สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบแจ้งสถานะของตู้ขายน้ำอัตโนมัติผ่าน GPRS

REPORTING STATUS OF VENDING MACHINE VIA GPRS

T104118

นายอานนท์ แย้มศิริ
นายเอกสิทธิ์ ศรีสังกา
นางสาวพัชราพร หินใส

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
เลขหมู่ 2551
เลขทะเบียน 104118
วัน,เดือน,ปี 30 ต.ค. 2552
b 12102482
i

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมการวัดคุม
ภาควิชาวิศวกรรมการวัดคุม คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2551

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

REPORTING STATUS OF VENDING MACHINE VIA GPRS

ARNON YEAMSIRI

EKKASIT SRISANGKA

PATCHARAPORN HINSAI

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

BACHELOR OF ENGINEERING IN INSTRUMENTATION ENGINEERING

DEPARTMENT OF INSTUMENTATION ENGINEERING

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2008

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาควิชาวิศวกรรมการวัดคุม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ใบรับรองปริญญานิพนธ์

.....

หัวข้อปริญญานิพนธ์	ระบบแจ้งสถานะของตู้ขายน้ำอัตโนมัติผ่าน GPRS		
	REPORTING STATUS OF VENDING MACHINE VIA GPRS		
นักศึกษาผู้จัดทำ	นายอานนท์	แย้มศิริ	**รหัสนักศึกษา** 48011116
	นายเอกสิทธิ์	ศรีสังกา	**รหัสนักศึกษา** 48011154
	นางสาวพัชราพร	หินใส	**รหัสนักศึกษา** 48012064
ปริญญา	วิศวกรรมศาสตรบัณฑิต		
สาขาวิชา	วิศวกรรมการวัดคุม		
ปีการศึกษา	2551		

อาจารย์ผู้ควบคุมปริญญานิพนธ์	ลายมือชื่อ
ผศ.เชื้อ นกอยู่	
ผศ.ดร.ฤดี มาสุจันท์	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	ระบบแจ้งสถานะของตู้ขายน้ำอัตโนมัติผ่าน GPRS			
	REPORTING STATUS OF VENDING MACHINE VIA GPRS			
นักศึกษาผู้จัดทำ	นายอานนท์	แย้มศิริ	รหัสนักศึกษา	48011116
	นายเอกสิทธิ์	ศรีสังกา	รหัสนักศึกษา	48011154
	นางสาวพัชราพร	หินใส	รหัสนักศึกษา	48012064
อาจารย์ที่ปรึกษา	ผศ.เชื้อ	นกอยู่		
	ผศ.ดร.ฤดี	มาสุจันท์		
ปีการศึกษา	2551			

บทคัดย่อ

โครงงานนี้สร้างขึ้นเพื่อเพิ่มประสิทธิภาพให้กับธุรกิจเครื่องขายสินค้าอัตโนมัติ เนื่องจากในปัจจุบันธุรกิจเครื่องขายสินค้าอัตโนมัติมีบริการอย่างแพร่หลาย ทำให้ยากในการบริหารจัดการเติมสินค้า เมื่อสินค้าในเครื่องขายสินค้าอัตโนมัติหมดทางผู้ประกอบการจะไม่สามารถทราบและทำการเติมสินค้าได้ทันตามความต้องการของลูกค้าได้ ทำให้เสียโอกาสในการจำหน่ายสินค้า โครงงานนี้จึงจัดทำขึ้นเพื่อแจ้งสถานะของเครื่องขายสินค้าอัตโนมัติ เพื่อใช้ในการวางแผนการบริหารธุรกิจและเพิ่มประสิทธิภาพให้กับธุรกิจประเภทนี้

โครงงานนี้จัดทำเครื่องจำหน่ายสินค้าอัตโนมัติเพื่อจำลองการขายสินค้าทำให้ระบบเกิดการส่งข้อมูลขึ้น โดยที่เครื่องจำหน่ายสินค้าอัตโนมัติจะรายงานสถานะของการขายสินค้า เช่น เวลา วันที่ ประเภทของสินค้า หมายเลขของเครื่องจำหน่ายสินค้าอัตโนมัติ เพื่อนำมาวางแผนการเติมสินค้าของเครื่องจำหน่ายสินค้าตามจุดต่าง ๆ เมื่อมีการขายสินค้าเครื่องจำหน่ายสินค้าอัตโนมัติจะทำการส่งข้อมูลไปยัง GPRS ผ่านทางพอร์ทอนุกรมซึ่ง GPRS จะส่งข้อมูลไปยัง Server เพื่อเก็บข้อมูลไว้ในระบบฐานข้อมูลที่สร้างขึ้นและสร้างเว็บเพจเพื่อแสดงข้อมูลผ่านระบบ Internet ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Reporting Status of Vending Machine Via GPRS	
Authors	Mr.Arnon	Yeamsiri
	Mr.Ekkasit	Srisangka
	Miss Patcharaporn	Hinsai
Thesis Advisor	Asst.Prof. Chuae	Nokyoo
	Asst.Prof.Dr. Ruedee	Masuchun
Year	2008	

ABSTRACT

Sell automatic machines which is open service extensively now have the problem that the Entrepreneur cannot know that some good are run out. To solve this problem, this project then purposes the system which can report status of vending machine to the entrepreneur via GPRS and show on webpage. The entrepreneur can know when goods in the cabinet are out of order. In this system, Microcontroller control the vending machine and send data via GPRS to computer server then the entrepreneur can see report on webpage and bring it for planning marketing.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้ สำเร็จลุล่วงได้ด้วยดีเพราะได้รับความเมตตาและการสนับสนุนจาก ผศ.เชื้อ นกอยู่ และ ผศ.ดร.ฤดี มาสุจันท์ ที่ได้ให้คำแนะนำแก่ผู้วิจัยตลอดเวลาอีกทั้งยังเอื้อเฟื้ออุปกรณ์และเครื่องมือต่าง ๆ ในการจัดทำปริญญานิพนธ์ครั้งนี้ คณะผู้จัดทำรู้สึกซาบซึ้งและขอขอบพระคุณอาจารย์ทั้งสองท่านเป็นอย่างสูง

ขอขอบพระคุณ อาจารย์ภาควิชาวิศวกรรมการวัดคุมทุกท่าน ที่ได้ให้คำแนะนำและสถานที่ภาควิชาอันเป็นประโยชน์ต่อการทำปริญญานิพนธ์ฉบับนี้

และที่ลืมเสียมิได้ ขอขอบพระคุณคุณพ่อ คุณแม่ และครอบครัวอันเป็นที่รัก ที่สนับสนุนและเป็นกำลังใจตลอดการทำการปริญญานิพนธ์ฉบับนี้

คุณค่าและประโยชน์อันพึงมีจากปริญญานิพนธ์ฉบับนี้ คณะผู้จัดทำขอมอบแด่ผู้มีพระคุณทุกท่าน

คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....I

บทคัดย่อภาษาอังกฤษ.....II

กิตติกรรมประกาศ.....III

สารบัญ.....IV

สารบัญตาราง.....IX

สารบัญรูป.....X

บทที่ 1 บทนำ.....1

1.1 ความเป็นมาและเหตุจูงใจของการทำวิจัย.....1

1.2 วัตถุประสงค์.....1

1.3 ขอบเขตของปริญญานิพนธ์.....1

1.4 ขั้นตอนการศึกษา.....2

บทที่ 2 ทฤษฎีและหลักการ.....2

2.1 ไมโครคอนโทรลเลอร์ตระกูล MCS-51.....3

2.1.1 โครงสร้างและการทำงานของพอร์ต.....3

2.1.2 การใช้งานเป็นพอร์ตอินพุต.....6

2.1.3 การใช้งานเป็นพอร์ตเอาต์พุต.....6

2.1.4 การจัดหน่วยความจำ.....7

2.1.4.1 หน่วยความจำโปรแกรม (Program memory).....7

2.1.4.2 หน่วยความจำข้อมูล (Data memory).....7

2.1.5 รีจิสเตอร์ที่เกี่ยวกับไมโครคอนโทรลเลอร์ MCS – 51.....10

2.1.6 โครงสร้างการอินเตอร์รัปต์ของไมโครคอนโทรลเลอร์ MCS – 51.....10

2.1.7 รีจิสเตอร์ฟังก์ชันพิเศษ.....11

2.1.7.1 Accumulator.....12

2.1.7.2 B Register.....12

2.1.7.3 Program status word (PSW).....12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

2.1.7.4 Stack Pointer.....12

2.1.7.5 Data Pointer Register.....13

2.1.7.6 Port 0 ถึง 3.....13

2.1.7.7 Serial Data Buffer (SBUF).....13

2.1.7.8 Control Register.....13

2.1.8 ไทเมอร์ / เคาน์เตอร์ของไมโครคอนโทรลเลอร์ MCS – 51.....14

2.1.9 พอร์ตอนุกรมของไมโครคอนโทรลเลอร์ MCS – 51.....14

2.2 GPRS.....15

2.2.1 เครือข่าย GSM.....17

2.2.2 การควบคุม GPRS Module.....18

2.3 รูปแบบการเชื่อมต่อ.....18

2.3.1 การติดต่อสื่อสารระหว่างไมโครคอนโทรลเลอร์กับ GPRS Module.....18

2.3.1.1 TCP/IP Stack Chip.....18

2.3.1.2 การสื่อสารแบบอนุกรม.....18

2.3.1.3 มาตรฐานพอร์ทอนุกรมแบบ RS-232.....20

2.3.1.4 คอนเน็กเตอร์สำหรับพอร์ต RS-232 และการเชื่อมต่อ.....21

2.3.2 การเชื่อมต่อ GPRS Module กับ Server.....22

2.3.2.1 อินเตอร์เน็ต.....22

2.3.2.2 โปรโตคอล TCP/IP.....23

2.3.3 การติดต่อ server/Web browser.....34

2.4 เว็บแอพพลิเคชัน (Web Application).....36

2.4.1 สถาปัตยกรรมไคลเอนต์/เซิร์ฟเวอร์ (Client/Server Architecture).....36

2.4.1.1 ประเภทของไคลเอนต์เซิร์ฟเวอร์ ตามลำดับชั้นในการทำงาน.....37

2.4.1.2 ประเภทของไคลเอนต์เซิร์ฟเวอร์ ตามประเภทการใช้งาน.....38

2.4.2 เว็บบราวเซอร์ (Web Browser.....38

2.4.3 Common Gateway Interface (CGI).....39

2.5 Application Server.....40

2.5.1 ระบบฐานข้อมูล (Database System).....40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

2.5.1.1 ความหมาย....40
2.5.1.2 ความสัมพันธ์ระหว่างข้อมูลในฐานข้อมูล....40
2.5.1.3 สถาปัตยกรรมมาตรฐานของระบบฐานข้อมูล....42
2.5.1.3.1 นิยามข้อมูลระดับภายนอก (External Schema....43
2.5.1.3.2 นิยามข้อมูลระดับแนวคิด (Conceptual Schema)....43
2.5.1.3.3 นิยามข้อมูลระดับภายใน (Internal Schema)....43
2.5.1.4 ฐานข้อมูลเชิงสัมพันธ์....44
2.5.1.4.1 ความหมาย....44
2.5.1.4.2 โครงสร้างข้อมูลแบบเชิงสัมพันธ์....44
2.5.1.4.3 นิยามข้อมูลระดับภายใน (Internal Schema)....46
2.5.1.5 กฎการคงสภาพของข้อมูล (Integrity Rule)....46
2.5.2 โปรแกรมฐานข้อมูลที่นิยมใช้....46
2.5.3 Apache....46
2.5.3.1 ประวัติ....46
2.5.3.2 สถาปัตยกรรมของของ Apache....47
2.5.3.3 ความสามารถ....48
2.5.4 PHP....49
2.5.4.1 การใช้ PHP....50
2.5.4.2 รูปแบบการเขียน PHP....51
2.5.4.3 การรองรับ PHP....52
2.5.5 My SQL....53
2.5.5.1 ชนิดของข้อมูลที่สนับสนุน....53
2.5.5.2 การใช้งาน....54
2.5.5.3 ส่วนเชื่อมต่อกับภาษาการพัฒนาอื่น (database connector)....54
2.5.5.4 โครงสร้างของภาษา SQL....55
2.6 Web page....56
2.6.1 Appserv....56
2.6.2 Macromedia Dreamweaver 8....58

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

2.6.2.1 ส่วนประกอบของ Dreamweaver 8.....58

บทที่ 3 การออกแบบและทดลองระบบ.....60

3.1 ส่วนของฮาร์ดแวร์ (Hardware).....61

3.1.1 ไมโครคอนโทรลเลอร์.....61

3.1.2 TCP/IP Stack Chip.....61

3.1.3 GPRS Module.....61

3.2 ส่วนของซอฟท์แวร์ (Software).....61

3.2.1 KEIL μVision3.....61

3.2.2 Apache.....61

3.2.3 My SQL, PHP.....62

บทที่ 4 การทดลองและผลการทดลอง.....63

4.1 การออกแบบการทดลอง.....63

4.2 ขั้นตอนการทดลอง.....63

4.3 บันทึกผลการทดลอง.....64

4.4 สรุปผลการทดลอง.....66

บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ.....67

5.1 สรุป.....67

5.2 ปัญหาทางเทคนิค.....67

5.3 แนวทางการพัฒนาต่อ.....67

บรรณานุกรม.....68

ภาคผนวก.....69

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ **หน้า**

2.1 รีจิสเตอร์ฟังก์ชันพิเศษ (SFRs) ของ MCS-51 11
2.2 ตาราง Flag 31
2.3 แสดง CGI ที่ใช้บน Platform ต่าง ๆ 40
2.4 ตัวอย่างการเก็บข้อมูลในฐานข้อมูลเชิงสัมพันธ์ 42
2.5 PHP 50

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่ **หน้า**

2.1 แสดง block Diagram AT89C51AC2....4
2.2 แสดงวงจรภายในของพอร์ตทุกพอร์ตในไมโครคอนโทรลเลอร์....5
2.3 แสดงวงจรพูลอัปภายในพอร์ตของไมโครคอนโทรลเลอร์....6
2.4 แสดงหน่วยความจำข้อมูลภายใน....8
2.5 แสดงหน่วยความจำข้อมูลส่วนล่าง....8
2.6 แสดงโครงสร้างของหน่วยความจำข้อมูลส่วนบน....9
2.7 แสดงรีจิสเตอร์ PSW....12
2.8 แสดงเครือข่ายการทำงานของ GPRS....16
2.9 แสดงเครือข่าย GSM....17
2.10 แสดงการทำงานของ TCP/IP Stack Chip....18
2.11 การต่ออุปกรณ์ภายนอกกับพอร์ตอนุกรมของคอมพิวเตอร์ในลักษณะต่าง ๆ....21
2.12 โครงสร้าง TCP/IP....25
2.13 IP Header....27
2.14 ICMP Header....28
2.15 UDP Header....29
2.16 TCP Header....30
2.17 การสื่อสารของ TCP....32
2.18 การสูญหายของแพคเก็ต....33
2.19 แสดงความแตกต่างระหว่าง HTML และ HTTP....35
2.20 ลักษณะของการส่งของข้อมูล....35
2.21 แสดงการทำงานแบบไม่จองสายของ HTTP....36
2.22 แสดงการทำงานบนระบบไคลเอนต์/เซิร์ฟเวอร์....37
2.23 แสดงสถาปัตยกรรมแบบ 2 Tier....37
2.24 แสดงสถาปัตยกรรมแบบ 3 Tier....38
2.25 แสดงการทำงานของ CGI....39
2.26 ตัวอย่างข้อมูลในฐานข้อมูลแบบแผนภูมิต้นไม้....41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

