สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบรักษาความปลอดภัยภายในหมู่บ้านไร้สาย

SECURITY SYSTEM OF VILLAGE VIA WIRELESS

โดย

นายดนย์ปภพ แก้ววิลัย

ว่าที่ร.ต.ทรงศักดิ์ แดนดี

นายนัสรัณ สันติวรกุล

เลขหมู่.....
เลขทะเบียน 72755
วัน,เดือน,ปี 22 มิ.ย. 2550

b 11727217x
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2549

ผ่านการตรวจรูปเล่มแล้ว

(ลงชื่อ).....ผู้ตรวจ

ผ่านการตรวจชิ้นงานแล้ว

(ลงชื่อ).....ผู้ตรวจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบรักษาความปลอดภัยภายในหมู่บ้านไร้สาย

SECURITY SYSTEM OF VILLAGE VIA WIRELESS

โดย

นายดนย์ปภพ แก้ววิลัย 47015048

ว่าที่ร.ต. ทรงศักดิ์ แดนดี 47015049

นายนัสรัณ สันติวรกุล 47015056

อาจารย์ที่ปรึกษา

รศ.ดร. กอบชัย เดชหาญ

อ. ศรวัฒน์ ชิวปรีชา

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2549

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง **ระบบรักษาความปลอดภัยภายในหมู่บ้านไร้สาย**

SECURITY SYSTEM OF VILLAGE VIA WIRELESS

ผู้จัดทำ

1. **นายดนย์ปภพ แก้ววิลัย 47015048**
2. **ว่าที่ ร.ต. ทรงศักดิ์ แดนดี 47015049**
3. **นายนัสรัณ สันติวรกุล 47015056**

.....อาจารย์ที่ปรึกษา

(รศ.ดร. กอบชัย เดชหาญ)

.....อาจารย์ที่ปรึกษา

(อ. ศรวัฒน์ ชิวปรีชา)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรื่อง ระบบรักษาความปลอดภัยภายในหมู่บ้านไร้สาย

SECURITY SYSTEM OF VILLAGE VIA WIRELESS

โดย นายคนย์ปภพ แก้ววิลัย 47015048

ว่าที่ร.ต. ทรงศักดิ์ แดนดี 47015049

นายนัสรัณ สันติวรกุล 47015056

อาจารย์ที่ปรึกษา รศ.ดร. กอบชัย เดชหาญ

อ. ศรวัฒน์ ชิวปรีชา

บทคัดย่อ

โครงงานนี้เป็นการนำเสนอระบบรักษาความปลอดภัยภายในหมู่บ้าน โดยใช้ไมโครคอนโทรเลอร์ Rabbit Core Module RCM 2200 มาประยุกต์ใช้งานผ่านเครือข่ายอินเตอร์เน็ท โดยใช้โปรโตคอล TCP/IP สามารถควบคุมอุปกรณ์ไฟฟ้าภายในบ้านได้จากหน้าเว็บเพจ โดยเซ็นเซอร์ที่ใช้เป็นแบบไร้สาย เมื่อมีผู้บุกรุกเข้ามาในบ้านหรือเกิดเหตุไฟไหม้จะทำการส่งข้อความสั้นเตือนภัยไปยังโทรศัพท์เคลื่อนที่ของเจ้าของบ้าน และทำการโทรศัพท์อัตโนมัติไปยังสถานีตำรวจหรือสถานีดับเพลิง และสามารถควบคุมอุปกรณ์ไฟฟ้าภายในบ้านผ่านโทรศัพท์เคลื่อนที่ได้

ABSTRACT

This project presents a security system of village via wireless system by using microcontroller Rabbit RCM2200 to apply with the Internet used protocol TCP/IP. So it could control electricity equipment from webpage used wireless sensor, while the intruder is presence come in the house area or it has some fire the system sends the short message to the mobile phone of the house owner, the system also automatically be called to police station and fire department. It is also able to control the electricity equipments in the house via mobile phone.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์นี้สามารถสำเร็จได้ด้วยดี ก็ด้วยความกรุณาของ รศ.ดร. กอบชัย เดชหาญ และ อ. ศรวัฒน์ ชิวปรีชา ซึ่งกรุณาเป็นอาจารย์ที่ปรึกษาให้และคอยให้คำปรึกษาเรื่องต่างๆ คณะผู้จัดทำขอขอบพระคุณเป็นอย่างสูง ขอขอบคุณอาจารย์ภาควิชาวิศวกรรมโทรคมนาคมทุกท่านที่คอยให้คำชี้แนะในเรื่องต่างๆ ทำให้คณะผู้จัดทำมีความรู้ในด้านต่างๆมากขึ้น ขอขอบคุณเพื่อนๆและน้องๆ ห้อง A,T ที่คอยให้ความช่วยเหลือในเรื่องต่างๆ ไม่ว่าจะเป็นเพื่อนคุย เพื่อนเล่น

สุดท้ายนี้ขอกราบขอบพระคุณ บิดา มารดา ที่คอยเป็นกำลังใจให้เสมอไม่ว่าจะทุกข์เพียงใด หรือมีปัญหามากมายแค่ไหน ท่านก็ไม่เคยที่จะไม่สนใจปัญหาของเราเลย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทที่ 1 บทนำ

1.1 วัตถุประสงค์ 1

1.2 ขอบเขตของโครงงาน 1

บทที่ 2 ทฤษฎีและหลักการ

2.1 ลักษณะที่สำคัญและความสามารถของ Rabbit Core Module RCM 2200 2

2.2 ส่วนประกอบและอุปกรณ์ย่อยของระบบ Rabbit Core Module RCM 2200 3

2.2.1 อินพุท/เอาท์พุทของ Rabbit Core Module RCM 2200 3

2.2.2 ขาสัญญาณอินพุท 3

2.2.3 ขาสัญญาณเอาท์พุท 4

2.2.4 อินพุท/เอาท์พุทที่ติดต่อหน่วยความจำ 7

2.3 รายละเอียดของไมโครโปรเซสเซอร์ Rabbit 2000 7

2.4 หน้าที่ของขาอินพุท/เอาท์พุทบนไมโครโปรเซสเซอร์ Rabbit 2000 8

2.5 พอร์ตในการเชื่อมต่อ Rabbit Core Module RCM 2200 9

2.5.1 พอร์ตอนุกรม 9

2.5.2 อีเทอร์เน็ตพอร์ต 10

2.5.3 พอร์ตการเขียนโปรแกรม 10

2.6 พอร์ตขนาน 11

2.6.1 พอร์ตขนาน A 11

2.6.2 พอร์ตขนาน B 11

2.6.3 พอร์ตขนาน C 12

2.6.4 พอร์ตขนาน D 12

2.6.5 พอร์ตขนาน E 13

2.7 โปรแกรม Dynamic C 13

2.7.1 Using Dynamic C 14

2.7.2 การเซ็ตค่า IP Addresses 14

2.7.3 IP Addresses Set Dynamically 14

2.7.4 BOOTP/DHCP Control Macros 15

2.7.5 Sizes for TCP/IP I/O Buffers 15

2.7.6 Number of Sockets 15

2.7.7 Passive Open 15

2.7.8 Active Open 15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
2.7.9 Delay a Connection	16
2.7.10 Skeleton Program	16
2.7.11 TCP/IP Stack Initialization	16
2.7.12 Packet Processing	16
2.7.13 Function Reference	16
2.7.14 Macros	17
2.8 HTTP Server	17
2.9 Configuration Macros	18
2.10 ทฤษฎีเกี่ยวกับเครือข่ายเน็ตเวิร์ค	19
2.10.1 ลำดับชั้นของกฎการสื่อสารข้อมูล	19
2.10.2 รูปแบบระบบเครือข่าย TCP/IP	21
2.10.3 ชั้นการสื่อสารระบบ TCP/IP	22
2.10.4 การกำหนด IP Address	24
2.11 คำสั่ง AT commands	25
2.11.1 AT คำสั่งเช็คความพร้อมของโทรศัพท์เคลื่อนที่	25
2.11.2 ATD คำสั่งที่ใช้สำหรับการโทรออก	25
2.11.3 ATH คำสั่งที่ใช้ในการวางสายโทรศัพท์ ยกเลิกการโทรออก	26
2.11.4 AT+CMGS คำสั่งที่ใช้สำหรับส่งข้อความ	26
2.11.5 AT+CMNI คำสั่งที่ทำให้รู้ว่าได้รับข้อความเข้ามาที่โทรศัพท์	26
2.11.6 AT+CMGR คำสั่งที่ใช้สำหรับอ่านข้อความ	26
2.11.7 AT+CMGL คำสั่งเรียกดูข้อความ	27
2.11.8 AT+CMGD คำสั่งที่ใช้สำหรับลบข้อความ	27
2.12 ระบบการทำงานการรับส่งข้อความในโทรศัพท์เคลื่อนที่	27
2.13 ภาคถอดรหัสสัญญาณ DTMF	33
2.13.1 โครงสร้างภายในไอซี MT 8870	34
2.13.2 ฟังก์ชันการทำงานภายในของไอซี MT 8870	35
2.14 โครงข่ายของระบบเครือข่าย (Topology)	39
2.14.1 แบบดาว (Star)	39
2.14.2 แบบวงแหวน (Ring)	40
2.14.3 แบบบัส (Bus)	40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
2.15 RF Module TRW 2.4 กิกะเฮริต์	40
2.15.1 การตั้งค่าการใช้งาน RF Module TRW 2.4 กิกะเฮริต์	40
บทที่ 3 การคำนวณและการสร้าง	
3.1 การออกแบบวงจรหลัก	46
3.2 อุปกรณ์ตรวจจับการเปิด / ปิด	46
3.3 อุปกรณ์ตรวจจับควัน	47
3.4 แหล่งจ่ายไฟหลักและไฟสำรอง	48
3.5 ไอซี TA7279P	49
3.6 การนำไอซี 74LS139 ไปใช้งาน	51
3.7 รีเลย์	52
3.8 สาย UTP (Unshieled Twisted Pair)	55
3.9 วิธีหา IP Address : Windows 2000/XP	57
3.10 หลักการอินเตอร์เฟสโทรศัพท์เคลื่อนที่	57
3.11 ขั้นตอนการออกแบบ Web Server	59
3.12 โฟลว์ชาร์ตการทำงานหลัก MCS – 51 ควบคุม RF Module Tx	60
3.13 โฟลว์ชาร์ตการทำงานหลัก MCS – 51 ควบคุมโทรศัพท์เคลื่อนที่	62
3.14 โฟลว์ชาร์ตการทำงานหลัก Rabbit Core Module RCM 2200	64
บทที่ 4 การทดลอง และผลการทดลอง	
4.1 อุปกรณ์ที่ใช้ในการทดลอง	68
4.2 การทดลองการรับค่าจากระบบเตือนภัยในบ้าน	71
4.3 การทดลองการวัดสัญญาณ DTMF จากโทรศัพท์เคลื่อนที่	74
4.4 การทดลองการสั่งงานอุปกรณ์ไฟฟ้าผ่านโทรศัพท์เคลื่อนที่	87
4.5 แสดงผลการวัดขาสัญญาณ (ขา Data , ขา Clk) ของ RF Module TRW 2.4 กิกะเฮริต์	89
บทที่ 5 บทสรุปและวิจารณ์	
5.1 สรุปปัญหาที่พบในโครงงาน	97
5.2 สรุปผลของการดำเนินโครงงาน	97
5.3 แนวทางการพัฒนาโครงงาน	97
ภาคผนวก	
กิตติกรรมประกาศ	
เอกสารอ้างอิง	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

หน้า

รูปที่ 2.1 บอร์ดวงจร Rabbit Core Module RCM 2200 2

รูปที่ 2.2 ระบบภายใน Rabbit Core Module RCM 2200 2

รูปที่ 2.3 หัวข้อต่อสัญญาณที่ต่อออกใช้งานภายนอกของ Rabbit Core Module RCM 2200 3

รูปที่ 2.4 การติดต่อกับหน่วยความจำ (RAM) 7

รูปที่ 2.5 โครงสร้างภายในของ Rabbit 2000 8

รูปที่ 2.6 บล็อกไดอะแกรมของพอร์ตอนุกรม 10

รูปที่ 2.7 พอร์ตการเขียนโปรแกรม 11

รูปที่ 2.8 ชั้นสื่อสาร,กฎการสื่อสารข้อมูลและการเชื่อมต่อ 20

รูปที่ 2.9 กฎการสื่อสารในรูปแบบ TCP/IP 24

รูปที่ 2.10 โครงสร้างที่อยู่แบบ IP 25

รูปที่ 2.11 การรับส่งข้อความแบบพีดียูโหมด 31

รูปที่ 2.12 แสดงรายละเอียดขาของไอซี MT 8870 34

รูปที่ 2.13 แสดงโครงสร้างภายในของไอซี MT 8870 35

รูปที่ 2.14 แสดงวงจรตรวจสอบสัญญาณและกำหนดเวลาการ์ดไทม์ 36

รูปที่ 2.15 แสดงการต่อวงจรภาคอินพุท 37

รูปที่ 2.16 แสดงการต่อวงจร Oscillator 37

รูปที่ 2.17 แสดงวงจรใช้งานของไอซี MT 8870 38

รูปที่ 2.18 แสดงลักษณะด้านบนของตัว RF Module TRW 2.4 กิกะเฮริต์ 40

รูปที่ 2.19 แสดงลักษณะด้านล่างของตัว RF Module TRW 2.4 กิกะเฮริต์ 41

รูปที่ 2.20 แสดงลักษณะขาต่างๆของตัว RF Module TRW 2.4 กิกะเฮริต์ 41

รูปที่ 3.1 Block diagram 46

รูปที่ 3.2 รูปร่างของสวิทช์แม่เหล็ก 47

รูปที่ 3.3 โครงสร้างของสวิทช์แม่เหล็ก 47

รูปที่ 3.4 วงจรเซ็นเซอร์ 47

รูปที่ 3.5 วงจรตรวจสอบควัน 48

รูปที่ 3.6 หลักการของระบบจ่ายไฟหลักและไฟสำรอง 48

รูปที่ 3.7 วงจรแหล่งจ่ายไฟหลักและไฟสำรอง 49

รูปที่ 3.8 ขาของไอซี TA7279P และโครงสร้างภายใน 50

รูปที่ 3.9 การต่อไอซี TA7279P เพื่อใช้งานในวงจรรวม 51

รูปที่ 3.10 ตำแหน่งขาของไอซี 74LS139 และโครงสร้างภายใน 51

รูปที่ 3.11 การต่อไอซี 74LS139 เพื่อใช้งาน 52

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

	หน้า
รูปที่ 3.12 แสดงสัญลักษณ์ของรีเลย์ที่ใช้งาน	53
รูปที่ 3.13 แสดงการใช้งานรีเลย์ในการควบคุมหลอดไฟฟ้า	54
รูปที่3.14 แสดงลักษณะของสาย UTP	55
รูปที่ 3.15 แสดงการหาค่า IP	57
รูปที่ 3.16 แสดงค่า IP	57
รูปที่ 3.17 แสดงจุดเชื่อมต่อ	58
รูปที่ 3.18 โฟลว์ชาร์ตการทำงานของโปรแกรมหลักควบคุม Module Tx	60
รูปที่ 3.19 โฟลว์ชาร์ตการทำงานของแต่ละส่วน	61
รูปที่ 3.20 โฟลว์ชาร์ตการทำงานของโทรศัพท์เคลื่อนที่	62
รูปที่ 3.20 โฟลว์ชาร์ตการทำงานของโทรศัพท์เคลื่อนที่ (ต่อ)	63
รูปที่ 3.21 โฟลว์ชาร์ตการทำงานหลักของ Rabbit Core Module RCM 2200	64
รูปที่ 3.22 โฟลว์ชาร์ตการทำงานส่วนของ Module Rx Scan	65
รูปที่ 3.22 โฟลว์ชาร์ตการทำงานส่วนของ Module Rx Scan (ต่อ)	66
รูปที่ 3.23 โฟลว์ชาร์ตการทำงานส่วนของสวิทช์ Scan	67
รูปที่ 4.1 วงจรเซ็นเซอร์และภาคส่งสัญญาณไร้สาย	68
รูปที่ 4.2 วงจรเพาเวอร์ซัพพลายและรีเลย์ของวงจรเซ็นเซอร์และภาคส่งสัญญาณไร้สาย	69
รูปที่ 4.3 วงจรภาครับสัญญาณไร้สาย	69
รูปที่ 4.4 วงจรเพาเวอร์ซัพพลายและรีเลย์ของวงจรภาครับสัญญาณไร้สาย	70
รูปที่ 4.5 การติดตั้งระบบเซ็นเซอร์	70
รูปที่ 4.6 หน้าเว็บเพจขณะสถานะปกติ	71
รูปที่ 4.7 เมื่อประตูถูกเปิดออก	71
รูปที่ 4.8 โทรศัพท์ที่อยู่กับวงจรเซ็นเซอร์ทำการโทรไปยังเจ้าของบ้าน	72
รูปที่ 4.9 แสดงการโทรเข้าที่โทรศัพท์ปลายทางเมื่อมีผู้บุกรุก	72
รูปที่ 4.10 แสดงข้อความสั้นเมื่อประตูที่ 1 มีผู้บุกรุกที่เครื่องโทรศัพท์เคลื่อนที่ปลายทาง	73
รูปที่ 4.11 แสดงผลหน้าเว็บเพจเมื่อประตูที่ 1 มีผู้บุกรุก	73
รูปที่ 4.12 แสดงข้อความสั้นเมื่อหน้าต่างที่ 1 มีผู้บุกรุกที่เครื่องโทรศัพท์เคลื่อนที่ปลายทาง	74
รูปที่ 4.13 สัญญาณขาที่ 11 ของโทรศัพท์เคลื่อนที่เครื่องรับเมื่อต้นทางกดหมายเลข 1	75
รูปที่ 4.14 สัญญาณขาที่ 11 ของโทรศัพท์เคลื่อนที่เครื่องรับเมื่อต้นทางกดหมายเลข 2	75
รูปที่ 4.15 สัญญาณขาที่ 11 ของโทรศัพท์เคลื่อนที่เครื่องรับเมื่อต้นทางกดหมายเลข 3	76
รูปที่ 4.16 สัญญาณขาที่ 11 ของโทรศัพท์เคลื่อนที่เครื่องรับเมื่อต้นทางกดหมายเลข 4	76

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

หน้า

รูปที่ 4.17 สัญญาณขาที่ 11 ของโทรศัพท์เคลื่อนที่เครื่องรับเมื่อต้นทางกดหมายเลข 5 77

รูปที่ 4.18 สัญญาณขาที่ 11 ของโทรศัพท์เคลื่อนที่เครื่องรับเมื่อต้นทางกดหมายเลข 6 77

รูปที่ 4.19 สัญญาณขาที่ 11 ของโทรศัพท์เคลื่อนที่เครื่องรับเมื่อต้นทางกดหมายเลข 7 78

รูปที่ 4.20 สัญญาณขาที่ 11 ของโทรศัพท์เคลื่อนที่เครื่องรับเมื่อต้นทางกดหมายเลข 8 78

รูปที่ 4.21 สัญญาณขาที่ 11 ของโทรศัพท์เคลื่อนที่เครื่องรับเมื่อต้นทางกดหมายเลข 9 79

รูปที่ 4.22 สัญญาณขาที่ 11 ของโทรศัพท์เคลื่อนที่เครื่องรับเมื่อต้นทางกดหมายเลข 0 79

รูปที่ 4.23 สัญญาณขาที่ 11 ของโทรศัพท์เคลื่อนที่เครื่องรับเมื่อต้นทางกดหมายเลข # 80

รูปที่ 4.24 สัญญาณขาที่ 11 ของโทรศัพท์เคลื่อนที่เครื่องรับเมื่อต้นทางกดหมายเลข * 80

รูปที่ 4.25 แสดงสัญญาณ Q1 ของไอซี MT 8870 เมื่อกดเลข 1 ที่โทรศัพท์เคลื่อนที่ต้นทาง 81

รูปที่ 4.26 แสดงสัญญาณ Q2 ของไอซี MT 8870 เมื่อกดเลข 2 ที่โทรศัพท์เคลื่อนที่ต้นทาง 81

รูปที่ 4.27 แสดงสัญญาณ Q1 , Q2 ของไอซี MT 8870 เมื่อกดเลข 3 ที่โทรศัพท์เคลื่อนที่ต้นทาง 82

รูปที่ 4.28 แสดงสัญญาณ Q3 ของไอซี MT 8870 เมื่อกดเลข 4 ที่โทรศัพท์เคลื่อนที่ต้นทาง 82

รูปที่ 4.29 แสดงสัญญาณ Q1 , Q3 ของไอซี MT 8870 เมื่อกดเลข 5 ที่โทรศัพท์เคลื่อนที่ต้นทาง 83

รูปที่ 4.30 แสดงสัญญาณ Q2 , Q3 ของไอซี MT 8870 เมื่อกดเลข 6 ที่โทรศัพท์เคลื่อนที่ต้นทาง 83

รูปที่ 4.31 แสดงสัญญาณ Q1 , Q2 , Q3 ของไอซี MT 8870 เมื่อกดเลข 7
ที่โทรศัพท์เคลื่อนที่ต้นทาง 84

รูปที่ 4.32 แสดงสัญญาณ Q4 ของไอซี MT 8870 เมื่อกดเลข 8 ที่โทรศัพท์เคลื่อนที่ต้นทาง 84

รูปที่ 4.33 แสดงสัญญาณ Q1 , Q3 ของไอซี MT 8870 เมื่อกดเลข 9 ที่โทรศัพท์เคลื่อนที่ต้นทาง 85

รูปที่ 4.34 แสดงสัญญาณ Q2 , Q4 ของไอซี MT 8870 เมื่อกดเลข 0 ที่โทรศัพท์เคลื่อนที่ต้นทาง 85

รูปที่ 4.35 แสดงสัญญาณ Q1, Q2 , Q4 ของไอซี MT 8870 เมื่อกดเลข *
ที่โทรศัพท์เคลื่อนที่ต้นทาง 86

รูปที่ 4.36 แสดงสัญญาณ Q3 , Q4 ของไอซี MT 8870 เมื่อกดเลข # ที่โทรศัพท์เคลื่อนที่ต้นทาง 86

รูปที่ 4.37 วงจรควบคุมการปิดหลอดไฟเมื่อกดเลข 3 ที่โทรศัพท์เคลื่อนที่ต้นทาง 87

รูปที่ 4.38 แสดงผลหน้าเว็บเพจเมื่อ LAMP ที่ 3 ปิด 87

รูปที่ 4.39 วงจรควบคุมการเปิดหลอดไฟเมื่อกดเลข 2 ที่โทรศัพท์เคลื่อนที่ต้นทาง 88

รูปที่ 4.40 แสดงผลหน้าเว็บเพจเมื่อ LAMP ที่ 3 เปิด 88

รูปที่ 4.41 แสดงสัญญาณทางด้านส่งของ RF Module TRW 2.4 กิกะเฮริต์ ในกรณีไม่มีผู้บุกรุก 89

รูปที่ 4.42 แสดงสัญญาณทางด้านรับของ RF Module TRW 2.4 กิกะเฮริต์ ในกรณีไม่มีผู้บุกรุก 90

รูปที่ 4.43 แสดงสัญญาณทางด้านส่งของ RF Module TRW 2.4 กิกะเฮริต์
ในกรณีมีผู้บุกรุกประตูที่ 1 90

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

รูปที่ 4.44 แสดงสัญญาณทางด้านรับของ RF Module TRW 2.4 กิกะเฮริต์ ในกรณีมีผู้บุกรุกประตูที่ 1 91

รูปที่ 4.45 แสดงสัญญาณทางด้านส่งของ RF Module TRW 2.4 กิกะเฮริต์ ในกรณีมีผู้บุกรุกประตูที่ 1,2 91

รูปที่ 4.46 แสดงสัญญาณทางด้านรับของ RF Module TRW 2.4 กิกะเฮริต์ ในกรณีมีผู้บุกรุกประตูที่ 1,2 92

รูปที่ 4.47 แสดงสัญญาณทางด้านส่งของ RF Module TRW 2.4 กิกะเฮริต์ ในกรณีมีผู้บุกรุกประตูที่ 1,2,3 92

รูปที่ 4.48 แสดงสัญญาณทางด้านรับของ RF Module TRW 2.4 กิกะเฮริต์ ในกรณีมีผู้บุกรุกประตูที่ 1,2,3 93

รูปที่ 4.49 แสดงสัญญาณทางด้านส่งของ RF Module TRW 2.4 กิกะเฮริต์ ในกรณีมีผู้บุกรุกประตูที่ 1,2,3,4 93

รูปที่ 4.50 แสดงสัญญาณทางด้านรับของ RF Module TRW 2.4 กิกะเฮริต์ ในกรณีมีผู้บุกรุกประตูที่ 1,2,3,4 94

รูปที่ 4.51 แสดงสัญญาณทางด้านส่งของ RF Module TRW 2.4 กิกะเฮริต์ ในกรณีมีผู้บุกรุกประตูที่ 1,2,3,4,5 94

รูปที่ 4.52 แสดงสัญญาณทางด้านรับของ RF Module TRW 2.4 กิกะเฮริต์ ในกรณีมีผู้บุกรุกประตูที่ 1,2,3,4,5 95

รูปที่ 4.53 แสดงสัญญาณทางด้านส่งของ RF Module TRW 2.4 กิกะเฮริต์ ในกรณี LAMP 3 ปิด 95

รูปที่ 4.54 แสดงสัญญาณทางด้านรับของ RF Module TRW 2.4 กิกะเฮริต์ ในกรณี LAMP 3 เปิด 96

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้า
ตารางที่ 2.1 ความสามารถในการนำไปใช้งานของคอนเน็คเตอร์ J4	4
ตารางที่ 2.1 ความสามารถในการนำไปใช้งานของคอนเน็คเตอร์ J4 (ต่อ)	5
ตารางที่ 2.2 ความสามารถในการนำไปใช้งานของคอนเน็คเตอร์ J5	6
ตารางที่ 2.3 โครงสร้าง TCP/IP	22
ตารางที่ 2.4 รายละเอียดของข้อมูลในการส่งข้อความสั้น	26
ตารางที่ 2.4 รายละเอียดของข้อมูลในการส่งข้อความสั้น (ต่อ)	27
ตารางที่ 2.5 มาตรฐานคู่มือ GSM 03.40 และ GSM 03.38	28
ตารางที่ 2.5 มาตรฐานคู่มือ GSM 03.40 และ GSM 03.38 (ต่อ)	29
ตารางที่ 2.5 มาตรฐานคู่มือ GSM 03.40 และ GSM 03.38 (ต่อ)	30
ตารางที่ 2.6 การแปลงฐานสิบหกเป็นเลขฐานสอง	31
ตารางที่ 2.7 การแปลงเลขฐานสองเป็นเลขพีดียูโค๊ด	32
ตารางที่ 2.8 การแปลงเลขพีดียูเป็นเลขฐานสอง	32
ตารางที่ 2.9 การแปลงเลขฐานสองเป็นรหัสแอสกี้โค๊ด	32
ตารางที่ 2.10 แสดงค่าความถี่ประจำหมายเลขในระบบโทรศัพท์แบบกดปุ่ม	33
ตารางที่ 2.11 แสดงตารางค่าถอดรหัสที่ได้จากความถี่ต่างๆ	35
ตารางที่ 2.11 แสดงตารางค่าถอดรหัสที่ได้จากความถี่ต่างๆ (ต่อ)	36
ตารางที่ 2.12 แสดงหน้าที่การทำงานของขา RF Module TRW 2.4 กิกะเฮริ์ต	42
ตารางที่ 2.13 แสดงหน้าที่ของบิต 121 – 120	42
ตารางที่ 2.14 แสดงหน้าที่ของบิต 119 – 104	43
ตารางที่ 2.15 แสดงหน้าที่ของบิต 103 – 24	43
ตารางที่ 2.16 แสดงหน้าที่ของบิต 23 – 16	43
ตารางที่ 2.17 แสดงหน้าที่ของบิต 15 – 8	44
ตารางที่ 2.18 แสดงหน้าที่ของบิต 12 – 10	45
ตารางที่ 2.19 แสดงหน้าที่ของบิต 9 – 8	45
ตารางที่ 2.20 แสดงหน้าที่ของบิต 7 – 0	45
ตารางที่ 3.1 แสดงสภาวะและการทำงานของมอเตอร์	50
ตารางที่ 3.2 แสดงหน้าที่ของสาย UTP	55
ตารางที่ 3.3 แสดงการเชื่อมต่อระหว่างคอมพิวเตอร์กับ Rabbit core Module RCM 2200	56
ตารางที่ 3.4 แสดงการต่อเชื่อมระหว่าง HUB กับ Rabbit core Module RCM 2200	56
ตารางที่ 3.5 พอร์ตของโทรศัพท์เคลื่อนที่ยี่ห้อซีเมนต์รุ่น C35i	58

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

ในปัจจุบันนี้ทุกคนมีภารกิจที่ต้องทำในแต่ละวัน ดังนั้นจำเป็นที่ต้องออกจากบ้านเพื่อทำภารกิจ จึงมีความจำเป็นต้องทิ้งบ้านไว้โดยไม่มีใครดูแล ทำให้ไม่ทราบว่าจะมีภัยที่ไม่พึงปรารถนาเกิดขึ้นที่บ้านหรือสำนักงานเมื่อไร โดยเฉพาะภัยที่สร้างความเสียหายแก่ชีวิตและทรัพย์สิน ด้วยเหตุผลดังที่กล่าวมานี้ จึงได้มีแนวคิดที่จะทำโครงงานที่ใช้ในการเตือนภัยหรือแจ้งเหตุขึ้น เพื่อเพิ่มความไว้วางใจในระดับหนึ่ง ซึ่งระบบการเตือนภัยที่ดีจำเป็นต้องมีความรวดเร็วในการแจ้งเตือนเหตุในทันทีทันใด

สำหรับรูปแบบการทำงาน เมื่อเกิดเหตุการณ์ที่ไม่พึงปรารถนา ระบบก็จะตรวจจับความผิดปกติ โดยจะส่งข้อมูลแบบไร้สาย แล้วส่งสัญญาณเข้า Rabbit Core Module RCM 2200 เพื่อแจ้งสถานะของเซ็นเซอร์ไปที่หน้าจอเว็บเพจอีกส่วนหนึ่งไมโครคอนโทรลเลอร์ทำการควบคุมโทรศัพท์เคลื่อนที่ที่อยู่ต้นทาง ให้ทำการแจ้งเหตุไปยังโทรศัพท์เคลื่อนที่ที่อยู่ปลายทางและทำการส่งข้อความสั้น ในขณะเดียวกันที่ปลายทางก็จะสามารถตรวจสอบสถานะที่เปลี่ยนแปลงที่เกิดขึ้นได้ ว่าเกิดขึ้นที่จุดใดบ้าง โดยตรวจสอบได้ที่เว็บบราวด์เซอร์ในเครือข่ายอินเตอร์เน็ท และสามารถควบคุมอุปกรณ์ไฟฟ้า ทางหน้าเว็บเพจและทางโทรศัพท์เคลื่อนที่ที่อยู่ปลายทางได้อีกด้วย

1.1 วัตถุประสงค์

- เพื่อสร้างระบบรักษาความปลอดภัยภายในหมู่บ้านผ่านทางโทรศัพท์เคลื่อนที่
- เพื่อศึกษาการโทรออกอัตโนมัติและส่งข้อความสั้น (SMS)
- เพื่อศึกษาการสั่งงานอุปกรณ์ไฟฟ้าผ่านโทรศัพท์เคลื่อนที่
- เพื่อศึกษาการทำงานของ Rabbit Core Module RCM 2200
- เพื่อสร้างระบบเตือนภัยในหน้าเว็บเพจ
- เพื่อสร้างระบบควบคุมผ่านทางเครือข่ายอินเตอร์เน็ท

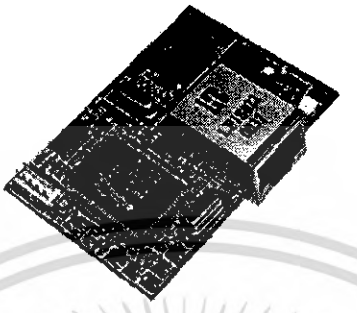
1.2 ขอบเขตของโครงงาน

ระบบรักษาความปลอดภัยภายในหมู่บ้านไร้สาย โดยใช้ไมโครคอนโทรลเลอร์ MCS-51 และ Rabbit Core Module RCM 2200 เป็นตัวเชื่อมต่อและควบคุมการทำงานของระบบ รวมทั้งตรวจสอบสถานะการทำงานของเซ็นเซอร์ผ่านทางเว็บบราวด์เซอร์ แล้วทำการแจ้งเตือนด้วยการโทรออกอัตโนมัติและส่งข้อความสั้น ไปยังโทรศัพท์เคลื่อนที่ปลายทางและสามารถควบคุมอุปกรณ์ไฟฟ้าภายในบ้านผ่านทางหน้าเว็บเพจได้ด้วย

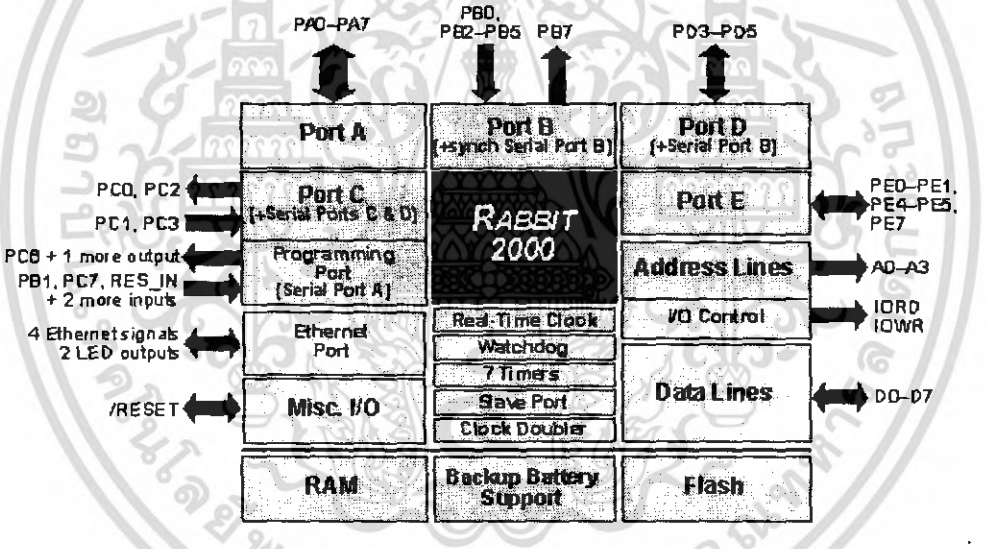
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการ

2.1 ลักษณะที่สำคัญและความสามารถของ Rabbit Core Module RCM 2200



รูปที่ 2.1 บอร์ดวงจร Rabbit Core Module RCM 2200



รูปที่ 2.2 ระบบภายใน Rabbit Core Module RCM 2200

- ขนาดเล็ก 1.60" x 2.30" x 0.86" หรือ (41 mm x 58 mm x 22 mm)
- ไมโครโปรเซสเซอร์ Rabbit 2000 ใช้ความถี่ในการทำงานที่ 22.1 เมกะเฮริ์ต
- มีขาที่เป็น อินพุท/เอาท์พุท ทั้งหมด 26 ขา เป็นแบบขนาน (Parallel)
- มี Data Bus 8 เส้น (D0 - D7)
- มี Address Bus 4 เส้น (A0 - A3)
- มี Flash Memory ขนาด 256 K และ Static Ram ขนาด 128 K

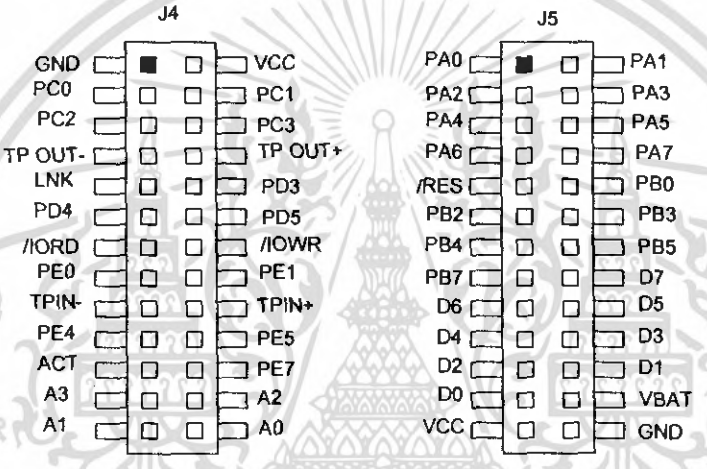
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- มีสัญญาณนาฬิกากำหนดจังหวะการทำงาน
- ใช้หัวต่อสัญญาณแบบ RJ – 45 เป็นอีเทอร์เน็ตพอร์ต (Ethernet Port) พอร์ตอนุกรม (Serial Port) เป็นชนิด CMOS ซึ่งเป็นอัตราบอร์ดเรดแบบ Asynchronous มีความถี่สูงสุดถึง 345,600 บิตต่อวินาที อัตราบอร์ดเรดที่ใช้แบบ Synchronous มีความถี่สูงสุดถึง 138,240 บิตต่อวินาที

2.2 ส่วนประกอบและอุปกรณ์ย่อยของระบบ Rabbit Core Module RCM 2200

2.2.1 อินพุท/เอาท์พุทของ Rabbit Core Module RCM 2200

Rabbit Core Module RCM 2200 มีอินพุท/เอาท์พุท ที่แบ่งเป็นพอร์ต 5 พอร์ต แต่ละพอร์ตมี 8 บิต อยู่ที่คอนเน็คเตอร์ J4 และ J5 โดยที่เส้นอินพุท/เอาท์พุท จะอยู่ที่ PA0 - PA7 , PD3 - PD5 และ PE0 ,PE1 , PE4 , PE5 , PE7 สำหรับคอนเน็คเตอร์ J4 และ J5 ดังแสดงในรูปที่ 2.3



รูปที่ 2.3 คอนเน็คเตอร์ที่ต่อออกใช้งานภายนอกของ Rabbit Core Module RCM 2200

2.2.2 ขาสัญญาณอินพุท

ในส่วนของขาที่ทำหน้าที่เป็นขาอินพุททั่วไปคือ ขา PB0 - PB5 เมื่อ Rabbit 2000 ไม่มีการใช้พอร์ต B หรือ ใช้พอร์ต B ในโหมด Asynchronous มีสี่ขาที่ใช้งานเป็นอินพุทคือ ขา PB2 -PB5 ขาเหล่านี้สามารถใช้สำหรับพอร์ตควบคุม (control port)

ขา PB2 และ PB3 คือขาที่ควบคุมการเขียนและควบคุมการอ่าน strobes ขณะที่ PB4 และ PB5 ใช้สำหรับควบคุมตำแหน่งบัส (address bus) คือขา SA0 , SA1 และใช้เพื่อเข้าถึงข้อมูลใน slave รีจิสเตอร์

ขา PC1 และ PC3 ขาสัญญาณทั้งสองนี้สามารถใช้แทนอินพุทและเอาท์พุทของข้อมูลสำหรับพอร์ต D และ C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3 ขาสัญญาณเอาท์พุท

ขาที่ทำหน้าที่เป็นขาเอาท์พุททั่วไป คือขา PB7 สามารถใช้ควบคุมพอร์ตเอาท์พุทของ SLAVE ATTN ขา PC0 และ PC2 เป็นขาสัญญาณเอาท์พุทเพียงอย่างเดียวสามารถแทนในการส่งออกข้อมูลสำหรับพอร์ต D และ C

ตารางที่ 2.1 ความสามารถในการนำไปใช้งานของคอนเน็คเตอร์ J4

ขาสัญญาณ		ชื่อขาสัญญาณ	Default Use	Alternate Use	Note
Header J4	1	GND			
	2	VCC			
	3	PC0	Output	TXD	
	4	PC1	Input	RXD	
	5	PC2	Output	TXC	
	6	PC3	Input	RXC	
	7	TPOUT-			Ethernet transmit port
	8	TPOUT+			
	9	LNK			Ethernet LNK LED Indicator
	10	PD3	Bitwise or Parallel Programmable I/O		
	11	PD4		ATXB output	
	12	PD5		ARXB input	
	13	/IORD	Input (I/O read strobe)		
	14	/IOWR	Output (I/O write strobe)		
	15	PE0	Bitwise or parallel programmable I/O	I0 control or INT1A input	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 ความสามารถในการนำไปใช้งานของคอนเน็คเตอร์ J4 (ต่อ)

ขาสัญญาณ		ชื่อขาสัญญาณ	Default Use	Alternate Use	Note
Header J4	16	PE1		I1 control or INT1A input	
	17	TPIN-			Ethernet Receiver Port
	18	TPIN+			
	19	PE4	Bitwise or parallel programmable I/O	I4 control or INT0B input	
	20	PE5		I5 control or INT1B input	
	21	ACT			Ethernet active (ACT) LED indicator
	22	PE7	Bitwise or parallel programmable I/O	I7 control or slave port chip select / SCS	
	23-26	A[3:0]			Rabbit 2000 address bus

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

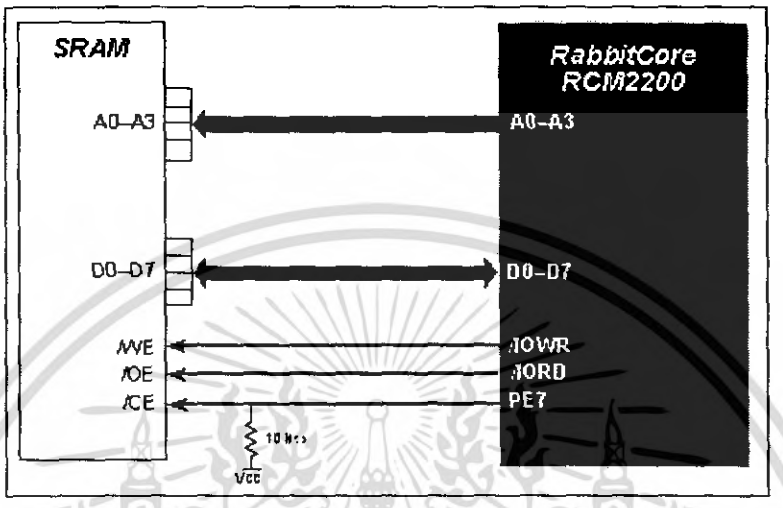
ตารางที่ 2.2 ความสามารถในการนำไปใช้งานของคอนเน็คเตอร์ J5

ขาสัญญาณ		ชื่อขาสัญญาณ	Default Use	Alternate Use	Notes
Header J5	1-8	PA[0:7]	Byte wide programmable parallel I/O	Slave port data bus SD0-SD7	
	9	/RESET	Reset output	Reset input	This weak output can be driven externally
	10	PB0	Input	Serial port clock CLKB input or output	
	11	PB2	Input	Slave port write /SWR	
	12	PB3	Input	Slave port read /SRD	
	13	PB4	Input	SA0	Slave port address lines
	14	PB5	Input	SA1	
	15	PB7	Output	Slave port attention line /SLAVEATTN	
	16-23	D[7:0]	Input/Output		Rabbit 2000 data bus
	24	VBAT	3 V battery input		
	25	Vcc			
	26	Gnd			

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.4 อินพุท/เอาท์พุทที่ติดต่อหน่วยความจำ

ประกอบด้วยตำแหน่งบัส จาก A0 - A3 และบัสข้อมูล (data bus) จาก D0 - D7 สำหรับการติดต่อกับอุปกรณ์ภายนอกจะใช้ตำแหน่งขาสัญญาณ IOWR และ IOWD เป็นอินพุท/เอาท์พุท ในการอ่านและเขียนข้อมูล

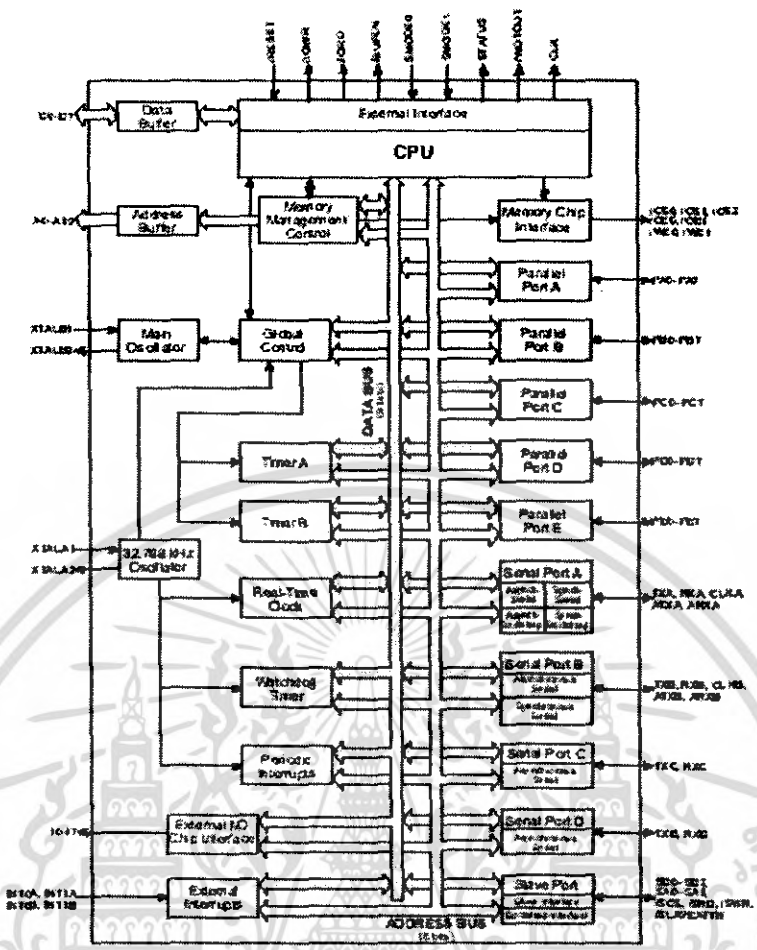


รูปที่ 2.4 การติดต่อกับหน่วยความจำ (RAM)

