

ระบบกล้องรักษาความปลอดภัยผ่านอินเทอร์เน็ต
INTERNET SECURITY CAMERA SYSTEM



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมโทรคมนาคม
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทปีการศึกษา 2547

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบกล้องรักษาความปลอดภัยผ่านอินเทอร์เน็ต

INTERNET SECURITY CAMERA SYSTEM

ผู้จัดทำ

1. นายธัญย์ ขัตตินานนท์ 44010200
2. นายธีระ ศิริเจริญ 44010226
3. นายธีระชัย ศาตราวหะ 44010228



(ดร.พิพัฒน์ พรหมมี)

(อาจารย์ที่ปรึกษา)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบกล้องรักษาความปลอดภัยผ่านอินเทอร์เน็ต
INTERNET SECURITY CAMERA SYSTEM

โดย นายธัญย์ ชัดดินานนท์ 44010201
นายธีระ สิริเจริญ 44010226
นายธีระชัย สาตราวาหะ 44010228

อาจารย์ที่ปรึกษา คร. พิพัฒน์ พรหมมี

บทคัดย่อ

โครงการนี้ได้เสนอการสร้างระบบกล้องรักษาความปลอดภัยผ่านทางอินเทอร์เน็ต โดยระบบตรวจจับได้ว่าการเคลื่อนไหวกรณีที่มีผู้บุกรุก ระบบก็จะเริ่มทำการบันทึกภาพผู้บุกรุก และ ทำการแจ้งเตือนไปยังผู้ใช้งานโดยผ่านระบบโทรศัพท์มือถือ และ E-Mail โดยผู้ใช้งานจะสามารถตรวจสอบภาพของผู้บุกรุกผ่านระบบอินเทอร์เน็ตมายัง เซิร์ฟเวอร์ของระบบได้ โดยผ่านทางเว็บเบราว์เซอร์

Abstract

This project presents an Internet Security Camera System. This system will recording the intruder picture after the system detected something moving of specific area. The recorded pictures have been stored in the hard disk of the server then also informed to owner by E-Mail and SMS. The owner can access the server by internet web browser to immediately find out intruder's recorded pictures.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทที่ 1 บทนำ	1
บทที่ 2 ทฤษฎีและหลักการ	3
2.1 อินเทอร์เน็ต (Internet)	3
2.2 เวิลด์ ไวด์ เว็บ (World Wide Web)	5
2.2.1 ส่วนประกอบของ เวิลด์ ไวด์ เว็บ	5
2.2.1.1 แหล่งข้อมูลหรือเว็บไซต์	5
2.2.1.2 โปรแกรมเว็บเบราว์เซอร์	6
2.3 Web Server และ Web Browser	6
2.4 โฮมเพจ	6
2.5 การรับและส่งข้อมูลในอินเทอร์เน็ต	6
2.6 IP Address	7
2.7 ดีเอ็นเอส (Domain Name System:DNS)	8
2.8 HTML (Hypertext Markup Language)	8
2.9 ซีจีไอ (CGI : Common Gateway Interface)	10
2.10 ภาษา Script	12
2.11 ASP (Active Server Pages)	12
2.11.1 หลักการทำงานของ ASP	12
2.11.2 ออบเจกต์พื้นฐานของ ASP	13
2.10.3 ข้อดีของ ASP	15
2.11 ซ็อกเก็ต (Socket)	16
2.11.1 ตัวอย่างการสื่อสารด้วย Socket	17
2.11.2 ชนิดของซ็อกเก็ต	17
2.12 สถาปัตยกรรมโปรโตคอล TCP/IP	18
2.13 การทำงานของแต่ละเลเยอร์ใน TCP/IP	19
2.13.1 Network Interface Layer	19
2.13.2 Internet Layer	19
2.13.3 Transport Layer	20
2.13.3.1 TCP (Transmission Control Protocol)	20
2.13.3.2 UDP (User Datagram Protocol)	20
2.13.4 Application Layer	21
2.14 ภาษาวิซวลซีพลัสพลัส (Visual C++)	22
2.15 การเขียนวิซวลซีพลัสพลัสติดต่อผ่าน I/O Serial Port	22

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.16	ฐานข้อมูลใน ASP	24
2.16.1	การเชื่อมต่อกับฐานข้อมูล	24
2.17	การติดต่อฐานข้อมูลแบบ Microsoft Access	26
2.17.1	การเขียนสคริปต์ใช้งานฐานข้อมูลในออบเจ็กต์ ADO	26
2.18	มาตรฐานคำสั่งของโมเด็ม (AT Command)	29
2.18.1	มาตรฐานคำสั่ง โมเด็มของ Hayes	29
2.18.2	การใช้งานชุดคำสั่ง AT	29
2.18.2.1	ประเภทของชุดคำสั่ง AT	29
2.18.2.2	การออนไลน์และออฟไลน์	30
2.18.2.3	รูปแบบของชุดคำสั่ง AT	30
2.18.2.4	การสนทนาคำสั่งของโมเด็ม	30
2.19	รูปแบบของ Protocol Description Unit (PDU)	31
2.20	ไปรษณีย์อิเล็กทรอนิกส์ (Electronic Mail:E-mail)	32
2.20.1	มาตรฐานที่ใช้ในการรับส่งอีเมลล์	33
2.20.2	รูปแบบเอกสารแนบของอีเมลล์	33
บทที่ 3	การออกแบบและการสร้าง	34
3.1	การทำงาน	34
3.2	โปรแกรมแสดงภาพ (Video Capture)	35
3.3	โปรแกรมบันทึกภาพวีดีโอ	37
3.4	เซนเซอร์ผ่านพอร์ตอนุกรม	39
3.5	การทำงานในส่วนเว็บเบราว์เซอร์	40
3.6	ขั้นตอนการแปลงข้อความให้อยู่ในรูปออกเต็ด	41
3.7	การส่งอีเมลล์แบบ SMTP ผ่านทางเมลล์เซิร์ฟเวอร์	41
บทที่ 4	การทดลองการทำงาน	42
4.1	โปรแกรมบันทึกภาพวีดีโอ	42
4.2	การเก็บข้อมูลลงดาต้าเบส	43
4.3	เว็บเบราว์เซอร์	44
4.4	โปรแกรมส่งSMS	46
4.5	การรับค่าจากไมโครคอนโทรลเลอร์	48
4.6	โปรแกรมส่งอีเมลล์	48
4.7	โปรแกรม Internet Security Camera System	49

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5 บทวิจารณ์และบทสรุป	51
5.1 สรุปผลการทดลอง	51
5.2 บทวิจารณ์	51
ภาคผนวก ก.	52
ก.1 โปรแกรมส่วนติดต่อฐานข้อมูล	53
ก.2 โปรแกรมส่ง SMS	53
ก.3 โปรแกรมส่งอีเมลล์	53
ก.4 โปรแกรมส่งค่าจากไมโครคอนโทรลเลอร์	54
ภาคผนวก ข.	55
บรรณานุกรม	



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

	หน้า
บทที่ 1 บทนำ	1
บทที่ 2 ทฤษฎีและหลักการ	3
รูปที่ 2.1 ระบบเครือข่ายอินเทอร์เน็ต	4
รูปที่ 2.2 ภาพรวมของเว็บไซต์ไวด์เว็บ	5
รูปที่ 2.3 ตัวอย่างหน้าจอแสดงผลภาษาHTML	10
รูปที่ 2.4 ตัวอย่างหน้าจอแสดงการใช้งานCGI	11
รูปที่ 2.5 หลักการทำงานของฝั่ง เซิร์ฟเวอร์แอปพลิเคชัน	13
รูปที่ 2.6 ความสัมพันธ์ของออบเจกต์ต่างๆ	13
รูปที่ 2.7 หลักการทำงานของแอปพลิเคชัน ASP	15
รูปที่ 2.8 ตัวอย่างการลงIIS	16
รูปที่ 2.9 แสดงการเปรียบเทียบของเลขอร์ OSI และ TCP/IP	18
รูปที่ 2.10 แสดงการรับส่งอีเมลแบบมาตรฐาน	33
บทที่ 3 การออกแบบและการสร้าง	34
รูปที่ 3.1 แสดงการทำงานโดยทั่วไปของระบบ	34
รูปที่ 3.2 Block Diagramการทำงานของเครื่องเซิร์ฟเวอร์	35
รูปที่ 3.3 Block Diagram การตรวจสอบของผู้ใช้	35
รูปที่ 3.4 การ์ดแคบเจอร์ที่ใช้ในการดึงภาพจากกล้อง	36
รูปที่ 3.5 กล้องวงจรปิดขนาดจิ๋ว	36
รูปที่ 3.6 โทรศัพท์มือถือ Siemens C35และสาย Serial port	37
รูปที่ 3.7 การทำงานส่วนโปรแกรมบันทึกภาพ	39
รูปที่ 3.8 โฟลชาร์ตการทำงานของไมโครคอนโทรลเลอร์	39
บทที่ 4 การทดลองการทำงาน	42
รูปที่ 4.1 ผลการดึงภาพโดยโปรแกรม capture	42
รูปที่ 4.2 ภาพนิ่งที่ถูกบันทึกโดยการกดปุ่ม Take Image	43
รูปที่ 4.3 คาด้าเบสที่เก็บไว้ใน Microsoft Access	44
รูปที่ 4.4 ไฟล์ที่ทำการบันทึกไว้บนเว็บเบราว์เซอร์	45
รูปที่ 4.5 ภาพวิดีโอที่ถูกเปิดขึ้นจากเว็บเบราว์เซอร์	45
รูปที่ 4.6 คำสั่ง AT ที่กำหนดให้โทรศัพท์เคลื่อนที่ส่ง Output the serial number (IMEI)	46
รูปที่ 4.7 โปรแกรมส่ง SMS แสดงค่าที่รับได้จากโทรศัพท์เคลื่อนที่	47
รูปที่ 4.8 คำสั่งส่ง SMS ผ่าน Hyper Terminal	47
รูปที่ 4.9 ค่าที่รับได้จากไมโครคอนโทรลเลอร์โดยใช้HyperTerminal	48
รูปที่ 4.10 ข้อความเตือนและไฟล์รูปภาพจากอีเมลล์ที่ได้รับ	49
รูปที่ 4.11 โปรแกรม Internet Security Camera System ขณะกำลังบันทึกภาพ	50



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้า
บทที่ 1 บทนำ	1
บทที่ 2 ทฤษฎีและหลักการ	3
ตารางที่ 2.1 ออบเจกต์พื้นฐานของ ASP	14
ตารางที่ 2.2 ชื่อโพรไวเดอร์ในการติดต่อฐานข้อมูลแต่ละชนิด	31
ตารางที่ 2.3 ข้อความ PDU ที่ใช้ในการส่ง SMS	31
บทที่ 3 การออกแบบและการสร้าง	34
ตารางที่ 3.1 การแปลงตัวอักษรให้เป็น septet	41
ตารางที่ 3.2 กระบวนการแปลงจาก septet ให้เป็นออกเต็ต	41
บทที่ 4 การทดลองการทำงาน	42
บทที่ 5 บทวิจารณ์และบทสรุป	50



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

วัตถุประสงค์

1. เพื่อใช้ในการติดตั้งร่วมกับระบบรักษาความปลอดภัยโดยการแสดงภาพจากกล้องวงจรปิดผ่านโปรแกรมแสดงภาพและบันทึกภาพเมื่อมีผู้บุกรุก
2. เพื่อให้มีการแจ้งเตือนไปยังผู้ควบคุมระบบ โดยทันทีที่สามารถตรวจสอบผู้บุกรุกได้ โดยจะแจ้งเตือนทาง SMS ผ่านโทรศัพท์มือถือ และทาง E-mail
3. เพื่อให้ผู้ควบคุมสามารถตรวจสอบไฟล์วิดีโอที่ทำการบันทึกไว้ได้ผ่านทาง Internet

แนวคิดและความเป็นมา

ในปัจจุบันนี้ระบบรักษาความปลอดภัยในหน่วยงานต่างๆ เป็นสิ่งที่จำเป็นและมีอามองข้ามไม่ได้ ทั้งเหตุผลจากการก่อการร้ายที่นับวันจะทวีความรุนแรงมากขึ้น และเนื่องจากความเจริญก้าวหน้าทางเทคโนโลยีทำให้วิธีหรือรูปแบบของการก่อการร้ายนั้นมีความหลากหลายและยากต่อการเตรียมการรับมือของเจ้าหน้าที่ ดังนั้นระบบความปลอดภัยก็จะต้องมีการพัฒนาในด้านขีดความสามารถต่างๆ เพื่อที่จะได้คอยรับมือจากเหตุการณ์ที่ไม่ได้คาดคิดไว้ล่วงหน้าได้อย่างทันท่วงที และสามารถยุติเหตุการณ์นั้นได้ด้วยการเกิดผลเสียที่น้อยที่สุด

เมื่อหลายปีก่อนระบบรักษาความปลอดภัยจะต้องมีผู้ควบคุม โดยผู้ควบคุมจะต้องคอยเฝ้าดูความเคลื่อนไหวต่างๆ ตลอดเวลา และจะต้องตรวจสอบอยู่ที่ห้องควบคุมระบบรักษาความปลอดภัยเท่านั้น ไม่สามารถที่จะตรวจสอบโดยที่ผู้ตรวจสอบอยู่ที่อื่นได้ ดังนั้นผู้ตรวจสอบต้องคอยตรวจสอบอยู่ตลอดเวลา ซึ่งเป็นระบบที่พึ่งพาการทำงานของมนุษย์มากเกินไป ทำให้ระบบรักษาความปลอดภัยแบบนี้มีแนวโน้มอาจเกิดความผิดพลาดต่างๆ ขึ้นได้โดยง่าย เนื่องจากมนุษย์นั้นมีขีดจำกัดของตัวเอง ดังนั้นจึงมีการพัฒนาระบบรักษาความปลอดภัยผ่านกล้องวงจรปิดในด้านต่างๆ มากมายเพื่อที่จะทำให้ลดภาระการทำงานของมนุษย์ให้น้อยลง โดยเพิ่มขีดความสามารถของระบบให้สามารถมีกระบวนการตรวจสอบต่างๆ ในตัวมันเองให้มากขึ้น ซึ่งจะช่วยลดความผิดพลาดต่างๆ ที่อาจจะเกิดขึ้น ทำให้ระบบรักษาความปลอดภัยนั้นเข้มแข็งขึ้น และยากต่อการกระทำที่อาจเกิดจากกลุ่มของผู้ไม่หวังดีได้

ระบบรักษาความปลอดภัยในโปรแกรมฉบับนี้นั้นเป็นระบบรักษาความปลอดภัยผ่านกล้องวงจรปิดที่มีการแสดงภาพจากพื้นที่รักษาความปลอดภัยที่มีการติดตั้งกล้องวงจรปิดไว้แบบตลอดเวลาผ่านโปรแกรมแสดงภาพที่เครื่อง Server และมี การติดตั้งเซิร์ฟเวอร์ไว้ที่ทางผ่านเข้าออกของพื้นที่รักษาความปลอดภัยเพื่อคอยตรวจสอบการบุกรุก ถ้าเซิร์ฟเวอร์สามารถจับความเปลี่ยนแปลงได้ก็จะส่งสัญญาณให้โปรแกรมแสดงภาพมีการบันทึกภาพวิดีโอเป็นเวลา 7 วินาที ในขณะเดียวกันก็จะทำการแจ้งเตือนไปยังผู้ควบคุมระบบทาง SMS ผ่านโทรศัพท์มือถือและทาง E-mail ทำให้ผู้ควบคุมระบบรักษาความปลอดภัยนี้สามารถทราบได้ว่ามีผู้บุกรุกแม้ไม่ได้กำลังคอยเฝ้าดูอยู่ที่หน้าจอคอมพิวเตอร์ที่เครื่อง Server จะได้รับทราบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เหตุการณ์และแก้ไขเหตุการณ์เฉพาะหน้านี้ได้ทันท่วงที โดยผู้ควบคุมระบบสามารถดูภาพจากกล้องวงจรปิดที่ทำการบันทึกไว้ผ่านทาง Internet ได้ตลอดเวลาและทุกที่ที่สามารถเชื่อมต่อ Internet ได้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและหลักการ

2.1 อินเทอร์เน็ต (Internet)

อินเทอร์เน็ตหรือทางด่วนข้อมูล เป็นระบบเครือข่ายคอมพิวเตอร์ขนาดใหญ่ที่เชื่อมโยงระหว่างคอมพิวเตอร์ทั่วโลกไว้ด้วยกัน เหมือนโครงข่ายใยแมงมุม โดยมีจุดประสงค์เริ่มแรกในการสำรองข้อมูลสำคัญๆต่างๆไม่ให้สูญหาย แต่ปัจจุบันอินเทอร์เน็ตถูกนำมาใช้มากกว่าการสำรองข้อมูล ไม่ว่าจะเป็นการติดต่อสื่อสารส่งข้อความข้ามประเทศ การประชาสัมพันธ์ การส่งข้อมูล การค้าขาย การประมูล การรักษาพยาบาล และอื่นๆอีกมากมายแล้วแต่มนุษย์จะคิดขึ้นได้ ทำให้ปัจจุบันมีผู้สนใจมากมายนำเครื่องคอมพิวเตอร์ของตนมาทำการเชื่อมต่อกับระบบอินเทอร์เน็ต เพื่อรับข้อมูลข่าวสารกันมากขึ้น ทำให้โครงข่ายอินเทอร์เน็ตมีอัตราการเติบโตที่สูงขึ้นจากจำนวนผู้ใช้ที่มากขึ้นนั่นเอง

อินเทอร์เน็ต เป็นเครือข่ายคอมพิวเตอร์ที่มีขนาดใหญ่มาก เกิดจากการเชื่อมต่อเครือข่ายคอมพิวเตอร์จำนวนมากมาในโลกเข้าด้วยกัน ไม่ว่าจะเป็นเครือข่ายขนาดเล็ก เช่น ระบบแบบใกล้เคียงหรือระบบเครือข่ายขนาดใหญ่ เช่น ระบบเครือข่าย หรือ โฮสต์ (Host) ซึ่งมีอยู่หลายชนิดหลายยี่ห้อ โดยจะมีการกำหนดข้อตกลงในการสื่อสารที่เรียกว่า โพรโตคอล (Protocol) ขึ้นมา เพื่อให้คอมพิวเตอร์แต่ละชนิดสามารถติดต่อสื่อสารถึงกันได้ โพรโตคอลมาตรฐานที่ใช้ในการสื่อสารบนอินเทอร์เน็ตจะมีชื่อเรียกว่า TCP/IP (Transmission Control Protocol/Internet Protocol)

ถ้าให้เปรียบเทียบโพรโตคอลเหมือนภาษาที่ใช้ในการสื่อสารของมนุษย์ซึ่งมีมากมายหลายภาษา เช่นเดียวกับโพรโตคอล ซึ่งจะมีโพรโตคอลอยู่หลายแบบด้วยกัน TCP/IP นั้นจะเปรียบได้กับภาษาอังกฤษ เนื่องจากเป็นโพรโตคอลที่คอมพิวเตอร์ของระบบเข้าใจและมีการใช้โพรโตคอลนี้ร่วมกัน เพื่อการเชื่อมต่อเข้าสู่โลกอินเทอร์เน็ต

เมื่อเครื่องคอมพิวเตอร์ต่างๆพูดคุยกันรู้เรื่องโดยใช้โพรโตคอลเดียวกัน การติดต่อสื่อสารก็จะเริ่มขึ้น โดยการติดต่อสื่อสารจะมีอยู่ 2 แบบ คือ แบบส่งข้อมูล และแบบรับข้อมูล โดยเครื่องคอมพิวเตอร์ที่ทำการส่งข้อมูลเราจะเรียกว่าเครื่องให้บริการ (Server) และเครื่องคอมพิวเตอร์ที่รับข้อมูลจะเรียกว่าเครื่องรับบริการ (Client) ซึ่งการติดต่อสื่อสารในลักษณะนี้เราเรียกว่าการติดต่อกันแบบ Client-Server

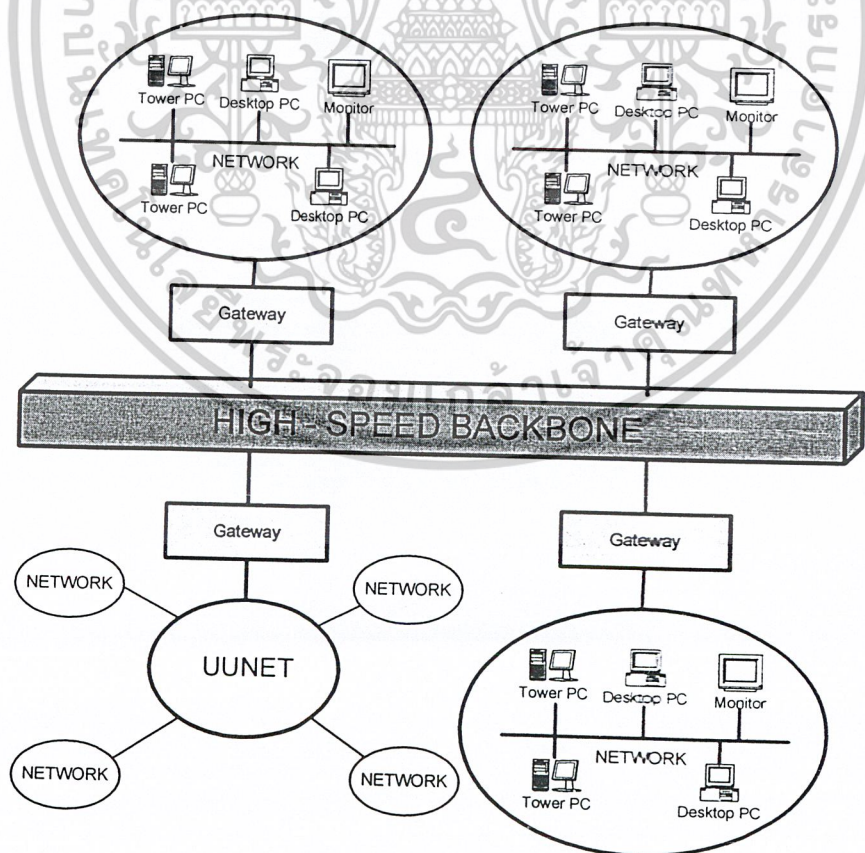
เมื่อไม่กี่ปีก่อนหน้านี้นี้มีผู้ที่สนใจใช้งานอินเทอร์เน็ตไม่มากนัก เนื่องจากการใช้งานอินเทอร์เน็ตไม่ว่าจะเป็นการค้นหาข่าวสารข้อมูล การรับส่งจดหมายอิเล็กทรอนิกส์ การทำสำเนาไฟล์ จะอยู่ในรูปแบบของตัวอักษร (Text mode) เท่านั้น ไม่มีการแสดงที่เป็นรูปภาพ เสียง และไม่มีตัวอักษรแบบต่างๆปรากฏให้เห็นแต่อย่างใด นอกจากนี้ผู้ใช้ต้องเรียนรู้และจดจำคำสั่งคอมพิวเตอร์มากมาย เช่น ต้องเรียนรู้คำสั่งเบื้องต้นของยูนิกซ์ (UNIX) เนื่องจากเมื่อจะมีการใช้อินเทอร์เน็ต เครื่องคอมพิวเตอร์ของผู้ใช้จะถูกเปลี่ยนให้เป็นเทอร์มินัลของโฮสต์คอมพิวเตอร์ที่ให้บริการอินเทอร์เน็ต และโฮสต์ส่วนมากจะทำงานอยู่ภายใต้ระบบปฏิบัติการยูนิกซ์ ดังนั้นจึงจะต้องเรียนรู้คำสั่งที่เป็นอักษรด้วยตนเองให้โฮสต์คอมพิวเตอร์ทำงานตามที่เรต้องการใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จนกระทั่งมีบริการที่เรียกว่า เวิลด์ ไวด์ เว็บ หรือเครือข่ายใยแมงมุมเกิดขึ้น ทำให้ความนิยมการใช้อินเทอร์เน็ตสูงขึ้นเป็นทวีคูณ เนื่องจาก เวิลด์ ไวด์ เว็บเป็นบริการอันหนึ่งที่มีอยู่ในอินเทอร์เน็ตทำให้การใช้งานอินเทอร์เน็ตเป็นสิ่งที่ยั่งยืนผู้ใช้ไม่ต้องจดจำคำสั่งของ UNIX อีกต่อไป การอ่านและค้นหาข่าวสารข้อมูลทำได้เพียงการคลิกปุ่มบนเมาส์เพียงอย่างเดียว นอกจากนี้ข่าวสารข้อมูลที่เก็บอยู่ในอินเทอร์เน็ตจะอยู่ในรูปแบบของสื่อผสม (Multimedia) ที่เรียกว่า “เว็บเพจ” (Web page) อันประกอบไปด้วยตัวอักษรข้อความที่มีรูปแบบต่างๆ รูปภาพ เสียง ภาพเคลื่อนไหว วิดีโอ และไฮเปอร์เท็กซ์ (Hypertext) ซึ่งเป็นการแสดงข้อความที่มีการเชื่อมโยงถึงกันได้เหมือนกับการเรียกใช้ Help ในวินโดวส์ โดยผู้ใช้สามารถเรียกดูเอกสารหนึ่งได้ ซึ่งเป็นลักษณะการเชื่อมต่อที่คล้ายกับ ใยแมงมุม จึงมีการตั้งชื่อบริการนี้ว่าเป็นเครือข่ายใยแมงมุม

อินเทอร์เน็ตประกอบด้วยสายสื่อสารความเร็วสูงเป็นแบคโบน(Back bone) เป็นสายโทรศัพท์ตามมาตรฐาน TS สามารถวิ่งด้วยความเร็ว 44.736 เมกะบิตต่อวินาที

เครือข่ายที่ต้องการเชื่อมโยงโดยตรงกับอินเทอร์เน็ตจะต่อกับแบคโบน ด้วยอุปกรณ์ที่เรียกว่า “เกตเวย์” (Gateway) ซึ่งเป็นเครื่องคอมพิวเตอร์ที่ทำหน้าที่ควบคุมการผ่านเข้า-ออกของข่าวสารระหว่างเครือข่ายกับแบคโบน เกตเวย์ทุกตัวสามารถกำหนดการติดต่อกับเกตเวย์ตัวอื่น หรือเครือข่ายอื่นได้โดยใช้ไอพีแอดเดรส (IP Address) ของเครือข่ายอ้างอิงถึงกัน เช่น อินเทอร์เน็ตประเทศไทยเป็นเกตเวย์ของเครือข่ายในกลุ่มติดต่อกับเกตเวย์ของ UNUNET ที่รัฐเวอร์จิเนีย สหรัฐอเมริกา

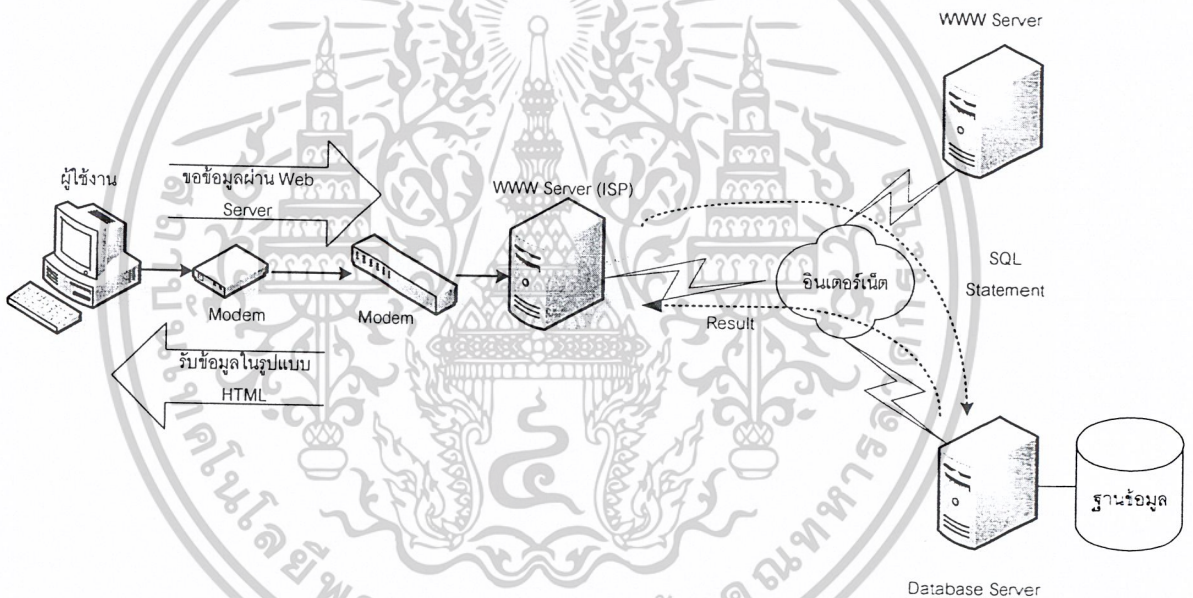


รูปที่ 2.1 ระบบเครือข่ายอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 เวิลด์ ไรด์ เว็บ (World Wide Web)

เป็นบริการรูปแบบหนึ่งในอินเทอร์เน็ตที่เชื่อแน่ว่าเราทุกคนต้องได้เคยสัมผัส หรือที่เราเรียกกันสั้นๆว่า เว็บ (Web) นั่นเอง การให้บริการในรูปแบบนี้คือการเรียก Browser เช่น IE, Netscape จากเครื่องของเราและทำการระบุ URL เพื่อใช้ในการอ้างที่อยู่ที่เกี่ยวข้อง โดยชื่อ URL ที่เราป้อนจะถูกส่งไปที่ DNS(Domain Name Server) ซึ่งเป็นเซิร์ฟเวอร์ที่ใช้ในการเปลี่ยนชื่อ URL (เช่น www.yahoo.com เป็นต้น) ให้กลายเป็นชื่อแบบตัวเลขหรือไอพีแอดเดรส(IP Address) โดยหมายเลขไอพีนั้นจะถูกใช้ในการอ้างอิงตำแหน่งเครื่องคอมพิวเตอร์ในอินเทอร์เน็ต โดยเครื่องทุกเครื่องที่อยู่ในอินเทอร์เน็ตจะมีหมายเลขไอพีไม่เหมือนกัน ทำให้เราสามารถระบุที่อยู่ของเครื่องที่เกี่ยวข้องที่เราต้องการเปิดดูได้ โดยเราอาจจะเปรียบเทียบหมายเลขไอพีเหมือนกับเลขที่บ้านของเราในการส่งจดหมายก็ได้ จากนั้นเมื่อเราได้เลขไอพีที่ต้องการแล้ว คำร้องขอเว็บของเราก็จะถูกส่งไปยังเครื่องเซิร์ฟเวอร์ที่มีหมายเลขไอพีนั้น และเมื่อเครื่องเซิร์ฟเวอร์ได้รับคำร้องขอจากเราก็จะส่งเว็บที่เกี่ยวข้องอยู่ในเครื่องเซิร์ฟเวอร์มาแสดงผลที่เครื่องเรา เท่านั้นเราก็สามารถเปิดดูเว็บได้แล้ว



รูปที่ 2.2 ภาพรวมของเวิลด์ ไรด์ เว็บ

2.2.1 ส่วนประกอบของ เวิลด์ ไรด์ เว็บ

การจะใช้บริการ เวิลด์ ไรด์ เว็บ ได้นั้นจำเป็นต้องมีส่วนประกอบดังนี้

2.2.1.1 แหล่งข้อมูลหรือเว็บไซต์

เว็บไซต์ หรือเว็บเซิร์ฟเวอร์ คือระบบคอมพิวเตอร์ที่เป็นแหล่งเก็บเว็บเพจที่ผู้ใช้บริการสามารถเรียกดูเว็บเพจที่เกี่ยวข้องในเว็บไซต่นั้นได้ ซึ่งเครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็นเว็บไซต์อาจจะใช้ระบบปฏิบัติการยูนิกซ์หรือวินโดวส์เอ็นทีก็ได้ และจะต้องมีการจัดการติดตั้งโปรแกรมจัดการ ที่จะทำให้เครื่องคอมพิวเตอร์เครื่องนั้นทำหน้าที่เป็นเว็บไซต์ ตัวอย่างเช่น ในกรณีที่เครื่องคอมพิวเตอร์นั้นใช้ระบบปฏิบัติการวินโดวส์เอ็นทีจะมีซอฟต์แวร์เว็บเซิร์ฟเวอร์ให้เลือกใช้ เช่น โปรแกรม Purveyou HTTP เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นับญาติให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Server ของสถาบัน European Microsoft Windows NT Academic Center (EMWAC) โปรแกรมเว็บไซต์ของบริษัท O'Reilly and Associate หรือถ้าใช้ระบบปฏิบัติการยูนิกซ์ก็อาจใช้เว็บเซิร์ฟเวอร์เป็น NSCA http, Apache เป็นต้น

ผู้ที่เป็นเจ้าของเว็บไซต์จะจัดสร้างเว็บเพจของตนเก็บไว้ที่เว็บไซต์นั้น เพื่อให้ผู้ใช้คนอื่นๆทั่วโลกสามารถเข้ามาดูข้อมูลที่เก็บไว้ในเว็บไซต์นั้นได้ ตัวอย่างเช่น เว็บเพจของสำนักข่าว CNN จะเก็บอยู่ที่เว็บไซต์ www.cnn.com เป็นต้น

2.2.1.2 โปรแกรมเว็บเบราว์เซอร์

เป็นโปรแกรมหรือซอฟต์แวร์ที่ใช้ในการเข้าสู่ เวิลด์ไวด์ เว็บ จะเปิดดูเว็บเพจที่เก็บอยู่ในเว็บไซต์ใดๆ ตัวอย่างของโปรแกรมเว็บเบราว์เซอร์ ที่มีให้ดาวน์โหลดใช้กันได้ฟรีได้แก่ Netscape Navigator จากบริษัท Netscape Communication และโปรแกรม Microsoft Internet Explorer จากบริษัท ไมโครซอฟต์ เป็นต้น ซึ่งโปรแกรมเว็บเบราว์เซอร์ส่วนใหญ่จะทำงานร่วมกับโปรแกรมวินโดวส์ และนอกจากจะใช้เพื่อดูเว็บเพจจากเว็บไซต์ใดๆแล้วหลายโปรแกรมยังมีความสามารถอื่นๆ เช่น บริการส่งจดหมาย อีเล็กทรอนิกส์ การค้นหาข้อมูล การโอนถ่าย โปรแกรมด้วย FTP เป็นต้น

2.3 Web Server และ Web Browser

ศัพท์อีก 2 คำที่เราควรมาทำความเข้าใจกันในตอนนี้คือ Web Server และ Web Browser

Web Server เป็นโปรแกรมที่อยู่ในเครื่องเซิร์ฟเวอร์ ทำหน้าที่ในการรับคำร้องขอ และทำการประมวลผลแล้วส่งไปให้เครื่องฝั่งไคลเอนต์ หรือจะเรียกว่าเป็นโปรแกรมที่ใช้ในการให้บริการเว็บนั่นเอง

Web Browser เป็นโปรแกรมที่อยู่เครื่องฝั่งไคลเอนต์ มีหน้าที่ในการส่งข้อมูลในการร้องขอเว็บ และนำเสนอข้อมูลเว็บโดยตัว เว็บเบราว์เซอร์จะมีความเข้าใจในภาษามาตรฐานของเว็บก็คือ ภาษาHTML และจะสามารถแปลงภาษาHTML ให้เป็นหน้าเว็บหรือหน้าเอกสารให้เราได้ใช้งาน

2.4 โฮมเพจ

โฮมเพจ (Home Page) จะหมายถึงหน้าแรกของเว็บเพจทั้งหมดที่ผู้ใช้บริการอินเทอร์เน็ตจะพบเมื่อมีการเข้าไปยังเว็บไซต์ใดๆ โฮมเพจเปรียบเสมือนสารบัญ และคำนำที่เจ้าของเว็บไซต์ สร้างขึ้นเพื่อจะใช้ประชาสัมพันธ์องค์กรของตนว่าให้บริการในสิ่งใดบ้าง นอกจากนี้แล้วภายใน โฮมเพจหนึ่งๆก็อาจมีการเชื่อมกับเว็บเพจอื่นๆเป็นจำนวนมากได้

2.5 การรับและส่งข้อมูลในอินเทอร์เน็ต

ตัวอย่างต่อไปนี้จะแสดงรายละเอียดการรับและการส่งข้อมูลระหว่างไคลเอนต์กับเซิร์ฟเวอร์ เมื่อไคลเอนต์พิมพ์ <http://www.test.com> ก็จะเกิดการ ทำงานในขั้นตอนต่างๆดังนี้

