาเพื่อแก้ไขสิ่งซึ่งอาจผิดพลาด ดังนั้นนอกจากข้อมูล (Data) ซึ่งเป็นข้อมูลหลักที่เราส่งไปเป็นตัวเลขฐาน 2 ซึ่งในตอนนี้อยู่ในรูปของสแกนพัลซ์พร้อมด้วยการอินเวิร์ตข้อมูลเพื่อรักษาค่าเวลาแล้วยังต้องมีข้อมูลอื่นเป็นส่วนประกอบ ซึ่งในกรณีนี้เราขอใช้ตัวอย่างของเครื่องเนชั่นแนลเป็นตัวเดินเรื่อง (ซึ่งโดยหลักการจะเหมือนกันทุกยี่ห้อ เพียงแต่ว่าใครจะใช้ข้อมูลกี่บิตและใช้ฐานเวลาเท่าไรนั่นเอง) นี่คือการยกตัวอย่างเพื่อนำไปสู่การทำความเข้าใจ และทำการเปรียบเทียบกับยี่ห้ออื่นๆต่อไป

1.ดาต้าโค้ด (Data Code) ในกรณีของรีโมทคอนโทรลซึ่งเป็นตัวส่งของเนชั่นแนลใช้ข้อมูลขนาด 6 บิตเป็นข้อมูลหลักที่ส่งออกไปควบคุมวงจรในส่วนภาครับโดยคีย์แมตริกซ์จะเป็นตัวสั่งงานเข้าสู่ระบบการเข้ารหัสข้อมูล กำหนดความเป็นไปของแต่ละฟังก์ชั่น

2.อินเวิร์ตดาต้าโค้ด (Inverse Data Code) เป็นการกลับลอจิกของข้อมูลหลักเพื่อรักษาฐานเวลาให้คงที่ทุกข้อมูล ซึ่งเป็นข้อมูลขนาด 6 บิตเหมือนกัน

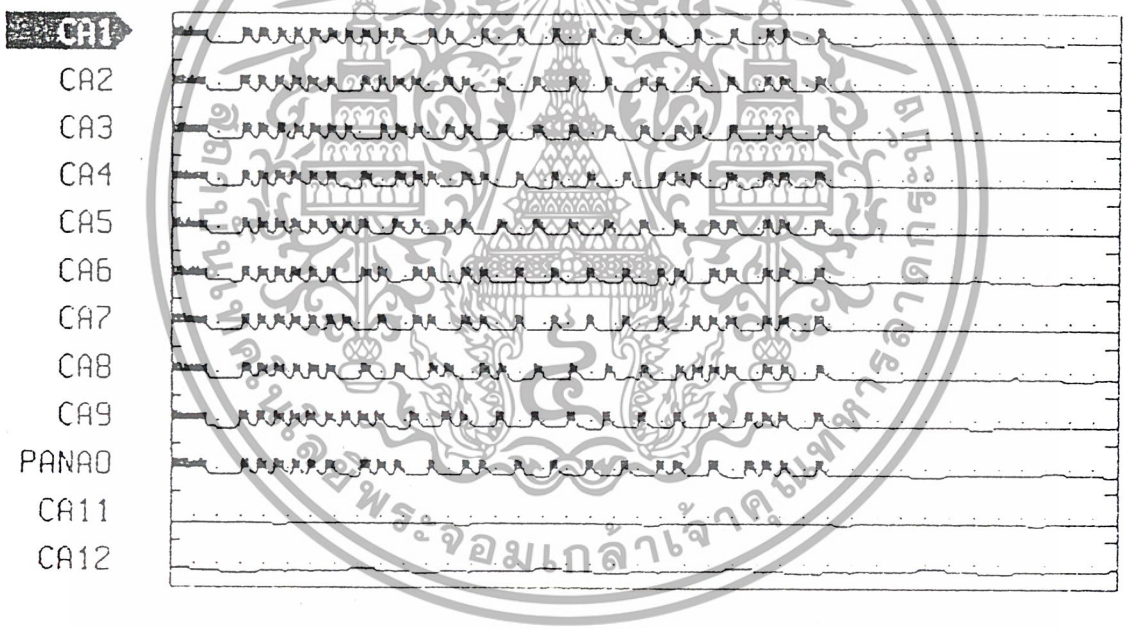
3.ดีไวส์โค้ด (Device Code) หรือบางครั้งใช้ศัพท์ “คัสตอมโค้ด” (Costom Code) เนื่องจากในปัจจุบันเครื่องใช้ต่าง ๆ ล้วนเป็นระบบรีโมทคอนโทรลแบบอินฟาเรดกันทั้งนั้น การสั่งงานจากรีโมทคอนโทรลอาจจะมีการรบกวนอุปกรณ์อื่นได้ อย่างเช่นเครื่องรับโทรทัศน์อาจจะพ่วงอยู่กับเครื่องเล่นวีดีโอเทป จูนเนอร์พ่วงอยู่กับเครื่องขยายเสียง และคอมแพคดิสก์ เลเซอร์ดิสก์พ่วงอยู่กับเครื่องรับโทรทัศน์ หรืออื่นๆ กรณีเช่นนี้หากเราสั่งเครื่องหนึ่งเครื่องใด เครื่องที่ต่อร่วมอยู่ด้วยก็สามมารับเอาข้อมูลฟังก์ชั่นการทำงานเข้าไปด้วย จึงมีการสร้างข้อมูลของ เครื่องเล่นแต่ละอย่างให้แตกต่างกันออกไป ตัวอย่างเช่น เครื่องรับโทรทัศน์เราใช้ดีไวส์โค้ด (Device Code) ซึ่งเป็นโค้ดที่ใช้แยก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประเภทของเครื่องใช้ระบบข้อมูล 00000 (5 บิต) ในขณะที่เครื่องเล่นวีดีโอเทปเราใช้ดีไวส์โค้ด 11111

4.อินเวิร์ตดีไวส์โค้ด (Inverse Device Code) เป็นการกลับข้อมูลดีไวส์โค้ดเพื่อรักษาเวลาเช่นเดียวกับระบบข้อมูลหลัก (Data Code) แน่นอนข้อมูลดังกล่าวต้องมี 5 บิต เหมือนดีไวส์โค้ด

5.เฮดพัลซ์ (Head Pulse) การกลับข้อมูลหรืออินเวิร์ตเป็นเพียงการรักษาเวลาของข้อมูล แต่การใส่เฮดพัลซ์เป็นวิธีการตรวจเช็คข้อมูลเพิ่มความแน่นอนของข้อมูลขึ้นอีกเพราะในบางครั้งเราอาจะกดคีย์แช่ไว้นานๆความต่อเนื่องของข้อมูลจะมีตลอด นั่นหมายความว่าเราจะแยกแยะได้อย่างไรว่าอะไรคือข้อมูลหลักอะไรคือข้อมูลรอง จึงมีเฮดพัลซ์ขึ้นมา โดยเฮดพัลซ์จะเป็นสัญญาณนำร่องก่อนจะมีข้อมูลต่างๆส่งเข้ามา และในขณะที่เราส่งอย่างต่อเนื่องจะมีเฮดพัลซ์ ส่งออกมาคั่นเป็นช่วงๆ ให้เครื่องรับสามารถแยกกลุ่มข้อมูลออกมาได้

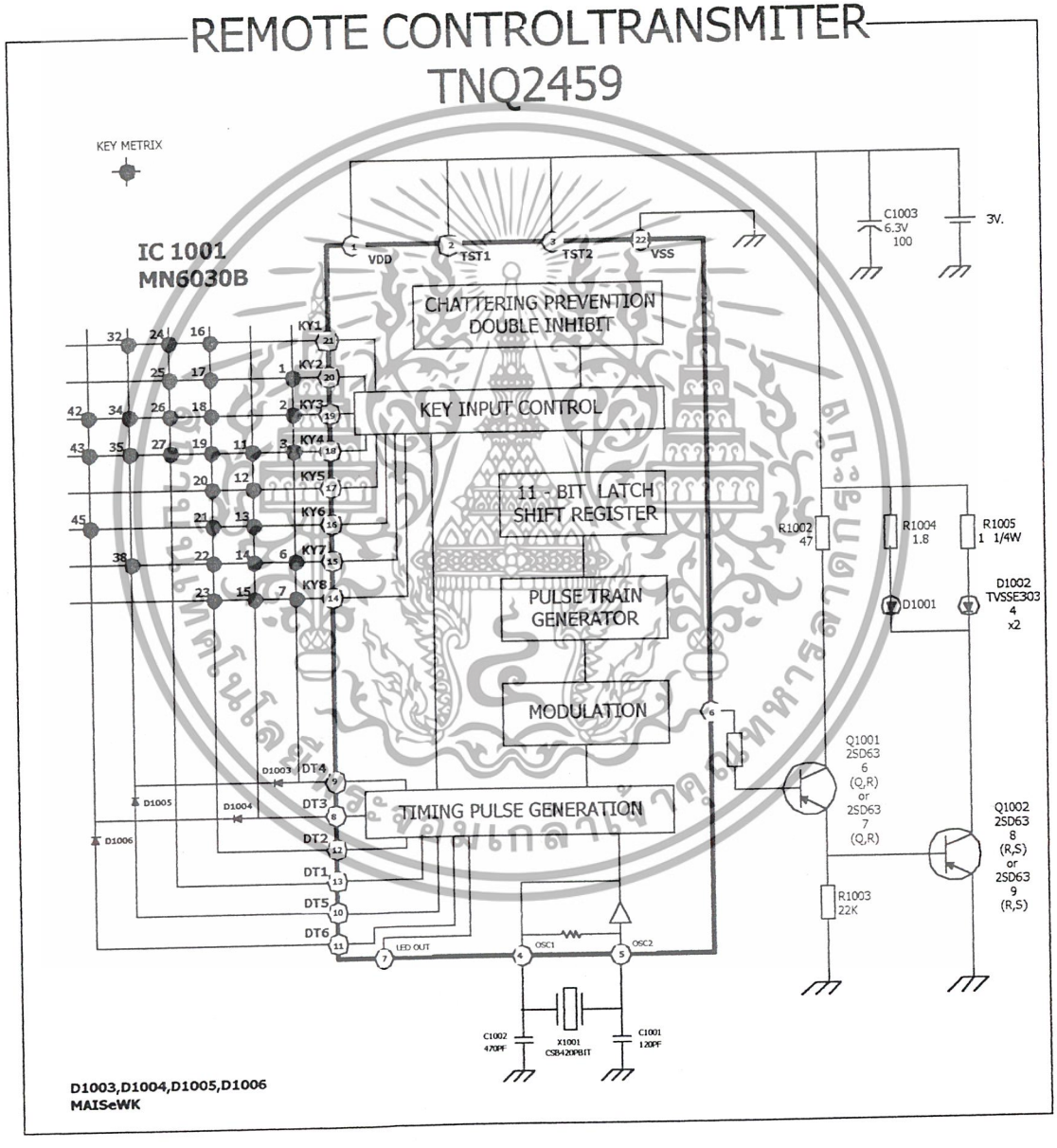


รูปที่ 2.21 ตัวอย่างสัญญาณรีโมทคอนโทรล

สังเกตุว่ารีโมทคอนโทรลจะส่งสัญญาณออกมาแต่ละคีย์ที่กดจะใช้เวลาที่เท่ากัน เนื่องจากมีการอินเวิร์ตดีไวส์โค้ดและ อินเวิร์ตดาต้าโค้ด ตามลำดับดาต้าสตรีม (Data Stream) หรือระบบในการเรียงข้อมูล จะเป็นไปตามรูปที่ 2.20 โดยเฮดพัลซ์จะเข้ามาเป็นอันดับแรกเป็นการบอกว่าตอนนี้ออปชั่นภายในตัวส่งสัญญาณรีโมทคอนโทรลแบบอินฟราเรด พร้อมที่จะทำงานแล้วซึ่งการที่ตัวตรวจสอบหรือตัวกระตุ้นความพร้อมของตัวรับว่าต่อไปนี้จะมีการส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเครื่องรับรับรู้การทำงาน (ซึ่งเรากดคีย์ออกไป สัญญาณเฮดพัลซ์บางทีเราเรียกว่าตัว “คอลล์”) เครื่องรับแสดงการรับรู้การเรียกข้อมูล (Call) ด้วยการกระพริบของแอลอีดี และข้อมูลลำดับต่อมาคือดีไวส์โค้ด 5 บิตตามด้วยดาต้าโค้ด 6 บิต อินเวิร์ตดีไวส์ 5 บิตและปิดท้ายด้วยอินเวิร์ตดาต้าโค้ด 6 บิตก่อนทิ้งช่วงให้เกิดเฮดพัลซ์ครั้งต่อไป ให้ตัวรับสามารถรับรู้ความต่อเนื่องของข้อมูลอีกครั้งหรือหลายๆครั้งต่อไป

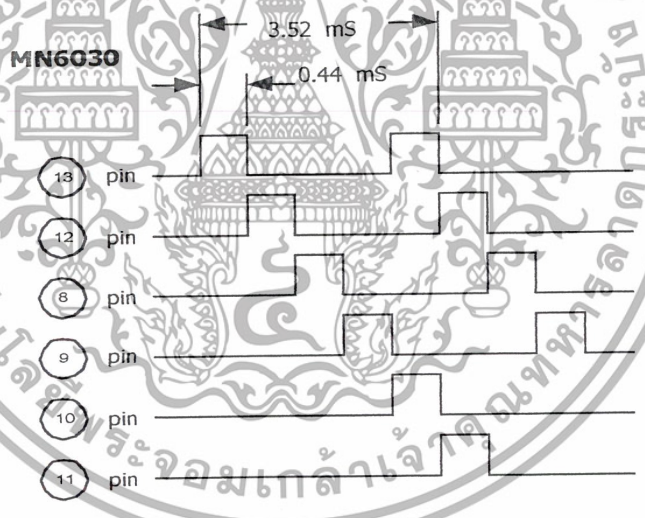


รูปที่ 2.22 เครื่องส่งรีโมทคอนโทรลที่ใช้ไอซี เบอร์ MN 60330B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การจะให้ไอซี ที่เป็นตัวส่งรีโมทคอนโทรลแบบรับรู้รหัสข้อมูลใดๆนั้น วงจรเอ็นโค้ดเดอร์จะต้องทำการเข้ารหัส โดยการนำเอาสแกนพัลส์ที่เกิดจากวงจรไทม์มิ่งเจนเนอเรเตอร์ไปเป็นตัวอ้างอิงให้คีย์เมตริกซ์แต่ละตัวส่งข้อมูลที่ไม่ซ้ำซ้อนกันเข้าสู่วงจรเข้ารหัสข้อมูล และจะมีการเรียงลำดับข้อมูลต่างๆ ให้เป็นขั้นตอนที่ถูกต้องเนื่องจากรีโมทคอนโทรลแบบนี้เป็นการส่งแบบไร้สาย จึงอาจจะเกิดการรบกวนในระหว่างที่ความถี่เหล่านี้เดินทางได้ (ความถี่ของแสงอินฟราเรด หรือความถี่ต่ำกว่าสีแดง) จึงต้องมีการผสมความถี่วิทยุพ่วงเข้าไปในขั้นตอนสุดท้าย

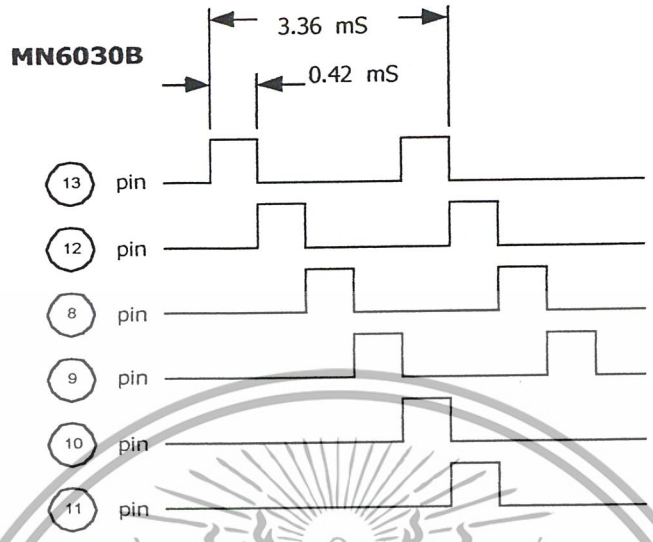
จากรูปที่ 2.22 เป็นวงจรตัวอย่างของเครื่องส่งรีโมทคอนโทรล เบอร์ MN 60330B โดยภาวะปกติเราจ่ายไฟ 3 โวลท์ให้กับเครื่องส่งชุดนี้แต่การที่วงจรจะนำกระแสได้นั้นต้องอยู่กับบอนดิ้งออปชั่นภายใน ดังนั้นแม้ว่าไฟเลี้ยงวงจรจะต่ออยู่ตลอดเวลา แต่ถ่าน 3 โวลท์ไม่ได้ถูกดึงกระแสแต่อย่างใด สวิทซ์ออน- ออฟของระบบจ่ายไฟในตัวเครื่องส่งรีโมทจึงไม่จำเป็นต้องมี เมื่อคีย์บอร์ดซึ่งเป็นเมตริกซ์ถูกกด ระบบอินพุตจะไปกระตุ้นบอนดิ้งออปชั่นให้มีการจ่ายไฟให้วงจรเข้าสู่สภาวะปกติวงจรออสซิลเลเตอร์ซึ่งควบคุมความถี่ด้วยคริสตอล 420 กิโลเฮิร์ตซ์สามารถทำงานได้



รูปที่ 2.23 สแกนพัลซ์ที่ออกจากคีย์เอาต์พุตขาต่างๆ

ระยะเวลาหรือความถี่ของสแกนพัลซ์ออกมาเท่ากันตลอดเวลา ไม่ว่าจะใช้ตัวส่งรีโมทคอนโทรลเบอร์ไหน แต่ข้อสำคัญคือตัวรับต้องถอดรหัสข้อมูลของตัวส่งให้ได้ก็พอ ปัญหาต่อมาคือเนื่องจากเราส่งสแกนพัลซ์ด้วยความถี่คงที่ อินพุตหรือระบบเอ็นโค้ดเดอร์จะแยกข้อมูลคำสั่งได้อย่างถูกต้อง หากพิจารณาจากรูปที่ 2.23 และรูปที่ 2.24 จะพบว่าแม้ว่าระยะเวลาหรือความถี่ของสแกนพัลซ์ในแต่ละขาจะเท่ากัน แต่เวลาของพัลซ์แต่ละตัวจะเหลื่อมกันอยู่ นี่เองที่ทำให้วงจรเข้ารหัสสามารถรับรู้คำสั่ง และแยกแยะคำสั่งที่แตกต่างกันออกมาได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.24 สแกนพัลซ์ของเบอร์ MN 60330B

สแกนพัลซ์ จะถูกส่งเข้าไปยังขาคีย์บอร์ดอินพุตซึ่งได้แก่ขา 14 ถึง 21 ส่งผ่านไปได้หรือไม่ได้ขึ้นอยู่กับการกดคีย์เมตริกซ์ของผู้ใช้เครื่อง คีย์อินพุตจะรับรู้ฟังก์ชั่นต่างๆ โดยการจับช่วงเวลาของพัลซ์เทียบกับฐานเวลาที่มาจากวงจรไทม์มิ่ง พัลซ์ เจนเนอเรเตอร์

ในระบบคีย์เมตริกซ์ทั่วไปที่ไม่ใช่ระบบสแกนพัลซ์ การเกิดฟังก์ชั่นการทำงานสามารถคำนวณฟังก์ชั่นได้จากขาอินพุตทางโรว์ (Row) และทางคอลัมน์ (Column) เอาสองอินพุตนี้มาคูณกัน อย่างเช่นเครื่องรับโทรศัพท์มีขาเมตริกซ์ทางโรว์ 4 ขา ทางคอลัมน์ 3 ขา ฟังก์ชั่นการใช้งานของคีย์กดโทรศัพท์แบบนี้จึงมีได้แค่ 12 ฟังก์ชั่นอย่างในกรณีของรีโมทตัวนี้ เราถือว่าขา 14 ถึง 21 เป็นขาคีย์อินพุตหรือขาทางโรว์ และขาคีย์เอาต์พุตซึ่งได้แก่ขา 8 ถึง 10 เป็นขาทางคอลัมน์ เราจะพบว่าฟังก์ชั่นทั่วไปเท่ากับ 8 คูณ 5 ซึ่งจะได้ 40 ฟังก์ชั่น แต่เราบอกว่าหากข้อมูล 6 บิต เราจะสามารถขยายฟังก์ชั่นได้ถึง 64 ฟังก์ชั่นตามความสามารถสูงสุด 64 ฟังก์ชั่น โดยการเพิ่มไดโอดเข้าไปที่ขาคีย์เมตริกซ์เพื่อนำเอาสแกนพัลซ์ของขาเอาต์พุตขาหนึ่งไปรวมกับอีกขาหนึ่งเสมือนกับการสร้างสแกนพัลซ์ขึ้นมาใหม่ โดยที่ไม่ต้องเพิ่มขาของไอซีแต่อย่างใด ดังรูปที่ 2.22 ที่เราเอาสแกนพัลซ์ของขา 9 ไปรวมกับขา 10หรือเอาขา 8 ไปรวมกับขา 10 สร้างสแกนพัลซ์ใหม่ขึ้นมาเป็นเมตริกซ์ในการใช้งานต่อไป

ระบบควบคุมจะทำการควบคุมให้ข้อมูลต่างๆ เป็นไปอย่างถูกต้อง มีการตรวจสอบคีย์กดหากเกิดดีเบ๊าส์ หรืออื่นๆ ก่อนส่งข้อมูล 6 บิต รวมทั้งดีไวส์จำนวน 5 บิต ผนวกกับเฮดพัลซ์ 1 ไซเกิล เข้าสู่ระบบการค้างและเลื่อนข้อมูล เราจะเอาข้อมูลหลักๆ ทั้งหมดเข้าสู่ระบบคาต้าสตรีม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อให้เกิดห้วงเวลาคงที่เมื่อส่งข้อมูลไปยังเครื่องรับแล้วไม่ต้องมีการซิงโครไนซ์ระหว่างเครื่องส่งและเครื่องรับเกิดการเรียงลำดับข้อมูลและอินเวิร์ตข้อมูล ก่อนจะส่งดาต้าสตรีมเข้าสู่วงจรมอดูเลชั่นผสมดาต้าสตรีมนี้เข้ากับความถี่ 35 กิโลเฮิร์ตซ์ ป้องกันสัญญาณรบกวนที่จะเกิดขึ้นในเวลาที่ส่งแสงแอลอีดีอินฟราเรดออกจากตัวส่งไปยังตัวรับซึ่งต้องใช้ระยะทางไกลๆ