2.3 รายละเอียดของไมโครโปรเซสเซอร์ Rabbit 2000

ไมโครโปรเซสเซอร์ Rabbit 2000 เป็นไมโครโปรเซสเซอร์ที่ผลิตโดยบริษัท Rabbit Semiconductor Corporation เป็นไมโครโปรเซสเซอร์ขนาด 16 บิตมีความเร็วของสัญญาณนาฬิกา 22.1 เมกะเฮริต์ ถูกออกแบบมาเพื่อให้ใช้กับงานที่เกี่ยวข้องกับระบบ Ethernet โดยเฉพาะ และไมโครโปรเซสเซอร์ Rabbit 2000 ก็เป็นเหมือนหัวใจหลักของ Rabbit Core Module RCM2200 ซึ่งคอยควบคุมการทำงานต่างๆไม่ว่าจะเป็นอุปกรณ์ภายในและภายนอกของ Rabbit Core Module RCM 2200

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 โครงสร้างภายในของ Rabbit 2000

2.4 หน้าที่ของขาอินพุท/เอาท์พุทบนไมโครโปรเซสเซอร์ Rabbit 2000

- CLK เป็นขาชนิดเอาท์พุทที่ป้อนสัญญาณนาฬิกาที่ใช้ในการประมวลผลออกมา
- RESET เป็นขาชนิดอินพุทที่ใช้ในการกำหนดค่าเริ่มต้นหรือการทำงานใหม่
- XTALB1 , XTALB2 เป็นขาชนิดอินพุท/เอาท์พุทที่ใช้สำหรับการเชื่อมต่อกับ Oscillator Quartz Crystal 11.0592 เมกะเฮริ์ต
- XTALA1 , XTALA2 เป็นขาชนิดอินพุท/เอาท์พุทใช้สำหรับการเชื่อมต่อกับ Oscillator Quartz Crystal 32.678 กิโลเฮริ์ต
- A0 -A19 เป็นขาชนิดเอาท์พุททำหน้าที่กำหนดการเลือกใช้และควบคุมหน่วยความจำภายนอก
- D0-D7 เป็นขาชนิดอินพุท/เอาท์พุท ใช้เป็นทางผ่านของข้อมูลระหว่างพอร์ตต่างๆ ของไมโคร โปรเซสเซอร์
- WDTOUT เป็นขาชนิดเอาท์พุทใช้ตรวจสอบว่าสัญญาณลูกคลื่นมีค่าต่ำกว่า 30 ไมโครวินาที หรือไม่ภายในระยะเวลาที่กำหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- STATUS เป็นขาชนิดเอาท์พุทใช้ตรวจดูสภาวะการทำงานของ Rabbit 2000

- SMODE0 , SMODE1 เป็นขาชนิดอินพุทใช้กำหนดเริ่มการทำงาน

- CS0,OE0 , WEO เป็นขาชนิดเอาท์พุทตามปกติใช้เป็นการ Enable เพื่อให้ทำการเขียนข้อมูลลง Flash Memory ใน Program Code

- CS1 , OE1 , WE1 เป็นขาชนิดเอาท์พุทตามปกติใช้เป็นการ Enable เพื่อให้ทำการเขียนข้อมูลลง Static RAM และทำให้ความเร็วและการสูญเสียพลังงานเหมาะสมกัน

- CS2 เป็นขาชนิดเอาท์พุท จุดประสงค์ทั่วไปใช้ในการ Memory Decode

- BUFFER เป็นขาชนิดเอาท์พุทปกติตั้งไว้เป็น Enable ทำงานกับอินพุท/เอาท์พุทภายนอก จะทำงานเมื่อไมโครโปรเซสเซอร์ Rabbit 2000 ส่งสัญญาณร้องขอ

- IORD , IOWR เป็นขาชนิดเอาท์พุทใช้ระบุการอ่านและเขียนข้อมูลกับส่วนอินพุท/เอาท์พุท

- PA0 - PA7 เป็นขาชนิดอินพุท/เอาท์พุทของพอร์ต A (Slave Port) ทำหน้าที่รีเซ็ตพอร์ต

- PB0 - PB7 เป็นขาชนิดอินพุท/เอาท์พุทของพอร์ต B (Slave Port) ทำหน้าที่ควบคุม Slave Port, Serial Port Clocks

- PC0-PC7 เป็นขาชนิดอินพุท/เอาท์พุทของพอร์ต C (Slave Port) โดยทั่วไปทำหน้าที่เป็น Serial Data ของ Slave Port

- PD0 - PD7 เป็นขาชนิดอินพุท/เอาท์พุทของพอร์ต D (Slave Port)

- PE0-PE7 เป็นขาชนิดอินพุท/เอาท์พุทของพอร์ต E (Slave Port) โดยทั่วไปทำหน้าที่เป็นอินเตอรัปต์จากภายนอก

- VBAT ใช้จ่ายพลังงานชนิด Real - Time Clock (25 ไมโครแอมป์ ที่ 2.3 โวลต์)

- VDD ใช้จ่ายพลังงานให้กับไมโครโปรเซสเซอร์ Rabbit 2000 ที่แรงดัน 2.7 โวลต์ - 5.0โวลต์

- VSS คือ กราวด์ (Ground)

2.5 พอร์ตในการเชื่อมต่อ Rabbit Core Module RCM 2200

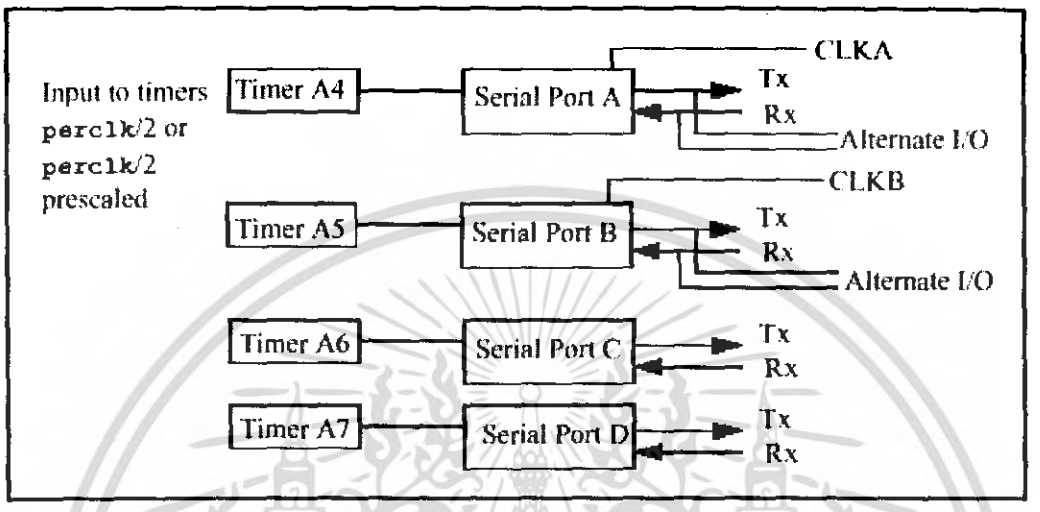
พอร์ตในการเชื่อมต่อ Rabbit Core Module RCM 2200 บอร์ดไม่มีสาย RS – 232 หรือสาย RS - 485 โดยตรงบนบอร์ด การจะติดต่อด้วยสาย RS - 232 หรือ RS - 485 สามารถทำได้โดยการติดต่อผ่านบอร์ด Prototyping ซึ่งมีการสนับสนุนมาตรฐาน RS - 232 ไว้ในบอร์ดเดียวกัน

2.5.1 พอร์ตอนุกรม

พอร์ตอนุกรมมีอยู่ 4 พอร์ตคือ พอร์ต A , B , C และ D ทั้งหมด 4 พอร์ตถูกใช้สำหรับการพัฒนาซอฟท์แวร์สามารถกระทำในโหมด Asynchronous ที่มีอัตราบอร์ดเรตสูง ในโหมด Asynchronous สามารถจัดการ 7 หรือ 8 บิตข้อมูล เมื่อมีการส่งข้อมูลข่าวสาร ไบต์ (byte) แรกของข้อมูลจะถูกกำหนดเพื่อให้ทราบถึงการรับส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พอร์ต A , B สามารถทำงานในโหมด Asynchronous ทำงานไปพร้อมกับพอร์ต C และ D ได้ อย่างไรก็ตามพอร์ต A และ B ยังสามารถที่จะกำหนดการทำงานให้เป็นอย่างใดอย่างหนึ่ง ระหว่างการรับหรือส่ง อุปกรณ์ในการติดต่อจะมีความสัมพันธ์กับสัญญาณนาฬิกา เมื่อ Rabbit 2000 สร้างสัญญาณนาฬิกาอัตราบอร์ดเรตสูง โดยคิดจากความถี่ของสัญญาณนาฬิกาหารด้วยค่า 128 , 138 หรือ 240 บิต ต่อวินาทีสำหรับ 22.1 เมกะเฮริด์



รูปที่ 2.6 บล็อกไดอะแกรมของพอร์ตอนุกรม

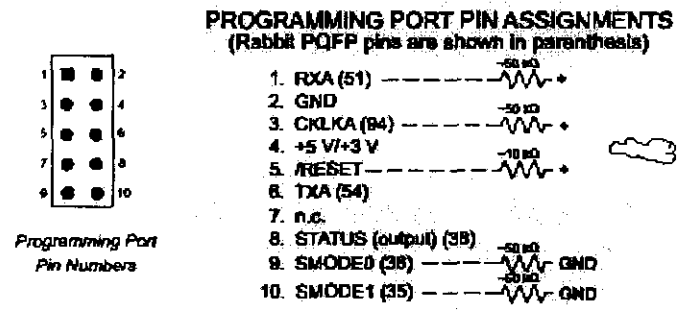
2.5.2 อีเทอร์เน็ตพอร์ต

อีเทอร์เน็ตพอร์ตที่ใช้บนบอร์ด Rabbit Core Module RCM 2200 จะใช้หัวต่อแบบ RJ – 45 ซึ่งจะมี LED อยู่สองดวงมีหน้าที่แสดงสถานะการทำงานของบอร์ด Rabbit Core Module RCM 2200 โดยขาสัญญาณ (LNK) มีหน้าที่บอกสถานะการเชื่อมต่อ ส่วนขาสัญญาณ (ACT) จะบอกสถานะการถ่ายโอนข้อมูล โดยอีเทอร์เน็ตพอร์ตจะอยู่ที่คอนเน็คเตอร์ J2

2.5.3 พอร์ตการเขียนโปรแกรม

พอร์ตการเขียนโปรแกรมสื่อสารกับ Rabbit Core Module RCM 2200 โดยทาง Chip serial-port พอร์ต A มีความสามารถพิเศษที่ใช้สำหรับการพัฒนาซอฟท์แวร์ภายใต้โปรแกรม Dynamic C ขาสัญญาณ (SMODE0,SMODE1) แสดงการเขียนโปรแกรมเชื่อมต่อกับอุปกรณ์ภายนอกตำแหน่งที่ใช้ในการเขียนโปรแกรมสำหรับ Rabbit Core Module RCM 2200 ในโหมดที่จะดาวน์โหลดโปรแกรมจากคอมพิวเตอร์ให้แก่ Rabbit Core Module RCM 2200 จนจบโปรแกรม พร้อมทั้งทำการตรวจสอบจุดบกพร่องในโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 พอร์ตการเขียนโปรแกรม

2.6 พอร์ตขนาน

Rabbit 2000 มีพอร์ตขนานทั้งหมด 5 พอร์ตโดยแต่ละพอร์ตจะมีพอร์ตละ 8 บิต ซึ่งพอร์ตขนานจะแบ่งออกเป็น A , B , C , D , E โดยขาของสัญญาณที่ใช้สำหรับพอร์ตขนานจะทำการแชร์กับฟังก์ชันอื่นๆมากมาย คุณสมบัติสำคัญของพอร์ตขนานสรุปได้ดังนี้

พอร์ต A - จะทำการแชร์กับส่วนที่ติดต่อข้อมูลของพอร์ต Slave

พอร์ต B - จะทำการแชร์กับเส้นทางของหน่วยควบคุมของพอร์ต Slave

พอร์ต C - จะทำการแชร์กับพอร์ตที่เป็นชุดข้อมูลของชุดอินพุท/เอาท์พุท

พอร์ต D - จะมีบิตอยู่ 4 บิตที่จะทำการแชร์และสลับกันของขาสัญญาณอินพุท/เอาท์พุท สำหรับพอร์ตอนุกรมของ A และ B ส่วนอีก 4 บิตจะไม่มีการแชร์ โดยพอร์ต D มีความสามารถที่จะแก้ไขเอาท์พุทของตัวเองตามเอาท์พุททั้งหมด

พอร์ต E - บิตทั้งหมดของพอร์ต E สามารถที่จะแก้ไขเช่นเดียวกับ I/O strobes โดยบิต 4 บิตของพอร์ตสามารถนำไปใช้ในการอินเตอร์รัปต์ภายในของ I/P ส่วนอีก 1 บิตจะทำการแชร์ใน พอร์ต Slave

2.6.1 พอร์ตขนาน A

พอร์ตขนาน A จะมีหน้าที่การอ่าน / เขียนรีจิสเตอร์ได้อย่างเดียวโดยจะมีข้อมูล PADR (adr = 030h)

รีจิสเตอร์นี้ไม่ควรใช้ถ้าพอร์ต Slave มีการใช้งาน รีจิสเตอร์ของหน่วยควบคุมพอร์ด Slave ถูกใช้ให้หน่วยควบคุมพอร์ตขนาน A ได้ทั้งอินพุทและเอาท์พุท เพื่อทำให้พอร์ต A เป็นอินพุทต้องให้เก็บค่า 080h ใน SPCR (slave port control register) ส่วนการทำให้พอร์ต A เป็นเอาท์พุทต้องให้เก็บค่า 084h ใน SPCR พอร์ต A จะทำการเซ็ตค่าอินพุทโดยการรีเซ็ต

2.6.2 พอร์ตขนาน B

พอร์ตขนาน B จะมีอินพุทอยู่ 6 ตัวและเอาท์พุทอยู่ 2 ตัว ซึ่งจะมีขาของสัญญาณ PB2 - PB7 เป็นตัวกำหนดฟังก์ชันของพอร์ต Slave อย่างไรก็ตามก็ยังสามารถที่จะอ่านโดยอาศัยขาสัญญาณ PB0 - PB5 ถึงแม้ว่าขาสัญญาณ PB2 - PB7 จะมีการนำไปใช้งานแล้วก็ตามและยังสามารถที่จะอ่านโดยอาศัยขาสัญญาณ PB6 และ PB7 ได้อีกด้วย โดยไม่คำนึงถึงว่าพอร์ต Slave จะถูกใช้หรือไม่ก็ตาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกจากพอร์ตอนุกรม B มีสัญญาณนาฬิกาภายในซึ่งก็เป็นสาเหตุที่ทำให้ขาของสัญญาณนี้ถูกใช้งานโดยสัญญาณนาฬิกาของพอร์ตอนุกรม ส่วนกรณีของขาสัญญาณ PB1 จะมีการคืนค่าหากมีการนำข้อมูลเข้ามานอกจากถ้าไม่ใช่พอร์ตอนุกรม A

2.6.3 พอร์ตขนาน C

พอร์ตขนาน C จะมีอินพุทอยู่ 4 บิตและเอาท์พุทอยู่ 4 บิต โดยจะแบ่งออกเป็นบิตที่เป็นเลขคู่จะประกอบไปด้วย PC0 , PC2 , PC4 และ PC6 จะเป็นเอาท์พุท และบิตที่เป็นเลขคี่จะประกอบไปด้วย PC1 , PC3 , PC5 และ PC7 จะเป็นอินพุท เมื่อรีจิสเตอร์ของข้อมูลทำการอ่าน บิตที่ 1 , 3 , 5 , 7 จะทำการคืนค่าของแรงดันไฟฟ้าบนขาของสัญญาณ ส่วนบิตที่ 0 , 2 , 4 , 6 จะทำการคืนค่าโดยการส่งสัญญาณบัฟเฟอร์ไปที่เอาท์พุทโดยในการส่งสัญญาณบัฟเฟอร์ผลลัพธ์ที่ได้และค่าของขาสัญญาณจะเหมือนกัน

พอร์ตขนาน C จะมีการแชร์กันของขาสัญญาณของพอร์ตอนุกรมทั้งหมด 4 บิต โดยขาสัญญาณของพอร์ตขนาน C สามารถเป็นอินพุทและยังเป็นขาสัญญาณของพอร์ตอนุกรมที่สามารถเป็นอินพุทได้ด้วย (พอร์ตอนุกรม A และ B สามารถใช้สลับบิต 7 และ 5 ตามลำดับในพอร์ต C ที่เป็นอินพุทและ Source ของอินพุทที่พอร์ตอนุกรมจะขึ้นอยู่กับหน่วยควบคุมพอร์ตโดยตรงของรีจิสเตอร์ควบคุมของพอร์ตอนุกรม) เส้นทางของข้อมูลสามารถอ่านได้จากพอร์ตขนาน C โดยรีจิสเตอร์ของข้อมูลเอาท์พุทของพอร์ตขนานสามารถถูกนำไปเลือกใช้พอร์ตอนุกรม โดยจะทำการเก็บในตำแหน่งที่ตรงกันของพอร์ต C ในรีจิสเตอร์ฟังก์ชัน (PCFR) เมื่อขาสัญญาณของพอร์ตขนานถูกเลือกใช้โดยผลลัพธ์ของพอร์ตอนุกรมค่าที่เก็บในรีจิสเตอร์ของข้อมูลจะไม่ถูกสนใจ

2.6.4 พอร์ตขนาน D

พอร์ตขนาน D มีขาสัญญาณทั้งหมด 8 บิต ซึ่งสามารถที่จะโปรแกรมให้เป็นอินพุทหรือเอาท์พุทได้ เมื่อโปรแกรมเป็นเอาท์พุทขาสัญญาณก็สามารถที่จะเลือกเอาท์พุทหรืออินพุทที่เป็นมาตรฐานขาสัญญาณของพอร์ต D สามารถเป็นบิตที่เป็นตำแหน่งแอดเดรส รีจิสเตอร์ของเอาท์พุทจะมีรูปแบบและมีไทม์เมอร์ ควบคุมตามลำดับขั้นตอนการทำงานจะมีการสร้างสัญญาณพัลส์ในการจับเวลา นอกจากนี้เอาท์พุทของพอร์ต D มีความสามารถในการส่งสัญญาณที่สูง พอร์ต D จะมีบิต 4 และ 5 ที่สามารถที่จะทำการสลับขาสัญญาณกันกับพอร์ตอนุกรม B และบิต 6 และ 7 ยังสามารถถูกใช้ในการสลับขาสัญญาณกับพอร์ตอนุกรม A การสลับกันของบิตพอร์ตอนุกรมจะกระทำในพอร์ตอนุกรม โดยจะมีการเชื่อมต่อกับขาสัญญาณในการติดต่อที่แตกต่างกันในเวลาเดียวกัน ในกรณีการรีเซ็ต ข้อมูลในรีจิสเตอร์โดยตรงจะมีค่าเท่ากับ " 0 " การทำงานของขาสัญญาณทั้งหมดจะเป็นอินพุท นอกจากนี้ในการส่งค่าให้กับรีจิสเตอร์หน่วยควบคุมมีค่าเท่ากับ " 0 " (บิต 0 , 1 , 4 , 5) ก็เพื่อที่จะให้แน่ใจว่าข้อมูลได้ถูกใช้สัญญาณนาฬิกาเข้าไปในรีจิสเตอร์เมื่อมีการบันทึกรีจิสเตอร์อื่นๆ ทั้งหมดที่เกี่ยวข้องกับพอร์ต D จะไม่ทำงานตอนเริ่มต้นเมื่อเราทำการรีเซ็ต

PDDR - รีจิสเตอร์ข้อมูลของพอร์ตขนาน D จะทำการอ่าน/เขียน

PDDDR - รีจิสเตอร์โดยตรงข้อมูลของพอร์ตขนาน D มีค่าเท่ากับ"1" ทำให้ขาสัญญาณเป็นเอาท์พุทใช้ในการเขียนเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PDDCR - รีจิสเตอร์ของหน่วยควบคุมของพอร์ตขนาน D มีค่าเท่ากับ "1" ทำให้ขาสัญญาณเป็นการเปิดสัญญาณเอาท์พุทใช้ในการเขียนเท่านั้น

PDFR - รีจิสเตอร์หน่วยควบคุมฟังก์ชันของพอร์ตขนาน D พอร์ตนี้อาจจะถูกใช้เพื่อหาตำแหน่งของพอร์ต 4 และ 6 ของเอาท์พุทอนุกรม ใช้ในการเขียนเท่านั้น

PDBxR - จะมีรีจิสเตอร์ทั้งหมด 8 ตัว โดยจะถูกใช้เพื่อเซ็ตตำแหน่งของพอร์ตที่จะใช้งาน

PDCR - รีจิสเตอร์หน่วยควบคุมของพอร์ตขนาน D รีจิสเตอร์นี้ถูกใช้ในการควบคุมสัญญาณนาฬิกาของด้าน High และ Low ของเอาท์พุทตัวสุดท้ายของ รีจิสเตอร์ของพอร์ต ส่วนการรีเซ็ตบิต 0 , 1, 4 และ 5 จะเท่ากับ 0

2.6.5 พอร์ตขนาน E

มีขาสัญญาณอินพุท/เอาท์พุททั้งหมด 8 บิตที่สามารถถูกโปรแกรมเฉพาะอย่าง ไม่ว่าจะเป็นอินพุทและเอาท์พุท พอร์ตขนาน E มีความสามารถในการส่งสัญญาณที่สูงกว่าพอร์ตอื่นๆ PE7 จะถูกใช้เมื่อพอร์ต Slave ถูกใช้งานเป็นเอาท์พุทของพอร์ตขนาน E สามารถที่จะแก้ไข I/O strobe นอกจากนี้ยังมีขาสัญญาณจำนวน 4 บิตของพอร์ตขนาน E สามารถเกิดการอินเตอร์รัปต์เมื่อมีการร้องขอการอินเตอร์รัปต์ที่อินพุทรีจิสเตอร์ของเอาท์พุทมีรูปแบบและไทม์เมอร์อย่างมีแบบแผนการทำงานเป็นไปตามการสร้างสัญญาณของเวลา

PEDR - รีจิสเตอร์ข้อมูลของพอร์ตขนาน E มีหน้าที่อ่านค่าจากขาสัญญาณ

PEDDR - รีจิสเตอร์โดยตรงข้อมูลของพอร์ตขนาน E หากเซ็ตเป็น "1" จะทำให้ขาสัญญาณเป็นเอาท์พุท รีจิสเตอร์นี้จะรีเซ็ตก็ต่อเมื่อมีค่าเท่ากับ 0

PEFR - รีจิสเตอร์ฟังก์ชันของพอร์ตขนาน E หากเซ็ตเป็น "1" จะทำให้เอาท์พุทตรงกับ I/O strobe โดยปกติ I/O strobe จะถูกควบคุมโดยรีจิสเตอร์ I/O ของหน่วยควบคุม (IBxCR) ต้องมีการเซ็ตเป็นเอาท์พุทสำหรับ I/O strobe ถึงจะทำงาน

PEBxR - รีจิสเตอร์นี้จะเป็นส่วนของรีจิสเตอร์ที่ใช้ในการเซ็ตให้เป็นเอาท์พุทของแต่ละบิต

PECR - รีจิสเตอร์หน่วยควบคุมของพอร์ตขนาน E เป็นรีจิสเตอร์ที่ถูกใช้เพื่อควบคุมสัญญาณนาฬิกาของด้าน High และ Low ของเอาท์พุทตัวสุดท้ายของรีจิสเตอร์ของพอร์ตในการรีเซ็ตบิต 0 , 1 , 4 และ 5 จะมีค่าเท่ากับ 0

2.7 โปรแกรม Dynamic C

Dynamic C คือระบบการพัฒนารวมของการเขียนซอฟท์แวร์สร้างมาจากระบบคอมพิวเตอร์ของ IBM ซึ่งออกแบบให้เข้ากับคอมพิวเตอร์ทั่วไปให้ใช้งานได้

Dynamic C มีการใช้ในเครือข่ายตั้งแต่ปี ค.ศ. 1989 ถูกออกแบบสำหรับการเขียนโปรแกรมฝังระบบและมีความสามารถตรวจสอบในตัวของมันเองอย่างรวดเร็ว สำหรับไมโครโปรเซสเซอร์ Rabbit 2000 ที่ใช้งานกันทั่วไปสามารถติดต่อกันโดย Down load ที่พอร์ต B โปรแกรมพื้นฐานของระบบมีข้อมูลประมาณ 1,000 ไบต์ ที่ใช้ในการจัดเตรียม Debugging และการติดต่อข้อมูลต่าง ๆ Dynamic C ต้องการ BIOS เพื่อใช้ในการตรวจสอบโปรแกรม ถ้าผู้ใช้หยุดการ Run โปรแกรมและใช้โปรแกรมใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

BIOS ก็จะตรวจสอบการทำงานใหม่ตลอด Dynamic C ออกแบบให้เข้ากับภาษาแอสเซมบลี้ หรือใช้ได้กับโปรแกรมภาษาซี

2.7.1 Using Dynamic C

ผู้ใช้โปรแกรมมีตัวเลือกในการที่จะพัฒนาซอฟท์แวร์ภาษาเขียนใน Flash Memory ขนาด 256 กิโลไบต์ (Kbyte) หรือใน Static Ram ขนาด 128 กิโลไบต์ ผลการทำงานในหน่วยความจำ คือการบันทึกข้อมูลสามารถบันทึกได้ถึง 100,000 ครั้งของการเขียน

ข้อเสียของการใช้ Flash Memory เมื่อมีการดีบัก โปรแกรมเพื่อขจัดจังหวะการทำงานจะทำให้อินเตอร์รัปต์เกิดข้อผิดพลาด การทำงานของโปรแกรมก็จะหยุดตามไปด้วย

Dynamic C เป็นภาษาที่ใช้ในการสนับสนุน TCP/IP โดย Dynamic C จะประกอบไปด้วย Libraries ต่างๆ โดยจะมี Libraries หลัก คือ DCRTCP.LIB อีกทั้ง Dynamic C ยังมี Libraries ส่วนของ DNS (Domain Name Server), IP, TCP, and UDP (User Datagram Protocol) คือ DNS.LIB, IP.LIB, NET.LIB, TCP.LIB and UDP.LIB ส่วนในการติดต่อหรือในส่วนของชั้นเครือข่ายของโปรโตคอล TCP/IP จะมี Libraries ที่ชื่อ ARP.LIB และ ICMP.LIB

ในส่วนของ Libraries หลัก DCRTCP.LIB จะประกอบไปด้วย macros ที่ทำการตั้งค่าไว้ส่วนโครงสร้างข้อมูลและฟังก์ชันที่ใช้กับ IP เวอร์ชัน 4 ได้มีการสนับสนุนโดย DCRTCP.LIB ในการคอมไพล์ TCP/IP จะต้องให้ส่วนของบอร์ดควบคุมรู้ค่า IP address, netmask and default gateway

2.7.2 การเซ็ตค่า IP Addresses

ค่า IP Address มีความจำเป็นที่จะต้องทำการเซ็ตค่าในช่วงของการคอมไพล์ โดยจะกำหนดการตั้งค่าไว้ที่ MY_IP_ADDRESS, MY_NETMASK, MY_GATEWAY และ MY_NAMESERVER ตามลำดับในช่วงของการคอมไพล์ในส่วนของฟังก์ชัน tcp_config sethostid sethostname โดยสามารถที่จะควบคุมโดย Macros

2.7.3 IP Addresses Set Dynamically

library BOOTP.LIB จะยอมให้บอร์ดในส่วนของ BOOTP หรือ DHCP ของ Client ให้เป็นส่วนที่จะนำไปใช้โดยโปรโตคอลจะยึดหลักของขนาดของเครื่องเซิร์ฟเวอร์ (Server) ที่ทำการติดตั้งบนเครือข่ายภายในเครื่องเซิร์ฟเวอร์ BOOTP และ DHCP จะติดตั้งในส่วนกลางของระบบเครือข่ายภายในและจะคอยจัดการในการวางแผนงานของระบบเครือข่าย

โปรโตคอลทั้ง 2 นี้ จะมีค่าพารามิเตอร์ที่ใช้ในการส่งไปยัง Client รวมถึง

- IP address ของ Client
- Net mask
- รายชื่อ Gateway
- Host และ รายชื่อโดเมนพื้นฐาน
- รายชื่อของเครื่องเซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการนำไปใช้งานต้องโปรแกรมดังนี้

#define USE_DHCP

#use DCRTCP.LIB

2.7.4 BOOTP/DHCP Control Macros

macros ต่างๆ สามารถที่จะทำการควบคุมการ DHCP หากมีการเซ็ตค่าก่อนบรรทัด #use "dcrtcp.lib" ในส่วนโปรแกรมแอพพลิเคชั่น (Application) USE_DHCP ถ้า macros นี้ถูกกำหนดจุดมุ่งหมายในการใช้ BOOTP หรือ DHCP เพื่อแก้ไขตัวแปรที่ต้องการ ถ้า USE_DHCP ไม่ถูกกำหนดจะทำให้ MY_IP_ADDRESS, MY_NETMASK, MY_GATEWAY และ MY_NAMESERVER อาจจะถูกกำหนดโดยในส่วนของโปรแกรมแอพพลิเคชั่น

2.7.5 Sizes for TCP/IP I/O Buffers

ในการเริ่มทำงานของ Dynamic C เวอร์ชัน 8.01 บัฟเฟอร์ TCP และ UDP I/O C ได้มีการแบ่งแยกออกมาเป็น

TCP_BUF_SIZE ได้มีการกำหนดขนาดบัฟเฟอร์ของ TCP ไว้ที่ 4096 ไบต์

UDP_BUF_SIZE ได้มีการกำหนดขนาดบัฟเฟอร์ของ UDP ไว้ที่ 4096 ไบต์

ถ้า SOCK_BUF_SIZE มีการกำหนดจะทำให้ค่า TCP_BUF_SIZE และ UDP_BUF_SIZE จะตรงกับ SOCK_BUF_SIZE แต่ถ้า SOCK_BUF_SIZE ไม่มีการกำหนด จะทำให้ค่า TCP_BUF_SIZE และ UDP_BUF_SIZE จะเท่ากับ tcp_MaxBufSize * 2

2.7.6 หมายเลขซ็อกเกต (Number of Sockets)

การเริ่มต้นการทำงาน Dynamic C เวอร์ชัน 8.01 จะมีการกำหนด macros 2 ตัวให้กับหมายเลขซ็อกเกตได้ดังนี้

MAX_TCP_SOCKET_BUFFERS จะมีการกำหนดหมายเลขสูงสุดให้กับซ็อกเกตของ TCP โดยจะมีการเรียก tcp_open() หรือ tcp_listen()

MAX_UDP_SOCKET_BUFFERS จะมีการกำหนดหมายเลขสูงสุดให้กับซ็อกเกตของ UDP udp_open()

2.7.7 Passive Open

จะมีอยู่ 2 เส้นทางในการเปิดซ็อกเกตของ TCP คือ passive และ active ซึ่งการเปิดซ็อกเกตนี้จะต้องทำการเรียก tcp_listen(); โดยจะทำการรอการติดต่อกับอุปกรณ์และชนิดของการเปิดจะใช้กับเซิร์ฟเวอร์ของอินเตอร์เน็ทหรืออาจจะให้ tcp_listen() เป็นตัวชี้ตำแหน่งของข้อมูล ของ tcp_Socket ถ้าคุณต้องการที่จะติดต่อโดยคุณได้ยอมรับข้อตกลงของหมายเลขพอร์ตและ IP Address โดยทำการเซ็ตค่าให้เป็นศูนย์หรือหนึ่งทั้งคู่

2.7.8 Active Open

เมื่อมีการเรียกหน้าเว็บบราวด์เซอร์ก็จะทำการเชื่อมต่อกับเซิร์ฟเวอร์เพื่อทำการเปิด โดยจะทำการเรียกใช้ tcp_open() และยังใช้พารามิเตอร์คล้ายๆ กับการใช้ tcp_listen() โดยหลักพารามิเตอร์นี้จะเกี่ยวกับ IP Address และหมายเลขพอร์ตคุณสามารถทำการติดต่อ โดยทำการเซ็ตค่าพารามิเตอร์ l พอร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ให้เป็นศูนย์ส่วน DCRTCP.LIB เป็นการแสดงถึงการเลือกพอร์ตระหว่าง 1024 และ 65535 หาก tcp_open() กลับค่ามาเป็นศูนย์อีกครั้งจะไม่สามารถทำการติดต่อได้

2.7.9 Delay a Connection

การยอมรับการร้องขอการติดต่อเมื่อกระบวนการหรือวิธีการร้องขอไม่เหมาะสมก็จะทำการเรียกฟังก์ชัน tcp_reserveport() เมื่อมีการตอบรับการติดต่อพารามิเตอร์ในส่วนหัวของ TCP จะทำการเซ็ตค่าเป็นศูนย์ในช่วงเวลานี้จะทำการรอค่าจากพารามิเตอร์ tcp_clearreserve (port number) เพื่อทำการดูแลในการติดต่อนอกจากนี้ยังมี macros USE_RESERVEDPORTS จะทำการกำหนดเพื่อให้เกิดการทำงานขึ้นของสองฟังก์ชันคือเมื่อมีการใช้ tcp_reserveport, 2MSL (Maximum Segment Lifetime)

2.7.10 Skeleton Program

โปรแกรมข้างล่างนี้ เป็นโครงสร้างพื้นฐานของ Dynamic C TCP/IP โดยส่วนแรกจะทำการกำหนดการเซ็ตค่า IP ของข้อมูลในบรรทัด "memmap" จะเป็นการให้โปรแกรมทำการคอมไพล์และตรวจสอบค่าส่วนในบรรทัด "use" จะเป็นตัวสั่งให้ตัวคอมไพล์ทำการคอมไพล์ของข้อมูลที่มีการตั้งค่าไว้ในส่วน TCP/IP ของ Dynamic C

```
#define MY_IP_ADDRESS "10.10.6.101"
#define MY_NETMASK "255.255.255.0"
#define MY_GATEWAY "10.10.6.19"
#memmap xmem
#use dcrtcp.lib
main() {
      sock_init();
      for (;;) {
          tcp_tick(NULL);
      }
}
```

2.7.11 TCP/IP Stack Initialization

ใน TCP/IP จะมีฟังก์ชัน Main () เป็นส่วนเริ่มแรกและจะมีการเรียก sock_init() เพื่อใช้กับโครงสร้างข้อมูลภายในและชิป Ethernet หรือชิป RealTek ส่วน DCRTCP.LIB จะคอยจัดการกับแพ็คเก็ตที่เข้ามา

2.7.12 Packet Processing

เมื่อไรก็ตามเมื่อแพ็คเก็ตมีการนำเข้ามาจะมีการเรียก tcp_tick() , tcp_open , udp_open, sock_read , sock_write , sock_close และ sock_abort ซึ่งเป็นวิธีที่ดีที่ tcp_tick() จะรู้ระยะเวลาในการประมวลผลโปรแกรมที่ส่งมาในรูปของแพ็คเก็ต

2.7.13 Function Reference

ในส่วนนี้จะกล่าวถึงฟังก์ชันที่ใช้ใน DCRTCP.LIB ในเริ่มต้นนั้น DCRTCP.LIB ของ Dynamic C 7.05 จะมีฟังก์ชันที่มีการใช้อยู่ภายนอก DNS.LIB , IP.LIB , NET.LIB , TCP.LIB และ UDP.LIB และยังมีฟังก์ชันอีกส่วนหนึ่งที่สามารถเลือกใช้ได้ เช่น ARP.LIB, ICMP.LIB, BSDNAME.LIB และ XMEM.LIB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.14 Macros

- MAX_SOCKETS macros จะเป็นตัวกำหนดหมายเลขซ็อกเกตเพื่อทำการจัดสรรพื้นที่
- MY_GATEWAY macros จะทำการส่งค่านี้เพื่อไปควบคุม Gateway ในช่วงเวลา Runtime
- MY_NETMASK macros เป็นพื้นฐานของ NETMASK ที่ใช้ในการควบคุม
- MY_NAMESERVER macros เป็นการแสดงชื่อของเซิร์ฟเวอร์ในช่วงรันโปรแกรม
- MY_DOMAIN macros จะทำการควบคุมตำแหน่งเริ่มต้นของโดเมนในช่วงเวลา Runtime
- MY_IP_ADDRESS macros จะทำการระบุตำแหน่ง IP Address เพื่อใช้ในการควบคุมในช่วงเวลา Runtime

2.8 HTTP Server

HTTP (Hypertext Transfer Protocol) เซิร์ฟเวอร์จะทำการสร้าง HTML (Hypertext Markup Language) เป็นส่วนที่ทำให้ Client ได้มีการเรียกใช้งานของเอกสารได้ง่ายขึ้น

ServerSpec Type Field

SSPEC_FILE	ข้อมูลของไฟล์ที่ใช้
SSPEC_VARIABLE	ข้อมูลชนิด variable ที่ใช้ใน HTTP
SSPEC_FUNCTION	ข้อมูลของฟังก์ชันที่เรียกใช้งาน

โครงสร้างข้อมูลเครื่องเซิร์ฟเวอร์ HTTP

โครงสร้างข้อมูลใน HTTP.LIB ที่น่าสนใจมีอยู่ 4 ส่วนเพื่อผู้พัฒนาเครื่องเซิร์ฟเวอร์ HTTP นำไปพัฒนาต่อคือ

- HttpSpec

โครงสร้างข้อมูล HttpSpec จะบรรจุแฟ้มของทั้งหมด ,ตัวแปร, และฟังก์ชันการทำงานของเครื่องเว็บเซิร์ฟเวอร์ที่ใช้ในการเข้าถึงข้อมูลโครงสร้าง ServerSpec มาจาก ZSERVER.LIB

type

ส่วนพื้นที่นี้จะเป็นตัวบอกเครื่องเซิร์ฟเวอร์ว่ามีไฟล์ข้อมูล,ตัวแปรหรือฟังก์ชันเข้ามา (HTTPSPEC_FILE, HTTPSPEC_VARIABLE หรือ HTTPSPEC_FUNCTION ตามลำดับ)

name

พื้นที่นี้จะทำการระบุชื่อเฉพาะที่ใช้ในการอ้างอิงส่วนที่มีการนำข้อมูลเข้ามา

data

เป็นส่วนของตำแหน่งทางกายภาพของข้อมูล

addr

ส่วนนี้จะทำหน้าที่เป็นตัวชี้ข้อมูลไฟล์ของข้อมูลทั้งหมดของตำแหน่งทางกายภาพ ซึ่งตัวแปรและฟังก์ชันจะต้องใช้ตัวชี้เป็นตัวชี้ตำแหน่งของข้อมูล

72755

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

vartype

ส่วนนี้จะเป็นตัวบอกชนิดของตัวแปร เช่น INT8 , INT16 , PTR16 , INT32 , FLOAT

format

ส่วนนี้จะเป็นการอ้างอิงถึงรูปแบบ printf เพื่อประกาศตัวแปร

realm

เป็นส่วนของชื่อและรหัสผ่านที่ใช้ในการเข้าถึงข้อมูล

- HttpType

โครงสร้าง HttpType จะเกี่ยวข้องกับการขยายไฟล์ข้อมูลร่วมกับ MIME (Multipurpose Internet Mail Extension) และฟังก์ชันที่เกี่ยวข้องกับ MIME ผู้ใช้ยังสามารถควบคุม HTTP_MAXNAME

- HttpRealm

โครงสร้าง HttpRealm จะมีการใช้งานที่ควบคู่กันคือ ID ของผู้ใช้และรหัสผ่านสำหรับพื้นที่การเรียก Realm โดย Realm จะมีการป้องกันข้อมูลในเครื่องเซิร์ฟเวอร์อีกทั้งยังมีการแบ่งพื้นที่เพื่อทำการเซ็ตส่วนของพื้นที่ที่ใช้ในการป้องกันข้อมูล

ในโครงสร้าง HttpSpec จะมีตัวชี้โดยตัวชี้นี้จะมีโครงสร้างชนิด HttpRealm เพื่อเป็นรหัสผ่านเพิ่มชื่อ การเพิ่มรหัสผ่านถ้าหากไม่ต้องการรหัสผ่านหรือการป้องกันก็ไม่ต้องใช้

- HttpState

การใช้สำหรับฟังก์ชัน CGI

2.9 Configuration Macros

macros ใน HTTP.LIB มีดังต่อไปนี้

HTTP_MAXNAME

เป็นขนาดความยาวที่สุดของชื่อในโครงสร้างของ HttpSpec จะมีตัวอักษรได้ 20 ตัวอักษร โดยความยาวสูงสุดของชื่อใดๆนั้นจะมีได้ 19 ตัวอักษรเพราะว่าหนึ่งตัวอักษรได้มีการถูกใช้สำหรับเป็นส่วนท้าย

HTTP_MAXRAMSPEC

เป็นหมายเลขสูงสุดของส่วน HttpSpec ซึ่งสามารถทำการเพิ่มการ runtime macros นี้ควบคุมโดย SSPEC_MAXSPEC

HTTP_MAXSERVERS

เป็นจำนวนสูงสุดของ HTTP server บนพอร์ตหมายเลข 80 แต่ไม่เกิน 4 เซิร์ฟเวอร์

HTTP_PORT

Macros นี้ยอมให้ผู้ใช้สามารถควบคุมพอร์ตโดยการกำหนดจะต้องกำหนดก่อนบรรทัด

#use http.lib

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **Functions - cgi_redirectto**

รูปแบบ

void cgi_redirectto(HttpState* state, char* url);

คำอธิบาย

เป็นส่วนที่ทำการเรียกฟังก์ชัน CGI

พารามิเตอร์

state = จะทำเป็นตัวเครื่องเซิร์ฟเวอร์หรือเป็นตัวรับฟังก์ชัน CGI

url = จะเป็นตัวแสดงคุณสมบัติของ url

Library

HTTP.LIB

- **http_handler**

รูปแบบ

int http_handler();

คำอธิบาย

เป็นเป็นฟังก์ชันพื้นฐานที่ใช้ขยายความของฟังก์ชันต่างๆ

Library

HTTP.LIB

- **http_init**

รูปแบบ

int http_init(void);

คำอธิบาย

เริ่มต้นการทำงานของ HTTP

Library

HTTP.LIB

2.10 ทฤษฎีเกี่ยวกับเครือข่ายเน็ตเวิร์ค

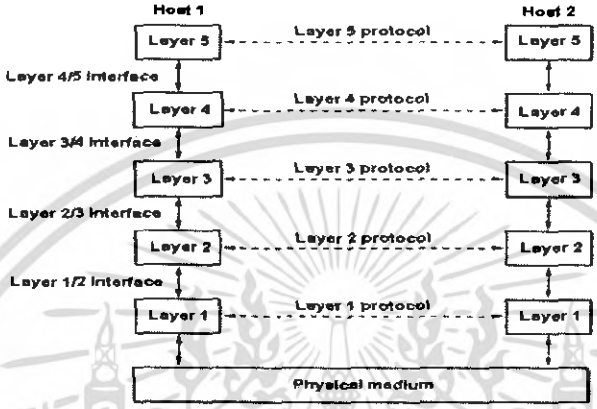
TCP/IP เป็นข้อกำหนดเกี่ยวกับรูปแบบการเชื่อมโยงในเครือข่าย (Network Protocol) จัดทำขึ้นเพื่อใช้เป็นเครื่องคอมพิวเตอร์ใช้งานร่วมกัน ในลักษณะของระบบเปิด (open system) คือ ไม่ว่าจะเป็นคอมพิวเตอร์ชนิดใดหรือระบบใดก็ตามจะสามารถติดต่อสื่อสารและแลกเปลี่ยนข้อมูลกันได้

2.10.1 ลำดับชั้นของกฎการสื่อสารข้อมูล

ระบบโปรแกรมเครือข่ายส่วนมากจะแบ่งแยกการทำงานออกเป็นหลายระดับ (Levels) หรือหลายชั้น (Layers) แต่ละชั้นจะสร้างฟังก์ชันการทำงานขึ้นโดยอาศัยการทำงานของเครือข่ายต่างๆ จะแตกต่างกันออกไป อย่างไรก็ตามทุกระบบจะมีแนวคิดอย่างเดียวกันคือ การใช้บริการจากชั้นล่างและการให้บริการแก่ชั้นบนโดยซ่อนรายละเอียดและความซับซ้อนของฟังก์ชันในแต่ละชั้นไว้ภายใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การสื่อสารที่เกิดขึ้นระหว่างผู้ส่งข้อมูลกับผู้รับข้อมูลนั้นจะเป็นการติดต่อของโปรแกรมแบบชั้นต่อชั้น หมายความว่าโปรแกรมในชั้นที่หนึ่งของโปรแกรมผู้ส่งจะติดต่อกับโปรแกรมในชั้นที่หนึ่งของโปรแกรมผู้รับ โปรแกรมชั้นสองจะติดต่อกับโปรแกรมชั้นสอง ฯลฯ กฎระเบียบในการติดต่อนี้เรียกว่า กฎการสื่อสารข้อมูล (Protocol) ซึ่งในแต่ละชั้นจะใช้กฎแตกต่างกัน กฎการสื่อสารข้อมูลนี้ช่วยให้ผู้ส่งข้อมูลและผู้รับข้อมูลสามารถติดต่อกันได้ หากผู้ใดผู้หนึ่งไม่ปฏิบัติตามกฎหรือไม่รู้จักกฎเหล่านี้การสื่อสารข้อมูลก็ไม่อาจกระทำได้