1. เมื่อไคลเอนต์พิมพ์ที่อยู่ URL ในที่นี้คือ <http://www.test.com> เบราวเซอร์ก็จะทำการ request HTTP พร้อมกับผ่านค่าชื่อที่อยู่ปลายทางไปยังซอฟต์แวร์ TCP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ซอฟต์แวร์ TCP ก็จะแปลงชื่อปลายทางไปยังที่อยู่ไอพี (IP address) โดยส่งการควิรี่ (Query) ไปยังตัวเซิร์ฟเวอร์ที่แก้ชื่อ URL ซึ่งเป็นเซิร์ฟเวอร์ DNS
3. จะเริ่มสร้างการติดต่อแบบเสมือนกับเซิร์ฟเวอร์ปลายทาง โดยมีการเซตค่าระยะเวลาที่กำหนด และการรับทราบ Handshaking ในแต่ละแพ็กเกจเพื่อให้แน่ใจได้ว่าการโอนย้ายข้อมูลระหว่างไคลเอนต์กับเซิร์ฟเวอร์มีความเชื่อถือได้ ในกรณีที่ข้อมูลใหญ่มาก TCP จะแบ่งข้อมูลออกเป็นแพ็กเกจย่อยๆแล้วเพิ่มส่วนหัวที่มีหมายเลขพอร์ตและอื่นๆให้แก่แพ็กเกจย่อยๆเหล่านั้น เช่น ที่อยู่ปลายทางและข้อมูลสัญญาณนาฬิกาเพื่อ synchronize กับข้อมูล
4. TCP จะส่งแพ็กเกจไปยังเลเยอร์ไอพี ในเลเยอร์ไอพีจะเพิ่มส่วนหัวไอพี ซึ่งจะเป็นที่อยู่ไอพี ทั้งไคลเอนต์และเซิร์ฟเวอร์ และอื่นๆให้กับข้อมูล และส่งต่อไปยัง Ethernet
5. ไดรเวอร์ Ethernet จะเพิ่มส่วนอื่นๆอีก และส่งแพ็กเกจไปยัง Router โดยขึ้นอยู่กับตารางเส้นทางภายใน จากนั้นตัว Router จะหาเส้นทางหรือเน็ตเวิร์คที่มีประสิทธิภาพสูงสุด เพื่อส่งแพ็กเกจออกไปยังเซิร์ฟเวอร์ปลายทาง
6. เมื่อถึงเซิร์ฟเวอร์แล้ว เลเยอร์ TCP ของเซิร์ฟเวอร์ปลายทางจะส่ง Acknowledgment ไปยังไคลเอนต์ว่าได้รับแพ็กเกจแล้ว ถ้าเซิร์ฟเวอร์ปลายทางไม่ได้รับในระยะเวลาที่กำหนดไว้ในไคลเอนต์จะเกิดขึ้น มีผลให้ไคลเอนต์ต้องส่งไปอีกรอบ เลเยอร์ TCP/IP ของเซิร์ฟเวอร์ปลายทางจะต่อส่วนหัว TCP/IP จากแต่ละแพ็กเกจให้เป็นข้อมูล ซึ่งจะเหมือนกับไคลเอนต์
7. TCP จะทำการส่ง Request ไปยัง HTTP เซิร์ฟเวอร์ ซึ่งจะกระจายค่าออกมา เมื่อกระจายค่าเรียบร้อยแล้วเซิร์ฟเวอร์จะส่งไฟล์ HTML พร้อมกับผ่านไฟล์ข้อมูลไปยังเลเยอร์ TCP/IP แล้วแพ็กเกจส่งไปตามเส้นทางกลับไปยังไคลเอนต์ (โดยใช้วิธีเดียวกับที่ไคลเอนต์ส่งไปยังเซิร์ฟเวอร์)
8. ท้ายสุดแล้วแพ็กเกจข้อมูลทั้งหมดจะถูกต่อใหม่เมื่อถึงฝั่งไคลเอนต์ และบราวเซอร์ก็จะกระจายข้อมูลแล้วแสดงบนหน้าจอไคลเอนต์

2.6 IP Address

เนื่องจากในระบบเครือข่ายอินเทอร์เน็ตจะใช้โปรโตคอล TCP/IP เป็นมาตรฐานในการสื่อสารข้อมูลซึ่งจะมีการกำหนดหมายเลขประจำตัวที่ไม่ซ้ำกัน ให้กับคอมพิวเตอร์ทุกเครื่องที่มีการเชื่อมกันอยู่ในระบบเครือข่าย หมายเลขประจำตัวนี้จะถูกเรียกว่า IP Address หรือหมายเลข IP โดยมีรูปแบบเป็นชุดของตัวเลข 4 ชุดที่คั่นด้วยเครื่องหมายจุด เช่น 202.44.192.43 ตัวเลขในแต่ละชุดจะมีขนาด 8 บิต ดังนั้นในแต่ละชุดจะมีค่าได้ตั้งแต่ 0 ถึง $2^8 - 1 = 225$ เท่านั้น ซึ่งจะทำให้สามารถกำหนดหมายเลข IP ให้กับเครื่องคอมพิวเตอร์ได้ทั้งหมดถึงสี่พันล้านเลขหมายที่ไม่ซ้ำกันเลย

ดังนั้นเมื่อมีการติดตั้งเครื่องคอมพิวเตอร์ที่เป็นโฮสต์คอมพิวเตอร์เพื่อเชื่อมต่อเข้ากับอินเทอร์เน็ตและให้บริการต่างๆจำเป็นต้องขอ หมายเลข IP ประจำหน่วยงาน Internet Network Information Center (InterNIC) ขององค์กร Network Solution Incorporated (NSI) ที่รัฐเวอร์จิเนีย สหรัฐอเมริกา แต่ถ้าผู้ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สมัครเป็นสมาชิกกับหน่วยงานที่ทำหน้าที่เป็นผู้ให้บริการอินเทอร์เน็ต (Internet Service Provider:ISP) (ในประเทศไทยมีอยู่หลายหน่วยงาน) ก็ไม่ต้องขอหมายเลข IP ให้แก่ผู้ใช้งาน

2.7 ดีเอ็นเอส (Domain Name System:DNS)

การใช้หมายเลข IP ซึ่งเป็นตัวเลขล้วนๆ ในการอ้างอิงถึงคอมพิวเตอร์แต่ละเครื่องนั้น จะมีข้อเสียคือจำได้ยาก และก่อให้เกิดความสับสนได้ง่าย จึงมีการพัฒนาวิธีการอ้างอิงถึงหมายเลข IP แบบใหม่ที่เรียกว่า Domain Name System (DNS) ขึ้นมา

ดีเอ็นเอส เป็นเทคนิคการเปลี่ยนหมายเลข IP ที่เป็นตัวเลขให้เป็นตัวอักษรแทน หมายเลข IP เป็น 202.12.97.1 ผู้ใช้บริการสามารถเขียนเป็นชื่อโดเมนคือ kku.kku.ac.th แทน เครื่องคอมพิวเตอร์ที่เรียกว่า ดีเอ็นเอส จะทำหน้าที่ในการแปลงจากชื่อ โดเมนเนมให้เป็นหมายเลข IP อีกทีหนึ่ง

2.8 HTML (Hypertext Markup Language)

HTML เป็นรูปแบบมาตรฐานในการสร้างหน้าเว็บ โดยที่ภาษา HTML เป็นภาษาที่ง่ายต่อการเรียนรู้และเข้าใจ โดยเราสามารถใช้เวลาสั้นๆ ในการสร้างเว็บเพจโดยใช้ HTML ได้ โดยมีพื้นฐานอยู่ในรูปแท็ก <...> โดยมีลักษณะการใช้แท็ก 2 แบบดังนี้

1. แท็กมีจุดเริ่มต้นและสิ้นสุด
2. แท็กเดี่ยวๆ

ตัวอย่างแท็ก HTML ที่จะต้องมีจุดเริ่มต้นและจุดสิ้นสุด เช่น และ ซึ่งเป็นการทำให้ตัวอักษรเป็นตัวหนา ถ้าเราเขียน ไว้ต้นเอกสาร และไม่ได้ปิดด้วย แล้ว ทั้งเอกสารจะกลายเป็นตัวหนาทั้งหมด เพราะไม่มี ปิดเพื่อให้รู้ว่าตรงไหนคือการสิ้นสุดการแสดงตัวหนานั้นเอง และโค้ด HTML ที่เป็นคำสั่งเดี่ยวๆ ก็คือ เขียน โค้ด ไปแล้ว ไม่ต้องเขียนจุดสิ้นสุด เช่น <hr> จะเป็นการสร้างเส้นกั้นแนวนอน หรือ <p> จะเป็นการขึ้นย่อหน้าใหม่ เป็นต้น

ในการเขียน HTML มีหลักการดังนี้

1. เริ่มต้นไฟล์จะต้องเขียนแท็ก <html> และท้ายไฟล์จะต้องปิดด้วย </html>
2. ต่อจาก <html> จะต้องเป็น <head>...</head> ซึ่งภายในแท็ก <head> จะใส่แท็กที่เป็นชื่อเรื่อง ซึ่งเป็นคำสั่งย่อยอีก 1 คำสั่ง คือ <title>...</title>
3. ต่อจาก <head>...</head> แล้วจะเป็นส่วนของ <body>...</body> เราจะเขียนสิ่งที่เราต้องการจะนำเสนอลงไประหว่างแท็ก <body>...</body> นี้

รูปแบบโครงสร้างพื้นฐานของไฟล์เอชทีเอ็มแอล

```
<html>
  <head>
    <title> ชื่อเรื่อง </title>
  </head>
  <body>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื้อหาที่จะแสดงผล

</body>

</html>

2.8.1 คำสั่งเริ่มต้นที่ใช้ในการสร้างโฮมเพจ

คำสั่ง (Tags) ที่ใช้ในภาษาเอชทีเอ็มแอลจะประกอบไปด้วยเครื่องหมายน้อยกว่า “<” ตามด้วยชื่อคำสั่ง และปิดท้ายด้วยเครื่องหมายมากกว่า “>” เป็นส่วนที่ทำหน้าที่แสดงผลข้อมูลต่างๆพื้นฐานหลัก โดยทั่วไปคำสั่งของเอชทีเอ็มแอลส่วนใหญ่จะเป็นคู่ จะมีแค่บางคำสั่งที่มีรูปแบบอยู่เพียงตัวเดียว โดยตัวเปิดของคำสั่งจะอยู่ในเครื่องหมาย “<>” และตัวปิดคำสั่งจะอยู่ในเครื่องหมาย “</>” อย่างข้างหน้าตัวอย่างคำสั่ง

1. หมายเหตุ

รูปแบบคำสั่ง <!...>

การใช้คำสั่งหมายเหตุในโปรแกรมภาษาเอชทีเอ็มแอล มีประโยชน์สำหรับแจ้งให้ผู้อ่านได้ทราบว่าเรากำลังทำอะไรอยู่ หรือรายละเอียดต่างๆที่เราต้องการจะบอกเกี่ยวกับโปรแกรมของเรา ซึ่งข้อความที่เขียนจะปรากฏอยู่แต่เฉพาะภายในโปรแกรมเอชทีเอ็มแอล จะไม่ปรากฏบนจอภาพโปรแกรมเว็บเบราว์เซอร์

2. คำสั่งเริ่มต้น

รูปแบบคำสั่ง <html>...</html>

คำสั่ง <html> เป็นคำสั่งที่เริ่มต้นในการเขียนโปรแกรมเอชทีเอ็มแอล และคำสั่ง </html> จะเป็นคำสั่งแสดงว่าสิ้นสุดโปรแกรมเอชทีเอ็มแอล

3. ส่วนหัวของโปรแกรม

รูปแบบคำสั่ง <head>...</head>

คำสั่ง <head> เป็นคำสั่งที่ใช้กำหนดข้อความในส่วนที่เป็นชื่อเรื่องของไฟล์เอชทีเอ็มแอล และภายในคำสั่งจะมีคำสั่งย่อยอีกคำสั่งหนึ่งคือ คำสั่ง <title> คำสั่งปิดคือ </title> และ </head>

4. กำหนดข้อความในส่วนหัวของโปรแกรมหรือไตเติลบาร์

รูปแบบคำสั่ง <title>...</title>

คำสั่งข้างต้นจะเป็นส่วนที่แสดงชื่อของเอกสาร หรือชื่อของไฟล์เอชทีเอ็มแอล ซึ่งข้อความภายในคำสั่งจะปรากฏหรือแสดงผลในส่วนของไตเติลบาร์ (Title Bar) ของโปรแกรมเว็บเบราว์เซอร์ แต่จะไม่แสดงในส่วนของการแสดงผลของโปรแกรมเว็บเบราว์เซอร์

5. ส่วนเนื้อหาของโปรแกรม

รูปแบบคำสั่ง <body>...</body>

คำสั่งในส่วนเนื้อหาของโปรแกรมนี้ เป็นคำสั่งที่สำคัญในส่วนการแสดงผลในเว็บเบราว์เซอร์ ซึ่งอาจจะประกอบไปด้วย ตัวอักษรและรูปภาพการฟิกต่างๆที่อยู่ในเว็บเพจของเรา ตัวอย่างโปรแกรม

<html>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<! Comment>

<head>

<title> test </title>

</head>

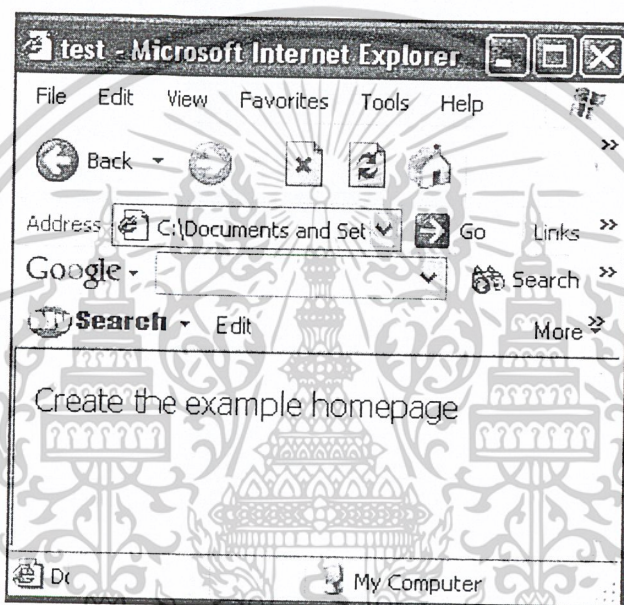
<body>

Create the example homepage

</body>

</html>

```



รูปที่ 2.3 ตัวอย่างหน้าจอแสดงผลภาษาHTML

2.9 ซีจีไอ (CGI : Common Gateway Interface)

เป็นวิธีมาตรฐานสำหรับการเขียนโปรแกรม ที่สามารถโต้ตอบและส่งข้อมูลไปมาระหว่างบราวเซอร์ เช่น เน็ตสเคป ดังนั้นผู้พัฒนาโฮมเพจสามารถดึงข้อมูลใหม่ๆ จากตัวเว็บเซิร์ฟเวอร์มาแสดงที่ตัวโฮมเพจได้ ในกรณีที่ใช้บราวเซอร์ และต้องการข้อมูลเหล่านั้นได้อย่างทันทีทันใด ตัวอย่างเช่น ถ้าต้องการจะค้นหาข้อมูลที่ www.yahoo.com ก็สามารถพิมพ์คำที่ต้องการหาเข้าไปในช่องว่างจากนั้นกดปุ่ม "search" ซึ่งการกดปุ่มนี้นั้น ตัวบราวเซอร์ก็จะส่งข้อมูลที่พิมพ์เข้าไปยังเซิร์ฟเวอร์ (Server) ที่ตัวเซิร์ฟเวอร์ก็จะเรียกโปรแกรมซีจีไอแบบใดแบบหนึ่งขึ้นมารับค่าที่พิมพ์เข้าไปเป็นพารามิเตอร์ จากนั้นก็ค้นหาในฐานข้อมูล เช่น ออราเคิล (Oracle) เมื่อเจอข้อมูลแล้วก็จะส่งกลับมายังตัวบราวเซอร์

โปรแกรมซีจีไอสามารถเขียนด้วยภาษาซี (C), ภาษาเพิร์ล (perl), ยูนิกซ์ (UNIX script) และภาษาจาวา (JAVA) แต่ข้อแม้ก็คือตัวเซิร์ฟเวอร์จะต้องมีความสามารถในการเรียกโปรแกรมมาทำงานได้ การใช้งานตัวซีจีไอนั้นมีอยู่หลากหลาย เช่น การค้นหาข้อมูลต่างๆ ให้ผู้ใช้ลงชื่อในสมุดลงชื่อ การดึงข้อมูลจากฐานข้อมูลต่างๆ และการใช้งานในลักษณะโต้ตอบกับผู้ใช้ ดังนั้นถ้าจะสรุปรายละเอียดของตัวซีจีไอโปรแกรมก็คือเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนักผู้จัดทำเนื้อหาไปใช้ประโยชน์ในการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

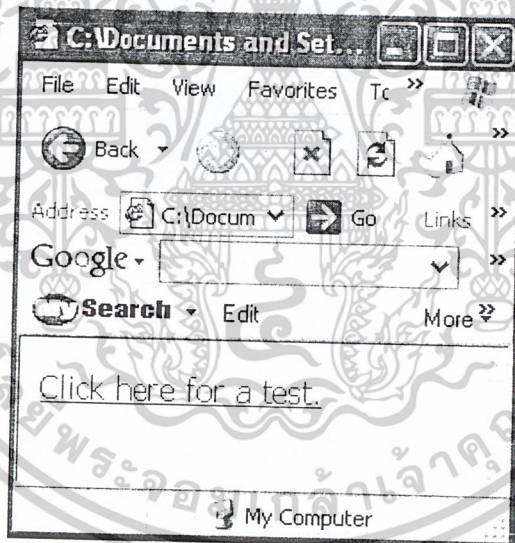
โปรแกรมที่รับข้อมูลเข้ามาโดยเฉพาะจากบราวเซอร์ จากนั้นก็ประมวลผลข้อมูลในแบบใดแบบหนึ่งและเมื่อได้ผลลัพธ์แล้วก็ส่งกลับไปยังตัวบราวเซอร์

ข้อมูลส่วนใหญ่ที่ส่งเข้าไปในซีจีไอ โปรแกรมจะส่งผ่านตัวแปรแวดล้อม (Environment variables) ส่วนข้อมูลที่เหลือสามารถใช้ความสามารถในเรื่องอินพุทเอาต์พุท (Input-Output) ของโปรแกรมเพื่ออ่านข้อมูลเหล่านั้นได้ ส่วนการประมวลผลข้อมูลนั้นก็อาจจะง่ายๆ เช่น ใส่ข้อความต่อท้ายที่ข้อมูลหรืออาจจะยุ่งยาก เช่นการคิวรี (Query) ฐานข้อมูลกลับมายังบราวเซอร์ได้หลายแบบ ดังนั้นตัวซีจีไอโปรแกรมจึงต้องพิมพ์ข้อความที่บรรทัดแรกของเอาต์พุทว่าข้อมูลประเภทใดที่มันผลิตออกมาโดยทั่วไปแล้วซีจีไอโปรแกรมจะผลิตเอชทีเอ็มแอลหรือโฮมเพจนั่นเอง ซึ่งข้อความบรรทัดแรกที่ส่งกลับมายังบราวเซอร์ก็จะเป็น “content-type : text/html”

การเขียนโปรแกรมซีจีไออย่างง่าย

ตัวอย่างต่อไปนี้จะเป็นการเขียนโปรแกรมซีจีไออย่างง่าย ขั้นแรกเราต้องเขียนโฮมเพจที่ไปเรียกโปรแกรมที่อยู่ในไดเรกทอรี cgi-bin

```
<html>
<a href="http://edison/cgi-bin/test1">Click here for a test. </a>
</html>
```



รูปที่ 2.4 ตัวอย่างหน้าจอแสดงการใช้งานCGI

เมื่อผู้ใช้คลิกเมาส์ที่ “Click here for test” ข้อมูลก็จะวิ่งจากตัวบราวเซอร์ไปยังเซิร์ฟเวอร์ซึ่งจะไปเรียกโปรแกรมซีจีไอชื่อ “test1” ขึ้นมาทำงาน ตัว “test1” จะเป็นซีจีไอโปรแกรมที่เราจะต้องเขียนขึ้นซึ่งทำหน้าที่ของโปรแกรม เพียงแต่พิมพ์ข้อความ “This is a test” จากนั้นก็ส่งข้อมูลกลับไปยังบราวเซอร์ สิ่งแรกที่โปรแกรมซีจีไอต้องพิมพ์ออกไปคือบอกว่าเอาต์พุทจะเป็นเอชทีเอ็มแอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยามให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10 ภาษา Script

ถ้าฟังเพียงแค่ HTML เว็บไซต์ก็จะเป็นเหมือนสมุดภาพเล่มหนึ่ง ที่ไม่มีการตอบโต้อะไรกลับมา ผู้ดูแลโฮมเพจจำเป็นต้องหาอะไรแปลกๆใหม่ๆมาลงเว็บไซต์เสมอ ไม่ว่าจะเป็นตัวนับ (Counter) ที่จะทำให้เรารู้ว่ามีคนเข้ามาเยี่ยมชมโฮมเพจของเราประมาณเท่าไรแล้ว หรือในโฮมเพจอาจมีเว็บบอร์ดสำหรับผู้ที่เข้ามาเยี่ยมชม

ซึ่งสิ่งต่างๆเหล่านี้ HTML ธรรมดาทำไม่ได้ เราจะต้องพึ่งโปรแกรมที่เรียกว่า ภาษาสคริปต์ (Scripting Language) ซึ่งในการเขียนภาษา Script สำหรับเว็บไซต์นั้น สามารถเขียนได้หลายภาษา และจะมีรูปแบบการเขียนอยู่ด้วยกันสองแบบ คือ

1. Client-Side Scripting เป็นการเขียนโปรแกรมภาษา Script ให้ทำงานบน Web browser โดยการเขียนโปรแกรมแทรกหรือฝัง (Embed) เข้าไปเป็นส่วนหนึ่งของเอกสาร HTML ภาษา Script ประเภทนี้ได้แก่ JavaScript, VBScript เป็นต้น
 2. Server-Side Scripting เป็นการเขียนโปรแกรมภาษา Script ที่ทำงานอยู่บนตัว Web server โดย Web browser จะเป็นแค่ตัวที่แสดงผลการทำงานเท่านั้น โปรแกรมที่ทำงานบน Web server เหล่านี้ เช่น ASP, ซึ่งเราสามารถเขียนด้วยภาษาต่างๆ เช่น JScript, VBScript เป็นต้น
- ซึ่งความแตกต่างระหว่าง Client-Side และ Server-Side ก็คือ Client-Side จะทำงานบน Web browser และเป็น Script ที่ฝังอยู่ใน HTML เลย

2.11 ASP (Active Server Pages)

เป็นรูปแบบการสร้างสรรค์เว็บรูปแบบหนึ่ง ที่นิยมใช้กันมากในการสื่อสารทางอินเทอร์เน็ต ซึ่งความสามารถของ ASP นั้นทำให้เว็บไม่เว็บไม่เพียงหน้าเอกสารให้เรารออ่านเท่านั้น แต่ยังสามารถเพิ่มลูกเล่นต่างๆ มากมาย เช่น การ Search หาข้อมูล, การคำนวณค่าใช้จ่ายในการซื้อขายผ่านอินเทอร์เน็ต, การให้บริการต่างเช่นเมลฟรี และอื่นๆ อีกมากมายไม่จำกัดซึ่งขึ้นอยู่กับความคิดสร้างสรรค์ต่างๆ ของเรา

2.11.1 หลักการทำงานของ ASP

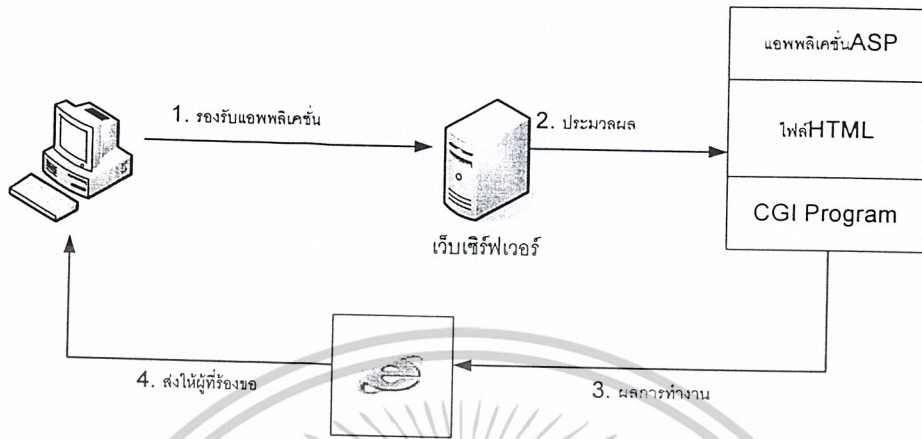
ASP จะทำงานอยู่บนเซิร์ฟเวอร์และการทำงานร่วมกับโปรแกรม Web Server จะทำหน้าที่ประมวลผลข้อมูลที่ได้จากผู้เข้ามาเยี่ยมชม และแสดงผลออกมาทางโฮมเพจ

ยกตัวอย่างการใช้งาน ASP ที่เห็นได้ชัดๆ เช่น เว็บไซต์ Yahoo (www.Yahoo.com) ซึ่งคงไม่มีใครที่ไม่รู้จัก เพราะเป็นเว็บไซต์ที่ใช้ในการค้นหาข้อมูลที่เก่าแก่ตัวหนึ่ง (ตั้งแต่สมัยที่อินเทอร์เน็ตมีอยู่บนระบบ UNIX) ที่เว็บไซต์ Yahoo จะมีช่องรับข้อความอยู่ช่องหนึ่ง ถ้าเราต้องการค้นหาอะไรเราก็พิมพ์ลงไป ในช่องนั้น และกดปุ่ม Search สักพัก Yahoo ก็จะแสดงรายชื่อของเว็บไซต์ที่ใช้จากการค้นหาออกมาให้

เรามาดูการทำงานของเว็บไซต์ Yahoo กันคร่าวๆ เพื่อให้รู้จัก ASP ทำงานอย่างไร จริงๆ แล้วเว็บไซต์จะมีการค้นหา และการแสดงผลที่ซับซ้อนกว่านี้ แต่ยกมาให้ดูเป็นบางส่วนเท่านั้น

ในเว็บไซต์ Yahoo นั้นจะมี ASP อยู่ตัวหนึ่ง และมีฐานข้อมูลอยู่ เมื่อเราเปิด Browser ไปที่เว็บไซต์ Yahoo และพิมพ์สิ่งที่ต้องการค้นหา เช่นคำว่า Zebra ลงในช่องรับข้อความและกดปุ่ม Search เพื่อ Submit (ตอบรับฟอร์ม) ข้อมูลที่เราพิมพ์จะถูกส่งไปกับ ASP ที่ทำงานบนเซิร์ฟเวอร์นั่นเอง เมื่อ ASP ที่อยู่เอกสารนี้เป็นเอกสารที่ส่งวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นับญาติให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บนเซิร์ฟเวอร์ได้รับการ Submit จากนั้น มันก็จะรู้ว่า เราได้ส่งคำว่า Zebra มาให้ จากนั้นตัว ASP จะเอาคำว่า Zebra นี้ไปเปรียบเทียบกับฐานข้อมูลของ Zebra อยู่ในฐานข้อมูลหรือเปล่า ถ้ามีให้ส่งผลลัพธ์มาปรากฏที่ Browser ของเรา

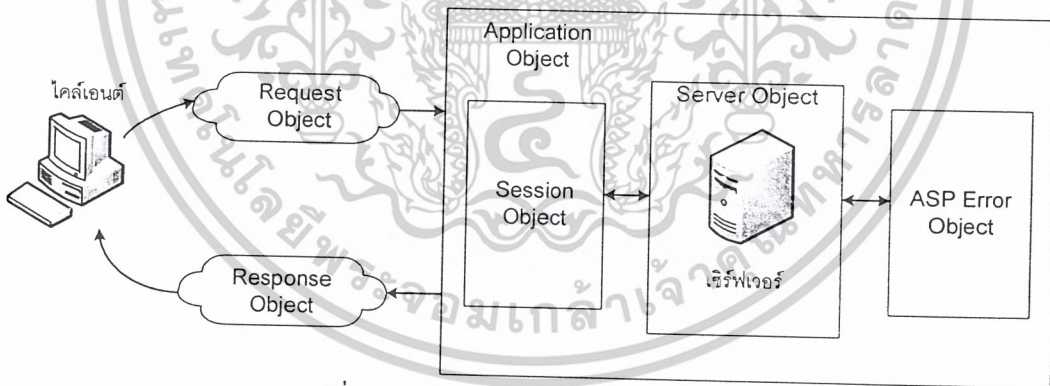


รูปที่ 2.5 หลักการทำงานของฝั่ง เซิร์ฟเวอร์แอปพลิเคชัน

2.11.2 ออบเจกต์พื้นฐานของ ASP

Active Server Pages นั้นมีออบเจกต์พื้นฐานให้ใช้งานกันอยู่ทั้งหมด 6 ชนิด ได้แก่ Request Object, Response Object, Session Object, Application Object, Server Object และ ASPError Object ซึ่งทั้ง 6 ชนิดเราสามารถเขียนสคริปต์เพื่อควบคุม และเรียกใช้งานได้ทันที

ถ้าหากมีความสัมพันธ์ของออบเจกต์ต่าง ๆ นั้นมีดังนี้



รูปที่ 2.6 ความสัมพันธ์ของออบเจกต์ต่างๆ

จากรูปจะเห็นว่า Request Object และ Response Object ทำหน้าที่ติดต่อกับผู้ใช้งานเข้ามา ส่วน Response Object ทำหน้าที่ส่งข้อมูลจากเว็บเซิร์ฟเวอร์กลับไปให้ผู้ใช้งานซึ่งทั้ง 2 ชนิดเป็นออบเจกต์ที่สื่อสารกับผู้ใช้งานโดยตรง

ออบเจกต์ที่อยู่ถัดมาเป็น Session Object ซึ่งทำหน้าที่เก็บข้อมูลของผู้ใช้งานแต่ละรายที่เข้ามาใช้แอปพลิเคชัน ASP นั้นไว้โดยสร้าง Session เป็นจำนวนเท่ากับจำนวนผู้ใช้งานที่เรียกใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับ Application Object ทำหน้าที่ติดต่อ ควบคุมการทำงานของแอปพลิเคชัน ASP และ Server Object ก็ทำหน้าที่ติดต่อระหว่างแอปพลิเคชัน ASP กับเว็บเซิร์ฟเวอร์

ส่วน ASPError Object ทำหน้าที่เก็บรายละเอียดการทำงานของแอปพลิเคชัน ASP ที่ผิดพลาดไว้ให้เรา โดยเราจะตรวจสอบรายละเอียดความผิดพลาดของแอปพลิเคชัน ASP ที่เกิดขึ้น ได้จากรายละเอียดต่างๆที่เก็บอยู่ภายใน แล้วแก้ไขให้การทำงานต่างๆเป็นไปอย่างถูกต้อง

ในบางครั้งเราจะต้องเรียกออบเจกต์พื้นฐานของ ASP ว่า Built-in Object ซึ่งพร้อมใช้งานได้ทันที

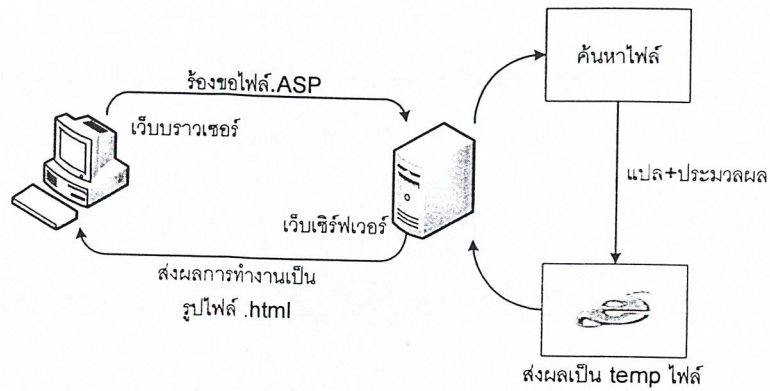
ออบเจกต์	คำอธิบาย
Request	ออบเจกต์ที่ใช้จัดการข้อมูลที่รับเข้ามาจากผู้ใช้งาน
Response	ออบเจกต์ที่ใช้จัดการข้อมูลที่ส่งกลับไปยังผู้ใช้งาน
Session	ออบเจกต์ที่เก็บรายละเอียดของผู้ใช้งานแอปพลิเคชัน ASP แต่ละคน
Application	ออบเจกต์ที่ใช้แลกเปลี่ยนข้อมูลกับแอปพลิเคชัน ASP
Server	ออบเจกต์ที่ทำหน้าที่จัดการและบริหารทรัพยากรของเซิร์ฟเวอร์
ASPErr	ออบเจกต์ที่ทำหน้าที่จัดการเมื่อเกิดความผิดพลาดจากการทำงานของแอปพลิเคชัน ASP

ตารางที่ 2.1 ออบเจกต์พื้นฐานของ ASP

นอกเหนือจากออบเจกต์พื้นฐานทั้ง 6 ชนิดแล้ว บางครั้งเราอาจต้องการออบเจกต์ที่มีคุณสมบัติพิเศษเพื่อเพิ่มเติมให้กับแอปพลิเคชัน หรือออบเจกต์ที่ใช้จัดการกับเว็บเซิร์ฟเวอร์ เราก็สามารถทำได้โดยมี 3 ทางเลือกคือ

1. Active Server Object - เป็นชุดออบเจกต์ที่เว็บเซิร์ฟเวอร์เตรียมไว้ให้ใช้นอกเหนือจากออบเจกต์พื้นฐาน 6 ชนิดที่เราได้เรียนรู้ไปแล้ว โดยออบเจกต์แต่ละตัวจะมีความสามารถในการทำงานเฉพาะด้านที่แตกต่างกัน
2. Third Party Component - เป็นออบเจกต์ที่มีบริษัทซอฟต์แวร์อื่นๆสร้างไว้ให้ โดยมีทั้งของฟรี และที่ต้องเสียเงินซื้อ
3. ออบเจกต์ที่สร้างขึ้นเอง - ซึ่งเราสามารถเขียนด้วย Visual Basic หรือ Visual C++ เขียนคอมโปเนนต์มาให้ใช้งานเฉพาะก็ได้ ซึ่งคอมโปเนนต์เหล่านั้นจะอยู่ในรูปของไฟล์ .dll ที่ทำงานบนเซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 หลักการทำงานของแอปพลิเคชัน ASP

ตอนนี้เราคงพอเข้าใจการทำงานของ ASP บ้างแล้ว จะเห็นได้ว่า ASP จะทำงานอยู่บน Server เท่านั้น มีเพียงแต่ผลลัพธ์เท่านั้นที่เราส่งกลับมาที่ Browser ของเรา ด้วยการทำงานเช่นนี้ทำให้เราสามารถสรุปผลดีของการใช้ ASP ได้ดังนี้

2.10.3 ข้อดีของ ASP

1. ASP ทำให้เว็บของคุณเป็นเว็บของไดนามิก (Dynamic) นั่นคือรูปแบบของเว็บที่แสดงผลออกมานั้นสามารถเปลี่ยนแปลงได้ ตามข้อมูลที่ ASP ได้รับ เช่นตัวอย่างการ Search ข้อมูลในเว็บไซด์ Yahoo ที่ผลลัพธ์จะเปลี่ยนไปตามคำที่เราใช้ Search
2. เพิ่มความเร็วในการดูเว็บ เนื่องจากการดูเว็บนั้นเรามักจะสูญเสียเวลาส่วนใหญ่มากกับการรอข้อมูลที่มาจากอินเทอร์เน็ต ยิ่งข้อมูลยิ่งมากยิ่งรอนาน ซึ่ง ASP สามารถช่วยในจุดนี้ได้ กล่าวคือ ASP จะทำการคำนวณต่างๆ จนเสร็จและส่งเฉพาะผลลัพธ์ที่เราต้องการเท่านั้น ทำให้ปริมาณการส่งข้อมูลน้อยลง เราก็จะเสียเวลารอข้อมูลน้อยลงและสามารถดูเว็บได้เร็วขึ้น
3. เพิ่มความปลอดภัยให้ระบบของเรา ในการเขียนโปรแกรมต่างๆ บางครั้งเราก็อ้างถึงไคเร็กทอรีที่เก็บฐานข้อมูล อย่างเช่นเว็บไซด์ Yahoo เป็นต้น ซึ่งการใช้ ASP ไคเร็กทอรีต่างๆ จะไม่ถูกแสดงที่ฝั่งผู้ดูเว็บ จะแสดงเฉพาะผลลัพธ์ที่เอามาจากฐานข้อมูลเท่านั้น ทำให้ผู้ดูเว็บไม่สามารถรู้ถึงโครงสร้างของเว็บเราได้ง่าย และป้องกันผู้ไม่หวังดีมาเจาะระบบของเราได้
4. ลดปัญหาความสามารถของเครื่องที่ใช้เว็บ เนื่องจาก ASP จะส่งเฉพาะผลลัพธ์สุดท้ายมาแสดงผลเท่านั้น ดังนั้นไม่ว่าเครื่องของคุณจะทันสมัยหรือล้าสมัยเพียงใด ก็ไม่ทำให้เวลาที่ใช้เปิดดูเว็บแตกต่างกันมากนัก เพราะว่าการประมวลผลทั้งหมดเสร็จสิ้นที่ฝั่ง Server แล้ว

