สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

# การควบคุมอุปกรณ์ไฟฟ้าภายในบ้านผ่านหลายช่องทาง

# HOME ELECTRIC DEVICES CONTROL VIA MULTI INTERFACE

โดย

นายปริพล ทู้ไพเราะห์

นายไพโรจน์ ศักดิ์ศรยุทธ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2544

เลขหมู่.....
เลขทะเบียน 46438
วัน, เดือน, ปี - 1 เม.ย. 2546

.b.....

.i.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | |
|---|---|
| **หัวข้อปริญญานิพนธ์** | การควบคุมอุปกรณ์ไฟฟ้าภายในบ้านผ่านหลายช่องทาง<br>HOME ELECTRIC DEVICES CONTROL VIA MULTI INTERFACE |
| **ชื่อนักศึกษา** | นายปริพล ทู้ไพเราะห์ รหัสประจำตัว 41014260<br>นายไพโรจน์ ศักดิ์ศรยุทธ รหัสประจำตัว 41014322 |
| **อาจารย์ผู้ควบคุมปริญญานิพนธ์** | อาจารย์มยุรี เลิศเวชกุล<br>อาจารย์บุณย์ชนะ ภู่ระหงษ์ |
| **ระดับการศึกษา** | ปริญญาวิศวกรรมศาสตรบัณฑิต<br>สาขาวิศวกรรมสารสนเทศ |
| **ภาควิชา** | วิศวกรรมสารสนเทศ |
| **ปีการศึกษา** | 2544 |

ปริญญานิพนธ์ฉบับนี้ได้รับความเห็นชอบจากอาจารย์ที่ปรึกษาเป็นที่เรียบร้อยแล้ว

(อาจารย์มยุรี เลิศเวชกุล)
**อาจารย์ผู้ควบคุมปริญญานิพนธ์**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | |
|---|---|
| **หัวข้อปริญญานิพนธ์** | การควบคุมอุปกรณ์ไฟฟ้าภายในบ้านผ่านหลายช่องทาง |
| **โดย** | นายปริพล ทู้ไพเราะห์ |
| | นายไพโรจน์ ศักดิ์ศรยุทธ |
| **ภาควิชา** | วิศวกรรมสารสนเทศ |
| **อาจารย์ที่ปรึกษา** | อาจารย์มยุรี เลิศเวชกุล |
| | อาจารย์บุณย์ชนะ ภู่ระหงษ์ |


**บทคัดย่อ**

ปริญญานิพนธ์ฉบับนี้ทำขึ้นเพื่อศึกษาถึงการนำเอาเทคโนโลยีสมัยใหม่มาประยุกต์เป็นช่องทางเพื่อควบคุมอุปกรณ์ไฟฟ้าภายในบ้าน โดยโครงงานนี้จะทำการศึกษาการควบคุมผ่านทางอินเทอร์เนตโดยใช้ Web application การควบคุมผ่านทางโทรศัพท์เคลื่อนที่ด้วย Wap application และการควบคุมด้วยเสียงพูดผ่านโปรแกรม Microsoft Speech API สำหรับการควบคุมผ่านทาง Web Application และ Wap Application นั้น server ทำการติดต่อไปยัง ISP โดยอัตโนมัติ ดังนั้น EDCS (Electric Devices Control Server) จึงไม่จำเป็นต้องเชื่อมต่อกับอินเทอร์เนตอยู่ตลอดเวลา ซึ่งเป็นการช่วยประหยัดค่าใช้จ่ายเช่น ค่าชั่วโมงอินเทอร์เนต อีกทั้งยังสามารถกำหนดเวลาการทำงานของอุปกรณ์ไฟฟ้า แต่ละชนิดได้ อุปกรณ์ไฟฟ้าจะถูกควบคุมด้วยคอนโทรลเลอร์ตระกูล MCS-51 model 89C2051 ทำการรับคำสั่งจากพอร์ต RS232 ของเครื่อง server มาทำการเปิด-ปิดอุปกรณ์ไฟฟ้าภายในบ้าน


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | | |
|---|---|---|
| **PROJECT REPORT** | HOME ELECTRIC DEVICES CONTROL VIA MULTI INTERFACE | |
| **BY** | MR. PARIPHOL | TOOPAIROOH |
| | MR. PAIROJ | SAKSORNYUTH |
| | DEPARTMENT OF | INFORMATION ENGINEERING |
| **ADVISOR** | MS. MAYUREE | LERTWATECHAKUL |
| | MR. BOONCHANA | PHOORAHONG |


## ABSTRACT

This thesis objective is to study about applying a new technology to be an interface that control home electric devices. We choose 3 ways of interfaces. First, controlling via web application. Second, controlling via wap application. Third, controlling via voice recognition. Electric Devices Control Server is designed to dial to ISP automatically when ringing tone is sensed by attached modem and then the server will update current IP at free host's database so it is not necessary to online on the internet all the time. The advantage is internet cost saving. User cound schedule operating time of electric devices those were controlled by microcontroller family MCS-51 model AT89C2051. The microcontroller receives command from the server through RS232 to switch on or switch of home electric devices.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้ สำเร็จลุล่วงได้เนื่องจากการสนับสนุนและให้กำลังใจจากบิดามารดา และด้วยความร่วมมือและความตั้งใจ ในการทำงานของเพื่อนร่วมงานในกลุ่มและได้รับคำแนะนำจากอาจารย์ที่ปรึกษาทางด้านเทคนิคและแนวความคิดต่าง ๆ นอกจากนี้ยังได้รับคำแนะนำในการออกแบบวงจรจาก พี่สิริชัย ธรรมารักษ์วัฒนะ ที่สำนักวิจัยฯ ทางคณะผู้จัดทำใคร่ขอขอบคุณมา ณ ที่นี้

นายปริพล ทู้ไพเราะห์
นายไพโรจน์ ศักดิ์ศรยุทธ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ


| เรื่อง | หน้า |
|---|---|
| บทคัดย่อภาษาไทย | I |
| บทคัดย่อภาษาอังกฤษ | II |
| กิตติกรรมประกาศ | III |
| สารบัญ | IV-V |
| สารบัญรูป | VI-IX |
| บทที่ 1 บทนำ | |
| 1.1 ความเป็นมาของปริญญานิพนธ์ | 1 |
| 1.2 จุดมุ่งหมายของปริญญานิพนธ์ | 1 |
| 1.3 แนวคิดและที่มา | 1 |
| 1.4 ขอบเขตของโครงงาน | 2 |
| บทที่ 2 เทคโนโลยีที่ใช้ในโครงงาน | |
| 2.1 Java Servlet Technology | 3 |
| 2.1.1 Introduction to Java Servlet Technology | 3 |
| 2.1.2 สถาปัตยกรรมของ Servlet | 4 |
| 2.2 WAP Technology | 4 |
| 2.2.1 Introduction to WAP Technology | 4 |
| 2.2.2 สถาปัตยกรรมของ WAP | 5 |
| 2.2.3 โมเดลการทำงานของ WEB และ WAP | 7 |
| 2.3 Microsoft Speech Technology | 9 |
| 2.3.1 Introduction to Microsoft Speech Technology | 9 |
| 2.3.2 การพัฒนาแอปพลิเคชั่นด้วย Speech SDK | 13 |
| บทที่ 3 การสื่อสารข้อมูล | |
| 3.1 การสื่อสารแบบขนาน | 15 |
| 3.2 การสื่อสารแบบอนุกรม | 16 |
| 3.3 การอินเตอร์เฟส | 19 |
| 3.4 การรับส่งข้อมูลสองทิศทาง | 22 |
| 3.5 ลักษณะการเชื่อมต่อและหน้าที่การทำงานของแต่ละขาที่สำคัญ | 27 |


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ


| เรื่อง | หน้า |
|---|---|
| **บทที่ 4 ฮาร์ดแวร์** | |
| 4.1 อุปกรณ์ที่ใช้ | 29 |
| 4.2 คุณสมบัติของ AT89C2051 | 29 |
| 4.3 แหล่งจ่ายไฟ | 32 |
| 4.4 วงจรรีเลย์ | 32 |
| 4.5 แหล่งจ่ายไฟสำหรับวงจรรีเลย์ | 35 |
| **บทที่ 5 การออกแบบ** | |
| 5.1 ภาพโดยรวมของระบบ | 36 |
| 5.2 การออกแบบระบบ | 36 |
| **บทที่ 6 การทดลอง** | |
| 6.1 การทดลอง | 48 |
| 6.2 สรุปผลการทดลอง | 68 |
| **เอกสารอ้างอิง** | 69 |
| **ภาคผนวก ก. คำสั่งพื้นฐาน AT** | 70 |
| **ภาคผนวก ข. การติดตั้งโปรแกรม** | 77 |
| **ภาคผนวก ค. รายละเอียดของวงจร** | 84 |


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญรูป


| รูปที่ | | หน้า |
|---|---|---|
| 2.1 | รูปแสดงการทำงานของ Servlet | 3 |
| 2.2 | สถาปัตยกรรมของ WAP | 5 |
| 2.3 | เปรียบเทียบเลเยอร์ของ WAP กับอินเทอร์เน็ต | 6 |
| 2.4 | โมเดลการทำงานของ WEB | 7 |
| 2.5 | โมเดลการทำงานของ WAP | 7 |
| 2.6 | การเชื่อมต่อ WAP ผ่าน ISP | 8 |
| 2.7 | การเชื่อมต่อ WAP ในลักษณะส่วนตัวหรือองค์กร | 8 |
| 2.8 | แสดงขั้นตอนการวิเคราะห์จดจำเสียง | 10 |
| 2.9 | แสดงขั้นตอนการสังเคราะห์เสียงพูดจากข้อความ | 12 |
| 2.10 | แสดงส่วนประกอบของ Microsoft Speech API 4.0 | 13 |
| 3.1 | แสดงการสื่อสารแบบขนาน | 16 |
| 3.2 | แสดงการสื่อสารข้อมูลแบบอนุกรม | 17 |
| 3.3 | แสดงอุปกรณ์ DTE เบื้องต้น | 21 |
| 3.4 | แสดงอุปกรณ์ทั้ง DTE และ DCE | 21 |
| 3.5 | แสดงอุปกรณ์ DTE และ DCE ซึ่งเป็นคู่อุปกรณ์ที่ทำงานตรงข้ามกัน | 22 |
| 3.6 | การทำงานของอุปกรณ์ที่สามารถส่งและรับข้อมูลได้ทั้งสองทิศทาง | 22 |
| 3.7 | แสดงคำจำกัดความค่าตรรกะที่เอาต์พุตของ RS-232 | 24 |
| 3.8 | แสดงคำจำกัดความค่าตรรกะที่อินพุตของ RS-232 | 24 |
| 3.9 | แสดงการเชื่อมต่อทั้งแบบ 25 Pin และ 9 Pin | 27 |
| 4.1 | แหล่งจ่ายไฟวงจรไมโครคอนโทรลเลอร์ | 30 |
| 4.2 | แสดงวงจรไมโครคอนโทรลเลอร์ | 31 |
| 4.3 | แหล่งจ่ายไฟวงจรไมโครคอนโทรลเลอร์ | 32 |
| 4.4 | แสดงวงจรรีเลย์ | 33 |
| 4.5 | แสดงวงจรรีเลย์ | 34 |
| 4.6 | แหล่งจ่ายไฟวงจรรีเลย์ | 35 |
| 5.1 | Diagram แสดงภาพโดยรวมของโครงงาน | 36 |
| 5.2 | ขั้นตอนการดำเนินการของ server เพื่อเชื่อมต่อเข้าสู่อินเทอร์เนต | 37 |
| 5.3 | ขั้นตอนการเชื่อมต่อกับ Home server ผ่าน Free host เพื่อควบคุมอุปกรณ์ไฟฟ้า | 37 |


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญรูป


| รูปที่ | | หน้า |
|---|---|---|
| 5.4 | ขั้นตอนการทำงานของระบบควบคุมอุปกรณ์ไฟฟ้าทางเสียงพูด | 38 |
| 5.5 | ขั้นตอนการทำงานของโปรแกรม ModemListener | 39 |
| 5.6 | ขั้นตอนการทำงานของโปรแกรม CheckSchedule | 40 |
| 5.7 | แสดงภาพโดยรวมของ Visual 3D Interface | 41 |
| 5.8 | ขั้นตอนการทำงานของ Web Application | 42 |
| 5.9 | แสดงภาพโดยรวมของ Mobile Interface | 43 |
| 5.10 | ขั้นตอนการทำงานของ Wap Application | 45 |
| 5.11 | แสดงภาพโดยรวมของ Voice Interface | 46 |
| 5.12 | ขั้นตอนการทำงานของ Voice Control Application | 47 |
| 6.1 | การทำงานของโปรแกรม CheckSchedule | 48 |
| 6.2 | ส่วนการรอรับสัญญาณจาก Modem ที่ COM1 | 48 |
| 6.3 | แสดงการทำงานเมื่อรับสัญญาณ Ringing tone ครบ 6 ครั้ง | 49 |
| 6.4 | แสดงการติดต่อไปยัง ISP แล้วทำการ Update current IP | 49 |
| 6.5 | แสดงการรอรับ request ของ Home Web Server | 49 |
| 6.6 | แสดงลิงค์บน Webpage ของ Free host | 50 |
| 6.7 | แสดงผลลัพธ์เมื่อกดปุ่ม Run บน Webpage | 50 |
| 6.8 | แสดงผลลัพธ์เมื่อกดปุ่ม Add บน Webpage | 51 |
| 6.9 | แสดงผลลัพธ์เมื่อเลือกลิงค์ View บน Webpage | 51 |
| 6.10 | แสดงผลลัพธ์เมื่อเลือกลิงค์ update บน Webpage | 52 |
| 6.11 | แสดงผลลัพธ์เมื่อกดปุ่ม History บน Webpage | 52 |
| 6.12 | แสดงผลลัพธ์เมื่อกดปุ่ม Config บน Webpage | 53 |
| 6.13 | แสดงภาพการใช้เมาส์เคลื่อนไปภายในบ้าน | 53 |
| 6.14 | แสดงลิงค์บนหน้า Wap page ของ Free host | 54 |
| 6.15 | แสดงรายการเมนูหลักของ Wap page | 54 |
| 6.16 | แสดงการเลือกรายการอุปกรณ์ไฟฟ้าภายในเมนู Command | 55 |
| 6.17 | แสดงผลลัพธ์เมื่อเลือก Send ภายในเมนู Command | 55 |
| 6.18 | แสดงผลลัพธ์เมื่อเลือกเมนูหลัก Current Status | 56 |
| 6.19 | แสดงรายการเมนูย่อยเมื่อเลือกเมนูหลัก Schedule | 56 |


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญรูป


| รูปที่ | | หน้า |
|---|---|---|
| 6.20 | แสดงฟอร์มสำหรับตั้งค่า เมื่อเลือกเมนูย่อย Add Schedule | 57 |
| 6.21 | แสดงผลลัพธ์เมื่อเลือก Add ภายในเมนูย่อย Add Schedule | 57 |
| 6.22 | แสดงผลลัพธ์เมื่อเลือกเมนูย่อย View Schedule | 58 |
| 6.23 | แสดงผลลัพธ์เมื่อเลือกลิงค์รายการภายในเมนูย่อย View Schedule | 58 |
| 6.24 | แสดงฟอร์มสำหรับเปลี่ยนแปลงค่าเมื่อเลือก Edit ภายในเมนูย่อย View Schedule | 59 |
| 6.25 | แสดงลำดับขั้นการเปลี่ยนแปลงการตั้งค่าและทำการเลือก Update | 59 |
| 6.26 | แสดงลำดับขั้นการเลือก Delete เพื่อทำการลบรายการที่ได้ตั้งไว้ | 60 |
| 6.27 | แสดงผลลัพธ์เมื่อทำการ Delete รายการแล้ว | 60 |
| 6.28 | แสดงผลลัพธ์เมื่อเลือกเมนูหลัก View History | 61 |
| 6.29 | แสดงรายการอุปกรณ์ไฟฟ้าเมื่อเลือกลิงค์ ภายในส่วน View History | 61 |
| 6.30 | แสดงหน้าจอหลักของ Voice Control Application | 62 |
| 6.31 | แสดงผลลัพธ์เมื่อพูด “On TV1” | 62 |
| 6.32 | แสดงผลลัพธ์เมื่อทำการพูด “Off TV1” | 63 |
| 6.33 | แสดงหน้าจอ Config เพื่อทำการกำหนดค่าให้แต่ละสวิตช์ | 63 |
| 6.34 | แสดงผลลัพธ์ที่ได้เมื่อทำการ Update ค่าอุปกรณ์ | 64 |
| 6.35 | แสดงผลลัพธ์เมื่อทำการพูด “On fan9” | 64 |
| 6.36 | แสดงผลลัพธ์เมื่อทำการพูด “Off fan9” | 65 |
| 6.37 | แสดงหน้าจอการตั้งวัน-เวลา และอุปกรณ์ไฟฟ้า | 65 |
| 6.38 | แสดงผลลัพธ์เมื่อกดปุ่ม Add ภายในส่วน Schedule | 66 |
| 6.39 | แสดงผลลัพธ์เมื่อเลือกรายการภายในส่วน Schedule List | 66 |
| 6.40 | แสดงผลลัพธ์เมื่อกดปุ่ม Update ภายในส่วน Schedule | 67 |
| 6.41 | แสดงผลลัพธ์เมื่อกดปุ่ม Delete ภายในส่วน Schedule | 67 |
| 6.42 | แสดงรายการคำสั่งที่ได้ทำไปแล้ว เมื่อกดปุ่ม History ที่หน้าจอหลัก | 68 |
| ข.1 | แสดงรายชื่อไฟล์ภายใน directory c:\jakarta-tomcat-3.2.2 | 78 |
| ข.2 | แสดงขั้นตอนการติดตั้ง Java 2 sdk | 79 |
| ข.3 | แสดงรายชื่อไฟล์ภายใน directory c:\commapi | 79 |
| ข.4 | แสดงการ copy ไฟล์ comm..jar และ javax.comm.properties | 80 |
| ข.5 | แสดงการ copy ไฟล์ win32com.dll | 80 |


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญรูป


| รูปที่ | | หน้า |
|---|---|---|
| ข.6 | แสดงการ unzip ไฟล์ Utility.zip | 81 |
| ข.7 | แสดงการติดตั้ง Microsoft Speech API | 81 |
| ข.8 | แสดงการติดตั้ง Adobe Atmosphere Plug-in | 82 |
| ข.9 | แสดงการติดตั้งโปรแกรม Voice Control | 82 |
| ข.10 | แสดงการติดตั้ง Adobe Atmosphere Plug-in | 83 |
| ค.1 | แสดงสัญลักษณ์ตัวอุปกรณ์ | 85 |
| ค.2 | แสดงลายวงจร Supply | 85 |
| ค.3 | แสดงลายวงจรรีเลย์ | 86 |
| ค.4 | แสดงสัญลักษณ์อุปกรณ์วงจรไมโครคอนโทรลเลอร์ | 87 |
| ค.5 | แสดงลายวงจรไมโครคอนโทรลเลอร์(ด้านบน) | 88 |
| ค.6 | แสดงลายวงจรไมโครคอนโทรลเลอร์(ด้านล่าง) | 89 |


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 1
# บทนำ

## 1.1 ความเป็นมาของปริญญานิพนธ์

โครงงานนี้ได้นำแนวคิดมาจากโครงงานเดิม ที่เป็นการควบคุมอุปกรณ์ไฟฟ้าภายในบ้านผ่านทาง Internet โดยได้นำมาพัฒนาแอพพลิเคชั่นให้มีอินเทอร์เฟซในการใช้งานได้หลากหลายชนิด ได้แก่ VISUAL 3D INTERFACE , MOBILE DEVICE INTERFACE , VOICE INTERFACE

## 1.2 จุดมุ่งหมายของปริญญานิพนธ์

เพื่อศึกษาการนำเอาเทคโนโลยีปัจจุบันมาประยุกต์ใช้ เป็นช่องทางในการควบคุมอุปกรณ์ไฟฟ้าภายในบ้าน ได้อย่างสะดวก รวดเร็ว และประหยัดค่าใช้จ่าย และมีส่วนแสดงผลที่น่าสนใจและมีความหลากหลายในด้านการใช้งาน

## 1.3 แนวความคิดและที่มา

เนื่องจากในอดีตการควบคุมอุปกรณ์ไฟฟ้าภายในบ้านมีข้อจำกัดอยู่เฉพาะภายในบริเวณบ้านและบริเวณที่สัญญาณรีโมทไปถึงเท่านั้น ไม่สามารถควบคุมจากระยะไกลได้ จึงได้มีแนวคิดที่จะนำเอาระบบ INTERNET ซึ่งเป็นระบบเครือข่ายที่เชื่อมโยงไปทั่วโลกมาช่วยในการควบคุมอุปกรณ์ไฟฟ้าภายในบ้าน ทำให้สามารถเข้าถึงได้หลายทางและหลายรูปแบบเช่น ผ่านทาง Web Application หรือผ่านทางโทรศัพท์เคลื่อนที่ Wap ทำให้สามารถควบคุมอุปกรณ์ได้สะดวกในเวลาที่เร่งด่วน และสามารถตั้งเวลาการ เปิด-ปิด อุปกรณ์ไฟฟ้าได้ตามต้องการ หรือแม้แต่การสั่งงานผ่านทางเสียง (VOICE CONTROL) ซึ่งช่วยอำนวยความสะดวกมากยิ่งขึ้นในขณะที่ผู้ใช้อยู่ภายในบ้าน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงงานประกอบด้วยส่วนสำคัญ 2 ส่วนใหญ่ ๆ คือ

HARDWARE

- อุปกรณ์ที่ต่อกับ COMPUTER SERVER ผ่านทางพอร์ตอนุกรมและไปควบคุมการสวิตช์เปิด-ปิด อุปกรณ์ไฟฟ้า

SOFTWARE

- โปรแกรมควบคุมการรับสัญญาณโทรศัพท์และทำการเชื่อมต่อINTERNETโดยอัตโนมัติ
- โปรแกรมควบคุมการ ปิด-เปิดอุปกรณ์ไฟฟ้าโดยอัตโนมัติ
- โปรแกรม Web Application
- โปรแกรม Wap Application
- โปรแกรม Voice Control Application

## 1.4 ขอบเขตของโครงงาน

โครงงานนี้ได้แนวคิดมาจาก โครงงานเก่าที่ได้พัฒนาอุปกรณ์ Hardware และ Software ที่ใช้ควบคุมอุปกรณ์ไฟฟ้าผ่าน Internet ซึ่งยังมีข้อจำกัดอยู่ที่จะต้องมีการเชื่อมต่ออินเทอร์เนตตลอดเวลาคือมีการใช้ static IP ซึ่งในการใช้งานนั้นจะมีการเรียกใช้งานเป็นบางเวลา จึงไม่เหมาะที่จะมีการเชื่อมต่ออินเทอร์เนตตลอดเวลา อีกทั้งยังมี Interface ที่เป็น web browser เพียงอย่างเดียว

โครงงานนี้ได้มีการพัฒนาให้สามารถใช้ โทรศัพท์เรียกผ่านโมเดมไปที่ server ที่บ้านให้ทำการเชื่อมต่ออินเทอร์เนตโดยอัตโนมัติ และนำ dynamic IP ใหม่ที่ได้จากการเชื่อมต่ออินเทอร์เนตในแต่ละครั้งมาทำการเก็บไว้ที่ free host's database แล้วจึงทำการเรียกใช้งาน web page ของ free host's database ที่จะมีลิงค์เชื่อมต่อไปยัง IP ของเครื่อง server อีกทีหนึ่ง และมีการเพิ่ม Interface ขึ้นมาอีกหลายทาง เช่น mobile device interface , voice interface

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 2
# เทคโนโลยีที่ใช้ในโครงงาน

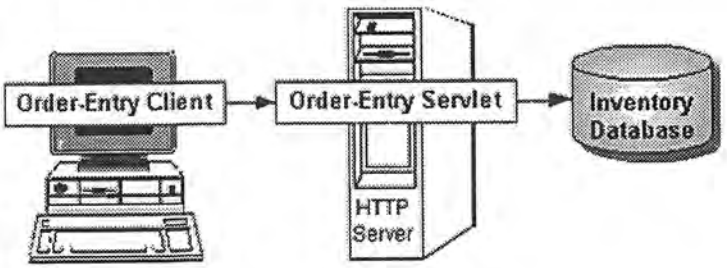
## 2.1 Java Servlet Technology

### 2.1.1 Introduction to Java Servlet Technology

Java Servlet คือ คือเทคโนโลยี ที่เกิดขึ้นมาจาก Java เพื่อใช้งานทางด้านการสร้างโปรแกรมในรูปแบบ CGI สำหรับทำงานบนเว็บเซิร์ฟเวอร์ เพื่อให้ผลการแสดงเว็บสู่ผู้อ่าน มีการเปลี่ยนแปลงตามการใช้งาน (On The Fly) ซึ่งเป็นเทคโนโลยีที่สืบเนื่องมาจากเหตุผลดังนี้

- ความต้องการให้หน้าเว็บเปลี่ยนตามข้อมูลที่ผู้อ่านส่งให้ เช่น เว็บที่ให้บริการระบบค้นหา (Search Engine) ทั้งหลาย ที่ได้รับข้อมูลจากผู้ใช้ คือข้อความที่ต้องการค้นหา และแสดงผลลัพธ์หน้าเว็บ ตามข้อมูลที่ค้นหาเจอแก่ผู้ใช้
- ความต้องการให้หน้าเว็บเปลี่ยนตามข้อมูลที่มีการเปลี่ยนแปลงบ่อยครั้ง เช่น เว็บที่เกี่ยวข้องกับการรายงานข่าว หรือสภาพอากาศ จะเห็นว่าข้อมูลที่รายงานสู่ผู้ใช้ ดังกล่าว มีการเปลี่ยนแปลงอยู่ตลอดเวลา อาจเป็นรายชั่วโมง หรือต่อนาทีก็ได้
- ความต้องการให้หน้าเว็บเปลี่ยนตามระบบฐานข้อมูล เช่น ระบบควบคุมคลังสินค้า ที่อาศัยการเก็บข้อมูลในระบบฐานข้อมูล สามารถแสดงผลจำนวนสินค้า ตามข้อมูลที่มีอยู่จริงในฐานข้อมูล ออกสู่ผู้ใช้บนระบบเว็บ

ประโยชน์ของ Servlet นั้นช่วยให้เราเขียน Program ที่สั่งให้ HTTP Server ทำงานหรือโต้ตอบกับ Program ของ Client เหมือนที่ CGI (Common Gateway Interface) ทำได้ซึ่ง Servlet ถูกนำมาใช้แทน CGI scripts ได้อย่างมีประสิทธิภาพและให้ความสะดวกได้อย่างมากคือสามารถเขียนได้ง่ายกว่าและทำงานได้เร็วกว่า CGI และ Servlet ยังได้มีการจัดการในส่วนของการโปรแกรมทางด้าน server-side ไว้โดยเฉพาะ Servlet ได้ถูกพัฒนาโดยใช้ Java Servlet API ซึ่งเป็นส่วนเพิ่มเติมมาตรฐานของ Java




รูปที่ 2.1 รูปแสดงการทำงานของ Servlet

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.1.2 สถาปัตยกรรมของ Servlet

Servlet API สนับสนุนการ interface ของ Servlet ทั้งแบบทางตรงหรือทางอ้อม โดยประกอบไปด้วยคลาสที่สนับสนุนการ interface เช่น Http Servlet ซึ่ง Servlet interface ประกอบด้วย method ที่ใช้เป็นตัวจัดการ Servlet และติดต่อสื่อสารกับ client ผู้เขียน Servlet จะกำหนด Method นี้ทั้งหมดหรือบางส่วนเมื่อต้องการพัฒนา Servlet

คลาสที่จะสามารถทำงานเป็น Servlet ได้ต้อง extends (สืบทอด) มาจากคลาส HttpServlet หากต้องการรับข้อมูลที่ส่งมาด้วยรูปแบบ GET ให้สร้างเมธอดชื่อ doGet หรือหากต้องการ รับข้อมูลที่ส่งมาด้วยรูปแบบ POST ให้สร้างเมธอดชื่อ doPost ในตัวอย่างแสดงเฉพาะเมธอด doGet ซึ่งทั้งสองต้องมีอาร์กิวเม็นต์ 2 ตัวคือ HttpServletRequest สำหรับอ้างข้อมูลที่ส่งข้อมูลจากผู้ใช้ (จากตัวอย่างอ้างด้วยตัวแปร request) และ HttpServletResponse (จากตัวอย่างอ้างด้วยตัวแปร response) สำหรับส่งข้อมูลกลับไปยังผู้ใช้ ทั้งที่เป็นข้อมูล Header และส่วนเนื้อความ (BODY) เมื่อต้องการส่งเนื้อความสู่ผู้ใช้ มักจะส่งไปให้คลาส PrintWriter

คลาสดังกล่าวมีส่วนตรวจสอบข้อผิดพลาด หรือ Exception Handling โดยอาศัยคลาส ServletException และ IOException โดยตรวจจับในเมธอด doGet และ doPost

- import ถูกใช้เพื่อเรียกใช้งานคลาสต่างๆ ในแพ็กเกจที่สนับสนุนการทำงาน Servlet ดังนี้
- import java.io.* สำหรับคลาส PrintWriter
- import javax.servlet.* สำหรับคลาส HttpServlet
- import javax.servlet.http.* สำหรับคลาส HttpServletRequest และ HttpServletResponse

## 2.2 WAP Technology

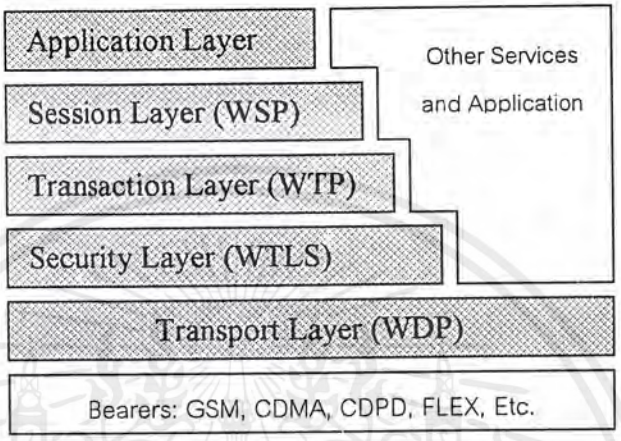
### 2.2.1 Inroduction to WAP Technology

WAP (Wireless Application Protocol) คือ โพรโตคอลที่จะทำให้ผู้ใช้โทรศัพท์มือถือสามารถเชื่อมต่อเข้ากับเครือข่ายอินเทอร์เน็ท และใช้บริการของอินเทอร์เน็ตผ่านโทรศัพท์มือถือได้ ซึ่งจะทำให้ผู้ใช้สามารถเข้าถึงข้อมูลที่ต้องการได้ตลอดเวลา โดยไม่มีข้อจำกัดเรื่องสถานที่ และยังเป็นการใช้โทรศัพท์มือถือและเครือข่ายอินเทอร์เน็ทได้อย่างคุ้มค่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.2.2 สถาปัตยกรรมของ WAP

WAP ถูกออกแบบให้มีโครงสร้างสถาปัตยกรรมเป็นชั้นๆ หรือที่เรียกว่าเลเยอร์(Layer) เพื่อความสะดวกในการปรับแต่งรวมทั้งการแก้ไขด้วย โดยอ้างอิงเทียบกับ OSI (Open System Interconnection) โดยแบ่งเป็น 5 เลเยอร์ ดังภาพ




รูปที่ 2.2 สถาปัตยกรรมของ WAP

1. **Application Layer หรือ Wireless Application Environment (WAE)** ซึ่งเป็นการทำงานของแอพพลิเคชั่นต่างๆ ของ WAP ประกอบด้วยส่วนสำคัญคือ WML และ WML Script หรือรวมเรียกว่า Wireless Telephony Application Interface (WTAI) ในเลเยอร์นั้นนี้จะเป็นการกำหนดว่า แอพพลิเคชั่นของ WAP จะเข้าถึงการทำงานภายในของอุปกรณ์แว็พ (WAP Device) ได้อย่างไร

   WML จะมีโครงสร้างคล้ายกับ HTML (Hyper Text Markup Language) โดยจะสร้างแว็พเพจเพื่อนำไปแสดงในแว็พเบราเซอร์(WAP Browser) ที่อยู่ในอุปกรณ์แว็พส่วน WMLScript จะเป็นภาษาสคริปต์ที่มีลักษณะคล้ายภาษา JavaScript โดยใช้ทำงานร่วมกับ WML

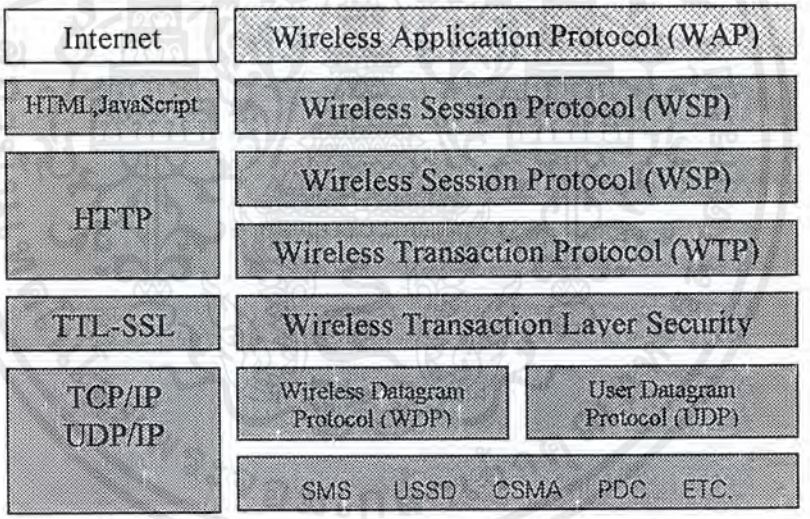
2. **Session Layer หรือ Wireless Session Protocol (WSP)** ในเลเยอร์นี้จะมีการกำหนดส่วนที่ควบคุมการเชื่อมต่อ (Session) หรืออินเตอร์เฟซ (Interface) ระหว่างไคลเอนท์กับเซิร์ฟเวอร์ 2 แบบ คือ Connection กับ Connectionless รวมทั้งการหยุดพักและกลับมาทำงาน (Suspend/Resume) ระหว่างการเชื่อมต่อได้อีกด้วย

3. **Transaction Layer หรือ Wireless Transaction Protocol (WTP)** ซึ่งจะเป็นข้อกำหนดในการร้องขอข้อมูลและตอบสนอง (Request/Response) ซึ่งจะใช้ในตอนที่มีการติดต่อกันระหว่างไคลเอนท์กับเซิร์ฟเวอร์ 3 แบบคือ one-way requests, Reliable one-way requests และ Reliable two-way request/response

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. **Security Layer** หรือ **Wireless Transport Layer Security (WTLS)** เป็นส่วนตรวจสอบความถูกต้องของข้อมูลและการรักษาความปลอดภัยต่างๆ เช่น การตรวจสอบการเข้าถึงด้วย ชื่อ และ รหัสผ่าน (Authentication) เป็นโมดูลที่ใช้เพิ่มความปลอดภัยในการใช้ข้อมูลหรือแอพพลิเคชั่น
5. **Transport Layer** หรือ **Wireless Datagram Protocol (WDP)** ทำหน้าที่ให้ เลเยอร์อื่นๆ ที่อยู่ด้านบนของเลเยอร์นี้สามารถเชื่อมต่อกับเครือข่ายสื่อสารไร้สายได้ เช่น GSM (Global System for Mobile Communication) หรือ CDMA (Code Division Multiple Access) เป็นต้น

การทำงานของ WAP โดยรวมนั้นจะต้องใช้โปรโตคอลทั้งของ WAP และอินเทอร์เน็ต ในดังภาพ เป็นการแสดงให้เห็นถึง เลเยอร์ต่างๆ ของ WAP เมื่อเทียบกับโปรโตคอลของอินเทอร์เน็ต

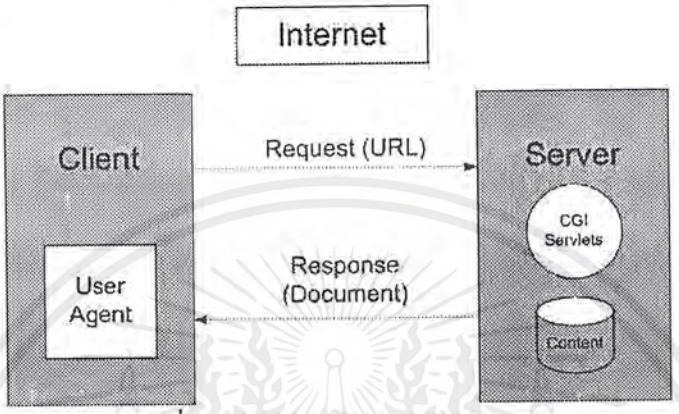



รูปที่ 2.3 เปรียบเทียบเลเยอร์ของ WAP กับอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.2.3 โมเดลการทำงานของ WEB และ WAP

