อัลกอลิทึมถอดรหัสสัญญาณแสงอินฟราเรดจากรีโมทควบคุมอุปกรณ์ไฟฟ้า

An Infrared Pulse Signal Decoding for Appliance Remote Controller

นาย คมสันต์ อินต๊ะ รหัส 45380013

นาย นิรันทร์ชัย ทินนา รหัส 45380067

นาย เสกสรร อินทร์ทรีย์ รหัส 45380181

I507550b

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ 15 พ.ย. 2549
เลขทะเบียน 4900158
เลขเรียกหนังสือ ปร. ค152อ 2548
มหาวิทยาลัยนเรศวร

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2548

ใบรับรองโครงงานวิศวกรรม

หัวข้อโครงงาน อัลกอลิทึมถอดรหัสสัญญาณแสงอินฟราเรดจากรีโมทควบคุมอุปกรณ์ไฟฟ้า

ผู้ดำเนินโครงงาน

นายคมสันต์	อินต๊ะ	รหัส 45380013
นายนิรันทร์ชัย	ทินนา	รหัส 45380067
นายเสกสรร	อินทร์ทรีย์	รหัส 45380181

อาจารย์ที่ปรึกษา ดร.อัครพันธ์ วงศ์กังแห

สาขาวิชา วิศวกรรมไฟฟ้า

ภาควิชา วิศวกรรมไฟฟ้าและคอมพิวเตอร์

ปีการศึกษา 2548

.....

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้โครงงานฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะกรรมการสอบโครงงานวิศวกรรม

.....ประธานกรรมการ
(ดร.อัครพันธ์ วงศ์กังแห)

.....กรรมการ
(ดร.สมยศ เกียรติวนิชวิไล)

.....กรรมการ
(ดร.สมพร เรืองสินชัยวานิช)

หัวข้อโครงงาน	อัลกอลิทึมถอดรหัสสัญญาณแสงอินฟราเรดจากรีโมทควบคุมอุปกรณ์ไฟฟ้า		
ผู้ดำเนินโครงงาน	นายคมสันต์	อินต๊ะ	รหัส 45380013
	นายนิรันทร์ชัย	ทินนา	รหัส 45380067
	นายเสกสรร	อินทร์ทรีย์	รหัส 45380181
อาจารย์ที่ปรึกษา	ดร.อัครพันธ์	วงศ์กังแห	
สาขาวิชา	วิศวกรรมไฟฟ้า		
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์		
ปีการศึกษา	2548		

.....

บทคัดย่อ

ปัจจุบันเทคโนโลยีของโลกเราได้พัฒนาไปอย่างรวดเร็ว ทำให้คนเรามีความ**สะดวกสบายมากยิ่งขึ้น** โครงงานนี้เป็นการนำเสนอเกี่ยวกับการใช้แสงอินฟราเรด ซึ่งเป็นการศึกษาเกี่ย**วกับหลักการและการถอดรหัส**จากสัญญาณอินฟราเรดที่ได้จากตัวรีโมททั่วๆไป เช่น รีโมทโทรทัศน์ **รีโมท**วีดีโอ เป็นต้น

ผลที่ได้จากโครงงานนี้ คือ โปรแกรมการถอดรหัสสัญญาณของรีโมทอินฟรา**เรดอันหนึ่งซึ่งใช้กับไม**โครคอนโทรลเลอร์ของ Intel 8051

Project title	An Infrared Pulse Signal Decoding for Appliance Remote Controller		
Name	Mr. Khomson	Inta	ID. 45380013
	Mr.Nirunchai	Thinna	ID. 45380067
	Mr.Sekson	Insee	ID. 45380181
Project advisor	Dr. Akaraphunt	Vongkunghae	
Major	Electrical Engineering		
Department	Electrical and Computer Engineering		
Academic year	2005		

.....

Abstract

Presently , the Infrared Remote Controllers are used for controlling many appliances. The use of there are in created everyday. In this project , we presents a method and a program for decoding the infrared signal from the remote controller .

The Intel 8051 Microcontroller is our target device to run the decoding program.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้เป็นการศึกษาเกี่ยวกับเรื่องการถอดรหัสสัญญาณแสงอินฟราเรดจากรีโมท ซึ่งจะไม่มีทางสำเร็จไปได้ถ้าไม่ได้รับความช่วยเหลือจากบุคคลดังต่อไปนี้

ขอขอบพระคุณ อาจารย์อัครพันธ์ วงศ์กังแห อาจารย์ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ มหาวิทยาลัยนเรศวร ผู้เป็นอาจารย์ที่ปรึกษาโครงงานที่ได้ให้ความรู้ ให้คำแนะนำและให้ความช่วยเหลือแก่คณะผู้จัดทำเป็นอย่างดีตลอดมา

ขอขอบพระคุณ อาจารย์ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ มหาวิทยาลัยนเรศวรทุกท่านที่ได้ให้ความรู้และให้คำสั่งสอนจนคณะผู้จัดทำมีความรู้ที่สามารถนำมาประยุกต์ใช้ในการจัดทำโครงงานในครั้งนี้

และที่สำคัญที่สุดขอขอบพระคุณบิดา มารดา ที่ได้เลี้ยงดูและอบรมสั่งสอนแก่คณะผู้จัดทำจนทำให้คณะผู้จัดทำทุกคนมีวันนี้ได้ ซึ่งเป็นพระคุณอันหาที่เปรียบไม่ได้

ท้ายนี้คณะผู้จัดทำใคร่ขอขอบพระคุณผู้ที่มีส่วนเกี่ยวข้องทุกท่านที่ไม่ได้กล่าวนามมา ณ ที่นี้ ที่มีส่วนร่วมในการให้ข้อมูล เป็นที่ปรึกษาในการทำปริญญานิพนธ์ฉบับนี้จนเสร็จสมบูรณ์ คณะผู้จัดทำจึงขอขอบพระคุณไว้ ณ ที่นี้

คณะผู้จัดทำโครงงาน

นายคมสันต์ อินต๊ะ รหัส 45380013

นายนิรันทร์ชัย ทินนา รหัส 45380067

นายเสกสรร อินทร์ทรีย์ รหัส 45380181

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....ก

บทคัดย่อภาษาอังกฤษ.....ข

กิตติกรรมประกาศ..... .ค

สารบัญ.....ง

สารบัญตาราง.....ฉ

สารบัญรูป.....ช

บทที่ 1 บทนำ

1.1 ที่มาและความสำคัญของโครงงาน.....1

1.2 วัตถุประสงค์.....1

1.3 ขอบข่ายงาน.....1

1.4 ขั้นตอนในการดำเนินงาน.....2

1.5 ผลที่คาดว่าจะได้รับ.....2

1.6 งบประมาณ.....2

บทที่ 2 หลักการและทฤษฎีเบื้องต้น

2.1 หลักการทำงานของรีโมท.....3

2.2 การสื่อสารอนุกรม (Serial Communication).....4

2.2.1 รูปแบบของการสื่อสารอนุกรม.....5

2.3 อัตราการรับส่งข้อมูล.....6

2.4 การสื่อสารไร้สาย (Wireless Communication).....6

2.5 Infrared : ลำแสงสีแดง.....6

บทที่ 3 วิธีการดำเนินงาน

3.1 การวัดสัญญาณของรีโมทอินฟราเรด.....7

3.2 การจับสัญญาณในไมโครคอนโทรลเลอร์ (TIMER).....8

3.2.1แบบวิธีของตัวจับเวลาและตัวบ่งชี้การล้น.....10

3.2.2 การเลือกใช้งานตัวจับเวลาและตัวนับ.....12

สารบัญ(ต่อ)

หน้า

3.2.3 ตัวจับเวลา (Timer) 12

3.2.4 ตัวนับ (Counter) 12

3.3 การต่อหลอด LED 7 Segment.....13

บทที่ 4 การทดลองและผลการทดลอง

4.1 การวัดสัญญาณของรีโมทอินฟราเรด.....15

4.2 โปรแกรมการตรวจจับสัญญาณรีโมท.....22

4.2.1 แผนภาพการเขียนโปรแกรมหลัก 24

4.2.2 แผนภาพการเขียนโปรแกรมฟังก์ชัน GETCODE.....25

4.2.3 แผนภาพการเขียนโปรแกรมฟังก์ชันCONVERT.....26

4.2.4 แผนภาพการเขียนโปรแกรมฟังก์ชัน CUTARRAY.....27

4.2.5 แผนภาพการเขียนโปรแกรมฟังก์ชัน CHECKCODE.....28

4.2.6 โปรแกรมที่ใช้ในการแสดงผลการตรวจจับสัญญาณรีโมท.....29

บทที่ 5 สรุปผลและการวิเคราะห์

5.1 สรุปผลการปฏิบัติงาน.....32

5.2 ข้อเสนอและแนวทางแก้ไข.....33

5.3 ข้อเสนอแนะในการพัฒนา.....33

เอกสารอ้างอิง.....34

ภาคผนวก.....35

ประวัติผู้เขียนโครงงาน.....55

สารบัญตาราง

ตารางที่ หน้า

3.1 รีจิสเตอร์ฟังก์ชันเฉพาะตัวจับเวลา (Timer Special Function Registers)..... .8

3.2 บิตต่าง ๆ รีจิสเตอร์ TMOD.....9

3.3 แบบวิธีของตัวจับเวลา (Timer Mode).....9

3.4 รูปแบบการแสดงผลและค่าของพอร์ต.....14

มหาวิทยาลัยนเรศวร

สารบัญรูป

รูปที่ หน้า

2.1 Waveform ของสัญญาณของรีโมท.....3

2.2 วงจรการทำงานของรีโมท 4

2.3 การรับส่งข้อมูลแบบ Simplex, Half duplex และ Full duplex 5

3.1 วงจรของ IR Remote 7

3.2 ตัวส่งสัญญาณรีโมทและตัวรับสัญญาณรีโมท 7

3.3 การทำงานของตัวจับเวลา/ตัวนับในแบบวิธีต่าง ๆ.....11

3.4 แหล่งจ่ายสัญญาณนาฬิกากับการเลือกตัวจับเวลา/ตัวนับ.....13

3.5 การต่อ LED 7 เซกเมนต์แบบแอโนดร่วมกับ P1.....14

4.1 การวัดสัญญาณจากวงจรของ IR Remote.....15

4.2 สัญญาณรีโมทจากการกดปุ่มหมายเลข 1.....16

4.3 สัญญาณรีโมทจากการกดปุ่มหมายเลข 2.....16

4.4 สัญญาณรีโมทจากการกดปุ่มหมายเลข 3.....17

4.5 สัญญาณรีโมทจากการกดปุ่มหมายเลข 4.....17

4.6 สัญญาณรีโมทจากการกดปุ่มหมายเลข 5.....18

4.7 สัญญาณรีโมทจากการกดปุ่มหมายเลข 6.....18

4.8 สัญญาณรีโมทจากการกดปุ่มหมายเลข 7.....19

4.9 สัญญาณรีโมทจากการกดปุ่มหมายเลข 8.....19

4.10 สัญญาณรีโมทจากการกดปุ่มหมายเลข 9.....20

4.11 สัญญาณรีโมทจากการกดปุ่มหมายเลข 0.....20

4.12 การต่อตัวรับเข้าสัญญาณรีโมทเข้ากับขา P 3.4.....21

4.13 โปรแกรม KEIL.....22

4.14 การลงโปรแกรม HEXSEND.....23

4.15 แผนภาพการเขียนโปรแกรมหลัก.....24

4.16 แผนภาพการเขียนโปรแกรมฟังก์ชัน GETCODE..... 25

4.17 แผนภาพการเขียนโปรแกรมฟังก์ชัน CONVERT 26

สารบัญรูป(ต่อ)

รูปที่ หน้า

4.18 แผนภาพการเขียนโปรแกรมฟังก์ชัน CUTARRAY 27
4.19 แผนภาพการเขียนโปรแกรมฟังก์ชัน CHECKCODE 28
4.20 โปรแกรมที่ใช้ในการแสดงผลการตรวจจับสัญญาณรีโมท.....29
4.21 โปรแกรม HYPER TERMINAL.....29
4.19 การแสดงผลของโปรแกรมจากการกดรีโมทหมายเลข 0 – 9.....30
4.20 การต่อบอร์ดไมโครคอนโทรเลอร์กับ 7 –SEGMENT.....30
4.21 การต่อบอร์ดไมโครคอนโทรเลอร์กับ 7-SEGMENT.....31
4.22 การแสดงผลของการส่งสัญญาณรีโมทออกทาง 7 –SEGMENT.....31

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญของโครงงาน

ปัจจุบันเทคโนโลยีของโลกเราได้พัฒนาไปอย่างรวดเร็ว ทำให้คนเรามีความสะดวกสบายมากยิ่งขึ้น จึงทำให้เกิดการแข่งขันกันของบริษัทต่างๆเพื่อประดิษฐ์เครื่องอำนวยความสะดวกต่างๆกันเกิดขึ้นโดยเฉพาะปัจจุบันจะเห็นได้ว่าอุปกรณ์ต่างๆ เช่น แอร์ โทรทัศน์ พัดลม ฯลฯ เป็นต้น จะมีรีโมทเป็นตัวสั่งการให้ทำงานซึ่งส่งผลให้เกิดความสะดวกสบายมากขึ้น

โครงงานนี้คณะผู้จัดทำได้ศึกษาเกี่ยวกับการรับส่งสัญญาณด้วยแสงอินฟราเรด หลักการทำงานของการรับส่งสัญญาณด้วยแสงอินฟราเรด และวิธีการรับส่งข้อมูลของรีโมทที่ใช้แสงอินฟราเรดเป็นตัวรับส่งสัญญาณ เพื่อนำมาประยุกต์ใช้งานกับอุปกรณ์ต่างๆ เพื่อเป็นสิ่งอำนวยความสะดวกให้แก่ผู้ใช้งาน ต่อไป

1.2 วัตถุประสงค์ของโครงงาน

1.2.1 เพื่อศึกษาหลักการทำงานของรีโมทที่มีการรับส่งสัญญาณด้วยแสงอินฟราเรด

1.2.2 เพื่อศึกษาการเขียนโปรแกรม ในการติดต่อกับวงจรรับสัญญาณพัลส์

1.2.3 เพื่อนำความรู้ที่ได้มาประยุกต์ใช้งานกับอุปกรณ์ต่างๆ

1.2.4 เพื่อนำความรู้ที่ได้มาพัฒนารีโมทให้สามารถใช้งานได้กับทุกๆอุปกรณ์

1.2.5 เพื่อนำความรู้ที่ได้มาเผยแพร่ให้กับบุคคลที่สนใจเพื่อศึกษาและพัฒนาต่อไป

1.3 ขอบเขตของโครงงาน

1.3.1 ศึกษาทฤษฎีเกี่ยวกับสิ่งเหล่านี้ ได้แก่

- หลักการการรับส่งสัญญาณของรีโมทด้วยแสงอินฟราเรด
- การเขียนโปรแกรม ในการติดต่อกับวงจรรับสัญญาณพัลส์
- หลักการทำงานและวิธีใช้งานของไมโครคอนโทรลเลอร์
- การสื่อสารข้อมูลผ่านแสงอินฟราเรด

1.3.2 เขียนโปรแกรมเพื่อส่งและรับค่ากับพอร์ตของไมโครคอนโทรลเลอร์

1.3.3 ทดสอบการเขียนโปรแกรมและสรุปผลใช้ทำงาน

1.3.4 สรุปผลและนำเสนอ

1.4 ขั้นตอนการดำเนินการ

หัวข้องาน	ตุลาคม	พฤศจิกายน	ธันวาคม	มกราคม	กุมภาพันธ์	มีนาคม	เมษายน
1. ศึกษาหลักการทำงานของรีโมทและศึกษาการสื่อสารข้อมูลผ่านแสงอินฟราเรด	←	→					
2. ศึกษาหลักการทำงานและวิธีใช้งานของไมโครคอนโทรลเลอร์			←	→			
3. เพื่อศึกษาการเขียนโปรแกรม ในการติดต่อกับ Hardware				←	→		
4. เขียนและทดสอบโปรแกรม					←	→	
5. สรุป และ วิเคราะห์ผล							← →

1.5 ผลที่คาดว่าจะได้รับ

1.5.1 มีความรู้ความเข้าใจในหลักการและวิธีการรับส่งสัญญาณด้วยแสงอินฟราเรด

1.5.2 มีความรู้ความเข้าใจในหลักการและวิธีการใช้งานของไมโครคอนโทรลเลอร์

1.5.3 มีความรู้ความเข้าใจในการเขียนโปรแกรม ติดต่อกับวงจรรับสัญญาณพัลส์

1.5.4 ได้โปรแกรมเพื่อส่งและรับค่ากับพอร์ตของไมโครคอนโทรลเลอร์

1.5.5 เป็นแนวทางในการพัฒนาใช้โปรแกรม ในการวิเคราะห์การสื่อสารผ่านทางพอร์ตของไมโครคอนโทรลเลอร์

1.6 งบประมาณที่ใช้

1.6.1 ค่าหนังสือประกอบการทำโครงงาน เป็นเงิน 1,000 บาท

1.6.2 ค่าปริ้นท์งาน เป็นเงิน 1,000 บาท

1.6.3 ค่าใช้จ่ายอื่นๆ เป็นเงิน 1,000 บาท

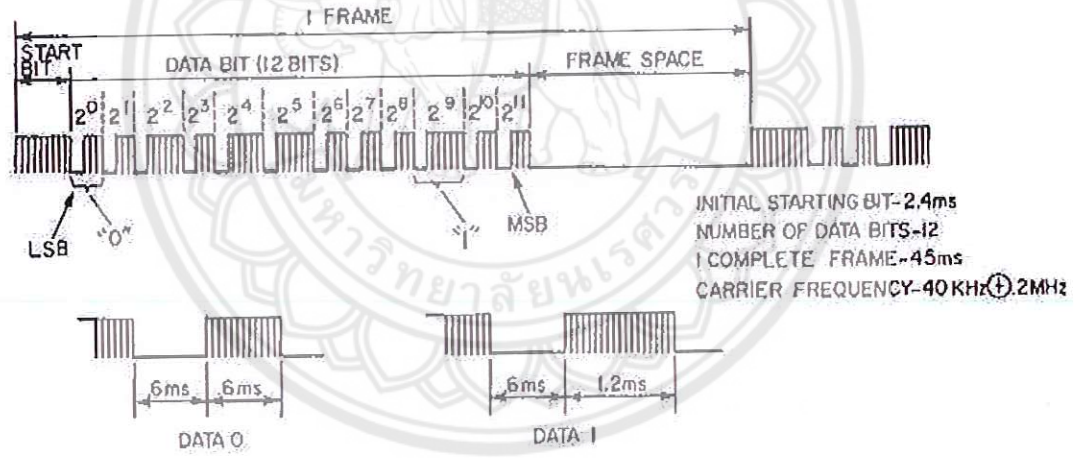
รวมเป็นเงินทั้งสิ้น 3,000 บาท

บทที่ 2
หลักการและทฤษฎีเบื้องต้น

2.1 หลักการทำงานของรีโมท

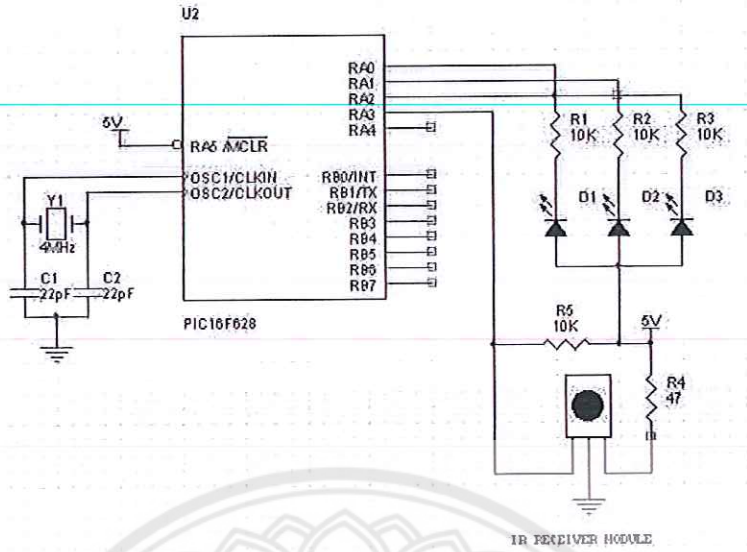
การสื่อสารข้อมูลกับอุปกรณ์ภายนอกแบบไร้สายเป็นอีกวิธีหนึ่งตัวอย่างการประยุกต์ใช้งานไมโครคอนโทรลเลอร์ที่น่าสนใจโดยเฉพาะอย่างยิ่งผ่านแสงอินฟราเรดรีโมทคอนโทรลโดยทั่วไปในปัจจุบัน เมื่อเรากดปุ่มบนตัว Remote จะเกิดความถี่เป็น Pulse ต่อเนื่อง (Continuous Stream) ที่มีความถี่ 37.9 KHz ถูกส่งออกมาโดยมอดดูเลทในรูปแสงอินฟราเรด Pulse นี้เมื่อผ่านตัวรับ (TV, VCR, VCD เป็นต้น) ก็จะถูกแปลงรหัสให้เป็นคำสั่งตามที่ได้โปรแกรมไว้ในอุปกรณ์นั้นๆ

การรับสัญญาณ จาก Remote นั้นต้องเข้าใจลักษณะรูปแบบของสัญญาณที่ส่งออกมาจาก Remote ก่อนซึ่งตัวอย่าง Waveform ของสัญญาณของ Remote ที่ใช้ในปริญญานิพนธ์นี้จะมีลักษณะดังรูป