โดยปกติแล้วในการใช้งาน ASP จะต้องทำการติดตั้งโปรแกรม IIS (Internet Information Services) ซึ่งเป็นเครื่องมือของ Microsoft (window2000, windowNT, windoXP Professional) ที่มีความสามารถในการจัดการกับเทคโนโลยีสำหรับแอปพลิเคชันฐานข้อมูลบนเว็บ ให้ทำได้รวดเร็วและง่ายคายด้วยการใช้โค้ดเพียงเล็กน้อย และจะใช้ในการจำลองตัวเองให้เป็นเซิร์ฟเวอร์เพื่อที่จะรัน ASP เนื่องจาก ASP เป็น script ที่รันบนฝั่งเซิร์ฟเวอร์ โดยจะสามารถทำการติดตั้ง IIS ได้ดังนี้

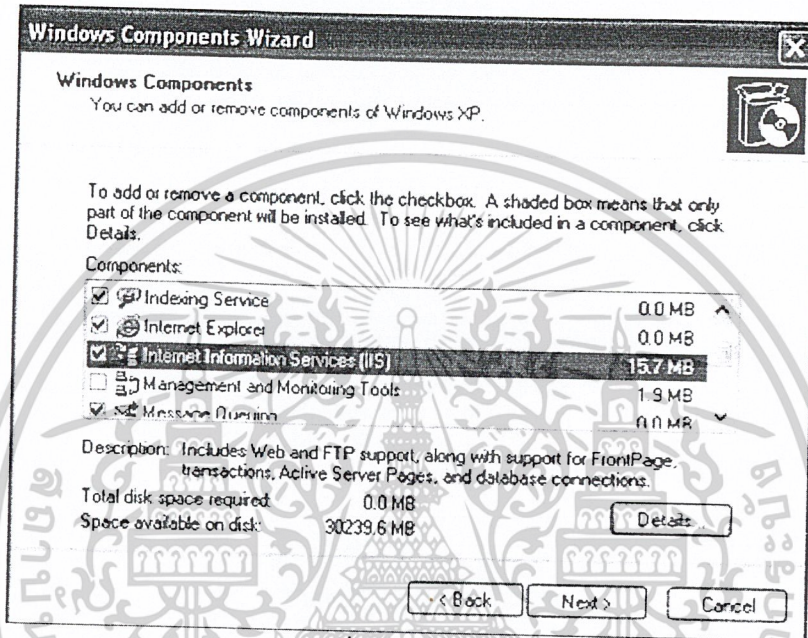
1. ให้เข้าไปที่ Start>Control Panel>Add or Remove Programs

2. คลิกเมนูด้านซ้ายมือตรง Add/Remove Windows Components

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เครื่องหมายถูกหน้า Internet Information Services(IIS)
4. คลิกดู Detail ที่ต้องการ เลือก Components ที่ต้องการ หรือทุกตัว
5. หากเครื่องยังไม่เคยติดตั้ง IIS มันจะเรียกหาแผ่น Windows XP เพื่อติดตั้ง Components ที่เลือกไว้
6. คลิก OK เสร็จการติดตั้ง
7. รีสตาร์ทเครื่องเพื่อความสมบูรณ์ของระบบ



รูปที่ 2.8 ตัวอย่างการลง IIS

2.11 ซ็อกเกต (Socket)

ซ็อกเกตถูกกำหนดหรือนิยามไว้ว่า เป็นคู่ของการสื่อสาร หรือคู่ของโพรเซส (หรือเซรค) โดยการสื่อสารบนเน็ตเวิร์คใช้คู่ของซ็อกเกตสำหรับแต่ละโพรเซส

สำหรับซ็อกเกตประกอบไปด้วย IP Address กับหมายเลข Port (Port Number)

โดยทั่วไป ซ็อกเกตใช้สถาปัตยกรรมไคลเอนต์เซิร์ฟเวอร์ เซิร์ฟเวอร์จะรอการเข้ามาตามการขอร้องของไคลเอนต์โดยการฟังที่ Port เฉพาะ เมื่อการขอร้องได้รับ เซิร์ฟเวอร์ก็จะยอมรับการต่อกรซ็อกเกตไคลเอนต์เพื่อให้สมบูรณ์ในการเชื่อมต่อ

เซิร์ฟเวอร์ที่สร้างการบริการเฉพาะ เช่น telnet, ftp, mail และ http จะฟัง (listen) ที่พอร์ตมีชื่อ เช่น เซิร์ฟเวอร์ telnet จะฟังที่พอร์ต 23, เซิร์ฟเวอร์ ftp จะฟังที่พอร์ต 21 หรือเซิร์ฟเวอร์ http จะฟังที่พอร์ต 80 เป็นต้น

หมายเลขพอร์ตทั้งหมดที่ต่ำกว่า 1,024 จะถูกพิจารณาว่าเป็นพอร์ตที่มีชื่อเสียง เราสามารถใช้พอร์ตเหล่านี้เพื่อสร้างการบริการตามมาตรฐานได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.11.1 ตัวอย่างการสื่อสารด้วย Socket

เมื่อเรดโคลเอนต์เริ่มต้นการขอร้องสำหรับการเชื่อมต่อ จะถูกกำหนดพอร์ตโดยโฮสต์คอมพิวเตอร์ (Host computer) พอร์ตนี้เป็นหมายเลขใดๆ ก็ได้ที่มากกว่า 1,024 ตัวอย่างเช่น ถ้าโคลเอนต์บนโฮสต์ A มี IP Address 2 ต้องการที่จะสร้างการเชื่อมต่อกับเซิร์ฟเวอร์ http (ซึ่งฟังที่พอร์ต 80) ที่มี IP Address 1 โฮสต์ A จะถูกกำหนดพอร์ต 2,544 และที่ฝั่งเซิร์ฟเวอร์จะเป็นพอร์ต 80 สถานการณ์สามารถแสดงได้

การเชื่อมต่อทั้งหมดเป็นคุณสมบัติเฉพาะ ดังนั้นถ้าโปรเซสอื่นๆ บนโฮสต์ A ต้องการสร้างการเชื่อมต่ออื่นๆ กับเซิร์ฟเวอร์ http เดียวกัน เซิร์ฟเวอร์จะกำหนดหมายเลขพอร์ตที่มากกว่า 1,024 และต้องไม่เท่ากับพอร์ต 2,544 (เนื่องจากถูกใช้ไปแล้ว) การทำอย่างนี้ เพื่อให้แน่ใจว่าการเชื่อมต่อทั้งหมดประกอบด้วยคู่ที่เป็นยูนิค (Unique) ของซ็อกเกตหรือเป็นสิ่งที่ไม่ซ้ำกับการเชื่อมต่ออื่นๆ ของซ็อกเกต

โดยปกติแล้วเซิร์ฟเวอร์จะมีหลายๆ การขอร้องที่พร้อมกัน จะต้องใช้ระยะเวลาหนึ่งที่โคลเอนต์ต้องรอคอยเพื่อที่จะถูกบริการโดยเซิร์ฟเวอร์เซรคเดียว ซึ่งจะไม่สามารถรับได้

เพื่อแก้ไขสถานการณ์นี้เซิร์ฟเวอร์ต้องจัดการการขอร้องพร้อมๆ กัน โดยการกำหนดเซรคแยกออกมาเพื่อบริการแต่ละการขอร้องที่เข้ามา ตัวอย่างเช่น เซิร์ฟเวอร์ http ไม่ว่าจะกำหนดเซรคแยกออกมาเพื่อบริการแต่ละการขอร้องสำหรับเว็บเพจ

2.11.2 ชนิดของซ็อกเกต

ชนิดของซ็อกเกตมีอยู่ 3 ชนิด คือ

1. Connection-Oriented Socket
2. Connectionless Socket
3. Raw Socket

Connection-Oriented Socket เป็นซ็อกเกตการเชื่อมต่อแบบต่อเนื่องที่อนุญาตให้โปรเซสเชื่อมต่อกับโปรเซสระยะไกล (Remote) ซึ่งใช้โปรโตคอล TCP (Transmission Control Protocol) ดังนั้นด้วยวิธีการนี้ทำให้ข้อมูลเชื่อถือได้ เมื่อการเชื่อมต่อได้เกิดขึ้นโปรเซสก็จะมีการส่งข้อมูลกลับไปจนกระทั่งฝั่งใดฝั่งหนึ่งหรืออื่นๆมีการปิดการเชื่อมต่อ ชนิดของซ็อกเกตนี้บางที่เราเรียกว่า สตรีมซ็อกเกต (Stream Socket) ทั้ง ftp และ http ก็ใช้แพ็คเกจนี้ในการสื่อสาร

Connectionless Socket หรือเรียกอีกอย่างว่า ดาต้าแกรม เป็นซ็อกเกตแบบไม่ต่อเนื่อง และนำมาใช้เป็นประโยชน์ในการส่งเมสเสจสั้นๆซึ่งไม่สามารถสนับสนุนส่วนหัว ดังนั้นจึงพิจารณาการเชื่อมต่อประเภทนี้เป็นแบบเชื่อถือไม่ได้ ซึ่งก็คือการไม่รับประกันข้อมูลที่ถูส่งออกไปไม่เหมือนกับซ็อกเกตการเชื่อมต่อแบบต่อเนื่องที่ซ็อกเกตปลายทางถูกตรวจสอบเมื่อแพ็คเกจถูกส่งออกไป

ซ็อกเกตแบบไม่ต่อเนื่องเปรียบเสมือนกับการบริการของไปรษณีย์ที่ผู้ส่งจดหมายไปตามที่อยู่แล้วใส่ในกล่องรับจดหมายโดยผู้ส่งจะไม่รู้ว่าผู้รับได้รับจดหมายหรือไม่ ซ็อกเกตแบบนี้นิยมใช้กันในเซิร์ฟเวอร์ DSN ที่ใช้ซ็อกเกตดาต้าแกรมในการตอบสนองต่อการขอร้องที่เข้ามาหลายๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกจากนี้จะใช้คำสั่งแกรมซ็อกเกตในการกระจาย (Broadcast) เมสเสจ หรือ Multicast เพื่อไปยังปลายทางหลายๆแห่งพร้อมกัน ซึ่งเหมือนกับการกระจายเสียงวิทยุหรือโทรทัศน์

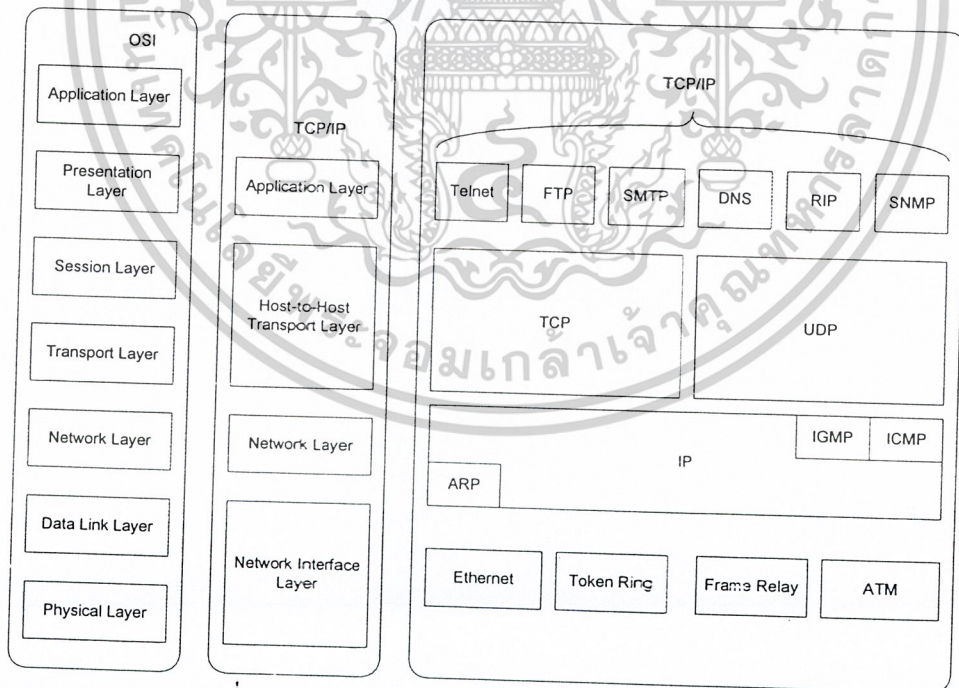
Raw Socket เป็นซ็อกเกตที่อนุญาตให้การเข้าถึงโปรโตคอล Transport Raw Socket ยังสามารถนำมาใช้เพื่อจัดการกับข้อมูลส่วนหัว ไอพี (IP header) นอกจากนี้แล้วการใช้ซ็อกเกตชนิดนี้ ต้องการความรู้อย่างมากของโครงสร้างโปรโตคอลพื้นฐาน

2.12 สถาปัตยกรรมโปรโตคอล TCP/IP

โปรโตคอล (Protocol) คือระเบียบ กฎ และข้อกำหนดต่างๆรวมถึงมาตรฐานที่ใช้ เพื่อให้ตัวรับและตัวส่งสามารถดำเนินการติดต่อสื่อสารได้สำเร็จ

TCP/IP เป็นชุดโปรโตคอลที่ออกแบบมาสำหรับการเชื่อมโยงระหว่างเน็ตเวิร์กขนาดใหญ่เชื่อมต่อกับเครือข่าย WAN (Wide Area Network) TCP/IP พัฒนาขึ้นในปี ค.ศ. 1969 โดยหน่วยงานของรัฐบาลอเมริกา มีชื่อว่า Defense Advanced Research Projects Agency หรือ DARPA โดยมีจุดประสงค์คือ เพื่อให้การเชื่อมโยงเน็ตเวิร์กการติดต่อสื่อสารที่ความเร็วสูง

ปกติแล้วเราทราบว่าตามองค์กรมมาตรฐานสากล(OSI) จะแบ่งโมเดลโปรโตคอลการสื่อสารออกเป็น 7 เลเยอร์ คือ Application, Presentation, Session, Transport, Network, Data Link และ Physical แต่โปรโตคอล TCP/IP สามารถแม้เป็นโมเดลในแนวความคิดไว้ 4 เลเยอร์ซึ่งรู้กันว่าโมเดล DARPA จะประกอบด้วยเลเยอร์ Application, Transport, Internet และ Network Interface



รูปที่ 2.9 การเปรียบเทียบของเลเยอร์ OSI และ TCP/IP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.13 การทำงานของแต่ละเลเยอร์ใน TCP/IP

2.13.1 Network Interface Layer หรือเรียกอีกอย่างหนึ่งว่า Network Access Layer มีหน้าที่รับผิดชอบในการวางแพ็กเกจ TCP/IP บนตัวกลางเน็ตเวิร์ค และการรับแพ็กเกจ TCP/IP ถูกออกแบบเพื่อที่จะเป็นอิสระต่อวิธีการเข้าถึงเน็ตเวิร์ค รูปแบบเฟรม (frame) และตัวกลาง (medium) ด้วยวิธีการทำให้ TCP/IP สามารถถูกใช้ในการต่อกับเน็ตเวิร์คชนิดที่แตกต่างกันได้ รวมทั้งเทคโนโลยี LAN (Local Area Network) เช่น Ethernet หรือ Token Ring และเทคโนโลยี WAN เช่น X.25 หรือ Frame Relay เนื่องจากการเป็นอิสระกับเทคโนโลยี มีผลให้ TCP/IP สามารถที่จะนำมาดัดแปลงใช้กับ Asynchronous Transfer Mode (ATM) ได้

2.13.2 Internet Layer มีหน้าที่รับผิดชอบการระบุที่อยู่ การทำแพ็กเก็ต และฟังก์ชันต่างๆ โดยมีโปรโตคอลแกนหลักของ Internet Layer ดังนี้

1. Internet Protocol (IP) ทำหน้าที่ให้บริการส่งผ่านข้อมูลที่มาจาก Host-to-Host layer เพื่อส่งข้ามไปยังเครือข่ายใดๆ ได้อย่างถูกต้อง แม้ว่าจะมีเครือข่ายเชื่อมต่อกันอยู่ในอินเทอร์เน็ตเป็นล้านๆ เครือข่ายก็ตาม เนื่องจากโปรโตคอลไอพีมีข้อมูลตำแหน่งไอพีปลายทางที่จะส่งข้อมูลไปให้โดยทำงานร่วมกับอุปกรณ์ Router เพื่อส่งข้อมูลข้ามเครือข่ายออกไปได้ ตัวโปรโตคอลไอพีจะทำงานแบบแพ็กเกจ switching คือมีการส่งข้อมูลผ่านสวิตช์ (Switch) ไปยังปลายทาง ตัววงจรผ่านหรือ Switch นี้อาจเป็นเกตเวย์ หรือ Router ในระบบเครือข่ายก็ได้ ซึ่งในข้อมูลของโปรโตคอลไอพีจะมีข้อมูลของหมายเลขไอพีปลายทางที่จะส่งข้อมูลไป และเมื่อถึงเครือข่ายปลายทางแล้ว จะมีกลไกแปลงหมายเลขไอพี ให้เป็นหมายเลขฮาร์ดแวร์ประจำเครื่องที่ต้องการอีกทีหนึ่ง ด้วยโปรโตคอล ARP ตามรูป 2.6 ที่จะแสดงการติดต่อกันระหว่างโปรโตคอลในชั้นของ Host-to-Host layer และ Internetwork layer
2. Address Resolution Protocol (ARP) รับผิดชอบการแก้ปัญหาที่อยู่ IP
3. Internet Control Message Protocol (ICMP) หน้าที่หลักของโปรโตคอล ICMP (Internet Control Protocol) คือการแจ้ง หรือแสดงข้อความจากระบบ เพื่อบอกให้ผู้ใช้ทราบว่าเกิดอะไรขึ้นในการส่งผ่านข้อมูลนั้น ซึ่งปัญหาส่วนมากที่พบคือส่งไปไม่ได้ หรือปลายทางรับข้อมูลไม่ได้ เป็นต้น นอกจากนี้โปรโตคอล ICMP ยังถูกเรียกใช้งานจากเครื่องเซิร์ฟเวอร์และ Router อีกด้วย เพื่อแลกเปลี่ยนข้อมูลที่ใช้ควบคุม ส่วนรูปแบบการทำงานของโปรโตคอล ICMP นั้นจะทำความคู่กับโปรโตคอลไอพีในระดับเดียวกัน และข้อความต่าง ๆ ที่แจ้งให้ทราบ จะถูกผนึกอยู่ภายในข้อมูลของไอพีอีกทีหนึ่ง โปรโตคอล ทีซีพี และ ยูดีพี อาศัยโปรโตคอล ที่อยู่ชั้นล่างเพื่อส่งข้อมูลระหว่างเครื่อง และในชั้น Internetwork Protocol ยังมีโปรโตคอล ICMP ทำหน้าที่ส่งข้อความแจ้งเตือนและโปรโตคอล ARP ทำหน้าที่แปลงเลขหมาย ไอพี ไปเป็นเลขหมายของฮาร์ดแวร์จริง
4. Internet Group Management Protocol (IGMP) รับผิดชอบการจัดการของกลุ่มมัลติคาส IP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.13.3 Transport Layer หรือเรียกอีกอย่างว่า Host-to-Host Transport Layer มีหน้าที่รับผิดชอบ สำหรับการให้บริการมีความเชื่อถือได้ และมีประสิทธิภาพ โดยมีโปรโตคอลที่เป็นแกนหลัก คือ โปรโตคอล TCP และ UDP

2.13.3.1 TCP (Transmission Control Protocol) โปรโตคอลที่ซีพีเป็นโปรโตคอลที่มีการรับส่ง ข้อมูลแบบ Stream Oriented Protocol หมายความว่า การรับส่งข้อมูลจะไม่คำนึงถึงปริมาณข้อมูลที่จะ ส่งไป แต่จะแบ่งข้อมูลเป็นส่วนย่อยๆก่อน แล้วจึงส่งไปยังปลายทางต่อเนื่องเป็นลำดับข้อมูล ในกรณีที่ ข้อมูลส่วนใดส่วนหนึ่งสูญหายไปก็จะส่งข้อมูลส่วนนั้นใหม่อีกครั้ง สำหรับปลายทาง ก็จะทำหน้าที่ จัดเรียงส่วนของข้อมูลหน่วยใหม่ให้ต่อเนื่องกันและประกอบกลับเป็นข้อมูลทั้งหมดได้ซึ่งจะแยกข้อมูลที่ ไม่ถูกต้องออกดังนั้นแอปพลิเคชัน หรือโพรเซสใดที่อาศัยการส่งผ่านข้อมูลด้วยโปรโตคอลที่ซีพี จะต้อง ใช้หน่วยความจำ และช่องสัญญาณ (Bandwidth) มากกว่ายูดีพี

การติดต่อระหว่างกัน จะต้องเป็นแบบ Connection – Oriented คือต้องมีการสร้างการติดต่อกัน เป็น Session ทั้ง 2 ด้านเสียก่อน แล้วจึงจะรับส่งข้อมูลไปได้พร้อมกัน (Full duplex) เหมือนกับการใช้ โทรศัพท์ติดต่อกัน เมื่อผู้ติดต่อต้นทางเรียกให้ตรงข้ามรับสายแล้ว จึงเริ่มสนทนา เช่น พูดคำว่า “สวัสดี” หรือ “ฮัลโล” กันก่อนเพื่อให้แน่ใจว่า ฝ่ายตรงข้ามพร้อมจะติดต่อดู้อย่างไร จากนั้นจึงเริ่มสนทนา และ เมื่อต้องการจะเลิกการติดต่อ ก็จะมีการพูดคำว่า “สวัสดี” ให้ฝ่ายตรงข้ามทราบว่า จะเลิกการติดต่อ และวาง สายไป ซึ่งในระหว่างการติดต่อกันนั้น แม้ว่าฝ่ายใดฝ่ายหนึ่งหรือทั้งสองฝ่ายจะเจียบไป ก็ไม่พูดอะไรเป็น เวลานานๆ แต่การเชื่อมโยงระหว่างทั้งสองด้าน ยังคงมีอยู่ไม่ขาดไปจนฝ่ายหนึ่งจะวางสาย เช่นเดียวกับ การติดต่อกันด้วยกลไกโปรโตคอล ที่ซีพี เมื่อแอปพลิเคชันต้องการส่งผ่านข้อมูล จะใช้โปรโตคอลที่ เหมาะสมในชั้น Protocol layer ติดต่อกัน และมีการสร้างช่องส่งข้อมูลผ่าน port ที่กำหนด เพื่อส่งผ่าน ข้อมูลไปยังโปรโตคอล ที่ซีพี

ในระหว่างการรับส่งข้อมูลนี้โปรโตคอลที่ซีพีจะเพิ่มขบวนการตรวจสอบข้อมูล เพื่อให้ข้อมูลมี ความถูกต้องไม่ผิดพลาดไปจากเดิมโดยการส่งสัญญาณตรวจทานข้อมูลและส่งข้อมูลให้ใหม่อีกครั้ง ถ้า ปลายทางไม่ได้รับหรือเกิดความผิดพลาดขึ้น

ความน่าเชื่อถือของการส่งผ่านข้อมูลโดยโปรโตคอลที่ซีพีจะมีมากกว่า แต่ก็ต้องอาศัยทรัพยากร ของระบบมากกว่าในการทำงานเช่นกัน ซึ่งรูปแบบของที่ซีพีที่แพ็คเกจจะเห็นว่ามีฟิลด์ Acknowledgement Number และข้อมูล Checksum เพื่อใช้ตรวจสอบการเดินทางของข้อมูล ส่วนข้อมูลนำทางมีข้อมูลมาก ทำ ให้ต้องอาศัยทรัพยากรของระบบทำงานมาก

2.13.3.2 UDP (User Datagram Protocol) ใน Host-to-Host-layer นอกจากจะมีโปรโตคอลที่ซีพี ทำงานแล้ว ก็ยังมีโปรโตคอลยูดีพี (User Datagram Protocol) ที่มีคุณสมบัติแตกต่างกันอยู่ด้วย ในการ รับส่งข้อมูลผ่าน โปรโตคอลยูดีพี จะเป็นแบบที่ทั้งสองนั้นไม่จำเป็นต้องอาศัย การสร้างช่องทางเชื่อมต่อกันระหว่างเครื่องเซิร์ฟเวอร์ให้บริการ กับเครื่องที่ขอใช้บริการ โดยไม่ต้องแจ้งให้ฝ่ายรับข้อมูลเหมือน โปรโตคอลที่ซีพี และไม่มีการตรวจสอบความถูกต้องครบถ้วนในการรับส่งข้อมูลนั้นๆ ด้วย เนื่องจาก โปรโตคอลยูดีพีไม่มีสัญญาณตรวจทานข้อมูลในการส่งข้อมูลแต่ละครั้ง และไม่มีการส่งข้อมูลใหม่อีกใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กรณีที่เกิดความผิดพลาดของการส่งข้อมูล เมื่อเป็นเช่นนี้แอปพลิเคชัน หรือ โพรเซสที่ต้องอาศัย โพรโทคอลยูติลิตี้ ในการส่งผ่านข้อมูล ก็อาจจะสร้างขบวนการตรวจสอบข้อมูลขึ้นมาเอง

จะเห็นว่าโพรโทคอลชั้นบนขึ้นไป ที่ใช้การส่งผ่านข้อมูลโดยโพรโทคอลยูติลิตี้ เช่น โพรโทคอล SNMP (ใช้ควบคุม และจัดการอุปกรณ์ในเครือข่าย) หรือโพรโทคอล DHCP (ใช้ส่งข้อมูลพารามิเตอร์ของเครือข่าย ให้กับเครื่องลูกข่ายได้ใช้งาน) การส่งข้อมูลเหล่านั้น ไม่ต้องรับทราบหรือตรวจสอบว่าข้อมูลไปถึงปลายทางถูกหรือไม่ แต่กลไกการตรวจสอบข้อมูลที่มีการรับส่ง จะไปทำในขั้นตอนของโพรโทคอลชั้นที่สูงกว่าแทน

ตัวอย่างขั้นตอนกลไกการทำงานโดยใช้โพรโทคอลยูติลิตี้ มีดังต่อไปนี้

- ในขั้นตอนของ Process layer เมื่อโพรโทคอล ควบคุมอุปกรณ์เครือข่ายเช่น โปรแกรม Network management ต้องการส่งข้อมูลไปยังอุปกรณ์ที่ต้องการแอปพลิเคชันนั้นจะติดต่อผ่านโพรโทคอล SNMP ในชั้น Process layer
- โพรโทคอล SNMP จะติดต่อกับโพรโทคอลยูติลิตี้ในชั้นถัดไป เพื่อขอติดต่อผ่าน port ที่กำหนด
- โพรโทคอล SNMP เตรียมข้อมูลที่จะส่งรวมทั้งที่อยู่ปลายทาง
- โพรโทคอล SNMP ส่งผ่านข้อมูลให้โพรโทคอลยูติลิตี้ที่อยู่ในชั้น Host-to-Host layer
- โพรโทคอลยูติลิตี้ทำหน้าที่ผนึกข้อมูลหรือ หน่วยข้อมูลนั้น ไปกับ โพรโทคอลไอพีในชั้นถัดลงไปเพื่อส่งข้อมูลออกจากเครื่อง

ซึ่งจะเห็นว่ามีกลไกที่ต่างจากการส่งข้อมูลด้วยโพรโทคอลที่ซีพีซึ่งจะต้องมีการติดต่อกันก่อนและทั้งสองฝ่ายรับทราบการรับส่งข้อมูลของช่องทางการส่งข้อมูลนั้น ซึ่งรูปแบบของยูติลิตี้ แพ็กเกจ จะมีฟิลด์ข้อมูลส่วนข้อมูลนำทางน้อยมาก และไม่มีข้อมูลส่วนการตรวจสอบข้อมูล ทำให้ยูติลิตี้แพ็กเกจมีขนาดเล็ก และใช้หน่วยความจำ หรือทรัพยากรของระบบน้อย

2.13.4 Application Layer จะทำหน้าที่ให้Application มีความสามารถเข้าถึงการบริการของเลเยอร์อื่นๆ และกำหนดโพรโทคอลที่ application เพื่อแลกเปลี่ยนข้อมูล มีหลายๆโพรโทคอล Application Layer และโพรโทคอลใหม่ที่กำลังอยู่ในระหว่างการพัฒนาโพรโทคอล Application Layer ที่รู้จักกันมาซึ่งถูกใช้ในการแลกเปลี่ยนข้อมูลดังนี้

1. Hypertext Transfer Protocol (HTTP) ใช้สำหรับการโอนย้ายไฟล์ที่ทำขึ้นบนเว็บเพจ
2. File Transfer Protocol (FTP) ใช้สำหรับการโอนย้ายไฟล์แบบโต้ตอบ (interactive)
3. Simple Mail Transfer Protocol (SMTP) ใช้สำหรับการโอนย้ายเมสเสจและไฟล์ที่แนบมา
4. Simple Network Management Protocol (SNMP) ใช้ระหว่างคอนโซลการจัดการเน็ตเวิร์กกับเน็ตเวิร์คคือไวัซ์ต่างๆ เช่น bridges, router และ hub เพื่อรวบรวมและแลกเปลี่ยนข้อมูลการจัดการเน็ตเวิร์ก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.14 ภาษาวิซวลซีพลัสพลัส (Visual C++)

Visual C++ ได้รับการพัฒนาขึ้นมาจาก Microsoft C/C++ ให้เป็น IDE ที่ทำงานบนระบบปฏิบัติการวินโดวส์ได้อย่างเต็มที่ รองรับการพัฒนาโปรแกรมบนวินโดวส์โดยมี MFC (Microsoft Foundation Class) เป็นไลบรารีที่จะช่วยอำนวยความสะดวกในการพัฒนาโปรแกรมบนวินโดวส์

ในตอนนี้อธิบายได้ว่า MFC นั้นเป็นส่วนหนึ่งของ Visual C++ เลยทีเดียวเพราะ Visual C++ เป็นเครื่องมือในการพัฒนาโปรแกรมบนวินโดวส์ที่มี MFC เป็นไลบรารีที่จะช่วยอำนวยความสะดวกแทนการใช้ SDK ซึ่งทำให้ MFC ก็ได้รับการพัฒนาให้มีประสิทธิภาพสูงขึ้นไปพร้อมกับ Visual C++ จนถึงทุกวันนี้

ข้อดีของการเขียนโปรแกรมด้วย Visual C++ และ MFC อีกข้อหนึ่งคือ ความสามารถในการพอร์ตเทเบิล (Portability) หมายความว่า หากเรามีซอร์สโค้ดที่เขียนด้วย Visual C++ ในเวอร์ชันต่ำกว่าได้ เราสามารถนำมาคอมไพล์และลิงค์ใหม่ได้ โดยใช้ Visual C++ หรือที่สูงกว่านี้ได้ และการเขียนโปรแกรมด้วยภาษา C++ นี้จะเป็นการเขียนโปรแกรมแบบโครงสร้าง (Structure) เป็นการรวมข้อมูลที่แตกต่างกันรวมเข้าไว้ด้วยกัน ภายใต้ชื่อเดียวกัน ซึ่งจะช่วยอำนวยความสะดวกในการเขียนโปรแกรมได้ง่ายขึ้นและความถูกต้องมากยิ่งขึ้น ลักษณะการจัดข้อมูลแบบโครงสร้างนี้ มีลักษณะเหมือนกับการจัดข้อมูลแบบ Record ในภาษาปาสคาล และแนวคิดของโครงสร้างข้อมูลนี้ก็พื้นฐานสำคัญในการพัฒนาแนวคิดการเขียนโปรแกรมแบบ OOP (Object Oriented Programming) ซึ่งเป็นแนวคิดในการเขียนโปรแกรมแบบหนึ่ง ที่ผู้รู้หลายๆท่านได้สรุปหรือได้ให้คำนิยามไว้ว่า เป็นการเขียนโปรแกรมเชิงวัตถุที่มีลักษณะที่คล้ายกับธรรมชาติของสิ่งของสิ่งหนึ่งซึ่งเราสามารถแบ่งแยกสิ่งต่างๆ ออกเป็นประเภทๆได้ ถ้าเราได้นำเอาแนวคิดของ OOP มาใช้ในการเขียนโปรแกรมและการจัดการข้อมูล เราจะพบว่าโปรแกรมหรือฟังก์ชันจะมีความเป็นอิสระแก่กันอย่างเห็นได้ชัด อธิบายง่ายๆก็คือโปรแกรมหรือฟังก์ชันแต่ละตัวถึงแม้จะมาจากที่เดียวกันแต่มันสามารถทำงานในคนละหน้าที่ เก็บข้อมูลคนละค่าได้ โดยจะไม่มายุ่งเกี่ยวกับกันแต่อย่างใด

Visual C++ 6.0 มีรูปแบบการอินเตอร์เฟซคล้ายกับ Visual C++ เวอร์ชันก่อน แต่จะมีจุดหนึ่งที่แตกต่างกันคือ ระบบช่วยเหลือของ Visual C++ 6.0 นั้นจะแยกออกจากตัวโปรแกรม (IDE) ซึ่งระบบช่วยเหลือจะอยู่ในรูปของ HTML Help และในการที่เราจะใช้ Visual C++ 6.0 ได้นั้นเราจะต้องติดตั้งโปรแกรม Internet Explorer 4.0 (IE4) ลงไปก่อนด้วย หากในระบบของเราไม่มี IE4 โปรแกรมติดตั้ง Visual C++ จะทำการติดตั้งให้เราก่อน

2.15 การเขียนวิซวลซีพลัสพลัสติดต่อผ่าน I/O Serial Port

ในการเขียนวิซวลซีพลัสพลัสติดต่อผ่าน ซีเรียลพอร์ต (serial port) นั้น สามารถกำหนดค่าเริ่มต้นได้ดังนี้

```
HANDLE hComm; // handle สำหรับ COM port
DCB dcb; // data structure ที่เก็บค่าสำหรับติดต่อ serial port
char * chCommPort = "COM1"; // ในที่นี้เราใช้ com port 1
int valReturn; // ใช้เก็บคืนค่ากลับในบ้างฟังก์ชัน
char chBuffer; // ไว้เก็บตัวอักษรในการรับและส่งข้อมูล
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
unsigned long nReadPort; // ใช้เก็บคืนค่ากลับในฟังก์ชัน ReadFile
unsigned long nWrittenPort; // ใช้เก็บคืนค่ากลับในฟังก์ชัน WriteFile
```

จากโค้ดข้างต้นจะเป็นการกำหนดตัวแปรที่ใช้ในโปรแกรมเพื่อรับส่งข้อมูล จากนั้นใช้คำสั่ง CreateFile ในการอ้างอิงไฟล์ข้อมูลชิ้นใหม่

```
hComm = CreateFile( chCommPort,
    GENERIC_READ | GENERIC_WRITE,
    // กำหนดแบบอ่านและเขียนไฟล์
    0,
    NULL, // ไม่กำหนดการป้องกัน attributes
    OPEN_EXISTING, // เพื่อเปิดช่องติดต่อสื่อสาร
    0, // ไม่ใช่ overlapped I/O
    NULL // เราจะใช้ NULL สำหรับติดต่อช่องทางสื่อสาร
);
```

และสามารถกำหนดค่าสำหรับติดต่อกับพอร์ตอนุกรมได้โดยใช้ data structure ชื่อ DBC ดังนี้

CBR_110 CBR_19200

CBR_300 CBR_38400

CBR_600 CBR_56000

CBR_1200 CBR_57600

CBR_2400 CBR_115200

CBR_4800 CBR_128000

CBR_9600 CBR_256000

CBR_14400

8 data bits, no parity, and 1 stop bit.