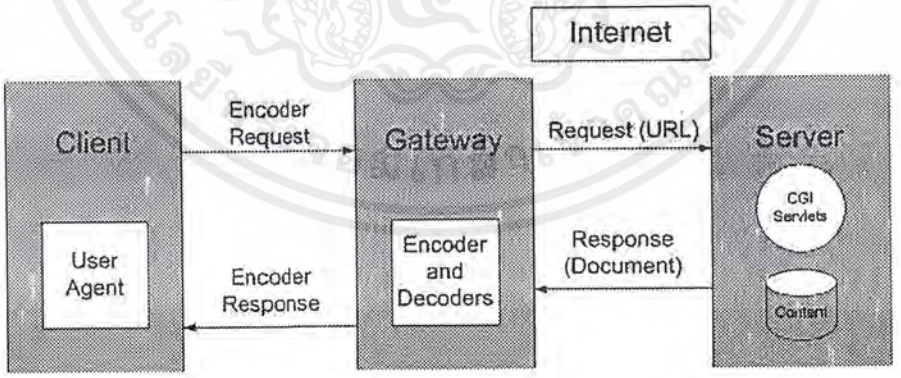
โครงสร้างการทำงานของเว็บและ WAP จะเป็นแบบไคลเอนท์และเซิร์ฟเวอร์ เมื่อไคลเอนท์ร้องขอข้อมูลไปยังเซิร์ฟเวอร์ (Request) แล้วเซิร์ฟเวอร์จะส่งข้อมูลตอบกลับไปให้ไคลเอนท์ (Response) ดังรูป




รูปที่ 2.4 โมเดลการทำงานของ WEB

การติดต่อระหว่างไคลเอนท์และเซิร์ฟเวอร์ของเว็บ จะใช้โปรโตคอลมาตรฐานของอินเทอร์เน็ต เช่น HTTP และ TCP/IP เป็นต้น โดยไคลเอนท์และเซิร์ฟเวอร์จะอยู่ในเครือข่ายเดียวกันหรือเชื่อมถึงกันได้ผ่านโปรโตคอลมาตรฐานของอินเทอร์เน็ท

สำหรับโมเดลการทำงานของ WAP นั้น จะมีส่วนที่เพิ่มเข้ามาคือ Gateway ที่เป็นตัวกลางระหว่างไคลเอนท์ (อุปกรณ์แว๊พ) กับเซิร์ฟเวอร์ที่ไคลเอนท์ต้องการติดต่อด้วยดังรูป

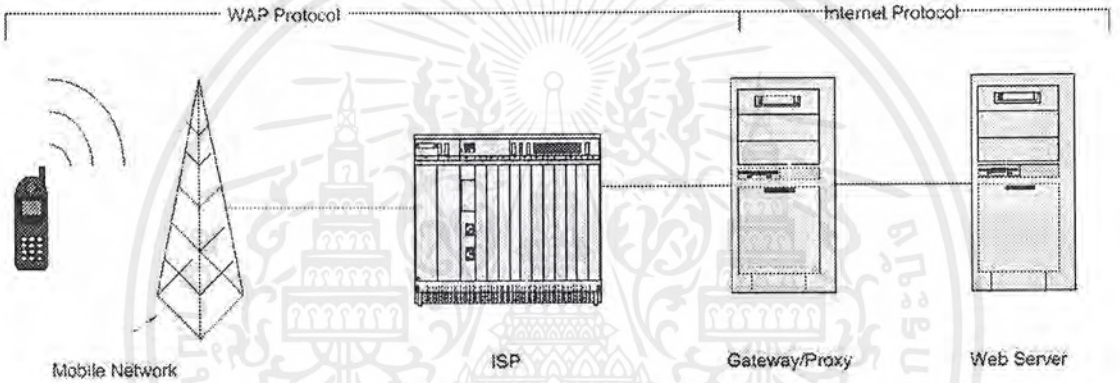



รูปที่ 2.5 โมเดลการทำงานของ WAP

การติดต่อระหว่างไคลเอนท์กับเซิร์ฟเวอร์ของ WAP จะต้องกระทำผ่าน Gateway โดยไคลเอนท์กับ Gateway จะติดต่อกันด้วยโปรโตคอลของ WAP และรับส่งข้อมูลแบบเลขฐานสองแต่ Gateway กับ เซิร์ฟเวอร์จะติดต่อกันด้วยโปรโตคอลมาตรฐานของอินเทอร์เน็ท

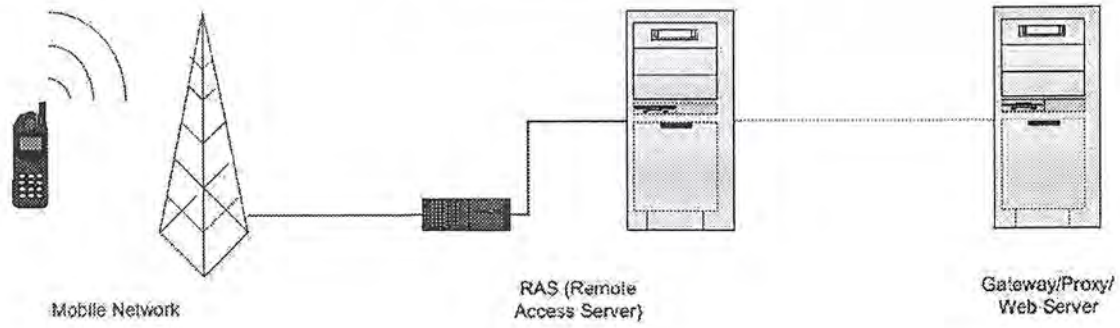
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Gateway จึงทำงานเหมือนเป็นตัวเชื่อมโปรโตคอลทั้ง 2 ฝั่งให้ติดต่อสื่อสารกันได้ โดยไคลเอนท์ซึ่งเป็นโทรศัพท์มือถือโทรเข้าใช้บริการผ่าน ISP (Internet Service Provider) โดยที่ Gateway และเว็บเซิร์ฟเวอร์ที่ให้บริการข้อมูล WAP อยู่ในเครือข่ายอินเทอร์เน็ท โดยการทำงานของในส่วนตั้งแต่ Gateway ไปถึง เซิร์ฟเวอร์จะเป็นการใช้โปรโตคอลของอินเทอร์เน็ทตามปกติ การที่อุปกรณ์แว็พจะติดต่อหรือใช้ Gateway ตัวใดนั้น ผู้ใช้จะสารมารกำหนดไว้ที่อุปกรณ์แว็พได้ว่า IP แอดเดรสของ Gateway ที่จะใช้เป็นอะไร เมื่อเชื่อมต่อกับอินเทอร์เน็ทได้และมีเรียกเปิดดูแว็พไซต์นั้น อุปกรณ์แว็พจะติดต่อกับ Gateway เอง ดังนั้น ISP จึงเป็นเพียงแค่ทางผ่านที่จะทำให้อุปกรณ์แว็พเชื่อมต่อกับเครือข่ายอินเทอร์เน็ทได้เท่านั้นไม่เกี่ยวกับการให้บริการหรือไม่ให้บริการ WAP แต่อย่างใด




รูปที่ 2.6 การเชื่อมต่อ WAP ผ่าน ISP

ในบางองค์กรอาจจะตั้ง WAP ขึ้นมาใช้ส่วนตัว ซึ่งทำได้โดยการติดตั้ง RAS (Remote Access Server) เชื่อมต่อกับ Gateway และเว็บเซิร์ฟเวอร์ขององค์กร บริการที่เหมาะสมสำหรับใช้งาน WAP ลักษณะนี้ก็อย่างเช่น เช็คเมล์, อ่านสรุปผลการทดลอง, การควบคุมอุปกรณ์ไฟฟ้าภายในองค์กร




รูปที่ 2.7 การเชื่อมต่อ WAP ในลักษณะส่วนตัวหรือองค์กร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.3 Microsoft Speech Technology

### 2.3.1 Introduction to Microsoft Speech Technology

Microsoft speech technology เป็นเทคโนโลยีที่ช่วยให้การจัดการกับเสียงพูดของมนุษย์ได้ สามารถที่จะทำการสังเคราะห์เสียง (Speech synthesis) จดจำเสียง (Speech recognition) และสามารถแยกแยะความแตกต่างของเสียงของแต่ละบุคคลได้ (Speech Identify) อีกทั้งสามารถนำเสียงที่วิเคราะห์ได้มาแปลงให้เป็นตัวอักษร (Voice dictation) และสามารถนำไปประยุกต์ใช้กับงานด้านเกี่ยวกับโทรศัพท์ (Telephony) ได้อีกด้วย

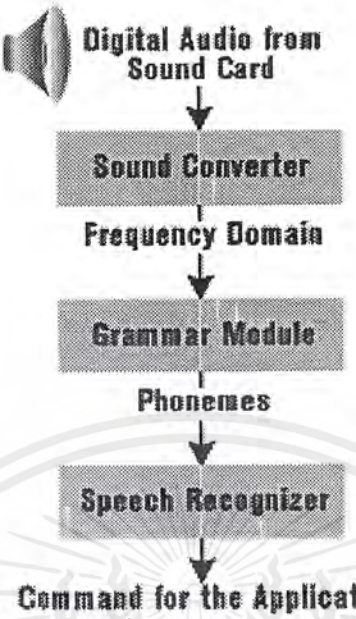
Microsoft ได้พัฒนา SDK (Software Development Kit) ชื่อว่า MSAPI 4.0 (Microsoft Speech Application Programming Interface 4.0) ขึ้นมาเพื่อช่วยให้การพัฒนา Application ที่ต้องการความสามารถในการประมวลผลกับเสียงทำได้ง่ายมากขึ้น MSAPI4.0 มีส่วนประกอบหลัก ๆ อยู่ 4 ส่วน คือ

1. SR (Speech Recognition engine)
2. DSR (Dictation Speech Recognition)
3. TEL (Telephony)
4. TTS (Text-To-Speech)

**SR (Speech Recognition engine)**

Speech Recognition เป็นความสามารถของคอมพิวเตอร์ที่เข้าใจและแปลคำพูด ซึ่ง Speech Recognition Engine จะช่วยให้เราพัฒนาแอพพลิเคชั่นที่สามารถนำเสียงพูดมาแทน Input device แบบเดิมเช่น mouse , keyboard หรือ joystick รูปที่ 1 จะแสดงภาพการทำงานของ Speech Recognition Engine ในขั้นแรก เสียงที่ผ่านเข้ามาทางไมโครโฟนจะถูกแปลงให้เป็นข้อมูลเสียงดิจิตอลโดย sound card เพื่อให้อยู่ในรูปแบบที่สามารถวิเคราะห์ได้ จากนั้น Converter ก็จะแปลงข้อมูลเสียงดิจิตอลให้อยู่ในรูปของส่วนประกอบความถี่ (Frequency component) ลำดับต่อไปจะทำการแยกแยะหน่วยเสียง (Phoneme) แต่ละส่วนประกอบความถี่ของเสียงจะถูกเปรียบเทียบกับหน่วยเสียงเฉพาะ (specific phoneme) ในกระบวนการนี้จะแปลงจากเสียงเป็นประโยค และขั้นตอนสุดท้ายจะทำการวิเคราะห์ข้อความและไวยากรณ์ ได้เป็นคำสั่งที่จะนำไปสั่งการแอพพลิเคชั่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 2.8 แสดงขั้นตอนการวิเคราะห์จดจำเสียง

**DSR (Dictation Speech Recognition engine)**

DSR เป็นกระบวนการในการแปลงเสียงให้เป็นข้อความ ในขณะที่ผู้ใช้กำลังพูดอยู่ทันที DSR จะมีการเก็บข้อมูลคำศัพท์จำนวนมากนำไว้เปรียบเทียบกับเสียงที่วิเคราะห์ได้ และใช้ Context-free grammar เพื่อนิยามกลุ่มของคำ ดังนั้นจึงต้องการกำลังในการประมวลผลสูงกว่า SR การวิเคราะห์เสียงพูดเกี่ยวข้องกับตัวแปรหลายอย่าง เช่น การพูดและสำเนียงของผู้พูดแต่ละคน การวิเคราะห์เสียง (Dictation) มี 2 ชนิด discrete และ continuous โดยชนิด discrete นั้น แอพพลิเคชั่นจะมีการกำหนดกลุ่มของคำที่สามารถวิเคราะห์ได้ และผู้ใช้จะต้องทำการหยุดพูดระหว่างคำ ส่วนแบบ continuous นั้นผู้ใช้ไม่มีข้อจำกัดในเรื่องนี้ ซึ่ง ใน Speech SDK 4.0 นี้จะสนับสนุนการวิเคราะห์ชนิด continuous ซึ่งมีความซับซ้อนกว่าแบบ discrete และต้องการกำลังในการประมวลผลสูงเพื่อที่จะได้ผลลัพธ์ที่ถูกต้อง

**TEL (Telephony)**

แอพพลิเคชั่นที่เกี่ยวกับโทรศัพท์ จะใช้ Telephony control object ในการทำงาน ซึ่งลักษณะของแอพพลิเคชั่นจะเป็นการแลกเปลี่ยนการถามและตอบระหว่างแอพพลิเคชั่นกับผู้ใช้ หรือเป็นลำดับขั้นของการสนทนา เช่น "You have three new messages. Do you want to listen to them?" ซึ่งคำตอบจะเป็นคำตอบที่สามารถคาดเดาได้เช่น ตอบว่า yes หรือ no

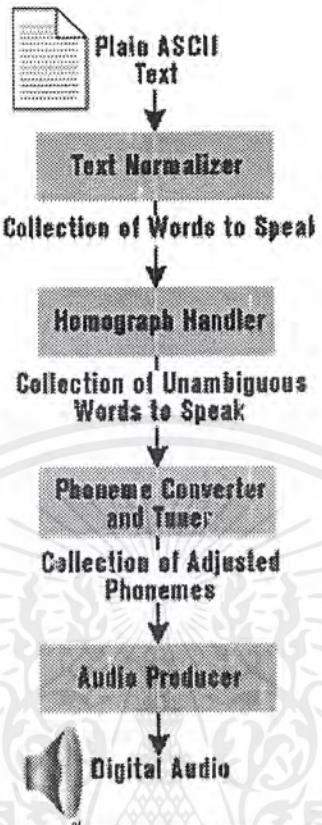
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**TTS (Text-To-Speech)**

TTSengine จะเป็นการตรงกันข้ามกับ SR และ DSR โดยจะทำการรับอินพุต ที่เป็นข้อความ และทำการแปลงเป็นเสียงอยู่ในรูปแบบ PCM mono, 8 bit, 11 kHz PCM (Pulse Code Modulation) เป็นวิธีทั่วไปที่ใช้แสดงถึงสัญญาณเสียง PCM จะประกอบไปด้วยลำดับของค่าที่ได้จากการ sampling มาจากแอมปลิจูดของสัญญาณเสียง

TTSengine จะทำการแปลงจากข้อความให้ไปเป็นหน่วยเสียงและแสดงออกมาเป็นเสียงพูดทางลำโพง หรืออาจจะทำการบันทึกเสียงเก็บไว้ในรูปแบบไฟล์ .WAV ได้ รูปที่ 2 จะแสดงให้เห็นถึงกระบวนการแปลงจากข้อความไปสู่ข้อมูลเสียงดิจิตอล สามารถแบ่งได้เป็น 4 ขั้นตอน ขั้นตอนแรกเรียกว่า text normalization ซึ่งจะกำหนดว่าแต่ละคำจะออกเสียงอย่างไร ซึ่งบางครั้งอาจจะจะต้องมีการขยายหรือแทนที่ คำที่จะพูด เช่น ตัวเลข , คำย่อ , เครื่องหมายทางการเงิน, เวลา , วันที่ เป็นต้น เมื่อทำการ normalize เสร็จสิ้น ข้อความก็จะถูกแปลงไปเป็นกลุ่มของคำที่จะออกเสียงได้ กลุ่มคำที่ได้อาจจะยังไม่สมบูรณ์นักเนื่องจากคำบางคำสามารถออกเสียงได้หลายอย่างขึ้นอยู่กับรูปประโยคเช่น read สามารถออกเสียงได้ทั้ง “reed” หรือ “red” เมื่อทำการวิเคราะห์ว่าเสียงใดควรออกเสียงอย่างไรแล้ว ก็จะทำการผ่านข้อมูลนั้นเข้าไปสู่ module การเทียบกับหน่วยเสียง ในขั้นตอนนี้จะกระทำได้ 2 ทาง ทางแรก module จะพยายามค้นหาคำใน pronunciation database ถ้าค้นหาไม่พบ ก็จะพยายามสร้างหน่วยเสียงขึ้นมาโดยใช้กฎ letter-to-sound ในขั้นนี้ engine ของแต่ละผู้พัฒนาอาจจะแตกต่างกันไป ในส่วนของ Microsoft Speech SDK นั้นจะพยายามปรับแต่งเสียงสุดท้ายโดยการเลือกความเร็ว,เวลา,ความแหลม,ความดัง ที่เหมาะสม และในท้ายที่สุดจะเป็นกระบวนการสร้างไฟล์ WAV โดยการแปลงจากหน่วยเสียงและข้อมูลเพิ่มเติมให้กลายเป็นลำดับค่าของแอมปลิจูดของเสียง ซึ่งใช้ เทคนิค word concatenation , synthesis, และ subword concatenation ซึ่งอธิบายได้ดังรูปที่ 3 การใช้งาน TTS , voice recognition และ dictation นั้นสามารถทำได้โดยใช้ Active X control, C++ class หรือ COM Object

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

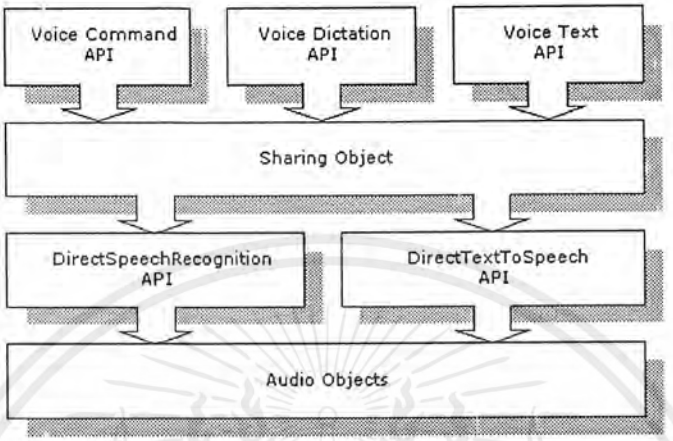



รูปที่ 2.9 แสดงขั้นตอนการสังเคราะห์เสียงพูดจากข้อความ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.3.2 การพัฒนาแอพพลิเคชั่นด้วย Speech SDK

Microsoft Speech API 4.0 ประกอบด้วยส่วนประกอบดังรูป




รูปที่ 2.10 แสดงส่วนประกอบของ Microsoft Speech API 4.0

**Voice Command API**

Voice Command อินเทอร์เฟซ เป็นอินเทอร์เฟซระดับสูงสำหรับ Speech recognition ถูกออกแบบมาเพื่อใช้สำหรับ command and control speech recognition ด้วยอินเทอร์เฟซนี้ ผู้ใช้สามารถที่จะสั่งการง่าย ๆ แก่คอมพิวเตอร์ได้ อย่างเช่น "Open the file" หรือ "yes/no" แต่อินเทอร์เฟซนี้ไม่สามารถทำการ dictation ได้

Voice Command เลียนแบบการทำงานแบบเมนูของ windows โดยจะทำการกำหนดเมนูของคำสั่งที่ผู้ใช้สามารถที่จะสั่งงานได้ โดยผู้พัฒนาแอพพลิเคชั่นจะออกแบบเมนูซึ่งสัมพันธ์กับสถานะของแต่ละ window ภายในแอพพลิเคชั่น ภายในเมนูจะประกอบไปด้วยรายการของ voice command ที่ผู้ใช้สามารถพูดได้ ซึ่งจะทำให้ผู้ใช้สามารถควบคุมแอพพลิเคชั่นได้โดยการพูดผ่านทางไมโครโฟน แทนการใช้เมาส์หรือคีย์บอร์ด

**Voice Dictation API**

Voice Dictation module จะทำให้ผู้ใช้สามารถที่จะสั่งพิมพ์ข้อความภายในแอพพลิเคชั่นได้ โดยการทำงานจะใช้การอุปมาคล้ายกับเป็น edit box ที่มองไม่เห็น ซึ่งจะรับข้อความเมื่อผู้ใช้ทำการพูด แอพพลิเคชั่นจะใช้ voice dictation object ในการสังเกตการเปลี่ยนแปลงของข้อความภายใน edit box แล้วจะทำการแสดงผลการเปลี่ยนแปลงให้แก่ผู้ใช้

Voice Dictation ยังสามารถที่จะทำการจัดรูปแบบข้อความโดยอัตโนมัติเช่นการ แปลงจากตัวอักษรเล็กเป็นตัวอักษรใหญ่ หรือการแปลงเป็นสัญลักษณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Voice Text API**

Voice Text API สามารถทำการแปลงจากข้อความให้เป็นเสียงพูดได้ การพัฒนาแอพพลิเคชั่นทำได้โดยใช้ voice text object ให้ทำการจัดการระหว่าง text-to-speech engine กับลำโพงหรือสายโทรศัพท์ โดยก่อนที่แอพพลิเคชั่นจะสามารถเริ่มใช้ voice text ได้ จะต้องทำการสร้าง instance ของ voice text object และ register เข้ากับ object นั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 3
# การสื่อสารข้อมูล

การสื่อสารข้อมูลที่จะกล่าวถึงในบทนี้เป็นการสื่อสารข้อมูลระหว่างคอมพิวเตอร์กับอุปกรณ์ภายนอก โดยมีรูปแบบของการติดต่อกับอุปกรณ์ภายนอกที่นิยมมี 2 รูปแบบคือ

1 การสื่อสารแบบขนาน

2 การสื่อสารแบบอนุกรม

แต่ก่อนที่จะทำความรู้จักกับการสื่อสารทั้งสองควรจะมีความเข้าใจเกี่ยวกับคำส่งหรือข้อมูลที่อยู่ในรูปของบิท ซึ่งประกอบด้วยหลายบิตมาประกอบกัน ข้อมูลในการสื่อสารแต่ละข้อมูลจะถูกแปลงให้อยู่ในรูปแบบของเลขฐานสอง แล้วนำมาประกอบกัน เช่น ถ้าข้อมูลที่ประกอบด้วย 4 บิตเราจะเรียกว่า 1 นิบเบิล หรือถ้าหากข้อมูลที่ประกอบด้วย 8 บิตเราจะเรียกว่า 1 ไบต์ เป็นต้น

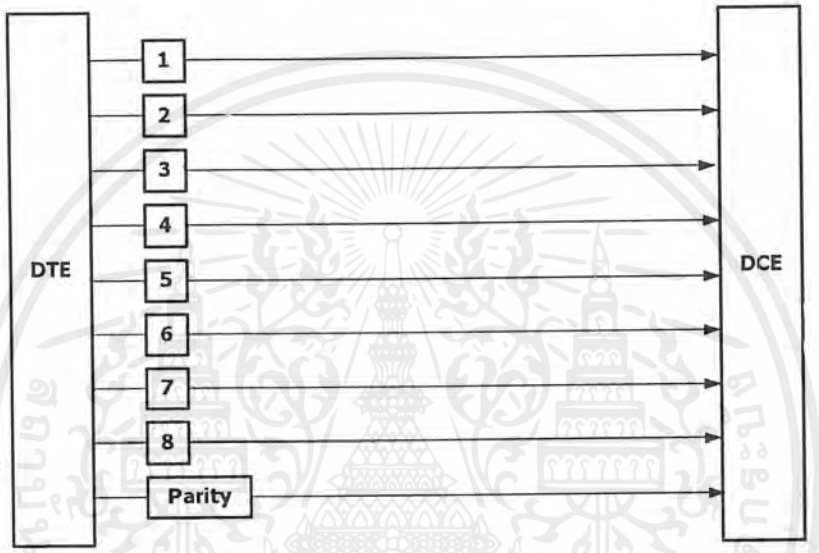
## 3.1 การสื่อสารแบบขนาน

การสื่อสารแบบขนานจะมีรูปแบบการส่งข้อมูลครั้งละ 1 ไบต์ก็คือจะทำการส่งข้อมูลครั้งละ 8 บิตนั่นเอง ซึ่งในการส่งต้องใช้สายไฟในการส่งข้อมูล 8 เส้น แล้วยังต้องใช้สายไฟอีก 1 เส้นในการควบคุมเช่น ใช้เป็นพาริตี้บิต หรืออาจจะมีมากกว่านั้นเพื่อใช้ในการควบคุมการโต้ตอบของการทำงาน (Hand-shake) ซึ่งรายละเอียดจะบอกใช้ทราบต่อไปในเรื่องของการ Hand-shake จึงสรุปได้ว่าในการสื่อสารแบบขนานนั้นต้องใช้สายอย่างน้อยที่สุด 9 เส้น

ดั้งนั้นในการส่งข้อมูลทีละ 1 ไบต์นั้นทำให้ข้อมูลทั้ง 8 บิตมาถึงปลายทางพร้อมกัน ทำให้ข้อมูลแบบขนานสามารถทำได้ด้วยความเร็วที่สูงมาก แต่ปัญหาที่สำคัญของการส่งข้อมูลแบบขนานคือคุณสมบัติของบิตกับแรงดัน เวลาที่บิตหรือแรงดันไฟฟ้าที่ใช้ในการเปลี่ยนสถานะจากหนึ่งเป็นศูนย์นั้นสั้นมาก โดยเร็วถึงระดับนาโนวินาที (หนึ่งในพันล้านของวินาที) การเปลี่ยนแปลงที่รวดเร็วนี้เป็นส่วนที่สำคัญมากต่อการส่งข้อมูล เพราะการเปลี่ยนแปลงระหว่างศูนย์และหนึ่งอย่างช้า ๆ จะไม่ถูกอ่านเป็นข้อมูลเลย และเมื่อสายไฟที่ใช้ส่งข้อมูลยาวขึ้น คุณสมบัติทางไฟฟ้าของสายไฟเช่น ค่าความจุไฟฟ้าและค่าความเหนี่ยวนำจะจำกัดความเร็วในการเปลี่ยนแปลงระหว่างศูนย์และหนึ่งของบิต ซึ่งจะทำให้ข้อมูลอาจสูญหายหรือทำให้การส่งข้อมูลล้มเหลวได้ ดังนั้นการส่งข้อมูลบนสายยาวอาจจะเป็นปัญหาได้หากใช้วิธีการสื่อสารแบบขนาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากข้อเสียของการส่งข้อมูลแบบขนานสองอย่างคือค่าใช้จ่ายที่สูงและการสูญหายของข้อมูล ทำให้การใช้งานของมันถูกจำกัดอยู่กับอุปกรณ์เพียงไม่กี่ชนิด เช่น เครื่องพิมพ์ที่มักจะอยู่ใกล้กับเครื่องคอมพิวเตอร์และต้องทำงานที่ความเร็วสูง แต่เรายังใช้วิธีการส่งข้อมูลแบบขนานนี้ภายในเครื่องคอมพิวเตอร์เนื่องจากไม่ต้องใช้สายไฟขนาดยาว




รูปที่ 3.1 แสดงการสื่อสารแบบขนาน

## 3.2 การสื่อสารแบบอนุกรม

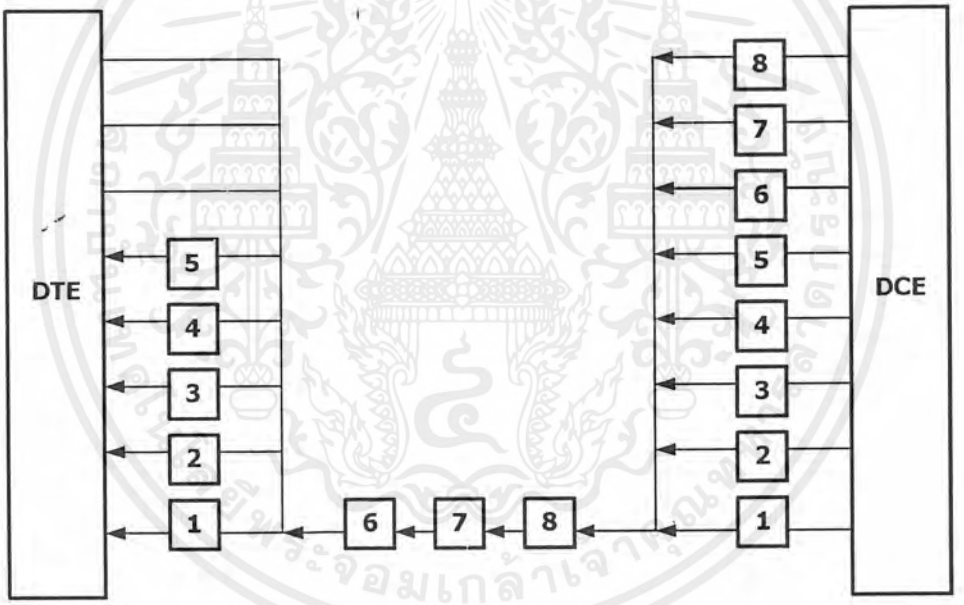
การสื่อสารแบบอนุกรมเป็นการส่งข้อมูลทีละบิต และข้อมูลจะถูกต่อรวมเข้าเป็นไบต์ใหม่ด้วยวิธีการส่งข้อมูลทีละบิตนี้ ทำให้สามารถใช้สายไฟเพียงสองเส้นในการส่งข้อมูล ซึ่งช่วยให้เราประหยัดค่าสายไฟไปได้มาก แต่ก็ทำให้ประสิทธิภาพการทำงานลดลงไปด้วย เพราะการส่งข้อมูลวิธีนี้ต้องใช้เวลานานขึ้นอย่างน้อยแปดเท่าของการส่งข้อมูลแบบขนาน แต่ความเร็วที่ลดลงไปนี้ยังไม่ถือว่าเป็นข้อจำกัดที่สำคัญทางการใช้งานนัก เพราะหากว่าเราพิจารณาที่อุปกรณ์ทั่วไป จะพบว่าอุปกรณ์ส่วนใหญ่ทำงานช้ามากเมื่อเทียบกับความเร็วในการทำงานภายในไมโครโปรเซสเซอร์ อุปกรณ์แต่ละตัวมีขั้นตอนการทำงานที่กินเวลานาน ซึ่งโดยทั่วไปมักจะเป็นกระบวนการทางกลไก (mechanic)ที่เป็นตัวจำกัดความเร็วของเครื่องลงไปอย่างมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



ตัวอย่างเช่น ความเร็วของเครื่องพิมพ์ถูกจำกัดที่ความเร็วของหัวพิมพ์ (printhead)ความเร็วของโมเด็มถูกจำกัดโดยขีดจำกัดความถี่ของสายโทรศัพท์ และความเร็วของดิสก์ไดรฟ์ถูกจำกัดโดยอัตราเร็วการหมุนของไดรฟ์ ดังนั้นความเร็วที่ได้มาจากการส่งข้อมูลแบบขนานจะเสียไปโดยเปล่าประโยชน์เมื่อนำมาใช้งานได้ แม้ว่าอัตราเร็วการส่งข้อมูลจะลดลงแต่ก็ยังสามารถใช้งานร่วมกับอุปกรณ์ประเภทนี้ได้อย่างมีประสิทธิภาพ ข้อเสียจากความเร็วที่ลดลงไปไม่อาจเทียบได้กับผลพวงที่ได้จากคุณภาพการส่งและระยะทางการส่งข้อมูลที่เพิ่มขึ้น




รูปที่ 3.2 แสดงการสื่อสารข้อมูลแบบอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
46438

ตารางที่ 3.1 ตารางเปรียบเทียบการทำงานระหว่างการสื่อสารข้อมูลแบบขนานและแบบอนุกรม

| การใช้งาน | แบบขนาน | แบบอนุกรม |
|---|---|---|
| 1.ระยะทาง | จะใช้งานได้ในระยะไม่เกิน100 ฟุต | จะสามารถใช้งานได้ตั้งแต่ในระยะใกล้ ๆ ไปจะถึงระยะทางที่มาก ๆ จนถึงหลักไมล์ |
| 2.ความเร็ว | อัตราความเร็วสูงมากในระยะที่ไม่ไกลมากนักกำหนดได้เป็นจำนวนบิทต่อวินาที | อัตราความเร็วของข้อมูลที่ใช้กันอยู่ทั่วไปจะอยู่ในช่วง 0-2 ล้านบิทต่อวินาที |
| 3.ระดับของสัญญาณ | ในการอินเตอร์เฟสจะใช้ระดับของสัญญาณที่ใช้กับอุปกรณ์ TTL คือสัญญาณลอจิก 1 และ 0 จะแทนด้วยระดับแรงดัน +5V และ 0V | ในมาตรฐานของ EIA – RS 232C ระบุว่ามีระดับสัญญาณไฟฟ้าขนาด 12V หรือใช้มาตรฐาน 20 mA current loop |
| 4.ความผิดพลาดของสัญญาณ | ในการใช้งานระยะไกลๆ ความผิดพลาดของข้อมูลจะเกิดขึ้นได้ง่าย | การใช้งานจะเกิดการผิดพลาดของสัญญาณจะมีน้อยมากสิ้นเปลืองน้อยกว่ามาก |
| 5.ค่าใช้จ่าย | ค่าใช้จ่ายจะสูงมากเพราะจะต้องใช้สายส่งสัญญาณหลายเส้น โดยเฉพาะการส่งในระยะทางไกล ๆ | ถึงแม้ว่าจะต้องใช้อุปกรณ์เปลี่ยนสัญญาณของข้อมูลจากขนานไปเป็นอนุกรมแล้วส่งผ่านสายส่ง แล้วกลับสัญญาณมาเป็นขนานอีกครั้งก็ตาม |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 3.3 การอินเตอร์เฟส

การส่งข้อมูลอยู่ภายในเครื่องคอมพิวเตอร์สามารถกระทำได้ง่าย เนื่องจากเราสามารถคาดเดาสภาพแวดล้อมภายในเครื่องได้ แต่ในการส่งข้อมูลสู่ภายนอก เราไม่สามารถคาดการณ์ได้ว่าเครื่องคอมพิวเตอร์และตัวข้อมูลจะต้องพบกับสภาพเช่นไรและจะมีผลกระทบต่อตัวข้อมูลและเครื่องคอมพิวเตอร์อย่างไร ดังนั้นในการออกแบบวงจรจึงมีสิ่งที่ต้องพิจารณาคือ จะต้องหาวิธีในการแยกข้อมูลออกจากสภาพแวดล้อมและสัญญาณรบกวนโดยเฉพาะอย่างยิ่งในกรณีที่มีสายส่งข้อมูลขนาดยาว สิ่งที่จะต้องพิจารณาอีกสิ่งหนึ่งก็คือ จะต้องหาวิธีการป้องกันคอมพิวเตอร์จากสภาพแวดล้อมอันไม่พึงประสงค์ด้วย นั่นก็คือจะต้องมีตัวอินเตอร์เฟส ซึ่งจะมีหน้าที่เป็นจุดเชื่อมต่อระหว่างคอมพิวเตอร์กับสิ่งแวดล้อมภายนอก และเนื่องจากอุปกรณ์อินเตอร์เฟสมีหน้าที่คล้ายกับเป็นประตูของเครื่องคอมพิวเตอร์ บางครั้งมันจึงถูกเรียกว่า I/O พอร์ต (I/O PORT)หรือบางครั้งเรียกสั้นๆ ว่าพอร์ต (PORT)

วัตถุประสงค์หลักของการอินเตอร์เฟสก็คือ การใช้อุปกรณ์อินเตอร์เฟสเป็นสื่อกลางของการส่งข้อมูล และวัตถุประสงค์ที่สำคัญอีกอย่างหนึ่งของการอินเตอร์เฟสก็คือ ความง่ายต่อการใช้งาน และเมื่อเราสามารถทำการอินเตอร์เฟสได้สำเร็จ ก็จะสามารถที่จะส่งข้อมูลสู่ภายนอกได้