รูปที่ 2.1 Waveform ของสัญญาณของรีโมท[5]

รูปที่ 2.2 คือ วงจรตัวอย่างอันหนึ่งที่มีหลักการทำงานก็คือจะมี output ที่ไปขับ LED ทั้งหมด 3 ดวง โดยควบคุมผ่าน Remote หมายเลข 1-3 ลักษณะคือ กดหมายเลข 1 หนึ่งครั้ง LED ที่ดวง 1 ก็จะติด ถ้ากดหมายเลข 1 อีกครั้ง LED ที่ ดวง 1 ก็จะดับ ส่วนดวง 2-3 ที่เหลือก็จะมีหลักการเช่นเดียวกัน



รูปที่ 2.2 วงจรการทำงานของรีโมท [5]

ในการส่งข้อมูลสัญญาณ จะมี Carrier Frequency ขนาด 40 KHz ร่วมไปกับข้อมูล โดยที่ข้อมูล LSB จะถูกส่งออกมาก่อนนั้นก็คือจะส่ง Command บิตต่ำสุดออกมาก่อนแล้วส่งต่อๆกันออกมาจนครบ 12 bits เราเรียกลักษณะแบบนี้ว่า Reverse Order เช่น Remote ของ TV จะมีรหัสอุปกรณ์(Device Code) เป็น 00001 ถ้าเรากดปุ่มเปิดปิด(Power Switch)ซึ่งมี Command ID เป็น 0010101 ข้อมูลทั้งหมดก็จะเป็น 000010010101 เมื่อจะส่งก็จะเริ่มด้วย สัญญาณ Start ขนาด 2.4 mSec จากนั้นก็จะตามด้วยข้อมูลขนาด 12 bit (Device 5 bits + Command 7 bits) ที่ถูกส่งแบบ Reverse Order ก็จะเป็น 101010010000 ลักษณะสัญญาณของข้อมูล " 0 " คือ จะมีสัญญาณ low นาน 600 us และ สัญญาณ high นาน 600 us ส่วนข้อมูล " 1 " จะมีสัญญาณ Low นาน 600 us และ สัญญาณ High นาน 1.2 ms หากมีการกดค้างชุด Package ข้อมูลเดิมกับชุด ข้อมูลใหม่ จะห่างกัน 25 ms

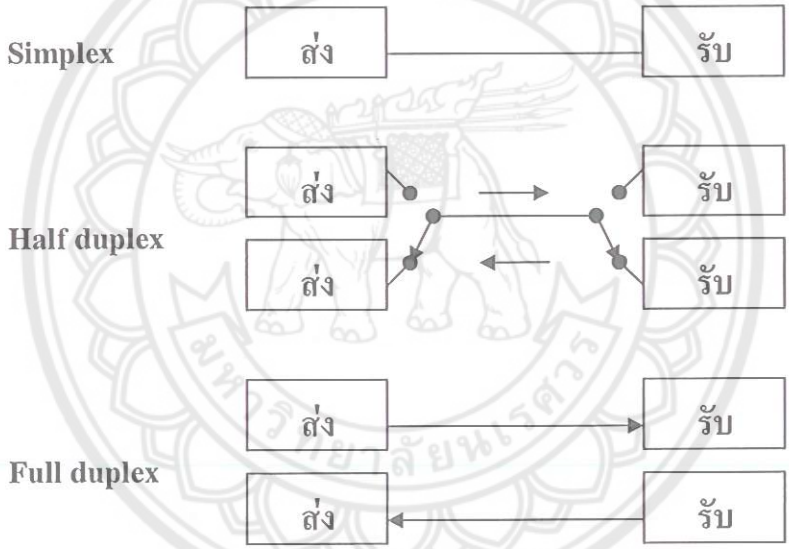
2.2 การสื่อสารอนุกรม (Serial Communication)

เมื่อไมโครโปรเซสเซอร์ต้องการติดต่อกับอุปกรณ์ภายนอก ตัวมันจะส่งข้อมูลออกมามีขนาดเป็นไบต์ หรือ 8 บิตแต่เนื่องจากไมโครโปรเซสเซอร์ มีบัสข้อมูลขนาด 8 บิต การถ่ายโอนข้อมูลต่างๆ จะทำแบบขนาน ถ้าต้องการส่งข้อมูลออกไปอนุกรมจะต้องเปลี่ยนข้อมูลแบบขนานนี้ ให้เป็นอนุกรมเสียก่อนแล้วจึงส่งออกไป ส่วนการรับข้อมูลแบบอนุกรมนี้จะรับข้อมูลเข้ามาครั้งละบิตและเปลี่ยนข้อมูลให้เป็นข้อมูลแบบขนานเพื่อส่งให้ ไมโครโปรเซสเซอร์ไปประมวลผลต่อไป ในปริญญานิพนธ์นี้ข้อมูลแบบอนุกรมจาก

หัวอ่านสัญญาณอินฟราเรดจะถูกแปลเป็นข้อมูลแบบขนานโดยใช้กระบวนการทางซอฟแวร์จัดรูปของข้อมูลจากแบบอนุกรมให้เป็นแบบขนานตามจำนวนบิตที่ต้องการ ซึ่งรายละเอียดจะกล่าวในบทต่อไป

2.2.1 รูปแบบของการสื่อสารอนุกรม

การสื่อสารข้อมูลระหว่างตัวรับและตัวส่งนั้นมีหลายวิธี ถ้าหากตัวส่งทำหน้าที่ส่งอย่างเดียวและตัวรับทำหน้าที่รับอย่างเดียวจะเรียกว่าการสื่อสารแบบ ซิมเพล็กซ์ (simplex) เช่น การส่งข้อมูลระหว่างคอมพิวเตอร์กับเครื่องพิมพ์ แต่ถ้าหากตัวรับและตัวส่งสามารถรับและส่งข้อมูลได้แต่ทำในเวลาต่างกัน เรียกว่า การสื่อสารแบบ ฮาล์ฟดูเพล็กซ์ (half duplex) เช่น วิทยุสื่อสาร แต่ถ้าหากตัวรับและตัวส่งสามารถรับและส่งข้อมูลได้สองทิศทางในเวลาเดียวกันจะเรียกว่า การสื่อสารแบบฟูลล์ดูเพล็กซ์ (full duplex) เช่น โทรศัพท์เคลื่อนที่ เป็นต้น การสื่อสารแต่ละแบบแสดงดังรูปที่ 2.3



รูปที่ 2.3 การรับส่งข้อมูลแบบ Simplex, Half duplex และ Full duplex [1]

สำหรับวิธีในการส่งข้อมูลแบบอนุกรม มีสองวิธีคือ การส่งแบบเข้าจังหวะเวลา(Synchronous) และการส่งแบบไม่เข้าจังหวะเวลา (Asynchronous) โดยการส่งแบบเข้าจังหวะเวลา (Synchronous) จะต้องมีสัญญาณนาฬิการ่วมไปด้วยเพื่อควบคุมการรับส่งข้อมูล แต่ การส่งแบบไม่เข้าจังหวะเวลา (Asynchronous) จะใช้การกำหนดอัตราเร็วในการรับส่งข้อมูลให้มีค่าเท่ากันเรียกว่า อัตราบอด หรือ บอดเรด (baud rate) ซึ่งมีหน่วยเป็นบิตต่อวินาที (bit per second: bps) [1]

2.3 Infrared : ลำแสงสีแดง

แสงอินฟราเรด(Infrared : IR) ระบบการสื่อสารไร้สาย (wireless) อีกรูปแบบหนึ่ง แสงอินฟราเรด (Infrared : IR) คือส่วนหนึ่งของสเปกตรัมแม่เหล็กไฟฟ้า ในย่านความถี่ของแสงต่ำกว่าแสงสีแดง ที่ตาของคนเราไม่สามารถมองเห็นเป็นลำแสงอีกชนิดหนึ่งที่ถูกนำมาใช้เพื่อการสื่อสารในระยะใกล้ โดยมีระยะในการส่งสัญญาณ 30 -80 ฟุต

คุณสมบัติเด่นของ Infrared

- คลื่นสั้น ทางเดินของแสงเป็นแนวตรง
- ราคาถูก
- ง่ายต่อการผลิต
- ไม่สามารถทะลุผ่านวัตถุ ทำให้สามารถติดตั้ง Infrared ในห้องทำงานติดกันได้

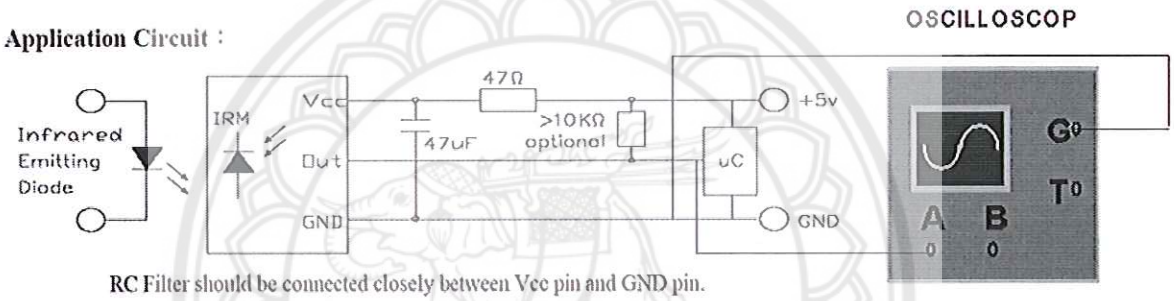
ตัวอย่างอุปกรณ์ที่ใช้ Infrared

- Remote Control ของโทรทัศน์
- โทรศัพท์มือถือ
- คอมพิวเตอร์
- เครื่องพิมพ์
- PDA, Palm

บทที่ 3
วิธีการดำเนินการ

3.1 การวัดสัญญาณของรีโมทอินฟราเรด

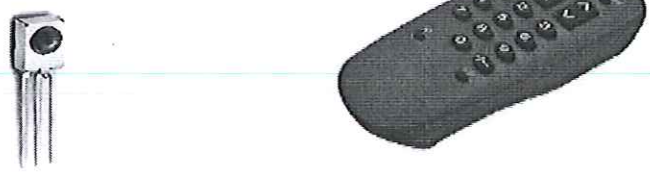
สัญญาณจากรีโมทอินฟราเรด เป็นสัญญาณพัลส์ (Pulse) ต่อเนื่อง (Continuous Stream) ที่มีความถี่ 37.9 KHz ถูกส่งออกมาโดยมอดดูเลทในรูปแสงอินฟราเรดดังนั้นในการจับสัญญาณของรีโมทจะใช้ออสซิลโลสโคปร่วมกับวงจรรับสัญญาณ ในการวัดสัญญาณที่ได้จากการส่งสัญญาณของรีโมทอินฟราเรด ซึ่งรูปวงจรภาครับสัญญาณรีโมทอินฟราเรด จะเป็นดังรูป 3.1



รูปที่ 3.1 วงจรของ IR Remote (อ้างอิงจากภาคผนวก ง)

ในการวัดสัญญาณโดยใช้ ออสซิลโลสโคป นั้นจะวัดที่ขา GND และ ขา OUT ของวงจรดังรูปที่ 3.1 ซึ่งตัวส่งสัญญาณจะเป็นรีโมททั่วไปซึ่งเมื่อกดปุ่มที่รีโมทจะสามารถวัดสัญญาณได้ ในปริญญานิพนธ์นี้ตัวรับสัญญาณอินฟราเรดเราจะใช้ตัวที่มีขายอยู่ในท้องตลาดซึ่งเรียกว่า IRM-8601S ดังแสดงในรูปที่ 3.2 (รายละเอียดคู่มือของ IRM-8601S แสดงอยู่ในภาคผนวก ง)

IRM-8601S



(1) ตัวรับสัญญาณรีโมท IRM – 5601S (2) ตัวส่งสัญญาณรีโมท

รูปที่ 3.2 ตัวรับสัญญาณอินฟราเรดและรีโมทส่งสัญญาณอินฟราเรด

3.2 การจับสัญญาณในไมโครคอนโทรลเลอร์ (TIMER) [6]

รีจิสเตอร์ ฟังก์ชันทั้งหมดที่เกี่ยวข้องกับสัญญาณเวลาและการนับแสดงอยู่ในตารางที่ 3.1 รีจิสเตอร์ TMOD เป็นรีจิสเตอร์ที่ใช้กำหนดแบบวิธีการทำงานของตัวจับเวลา เป็นรีจิสเตอร์ขนาด 8 บิตแบ่งออกเป็น 2 ส่วน ๆ ละ 4 บิต ดังตารางที่ 3.2 โดยแบ่งเป็นส่วนกำหนดแบบวิธีการทำงานของตัวจับเวลา 1 และตัวจับเวลา 0 สำหรับการกำหนดค่าให้รีจิสเตอร์ตัวนี้ กำหนดข้อมูลเป็นไบต์ได้เท่านั้น(ไม่สามารถเข้าถึงข้อมูลในระดับบิตได้) สำหรับผลของการทำงานของตัวจับเวลา/ตัวนับ (จำนวนลูกสัญญาณที่นับได้) สามารถดูจากรีจิสเตอร์ฟังก์ชันเฉพาะ (SFRs) ชื่อ THx และ TLx

ตารางที่ 3.1 รีจิสเตอร์ฟังก์ชันเฉพาะตัวจับเวลา (Timer SFR) [6]

Timer SFR	ประโยชน์	เลขที่อยู่	Bit-addressable
TCON	ควบคุมการทำงาน (Control)	88H	**Yes**
TMOD	กำหนดแบบวิธีการทำงาน (Mode)	89H	No
TL0	รีจิสเตอร์ไบต์ต่ำของตัวจับเวลา 0	8AH	No
TL1	รีจิสเตอร์ไบต์ต่ำของตัวจับเวลา 1	8BH	No
TH0	รีจิสเตอร์ไบต์สูงของตัวจับเวลา 0	8CH	No
TH1	รีจิสเตอร์ไบต์สูงของตัวจับเวลา 1	8DH	No
T2CON*	ควบคุมการทำงานตัวจับเวลา 2	C8H	**Yes**
RCAP2L*	รีจิสเตอร์ไบต์ต่ำ(capture) ของตัวจับเวลา 2	CAH	No
RCAP2H*	รีจิสเตอร์ไบต์สูง (capture) ของตัวจับเวลา 2	CBH	No
TL2*	รีจิสเตอร์ไบตต่ำของตัวจับเวลา 2	CCH	No
TH2*	รีจิสเตอร์ไบต์สูงของตัวจับเวลา 2	CDH	No

ตารางที่ 3.2 บิตต่าง ๆ รีจิสเตอร์ TMOD [6]

บิต	ชื่อ	ตัวจับเวลา	คำอธิบาย
7	GATE	1	เมื่อเซต จะต้องให้ขามีสภาวะเป็นลอจิก 1 ตัวจับเวลา 1 จึงจะทำงานได้ แต่ถ้ารีเซต ตัวจับเวลา 1 จะทำงานได้โดยไม่ต้องรอสภาวะลอจิก 1 ที่ขา
6	C/$\overline{T}$	1	บิตเลือก ตัวจับเวลา/ตัวนับ 1 เลือกตัวนับ 0 เลือกตัวจับเวลา
5	M1	1	บิต 1 เลือกแบบวิธี (Mode bit 1) ของตัวจับเวลา 1 ดูตารางที่ 3.3
4	M0	1	บิต 0 เลือกแบบวิธี (Mode bit 0) ของตัวจับเวลา 1 ดูตารางที่ 3.3
3	GATE	0	ทำงานเช่นเดียวกับบิต 7 แต่ใช้สำหรับตัวจับเวลา 0
2	C/$\overline{T}$	0	บิตเลือก ตัวจับเวลา/ตัวนับ 0
1	M1	0	บิต 1 เลือกแบบวิธี (Mode bit 1) ของตัวจับเวลา 0 ดูตารางที่ 3.3
0	M0	0	บิต 0 เลือกแบบวิธี (Mode bit 0) ของตัวจับเวลา 0 ดูตารางที่ 3.3

ตารางที่ 3.3 แบบวิธีของตัวจับเวลา (Timer Mode) [6]

M1	M0	แบบวิธี	คำอธิบาย
0	0	0	ตัวจับเวลา 13 บิต (13-bit timer mode or 8048 mode)
0	1	1	ตัวจับเวลา 16 บิต (16-bit timer mode)
1	0	2	ตัวจับเวลา 8 บิตใส่ค่าใหม่อัตโนมัติ (8-bit auto-reload mode)
1	1	3	ตัวจับเวลาแบบแบ่งส่วน (Split timer mode) ตัวจับเวลา 0 : TL0 เป็นตัวจับเวลา 8 บิตที่ควบคุมโดยตัวจับเวลา 0 TH0 เป็นตัวจับเวลา 8 บิตที่ควบคุมโดยตัวจับเวลา 1 ตัวจับเวลา 1 : ไม่ทำงาน

3.2.1 แบบวิธีของตัวจับเวลาและตัวบ่งชี้การล้น

ผลของการทำงาน (การนับ) ของตัวจับเวลาจะอยู่ในรีจิสเตอร์ THx และ TLx และผลของการนับล้นจะแสดงที่รีจิสเตอร์ตัวบ่งชี้การล้นของตัวจับเวลา (Timer Overflow Flag) หรือรีจิสเตอร์ TFx โดยที่อักษร x แทนชื่อตัวจับเวลา 0 หรือ 1 ดังรูปที่ 3.3

แบบวิธีของตัวจับเวลาและการเซตค่าตัวจับเวลา สามารถเซตตามคำอธิบายในตารางที่ 3.2 โดยมีแบบวิธีการนับสัญญาณภาย 4 แบบวิธี ดังต่อไปนี้

3.2.1.1. ตัวจับเวลา 13 บิต (13 Bit Timer Mode : Mode 0)

ดังรูปที่ 3.3 (ก) แบบวิธี 0 เป็นการนับ 13 บิต โดยจะใช้รีจิสเตอร์ไบต์บน THx จำนวน 8 บิต (b7-b0) และใช้รีจิสเตอร์ไบต์ล่าง TLx จำนวน 5 บิต (b4-b0) ส่วน 3 บิตบนที่เหลือไม่มีการใช้งาน ทำให้สามารถนับสัญญาณนาฬิกาได้ตั้งแต่ 0000H ถึง 1FFFH โดยจะเกิดการล้นเมื่อผลการนับเปลี่ยนจาก 1FFFH เป็น 2000H ยังผลให้ตัวบ่งชี้การล้น (TFx) เซต

3.2.1.2. ตัวจับเวลา 16 บิต (16 -Bit Timer Mode : Mode 1)

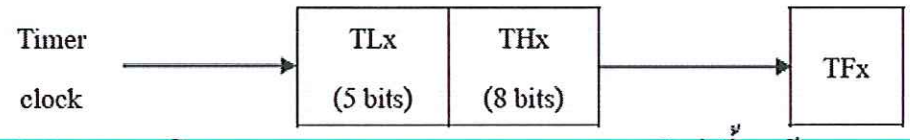
ดังรูปที่ 3.3 (ข) การทำงานแบบวิธีนี้คล้ายกับตัวจับเวลา 13 บิต ต่างกันที่มีขนาด 16 บิต มีรีจิสเตอร์ที่ใช้งานคือ THx ขนาด 8 บิต เป็น ไบต์สูง และ TLx ขนาด 8 บิตเป็นไบต์ต่ำ รวมกันแล้ว เป็นรีจิสเตอร์ขนาด 16 บิต ทำให้สามารถนับสัญญาณนาฬิกาได้ตั้งแต่ 0000H ถึง FFFFH จะเกิดการ ล้นเมื่อผลการนับเปลี่ยนจาก FFFFH เป็น 0000H ยังผลให้ตัวบ่งชี้การล้น (TFx) เซต ตัวจับเวลาจะยังคงนับต่อไป ตัวบ่งชี้การล้น TFx จะอยู่ใน เรจิสเตอร์ TCON ที่สามารถอ่านหรือเขียนได้ด้วยส่วนชุดคำสั่ง (Software)

3.2.1.3. ตัวจับเวลา 8 บิตใส่ค่าใหม่อัตโนมัติ (8-Bit Auto - Reload Mode : Mode 2)

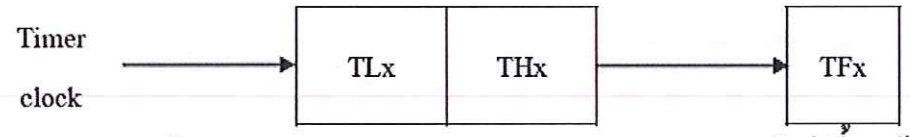
ดังรูปที่ 3.3 (ค) การทำงานของตัวจับเวลานี้จะใช้รีจิสเตอร์ THx เป็นตัวตั้งเวลาขนาด 8 บิตและใช้รีจิสเตอร์ TLx เป็นตัวเก็บผลการนับ เมื่อมีการนับล้นจาก FFH ไปเป็น 00H ยังผลให้ตัวบ่งชี้การล้น (TFx) เซตและยังผลให้ข้อมูลจาก THx ถูกโอนย้าย (Reload) ไปยัง TLx ใหม่ กระบวนการนับนั้นค่าที่เริ่มนับคือค่าใน TLx นับเพิ่มไปจนถึง FFH การนับในแบบวิธีนี้สามารถทำให้ตัวบ่งชี้การ ล้นทำงานด้วยคาบเวลาคงที่ได้ตามต้องการด้วยการกำหนดค่าในรีจิสเตอร์ TMOD และรีจิสเตอร์ THx

