สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบรักษาความปลอดภัยและอำนวยความสะดวกผ่านโทรศัพท์มือถือ SMART HOME

นายชวิส จิรวัฒน์ศิวาพร
นายฐิติพงษ์ นิ่มเจริญวรรณ
นายณัฐวุฒิ จันทรสิทธิผล

วท.
ช296ร
2547

เลขหมู่.....
เลขทะเบียน 61760
วัน,เดือน,ปี 21 ก.ค. 2549

b. 1160315x
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบรักษาความปลอดภัยและอำนวยความสะดวกผ่านโทรศัพท์มือถือ SMART HOME

โดย

นายชวิศ จิรวัฒน์ศิวาพร

นายฐิติพงษ์ นิ่มเจริญวรรณ

นายณัฐวุฒิ จันทรสิทธิผล

อาจารย์ที่ปรึกษา

อ.เจริญ วงษ์ชุ่มเย็น

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2547

ภาควิชา วิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบรักษาความปลอดภัยและอำนวยความสะดวกผ่านโทรศัพท์มือถือ SMART HOME

ผู้จัดทำ

1. นายชวิศ จิรวัฒน์ศิวาพร รหัสประจำตัว 44010096
2. นายฐิติพงษ์ นิ่มเจริญวรรณ รหัสประจำตัว 44010127
3. นายณัฐวุฒิ จันทรสิทธิผล รหัสประจำตัว 44010162

_____ อาจารย์ที่ปรึกษา

(อ.เจริญ วงษ์ชุ่มเย็น)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบรักษาความปลอดภัยและอำนวยความสะดวกผ่านโทรศัพท์มือถือ

นายชวิศ จิรวัฒน์ศิวาพร 44010096
นายฐิติพงษ์ นิ่มเจริญวรรณ 44010127
นายณัฐวุฒิ จันทรสิทธิผล 44010162
อ.เจริญ วงษ์ชุ่มเย็น อาจารย์ที่ปรึกษา
ปีการศึกษา 2547

บทคัดย่อ

ระบบรักษาความปลอดภัยและอำนวยความสะดวกผ่านโทรศัพท์มือถือ เป็นระบบที่พัฒนาขึ้นเพื่ออำนวยความสะดวกสบายให้กับผู้อยู่อาศัยในการควบคุมอุปกรณ์ไฟฟ้าต่างๆ ภายในบ้าน โดยสามารถตั้งเวลาควบคุมการเปิดปิดอุปกรณ์ล่วงหน้า ทำให้อุปกรณ์สามารถเปิดปิดได้เองตามตารางเวลาที่ตั้งไว้โดยอัตโนมัติ ซึ่งสามารถช่วยประหยัดพลังงานและลดค่าใช้จ่ายลงได้

นอกจากนี้ยังทำหน้าที่ในการรักษาความปลอดภัยภายในบ้านได้อีกด้วย โดยหากมีเหตุฉุกเฉินหรือสิ่งผิดปกติภายในบ้านเกิดขึ้น ระบบจะทำการเปิดสัญญาณเตือนภัยและส่งข้อความผ่านโทรศัพท์มือถือออกไปแจ้งเตือนผู้อยู่อาศัยทันที โดยสามารถใช้งานระบบผ่านทางโทรศัพท์มือถือหรือเว็บบราวเซอร์ ซึ่งจะทำการติดต่อไปยังเว็บเซอร์วิส เพื่อทำการสั่งงานไปยังชุดควบคุมอุปกรณ์ และเรียกดูข้อมูลของอุปกรณ์ต่างๆ รวมไปถึงการตั้งค่าการทำงานของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Smart Home

Chawit Jirawatsivaporn
Thitipong Nimjarunevan
Nuttawut Jantarasittiphol
Mr.Charoen Vongchumyen Advisor

ABSTRACT

The security system and making convenient via mobile phone is system that developed to make convenient for the residents to control the home appliances. This system can make schedules for controlling devices which can automatically turn on or turn off. So it can save cost and energy for the residents

Moreover, it can examine for the security that will alert to the residents, when the emergency happened, by sending short message and turn on siren. It's used via mobile phone or web browser that will connect to web service for sending command to control devices, viewing the device data and setting its system.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงได้เป็นอย่างดีด้วยคำปรึกษา คำแนะนำ และการคอยดูแลจากหลายฝ่ายด้วยกัน โดยเฉพาะอาจารย์ที่ปรึกษาที่ให้โอกาสข้าพเจ้าได้ทำปริญญานิพนธ์ฉบับนี้ซึ่งคอยให้ความเอาใจใส่ดูแลแนะนำและให้ความช่วยเหลือเสมอมา คือ อ.เจริญ วงษ์ชุ่มเย็น

นอกจากนี้ขอขอบคุณภาควิชาวิศวกรรมศาสตร์ที่ได้จัดสถานที่และสิ่งอำนวยความสะดวก อีกทั้งยังมีอินเตอร์เน็ตความเร็วสูง สำหรับการค้นคว้าหาความรู้ต่างๆ ซึ่งเป็นประโยชน์อย่างมากในการพัฒนาโครงงานนี้ และขอขอบคุณห้องฮาร์ดแวร์ที่ให้ที่อยู่สำหรับการทำงาน และเป็นที่ซื้อขนม เพื่อให้มีแรงกายในการทำงานต่อไป

และท้ายสุดขอขอบคุณบุคคลที่สำคัญที่สุดในชีวิตที่ทำให้ข้าพเจ้ามีวันนี้ นั่นคือ บิดา มารดา และบุคคลในครอบครัว อันเป็นที่เคารพรักยิ่ง ซึ่งได้เลี้ยงดูและคอยสั่งสอนข้าพเจ้ามาเป็นอย่างดี พร้อมให้โอกาสในการศึกษาอย่างเต็มที่ และยังให้ความรักและกำลังใจเสมอมา ข้าพเจ้าขอกราบขอบพระคุณมา ณ ที่นี้ด้วย

นายชวิศ จิรวัฒน์ศิวาพร
นายฐิติพงษ์ นิ่มเจริญวรรณ
นายณัฐวุฒิ จันทรสิทธิผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้าที่
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญภาพ	IX
สารบัญตาราง	XIV
บทที่ 1 บทนำ	**1**
1.1 ความสำคัญและที่มา	1
1.2 วัตถุประสงค์ของโครงงาน	1
1.3 ขอบเขตของโครงงาน	1
1.4 วิธีการดำเนินงาน	2
บทที่ 2 ทฤษฎีและหลักการ	**3**
2.1 แนวคิดของระบบควบคุมภายในบ้าน	3
2.1.1 แนวคิดและหลักการทั่วไป	3
2.1.2 แนวคิดและหลักการของระบบ Smart Home	3
2.2 เว็บเซอร์วิส (Web Service)	4
2.2.1 หลักการเว็บเซอร์วิส	5
2.2.2 คุณลักษณะของ Web Service	5
2.2.3 โครงสร้างของเว็บเซอร์วิส	6
2.3 SOAP	6
2.3.1 ส่วนประกอบของ SOAP	7
2.3.2 SOAP Fault Element	9
2.3.3 SOAP Encoding	10
2.3.4 SOAP สำหรับ RPC	11
2.4 XML	11
2.4.1 แนะนำ XML	12
2.4.2 โครงสร้างของ XML	13
2.5 J2ME	13
2.5.1 ส่วนการปรับแต่งใน จาวาทูเอ็มอี (Configuration)	14
2.5.2 เวอร์ชวลแมชชีนของจาวาทูเอ็มอี	15
2.5.3 โพรไฟล์ของจาวาทูเอ็มอี	15
2.5.4 มิดเล็ต (MIDlet)	16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.5 การใช้งาน API ระดับสูงในการพัฒนาส่วนติดต่อกับผู้ใช้ 17
2.6 ASP.NET 21
2.6.1 องค์ประกอบในการใช้งาน .NET เว็บเซอร์วิส 21
2.6.2 การสร้าง ASP.NET เว็บเซอร์วิส 21
2.6.3 การเรียกใช้งานเว็บเซอร์วิส 23
2.6.4 การจัดการสเตทในเว็บเซอร์วิส 23
2.6.5 แอพพลิเคชันอ็อบเจ็กต์ 24
2.6.6 เซสชั่นอ็อบเจ็กต์ 24
2.6.7 โหมดการจัดการเซสชั่นและการคอนฟิกค่า 25
2.7 ไมโครคอนโทรลเลอร์ 25
2.7.1 คุณสมบัติของไมโครคอนโทรเลอร์ MCS-51 25
2.7.2 โครงสร้างของไมโครคอนโทรลเลอร์ตระกูล MCS – 51 26
2.7.3 โครงสร้างภายในของ 8051 27
2.7.4 ผังเวลาของ CPU (CPU Timing) 30
2.7.5 การแบ่งประเภทของหน่วยความจำ 32
2.7.6 พอร์ตสื่อสารข้อมูลแบบอนุกรมใน MCS-51 33
2.8 รีเลย์ 35
2.8.1 หลักการทำงานของรีเลย์ 35
2.9 ไตรแอก 35
2.9.1 คุณสมบัติของไตรแอกและหลักการทำงาน 35
2.9.2 หลักการควบคุมทางเฟส (Phase Trigger) 36
2.10 พาสซีฟอินฟราเรดดีเท็กเตอร์(Passive Infrared detector - PIR) 37
2.10.1 หลักการทำงานของ PIR 37
2.11 เครื่องตรวจจับควัน(Smoke Detector) 38
2.12 ตัวแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล (Analog to Digital Convertor) 38
2.12.1 ทฤษฎีของตัวแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล 38
2.12.2 เทคนิคการแปลงสัญญาณแบบประมาณค่าหลายครั้ง (Successive Approximation Convertor) 39
2.13 ออปแอมป์ 40
2.13.1 ทฤษฎีของออปแอมป์ 40
2.13.2 คุณสมบัติทั่วไปของออปแอมป์ 40
2.13.3 การเปรียบเทียบแรงดันไฟฟ้า 40
2.14 ออปโต้คัปเปลอร์ (Opto-Coupler) 41
2.14.1 การทำงานของออปโต้คัปเปลอร์ (Opto-Coupler) 41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.15 คลื่นวิทยุ(RADIO FREQUENCY : RF) 42
2.15.1 ระบบการสื่อสารแบบอนาลอก 42
2.15.2 ระบบการสื่อสารแบบดิจิตอล 43
2.15.3 การส่งสัญญาณ 44
2.15.4 การสื่อสารข้อมูลดิจิตอล (Transmission of Digital Data) 45
บทที่ 3 โครงสร้างและการออกแบบ 48
3.1 ภาพรวมของระบบ 48
3.1.1 ระบบรักษาความปลอดภัยและอำนวยความสะดวกผ่านโทรศัพท์มือถือ 49
3.1.2 การติดต่อสื่อสารระหว่างสมาร์ทโฮมเซิร์ฟเวอร์กับบอร์ดควบคุม 49
3.1.3 โปรโตคอลการสื่อสาร 52
3.2 การออกแบบโปรแกรมแม่ข่าย (Smart Home Server) 53
3.2.1 กระบวนการเกี่ยวกับเฟรมข้อมูล 53
3.2.2 ระบบตารางเวลาทำงาน (Scheduler) 55
3.2.3 การแปลงข้อมูลเพื่อส่ง SMS ผ่านโมเด็มบนโทรศัพท์มือถือ 56
3.2.4 กระบวนการส่งข้อมูลควบคุมเครื่องใช้ไฟฟ้า 57
3.2.5 กระบวนการจับภาพจากกล้อง (Camera Capture) 58
3.2.6 กระบวนการกับเหตุการณ์และการตอบสนอง 58
3.2.7 การออกแบบฐานข้อมูล 59
3.3 การออกแบบโปรแกรมลูกข่ายบนโทรศัพท์มือถือ 61
3.3.1 เทคนิคหรือเทคโนโลยีที่ใช้ 61
3.3.2 เครื่องมือที่ใช้ในการพัฒนา 61
3.3.3 โครงสร้างและการทำงานของโปรแกรมลูกข่ายบนโทรศัพท์มือถือ 62
3.3.4 การติดต่อสื่อสารระหว่างโทรศัพท์มือถือกับเว็บเซอร์วิส 63
3.3.5 รูปแบบการในการติดต่อสื่อสารระหว่างเว็บเซอร์วิสและโทรศัพท์มือถือ 64
3.4 การออกแบบโปรแกรมลูกข่าย บนเว็บบราวเซอร์ 66
3.4.1 รูปแบบการในการติดต่อสื่อสารระหว่างเว็บเซอร์วิสและเว็บบราวเซอร์ 66
3.5 โครงสร้างการออกแบบในส่วนของฮาร์ดแวร์ 67
3.5.1 การออกแบบและระบบสมาร์ทโฮม 67
3.5.2 การทำงานของระบบ 69
3.5.3 รายละเอียดโครงสร้างทางฮาร์ดแวร์ 70
3.5.4 การออกแบบวงจรบอร์ดควบคุม 74
3.5.5 การออกแบบวงจรบอร์ดสวิตช์ 76
3.5.6 การออกแบบวงจรบอร์ดปรับระดับความสว่าง 79
3.5.7 การออกแบบวงจรบอร์ดระบบรักษาความปลอดภัย 82

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5.8 การออกแบบวงจรแหล่งจ่ายไฟ 83
3.5.9 การออกแบบวงจรรับ-ส่งคลื่นวิทยุ 84
3.6 การออกแบบโปรแกรมควบคุมฮาร์ดแวร์ 85
3.6.1 โปรแกรมควบคุมของไมโครคอนโทรลเลอร์มาสเตอร์ 85
3.6.2 โปรแกรมควบคุมของไมโครคอนโทรลเลอร์สเลฟ 86
3.6.3 โปรแกรมควบคุมการเปิด-ปิดอุปกรณ์ไฟฟ้า 86
3.6.4 โปรแกรมควบคุมความสว่างของหลอดไฟ 87
3.6.5 โปรแกรมควบคุมระบบรักษาความปลอดภัย 88
3.7 RF Protocol 90

บทที่ 4 การทดลองและทดสอบการทำงาน 92
4.1 เงื่อนไขและสภาวะในการทดลอง 92
4.2 การทดลองการเพิ่ม Interface Module เข้าสู่ระบบ 93
4.3 การทดลองการกำหนดค่าให้กับ Interface Module 93
4.4 การทดลองการนำ Interface Module ออกจากระบบ 94
4.5 การควบคุมการทำงานของ Switch Module ผ่านเว็บบราวเซอร์ 94
4.6 การควบคุมการทำงานของ Switch Module ผ่านโทรศัพท์มือถือ 96
4.7 การควบคุมการทำงานของ Switch Module แบบ Manual 97
4.8 การควบคุมการทำงานของ Dimmer Module ผ่านเว็บบราวเซอร์ 97
4.9 การควบคุมการทำงานของ Dimmer Module ผ่านโทรศัพท์มือถือ 98
4.10การทดลองปรับระดับความสว่างของ Dimmer Module แบบ Manual 99
4.11การควบคุมการทำงานของ Security Module ผ่านเว็บบราวเซอร์ 100
4.12 การควบคุมการทำงานของ Security Module ผ่านโทรศัพท์มือถือ 103
4.13 การทดสอบการทำงานของระบบ Macro 104
4.13.1 การทดสอบการเพิ่ม Macro 104
4.13.2 การทดสอบการลบ Macro 105
4.13.3 การทดสอบการแจ้งเตือนด้วย Macro 106
4.14 การทดสอบการทำงานของระบบตารางเวลา 106
4.14.1 การทดลองการเพิ่มตารางเวลาแบบ Once ผ่านเว็บบราวเซอร์ 106
4.14.2 การทดลองการเพิ่มตารางเวลาแบบ Once ผ่านโทรศัพท์มือถือ 107
4.14.3 การทดลองการเพิ่มตารางเวลาแบบ Daily ผ่านเว็บบราวเซอร์ 107
4.14.4 การทดลองการเพิ่มตารางเวลาแบบ Daily ผ่านโทรศัพท์มือถือ 108
4.14.5 การทดลองการเพิ่มตารางเวลาแบบ Weekly ผ่านเว็บบราวเซอร์ 109
4.14.6 การทดลองการเพิ่มตารางเวลาแบบ Weekly ผ่านโทรศัพท์มือถือ 110

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.14.7 การทดลองการเพิ่มตารางเวลาแบบ Monthly ผ่านเว็บบราวเซอร์ 110
4.14.8 การทดลองการเพิ่มตารางเวลาแบบ Monthly ผ่านโทรศัพท์มือถือ 111
4.14.9 การทดลองลบตารางเวลา ผ่านเว็บบราวเซอร์ 112
4.14.10 การทดลองลบตารางเวลา ผ่านโทรศัพท์มือถือ 113
4.14.11 การทดลองแก้ไขตารางเวลา ผ่านเว็บบราวเซอร์ 113
4.14.12 การทดลองแก้ไขตารางเวลา ผ่านโทรศัพท์มือถือ 114
4.15การทดสอบการใช้งานกล้อง 115
4.15.1 การทดสอบการลงทะเบียนกล้อง 115
4.15.2 การทดสอบการบันทึกภาพแบบด้วยตนเอง ผ่านเว็บบราวเซอร์ 116
4.15.3 การทดสอบการบันทึกภาพแบบด้วยตนเอง ผ่านโทรศัพท์มือถือ 117
4.15.4 การทดสอบการตั้งเวลาการบันทึกภาพ ผ่านเว็บบราวเซอร์ 118
4.15.5 การทดสอบการตั้งเวลาการบันทึกภาพ ผ่านโทรศัพท์มือถือ 119
4.15.6 การทดสอบการยกเลิกการใช้งานกล้อง 119
4.15.7 การทดสอบการตั้งค่ากล้อง 120
4.16 การทดสอบการตั้งค่า Server 120
4.17 การทดสอบการดูบันทึก Log 120
4.18 การทดลองการโหลดค่าสถานะล่าสุดให้ Interface Module 121
บทที่ 5 บทวิจารณ์และสรุป **123**
5.1 บทวิจารณ์และสรุป 123
5.2 ปัญหาที่เกิดและแนวทางการแก้ไข 124
5.3 แนวทางการพัฒนาต่อ 124
บรรณานุกรม **126**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

รูปที่	หน้าที่
รูปที่ 2-1 แสดงแนวคิดของระบบรักษาความปลอดภัยและอำนวยความสะดวกผ่านเครือข่าย	4
รูปที่ 2-2 แสดงลักษณะการให้บริการของเว็บเซอร์วิส	4
รูปที่ 2-3 แสดงโครงสร้างของ Web Service	5
รูปที่ 2-4 แสดงลักษณะการติดต่อใช้บริการ Web Service	6
รูปที่ 2-5 แสดงโครงสร้างของ SOAP	7
รูปที่ 2-6 แสดงแพลตฟอร์มของจาวาทู	13
รูปที่ 2-7 แสดงความสัมพันธ์ระหว่าง J2SE และคลาสไลบารีใน CDC และ CLDC	15
รูปที่ 2-8 แสดงวงจรการทำงานของ MIDlet	16
รูปที่ 2-9 แสดงการติดต่อจากไคลเอนต์ไปยังเว็บเซิร์ฟเวอร์ผ่านพร็อกซี่อ็อบเจ็กต์	23
รูปที่ 2-10 แสดงบล็อกไดอะแกรมของ MCS 8051	27
รูปที่ 2-11 แสดงโครงสร้าง port 0 (bit)	28
รูปที่ 2-12 แสดงโครงสร้าง port 1 (bit)	28
รูปที่ 2-13 แสดงโครงสร้าง port 2 (bit)	29
รูปที่ 2-14 แสดงโครงสร้าง port 3(bit)	29
รูปที่ 2-15 แสดงการทำงานของคำสั่งต่างๆ	30
รูปที่ 2-16 เป็นผังเวลาของสัญญาณซึ่งเกี่ยวข้องกับ Fetch	31
รูปที่ 2-17 แสดงโครงสร้าง port 2 (bit)	31
รูปที่ 2-18 ผังหน่วยความจำสำหรับโปรแกรมสำหรับเบอร์ 8051	32
รูปที่ 2-19 ผังหน่วยความจำสำหรับเก็บโปรแกรมสำหรับเบอร์ 8052	33
รูปที่ 2-20 แสดงการทำงานของรีเลย์	35
รูปที่ 2-21 สัญลักษณ์และวงจรพื้นฐานของไตรแอก	36
รูปที่ 2-22 การเปลี่ยนแปลงค่าของกำลังไฟฟ้าที่ป้อนให้แก่โหลด	36
รูปที่ 2-23 แสดงพาสซีฟอินฟราเรดดีเท็กเตอร์ (PIR)	38
รูปที่ 2-24 เครื่องตรวจจับควัน (Smoke Detector)	38
รูปที่ 2-25 แสดงบล็อกไดอะแกรมการแปลงค่าแบบ SA	39
รูปที่ 2-26 การทำงานโดยรวมของออปแอมป์	40
รูปที่ 2-27 วงจรเปรียบเทียบแรงดัน	40
รูปที่ 2-28 วงจรออปโต้คัปเปลอร์พื้นฐาน	41
รูปที่ 2-29 การกำหนดความกว้างของแถบ	42
รูปที่ 2-30 การ รับ – ส่ง ข้อมูลสัญญาณอนาลอก	42
รูปที่ 2-31 การ รับ – ส่ง ข้อมูลสัญญาณอนาลอก ระบบมอดูเลต	43
รูปที่ 2-32 การสื่อสารแบบดิจิตอล	43

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2-33 การสื่อสารทั้งแบบอนาลอกและดิจิตอล 44
รูปที่ 2-34 การส่งสัญญาณแบบทิศทางเดียว 44
รูปที่ 2-35 การส่งสัญญาณแบบผลัดกันรับ-ผลัดกันส่ง 44
รูปที่ 2-36 การส่งสัญญาณสองทิศทางที่เวลาเดียวกัน 44
รูปที่ 2-37 การสื่อสารข้อมูลแบบขนาน 45
รูปที่ 2-38 การสื่อสารข้อมูลแบบอนุกรม 46
รูปที่ 2-39 การส่งสื่อสารข้อมูลอนุกรมแบบอะซิงโครนัส 46
รูปที่ 2-40 การส่งสื่อสารข้อมูลอนุกรมแบบซิงโครนัส 47
รูปที่ 3-1 แสดงภาพรวมของระบบรักษาความปลอดภัยและ
อำนวยความสะดวกผ่านโทรศัพท์มือถือ 48
รูปที่ 3-2 แสดงโครงสร้างของเฟรมข้อมูลสำหรับสื่อสารกับบอร์ดควบคุม 49
รูปที่ 3-3 แสดงโปรโตคอล Stop and Wait with ARQที่มีการรอ Acknowledgement 52
รูปที่ 3-4 แสดงโปรโตคอล Stop and Wait with ARQ ที่มีการส่งข้อมูลซ้ำเวลาเกิด Timeout 52
รูปที่ 3-5 แสดงสถาปัตยกรรมทั่วไปของโปรแกรมแม่ข่าย 53
รูปที่ 3-6 แสดงกระบวนการการตัดเฟรมข้อมูล 54
รูปที่ 3-7 แสดงกระบวนการแยกแยะผลลัพธ์แก่ผู้ใช้งาน 54
รูปที่ 3-8 แสดงตัวอย่างโปรแกรมการทำงานโปรโตคอล Stop and wait with ARQ 55
รูปที่ 3-9 แสดงการสถาปนาการเชื่อมต่อ 55
รูปที่ 3-10 แผนผังแสดงการคำนวณเวลาทำงานของตารางเวลา 56
รูปที่ 3-11 การเปลี่ยนแปลงลำดับการทำงานของตารางเวลาแบบใช้คิว 56
รูปที่ 3-12 แสดงการใช้โทรศัพท์มือถือส่ง SMS 57
รูปที่ 3-13 แสดงกระบวนสร้างคำสั่งเพื่อส่งไปยังอุปกรณ์ 57
รูปที่ 3-14 แสดงโครงสร้างการจัดเก็บไฟล์ของภาพถ่ายจากกล้อง 58
รูปที่ 3-15 แสดงกระบวนการก่อนที่จะตรวจสอบเหตุการณ์ 59
รูปที่ 3-16 แสดงการทำงานของโปรแกรมในแต่ละส่วน 62
รูปที่ 3-17 แสดงการทำงานในส่วนของอุปกรณ์ไฟฟ้าและการรักษาความปลอดภัย 63
รูปที่ 3-18 แสดงการทำงานในส่วนของกล้องและการตั้งเวลาในการทำงาน 63
รูปที่ 3-19 แสดงการรับ – ส่งข้อมูลของแอพพลิเคชั่น 64
รูปที่ 3-20 แสดงถึงภาพรวมของฮาร์ดแวร์ 67
รูปที่ 3-21a แสดงการทำงานโดยรวมในรูปแบบมาสเตอร์สเลฟ 69
รูปที่ 3-21b แสดงการทำงานของระบบในรูปแบบมาสเตอร์สเลฟ 70
รูปที่ 3-22 แสดงถึงรายละเอียดของบอร์ดควบคุม 70
รูปที่ 3-23 แสดงถึงรายละเอียดของบอร์ดสวิตช์ 72
รูปที่ 3-24 แสดงถึงรายละเอียดของบอร์ดปรับระดับความสว่าง 72

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3-25 แสดงถึงรายละเอียดของบอร์ดระบบรักษาความปลอดภัย 74
รูปที่ 3-26 แสดงวงจรควบคุมของบอร์ดควบคุม 75
รูปที่ 3-27 บอร์ดควบคุม 75
รูปที่ 3-28 แสดงวงจรติดต่อสื่อสารแบบอนุกรมRS-232 76
รูปที่ 3-29 แสดงภาพรวมทั้งหมดของวงจรบอร์ดควบคุม 76
รูปที่ 3-30 บอร์ดสวิตซ์ 77
รูปที่ 3-31 แสดงวงจรของรีเลย์ 77
รูปที่ 3-32 แสดงวงจรของออปโตคัปเปลอร์ 77
รูปที่ 3-33 แสดงวงจรของรีเลย์ 78
รูปที่ 3-34 แสดงภาพรวมทั้งหมดของวงจรเปิด-ปิดอุปกรณ์ไฟฟ้า 78
รูปที่ 3-35 บอร์ดปรับระดับความสว่าง 79
รูปที่ 3-36 แสดงวงจรเปรียบเทียบแรงดันโดยใช้ออปแอมป์ 80
รูปที่ 3-37 แสดงวงจรควบคุมโหลดทางเอาท์พุตโดยใช้ไตรแอก 80
รูปที่ 3-38 แสดงวงจรแยกกราวน์แรงดันต่ำกับแรงดันสูงโดยใช้ออปโตคัปเปลอร์ 81
รูปที่ 3-39 แสดงวงจรแยกกราวน์แรงดันต่ำกับแรงดันสูงโดยใช้ออปโตคัปเปลอร์ 81
รูปที่ 3-40 แสดงวงจรอ่านค่าจากอนากลอกเป็นดิจิตอลโดยใช้ A/D 82
รูปที่ 3-41 แสดงภาพรวมทั้งหมดของวงจรปรับระดับความสว่าง 82
รูปที่ 3-42 บอร์ดระบบรักษาความปลอดภัย 83
รูปที่ 3-43 แสดงภาพรวมทั้งหมดของวงจรรักษาความปลอดภัย 83
รูปที่ 3-44 แสดงวงจรเร็กกูเลตเตอร์ 5 โวลต์ 84
รูปที่ 3-45 แสดงวงจรรับ-ส่งคลื่นวิทยุ 84
รูปที่ 3-46 โฟลว์ชาร์ตแสดงการทำงานของโปรแกรมไมโครคอนโทรลเลอร์มาสเตอร์ 85
รูปที่ 3-47 แสดงโฟลว์ชาร์ตการทำงานของโปรแกรมไมโครคอนโทรลเลอร์สเลฟ 86
รูปที่ 3-48 แสดงโฟลว์ชาร์ตการทำงานของโปรแกรมควบคุมการเปิด-ปิดอุปกรณ์ไฟฟ้า 87
รูปที่ 3-49 แสดงโฟลว์ชาร์ตการทำงานของโปรแกรมควบคุมความสว่างของหลอดไฟ 88
รูปที่ 3-50 แสดงโฟลว์ชาร์ตการทำงานของโปรแกรมระบบรักษาความปลอดภัย 89
รูปที่ 3-51 แสดงรูปแบบของ RF Protocol 90
รูปที่ 4-1 แสดงการลงทะเบียนของ Switch Module เข้าสู่ระบบ 93
รูปที่ 4-2 แสดงหน้าจอในการกำหนดชื่อให้อุปกรณ์ 93
รูปที่ 4-3 แสดงผลลัพธ์หลังจากกำหนดชื่อให้อุปกรณ์ 94
รูปที่ 4-4 แสดงภาพอุปกรณ์ที่ Fail ซึ่งจะไม่สามารถใช้งานได้ 94
รูปที่ 4-5 แสดงสถานะก่อนการสั่งเปิดอุปกรณ์ที่แอดแดรส FW 95
รูปที่ 4-6 แสดงไฟสถานะรีเลย์ของ Switch Module ก่อนการสั่งเปิดอุปกรณ์ที่แอดแดรส FW 95
รูปที่ 4-7 แสดงสถานะหลังการสั่งเปิดอุปกรณ์ที่แอดแดรส FW 95

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4-8 แสดงไฟสถานะรีเลย์ของ Switch Module หลังการสั่งเปิดอุปกรณ์ที่แอดแดรส FW 96
รูปที่ 4-9 รูปแสดงสถานะของอุปกรณ์ชื่อ Light 1 เมื่อกดปุ่ม Select 96
รูปที่ 4-10 แสดงการเปลี่ยนสถานะของอุปกรณ์ชื่อ Light 1 96
รูปที่ 4-11 แสดงการกดปุ่มเพื่อสั่งปิดอุปกรณ์ไฟฟ้าที่แชนแนล 4 97
รูปที่ 4-12 แสดงสถานะหลังการสั่งปิดอุปกรณ์ไฟฟ้าที่แชนแนล 4 97
รูปที่ 4-13 แสดงการระดับความสว่างก่อนการตั้งค่าอุปกรณ์ที่แอดแดรส XA 98
รูปที่ 4-14 แสดงการระดับความสว่างหลังการตั้งระดับความสว่างที่แอดแดรส XA 98
รูปที่ 4-15 แสดงระดับความสว่างของอุปกรณ์ชื่อ Room Dimmer เมื่อกดปุ่ม Select 98
รูปที่ 4-16 แสดงการปรับระดับความสว่างของอุปกรณ์ชื่อ Room Dimmer 99
รูปที่ 4-17 แสดงปุ่มปรับความสว่างแบบ Manual 99
รูปที่ 4-18 แสดงระดับความสว่างที่ 0 , 3 และ 7 ตามลำดับซ้ายไปขวา 100
รูปที่ 4-19 แสดงเหตุการณ์เมื่อมีคนเดินผ่าน PIR Sensor (Front Door) 100
รูปที่ 4-20 แสดงสถานะของเซ็นเซอร์ Front Door ที่ตรวจพบความผิดปรกติ 101
รูปที่ 4-21 แสดงเหตุการณ์เมื่อเกิดเพลิงไหม้ที่ Bedroom Smoke Detector 101
รูปที่ 4-22 แสดงสถานะของเซ็นเซอร์ Bedroom Smoke Detector ที่ตรวจพบความผิดปกติ 102
รูปที่ 4-23 แสดงการทำงานของไซเรนและไฟแสดงสถานะการทำงาน 102
รูปที่ 4-24 แสดงสถานะของเซ็นเซอร์หลังการตั้งค่าสู่สถานะปกติ 103
รูปที่ 4-25 แสดงการเปลี่ยนสถานะของอุปกรณ์ชื่อ Bedroom Door 103
รูปที่ 4-26 แสดงหน้าจอการเพิ่ม Macro 104
รูปที่ 4-27 แสดงผลลัพธ์หลังจากการเพิ่ม Macro 105
รูปที่ 4-28 แสดงหน้าจอรายการ Macro ที่มีอยู่ก่อนการลบออก 105
รูปที่ 4-29 แสดงหน้าจอรายการ Macro ที่มีอยู่หลังการลบออก 105
รูปที่ 4-30 แสดงข้อความ SMS 106
รูปที่ 4-31 แสดงหน้าจอการเพิ่มตารางเวลาแบบ Once 106
รูปที่ 4-32 แสดงผลลัพธ์หลังจากการเพิ่มตารางเวลาแบบ Once 107
รูปที่ 4-33 แสดงหน้าจอโทรศัพท์มือถือในการเพิ่มตารางเวลาแบบ Once 107
รูปที่ 4-34 แสดงหน้าจอการเพิ่มตารางเวลาแบบ Daily 108
รูปที่ 4-35 แสดงผลลัพธ์หลังจากการเพิ่มตารางเวลาแบบ Daily 108
รูปที่ 4-36 แสดงหน้าจอโทรศัพท์มือถือในการเพิ่มตารางเวลาแบบ Daily 109
รูปที่ 4-37 แสดงหน้าจอการเพิ่มตารางเวลาแบบ Weekly 109
รูปที่ 4-38 แสดงผลลัพธ์หลังจากการเพิ่มตารางเวลาแบบ Weekly 110
รูปที่ 4-39 แสดงหน้าจอโทรศัพท์มือถือในการเพิ่มตารางเวลาแบบ Weekly 110
รูปที่ 4-40 แสดงหน้าจอการเพิ่มตารางเวลาแบบ Monthly 111
รูปที่ 4-41 แสดงผลลัพธ์หลังจากการเพิ่มตารางเวลาแบบ Monthly 111

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4-42 แสดงหน้าจอโทรศัพท์มือถือในการเพิ่มตารางเวลาแบบ Monthly 112
รูปที่ 4-43 แสดงหน้าจอก่อนการลบรายการออกจากตารางเวลาแบบ Once 112
รูปที่ 4-44 แสดงหน้าจอหลังการลบรายการออกจากตารางเวลาแบบ Once 112
รูปที่ 4-45 แสดงการลบตารางเวลาผ่านโทรศัพท์มือถือ 113
รูปที่ 4-46 แสดงหน้าจอตารางเวลาแบบ Once ก่อนการแก้ไขตาราง 113
รูปที่ 4-47 แสดงหน้าจอการแก้ไขรายการแบบ Once ให้มีเวลาเริ่มทำงานที่ 20.00 น. 114
รูปที่ 4-48 แสดงหน้าจอตารางเวลาแบบ Once หลังการแก้ไขเวลา 114
รูปที่ 4-49 แสดงการแก้ไขค่าตารางเวลาผ่านโทรศัพท์มือถือ 115
รูปที่ 4-50 แสดงการลงทะเบียนกล้อง 116
รูปที่ 4-51 แสดงรายการกล้องที่ได้เพิ่มเข้าไป 116
รูปที่ 4-52 แสดงการจับภาพด้วยกล้อง 117
รูปที่ 4-53 แสดงการบันทึกภาพด้วยตนเอง 117
รูปที่ 4-54 แสดงการตั้งเวลาบันทึกภาพทุกๆ 15 นาทีก่อนการกด Set 118
รูปที่ 4-55 แสดงการตั้งเวลาบันทึกภาพทุกๆ 15 นาทีหลังการกด Set 118
รูปที่ 4-56 แสดงการตั้งเวลาการบันทึกภาพผ่านโทรศัพท์มือถือ 119
รูปที่ 4-57 แสดงหน้าจอก่อนการลบรายการกล้องออกจากตาราง 119
รูปที่ 4-58 แสดงหน้าจอหลังการลบรายการกล้องออกจากตาราง 119
รูปที่ 4-59 แสดงหน้าจอการกำหนดค่ากล้อง 120
รูปที่ 4-60 แสดงหน้าจอกำหนดค่าเซิร์ฟเวอร์ 120
รูปที่ 4-61 แสดงหน้าจอผลการบันทึก Log การทำงานของ Server 121
รูปที่ 4-62 แสดงสถานะตอน Switch Module ไม่ได้อยู่ในระบบ 121
รูปที่ 4-63 แสดงสถานะหลังจาก Switch Module เชื่อมต่อสู่ระบบอีกครั้ง 122

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้าที่
ตารางที่ 2-1 แสดง element ย่อยของ SOAP fault element	10
ตารางที่ 2-2 แสดงค่า faultcode	10
ตารางที่ 2-3 แสดงผลการทดลองการเปรียบเทียบแรงดัน	41
ตารางที่ 3-1 แสดงรายละเอียดของโปรโตคอลที่ใช้ในการติดต่อระหว่างเซิร์ฟเวอร์กับบอร์ดควบคุม	52
ตารางที่ 3-2 แสดงตาราง HDev	59
ตารางที่ 3-3 แสดงตาราง HChannel	59
ตารางที่ 3-4 แสดงตาราง iCameraDev	60
ตารางที่ 3-5 แสดงตาราง iDimDev	60
ตารางที่ 3-6 แสดงตาราง iSwDev	60
ตารางที่ 3-7 แสดงตาราง iSecurityDev	60
ตารางที่ 3-8 แสดงตาราง iSchedule	60
ตารางที่ 3-9 แสดงตาราง iEvent	61
ตารางที่ 3-10 แสดงตาราง iEventMonitor	61
ตารางที่ 3-11 แสดงรายละเอียดคำสั่งของ RF Protocol	91

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความสำคัญและที่มา

ระบบสมาร์ทโฮม(Smart Home) มาจากแนวคิดที่จะทำสิ่งต่างๆ ภายในบ้านให้สามารถควบคุมและจัดการได้ง่ายขึ้นจนกระทั่งสามารถทำงานได้อย่างอัตโนมัติและเป็นไปตามความต้องการของผู้อยู่อาศัย ในปัจจุบัน โดยส่วนใหญ่ผู้ใช้งานระบบสามารถสั่งงานผ่านทางหน้าเว็บ หรือทางระบบโทรศัพท์บ้าน ซึ่งในบางครั้งอาจไม่ครอบคลุมการใช้งานในบางอย่างหรืออาจไม่สะดวกในการใช้งานในบางรูปแบบ

การใช้งานโทรศัพท์มือถือในปัจจุบันมีปริมาณมากขึ้นอย่างมาก ซึ่งไม่ได้ทำหน้าที่เป็นเพียงอุปกรณ์สื่อสารที่ใช้ในการสนทนาเพียงอย่างเดียว แต่ยังสามารถนำมาทำอย่างอื่นได้อีกมากมาย เช่น เตือนการนัดหมาย เล่นเกมส์ ท่องอินเตอร์เน็ต ซึ่งเป็นผลจากการพัฒนาที่เป็นไปอย่างรวดเร็ว ดังนั้นจึงได้เกิดแนวคิดในการนำโทรศัพท์มือถือมาประยุกต์ใช้งานในด้านอื่น อย่างเช่นการนำมาใช้ในการควบคุมอุปกรณ์ไฟฟ้าต่างๆ เช่น หลอดไฟ โทรทัศน์ เป็นต้น

ดังนั้นโครงงานนี้จึงได้พัฒนาระบบอำนวยความสะดวกและรักษาความปลอดภัยผ่านโทรศัพท์มือถือขึ้นมาเพื่อที่จะเพิ่มความสะดวกสบายแก่ผู้ใช้งานมากขึ้น อีกทั้งเป็นการเพิ่มช่องทางในการใช้งานอีกแบบหนึ่ง

1.2 วัตถุประสงค์ของโครงงาน

1. เพื่อศึกษาและออกแบบระบบรักษาความปลอดภัยและอำนวยความสะดวกผ่านโทรศัพท์มือถือ
2. เพื่อศึกษา ออกแบบ และพัฒนาแอพพลิเคชั่นแบบไคลเอนต์ เซิร์ฟเวอร์
3. เพื่อศึกษา ออกแบบ และพัฒนาแอพพลิเคชั่นบนโทรศัพท์มือถือ
4. เพื่อศึกษา ออกแบบ และพัฒนาการรับ-ส่ง ข้อมูลแบบไร้สาย
5. เพื่อสร้างระบบที่เหมาะกับการใช้งานภายในบ้าน

1.3 ขอบเขตของโครงงาน

เพื่อออกแบบและพัฒนาระบบ ที่สามารถนำไปประยุกต์ใช้งาน และติดตั้งกับบ้านทั่วๆไป โดยจะเป็นระบบที่ประกอบด้วยชุดรักษาความปลอดภัยและชุดอำนวยความสะดวก ซึ่งสามารถควบคุมและติดต่อสื่อสารกันได้แบบไร้สาย และสามารถเข้าถึงข้อมูลได้ผ่านทางโทรศัพท์มือถือและเว็บบราวเซอร์ โดยจะประกอบด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชุดอำนวยความสะดวก ประกอบด้วย

- ชุดสวิตซ์ควบคุมอุปกรณ์ไฟฟ้า จะทำหน้าที่ในการสั่งงานเปิด-ปิด อุปกรณ์ไฟฟ้า
- ชุดปรับระดับความสว่างของหลอดไฟ สามารถควบคุมความสว่างของหลอดไส้ได้

ชุดรักษาความปลอดภัย ประกอบด้วย

- ชุดป้องการการบุกรุกและอัคคีภัย โดยจะมีเซ็นเซอร์ในการการตรวจจับการผ่านบริเวณ และการตรวจจับเมื่อระบบพบความผิดปกติเกิดขึ้นจะทำการเตือนภัยให้ผู้ใช้ทราบโดยการเปิดสัญญาณไซเรน หรือแจ้งเตือนผ่านทาง SMS
- ชุดกล้องรักษาความปลอดภัย จะสามารถบันทึกภาพตามเวลาที่ได้กำหนดไว้ หรือจะเรียกดูภาพ ณ เวลาขณะนั้นด้วยตนเอง

โดยทั้งอุปกรณ์ไฟฟ้าและเซ็นเซอร์จะสามารถกำหนดตารางเวลาในการทำงานได้ อีกทั้งยังสามารถทำงานได้ในแบบ Macro คือเมื่อมีเหตุการณ์ใดๆ เกิดขึ้นกับอุปกรณ์หนึ่ง จะส่งผลไปถึงอุปกรณ์อื่นๆ เช่น เมื่อมีการเปิดไฟในห้องนอนจะทำให้โทรทัศน์ปิดและไฟในห้องนั่งเล่นดับลง เป็นต้น และเมื่อเกิดปัญหาไฟฟ้าดับ อุปกรณ์ยังคงสามารถทำงานตามที่โปรแกรมไว้ได้ต่อเนื่องหลังจากมีไฟฟ้า

1.4 วิธีดำเนินงาน

เริ่มจากการค้นคว้าและรวบรวมข้อมูลต่างๆ เกี่ยวกับระบบรักษาความปลอดภัยและอำนวยความสะดวกผ่านโทรศัพท์มือถือ และส่วนที่เกี่ยวข้อง จากนั้นทำการศึกษาภาษาที่ใช้ในการเขียนโปรแกรม และโปรแกรมต่างๆ ที่ต้องใช้ในการพัฒนา หลังจากได้ทำการศึกษาทฤษฎีต่างๆ แล้วจะแบ่งงานออกเป็น 3 ส่วนหลักๆ คือ

1. ส่วนของการพัฒนาโปรแกรมแม่ข่าย โดยจะเป็นศูนย์กลางในการควบคุมการทำงานระหว่างส่วนที่ 1 และส่วนที่ 2
2. ส่วนของการพัฒนาโปรแกรมลูกข่ายบนโทรศัพท์มือถือ และเว็บบราวเซอร์เพื่อให้ผู้ใช้ใช้งานในการควบคุมและตรวจสอบอุปกรณ์ไฟฟ้า
3. ส่วนของการพัฒนาชุดรักษาความปลอดภัยและชุดอำนวยความสะดวก ซึ่งรับคำสั่งมาจากส่วนที่ 1 แล้วไปทำการควบคุมอุปกรณ์ไฟฟ้าต่างๆ และตรวจสอบการทำงานของชุดเซ็นเซอร์

โดยการพัฒนาจะเริ่มทำการพัฒนาแต่ละส่วนไปพร้อมๆ กัน จากนั้นจึงทำการทดสอบการทำงานในแต่ละส่วน แล้วจึงรวมแต่ละส่วนเข้าด้วยกันเพื่อทำการทดสอบการทำงานทั้งระบบ แล้วจึงทำการแก้ไขข้อบกพร่องต่างๆ เพื่อให้ระบบทำงานได้อย่างสมบูรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการ

2.1 แนวคิดของระบบควบคุมภายในบ้าน

2.1.1 แนวคิดและหลักการทั่วไป

ตั้งแต่อดีตถึงปัจจุบันสิ่งที่เราพบเห็นอยู่เสมอก็คือการวิวัฒนาการของเทคโนโลยี สิ่งที่เป็นปัจจัยทำให้เกิดการพัฒนานั้นก็คือความต้องการของมนุษย์เพื่อเพิ่มความสะดวกสบายและความปลอดภัยในชีวิตและทรัพย์สิน ดังจะเห็นได้จากอดีตการซักผ้าเราใช้หิน และไม้ทุบผ้าให้สะอาด ต่อมามีการพัฒนาเครื่องมือเครื่องใช้ให้มีความสามารถมากขึ้น ลดเวลาในการทำงานลง และได้ผลลัพธ์ที่ดีขึ้น เป็นลำดับ

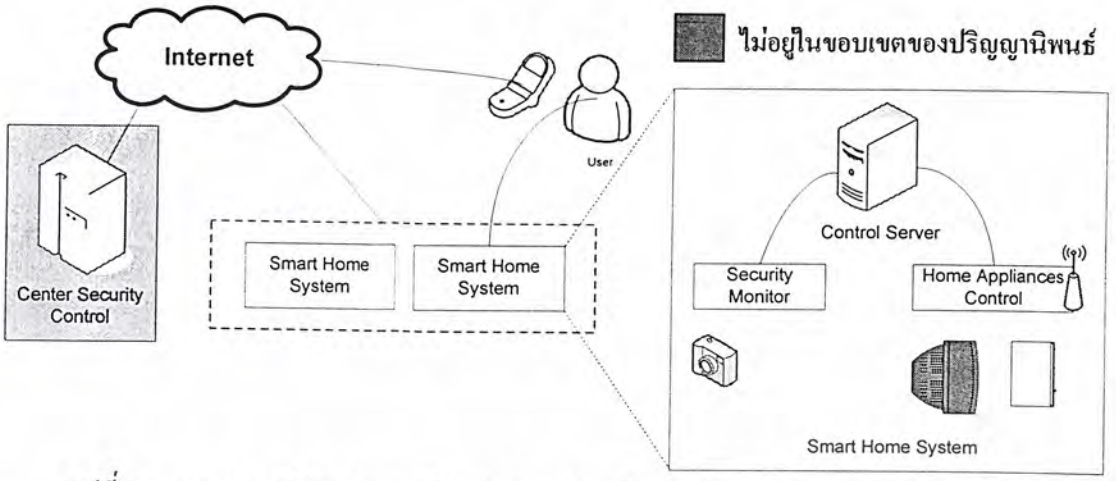
ระบบสมาร์ทโฮม (Smart Home) มาจากแนวคิดจะทำสิ่งต่าง ๆ ภายในบ้าน ให้สามารถควบคุมจัดการได้ง่ายขึ้น จนกระทั่งสามารถทำงานได้อย่างอัตโนมัติ เป็นไปตามความต้องการของผู้อยู่อาศัย และเพิ่มมาตรฐานความเป็นอยู่ในด้านต่าง ๆ ให้ดีขึ้น ทั้งในด้านชีวิต ทรัพย์สิน ความปลอดภัย เป็นต้น ตัวอย่างเช่น ตอนเช้าตรู่ก่อนที่ผู้อยู่อาศัยจะตื่น แอร์จะถูกลดอุณหภูมิเพื่อประหยัดไฟฟ้า กาต้มน้ำในห้องครัวเริ่มทำงานเพื่อเตรียมพร้อมสำหรับกาแฟอุ่นๆ เมื่อถึงเวลา 6.00 น. ไฟในห้องนอนสว่างขึ้นสลัวๆ บานเกร็ดเปิดเพื่อรับลมยามเช้า ไฟประตูรั้วปิดเพื่อประหยัดพลังงาน สปริงเกอร์เริ่มทำการรดน้ำต้นไม้ แต่ถ้าเมื่อคืนฝนตกซึ่งตรวจจับได้จากเซ็นเซอร์ (Sensor) สปริงเก้อก็จะไม่ทำงานเพื่อเป็นการประหยัดน้ำ เป็นต้น

2.1.2 แนวคิดและหลักการของระบบ Smart Home

ระบบรักษาความปลอดภัยและอำนวยความสะดวกผ่านเครือข่าย (Smart Home) จะเน้นที่ระบบควบคุมดูแล (Control and monitor system) ที่รวมเอาความสามารถด้านเครือข่ายและการควบคุมระยะไกลให้สามารถใช้งานได้กับอุปกรณ์ที่หลากหลาย และเอื้ออำนวยในการผสานเข้ากับระบบควบคุมอื่นๆ อันจะนำมาถึงการพัฒนาที่มีความซับซ้อนยิ่งขึ้นในอนาคต ดังเช่น ระบบรักษาความปลอดภัยกลาง (Center Security Control) ที่บริการการดูแลความปลอดภัยให้กับลูกบ้านที่เป็น Smart Home โดยถ้าตรวจความผิดปรกติของลูกบ้าน ก็จะสามารถจะควบคุมดูแลอีกชั้นหนึ่ง ซึ่งช่วยให้มีความปลอดภัยมากยิ่งขึ้น เช่น การปิดกั้นทางออกของโจรจากบริเวณ การแจ้งสัญญาณเตือนภัยแก่บริเวณใกล้เคียง เป็นต้น

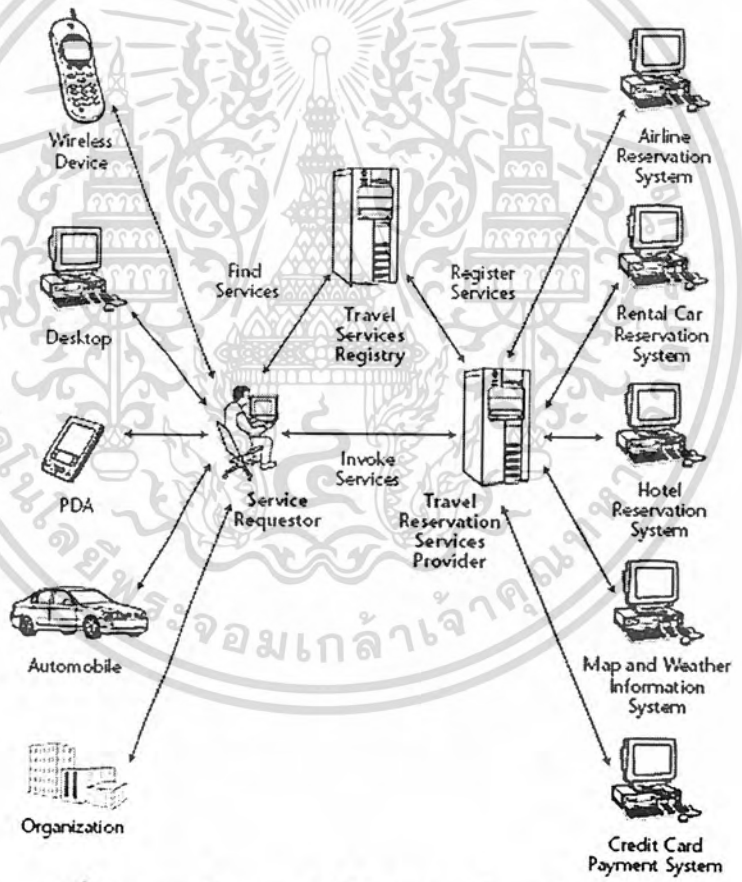
จากแนวคิดที่กล่าวข้างต้น ในขั้นแรกผู้พัฒนาจะพัฒนาระบบที่เน้นการควบคุมภายในบ้านแต่ละหลังให้มีความสามารถเพียงพอที่จะทำงานได้เองโดยลำพัง การวางเครือข่ายของการสื่อสารของอุปกรณ์เครื่องใช้ไฟฟ้า การเชื่อมต่อบริการจากภายนอก การให้บริการต่าง ๆ และส่วนในการใช้งานของระบบผ่านอุปกรณ์ที่แพร่หลาย เช่น โทรศัพท์มือถือ เว็บบราวเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-1 แสดงแนวคิดของระบบรักษาความปลอดภัยและอำนวยความสะดวกผ่านเครือข่าย

2.2 เว็บเซอร์วิส (Web Service)



รูปที่ 2-2 แสดงลักษณะการให้บริการของเว็บเซอร์วิส

เว็บเซอร์วิส คือ แอพลิเคชัน หรือ โปรแกรมที่ทำงานอย่างใดอย่างหนึ่งในลักษณะให้บริการ (Service) ที่จะถูกเรียกใช้งานจากแอพลิเคชันอื่นๆ ในรูปแบบ RPC (Remote Procedure Call) หรือระบบสั่งงานระยะไกล โดยการให้บริการจะมีเอกสาร ที่อธิบายคุณสมบัติของบริการกำกับไว้ มีภาษาที่ถูกใช้เป็นสื่อในการแลกเปลี่ยนคือ XML ทำให้ผู้ใช้สามารถเรียกใช้ส่วนประกอบใดๆ ใน แพลตฟอร์ม ใด ๆ ก็ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บนโปรโตคอล HTTP สำหรับ World Wide Web อันเป็นช่องทางที่ได้รับการยอมรับทั่วโลก ในการติดต่อสื่อสารกันระหว่าง แอพลิเคชัน กับ แอพลิเคชัน

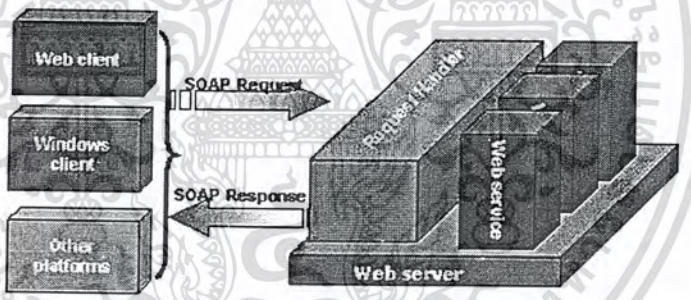
เว็บเซอร์วิสมีพื้นฐานมาจากแนวคิดของสถาปัตยกรรมการให้บริการหรือเอสโอเอ (SOA-Service-oriented architecture) เป็นแนวคิดใหม่ของการระบบแบบกระจายศูนย์กลาง (Distributed computing) ซึ่งทำให้ซอฟท์แวร์คอมโพแนนท์ (Software components) ฟังก์ชัน, ออปเจค สามารถใช้งานร่วมกันได้จากระบบที่สถาปัตยกรรมมีความแตกต่างกัน โดยใช้ประโยชน์จากอินเทอร์เน็ตเป็นโปรตอลคอลเพื่อค้นหา, ลงทะเบียนหรือการเรียกบริการนั้นๆ

2.2.1 หลักการเว็บเซอร์วิส

เว็บเซอร์วิส (Web Service) คือ การติดต่อระหว่างเครื่องคอมพิวเตอร์ที่กล่าวถึงกลุ่มของ Operation ที่เป็นการเข้าถึงผ่านเครือข่ายโดยอยู่ภายใต้มาตรฐานของ XML massaging หรือเราอาจกล่าวได้ว่าความหมายโดยรวมแล้ว เว็บเซอร์วิส หมายถึง บริการที่ให้โปรแกรมใช้ติดต่อระหว่างกันผ่านเครือข่าย

WSDL (Web Service Description Language) คือข้อกำหนดที่ใช้สร้างเอกสารสำหรับอธิบายรายละเอียดของ Web Service ซึ่งได้แก่ ชื่อของ Web service , URI, ฟังก์ชันการทำงาน ฯลฯ

UDDI (Universal Description, Discovery, and Integration) คือเป็น Directory สำหรับค้นหาบริการของ web service ทั้งนี้เพื่อให้ web service สามารถถูกค้นหาได้จาก internet เพื่อใช้งาน



รูปที่ 2-3 แสดงโครงสร้างของ Web Service

2.2.2 คุณลักษณะของ Web Service

- XML Base
- Message Base
- ไม่ขึ้นกับ Programming Language
- สามารถสร้างบริการใหม่ โดยการรวมหลายๆบริการ (aggregation)
- ใช้งานได้ผ่านเครือข่ายอินเทอร์เน็ต
- ใช้มาตรฐานสากล
- รายละเอียดการทำงานของ web service จะถูกซ่อนไว้ โดยผู้ใช้จะมองเห็นแค่รูปแบบและลักษณะการติดต่อที่ผู้ให้บริการประกาศไว้เท่านั้น

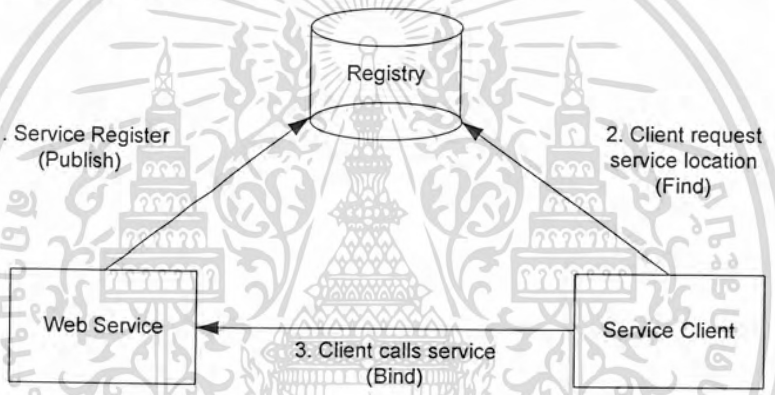
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ซอฟต์แวร์ที่ทำงานบนระบบ web service สามารถนำมาแก้ไขรายละเอียดภายในได้ โดยไม่ส่งผลกระทบเป็นลูกโซ่ และผู้ใช้ปลายทางก็ไม่จำเป็นต้องโหลดซอฟต์แวร์ติดตัวเกินความจำเป็น

2.2.3 โครงสร้างของเว็บเซอร์วิส

ประกอบไปด้วยส่วนสำคัญ 3 ส่วนคือ ผู้ให้บริการ (Service Provider),ผู้ขอบริการ (Service Requestor),ตัวแทนของผู้ให้บริการ (Service Register) โดยทั้ง 3 ส่วนจะติดต่อกันภายใต้เงื่อนไขดังนี้

1. ผู้ให้บริการการทำการประกาศบริการไปยังตัวแทนของผู้ให้บริการ ซึ่งจะถูกบันทึกไว้ใน Service Directory ของตัวแทนผู้ให้บริการ
2. ผู้ขอใช้บริการจะทำการค้นหาบริการจากตัวแทนของผู้ให้บริการ
3. เมื่อพบบริการที่ต้องการแล้ว ผู้ขอใช้บริการจะทำการติดต่อไปยังผู้ให้บริการ โดยร้องขอบริการไปยังผู้ให้บริการนั้น



รูปที่ 2-4 แสดงลักษณะการติดต่อใช้บริการ Web Service

2.3 SOAP

SOAP (Simple Object Access Protocol) เป็นโพรโตคอลที่ใช้ XML เป็นพื้นฐาน เพื่อให้ซอฟต์แวร์คอมโพเนนต์ และแอพพลิเคชันสามารถติดต่อกันผ่าน HTTP ซึ่งเป็นมาตรฐานอินเตอร์เน็ตโพรโตคอลได้

เนื่องจากการสื่อสารระหว่างแอพพลิเคชันจำเป็นต่อการพัฒนาแอพพลิเคชันบนอินเตอร์เน็ต เป็นอย่างมาก แต่แอพพลิเคชันแบบกระจายกันทำงาน (distributed application) ในปัจจุบันใช้ Remote Procedure Call (RPC) สื่อสารกันระหว่างออบเจ็กต์ เช่น DCOM และ CORBA ซึ่งไม่ได้ใช้พอร์ตเดียวกับ HTTP ที่เป็นพอร์ตมาตรฐานสำหรับให้บริการเว็บ ดังนั้น RPC จึงนำมาปรับใช้กับอินเตอร์เน็ตได้ยาก และมีปัญหาทางด้านความปลอดภัย ไฟร์วอลล์และพร็อกซีเซิร์ฟเวอร์จะไม่ยอมให้ส่งข้อมูลชนิดนี้ได้ตามปกติ

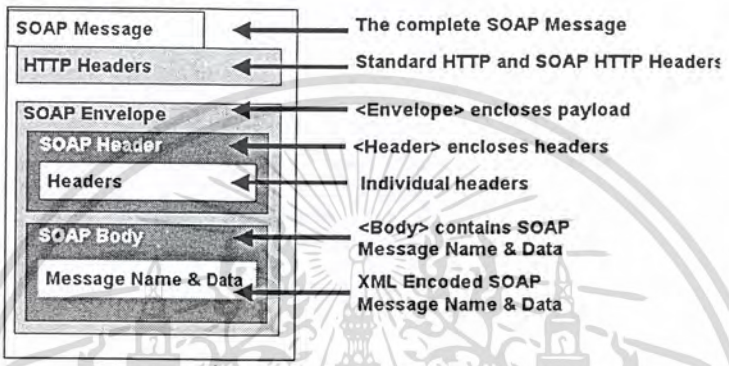
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีที่ดีกว่าคือใช้ HTTP เพราะเป็นที่ยอมรับโดยอินเตอร์เน็ตบราวเซอร์ และเซิร์ฟเวอร์ทุกชนิด ซึ่ง SOAP ถูกสร้างมาเพื่อใช้ในกรณีนี้

ข้อดีของ SOAP ก็คือ SOAP ไม่ขึ้นกับคอมโพเนนต์เทคโนโลยี และภาษาการเขียนโปรแกรมใดๆ สามารถเขียนได้ง่ายและขยายเพิ่มเติมได้

2.3.1 ส่วนประกอบของ SOAP

SOAP เมสเซจใช้ไวยากรณ์ของ XML ในการสร้าง ประกอบด้วย 3 อีลีเมนต์มาตรฐาน คือ SOAP Envelope, SOAP Header และ SOAP Body



รูปที่ 2-5 แสดงโครงสร้างของ SOAP

SOAP เมสเซจ จะต้องเป็นไปตามกฎนี้

- Envelope เป็นอีลีเมนต์ที่อยู่บนสุด ต้องมีอีลีเมนต์นี้เสมอ
- Header อาจจะมีหรือไม่มีก็ได้ แต่ถ้ามีต้องเป็น child element แรกของ Envelope
- Body ต้องเป็น child element แรกของ envelope หรือ header

ตัวอย่างของ SOAP เมสเซจ ตัวอย่างนี้คือ GetLastTradePrice SOAP Request ที่ส่งไปยัง StockQuote service ประกอบด้วยสตริงพารามิเตอร์ symbol และคืนค่า float กลับมากับ SOAP response SOAP envelope element อยู่ที่จุดบนสุดของเอกสาร XML XML เนมสเปซ ใช้เพื่อป้องกันการสับสนของ SOAP identifier

SOAP request จะเป็นดังนี้

```
POST /StockQuote HTTP/1.1
Host: www.stockquoteserver.com
Content-Type: text/xml; charset="utf-8"
Content-Length: nnnn
SOAPAction: "Some-URI"

<SOAP-ENV:Envelope
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
<SOAP-ENV:Body>
<m:GetLastTradePrice xmlns:m="Some-URI">
<symbol>DIS</symbol>
</m:GetLastTradePrice>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SOAP response จะเป็นดังนี้

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset="utf-8"
Content-Length: nnnn
<SOAP-ENV:Envelope
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"/>
<SOAP-ENV:Body>
<m:GetLastTradePriceResponse xmlns:m="Some-URI">
<Price>34.5</Price>
</m:GetLastTradePriceResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

นอกจากนั้นใน SOAP เมสเซจ จะมีการใช้ XML เนมสเปซ ทุกๆ อีลีเมนต์ในเอกสารจะขึ้นต้นด้วย namespace จะถูกกำหนดโดยใช้ xmlns attribute ดังนี้

```
<SOAP-ENV:Envelope
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
```

แต่หากไม่ต้องการให้ขึ้นต้นด้วย namespace ก็สามารถทำได้ ดังนี้

```
<Envelope
xmlns="http://schemas.xmlsoap.org/soap/envelope/">
```

Envelope

envelope element เป็นอีลีน3648 .มนต์บนสุดของเอกสาร XML ที่ใช้แสดงเมสเซจ อาจประกอบด้วยแอตทริบิวต์ คือ envelope เนมสเปซ และ encodingStyle

1. Envelope เนมสเปซ – SOAP เมสเซจจะแสดงเวอร์ชันโดยใช้ เนมสเปซ ของ envelope element เนมสเปซ จะถูกนำไปใช้ขึ้นต้น envelope, header และ body element

2. EncodingStyle Attribute – ใช้แสดงวิธี serialization ของ SOAP เมสเซจ แอตทริบิวต์นี้สามารถมีได้ในทุกอีลีเมนต์และจะมีผลกับ content ของอีลีเมนต์นั้นและ child element ทั้งหมดของมันที่ไม่ได้ประกาศแอตทริบิวต์

Body

เป็นส่วนของข้อมูลที่ใช้ในการแลกเปลี่ยน โดยทั่ว ๆ ไปข้อมูลจะเป็นการ marshall RPC call และ error report child element ของ body element จะถูกเรียกว่า body entry จะต้องเป็นไปตามกฎนี้

- body entry จะต้องแสดงด้วยชื่อเต็มที่ประกอบด้วย namespce URI และ ชื่อของมัน (local name)
- SOAP encodingStyle attribute อาจจะมีได้ เพื่อแสดงถึงวิธี encode ของ body entry นั้น

Header

เป็นส่วนเพิ่มเติมของเมสเซจสามารถประกอบด้วยอินฟอร์เมชันที่ระบุไปยังแอพพลิเคชัน ส่วนเพิ่มเติมนี้อาจนำไปใช้อิมพลีเม็นต์เป็น authentication, transaction management เป็นต้น ทุก ๆ child element ของ header element จะถูกเรียกว่า header entry

header entry จะต้องเป็นไปตามกฎดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- header entry จะต้องแสดงด้วยชื่อเต็มที่ประกอบด้วย เนมสเปซ URI และ ชื่อของมัน (local name)
- อาจมี SOAP encodingStyle attribute ที่ใช้สำหรับ header entry
- SOAP mustUnderStand attribute และ SOAP actor attribute จะมีหรือไม่ก็ได้ เพื่อใช้บอกว่าจะจัดการกับ entry นั้นอย่างไรและโดยใคร