เมื่อทำการกำหนดค่าแล้วเราจะสามารถทำการรับส่งข้อความผ่านพอร์ตอนุกรมได้โดยการใส่ฟังก์ชัน WriteFile ในการส่งหรือ ReadsFile ในการรับข้อความ โดยมีรูปแบบดังนี้

```
WriteFile(hCom,m_text2,8,&nWrittenPort, NULL);
```

และ

```
ReadFile(hCom,&ch,8,&nReadPort,NULL);
```

โดยที่ &ch คือตัวแปรที่หน้าที่เป็น Buffer เก็บตัวอักษร และเลขแปดคือจำนวนของตัวอักษรที่รับเข้ามาแต่ละครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.16 ฐานข้อมูลใน ASP

ฐานข้อมูล (Database) ก็คือไฟล์ชนิดหนึ่งที่ใช้ในการเก็บข้อมูลต่างๆ ที่เราต้องการใช้ข้อมูลเหล่านั้น ก็สามารถเปิดแล้วนำไปใช้ได้ทันที ฐานข้อมูลนั้นถือว่าเป็นองค์ประกอบหลักในการใช้งานโปรแกรมเกือบทุกประเภท เพราะโปรแกรมสำเร็จรูปต่างๆ คือการใช้ฐานข้อมูลในรูปแบบหนึ่ง เมื่อเราเรียกใช้โปรแกรมก็คือเราเรียกใช้ข้อมูลในฐานข้อมูลนั่นเองเพียงแต่เราไม่ต้องเขียน คำสั่งในการเรียกฐานข้อมูลนั่นเอง เนื่องจากบริษัทผู้ผลิตได้มีการเขียนมาให้เราใช้งานได้ทันที

ฐานข้อมูลนั้นในปัจจุบันถูกผลิตออกมาโดยผู้ผลิตหลายรายที่เรารู้จักและคุ้นเคยกันดี ก็เช่น Microsoft Access, Microsoft SQL Sever, Oracle, Dbase, Foxpro และ Paradox เป็นต้น ซึ่งแต่ละชนิดก็มีข้อดีข้อเสียแตกต่างกันไป อยู่ที่ความเหมาะสมของการใช้งานในแต่ละชนิด แต่ที่นิยมใช้ใน ASP ก็คือฐานข้อมูล Microsoft Access ซึ่งปัจจุบันมีเวอร์ชัน XP แล้ว

2.16.1 การเชื่อมต่อกับฐานข้อมูล

แต่เดิมนั้นการจะเรียกใช้งานฐานข้อมูลแต่ละครั้งนั้น ไม่ใช่เรื่องง่าย เพราะความแตกต่างของชนิดของฐานข้อมูล การจะเขียนโปรแกรมเพื่อติดต่อฐานข้อมูลแต่ละชนิดย่อมแตกต่างกันไปด้วย ไมโครซอฟท์เล็งเห็นตรงจุดนี้จึงได้สร้างรูปแบบการติดต่อฐานข้อมูลที่เป็นมาตรฐานขึ้น โดยการติดต่อผ่านตัวกลางออบเจ็กต์ที่เรียกว่า ADO (ActiveX Data Object) ซึ่งการติดต่อกับขณะนี้จะทำให้ไม่ว่าเราใช้ฐานข้อมูลชนิดไหนก็ตามก็สามารถเขียนโปรแกรมในการติดต่อในรูปแบบเดียวกันทั้งหมด

ภายในตัว ADO นั้นก็มีรูปแบบการเชื่อมต่อฐานข้อมูลที่แตกต่างกันออกไปอีกหลายวิธี วิธีหลักๆ ที่เรานิยมใช้กันมีอยู่ 3 วิธีดังนี้

1. การเชื่อมต่อฐานข้อมูลแบบ DSN (Data Source Name) การที่เราจะเชื่อมต่อฐานข้อมูลแบบนี้ได้นั้น เราต้องติดตั้ง ODBC Data Source เสียก่อน ซึ่งตัว ODBC นี้จะทำหน้าที่เป็นตัวกลางอีกทีหนึ่ง การเชื่อมต่อด้วยวิธีนี้จะเขียนสคริปต์ที่สั้นและสะดวกกว่ากรณีการเชื่อมต่อฐานข้อมูลแบบอื่น แต่ข้อเสียของวิธีนี้ก็คือความล่าช้าในการเชื่อมต่อฐานข้อมูลอีกทั้งโฮสต์บางที่อาจไม่อนุญาตให้ใช้วิธีนี้ได้ เพราะการจะใช้งาน ODBC แต่ละครั้งจำเป็นต้องมีการตั้งค่าใหม่อยู่ตลอด ซึ่งการตั้งถ่วงนั้นต้องกระทำโดยผู้ดูแลเซิร์ฟเวอร์เท่านั้น ปัจจุบันวิธีนี้ไม่ค่อยมีใครนิยมใช้และ ไมโครซอฟท์ประกาศว่าจะตัดวิธีนี้ออกจากการติดต่อฐานข้อมูลใน ASP.NET ด้วย

รูปแบบ

```
Set ตัวแปร = Sever.CreaObject("ADODB.Connection")
```

```
ตัวแปร.Open "ชื่ออ้างอิงใน ODBC"
```

ตัวอย่าง

```
Set Conn = Sever.CreaObject("ADODB.Connection")
```

```
Conn.Open "b"
```

2. การเชื่อมต่อฐานข้อมูลแบบ DSNLess เป็นการเชื่อมต่อฐานข้อมูลแบบไม่ต้องผ่าน DSN เป็นการเชื่อมต่อโดยใช้สิ่งที่เรียกว่า Driver ของฐานข้อมูลแต่ละชนิด การเชื่อมต่อวิธีนี้ถือว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สะดวกกว่าแบบวิธีผ่าน DSN เพราะการใช้งานเราไม่ต้องคิดตั้งอะไรเพิ่มเติมทำให้สะดวกต่อการนำไปใช้งาน โสสต์ส่วนใหญ่รองรับการติดต่อฐานข้อมูลวิธีนี้ แต่ข้อเสียก็คือใน ASP.NET เราจะไม่สามารถติดต่อฐานข้อมูลได้ด้วยวิธีนี้และความเร็วในการเชื่อมต่อเป็นรองวิธีการเชื่อมต่อแบบผ่าน OleDb

รูปแบบ

```
Set ตัวแปร = Sever.CreaObject("ADODB.Connection")
```

```
ตัวแปร.Open "DRIVER = {ชนิดฐานข้อมูล} ; DBQ = " &Sever.MapPath ("ตำแหน่งฐานข้อมูล")
```

ตัวอย่าง

```
Set Conn = Sever.CreaObject("ADODB.Connection")
```

```
Conn.Open "DRIVER = {Microsoft Access Driver (*.mdb)} ; DBQ = " &Sever.MapPath ("a.mdb")
```

- 3. การเชื่อมต่อฐานข้อมูลแบบ OLEDB เป็นการเชื่อมต่อฐานข้อมูลผ่านโปรไวเดอร์ OLEDB ซึ่งถือว่าเป็นวิธีล่าสุดของการติดต่อฐานข้อมูลและเป็นวิธีที่มีผู้ทดสอบแล้วว่าความเร็วในการเชื่อมต่อนั้นเร็วที่สุดในบรรดาวิธีทั้งหมด นอกจากนี้จุดแข็งอีกอย่างหนึ่งก็คือใน ASP.NET นั้นจำเป็นต้องใช้การเชื่อมต่อวิธีนี้เท่านั้น เท่ากับว่าเราศึกษาวิธีนี้ไว้แล้วสามารถใช้ได้ถึง 2 ภาษาเลย

รูปแบบ

```
Set ตัวแปร = Sever.CreaObject("ADODB.Connection")
```

```
ตัวแปร . Provider = "โปรไวเดอร์ของฐานข้อมูลที่ใช้"
```

```
ตัวแปร . Open Sever.MapPath ("ตำแหน่งฐานข้อมูล")
```

ตัวอย่าง

```
Set Conn = Sever.CreaObject("ADODB.Connection")
```

```
Conn.Provider = "Microsoft.Jet.OLEDB.4.0"
```

จากตัวอย่าง เราจะประกาศตัวแปรออบเจ็กต์มาเพื่อเปิดฐานข้อมูล โดยกำหนดโปรไวเดอร์เป็น "Microsoft.Jet.OLEDB.4.0" เพื่อเปิดฐานข้อมูลชนิด Microsoft Access ซึ่งตำแหน่งของฐานข้อมูลชื่อ a.mdb นั้นเก็บไว้ในไดเรกทอรีรากของเว็บเซิร์ฟเวอร์

นอกจากจะใช้ ADO ติดต่อกับฐานข้อมูลชนิด Microsoft Access แล้ว ยังสามารถใช้ติดต่อฐานข้อมูลชนิดอื่นได้อีกด้วยโดยกำหนดชื่อของโปรไวเดอร์ เพื่อติดต่อฐานข้อมูลที่ต้องการได้ โดยรายละเอียดของชื่อโปรไวเดอร์ที่ใช้ในการติดต่อฐานข้อมูลเป็นดังตาราง

ชื่อโพรไวเดอร์	ฐานข้อมูลที่ทำกรติดต่อ
ADSDSOObject	Active Directory Services
Microsoft.Jet.OLEDB.4.0	Microsoft Jet databases
MSDAIPP.DSO.1	Microsoft Internet Publishing
MSDAORA	Oracle databases
MSDAOSP	Simple text files
MSDASQL	Microsoft OLE DB provider for ODBC
MSDataShape	Microsoft Data Shape
MSPersist	Locally saved files
SQLOLEDB	Microsoft SQL Server

ตารางที่ 2.2 ชื่อโพรไวเดอร์ในการติดต่อฐานข้อมูลแต่ละชนิด

2.17 การติดต่อฐานข้อมูลแบบ Microsoft Access

ต่อมาเราจะมาลองใช้งานฐานข้อมูลที่ทำงานร่วมกับ ADO คู่บ้าง ซึ่งฐานข้อมูลที่เรานิยมนำมาใช้ที่ได้รับความนิยมมากที่สุดนั้นก็คือ Microsoft Access ที่คุ้นเคยกันดี ซึ่งฐานข้อมูลชนิดนี้เป็นหนึ่งในโปรแกรมชุด Microsoft Office ซึ่งผลิตออกมาหลายเวอร์ชันแล้ว โดยเวอร์ชันปัจจุบันก็คือ เวอร์ชัน XP แต่การใช้งานฐานข้อมูลใน ASP นั้นเราแทบจะไม่ต้องสนใจเวอร์ชันมากนักเนื่องจากเราจะใช้ Access นั้นเพียงแค่สร้างฐานข้อมูลธรรมดาๆขึ้นมาตัวหนึ่ง ไม่ได้ใช้ความสามารถอะไรต่างๆที่เพิ่มขึ้นมาในแต่ละเวอร์ชัน ดังนั้นแล้วไม่ว่าเราจะใช้เวอร์ชันใดค่าที่ได้ใน ASP ก็ไม่แตกต่างกัน ทางที่ดีเราควรเลือกใช้เวอร์ชันเดียวกับโฮสต์ที่เราฝากเว็บไซต์ไว้จะเป็นการดีที่สุด

2.17.1 การเขียนสคริปต์ใช้งานฐานข้อมูลในออบเจกต์ ADO

หลังจากเราได้สร้างไฟล์ฐานข้อมูลมาเรียบร้อยแล้วขั้นตอนต่อไปเราจะมาลองเขียนสคริปต์เพื่อใช้งานใน ASP ซึ่งการเขียนสคริปต์เพื่อใช้งานฐานข้อมูลในออบเจกต์ ADO นั้นจะมีหลักการสำคัญๆซึ่งเป็นมาตรฐานอยู่ 5 ขั้นตอนดังต่อไปนี้

1. ติดต่อไปที่ฐานข้อมูลเพื่อบอกว่าต่อไปนี้จะติดต่อฐานข้อมูลโดยใช้ออบเจกต์ ADO โดยเราจะต้องสร้างตัวแปรออบเจกต์ขึ้นมาตัวหนึ่ง เพื่อใช้เป็นตัวแทนในการเปิดฐานข้อมูลต่อไป

รูปแบบ

```
Set ตัวแปร = Sever.CreateObject("ADODB.Connection")
```

ตัวอย่าง

```
Set Conn = Sever.CreateObject("ADODB.Connection")
```

โดยในที่นี้เราจะได้ตัวแปรที่ชื่อว่า Conn เพื่อเป็นตัวแปรในการอ้างอิงเพื่อเปิดใช้งานฐานข้อมูล

2. เปิดฐานข้อมูล หลังจากเราติดต่อกันฐานข้อมูลแล้วเราก็จะนำมาเปิดฐานข้อมูล ซึ่งการจะเปิด

นั้นเราจะมีวิธีเชื่อมต่ออยู่ 3 วิธีตามที่เราได้กล่าวมาแล้วในหัวข้อที่ผ่านมา ในที่นี้เราจะขอใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนักผู้จัดทำเห็นว่าเว็บไซต์นี้เป็นการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีเชื่อมต่อผ่านโพรไวเดอร์ OLEDB กัน ซึ่งเมื่อเรากำหนดเรียบร้อยแล้ว เราจะใช้เมธอด Open พร้อมทั้งระบุตำแหน่งไฟล์ฐานข้อมูลที่ต้องการจะเปิดดังนี้

รูปแบบ
ตัวแปรจากขั้นตอนแรก .Provider = “ชื่อของโพรไวเดอร์”

ตัวแปรจากขั้นตอนแรก .Open Server . MapPath (“ตำแหน่งของไฟล์ข้อมูล”)

ตัวอย่าง

Conn. Provider = “Microsoft. Jet. OLEDB. 4. 0”

Conn. Open Server. Mappath (“a.mdb”)

โดยในที่นี้เราใช้ตัวแปรจากขั้นตอนแรกที่ชื่อว่า Conn ในการอ้างอิงเพื่อเปิดใช้งานฐานข้อมูล เชื่อมต่อผ่านโพรไวเดอร์ Microsoft.Jet.OLEDB.4.0 ซึ่งเป็นโพรไวเดอร์ของ Microsoft Access โดยเฉพาะ หลังจากนั้นจะระบุตำแหน่งของไฟล์ฐานข้อมูลซึ่งในที่นี้มีชื่อว่า a.mdb อยู่ในโฟลเดอร์ราก

- 3. ใช้คำสั่ง SQL เพื่อเลือกข้อมูลจากตาราง หลังจากเราได้เปิดฐานข้อมูลขึ้นมาพร้อมทั้งรู้ตำแหน่งและชื่อไฟล์ฐานข้อมูลเรียบร้อยแล้ว ขั้นตอนต่อไปเราจะต้องเลือกว่าจะนำข้อมูลจากคอลัมน์ใดบ้างมาใช้งาน ซึ่งขั้นตอนนี้เราจะใช้คำสั่ง SQL ในการเลือกข้อมูลแต่ละคอลัมน์ของเทเบิลในฐานข้อมูล

รูปแบบ

ตัวแปร = “SELECT ชื่อของคอลัมน์ในเทเบิล FROM ชื่อของเทเบิล ”

ตัวอย่าง

Sql = “ SELECT * FROM member ”

โดยในที่นี้เราจะสร้างตัวแปรที่มีชื่อว่า Sql เพื่อเก็บข้อมูลที่เลือกได้ตามเงื่อนไขจากฐานข้อมูล ซึ่งในที่นี้เราเลือกทุกคอลัมน์ (* คือการเลือกทุกคอลัมน์) ในเทเบิลชื่อว่า member

- 4. หลังจากเราได้ข้อมูลต่างๆจากฐานข้อมูลมาเรียบร้อยแล้ว เราก็จะมาจัดการกันว่าจะนำข้อมูลดังกล่าวมาทำอะไรต่อไป ซึ่งเราจะต้องใช้งานข้อมูลต่างๆผ่านออบเจ็กต์อีกหนึ่งก่อน ซึ่งการใช้งานออบเจ็กต์จะต้องสร้างตัวแปรขึ้นมาอีกตัวหนึ่งเพื่อรองรับข้อมูลจากฐานข้อมูล หลายคนอาจสงสัยว่าขั้นตอนนี้อาจสงสัยว่าขั้นตอนนี้แตกต่างจากขั้นตอนที่ 1 อย่างไร ทำไมต้องประกาศตัวแปรหลายๆตัว สาเหตุที่ต้องทำเช่นนี้ก็เพราะว่าตัวแปรที่ประกาศในขั้นตอนที่ 1 นั้นใช้ เพื่อในการเปิดฐานข้อมูลเท่านั้น ส่วนตัวแปรในขั้นตอนนี้สร้างขึ้นเพื่อใช้จัดการข้อมูลต่างๆ

รูปแบบ

Set ตัวแปร = Sever.CreateObject(“ADODB. Recordset”)

ชื่อตัวแปร .Open ตัวแปรจากขั้นตอนที่ 3, ตัวแปรจากขั้นตอนที่ 1, 1, 3

ตัวอย่าง

Set rs = Sever.CreateObject(“ADODB. Recordset”)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

rs. Open Sql, Conn, 1, 3

โดยในที่นี้เราจะสร้างตัวแปรที่มีชื่อว่า rs ซึ่งตัวแปรดังกล่าวจะทำหน้าที่ไว้เป็นตัวอ้างอิงต่อไป เมื่อเราเรียกใช้ข้อมูลในฐานข้อมูล หลังจากนั้นจะใช้เมธอด Open เพื่อเปิดตัวแปร rs ขึ้นมาเพื่อรับค่าต่างๆ ซึ่งจะมีรายละเอียดคอลัมน์ที่ต้องการใช้งานตามที่ได้สร้างไว้แล้วในขั้นตอนที่ 3 จากตัวอย่างโปรแกรมจะเห็นว่ามิตัวเลขอีก 2 ตัวซึ่งเราจำเป็นต้องกล่าวถึงด้วย

- ตัวเลขตัวแรกซึ่งในที่นี้คือหมายเลข 1 หมายถึง การกำหนดชนิดของคำสั่งที่ใช้ในการเลือกคอลัมน์ซึ่งในที่นี้ก็คือคำสั่ง SQL นั่นเอง
- ตัวเลขที่สองในที่นี้คือหมายเลข 3 หมายถึงการกำหนดการอนุญาตหรือพูดง่าย ๆ ว่าเป็นการกำหนดสิทธิการใช้งานนั่นเองว่าเรามีสิทธิที่จะทำอะไรกับไฟล์นี้บ้าง ในที่นี้กำหนดให้เราสามารถกระทำต่อสิ่งใดกับไฟล์ก็ได้ทั้งอ่าน เขียน หรือเรียกให้ทำงาน

5. ปิดฐานข้อมูลที่ไม่ต้องการใช้เมื่อเราใช้งานฐานข้อมูลเสร็จแล้ว เราต้องเขียนคำสั่งเพื่อปิดฐานข้อมูลนั้น โดยเราจะใช้เมธอด Close เพื่อปิดฐานข้อมูลนั้นเสีย

รูปแบบ

ตัวแปรจากข้อแรก.Close

ตัวอย่าง

Conn.Close

นอกจากการสั่งปิดฐานข้อมูลแล้วเราอาจจะเพิ่มคำสั่งเข้าไปอีกส่วนหนึ่ง เนื่องจากว่าแม้เราจะเขียนคำสั่งเพื่อปิดฐานข้อมูลแล้ว แต่ตัวแปรที่เราได้สร้างขึ้นมาเพื่อเป็นตัวแทนในการเปิดฐานข้อมูลนั้นยังไม่ได้ถูกลบออกไปจากระบบ ซึ่งหากเราไม่ต้องการใช้ตัวแปรนี้อีกต่อไปก็ควรที่จะลบตัวแปรนี้ทิ้ง เพื่อให้ทรัพยากรของระบบเรามีมากขึ้นเว็บเซิร์ฟเวอร์จะทำงานได้เร็วขึ้น ซึ่งเมื่อเราได้เพิ่มแล้วสคริปต์จะออกมาดังนี้

รูปแบบ

ตัวแปรจากข้อแรก.Close

Set ตัวแปรจากข้อแรก = Nothing

ตัวอย่าง

Conn.Close

Set Conn=Nothing

จึงเท่ากับเราเพิ่มส่วน Set ตัวแปร =Nothing เข้าไปแค่นั้นเองนอกนั้นก็ไม่มีอะไรที่เปลี่ยนแปลงไปจากเดิม

2.18 มาตรฐานคำสั่งของโมเด็ม (AT Command)

บริษัท Hayes Microcomputer Products Inc. เป็นผู้คิดชุดคำสั่งชุดหนึ่งขึ้นมาเพื่อสั่งงานโมเด็มสำหรับเครื่องคอมพิวเตอร์ส่วนบุคคลและได้รับความนิยมเป็นอย่างมากจนถือเป็นมาตรฐานอันหนึ่ง มาตรฐานคำสั่งนี้เรียกว่า Hayes Command Set เป็นคำสั่งที่ช่วยให้ผู้ใช้สามารถกำหนดการทำงานต่างๆ ของโมเด็มได้โดยใช้ซอฟต์แวร์สั่งจากคอมพิวเตอร์ไปยังโมเด็มโดยตรง ทำให้เราไม่ต้องปรับสวิตช์เพื่อเลือกการทำงานแบบต่างๆ ของโมเด็มอีกต่อไป โมเด็มที่เราใช้กับเครื่องคอมพิวเตอร์ส่วนบุคคลหรือแมคอินทอช (Macintosh) เกือบทั้งหมดจะรับคำสั่งตามมาตรฐานของ Hayes

เมื่อคำสั่งของโมเด็มเป็นสิ่งจำเป็นสำหรับการใช้งานโมเด็มสมัยใหม่ ทาง CCITT หรือ ITU-T ก็ได้มีการกำหนดมาตรฐานของคำสั่งโมเด็มขึ้นเช่นกัน เรียกว่ามาตรฐานคำสั่งแบบ V.25 bis แต่เนื่องจากว่าโมเด็มที่ใช้กับเครื่องคอมพิวเตอร์ และซอฟต์แวร์สั่งงานโมเด็มเกือบทั้งหมดใช้คำสั่งตามมาตรฐานของ Hayes ดังนั้นมาตรฐานคำสั่งโมเด็มแบบ V.25 bis จึงไม่ค่อยมีใช้กันเท่าใดนัก เรียกได้ว่าเครื่องคอมพิวเตอร์ติดต่อกับโมเด็มโดยใช้คำสั่งแบบ Hayes Command ทั้งหมด ส่วนคำสั่งแบบ V.25 bis มีใช้บ้างเหมือนกัน แต่เป็นในระบบชุมสายตามมาตรฐานของ CCITT หรือ ITU-T แต่โมเด็มบางชนิดสามารถรับคำสั่งได้ทั้ง 2 แบบ แต่โมเด็มราคาถูกที่ใช้กับคอมพิวเตอร์ส่วนมากมักจะรับคำสั่งได้แต่ Hayes Command เท่านั้น ผู้ผลิตบริษัทโมเด็มอื่นๆ ได้ใช้มาตรฐานคำสั่งตามอย่างบริษัท Hayes โดยรับคำสั่งจากเครื่องคอมพิวเตอร์ตามแบบคำสั่งของ Hayes ทุกประการ เรียกว่าเป็นโมเด็มแบบ Hayes Compatible ซึ่งโมเด็มที่ใช้คำสั่งแบบ Hayes นั้นนอกจากจะรับคำสั่งจากเครื่องคอมพิวเตอร์เหมือนคำสั่งของบริษัท Hayes แล้วโมเด็มยังต้องตอบสนองต่อคำสั่งต่างๆ เหมือนกับที่บริษัท Hayes ใช้ รวมถึงมีที่เก็บตัวแปรต่างๆ เหมือนกันอีกด้วยเพื่อให้ซอฟต์แวร์ทำงานได้ต่อกับโมเด็มได้ถูกต้อง

ภายในตัวโมเด็มจะมีหน่วยความจำพิเศษสำหรับเก็บตัวแปรในการทำงานที่โมเด็มแบบเก่าจะยังคงเก็บค่าต่างๆเอาไว้แม้ว่าปิดหรือดึงปลั๊กโมเด็มออกก็ตาม แต่โมเด็มที่ใช้คำสั่งของ Hayes เรียกหน่วยความจำส่วนนี้ว่า S-Register เอาไว้ใช้เก็บพารามิเตอร์ในการทำงานของโมเด็ม

2.18.1 มาตรฐานคำสั่งโมเด็มของ Hayes

Hayes Command เป็นคำสั่งที่ใช้สั่งงานโมเด็ม มีอีกชื่อหนึ่งเรียกว่า AT Command เพราะคำสั่งทุกคำสั่งจะขึ้นต้นด้วยตัวอักษร AT เสมอ เมื่อจบคำสั่งให้ปิดท้ายด้วยรหัส ASCII ตัวที่ 13 คือ Carriage Return หรือกดปุ่ม Enter โมเด็มก็จะรับคำสั่งนั้นไปทำงานทันที และตอบคำว่า OK กลับมา บางคำสั่งจะสังเกตว่ามีรหัสหรือตัวเลขต่อท้ายเพื่อระบุวิธีการทำงานโดยละเอียดทีหนึ่ง เช่น ATB อาจตามด้วย 0 หรือ 1 ดังนั้นคำสั่งที่ใช้จริงอาจเป็น ATB1 ก็ได้

2.18.2 การใช้งานชุดคำสั่ง AT

2.18.2.1 ประเภทของชุดคำสั่ง AT

โดยพื้นฐานแล้วเราจะสามารถแบ่งชุดคำสั่ง AT ออกเป็น 2 ประเภทใหญ่ๆได้ดังนี้ ประเภทแรกคือชุดคำสั่งที่ใช้ในการปฏิบัติงานอย่างเช่น ATD (คำสั่งให้หมุนหมายเลขโทรศัพท์) หรือ ATH (คำสั่งให้วางสายโทรศัพท์) เป็นต้น ส่วนประเภทที่สองคือชุดคำสั่งที่ใช้กำหนดค่าหรือเปลี่ยนแปลงค่าต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อย่างเช่น AT57=90 เป็นการกำหนดค่าให้รีจิสเตอร์ S7 มีค่า 90 ซึ่งก็คือให้โมเด็มรอการตอบรับจากโมเด็มปลายทางเป็นเวลา 90 วินาที หลังจากนั้นหากยังไม่ได้รับสัญญาณใดๆ ตอบกลับมาโมเด็มจะวางสายทันที

2.18.2.2 การออนไลน์และออฟไลน์

สถานะออฟไลน์สามารถเรียกได้อีกอย่างหนึ่งว่าสถานะคำสั่ง (Command state) หมายถึงสถานะที่ผู้ใช้สามารถที่จะส่งคำสั่งต่างๆ ไปยังโมเด็มได้ หรือพูดอีกนัยหนึ่งก็คือ สถานะที่โมเด็มจะแปลความหมายของข้อมูลที่ได้รับมาจาก PC ให้เป็นคำสั่งเท่านั้น ซึ่งสถานะนี้โมเด็มไม่ได้รับคำสั่งข้อมูลกับโมเด็มปลายทางแต่จะสื่อสารกันกับ PC เท่านั้น ส่วนสถานะออนไลน์ หมายถึง สถานะที่โมเด็มได้เชื่อมต่อกับโมเด็มปลายทางเป็นที่เรียบร้อยแล้ว ข้อมูลที่ส่งออกมาจาก PC ก็จะผ่านจากโมเด็มต้นทางไปยังโมเด็มปลายทางเสมอ ในกรณีนี้หากผู้ใช้ต้องการที่จะส่งคำสั่งให้กับโมเด็มหรือต้องการจะให้โมเด็มกลับมาอยู่ในสถานะออฟไลน์ เพื่อรับคำสั่งจาก PC ก็จะสามารถทำได้ โดยมีวิธีการอยู่ 2 วิธีคือ หนึ่งให้โมเด็มวางสายแล้วกลับมาอยู่ในสถานะออนไลน์ใหม่และวิธีที่สองคือ ส่งชุดอักขระ Escape Sequence เข้าไปยังโมเด็มในขณะที่ออนไลน์ ซึ่งวิธีหลังจะมีความเหมาะสมกว่า เพราะการเชื่อมต่อระหว่างโมเด็มจะยังคงดำเนินไปอยู่ และหลังจากที่ได้ส่งคำสั่งต่างๆ ให้กับ โมเด็มเป็นที่เรียบร้อยแล้วก็จะสามารถกลับเข้าไปอยู่ในสถานะออนไลน์ได้เช่นเดิม วิธีการส่งชุดอักขระ Escape Sequence คือให้รอ 1 วินาที (เรียกว่า Guard Time) แล้วกดปุ่ม + ติดต่อกัน 3 ครั้ง (+++) การที่ติดต่อกัน 1 วินาทีก่อนที่จะกดปุ่มเครื่องหมาย + เป็นสิ่งจำเป็นเนื่องจากทำให้โมเด็มได้รู้ว่าอักษร + นั้นเป็นส่วนของอักขระ Escape Sequence ไม่ใช่ส่วนของข้อมูลที่จะต้องส่งไปยังโมเด็มปลายทาง

2.18.2.3 รูปแบบของชุดคำสั่ง AT

การใช้ชุดคำสั่ง AT จำเป็นต้องเริ่มด้วยอักษร AT เสมอ และจะต้องจบด้วยการกด Enter หรือ Carriage Return ยกเว้นคำสั่ง A/ (ไม่ต้องกดปุ่ม Enter) ซึ่งหมายถึงให้โมเด็มกลับไปทำคำสั่งล่าสุดซ้ำอีกครั้งหนึ่ง การที่ Hayes ได้กำหนดคำสั่งต่างๆ ขึ้นต้นด้วยอักษร AT ก็เพราะต้องการให้โมเด็มรับรู้ถึงความเร็วและรูปแบบของอักษรคำสั่ง ที่ถูกส่งออกมาจากพอร์ตสื่อสารอนุกรมของ PC ซึ่งอย่าสับสนระหว่างความเร็วของพอร์ตสื่อสารอนุกรม (ความเร็วของ DTE) และความเร็วของโมเด็ม (ความเร็วของ DCE) ในกรณีนี้จะหมายถึงความเร็วของ DTE ตัวอย่างการใช้ชุดคำสั่ง AT ก็ได้แก่คำสั่ง ATH หมายถึงคำสั่งที่ให้โมเด็มวางสายโทรศัพท์เป็นต้น แต่บางคำสั่งต้องการพารามิเตอร์เพิ่มเติมอย่างเช่น คำสั่ง ATDT 3197707 หมายถึงให้โมเด็มหมุนหมายเลข 3197707 โดยหมุนแบบระบบโทน (Tone : T) แต่ถ้าต้องการให้โมเด็มหมุนแบบระบบพัลส์ (Pulse) ต้องใช้คำสั่ง ATDP 3197707 เป็นต้น

2.17.2.4 การสนองคำสั่งของโมเด็ม

เมื่อผู้ใช้ส่งคำสั่งต่างๆ ไปให้กับโมเด็มแล้ว โมเด็มก็จะตอบสนองคำสั่ง โดยจะส่งข้อความที่เรียกว่า Result code กลับมายังเครื่อง PC และจะปรากฏขึ้นบนจอภาพขณะที่รันโปรแกรมสื่อสารไว้ Result code เหล่านี้จะเป็นภาษาอังกฤษ อย่างเช่น OK, ERROR, CONNECT 2400 หรืออาจจะเป็นตัวเลขอื่นๆ ชุดคำสั่ง AT ที่สำคัญ เช่น ATA (รับโทรศัพท์) และ ATD (หมุนหมายเลข โทรศัพท์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.19 รูปแบบของ Protocol Description Unit (PDU)

มีสองช่องทางที่ใช้ในการส่งและรับข้อความ SMS คือ โหมดอักษรและโหมด PDU โดยโหมดอักษรเป็นการเข้ารหัสสตรีมมิ่งบิตที่แสดงในโหมด PDU

วิธีการส่งข้อความในโหมด PDU โดยการใช้คำสั่ง

AT+CMGS=23

> 0011000A916641429488000AA0A4166514A0D328B522A <ctrl-z>

ตัวเลข 23 คือ จำนวนออกเต็ทของข้อความ ส่วนข้อความ PDU จะแสดงได้ดังตาราง

ออกเต็ท	ลักษณะ
00	ความยาวของข้อมูล SMSC ในที่นี้ค่าความยาวเป็น 0 หมายความว่าจะนำ SMSC ที่ถูกเก็บไว้ในโทรศัพท์มาใช้
11	ออกเต็ทแรกของข้อความ SMS-SUBMIT
00	TP-Message-Reference "00" คือการให้โทรศัพท์อ้างอิงข้อความด้วยตัวเอง
0A	ความยาวของแอดเดรส คือ ความยาวของเบอร์โทรศัพท์ ในที่นี้คือมี 10 ตัว
91	ประเภทของแอดเดรส โดย 91 จะบอกว่าเบอร์โทรศัพท์เป็นรูปแบบสากล
66 41 42 94 88	เบอร์โทรศัพท์ในรูปแบบเขมออกเต็ท ในที่นี้ใช้เบอร์ 6614244988 แล้วทำการสลับตำแหน่งกันเป็นคู่ๆ
00	TP-PID เป็นการระบุโปรโตคอล
00	TP-DCS
AA	TP-Validity-Period บอกถึงระยะเวลาในการตรวจสอบว่าการส่งสำเร็จหรือไม่
0A	TP-User-Data-Length จำนวนอักษรในข้อความที่ทำการส่ง
41 66 51 4A 0D 32 8B 52 2A	TP-User-Data เป็นออกเต็ทที่แสดงแทนข้อความในที่นี้คือ ALERTALERT

ตารางที่ 2.3 ข้อความ PDU ที่ใช้ในการส่ง SMS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.20 ไปรษณีย์อิเล็กทรอนิกส์ (Electronic Mail: E-mail)

ไปรษณีย์อิเล็กทรอนิกส์หรืออีเมลล์ คือซอฟต์แวร์ (Software) ที่ใช้ในการแลกเปลี่ยนข้อมูลระหว่างเครื่องคอมพิวเตอร์หลายๆเครื่อง โดยอาศัยโมเด็มเป็นตัวเชื่อมเพื่อที่จะทำการแลกเปลี่ยนข้อมูลกันได้ ในการใช้โมเด็มทำให้เราสามารถที่จะเชื่อมสัญญาณได้ทั่วโลก ไปจนถึงสามารถที่จะทำการติดต่อได้กับบุคคลทั่วโลก

การใช้งานอีเมลล์เป็นการใช้งานที่ได้รับความนิยมแพร่หลายมากประเภทหนึ่งในกลุ่มผู้ใช้งานอินเทอร์เน็ต วิธีใช้ก็ง่ายแต่มีประโยชน์มากโดยเริ่มจากการพิมพ์ข้อความเข้าไปเป็นจดหมาย และสั่งให้มีการส่งจดหมายที่อยู่ในรูปของอิเล็กทรอนิกส์นั้นผ่านเข้าไปยังเครือข่ายอินเทอร์เน็ตซึ่งจะทำหน้าที่เสมือนบุรุษไปรษณีย์นำจดหมายนั้นส่งต่อไปยังผู้รับปลายทาง และผู้รับก็จะอ่านจดหมายนั้นได้จากการเข้าใช้อินเทอร์เน็ตอีกเช่นกัน นอกจากนี้เราสามารถที่จะส่งข้อมูลอื่นนอกเหนือจากตัวอักษรทั่วไป เช่น ข้อมูลเป็นรูปภาพ ข้อมูลเสียง เป็นต้น รวมไปถึงจดหมายได้อีกด้วย

ในการรับส่งสัญญาณระหว่างเครื่องคอมพิวเตอร์นั้น จะต้องมีการติดตั้งระบบอีเมลล์ในระบบเครือข่าย โดยการตั้งไอพีแอดเดรส (IP Address) และตู้ไปรษณีย์ (Mailbox) เป็นของตัวเอง และต้องมีการตั้งรหัสบัญชีของผู้ใช้ (User ID) และรหัสผ่าน (Password)

ส่วนประกอบของระบบไปรษณีย์อิเล็กทรอนิกส์

ระบบไปรษณีย์อิเล็กทรอนิกส์จะประกอบไปด้วย 2 ส่วนหลักได้แก่

1. Front End หรือ User Agent (UA) เป็นส่วนแสดงผลและรับคำสั่งจากผู้ใช้ เป็นส่วนที่ผู้ใช้งานมองเห็นโดยตรง ผู้พัฒนาแต่ละบริษัทจะออกแบบให้ผู้ใช้งานสามารถใช้งานได้ง่าย ซึ่งมักจะประกอบไปด้วยเมนูที่มีฟังก์ชันการทำงานต่างๆดังนี้

- Create ใช้ในการสร้างไปรษณีย์อิเล็กทรอนิกส์ที่ใช้ในการส่ง
- Send ใช้ส่งไปรษณีย์อิเล็กทรอนิกส์ที่สร้างแล้ว
- Notification เตือนเมื่อได้รับไปรษณีย์อิเล็กทรอนิกส์ใหม่
- Inbox ส่วนแสดงรายชื่อของไปรษณีย์อิเล็กทรอนิกส์ที่ได้รับ
- Read ใช้อ่านไปรษณีย์อิเล็กทรอนิกส์
- Save ใช้ทำสำเนาไฟล์
- Forward เพื่อส่งไปรษณีย์อิเล็กทรอนิกส์ที่ได้รับมาต่อไปยังผู้รับคนอื่น
- Carbon Copy เพื่อส่งไปรษณีย์อิเล็กทรอนิกส์ไปยังผู้รับคนอื่น
- Attachment เพื่อแนบไฟล์อื่นมาด้วย
- Mailing List แสดงรายชื่อผู้รับเพื่อสะดวกในการส่ง
- Viewing Attach File เพื่อตรวจดูไฟล์ที่แนบมาอาจจะอยู่ในรูปแบบของไฟล์รูปภาพ หรือไฟล์แบบกระดาษ หรือไฟล์แบบฐานข้อมูล
- Rule-Based Message Management เพื่อจัดเก็บไฟล์ฐานข้อมูลที่ได้รับ