3.2.1.4. ตัวจับเวลาแบบแบ่งส่วน (Split time Mode : Mode 3)

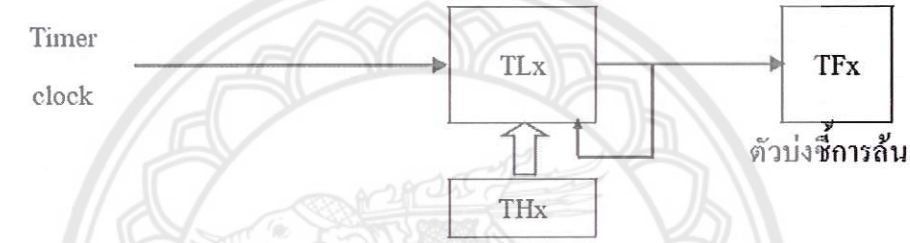
ในแบบวิธี 3 นี้จะทำให้มีตัวนับ/ตัวจับเวลาทั้งหมด 3 ตัว ดังรูปที่ 3.3 (ง) การทำงานในแบบนี้จะแบ่งตัวจับเวลา 0 เป็นตัวจับเวลาขนาด 8 บิต 2 ตัวคือ TL0 และ TH0 โดยใช้ตัวบ่งชี้การล้น TF0 และ TF1 แสดงการล้นของตัวจับเวลาตามลำดับสำหรับตัวจับเวลา 1 ไม่สามารถทำงานในแบบวิธี 3 นี้ แม้จะสามารถใช้นับได้ แต่เมื่อเกิดการล้นจะไม่มีผลกับTF1



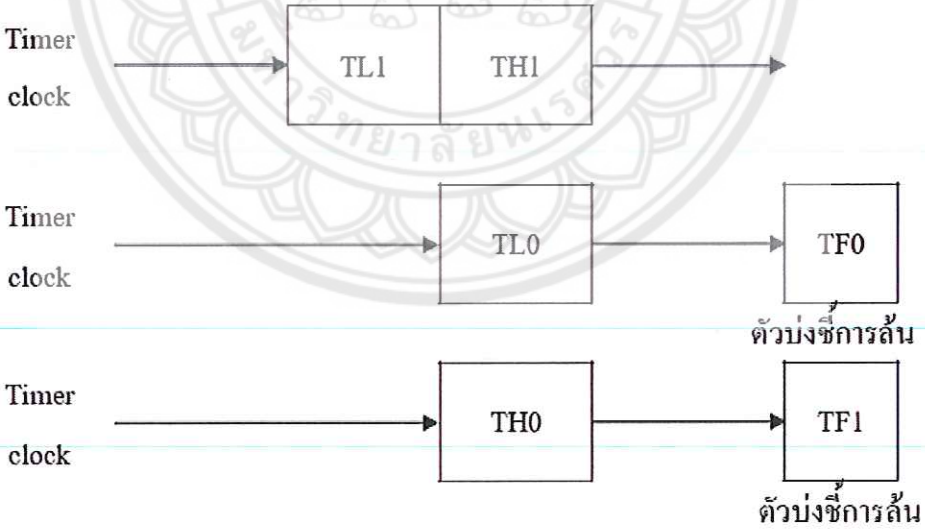
ก. ตัวจับเวลา 13 บิต (Mode 0) ตัวบ่งชี้การล้น (Overflow flag)



ข. ตัวจับเวลา 16 บิต (Mode 1) ตัวบ่งชี้การล้น



ค. ตัวจับเวลา 8 บิตใส่ค่าใหม่อัตโนมัติ (Mode 2)



ง. ตัวจับเวลาแบบแบ่งส่วน (Mode 3)

รูปที่ 3.3 การทำงานของตัวจับเวลา/ตัวนับในแบบวิธีต่าง ๆ [6]

3.2.2 การเลือกใช้งานตัวจับเวลาและตัวนับ

แหล่งจ่ายสัญญาณนาฬิกาสำหรับตัวจับเวลา (Timer Clock) จะมี 2 แหล่ง แหล่งแรกเป็นสัญญาณนาฬิกาภายใน (On-chip oscillator) และแหล่งที่สองเป็นสัญญาณภายนอก (ป้อนเข้าทางขา T0 หรือ T1) ดังรูปที่ 3.4 ในการเลือกการจับเวลาหรือการนับสัญญาณนั้นทำได้โดยการเลือกบิตตัวนับ/ตัวจับเวลา (C/T) ในรีจิสเตอร์ TMOD

3.2.3 ตัวจับเวลา (Timer)

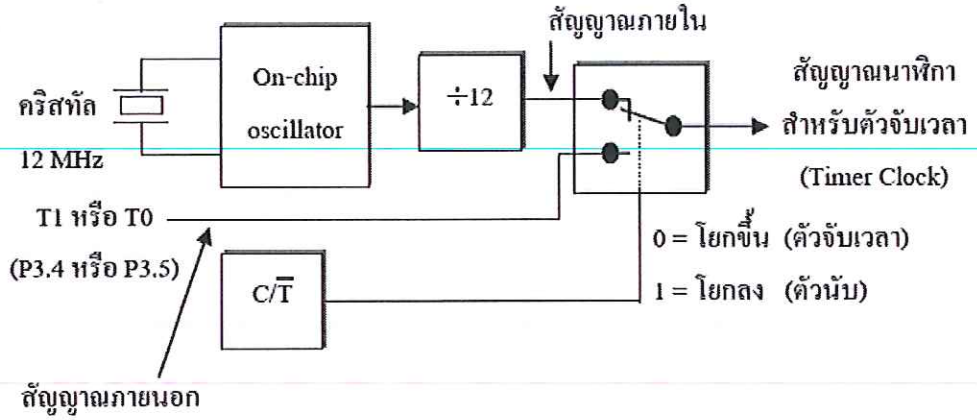
การนับสัญญาณนาฬิกาภายใน (Interval Timing) เมื่อบิต C/T = 0 (ในรีจิสเตอร์ TMOD) ตัวจับเวลาจะทำงานโดยการรับสัญญาณนาฬิกาภายในชิพ สัญญาณนี้ถูกสร้างเป็นสัญญาณนาฬิกาจากวงจรออสซิลเลเตอร์ภายใน (On-chip oscillator) และถูกหารด้วย 12 เพื่อสร้างสัญญาณนาฬิกาสำหรับตัวจับเวลา (Timer Clock) ถ้าใช้คริสตอล 12 MHz จะได้สัญญาณที่มีความถี่ 1 MHz (การนับสัญญาณนาฬิกาหนึ่งลูกเท่ากับ 1 ไมโครวินาที) สัญญาณนี้จะถูกส่งไปยังวงจรจับเวลาที่สามารถกำหนดแบบวิธี (Mode) ต่าง ๆ ดังรูปที่ 3.4

3.2.4 ตัวนับ (Counter)

การนับสัญญาณนาฬิกาภายนอก (Event Counting) เมื่อบิต C/T = 1 (ในรีจิสเตอร์ TMOD) ตัวจับเวลาจะทำงานโดยรับสัญญาณจากแหล่งภายนอก ที่ต้องป้อนเข้ามาทางขา T0 และ T1 (ขา P3.4 และ P3.5 ตามลำดับ) ดังรูปที่ 3.4

รีจิสเตอร์ THx/TLx ของตัวนับจะเพิ่มค่า เมื่อสัญญาณที่ป้อนเข้ามีการเปลี่ยนจากสภาวะจากลอจิก 1 เป็น 0 (ขอบขาลงของสัญญาณ) โดยสัญญาณภายนอกที่ป้อนเข้ามานี้จะถูกสุ่ม (Sampling) ระหว่างสเต็ปที่ 5 พัลส์ลูกที่ 2 (S5P2) ของทุก ๆ รอบของเครื่อง (Machine Cycle)

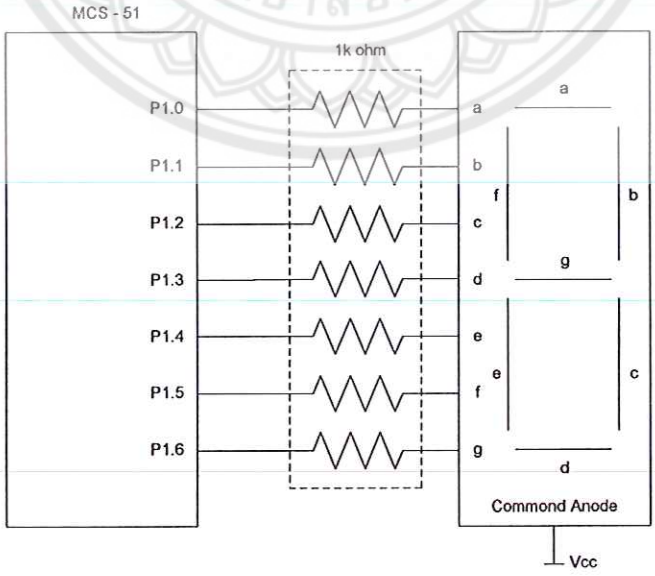
เมื่อต่อXTAL 12 MHz เข้าไปในรูปที่ 3.4 เวลาที่ใช้ตั้งแต่รับสัญญาณนาฬิกาเข้าที่ขา T0/T1 จนถึงการนับเพิ่มค่าในรีจิสเตอร์ THx/TLx จะใช้เวลา 2 รอบของเครื่องหรือ 2 ไมโครวินาที ดังนั้นไมโครคอนโทรลเลอร์ 8051 ตัวนี้จะมีข้อจำกัดในการรับสัญญาณจากภายนอกได้ความถี่สูงสุด 500 kHz (1/(2x10-6 s)) เท่านั้น ถ้าความถี่สูงกว่านี้ ตัวนับภายในไมโครคอนโทรลเลอร์จะไม่สามารถทำงานได้



รูปที่ 3.4 แหล่งจ่ายสัญญาณนาฬิกากับการเลือกตัวจับเวลา/ตัวนับ

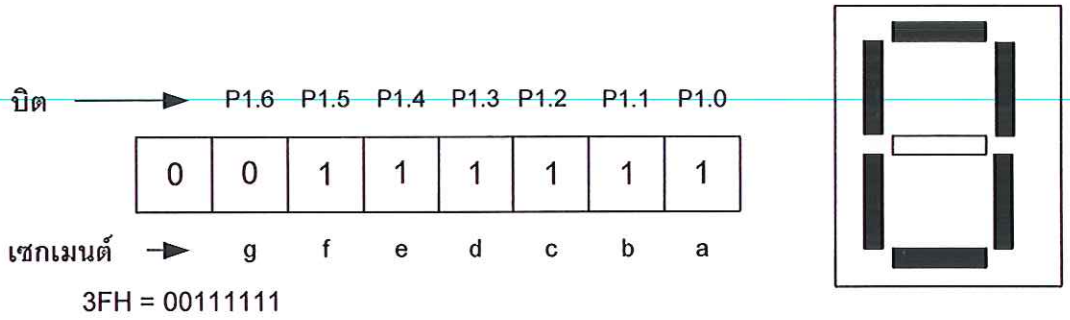
3.3 การต่อหลอด LED 7 Segment [4]

ในปริญญานิพนธ์นี้จะมีตัววงจรอยู่ส่วนหนึ่งคือบอร์ดแสดงผล 7-Segment เพื่อแสดงค่าการทำงานของโปรแกรมคือเมื่อกดรีโมทหมายเลข 1 จะให้บอร์ด7-Segment นี้แสดงค่าออกมา เป็นเลข 1 ด้วยซึ่งการทำงานนั้นสามารถวงจร 7-Segment นำมาต่อกับพอร์ตของไมโครคอนโทรลเลอร์ได้ ซึ่งการต่อบอร์ด7-Segment ถ้าหากต่อกับพอร์ต P1 จะต้องกับบัฟเฟอร์ช่วยขับกระแสเนื่องจาก LED แต่ละหลอดจะต้องการกระแสประมาณ 15 มิลลิแอมป์และถ้าให้ LED ทุกหลอดสว่างพร้อมกันหมดจะต้องใช้กระแสค่อนข้างมาก ซึ่งตัว MCS-51 เองจะไม่สามารถขับกระแสได้ แต่ถ้าต่อกับพอร์ต 8255 ก็สามารถต่อได้ทันที ถ้าหากต่อ LED 7 เซกเมนต์แบบแอโนดร่วมกับพอร์ต P1 สามารถทำได้ดังรูปที่ 3.5



รูปที่3.5 การต่อ LED 7 เซกเมนต์แบบแอโนดร่วมกับ P1

จากรูปที่ 3.5 ถ้าหากส่งข้อมูลออกทางพอร์ต P1 เป็นค่า 3FH จะทำให้สว่างเป็นเลขศูนย์ดังรูปที่ 3.6



รูปที่3.6 ค่าแต่ละบิตของพอร์ต P1 และการแสดงผลแบบแคโทด

ถ้าหากนำ LED 7 เซกเมนต์ไปต่อกับพอร์ตโดยให้บิตต่ำสุดต่อกับเซกเมนต์ a บิตต่อมาต่อกับเซกเมนต์ b เรียงไปเรื่อยๆ จนถึงเซกเมนต์ g ถ้าหากต้องการแสดงผลเป็นเลขต่างๆจะต้องส่งผลข้อมูลออกไปดังตารางที่ 3.4

ตารางที่ 3.4 รูปแบบการแสดงผลและค่าของพอร์ตแบบแอโนด

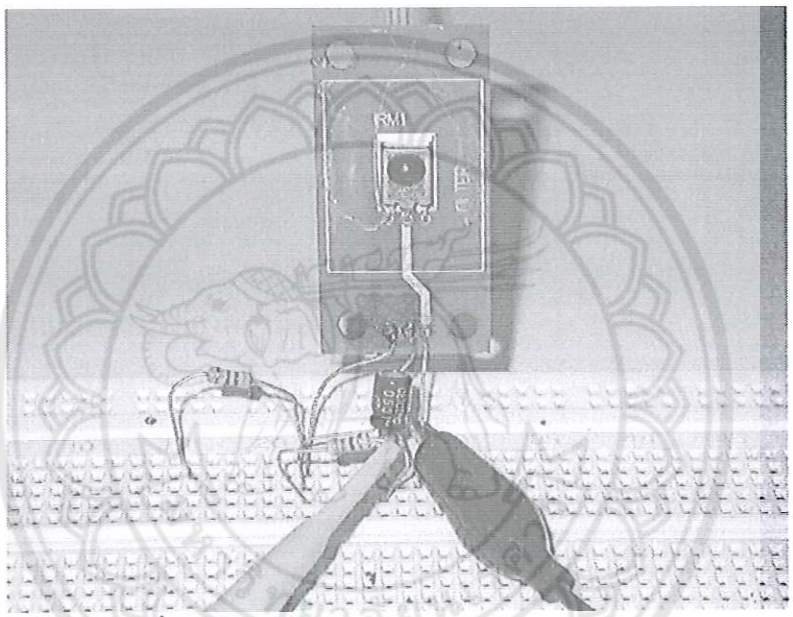
ตัวเลข	g	f	e	d	c	b	a	ค่าพอร์ต
0	1	0	0	0	0	0	0	C0H
1	1	1	1	1	0	0	1	F9H
2	0	1	1	0	1	0	0	A4H
3	0	1	1	0	0	0	0	B0H
4	0	0	1	1	0	0	1	99H
5	0	0	1	0	0	1	0	92H
6	0	0	0	0	0	1	0	82H
7	1	1	1	1	0	0	0	F8H
8	0	0	0	0	0	0	0	80H
9	0	0	1	0	0	0	0	90H

บทที่4

การทดลองและผลการทดลอง

4.1 การวัดสัญญาณของรีโมทอินฟราเรด

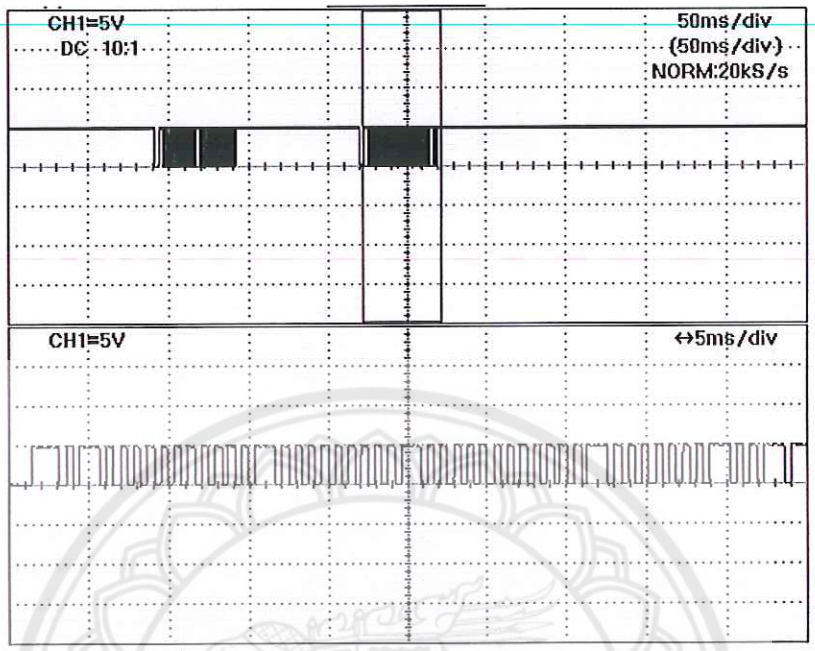
จากการต่อวงจรในรูปที่ 3.1 นั้นสามารถวัดสัญญาณรีโมทซึ่งจะใช้ ออสซิลโลสโคปรุ่น DL1520 ยี่ห้อ YOKOGAWA [7] วัดสัญญาณโดยจับที่ขา OUT และขา GND ตามวงจรที่ได้ต่อไว้ตามรูปวงจรที่ 4.1



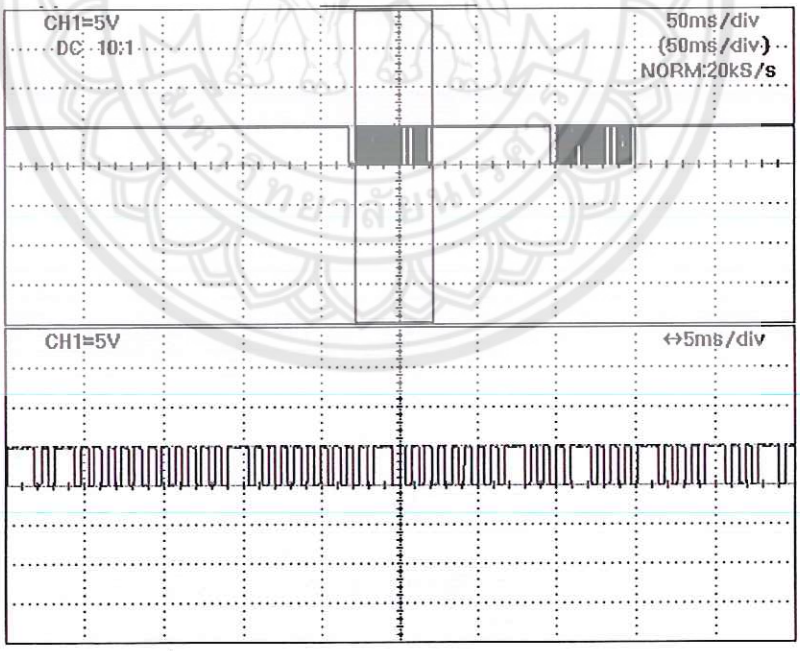
รูปที่ 4.1 การวัดสัญญาณจากวงจรของ IR Remote

การวัดสัญญาณจากการใช้ออสซิลโลสโคปรุ่น DL1520 ยี่ห้อ YOKOGAWA [7] นั้นควรตั้งค่า time / div ไว้ที่ 50 – 200 ms. เพราะเนื่องจากสัญญาณที่เข้ามามีค่าความเร็วที่สูงทำให้จับสัญญาณได้ยาก และตั้งค่า Vdc. / div อยู่ที่ +5 Vdc. / div เพื่อให้ขนาดของสัญญาณมีขนาดที่พอเหมาะซึ่งตอนแรกก่อนที่มีสัญญาณเข้ามานั้นสัญญาณจะคงที่อยู่ที่ +5 Vdc เมื่อกดปุ่มรีโมทก็จะมีสัญญาณวิ่งเข้ามาสัญญาณก็จะดรอปลงสู่ 0 +Vdc.สลับกับขึ้น +5 Vdc เป็นลูกคลื่นจำนวน 49 ลูกคลื่นและก็จะคงที่เท่ากับ +5 Vdc ดังนั้นเมื่อมีกดปุ่มรีโมทก็ให้ กดปุ่ม stop ที่เครื่อง ออสซิลโลสโคป แล้วทำการขยายภาพสัญญาณภาพให้มีขนาดใหญ่พอเหมาะ