```
<soap:Header>
<m:local xmlns:m="http://www.w3schools.com/local/"
soap:actor="http://www.w3schools.com/appml" />
<m:language>en</m:language>
<m:currency>USD</m:currency>
</m:local>
</soap:Header>
```

1. SOAP Actor Attribute – SOAP เมสเซจสามารถถูกส่งไปยังปลายทางผ่านตัวกลางของ SOAP (SOAP intermediary) ตัวกลางของ SOAP คือแอพพลิเคชันที่มีความสามารถทั้งรับและส่งต่อ SOAP เมสเซจ ในการส่งต่อตัวกลางจะต้องไม่ส่งต่อ header element ไปให้กับแอพพลิเคชัน โดยตัวกลางนี้จะถูกกำหนดเป็น URI ถ้าไม่มี attribute นี้หมายถึงผู้รับ SOAP เมสเซจเป็นปลายทางสุดท้าย

2. SOAP MustUnderStand Attribute – ใช้แสดงว่า header entry นี้จำเป็นหรือเป็นเพียงอ็อปชันสำหรับผู้รับที่จะนำไปประมวลผล ค่าของมันคือ “1” หรือ “0” ถ้าไม่มีการระบุจะมีค่าเท่ากับ “0” โดย “0” หมายถึงเป็นอ็อปชัน และ “1” หมายถึงจำเป็น โดยจะต้องประมวลผลให้ถูกต้องตาม semantic หรือต้อง fail เช่น เมสเซจเป็นส่วนหนึ่งของ transaction ถ้าปลายทางไม่รองรับ transaction จะต้องไม่ประมวลผลและส่งผลผิดพลาดกลับไปในกรณีที่ mustUnderStand attribute เป็น “1” แต่ถ้าเป็น “0” จะยังคงประมวลผลต่อไป

2.3.2 SOAP Fault Element

ข้อความแสดงความผิดพลาดจากแอพพลิเคชันของ SOAP จะเก็บอยู่ใน fault element ซึ่งถ้ามีจะต้องปรากฏใน body element เพียงครั้งเดียวใน SOAP เมสเซจ ตัวอย่างอาจเป็นดังนี้

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
soap:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
<soap:Body>
<soap:Fault>
<faultcode>soap:MustUnderstand</faultcode>
<faultstring>Mandatory Header error.</faultstring>
<faultactor>http://www.wrox.com/heroes/endpoint.asp</faultactor>
<detail>
<w:source xmlns:w="http://www.wrox.com/">
<module>endpoint.asp</module>
<line>203</line>
</w:source>
</detail>
</soap:Fault>
</soap:Body>
</soap:Envelope>
```

SOAP fault element มี element ย่อย ๆ ดังตาราง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Sub Element	Description
<faultcode>	โค้ดที่ระบุถึงการ error
<faultstring>	ข้อความการ error
<faultactor>	ใครเป็นสาเหตุของการ error
<detail>	ระบุอินฟอร์เมชันของการ error

ตารางที่ 2-1 แสดง element ย่อยของ SOAP fault element

ค่าของ faultcode สามารถมีค่าได้ดังตารางด้านล่าง

Error	Description
VersionMismatch	เนมสเปซ ภายใน SOAP Envelope element ไม่ถูกต้อง
MustUnderstand	child element ของ Header element กับ mustUnderstand attribute ที่มีค่า "1" ผู้รับไม่รองรับ
Client	เมสเซจมีรูปแบบไม่ถูกต้อง หรือมีอินฟอร์เมชันที่ไม่ถูกต้อง
Server	เกิดปัญหากับเซิร์ฟเวอร์ ไม่สามารถประมวลผลได้

ตารางที่ 2-2 แสดงค่า faultcode

2.3.3 SOAP Encoding

SOAP encoding มีวิธีการ map จาก type ของโปรแกรมมิ่งไปเป็น XML 2 วิธี คือจากภายนอก โดยใช้ WSDL (Web Services Description Language) ที่บอกถึง type ของข้อมูลที่รับหรือส่ง หรือใช้ xsi:type attribute ในกรณีที่ภาษาที่ใช้ไม่รองรับ WSDL โดยทั้ง 2 วิธีจะใช้ XML schema ในการระบุ type ตัวอย่างของการใช้ xsi จะเป็นดังนี้

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
soap:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:xsi="http://www.w3.org/1999/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/1999/XMLSchema">
<soap:Body>
<m:MixedMessage xmlns:m="http://www.wrox.com/mix/">
<param1 xsi:type="xsd:string">OU812</param1>
<param2 xsi:type="xsd:integer">2001</param2>
<param3 xsi:type="xsd:double">3.14159</param3>
</m:MixedMessage>
</soap:Body>
</soap:Envelope>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.4 SOAP สำหรับ RPC

จุดประสงค์ของการออกแบบ SOAP คือทำ RPC โดยใช้ XML การเรียก RPC นั้นจะ map เข้ากับ HTTP request และ RPC response จะ map กับ HTTP response

ในการทำ mechod call จำเป็นที่จะต้องมีอินฟอร์เมชันดังนี้

1. URI ของออบเจ็กต์ปลายทาง (target object)
2. ชื่อของเมธอด
3. คำอธิบายของเมธอด (method signature) ส่วนนี้เป็นอ็อปชัน
4. พารามิเตอร์สำหรับเมธอด
5. ข้อมูลส่วนหัว (header data) เป็นอ็อปชัน

พิจารณาเมธอดดังนี้

```
double GetStockQuote ( [in] stirng sSymbol);
```

ถ้า เนมสเปซ ของเมธอดคือ "http://www.worxstox.com/" แล้วการเรียกเมธอดโดย request stock quote ใช้ symbol OU812 จะเป็นดังนี้

```
<q:GetStockQuote xmlns:q="http://www.wroxstox.com/">
<q:sSymbol xsi:type="xsd:string">OU812</q:sSymbol>
</q:GetStockQuote>
```

ชื่อเมธอดและชื่อของอีลีเมนต์จะต้อง match กันเช่นเดียวกับพารามิเตอร์ สำหรับ response จะต้องชื่ออีลีเมนต์จะต้องเป็นชื่อเมธอดตามด้วย response ดังนี้

```
<q:GetStockQuoteResponse xmlns:q="http://www.wroxstox.com/"
<q:ret xsi:type="xsd:double">100.0</q:ret>
</q:GetStockQuoteResponse>
```

ชื่ออีลีเมนต์ของค่าที่คืนมาสามารถเปลี่ยนได้ โดยจะกำหนดได้จากโปรแกรมที่เขียนหรือจาก WSDL

2.4 XML

สำหรับประโยชน์ของ XML นั้น เป็นด้านความยืดหยุ่นในการใช้งานสำหรับแอพพลิเคชันที่อิงกับ Web Base ที่ใช้ง่ายในการค้นหาข้อมูล มีความยืดหยุ่นในการพัฒนาเว็บ สามารถผสมผสานข้อมูลจากหลายแหล่ง จากแอพพลิเคชันที่ต่างกัน สามารถแสดงข้อมูลแบบต่างๆ และสามารถ update ข้อมูลให้ทันสมัยเสมอ และคาดว่าจะเป็นมาตรฐานใหม่ของระบบเปิด ซึ่งนับเป็น format ใหม่สำหรับการส่งข้อมูลบนเว็บที่มากด้วยข้อมูลหลายแบบ แต่ส่งผ่านด้วยเทคโนโลยีที่บีบอัดข้อมูลที่ให้ความเร็วได้รับการสนับสนุนจากผลิตภัณฑ์ค่ายไมโครซอฟท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

XML เป็นการทำงานในระดับกลาง middle tier ที่สามารถเรียกใช้ฐานข้อมูลได้หลากหลายระบบ ฐานข้อมูลและโอนข้อมูลให้อยู่ในรูปแบบของ XML และมีการให้รายละเอียดเกี่ยวกับตัวข้อมูลโครงสร้างต่างๆ ของระบบฐานข้อมูลได้ XML เป็นระบบเปิดที่นำเสนอข้อมูลในรูปแบบ text ผ่านทาง HTTP เหมือนกับ HTML แต่จะมีคุณสมบัติในการให้ข้อมูลแบบ real time อัพเดทหรือเปลี่ยนแปลงได้ตามข้อกำหนด การแสดงข้อมูลจาก XML ใน HTML จะเป็นการเพิ่มในส่วนของรายละเอียดข้อมูล ที่มีการเรียกใช้จากแหล่งหรือฐานข้อมูลที่เชื่อมโยงกันในหลายแหล่งเพื่อให้HTMLมีความสมบูรณ์มากขึ้น

ในอนาคตการพัฒนาเว็บหรือการเขียน และสร้าง HTML ไม่จำเป็นต้องมีการเขียนชุดคำสั่งที่ยุ่งยากซับซ้อนมากก็สามารถทำงานร่วมกับระบบข้อมูลได้อย่างมีประสิทธิภาพ XML จะทำการกำหนดค่าสำหรับโครงสร้าง ข้อมูลที่จะนำไปแสดงใน HTML นอกจากนั้นยังสามารถนำไปสนับสนุนระบบการแลกเปลี่ยนข้อมูลข่าวสารทาง Electronic ได้อย่างดีอีกด้วย

2.4.1 แนะนำ XML

XML ย่อมาจาก eXtensible Markup Language XML ถูกออกแบบมาเพื่อใช้อธิบายข้อมูลใน XML จะไม่มีแท็ก (tag) ที่กำหนดไว้ล่วงหน้า เราจะต้องสร้างแท็กของเราขึ้นมาเอง และจะใช้Document Type Definition (DTD) ในการอธิบายรูปแบบของข้อมูล

ข้อแตกต่างที่สำคัญระหว่าง XML และ HTML คือ XML ถูกออกแบบมาเพื่อใช้แลกเปลี่ยนข้อมูล ไม่ใช่มาแทนที่ HTML XML กับ HTML ถูกออกแบบมาเพื่อจุดประสงค์ที่ต่างกัน HTML จะเกี่ยวข้องกับการแสดงข้อมูล XML จะเกี่ยวข้องกับการอธิบายข้อมูล แท็กของ HTML เช่น <H1></H1>เป็นแท็กสำหรับบอกรูปแบบการแสดงผล ในขณะที่แท็กของ XML ต้องสร้างขึ้นมาเองเช่น

```
<BOOK>
      <NAME>Beginning XML</NAME>
      <ISDN>1-861003-41-2</ISDN>
      <PAGE>800</PAGE>
<BOOK>
```

จะใช้แสดงข้อมูลของหนังสือที่ประกอบด้วย element BOOK , NAME , ISDN และ PAGE ข้อมูลที่อยู่ในรูปแท็กซ์นี้เป็นอิสระต่อซอฟต์แวร์และฮาร์ดแวร์ จึงเหมาะที่จะนำมาใช้ในการแลกเปลี่ยนข้อมูลระหว่างระบบบนอินเทอร์เน็ต การเปลี่ยนข้อมูลให้อยู่ในรูป XML เป็นการลดความซับซ้อนและสามารถสร้างข้อมูลที่ถูกอ่านโดยแอพพลิเคชันใด ๆ ก็ได้ นอกจากนั้นการเพิ่มเติม element จะไม่ทำให้แอพพลิเคชันเสียหาย เนื่องจากการอ่านข้อมูลโดย XML parser จะใช้วิธีแยกข้อมูลจาก element ที่ต้องการจาก XML ข้างต้นนี้จะแยกข้อมูล element NAME , ISDN และ PAGE หากเพิ่ม element AUTHOR เข้าไป แอพพลิเคชันก็ยังคงดึงข้อมูลใน element เดิมโดยจะมองข้าม element ที่ไม่ได้กำหนดให้แยกข้อมูลออกมาในตอนพัฒนาแอพพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2 โครงสร้างของ XML

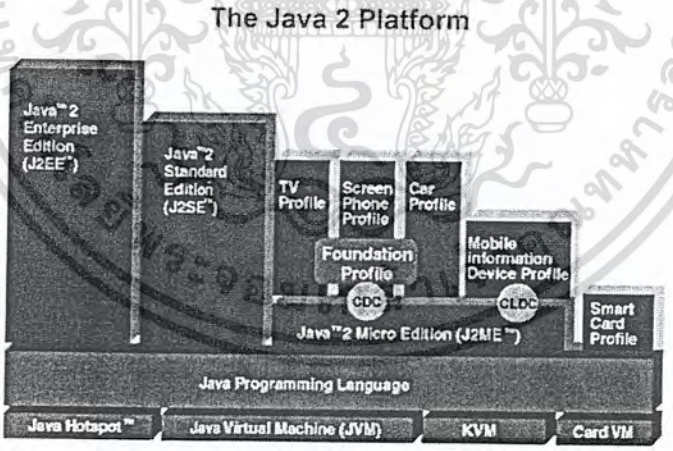
เอกสาร XML จะถูกสร้างจากส่วนต่างๆ ดังนี้

- Element เป็นโครงสร้างหลักของ XML
- Tag ใช้ในการกำหนด markup element
- Attribute ให้ข้อมูลเพิ่มเติมกับ element
- Entity เป็นตัวแปรที่ใช้ในการกำหนดเท็กซ์ทั่ว ๆ ไป entity reference เป็นการอ้างถึง entity เช่น entity ชื่อ domain มีค่าเท่ากับ www.w3.org entity reference คือ &domain หากภายในเอกสาร XML มี &domain ค่าของมันจะมีค่าเท่ากับ www.w3.org
- PCDATA -หมายถึง parse character data PCDATA เป็นเท็กซ์ที่จะถูก parse โดย parser
- CDATA -หมายถึง character data เป็นเท็กซ์ที่จะไม่ถูก parse โดย

2.5 J2ME

จาวาทูเอ็มอีนี้เป็นจาวารุ่นหนึ่งของแพล็ตฟอร์มจาวาทู เป็นเป้าหมายสำหรับใช้งานบนอุปกรณ์ขนาดเล็ก โดยตัดฟังก์ชั่นที่ไม่จำเป็นออกไปเพื่อให้สามารถทำงานได้ในสภาวะที่มีทรัพยากรจำกัด

จาวาทูเอ็มอีได้นำโครงสร้างแบบโมดูลที่มีความยืดหยุ่นสูงเข้ามาใช้งาน เพื่อให้สามารถรองรับการใช้งานที่หลากหลาย โดยชั้นของซอฟต์แวร์นั้นมี 3 เลเยอร์ด้วยกัน โดยเลเยอร์ทั้งหมดจะอยู่เหนือระบบปฏิบัติการดังรูปที่ 2-6 ซึ่งมีดังนี้



รูปที่ 2-6 แสดงแพลตฟอร์มของจาวาทู

เลเยอร์ที่ 1 : จาวาเวอร์ชวลแมชชีนเลเยอร์ (Java Virtual Machine Layer) เป็นเลเยอร์ สำหรับการสร้าง จาวาเวอร์ชวลแมชชีนเลเยอร์ ซึ่งต้องปรับแต่งให้เข้ากับระบบปฏิบัติการของอุปกรณ์แต่ละชนิด และรองรับ การปรับแต้งแต่ละรูปแบบของจาวาทูเอ็มอี โดยจากรูปที่ 2-6 จะเห็นว่า เวอร์ชวลแมชชีน สำหรับจาวาทูเอ็มอี ก็คือ CVM และ KVM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เลเยอร์ที่ 2 : เลเยอร์การปรับแต่ง(Configuration Layer) การปรับแต่งจาวาทูเอ็มอีจะระบุตัว คลาสไลบรารี(class libraries) ตามกลุ่มของชนิดอุปกรณ์ที่ใช้งานทั่วไปภายใต้พื้นฐานความต้องการของหน่วยความจำที่จำกัด และ หน่วยประมวลผลที่จำกัด จากรูปที่ 2-6 การปรับแต่ง(Configuration) ของ จาวาทู มี 2 แบบคือ CDC (Connected Device Configuration) และ CLDC (Connected Limited Device Configuration)

เลเยอร์ที่ 3 : โพร์ไฟล์เลเยอร์ (Profile Layer) เป็นเลเยอร์ ที่ถูกสร้างไว้เหนือ เลเยอร์การปรับแต่ง โดยได้นิยาม คลาสไลบรารีต่าง ๆ เอาไว้เพื่อสำหรับ ความต้องการของตลาดเฉพาะกลุ่ม ตัวอย่างของ โพร์ไฟล์ในเลเยอร์ นี้ก็เช่น PDAP, MIDP, โพร์ไฟล์พื้นฐาน(Foundation Profile)และ โพร์ไฟล์บุคคล(Personal Profile) เป็นต้น

2.5.1 ส่วนการปรับแต่งใน จาวาทูเอ็มอี (Configuration)

การปรับแต่ง(Configuration) และ โพร์ไฟล์ ถือเป็นส่วนหลักสำคัญของ จาวาทูเอ็มอี ซึ่งจุดประสงค์ของการกำหนดสองส่วนนี้ก็เพื่อที่จะให้ได้ เวอร์ชวลแมชชีนและ คลาสไลบรารีที่เหมาะสมสำหรับอุปกรณ์แต่ละกลุ่มประเภท

การปรับแต่ง(Configuration) หมายถึงชุดคุณสมบัติขั้นต่ำของ จาวาเวอร์ชวลแมชชีน และ จาวาคลาสไลบรารี สำหรับอุปกรณ์แต่ละประเภท หรืออาจกล่าวได้ว่า การปรับแต่ง(Configuration)นั้นเป็นตัวกำหนดคุณสมบัติขั้นต่ำของแพลตฟอร์มจาวาที่นักพัฒนาคาดว่าจะต้องมีในทุกอุปกรณ์ ขณะที่คลาสไลบรารีที่กำหนดใน การปรับแต่ง(Configuration) จะมีในทุกอุปกรณ์ที่อยู่ในกลุ่มเดียวกัน

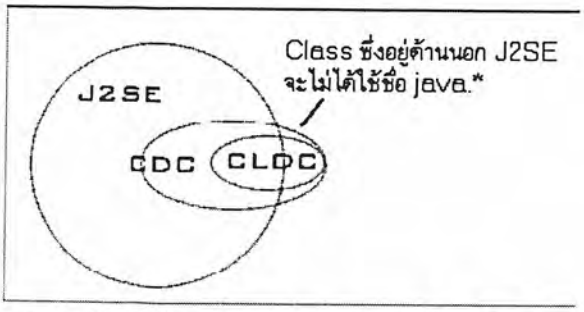
ปัจจุบันได้แบ่งการปรับแต่ง(Configuration) สำหรับจาวาทูเอ็มอี ออกเป็น 2 ประเภทคือ CDC (Connected Device Configuration) และ CLDC (Connected Limited Device Configuration) โดยมีเป้าหมายอยู่ที่อุปกรณ์ 2 ประเภทซึ่งต้องการหน่วยความจำและหน่วยประมวลผลพื้นฐานใกล้เคียงกัน

CDC เป็นอุปกรณ์ใช้ร่วมกับอุปกรณ์อื่น ติดตั้งตายตัว และใช้เชื่อมต่อข้อมูล โดยปรกติแล้ว มักมีส่วนติดต่อกับผู้ใช้หลากหลายแบบมีหน่วยความจำประมาณ 2-16 เม็กกะไบต์ ใช้หน่วย ประมวลผลแบบ 32 บิตหรือมากกว่าเชื่อมต่อเครือข่ายที่มีช่องสัญญาณสูงโดยอาศัยพอร์ต TCP/IP

CLDC เป็นอุปกรณ์ส่วนบุคคล พกพาได้ และใช้เชื่อมต่อข้อมูล โดยปรกติจะมีส่วนติดต่อกับผู้ใช้แบบง่าย ๆ มีหน่วยความจำประมาณ 128 กิโลไบต์ ถึง 1 เม็กกะไบต์ ใช้หน่วยประมวลผลแบบ 16 บิต หรือ 32 บิต เชื่อมต่อกับเครือข่ายที่มีแบนด์วิดท์ต่ำเป็นระยะเวลาสั้น ๆ โดยไม่อาศัยพอร์ต TCP/IP

ในชั้นของ การปรับแต่ง(Configuration) นั้นมีคลาส 2 ประเภทด้วยกัน คือ คลาสที่นำมาจาก J2SE และที่ออกแบบเฉพาะอุปกรณ์ขนาดเล็ก คลาสที่นำมาจาก J2SE จะมีคุณสมบัติเดียวกับคลาสใน J2SE หรือเป็นซับคลาสใน J2SE เช่น แพ็คเก็จ java.lang ,java.io และ java.util

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-7 แสดงความสัมพันธ์ระหว่าง J2SE และคลาสไลบารีใน CDC และ CLDC

2.5.2 เวอร์ชวลแมชชีนของจาวาทูเอ็มอี

เวอร์ชวลแมชชีน ของ CDC คือ ซีเวอร์ชวลแมชชีน(C Virtual Machine หรือ CVM) มีคุณสมบัติครบถ้วนเหมือน จาวาทูเวอร์ชวลแมชชีน แต่ขนาดเล็กกว่า ออกแบบมาสำหรับอุปกรณ์ที่สลับซับซ้อน CVM มีความ ต้องการหน่วยความจำ 256 กิโลไบต์ ขณะที่หน่วยความจำแบบ ROM ของ CDC มีขนาด 1 เม็กกะไบต์

เวอร์ชวลแมชชีน ของ CLDC คือ เคเวอร์ชวลแมชชีน(K Virtual Machine หรือ KVM) ออกแบบมาใช้งานกับอุปกรณ์ที่มีทรัพยากรจำกัด ซึ่งมีหน่วยความจำประมาณ 128 กิโลไบต์ หน่วยประมวลผลแบบ 16 หรือ 32 บิต

2.5.3 โพรไฟล์ของจาวาทูเอ็มอี

โพรไฟล์ของจาวาทูเอ็มอีจะกำหนดชุดของ API (Application Programming Interface) ที่ต้องใช้เพิ่มเติม เช่น ส่วนติดต่อกราฟฟิกกับผู้ใช้ การเชื่อมต่อเครือข่าย หน่วยเก็บข้อมูล เป็นต้น ซึ่งโปรไฟล์เหล่านี้ได้แก่

- โพร์ไฟล์พื้นฐาน (Foundation Profile) ซึ่งสร้างบน CDC โดยใช้งานกับอุปกรณ์ที่ต้องติดตั้งจาวาเวอร์ชวลแมชชีน ตัวเต็มควบคู่กับแพลตฟอร์มจาวาทู และ API ชุดมาตรฐาน
- RMI(Remote Method Invocation) โพร์ไฟล์ ซึ่งสร้างบน CDC โดยระบุซับเซ็ตขั้นต่ำของ RMI ใน J2SE รุ่น 1.3
- โพร์ไฟล์บุคคล (Personal Profile) โพร์ไฟล์ ซึ่งสร้างบน CDC โดยมีเพื่อเป็นตัวเสริม เพอร์ซันนอลจาวา เอนไวรอนเมนต์(PersonalJava Environment) ของซันไมโครซิสเตมส์
- PDAP (PDA Profile) ซึ่งสร้างบน CLDC มีส่วนติดต่อกับผู้ใช้ และ API ด้านการเก็บข้อมูลสำหรับอุปกรณ์พกพาขนาดเล็ก
- MIDP ซึ่งสร้างบน CLDC โดยเตรียมส่วนติดต่อกับผู้ใช้และหน่วยเก็บข้อมูล มีความสามารถด้านเครือข่าย แบบจำลอง API สำหรับโปรแกรมประยุกต์ไว้ให้อุปกรณ์ไร้สาย เช่น โทรศัพท์ที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.4 มิดเล็ต (MIDlet)

มิดเล็ต(MIDlet) หมายถึง โปรแกรมประยุกติบน MIDP อาจกล่าวได้ว่า มิดเล็ตนั้นมีส่วนคล้ายกับจาวาแอปเพล็ต แม้จะไม่มีเมธอด main() แต่มิดเล็ตก็นำคลาส javax.microedition.midlet.MIDlet เข้ามาใช้ นอกจานี้มิดเล็ตยังได้กำหนด คอนสตักเตอร์(constructor) แบบ สาธารณะ(public) ที่ไม่มีอาร์กิวเมนต์ใด ๆ อีกด้วย

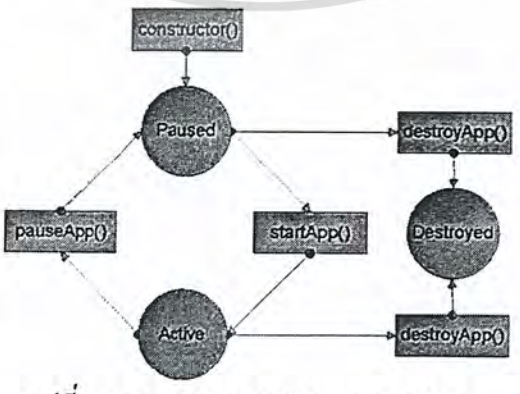
วงจรการทำงานของ มิดเล็ต

การทำงานของ มิดเล็ต ประกอบด้วย 3 สถานะคือ กำลังทำงาน หยุดทำงานชั่วคราวและถูกทำลาย การเปลี่ยนสถานะหนึ่งไปยังอีกสถานะหนึ่งควบคุมโดยเมธอด startApp() ,pauseApp() และ destroy()

เมื่อมิดเล็ต พร้อมสั่งกระทำการ ซอฟต์แวร์จัดการแอพพลิเคชั่นจะสร้างตัวอย่างมิดเล็ตขึ้นมาก่อน โดยใช้ คอนสตรักเตอร์แบบสาธารณะ ที่ไม่มีอาร์กิวเมนต์ใด ๆ โดยมิดเล็ตจะอยู่ในสถานะหยุดชั่วคราว

จากนั้นซอฟแวร์จัดการแอพพลิเคชั่นจะเรียกเมธอด startApp() ขึ้นมา และ มิดเล็ตจะเข้าสู่สถานะกำลังทำงานเปิดรับทรัพยากรที่ต้องการและเริ่มต้นการทำงาน ในสถานะนี้มิดเล็ตจะทำงานและดึงทรัพยากรที่ต้องการไว้ใช้งาน เมื่อซอฟแวร์จัดการแอพพลิเคน3594 ั่นไม่ต้องการให้มิดเล็ตทำงานต่อไปก็จะเรียกเมธอด pauseAPP() จากนั้นมิดเล็ตจะหยุดทำงานและกลับเข้าสู่สถานะหยุดชั่วคราว คืนทรัพยากรที่ดึงมาใช้งานและเข้าสู่สภาวะไม่ทำงาน มิดเล็ตจะสามารถเข้าไปสู่สถานะกำลังทำงานได้เมื่อซอฟต์แวร์จัดการแอพพลิเคชั่นเรียกเมธอด startAPP() ขึ้นมา และเมื่อซอฟต์แวร์จัดการแอพพลิเคชั่นไม่ต้องการเรียกใช้งาน MIDlet หรือต้องการเคลียร์หน่วยความจำเพื่อให้โปรแกรมอื่นใช้งานได้ ก็จะส่งสัญญาณทำลายมิดเล็ต โดยการเรียกเมธอด destroyApp() แต่หาก มิดเล็ตอยู่ระหว่างการกระทำการขั้นตอนสำคัญอยู่ก็อาจร้องขอไม่ให้เข้าสู่สถานะถูกทำลายได้โดยเรียกใช้ MIDletStateChangeException อย่างไรก็ตาม ซอฟต์แวร์จัดการแอพพลิเคชั่นอาจปฏิเสธหรือยินยอมการร้องขอนี้ก็ได้ซึ่ง ตัวแปร uncondition ซึ่งเป็นตัวแปรบูลีนที่อยู่ใน destroyApp() นั้นหากเป็นจริง การร้องขอก็จะไม่ได้รับการสนอง แต่หากเป็น เท็จการร้องขอก็จะได้รับการตอบสนอง

หากมิดเล็ตต้องการหรือเข้าสู่สถานะหยุดชั่วคราวหรือถูกทำลายด้วยตัวเอง ก็สามารถทำได้โดยการเรียกใช้ เมธอด notifyPause(),notifyDestroy() ตามลำดับ



รูปที่ 2-8 แสดงวงจรการทำงานของ MIDlet

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

2.5.5 การใช้งาน API ระดับสูงในการพัฒนาส่วนติดต่อกับผู้ใช้

ลิสต์และช้อย

ลิสต์และช้อย กำหนดไว้ในแพ็กเกจ javax.microedition.lcdui โดยคลาสลิสต์นั้นมี คอนสตรักเตอร์ที่ถูกกำหนดไว้ 2 แบบดังนี้ คือ List(String title,int listType)และ List(Stirng title,intlistType,String[] stringElements,Image[] imageElements) โดย คอนสตรักเตอร์ ตัวแรกจะทำหน้าที่ในการสร้าง ลิสต์เปล่าๆ ขึ้นมาซึ่งสามารถเพิ่มรายการของ ช้อยเข้าไปได้ภายหลัง ส่วน คอนสตรักเตอร์ ตัวที่สองนั้นจะสร้างลิสต์ที่มีค่าเริ่มต้นอยู่ใน อาร์เรย์ stringElements ซึ่งต้องไม่เป็น null และสามาชิกทุกตัวในอาร์เรย์ต้องไม่เป็น null เช่นเดียวกันความยาวของอาร์เรย์ stringElements จะกำหนดรายการในลิสต์ อาร์เรย์ imageElements นั้นอาจมีค่าเป็น null ซึ่งหมายถึงรายการในลิสต์นั้นไม่มีภาพอยู่ ถ้าอาร์เรย์ imageElements นั้นไม่มีค่าเป็น null ก็จะต้องมีความยาวเท่ากับอาร์เรย์ stringElements รายการแต่ละรายการในอาเรย์ imageElements อาจจะเป็น null ได้หมายความว่าไม่มีภาพในรายการของลิสต์ที่ตรงกัน เมธอดทั้งหมดที่ใช้เรียกดูและเปลี่ยนแปลงรายการในลิสต์จะกำหนดไว้ในอินเตอร์เฟสช้อย

อินเตอร์เฟส ช้อย จะกำหนด API สำหรับคอมโพเน็นต์ส่วนติดต่อกับผู้ใช้โดยนำตัวเลือกจากรายการที่กำหนดไว้ล่วงหน้าจำนวนหนึ่งมาใช้ โดยอินเตอร์เฟสช้อย มีเมธอด เช่น

```
Int append(String stringPart,Image imagePart)
void delete(int elementNum)
Image getImage(int elementNum)
Int getSelectedIndex()
String getString(int elementNu,)
void insert(int elementNum,String stringPart,Image imangePart)
boolean isSelected(int elementNum)
void set(int elementNum,String stringPart,Image imagePart)
void setSelectedIndex(int elementNum,Boolean selected)
int size()
```

ออบเจ็กต์ช้อยมี 3 ประเภทได้แก่ อิมพลิซิท-ช้อย(implicit-choice) เอ็กคลูซิบ-ช้อย (exclusive-choice) และมัลติเปิ้ล-ช้อย(multiple-choice)

- เอ็กคลูซิบ-ช้อย จะต้องเลือกรายการใดรายการหนึ่งในแต่ละครั้ง เว้นเสียแต่ว่าไม่มีรายการใดๆ ให้เลือก
- อิมพลิซิท-ช้อย เป็นกรณีพิเศษของ เอ็กคลูซิบ-ช้อย เมื่อรายการที่อยู่ในโฟกัสถูกเลือกโดยอ้อมเมื่อเริ่มต้นซึ่งช้อยประเภทนี้ใช้กับลิสต์เท่านั้น
- มัลติเปิ้ล-ช้อย สามารถเลือกได้หลายรายการพร้อมกันในแต่ละครั้ง หรือจะไม่เลือกเลยก็ได้

การแก้ไขออบเจ็กต์ช้อย

เราสามารถใช้เมธอด append(StringstringPart,Image imgePart) และ insert(int elementNum,String stringPart) ในการเพิ่มออบเจ็กต์ช้อย ซึ่งหลังจากที่ได้ทำการเพิ่มแล้วเรายังสามารถแก้ไขภาพหรือสตริงได้โดยใช้เมธอดset() การลบออกเราก็ทำได้โดยใช้เมธอด delete(int elementNum)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
61760

เท็กซ์บ็อก (Text box)

คลาสเท็กซ์บ็อก เป็นรูปแบบที่มีโครงสร้างกำหนดไว้อยู่แล้ว ซึ่งผู้ใช้สามารถป้อนหรือแก้ไขข้อความในเท็กซ์บ็อกได้ การาสร้างออบเจ็กเท็กซ์บล็อกนั้นใช้คอนสตรักเตอร์ดังนี้

TextBox(String title,String text,int maxSize,int constraints) โดยฟังก์ชั่นที่ใช้ได้สำหรับเท็กซ์บ็อกมีดังนี้

```
void delete(int offset,int lenght)
int getCaretPosition()
int getChars(char[] data)
int getConstraints()
int getMaxSize()
String getString()
void insert(char[] data,int offset,int length,int length,int
position)
void insert(String src,int position)
void setChars(char[] data,int offset,int lenght)
void setConstriants(int constraints)
int setMaxSize(int maxSize)
void setString(String text)
int size()
```

ขนาดสูงสุดของข้อความในเท็กซ์บ็อก เนื่องจากว่า เท็กซ์บ็อกจะเก็บข้อความเป็นอาร์เรย์ของตัวอักษรดังนั้นจึงมีขนาดจำกัด โดยเราสามารถกำหนดขนาดสูงสุดได้โดยใช้เมธอดดังนี้ TextBox(String title,String text,int MaxSize,int constrains) หรืออาจใช้เมธอด getMaxSize() ก็ได้ ซึ่งขีดจำกัดว่าเราสามารถสร้างเท็กซ์บ็อกให้มีขนาดเท่าไหร่นั้นอยู่ที่ MIDP เพราะว่าหน่วยความจำมีขนาดจำกัด

อเลิร์ท (Alert)

คลาส อเลิร์ท มีรูปแบบหรือโครงสร้างใช้สำหรับแสดงข้อมูลให้ผู้ใช้และจะคอยครู่หนึ่ง (กำหนดโดย timeout) ก่อนที่จะแสดงหน้าจอถัดไป คอนสตรักเตอร์มี 2 เมธอดดังนี้

```
Alert(String title)
Alert(String title,String alertText,Image alertImage,AlertType
alertType)
```

ฟอร์ม (form) และ ไอเท็ม (item)

ฟอร์มเป็นออบเจ็กต์ซึ่งสามารถบรรจุไอเท็มต่าง ๆ ได้เช่นภาพ ข้อความ หรือ ช้อย จำนวนเท่าไหร่ก็ได้ การสร้างออบเจ็กต์ฟอร์มทำได้ 2 แบบคือ สร้างฟอร์มเปล่าที่ไม่มีไอเท็มใด ๆ โดยใช้เมธอด Form(String title) หรือสร้าง ฟอร์มเริ่มต้นที่มีไอเท็มโดยใช้ เมธอด Form(String title,Item[]) เมธอดที่สามารถใช้ในการจัดการไอเท็มในฟอร์มคือ

```
int append(Image img)
int append(Item item)
int append(String str)
void delete(int itemNum)
void insert(int itemNum,Item item)
void set(int itemNum,Item item)
```

ไอเท็ม (Item)

คลาส ไอเท็มเป็นซุปเปอร์คลาสสำหรับคอมโพเน้นต์ต่าง ๆ ที่สามารถใส่ลงไปในฟอร์มได้แต่ละไอเท็มจะมีป้ายข้อความกำกับ ซึ่งเป็นฟิลด์สตริงที่ติดอยู่กับไอเท็ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สตริงไอเท็ม(StringItem)

ออบเจ็กต์สตริงไอเท็ม ประกอบด้วยป้ายและสตริงข้อความที่แสดงผลอย่างเดียว ผู้ใช้ไม่สามารถแก้ไขค่าได้โดยตรง ซึ่งต้องทำผ่านเมธอด setText() การสร้างออบเจ็กต์ของ StringItem ทำได้ ดังนี้ StringItem(String label,String text) การดูค่าทำได้โดยใช้ เมธอด getText()

อิมเมจไอเท็ม(ImageItem)

ออบเจ็กต์อิมเมจไอเท็ม อาจมีออบเจ็กต์อิมเมจ(Image) ซึ่งสามารถเปลี่ยนแปลงได้การสร้างออบเจ็กต์อิมเมจไอเท็มทำได้ดังนี้

```
ImageItem(String label,Image img,int layout,String altText)
```

ข้อความทางเลือก (Alternative text)

ข้อความทางเลือกนั้นใช้ในกรณีที่ว่าอุปกรณ์เหล่านั้นไม่สามารถแสดงภาพได้ ซึ่งกำหนดโดยพารามิเตอร์ altText โดยใช้เมธอด setAltText()

เดตฟิลด์ (DateField)

เดตฟิลด์ คือ ไอเท็มที่แก้ไขได้ ใช้สำหรับแสดงวัน เดือน ปี และเวลา สามารถสร้างออบเจ็กต์เดตฟิลด์ได้โดยใช้คอนสตรักเตอร์ดังนี้

```
DataField(String label,int mode)
DataField(String label,int mode,TimeZone timeZone)
```

การป้อนข้อมูลสามารถทำได้ 3 โหมดด้วยกัน

- วัน(DATE) -> ผู้ใช้กำหนดได้เฉพาะข้อมูลวันเดือนปีเท่านั้น
- เวลา(Time) -> ผู้ใช้กำหนดได้เฉพาะข้อมูลเวลาเท่านั้น ส่วนวัน เดือน ปี จะถูกกำหนดเป็น 1 มกราคม ค.ศ. 1970 และไม่สามารถแก้ไขได้
- วันเวลา(DATE_TIME) -> ผู้ใช้กำหนดได้ทั้งวันเดือนปีและเวลา

เท็กซ์ฟิลด์ (TextField)

คลาส เท็กซ์ฟิลด์ เป็นไอเท็ม ที่มีสตริงข้อความอยู่ซึ่งให้ผู้ใช้สามารถแก้ไขข้อความได้ การสร้างออบเจ็คเท็กซ์ฟิลด์ สามารถทำได้โดยใช้เมธอด TextField(String label,String text,int maxSize,int constraints)

เท็กซ์ฟิลด์ จะมีรูปแบบคล้าย ๆ กับเท็กซ์บ็อก ต่างที่เท็กซ์บ็อกสามารถแสดงผลได้ทันทีแต่เท็กซ์ฟิลด์ต้องเอาไปใส่ไว้ในฟอร์มก่อนจึงจะแสดงผลได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน่วยเก็บข้อมูลแบบคงตัว

หน่วยเก็บข้อมูลแบบคงตัว ในที่นี้หมายถึงการเก็บข้อมูลลงในฐานข้อมูล ซึ่งตารางในฐานข้อมูลเราจะเรียกว่า เรคคอร์ดสตอร์(record store) แต่ละข้อมูลที่อยู่ใน เรคคอร์ดสตอร์ เราเรียกว่า เรคคอร์ด

เรคคอร์ดสตอร์ (Record store)

RMS ที่มีใน MIDP เป็นระบบฐานข้อมูลแบบง่าย ๆ เรคคอร์ดสตอร์ ซึ่งเทียบได้เท่ากับตารางงานข้อมูลคือไฟล์ที่ประกอบด้วยชุดเรคคอร์ด

การทำงานของเรคคอร์ดสตอร์

คลาส RecordStore ได้จัดเตรียมวิธีการจัดการ เรคคอร์ดสตอร์ โดยมีคำสั่ง เปิด,ปิด และสร้างดังนี้

```
Public static RecordStore openRecordStore(String
recordStrname,Boolean createIfNecessary)
public void closeRecordStore()
public static void deleteRecordStore(String recordStrname)
```

MIDlet สร้าง เรคคอร์ดสตอร์ ขึ้นมาใหม่โดยใช้เมธอด RecordStore.openRecordStore(true) แต่ถ้า recordStoreName ไม่มีอยู่ก่อน ก็จะสร้างมาใหม่ และหากมีอยู่แล้วก็จะเป็นการเปิดขึ้นมา

การจัดการ เรคคอร์ด

กลุ่มของข้อมูล เรคคอร์ดสตอร์ เราเรียกว่า เรคคอร์ดซึ่ง โครงสร้างของเรคคอร์ดเกิดจากอาร์เรย์ของไบท์ละจะกำหนดค่าสำหรับแต่ละเรคคอร์ดด้วย RecordTD เรคคอร์ดสามารถสร้างใหม่ลบ เรียกดูหรือแก้ไขได้ก็ต่อเมื่อได้เปิด เรคคอร์ดสตอร์ ด้วยเมธอดดังนี้

```
public int addRecord(byte[] data,int offset,int numBytes)
public void deleteRecord(int recordid)
public int getRecord(int recorded,byte[] buffer,int offset)
public byte[] getRecord(int recordId)
public void setRecord(int recorded,byte[] newData,int offset,int
numByte)
```

การทำงานของส่วนการเชื่อมต่อเครือข่าย

ในส่วนนี้เราจะกล่าวถึงการติดต่อเครือข่ายสำหรับอุปกรณ์ไร้สายโดยใช้ HttpConnection ซึ่งจุดประสงค์ในการนำ HttpConnection มาใช้งานในการติดต่อสื่อสารในแอพพลิเคชั่นมีข้อดีหลายประการด้วยกันเช่น

อุปกรณ์ MIDP บางตัวไม่สนับสนุนการติดต่อสื่อสารแบบซ็อกเก็ตและดาต้าแกรม ขณะที่อุปกรณ์ทุกตัวรองรับการติดต่อแบบ HTTP อยู่แล้ว

ในโครงงานนี้มีความจำเป็นที่ต้องรองขอเครื่องให้บริการ(server) เพื่ออ่านเอกสาร XML ดังนั้นการใช้ HTTP จึงสะดวกที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการสร้าง การร้องขอผ่าน HTTP นั้นจะใช้ เมธอด Connector.open ซึ่งเราสามารถเขียนได้ดังนี้

```
Import javax.microedition.io.*;
HttpConnection conn = Connector.open("http://www.java.sun.com/jdc");
```

แต่หากมีการ ส่งต่อ(redirect) เกิดขึ้น วิธีนี้ก็ไม่สามารถไปยัง URL ที่ให้ส่งต่อไปได้ ซึ่งเราต้องเขียน Class ขึ้นมาเองซึ่งจะไม่ขออธิบายไว้นะที่นี้

2.6 ASP.NET

เว็บเซอร์วิสช่วยให้เราเรียกใช้ ซอฟแวร์คอมโพเนนต์ ผ่านทางเว็บโปรโตคอลเช่น HTTP โดยใช้อินเทอร์เน็ตและ XML เราสามารถสร้าง software component เพื่อติดต่อกับคนอื่นๆ โดยไม่ต้องสนใจในเรื่องของภาษา และแพลตฟอร์ม จากเดิมแต่ละค่ายจะมีการทำการติดต่อผ่านโปรโตคอลของค่ายตัวเอง มีอินเทอร์เฟสที่แตกต่างกันไป ในเว็บเซอร์วิสนี้สิ่งเหล่านี้จะเป็นมาตรฐานในการติดต่อส่งผ่านกันให้เป็นเรื่องง่าย โดยผ่าน SOAP และ XML เป็นเครื่องมือสำคัญ

2.6.1 องค์ประกอบในการใช้งาน .NET เว็บเซอร์วิส

รูปแบบการเชื่อมต่อ

Microsoft .NET Web Services ตอนนี้รองรับสามโปรโตคอล ได้แก่ HTTP GET, HTTP POST และ SOAP (Simple Object Access Protocol) เนื่องจากโปรโตคอลเหล่านี้เป็นมาตรฐานของเว็บอยู่แล้ว จึงง่ายที่ไคลเอนต์จะเรียกใช้

HTTP GET

query string จะรวมอยู่ใน URL ของเซิร์ฟเวอร์นั้น

HTTP POST

จะทำการรวม ชื่อ ค่าตัวแปรต่างๆ ไว้ในส่วนของบอดี้ ของ HTTP request message

SOAP

SOAP จะแตกต่างจากทั้งสองวิธีข้างต้น มันจะใช้ XML ในการจัดวางรูปแบบเมสเสจที่ส่งไปจะมีโครงสร้างที่ดีกว่าและสามารถส่งผ่านข้อมูลที่ซับซ้อนได้ ข้อแตกต่างอีกอย่างคือ SOAP สามารถใช้บน transport protocols อื่นๆได้ เช่น SMTP นอกเหนือจาก HTTP

2.6.2 การสร้าง ASP.NET เว็บเซอร์วิส

ขั้นตอนในการสร้าง Web Service

1. สร้างไฟล์ .asmx สำหรับ web service ที่สร้างด้วย ASP.NET ต้องประกอบไปด้วย directive <% webservice . . . %> การเขียนก็คล้ายกับการเขียนคลาส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. WebService Class นั้นต้องทำการ inherite มาจาก System.Web.Services namespace รวมทั้งทำการกำหนด namespace ของเว็บเซอร์วิสของคุณด้วยเนื่องจาก defalut ของมันคือ http://tempuri.org/ ซึ่งจะไม่ unique
3. เมธอดที่ต้องการให้เป็น เมธอดสาธารณะที่คนอื่นๆเข้ามาเรียกใช้ได้ ต้องทำการกำหนดให้เป็น <Web Method> ด้วย
4. นำไฟล์ .asmx ที่สร้างไปไว้ในไดเร็กทอรีที่ต้องการใน IIS

ตัวอย่างเว็บเซอร์วิส

```
<%@ WebService Language="VB" Class="Math"%>
Imports System.Web.Services
Imports System
<WebService(Namespace:="http://www.olala.com/")> _
Public Class Math
<WebMethod()> _
Public Function Add(num1 As Integer, num2 As Integer) _
As Integer
Return num1 + num2
End Function
<WebMethod()> _
Public Function Hello() _
As String
Return "Hello World"
End Function
End Class
```

การกำหนดค่าแอททริบิวต์ต่างๆให้แก่เว็บเซอร์วิส

- BufferResponse (boolean) ควบคุมให้มี buffer ของผลตอบรับของเมธอดหรือไม่
- CacheDuration กำหนดค่าช่วงเวลา(วินาที) ที่ทำการเก็บ method response ใน cache ค่าปกติคือ 0 วินาที
- Description กำหนดคำอธิบายของเมธอดนี้
- EnableSession (boolean) ให้มีการใช้ เซสชั่นสำหรับเมธอดนี้หรือไม่ โดยค่าปกติเป็น true ถ้าเราไม่ได้ ใช้งานควรกำหนดเป็น false เพื่อเพิ่มประสิทธิภาพ
- MessageName เพื่อใช้ในการแย่งแยกเว็บเมธอดในกรณีที่มี ชื่อของเมธอดเหมือนกัน
- TransactionOption เป็นการกำหนด transaction ของเมธอดนั้น มีทั้งหมด 5 แบบ ดังที่ได้กล่าวไปแล้วในเรื่อง Transaction ได้แก่
    1. Disabled object ที่สร้างขึ้นจะไม่มี transaction ใน COM+ transaction แต่สามารถติดต่อโดยตรงกับ DTC ในกรณีที่ต้องการการสนับสนุน
    2. NotSupported ไม่มีการทำ transaction เลย
    3. Supported ถ้า object นั้นจะถูกรันใน context transaction ของตัวที่สร้างมันขึ้นซึ่งมามีการทำ transaction แต่ถ้าตัวมันเอง เป็นตัวเริ่ม (root ไม่ได้ถูกใครสร้างมาอีกที) หรือ context ที่สร้างมัน ไม่ทำ transaction มันก็จะไม่ทำ transaction

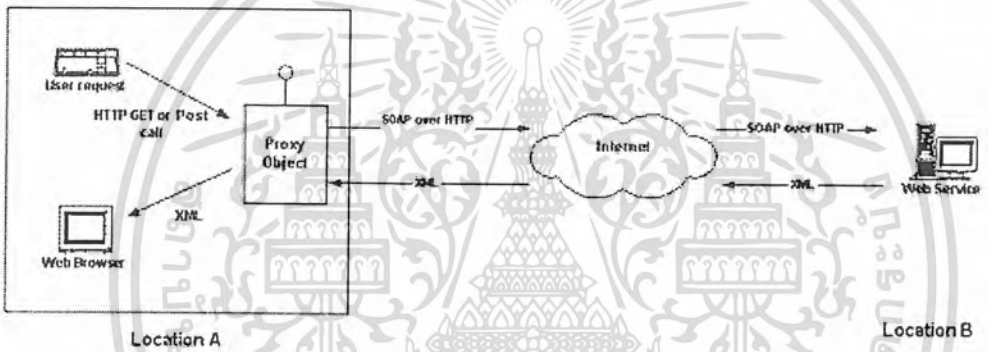
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. Required ถ้า object นั้นจะถูกรันใน context transaction ของตัวที่สร้างมันขึ้นซึ่งมามีการทำ transaction , แต่ถ้าตัวมันเอง เป็นตัวเริ่ม(root ไม่ได้ถูกใครสร้างมาอีกที) หรือ context ที่สร้างมัน ไม่ทำ transaction มันก็จะทำ transactionใหม่
5. RequiresNew - บอกว่า object ต้องทำ transaction ใหม่

2.6.3 การเรียกใช้งานเว็บเซอร์วิส

ในการเรียกใช้งานเว็บเซอร์วิสนั้นเราสามารถที่จะเรียกใช้ได้ทั้ง 3 แบบ คือ HTTP GET, HTTP POST , SOAP

HTTP GET นั้นเราสามารถเรียกใช้ Web Service ได้โดยตรงผ่านทาง URL โดยใส่ parameter รวมไปด้วยเลย เช่น http://localhost/Xtest.asmx/Add?num1=3&num2=6 แต่ในส่วนของ HTTP POST และ SOAP นั้นเราต้องทำการสร้าง Proxy Class ขึ้นมาก่อนเพื่อทำการติดต่อกับ Web Service



รูปที่ 2-9 แสดงการติดต่อจากไคลเอนต์ไปยังเว็บเซิร์ฟเวอร์ผ่านพร็อกซี่อ็อบเจ็กต์

ขั้นตอนการสร้าง Proxy Class

1. ใช้ wsdl.exe ในการคอมไพล์โดยใส่พารามิเตอร์เป็น ชนิดโปรโตคอล, ตำแหน่งของเอกสาร wsdl ภาษาที่ใช้ และอื่นๆ
2. จะได้ไฟล์ออกมา .cs ซึ่งเป็นไฟล์ของ C# เราใช้ตัวคอมไพล์ของ C# ก็จะได้ไฟล์ .dll ออกมา
3. นำไฟล์ .dll นี้ไป add reference ในโปรเจ็คที่จะเรียกใช้

2.6.4 การจัดการสเตทในเว็บเซอร์วิส

ในส่วนของ ASP.NET มีฟีเจอร์ในการจัดการสเตท ทั้งในส่วนของระดับแอพพลิเคชันและระดับของเซสชั่นของผู้ใช้ ในส่วนนี้เราจะกล่าวถึงการใช้งานสเตทของ ASP.NET เพื่อนำมาพัฒนาสเตทฟูลเว็บเซอร์วิส

คำว่า สเตท นั้น เราอาจจะนิยามได้ว่า มันคือความสามารถของอ็อบเจ็กต์,ตัวแปร หรือคอนเทนเนอร์อื่นๆที่สามารถจดจำค่าของมันได้ ซึ่งเรามีอยู่สองระดับคือ แอพพลิเคชัน และเซสชั่น ในระดับแอพพลิเคชันนั้น เปรียบเหมือนร้านอาหาร บริกรต้องจำได้ว่า วันนี้ร้านมีเมนูพิเศษอะไรบ้าง ซึ่งจะนำไปบอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลูกค้าทุกโต๊ะเหมือนกันหมด แต่ระดับเซสชั่น บริกรจะจำได้ว่า ลูกค้าคนนี้สั่งอะไร แต่ละคนสั่งคนละอย่างไม่เกี่ยวกันทั้งสองระดับที่กล่าวมานั้นใน ASP.NET เรียกว่า แอพพลิเคชันอ็อบเจ็กต์ ซึ่งให้บริการกระบวนการในกาจัดเก็บข้อมูล ซึ่งสามารถเข้าถึงได้จากทุกโค้ด ที่อยู่ในเว็บแอพพลิเคชันนั้น และเซสชั่นอ็อบเจ็กต์ ซึ่งจัดเก็บข้อมูลตามเซสชั่นของไคลเอนท์แต่ละคนที่เข้ามา

2.6.5 แอพพลิเคชันอ็อบเจ็กต์

แอพพลิเคชัน อ็อบเจ็กต์จะถูกสร้างขึ้น หลังจากเกิดการเรียกใช้เว็บหรือเว็บเซอร์วิสนั้นขึ้นเป็นครั้งแรกจนกระทั่ง ปิดระบบเว็บเซอร์วิสลงไป ซึ่งแอพพลิเคชันอ็อบเจ็กต์นั้นให้การข้อดีต่างๆดังเช่น

- ง่ายต่อการเรียกใช้สเตท และสามารถใช้จากภาษาใดๆก็ได้ที่ .NET รองรับ
- มีฟังก์ชั่นในการจัดการ ซิงโครไนเซชั่น ทำให้นักพัฒนาง่ายต่อการจัดการการเข้าถึง ตัวแปรแบบโกบอลใน แอพพลิเคชันสเตท
- แอพพลิเคชันสเตท เข้าถึงได้เพียงโค้ดในส่วนของแอพพลิเคชันนั้นๆเท่านั้น ไม่สามารถที่แอพพลิเคชันอื่นจะเข้าถึงได้

ตัวอย่างการเรียกใช้งานแอพพลิเคชัน อ็อบเจ็กต์

```
Application["DatabaseName"] = "DataStore" ;
```

2.6.6 เซสชั่นอ็อบเจ็กต์

เซสชั่นหมายถึง ช่วงเวลาที่ผู้ใช้คนฟนึ่งๆทำการติดต่อกับเว็บแอพพลิเคชันหนึ่ง ซึ่งใน ASP.NET มีการทำงานที่ดีขึ้นกว่า ASP เวอร์ชั่นก่อนหน้าดังนี้

- เซสชั่น อ็อบเจ็กต์โปรเซสจะถูกแยกออกมา ไม่ได้ใน3629 .อยู่ในโปรเซสของ ASP เหมือนของเดิม ดังนั้นถ้าโปรเซสของ ASP ล้มเหลว ตัวเซสชั่นก็ยังอยู่
- เซสชั่น อ็อบเจ็กต์ใน ASP.NET ให้การสนับสนุนการใช้ เซิร์ฟเวอร์หลายๆตัว(server farm) จากแนวคิดของ out-of-process ข้างต้น เราจึงสามารถทำให้ เซิร์ฟเวอร์หลายๆใช้เซสชั่นสเตท ร่วมกันได้
- เซสชั่นอ็อบเจ็กต์ สามารถเรียกใช้โดยไม่ผ่าน คุกกี้ ได้ง่ายการจัดการของเซสชั่นนั้น จะบอกได้จาก เลขเซสชั่นไอดี ซึ่งเป็นเลข 128 บิต หรือเรารู้จักกันดีว่า GUID's(Globally Unique Identifier) ซึ่งถูกสร้างขึ้นจากอัลกอริทึมที่มั่นใจว่าจะไม่เกิดการซ้ำกัน เลขเซสชั่นนี้จะถูกจัดเก็บในสองรูปแบบ คือ
    1. ผ่านคุกกี้ ซึ่งคุกกี้ นั้นสามารถบ่งบอกถึงเซสชั่นของไคลเอนต์
    2. จากการรวมค่าของไอดี เข้าไปยัง URL โดยตรง

เว็บเซอร์วิสนั้นก็ต้องการตัวในการจัดการสเตทกับไคลเอนต์ เครื่องไคลเอนต์ที่จะเรียกใช้เว็บเซอร์วิสต้องสนับสนุนการใช้งานคุกกี้ด้วย เพื่อที่จะเก็บ เซสชั่นไอดี