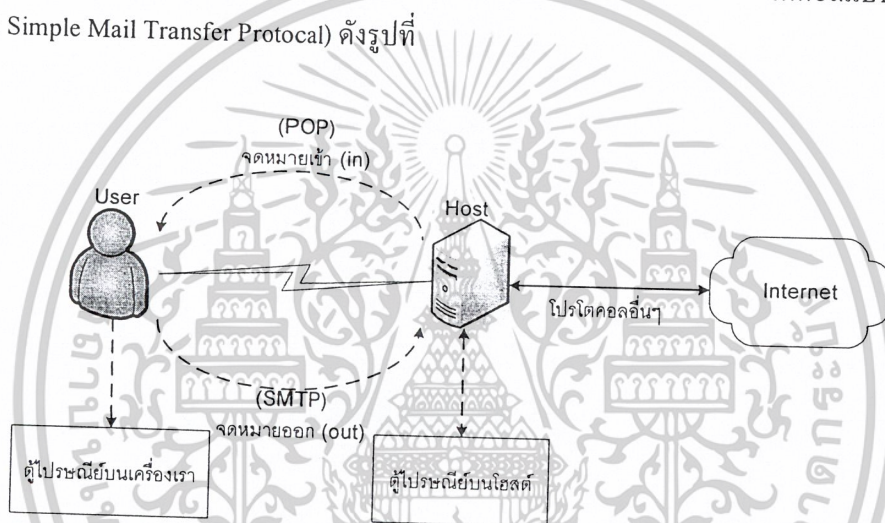
2. Black End หรือ Transport Agent - TA หรือ Message Transfer Agent เป็นส่วนสำคัญที่สุดของไปรษณีย์อิเล็กทรอนิกส์ มีหน้าที่รับส่งข่าวสารและจัดการระบบไปรษณีย์อิเล็กทรอนิกส์ ส่วนนี้ผู้ใช้จะเอกสารนี้เป็นเอกสารที่ส่งวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มองไม่เห็นโดยตรง บางครั้งอาจจะเรียกว่า Message Transport Agent – MTA สามารถแบ่งออกเป็น ส่วนย่อยหลายส่วนเช่น

- Transport Service ส่วนบริการส่งไปรษณีย์อิเล็กทรอนิกส์
- Directory Service ส่วนบริการไดเรกทอรีเพื่อให้ระบบไปรษณีย์อิเล็กทรอนิกส์สามารถปรับเลขหมายปลายทางของระบบไปรษณีย์อิเล็กทรอนิกส์อื่นให้มีเลขที่เดียวกัน
- Message Store ส่วนเก็บข่าวสารของไปรษณีย์อิเล็กทรอนิกส์

2.20.1 มาตรฐานที่ใช้ในการรับส่งอีเมลล์

โปรโตคอลหรือมาตรฐานในการติดต่อระหว่างเครื่องในการรับส่งอีเมลล์ที่นิยมใช้กันระหว่าง โฮสต์ของไอเอสพี (ISP) กับเครื่องพีซีของเราคือ ป๊อป (POP : Post Office Protocol) ซึ่งใช้ในการรับอีเมลล์จากโฮสต์คอมพิวเตอร์ ส่วนในการส่งอีเมลล์กลับไปยังโฮสต์จะใช้โปรโตคอลแบบเอสเอ็มทีพี (SMTP : Simple Mail Transfer Protocol) ดังรูปที่



รูปที่ 2.10 แสดงการรับส่งอีเมลล์แบบมาตรฐาน

2.20.2 รูปแบบเอกสารแนบของอีเมลล์

การส่งไฟล์ข้อมูลใดๆเป็นเอกสารแนบกับอีเมลล์มานั้น ไฟล์นั้นๆจะต้องอยู่ในรูปแบบที่โปรแกรมฝั่งผู้รับรู้จักและสามารถแปลงเป็นไฟล์ต้นฉบับเดิมได้อย่างถูกต้อง ปัญหาสำคัญคือโปรแกรมจัดการอีเมลล์บนเครื่องที่เป็นโฮสต์คอมพิวเตอร์ทั้งหลายมักจะรับส่งอีเมลล์โดยถือแต่ละไบต์(แต่ละตัวอักษร) เป็นข้อมูลแบบ ASCII แค่ 7 บิตคือมีเฉพาะตัวอักษรในภาษาอังกฤษเท่านั้น ทำให้ยุ่งยากในการที่จะต้องแปลงข้อมูลแบบอื่นๆซึ่งอาจเป็นแบบ 8 บิต เช่น อักษรภาษาไทย ภาพ เสียง โปรแกรมในรูปแบบของไบนารีไฟล์ ให้กลับไปอยู่ในรูปข้อมูล 7 บิตหรือน้อยกว่านั้นก่อนจึงจะส่งไปกับอีเมลล์ได้ และจะต้องแปลงกลับคืนสภาพเดิมได้อย่างถูกต้องที่ปลายทางด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

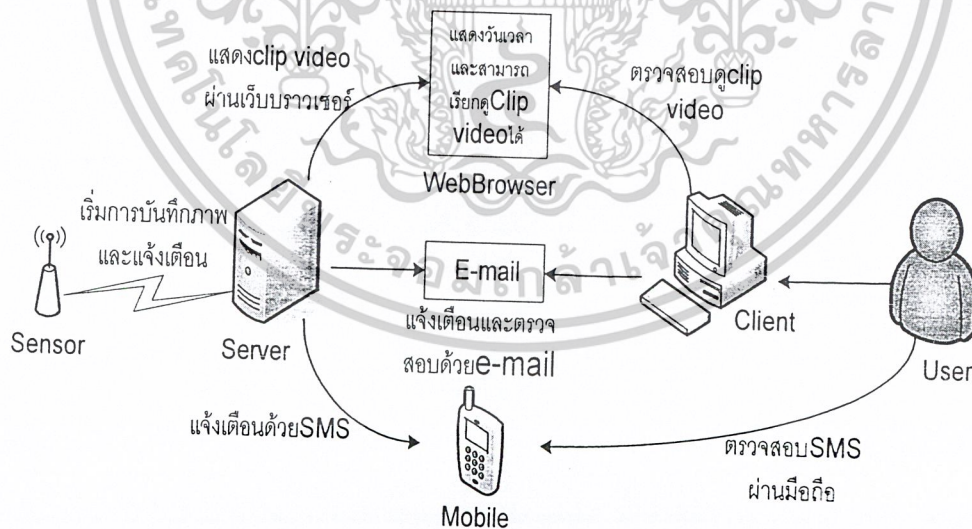
การออกแบบและการสร้าง

3.1 การทำงาน

การทำงานของระบบจะเริ่มต้นที่เมื่อมีการเปิดโปรแกรมแสดงภาพที่เครื่อง server โดยโปรแกรมแสดงภาพจะแสดงภาพจากกล้องวงจรปิดตลอดเวลา ซึ่งการทำงานของโปรแกรมนี้อาจมีส่วนของการรองรับสัญญาณที่ส่งมาจาก sensor หากโปรแกรมตรวจสอบว่ามีการส่งสัญญาณก็จะทำการบันทึกภาพ ซึ่งช่วงเวลาของการบันทึกภาพแต่ละครั้งนั้นได้มีการกำหนดให้บันทึกภาพเป็นเวลา 7 วินาทีต่อการบันทึก 1 ครั้ง ซึ่งชื่อไฟล์วิดีโอที่ทำการบันทึกนั้นจะเป็นวันที่เดือนปีค.ศ.และเวลาที่ทำการบันทึก พร้อมกันนั้นก็จะมีแจ้งเตือนไปยังผู้ควบคุมระบบทาง SMS ผ่านโทรศัพท์มือถือโดยจะทำการส่งเป็นข้อความสั้นๆว่า "ALERTALERT" และก็จะทำการแจ้งเตือนผ่านทาง E-mail ในเวลาเดียวกัน ผู้ควบคุมระบบสามารถที่จะเรียกดูไฟล์วิดีโอที่ทำการบันทึกได้ผ่านทาง Internet

โดยในส่วนการทำงานของระบบจะสามารถแบ่งออกเป็นส่วนๆ ได้ดังนี้

1. ส่วนของการแสดงภาพจากกล้องวงจรปิด
2. ส่วนของ sensor ที่คอยตรวจสอบผู้บุกรุก
3. ส่วนของการบันทึกภาพพร้อมทั้งวันเดือนปีและเวลาที่ทำการบันทึก
4. ส่วนของการแจ้งเตือนไปยังผู้ตรวจสอบทาง SMS และผ่านทาง E-mail
5. ส่วนของ Web site เพื่อที่ผู้ตรวจสอบสามารถตรวจสอบผ่านทาง Internet ได้

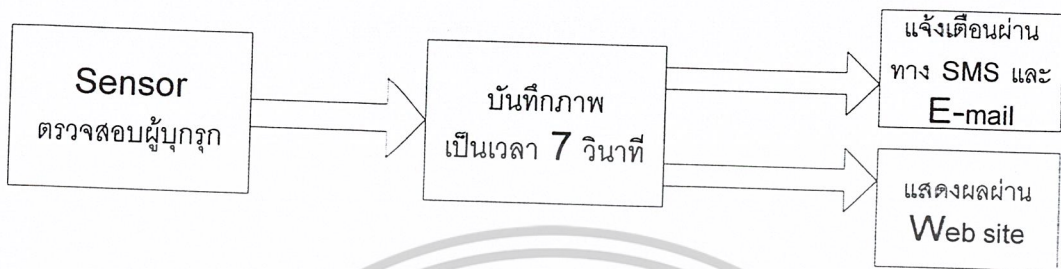


รูปที่ 3.1 การทำงานโดยทั่วไปของระบบ

การทำงานของระบบเมื่อมองทางด้านการใช้งานแล้ว จะแบ่งลักษณะของการใช้งานออกเป็น 2 ส่วนใหญ่ๆคือ 1. การทำงานด้าน Server 2. การทำงานด้าน Client

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. การทำงานด้าน Server จะประกอบด้วยการทำงานในส่วนของตัวเซิร์ฟเวอร์ซึ่งจะคอยตรวจสอบผู้บุกรุก ถ้าตรวจพบโปรแกรมที่ทำหน้าที่บันทึกภาพก็จะทำการบันทึกภาพวีดีโอไว้ และก็จะมีการแจ้งเตือนไปยังผู้ใช้งานทั้งทาง SMS และ e-mail รวมทั้งแสดงภาพวีดีโอผ่านทางเว็บเบราว์เซอร์เพื่อให้ผู้ใช้สามารถตรวจสอบภาพวีดีโอได้



รูปที่ 3.2 Block Diagramการทำงานของเครื่อง Server

2. การทำงานด้าน Client จะเป็นส่วนที่ผู้ใช้งานสามารถใช้เพื่อการตรวจสอบภาพวีดีโอจากการแจ้งเตือนของระบบ ซึ่งมีการทำงานตาม Block Diagram ที่ 3.2



รูปที่ 3.3 Block Diagram การตรวจสอบของผู้ใช้

3.2 โปรแกรมแสดงภาพ (Video Capture)

การที่ภาษา C เป็นภาษาที่มีความยืดหยุ่นสูง เข้าใจง่าย มีคำสั่งอำนวยความสะดวกมากมาย มีประสิทธิภาพของภาษาสูง และภาษา C เป็นการเขียนโปรแกรมแบบโครงสร้าง (Structure) เป็นการรวมเอาข้อมูลที่ต่างกัน รวมเข้าไว้ด้วยกันภายใต้ชื่อเดียวกัน ซึ่งจะช่วยอำนวยความสะดวกในการเขียนโปรแกรมได้ง่ายขึ้นและมีความถูกต้องมากยิ่งขึ้น

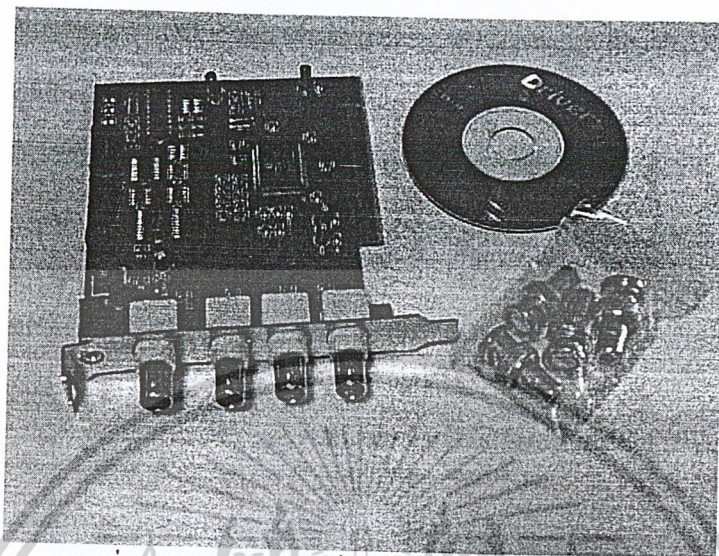
ซึ่งในการแสดงภาพนั้นเราได้ใช้การ์ดแคปเจอร์ในการแสดงสัญญาณภาพจากกล้องวงจรปิด โดยกล้องวงจรปิดที่ใช้เป็นกล้องวงจรปิดขนาดจิ๋วชนิด Cmos

การ์ดที่ใช้เป็น PCI Card สำหรับต่อกล้องวงจรปิดเข้า computer 25 เฟรม และคุณสมบัติทั่วไปของการ์ดมีดังนี้

- รองรับกล้องได้ถึง 4 กล้องต่อ 1 การ์ด ใส่การ์ดได้ถึง 4 การ์ดในหนึ่งเครื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ใช้ได้กับกล่องวงจรปิดทุกรุ่นไม่ว่าจะเป็นสีหรือขาวดำ แบบเดินสายหรือแบบไร้สาย
- ติดตั้งง่ายใช้ได้กับคอมพิวเตอร์ PC ทุกรุ่น (PENTIUM 300Hz ขึ้นไป)



รูปที่ 3.4 การ์ดแคปเจอร์ที่ใช้ในการดึงภาพจากกล้อง

กล่องวงจรปิดขนาดจิ๋วที่มีคุณสมบัติโดยทั่วไปดังนี้

- ตัวกล่องแสดงภาพสี และมีเสียง มีความคมชัด ภาพชัดเจน (ภาพสี)
- ตัวกล่องมีขนาด (กว้าง 3 ซม. ยาว 3 ซม. หนา 1 ซม.)
- สามารถต่อสายตรงเข้า ช่อง AV ชมภาพผ่านหน้าจอทีวีได้ทันที



รูปที่ 3.5 กล่องวงจรปิดขนาดจิ๋ว

และในส่วนของงานแจ้งเตือนทาง SMS ผ่านโทรศัพท์มือถือนั้น ได้ใช้โทรศัพท์มือถือยี่ห้อ Siemens รุ่น C35 แล้วเชื่อมต่อกับคอมพิวเตอร์ผ่านทางสายซีเรียลพอร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 โทรศัพท์มือถือ Siemens C35และสาย Serial port

ส่วนโปรแกรม Video capture ISCS (Internet Security Camera System) ของโครงการนี้ ถูกเขียนขึ้นด้วยภาษา C++ โดยโปรแกรมที่ใช้ในการเขียนคือ Visual C++ Version 6.0

3.3 โปรแกรมบันทึกภาพวิดีโอ

การเขียนโปรแกรมการดึงภาพที่ใช้ในรายงานฉบับนี้นั้น ทำการเขียนโดยใช้ไคอะลือกอิดิเตอร์สร้างไคอะลือกริซอร์สขึ้นมา แล้วทำการเปลี่ยนค่า ID ของไคอะลือกอิดิเตอร์ พร้อมทั้งทำการปรับแต่งไคอะลือกอิดิเตอร์ให้เหมาะสมกับการใช้งาน โดยมีส่วนของปุ่มควบคุมการทำงานต่างๆ มีส่วนของจอแสดงภาพจากกล้องวิดีโอ จากนั้นเพิ่มคลาสใหม่ด้วย ClassWizard เพราะไคอะลือกริซอร์สที่สร้างขึ้นใหม่นั้นจะยังไม่สามารถใช้งานได้เพราะยังไม่มีคลาสรองรับ เราจะต้องสร้างคลาสขึ้นมาจัดการกับไคอะลือกอิดิเตอร์ (ทุกๆ ไคอะลือกอิดิเตอร์ที่สร้างจะต้องมีคลาสเป็นของตัวเอง) ซึ่งในขั้นตอน ClassWizard นี้จะสร้างซอร์สโค้ดขึ้นมาด้วยกัน 2 ไฟล์คือ ไฟล์Dlg.h ที่เป็นไฟล์ส่วนหัวทำหน้าที่ในการประกาศคลาส และ ไฟล์Dlg.cpp ที่เป็นไฟล์ส่วนที่ใส่แสดงรายละเอียดของคลาส และเก็บฟังก์ชันต่างๆ จากนั้นจะถึงส่วนของการเพิ่มโค้ดโปรแกรมเพื่อให้ได้การทำงานที่เราต้องการ

โปรแกรมจะบันทึกภาพเป็นวิดีโอเก็บไว้ในฟอร์แมต (format) .AVI โดยการบันทึกภาพของโปรแกรมนี้จะเริ่มบันทึกในกรณีที่มีผู้บุกรุกเข้ามาในเขตพื้นที่รักษาความปลอดภัยซึ่งในส่วนของเซนเซอร์จะส่งสัญญาณเพื่อให้มีการบันทึกภาพเกิดขึ้นตามที่ได้กำหนดเวลาไว้ จากนั้นเซนเซอร์จะคอยตรวจสอบในลักษณะนี้ไปเรื่อยๆ

ไฟล์วิดีโอที่บันทึกจะถูกกำหนดเพิ่มปลายทางไว้ที่โพลเดอร์ราก (wwwroot) และทำการเก็บชื่อไฟล์และโพลเดอร์ไว้ในเทเบิลดาต้าเบส เพื่อสามารถเรียกดูได้จากเว็บเซิร์ฟเวอร์ โดยการติดต่อผ่าน ODBC และใช้ Microsoft Access เป็นฐานข้อมูลในการเก็บข้อมูล โดยการเพิ่มคลาสเข้าไปใหม่ในโปรเจกต์หรือโปรแกรมที่เราเขียนไว้ โดยกำหนดเป็น คลาส CRecordset และเลือกเทเบิลของดาต้าเบสที่ต้องการเอกสารนี้เป็นเอกสารที่ส่งวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ติดต่อ และกำหนดชื่อ DSN ให้ตรงกับที่เชื่อมต่อเทเบิลไว้กับ ODBC ซึ่งจะได้คลาสที่สร้างขึ้นเพื่อติดต่อกับฐานข้อมูลดังนี้

```
CmcSet2::CmcSet2(CDatabase* pdb)
    : CRecordset(pdb)
{
   //{{AFX_FIELD_INIT(CmcSet2)
    m_ID = 0;
    m_filename = _T("");
    m_nFields = 2;
   //}}AFX_FIELD_INIT
    m_nDefaultType = dynaset;
}

CString CmcSet2::GetDefaultConnect()
{
    return _T("ODBC;DSN=VideoDB1");
}

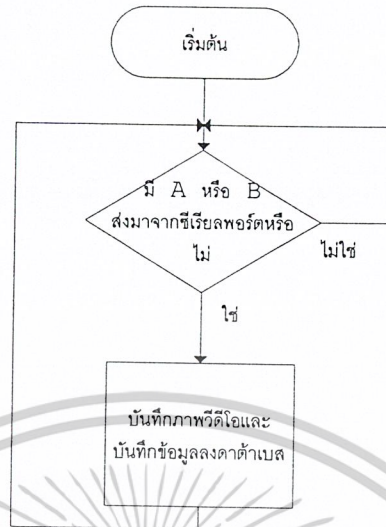
CString CmcSet2::GetDefaultSQL()
{
    return _T("[Table1]");
}
```

ซึ่งจะเห็นว่าคลาสจะติดต่อผ่าน ODBC ที่มี DSN เป็น VideoDB1 ติดต่อกับเทเบิลชื่อ Table1 ซึ่งประกอบด้วย คอลัมน์ ID และ filename

```
เราจะสามารถเรียกใช้หรือเพิ่มและลบข้อมูลในเทเบิลได้โดยการเรียกใช้ผ่านตัวแปร m_pSet ดังนี้
m_pSet.Open();
m_pSet.MoveLast();
m_pSet.AddNew();
m_pSet.m_filename= str4 + str2 + str3;
UpdateData(true);
m_pSet.Update();
m_pSet.Requery();
m_pSet.Close();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

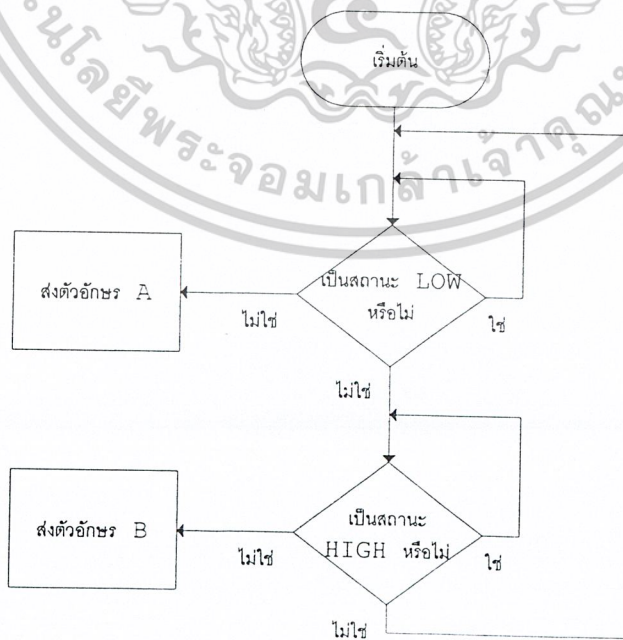
ในที่นี้จะเป็นการบันทึกชื่อไฟล์และโฟลเดอร์ ของไฟล์วีดิโอที่ทำการบันทึกไว้ในค่าเบส
 คอลัมน์ filename



รูปที่ 3.7 การทำงานส่วนโปรแกรมบันทึกภาพ

3.4 เซนเซอร์ผ่านพอร์ตอนุกรม

ส่วนของเซนเซอร์ที่ใช้ในการตรวจจับจะใช้วงจรถ่ายอินฟราเรดเซนเซอร์ที่เชื่อมต่อกับไมโครคอนโทรลเลอร์ เพื่อส่งสัญญาณเข้าพอร์ตอนุกรม โดยไมโครคอนโทรลเลอร์จะคอยทำการตรวจสอบสถานะของอินฟราเรดเซนเซอร์ว่ามีการเปลี่ยนแปลงหรือไม่ ถ้าไม่ก็จะวนตรวจสอบสถานะไปเรื่อยๆ ถ้าสถานะเปลี่ยน เช่น จากสถานะ Low เป็น High ไมโครคอนโทรลเลอร์ก็จะส่งค่าตัวอักษร A ผ่านพอร์ตอนุกรมเพื่อให้คอมพิวเตอร์รับค่าไปและนำไปใช้ในโปรแกรมส่วนต่อไปได้



รูปที่ 3.8 โฟลชาร์ตการทำงานของไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 การทำงานในส่วนเว็บเบราว์เซอร์

การทำงานในส่วนนี้จะเป็นการแสดงภาพ (Clip video) ที่ได้ทำการบันทึกไว้โดยเครื่องเซิร์ฟเวอร์ โดยจะสามารถเรียกดูภาพที่บันทึกไว้ย้อนหลังได้ตามวัน เวลา ที่มีการบันทึกเกิดขึ้น

และขั้นตอนในการบันทึกภาพของโปรแกรมบันทึกภาพวีดิโอ นั้น จะมีการบันทึกข้อมูลเป็นชื่อไฟล์และไฟล์เตอร์ไว้ในฐานข้อมูลด้วย ดังนั้นในส่วนของเว็บเบราว์เซอร์ก็เรียกใช้ข้อมูลจากฐานข้อมูลนี้ ในการแสดงไฟล์วีดิโอที่ได้บันทึกไว้ โดยผ่านการเชื่อมต่อฐานข้อมูลแบบ OLEDB ด้วยการเป็นสคริปต์ดังนี้

```
<%Set Conn = Server.CreateObject("ADODB.connection")
```

```
Conn.Provider="Microsoft.Jet.OLEDB.4.0"
```

```
Conn.Open Server.MapPath("db2.mdb")
```

```
sql = "Select * From Table1"
```

```
set rs = Server.CreateObject("ADODB.Recordset")
```

```
rs.Open sql,Conn,1,3%>
```

จากสคริปต์ข้างบนนี้ทำให้เราสามารถติดต่อกับฐานข้อมูลที่ต้องการได้แล้ว (ฐานข้อมูลอันเดียวกับที่โปรแกรมบันทึกภาพวีดิโอบันทึก) จากนั้นเราก็สามารถเรียกข้อมูลในเทเบิลออกมาดูได้โดยสร้างตารางดังนี้

```
<table width="57%" border="1" cellpadding="0" cellspacing="0" bordercolor="#ECE9D8">
```