ข้อมูลทั้งหมดนี้จะถูกขับออกมายังขา 6 ก่อนที่จะส่งไปขยายด้วยทรานซิสเตอร์เพื่อให้ส่งได้ไกล ทรานซิสเตอร์ที่ใช้อาจจะใช้ตัวเดียวหรือ 2 ตัวก็ได้ ขึ้นอยู่กับว่าเราจะให้ส่งได้ไกลแค่ไหน หากใช้หลายตัวการสิ้นเปลืองกระแสไฟฟ้าของถ่านก็จะสูงขึ้นด้วย ดังนั้นโดยปกติหากเราไม่กดคีย์ที่เอาต์พุต คือขา 6 ของไอซีจะไม่มีการจ่ายไบอัสให้ทรานซิสเตอร์ จึงไม่มีการสิ้นเปลืองกระแส รวมถึงระบบบอนดิ้งออปชั่นซึ่งมีสภาพเหมือนเครื่องคิดเลขที่จะทำการหยุดนำกระแสเองหากเราหยุดกดคีย์

ขา 11 เรายังไม่ได้เอาไปใช้เนื่องจากวงจรนี้เป็นของเครื่องรับโทรทัศน์ เราไม่นำเอาขานี้ไปใช้เนื่องจากเราต้องการดีไวส์โค้ดเป็น 000000 หากเป็นเครื่องเล่นวิดีโอเทปเขาจะเอาตัวนี้ไปสร้างดีไวส์โค้ด 111111 นั่นหมายความว่าเราใช้งานหมดทุกขา ส่วนขา 7 มักจะไม่ค่อยใช้งาน เพราะไม่มีความจำเป็นเนื่องจากว่าหากตัวเครื่องรับรีโมทอยู่ในสภาพปกติจะมีตัวสั่ง "คอลล์" ให้แอลอีดีของระบบรีโมทในเครื่องรับติดโชว์ขึ้น โดยที่ตัวส่งไม่จำเป็นต้องโชว์ เพราะเฮดพัลซ์จะออกไปทุกครั้งที่มีการกดคีย์ และการกดอย่างต่อเนื่องเฮดพัลซ์ก็จะออกไปอย่างต่อเนื่องเช่นเดียวกัน หากสงสัยว่าระบบการส่งยังใช้งานได้หรือไม่ก็สามารถทำได้โดยการต่อแอลอีดีเข้าไปที่ขานี้ เทียบกับกราวด์ เมื่อกดคีย์แมตริกซ์แอลอีดีต้องกระพริบโชว์ จึงจะถือว่าปกติ

2.4.2 รหัสสัญญาณ Remote Control ของเครื่องใช้ไฟฟ้าในปัจจุบัน

ในปัจจุบันเครื่องใช้ไฟฟ้าส่วนใหญ่ที่ใช้รีโมทคอนโทรล จะใช้ความถี่ในการส่งผ่านสัญญาณ(Carrier frequency) ที่ 40 KHZ ในที่นี้จะยกตัวอย่าง เครื่องใช้ไฟฟ้าที่ใช้กันแพร่หลาย 2 ยี่ห้อคือ Sony และ NEC สำหรับยี่ห้อ NEC จะผลิตชิปที่ใช้ในการส่งสัญญาณรีโมทคอนโทรลให้กับหลายยี่ห้อเช่น Pioneer ,Canon ,Goldstar ,Hitachi ,Kenwood และอีกหลายยี่ห้อ ดังนั้นรูปฟอร์มของสัญญาณก็จะมีมาตรฐานสัญญาณเดียวกับ NEC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

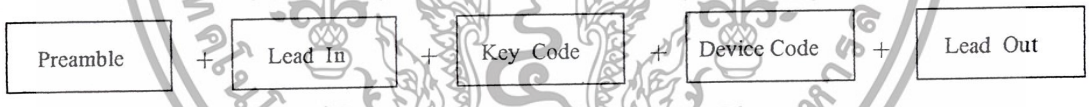
สำหรับ SONY มีคุณลักษณะทางเทคนิคดังนี้

Parameter	Decimal Value	HEX Value
Carrier Frequency	40 kHZ	
Unit of Brust Time	25 cycles of the carrier	
Lead In Brust	24	0060 0018
" 1 " Brust Pattern	24	0030 0018
" 0 " Brust Pattern	24	0018 0018
Lead Out	X , 1024	0018 03f6 or 0030 03f6

ตารางที่ 2.6 คุณลักษณะทางเทคนิคของ SONY

จากคุณลักษณะนี้ สัญญาณ "1" และ "0" จะใช้ T_{on} เป็น 22 เท่ากัน ส่วนค่า T_{off} ถ้าเป็น "1" ต้อง 96 เท่านั้น มากหรือน้อยกว่าถือว่าเป็น "0"

โครงสร้างสัญญาณ



Lead Out จะอยู่ตอนท้ายของสัญญาณเพื่อให้ภาครับทราบว่าสัญญาณจบการส่งแล้วต่อหนึ่งรอบของสัญญาณ 7 บิตหลังจาก Lecad In จะบอกถึงปุ่มที่กดบนแป้นรีโมท เมื่อทราบปุ่มก็จะทราบฟังก์ชั่นการทำงานของปุ่มนั้นว่าจะให้เครื่องใช้ทำอะไร ส่วน Device Code คือตัวบอกว่าเป็นเครื่องอะไร โดยทั่วไป Device Code จะมีจำนวนบิตเป็น 5 บิต 8 บิต หรือ 13 บิต ตามยุคสมัยต่อไปนี้เป็นตัวอย่างของสัญญาณรีโมทคอนโทรล SONY DVD S7000 มีลักษณะเป็นเลขฐาน 16 ดังต่อไปนี้

0000 0067 0000 0015 0060 0018 0018 0018 0030 0018 0030
0018 0030 0018 0018 0018 0030 0018 0018 0018 0018 0018
0030 0018 0018 0018 0030 0018 0030 0018 0030 0018 0018
0018 0018 0018 0030 0018 0018 0018 0018 0018 0030 0018
0018 03f6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในแต่ละ Word มีความหมายดังนี้

Preamble (4 word) 0000 0067 0000 0015

Word 1 เป็น 0 เสมอให้ภาครับ ทราบถึงสัญญาณรีโมทที่จะตามมา

Word 2 เท่ากับ 103 ในเลขฐานสิบ จะบอกถึงความถี่ที่ใช้ Carrier มีสูตรเป็น

$$\text{Frequency} = 1 \times 10^{6} / (N \times 0.241246)$$

; N = เลขฐาน 10 ใน Word 2

เมื่อแทนลงในสูตรจะได้ $40244 \approx 40$ kHZ นั่นเอง

Word 3 คือOne Sequence ถ้ามีค่าใน word นี้จะบอกถึงจำนวนบิตที่ส่งออกไปให้ภาครับ แต่จะส่งสัญญาณเพียงครั้งเดียวไม่มีการวนสัญญาณซ้ำ ในกรณีนี้เท่ากับ 0 จึงไม่ใช้วิธีการนี้

Word 4 คือ Repeat Sequence ถ้ามีค่าใน word นี้จะบอกถึงจำนวนบิตที่ส่งให้ภาครับเช่นกัน แต่การส่งจะส่งสัญญาณวนซ้ำไปเรื่อยๆจนกว่าจะปล่อยปุ่ม ในที่นี้มีค่าเป็น 21 บิต การส่งสัญญาณของการวนซ้ำในส่วนของ 21 บิตนี้คือ

LEAD IN (1) + KEY CODE (7) + DEVICE CODE (12) + LEAD OUT (1) = 21 #

Word 5 , 6 $(0060\ 0018)_{16} = (96\ 24)_{10}$ คือ LEAD IN ให้เครื่องทราบถึงส่วนของสัญญาณควบคุมที่จะตามมา

Word 7 ,8 $(0018\ 0018)_{16} = (24\ 24)_{10}$ → KEY CODE No.1 = " 0 "

Word 9 ,10 $(0030\ 0018)_{16} = (48\ 24)_{10}$ → KEY CODE No.2 = " 1 "

Word 11 ,12 $(0030\ 0018)_{16} = (48\ 24)_{10}$ → KEY CODE No.3 = " 1 "

Word 13 ,14 $(0030\ 0018)_{16} = (48\ 24)_{10}$ → KEY CODE No.4 = " 1 "

Word 15 ,16 $(0018\ 0018)_{16} = (24\ 24)_{10}$ → KEY CODE No.5 = " 0 "

Word 17 ,18 $(0030\ 0018)_{16} = (48\ 24)_{10}$ → KEY CODE No.6 = " 1 "

Word 19 ,20 $(0018\ 0018)_{16} = (24\ 24)_{10}$ → KEY CODE No.7 = " 0 "

จากข้อมูลด้านบนถอดรหัสได้เป็น 0111010 เรียงรหัสใหม่จาก No.7 ไปหา No.1 จะได้เลข Binary เป็น $(0101110)_2 = (46)_{10}$

ต่อจาก Key Code จะเป็น Device Code

Word 21 ,22 $(0018\ 0018)_{16} = (24\ 24)_{10}$ → DEVICE CODE No.1 = " 0 "

Word 23 ,24 $(0030\ 0018)_{16} = (48\ 24)_{10}$ → DEVICE CODE No.2 = " 1 "

Word 25 ,26 $(0018\ 0018)_{16} = (24\ 24)_{10}$ → DEVICE CODE No.3 = " 0 "

Word 27 ,28 $(0030\ 0018)_{16} = (48\ 24)_{10}$ → DEVICE CODE No.4 = " 1 "

Word 29 ,30 $(0030\ 0018)_{16} = (48\ 24)_{10}$ → DEVICE CODE No.5 = " 1 "

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Word 31 ,32 $(0030\ 0018)_{16} = (48\ 24)_{10}$ → DEVICE CODE No.6 = " 1 "

Word 33 ,34 $(0018\ 0018)_{16} = (24\ 24)_{10}$ → DEVICE CODE No.7 = " 0 "

Word 35 ,36 $(0018\ 0018)_{16} = (24\ 24)_{10}$ → DEVICE CODE No.8 = " 0 "

Word 37 ,38 $(0030\ 0018)_{16} = (48\ 24)_{10}$ → DEVICE CODE No.9 = " 1 "

Word 39 ,40 $(0018\ 0018)_{16} = (24\ 24)_{10}$ → DEVICE CODE No.10 = " 0 "

Word 41 ,42 $(0018\ 0018)_{16} = (24\ 24)_{10}$ → DEVICE CODE No.11 = " 0 "

Word 43 ,44 $(0030\ 0018)_{16} = (48\ 24)_{10}$ → DEVICE CODE No.12 = " 1 "

Word 45 ,46 $(0018\ 0018)_{16} = (24\ 24)_{10}$ → LEAD OUT = " 0 "

จากข้อมูลด้านบนถอดรหัสได้เป็น 0101110010010 เรียงรหัสจาก LEAD OUT → No.12 → No.1 ได้เป็น $(0100100111010)_2 = (2362)_{10}$

เมื่อรวมรหัส KEY CODE และ DEVICE CODE ได้ความว่า SONY DVD S7000 มีรหัสประเภทอุปกรณ์เป็น 2362 และ ฟังก์ชั่นที่ 46 ถูกเรียกใช้ ซึ่งเป็นปุ่ม POWER ON

จากข้อมูลของDEVICE CODE จะเห็นว่า SONY ได้เอา LEAD OUT ไปรวมกับ DEVICE CODE ซึ่งจะเป็นเช่นนี้ไปจนกว่าจะถึงการวนรอบข้อมูลสุดท้าย LEAD OUT จึงถูกนำมาใช้เพื่อให้เครื่องทราบว่าจบข้อมูล

รีโมทคอนโทรลของ SONY มีภาครับที่มีประสิทธิภาพในการกรองสัญญาณรบกวนสูง ทั้งภาคส่งยังส่งได้แรง จึงแทบไม่จำเป็นจะต้องมีส่วนของการตรวจสอบสัญญาณเลย

สำหรับ NEC มีมาตรฐานการส่งสัญญาณดังนี้

Parameter	Decimal Value	HEX Value
Carrier Frequency	40 kHZ	
Unit of Brust Time	22 cycles of the carrier	
Lead In Brust	341 171	0156 00ab
" 1 " Brust Pattern	22 96	0016 0060
" 0 " Brust Pattern	22 24	0016 0016
Lead Out	22 127	0016 0593

ตารางที่ 2.7 คุณลักษณะทางเทคนิคของ NEC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NEC จะใช้ระบบ Pulse Width Modulation (PWM) ที่ทำให้ข้อมูลแต่ละการกดปุ่มมีค่าเวลาในการส่งเท่ากันทั้งหมด โดยหากเปรียบเทียบกับ SONY แล้ว NEC จะใช้จำนวนบิตในการส่งข้อมูลจำนวนมากกว่า สัญญาณของ NEC จะแบ่งเป็น 16 บิตสำหรับส่วนข้อมูล และ 16 บิต สำหรับการตรวจสอบความคลาดเคลื่อนของสัญญาณ ดังแสดงดังนี้

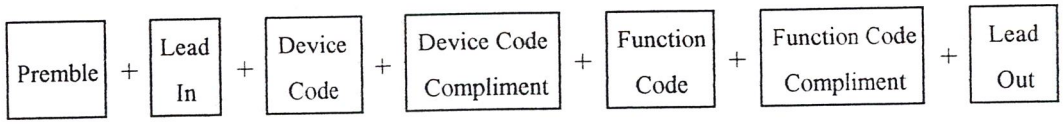
Device Code	Device Code Compliment	Function Code	Function Code Compliment

Code จะถูกแบ่งเป็น 4 ส่วน ,ส่วนละ 8 บิต รวมทั้งสิ้น 32 บิต ส่วนของ Device Code จะมีค่าตั้งแต่ 0 – 255 (2^8) จึงสามารถรองรับเครื่องใช้ได้ 256 ประเภท เช่นเดียวกับ Function Code จะได้ 256 คำสั่งการใช้งานเครื่องใช้ ในส่วนของค่ากลับของสัญญาณ (Compliment field) ใช้วิธีการทำ 1's compliment กับสัญญาณนั้น (เหตุผลการทำ 1's compliment อ่านค่าได้จาก ทฤษฎีการส่งสัญญาณรีโมทคอนโทรลในบทต้นๆ) NEC จะใช้ LEAD IN และ LEEAD OUT แยกอีกต่างหาก ดังนั้นจึงรวมเป็น 34 บิต ต่อการส่งสัญญาณ 1 ชุด

ตัวอย่างของสัญญาณต่อไปนี้คือ เครื่องใช้ไฟฟ้ายี่ห้อ Pioneer ซึ่งใช้มาตรฐานสัญญาณของ NEC ในการส่งข้อมูลชื่อรุ่นคือ CLD79 Elite Laser Disk Player มีค่าเลขฐาน 16 เป็น

0000 0067 0000 0022 0156 00ab 0016 0060 0016 0060 0016 0060
0016 0016 0016 0060 0016 0016 0016 0060 0016 0016 0016 0016
0016 0016 0016 0016 0016 0060 0016 0016 0016 0060 0016 0016
0016 0060 0016 0060 0016 0016 0016 0060 0016 0016 0016 0016
0016 0060 0016 0060 0016 0060 0016 0016 0016 0060 0016 0016
0016 0060 0016 0060 0016 0016 0016 0016 0016 0016 0016 0593

โครงสร้างสัญญาณ



วิธีการอ่านโค้ดเหมือนกับการอ่านโค้ดของ SONY จึงจะไม่อธิบายซ้ำอีก สรุปการอ่านโค้ดได้ว่า ใช้ความถี่สัญญาณพาหะที่ 40 kHZ มี 34 บิต ต่อการส่งสัญญาณ 1รอบใช้การส่งแบบวนสัญญาณทั้ง 34 บิต จนกว่าจะปล่อยปุ่มรีโมทคอนโทรล มีสัญญาณที่ถอดรหัสดิจิตอลได้เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

00010101 11101010 01011000 10100111 แบ่งตามฟอร์มมาตรฐานเป็น Device Code, Device Code Compliment, Function Code และFunction Code Compliment ตามลำดับ

Device Code ถอดรหัสได้เป็น 00010101 เรียงรหัสย้อนกลับได้เลขฐาน 2 เป็น 10101000 แปลงเป็นฐานสิบได้เป็น 168 #

Function Code ถอดรหัสได้เป็น 01011000 เรียงรหัสย้อนกลับได้เลขฐาน 2 เป็น 00011010 แปลงเป็นฐานสิบได้เป็น 26 #

เมื่อนำรหัสมาตีความจะได้เป็น Pioneer รุ่น CLD79 Elite Laser Disk Player ใช้ Device Code หมายเลข 168 ปุ่มที่กดเรียกใช้ฟังก์ชั่นที่ 26 มีความหมายให้ Power On

ส่วนของ Compliment มีไว้เพื่อตรวจความคลาดเคลื่อนของสัญญาณทั้ง 2 เวลารวมของ Code และ Compliment Code ต้องเท่ากัน

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

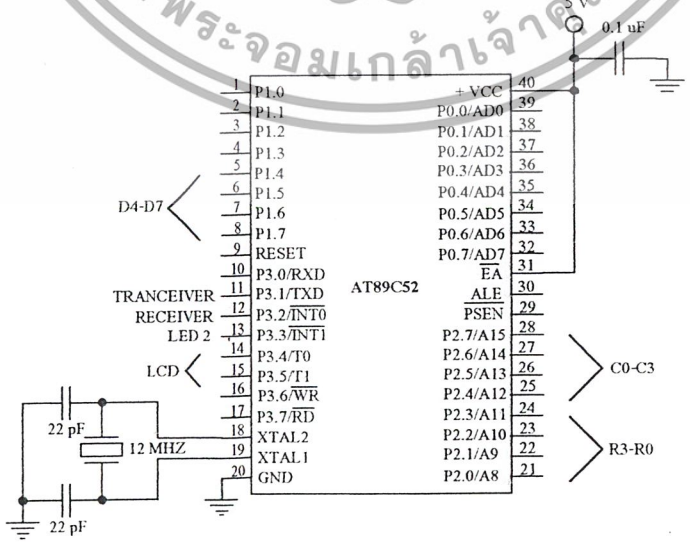
การออกแบบวงจร

การออกแบบวงจรของ เครื่องรับส่งสัญญาณด้วยรีโมทคอนโทรล ในโครงงานนี้แบ่งการออกแบบเป็น 2 ส่วน คือวงจรทางด้านฮาร์ดแวร์ และซอฟแวร์ควบคุมวงจร โดยวงจรทางด้านฮาร์ดแวร์นี้จะใช้ไมโครคอนโทรลเลอร์ เป็นหัวใจหลักในการประมวลผลสัญญาณอินฟราเรดรีโมทคอนโทรล และควบคุมส่วนของการแสดงผล ไมโครคอนโทรลเลอร์ตระกูล MCS-51 มีคุณสมบัติเหมาะสมเนื่องจากมีพอร์ตในตัวใช้งานง่าย และในโครงงานนี้เลือกใช้ไมโครคอนโทรลเลอร์เบอร์ AT89C52 มีหน่วยความจำ 8 Kbyte ซึ่งเพียงพอกับโครงงานนี้ทางด้านซอฟแวร์ควบคุมวงจรใช้ภาษาแอสแซมบลีของ MCS-51 ในการเขียนโปรแกรมเพื่อกำหนดการรับสัญญาณอินฟราเรด อ่านสัญญาณอินฟราเรดที่รับเข้ามาแสดงผลที่จอแอลซีดี ทดสอบค่าที่รับเข้ามาเพื่อความถูกต้องของสัญญาณอินฟราเรด

3.1 การออกแบบวงจรฮาร์ดแวร์

3.1.1 ส่วนประมวลผล

การออกแบบส่วนประมวลผลในโครงงานนี้ใช้ไมโครคอนโทรลเลอร์เบอร์ AT89C52 ของบริษัท ATMEL ซึ่งมีพอร์ตใช้งานและมีหน่วยความจำที่เพียงพอสำหรับโครงงานนี้ สามารถโปรแกรมซ้ำ ได้ 1000 ครั้ง และมีการต่อขาใช้งานต่าง ๆ ดังต่อไปนี้



รูปที่ 3.1 ขาต่อใช้งานของไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขา 5 ถึง ขา 8 คือ พอร์ต 1.4 ถึง พอร์ต 1.7 ต่อกับขา D4 ถึง D7 เพื่อส่งข้อมูลให้กับ DDRAM ของ LCD MODULE เพื่อใช้ในการส่งสัญญาณควบคุมการแสดงผล

ขา 9 เป็นขารีเซต (Reset) ของไมโครคอนโทรลเลอร์ เพื่อให้ไมโครคอนโทรลเลอร์ เริ่มต้นการทำงานของโปรแกรมใหม่ทั้งหมด

ขา 11 คือ พอร์ต 3.1 ใช้ในการควบคุมการส่งสัญญาณอินฟราเรดรีโมทคอนโทรลที่อ่านได้นั้นออกมาเพื่อทดสอบว่าค่าที่อ่านได้นั้นถูกต้องหรือไม่

ขา 12 คือ พอร์ต 3.2 ทำหน้าที่รับสัญญาณอินฟราเรด ที่ส่งมาจากตัวรับสัญญาณอินฟราเรด เพื่อให้ไมโครคอนโทรลเลอร์ประมวลผล

ขา 13 คือ พอร์ต 3.3 ต่อ LED เพื่อแสดงผลการรับและทดสอบสัญญาณรีโมทคอนโทรล

ขา 14 คือ พอร์ต 3.4 ต่อกับขาอีนาเบิ้ล (ENABLE) ของ LCD MODULE เพื่อใช้ในการกำหนดสภาพการรับเขียนอ่านข้อมูล

ขา 15 คือ พอร์ต 3.5 ต่อ RS (REGISTER SELECTION) ของ LCD MODULE เพื่อใช้ในการเซตการทำงานของ LCD MODULE