ตัวอย่างการใช้งาน เซสชั่นอ็อบเจ็กต์ใน ASP.NET

```
Session["UserName"] = "John Smith" ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.7 โหมดการจัดการเซสชั่นและการคอนฟิกค่า

การคอนฟิกค่าต่างๆใน ASP.NET แบ่งออกได้เป็นสองส่วนด้วยกัน คือ machine.config และ web.config ไฟล์แรกนั้นจะส่งผลกับทุกเว็บแอพพลิเคชันในเครื่อง และ ไฟล์หลังจะเป็นการคอนฟิกเฉพาะแอพพลิเคชันซึ่งจะทับแมชชีนคอนฟิกอีกที

โหมดในการใช้งานเซสชั่นนั้นมีด้วยกัน 3 โหมด ได้แก่

1. In Proc เป็นค่าเริ่มต้นที่ติดตั้ง คือจะจัดเก็บเซสชั่นในโปรเซสเดียวกับเว็บเซิร์ฟเวอร์
2. State Server ทำการจัดเก็บลงในแบบ out-of-process ลงในเมมโมรี่ โดยเครื่องที่จะทำเซิร์ฟเวอร์นี้ สามารถเป็นเครื่องเดียวกับ เว็บเซิร์ฟเวอร์ได้

   stateConnectionString="tcpip=127.0.0.1:42424"
3. SqlServer ทำการจัดเก็บลงในแบบ out-of-process เช่นกัน แต่จัดเก็บลงในดาต้าเบส

   sqlConnectionString="data source=127.0.0.1;user id=sa;password="

ค่าอื่นๆที่สามารถปรับแต่งได้ก็คือ โหมดของการใช้ คุกกี้ ว่าจะเป็นแบบ ใช้คุกกี้ หรือว่า ผ่าน url และการกำหนดค่าช่วงเวลาของคุกกี้

cookieless="true"

timeout="40"

ในเรื่องของประสิทธิภาพ โหมด In Proc มีประสิทธิภาพมากกว่า State Server และ SqlServer ตามลำดับ แต่ สองโหมดหลังจะสามารถใช้กับแบบ เซิร์ฟเวอร์ฟาร์มได้ โหมด SqlServer ก็จะช้าที่สุดแต่ก็มีความน่าเชื่อถือที่สุดเช่นกัน

2.7 ไมโครคอนโทรลเลอร์

ไมโครคอนโทรลเลอร์ตระกูล MCS -51 เป็น ไมโครคอนโทรลเลอร์ แบบชิพเดี่ยวที่ผลิตโดยบริษัทอินเทลมีอยู่ด้วยกันหลายเบอร์ด้วยกันซึ่งแต่ละเบอร์นั้นมีคุณสมบัติที่แตกต่างกันตามแต่จะเลือกใช้งานให้เหมาะสมโดยทั่วไปแล้วคุณสมบัติที่แตกต่างกันในแต่ละเบอร์ก็คือ จำนวนหน่วยความจำรอมและแรม, บิต อินพุตเอาท์พุต

2.7.1 คุณสมบัติของไมโครคอนโทรเลอร์ MCS-51

- ต้องการแหล่งจ่ายไฟ +5 โวลต์ ชุดเดียว
- มีหน่วยความจำโปรแกรม (Program Memory) ขนาด 4 กิโลไบต์สำหรับเบอร์ 8051 และ8031 สำหรับเบอร์ 8052 มีหน่วยความจำถึง 8 กิโลไบต์-
- มีหน่วยความจำสำหรับเก็บข้อมูล (Data Memory) ขนาด 128 ไบต์ สำหรับเบอร์ 8052 ขึ้นไปมีถึง256 ไบต์
- หน่วยความจำสำหรับเก็บโปรแกรมและข้อมูลแยกจากกันอย่างละ 64 กิโลไบต์
- มีไทม์เมอร์ เคาน์เตอร์ ขนาด 16 บิต 2 ชุด (สำหรับ 8052 มี 3 ชุด) ทำงานได้ 4 โหมด
- รับอินเตอร์รัพท์ได้ 6 แหล่ง 5 เวคเตอร์ สำหรับเบอร์ 8052 ขึ้นไปมี 8 แหล่ง 6 เวคเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- มีพอร์ทรับส่งข้อมูลอนุกรม (UART) 2 พอร์ทแบบรับ-ส่งพร้อมกัน2ทิศทาง (Full Duplex) เลือกรูปแบบได้ 4 โหมด

- มีคำสั่งในการทำ AND, OR หรือ Complement ได้ทั้งแบบ 8 บิตและ 1 บิต

2.7.2 โครงสร้างของไมโครคอนโทรลเลอร์ตระกูล MCS – 51

เราจะกล่าวถึงรายละเอียดคร่าวๆของไมโครคอนโทรลเลอร์ตระกูล MCS – 51 โดยมีจุดประสงค์เพื่อให้ผู้อ่านทำความเข้าใจและเห็นภาพกว้างๆ ของไมโครคอนโทรลเลอร์ตระกูลนี้เพื่อเป็นพื้นฐานในการศึกษารายละเอียดต่อไป ไมโครคอนโทรลเลอร์ตระกูล MCS -51 มีสมาชิกในตระกูลหลายเบอร์ด้วยกันแต่ละเบอร์จะมีคุณสมบัติพิเศษบางอย่างแตกต่างกัน เช่น มีหน่วยความจำภายในสำหรับเก็บโปรแกรมและข้อมูลภายในชิพเพิ่มขึ้น มีวงจรเปลี่ยนค่าสัญญาณอนาล็อกเป็นดิจิตอลในตัว สามารถรับสัญญาณอินเตอร์รัปต์ได้หลายชนิด ทำกระบวนการ DMA (Direct Memory Access) ได้ในตัว มีรีจิสเตอร์สำหรับใช้เป็นไทม์เมอร์หรือเคาน์เตอร์เพิ่มขึ้น คุณสมบัติพิเศษที่แตกต่างกันของไมโครคอนโทรลเลอร์แต่ละเบอร์ในตระกูล นี้

ไมโครคอนโทรลเลอร์เบอร์ที่นับได้ว่าเป็นเบอร์พื้นฐานสำหรับตระกูล MCS -51 นี้ได้แก่เบอร์ 8051, 8031 , 8751 โดยเบอร์ 8051 จัดเป็นสมาชิกตัวแรกในตระกูลซึ่งมีหน่วยความจำสำหรับเก็บโปรแกรมภายในชิพเป็น ROM ขนาด 4 กิโลไบต์ และหน่วยความจำ สำหรับเก็บข้อมูลทั่วไปภายใน MCS -51 (RAM) เองจำนวน 128 ไบต์ มีพอร์ตขนาด 8 บิต 4 พอร์ต มีรีจิสเตอร์สำหรับใช้เป็นไทม์เมอร์หรือเคาน์เตอร์ขนาด 16 บิตรวม 2 ตัว รับสัญญาณอินเตอร์รัปต์ จากภายนอกได้ 2 ชนิดสามารถรับส่งข้อมูลแบบอนุกรมผ่านทางพอร์ตสื่อสารข้อมูลแบบอนุกรม มีวงจรออสซิลเลเตอร์เพื่อ สร้างสัญญาณนาฬิกาควบคุมการทำงานในตัว ส่วนเบอร์ 8751 จะมีคุณสมบัติเหมือนเบอร์ 8051 ทุกอย่างต่างกันเพียงชนิดของหน่วยความจำสำหรับเก็บโปรแกรมภายในชิพของเบอร์ 8751 จะเป็นอีพรอมแทนที่จะเป็นรอมส่วนเบอร์ 8031 จะเหมือนกับเบอร์ 8051 ต่างเพียงในเบอร์ 8031 ไม่มีหน่วยความจำสำหรับเก็บโปรแกรมภายในชิพเท่านั้น

ไมโครคอนโทรลเลอร์ตระกูล MCS -51 ทุกเบอร์ใช้แรงดันไฟเพียง 5 โวลต์ในการทำงานส่วนกระแสไฟฟ้าที่ใช้จะแตกต่างกันไปตามชนิดของเทคโนโลยีที่ใช้ในการผลิต เบอร์ของไมโครคอนโทรล-เลอร์ตระกูลนี้ที่มีตัวอักษร C อยู่ตรงกลางเบอร์ เช่น 80C31, 80C51 จะเป็นเบอร์ของชิพที่ผลิตโดยอาศัยเทคโนโลยี CHMOS ซึ่งใช้พลังงานในการทำงานน้อยกว่าและสามารถควบคุมการใช้พลังงานของตัวชิพได้จากโปรแกรมเพื่อการประหยัดพลังงานในระบบ

MCS -51 เป็นตระกูลของไมโครคอนโทรลเลอร์ที่ถูกพัฒนาขึ้นมาจากตระกูล MCS -48 ดังนั้นจึงมีความสามารถเหนือกว่าหลายอย่าง เช่น ความเร็วในการประมวลผลของ MCS -51 สามารถใช้ความถี่ได้ถึง 12เมกะเฮิรตซ์ หรือบางเบอร์ในตระกูล สามารถใช้ได้ถึง 16 เมกะเฮิรตซ์ ทำให้ช่วงระยะเวลาในการทำงานแต่ละคำสั่งน้อยมาก

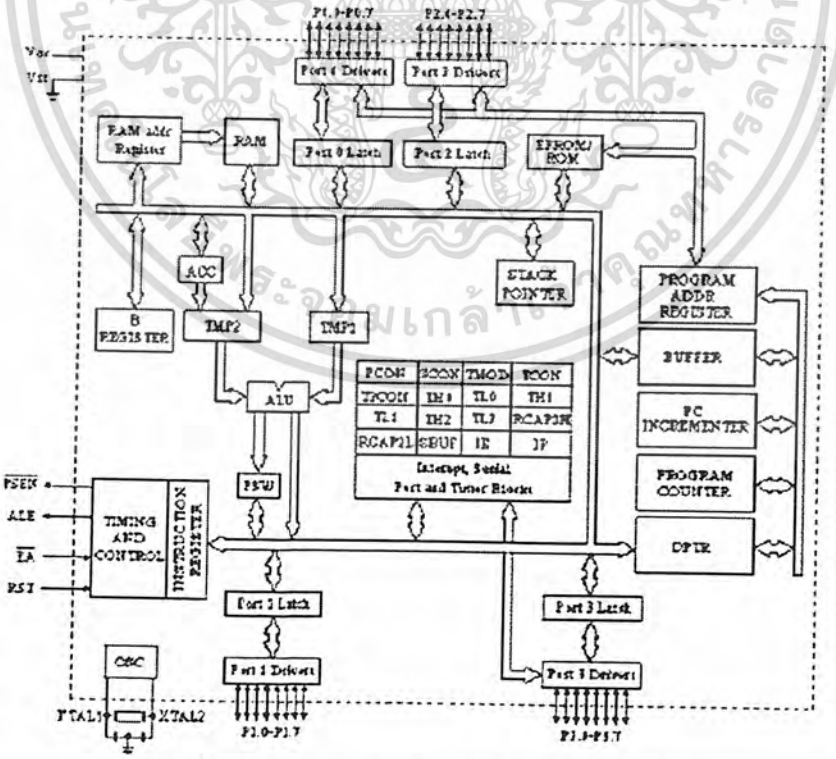
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าที่การใช้งานแต่ละขาของชิพไมโครคอนโทรลเลอร์ตระกูล MCS -51 มีดังนี้

- ขา Vss (ขา 20) สำหรับ ต่อลง GND
- ขา Vcc (ขา 40) สำหรับ ต่อลงแหล่งจ่ายกระแสตรงขนาด 5V (DC)
- ขา พอร์ต 0 (ขา32 - 39) มี 8 ขาใช้เป็นขาสำหรับพอร์ต 0 บิตขนาด8 บิต (P0.0 – P0.7) แบบ Open Drain Bidirectional พอร์ตนี้สามารถใช้งานเป็นอินพุทและเอาท์พุทพอร์ตทั่วไปได้โดย หากใช้งานเป็นอินพุทพอร์ต ต้องโหลดค่า 1 ไปยังแต่ละบิตของพอร์ตนี้เพื่อบังคับให้ขาอยู่ในสถานะถูกปล่อยลอย (มีสถานะ High Impedance) นอกจากใช้งานเป็น อินพุทและเอาท์พุทพอร์ตแล้ว พอร์ต 0 ยังใช้ในการติดต่อหน่วยความจำสำหรับเก็บโปรแกรมและข้อมูลภายนอกชิพด้วย โดยส่งค่าแอดเดรสบัสไบต์ต่ำ(A0 – A7) และมัลติเพลกซ์กับการรับข้อมูล(D0 – D7) จากหน่วยความจำภายนอกในระหว่างการเขียนหรืออ่านข้อมูลโดยมีวงจรพูลอัพภายใน
- ขาพอร์ต 1 (ขา 1 – 8) มี 8 ขา ใช้เป็นขาสำหรับพอร์ต 1 (P1.0 – P1.7) สามารถใช้งานเป็นอินพุทและเอาท์พุทพอร์ต

2.7.3 โครงสร้างภายในของ 8051

ไมโครคอนโทรเลอร์ MCS-51 ใช้เทคโนโลยีในการผลิตเป็นแบบ NMOS และ CMOS เบอร์ 8032 และ 8052 จะมีรอมแบบพื้นฐาน อยู่ภายในจึงสะดวกสำหรับโปรแกรมเมอร์ที่จะเขียนโปรแกรมด้วยภาษาเบสิก โครงสร้างภายในสำหรับเบอร์ 8051 เป็นดังรูปที่ 2-10 โครงสร้างภายในของ 8051



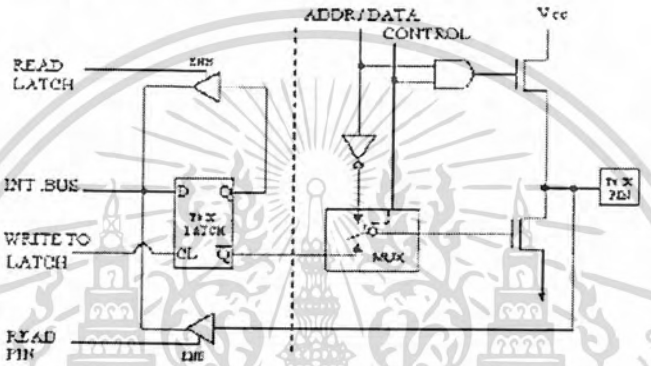
รูปที่ 2-10 แสดงบล็อกไดอะแกรมของ MCS 8051

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Port ต่างๆ ภายใน 8051

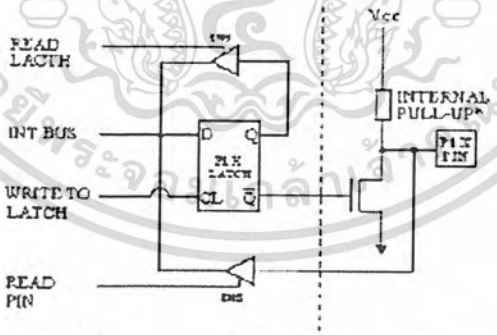
พอร์ทของ 8051 เป็น ไมโครคอนโทรเลอร์ ขนาด 40 ขา ซึ่งมีขาต่าง ๆดังนี้

- Vcc (ขา 40) ต่อกับ +5V
- Vss (ขา 20)เป็นขา GND
- พอร์ท 0 (ขา 32-39) มีทั้งหมด 8 บิต คือ P0.7- P0.0 ใช้งานได้ 2 หน้าที่คือ แอดเดรสบัสและดาต้าบัส เมื่อต้องการให้ทำงานเป็นอินพุตพอร์ทต้องส่งลอจิก "1" ไปยังพอร์ทนี้ จะมีผลให้ Q bar ของ D-FF เป็น "0" ทำให้ FETตัวล่างมีสถานะเป็น OFF สัญญาณที่ใช้อ่านอินพุทพอร์ทแลทช์ โดยส่งสัญญาณ READ LATCH ไปกระตุ้นที่ Tri State Buffer ตัวบนและการอ่านพอร์ท(pin) จะใช้สัญญาณ Read(pin)



รูปที่ 2-11 แสดงโครงสร้าง port 0 (bit)

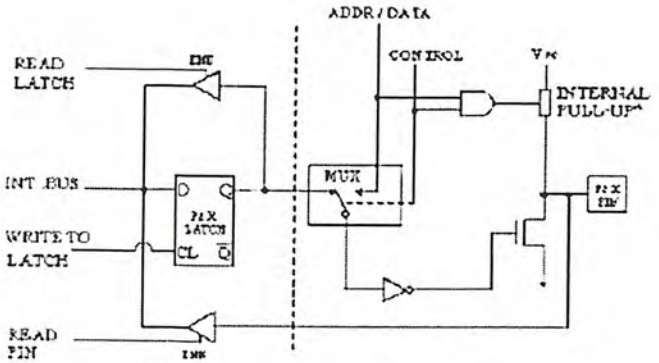
- พอร์ท 1 (ขา 1-8) มีทั้งหมด 8 บิตคือ (P1.0- P1.7) มีโครงสร้างคล้าย พอร์ท 0 แต่จะใช้ความต้านทานภายในพูลอัพแทน (Internal Pull up Register) มีโครงสร้างดังรูปที่ 2-12



รูปที่ 2-12 แสดงโครงสร้าง port 1 (bit)

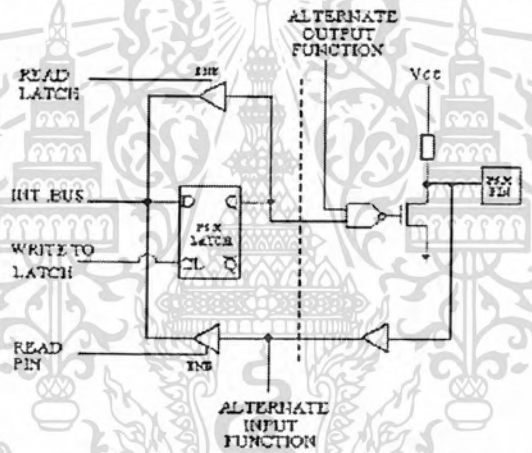
- พอร์ท 2 (ขา 21-28) มีทั้งหมด 8 บิต คือขา (P2.7- P2.0)มีโครงสร้างคล้าย พอร์ท 0 โดยมี FETตัวล่างตัวเดียวส่วนด้านบนใช้ความต้านทานพูลอัพแทน(Internal Pull up) พอร์ทนี้ทำงาน 2 หน้าที่ คือ สามารถใช้เป็น แอดเดรสบัสขนาด 8 บิต และเป็น ไอโอพอร์ทใช้งานทั่วไปเมื่อจะใช้งานเป็นอินพุทพอร์ทต้องส่งลอจิก "1"มาที่พอร์ทนี้ก่อนเพื่อบังคับให้ FETอยู่ในสภาวะ off ดังรูปที่ 2-13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-13 แสดงโครงสร้าง port 2 (bit)

- พอร์ท 3 (ขา 10-17) มีทั้งหมด 8 บิต คือ ขา P3.7-P3.0 มีโครงสร้างคล้าย พอร์ท 1 ทำงานได้ 2 หน้าที่ คือ เป็นไอโอพอร์ท ถ้าจะโปรแกรมให้เป็น อินพุทพอร์ทต้องส่ง ลอจิก "1" มาที่พอร์ทนี้ก่อน



รูปที่ 2-14 แสดงโครงสร้าง port 3(bit)

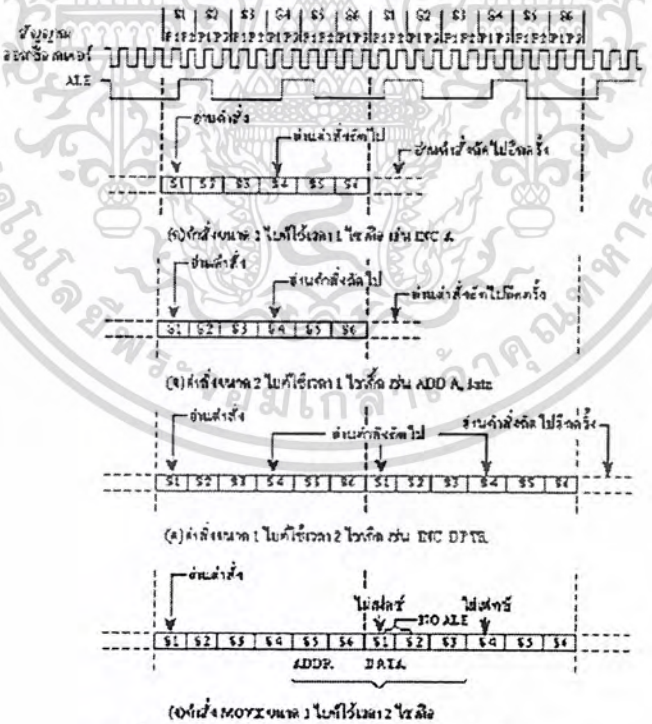
อีกหน้าที่หนึ่งก็คือ ใช้ส่งสัญญาณควบคุมออกมาและรับสัญญาณเข้าไปสัญญาณต่าง ๆ ดังนี้

- **P3.0/RXD (Serial Input Port)** เป็นขาที่ใช้รับข้อมูลแบบอนุกรม (UART)
- **P3.1/TXD (Serial Output Port)** เป็นขาที่ใช้ส่งข้อมูลแบบอนุกรม (UART)
- **P3.2/INT0 (External Interrupt 0)** ใช้รับสัญญาณการขัดจังหวะจากภายนอกเบอร์ 0
- **P3.3/INT1 (External Interrupt 1)** ใช้รับสัญญาณการขัดจังหวะจากภายนอกเบอร์ 1
- **P3.4/T0 (Counter 0 External Input)** ขารับสัญญาณพัลซ์อินพุทเข้าไปยังวงจรเคาท์เตอร์0
- **P3.5/T1 (Counter 1 External Input)** ขารับสัญญาณพัลซ์อินพุทเข้าไปยังวงจรเคาท์เตอร์1
- **P3.6/WR (External Data Memory Write Strobe)** ขาสัญญาณควบคุมการเขียนข้อมูลลงหน่วยความจำข้อมูลภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **P3.7/RD (External Data Memory Read Strobe)** ขาสัญญาณควบคุมการอ่านข้อมูล จากหน่วยความจำข้อมูลภายนอก
- **ALE** (ขา 30) เป็นขาส่ง Strobe สำหรับใช้ในการแล็ชแอดเดรสไบต์ ต่ำ (A7-A0) ที่ส่งออกมาจาก port 0สัญญาณนี้จะ active ทุก ๆ 2 ครั้งในแมชชีนไซเกิล
- **PSEN** (ขา 29) เป็นขา Strobe สำหรับใช้อ่านข้อมูลจากโปรแกรมเมมโมรี่ภายนอกสัญญาณนี้จะส่งออกมา 2 ครั้งในแต่ละแมชชีนไซเกิลแต่ถ้าเป็นการอ่านโปรแกรมเมมโมรี่ภายในจะไม่มีสัญญาณออกที่ขานี้
- **EA** (ขา 31) ใช้เลือกหน่วยความจำโปรแกรมภายนอก ป้อน " 0 " จะอ่านโปรแกรมจากภายนอกชิพ ป้อน "1"จะอ่านโปรแกรมจากภายในชิพ
- **RST** (ขา 9) ขารีเซ็ทจะรีเซ็ทได้ก็ต่อเมื่อป้อนลอจิก " 1 " เข้าที่ขานี้นานอย่างน้อย 2 แมชชีนไซเกิล
- **XTAL1** (ขา 19) ใช้ต่อคริสตอลภายนอกโดยเป็นอินพุทเข้าสู่วงจรออสซิสเลเตอร์ภายใน
- **XTAL2** (ขา 18) ใช้ต่อคริสตอลภายนอกโดยเป็นเอาท์พุทของวงจรออสซิสเลเตอร์ภายใน

2.7.4 ผังเวลาของ CPU (CPU Timing)



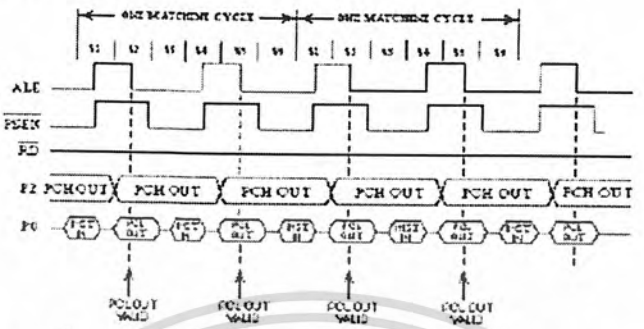
รูปที่ 2-15 แสดงการทำงานของคำสั่งต่างๆ

การทำงานใน 1 คำสั่งต่ำสุดจะกินเวลาเพียง 1 ไมโครวินาที เช่นคำสั่ง INC A ซึ่งเป็นคำสั่ง 1 ไบต์ 1แมชชีนไซเกิลซึ่งจะใช้สัญญาณนาฬิกาไปเท่ากับ 12 ลูก โดยสัญญาณนาฬิกาลูกที่ 1 และ 2 จะอยู่ในช่วง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

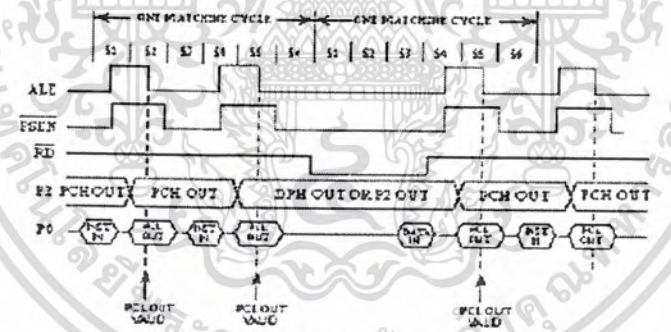
S1 P1และ S1 P2 และ สัญญาณนาฬิกาลูกที่ 12 ก็จะอยู่ในช่วง S6P2 นั่งเอง (ปรกติแล้ว CPU จะทำงานด้วยความเร็วเท่ากับ 12 MHz ดังนั้นสัญญาณนาฬิกา 12 ลูกจะกินเวลาเท่ากับ 1 ไมโครวินาที)

*** คำว่า 1 แมชชีนไซเกิล คือช่วงการทำงานตั้งแต่ S1 จนถึง S6 ***



รูปที่ 2-16 เป็นผังเวลาของสัญญาณซึ่งเกี่ยวข้องกับ Fetch

รูปที่ 2-16 เป็นผังเวลาของสัญญาณซึ่งเกี่ยวข้องกับ Fetch เมื่อส่วนของหน่วยความจำโปรแกรมอยู่ภายนอก ดังนั้น สัญญาณที่จำนำไปใช้อ่านออปโค้ด จากหน่วยความจำโปรแกรมก็คือ PSEN ซึ่งจะ active 2 ครั้งใน 1 แมชชีนไซเกิล ดังนั้น สัญญาณที่ใช้อ่านข้อมูลจาก หน่วยความจำโปรแกรม จะใช้สัญญาณ PSEN



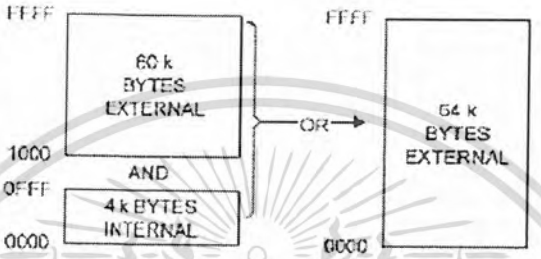
รูปที่ 2-17 แสดงโครงสร้าง port 2 (bit)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

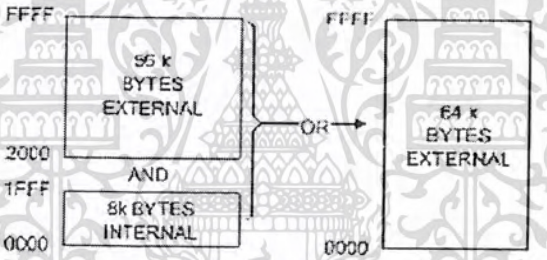
2.7.5 การแบ่งประเภทของหน่วยความจำ

หน่วยความจำที่ใช้กับ MCS –51 มีอยู่ด้วยกัน 2 ชนิด คือ

- หน่วยความจำสำหรับเก็บโปรแกรม (Program Memory)
- หน่วยความจำสำหรับเก็บข้อมูล (Data Memory)
- หน่วยความจำสำหรับเก็บโปรแกรม เป็นหน่วยความจำที่ใช้เก็บโปรแกรมสั่งงานบรรจุอยู่ในชิพ 8051ส่วนที่เป็น หน่วยความจำโปรแกรม ก็คือ ROM ขนาด 4 กิโลไบต์นั่นเอง แต่ถ้าเป็นเบอร์ 8052 จะมีROM ขนาด 8 กิโลไบต์ ดังแสดงในรูปที่ 2-18 และ 2-19 ตามลำดับ



รูปที่ 2-18 ผังหน่วยความจำสำหรับโปรแกรมสำหรับเบอร์ 8051



รูปที่ 2-19 ผังหน่วยความจำสำหรับเก็บโปรแกรมสำหรับเบอร์ 8052

พื้นที่หน่วยความจำที่เข้าถึงข้อมูลทางอ้อมเท่านั้น (Indirect Addreses Area)

พื้นที่หน่วยความจำบริเวณ (80h – FFh) เป็นพื้นที่ซ้อนกันอยู่อย่างละ 128 ไบต์ โดยส่วนแรกจะเป็น SFR address และ Indirect Address Area ดังนั้นถ้าจะติดต่อกับ SFR จะต้องใช้คำสั่งแบบเข้าถึงข้อมูลโดยตรงเท่านั้น (Direct Address Area) ส่วนพื้นที่อีกส่วนหนึ่งจะเข้าถึงข้อมูลแบบทางอ้อมเท่านั้น (Indirect AddressArea) ส่วนตำแหน่ง (00h – 7Fh) จะเข้าถึงข้อมูลได้ทั้ง 2 แบบ

พื้นที่หน่วยความจำที่เข้าถึงข้อมูลโดยตรงและทางอ้อม (Direct and Indirect Address Area)

- Register Bank (Register Banks 0-3) ตั้งแต่ตำแหน่ง (00-Fh) จะเป็นส่วนของรีจิสเตอร์แบงค์ (0-3) โดยแบ่งเป็นแบงค์ แบงค์ละ 8 ไบต์รวมแล้วได้ 32 ไบต์ (แต่ละแบงค์นั้นจะมีรีจิสเตอร์ R0,R1,R2,R3,R4,R5,R6,R7) ถ้า CPU ทำงานอยู่ที่แบงค์ 3 เมื่อถูก reset ก็จะกลับมาทำงานที่แบงค์ 0 เสมอและ SP จะมาเริ่มต้นที่ตำแหน่ง 07h ทันที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- บริเวณหน่วยความจำที่ใช้คำสั่งอ่านเขียนทีละกับบิตได้ (Bit Addressable Area) พื้นที่ตั้งแต่ address (20h-7Fh) จำนวน 16 ไบต์หรือแบ่งเป็นบิตจะได้เท่ากับ 128 บิตซึ่งจะมีตำแหน่งบิตดังนี้ 00,01,02,03,04,05,06,07 จนถึง 7Fh
- บริเวณหน่วยความจำที่ใช้งานทั่วไป (Scratch Pad Area) พื้นที่ตั้งแต่ (30h-7Fh) จะเขียนข้อมูลได้ทีละไบต์เท่านั้น ไม่สามารถใช้คำสั่งเกี่ยวกับบิตได้ถ้าย้ายเนื้อที่แสตคมาบริเวณนี้

2.7.6 พอร์ตสื่อสารข้อมูลแบบอนุกรมใน MCS-51

การใช้งานสื่อสารข้อมูลอนุกรมใน MCS-51 มีความสะดวกและคล่องตัวสูงทั้งนี้เนื่องจากผู้ใช้สามารถกำหนดการทำงานที่แตกต่างกันได้ถึง 4 ประเภทโดยสามารถกำหนดได้จากค่าของบิตในรีจิสเตอร์ใช้งานเฉพาะ SCON การใช้งานที่แตกต่างกัน 4 ประเภทนี้ มีจุดประสงค์ เพื่อความคล่องตัวในการรับหรือส่งข้อมูลแบบอนุกรมแต่ละประเภทดังนี้

- โหมด 0 การทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมในโหมด 0 ขา RXD จะสำหรับรับข้อมูลและส่งข้อมูล ส่วนขา TXD มีไว้เพื่อใช้สร้างสัญญาณ shift clock เพื่อกำหนดจังหวะในการรับและส่งข้อมูลในโหมดนี้ การรับส่งข้อมูลจะเป็นแบบ 8 บิต โดยเริ่มนับและส่งบิตต่ำสุดก่อน อัตราการรับส่งข้อมูลในการทำงานโหมด 0 ถูกกำหนดไว้ที่ 1/12 ของความถี่ออสซิลเลเตอร์ที่ใช้ การทำงานของพอร์ตสื่อสารอนุกรมในโหมด 0 จะไม่มีบิตเริ่มต้นของข้อมูลและบิตสิ้นสุดของข้อมูล เพราะจังหวะในการรับและส่งข้อมูลถูกกำหนดจากสัญญาณ shift clock

- โหมด1 ทำงานแบบที่สองหรือการทำงานโหมด 1 นี้ มีการรับส่งข้อมูลครั้งละ 10 บิต ข้อมูลจะถูกส่งออกไปภายนอกผ่านทางขา TXD และ การรับส่งข้อมูลเข้ามาทางขา RXD ข้อมูลทั้ง 10 บิต ประกอบไปด้วยบิตเริ่มต้นของข้อมูล 1 บิต (มีค่าเป็น 0 เสมอ) บิตข้อมูล 8 บิต (รับและส่งบิตต่ำสุดก่อน) และบิตสิ้นสุดข้อมูลอีก 1 บิต (มีค่าเป็น 1 เสมอ) ในขณะทำการรับส่งข้อมูล ค่าในบิตสิ้นสุดของข้อมูลที่ได้รับจะได้ไปอยู่ในบิต RB8 ของ รีจิสเตอร์ใช้งานเฉพาะ SCON อัตราเร็วในการรับหรือส่งข้อมูลของพอร์ตสื่อสารข้อมูลแบบอนุกรมในโหมดนี้สามารถเปลี่ยนแปลงได้

- โหมด 2 การทำงานแบบที่สามหรือโหมด 2 จะมีการรับและส่งข้อมูลครั้งละ 11 บิต ข้อมูลจะถูกส่งออกภายนอกผ่านทางขา TXD และรับเข้ามาผ่านทางขา RXD ข้อมูลที่รับและส่งทั้ง 11 บิตจะประกอบไปด้วยบิตเริ่มต้นของข้อมูล 1 บิต บิตข้อมูล 8 บิต ตามด้วยบิตที่ 9 ซึ่งเป็นบิตที่สามารถกำหนดให้มีค่าเป็นศูนย์หรือหนึ่งก็ได้ และบิตสุดท้ายคือบิตสิ้นสุดข้อมูล ดังนั้นจึงมีจำนวนบิตที่รับส่งทั้งหมด 11 บิต

- โหมด 3 การทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมแบบสุดท้าย คือการทำงานในโหมด 3 ในการทำงานโหมดนี้ข้อมูลจำนวน 11 บิตจะถูกส่งออกทางขา TXD และรับข้อมูลผ่านทางขา RXD ข้อมูลทั้ง 11 บิตประกอบไปด้วยบิตเริ่มต้นของข้อมูล 1 บิต บิตข้อมูล 8 บิตตามด้วยบิตที่ 9 ซึ่งเป็นบิตที่สามารถกำหนดค่าได้เหมือนในโหมดที่ 2 และบิตสุดท้ายคือบิตสิ้นสุดข้อมูล อัตราเร็วในการรับหรือส่งข้อมูลสามารถเปลี่ยนแปลงได้ ซึ่งในโหมดนี้สามารถกำหนดอัตราเร็วในการรับหรือส่งข้อมูลได้ตามความต้องการของผู้ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อัตราเร็วในการรับส่งข้อมูล

Baud rate หมายความถึงอัตราเร็วในการรับหรือส่งข้อมูล โดยใน MCS-51 ค่าอัตราเร็วในการรับส่งข้อมูลจะมีค่าเท่าใดขึ้นอยู่กับการทำงานในแต่ละโหมดของ port สื่อสารข้อมูลแบบอนุกรมดังนี้

$$\text{Baud rate โหมด 0} = \frac{\text{ความถี่ออสซิสเลเตอร์ที่ใช้}}{12}$$

หากใช้คริสตอลความถี่ 12 MHz ค่า baud rate ของ port สื่อสารข้อมูลแบบอนุกรมโหมด 0 จะมีค่าสูงถึง 1 MHz. ในโหมดที่2 ค่า baud rate ขึ้นอยู่กับค่าของ บิต SMOD ที่อยู่ในรีจิสเตอร์ใช้งานเฉพาะ PCON โดยบิต SMOD = 0 ค่า baud rateจะเป็น 1/64 ของความถี่ออสซิลเลเตอร์ที่ใช้ บิต SMOD = 1 ค่า baud rate จะเป็น 1/32 ของความถี่ออสซิลเลเตอร์ที่ใช้หลังจากการ reset MCS-51 ค่าในบิต SMOD จะเป็น 0 เสมอและเราสามารถเขียนสูตรสำหรับคำนวณค่า baud rate ได้ดังสมการนี้

$$\text{Baud rate โหมด 2} = \frac{[\,2^{(\text{SMOD})} \times (\text{ความถี่ออสซิสเลเตอร์ที่ใช้})\,]}{64}$$

หากใช้คริสตอลความถี่ 12 MHz baud rate สูงสุดในการทำงานโหมดนี้คือ 375 K

อธิบายการทำงานพอร์ตสื่อสารข้อมูลแบบอนุกรม (โหมด 1 เพิ่มเติม)

การทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมโหมด 1 ข้อมูล 10 บิต ถูกส่งออกไปภายนอกผ่านขาTXD และรับเข้ามาจากภายนอกผ่านทางขา RXD ข้อมูลที่รับหรือส่งจะประกอบไปด้วย บิตเริ่มต้นของข้อมูล 1 บิต (0) บิตข้อมูล 8 บิตที่เริ่มรับหรือส่งด้วยบิตต่ำสุดก่อน (LSB first) ตามด้วยบิตสิ้นสุดของข้อมูลอีก 1 บิต (1)

ในการรับข้อมูล บิตสิ้นสุดของข้อมูลจะถูกนำไปไว้ในบิต RB 8 ของรีจิสเตอร์ใช้งานเฉพาะ SCONค่า baud rate ถูกกำหนดโดยอัตราการเกิด overflow ของไทม์เมอร์ 1 (timer 1 overflow rate) ส่วนใน 8052 ซึ่งมีไทม์เมอร์ 2 อย่างใดอย่างหนึ่งเพิ่มขึ้นมา ค่า baud rate สามารถถูกกำหนดโดยอัตราการเกิด overflow ของไทม์เมอร์ 1 หรือ ไทม์เมอร์ 2 อย่างใดอย่างหนึ่งหรือสองอย่าง (ใช้ตัวหนึ่งสำหรับการกำหนด + ในการส่งข้อมูลและอีกตัวหนึ่งสำหรับการับข้อมูล) โดยใช้บิต RCLK และบิต TCLK เป็นตัวเลือก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

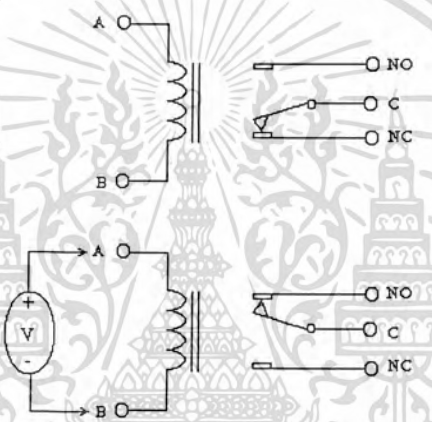
2.8 รีเลย์

2.8.1 หลักการทำงานของรีเลย์

รีเลย์เป็นอุปกรณ์แม่เหล็ก (Magnetic device) ซึ่งภายในโครงสร้างของ รีเลย์ จะประกอบไปด้วย ขดลวด (Coil) 1 ชุด และ หน้าสัมผัส (Contactor) และในหน้าสัมผัส 1 ชุด จะประกอบไปด้วย

- หน้าสัมผัสแบบปกติปิด (Normally Close หรือ NC.) ซึ่งในสภาวะปกติ ขานี้จะต่ออยู่กับขาร่วม (Common)
- หน้าสัมผัสแบบปกติเปิด (Normally Open หรือ NO.)ขานี้จะต่อเข้ากับขาร่วม(Common) เมื่อขดลวดมีแรงดันตกคร่อม หรือกระแสไหลผ่าน (ในปริมาณที่เพียงพอ)

เมื่อขดลวดได้รับแรงดันตกคร่อม (ขา A และ B) จะทำให้มีกระแสไหลผ่านขดลวด ซึ่งจะทำให้เกิดอำนาจสนามแม่เหล็ก ดึงดูดให้หน้าสัมผัส NO และ C ติดกัน



รูปที่ 2-20 แสดงการทำงานของรีเลย์

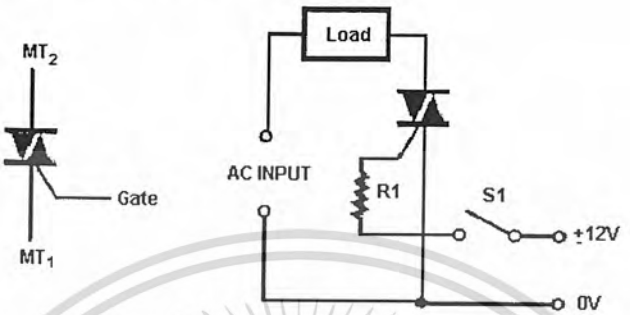
2.9 ไตรแอก

2.9.1 คุณสมบัติของไตรแอกและหลักการทำงาน

1. โดยปกติ ถ้าไม่มีสัญญาณทริกที่เกต ไตรแอกจะไม่ทำงานโดย จะมีลักษณะเหมือนกับสวิตช์ที่ถูกเปิดวงจร
2. ถ้าในกรณีที่ MT2 และ MT1 ถูกป้อนด้วยแรงดันบวกและลบตามลำดับไตรแอกจะถูกกระตุ้นให้ทำงานได้โดยการป้อนสัญญาณพัลส์เพียงสั้น ๆ ที่เกตของมัน โดยจะมีแรงดันตกคร่อมตัวมัน มีค่าประมาณ 1 หรือ 2 โวลต์ เท่านั้น และก็เช่นกันคือเมื่อไตรแอกเริ่มทำงานแล้ว ก็จะสามารถคงสภาพการทำงานอยู่เช่นนั้นต่อไปเรื่อย ๆ ตราบเท่าที่ยังมีกระแสไหลผ่านตัวมันอย่างต่อเนื่อง
3. หลังจากที่ไตรแอกคงสภาพการทำงานอยู่นั้น ทางเดียวที่จะหยุดการทำงานลงได้ ก็โดยการลดปริมาณกระแสที่ไหลผ่านตัวมันลง ให้มีค่าต่ำกว่ากระแสโฮลดิ้งของมัน ในกรณีที่ใช้ไตรแอกในการจ่ายกระแส AC การหยุดทำงานจะเกิดขึ้นอย่างอัตโนมัติ เมื่อแรงดันของไฟสลับเข้าใกล้จุดตัดศูนย์ที่เกิดขึ้น ทุก ๆ ครึ่งคลื่น นั่นคือกระแสจะลดลงเป็นศูนย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ไตรแอกถูกกระตุ้นให้ทำงานได้ ทั้งสัญญาณแบบบวกและลบที่ป้อนให้แก่ขาเกต โดยไม่คำนึงถึงขั้วที่ต่ออยู่ที่ MT1 และ MT2
5. ไตรแอกสามารถทนการกระชากของกระแสได้สูง เช่นโดยปกติสำหรับไตรแอกที่ทนกระแสปกติได้ 10 แอมแปร์ (rms) สามารถทนการกระชากของกระแสในช่วงหนึ่ง คาบเวลาของไฟ 60 เฮิรตซ์ได้สูงถึง 100 แอมแปร์ เป็นต้น



รูปที่ 2-21 สัญลักษณ์และวงจรพื้นฐานของไตรแอก

2.9.2 หลักการควบคุมทางเฟส (Phase Trigger)

หลักการควบคุมทางเฟสนี้ โดยทั่วไปจะใช้ไตรแอกเป็นตัวควบคุมกำลังไฟที่จ่ายให้แก่โหลด โดยแทนที่จะทริกขาเกตด้วยสัญญาณไฟตรงนั้นตรง ๆ ก็จะทริกโดยมีการหน่วงของเฟสด้วยวงจรอีกส่วนหนึ่ง

รูปที่ 2-22 การเปลี่ยนแปลงค่าของกำลังไฟฟ้าที่ป้อนให้แก่โหลด
โดยกำหนดได้จากตำแหน่งเวลาของการทริกที่ให้แก่ไตรแอก

การหน่วงเฟสมีผลดังนี้คือ ถ้าไตรแอกถูกทริกที่ตำแหน่งเฟส 10 องศาหลังจากที่ทุก ๆ ครึ่งรูปคลื่นเริ่มเข้ามากำลังไฟเกือบทั้งหมดก็จะถูกป้อนให้แก่โหลด แต่ถ้าการทริกที่ตำแหน่งเฟส 90 องศาหลังจากทุก ๆ ครึ่งคลื่นเริ่มเข้ามา จะทำให้กำลังไฟที่ป้อนให้แก่โหลดนั้น ลดลงเหลือเพียงครึ่งหนึ่งของกำลังทั้งหมด และถ้าไปทริกที่ตำแหน่งเฟส 170 องศา หลังจากที่ทุก ๆ ครึ่งรูปคลื่นเข้ามาแล้ว จะมีเพียงกำลังไฟส่วนน้อยเท่านั้นที่ป้อนให้แก่โหลด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10 พาสซีฟอินฟราเรดดีเท็กเตอร์(Passive Infrared detector - PIR)

PIR ดักจับทุกความเคลื่อนไหวเป็นอุปกรณ์สำหรับดักจับความเคลื่อนไหวที่มีชื่อเต็มว่า ไพโรอิเล็กตริก เป็นอุปกรณ์จำพวก พาสซีฟอินฟราเรดดีเท็กเตอร์ (Passive Infrared detector - PIR) หรือตัวตรวจจับรังสีอินฟราเรดแบบหนึ่ง

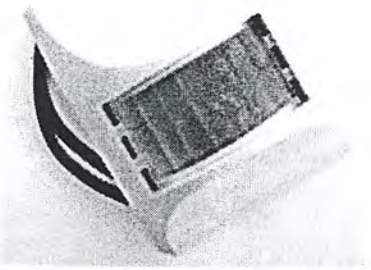
2.10.1 หลักการทำงานของ PIR

การทำงานของ PIR เมื่อมันตรวจจับพบความเปลี่ยนแปลงของรังสี อินฟราเรดที่แผ่ออกมาตัวคนหรือสัตว์ ในขณะที่มีการเคลื่อนไหว ในตัวคนหรือสัตว์จะมีรังสีความร้อนแผ่ออกมารอบๆตัวในปริมาณที่แน่นอนอยู่จำนวนหนึ่ง เมื่อเกิดการเคลื่อนไหวหรือเคลื่อนที่ก็จะทำให้อุณหภูมิในบริเวณนั้นเกิดการเปลี่ยนแปลง ส่งผลให้คลื่นรังสีความร้อนที่ว่านี้แผ่กระจายออกมา มีความยาวคลื่นประมาณ 0.74-300 ไมโครเมตร

ความถี่ในย่านอินฟราเรดพอดีภายใน PIR ประกอบด้วยเลนส์ที่เรียกว่า ฟรีสเนลเลนส์ (fresnel lenses) ซึ่งเป็นเลนส์ที่มีขนาดเล็กจำนวน มากเพื่อสร้างแพตเทิร์นการแทรกสอด (interfered) ของแสงย่านอินฟราเรด ขณะที่ยังไม่มีใครเข้ามาในรัศมีรูปแบบการแทรกสอดของ แสงนั้นจะมีแพตเทิร์นหยุดนิ่งคงที่ แต่เมื่อวัตถุนั้นมีการเคลื่อนไหวเกิดขึ้น แพตเทิร์นการแทรกสอดของคลื่นแสงที่ปรากฏบนตัวเซนเซอร์ PIR ก็จะเปลี่ยนไปเป็นสัญญาณไฟฟ้า ตามการเคลื่อนไหวนั้นออกมาทางขา เอาต์พุต แล้วจะถูกป้อนเข้าสู่ไอซี MPCC เพื่อทำการขยายสัญญาณต่อ

ตัวตรวจจับ มีโครงสร้างภายในที่สำคัญคือ ตัวเซนเซอร์ไวแสง ที่ทำจากผลึกของลิเทียมซัลเฟต 2 ชุด และเฟต 1ตัวประกอบ เข้าด้วยกันในตัวถังแบบ TO-5 ชิ้นของผลึกแร่ขนาด 2*1มิลลิเมตรต่ออนุกรมกันอยู่แต่ต้องต่อกลับขั้วเมื่อสัญญาณรังสีสามารถผ่านกระจกมา ตกกระทบที่ชิ้นสารทั้งสอง ก็ทำให้เกิดความแตกต่างขึ้นตามสัญญาณที่ตกมาตกกระทบ จากนั้นต้องทำการขยายสัญญาณให้แรงขึ้น ก่อนนำไปประยุกต์ใช้งานสัญญาณที่ตรวจจับได้ จะมีความยาวคลื่นอยู่ในช่วง 1-15ไมโครเมตร ความถี่จากผลของการตรวจจับความเคลื่อนไหวจะอยู่ในช่วง 0.3-3 เฮิรตซ์ มีความแรงเพียง 1 มิลลิโวลต์พีกทูพีก ดังนั้นจึงต้องมีการต่อวงจรขยายสัญญาณ ซึ่งในอดีตมักจะใช้ไอซีออปแอมป์ที่มีอัตราขยายสูงๆแต่ผลที่ได้คือวงจรขนาดใหญ่ที่มีอุปกรณ์มากมาย มีความยุ่งยากมากในการสร้างค่อนข้างมากมาในยุคนี้ต้อง เลือกใช้อุปกรณ์ให้ถูกกับงาน สำหรับ PIR ต้องใช้ไอซีพิเศษเฉพาะงานที่เรียกว่า Master PIR Control Chip หรือ MPCC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-23 แสดงพาสซีฟอินฟราเรดดีเท็กเตอร์ (PIR)

2.11 เครื่องตรวจจับควัน(Smoke Detector)

เครื่องตรวจจับควันนี้สามารถตรวจจับกลุ่มควันที่มีขนาดใหญ่ เช่น ควันไฟ ควันบุหรี่ เป็นต้น โดยจะอาศัยการดักจับอิออนของกลุ่มควัน โดยจะมีไอซีชนิดพิเศษสำหรับการตรวจจับควันคือ MC14467-1 พัฒนาโดยบริษัทโมโตโรลา เซมิคอนดักเตอร์ จำกัด ซึ่งไอซีตัวนี้จะมีความสามารถพิเศษคือ เมื่อมีอิออนของควันมาตกกระทบที่ขาตรวจจับควัน ไอซีจะส่งสัญญาณเสียงเตือนผ่านทางลำโพงเพียซโซ ทำให้เกิดเสียงดังขึ้น

รูปที่ 2-24 เครื่องตรวจจับควัน (Smoke Detector)

2.12 ตัวแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล (Analog to Digital Convertor)

2.12.1 ทฤษฎีของตัวแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล

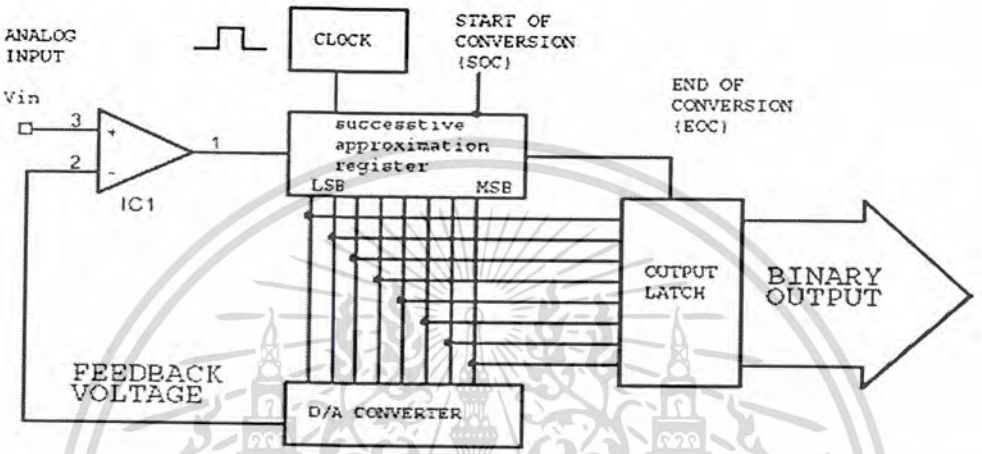
การแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล หรือที่มักเรียกว่า ADC หรือ A/D ใช้สำหรับการแปลงสัญญาณอินพุตที่เป็นอนาลอกให้เป็นดิจิตอลหรือเลขฐานสอง ผลลัพธ์ที่ได้จะอยู่ในรูปแบบของเวิร์ด เทคนิคการแปลงสัญญาณของA/Dมีหลายแบบได้แก่ การแปลงสัญญาณแบบแฟลช การแปลงสัญญาณแบบความชันเดี่ยว การแปลงสัญญาณแบบความชันคู่ การแปลงสัญญาณแบบป้อนกลับ และการแปลงสัญญาณแบบประมาณค่าหลายครั้ง แต่ในที่นี้จะขออธิบายเฉพาะการแปลงสัญญาณแบบประมาณค่าหลายครั้งเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.12.2 เทคนิคการแปลงสัญญาณแบบประมาณค่าหลายครั้ง (Successive Approximation Convertor)

เทคนิคแบบการประมาณค่าหลายครั้งมีชื่อเรียกสั้นๆว่า SA เป็นเทคนิคที่น่าเลือกใช้เพราะมีราคาถูก มีความละเอียดพอสมควรและเป็นตัวแปลงสัญญาณที่มีความเร็วสูง ใช้งานได้ดีและมีประสิทธิภาพสูง เพราะไม่เกิดการออสซิลเลตแต่มีกระบวนการที่เข้าใจได้ยากกว่าเทคนิคอื่นๆ

หัวใจของการประมาณค่าหลายครั้ง คืออุปกรณ์ที่เรียกว่า SAR (Successive Approximation Register) ซึ่งเป็นอุปกรณ์ที่มีจุดประสงค์ต่างจากวงจรนับทั่วไปดังรูปที่ 2-25



รูปที่ 2-25 แสดงบล็อกไดอะแกรมการแปลงค่าแบบ SA

วัฏจักรการแปลงผันเริ่มต้นเมื่อสัญญาณอนาลอกถูกป้อนให้กับตัวแปลงสัญญาณและพัลส์ของการแปลงเริ่มต้น (start of conversation pulse : SOC) ถูกป้อนให้กับตัว SAR พัลส์สัญญาณนาฬิกาลูกแรกที่ป้อนให้กับตัว SAR จะ "on" เอาท์พุตของบิตนับสูงสุด ดังนั้นจึงเป็นการปรับให้เอาท์พุตของ D/A เป็น 50 เปอร์เซ็นต์ของแรงดันเอาท์พุตของอินเวอร์เตอร์ ตัว SAR ตรวจสอบไปยังเอาท์พุตของวงจรเปรียบเทียบว่า เอาท์พุตของ D/A มีค่ามากกว่าหรือน้อยกว่าสัญญาณอนาลอกทางอินพุต ถ้าแรงดัน D/A มีค่ามากกว่าวงจรเปรียบเทียบจะยังคงสภาวะไม่ทำงาน ดังนั้นตัว SAR จะยังคงให้บิตนัยสำคัญสูงสุดทำงานอยู่ เรียกสภาวะนี้ว่า "1" ซึ่งสภาวะ "0" หรือ "1" จะกระทำภายในพัลส์ของสัญญาณนาฬิกาเพียงลูกเดียวบนสัญญาณนาฬิกาถัดไป ตัว SAR จะทำงาน บิตนัยสำคัญอันดับสอง และทำการตรวจสอบผลลัพธ์ที่ได้อีกครั้งหนึ่งจากวงจรเปรียบเทียบ ถ้าสัญญาณจาก D/A ครั้งใหม่มีค่ามากกว่าแรงดันอินพุต เอาท์พุตของวงจรเปรียบเทียบจะยังไม่มี ดังนั้นตัว SAR จะยังไม่ทำงานบิตนัยสำคัญสูงสุดอันดับสอง เรียกว่า "0" แต่ถ้าสัญญาณจาก D/A มีค่าน้อยกว่า วงจรเปรียบเทียบจะทำงานและตัว SAR จะปล่อยให้บิตนัยสำคัญสูงสุดอันดับสองทำงาน

ตัว SAR จะพิจารณาแต่ละบิตด้วยวิธีเดียวกันจนครบทุกบิต เนื่องจากแต่ละบิตหาค่าได้ภายในหนึ่งพัลส์ ฉะนั้น A/D ขนาด 8 บิต จังใช้สัญญาณนาฬิกาเพียง 8 พัลส์ ก็สามารถแปลงได้จนครบ เมื่อบิตนัยสำคัญต่ำสุดถูกพิจารณาแล้ว ตัว SAR จะส่งสัญญาณการแปลง (End of Convertor : EOC) ไปทำการค้างผลลัพธ์ที่ได้ซึ่งเป็นเลขฐานสองทางเอาท์พุตไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

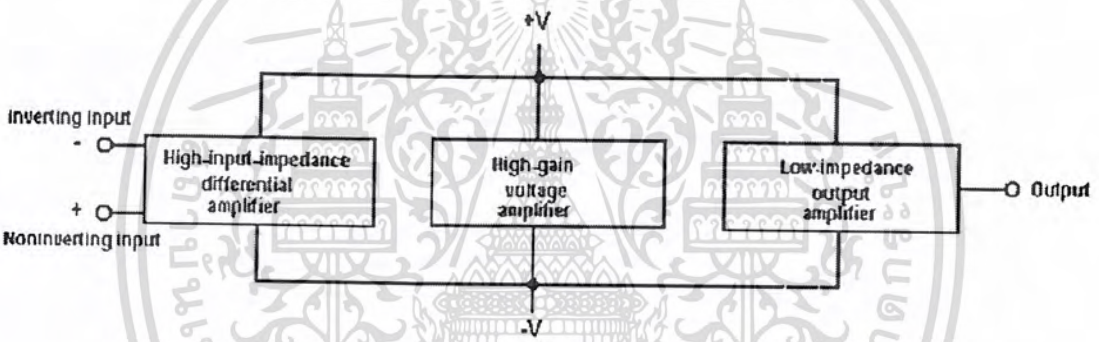
2.13 ออปแอมป์

2.13.1 ทฤษฎีของออปแอมป์

ออปแอมป์ คือ อุปกรณ์ชนิดหนึ่งซึ่งถูกออกแบบมาให้สามารถทำงานได้หลายรูปแบบ และยังเน้นความสะดวกในการนำไปใช้งานอีกด้วย โดยสามารถประกอบเป็นวงจรได้โดยการต่อรวมกับอุปกรณ์ภายนอกเพียงไม่กี่ตัวเท่านั้น ซึ่งคุณสมบัติของไอซี ออปแอมป์ที่พัฒนาขึ้นทำให้อุปกรณ์ชนิดนี้เป็นที่รู้จักกันแพร่หลาย

2.13.2 คุณสมบัติทั่วไปของออปแอมป์

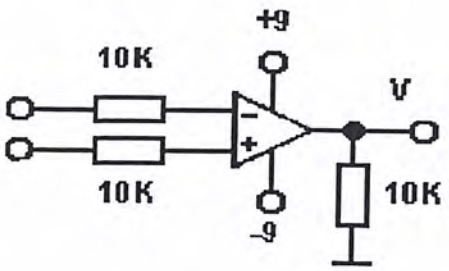
1. เนื่องจากอินพุตอิมพิแดนซ์ของออปแอมป์มีค่าสูงเป็นอนันต์ กระแสเข้าอินพุตจะต่ำจนเกือบเท่าศูนย์หรืออีกนัยหนึ่งไม่มีกระแสอินพุตเข้าสู่ออปแอมป์เลย
2. อัตราขยายขณะเปิดลูปจะมีค่าสูงมาก ซึ่งหมายความว่า แรงดันระหว่างขั้วอินพุตควรจะมีค่าเข้าใกล้ศูนย์
3. เอาท์พุตอิมพิแดนซ์มีค่าต่ำมากจนไม่ทำตัวเป็นโหลดต่อภาคเอาท์พุตของวงจรขยาย



รูปที่ 2-26 การทำงานโดยรวมของออปแอมป์

2.13.3 การเปรียบเทียบแรงดันไฟฟ้า

การทำงานของวงจรคอมพาราเตอร์ในขณะเปิดลูปนั้น ออปแอมป์จะสามารถเปรียบเทียบระดับสัญญาณระหว่างขั้วอินพุตทั้งสองได้อย่างแม่นยำ โดยอาศัยความสัมพันธ์ระหว่างขั้วอินพุต เมื่อแรงดันที่ขั้วลบมีค่าเป็นบวกสูงกว่าแรงดันอินพุตที่ขั้วบวกสัญญาณที่เอาท์พุตจะเป็นลบและมีขนาดเท่ากับ –Vcc จากแหล่งจ่าย ดังรูปที่ 2-27



รูปที่ 2-27 วงจรเปรียบเทียบแรงดัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Input Voltage		Output Voltage
V_1	V_2	$\pm V_{sat}$
+1	+2	+8
+2	+1	-8
0	0	Undefined
+1	-1	-8
-1	+1	+8
-1	-2	-8
-2	-1	+8

ตารางที่ 2-3 แสดงผลการทดลองการเปรียบเทียบแรงดัน

ผลจากการทดลองข้างต้นทำให้เราได้สูตร V_{out}

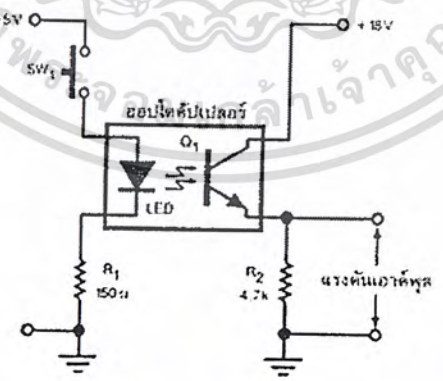
$$V_{out} = V_{sat} * \text{sign}(V_2 - V_1)$$

2.14 ออปโต้คัปเปลอร์ (Opto-Coupler)

ออปโต้คัปเปลอร์เป็นอุปกรณ์ที่ประกอบด้วย LED ซึ่งปกติเป็นชนิดอินฟราเรดและโฟโต้ทรานซิสเตอร์ หรือ โฟโต้ไดโอด ที่ถูกผลิตมาเป็นคู่กัน รวมอยู่ในตัวถังเดียวกัน

2.14.1 การทำงานของออปโต้คัปเปลอร์ (Opto-Coupler)

หน้าที่หลักๆของออปโต้คัปเปลอร์ คือ แยกกราวน์ทางไฟฟ้า (Ground) ของวงจรออกจากกันอย่างสิ้นเชิงดังตัวอย่างในรูปที่ 2-28



รูปที่ 2-28 วงจรออปโต้คัปเปลอร์พื้นฐาน

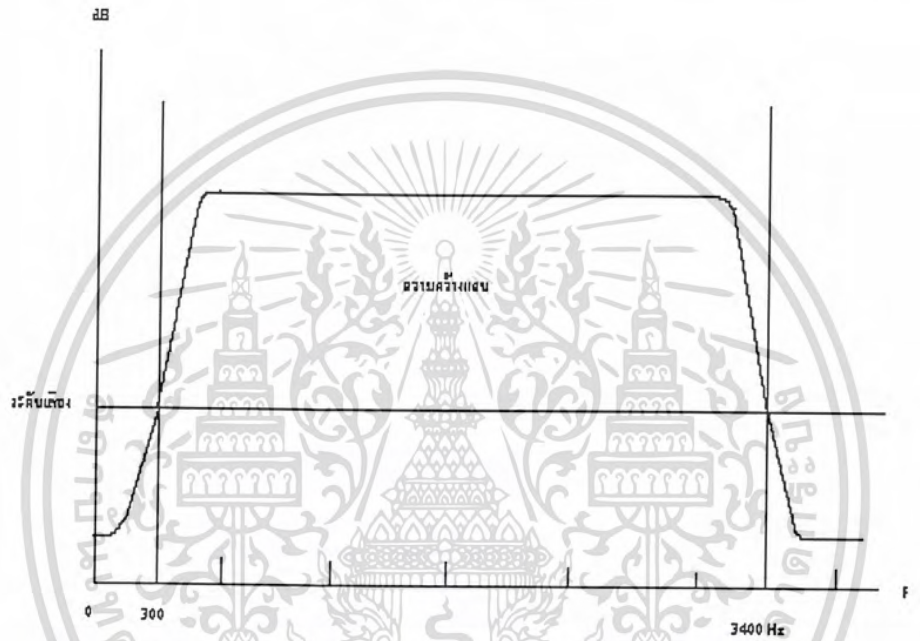
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.15 คลื่นวิทยุ(RADIO FREQUENCY : RF)

2.15.1 ระบบการสื่อสารแบบอนาลอก

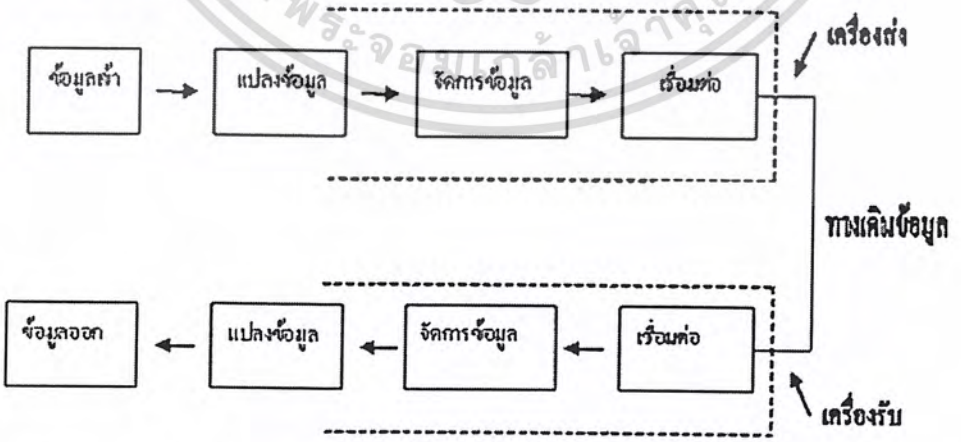
สิ่งที่จะใช้ในการพิจารณาระบบนี้คือ อัตราส่วนของกำลังสัญญาณหลัก ต่อกำลังสัญญาณรบกวน (Signal to Noise Ratio ; S / N) โดยถ้า S / N สูงแสดงว่าระบบนั้นมีประสิทธิภาพดี แต่ถ้ามีค่า S / N ต่ำ แสดงว่าระบบมีประสิทธิภาพไม่ดี

และอีกอย่างก็คือ ค่า แบนด์วิธ (Band Width) ซึ่งหมายถึง ช่วงความถี่ที่ครอบคลุมกำลังส่วนมาก หรือช่วงความถี่ที่มีอัตราขยาย หรือ ค่าลดทอนเล็กน้อยในช่วงกลางๆของ แบนด์วิธ โดยทั่วไปจะกำหนดเขตความกว้างที่ 3 dB หรือครึ่งหนึ่งของกำลังสูงสุด ดังรูปที่ 2-29 ซึ่งมีค่า Band Width เท่ากับ 3,000 Hz ที่ 3 dB



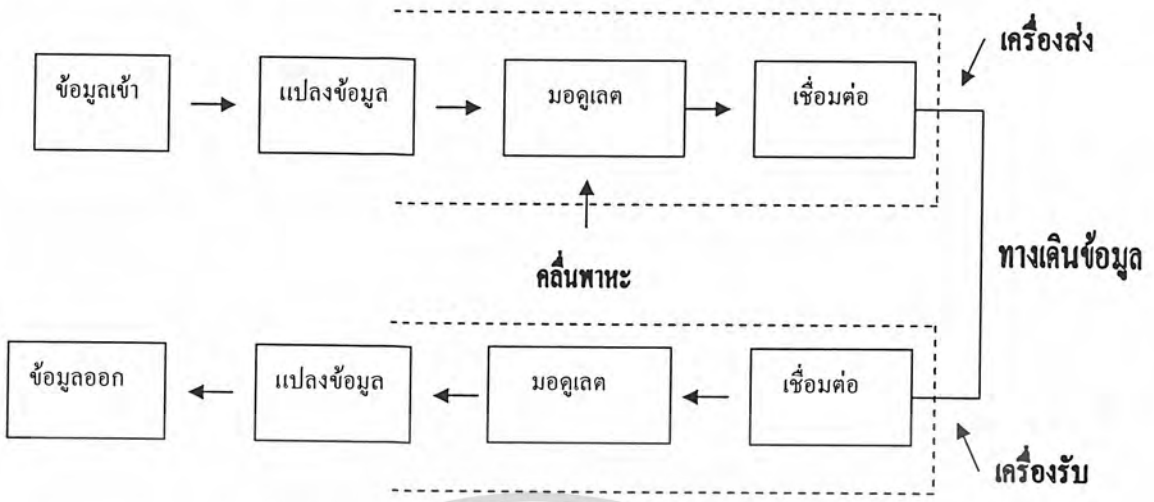
รูปที่ 2-29 การกำหนดความกว้างของแถบ

การ รับ – ส่ง ข้อมูลแบบอนาลอก



รูปที่ 2-30 การ รับ – ส่ง ข้อมูลสัญญาณอนาลอก เบสแบนด์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



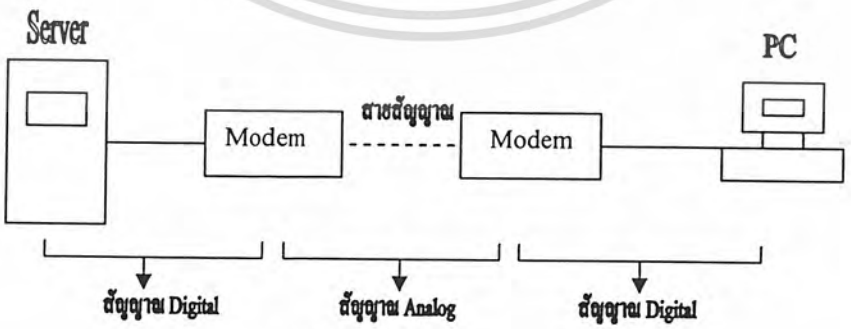
รูปที่ 2-31 การ รับ – ส่ง ข้อมูลสัญญาณอนาลอก ระบบมอดูเลต

รูปที่ 2-30แสดง ระบบ เบสแบนด์ ที่มีลักษณะสำคัญคือ สัญญาณเอาท์พุต ที่ได้ออกมาจะมีสเปคตัมของความถี่เดียวกัน เหมือนกันกับ สัญญาณอินพุต ซึ่งหมายถึง ไม่มีการมอดูเลตกับคลื่นพาหะที่มีความถี่สูงกว่า

ส่วนรูปที่ 2-31 แสดงถึงระบบการสื่อสารแบบอนาลอกที่มีการมอดูเลต และ ดีมอดูเลต นั่นก็คือ เพื่อจะใช้ในการเปลี่ยนรูปสเปคตัมความถี่ให้เข้ากับความถี่ที่ได้เลือกไว้ และเพื่อเป็นการป้องกันไม่ให้สัญญาณที่มีช่วงความถี่เดียวกันอื่นเข้ามารบกวน ซึ่งจะใช้ในการสื่อสารระบบ AM และ FM

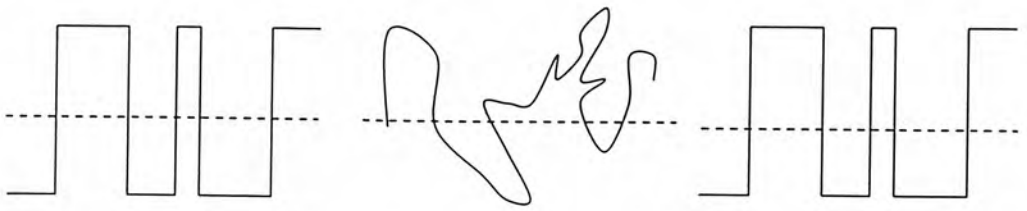
2.15.2 ระบบการสื่อสารแบบดิจิตอล

การสื่อสารแบบนี้จะใช้ข้อมูลเป็นรหัส " 0 " หรือ " 1 " ได้แก่ เลขฐานสอง และ เลขฐานสิบหก เป็นต้น โดยการเปลี่ยนสัญญาณอนาลอกเป็นดิจิตอลนั้น จะใช้วิธีการสุ่มตัวอย่าง (Sampling) จากสัญญาณ Analog โดยวิธีการทางคณิตศาสตร์ ซึ่งจะได้เป็น เลขฐานสองออกมา (0 , 1)



รูปที่ 2-32 การสื่อสารแบบดิจิตอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-33 การสื่อสารทั้งแบบอนาลอกและดิจิตอล

2.15.3 การส่งสัญญาณ

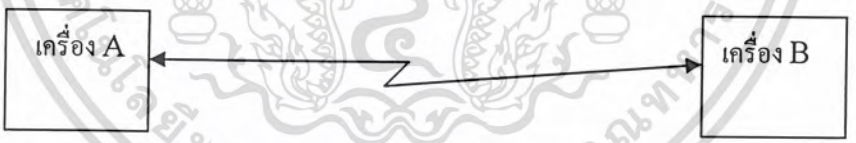
การส่งแบบทิศทางเดียว (Simplex)

หมายถึง การส่งสัญญาณจากด้านส่ง ไปยังด้านรับเพียงด้านเดียว โดยที่ไม่สามารถโต้ตอบระหว่างกันได้ เช่น การกระจายเสียงของสัญญาณวิทยุ หรือ โทรทัศน์ เป็นต้น

รูปที่ 2-34 การส่งสัญญาณแบบทิศทางเดียว

การส่งแบบผลัดกันรับ-ผลัดกันส่ง(Half – duplex)

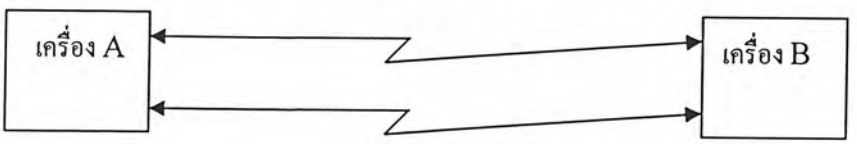
หมายถึง ทั้งทางด้านส่งและด้านรับสามารถโต้ตอบระหว่างกันได้ โดยมีข้อกำหนดว่าต้องมีด้านหนึ่งเป็นด้านรับเสมอ (ผลัดกันส่ง) เช่น วิทยุสมัครเล่น (รับส่งได้ทั้ง 2 ทิศทาง โดยส่งคนละช่วงเวลา)



รูปที่ 2-35 การส่งสัญญาณแบบผลัดกันรับ-ผลัดกันส่ง

การส่งแบบสองทิศทางที่เวลาเดียวกัน (Full – duplex)

หมายถึง การที่ด้านรับและด้านส่งสามารถส่งสัญญาณได้พร้อมกันในเวลาเดียวกัน โดยที่ไม่ต้องผลัดกันส่งเหมือนแบบ Half –duplex (รับส่งได้ทั้ง 2 ทิศทาง ในช่วงเวลาเดียวกัน)



รูปที่ 2-36 การส่งสัญญาณสองทิศทางที่เวลาเดียวกัน

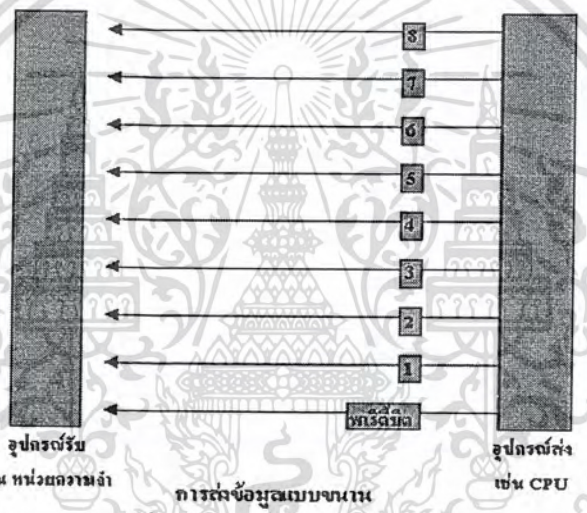
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.15.4 การสื่อสารข้อมูลดิจิตอล (Transmission of Digital Data)

การส่งสัญญาณข้อมูลแบ่งออกได้เป็น 2 ประเภทคือ การส่งแบบขนานและแบบอนุกรม

การสื่อสารแบบขนาน (Parallel Transmission)

การส่งแบบขนานนั้นจะทำการส่งข้อมูลทีละหลาย ๆ บิต เช่น ส่ง 10011110 ทั้ง 8 บิต ออกไปพร้อมกันโดยผ่านสายส่งข้อมูลที่มี 8 เส้น ส่วนการส่งข้อมูลแบบอนุกรม ข้อมูลจะถูกส่งออกไปทีละบิตต่อเนื่องกันไป เช่นถ้าข้อมูลคือ 10011110 เลข 0 ทางขวามือสุดเป็นบิตที่ 1 เรียงลำดับไปจนครบ 8 บิต โดยการส่งนั้นจะใช้สายส่งเส้นเดียวเท่านั้น ดังภาพ แสดงการส่งข้อมูลแบบขนานและแบบอนุกรม ตัวอย่างการใช้งานที่เห็นชัดของการส่งข้อมูลแบบขนาน เช่น การต่อเครื่องพิมพ์เข้ากับเครื่องคอมพิวเตอร์ ซึ่งปกติจะใช้สายยาว 5 เมตร ถึง 10 เมตรเท่านั้นและตัวอย่างการส่งข้อมูลแบบอนุกรม เช่นการต่อเทอร์มินัลเข้ากับคอมพิวเตอร์แม่ที่อยู่ห่างกันสัก 100 เมตร ซึ่งทำให้ประหยัดสาย



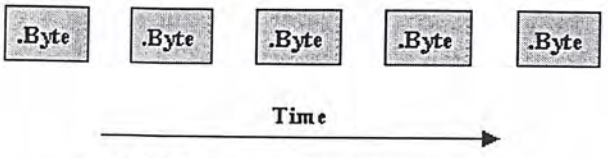
รูปที่ 2-37 การสื่อสารข้อมูลแบบขนาน

การสื่อสารแบบอนุกรม (Serial Transmission)

การส่งข้อมูลแบบอนุกรมแบ่งออกได้เป็น 2 ประเภทได้แก่

1. **การส่งข้อมูลแบบอะซิงโครนัส** มักจะใช้กับเทอร์มินัลธรรมดา (Dump terminal) ไว้สำหรับรับข้อมูลจากคอมพิวเตอร์แม่และแสดงผลที่จอ โดยไม่สามารถเปลี่ยนแปลงข้อมูลได้ การส่งข้อมูลแบบนี้มักจะมีอัตราในการรับส่งข้อมูลที่แน่นอนมีหน่วยเป็นบิตต่อวินาที (bit per second) เมื่ออุปกรณ์อะซิงโครนัสจะส่งข้อมูล 1 ไบต์ ก็จะส่งบิตเริ่มต้น (start bit) ก่อน ซึ่งมักจะเป็น "0" และตามด้วยข้อมูลทั้ง 8 บิตใน 1 ไบต์ แล้วจึงจะส่งบิตหยุด (stop bit) ซึ่งมักจะเป็น "1" บิตทั้งหมดนี้ จะรวมกันเป็น 10 บิต ในการส่งข้อมูลเรียงตามลำดับดังนี้ 1 บิตเริ่มต้น 7 บิตข้อมูล (data bit) 1 บิต ภาวะเสมอมูล และ 1 บิตหยุด กระบวนการเหล่านี้จะห่างกัน 1 วินาที ที่จะส่งข้อมูลชุดต่อไป ซึ่งก็หมายถึงว่าเมื่อคอมพิวเตอร์แม่ได้รับบิตเริ่มต้น ก็คาดหวังว่าจะได้รับอีก 9 บิตภายในเวลา 1 วินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



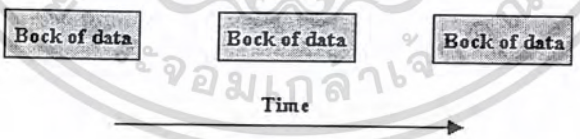
การรับส่งข้อมูลแบบอสมวาร (Asynchronous)

รูปที่ 2-38 การส่งสื่อสารข้อมูลอนุกรมแบบอะซิงโครนัส

ในระบบนี้จะเกี่ยวข้องกับเวลาว่าเมื่อไรบิตต่อไปจะมาถึง ถ้าไม่ตรงตามที่กำหนดไว้ การส่งข้อมูลก็จะล้มเหลว ระบบนี้เหมาะในการส่งอักขระจากเทอร์มินัลมายังคอมพิวเตอร์แม่ทันที เคาะแป้นพิมพ์ของเทอร์มินัลก็จะรู้ทันทีว่าจะต้องส่งไบต์ใดโดยเติมบิตเริ่มต้นและบิตหยุดที่หัวและท้ายของข้อมูลไบต์นั้น ตามลำดับให้ครบ 10 บิตที่จะส่ง ในการส่งข้อมูลอัตราการส่งข้อมูลอาจจะเป็น 110, 300, 1,200, 2,400, 4,800, 9,600, 19,200 บิตต่อวินาที โดยที่ทางด้านส่งและด้านรับต้องตั้งค่าความเร็วให้เท่ากัน

2. การส่งข้อมูลแบบซิงโครนัส จะไม่ใช้บิตเริ่มต้นและบิตหยุด จะไม่มีการการหยุดชั่วขณะระหว่างอักขระ จะใช้วิธีให้จังหวะเวลาทั้งสองทางที่ติดต่อกัน มีอยู่สองวิธีที่ปฏิบัติคือ ใช้อักขระซิงก์ (synchronize character) หรือใช้สัญญาณนาฬิกา (clock signal) การใช้อักขระซิงก์ไว้หน้าบล็อก (block) ของอักขระที่ใหญ่ โดยการใส่อักขระซิงก์ไว้หน้าบล็อกของข้อมูลอักขระซิงก์นี้เป็นบิตจำนวนหนึ่งที่ทางอุปกรณ์เครื่องรับสามารถใช้ในการกำหนดอัตราเร็วของข้อมูลให้ตรงกับทางอุปกรณ์เครื่องส่ง การใช้สัญญาณนาฬิกาของด้านส่ง และสัญญาณนาฬิกาของด้านรับจะใช้คนละสายหรือคนละช่องสัญญาณในการส่งข่าวสารเกี่ยวกับเวลาของ ข้อมูลที่จะส่ง โดยทั่วไปการส่งข้อมูลแบบซิงโครนัสจะทำงานภายใต้การควบคุมของโปรโตคอลในระบบนั้น ๆ และนิยมใช้กับเทอร์มินัลฉลาดและเทอร์มินัลอัจฉริยะ

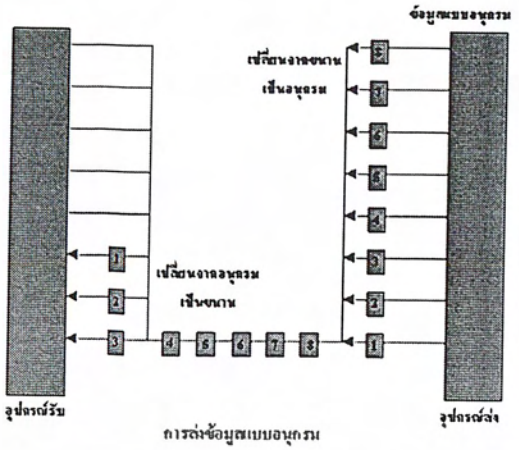
การส่งข้อมูลจะนำข้อมูล 1 ไบท์ มาส่งไปตามสายเรียงกันไปจนครบ 8 บิต ซึ่งเท่ากับ 1 ตัวอักษร



การรับส่งข้อมูลแบบสมวาร (Synchronous)

รูปที่ 2-39 การส่งสื่อสารข้อมูลอนุกรมแบบซิงโครนัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-40 การสื่อสารข้อมูลแบบอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

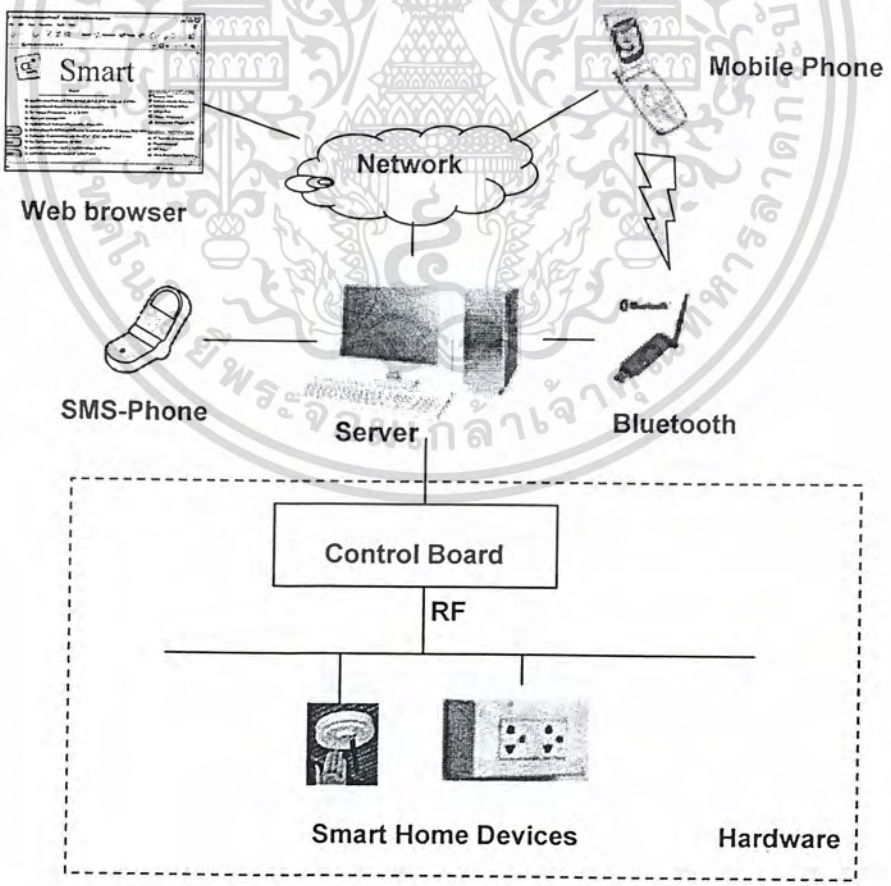
บทที่ 3
โครงสร้างและการออกแบบ

3.1 การออกแบบระบบโดยภาครวม

สมาร์ทโฮมเป็นระบบควบคุมที่ให้บริการผ่านเครือข่าย ที่ต้องการให้โปรแกรมบนอุปกรณ์หรือเครื่องมือต่างๆ สามารถสื่อสารด้วยกันได้ โดยใช้เทคโนโลยีเว็บเซอร์วิสที่เอื้ออำนวยต่อการใช้งานร่วมกันระหว่างหลาย ๆ แอพพลิเคชันผ่านเครือข่ายอินเทอร์เน็ต ไม่ขึ้นกับระบบปฏิบัติการและสถาปัตยกรรม ทำให้การพัฒนาโปรแกรมในอุปกรณ์ต่างๆ ไม่ว่าจะเป็นโทรศัพท์มือถือ (Mobile Phone) พีดีเอ (PDA) เว็บเซิร์ฟเวอร์ (Web Server) หรืออุปกรณ์ฝังตัวต่างๆ (Embedded device) หรือการเชื่อมต่อกันระหว่างเซิร์ฟเวอร์ทำได้โดยไม่ยากเย็น

การสื่อสารภายในระบบใช้การสื่อสารด้วยคลื่นวิทยุ (RF-Radio Frequency) ระหว่างอุปกรณ์ควบคุมต่าง ๆ เพื่อความสะดวกในการติดตั้งและเคลื่อนย้าย และการสื่อสารด้วยบูลทูธ (Bluetooth) สำหรับโทรศัพท์มือถือ เพื่อลดค่าใช้จ่ายในการเชื่อมต่ออินเทอร์เน็ต

อุปกรณ์ที่สามารถใช้งานได้ในระบบ คือ โทรศัพท์มือถือที่สามารถมีจาวารันไทม์ (Java runtime) และเว็บบราวเซอร์



รูปที่ 3-1 แสดงภาพรวมของระบบรักษาความปลอดภัยและอำนวยความสะดวกผ่านโทรศัพท์มือถือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.1 ระบบรักษาความปลอดภัยและอำนวยความสะดวกผ่านโทรศัพท์มือถือ

จะแบ่งชุดการทำงานได้เป็น 2 ชุด คือ ชุดอำนวยความสะดวก และชุดรักษาความปลอดภัย

ชุดอำนวยความสะดวก ประกอบด้วย

- ชุดสวิตซ์ควบคุมอุปกรณ์ไฟฟ้า จะทำหน้าที่ในการสั่งงานเปิด-ปิด อุปกรณ์ไฟฟ้า โดย 1 ชุด สามารถต่ออุปกรณ์ไฟฟ้าได้ 4 ช่อง แต่ละช่องกินกระแสไม่เกิน 3 แอมป์
- ชุดปรับระดับความสว่างของหลอดไฟ สามารถควบคุมความสว่างของหลอดไส้ได้ โดย 1 ชุด สามารถต่ออุปกรณ์ไฟฟ้าได้เพียงช่องเดียว

ชุดรักษาความปลอดภัย ประกอบด้วย

- ชุดป้องการการบุกรุกและอัคคีภัย โดยจะมีเซ็นเซอร์ในการการตรวจจับการผ่านบริเวณ และการตรวจจับเมื่อระบบพบความผิดปกติเกิดขึ้นจะทำการเตือนภัยให้ผู้ใช้ทราบโดยการเปิดสัญญาณไซเรน หรือแจ้งเตือนผ่านทาง SMS โดย 1 ชุด จะต่อกับไซเรน 1 ช่อง ส่วน 8 ช่องที่เหลือเอาไว้ต่อเซ็นเซอร์ตรวจจับควัน (Smoke Detector) และเซ็นเซอร์ตรวจจับการผ่านบริเวณ (PIR Detector)
- ชุดกล้องรักษาความปลอดภัย จะสามารถบันทึกภาพตามเวลาที่ได้กำหนดไว้ หรือจะจับภาภาพ ณ เวลาขณะนั้นด้วยตนเอง โดยภายในภาพจะระบุวันเวลาที่ได้ทำการบันทึกไว้

ทั้งอุปกรณ์ไฟฟ้าและเซ็นเซอร์จะสามารถกำหนดตารางเวลาในการทำงานได้ โดยจะแบ่งการทำงานออกเป็น 4 แบบ คือ

- การทำงานแบบรายวัน (Once) เป็นการทำงานเพียงครั้งเดียวเท่านั้น
- การทำงานแบบรายวัน (Daily) เป็นการทำงานทุกๆ วัน โดยสามารถเลือกได้ว่าต้องการให้ทำงานวันไหนบ้าง ในรอบสัปดาห์
- การทำงานแบบรายสัปดาห์ (Weekly) เป็นการทำงานวันใดวันหนึ่งในรอบสัปดาห์
- การทำงานแบบรายเดือน (Monthly) เป็นการทำงานเดือนละครั้ง

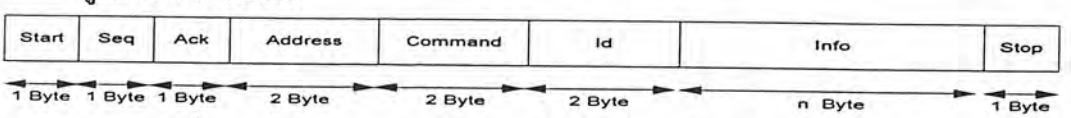
นอกจากนี้ยังสามารถทำงานได้ในแบบ Macro คือเมื่อมีเหตุการณ์ใดๆ เกิดขึ้นกับอุปกรณ์หนึ่ง จะส่งผลไปถึงอุปกรณ์อื่นๆ เช่น เมื่อมีการเปิดไฟในห้องนอนจะทำให้โทรทัศน์ปิดและไฟในห้องนั่งเล่นดับลง เป็นต้น และเมื่อเกิดปัญหาไฟฟ้าดับ อุปกรณ์ยังคงสามารถทำงานตามที่โปรแกรมไว้ได้ต่อเนื่องหลังจากมีไฟฟ้า

3.1.2 การติดต่อสื่อสารระหว่างโฮมเซิร์ฟเวอร์ (Home Server) กับบอร์ดควบคุม (Control board)

รูปแบบการสื่อสาร

ใช้การสื่อสารด้วยพอร์ตอนุกรม (Serial port) ที่อัตราเร็ว 9600 บิตต่อวินาที

รูปแบบเฟรมสื่อสาร



รูปที่ 3-2 แสดงโครงสร้างของเฟรมข้อมูลสำหรับสื่อสารกับบอร์ดควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เฟรม 1 เฟรมจะประกอบด้วยข้อมูลที่มีความหมายต่าง ๆ ดังนี้

- Start คือจุดเริ่มต้นของเฟรม
- Seq คือลำดับเฟรมข้อมูลที่ส่ง
- Ack คือเลขลำดับการตอบรับเฟรมที่สมบูรณ์
- Address คือที่อยู่ของเครื่องใช้ไฟฟ้า เนื่องจากการสื่อสารในมุมมองทางโลจิคอล (Logical view) ผู้รับผู้ส่งจะเกิดขึ้นกับเฉพาะสมาร์ทโฮมเซิร์ฟเวอร์กับเครื่องใช้ไฟฟ้าซึ่งมีหลายตัวเท่านั้น ดังนั้นค่าที่แอดเดรส(Address) ของสมาร์ทโฮมเซิร์ฟเวอร์จึงไม่มีความจำเป็น
    - ผู้ส่งคือเครื่องใช้ไฟฟ้า ค่าแอดเดรส คือที่อยู่ของเครื่องใช้ไฟฟ้าที่ส่งข้อมูล
    - ผู้ส่งคือสมาร์ทโฮมเซิร์ฟเวอร์ ค่าแอดเดรส คือที่อยู่ของเครื่องใช้ไฟฟ้าที่รับข้อมูล
    - FF หมายถึงการทุกๆอุปกรณ์ (หมายถึงเครื่องใช้ไฟฟ้าทุกๆตัว)
- Command คือรหัสคำสั่ง
- Info คือ คำอธิบาย หรือส่วนขยายของคำสั่ง (อาจจะมีหรือไม่มีก็ได้)
- Stop คือ จุดสิ้นสุดของเฟรม

รหัสคำสั่ง Command

คำสั่ง	**เปิด (OFF)**
Code: **Info**	• รหัส 100 • (แชลแนลที่ต้องการ) ความหมาย เปิด ตัวอย่าง (0)(1)
คำสั่ง	**ปิด (OFF)**
Code: **Info**	• รหัส 101 • (แชลแนลที่ต้องการ) ความหมาย ปิด ตัวอย่าง (0)(1)
คำสั่ง	**เปลี่ยนระดับความสว่างหลอดไฟ (DIMMER)**
Code: **Info**	• รหัส 102 • (แชลแนล,ระดับความสว่าง) ความหมาย เปลี่ยนระดับ ตัวอย่าง (1,10)(2,0)(4,5)
คำสั่ง	**อ่านค่าอุปกรณ์ (อ่านค่าอุปกรณ์)**
Code:	• รหัส 200

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Info	• รหัสการอ่านค่า (info code) **ความหมาย** ขอรายละเอียดของอุปกรณ์ **รหัสการอ่านค่า (Info code)** '0' = Power switch ' 1' = Dimmer '2' = Caption '3' = Security **ผลลัพธ์** : ส่งรายละเอียดของอุปกรณ์ทุกแชลแนลตามรหัสการอ่านค่า โดยการส่งรายละเอียดจะใช้เฟรม "รายงานค่าอุปกรณ์"
คำสั่ง	**รายงานค่าอุปกรณ์ (Report)**
Code: **Info**	• รหัส 201 • (ฮาร์ดแวร์แอดเดรส, รหัสคำสั่ง, สถานะทุกแชลแนล, ...) **ความหมาย** ส่งรายละเอียดอุปกรณ์ตามแชลแนล **ตัวอย่าง** (mac,cmdoff,0,1,0,0) **หมายเหตุ** กรณีรหัสคำสั่งของ Security มีความหมายดังนี้ ปกติ=0 ไม่ปกติ=1
คำสั่ง	**ลงทะเบียน (Device Register)**
Code: **Info**	• รหัส 202 • (โมเดลอุปกรณ์,จำนวนแชลแนลที่มี) **ความหมาย** ลงทะเบียนให้สมาร์ทเซิร์ฟเวอร์รู้จัก **Model Code** Sw = 0 Sw-dim = 1 Camera = 2 Security = 3
คำสั่ง	**ผลการทำงาน (Result)**
Code: **Info**	• รหัส 203 • Result Code **ความหมาย** ผลจากการสั่งงานอุปกรณ์ ใช้กับคำสั่งรหัส 100 ถึง 104 ,202, 205 **Result Code** 1 = ok 0 = false or error 2 = timeout (ไม่ได้รับการตอบรับ)

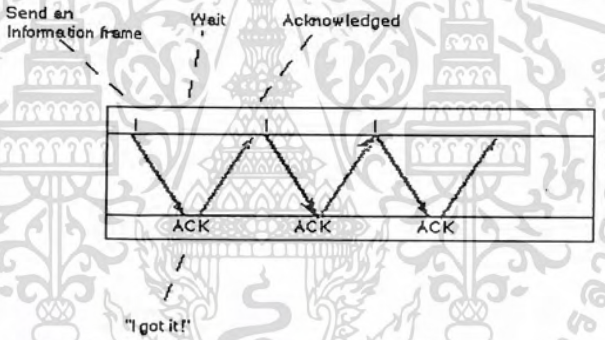
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่ง	ความผิดพลาด (Error)
Code:	• รหัส 204
Info	• Error Code ความหมาย รายงานความผิดพลาดหรือความผิดปรกติ **Error Code** 0 = not define 1 = อุปกรณ์Device not available

ตารางที่ 3-1 แสดงรายละเอียดของโปรโตคอลที่ใช้ในการติดต่อระหว่างเซิร์ฟเวอร์กับบอร์ดควบคุม

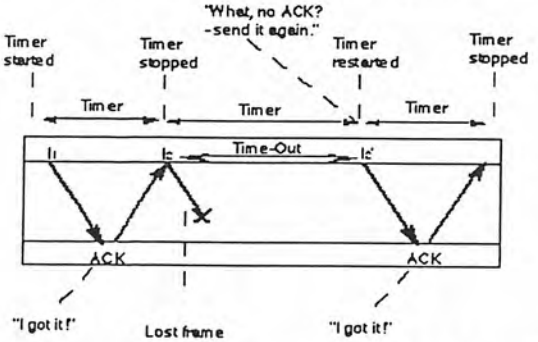
3.1.3 โปรโตคอลการสื่อสาร

เริ่มการสื่อสารทุกครั้งจะมีต้องสถาปนาการเชื่อมต่อก่อน เมื่อสามารถทำการเชื่อมต่อได้แล้วจะใช้โปรโตคอลการสื่อสารแบบสต๊อบแอนเวท (Stop and wait with ARQ) โดยมีหลักการพื้นฐานคือการบังคับให้ผู้รับ ส่งข่าวสารตอบกลับมาที่ผู้ส่ง ก่อนที่ผู้ส่งจะสามารถส่งเฟรมข้อมูลต่อไปได้



รูปที่ 3-3 แสดงโปรโตคอล Stop and Wait with ARQที่มีการรอ Acknowledgement

กรณีเฟรมข้อมูลเกิดสูญหายระหว่างทาง ทางตัวผู้ส่งจะจับเวลาส่ง ถ้าเฟรมตอบรับไม่ได้รับในเวลาที่กำหนด (Timeout) ผู้ส่งจะส่งเฟรมข้อมูลเดิมนั้นใหม่ (Retransmission) และถ้าพยายามส่งใหม่จนถึงเวลาหนึ่งแล้วไม่สามารถส่งให้ถึงผู้รับ ได้หรือไม่มีเฟรมตอบกลับมาจากผู้รับ ก็จะหยุดการส่ง (Terminate) และบอกชั้นที่สูงขึ้นไปว่าไม่สามารถส่งได้ แล้วจะพยายามสถาปนาการเชื่อมต่อครั้งใหม่ไปเรื่อยๆจนกว่าจะได้ ขนาดของวินโดว์ (Window) เท่ากับ 1 ดังนั้นเลขซีแคว้น (Sequence Number) จะมีตั้งแต่ 0 ถึง 1

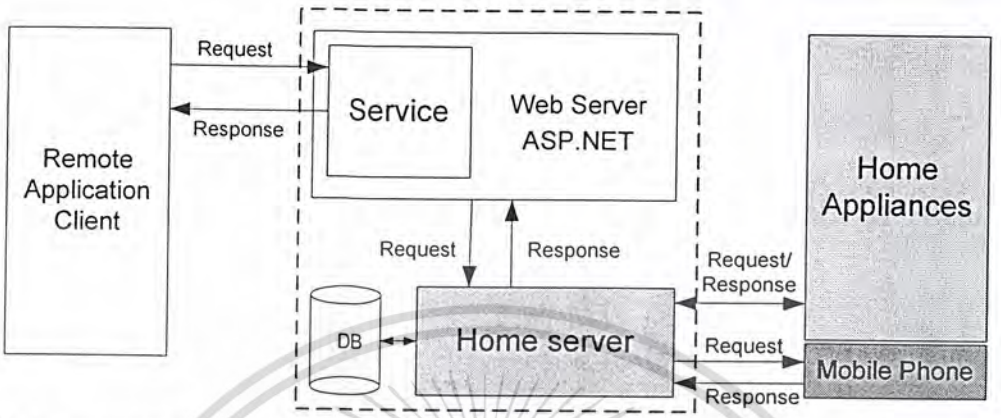


รูปที่ 3-4 แสดงโปรโตคอล Stop and Wait with ARQ ที่มีการส่งข้อมูลซ้ำเวลาเกิด Timeout

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การออกแบบโปรแกรมแม่ข่าย (Smart Home Server)

โปรแกรมแม่ข่ายเป็นโปรแกรมมีหน้าที่รับผิดชอบให้บริการแก่ไคล์เอ็น(Client) ได้แก่ รีโมท แอพพลิเคชัน (Remote Application), การประมวลผล และควบคุมสั่งงานอุปกรณ์เครื่องใช้ไฟฟ้าต่างๆ



รูปที่ 3-5 แสดงสถาปัตยกรรมทั่วไปของโปรแกรมแม่ข่าย (The general architecture of smart home server)

โปรแกรมแม่ข่ายประกอบด้วย 2 ส่วนหลัก

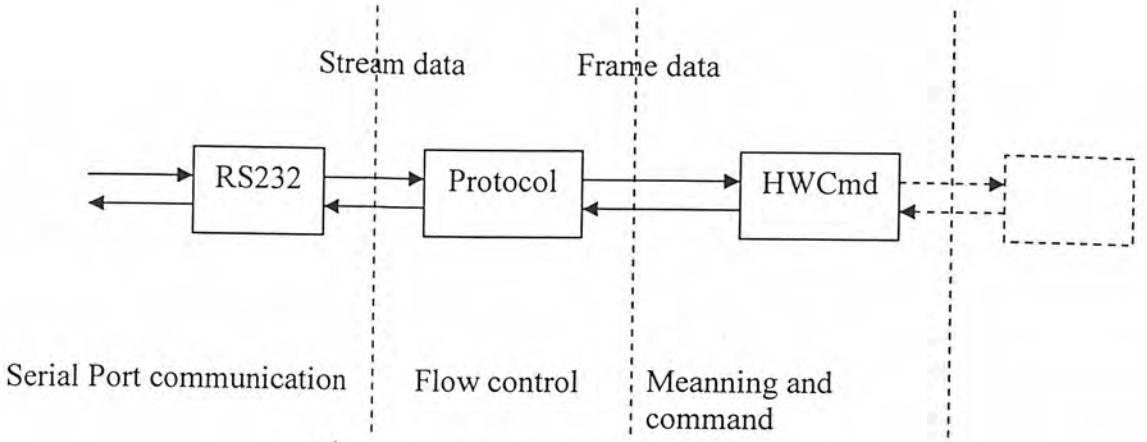
1. เว็บเซิร์ฟเวอร์ (Web Server) บริการเว็บเซอร์วิส (Web Service) แก่ไคลเอ็นผ่านทางเอสทีทีพีโปรโตคอล (HTTP) และให้บริการเป็นเว็บแอพพลิเคชันเพื่อผู้ใช้สามารถเข้ามาควบคุมระบบ ในมุมมองในฝั่งของเว็บเซิร์ฟเวอร์คือนำบริการต่างๆ ให้ผู้ใช้หรือแอพพลิเคชันอื่นๆสามารถเข้ามาใช้งานได้
2. โฮมเซิร์ฟเวอร์ (Home Server) เป็นโปรแกรมที่ประมวลผลข้อมูล สั่งการควบคุม และดูแลความถูกต้องของระบบทั้งหมด การทำงานต่าง ๆ จะถูกเรียกใช้ที่นี่ ทั้งการติดต่อสื่อสารกับอุปกรณ์เครื่องใช้ไฟฟ้า, การติดต่อโทรศัพท์มือถือส่งเอสเอ็มเอส (SMS), ดาต้าเบส (Database) ลักษณะการสื่อสารข้อมูลกับส่วนนี้ใช้การทำรีโมทโพรซีเจอร์คอล (Remote Procedure call)

ภาษาที่ใช้ในการพัฒนาโปรแกรม คือ ภาษา C# และ ASP.NET เนื่องจากสามารถเข้ากันได้กับระบบปฏิบัติการวินโดว์ (Windows) ที่ใช้พัฒนา ทั้งนี้ภาษา C# ยังสามารถดัดแปลงเพื่อไปทำงานบนลินุกซ์ (Linux) ได้อีกด้วย ผ่านเฟรมเวิร์ค (Framework) ที่ชื่อว่าโมโน (Mono) ซึ่งปัจจุบันกำลังพัฒนาอยู่ให้ครอบคลุมและมีความสามารถสมบูรณ์ทัดเทียมกับดอทเน็ตเฟรมเวิร์ค (.NET Framework) รายละเอียดสามารถดูได้ที่ http://www.mono-project.com/

3.2.1 กระบวนการเกี่ยวกับเฟรมข้อมูล

โฮมเซิร์ฟเวอร์ติดต่อกับบอร์ดควบคุมผ่านทางพอร์ตขนาน (Serial Port) โดยมีโมดูลที่รับผิดชอบคือ RS232 ซึ่งมองข้อมูลที่เข้ามาเป็นสายของข้อมูลอักขระ เมื่อมีข้อมูลเข้ามาจากพอร์ตขนาน ข้อมูลจะถูกทยอยส่งมาที่โมดูล Protocol ซึ่งรับผิดชอบเกี่ยวกับการควบคุมการสื่อสารให้ถูกต้องและเป็นไปตามโปรโตคอล ในโมดูลชั้นนี้จะมีส่วนในการตัดข้อมูลให้เป็นเฟรม การพิจารณาเกี่ยวกับลำดับหมายเลขการส่ง (Sequence number) และหมายเลขรับรองเฟรมข้อมูล(Acknowledgement) ถ้าข้อมูลเป็นไปตามโปรโตคอลที่วางไว้จะถูกส่งต่อไปยัง HwCmd เพื่อตีความและนำไปทำงานต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



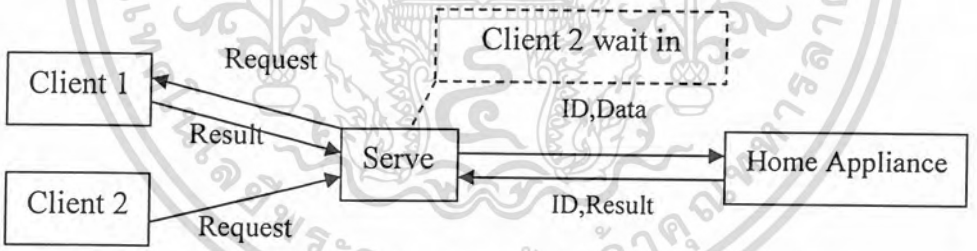
รูปที่ 3-6 แสดงกระบวนการการตัดเฟรมข้อมูล

กระบวนการตัดพารามิเตอร์

ใช้ Regular Expression ในการตัดคำเพื่อแยกพารามิเตอร์ที่ต้องการออกมา โดยในดอทเน็ตเฟรมเวิร์ค (.Net Framework) จะมี Class ที่ทำหน้าที่เกี่ยวกับด้าน Regular Expression สนับสนุนอยู่

กระบวนการเลือกส่งผลลัพธ์ให้กับผู้ใช้งาน

กรณีที่มีผู้ใช้ติดต่อเข้ามาใช้งานพร้อมกันหลายๆคน เวลาส่งผลลัพธ์คืนจะมีหมายเลข(ID)ที่ช่วยให้แยกแยะว่าผลลัพธ์ที่ส่งคืนนั้นเป็นของผู้ใด ซึ่งจะตรงกับหมายเลขของผลลัพธ์ที่จะส่งกลับ โดยในทุกครั้งที่มีการเชื่อมต่อเข้ามาจะมีการสร้างหมายเลขที่ไม่ซ้ำให้กับการเชื่อมต่อนั้น



รูปที่ 3-7 แสดงกระบวนการแยกแยะผลลัพธ์แก่ผู้ใช้งาน

จากรูปที่ 3-7 Client 1 ติดต่อเข้ามาขอบริการ Server จะสร้างหมายเลขให้กับการเชื่อมต่อนี้ และนำหมายเลขแปะไว้ในเฟรมข้อมูล เมื่ออุปกรณ์ปลายทางได้รับจะตอบผลลัพธ์กลับมาพร้อมด้วยหมายเลขประจำการเชื่อมต่อนี้ ทำให้ Server สามารถส่งผลลัพธ์ต่อไปยัง Client ได้ถูกตัว

การโปรแกรมโปรโตคอล Stop and wait protocol with ARQ และการควบคุมระดับโปรโตคอล

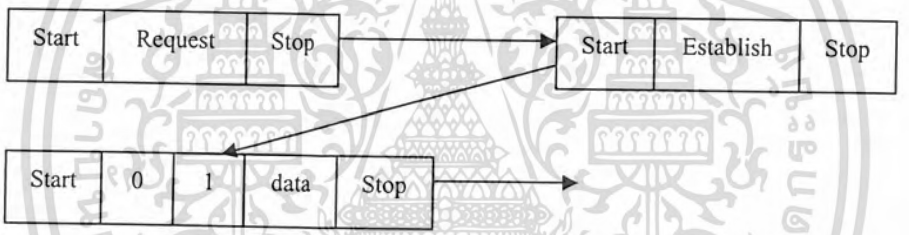
การเชื่อมต่อกับบอร์ดควบคุม การสื่อสารได้ออกแบบให้มีการตรวจสอบความถูกต้องในระดับหนึ่ง เช่น การส่งข้อมูลซ้ำ (Retransmission)เมื่อคาดว่าอีกฝ่ายไม่ได้รับ การสิ้นสุดการเชื่อมต่อเมื่อเกินเวลาที่กำหนด (Timeout) การสถาปนาการเชื่อมต่อ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
private void protocol(psk0_event event)
    {
        switch(event)
        {
          /* a packet has arrived from the network layer */
          case psk0_event.PacketReady :
             timer_wait_piggyback.Stop();
             strTxframe = getFrame();
             x.info = strTxframe;                    /* place packet in frame */
             x.seq = NextFrameToSend;
             x.ack = 1-FrameExpected;
             send_frame(x);                          /* send it to physical layer */
             timer_retransmission.Start();          // start timer retransmission
             timer_terminate.Start();               // start timer terminate connection
             break;
          /* a frame has arrived from the physical layer */
          case psk0_event.FrameArrival :
                y = getFrameFromPhysicalLayer();
                if (y.seq==FrameExpected)           // frame is wanted
                {
                   FrameExpected = 1-FrameExpected;
                   timer_wait_piggyback.Start();  // wait to ack without data
                   send_to_networkLayer(y.info);
                }
                if(y.ack==NextFrameToSend)          // acknowledgment has arrived
                {
                   timer_terminate.Stop();
                   timer_retransmission.Stop();
                   NextFrameToSend = 1-NextFrameToSend;
                }
          }
          break;
          /* a frame has not been ACKed in time,then retransmss */
          case psk0_event.TimeOut :
             timer_retransmission.Stop();
             x.info = strTxframe;
             x.seq = NextFrameToSend,
             x.ack = 1-FrameExpected,
             send_frame(x);
             timer_retransmission.Start();
             break;
          /* no frame in buffer, then ack with no data */
          case psk0_event.ChannelIdle:
             timer_wait_piggyback.Stop();
             x.info = "";
             x.seq = NextFrameToSend;
             x.ack = 1-FrameExpected;
             send_frame(x);
             NextFrameToSend = 1-NextFrameToSend;
             break;

          case psk0_event.Terminate:
             timer_terminate.Stop();
             timer_retransmission.Stop();
             send_error_toNetworkLayer();
             break;
        }
```