รูปที่ **หน้า**

2.27 ตัวอย่างข้อมูลในฐานข้อมูลแบบเครือข่าย.....41
2.28 สถาปัตยกรรม 3 นิยามข้อมูล (The 3-schema Architecture).....43
2.29 แสดงรายละเอียดของรีเลชั่น S.....44
2.30 สถาปัตยกรรมของ Apache.....48
2.31 แสดงหน้าต่างของโปรแกรม Dreamweaver 8.....58
3.1 แสดงการทำงานของระบบทั้งหมด.....60
4.1 แสดงหน้า Login เข้าสู่หน้าหลัก.....64
4.2 แสดงหน้าหลักเพื่อเลือกดูข้อมูลของแต่ละเครื่อง.....64
4.3 หน้าแสดงผลข้อมูลการซื้อสินค้า.....65
4.4 หน้าเว็บเพจการเติมสินค้าและแก้ไขข้อมูลสินค้า.....65
4.5 หน้าเพิ่มสินค้า.....66
4.6 หน้ารายงานการซื้อขายสินค้า.....66

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและเหตุจูงใจของการทำวิจัย

เนื่องจากการใช้ชีวิตในปัจจุบัน อาศัยความสะดวกสบายเข้ามาเป็นองค์ประกอบสำคัญ การจัดทำโครงงานนี้เพื่อช่วยในการบริหารจัดการเครื่องขายสินค้าอัตโนมัติ โดยสามารถตรวจสอบข้อมูลได้จากการส่งสถานะของเครื่องขายสินค้าอัตโนมัติผ่านระบบ GPRS ทำให้เราสามารถทราบสถานะของเครื่องขายสินค้าอัตโนมัติได้จากการส่งข้อมูลผ่านระบบ GPRS โดยมีเครื่องเซิร์ฟเวอร์เป็นตัวจัดเก็บข้อมูลและส่งผ่านต่อให้เครื่องไคลเอ็นท์ ทำให้มนุษย์ไม่ต้องออกไปตรวจสอบของในตู้ว่าขายได้เท่าไรและเหลือเท่าไร ซึ่งในชีวิตประจำวันในปัจจุบันต้องลดเวลาในการทำงานที่ต้องสิ้นเปลืองเวลาออกไป การจัดทำโครงงานนี้จึงได้แก้ไขโดยการใช้เทคโนโลยีเข้ามาช่วยในการตรวจสอบและบันทึกค่าได้

1.2 วัตถุประสงค์

1. เพื่อศึกษาการทำงานของไมโครคอนโทรลเลอร์และนำไปประยุกต์ใช้ในงานควบคุม
2. เพื่อจำลองเครื่องจำหน่ายสินค้าสำเร็จรูปแบบอัตโนมัติก่อนไปใช้จริง
3. เพื่อรายงานสถานะของเครื่องจำหน่ายสินค้าอัตโนมัติ ผ่านทางระบบ GPRS โดยทำการแสดงค่าบนเว็บเพจ
4. เพื่อศึกษาระบบการทำงานของ GPRS เพื่อใช้ในการส่งข้อมูล
5. เพื่อทำการเก็บข้อมูลการขายจากเครื่องขายสินค้าอัตโนมัติ ซึ่งจะสร้างความสะดวกให้แก่ผู้ประกอบการ ให้สามารถเรียกดูข้อมูลการจำหน่ายสินค้า เพื่อการบริหารจัดการในอนาคต

1.3 ขอบเขตของปริญญานิพนธ์

1. สามารถสร้างเครื่องขายสินค้าอัตโนมัติเพื่อให้มีการจำลองการขายสินค้าเพื่อส่งข้อมูล
2. สามารถส่งข้อมูลการซื้อสินค้าจากไมโครคอนโทรลเลอร์ผ่านทางระบบ GPRS ไปเก็บยังระบบฐานข้อมูลได้
3. สามารถสร้างระบบฐานข้อมูลเพื่อจัดเก็บและจัดการกับข้อมูลได้
4. สามารถสร้างเว็บเพจแสดงการขายสินค้าของแต่ละเครื่องได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 ขั้นตอนการศึกษา

1. ออกแบบและกำหนดขอบเขตของเครื่องจำลองขายสินค้าอัตโนมัติ
2. ศึกษาข้อมูลการทำงานของไมโครคอนโทรลเลอร์ตระกูล 8051
3. เขียนโปรแกรมภาษาซีให้กับไมโครคอนโทรลเลอร์ เพื่อทำการควบคุมการทำงานของเครื่องขายสินค้าอัตโนมัติ
4. ทดสอบโปรแกรมการทำงานของเครื่องขายสินค้าอัตโนมัติ
5. ศึกษาและทดลองการส่งข้อมูลของ GPRS และ TCP/IP Stack
6. สร้างเครื่อง Server เพื่อเก็บฐานข้อมูล
7. สร้างเว็บเพจเพื่อแสดงสถานะของเครื่องขายสินค้าอัตโนมัติ
8. เชื่อมต่อระบบเข้าด้วยกันทั้งหมดและทดลองส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการ

2.1 ไมโครคอนโทรลเลอร์ตระกูล MCS-51

ไมโครคอนโทรลเลอร์ตระกูล MCS-51 เป็นไมโครคอนโทรลเลอร์ขนาด 8 บิต ที่มีอุปกรณ์สนับสนุนประกอบอยู่ภายในหลายอย่าง ได้แก่ หน่วยความจำสำหรับเก็บข้อมูล หน่วยความจำสำหรับเก็บโปรแกรม ตัวตั้งเวลา/ตัวนับ อุปกรณ์รับส่งข้อมูลแบบอนุกรม เนื่องจากโครงสร้างของไมโครคอนโทรลเลอร์ มีอุปกรณ์สนับสนุนประกอบอยู่ภายในนี้เอง ทำให้การใช้งานง่ายขึ้นและมีประสิทธิภาพมากขึ้น โดยไม่ต้องมีการเชื่อมต่ออุปกรณ์ภายนอกเพิ่มเติมมากเหมือนกับตัวไมโครโปรเซสเซอร์ทั่วไป นอกจากนี้หากเราต้องการใช้งานไมโครคอนโทรลเลอร์ร่วมกับอุปกรณ์อื่น หรือหน่วยความจำภายนอกเรายังสามารถนำมาเชื่อมต่อเพิ่มเติมเข้ากับตัวไมโครคอนโทรลเลอร์ได้อีกด้วย

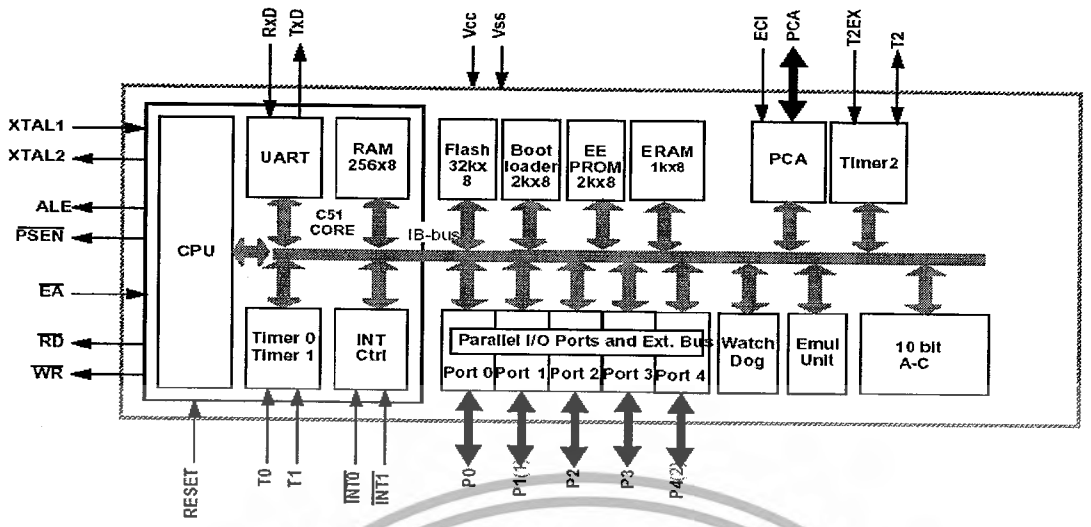
คุณสมบัติเบื้องต้นของไมโครคอนโทรลเลอร์ในตระกูล MCS-51 จะมีดังนี้

1. CPU 8 บิตและสามารถประมวลผลทางลอจิกกับข้อมูลในระดับบิตได้
2. หน่วยความจำข้อมูลภายใน 256 ไบต์
3. หน่วยความจำแฟลชภายในสำหรับเก็บโปรแกรมขนาด 32 กิโลไบต์
4. ตัวตั้งเวลาและตัวนับเวลาขนาด 16 บิต 2 ชุด
5. พอร์ตควบคุมการสื่อสารอนุกรมแบบ Full Duplex ซึ่งสามารถรับส่งข้อมูลพร้อมกันได้
6. พอร์ตขนานสำหรับติดต่อกับอุปกรณ์ภายนอกจำนวน 4 พอร์ต พอร์ตละ 8 บิต
7. มีตัวกำเนิดความถี่ Clock ภายใน

2.1.1 โครงสร้างและการทำงานของพอร์ต

ไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชมีพอร์ตให้ใช้งานทั้งสิ้น 4 พอร์ต 0 ถึงพอร์ต 3 แต่ละพอร์ตมีขนาด 8 บิต เป็นพอร์ตแบบสองทิศทาง กล่าวคือจะสามารถเป็นได้ทั้งอินพุตสำหรับสัญญาณข้อมูลเข้าและเอาท์พุตสำหรับส่งสัญญาณข้อมูลออก ทุกพอร์ตของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชมีวงจรแลตช์และวงจรขับตลอดจนบัฟเฟอร์อินพุต ดังแสดงให้เห็นดังรูปแสดง block Diagram AT89C51AC2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 แสดง block Diagram AT89C51AC2

ที่พอร์ต 0 และพอร์ต 2 จะใช้งานเป็นพอร์ตอินพุตและเอาท์พุตสำหรับงานทั่วไป และใช้ในการติดต่อกับหน่วยความจำภายนอก สำหรับพอร์ต 3 ทั้งพอร์ตและพอร์ต 1 บางขานอกจากจะใช้เป็นขาพอร์ตอินพุตเอาท์พุตตามปกติแล้ว ยังสามารถใช้งานในหน้าที่พิเศษได้อีก

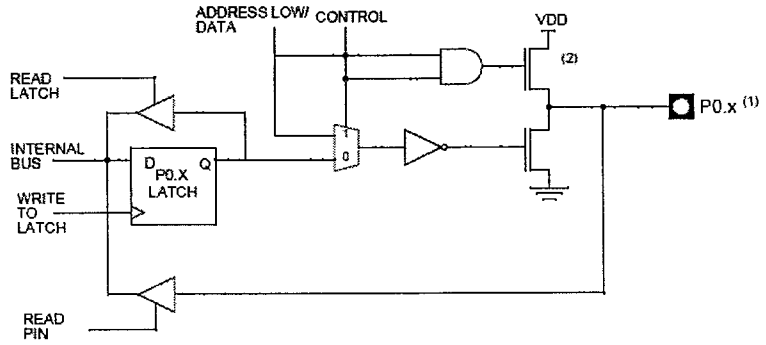
ในรูป 2.2 แสดงวงจรภายในของแต่ละพอร์ตของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลต โดยในรูป (ก) เป็นวงจรของพอร์ต 0 วงจรแลตช์ ของแต่ละบิตในแต่ละพอร์ตก็คือวงจรดีฟลิฟลอปนั่นเอง การอ่านค่าสถานะของวงจรแลตช์ สามารถกระทำได้อย่างอิสระด้วยสัญญาณที่แยกออกจากกัน นั่นคือสัญญาณอ่านข้อมูลจากขาพอร์ตและสัญญาณอ่านข้อมูลจากวงจรแลตช์ ส่วนการเขียนข้อมูลมายังพอร์ตต้องส่งสัญญาณมายังขา CLK ของดีฟลิฟลอปในขณะที่ข้อมูลจะผ่านมาทางขาบัสข้อมูลภายในเข้าสู่ขา D ของดีฟลิฟลอป โดยที่พอร์ตนี้มีวงจรมัลติเพลกซ์สำหรับกำหนดลักษณะการทำงานของพอร์ตว่าต้องการใช้งานเป็นขาพอร์ตอินพุตเอาท์พุตปกติ หรือใช้ในการติดต่อกับหน่วยความจำภายนอกไมโครคอนโทรลเลอร์

ในรูป (ก) เนื่องจากที่พอร์ต 0 ไม่มีวงจรพลูอัปภายในหากมีการนำพอร์ต 0 ไปใช้งานเป็นพอร์ตอินพุตจะต้องต่อตัวต้านทานพลูอัปภายนอกที่ขาพอร์ต 0 ทุกขาด้วย