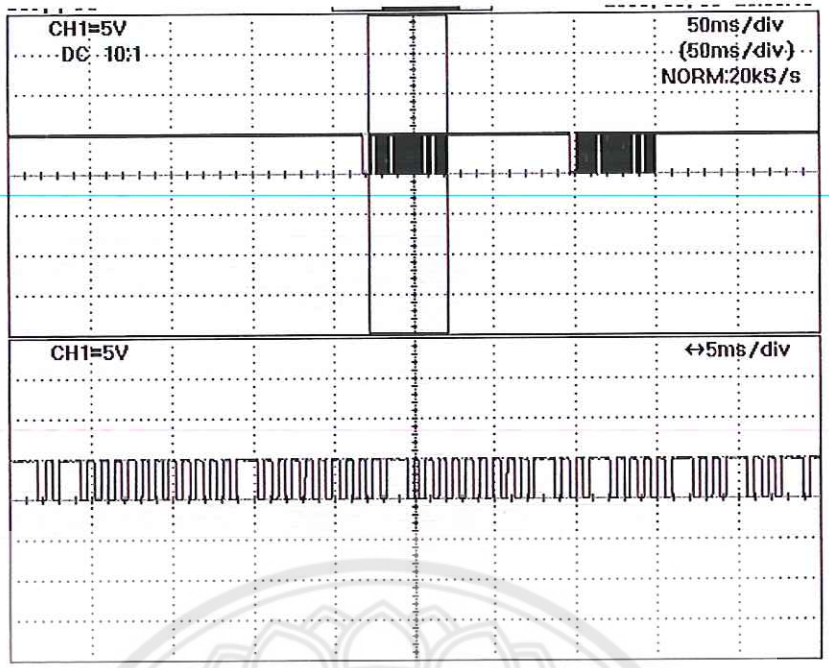
ผลการทดลองจะได้รูปกราฟสัญญาณของรีโมทออกมาเหมือนกัน โดยการกดหมายเลข 0 – 9 ซึ่งจะได้รูปดังนี้



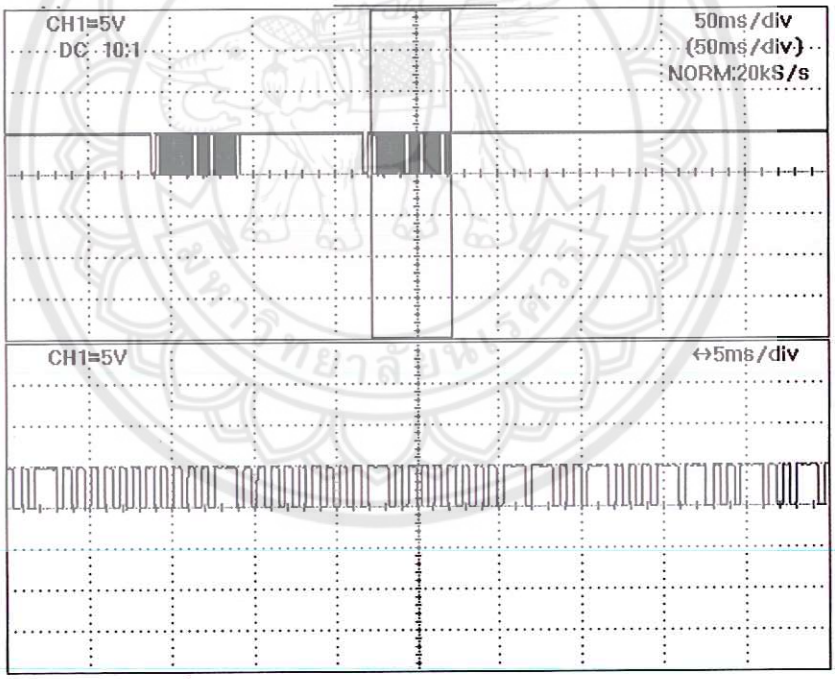
รูปที่ 4.2 สัญญาณรีโมทจากการกดปุ่มหมายเลข 1



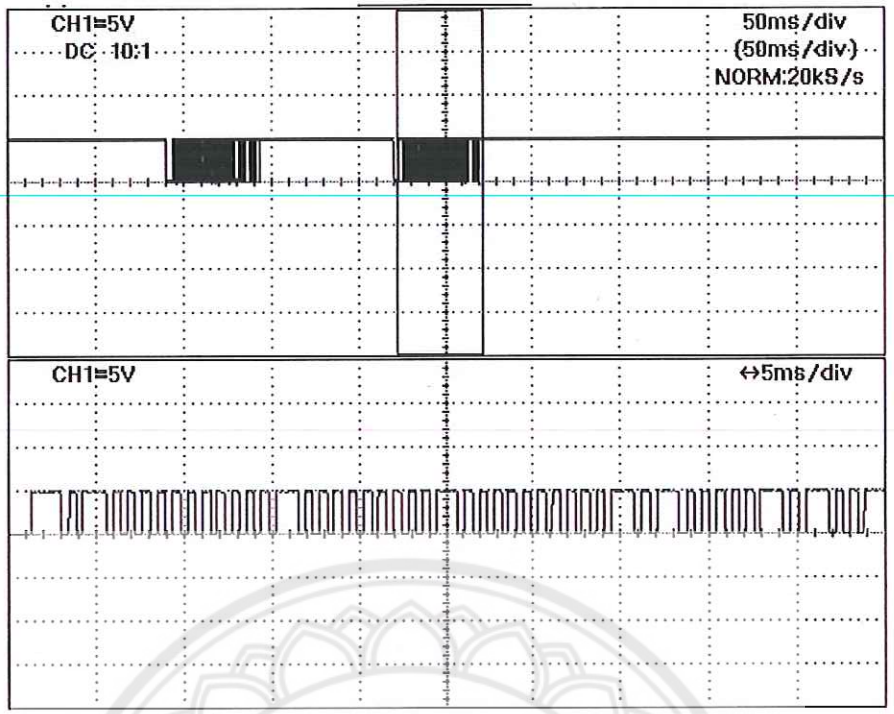
รูปที่ 4.3 สัญญาณรีโมทจากการกดปุ่มหมายเลข 2



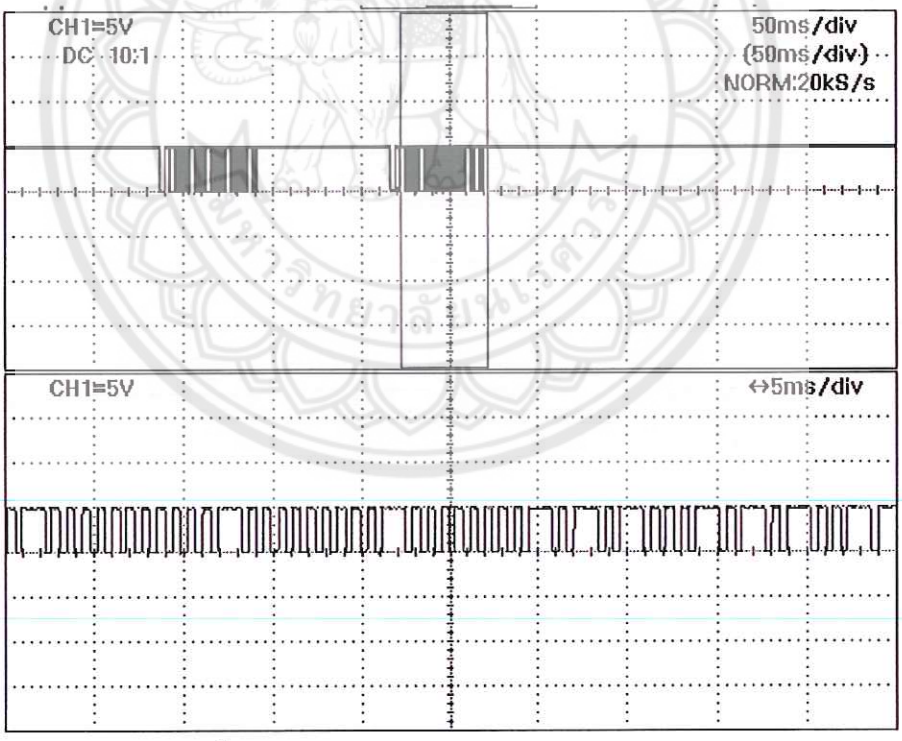
รูปที่ 4.4 สัญญาณรีโมทจากการกดปุ่มหมายเลข 3



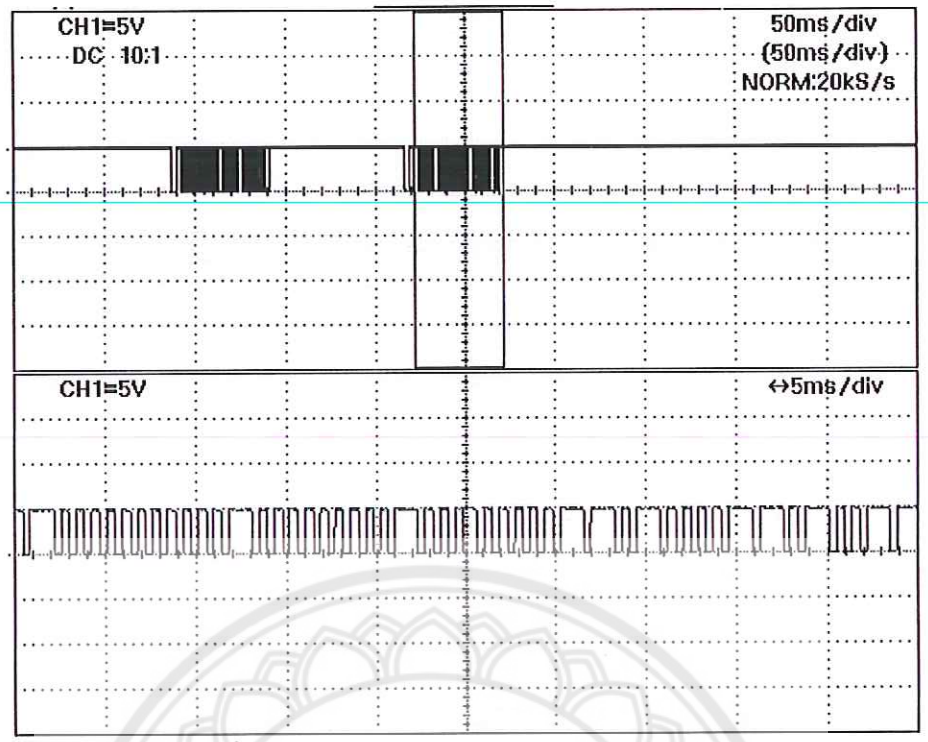
รูปที่ 4.5 สัญญาณรีโมทจากการกดปุ่มหมายเลข 4



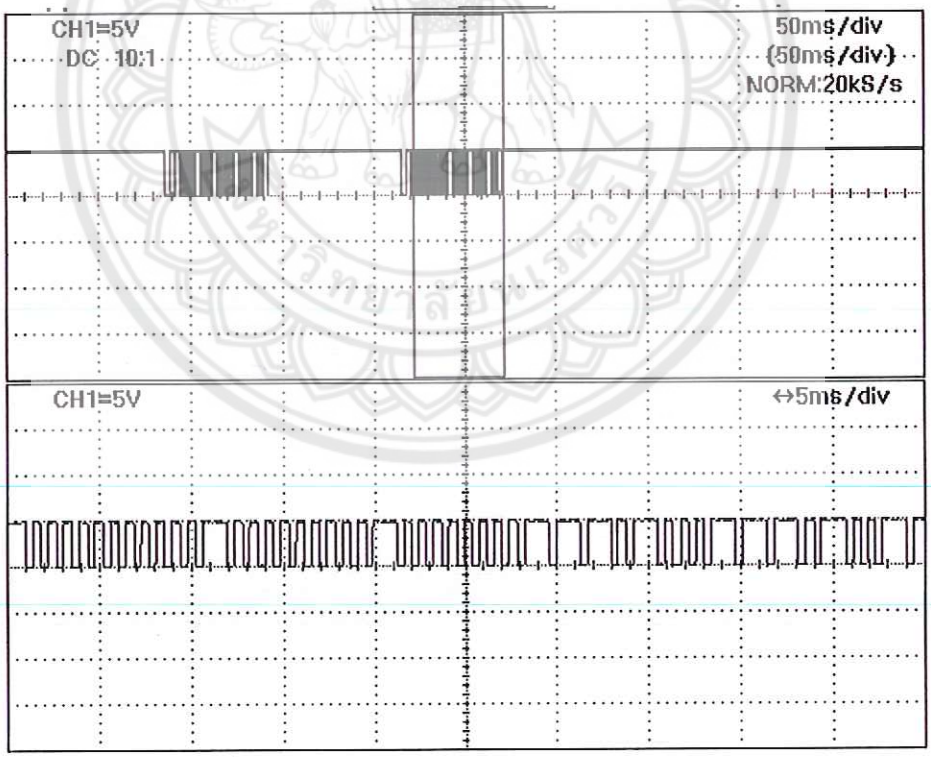
รูปที่ 4.6 สัญญาณรีโมทจากการกดปุ่มหมายเลข 5



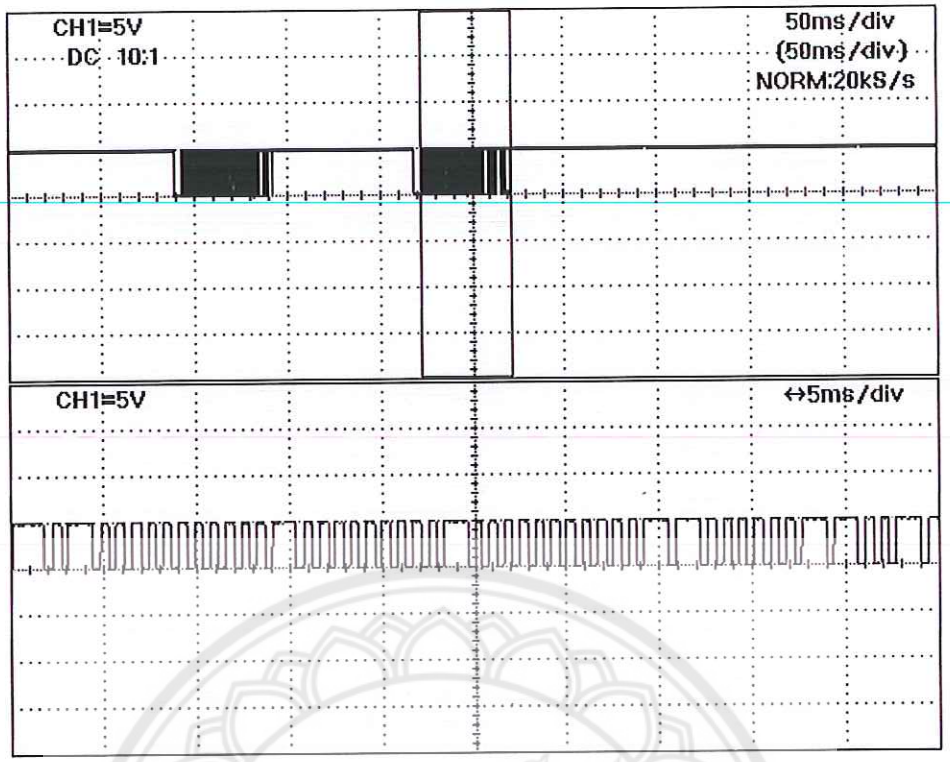
รูปที่ 4.7 สัญญาณรีโมทจากการกดปุ่มหมายเลข 6



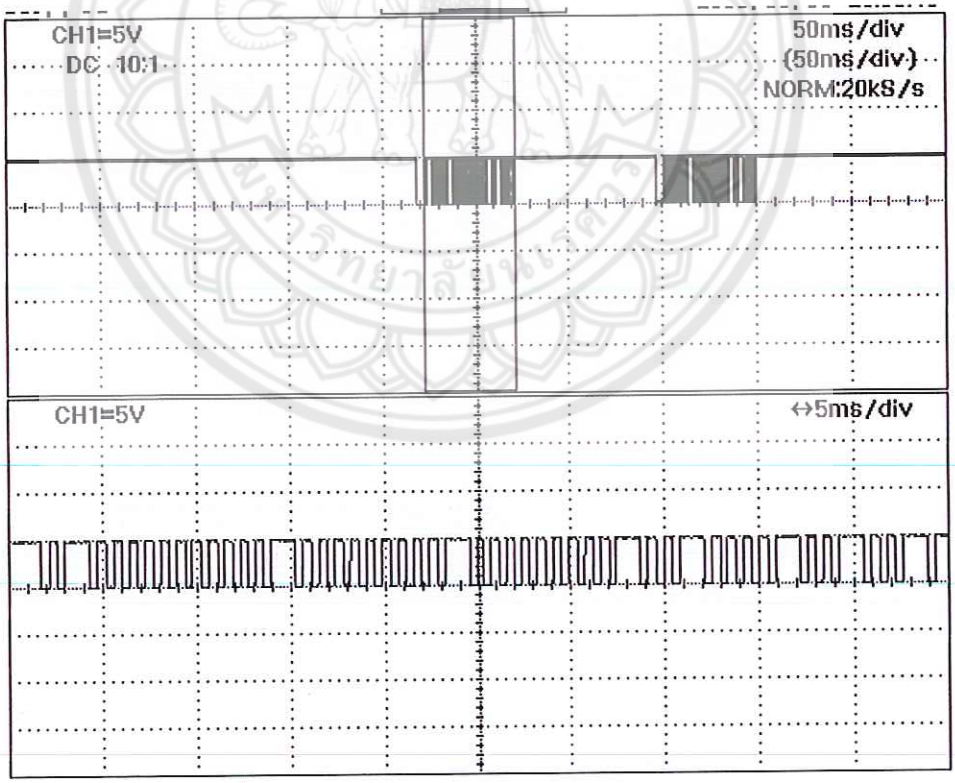
รูปที่ 4.8 สัญญาณรีโมทจากการกดปุ่มหมายเลข 7



รูปที่ 4.9 สัญญาณรีโมทจากการกดปุ่มหมายเลข 8

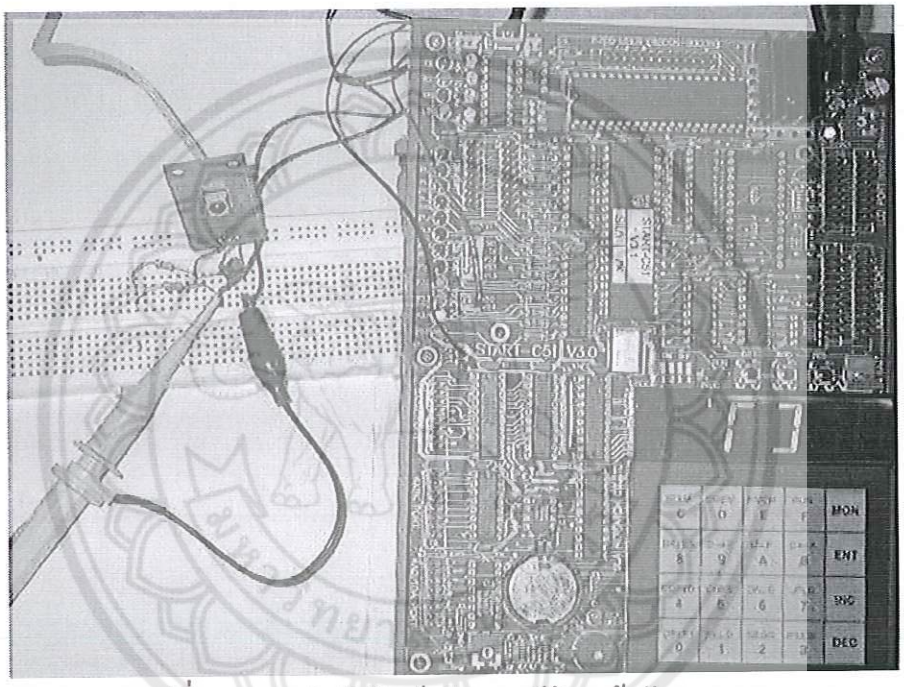


รูปที่ 4.10 สัญญาณรีโมทจากการกดปุ่มหมายเลข 9



รูปที่ 4.11 สัญญาณรีโมทจากการกดปุ่มหมายเลข 0

จากการวัดสัญญาณของการกดรีโมทตั้งแต่หมายเลข 0 ถึง 9 สามารถวิเคราะห์กราฟได้ว่ากราฟที่ได้จะมีทั้งหมด 49 ลูกคลื่นจะมี Start bit แรกที่เป็นตัวบอกว่ามีการเข้าสู่สัญญาณของรีโมทซึ่งก่อนกดปุ่มสัญญาณจะเป็น "1" หรือ +5Vdc. ก่อนและเมื่อมีการกดก็จะเริ่มเข้าสู่สัญญาณของรีโมทซึ่งจะมีการดรอปมาเป็น "0" สลับเป็นลูกคลื่นไปซึ่งจากการวิเคราะห์สัญญาณจะพบว่าสัญญาณรีโมทจะมีลูกคลื่นสัญญาณที่เหมือนกันอยู่ 33 ลูกคลื่นสัญญาณ ที่เหมือนกันซึ่งเรียกว่า Device Code และจะมีสัญญาณอีก 16 ลูกคลื่นสัญญาณที่แตกต่างกันซึ่งเรียกว่า Command ID จากการวิเคราะห์นี้เป็นผลจากการนำขาสัญญาณมาต่อเข้ากับไมโครคอนโทรลเลอร์ START – C51 ที่ P 3.4 และต่อ GND เข้าร่วมกับ GND บอร์ดดังรูปที่ 4.12