รูปที่ 3-8 แสดงตัวอย่างโปรแกรมการทำงานโปรโตคอล Stop and wait with ARQ



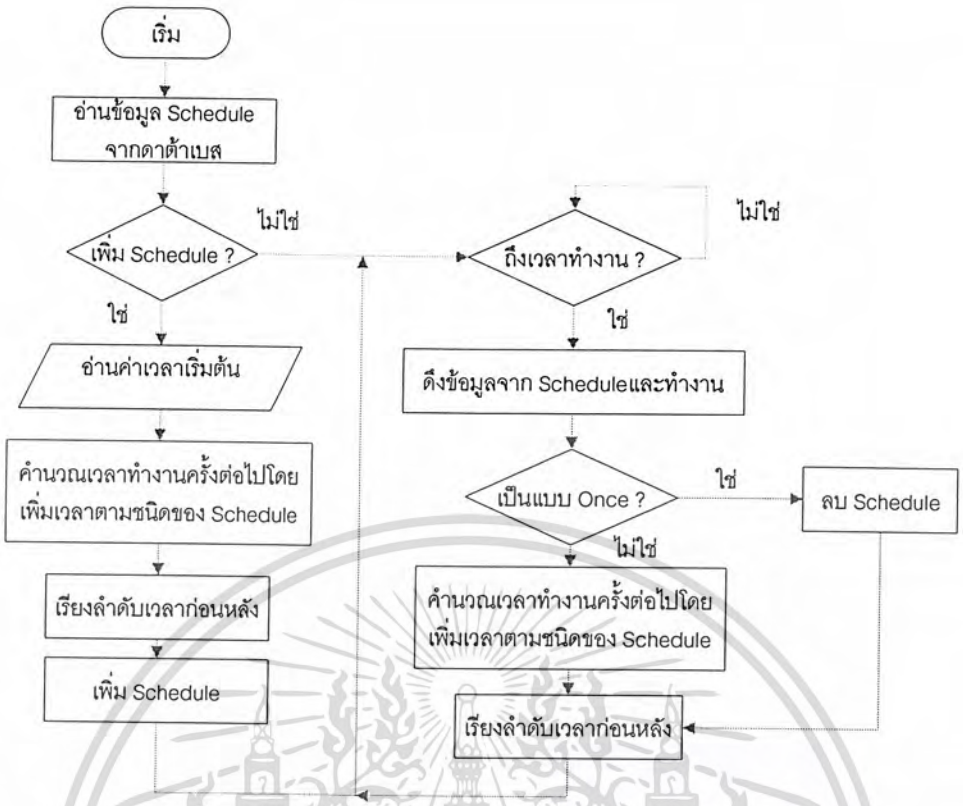
รูปที่ 3-9 แสดงการสถาปนาการเชื่อมต่อ

3.2.2 ระบบตารางเวลาทำงาน (Scheduler)

หลักการจัดการเกี่ยวกับตารางเวลาคือการทำให้งานที่กำหนดไว้ถูกกระทำในเวลาที่ถูกต้องตามที่กำหนด ทั้งนี้จะต้องสามารถจัดเก็บข้อมูล และเรียกมาใช้ใหม่ได้ตามต้องการ

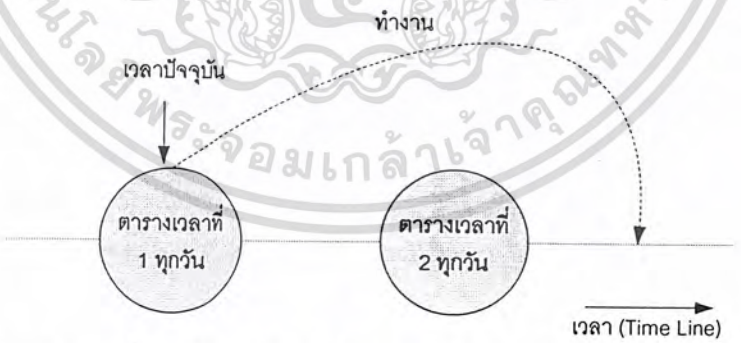
กระบวนการทำงานมีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3-10 แผนผังแสดงการคำนวณเวลาทำงานของตารางเวลา

หลังจากที่ตารางเวลาทำงานไปแล้ว ตารางเวลาที่ทำงานครั้งเดียวจะถูกลบทิ้งไป ถ้าไม่ใช่จะคำนวณเวลาที่จะทำถัดไป และลำดับใหม่ตามเวลาก่อนหลัง ในภาพด้านล่างคือตัวอย่างตารางเวลาที่เป็นประเภททำทุกๆวัน 2 อัน ซึ่งตารางเวลาที่ 1 ทำงานก่อน และหลังจากทำงานได้ถูกลำดับให้ไปอยู่ในอันดับ 2 แทน ซึ่งการลำดับเช่นนี้จะเหมือนกับการต่อเข้าคิว

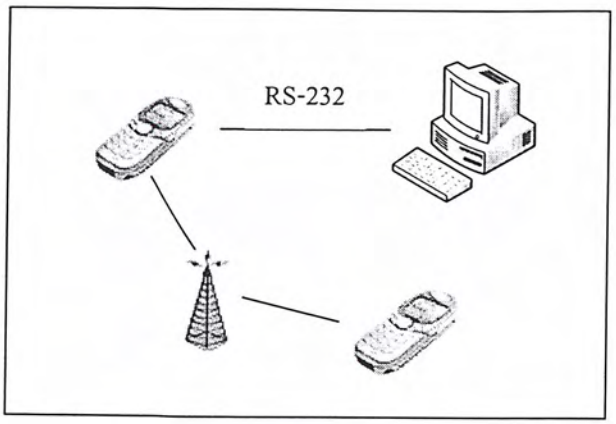


รูปที่ 3-11 การเปลี่ยนแปลงลำดับการทำงานของตารางเวลาแบบใช้คิว (Queue)

3.2.3 การแปลงข้อมูลเพื่อส่ง SMS ผ่านโมเด็มบนโทรศัพท์มือถือ

การส่ง SMS โดยการติดต่อกับโมเด็มของโทรศัพท์มือถือโดยตรงนั้น จะต้องแปลงข้อมูลให้เหมาะสมสำหรับใช้ในการส่ง สำหรับโทรศัพท์มือถือที่ใช้สำหรับระบบนี้คือ SIEMENS M55 โดยข้อมูลจะต้องถูกแปลงอยู่ในรูปแบบที่เรียกว่าพีดียู (PDU) ก่อนที่จะทำการติดต่อกับโทรศัพท์มือถือ

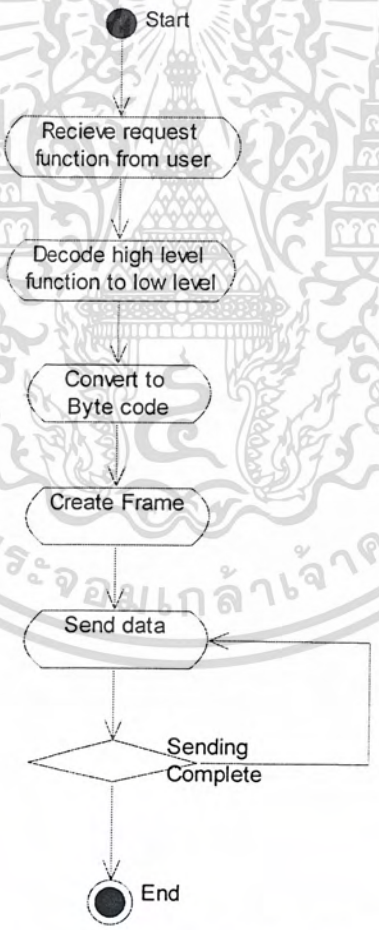
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3-12 แสดงการใช้โทรศัพท์มือถือส่ง SMS

3.2.4 กระบวนการส่งข้อมูลควบคุมเครื่องใช้ไฟฟ้า

จะใช้ในการสั่งเปิด – ปิดอุปกรณ์ไฟฟ้า หรือปรับระดับความสว่าง ทั้งแบบตั้งเวลาหรือสั่งงานเอง



รูปที่ 3-13 แสดงกระบวนสร้างคำสั่งเพื่อส่งไปยังอุปกรณ์

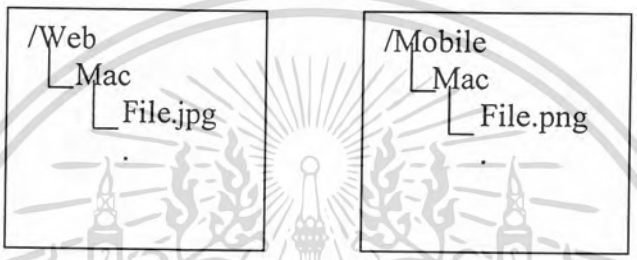
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.5 กระบวนการจับภาพจากกล้อง (Camera Capture)

เรียกดูภาพและควบคุมกล้อง โดยต้องแสดงตำแหน่ง เวลา แบบภาพนิ่ง ภาพที่จะเรียกดูได้จะถูกบันทึกเป็น Format JPEG สำหรับเรียกดูผ่านเว็บ และเป็น Format PNG สำหรับเรียกดูผ่านมือถือ เนื่องจากเป็น PNG เป็นรูปแบบพื้นฐานที่มือถือทั่วไปสามารถใช้งานได้

การติดต่อกับกล้องจะใช้โมดูลสำเร็จรูปที่ใช้ติดต่อกับไดร์ฟเวอร์ของกล้อง ชื่อ VideoOcx ซึ่งเป็นโมดูลที่ใช้งานง่าย และสนับสนุนกล้องได้หลากหลายชนิด รวมถึงมีไลบารี่สนับสนุนอื่นๆ ให้เรียกใช้งานมากมาย

กระบวนการจับภาพนั้นใช้การติดต่อผ่าน VideoOcx เพื่อจับภาพและทำการใส่วันเวลาสถานที่ลงในภาพ จากนั้นนำไปเก็บในโฟล์เดอร์ซึ่งมีโครงสร้างดังนี้



รูปที่ 3-14 แสดงโครงสร้างการจัดเก็บไฟล์ของภาพถ่ายจากกล้อง

รูปแบบชื่อไฟล์
dd-mm-yy_hh-mm-ss.jpg

ตัวอย่างชื่อไฟล์
03-03-48_21-01-57.jpg

ความหมายของรหัสชื่อไฟล์
dd คือ วันที่
mm คือ เดือน
yy คือ ปี
hh คือ ชั่วโมง
mm คือ นาที
ss คือ วินาที

3.2.6 กระบวนการกับเหตุการณ์และการตอบสนอง (Event and Macro)

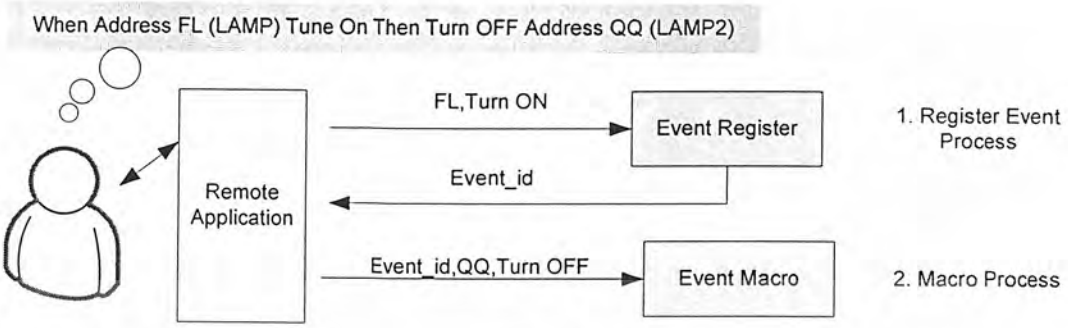
เป็นฟังก์ชันเพื่อให้ผู้ใช้สามารถกำหนดการตอบสนองต่อเหตุการณ์ได้ เป็นชุดของคำสั่งได้ ซึ่งสามารถนำมาประยุกต์ใช้ได้มากมาย เช่น การตอบสนองต่อเหตุร้าย ด้วยการแจ้ง SMS สั่งงานอุปกรณ์อื่นๆ เช่น เปิดกระดิ่งเตือนภัย ฯลฯ

กระบวนการก่อนที่จะสามารถตรวจสอบเหตุการณ์ได้มี 2 ขั้นตอน ดังนี้

1. การลงทะเบียนเหตุการณ์ ในขั้นตอนนี้จะเป็นการโหลดเหตุการณ์ที่ต้องการตรวจสอบเข้าไปในฐานข้อมูล และคอยตรวจสอบความเปลี่ยนแปลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. การกำหนดการตอบสนอง ในขั้นตอนนี้เป็นการโหลดคำสั่งที่ต้องการให้ทำงานหลังจากการเกิดเหตุการณ์ลงในฐานข้อมูล



รูปที่ 3-15 แสดงกระบวนการก่อนที่จะตรวจสอบเหตุการณ์

การทำงานคือนำเอาเหตุการณ์ที่เกิดขึ้นจากอุปกรณ์มาเปรียบเทียบกับข้อมูลในดาต้าเบส ถ้าพบก็จะอ่านค่า Event_id ของเหตุการณ์นั้นๆ เพื่อไปค้นในดาต้าเบส แล้วนำเอาคำสั่งที่เก็บไว้มาทำงาน

3.2.7 การออกแบบฐานข้อมูล

ฐานข้อมูลที่ใช้จัดเก็บข้อมูลในระบบรักษาความปลอดภัยและอำนวยความสะดวกผ่านโทรศัพท์มือถือ ใช้ระบบฐานข้อมูลของไมโครซอฟแอกเซส (Microsoft Access) โดยหลักจากการ Normalize แล้วได้ตารางดังนี้

1. ตาราง HDev (Hardware Device) เก็บข้อมูลอุปกรณ์ควบคุม ซึ่งในหนึ่งอุปกรณ์ควบคุมนั้นอาจประกอบด้วยแชลแนลมากกว่าหนึ่งแชลแนล

	ชื่อเขตข้อมูล	ชนิดข้อมูล	
🔑	mac	Text	
	channels	Number	channel ทั้งหมด
	model	Text	รุ่น ซึ่งจะบอกประเภท
	pos	Text	ที่ตั้ง
	status	Number	ติดต่อได้หรือไม่

ตารางที่ 3-2 แสดงตาราง HDev

2. ตาราง HChannel (Hardware by Channel) เก็บข้อมูลอุปกรณ์ควบคุมแยกกันในแต่ละ channel

	ชื่อเขตข้อมูล	ชนิดข้อมูล	
🔑	mac	Text	mac address
🔑	channel	Number	ช่อง output
	address	Text	
	name	Text	ชื่ออุปกรณ์
	regis_no	AutoNumber	ลำดับการลงทะเบียน
▶			

ตารางที่ 3-3 แสดงตาราง HChannel

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ตาราง iCameraDev เก็บข้อมูลเกี่ยวกับกล้องจับภาพ การเชื่อมต่อ โหมดต่างๆ

	ชื่อเขตข้อมูล	ชนิดข้อมูล	
🔑	port	Number	port ที่ใช้ติดต่อ
	name	Text	
	details	Text	รายละเอียดกล้อง
	cap_time	Number	ช่วงเวลาการถ่าย
	enable	Number	เปิด-ปิด option cap หรือไม่

ตารางที่ 3-4 แสดงตาราง iCameraDev

4. ตาราง iDimDev เก็บข้อมูลเกี่ยวกับอุปกรณ์ปรับระดับความสว่าง สถานะ

	ชื่อเขตข้อมูล	ชนิดข้อมูล	
🔑	addr	Text	
	dim_level	Number	สถานะ

ตารางที่ 3-5 แสดงตาราง iDimDev

5. ตาราง iSwDev เก็บข้อมูลเกี่ยวกับอุปกรณ์ควบคุมเปิดปิด สถานะ

	ชื่อเขตข้อมูล	ชนิดข้อมูล	
🔑	addr	Text	
	power	Number	สถานะ

ตารางที่ 3-6 แสดงตาราง iSwDev

6. ตาราง iSecurityDev เก็บข้อมูลเกี่ยวกับอุปกรณ์รักษาความปลอดภัย

	ชื่อเขตข้อมูล	ชนิดข้อมูล	
🔑	addr	Text	
	security_alert	Number	สถานะความผิดปรกติ
	power	Number	สถานะเปิดปิด

ตารางที่ 3-7 แสดงตาราง iSecurityDev

7. ตาราง iSchedule เก็บข้อมูลเกี่ยวกับตารางเวลา ชื่อ เวลาที่ทำงาน คำสั่ง พารามิเตอร์ต่างๆ

	ชื่อเขตข้อมูล	ชนิดข้อมูล	
	id	AutoNumber	รหัส
🔑	name	Text	ชื่อ
	type	Number	ประเภท
	next_time	Date/Time	เวลาเริ่ม
	days	Text	วันที่ให้ทำงาน (0000000) 0=ไม่ทำ, 1=ทำ
	addr	Text	หมายเลขอุปกรณ์ที่ให้ทำงาน
	cmd	Text	รหัสคำสั่ง
	param	Text	พารามิเตอร์

ตารางที่ 3-8 แสดงตาราง iSchedule

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. ตาราง iEvent เก็บข้อมูลเกี่ยวกับรายการคำสั่งของแต่ละ Event

	ชื่อเขตข้อมูล	ชนิดข้อมูล	
🔑	event_no	AutoNumber	
	event_id	Number	Event id
	address	Text	ปลายทาง
	name	Text	ชื่อ event
	cmd	Number	คำสั่ง
	param	Text	พารามิเตอร์

ตารางที่ 3-9 แสดงตาราง iEvent

9. ตาราง iEventMonitor เก็บข้อมูลเกี่ยวกับ Event ที่ต้องทำการตรวจจับ

	ชื่อเขตข้อมูล	ชนิดข้อมูล	
🔑	address	Text	ที่อยู่อุปกรณ์
🔑	event_type	Number	ประเภทเหตุการณ์
	event_id	AutoNumber	

ตารางที่ 3-10 แสดงตาราง iEventMonitor

3.3 การออกแบบโปรแกรมลูกข่ายบนโทรศัพท์มือถือ

3.3.1 เทคนิคหรือเทคโนโลยีที่ใช้

- ภาษาจาวา โดยเลือกใช้ J2ME (Java 2 Platforms Micro Edition) ในการพัฒนา
- KSOAP2 เป็น Library ที่ช่วยในการพัฒนาในส่วนของการรับส่งข้อมูลระหว่างเว็บเซอร์วิสกับโทรศัพท์มือถือโดยอาศัย SOAP
- โปรแกรม GNUBox เป็นโปรแกรมที่ติดตั้งลงบนโทรศัพท์มือถือ Nokia 6600 โดยจะมีผลทำให้โทรศัพท์สามารถติดต่อกับอุปกรณ์รับส่งสัญญาณบลูทูธได้ และสามารถใช้งานอินเตอร์เน็ตได้ ทั้งนี้ต้องมีการตั้งค่าการทำงานบนเครื่องคอมพิวเตอร์ที่ต่ออยู่กับอุปกรณ์บลูทูธสียก่อน โดยจะใช้ได้กับเครื่องโทรศัพท์ Nokia 6600 และ Nokia 7610

3.3.2 เครื่องมือที่ใช้ในการพัฒนา

Software

- J2ME Wireless Toolkit 2.2 ใช้เป็น emulator ในการทดสอบโปรแกรมที่เขียน
- Nokia Series 60 MIDP Concept SDK Beta 0.3.1 เป็น emulator ของ Nokia Series 60 โดยทำงานร่วมกับ WTK 2.2
- JCreator Pro 3.0 ใช้ในการเขียน J2ME

Hardware

- โทรศัพท์มือถือ ใช้เป็นอุปกรณ์เพื่อใช้ในการทดสอบการทำงานของแอพพลิเคชั่นจริง โดยต้องสนับสนุน Java MIDP 2.0, SMS, MMS, JSR 82 (Java Bluetooth API) และ การเชื่อมต่อ บลูทูธและจีพีอาร์เอส โดยในโครงงานนี้ได้เลือกใช้โทรศัพท์มือถือ Nokia 6600 เนื่องจากสามารถใช้โปรแกรม GNUBox ได้ซึ่งช่วยในการเชื่อมต่อบลูทูธ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- อุปกรณ์รับส่งสัญญาณบลูทูธ(Bluetooth Dongle)

3.3.3 โครงสร้างและการทำงานของโปรแกรมลูกข่ายบนโทรศัพท์มือถือ

สำหรับการทำงานของแอพพลิเคชั่นนี้จะทำงานบนโทรศัพท์มือถือ ซึ่งในโครงงานนี้ได้พัฒนาด้วยภาษา J2ME เพราะเป็นภาษาที่ใช้กันอย่างกว้างขวาง โดยจะใช้การเชื่อมต่อผ่านทางจีพีอาร์เอสสำหรับการใช้งานภายนอกบ้านหรือบลูทูธสำหรับการใช้งานภายในบ้านเพื่อติดต่อไปยังเว็บเซอร์วิส เพื่อที่จะทำการตรวจดูและสั่งการทำงานในส่วนต่างๆ โดยจะแบ่งการทำงานได้เป็น 3 ส่วนหลักๆ คือ

ส่วนติดต่อเว็บเซอร์วิส

เป็นส่วนที่ใช้เชื่อมต่อกับไปยังเว็บเซอร์วิสโดยจะทำการรับส่งข้อมูลผ่านทาง SOAP

ส่วนติดต่อผู้ใช้งาน(User Interface)

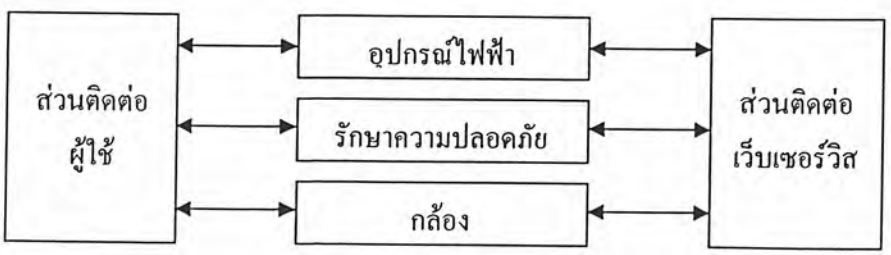
เป็นส่วนติดต่อผู้ใช้งานซึ่งจะใช้ High Level UI ของ J2ME เพื่อสร้างอินเตอร์เฟสให้ผู้ใช้ใช้งาน

ส่วนฟังก์ชันที่เรียกใช้งานของโปรแกรม

เป็นส่วนที่ดูแลในเรื่องฟังก์ชันต่างๆ ในการสั่งงานไปยังเว็บเซอร์วิส โดยจะแบ่งได้เป็น

1. อุปกรณ์ไฟฟ้าภายในบ้าน (Home Appliance) จะทำการสั่งงานเปิด-ปิด และดูสถานะของเครื่องใช้ไฟฟ้าภายในบ้าน
2. รักษาความปลอดภัย (Security) ทำการตรวจดูสถานะของเซ็นเซอร์ตรวจจับต่างๆ และตั้งค่าเบอร์โทรศัพท์ที่จะให้ส่ง SMS มาแจ้งเตือน
3. กล้อง (Camera) จะทำการเลือกดูภาพนิ่งจากกล้อง(View Picture) และตั้งค่าการบันทึกภาพว่าจะให้บันทึกทุกๆ กี่นาที
4. การตั้งเวลาในการทำงาน (Schedule) เพื่อกำหนดเวลาในการทำงานของอุปกรณ์ไฟฟ้าต่างๆ และเซ็นเซอร์ โดยสามารถกำหนดการทำงานได้หลายแบบ ดังนี้
   - กำหนดการทำงานที่ทำเพียงครั้งเดียว (Once)
   - กำหนดการทำงานรายวัน (Daily)
   - กำหนดการทำงานรายสัปดาห์ (Weekly)
   - กำหนดการทำงานรายเดือน (Monthly)

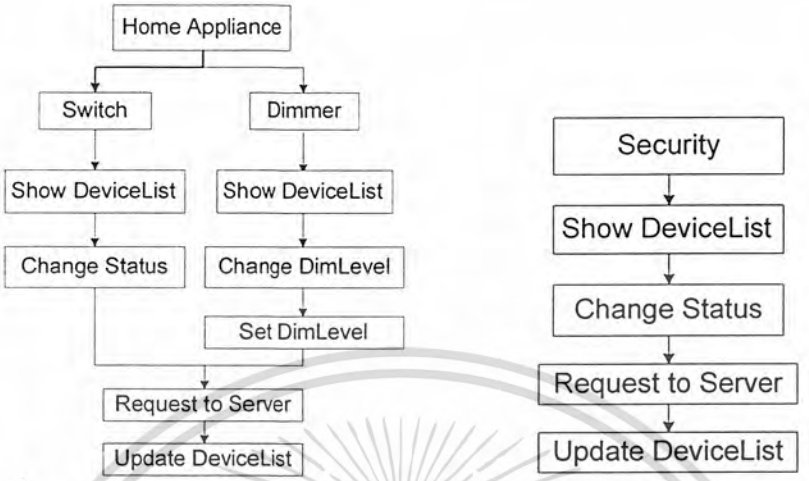
   โดยทำการเลือกวันที่และเวลาที่จะให้เริ่มทำงาน



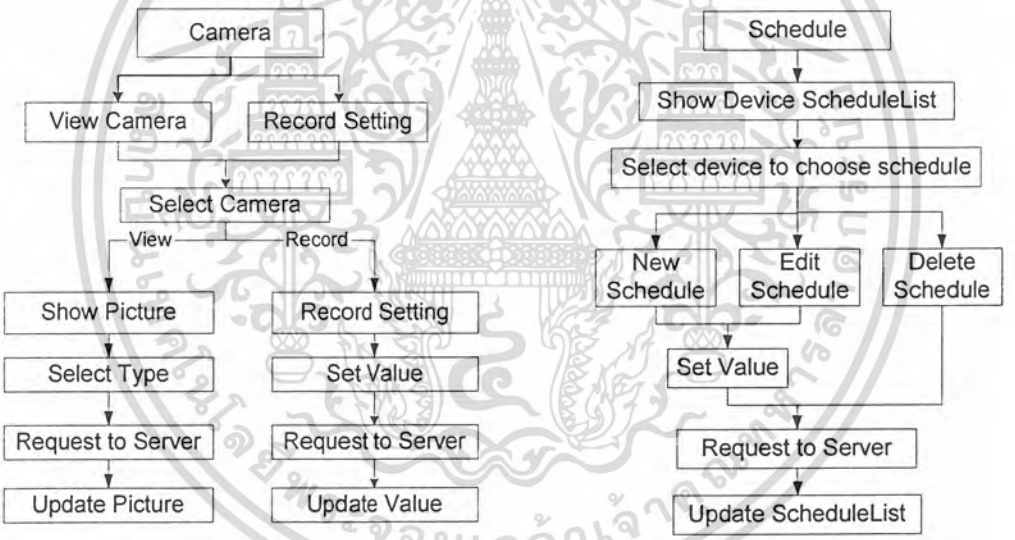
รูปที่ 3-16 แสดงการทำงานของโปรแกรมในแต่ละส่วน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากโทรศัพท์มือถือมีข้อจำกัดในเรื่องของขนาดหน้าจอในการแสดงผล จึงได้ออกแบบให้มีฟังก์ชันการทำงานครอบคลุมเท่าที่จะเป็นไปได้ โดยจะเน้นให้ใช้งานง่ายและมีปริมาณการรับส่งข้อมูลไม่มากนัก