ขา 18 กับ 19 ต่อคริสตอลเพื่อเป็นฐานเวลาให้กับไมโครคอนโทรลเลอร์ โดยอาศัยความถี่ของคริสตอลเป็นตัวกำหนดฐานเวลา และฐานเวลานี้เองที่เป็นตัวบอกให้ทราบว่าในคำสั่งหนึ่งคำสั่งใช้เวลาประมวลผลเท่าไร

ขา 20 เป็นขากราวด์ของไมโครคอนโทรลเลอร์

ขา 21 ถึง 28 คือ พอร์ต 2.0 ถึง พอร์ต 2.7 ต่อกับคีย์แพดเพื่อนำสัญญาณที่เกิดจากการกดคีย์ไปประมวลผลว่าได้ทำการกดคีย์แพดที่แถวใดและหลักใด (เป็นการสแกนคีย์นั่นเอง) เพื่อที่ได้ทราบว่าเรากดปุ่มใดไปจากนั้นเมื่อรู้ว่ากดปุ่มใดแล้ว ไมโครคอนโทรลเลอร์ก็จะทำการประมวลผลว่าปุ่มนั้นมีคำสั่งให้ทำอะไรต่อไป

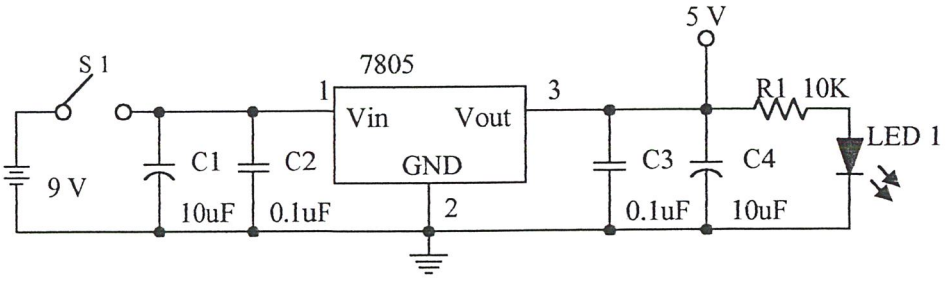
ขา 31 เป็นขาอีนาเบิ้ลของไมโครคอนโทรลเลอร์ เนื่องจากขานี้ไม่ใช้จึงต่อไว้กับไฟ 5 โวลท์ ทั้งนี้ขาอีนาเบิ้ลเป็นแอคทีฟโลว์ (Active Low)

ขา 40 ต่อกับแรงดันไฟ 5 โวลท์

3.1.2 ภาคจ่ายไฟ

ในโครงงานนี้ใช้ไอซีเร็กกูเลต (REGURATE) เบอร์ 7805 เพื่อแปลงแรงดันไฟจาก 9 โวลท์ ให้เหลือแรงดัน 5 โวลท์ และป้อนให้กับวงจรในส่วนต่างๆ เช่น ไมโครคอนโทรลเลอร์ จอแอลซีดี และตัวรับสัญญาณรีโมทคอนโทรล

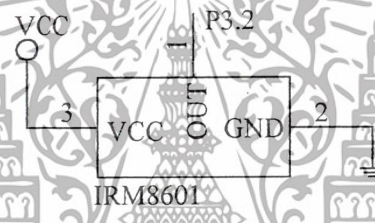
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 วงจรภาคจ่ายไฟ

3.1.3 ส่วนของวงจรรับสัญญาณอินฟราเรดรีโมทคอนโทรล

ตัวรับสัญญาณอินฟราเรดรีโมทคอนโทรล ทำหน้าที่รับสัญญาณรีโมทคอนโทรล และส่งไปยังพอร์ต 3.2 ของไมโครคอนโทรลเลอร์ เพื่อทำการประมวลผล



รูปที่ 3.3 ส่วนรับสัญญาณอินฟราเรด

3.1.4 ส่วนของวงจรส่งสัญญาณอินฟราเรดรีโมทคอนโทรล

วงจรส่งสัญญาณอินฟราเรดรีโมทคอนโทรลใช้ทรานซิสเตอร์ ทำหน้าที่เหมือนสวิตซ์เพื่อเปิด-ปิดวงจรทำให้ LED อินฟราเรดส่งสัญญาณอินฟราเรดออกมาตามค่าที่อ่านได้ ค่าสัญญาณที่อ่านได้จะเก็บไว้ใน RAM (ซึ่งสัญญาณถูกถอดคลื่นพาหะแล้ว) เมื่อนำมาสร้างใหม่ที่ส่วนวงจรส่งสัญญาณอินฟราเรดรีโมทคอนโทรล ซึ่งความละเอียดของสัญญาณใหม่นี้อาจจะไม่เท่ากับสัญญาณต้นแบบ

การคำนวณ

<u>จากสูตร</u>

$$f = \frac{1}{T} = \frac{1.44}{(RA + 2RB)C} \qquad (3.1)$$

พิจารณามาจากรูปวงจรของ Pulse Position Modulation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยโครงงานนี้ต้องการความถี่ 38 kHZ จึงมาคำนวณเพื่อหาค่า RA , RB และค่า C มีการคำนวณดังนี้

สมมุติให้

ค่า RB มีค่า 2.2 kohm

ค่า C มีค่า 4.7 nF

เพื่อหาค่า RA จากสูตร $f = \frac{1}{T} = \frac{1.44}{(RA + 2RB)C}$ แทนค่า RB , C ลงในสมการ (3.1)

$$38\text{ k} = \frac{1.44}{(RA + 2(2.2k)4.7n}$$

$$(\text{RA}+4.4\text{ k}) = \frac{1.44}{38k \times 4.7n}$$

$$\text{RA}+4.4\text{ k} = \frac{1.44}{0.1786m}$$

$$\text{RA} = 8.06\text{ k} - 4.4\text{ k}$$

$$\text{RA} \approx 3.6\text{ k}$$

จะได้ RA ≈ 3.6 kohm

เมื่อ RA มีค่าประมาณ 3.6 kohm จึงเลือกใช้ค่า RA ดังนี้

เลือกค่า RA คงที่ 470 ohm และ RA ปรับค่า 5 kohm ดังนั้นความถี่ที่ได้มีค่าเป็นช่วง

ที่ RA มีค่าเท่ากับ 470 ohm จะได้ความถี่เท่ากับ

$$f = \frac{1}{T} = \frac{1.44}{(470 + 2(2.2k)4.7n}$$

$$= 62.9\text{ kHZ}$$

ที่ RA มีค่าเท่ากับ 5470 ohm จะได้ความถี่เท่ากับ

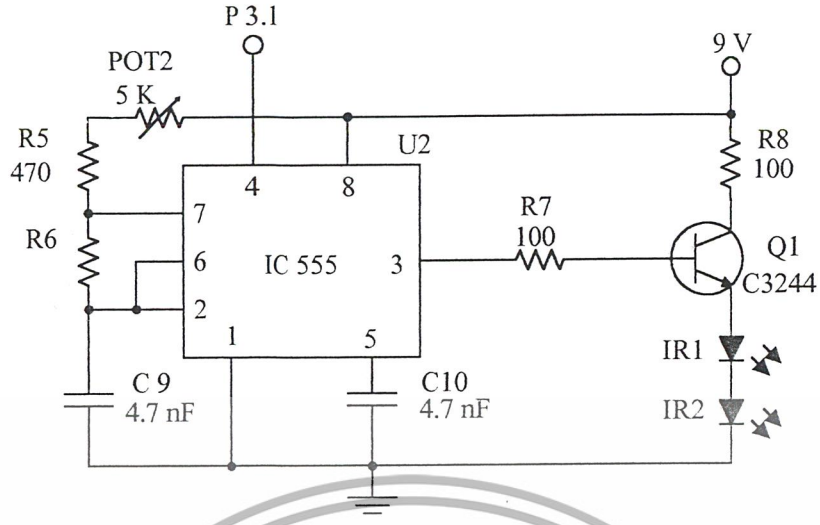
$$f = \frac{1}{T} = \frac{1.44}{(5470 + 2(2.2k)4.7n}$$

$$= 31\text{ kHZ}$$

ดังนั้นความถี่ที่ได้ก็จะอยู่ในช่วง 31 – 62.9 kHZ

นำค่า RA , RB และ C มาเขียนวงจรได้ดังรูป

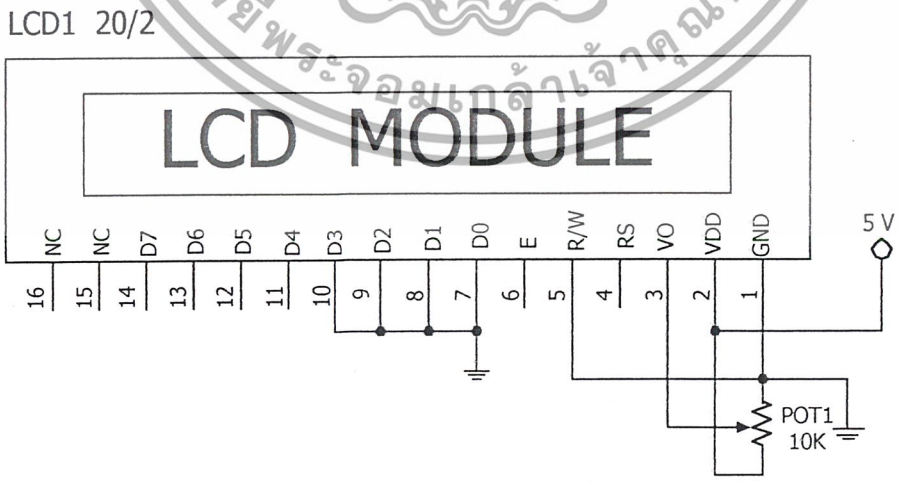
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 วงจรควบคุมการส่งสัญญาณอินฟราเรดรีโมทคอนโทรล

3.1.5 ส่วนแสดงผล

ส่วนแสดงผลใช้ DOT MATRIX LCD MODULE ขนาด 20 ตัวอักษร 2 บรรทัด เพื่อใช้แสดงผล ซึ่งภายในตัว LCD MODULE ประกอบด้วย HD44780 ซึ่งเป็นคอนโทรลเลอร์ทำหน้าที่ควบคุมการแสดงผล รูปแบบตัวอักษร หรือ สัญลักษณ์ต่างๆ โดยที่การต่อ LCD MODULE เราสามารถติดต่อกับ LCD MODULE แบบ 4 บิต หรือ แบบ 8 บิต ก็ได้ ซึ่งในโครงงานนี้จะใช้การติดต่อแบบ 4 บิต คือ D4-D7 เพื่อประหยัดพอร์ตในการใช้งานของไมโครคอนโทรลเลอร์ ส่วนบิตที่เหลือคือ D0-D3 ต่อลงกราวด์ การต่อวงจรแสดงดังรูปที่ 3.5

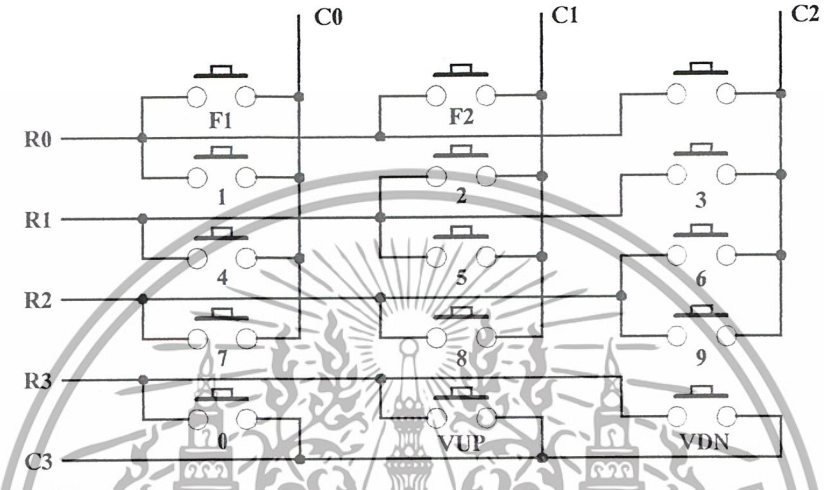


รูปที่ 3.5 การต่อวงจรเพื่อควบคุมจอแอลซีดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.6 ส่วนของคีย์เมตริกซ์

ในส่วนของคีย์เมตริกซ์นี้จะใช้ขนาด 5×3 ซึ่งต่อกับพอร์ต 2.0-2.7 ของไมโครคอนโทรลเลอร์ โดยคีย์เมตริกซ์จะเป็นอินพุตให้ไมโครคอนโทรลเลอร์ทำงานตามฟังก์ชั่นต่างๆของคีย์แต่ละคีย์ทั้งนี้ จะต้องจะต้องทำการประมวลผลในส่วนของสแกนคีย์ก่อนจึงจะทำงานตามฟังก์ชั่นต่างๆได้



รูปที่ 3.6 วงจรของคีย์เมตริกซ์

3.2 การออกแบบซอฟแวร์ควบคุมวงจร

การออกแบบจะคำนึงถึงฮาร์ดแวร์แต่ละชิ้นเป็นหลักซึ่งมี LCD MODULE KEY METRIX ภาครับ ภาคส่ง สัญญาณอินฟราเรด ดังนั้น การออกแบบจะแบ่งออกเป็น 4 ส่วนหลักๆคือ

3.2.1 การออกแบบซอฟแวร์ส่วนแสดงผล

3.2.2 การออกแบบซอฟแวร์ส่วนควบคุมของคีย์แพด

3.2.3 การออกแบบซอฟแวร์ส่วนรับสัญญาณอินฟราเรด

3.24 การออกแบบซอฟแวร์ส่วนทดสอบค่าสัญญาณอินฟราเรด

การออกแบบซอฟแวร์แต่ละส่วนนั้น สามารถเขียนเป็นโฟลวชาร์ตการทำงานได้ดังนี้

3.2.1 การออกแบบซอฟแวร์ส่วนแสดงผล

การออกแบบซอฟแวร์ส่วนนี้เป็นส่วนที่เกี่ยวกับ การแสดงผลอักษรวิ่งชื่อโปรแกรม การแสดงผลฐานเวลา การแสดงผลเลือกฟังก์ชั่นการทำงานหน้าที่ต่างๆ ซึ่งโปรแกรมกำหนดให้อยู่ในภาคผนวก ข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การแสดงผลที่หน้าจอ แอลซีดีนั้นมีอยู่ 2 แบบคือ การแสดงผลโดยแสดงตัวอักษรข้อความธรรมดา และการแสดงผลรหัสสัญญาณ ซึ่งการแสดงผลรหัสสัญญาณต้องนำค่าที่เก็บไว้ในแรมมาแสดงผล การออกแบบซอฟแวร์สามารถดูได้จากโฟลวชาร์ตการทำงานในบทนี้

3.2.2 การออกแบบซอฟแวร์ส่วนควบคุมของคีย์แพด

ในส่วนนี้จะใช้โปรแกรมการสแกนคีย์ ซึ่งมีข้อมูลอ้างอิงสืบค้นได้จากหนังสืออ้างอิงท้ายเล่ม สรุปการทำงานของคีย์แพดอย่างย่อคือ สมมุติว่ากดปุ่ม " F1" จะมีค่าซึ่งผ่านโปรแกรมดังกล่าวเก็บอยู่ใน KEYPAD_DATA แล้วนำค่าดังกล่าวไปเปรียบว่าจะใช้งานปุ่ม " F1" ในขั้นตอนใดของโปรแกรม โฟลวชาร์ตของโปรแกรมย่อย SCAN KEY สามารถศึกษาได้ในบทนี้

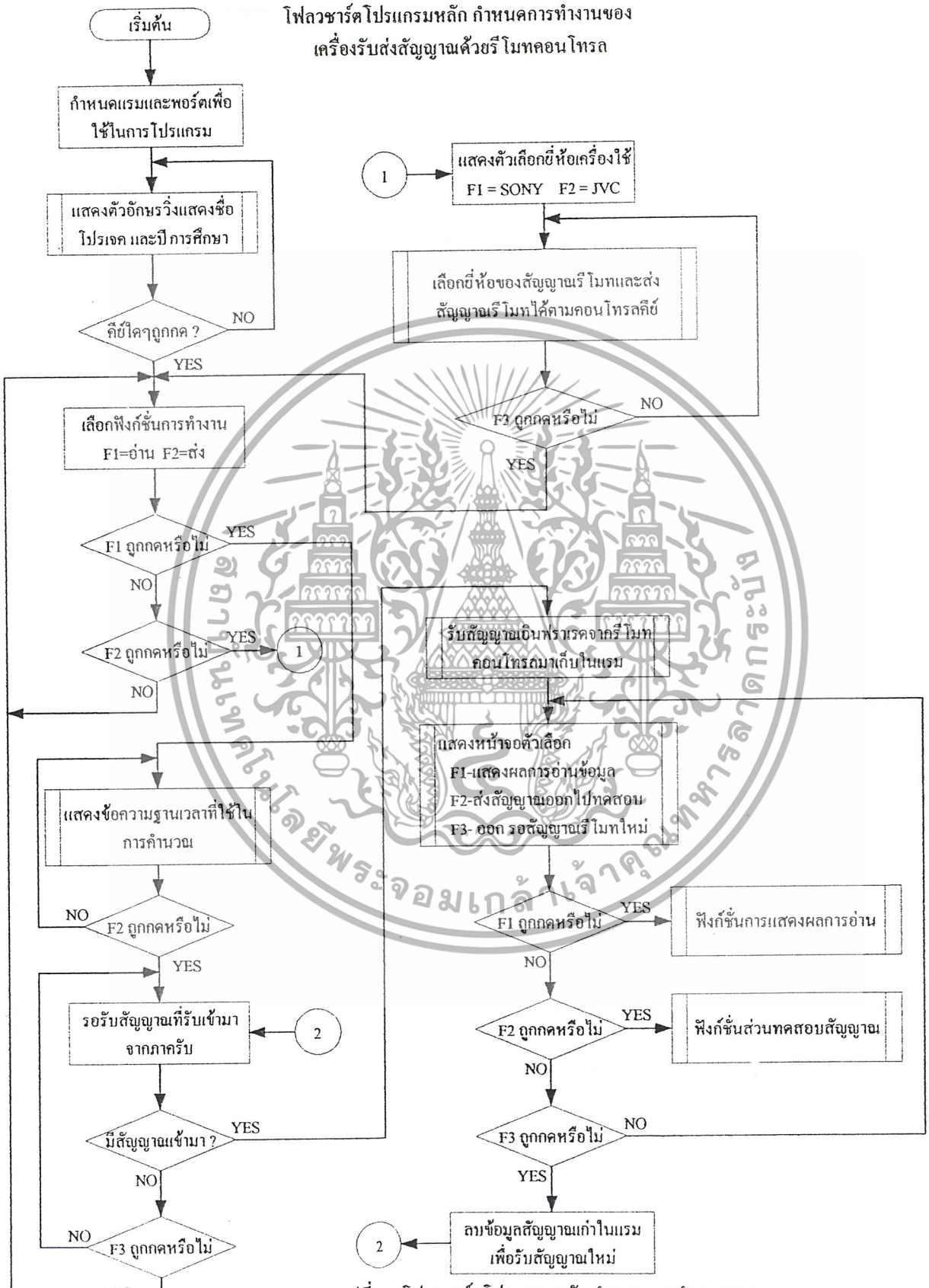
3.2.3 การออกแบบซอฟแวร์ส่วนรับสัญญาณอินฟราเรด

การออกแบบซอฟแวร์ส่วนรับสัญญาณอินฟราเรดจากรีโมทคอนโทรลนั้น เมื่อเริ่มมีข้อมูลเข้ามาจะทำการเซ็ตไทเมอร์ แล้วสถานะของอินพุตเปลี่ยนไปให้นำค่าไทเมอร์นั้นไปเก็บไว้ในแรมและเริ่มเซ็ตไทเมอร์ใหม่ การออกแบบซอฟแวร์ส่วนรับสัญญาณอินฟราเรดจากรีโมทคอนโทรลสามารถดูได้จาก โปรแกรมย่อยส่วนรับสัญญาณอินฟราเรดในบทนี้

3.2.4 การออกแบบซอฟแวร์ส่วนทดสอบค่าสัญญาณอินฟราเรด

การออกแบบซอฟแวร์ส่วนทดสอบค่าสัญญาณอินฟราเรดต้องนำค่าสัญญาณที่เก็บไว้ในแรมสร้างเป็นรูปคลื่นพัลซ์ส่งออกมา แล้วมอดูเลทรวมกับความถี่ที่ภาคส่งสร้างขึ้น ซึ่งใช้ความถี่ 40 KHZ เป็นความถี่พาหะ เนื่องจากเครื่องใช้ไฟฟ้าส่วนใหญ่ที่ใช้อินฟราเรดรีโมทคอนโทรลในการควบคุมนิยมใช้ความถี่นี้ในการส่งสัญญาณควบคุม การส่งสัญญาณที่มีลอจิก " 1 " โปรแกรมจะทำการเซ็ตพอร์ตและเคลียร์พอร์ต 3.1 แล้วหน่วงเวลาที่ส่งนานตามค่าเวลาที่เก็บไว้ในแรม ส่วนสัญญาณที่เป็นลอจิก "0" โปรแกรมจะทำในลักษณะเดียวกัน การออกแบบซอฟแวร์ส่วนทดสอบค่าสัญญาณอินฟราเรดที่รับเข้ามาสามารถดูได้จากโฟลวชาร์ตโปรแกรมย่อยการส่งสัญญาณที่รับเข้ามาเพื่อทดสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 โฟลวชาร์ตโปรแกรมหลัก กำหนดการทำงานของ
เครื่องรับส่งสัญญาณด้วยรีโมทคอนโทรล