รูปที่ 2.8 ชั้นสื่อสาร,กฎการสื่อสารข้อมูลและการเชื่อมต่อ

ในรูปที่ 2.8 แสดงให้เห็นโปรแกรมสื่อสารแบบห้าชั้น ในแต่ละชั้นที่แสดงในรูปจะแทนด้วยรูปสี่เหลี่ยมเล็กๆ นั้น ก็คือโปรแกรมที่ทำหน้าที่สำหรับแต่ละชั้นที่กล่าวถึง โดยแต่ละชั้นจะมีกฎการสื่อสารข้อมูลเป็นของตนเอง ดังนั้นการสื่อสารที่เกิดขึ้นจึงหมายถึงการติดต่อระหว่างโปรแกรมเหล่านี้นั่นเอง

ในความเป็นจริงแล้วการสื่อสารจะเกิดขึ้นจริงโดยผ่านสายสื่อสารที่อยู่ใต้ชั้นล่างสุดเท่านั้น ข้อมูลที่สื่อสารในระหว่างชั้นต่างๆ จะถูกส่งต่อกันเป็นลำดับดังนี้ ชั้นบนสุด (ชั้นห้า) ของผู้ส่งจะส่งข้อมูลลงมายังชั้นสี่ ซึ่งโปรแกรมชั้นที่สี่ก็จะส่งข้อมูลนั้นรวมทั้งข้อมูลของตนเองลงมายังชั้นสาม และต่อมาเรื่อยๆ จนถึงชั้นล่างสุด ซึ่งข้อมูลทั้งหมดก็จะส่งผ่านสายสื่อสารจากผู้ส่งไปยังผู้ใช้ ทางฝ่ายผู้ใช้ก็จะรับข้อมูลในระดับชั้นล่างสุดก่อน ข้อมูลที่เป็นของชั้นนี้จะถูกแยกออกไป ส่วนข้อมูลที่เหลือก็จะถูกส่งขึ้นไปชั้นสอง ในทำนองเดียวกันข้อมูลที่เป็นของชั้นสองจะถูกแยกออกแล้วส่งส่วนที่เหลือต่อไปชั้นสาม ลักษณะเช่นนี้จะดำเนินต่อไป จนในที่สุดข้อมูลที่เป็นของชั้นห้าเท่านั้นที่จะถูกส่งขึ้นไปให้โปรแกรมชั้นที่ห้าเพื่อประมวลผลต่อไป ในระหว่างชั้นที่อยู่ติดกันของโปรแกรมสื่อสารจะมีส่วนที่ติดต่อถึงกัน (Interface) ซึ่งเราจะต้องมีการกำหนดมาตรฐานให้ชัดเจน มาตรฐานนี้คือการกำหนดรูปแบบการติดต่อในทางโปรแกรมคอมพิวเตอร์ อันได้แก่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. กำหนดจำนวนฟังก์ชันการทำงานพื้นฐานที่จำเป็นทั้งหมดที่ใช้ในการรับ-ส่งข้อมูลระหว่างชั้น

2. กำหนดหน้าที่ของแต่ละฟังก์ชันให้ชัดเจน

3. กำหนดจำนวนและประเภทตัวแปรต่างๆ ของแต่ละฟังก์ชัน

ทั้งนี้ในแต่ละชั้นยังมีฟังก์ชันการทำงานอื่นๆ อีกจำนวนหนึ่งซึ่งไม่เกี่ยวข้องกับส่วนติดต่อและการทำงานของชั้นอื่น โปรแกรมเหล่านี้เรียกได้ว่ามีความเป็นอิสระ (Independence) ต่อโปรแกรมส่วนอื่นๆ ดังนั้นการแก้ไขหรือเปลี่ยนแปลงโปรแกรมในส่วนนี้จึงสามารถทำได้โดยไม่มีข้อจำกัด ตัวอย่างที่ชัดเจนได้แก่ การเปลี่ยนอุปกรณ์สื่อสารข้อมูลในชั้นล่างสุดเป็นอุปกรณ์ชนิดใหม่จะบังคับให้แก้ไขโปรแกรมชั้นล่างสุดเพียงชั้นเดียวเท่านั้น โปรแกรมในชั้นอื่นจะไม่มีผลกระทบแต่อย่างใดหรือในกรณีที่ต้องการปรับปรุงการทำงานของโปรแกรมในชั้นใดชั้นหนึ่ง สิ่งที่ผู้เขียนโปรแกรมจะต้องระลึกถึงคือ การรักษามาตรฐานในส่วนของการติดต่อไว้เท่านั้น เมื่อปรับปรุงเสร็จแล้วก็สามารถนำโปรแกรมใหม่มาใช้แทนโปรแกรมเก่าได้ทันที

2.10.2 รูปแบบระบบเครือข่าย TCP/IP

ระบบเครือข่ายระดับโลกที่มีใช้อยู่ในปัจจุบันคือ ระบบอินเตอร์เน็ทนั้นมีกำเนิด มาจากระบบเครือข่ายชื่อ Arpanet ซึ่งได้รับการสนับสนุนให้ดำเนินการวิจัยโดยมีกระทรวงกลาโหมประเทศสหรัฐอเมริกาเป็นผู้ออกค่าใช้จ่าย ในยุคแรกๆนั้นเป็นระบบเครือข่ายที่เชื่อมการติดต่อระหว่างเครื่องคอมพิวเตอร์ในมหาวิทยาลัยและสถานที่ราชการหลายร้อยแห่งในสหรัฐอเมริกาเข้าด้วยกัน โดยใช้สายโทรศัพท์เช่า (Leased Lines) เป็นสายสื่อสารหลัก ต่อมาเมื่อระบบการสื่อสารแบบดิจิทัลความถี่สูงและการสื่อสารดาวเทียมเริ่มเข้ามามีบทบาทและนำมาใช้ในระบบมากขึ้นทำให้กฎการสื่อสารที่เคยใช้ได้ผลดีเกิดปัญหาไม่สามารถใช้งานได้อีกต่อไป

กฎการสื่อสารรุ่นต่อมาจึงได้รับการออกแบบเพื่อนำมาใช้ทดแทนแบบเก่าโดยมีวัตถุประสงค์ในการเชื่อมการติดต่อระหว่างระบบที่มีความแตกต่างกันเป็นเรื่องหลัก ผลที่ได้รับคือกฎการสื่อสารที่เรียกว่า กฎสื่อสารมาตรฐานแบบ TCP/IP ซึ่งได้รับการปรับปรุงจนนำมาใช้งานจริงได้ในปี ค.ศ.1974 การปรับปรุงรุ่นต่อมา สำเร็จในปี ค.ศ.1988

กฎสื่อสารมาตรฐานแบบ TCP/IP ยังมีวัตถุประสงค์หลักอีกสองข้อสำคัญคือ ความสามารถในการแก้ไขปัญหาที่เกิดขึ้นในระบบเครือข่าย เช่น ในกรณีที่ผู้ส่งและผู้รับยังคงมีการติดต่อกันอยู่แต่โหนดกลางที่ใช้เป็นผู้ช่วยรับ-ส่งข้อมูลเกิดเสียหายใช้การไม่ได้ หรือสายสื่อสารบางช่วงถูกตัดขาด กฎการสื่อสารนี้จะต้องสามารถจัดการหาทางเลือกอื่น เพื่อทำให้การสื่อสารดำเนินต่อไปได้โดยอัตโนมัติ ข้อที่สอง คือ จะต้องมีความอ่อนตัวต่อการสื่อสารข้อมูลได้หลายชนิดทั้งแบบที่ไม่มีความเร่งด่วน เช่น การจัดส่งแฟ้มข้อมูล และแบบที่ต้องการรับประกันความเร่งด่วนของข้อมูล เช่น การสื่อสารแบบ Real - Time หรือ การสื่อสารแบบโทรศัพท์ (Voice)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10.3 ชั้นการสื่อสารระบบ TCP/IP

ตารางที่ 2.3 โครงสร้าง TCP/IP

Application layer
Transport layer
Network layer
Data Link layer
Physical layer

ชั้นที่1. ชั้นฟิสิกส์คัล

ภายในชั้นฟิสิกส์คัล จะเป็นการเกี่ยวพันกับการส่งข้อมูลดิบเป็นบิตผ่านช่องสื่อสารข้อมูล วิธีการออกแบบจะต้องทำให้แน่ใจได้ว่าข้อมูล "1" ที่ส่งออกไปและที่ปลายทางสามารถรับ "1" ได้ถูกต้อง คำถามที่เกี่ยวข้องก็คือ จะต้องใช้แรงดันเท่าใดสำหรับแทนเลข "1" และเท่าใดสำหรับแทนเลข "0" และแต่ละบิดจะต้องห่างกันเท่าใด การส่งข้อมูลสามารถทำแบบสองทิศทางได้หรือไม่ ก่อนจะเริ่มต้นส่งข้อมูลจะต้องทำให้เกิดการเชื่อมต่ออย่างไร และเมื่อส่งเสร็จแล้วจะยกเลิกการต่ออย่างไร ลึกลงไปอีกก็คือ คอนเน็คเตอร์ (คอนเน็คเตอร์ หรืออุปกรณ์ที่ทำหน้าที่เป็นขั้วต่อของสายเข้ากับเครื่องคอมพิวเตอร์หรืออุปกรณ์สื่อสาร) ของเน็ตเวิร์ค จะใช้กี่ขาและแต่ละขาทำหน้าที่อะไร จะเห็นว่าประเด็นที่ต้องสนใจจะเกี่ยวกับทั้งทางไฟฟ้าทางรูปร่างภายนอกและก็ทั้งวิธีการเชื่อมต่อ รวมถึงตัวกลางที่อยู่ใต้ชั้นฟิสิกส์คัลซึ่งจะทำหน้าที่ส่งข้อมูลจริงๆ

ชั้นที่ 2. ชั้นดาต้าลิงค์

จุดประสงค์หลักของชั้นดาต้าลิงค์ ก็คือ พยายามทำให้การส่งข้อมูลดิบเหมือนกับไม่มีความผิดพลาดเกิดขึ้น ทำให้ชั้นที่สูงขึ้นไป (ซึ่งก็คือ ชั้นเน็ตเวิร์ค) นำไปใช้งานได้อย่างถูกต้อง วิธีการก็คือทางฝ่ายส่งจะทำการแตกข้อมูลออกเป็นก้อนๆเรียกว่า เฟรมข้อมูล (Data – Frame) (ปกติประมาณไม่กี่ร้อยไบต์) ทำการส่งเฟรมข้อมูลออกไปทีละชุดและรอรับการตอบรับ (Acknowledge Frame) ซึ่งตอบกลับมาโดยข้อมูล โดยปกติแล้วชั้นฟิสิกส์คัล จะไม่สนใจข้อมูลว่ามีความหมายใด จึงเป็นหน้าที่ของชั้นดาต้าลิงค์ ที่จะต้องทำการสร้างและตรวจจับขอบเขตของเฟรม (Frame Boundary) ซึ่งสามารถทำได้โดยการเติมบิตเข้าไปในจุดเริ่มต้นและจุดสุดท้ายของเฟรม แต่จุดที่ต้องระวังก็คือถ้าหากข้อมูลที่เพิ่มเข้าไปซ้ำกับข้อมูลที่ต้องการส่งจริงๆ

สัญญาณรบกวนจากภายนอกก็เป็นปัญหาหนึ่งที่อาจจะทำให้เฟรมขาดหายไปได้ ในกรณีนี้โปรแกรมที่ควบคุมชั้นดาต้าลิงค์ ที่เครื่องต้นทางจะส่งข้อมูลซ้ำมาใหม่ อย่างไรก็ตามการส่งเฟรมเดียวกันออกมาหลายๆ ครั้งก็อาจจะทำให้เกิดเฟรมซ้ำกันได้ วิธีการป้องกันก็คือเฟรมที่ซ้ำกันจะส่งออกไปก็ต่อเมื่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีการตอบรับ (Acknowledgment Frame) ส่งมาบอกว่าข้อมูลหายหรือถูกทำลายไปก่อนภายในชั้นดาต้าลิงค์ จะมีการเชื่อมต่อหลายๆ แบบเพื่อให้ชั้นเน็ตเวิร์คสามารถเลือกใช้ได้

ชั้นที่ 3. ชั้นเน็ตเวิร์ค

ภายในชั้นเน็ตเวิร์ค จะเกี่ยวข้องกับการควบคุมการทำงานของเครือข่ายย่อย ประเด็นที่สำคัญก็คือการพิจารณาว่า แพ็คเก็ตจะถูกส่งจากต้นทางไปยังปลายทางได้อย่างไร การหาเส้นทาง (Route) อาจจะวางอยู่บนตารางที่คงที่อยู่และเชื่อมโดยตรงเข้ากับเน็ตเวิร์คและมีการเปลี่ยนตารางน้อยมาก การกำหนดเส้นทางจะกำหนดตอนเริ่มต้นติดต่อเลยก็ได้หรืออาจจะใช้วิธีที่สามารถเปลี่ยนแปลงได้ตลอดเวลา (Dynamic) ซึ่งเส้นทางที่แพ็คเก็ตเดินทางไปจะถูกกำหนดแบบแพ็คเก็ตต่อแพ็คเก็ต

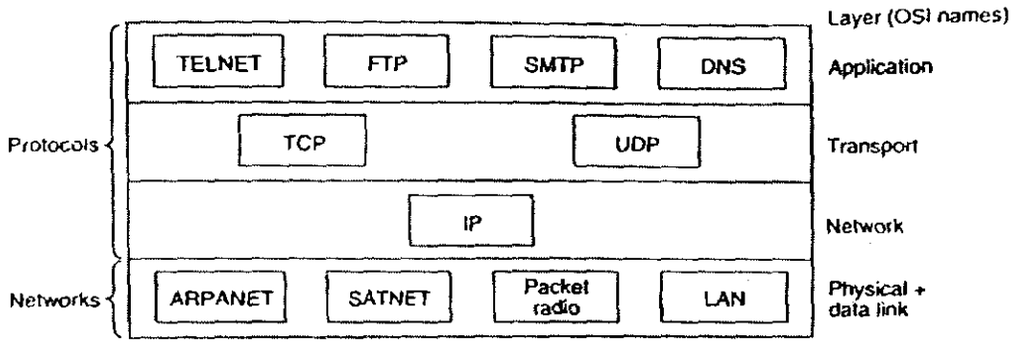
ถ้ามีจำนวนแพ็คเก็ตมากเกินไปภายในเครือข่ายย่อย จะทำให้เกิดปัญหาคอขวด คือจำนวนแพ็คเก็ต เข้ามามากแต่ทางออกน้อย ภายในชั้นเน็ตเวิร์คจะต้องจัดการกับปัญหาเหล่านี้ด้วย เมื่อแพ็คเก็ตจะต้องเดินทางจาก เน็ตเวิร์คหนึ่งไปยังเน็ตเวิร์คหนึ่งจะทำให้เกิดปัญหาต่างๆขึ้นเช่นแอดเดรสของเน็ตเวิร์คที่ 2 อาจจะแตกต่างจากเน็ตเวิร์คที่ 1 ทำให้เน็ตเวิร์คที่ 2 ไม่สามารถรับแพ็คเก็ตนี้ได้เลยหรือโปรโตคอลที่ใช้อาจจะแตกต่างกันชั้นเน็ตเวิร์คจะต้องจัดการกับปัญหาเหล่านี้เพื่อให้เน็ตเวิร์คทั้งหลายต่อถึงกันได้เสมือนเน็ตเวิร์คเดียวกัน ในระบบเน็ตเวิร์คแบบกระจาย (Broadcast Network) วิธีการหาเส้นทางของแพ็คเก็ตจะง่ายมาก ทำให้ชั้นเน็ตเวิร์คมีขนาดเล็กหรืออาจจะไม่มีเลยก็ได้

ชั้นที่ 4. ชั้นสื่อสารนำส่งข้อมูล

ชั้นสื่อสารนำส่งข้อมูลซึ่งมีหน้าที่การทำงานเหมือนกันกับชั้นจัดการนำส่งข้อมูลของมาตรฐาน OSI แบ่งออกเป็นโปรโตคอลสองประเภท ประเภทแรกเรียกว่า TCP (Transmission Control Protocol) เป็นแบบที่มีการกำหนดช่วงการสื่อสารตลอดระยะเวลาการสื่อสาร (Connection-Oriented) ซึ่งจะยอมให้มีการส่งข้อมูลเป็นกระแสไบต์ (Byte Stream) ที่ไว้ใจได้ (Reliable) โดยไม่มีข้อผิดพลาด ข้อมูลที่มีปริมาณมากจะถูกแบ่งออกเป็นส่วนเล็กๆ เรียกว่า Message ซึ่งจะถูกส่งไปยังผู้รับผ่านทางชั้นสื่อสารของอินเตอร์เน็ททางฝ่ายผู้รับจะนำ Message มาเรียงต่อกันตามลำดับเป็นข้อมูลดั้วเดิม TCP ยังมีความสามารถในการควบคุมการไหลของข้อมูลเพื่อป้องกันไม่ให้ผู้ส่ง ส่งข้อมูลเร็วเกินกว่าที่ผู้รับจะทำงานได้ทันอีกด้วย

กฎการส่งข้อมูลที่สองเรียกว่า UDP (User Datagram Protocol) เป็นการติดต่อแบบไม่ต่อเนื่อง (Connection Less) ไม่มีการตรวจสอบความถูกต้องของข้อมูลเหมือนกับแบบ TCP อย่างไรก็ตาม วิธีการนี้มีข้อดีในด้านความรวดเร็วในการส่งข้อมูลจึงนิยมใช้ในระดับผู้ให้และผู้ใช้บริการ (Client/Server System) ซึ่งมีการสื่อสารแบบถาม/ตอบ (Request/Reply) นอกจากนั้นยังใช้ในการส่งข้อมูลประเภทภาพเคลื่อนไหวหรือการส่งเสียงทางอินเตอร์เน็ท รูปที่ 2.9 แสดงให้เห็นความสัมพันธ์ของ IP, TCP, และUDP กฎการสื่อสารแบบ IP นี้เป็นระบบที่ได้รับความนิยมมากและได้รับการนำไปใช้ในระบบสื่อสารหลายระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 กฎการสื่อสารในรูปแบบ TCP/IP

ชั้นที่ 5. ชั้นสื่อสารการประยุกต์

ตามมาตรฐาน TCP/IP ไม่มีการกำหนดชั้นสื่อสารนำส่งข้อมูลและชั้นควบคุมหน้าต่างสื่อสารตามที่ปรากฏในรูปแบบมาตรฐาน OSI ทั้งนี้จากการกำหนดมาตรฐานของ OSI พบว่าโปรแกรมสำหรับชั้นควบคุมการสื่อสารสองชั้นนี้จะมีประโยชน์ในการใช้งานจริงน้อยมาก ระบบเครือข่าย TCP/IP จึงตัดทั้งสองชั้นนี้ออกไป ดังนั้นชั้นสื่อสารการประยุกต์ จึงกลายเป็นชั้นที่อยู่เหนือชั้นสื่อสารนำส่งข้อมูล

โปรแกรมในชั้นสื่อสารการประยุกต์จะรวมเอาหน้าที่การทำงานที่จำเป็นของสองชั้นที่หายไปไว้ที่นี่ อันได้แก่ โปรโตคอลสำหรับสร้างจอเทอร์มินัลเสมือน เรียกว่า TELNET โปรโตคอลสำหรับการจัดการแฟ้มข้อมูล เรียกว่า FTP และโปรโตคอลสำหรับการให้บริการด้านบริการจดหมายอิเล็กทรอนิกส์ เรียกว่า SMTP โปรโตคอลสำหรับจอเทอร์มินัลเสมือนช่วยให้ผู้ใช้สามารถติดต่อกับเครื่อง Host ที่อยู่ไกลออกไปโดยผ่านทางอินเตอร์เน็ทและสามารถทำงานได้เสมือนกับว่ากำลังนั่งทำงานอยู่ที่ Host นั้น โปรโตคอลสำหรับการจัดการแฟ้มข้อมูลช่วยในการคัดลอกแฟ้มข้อมูลมาจากเครื่องอื่นที่อยู่ในระบบเครือข่ายหรือส่งสำเนาแฟ้มข้อมูลไปยังเครื่องใดๆ ก็ได้ โปรโตคอลสำหรับให้บริการจดหมายอิเล็กทรอนิกส์ช่วยในการจัดส่งข้อความไปยังผู้ใช้ในระบบหรือรับข้อความที่มีผู้ส่งเข้ามา

2.10.4 การกำหนด IP Address

ที่อยู่บนอินเตอร์เน็ทกำหนดโดยใช้เลขฐานสองความยาว 4 ไบต์ หรือ 32 บิตซึ่งจะไม่มีที่อยู่ที่มีหมายเลขซ้ำกันเลย ประกอบด้วยตัวเลข 3 ประเภทคือ เลขบอก Class เลขบอกเครือข่าย และ เลขบอก Host การกำหนดที่อยู่แบ่งออกเป็น 4 Class คือ A, B, C, D, และ E ดังแสดงในรูปที่ 2.10 มีรายละเอียดดังนี้

Class A : บิตแรกเป็น " 0 " เลขบอกเครือข่ายยาว 7 บิตประกอบด้วย 126 เครือข่ายแต่ละเครือข่ายมี 16 ล้าน Host บอกด้วยตัวเลข 24 บิต ขอบเขตหมายเลขที่อยู่คือ 1.0.0.0 ถึง 127.255.255.255

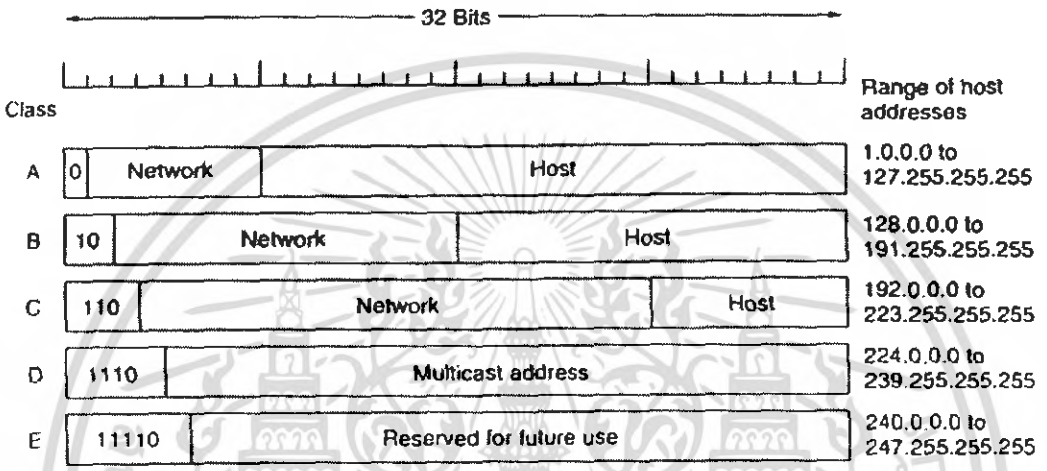
Class B : สองบิตแรกเป็น " 10 " เลขบอกเครือข่ายยาว 14 บิตประกอบด้วย 16,382 เครือข่าย แต่ละเครือข่ายมี 65,536 Host บอกด้วยตัวเลข 16 บิต ขอบเขตหมายเลขที่อยู่คือ 128.0.0.0 ถึง 191.255.255.255

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Class C : สามบิตแรกเป็น "110" เลขบอกเครือข่ายยาว 21 บิตประกอบด้วย 2,097,152 เครือข่ายแต่ละเครือข่ายมี 254 Host บอกด้วยตัวเลข 8 บิต ขอบเขตหมายเลขที่อยู่คือ 192.0.0.0 ถึง 223.255.255.255

Class D : สี่บิตแรกเป็น "1110" ใช้สำหรับการกระจายข้อมูลข่าวสารแบบ หลายจุดขอบเขตหมายเลขที่อยู่คือ 224.0.0.0ถึง 239.255.255.255

Class E : ห้าบิตแรกเป็น "11110" สำรองไว้ใช้ในอนาคต ขอบเขตหมายเลขที่อยู่คือ 240.0.0.0 ถึง 247.255.255.255



รูปที่ 2.10 โครงสร้างที่อยู่แบบ IP

2.11 คำสั่ง AT commands

คือชุดคำสั่งที่ใช้ในการติดต่อกับเครื่องโทรศัพท์หรืออุปกรณ์ติดต่อโดยส่งรหัสตัวอักษรที่ขึ้นต้นด้วย AT+คำสั่ง ตามด้วย รหัส CR + LF (Return)+(LineFeed)

ในรูปแบบคำสั่ง AT commands (Attention Commands) ใช้ในการตรวจสอบการติดต่อสื่อสารระหว่างโทรศัพท์และอุปกรณ์ คำสั่ง AT commands ที่ใช้ในโครงงานมีดังนี้

2.11.1 AT คือ คำสั่งตรวจสอบความพร้อมของโทรศัพท์เคลื่อนที่

รูปแบบคำสั่ง

AT<CR>

2.11.2 ATD คือ คำสั่งที่ใช้สำหรับการโทรออก

รูปแบบคำสั่ง

ATDyyyxxxxxxx;<CR>

- yyy หมายถึง รหัสขึ้นต้นเบอร์โทรศัพท์ เช่น 081 , 086 , 089
- xxxxxxx หมายถึง เบอร์โทรศัพท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.11.3 ATH คือ คำสั่งที่ใช้ในการวางสายโทรศัพท์ ยกเลิกการโทรออก

รูปแบบคำสั่ง

ATH<CR>

2.11.4 AT+CMGS คือ คำสั่งที่ใช้สำหรับส่งข้อความ

รูปแบบคำสั่ง

AT+CMGS = ความยาวของข้อมูลที่จะส่งตามด้วยรหัสแอสกี้ , <CR>

โดยความยาวของข้อมูลที่จะส่ง คือ

ความยาวของส่วน (Header + ความยาวของข้อมูลที่เป็น PDU Code –3)/2

หลังจากนั้นโทรศัพท์จะตอบกลับ ">" ให้เราส่งข้อมูลส่วน Header + PDU Code ตามด้วยรหัสแอสกี้ 26

2.11.5 AT+CMNI คือ คำสั่งที่ทำให้รู้ว่าได้รับข้อความเข้ามาที่โทรศัพท์ เอาไว้เก็บลำดับข้อความที่เข้ามาว่าเป็นข้อความที่เท่าไหร่

รูปแบบคำสั่ง

AT+CNMI=<mode>,<mt>,<bm>,<ds>,<bfr><CR>

กำหนดให้ใช้

AT+CNMI=1,1,2,2,1<CR>

2.11.6 AT+CMGR คือ คำสั่งที่ใช้สำหรับอ่านข้อความ

รูปแบบคำสั่ง

AT+CMGR=<index><CR>

- index ลำดับของข้อความ

เมื่อใช้คำสั่งอ่านข้อความจะได้รายละเอียดของข้อมูลดังนี้

ตารางที่ 2.4 รายละเอียดของข้อมูลในการส่งข้อความสั้น

ตัวอย่างข้อมูล การส่งข้อความสั้นเป็นข้อความว่า "Fire!" ไปที่หมายเลข 0407349 57	
06916681118088040A916686091635000030302002477182054669726521	
06	ความยาวของ SMSC Information 6 Octets (ไบต์)
91	รูปแบบเลขหมาย SMSC 91 หมายถึงเลขหมายแบบสากล
66 81 11 80 88	เลขหมาย SMSC (แบบ decimal semi-octets) ซึ่งจะเป็นเลขฐานสิบสลับ nibble
04	เป็น Octet แรกที่ใช้ในการส่งข้อความ
0A	ความยาวของหมายเลขผู้ส่ง (10 ตัว)
91	ประเภทของ Address

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.4 รายละเอียดของข้อมูลในการส่งข้อความสั้น (ต่อ)

66 86 09 16 35	เลขหมายผู้ส่ง (แบบ decimal semi-octets) ซึ่งจะเป็นเลขฐานสิบสลับ nibble
00	ระบุตัวโปรโตคอล
00	Data Coding Scheme
30 30 20 02 47 71 82	TP-SCTS. ข้อมูล Time stamp
05	จำนวนตัวอักษร
46 69 72 65 21	เป็น PDU Code ซึ่งมีข้อความ "Fire!"

2.11.7 AT+CMGL คือ คำสั่งเรียกดูข้อความ

รูปแบบคำสั่ง

AT+CMGL=<Stat><013>

- <Stat> = (0,1,2,3,4)

0 = ข้อความที่ยังไม่อ่าน

1 = ข้อความที่อ่านแล้ว

2 = ข้อความที่ไม่ได้ส่ง

3 = ข้อความที่ส่งแล้ว

4 = ดูข้อความทั้งหมด

2.11.8 AT+CMGD คือ คำสั่งที่ใช้สำหรับลบข้อความ

รูปแบบคำสั่ง

AT+CMGD=<index><CR>

- index ลำดับของข้อความ

2.12 ระบบการทำงานการรับส่งข้อความในโทรศัพท์เคลื่อนที่

European Tel ecommunication S tandard Institute (ETIS) เป็นองค์กรทำหน้าที่กำหนดมาตรฐานทางด้านโทรคมนาคมได้กำหนดมาตรฐานการส่งข้อความสั้นไว้ในคู่มือ GSM 03.40 และ GSM 03.38 สามารถส่งได้ถึง 160 ตัวอักษร โดยแต่ละตัวอักษรใช้รหัสขนาด 7 บิต ที่กำหนดไว้ในตารางที่ 2.5 นอกจากนั้นยังมีการใช้ตัวอักษรชนิดอื่นๆ เช่น ขนาด 8 บิตหรือ16 บิต ซึ่งมีวัตถุประสงค์เพื่อการใช้งานที่แตกต่างออกไป ซึ่งในโครงงานฉบับนี้จะพูดถึงเฉพาะแบบ 7 บิตเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.5 มาตรฐานคู่มือ GSM 03.40 และ GSM 03.38

HEX	Dec	Character name	Character	ISO-8859-1
0X00	0	COMMERCIALAT	@	64
0X01	1	POUND SIGN	£	163
0X02	2	DOLLAR SIGN	$	36
0X03	3	YEN SIGN	¥	165
0X04	4	LATIN SMALL LETTER E WITH GRAVE	è	232
0X05	5	LATIN SMALL LETTER E WITH ACUTE	é	233
0X06	6	LATIN SMALL LETTER U WITH GRAVE	ù	249
0X07	7	LATIN SMALL LETTER I WITH GRAVE	ì	236
0X08	8	LATIN SMALL LETTER O WITH GRAVE	ò	242
0X09	9	LATIN CAPITAL LETTER C WITH	Ç	199
0X0A	10	LINE FEED		10
0X0B	11	LATIN CAPITAL LETTER O WITH	Ø	216
0X0C	12	LATIN SMALL LETTER O WITH	ø	248
0X0D	13	CARRIAGE RETURN		13
0X0E	14	LATIN CAPITAL LETTER A WITH RING	Å	197
0X0F	15	LATIN SMALL LETTER A WITH RING	å	229
0X10	16	GREEK CAPITAL LETTER DELTA	Δ	
0X11	17	LOW LINE		95
0X12	18	GREEK CAPITAL LETTER PHI	Φ	
0X13	19	GREEK CAPITAL LETTER GAMMA	Γ	
0X14	20	GREEK CAPITAL LETTER LAMBDA	Λ	
0X15	21	GREEK CAPITAL LETTER OMEGA	Ω	
0X16	22	GREEK CAPITAL LETTER PI	Π	
0X17	23	GREEK CAPITAL LETTER PSI	Ψ	
0X18	24	GREEK CAPITAL LETTER SIGMA	Σ	
0X19	25	GREEK CAPITAL LETTER THETA	Θ	
0X1A	26	GREEK CAPITAL LETTER XI	Ξ	
0X1B	27	ESCAPE TO EXENSION TABLE		
0X1B0A	27 10	FORM FEED		12
0X1B14	27 20	CIRCUMLEX ACCENT	^	94

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.5 มาตรฐานคู่มือ GSM 03.40 และ GSM 03.38 (ต่อ)

0X1B28	27 40	LEFT CURLY BRACKET	{	123
0X1B29	27 41	RIGHT CURLY BRACKET	}	125
0X1B2F	27 47	REVERSE SOLIDUS (BACKLASH)	\	92
0X1B3C	27 60	LEFT SQUARE BRACKET	[	91
0X1B3D	27 61	TILDE	~	126
0X1B3E	27 62	RIGHT SQUARE BRACKET	]	93
0X1B40	27 64	VERTICAL BAR	|	124
0X1B65	27 101	EURO SIGN	€	164(ISO-8859-1)
0X1C	28	LATIN CAPITAL LETTER AE	Æ	198
0X1D	29	LATIN SMALL LETTER AE	æ	230
0X1E	30	LATIN SMALL LETTER SHARP S	ß	223
0X1F	31	LATIN SMALL LETTER E WITH	É	201
0X20	32	SPACE		32
0X21	33	EXCLAMATION MARK	!	33
0X22	34	QUOTATION MARK	"	34
0X23	35	NUMBER SIGN	#	35
0X24	36	CURRENCY SIGN	¤	164(ISO-8859-1)
0X25	37	PERCENT SIGN	%	37
0X26	38	AMPERSAND	&	38
0X27	39	APOSTROPHE	'	39
0X28	40	LEFT PARENTHESIS	(	40
0X60	96	INVERTED QUESTION MARK	¿	191
0X61	97	LATIN SMALL LETTER A	a	97
0X62	98	LATIN SMALL LETTER B	b	98
0X63	99	LATIN SMALL LETTER C	c	99
0X64	100	LATIN SMALL LETTER D	d	100
0X65	101	LATIN SMALL LETTER E	e	101
0X66	102	LATIN SMALL LETTER F	f	102
0X67	103	LATIN SMALL LETTER G	g	13
0X68	104	LATIN SMALL LETTER H	h	104
0X69	105	LATIN SMALL LETTER I	i	105

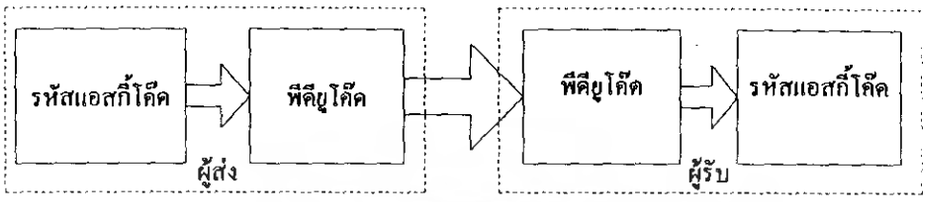
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.5 มาตรฐานคู่มือ GSM 03.40 และ GSM 03.38 (ต่อ)

0X6A	106	LATIN SMALL LETTER J	j	106
0X6B	107	LATIN SMALL LETTER K	k	107
0X6C	108	LATIN SMALL LETTER L	l	108
0X6D	109	LATIN SMALL LETTER M	m	109
0X6E	110	LATIN SMALL LETTER N	n	110
0X6F	111	LATIN SMALL LETTER O	o	111
0X70	112	LATIN SMALL LETTER P	p	112
0X71	113	LATIN SMALL LETTER Q	q	113
0X72	114	LATIN SMALL LETTER R	r	114
0X73	115	LATIN SMALL LETTER S	s	115
0X74	116	LATIN SMALL LETTER T	t	116
0X75	117	LATIN SMALL LETTER U	u	117
0X76	118	LATIN SMALL LETTER V	v	118
0X77	119	LATIN SMALL LETTER W	w	119
0X78	120	LATIN SMALL LETTER X	x	120
0X79	121	LATIN SMALL LETTER Y	y	121
0X7A	122	LATIN SMALL LETTER Z	z	122
0X7B	123	LATIN SMALL LETTER A WITH	ä	228
0X7C	124	LATIN SMALL LETTER O WITH	ö	246
0X7D	125	LATIN SMALL LETTER N WITH TILDE	ñ	241
0X7E	126	LATIN SMALL LETTER U WITH	ü	252
0X7F	127	LATIN SMALL LETTER A WITH GRAVE	à	224

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การรับส่งข้อความแบบพีดียูโหมด คือ ในการเขียนตัวอักษรหรือข้อความจะเหมือนกับเท็กซ์โหมด แต่ในการส่งข้อความจะมีการแปลงจากรหัสแอสกี้โค๊ดให้เป็นรหัสพีดียูโค๊ดและนำรหัสพีดียูโค๊ดที่ได้ส่งให้กับเลขหมายปลายทาง และเลขหมายปลายทางจะต้องแปลงจากรหัสพีดียูโค๊ดให้เป็นรหัสแอสกี้โค๊ดโดยแสดงดังรูปที่ 2.11



รูปที่ 2.11 การรับส่งข้อความแบบพีดียูโหมด

การแปลงข้อมูลแอสกี้โค๊ดเป็นพีดียูโค๊ด เนื่องจากเป็นขบวนการย้อนกลับ กล่าวคือให้เปลี่ยนรหัสแอสกี้โค๊ดทั้งหมด แปลงให้อยู่ในเลขฐานสองหลังจากนั้นให้ตัดบิตบนสุดของแต่ละไบต์ทิ้งให้เหลือ 7 บิต ดังนี้

ตารางที่ 2.6 การแปลงเลขฐานสิบหกเป็นเลขฐานสอง

เลขฐานสิบหก	เลขฐานสอง
46	100 0110
69	110 1001
72	111 0010
65	110 0101
21	010 0001

หลังจากนั้นให้ทำการชิพบิตไปทางขวา โดยการชิพบิตมีหลักเกณฑ์ดังนี้ คือให้เริ่มชิพจากบรรทัดที่สองเป็นต้นไป โดยจำนวนการชิพจะเท่ากับจำนวนบรรทัดลบด้วย 1 เสมอ เมื่อชิพบิตแล้วข้อมูลในบรรทัดนั้นจะเหลือเท่ากับ 7 เช่น ชิพบรรทัดที่สองจำนวนชิพเท่ากับ 2-1 ข้อมูลบรรทัดจะเท่ากับ 7-1 คือเท่ากับ 6 ตัว ในบรรทัดที่สามจะเหลือเท่ากับ 5 ตัวเป็นต้น ข้อมูลที่ถูกชิพออกมาให้นำไปต่อที่หัวบรรทัดก่อนหน้าเสมอ แต่กรณีที่ชิพจนถึง 7 บิตแล้วจะเป็นการเริ่มขบวนการใหม่ จะได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.7 การแปลงเลขฐานสองเป็นเลขพีดียูโค๊ด

เลขฐานสอง(แอสกี้โค๊ด)	เลขฐานสอง(พีดียู)	พีดียูโค๊ด
100 0110	1100 0110	C6
110 1001	1011 0100	B4
111 0010	1011 1100	BC
110 0101	0001 1100	1C
010 0001	0000 0010	02
000 0000		

การแปลงข้อมูลพีดียูโค๊ดเป็นแอสกี้โค๊ด เริ่มจากนำพีดียูโค๊ดทั้งหมดแปลงให้อยู่ในเลขฐานสองเพื่อง่ายต่อความเข้าใจดังนี้

ตารางที่ 2.8 การแปลงเลขพีดียูเป็นเลขฐานสอง

พีดียูโค๊ด	เลขฐานสอง
C6	1100 0110
B4	1011 0100
BC	1011 1100
1C	0001 1100
02	0000 0010

หลังจากนั้นให้ทำการชิพบิตบนสุดไปทางซ้าย โดยการชิพบิตมีหลักเกณฑ์ดังนี้ คือให้เริ่มชิพจากบรรทัดแรกโดยจำนวนการชิพจะเท่ากับจำนวนบรรทัด เมื่อชิพบิตแล้วให้นำเลข "0" มาใส่ไว้ข้างหน้าหนึ่งตัวเสมอ ข้อมูลที่ถูกชิพออกมาในแต่ละบรรทัดให้นำไปต่อที่บิตสุดท้ายของบรรทัดถัดไป เช่น ข้อมูลที่ถูกชิพมาจากบรรทัดที่หนึ่งจะถูกชิพมาต่อท้ายบรรทัดที่สองดังนี้

ตารางที่ 2.9 การแปลงเลขฐานสองเป็นรหัสแอสกี้โค๊ด

เลขฐานสอง	เลขฐานสิบหก	รหัสแอสกี้โค๊ด
1100 0110	46	F
1011 0100,1	69	i
1011 1100,10	72	r
0001 1100,101	65	e
0000 0010,0001	21	!