รูปที่ 3-17 แสดงการทำงานในส่วนของอุปกรณ์ไฟฟ้าและการรักษาความปลอดภัย

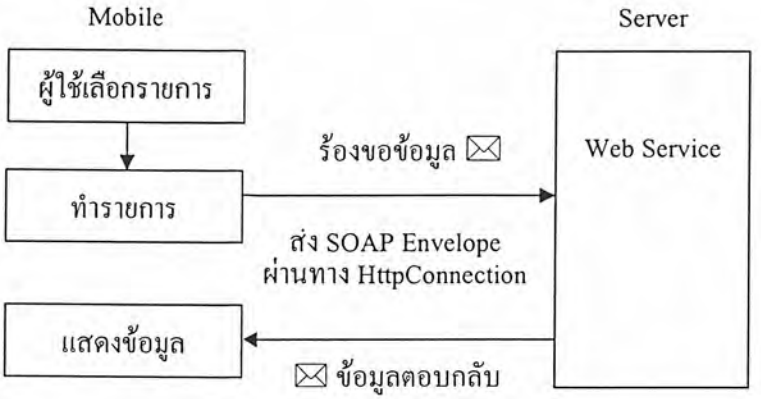


รูปที่ 3-18 แสดงการทำงานในส่วนของกล้องและการตั้งเวลาในการทำงาน

3.3.4 การติดต่อสื่อสารระหว่างโทรศัพท์มือถือกับเว็บเซอร์วิส

สำหรับส่วนติดต่อกับเว็บเซอร์วิสนั้นจะใช้การติดต่อด้วยโปรโตคอล SOAP ซึ่งจะใช้งานผ่านทาง HttpConnection โดยในโครงงาน นี้ได้เลือกใช้ KSOAP2 Package ซึ่งเป็น Library ที่ช่วยในการติดต่อด้วย SOAP ที่มีขนาดเล็กและใช้งานง่ายในการรับส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3-19 แสดงการรับ – ส่งข้อมูลของแอพพลิเคชั่น

กระบวนการในการรับ-ส่งข้อมูล จะเริ่มจากเมื่อมีการเลือกรายการใดๆ ขึ้นมา จากนั้นจะเริ่มทำรายการโดยการส่งข้อมูลออกไปผ่านทาง SOAP Envelope เพื่อทำการร้องขอข้อมูลที่ต้องการจากเว็บเซอร์วิส แล้วรอการตอบค่าที่ต้องการกลับมาแล้วจึงนำไปแสดงผลต่อไป

ในส่วนของการเชื่อมต่อนั้นจะใช้การเชื่อมต่อผ่านจีพีอาร์เอสหรือบลูทูธ โดยถ้าใช้จีพีอาร์เอสจะสามารถเชื่อมต่อได้โดยตรงจากโทรศัพท์มือถืออยู่แล้ว แต่ถ้าใช้บลูทูธนั้นจะต้องทำการลงโปรแกรม GNUBox ที่โทรศัพท์มือถือ เพื่อทำให้สามารถใช้งานผ่านบลูทูธได้

3.3.5 รูปแบบการในการติดต่อสื่อสารระหว่างเว็บเซอร์วิสและโทรศัพท์มือถือ

การส่งข้อมูลไปยังเว็บเซอร์วิส

จะใช้ KSOAP2 Library ช่วยในการส่งข้อมูลกับเว็บเซอร์วิส โดยผ่านทาง SOAP โดยเริ่มจากการสร้าง SoapObject ขึ้นมาโดยกำหนดค่า SOAP Namespace และเมดทรอดที่จะเรียกใช้ให้มัน จากนั้นทำการเรียกฟังก์ชัน addProperty เพื่อทำการใส่ค่าแอตทริบิวต์ต่างๆ ตามที่เมดทรอดนั้นต้องการ แล้วจึงสร้าง SoapSerializationEnvelope เพื่อทำหน้าที่เป็นซองจดหมายที่จะส่งออกไป โดยจะกำหนดค่า bodyOut หรือเนื้อความที่จะส่งไปให้เป็น SoapObject จากนั้นจึงทำการส่งซองจดหมายออกไปร้องขอข้อมูลกับเว็บเซอร์วิส โดยการสร้าง HttpTransport ไปยัง SOAP URL ที่ต้องการ แล้วจึงเรียกฟังก์ชัน call ไปยัง SOAP Action ที่ต้องการต่อไป

ตัวอย่างการส่งข้อมูลให้เว็บเซอร์วิส

```
SoapObject rpc = new SoapObject("urn:delayed-quotes","getQuote");
rpc.addProperty("symbol","IBM");
SoapSerializationEnvelope envelope =
      new SoapSerializationEnvelope(SoapEnvelope.VER10);
envelope.bodyOut = rpc;
HttpTransport ht =
      new HttpTransport("http://services.xmethods.net/soap");
ht.call("urn:xmethods-delayed-quotes#getQuote", envelope);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การรับข้อมูลจากเว็บเซอร์วิส

เมื่อได้ส่งซองจดหมายออกไปแล้ว โทรศัพท์มือถือจะรอรับซองจดหมายตอบกลับจากเว็บเซอร์วิส ซึ่งเนื้อความที่ได้จะอยู่ในรูปของ XML จากนั้นจึงนำเนื้อความที่เป็น XML นั้นมาทำการอ่านค่าเพื่อเก็บลงในตัวแปรชนิด Vector อีกครั้ง แล้วจึงนำออกไปใช้งานเพื่อแสดงผลต่อไป

ตัวอย่างการรับข้อมูลจากเว็บเซอร์วิส

```
Object resultOb = envelope.getResult();
```

ข้อมูลที่ได้จะเป็น XML แบบด้านล่าง

```
<Device_List>
  <Device>
    <model>SW</model>
    <name>door light</name>
    <address>E0</address>
    <status>0</status>
  </Device>
  <Device>
    <model>SW</model>
    <name>siren</name>
    <address>E1</address>
    <status>0</status>
  </Device>
</Device_List>
```

ตัวอย่างการอ่านค่าจาก XML ที่ได้ โดยจะทำการไล่อ่านเข้าไปจาก root node แล้วเก็บค่าลงไปไว้ใน result โดยใช้ฟังก์ชัน addElement()

```
Vector result;
InputStream is;
is = new ByteArrayInputStream(resultOb.toString().getBytes());
XmlPullParser parser = new KXmlParser();

parser.setFeature(XmlPullParser.FEATURE_PROCESS_NAMESPACES, true);
parser.setInput (is, null);

parser.nextTag();
parser.require(XmlPullParser.START_TAG, null, null);
while (parser.nextTag () != XmlPullParser.END_TAG){
      Hashtable hash = new Hashtable();
      parser.require(XmlPullParser.START_TAG, null, null);

      while (parser.nextTag() != XmlPullParser.END_TAG) {
            parser.require(XmlPullParser.START_TAG, null, null);
            String text = parser.nextText();
            String name = parser.getName();

            hash.put(name,text);
            parser.require(XmlPullParser.END_TAG, null, name);
      }
      result.addElement(hash);
      parser.require(XmlPullParser.END_TAG, null, null);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างการนำตัวแปรชนิด Vector ที่ได้ไปใช้งาน โดยใช้ getKey ดึงค่า ตามคีย์ที่ต้องการขึ้นมา จากนั้นจึงนำไปแสดงผลต่อไป

```
String status = getKey("status",0).toString();
//public Object getKey(String key,int obnum){
//    if (result.size()==0 || result.size()<obnum || !XML)
//          return null;
//    else{
//          Hashtable hash = (Hashtable) result.elementAt(obnum);
//          return hash.get(key);
//    }
//}
```

สำหรับการรับภาพเพื่อมาแสดงบนโทรศัพท์มือถือนั้นจะใช้การเปิดอินพุตสตรีม แล้วติดต่อไปยัง URL ของรูปภาพที่อยู่ที่เครื่องเซิร์ฟเวอร์ผ่านทาง HttpConnection แล้วรอรับข้อมูลรูปนั้นมา เพื่อทำการแสดงขึ้นบนหน้าจอโทรศัพท์มือถือต่อไป

3.4 การออกแบบโปรแกรมฝั่งข่าย บนเว็บบราวเซอร์

ในส่วนของเว็บบราวเซอร์จะเขียนด้วยภาษา ASP.NET ซึ่งจะมีฟังก์ชันในการทำงานคล้ายกับบนโทรศัพท์มือถือแต่จะสามารถดูรายละเอียดของข้อมูลได้ครบถ้วนมากกว่า และยังสามารถตั้งค่าต่างๆ ได้ โดยจะแบ่งการทำงานเป็น 3 ส่วนเช่นกัน คือ ส่วนติดต่อเว็บเซอร์วิส ส่วนติดต่อผู้ใช้งาน และส่วนฟังก์ชันที่เรียกใช้งานโปรแกรม ซึ่งในส่วนสุดท้ายนี้จะทำงานได้ละเอียดมากขึ้น โดยจะเพิ่มในเรื่องของ

- การเพิ่ม ลบ และแก้ไขผู้ใช้งาน
- การตั้งรหัสผ่าน
- การกำหนดชื่ออุปกรณ์ต่างๆ
- การกำหนดชื่อสถานที่
- การกำหนดค่าคอมพอร์ตที่ต่อกับชุดควบคุม

3.4.1 รูปแบบการในการติดต่อสื่อสารระหว่างเว็บเซอร์วิสและเว็บบราวเซอร์

การรับส่งข้อมูลกับเว็บเซอร์วิส

จะทำการเรียกใช้เมธทอดที่จะใช้งานได้เลย โดยส่งค่าไปตามเมธทอดที่เรียก จากนั้นจะมีการตอบกลับค่าที่ต้องการจากเว็บเซอร์วิสกลับมา เพื่อนำไปใช้ในการแสดงผลให้ผู้ใช้ดูต่อไป

ตัวอย่างคำสั่งที่ใช้เรียกในการรับส่งข้อมูลกับเว็บเซอร์วิส โดยทำการเรียกเมธทอดที่ชื่อว่า Exec_wListDevice ซึ่งต้องส่งค่า mode ไปด้วย จากนั้นจะได้รับค่าตอบกลับกลับมาจากเว็บเซอร์วิสในรูปของ XML เหมือนกับในโทรศัพท์มือถือ

```
web myService = new web();
string data = myService.Exec_wListDevice(mode);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างการรับข้อมูล XML มาเพื่อมานำมาเก็บไว้ใน DataSet เพื่อนำไปใช้ในการแสดงผลต่อไป