```
<tr>
```

```
<td>ID</td>
```

```
<td>filename</td>
```

```
</tr>
```

```
<% do while not rs.EOF %>
```

```
<tr>
```

```
<td>
```

```
<% =rs("ID") %>
```

```
</td>
```

```
<td>
```

```
<a href="page1.asp?id=<%=RS("ID")%>"
```

```
target="_blank">
```

```
<% =rs("filename") %></a>
```

```
</td>
```

```
</tr>
```

```
<%
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
rs.movenext
```

```
loop
```

```
%>
```

```
</table>
```

ตารางที่แสดงออกมาจะเป็นตารางที่แสดงข้อมูลทั้งหมดจากฐานข้อมูล ซึ่งจะแสดงชื่อไฟล์ของวิดีโอและโพลเดอร์ที่เก็บไฟล์นั้น และในขณะเดียวกันก็มีการส่งค่า ID ที่ ชื่อไฟล์นั้นๆเพื่อให้สามารถเรียกดูภาพวิดีโอได้ตามวัน เวลา ที่แสดงในชื่อไฟล์

3.6 ขั้นตอนการแปลงข้อความให้อยู่ในรูปออกเต็ด

เนื่องจากการส่ง SMS โดย AT Command นั้นจะเป็นการส่งไปในรูปแบบของ PDU ที่ให้เราต้องแปลงข้อความที่เราต้องการส่งมาทำการแปลงให้อยู่ในรูปออกเต็ดก่อน ข้อความที่จะส่งคือ "ALERT ALERT" ประกอบด้วย 10 ตัวอักษรเรียกว่า Septets ที่ถูกแสดงเป็น 7 บิต ซึ่งเซปเต็ตนี้จะต้องถูกแปลงให้เป็นออกเต็ดเพื่อใช้ในการส่ง SMS

A	L	E	R	T	A	L	E	R	T
65	76	69	82	84	65	76	69	82	84
1000001	1001100	1000101	1010010	1010100	1000001	1001100	1000101	1010010	1010100
1000001	1001100	1000101	1010010	1010100	1000001	1001100	1000101	1010010	1010100

ตารางที่ 3.1 การแปลงตัวอักษรให้เป็นเซปเต็ต

เซปเต็ตแรก (A) จะถูกเปลี่ยนให้เป็นออกเต็ด โดยการนำบิตขวาสุดของเซปเต็ตที่สองมาต่อด้านซ้ายสุดของเซปเต็ตแรก ซึ่งจะได้เป็น $0 + 1000001 = 01000001$ ("41") และต่อมาเซปเต็ตที่สองก็ต้องการสองบิตขวาสุดของเซปเต็ตที่สาม มาต่อด้านซ้ายสุด เพื่อทำให้เป็น 8 บิตออกเต็ด เราจะทำกระบวนการนี้ไปเรื่อยๆจนไม่สามารถแปลงเป็นออกเต็ดได้ เมื่อแปลงเสร็จเราจะได้ค่าดังตาราง

01000001	01100110	01010001	01001010	00001101	00110010	10001011	01010010	101010
41	66	51	4A	0D	32	8B	52	2A

ตารางที่ 3.2 กระบวนการแปลงจากเซปเต็ตให้เป็นออกเต็ด

เมื่อเสร็จจะได้ 9 ออกเต็ดจากประโยค "ALERTALERT" คือ 41 66 51 4A 0D 32 8B 52 2A

3.7 การส่งอีเมลล์แบบ SMTP ผ่านทางเมล์เซิร์ฟเวอร์

ในการส่งจะใช้คอมโพเนนต์ที่มีรูปแบบการส่งแบบ SMTP โดยตัวอย่างของคำสั่งจะแสดงไว้ที่

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

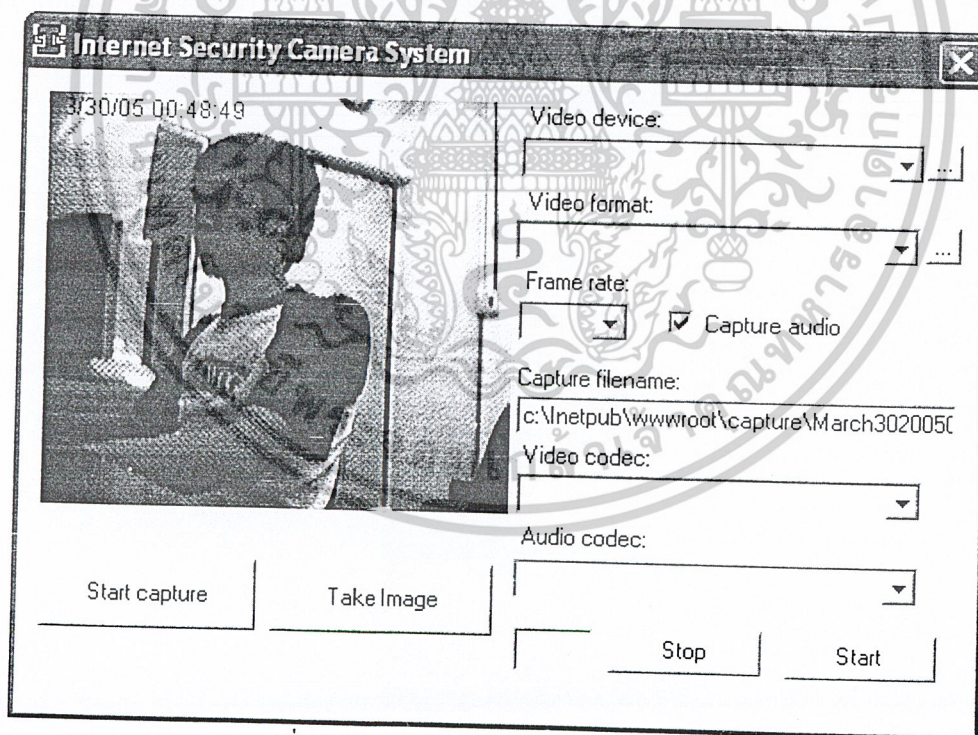
บทที่ 4

การทดลองและผลการทดลอง

การทดลองและผลการทดลองการทำงานในส่วนของการแสดงภาพวิดีโอจากกล้องวงจรปิด การบันทึกภาพวิดีโอ การบันทึกภาพนิ่ง การติดต่อเซิร์ฟเวอร์ผ่านพอร์ตอนุกรม การส่ง SMS การจัดเก็บข้อมูลในรูปแบบของดาต้าเบส และเว็บเบราว์เซอร์ที่ใช้ในการเรียกดูไฟล์ภาพวิดีโอที่ทำการบันทึกไว้

4.1 โปรแกรมบันทึกภาพวิดีโอ

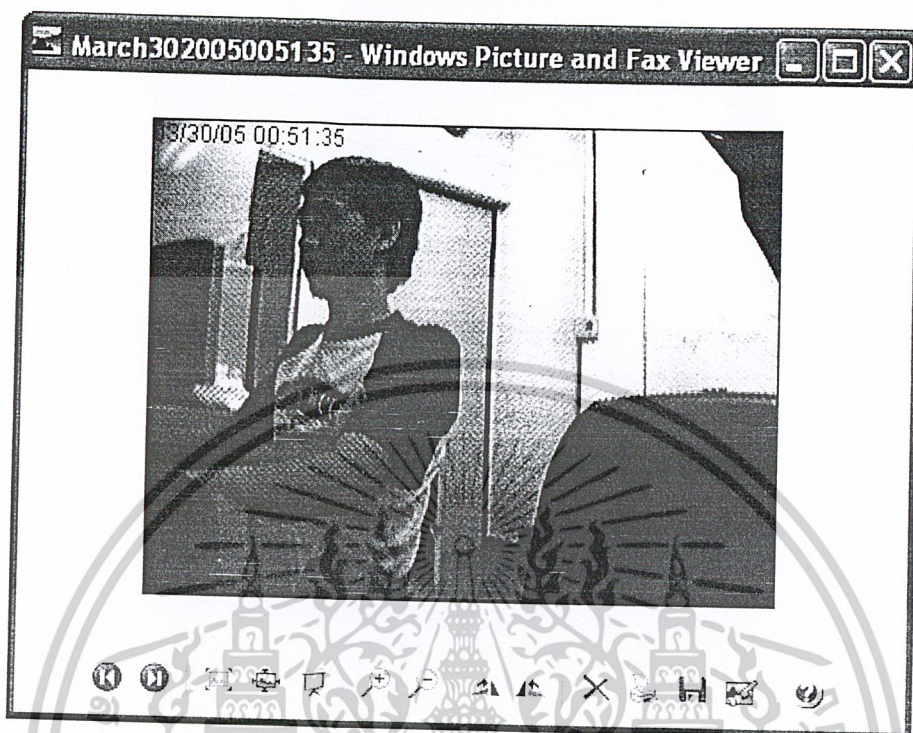
การทดลองการดึงภาพของโปรแกรมดึงภาพจากการ์ดแค็บเจอร์ เมื่อรันโปรแกรมนี้ขึ้นมา โปรแกรม Capture นี้ก็จะเริ่มทำการดึงภาพมาแสดง และบนหน้าจอวินโดว์ของโปรแกรมก็จะมีฟังก์ชันการทำงานประกอบด้วย Start Capture, Take Image, Capture filename เป็นต้น (ดังรูปที่ 4.1) เมื่อต้องการที่จะบันทึกภาพ คลิกที่ปุ่ม Start capture และเมื่อต้องการจะหยุดการบันทึกภาพก็ทำการคลิกปุ่ม Start capture อีกครั้ง โปรแกรมก็จะยกเลิกการบันทึกแต่ยังสามารถแสดงภาพจากกล้องวิดีโออยู่ ส่วนการบันทึกภาพนิ่ง คลิกที่ปุ่ม Take Image ทั้งการบันทึกภาพเป็นวิดีโอและการบันทึกภาพนิ่งจะทำการบันทึกเป็นชื่อไฟล์ เดือน วันที่ ปีค.ศ. ชั่วโมง นาที วินาที ที่เริ่มทำการบันทึก จากนั้นจะเก็บข้อมูลลงในฐานข้อมูลของระบบ



รูปที่ 4.1 ผลการดึงภาพโดยโปรแกรม Capture

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อกด Take Image โปรแกรมจะทำการ บันทึกเป็นไฟล์ jpg โดยการบันทึกก็จะเป็นการบันทึกเป็นไฟล์ที่มีชื่อเป็น เดือน วันที่ ปีค.ศ ชั่วโมง นาที วินาที เช่นเดียวกับการบันทึกภาพเป็นวีดีโอ



รูปที่ 4.2 ภาพนิ่งที่ถูกบันทึกโดยการกดปุ่ม Take Image

4.2 การเก็บข้อมูลลงดาต้าเบส

เป็นการเก็บข้อมูลลงดาต้าเบส โดยใช้โปรแกรมบันทึกภาพวีดีโอในการเก็บข้อมูลทั้งไฟล์วีดีโอและข้อมูลชื่อไฟล์และโฟลเดอร์ของไฟล์วีดีโอลงในดาต้าเบส เพื่อให้เว็บเซิร์ฟเวอร์นำไปใช้ในการเรียกดูได้โดยดาต้าเบสที่ใช้ก็คือ Microsoft Access ดังรูปที่4.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ID	filename
325	capture\March242005125703.avi
326	capture\March242005125752.avi
327	capture\March242005135344.avi
328	capture\March242005150551.avi
329	capture\March242005152509.avi
330	capture\March242005152652.avi
331	capture\March242005152757.avi
332	capture\March242005154618.avi
333	capture\March282005195728.avi
334	capture\March282005200137.avi
* (AutoNumber)	

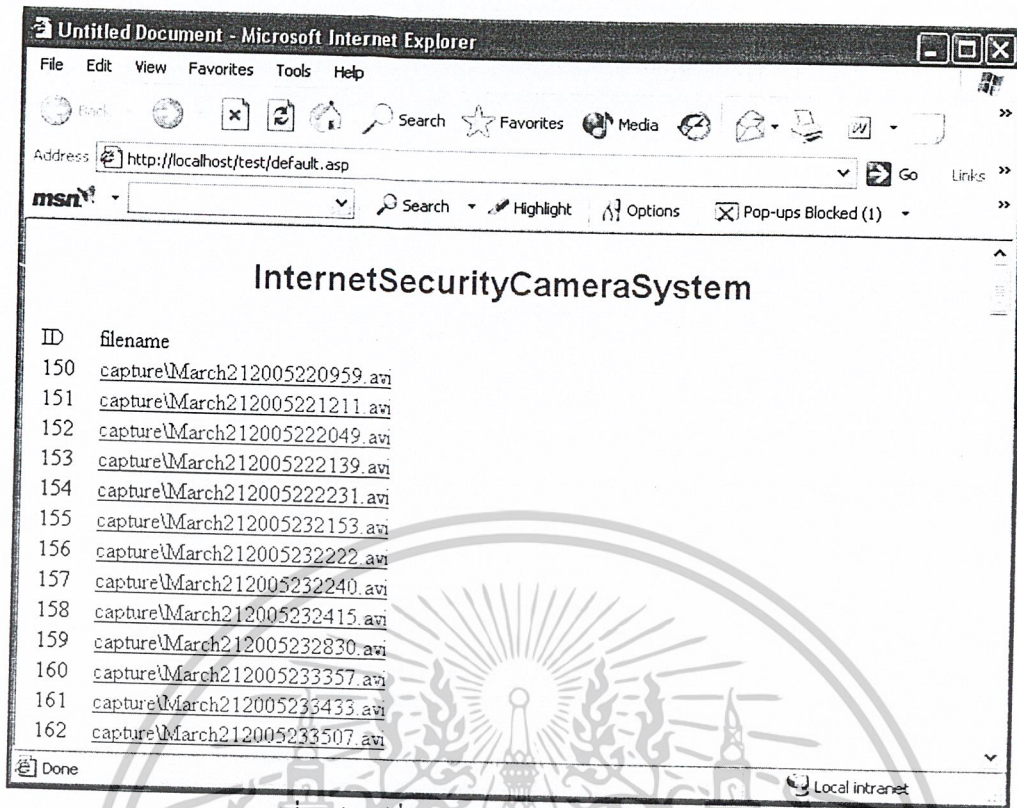
Record: 5 of 160

รูปที่ 4.3 คาดำเบสที่เก็บไว้ใน Microsoft Access

4.3 เว็บเบราว์เซอร์

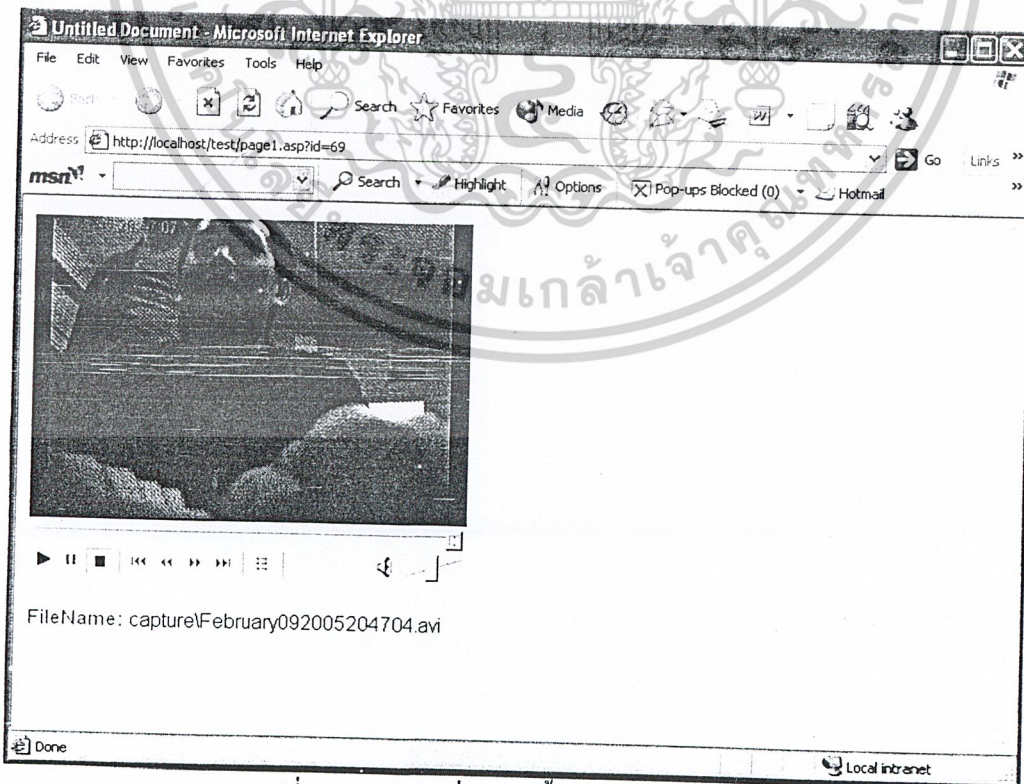
ไฟล์วิดีโอที่ทำการบันทึกจะถูกนำมาเก็บไว้ที่ฐานข้อมูลของระบบ พร้อมทั้งแสดงที่เว็บเบราว์เซอร์เพื่อให้ผู้ใช้งานสามารถดูไฟล์วิดีโอที่ทำการบันทึกได้ และเนื่องจากชื่อไฟล์ที่ทำการบันทึกนั้นเป็นแสดงวันเดือนปีและเวลาที่ทำการบันทึก(ดังรูปที่ 4.4) จึงทำให้ผู้ใช้งานสะดวกในการเข้าใจที่มาของไฟล์วิดีโอต่างๆที่ทำการบันทึกไว้เช่น February092005204704.avi จะหมายถึง ไฟล์วิดีโอนี้ได้ถูกบันทึกเมื่อวันที่ 9 เดือน กุมภาพันธ์ ปี 2005 เวลา 20 นาฬิกา 47 นาที 4 วินาที เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 ไฟล์ที่ทำการบันทึกไว้บนเว็บเบราว์เซอร์

เมื่อกดคลิกเข้าไปที่ไฟล์ capture\February092005204704.avi ก็จะได้แสดงภาพวิดีโอที่ทำการบันทึกไว้ โดยระยะเวลาที่ทำการบันทึกได้มีการกำหนดให้บันทึกเป็นระยะเวลาสั้น 7 วินาที (ดังรูปที่ 4.5)



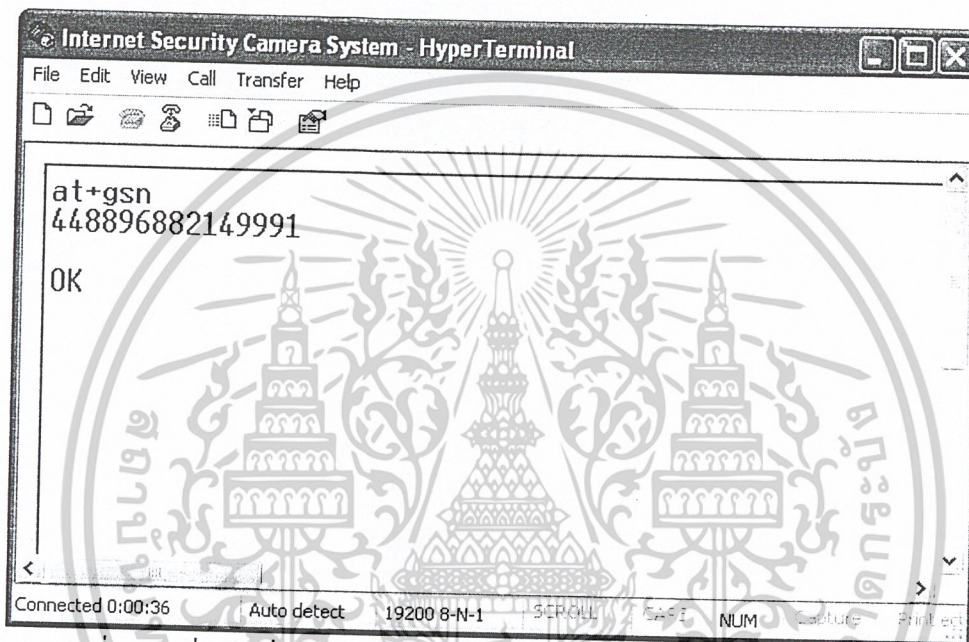
รูปที่ 4.5 ภาพวิดีโอที่ถูกเปิดขึ้นจากเว็บเบราว์เซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 โปรแกรมส่ง SMS

การทำงานในส่วนโปรแกรมส่ง SMS นี้ ได้มีการใช้งานชุดคำสั่ง AT เพื่อติดต่อกับโทรศัพท์เคลื่อนที่ผ่านทางพอร์ตอนุกรม โดยโปรแกรมในส่วนนี้เขียนด้วยภาษา Visual C++ version 6.0 ที่มีการทำงานคือส่งคำสั่ง AT เพื่อไปสั่งการให้โทรศัพท์เคลื่อนที่ทำการส่ง SMS ไปยังปลายทางที่กำหนดไว้ ตัวอย่างการใช้คำสั่ง AT เพื่อควบคุมการทำงานของโทรศัพท์เคลื่อนที่ผ่านพอร์ตอนุกรม ซึ่งทดสอบโดยการใช้ HyperTerminal ในการส่งคำสั่ง

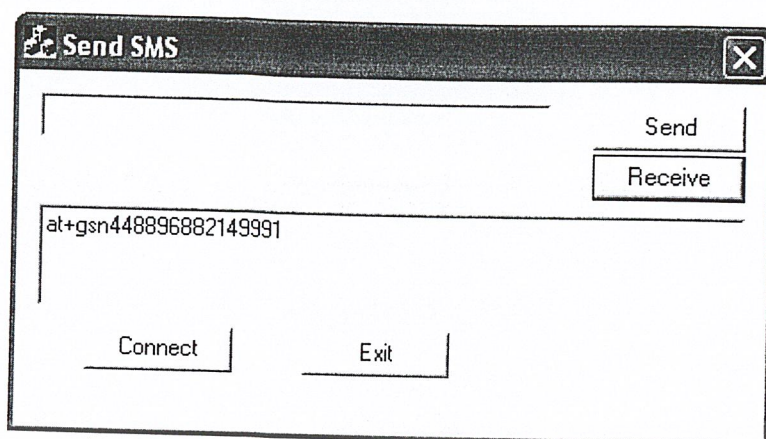
ตัวอย่างที่ 1 คำสั่ง AT+GSN เพื่อให้โทรศัพท์เคลื่อนที่ส่ง Output Serial Number (IMEI)



รูปที่ 4.6 คำสั่ง AT ที่กำหนดให้โทรศัพท์เคลื่อนที่ส่ง Output Serial Number (IMEI)

จากรูปจะเห็นได้ว่าเมื่อมีการใช้คำสั่ง AT+GSN จะมีการส่ง 448896882149991 ซึ่งตัวเลขดังกล่าวแสดงเลข IMEI ประจำเครื่องโทรศัพท์เคลื่อนที่

และเมื่อนำคำสั่ง AT+GSN มาใช้ในการสั่งการทำงานโทรศัพท์เคลื่อนที่โดยใช้โปรแกรมที่เขียนด้วย Visual C++ ดังรูปที่ 4.7 และกำหนดให้ช่องว่างเป็นช่องสำหรับแสดงคำสั่งที่โทรศัพท์เคลื่อนที่รับได้ และส่งกลับมา



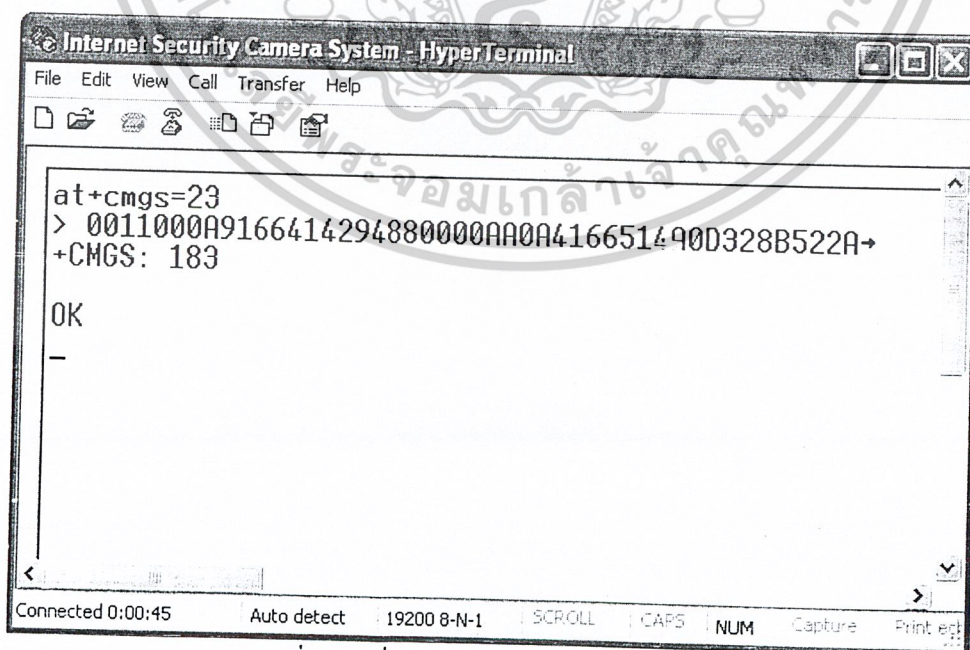
รูปที่ 4.7 โปรแกรมส่ง SMS แสดงค่าที่รับได้จากโทรศัพท์เคลื่อนที่

จะเห็นได้ว่าค่าที่ได้นั้นเป็นค่าเดียวกับค่าที่ทดลองใน HyperTerminal

โดยโครงงานฉบับนี้ได้กำหนดให้ส่ง SMS ที่มีข้อความว่า “ALERTALERT” ซึ่งคำสั่ง AT ที่ใช้นำมาจาก AT Command Set Siemens C35i จากภาคผนวกด้านหลัง และคำสั่งที่เลือกใช้คือ AT+CMGS=<Length><CR> PDU is given <ctrl-Z/ESC> โดย <Length> คือ ความยาวของ PDU <CR> คือ Carrier return PDU is given คือ ค่าที่ได้จากการแปลงดังที่กล่าวมาในบทที่ 3 ดังนั้นจะได้คำสั่ง AT ดังนี้ AT+CMGS=23

> 0011000A9166414294880000AA0A4166514A0D328B522A <ctrl-z>

เมื่อทดลองส่งคำสั่งผ่าน HyperTerminal จะได้ดังรูปที่ 4.8



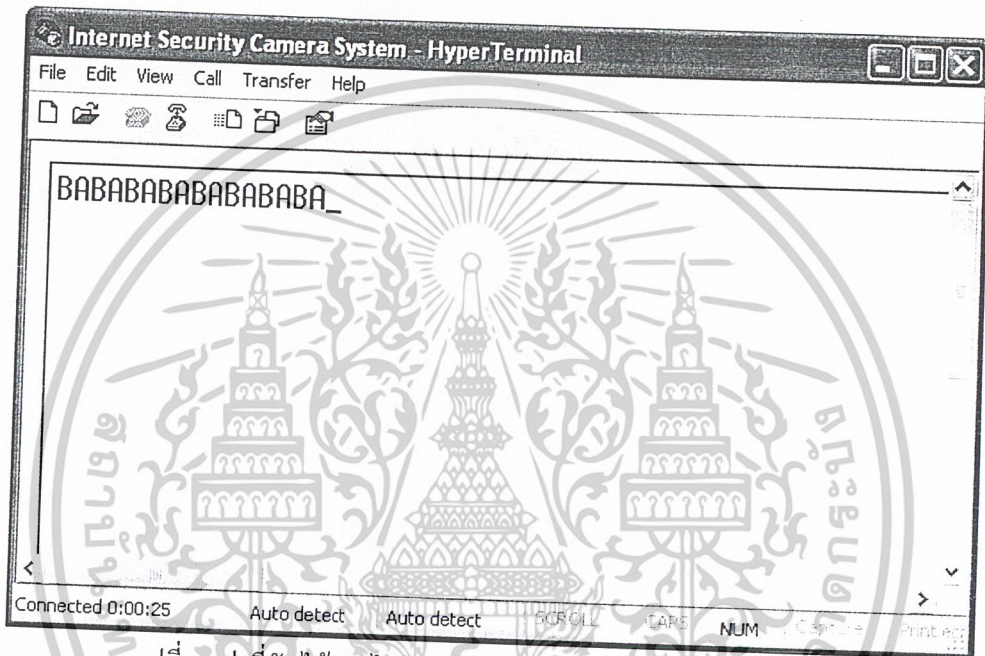
รูปที่ 4.8 คำสั่งส่ง SMS ผ่าน HyperTerminal

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และผลที่ได้คือโทรศัพท์เคลื่อนที่ปลายทางจะได้รับ SMS โดยมีข้อความว่า “ALERTALERT”

4.5 การรับค่าจากไมโครคอนโทรลเลอร์

จากโพลชาร์ตการทำงานของไมโครคอนโทรลเลอร์ จะมีการส่งค่า A หรือ B ผ่านพอร์ตอนุกรมเมื่อเซนเซอร์สามารถตรวจจับได้ ซึ่งเมื่อทดลองกับ HyperTerminal แล้วได้ผลดังนี้ (โดยการทดสอบจะมีการเพิ่มให้ไมโครคอนโทรลเลอร์ส่งค่าตัวอักษร B ออกไปก่อนจำนวนหนึ่งเพื่อดูว่าสามารถส่งค่ามาที่พอร์ตอนุกรมได้)



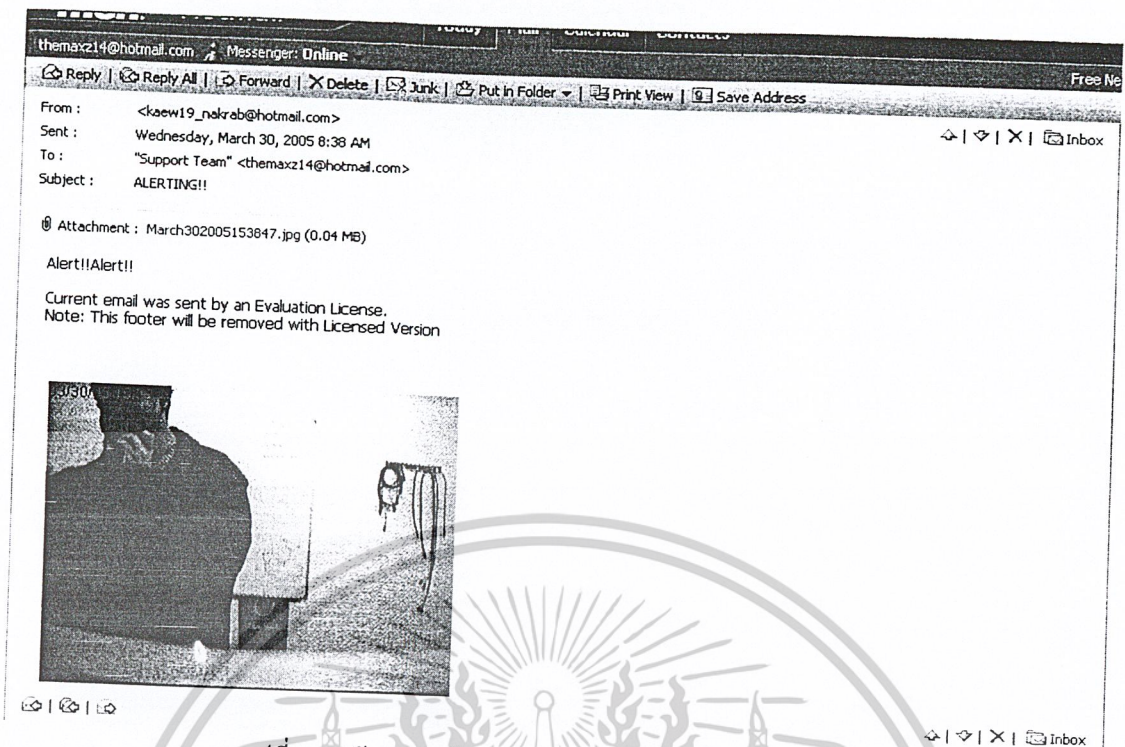
รูปที่ 4.9 ค่าที่รับได้จากไมโครคอนโทรลเลอร์โดยใช้ HyperTerminal

4.6 โปรแกรมส่งอีเมลล์

การทำงานในส่วนของโปรแกรมส่งอีเมลล์นี้จะทำการส่งข้อความและไฟล์รูปภาพเมื่อเซนเซอร์ทำการส่งสัญญาณผ่านทางเมล์เซิร์ฟเวอร์ ซึ่งการส่งอีเมลล์นี้จะเกิดขึ้นพร้อมๆกับการบันทึกภาพเป็นไฟล์วีดีโอ และการแจ้งเตือนทาง SMS ผ่านโทรศัพท์มือถือ

ซึ่งในโครงงานฉบับนี้ได้กำหนดให้โปรแกรมทำการส่งอีเมลล์ไปยังอีเมลล์ของทาง Hotmail ซึ่งอีเมลล์ที่ส่งไปนั้นได้กำหนดข้อความในส่วนของ Subject คือ ALERTING!! พร้อมทั้งส่งภาพนิ่งที่ได้ทำการบันทึกไว้เมื่อเซนเซอร์ส่งสัญญาณมายังโปรแกรมเพื่อทำการบันทึกภาพนิ่งเมื่อให้ผู้ควบคุมระบบนั้นได้ทราบว่าในขณะนั้นมีผู้บุกรุกเป็นใคร แล้วจากนั้นผู้ควบคุมก็สามารถเข้าไปดูไฟล์วีดีโอได้จากทางเว็บไซต์ของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



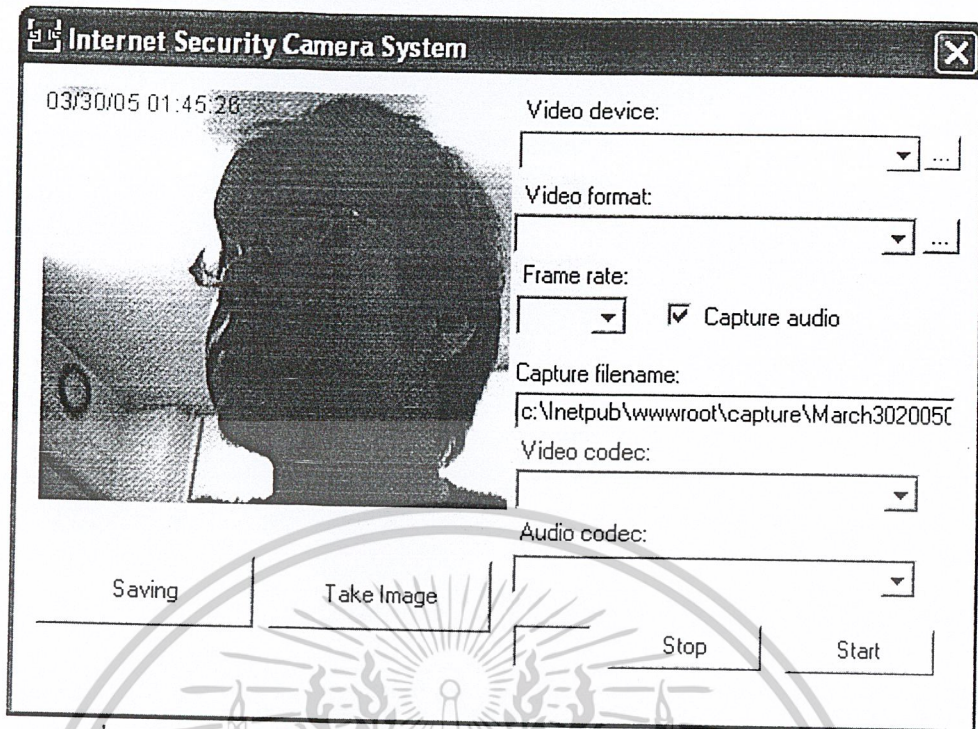
รูปที่ 4.10 ข้อความเตือนและไฟล์รูปภาพจากอีเมลที่ได้รับ

จากรูปที่ 4.10 นั้นแสดงถึงผลจากการส่งอีเมลจะเห็นได้ว่าแสดงภาพนิ่งที่ทำการบันทึกไว้เมื่อเซนเซอร์ส่งสัญญาณมายังโปรแกรม พร้อมทั้งระบุวันเดือนปีและเวลาที่ทำการบันทึกด้วยเพื่ออำนวยความสะดวกตรวจสอบในภายหลัง

4.7 โปรแกรม Internet Security Camera System

เมื่อนำโปรแกรมในส่วนของการแสดงภาพ โปรแกรมบันทึกภาพ ส่วนของการติดต่อกับดาต้าเบส มารวมกับส่วนของการรับค่าจากเซนเซอร์และส่วนของการแจ้งเตือนทาง SMS ผ่านโทรศัพท์มือถือ ก็จะได้โปรแกรม Internet Security Camera System ที่สามารถทำการบันทึกภาพได้เมื่อรับสัญญาณจากเซนเซอร์ ซึ่งในขณะที่ทำการบันทึกนั้น ที่ปุ่ม Start capture จะเปลี่ยนมาแสดงคำว่า Saving ซึ่งหมายถึงว่ากำลังบันทึกภาพอยู่และบันทึกเป็นเวลา 7 วินาที พร้อมกันนั้นจะทำการส่ง SMS คำว่า "ALERTALERT" ผ่านโทรศัพท์มือถือ Siemens C35 ไปยังปลายทางที่ได้กำหนดไว้ใน PDU (AT command) หลังจากนั้นโปรแกรมก็จะวนรอบรอรับค่าจากเซนเซอร์ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 โปรแกรม Internet Security Camera System ขณะกำลังบันทึกภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลและวิจารณ์

5.1 สรุปผลการทดลอง

จากระบบการทำงานโดยทั่วไปของโครงการนี้ จะประกอบด้วยส่วนต่างๆดังนี้

1. ส่วนของการแสดงภาพจากกล้องวงจรปิด
2. ส่วนของ sensor ที่คอยตรวจสอบผู้บุกรุก
3. ส่วนของการบันทึกภาพพร้อมทั้งวันเดือนปีและเวลาที่ทำการบันทึก
4. ส่วนของการแจ้งเตือนไปยังผู้ตรวจสอบทาง SMS และผ่านทาง E-mail
5. ส่วนของ Web site เพื่อที่ผู้ตรวจสอบสามารถตรวจสอบผ่านทาง Internet ได้

โดยในแต่ละส่วนนั้นได้มีการทำงานที่ต่อเนื่องกัน เช่นส่วนของโปรแกรมการบันทึกภาพวีดีโอนั้นจะสามารถเริ่มงานโดยอัตโนมัติได้ก็ต่อเมื่อส่วนของเซนเซอร์สามารถทำงาน นอกจากนั้นยังส่งผลต่อส่วนของการแจ้งเตือนผ่าน SMS และ E-mail ซึ่งการทำงานของระบบนี้จะดำเนินไปอย่างอัตโนมัติเสมือนเป็นผู้ช่วยผู้ควบคุมระบบ นอกจากนี้ยังได้มีการพัฒนาซอฟต์แวร์ที่ใช้ในการบันทึกภาพให้สามารถเปลี่ยนไปตามวันเดือนปีและเวลาที่เริ่มทำการบันทึกภาพ ทำให้ง่ายในการจัดเก็บข้อมูลและตรวจสอบในภายหลัง

โปรแกรมของโครงการฉบับนี้นั้นมีประสิทธิภาพเป็นที่น่าพอใจสามารถนำไปพัฒนาเพื่อประยุกต์ในการใช้งานจริงได้หากมีการเปลี่ยนมาใช้กล้องวงจรปิดที่มีมาตรฐานสูงกว่านี้ โครงการนี้เหมาะสำหรับการเฝ้าระวังหรือรักษาความปลอดภัยพื้นที่ที่ต้องการความปลอดภัยค่อนข้างสูงและคอยช่วยเหลือผู้ควบคุมระบบได้อีกทางหนึ่ง

5.2 บทวิจารณ์

เนื่องจากโครงการนี้เป็นโครงการเกี่ยวกับระบบรักษาความปลอดภัยโดยกล้องวงจรปิดผ่านเครือข่าย internet ซึ่งจะพบว่าเกิดปัญหาในหลายๆด้านในขั้นตอนการดำเนินงาน ทั้งในด้านของราคาซึ่งอุปกรณ์แต่ละชนิดที่ใช้กับโครงการนั้นมีราคาค่อนข้างสูงไม่ว่าจะเป็น Capture card และกล้องวงจรปิด เป็นต้น ปัญหาในส่วนของเซนเซอร์ที่มีรัศมีการตรวจจับที่ไม่ไกลมากเนื่องจากเซนเซอร์ที่มีคุณภาพดีนั้นมีราคาค่อนข้างสูง อีกทั้งปัญหาในด้านของการศึกษาค้นคว้าข้อมูล ในเรื่องการเขียนโปรแกรมบันทึกภาพวีดีโอที่ใช้ภาษา C++ การใช้งานคำสั่ง AT เพื่อส่งการทำงานโทรศัพท์เคลื่อนที่ในการส่ง SMS ซึ่งในการเขียนโปรแกรมนั้นเป็นเรื่องใหม่สำหรับสมาชิกในกลุ่ม จึงต้องใช้ระยะเวลาในการศึกษาเรื่องดังกล่าว และได้แก้ปัญหาด้วยการศึกษจากหนังสือต่างๆ และด้วยการขอคำแนะนำจากผู้ที่มีความรู้ทางด้านนี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. โปรแกรมส่วนติดต่อดูฐานข้อมูล

```
m_pSet.Open();
m_pSet.MoveLast();
m_pSet.AddNew();
m_pSet.m_filename= str4 + str2 + str3;
UpdateData(true);
m_pSet.Update();
m_pSet.Requery();
m_pSet.Close();
```

2. โปรแกรมส่งSMS

```
char re[5];
char smsBuf[20] = "atd(065418400);\r\n";
CString at = "AT+CMGC=23";
CString pdu = "0011000A9166913389220000AA0A4166514A0D328B522A";
char cr = 0x0D;
char lf = 0x0A;
char z = 0x1A;
CString sms1;
CString sms2;
sms1 = at + cr + lf;
sms2 = pdu + z;
port2.Write(sms1, strlen(sms1));
port2.Flush();
Delay(2);
if (port2.Read(re,1))
{
    Port2.Write(sms2, strlen(sms2));
    Port2.Flush();
}
```

3. โปรแกรมส่วนส่งอีเมลล์

```
::CoInitialize( NULL );
IMailPtr oSmtip = NULL;
oSmtip.CreateInstance("AOSMTP.Mail");

oSmtip->ServerAddr = _bstr_t( "mail.kmitl.ac.th" );
//send email via DNS lookup, ANSMTP lookups SMTP server automatically.

oSmtip->FromAddr = _bstr_t( "kaew19_nakrab@hotmail.com" );

oSmtip->AddRecipient( _bstr_t("Support Team"),
    _bstr_t("themaxz14@hotmail.com"), 0);

oSmtip->Subject = _bstr_t("ALERTING!!");
oSmtip->BodyText = _bstr_t("Alert!!Alert!!");

CString m_strAttachment = str1+str2+str4;
CString fileName = m_strAttachment;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

oSmtmp->AddAttachment( _bstr_t((LPCSTR)fileName));
if( oSmtmp->SendMail() == 0 )
    cout << "Message delivered!" << endl;
else
    cout << (const char*)(oSmtmp->GetLastErrDescription()) << endl;

```

4. โปรแกรมส่วนส่งค่าจากไมโครคอนโทรลเลอร์

```

ORG 0000H
MOV pcon, #00h
MOV SCON, #50h
MOV tmod, #20h
MOV th1, #0fdh
SETB TR1
check:
    JNB P1.7, LOW

HIGH:    JB P1.7, HIGH
MOV A, #42h
MOV SBUF, A
JNB TI, $
CLR TI
JNB P1.7, $
SJMP check

LOW:    JNB P1.7, LOW
MOV A, #41h
MOV SBUF, A
JNB TI, $
CLR TI
JB P1.7, $
SJMP check
END

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Features

- Compatible with MCS-51™ Products
- 8 Kbytes of In-System Reprogrammable Flash Memory
Endurance: 1,000 Write/Erase Cycles
- Fully Static Operation: 0 Hz to 24 MHz
- Three-Level Program Memory Lock
- 256 x 8-Bit Internal RAM
- 32 Programmable I/O Lines
- Three 16-Bit Timer/Counters
- Eight Interrupt Sources
- Programmable Serial Channel
- Low Power Idle and Power Down Modes

Description

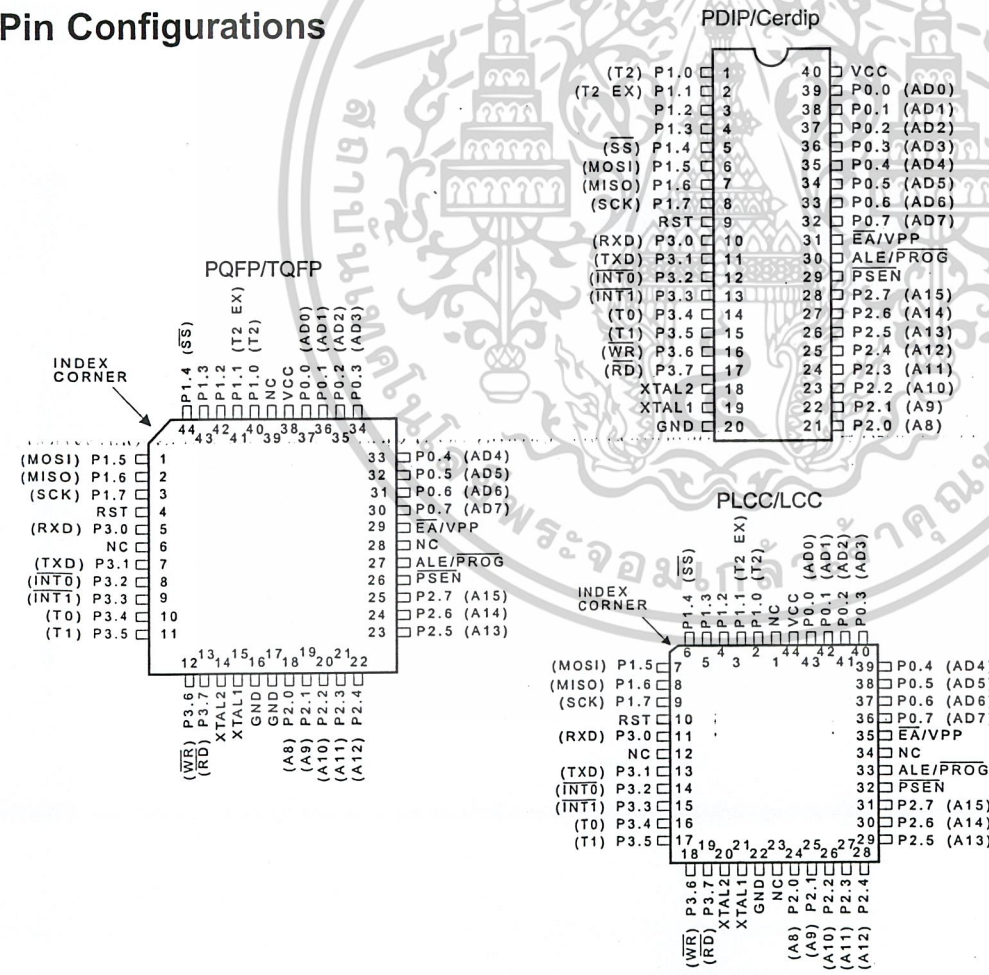
The AT89C52 is a low-power, high-performance CMOS 8-bit microcomputer with 8 Kbytes of Flash programmable and erasable read only memory (PEROM). The device is manufactured using Atmel's high density nonvolatile memory technology and is compatible with the industry standard 80C51 and 80C52 instruction set and pinout. The on-chip Flash allows the program memory to be reprogrammed in-system or by a conventional nonvolatile memory programmer. By combining a versatile 8-bit CPU with Flash on a monolithic chip, the Atmel AT89C52 is a powerful microcomputer which provides a highly flexible and cost effective solution to many embedded control applications.