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่กรณีที่ชิพจนถึง 7 บิตแล้วจะเป็นการเริ่มขบวนการใหม่นั้นเอง ในกรณีที่เมื่อทำจนถึงตัวสุดท้าย แต่ว่าข้อมูลตัวสุดท้ายไม่ครบ 8 บิตที่จะอ่านเป็นแอสกี้โค๊ดได้ให้เติม "0" จนครบแล้วอ่านข้อมูลที่เป็นเลขฐานสองเปลี่ยนให้เป็นเลขฐานสิบหกและแทนในรหัสแอสกี้โค๊ด

2.13 ภาคถอดรหัสสัญญาณ DTMF

โทรศัพท์ชนิดกดปุ่มแบบ DTMF ประกอบด้วยปุ่มกดจำนวน 12 ปุ่ม มีการทำงานเป็นแบบ Dual Tone Multi Frequency แป้นกดจะแบ่งเป็นแถว (Row) 4 แถว และหลัก (Column) 3 หลัก ประกอบกันเป็นรูป Matrix ในแต่ละหลักจะมีค่าความถี่ประจำตำแหน่งอยู่ ดังตารางที่ 2.10

ตารางที่ 2.10 แสดงค่าความถี่ประจำหมายเลขในระบบโทรศัพท์แบบกดปุ่ม

	C1 1209 เฮริต์	C2 1336 เฮริต์	C3 1447 เฮริต์
R1 697 เฮริต์	1	2	3
R2 770 เฮริต์	4	5	6
R3 852 เฮริต์	7	8	9
R4 941 เฮริต์	*	0	#

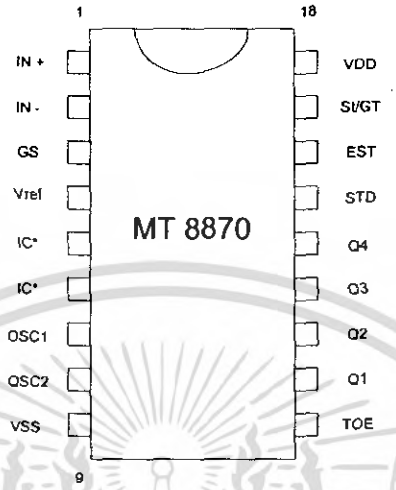
เมื่อกดปุ่มหมายเลขใดเลขหนึ่ง ปุ่มหนึ่งจะประกอบไปด้วย Tone เสียง สองความถี่ด้วยกัน คือ ความถี่สูง ความถี่ต่ำ (ทางด้านหลักและแถวตามลำดับ) ซึ่งแต่ละหมายเลขจะให้ค่าความถี่ออกมาสองความถี่ด้วยกัน จากปุ่มกดจะพบว่าหมายเลข 1 , 4 , 7 และ * อยู่ในหลักที่ 1 โดยหมายเลข 1 , 2 , 3 อยู่ในแถวที่ 1 ตัวเลขแต่ละตัวเป็นการพบกันของความถี่ทางแถว และความถี่ทางหลัก ยกตัวอย่างเช่น เมื่อกดปุ่มหมายเลข " 5 " จะอยู่ในหลักของ 1336 เฮริต์ และแถวของ 770 เฮริต์ ดังนั้นจะได้ความถี่เอาท์พุทออกมาสองความถี่คือ 1336 เฮริต์ และ 770 เฮริต์ ซึ่งเรียกว่า DTMF ดังนั้นในการสร้างวงจรก็เพื่อถอดรหัสค่าความถี่เหล่านี้นั่นเอง ในปัจจุบันได้มีอุปกรณ์ที่ทำหน้าที่นี้โดยเฉพาะคือ ไอซี MT 8870 ของบริษัท Intel ซึ่งทำหน้าที่รับสัญญาณ DTMF แปลงให้เป็นค่าตัวเลขฐานสองขนาด 4 บิต ซึ่งโครงงานนี้จะใช้ไอซีเบอร์นี้เป็นตัวถอดรหัสสัญญาณ DTMF

คุณสมบัติของไอซี MT 8870

- เป็นตัวรับและถอดรหัสความถี่ DTMF
- กินไฟน้อย ใช้ไฟเลี้ยงระดับเดียวกับ TTL
- สามารถตั้งอัตราการขยายภายในตัวไอซีได้
- สามารถปรับการ์ดไทม์
- เป็นไอซีคุณภาพสูง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไอซี MT 8870 นี้จะทำหน้าที่รับสัญญาณ DTMF ซึ่งเป็นสัญญาณอนาล็อกสองความถี่ ได้เป็น อย่างดี ถึงแม้ว่าสัญญาณที่ได้จากการกดปุ่มของโทรศัพท์แบบทัชโทนจะมีลักษณะไม่เป็นคลื่นไซน์ที่แท้จริงก็ตาม ไอซีเบอร์นี้ก็ยังสามารถยอมรับและถอดรหัสออกมาได้



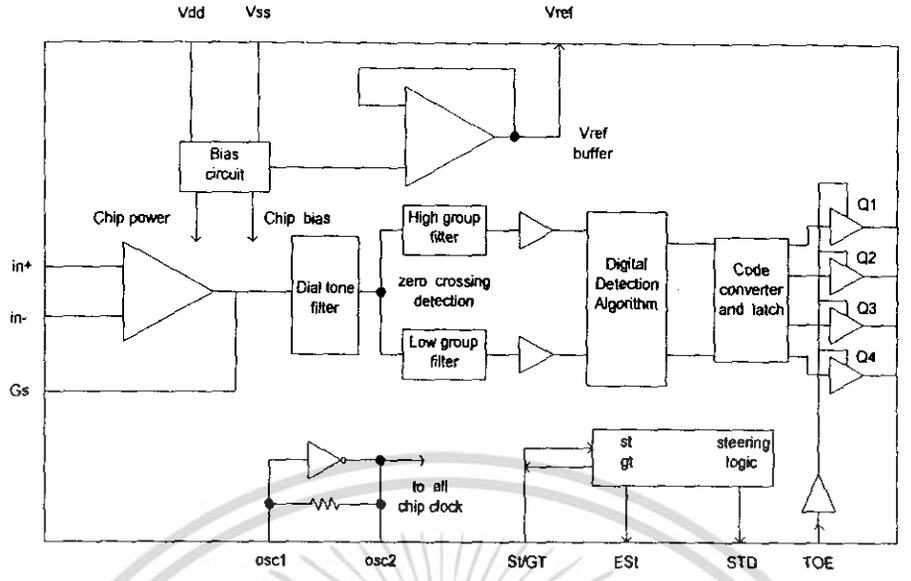
รูปที่ 2.12 แสดงรายละเอียดขาของไอซี MT 8870

2.13.1 โครงสร้างภายในไอซี MT 8870

เนื่องจากเครื่องโทรศัพท์ที่ใช้อยู่ในปัจจุบันนั้นส่วนมาก เป็นระบบกดปุ่มหรือที่เรียกกันว่า DTMF ซึ่งTone เสียงดังกล่าวสามารถนำมาถอดรหัสให้เป็นสัญญาณดิจิตอล เพื่อนำมาใช้ประโยชน์ทางด้านการควบคุม เช่น งานด้านรีโมทคอนโทรล

โครงสร้างของไอซี MT 8870 ประกอบไปด้วยวงจรกรองความถี่ และวงจรถอดรหัสฟังก์ชันทางดิจิตอล ซึ่งเป็นไอซีที่สร้างขึ้นโดยใช้เทคโนโลยี ISO2 – CMOS ในส่วนของวงจรกรองความถี่ใช้เทคนิคของสวิทช์คาปาซิเตอร์ สำหรับกรองความถี่สูงและต่ำ ส่วนวงจรถอดรหัสใช้เทคนิคการนับทางดิจิตอล เพื่อตรวจจับสัญญาณและถอดรหัสทั้ง 16 ความถี่ให้ออกเป็นเลขฐานสองขนาด 4 บิตและเช็คช่วงเวลาที่สัญญาณเข้ามาส่วนภาคอินพุทเป็นออปแอมป์ซึ่งสามารถปรับอัตราขยายได้ ส่วนภาควงจรเอาท์พุทเป็นวงจรแลตช์ 3 สถานะดังรูปที่ 2.13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 แสดงโครงสร้างภายในของไอซี MT 8870

2.13.2 ฟังก์ชันการทำงานภายในของไอซี MT 8870

1. ภาคกรองความถี่ (Filter Section) ในส่วนนี้จะแยกสัญญาณ DTMF เข้ามาออกเป็นสองกลุ่มความถี่คือ ช่วงความถี่สูงและช่วงความถี่ต่ำโดยใช้วงจรกรองแถบความถี่อันดับ 6 ชนิดสวิทช์คาปาซิเตอร์ (Six – Order Switched Capacitor Band Pass Filter)

2. ภาคถอดรหัส (Decoder) ความถี่ที่ได้ถูกกรองเรียบร้อยแล้วจะผ่านเข้าวงจรถอดรหัสความถี่ออกเป็นตัวเลข โดยใช้เทคนิคการนับแบบดิจิตอลและมีการตรวจสอบความถี่ที่เข้ามาว่าเป็นความถี่มาตรฐาน DTMF หรือไม่เพื่อป้องกันความถี่อื่นเข้ามาผสม เมื่อตรวจสอบได้ว่าความถี่ถูกต้องสัญญาณที่ขา Est (Early Steering) ก็จะแอคทีฟ สำหรับค่าที่ถอดรหัสได้จากความถี่ต่างๆ ดัง ตารางที่ 2.11

ตารางที่ 2.11 แสดงตารางค่าถอดรหัสที่ได้จากความถี่ต่างๆ

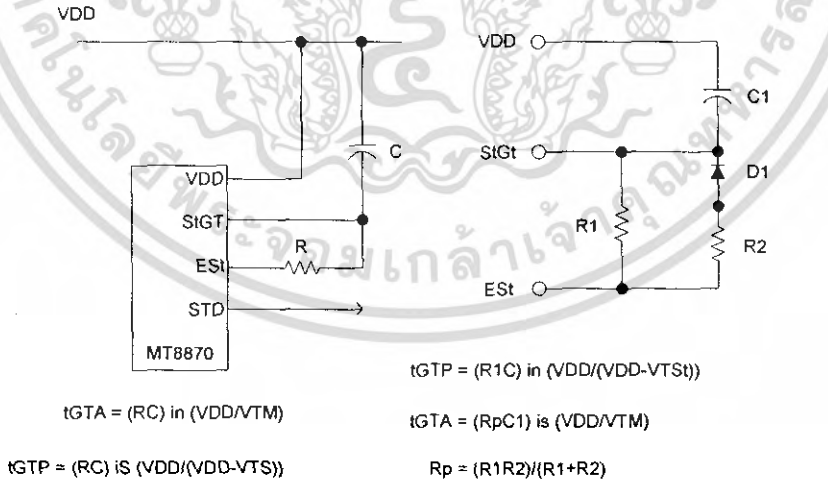
F_{Low}	F_{High}	NO.	TOE	Q4	Q3	Q2	Q1
697	1209	1	H	0	0	0	1
697	1336	2	H	0	0	1	0
697	1477	3	H	0	0	1	1
770	1209	4	H	0	1	0	0
770	1336	5	H	0	1	0	1
770	1477	6	H	0	1	1	0
852	1209	7	H	0	1	1	1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.11 แสดงตารางค่าถอดรหัสที่ได้จากความถี่ต่างๆ (ต่อ)

852	1336	8	H	1	0	0	0
852	1477	9	H	1	0	0	1
941	1336	0	H	1	0	1	0
941	1209	*	H	1	0	1	1
941	1477	#	H	1	1	0	0
697	1633	A	H	1	1	0	1
770	1633	B	H	1	1	1	0
852	1633	C	H	1	1	1	1
941	1633	D	H	0	0	0	0
-	-	-	ANY	Z	Z	Z	Z

3. ภาคตรวจสอบสัญญาณ (Steering Circuit) ก่อนที่จะมีการถอดรหัสความถี่ออกไปที่เอาท์พุท จะมีการตรวจสอบช่วงความถี่ที่เข้ามาว่ามีระยะเวลาตามกำหนดหรือไม่ โดยสังเกตจากระยะเวลาการกดปุ่มที่หน้าปัดของโทรศัพท์ ซึ่งต้องกดให้มีความถี่ออกมาเป็นช่วง ๆ เป็นเวลาพอสมควร ไม่เช่นนั้นวงจรส่วนนี้จะรับไม่ได้ เนื่องจากว่าสัญญาณนั้นไม่ถูกต้อง ส่วนช่วงเวลาจะยาวเท่าใดนั้น สามารถตั้งได้โดย RC ต่อภายนอก สัญญาณที่ขา Est จะเป็นระดับ High นานใกล้เคียงกับระยะเวลาที่ความถี่ DTMF เข้ามา เมื่อขา Est เป็น "High" ทำให้ Vc สูงขึ้นถึงค่าเทรชโฮลด์วงจรถอดรหัสจึงจะถอดรหัสออกเป็นเลข 4 บิต ดังรูปที่ 2.14



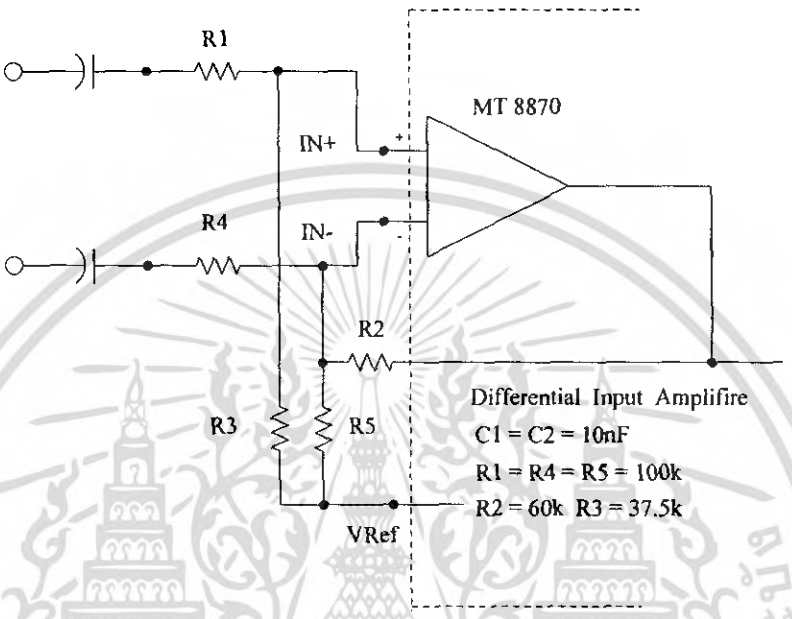
รูปที่ 2.14 แสดงวงจรตรวจสอบสัญญาณและกำหนดเวลาการ์ดไทม์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ภาคขยายสัญญาณความแตกต่าง (Differential Input) วงจรส่วนอินพุทของไอซี MT 8870 เป็นภาคขยายออปแอมป์ที่สามารถปรับอัตราขยาย โดยต่อวงจรภายนอกเพิ่มเข้าไปดังรูปที่ 2.15 แสดงการต่อวงจรภายนอกเข้ากับอินพุท ซึ่งสามารถคำนวณอัตราขยายความแตกต่างของอินพุทอิมพิแดนซ์ได้

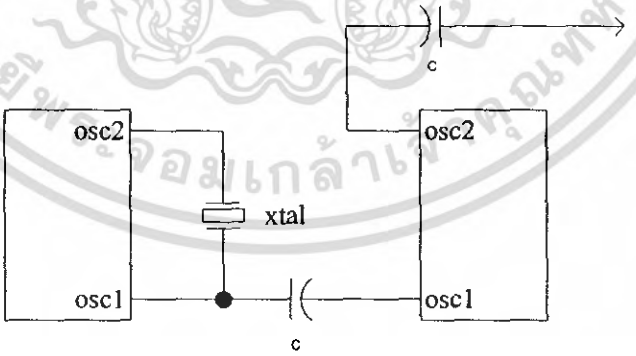
อัตราขยาย (Av diff) = R_5/R_1

อินพุทอิมพิแดนซ์ (Zin diff) = $2((R_1^2 + (1/Wc)^2)^{1/2})$



รูปที่ 2.15 แสดงการต่อวงจรภาคอินพุท

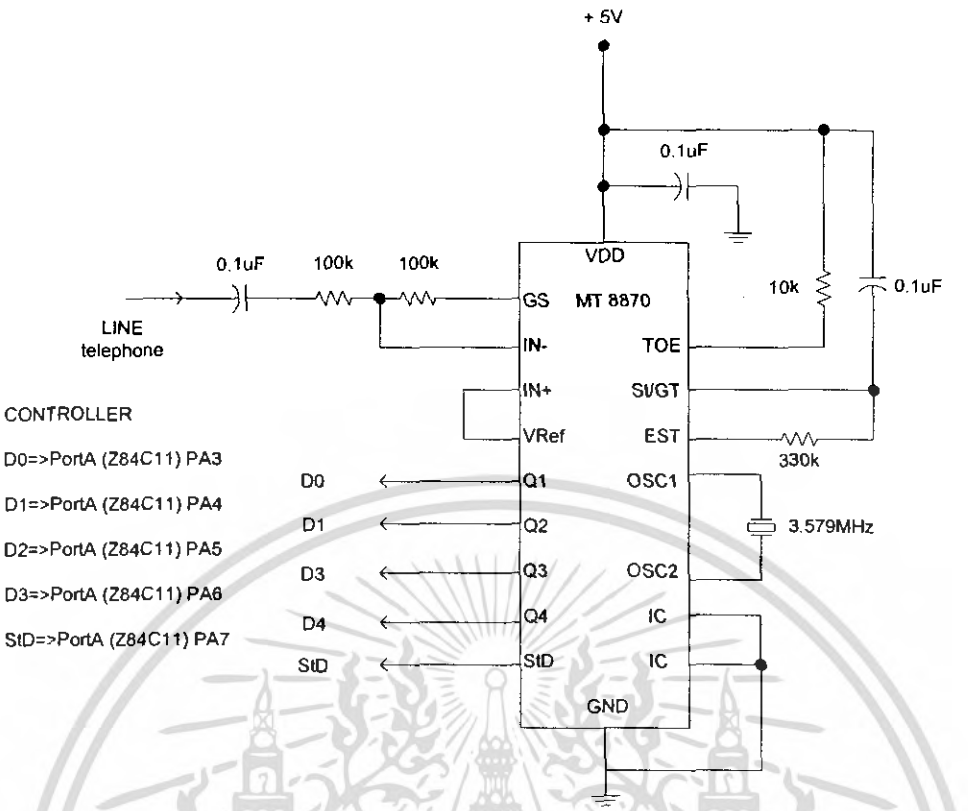
5. ภาค Oscillator ในภาคนี้ไอซีจะมีวงจรอยู่แล้วภายใน เพียงแค่ต่อคริสตอล ก็สามารถใช้งานได้ทันทีดังรูปที่ 2.16 และรูปที่ 2.17



c = 30pf xtal = 3.58MHz

รูปที่ 2.16 แสดงการต่อวงจร Oscillator

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.17 แสดงวงจรใช้งานของไอซี MT 8870

สัญญาณ DTMF ที่ถูกส่งมาจากภาคส่งสัญญาณ ภาคส่งสัญญาณจะถูกแยกองค์ประกอบของสัญญาณเพื่อนำเอาข้อมูลที่ได้ไปเข้ารหัสเลขฐานสองอีกครั้งหนึ่ง ก่อนส่งไปควบคุมที่ภาคเอาท์พุท เกือบทั้งหมดนี้เราสามารถทำงานโดยใช้ไอซีเพียงตัวเดียว ในโครงงานนี้ใช้ ไอซี MT 8870 ซึ่งเป็นไอซีถอดรหัสที่ใช้กันอย่างแพร่หลาย

โครงสร้างภายในของตัวมันประกอบไปด้วย ภาคกรองความถี่ ภาคถอดรหัส ภาคตรวจสอบสัญญาณ ภาคขยายอินพุทและภาคกำเนิดความถี่

การทำงานจะเริ่มจากสัญญาณจากภาคส่งสัญญาณถูกป้อนเข้ามาทางอินพุท ซึ่งจะถูกขยายโดยออปแอมป์ จากนั้นสัญญาณที่ถูกขยายแล้วจะถูกส่งต่อไปยังภาคกรองสัญญาณ ซึ่งมีอยู่ 2 ส่วน คือ ส่วนกรองสัญญาณ DTMF เป็นแบนด์พาสฟิลเตอร์ ทำหน้าที่กรองเอาเฉพาะสัญญาณ DTMF ออกมาเนื่องจากสัญญาณที่ส่งมาจากภาคส่งสัญญาณ อาจมีสัญญาณอื่นปะปนเข้ามาระหว่างทางได้ จึงจำเป็นต้องกรองออกไป และยังมีส่วนกรองสัญญาณความถี่สูงกับกรองความถี่ต่ำ

ที่ภาคกรองความถี่สูงจะกรองเฉพาะกลุ่มความถี่สูงของ DTMF ภาคกรองความถี่ต่ำจะกรองเฉพาะกลุ่มความถี่ต่ำของ DTMF จากนั้นสัญญาณทั้งสองส่วนที่ผ่านวงจรกรองแล้วจะถูกส่งต่อไปยังภาคถอดรหัสเพื่อทำการแปลงความถี่ DTMF ที่ได้นั้นให้เป็นรหัสเลขฐานสอง

แต่ถึงแม้ว่าภาคกรองความถี่จะมีถึง 2 ส่วน แต่ก็ไม่แน่ว่าความถี่ที่ผ่านวงจรกรองมาได้จะเป็นความถี่ DTMF ที่ถูกต้องเสมอไป แต่อาจเป็นสัญญาณรบกวนที่มีความถี่ใกล้เคียงกันก็ได้ ดังนั้นจึงต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีการตรวจสอบสัญญาณเพื่อตรวจสอบว่า ความถี่ที่ผ่านวงจรกรองมาได้ ใช่ความถี่ DTMF ที่ต้องการหรือไม่ โดยมีการตรวจสอบทั้งที่ภาคถอดรหัสสัญญาณ และภาคตรวจสอบสัญญาณ ซึ่งที่ภาคตรวจสอบสัญญาณจะตรวจสอบดูว่าช่วงเวลาที่สัญญาณปรากฏ สั้นกว่าเวลาที่ตั้งไว้ก็จะถือว่าสัญญาณนั้นมีความผิดพลาด ไม่ใช่สัญญาณที่ต้องการ ก็จะไปควบคุมสภาวะเอาท์พุทบัฟเฟอร์ของไอซีให้เป็นไฮอิมพิแดนซ์ นั่นคือจะไม่มีสัญญาณเอาท์พุทของไอซี

การถอดรหัส DTMF จะใช้ ไอซี MT 8870 เป็นตัวถอดรหัส โดยสัญญาณ DTMF ที่ถูกส่งมาจากภาคกำเนิดสัญญาณควบคุมจะถูกคับปลิ้งผ่าน C1 ไปยังไอซีถอดรหัส ซึ่งจะทำการแปลงสัญญาณ DTMF ให้เป็นรหัสเลขฐานสอง 4 บิต ออกมาทางขา Q1 , Q2 , Q3 และ Q4 โดยที่ Q1 เป็นตำแหน่งบิตสูงสุดและ Q4 เป็นตำแหน่งบิตต่ำสุด จากนั้นรหัสเลขฐานสอง 4 บิต จะถูกส่งต่อไปยังไอซีเบอร์ 74154 ซึ่งเป็นไอซีถอดรหัสเข้า 4 ออก 16 รหัสไบนารี่จะถูกแปลงเป็นหมายเลขต่าง ๆ แล้วแสดงผลออกทางขาหมายเลขนั้น ซึ่งจะถูกใช้เป็นสัญญาณนาฬิกาป้อนให้กับภาคตัดต่อสวิทช์

ในส่วนไอซี MT 8870 นี้จะใช้คริสตอล 3.579 เมกะเฮริต์ ต่อเข้ากับขา Oscillator ทั้ง 2 ขาของไอซี ซึ่งภายในจะเป็นนอตเกต ทำหน้าที่เป็นตัวกำเนิดสัญญาณอ้างอิงภายในไอซีเพื่อใช้ถอดรหัสความถี่ DTMF

2.14 โครงข่ายของระบบเครือข่าย (Topology)

เป็นการเชื่อมโยงเครือข่ายของระบบ LAN วิธีหนึ่ง ซึ่งนิยมใช้กันแพร่หลายสามารถแบ่งออกเป็น 3 แบบด้วยกัน คือ

2.14.1 แบบดาว (Star)

2.14.2 แบบวงแหวน (Ring)

2.14.3 แบบบัส และ ทรี (Bus and Tree)

2.14.1 แบบดาว

ในโทโปโลยีแบบดาว นั้นจะเป็นลักษณะของการต่อเครือข่ายที่ Work station แต่ละตัวต่อรวมเข้าสู่ศูนย์กลางสวิทช์ เพื่อสลับตำแหน่งของเส้นทางของข้อมูลใด ๆ ในระบบ ดังนั้นในโทโปโลยี แบบดาว คอมพิวเตอร์จะติดต่อกันได้ใน 1 ครั้งต่อ 1 คู่สถานีเท่านั้น เมื่อสถานีใดต้องการส่งข้อมูลมันจะส่งข้อมูลไปยังศูนย์กลางสวิทช์ก่อน เพื่อบอกให้ศูนย์กลางสวิทช์มันสลับตำแหน่งของคู่สถานีไปยังสถานีที่ต้องการติดต่อด้วย ดังนั้นข้อมูลจึงไม่เกิดการชนกันเอง ทำให้การสื่อสารได้รวดเร็วเมื่อสถานีใดสถานีหนึ่งเสีย ทั้งระบบจึงยังคงใช้งานได้ ในการค้นหาข้อบกพร่องจุดเสียต่างๆ จึงหาได้ง่ายตามไปด้วย แต่ก็มีข้อเสียที่ว่าต้องใช้งบประมาณสูงในการติดตั้งครั้งแรก

2.14.2 แบบวงแหวน

ในโทโปโลยี แบบวงแหวนนั้น ได้ถูกออกแบบให้ใช้ MAU (Media Access Units) ต่อรวมกันแบบเรียงลำดับเป็นวงแหวน แล้วจึงต่อ คอมพิวเตอร์ ที่เป็น Workstation หรือ Server เข้ากับ MAU ใน MAU 1 ตัวจะสามารถต่อออกไปได้ถึง 8 สถานี เมื่อสถานีถัดไปนั้นรับรู้ว่าต้องรับข้อมูล แล้วมันจึงส่งข้อมูลกลับ เป็นการตอบรับ เมื่อสถานีที่จะส่งข้อมูลได้รับสัญญาณตอบรับ แล้วมันจึงส่งข้อมูลครั้งแรก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แล้วมันจะลบข้อมูลออกจากระบบ เพื่อให้ได้ใช้ข้อมูลอื่นๆ ต่อไป ดังนั้นทุกสถานีบนโทโปโลยีวงแหวนจะได้ทำงานทั้งหมดซึ่งจะคอยเป็นผู้รับและผู้ส่งแล้วยังเป็นรีพีทเตอร์ในตัวอีกด้วย ข้อมูลที่ผ่านไปแต่ละสถานีนั้น ข้อมูลที่เป็นตำแหน่งที่อยู่ตรงกับสถานีใด สถานีนั้นจะรับข้อมูลเก็บไว้แต่มันจะไม่ลบข้อมูลออกจากระบบ มันยังคงส่งข้อมูลต่อไป ดังนั้นผู้ส่งข้อมูลครั้งแรกเท่านั้นที่จะเป็นผู้ลบข้อมูลออกจากระบบ ครั้นเมื่อสถานีส่ง TOKEN มาถามสถานีถัดไปแล้วแต่กลับไม่ได้รับคำตอบ สถานีส่ง TOKEN จะทวนซ้ำข้อมูลเป็นครั้งที่สอง ถ้ายังคงไม่ได้รับคำตอบ จึงส่งข้อมูลออกไปได้ เหตุการณ์ดังกล่าวนี้เป็นอีกแนวทางหนึ่งในการแก้ปัญหา ที่ไม่ให้ระบบหยุดชะงักการทำงานลงของระบบเนื่องจากสถานีหนึ่งเกิดข้อบกพร่องหรือชำรุด ระบบจึงยังคงสามารถทำงานต่อไปได้

2.14.3 แบบบัส

ในโทโปโลยี แบบบัสและทรี นั้นได้มีการทำงานที่คล้ายกันกล่าวคือ แบบบัส จะมีเคเบิลต่อถึงกันแบบขนาน ของแต่ละโหนด ส่วนแบบทรีนั้น จะมีการต่อแยกออกเป็นสาขาออกไปจากเคเบิลที่ใช้แบบบัสนั้นเอง ดังนั้นเมื่อมีการส่งข้อมูลจากโหนดใดทุกๆ โหนดบนระบบข้อมูลจะเข้าถึงได้ เนื่องจากอยู่บนเส้นทางสื่อสารเดียวกัน ในการส่งข้อมูลนั้น จะส่งเป็นเฟรม ข้อมูลซึ่งจะมีที่อยู่ของผู้รับติดไปด้วย เมื่อที่อยู่ผู้รับตรงกับตำแหน่งของโหนดใดๆ บนระบบ โหนดนี้จะรับข้อมูลเข้าไปและส่งข้อมูลมาพร้อมกันนั้นจะเกิดการชนกันของข้อมูล แล้วจะสุ่มเวลาขึ้นใหม่เพื่อส่งข้อมูลต่อไป ในการสื่อสารตามมาตรฐาน 802.4 นั้นมีด้วยกัน 3 แบบคือ แบบที่ 1 มีความเร็ว 1 เมกะบิตต่อวินาที ใช้สายข้อมูลแบบ โคแอกเชียล 75 โอห์ม และสายเคเบิลหลักจะต้องไม่มีการต่อแยกแขนงออกไป ในแบบที่ 2 ซึ่งเรียกกันว่าแบบเบสแบนด์นั้นจะมีความเร็ว 5-10 เมกะบิตต่อวินาที ใช้สายแบบเดียวกับแบบที่ 1 แต่สัญญาณภายในจะเข้ารหัสแบบ FSK และแบบที่ 3 หรือ แบบบอร์ดแบนด์ จะใช้สายทรังก์ ซึ่งสามารถใช้ได้กับความเร็ว 1 ,5 และ 10 เมกะบิตต่อวินาที ซึ่งสัญญาณภายในสายจะเป็นแบบ AM นั้นเอง

2.15 RF Module TRW 2.4 กิกะเฮริต์

ลักษณะของตัว RF Module TRW 2.4 กิกะเฮริต์ เป็น Module ตัวรับส่งข้อมูลผ่านคลื่นวิทยุระยะไกล ซึ่งมีความสามารถในการส่งข้อมูลที่รวดเร็ว มีเสถียรภาพสูงและขนาดเล็ก

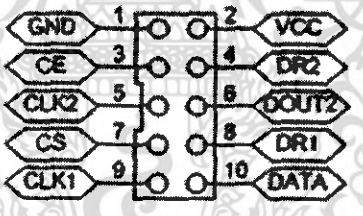
รูปที่ 2.18 แสดงลักษณะด้านบนของตัว RF Module TRW 2.4 กิกะเฮริต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.19 แสดงลักษณะด้านล่างของตัว RF Module TRW 2.4 กิกะเฮริต์

คุณสมบัติทั่วไป

- สามารถเป็นตัวรับและตัวส่งในตัวเดียวกัน
- ส่งได้ไกลถึง 280 เมตรในที่โล่ง
- ทำงานที่แรงดัน 1.9 โวลต์ - 3.6 โวลต์
- กินกระแส 10.5 มิลลิแอมป์ ขณะส่งข้อมูล และกินกระแส 18 มิลลิแอมป์ ขณะรับข้อมูล
- มีเสาอากาศในตัว
- ย่านความถี่ในการใช้งาน 2.40 กิกะเฮริต์ - 2.54 กิกะเฮริต์
- อัตราการส่งข้อมูล 250 กิโลบิตต่อวินาที - 1 เมกะบิตต่อวินาที
- ส่งได้ 125 channel



รูปที่ 2.20 แสดงลักษณะขาต่างๆของตัว RF Module TRW 2.4 กิกะเฮริต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.12 แสดงหน้าที่การทำงานของขา RF Module TRW 2.4 กิกะเฮริต์

ขาที่		หน้าที่
1	GND	ขากราวด์
2	VCC	ไฟเลี้ยง 1.9 โวลต์ – 3.6 โวลต์
3	CE	เป็นขาควบคุมการทำงานของการรับ (RX) และการส่ง (TX)
4	DR2	ขานี้เป็น "1" เมื่อได้รับข้อมูล Channel 2
5	CLK2	เป็นขาสัญญาณนาฬิกา Channel 2
6	DOUT2	ขารับข้อมูลและส่งข้อมูล Channel 2
7	CS	เป็นขาเซ็ตโหมดการทำงานของ TRW 2.4 กิกะเฮริต์
8	DR1	ขานี้เป็น "1" เมื่อได้รับข้อมูล Channel 1
9	CLK1	เป็นขาสัญญาณนาฬิกา Channel 2
10	DATA	ขารับข้อมูลและส่งข้อมูล Channel 1

โดยภายในตัว RF Module TRW 2.4 กิกะเฮริต์ จะมีการตรวจสอบบิตทั้งหมด 144 บิต โดยที่แต่ละบิตจะมีหน้าที่ในการสั่งงานให้ตัว Module ทำงานต่างๆ โดยจะช่วงบิตต่างในการสั่งงาน

2.15.1 การตั้งค่าการใช้งาน RF Module TRW 2.4 กิกะเฮริต์

บิตที่ 143 – 120

ตั้งเป็นค่าเริ่มต้นการใช้งานของตัว Module คือ 8E081C

บิตที่ 121 – 120 PLL_CTRL

จะทำหน้าที่เป็นตัวกำหนด Module ทำงานในโหมดของการเป็นตัวส่งอย่างเดียว หรือตัวรับอย่างเดียว หรือเป็นทั้งตัวรับและตัวส่ง

ตารางที่ 2.13 แสดงหน้าที่ของบิต 120 – 121

PLL CTRL		
D121	**D120**	**PLL**
0	0	Open Tx/Closed Rx
0	1	Open Tx/Open Rx
1	0	Closed Tx/closed Rx
1	1	Closed Tx/Open Rx

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บิตที่ 119 – 104 DATAx_W

โดยจะแบ่งเป็น 2 ส่วนด้วยกัน คือจะเป็นตัวกำหนดขนาดของจำนวนบิตในการรับ โดยจะแบ่งเป็น Channel 1 และ Channel 2 สามารถรับและส่งข้อมูลได้มากที่สุด 256 บิต

ตารางที่ 2.14 แสดงหน้าที่ของบิต 119 – 104

DATA2_W							
119	118	117	116	115	114	113	112

DATA1_W							
111	110	109	108	107	106	105	104

โดยที่จำนวน data ที่จะรับทั้งหมดจะมีค่าเท่ากับจำนวน

$$DATAx_W(bits) = 256 - ADDR_W - CRC$$

บิตที่ 103– 24 ADDRx

จำนวนบิตแอดเดรส ที่จะใช้ในการจองพื้นที่ในการรับข้อมูลของ Channel 1 และ Channel 2 โดยแต่ละ Channel จะมีจำนวนบิตทั้งหมด 40 บิต

ตารางที่ 2.15 แสดงหน้าที่ของบิต 103 – 24

ADDR2											
103	102	101	.....	71	70	69	68	67	66	65	64

ADDR1											
63	62	61	.....	31	30	29	28	27	26	25	24

บิตที่ 23–16 ADDR_W & CRC

ตารางที่ 2.16 แสดงหน้าที่ของบิต 23 – 16

ADDR_W						CRC_L	CRC_EN
23	22	21	20	19	18	17	16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบ่งการทำงานแต่ละบิตออกเป็น 3 ส่วน

1. บิต 23 – 18

ADDR_W กำหนดช่วงความถี่ที่จะใช้รับข้อมูลจากตัวส่งโดยที่ตัวส่งและตัวรับจะต้องกำหนดค่าความถี่ให้ตรงกัน

2. บิต 17

CRC_L : กำหนดค่าจำนวนบิตที่ส่งในโหมด Shock Burst

Logic 0 : 8 บิต CRC

Logic 1 : 16 บิต CRC

3. บิต 16

CRC_EN : ใช้ตรวจสอบและพิสูจน์ on-chip CRC

Logic 0 : On-chip CRC /checking disabled

Logic 1 : On-chip CRC /checking enabled

บิตที่ 15 – 8

ตารางที่ 2.17 แสดงหน้าที่ของบิต 15 – 8

RX2_EN	CM	RFDR_SB	XO_F			RF_PWR	
15	14	13	12	11	10	9	8

แบ่งการทำงานออกแต่ละบิตออกเป็น 5 ส่วน

1. บิต 15

RX2_EN

Logic 0 : กำหนดให้ Channel 1 ทำการรับข้อมูล

Logic 1 : กำหนดให้ Channel 2 ทำการรับข้อมูล

2. บิต 14

CM Communication Mode

Logic 1 : กำหนดให้เป็นการส่งข้อมูลแบบ ShockBurst

3. บิต 13

RF Data Rate

Logic 0 : กำหนดส่งข้อมูล 256 กิโลบิตต่อวินาที

Logic 0 : กำหนดส่งข้อมูล 1 เมกะบิตต่อวินาที

4. บิต 12 – 10

กำหนดเป็นค่าเริ่มต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.18 แสดงหน้าที่ของบิต 12 – 10

D12	D11	D10
0	1	1

5. บิต 9 – 8

RF_PWR : กำหนดค่ากำลังงานเพาเวอร์ในการส่งข้อมูล

ตารางที่ 2.19 แสดงหน้าที่ของบิต 9 – 8

RF OUTPUT POWER		
D9	D8	P [dBm]
0	0	-20
0	1	-10
1	0	-5
1	1	0

บิตที่ 7 – 0 RF Channel & direction

ตารางที่ 2.20 แสดงหน้าที่ของบิต 7 – 0

RF_CH#							RXEN
7	6	5	4	3	2	1	0

แบ่งการทำงานแต่ละบิตออกเป็น 2 ส่วน

1. บิต 7 - 1

RF_CH# : กำหนดค่าในการส่งจะใช้ความถี่เท่าใดโดยที่ความถี่อยู่ระหว่าง 2400 เมกะเฮริ์ต – 2527 เมกะเฮริ์ต

2. บิต 0

Set active mode

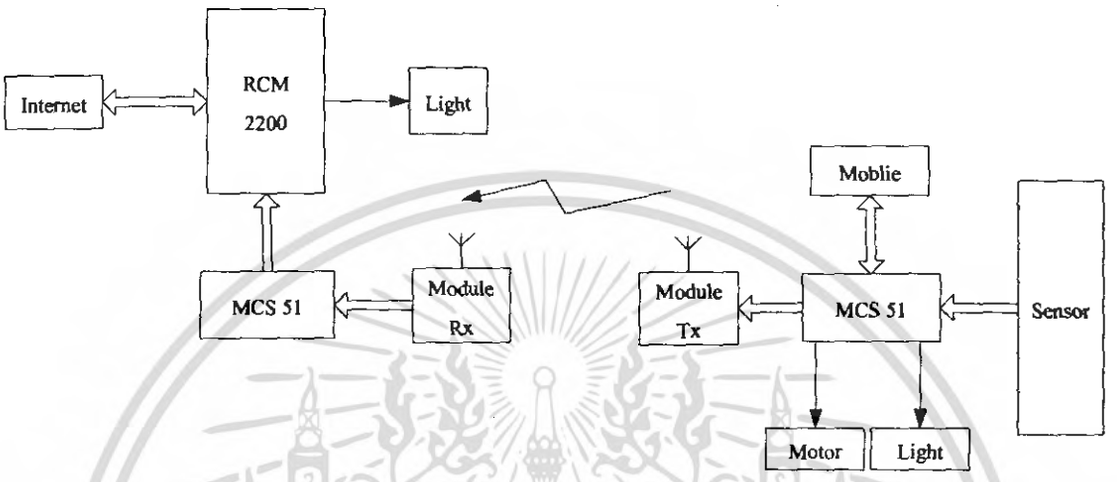
Logic 0 : เป็นการกำหนดให้ transmit module

Logic 1 : เป็นการกำหนดให้ receive module

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การคำนวณและการสร้าง

3.1 การออกแบบวงจรหลัก



รูปที่ 3.1 Block diagram

วงจรส่วนแจ้งเตือนภัย เป็นการเตือนภัยเมื่ออุปกรณ์ตรวจจับรับเหตุแล้วจะส่งสัญญาณไปที่ ชุดรับสัญญาณเพื่อทำการแจ้งเตือนมี 2 ขั้นตอนคือ จะทำการโทรศัพท์แจ้งเตือนก่อนเมื่อโทรเสร็จแล้ว ก็จะทำการส่งข้อความสั้น เพื่อแจ้งให้ทราบว่ามีเหตุผิดปกติขึ้นที่จุดใด

วงจรส่วนแจ้งเตือนภัยด้วยการโทรศัพท์และส่งข้อความสั้น ในโครงงานนี้จะใช้ MCS - 51 ในการทำการติดต่อกับโทรศัพท์เคลื่อนที่ ยี่ห้อซีเมนต์รุ่น C35i โดยจะส่งข้อความสั้นเมื่อระบบมีการเปิดและเกิดเหตุผิดปกติขึ้นภายในบ้าน เช่น มีการเปิดประตู หน้าต่างหรือเกิดควันขึ้นภายในบ้านและจะทำการส่งสัญญาณผ่านโมดูลไร้สายเพื่อนำสัญญาณไปให้ Rabbit Core Module RCM 2200 เพื่อตรวจสอบสถานะที่หน้าเว็บเพจ

3.2 อุปกรณ์ตรวจจับการเปิด/ปิด

ในส่วนของอุปกรณ์ตรวจจับนี้จะใช้สวิทช์แม่เหล็ก (Magnetic Switch) จากรูปที่ 3.2 รูปร่างของสวิทช์แม่เหล็กที่นำมาใช้เป็นตัวตรวจจับเซ็นเซอร์ รูปที่ 3.2 (ก) รูปร่างของสวิทช์แม่เหล็กแบบปกติปิด (NC) รูปที่ 3.2 (ข) รูปร่างของสวิทช์แม่เหล็กแบบปกติเปิด (NO)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

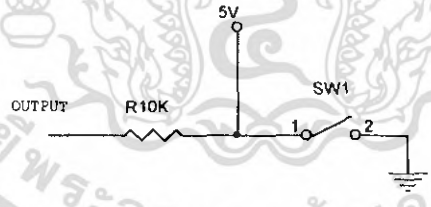
(ก) (ข)

รูปที่ 3.2 รูปร่างของสวิทช์แม่เหล็ก

จากรูปที่ 3.2 เป็นโครงสร้างของสวิทช์แม่เหล็กที่นำมาใช้งาน ส่วนรูปที่ 3.3 (ก) เป็นโครงสร้างของสวิทช์แบบปกติปิดหน้าสัมผัสของสวิทช์ต่อกันอยู่ และรูปที่ 3.3 (ข) เป็นโครงสร้างของสวิทช์แบบปกติเปิดหน้าสัมผัสของสวิทช์จะจากกัน

รูปที่ 3.3 โครงสร้างของสวิทช์แม่เหล็ก

การออกแบบวงจรเซ็นเซอร์ จะใช้สวิทช์แม่เหล็กเป็นตัวตรวจจับสิ่งผิดปกติที่เกิดขึ้นโดยสวิทช์จะยึดติดอยู่กับขอบประตู

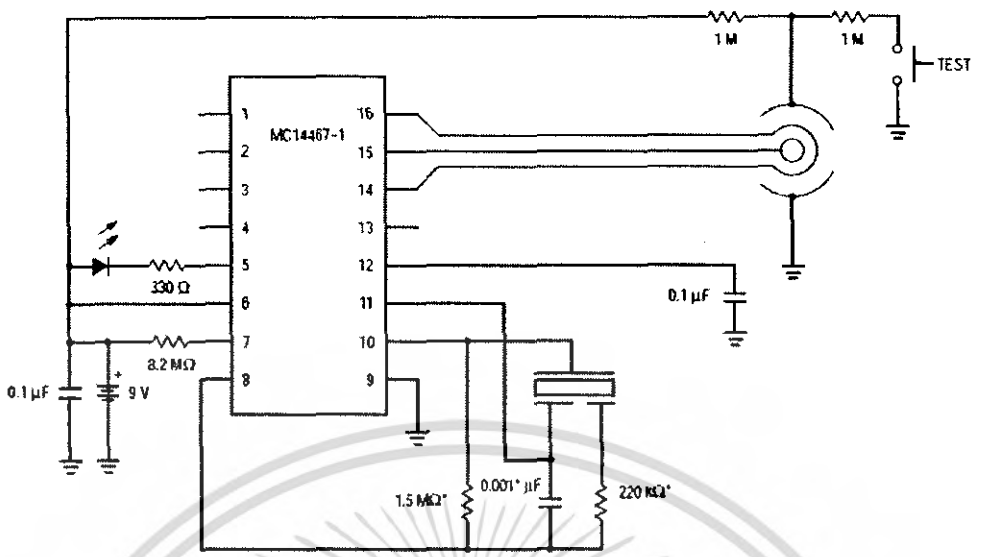


รูปที่ 3.4 วงจรเซ็นเซอร์

3.3 อุปกรณ์ตรวจจับควัน

อุปกรณ์ตรวจจับควันโดยอาศัยหลักการเกิดไอออน (Ionization Smoke Detection) ส่วนใหญ่จะเป็นอุปกรณ์ชนิดจุด ประกอบด้วยสารกัมมันตภาพรังสีบรรจุอยู่ในกล่องตรวจจับ เป็นผลให้อากาศภายในกล่องเกิดไอออนมีสภาพเป็นตัวนำและทำให้กระแสไฟฟ้าไหลผ่านได้ระหว่างขั้วอิเล็กโทรด เมื่ออนุภาคควันผ่านเข้ากล่อง อนุภาคควันจะจับตัวรวมกับไอออนทำให้การเคลื่อนที่ช้าลงและค่าความนำไฟฟ้าลดลง อุปกรณ์ตรวจจับจะเริ่มสัญญาณเมื่อค่าความนำลดลงถึงจุดพิกัดที่กำหนดไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

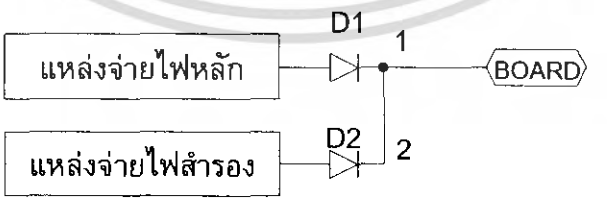


รูปที่ 3.5 วงจรตรวจสอบควัน

3.4 แหล่งจ่ายไฟหลักและไฟสำรอง

ในส่วนของระบบไฟสำรองนี้ ทำหน้าที่ในการตัดต่อระบบไฟหลักในกรณีไฟฟ้าดับจะทำการตัดต่อระบบไฟให้มาใช้ในส่วนของระบบไฟสำรองซึ่งในการตัดต่อวงจรในโครงงานนี้ใช้ไดโอดในการตัดต่อระบบซึ่งแสดงหลักการของระบบไฟสำรองได้ดังรูปที่ 3.6

จากรูปที่ 3.6 ใช้คุณสมบัติการนำกระแสของไดโอด คือเมื่อแรงดันไฟหลักทำงานทำให้ไดโอด D1 นำกระแส ซึ่งมีแรงดันตกคร่อมตัวมัน 0.7 โวลต์หรือตามชนิดของไดโอด ซึ่งการที่ทำให้ไดโอด D1 จะนำกระแสนั้นขึ้นอยู่กับศักดาไฟฟ้าที่จุด 1 ต้องมากกว่าศักดาไฟฟ้าที่จุด 2 เมื่อศักดาไฟฟ้าจุดที่ 1 มากกว่า ทำให้ไดโอด D2 ไบอัสกลับ จึงทำให้กระแสจากแหล่งจ่ายไฟสำรองไม่ไหลผ่านออกมา แต่เมื่อเกิดกรณีไฟฟ้าดับทำให้แหล่งจ่ายไฟหลักไม่สามารถจ่ายกระแสออกมาได้เป็นผลให้ศักดาที่จุด 1 ไม่มีจึงทำให้ไดโอด D2 สามารถนำกระแสได้ (Forward Bias) ซึ่งทำให้ระบบยังสามารถทำงานได้ถึงแม้จะเกิดเหตุไฟฟ้าดับ

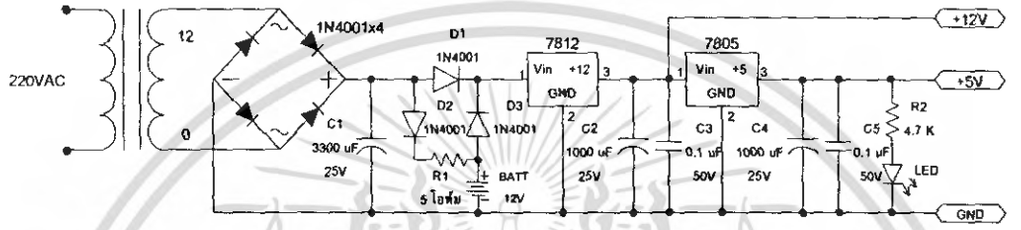


รูปที่ 3.6 หลักการของระบบจ่ายไฟหลักและไฟสำรอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการออกแบบวงจรแหล่งจ่ายไฟหลักและไฟสำรอง เลือกใช้ไดโอด IN4001 ชนิดซิลิกอน โดยมีจุดเบรกดาวน์ที่ 50 โวลต์ 1 แอมป์ สามารถอธิบายหลักการทำงานดังนี้