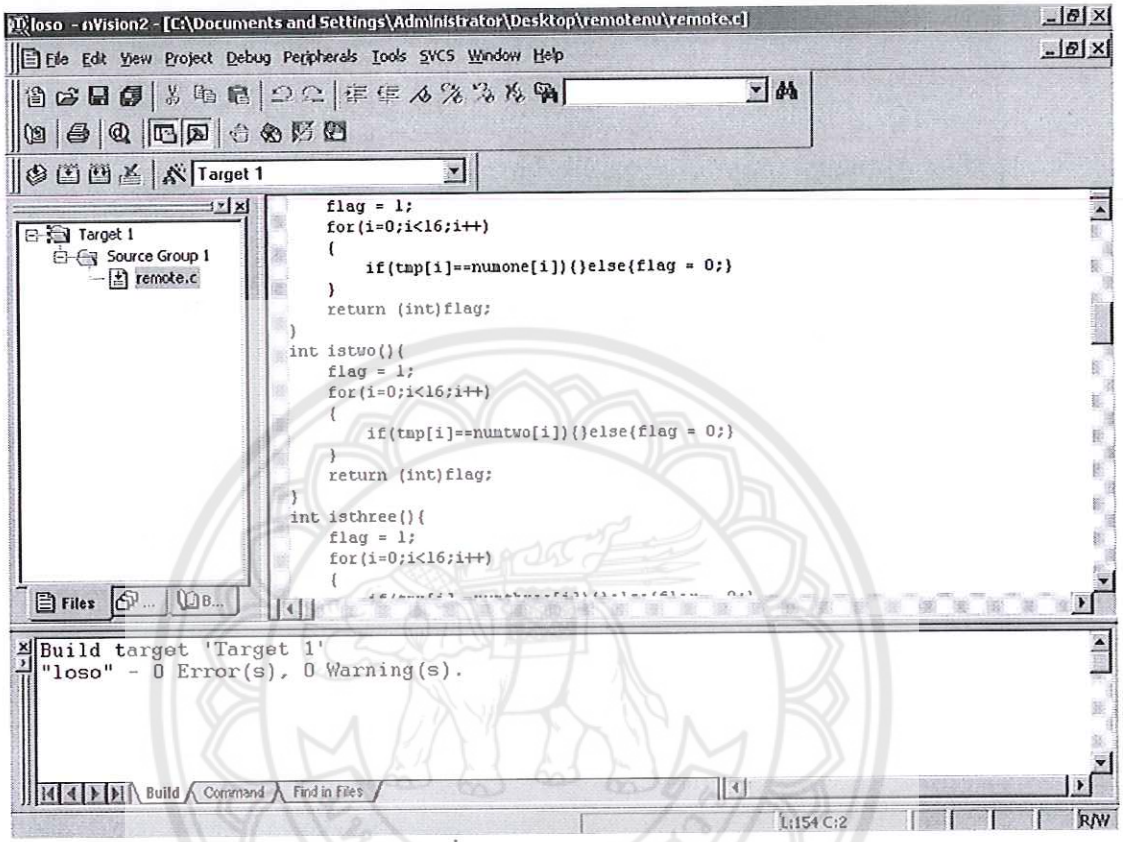


รูปที่ 4.12 การต่อตัวรับเข้าสัญญาณรีโมทเข้ากับขา P 3.4

และทำการเขียนโปรแกรมเพื่อวิเคราะห์สัญญาณที่รับมาได้ซึ่งโปรแกรมมีอัลกอลิทึม และส่วนของรายละเอียดแสดงอยู่ในบทย่อยถัดไป

4.2 โปรแกรมการตรวจจับสัญญาณรีโมท

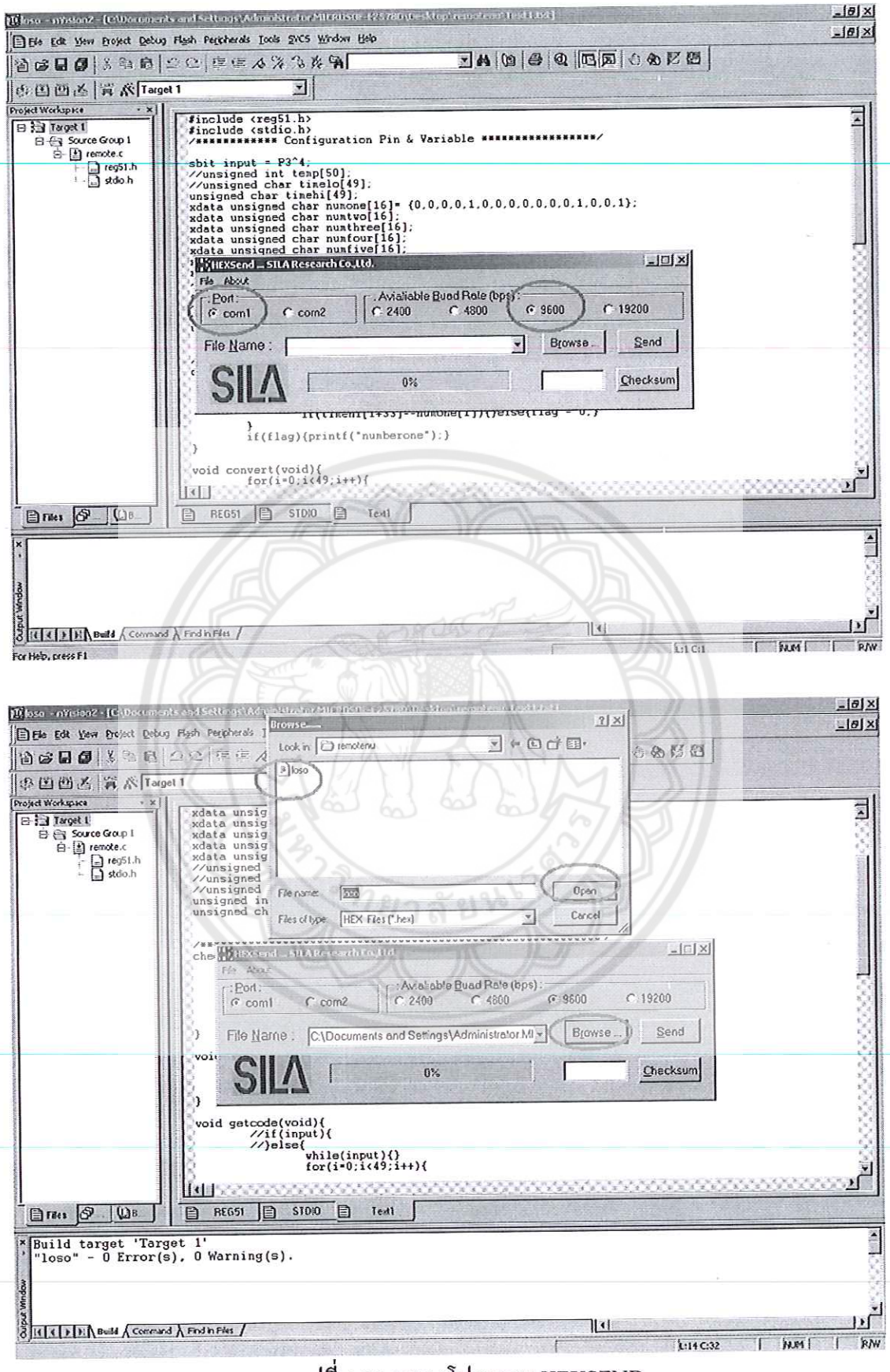
โปรแกรมที่ใช้ในการเขียนโปรแกรมการตรวจจับสัญญาณรีโมทคือ โปรแกรม KEIL ซึ่งใช้ภาษาซี ในการเขียนโปรแกรม



รูปที่ 4.13 โปรแกรม KEIL

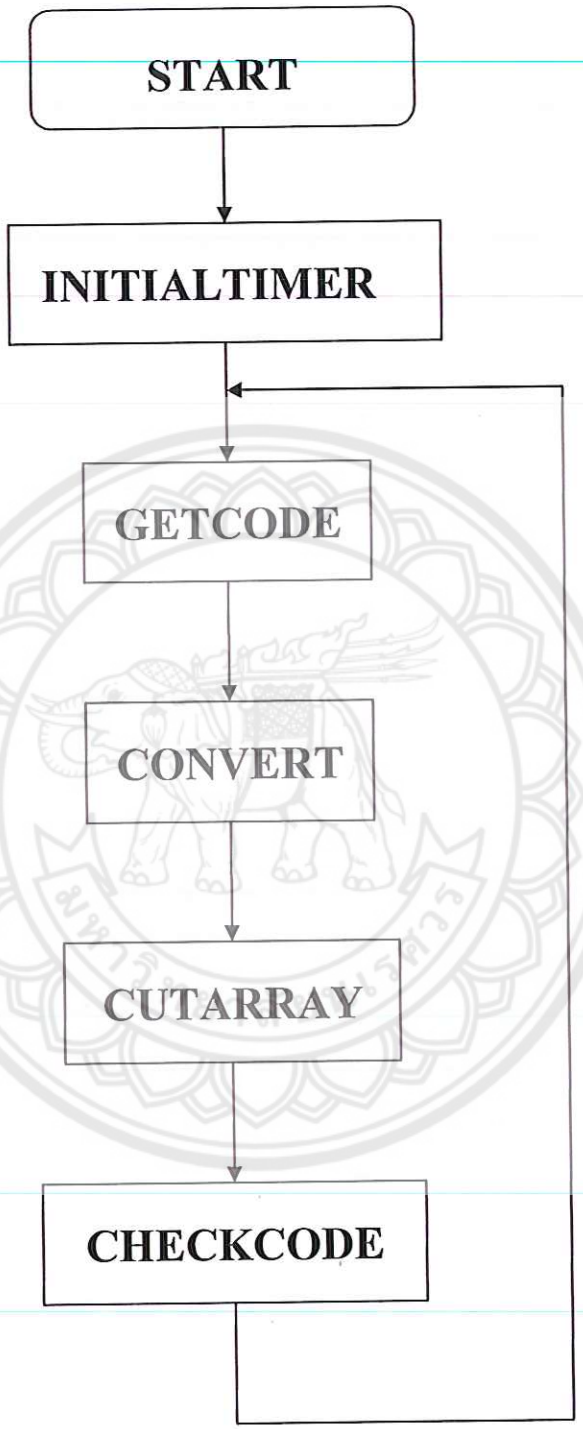
หลังจากที่ทำการเขียนโปรแกรมซึ่งอยู่ในภาคผนวก ข ในการตรวจจับสัญญาณเสร็จก็ทำการลงโปรแกรมไปที่ บอร์ด ไมโครคอนโทรลเลอร์ START – C51 โดยมีวิธีการลงโปรแกรมคือต่อสาย RS 232 เข้ากับคอมพิวเตอร์ทางพอร์ต Serialโปรแกรมที่ใช้ในการลงโปรแกรมของบอร์ดไมโครคอนโทรลเลอร์ START – C51 คือ โปรแกรม HEXSEND ดังรูป 4.14

ซึ่งจะการเขียนโปรแกรมเสร็จจะได้ขนาดโปรแกรมที่ได้มีการคอมไพล์โปรแกรมแล้วเท่ากับ 1,980 bytes และหน่วยความจำภายในที่ใช้เท่ากับ 61 bytes หน่วยความจำภายนอกจะมีค่าเท่ากับ 176 bytes เนื่องจากจะใช้หน่วยความจำภายในของไอซีทั้งหมดเพราะตัวไอซีจะใช้ CPU ของ PHILIPS เบอร์ P89C51RD2/V51RD2 ซึ่งจะมีค่า EEPROM เท่ากับ 2 Kbytes ,และตัวโปรแกรมจะมีการรันโปรแกรมแบบ 6 CLOCK 1 คำสั่ง และบอร์ดจะมีค่าความถี่เท่ากับ 18.432 MHz



รูปที่ 4.14 การลงโปรแกรม HEXSEND

4.2.1 แผนภาพการเขียนโปรแกรมหลัก

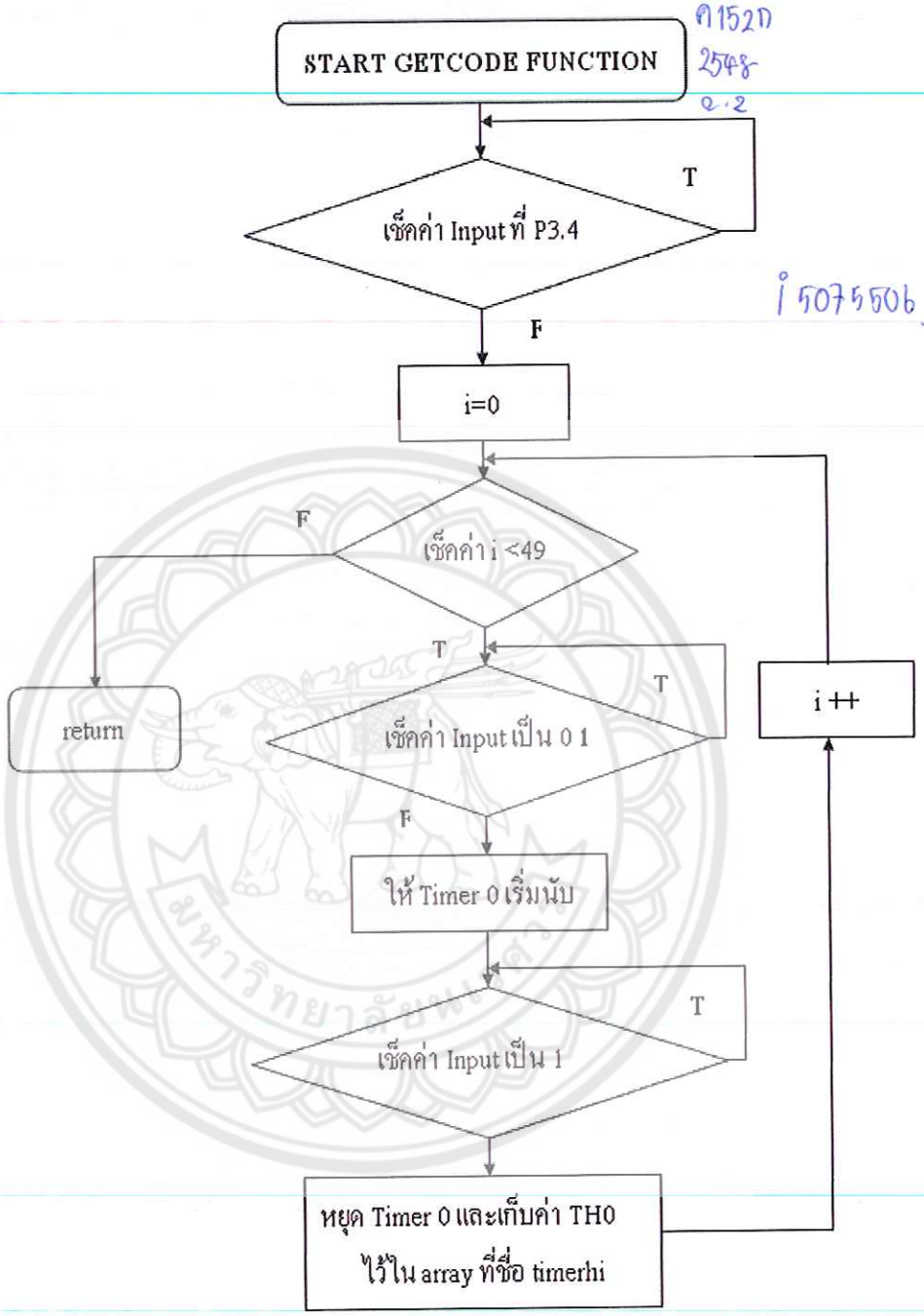


รูปที่ 4.15 แผนภาพการเขียนโปรแกรมหลัก

ห้องสมุดคณะวิศวกรรมศาสตร์

4.2.2 แผนภาพโปรแกรมย่อยของ GETCODE ฟังก์ชัน

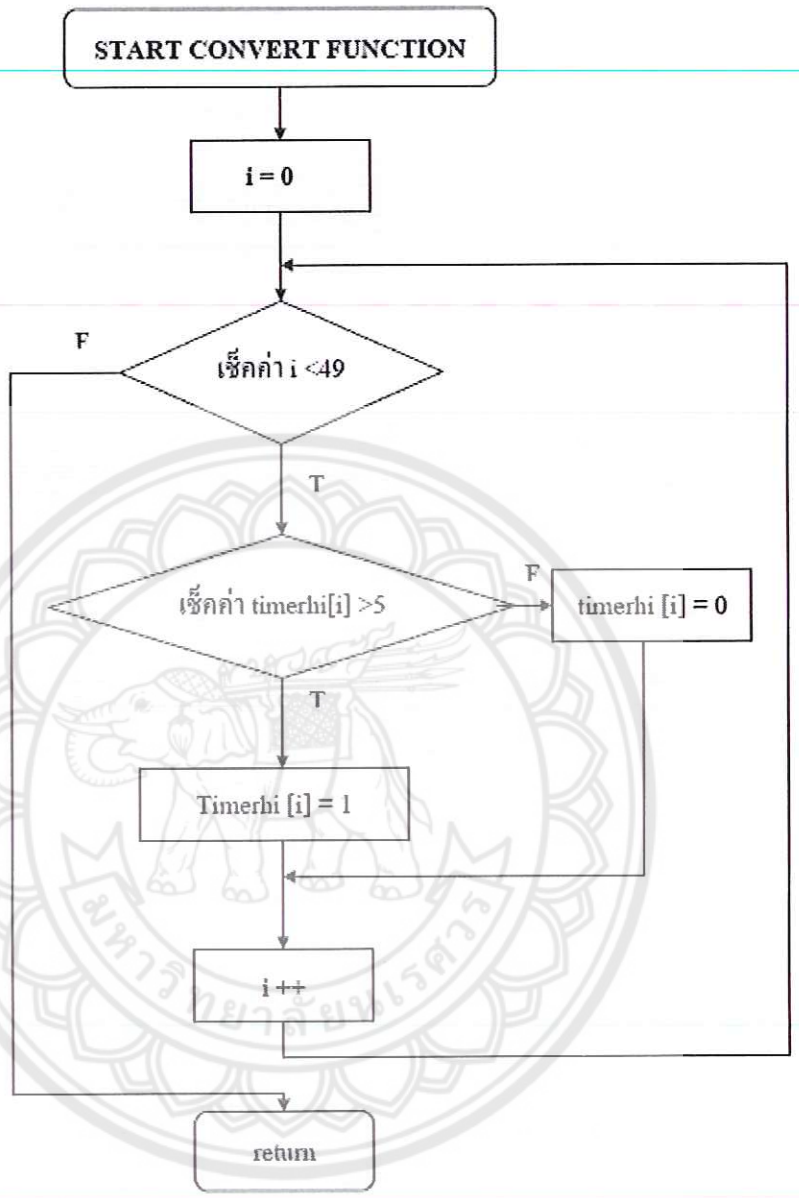
วร. 4900158
ค152ก
2548
c.2
I 5075506



รูปที่ 4.16 แผนภาพการเขียนโปรแกรมฟังก์ชัน GETCODE

ฟังก์ชัน GETCODE นี้จะเป็นส่วนของการเริ่มนับสัญญาณที่จะเข้าทาง P3.4 ของไมโครคอนโทรลเลอร์โดยจะมีการเช็คว่าถ้าเป็นกราฟสัญญาณที่เป็น input ก็จะให้วนลูปไปเรื่อยๆ จนกว่าจะมาเข้าลูปของ i ตั้งแต่ 0 ถึง 48 คือจะนับจำนวนของลูกคลื่นสัญญาณทั้งหมด 49 ลูกคลื่นโดยใช้ TH0 ในจากในการนับสัญญาณวนลูปไปเรื่อยๆ และนำไปเก็บใน array ที่ชื่อ timerhi

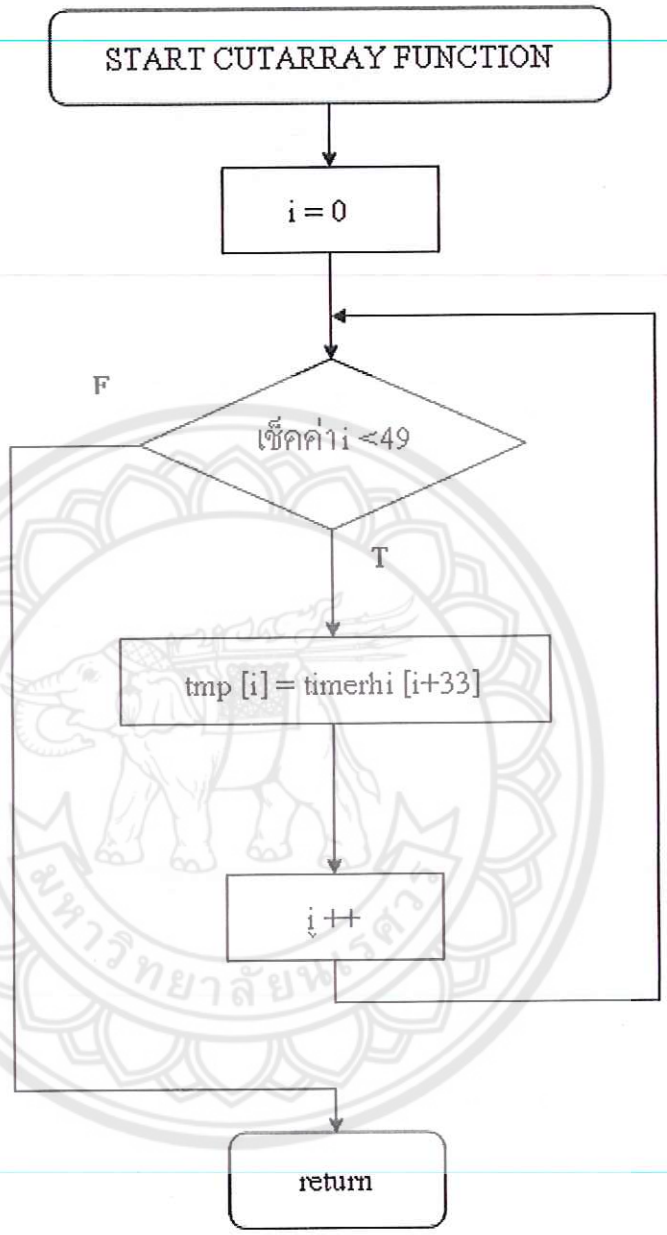
4.2.3 แผนภาพโปรแกรมย่อยของ CONVERT ฟังก์ชัน