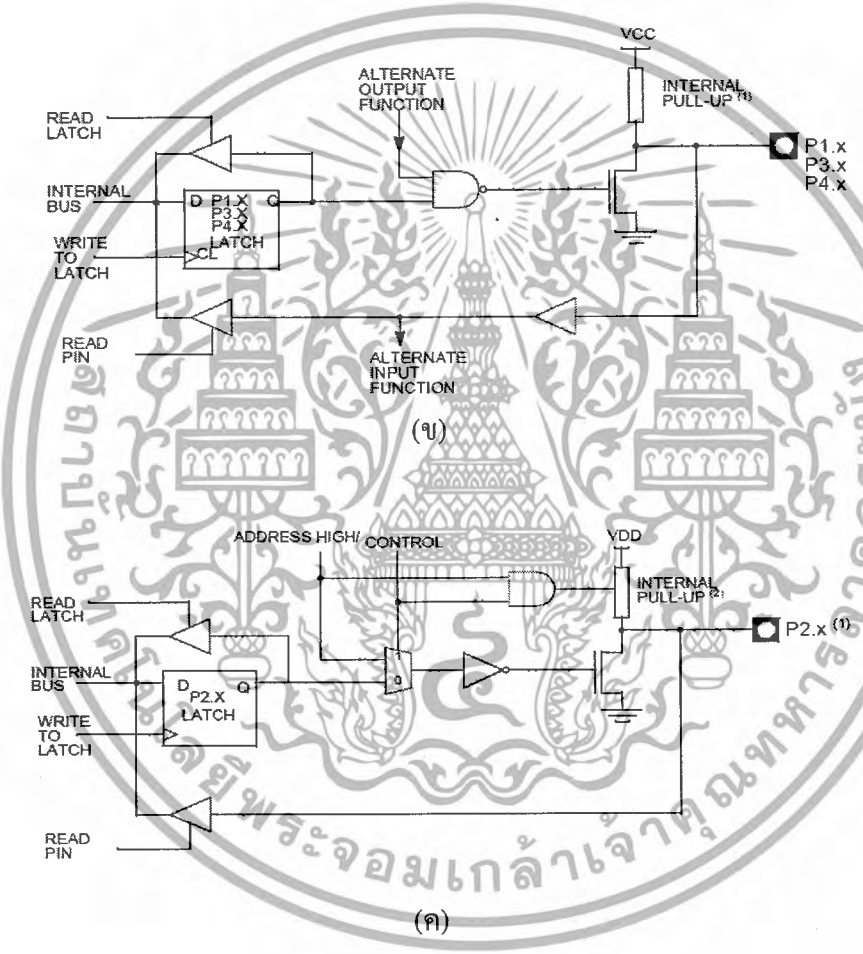
ในรูป (ข) เป็นวงจรภายในของพอร์ต 1,3,4 ซึ่งมีลักษณะโดยทั่วไปคล้ายกับพอร์ต 0 หากแต่ไม่มีวงจรมัลติเพล็กซ์ เนื่องจากพอร์ตนี้ไม่ใช้การติดต่อกับหน่วยความจำภายนอก แต่จะมีวงจรพลูอัปภายในที่แต่ละบิตของพอร์ตนี้แทนรายละเอียดของวงจรพลูอัปดังแสดงในรูป 2.3

ในรูป (ค) เป็นวงจรภายในของพอร์ต 2 ซึ่งจะคล้ายกับพอร์ต 0 มาก ต่างกันเพียงมีวงจรพลูอัปเพิ่มเติมเข้ามา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

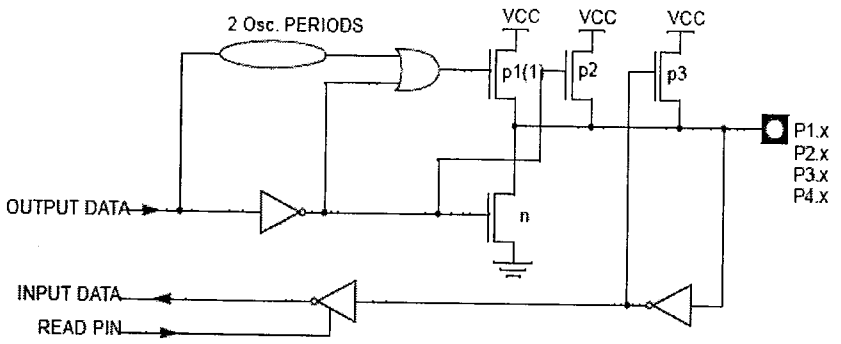


(ก)



รูปที่ 2.2 แสดงวงจรภายในของพอร์ตทุกพอร์ตในไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 แสดงวงจรพูลอัปภายในพอร์ตของไมโครคอนโทรลเลอร์

2.1.2 การใช้งานเป็นพอร์ตอินพุต

เนื่องจากพอร์ตทั้งหมดของไมโครคอนโทรลเลอร์ จะสามารถเป็นได้ทั้งอินพุตและเอาต์พุต ดังนั้น จึงมีความจำเป็นอย่างยิ่งต้องทำความเข้าใจถึงการกำหนดลักษณะการทำงานให้แก่พอร์ตของไมโครคอนโทรลเลอร์

ในการกำหนดให้เป็นพอร์ตอินพุต จะต้องเริ่มต้นด้วยการเขียนข้อมูล "1" มาที่บิตของพอร์ตที่ต้องการใช้งานเป็นอินพุต เพื่อที่จะหยุดการทำงานของเฟสที่ใช้ในการขับสัญญาณเอาต์พุตของบิตนั้น ๆ จะทำให้ขาสัญญาณของพอร์ตเชื่อมต่อ เข้ากับวงจรพูลอัปภายในโดยตรง ส่งผลให้ขาพอร์ตนั้นมีลอจิก "1" และสามารถรับสัญญาณลอจิก "0" จากอุปกรณ์ภายนอกได้ง่าย สัญญาณข้อมูลจากอุปกรณ์ภายนอก จะถูกส่งเข้ามาแล้วเก็บไว้ในวงจรบัฟเฟอร์ภายในพอร์ตและรอให้ CPU มาอ่านค่าเข้าไป เมื่อเช่นนี้อุปกรณ์ภายนอกที่เชื่อมต่อพอร์ตอินพุตของไมโครคอนโทรลเลอร์ควรกำหนดให้ทำงานในสภาวะลอจิก "0" จะดีและสะดวกที่สุด

2.1.3 การใช้งานเป็นพอร์ตเอาต์พุต

โดยปกติแล้ว ขาพอร์ตจะกำหนดให้มีลักษณะเป็นเอาต์พุตอยู่แล้ว ดังนั้นจึงสามารถส่งข้อมูลออกไปได้อย่างง่าย กล่าวคือ เมื่อต้องการส่งข้อมูล "0" ออกไปทางเอาต์พุตก็ให้เขียนข้อมูล "0" ไปยังวงจรแลตช์ซึ่งก็จะส่งไปขับเฟส ทำให้เฟสทำงานที่ขาพอร์ตที่กำหนดให้ทำงานก็จะเกิดลอจิก "0" ขึ้น ในทางตรงข้ามหากต้องการส่งข้อมูลหนึ่งออกไป ก็ให้เขียนข้อมูลหนึ่งไปยังวงจรแลตช์ วงจรขับก็จะหยุดทำงานทำให้ที่ขาพอร์ตเชื่อมต่อกับวงจรพูลอัปภายในเกิดเป็นลอจิก "1" ที่ขาพอร์ตนั้น ซึ่งจะคล้ายกับการกำหนดให้เป็นขาอินพุตมาก เพียงแต่แตกต่างกันที่กระบวนการในการเคลื่อนย้ายข้อมูล โดยถ้าอินพุตจะมีสัญญาณมาอ่านข้อมูลที่บัฟเฟอร์ แต่ถ้าเป็นเอาต์พุตจะไม่มีการอ่านค่าข้อมูลที่บัฟเฟอร์แต่อย่างใด เว้นแต่ในกรณีที่ต้องการตรวจสอบข้อมูลที่ส่งออกมาทางเอาต์พุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.4 การจัดหน่วยความจำ

ในไมโครคอนโทรลเลอร์จะมีหน่วยความจำภายในหลัก ๆ อยู่ 2 ส่วนคือ หน่วยความจำโปรแกรมและหน่วยความจำข้อมูล

2.1.4.1 หน่วยความจำโปรแกรม (Program memory)

ในการจัดหน่วยความจำโปรแกรมของไมโครคอนโทรลเลอร์ จะสามารถติดต่อหน่วยความจำโปรแกรมได้สูงสุด 64 กิโลไบต์ โดยสามารถเลือกใช้หน่วยความจำโปรแกรมภายในอย่างเดียวหรือรวมกับภายนอกหรือเลือกใช้หน่วยความจำภายนอกอย่างเดียวก็ได้

หน่วยความจำโปรแกรม จะใช้เก็บข้อมูลของโปรแกรมควบคุมการทำงานสำหรับ MCS-51 ที่เรียกว่า โปรแกรมมอนิเตอร์ (monitor program) หากใช้หน่วยความจำภายนอกมักจะบรรจุอยู่ในหน่วยความจำชนิด อีพรอม (EPROM : Erasable Programmable Read-Only Memory) ซึ่งสามารถทำการอ่านได้เพียงอย่างเดียว

หน่วยความจำโปรแกรมมีแอดเดรสเริ่มต้นที่ 000H เมื่อซีพียูได้รับการรีเซตให้เริ่มต้นการทำงาน ต้องมาเริ่มต้นที่แอดเดรส 000H นี้เสมอ อย่างไรก็ตามในพื้นที่ของหน่วยความจำโปรแกรมไม่ว่าจะใช้งานจากภายในหรือภายนอกก็ตาม จะต้องมีการสงวนพื้นที่บางตำแหน่งเอาไว้สำหรับการบริการอินเตอร์รัปต์ 6 ประเภท ประเภทละ 8 ไบต์ประกอบด้วย

พื้นที่สำหรับบริการอินเตอร์รัปต์ 0 จากภายนอก กำหนดไว้ที่แอดเดรส 0003H

พื้นที่สำหรับบริการอินเตอร์รัปต์จากไทเมอร์ 0 กำหนดไว้ที่แอดเดรส 000BH

พื้นที่สำหรับบริการอินเตอร์รัปต์ 1 จากภายนอก กำหนดไว้ที่แอดเดรส 0013H

พื้นที่สำหรับบริการอินเตอร์รัปต์จากไทเมอร์ 1 กำหนดไว้ที่แอดเดรส 001BH

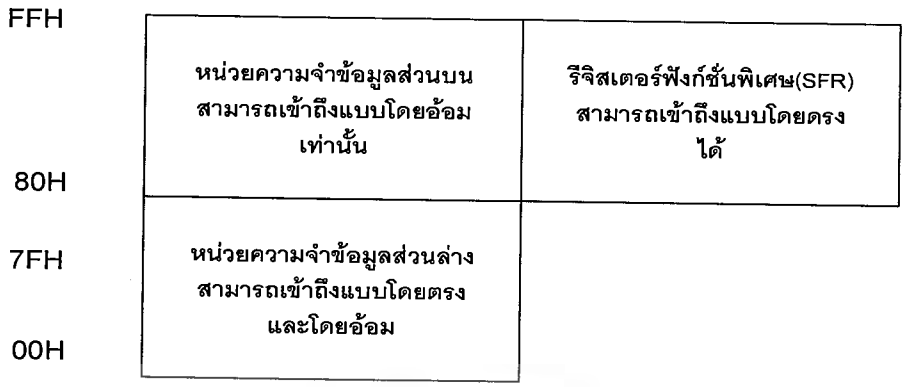
พื้นที่สำหรับบริการอินเตอร์รัปต์ของการสื่อสารอนุกรมกำหนดไว้ที่แอดเดรส 0023H

พื้นที่สำหรับบริการอินเตอร์รัปต์จากไทเมอร์ 2 กำหนดไว้ที่แอดเดรส 002BH

2.1.4.2 หน่วยความจำข้อมูล (Data memory)

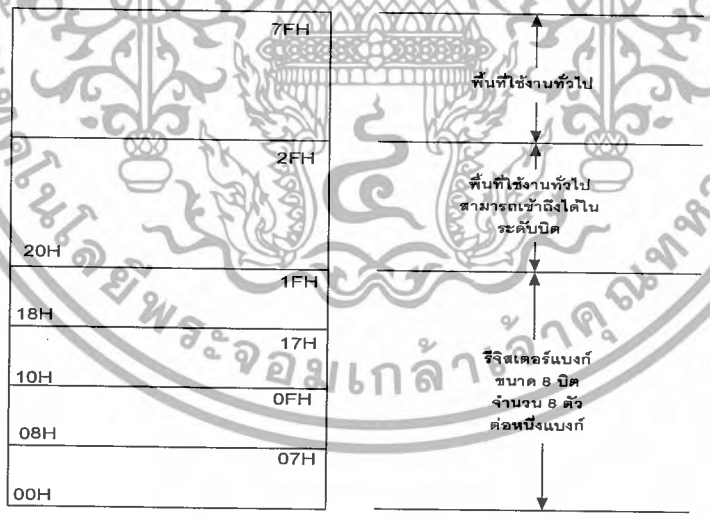
มี 2 ส่วนด้วยกันคือ หน่วยความจำข้อมูลภายนอกและหน่วยความจำข้อมูลภายในโดยตัวไมโครคอนโทรลเลอร์ สามารถติดต่อกับหน่วยความจำข้อมูลภายนอกได้สูงสุด 64 กิโลไบต์ และมีหน่วยความจำข้อมูลภายในเป็นแบบแรม (RAM : Random Access Memory) ในเบอร์ AT89C51AC2 นี้มีหน่วยความจำข้อมูลภายในขนาด 256 ไบต์ ใช้สำหรับการจัดสรรหน่วยความจำข้อมูลภายในจะแบ่งเป็น 3 ส่วนคือ หน่วยความจำข้อมูลส่วนล่าง (lower), หน่วยความจำข้อมูลส่วนบน (upper), รีจิสเตอร์ฟังก์ชั่นพิเศษ (SFR : Special Function Register) แต่ละส่วนมีขนาด 128 ไบต์ ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 แสดงหน่วยความจำข้อมูลภายใน

หน่วยความจำข้อมูลส่วนล่าง เป็นหน่วยความจำข้อมูลภายใน 128 ไบต์แรก แบ่งเป็นส่วนการใช้งานดังนี้คือ 32 ไบต์แรก (แอดเดรส 00H ถึง 1FH) จะแบ่งออกเป็น 4 ชุด ชุดละ 8 ไบต์ ในแต่ละไบต์จะใช้งานเป็นรีจีสเตอร์ R0-R7 โดยดังรูป ดังนั้นจึงมีรีจิสเตอร์ R0-R7 อยู่ทั้งหมด 4 ชุด (แอดเดรส 20H ถึง 2FH) ถัดมาเป็นการเข้าถึงแบบบิตและ 80 ไบต์ที่เหลือเราสามารถใช้เก็บข้อมูลทั่วไปได้