เมื่อเริ่มเปิดสวิทช์วงจรเริ่มทำงานไฟฟ้ากระแสสลับ 220 โวลต์เอซีไหลผ่านวงจรบริดจ์เรกติฟายเออร์ทำการแปลงแรงดันไฟฟ้ากระแสสลับไปเป็นไฟฟ้ากระแสตรงและผ่าน C1 ทำการกรองแรงดัน ไฟฟ้ากระแสตรงให้เรียบยิ่งขึ้น ส่วนไดโอด D1 ทำหน้าที่ให้กระแสไฟฟ้าไหลผ่านทางเดียว ไม่ให้ไหลย้อนกลับ โดยมีไดโอด D2 กับตัวต้านทาน R1 ทำหน้าที่เติมประจุให้กับแบตเตอรี่ทีละน้อยๆ ไดโอด D1 นำกระแสเมื่อไฟฟ้ามาปกติทำให้ D3 ไม่นำกระแส แบตเตอรี่จึงไม่สามารถจ่ายกระแสได้



รูปที่ 3.7 วงจรแหล่งจ่ายไฟหลักและไฟสำรอง

โดยเอาท์พุทจะถูกกำหนดให้มีแรงดันคงที่ด้วยไอซีรักษาระดับแรงดัน (Voltage Regulator) แรงดันไฟ 12 โวลต์เลือกใช้ไอซีเบอร์ LM7812 และมี C2 เป็นตัวกรองแรงดันให้เรียบยิ่งขึ้น ส่วน C3 เป็นตัวบายพาสสัญญาณรบกวนความถี่สูงลงกราวด์ และในส่วนของไฟ 5โวลต์ใช้ไอซีเบอร์ LM7805 และมี C4 เป็นตัวกรองแรงดันให้เรียบยิ่งขึ้น ส่วน C5 เป็นตัวบายพาสสัญญาณรบกวนความถี่สูงลงกราวด์ ส่วน R2 ทำหน้าที่จำกัดกระแสที่จะไบอัสให้แก่หลอดแอลอีดีเพื่อเป็นตัวบอกสถานะว่าขณะนี้มีไฟเลี้ยงวงจร ดังนั้นจะเห็นว่าเมื่อไฟฟ้าดับระบบรักษาความปลอดภัยก็ยังสามารถทำงานได้ตามปกติ สามารถแสดงวงจรแหล่งจ่ายไฟหลักและไฟสำรองได้ดังรูปที่ 3.7

3.5 ไอซี TA7279P

ภายในไอซีจะประกอบด้วยวงจร H-Bridge SW จำนวน 2 ชุด (สามารถควบคุมมอเตอร์ได้ 2 ตัว) แต่ละชุดมีคุณสมบัติดังนี้

- รับแรงดันได้ตั้งแต่ 0โวลต์ –18 โวลต์
- เอาท์พุทจ่ายกระแสได้ถึง 1.0 แอมป์ (AVE) 3.0 แอมป์ (Peak)

นอกจากนั้นภายในยังมีฟังก์ชันหยุดการทำงานเมื่ออุณหภูมิสูงเกินไปและยังมีวงจรจำกัดกระแสอีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

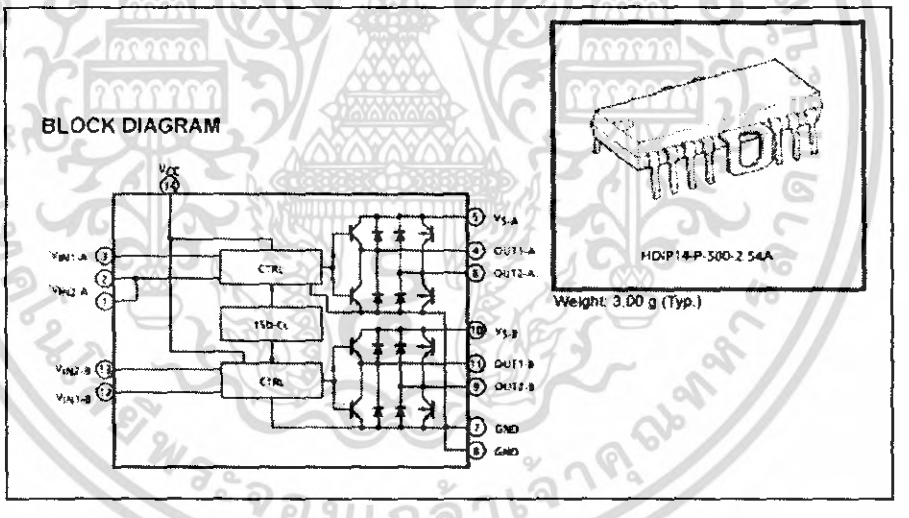
การนำไปใช้งาน

การนำไอซี TA7279P ไปใช้งานนั้นไม่จำเป็นต้องต่ออุปกรณ์มากมาย คือเราจะต่อไฟเลี้ยงให้กับตัวไอซีที่ 12 โวลต์ แล้วเราก็จะทำการป้อนอินพุท P3.2 , P3.3 , P3.4 , P3.5 ให้กับตัวไอซี TA7279P เพื่อควบคุมการหมุนดังนี้

ตารางที่ 3.1 แสดงสภาวะและการทำงานของมอเตอร์

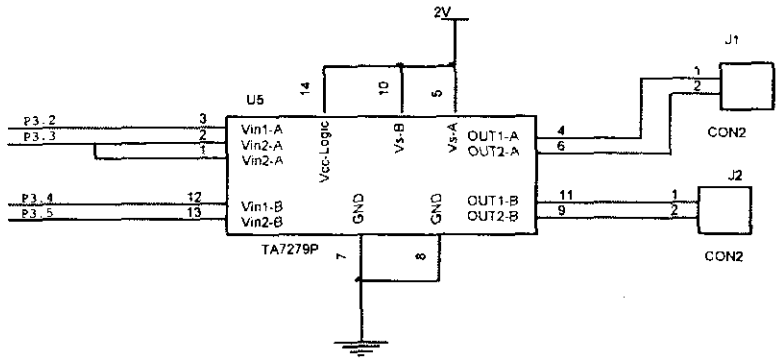
อินพุท	เอาท์พุท
00	มอเตอร์ไม่หมุน
01	มอเตอร์หมุนซ้าย
10	มอเตอร์หมุนขวา
11	มอเตอร์ไม่หมุน

จากวงจรรวมนั้นเราจะใช้ควบคุมมอเตอร์ตัวเดียวซึ่งไอซี TA7279P นั้นจะสามารถควบคุมมอเตอร์ได้ 2 ตัวโดยอินพุทจะมี 2 ชุดคือขา 2 , 3 และขา 12 , 13 ส่วนเอาท์พุทก็จะมี 2 ชุดเช่นกันคือ ขา 4 , 6 , 9และขา 11



รูปที่ 3.8 ขาของไอซี TA7279P และโครงสร้างภายใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

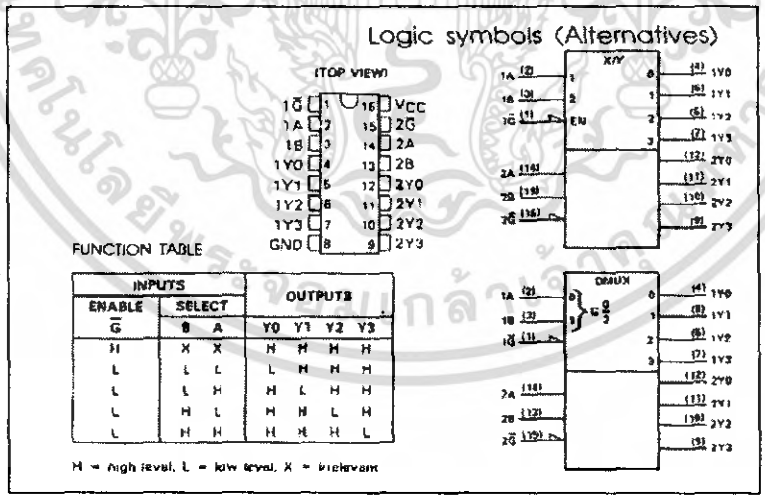


รูปที่ 3.9 การต่อไอซี TA7279P เพื่อใช้งานในวงจรรวม

เนื่องจากไอซี TA7279P นี้ใช้งานง่ายแต่เราควรระวังเรื่องของเอาท์พุทของตัวไอซี เนื่องจากไอซีชนิดนี้เป็นไอซีที่ใช้ขับกระแสของมอเตอร์ ดังนั้นห้ามต่อเอาท์พุทช็อตกันเด็ดขาดเพราะอาจจะทำให้ไอซีพังได้ ส่วนในวงจรขนาดใหญ่นั้นควรจะมีตัวคาปาซิเตอร์ต่อบายพาสเพื่อขับกระแสออกเอาท์พุทเพราะในวงจรใหญ่จะมีโหลดอยู่มากมายทำให้กระแสที่ได้นั้นไม่เพียงพอสำหรับขับมอเตอร์ได้

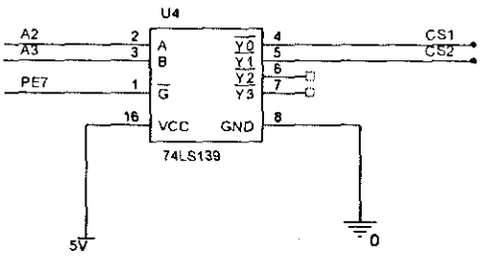
3.6 การนำไอซี เบอร์ 74LS139 ไปใช้งาน

การออกแบบวงจรถอดรหัสโดยใช้ไอซี 74LS139 เริ่มต้นโดยการต่อสัญญาณ A2,A3 จาก RCM2200 Rabbit Core Module ไปยังขา 2 , 3 ไอซี 74LS139 ซึ่ง A2 , A3 เป็นขาแอดเดรสที่ส่งมาจาก RCM2200 Rabbit Core Module ต่อขา PE เข้ายังขา 1 ของ 74LS139 ซึ่งขา PE7 ทำหน้าที่เป็นตัว Enable ให้กับไอซี 8255 ทำงานโดยกำหนดวางแอดเดรสไว้ที่ E000-E007



รูปที่ 3.10 ตำแหน่งขาของไอซี 74LS139 และโครงสร้างภายใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11 การต่อไอซี 74LS139 เพื่อใช้งาน

3.7 รีเลย์

รีเลย์ เป็นอุปกรณ์ทำหน้าที่เป็นสวิทช์มีหลักการทำงานคล้ายกับขดลวดแม่เหล็กไฟฟ้าหรือโซลินอยด์รีเลย์ใช้ในการควบคุมวงจรไฟฟ้าได้อย่างหลากหลาย รีเลย์เป็นสวิทช์ควบคุมที่ทำงานด้วยไฟฟ้า แบ่งออกตามลักษณะการใช้งานได้เป็น 2 ประเภทคือ

1. รีเลย์กำลัง (Power relay) หรือมักเรียกกันว่าคอนแทกเตอร์ (Contactor or Magneticcontactor) ใช้ในการควบคุมไฟฟ้ากำลัง มีขนาดใหญ่กว่ารีเลย์ธรรมดา

2. รีเลย์ควบคุม (Control Relay) มีขนาดเล็กกำลังไฟฟ้าต่ำ ใช้ในวงจรควบคุมทั่วไปที่มีกำลังไฟฟ้าไม่มากนัก หรือเพื่อการควบคุมรีเลย์หรือคอนแทกเตอร์ขนาดใหญ่ รีเลย์ควบคุมบางทีเรียกกันง่าย ๆ ว่า "รีเลย์"

ชนิดของรีเลย์แบ่งตามลักษณะของคอยล์หรือแบ่งตามลักษณะการใช้งานได้แก่รีเลย์ดังต่อไปนี้

1. รีเลย์กระแส (Current relay) คือรีเลย์ที่ทำงานโดยใช้กระแสมีทั้งชนิดกระแสขาด (Under - current) และกระแสเกิน (Over - current)

2. รีเลย์แรงดัน (Voltage relay) คือรีเลย์ที่ทำงานโดยใช้แรงดันมีทั้งชนิดแรงดันขาด (Under - voltage) และ แรงดันเกิน (Over - voltage)

3. รีเลย์ช่วย (Auxiliary relay) คือรีเลย์ที่เวลาใช้งานจะต้องประกอบเข้ากับรีเลย์ชนิดอื่นจึงจะทำงานได้

4. รีเลย์กำลัง (Power relay) คือรีเลย์ที่รวมเอาคุณสมบัติของรีเลย์กระแสและรีเลย์แรงดันเข้าด้วยกัน

5. รีเลย์เวลา (Time relay) คือรีเลย์ที่ทำงานโดยมีเวลาเข้ามาเกี่ยวข้องด้วยมีอยู่ด้วยกัน 4 แบบ

- รีเลย์กระแสเกินชนิดเวลาผกผันกับกระแส (Inverse time over current relay) คือ รีเลย์ที่มีเวลาทำงานเป็นส่วนกลับกับกระแส

- รีเลย์กระแสเกินชนิดทำงานทันที (Instantaneous over current relay) คือรีเลย์ที่ทำงานทันทีทันใดเมื่อมีกระแสไหลผ่านเกินกว่าที่กำหนดที่ตั้งไว้

- รีเลย์แบบดิฟฟินิตไทม์เล็ก คือรีเลย์ที่มีเวลาการทำงานไม่ขึ้นอยู่กับความมากน้อยของกระแสหรือค่าไฟฟ้าอื่นๆ ที่ทำให้เกิดงานขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- รีเลย์แบบอินเวอสดิฟฟินิตมินิมั่มไทม์เล็ก คือรีเลย์ ที่ทำงานโดยรวมเอาคุณสมบัติของเวลาผกผันกับกระแส (Inverse time) และ แบบดิฟฟินิตไทม์แล็กเข้าด้วยกัน

6. รีเลย์กระแสต่าง (Differential relay) คือรีเลย์ที่ทำงานโดยอาศัยผลต่างของกระแส

7. รีเลย์มีทิศ (Directional relay) คือรีเลย์ที่ทำงานเมื่อมีกระแสไหลผิดทิศทาง มีแบบรีเลย์กำลังมีทิศ (Directional power relay) และรีเลย์กระแสมีทิศ (Directional current relay)

8. รีเลย์ระยะทาง (Distance relay) คือรีเลย์ระยะทางมีแบบต่างๆ ดังนี้

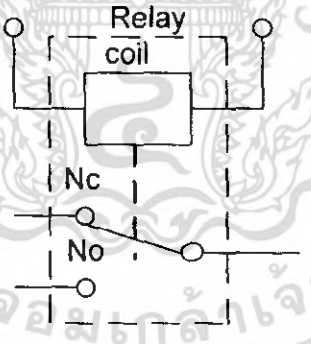
- รีแอกแตนซ์รีเลย์
- อิมพีแดนซ์รีเลย์
- โมห์รีเลย์
- โอห์มรีเลย์
- โพลาไรซ์โมห์รีเลย์
- ออฟเซทโมห์รีเลย์

9. รีเลย์อุณหภูมิ (Temperature relay) คือรีเลย์ที่ทำงานตามอุณหภูมิที่ตั้งไว้

10. รีเลย์ความถี่ (Frequency relay) คือรีเลย์ที่ทำงานเมื่อความถี่ของระบบต่ำกว่าหรือมากกว่าที่ตั้งไว้

11. บูคโฮลซ์รีเลย์ คือรีเลย์ที่ทำงานด้วยก๊าซใช้กับหม้อแปลงที่แช่อยู่ในน้ำมันเมื่อเกิดฟอลต์ขึ้นภายในหม้อแปลง จะทำให้น้ำมันแตกตัวและเกิดก๊าซขึ้นภายในไปดันหน้าสัมผัสให้รีเลย์ทำงาน

ซึ่งในโครงงานนี้จะใช้รีเลย์ควบคุม เพื่อทำการควบคุมการปิด/เปิดสวิทช์หลอดไฟฟ้า ซึ่งสั่งงานมาจากโทรศัพท์รวมไปถึงทางหน้าเว็บเพจ

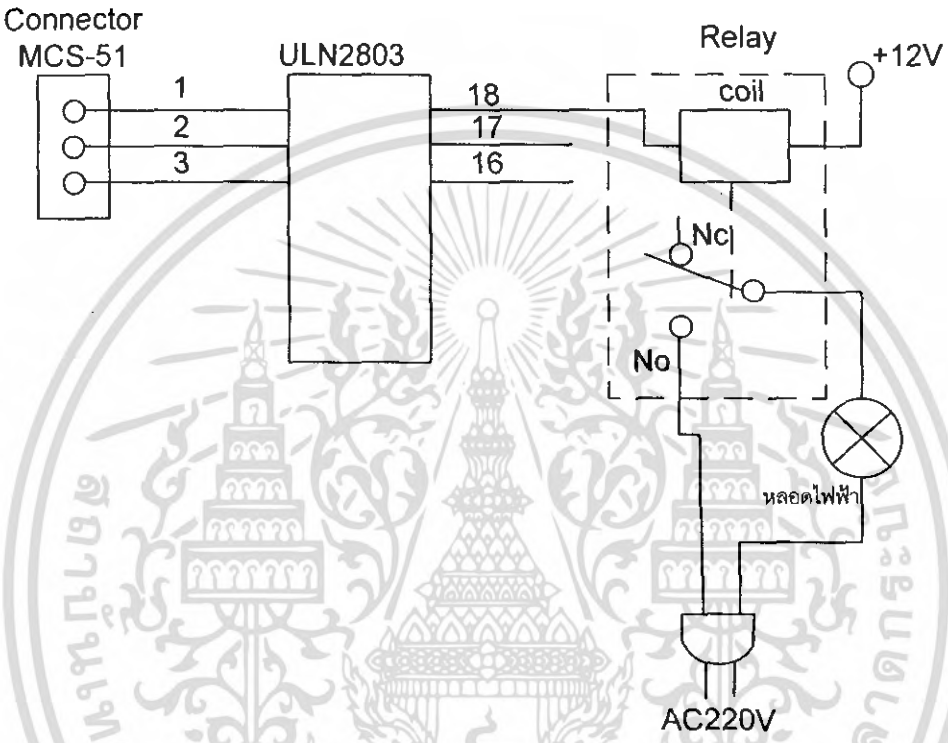


รูปที่ 3.12 แสดงสัญลักษณ์ของรีเลย์ที่ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.12 เป็นการแสดงสัญลักษณ์ของรีเลย์ จากรูปจะประกอบไปด้วย coil สถานะของรีเลย์ปกติปิดและสถานะของรีเลย์ปกติเปิด

การทำงานของรีเลย์ โดยรีเลย์จะอาศัยหลักการเหนี่ยวนำของแม่เหล็กซึ่งเมื่อจ่ายแรงดันให้กับ coil ของรีเลย์ ก็จะทำให้เกิดการสวิทช์จากสถานะปกติปิดก็จะเปิดและเช่นเดียวกันสถานะปกติเปิดก็จะปิด จากสถานการณ์ทำงานข้างต้นที่ได้กล่าวมาแล้วเราก็สามารถทำการควบคุมรีเลย์ให้ทำการเปิดไฟฟ้าหรือปิดไฟฟ้า โดยการควบคุมแรงดันไฟฟ้าที่ผ่าน coil ได้



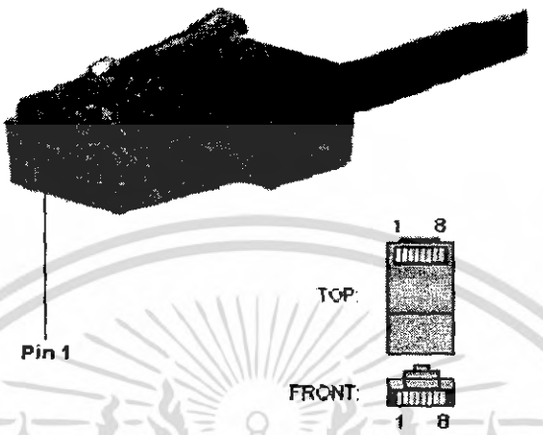
รูปที่ 3.13 แสดงการใช้งานรีเลย์ในการควบคุมหลอดไฟฟ้า

จากรูปที่ 3.13 แสดงการใช้งานรีเลย์โดยจะควบคุมการเปิดและปิดหลอดไฟฟ้าจากไมโครคอนโทรลเลอร์ โดยสัญญาณควบคุมจะส่งผ่านไปยังอินพุทของไอซี ULN2803 ซึ่งเป็นไอซีขับแบบอินเวอร์เตอร์ที่สามารถจ่ายกระแสได้สูงสุด 500 มิลลิแอมป์ และภายในตัวไอซีจะมีไดโอดป้องกันการยุบตัวของสนามแม่เหล็กที่ coil ของรีเลย์ เราสามารถนำเอาสัญญาณเอาท์พุทจากไอซี ULN2803 ไปทำการขับรีเลย์ได้เลย โดยที่อีกขั้วหนึ่งของรีเลย์จะต่อกับแรงดันไฟ 12 โวลต์ เมื่อรีเลย์ทำงานก็จะทำให้วงจรของหลอดไฟฟ้าครบวงจรหลอดไฟฟ้าก็จะติดและเช่นเดียวกัน ถ้าหากรีเลย์ยังไม่ทำงาน วงจรของหลอดไฟฟ้าก็ยังไม่ครบวงจรหลอดไฟฟ้าก็จะดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.8 สาย UTP (Unshieled Twisted Pair)

ลักษณะของสายที่ใช้กับ RJ-45 เป็นแบบ 8 เส้น โดยตีเกลียวกัน 4 คู่ โดยไม่มีการชีลด์ เรียกว่า UTP เป็นสายมาตรฐานที่ใช้กันในปัจจุบันคือสายแบบ CAT-5



รูปที่ 3.14 แสดงลักษณะของสาย UTP

ตารางที่ 3.2 แสดงหน้าที่ของสาย UTP

การกำหนดหน้าที่ของสายย่อยภายในสายแบบ UTP	
หมายเลขสาย	หน้าที่ของสายสัญญาณ
1	Output Transmit Data +
2	Output Transmit Data -
3	Input Receive Data +
6	Input Receive Data -
4, 5, 7, 8	Reserved for other use

การเข้าหัวสายกับขั้วต่อ มี 2 แบบ คือ

1. การต่อเชื่อมระหว่างคอมพิวเตอร์ กับ RCM 2200 เป็นการต่อเชื่อมแบบ Peer to Peer 2 เครื่อง ไม่ต้องใช้ HUB การต่อสายต้องมีการสลับสายนำสัญญาณทั้งสองด้าน ดังตาราง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.3 แสดงการเชื่อมต่อระหว่างคอมพิวเตอร์ กับ Rabbit core Module RCM 2200

Computer	ขาที่	RCM 2200
ขาวส้ม	1	ขาวเขียว
ส้ม	2	
ขาวเขียว	3	ขาวส้ม
น้ำเงิน	4	น้ำเงิน
ขาวน้ำเงิน	5	ขาวน้ำเงิน
	6	ส้ม
ขาวน้ำตาล	7	ขาวน้ำตาล
น้ำตาล	8	น้ำตาล

2. การต่อเชื่อมระหว่าง HUB กับ RCM 2200 จะมีการเข้าสายกับขั้ว RJ-45 จะเหมือนกันทั้งสองข้าง

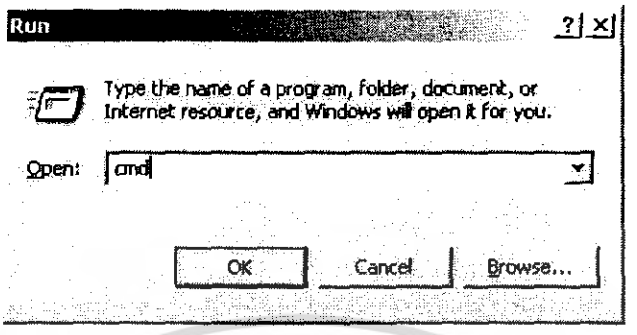
ตารางที่ 3.4 แสดงการต่อเชื่อมระหว่าง HUB กับ Rabbit core Module RCM 2200

ขาวส้ม	1	ขาวส้ม
ส้ม	2	ส้ม
ขาวเขียว	3	ขาวเขียว
น้ำเงิน	4	น้ำเงิน
ขาวน้ำเงิน	5	ขาวน้ำเงิน
เขียว	6	เขียว
ขาวน้ำตาล	7	ขาวน้ำตาล
น้ำตาล	8	น้ำตาล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.9 วิธีหา IP Address : Windows 2000/XP

1. ให้พิมพ์คำว่า "cmd" แล้วคลิกที่ปุ่ม [OK]



รูปที่ 3.15 แสดงการหาค่า IP

2. จากนั้นให้พิมพ์คำว่า "ip config" ณ ตำแหน่งที่ cursor กระพริบอยู่แล้วกด Enter จะปรากฏเลข IP Address ของเครื่องดังภาพ

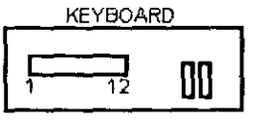
รูปที่ 3.16 แสดงค่า IP

3.10 หลักการอินเตอร์เฟสโทรศัพท์เคลื่อนที่

ในที่นี้จะกล่าวถึงแต่เฉพาะโทรศัพท์เคลื่อนที่ยี่ห้อ ซีเมนต์ รุ่น C35i ในส่วนของการติดต่อกันระหว่างโทรศัพท์เคลื่อนที่กับไมโครคอนโทรเลอร์นั้น เราใช้สายส่งข้อมูล (Data Link) ในการส่งข้อมูลระหว่างกันโดยสัญญาณของโทรศัพท์เคลื่อนที่และไมโครคอนโทรเลอร์นั้น เป็นสัญญาณระดับ TTL ทั้งคู่ ดังนั้นจึงไม่ต้องทำการแปลงสัญญาณใดๆทั้งสิ้น ในการใช้งานจริงเราจะใช้แค่ 5 ขา คือ ขา 1 , 4 , 5 , 6 และ 11 ของพอร์ตข้อมูลของโทรศัพท์เคลื่อนที่ยี่ห้อซีเมนต์รุ่น C35i ดังตารางที่3.5 รูปแบบข้อมูลของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โทรศัพท์เคลื่อนที่ซีเมนต์รุ่น C35i ส่งข้อมูลในลักษณะ 19200-N-8-1 คือมีอัตราการส่งข้อมูล 19200บิต ต่อวินาที ไม่มีพาริตี้บิต , ข้อมูลมีขนาด 8 บิต และมีบิตหยุด 1 บิต ตัวอย่างรหัสคำสั่งที่ใช้มีดังนี้



รูปที่ 3.17 แสดงจุดเชื่อมต่อ

ตารางที่ 3.5 พอร์ตของโทรศัพท์เคลื่อนที่ยี่ห้อซีเมนต์รุ่น C35i

Pin	Name	Function	in/out
1	GND	Ground	
2	SELF-SERVICE	Recognition/control battery charger	in/out
3	LOAD	Charging voltage	in
4	BATTERY	Battery	out
5	DATA OUT	Data sent	out
6	DATA IN	Data received	in
7	Z_CLK	Recognition/control accessories	
8	Z_DATA	Recognition/control accessories	
9	MICG	Ground for microphone	in
10	MIC	Microphone input	
11	AUD	Loudspeaker	out
12	AUDG	Ground for external speaker	

3.11 ขั้นตอนการออกแบบ Web Server

ขั้นตอนการออกแบบ web server บน Rabbit Core Module RCM 2200 เพื่อใช้ในโครงงาน จะต้องมีการกำหนดค่าต่างๆในโปรแกรมดังนี้

1. ไฟล์ HTML โดยมีการประกาศตัวแปรในไฟล์ HTML เพื่อรับค่าตัวแปรจากโปรแกรมหลัก โดยมีรูปแบบการประกาศตัวแปรในไฟล์ HTML ดังนี้

```
<html>
<img src="<!--#echo var="door1"-->">
</html>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการกำหนดค่าข้างต้นจะทำให้ไฟล์ HTML สามารถปรับเปลี่ยนรูปภาพ เพื่อใช้ในการแสดงผลของ เซ็นเซอร์ โดยใช้ไฟล์ภาพในการแสดงสถานะแทน โดยรูปแบบในการแสดงผลจะมีดังนี้

2. ทำการ โยงค่าตัวแปร โปรแกรมภายในโปรแกรมหลักให้บราวด์เซอร์สามารถใช้ค่าตัวแปรเดียวกับในโปรแกรมหลักได้ โดยมีรูปแบบการ โยงค่าตัวแปร โดยใช้ฟังก์ชัน HTTPSPEC_ VARIABLE โดยใช้รูปแบบคำสั่งดังนี้

{ HTTPSPEC_VARIABLE, "door1", 0, door1, PTR16, "%s", NULL},

หลังจากมีการเรียกใช้งานฟังก์ชัน HTTPSPEC_VARIABLE โปรแกรมหลักจะสามารถทำการส่งค่าตัวแปร door1 ไปยัง HTTP SERVER ได้

3. ทำการสร้าง link ในไฟล์ HTML เพื่อให้ไฟล์ HTML สามารถทำการเรียก ฟังก์ชัน CGI ที่เราเขียนรอเอาไว้ในโปรแกรมหลักได้โดยมีรูปแบบการเขียน Code HTML ดังนี้

```
<html>
<TD> <A HREF="/led1toggle.cgi"> <img SRC="button.gif"> </A> </TD>
</html>
```

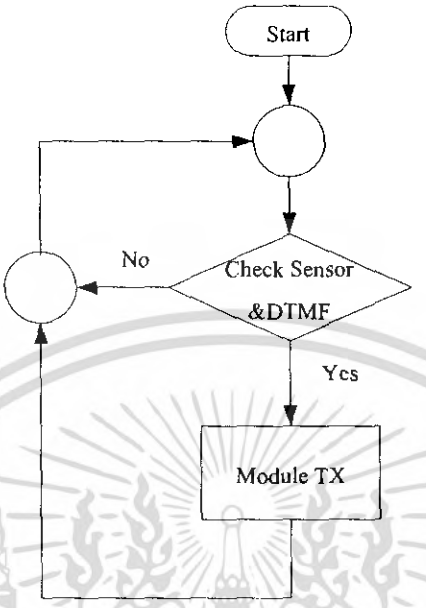
4. ทำการเรียกฟังก์ชัน SSPEC_FILE เพื่อระบุไฟล์ต่างๆที่ ต้องการใช้บน HTTP SERVER โดยมีรูปแบบการใช้งานดังนี้