**มาตรฐานการอินเตอร์เฟส RS-232**

มาตรฐานการอินเตอร์เฟส RS-232 ได้เกิดขึ้นตั้งแต่ปี 1969 EIA (Electronic Industries Association) ห้องวิจัย Bell และบรรดาผู้ผลิตอุปกรณ์สื่อสารได้ร่วมกันจัดมาตรฐาน EIA RS-232 ซึ่งต่อมาไม่นานนักก็ได้มีการปรับปรุงแก้ไขอีกเล็กน้อยกลายเป็น RS-232 C และเมื่อไม่นานมานี้ก็ได้ออกมาตรฐาน RS-232 D

เพื่อทำความเข้าใจกับวัตถุประสงค์หลักของ RS-232 ได้ดียิ่งขึ้น ควรทำความเข้าใจกับวัตถุประสงค์หลักของ RS-232 ก่อน ซึ่งได้แสดงไว้อย่างชัดเจนในหัวข้อของเอกสารคือ

Interface Between Data Terminal Equipment and Data Communications Equipment Employing Serial Binary Data Interchange (การอินเตอร์เฟสระหว่างอุปกรณ์เทอร์มินัลและอุปกรณ์สื่อสารข้อมูลที่ใช้วิธีการแลกเปลี่ยนข้อมูลไบนารีแบบอนุกรม)

ซึ่งในหัวข้อสำคัญจะอธิบายถึงการอินเตอร์เฟสระหว่างเทอร์มินัล (Data Terminal Equipment หรือ DTE) กับโมเด็ม (Data Communications Equipment หรือ DCE) เพื่อใช้ส่งข้อมูลแบบอนุกรม โดยจะประกอบด้วย 4 ส่วนคือ

- คุณสมบัติทางไฟฟ้าของสัญญาณ (Electrical Signal Characteristics) ซึ่งจะอธิบายถึงรูปแบบของสัญญาณไฟฟ้าที่ตัวอินเตอร์เฟสจะส่งออก และรับเข้ามาจากภายนอก ระดับแรงดันไฟฟ้าที่แสดงถึง ตรรกะ 0 และ 1 ก็จะมีกำหนดไว้ในส่วนนี้ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- คุณสมบัติทางกลไกการอินเตอร์เฟส : คอนเน็กเตอร์ (Interface Mechanical Characteristics : Connectors) ซึ่งเป็นข้อกำหนดเกี่ยวกับตัวอินเตอร์เฟส ประกอบด้วยส่วนที่เป็นปลั๊ก (plug) และเต้าเสียบ (receptacle) โดยเต้าเสียบจะต้องอยู่บน DCE สำหรับ RS-232 A-C ไม่ได้มีการกำหนดคอนเน็กเตอร์รูปตัว D (D-Shaped) ซึ่งมีใช้กันอยู่ทั่วไปนั้น ทั้งนี้เพราะว่าอุปกรณ์ตัวนี้ได้รับการคุ้มครองโดยลิขสิทธิ์ และเมื่อสิทธิบัตรนั้นหมดอายุลงใน RS-232 D จึงได้เพิ่มข้อกำหนดคอนเน็กเตอร์ DB-25 เข้าไว้ในมาตรฐานด้วย
- หน้าที่การทำงานของวงจรการแลกเปลี่ยน (Functional Description of Interchange Circuit) ในส่วนนี้กำหนดหน้าที่และตั้งชื่อให้กับสัญญาณไฟฟ้าต่าง ๆ ที่นำมาใช้ เช่น Transmitted Data (ข้อมูลส่งออก) ได้ถูกกำหนดไว้ให้กับขา 2
- มาตรฐานการอินเตอร์เฟสสำหรับระบบการสื่อสารเฉพาะอย่าง (Standard Interfaces for Selected Communications System Configurations) ในส่วนนี้เป็นรายละเอียดต่าง ๆ สำหรับการติดต่อระหว่างโมเด็มกับเทอร์มินัลทั่วไป

**พื้นฐานการอินเตอร์เฟส RS-232**

ก่อนที่จะทำความรู้จักกับพื้นฐานการอินเตอร์เฟส RS-232 ก่อนอื่นต้องทำความเข้าใจกับคำว่า DTE และ DCE ก่อน

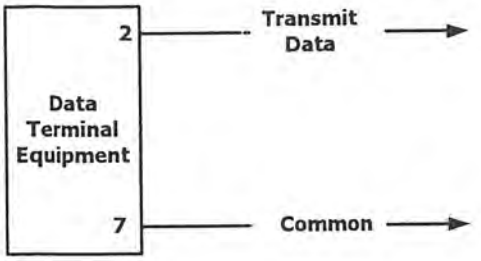
DTE = Data Terminal Equipment ซึ่งก็คือ คอมพิวเตอร์ นั่นเอง (ตัวส่งข้อมูล)

DCE = Data Communications Equipment อุปกรณ์เหล่านี้ ได้แก่ โมเด็ม,TA อแดปเตอร์, พล็อตเตอร์ เป็นต้น (ตัวรับข้อมูล)

ตามมาตรฐานของ EIA ได้กำหนดไว้ว่า อุปกรณ์ DTE หมายถึง อุปกรณ์ที่ข้อมูลมาสิ้นสุด และอุปกรณ์ DCE เป็นอุปกรณ์ที่ใช้ส่งผ่านข้อมูล ดังนั้นคอมพิวเตอร์ซึ่งสามารถทำได้ทั้งส่งผ่านข้อมูลและรับข้อมูล จึงไม่อาจระบุได้ว่าเป็นอุปกรณ์ DCE หรือ DTE

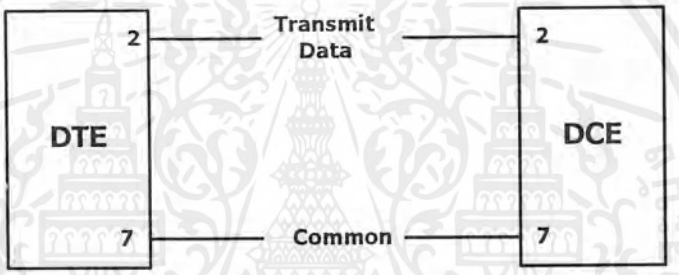
ในการพิจารณาโครงสร้างเบื้องต้นของการอินเตอร์เฟส RS-232 นั้นจะประกอบด้วยเส้นสายไฟเพียง 2 เส้น เส้นหนึ่งใช้ในการส่งข้อมูล และอีกเส้นหนึ่งสำหรับอ้างอิงแรงดันของวงจรอินเตอร์เฟส ( circuit common) ซึ่งมักจะมีความเข้าใจผิดว่าคือกราวนด์ (ground) แต่แท้จริงแล้วไม่ใช่ ดังรูปที่ 3.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 3.3 แสดงอุปกรณ์ DTE เบื้องต้น

จากรูปที่ 3.3 เราจะพบว่ามีเพียงตัวส่งข้อมูล แต่ในความจริงแล้วในการอินเตอร์เฟสข้อมูลใดๆ จะต้องมีส่วนที่รับข้อมูลด้วย ซึ่งก็คือ DCE นั่นเอง ดังรูป 3.4

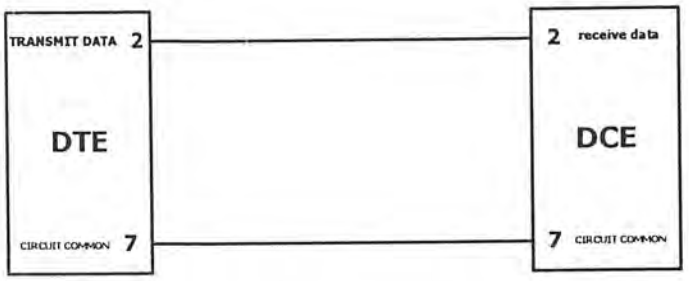



รูปที่ 3.4 แสดงอุปกรณ์ทั้ง DTE และ DCE

เมื่อพิจารณารูปที่ 3.4 จะพบว่าข้อมูลที่ถูกส่งออกจากขา 2 ของ DTE จะรับเข้าไปยังขา 2 ของ DCE เช่นกัน โดยข้อมูลที่รับทาง DCE แน่นอนจะต้องเป็นข้อมูลเดียวกันที่ส่งออกมาจาก DTE ทำให้สรุปได้ว่า " Transmit Data" มิได้มีส่วนในการกำหนดว่าอุปกรณ์ใดเป็นตัวต้นทางหรือปลายทาง แต่จะขึ้นอยู่กับว่าจะพิจารณาเช่นไร

ในที่นี้จะใช้การพิจารณาดังนี้คือ สัญญาณที่ส่งออกไปเรียกว่า "เอาต์พุต" และสัญญาณที่รับเข้ามาเรียกว่า "อินพุต" โดยถือว่าสัญญาณเป็นกิจกรรมทางไฟฟ้าที่เกิดขึ้นในกระบวนการอินเตอร์เฟส ดังรูปที่ 3.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 3.5 แสดงอุปกรณ์ DTE และ DCE ซึ่งเป็นคู่อุปกรณ์ที่ทำงานตรงข้ามกัน

## 3.4 การรับส่งข้อมูลสองทิศทาง

เมื่อพิจารณารูปที่ 3.5 แล้วเราจะพบว่าเมื่ออุปกรณ์ DTE รับข้อมูลแล้วจะต้องส่งผ่านไปยังอุปกรณ์ DCE (โมเด็ม) แล้วโมเด็มก็จะทำหน้าที่ในการส่งข้อมูลต่อออกไปยังสายโทรศัพท์ ซึ่งจะเป็นได้ว่าอุปกรณ์ DTE และอุปกรณ์ DCE สามารถเป็นได้ทั้งอุปกรณ์ส่งและรับข้อมูล และยังสามารถส่งและรับข้อมูลในทิศทางตรงกันข้ามได้อีกด้วย ดังรูปที่ 3.6

รูปที่ 3.6 แสดงการทำงานของอุปกรณ์ที่สามารถส่งและรับข้อมูลได้ทั้งสองทิศทาง

ซึ่งจากรูปที่ 3.6 ก็คงจะพอสรุปความแตกต่างระหว่างอุปกรณ์ DTE และ DCE ได้ดังนี้

DTE ส่งเอาต์พุตที่ขา 2 และรับอินพุตที่ขา 3

DCE ส่งเอาต์พุตที่ขา 3 และรับอินพุตที่ขา 2

โดยในการอินเตอร์เฟสนั้นจะต้องทำการตรวจสอบทิศทางของสัญญาณข้อมูลที่ขา 2 และที่ขา 3 ก่อนเสมอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**การแฮนด์เชคใน RS-232**

การแฮนด์เชคหมายถึง กระบวนการที่อุปกรณ์หนึ่งใช้ตรวจสอบสถานะของอีกอุปกรณ์ที่ต่อเข้าด้วยกัน และตอบสนองสถานะนั้นอย่างเหมาะสมและถูกจังหวะเวลา ซึ่งก็คือวิธีการควบคุมการทำงานของอุปกรณ์สองตัวให้สัมพันธ์กันในการรับส่งข้อมูลนั่นเอง การแฮนด์เชคนั้นแบ่งออกเป็น 2 ประเภทคือ การแฮนด์เชคทางฮาร์ดแวร์ และ การแฮนด์เชคทางซอฟต์แวร์

การแฮนด์เชคทางซอฟต์แวร์ (software handshaking) เป็นวิธีการหนึ่งในการควบคุมการทำงานของอุปกรณ์รับข้อมูลโดยส่งผ่านสัญญาณควบคุมไปพร้อมกับตัวข้อมูลที่ต้องการส่ง

การแฮนด์เชคทางฮาร์ดแวร์ (hardware handshaking) สามารถควบคุมได้ตั้งแต่ระดับฮาร์ดแวร์โดยการเปลี่ยนระดับแรงดันในสายสัญญาณควบคุมเป็นตัวระงับไม่ให้คอมพิวเตอร์ส่งข้อมูลเพิ่มเข้ามาอีก ซึ่งเป็นการหลีกเลี่ยงการใช้รหัสหรือโปรแกรม แต่การแฮนด์เชคทางฮาร์ดแวร์นั้นมีข้อจำกัด คือจำเป็นต้องมีสายสัญญาณควบคุมต่างหากโดยเฉพาะ ทำให้วิธีนี้ไม่เหมาะที่จะนำมาใช้ในการอินเตอร์เฟสกับโมเด็ม

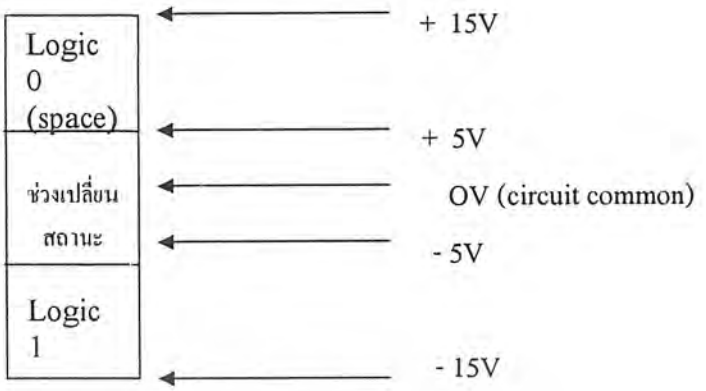
**ขอบเขตความคอมแพติเบิล (compatible) กับ RS-232**

คุณสมบัติทางไฟฟ้า (ระดับแรงดัน ฯลฯ) ของการอินเตอร์เฟสได้รับการตรวจรับรอง ถ้าอุปกรณ์นั้นถูกอ้างว่าคอมแพติเบิลกับ RS-232 ย่อมหมายความว่าเราสามารถนำอุปกรณ์นั้นไปติดต่อกับอุปกรณ์อื่นที่คอมแพติเบิลกับ RS-232 ได้ โดยไม่ทำให้เกิดความเสียหายแก่อุปกรณ์ทั้งคู่ ซึ่งเงื่อนไขนี้ช่วยให้เรามั่นใจได้ว่าอุปกรณ์ทั้งคู่มีระบบไฟฟ้าที่ทำงานด้วยกันได้ โดยไม่ทำให้เกิดความบกพร่องในการรับส่งข้อมูล ระดับแรงดันสำหรับค่า "ศูนย์" และ "หนึ่ง" ต้องเป็นไปตามที่ระบุไว้ในมาตรฐาน

**มาตราฐนระดับตรรกะ (logic) ใน RS-232**

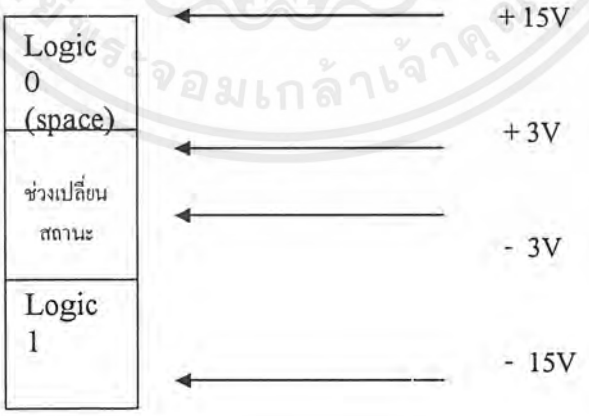
การส่งข้อมูลจากวงจรอินเตอร์เฟสมีลักษณะ " กลับหัว " กับวงจรที่ใช้งานกันอยู่ทั่วไป โดยความสัมพันธ์ระหว่างแรงดันกับค่าตรรกะบนอินเตอร์เฟสคือ แรงดันบวกบนอินเตอร์เฟสจะถูกแทนด้วย 0 ในขณะที่แรงดันลบแทนด้วยค่า 1 ดังรูปที่ 3.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 3.7 แสดงคำจำกัดความค่าตรรกะที่เอาต์พุตของ RS-232

ขอให้สังเกตค่าตรรกะที่กลับกันให้ดีคือ แรงดันลบแทนด้วยค่า 1 และแรงดันบวกแทนด้วยค่า 0 เพื่อให้แน่ใจค่าตรรกะ 0 แรงดันไฟที่เอาต์พุตจะต้องอยู่ในช่วง +5V ถึง +15V และในทำนองเดียวกันในการแทนระดับ 1 ระดับแรงดันที่เอาต์พุตจะต้องอยู่ในช่วง –5V ถึง –15V สำหรับช่องว่างหรือ dead-band ที่อยู่ในช่วง +5V ถึง –5V มีชื่อเรียกว่า ช่วงเปลี่ยนสถานะ (transition region) เป็นที่ไม่สามารถกำหนดค่าตรรกะได้ ซึ่งหมายความว่าค่าแรงดันเอาต์พุตในช่วง +5V ถึง –5V นั้นอาจถูกแปลความหมายให้เป็น 0 หรือ 1 ก็ได้ ดังรูปที่ 3.8




รูปที่ 3.8 แสดงคำจำกัดความค่าตรรกะที่อินพุตของ RS-232

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อแตกต่างเพียงข้อเดียวระหว่างคำจำกัดความสำหรับอินพุตกับเอาต์พุตคือ ความกว้างของช่วงเปลี่ยนสถานะ (transition region) โดยช่วงที่ไม่สามารถกำหนดค่าตรรกะได้ของอินพุตกว้างเพียง 6 V ( +3V ถึง –3V ) ในขณะที่ช่วงเดียวกันนี้สำหรับเอาต์พุตกว้างถึง 10 V ( + 5V ถึง –5V ) ซึ่งความแตกต่างนี้มีความสำคัญอย่างมากทีเดียว

**ช่วงการยอมรับสัญญาณรบกวน**

ความแตกต่างระหว่างคำจำกัดความของแรงดันต่ำสุดที่วงจรยอมรับได้เรียกว่า ช่วงการยอมรับสัญญาณรบกวน (noise margin) ซึ่งหมายความว่าวงจรยอมให้มีสัญญาณรบกวนออกจากเอาต์พุตเข้าสู่อินพุตได้โดยไม่ส่งผลต่อค่าตรรกะที่อินพุต ซึ่งคุณสมบัติข้อนี้มีประโยชน์มากในเวลาที่จำเป็นต้องเดินสายข้อมูลผ่านอุปกรณ์ที่เป็นตัวสร้างสัญญาณรบกวน เช่น มอเตอร์ไฟฟ้า หลอดไฟฟลูออเรสเซนต์ วงจรหรี่ไฟ และอุปกรณ์การสื่อสารต่าง ๆ

ส่วนต่างระหว่างช่วงเปลี่ยนสถานะของอินพุตและเอาต์พุตนอกจากจะทำหน้าที่เป็นช่วงยอมรับสัญญาณรบกวนแล้ว ยังทำหน้าที่เป็นช่วงปลอดภัย (safety margin) ด้วย โดยการให้แรงดันเผื่อสำหรับแรงดันที่ตกคร่อมสายเคเบิล ทำให้วงจรสามารถรับแรงดันที่ลดลงจากเอาต์พุตได้ถึงสองโวลต์โดยข้อมูลไม่ตกเข้าสู่ช่วงที่กำหนดตรรกะไม่ได้ของอินพุต

เนื่องจากแรงดันไฟกระแสตรง (direct current voltage) สูญเสียไปน้อยมากในสายเคเบิลจนสามารถตัดทิ้งไปได้แม้ในสายไฟขนาดยาวๆ ดังนั้นมาตรฐาน RS-232 จึงมีข้อกำหนดสำหรับสัญญาณควบคุมน้อยกว่าสัญญาณข้อมูล เนื่องจากสัญญาณควบคุมและสัญญาณการแฮนด์เชคเป็นสัญญาณแรงดันไฟฟ้ากระแสตรง

**Mark และ Space**

การรักษาระดับกระแสไฟให้คงที่ในระหว่างช่วงรอทำงาน (idle) ซึ่งเป็นช่วงที่ไม่มีการส่งข้อมูลออกมาของการส่งข้อมูล สามารถเพิ่มความน่าเชื่อถือของอุปกรณ์จะเพิ่มขึ้นอย่างมาก การส่งข้อมูลสามารถทำได้โดยการขัดจังหวะ (interrupt) กระแสไฟขณะหยุดรอการทำงานนี้ (มีกระแสไฟไหล) ได้มีการกำหนดให้มีชื่อว่า Mark ในทางกลับกัน เมื่อไม่มีกระแสไหล (เช่นในช่วงที่มีการส่งข้อมูลจริง) สถานะทางตรรกะได้ถูกกำหนดให้มีค่าเป็น 0 หรือสถานะ Space ( Mark = logic “1” , Space = logic “0”)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.2 ตารางมาตรฐานของการใช้แรงดันไฟฟ้า RS-232

| แรงดันไฟฟ้า | สถานภาพลอจิก | สถานภาพของสัญญาณ | ฟังก์ชั่นในการควบคุม |
|---|---|---|---|
| บวก | 0 | Mark | on |
| ลบ | 1 | Space | off |

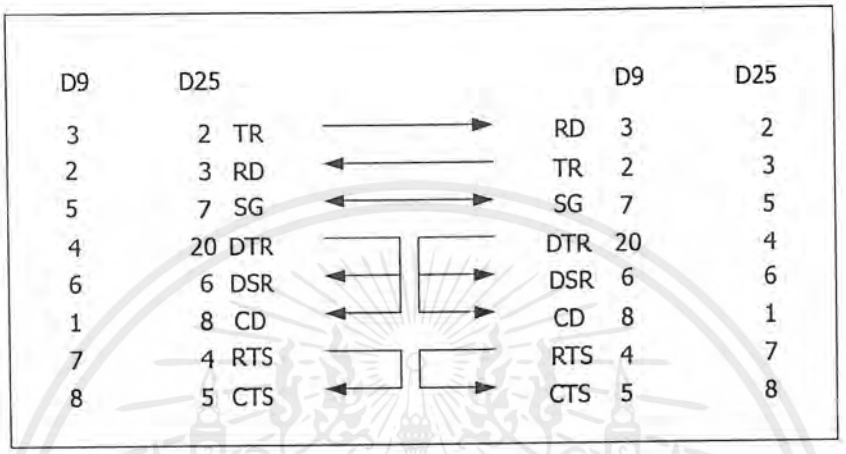
**UART ( Universal Asynchronous Receiver/Transmitter )** เป็นอุปกรณ์ที่มีความสำคัญมากในการสื่อสารแบบอนุกรม มีหน้าที่หลักอยู่ 3 อย่างที่สำคัญคือ

- Transmitter ( ตัวส่งข้อมูล ) ทำหน้าที่แปลงไบต์ข้อมูลขนาด 8 บิตไปเป็นอนุกรมของข้อมูล 8 บิต
- Receiver ( ตัวรับข้อมูล ) จะทำหน้าที่ตรงกันข้ามกับ Transmitter คือทำหน้าที่แปลงข้อมูลขนาด 8 บิต ไปเป็นไบต์ข้อมูล
- Controller Status มีหน้าที่ในการเฝ้าดูสถานะทางตรรกะของขาอินพุทต่างๆ และเมื่อโปรแกรมถูกเรียก ก็จะทำหน้าที่เปลี่ยนสถานะทางตรรกะของขาเอาต์พุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 3.5 ลักษณะการเชื่อมต่อและหน้าที่การทำงานของแต่ละขาที่สำคัญ

ในปัจจุบันพอร์ตอนุกรมนั้นจะมีอยู่ด้วยกัน 2 ขนาดคือคอนเน็คเตอร์แบบ D-type ตัวผู้ขนาด 25 Pin และ ตัวผู้ขนาด 9 Pin ซึ่งแสดงการเชื่อมต่อดังรูปที่ 3.9




รูปที่ 3.9 แสดงการเชื่อมต่อทั้งแบบ 25 Pin และ 9 Pin

**หน้าที่การทำงานของแต่ละขา (ของชนิด 25 Pin) โดยพิจารณาด้าน DTE**

- ขา 2 Transmitted Data ( TD ) ส่งข้อมูลจาก DTE ไป DCE
- ขา 3 Received Data ( RD ) ส่งข้อมูลจาก DCE ไป DTE
- ขา 4 Request to Send ( RTS ) เอาต์พุตอเนกประสงค์ สามารถนำไปประยุกต์ใช้ได้หลากหลาย ในโมเด็มแบบ half duplex ใช้สัญญาณนี้แสดงความต้องการส่งข้อมูล
- ขา 5 Clear to Send (CTS ) อินพุตอเนกประสงค์ นำไปใช้งานได้หลากหลาย ในโมเด็มแบบ half duplex สัญญาณนี้ใช้อนุญาตให้ส่งข้อมูลได้
- ขา 6 Data Set Ready ( DSR ) อินพุตอเนกประสงค์ที่ใช้แจ้ง DTE ว่าอุปกรณ์ DCE มีไฟเลี้ยงและพร้อมที่จะทำงาน
- ขา 7 Signal Ground ( SG ) จุดอ้างอิงแรงดันสำหรับทุกสัญญาณในกระบวนการอินเฟส ( ต้องมี )
- ขา 8 Data Carrier Detect ( CD ) สำหรับโมเด็มจะส่งสัญญาณ DCD เมื่อมันรับรู้การติดต่อกับโมเด็มที่อยู่ห่างออกไป สำหรับ DTE สัญญาณ DCD จะถูกนำไปใช้ในการยกเลิกการรับข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ขา 20 Data Terminal Ready ( DTR ) เอาต์พุตอเนกประสงค์โดยทั่วไปใช้เป็นสัญญาณบอก DCE ว่าอุปกรณ์ DTE ที่มันอินเตอร์เฟสด้วยมีไฟเลี้ยงและพร้อมที่จะทำงาน

นอกจาก 9 ขาที่กล่าวถึงข้างต้น ยังมีขาอื่นๆ อีกที่ใช้ในการอินเตอร์เฟส แต่สัญญาณสำคัญต่างๆ ที่มีการนำไปใช้เป็นประจำก็มักมาจาก 9 ขานี้เท่านั้น ซึ่งของขาคอนเน็กเตอร์ขา 2,3,4,5,6,7,8 และ 20 ว่ากลุ่ม " BIGEIGHT " ส่วนขาสัญญาณอื่นๆ มีไว้สำหรับเป็นทางเลือกที่ผู้ผลิต แต่ละรายจะนำไปประยุกต์ใช้ได้ตามความต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 4
# ฮาร์ดแวร์

## 4.1 อุปกรณ์ที่ใช้

### วงจรกำหนดสัญญาณนาฬิกา

ใช้ผลึก Crystal ความถี่ 11.0592 MHz เป็นออสซิเลเตอร์

### วงจรแปลงสัญญาณแบบ RS-232 ไปเป็น TTL

ใช้ไอซี Max 232 ซึ่งมีข้อดีคือมีทั้ง driver และ receiver ในตัวเองและใช้ โวลท์เตจระดับเดียวคือ 5 โวลท์ การนำเอา max 232 ไปใช้นั้นต้องมีการต่อตัวเก็บประจุเพิ่มเข้าไปอีกเล็กน้อย

### วงจร Latch ข้อมูล

ใช้ไอซี 74374 ทำหน้าที่เป็น ดี-ฟลิปฟลอป (D- FlipFlop) เพื่อคงค่าของข้อมูลขนาด 8 บิท ก่อนที่จะส่งให้กับวงจรรีเลย์

### Microcontroller 8 bit At892051

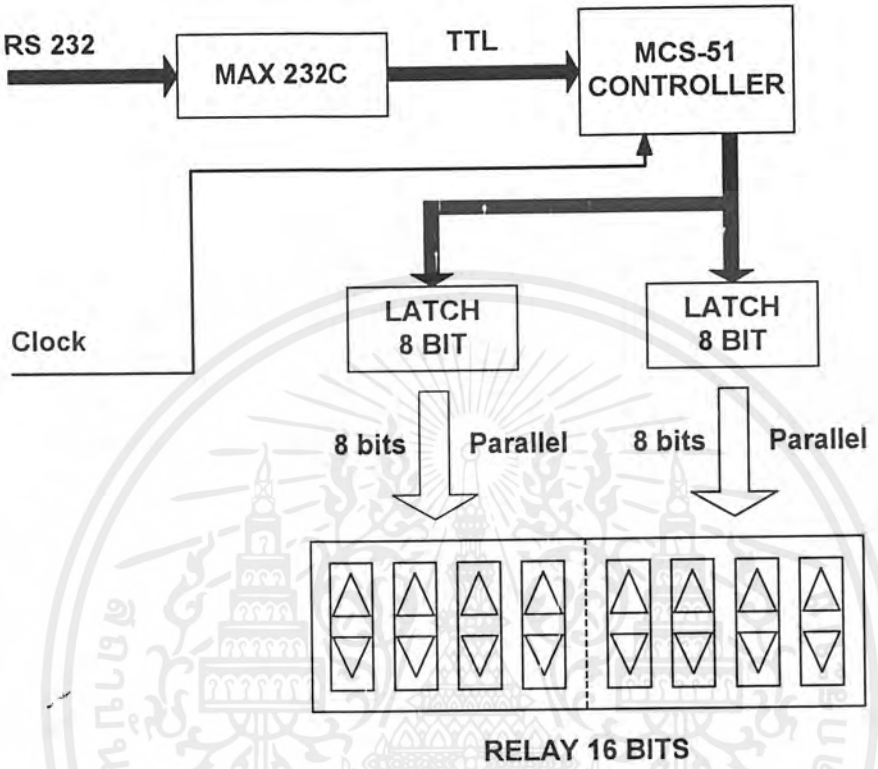
At89C2051 เป็นไมโครคอนโทรลเลอร์ 8 บิท ขนาด 20 ขา สร้างด้วย Atmel's high Density nonvolatile memory technology โดยมีหน่วยความจำภายในขนาด 2kbyte ที่เรียกว่า อีพรอม (EPROM : Erasable and Programmable Read only memory) ผลิตโดย Atmel ชุดคำสั่งและสถาปัตยกรรมภายในเหมือนไมโครคอนโทรลเลอร์ตระกูล MCS-51 ซึ่งผลิตโดย Intel โดยสามารถเขียนและลบได้ไม่น้อยกว่า 1000 ครั้ง

## 4.2 คุณสมบัติของ AT89C2051

1. สามารถใช้แทนไมโครคอนโทรลเลอร์ตระกูล MCS-51
2. หน่วยความจำโปรแกรมเป็นแบบ flash memory ขนาด 2Kbyte โดยสามารถเขียนหรือลบโปรแกรมได้ถึง 1000 ครั้ง
3. ใช้ไฟเลี้ยงตั้งแต่ 2.7 - 6.0 โวลท์
4. ทำงานได้ในช่วงความถี่ 0 Hz ถึง 24 MHz
5. สามารถล็อคโปรแกรมได้ 2 ระดับ
6. มีแรมภายใน 128* 8 บิท

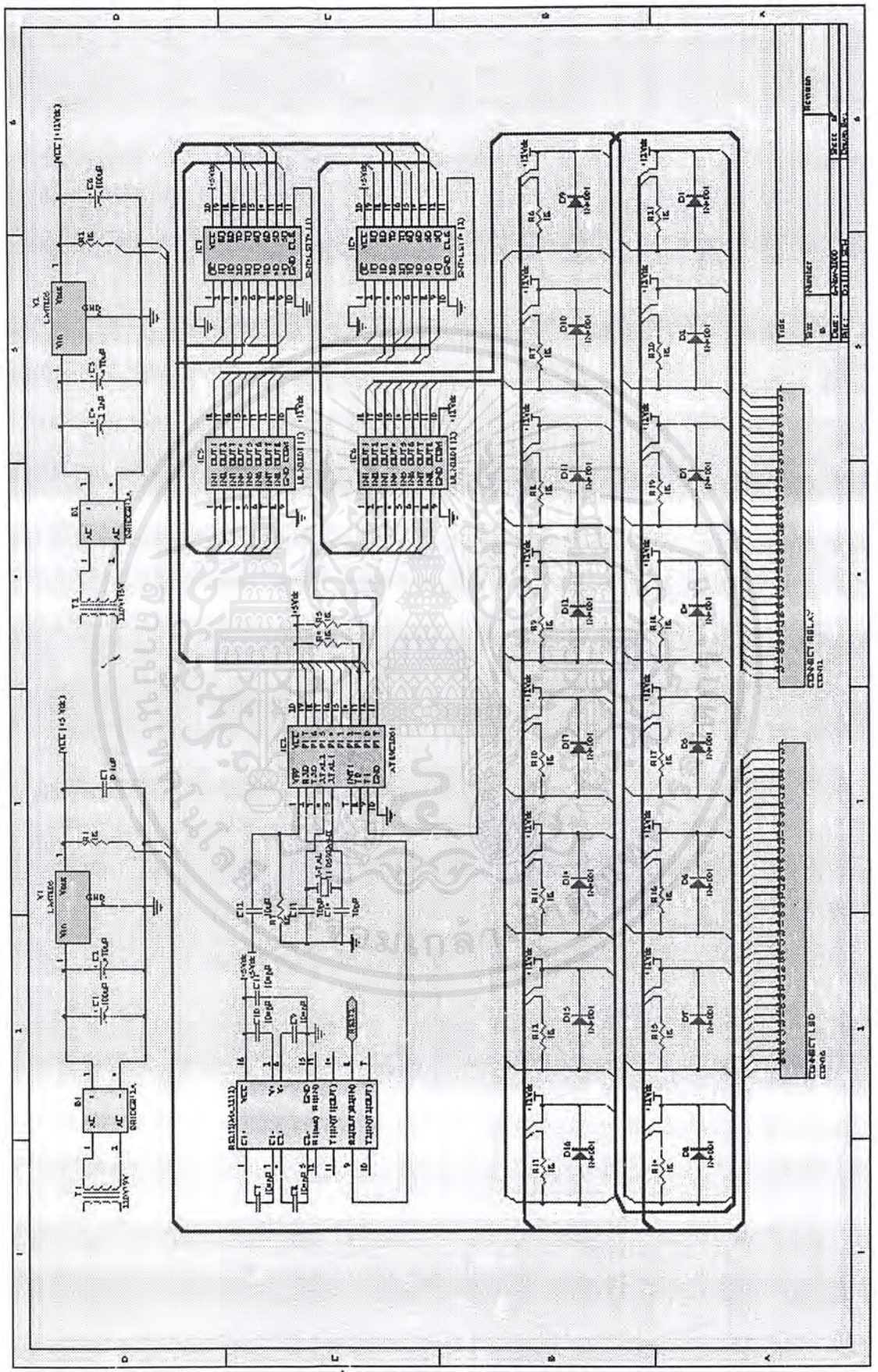
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. มีอินเทอร์รัพท์ (Interrupt) 5 ตำแหน่ง
8. มี Counter/Timer ขนาด 16 บิท 2 ตัว




รูปที่ 4.1 ส่วนประกอบของฮาร์ดแวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

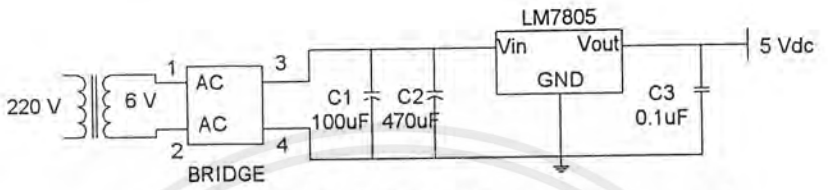


**รูปที่ 4.2 แสดงวงจรไมโครคอนโทรลเลอร์**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการ[illegible]ไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 4.3 แหล่งจ่ายไฟ

จากหม้อแปลงกระแสสลับ 220V / 6V โดยใช้ไดโอดบริดจ์ , ไอซี่ เบอร์ 7805 และตัวเก็บประจุต่อร่วมกันเป็นวงจรควบคุมแรงดัน เพื่อจ่ายโวลท์เตจกระแสตรง 5 Vdc เป็นไฟเลี้ยงให้กับวงจรไมโครคอนโทรลเลอร์