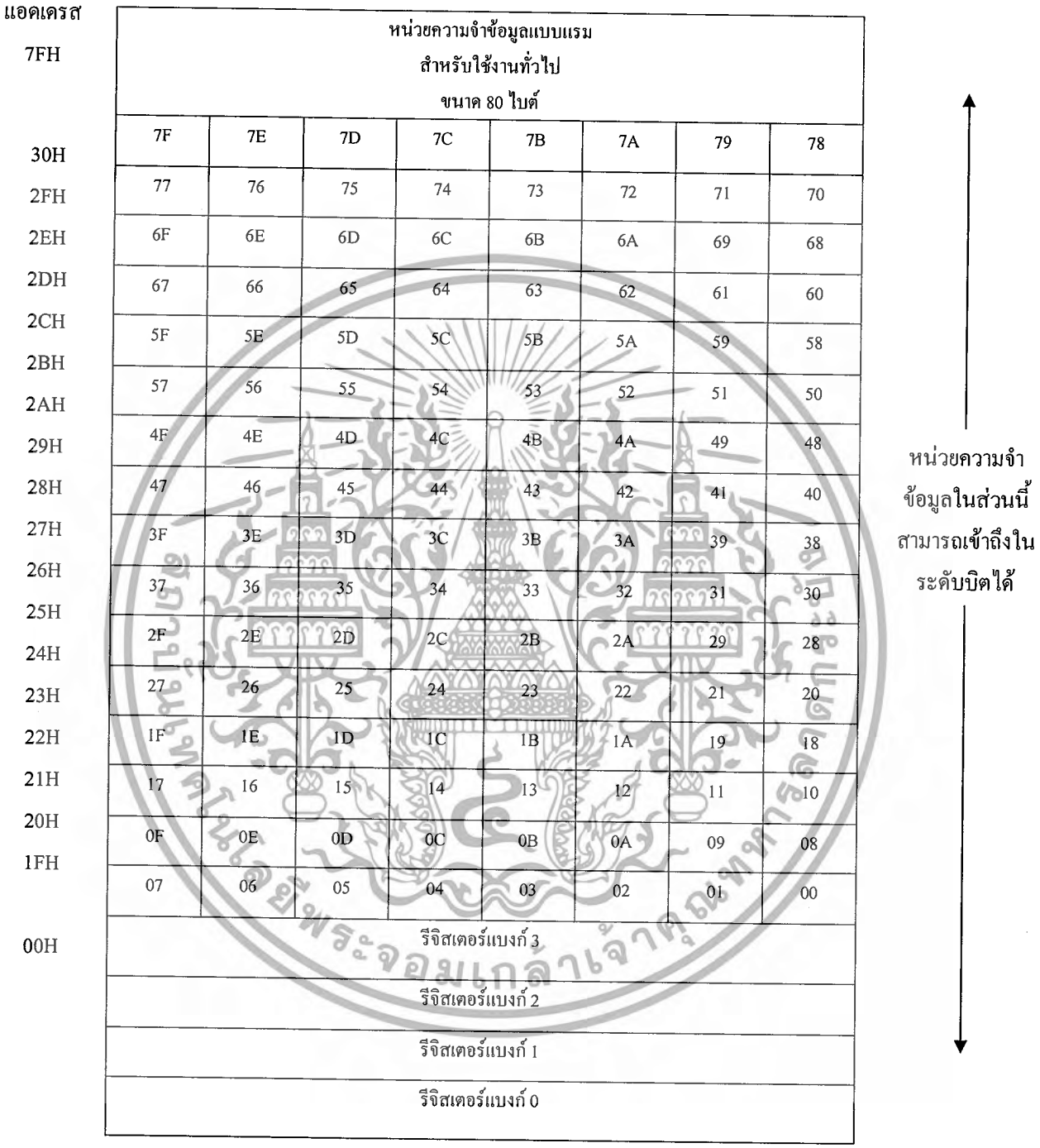


รูปที่ 2.5 แสดงหน่วยความจำข้อมูลส่วนล่าง

หน่วยความจำข้อมูลส่วนบน จะเป็นหน่วยความจำข้อมูลภายใน 128 ไบต์ถัดมา (แอดเดรส 80H ถึง FFH) จะถูกใช้งานเป็นรีจิสเตอร์พิเศษ (Special Function Register) ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รีจิสเตอร์แต่ละตัวสามารถเข้ามาถึงข้อมูลแบบบิต เช่น รีจิสเตอร์ PSW เริ่มต้นที่ตำแหน่ง D0H ถ้าเราต้องการอ้างอิงถึงบิตที่แทนตัวทด (Carry Flag) สามารถอ้างได้ด้วยตำแหน่ง D7H เป็นต้น



รูปที่ 2.6 แสดงโครงสร้างของหน่วยความจำข้อมูลส่วนบน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.5 รีจิสเตอร์ที่เกี่ยวกับไมโครคอนโทรลเลอร์ MCS – 51

รีจิสเตอร์ในกลุ่มนี้จะเป็นรีจิสเตอร์ขนาด 16 บิตที่ใช้งานเพื่อเก็บข้อมูลของตัวแอดเดรสเป็นสำคัญ โดยค่าที่อยู่ภายในแอดเดรสนี้จะนำไปเป็นค่าของข้อมูลที่ส่งออกไปทางบัสแอดเดรสในส่วนของไมโครคอนโทรลเลอร์เพื่อบอกตำแหน่งที่ต้องการติดต่อ รีจิสเตอร์ที่จัดอยู่ในกลุ่มนี้ประกอบด้วยรีจิสเตอร์ใช้งานทั่วไป (GENERAL–PURPOSE REGISTERS) ซึ่งรีจิสเตอร์ในกลุ่มนี้จัดเป็นพื้นที่หน่วยความจำ ที่ใช้ในการสนับสนุนการประมวลผลการทำงานจากหน่วยประมวลผลทางคณิตศาสตร์และลอจิก (ALU) เพื่อให้สามารถจัดการข้อมูลให้เร็วที่สุด นอกจากนี้โปรแกรมที่ไม่ได้ใช้คำสั่งเหล่านี้ก็ยังใช้เป็นการเก็บข้อมูลตัวแปรภายในโปรแกรม ซึ่งจะเห็นได้ว่าชื่อของรีจิสเตอร์ไม่ว่าจะอยู่ในรีจิสเตอร์แบงก์ใดก็จะมีชื่อว่า R0 ถึง R7 เหมือนกันทั้งสิ้น ดังนั้นในการใช้งานผู้ใช้จะต้องให้ความระมัดระวังว่า ต้องการรีจิสเตอร์นั้น ๆ จากแบงก์ใด ๆ ซึ่งการกำหนดเลือกแต่ละกลุ่มของรีจิสเตอร์นี้ก็ทำได้ง่าย เพียงการกำหนดค่าของบิตที่อยู่ภายในแฟลก (PSW) เท่านั้น อย่างไรก็ตามโดยทั่วไปก็มักจะมีการใช้งานรีจิสเตอร์ R0 ถึง R7 เฉพาะในแบงก์ 0 เท่านั้น ดังนั้นพื้นที่ของแบงก์อื่น ๆ ที่เหลือก็สามารถนำมาใช้ในลักษณะของหน่วยความจำแรม

2.1.6 โครงสร้างการอินเตอร์รัปต์ของไมโครคอนโทรลเลอร์ MCS–51

โครงสร้างระบบอินเตอร์รัปต์ของไมโครคอนโทรลเลอร์ MCS–51 สัญญาณที่เข้ามาทำการอินเตอร์รัปต์ MCS–51 นั้นเกิดขึ้นได้ 5 ลักษณะ โดยจะเห็นได้ว่าสามารถที่จะกำหนดเลือกเพื่อยินยอม (หรืออีนาเบิล : ENABLE) และห้าม (หรือดิสเอเบิล : DISABLE) ไม่ให้มีการอินเตอร์รัปต์แต่ละประเภทได้ โดยการกำหนดบิตของข้อมูลที่เกี่ยวข้องซึ่งมักจะอยู่ภายในรีจิสเตอร์ TCON และ SCON นอกจากนี้ยังมีตำแหน่งบิตภายในรีจิสเตอร์ IE (INTERRUPT ENABLE REGISTER) ซึ่งทำหน้าที่เสมือนกับเป็นสวิตซ์หลัก ที่เกี่ยวข้องกับสัญญาณอินเตอร์รัปต์ทั้งหมด หากว่ากำหนดไม่ให้เกิดการอินเตอร์รัปต์แล้วการกำหนดบิตเพื่อห้าม หรือยินยอมของแต่ละอินเตอร์รัปต์ก็จะไม่มีผลใด ๆ เกิดขึ้นยังแสดงให้เห็นว่าสัญญาณอินเตอร์รัปต์แต่ละประเภทนั้นยังสามารถกำหนดระดับความสำคัญ (PRIORITY) ของการอินเตอร์รัปต์ได้สองลักษณะ คือ ระดับความสำคัญสูงหรือต่ำ (HIGH OR LOW PRIORITY) กล่าวคือ ขณะที่กำลังประมวลผลอยู่ ภายในส่วนของโปรแกรมย่อยบริการอินเตอร์รัปต์ของสัญญาณที่มีระดับความสำคัญต่ำอยู่ ก็อาจจะถูกขัดจังหวะให้ไปทำการประมวลผลของสัญญาณอินเตอร์รัปต์ที่มีระดับความสำคัญสูงกว่า แต่หากว่าเป็นสัญญาณอินเตอร์รัปต์ที่มีระดับความสำคัญต่ำเช่นเดียวกันก็ต้องรอให้เสร็จสิ้นการประมวลผลที่ดำเนินการอยู่ก่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.7 รีจิสเตอร์ฟังก์ชันพิเศษ

ตารางที่ 2.1 รีจิสเตอร์ฟังก์ชันพิเศษ(SFRs) ของ MCS-51

Symbol	Name	Address (Hex)
ACC	Accumulator	EO
B	B register	FO
PSW	Program Status Word	DO
SP	Stack Pointer	81
DPTR DPL DPH	Data Pointer 2 byte Low byte High byte	 82 83
P0	Port 0	80
P1	Port 1	90
P2	Port 2	A0
P3	Port 3	B0
IP	Interrupt Priority Control	B8
IE	Interrupt Enable Control	A8
TMOD	Timer/Counter Mode Control	89
TCON	Timer/Counter Control	88
TH0	Timer/Counter 0 High byte	8C
TL0	Timer/Counter 0 Low byte	8A
TH1	Timer/Counter 1 High byte	8B
TL1	Timer/Counter 1 Low byte	8D
SCON	Serial Control	99
SBUF	Serial Data Buffer	99
PCON	Power Control	87

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รีจิสเตอร์เหล่านี้จะใช้พื้นที่หน่วยความจำข้อมูลตั้งแต่แอดเดรสที่ 80H ถึง 0FFH (ไบต์ที่ 128 จนถึงไบต์ที่ 255) ส่วนมากจะใช้สำหรับการคำนวณกับ ALU ส่วนรีจิสเตอร์อื่นที่เหลือจะถูกใช้ในการติดต่อกับอุปกรณ์ภายนอกรีจิสเตอร์ที่เกี่ยวข้องกับ ALU มีดังนี้

2.1.7.1 Accumulator

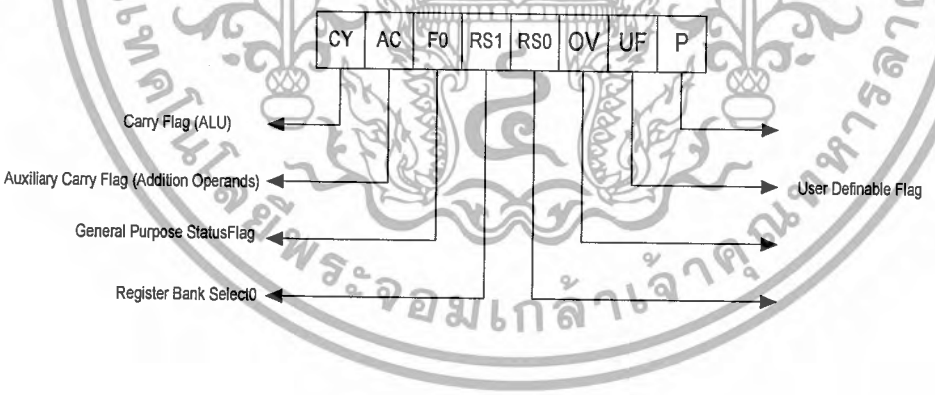
ACC คือ accumulator ซึ่งจะมีตำแหน่งในหน่วยความจำภายในเท่ากับ 0E0H มีขนาด 8 บิต เป็น Register ที่ใช้กันมาก ในรหัสคำสั่งช่วยจำจะใช้อักษร A

2.1.7.2 B Register

จะมีตำแหน่งในหน่วยความจำภายในเท่ากับ 0F0H มีขนาด 8 บิต จะใช้ในการคูณและการหารเท่านั้นหรืออาจใช้ในการเก็บข้อมูลอื่น ๆ ที่ต้องการก็ได้

2.1.7.3 Program status word (PSW)

จะมีตำแหน่งในหน่วยความจำภายในเท่ากับ D0H มีขนาด 8 บิต แต่ละบิต จะบอกถึงสภาวะต่าง ๆ ในการทำงานของ CPU(Flag) ซึ่งแต่ละบิต ของ PSW สามารถกำหนดให้เป็น 1 หรือ 0 ได้โดยคำสั่ง SETB หรือ CLRB ตามลำดับค่าตำแหน่ง bit address 0 ถึง bit 7 ของ PSW เท่ากับ D0H ถึง D7H



รูปที่ 2.7 แสดงรีจิสเตอร์ PSW

2.1.7.4 Stack Pointer

จะมีตำแหน่งในหน่วยความจำภายในเท่ากับ 081H มีขนาด 8 บิต Register นี้ใช้ชี้ตำแหน่งในหน่วยความจำภายใน 8051 ที่จะใช้สร้างเป็น Stack ในการทำงานของ MCS-51 ค่าของ SP นี้จะมีค่าที่เพิ่มขึ้น ก่อนที่จะมีการเก็บข้อมูลเข้าไปด้วยคำสั่ง PUSH หรือ CALL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การกำหนดตำแหน่งของ Memory ที่จะสร้างเป็น Stack นั้นสามารถกำหนดให้เป็นที่ใดก็ได้ใน Internal data memory และเมื่อทำการ Reset ค่าเริ่มต้นของ SP จะค่าเป็น 07H ซึ่งจะทำให้การเก็บข้อมูลในตำแหน่งแรกของ Stack เริ่มที่ Internal data memory ตำแหน่งที่ 08

2.1.7.5 Data Pointer Register

Data Pointer (DPTR) จะอยู่ในตำแหน่งหน่วยความจำภายในเท่ากับ 82H และ 83H DPTR นี้ประกอบไปด้วย Register ขนาด 8 บิต 2 ตัวคือ DPH และ DPL ซึ่ง DPTR นี้ จะใช้ในการชี้ตำแหน่งของข้อมูลของ External data memory แบบ 16 บิต ในการแก้ไขข้อมูลใน Register DPTR จะทำได้ทีละ 16 บิต หรือกระทำทีละ 8 บิตก็ได้ (DPH, DPL)

2.1.7.6 Port 0 ถึง 3

ตำแหน่งในหน่วยความจำภายในเท่ากับ 80H, 90H, 0A0H, 0B0H เป็น Register ขนาด 8 บิต การเขียนข้อมูลไปยังหน่วยความจำแต่ละตำแหน่งเป็นการส่งข้อมูลไปยังพอร์ตนั้น ๆ ของ MCS-51 ข้อมูลที่เขียนออกไปจะถูกแลตช์ค้างไว้ที่ Register นี้ และปรากฏแต่ละบิตของ Port เช่น ถ้าเขียนข้อมูล 18H ไปในหน่วยความจำตำแหน่ง 80H ก็จะปรากฏ Logic 0001 1000 ที่ขา 7 ถึง 0 ของ Port 0 ในการอ่านข้อมูลจาก Register แต่ละตัวจะเป็นการอ่านสภาวะ Logic ที่มีปรากฏอยู่แต่ละขาของ Port นั้น ๆ