```
private DataSet XMLtoDataSet(int mode)
{
        web myService = new web();
        string data = myService.Exec_wListDevice(mode);
        MemoryStream stream = new MemoryStream();
        byte[] bt = Encoding.ASCII.GetBytes(data);
        stream.Write(bt,0,bt.Length);
        stream.Position = 0;
        DataSet ds = new System.Data.DataSet();
        ds.ReadXml(stream);
        return ds;
}
```

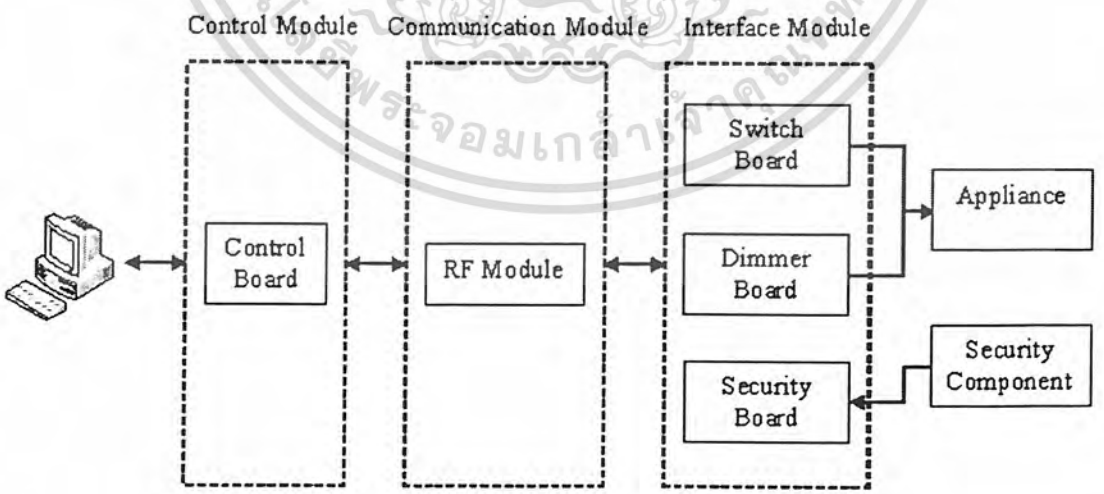
3.5 โครงสร้างการออกแบบในส่วนของฮาร์ดแวร์

3.5.1 การออกแบบและระบบสมาร์ทโฮม

จากความต้องการเบื้องต้นที่จะต้องการควบคุมเครื่องใช้ไฟฟ้าภายในบ้านให้มีขอบเขตที่กว้างขึ้น โดยไม่จำเป็นที่จะต้องพึ่งการเปิด/ปิดที่สวิตซ์ของอุปกรณ์นั้นเพียงอย่างเดียว ซึ่งควบคู่กับระบบรักษาความปลอดภัยภายในบ้านที่มีทั้งระบบตรวจจับควัน ตรวจจับผู้บุกรุก โดยที่สามารถตรวจสอบความปลอดภัยภายในบ้านได้ไม่ว่าจะอยู่ที่ไหน เพียงแค่มีโทรศัพท์มือถือเท่านั้น

ดังนั้นงานทางด้านฮาร์ดแวร์นี้จะประกอบไปด้วย 3 ส่วนหลักๆ คือ

1. ส่วนที่ทำหน้าที่เป็นตัวควบคุมการสั่งงานอุปการณ์ต่างๆ(Control Module)
2. ส่วนที่ใช้ในการติดต่อสื่อสาร(Communication Module)
3. ส่วนที่ใช้ในการควบคุมเครื่องใช้ไฟฟ้าหรืออุปกรณ์รักษาความปลอดภัย(Interface Module) โดยเป็นส่วนที่ใช้ติดต่อโดยตรงกับอุปกรณ์ที่ต้องการควบคุม ซึ่งจะมีการแยกประเภทการใช้งานให้เหมาะสมกับการใช้งานในแต่ละประเภท



รูปที่ 3-20 แสดงถึงภาพรวมของฮาร์ดแวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนควบคุม (Control Module)

เป็นส่วนกลางการควบคุม โดยจะทำการติดต่อสื่อสารข้อมูลผ่านทางพอร์ตอนุกรมของคอมพิวเตอร์ (RS-232) ซึ่งจะส่งข้อมูลแบบผลัดกันรับ-ผลัดกันส่ง เพื่อนำข้อมูลมาทำการประมวลผลคำสั่ง แล้วส่งคำสั่งไปให้กับอินเทอเฟซโมดูล ต่างๆ เพื่อให้ อินเทอเฟซโมดูลทำงาน

ส่วนสื่อสาร (Communication Module)

เป็นโมดูลที่ใช้ในการกำหนดรูปแบบในการสื่อสารว่าจะใช้การสื่อสารแบบใดในการติดต่อควบคุมอุปกรณ์เครื่องใช้ไฟฟ้าและอุปกรณ์รักษาความปลอดภัย ซึ่งส่วนสื่อสาร จะเป็นตัวกลางที่ทำหน้าที่ติดต่อสื่อสารระหว่าง ส่วนควบคุม กับ อินเทอเฟซโมดูล ต่างๆ ให้สามารถติดต่อสื่อสารกันได้

แต่เนื่องจาก ส่วนควบคุม และ อินเทอเฟซโมดูล ไม่ได้อยู่ใกล้กัน กล่าวคือ อินเทอเฟซโมดูล จะกระจายอยู่ตามจุดต่างๆในบ้าน แต่ ส่วนควบคุม จะอยู่ใกล้กับคอมพิวเตอร์ดังนั้น วิธีที่จะใช้ในการติดต่อสื่อสารที่เป็นไปได้มี2วิธีคือ

วิธีการแรกคือการรับ-ส่งข้อมูลผ่านทางคลื่นวิทยุหรือ RF (Radio Frequency) แต่เนื่องจากภายในบ้านมีกำแพงมากมาย จึงเกิดการลดทอนสัญญาณคลื่นวิทยุ ทำให้ระยะทางในการรับ-ส่งสั้นลง และการออกแบบวงจรรับ-ส่งนั้น ค่อนข้างยุ่งยากมาก

วิธีที่สองคือการรับ-ส่งข้อมูลผ่านสายไฟบ้านหรือ PLC (Power Line Communication) พบว่าต่างประเทศนิยมใช้กันมาก เพราะทุกๆบ้านจะต้องมีสายไฟบ้านเดินอยู่รอบบ้านอยู่แล้ว หากเรานำข้อมูลนั้นมาทำการมอดดูเลตลงไปบนสายไฟ จะทำให้เกิดการประหยัดในการหาสื่อที่ใช้ในการส่งข้อมูล แต่วิธีนี้ก็มีข้อเสีย คือ สายไฟบ้านนั้น ไม่ได้ถูกออกแบบมาสำหรับการส่งข้อมูล ดังนั้นย่อมเกิดการสูญเสียจากลักษณะคุณสมบัติของสาย และมีสัญญาณรบกวนที่มาจากอุปกรณ์อื่นๆค่อนข้างมาก จนทำให้บางครั้งไม่สามารถสื่อสารข้อมูลกันได้เลย และปัญหาที่สำคัญคือไฟบ้านแบ่งเป็น 3 เฟส จึงเป็นปัญหาที่ทำให้ไม่สามารถส่งข้อมูลข้ามเฟสกันได้

จากการศึกษาข้อมูลทั้ง2วิธี พบว่าปัญหาจากการรับ-ส่งข้อมูลผ่านสายไฟบ้าน มีปัญหาที่ยุ่งยากมากกว่าการรับ-ส่งข้อมูลผ่านทางคลื่นวิทยุ และเป็นปัญหาที่แก้ไขได้ยาก ดังนั้นส่วนสื่อสาร จึงใช้รูปแบบการรับ-ส่ง ข้อมูลผ่านทางคลื่นวิทยุโดยใช้อาร์เอฟโมดูล (RF Module)

อินเทอเฟซโมดูล (Interface Module)

อินเทอเฟซโมดูลนี้จะติดต่อโดยตรงกับอุปกรณ์ไฟฟ้าและอุปกรณ์รักษาความปลอดภัย ซึ่งจะมีการแยกประเภทของการใช้งานของอินเทอเฟซโมดูลไว้ให้เหมาะสมกับการใช้งานโดยในโครงงานนี้จะมีทั้งหมด 2 ประเภทหลักๆ คือ ใช้กับอุปกรณ์ไฟฟ้า และ ใช้กับอุปกรณ์รักษาความปลอดภัย

1. อินเทอเฟซโมดูลที่ใช้กับอุปกรณ์ไฟฟ้า มีดังนี้
   - บอร์ดสวิตซ์ (Switch Board)ทำหน้าที่ เปิด-ปิด อุปกรณ์ไฟฟ้าด้วยรีเลย์
   - บอร์ดปรับระดับความสว่าง (Dimmer Board) ทำหน้าที่ควบคุมความสว่างของหลอดไฟด้วยไตรแอก ซึ่งสามารถใช้กับหลอดไส้เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

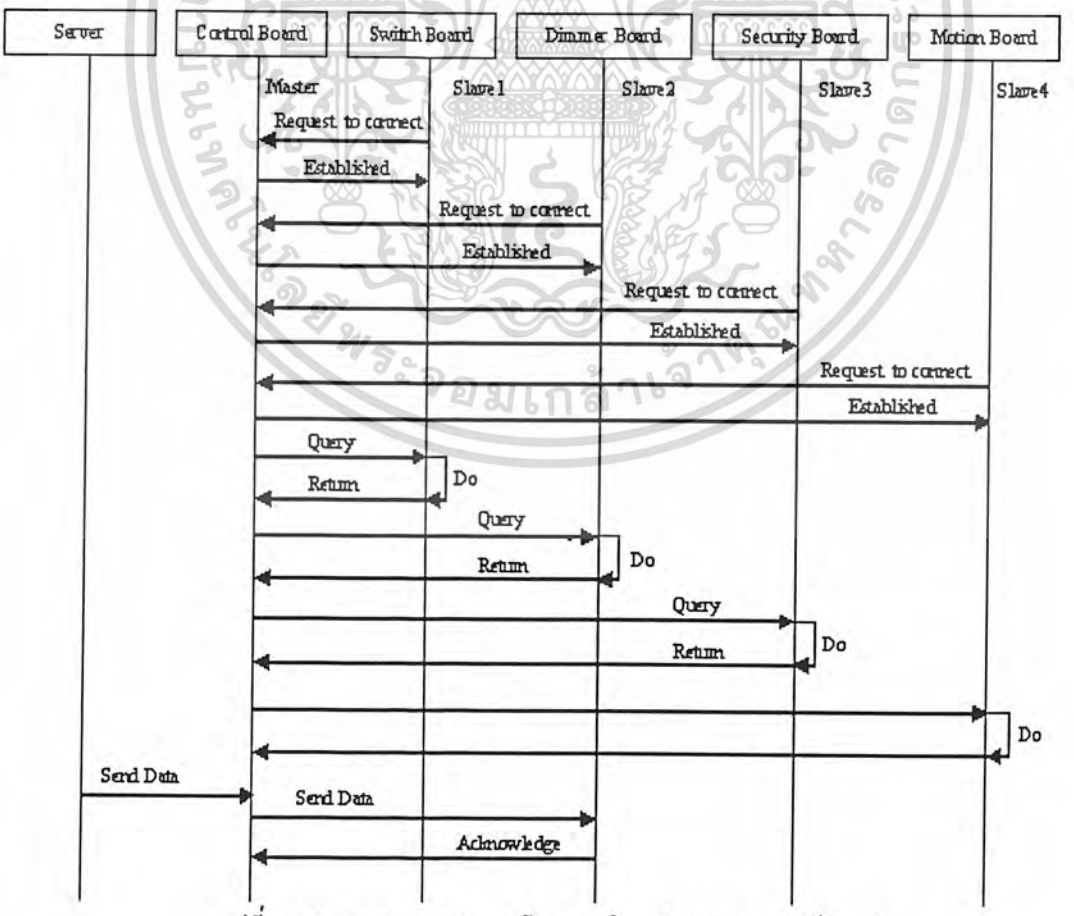
2. อินเทอเฟซโมดูลที่ใช้กับอุปกรณ์รักษาความปลอดภัย มีดังนี้

- บอร์ดระบบรักษาความปลอดภัย (Security Board) ทำหน้าที่ตรวจสอบความผิดปรกติภายในบ้านซึ่งจะสามารถตรวจจับผู้บุกรุก โดยใช้ PIR และตรวจจับควันหากเกิดเพลิงไหม้ภายในบ้านด้วยเครื่องตรวจจับควัน ถ้ามีความผิดปรกติเกิดขึ้นจะทำการส่งเสียงเตือนให้คนในบ้านได้รับรู้ และแจ้งความผิดปรกติให้กับบอร์ดควบคุม (Control Board)

3.5.2 การทำงานของระบบ

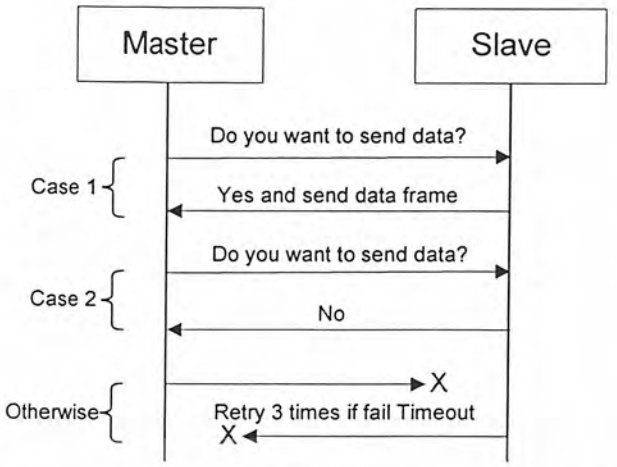
การทำงานของระบบจะใช้หลักการของมาสเตอร์สเลฟแพทเทิร์นในการทำงาน เนื่องจากความง่ายต่อการสร้างโปรแกรมและการขยายระบบเพิ่มหรือเพิ่มสเลฟ การทำงานหลักๆคือ อินเทอเฟซโมดูลหรือสเลฟ จะร้องขอการเชื่อมต่อกับ บอร์ดควบคุม หรือมาสเตอร์ แล้วมาสเตอร์จะทำการสร้างการเชื่อมต่อกับสเลฟ

หลังจากนั้นมาสเตอร์จะเป็นคนถาม สเลฟแต่ละตัวว่า ต้องการส่งข้อมูลหรือไม่ ถ้าไม่ก็จะถามสเลฟตัวต่อไปเป็นอย่างนี้ไปเรื่อยๆในรูปแบบของราวน์โรบิน (Round Robin) ถ้าสเลฟต้องการส่งข้อมูลก็จะรอจนถึงคิวของตัวเอง แล้วทำการส่งข้อมูลไปยังมาสเตอร์ด้วยโปรโตคอลที่กล่าวต่อไปในภายหลัง ผ่านทางคลื่นวิทยุ แต่ถ้ามาสเตอร์ต้องการส่งข้อมูลไปให้สเลฟตัวไหนก็ตาม จะสามารถส่งได้ทันที ดังนั้นโอกาสที่จะเกิดการชนกันของข้อมูลจึงมีน้อยมากเพราะมาสเตอร์จะเป็นคนจัดการเพียงคนเดียว



รูปที่ 3-21a แสดงการทำงานโดยรวมในรูปแบบมาสเตอร์สเลฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

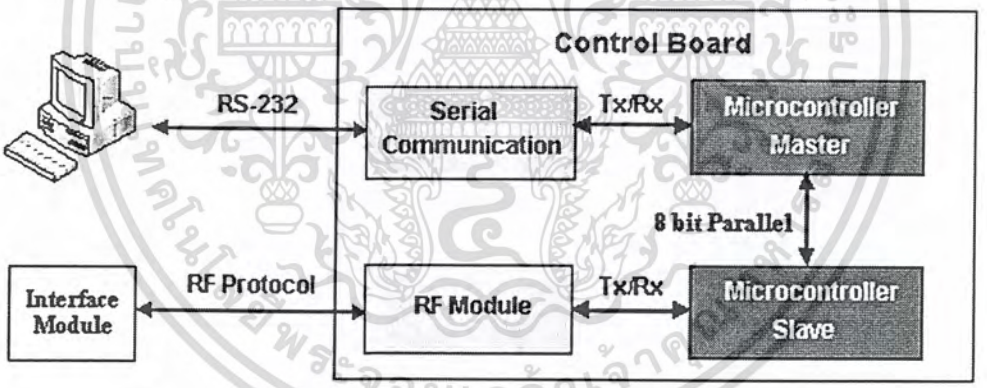


รูปที่ 3-21b แสดงการทำงานของระบบในรูปแบบมาสเตอร์สเลฟ

3.5.3 รายละเอียดโครงสร้างทางฮาร์ดแวร์

ในทุกๆบอร์ดจะมีไมโครคอนโทรลเลอร์ 8051 ทำหน้าที่ประมวลผลคำสั่ง และมี อาร์เอฟโมดูล ทำหน้าที่ติดต่อสื่อสารกับบอร์ดควบคุม โดยจะทำการติดต่อกันแบบผลัดกันรับ-ผลัดกันส่ง ซึ่งรายละเอียดต่างๆของแต่ละบอร์ดมีดังต่อไปนี้

รายละเอียดของบอร์ดควบคุม



รูปที่ 3-22 แสดงถึงรายละเอียดของบอร์ดควบคุม

ส่วนประกอบต่างๆที่สำคัญของบอร์ดควบคุม ประกอบด้วยดังนี้

1. การสื่อสารแบบอนุกรม(Serial Communication)จะทำหน้าที่ติดต่อข้อมูลระหว่างคอมพิวเตอร์กับบอร์ดควบคุม
2. ไมโครคอนโทรลเลอร์มาสเตอร์ (Microcontroller Master)
   - กรณีที่เซิร์ฟเวอร์ส่งข้อมูลมาให้ ไมโครคอนโทรลเลอร์มาสเตอร์ จะทำหน้าที่เป็นประมวลคำสั่งจากข้อมูลที่เซิร์ฟเวอร์ได้ส่งมา แล้วทำการส่งข้อมูลที่ได้ประมวลผลแล้วไปยัง ไมโครคอนโทรลเลอร์สเลฟ (Microcontroller Slave) โดยส่งข้อมูลแบบขนาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- กรณีที่ไมโครคอนโทรลเลอร์สเลฟ ส่งข้อมูลมาให้ ไมโครคอนโทรลเลอร์มาสเตอร์ จะทำการประมวลผล แล้วทำการส่งข้อมูลที่ได้ประมวลผลแล้วไปให้กับเซิร์ฟเวอร์
- ทำหน้าที่ในการเก็บฐานข้อมูลของอินเทอเฟซโมดูลแต่ละตัว

3. ไมโครคอนโทรลเลอร์สเลฟ ทำหน้าที่ในการรอรับการร้องขอเชื่อมต่อจากอินเทอเฟซโมดูล และตรวจสอบสถานะการทำงานของอินเทอเฟซโมดูลและรับ-ส่งข้อมูลกับอินเทอเฟซโมดูล หรือไมโครคอนโทรลเลอร์มาสเตอร์ ซึ่งข้อมูลที่รับ-ส่งกับอินเทอเฟซโมดูล จะใช้ อาร์เอฟโมดูล เป็นตัวรับ-ส่ง ข้อมูลที่ส่งจะเป็นข้อมูลแบบอนุกรม
4. อาร์เอฟโมดูล ทำหน้าที่ในการติดต่อสื่อสารระหว่างบอร์ดควบคุม กับ อินเทอเฟซโมดูล

หลักการทำงานของบอร์ดควบคุม

บอร์ดควบคุม จะมี ส่วนสื่อสาร เป็นสื่อกลางซึ่งทำหน้าที่เชื่อมต่อระหว่างเซิร์ฟเวอร์กับบอร์ดควบคุม ทางพอร์ตอนุกรม (RS-232) โดยส่วนสื่อสาร จะแปลงสัญญาณให้อยู่ในระดับ TTL เพื่อให้ บอร์ดควบคุม กับ เซิร์ฟเวอร์สื่อสารกันได้

ไมโครคอนโทรลเลอร์มาสเตอร์ จะเริ่มการเชื่อมต่อกับเซิร์ฟเวอร์ และรอรับการเชื่อมต่อจาก อินเทอเฟซโมดูล เมื่อมีอินเทอเฟซโมดูล เข้ามาร้องขอการเชื่อมต่อ ไมโครคอนโทรลเลอร์สเลฟ ก็จะสร้างการเชื่อมต่อ พร้อมทั้งเก็บสถานะการเชื่อมต่อของอินเทอเฟซโมดูล และส่งข้อมูลไปบอกไมโครคอนโทรลเลอร์มาสเตอร์ จากนั้นไมโครคอนโทรลเลอร์มาสเตอร์ จะบันทึกข้อมูลต่างๆของอินเทอเฟซโมดูล ลงฐานข้อมูล และส่งข้อมูลไปบอกเซิร์ฟเวอร์ว่ามีอินเทอเฟซโมดูลได้เข้ามาอยู่ในระบบแล้ว

ถ้ายังไม่มีการส่งข้อมูลใดๆ ไมโครคอนโทรลเลอร์สเลฟ จะทำการตรวจสอบสถานะการทำงานของอินเทอเฟซโมดูล อยู่เรื่อยๆในรูปแบบมาสเตอร์สเลฟตามที่ได้กล่าวไว้แล้ว

เมื่ออินเทอเฟซโมดูล ส่งข้อมูลมา ไมโครคอนโทรลเลอร์สเลฟ จะส่งต่อไปให้ไมโครคอนโทรลเลอร์มาสเตอร์ เมื่อ ไมโครคอนโทรลเลอร์มาสเตอร์ ประมวลผลเสร็จจะทำการบันทึกลงฐานข้อมูลและส่งข้อมูลไปให้เซิร์ฟเวอร์

ถ้าเซิร์ฟเวอร์ทำการส่งข้อมูลมาให้ ไมโครคอนโทรลเลอร์มาสเตอร์ จะทำการประมวลผลและบันทึกข้อมูลลงฐานข้อมูล แล้วส่งต่อไปยัง ไมโครคอนโทรลเลอร์สเลฟ อีกที

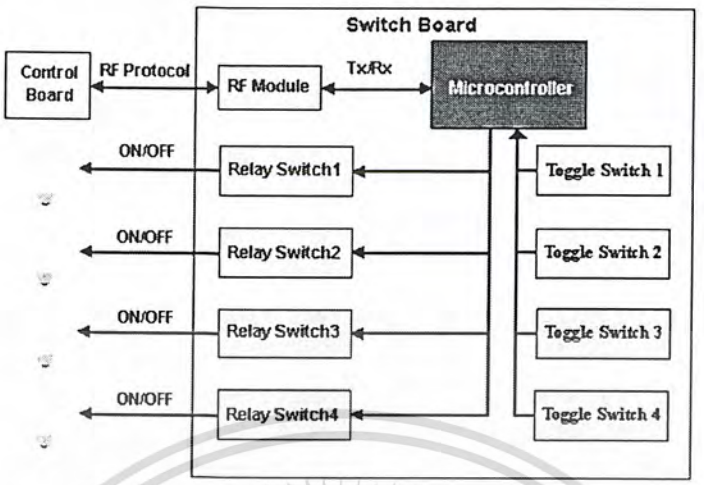
รายละเอียดของบอร์ดสวิตช์

ส่วนประกอบต่างๆที่สำคัญของบอร์ดสวิตช์ ประกอบด้วยดังนี้

1. อาร์เอฟโมดูล ทำหน้าที่ในการติดต่อสื่อสารระหว่างบอร์ดสวิตช์กับบอร์ดควบคุม
2. ไมโครคอนโทรลเลอร์ ทำหน้าที่ในการประมวลผลข้อมูลที่ได้รับจาก บอร์ดควบคุม แล้วสั่งให้รีเลย์ทำการเปิด-ปิดอุปกรณ์ไฟฟ้า ซึ่งจะมีรีเลย์ทั้งหมด 4 ตัว นั่นคือสามารถควบคุมเปิด-ปิดอุปกรณ์ไฟฟ้าภายในบ้านได้ถึง 4 ตัว ต่อ 1 บอร์ด และในการควบคุมการทำงานของรีเลย์จะใช้เพียง 1 บิต ต่อการควบคุมการทำงานของรีเลย์ 1 ตัว
3. รีเลย์ (Relay Switch) เป็นสวิตช์อัตโนมัติ ที่ทำหน้าที่ในการเปิด-ปิดอุปกรณ์ไฟฟ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ปุ่มทอกเกิล (Toggle Switch) เป็นสวิตซ์กดติดปล่อยดับ ทำหน้าเปิดปิดอุปกรณ์ไฟฟ้าด้วยมือ



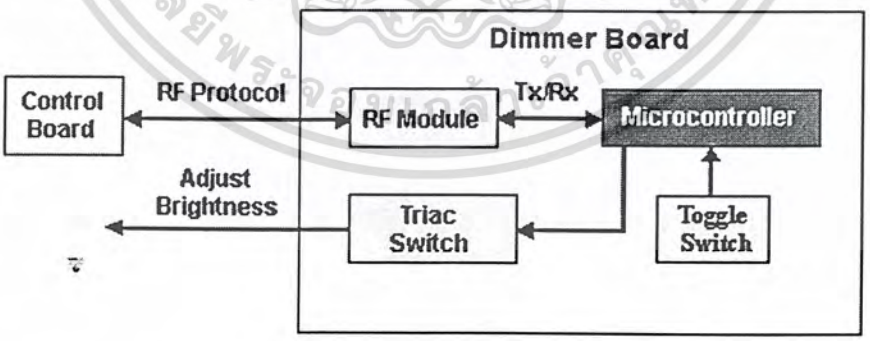
รูปที่ 3-23 แสดงรายละเอียดของบอร์ดสวิตซ์

หลักการทำงานของ บอร์ดสวิตซ์

บอร์ดสวิตซ์ จะทำการร้องขอการเชื่อมต่อกับ บอร์ดควบคุม เมื่อสามารถติดต่อได้แล้วก็จะส่งสถานะการทำงานของรีเลย์ทุกตัวไปให้บอร์ดควบคุม

สำหรับการสั่งงานเปิด-ปิดอุปกรณ์ไฟฟ้านั้น สามารถสั่งได้2ทางคือบอร์ดควบคุม ส่งข้อมูลมาเปิด-ปิดอุปกรณ์ไฟฟ้า หรือ เปิด-ปิดอุปกรณ์ไฟฟ้าด้วยมือ โดยใช้ ปุ่มทอกเกิล และจะมี LED แสดงสถานะการทำงานของรีเลย์แต่ละตัวด้วย

รายละเอียดของ บอร์ดปรับระดับความสว่าง



รูปที่ 3-24 แสดงรายละเอียดของบอร์ดปรับระดับความสว่าง

ส่วนประกอบต่างๆที่สำคัญของบอร์ดปรับระดับความสว่าง ประกอบด้วยดังนี้

1. อาร์เอฟโมดูล ทำหน้าที่ในการติดต่อสื่อสารระหว่างบอร์ดปรับระดับความสว่างกับบอร์ดควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ไมโครคอนโทรลเลอร์ ทำหน้าที่ในการประมวลผลข้อมูลที่ได้รับจาก บอร์ดควบคุม แล้วทำการควบคุมไตรแอกให้ทำงาน เพื่อปรับความสว่างของหลอดไฟ
3. ไตรแอก (Triac Switch) ทำหน้าที่ในการปรับความสว่างของหลอดไฟ โดยไตรแอกจะทำงานแบบสวิตช์ชิ่งความเร็วสูง
4. ปุ่มทอกเกิลทำหน้าที่ในการเลือกรูปแบบการควบคุมแบบอัตโนมัติ หรือ ความคุมด้วยมือ

หลักการทำงานของ บอร์ดปรับระดับความสว่าง

บอร์ดปรับระดับความสว่าง จะทำการร้องขอการเชื่อมต่อกับ บอร์ดควบคุม เมื่อสามารถติดต่อได้แล้วก็จะส่งสถานะของระดับสว่างปัจจุบันไปให้บอร์ดควบคุม

สำหรับการสั่งงานให้ปรับระดับความสว่างของหลอดไฟนั้น สามารถสั่งได้2ทางคือบอร์ดควบคุม ส่งข้อมูลมาให้ปรับระดับความสว่างของหลอดไฟ หรือปรับระดับความสว่างของหลอดไฟด้วยมือ โดยใช้ ปุ่มทอกเกิลและจะมี LED แสดงสถานะการทำงานว่าตอนนี้อยู่ในรูปแบบการควบคุมแบบอัตโนมัติหรือควบคุมด้วยมือ

สำหรับการปรับความสว่างนั้นไมโครคอนโทรลเลอร์จะทำการควบคุมไตรแอก ซึ่งไตรแอกจะทำงานแบบสวิตชิ่งด้วยอัตราการสวิตช์ครึ่งคลื่นของความถี่ไฟบ้าน ซึ่งการทำงานของไตรแอกจะได้รับการกระตุ้นด้วยความถี่ไฟบ้านในครึ่งคลื่นรูปสัญญาณเพื่อจ่ายแรงดันไปยังหลอดไฟที่มีการกินกำลังงานต่ำ นั่นคือช่วงที่มีการหรี่ไฟเกิดขึ้น และการทำงานหนึ่งคือการทำงานที่กำลังสูงหรือทำงานเต็มที่ ไตรแอกจะได้รับการกระตุ้นจากความถี่ไฟบ้านในครึ่งคลื่นรูปสัญญาณเช่นกัน แต่จะจ่ายแรงดันไฟบ้านไปยังหลอดไฟอย่างเต็มที่ นั่นคือช่วงเวลาที่ไม่เกิดการหรี่ไฟ ซึ่งการควบคุมนี้เราจะใช้ “การควบคุมทางเฟส” เป็นตัวควบคุมการทำงานของไตรแอก

การทำงานเป็นสวิตชิ่งความเร็วสูงของไตรแอกจะก่อให้เกิดสัญญาณทรานเซียนต์ อยู่ในย่านความถี่ไฟบ้านขึ้นไปจนถึง 30 เมกะเฮิรตซ์ หรือมากกว่า ผลที่ตามมาจากสัญญาณทรานเซียนต์นี้ก็คือ จะได้ยินเสียงวี๊ดหรือเป็นสัญญาณรบกวนออกมาทางลำโพงของเครื่องรับวิทยุ ซึ่งการแก้ไขปัญหานี้จะกล่าวต่อไปในเรื่องของการออกแบบวงจร

ในส่วนของบอร์ดปรับระดับความสว่าง นี้ ได้ออกแบบเพื่อใช้ในการปรับความสว่างของหลอดไฟเท่านั้น ซึ่งสามารถใช้ได้แบบหลอดไส้เพียงอย่างเดียว

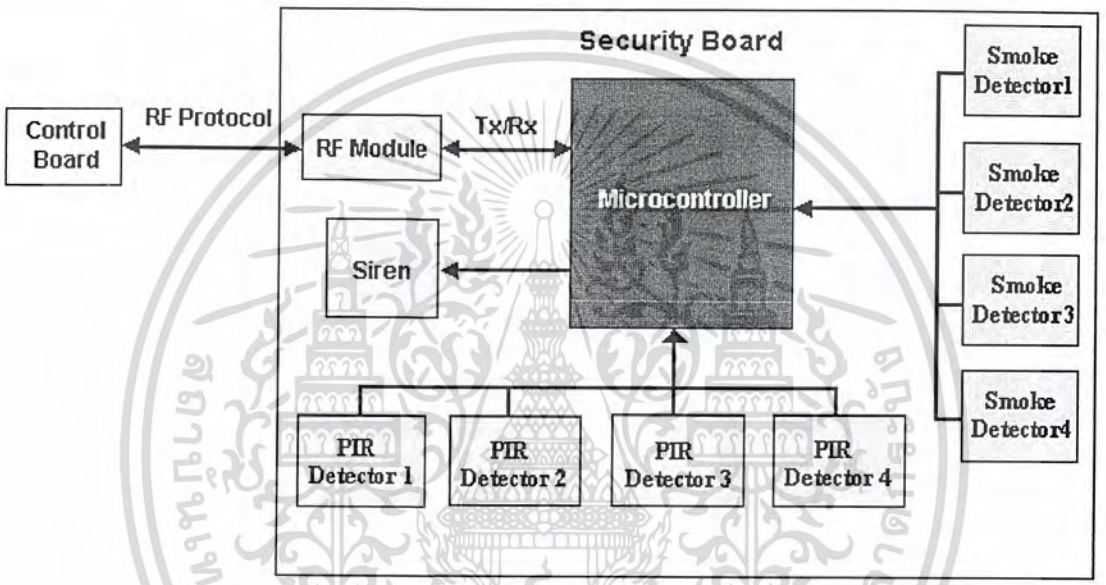
รายละเอียดของ บอร์ดระบบรักษาความปลอดภัย

ส่วนประกอบต่างๆที่สำคัญของบอร์ดระบบรักษาความปลอดภัย ประกอบด้วยดังนี้

1. อาร์เอฟโมดูล ทำหน้าที่ในการติดต่อสื่อสารระหว่างบอร์ดระบบรักษาความปลอดภัยกับบอร์ดควบคุม
2. ไมโครคอนโทรลเลอร์ ทำหน้าที่รับข้อมูลจากเซ็นเซอร์ต่างๆแล้วประมวลผล และควบคุมการทำงานของไซเรนเตือนภัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เครื่องตรวจจับความเคลื่อนไหว (PIR Detector) เป็นเซ็นเซอร์สำเร็จรูปที่ใช้สำหรับตรวจจับผู้บุกรุก ซึ่งสามารถต่อได้สูงสุด 4 ชุด ต่อ 1 บอร์ด ซึ่งจะติดตามประตู,หน้าต่างหรือตามทางเดินก็ได้ การทำงานของเซ็นเซอร์ชนิดนี้คือ เซ็นเซอร์จะตรวจจับรังสีอินฟราเรดที่มนุษย์หรือสัตว์แผ่ออกมา
4. เครื่องตรวจจับควัน (Smoke Detector) เป็นเซ็นเซอร์สำเร็จรูปที่ใช้สำหรับตรวจจับควันซึ่งสามารถใช้ในการตรวจสอบได้ว่าเกิดเพลิงไหม้หรือไม่ โดยกลุ่มควันที่ตรวจจับจะต้องเป็นกลุ่มควันขนาดใหญ่
5. Siren เป็นไซเรนเตือนภัย เมื่อไซเรนนี้ดังขึ้นแสดงว่าได้เกิดเหตุการณ์ผิดปรกติแล้ว



รูปที่ 3-25 แสดงรายละเอียดของบอร์ดระบบรักษาความปลอดภัย

หลักการทำงานของ บอร์ดระบบรักษาความปลอดภัย

บอร์ดระบบรักษาความปลอดภัย จะทำการร้องขอการเชื่อมต่อกับ บอร์ดควบคุม เมื่อสามารถติดต่อได้แล้วก็จะส่งสถานะปัจจุบันของเซ็นเซอร์ทุกตัวไปให้ บอร์ดควบคุม

เซ็นเซอร์ต่างๆจะทำการตรวจสอบความผิดปรกติ เมื่อพบว่ามีความผิดปรกติเกิดขึ้น ก็จะส่งข้อมูลไปให้ไมโครคอนโทรลเลอร์ จากนั้นไมโครคอนโทรลเลอร์จะสั่งให้ไซเรนดังขึ้น แล้วทำการส่งสถานะของเซ็นเซอร์ทุกตัวไปให้ บอร์ดควบคุม

3.5.4 การออกแบบวงจรบอร์ดควบคุม

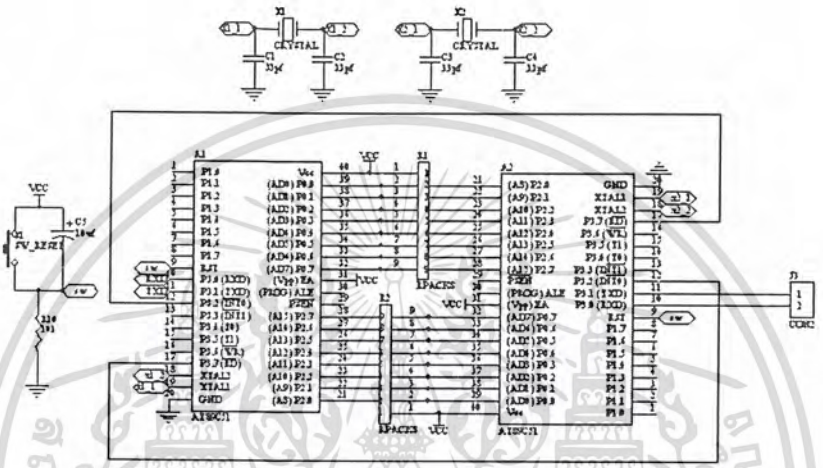
จะประกอบด้วยส่วนต่างๆดังต่อไปนี้

ส่วนควบคุม

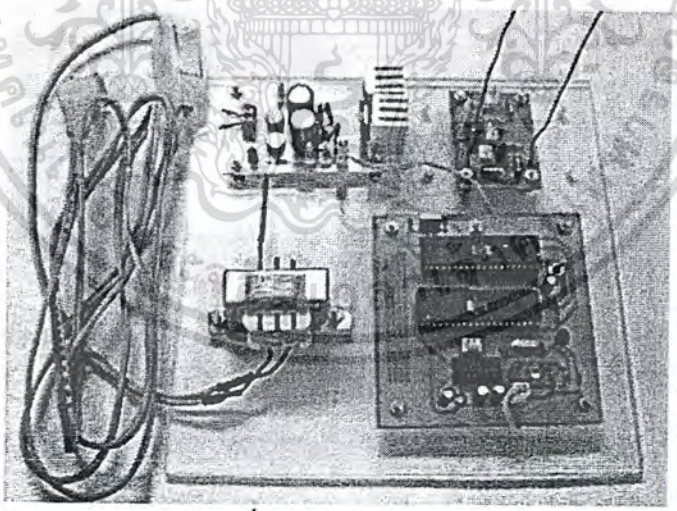
จะใช้ไมโครคอนโทรลเลอร์ 8051 จำนวน2 ตัว ในการควบคุม ซึ่งจะทำงานแบบมาสเตอร์และสเลฟ โดยมาสเตอร์จะส่งข้อมูลอนุกรมโดยผ่านทางพอร์ต Rx และ Tx แต่ระดับของสัญญาณจะอยู่ใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระดับของ TTL คือ ลอจิก "1" จะมีแรงดัน 5 โวลต์ และ ลอจิก "0" จะมีแรงดัน 0 โวลต์ และมาสเตอร์จะใช้พอร์ต0 ในการส่งข้อมูลไปให้สเลฟ โดยใช้พอร์ต3.7ในการส่งสัญญาณอินเตอร์รัปต์ในการบอกสเลฟว่ามีการส่งข้อมูลมาให้ และรับข้อมูลจากสเลฟผ่านทางพอร์ต2 โดยสเลฟจะส่งสัญญาณอินเตอร์รัปต์มาที่พอร์ต3.2 ของมาสเตอร์เพื่อบอกให้มาสเตอร์ทำการรับข้อมูล ในทำนองเดียวกันสเลฟจะใช้พอร์ต2 ในการรับข้อมูล โดยมาสเตอร์จะส่งสัญญาณอินเตอร์รัปต์มาที่พอร์ต3.2 และสเลฟส่งข้อมูลผ่านทางพอร์ต0 โดยจะส่งสัญญาณอินเตอร์รัปต์ไปบอกมาสเตอร์ผ่านทางพอร์ต3.7 ของสเลฟ และสเลฟจะส่งข้อมูลอนุกรมผ่านทางพอร์ต Rx และ Tx ไปยัง อาร์เอฟโมดูล



รูปที่ 3-26 แสดงวงจรควบคุมของบอร์ดควบคุม

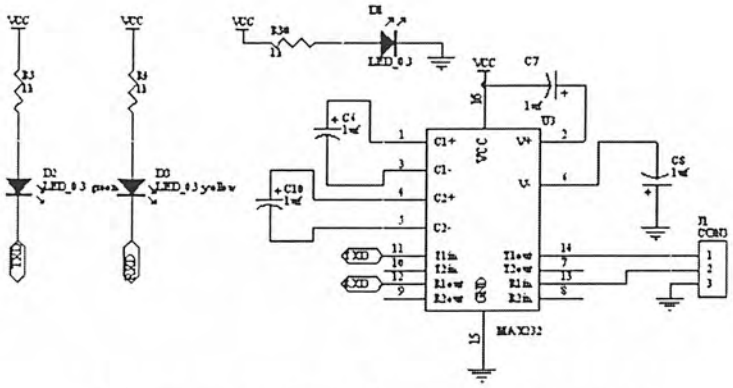


รูปที่ 3-27 บอร์ดควบคุม

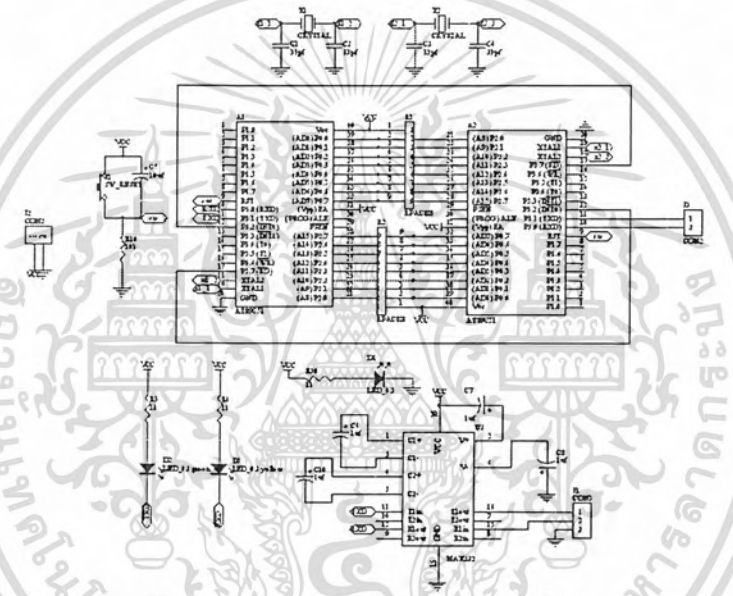
ส่วนติดต่อพอร์ตอนุกรม

พอร์ตสื่อสารข้อมูลแบบอนุกรม เป็นพอร์ตภายในของไมโครคอนโทรลเลอร์ ซึ่งติดต่อได้ด้วยซอร์ฟแวร์โดยใช้ไอซี MAX232 เป็นการสื่อสารแบบ RS-232 โดยมาสเตอร์จะสื่อสารกับเซิร์ฟเวอร์แบบผลัดกันรับ-ผลัดกันส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3-28 แสดงวงจรติดต่อสื่อสารแบบอนุกรมRS-232



รูปที่ 3-29 แสดงภาพรวมทั้งหมดของวงจรบอร์ดควบคุม

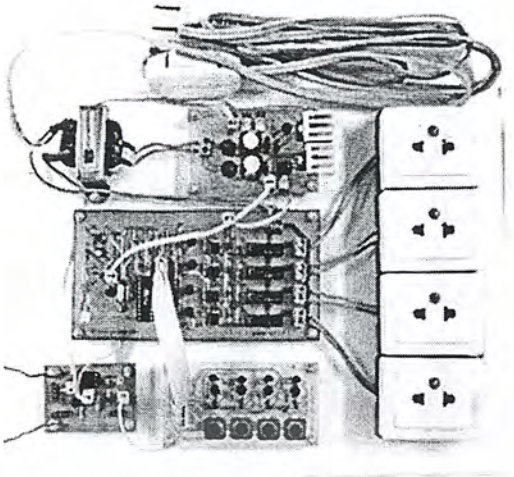
3.5.5 การออกแบบวงจรบอร์ดสวิตซ์

ประกอบด้วยส่วนต่างๆ ดังต่อไปนี้

ส่วนควบคุม

จะใช้ไมโครคอนโทรเลอร์ 8051 เป็นตัวควบคุมการทำงานทั้งหมด โดยจะส่งข้อมูลอนุกรมผ่านทางพอร์ต Rx และ Tx ไปยังอาร์เอฟโมดูล ใช้พอร์ต1.0-1.3 เป็นพอร์ตเอาท์พุตสำหรับควบคุมการทำงานของรีเลย์ พอร์ต1.4-1.7 เป็นพอร์ตอินพุตสำหรับเก็บสถานะการกดปุ่มทอกเกิลและพอร์ต3.2-3.5 เป็นพอร์ตเอาท์พุตสำหรับแสดงสถานะการทำงานของรีเลย์ด้วย LED

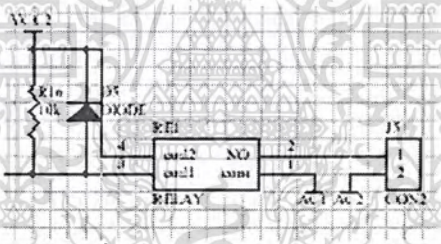
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3-30 บอร์ดสวิตซ์

ส่วนเปิด-ปิดอุปกรณ์ไฟฟ้า

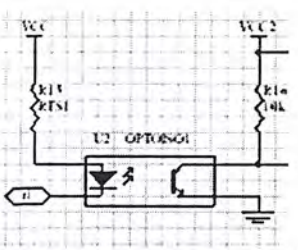
ใช้รีเลย์ในการควบคุมการเปิด-ปิดอุปกรณ์ไฟฟ้า โดยจะมีรีเลย์ทั้งหมด 4 ตัว ดังนั้นจะสามารถเปิด-ปิดอุปกรณ์ไฟฟ้าได้ทั้งหมด4ตัว ซึ่งการควบคุมรีเลย์นั้นจะมีไมโครคอนโทรลเลอร์เป็นตัวควบคุมอีกทีหนึ่งโดยจะจ่ายแรงดันไฟฟ้าให้ตกคร่อมที่ขดลวดภายในรีเลย์ ทำให้ขดลวดเกิดการเหนี่ยวนำสนามแม่เหล็กไฟฟ้า ซึ่งจะทำให้หน้าสัมผัสของรีเลย์ติดกัน นั่นก็คือการเปิดอุปกรณ์ไฟฟ้านั่นเอง



รูปที่ 3-31 แสดงวงจรของรีเลย์

ส่วนแยกกราวน์วงจร

เนื่องจากเมื่อรีเลย์ทำงานจะก่อให้เกิดสัญญาณรบกวนจนอาจจะทำให้ไมโครคอนโทรลเลอร์หยุดทำงาน จึงต้องมีออปโต้คัปเปลอร์ใช้ในการแยกกราวน์ของส่วนควบคุม กับ ส่วนเปิดปิดอุปกรณ์ไฟฟ้า

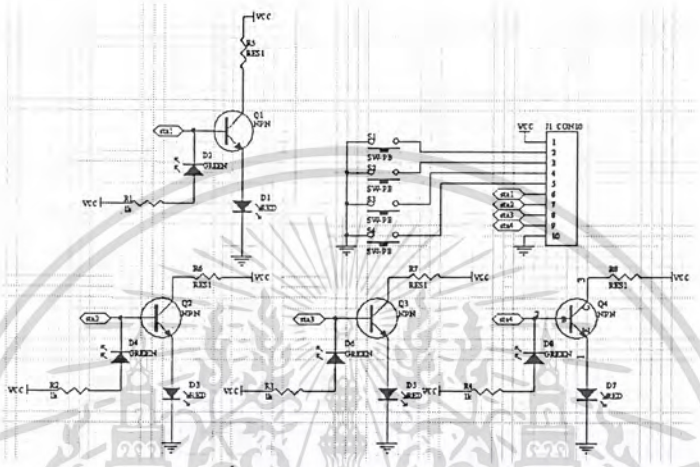


รูปที่ 3-32 แสดงวงจรของออปโตคัปเปลอร์

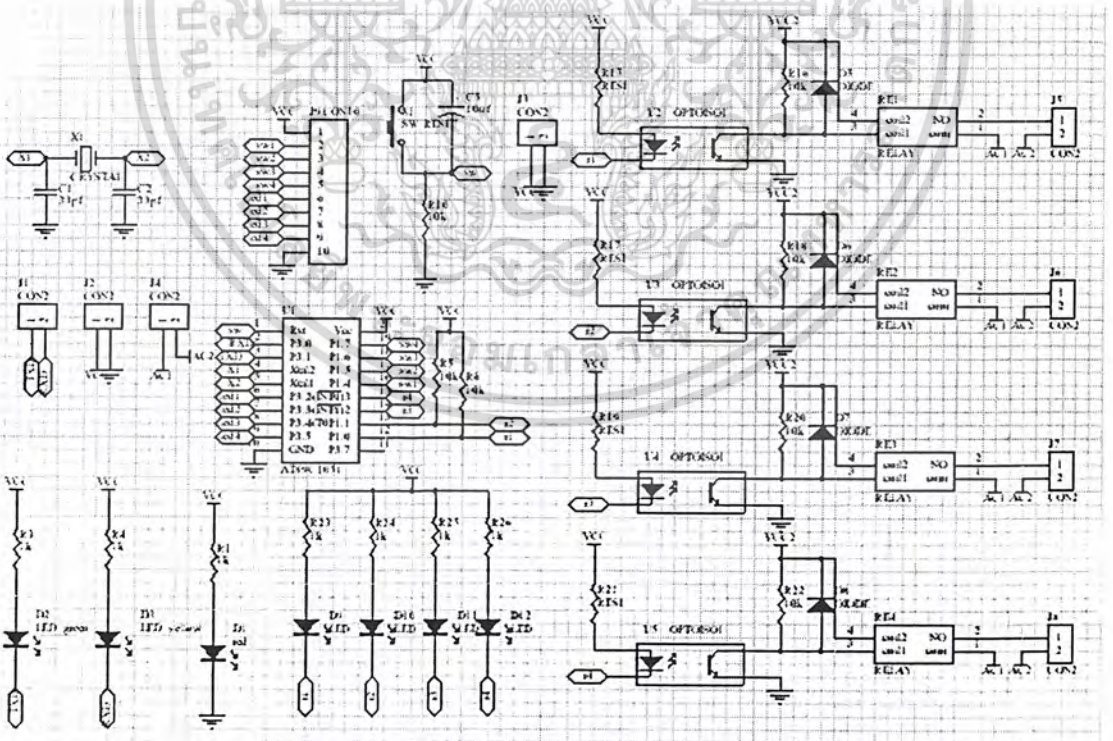
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนทอกเกิล

จะมีสวิตซ์กดติดปล่อยดับจะเป็นอินพุตให้กับไมโครคอนโทรเลอร์ โดยใช้สำหรับเปิด-ปิดอุปกรณ์ไฟฟ้า เมื่อทำการกดสวิตซ์จะทำให้อินพุตเป็นลอจิก “0” และเมื่อปล่อยสวิตซ์จะทำให้อินพุตเป็นลอจิก “1” และจะมีไฟแสดงสถานะการทำงานของรีเลย์ทั้งหมด 4 ชุด ซึ่งจะขออธิบายเฉพาะชุดแรกเท่านั้น โดยเมื่อมีลอจิก “0” ผ่านเข้ามาที่ sta1 จะทำให้ D2 สว่าง และ D1 ดับ แต่ถ้ามีลอจิก “1” ผ่านเข้ามาจะทำให้ D2 ดับ และ D1 สว่าง



รูปที่ 3-33 แสดงวงจรของรีเลย์



รูปที่ 3-34 แสดงภาพรวมทั้งหมดของวงจรเปิด-ปิดอุปกรณ์ไฟฟ้า

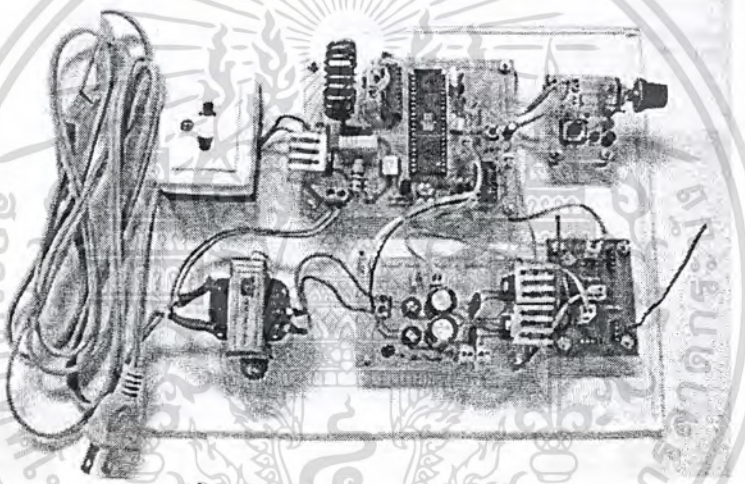
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5.6 การออกแบบวงจรบอร์ดปรับระดับความสว่าง

ประกอบด้วยส่วนต่างๆดังต่อไปนี้

ส่วนควบคุม

จะใช้ไมโครคอนโทรเลอร์ 8051 เป็นตัวควบคุมการทำงานทั้งหมด โดยจะส่งข้อมูลอนุกรมผ่านทางพอร์ต Rx และ Tx ไปยังอาร์เอฟโมดูล ใช้พอร์ต2 เป็นพอร์ตอินพุตจากA/D ใช้สำหรับอ่านค่าที่ A/D อ่านได้ พอร์ต1.1 เป็นพอร์ตจากสัญญาณชิงโครนัสที่ได้จากการเปรียบเทียบแรงดันของออปแอมป์ พอร์ต 1.6 เป็นพอร์ตอินพุทสำหรับเก็บสถานะการกดปุ่มทอกเกิล พอร์ต1.0 เป็นพอร์ตเอาท์พุตสำหรับการควบคุมการทำงานของไตรแอก พอร์ต1.7 เป็นพอร์ตที่ควบคุมการแสดงสถานะของรูปแบบการทำงานแบบอัตโนมัติหรือควบคุมด้วยมือ พอร์ต3.2 เป็นพอร์ตอินพุตสำหรับตรวจสอบอินเตอร์รัปต์จากA/D พอร์ต3.6เป็นพอร์ตเอาท์พุตสำหรับการควบคุมให้A/D เขียน และพอร์ต3.7เป็นเอาท์พุตสำหรับควบคุมให้ A/D อ่านค่าแล้วค้างค่าไว้

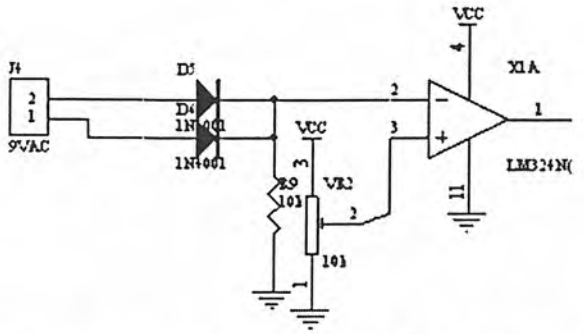


รูปที่ 3-35 บอร์ดปรับระดับความสว่าง

วงจรเปรียบเทียบแรงดัน

เนื่องจากการควบคุมความสว่างของหลอดไฟด้วยไตรแอกนั้น จะต้องมีสัญญาณที่จะมาอ้างอิงกับสัญญาณไฟบ้าน เพื่อที่ไมโครคอนโทรลเลอร์จะได้ทริกที่เฟสของไฟบ้านได้อย่างถูกต้อง ซึ่งวิธีการที่ง่ายที่สุดคือ การนำเอาออปแอมป์มาทำการเปรียบเทียบแรงดัน โดยอินพุตที่ขา inverter จะเป็นแรงดันไฟฟ้า 12 VAC ที่ผ่านไดโอดมาแล้วในลักษณะครึ่งคลื่นของสัญญาณ และอินพุตที่ขา noninverter จะเป็นแรงดันอ้างอิงที่ใช้สำหรับเปรียบเทียบ โดยจะมีค่าประมาณ 2 โวลต์ และเอาท์พุตที่ได้จะเป็นสัญญาณอ้างอิงให้กับไมโครคอนโทรลเลอร์ในลักษณะกลับขั้วกับสัญญาณอินพุตด้านบวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

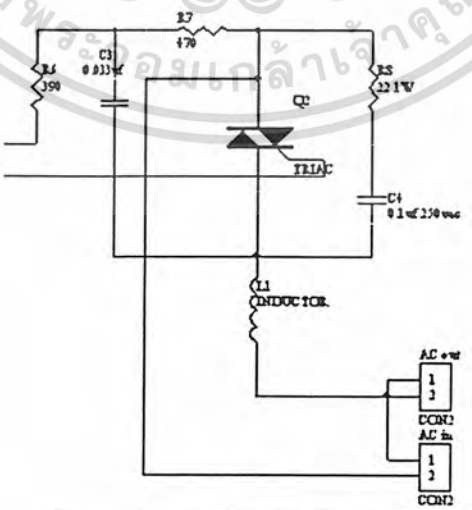


รูปที่ 3-36 แสดงวงจรเปรียบเทียบแรงดันโดยใช้ออปแอมป์

ส่วนไตรแอก

การควบคุมโหลดทางเอาท์พุตของไตรแอก จะมีขาเกต(G) เป็นตัวกระตุ้นกระแสเกตให้กับไตรแอก โดยจะมี R6 และ R7 กำหนดค่ากระแสร่วมกับขาเกต R8 และ C4 ทำหน้าที่เป็นวงจรสนับเบอร์ (snubber) ซึ่งจะช่วยให้ไตรแอกทำงานได้อย่างเหมาะสมที่ช่วงครึ่งคลื่นสุดท้ายของวงรอบการกระตุ้นเมื่อโหลดที่มาต่อกับเอาท์พุตเป็นพวกตัวเหนี่ยวนำเช่น หลอดไส้ เป็นต้น นอกจากนี้ยังมี C3 ทำหน้าที่แท็ปออกเพื่อป้องกันสัญญาณรบกวนต่างๆ ที่จะรบกวนขณะไตรแอกทำงาน

จากการที่ไตรแอกทำงานแบบสวิตชิ่งความเร็วสูงจะก่อให้เกิดสัญญาณทรานเซียนต์ อยู่ในย่านความถี่ไฟบ้านขึ้นไปถึง 30 เมกะเฮิรตซ์หรืออาจจะมากกว่า ผลที่ตามมาคือจะได้ยินเสียงวี๊ดหรือเป็นสัญญาณรบกวนออกทางลำโพงของเครื่องรับวิทยุ ซึ่งแก้ไขโดยการเพิ่มส่วนของวงจรป้องกันและลดทอนสัญญาณความถี่วิทยุออกด้วยการกรองความถี่ต่ำ โดยใช้ L1 ซึ่งมีประโยชน์มากสามารถลดการแพร่กระจายสัญญาณรบกวนลงได้มากและมีผลดีมาก ซึ่ง L1 ถูกสร้างมาบนผงเหล็กอัดแบบทอรอยด์และใช้ขดลวดที่มีค่า Q แฟกเตอร์ต่ำๆ ก็จะสามารถหยุดยั้งการออสซิเลตสัญญาณรบกวนเนื่องจากตัว L1 เองลงได้มากด้วย เมื่อต้องใช้ร่วมกับวงจรไตรแอกสวิตชิ่งความเร็วสูง ดังรูป

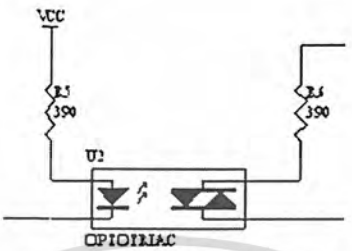


รูปที่ 3-37 แสดงวงจรควบคุมโหลดทางเอาท์พุตโดยใช้ไตรแอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนแยกกราวน์ของวงจร

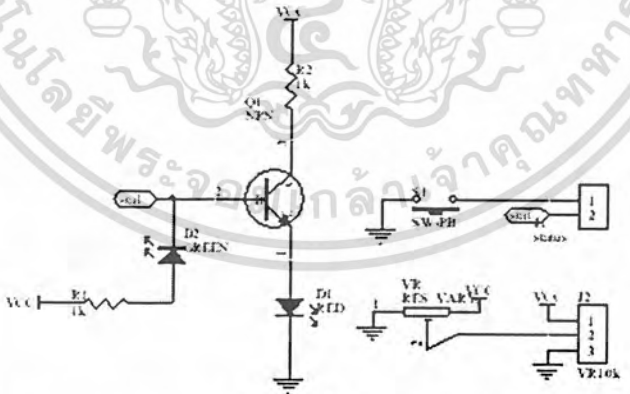
ออปโตคัปเปลอร์จะทำงานร่วมกับวงจรสนับเบอร์และทำการป้องกันการกระโดดข้ามและหลีกเลี่ยงการใช้งานกราวด์ร่วมกันของวงจรแรงดันต่ำกับวงจรแรงดันสูง



รูปที่ 3-38 แสดงวงจรแยกกราวน์แรงดันต่ำกับแรงดันสูงโดยใช้ออปโตคัปเปลอร์

ส่วนทอกเกิล

จะมีสวิตซ์กดติดปล่อยดับจะเป็นอินพุตให้กับไมโครคอนโทรเลอร์ โดยใช้สำหรับการเลือกสถานะการทำงานแบบอัตโนมัติหรือการควบคุมแบบมือ เมื่อทำการกดสวิตซ์จะทำให้อินพุตเป็นลอจิก “0” และเมื่อปล่อยสวิตซ์จะทำให้อินพุตเป็นลอจิก “1” และจะมีไฟแสดงสถานะการทำงานของรูปแบบการใช้งาน โดยเมื่อมีลอจิก “0” ผ่านเข้ามาที่ sta1 จะทำให้ D2 สว่าง และ D1 ดับ แต่ถ้ามีลอจิก “1” ผ่านเข้ามาจะทำให้ D2 ดับ และ D1 สว่าง นอกจากนี้ยังมี VR ซึ่งทำหน้าที่ในการปรับระดับแรงดันไฟฟ้าระหว่าง 0-5 โวลต์ เพื่อเป็นอินพุตให้กับวงจร A/D ในการอ่านค่าแรงดันอ้างอิงเพื่อใช้สำหรับปรับระดับความสว่างของหลอดไฟ



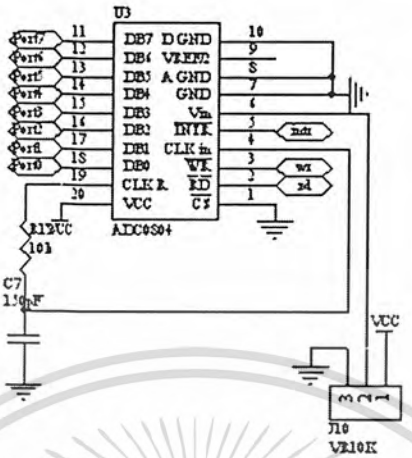
รูปที่ 3-39 แสดงวงจรแยกกราวน์แรงดันต่ำกับแรงดันสูงโดยใช้ออปโตคัปเปลอร์

ส่วนของA/D

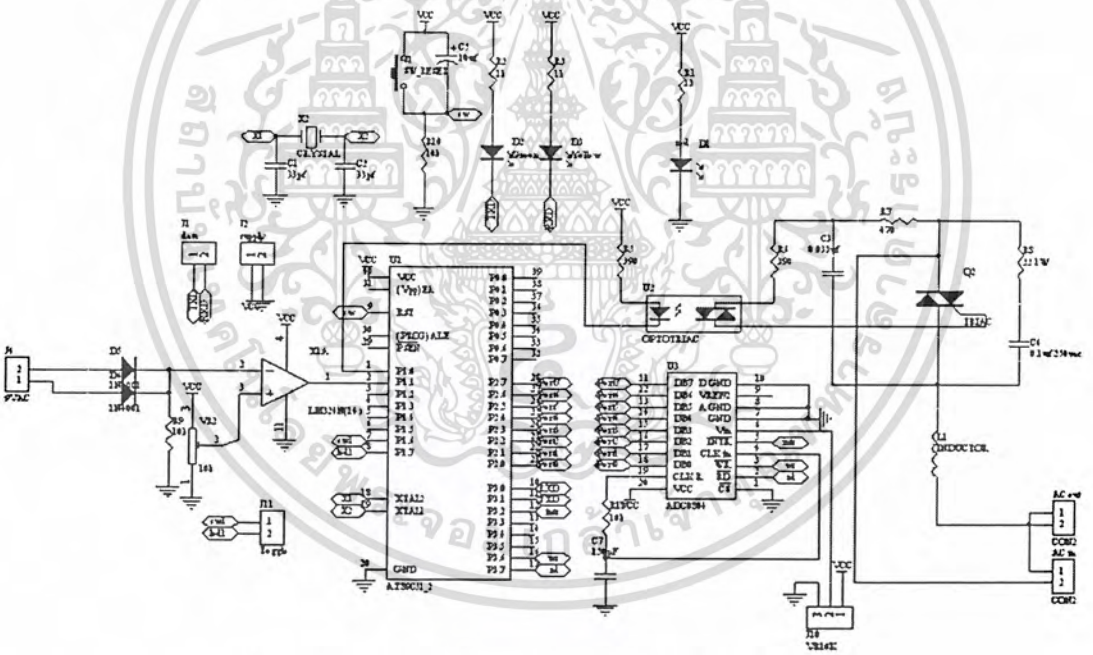
ในโครงงานนี้จะใช้เบอร์ ADC0804 เพราะสามารถหาซื้อได้ง่าย มีราคาถูก และใช้อัลกอริทึมแบบ SA โดย A/D นี้ จะมี1 อินพุต และ เอาท์พุตขนาด 8 บิต ดังนั้นสามารถอ่านค่าได้ทั้งหมด256ค่า โดย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อินพุตจะมาจาก VR ที่ได้ทำการปรับแรงดันไฟฟ้า แล้วแปลงให้อยู่ในรูปแบบข้อมูลดิจิตอล เพื่อให้ไมโครคอนโทรลเลอร์ทำการอ่านค่า