```
{ HTTPSPEC_FILE, "/websun.shtml", webs_shtml, NULL, 0, NULL, &admin},
{ HTTPSPEC_FILE, "/ledon.gif", ledon_gif, NULL, 0, NULL, NULL},
{ HTTPSPEC_FILE, "/ledoff.gif", ledoff_gif, NULL, 0, NULL, NULL},
```

หลังจากมีการเรียกใช้ฟังก์ชันจะทำให้ HTTP SERVER สามารถระบุไฟล์ที่มีอยู่ในระบบได้อย่างถูกต้อง

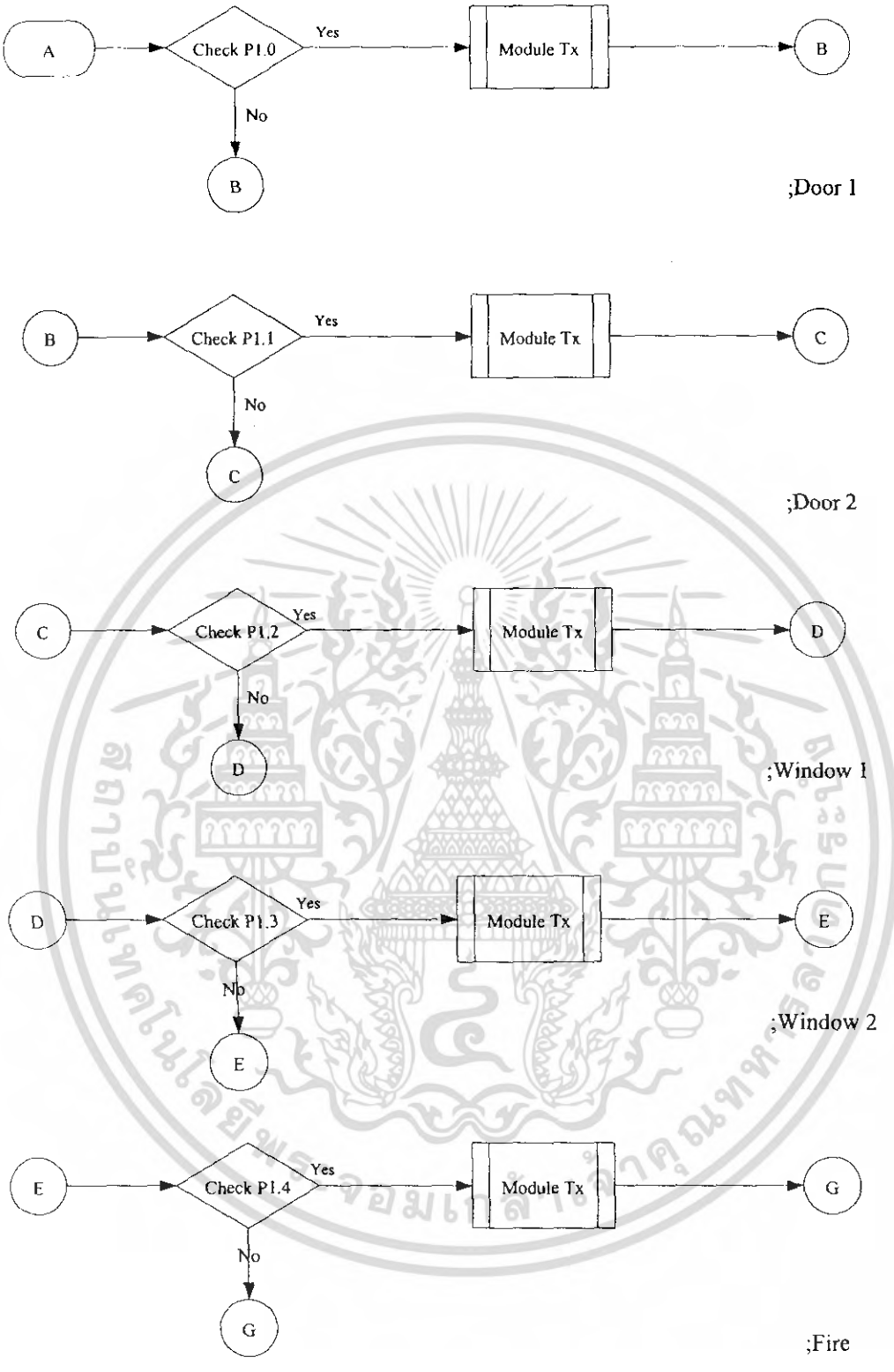
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.12 โฟลว์ชาร์ตการทำงานหลัก MCS – 51 ควบคุม RF Module Tx



รูปที่ 3.18 โฟลว์ชาร์ตการทำงานของโปรแกรมหลักควบคุม Module Tx

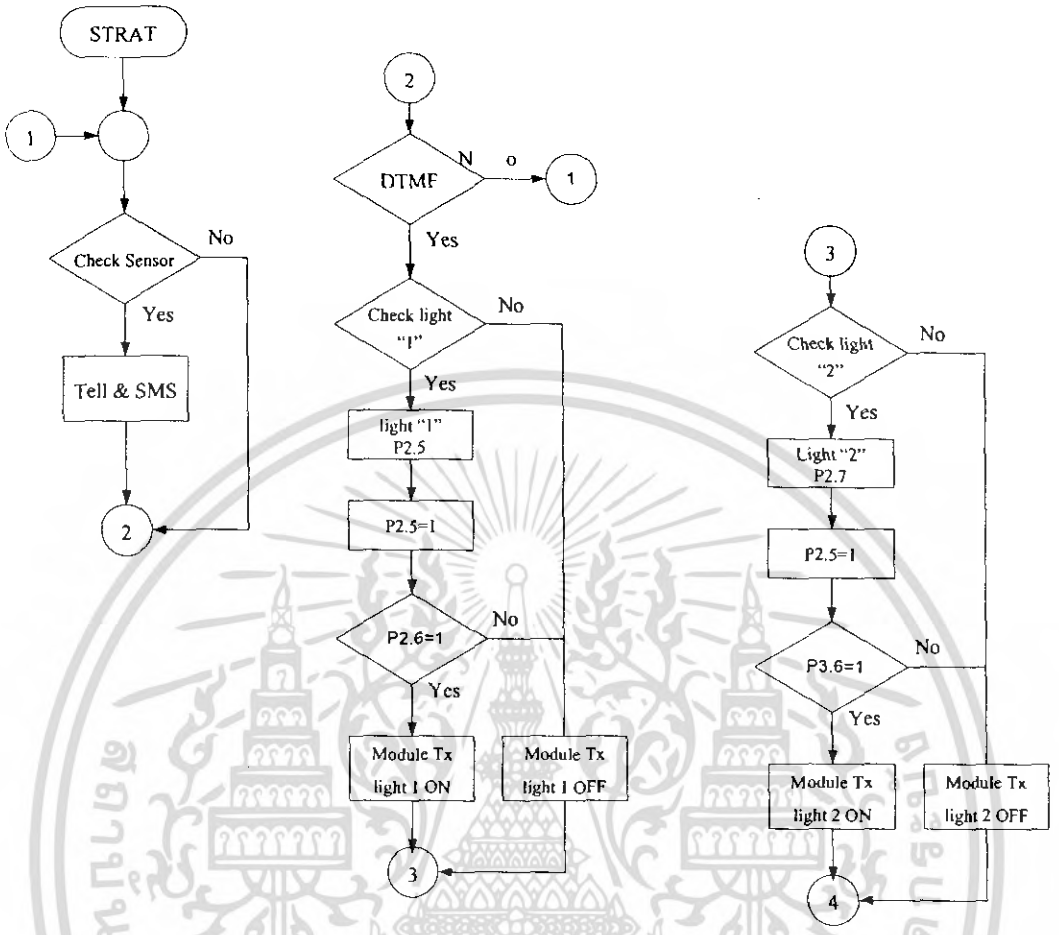
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.19 โฟลว์ชาร์ตการทำงานของแต่ละส่วน

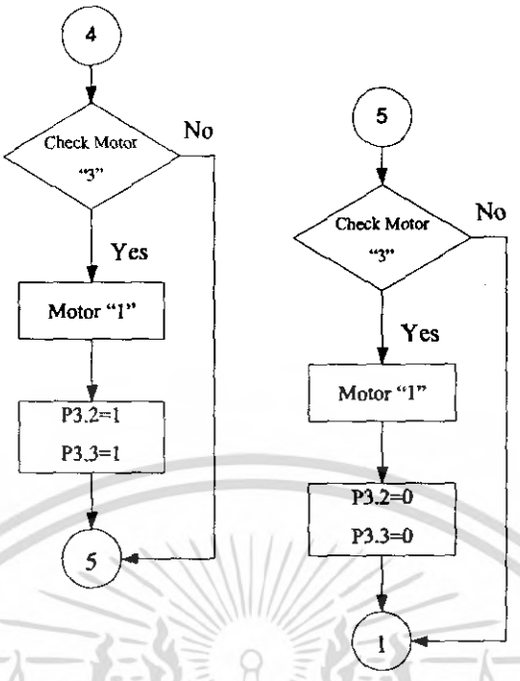
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.13 โฟลว์ชาร์ตการทำงานหลัก MCS – 51 ควบคุมโทรศัพท์เคลื่อนที่



รูปที่ 3.20 โฟลว์ชาร์ตการทำงานของโทรศัพท์เคลื่อนที่

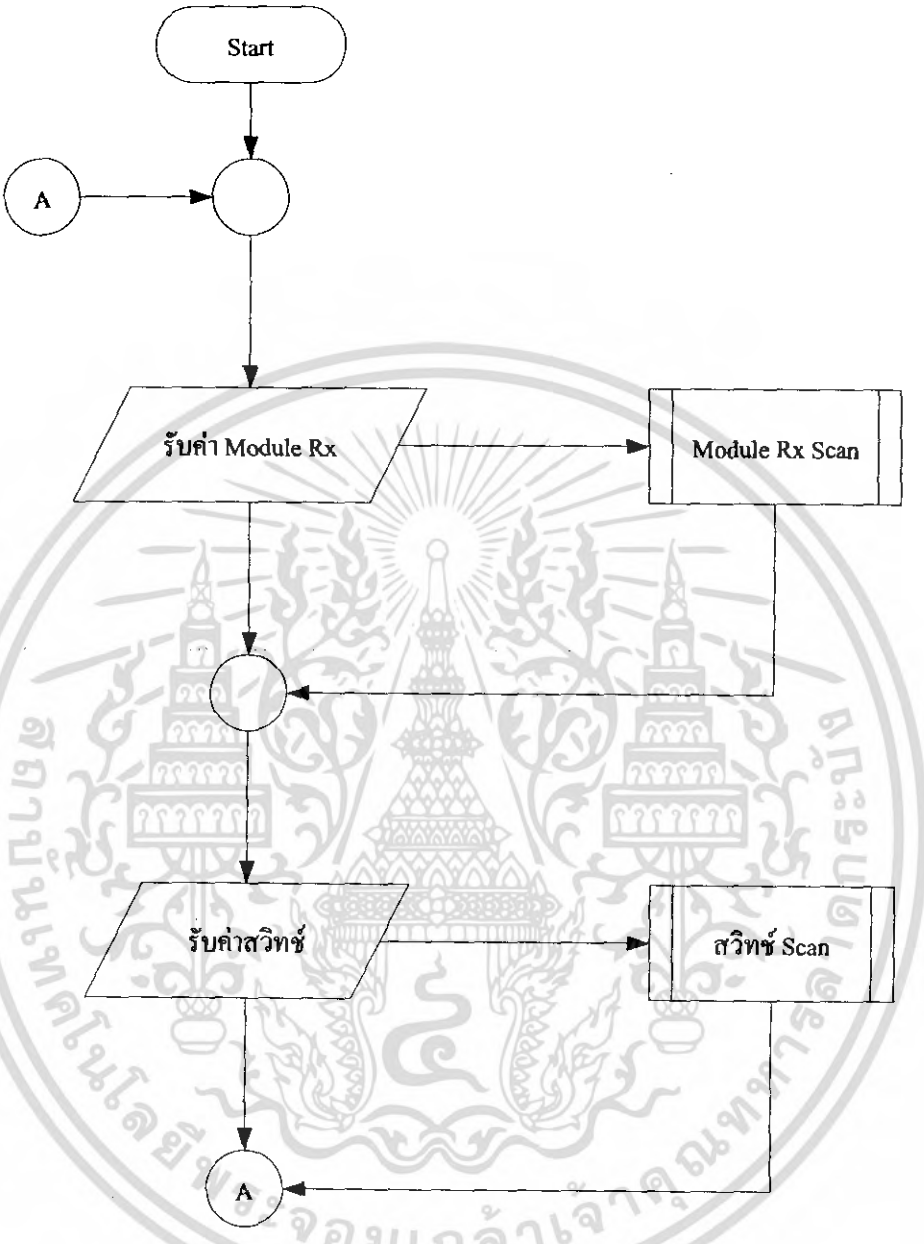
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.20 โฟลว์ชาร์ตการทำงานของโทรศัพท์เคลื่อนที่ (ต่อ)

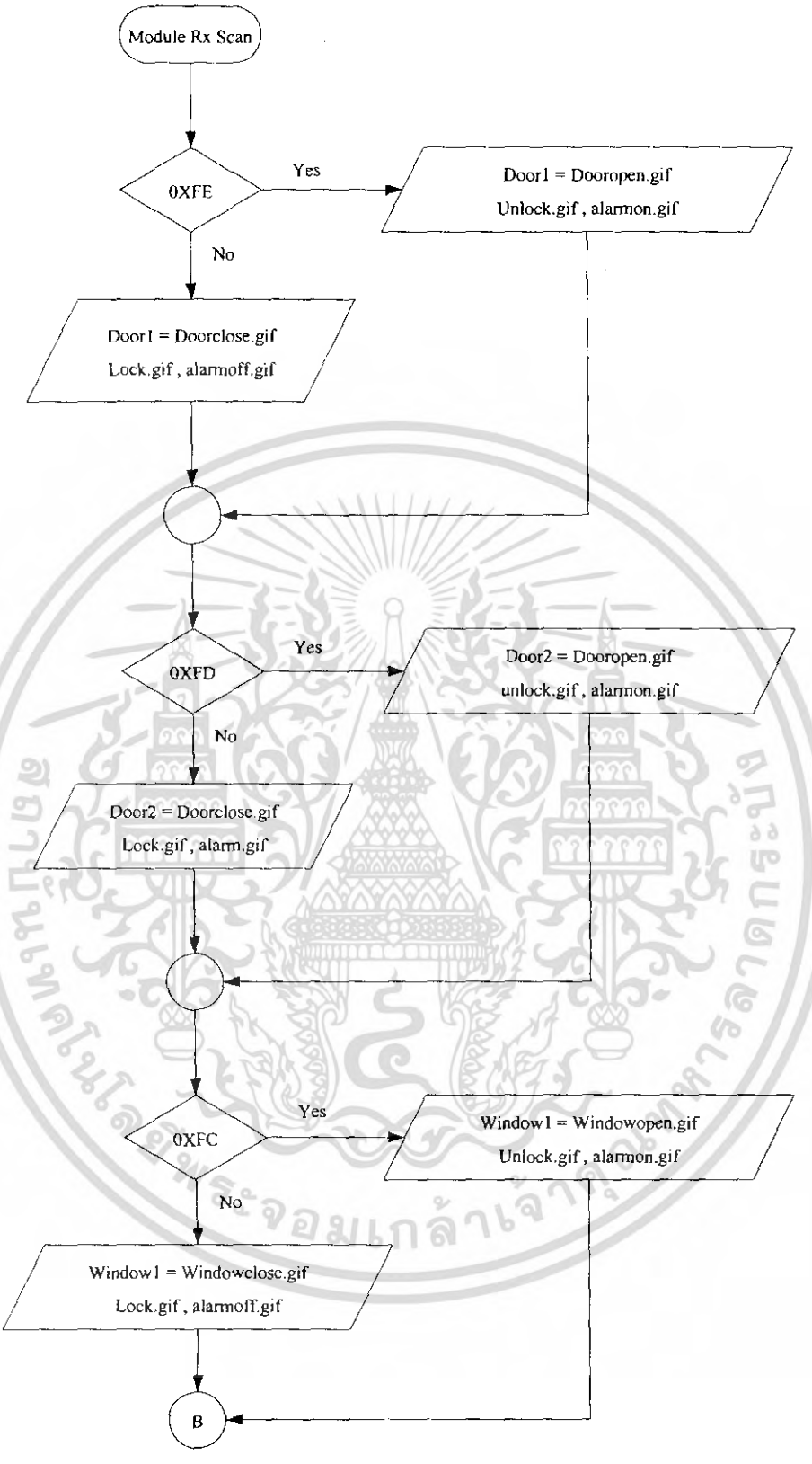
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.14 โฟลว์ชาร์ตการทำงานหลัก Rabbit Core Module RCM 2200



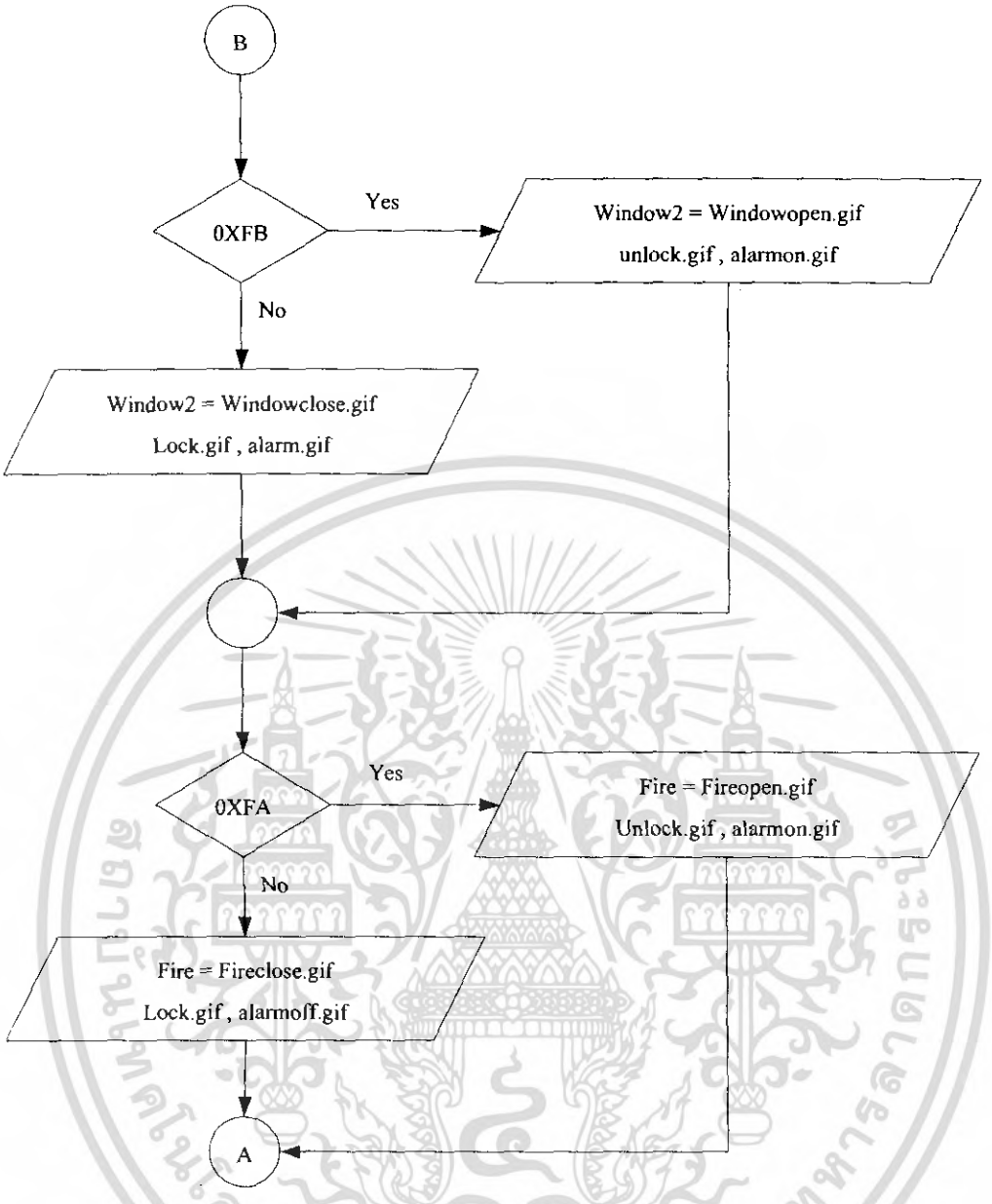
รูปที่ 3.21 โฟลว์ชาร์ตการทำงานหลักของ Rabbit Core Module RCM 2200

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



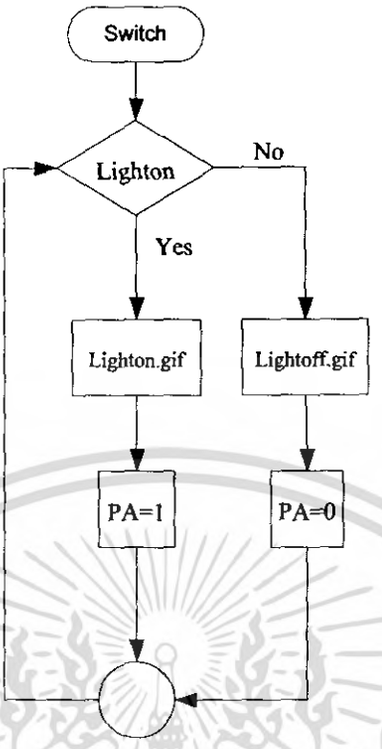
รูปที่ 3.22 โฟลว์ชาร์ตการทำงานส่วนของ Module Rx Scan

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.22 โฟลว์ชาร์ตการทำงานส่วนของ Module Rx Scan (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



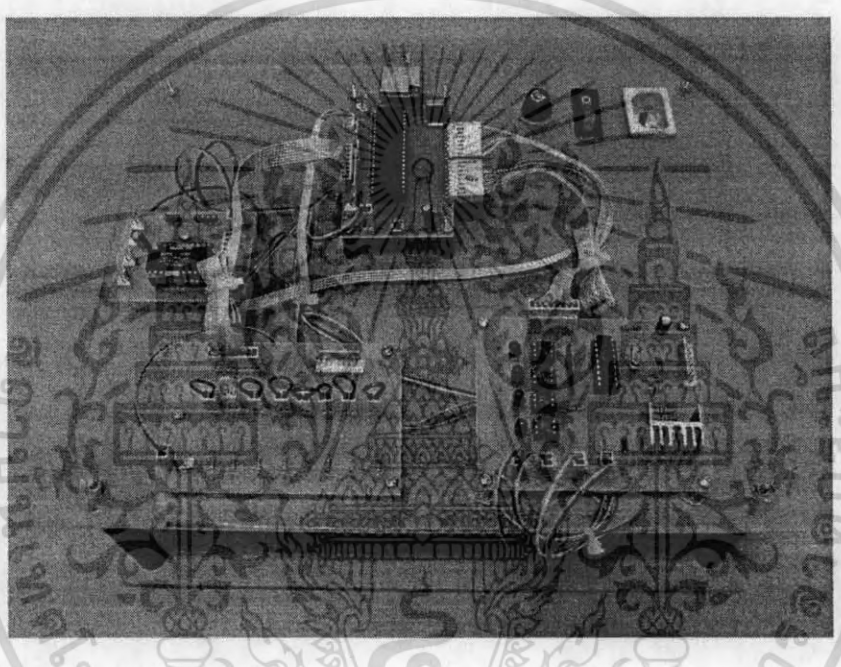
รูปที่ 3.23 โฟลว์ชาร์ตการทำงานส่วนของสวิทช์ Scan

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดลอง และผลการทดลอง

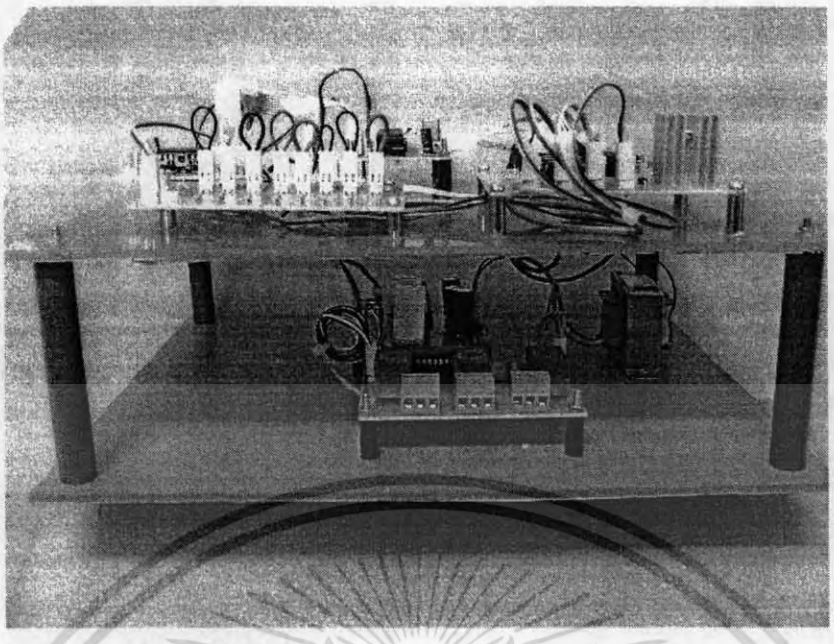
4.1 อุปกรณ์ที่ใช้ในการทดลอง

- วงจรเซ็นเซอร์และภาคส่งสัญญาณไร้สาย
- วงจรภาครับสัญญาณไร้สายจากเซ็นเซอร์
- คอมพิวเตอร์ 1 เครื่อง ใช้ในการแสดงผลหน้าเว็บเพจ
- วงจรสื่อสารกับโทรศัพท์เคลื่อนที่ พร้อมกับโทรศัพท์เคลื่อนที่ยี่ห้อซีเมนต์รุ่น C35i
- แหล่งจ่ายไฟ

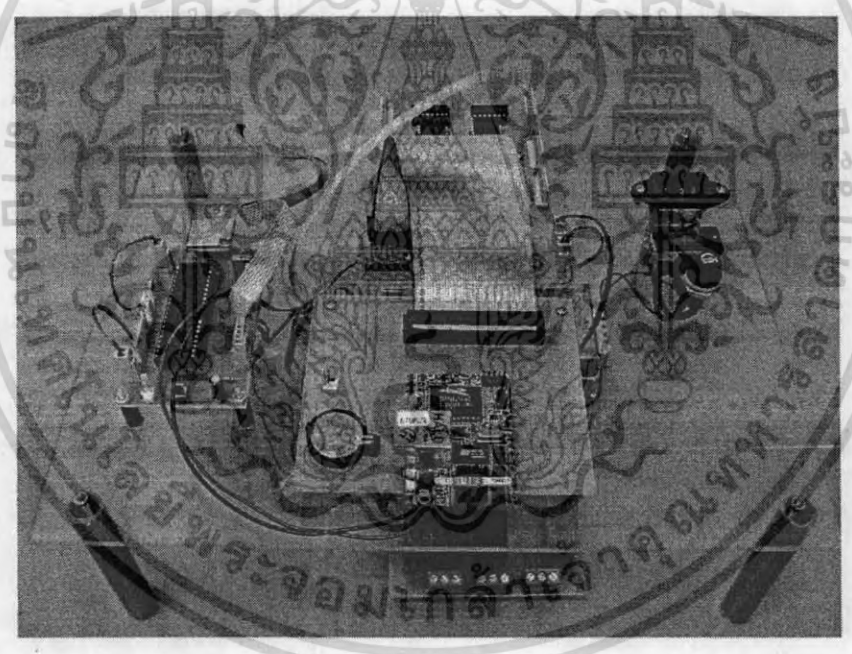


รูปที่ 4.1 วงจรเซ็นเซอร์และภาคส่งสัญญาณไร้สาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

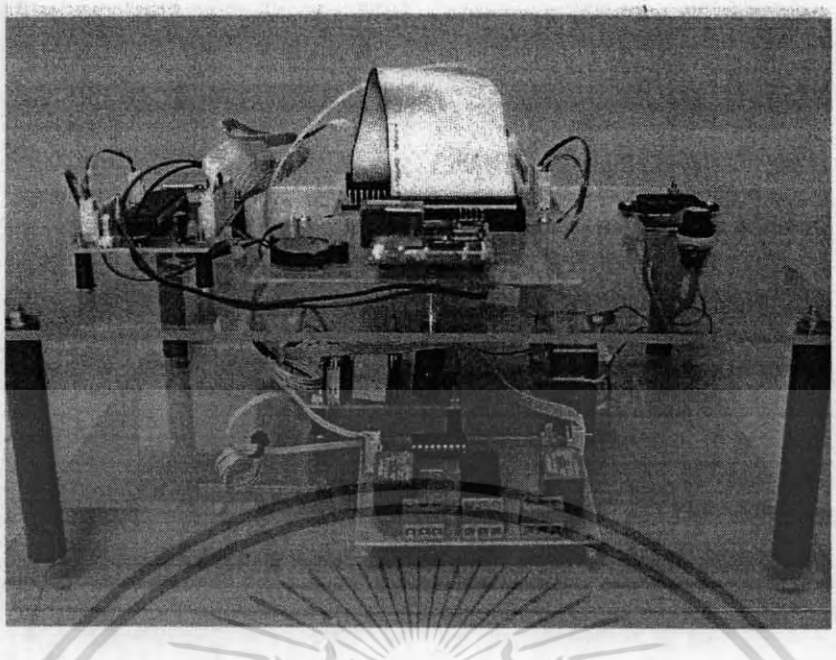


รูปที่ 4.2 วงจรเพาเวอร์ซัพพลายและรีเลย์ของวงจรเซ็นเซอร์และภาคส่งสัญญาณไร้สาย

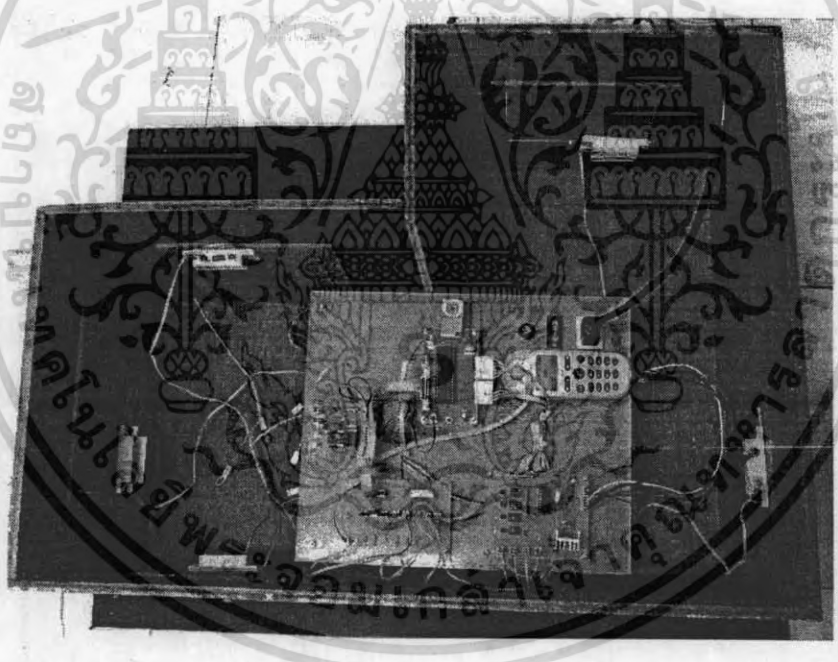


รูปที่ 4.3 วงจรภาครับสัญญาณไร้สาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 วงจรเพาเวอร์ซัพพลายและรีเลย์ของวงจรภาครับสัญญาณไร้สาย



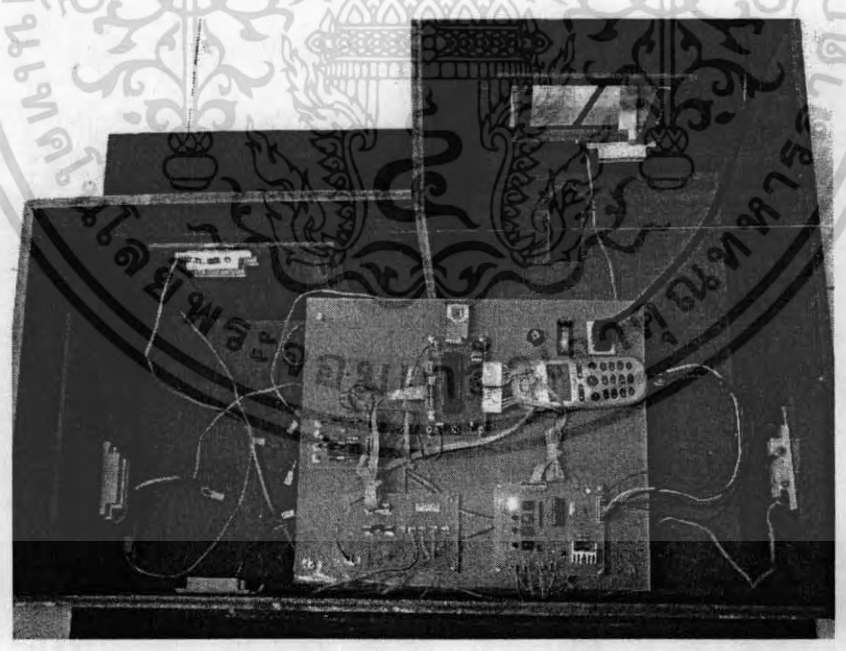
รูปที่ 4.5 การติดตั้งระบบเซ็นเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.6 หน้าเว็บเพจขณะสถานะปกติ

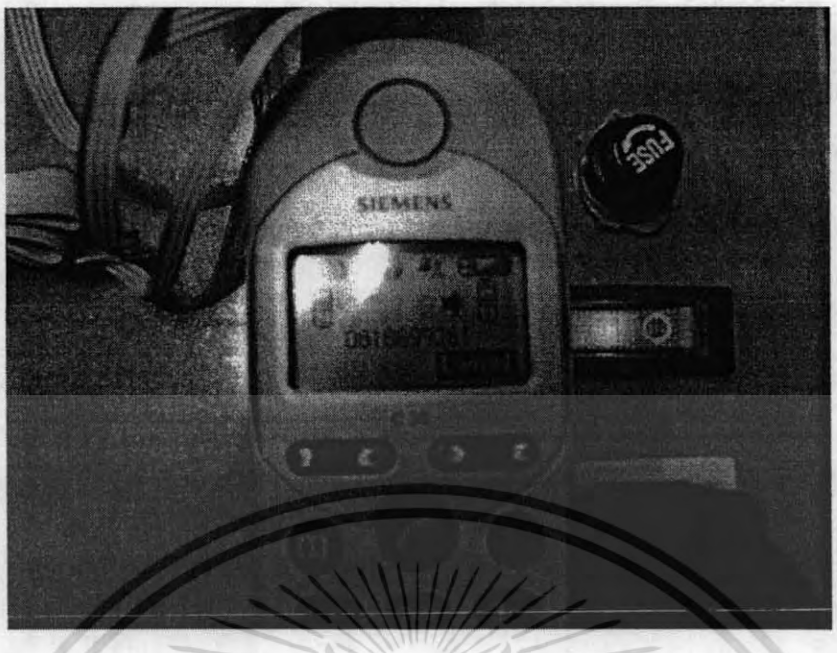
4.2 การทดลองการรับค่าจากระบบเตือนภัยในบ้าน

ในโครงงานนี้จะทำการรับค่าจากอุปกรณ์เตือนภัย เมื่ออุปกรณ์ตรวจจับส่งสัญญาณเตือนว่ามีผู้บุกรุกเปิดประตูเข้ามาอุปกรณ์เตือนภัยจะทำการโทรศัพท์มายังเจ้าของบ้าน เพื่อแจ้งให้ทราบว่าประตูเปิดออกโดยผู้บุกรุก



รูปที่ 4.7 เมื่อประตูเปิดออก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.8 โทรศัพท์ที่อยู่กับวงจรเซ็นเซอร์ทำการโทรไปที่เบอร์เจ้าของบ้าน

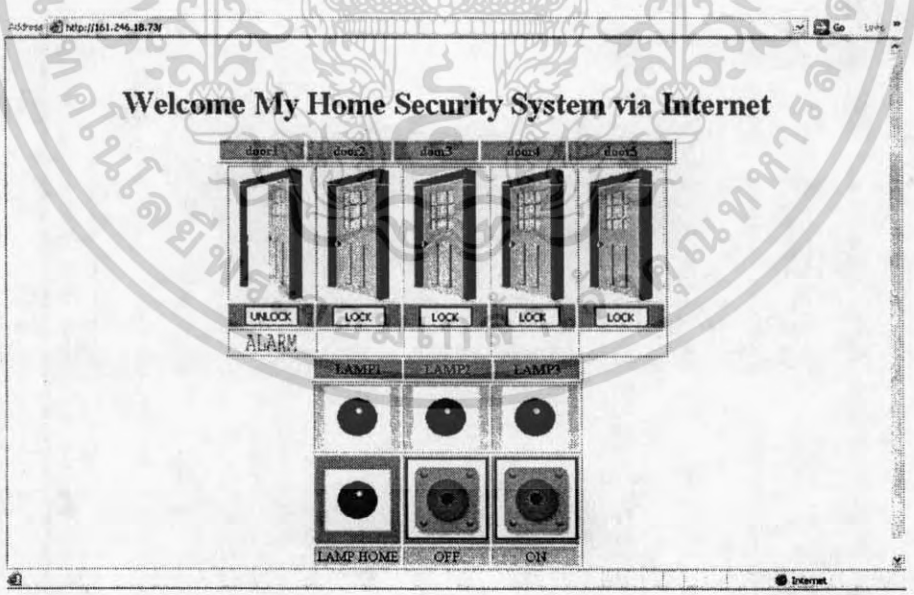
รูปที่ 4.9 แสดงการโทรเข้าที่โทรศัพท์ปลายทางเมื่อมีผู้บุกรุก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อทำการโทรไปยังเจ้าของบ้านและจะทำการส่งข้อความสั้นเพื่อแสดงให้ดูว่าประตูใดมีผู้บุกรุกเข้ามา

รูปที่ 4.10 แสดงข้อความสั้นเมื่อประตูที่ 1 มีผู้บุกรุกที่เครื่องโทรศัพท์เคลื่อนที่ปลายทาง

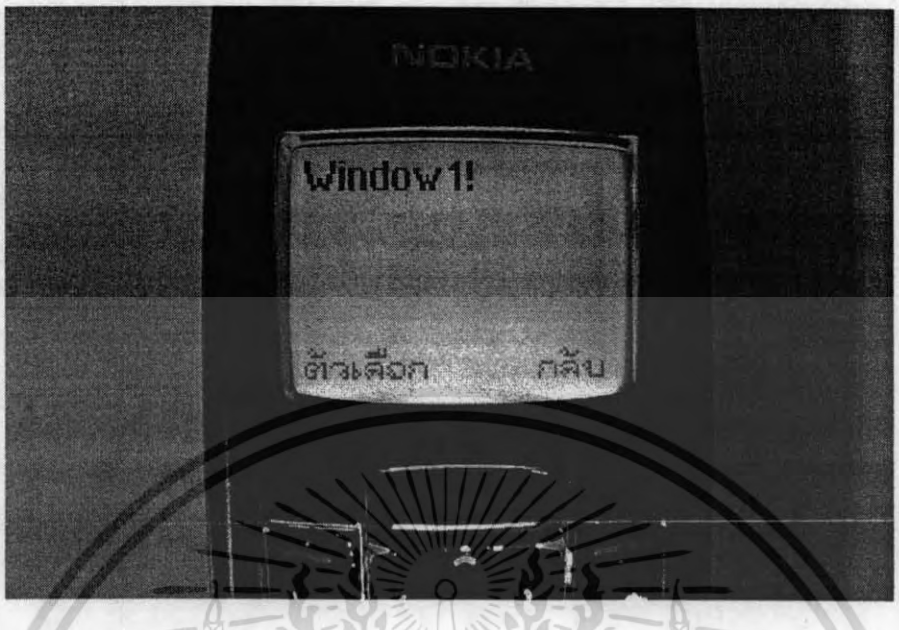
และวงจรเซ็นเซอร์จะส่งสัญญาณไปยังภาครับสัญญาณ เพื่อแสดงสถานะออกหน้าเว็บเพจเมื่อประตูที่ 1 เปิด



รูปที่ 4.11 แสดงผลหน้าเว็บเพจเมื่อประตูที่ 1 เปิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และเมื่อมีผู้บุกรุกที่หน้าต่าง จะทำการส่งข้อความสั้นไปยังโทรศัพท์ปลายทางเพื่อแจ้งให้ทราบ



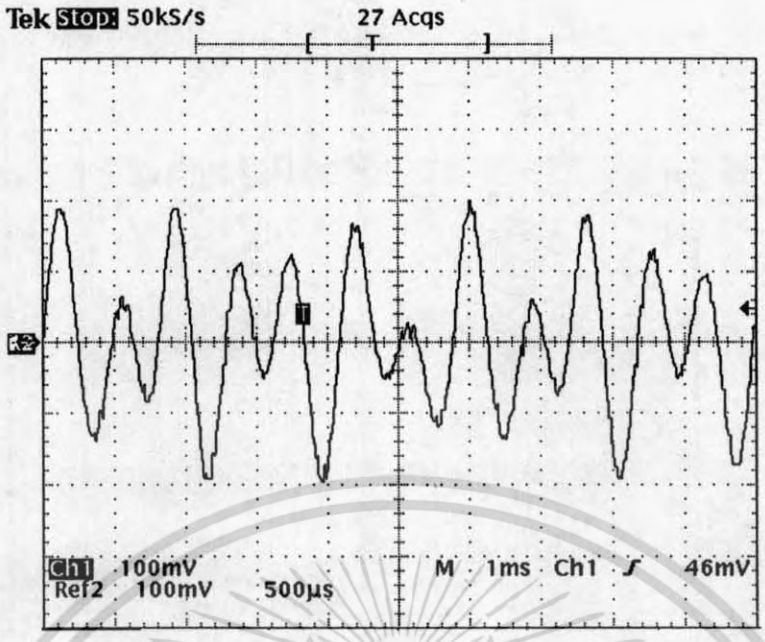
รูปที่ 4.12 แสดงข้อความสั้นเมื่อหน้าต่างที่ 1 มีผู้บุกรุกที่เครื่องโทรศัพท์เคลื่อนที่ปลายทาง

และวงจรเซ็นเซอร์จะส่งสัญญาณไปยังภาครับสัญญาณ เพื่อแสดงสถานะที่หน้าเว็บเพจเมื่อหน้าต่างที่ 1 เปิด

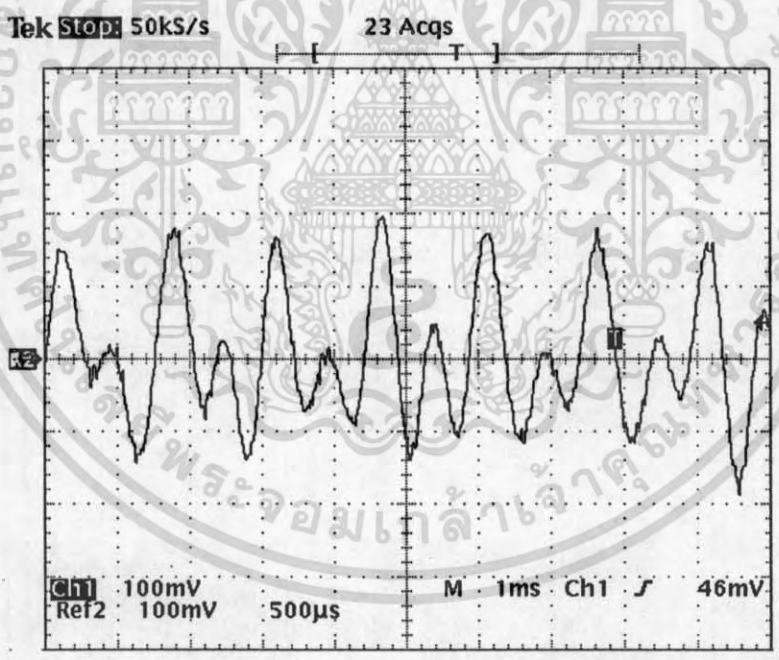
4.3 การทดลองการวัดสัญญาณ DTMF จากโทรศัพท์เคลื่อนที่

ในส่วนการรับค่าการสั่งงานจากโทรศัพท์เคลื่อนที่ขาที่ 11 ของโทรศัพท์เคลื่อนที่ซีเมนต์ C35i จะมีสัญญาณสองความถี่ผสมกันอยู่ จะนำสัญญาณส่งไปยังไอซี MT8870 จะได้ค่าเอาท์พุท Q1- Q4 และนำสัญญาณที่ได้ส่งไป MCS – 51 เพื่อนำไปควบคุมอุปกรณ์ไฟฟ้าต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

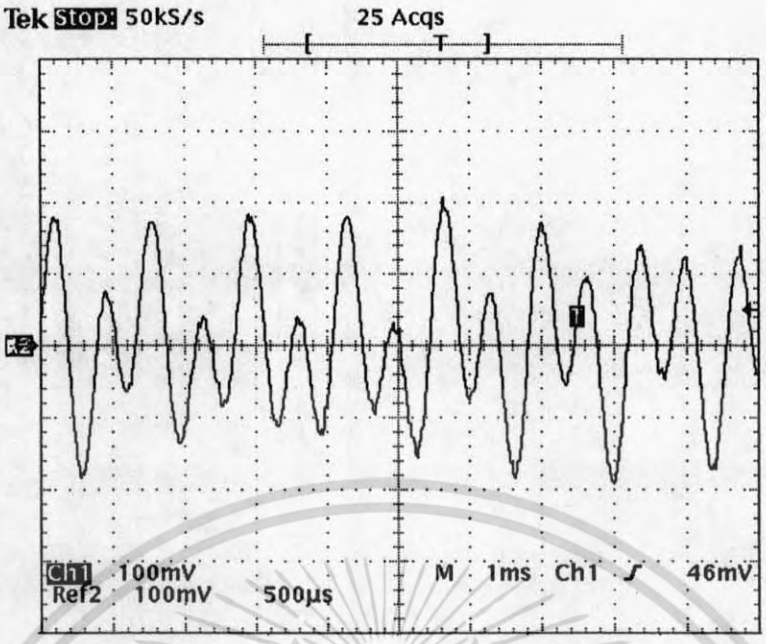


รูปที่ 4.13 สัญญาณขาที่ 11 ของโทรศัพท์เคลื่อนที่เครื่องรับเมื่อต้นทางกดหมายเลข 1

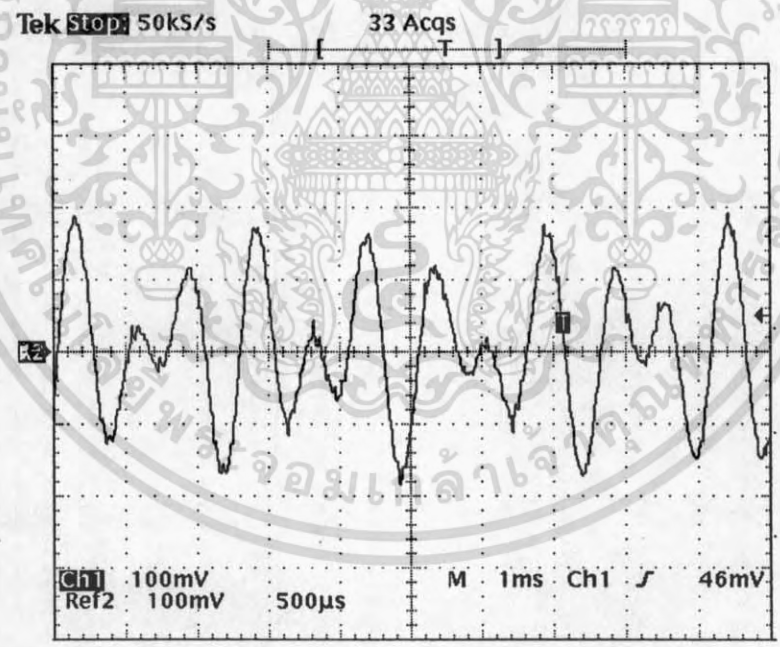


รูปที่ 4.14 สัญญาณขาที่ 11 ของโทรศัพท์เคลื่อนที่เครื่องรับเมื่อต้นทางกดหมายเลข 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

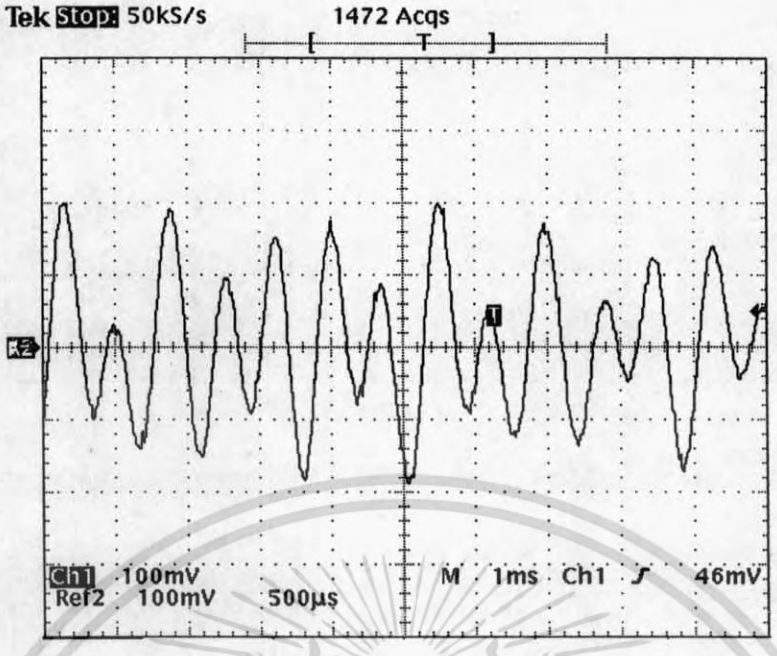


รูปที่ 4.15 สัญญาณขาที่ 11 ของโทรศัพท์เคลื่อนที่เครื่องรับเมื่อต้นทางกดหมายเลข 3

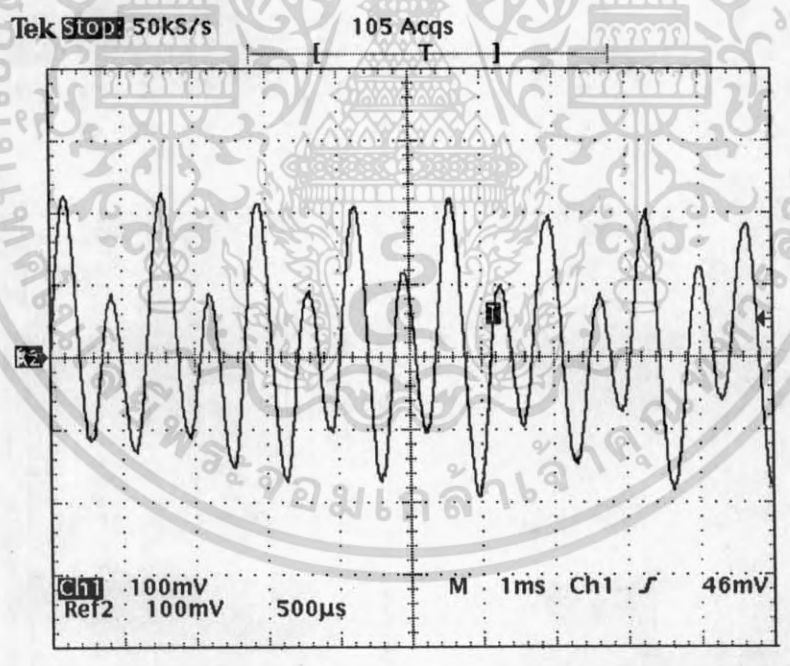


รูปที่ 4.16 สัญญาณขาที่ 11 ของโทรศัพท์เคลื่อนที่เครื่องรับเมื่อต้นทางกดหมายเลข 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

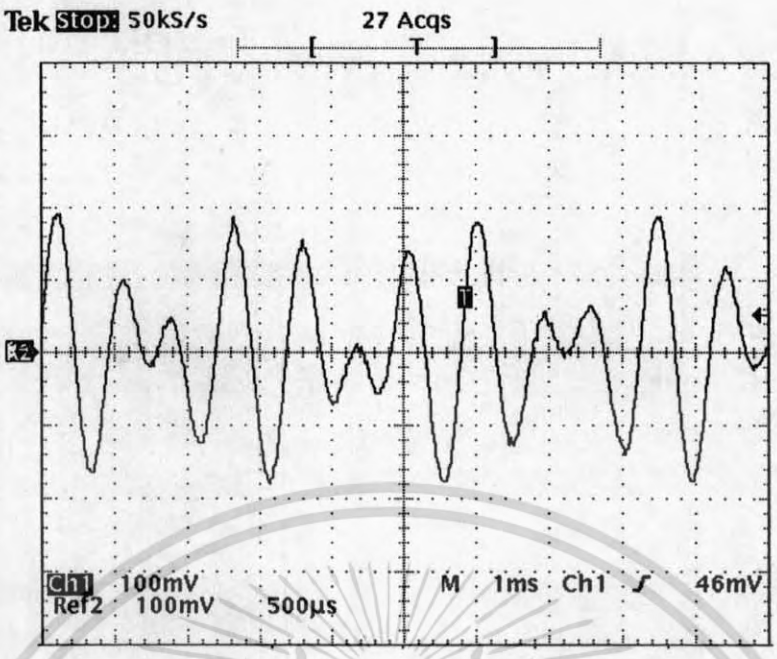


รูปที่ 4.17 สัญญาณขาที่ 11 ของโทรศัพท์เคลื่อนที่เครื่องรับเมื่อต้นทางกดหมายเลข 5

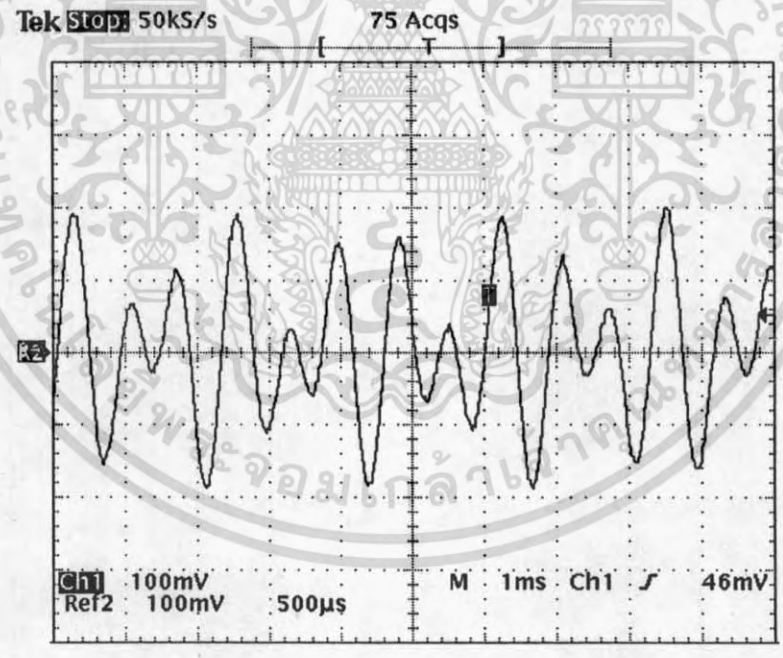


รูปที่ 4.18 สัญญาณขาที่ 11 ของโทรศัพท์เคลื่อนที่เครื่องรับเมื่อต้นทางกดหมายเลข 6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

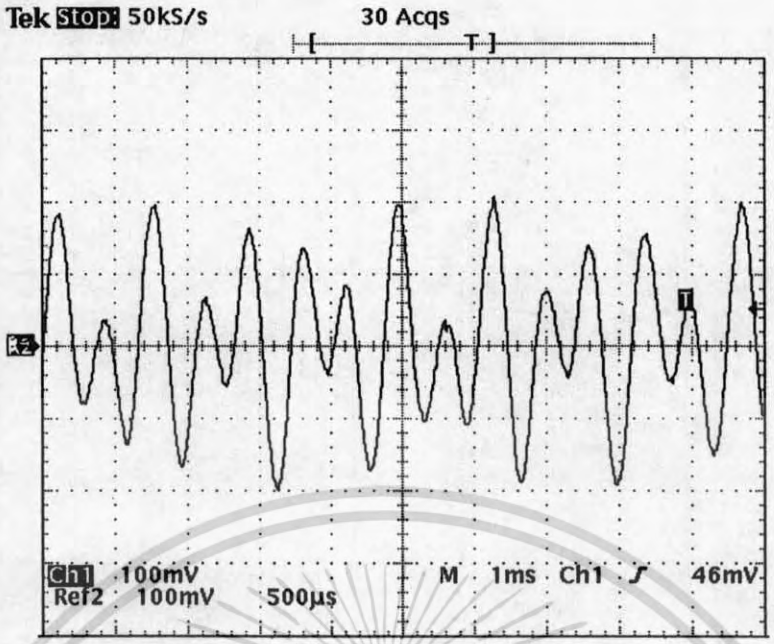


รูปที่ 4.19 สัญญาณขาที่ 11 ของโทรศัพท์เคลื่อนที่เครื่องรับเมื่อต้นทางกดหมายเลข 7

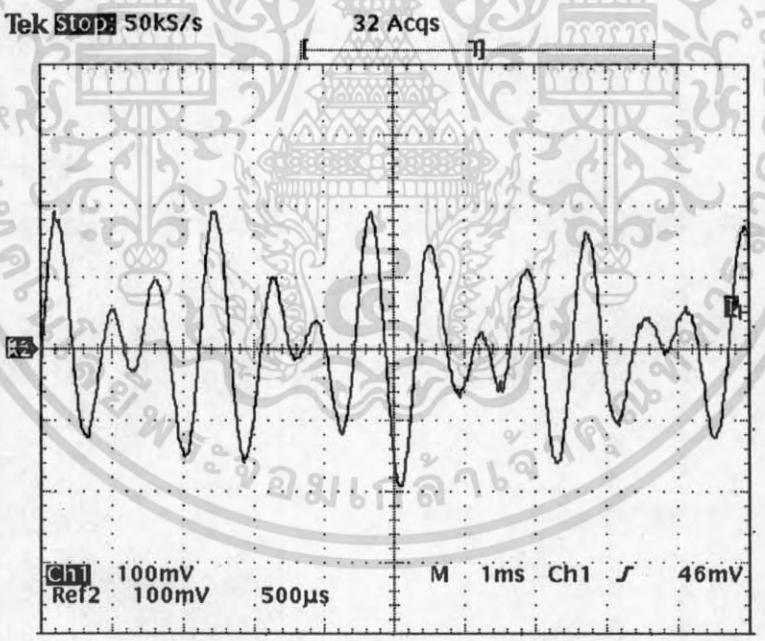


รูปที่ 4.20 สัญญาณขาที่ 11 ของโทรศัพท์เคลื่อนที่เครื่องรับเมื่อต้นทางกดหมายเลข 8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

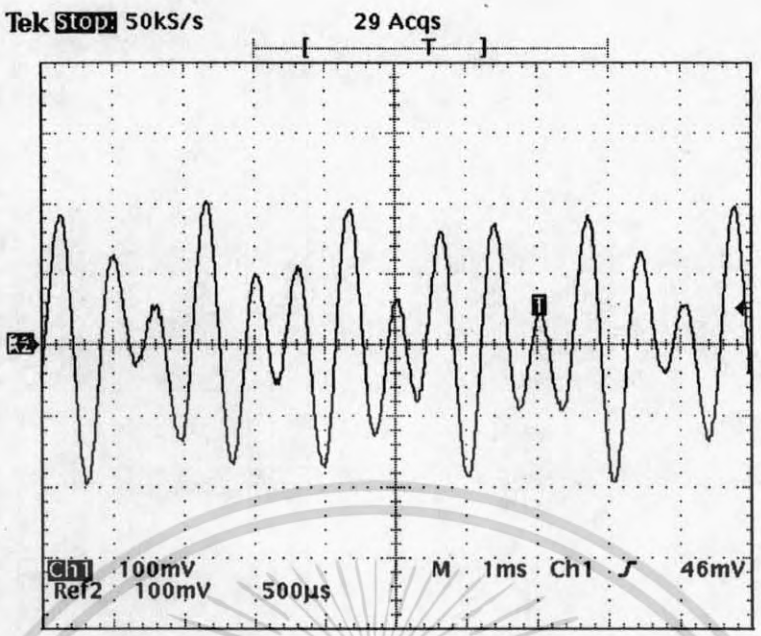


รูปที่ 4.21 สัญญาณขาที่ 11 ของโทรศัพท์เคลื่อนที่เครื่องรับเมื่อต้นทางกดหมายเลข 9

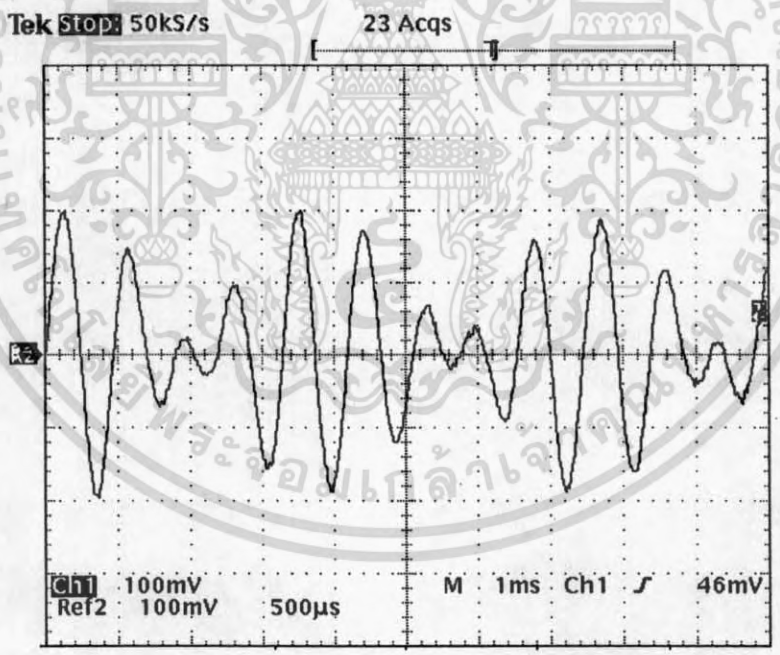


รูปที่ 4.22 สัญญาณขาที่ 11 ของโทรศัพท์เคลื่อนที่เครื่องรับเมื่อต้นทางกดหมายเลข 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

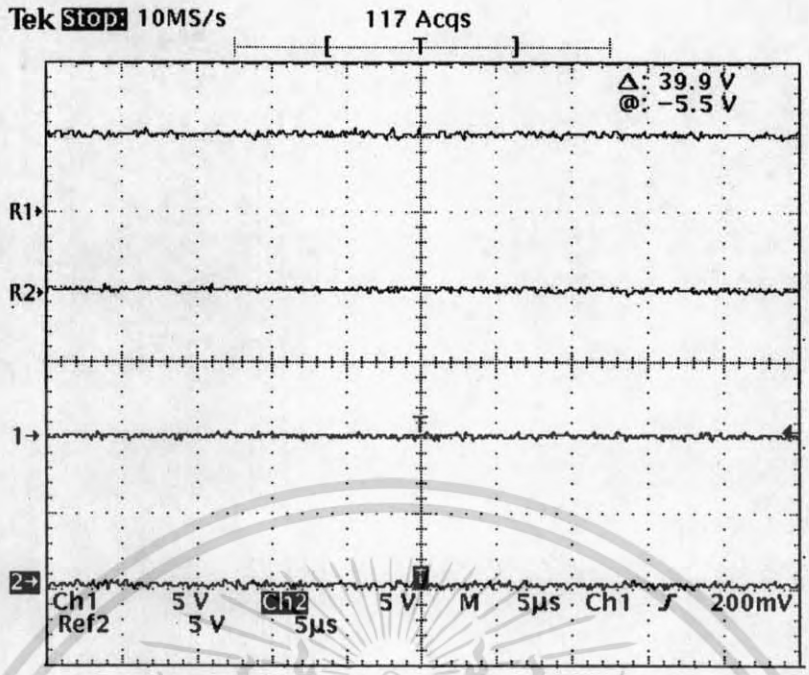


รูปที่ 4.23 สัญญาณขาที่ 11 ของโทรศัพท์เคลื่อนที่เครื่องรับเมื่อต้นทางกดหมายเลข #

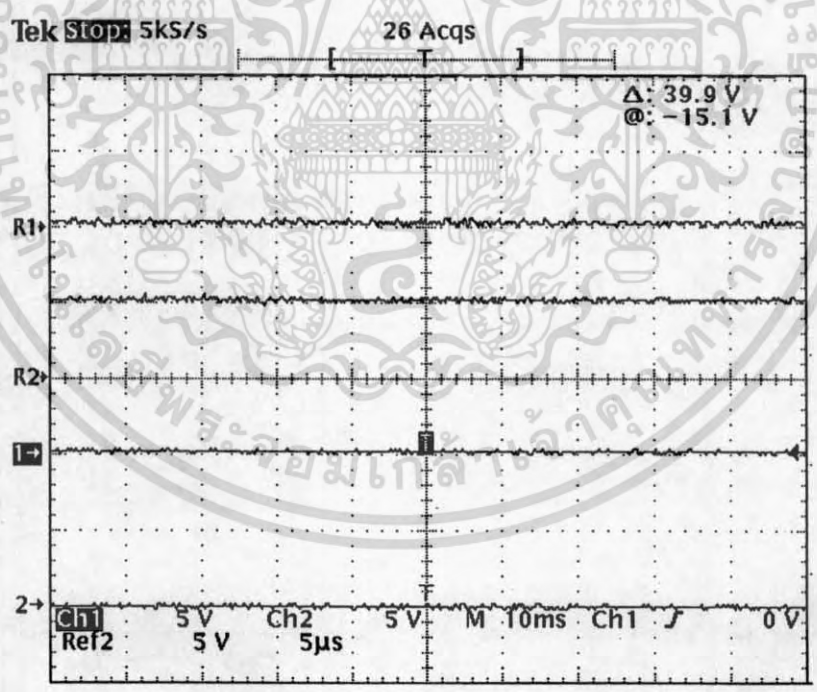


รูปที่ 4.24 สัญญาณขาที่ 11 ของโทรศัพท์เคลื่อนที่เครื่องรับเมื่อต้นทางกดหมายเลข *

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

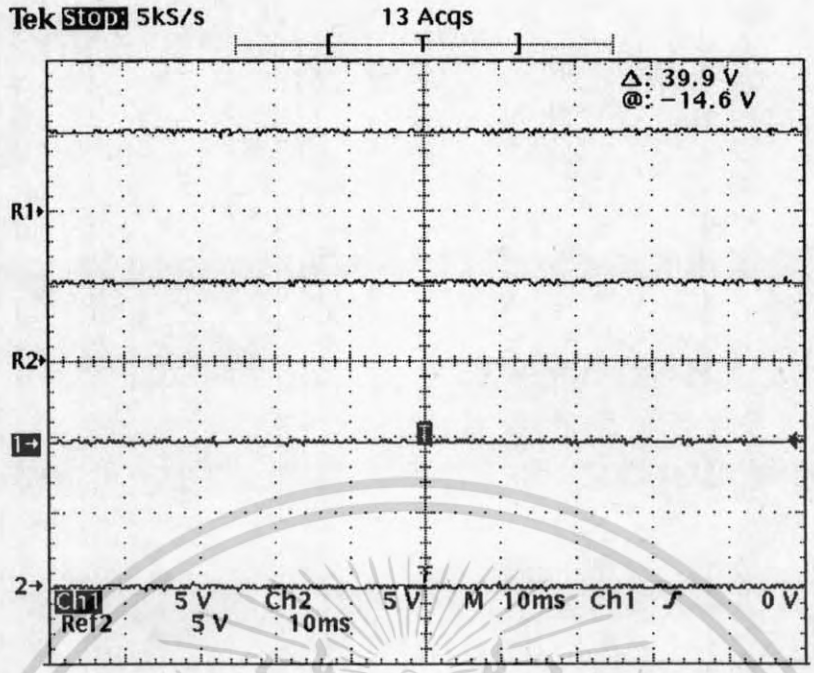


รูปที่ 4.25 แสดงสัญญาณ Q1 ของไอซี MT 8870 เมื่อกดเลข 1 ที่โทรศัพท์เคลื่อนที่ต้นทาง

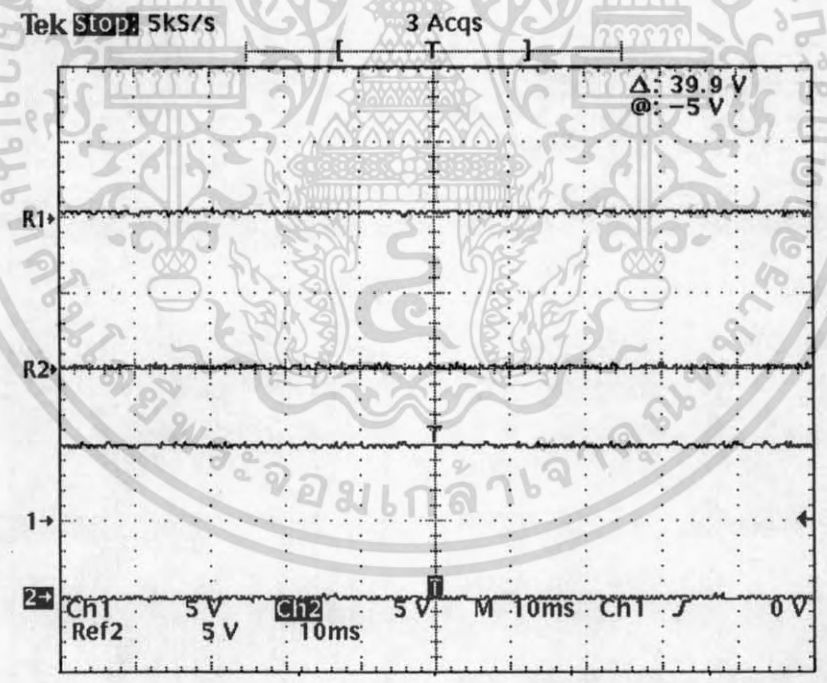


รูปที่ 4.26 แสดงสัญญาณ Q2 ของไอซี MT 8870 เมื่อกดเลข 2 ที่โทรศัพท์เคลื่อนที่ต้นทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

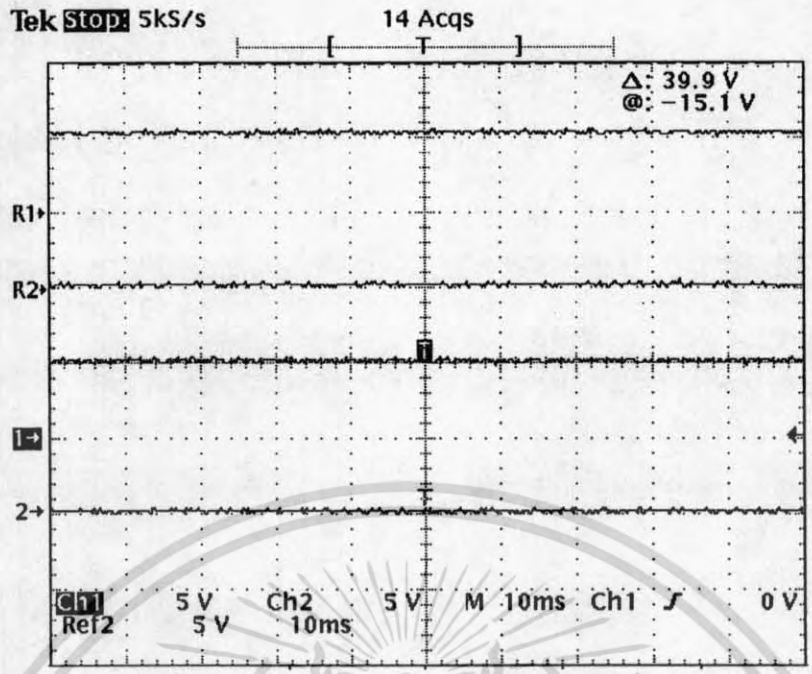


รูปที่ 4.27 แสดงสัญญาณ Q1 , Q2 ของไอซี MT 8870 เมื่อกดเลข 3 ที่โทรศัพท์เคลื่อนที่ต้นทาง

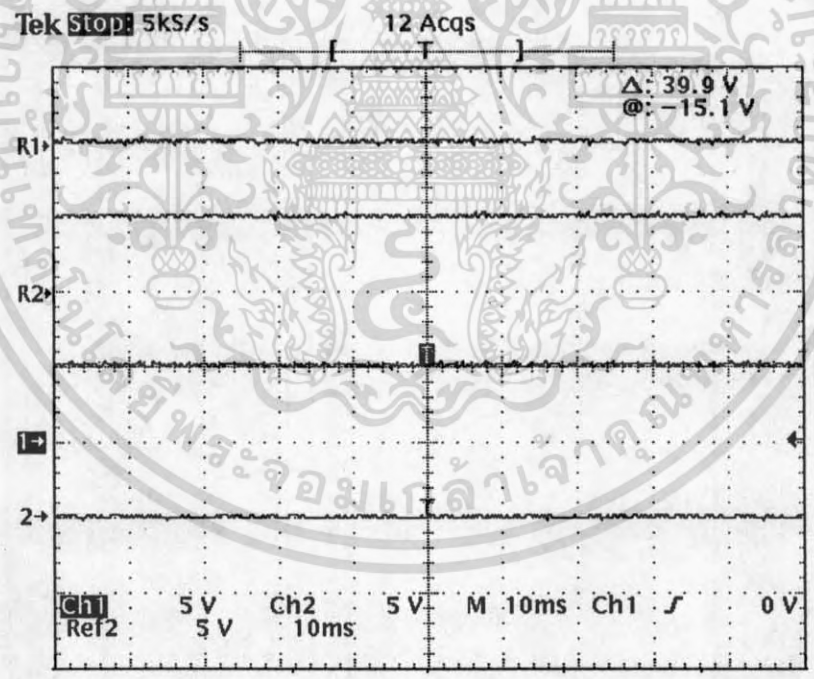


รูปที่ 4.28 แสดงสัญญาณ Q3 ของไอซี MT 8870 เมื่อกดเลข 4 ที่โทรศัพท์เคลื่อนที่ต้นทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

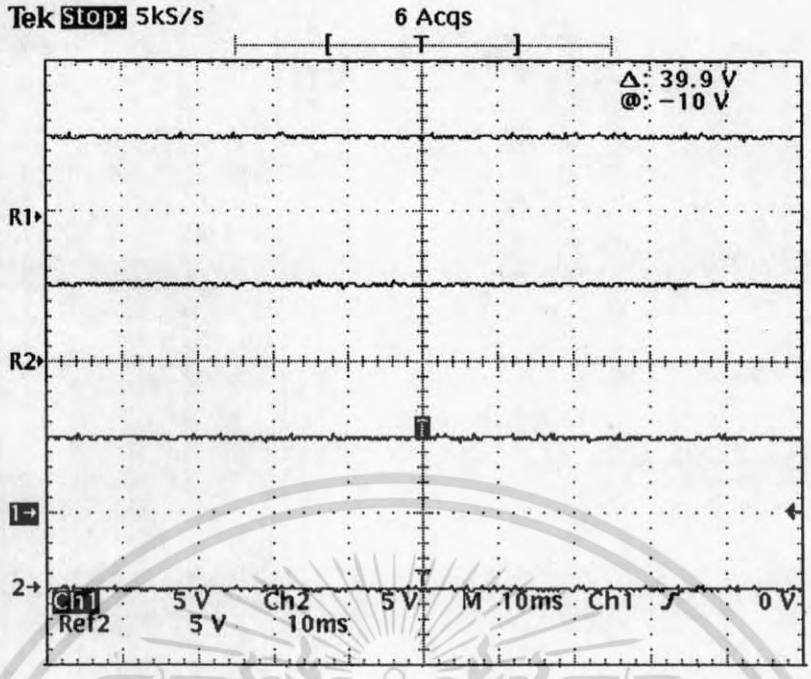


รูปที่ 4.29 แสดงสัญญาณ Q1 , Q3 ของไอซี MT 8870 เมื่อกดเลข 5 ที่โทรศัพท์เคลื่อนที่ต้นทาง

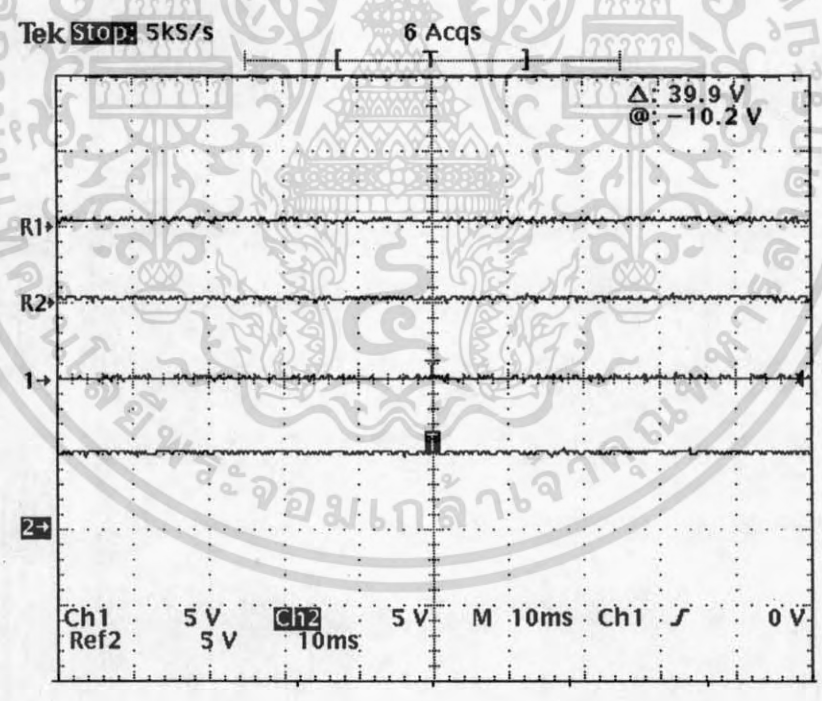


รูปที่ 4.30 แสดงสัญญาณ Q2 , Q3 ของไอซี MT 8870 เมื่อกดเลข 6 ที่โทรศัพท์เคลื่อนที่ต้นทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

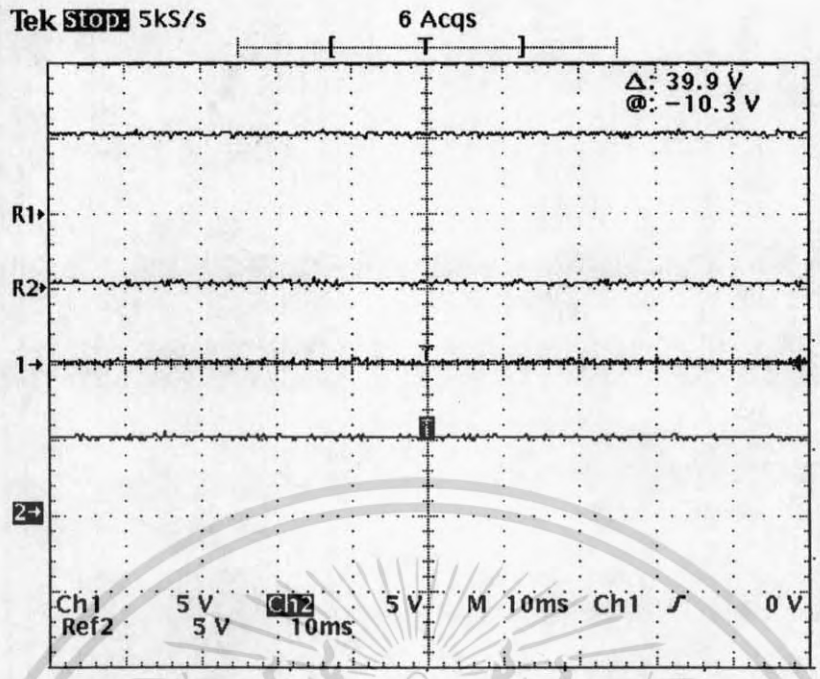


รูปที่ 4.31 แสดงสัญญาณ Q1 , Q2 , Q3 ของไอซี MT 8870 เมื่อกดเลข 7 ที่โทรศัพท์เคลื่อนที่ต้นทาง

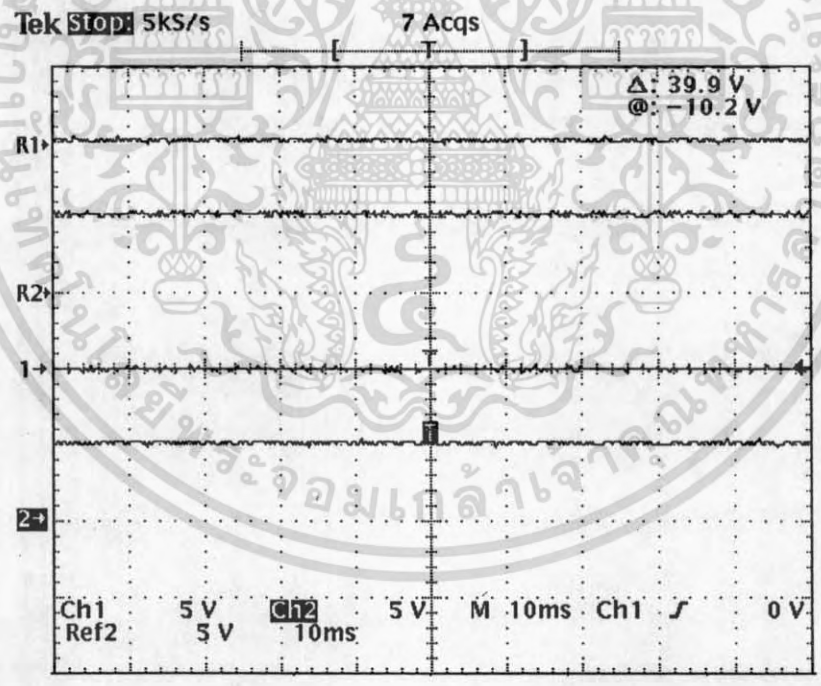


รูปที่ 4.32 แสดงสัญญาณ Q4 ของไอซี MT 8870 เมื่อกดเลข 8 ที่โทรศัพท์เคลื่อนที่ต้นทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

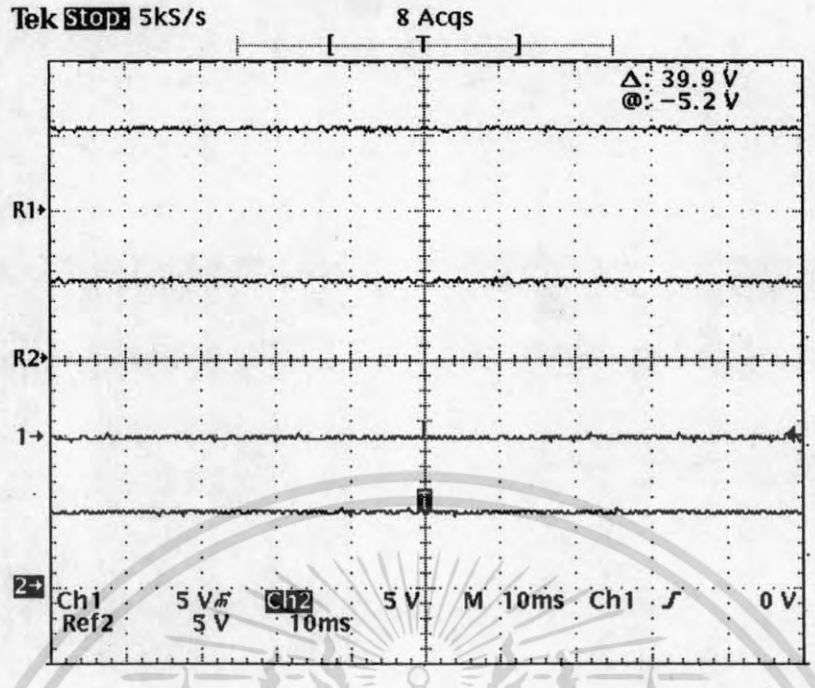


รูปที่ 4.33 แสดงสัญญาณ Q1 , Q3 ของไอซี MT 8870 เมื่อกดเลข 9 ที่โทรศัพท์เคลื่อนที่ต้นทาง

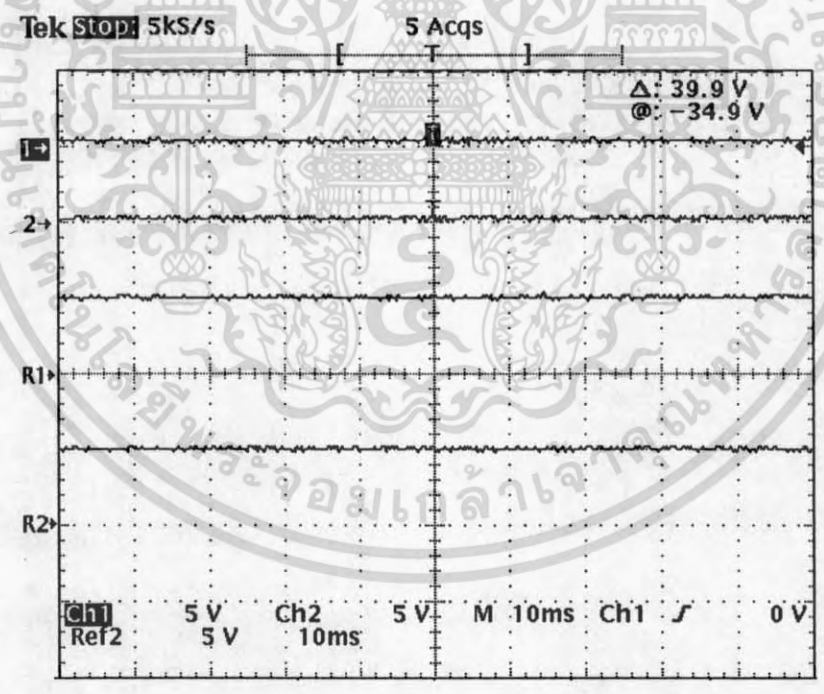


รูปที่ 4.34 แสดงสัญญาณ Q2 , Q4 ของไอซี MT 8870 เมื่อกดเลข 0 ที่โทรศัพท์เคลื่อนที่ต้นทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.35 แสดงสัญญาณ Q1, Q2 , Q4 ของไอซี MT 8870 เมื่อกดเลข * ที่โทรศัพท์เคลื่อนที่ต้นทาง

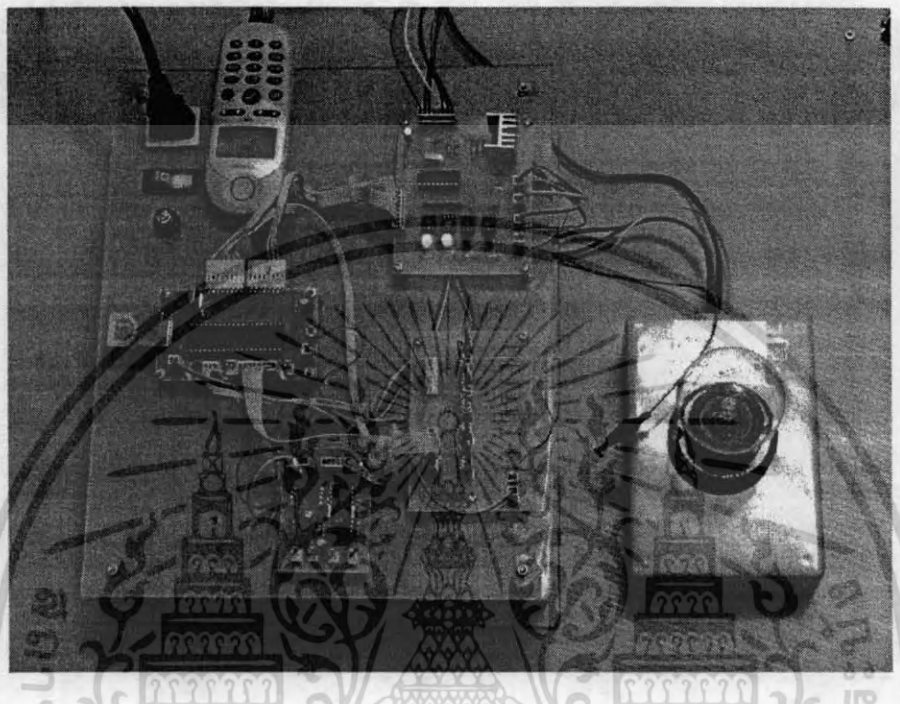


รูปที่ 4.36 แสดงสัญญาณ Q3 , Q4 ของไอซี MT 8870 เมื่อกดเลข # ที่โทรศัพท์เคลื่อนที่ต้นทาง

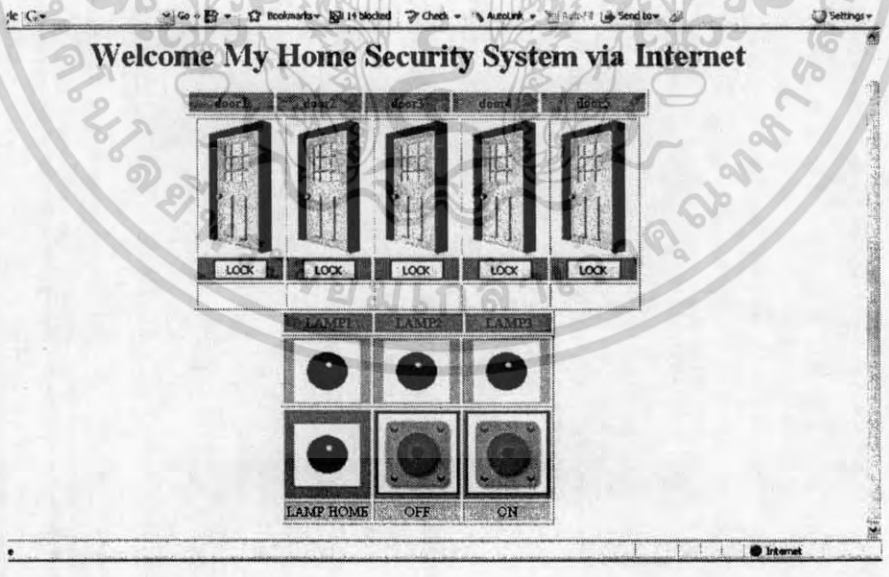
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 การทดลองการสั่งงานอุปกรณ์ไฟฟ้าผ่านโทรศัพท์เคลื่อนที่

เราจะนำสัญญาณที่ได้จากไอซี MT8870 ส่งไปที่ MCS – 51 เพื่อนำไปสั่งงานกับอุปกรณ์ไฟฟ้า และสัญญาณที่ออกจาก MCS – 51 จะส่งไปที่วงจรชุดควบคุมมอเตอร์ไอซี TA7279PA และส่งไปที่วงจรรีเลย์เพื่อควบคุมหลอดไฟ



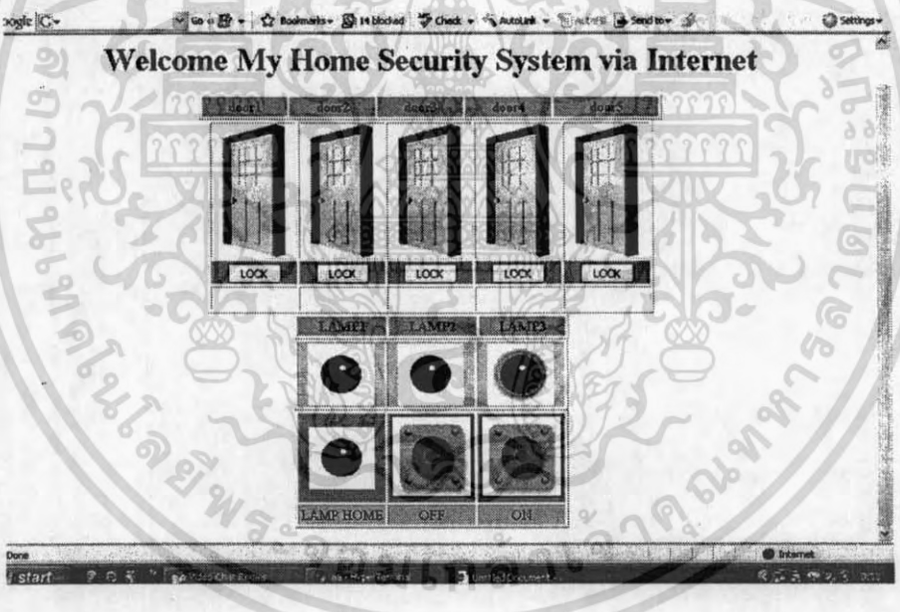
รูปที่ 4.37 วงจรควบคุมการปิดหลอดไฟเมื่อกดเลข 3 ที่โทรศัพท์เคลื่อนที่ต้นทาง



รูปที่ 4.38 แสดงผลหน้าเว็บเพจเมื่อ LAMP ที่ 3 ปิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.39 วงจรควบคุมการเปิดหลอดไฟเมื่อกดเลข 2 ที่โทรศัพท์เคลื่อนที่ต้นทาง

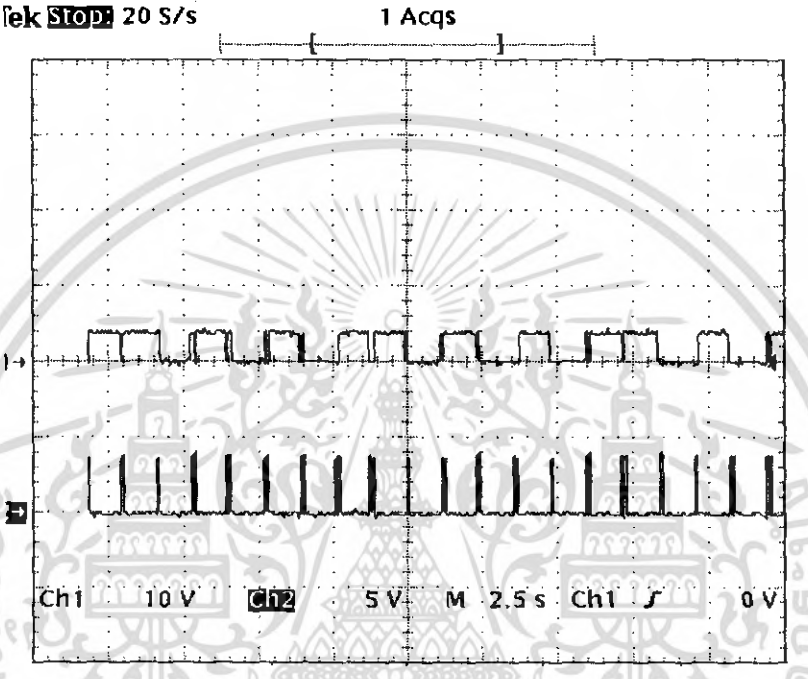


รูปที่ 4.40 แสดงผลหน้าเว็บเพจเมื่อ LAMP ที่ 3 เปิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

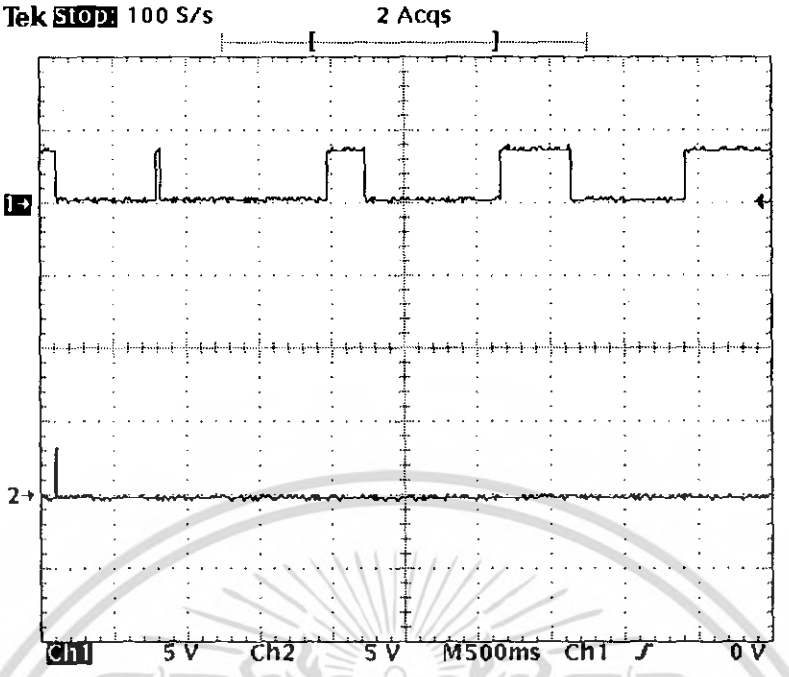
4.5 แสดงผลการวัดขาสัญญาณ (ขา Data , ขา Clk) ของ TRW 2.4 กิกะเฮริต์

เมื่อเซ็นเซอร์ทำงานหรือมีผู้บุกรุกเข้ามาภายในบ้าน ตัวควบคุมหรือ MCS – 51 จะทำการส่งข้อมูลผ่าน TRW 2.4 กิกะเฮริต์ ไปยังเครื่องรับเพื่อแสดงสถานะทางหน้าเว็บเพจ เป็นการแจ้งเตือนให้กับเจ้าของบ้านทราบว่าขณะนี้มีผู้บุกรุก ในการส่งข้อมูลจากเครื่องส่งไปยังเครื่องรับนั้นเราต้องทำการเซ็ตค่าต่างๆให้กับ TRW 2.4 กิกะเฮริต์ เพื่อกำหนดการทำงานให้กับ TRW 2.4 กิกะเฮริต์ โดยที่ 1 คือ ขาสัญญาณ Data และ 2 คือ ขาสัญญาณ Clk

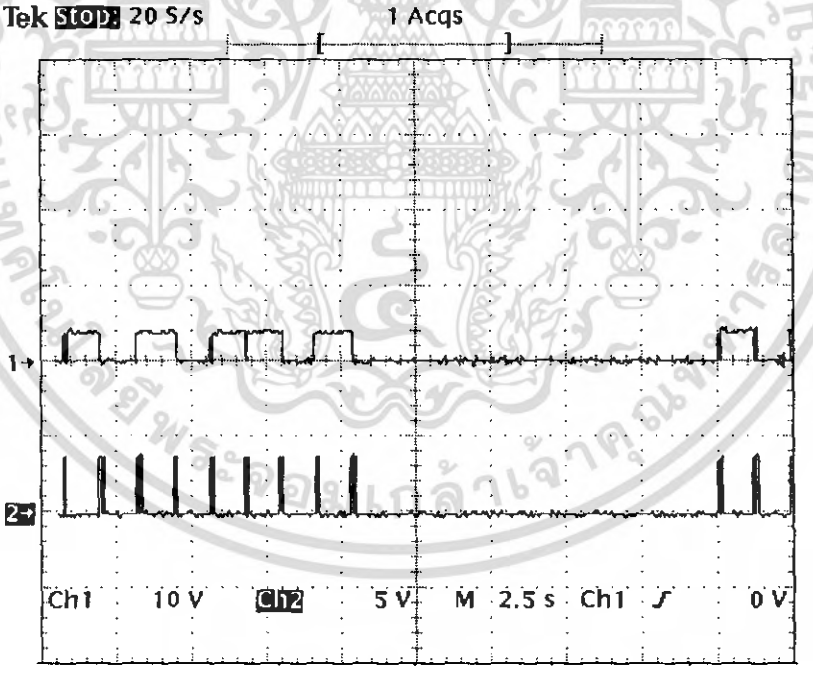


รูปที่ 4.41 แสดงสัญญาณทางด้านส่งของ RF Module TRW 2.4 กิกะเฮริต์ ในกรณีไม่มีผู้บุกรุก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

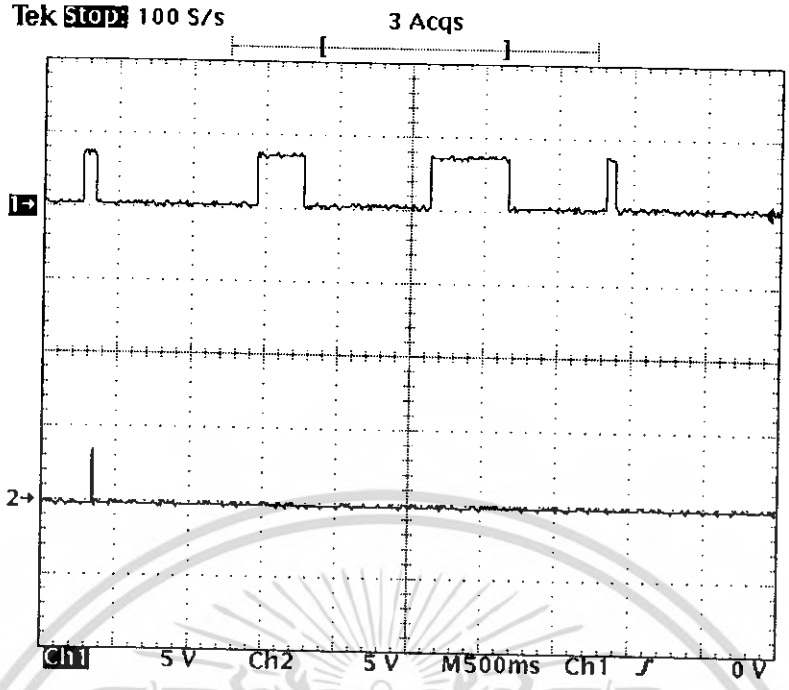


รูปที่ 4.42 แสดงสัญญาณทางด้านรับของ RF Module TRW 2.4 กิกะเฮริ์ต ในกรณีไม่มีผู้บุกรุก

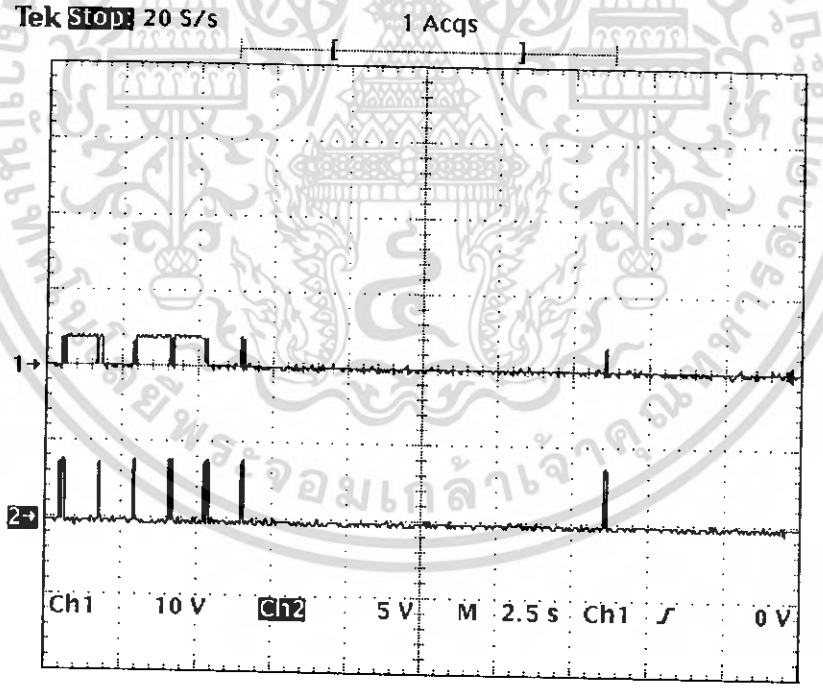


รูปที่ 4.43 แสดงสัญญาณทางด้านส่งของ RF Module TRW 2.4 กิกะเฮริ์ต ในกรณีมีผู้บุกรุกประตูที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

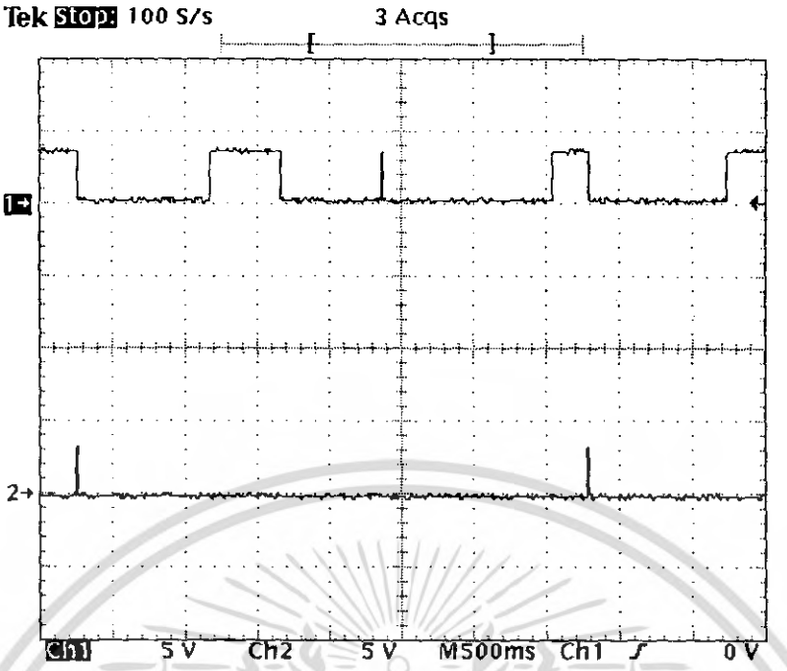


รูปที่ 4.44 แสดงสัญญาณทางด้านรับของ RF Module TRW 2.4 กิกะเฮริ์ต ในกรณีมีผู้บุกรุกประตูที่ 1

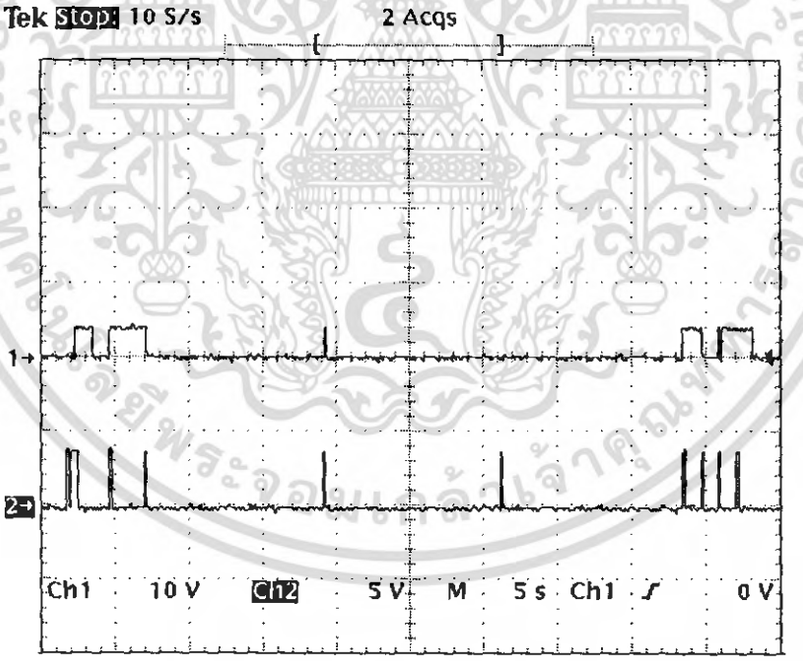


รูปที่ 4.45 แสดงสัญญาณทางด้านส่งของ RF Module TRW 2.4 กิกะเฮริ์ต ในกรณีมีผู้บุกรุกประตูที่ 1,2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

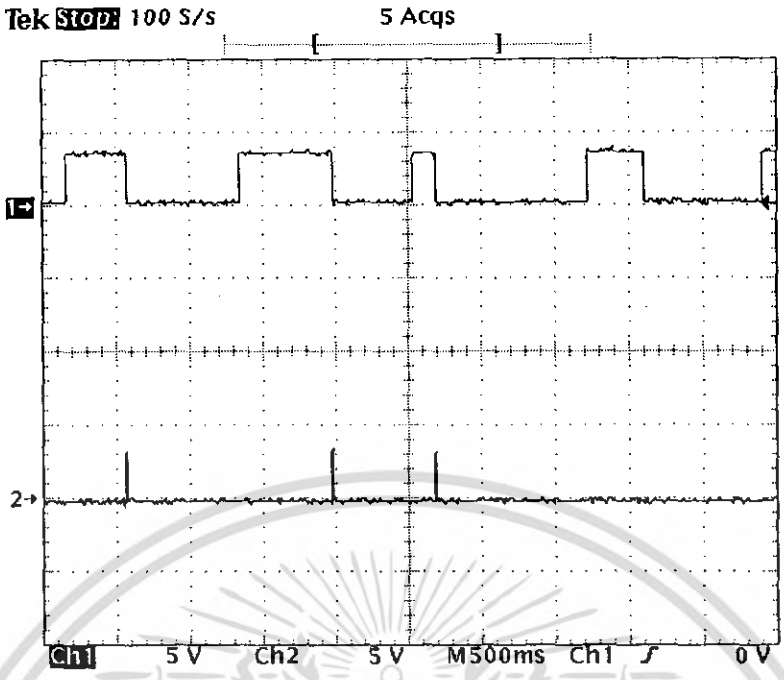


รูปที่ 4.46 แสดงสัญญาณทางด้านรับของ RF Module TRW 2.4 กิกะเฮริ์ต
ในกรณีมีผู้บุกรุกประตูที่ 1,2

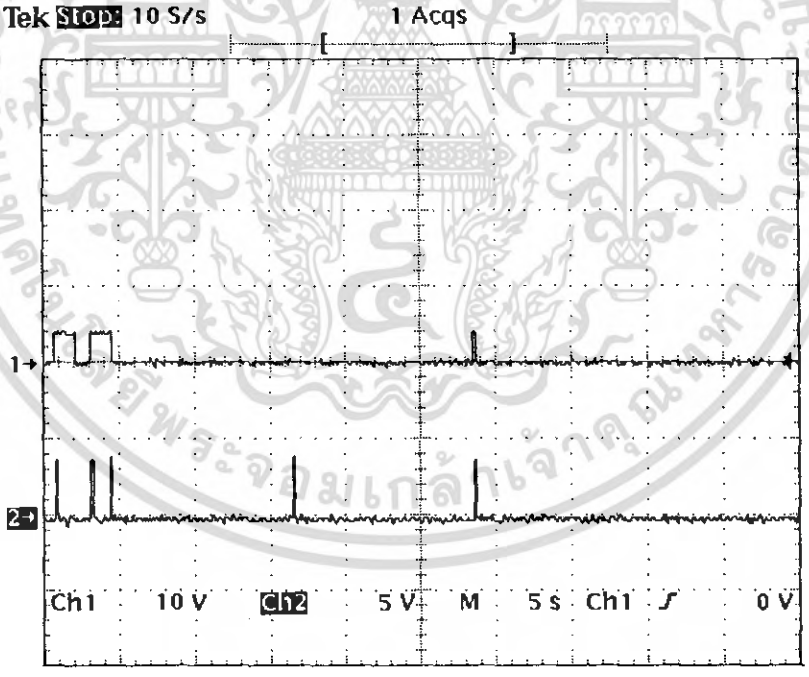


รูปที่ 4.47 แสดงสัญญาณทางด้านส่งของ RF Module TRW 2.4 กิกะเฮริ์ต
ในกรณีมีผู้บุกรุกประตูที่ 1,2,3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

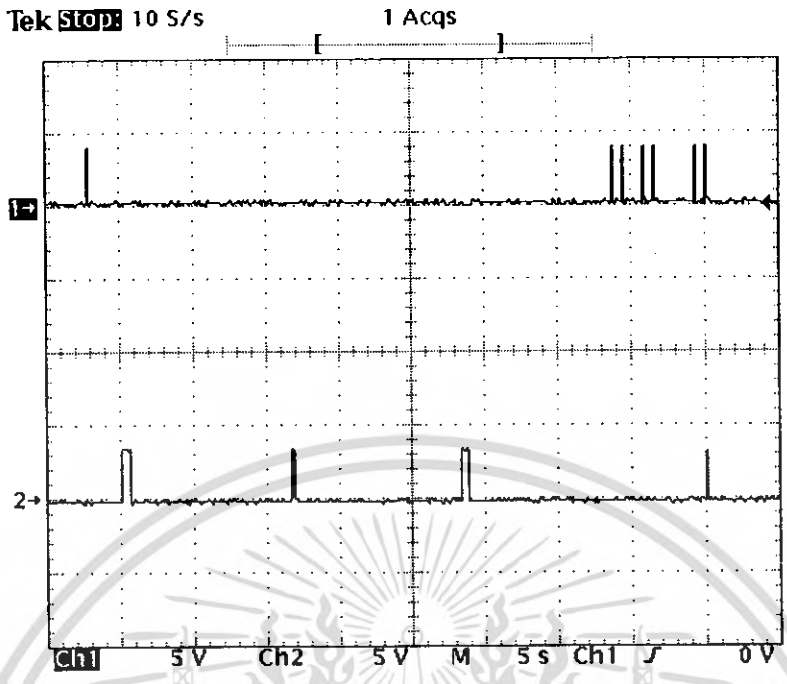


รูปที่ 4.48 แสดงสัญญาณทางด้านรับของ RF Module TRW 2.4 กิกะเฮริด์ ในกรณีมีผู้บุกรุกประตูที่ 1,2,3

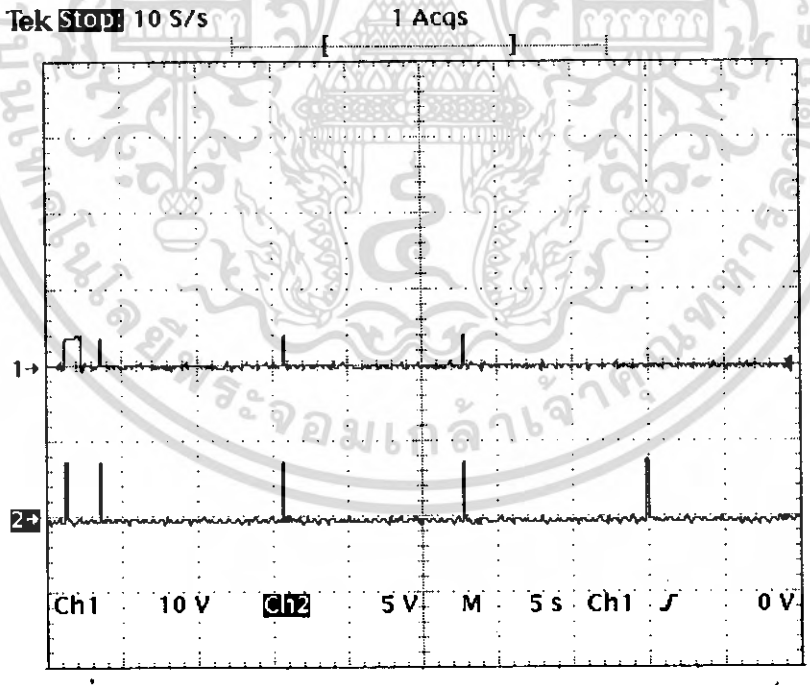


รูปที่ 4.49 แสดงสัญญาณทางด้านส่งของ RF Module TRW 2.4 กิกะเฮริด์ ในกรณีมีผู้บุกรุกประตูที่1,2,3,4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

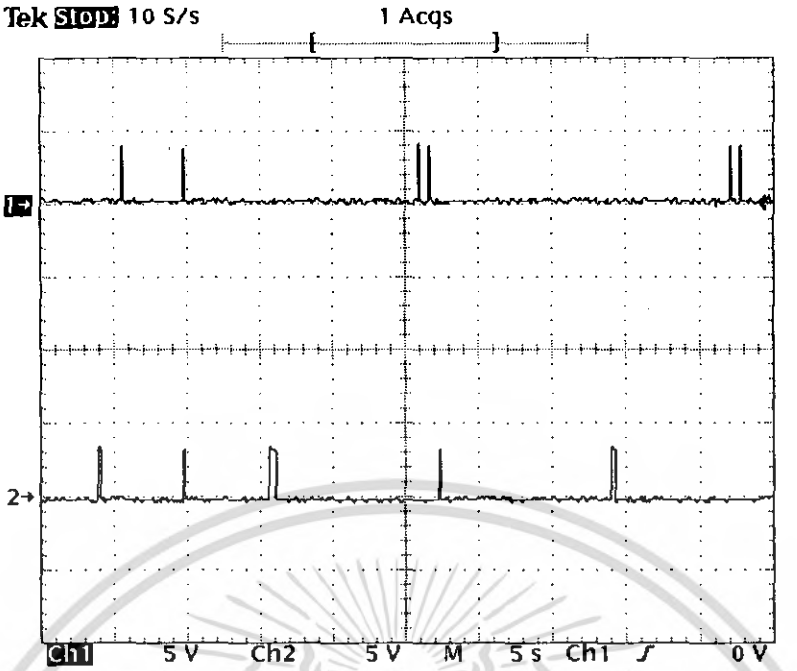


รูปที่ 4.50 แสดงสัญญาณทางด้านรับของ RF Module TRW 2.4 กิกะเฮริ์ต
ในกรณีมีผู้บุกรุกประตูที่1,2,3,4

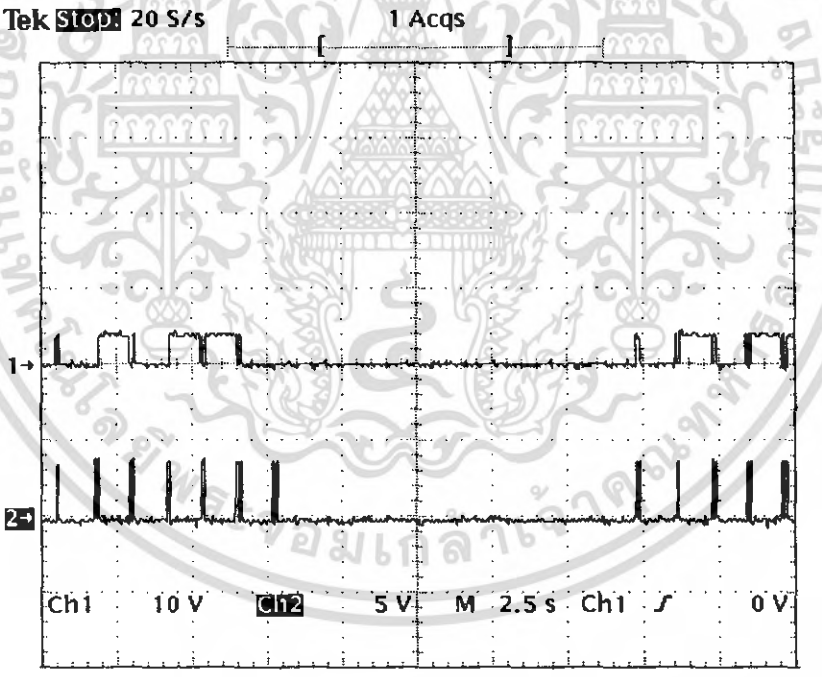


รูปที่ 4.51 แสดงสัญญาณทางด้านส่งของ RF Module TRW 2.4 กิกะเฮริ์ต
ในกรณีมีผู้บุกรุกประตูที่1,2,3,4,5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

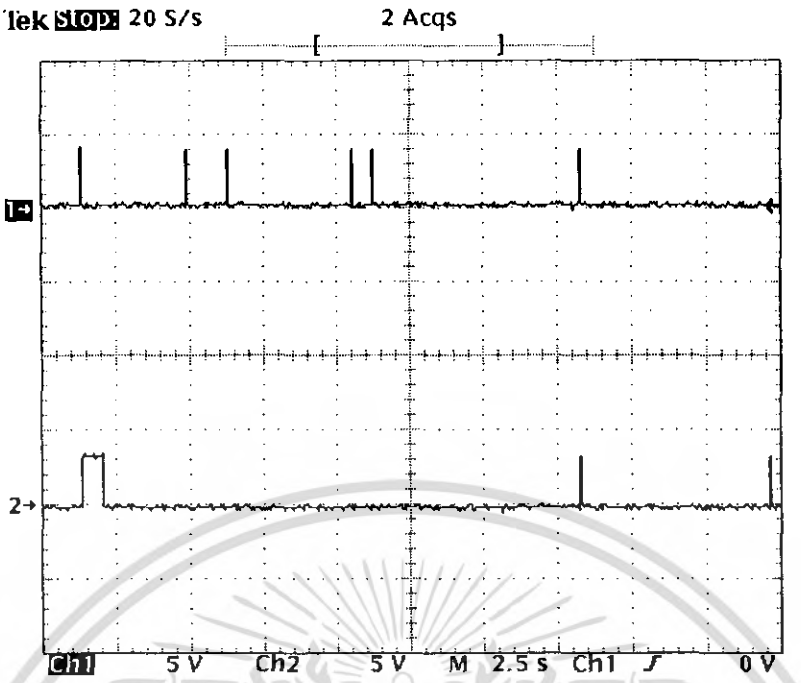


รูปที่ 4.52 แสดงสัญญาณทางด้านรับของ RF Module TRW 2.4 กิกะเฮริ์ต ในกรณีมีผู้บุกรุกประตูที่ 1,2,3,4,5



รูปที่ 4.53 แสดงสัญญาณทางด้านส่งของ RF Module TRW 2.4 กิกะเฮริ์ต ในกรณี LAMP 3 ปิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.54 แสดงสัญญาณทางด้านรับของ RF Module TRW 2.4 กิกะเฮริต์ ในกรณี LAMP 3 เปิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทสรุปและวิจารณ์

5.1 สรุปปัญหาที่พบในโครงงาน

โทรศัพท์เคลื่อนที่ที่นำมาใช้งานเป็นโทรศัพท์ยี่ห้อซีเมนต์รุ่น C35i เพราะจะมีขา Tx,Rx รองรับการสั่งงานจาก MCS-51 จึงเป็นการจำกัดการใช้งานโทรศัพท์เฉพาะรุ่น

ในการสั่งงานควบคุมอุปกรณ์ไฟฟ้าจากโทรศัพท์เคลื่อนที่นั้น จะใช้สัญญาณจากไอซี MT8870 ส่งสัญญาณให้กับ MCS – 51 แต่ค่าแรงดันที่ออกมาจากไอซี MT8870 มีค่าต่ำเกินไปจึงต้องพิจารณาค่าของความต้านทานในการใช้ เพราะถ้ามากไปก็จะไม่สามารถสั่งให้ MCS – 51 ทำงานได้

5.2 สรุปผลของการดำเนินโครงงาน

เมื่อมีผู้บุกรุกเข้ามาในบ้าน เซ็นเซอร์จะส่งสัญญาณให้ MCS - 51 สั่งงานให้โทรแจ้งเตือนและส่งข้อความสั้นไปยังโทรศัพท์เคลื่อนที่ปลายทาง แจ้งว่าเกิดเหตุผิดปกติที่ไหน เช่น มีผู้บุกรุก ประตูที่ 1 ก็จะทำการโทรไปแจ้งเพื่อบอกให้ทราบเบื้องต้นว่าถูกบุกรุก จากนั้นจะส่งข้อความสั้นว่า "Door 1!" และเมื่อมีผู้บุกรุกหน้าต่างที่ 1 จะส่งข้อความสั้นว่า "Window1!" ไปที่โทรศัพท์เคลื่อนที่ปลายทางเพื่อแจ้งให้ทราบ ซึ่งเราสามารถสั่งชุดควบคุมอุปกรณ์ไฟฟ้าภายในบ้านผ่านโทรศัพท์เคลื่อนที่ได้ด้วย

5.3 แนวทางการพัฒนาโครงงาน

เพื่อให้มีความปลอดภัยมากยิ่งขึ้นควรพัฒนาในด้านการส่งภาพหรือกล้องวงจรปิดผ่านไมโครคอนโทรเลอร์ Rabbit เพื่อให้สามารถตรวจสถานะภายในบ้านหรือรอบๆ บ้านได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนังสืออ้างอิง

[1] Z-World , " Dynamic C " [CD-ROM] , Z-World,1999.

[2] เรืองไกร รังสีพล , " เจาะระบบ TCP/IP ", บริษัท โปรวิชั่น จำกัด, 2001.

[3] สมยศ จุณณะปิยะ, "การประยุกต์ใช้งานไมโครคอนโทรลเลอร์" , คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.

[4] ชัยวัฒน์ ลิ้มพรจิตรวิไล, "เรียนรู้และปฏิบัติการไมโครคอนโทรลเลอร์" , บริษัท Innovative Experiment Press จำกัด , 2000

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#class auto
//............//
#define TCPCONFIG 1
#define TCP_BUF_SIZE 2048
#define HTTP_MAXSERVERS 1
#define MAX_TCP_SOCKET_BUFFERS 1
#define REDIRECTHOST          _PRIMARY_STATIC_IP
//#define REDIRECTHOST  "proxy.domain.com:1212"
//.................
#define DINBUFSIZE 15
#define DOUTBUFSIZE 15
//#define SSPEC_MAXSPEC 24
//............/*/
#define REDIRECTTO            "http://" REDIRECTHOST "/websun.shtml"
#memmap xmem
#use "dcrtcp.lib"
#use "http.lib"
//............//
#ximport "c:/DCRABBIT_8.01/2548/dooropen.gif"    dooropen_gif
#ximport "c:/DCRABBIT_8.01/2548/doorclose.gif"   doorclose_gif
#ximport "c:/DCRABBIT_8.01/2548/lock.gif"        lock_gif
#ximport "c:/DCRABBIT_8.01/2548/unlock.gif"      unlock_gif

//............//
#define PORT_A1   0xe000
#define PORT_B1   0xe001
#define PORT_C1   0xe002
#define PORT_CON1 0xe003
//............//
const HttpType http_types[] =
{
   { ".shtml", "text/html", shtml_handler}, // ssi
   { ".html", "text/html", NULL},           // html
   { ".cgi", "", NULL},                     // cgi
   { ".gif", "image/gif", NULL}
};
//............//
const HttpRealm admin =
      {
            "aaaaa","aaaaa","admin"
      };
//............//
char door1[15];
char door2[15];
char door3[15];
char door4[15];
char door5[15];
char door6[15];
char door7[15];
char door8[15];
char close1[15];
char close2[15];
char alarm3[15];
char alarm4[15];
char alarm5[15];
char alarm6[15];
char alarm7[15];
char alarm8[15];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
char light[15];

unsigned char censer,motorsw;
      void i8255_int1(void)
      {
            WrPortI ( SPCR, &SPCRShadow, 0x84 );
            // Disable slave port function
            WrPortI ( PEFR, &PEFRShadow, (PEFRShadow | 0x80));     // Set
PE7 to be external I/O Chip Select
            WrPortI ( PEDDR, &PEDDRShadow, (PEDDRShadow | 0x80)); // Set
PE7 as an output
            WrPortI ( IB7CR, &IB0CRShadow, 0x08 );
            WrPortE(PORT_CON1, NULL, 0x8B);                        // Set port
B1 and C1 to be input set port 8255 input/output
            WrPortE(PORT_CON2,NULL, 0x80);                         // Set
8255_2 to All output  set port 8255 input/output
      }
//.....................
void delay()
{
      int   i;
      i=0;

      while (i!=1) {
            costate {
                  waitfor(DelayMs(45));   // waits until 200 ms have
passed since
                  //printf("i = %d\n", i);
                  i++;
            }
      }
}
void delay2()
{
      int   i;
      i=0;

      while (i!=1) {
            costate {
                  waitfor(DelayMs(100));  // waits until 200 ms have
passed since
                  //printf("i = %d\n", i);
                  i++;
            }
      }
}
//............//
void doorscan(void)
{
                  if((censer|0xfe)==0xfe)
                 {strcpy(door1,"doorclose.gif");
               strcpy(close1,"lock.gif");
               strcpy(alarm1,"off.gif");
               }
              else
              {strcpy(door1,"dooropen.gif");
               strcpy(close1,"unlock.gif");
               strcpy(alarm1,"on.gif");
               }

         if((censer|0xfd)==0xfd)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        {strcpy(door2,"doorclose.gif");
      strcpy(close2,"lock.gif");
      strcpy(alarm2,"off.gif");
     }
     else
     {strcpy(door2,"dooropen.gif");
      strcpy(close2,"unlock.gif");
      strcpy(alarm2,"on.gif");
      }

if((censer|0xfb)==0xfb)
        {strcpy(door3,"doorclose.gif");
      strcpy(close3,"lock.gif");
      strcpy(alarm3,"off.gif");
      }
     else
     {strcpy(door3,"dooropen.gif");
      strcpy(close3,"unlock.gif");
      strcpy(alarm3,"on.gif");
      }

if((censer|0xf7)==0xf7)
        {strcpy(door4,"doorclose.gif");
      strcpy(close4,"lock.gif");
      strcpy(alarm4,"off.gif");
      }
     else
     {strcpy(door4,"dooropen.gif");
      strcpy(close4,"unlock.gif");
      strcpy(alarm4,"on.gif");
      }

if((censer|0xef)==0xef)
        {strcpy(door5,"doorclose.gif");
      strcpy(close5,"lock.gif");
      strcpy(alarm5,"off.gif");
      }
     else
     {strcpy(door5,"dooropen.gif");
      strcpy(close5,"unlock.gif");
      strcpy(alarm5,"on.gif");
     }

if((censer|0xdf)==0xdf)
        {strcpy(door6,"lightoff.gif");

     }
     else
     {strcpy(door6,"lighton.gif");

      }

if((censer|0xbf)==0xbf)
        {strcpy(door7,"lightoff.gif");

      }
     else
     {strcpy(door7,"lighton.gif");

     }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
          if((censer|0x7f)==0x7f)
                 {strcpy(door8,"lightoff.gif");

                  }
                else
                {strcpy(door8,"lighton.gif");

                }
}
//.................//
void  swscan(void)          //Check the door function
        {
        switch(motorsw)
          {
         case 0x7f:

            {
             if(strcmp(close1,"unlock.gif")==0)
               {
                      led1toggle();
                      }
                  }

                  break;
        case 0xbf:
            {
                        if(strcmp(close2,"unlock.gif")==0)
                                {
                         led2toggle();
                         }
                  }
                  break;
                  case 0xdf:
                                {
                                if(strcmp(close3,"unlock.gif")==0)
                                {
                               led3toggle();
                  }
                  }
               break;
                  case 0xef:
                                {
                                if(strcmp(close4,"unlock.gif")==0)
                  {
                        led4toggle();
                        }
                        }
                  break;
                   case 0xf7:
               {
               if(strcmp(close5,"unlock.gif")==0)
                                        {
                        led5toggle();
                        }
                     }
                  break;
              /*case 0xfb:
                        {
                              if(strcmp(close6,"unlock.gif")==0)
                              {
                        led6toggle();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                        }
                    }
                        break;
            case 0xfd:
              {
                              if(strcmp(close7,"unlock.gif")==0)
                                    {
                        led7toggle();
                        }
                    }
                        break;
            case 0xfe:
              {
                              if(strcmp(close8,"unlock.gif")==0)
                                    {
                       led8toggle();
                        }
                     }
                   break;*/
              }
        }


//.................//
int led1toggle(HttpState* state)
{
   if (strcmp(door1,"dooropen.gif")==0)
      {
                WrPortE(PORT_A2,NULL,0x04); //DOOR_1 LOCK
                delay();
               WrPortE(PORT_A2,NULL,0x01);
                delay2();
                //WrPortE(PORT_A2,NULL,0x00); // NORMAL MODE
                //delay();
        }
       else if(strcmp(door1,"doorclose.gif")==0)

        {
        WrPortE(PORT_A2,NULL,0x02); //DOOR_1 UNLOCK
                delay();
                WrPortE(PORT_A2,NULL,0x08);
                delay2();
                //WrPortE(PORT_A2,NULL,0x00); // NORMAL MODE
                //delay();
        }

   cgi_redirectto(state,REDIRECTTO);
   return 0;
}
int led2toggle(HttpState* state)
{

   if(strcmp(door2,"dooropen.gif")==0)
      {
                WrPortE(PORT_A2,NULL,0x40); //DOOR_2 LOCK
                delay();
                WrPortE(PORT_A2,NULL,0x10);
                delay();
                //WrPortE(PORT_A2,NULL,0x00); // NORMAL MODE
                //delay();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        }
       else if(strcmp(door2,"doorclose.gif")==0)

            {

          WrPortE(PORT_A2,NULL,0x20); //DOOR_2 UNLOCK
                    delay();
          WrPortE(PORT_A2,NULL,0x80);
                    delay();
                    //WrPortE(PORT_A2,NULL,0x00); // NORMAL MODE
                    //delay();
            }


      cgi_redirectto(state,REDIRECTTO);
      return 0;
}


int led3toggle(HttpState* state)
{

   if(strcmp(door3,"dooropen.gif")==0)
      {

            WrPortE(PORT_B2,NULL,0x04); //DOOR_3 LOCK
                    delay();
                    WrPortE(PORT_B2,NULL,0x01);
                    delay();
                    WrPortE(PORT_B2,NULL,0x00); // NORMAL MODE
                    delay();

       }
      else if(strcmp(door3,"doorclose.gif")==0)

            {
            WrPortE(PORT_B2,NULL,0x02); //DOOR_3 UNLOCK
                    delay();
                    WrPortE(PORT_B2,NULL,0x08);
                    delay();
                    WrPortE(PORT_B2,NULL,0x00); // NORMAL MODE
                    delay();
            }


      cgi_redirectto(state,REDIRECTTO);
      return 0;
}

int led4toggle(HttpState* state)
{

   if(strcmp(door4,"dooropen.gif")==0)
      {
        WrPortE(PORT_B2,NULL,0x40); //DOOR_4 LOCK
                    delay();
                    WrPortE(PORT_B2,NULL,0x10);
                    delay();
                    WrPortE(PORT_B2,NULL,0x00); // NORMAL MODE
                    delay();
             }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        else if(strcmp(door4,"doorclose.gif")==0)

              {
              WrPortE(PORT_B2,NULL,0x20); //DOOR_4 UNLOCK
                    delay();
                    WrPortE(PORT_B2,NULL,0x80);
                    delay();
                    WrPortE(PORT_B2,NULL,0x00); // NORMAL MODE
                    delay();
              }


       cgi_redirectto(state,REDIRECTTO);
       return 0;
}
int led5toggle(HttpState* state)
{

   if(strcmp(door5,"dooropen.gif")==0)
      {
                    WrPortE(PORT_C2,NULL,0x04); //DOOR_5 LOCK
                    delay();
                    WrPortE(PORT_C2,NULL,0x01);
                    delay();
                    WrPortE(PORT_C2,NULL,0x00); // NORMAL MODE
                    delay();
                      }
       else if(strcmp(door5,"doorclose.gif")==0)
         {
              WrPortE(PORT_C2,NULL,0x02); //DOOR_5 UNLOCK
                    delay();
                    WrPortE(PORT_C2,NULL,0x08);
                    delay();
                    WrPortE(PORT_C2,NULL,0x00); // NORMAL MODE
                    delay();
              }


       cgi_redirectto(state,REDIRECTTO);
       return 0;
}
int led6toggle(HttpState* state)
{

   if(strcmp(door6,"dooropen.gif")==0)
      {
                    WrPortE(PORT_C2,NULL,0x40); //DOOR_6 LOCK
                    delay();
                    WrPortE(PORT_C2,NULL,0x10);
                    delay();
                    WrPortE(PORT_C2,NULL,0x00); // NORMAL MODE
                    delay();
              }
       else if(strcmp(door6,"doorclose.gif")==0)

              {
              WrPortE(PORT_C2,NULL,0x20); //DOOR_6 UNLOCK
                    delay();
                    WrPortE(PORT_C2,NULL,0x80);
                    delay();
                    WrPortE(PORT_C2,NULL,0x00); // NORMAL MODE
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                delay();
            }

      cgi_redirectto(state,REDIRECTTO);
      return 0;
}
int led7toggle(HttpState* state)
{

   if(strcmp(door7,"dooropen.gif")==0)
      {
                WrPortE(PORT_A1,NULL,0x04); //DOOR_7 LOCK
                delay();
                WrPortE(PORT_A1,NULL,0x01);
                delay();
                WrPortE(PORT_A1,NULL,0x00); // NORMAL MODE
                delay();
   }   else if(strcmp(door7,"doorclose.gif")==0)
                {
           WrPortE(PORT_A1,NULL,0x02); //DOOR_7 UNLOCK
                delay();
                WrPortE(PORT_A1,NULL,0x08);
                delay();
                WrPortE(PORT_A1,NULL,0x00); // NORMAL MODE
                delay();
            }

          cgi_redirectto(state,REDIRECTTO);
      return 0;
}

int led8toggle(HttpState* state)
{

   if(strcmp(door8,"dooropen.gif")==0)
      {
           WrPortE(PORT_A1,NULL,0x40); //DOOR_8 LOCK
           delay();
           WrPortE(PORT_A1,NULL,0x10);
           delay();
           WrPortE(PORT_A1,NULL,0x00); // NORMAL MODE
           delay();

        }
 else if(strcmp(door8,"doorclose.gif")==0)

           {
           WrPortE(PORT_A1,NULL,0x20); //DOOR_8 UNLOCK
                delay();
                WrPortE(PORT_A1,NULL,0x80);
                delay();
                WrPortE(PORT_A1,NULL,0x00); // NORMAL MODE
                delay();
            }


      cgi_redirectto(state,REDIRECTTO);
return 0;
}
/*int led9toggle(HttpState* state)
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   if(strcmp(light,"lightoff.gif")==0)
      {
        WrPortI(SPCR,&SPCRShadow,0x84);
        BitWrPortI(PADR,&PADRShadow,1,1); //on
        WrPortI ( SPCR, &SPCRShadow, 0x84 );    // Disable slave port
function
              BitWrPortI(PADR,&PADRShadow,1,5);
                        //open
        strcpy(light,"lighton.gif");
         }
 else if (strcmp(light,"lighton.gif")==0)

           {
       WrPortI(SPCR,&SPCRShadow,0x84);
       BitWrPortI(PADR,&PADRShadow,0,1);  //off
       WrPortI ( SPCR, &SPCRShadow, 0x84 );     // Disable slave port
function
             BitWrPortI(PADR,&PADRShadow,0,5);
       strcpy(light,"lightoff.gif");
           }


     cgi_redirectto(state,REDIRECTTO);
return 0;
}*/

//.................................

const HttpSpec http_flashspec[] =
{
   { HTTPSPEC_FILE,  "/",              webs_html,        NULL, 0, NULL,
&admin},
   { HTTPSPEC_FILE,  "/websun.shtml",  webs_html,        NULL, 0, NULL,
&admin},
   { HTTPSPEC_FILE,  "/dooropen.gif",  dooropen_gif,     NULL, 0, NULL,
NULL},
   { HTTPSPEC_FILE,  "/doorclose.gif", doorclose_gif,    NULL, 0, NULL,
NULL},
   { HTTPSPEC_FILE,  "/lock.gif",      lock_gif,         NULL, 0, NULL,
NULL},
   { HTTPSPEC_FILE,  "/unlock.gif",    unlock_gif,       NULL, 0, NULL,
NULL},
   { HTTPSPEC_FILE,  "/kmitl.gif",     kmitl_gif,        NULL, 0, NULL,
NULL},
   { HTTPSPEC_FILE,  "/kmitl1.gif",    kmitl1_gif,       NULL, 0, NULL,
NULL},
   { HTTPSPEC_FILE,  "/button.gif",    button_gif,       NULL, 0, NULL,
NULL},
   { HTTPSPEC_FILE,  "/on.gif",        on_gif,           NULL, 0, NULL,
NULL},
   { HTTPSPEC_FILE,  "/off.gif",       off_gif,          NULL, 0, NULL,
NULL},
   { HTTPSPEC_FILE,  "/lighton.gif",   lighton_gif,      NULL, 0, NULL,
NULL},
   { HTTPSPEC_FILE,  "/lightoff.gif",  lightoff_gif,     NULL, 0, NULL,
NULL},

   { HTTPSPEC_VARIABLE, "close1", 0,   close1, PTR16,   "%s", NULL},
   { HTTPSPEC_VARIABLE, "close2", 0,   close2, PTR16,   "%s", NULL},
   { HTTPSPEC_VARIABLE, "close3", 0,   close3, PTR16,   "%s", NULL},
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   { HTTPSPEC_VARIABLE, "close4", 0,     close4, PTR16,   "%s", NULL},
   { HTTPSPEC_VARIABLE, "close5", 0,     close5, PTR16,   "%s", NULL},
   { HTTPSPEC_VARIABLE, "close6", 0,     close6, PTR16,   "%s", NULL},
   { HTTPSPEC_VARIABLE, "close7", 0,     close7, PTR16,   "%s", NULL},
   { HTTPSPEC_VARIABLE, "close8", 0,     close8, PTR16,   "%s", NULL},

   { HTTPSPEC_VARIABLE, "alarm1", 0,     alarm1, PTR16,   "%s", NULL},
   { HTTPSPEC_VARIABLE, "alarm2", 0,     alarm2, PTR16,   "%s", NULL},
   { HTTPSPEC_VARIABLE, "alarm3", 0,     alarm3, PTR16,   "%s", NULL},
   { HTTPSPEC_VARIABLE, "alarm4", 0,     alarm4, PTR16,   "%s", NULL},
   { HTTPSPEC_VARIABLE, "alarm5", 0,     alarm5, PTR16,   "%s", NULL},
   { HTTPSPEC_VARIABLE, "alarm6", 0,     alarm6, PTR16,   "%s", NULL},
   { HTTPSPEC_VARIABLE, "alarm7", 0,     alarm7, PTR16,   "%s", NULL},
   { HTTPSPEC_VARIABLE, "alarm8", 0,     alarm8, PTR16,   "%s", NULL},
   { HTTPSPEC_VARIABLE, "light",  0,     light,  PTR16,   "%s", NULL},

   { HTTPSPEC_FUNCTION, "/led1toggle.cgi", 0, led1toggle, 0, NULL, NULL},
   { HTTPSPEC_FUNCTION, "/led2toggle.cgi", 0, led2toggle, 0, NULL, NULL},
   { HTTPSPEC_FUNCTION, "/led3toggle.cgi", 0, led3toggle, 0, NULL, NULL},
   { HTTPSPEC_FUNCTION, "/led4toggle.cgi", 0, led4toggle, 0, NULL, NULL},
   { HTTPSPEC_FUNCTION, "/led5toggle.cgi", 0, led5toggle, 0, NULL, NULL},
   /*{ HTTPSPEC_FUNCTION, "/led6toggle.cgi", 0, led6toggle, 0, NULL,
NULL},
   { HTTPSPEC_FUNCTION, "/led7toggle.cgi", 0, led7toggle, 0, NULL, NULL},
   { HTTPSPEC_FUNCTION, "/led8toggle.cgi", 0, led8toggle, 0, NULL, NULL},
   { HTTPSPEC_FUNCTION, "/led9toggle.cgi", 0, led8toggle, 0, NULL,
NULL},*/
};
//..............//
void main()
{
unsigned char tmp1,tmp2;
strcpy(close1,"lock.gif");
strcpy(close2,"lock.gif");
strcpy(close3,"lock.gif");
strcpy(close4,"lock.gif");
strcpy(close5,"lock.gif");
strcpy(close6,"lock.gif");
strcpy(close7,"lock.gif");
strcpy(door8,"lightoff.gif");
strcpy(alarm1,"off.gif");
strcpy(alarm2,"off.gif");
strcpy(alarm3,"off.gif");
strcpy(alarm4,"off.gif");
strcpy(alarm5,"off.gif");
strcpy(alarm6,"off.gif");
strcpy(alarm7,"off.gif");
strcpy(alarm8,"off.gif");
strcpy(light,"lightoff.gif");
//.............//
censer=0x00;
motorsw=0x00;
serDopen(19200);
serDclose;
   i8255_intl();
   sock_init();
   http_init();
      tcp_reserveport(80);
           while(1)
               {
                        http_handler();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    tmp1=RdPortE(PORT_C1);// READ DATA FROM 8255_PORT
C1 TRW
                tmp2=RdPortE(PORT_B1);// READ DATA FROM 8255_PORT B1
Light
                //printf("%x ",tmp2);


                if (censer != tmp1)
                        { censer=tmp1;
                                        doorscan();}
                if (motorsw != tmp2)
                      { motorsw=tmp2;
                     swscan();}
                  // alarmscan();
                }
}
```