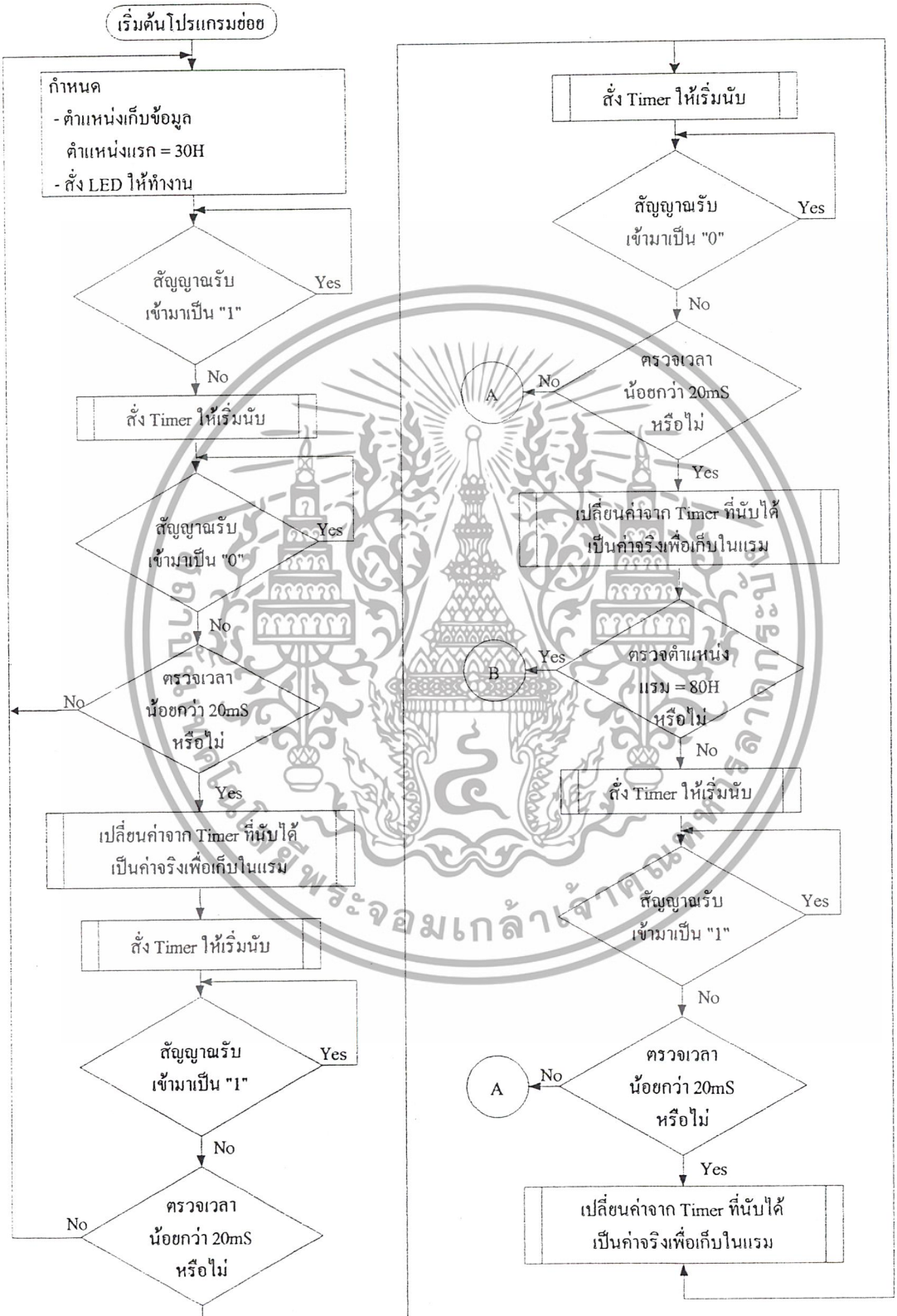
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โฟลวชาร์ตโปรแกรมย่อยแสดงรหัสสัญญาณอินฟราเรดที่รับเข้ามา

รูปที่ 3.8 โฟลวชาร์ตโปรแกรมย่อยแสดงรหัสสัญญาณอินฟราเรดที่รับเข้ามา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

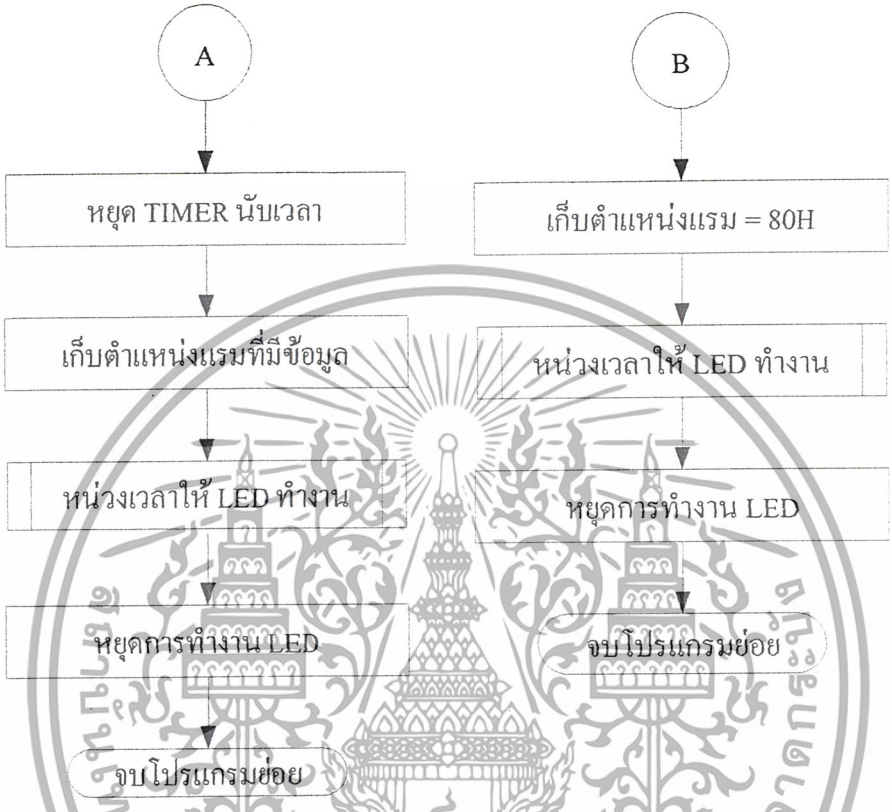
โฟลวชาร์ตย่อยส่วนรับสัญญาณอินฟราเรด



รูปที่ 3.9 โฟลวชาร์ตโปรแกรมย่อย
ส่วนรับอินฟราเรด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โฟลวชาร์ตส่วนย่อยการรับสัญญาณอินฟราเรด (ต่อ)



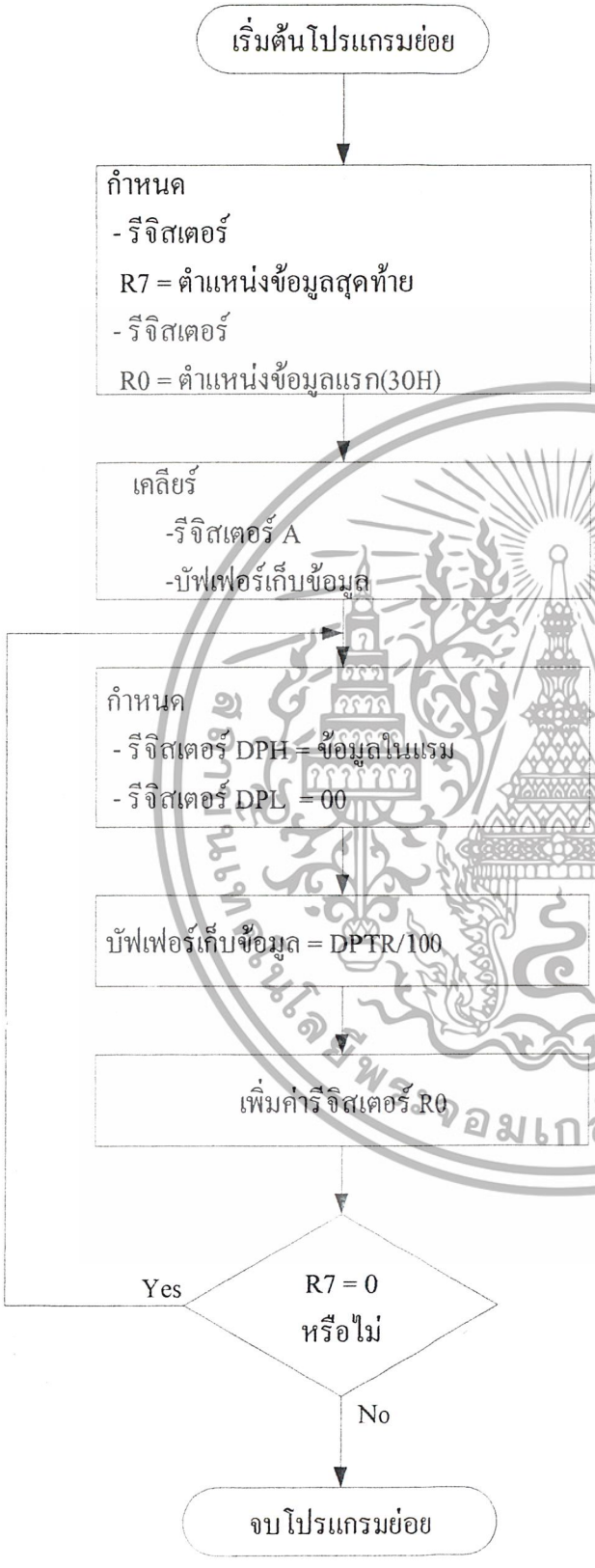
รูปที่ 3.9(ต่อ) โฟลวชาร์ตส่วนย่อยการรับสัญญาณอินฟราเรด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.10 โฟลชาร์ตโปรแกรมย่อยส่วนทดสอบสัญญาณ

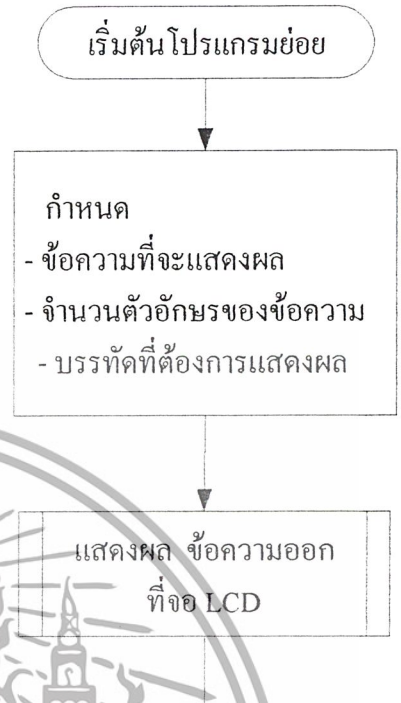
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โฟลวชาร์ตโปรแกรมย่อยสุ่มเก็บ
สัญญาณด้วยฐานเวลา 100 uS



รูปที่ 3.11 โฟลวชาร์ตโปรแกรมย่อยสุ่มเก็บ
สัญญาณด้วยฐานเวลา 100 uS

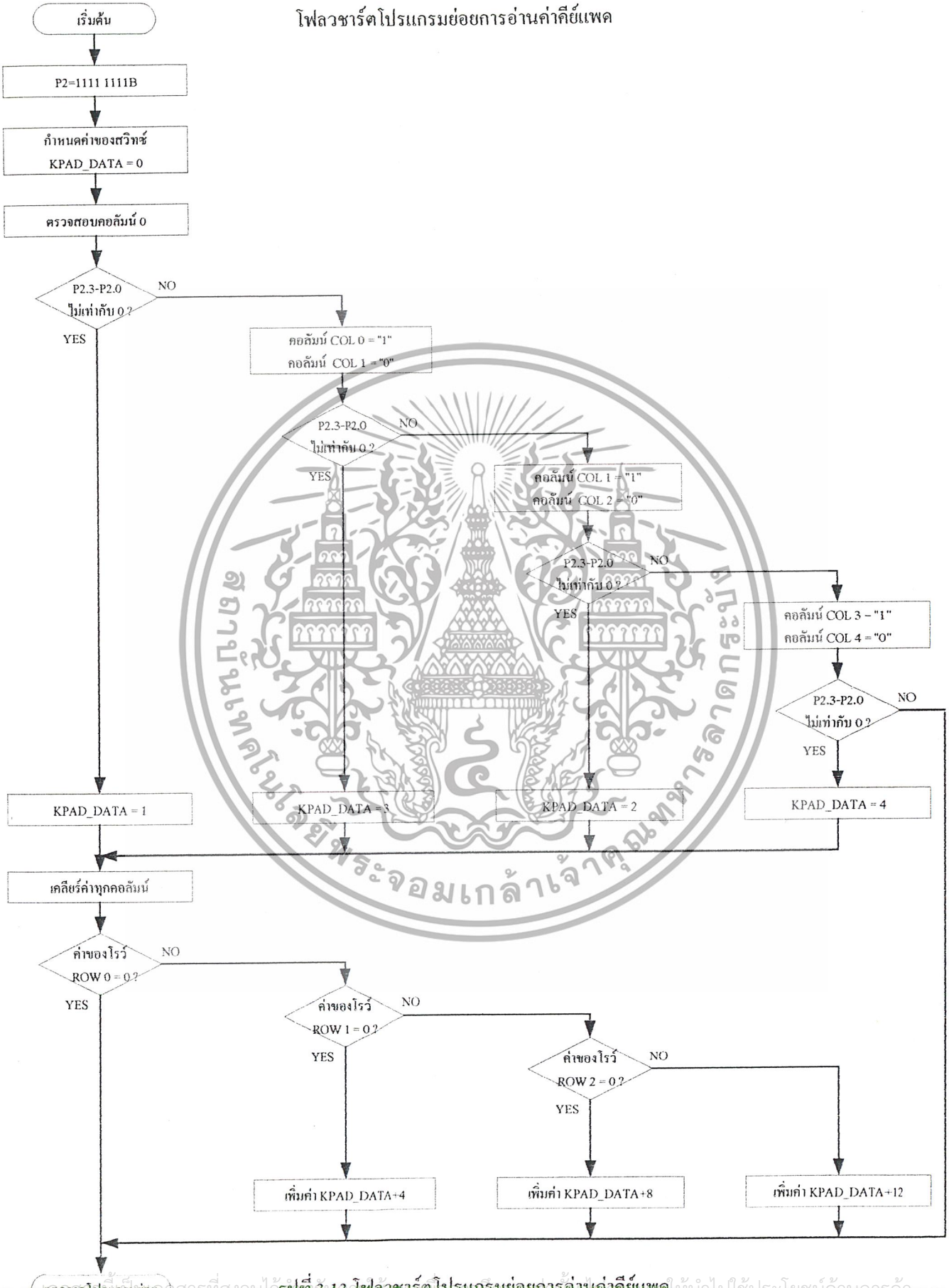
โฟลวชาร์ตโปรแกรมแสดงผลข้อความ



รูปที่ 3.12 โฟลวชาร์ตแสดงผลข้อความ

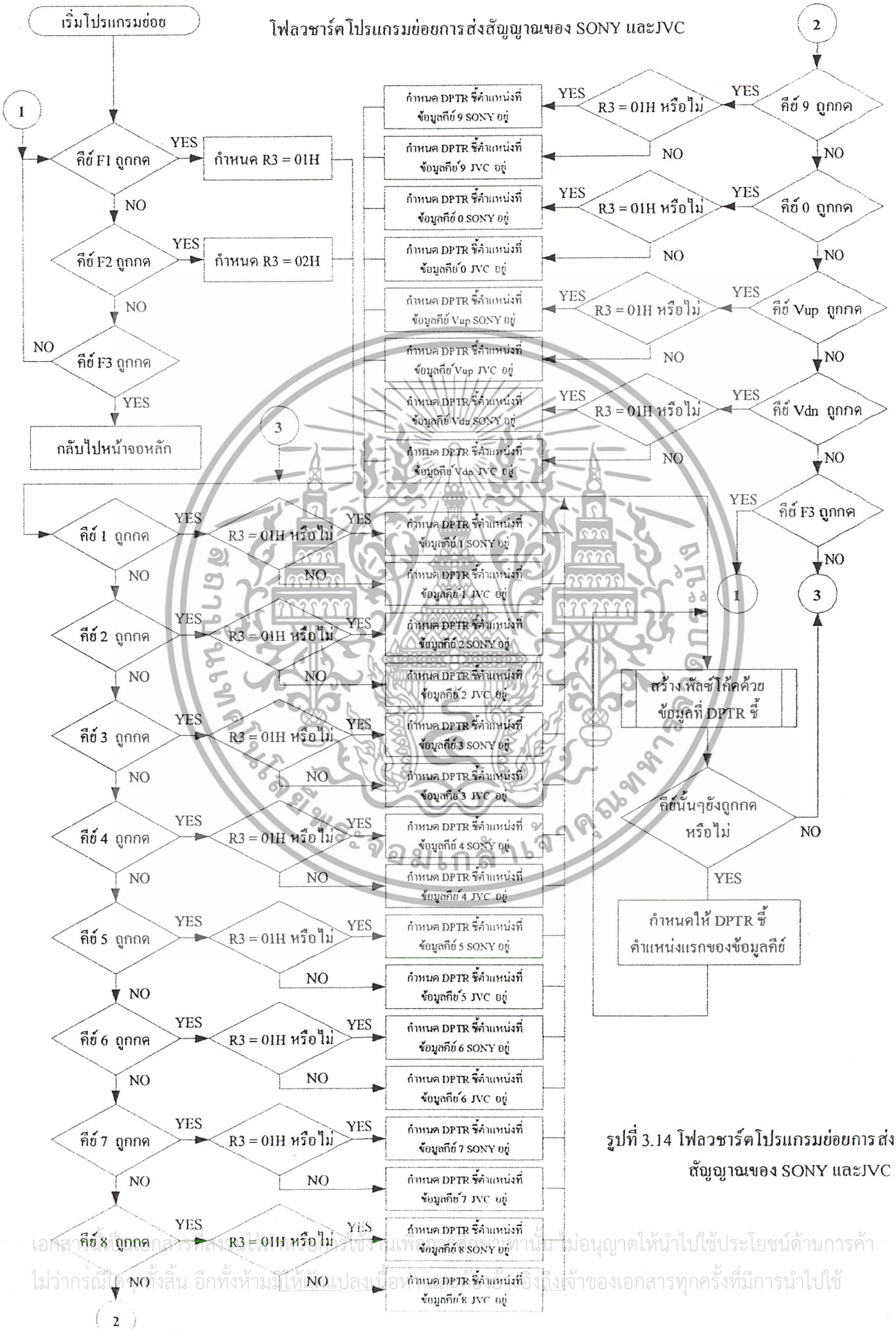
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โฟลวชาร์ตโปรแกรมย่อยการอ่านค่าคีย์แพด



รูปที่ 3.13 โฟลวชาร์ตโปรแกรมย่อยการอ่านค่าคีย์แพด

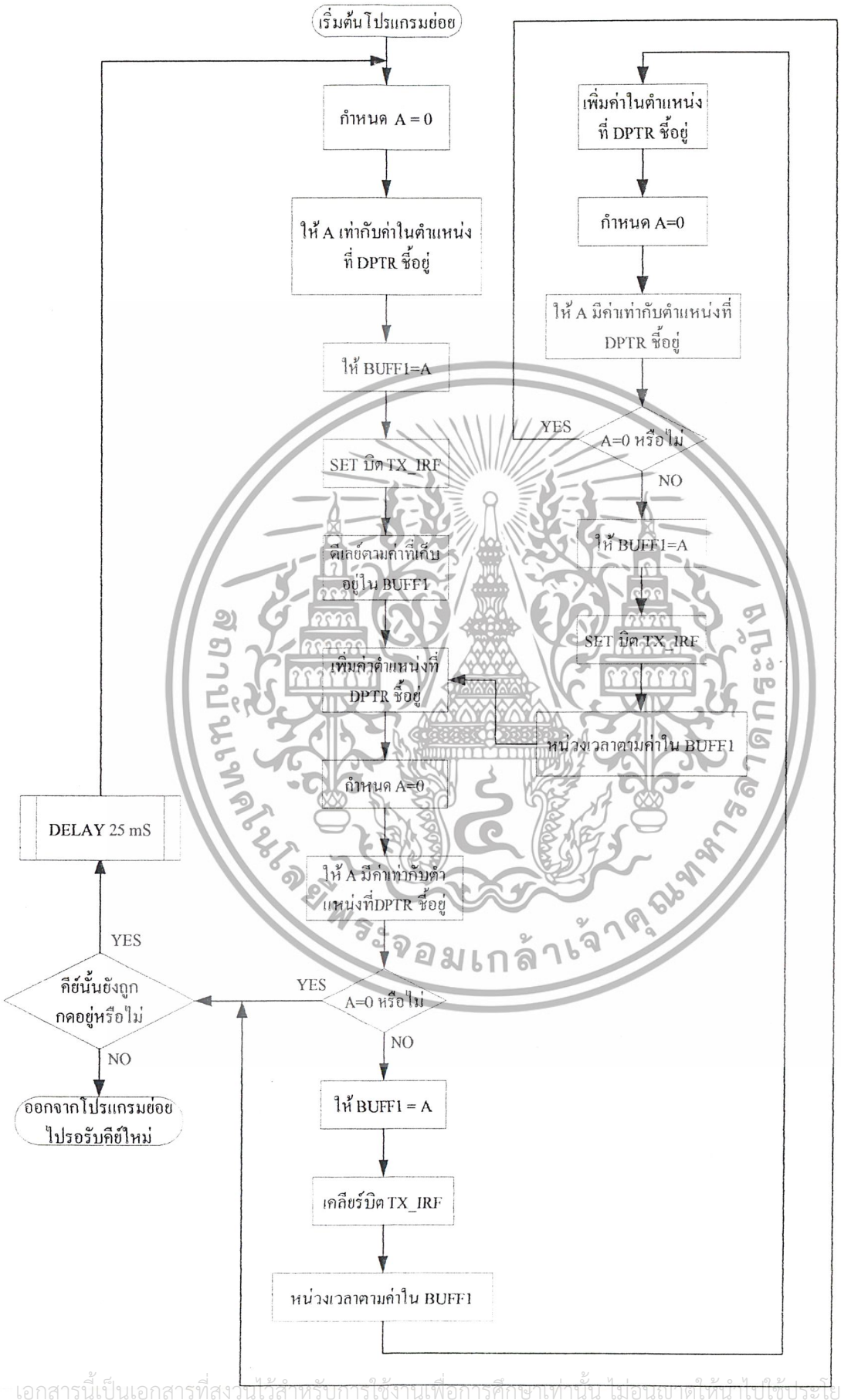
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.14 โฟลวชาร์ตโปรแกรมย่อยการส่งสัญญาณของ SONY และJVC

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โฟลวชาร์ตโปรแกรมย่อยในการสร้างพัลซ์



รูปที่ 3.15 โฟลวชาร์ตโปรแกรมย่อยในการสร้างพัลซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