รูปที่ 4.17 แผนภาพการเขียนโปรแกรมฟังก์ชัน CONVERT

ฟังก์ชัน CONVERT จะเป็นส่วนของโปรแกรมที่ใช้ในการเช็ค ค่าของ array ที่ชื่อ timerhi ที่เก็บไว้ 49 ช่องซึ่งโปรแกรมในส่วนนี้ก็ตรวจสอบว่า ค่าที่เก็บไว้ใน timerhi โดยจะตรวจสอบว่าค่าที่เก็บไว้ใน ช่องที่ 0 ถึง 49 นั้นค่าที่ได้จะมีอยู่ 4 ค่าคือ 2 , 3 , 9 , 12 ซึ่งฟังก์ชันนี้จะคอยตรวจสอบว่าในแต่ละช่องมีค่ามากกว่าหรือน้อยกว่า 5 ถ้ามากกว่าให้เป็น 1 ถ้าน้อยกว่าให้ มีค่าเป็น 0 วนไปเรื่อยๆ จนครบ 49 ช่องก็ให้ไปที่ return

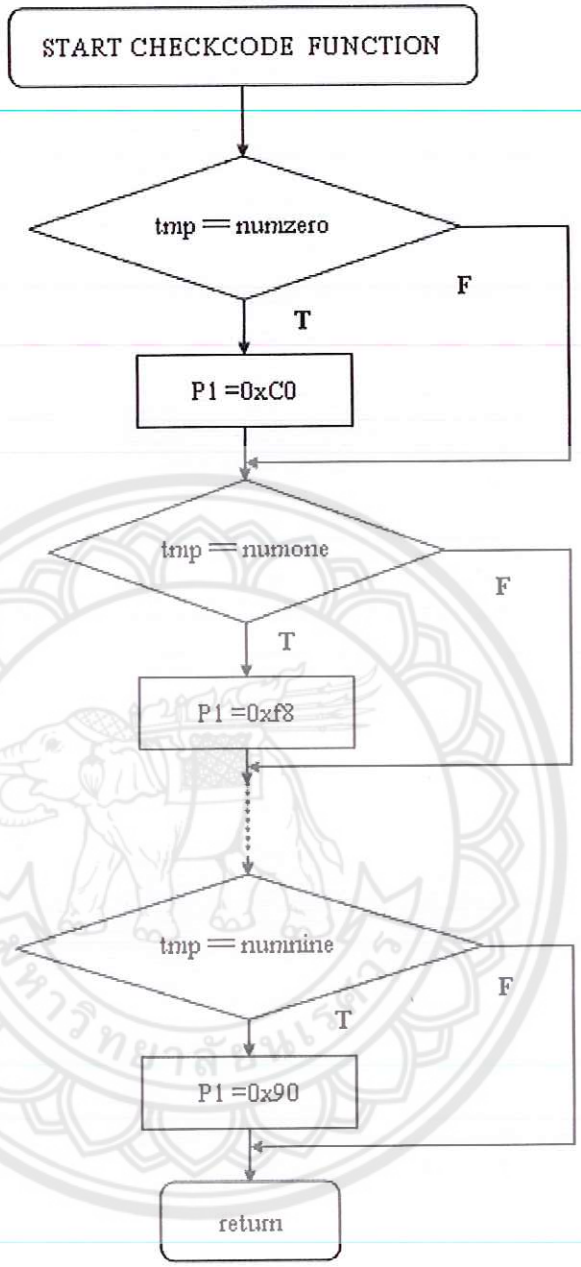
4.2.4 แผนภาพโปรแกรมย่อยของ CUTARRAY ฟังก์ชัน



รูปที่ 4.18 แผนภาพการเขียนโปรแกรมฟังก์ชัน CUTARRAY

ฟังก์ชัน CUTARRAY เป็นส่วนของโปรแกรมที่ใช้ในการ ตัดจำนวนช่องของ array .ให้มีค่าน้อยลงจากการที่สังเกตกราฟสัญญาณรีโมทจะพบว่ารีโมทจะมีลูกคลื่นสัญญาณที่มีความเหมือนกันของแต่ละปุ่มกดจำนวน 33 ลูกคลื่นที่เรียกว่า Device Code และจำนวนอีก 16 ลูกคลื่นที่มีความแตกต่างกัน ดังนั้นโปรแกรมตัวนี้จะเป็นตัวที่ใช้ตัดจำนวนของ array ที่ชื่อ timerhi จาก 49 ช่องให้เหลือ 16 ช่องโดยจะเปลี่ยนชื่อของ array ใหม่ ที่ชื่อ tmp

4.2.5 แผนภาพโปรแกรมย่อยของ CHECKCODE ฟังก์ชัน

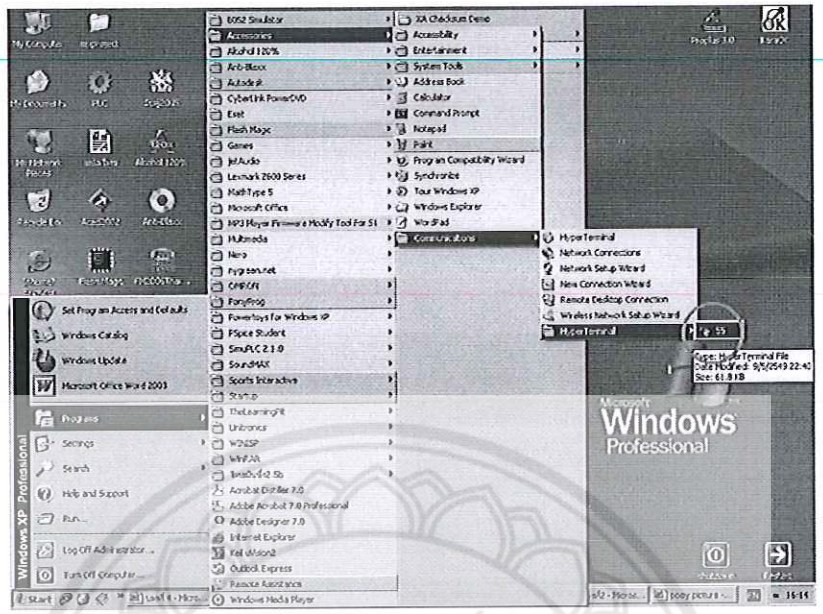


รูปที่ 4.19 แผนภาพการเขียนโปรแกรมฟังก์ชัน CHECKCODE

ฟังก์ชัน CHECKCODE เป็นส่วนของโปรแกรมที่ใช้ในการ เช็คค่าโปรแกรมคือ ในตัวโปรแกรมเราจะมีตัวแปรที่เก็บไว้ อยู่ 10 ตัวที่ส่วนของ Xdata ซึ่งแต่ละตัวแปรก็จะเก็บค่าไว้อยู่ตัวแปรละ 16 ค่า เมื่อเราก็สัญญาณรีโมทเข้าไปสัญญาณก็จะไปผ่านฟังก์ชันต่างๆมาก็จะเหลือ array ที่ชื่อ tmp ซึ่งจะมีตัวแปรอยู่ 16 ค่า ก็จะนำมาเทียบกับตัวแปรในส่วนของ Xdata ที่ละตัวถ้าตรงกับตัวแปรไหนก็จะให้มันคืนค่า flag ตัวนั้นออกมาทาง พอร์ตตามที่ได้กำหนดค่าไว้ก็จะไปแสดงที่ 7 –segment แต่ถ้าไม่ตรงก็ให้ว่างไว้ไม่ให้ทำอะไร

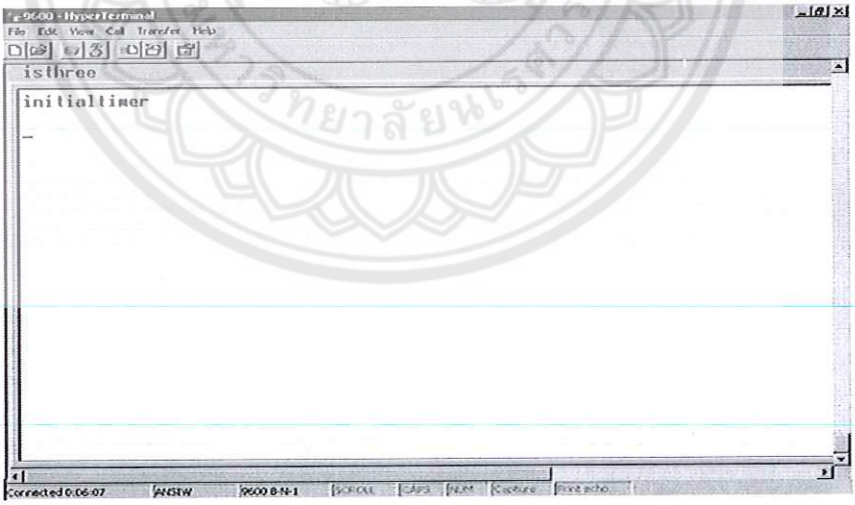
4.2.6 โปรแกรมที่ใช้ในการแสดงผลการตรวจจับสัญญาณรีโมท

โปรแกรมที่ใช้ในการแสดงผลการตรวจจับสัญญาณรีโมท



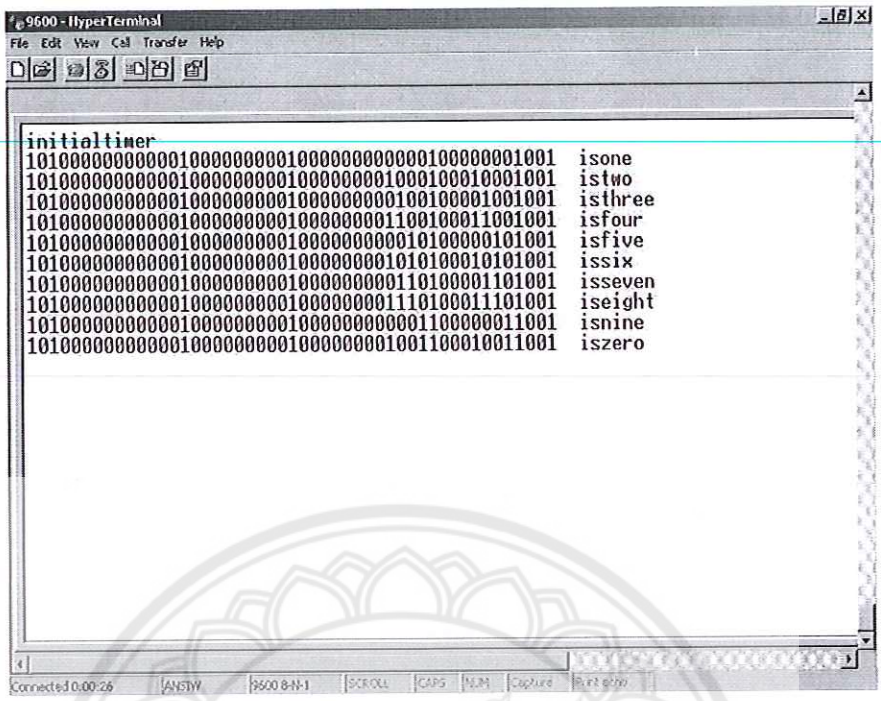
รูปที่ 4.20 โปรแกรมที่ใช้ในการแสดงผลการตรวจจับสัญญาณรีโมท

เมื่อเปิดโปรแกรมและได้ RUN โปรแกรมที่ได้ลงไว้ในบอร์ดไมโครคอนโทรลเลอร์ START – C51 และทำการเปิดโปรแกรม HYPER TERMINAL จะได้หน้าจอออกมาดังนี้



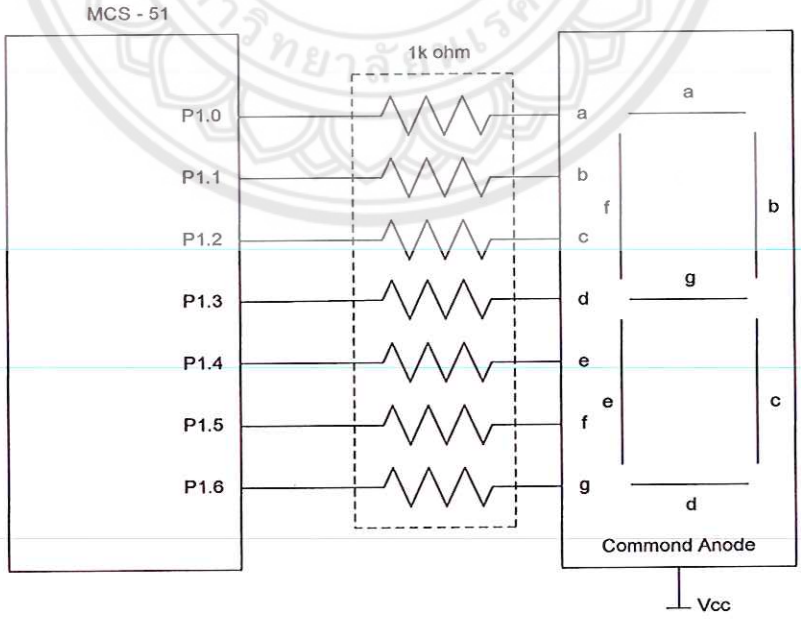
รูปที่ 4.21 โปรแกรม HYPER TERMINAL

เมื่อเราทำการกดหมายเลข 0 – 9 ที่ ตัวส่งรีโมทโปรแกรม HYPER TERMINAL ก็จะแสดงค่าสัญญาณที่ส่งเข้าไปในบอร์ดและรับค่าก็จะประมวลผลออกมาเป็นค่า ตามที่เราสั่งให้แสดงออกมาโดยเพิ่มโปรแกรมให้แสดงค่าในส่วนของฟังก์ชัน CONVERT จะได้แสดงดังรูปที่ 4.22

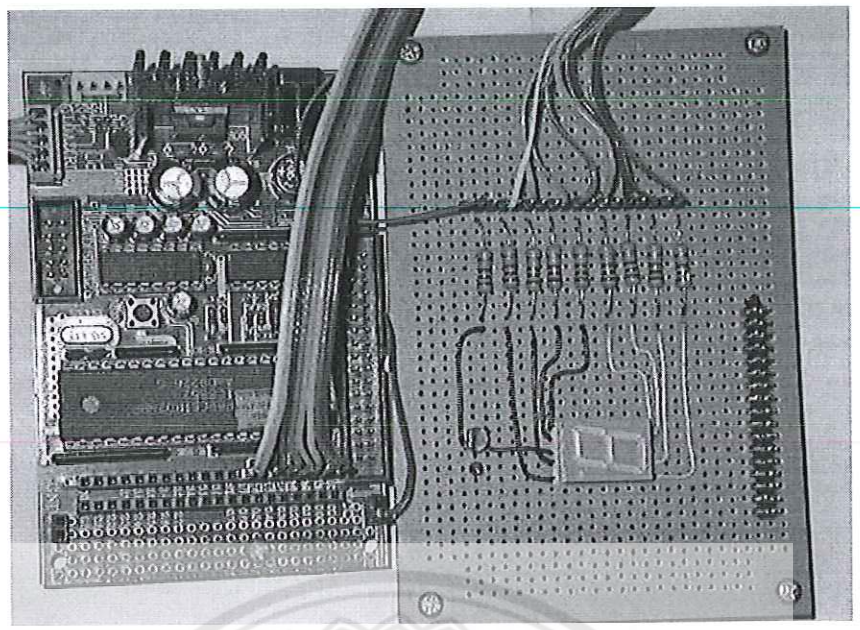


รูปที่ 4.22 การแสดงผลของโปรแกรมจากการกดรีโมทหมายเลข 0 – 9

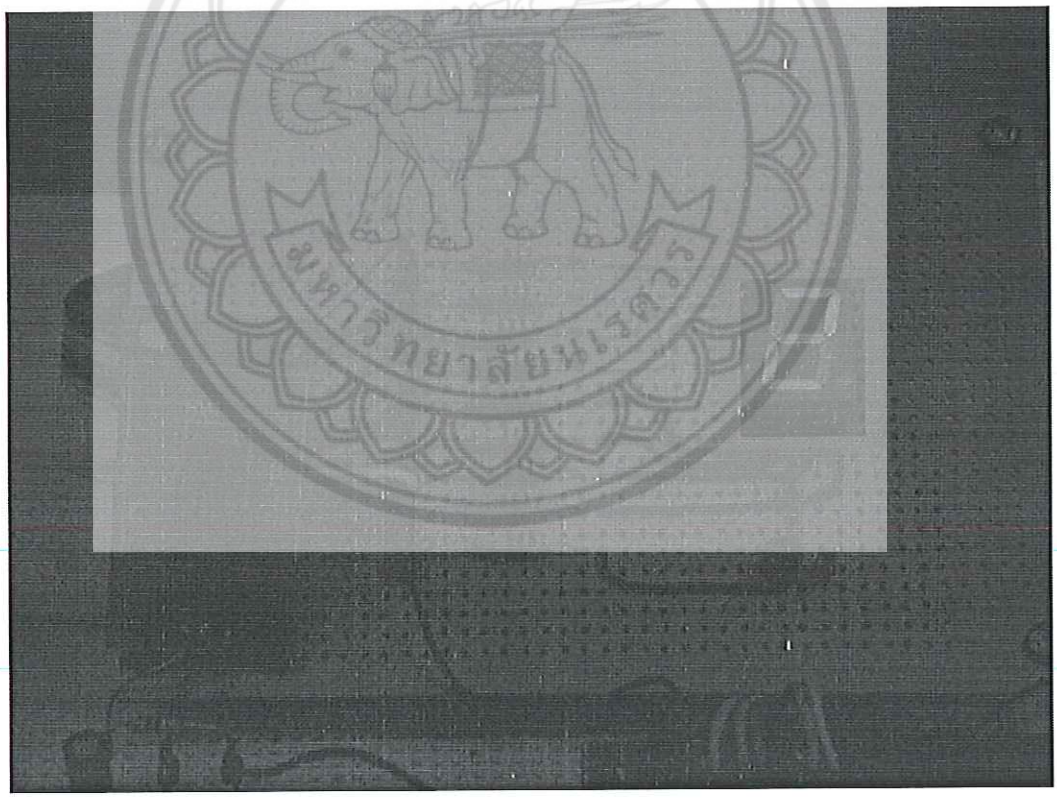
เมื่อเราจับสัญญาณได้แล้วว่าเวลาเรากดหมายเลขของรีโมทหมายเลขไหนก็ให้แสดงผลออกมาทางโปรแกรม HYPER TERMINAL ดังนั้นเราก็จะมาให้แสดงออกทาง7- SEGMENT คือจะต่อตามรูป



รูปที่ 4.23 การต่อบอร์ดไมโครคอนโทรลเลอร์กับ 7- SEGMENT



รูปที่ 4.24 การต่อบอร์ดไมโครคอนโทรลเลอร์กับ7- SEGMENT



รูปที่ 4.25 การแสดงผลของการส่งสัญญาณรีโมทออกทาง 7- SEGMENT

บทที่ 5
สรุปผลและวิเคราะห์ผล

โครงงานนี้เป็นศึกษาการทำงานของอินฟราเรดรีโมทที่ใช้กันโดยทั่วไปโดยไม่เจาะจงว่าจะเป็นรีโมทยี่ห้อใด

เมื่อได้ทำการทดลองและได้ทำการพัฒนาโครงงานนี้ ทำให้พบข้อมูลบางสิ่งบางอย่างจากการพัฒนาครั้งนี้ และพบปัญหาที่เกิดขึ้นขณะดำเนินการและการใช้งานในบางประการ อันเนื่องมาจากหลายสาเหตุทำให้สามารถสรุปผลของโครงงานนี้ออกเป็นส่วนๆ ดังนี้คือ

5.1 สรุปผลการปฏิบัติงาน

จากการที่ได้ใช้หลักการและทฤษฎีที่ได้ศึกษามา การที่เราจะสามารถจับสัญญาณอินฟราเรดรีโมทได้และสามารถแยกสัญญาณแต่ละปุ่มกดให้มีความแตกต่างกันได้ต้องอาศัยหลักการดังนี้คือ

1. สัญญาณที่ได้ต้องมีสถานะที่แน่นอนคือสัญญาณที่ได้จะต้องไม่ดรอปจะต้องให้ค่าเป็น “1” หรือ +5Vdc จะเป็น 2.5 +Vdc ไม่ได้ เพราะโปรแกรมจะปรับให้เป็น 0 ไม่ใช่ 1
2. การกดรีโมทนั้นจะต้องทำการกดรีโมทโดยการให้ตัวส่งสัญญาณกับตัวรับสัญญาณหันหน้าเข้าหากันคือ ถ้าหันหน้าของตัวรีโมทไปทางอื่นสัญญาณที่ได้อาจไม่ถูกต้องอาจ ดรอปลงได้
3. ทางด้านซอฟต์แวร์ ในการประมวลผลสัญญาณในโปรแกรมนี้จะสามารถที่จะรองรับการกดได้ 10 ปุ่มกดถ้าต้องการเพิ่มปุ่มกดจะต้องทำการเพิ่มโปรแกรมในส่วนของฟังก์ชัน CUTARRAY และ ตรง ข้อมูลที่ Xdata แล้วเพิ่มในส่วนของการ CHECKCODE เข้าไปก็สามารถที่เพิ่มการกดของปุ่มกดรีโมทให้มากได้
4. นำข้อมูลจากการจับสัญญาณโดยใช้โปรแกรมและจากการใช้ออสซิลโลสโคปมาเปรียบเทียบกับจะพบว่าค่าที่ได้หรือลูกคลื่นที่ได้จะมีความสอดคล้องกัน
5. ในการแสดงผลออกทาง 7 - SEGMENT อาจใช้ไอซีขับกระแสและตัวต้านทางมาต่อเพื่อป้องกันความเสียหายทาง พอร์ตไมโครคอนโทรเลอร์และ 7 - SEGMENT