2.1.7.7 Serial Data Buffer (SBUF)

ตำแหน่งหน่วยความจำภายในเท่ากับ 99H มีขนาด 8 บิต แต่จากโครงสร้างภายในแล้วมันคือ Register 2 ตัวที่มีชื่อเดียวกัน ตัวหนึ่งสำหรับเก็บข้อมูลที่จะส่งแบบอนุกรม และอีกตัวหนึ่งสำหรับรับข้อมูลแบบอนุกรมที่เข้ามา ในการเขียนข้อมูลเข้าที่ SBUF มันจะถูกเขียนไปยังที่สำหรับเก็บข้อมูลสำหรับส่ง และเริ่มต้นการส่งข้อมูล ส่วนการอ่านข้อมูลจาก SBUF ก็จะเป็นการอ่านค่าของข้อมูลที่ได้รับเข้ามา

2.1.7.8 Control Register

สำหรับการกำหนดรูปแบบการใช้พลังงานของตัว CPU เอง ตัว SFR ที่ชื่อ IP, IE จะกำหนดรูปแบบการทำงาน และสถานะของการ Interrupt TMOD, TCON, T2CON จะกำหนดรูปแบบการทำงานของ Timer/Counter และ SCON จะกำหนดรูปแบบการทำงานของ Serial port

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.8 ไทเมอร์/เคาน์เตอร์ของไมโครคอนโทรลเลอร์ MCS – 51

ไทเมอร์/เคาน์เตอร์ เป็นอีกส่วนหนึ่งที่สำคัญของไมโครคอนโทรลเลอร์ เนื่องจากในการทำงานของไมโครคอนโทรลเลอร์จะต้องมีการเก็บ และตรวจสอบค่าของเวลาและจำนวนสัญญาณนาฬิกาอยู่ตลอดเวลา ทั้งนี้เพื่อประโยชน์ในการสร้างฐานเวลา สร้างสัญญาณพัลส์ เปรียบเทียบค่าเวลา หรือ ค่าการนับ รวมไปถึงการกำหนดอัตราเร็วในการสื่อสารข้อมูลของพอร์ตอนุกรมด้วย

ในการใช้งานไทเมอร์/เคาน์เตอร์นั้น จะต้องมีการกำหนดหรือควบคุมการทำงานของไทเมอร์/เคาน์เตอร์ โดยใช้รีจิสเตอร์ที่สำคัญ 2 ตัว คือ

TCON (Timer / Counter Control Register) : มีแอดเดรสอยู่ที่ 88H เป็นรีจิสเตอร์ที่ใช้ควบคุมการเปิด/ปิดไทเมอร์แต่ละตัว,และแสดงถึงการเกิดโอเวอร์โฟรวที่เกิดจากไทเมอร์/เคาน์เตอร์ อีกทั้งเป็นตัวกำหนดลักษณะการเกิดอินเตอร์รัปต์ของสัญญาณภายนอกด้วย

TMOD (Timer / Counter Mode Control Register) : มีแอดเดรสอยู่ที่ 89H เป็นรีจิสเตอร์ที่ใช้กำหนดการทำงานว่าเป็นไทเมอร์ หรือ เคาน์เตอร์ อีกทั้งเป็นตัวกำหนดโหมดการทำงานของไทเมอร์/เคาน์เตอร์ ซึ่งมีอยู่ 4 โหมดด้วยกัน คือ

โหมด 0 : ไทเมอร์/เคาน์เตอร์ 13 บิต

โหมด 1 : ไทเมอร์/เคาน์เตอร์ 16 บิต

โหมด 2 : ไทเมอร์/เคาน์เตอร์ 8 บิต แบบตั้งค่าอัตโนมัติ

โหมด 3 : ไทเมอร์/เคาน์เตอร์แยกส่วน

2.1.9 พอร์ตอนุกรมของไมโครคอนโทรลเลอร์ MCS–51

ไมโครคอนโทรลเลอร์ตระกูล MCS–51 มีวงจรสื่อสารอนุกรมแบบฟลูดูเพล็กซ์ 1 ชุด โดยใช้ขาสัญญาณของพอร์ต 3 คือ ขา P3.0 เป็นขารับข้อมูลหรือ RxD และขา P3.1 เป็นขาส่งข้อมูลหรือ TxD โดยวงจรสื่อสารข้อมูลแบบอนุกรมของ MCS–51 เป็นแบบอะซิงโครนัส โดยปกติแล้วพอร์ตอนุกรมจะใช้ติดต่อสื่อสารกับพอร์ตอนุกรมของคอมพิวเตอร์ โดยใช้มาตรฐาน RS–232 แต่ในปัจจุบัน สามรถติดต่อกับ RS-422 หรือ RS-485 ได้แล้ว โดยใช้ IC พิเศษ ทำหน้าที่ในการแปลงสัญญาณการสื่อสาร

รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของพอร์ตอนุกรมใน MCS–51 ได้แก่ SUBF (Serial data buffer register) ซึ่งเป็นบัฟเฟอร์สำหรับ ส่งข้อมูล และรับข้อมูล และรีจิสเตอร์อีกตัวที่สำคัญคือ SCON (Serial Port Control)

ซึ่งใช้ในการกำหนดโหมดการทำงานของพอร์ตอนุกรม ซึ่งมี 4 โหมด ด้วยกัน คือ

1. โหมด 0 เป็นการกำหนดให้พอร์ตอนุกรมทำงานในลักษณะชิฟรีจิสเตอร์
2. โหมด 1 เป็นการกำหนดให้ UART ขนาด 8 บิต สามารถเลือกอัตราบอร์ดได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. โหมด 2 UART ขนาด 8 บิต สามารถเลือกอัตราบอร์ดได้

4. โหมด 3 เป็นตัวกำหนดให้เป็น UART ขนาด 9 บิต สามารถเลือกอัตราบอร์ดได้

2.2 GPRS

GPRS เป็นตัวย่อจากภาษาอังกฤษ " General Packet Radio Service" ซึ่งเป็นการส่งข้อมูลในรูปแบบแพ็กเก็ตต่าง ๆ การเชื่อมต่อแบบใหม่ที่ใช้ระบบ GPRS เข้ามาก็จะเป็นการเชื่อมต่อ และวิธีการส่งข้อมูลที่มีลักษณะเช่นเดียวกับอินเตอร์เน็ตก็คือ เมื่อต้องการข้อมูลหรือส่งข้อมูลอะไรก็แล้วแต่ จะเป็นการส่งข้อมูลลักษณะนั้นเข้าไปในเครือข่ายโทรศัพท์เคลื่อนที่เท่านั้น ไม่จำเป็นต้องจองเวลาไว้ตลอดเวลา จึงทำให้วิธีการใช้งานของ GPRS ในแบบใหม่นี้จะเห็นได้ว่าจะมีการพูดถึงการเก็บเงินที่เป็นจำนวนข้อมูลที่รับและส่งออกมา มากกว่าวิธีการติดต่อสื่อสารจากวิธีเดิมที่คิดจำนวนเวลาในการติดต่อสื่อสารแต่ละครั้ง

การติดต่อด้วยระบบ GPRS สามารถติดต่อสื่อสารได้ด้วยเสียง ในขณะที่เราสามารถติดต่อสื่อสารผ่านโลกอินเตอร์เน็ตในขณะเดียวกัน ซึ่งก็คือ เราสามารถติดต่อสื่อสารทั้ง 2 ระบบภายในช่วงเวลาเดียวกัน แต่ทั้งนี้ต้องขึ้นอยู่กับโทรศัพท์เคลื่อนที่ในแต่ละรุ่นที่ผลิตออกมา แต่เท่าที่ทราบในขณะนี้โทรศัพท์เคลื่อนที่แต่ละรุ่นยังไม่สามารถติดต่อสื่อสารพร้อม ๆ กันได้ GPRS เชื่อมโลกอินเตอร์เน็ต บนโทรศัพท์เคลื่อนที่ GPRS ไม่ได้เป็นลักษณะที่จะสามารถให้บริการได้ด้วยตัวของระบบเอง แต่ตัวมันเองเป็นเพียงแค่ Bearer ให้กับ Application ต่าง ๆ ที่ต้องการใช้ความเร็วที่เพิ่มมากกว่าปกติในระบบ GSM ที่เคยรองรับอยู่เดิมมาก่อน และระบบ GPRS จะต้องต่อไปยัง Packet Data Network ที่เป็น IP Network อีกต่อหนึ่ง

ดังนั้นผู้ให้บริการเครือข่ายโทรศัพท์เคลื่อนที่ ที่จะเปิดใช้ในระบบ GPRS ได้นั้นจะต้องทำการติดตั้งระบบเครือข่ายที่ประกอบด้วยหน่วยหลัก ๆ 2 หน่วย ด้วยกันคือ

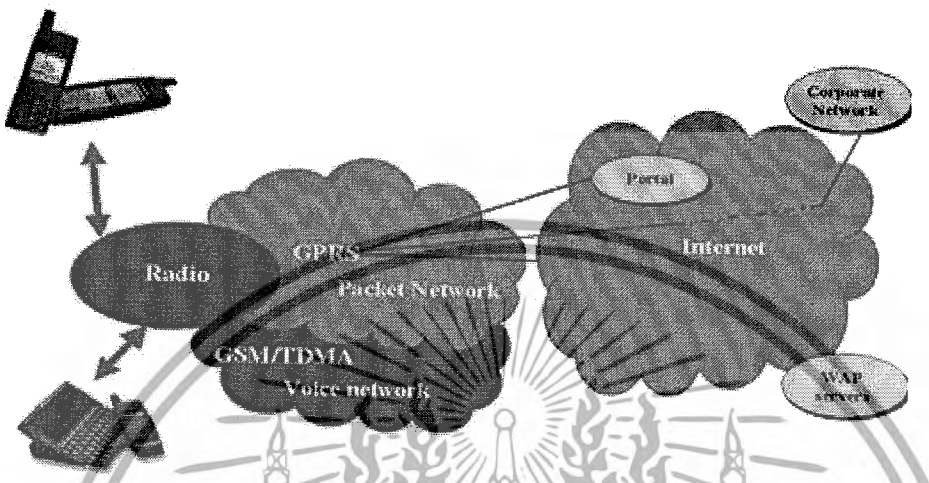
1. SGSN(Serving GPRS Supports Node)
2. GGSN(Gateway GPRS Supports Node)

โดยทั้งสองหน่วยหลักขององค์ประกอบนี้ จะถูกเชื่อมต่อเข้าด้วยกันโดยมีอุปกรณ์อื่น ๆ เป็นตัวช่วยเพื่อไปร่วมใช้ Radio Interface จาก Base Station โดยผ่านตัวควบคุมที่เรียกว่า PCU (Packet Control Unit) ที่ติดตั้งไว้ที่ BSC (Base Station Controller) ทั้งนี้อาจมอง Network นี้เป็นอีก Network หนึ่ง ซึ่งเข้ากับ Mobile Phone ผ่านทาง Radio Interface ของระบบ GSM Network เดิมโดยเป็นบริการที่เกี่ยวเนื่องกับการรับส่งข้อมูลเป็น Packet โดยตรง

บริการส่งข้อมูลสำหรับโทรศัพท์มือถือแบบ GSM GPRS นั้น ได้รับการพัฒนาโดย 3GP (Third Generation Partnership Project) และเป็นการเชื่อมต่อเครือข่ายโทรศัพท์เคลื่อนที่โดยใช้ PDN (Public Data Network) ภายในเครือข่าย GSM โดยจะเป็นการรับส่งข้อมูล Data (ปกติเราใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลแบบ Voice คือโทรเข้า รับสาย และส่งข้อความ SMS) โดยจะมีการแบ่งสรรข้อมูลที่เรียกกันว่า CPU (Packet Control Unit), SGSN (Serving GPRS Supports Node)และ GGSN (Gateway GPRS Supports Node) เพื่อจัดการการรับส่งข้อมูลระหว่างเครื่องรับโทรศัพท์และเครือข่ายต้นทาง



รูปที่ 2.8 แสดงเครือข่ายการทำงานของ GPRS

คุณสมบัติเด่น ๆ ของระบบ GPRS

1. การโอนถ่ายข้อมูลที่มีความสามารถในการทำรับ-ส่ง โดยผ่านเครือข่ายอินเตอร์เน็ตได้สูงถึง 40 kbps (รวดเร็วขึ้นกว่าระบบ CSD ซึ่งเป็นระบบส่งข้อมูลแบบเก่าที่สามารถส่งได้เพียง 9.6 Kbps ถึง 10 เท่า) ซึ่งจะทำให้สามารถรับ-ส่งข้อมูลที่เป็น VDO Mail หรือภาพเคลื่อนไหวต่าง ๆ ได้ พร้อมทั้งกับทำการเชื่อมต่อกับเครือข่ายอินเตอร์เน็ตได้เร็ว และมีประสิทธิภาพมากกว่าเดิมรวมถึงการ Down lode/Up lode ได้ง่ายยิ่งขึ้น

2. Always On นั่นหมายความว่าคุณสามารถ Connect เข้าสู่โลกอินเตอร์เน็ตได้ตลอดเวลา โดยไม่มีความจำเป็นต้องทำการ Log On หรือใช้ User Name และ Password เหมือนการ connect เข้าสู่อินเตอร์เน็ตทั่วไปแต่อย่างใด ทั้งนี้ยังสามารถที่จะเปลี่ยน Mode มาทำการโทรออกหรือรับสายเข้าโดยคุณยังสามารถ Connect กับโลกอินเตอร์เน็ตในระบบ GPRS อยู่ตลอดเวลา

3. Wireless Internet ที่เชื่อมต่อเข้ากับ Terminal เช่น PDA หรือ Note Book สามารถที่จะโอนถ่ายข้อมูลได้เร็วขึ้นจากที่เคยเป็นอยู่

4. เรื่องค่าบริการของในการต่อเชื่อมกับโลกอินเตอร์เน็ตผ่านเทคโนโลยี GPRS นั้นไม่แพงอย่างที่คิด ทั้งนี้เพราะการคิดค่าบริการในระบบนี้จะเกิดขึ้นก็ต่อเมื่อคุณได้ทำการรับหรือส่งข้อมูลเท่านั้น คุณจะไม่มีค่าใช้จ่ายใด ๆ ในกรณีที่คุณไม่ได้ทำการรับส่งข้อมูล ถึงแม้คุณจะ Connect กับโลกอินเตอร์เน็ตอยู่ก็ตาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