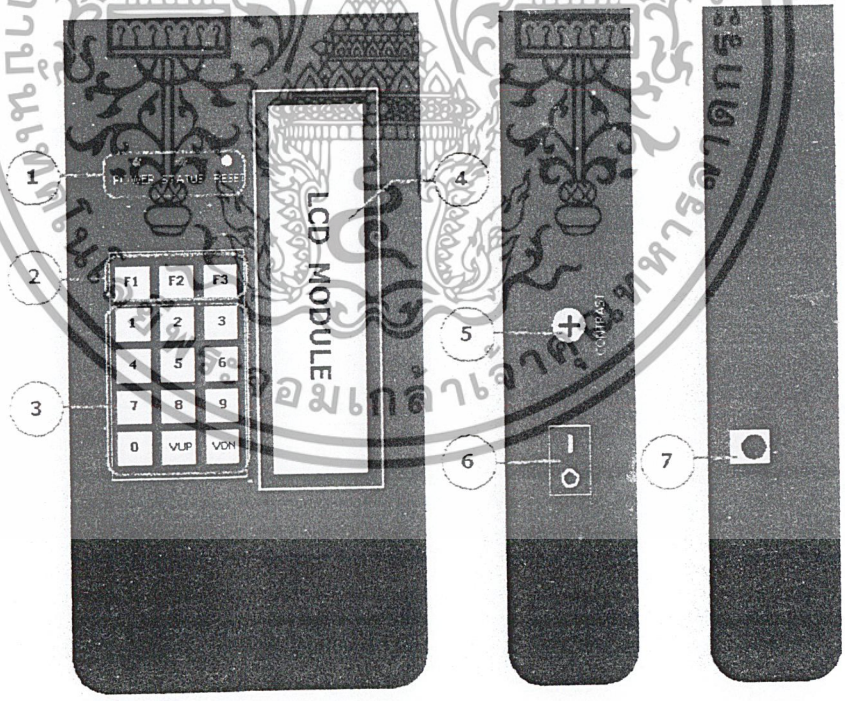
บทที่ 4

การทดลองและผลการทดลอง

4.1 คุณสมบัติเครื่องรับส่งสัญญาณด้วยรีโมทคอนโทรล

- รับรหัสสัญญาณอินฟราเรดไม่เกิน 20 มิลลิวินาที
- เก็บรหัสสัญญาณอินฟราเรดได้ 80 ตำแหน่ง
- ทดสอบสัญญาณที่รับเข้ามาได้ทางอินฟราเรดแอลอีดีตัวส่ง
- แสดงผลการทำงานเป็นค่ารหัสสัญญาณ ทำให้ง่ายต่อการตรวจสอบค่า
- สามารถส่งสัญญาณควบคุมเครื่องใช้ไฟฟ้าได้
- ใช้แรงดัน 9 โวลท์ในการทำงาน โดยเร็กกูเลตเปลี่ยนแรงดันเหลือ 5 โวลท์

4.2 โครงสร้างส่วนต่างๆของโครงงาน



รูปที่ 4.1 โครงสร้างส่วนต่างๆของโครงงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำอธิบาย

1.หลอด LED แสดงผลและปุ่มรีเซต โดยหลอดสีเขียวจะสว่างตลอดเวลาที่ใช้ชุดทดลอง และหลอดสีแดงใช้เพื่อแสดงสภาวะการทำงานของโปรแกรม

2.Function key ประกอบด้วย คีย์ F1, F2 และ F3 ใช้ในการควบคุมโปรแกรมผ่านทาง จอแสดงผล

3.Control key ประกอบด้วย key 0-9, คีย์ Vup (Volume Up) และคีย์ Vdn (Volume Down) ใช้ในฟังก์ชั่นการส่งสัญญาณออกไปควบคุมเครื่องใช้

4. จอแสดงผล เป็นจอ LCD Module ขนาด 20 ตัวอักษร 2 บรรทัด ใช้เพื่อแสดงผลการอ่านและลำดับการทำงานของโปรแกรม

5. ปุ่มปรับความเข้มของจอ LCD ใช้ในกรณีที่ด่านอ่อนจนจอซีดจาง

6. ปุ่มปิดชุดทดลอง

7. ส่วนรับสัญญาณรีโมทคอนโทรล ถ้าต้องการอ่านสัญญาณรีโมทในฟังก์ชั่นการอ่าน ต้องนำสัญญาณรีโมทนั้นมาส่ง ณ บริเวณนี้

4.3 การใช้งานเครื่องรับส่งสัญญาณด้วยรีโมทคอนโทรล

1. เมื่อเปิดสวิทซ์จ่ายไฟให้โครงงานไฟแสดงผลหมายเลข 3 จะติด ที่จอ LCD จะมีอักษรวิ่งที่แถวบนว่า Remote Control Reader Press Anykey Project 1&2/2543 แถวล่างตัวอักษรวิ่งอยู่กับที่ว่า Project 1&2/2543 ให้ทำการกดปุ่มใดก็ได้บนคีย์แพดแล้วปล่อย

รูปที่ 4.2 หน้าจอเมื่อเริ่มเปิดเครื่อง

2. เมื่อกดปุ่มใดๆแล้ว จะมีหน้าจอเลือกฟังก์ชั่นการทำงาน กด F1 เมื่อต้องการอ่านรหัส กด F2 เมื่อต้องการส่ง

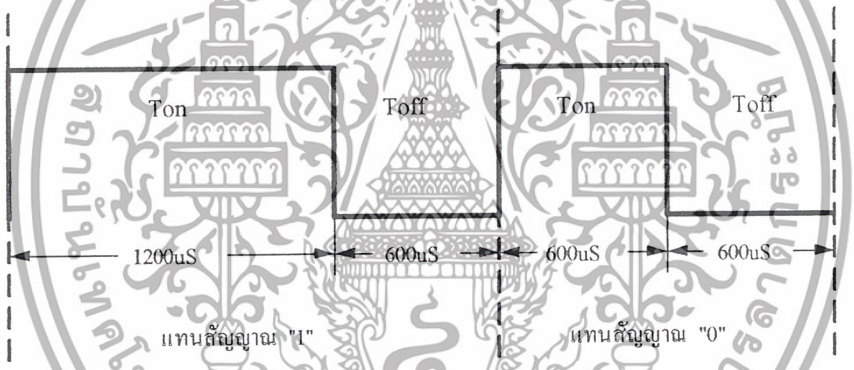
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SELECT FUNCTION

READ(F1) SEND(F2)

รูปที่ 4.3 หน้าจอเมื่อเครื่องพร้อมที่จะทำงานตามฟังก์ชั่น

3. ถ้ากด F1 หน้าจอจะแสดงฐานเวลาที่ 100 uS หมายถึง เมื่อจอแสดงผลแสดงตัวเลขออกมา เช่น 12 6 6 6 ให้นำ 100×10^{-6} ไปคูณทุกค่าที่แสดงออกมาผลลัพธ์จะแสดงออกมาเป็นช่วง T_{on} และ T_{off} ของสัญญาณโดยช่วงสัญญาณที่มี T_{on} นานกว่าจะเป็น 1 และที่น้อยกว่าจะเป็นศูนย์ จากค่าตัวอย่างอ่านได้



รูปที่ 4.4 ตัวอย่างของค่าที่อ่านได้

4. เมื่อกด F2 จะแสดงสภาวะรอสัญญาณรีโมทให้นำรีโมทคอนโทรลที่ต้องการอ่านค่ามาส่งสัญญาณ ณ จุดรับสัญญาณ หรือ กด F3 เพื่อออกจากการทำงานไปที่หน้าจอหลัก

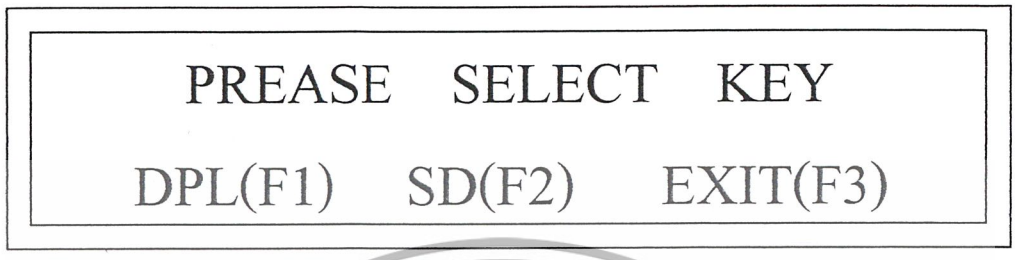
PRESS ONE REMOTE KEY

EXIT(3)

รูปที่ 4.5 หน้าจอเมื่อเครื่องพร้อมรับสัญญาณอินฟราเรดจากรีโมทคอนโทรล

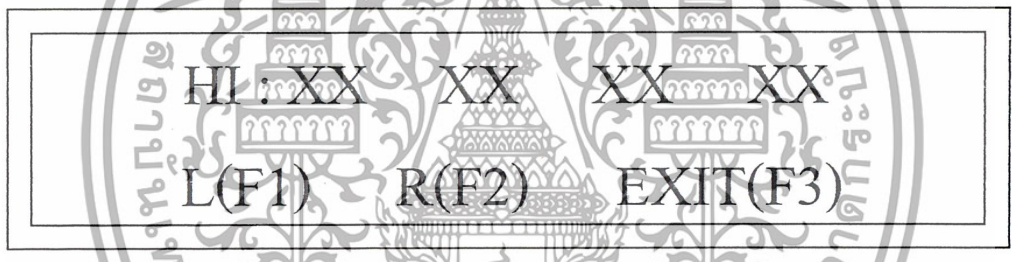
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. ถ้ากดสัญญาณแล้วเครื่องรับได้รับแล้วจะขึ้นหน้าจอเลือกการทำงาน แสดงผล (DPL) กด F1 ลองส่ง (SD) กด F2 และออกจากระบบไปรอสัญญาณปุ่มรีโมทใหม่ (EXIT) กด F3



รูปที่ 4.6 หน้าจอเมื่อเครื่องรับสัญญาณเรียบร้อยแล้ว

6. เลือกกด F1 จะแสดงหน้าจอดังนี้



รูปที่ 4.7 หน้าจอเมื่ออ่านค่าของสัญญาณที่รับมาจากรีโมทคอนโทรล

สัญญาณรีโมทจะเริ่มต้นที่ช่วง T_{on} ก่อนแล้วไล่ไปเรื่อยๆ ดังนั้น 1 คู่ ข้อมูลจะหมายถึง 1 cycle ของพัลซ์ 1 ลูก ถ้าต้องการอ่านพัลซ์ลูกต่อไป กด F2 เพื่อเลื่อนไปทางขวาถ้าอยากอ่านข้อมูลของพัลซ์ลูกแรกๆให้ กด F1 เพื่อเลื่อนไปทางซ้าย หรือออกไป 1 ขั้นให้ กด F3 เพื่อออกไปยังข้อ 5

จากหน้าจอหลัก ถ้าเลือกกด F2 จะเข้าสู่ฟังก์ชั่นการส่งซึ่งมีการใช้งานดังต่อไปนี้

1. หลังจากกด F2 จอแสดงผลจะให้เลือกยี่ห้อของเครื่องใช้ไฟฟ้า ซึ่งชุดทดลองเองสามารถเลียนแบบสัญญาณได้ ในที่นี้ได้ทดลองกับโทรทัศน์เพียง 2 ยี่ห้อ จึงมีให้เลือกดังนี้ คือกด F1 ใช้กับโทรทัศน์ SONY และ F2 ใช้กับ โทรทัศน์ JVC ถ้ากด F3 จะออกไปยังหน้าจอหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PLEASE SELECT KEY

SONY (F1) JVC (F2)

รูปที่ 4.8 หน้าจอการเลือกยี่ห้อเครื่องใช้ไฟฟ้า

2. หลังจากกดเลือกยี่ห้อแล้วฟังก์ชั่นการทำงานจะเหมือนกัน ผู้ใช้สามารถสั่งการให้ TV ยี่ห้อนั้นๆ ผ่านทางชุดทดลองได้ด้วยปุ่ม Control key ซึ่งจะเป็นปุ่มของการเลือกช่อง 1-9 ปุ่ม Power on-off ปุ่มเพิ่มเสียงและปุ่มลดเสียง ตามคำอธิบายข้างต้น หาต้องการออก ไปใช้อีกยี่ห้อ ให้กด F3

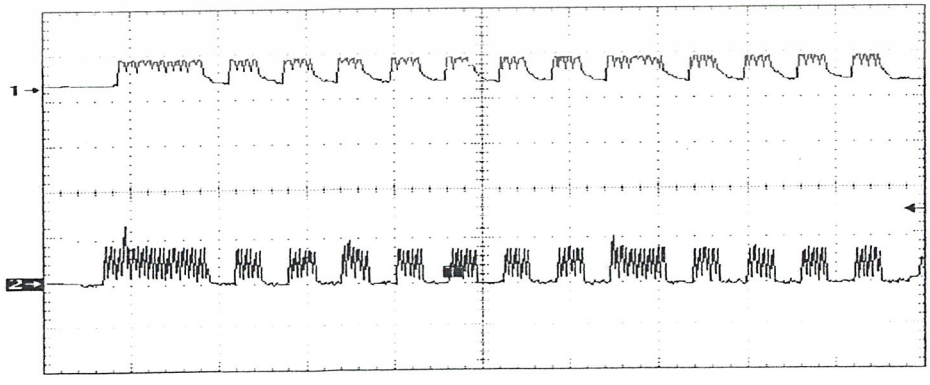
PRESS CONTROL KEY

EXIT (F3)

รูปที่ 4.9 หน้าจอการใช้คอนโทรลคีย์

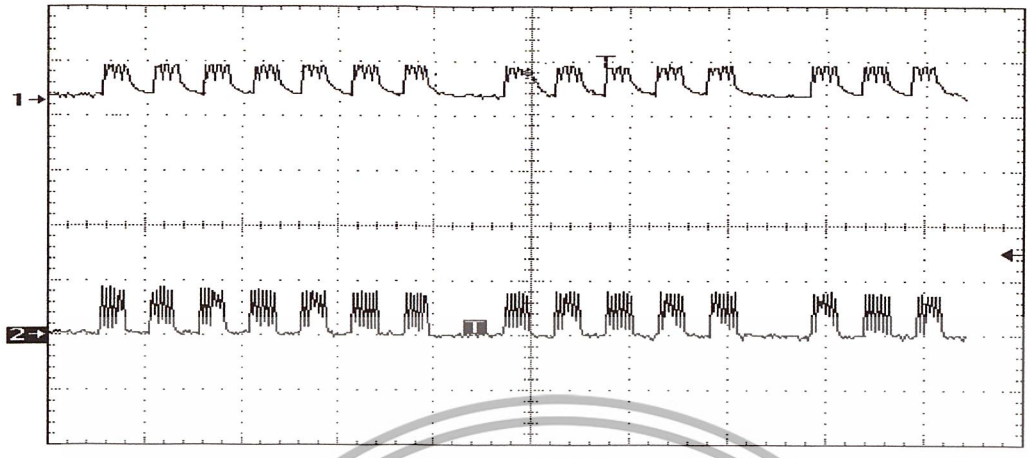
ผลการทดลอง

1. ในขั้นตอนแรก ทำการใช้ Oscilloscope จับสัญญาณระหว่าง สัญญาณจากรีโมทต้นแบบ เปรียบเทียบกับ สัญญาณที่ชุดทดลองเลียนแบบขึ้น ได้ผลดังนี้



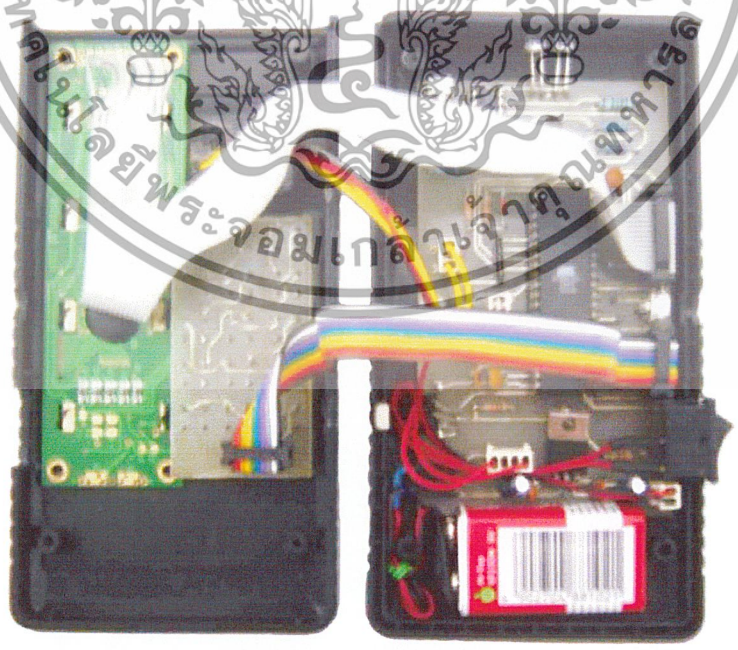
รูปที่ 4.10 สัญญาณรีโมทต้นแบบยี่ห้อ SONY กับสัญญาณที่ชุดทดลองเลียนขึ้นมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



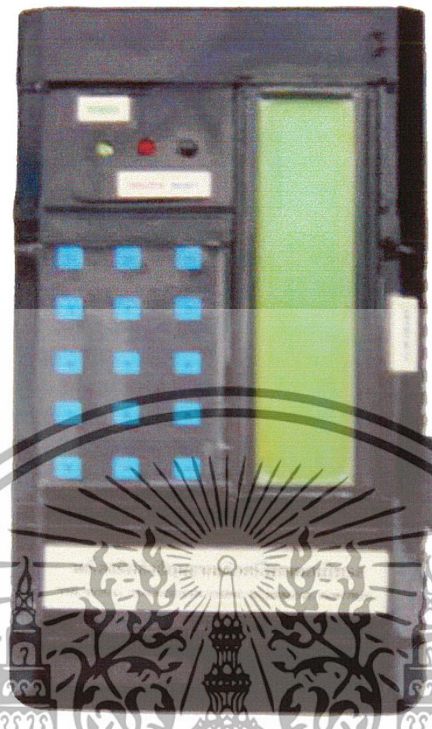
รูปที่ 4.11สัญญาณรีโมทต้นแบบยี่ห้อ JVC กับสัญญาณที่ชุดทดลองเลียนขึ้นมา

จากภาพทั้ง 2 อยู่ในฟังก์ชั่นการส่ง CH1 (บน) เป็นสัญญาณที่เลียนขึ้น ส่วน CH2 (ล่าง) เป็นสัญญาณต้นแบบ หากเปรียบเทียบกันในแต่ละช่วงเวลาของพัลซ์ จะเห็นว่ามีความคลาดเคลื่อนกันเล็กน้อยซึ่งอาจเกิดจากการนำค่าในหน่วยความจำมาอ่าน แล้วเขียนลงในโปรแกรม แต่เมื่อนำมาใช้งานจริง ทุกคีย์สามารถทำงานได้ตามต้องการอย่างแม่นยำในระยะที่น่าพอใจ (5 เมตร ถึง 7 เมตร)



รูปที่ 4.12 ภาพถ่ายภายในของเครื่องรับส่งสัญญาณด้วยรีโมทคอนโทรล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.13 ภาพถ่ายด้านบนของเครื่องรับส่งสัญญาณด้วยรีโมทคอนโทรล

ในฟังก์ชั่นการอ่านจะมีส่วนการทดลองสัญญาณ ซึ่งเป็นการนำค่าในหน่วยความจำภายในของ Microcontroller มาสร้างเป็นพัลซ์ แล้วมอดูเลตกับความถี่พาหะ ส่งออกทางภาคส่งอินฟราเรด ไปควบคุมเครื่องใช้เพื่อยืนยันความถูกต้องของข้อมูลพัลซ์ ซึ่งนำมาเก็บไว้ในหน่วยความจำภายใน ในส่วนนี้ก็สามารถใช้งานได้ดี เช่นเดียวกันทั้ง 2 ยี่ห้อข้างต้น แต่กับยี่ห้ออื่นๆยังไม่สามารถยืนยันได้ เพราะการส่งสัญญาณของเครื่องใช้ไฟฟ้าแต่ละยี่ห้อมีระบบการส่งแตกต่างกันออกไป ทั้งนี้เนื่องจากมีการป้องกันการซ้ำซ้อนของข้อมูลไปอีกระดับหนึ่ง ถือเป็นขีดจำกัดของโครงงานนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปและวิจารณ์ผลการทดลอง

โครงงานนี้เป็นการประยุกต์ใช้งานของรีโมททั่วๆไปให้มีฟังก์ชั่นการทำงานเพิ่มขึ้น มีส่วนของการอ่านสัญญาณ และแสดงผลผ่านจอ LCD หรือแม้แต่ การส่งสัญญาณไปมอดูเลตกับความถี่ผ่านทาง IC LM555 แต่โดยรวมจะเน้นหนักไปที่โปรแกรมใน Microcontroller เป็นหลัก จึงใช้เวลาส่วนใหญ่กับการเขียนโปรแกรม รวมทั้งการออกแบบลายวงจร ต้องทำให้เล็กพอเหมาะที่จะใช้เป็นรีโมทคอนโทรลได้ จึงต้องทำหลายครั้งเพื่อหาจุดที่เหมาะสมที่สุด

ส่วนผลการทดลอง ก็ได้ผลเป็นที่น่าพอใจสำหรับในส่วนของฟังก์ชั่นการส่งสามารถทำให้เครื่องใช้ไฟฟ้า ทำงานตามความประสงค์ได้เหมือนกับรีโมทต้นแบบ แม้ว่าส่วนการอ่านจะยังอ่านคลาดเคลื่อนอยู่บ้าง แต่สามารถนำค่าจากการอ่านซึ่งอยู่ใน RAM ของ Microcontrollerไปสร้างพัลซ์และควบคุมเครื่องใช้ไฟฟ้า ได้ตามที่จะสามารถหามาทดสอบได้ ด้านข้อมูลที่อ่านได้และแสดงผลบนจอ LCD ก็สามารถนำไปอ้างอิง เขียนเป็นโปรแกรมคงที่ไว้ใน Microcontroller ได้เพื่อใช้สร้างรีโมทใดๆ เพิ่มเติมในส่วนของฟังก์ชั่นส่งได้อีก ตามขนาดของหน่วยความจำของ Microcontroller นั้นๆ โดยอาศัยโครงสร้างโปรแกรม ซึ่งใช้งานได้ของโครงงานชุดนี้