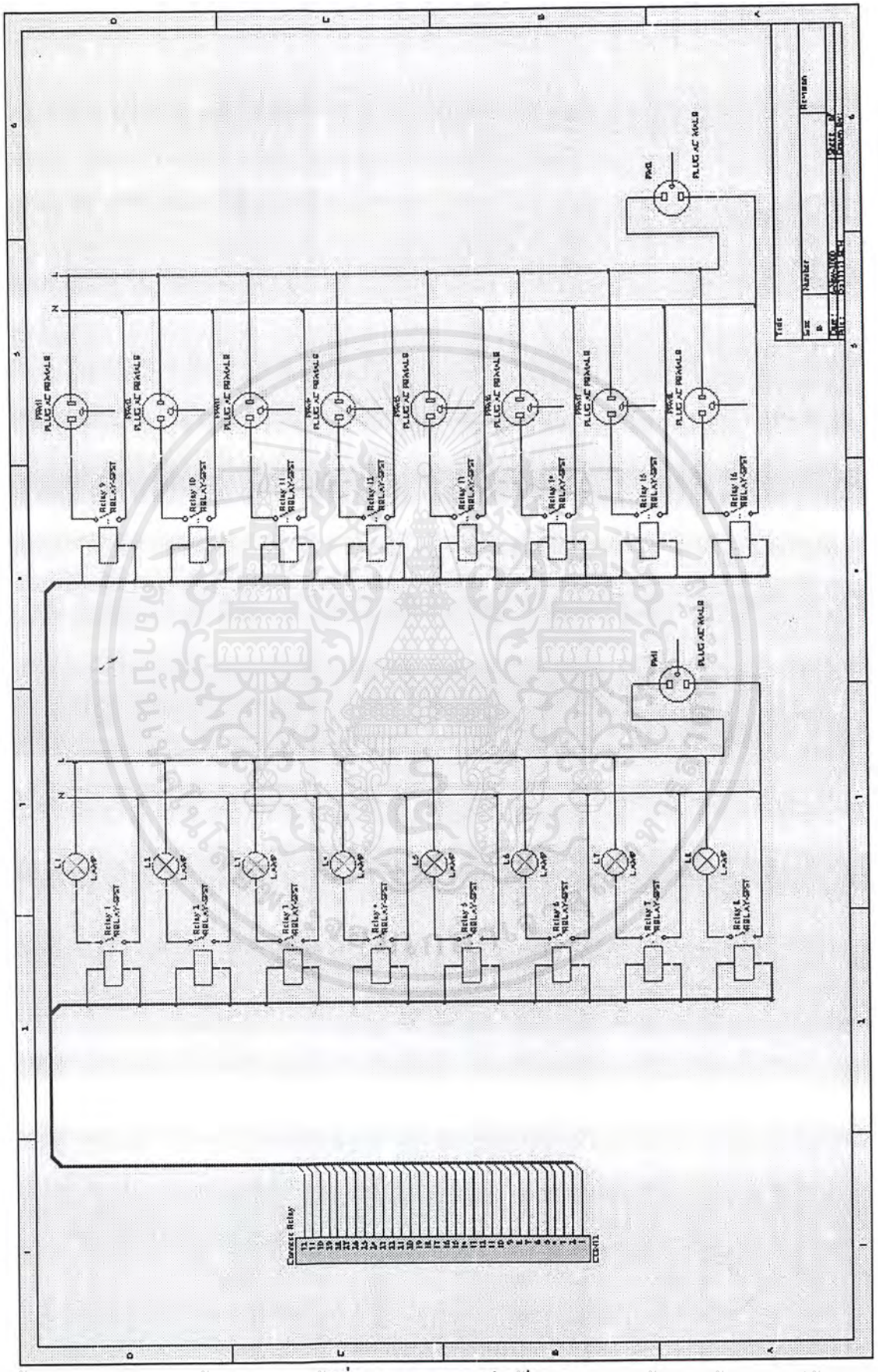

รูปที่ 4.3 แหล่งจ่ายไฟวงจรไมโครคอนโทรลเลอร์

## 4.4 วงจรรีเลย์

เอาท์พุทจากไอซี ULN 2803 AN ทั้ง 2 ตัวจะทำงานเสมือนเป็นกราวด์ของสวิทช์รีเลย์ เมื่อเอาท์พุทจาก latch บิทใดมีลอจิก 1 เพื่อผ่านคอนเวอร์เตอร์ของไอซี ULN 2803 AN จะเป็นศูนย์โวลท์จะมีกระแสไหล ผ่านขดลวดรีเลย์ ขดลวดของรีเลย์จะสร้างสนามแม่เหล็กดูดหน้าสัมผัสรีเลย์เข้าหากัน หน้าสัมผัสนี้จะถูกนำไปใช้เพื่อควบคุมการจ่ายกำลังให้กับเครื่องไฟฟ้าที่เราต้องการจะควบคุม

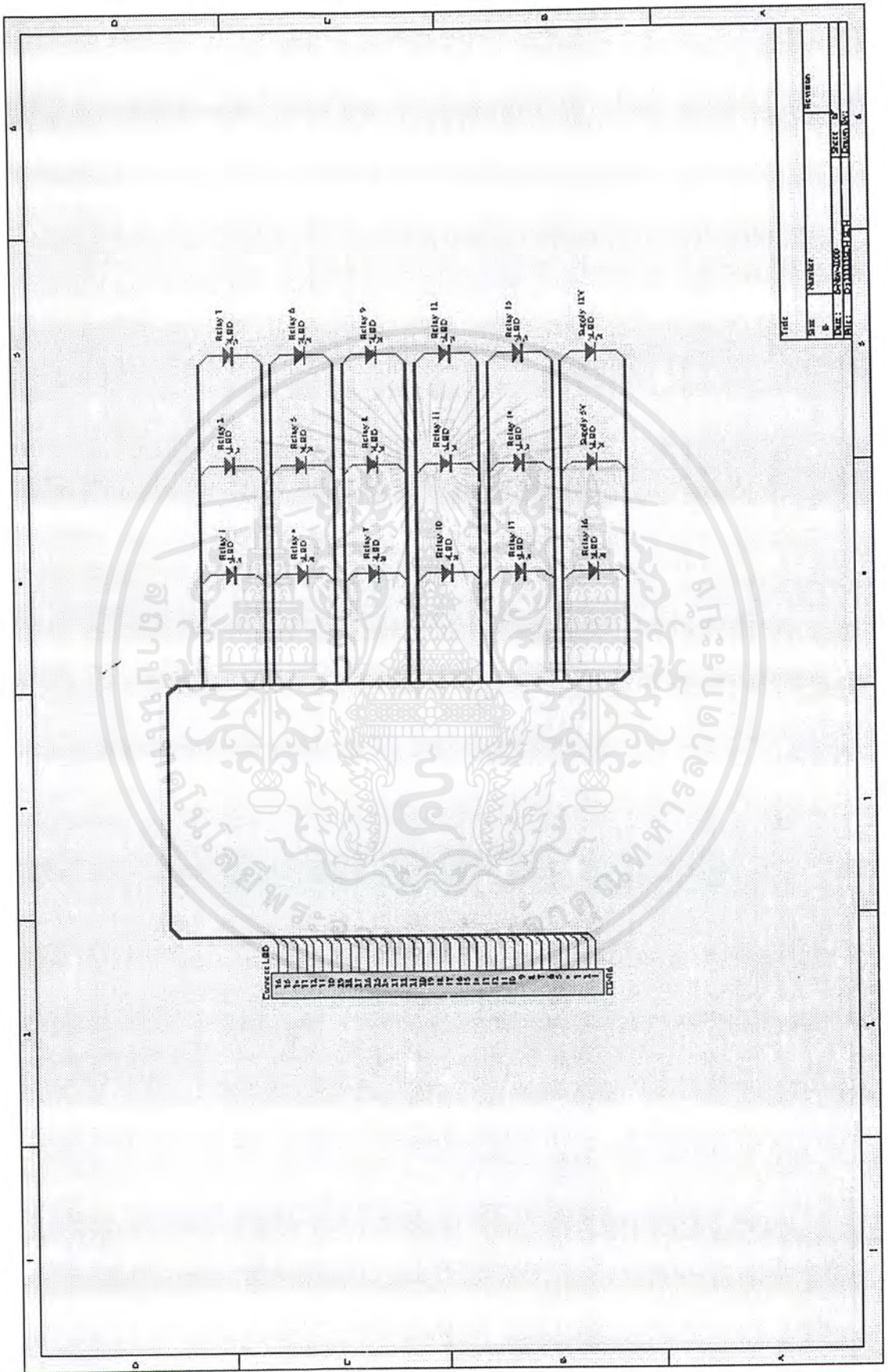
ให้อนุกรมหน้าสัมผัสเข้ากับแหล่งจ่ายแรงดัน 220 โวลท์กับหลอดไฟ และปลั๊กตัวเมียตามบอร์ดทดลอง ซึ่งสามารถควบคุมการเปิด-ปิด เครื่องไฟฟ้านี้ได้จากเทอร์มินัล ซึ่งอยู่ห่างออกไปได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**รูปที่ 4.4 แสดงวงจรรีเลย์**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

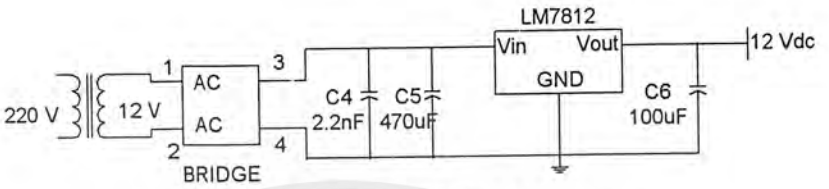


รูปที่ 4.5 แสดงวงจรรีเลย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 แหล่งจ่ายไฟสำหรับวงจรรีเลย์

จากหม้อแปลงกระแสสลับ 220 V / 12 V เราใช้ไดโอดบริดจ์, ไอซี่เบอร์ 7812 และตัวเก็บประจุต่อร่วมกันเป็นวงจรควบคุมแรงดัน เพื่อจ่ายโวลท์เตจกระแสตรง 12 Vdc เป็นไฟเลี้ยงให้กับวงจรรีเลย์



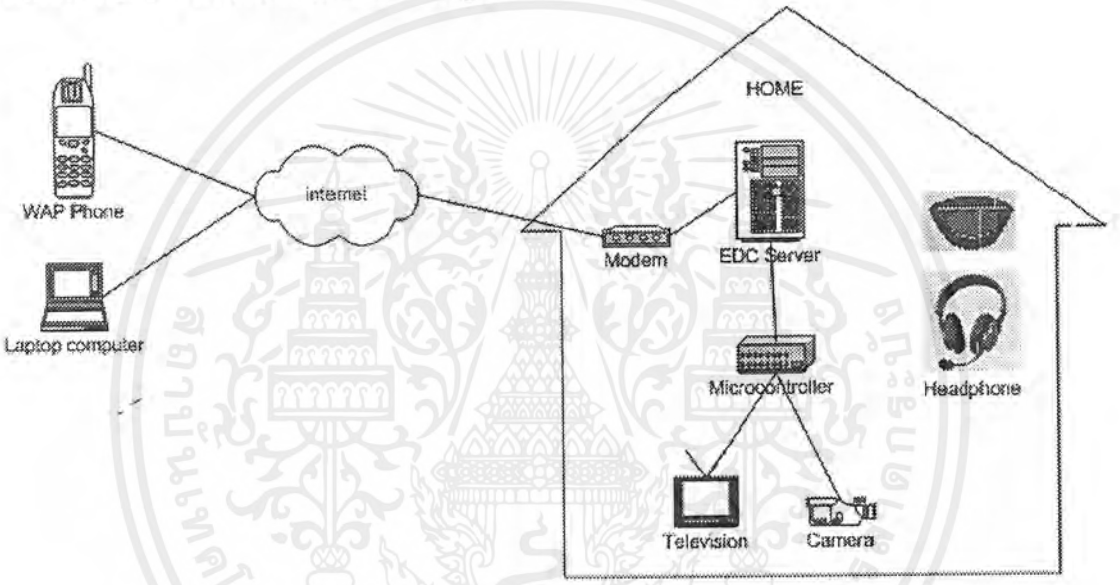

รูปที่ 4.6 แหล่งจ่ายไฟวงจรรีเลย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 5
# การออกแบบ

## 5.1 ภาพโดยรวมของระบบ

ระบบนี้เป็นการนำเอาความสามารถของ Internet มาประยุกต์ใช้กับระบบควบคุมภายในบ้าน ทำให้สามารถควบคุมอุปกรณ์ไฟฟ้าภายในบ้านได้อย่างสะดวกรวดเร็ว และไม่จำกัดอยู่เพียงแค่คอมพิวเตอร์ตั้งโต๊ะเพียงอย่างเดียว โดยได้มีการนำเอาอินเทอร์เฟซ อย่างอื่นมาช่วยให้การใช้งานได้สะดวกยิ่งขึ้น ภาพโดยรวมแสดงได้ดังรูป



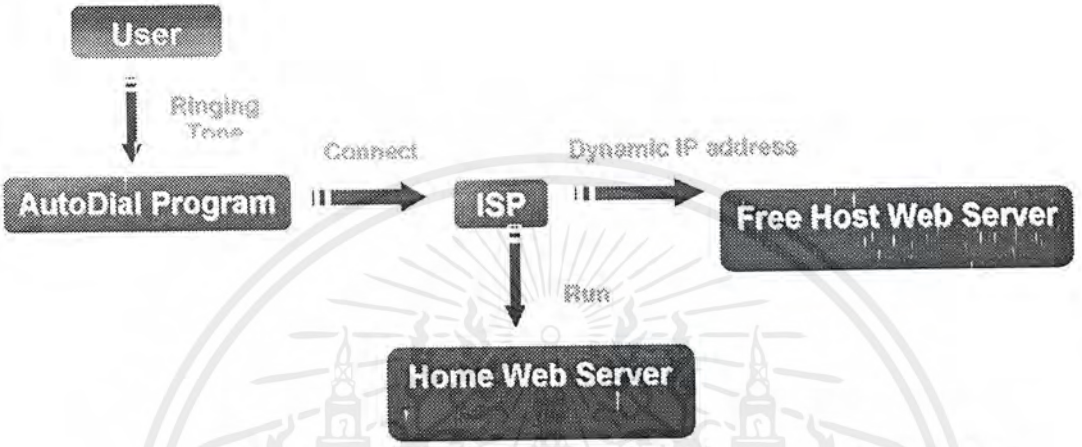

รูปที่ 5.1 Diagram แสดงภาพโดยรวมของโครงงาน

## 5.2 การออกแบบระบบ

เนื่องจากในระบบก่อนการควบคุมอุปกรณ์ไฟฟ้าระยะไกล จะต้องติดต่ออยู่กับเครื่อง server ที่ต้องเชื่อมต่ออินเทอร์เนตตลอดเวลา และมีเพียงช่องทางเดียวคือ web application แสดงผลในลักษณะ 2 มิติ เราจึงได้ทำการแก้ไข โดยทำให้สามารถสั่งให้ server ทำการเชื่อมต่ออินเทอร์เนตเมื่อต้องการควบคุมอุปกรณ์ไฟฟ้าเท่านั้น และสามารถทำการตั้งเวลาการทำงานล่วงหน้าได้ ซึ่งช่วยประหยัดค่าใช้จ่ายได้และได้เพิ่มช่องทางการควบคุมเพิ่มอีก 2 ช่องทาง คือ การควบคุมผ่านทางโทรศัพท์เคลื่อนที่ และผ่านทางเสียงพูดโดยตรงในกรณีที่อยู่ภายในบ้าน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานจะเริ่มจากผู้ใช้ทำการโทรศัพท์เข้ามาที่โมเดม ซึ่งมีโปรแกรมคอยรับสัญญาณ Ringing tone ที่ผ่านเข้ามาเมื่อครบ 6 ครั้ง ก็จะทำการติดต่อไปยัง ISP เมื่อติดต่อสำเร็จก็จะนำค่า IP ปัจจุบันที่ได้ไปทำการเก็บไว้ที่ Free host's database เพื่อใช้ในการเชื่อมต่อกับ server ภายในบ้านต่อไป



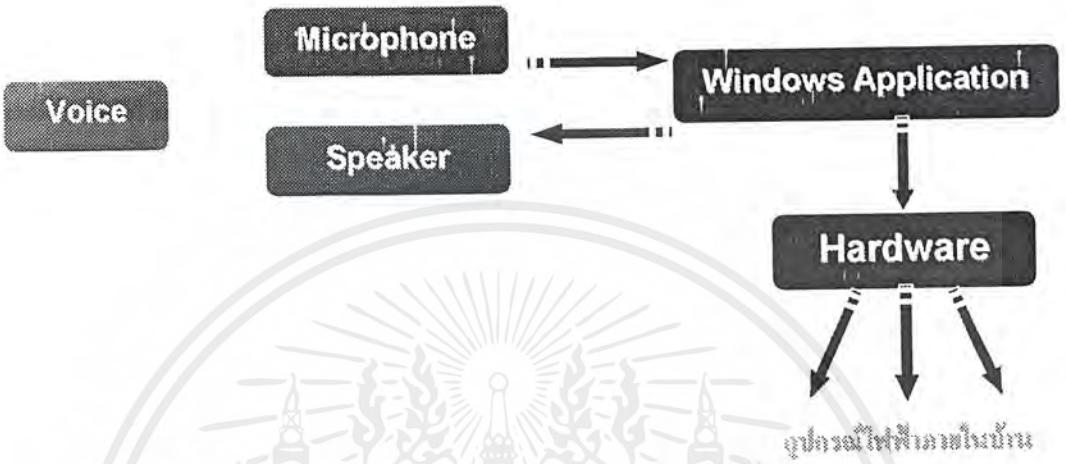

รูปที่ 5.2 ขั้นตอนการดำเนินการของ server เพื่อเชื่อมต่อเข้าสู่อินเทอร์เนต

เมื่อผู้ใช้จะทำการควบคุมอุปกรณ์ไฟฟ้าก็สามารถเชื่อมต่อมาที่ web page หรือ wap page ที่ Free host เพื่อทำการลิงค์ไปยัง web page หรือ wap page ของ server ภายในบ้าน และสามารถใช้ web application หรือ wap application ส่งคำสั่งไปที่ Hardware แล้วทำการควบคุมอุปกรณ์ไฟฟ้าภายในบ้าน

รูปที่ 5.3 ขั้นตอนการเชื่อมต่อกับ Home server ผ่าน Free host เพื่อควบคุมอุปกรณ์ไฟฟ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในกรณีที่เป็นการใช้งานภายในบ้าน ก็จะมีแอพพลิเคชั่นที่ server ทำการรับเสียงพูดจากผู้ใช้แล้วทำการวิเคราะห์แล้วจึงทำการควบคุมอุปกรณ์ไฟฟ้าตามคำสั่งที่ผู้ใช้ได้พูด




รูปที่ 5.4 ขั้นตอนการทำงานของระบบควบคุมอุปกรณ์ไฟฟ้าทางเสียงพูด

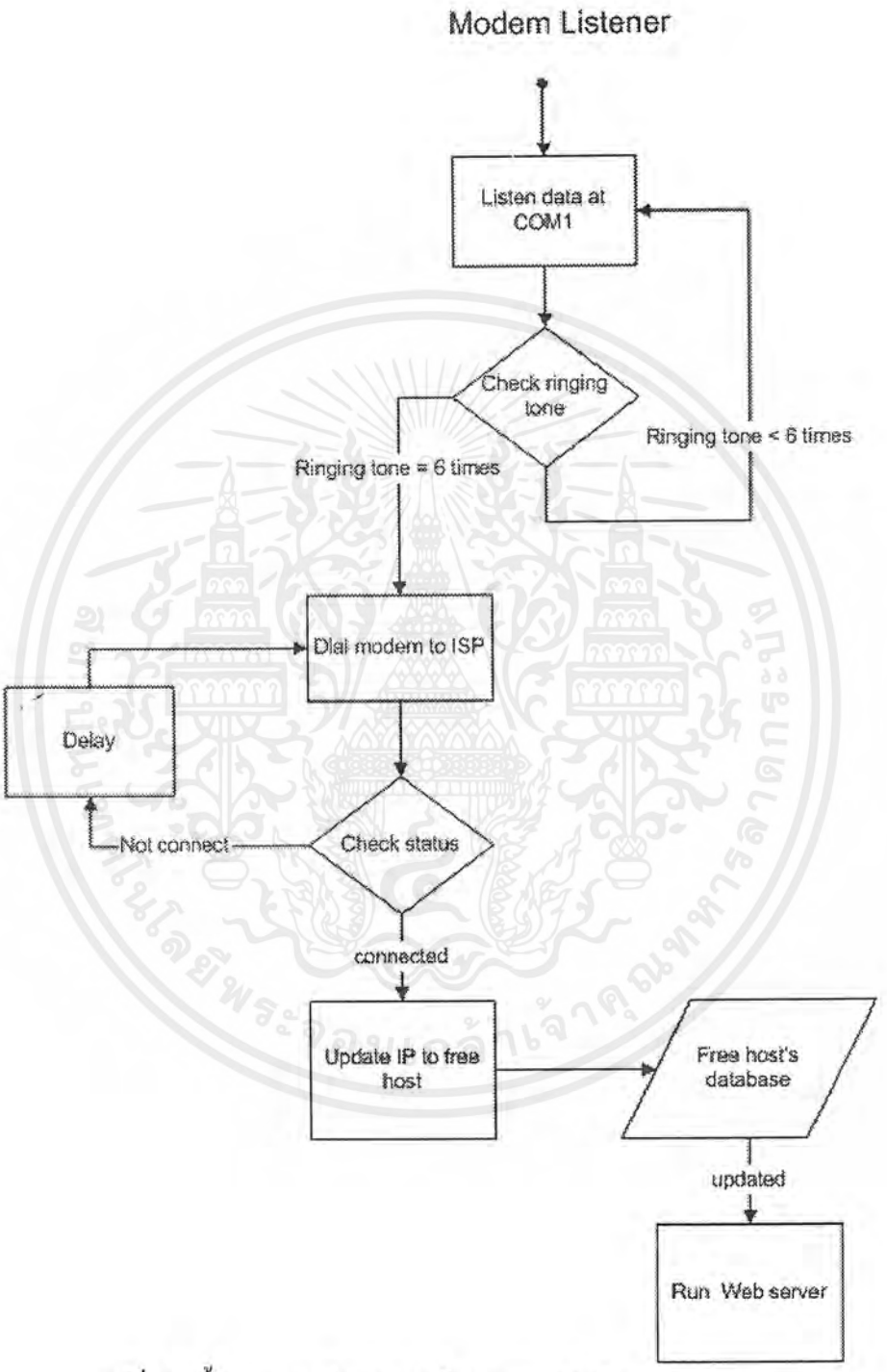
การออกแบบจะแบ่งออกเป็น 5 ส่วนหลัก ๆ คือ

1.ส่วนของการรอรับสัญญาณโทรศัพท์แล้วทำการติดต่อกับ ISP เพื่อต่อกับ Internet

โปรแกรม ModemListener จะออกแบบให้คอมพิวเตอร์ทำการติดต่อกับ ISP เมื่อทำการติดต่อได้แล้วก็จะนำ dynamic IP ที่ได้ไปทำการเก็บบันทึกไว้ที่ Free host's database โดยใช้ภาษา Java ทำการพัฒนาโปรแกรมในการรอรับสัญญาณ Ringing ของ modem เพื่อเป็นสัญญาณบอกให้ server ทำการติดต่อกับ ISP เมื่อติดต่อได้แล้วก็จะทำการส่งข้อมูล dynamic IP ที่ได้ไปให้กับ database เพื่อบันทึกข้อมูลหมายเลข IP ที่เป็นปัจจุบันของ server ที่ตั้งอยู่ภายในบ้านโดยอัตโนมัติ ทำให้ทุกครั้งที่เราทำการติดต่อกับ ISP ก็จะมีการ Update IP เสมอ ทำให้เราไม่จำเป็นต้องมี static IP เพื่อทำการตั้ง server ในการใช้งานอินเทอร์เฟซทั้ง Visual 3D และ Mobile Interface นั้นจะทำได้โดยทำการเรียกไปยัง website ของ Free host ซึ่งใช้ servlet ที่ทำการติดต่อกับดาต้าเบสและเรียกข้อมูล IP ที่เป็นของ server ปัจจุบันขึ้นมาและทำการ redirect ไปทำการติดต่อกับ web server ที่ตั้งอยู่ภายในบ้านต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กระบวนการทำงานของโปรแกรมแสดงได้ดังรูป

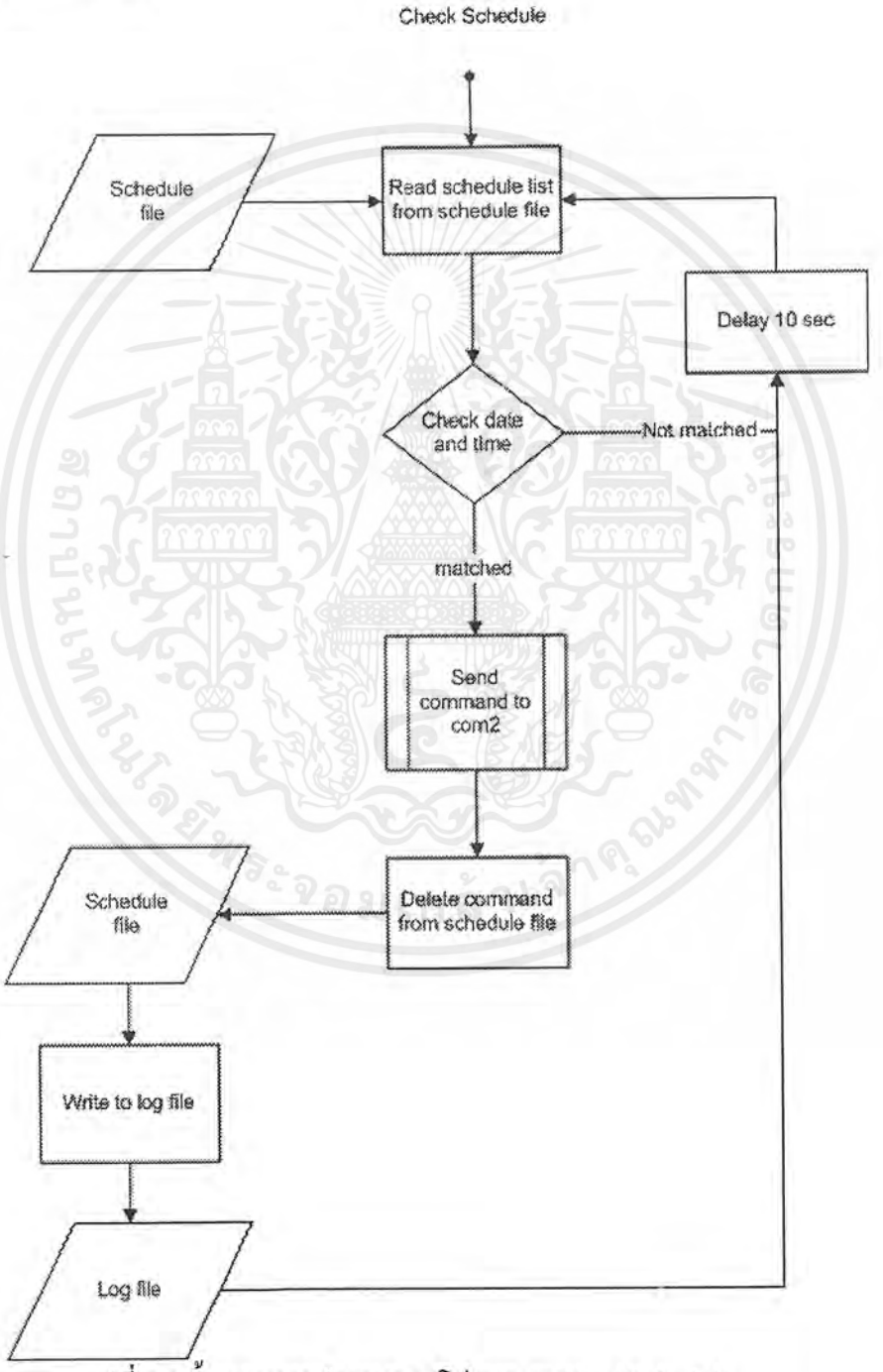



รูปที่ 5.5 ขั้นตอนการทำงานของโปรแกรม ModemListener

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ส่วนของการตรวจสอบรายการการตั้งค่าการทำงานกับเวลาปัจจุบัน

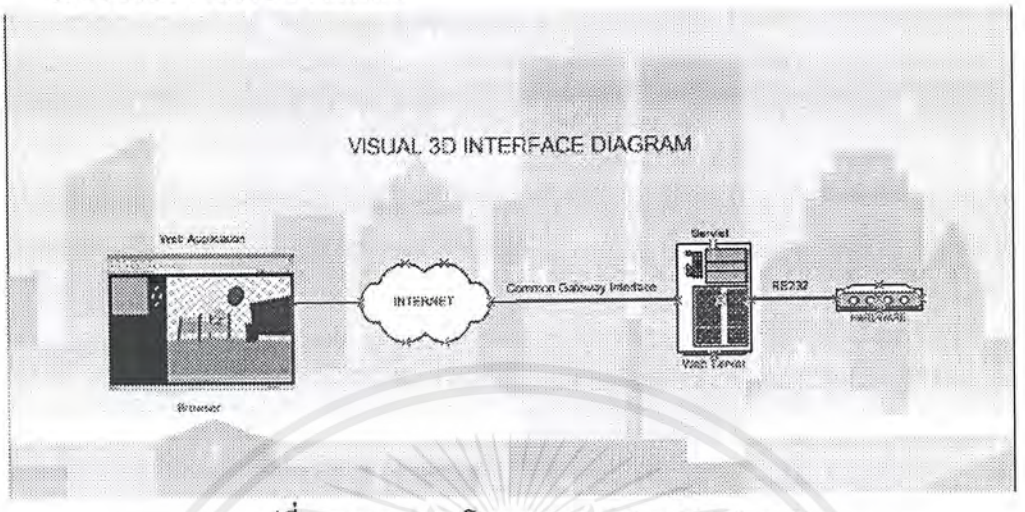
โปรแกรม check schedule ซึ่งเป็นโปรแกรมที่ทำการตรวจสอบค่าการตั้งเวลาการทำงานของอุปกรณ์ไฟฟ้าที่บันทึกไว้ โดยจะทำการตรวจสอบทุก ๆ 10 วินาที เมื่อมีค่าการตั้งเวลาที่ตรงกับเวลาปัจจุบัน ก็จะทำการนำค่าที่ตั้งไว้ ส่งผ่าน COM2 ไปให้แก่ อุปกรณ์ควบคุมเพื่อทำการสั่งให้อุปกรณ์ไฟฟ้าทำงาน




รูปที่ 5.6 ขั้นตอนการทำงานของโปรแกรม CheckSchedule

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ส่วนของ Visual 3D Interface




รูปที่ 5.7 แสดงภาพโดยรวมของ Visual 3D Interface

จะออกแบบให้เป็นในลักษณะของ Webbased Application โดยจะมีการพัฒนาเวปแอพพลิเคชั่น (Web Application) และพัฒนา Visual 3D Interface โดยใช้ Adobe Atmosphere (แอพพลิเคชั่นที่ใช้ในการสร้างเวปแบบ 3D) เพื่อเป็นอินเทอร์เฟซให้แก่ผู้ใช้ ผ่านทางบราวเซอร์ โดยจะทำการพัฒนา Web page ที่มีการผัง ActiveX Control ของ Adobe Atmosphere เพื่อใช้ในการแสดงอินเทอร์เฟซ 3D และเก็บไฟล์เหล่านี้ไว้ที่ Web Server เมื่อผู้ใช้ทำการคลิกที่อุปกรณ์ไฟฟ้าภายในอินเทอร์เฟซ Visual 3D ก็จะทำการส่ง request ไปทำการติดต่อกับ Servlet แล้วเรียกใช้ Comm API ทำการส่งข้อมูลไปที่ Serial port ผ่าน RS232 ไปที่ Hardware ควบคุมอุปกรณ์ไฟฟ้า

และสามารถทำการกำหนดชื่ออุปกรณ์ไฟฟ้า (Config) ตั้งเวลาการทำงาน (Schedule) หรือเรียกดูการทำงานที่ผ่านไปแล้ว (History) ได้

ทางบราวเซอร์ในลักษณะของ Visual 3D Webbased Application มีลักษณะการทำงานดังนี้

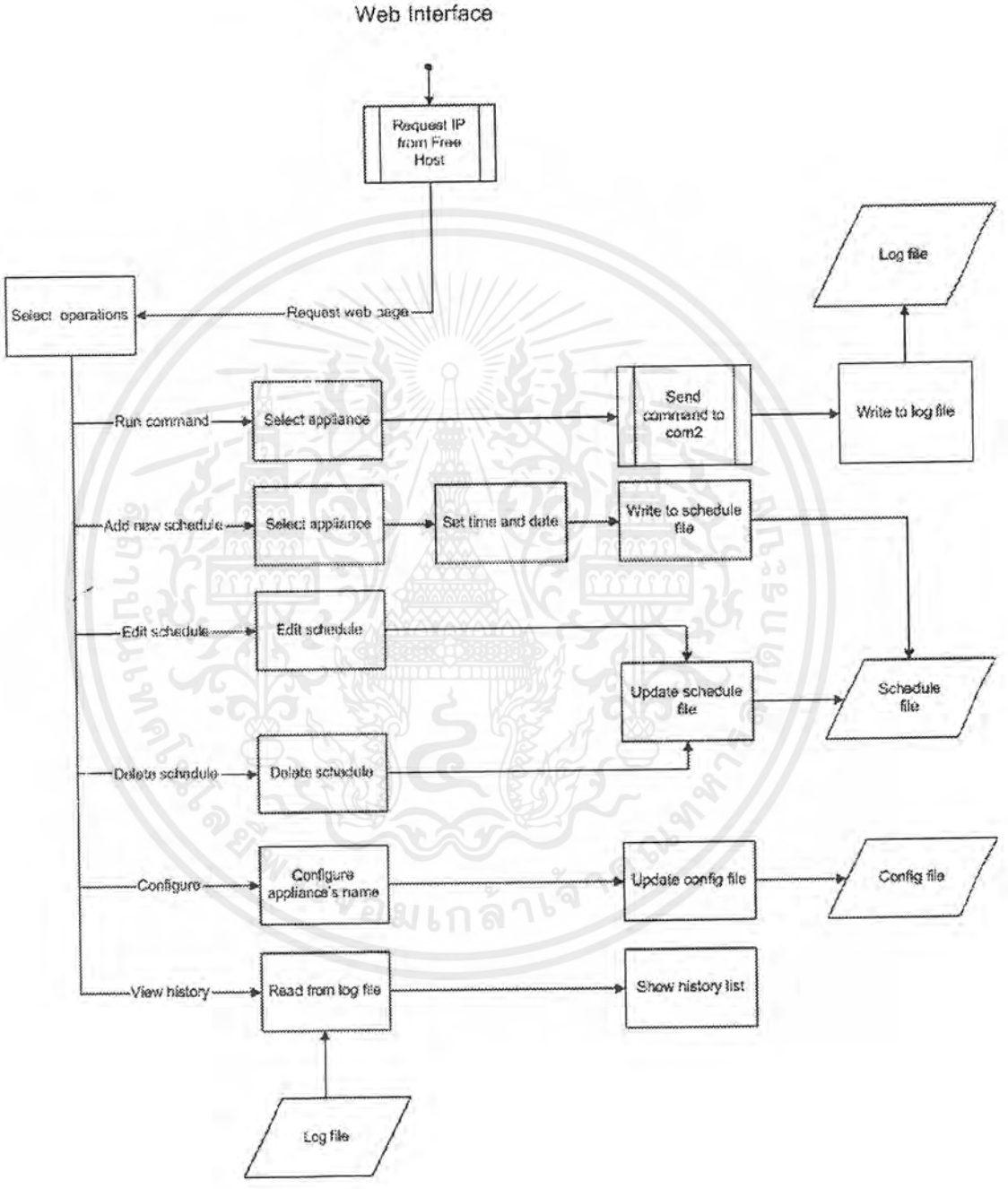
- ผู้ใช้ทำการเข้าไปที่ http://www.freehost.com/web

จะปรากฏหน้าเวปที่แสดงถึงลิงค์ที่จะทำการ redirect ไปยัง Home Web server เพื่อเรียกหน้าเวปที่แสดงอินเทอร์เฟซ Visual 3D ขึ้นมา

- จากนั้นผู้ใช้สามารถใช้เมาส์ หรือ คีย์บอร์ด ทำการเลื่อนเปรียบเสมือนกับว่ากำลังเดินอยู่ภายในห้องที่จำลองตำแหน่งการวางของอุปกรณ์ไฟฟ้าภายในบ้าน หรือผู้ใช้สามารถทำการเลือกรายการอุปกรณ์ไฟฟ้าที่ต้องการปิด-เปิด และทำการกดปุ่ม Run เพื่อสั่งให้ทำงานทันที หรือจะตั้งวัน-เวลา และเลือกรายการอุปกรณ์ไฟฟ้าที่จะทำการปิด-เปิด ได้โดยกดปุ่ม Add หรือทำการกำหนดชื่ออุปกรณ์ไฟฟ้าที่ต่ออยู่กับสวิตช์ได้ โดยกดปุ่ม Config หรือทำการเรียกดูสถานะที่ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