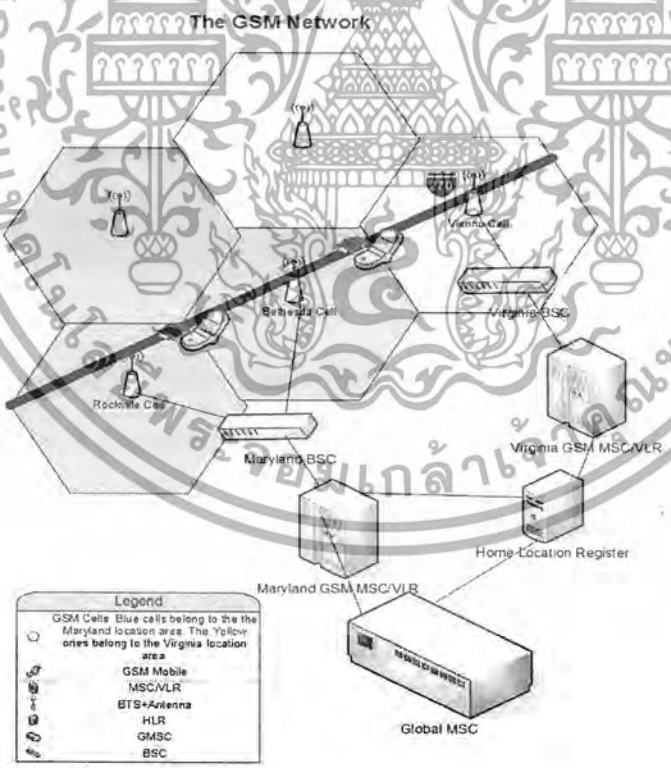
สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

2.2.1 เครือข่าย GSM

GSM ย่อมาจาก Global System for Mobile Communications เป็นมาตรฐานของเทคโนโลยีโทรศัพท์มือถือที่ได้รับความนิยมมากที่สุดในโลก ปัจจุบันมีผู้ใช้มากกว่า 2 พันล้านคนใน 213 ประเทศ ถึง 84.2% ของมือถือทั่วโลก ประเทศจีนมีผู้ใช้งานเป็นอันดับหนึ่งของโลก มากกว่า 370 ล้านคน ตามด้วยประเทศรัสเซีย 145 ล้านคน, อินเดีย 83 ล้านคน และสหรัฐอเมริกาถึง 78 ล้านคน GSM เป็นมาตรฐานเปิดภายใต้การดูแลของ 3GPP

1. GSM ใช้เทคโนโลยีดิจิทัลสำหรับช่องสัญญาณควบคุมและสัญญาณเสียงแบบTDMA ซึ่งแตกต่างจากเทคโนโลยีโทรศัพท์มือถือก่อนหน้านั้น จึงถือว่าเป็นโทรศัพท์มือถือในยุคที่สอง หรือ 2G ซึ่งหมายถึง การพัฒนาระบบขึ้นไปอีกขั้นหนึ่ง

การพัฒนาอย่างแพร่หลายของ GSM เป็นประโยชน์ต่อผู้บริโภคที่สามารถใช้งานได้อย่างสะดวกสบายมากขึ้น และนอกจากนี้ยังเป็นประโยชน์ต่อผู้ควบคุมระบบเนตเวิร์ค ให้มีตัวเลือกในการใช้งานมากขึ้น เนื่องจากมีผู้จัดทำอย่างแพร่หลาย GSM เริ่มต้นด้วยทางเลือกใหม่ ซึ่งมีราคาที่ถูกเป็นอีกหนึ่งทางเลือกในการติดต่อสื่อสาร นั่นก็คือ Short message service(SMS) หรือเรียกอีกอย่างว่าเท็กเมสเสสจิ้ง ซึ่งโทรศัพท์มือถือทั่วไปสามารถรองรับได้อย่างดี



รูปที่ 2.9 แสดงเครือข่าย GSM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
104118

2.2.2 การควบคุม GPRS Module

AT Command เป็น Protocol ที่โทรศัพท์เคลื่อนที่ส่วนใหญ่ยึดใช้เป็น Protocol หลัก ในการอินเตอร์เฟสกับคอมพิวเตอร์ ซึ่ง AT Command เป็นชุดคำสั่งที่มีไว้สำหรับโมเด็มโดยเฉพาะ โดยคอมพิวเตอร์จะมองโทรศัพท์เคลื่อนที่เป็นโมเด็มเหมือนกัน โดยที่การเชื่อมต่อด้านฮาร์ดแวร์นั้นจะนิยมใช้ 2 ช่องทาง คือ

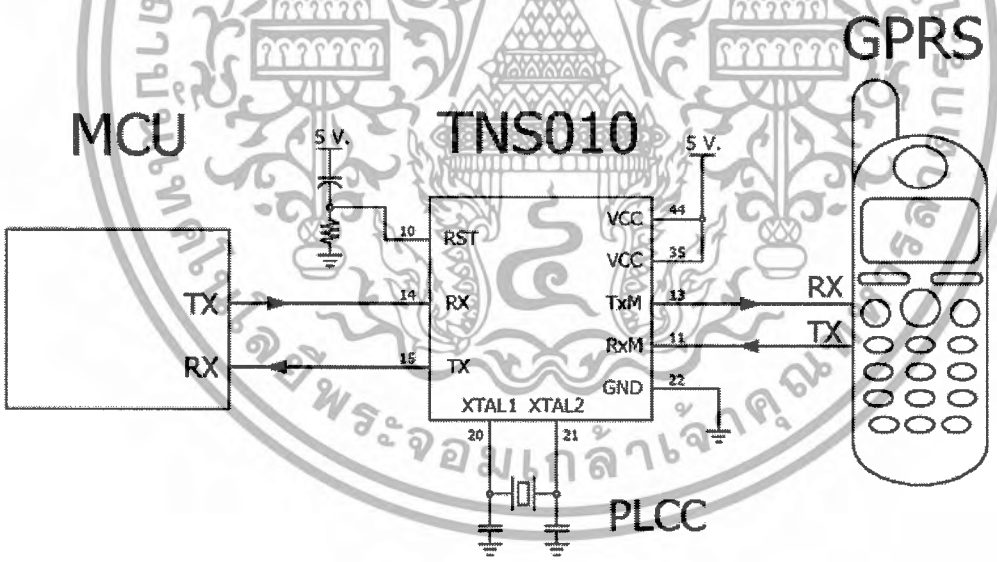
1. Serial Port
2. IrDa (Infrared Port)

2.3 รูปแบบการเชื่อมต่อ

2.3.1 การติดต่อสื่อสารระหว่างไมโครคอนโทรลเลอร์กับ GPRS Module

การติดต่อสื่อสารระหว่างไมโครคอนโทรลเลอร์กับ GPRS Module นั้นใช้การสื่อสารผ่านพอร์ตอนุกรม RS232 ทาง GPRS Module โดยผ่านทางบอร์ด TCP/IP Stack Chip

2.3.1.1 TCP/IP Stack Chip



รูปที่ 2.10 แสดงการทำงานของ TCP/IP Stack Chip

2.3.1.2 การสื่อสารแบบอนุกรม

การสื่อสารแบบอนุกรม แบ่งออกได้เป็น 2 แบบ คือ การสื่อสารอนุกรมแบบซิงโครนัสและการสื่อสารอนุกรมแบบอะซิงโครนัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. การสื่อสารอนุกรมแบบซิงโครนัส (Synchronous)

สำหรับการสื่อสารแบบซิงโครนัสนี้ จะใช้สัญญาณนาฬิกาควบคุมการรับ-ส่งสัญญาณ เช่น สายคีย์บอร์ดคอมพิวเตอร์ โดยจะมีสายสัญญาณเส้นหนึ่งเป็นสายสัญญาณนาฬิกา ส่วนอีกเส้นหนึ่งเป็นสายของข้อมูล(และมักจะมีสายกราวน์ด้วย) สำหรับการสื่อสารแบบซิงโครนัสเหมาะสำหรับการทำงานในระยะใกล้ข้อมูลที่จะส่งมีไม่มากนัก เพราะถ้าระยะทางไกลขึ้นจะทำให้สัญญาณนาฬิกามีปัญหาอีก ทั้งต้องมีสายหลายเส้นทำให้สิ้นเปลืองมาก

2. การสื่อสารแบบอะซิงโครนัส(Asynchronous)

สำหรับการสื่อสารแบบอะซิงโครนัสนั้น จะใช้สายสัญญาณเพียงตัวเดียวแต่จะใช้รูปแบบการส่งข้อมูล หรือ Bit Pattern เป็นตัวกำหนดว่าส่วนไหนเป็นตัวเริ่มต้นข้อมูล ส่วนไหนเป็นตัวข้อมูล ส่วนไหนจะเป็นตัวตรวจสอบความถูกต้องของข้อมูล และส่วนไหนเป็นส่วนปิดท้ายของข้อมูล โดยต้องกำหนดให้สัญญาณนาฬิกาเท่ากันทั้งภาครับและภาคส่ง ซึ่งจะมีอุปกรณ์พิเศษที่เรียกว่า UART หรือ Universal Asynchronous Reciever/Trasmitter คอยควบคุมการรับและการส่งข้อมูล อีกแบบที่ได้รับความนิยมอย่างสูงตั้งแต่ อดีตจนถึงปัจจุบันนี้ก็คือมาตรฐาน RS -232C

รูปแบบของข้อมูลอนุกรมแบบอะซิงโครนัส เมื่อไม่มีข้อมูลที่จะส่งขา DATA จะมีสถานะลอจิก “ 1 ” วงจรจะเรียกสถานะนี้ว่าสถานะหยุดรอ (Waiting Stage) การเริ่มต้นส่งข้อมูลจะเริ่มให้จากการให้ขา DATA มีลอจิก “ 0 ” ด้วยช่วงระยะเวลา 1 บิต เรียกบิตนี้ว่า บิตเริ่มต้น จากนั้นบิตข้อมูลจะถูกส่งออกไป โดยเริ่มจากบิตที่มีนัยสำคัญต่ำสุด (LSB) ก่อน ซึ่งข้อมูลในไบต์ที่จะส่งอาจจะมีจำนวนบิต 5, 6, 7 หรือ 8 บิตก็ได้ จากนั้นก็ตามด้วยบิตพาริตี้ ซึ่งใช้ตรวจสอบความผิดพลาดที่อาจจะเกิดขึ้นกับการส่งข้อมูล บิตสุดท้ายที่ส่งคือ บิตปิดท้าย ซึ่งจะให้สถานะขาดาต้ามีสถานะลอจิก “ 1 ” อีกครั้ง ด้วยระยะเวลาอย่างน้อย 1 บิต, 1.5 บิต หรือ 2 บิต เพื่อเป็นการแสดงว่าสิ้นสุดข้อมูลแล้ว

อุปกรณ์พิเศษ ที่ได้รับการออกแบบมาใช้งานสำหรับทำการรับและทำกาส่งข้อมูลแบบอะซิงโครนัสเรียกว่า Universal Asynchronous Receiver / Transmitter หรือ UART อัตราความเร็วในการรับและส่งข้อมูลของการรับส่งข้อมูลแบบอะซิงโครนัส คือ ค่าบอดเรต ซึ่งก็คือค่าจำนวนบิตต่อวินาทีที่ใช้ในการรับและส่งข้อมูล บอดเรตมาตรฐานที่ใช้ส่งพอร์ตอนุกรม RS-232 ได้แก่ 110, 150, 300, 600, 1200, 2400, 4800, 9600 และ 19200 บิตต่อวินาที และมีค่าเพิ่มมากขึ้นตามเทคโนโลยีของคอมพิวเตอร์ ซึ่งการรับส่งแบบอนุกรมโดยไม่ผ่านโมเด็มอาจจะสามารถกำหนดค่าบอดเรตได้สูงถึง 115,200 บิตต่อวินาที เนื่องจากบอดเรตคือจำนวนบิตของข้อมูลที่สามารถถ่ายทอดได้ภายใน 1 วินาที ยกตัวอย่าง ข้อมูลอนุกรมถูกส่งในลักษณะ 8 บิต ไม่มีการตรวจสอบพาริตี้ มีบิตเริ่มต้น 1 บิต และบิตปิดท้าย 1 บิต ความยาวของข้อมูลที่รับส่งนี้เท่ากับ 10 บิต ถ้าใช้อัตราการถ่ายทอดข้อมูลในการส่งข้อมูลเท่ากับ 9600 บิตต่อวินาทีก็จะสามารถรับส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ได้ด้วยความเร็ว 960 ไบต์ต่อวินาที และถ้ามีการใช้พาริตี้ความเร็วในการรับส่งข้อมูลจะเหลือเป็น 872 ไบต์ต่อวินาที

การตรวจสอบพาริตี้สามารถกำหนดให้เป็นแบบคี่ (odd), แบบคู่ (even) หรือไม่มีการตรวจสอบพาริตี้ก็ได้ พาริตี้คู่หรือพาริตี้คี่แสดงถึงจำนวนลอจิก "1" ทั้งหมดภายในที่ส่งไป 1 ไบต์ รวมบิตพาริตี้ว่าจำนวนเลคู่หรือเลขคี่ ยกตัวอย่าง ข้อมูลที่จะทำการส่งมีขนาด 8 บิต มีค่าเท่ากับ 99H หรือ 10011001B จะเห็นว่าข้อมูลในไบต์นี้มีจำนวนลอกจิก "1" จำนวน 4 ตัว ซึ่งเป็นเลขคู่ ดังนั้น ถ้ากำหนดค่าพาริตี้เป็นคู่ ค่าของพาริตี้จะต้องมีลอจิกเป็น "0" แต่ถ้ากำหนดพาริตี้เป็นคี่ ค่าของพาริตี้จะต้องบเป็น "1" เพื่อให้ข้อมูล 1 ไบต์รวมทั้งพาริตี้เป็นคี่

คอมพิวเตอร์ในรุ่น AT เกือบทั้งหมดจะใช้ไอซี UART เบอร์ 16450 และ 16550 ส่วนคอมพิวเตอร์ในรุ่น XT ใช้ไอซี UART เบอร์ 8250 ไอซี UART เหล่านี้มีระดับแรงดันของลอจิกเป็นแบบทีทีแอล (+5V) แต่เพื่อให้มีแรงดันเป็นไปตามมาตรฐาน RS-232 และเพื่อให้การรับส่งข้อมูลสามารถทำได้ที่ระยะทางไกลมากขึ้น ระดับแรงดันทีทีแอลจะถูกดัดแปลงเป็นระดับแรงดันที่สูงขึ้นโดยลอจิก "0" จะมีระดับแรงดัน -3V ถึง -12V และลอจิก "1" มีระดับแรงดัน +3V ถึง +12V