ปัญหาที่เกิดขึ้นกับการทำงาน

1. ไม่สามารถใช้ Microcontroller สร้างความถี่ที่ 40 kHz ได้พอดี ทำให้การส่งไม่สามารถควบคุมเครื่องใช้ไฟฟ้าได้

การแก้ไข ใช้อุปกรณ์ทาง Hardware คือ IC LM555 มาช่วยสร้างความถี่ กับวงจรโดยให้ Microcontroller รับผิดชอบเพียงส่งพัลซ์ ตามจากเวลาที่ถูกต้องออกมารอการมอดูเลตเพียงอย่างเดียว นอกจากนี้ความถี่ซึ่งกำเนิดขึ้นยังสามารถปรับได้จึงสะดวกมาก ถ้าความถี่คลาดเคลื่อน

2. ระยะการส่งสั้นจากที่ควรเป็น 4-5 เมตร เหมือนกับรีโมททั่วๆ ไป

การแก้ไข เกิดได้สองกรณี คือ ความถี่คลาดเคลื่อน ในโครงงานนี้จะมีช่องเล็กๆ ใช้ในการปรับความถี่ให้แม่นยำขึ้น อยู่บริเวณทางด้านซ้ายของอินฟราเรดตัวส่ง

อีกกรณีหนึ่ง ถ้ายังไม่ลงอุปกรณ์ลงในลายวงจร อาจจะทำให้ระยะการส่งสั้นลงไปอีกมาก เพราะความถี่จะมีผลกับ Noise ในบอร์ดทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ลายวงจรผิดพลาด เนื่องจากโปรแกรมที่ใช้ในการออกแบบลายวงจร ทำให้ลายวงจรในส่วนของคีย์แพด โรว์ กับ คอลัมน์สลับกัน

การแก้ไข แก้ไขทางโปรแกรมโดยกำหนด ค่าตำแหน่งของคีย์แพดใหม่ให้มีความเหมาะสม และสะดวกต่อการใช้งานมากขึ้น และควรตรวจสอบวงจรบนแผ่นลายวงจรให้แน่นอนก่อนการลงมือทำแผ่นลายวงจรจริง

4. แหล่งจ่ายไฟมีขนาดใหญ่เกินไปที่จะเป็นรีโมทคอนโทรล เพราะต้องใช้ขับจอ LCD ใช้เพิ่มระยะการส่งอินฟราเรด อย่างต่ำต้องเป็น 5 Volt เพื่อเลี้ยงวงจร Microcontroller

การแก้ไข ใช้ถ่าน 9 Volt แล้วใช้ชุดปรับแรงดันให้คงที่ที่ 5 Volt สำหรับเลี้ยงวงจร Microcontroller และ 9 Volt โดยตรงสำหรับภาคส่งอินฟราเรด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. คู่มือ DOT MATRIX LCD MODULE , SILA CO., LTD.
2. เจน สงสมพันธุ์ , "โทรทัศน์สีภาคสนาม " โรงเรียนอิเล็กทรอนิกส์กรุงเทพ, 2534
3. ชัยวัฒน์ ลิ้มพรจิตรวิไล, วรพจน์ กรแก้ววัฒนกุล, " เรียนรู้และปฏิบัติการไมโครคอนโทรลเลอร์ MCS- 51 แบบแฟลช " บริษัท อินโนเวตีฟ เอ็กเพอริเมนต์ จำกัด

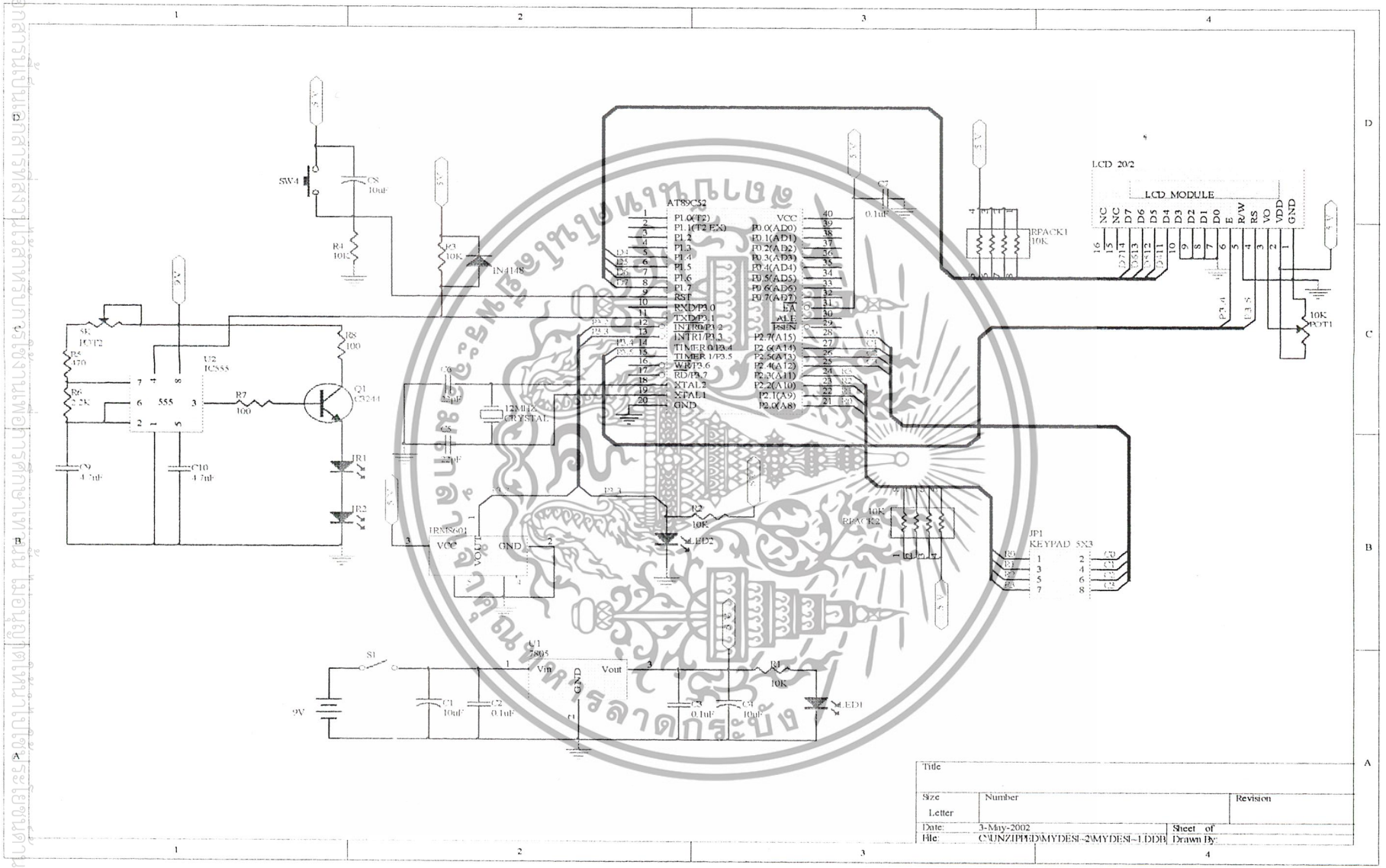
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

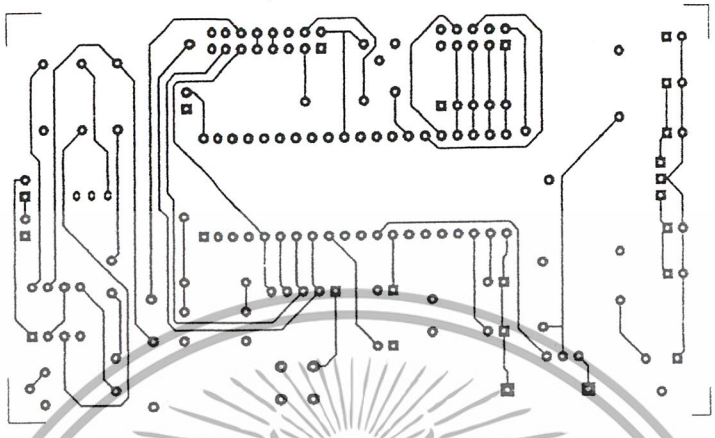
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



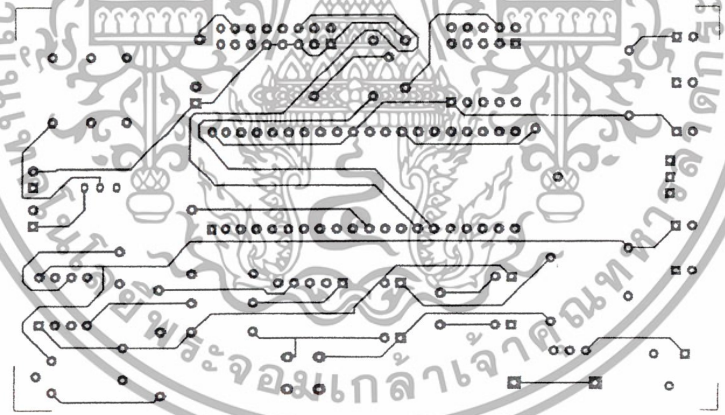
รูปที่ 1 วงจรจรรวม

ห้องสมุดพระจุฬาจอมเกล้า จุฬาลงกรณ์ทหารลาดกระบัง

รูปที่ 2 วงจรของการออกแบบคีย์แพด

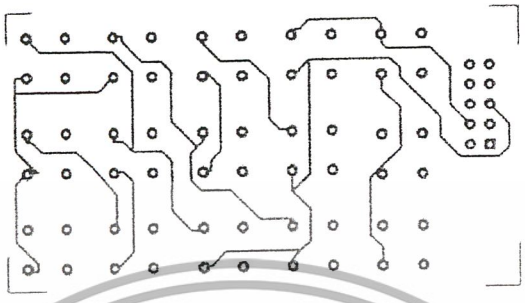


รูปที่ 3 ลายวงจรของวงจรรวม (ด้านบน)

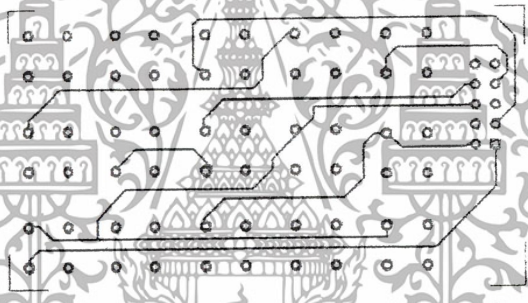


รูปที่ 4 ลายวงจรของวงจรรวม (ด้านล่าง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

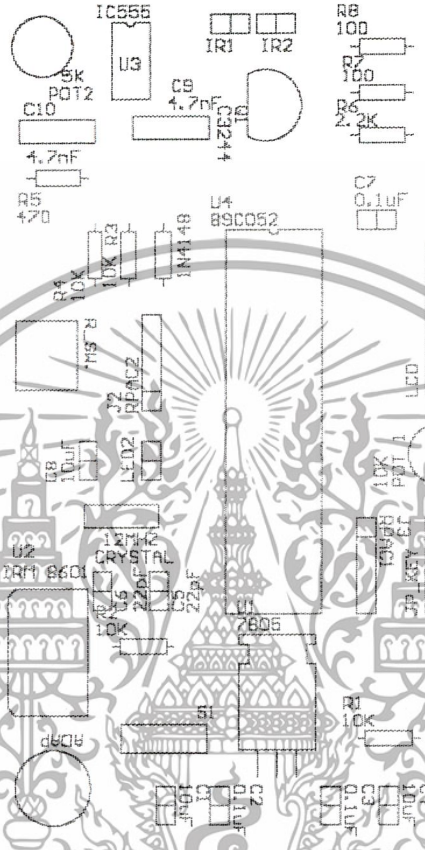


รูปที่ 5 ลายวงจรของคีย์แพด (ด้านบน)



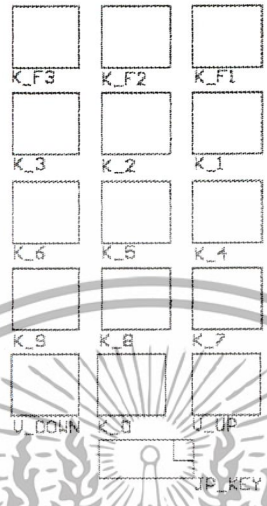
รูปที่ 6 ลายวงจรของคีย์แพด (ด้านล่าง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7 การวางอุปกรณ์ของวงจรรวม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 8 แสดงการวางอุปกรณ์ของคีย์แพด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ORG   0000H
;****************SET PORT*****************

;*****INPUT PORT CONNECT TO KEY METRIX *****
KPAD_ROW0  BIT        P2.0        ; Keypad Input Row 0
KPAD_ROW1  BIT        P2.1        ; Keypad Input Row 1
KPAD_ROW2  BIT        P2.2        ; Keypad Input Row 2
KPAD_ROW3  BIT        P2.3        ; Keypad Input Row 3
KPAD_COL3  BIT        P2.4        ; Keypad Output Column 3
KPAD_COL2  BIT        P2.5        ; Keypad Output Column 2
KPAD_COL1  BIT        P2.6        ; Keypad Output Column 1
KPAD_COL0  BIT        P2.7        ; Keypad Output Column 0

;*************** SET PORT ******************
TX_IRF     BIT   P3.1    ;Port for send infraled signal
RX_IRF     BIT   P3.2    ;Port for recive infraled signal
LED_WORK   BIT   P3.3    ;Port for STATUS LED
E_LCD      BIT   P3.4    ;To ENABLE LCD Module
RS_LCD     BIT   P3.5    ;To RS LCD Module

;************ Define User Register ***********
LINE            EQU   20H     ;Line LCD
COUNT_TABLE     EQU   21H     ;Count Table For Display
COUNT_DATA      EQU   22H     ;Count Data From "RX_IRF" Port
COUNT_H2D       EQU   23H     ;Count HEX Data At Show Display
DD_SHOW         EQU   24H     ;Tell Address Show At LCD
CHK_LEFT        EQU   25H     ;Check Data Initial For Display LCD
BUFF1           EQU   26H     ;Buffer1 for Sub Program
KPAD_DATA       EQU   27H     ;For keep Keypad Data
LCD_DATA        EQU   28H     ;Keep ASCII Data to Show LCD
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;*************SET DATA INITIAL******************
                CLR     TX_IRF
                CLR     LED_WORK
                MOV     P2,0FFH
BUFF_INIT:      MOV     R0,#127

LOOP_BUFF_INIT:
                MOV     @R0,#00
                DJNZ    R0,LOOP_BUFF_INIT
                MOV     COUNT_H2D,#30H
                MOV     TMOD,000000001B
                ACALL   LCD_INIT

;*************MAIN PROGRAM*************************
MAIN:           CLR     P1
                MOV     DPTR,#TB_NAME2    ;Show Year Of Project
                MOV     COUNT_TABLE,#20
                MOV     LINE,#02
                ACALL   SEND_LINE

DP_L:           MOV     R3,#00

DP_LEFT:        MOV     R2,#20      ;Counter Display 20 Adderss (27H - 3BH)
                MOV     A,R3
                MOV     R0,A
                MOV     R1,#BUFF1
                MOV     DPTR,#TB_NAME1    ;Show Name of Project Run from

DP_LFT1:        MOV     A,R0              ;Right To Left
                MOVC    A,@A+DPTR
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV     @R1,A
        INC     R0
        INC     R1
        DJNZ    R2,DP_LFT1
        ACALL   GET_KPAD
        MOV     A,KPAD_DATA
        JZ      WR_DP
        JNZ     K_00

WR_DP:  MOV     R0,#BUFF1
        MOV     A,#80H
        ACALL   WR_INS
        MOV     R2,#20

WR_DP1: MOV     A,@R0
        SETB    RS_LCD
        ACALL   WRITE
        INC     R0
        DJNZ    R2,WR_DP1
        INC     R3
        CJNE    R3,#60,DP_LEFT
        AJMP    DP_L
;-----------------------------------------

K_00    CLR     P1
        MOV     DPTR,#TB_FUNC       ;Show Messege "SELECT FUNCTION"
        MOV     COUNT_TABLE,#20     ;At Upper Line
        MOV     LINE,#01
        ACALL   SEND_LINE
        MOV     DPTR,#TB_FKEY       ;Show Messege "READ(F1)  SEND(F2)"
        MOV     COUNT_TABLE,#20     ;At Lower Line
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV     LINE,#02
                ACALL   SEND_LINE
;-----------------------------------------------
K_10:           MOV     R1,#01H
                ACALL   CHK_KEY
                CJNE    A,#00H,K_12
                JZ      K_01

K_12:           MOV     R1,#05H
                ACALL   CHK_KEY
                CJNE    A,#00H,K_10
                JZ      X0
X0:             AJMP    SEND_FUNC
;*************** READ FUNCTION**********************
K_01:           CLR     P1
                MOV     DPTR,#TB_TIME       ;Show Messege "TIME BASE = 100uS"
                MOV     COUNT_TABLE,#20
                MOV     LINE,#01
                ACALL   SEND_LINE
                MOV     DPTR,#TB_SAM        ;Show Messege "SAMPLING PULSE (2)"
                MOV     COUNT_TABLE,#20
                MOV     LINE,#02
                ACALL   SEND_LINE
;-----------------------------------------------
K_11:           MOV     R1,#05H
                ACALL   CHK_KEY
                CJNE    A, #00H,K_11
                JNZ     K_11
K_111:          CLR     P1
                MOV     DPTR,#TB_ONE        ;Show Messege "PRESS ONE REMOTE KEY"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV     COUNT_TABLE,#20
                MOV     LINE,#01
                ACALL   SEND_LINE
                MOV     DPTR,#TB_QUIT       ;Show Messege "EXIT(3)"
                MOV     COUNT_TABLE,#20
                MOV     LINE,#02
                ACALL   SEND_LINE
;-------------------------------------------------
K_21:           JB      RX_IRF,K_22
                ACALL   IN_REMOTE

END_IN_REMOTE:
                ACALL   HEX_INPUT_RX

END_PT:         MOV     DPTR,#TB_SELC1      ;Show Messege "PLEASE SELECT KEY"
                MOV     COUNT_TABLE,#20
                MOV     LINE,#01
                ACALL   SEND_LINE
                MOV     DPTR,#TB_SELC2      ;Show Messege Key "DPL(1) SD(2) EX(3)"
                MOV     COUNT_TABLE,#20
                MOV     LINE,#02
                ACALL   SEND_LINE
                AJMP    K_31

K_22:           MOV     R1,#09H
                ACALL   CHK_KEY
                CJNE    A,#00H,K_21
                JZ      HELP1
HELP1:          AJMP    K_00
;-------------------------------------------------
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
K_31:       MOV     R1,#01H
            ACALL   CHK_KEY
            CJNE    A,#00H,K_32
            JZ      RUN_RR

K_32:       MOV     R1,#05H         ;Check KEY 2
            ACALL   CHK_KEY
            CJNE    A,#00H,K_33
            INC     COUNT_DATA
            AJMP    TX_DATA

K_33:       MOV     R1,#09H         ;Check KEY 3
            ACALL   CHK_KEY
            CJNE    A,#00H,K_31
            JZ      HELP2
HELP2:      AJMP    CLEAR_RAM

;***************SUB PROGRAM*******************
;****SUB SHOW DATA TO DISPLAY LCD (LEFT OR RIGHT)******
;-------------------------------------------------
RUN_RR:     MOV     A,#01H
            ACALL   WR_INS
            MOV     R0,#30H

            MOV     DPTR,#TB_HI     ;Show "HI:"
            MOV     COUNT_TABLE,#03
            MOV     LINE,#01
            ACALL   SEND_LINE
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV     DPTR,#TB_LRQ        ;Show " L(F1) R(F2) EXIT(3) "
                MOV     COUNT_TABLE,#20
                MOV     LINE,#02
                ACALL   SEND_LINE

SHOW_DATA1:     ACALL   WR_SP16             ;Clear Display 16 BIT

SHOW_DATA:      ACALL   DSP_LCD1            ;Show 1st Data Of Page
                ACALL   WR_SPACE
                ACALL   DE_100mS

                INC     R0                  ;Show 2nd Data Of Page
                ACALL   DSP_LCD2
                ACALL   WR_SPACE
                ACALL   DE_100mS

                INC     R0                  ;Show 3rd Data Of Page
                ACALL   DSP_LCD3
                ACALL   WR_SPACE
                ACALL   DE_100mS

                INC     R0                  ;Show 4th Data Of Page
                ACALL   DSP_LCD4
                ACALL   WR_SPACE
                ACALL   DE_100mS
                AJMP    NK_01

NK_01:          MOV     R1,#01              ;Check Key For Shift Left Data
                ACALL   CHK_KEY
                CJNE    A,#00H,NK_02
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                AJMP    L_SFT

NK_02:          MOV     R1,#05      ;Check Key For Shift Right Data
                ACALL   CHK_KEY
                CJNE    A,#00H,NK_03
                AJMP    R_SFT

NK_03:          MOV     R1,#09      ;Check Key For Quit
                ACALL   CHK_KEY
                CJNE    A,#00H,NK_01
                AJMP    END_PT

;****** CHECK SHIFT LEFT OR SHIFT RIGHT DATA *******

L_SFT:          MOV     A,R0        ;Can't Shift Left At Data RAM Add.30H
                SUBB    A,#07H
N_VAUVE3:       CJNE    A,#2CH,BACK_L
                AJMP    WAIT
BACK_L:         MOV     R0,A
                AJMP    SHOW_DATA1
WAIT:           AJMP    NK_01

R_SFT:          INC     R0          ;Shift 4 Add.,No More Than 80H
                CJNE    R0,#80H,SHOW_DATA1
                AJMP    NK_01

;********** SHOW DATA AT LCD ************

DSP_LCD1:       ACALL   DATA_HI
                MOV     A,#84H      ;1st Data Display RAM At Add.80H
                ACALL   SS01        ;Show HI BIT At LCD DDRAM Add.84H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        ACALL   DATA_LOW        ;Show LOW BIT At LCD DDRAM Add.85H
        MOV     A,#85H
        ACALL   SS01
        RET

DSP_LCD2:   ACALL   DATA_HI     ;Show HI BIT At LCD DDRAM Add.88H
        MOV     A,#88H
        ACALL   SS01
        ACALL   DATA_LOW
        MOV     A,#89H          ;Show LOW BIT At LCD DDRAM Add.89H
        ACALL   SS01
        RET