8-Bit Microcontroller with 8 Kbytes Flash

AT89C52

(continued)

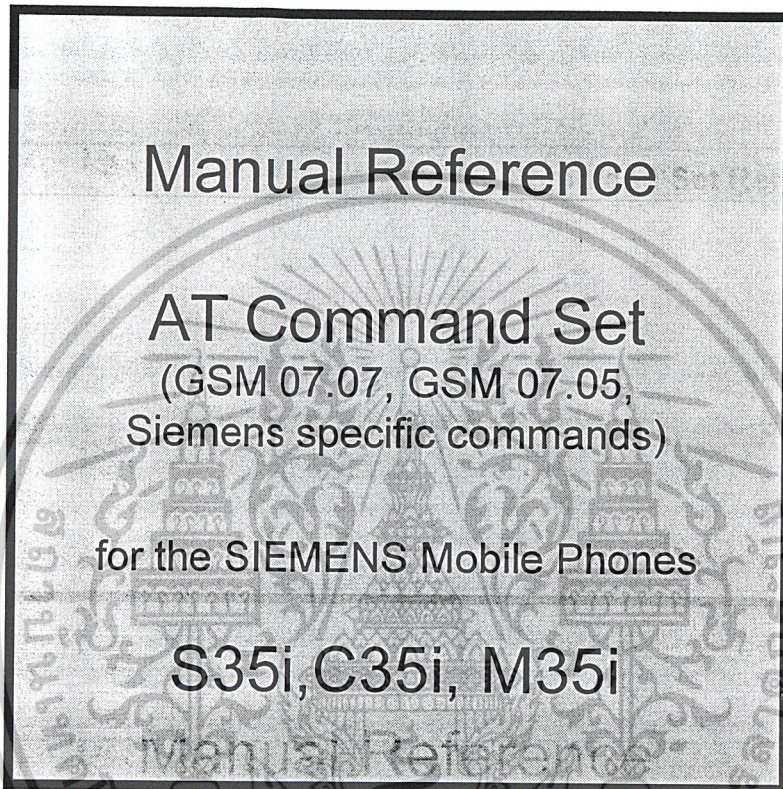
Pin Configurations



0313E



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The command descriptions or example sequences in this document imply no liability or warranty in any way. The author therefore will take no responsibility and will accept no liability which results of using the content of this document in any way.

All rights reserved. No part of this work covered by the copyrights hereof may be reproduced or copied in any form or by any means (graphic, electronic, or mechanical, including photocopying, taping, or information storage and retrieval systems) without written permission of the publisher.

Revisions Overview

Date	Version	Name	Description of revision
15-03-2000	1.0	Kel	created

Table of Contents

Revisions Overview.....	2
Table of Contents.....	2
1. Software Interface.....	3
1.1. OVERVIEW OF THE SUPPORTED AT COMMAND SET.....	3
1.2. AT COMMAND SET.....	5
1.2.1. Hayes-Standard Commands.....	6
1.2.2. Acknowledgments for Normal Data Communication.....	6
1.3. AT COMMANDS AND RESPONSES ACCORDING TO GSM 07.07 AND GSM 07.05.....	7
1.3.1. AT Cellular Commands According to GSM 07.07.....	7
1.3.2. AT Commands According to GSM 07.05 for SMS.....	26
1.3.3. User-Defined Commands for Controlling the GSM Mobile Phone.....	33
1.3.4. Summary of All Unexpected Messages.....	44
APPENDIX A.....	45
Factory settings made by AT&F.....	45
Features of the Telephone-Book Memory.....	45
Writing to the FDN Phonebook / FDN Replacement.....	47
Special hints for using +CPBR/+CPBW command.....	48
APPENDIX B.....	50
Example for creating/interrogation of an organizer entry.....	50

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. Software Interface

1.1. Overview of the Supported AT Command Set

Page	Commands 07.07	Function
7	AT+CGMI	Issue manufacturer ID code
7	AT+CGMM	Issue model ID code
7	AT+CGMR	Output the GSM telephone version
8	AT+CGSN	Output the serial number (IMEI)
8	AT+GSN	Output the serial number (IMEI)
8	AT+CHUP	Terminate call
8	AT+CEER	Query the reason for disconnection of last call
9	AT+CREG	Network registration
9	AT+COPS	Commands concerning selection of network operator
10	AT+CLCK	Switch locks on and off
10	AT+CPWD	Change password to a lock
11	AT+CLIP	Display telephone number of calling party
11	AT+CCFC	Call forwarding
12	AT+CHLD	Call hold and multiparty
12	AT+CPAS	Query the telephone status
13	AT+CPIN	Enter PIN and query lock
13	AT+CBC	Battery charge
14	AT+CSQ	Output signal quality
14	AT+CPBS	Select a telephone book
15	Fehler! Kein gültiges Resultat für Tabelle.	Read a telephone-book entry
15	AT+CPBW	Write a telephone-book entry
16	AT+CMEE	Expanded error messages according to GSM 07.07
17	AT+VTS	Send a DTMF tone
18	AT+VTD	Set duration of a DTMF tone
18	AT+WS46	Select wireless network
18	AT+CSCS	Select TE character set
19	AT+CAOC	Advice of charge
19	AT+CSSN	Supplementary service notifications
20	AT+CRSM	Restricted SIM access
20	AT+CIMI	Output of IMSI
21	AT+CACM	Accumulated call meter
21	AT+CAMM	Accumulated call meter maximum
22	AT+CLCC	List Current Calls
23	AT+CCLK	Clock
23	AT+COPN	Read operator names
23	AT+CPUC	Price per unit and currency table
24	AT+CALM	Alert sound mode
24	AT+CRSL	Ringer sound level
24	AT+CLVL	Loudspeaker volume level
24	AT+CMUT	Mute control
25	AT+CVIB	Vibrator mode

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Page	Commands 07.05	Function
26	AT+CSMS	Selection of message service
27	AT+CPMS	Selection of SMS memory
27	AT+CMGF	SMS format
28	AT+CSCA	Address of the SMS service center
28	AT+CNMI	Display new incoming SMS
29	AT+CNMA	Acknowledgment of a short message directly output
30	AT+CMGL	List SMS
31	AT+CMGR	Read in an SMS
31	AT+CMGS	Send an SMS
31	AT+CMSS	Send an SMS from the SMS memory
32	AT+CMGW	Write an SMS to the SMS memory
32	AT+CMGD	Delete an SMS in the SMS memory
32	AT+CSCB	Select cell broadcast messages
32	AT+CMGC	Send an SMS command

Page	Siemens-specific commands	Function
33	AT^SPBS	Select a telephone book (including Siemens-specific books)
33	AT^SDLD	Delete the "last number redial" memory"
34	AT^SPBC	Seek the first entry in the sorted telephone book which begins with the selected (or next available) letter
34	AT^SPBG	Read entry from the sorted telephone book via the sorted index
35	AT^SLCK	Switch locks (including user-defined locks) on and off
35	AT^SPWD	Change password to a lock (including user-defined locks)
36	AT^SACM	Output ACM (accumulated call meter) and ACMmax
36	AT^SPLM	Read the PLMN
36	AT^SPLR	Read an entry from the preferred-operator list
36	AT^SPLW	Write an entry to the preferred-operator list
37	AT^SCNI	Output call number information
37	AT^SNFV	Set the volume
37	AT^SNFS	Select NF hardware
38	AT^SRTC	Set the ringing tone
38	AT^SCID	Output card ID
38	AT^SCKS	Output SIM card status
39	AT^SPIC	Output PIN counter
39	AT^SMGO	SMS overflow indicator
40	AT^SMGL	List SMS (without status change from unread to read)
40	AT^SMGR	Read SMS record without Changing unread->read
40	AT^SMSO	Switch device off
41	AT^SLNG	Language settings
41	AT^SSTK	SIM Toolkit
41	AT^SBNW	Binary Write
43	AT^SBNR	Binary Read

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2. AT Command Set

Remote control operation of the GSM mobile telephone runs via a serial interface (data cable or infrared connection), where AT+C commands according to ETSI GSM 07.07 and GSM 07.05 specification as well as several manufacturer specific AT commands are available. These commands are described in more detail later on.

The modem guideline V.25ter applies to the sequence of the interface commands. According to this guideline, commands should begin with the character string "AT" and end with "<CR>" (= 0x0D). The input of a command is acknowledged by the display of "OK" or "ERROR". **A command currently in process is interrupted by each additional character entered.** This means that you should not enter the next command until you have received the acknowledgment; otherwise the current command is interrupted.

The commands supported are listed in the following tables:



1.2.1. Hayes-Standard Commands

The Hayes-standard commands correspond to the commands of AT Hayes-compatible modems.

Command	Function
A/	Repeat last command
AT...	Prefix for all other commands
ATA	Accept call
ATD<str>;	Dial the dialing string <str> with the voice utility Valid dial modifiers: "T" (tone dialing), "P" (pulse dialing) is ignored. The character ";" is important, for this tells the phone that the call should be set up with the voice utility. Otherwise an attempt is made to set up a data call, which the phone immediately acknowledges with "ERROR". The dial command responds with OK to the user right after starting a voice call. Other behavior like *# sequences in the dial command and also data calls remain unchanged.
ATD><n>;	Dial the telephone number from the current telephone book location number <n> The telephone book is selected with the command at+cpbs (or at^spbs).
ATD><mem><n>;	Dial the telephone number from the telephone book <mem> location number <n>
ATDL	Dial last telephone number
ATE0	Deactivate command echo
ATE1	Activate command echo
ATH[0]	Separate connection
ATQ0	Display acknowledgments
ATQ1	Suppress acknowledgments
ATV0	Output acknowledgments as numbers
ATV1	Output acknowledgments as text
AT&F[0]	Reset to factory profile
ATZ	Set to default configuration
AT+GCAP	Output the capabilities list

1.2.2. Acknowledgments for Normal Data Communication

Response	Numeric	Meaning
OK	0	Command executed, no errors
RING	2	Ring detected
NO CARRIER	3	Link not established or disconnected
ERROR	4	Invalid command or command line too long
NO DIALTONE	6	No dial tone, dialing impossible, wrong mode
BUSY	7	Remote station busy

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3. AT Commands and Responses According to GSM 07.07 and GSM 07.05

According to GSM, it is possible to execute an AT command in various forms.

Test command	AT+CXXX=?	The telephone responds by sending the list of parameters and value ranges; these can be set using the affiliated Write command or by means of internal processes.
Read command	AT+CXXX?	This command tells you the current value setting of the parameter(s).
Write command	AT+CXXX=<...>	This command is used to set parameters that can be set.
Execute command	AT+CXXX	The Execute command reads non-settable parameters which are influenced by internal processes in the telephone.

1.3.1. AT Cellular Commands According to GSM 07.07

AT+CGMI	Issue manufacturer ID code
Test command AT+CGMI=?	Response OK
Execute command AT+CGMI	Response <manufacturer> Parameter <manufacturer> Name of manufacturer (SIEMENS) Important: There is a leading output prefix +CGMI in models before the S25.

AT+CGMM	Issue model ID code
Test command AT+CGMM=?	Response OK
Execute command AT+CGMM	Response <model> Parameter <model> Name of telephone (MOBILE) Important: There is a leading output prefix +CGMM in models before the S25.

AT+CGMR	Output the GSM telephone version
Test command AT+CGMR=?	Response OK
Execute command AT+CGMR	Response <revision> Parameter <revision> Version of the telephone software Important: There is a leading output prefix +CGMR in models before the S25.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OK/ERROR/+CME ERROR

<p>AT+CLCK</p>	<p>Switch locking on and off Revision to GSM 07.07 according to CR TDOC ETSI/SMG4 187/96</p>
<p>Test command AT+CLCK=?</p>	<p>Response +CLCK: (list of supported <fac>s) OK/ERROR/+CME ERROR</p> <p>Parameter <fac></p> <ul style="list-style-type: none"> "CS" Keyboard lock "PS" Phone locked to SIM (device code) "SC" SIM card (PIN) "FD" FDN lock "AO" BAOC (bar all outgoing calls) "OI" BOIC (bar outgoing international calls) "OX" BOIC-exHC (bar outgoing international calls except to home country) "AI" BAIC (bar all incoming calls) "IR" BIC-Roam (bar incoming calls when roaming outside the home country) "AB" All Barring services "AG" All outgoing barring services "AC" All incoming barring services
<p>Write command AT+CLCK=<fac>, <mode>[, <passwd> [, <class>]]</p>	<p>Parameter <fac> See Test command <mode> 0 Cancels lock 1 Activates lock 2 Queries lock status <passwd> Password <class> 1 Voice 2 Data 4 Fax 7 All classes (default value)</p> <p>Response If <mode>=2 and command is successful +CLCK: <status>[, <class1>[<CR><LF> +CLCK: <status>, class2....]]</p> <p>Parameter <status> 0 Off 1 On</p> <p>OK/ERROR/+CME ERROR</p>

<p>AT+CPWD</p>	<p>Change password to a lock</p>
<p>Test command AT+CPWD=?</p>	<p>Response +CPWD: list of supported (<fac>, <pwdlength>) OK/ERROR/+CME ERROR</p> <p>Parameter <fac> "P2" PIN2 otherwise See Test command for AT+CLCK command, without "FD" <pwdlength> Password length</p>
<p>Write command AT+CPWD= <fac>, <oldpwd>, <newpwd></p>	<p>Parameter <fac> See Test command for AT+CLCK command <oldpwd>, <newpwd> Old and new password</p> <p>Response OK/ERROR/+CME ERROR</p>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อจุดประสงค์เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT+CLIP		Display telephone number of calling party	
Test command AT+CLIP=?	Response +CLIP: (list of supported <n>s) OK/ERROR/+CME ERROR	Parameter <n>	0 Suppresses the unexpected messages 1 Displays the unexpected messages
Read command AT+CLIP?	Response +CLIP: <n>, <m> OK/ERROR/+CME ERROR	Parameter <n> <m>	See Test command 0 CLIP not booked 1 CLIP booked 2 Unknown
Write command AT+CLIP=[<n> >]	Parameter <n>	See Read command	
	Response OK/ERROR/+CME ERROR		
	Unexpected message +CLIP: <num>, <type>	Telephone number of caller	

AT+CCFC		Call forwarding	
Test command AT+CCFC=?	Response +CCFC: (list of supported <reas>s) OK/ERROR/+CME ERROR	Parameter <reas>	0 Always 1 If busy 2 If no answer 3 If not available 4 All reasons (0-3) 5 All conditional reasons (1-3)
Write command AT+CCFC=<reas>, <mode>[, <num> >[, <type>[, <class> >[, <time>]]]]	Parameter <reas> <mode> <num> <type> <class> <time>	See Test command 0 Deactivate 1 Activate 2 Query 3 Install 4 Delete Telephone number Type of telephone number 1 Voice 2 Data 4 Fax 7 All classes 1-30 Time, rounded to a multiple of five seconds	Response If <mode>=2 and command is successful +CCFC: <status>, <class1>[, <num>, <type>[, <time>]]][<CR><LF>+CCFC:] OK/ERROR/+CME ERROR Parameter <status>
	Response If <mode>=2 and command is successful +CCFC: <status>, <class1>[, <num>, <type>[, <time>]]][<CR><LF>+CCFC:] OK/ERROR/+CME ERROR	Parameter <status>	0 Not active 1 Active

AT+CHLD		Call hold and multiparty	
Test command AT+CHLD=?	Response +CHLD: (list of supported <n>s) OK/ERROR/+CME ERROR		
Write command AT+CHLD= [<n>]	Parameter <n>	0	Terminates all held calls or sets UDUB (User Determined User Busy) for a waiting call
		1	Terminates all active calls (if there are any) and accepts the other call (waiting call or held call)
		1X	Terminates call number X (X= 1-7)
		2	Puts all active calls on hold (if there are any) and accepts the other call (waiting call or held call) as active
		2X	Puts all active calls except call X (X= 1-7) on hold
		3	Connects the call put on hold to the active call
	For terminating	Terminating all calls except waiting calls is done with "AT+CHUP"	
	Note:	Command scope depends on the SIM clearing and/or on the network support	
	Response OK/ERROR/+CME ERROR		

AT+CPAS		Query the telephone status	
Test command AT+CPAS=?	Response +CPAS: (list of supported <pas>s) OK/ERROR/+CME ERROR		
	Parameter <pas>	0	Ready
		3	Incoming call (phone is ringing)
		4	Call is active
Execute command AT+CPAS	Response +CPAS: <pas> OK/ERROR/+CME ERROR		
	Parameter <pas>	See Test command	
	OK/ERROR/+CME ERROR		

AT+CPIN	Enter PIN and query lock
Test command AT+CPIN=?	Response OK
Read command AT+CPIN?	Response +CPIN: <code> OK/ERROR/+CME ERROR Parameter <code> READY No further input necessary SIM PIN SIM PIN input necessary SIM PUK SIM PUK input necessary PH-SIM PIN Device-code (theft protection) input necessary PH-SIM PUK Device-code PUK (theft protection) input necessary SIM PIN2 PIN2, e.g. for editing the FDN book; only possible if previous command was acknowledged with +CME ERROR:17 SIM PUK2 Only possible if previous command was acknowledged with error +CME ERROR:18 The required error message can (must) be provoked by an attempted Write command.
Write command AT+CPIN=<pin> [,<new pin>]	Parameter <pin> Password for appropriate lock; if the lock is a PUK, then a <new pin> is necessary. <new pin> New password for the lock Response OK/ERROR/+CME ERROR

AT+CBC	Battery charge
Test command AT+CBC=?	Response +CBC: (list of supported <bcs>s),(list of supported <bcl>s) OK/ERROR/+CME ERROR Parameter <bcs> 0 ME is supplied from battery 1 ME has battery but is not supplied from there 2 ME has no battery connected 3 Error <bcl> 0 Battery is flat, but no more actions possible 1-100 charge in per cent
Execute command AT+CBC	Response +CBC: <bcs>,<bcl>

AT+CSQ	Output signal quality
Test command AT+CSQ=?	Response +CSQ: (list of supported <rssi>s), list of supported <ber> OK/ERROR/+CME ERROR Parameter <rssi> Reception level: 0 -113 dBm or less 1 -111 dBm 2-30 -109 to -53 dBm 31 -51 dBm or more 99 Unknown <ber> Bit error rate: 0-7 Like RXQUAL values from Table GSM 05.08 in Section 8.2.4 99 Unknown
Execute command AT+CSQ	Response +CSQ: <rssi>, <ber> OK/ERROR/+CME ERROR Parameter <rssi> See Test command <ber> See Test command

AT+CPBS	Select a telephone book
Test command AT+CPBS=?	Response +CPBS: (list of supported <sto>s) OK/ERROR/+CME ERROR Parameter <sto> "FD" SIM fix-dialing phonebook "SM" SIM phonebook "ME" ME phonebook "DC" ME Dialed Calls List "ON" SIM (or ME) own numbers (MSISDNs) list "LD" SIM last-dialling phonebook "MC" ME missed (unanswered received) calls list "RC" ME received calls list *For description of telephone-book features, see Appendix A Note: "DC" and "LD" are never both available.
Read command AT+CPBS?	Response +CPBS: <sto> OK/ERROR/+CME ERROR Parameter <sto> See Test command
Write command AT+CPBS=<sto>	Parameter <sto> See Test command Response OK/ERROR/+CME ERROR

AT+CPBR		Read a telephone-book entry
Test command AT+CPBR=?	Response +CPBR: (list of supported <index>s), <nlength>, <tlength> OK/ERROR/+CME ERROR	Parameter <index> Location number <nlength> Max. length of telephone number <tlength> Max. length of text corresponding to the number
Write command AT+CPBR= <index1> [,<index2>]	Response +CPBR: <index1>, <number>, <typ>, <text>[<CR><LF> +CPBR: +CPBR: <index2>, <number>, <typ>, <text>] OK/ERROR/+CME ERROR	Parameter <index1> Location number where the read of the entry starts <index2> Location number where the read of the entry ends <number> Telephone number <typ> Type of number <text> Text corresponding to the telephone number NOTE: In the <text> field, there may appear special characters like <code>"</code> (0x22), <code>@</code> (0x00), <code>`</code> (0x08), <code>Ö</code> (0x5c)... (See also +CPBW and Appendix A: Special hints for using +CPBR/+CPBW command) In models before the S25, empty phonebook records are reported as follows: +CPBR: <index1>,empty In S25ff, those empty entries don't produce any output.

AT+CPBW		Write a telephone-book entry																														
Test command AT+CPBW=?	Response +CPBW: (list of supported <index>s), <nlength>,(list of supported <type>s), <tlength> OK/ERROR/+CME ERROR	Parameter <index> Location number <nlength> Max. length of telephone number <tlength> Max. length of text corresponding to the number																														
Write command AT+CPBW= [<index>] [,<number> [,<typ>[,<text>]]]	Parameter <index> Location number at which the entry is written <number> Telephone number <typ> Type of number <text> Text corresponding to the telephone number Response OK/ERROR/+CME ERROR Note: The following characters in <text> must be entered via the escape sequence (see also Appendix A: Special hints for using +CPBR/+CPBW command)	<table border="1"> <thead> <tr> <th>GSM Char</th> <th>Hex char.</th> <th>ASCII</th> <th>GSM Esc Seq</th> <th>Seq.(hex)</th> <th>Note</th> </tr> </thead> <tbody> <tr> <td>Ö</td> <td>5C</td> <td>\</td> <td>Ö5C</td> <td>5C 35 43</td> <td>Backslash</td> </tr> <tr> <td>"</td> <td>22</td> <td>"</td> <td>Ö22</td> <td>5C 32 32</td> <td>String delim</td> </tr> <tr> <td>ò</td> <td>08</td> <td>BSP</td> <td>Ö08</td> <td>5C 30 38</td> <td>Backspace</td> </tr> <tr> <td>@</td> <td>00</td> <td>NULL</td> <td>Ö00</td> <td>5C 30 30</td> <td>GSM Null</td> </tr> </tbody> </table> <p>y cause problems on application level when using the function strlen() and should thus be represented by an escape sequence</p>	GSM Char	Hex char.	ASCII	GSM Esc Seq	Seq.(hex)	Note	Ö	5C	\	Ö5C	5C 35 43	Backslash	"	22	"	Ö22	5C 32 32	String delim	ò	08	BSP	Ö08	5C 30 38	Backspace	@	00	NULL	Ö00	5C 30 30	GSM Null
GSM Char	Hex char.	ASCII	GSM Esc Seq	Seq.(hex)	Note																											
Ö	5C	\	Ö5C	5C 35 43	Backslash																											
"	22	"	Ö22	5C 32 32	String delim																											
ò	08	BSP	Ö08	5C 30 38	Backspace																											
@	00	NULL	Ö00	5C 30 30	GSM Null																											

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

560	FEATURE PUK REQUIRED
The following CMS errors have been defined for SMS:	
300	ME failure
301	SMS service of ME reserved
302	operation not allowed
303	operation not supported
304	invalid PDU parameter
305	invalid TEXT mode
310	SIM not inserted
311	SIM PIN necessary
312	PH-SIM PIN necessary
313	SIM failure
314	SIM busy
315	SIM wrong
320	memory failure
321	invalid memory failure
322	memory full
330	SMSC address unknown
331	no network service
332	network timeout
340	NO +CNMA ACK EXPECTED
500	unknown error

AT+VTS	Send a DTMF tone <small>PDU parameter</small>
<small>Test command</small> AT+VTS=?	<small>Response</small> (list of supported <dtmf>s), (list of supported <duration>s) OK/ERROR/+CME ERROR <small>Parameter</small> <dtmf> 0-9,#,*,A-D, exactly one character <duration> Duration of tone in (duration/10) seconds
<small>Write command</small> AT+VTS= <dtmf> [,<duration>] or AT+VTS= <dtmf-string>	<small>Parameter</small> <dtmf> One character from the list, see Test command See Test command <dtmf-string> max. 29 characters in quotation marks ("..."), then a duration cannot be specified <small>Response</small> OK/ERROR/+CME ERROR Important: There is a leading output prefix +VTS in models before the S25.

AT+VTD		Set duration of a DTMF tone
Test command AT+VTD=?	Response +VTD: (list of supported <duration>s) OK/ERROR/+CME ERROR	Parameter <duration> 1-255 Duration of tone in (duration/10) seconds
Read command AT+VTD?	Response +VTD: <duration> OK/ERROR/+CME ERROR	
Write command AT+VTD= <duration>	Parameter <duration> See Test command	Response OK/ERROR
Important: There is a leading output prefix +VTD in models before the S25.		

AT+WS46		Select wireless network
Test command AT+WS46=?	Response (list of supported <n>s) OK	
Read command AT+WS46?	Response <n> OK/ERROR/+CME ERROR	Parameter <n> Integer; WDS side stack 12 GSM digital cellular
Write command AT+WS46=[<n>]	Response OK/ERROR/+CME ERROR	
Important: There is a leading output prefix +WS46 in models before the S25.		

AT+CSCS		Select TE character set
Test command AT+CSCS=?	Response +CSCS: (list of supported <chset>s) OK	
Read command AT+CSCS?	Response +CSCS: <chset> OK/ERROR/+CME ERROR	Parameter <chset> String; determines which TE character set is used
Write command AT+CSCS= [<chset>]	Response OK/ERROR/+CME ERROR	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT+CAOC	Advice of charge
Test command AT+CAOC=?	Response +CAOC: (list of supported <mode>s) Parameter <mode> 0 query CCM value
Read command AT+CAOC?	Response +CAOC: <mode> Parameter <mode> 0 See Test command
Write command AT+CAOC=<mode>	Response OK Parameter <mode> 0 See Test command
Execute command AT+CAOC	Response +CAOC: <ccm> OK/ERROR/+CME ERROR Parameter <ccm> Updated hexadecimal call meter, measured in home units; coding analogous to ACMmax on the SIM

AT+CSSN	Supplementary service notifications Revision according to GSM 07.07 Version 5.0.0
Test command AT+CSSN=?	Response +CSSN: (list of supported <n>s), (list of supported <m>s) Parameter <n> 0 Suppresses the +CSSI messages 1 Activates the +CSSI messages <m> 0 Suppresses the +CSSU messages 1 Activates the +CSSU messages For supported +CSSI/+CSSU messages, see also 1.3.4. Summary of All UnexpectedMessages
Read command AT+CSSN?	Response +CSSN: <n>, <m> Parameter <n> See Test command <m> See Test command
Write command AT+CSSN=<n>[, <m>]	Parameter <n> See Read command <m> See Read command
	Unexpected message +CSSI: <code1> +CSSU: <code2> Parameter <code1> Intermediate result code 3 Waiting call is pending <code2> Unsolicited result code 5 Held call was terminated

AT+CRSM		Restricted SIM access
Test command AT+CRSM=?	Response OK	
Write command +CRSM=<command> [,<fileid> [,<P1>,<P2>,<P3> [,<data>]]]	Response +CRSM: <sw1>,<sw2>[,<response>] OK/ERROR/+CME ERROR	Parameter <command>: 176 READ BINARY 178 READ RECORD 192 GET RESPONSE 214 UPDATE BINARY 220 UPDATE RECORD 242 STATUS <fileid>: Integer, identifier of the data file on the SIM, mandatory for every command except STATUS (see GSM 11.11) <P1>,<P2>,<P3>: Integer, transferal parameter from ME to SIM, mandatory for every command except GET RESPONSE,STATUS (see GSM 11.11) <data>: Hexadecimal string; information that is to be written to the SIM <sw1>,<sw2>: Integer; information from the SIM as to how/whether the command was executed <response>: Hexadecimal string; given when a command was successfully processed Note: The write access to CK boxes receives only limited support and differs from device to device.

AT+CIMI		Output of IMSI
Test command AT+CIMI=?	Response OK	
Execute command AT+CIMI	Response <imsi> Parameter <imsi>	International Mobile Subscriber Identity (IMSI)

AT+CACM	Accumulated call meter
Test command AT+CACM=?	Response OK
Read command AT+CACM?	Response +CACM: <acm> OK/ERROR/+CME ERROR Parameter <acm> Accumulated call meter in hexadecimal format, measured in home units; coding analogous to ACMmax on the SIM
Write command AT+CACM=[<passwd>]	Response OK/ERROR/+CME ERROR Parameter <passwd> String type; usually PIN2

AT+CAMM	Accumulated call meter maximum
Test command AT+CAMM=?	Response OK
Read command AT+CAMM?	Response +CAMM: <acmmax> OK/ERROR/+CME ERROR Parameter <acmmax> Accumulated call meter maximum in hexadecimal format, measured in home units; coding analogous to ACMmax on the SIM
Write command AT+CAMM=[<acmmax>[,<passwd>]]	Response OK/ERROR/+CME ERROR Parameter <acmmax> (see Read command) <passwd> String type; usually PIN2



Copyright © 2007 Siemens AG. All rights reserved. Printed in Thailand.

<p>AT+CLCC</p>	<p>List Current Calls</p>
<p>Test command AT+CLCC=?</p>	<p>Response OK</p>
<p>Execute command AT+CLCC</p>	<p>Response [+CLCC: <id1>,<dir>,<stat>,<mode>,<empty>,<number>,<type>] [<CR><LF>+CLCC: <id2>,<dir>,<stat>,<mode>,<empty>,<number>,<type> [...]]</p> <p>OK/ERROR/+CME ERROR</p> <p>Parameter <idx>: integer type; call identification number as described in GSM 02.30 [19] subclause 4.5.5.1; this number can be used in +CHLD command operations</p> <p><dir>: 0 mobile originated (MO) call 1 mobile terminated (MT) call</p> <p><stat> (state of the call): 0 active 1 held 2 dialing (MO call) 3 alerting (MO call) 4 incoming (MT call) 5 waiting (MT call)</p> <p><mode> (bearer/teleservice): 0 voice 1 data 2 fax 3 voice followed by data, voice mode 4 alternating voice/data, voice mode 5 alternating voice/fax, voice mode 6 voice followed by data, data mode 7 alternating voice/data, data mode 8 alternating voice/fax, fax mode 9 unknown</p> <p><empty>: 0 call is not one of multiparty (conference) call parties 1 call is one of multiparty (conference) call parties</p> <p><number>: string type phone number in format specified by <type> <type>: type of address octet in integer format</p>

AT+CCLK	Clock
Test command AT+CCLK=?	Response OK
Read command AT+SCLK?	Response +CCLK: <time> OK/ERROR/+CME ERROR Parameter: <time>: string type value; format is "yy/MM/dd,hh:mm:ss", where characters indicate year (two last digits), month, day, hour, minutes; E.g. 6th of May 1994, 22:10:00 hours equals to „94/05/06,22:10:00"
Write command AT+CCLK=<time>	Response OK/ERROR/+CME ERROR Parameter: <time> see Test command

AT+COPN	Read operator names
Test command AT+COPN=?	Response OK
Execute command AT+COPN	Response +COPN:numeric <oper>,long alphanumeric <oper><CR><LF> +COPN:..... OK/ERROR/+CME ERROR Parameter <oper> Network operator in numeric and alphanumeric notation see AT+SPLM

AT+CPUC	Price per unit and currency table
Test command AT+CPUC=?	Response OK
Read command AT+CPUC?	Response +CPUC: <currency>, <ppu> OK/ERROR/+CME ERROR Parameter <currency> three-character currency code (e.g. "FRA", "DEM") <ppu> price per unit; dot is used as a decimal separator (e.g. "1.33")
Write command AT+CPUC= <currency>,<ppu>[, <passwd>]	Response OK/ERROR/+CME ERROR Parameter <passwd> String type; usually PIN2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT+CALM	Alert sound mode
Test command AT+CALM=?	Response +CALM: (list of supported <mode>s) OK
Read command AT+CALM?	Response +CALM: <mode> OK/ERROR/+CME ERROR
Write command AT+CALM=<mode>	Response OK/ERROR/+CME ERROR Parameter <mode>: 0 normal mode 1 silent mode (all sounds are prevented) 2 beep (only a short beep indicates an incoming call)

AT+CRSL	Ringer sound level
Test command AT+CRSL=?	Response +CRSL: (list of supported <level>s) OK
Read command AT+CRSL?	Response +CRSL: <level> OK/ERROR/+CME ERROR
Write command AT+CRSL=<level>	Response OK/ERROR/+CME ERROR Parameter <level>: Ringer Sound Level

AT+CLVL	Loudspeaker volume level
Test command AT+CLVL=?	Response +CLVL: (list of supported <level>s) OK
Read command AT+CLVL?	Response +CLVL: <level> OK/ERROR/+CME ERROR
Write command AT+CLVL=<level>	Response OK/ERROR/+CME ERROR Parameter <level>: Loudspeaker Volume Level

AT+CMUT	Mute control
Test command AT+CMUT=?	Response +CMUT: (list of supported <n>s) OK
Read command AT+CMUT?	Response +CMUT: <n> OK/ERROR/+CME ERROR
Write command AT+CMUT=<n>	Response OK/ERROR/+CME ERROR Parameter <n>: 0 mute off 1 mute on

AT+CVIB	Vibrator mode
Test command AT+CVIB=?	Response +CVIB: (list of supported <mode>s) OK
Execute command AT+CVIB	Response +CVIB: <mode> OK/ERROR/+CME ERROR
Write command AT+CVIB=<mode>	Response OK/ERROR/+CME ERROR Parameter <mode>: Vibrator mode 0 disable 1 enable 16 vibrate then ring (not available in every model)



1.3.2. AT Commands According to GSM 07.05 for SMS

The GSM 07.05 commands are used for operating the SMS functions of the GSM mobile phone. The GSM module MOBILE supports the SMS PDU mode.