2.3.1.3 มาตรฐานพอร์ทอนุกรมแบบ RS-232

มาตรฐานการเชื่อมต่อแบบอนุกรม RS-232 เป็นมาตรฐานอุตสาหกรรมที่ออกแบบมาเพื่อใช้ในการส่งข้อมูลอนุกรมแบบอะซิงโครนัส 2 ทิศทาง โดยมาตรฐาน RS-232 ในอดีตนั้นถูกออกแบบมาเพื่อการส่งผ่านข้อมูลจากคอมพิวเตอร์ไปยังโมเด็มเพียงอย่างเดียว เพื่อที่จะนำข้อมูลจากโมเด็มนี้สื่อสารผ่านสายโทรศัพท์ ไปยังคอมพิวเตอร์อีกชุดหนึ่งที่ห่างไกลกันโดยคณะกรรมการที่เรียกว่าสมาคมอุตสาหกรรมอีเล็กทรอนิกส์ (Electronic Industries Association: EIA) ได้วางมาตรฐานที่มีชื่อเรียกกันว่า EIA RS-232 มาตรฐานนี้ในช่วงแรกจะใช้คอนเน็กเตอร์เป็นแบบ DB-25 โดยกำหนดความยาวสูงสุดของสายสัญญาณสูงสุดไว้ที่ 50 ฟุต มีระดับสัญญาณตั้งแต่ -3 ถึง -12 V แสดงว่ามีข้อมูล (Mark) และ +3 ถึง +12 V แสดงว่าเป็นช่องว่าง (Space)

มาตรฐาน RS-232 ได้กำหนดรูปแบบของอุปกรณ์เชื่อมต่อข้อมูล (Data Terminal Equipment: DTE) กับวงจรข้อมูลปลายทาง (Data Circuit Terminating: DCE) ไว้ว่า DTE จะต้องเป็นอุปกรณ์ที่มีการประมวลผลในตัวเช่น ไมโครคอลโทรลเลอร์ หรือไมโครคอมพิวเตอร์ ซึ่งมีความสามารถในการสร้างข้อมูลแบบอนุกรมได้ ส่วนอุปกรณ์แบบ DCE จะทำหน้าที่เป็นตัวรับข้อมูลที่ส่งมาจาก DTE เท่านั้นโดยการส่งข้อมูลระหว่างอุปกรณ์ทั้งสองจะทำผ่านมาตรฐาน RS-232

ข้อแตกต่างระหว่างอุปกรณ์ DTE และ DCE อย่างหนึ่งที่เห็นได้ชัดคือ คอนเน็กเตอร์ของ DTE จะเป็นตัวผู้ ส่วนของ DCE จะเป็นตัวเมียซึ่งพอร์ตอนุกรมของคอมพิวเตอร์ที่ใช้กันอยู่ทั่วไปจะเป็นแบบ DTE ส่วนคอนเน็กเตอร์ที่อยู่ที่โมเด็มจะเป็นแบบ DCE

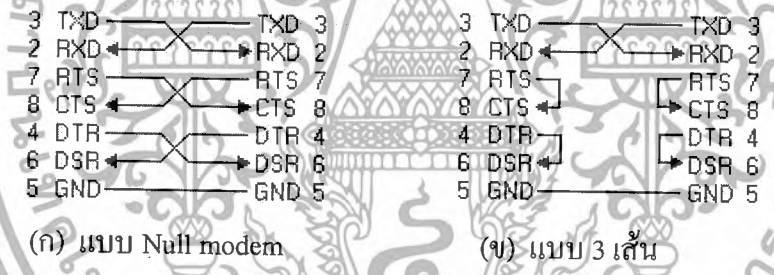
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับการใช้งานบนคอมพิวเตอร์ พอร์ตอนุกรม RS-232 มักถูกใช้เชื่อมต่อกับโมเด็มหรือเมาส์ โดยสามารถรับส่งข้อมูลได้ที่ความยาวของสายสัญญาณสูงสุดถึง 20 เมตร

2.3.1.4 คอนเน็กเตอร์สำหรับพอร์ต RS-232 และการเชื่อมต่อ

มาตรฐานการเชื่อมต่อแบบ RS- 232 จะใช้คอนเน็กเตอร์แบบ DB 25 ตัวผู้ หรือ DB 9 ตัวผู้ซึ่งคอนเน็กเตอร์แบบ DB 25 จะมีขาต่อใช้งานเพียง 9 เส้นเช่นเดียวกับคอนเน็กเตอร์แบบ DB 9 เนื่องจากขาอื่น ๆ ที่ใช้งานในอดีต ปัจจุบันมีการใช้งานไม่มากนักจึงถูกยกเลิกไป โดยแสดงรูปร่างและตำแหน่งขาในรูปที่ 2.11

สำหรับการเชื่อมต่อคอมพิวเตอร์กับอุปกรณ์ภายนอกแสดงดังในรูปที่ 2.11 ลูกศรในรูปแสดงถึงทิศทางข้อมูล ในรูปที่ 2.11 (ก) เป็นการเชื่อมต่อแบบ Null Modem หรือการเชื่อมต่อโดยตรงไม่ต้องผ่านโมเด็ม โดยมีการตรวจสอบหรือแฮนด์เช็กเต็มรูปแบบ ส่วนในรูปที่ 2.11 (ข) เป็นการเชื่อมต่อแบบ Null Modem ในลักษณะที่ใช้สายสัญญาณเพียง 3 เส้น โดยเส้นหนึ่งสำหรับส่งข้อมูล อีกเส้นสำหรับรับข้อมูล และเส้นสุดท้ายเป็นกราวด์ สำหรับรายระเอียดหน้าที่การทำงานในแต่ละขาของพอร์ตอนุกรม RS-232 มี ดังนี้



รูปที่ 2.11 การต่ออุปกรณ์ภายนอกกับพอร์ตอนุกรมของคอมพิวเตอร์ในลักษณะต่าง ๆ

1. ขา Data Carrier Detect : DCD หรืออาจจะเรียกว่า Carrier Detect: CD ขานี้จะแอกติฟเมื่อมีการส่งสัญญาณพาห์จากอุปกรณ์สื่อสารข้อมูลเช่นโมเด็ม สำหรับการใช้งานปกติ ขานี้จะไม่ถูกใช้งานมากนัก

2. ขา Receive Data : RD หรือ RXD ขานี้ใช้เพื่อรับสัญญาณอนุกรมเข้ามายังคอมพิวเตอร์โดยนำข้อมูลที่อ่านได้เก็บไว้ใน รีจิสเตอร์บัฟเฟอร์

3. ขา Transmitted Data: TD หรือ TXD ใช้ส่งข้อมูลออกจากคอมพิวเตอร์โดยนำข้อมูลที่เก็บอยู่ในบัฟเฟอร์สำหรับส่งข้อมูลส่งออกไป

4. ขา Data Terminal Ready: DTR เป็นขาสัญญาณที่ส่งออกจากคอมพิวเตอร์เพื่อให้อุปกรณ์ปลายทางรับรู้ว่าต้องการติดต่อด้วยโดยขา DTR นี้ต้องเชื่อมต่อกับขา DSR ของอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปลายทาง และขา DTR ของอุปกรณ์ทางต้องเชื่อมต่อกับขา DSR ของคอมพิวเตอร์ ถ้าใช้การเชื่อมต่อเป็นแบบ Null Modem ซึ่งใช้สายในการเชื่อมต่อเพียง 3 เส้น จะต้องต่อขา DTR และ DSR ของตัวมันเองเข้าด้วยกันและต้องต่อกับขา DCD ด้วยในกรณีที่โปรแกรมสื่อสารที่ใช้มีการตรวจจับสัญญาณพาร์

5. ขา Signal Ground: GND กราวด์ระบบ

6. ขา Data Set Ready: DSR ขานี้จะใช้คู่กับขา DTR เพื่อตรวจสอบการเชื่อมต่อกันของคอมพิวเตอร์กับอุปกรณ์ปลายทางซึ่งขา DSR นี้จะเป็นขาสำหรับข้อมูลจากภายนอกซึ่งถูกส่งมาจากขา DTR

7. ขา Request To Send: RTS เป็นขาสำหรับส่งสัญญาณร้องขอให้ทางอุปกรณ์ปลายทางส่งข้อมูลกลับมาทางคอมพิวเตอร์ โดยขาที่รับสัญญาณ RTS คือขา CTS ในกรณีที่ใช้การเชื่อมต่อแบบ Null Modem 3 สาย จะต้องเชื่อมต่อขา RTS และ CTS ของตัวมันเองเข้าด้วยกันเพื่อจะให้การรับและส่งข้อมูลสามารถเกิดขึ้นได้ตลอดเวลา

8. ขา Clear To Send: CTS ขานี้จะคอยรับสัญญาณจากขา RTS เมื่อรับสัญญาณได้ข้อมูลที่ขา TXD จะถูกส่งออกไป ดังนั้นขานี้จึงถูกใช้เพื่อตรวจสอบอุปกรณ์ต่อพ่วงว่าพร้อมที่จะรับข้อมูลหรือไม่

9. ขา Ring Indicator: RI ใช้แสดงสถานะสัญญาณเรียกจากโทรศัพท์ ปกติในการสื่อสารโดยทั่วไปสายนี้จะไม่ถูกใช้งาน จะใช้งานก็ต่อเมื่อมีการเชื่อมต่อกับโมเด็มและโปรแกรมมีการตรวจสอบสัญญาณนี้เท่านั้น

2.3.2 การเชื่อมต่อ GPRS Module กับ Server

2.3.2.1 อินเตอร์เนต

อินเตอร์เนต : คือ ระบบเน็ตเวิร์กขนาดใหญ่ที่ต่อเชื่อมระบบต่าง ๆ ทั่วโลก เข้าด้วยกัน เปรียบเสมือนห้องสมุดสาธารณะชนขนาดมหึมาที่ใหญ่ที่สุดในโลกก็ว่าได้ที่มีข้อมูลต่าง ๆ มากมาย และสามารถนำไปใช้ประโยชน์ด้านต่าง ๆ ได้ไม่ว่าจะเป็นในการค้นคว้าวิจัยหรือเพื่อความบันเทิง เป็นต้น

อินเตอร์เนต เป็นคำที่หลาย ๆ คนคุ้นหูและเป็นที่รู้จักกันดีในขณะนี้ นอกจากนั้นมันไม่ใช่เป็นแค่ระบบเน็ตเวิร์กขนาดใหญ่ธรรมดาเท่านั้น แต่มันเป็นอภิมหาเน็ตเวิร์กโลก(เน็ตเวิร์กที่เชื่อมต่อกันหลาย ๆ เน็ตเวิร์กจากทั่วทุกมุมโลกเข้าไว้ด้วยกัน) ที่ ๆ เสมือนถนนที่จะพาผู้ที่เป็นสมาชิกบนเน็ตเวิร์กไปได้ทุกแห่งทุกหนบนโลกอินเตอร์เนต

อินเตอร์เนตมีการพัฒนาการมาจาก อาร์พาเน็ต(ARPAnet หรือที่เรียกกัน สั้น ๆ ว่า อาร์พา) ที่ตั้งขึ้นในปี พ.ศ. 2512 เป็นเครือข่ายคอมพิวเตอร์ของกระทรวงกลาโหม ที่ใช้กันในงานวิจัยทางด้านการทหาร (ARPA : Advanced Research Project Agency) จนกระทั่งมาถึงในปี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พ.ศ. 2515 หลังจากที่เครือข่ายทดลองของอาร์พาประสบกับความสำเร็จเป็นอย่างมาก ก็ได้มีการปรับปรุงหน่วยงานจาก อาร์พา มาเป็น ดาร์พา (Defence Comunication Agency) และในปี พ.ศ. 2526 อาร์พาเน็ต ก็ได้แบ่งออกเป็น 2 เครือข่ายด้วยกัน คือ เครือข่ายด้านงานวิจัย ใช้ชื่อ อาร์พาเน็ต เหมือนเดิม ส่วนเครือข่ายของกองทัพใช้ มิลเน็ต (MILNET : Military Network) ซึ่งมีการเชื่อมต่อโดยใช้โพรโตคอล TCP / IP (Transmission Control Protocol / Internet Protocol) เป็นครั้งแรกในปีพ.ศ. 2528 โดยมูลนิธิวิทยาศาสตร์แห่งชาติของอเมริกา NSF ได้ให้เงินทุนในการสร้างศูนย์ซุปเปอร์คอมพิวเตอร์จำนวน 6 แห่ง และให้ใช้ชื่อว่า NSFNET จนกระทั่งพอมาถึงปีพ.ศ. 2533 อาร์พาเน็ต รองรับภาระที่เป็นกระดูกสันหลัง (BackBone) ของระบบไม่ได้ จึงได้ยุติอาร์พาเน็ตและเปลี่ยนไปใช้ NSFNET และเครือข่ายอื่น ๆ แทนจนมาเป็นเครือข่ายขนาดมหึมาจนกระทั่งถึงทุกวันนี้ และเรียกเครือข่ายนี้ว่าอินเตอร์เนต(Internet) โดยเครือข่ายส่วนใหญ่จะอยู่ในประเทศสหรัฐอเมริกา และปัจจุบันนี้ได้มีเครือข่ายย่อยมากถึง 25,000 เครือข่ายเลยทีเดียว

2.3.2.2 โปรโตคอล TCP/IP

โปรโตคอล TCP/IP หรือ Transmission Control Protocol/Internet Protocol เป็นระเบียบวิธีการสื่อสารระหว่างคอมพิวเตอร์ที่ใช้กันมาแต่เดิมในระบบปฏิบัติการ Unix ซึ่งมีการใช้งานอย่างกว้างขวางมากจนถือเป็นมาตรฐานได้ จุดกำเนิดของโปรโตคอล TCP/IP นี้เริ่มขึ้นในราว พ.ศ. 2512 ที่กระทรวงกลาโหมของสหรัฐ เมื่อพบปัญหาในการเชื่อมโยงเครือข่ายคอมพิวเตอร์ในหน่วยงานต่าง ๆ ของตนซึ่งจะต้องมีการส่งข้อมูลระหว่างกัน และไปยังหน่วยงานภายนอกอื่น ๆ เช่น มหาวิทยาลัย ห้องทดลองต่าง ๆ (ส่วนใหญ่มีเครื่องที่ใช้ระบบ Unix อยู่เป็นจำนวนมาก) เนื่องจากแต่ละแห่งก็จะมีระบบคอมพิวเตอร์ของตนเองที่แตกต่างกันไป การต่อเชื่อมกันก็เป็นไปในลักษณะต่างคนต่างทำไม่เหมือนกัน ดังนั้นข่าวสารข้อมูลทั้งหลายจึงถ่ายเทไปมาได้อย่างยากลำบาก กระทรวงกลาโหมสหรัฐได้จัดตั้งหน่วยงาน Advanced Research Projects Agencies (ARPA) ขึ้นมาเพื่อหาทางแก้ไขปัญหาที่เกิดขึ้นนี้ ผลลัพธ์ที่หน่วยงาน ARPA ได้จัดทำขึ้นคือการกำหนดมาตรฐานในการสื่อสารข้อมูลและได้จัดตั้งเครือข่าย ARPANET ขึ้นโดยใช้โปรโตคอล TCP/IP ต่อมาก็กลายมาเป็นมาตรฐานจริงจัง ในราวปี พ.ศ. 2525 ความสัมพันธ์ระหว่าง TCP/IP กับระบบปฏิบัติการ Unix เกิดขึ้น เนื่องจากมหาวิทยาลัยแคลิฟอร์เนียที่เบอร์คเลย์ ได้พัฒนาระบบปฏิบัติการ Unix ซึ่งมีการผนวกเข้ากับโปรโตคอล TCP/IP สำหรับใช้ในการสื่อสารระหว่างระบบออกมาและเผยแพร่ต่อไปยังหน่วยงานต่าง ๆ ทำให้การสื่อสารกันของเครื่องที่ใช้ระบบปฏิบัติการ Unix มักจะต้องใช้โปรโตคอล TCP/IP เสมอ และมีบทบาทเป็นสิ่งที่คู่กันต่อมาถึงปัจจุบัน