DSP_LCD3:   ACALL   DATA_HI
        MOV     A,#8CH
        ACALL   SS01
        ACALL   DATA_LOW
        MOV     A,#8DH
        ACALL   SS01
        RET

DSP_LCD4:   ACALL   DATA_HI
        MOV     A,#90H
        ACALL   SS01
        ACALL   DATA_LOW
        MOV     A,#91H
        ACALL   SS01
        RET

SS01:   ACALL   WR_INS
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ACALL   W_LCD3
                RET

W_LCD3:         CLR     A
                MOV     A,LCD_DATA
                ACALL   WR_LCD
                RET

DATA_HI:        MOV     A,@R0       ;Compare HI BIT Value In RAM
                ANL     A,#0F0H     ;For Show At LCD
                SWAP    A
                ACALL   COMPARE
                RET

DATA_LOW:       MOV     A,@R0       ;Compare LOW BIT Value In RAM
                ANL     A,#0FH      ;For Show At LCD
                ACALL   COMPARE
                RET

WR_SPACE:       MOV     A,#14H      ;Write Space 2 BIT
                ACALL   WR_INS
                MOV     A,#14H
                ACALL   WR_INS
                RET
WR_SP16:        MOV     R3,#16
                MOV     A,#84H
WR_SP:          MOV     LCD_DATA,#''
                ACALL   SS01
                DJNZ    R3,WR_SP
                RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;********** COMPARE DATA TO SHOW AT LCD ************

COMPARE:    CJNE    A,#00H,CP_NO1
            MOV     LCD_DATA,#'0'
            RET
CP_NO1:     CJNE    A,#01H,CP_NO2
            MOV     LCD_DATA,#'1'
            RET
CP_NO2:     CJNE    A,#02H,CP_NO3
            MOV     LCD_DATA,#'2'
            RET
CP_NO3:     CJNE    A,#03H,CP_NO4
            MOV     LCD_DATA,#'3'
            RET
CP_NO4:     CJNE    A,#04H,CP_NO5
            MOV     LCD_DATA,#'4'
            RET
CP_NO5:     CJNE    A,#05H,CP_NO6
            MOV     LCD_DATA,#'5'
            RET
CP_NO6:     CJNE    A,#06H,CP_NO7
            MOV     LCD_DATA,#'6'
            RET
CP_NO7:     CJNE    A,#07H,CP_NO8
            MOV     LCD_DATA,#'7'
            RET
CP_NO8:     CJNE    A,#08H,CP_NO9
            MOV     LCD_DATA,#'8'
            RET
CP_NO9:     CJNE    A,#09H,CP_NOA
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV     LCD_DATA,#'9'
                RET
CP_NOA:         CJNE    A,#0AH,CP_NOB
                MOV     LCD_DATA,#'A'
                RET
CP_NOB:         CJNE    A,#0BH,CP_NOC
                MOV     LCD_DATA,#'B'
                RET
CP_NOC:         CJNE    A,#0CH,CP_NOD
                MOV     LCD_DATA,#'C'
                RET
CP_NOD:         CJNE    A,#0DH,CP_NOE
                MOV     LCD_DATA,#'D'
                RET
CP_NOE:         CJNE    A,#0EH,CP_NOF
                MOV     LCD_DATA,#'E'
                RET
CP_NOF:         MOV     LCD_DATA,#'F'
                RET

DE_100mS:       MOV     R5,#64H
                ACALL   DELAY
                RET
;********** SUB CLEAR RAM ************************

CLEAR_RAM:      MOV     R0,#127

CLEAR_1:        MOV     @R0,#00
                DJNZ    R0,CLEAR_1
                MOV     COUNT_H2D,#30H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                CLR     C
                AJMP    K_111

;********SUB SEND DATA TO "TX_IRF (P3.1)"*********

TX_DATA:        MOV     R0,#30H
                SETB    LED_WORK

TX_LOOP:        MOV     BUFF1,@R0
                SETB    TX_IRF
                ACALL   DE_13US
                INC     R0
                MOV     A,R0
                CJNE    A,COUNT_DATA,TX_LOOP1
                JNC     END_TX

TX_LOOP1:       MOV     BUFF1,@R0
                CLR     TX_IRF
                ACALL   DE_13US
                INC     R0
                MOV     A,R0
                CJNE    A,COUNT_DATA,TX_LOOP
                JNC     END_TX

END_TX:         MOV     R1,#05H
                ACALL   CHK_KEY
                JZ      SET
                DEC     COUNT_DATA
                CLR     LED_WORK
                AJMP    K_31
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SET:            ACALL   DELAY_25mS
                AJMP    TX_DATA
;---------------------------------------------------------
DE_13US:        MOV     R7,#3
DE_1:           NOP
                DJNZ    R7,DE_1
                NOP
                DJNZ    BUFF1,DE_13US
                RET
;************SUB IN REMOTE*****************************
;--------------CHECK START PULSE-----------------------
IN_REMOTE:      MOV     R0,#30H         ;First RAM Address For Save Data
                SETB    LED_WORK
                JB      RX_IRF,$
;----------SAVE START PULSE RX(L)-----------------------
                ACALL   LOAD_TIMER
                JNB     RX_IRF,$
                JB      TF0,IN_REMOTE
                ACALL   SUBB_TH0
;---------SAVE START PULSE RX(H)------------------------
                ACALL   LOAD_TIMER
                JB      RX_IRF,$
                JB      TF0,IN_REMOTE
                ACALL   SUBB_TH0
;---------SAVE DATA PULSE RX(L)-----------------------
RX_LOW_PULSE:
                ACALL   LOAD_TIMER
                JNB     RX_IRF,$
                JB      TF0,ERROR_PULSE
                ACALL   SUBB_TH0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                CJNE    R0,#81H,RX_HIGH_PULSE
                MOV     COUNT_DATA,#80H
                MOV     R5,#0FFH        ;Delay 5mSec For Status LED
                ACALL   DELAY
                CLR     LED_WORK
                RET
;----------SAVE DATA PULSE RX(H)---------------------
RX_HIGH_PULSE:
                ACALL   LOAD_TIMER
                JB      RX_IRF,$
                JB      TF0,ERROR_PULSE
                ACALL   SUBB_TH0
                AJMP    RX_LOW_PULSE
;-----------------------------------------------------
ERROR_PULSE:
                CLR     TR0
                MOV     COUNT_DATA,R0
                MOV     R5,#0FFH
                ACALL   DELAY
                CLR     LED_WORK
                AJMP    END_IN_REMOTE
;-----------------------------------------------------
LOAD_TIMER:     CLR     TR0             ;For Check Infraled Signal<=20mSec
                CLR     TF0
                MOV     TH0,#0B1H
                MOV     TL0,#0E0H
                SETB    TR0
                RET
;-------------------------------------------------------------
SUBB_TH0:       MOV     A,TH0           ;Change Data At TH0 To Real
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                CLR     C           ;For Save To RAM
                SUBB    A,#0B1H
                MOV     @R0,A
                INC     R0
                RET

;**********SUB CHANGE RX_IRF TO REAL HEX DATA &SAVE RAM**********

;**********SAMPLING PULSE WITH TIME BASE 100uSec*****************
HEX_INPUT_RX:
                MOV     R7,COUNT_DATA
                MOV     R0,#30H

LOOP_HEX_INPUT_RX:
                MOV     BUFF1,#00
                CLR     A
                MOV     DPH,@R0
                MOV     DPL,#00H
                MOV     @R0,#00
                MOV     R6,#10H
;-------------------------------------------------------------
DIV:            MOV     PSW,#00
                MOV     A,DPL
                RLC     A
                MOV     DPL,A
                MOV     A,DPH
                RLC     A
                MOV     DPH,A
                MOV     A,@R0
                RL      A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV     @R0,A
                MOV     A,BUFF1
                RLC     A
                MOV     BUFF1,A
                CJNE    A,#64H,CHK_FLAG
                JNC     CHK_1

CHK_FLAG:       JC      CHK_0

CHK_1:          SUBB    A,#64H
                MOV     BUFF1,A
                MOV     A,@R0
                ADD     A,#01H
                MOV     @R0,A

CHK_0:          DJNZ    R6,DIV

CHK_PASS:       INC     R0
                DJNZ    R7,LOOP_HEX_INPUT_RX
                RET

;************SEND FUNCTION**********************
 ;***PROGRAM FOR TESTING DATA FROM READ FUNCTION**

SEND_FUNC:      MOV     DPTR,#TB_SELC1   ;Show Messege " PREASE SELECT KEY  "
                MOV     COUNT_TABLE,#20
                MOV     LINE,#01
                ACALL   SEND_LINE
                MOV     DPTR,#TB_BND     ;Show Messege " SONY(F1) JVC(F2)  "
                MOV     COUNT_TABLE,#20
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV     LINE,#02
                ACALL   SEND_LINE

BND_LOOP:       MOV     R1,#01
                ACALL   CHK_KEY
                JZ      SLOOP

                MOV     R1,#05
                ACALL   CHK_KEY
                JZ      JLOOP

                MOV     R1,#09
                ACALL   CHK_KEY
                JZ      QUIT
                AJMP    BND_LOOP

SLOOP:          AJMP    SONY_KEY
JLOOP:          AJMP    JVC_KEY
QUIT:           AJMP    K_00

;----------SONY CONTROL KEY----------

SONY_KEY:       MOV     R3,#01
                AJMP    POINT

SKEY_1:         ACALL   INI_SD
                MOV     DPTR,#SONY_K_1
                AJMP    DATA_PULSE

SKEY_2:         ACALL   INI_SD
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV     DPTR,#SONY_K_2
                AJMP    DATA_PULSE

SKEY_3:         ACALL   INI_SD
                MOV     DPTR,#SONY_K_3
                AJMP    DATA_PULSE

SKEY_4:         ACALL   INI_SD
                MOV     DPTR,#SONY_K_4
                AJMP    DATA_PULSE

SKEY_5:         ACALL   INI_SD
                MOV     DPTR,#SONY_K_5
                AJMP    DATA_PULSE

SKEY_6:         ACALL   INI_SD
                MOV     DPTR,#SONY_K_6
                AJMP    DATA_PULSE

SKEY_7:         ACALL   INI_SD
                MOV     DPTR,#SONY_K_7
                AJMP    DATA_PULSE

SKEY_8:         ACALL   INI_SD
                MOV     DPTR,#SONY_K_8
                AJMP    DATA_PULSE

SKEY_9:         ACALL   INI_SD
                MOV     DPTR,#SONY_K_9
                AJMP    DATA_PULSE
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SKEY_0:     ACALL   INI_SD
            MOV     DPTR,#SONY_K_0
            AJMP    DATA_PULSE

SKEY_VUP:   ACALL   INI_SD
            MOV     DPTR,#SONY_V_UP
            AJMP    DATA_PULSE

SKEY_VDN:   ACALL   INI_SD
            MOV     DPTR,#SONY_V_DN
            AJMP    DATA_PULSE

;----------JVC CONTROL KEY ----------

JVC_KEY:    MOV     R3,#02
            AJMP    POINT

JKEY_1:     ACALL   INI_SD
            MOV     DPTR,#JVC_K_1
            AJMP    DATA_PULSE

JKEY_2:     ACALL   INI_SD
            MOV     DPTR,#JVC_K_2
            AJMP    DATA_PULSE

JKEY_3:     ACALL   INI_SD
            MOV     DPTR,#JVC_K_3
            AJMP    DATA_PULSE

JKEY_4:     ACALL   INI_SD
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV     DPTR,#JVC_K_4
                AJMP    DATA_PULSE

JKEY_5:         ACALL   INI_SD
                MOV     DPTR,#JVC_K_5
                AJMP    DATA_PULSE

JKEY_6:         ACALL   INI_SD
                MOV     DPTR,#JVC_K_6
                AJMP    DATA_PULSE

JKEY_7:         ACALL   INI_SD
                MOV     DPTR,#JVC_K_7
                AJMP    DATA_PULSE

JKEY_8:         ACALL   INI_SD
                MOV     DPTR,#JVC_K_8
                AJMP    DATA_PULSE

JKEY_9:         ACALL   INI_SD
                MOV     DPTR,#JVC_K_9
                AJMP    DATA_PULSE

JKEY_0:         ACALL   INI_SD
                MOV     DPTR,#JVC_K_0
                AJMP    DATA_PULSE

JKEY_VUP:       ACALL   INI_SD
                MOV     DPTR,#JVC_V_UP
                AJMP    DATA_PULSE
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
JKEY_VDN:   ACALL   INI_SD
            MOV     DPTR,#JVC_V_DN
            AJMP    DATA_PULSE

;---------- NEW SET----------------
NEW_SET:    CLR     TX_IRF
            ACALL   DELAY_25mS
            ACALL   GET_KPAD
            MOV     A,KPAD_DATA
            JZ      S_0

            MOV     R1,#13
            ACALL   CHK_KEY
            JZ      N_KEY1

            MOV     R1,#02
            ACALL   CHK_KEY
            JZ      N_KEY2

            MOV     R1,#06
            ACALL   CHK_KEY
            JZ      N_KEY3

            MOV     R1,#10
            ACALL   CHK_KEY
            JZ      N_KEY4

            MOV     R1,#14
            ACALL   CHK_KEY
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        JZ      N_KEY5

        MOV     R1,#03
        ACALL   CHK_KEY
        JZ      N_KEY6

        MOV     R1,#07
        ACALL   CHK_KEY
        JZ      N_KEY7

        MOV     R1,#11
        ACALL   CHK_KEY
        JZ      N_KEY8

        MOV     R1,#15
        ACALL   CHK_KEY
        JZ      N_KEY9

        MOV     R1,#08
        ACALL   CHK_KEY
        JZ      N_KEY0

        MOV     R1,#04
        ACALL   CHK_KEY
        JZ      N_KEYVUP

        MOV     R1,#12
        ACALL   CHK_KEY
        JZ      N_KEYVDN
        AJMP    SAM_KEY
S_0:
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
N_KEY1:     CJNE    R3,#01,J_1
            AJMP    SKEY_1
J_1:        AJMP    JKEY_1

N_KEY2:     CJNE    R3,#01,J_2
            AJMP    SKEY_2
J_2:        AJMP    JKEY_2

N_KEY3:     CJNE    R3,#01,J_3
            AJMP    SKEY_3
J_3:        AJMP    JKEY_3

N_KEY4:     CJNE    R3,#01,J_4
            AJMP    SKEY_4
J_4:        AJMP    JKEY_4

N_KEY5:     CJNE    R3,#01,J_5
            AJMP    SKEY_5
J_5:        AJMP    JKEY_5

N_KEY6:     CJNE    R3,#01,J_6
            AJMP    SKEY_6
J_6:        AJMP    JKEY_6

N_KEY7:     CJNE    R3,#01,J_7
            AJMP    SKEY_7
J_7:        AJMP    JKEY_7

N_KEY8:     CJNE    R3,#01,J_8
            AJMP    SKEY_8
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
J_8:            AJMP    JKEY_8

N_KEY9:         CJNE    R3,#01,J_9
                AJMP    SKEY_9
J_9:            AJMP    JKEY_9

N_KEY0:         CJNE    R3,#01,J_0
                AJMP    SKEY_0
J_0:            AJMP    JKEY_0

N_KEYVUP:       CJNE    R3,#01,J_VUP
                AJMP    SKEY_VUP
J_VUP:          AJMP    JKEY_VUP

N_KEYVDN:       CJNE    R3,#01,J_VDN
                AJMP    SKEY_VDN
J_VDN:          AJMP    JKEY_VDN

;---------- SHOW KEY IN REMOTE ----------

SHOW_KEY:       MOV     DPTR,#TB_KSEL    ;Show Messege "PRESS KEY 1 - 9 "
                MOV     COUNT_TABLE,#20
                MOV     LINE,#01
                ACALL   SEND_LINE
                MOV     DPTR,#TB_QUIT    ;Show Messege "  EXIT(3)  "
                MOV     COUNT_TABLE,#20
                MOV     LINE,#02
                ACALL   SEND_LINE
                RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;--------- DELAY 25mS LOOP----------------

DELAY_25mS:   MOV     R5,#25
              ACALL   DELAY

;--------INITIAL STATUS PORT----------------
INI_SD:       CLR     TX_IRF
              SETB    LED_WORK
              CLR     A
              RET

;------- POINT DATA WHEN PRESS KEY ---------

POINT:        ACALL   SHOW_KEY
SAM_KEY:      CLR     LED_WORK
              CLR     TX_IRF
SS_KEY1:      MOV     R1,#13
              ACALL   CHK_KEY
              CJNE    A,#00H,SS_KEY2
              CJNE    R3,#01,JK1
              AJMP    SKEY_1
JK1:          AJMP    JKEY_1

SS_KEY2:      MOV     R1,#02
              ACALL   CHK_KEY
              CJNE    A,#00H,SS_KEY3
              CJNE    R3,#01,JK2
              AJMP    SKEY_2
JK2:          AJMP    JKEY_2
SS_KEY3:      MOV     R1,#06
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ACALL   CHK_KEY
                CJNE    A,#00H,SS_KEY4
                CJNE    R3,#01,JK3
                AJMP    SKEY_3
JK3:            AJMP    JKEY_3

SS_KEY4:        MOV     R1,#10
                ACALL   CHK_KEY
                CJNE    A,#00H,SS_KEY5
                CJNE    R3,#01,JK4
                AJMP    SKEY_4
JK4:            AJMP    JKEY_4

SS_KEY5:        MOV     R1,#14
                ACALL   CHK_KEY
                CJNE    A,#00H,SS_KEY6
                CJNE    R3,#01,JK5
                AJMP    SKEY_5
JK5:            AJMP    JKEY_5

SS_KEY6:        MOV     R1,#03
                ACALL   CHK_KEY
                CJNE    A,#00H,SS_KEY7
                CJNE    R3,#01,JK6
                AJMP    SKEY_6
JK6:            AJMP    JKEY_6

SS_KEY7:        MOV     R1,#07
                ACALL   CHK_KEY
                CJNE    A,#00H,SS_KEY8
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                CJNE    R3,#01,JK7
                AJMP    SKEY_7
JK7:            AJMP    JKEY_7

SS_KEY8:        MOV     R1,#11
                ACALL   CHK_KEY
                CJNE    A,#00H,SS_KEY9
                CJNE    R3,#01,JK8
                AJMP    SKEY_8
JK8:            AJMP    JKEY_8

SS_KEY9:        MOV     R1,#15
                ACALL   CHK_KEY
                CJNE    A,#00H,SS_KEY0
                CJNE    R3,#01,JK9
                AJMP    SKEY_9
JK9:            AJMP    JKEY_9

SS_KEY0:        MOV     R1,#08
                ACALL   CHK_KEY
                CJNE    A,#00H,SS_KV_UP
                CJNE    R3,#01,JK0
                AJMP    SKEY_0
JK0:            AJMP    JKEY_0

SS_KV_UP:       MOV     R1,#04
                ACALL   CHK_KEY
                CJNE    A,#00H,SS_KV_DN
                CJNE    R3,#01,JKVU
                AJMP    SKEY_VUP
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
JKVU:       AJMP    JKEY_VUP

SS_KV_DN:   MOV     R1,#12
            ACALL   CHK_KEY
            CJNE    A,#00H,QS_LOOP
            CJNE    R3,#01,JKVD
            AJMP    SKEY_VDN
JKVD:       AJMP    JKEY_VDN

QS_LOOP:    MOV     R1,#09H
            ACALL   CHK_KEY
            CJNE    A,#00H,S_1
            JZ      QUIT1

QUIT1:      AJMP    SEND_FUNC

S_1:        AJMP    SAM_KEY

;----------SUB PROGRAM TO BUILD SIGNAL--------
DATA_PULSE: MOVC    A,@A+DPTR
            MOV     BUFF1,A
            SETB    TX_IRF
            ACALL   DE_13US

JK_LOOP:    INC     DPTR
            CLR     A
            MOVC    A,@A+DPTR
            JZ      H_3
            MOV     BUFF1,A
            CLR     TX_IRF
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ACALL   DE_13US

                INC     DPTR
                CLR     A
                MOVC    A,@A+DPTR
                JZ      H_3
                MOV     BUFF1,A
                SETB    TX_IRF
                ACALL   DE_13US
                AJMP    JK_LOOP

H_3:            AJMP    NEW_SET

;-----------DELAY 25uSEC-----------------
DELAY_LOOP:     MOV     R6,#07
DELOOP1:        NOP                     ;1 Loop = 25uS
                DJNZ    R6,DELOOP1
                NOP
                DJNZ    BUFF1,DELAY_LOOP    ;Varie by BUFF1
                RET

;***************SUB SEND DATA TO LCD*********************
;Sub SEND_LINE Used SEND TABLE
;Sub SD_LINE   Used SEND DATA AT RAM
SEND_LINE:      MOV     A,LINE
                CJNE    A,#01,SEND_LINE2    ;Set Address Of LINE1
                MOV     A,#80H
                AJMP    SS1
SEND_LINE2:     MOV     A,#0C0H             ;Set Address Of LINE2
SS1:            ACALL   WR_INS
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ACALL   W_LCD1
                RET

SD_LINE:        MOV     A,DD_SHOW
                ACALL   WR_INS
                ACALL   W_LCD2
                RET

;*******SUB CONFIG FOR LCD**************************
;-------------SUB LCD INIT 4 BIT--------------------
LCD_INIT:       CLR     RS_LCD
                MOV     A,#33H      ;Set DL=1 Three Time
                ACALL   WR_INS      ;Set DL=0 One  Time
                MOV     A,#32H
                ACALL   WR_INS
                MOV     A,#28H      ;Set 4-BIT Function For Send Data
                ACALL   WR_INS
                MOV     A,#0FH      ;Display Blink
                ACALL   WR_INS
                MOV     A,#06H      ;Set Entry Mode
                ACALL   WR_INS
                MOV     A,#01H      ;Clear Display
                ACALL   WR_INS
                RET
;---------SUB WRITE INSTRUCTION LCD------------------
WR_INS:         CLR     RS_LCD
                AJMP    WRITE
;---------SUB WRITE DATA TO LCD----------------------
W_LCD1:         CLR     A
                MOVC    A,@A+DPTR
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ACALL   WR_LCD
                INC     DPTR
                DJNZ    COUNT_TABLE,W_LCD1
                RET