AT+CSMS	Selection of message service Revision according to GSM 07.05 Version 5.0.0
Test command AT+CSMS=?	Response +CSMS: (list of supported <service>S) Parameter <service> 0 GSM 3.40 and 3.41 1 GSM 3.40 and 3.41 and compatibility of the AT command syntax for phase 2+ NOTE: Deactivating the phase 2+ compatibility is only possible if the direct output of short messages +CNMI=1,2 or +CNMI=1,3 is not activated. If necessary, the latter should be deactivated first.
Read command AT+CSMS?	Response +CSMS: '<service>,<mt>,<mo>,<bm>' Parameter <service> 0 GSM 3.40 and 3.41 <mt> 1 Mobile terminated messages 0 Type supported <mo> 1 Mobile originated messages 0 Type supported <bm> 1 Broadcast type messages 0 Type not supported
Write command AT+CSMS= <service>	Parameter <service> 0 GSM 3.40 and 3.41 Response +CSMS: <mt>,<mo>,<bm> OK/ERROR/+CMS ERROR



<p>AT+CPMS</p>	<p>Selection of SMS memory Revision according to GSM 07.05 Version 4.7.0</p>
<p>Test command AT+CPMS=?</p>	<p>Response +CPMS: (list of supported <mem1>s),(list of supported <mem2>s) ,(list of supported <mem3>s) Parameter <mem1> Memory from which messages are read and deleted "SM" SIM-messages memory <mem2> Memory to which messages are written and sent "SM" SIM-messages memory <mem3> Memory in which received messages are stored, if forwarding to the PC is not set ("+CNMI") "SM" SIM-messages memory</p>
<p>Read command AT+CPMS?</p>	<p>Response +CPMS: <mem1>,<used1>,<total1>,<mem2>,<used2>,<total2> ,<mem3>,<used3>,<total3> Parameter <memx> Memory from which messages are read and deleted <usedx> Number of messages currently in <memx> <totalx> Number of storable messages in <memx></p>
<p>Write command AT+CPMS= <mem1> [,<mem2> [,<mem3>]]</p>	<p>Parameter <mem1> See Test command <mem2> See Test command <mem3> See Test command Response +CPMS: <used1>,<total1>,<used2>,<total3>,<used3>,<total3> OK/ERROR/+CMS ERROR</p>

<p>AT+CMGF</p>	<p>SMS format</p>
<p>Test command AT+CMGF=?</p>	<p>Response +CMGF: (list of supported <mode>s) Parameter <mode>: 0 PDU mode</p>
<p>Read command AT+CMGF?</p>	<p>Response +CMGF: <mode> Parameter <mode>: 0 PDU mode</p>
<p>Write command AT+CMGF=[< mode>]</p>	<p>Parameter <mode>: 0 PDU mode Response OK/ERROR</p>

AT+CSCA	Address of the SMS service center
Test command AT+CSCA=?	Response OK
Read command AT+CSCA?	Response +CSCA: <sca>,<tosca> Parameter <sca> Service-center address in string format <tosca> Service-center address format
Write command AT+CSCA= <sca>[,<tosca>]	Parameter <sca> Service-center address in string format <tosca> Service-center address format Response OK/ERROR

AT+CNMI	Display new incoming SMS Revision according to GSM 07.05 Version 4.7.0
Test command AT+CNMI=?	<p>Response +CNMI: (list of supported <mode>s),(list of supported <mt>s),(list of supported <bm>s),(list of supported <ds>s),(list of supported <bfr>s)</p> <p>Parameter <mode> 0 Buffers unexpected messages (but is equivalent to rejecting; see <bfr>) 1 Discard indication and reject new received message unsolicited result codes when TA-TE link is reserved. Otherwise forward them directly to the TE. (only with S25ff) 2 Buffers unexpected messages if serial interface is occupied, otherwise they are output (only models before S25)</p> <p><mt> 0 Suppresses unexpected messages for incoming short messages 1 Unexpected messages of a received short message (SMS-DELIVER) that is stored on a chip card are output in the form +CMTI: <mem>,<index> 2 Unexpected messages of a received short message (SMS-DELIVER) (except class 2 and the message "Waiting Indication Group: store message") are output in the form +CMT: [<alpha>],<length><CR><LF><pdu> (<alpha> is not supported) 3 Class 2 and the message "Waiting Indication Group: store message" are output as <mt>=1 3 Unexpected messages of a received short message (SMS-DELIVER) class 3 are output as <mt>=2. Messages with other data coding schemes are output as <mt>=1.</p> <p>NOTE: <mt>=2 and <mt>=3 are not possible unless the Phase 2+ compatibility has been activated by means of +CSMS=1</p> <p><bm> 0 Suppresses unexpected messages for incoming cell broadcast messages 2 Outputs unexpected messages for cell broadcast messages in the form +CBM: <length><CR><LF><pdu></p> <p><ds> 0 Suppresses unexpected messages for incoming SMS status reports 2 Outputs unexpected messages for SMS status reports in the form +CDS: <length><CR><LF><pdu></p>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งาน... ไม่ควรแก้ไขใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	<p><bfr> 1 Buffered unexpected messages are rejected when switching from <mode> 0 to <mode> 2.</p> <p><mem> See +CPMS <index> Index of the record on the chip card <alpha> alphanumeric representation of the sender address <length> Length of <pdu> <pdu> See +CMGL</p>
<p>Read command AT+CNMI?</p>	<p>Response +CNMI: <mode>,<mt>,<bm>,<ds>,<bfr></p> <p>Parameter <mode> See Test command <mt> See Test command <bm> See Test command <ds> See Test command <bfr> See Test command</p>
<p>Write command AT+CNMI= [<mode> [,<mt>[,<bm> [,<ds>[,<bfr>]]]]]</p>	<p>Parameter <mode> See Test command <mt> See Test command <bm> See Test command <ds> See Test command <bfr> See Test command</p> <p>Response OK/ERROR/+CMS ERROR</p>
	<p>Unexpected message +CMTI: <mem>,<index> Indication that new message has arrived +CMT: ,<length><CR><LF><pdu> Direct output of the short message +CDS: <length><CR><LF><pdu> Direct output of the status report +CBM: <length><CR><LF><pdu> Direct output of the cell broadcast message</p>

<p>AT+CNMA</p>	<p>Acknowledgment of a short message directly output (without storing on the chip card) Revision according to GSM 07.05 Version 5.0.0 <i>(NOTE: This command is not possible unless the Phase 2+ compatibility has been activated by means of +CSMS=1)</i></p>
<p>Test command AT+CNMA=?</p>	<p>Response +CNMA: (list of supported <n>s)</p> <p>Parameter <n> 0 Mode of functioning analogous to GSM 07.05 text mode</p>
<p>Write command AT+CNMA[=<n>]</p>	<p>Parameter <n> See Test command</p> <p>Response OK/ERROR/+CMS ERROR: <err></p>

<p>AT+CMGL</p>	<p>List SMS Revision according to GSM 07.05 Version 4.7.0</p>										
<p>Test command AT+CMGL=?</p>	<p>Response +CMGL: (list of supported <stat>s)</p> <p>Parameter <stat></p> <table border="0"> <tr> <td>0</td> <td>"REC UNREAD": received unread messages (default)</td> </tr> <tr> <td>1</td> <td>"REC READ": received read messages</td> </tr> <tr> <td>2</td> <td>"STO UNSENT": stored unsent messages</td> </tr> <tr> <td>3</td> <td>"STO SENT": stored sent messages</td> </tr> <tr> <td>4</td> <td>"ALL": all messages</td> </tr> </table>	0	"REC UNREAD": received unread messages (default)	1	"REC READ": received read messages	2	"STO UNSENT": stored unsent messages	3	"STO SENT": stored sent messages	4	"ALL": all messages
0	"REC UNREAD": received unread messages (default)										
1	"REC READ": received read messages										
2	"STO UNSENT": stored unsent messages										
3	"STO SENT": stored sent messages										
4	"ALL": all messages										
<p>Write command AT+CMGL [=<stat>]</p>	<p>Parameter <stat> See Test command</p> <p>Response If PDU mode (+CMGF=0) and command are successful: +CMGL:<index>,<stat>,[<alpha>],<length> <CR><LF><pdu><CR><LF> +CMGL: <index>,<stat>,<alpha>,<length> <CR><LF><pdu><CR><LF> [...]</p>										
	<p>Parameter <pdu> The PDU begins with the service-center address (according to GSM04.11), followed by the TPDU according to GSM03.40 in hexadecimal format otherwise: +CMS ERROR: <err></p>										



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT+CMGR	Read in an SMS Revision according to GSM 07.05 Version 4.7.0
Test command AT+CMGR=?	Response OK
Write command AT+CMGR= <index>	Parameter <index> Index of message in selected memory <mem1> Response If PDU mode (+CMGF=0) and command are successful: +CMGR: <stat>,<[alpha]>,<length><CR><LF><pdu> Parameter <pdu> Siehe "AT+CMGL" otherwise: +CMS ERROR: <err>

AT+CMGS	Send an SMS
Test command AT+CMGS=?	Response OK
Write command If PDU mode (+CMGF=0) +CMGS=<length><CR> <i>PDU is given</i> <ctrl-Z/ESC>	Parameter <length> Length of PDU <pdu> See "AT+CMGL" <mr> Message reference Response If sending is successful: +CMGS: <mr> If sending is not successful: +CMS ERROR: <err>

AT+CMSS	Send an SMS from the SMS memory
Test command AT+CMSS=?	Response OK
Write command +CMSS=<index>,<da>,<toda>]]	Parameter <index> Index of message in selected memory <mem1> <da> Destination address in string format <toda> Format of destination address <mr> Message reference Response If sending is successful: +CMSS: <mr> If sending is not successful: +CMS ERROR: <err>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT+CMGW	Write an SMS to the SMS memory
Test command AT+CMGW=?	Response OK
Write command If PDU mode (+CMGF=0) AT+CMGW=<length>[,<stat>]<CR>PDU is given <ctrl-Z/ESC>	Parameter <length> Length of PDU <stat> See command +CMGL <pdu> See "AT+CMGL" <index> Index of message in selected memory <mem1> Response +CMGW: <index> +CMS ERROR: <err>

AT+CMGD	Delete an SMS in the SMS memory
Test command At+CMGD=?	Response OK
Write command AT+CMGD= <index>	Parameter <index> Index of message in the selected memory <mem1> Response OK/ERROR/+CMS ERROR

AT+CSCB	Select cell broadcast messages
Test command AT+CSCB=?	Response +CSCB: (list of supported <mode> s) Parameter <mode> 0 Accepts messages that are defined in <mids> and <dcss> 1 Does not accept messages that are defined in <mids> and <dcss>
Read command AT+CSCB?	Response +CSCB: <mode>,<mids>,<dcss> Parameter <mode> See Test command <mids> String type; combinations of CBM message IDs <dcss> String type; combinations of CBM data coding schemes
Write command AT+CSCB=[<mode>[,<mids> >[,<dcss>]]]	

AT+CMGC	Send an SMS command
Test command AT+CMGC=?	Response OK
Write command If PDU mode (+CMGF=0) +CMGC=<length><CR>PDU is given <ctrl-Z/ESC>	Parameter <length> Length of PDU <pdu> See "AT+CMGL" <mr> Message reference Response If sending is successful: +CMGC: <mr> If sending is not successful: +CMS ERROR: <err>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานี้เท่านั้น ไม่ควรแก้ไขหรือดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3.3. User-Defined Commands for Controlling the GSM Mobile Phone

Since user-defined commands cannot be implemented according to official syntax, the character string "+C" is replaced by "^S" ("^" = 0x5E). For the future: if a user-defined command is accepted in the same syntax in GSM recommendations, the command can be addressed using both command strings.

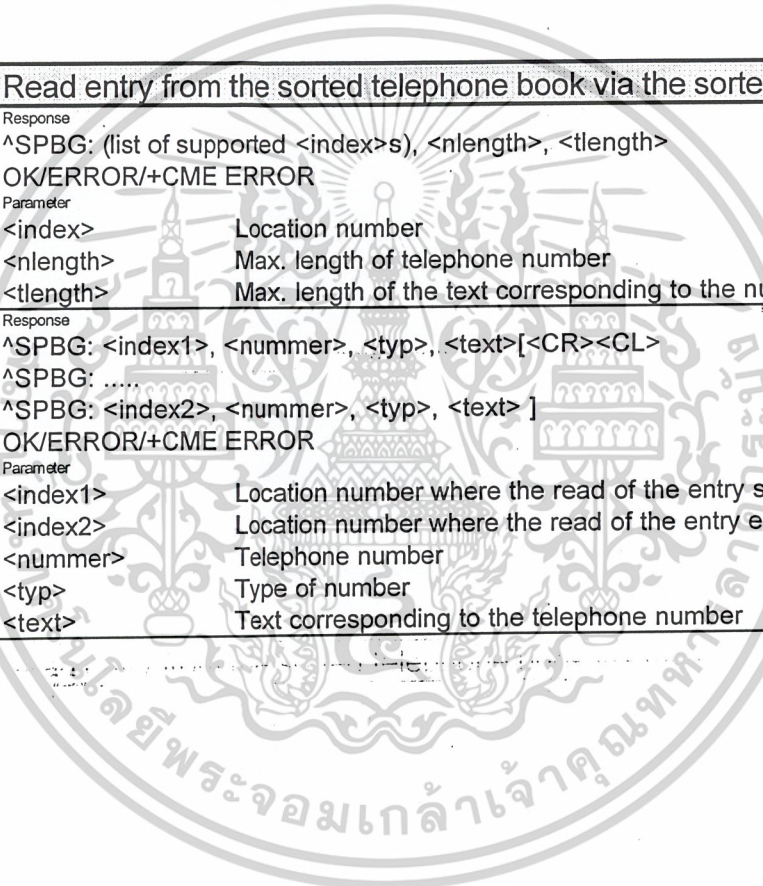
<p>AT^SPBS</p>	<p>Select a telephone book (including Siemens-specific books)</p>
<p>Test command AT^SPBS=?</p>	<p>Response ^SPBS: (list of supported <sto>s) OK/ERROR/+CME ERROR</p> <p>Parameter <sto></p> <ul style="list-style-type: none"> "FD" SIM fix-dialing telephone book "SM" SIM telephone book "ME" Telephone book in device "DC" ME Dialed Calls List "ON" Own telephone numbers "LD" SIM last dialing number "MC" ME Missed Calls List "RC" ME Received Calls List "MD" Last number redial memory in telephone device "OW" Own numbers "BD" Barred dialing numbers "SD" Service dialing numbers "MS" Missed dialing numbers (unanswered calls) "CD" Callback dialing numbers (answered calls) "BL" Blacklist dialing numbers (barred numbers from remote) E.S. "MB" Mailbox dialing numbers (network-operator mailbox) "CS" Common sortable telephone book (sorted combination of "SM", "ME", "FD"; access only via ^SPBC, ^SPBG) "RD" Red book (all entries in "CS" whose name portions have an exclamation point (!) as the final character) <p>*For a description of the telephone-book features, see Appendix A</p>
<p>Read command AT^SPBS?</p>	<p>Response ^SPBS: <sto> OK/ERROR/+CME ERROR</p> <p>Parameter <sto> See Test command</p>
<p>Write command AT^SPBS= <sto></p>	<p>Parameter <sto> See Test command</p> <p>Response OK/ERROR/+CME ERROR</p>

<p>AT^SDLD</p>	<p>Delete the "last number redial" memory</p>
<p>Test command AT^SDLD=?</p>	<p>Response OK</p>
<p>Execute command AT^SDLD</p>	<p>Response OK/ERROR/+CME ERROR</p>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT^SPBC	Seek the first entry in the sorted telephone book which begins with the selected (or next available) letter
Test command AT^SPBC=?	Response ^SPBC: (list of sorted telephone books supported <mem>s) See AT+CPBS/AT^SPBS OK/ERROR/+CME ERROR
Write command AT^SPBC=<char> ar>	Parameter <char> First letter of sought entry "A" to "Z" (with any other character, the index of the first entry that begins with a special character is sent back) <index> Index in the sorted telephone book (access via AT^SPBG) Response ^SPBC: <index> OK/ERROR/+CME ERROR

AT^SPBG	Read entry from the sorted telephone book via the sorted index
Test command AT^SPBG=?	Response ^SPBG: (list of supported <index>s), <nlength>, <tlength> OK/ERROR/+CME ERROR Parameter <index> Location number <nlength> Max. length of telephone number <tlength> Max. length of the text corresponding to the number
Write command AT^SPBG= <index1> [, <index2>]	Response ^SPBG: <index1>, <number>, <typ>, <text>[<CR><CL> ^SPBG: ^SPBG: <index2>, <number>, <typ>, <text>] OK/ERROR/+CME ERROR Parameter <index1> Location number where the read of the entry starts <index2> Location number where the read of the entry ends <number> Telephone number <typ> Type of number <text> Text corresponding to the telephone number



AT^SLCK	Switch locks (including user-defined locks) on and off
<p>Test command</p> <p>AT^SLCK=?</p>	<p>Response</p> <p>^SLCK: (list of supported <fac>s) OK/ERROR/+CME ERROR</p> <p>Parameter</p> <p><fac></p> <ul style="list-style-type: none"> "PS" Phone locked to SIM (device code) "SC" SIM card (PIN) "FD" FDN lock "AO" BAOC (bar all outgoing calls) "OI" BOIC (bar outgoing international calls) "OX" BOIC-exHC (bar outgoing international calls except to home country) "AI" BAIC (bar all incoming calls) "IR" BIC-Roam (bar incoming calls when roaming outside the home country) "AB" All barring services "AG" All outgoing barring services "AC" All incoming barring services <ul style="list-style-type: none"> "PN" Network personalization (GSM 02.22) "PC" Corporate personalization (GSM 02.22) "PU" Network subset personalization (GSM 02.22) "PP" Service provider personalization (GSM 02.22) "PF" Phone locked to very first inserted SIM
<p>Write command</p> <p>AT^SLCK = <fac>, <mode> [,<passwd> [,<class>]]</p>	<p>Parameter</p> <p><fac> See Test command</p> <p><mode></p> <ul style="list-style-type: none"> 0 Cancels lock 1 Activates lock 2 Queries lock status <p><passwd> Password</p> <p><class></p> <ul style="list-style-type: none"> 1 Voice 2 Data 4 Fax 7 All classes (default value) <p>Response</p> <p>If <mode>=2 and command is successful</p> <p>^SLCK: <status>[,<class1>][<CR><LF> ^SLCK: <status>, class2....]]</p> <p>Parameter</p> <p><status></p> <ul style="list-style-type: none"> 0 Off 1 On <p>OK/ERROR/+CME ERROR</p>

AT^SPWD	Change password to a lock (including user-defined locks)
<p>Test command</p> <p>AT^SPWD=?</p>	<p>Response</p> <p>^SPWD: list of supported (<fac>, <pwdlength>s) OK/ERROR/+CME ERROR</p> <p>Parameter</p> <p><fac> "P2" PIN2</p> <p>otherwise See Test command for the command AT^SLCK, without "FD"</p> <p><pwdlength> Length of password</p>
<p>Write command</p> <p>AT^SPWD = <fac>,<oldpwd>, <newpwd></p>	<p>Parameter</p> <p><fac> See Test command for the command AT^SLCK</p> <p><oldpwd>, <newpwd> Old and new password</p> <p>Response</p> <p>OK/ERROR/+CME ERROR</p>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT^SACM	Output ACM (accumulated call meter) and ACMmax
Test command AT^SACM=?	Response ^SACM: (list of supported <n>s)
Execute command AT^SACM	Response ^SACM: <n>, <acm>, <acm_max> OK/ERROR/+CME ERROR Parameter <n> See Test command <acm> Accumulated call meter <acm_max> Maximum accumulated call meter
Write command AT^SACM=<n>	Parameter <n> 0 Suppresses the unexpected message 1 Outputs the unexpected message

AT^SPLM	Read the PLMN list
Test command AT^SPLM=?	Response OK
Execute command AT^SPLM	Response ^SPLM:numeric <oper>, long alphanumeric <oper><CR><LF> ^SPLM:..... OK/ERROR/+CME ERROR Parameter <oper> Network operator in numeric and alphanumeric notation

AT^SPLR	Read an entry from the preferred-operator list
Test command AT^SPLR=?	Response ^SPLR: (list of supported <index>s) OK/ERROR/+CME ERROR Parameter <index> Location numbers
Write command AT^SPLR=<index1> [, <index2>]	Response ^SPLR: <index1>, numeric <oper> ^SPLR: ^SPLR: <index2>, numeric <oper> OK/ERROR/+CME ERROR Parameter <index1> Location number where the read of the entry starts <index2> Location number where the read of the entry ends <oper> Network operator in numeric form

AT^SPLW	Write an entry to the preferred-operator list
Test command AT^SPLW=?	Response ^SPLW: (list of supported <index>s) OK/ERROR/+CME ERROR Parameter <index> Location number
Write command AT^SPLW=<index>[, <oper>]	Parameter <index> Location number at which the entry is written <oper> Network operator in numeric form Response OK/ERROR/+CME ERROR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT^SCNI	
Output call number information	
Test command AT^SCNI=?	Response OK
Execute command AT^SCNI	Response ^SCNI: 1[,<cs>[,<number>,<type>]]<CR><LF> ^SCNI: 2[,<cs>[,<number>,<type>]]<CR><LF> ^SCNI: 3[,<cs>[,<number>,<type>]]<CR><LF> ^SCNI: 4[,<cs>[,<number>,<type>]]<CR><LF> ^SCNI: 5[,<cs>[,<number>,<type>]]<CR><LF> ^SCNI: 6[,<cs>[,<number>,<type>]]<CR><LF> ^SCNI: 7[,<cs>[,<number>,<type>]] OK/ERROR/+CME ERROR Parameter <cs> Call status of affiliated call number (first parameter) 0 Call on hold 1 Active call 2 Waiting call <number> Telephone number <type> Type of number

AT^SNFV	
Set the volume	
Test command AT^SNFV=?	Response ^SNFV: (list of supported <vol>s) Parameter <vol> Value range of volume (0 to 4) (0 low, ..., 4 max. volume; approx. 3 dB/level)
Read command AT^SNFV?	Response ^SNFV: <vol> Parameter <vol> See Test command
Write command AT^SNFV=<vol>	Parameter <vol> See Test command Response OK/ERROR

AT^SNFS	
Select NF hardware	
Test command AT^SNFS=?	Response ^SNFS: (list of supported <dev>s) Parameter <dev> 0 Cell phone mode 1 Handsfree
Read command AT^SNFS?	Response ^SNFS: <dev> Parameter <dev> See Test command Note: Volume should be temporarily set to „0“ before NF hardware is changed.
Write command AT^SNFS=<dev>	Parameter <dev> See Test command Response OK/ERROR

AT^SRTC	Set the ringing tone
Test command AT^SRTC=?	Response ^SRTC: (list of supported <type>s), (list of supported <vol>s) Parameter <type> 1-X Number of ringing tone 0 Mutes the ringing tone; when MTC is set, the phone does not ring and the volume is ignored. <vol> 0-Y Volume of ringing tone
Read command AT^SRTC?	Response ^SRTC: <type>, <vol>, <ringing> Parameter <type> See Test command <vol> See Test command <ringing> 0 Test-ring is switched off 1 Test-ring is switched on
Write command AT^SRTC=[<type>][,<vol>]	Parameter <type> See Test command <vol> See Test command Response OK/ERROR
Execute command AT^SRTC	Response The ringing tone sounds on the current NF device; it is selected using "AT+CNFS" until AT^SRTC is called up again OK/ERROR/+CME ERROR Note: If an MTC arrives while the test-ring is active, the latter is switched off and the "normal" ring is switched on.

AT^SCID	Output card ID
Test command AT^SCID=?	Response OK/ERROR/+CME ERROR
Execute command AT^SCID	Response ^SCID: <cid> OK/ERROR/+CME ERROR Parameter <cid> Number of SIM card

AT^SCKS	Output SIM card status
Test command AT^SCKS=?	Response ^SCKS: (list of supported <n>s) Parameter <n> 0 Suppresses the unexpected messages 1 Outputs the unexpected messages
Read command AT^SCKS?	Response ^SCKS: <n>, <m> Parameter <m> 0 No card 1 Card in card reader
Write command AT^SCKS=<n>	Parameter <n> See Test command Response OK/ERROR
	Unexpected message ^SCKS: <m> See Read command

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT^SMGL	List SMS (without status change from <i>unread</i> to <i>read</i>) Revision according to GSM 07.05 Version 4.7.0
Test command AT^SMGL=?	Response ^SMGL: (list of supported <stat>s) Parameter <stat> 0 "REC UNREAD": received unread messages (default) 1 "REC READ": received read messages 2 "STO UNSENT": stored unsent messages 3 "STO SENT": stored sent messages 4 "ALL": all messages
Write command AT^SMGL [=<stat>]	Parameter <stat> See Test command Response If PDU mode (+CMGF=0) and command is successful: ^SMGL: <index>, <stat>, [<alpha>], <length> <CR><LF><pdu> [<CR><LF>^SMGL: <index>, <stat>, [<alpha>], <length> <CR><LF><pdu> [...]]
	Parameter <pdu> The PDU begins with the service-center address (according to GSM04.11), followed by the TPDU according to GSM03.40 in hexadecimal format otherwise: +CMS ERROR: <err>

AT^SMGR	Read SMS record without Changing unread->read Syntax like AT+CMGR
Test command AT^SMGR=?	Response OK
Write command AT^SMGR= <index>	Parameter <index> Index of message in selected memory <mem1> Response If PDU mode (+CMGF=0) and command are successful: ^SMGR: <stat>, [<alpha>], <length><CR><LF><pdu> Parameter <pdu> Siehe "AT+CMGL" otherwise: +CMS ERROR: <err>

AT^SMSO	Switch device off
Test command AT^SMSO=?	Response OK
Execute command AT^SMSO	Response OK Device switches off

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT^SLNG		Language settings	
Test command AT^SLNG=?	Response ^SLNG: (list of supported languages <lng>s) Parameter: <lng>: Integer; language coded according to GSM 03.38 or mobile-specific language (>100)		
Read command AT^SLNG?	Response ^SLNG: <lng>		
Write command AT^SLNG=<lng>	Response OK/ERROR/+CME ERROR		

AT^SSTK		SIM Toolkit	
Test command AT^SSTK=?	Response ^SSTK: <profile> Parameter: <profile>	ME profile according to GSM 11.14	
Write command AT^SSTK=<length>[,<mode>]<CR> PDU is given <ctrl-Z/ESC>	Response: OK/ERROR/CME ERROR Parameter: <length>: Length of PDU in bytes <mode>: 0: Single command 1: Sequence of commands <pdu>: SIM Toolkit commands, see GSM 11.14 Limitation: The maximum PDU length is 176 bytes.		

AT^SBNW		Binary Write	
Test command AT^SBNW=?	Response ^SBNW: ((list of supported <types>s, list of supported <subtype>s)) OK/ERROR/+CME ERROR Parameter: <type>	„bmp“ bitmap Windows bitmap format without compression; 2/16/256 colours, at least 97x26 pixels <subtype> 0 shown permanently when registered in home network 1 shown temporarily, deleted by more important display contents „mid“ ring tones in standard MIDI format 0, without polyphony specification: http://www.midi.org <subtype> 0 first (and only) entry of type “mid” „vcs“ vcal format specification: http://www.imc.org/pdi	

	<p><subtype> 0 first entry of type "vcs" 1 entry of type "vcs" ...</p> <p><actNumber> 0 deletes entry of the act. subtype other actual packet number</p> <p><maxNumber> maximum number of packets</p>
<p>Write command AT^SBNW=<type>,<subtype>, [<actNumber>[, <maxNumber>]]<CR> PDU is given<ctrl-Z/ESC></p>	<p>Response OK/ERROR/+CME ERROR</p> <p>Parameter: <type> see Test commnd <subtype> see Test commnd <actNumber> see Test commnd <maxNumber> see Test commnd</p> <p>Note: -It is not possible to upload data when a call is active or in progress. If a call is active the mobile responses with +CME ERROR: PHONE BUSY and the actual upload sequence is aborted and all data packets are discarded. -If uploaded data is not useable (e.g. wrong data format) the mobile responses with +CME ERROR: INV CHAR IN TEXT after the last packet is uploaded. -To get the extended +CME-ERROR-responses AT+CMEE=2 has to be sent before. Otherwise the mobile responses only with ERROR . (see GSM07.07) -If <actNumber> and <maxNumber> during the upload are omitted, the mobile aborts the whole input sequence for the current subtype. -If <actNumber> is 0 during the upload and <maxNumber> is omitted, the mobile deletes the actual record with index <subtype> -Packets have to be uploaded in the right order! -Limitation:The maximum pdu size is 176 bytes (or 352 characters) See Appendix B for examples.</p>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT^SBNR	Binary Read
Test command AT^SBNR=?	Response ^SBNR: (list of supported <types>s, (list of supported <subtype>s)) OK/ERROR/+CME ERROR Parameter: <type> see AT^SBNW command <subtype> see AT^SBNW command
Write command AT^SBNR=<type>,<subtype>	Response ^SBNR: <type>,<subtype>,1,<maxNumber> <CR><LF><data><CR><LF> ^SBNR: <type>,<subtype>,2,<maxNumber> <CR><LF><data><CR><LF>[...] OK/ERROR/+CME ERROR Parameter: <type> see AT^SBNW command <subtype> see AT^SBNW command <data> data in hexadecimal form (PDU) <maxNumber> see AT^SBNW command See Appendix B for examples.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3.4. Summary of All Unexpected Messages

Message	Meaning
+CREG: <stat>	Network registration
+CLIP: <num>	Telephone number of caller
+CMTI: <mem>, <index>	Indication of a new short message
+CMT: , <length><CR><LF> <pdu>	Short message
+CSSI: <code1> +CSSU: <code2>	Supplementary service intermediate/unsolicited result code
^SMGO: <mode>	SMS overflow indicator
^SCKS: <m>	Message indicating whether card has been removed or inserted
^SACM: <m>	Message indicating if ACM has reached the maximum value ACMmax



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Appendix A

Factory settings made by AT&F

ATE1 (only in case of RCCP mode)
 ATQ0
 ATV1

AT+CREG=0
 AT+CLIP=0
 AT+CRC=0
 AT+CAOC=0
 AT+CMEE=0
 AT+CPBS=SM (if available)
 AT+COPS=0
 AT+VTS=1
 AT+CSCS="GSM"
 AT+CSSN=0,0
 AT^SCKS=0
 Reset pending locks (Phone Pin/Puk, Pin2/Puk2)
 which are give as answer of AT+CPIN?

AT+CSMS=0
 AT+CNMI=0,0,0,0,1
 AT^SMGO=0
 AT+CSCB=0

Features of the Telephone-Book Memory

Name	Description	Category / Access	Write	Delete completely
FD	Fix-dialing number (SIM fix-dialing telephone book)	GSM 07.07 / +CPBS or ^SPBS	Allowed (PIN2 required)	
SM	Abbreviate dialing number (SIM telephone book)	GSM 07.07 / +CPBS or ^SPBS	Allowed (device code required if FDN replacement is active)	
DC (MD)	Mobile last dialing number (last number redial memory; only if "LD" is not available)	GSM 07.07 / +CPBS or ^SPBS	Not allowed	By means of AT^SDLD
ON (OW)	Own Numbers (SIM own telephone numbers)	GSM 07.07 (Siemens) / +CPBS (historical)	Allowed	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ (historical) นั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LD	SIM last dialing number (last number redial memory on SIM)	GSM 07.07 / +CPBS or ^SPBS	Not allowed	By means of AT^SDLD
ME	Mobile-equipment telephone book (ME dialing numbers)	GSM 07.07 / +CPBS or ^SPBS	Allowed (device code required if FDN replacement is active)	
BD	Barred dialing numbers (blocked numbers)	Siemens / ^SPBS	Not allowed	
SD	Service dialing numbers (Service numbers)	Siemens / ^SPBS	Not allowed	
MC (MS)	Missed dialing numbers (unanswered calls)	GSM 07.07 (Siemens) / +CPBS, ^SPBS	Not allowed	
RC (CD)	Callback dialing numbers (answered calls)	GSM 07.07 (Siemens) / +CPBS, ^SPBS	Not allowed	
BL	Blacklist dialing numbers (numbers that are blocked for a certain time in order to prevent continuous accesses from remote control)	Siemens / ^SPBS	Not allowed	
MB	Mailbox dialing numbers (network-operator mailbox)	Siemens / ^SPBS	Not allowed	
CS	Common sortable numbers (sorted combination of "SM", "ME", "FD")	Siemens / ^SPBS / ^SPBC / ^SPBG	Not allowed	
RD	Red book numbers ("CS" entries with '!' at the end of the name portion)	Siemens / ^SPBS / ^SPBC / ^SPBG	Not allowed	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Writing to the FDN Phonebook / FDN Replacement

Writing to the fix-dialing number phonebook is protected by PIN2.
A Write sequence (to e.g. record 5) runs as follows:

```

AT+CMEE=2 //Activate expanded error message
OK

AT+CPBS=? // Listing of available telephone books
+CPBS: ("FD","SM","LD")
OK

AT+CPBS="FD" // Selection of the FDN telephone book
OK

AT+CPBW=5,"1234",,"test" // A Write to record 5 is attempted...
+CME ERROR: SIM PIN2 REQUIRED // ... PIN2 is required for this purpose

AT+CPIN? // Query of the PIN status...
+CPIN: SIM PIN2 // ... PIN2 is to be entered

AT+CPIN="12345678" // Input of PIN2
OK

AT+CPBW=5,"1234",,"test" // A Write to record 5 is attempted...
OK // PIN2 remains active as long as you use the commands
// +CPIN, +CPBS, +CPBR, +CPBW, +CACM,
// +CMM, +CPUC
// ^SPIC, ^SPBS, ^SPBC, ^SPBG,;
// If you use other commands or if none of the
// above commands are executed within five
// minutes, the validity of PIN2 is voided.

AT+CPBW=6,"5678",,"new test" // A Write to record 6 is attempted...
OK

```

In addition, if there is no FDN phonebook available on the SIM, it is possible to activate a feature which activates FDN-like behavior for the "SM" and "ME" phonebooks (FDN replacement). (Currently this feature can only be activated via the MMI lock/device lock/excluding telephone book.)

In this case, the Write to the "SM" and "ME" phonebooks is ensured by the device code (PH-SIM PIN and PH-SIM PUK, respectively).

The sequence for entering the device code is analogous to the above example.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Special hints for using +CPBR/+CPBW command

String parameters ,like the <text> in +CPBW command shall be entered using quotation marks "" (Ascii=Windows=GSM=0x22).e.g. "Doe Joe"

It is possible to enter string parameters without "" but not recommended, because following problems may occur:

If no "" are used:

- SPACEs (Space, Blank, Ascii=Windows=GSM=0x20) are skipped.
E.g. at+cpbw=1,"123",,K. H. results in "K.H." ☹
at+cpbw=1,"123",,"K. H." SPACEs are kept ☺
- No `;` (Ascii=Windows=GSM=0x2C) and `;` (Ascii=Windows=GSM=0x3B) in <text> is possible, because this characters are used as separator of parameters/commands.
E.g. at+cpbw=1,"123",,Kurz,Helmut result in ERROR ☹
at+cpbw=1,"123",,"Kurz,Helmut" ☺

But there are also some points to note when using quotation marks "":

There are some characters which cannot be entered in normal way:

e.g. quotation mark "" character itself, because this is interpreted as the end of the <text>

To make this (and some other special characters) possible to be entered, the character with hex value 0x5c is used as escape character. In the ASCII character set this is equal the `\
Unfortunately there is no `\
in GSM character set. The 0x5C equals the `Ö`

The escape sequence has the following structure:

- The sequence begins with the escape character 0x5C (ASCII=Windows=`, GSM=`Ö`)
- The special character follows and is entered 2 Byte representation of the GSM character set value .
e.g. the 2 Byte representation of the `@` (GSM=0x00) is `00`

Following special characters shall be entered by using the escape sequence:

GSM Char	Hex char.	ASCII	GSM Esc Seq	Seq.(hex)	Note
Ö	5C	\	Ö5C	5C 35 43	Backslash
"	22	"	Ö22	5C 32 32	String delimiter
ò	08	BSP	Ö08	5C 30 38	Backspace
@	00	NULL	Ö00	5C 30 30	GSM NULL

Examples:

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

GSM string wanted in Phonebook	String in AT+CPBW Command(GSM)	String in AT+CPBW Command (Hex)
Ölhändler	"Ö5ClhÖ7Bndler"	22 5C 35 43 6C 68 7B 6E 64 6C 65 72 22
"Eddi" Kurz	"Ö22EddiÖ22 Kurz"	22 5C 32 32 45 64 64 69 5C 32 32 20 4B 75 72 7A 22
Ooo	"OÖ08o" 22 4F 5C	30 38 6F 22
@Adr.	"Ö00Adr."	22 5C 30 30 41 64 72 2E 22
		[no problems with strlen()]
	"@Adr."	22 00 41 64 72 2E 22
		[may cause problems with strlen() in application]

Note:

When reading phonebook records, there is NO replacement. Every character will appear in normal GSM character set notation (like the left column in the example above).



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Appendix B

Example for creating/interrogation of an organizer entry

-vcs object which has to be uploaded:

```
BEGIN:VCALENDAR
VERSION:1.0
BEGIN:VEVENT
CATEGORIES:ANNIVERSARY
DTSTART:19991213T100000
DESCRIPTION:W. von Siemens
END:VEVENT
END:VCALENDAR
```

-hexadecimal representation of this object:

```
424547494E3A5643414C454E4441520D0A56455253494F4E3A312E300D0A4245
47494E3A564556454E540D0A43415445474F524945533A414E4E4956455253415
2590D0A445453544152543A3139393931323133543130303030300D0A44455343
52495054494F4E3A572E20766F6E205369656D656E730D0A454E443A56455645
4E540D0A454E443A5643414C454E4441520D0A
```

-upload of an entry on record 20

```
at^sbnw="vcs",20,1,3<CR>
<CR><LF> > <Space>
424547494E3A5643414C454E4441520D0A56455253494F4E3A312E300D0A4245
47494E3A564556454E540D0A43415445474F<Ctrl-Z>
<CR><LF>OK<CR><LF>
```

```
at^sbnw="vcs",20,2,3<CR>
<CR><LF> > <Space>
524945533A414E4E49564552534152590D0A445453544152543A3139393931323
133543130303030300D0A44455343524950<Ctrl-Z>
<CR><LF>OK<CR><LF>
```

```
at^sbnw="vcs",20,3,3<CR>
<CR><LF> > <Space>
54494F4E3A572E20766F6E205369656D656E730D0A454E443A564556454E540D
0A454E443A5643414C454E4441520D0A<Ctrl-Z>
<CR><LF>OK<CR><LF>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
All characters are answered with an echo.

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้.

Echoing can be switched off with „ATE0“.

In this example the organizer entry is uploaded in 50 bytes packets (100 input characters in every pdu).

The blue painted characters characterize the responses of the mobile.

-interrogation of the current <type>,<subtype>,<actNumber>,<maxNumber>

```
at^sbnw?<CR>
<CR><LF>^SBNW: "vcs",20,2,3<CR><LF>
<CR><LF>OK<CR><LF>
```

description: The actual object which is uploaded is an VCS object.

It has to be stored on record 20.

2 of 3 packets are already uploaded.

-deleting of record 20

```
at^sbnw="vcs",20,0<CR>
<CR><LF>OK<CR><LF>
```

-download entry from record 20

```
at^sbnr="vcs",20<CR>
<CR><LF>^SBNR:<space>"vcs",20,1,1<CR><LF>
424547494E3A5643414C454E4441520D0A56455253494F4E3A312E300D0A4245
47494E3A564556454E540D0A43415445474F524945533A414E4E4956455253415
2590D0A445453544152543A313939393132313335431303030300D0A44455343
52495054494F4E3A572E20766F6E205369656D656E730D0A454E443A56455645
4E540D0A454E443A5643414C454E4441520D0A<CR><LF>
<CR><LF>OK<CR><LF>
```

The mobile segments the record entry in 176 byte (=176*2 characters) packets.

-Download of an empty record 20

```
at^sbnr="vcs",20<CR>
<CR><LF>OK<CR><LF>
```

-Test command of AT^SBNW

```
at^sbnw=?<CR>
<CR><LF>^SBNW:<space>("bmp",(0)),("mid",(0)),("vcs",(1-30)) <CR><LF>
<CR><LF>OK<CR><LF>
```

สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

description: The mobile supports bitmaps with suptype 0, midi objects with suptype 0 and vcs objects with the suptypes 1 up to 30.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้