รูปที่ 3-40 แสดงวงจรอ่านค่าจากอนากลอกเป็นดิจิตอลโดยใช้ A/D



รูปที่ 3-41 แสดงภาพรวมทั้งหมดของวงจรปรับระดับความสว่าง

3.5.7 การออกแบบวงจรบอร์ดระบบรักษาความปลอดภัย

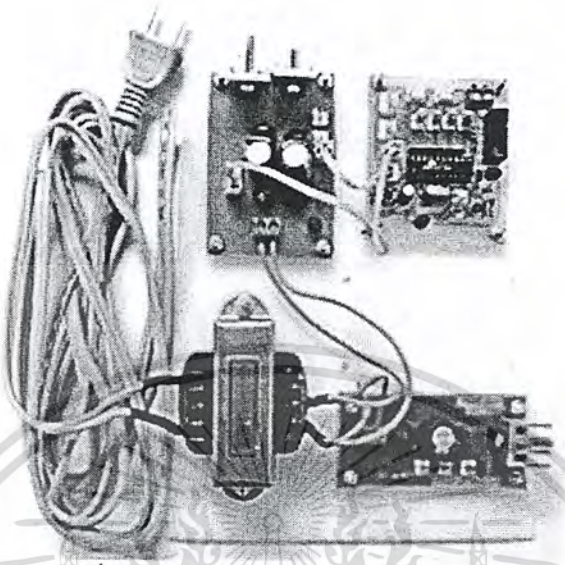
จะประกอบด้วยส่วนหลักๆดังนี้

ส่วนควบคุม

จะใช้ไมโครคอนโทรเลอร์ 8051 เป็นตัวควบคุมการทำงานทั้งหมด โดยจะส่งข้อมูลอนุกรมผ่านทางพอร์ต Rx และ Tx ไปยังอาร์เอฟโมดูล ใช้พอร์ต1.0-1.3 เป็นพอร์ตอินพุตจากเครื่องตรวจจับควัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

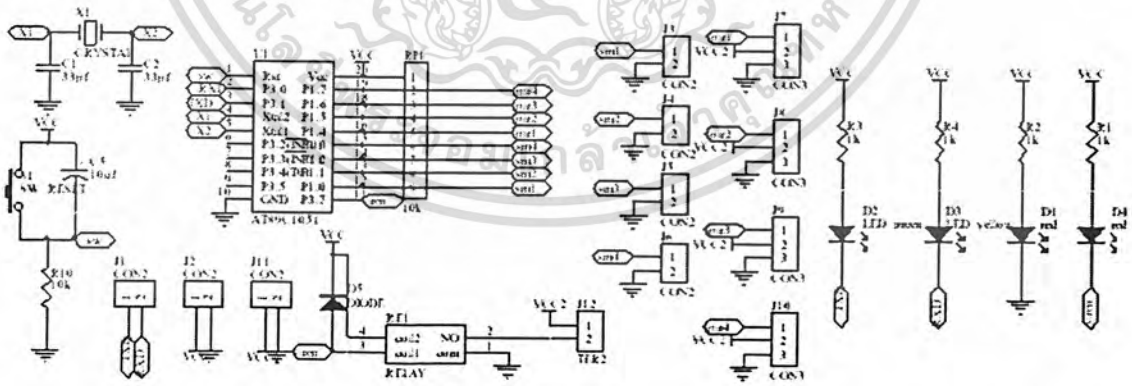
พอร์ต 1.4-1.7 เป็นพอร์ตอินพุตจากเครื่องดักจับความเคลื่อนไหว และ พอร์ต3.7 เป็นพอร์ตเอาท์พุต สำหรับควบคุมการทำงานของรีเลย์



รูปที่ 3-42 บอร์ดระบบรักษาความปลอดภัย

ส่วนเปิด-ปิดไซเรน

ใช้รีเลย์ในการควบคุมการเปิด-ปิดไซเรน ซึ่งการควบคุมรีเลย์นั้นจะมีไมโครคอนโทรลเลอร์เป็นตัวควบคุมอีกทีหนึ่งโดยจะจ่ายแรงดันไฟฟ้าให้ตกคร่อมที่ขดลวดภายในรีเลย์ ทำให้ขดลวดเกิดการเหนี่ยวนำสนามแม่เหล็กไฟฟ้า ซึ่งจะทำให้หน้าสัมผัสของรีเลย์ติดกัน นั่นก็คือการเปิดไซเรนนั่นเอง



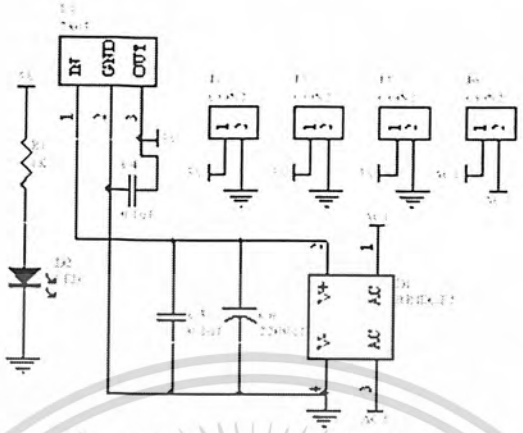
รูปที่ 3-43 แสดงภาพรวมทั้งหมดของวงจรรักษาความปลอดภัย

3.5.8 การออกแบบวงจรแหล่งจ่ายไฟ

จะทำหน้าที่แปลงไฟกระแสสลับให้เป็นไฟกระแสตรง เพื่อจ่ายไปยังบอร์ดต่างๆ โดยจะมีไดโอดบริดจ์ทำหน้าที่เรียงกระแส แล้วผ่านตัวเก็บประจุขนาด 2,200 uF เพื่อทำให้แรงดันเรียบขึ้น โดยจะมีตัวเก็บ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

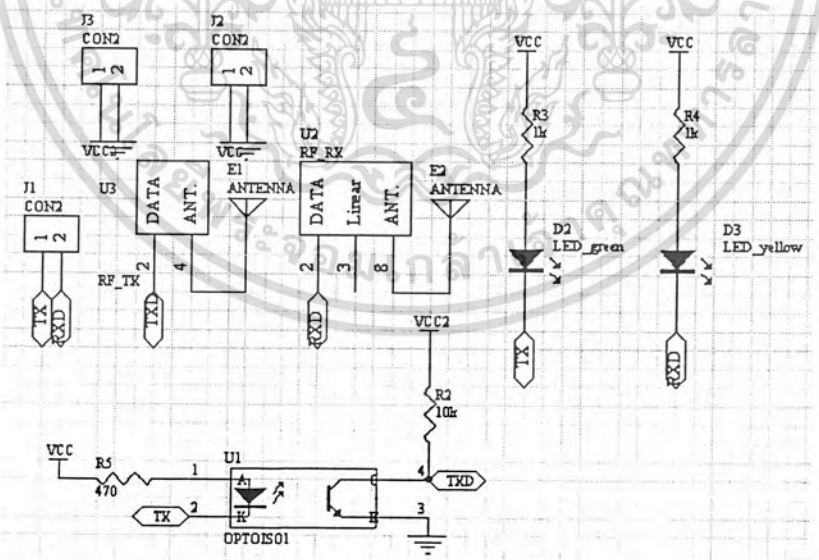
ประจุขนาดเล็กๆ 0.1 uF เพื่อกำจัดสัญญาณรบกวนทำให้แรงดันมีความคงที่มากขึ้น จากนั้นผ่านไอซี 7805 ซึ่งเป็นไอซีเร็กกูเลเตอร์ เพื่อให้ได้แรงดันเอาท์พุตเป็น 5 โวลต์



รูปที่ 3-44 แสดงวงจรเร็กกูเลตเตอร์ 5 โวลต์

3.5.9 การออกแบบวงจรรับ-ส่งคลื่นวิทยุ

ใช้โมดูลสำเร็จรูป TLP434A สำหรับการส่งข้อมูล และ RLP434A สำหรับการรับข้อมูล โดยจะมีไฟเลี้ยงวงจร 2 ชุด แต่ละชุดมีแรงดันไฟฟ้า 5 โวลต์ และมีออปโตคัปเปลอร์ทำหน้าที่แยกกราวน์ไฟฟ้า ที่ต้องแยกกราวด์ไฟฟ้าเพราะ ตัวรับและตัวส่ง ถ้าใช้กราวด์เดียวกันจะทำให้เกิดสัญญาณรบกวนอย่างมาก จนทำให้อาจไม่สามารถสื่อสารข้อมูลได้



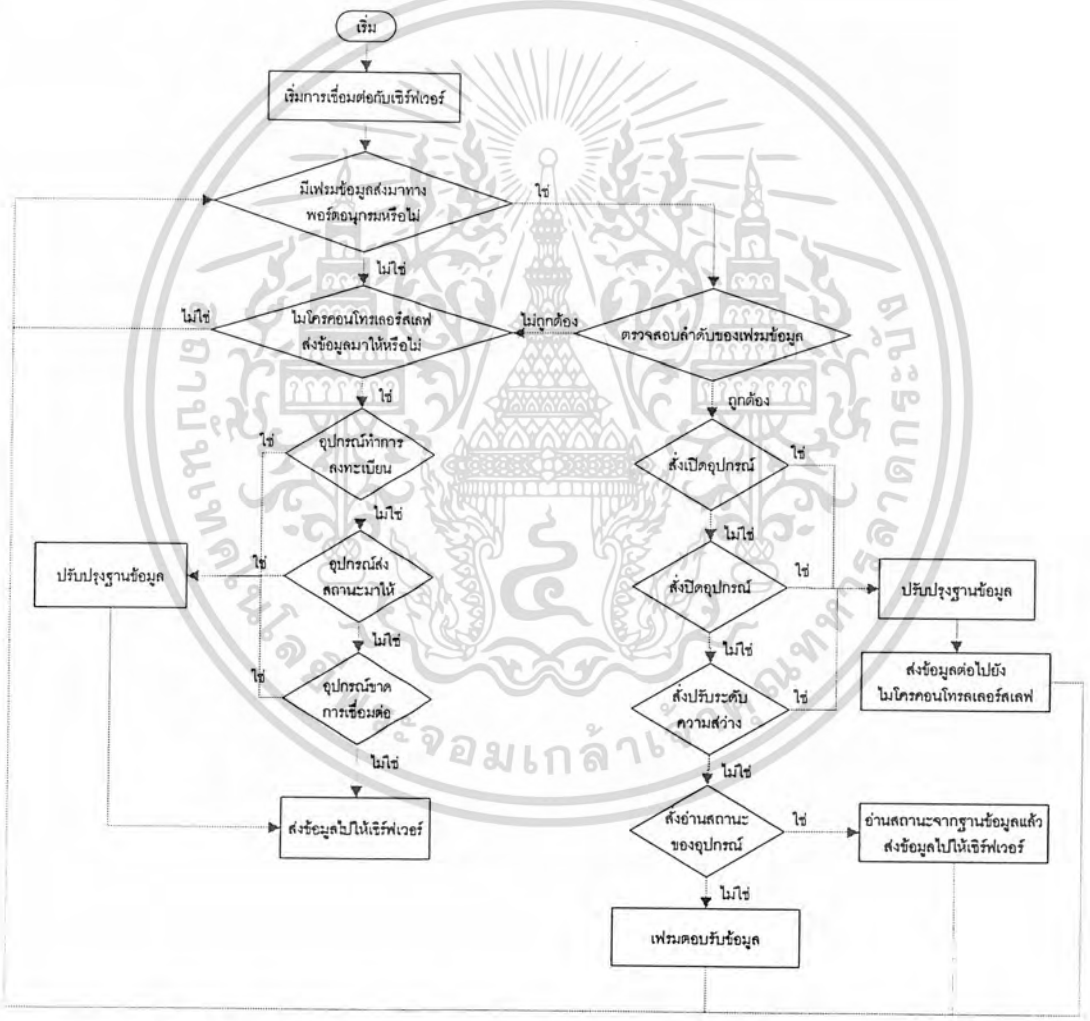
รูปที่ 3-45 แสดงวงจรรับ-ส่งคลื่นวิทยุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6 การออกแบบโปรแกรมควบคุมฮาร์ดแวร์

3.6.1 โปรแกรมควบคุมของไมโครคอนโทรลเลอร์มาสเตอร์

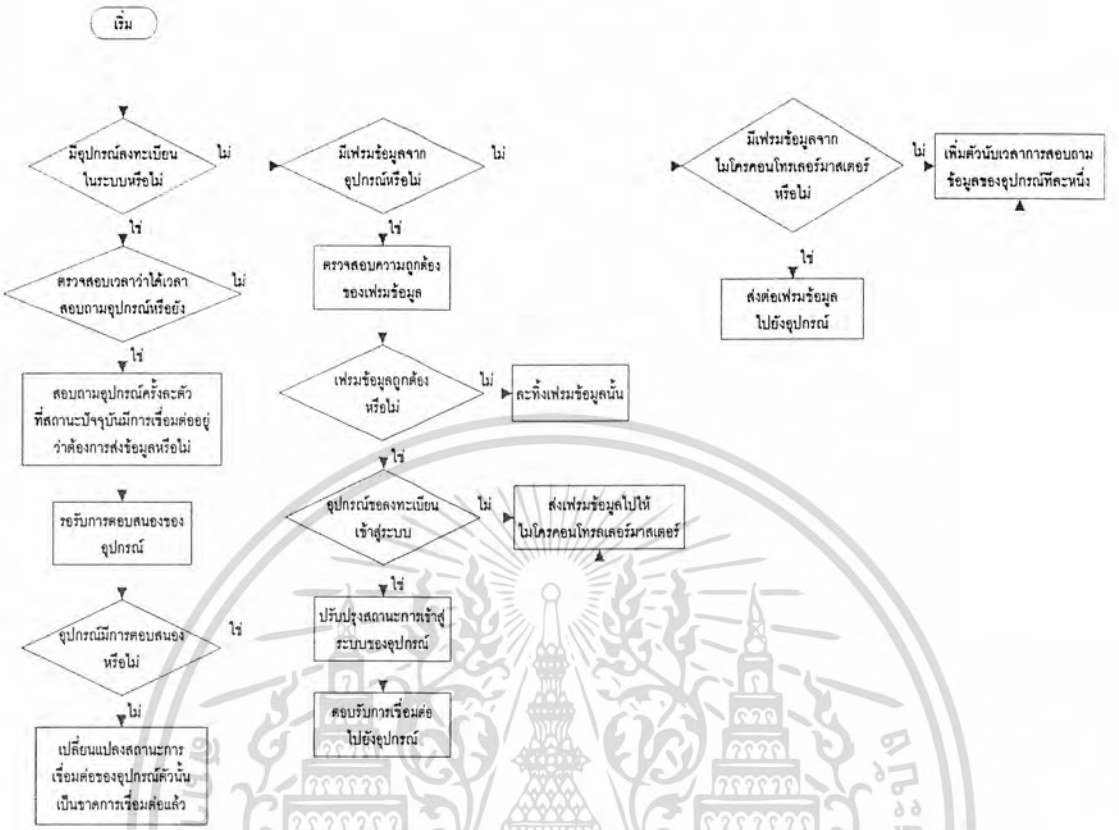
จากรูปที่ 3-46 เริ่มต้นการทำงานต้องทำการสร้างการเชื่อมต่อกับเซิร์ฟเวอร์ แล้วทำการตรวจสอบว่ามีเฟรมข้อมูลส่งมาจากเซิร์ฟเวอร์หรือไม่ ถ้ามีให้ทำการตรวจสอบลำดับการส่งของเฟรมข้อมูลว่าถูกต้องหรือไม่ หากลำดับเฟรมข้อมูลถูกต้อง จะทำการตรวจสอบคำสั่งในเฟรมข้อมูลว่าให้ทำอะไร เช่น สั่งเปิด-ปิดอุปกรณ์ สั่งให้ปรับความสว่าง ก็จะไปทำการปรับปรุงฐานข้อมูล แล้วส่งข้อมูลต่อไปยังไมโครคอนโทรลเลอร์สเลฟ ถ้าไม่มีการส่งเฟรมข้อมูลจากเซิร์ฟเวอร์ก็จะตรวจสอบว่ามีเฟรมข้อมูลจากไมโครคอนโทรลเลอร์สเลฟ ส่งมาหรือไม่ แล้วทำการตรวจสอบว่าให้ทำอะไร แล้วทำการปรับปรุงฐานข้อมูล ก่อนส่งต่อไปยังเซิร์ฟเวอร์



รูปที่ 3-46 โฟลว์ชาร์ตแสดงการทำงานของโปรแกรมไมโครคอนโทรลเลอร์มาสเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6.2 โปรแกรมควบคุมของไมโครคอนโทรลเลอร์สเลฟ



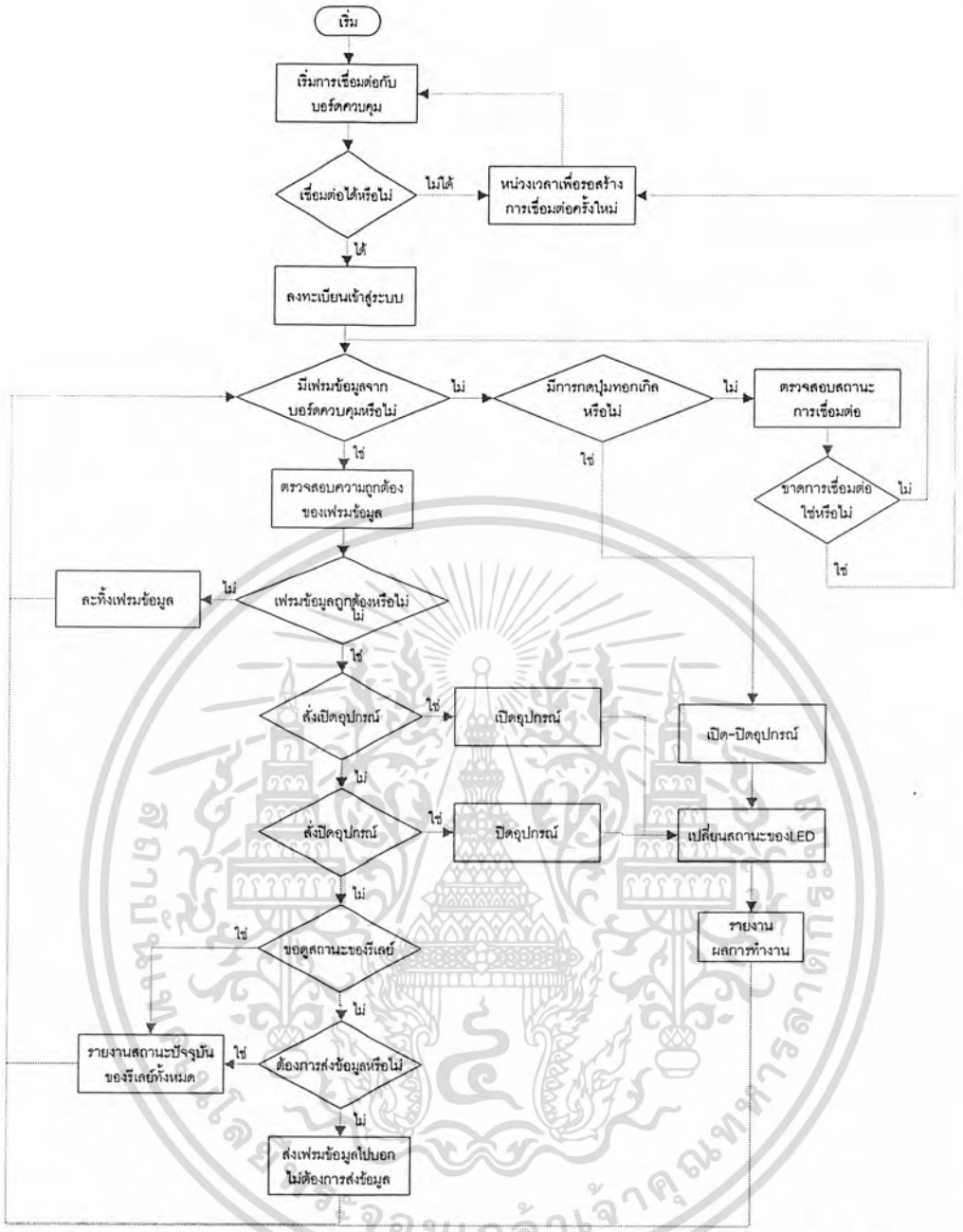
รูปที่ 3-47 แสดงโฟลว์ชาร์ตการทำงานของโปรแกรมไมโครคอนโทรลเลอร์สเลฟ

จากรูปที่ 3-47 เริ่มต้นการทำงานจะทำการถามอุปกรณ์ที่ได้ลงทะเบียนไว้และยังมีการเชื่อมต่ออยู่ โดยการถามนั้นจะถามครั้งละหนึ่งอุปกรณ์หมุนเวียนกันไปในลักษณะของราวด์โรบิน โดยจะถามว่าต้องการส่งข้อมูลหรือไม่ แล้วรอรับการตอบสนอง หากไม่มีการตอบสนองแสดงว่าอุปกรณ์นั้นได้ขาดการเชื่อมต่อแล้ว แล้วจะไม่วนมาถามอีก แต่ถ้ามีการตอบสนองแสดงว่าอุปกรณ์ยังไม่ขาดการเชื่อมต่อ ก็จะดูเฟรมข้อมูลที่อุปกรณ์ส่งมาให้ว่า ถ้าในเฟรมข้อมูลนั้นต้องทำโปรเซส เช่น อุปกรณ์ขอทำการลงทะเบียน ก็จะตอบรับการเชื่อมต่อกับอุปกรณ์แล้วไปทำการปรับปรุงสถานะของการเชื่อมต่อ แต่ถ้าเป็นเฟรมข้อมูลที่ไม่ต้องโปรเซสก็จะส่งต่อเฟรมข้อมูลนี้ไปให้ ไมโครคอนโทรลเลอร์มาสเตอร์

3.6.3 โปรแกรมควบคุมการเปิด-ปิดอุปกรณ์ไฟฟ้า

จากรูปที่ 3-48 เริ่มต้นการทำงานจะพยายามเชื่อมต่อกับบอร์ดควบคุมโดยการขอลงทะเบียนเข้าสู่ระบบ หลังจากนั้นก็จะคอยรับเฟรมข้อมูลจากบอร์ดควบคุมแล้วตรวจสอบว่าบอร์ดควบคุมสั่งให้ทำอะไร เช่น สั่งให้เปิดอุปกรณ์ไฟฟ้า , สั่งให้ปิดอุปกรณ์ไฟฟ้า หรือ สั่งให้ส่งสถานะการทำงานของรีเลย์ แต่ถ้าตัวเองต้องการจะส่งข้อมูล ในกรณีนี้ต้องรอคิวจากบอร์ดควบคุม โดยถ้าถึงคิวแล้วบอร์ดควบคุมจะมาถามเอง จากนั้นจึงจะส่งเฟรมข้อมูลได้ และตรวจสอบสถานะการกดปุ่มทอกเกิล ซึ่งเมื่อมีการกดทอกเกิลทุกครั้งโปรแกรมจะรอส่งข้อมูลเพื่อไปบอกสถานะการทำงานของรีเลย์ปัจจุบัน เพื่อให้ข้อมูลมีความทันสมัยที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



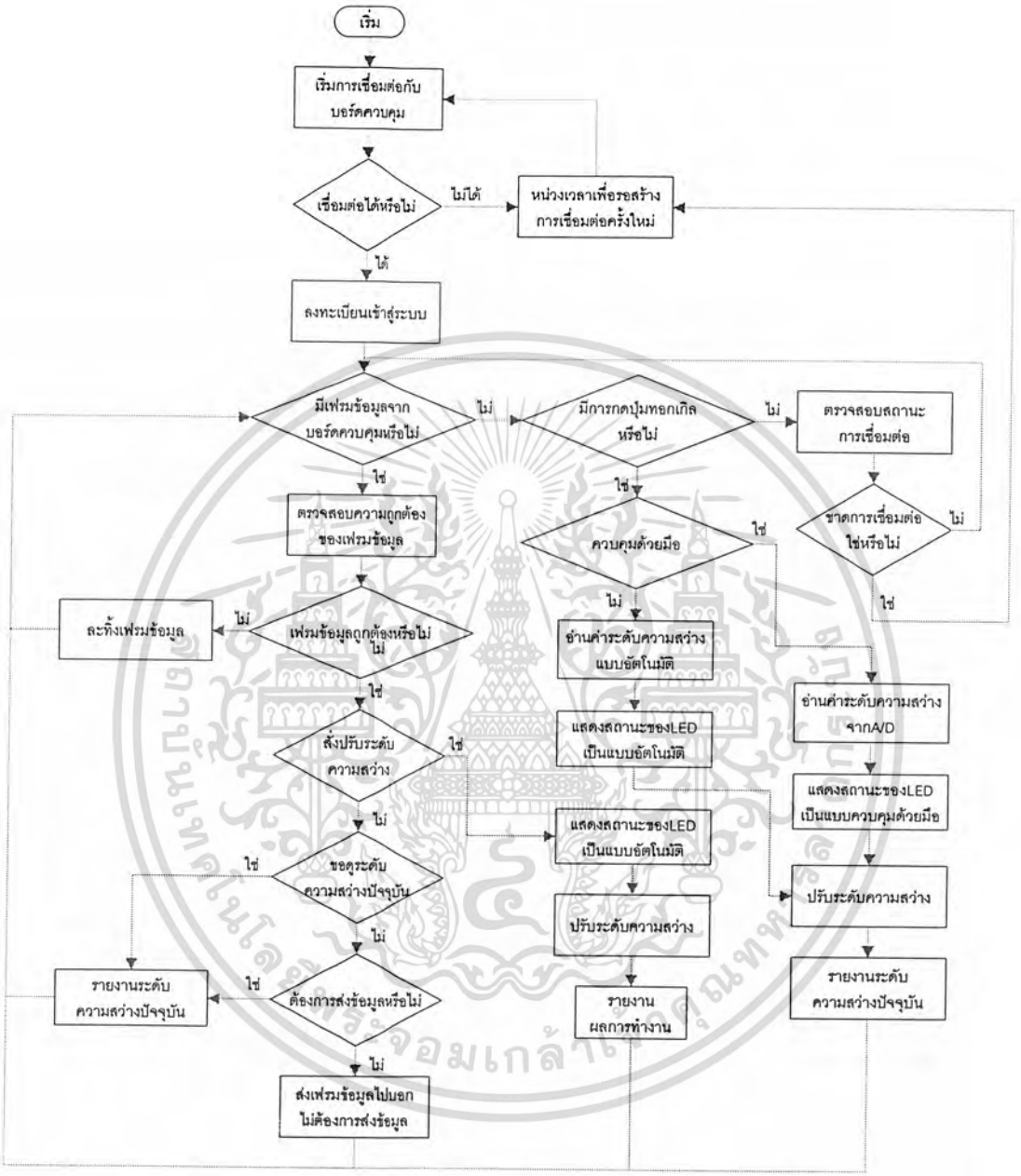
รูปที่ 3-48 แสดงโฟลว์ชาร์ตการทำงานของโปรแกรมควบคุมการเปิด-ปิดอุปกรณ์ไฟฟ้า

3.6.4 โปรแกรมควบคุมความสว่างของหลอดไฟ

จากรูปที่ 3-49 เริ่มต้นการทำงานจะพยายามเชื่อมต่อกับบอร์ดควบคุมโดยการขอลงทะเบียนเข้าสู่ระบบ หลังจากนั้นก็จะคอยรับเฟรมข้อมูลจากบอร์ดควบคุมแล้วตรวจสอบว่าบอร์ดควบคุมสั่งให้ทำอะไร เช่น สั่งให้ปรับระดับความสว่าง หรือสั่งให้รายงานระดับความสว่างปัจจุบัน แต่ถ้าตัวเองต้องการจะส่งข้อมูล ในกรณีนี้ต้องรอคิวจากบอร์ดควบคุม โดยถ้าถึงคิวแล้วบอร์ดควบคุมจะมาถามเอง จากนั้นจึงจะส่งเฟรมข้อมูลได้ และยังตรวจจับสถานะทอกเกิล โดยถ้าอยู่ในรูปแบบควบคุมด้วยมือจะไปทำการอ่านค่าระดับความสว่างจากA/D แล้วปรับความสว่าง หรือถ้าอยู่ในรูปแบบทำงานแบบอัตโนมัติจะอ่านค่าระดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความสว่างที่ได้เก็บไว้ในบัฟเฟอร์มาทำการปรับระดับความสว่าง ซึ่งเมื่อมีการกดทอกเกิลทุกครั้ง โปรแกรมจะรอส่งข้อมูลเพื่อไปบอกระดับความสว่างปัจจุบัน เพื่อให้ข้อมูลมีความทันสมัยที่สุด



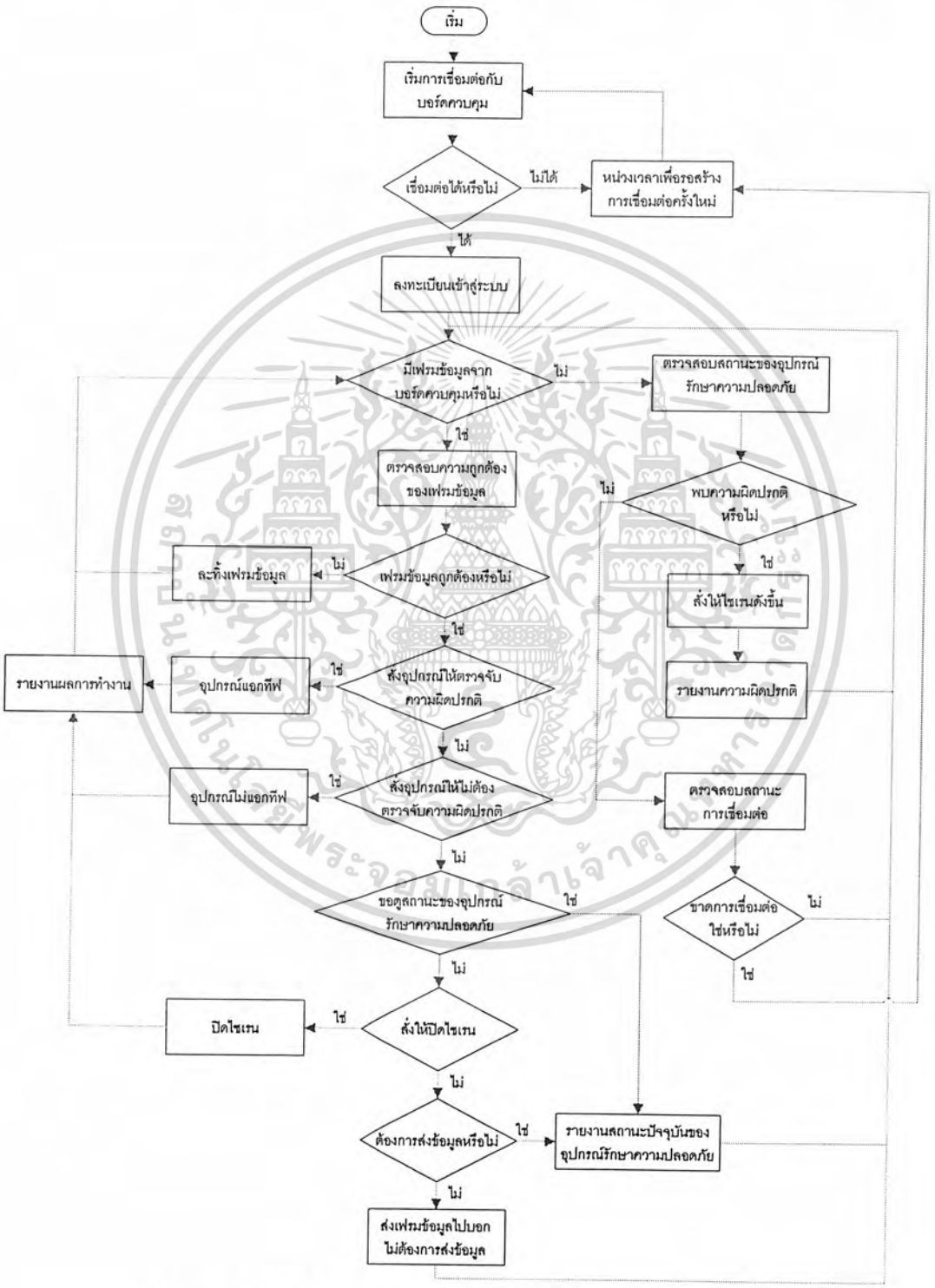
รูปที่ 3-49 แสดงโฟลว์ชาร์ตการทำงานของโปรแกรมควบคุมความสว่างของหลอดไฟ

3.6.5 โปรแกรมควบคุมระบบรักษาความปลอดภัย

จากรูปที่ 3-50 เริ่มต้นการทำงานจะพยายามเชื่อมต่อกับบอร์ดควบคุมโดยการขอลงทะเบียนเข้าสู่ระบบ หลังจากนั้นก็จะคอยรับเฟรมข้อมูลจากบอร์ดควบคุมแล้วตรวจสอบว่าบอร์ดควบคุมสั่งให้ทำอะไร เช่น สั่งให้ปรับระดับความสว่าง หรือสั่งให้รายงานระดับความสว่างปัจจุบัน แต่ถ้าตัวเองต้องการจะส่งข้อมูล ในกรณีนี้ต้องรอคิวจากบอร์ดควบคุม โดยถ้าถึงคิวแล้วบอร์ดควบคุมจะมาถามเอง จากนั้นจึงจะส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เฟรมข้อมูลได้ และยังตรวจจับสถานะทอกเกิล โดยถ้าอยู่ในรูปแบบควบคุมด้วยมือจะไปทำการอ่านค่าระดับความสว่างจากA/D แล้วปรับความสว่าง หรือถ้าอยู่ในรูปแบบทำงานแบบอัตโนมัติจะอ่านค่าระดับความสว่างที่ได้เก็บไว้ในบัฟเฟอร์มาทำการปรับระดับความสว่าง ซึ่งเมื่อมีการกดทอกเกิลทุกครั้ง โปรแกรมจะรอส่งข้อมูลเพื่อไปบอกระดับความสว่างปัจจุบัน เพื่อให้ข้อมูลมีความทันสมัยที่สุด



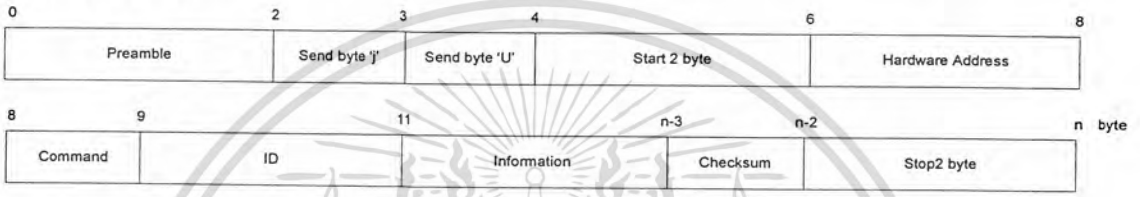
รูปที่ 3-50 แสดงโฟลว์ชาร์ตการทำงานของโปรแกรมระบบรักษาความปลอดภัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการตรวจสอบการทำงานของอุปกรณ์รักษาความปลอดภัยแต่ละตัว หากอุปกรณ์รักษาความปลอดภัยใดตรวจพบความผิดปรกติ ก็จะสั่งให้ไซเรนทำงานแล้วรอจนถึงคิวตนเองส่งข้อมูลจึงจะรายงานเหตุการณ์ผิดปรกติที่เกิดขึ้นให้บอร์ดควบคุมทราบ และการปิดไซเรนนั้นจะต้องรอบอร์ดควบคุมสั่งให้ปิดเท่านั้น

3.7 RF Protocol

การสื่อสารระหว่างบอร์ดโดยใช้ RF นั้น เป็นโปรโตคอลที่ดัดแปลงจากโปรโตคอลที่ใช้ในการสื่อสารระหว่างเซิร์ฟเวอร์กับบอร์ดควบคุม ให้เหมาะสมกับการใช้งาน RF โดยจะส่งในรูปแบบของบิตสตรีมและบิตสตรีมนี้เป็นข้อมูลแบบเดียวกับข้อมูลอนุกรม สำหรับรูปแบบเฟรมการส่งข้อมูลนั้นได้แสดงไว้ในตารางข้างล่าง



รูปที่ 3-51 แสดงรูปแบบของ RF Protocol

รายละเอียดของแต่ละคำสั่ง

คำสั่ง	100 = ON เปิด
Info:	(channel) Ex('1')('2') คือสั่งเปิดอุปกรณ์ช่องที่ 1 กับ 2
คำสั่ง	101 = OFF ปิด
Info:	(channel) Ex('1')('3') คือสั่งปิดอุปกรณ์ช่องที่ 1 กับ 3
คำสั่ง	102 = Dim ปรับระดับความสว่าง
Info:	(channel,dim_level) Ex : ('1','5')('2', '0') ปรับความสว่างของหลอดไฟ ช่องที่ 1 ที่ระดับ 5 กับ ช่องที่ 2 ที่ระดับ 0
คำสั่ง	200 = Read สั่งให้อ่านค่าอุปกรณ์
Info :	(model) '0' = Switch Board '1' = Dimmer Board '3' = Security Board
คำสั่ง	201 = information ส่งรายละเอียดสถานะของอุปกรณ์
Info:	สำหรับ Switch Board (HW Addr,Off,ch1,ch2,ch3,ch4) Ex : ('51',101,'0','1','0','0') คือ มีช่องที่ 2 เปิดอยู่ ช่อง 1,3และ4 ปิดอยู่ สำหรับ Dimmer Board (HW Addr,Dim,ch)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	Ex: ('73',102,'4') คือ ช่องที่ 1 ระดับความสว่างระดับ 4 สำหรับ Security Board (mac,pir1pir2,pir3,pir4,sm1,sm2,sm3,sm4,alert) เหตุการณ์ปกติ=0 ไม่ปกติ=1 Ex: ('67','0','0','0','0','0','0','0','0','0','0') คือ เซ็นเซอร์ทุกตัวไม่พบเหตุการณ์ผิดปรกติ และไซเรนไม่ดัง (เหตุการณ์ปรกติ คือ '0' เหตุการณ์ไม่ปรกติ คือ '1')
คำสั่ง	202 = Device Register ทำการลงทะเบียนเพื่อขอการเริ่มการเชื่อมต่อ
Info:	(model,channels_no) Ex: ('0','4') คือ ประเภท Switch Board จำนวน 4 ช่อง
คำสั่ง	203 = Result (บอร์ดควบคุม จะเป็นคนตอบกลับไปหาserver ว่าทำคำสั่งที่serverให้ทำเรียบร้อยแล้ว)
Info:	ใช้กับคำสั่ง On , Off , Dim , Device Register and Ask Alive '1' = ok '0' = false or error
คำสั่ง	204 = Error
Info:	รหัสความผิดพลาด '0' = not define '1' = Device not available กรณีมีอยู่ แต่ไม่สามารถติดต่อกับอุปกรณ์ได้
คำสั่ง	205 = Ask Alive
Info:	ไม่มี

ตารางที่ 3-11 แสดงรายละเอียดคำสั่งของ RF Protocol

เทคนิคของการส่งข้อมูลชุด RF นี้ ในขั้นแรกจำเป็นจะต้องมีการส่งรหัสนำหน้าหรือที่เรียกว่า Preamble ในลักษณะ 0101010101010101 ออกไปก่อนประมาณ2ไบต์ และหลังจากส่ง Preamble แล้วให้ส่ง Code Check โดยใช้ไบต์ข้อมูล 'j' และ 'U' เพื่อให้ภาครับทำการตรวจสอบ เมื่อสามารถตรวจสอบไบต์ข้อมูล 'j' และ 'U' ได้แล้วก็จะรับประกันได้ว่า สามารถส่งข้อมูลได้โดยไม่มีปัญหา จากนั้นก็ส่งStart byte , Hardware Address ,Command ,ID , Information , Checksum และ Stop Byte ตามลำดับ

สำหรับในส่วนเทคนิคการรับข้อมูลในภาครับนั้น ไม่ต้องทำการตรวจสอบการรับ Preamble ให้คอยตรวจสอบการรับค่าไบต์ข้อมูล 'j' และ ไบต์ข้อมูล'U' หากสามารถรับค่าตามนี้ได้ ไบต์ข้อมูลถัดมาจะเป็นค่าของไบต์ข้อมูลจริงที่ต้องจากส่งจากภาคส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดลองและทดสอบการทำงาน

4.1 เงื่อนไขและสภาวะในการทดลอง

อุปกรณ์ที่ใช้ในการทดสอบ

1. เครื่องคอมพิวเตอร์
   - ระบบปฏิบัติการ Windows XP Service Pack 2
   - หน่วยความจำ 256 MB
   - ซีพียู 1.54 GHz
   - เนื้อที่ฮาร์ดดิสค์ 40 GB
2. โทรศัพท์มือถือ
   - Nokia 6600
   - มีอุปกรณ์ Bluetooth
   - GPRS
   - MIDP 2.0
3. บอร์ดควบคุม
4. Interface Module
5. เครื่องใช้ไฟฟ้า

เงื่อนไขและสภาวะในการทดลอง

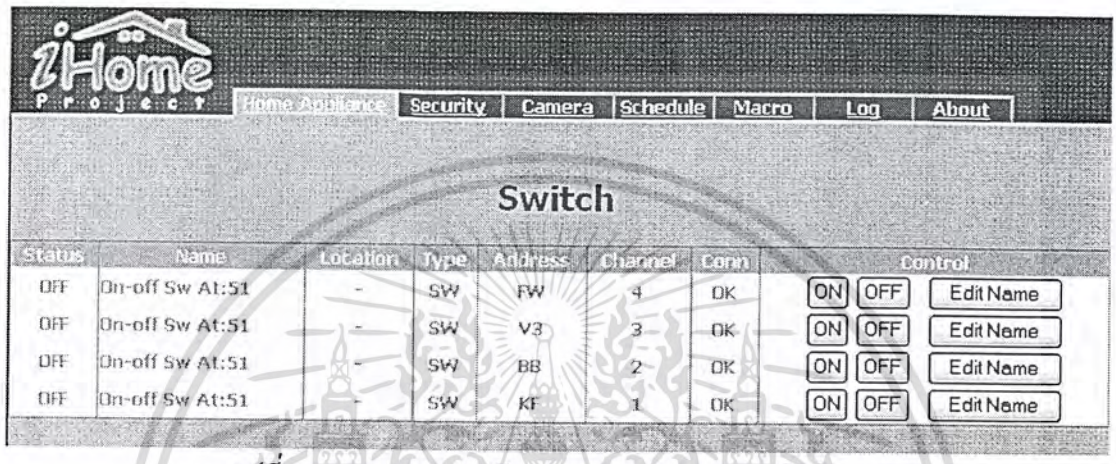
- สื่อสาร โดยใช้สายเชื่อมระหว่าง Interface Module แต่ละตัว
- เครื่องคอมพิวเตอร์ที่สมาร์ทโฮมเซิร์ฟเวอร์ทำงานไม่มีการทำงานอย่างอื่นควบคู่ที่อาจจะส่งผลรบกวนการทำงานของโปรแกรมได้ เช่น โปรแกรมที่ใช้กำลังซีพียูหรือหน่วยความจำอย่างมาก โปรแกรมที่ติดต่อใช้งานพอร์ตอนุกรม เป็นต้น
- เครื่องสมาร์ทโฮมเซิร์ฟเวอร์มีการกำหนดค่าเกี่ยวกับการเชื่อมต่อเครือข่ายอย่างถูกต้อง เช่น อินเทอร์เน็ต (Internet) บูลทูธ (Bluetooth) เป็นต้น
- อุปกรณ์ไฟฟ้าที่ใช้กับ Interface Module ไม่เกินมาตรฐานที่ Interface Module รองรับได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การทดลองการเพิ่ม Interface Module เข้าสู่ระบบ

เมื่อ Interface Module เริ่มการทำงาน Interface Module จะร้องขอเพื่อทำการลงทะเบียนเข้าระบบสมาร์ทเซิร์ฟเวอร์จะตรวจสอบประเภทของอุปกรณ์ จำนวนแชนแนล ถ้าเป็นอุปกรณ์ที่รู้จักก็จะทำการลงทะเบียนอุปกรณ์นั้นๆ เข้าสู่ระบบ เพื่อให้ผู้ใช้สามารถใช้งานได้

จากรูปเมื่อมีชุด Switch Module เข้ามาใหม่ ระบบจะทำการลงทะเบียนอุปกรณ์ต่างๆ แล้วตรวจสอบจำนวนแชนแนล พร้อมทั้งกำหนดชื่ออุปกรณ์ให้อัตโนมัติ

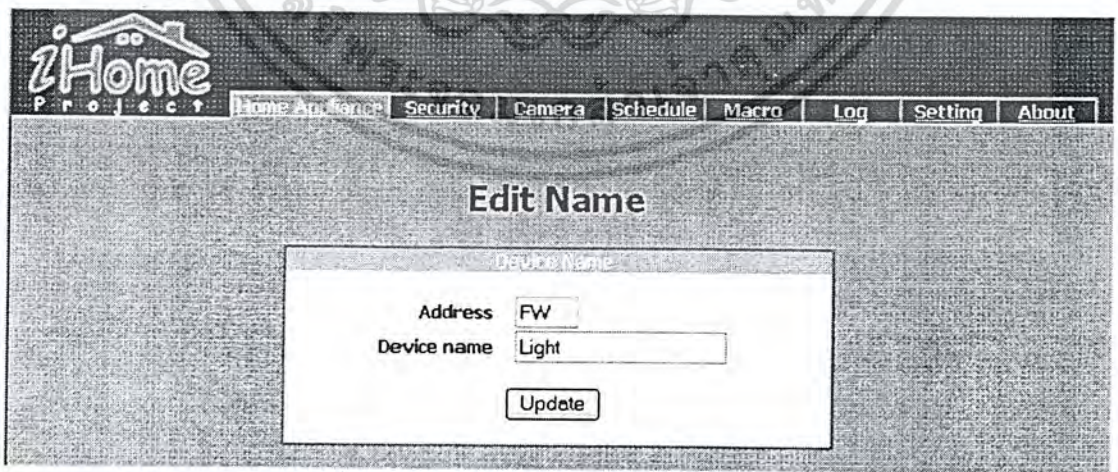


รูปที่ 4-1 แสดงการลงทะเบียนของ Switch Module เข้าสู่ระบบ

4.3 การทดลองการกำหนดค่าให้กับ Interface Module

หลังจากการลงทะเบียน ผู้ใช้จะเห็นว่ามี Interface Module ใหม่เพิ่มเข้ามาจากหน้าจอควบคุม ซึ่งจะถูกกำหนดค่าเริ่มต้นเป็นค่ามาตรฐาน ทั้งนี้สามารถกำหนดใหม่ภายหลังได้

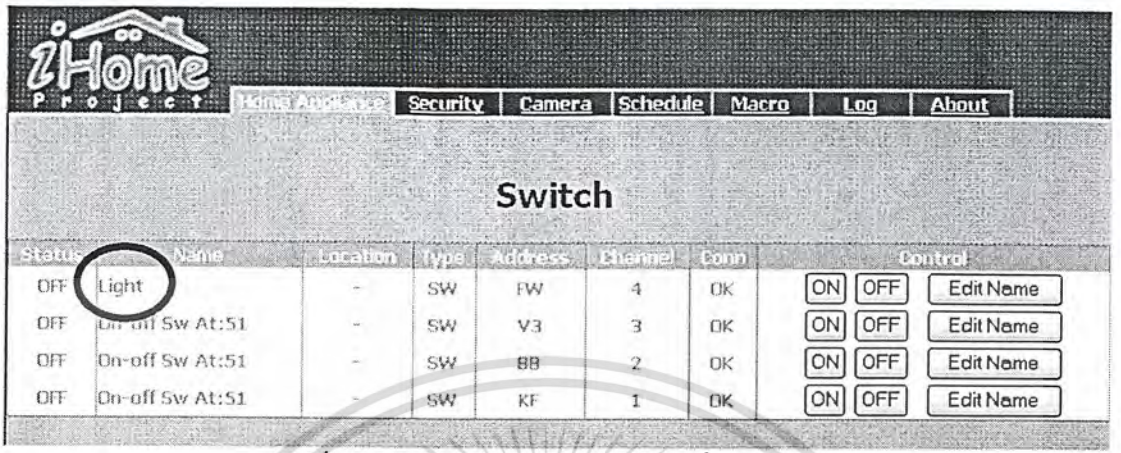
จากรูปเป็นการกำหนดชื่อให้กับอุปกรณ์ที่แอดแดรส FW ให้มีชื่อว่า “Light”



รูปที่ 4-2 แสดงหน้าจอในการกำหนดชื่อให้อุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

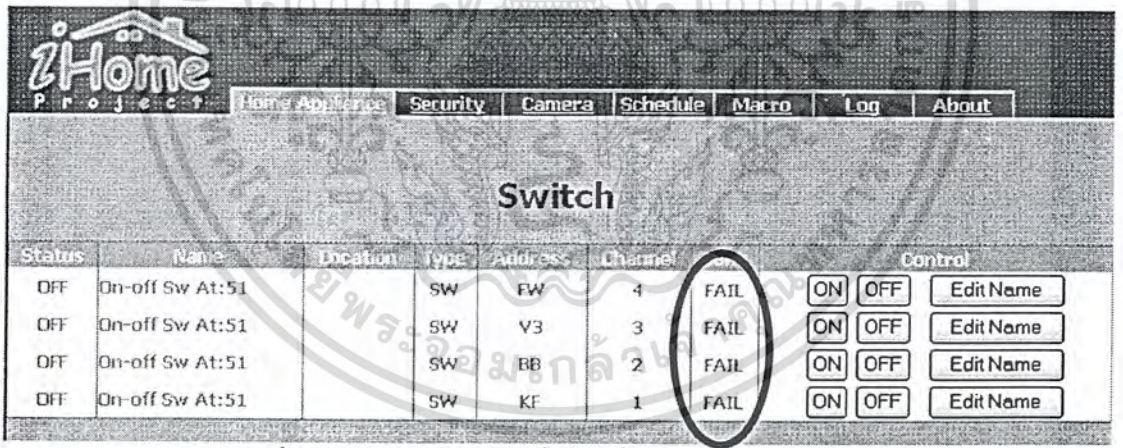
เมื่อกดปุ่ม Update ระบบจะทำการปรับปรุงชื่อที่แอดแดรส FW ให้ตามชื่อที่ได้ตั้งไว้



รูปที่ 4-3 แสดงผลลัพธ์หลังจากกำหนดชื่อให้อุปกรณ์

4.4 การทดลองการนำ Interface Module ออกจากระบบ

อุปกรณ์ที่เชื่อมต่อเข้ามาและพร้อมใช้งาน ในฟิลด์ Conn ต้องเท่ากับ OK ถ้าเท่ากับ Fail จะหมายถึงอุปกรณ์ขาดการเชื่อมต่อ ซึ่งผู้ใช้จะไม่สามารถควบคุมอุปกรณ์นั้นๆ ได้



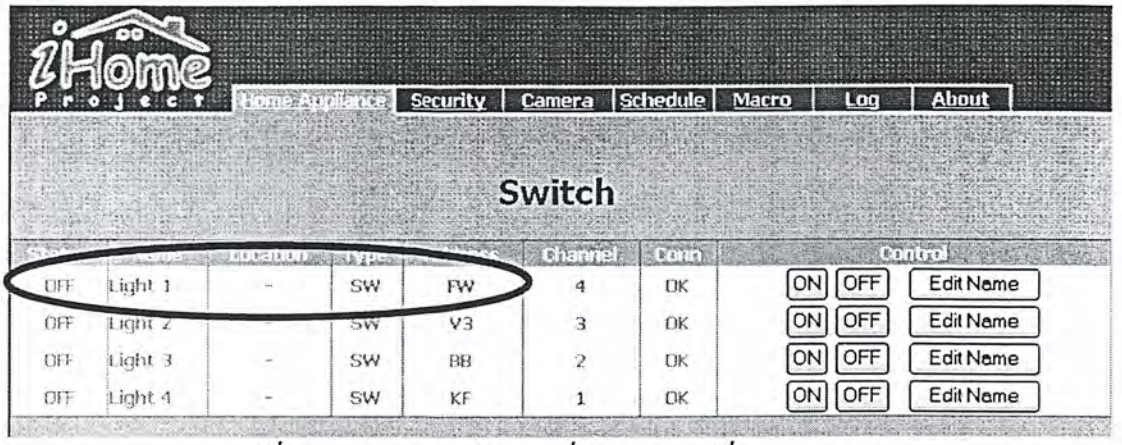
รูปที่ 4-4 แสดงภาพอุปกรณ์ที่ Fail ซึ่งจะไม่สามารถใช้งานได้

4.5 การควบคุมการทำงานของ Switch Module ผ่านเว็บบราวเซอร์

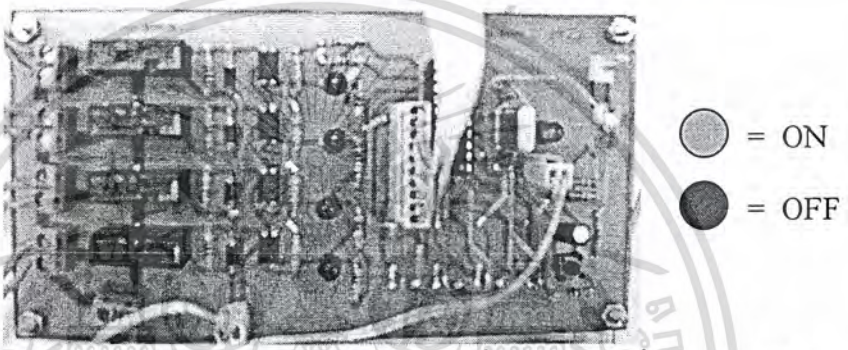
การควบคุมของ Switch Module มี 2 ฟังก์ชันคือการเปิดและการปิด การเปิดคือการจ่ายกระแสไฟฟ้า เครื่องใช้ไฟฟ้าที่ต่ออยู่กับแชนแนลที่ทำการเปิดก็จะสามารถทำงานได้ ในทางตรงข้าม การปิดจะเป็นการหยุดจ่ายกระแสไฟฟ้า เครื่องใช้ไฟฟ้าก็จะหยุดการทำงาน ทั้งนี้อุปกรณ์ที่นำมาต่อกับ Switch Module จึงควรเป็นอุปกรณ์ที่ไม่เสียหายจากการเกิดกระแสไฟฟ้าดับกะทันหัน เช่น โคมไฟ วิทยุ เป็นต้น

จากรูปจะเป็นการสั่งเปิดอุปกรณ์ที่แอดแดรส FW

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

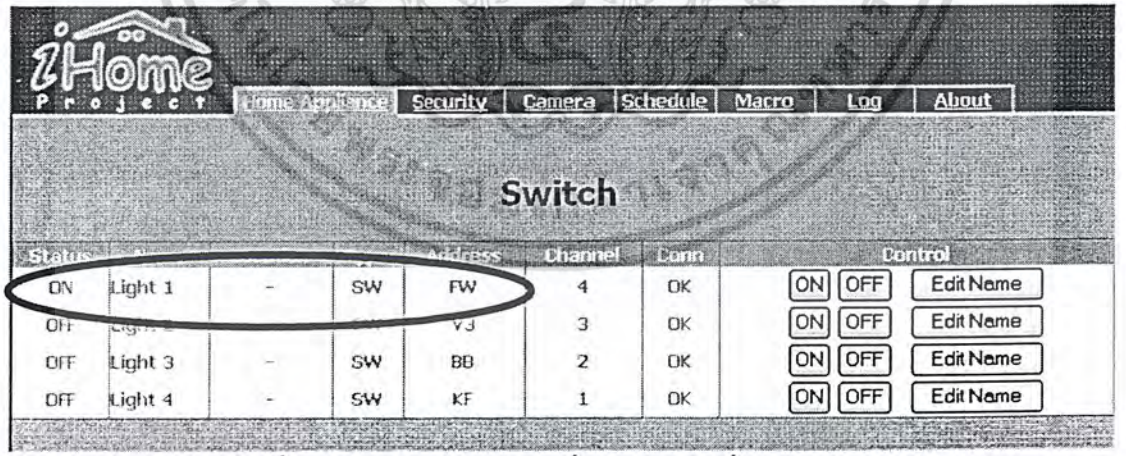


รูปที่ 4-5 แสดงสถานะก่อนการสั่งเปิดอุปกรณ์ที่แอดแดรส FW



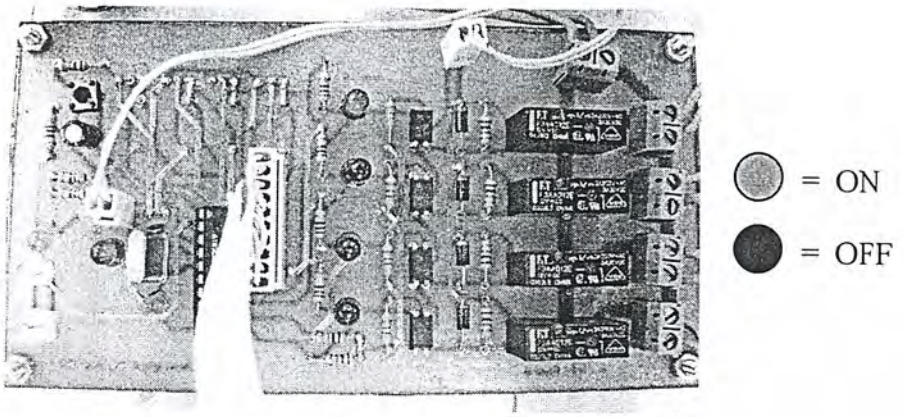
รูปที่ 4-6 แสดงไฟสถานะรีเลย์ของ Switch Module ก่อนการสั่งเปิดอุปกรณ์ที่แอดแดรส *FW*

หลังจากกดปุ่ม ON ค่าสถานะของอุปกรณ์ที่แอดแดรส FW จะเปลี่ยนเป็นจาก OFF เป็น ON



รูปที่ 4-7 แสดงสถานะหลังการสั่งเปิดอุปกรณ์ที่แอดแดรส FW

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

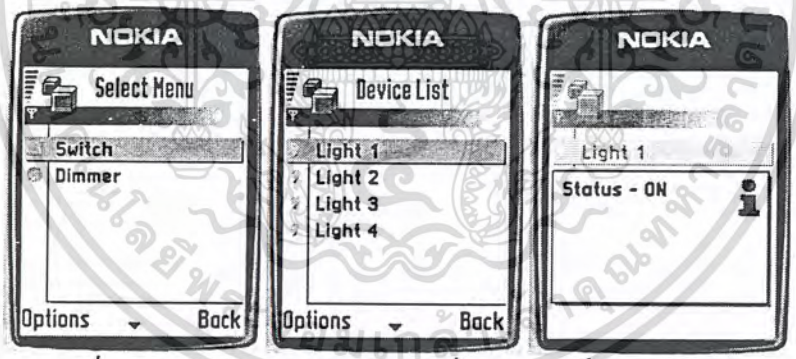


รูปที่ 4-8 แสดงไฟสถานะรีเลย์ของ Switch Module หลังการสั่งเปิดอุปกรณ์ที่แอดแดรส FW

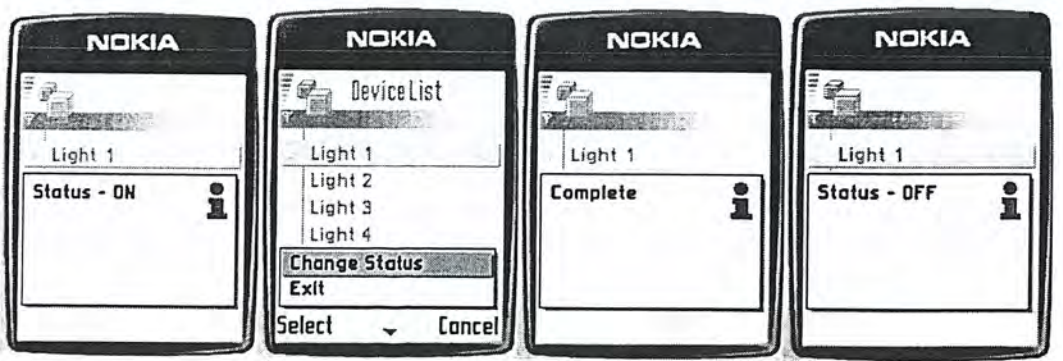
4.6 การควบคุมการทำงานของ Switch Module ผ่านโทรศัพท์มือถือ

เมื่อต้องการเปลี่ยนสถานะการทำงานของอุปกรณ์ให้เลือก Change Status จากนั้นกด Select แล้วรอผลลัพธ์ตอบกลับมา ถ้า Complete แสดงว่าอุปกรณ์มีการเปลี่ยนแปลงสถานะเรียบร้อยแล้ว แต่ถ้า Fail แสดงว่าอุปกรณ์ยังไม่มีการเปลี่ยนสถานะ ซึ่งอาจเกิดจากความผิดพลาดบางอย่าง โดยสามารถทราบสถานะของอุปกรณ์ได้จากการกดปุ่ม Select ที่อุปกรณ์หรือสังเกตจากสีของรูปภาพหลอดไฟด้านหน้า

จากรูปเมื่อกด Change Status เพื่อเปลี่ยนสถานะของ Light 1 จาก เปิดเป็นปิด จากนั้นมีการตอบ Complete กลับมา แสดงว่าอุปกรณ์ได้ทำการปิดเรียบร้อยแล้ว แล้วจึงปรับปรุงการแสดงผลที่หน้าจอให้ Light 1 มีสถานะเป็น OFF



รูปที่ 4-9 รูปแสดงสถานะของอุปกรณ์ชื่อ Light 1 เมื่อกดปุ่ม Select

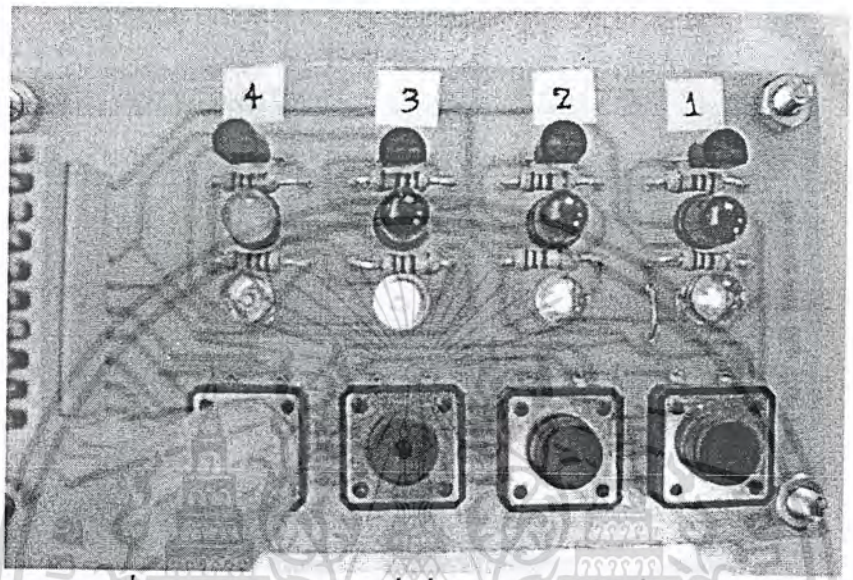


รูปที่ 4-10 แสดงการเปลี่ยนสถานะของอุปกรณ์ชื่อ Light 1

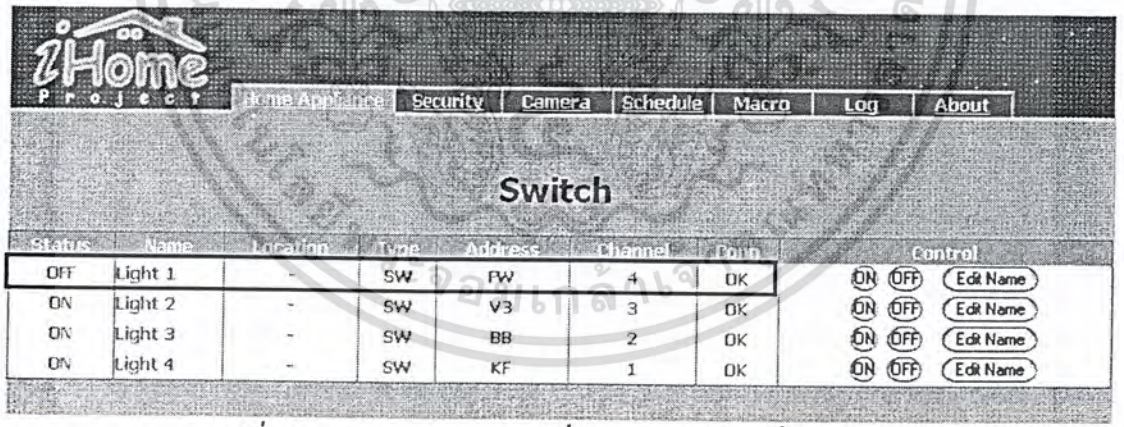
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.7 การควบคุมการทำงานของ Switch Module แบบ Manual

Switch Module คืออุปกรณ์ที่ช่วยควบคุมการเปิดปิดของเครื่องใช้ไฟฟ้า ซึ่งในปริญญานิพนธ์นี้ออกแบบให้สามารถควบคุมเครื่องใช้ไฟฟ้าได้ 4 แชนแนล ในการใช้งานแบบ Manual นั้น เมื่อผู้ใช้มีการกดปุ่มเพื่อเปลี่ยนสถานะ เช่น จากปิดเป็นเปิด หรือเปิดเป็นปิด จะมีการรายงานสถานะไปที่เซิร์ฟเวอร์ โดยผู้ใช้สามารถตรวจสอบสถานะได้จากบนเว็บหรือบนโทรศัพท์มือถือ



รูปที่ 4-11 แสดงการกดปุ่มเพื่อสั่งปิดอุปกรณ์ไฟฟ้าที่แชนแนล 4



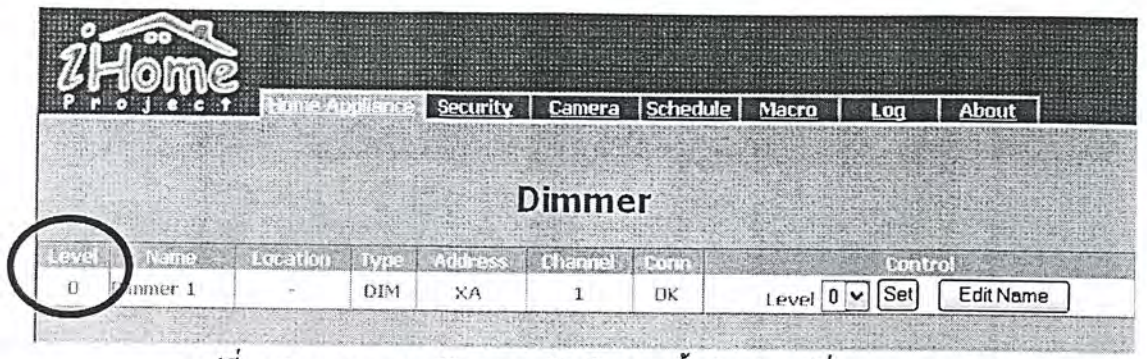
รูปที่ 4-12 แสดงสถานะหลังการสั่งปิดอุปกรณ์ไฟฟ้าที่แชนแนล 4

4.8 การควบคุมการทำงานของ Dimmer Module ผ่านเว็บบราวเซอร์

การควบคุมของ Dimmer Module มี 1 ฟังก์ชันคือการควบคุมระดับความสว่าง ซึ่งแบ่งเป็น 8 ระดับ คือ 0 - 7 ระดับ 0 คือ ปิด ระดับ 7 คือเปิดให้สว่างสุด

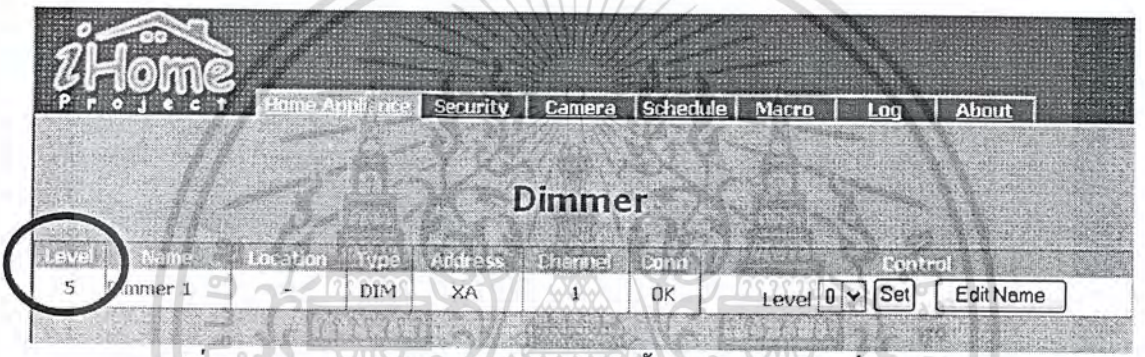
จากรูประดับความสว่างของอุปกรณ์ที่แอดแดรส XA ตอนนี้อยู่ที่ระดับ 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4-13 แสดงการระดับความสว่างก่อนการตั้งค่าอุปกรณ์ที่แอดแดรส XA

เมื่อทำการปรับระดับความสว่างไปที่ 5 แล้วกดปุ่ม Set ระบบจะทำการเปลี่ยนระดับความสว่างจากระดับ 0 เป็นระดับ 5 แล้วแสดงผลกลับมาที่หน้าจอ

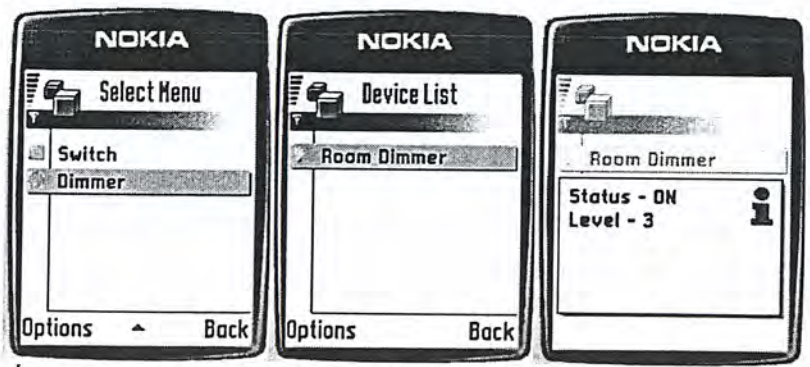


รูปที่ 4-14 แสดงการระดับความสว่างหลังการตั้งระดับความสว่างที่แอดแดรส XA

4.9 การควบคุมการทำงานของ Dimmer Module ผ่านโทรศัพท์มือถือ

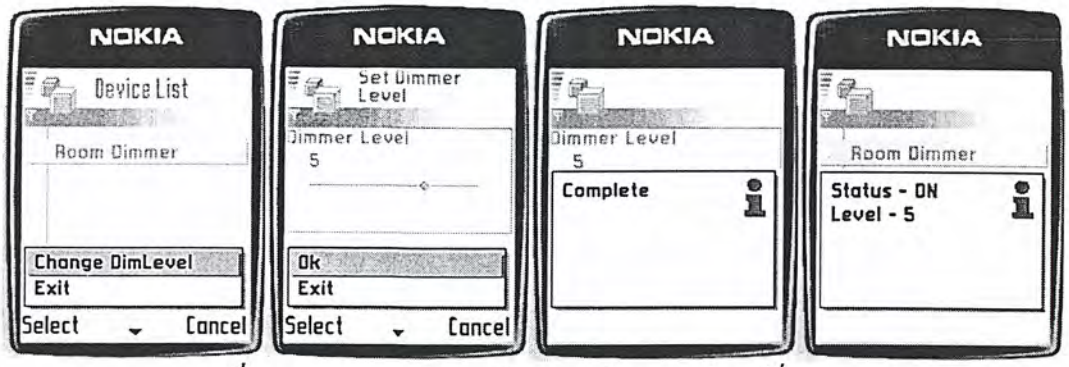
เมื่อต้องการเปลี่ยนระดับความสว่างของอุปกรณ์ ให้เลือกไปที่ Change DimLevel จากนั้นทำการกำหนดค่าระดับความสว่าง แล้วกด Ok รอการตอบกลับ โดยถ้าตอบ Complete กลับมาแสดงว่ามีการปรับระดับความสว่างเรียบร้อยแล้ว แต่ถ้าตอบ Fail แสดงว่าเกิดความผิดพลาดเกิดขึ้น

จากรูปจะเป็นการปรับระดับความสว่างของอุปกรณ์ชื่อ Room Dimmer ซึ่งมีความสว่างอยู่ที่ระดับ 3 ให้มีความสว่างที่ระดับ 5 เมื่อกด Ok แล้ว ได้รับการตอบ Complete กลับมา แสดงว่าได้ปรับระดับความสว่างเป็นระดับ 5 เรียบร้อยแล้ว



รูปที่ 4-15 แสดงระดับความสว่างของอุปกรณ์ชื่อ Room Dimmer เมื่อกดปุ่ม Select

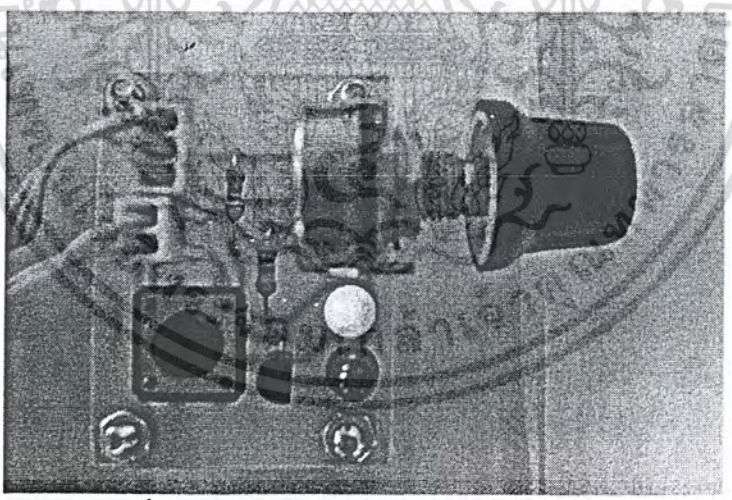
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4-16 แสดงการปรับระดับความสว่างของอุปกรณ์ชื่อ Room Dimmer

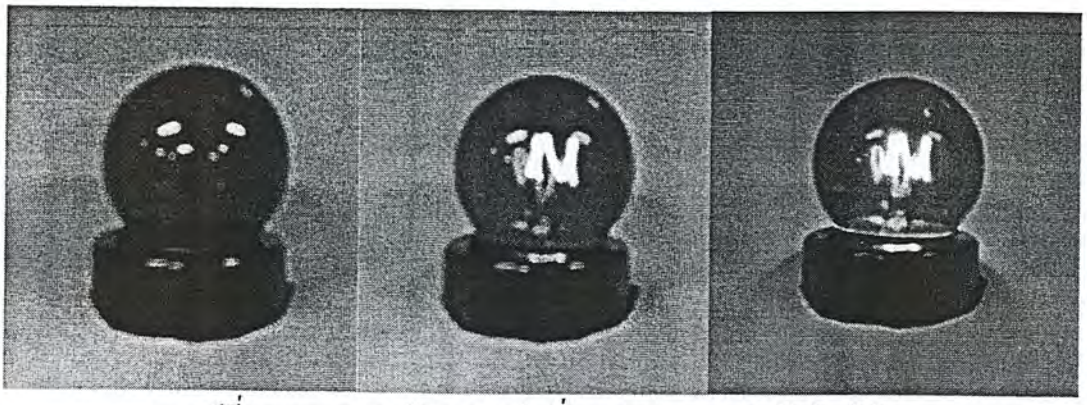
4.10 การทดลองปรับระดับความสว่างของ Dimmer Module แบบ Manual

Dimmer Module คืออุปกรณ์ที่ช่วยควบคุมความสว่างของหลอดไฟ ซึ่งในปริญญานิพนธ์นี้ออกแบบให้สามารถควบคุมได้ 1 แชนแนล ในการใช้งานแบบ Manual นั้น ผู้ใช้ต้องกดปุ่ม Toggle เพื่อเปลี่ยนสถานะให้อยู่ในโหมด Manual แล้วทำการหมุนปุ่มปรับระดับความสว่าง ซึ่งมีทั้งหมด 8 ระดับ (0-7) และจะมีการรายงานสถานะไปที่เซิร์ฟเวอร์ โดยผู้ใช้สามารถตรวจสอบสถานะได้จากบนเว็บหรือบนโทรศัพท์มือถือ



รูปที่ 4-17 แสดงปุ่มปรับความสว่างแบบ Manual

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

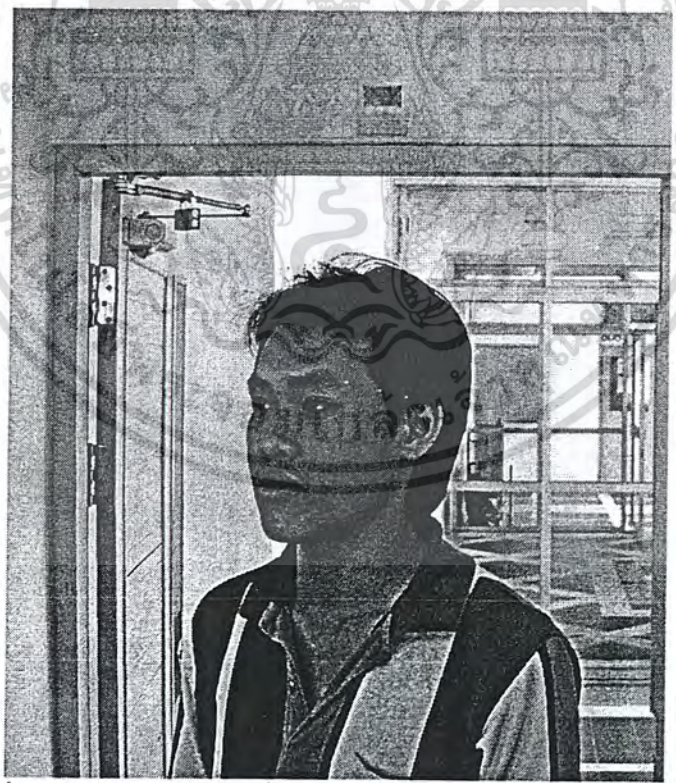


รูปที่ 4-18 แสดงระดับความสว่างที่ 0 , 3 และ 7 ตามลำดับซ้ายไปขวา

4.11 การควบคุมการทำงานของ Security Module ผ่านเว็บบราวเซอร์

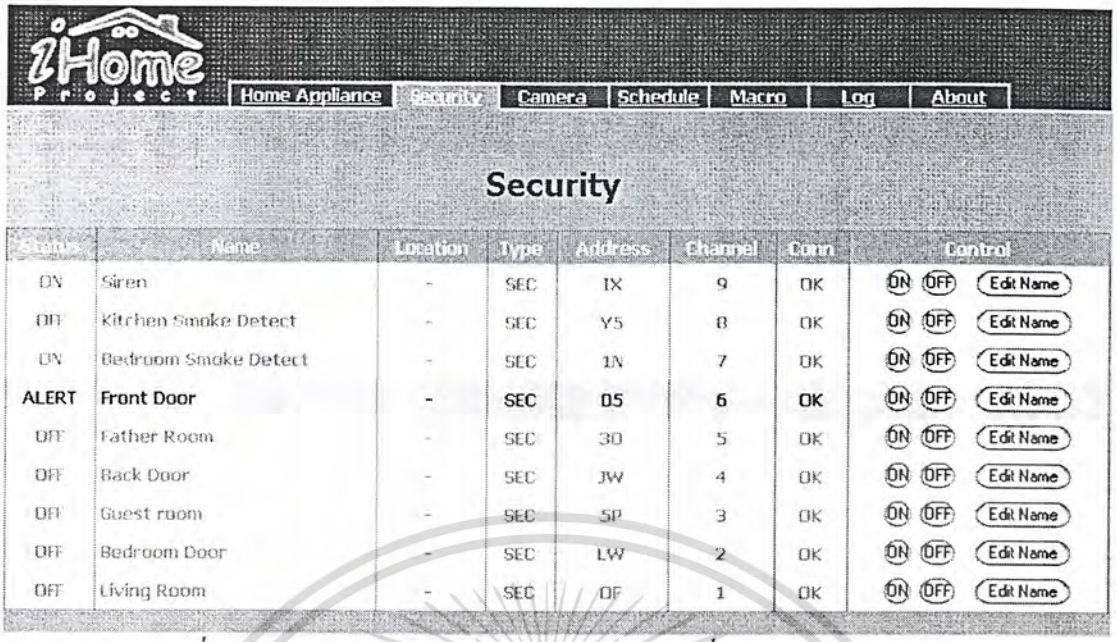
การควบคุมของ Security Module มี 2 ฟังก์ชันคือการเปิดและการปิด การเปิดคือการทำให้ช่องรับข้อมูลของ Security Module ซึ่งต่ออยู่กับ Sensor อยู่ในสถานะพร้อมทำงาน และสามารถตรวจจับความผิดปรกติได้ ส่วนการปิดคือการปิดช่องรับข้อมูล ซึ่งจะไม่มีการแจ้งเตือนหรือรายงานใดๆทั้งสิ้นในช่องรับข้อมูลที่ได้ทำการปิดไว้

การทดลองต่อไปนี้ ทดลองโดยให้คนเดินผ่านบริเวณที่มีการตั้ง Sensor จับการเคลื่อนไหวไว้



รูปที่ 4-19 แสดงเหตุการณ์เมื่อมีคนเดินผ่าน PIR Sensor (Front Door)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

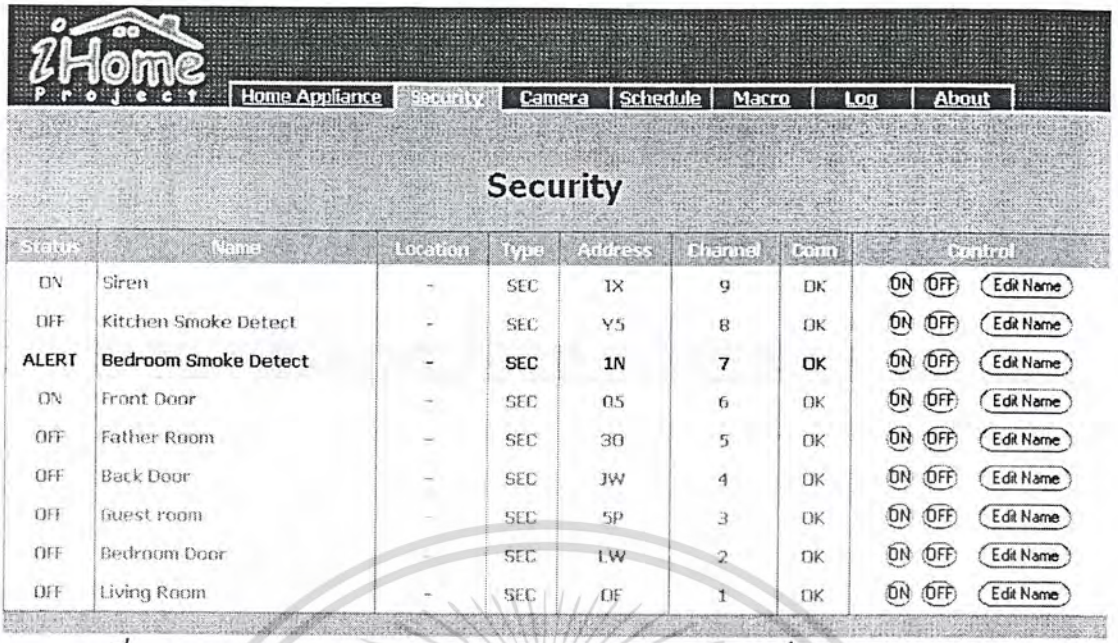


รูปที่ 4-20 แสดงสถานะของเซ็นเซอร์ Front Door ที่ตรวจพบความผิดปกติ

ในลักษณะเดียวกัน เป็นการทดลองโดยใช้เซ็นเซอร์ตรวจจับควัน (Smoke Detector) โดยจุดไฟเพื่อให้กำเนิดควันไปที่ตัวตรวจจับควัน

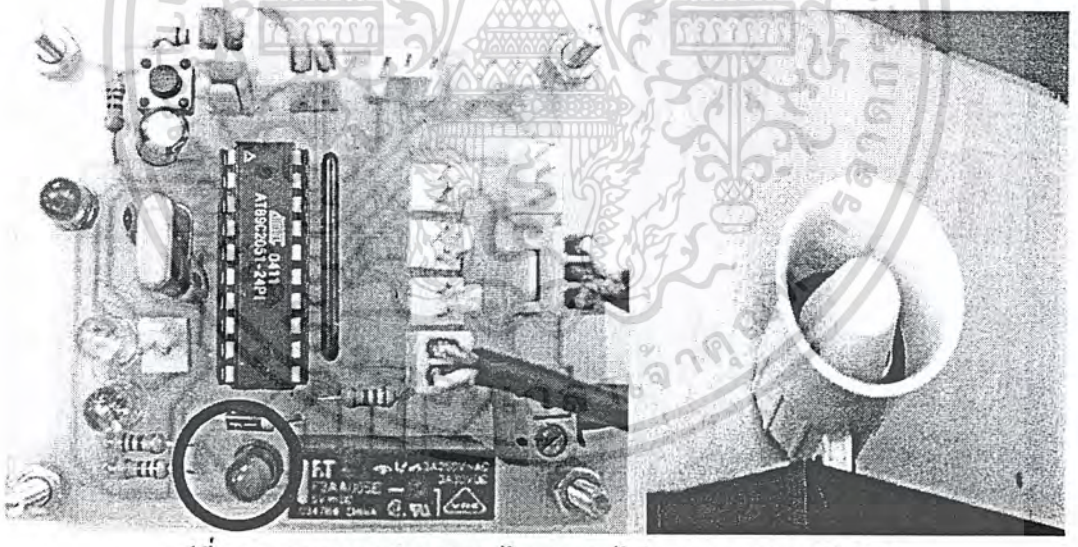
รูปที่ 4-21 แสดงเหตุการณ์เมื่อเกิดเพลิงไหม้ที่ Bedroom Smoke Detector

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



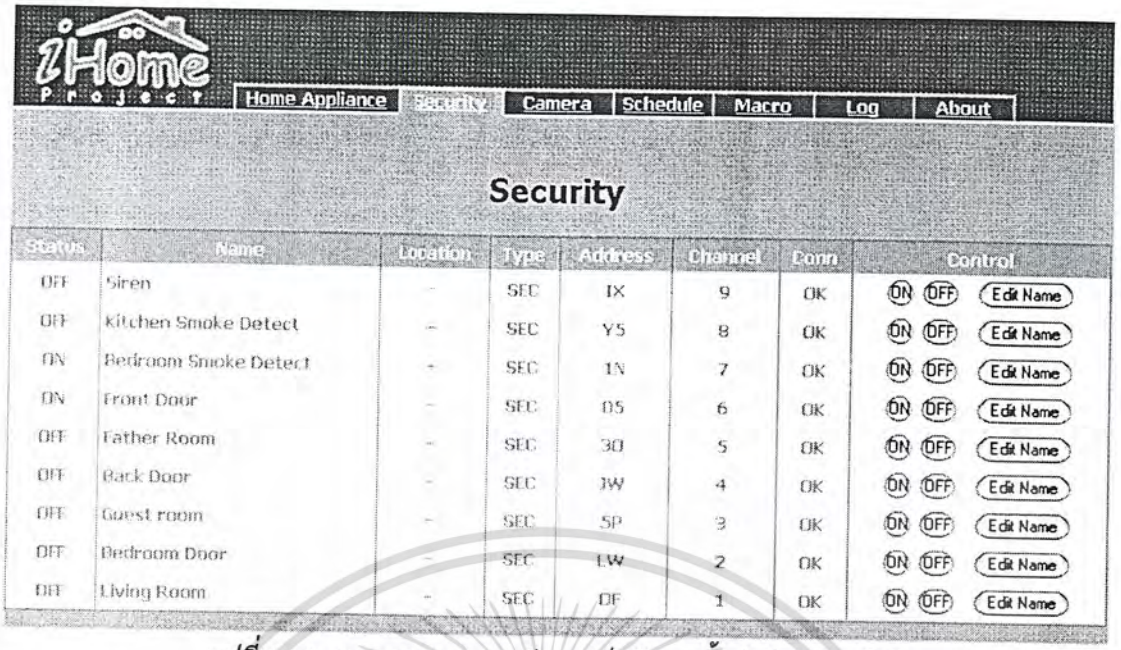
รูปที่ 4-22 แสดงสถานะของเซ็นเซอร์ Bedroom Smoke Detector ที่ตรวจพบความผิดปกติ

ขณะเดียวกันที่เซ็นเซอร์ตรวจจับความผิดปรกติได้ Security Module จะสั่งให้ Siren ทำงาน สังเกตที่ไฟแสดงสถานการทำงานของไซเรนจะติดขึ้น



รูปที่ 4-23 แสดงการทำงานของไซเรนและไฟแสดงสถานะการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

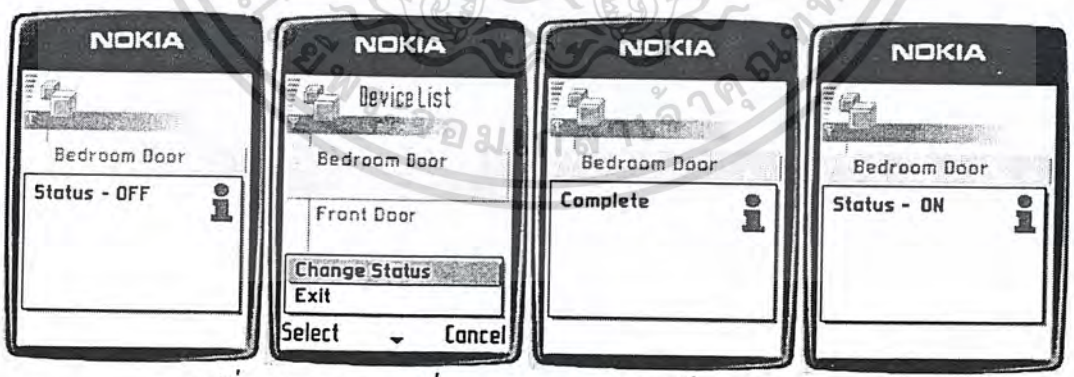


รูปที่ 4-24 แสดงสถานะของเซ็นเซอร์หลังการตั้งค่าสู่สถานะปกติ

4.12 การควบคุมการทำงานของ Security Module ผ่านโทรศัพท์มือถือ

เมื่อต้องการเปลี่ยนสถานะการทำงานของอุปกรณ์ให้เลือก Change Status จากนั้นกด Select แล้วรอผลลัพธ์ตอบกลับมาว่าเปิดปิดสำเร็จหรือไม่ โดยสามารถทราบสถานะของอุปกรณ์ได้จากการกดปุ่ม Select ที่อุปกรณ์หรือสังเกตจากสีของรูปภาพหลอดไฟด้านหน้า

จากรูปเมื่อกด Change Status เพื่อเปลี่ยนสถานะของอุปกรณ์ชื่อ Bedroom Door จาก ปิดเป็นเปิด จากนั้นมีการตอบ Complete กลับมา แสดงว่าอุปกรณ์ได้ทำการเปิดการทำงานเรียบร้อยแล้ว แล้วจึงปรับปรุงการแสดงผลที่หน้าจอให้ Bedroom Door มีสถานะเป็น ON



รูปที่ 4-25 แสดงการเปลี่ยนสถานะของอุปกรณ์ชื่อ Bedroom Door

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.13 การทดสอบการทำงานของระบบ Macro

ระบบ Macro นั้นทำงานในลักษณะการสนองตอบต่อเหตุการณ์ คือ เมื่อมีเหตุการณ์เกิดขึ้นแล้วจะให้ทำอะไร ในการตอบสนองของเหตุการณ์สามารถเป็นไปได้ในหลายลักษณะทั้งการสั่งงาน หรือการแจ้งเตือน ทั้งนี้สามารถทำเป็นชุดของคำสั่งต่อเนื่องกันได้

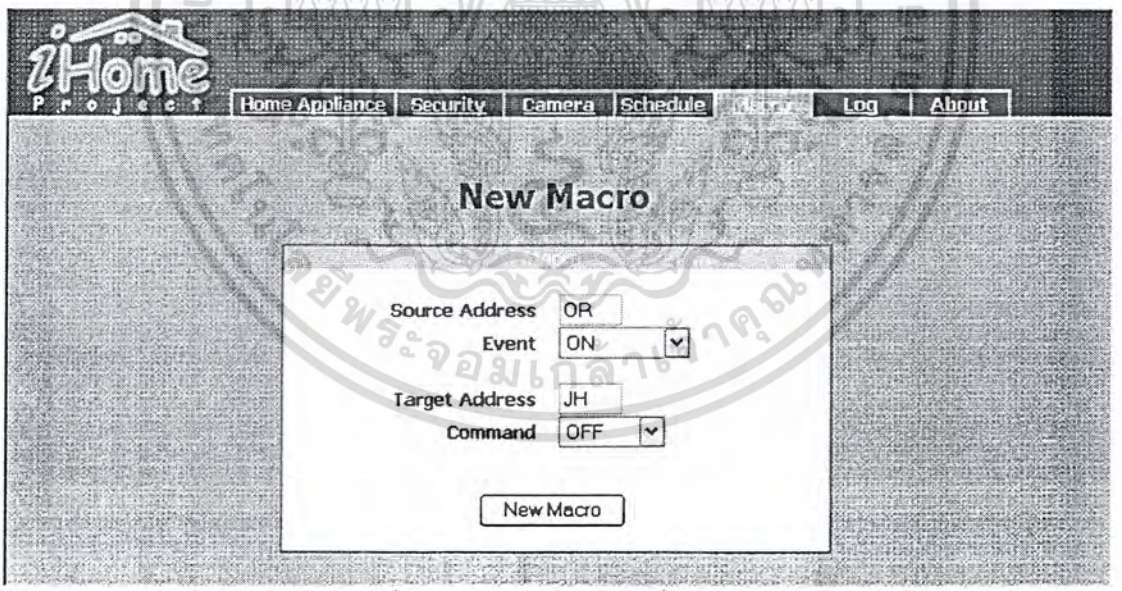
เหตุการณ์ที่สามารถตรวจสอบได้มีดังนี้

1. การเปิด
2. การปิด
3. การเปลี่ยนสถานะ คือ จากเปิดไปปิด หรือจากปิดไปเปิด
4. การแจ้งเหตุจาก Security Module
5. การถ่ายรูป

คำสั่งที่สนับสนุนต่อการตอบสนองต่อเหตุการณ์ สามารถทำได้ทุกคำสั่ง เช่น การเปิดปิด การควบคุมระดับความสว่าง การควบคุม Security Module การถ่ายรูป การส่ง SMS เป็นต้น

4.13.1 การทดสอบการเพิ่ม Macro

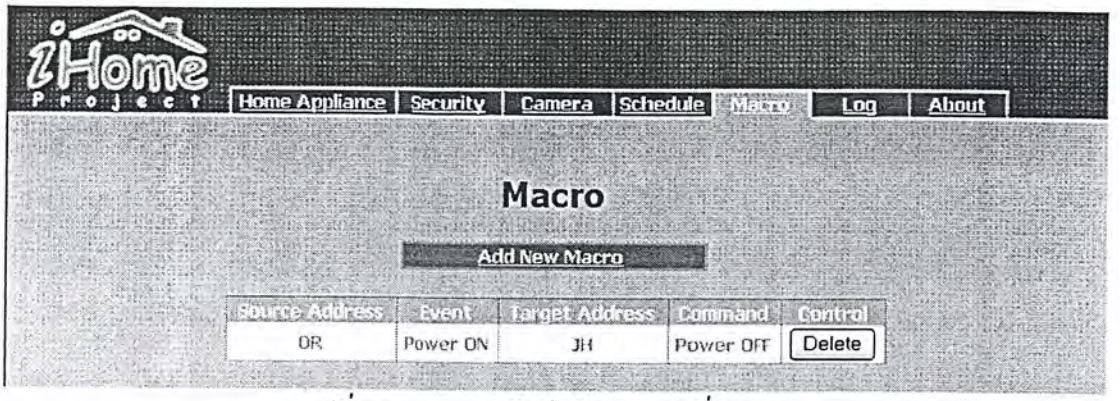
ทดสอบการเพิ่มรายการ Macro เข้าไปในระบบ โดยจะทดลองกำหนดค่าให้ เมื่ออุปกรณ์ที่แอดแดรส 0R ถูกเปิดขึ้นมา จะมีผลทำให้อุปกรณ์ที่แอดแดรส JH ถูกปิดลงเอง



รูปที่ 4-26 แสดงหน้าจอการเพิ่ม Macro

หลังจากกดปุ่ม New Macro ระบบจะทำการบันทึกค่าเก็บไว้ แล้วแสดงผลการเพิ่ม Macro ออกมาให้ดูทางหน้าจอ ซึ่งจะพบรายการ Macro ใหม่แสดงอยู่ โดยมีค่าการทำงานตามที่กำหนดไว้

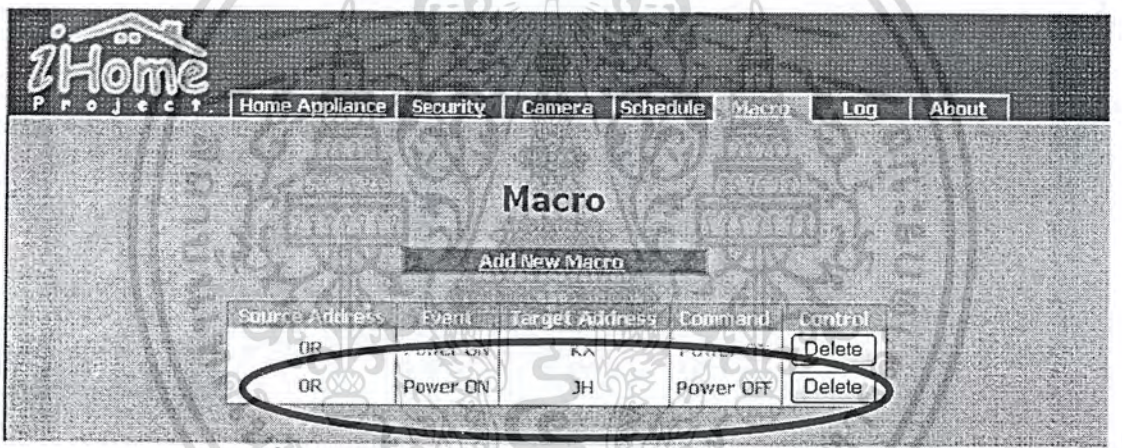
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4-27 แสดงผลลัพธ์หลังจากการเพิ่ม Macro