โปรแกรมใช้งาน MCS – 51 ตัวส่งและCENSER

```
/***************start TX#1***************/
/*########################################*/
/*  include */
/*########################################*/
#include <reg52.h>
#include <absacc.h>
#include <ctype.h>
#include <intrins.h>
#include <math.h>
#include <stdarg.h>
#include <stdio.h>
#include <stdlib.h>
#include "trw_2410_CH1.c"
/*########################################*/
/*          Main Program                  */
/*########################################*/

/********  IN  PUT   ####################*/
sbit sw1=P0^0;
sbit sw2=P0^1;
sbit sw3=P0^2;
sbit sw4=P0^3;
sbit sw5=P0^4;
sbit sw6=P0^5;
sbit sw7=P0^6;
sbit sw8=P0^7;
/*##############################*/
/*        out put lamp          */
/*##############################*/

sbit lamp1=P2^5;
sbit lamp2=P2^6;
sbit lamp3=P2^7;
void main (void)
{
      /*##########    out put sw=1#########*/
      unsigned char
Data_out[]={0x00,0x01,0x02,0x04,0x08,0x10,0x20,0x40,0x80};
      /*##############    out put sw=0 #####*/
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      unsigned char
Data_out2[]={0x03,0x05,0x06,0x07,0x09,0x0A,0x0B,0x0C,0x0D};
      P0=0x00;
      P1=0x00;
      P2=0x00;
      dmsec(50);
      Init();                          /*initial port TRW-2.4G*/
      dmsec(50);
    SetMode(1);             /* set trw out*/
      dmsec(50);
      serial_init();
      dmsec(50);
      while(1)
      {

                        P0=0xFF;
                        Chkring();
                        dmsec(10);
                        dtmf_int();
                        dmsec(10);
                        if(P1==0x03)
                        {
                              lamp1=0;
                              send(Data_out2[0]);
                              dmsec(2);
                        }
                          if(P1==0x02)
                         {
                              lamp1=1;
                              dmsec(2);
                                    if(lamp2==1)
                                          {
                                                send(Data_out[1]);
                                                dmsec(5);
                                          }
                                    else if(lamp2==0)
                                          {
                                                send(Data_out2[0]);
                                                dmsec(5);
                                          }
                        }
                        dmsec(10);


                        if(sw2==0)
                              {
                                    send(Data_out2[1]);
                              }
                        if(sw2==1)
                              {
                                    send(Data_out[2]);
                                    phone();
                              }

                        if(sw3==0)
                              {
                                    send(Data_out2[2]);
                              }

                         if(sw3==1)
                              {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                send(Data_out[3]);
                phone();
        }


if(sw4==0)
        {
                send(Data_out2[3]);
        }

if(sw4==1)
        {
                send(Data_out[4]);
                phone();
        }

if(sw5==0)
        {
                send(Data_out2[4]);
        }
if(sw5==1)
        {
                send(Data_out[5]);
                phone();
        }


if(sw6==0)
        {
                send(Data_out2[5]);
        }
if(sw6==1)
        {
                send(Data_out[6]);
                phone();
        }


if(sw7==0)
        {
                send(Data_out2[6]);
        }

if(sw7==1)
        {
                send(Data_out[7]);
                phone();
        }

if(sw8==0)
        {
                send(Data_out2[7]);
        }
if(sw8==1)
        {
                send(Data_out[8]);
                dmsec(200);
                sms1();
                dmsec(200);
                sms2();
                sms3();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                                    sms4();
                                    sms5();
                                    dmsec(200);
                                    if("ERROR")
                                          {
                                                lamp1=1;
                                                dmsec(200);
                                                lamp1=0;
                                                dmsec(200);
                                          }
                              }
      }
}
```