5.2 ข้อเสนอและแนวทางแก้ไข

1. เนื่องจากในการทดลองนั้นได้ทำการซื้อบอร์ดไมโครคอนโทรลเลอร์เพื่อนำมาศึกษาของการทำงานโดยการเปรียบกับตัวรับสัญญาณที่ได้ทำขึ้นมาว่าสัญญาณที่ได้มีคุณสมบัติที่เหมือนกันหรือไม่ทำให้มีค่าใช้จ่ายที่สิ้นเปลืองในส่วนนี้
2. เนื่องจากขาดความชำนาญในด้านการใช้โปรแกรม KEIL ทำให้ต้องใช้เวลาในการศึกษาการทำงานของโปรแกรมและการเขียนโปรแกรมนั้นอาจใช้โปรแกรมตัวอื่นทำได้เช่น โปรแกรมแอสเซมบลี
3. อย่าหันหน้าตัวส่งรีโมทไปในทางที่ไม่ถูกต้องคือต้องหันหัวตัวส่งสัญญาณรีโมทให้ตรงกับตัวรับและคอยตรวจสอบแบตเตอรี่ของตัวส่งรีโมทให้อยู่ในสภาพที่พร้อมใช้งานเสมอ เพราะ สัญญาณที่ออกมาอาจไม่ได้ตามต้องการทำให้โปรแกรมไม่สามารถประมวลผลได้
4. ในส่วนของบอร์ดไมโครคอนโทรเลอร์ตอนแรกจะใช้บอร์ด START – C51 แต่เนื่องจากบอร์ดไม่มีพอร์ตออกมาทาง พอร์ตต่างๆ ต้องต่อออกทาง 8255 ซึ่งเป็นการยุ่งยากในการเขียนโปรแกรมจึงทำการใช้บอร์ดไมโครคอนโทรลเลอร์ CP -SPI/S8252 V1 แทนซึ่งทำให้ต่อเข้ากับบอร์ดของ 7 – SEGMENT ได้ง่ายขึ้นตามรูปที่ 4.24 และการลงโปรแกรมซึ่งแสดงในภาคผนวก ค

5.3 ข้อเสนอแนะในการพัฒนา

1. จากการปฏิบัติการจะพบว่าสัญญาณที่สามารถรับได้อย่างมีประสิทธิภาพนั้นยังไม่สามารถที่จะส่งสัญญาณได้ในระยะที่ไกลดังนี้สามารถที่จะพัฒนาต่อไปเพื่อที่จะนำไปใช้ในด้านที่มีประโยชน์ให้กว้างขวางมากยิ่งขึ้น
2. จากการที่เราจะนำสัญญาณเพื่อไปออกทาง 7 –SEGMENT นั้นเราสามารถนำค่าที่ได้ไปทำการสั่งรีเลย์ทำงานหรือใช้เป็น สวิตซ์ เปิด – ปิด อุปกรณ์ทางไฟฟ้าได้

เอกสารอ้างอิง

[1] รศ.ธีรวัฒน์ ประกอบผล , " **ภาษาแอสเซมบลีสำหรับ MCS-51** " , พิมพ์ครั้งที่ 3 , กรุงเทพฯ , สมาคมส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น) , 2546

[2] รศ.ธีรวัฒน์ ประกอบผล , " **การประยุกต์ใช้งานไมโครคอนโทรลเลอร์** " , กรุงเทพฯ , สมาคมส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น)

[3] นคร ภักดีชาติ , ชัยวัฒน์ ลิ้มพรจิตรวิไล , " **ทดลองและใช้งานไมโครคอนโทรลเลอร์ MCS-51 ด้วยโปรแกรมภาษา C ฉบับ P89V51RD2** " , กรุงเทพฯ , บริษัท อินโนเวตีฟ เอ็กเพอริเมนต์ จำกัด

[4] ผศ.ธีรวัฒน์ ประกอบผล , " การพัฒนาไมโครคอนโทรลเลอร์ด้วยภาษาซี " , พิมพ์ครั้งที่ 4 , กรุงเทพฯ , สมาคมส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น) , 2545

[5] " **การรับสัญญาณ Remote จาก Remote โทรทัศน์** " [online] . available : http://www.thaimcu.com/article/articleindex.htm . 2006

[6] อาจารย์บรรพต กำคำ , " **ไมโครโปรเซสเซอร์ (Microprocessor)** " , [online] . available : http://www.iet.rru.ac.th/index.php

[7] **"คู่มือออสซิลโลสโคป รุ่น DL 1520 / 1520L Digital Oscilloscope"** , Japan , YOKOGAWA ELECTRIC CORPORATION

ภาคผนวก

ภาคผนวก ก

โปรแกรม keil51v6.20

1. ปริญญานิพนธ์นี้ใช้โปรแกรม keil51v6.20

This file contains release notes and last minute changes pertainingto uVision2 - Version 6.20 and the related tool chains. Informationin this file, the accompanying manual, and software is Copyright (c)2001 Keil Software and Keil Elektronik. All rights reserved.

Keil uVision2.lnk

รูปที่ ก.1 SHORTCUT โปรแกรม KEIL

ถ้าต้องการ support program สามารถที่จะ support ได้ที่ http://www.keil.com/support

ภาคผนวก ข
ไฟล์โปรแกรม

```
#include <reg51.h>
#include <stdio.h>
/************ Configuration Pin & Variable *****************/

sbit input = P3^4;
unsigned char timehi[49];
xdata unsigned char tmp[16];
xdata unsigned char numzero[16] = {1,0,0,1,1,0,0,0,1,0,0,1,1,0,0,1};
xdata unsigned char numone[16] = {0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,1};
xdata unsigned char numtwo[16] = {1,0,0,0,1,0,0,0,1,0,0,0,1,0,0,1};
xdata unsigned char numthree[16] = {0,1,0,0,1,0,0,0,0,1,0,0,1,0,0,1};
xdata unsigned char numfour[16] = {1,1,0,0,1,0,0,0,1,1,0,0,1,0,0,1};
xdata unsigned char numfive[16] = {0,0,1,0,1,0,0,0,0,0,1,0,1,0,0,1};
xdata unsigned char numsix[16] = {1,0,1,0,1,0,0,0,1,0,1,0,1,0,0,1};
xdata unsigned char numseven[16] = {0,1,1,0,1,0,0,0,0,1,1,0,1,0,0,1};
xdata unsigned char numeight[16] = {1,1,1,0,1,0,0,0,1,1,1,0,1,0,0,1};
xdata unsigned char numnine[16] = {0,0,0,1,1,0,0,0,0,0,0,1,1,0,0,1};
unsigned int i = 0;
unsigned char flag;
/****************** Function ******************************/
void delay (int time)
{
        int time1,time2;
        for (time1=0;time1<time;time1++)
                for (time2=0;time2<time;time2++);
                //P1 = 0xff;
}
```

```
void cutarray(void){
        for(i=0;i<16;i++)
        {
                tmp[i]=timehi[i+33];
        }
}
int iszero(){
        flag = 1;
        for(i=0;i<16;i++)
        {
                if(tmp[i]==numzero[i]){}else{flag = 0;}
        }
        return (int)flag;
}

int isone(){
        flag = 1;
        for(i=0;i<16;i++)
        {
                if(tmp[i]==numone[i]){}else{flag = 0;}
        }
        return (int)flag;
}
int istwo(){
        flag = 1;
        for(i=0;i<16;i++)
        {
                if(tmp[i]==numtwo[i]){}else{flag = 0;}
        }
        return (int)flag;
}
```

```
int isthree(){
        flag = 1;
        for(i=0;i<16;i++)
        {
                if(tmp[i]==numthree[i]){}else{flag = 0;}
        }
        return (int)flag;
}
int isfour(){
        flag = 1;
        for(i=0;i<16;i++)
        {
                if(tmp[i]==numfour[i]){}else{flag = 0;}
        }
        return (int)flag;
}
int isfive(){
        flag = 1;
        for(i=0;i<16;i++)
        {
                if(tmp[i]==numfive[i]){}else{flag = 0;}
        }
        return (int)flag;
}
int issix(){
        flag = 1;
        for(i=0;i<16;i++)
        {
                if(tmp[i]==numsix[i]){}else{flag = 0;}
        }
        return (int)flag;
}
```

```
int isseven(){
        flag = 1;
        for(i=0;i<16;i++)
        {
                if(tmp[i]==numseven[i]){}else{flag = 0;}
        }
        return (int)flag;
}
int iseight(){
        flag = 1;
        for(i=0;i<16;i++)
        {
                if(tmp[i]==numeight[i]){}else{flag = 0;}
        }
        return (int)flag;
}
int isnine(){
        flag = 1;
        for(i=0;i<16;i++)
        {
                if(tmp[i]==numnine[i]){}else{flag = 0;}
        }
        return (int)flag;
}
void checkcode(void){
                if(iszero())
                {P1 = 0xc0;
                        delay(500);
                        }
                if(isone())
                {P1 = 0xf9;
                        delay(500);
```

```
        }
if(istwo())
{P1 = 0xa4;
        delay(500);
        }
if(isthree())
{P1 = 0xb0;
        delay(500);
        }
if(isfour())
{P1 = 0x99;
        delay(500);
        }
if(isfive())
{P1 = 0x92;
        delay(500);
        }
if(issix())
{P1 = 0x82;
        delay(500);
        }
if(isseven())
{P1 = 0xf8;
        delay(500);
        }
if(iseight())
{P1 = 0x80;
        delay(500);
        }
if(isnine())
{P1 = 0x90;
        delay(500);
```

```
                }
        }
void convert(void){
        for(i=0;i<49;i++){
                if((int)timehi[i]>5){timehi[i]=1;}else{timehi[i]=0;}
                printf("%d",(int)timehi[i]);
        }
}
void getcode(void){
                while(input){}
                for(i=0;i<49;i++){
                        while(!input){}
                                TR0 = 1;
                        while(input){}
                                TR0 = 0;
                                TF0 = 0;
                                timehi[i]=TH0;
                                TH0 = 0x00;
                                TL0 = 0x00;
                }
}
void initialtimer(void){
        TMOD = (TMOD & 0xF0) | 0x01;  /* Set T/C0 Mode */
        TH0 = 0;
        TL0 = 0;
        TF0 = 0;
}
void main()
{
        P1 = 0xbf;
        delay(100);
        initialtimer();
```

```
        while(1){
                getcode();
                convert();
                cutarray();
                checkcode();
        }
}
```

ภาคผนวก ค

การลงโปรแกรม

บอร์ดไมโครคอนโทรลเลอร์ CP -SPI/S8252 V3

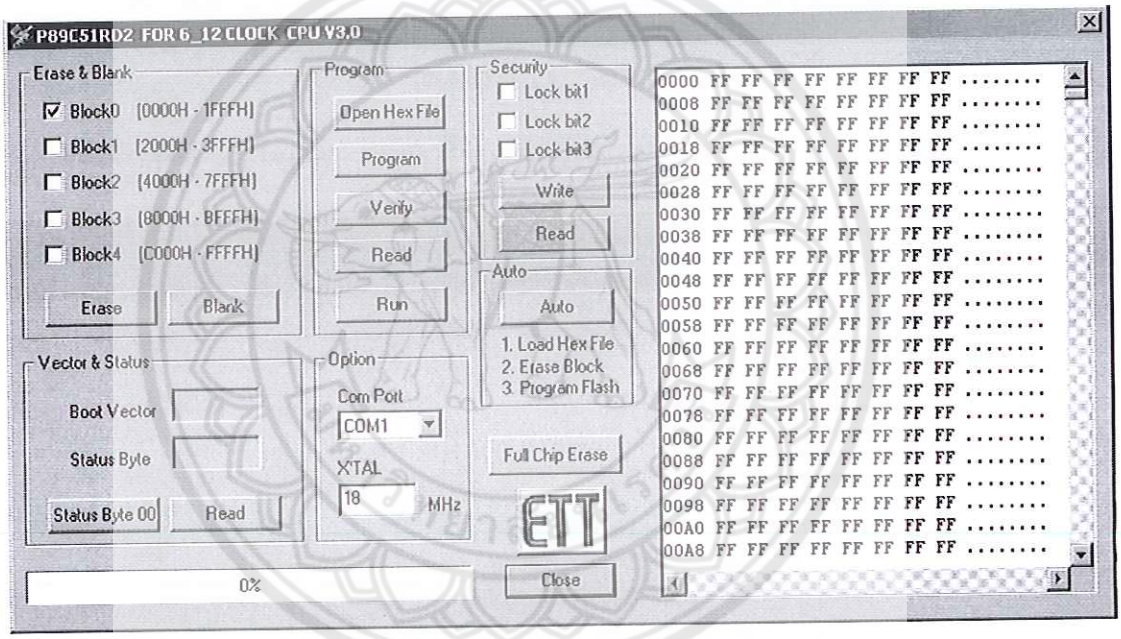
บริษัท อีทีที จำกัด(ETT Co., LTD) โปรแกรม P89C51RD2V3 เวอร์ชัน 3

การลงโปรแกรมเริ่มจากทำการคลิกที่ SHORTCUT ดังรูป

P89C51RD2V3.exe

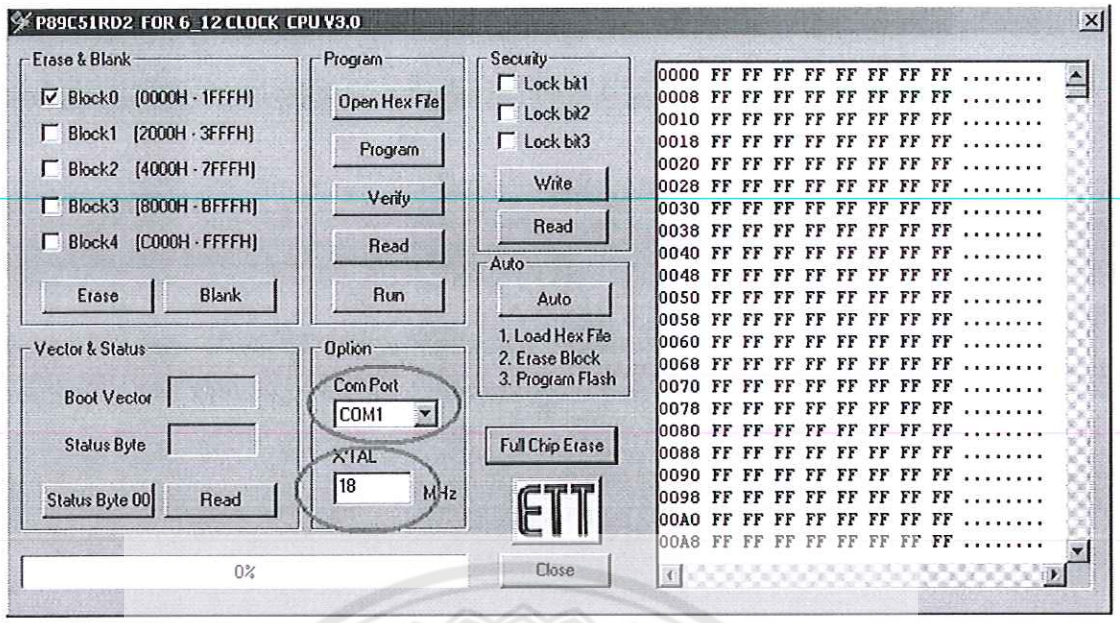
รูปที่ ค.1 SHORTCUT ตัวลงโปรแกรมบอร์ด

จะแสดงหน้าจอดังรูป

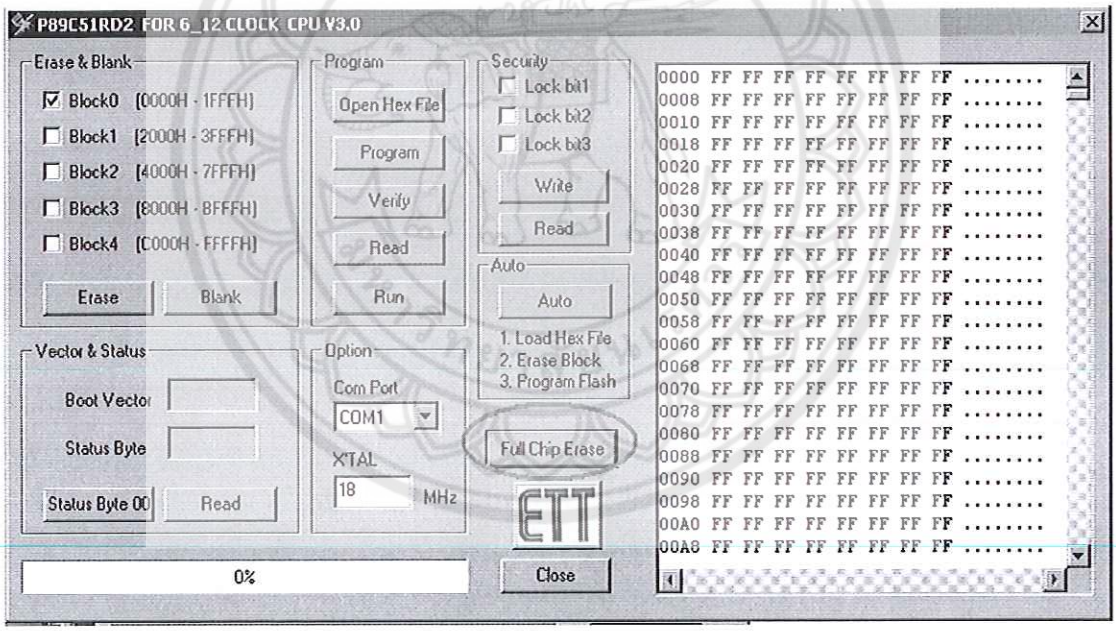


รูปที่ ค.2 หน้าจอตัวลงโปรแกรม

รูปที่ ค.3 ตัวตรวจสอบการติดตั้งโปรแกรมและบอร์ด

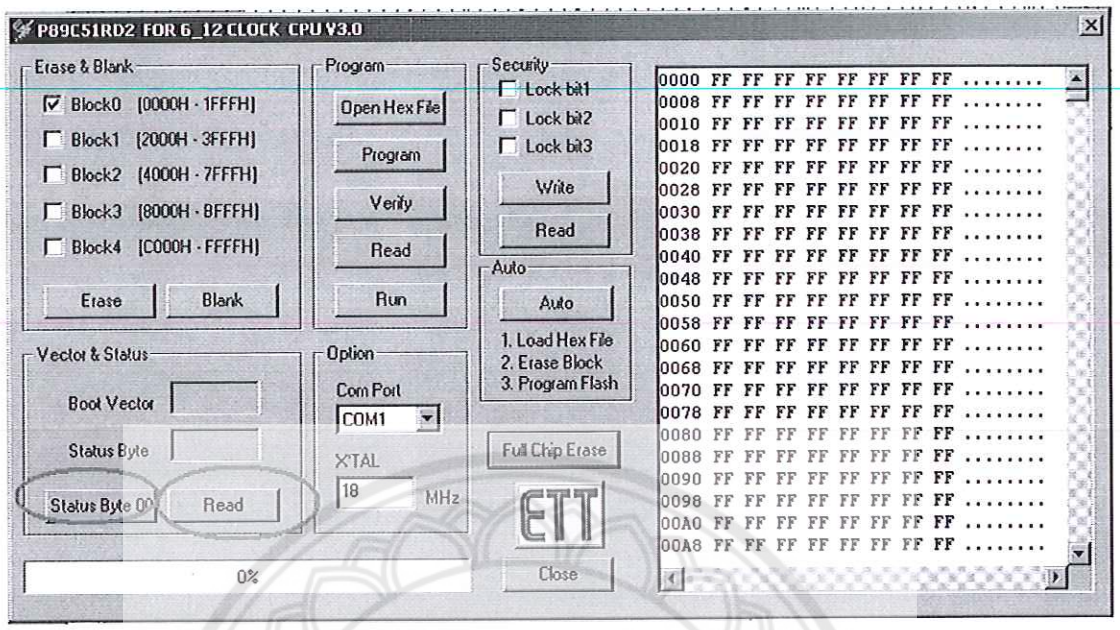


รูปที่ ค.4 ตรวจสอบ COM PORT และค่า XTAL



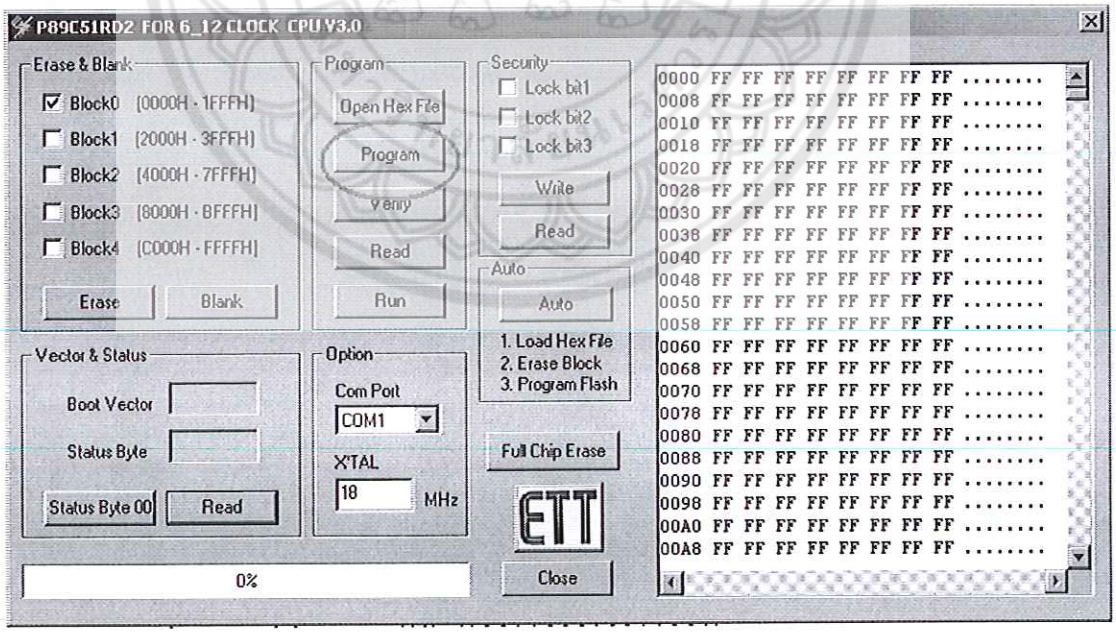
รูปที่ ค.5 การลบโปรแกรมเก่าที่อยู่ในบอร์ด

ทำการคลิกที่ STATUS BYTE 00 และ READ ตามลำดับ



รูปที่ ค.6 การ STATUS BYTE และ READ ข้อมูล

ทำการคลิกที่PROGRAM จะทำการลงโปรแกรมจนเสร็จสมบูรณ์



รูปที่ ค.7 การลงโปรแกรมขั้นสุดท้าย

ภาคผนวก ง

IRM-8601S [6]

Features

- High protection ability against EMI .
- Oval lens to improve the receive angles.
- Line-up for various center carrier frequencies.
- Low voltage and low power consumption.
- High immunity against ambient light.
- Photodiode with integrated circuit.
- TTL and CMOS compatibility.
- Long reception distance.
- High sensitivity.