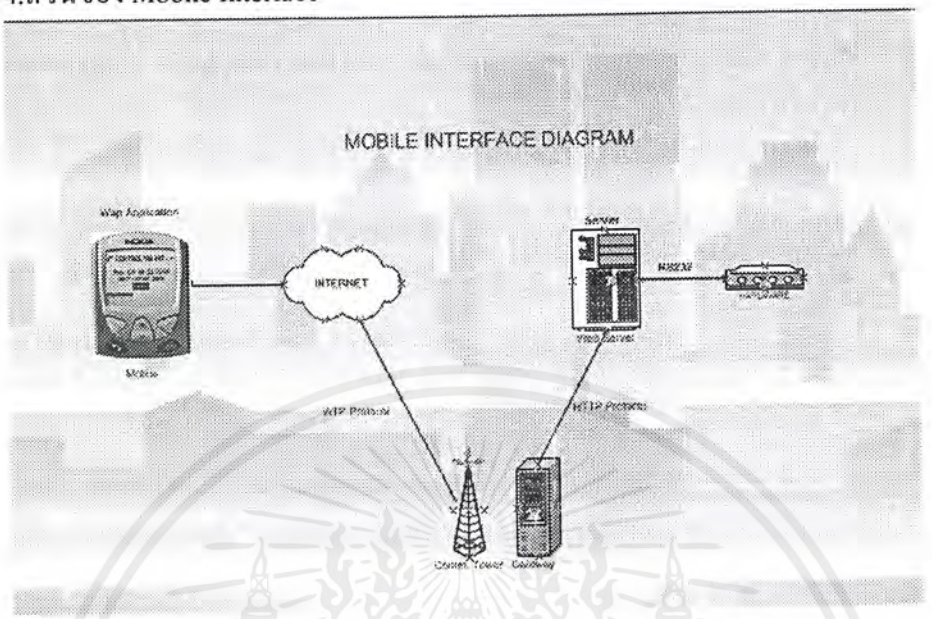
ทำงานผ่านไปแล้วได้โดยกดปุ่ม History และผู้ใช้ยังสามารถทำการเรียกดูรายละเอียดของการตั้งเวลาพร้อมทั้งสามารถทำการเปลี่ยนแปลงหรือลบ ได้โดยกดที่ view และ edit ตามลำดับ




รูปที่ 5.8 ขั้นตอนการทำงานของ Web Application

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.ส่วนของ Mobile Interface




รูปที่ 5.9 แสดงภาพโดยรวมของ Mobile Interface

เนื่องจากความแตกต่างกันของการแสดงผลและระบบการทำงานของโทรศัพท์เคลื่อนที่แต่ละรุ่น ทำให้การออกแบบส่วน user interface ของ Wap Application อาจจะเกิดปัญหาขึ้นเช่น ปัญหาของเมนู option ของโทรศัพท์เคลื่อนที่บางรุ่นอาจไม่สามารถแสดงได้ในอีกรุ่นหนึ่ง ทำให้ผู้ใช้ไม่สามารถเรียกใช้เมนูนั้น ๆ ได้ เราจึงแก้ปัญหาโดยมีการเรียกใช้เมนูคำสั่งทั้งในส่วนเมนู option และทำเป็น anchor link (ลิงค์คำสั่ง) ทำให้แก้ปัญหานี้ได้

ส่วนของการออกแบบและพัฒนา Wap Application นั้นได้พัฒนาไว้ที่ Home Web Server เมื่อมีการ request จากผู้ใช้ข้อมูลการ request จะถูกส่งผ่าน WTP Protocol ไปยัง gateway เพื่อทำการแปลงจาก WTP Protocol ให้เป็น HTTP Protocol ทำการเรียกไปยัง Web Server จากนั้น Web Server จะทำการ response ข้อมูล wap page ไปยัง gateway เพื่อทำการแปลงจาก HTTP Protocol ให้เป็น WTP Protocol ไปแสดงผลที่ Mobile Device เมื่อผู้ใช้ ทำการเลือกอุปกรณ์ไฟฟ้าที่จะทำการประมวลผลก็จะส่ง request ไปทำการประมวลผลที่ Servlet แล้วใช้ Comm API ทำการติดต่อกับ Serial port ไปยัง hardware เพื่อควบคุมอุปกรณ์ไฟฟ้า

ผู้ใช้สามารถทำการเรียกดูสถานะคำสั่งล่าสุด (current status) ของอุปกรณ์ไฟฟ้าได้ และยังสามารถตั้งเวลาการทำงาน (schedule) และเรียกดูการทำงานที่ผ่านไปแล้ว (history) ได้

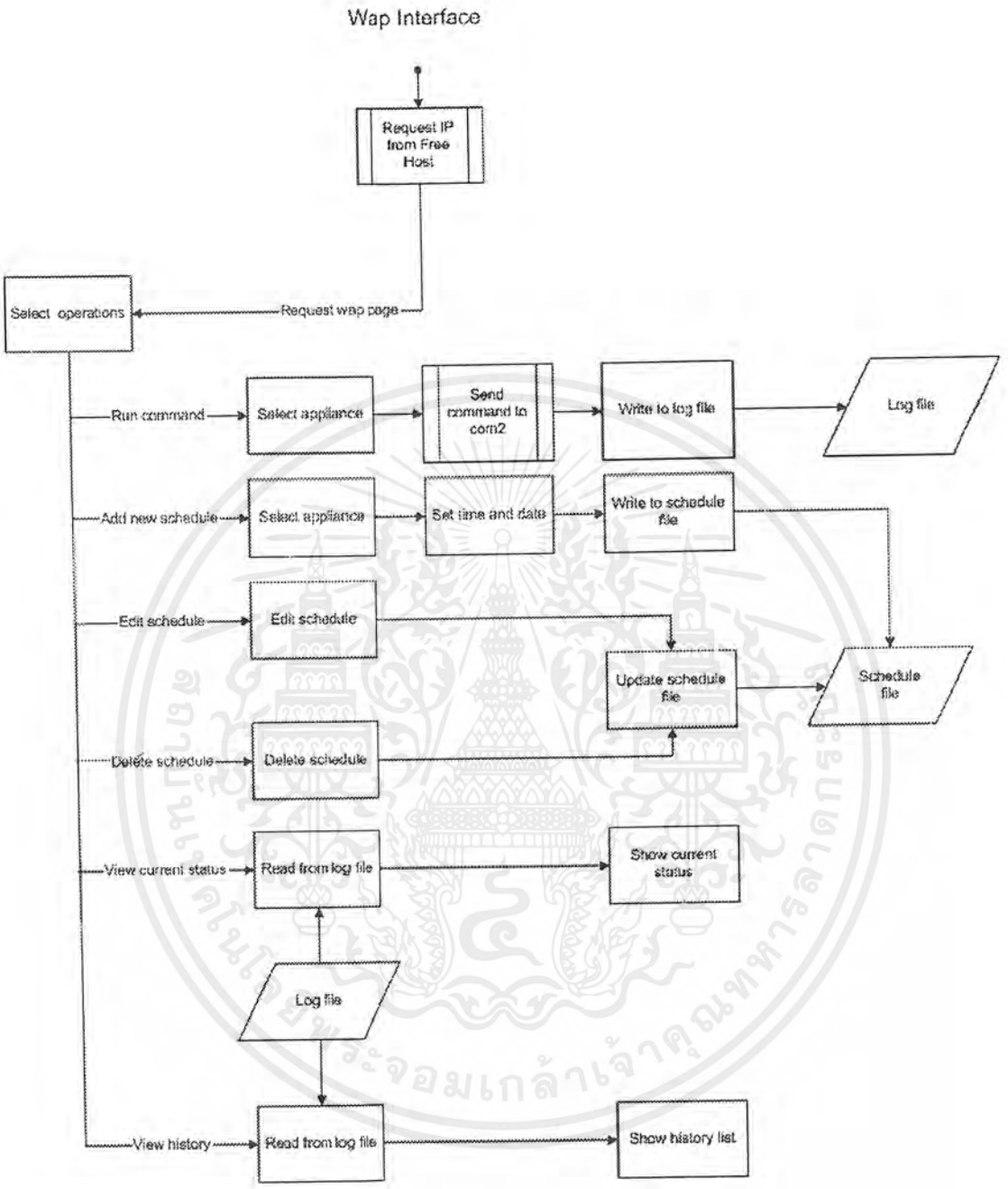
ทาง Mobile Interface โดยใช้ Wap Technology มีขั้นตอนการทำงานดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ผู้ใช้ทำการเรียกไปยัง http://www.freehost.com/wap ก็จะได้หน้าแวป ที่แสดงถึง link ที่จะทำการ link ไปยังหน้าแวป ที่ web server จากนั้นผู้ใช้สามารถเลือกเมนูที่ต้องการใช้งานได้เช่น เลือกเมนู

| | |
|---|---|
| Command | เพื่อเลือกอุปกรณ์ไฟฟ้าที่ต้องการควบคุม เมื่อผู้ใช้ทำการ เลือก send แล้ว Mobile device จะทำการส่ง request และ parameter ที่ได้จากการเลือกไปยัง Web server แล้วส่งไปยัง servlet เพื่อไปทำการส่งสัญญาณไปยังอุปกรณ์ควบคุมเพื่อทำการเปิด-ปิดอุปกรณ์ไฟฟ้า |
| Current status | เพื่อเรียกดูสถานะคำสั่งล่าสุดของอุปกรณ์ไฟฟ้า |
| Schedule | ซึ่งภายในจะมีเมนูย่อย |
| Add schedule | เพื่อทำการตั้งวัน-เวลา และอุปกรณ์ไฟฟ้าที่จะทำการปิด-เปิด |
| View schedule | เพื่อเรียกดูรายละเอียดของการตั้งวัน-เวลา และอุปกรณ์ไฟฟ้าที่ทำการเลือกไว้ ซึ่งผู้ใช้สามารถทำการเปลี่ยนแปลงหรือลบรายการการตั้งนั้นได้ |
| History | เพื่อเรียกดูรายการคำสั่งที่ได้ทำงานผ่านไปแล้ว |

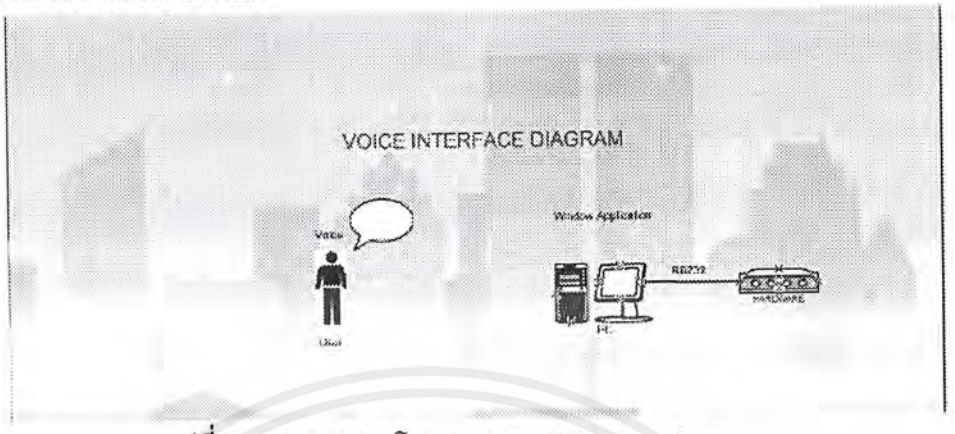
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 5.10 ขั้นตอนการทำงานของ Wap Application

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.ส่วนของ Voice Interface




รูปที่ 5.11 แสดงภาพโดยรวมของ Voice Interface

ในส่วนนี้การออกแบบมีจุดประสงค์เพื่อการใช้งานภายในบ้าน และผู้ใช้สามารถใช้เสียงพูดสั่งการไปยังคอมพิวเตอร์ได้ โดยการทำงานผู้ใช้จะส่งสัญญาณเสียงพูดผ่านไปยังไมโครโฟนที่คอมพิวเตอร์ จะส่งสัญญาณเสียงไปให้แก่ Voice Control Application ที่พัฒนาโดยใช้ ภาษา Visual Basic 6.0 และเรียกใช้ Microsoft Speech API 4.0 และ Microsoft Comm Control เพื่อทำการวิเคราะห์เสียงที่ได้ และทำการส่งข้อมูลควบคุมผ่าน Serial port ไปยัง Hardware เพื่อทำการควบคุมอุปกรณ์ไฟฟ้า และภายใน Voice Control Application ยังสามารถทำการกำหนดชื่ออุปกรณ์ไฟฟ้า (config) ให้กับแต่ละสวิตช์ , ทำการตั้งเวลาการทำงาน (schedule) , ทำการเรียกดูการทำงานที่ผ่านไปแล้ว (history) ซึ่งพัฒนาโดยใช้ Voice Text API และ TTS API และ Active-X Control

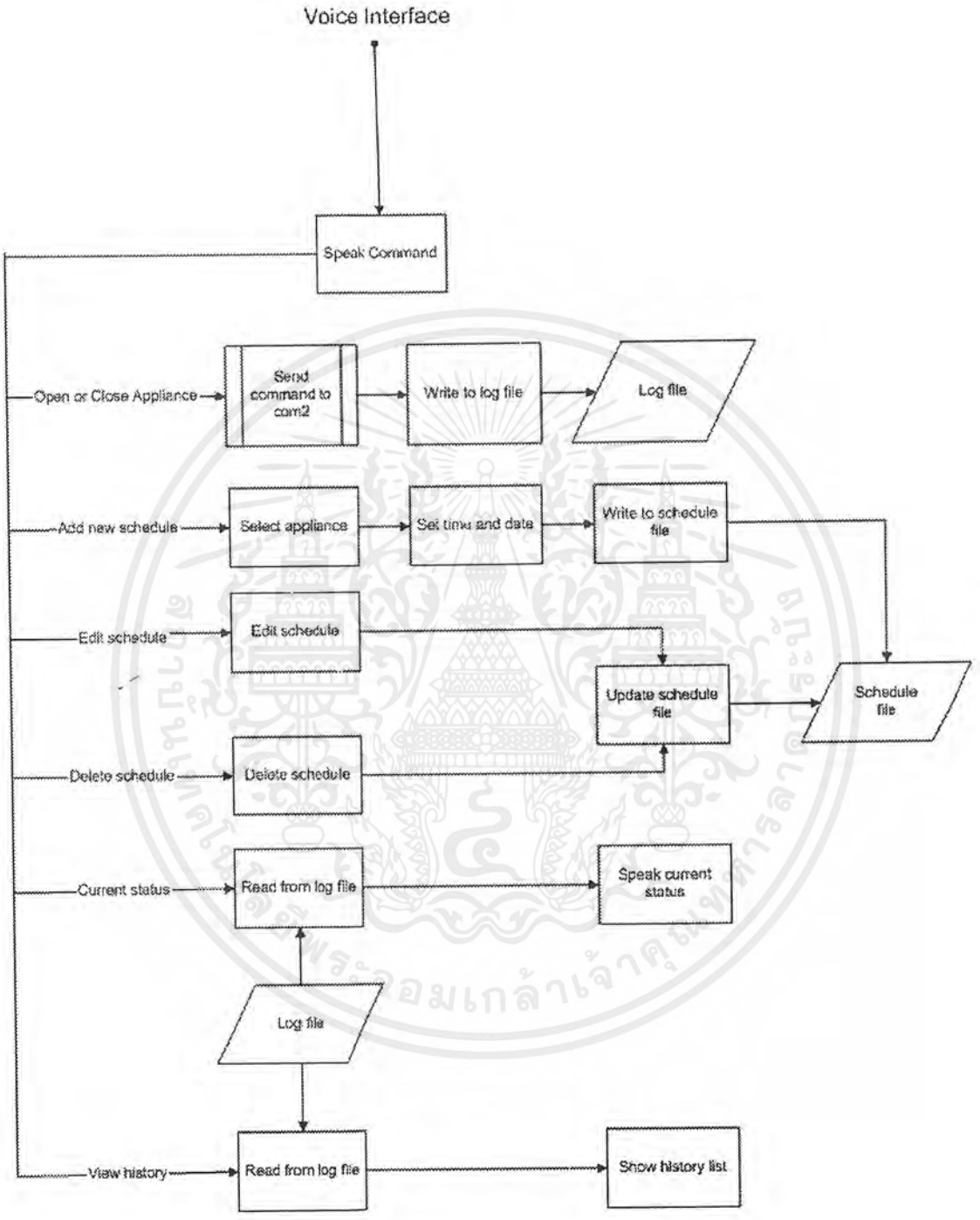
ในส่วนของ Voice Interface นั้นจะสามารถทำงานได้โดยไม่ต้องเชื่อมต่ออินเทอร์เนต โดยผู้ใช้จะทำการพูดคำสั่งผ่านเข้าไปทางไมโครโฟนที่ติดอยู่กับ คอมพิวเตอร์ จากนั้น Voice Control Application ก็จะทำการวิเคราะห์เสียงพูดนั้นว่าตรงกับคำสั่งใด เช่น พูดว่า “on” หรือ “off” แล้วตามด้วยชื่ออุปกรณ์ไฟฟ้าที่กำหนดไว้ จะเป็นการสั่งให้เปิดหรือปิดอุปกรณ์ไฟฟ้านั้นตามลำดับ แอพพลิเคชั่นก็จะทำการติดต่อไปยังพอร์ตอนุกรมและทำการส่งสัญญาณควบคุมไปยัง hardware เพื่อทำการควบคุมอุปกรณ์ไฟฟ้า ภายในแอพพลิเคชั่นยังมีหน้าต่างย่อย ดังนี้

Config เป็นส่วนของการกำหนดชื่อให้กับอุปกรณ์ไฟฟ้าว่าได้ต่อกับหมายเลขสวิตช์ใด

Schedule เป็นส่วนของการตั้งวัน-เวลา การทำงานของอุปกรณ์ไฟฟ้า และสามารถทำการเรียกดูรายละเอียดของการตั้ง พร้อมทั้งสามาถทำการเปลี่ยนแปลงและลบรายการที่ได้ตั้งไว้ได้

History เป็นส่วนของการเรียกดูรายการที่ได้ทำงานผ่านไปแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 5.12 ขั้นตอนการทำงานของ Voice Control Application

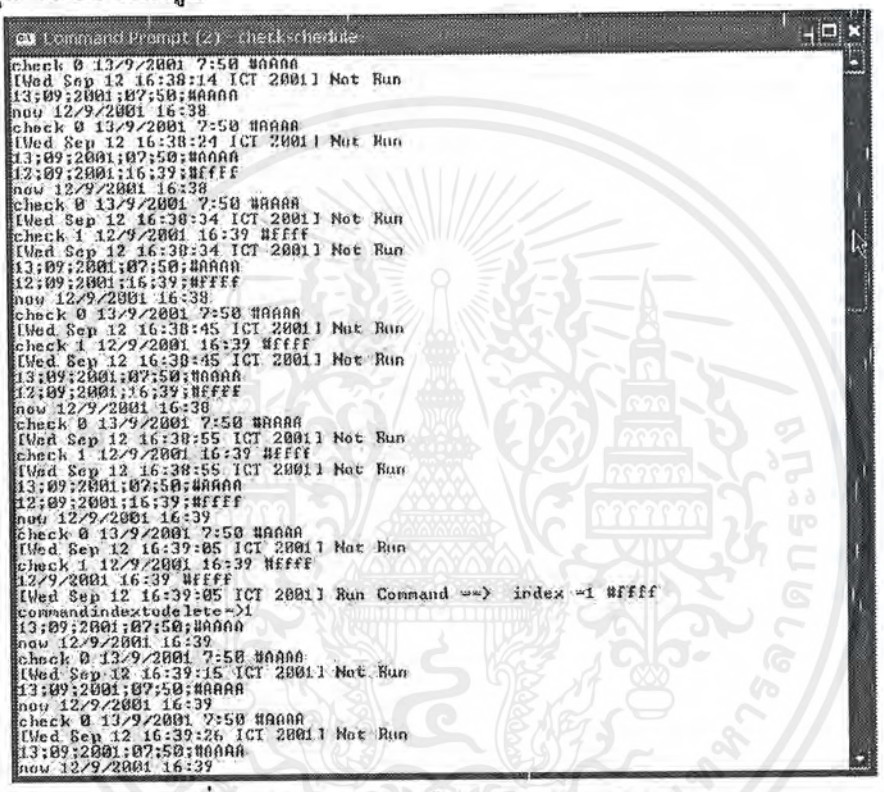
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 6

# การทดลอง

## 6.1 การทดลอง

ที่ server ทำการ run โปรแกรม CheckSchedule เพื่อทำการวนตรวจสอบ การทำรายการที่ได้ตั้งไว้ ทุก 10 วินาที ดังรูป




รูปที่ 6.1 การทำงานของโปรแกรม CheckSchedule

จะเริ่มจากที่ server ทำการเรียกใช้โปรแกรม Modem Listener ทำการรอรับสัญญาณ Ringing tone ที่ผ่าน modem เข้ามายัง COM1 ดังรูป

```
Hanging up...
Sent ATH0<CR> to OutPutStream
                    DATA  AVAILABLE =1 - ATH0

OK
                    DATA  AVAILABLE =2
```

รูปที่ 6.2 ส่วนการรอรับสัญญาณจาก Modem ที่ COM1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อได้รับสัญญาณ Ringing Tone 6 ครั้ง ก็จะทำการติดต่อไปยัง ISP ดังรูป

```
Hanging up...
Sent ATH0<CR> to OutPutStream
Reading response from InputStream -                    DATA  AVAILABLE =1
                    DATA  AVAILABLE =2
                    DATA  AVAILABLE =3
                    DATA  AVAILABLE =4
                    DATA  AVAILABLE =5
                    DATA  AVAILABLE =6
Sent +++<CR> to OutPutStream
Reading response from InputStream - 0 bytes available:
```

รูปที่ 6.3 แสดงการทำงานเมื่อรับสัญญาณ Ringing tone ครบ 6 ครั้ง

เมื่อติดต่อไปยัง ISP ได้สำเร็จ ก็จะทำการนำค่า IP ที่ได้ไปทำการอัพเดต ดังรูป

```
Connecting to CSINTERNET...
Verifying username and password...
Registering your computer on the network...
Successfully connected to CSINTERNET.
Command completed successfully.
Update Ipth set Ip='161.246.48.30'
Result Update=1
```

รูปที่ 6.4 แสดงการติดต่อไปยัง ISP แล้วทำการ Update current IP

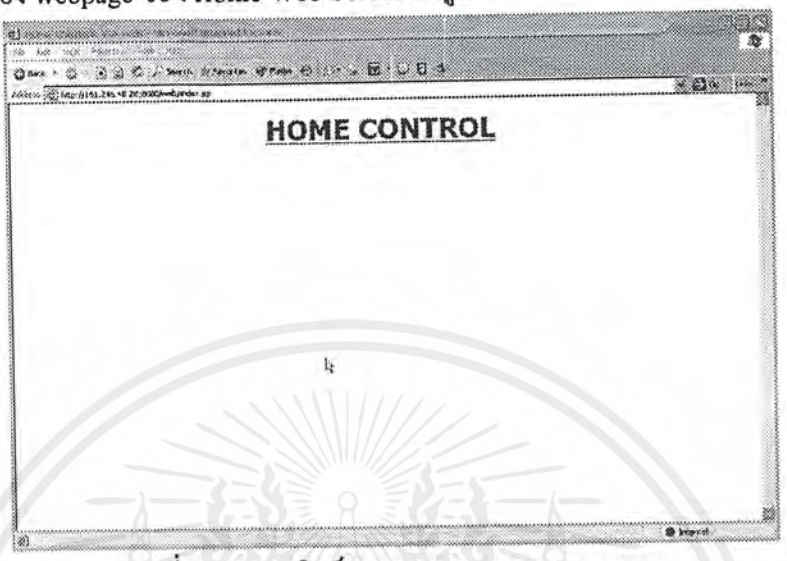
จากนั้นเปิดใช้การทำงาน Home Web Server ดังรูป

```
2001-09-12 05:51:27 - PoolTcpConnector: Starting HttpConnectionHandler on 8080
2001-09-12 05:51:27 - PoolTcpConnector: Starting Ajp12ConnectionHandler on 8007
```

รูปที่ 6.5 แสดงการรอรับ request ของ Home Web Server

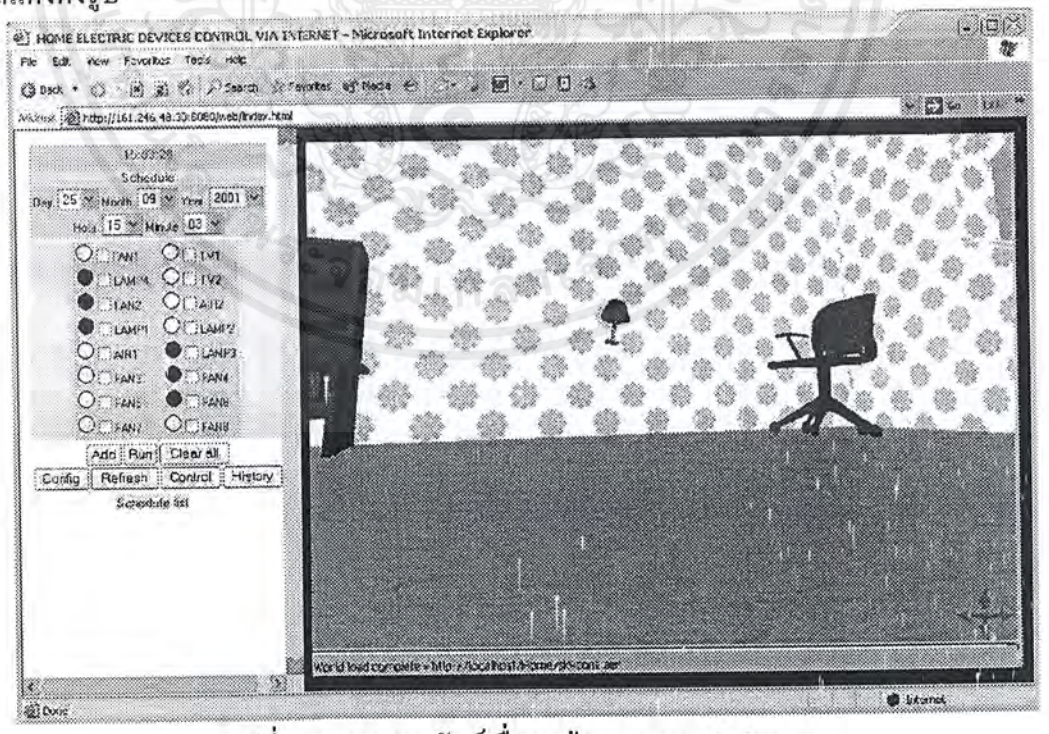
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นทดลองใช้บราวเซอร์เรียกไปยัง **http://161.246.48.28:8080/web** จะปรากฏหน้าเวปที่แสดงลิงค์ ไปยัง webpage ของ Home Web Server ดังรูป




รูปที่ 6.6 แสดงลิงค์บน Webpage ของ Free host

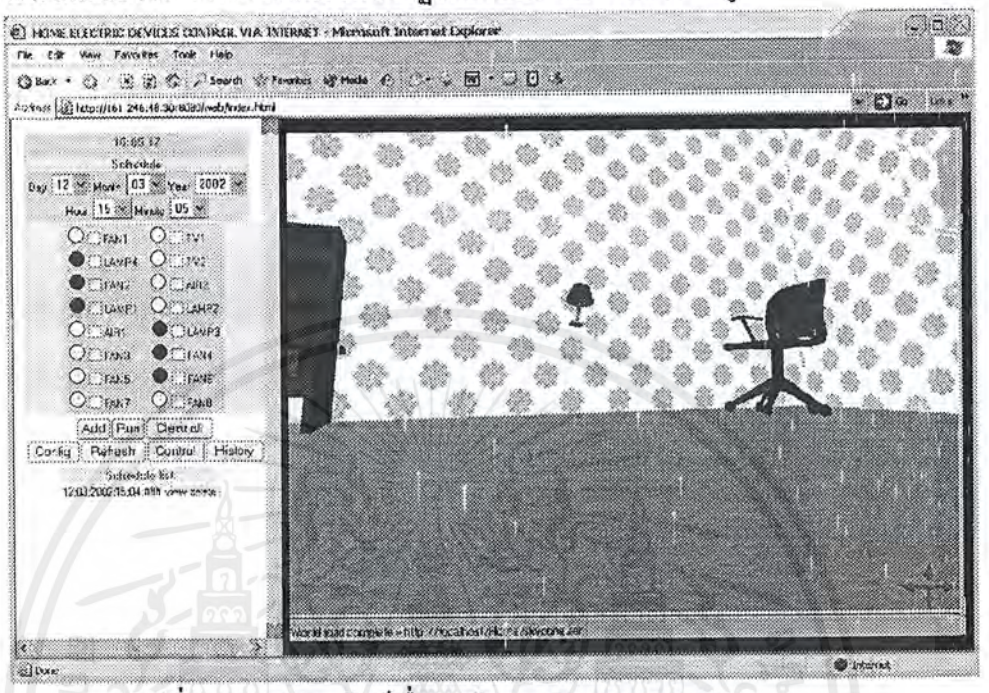
เมื่อทำการคลิกก็จะปรากฏ หน้าเวป แสดงอินเทอร์เฟซ visual 3D ทำการเลือกอุปกรณ์ไฟฟ้าที่ต้องการ จากนั้น กดปุ่ม Run เพื่อส่งคำสั่งให้ทำงาน อุปกรณ์ไฟฟ้าที่เปิดอยู่จะแสดงเป็นรูปวงกลมสีแดงดังรูป




รูปที่ 6.7 แสดงผลลัพธ์เมื่อกดปุ่ม Run บน Webpage

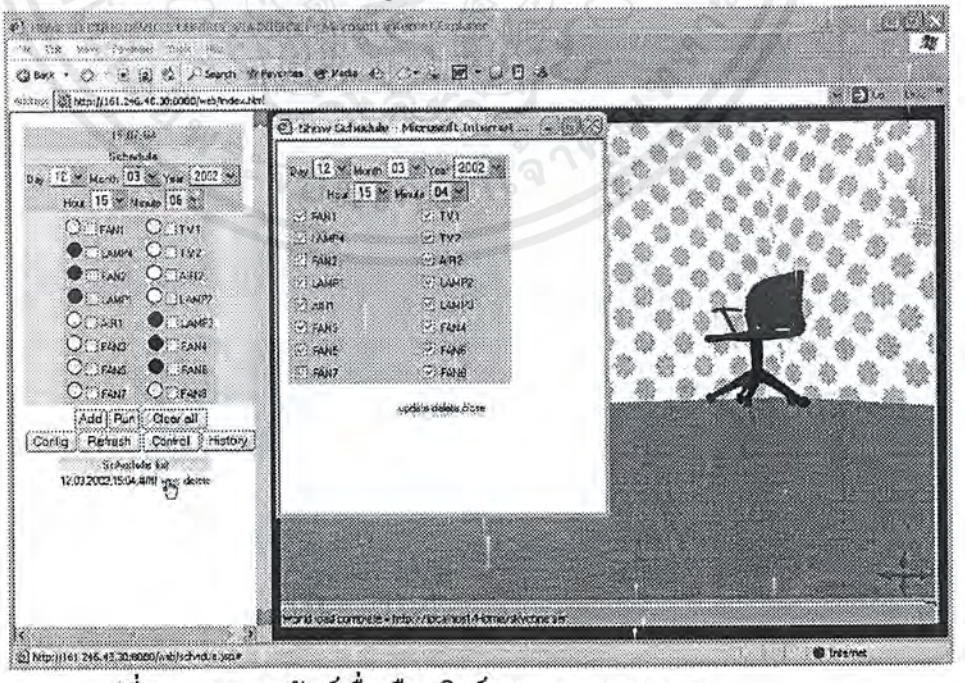
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการตั้งวัน-เวลา และเลือกอุปกรณ์ไฟฟ้า ที่ต้องการ จากนั้น กดปุ่ม Add เพื่อทำการบันทึกรายการที่ได้ตั้งเอาไว้ รายการใหม่จะปรากฏให้เห็นบริเวณด้านล่าง ดังรูป



รูปที่ 6.8 แสดงผลลัพธ์เมื่อกดปุ่ม Add บน Webpage

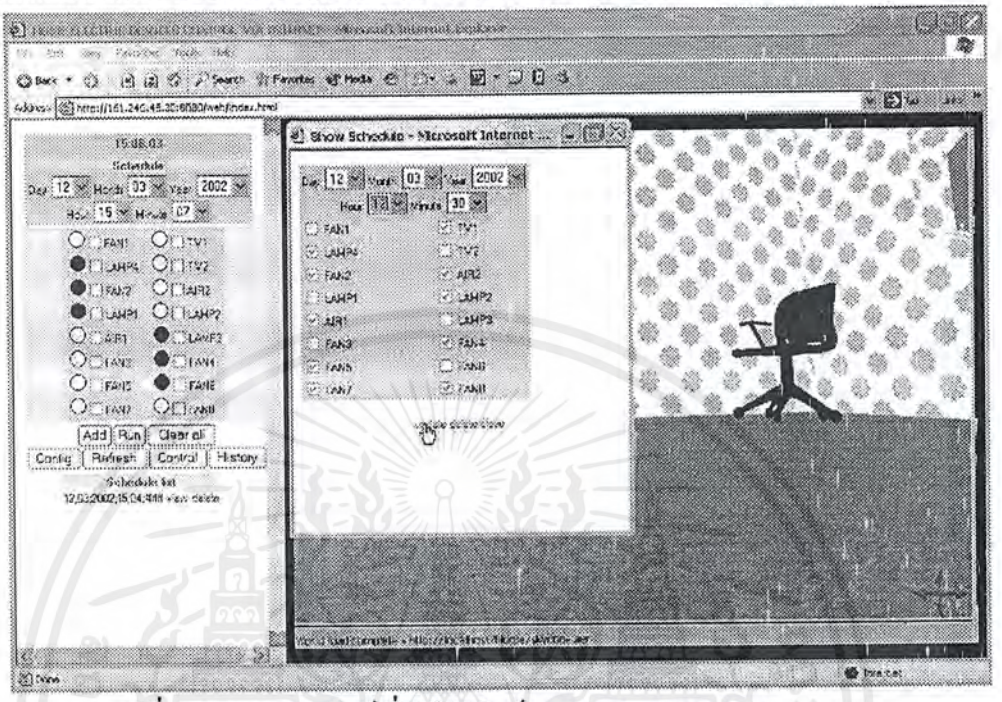
สามารถดูรายละเอียดของแต่ละรายการที่ได้ทำการตั้งเวลาไว้ได้โดยกดที่ view จะแสดงหน้าต่างแสดงรายละเอียดของรายการที่ตั้งไว้ ดังรูป



รูปที่ 6.9 แสดงผลลัพธ์เมื่อเลือกลิงค์ View บน Webpage

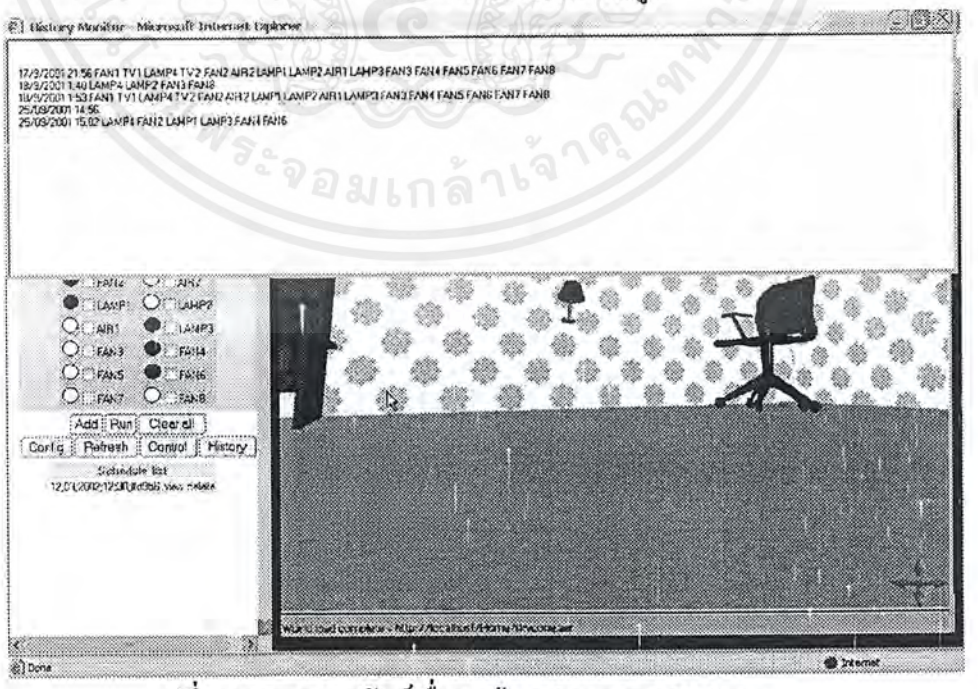
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถทำการเปลี่ยนแปลงรายการของอุปกรณ์ไฟฟ้าที่เลือกไว้ได้ จากนั้นทำการกดที่ update เพื่อทำการอัพเดต หรือ กดที่ close เมื่อไม่ต้องการเปลี่ยนแปลงและปิดหน้าต่าง ดังรูป