โปรแกรมย่อย TRW 2.4GHz

```
/***************TRW-24G_TX_2410_CH#1***************/
/*########################################*/
/*                Define I/O Ports        */
/*########################################*/
sbit CS           =   P2^0;
sbit CE           =   P2^1;
sbit Date         =   P2^2;
sbit CLK1         =   P2^3;
sbit DR1          =   P2^4;
//sbit CLK2       =   P1^1;
//sbit DR2        =   P1^2;
//sbit Dout2      =   P1^3;
int i=0,j=0;
/*########################################*/
/*                Delay mSec              */
/*########################################*/
void dmsec (unsigned int count)  /*Delay mSec Xtal=11.0592 Mhz*/
{
    unsigned int i;
    while(count)
      {
        i=225;
            while (i>0)
            i--;
        count--;
    }
}
/*########################################*/
/*                Wait                    */
/*########################################*/
void Wait(unsigned int x)
{
      unsigned int i;
      for (i=0;i<x;i++)
      {}
}
/*########################################*/
/*             Initial TRW-2.4G           */
/*########################################*/
void Init(void)
{
  CE    =  0;
  CS    =  0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  CLK1   =  0;
  Date   =  0;
  DR1    =  0;
}
/*########################################*/
/*                    Set clock                    */
/*########################################*/
void CLK(void)
{
      CLK1 = 0;
      dmsec(1);
      CLK1 = 1;
      dmsec(1);
}
/*########################################*/
/*                Write TRW-2.4G              */
/*########################################*/
void Write(unsigned char Data)
{
      unsigned char i;
      bit Out;
      for (i=0;i<8;i++)
      {
            Out = Data & 0x80;
            Date = Out;
            CLK();
            Data = Data << 1;
      }
}
/*########################################*/
/*                set Mode TRW-2.4G             */
/*########################################*/
void SetMode(unsigned char Mode)
{
      Wait(500);
      CE = 0;
      CS = 1;

      Write(0x8E); /* MSB(1)*/
      Write(0x08); /* MSB(0)*/
      Write(0x1C); /* MSB(0)*/

      Write(0x08); /* Length of Bit Ch 2 */
      Write(0x08); /* Length of Bit Ch 1 */

      Write(0xC0); /* Address 5 Byte Ch 2 */
      Write(0xAA);
      Write(0x55);
      Write(0xAA);
      Write(0x55);

      Write(0xAA); /* Address 5 Byte Ch 1 */
      Write(0x55);
      Write(0xAA);
      Write(0x55);
      Write(0xAA);

      Write(0xA3); /* Number of Address bit + CRC */
      Write(0x4F); /* 1 CH 250Kbps */

      if (Mode == 1)          /* Tx Mode */
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      { Write(0x14); }  /* Tx Mode 2410MHz */
      else                    /* Rx_Mode */
      { Write(0x15);          /* Rx Mode 2410MHz */
        Date=1; DR1=1; CE=1;
      }
      CS = 0;
      Wait(200);
}
/*########################################*/
/*                      Send data                        */
/*########################################*/
void send(unsigned char dat)
{
      Wait(500);
      CS = 0;
      CE = 1;

      Write(0xAA);       /*send ADDR CH1*/
      Write(0x55);
      Write(0xAA);
      Write(0x55);
      Write(0xAA);

      Write(dat);        /*send Data*/
      Wait(250);
      CLK1= 0;
      CE    = 0;
      Wait(250);
      dmsec(500);
}
////////////////Finish Program////////////////////
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้