Descriptions

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

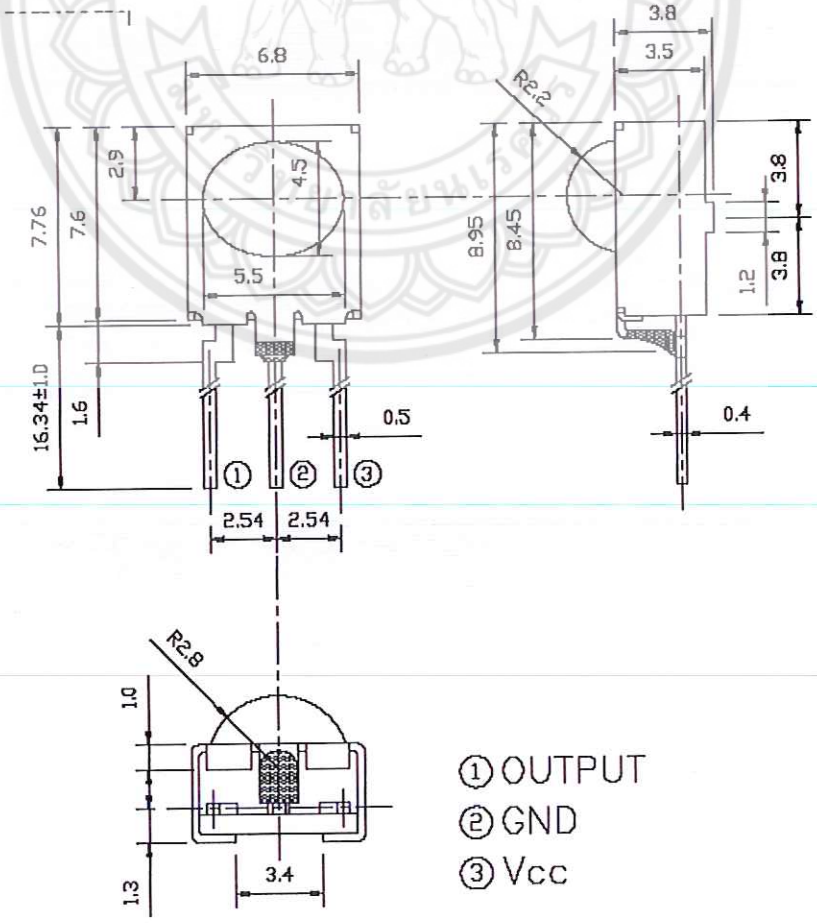
Applications

1. Optical switch
2. Light detecting portion of remote control
   - AV instruments such as Audio, TV, VCR, CD, MD, etc.
   - Home appliances such as Air-conditioner, Fan , etc.
   - The other equipments with wireless remote control.
   - CATV set top boxes
   - Multi-media Equipment

Device Selection Guide

PART	MATERIAL	COLOR
Chip	Silicon	------
Metal can	Tinplate	Silver-white
Package	Epoxy	Black

Package Dimensions



Notes: 1. All dimensions are in millimeters.

2. Tolerances unless dimensions ±0.3mm.

Absolute Maximum Ratings (Ta=25°C)

Parameter	Symbol	Rating	Unit	Notice
Supply Voltage	Vcc	0~6	V	
Operating Temperature	Topr	-25 ~ +85	℃	
Storage Temperature	Tstg	-40 ~ +85	℃	
Soldering Temperature	Tsol	260	℃	4mm from mold body less than 10 seconds

Recommended Operating Condition Supply Voltage Rating: Vcc 4.5V to 5.5V

Electro-Optical Characteristics (Ta=25°C, and Vcc=5 V)

Parameter	Symbol	MIN.	TYP.	MAX.	Unit	Condition
Consumption Current	Icc	---	---	3	mA	No signal input
B.P.F Center Frequency	Fo	---	38	---	KHz	
Peak Wavelength	λ_p	---	940	---	nm	
Reception Distance	L_0	8	---	---	m	At the ray axis *1
	L_{45}	4	---	---		
Half Angle(Horizontal)	Θ_h	---	45	---	deg	
Half Angle(Vertical)	Θ_v	---	45	---	deg	
High Level Pulse Width	T_H	400	---	800	μs	At the ray axis *2
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	

Notes:

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

*2:A range from 30cm to the arrival distance. Average value of 50 pulses.

Test Method :

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

1. **Measurement place**

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

2. **External light**

Project the light of ordinary white fluorescent lamps which are not high Frequency lamps and must be less then 10 Lux at the module surface. (Ee $\leq$ 10Lux)

3. **Standard transmitter**

A transmitter whose output is so adjusted as to Vo=400mVp-p and the output Wave form shown in Fig.-1.According to the measurement method shown in Fig.-2 the standard transmitter is specified.

However , the infrared photodiode to be used for the transmitter should be λp=940nm,$\Delta\lambda$=50nm. Also, photodiode is used of PD438B(Vr=5V). (Standard light / Light source temperature 2856°K).

4. **Measuring system**

According to the measuring system shown in Fig.-3

Fig.-1 Transmitter Wave Form D.U.T output Pulse

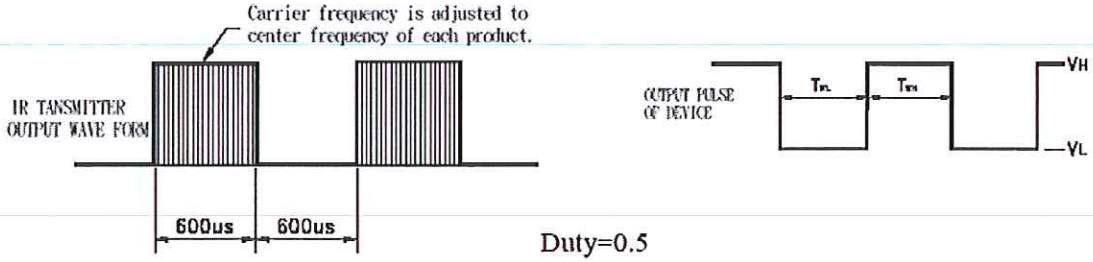


Fig.-2 Measuring Method

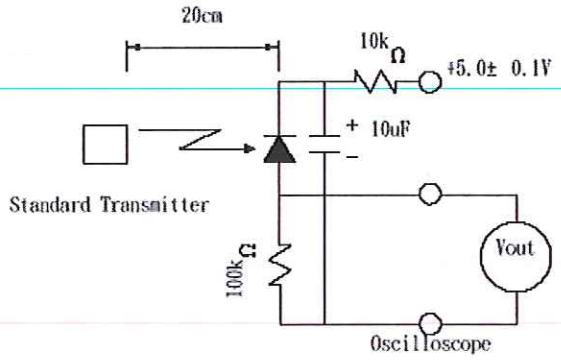
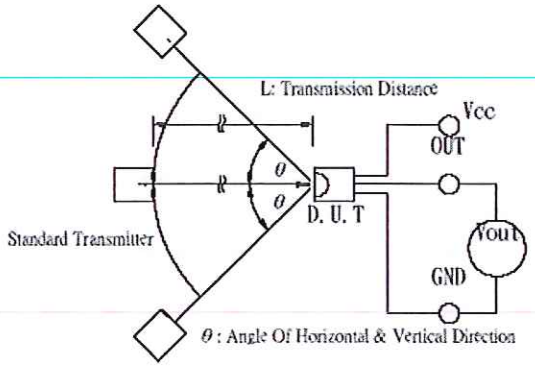
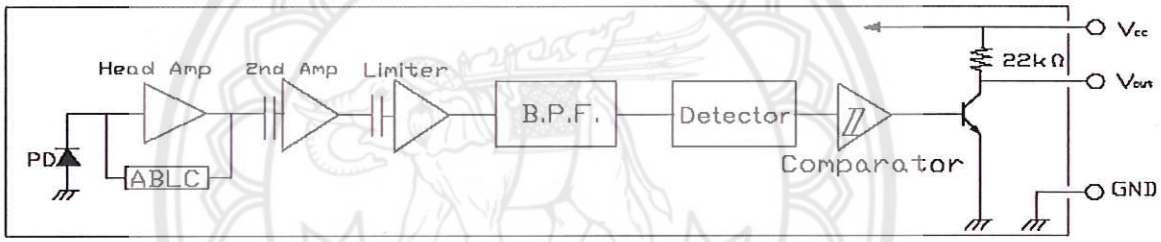


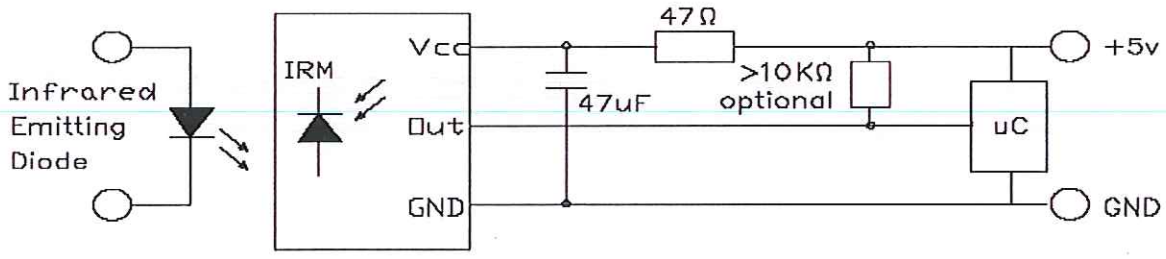
Fig.-3 Measuring System



Block Diagram :



Application Circuit :



RC Filter should be connected closely between Vcc pin and GND pin.

Typical Electro-Optical Characteristics Curves

Typical Electro-Optical Characteristics Curves

Fig.-4 Relative Spectral Sensitivity vs. Wavelength

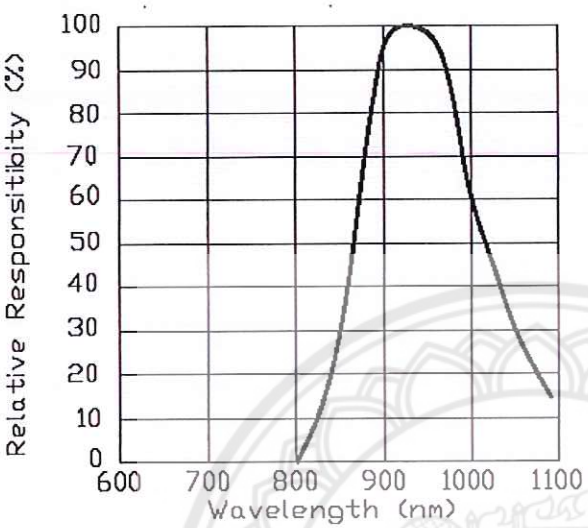


Fig.-5 Relative Transmission Distance vs. Direction

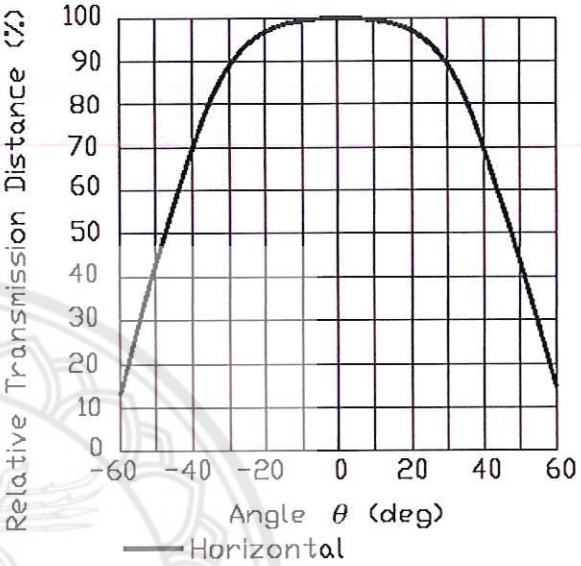


Fig.-6 Output Pulse Length vs. Arrival Distance

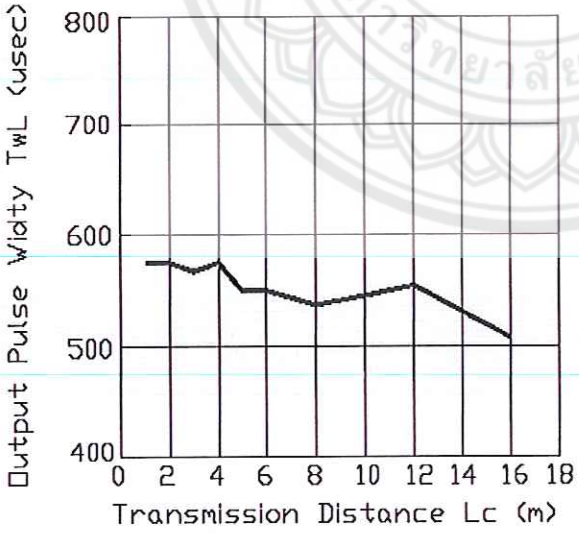
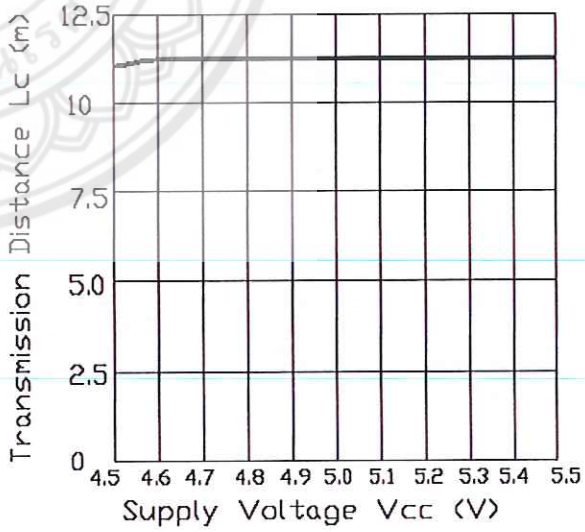


Fig.-7 Arrival Distance vs. Supply Voltage



Typical Electro-Optical Characteristics Curves

Fig.-8 Relative Transmission Distance vs. Center Carrier Frequency

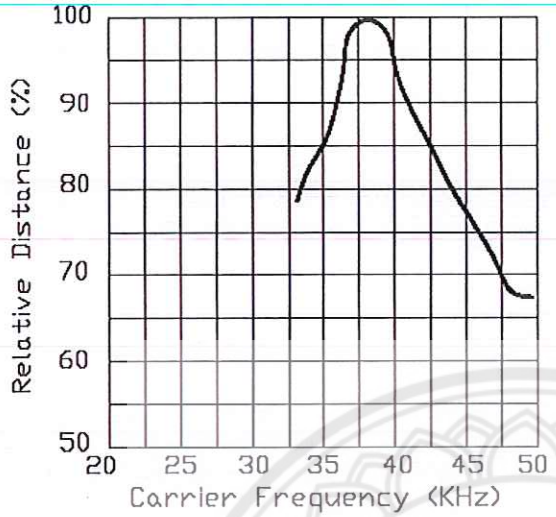
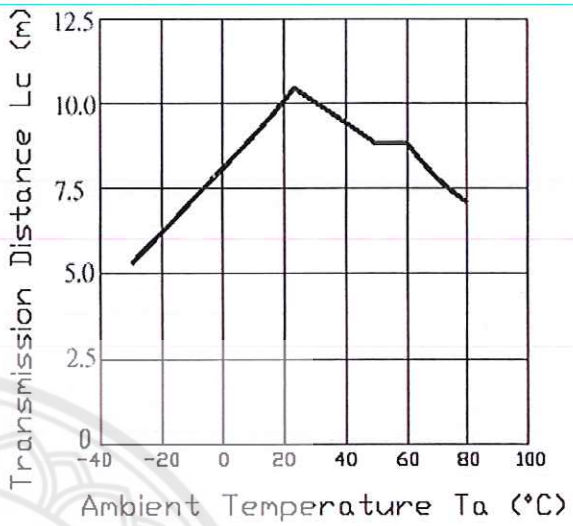


Fig.-9 Arrival Distance vs. Ambient Temperature



Reliability Test Item And Condition

The reliability of products shall be satisfied with items listed below.

Confidence level : 90% LTPD : 10%

Test Items	Test Conditions	Failure Judgement Criteria	Samples(n) Defective(c)
Temperature cycle	1 cycle -25℃ ⟷ +85℃ (30min)(5min)(30min) 300 cycle test	$L_0 \leqq$ Lx0.8 $L_{45} \leqq$ Lx0.8 L: Lower specification limit	n=22,c=0
High temperature test	Temp: +85℃ Vcc:5V 1000hrs		n=22,c=0
Low temperature storage	Temp: -40℃ 1000hrs		n=22,c=0
High temperature High humidity	Ta: 85℃,RH:85% 1000hrs		n=22,c=0
Solder heat	Temp: 260±5℃ 10sec 4mm From the bottom of the package.		n=22,c=0

Packing Quantity Specification

1. 1000 PCS/1Box

2. 10 Boxes/1Carton

Label Form Specification

CPN: Customer's Production Number P/N : Production Number

QTY: Packing Quantity CAT: Ranks

HUE: Peak Wavelength REF: Reference

LOT No: Lot Number

Notes

1. Above specification may be changed without notice. EVERLIGHT will reserve authority on material change for above specification.

2. When using this product, please observe the absolute maximum ratings and the instructions for using outlined in these specification sheets. EVERLIGHT assumes no responsibility for any damage resulting from use of the product which does not comply with the absolute maximum ratings and the instructions included in these specification sheets.

3. These specification sheets include materials protected under copyright of EVERLIGHT corporation. Please don't reproduce or cause anyone to reproduce them without EVERLIGHT's consent.

ประวัติผู้จัดทำโครงงาน

ชื่อ นายคมสันต์ อินต๊ะ

ภูมิลำเนา 115 ม.10 ต.แม่นาเรือ อ.เมือง จ.พะเยา 56000

ประวัติการศึกษา

- ประถมศึกษา จากโรงเรียนบุญสิษฐ์วิทยา
- มัธยมศึกษาตอนปลาย จากโรงเรียนฟากกว๊านวิทยาคม
- ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 4 สาขาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail : nanae_01@hotmail.com

ชื่อ นายนิรันทร์ชัย ทินนา

ภูมิลำเนา 132/2 ม. 4 ต. แม่กา อ. เมือง จ. พะเยา 56000

ประวัติการศึกษา

- ประถมศึกษา จากโรงเรียนบ้านแม่กาโทกหวาก
- มัธยมศึกษาตอนปลาย จากโรงเรียนพะเยาพิทยาคม
- ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 4 สาขาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail : renuko1@hotmail.com

ชื่อ นายเสกสรร อินทร์ทรีย์

ภูมิลำเนา 134 ม.1 ต. สันโค้ง อ. ดอกคำใต้ จ. พะเยา 56120

ประวัติการศึกษา

- ประถมศึกษา จากโรงเรียนบุญสิษฐ์วิทยา
- มัธยมศึกษาตอนปลาย จากโรงเรียนพะเยาพิทยาคม
- ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 4 สาขาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail : sekson_in@hotmail.com