รูปที่ 6.10 แสดงผลลัพธ์เมื่อเลือกลิงค์ update บน Webpage

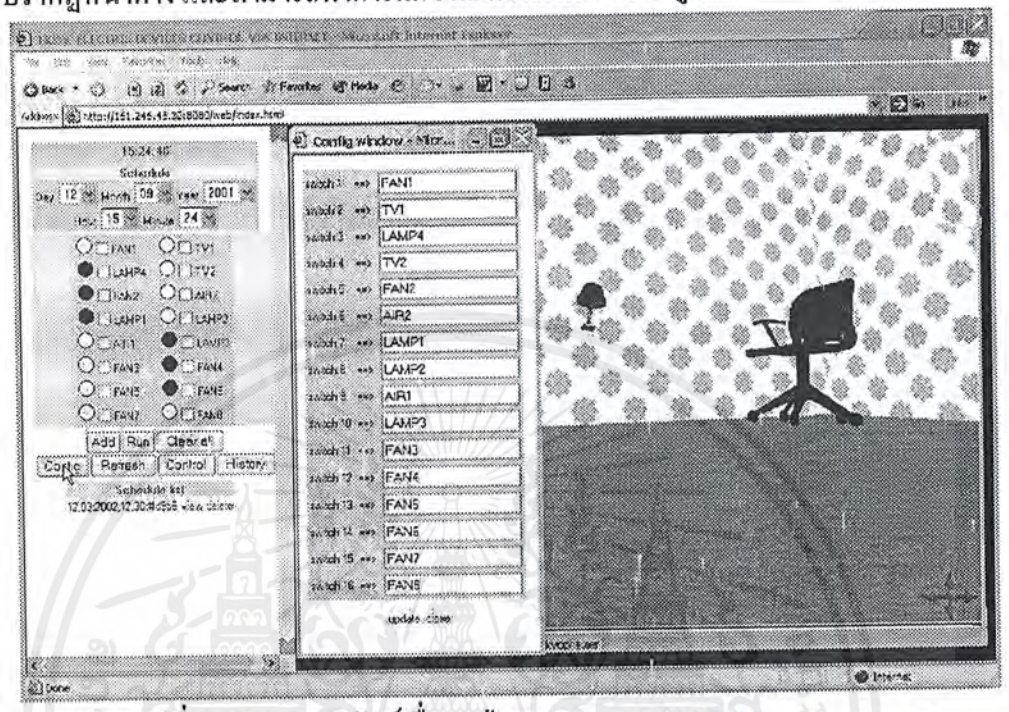
สามารถดูรายการที่ได้ทำงานไปแล้วได้โดยกดที่ปุ่ม History จะแสดงหน้าต่างแสดงรายการที่ได้ทำผ่านไปแล้ว ดังรูป




รูปที่ 6.11 แสดงผลลัพธ์เมื่อกดปุ่ม History บน Webpage

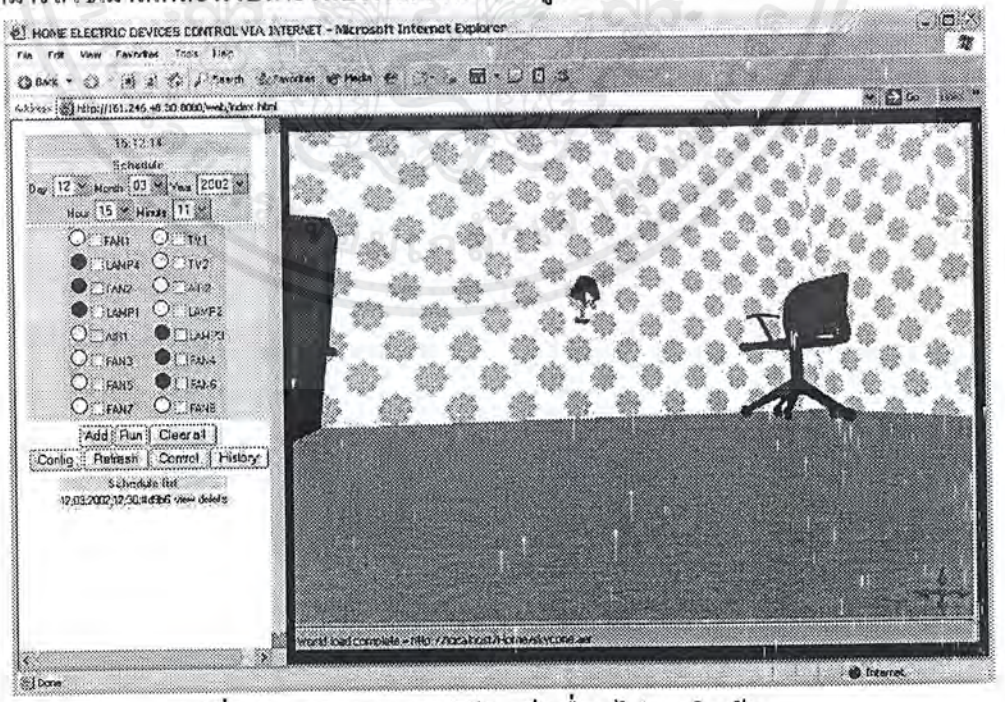
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถทำการกำหนดชื่ออุปกรณ์ไฟฟ้าที่ต่ออยู่กับสวิตช์แต่ละหมายเลขได้ โดยกดปุ่ม Config จะปรากฏหน้าต่าง และสามารถทำการแก้ไขเปลี่ยนแปลงได้ ดังรูป




รูปที่ 6.12 แสดงผลลัพธ์เมื่อกดปุ่ม Config บน Webpage

สามารถใช้เมาส์เคลื่อนไปภายในบริเวณบ้านได้ ดังรูป



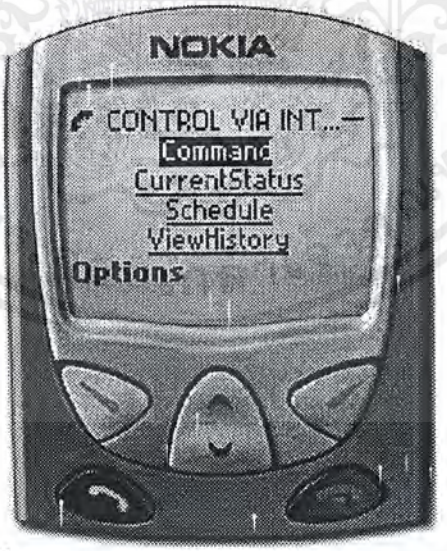

รูปที่ 6.13 แสดงภาพการใช้เมาส์เคลื่อนไปภายในบ้าน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นทดลองใช้ โทรศัพท์เคลื่อนที่ที่สามารถใช้งาน Wap ได้ทำการเรียกไปยัง **http://161.246.48.28:8080/wap** จะปรากฏหน้าแวป แสดงถึงลิงค์ ที่ลิงค์ไปยัง wap page ของ Home Web Server ดังรูป

รูปที่ 6.14 แสดงลิงค์บนหน้า Wappage ของ Free host

และทำการเลือกลิงค์ จากนั้นจะได้ wap page โดยมีรายการเมนูให้เลือกดังรูป




รูปที่ 6.15 แสดงรายการเมนูหลักของ Wappage

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เลือกเมนู command จะแสดงรายการของอุปกรณ์ไฟฟ้า จากนั้นทำการเลือกอุปกรณ์ไฟฟ้าที่ต้องการ ดังรูป

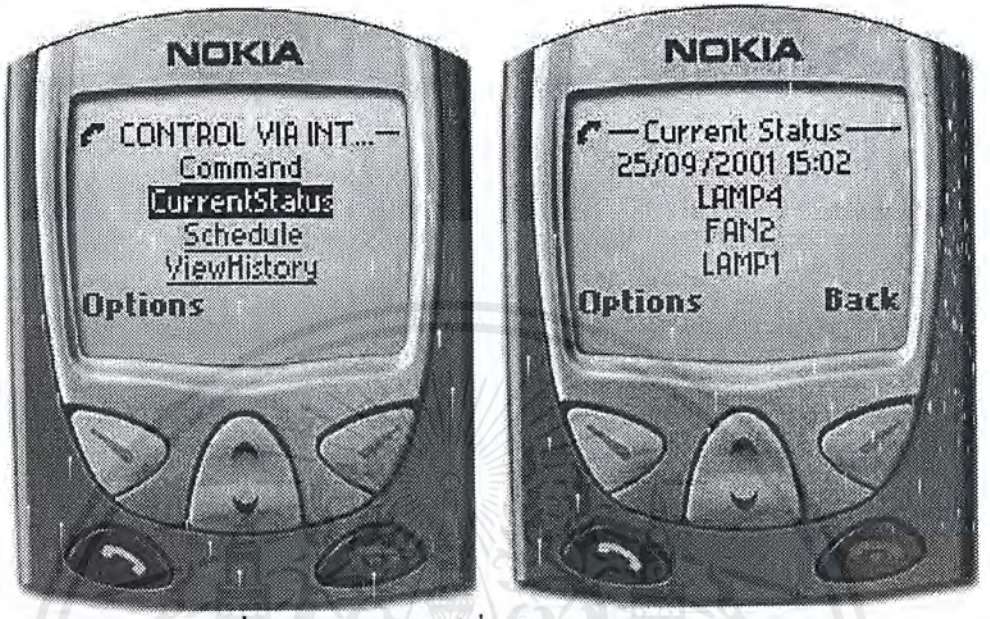
รูปที่ 6.16 แสดงการเลือกรายการอุปกรณ์ไฟฟ้าภายในเมนู Command

จากนั้นเลือก Send เพื่อทำการส่งคำสั่ง และจะแสดงรายละเอียดของคำสั่ง ขึ้นมา ดังรูป

รูปที่ 6.17 แสดงผลลัพธ์เมื่อเลือก Send ภายในเมนู Command

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเลือกรายการ Current status จะแสดงรายการของอุปกรณ์ไฟฟ้าที่เปิดอยู่ ดังรูป



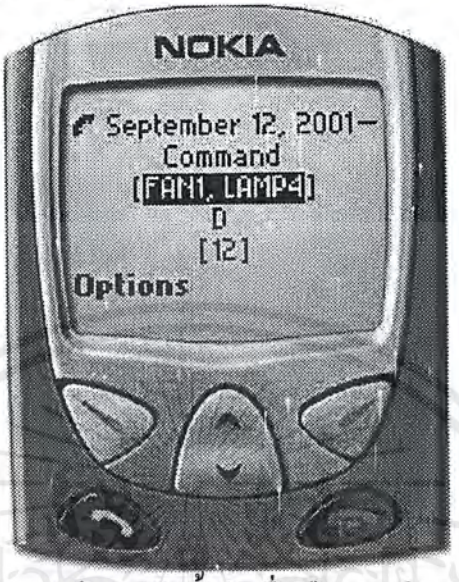

รูปที่ 6.18 แสดงผลลัพธ์เมื่อเลือกเมนูหลัก Current Status

เมื่อเลือกรายการ Schedule จะปรากฏ เมนูรายการย่อย ดังรูป

รูปที่ 6.19 แสดงรายการเมนูย่อยเมื่อเลือกเมนูหลัก Schedule

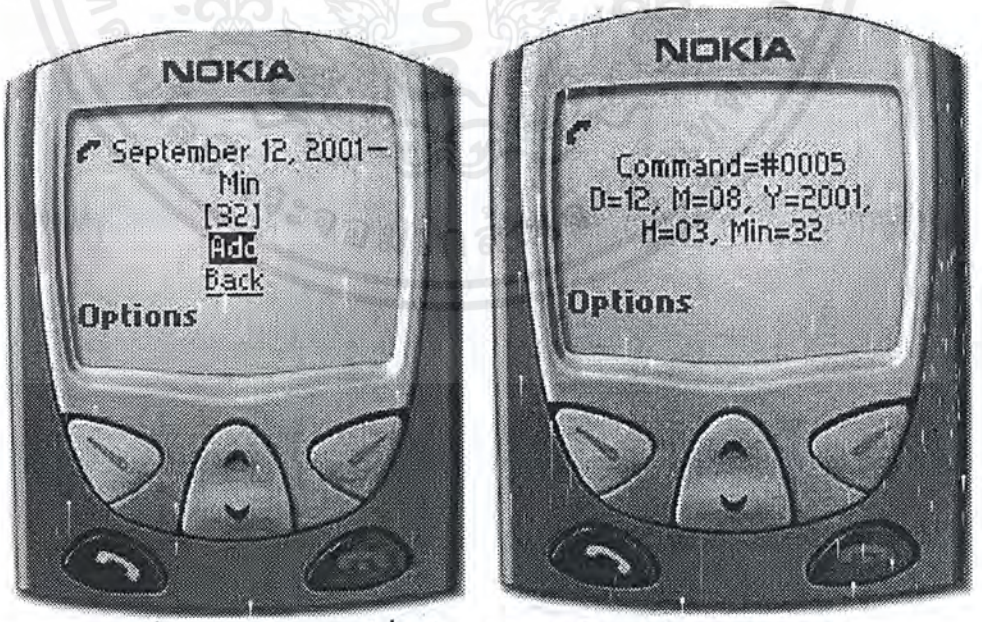
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเลือก Add Schedule จะปรากฏหน้าฟอร์มให้ทำการเลือก อุปกรณ์ไฟฟ้าที่ต้องการเปิด-ปิด และวัน-เวลา ที่ต้องการให้ทำงาน ดังรูป




รูปที่ 6.20 แสดงฟอร์มสำหรับตั้งค่า เมื่อเลือกเมนูย่อย Add Schedule

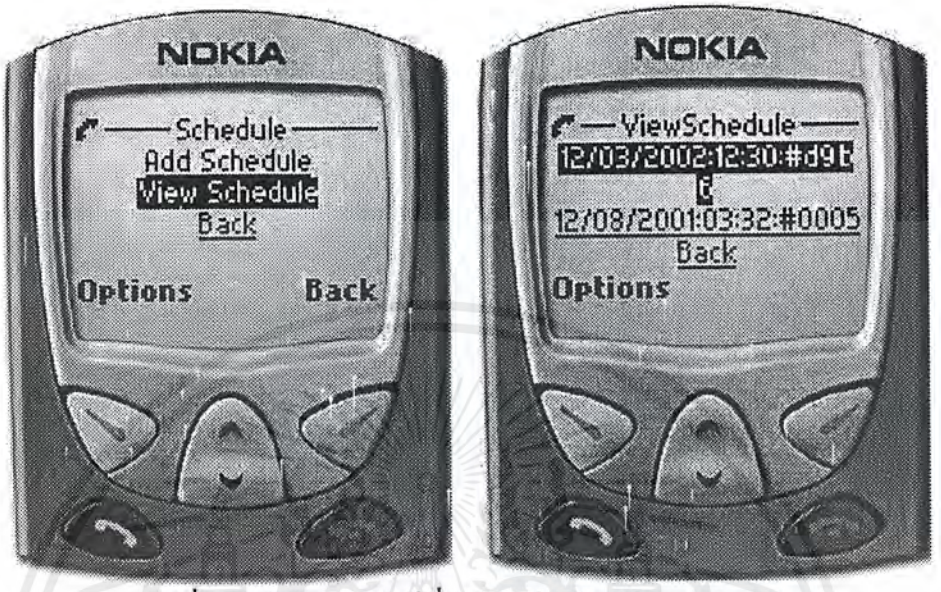
จากนั้นเลือก Add เพื่อทำการบันทึกรายการที่ได้ตั้งไว้ และจะแสดงรายละเอียดของรายการคำสั่งนั้น ดังรูป




รูปที่ 6.21 แสดงผลลัพธ์เมื่อเลือก Add ภายในเมนูย่อย Add Schedule

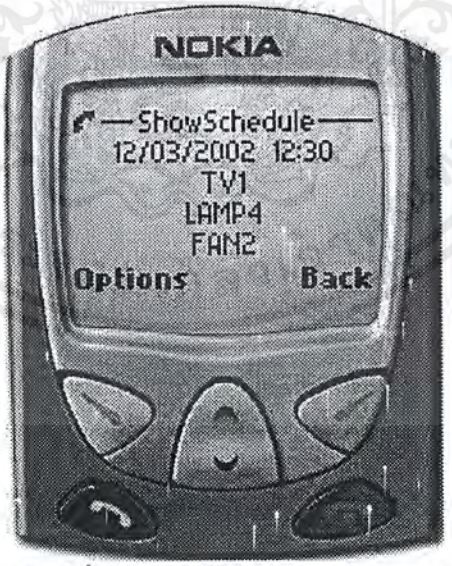
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเลือก View Schedule จะปรากฏหน้าฟอร์มแสดงลิงค์ของรายการที่ได้ทำการตั้งเวลาไว้ ดังรูป




รูปที่ 6.22 แสดงผลลัพธ์เมื่อเลือกเมนูย่อย View Schedule

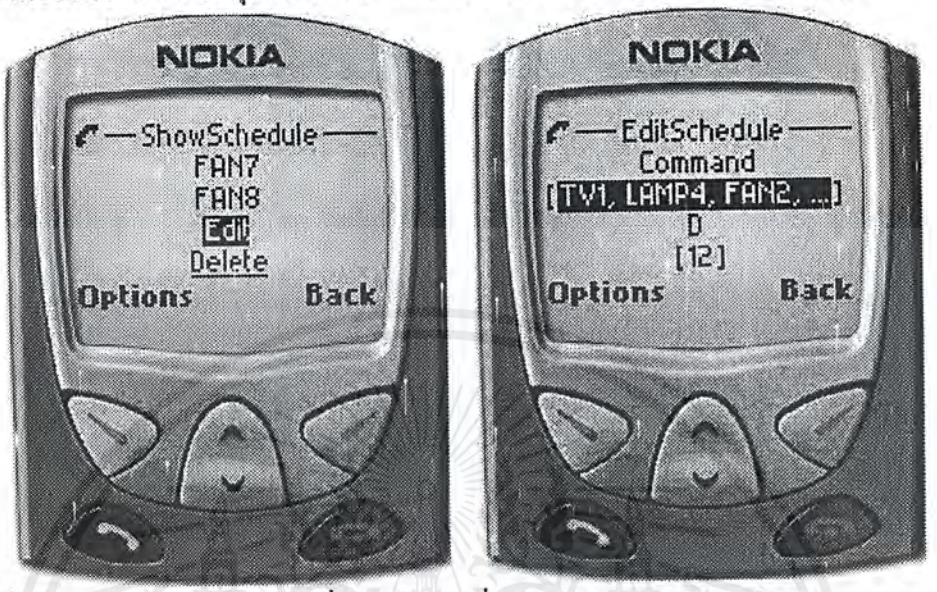
เมื่อทำการเลือกที่ลิงค์รายการ จะปรากฏรายการของอุปกรณ์ไฟฟ้าที่ได้เลือกไว้ ดังรูป




รูปที่ 6.23 แสดงผลลัพธ์เมื่อเลือกลิงค์รายการภายในเมนูย่อย View Schedule

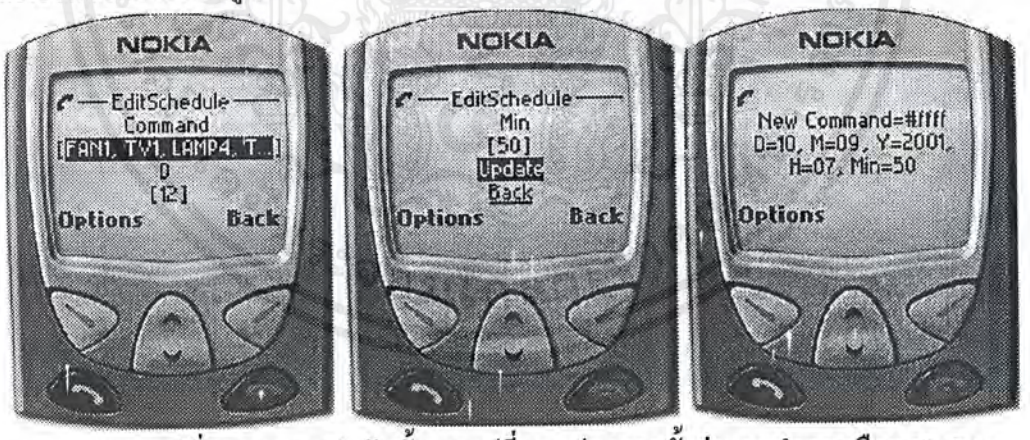
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราสามารถทำการเปลี่ยนแปลงรายการอุปกรณ์ไฟฟ้าที่เลือกไว้ได้โดยเลือกที่ Edit จะปรากฏหน้าฟอร์มของการเลือกอุปกรณ์ไฟฟ้า และวัน-เวลา ให้ทำการเปลี่ยนแปลง ดังรูป




รูปที่ 6.24 แสดงฟอร์มสำหรับเปลี่ยนแปลงค่าเมื่อเลือก Edit ภายในเมนูย่อย View Schedule

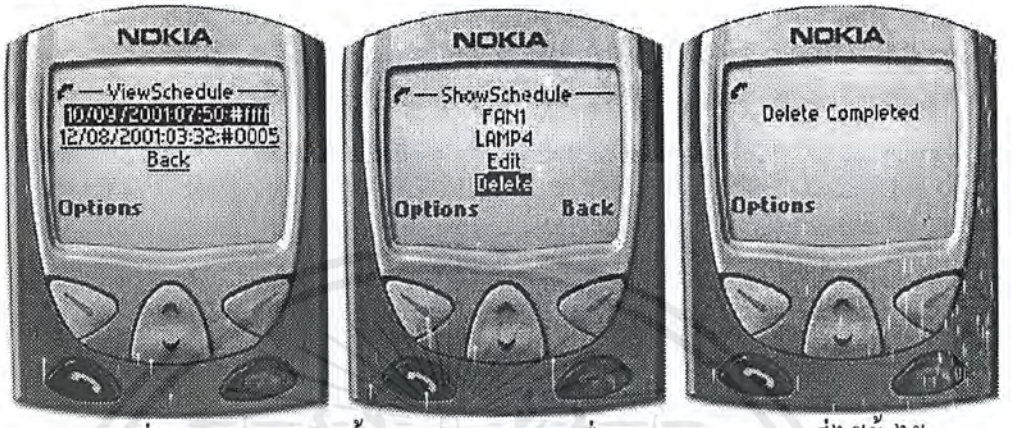
จากนั้นเลือก Update เพื่อทำการอัพเดตค่าของรายการที่เปลี่ยนแปลง และจะแสดงรายละเอียดของคำสั่ง ดังรูป




รูปที่ 6.25 แสดงลำดับขั้นการเปลี่ยนแปลงการตั้งค่าและทำการเลือก Update

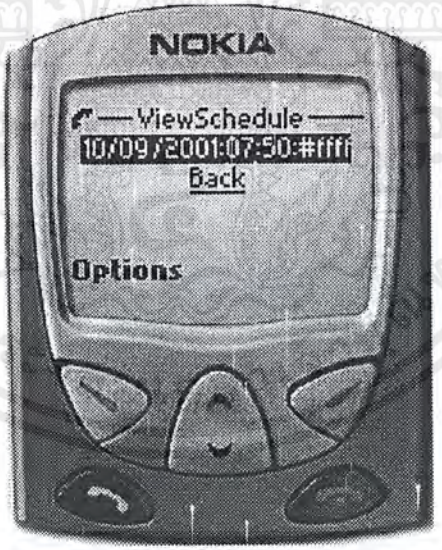
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราสามารถทำการลบรายการที่ได้ทำการตั้งไว้ ได้โดยกดที่ Delete และจะแสดง คำว่า "Delete Completed" ดังรูป




รูปที่ 6.26 แสดงลำดับขั้นการเลือก Delete เพื่อทำการลบรายการที่ได้ตั้งไว้

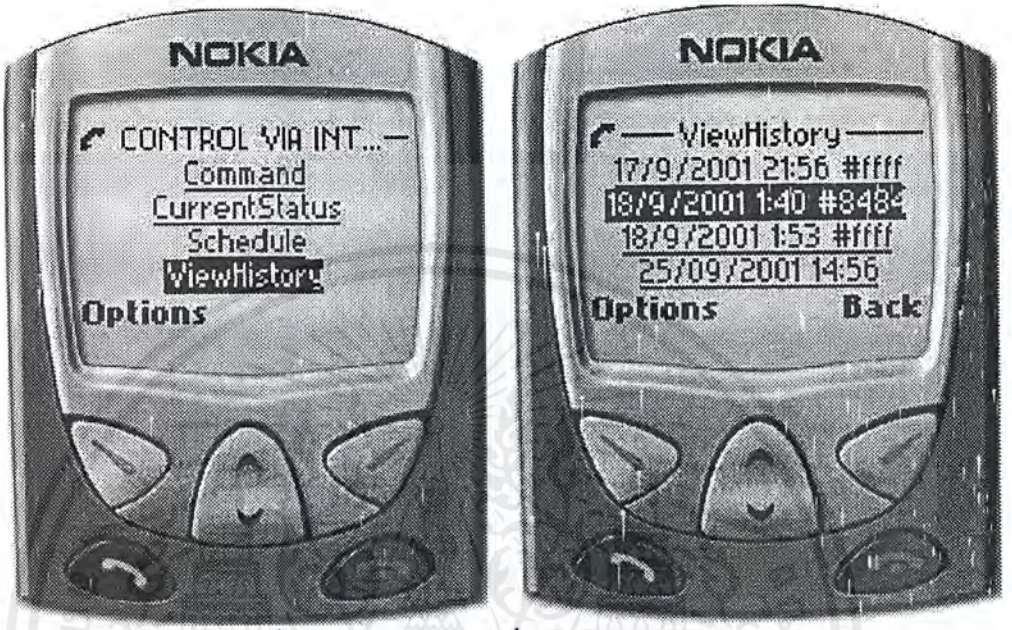
ผลที่ได้จากการลบแสดงดังรูป




รูปที่ 6.27 แสดงผลลัพธ์เมื่อทำการ Delete รายการแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเลือก เมนูรายการ View History จะแสดง ลิงค์รายการของคำสั่งที่ได้ทำผ่านไปแล้ว ดังรูป



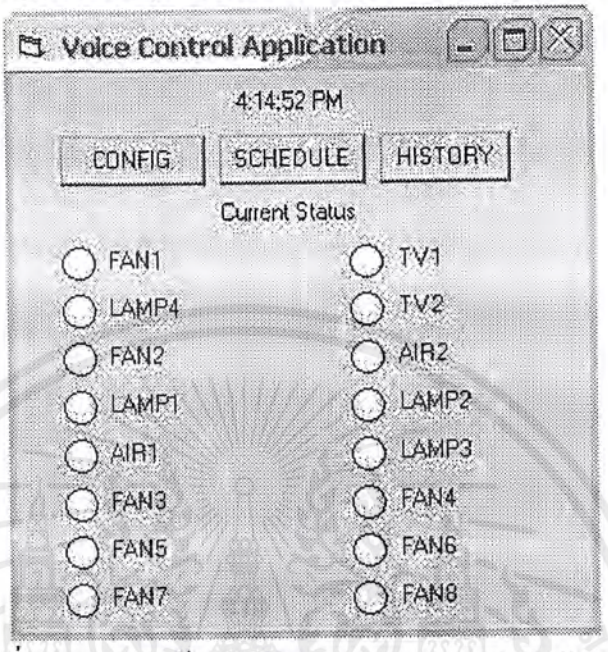

รูปที่ 6.28 แสดงผลลัพธ์เมื่อเลือกเมนูหลัก View History

เราสามารถเข้าไปดูรายละเอียดของแต่ละรายการได้โดยเลือกที่รายการนั้น จะแสดงรายละเอียดของแต่ละรายการคำสั่ง ดังรูป

รูปที่ 6.29 แสดงรายการอุปกรณ์ไฟฟ้าเมื่อเลือกลิงค์ ภายในส่วน View History

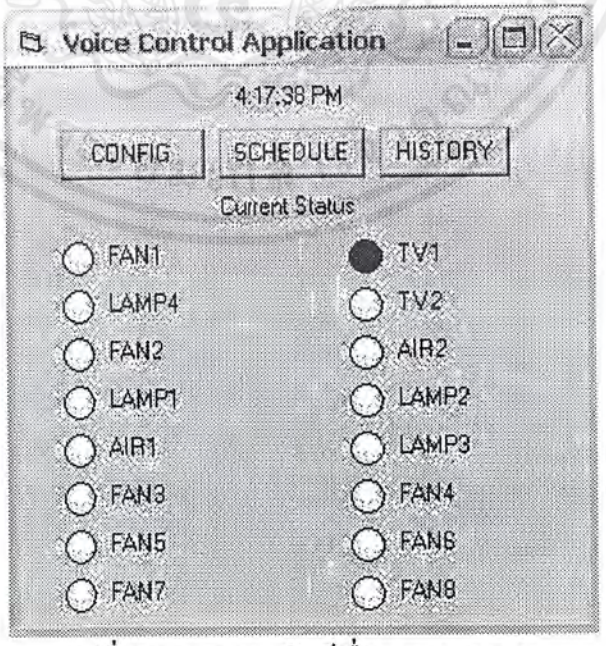
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนสุดท้าย ทำการทดลองใช้เสียงพูดสั่งงานคอมพิวเตอร์ โดยทำการเปิดแอพพลิเคชั่นที่ใช้ในการวิเคราะห์และสั่งงานคอมพิวเตอร์ (Voice Control Application) ดังรูป




รูปที่ 6.30 แสดงหน้าจอหลักของ Voice Control Application

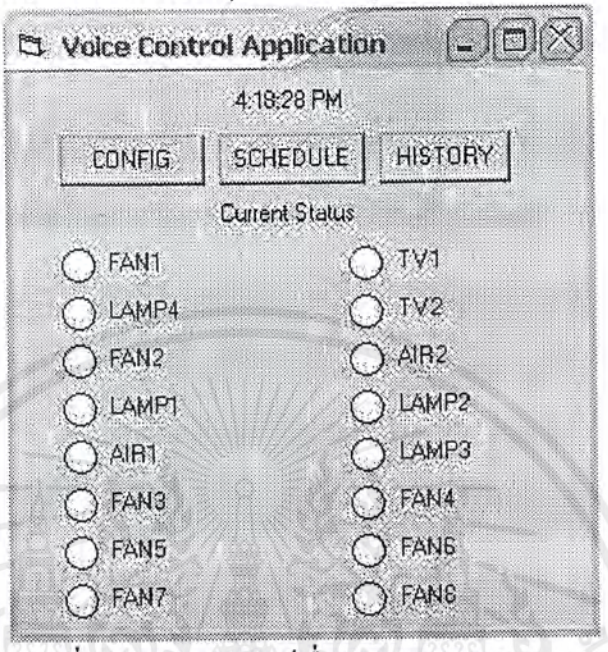
จากนั้นทดลองพูด คำว่า "On TV1" ผ่านไมโครโฟน จะปรากฏว่า วงกลมสีแดงที่ตำแหน่งของ TV1 และโปรแกรมจะทำการส่งคำสั่งไปยังอุปกรณ์ควบคุมสั่งให้หลอด LED ที่ hardware ติดขึ้นมา ดังรูป




รูปที่ 6.31 แสดงผลลัพธ์เมื่อพูด "On TV1"

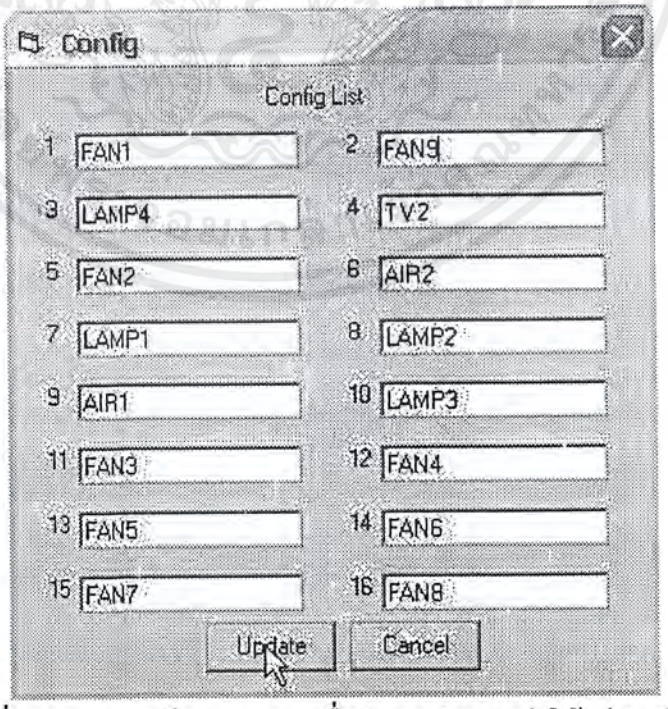
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นลองพูดคำว่า "Off TV1" ผ่านไมโครโฟน จะปรากฏว่าวงกลมสีแดงที่ TV1 จะกลายเป็นสีขาว และโปรแกรมจะส่งคำสั่งไปยังอุปกรณ์ควบคุมเพื่อสั่งให้ led ดับ ดังรูป




รูปที่ 6.32 แสดงผลลัพธ์เมื่อทำการพูด "Off TV1"

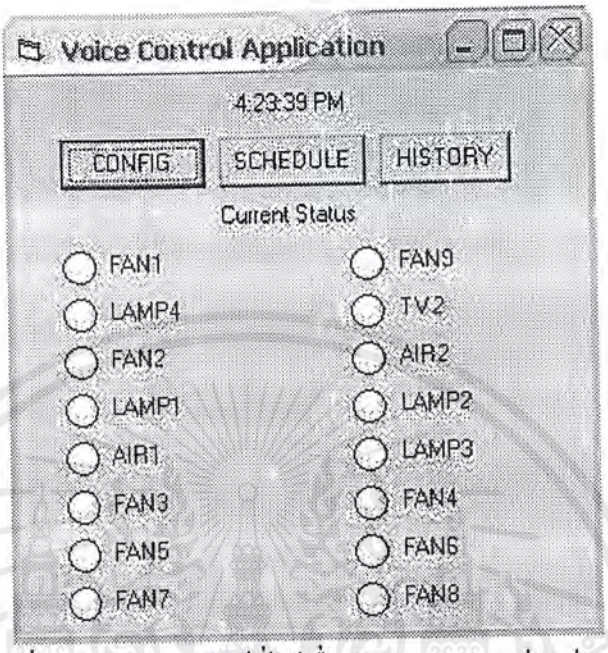
จากนั้นทำการกำหนดชื่ออุปกรณ์ไฟฟ้าให้กับสวิตช์แต่ละตัวได้โดยกดที่ปุ่ม Config จะแสดงหน้าต่าง ดังรูป




รูปที่ 6.33 แสดงหน้าจอ Config เพื่อทำการกำหนดค่าให้แต่ละสวิตช์

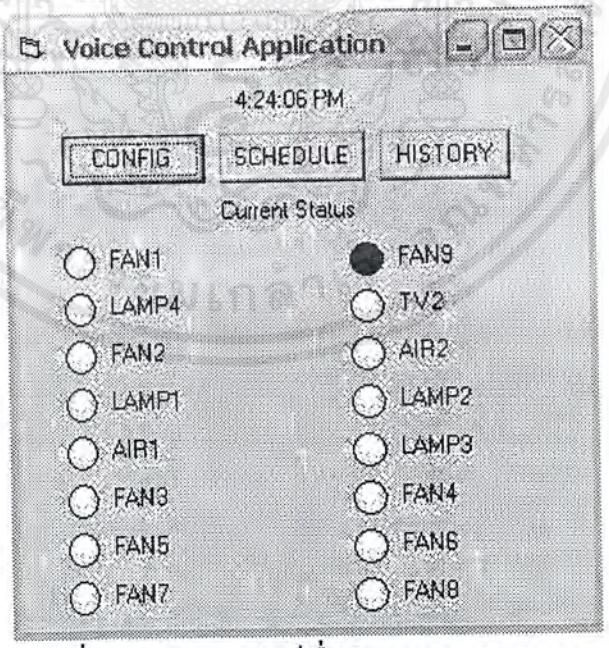
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการเปลี่ยนแปลงค่าจาก TV1 เป็น FAN9 จากนั้นกดปุ่ม Update เพื่อทำการบันทึกค่าที่เปลี่ยนแปลง ดังรูป