W_LCD2:         MOV     R1,#2CH
                MOV     R4,#04

WLCD:           MOV     A,@R1
                ACALL   WR_LCD
                INC     R1
                DJNZ    R4,WLCD
                RET

WR_LCD:         SETB    RS_LCD
WRITE:          MOV     B,A     ;Send Data To LCD At DB4-DB7
                ANL     A,#0F0H
                MOV     P1,A    ;High Byte
                ACALL   EN_LCD
                MOV     A,B     ;Low Byte
                SWAP    A
                ANL     A,#0F0H
                MOV     P1,A
                ACALL   EN_LCD
                RET
;--------SUB ENABLE PULSE LCD----------------------
EN_LCD:         CLR     E_LCD
                MOV     R5,#05
                ACALL   DELAY
                SETB    E_LCD
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                RET
;***********SUB KEY DEFINING*****************
CHK_KEY:        ACALL   GET_KPAD        ;KEY 3
                MOV     A,KPAD_DATA
                XRL     A,R1
                RET

;********SUB DELAY TIME AT 12MHz*******************
DELAY:          MOV     R6,#04      ;Delay Time 1mSec
DE:             MOV     R7,#7CH     ;Varie By Reg.,R5

DE1:            DJNZ    R7,DE1
                DJNZ    R6,DE
                DJNZ    R5,DELAY
                RET
;-------------------------------------------------------------
; Keypad Scan key Subroutine
;-------------------------------------------------------------
GET_KPAD:       MOV     P2,#0FFH            ; Pull P2 to High
                MOV     KPAD_DATA,#0        ; Clear Keypad Data
                CLR     A

CHK_COL0:       CLR     KPAD_COL0           ; Begin Scan Column 0
                MOV     A,P2                ; Get Port2 Value
                ANL     A,#00FH             ; Get only lower 4 bit
                CJNE    A,#00FH,COL0_DETECT     ; Check All rows '1'?
                AJMP    CHK_COL1    ; All rows '1' => check next column

COL0_DETECT:    MOV     KPAD_DATA,#01       ; Initial KPAD_DATA = 1
                AJMP    GET_ROW             ; Jump to get row value
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CHK_COL1:       SETB    KPAD_COL0           ; Stop Scan Column 0
                CLR     KPAD_COL1           ; Begin Scan Column 1
                MOV     A,P2                ; Get Port2 Value
                ANL     A,#00FH             ; Get only lower 4 bit
                CJNE    A,#00FH,COL1_DETECT     ; Check All rows '1'?
                AJMP    CHK_COL2            ; All rows '1' => check next column

COL1_DETECT:    MOV     KPAD_DATA,#02       ; Initial KPAD_DATA = 2
                AJMP    GET_ROW             ; Jump to get row value

CHK_COL2:       SETB    KPAD_COL1           ; Stop Scan Column 1
                CLR     KPAD_COL2           ; Begin Scan Column 2
                MOV     A,P2                ; Get Port2 Value
                ANL     A,#00FH             ; Get only lower 4 bit
                CJNE    A,#00FH,COL2_DETECT     ; Check All rows '1'?
                AJMP    CHK_COL3

COL2_DETECT:    MOV     KPAD_DATA,#03       ; Initial KPAD_DATA = 2
                AJMP    GET_ROW             ; Jump to get row value

CHK_COL3:       SETB    KPAD_COL2           ; Stop Scan Column 1
                CLR     KPAD_COL3           ; Begin Scan Column 2
                MOV     A,P2                ; Get Port2 Value
                ANL     A,#00FH             ; Get only lower 4 bit
                CJNE    A,#00FH,COL3_DETECT     ; Check All rows '1'?
                RET                         ; All rows '1' => return

COL3_DETECT:    MOV     KPAD_DATA,#04       ; Initial KPAD_DATA = 4
GET_ROW:        CLR     KPAD_COL0           ; Enable all Column to find Crosspoint
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                CLR     KPAD_COL1
                CLR     KPAD_COL2
                CLR     KPAD_COL3

                JB      KPAD_ROW0,CHK_ROW1      ; Check Row 0 Detect?
                RET                             ; Row 0 Detect => return

CHK_ROW1:       JB      KPAD_ROW1,CHK_ROW2      ; Check Row 2 Detect?
                MOV     A,KPAD_DATA             ; Add 4 with KPAD_DATA
                ADD     A,#4
                MOV     KPAD_DATA,A
                RET                             ; Return

CHK_ROW2:       JB      KPAD_ROW2,CHK_ROW3      ; Check Row 2 Detect?
                MOV     A,KPAD_DATA             ; Add 8 with KPAD_DATA
                ADD     A,#8
                MOV     KPAD_DATA,A
                RET                             ; Return

CHK_ROW3:       MOV     A,KPAD_DATA             ; Add 12 with KPAD_DATA
                ADD     A,#12
                MOV     KPAD_DATA,A
                RET                             ; Return

;********SUB TABLE SHOW DISPLAY*******************
TB_NAME1:       DB      "       "

                DB      "REMOTE CONTROL CODE "
                DB      "READER PRESS ANY KEY"
                DB      "       "
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
TB_NAME2:       DB          " PROJECT 1&2/2543 "

TB_FUNC:        DB          " SELECT FUNCTION  "

TB_FKEY:        DB          " READ(F1)  SEND(F2) "

TB_KSEL:        DB          " PRESS COMMAND KEY  "

TB_BND:         DB          " SONY(F1)  JVC(F2)  "

TB_TIME:        DB          "  TIME BASE=13uS  "

TB_SAM:         DB          " SAMPLING PULSE(F2) "

TB_ONE:         DB          "PRESS ONE REMOTE KEY"

TB_QUIT:        DB          "   EXIT(F3)    "

TB_SELC1:       DB          " PLEASE SELECT KEY  "

TB_SELC2:       DB          "DPL(F1)SD(F2)EX(F3) "

TB_HI:          DB          "HI:"

TB_LRQ:         DB          "L(F1) R(F2) EXIT(F3)"

;-------------------------------------------------------

;**********SUB TABLE  REMOTE CONTROL'S DATA************
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;**************** DATA KEY FOR JVC ********************

JVC_K_1:    DB    23H,7DH,26H,7DH,23H,2BH,21H,2EH,21H,2EH,21H,2BH
            DB    21H,2BH,21H,2EH,21H,80H,23H,2BH,21H,2BH,21H,2BH
            DB    21H,2EH,21H,80H,26H,2BH,21H,2BH,21H,21H,00H

JVC_K_2:    DB    26H,7BH,26H,7DH,26H,2BH,23H,2BH,23H,2BH,23H,2BH
            DB    23H,2BH,23H,2BH,23H,2BH,23H,7DH,26H,28H,23H,2BH
            DB    23H,2BH,23H,7DH,26H,28H,23H,2BH,23H,23H,00H

JVC_K_3:    DB    26H,7DH,26H,7DH,23H,2BH,21H,2BH,21H,2BH,21H,2BH
            DB    21H,2BH,21H,2EH,21H,80H,23H,7DH,23H,2BH,21H,2BH
            DB    21H,2EH,21H,80H,26H,2BH,21H,2BH,21H,21H,00H

JVC_K_4:    DB    26H,7DH,26H,7AH,26H,28H,23H,2BH,21H,2BH,21H,2BH
            DB    21H,2BH,21H,2EH,21H,2BH,21H,2BH,21H,7DH,26H,2BH
            DB    21H,2BH,21H,7DH,26H,2BH,23H,2BH,23H,23H,00H

JVC_K_5:    DB    2BH,7AH,28H,7AH,28H,26H,21H,2BH,21H,30H,23H,30H
            DB    23H,30H,23H,30H,23H,7DH,26H,2BH,23H,7DH,26H,2BH
            DB    23H,2BH,23H,7DH,26H,2BH,23H,2BH,23H,23H,00H

JVC_K_6:    DB    26H,7DH,26H,7DH,26H,2BH,21H,2BH,21H,2BH,21H,2EH
            DB    21H,2EH,21H,2EH,21H,2BH,21H,80H,23H,7DH,26H,2BH
            DB    21H,2BH,21H,7DH,26H,23H,21H,2BH,21H,21H,00H

JVC_K_7:    DB    26H,7DH,26H,7DH,26H,28H,23H,2BH,23H,2BH,23H,2BH
            DB    23H,2BH,23H,2EH,23H,7DH,26H,7DH,26H,7DH,26H,28H
            DB    23H,2BH,23H,7DH,23H,28H,23H,2BH,23H,23H,00H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
JVC_K_8:    DB    26H,7DH,26H,7AH,26H,2BH,23H,2BH,23H,2BH,23H,2BH
            DB    23H,2BH,23H,2BH,23H,2BH,23H,2BH,23H,2BH,23H,7DH
            DB    26H,2BH,23H,7DH,26H,2BH,23H,2BH,23H,23H,00H

JVC_K_9:    DB    26H,7DH,26H,7AH,26H,28H,23H,28H,23H,2BH,23H,28H
            DB    23H,2BH,23H,2BH,23H,7DH,26H,28H,23H,2BH,23H,7DH
            DB    26H,28H,23H,7DH,26H,28H,23H,2BH,23H,23H,00H

JVC_K_0:    DB    26H,7DH,23H,7DH,23H,2BH,21H,2BH,21H,2BH,21H,2BH
            DB    21H,2BH,21H,2EH,21H,80H,23H,80H,23H,7DH,26H,2BH
            DB    23H,7DH,26H,2BH,23H,2BH,23H,2BH,23H,23H,00H

JVC_V_UP:   DB    26H,7DH,26H,7AH,26H,28H,23H,28H,23H,2BH,23H,2BH
            DB    23H,2BH,23H,2BH,23H,2BH,2BH,7DH,26H,7DH,26H,7DH
            DB    23H,7DH,26H,2BH,23H,2BH,23H,2BH,23H,23H,00H

JVC_V_DN:   DB    26H,7DH,26H,7AH,26H,28H,26H,28H,23H,2BH,23H,2BH
            DB    23H,2BH,23H,2BH,26H,7DH,26H,7DH,26H,7DH,26H,7DH
            DB    26H,7AH,26H,28H,26H,28H,23H,2BH,23H,23H,00H

;****************** DATA KEY FOR SONY ******************

SONY_K_1:   DB    0BAH,2EH,2BH,30H,2BH,30H,2BH,30H,2BH,30H,2BH,30H,2BH
            DB    30H,2BH,33H,5CH,30H,2BH,30H,2BH,30H,2BH,30H,2BH,00H  ;30...40

SONY_K_2:   DB    0BDH,2EH,5CH,30H.2BH,30H,2BH,30H,2BH,30H,2BH,30H,2BH
            DB    30H,2BH,30H,5CH,30H,2BH,30H,2BH,30H,2BH,30H,2BH,00H

SONY_K_3:   DB    0BAH,30H,28H,33H,59H,30H,2BH,30H,2BH,30H,2BH,30H,2BH
            DB    30H,2BH,33H,5CH,30H,2BH,30H,2BH,30H,2BH,30H,2BH,00H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SONY_K_4:   DB    0BDH,30H,59H,30H,5CH,30H,2BH,30H,2BH,30H,2BH,30H,2BH
            DB    30H,2BH,33H,5CH,30H,2BH,30H,2BH,30H,2BH,30H,2BH,00H

SONY_K_5:   DB    0BDH,2EH,2EH,30H,2BH,30H,5CH,30H,2BH,30H,2BH,30H,2BH
            DB    30H,2EH,33H,5CH,30H,2BH,30H,2BH,30H,2BH,30H,2EH,00H

SONY_K_6:   DB    0BAH,2EH,5CH,30H,2EH,2EH,5CH,30H,2BH,30H,2BH,30H,2BH
            DB    30H,2BH,33H,5CH,30H,2BH,30H,2BH,30H,2BH,30H,2BH,00H

SONY_K_7:   DB    0BAH,2EH,2BH,30H,5CH,30H,5CH,30H,2BH,30H,2BH,30H,2BH
            DB    30H,2BH,33H,5CH,30H,2BH,30H,2BH,30H,2BH,30H,2BH,00H

SONY_K_8:   DB    0BAH,2EH,5CH,30H,5CH,30H,5CH,30H,2BH,30H,2BH,30H,2BH
            DB    30H,2BH,33H,5CH,30H,2BH,30H,2BH,30H,2BH,30H,2BH,00H

SONY_K_9:   DB    0BDH,2BH,2EH,2EH,2EH,2EH,2EH,2EH,5EH,2EH,2EH,2EH,30H
            DB    2BH,2EH,30H,5EH,2EH,2EH,2EH,30H,2BH,2EH,2EH,2BH,00H

SONY_K_0:   DB    0BAH,2EH,5CH,30H,2BH,30H,59H,30H,2EH,2EH,5CH,30H,2EH
            DB    2EH,2BH,30H,5CH,30H,2EH,30H,2EH,2EH,2BH,30H,2BH,00H

SONY_V_UP:  DB    0BDH,2EH,2BH,30H,5CH,30H,2BH,30H,2EH,30H,5CH,30H,2BH
            DB    30H,2BH,33H,5CH,30H,2BH,30H,2BH,30H,2BH,30H,2BH,00H

SONY_V_DN:  DB    0BAH,33H,57H,35H,57H,33H,28H,33H,28H,35H,57H,33H,28H
            DB    33H,28H,35H,57H,33H,28H,33H,28H,33H,28H,33H,28H,00H

            END
```

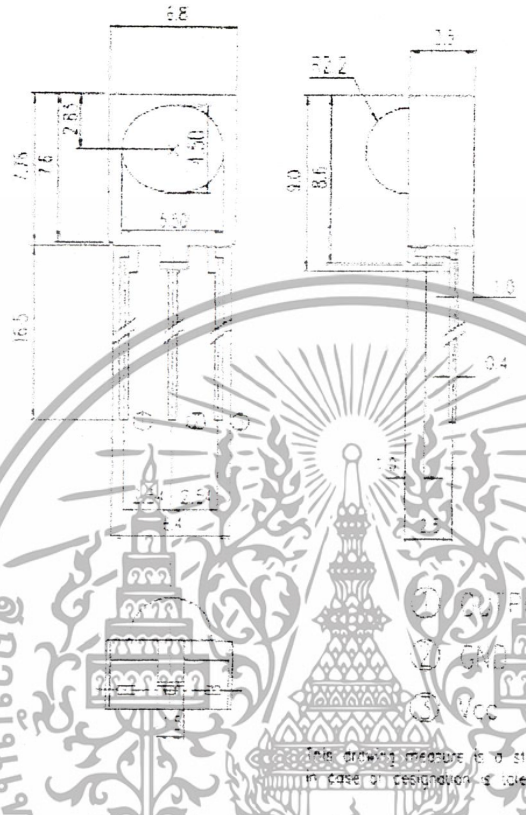
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MODEL NO. IRM8610S

■ PACKAGE DIMENSIONS:



UNIT : mm

ABOVE SPECIFICATION MAY BE CHANGED WITHOUT NOTICE. SUPPLIER'S WILL RESERVE AUTHORITY ON MATERIAL CHANGE FOR ABOVE SPECIFICATION.

Description :

The device is a miniature type infrared remote control system receiver which has been developed and designed by utilizing the most updated IC technology. PIN diode and preamplifier are assembled on lead frame, the epoxy package is designed as IR filter. The demodulated output signal can directly be decoded by a microprocessor.

Feature

- Low voltage and low power consumption .
- Photodiode with integrated circuit.
- High sensitivity.
- TTL and CMOS compatibility.
- High immunity against ambient light.
- High protection ability to EMI and metal case can be customized.
- Long reception distance.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Application :

O Light detecting portion of remote control

- TVs
- VCRs
- Audio Equipment
- Air Conditioners
- CATV set top boxes
- Electric Fan
- Multi-media Equipment

O Optical switch

Performance

The specified electro-optical characteristics is satisfied under the following conditions at the controllable distance.

①Measurement place

A place that is nothing of extreme light reflect in the room.

②External light

Project the light of ordinary white fluorescent lamps which are not high frequency lamps and must be less then 10 Lx at the module surface.

③Standard transmitter

A transmitter whose output is so adjusted as to Vo=600mVp-p and the output waveform shown in Fig.-1.According to the measurement method shown in Fig.-2 is specified as the standard transmitter.

However , the infrared photodiode to be used for the transmitter should be λp=940nm,Δλ=50nm.

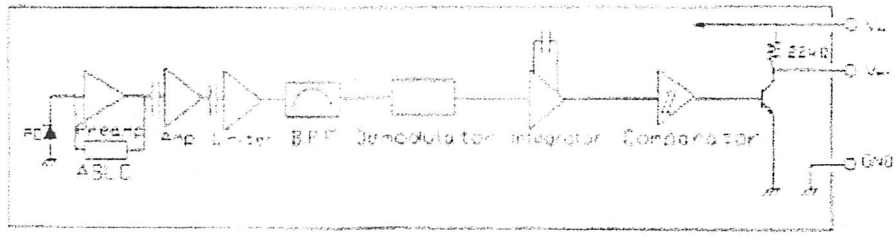
Also, photodiode is used of PD438B(Vr=5V)

(Standard light/ Light source temperature 2856°K)

④Measuring system

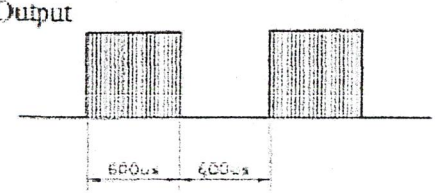
According to the measuring system shown in Fig.-3

■ Block Diagram

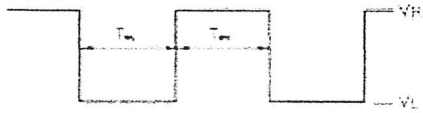


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Transmitter Output



D.U.T output Pulse



Absolute maximum ratings (Ta=25 ℃)

Parameter	Symbol	Ratings	Unit	Notice
Supply Voltage	Vcc	0 ~ 6.5	V	
Operating Temperature	Topr	-30 ~ +85	℃	
Storage Temperature	Tstg	-40 ~ +85	℃	
Soldering Temperature	Tsol	260	℃	4mm from mold body less than 5 seconds

Electro Optical Characteristics (Ta=25 ℃)

Parameter	Symbol	MIN	TYP	MAX	Unit	Condition
Supply Voltage	Vcc	4.5	5	5.5	V	DC voltage
Supply Current	Icc	-	-	3	mA	No signal input
B.P.F Center Frequency	fo	-	38	-	KHz	
Peak Wavelength	λ p	-	940	-	nm	
Reception Distance	d	8	-	-	m	At the ray axis *1
		4				
Half Angle (Horizontal)	θ h	-	45	-	deg	
Half Angle (Vertical)	θ v	-	35	-	deg	
High Level Pulse Width	T_H	400	-	800	μs	
Low Level Pulse Width	T_L	400	-	800	μs	
High Level Output Voltage	V_H	4.5	-	-	V	
Low Level Output Voltage	V_L	-	0.2	0.5	V	

*1:The ray receiving surface at a vertex and relation to the ray axis in the range of ϕ= 45° and ϕ= 0.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
---[ SONY CD Infrared Remote Control codes (RM-DX55) ]----------------
----------
'
'
'                Hex code       Binary code
'                                msb          lsb
'          power = &HA91      : 1010 1001 0001
'           play = &H4D1      : 0100 1101 0001
'           stop = &H1D1      : 0001 1101 0001
'          pause = &H9D1      : 1001 1101 0001
'       continue = &HB91      : 1011 1001 0001
'        shuffle = &HAD1      : 1010 1101 0001
'        program = &HF91      : 1111 1001 0001
'           disc = &H531      : 0101 0011 0001
'              1 = &H011      : 0000 0001 0001
'              2 = &H811      : 1000 0001 0001
'              3 = &H411      : 0100 0001 0001
'              4 = &HC11      : 1100 0001 0001
'              5 = &H211      : 0010 0001 0001
'              6 = &HA11      : 1010 0001 0001
'              7 = &H611      : 0110 0001 0001
'              8 = &HE11      : 1110 0001 0001
'              9 = &H111      : 0001 0001 0001
'              0 = &H051      : 0000 0101 0001
'            >10 = &HE51      : 1110 0101 0001
'          enter = &HD11      : 1101 0001 0001
'          clear = &HF11      : 1111 0001 0001
'        repeate = &H351      : 0011 0101 0001
'         disc - = &HBD1      : 1011 1101 0001
'         disc + = &H7D1      : 0111 1101 0001
'             |<< = &H0D1      : 0000 1101 0001
'             >>| = &H8D1      : 1000 1101 0001
'              << = &HCD1      : 1100 1101 0001
'              >> = &H2D1      : 0010 1101 0001


'---[ SONY Cassette Infrared Remote Control codes (RM-J901) ]--------
-----------
'Deck A:
'           stop = &H1C1      : 0001 1100 0001
'         play > = &H4C1      : 0100 1100 0001
'         play < = &HEC1      : 1110 1100 0001
'             >> = &H2C1      : 0010 1100 0001
'             << = &HCC1      : 1100 1100 0001
'         record = &H6C1      : 0110 1100 0001
'          pause = &H9C1      : 1001 1100 0001
'Deck B:
'           stop = &H18E      : 0001 1000 1110
'         play > = &H58E      : 0101 1000 1110
'         play < = &H04E      : 0000 0100 1110
'             >> = &H38E      : 0011 1000 1110
'             << = &HD8E      : 1101 1000 1110
'         record = &H78E      : 0111 1000 1110
'          pause = &H98E      : 1001 1000 1110
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
---[ SONY TV Infrared Remote Control codes (RM-694) ]----------------
----------
'
'    program + = &H090     : 0000 1001 0000
'    program - = &H890     : 1000 1001 0000
'     volume + = &H490     : 0100 1001 0000
'     volume - = &HC90     : 1100 1001 0000
'        power = &HA90     : 1010 1001 0000
' sound on/off = &H290     : 0010 1001 0000
'            1 = &H010     : 0000 0001 0000
'            2 = &H810     : 1000 0001 0000
'            3 = &H410     : 0100 0001 0000
'            4 = &HC10     : 1100 0001 0000
'            5 = &H210     : 0010 0001 0000
'            6 = &HA10     : 1010 0001 0000
'            7 = &H610     : 0110 0001 0000
'            8 = &HE10     : 1110 0001 0000
'            9 = &H110     : 0001 0001 0000
'            0 = &H910     : 1001 0001 0000
'         -/-- = &HB90     : 1011 1001 0000
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้