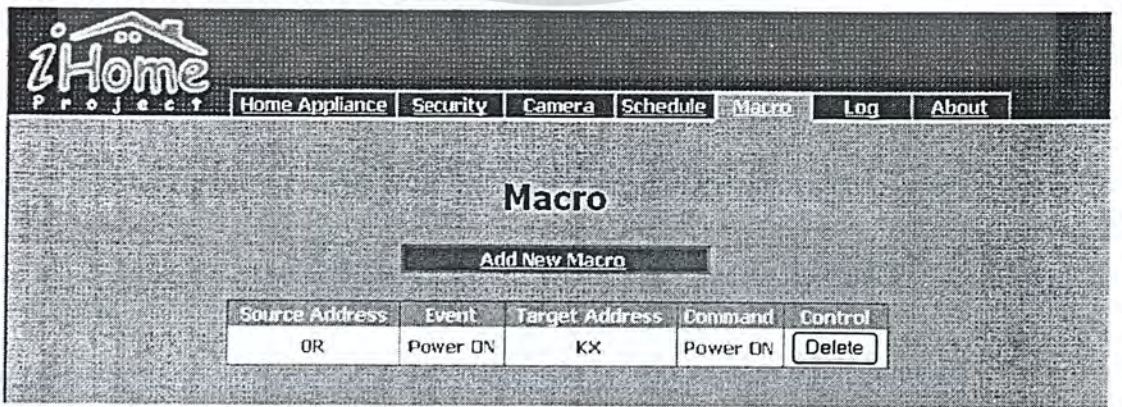
4.13.2 การทดสอบการลบ Macro

เมื่อมีรายการ Macro ใดที่ผู้ใช้ไม่ต้องการ จะสามารถลบ Macro ออกได้ โดยการกดปุ่ม Delete ที่รายการนั้นๆ จากรูปจะทำการลบรายการที่สองออกไปจากตาราง Macro



รูปที่ 4-28 แสดงหน้าจอรายการ Macro ที่มีอยู่ก่อนการลบออก

หลังจากกดปุ่ม Delete ระบบจะทำการปรับปรุงฐานข้อมูลแล้วแสดงผลออกมาทางหน้าจอโดยรายการที่ต้องการจะลบจะหายไปจากรายการ Macro

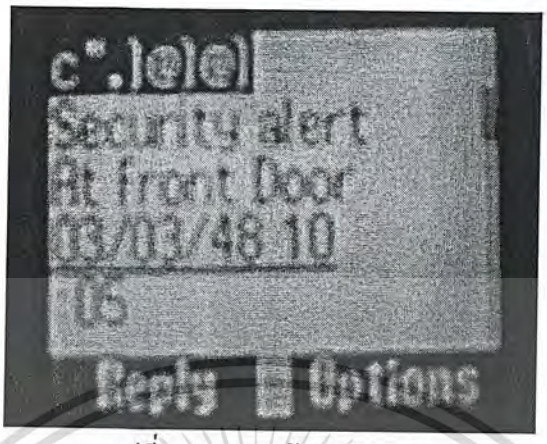


รูปที่ 4-29 แสดงหน้าจอรายการ Macro ที่มีอยู่หลังการลบออก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.13.3 การทดสอบการแจ้งเตือนด้วย Macro

หลังจากทำการเพิ่ม Macro ของเหตุการณ์ผิดปกติ โดยให้ส่ง SMS ออกไปได้ผลดังนี้



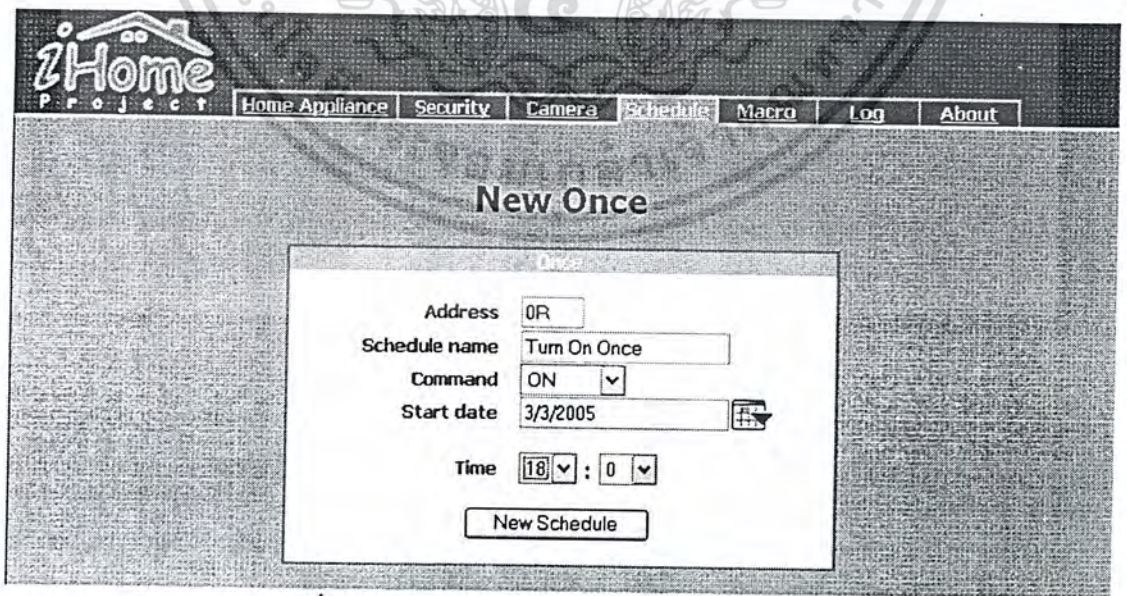
รูปที่ 4-30 แสดงข้อความ SMS

4.14 การทดสอบการทำงานของระบบตารางเวลา

ตารางเวลาคือการกำหนดตารางการทำงานให้กับแต่ละ Interface Module ซึ่งจะทำงานตามที่กำหนดไว้ และสามารถตรวจสอบผลการทำงานได้จาก Log

4.14.1 การทดลองการเพิ่มตารางเวลาแบบ Once ผ่านเว็บบราวเซอร์

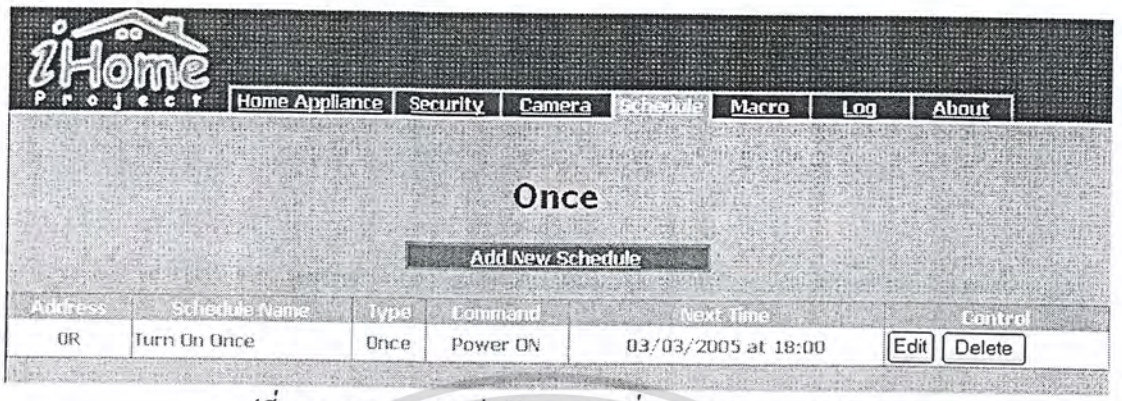
ตารางเวลาแบบ Once จะเป็นการกำหนดให้อุปกรณ์ทำงานเพียงแค่ครั้งเดียวเท่านั้น จากรูปจะเป็นการกำหนดตารางเวลาแบบ Once ให้อุปกรณ์ที่แอดเดรส 0R โดยกำหนดชื่อตารางเวลาเป็น Turn ON Once และกำหนดคำสั่งให้เปิด โดยให้เริ่มทำงานในวันที่ 3/3/2005 เวลา 18.00 น.



รูปที่ 4-31 แสดงหน้าจอการเพิ่มตารางเวลาแบบ Once

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

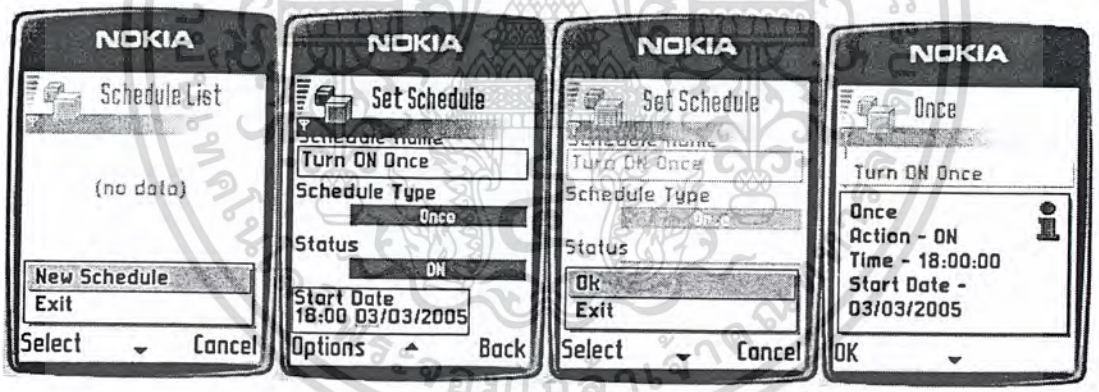
หลังจากกดปุ่ม New Schedule แล้วจะทำการบันทึกค่าลงฐานข้อมูล แล้วแสดงผลออกทางหน้าจอ โดยจะพบรายการใหม่ที่ได้เพิ่มเข้าไป ซึ่งจะมีการทำงานตามที่ได้กำหนดไว้



รูปที่ 4-32 แสดงผลลัพธ์หลังจากการเพิ่มตารางเวลาแบบ Once

4.14.2 การทดลองการเพิ่มตารางเวลาแบบ Once ผ่านโทรศัพท์มือถือ

เริ่มจากเลือก New Schedule จากนั้นทำการกำหนดค่าต่างๆ ลงไป โดยให้ชื่อว่า Turn ON Once ให้ทำคำสั่งเปิด โดยเริ่มทำงานวันที่ 3/3/2005 เวลา 18.00 น. จากนั้นทำการยืนยันการเพิ่มตารางเวลา แล้วรอผลตอบกลับมา แล้วจึงไปทำการเรียกดูตารางเวลาอีกครั้งจะพบว่ามีรายการใหม่ชื่อ Turn ON Once ซึ่งมีค่าการทำงานตามที่ได้กำหนดไว้



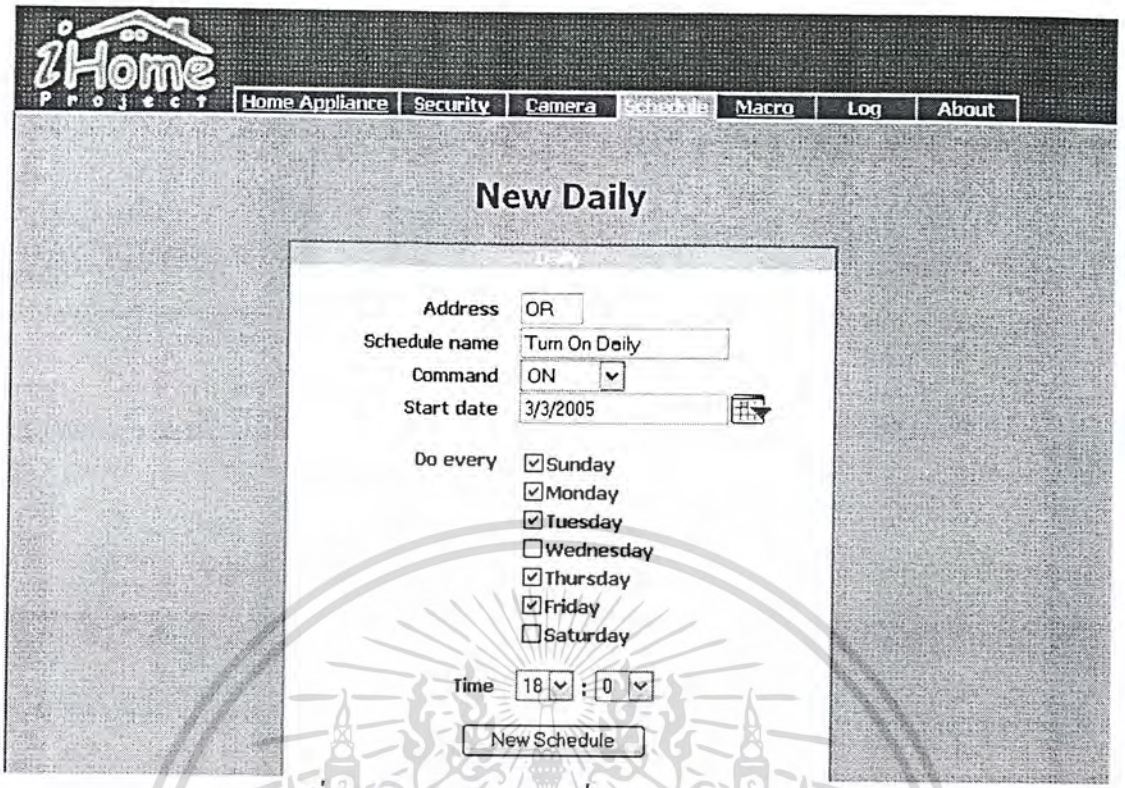
รูปที่ 4-33 แสดงหน้าจอโทรศัพท์มือถือในการเพิ่มตารางเวลาแบบ Once

4.14.3 การทดลองการเพิ่มตารางเวลาแบบ Daily ผ่านเว็บบราวเซอร์

ตารางเวลาแบบ Daily จะเป็นการกำหนดให้อุปกรณ์ทำงานได้ทุกวันในรอบสัปดาห์ โดยสามารถกำหนดได้ว่าจะให้อุปกรณ์ทำงานวันไหนบ้าง

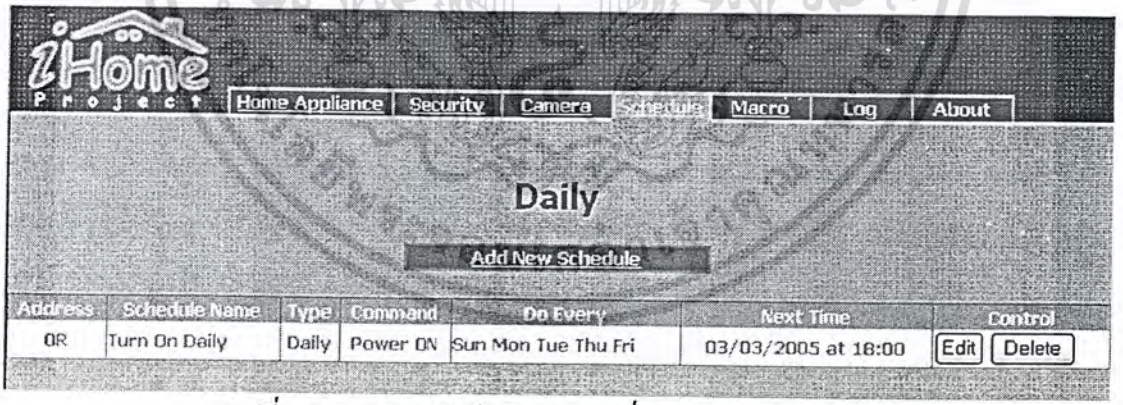
จากรูปจะเป็นการกำหนดตารางเวลาแบบ Daily ให้อุปกรณ์ที่แอดเดรส 0R โดยกำหนดชื่อตารางเวลาเป็น Turn On Daily และกำหนดคำสั่งให้เปิด โดยเริ่มทำงานในวันที่ 3/3/2005 เวลา 18.00 น. และให้ทำงานทุกๆ วันอาทิตย์ วันจันทร์ วันอังคาร วันพฤหัสบดี และวันศุกร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4-34 แสดงหน้าจอการเพิ่มตารางเวลาแบบ Daily

หลังจากกดปุ่ม New Schedule แล้วจะทำการบันทึกค่าลงฐานข้อมูล แล้วแสดงผลออกทางหน้าจอ โดยจะพบรายการใหม่ได้ทำการเพิ่มเข้าไป ซึ่งมีค่าการทำงานตามที่ได้กำหนดไว้

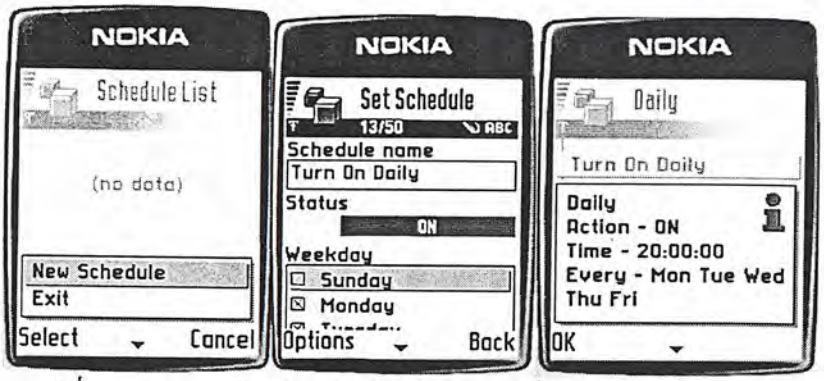


รูปที่ 4-35 แสดงผลลัพธ์หลังจากการเพิ่มตารางเวลาแบบ Daily

4.14.4 การทดลองการเพิ่มตารางเวลาแบบ Daily ผ่านโทรศัพท์มือถือ

เริ่มจากเลือก New Schedule จากนั้นทำการกำหนดค่าต่างๆ ลงไป โดยให้ชื่อว่า Turn ON Daily ให้ทำคำสั่งเปิด โดยเริ่มทำงานวันที่ 3/3/2005 เวลา 20.00 น. และจะทำทุกๆ วันจันทร์ วันอังคาร วันพุธ วันพฤหัสบดี วันศุกร์ จากนั้นทำการยืนยันการเพิ่มตารางเวลา แล้วรอผลตอบกลับมา แล้วจึงไปทำการเรียกดูตารางเวลาอีกครั้งจะพบว่ามีรายการ Turn ON Daily เพิ่มขึ้นมา ซึ่งมีค่าการทำงานตามที่ได้กำหนดไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

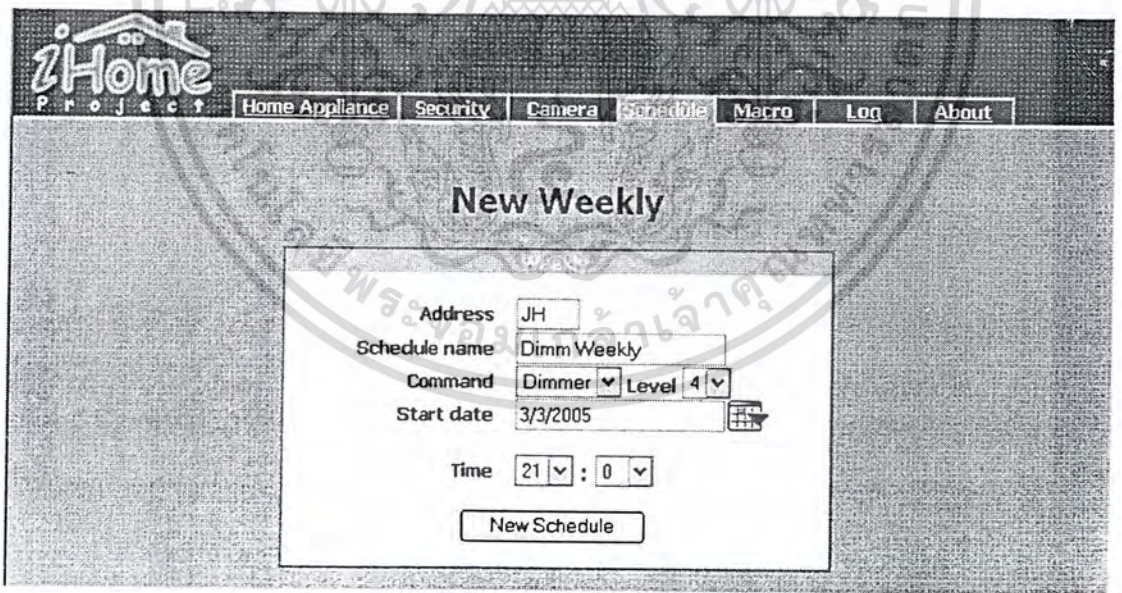


รูปที่ 4-36 แสดงหน้าจอโทรศัพท์มือถือในการเพิ่มตารางเวลาแบบ Daily

4.14.5 การทดลองการเพิ่มตารางเวลาแบบ Weekly ผ่านเว็บบราวเซอร์

ตารางเวลาแบบ Weekly จะเป็นการกำหนดให้อุปกรณ์ทำงานตามวันที่กำหนดในทุกๆ สัปดาห์ โดยใส่วันที่ที่ต้องการให้เริ่มทำงานลงไป แล้วระบบจะรู้เองว่าเป็นวันไหนของสัปดาห์ ซึ่งจะต่างจากตารางเวลาแบบ Daily คือจะกำหนดได้เพียงแค่วันเดียวในรอบสัปดาห์

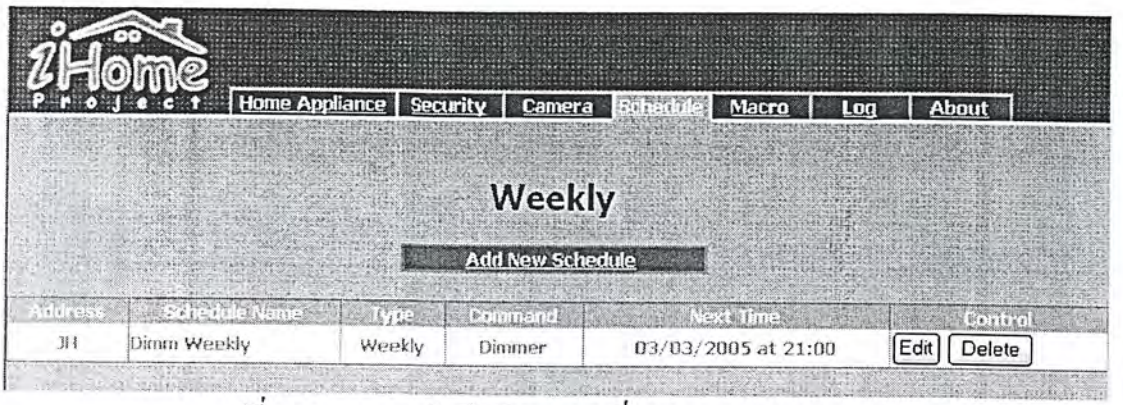
จากรูปจะเป็นการกำหนดตารางเวลาแบบ Weekly ให้อุปกรณ์ที่แอดแดรส JH โดยกำหนดชื่อตารางเวลาเป็น Dimmer Weekly และกำหนดคำสั่งให้ปรับค่าความสว่างไปที่ระดับ 4 ให้เริ่มทำงานในวันที่ 3/3/2005 เวลา 21.00 น.



รูปที่ 4-37 แสดงหน้าจอการเพิ่มตารางเวลาแบบ Weekly

หลังจากกดปุ่ม New Schedule แล้วจะทำการบันทึกค่าลงฐานข้อมูล แล้วแสดงผลออกทางหน้าจอ โดยจะพบรายการใหม่ที่ได้ถูกเพิ่มเข้าไป โดยมีการทำงานต่างๆ ตามที่ได้กำหนดไว้

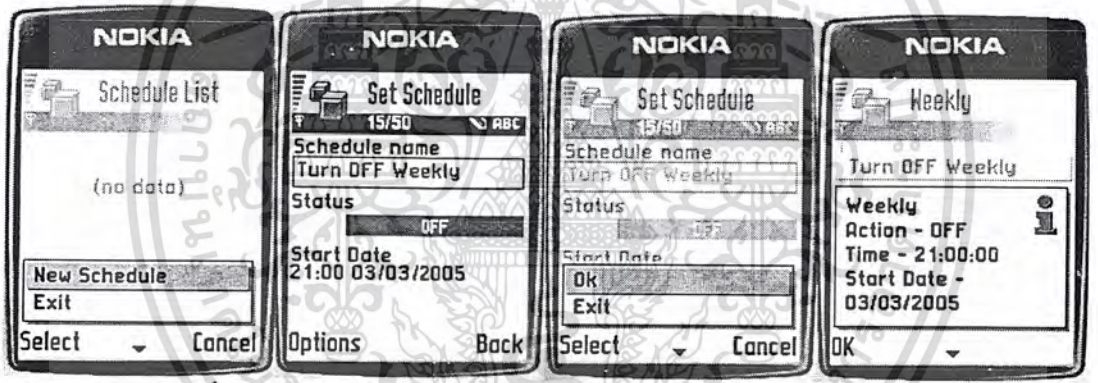
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4-38 แสดงผลลัพธ์หลังจากการเพิ่มตารางเวลาแบบ Weekly

4.12.6 การทดลองการเพิ่มตารางเวลาแบบ Weekly ผ่านโทรศัพท์มือถือ

เริ่มจากเลือก New Schedule จากนั้นทำการกำหนดค่าต่างๆ ลงไป โดยให้ชื่อว่า Turn OFF Weekly ให้ทำคำสั่งปิด โดยเริ่มทำงานวันที่ 3/3/2005 เวลา 21.00 น. จากนั้นทำการยืนยันการเพิ่มตารางเวลา แล้วรอผลตอบกลับมา เมื่อทำการเรียกดูตารางเวลาอีกครั้งจะพบว่ามีรายการตารางเวลาชื่อ Turn OFF Weekly เพิ่มเข้ามา



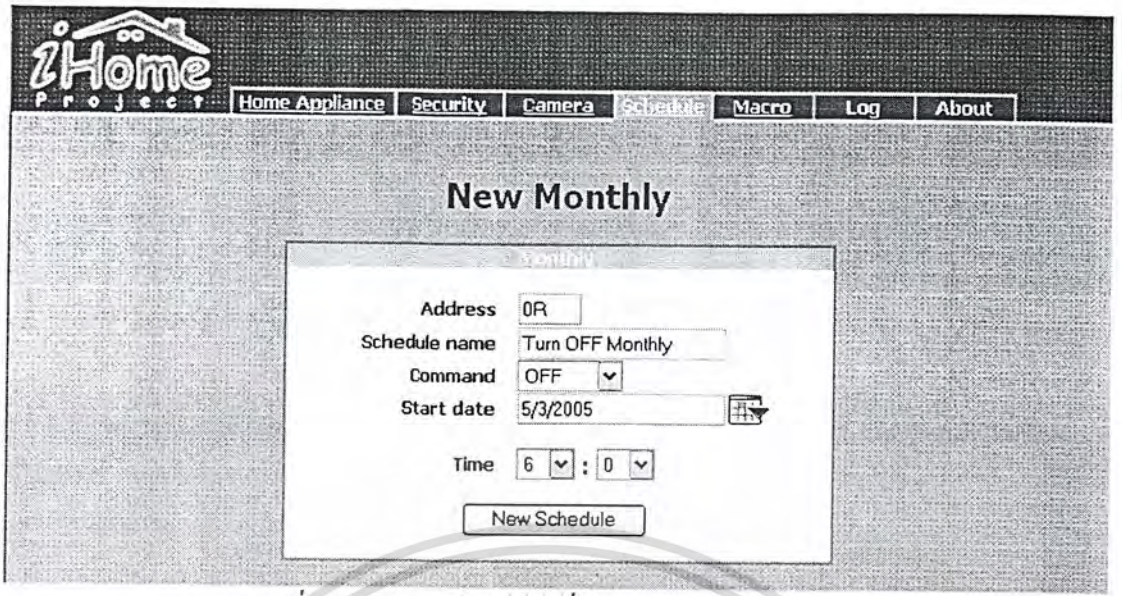
รูปที่ 4-39 แสดงหน้าจอโทรศัพท์มือถือในการเพิ่มตารางเวลาแบบ Weekly

4.14.7 การทดลองการเพิ่มตารางเวลาแบบ Monthly ผ่านเว็บบราวเซอร์

ตารางเวลาแบบ Monthly จะเป็นการกำหนดให้อุปกรณ์ทำงานแบบรายเดือน โดยการกำหนดวันที่ลงไปว่าทำให้ทำงานวันที่เท่าไรของเดือน

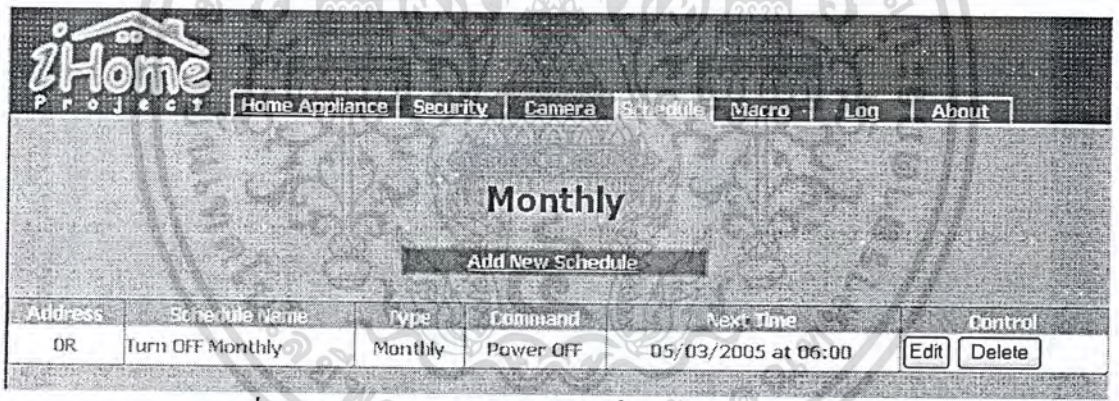
จากรูปจะเป็นการกำหนดตารางเวลาแบบ Monthly ให้อุปกรณ์ที่แอดแดรส 0R โดยกำหนดชื่อตารางเวลาเป็น Turn OFF Monthly และกำหนดคำสั่งให้ปิด โดยให้เริ่มทำงานในวันที่ 5/3/2005 เวลา 06.00 น.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4-40 แสดงหน้าจอการเพิ่มตารางเวลาแบบ Monthly

หลังจากกดปุ่ม New Schedule แล้วจะทำการบันทึกค่าลงฐานข้อมูล แล้วแสดงผลออกทางหน้าจอ โดยจะพบรายการใหม่ที่ได้ถูกเพิ่มเข้าไป ซึ่งจะมีค่าการทำงานตามที่ได้กำหนดไว้

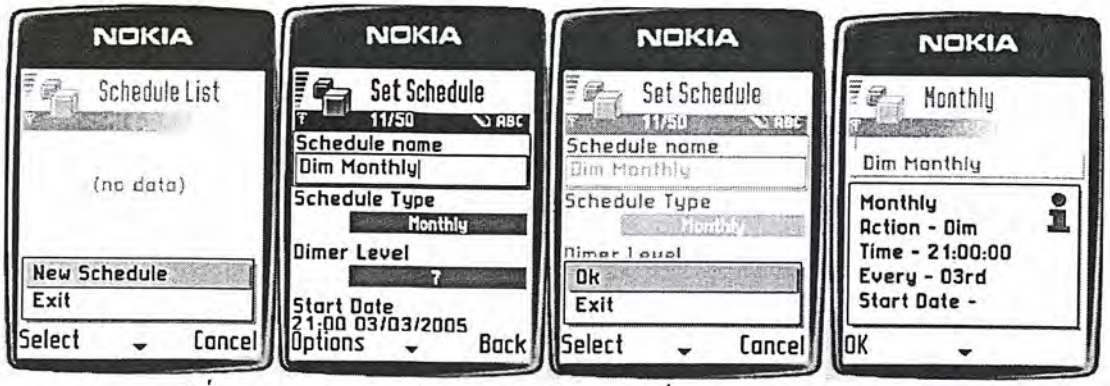


รูปที่ 4-41 แสดงผลลัพธ์หลังจากการเพิ่มตารางเวลาแบบ Monthly

4.14.8 การทดลองการเพิ่มตารางเวลาแบบ Monthly ผ่านโทรศัพท์มือถือ

เริ่มจากเลือก New Schedule จากนั้นทำการกำหนดค่าต่างๆ ลงไป โดยให้ชื่อว่า Dim Monthly ให้ทำการปรับระดับความสว่างไปที่ระดับ 7 โดยเริ่มทำงานวันที่ 3/3/2005 เวลา 21.00 น. จากนั้นทำการยืนยันการเพิ่มตารางเวลา แล้วรอผลตอบกลับมา เมื่อทำการเรียกดูตารางเวลาอีกครั้งจะพบว่ามีรายการตารางเวลาชื่อ Dim Monthly เพิ่มเข้ามาโดยมีการทำงานตามที่ได้กำหนดไว้

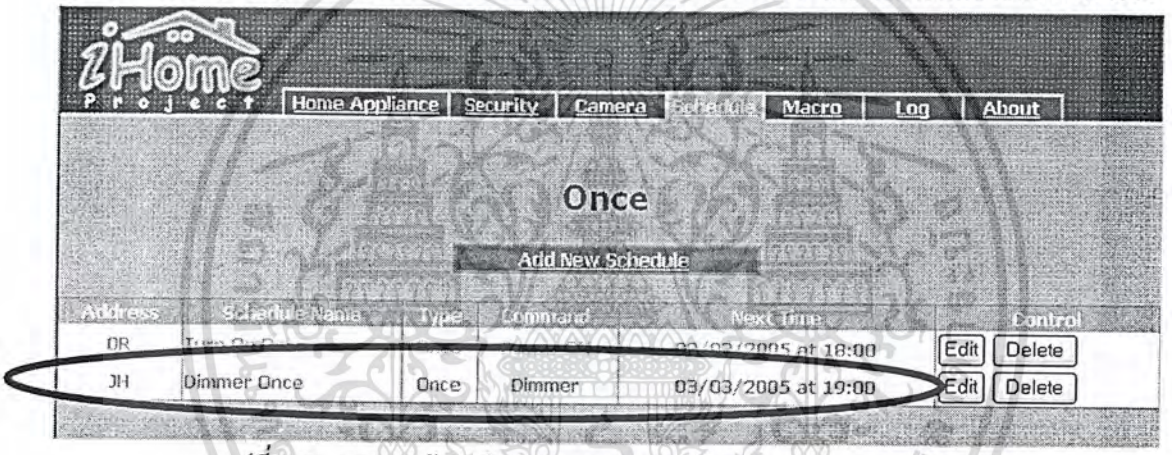
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4-42 แสดงหน้าจอโทรศัพท์มือถือในการเพิ่มตารางเวลาแบบ Monthly

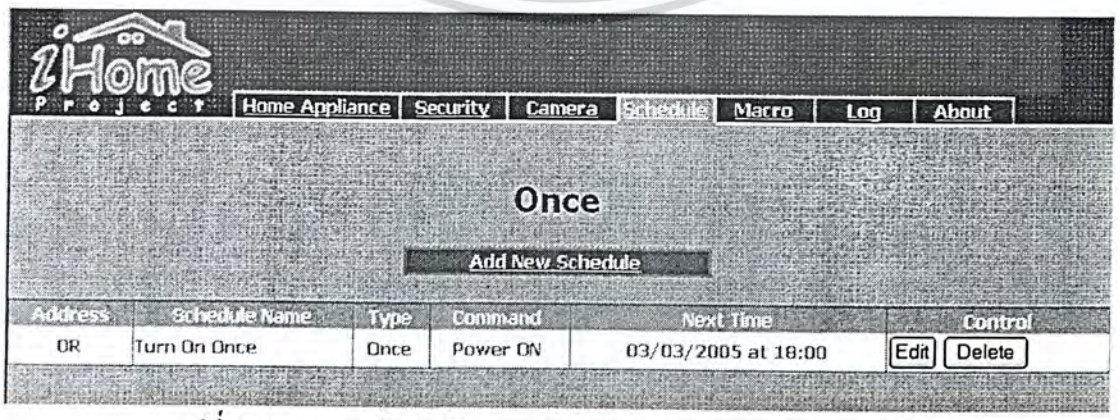
4.14.9 การทดลองลบตารางเวลา ผ่านเว็บบราวเซอร์

เมื่อไม่ต้องการใช้งานตารางเวลาใดๆ ก็สามารถลบรายการนั้นออกได้ โดยกดปุ่ม Delete ที่ตารางเวลานั้นๆ จากรูปจะทำการลบรายการตารางเวลาแบบ Once โดยจะลบรายการที่สองออกจากตาราง



รูปที่ 4-43 แสดงหน้าจอก่อนการลบรายการออกจากตารางเวลาแบบ Once

เมื่อกดปุ่ม Delete จะทำการลบค่าตารางเวลาออกจากฐานข้อมูลแล้วทำการแสดงผลออกทางหน้าจอ โดยจะพบว่ารายการต้องการลบได้หายไปแล้ว



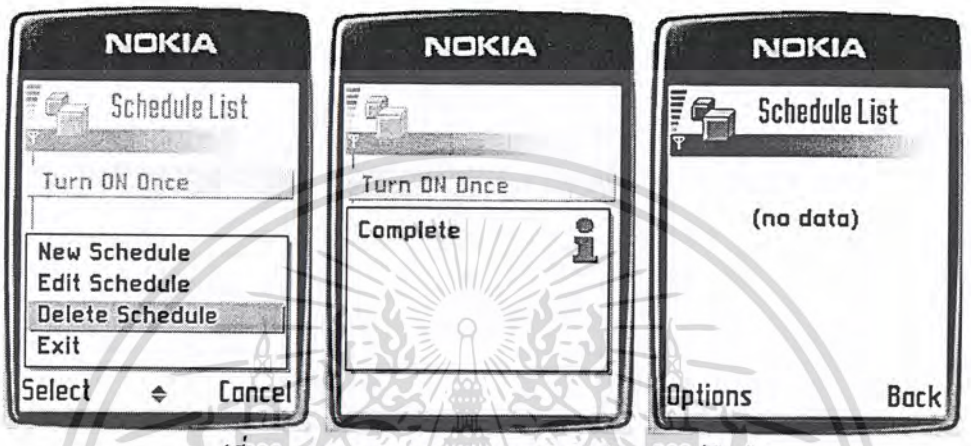
รูปที่ 4-44 แสดงหน้าจอหลังการลบรายการออกจากตารางเวลาแบบ Once

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.14.10 การทดลองลบตารางเวลา ผ่านโทรศัพท์มือถือ

เริ่มจากการเลือก Delete Schedule ในรายการตารางเวลาที่ต้องการ จากนั้นรอการตอบกลับมา ว่าสามารถลบได้หรือไม่ เมื่อลบตารางเวลาได้แล้วจะไม่พบรายการนั้น ในรายการตารางเวลาอีก

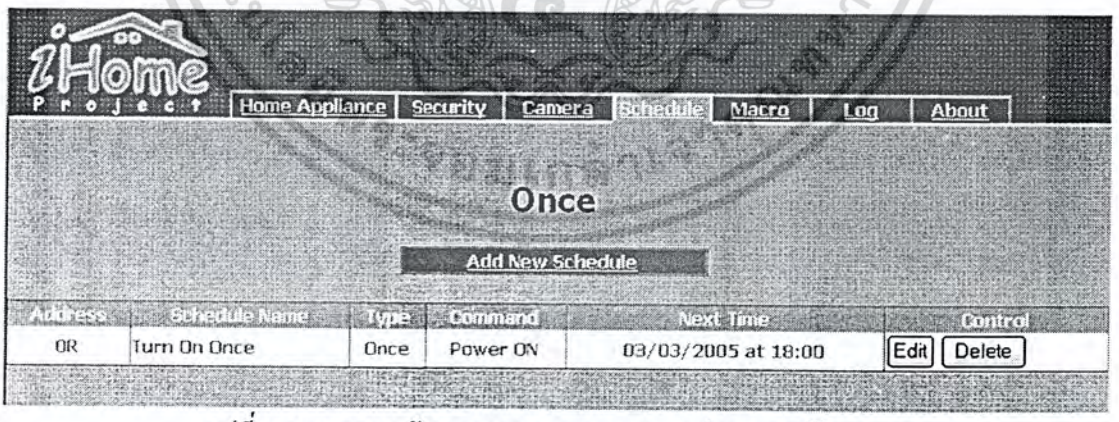
จากรูปได้เลือกลบรายการ Turn ON Once โดยเมื่อเลือก Delete Schedule จากนั้นจึงได้รับผลตอบกลับมาว่า Complete แสดงว่าลบได้สำเร็จ เมื่อทดลองเรียกดูตารางเวลาอีกครั้งพบว่า ไม่พบรายการ Turn ON Once แล้ว



รูปที่ 4-45 แสดงการลบตารางเวลาผ่านโทรศัพท์มือถือ

4.14.11 การทดลองแก้ไขตารางเวลา ผ่านเว็บบราวเซอร์

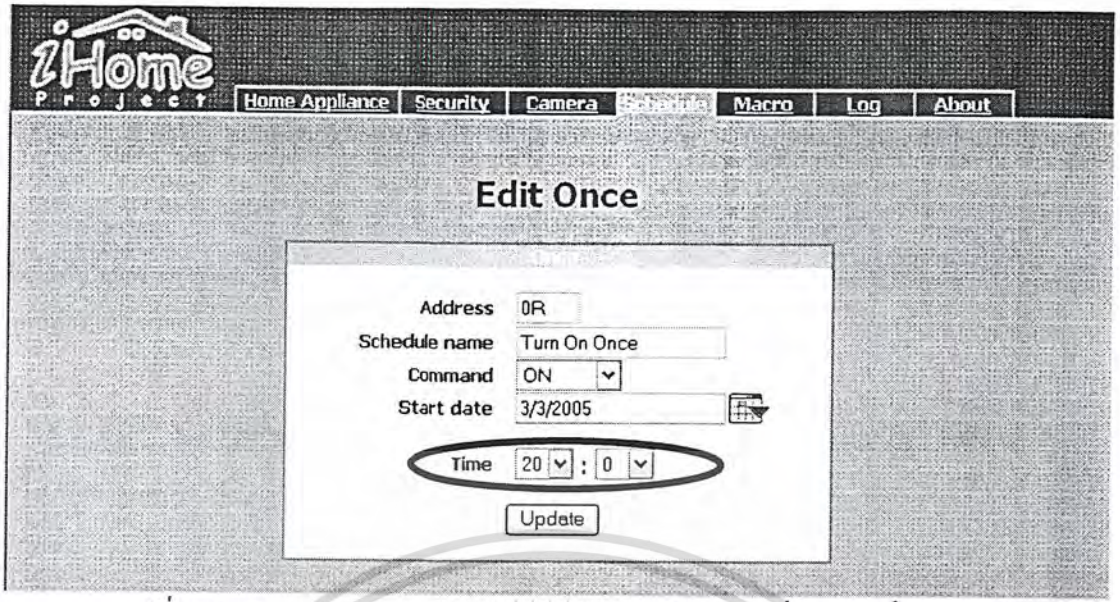
เมื่อมีตารางเวลาเดิมอยู่แล้ว แต่ต้องการปรับเปลี่ยนค่าบางอย่าง เช่น คำสั่งในการทำงาน วันที่เริ่มทำงาน เวลา เป็นต้น สามารถเข้าไปแก้ไขได้โดยกดปุ่ม Edit ที่ตารางเวลานั้นๆ



รูปที่ 4-46 แสดงหน้าจอตารางเวลาแบบ Once ก่อนการแก้ไขตาราง

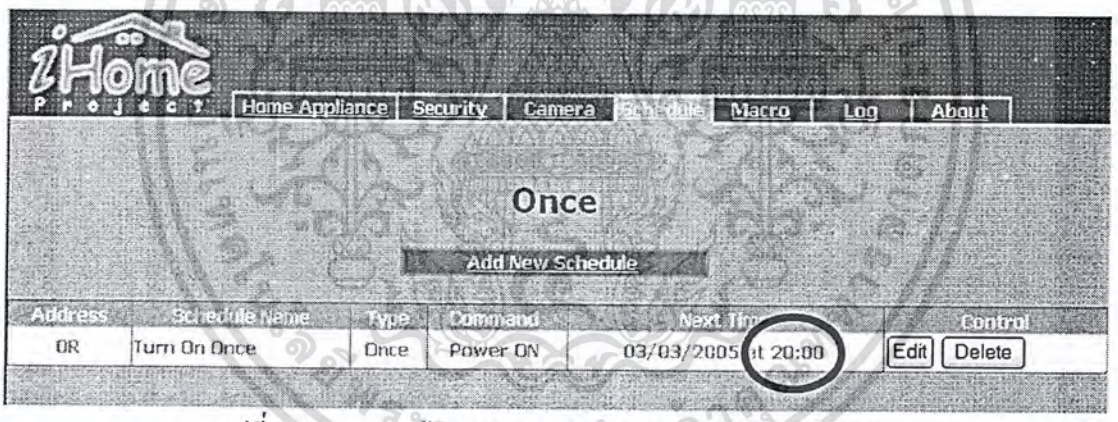
หลังจากกดปุ่ม Edit จะเข้าสู้หน้าจอการแก้ไขข้อมูลซึ่งในการทดลองนี้จะทำการแก้ไขเวลาจาก 18.00 น. ให้เป็น 20.00 น.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4-47 แสดงหน้าจอการแก้ไขรายการแบบ Once ให้มีเวลาเริ่มทำงานที่ 20.00 น.

เมื่อกดปุ่ม Update จะทำการปรับปรุงฐานข้อมูลแล้วแสดงผลการปรับปรุงข้อมูลออกมาทางหน้าจอ โดยเวลาได้ถูกแก้ไขจาก 18.00 เป็น 20.00 ตามที่ได้ทำการแก้ไขตารางเวลาไป



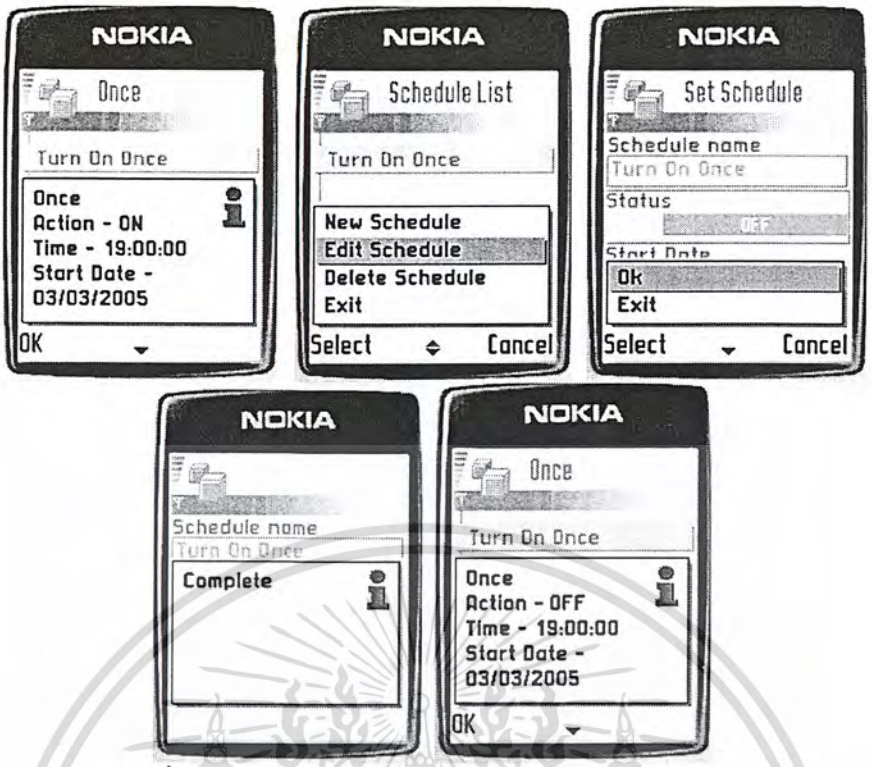
รูปที่ 4-48 แสดงหน้าจอตารางเวลาแบบ Once หลังการแก้ไขเวลา

4.14.12 การทดลองแก้ไขตารางเวลา ผ่านโทรศัพท์มือถือ

เมื่อต้องการแก้ไขตารางเวลา จะทำโดยการเลือก Edit Schedule จากนั้นจึงทำการแก้ไขข้อมูลต่างๆ เมื่อเรียบร้อยแล้วจึงกด Ok แล้วรอการตอบกลับมา ถ้าสามารถแก้ไขได้ เมื่อเรียกดูค่าจะต้องถูกต้องตามที่ได้กำหนดไว้

จากรูปตารางเวลาเดิมเป็นตารางเวลาแบบ Once โดยจะทำการเปิดอุปกรณ์ ในวันที่ 3/3/2005 เวลา 19.00 น. จากนั้นทำการเลือก Edit Schedule เพื่อแก้ไขตารางเวลา โดยแก้จากการทำคำสั่ง ON เป็นการทำคำสั่ง OFF แทน แล้วจึงกด Ok จากนั้นมีผลตอบกลับมาว่า Complete และเมื่อไปทำการเรียกดูตารางเวลาอีกครั้งพบว่าคำสั่งได้เปลี่ยนเป็น OFF ตามที่ได้แก้ไขไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4-49 แสดงการแก้ไขค่าตารางเวลาผ่านโทรศัพท์มือถือ

4.15 การทดสอบการใช้งานกล้อง

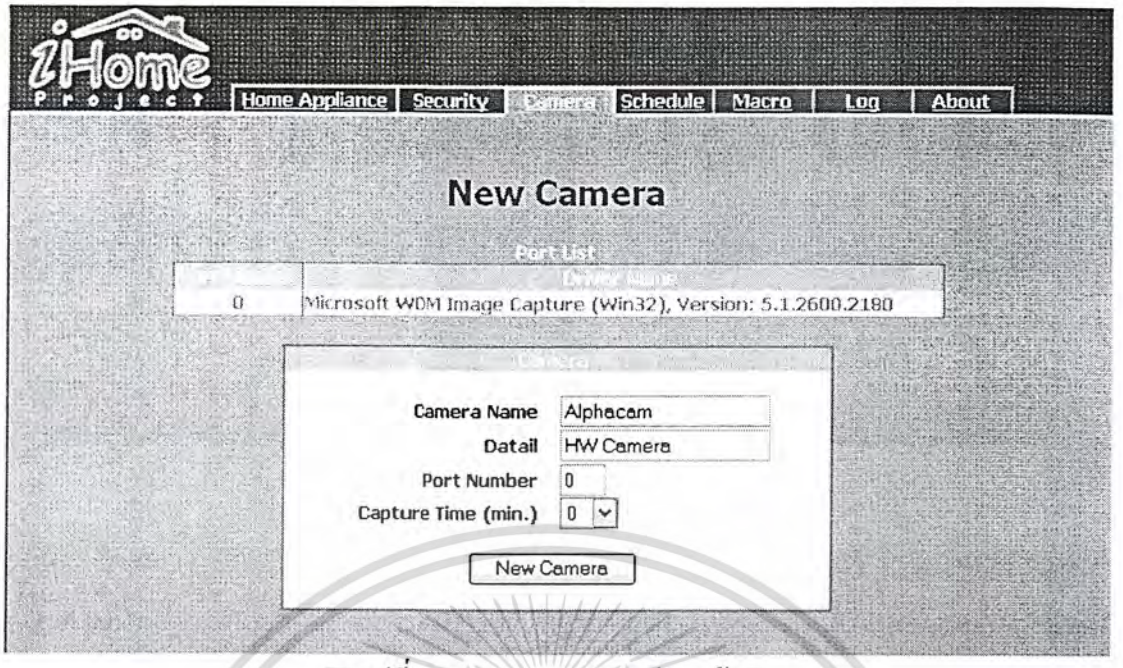
กล้องที่สามารถใช้กับระบบนี้ต้องเป็นกล้องที่สามารถติดต่อได้ผ่าน Interface มาตรฐานของวินโดว์ ซึ่งโปรแกรมจะสามารถตรวจจับได้อัตโนมัติ

4.15.1 การทดสอบการลงทะเบียนกล้อง

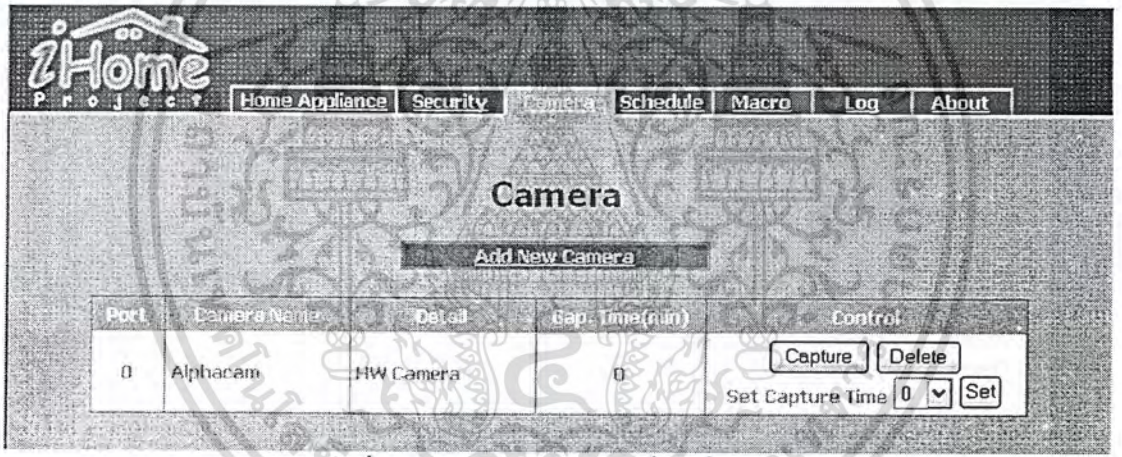
ในการใช้งานกล้องได้นั้น จำเป็นต้องลงทะเบียน เพื่อระบบจะได้รับทราบว่าผู้ใช้ต้องการใช้งานกล้องนั้น ๆ และเป็นการกำหนดค่าเริ่มต้นต่างๆ ให้กับอุปกรณ์ด้วย โดยจะต้องกำหนดชื่อกล้อง รายละเอียด พอร์ตที่ใช้ซึ่งดูได้จาก และเวลาในการบันทึกภาพ ซึ่งพอร์ตที่ใช้จะดูได้จากตารางพอร์ตที่มีอยู่

จากรูปเป็นการลงทะเทียนกล้องโดยให้มีชื่อว่า Alphacam รายละเอียดเป็น HW Camera กำหนดให้ใช้พอร์ต 0 และเวลาในการบันทึกภาพ ทุกๆ 0 นาที คือไม่มีการบันทึกภาพ จากนั้นเมื่อกด New Camera ก็จะพบรายการใหม่ที่ได้เพิ่มเข้าไป โดยมีรายละเอียดการทำงานตามที่กำหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4-50 แสดงการลงทะเบียนกล้อง



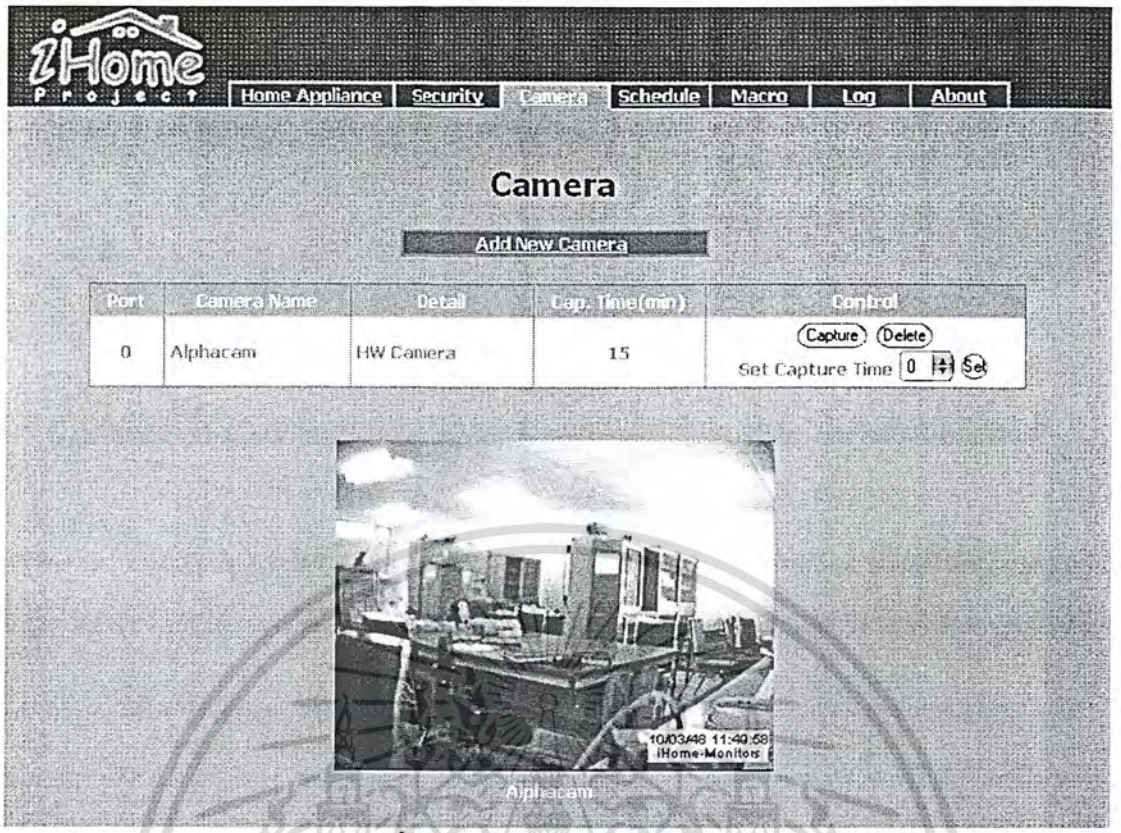
รูปที่ 4-51 แสดงรายการกล้องที่ได้เพิ่มเข้าไป

4.15.2 การทดสอบการบันทึกภาพแบบด้วยตนเอง ผ่านเว็บบราวเซอร์

เมื่อต้องการบันทึกภาพขณะนั้นด้วยตนเองโดยไม่ใช้การบันทึกภาพตามที่ตั้งค่าไว้ จะทำโดยการกดปุ่ม Capture ซึ่งจะทำการจับภาพจากกล้อง ณ เวลาขณะนั้นมาแสดงผลยังหน้าเว็บ ซึ่งจะแสดงรูปภาพและชื่อของกล้องที่ได้ทำการจับภาพมา โดยรูปภาพที่ได้จะมีการระบุวันและเวลาที่ได้ทำการบันทึกไว้ด้วย

จากรูปเมื่อกดปุ่ม Capture จะทำการจับภาพ ณ เวลาขณะนั้น คือ วันที่ 3/3/48 เวลา 21.20 น. ออกมาพร้อมแจ้งว่ากล้องที่ทำการจับภาพชื่อ Alphacam

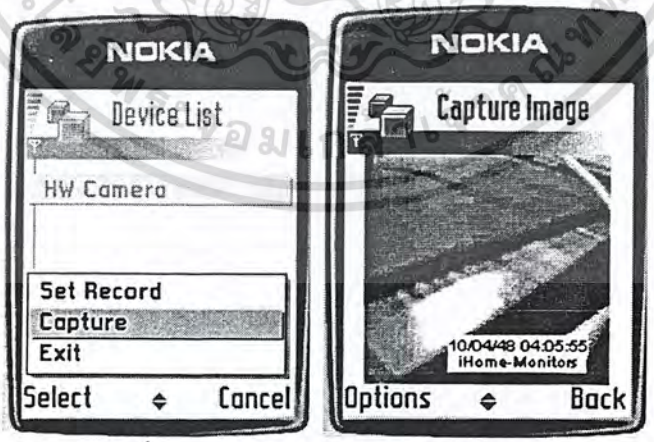
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4-52 แสดงการจับภาพด้วยกล้อง

4.15.3 การทดสอบการบันทึกภาพแบบด้วยตนเอง ผ่านโทรศัพท์มือถือ

เมื่อต้องการเรียกบันทึกภาพด้วยตนเองให้เลือก Capture จากนั้นรอผลการแสดงรูปภาพที่จับได้ โดยรูปที่ได้จะเป็นรูป ณ เวลาขณะนั้น



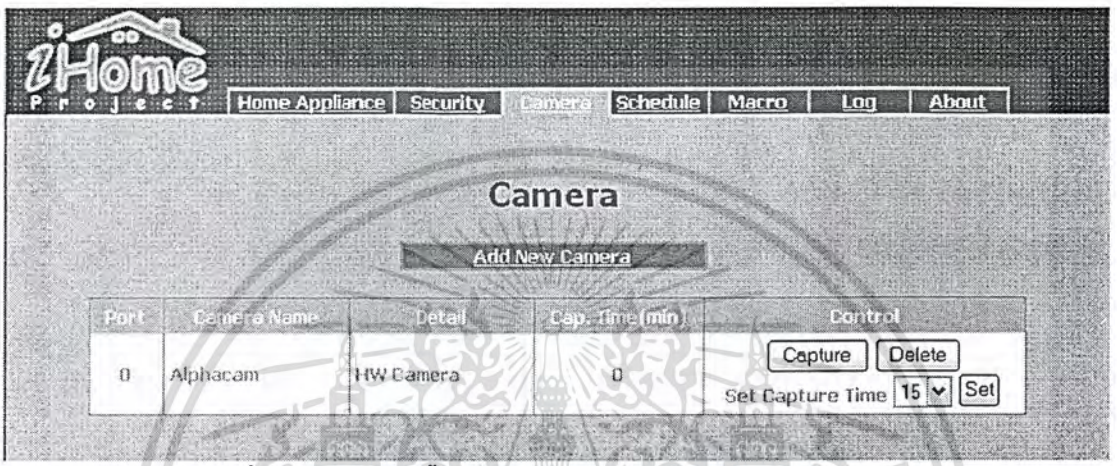
รูปที่ 4-53 แสดงการบันทึกภาพด้วยตนเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.15.4 การทดสอบการตั้งเวลาการบันทึกภาพ ผ่านเว็บบราวเซอร์

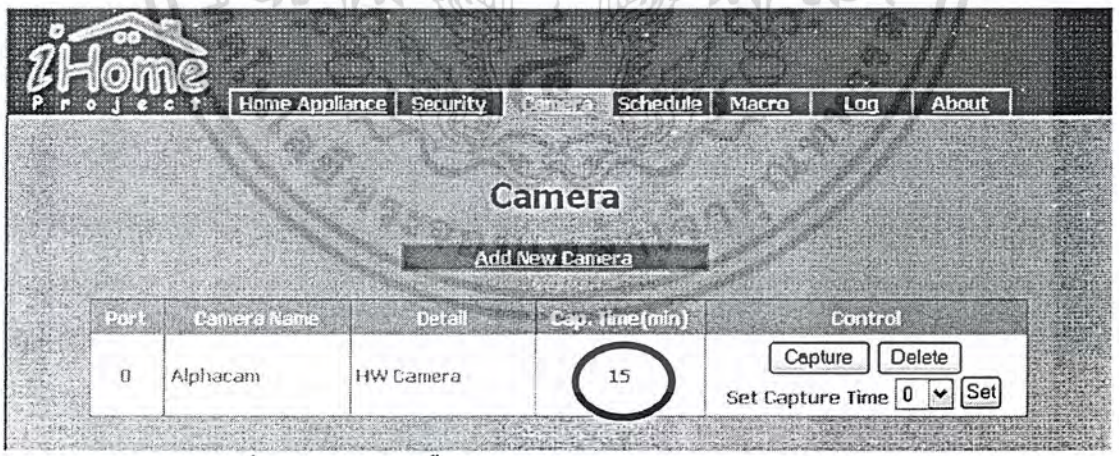
เมื่อต้องการตั้งเวลาการบันทึกภาพ ว่าจะให้บันทึกภาพทุกๆ กี่นาที สามารถทำได้โดยการไปตั้งค่าเวลาการบันทึกภาพซึ่งจะมีให้เลือก คือ 0, 5, 10, 15, 30 และ 60 นาที จากนั้นกดปุ่ม Set เพื่อทำการตั้งเวลาการบันทึกให้กับกล้องตัวนั้น โดยภาพที่ได้จากการบันทึกสามารถเข้าไปดูได้ในเมนู Camera -> History

จากรูปเป็นการตั้งเวลาการบันทึกภาพเป็นทุกๆ 15 นาทีจากเดิมที่กำหนดไว้ทุกๆ 0 นาที



รูปที่ 4-54 แสดงการตั้งเวลาบันทึกภาพทุกๆ 15 นาทีก่อนการกด Set

จากนั้นเมื่อกดปุ่ม Set จะทำการตั้งค่าการบันทึกภาพให้เป็นทุกๆ 15 นาที

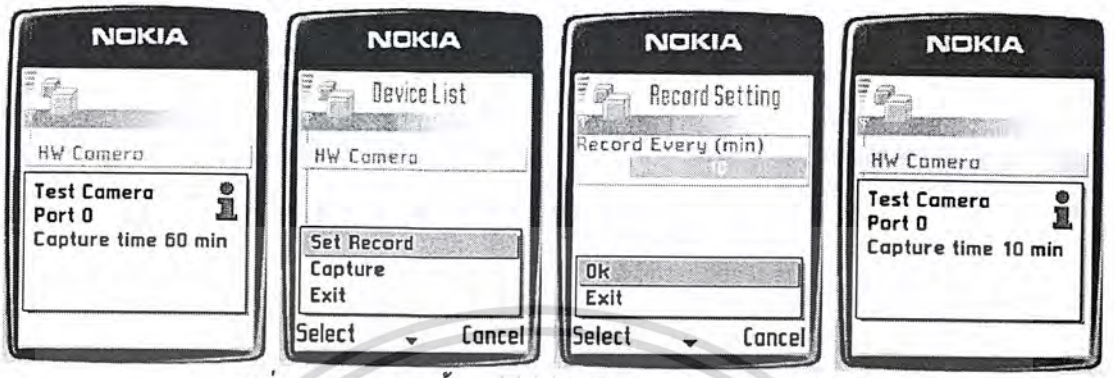


รูปที่ 4-55 แสดงการตั้งเวลาบันทึกภาพทุกๆ 15 นาทีหลังการกด Set

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.15.5 การทดสอบการตั้งเวลาการบันทึกภาพ ผ่านโทรศัพท์มือถือ

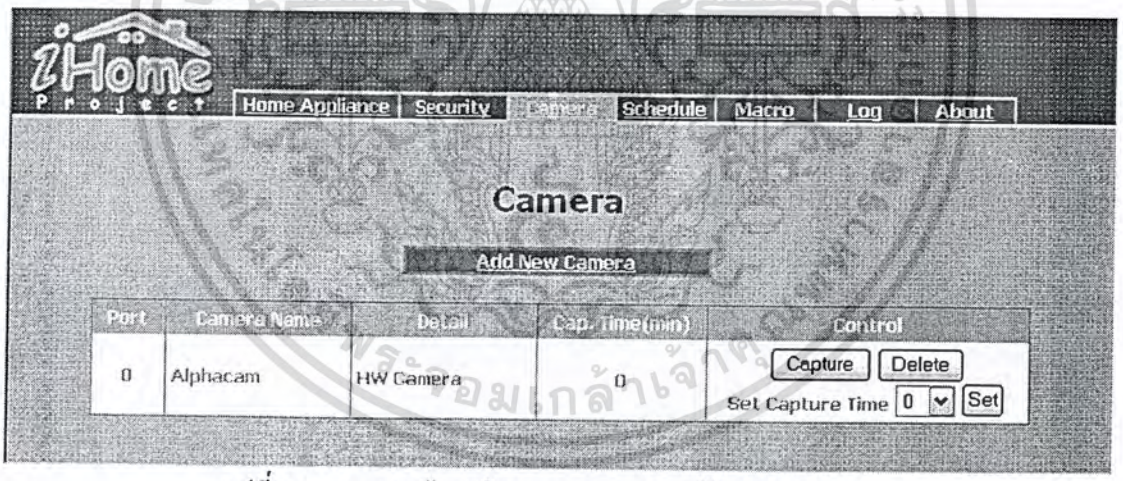
ให้เลือก Set Record เพื่อทำการปรับปรุงค่าเวลาในการบันทึกภาพ จากนั้นรอผลการตอบกลับมา จากรูปเป็นการตั้งเวลาการบันทึกภาพจาก 60 นาที ให้เป็น 10 นาที



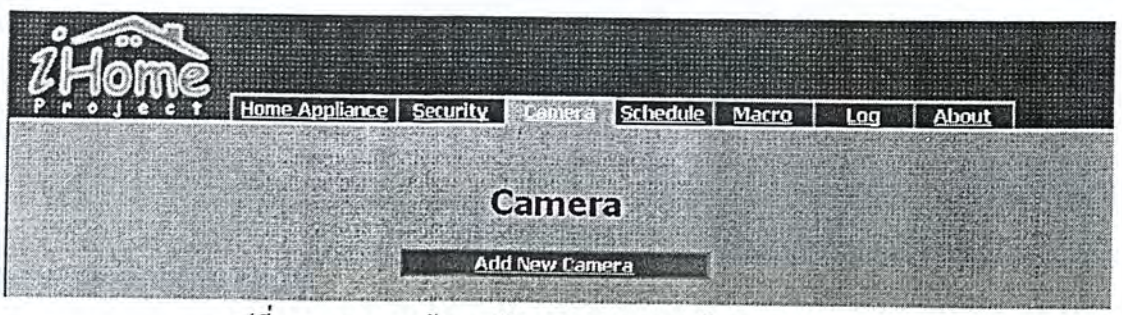
รูปที่ 4-56 แสดงการตั้งเวลาการบันทึกภาพผ่านโทรศัพท์มือถือ

4.15.6 การทดสอบการยกเลิกการใช้งานกล้อง

เมื่อต้องการยกเลิกการใช้งานกล้องสามารถทำได้โดยการกดปุ่ม Delete โดยจะทำการปรับปรุงฐานข้อมูล จากนั้นเมื่อทำการเรียกดูรายการกล้องอีกครั้งจะพบว่ารายการที่ต้องการลบได้หายไปแล้ว



รูปที่ 4-57 แสดงหน้าจอก่อนการลบรายการกล้องออกจากตาราง

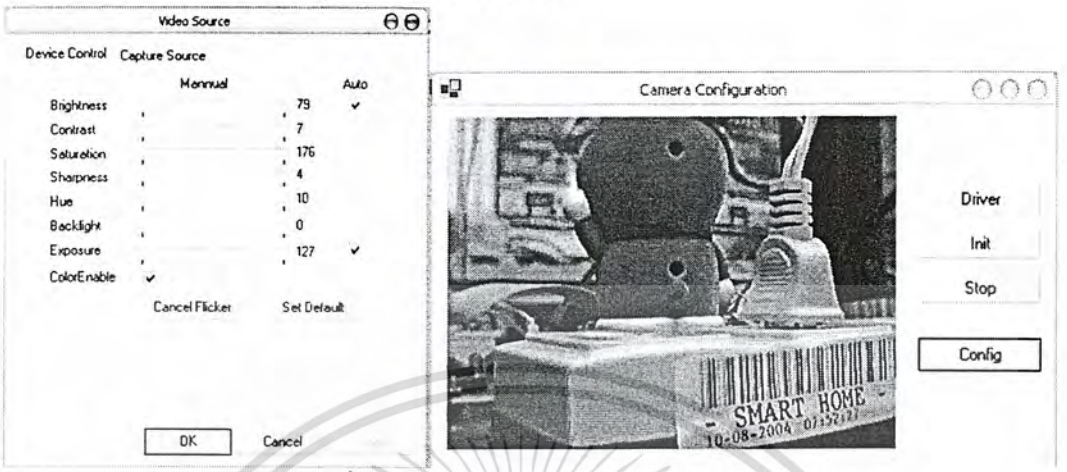


รูปที่ 4-58 แสดงหน้าจอหลังการลบรายการกล้องออกจากตาราง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.15.7 การทดสอบการตั้งค่ากล้อง

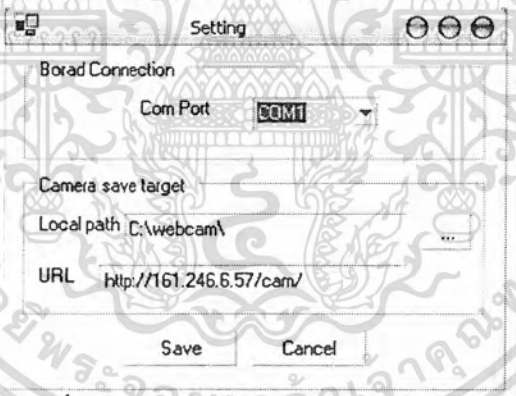
เป็นการปรับแต่งกล้องให้พร้อมใช้งาน ในด้านการบันทึกภาพ เช่นการปรับความเข้มของสี ความสว่างภาพภาพ หรือการปรับแต่งคุณสมบัติอื่นๆของกล้อง



รูปที่ 4-59 แสดงหน้าจอการกำหนดค่ากล้อง

4.16 การทดสอบการตั้งค่า Server

เป็นการปรับแต่งค่าให้ระบบ เพื่อให้สามารถทำงานได้ ประกอบไปด้วยการกำหนดพอร์ตในการติดต่อกับคอนโทรลบอร์ด การกำหนดที่ในการบันทึกข้อมูลภาพ การกำหนด URL ในการเข้าถึงภาพ



รูปที่ 4-60 แสดงหน้าจอกำหนดค่าเซิร์ฟเวอร์

4.17 การทดสอบการดูบันทึก Log

Log คือบันทึกการทำงาน เพื่อให้สามารถตรวจสอบได้ ลักษณะการบันทึกของ Log จะบันทึกต่อเนื่องในขณะที่ Server ยังทำงานอยู่ ถ้าเกิด Server เริ่มต้นทำงานใหม่ ค่าใน Log จะถูกลบทิ้งและเริ่มบันทึกใหม่ เนื่องจากค่าที่บันทึกนั้นไม่มีความสำคัญอะไรมากนัก

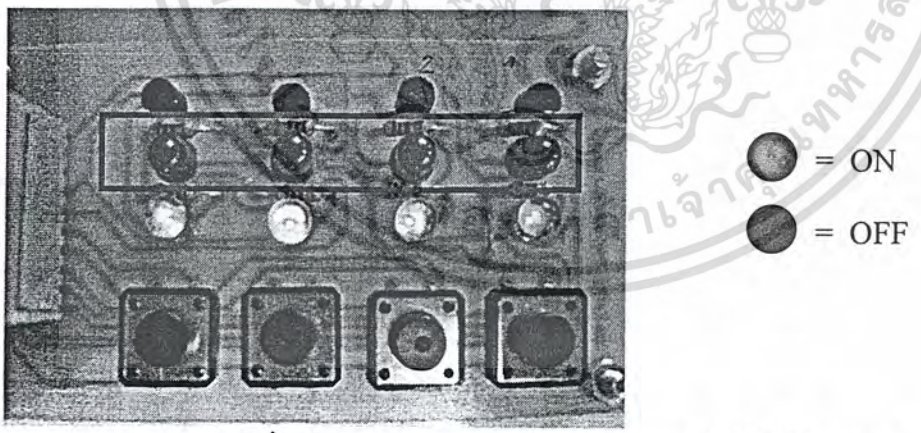
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4-61 แสดงหน้าจอผลการบันทึก Log การทำงานของ Server

4.18 การทดลองการโหลดค่าสถานะล่าสุดให้ Interface Module

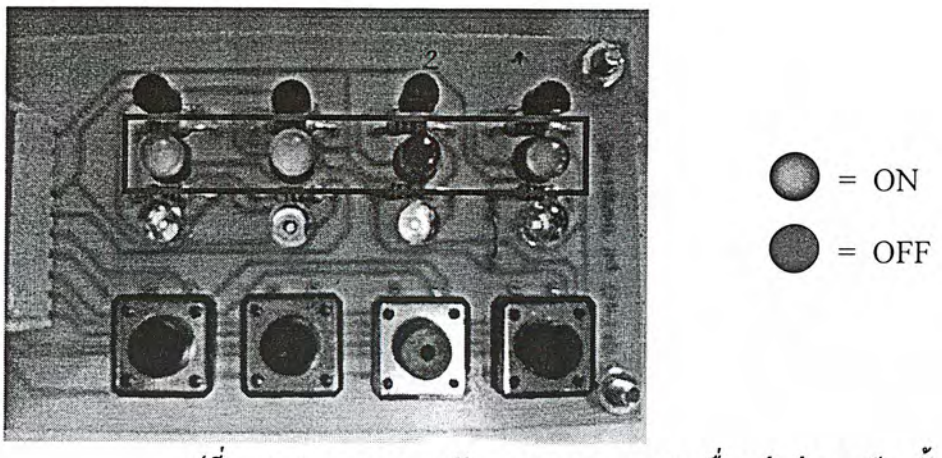
คือการนำค่าสถานะที่เก็บไว้ล่าสุดของ Interface Module ก่อนที่จะหลุดไปจากระบบ นำมากำหนดให้กับ Interface Module อีกครั้งหนึ่ง เมื่อ Interface Module เชื่อมต่อเข้ามาในระบบ

ภาพข้างล่างเป็นการทดลองโดยใช้ Switch Module ซึ่งขณะเชื่อมต่ออยู่มีสถานะคือ ปิด ปิด เปิด ปิด เรียงจากซ้ายไปขวา หลังจากนั้นทำให้หลุดการเชื่อมต่อ แล้ว เซตค่าเป็นสถานะเปิดทั้งหมด แล้วทำให้เชื่อมต่อมาที่ระบบใหม่



รูปที่ 4-62 แสดงสถานะตอน Switch Module ไม่ได้อยู่ในระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4-63 แสดงสถานะหลังจาก Switch Module เชื่อมต่อสู่ระบบอีกครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทวิจารณ์และสรุป

5.1 บทวิจารณ์และสรุป

ระบบรักษาความปลอดภัยและอำนวยความสะดวก (Smart Home) ที่ได้พัฒนาขึ้นนั้นสามารถทำงานได้ตรงตามแนวคิดที่ตั้งไว้ คือเป็นระบบที่เอื้ออำนวยให้อุปกรณ์เครื่องใช้ไฟฟ้าต่างๆสามารถสื่อสารและนำเอาข้อมูลนั้นไปใช้ประโยชน์ โดยสามารถจัดการเกี่ยวกับเหตุการณ์ที่เกิดขึ้นกับอุปกรณ์ตารางเวลาการทำงาน การควบคุมการทำงานผ่านทางเว็บบราวเซอร์ โทรศัพท์มือถือ หรือการแจ้งเตือนทาง SMS และเป็นไปตามเป้าหมายคือสามารถเฝ้าดูและรายงานความผิดปกติ โดยนำเอาอุปกรณ์ตรวจจับลักษณะต่าง ๆ เช่น ตรวจจับควัน ตรวจจับการผ่าน กล้องมาเชื่อมต่อเพื่อดูแลรักษาความปลอดภัย การควบคุมอุปกรณ์ไฟฟ้าให้ทำงานตามต้องการ

การทำงานของระบบโดยภาพรวมถ้าถือว่าการสื่อสารทางคลื่นวิทยุไม่มีปัญหาแล้ว ระบบจัดว่าสามารถทำงานได้ในสภาพที่เสถียร แต่ปัญหาหลักคือเรื่องการสื่อสาร เพราะในสภาวะแวดล้อมจริงๆ มีสิ่งรบกวนมากมาย ทำให้เกิดความผิดพลาดขึ้นในระดับที่สูง ทั้งจากคลื่นรบกวนจากภายนอก และคุณภาพโมดูลสื่อสารที่ใช้ ในระดับที่การควบคุมความผิดพลาดด้วยโปรโตคอลไม่สามารถทำได้ เนื่องจากข้อมูลที่ส่งนั้นมีขนาดที่ใหญ่ ส่งผลไปถึงระบบทำให้ทำงานได้ไม่เต็มประสิทธิภาพ ทางผู้พัฒนาได้พยายามหาอุปกรณ์ที่มีคุณภาพดีแล้ว แต่เนื่องด้วยอุปกรณ์นั้นๆ มีราคาที่ค่อนข้างสูง และจำเป็นต้องใช้หลายตัว จึงได้เปลี่ยนวิธีการทดสอบเป็นการใช้สายส่งสัญญาณต่อถึงกันระหว่างแต่ละ Interface Module ผลคือระบบสามารถทำงานได้อย่างดี

ระบบที่สร้างขึ้นนี้สามารถติดต่อใช้งานผ่านเว็บบราวเซอร์ และโทรศัพท์มือถือ ซึ่งอำนวยความสะดวกในการใช้งานจากระยะไกล และมี Interface Module ที่สามารถใช้งานทั้งเป็นแบบ Stand Alone หรือแบบถูกควบจากโปรแกรมได้ ทำให้แม้ไม่มีโปรแกรมควบคุมก็ยังคงทำงานได้ และมีความสามารถในการค้นหาอุปกรณ์อัตโนมัติ ทำให้สามารถใช้งานอุปกรณ์ได้ทันที โดยไม่ต้องเพิ่มอุปกรณ์เข้าไปเองเหมือนระบบ Smart Home อื่นๆ การรักษาค่าปัจจุบันของอุปกรณ์ ถึงแม้อุปกรณ์จะหลุดการเชื่อมต่อหรือไม่มีไฟเลี้ยง แต่เมื่อกลับเข้ามาในระบบใหม่ สถานะก่อนหน้าจะถูกโหลดขึ้นไปให้อุปกรณ์โดยอัตโนมัติ การควบคุมปิดเปิด ระดับความสว่าง การแจ้งเตือน ตารางเวลา Event & Macro หรือการโปรแกรมการทำงานที่ตอบสนองต่อเหตุการณ์ โดยฟังก์ชันก็กล่าวมาทั้งหมดนี้เป็นฟังก์ชันการทำงานที่เป็นจุดเด่นของระบบ

โดยสรุปรวมแล้วปัญหาที่เกิดขึ้นส่วนใหญ่เป็นปัญหาด้านการสื่อสารแบบไร้สาย ซึ่งการนำเอาอุปกรณ์ที่มีประสิทธิภาพดีมาใช้ย่อมมีค่าใช้จ่ายที่สูงตามตัว และในส่วนของปัญหาอื่นๆในระบบ มีรายละเอียดดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 ปัญหาที่เกิดและแนวทางการแก้ไข

1. สัญญาณรบกวนการสื่อสาร เนื่องจากใช้คลื่นวิทยุ ทำให้ยากต่อการออกแบบและการสร้างต้นแบบทั้งในด้านซอฟแวร์คือโปรโตคอลและฮาร์ดแวร์ ซึ่งแนวทางแก้ไขปัญหานี้คือการใช้โมดูลรับส่งด้วยคลื่นวิทยุที่มีประสิทธิภาพสูงขึ้น
2. ความแตกต่างกันของเทคโนโลยีต่างค่าย คือ ดอทเน็ตของไมโครซอฟต์และจาวา ทำให้การพัฒนามีปัญหาด้านเทคนิค และล่าช้ากว่าการใช้พัฒนาโดยใช้เทคโนโลยีเดียวกันไปบ้าง แต่ด้วยวิธีการสื่อสารโดยใช้โปรโตคอลที่เป็นมาตรฐานกลางเดียวกัน ทำให้ปัญหานี้เป็นเรื่องที่สามารถแก้ได้โดยไม่ยากเย็นนัก
3. การประมวลผลกับบอร์ดควบคุมที่มีหน้าที่สื่อสารกับอุปกรณ์ไฟฟ้าเป็นการประมวลผลทีละคำสั่ง ซึ่งอัตราการตอบสนองค่อนข้างช้า ทำให้ถ้าผู้ใช้บริการมีจำนวนมากจะทำให้เกิดการรอนานจนเกิดไทม์เอาท์ (Timeout) ได้ แนวทางการแก้ไขคือใช้โมดูลรับส่งด้วยคลื่นวิทยุที่มีประสิทธิภาพสูงขึ้นและปรับปรุงโปรโตคอลไม่ให้เป็นแบบหยุดรอคอย (Stop and wait protocol)
4. หน้าจอการแสดงผลของโทรศัพท์มือถือแต่ละรุ่นไม่เหมือนกัน ทำให้เมื่อนำโปรแกรมไปใช้อาจมีปัญหาในการแสดงผล ซึ่งแนวทางแก้ปัญหาคือเขียนโปรแกรมโดยใช้รูปแบบ GUI มาตรฐาน แต่รูปแบบที่ได้จะไม่สวยงามนัก
5. กล้องที่ใช้ในการจับภาพ สามารถใช้ได้เพียง 1 ตัวเท่านั้นในระหว่างที่ทำงานอยู่ เนื่องจาก library ที่ใช้ในการจับภาพไม่สามารถแก้ไขกล้องที่ใช้โดยการเขียนโปรแกรมไปควบคุมได้ จำเป็นต้องเรียกฟอร์มหน้าจอควบคุมของ library นั้นมาแสดง ซึ่งในระบบนี้ต้องการให้ผู้ใช้ควบคุมได้ผ่านเว็บเพจ จึงไม่สามารถทำให้ใช้งานได้พร้อมกันหลายๆตัว ทางผู้พัฒนาได้ลองติดต่อโดยใช้ DirectShow ซึ่งสามารถควบคุมได้ละเอียดกว่า แต่มีปัญหาเรื่อง Component COM+ ของ DirectShow กับการเรียกใช้งานผ่านการ remote
6. ในส่วนของการเนื้อที่เก็บภาพที่เก็บจากการถ่ายจากกล้องนั้น จะต้องดูแลเรื่องขนาดของพื้นที่ที่ใช้ในการเก็บ ไม่ให้ไปรบกวนเนื้อที่ที่ระบบปฏิบัติการทำงาน เนื่องจากโปรแกรมจะบันทึกข้อมูลไปเรื่อยๆ ถ้าผู้ใช้ไม่ดูแลและปล่อยทิ้งไว้เป็นระยะเวลาที่ยาวนาน อาจจะทำให้ระบบปฏิบัติการไม่มีพื้นที่เพียงพอให้ใช้ในการทำงานได้

5.3 แนวทางพัฒนาต่อ

ระบบที่ได้กล่าวไว้ในปริญญานิพนธ์นี้ แกนหลักคือระบบควบคุมที่ให้บริการผ่านเครือข่ายด้วยเว็บเซอร์วิส ซึ่งสามารถนำเอาแนวคิดนี้ไปสร้างเป็นบริการเพิ่มเติมได้อีกในอนาคต โดยการพัฒนาต่อนั้นมีได้หลายแนวทาง ไม่ว่าจะเป็นทางด้านฮาร์ดแวร์ ระบบควบคุม ด้านความปลอดภัย ฯลฯ ตัวอย่างเช่น ระบบควบคุมที่มีความซับซ้อน ระบบที่มีการตอบสนองผู้ใช้ได้ง่าย ฮาร์ดแวร์ที่ช่วยอำนวยความสะดวก หรือการเชื่อมโยงกันเป็นเครือข่ายเพื่อการให้บริการ ซึ่งผู้จัดทำขอแนะนำไว้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. การทำระบบอัตโนมัติให้ตอบสนองอย่างเหมาะสมกับพฤติกรรมของผู้อยู่อาศัย โดยอาจใช้การวิเคราะห์ทางสถิติ
2. การทำระบบควบคุมสภาวะแวดล้อมในบ้าน เช่น อุณหภูมิ แสง เสียง
3. การทำระบบสื่อสารแบบมัลติมิเดียภายในบ้าน

4. ระบบให้บริการกลาง ซึ่ง Smart Home แต่ละบ้านจะสามารถเชื่อมต่อบริการไปยังศูนย์ ทั้งนี้อาจจะเป็นระบบรักษาความปลอดภัยศูนย์กลาง ที่อาจจะเป็นศูนย์ดูแลความปลอดภัยของหมู่บ้านเป็นผู้ให้บริการ ซึ่งกล่าวถึงตอนต้นของปริญญานิพนธ์ หรือบริการของร้านค้าต่าง ๆ เช่นบริการรับสั่งสินค้า ทั้งนี้อาจนำเอาไปผนวกเข้ากับ sensor เพื่อการสั่งงานแบบอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] มณีโชติ สมานไทย (2546) : "การเขียนโค้ด ASP.NET ฉบับสมบูรณ์", อินโฟเพรส, 2546

[2] Andrew S. Tanenbaum (1999) : "Computer Networks",Pearson Education Indochina, 1999, pp.140-191

[3] สราวุธ อ้อยศรีสกุล(2544) : "ถอดรหัส .NET+Web Services", วิตตี้ กรุ๊ป, 2544

[4] Mark Schmidt : "Microsoft Visual C# .NET 2003"

[5] ศิวณัฏฐ์ มาศสุรางค์ (2546) : "J2ME", เอ.อาร์.อินฟอเมชั่น แอน พับลิเคชั่น, 2546

[6] ทรงเกียตรติ ภาวดี (2546) : "เก่ง J2ME ครบสูตร",วิตตี้ กรุ๊ป, 2546

[7] พอ.เจนวิทย์ เหลืองอร่าม (2546) : "การเขียนโปรแกรมสำหรับ Wireless Application ด้วย J2ME", ซีเอ็ดยูเคชั่น, 2546

[8] เซมิคอนดักเตอร์ อิเล็กทรอนิกส์ ฉบับที่ 162, pp.38-44

[9] เซมิคอนดักเตอร์ อิเล็กทรอนิกส์ ฉบับที่ 243, pp.29-32

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้