รูปที่ 6.34 แสดงผลลัพธ์ที่ได้เมื่อทำการ Update ค่าอุปกรณ์

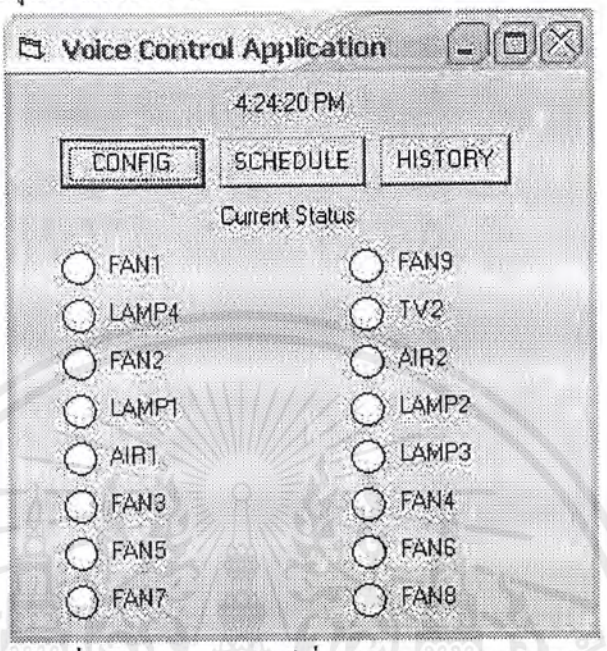
จากนั้นทดลองพูดคำว่า "On fan9" วงกลมสีแดงหน้า FAN9 จะปรากฏขึ้น ดังรูป




รูปที่ 6.35 แสดงผลลัพธ์เมื่อทำการพูด "On fan9"

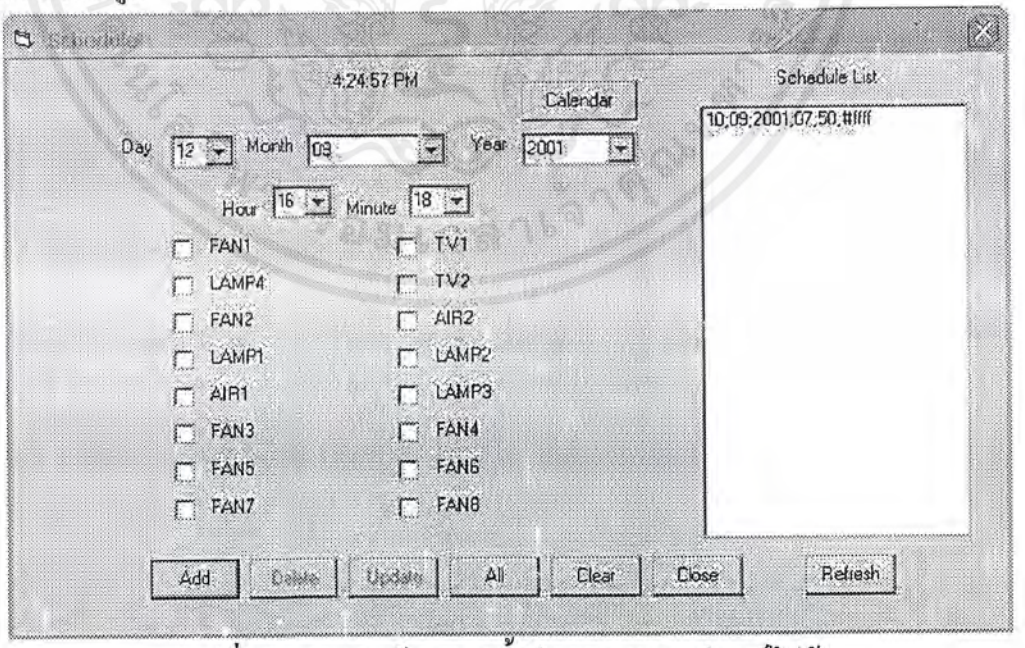
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นทดลองพูดคำว่า "Off fan9" วงกลมสีแดงหน้า FAN9 จะเปลี่ยนเป็นสีขาว ดังรูป แสดงว่าการเปลี่ยนแปลงชื่ออุปกรณ์นั้นได้ผล




รูปที่ 6.36 แสดงผลลัพธ์เมื่อทำการพูด "Off fan9"

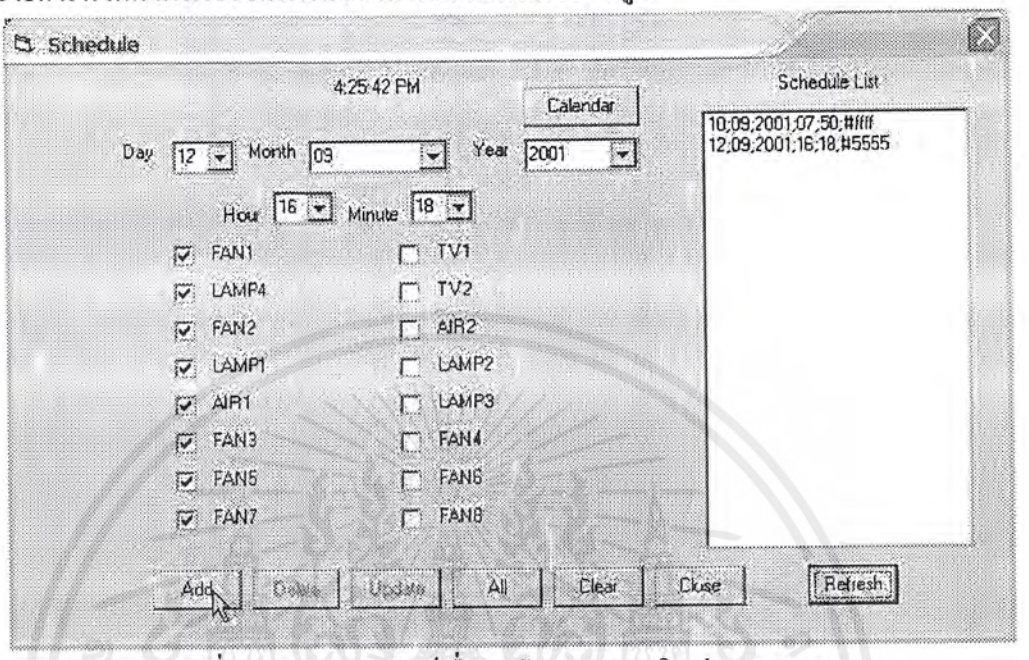
จากนั้นทำการตั้งวัน-เวลา และอุปกรณ์ไฟฟ้า โดยกดที่ปุ่ม Schedule ซึ่งจะแสดงหน้าต่าง Schedule ขึ้นมาดังรูป




รูปที่ 6.37 แสดงหน้าจอการตั้งวัน-เวลา และอุปกรณ์ไฟฟ้า

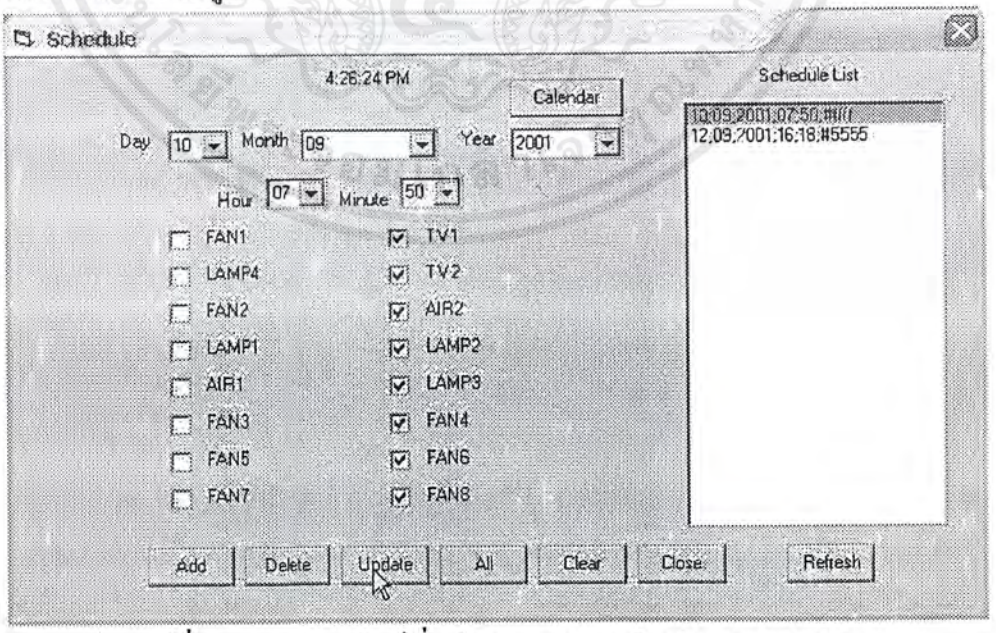
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการเลือกวัน-เวลา และอุปกรณ์ไฟฟ้าที่ต้องการ จากนั้นทำการบันทึกโดยกด ปุ่ม Add จะปรากฏ รายการที่ได้ตั้งเวลาไว้ขึ้นมาในช่องค่าที่ได้กำหนดไว้ ดังรูป




รูปที่ 6.38 แสดงผลลัพธ์เมื่อกดปุ่ม Add ภายในส่วน Schedule

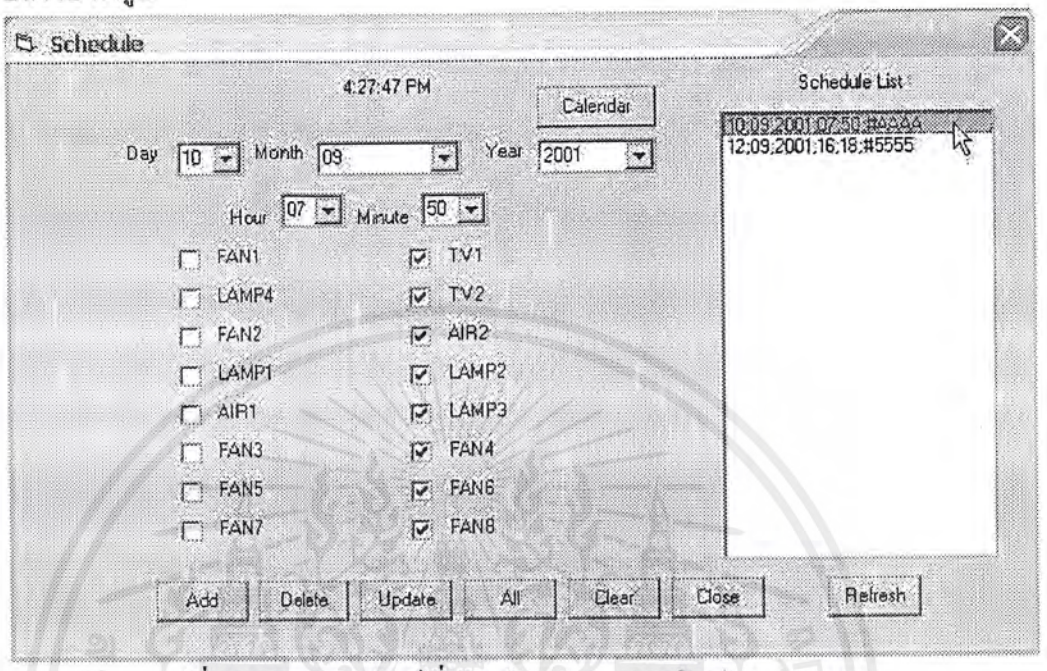
ทำการ เปลี่ยนแปลงค่าที่ได้ทำการตั้งไว้ โดยเลือกที่รายการที่ต้องการเปลี่ยนแปลงภายในช่องค่าที่ได้กำหนดไว้ จะแสดงค่าต่าง ๆ ที่เลือกไว้ จากนั้นทำการเปลี่ยนแปลงวัน-เวลา และ ค่าอุปกรณ์ไฟฟ้าที่ต้องการ ดังรูป




รูปที่ 6.39 แสดงผลลัพธ์เมื่อเลือกรายการภายในส่วน Schedule List

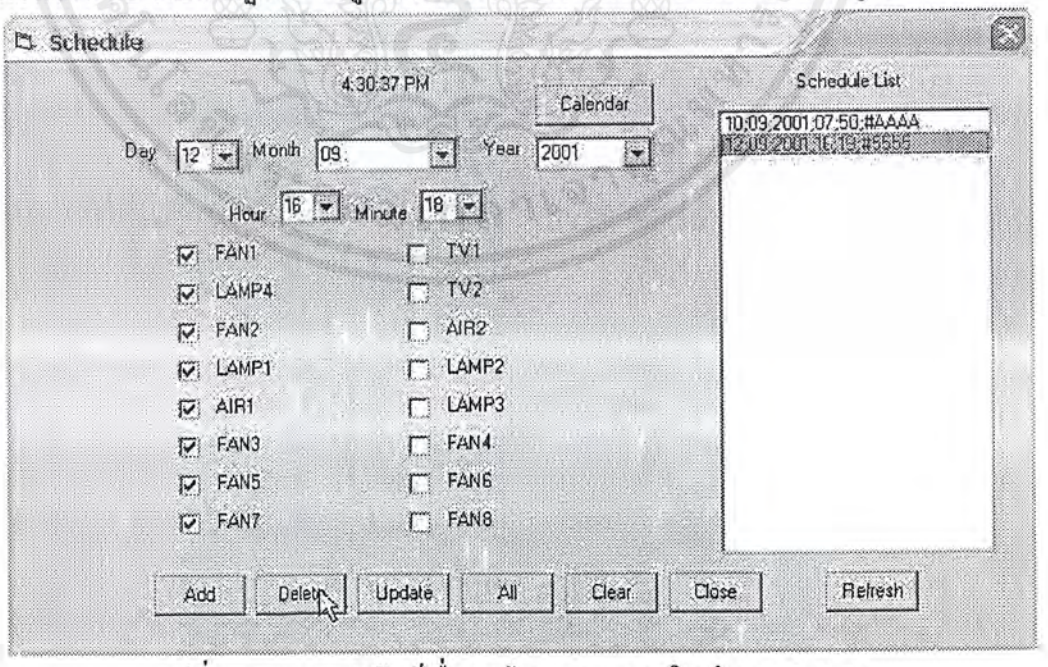
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้น กดปุ่ม Update จะเห็นว่าค่าของรายการภายในช่องค่าที่ได้กำหนดไว้ นั้นมีการเปลี่ยนแปลงไป ดังรูป




รูปที่ 6.40 แสดงผลลัพธ์เมื่อกดปุ่ม Update ภายในส่วน Schedule

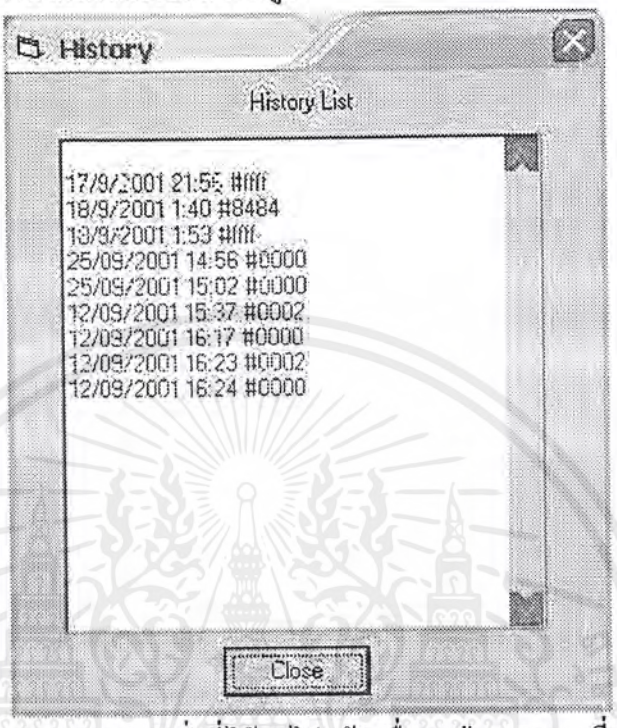
เราสามารถทำการลบรายการที่ได้ตั้งไว้ได้ โดยเลือกรายการที่ต้องการลบ จากนั้นกดปุ่ม Delete เพื่อทำการลบ จะปรากฏว่าค่าที่อยู่ภายในช่องค่าที่ได้กำหนดไว้ นั้นหายไป ดังรูป




รูปที่ 6.41 แสดงผลลัพธ์เมื่อกดปุ่ม Delete ภายในส่วน Schedule

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราสามารถเรียกดูรายการของค่าที่ได้ทำงานไปแล้วได้โดยกดที่ปุ่ม History จะปรากฏหน้าต่างแสดงรายการของค่าที่ได้ทำงานไปแล้ว ดังรูป




รูปที่ 6.42 แสดงรายการคำสั่งที่ได้ทำไปแล้ว เมื่อกดปุ่ม History ที่หน้าจอหลัก

## 6.2 สรุปผลการทดลอง

จากผลการทดลองใช้อินเทอร์เฟซทั้ง 3 แบบนั้นปรากฏว่าสามารถทำงานได้ระดับหนึ่ง และยังต้องมีการปรับปรุงบางอย่าง เช่น ที่ server ควรที่จะมี ระบบของการรับส่ง SMS (Short Message Service) เพื่อทำการติดต่อไปยัง ISP และแจ้งบอกว่าการติดต่อทำได้สำเร็จ ส่วนของ Visual 3D ควรที่จะมีการแสดงให้เห็นถึงสถานะของอุปกรณ์ไฟฟ้าภายในบ้านแต่ละชนิดว่าปิดหรือเปิดอยู่ และอาจมีส่วนของกล้องที่แสดงให้เห็นภาพบรรยากาศจริงภายในบ้านมาแสดง เมื่อมีการสั่งงานก็จะให้เห็นได้ว่าอุปกรณ์ที่สั่งไปนั้นทำงานได้ ส่วนของ Mobile นั้นควรที่จะมี และส่วนของการสั่งงานด้วยเสียงนั้น ควรที่จะสามารถทำการพูดและสั่งให้คอมพิวเตอร์ทำการพิมพ์ข้อความสั้น ๆ กะทัดรัด และส่งเป็นอีเมล์หรือ SMS (Short Message Service) ไปยังบุคคลที่ได้มีการกำหนดไว้ได้ และในส่วนของอุปกรณ์ hardware ควรที่จะเป็นระบบไร้สาย (wireless) เพราะจะทำให้การติดตั้งทำได้สะดวกยิ่งขึ้น เหมาะแก่การเคลื่อนย้ายไปติดตั้งยังสถานที่อื่นได้สะดวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# เอกสารอ้างอิง

1. ดร.อิน มาน หยาง , ทศพล ปราชญสมพงษ์ , "การออกแบบระบบไมโครคอนโทรลเลอร์ ตระกูล 8051" ศูนย์หนังสือมหาวิทยาลัยรังสิต, 2539
2. รศ.สมยศ จุณณะปิยะ , "การประยุกต์ใช้งานไมโครคอนโทรลเลอร์ ตระกูล MCS-51" คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง 2541

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ภาคผนวก ก.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## คำสั่งพื้นฐาน AT

คำสั่ง AT สำหรับโมเด็ม ได้แสดงไว้ตามตารางข้างล่างนี้

## รูปแบบของคำสั่ง AT

บรรทัดคำสั่ง หมายถึง บรรทัดบนจอคอมพิวเตอร์ที่พร้อมรอรับคำสั่งต่างๆ โดยชุดตัวอักษรจากบรรทัดคำสั่งนี้จะถูกส่งจากอุปกรณ์แสดงข้อมูลปลายทาง (Data Terminal Equipment - DTE) ไปยังบริเวณที่จะทำหน้าที่คำนวณเส้นทางออก (Distributed Computing Environment - DCE ) เช่นชุดคำสั่ง & mdash; the modem — โดยแต่ละคำสั่งต้องมีเครื่องหมายนำหน้า (prefix) ส่วนของคำสั่ง และเครื่องหมายลงท้าย แต่ละบรรทัดคำสั่ง (ยกเว้นคำสั่ง A/) จะต้องขึ้นต้นด้วย AT และลงท้ายด้วยเครื่องหมายจบบรรทัด (Carriage return) สามารถใช้ได้ทั้งตัวอักษรตัวใหญ่และเล็กกับคำสั่งเหล่านั้น แต่อย่างไรก็ตามทั้งตัวอักษร A และ T ต้องเป็นอักษรใหญ่หรือเล็กเหมือนกันคือ "AT" หรือ "at" เครื่องหมายลง ท้ายโดยทั่วไป (default) จะถือว่าเป็น Enter ตัวอักขระที่มาก่อนเครื่องหมายนำหน้า (prefix) ของคำสั่ง AT จะไม่ถูกประมวลผล

ตัวอักษรที่อยู่ภายในบรรทัดคำสั่งจะถูกส่งเป็นคำสั่งไปพร้อมกับค่าพารามิเตอร์ต่าง โดยพื้นฐานของคำสั่งจะประกอบด้วย ตัวอักษรแอสกี (ASCII) ที่มีเครื่องหมายนำหน้า (เช่น "&" หรือ "+") และตามด้วยพารามิเตอร์ทศนิยม (decimal parameter) ซึ่งถ้าไม่มีจะถือว่าเป็น 0

## ชุดรหัสพิเศษ (Escape Code Sequence)

เมื่อโมเด็มทำการสร้างการติดต่อ และเข้าสู่โหมดออนไลน์ ก็สามารถหยุด (break) การส่งข้อมูลของโมเด็มเพื่อสร้างชุดคำสั่งบนบรรทัดคำสั่งได้ ซึ่งกระทำได้โดย DTE จะส่งคำสั่ง +++AT ไปยังโมเด็ม เพื่อสร้างชุดรหัสพิเศษ +++ATH ซึ่งมีรูปแบบการส่งที่สอดคล้องกับ TIES (Time Independent Escape Sequence)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่งพื้นฐาน

| คำสั่ง | หน้าที่ |
|---|---|
| AT | Attention — ใช้นำหน้าทุกๆคำสั่ง ยกเว้น A/ |
| A/ | ทำการประมวลผลคำสั่งก่อนหน้า &mdash และไม่จำเป็นต้องมีเครื่องหมายจบบรรทัด <CR> |
| A | มีผลให้โมเด็มทำการยกหูโทรศัพท์ ถ้ามีสัญญาณเรียกเข้ามา โมเด็มจะอาศัยการทำงานของโพรซีเดอร์ (procedure) เล็กๆที่ทำหน้าที่อย่างเงียบๆชั่วขณะ จากนั้นโพรซีเดอร์จะส่งสัญญาณตัวนำไปให้แก่โมเด็มเพื่อทำการส่งเสียงตอบรับ แล้วจึงหยุดการทำงานของตัวเอง (abort) ซึ่งสามารถปรับเปลี่ยนค่าช่วงเวลาในการรอสัญญาณของโมเด็มได้ที่รีจิสเตอร์ S7 |
| B0 | เลือกใช้มาตราฐาน CCITT V.22 (1200 bps) |
| B1 | เลือกใช้มาตราฐาน Belll 212A (1200 bps) |
| B2 | เลือกใช้มาตราฐาน CCITT V23<br>ณ โหมดเริ่มต้น (original mode) จะทำการส่งข้อมูลที่อัตราเร็ว 75 bps และรับข้อมูลที่อัตราเร็ว1200 bps.<br>ณ โหมดตอบรับ (Answer mode) จะทำการส่งข้อมูลที่อัตราเร็ว 1200 bps และรับข้อมูลที่อัตราเร็ว 75 bps. โดยได้มีการใช้คำสั่ง N0 (Disable auto mode) ไว้ก่อนแล้ว |
| D | สั่งให้โมเด็มทำการยกหูโทรศัพท์ และรอสัญญาณโทร (ศึกษาเพิ่มเติม คำสั่ง X เพื่อดูข้อยกเว้น)สามารถทำการปรับเปลี่ยนช่วงเวลาในการรอสัญญาณโทรได้ที่รีจิสเตอร์ S6 |
| Dmn<br><br>L<br><br>W<br><br>T | ATDmn จะทำการหมุนโทรศัพท์หมายเลขที่ n โดย m คือคำสั่งขยายเช่นL, W, ,, ;, @, !, หรือ S<br>ทำการหมุนไปหมายเลขเดิม<br>รอเสียงสัญญาณโทร ถ้าคุณเคยเลือกใช้ค่า X0 หรือ X1 ดังนั้นคุณสามารถใช้ค่า W นี้แทนค่าเดิมได้<br>ทำการโทรแบบสัมผัส (Tone dialing)<br>บนบรรทัดคำสั่งเดียวกันนั้น ไม่สามารถใช้ทั้งคำสั่ง Pulseและ Tone dialing |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | |
|---|---|
| | ร่วมกันได้ |
| P | ทำการหมุนโทรศัพท์ ซึ่งการหมุนแบบนี้นั้นสามารถอนุญาติให้โมเด็มทำงานบนเครือข่ายโทรศัพท์ได้ ในขณะที่การโทรแบบสัมผัสนั้นทำไม่ได้บนบรรทัดคำสั่งเดียวกันนั้น ไม่สามารถใช้ทั้งคำสั่ง PulseและTone dialing ร่วมกันได้ |
| , | ให้ชลอการโทร (dial) ชั่วขณะ สามารถทำการปรับเปลี่ยนช่วงเวลาการชลอนี้ได้ที่รีจีสเตอร์ S8 |
| ; | กลับเข้าสู่โหมดคำสั่งเมื่อ dial เสร็จ โดยจะไม่รอสัญญาณตัวนำ (carrier) หรือวางหูโทรศัพท์ไป |
| @ | หยุดรอเงียบๆ 5 วินาที โดยคำสั่งนี้ใช้กับระบบที่ไม่มีการสื่อสัญญาณการโทร |
| ! | Hook flash. เป็นผลให้โมเด็มทำการวางหู 0.5 วินาที คำสั่งนี้ใช้กับระบบชุมสาย (PBX) และ รูปแบบที่มีการใช้เสียง |
| ^ | ส่งเสียงสัญญาณเรียกเข้าที่ 1300Hz |
| S=(0-9) | ทำการหมุนหมายเลขที่เคยมีการบันทึกไว้ ซึ่งสามารถทำการบันทึกไว้ได้ทั้งหมด 10 หมายเลข โดยจะให้ตำแหน่งแอดเดรสที่ 0-9 และการเก็บบันทึกหมายเลขทำได้โดยใช้คำสั่ง &Z |
| E0 | เป็นตัวระบุมิให้มีการสะท้อนคำสั่งที่เคยป้อนไปแล้วนั้น กลับมายังเครื่องแสดงผลท้องถิ่น (local terminal) อีก ซึ่งคำสั่งนี้จะมีผลเฉพาะเมื่ออยู่ในโหมดคำสั่งเท่านั้นและจะไม่มีผลกระทบต่อความสามารถในการส่งรหัสของโมเด็ม |
| E1 | เป็นตัวระบุให้มีการสะท้อนคำสั่งที่เคยป้อนไปแล้วนั้น กลับมายังเครื่องแสดงผลท้องถิ่น (local terminal) ด้วย |
| H0 | สั่งให้โมเด็มทำการวางหูโทรศัพท์ |
| H1 | สั่งให้โมเด็มทำการยกหูโทรศัพท์ (เพื่อตอบรับ หรือ dial) |
| I0 | ให้ทำการส่งกลับค่ารหัสผลิตภัณฑ์ (product code) |
| I1 | ให้ทำการส่งกลับค่ารหัสความผันแปรของฮาร์ดแวร์ (hardware variation code) |
| I2 | รายงานแสดงค่ารหัสภายใน (internal code) |
| I3 | รายงานแสดงเวอร์ชันของซอฟต์แวร์ |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | |
|---|---|
| I4 | รายงานแสดงรายละเอียดรูปแบบของผลิตภัณฑ์ (product feature listing) |
| I5 | มีอัตราความเร็วการแกว่งสูงสุด (Highest Speed Modulation) |
| L0 | กำหนดความดังเสียงของลำโพงเป็น 0 |
| L1 | กำหนดความดังเสียงของลำโพงให้ต่ำ |
| L2 | กำหนดความดังเสียงของลำโพงให้ต่ำ |
| L3 | กำหนดความดังเสียงของลำโพงให้ต่ำ (ฮาร์ดแวร์เป็นตัวจำกัดให้มีการปรับเสียงเป็นแบบ on/off) |
| M0 | กำหนดให้ลำโพงปิด (off) เสมอ |
| M1 | กำหนดให้ลำโพงเปิดอยู่จนกว่าจะมีการตรวจจับสัญญาณตัวนำ |
| M2 | กำหนดให้ลำโพงเปิด (on) เสมอ |
| M3 | กำหนดให้ลำโพงเปิด (on) เฉพาะเมื่อมีการตอบรับเท่านั้น |
| N0 | ทำการ disable โหมดอัตโนมัติซึ่งเป็นผลบังคับให้โมเด็มทำการติดต่อที่ความเร็วตามที่ระบุไว้ในรีจีสเตอร์ S37. |
| N1 | ทำการ enable โหมดอัตโนมัติ โมเด็มจะทำการติดต่อที่ความเร็วสูงสุด และไม่สนใจต่อคำสั่ง ATBn |
| O0 | กลับสู่โหมดข้อมูล ถ้าคุณกำลังอยู่ในโหมดคำสั่งที่ใช้ TIES คำสั่งนี้จะทำให้คุณกลับเข้าสู่โหมดข้อมูลโดยไม่ต้องวางหูโทรศัพท์ |
| O1 | ให้โมเด็มกลับสู่จุดเริ่มต้นใหม่ (retrain the modem) ถ้าสถานภาพของสายเกิดการเปลี่ยนแปลงอันเนื่องมาจากการติดต่อเดิม คำสั่งนี้จะเป็นผลให้โมเด็มสร้างการติดต่อใหม่ เพื่อพยายามให้ได้อัตราเร็วที่มีประสิทธิภาพมากที่สุดสำหรับสภาวะของสายขณะนั้นๆ |
| Q0 | enable ให้สามารถตอบสนองกับเครื่องแสดงผลปลายทาง (DTE) |
| Q1 | disable ให้ไม่สามารถตอบสนองกับ เครื่องแสดงผลปลายทาง (DTE) ไม่ว่าจะทำการป้อนคำสั่งใดๆก็ตาม (นอกจากคำสั่ง ATZ ที่จะตั้งค่าต่างๆของโมเด็มกลับไปที่ default ) |
| Sn | ทำการระบุให้รีจีสเตอร์ Sn เป็นรีจีสเตอร์ default ผลลัพธ์จากคำสั่งใดๆก็ตาม จะถูกเก็บที่รีจีสเตอร์นี้ |
| Sn=m | กำหนดให้รีจีสเตอร์ n มีค่าเป็น m |
| Sn? | ทำการส่งกลับค่าที่อยู่ในรีจีสเตอร์ n |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | |
|---|---|
| V0 | รหัสผลลัพธ์ (result codes) จะต้องถูกส่งกลับในรูปของตัวเลข. (ดูรายะเอียดตารางแสดงรหัสผลลัพธ์ ) |
| V1 | รหัสผลลัพธ์ (result codes) จะต้องถูกส่งกลับในรูปของคำ. (ดูรายะเอียด ตารางแสดงรหัสผลลัพธ์ ) |
| W0 | ทำการแสดงรายงานเฉพาะอัตราเร็วของ DTE เท่านั้น นั่นหมายถึงว่า หลังจากการติดต่อกันแล้ว จะไม่มีรายงานเกี่ยวกับการแก้ไขข้อบกพร่องต่างๆ หรือโปรโตรคอลที่ใช้ไชการบีบอัดข้อมูล |
| W1 | ทำการแสดงรายงานอัตราเร็วของ DCE, การแก้ไขข้อบกพร่องต่างๆ, โปรโตรคอลที่ใช้ไชการบีบอัดข้อมูล และอัตราเร็วของ DTE |
| W2 | ทำการแสดงรายงานเฉพาะอัตราเร็วของ DCE เท่านั้น |
| X0 | ทำการส่งรหัสผลลัพธ์ OK, CONNECT, RING, NO CARRIER, ERROR และ NO ANSWER. โดย Busy และ Dial Tone Detection จะถูก disabled เอาไว้ |
| X1 | ส่งข้อความของ X0 และ ค่าอัตราเร็วในการติดต่อ |
| X2 | ส่งข้อความของX1และ NO DIALTONE |
| X3 | ส่งข้อความของX2และBUSY และ RING BACKโดยที่ Dial Tone Detection จะถูก disabled เอาไว้ |
| X4 | ส่งการตอบสนองทั้งหมด |
| Y0 | ทำการ disable long space disconnect |
| Y1 | ทำการ enable long space disconnect; with error correction, hang up after sending 1.6 second long space; without error correction, hang up after 4 second long space. |
| Y1Dn | ให้ n = จำนวนดิจิตที่ใช้ในการ dial<br>โมเด็มจะยกหูโทรศัพท์ แล้วทำการส่งข้อมูลดิจิต DTMF ไปอย่างต่อเนื่อง ทำให้ง่ายต่อการปรับ / วัดระดับและการบิด |
| Y113 | ทำการ disablesมิให้มีการ retrain โมเด็มกรณีสัญญาณมีคุณภาพต่ำ (Poor Signal Quality Retrain feature) ซึ่งมีประโยชน์ต่อการตรวจสอบสเปคตรัมการส่งข้อมูลในโมเด็ม |
| Z0 | ใช้ค่าต่างๆของโปรไฟล์ที่ระบุไว้ใน& W0 มาทำการกำหนดค่าต่างให้ระบบใหม่ แล้วเก็บบันทึกไว้ที่โปรไฟล์ 0 |

เอกสารนี้เป็นเอกสารที่สงวนไวสำหรับการใชงานเพื่อการศึกษาเท่านั้น ไมอนุญาตใหนำไปใชประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | |
|---|---|
| Z1 | ใช้ค่าต่างๆของโปรไฟล์ที่ระบุไว้ใน& W1 มาทำการกำหนดค่าต่างให้ระบบใหม่ แล้วเก็บบันทึกไว้ที่โปรไฟล์ 1 |
| =n | กำหนดค่าให้แก่ default รีจีสเตอร์ |
| ? | ทำการรายงานค่าที่เก็บใน default รีจิสเตอร์ S |

**รหัสผลลัพธ์**

| คำสั่งเต็ม | รหัสย่อ | คำอธิบาย |
|---|---|---|
| OK | 0 | โมเด็มทำการประมวลคำสั่ง AT ได้สำเร็จ |
| CONNECT | 1 | ทำการสร้างการติดต่อได้สำเร็จ |
| RING | 2 | โมเด็มตรวจพบสัญญาณเรียกเข้า |
| NO CARRIER | 3 | โมเด็มไม่สามารถตรวจพบตัวนำสัญญาณสื่อทางไกลภายในรีจิสเตอร์S7ได้ |
| ERROR | 4 | โมเด็มตรวจพบข้อผิดพลาดในคำสั่ง AT |
| CONNECT 1200 | 5 | ทำการติดต่อที่ความเร็ว 1200 bps |
| NO DIALTONE | 6 | โมเด็มไม่พบสัญญาณการหมุนภายใน 5 วินาทีหลังการยกหูโทรศัพท์ |
| BUSY | 7 | โมเด็มตรวจพบสัญญาณไม่ว่าง |
| NO ANSWER | 8 | โมเด็มไม่พบสัญญาณเงียบ 5 วินาที เมื่อใช้ตัวขยาย@ ในคำสั่งสำหรับการหมุน |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ภาคผนวก ข.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การติดตั้งโปรแกรม

- ที่ Free host จะติดตั้งส่วนของ web page และ wap page และ database ที่ใช้ในการเก็บค่า IP โดยจะต้องสามารถใช้งาน jsp (java server page) และ Mysql ได้

**Web Application**

การติดตั้งขึ้นอยู่กับ free host ที่ใช้ โดยมีส่วนเอกสาร html และ jsp zip อยู่ที่ไฟล์ webpagefreehost.zip

**Wap Application**

การติดตั้งขึ้นอยู่กับ free host ที่ใช้ โดยมีส่วนเอกสาร html และ jsp zip อยู่ที่ไฟล์ wappagefreehost.zip

**Homecontroldb**

การติดตั้งขึ้นอยู่กับ free host โดยมีข้อมูล database อยู่ที่ไฟล์ homecontroldb.zip โดยส่วนมากจะทำการ unzip ไฟล์ database ไปไว้ภายใน directory /data ของ mysql server

- ที่ Home Server ทำการติดตั้ง

**Tomcat Web Server**

ทำการ unzip ไฟล์ tomcatwebserver.zip ไว้ที่ c:\ จะได้ directory c:\jakarta-tomcat3.2.2 ดังรูป

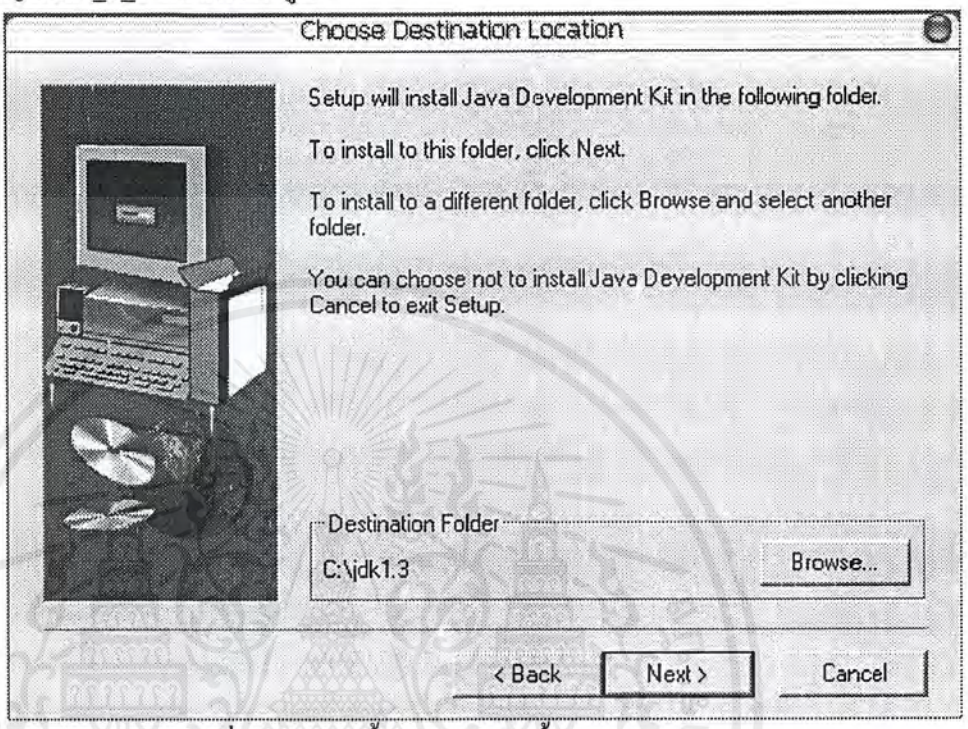
```
03/25/2000  12:21 AM    <DIR>          .
03/25/2000  12:21 AM    <DIR>          ..
05/28/2001  06:51 PM             2,811 LICENSE
03/25/2000  12:21 AM    <DIR>          webapps
03/25/2000  12:23 AM    <DIR>          conf
03/25/2000  12:23 AM    <DIR>          doc
03/25/2000  12:23 AM    <DIR>          lib
03/25/2000  12:24 AM    <DIR>          bin
03/25/2000  12:24 AM    <DIR>          logs
03/25/2000  12:24 AM    <DIR>          src
03/25/2000  12:24 AM    <DIR>          work
               1 File(s)          2,811 bytes
              10 Dir(s)   9,207,693,312 bytes free

C:\jakarta-tomcat-3.2.2>
```

รูปที่ ข.1 แสดงรายชื่อไฟล์ภายใน directory c:\jakarta-tomcat-3.2.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Java 2 sdk**

ทำการ execute ไฟล์ แล้วทำตามขั้นตอนติดตั้งจนเสร็จ โดยทำการติดตั้งไว้ใน directory c:\j2sdk1_3_0-win.exc ดังรูป




รูปที่ ข.2 แสดงขั้นตอนการติดตั้ง Java 2 sdk

**Java Comm API**

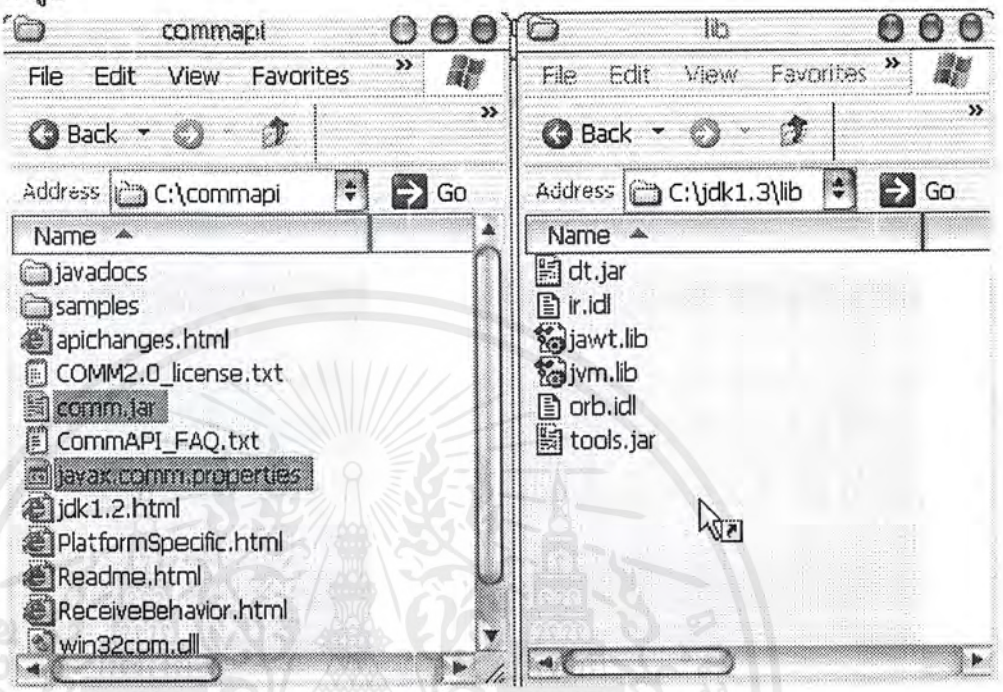
ทำการ unzip ไฟล์ javacomm20-win32.zip ไว้ที่ c:\commapi ดังรูป

```
 Directory of C:\commapi

03/25/2000  12:52 AM    <DIR>          .
03/25/2000  12:52 AM    <DIR>          ..
11/15/1998  04:00 PM            27,648 win32com.dll
11/15/1998  04:00 PM            28,043 comm.jar
11/15/1998  03:59 PM             3,913 Readme.html
11/15/1998  03:59 PM             1,821 ReceiveBehavior.html
11/15/1998  03:59 PM             2,182 jdk1.2.html
11/15/1998  03:59 PM             3,715 PlatformSpecific.html
11/15/1998  03:59 PM             8,141 COMM2.0_license.txt
11/15/1998  03:59 PM             3,335 apichanges.html
11/15/1998  03:59 PM             5,374 CommAPI_FAQ.txt
11/15/1998  03:59 PM               467 javax.comm.properties
03/25/2000  12:52 AM    <DIR>          javadocs
03/25/2000  12:52 AM    <DIR>          samples
              10 File(s)         84,639 bytes
               4 Dir(s)   8,912,961,536 bytes free

C:\commapi>
```

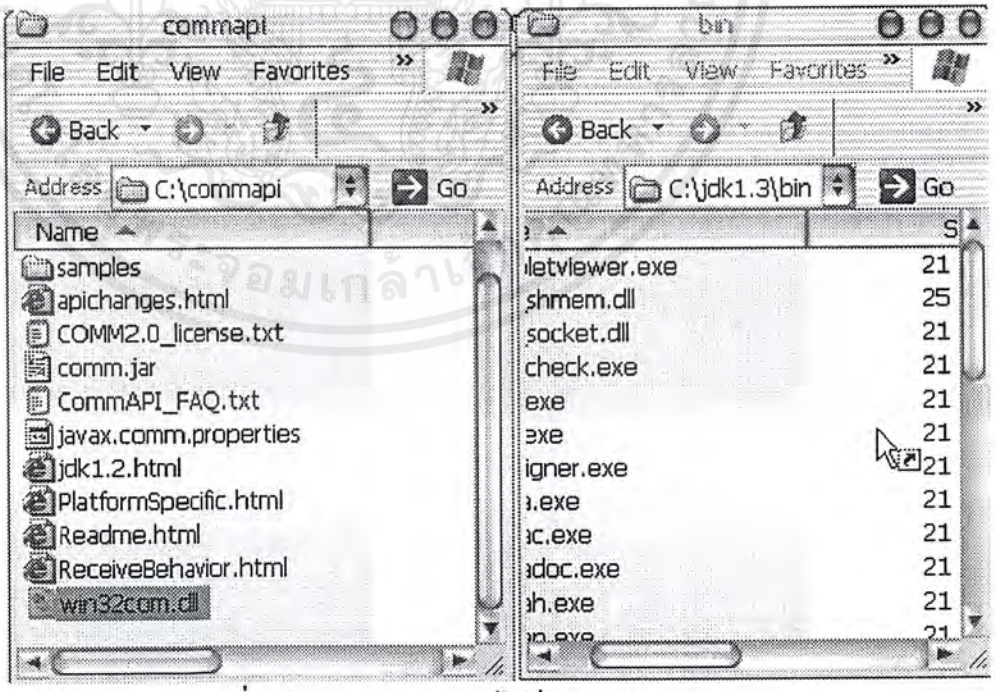
รูปที่ ข.3 แสดงรายชื่อไฟล์ภายใน directory c:\commapi

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นทำการ copy ไฟล์ comm.jar javax.comm.properties ไปไว้ที่ directory c:\jdk1.3\lib จากนั้น copy ไฟล์ win32com.dll ไปไว้ที่ directory c:\jdk1.3\bin ดังรูป




รูปที่ ข.4 แสดงการ copy ไฟล์ comm..jar และ javax.comm.properties

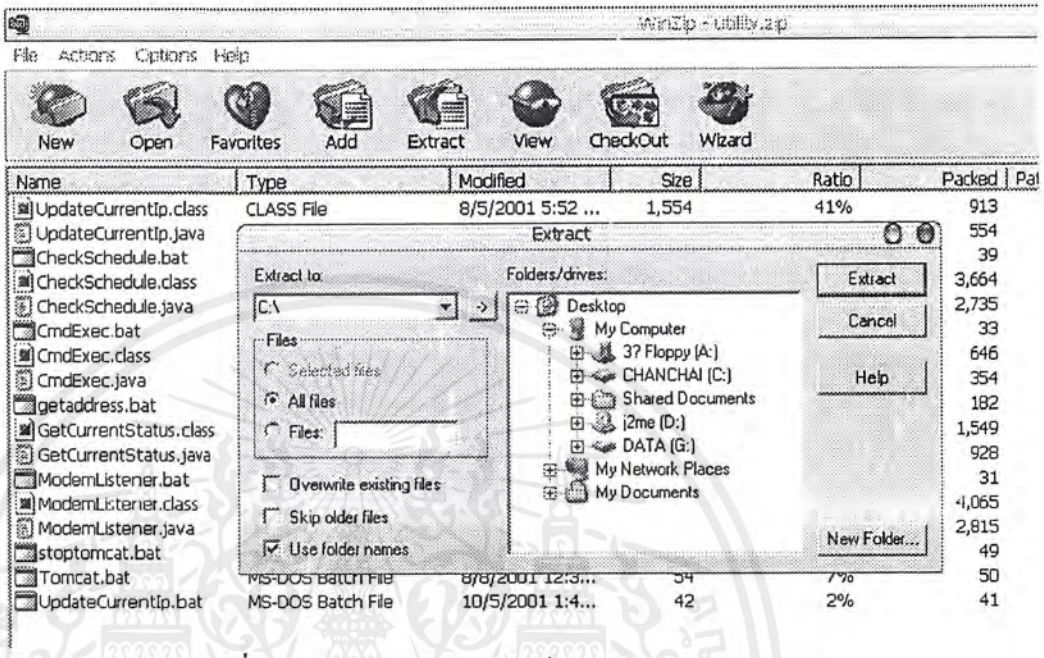



รูปที่ ข.5 แสดงการ copy ไฟล์ win32com.dll

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Utility.zip**

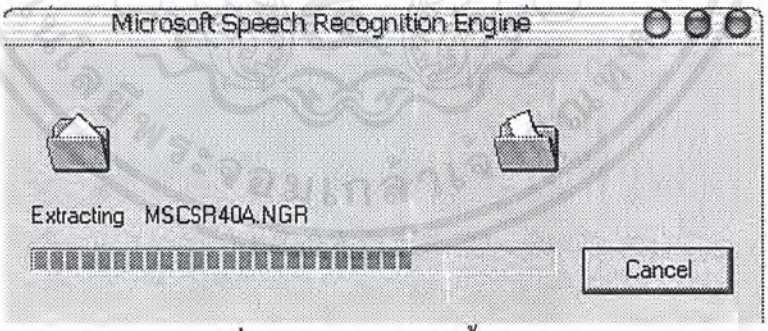
ทำการ unzip ไฟล์ utility.zip ไว้ที่ c:\ ดังรูป




รูปที่ ข.6 แสดงการ unzip ไฟล์ Utility.zip

**Microsoft Speech API**

ทำการ execute ไฟล์ SAPI4SUITE.EXE แล้วติดตั้งจนเสร็จ ดังรูป

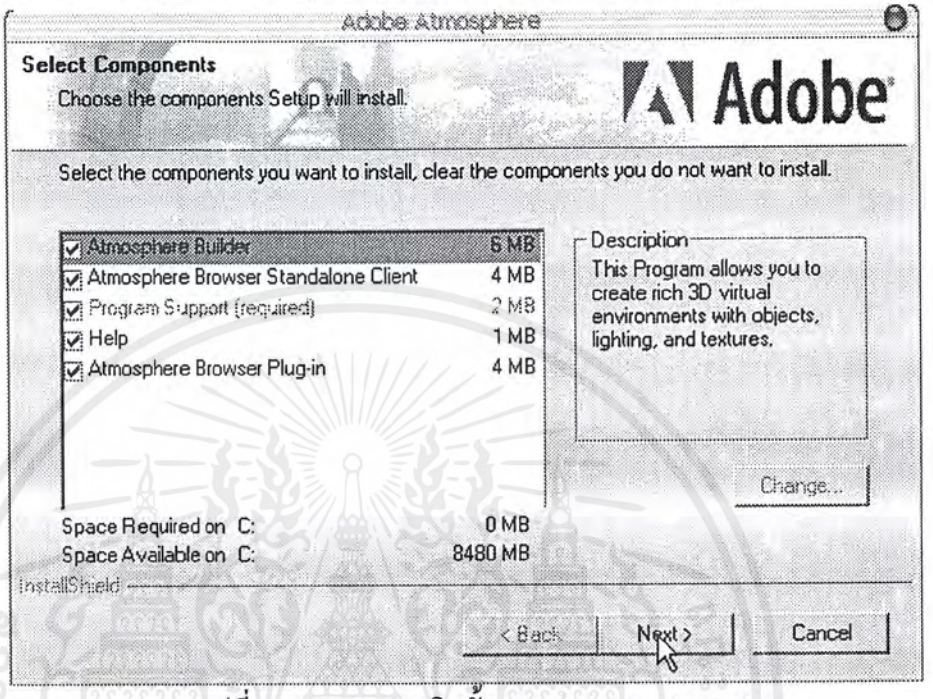



รูปที่ ข.7 แสดงการติดตั้ง Microsoft Speech API

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Atmosphere Plug-in**

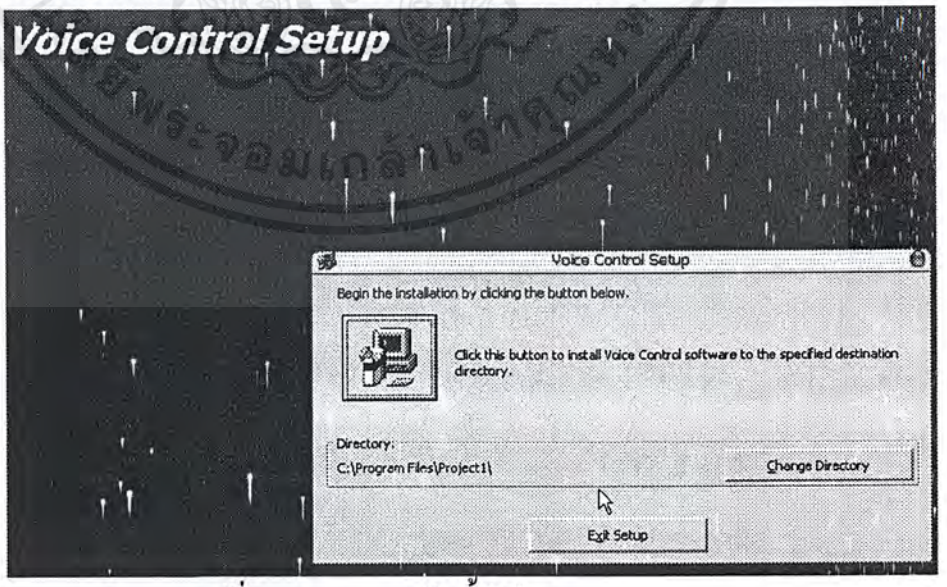
ทำการ execute ไฟล์ Atmosphere_Full.exe แล้วติดตั้งจนเสร็จ ดังรูป




รูปที่ ข.8 แสดงการติดตั้ง Adobe Atmosphere Plug-in

**Voice Control Program**

ทำการ execute ไฟล์ setup.exe แล้วติดตั้งจนเสร็จ ดังรูป



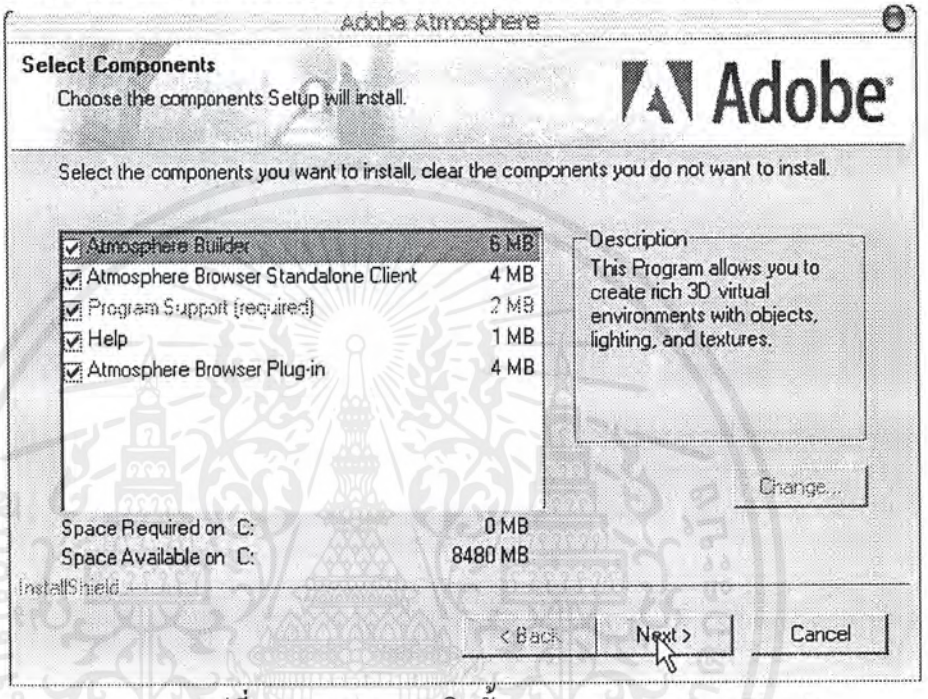

รูปที่ ข.9 แสดงการติดตั้งโปรแกรม Voice Control

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ที่ Client ทำการติดตั้ง

**Atmosphere Plug-in**

ทำการ execute ไฟล์ Atmosphere_Full.exe แล้วติดตั้งจนเสร็จ ดังรูป

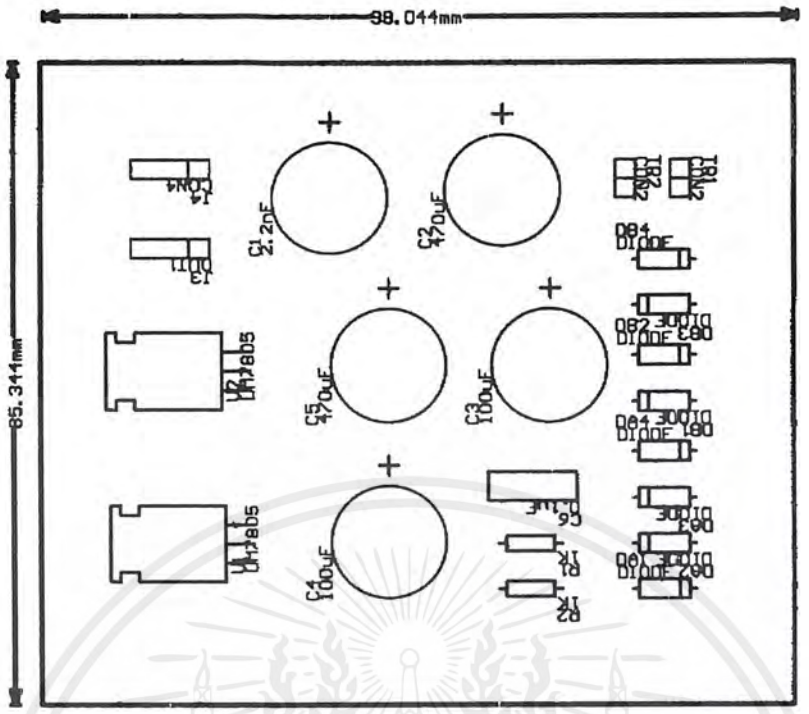



รูปที่ ข.10 แสดงการติดตั้ง Adobe Atmosphere Plug-in

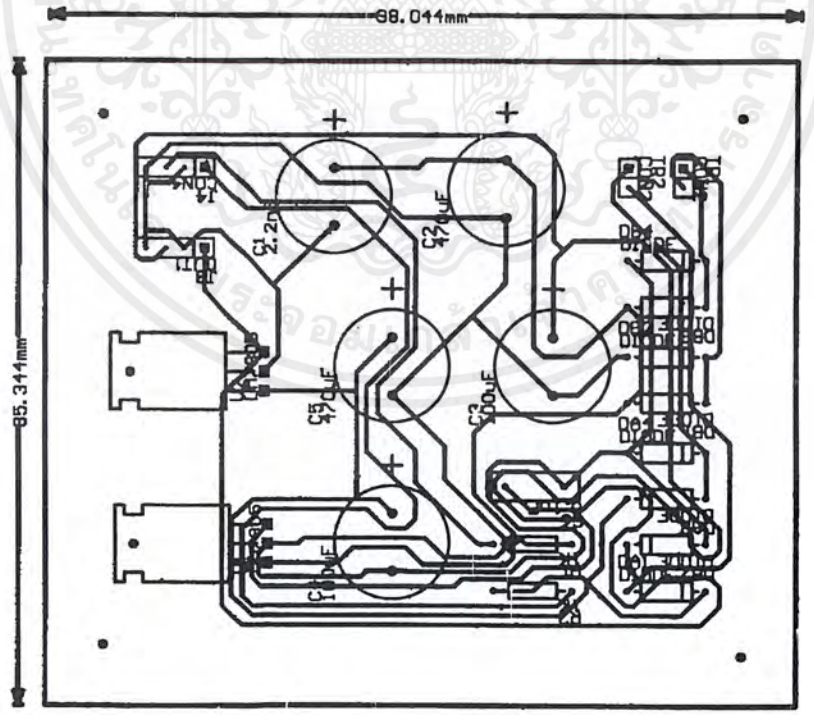
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ภาคผนวก ค.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

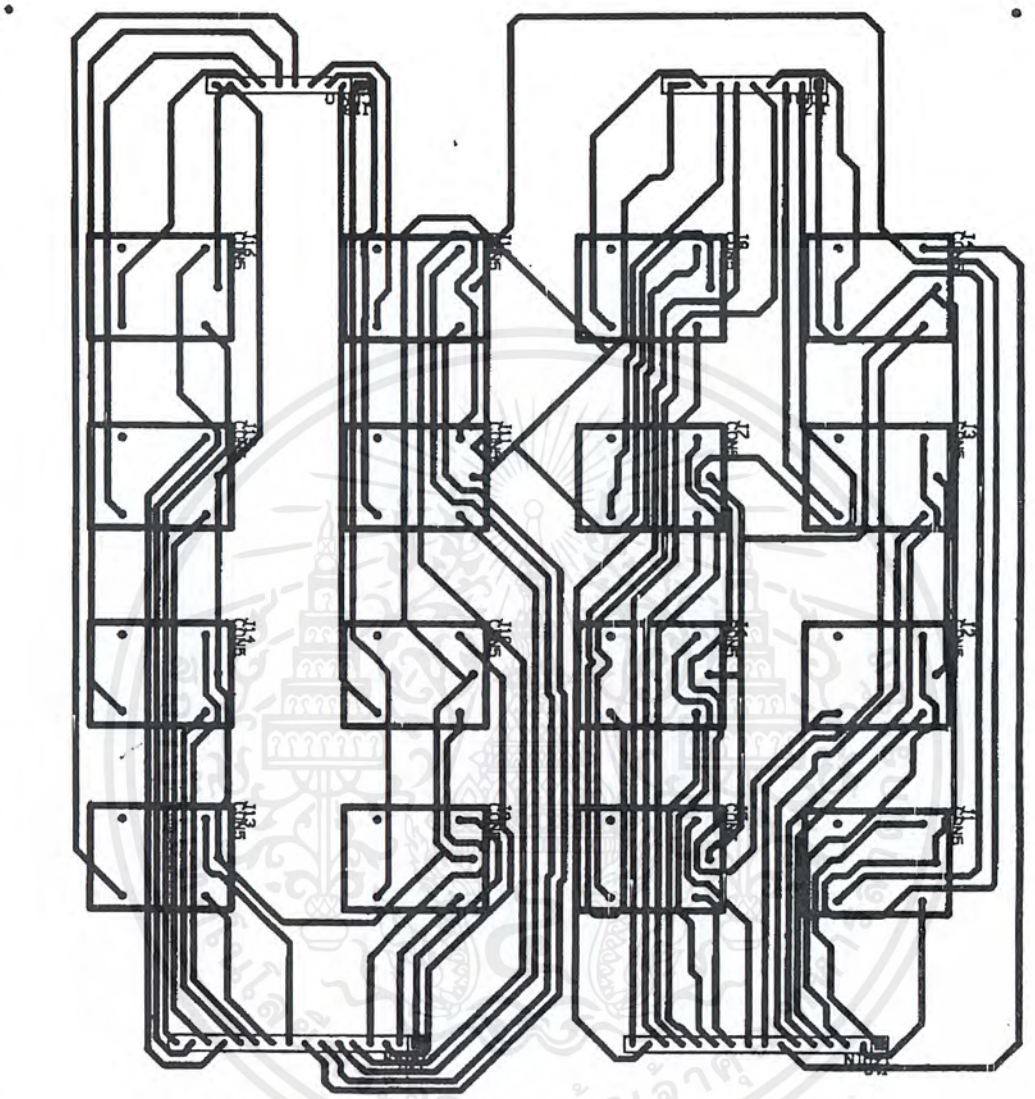



รูปที่ ค.1 แสดงสัญลักษณ์ตัวอุปกรณ์



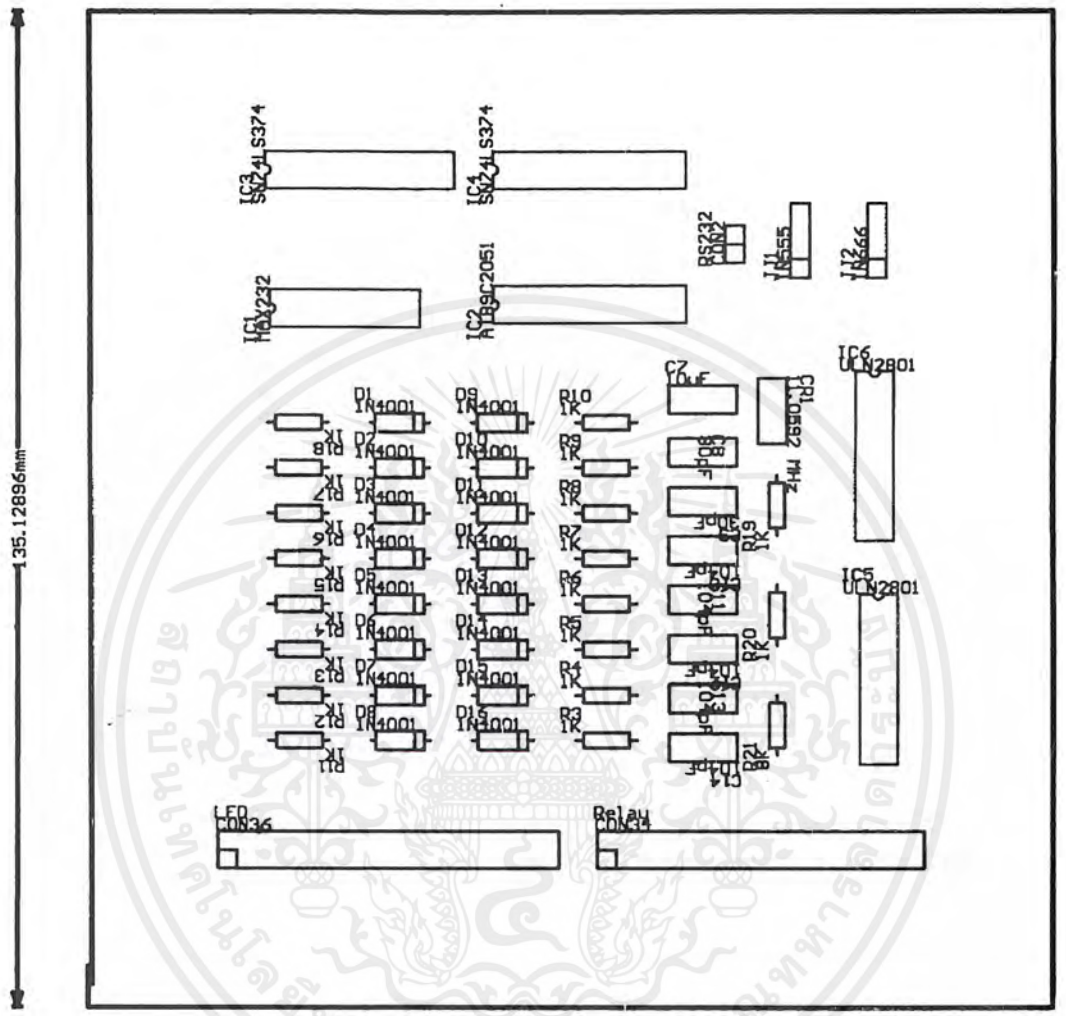

รูปที่ ค.2 แสดงลายวงจร Supply

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



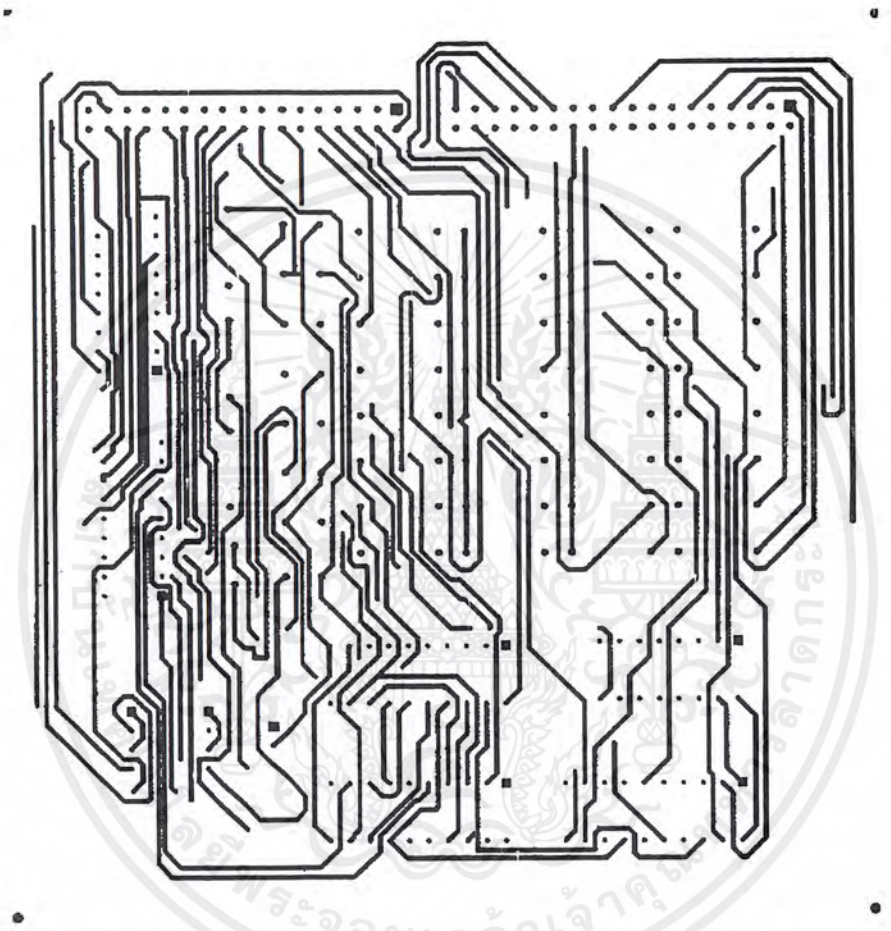
รูปที่ ค.3 แสดงลายวงจรรีเลย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



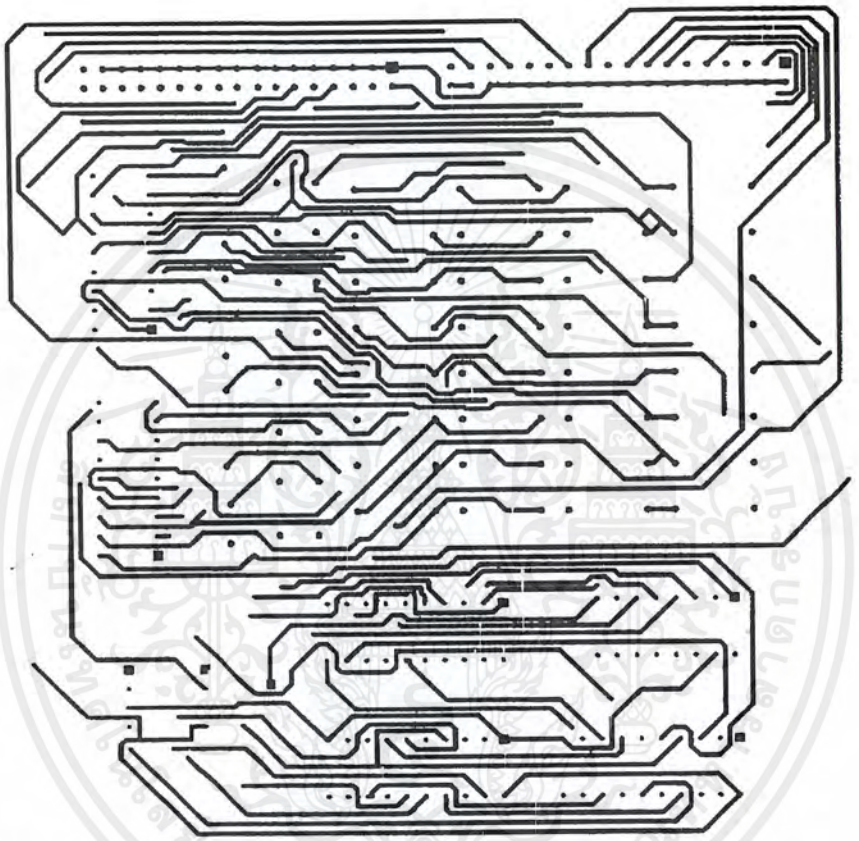

รูปที่ ค.4 แสดงสัญลักษณ์อุปกรณ์วงจรไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ค.5 แสดงลายวงจรไมโครคอนโทรลเลอร์(ด้านบน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ค.6 แสดงลายวงจรไมโครคอนโทรลเลอร์(ด้านล่าง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้