ในปัจจุบันนี้ ไม่ว่าคอมพิวเตอร์ส่วนบุคคลของผู้ใช้จะเป็นแบบใดก็ตาม เช่น พีซี หรือแมคอินทอช ก็สามารถใช้งานโปรโตคอล TCP/IP เพื่อต่อเชื่อมเข้าสู่อินเตอร์เน็ตได้ วิธีการก็

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คือเพียงแต่ติดตั้งใช้งานซอฟต์แวร์โปรโตคอล TCP/IP เท่านั้น ส่วนวิธีการและโปรแกรมที่ติดตั้งจะแตกต่างกันขึ้นกับระบบที่ใช้ซึ่งจะกล่าวต่อไป หมายเลข IP (IP Address) การสื่อสารกันในระบบเครือข่ายอินเตอร์เน็ตที่มีโปรโตคอล TCP/IP เป็นมาตรฐานนี้ เครื่องคอมพิวเตอร์ทุกเครื่องที่เชื่อมต่ออยู่จะต้องมีหมายเลขประจำตัวเอาไว้อ้างอิงให้เครื่องคอมพิวเตอร์อื่น ๆ ได้ทราบเหมือนกับคนทุกคนต้องมีชื่อให้คนอื่นเรียก หมายเลขอ้างอิงดังกล่าวเราเรียกว่า IP Address หรือหมายเลข IP หรือบางทีก็เรียกว่า " แอดเดรส IP " (IP ในที่นี้ก็คือ Internet Protocol ตัวเดียวกับใน TCP/IP นั่นเอง) ซึ่งถูกจัดเป็นตัวเลขชุดหนึ่งขนาด 32 บิต ใน 1 ชุดนี้จะมีตัวเลขถูกแบ่งออกเป็น 4 ส่วน ส่วนละ 8 บิตเท่า ๆ กัน เวลาเขียนก็แปลงให้เป็นเลขฐานสิบ ก่อนเพื่อความง่ายแล้วเขียนโดยคั่นแต่ละส่วนด้วยจุด ดังนั้นในตัวเลขแต่ละส่วนนี้จึงมีค่าได้ตั้งแต่ 0 จนถึง 28-1 = 255 เท่านั้น เช่น 192.10.1.101 เป็นต้น ตัวเลข IP Address ชุดนี้จะเป็นสิ่งที่สำคัญคล้ายเบอร์โทรศัพท์ที่เรามีใช้อยู่และไม่ซ้ำกัน เพราะสามารถกำหนดเป็นตัวเลขได้รวมทั้งสิ้นกว่า 4 พันล้านเลขหมาย แต่การกำหนดให้คอมพิวเตอร์มีเลขหมาย IP Address นี้ไม่ได้เริ่มต้นจากหมายเลข 1 และนับขึ้นไปเรื่อย ๆ หากแต่จะมีการจัด แบ่งออกเป็น 2 ส่วน คือ

ส่วนแรกเป็นหมายเลขของเครือข่าย (Network Number)

ส่วนที่สองเรียกว่าหมายเลขของคอมพิวเตอร์ที่อยู่ในเครือข่ายนั้น (Host Number) เพราะในเครือข่ายใด ๆ อาจจะมีเครื่องคอมพิวเตอร์เชื่อมต่ออยู่ได้มากมาย ในเครือข่ายที่อยู่คนละระบบอาจมีหมายเลข Host ซ้ำกันก็ได้ แต่เมื่อรวมกับหมายเลข Network แล้ว จะได้เป็น IP Address ที่ไม่ซ้ำกันเลย

ในการจัดตั้งหรือกำหนดหมายเลข IP Address นี้ก็มีวิธีการกำหนดที่ชัดเจน และมีกฎเกณฑ์ที่รัดกุม ผู้ใช้ที่อยากจัดตั้งโฮสต์คอมพิวเตอร์ เพื่อเชื่อมต่อเข้ากับอินเตอร์เน็ตและให้บริการต่าง ๆ สามารถขอหมายเลข IP Address ได้ที่หน่วยงาน Internet Network Information Center (InterNIC) ขององค์กร Network Solution Incorporated (NSI) ที่รัฐเวอร์จิเนีย สหรัฐอเมริกา แต่ถ้าผู้ใช้สมัครเข้าเป็นสมาชิกขอใช้บริการอินเตอร์เน็ตจากบริษัทผู้ให้บริการ(Internet Service Provider) เรียกย่อ ๆ ว่าหน่วยงาน ISP รายใดก็แล้วแต่ ก็ไม่ต้องติดต่อขอ IP Address เนื่องจากหน่วยงาน ISP เหล่านั้นจะกำหนดหมายเลข IP ให้ใช้ หรือส่งค่า IP ชั่วคราวให้ใช้งาน ทั้งนี้ขึ้นอยู่กับแบบการขอใช้บริการที่จะกล่าวต่อไป

โครงสร้างของแอดเดรสที่ใช้ใน class ต่าง ๆ ของเครือข่าย ซึ่งทั้งหมดยาว 32 บิต IP Address นี้มีการจัดแบ่งออกเป็นทั้งหมด 5 ระดับ (Class) แต่ที่ใช้งานในทั่วไปจะมีเพียง 3 ระดับ คือ Class A, Class B, Class C ซึ่งก็แบ่งตามขนาดความใหญ่ของเครือข่ายนั่นเอง ถ้าเครือข่ายใดมีจำนวนเครื่องคอมพิวเตอร์เชื่อมต่ออยู่มากก็จะมีหมายเลขอยู่ใน Class A ถ้ามีจำนวนเครื่องต่ออยู่ลดหลั่นกันลงมาก็จะอยู่ใน Class B และ Class C ตามลำดับ หมายเลข IP ของ Class A มีตัวแรกเป็น 0 และหมายเลขของเครือข่าย (Network Number) ขนาด 7 บิต และมีหมายเลขของเครื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คอมพิวเตอร์ (Host Number) ขนาด 24 บิต ทำให้ในหนึ่งเครือข่ายของ Class A สามารถมีคอมพิวเตอร์เชื่อมต่ออยู่ในเครือข่ายได้ถึง 224 = 16 ล้านเครื่อง เหมาะสำหรับองค์กรหรือบริษัทยักษ์ใหญ่ แต่ใน Class A นี้จะมีหมายเลขเครือข่ายได้ 128 ตัวเท่านั้นทั่วโลก ซึ่งหมายความว่าจะมีเครือข่ายยักษ์ใหญ่แบบนี้ได้เพียง 128 เครือข่ายเท่านั้น สำหรับ Class B จะมีหมายเลขเครือข่ายแบบ 14 บิต และหมายเลขเครื่องคอมพิวเตอร์แบบ 16 บิต (ส่วนอีก 2 บิตที่เหลือบังคับว่าต้องขึ้นต้นด้วย 102) ดังนั้นจึงสามารถมีจำนวนเครือข่ายที่อยู่ใน Class B ได้มากกว่า Class A คือมีได้ถึง 214 = กว่า 16,000 เครือข่าย และก็สามารถมีเครื่องคอมพิวเตอร์เชื่อมต่อกันในเครือข่าย Class B แต่ละเครือข่ายได้ถึง 216 หรือมากกว่า 65,000 เครื่อง สุดท้ายคือ Class C ซึ่งมีหมายเลขเครื่องคอมพิวเตอร์แบบ 8 บิตและมีหมายเลขเครือข่ายแบบ 21 บิต ส่วนสามบิตแรกบังคับว่าต้องเป็น 1102 ดังนั้นใน แต่ละเครือข่าย Class C จะมีจำนวนเครื่องต่อเชื่อมได้เพียงไม่เกิน 254 เครื่องในแต่ละเครือข่าย (28 = 256 แต่หมายเลข 0 และ 255 จะไม่ถูกใช้งาน จึงเหลือเพียง 254) ดังนั้นวิธีการสังเกตได้ง่าย ๆ ว่าเราเชื่อมต่ออยู่ที่เครือข่าย Class ใดก็สามารถดูได้จาก IP Address ในส่วนหน้า (ส่วน Network Address) โดย

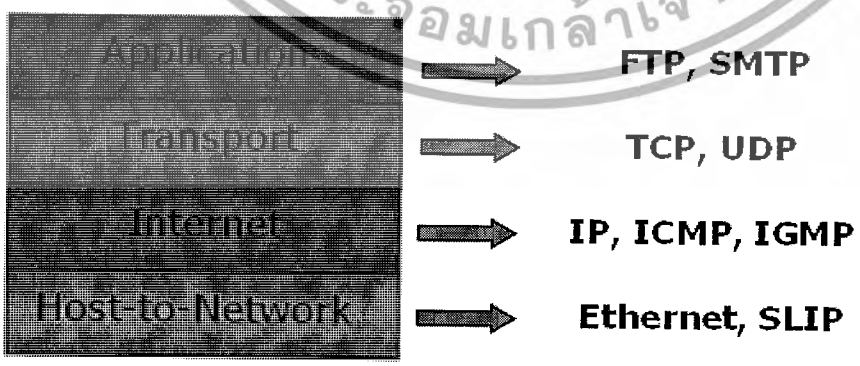
Class A จะมี Network address ตั้งแต่ 0 ถึง 127 (บิตแรกเป็น 0 เสมอ)

Class B จะมี Network address ตั้งแต่ 128 ถึง 191 (เพราะขึ้นต้นด้วย 102 เท่านั้น)

Class C จะมี Network address ตั้งแต่ 192 ถึง 223 (เพราะขึ้นต้นด้วย 1102 เท่านั้น)

เช่น ถ้าเครื่องคอมพิวเตอร์ในอินเตอร์เน็ตมีหมายเลข IP ดังนี้ 181.11.82.22 ตัวเลข 181.11 แสดงว่าเป็นเครือข่ายใน Class B ซึ่งหมายเลขเครือข่ายเต็มๆ จะใช้ 2 ส่วนแรกคือ 181.11 และมีหมายเลขคอมพิวเตอร์คือ 82.22 หรือถ้ามี IP Address เป็น 192.131.10.101 ทำให้ทราบว่าเครื่องคอมพิวเตอร์นั้นเชื่อมต่ออยู่ใน Class C มีหมายเลขเครือข่ายคือ 3 ส่วนแรก ได้แก่ 192.131.10 และหมายเลขประจำ เครื่องคือ 101 เป็นต้น

ในแต่ละเลเยอร์ของโครงสร้าง TCP/IP สามารถอธิบายได้ดังนี้



รูปที่ 2.12 โครงสร้าง TCP/IP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ชั้นโฮสต์-เครือข่าย(Host-to-NetworkLayer)

เป็นโพรโตคอลสำหรับการควบคุมการสื่อสาร ในชั้นนี้เป็นสิ่งที่ไม่มีการกำหนดรายละเอียดอย่างเป็นทางการ หน้าที่หลักคือการรับข้อมูลจากชั้นสื่อสาร IP มาแล้วส่งไปยังโหนดที่ระบุไว้ในเส้นทางเดินข้อมูลทางด้านผู้รับก็จะทำงานในทางกลับกัน คือ รับข้อมูลจากสายสื่อสารแล้วนำส่งให้กับโปรแกรมในชั้นสื่อสาร

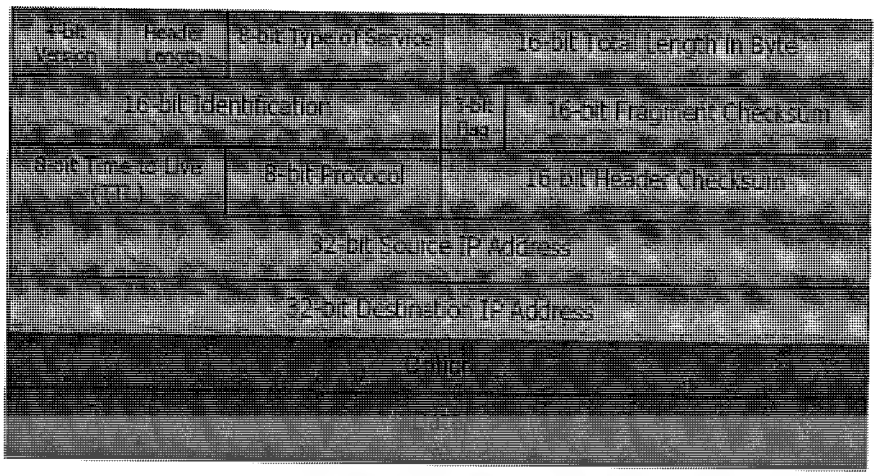
2. ชั้นสื่อสารอินเทอร์เน็ต (The Internet Layer)

ใช้ประเภทของระบบการสื่อสาร ที่เรียกว่า ระบบเครือข่ายแบบสลับช่องสื่อสารระดับแพ็กเก็ต (packet-switching network) ซึ่งเป็นการติดต่อแบบไม่ต่อเนื่อง (Connectionless) หลักการทำงานคือการปล่อยให้ข้อมูลขนาดเล็กที่เรียกว่า แพ็กเก็ต (Packet) สามารถไหลจากโหนดผู้ส่งไปตามโหนดต่าง ๆ ในระบบจนถึงจุดหมายปลายทางได้โดยอิสระ หากว่ามีการส่งแพ็กเก็ตออกมาเป็นชุดโดยมีจุดหมายปลายทางเดียวกันในระหว่างการเดินทางในเครือข่าย แพ็กเก็ตแต่ละตัวในชุดนี้ก็จะเป็นอิสระแก่กันและกัน ดังนั้น แพ็กเก็ตที่ส่งไปถึงปลายทางอาจจะไม่เป็นไปตามลำดับก็ได้

2.1 IP(Internet Protocol)

IP เป็นโปรโตคอลในระดับเน็ตเวิร์คเลเยอร์ ทำหน้าที่จัดการเกี่ยวกับแอดเดรสและข้อมูล และควบคุมการส่งข้อมูลบางอย่างที่ใช้ในการหาเส้นทางของแพ็กเก็ต ซึ่งกลไกในการหาเส้นทางของ IP จะมีความสามารถในการหาเส้นทางที่ดีที่สุด และสามารถเปลี่ยนแปลงเส้นทางได้ในระหว่างการส่งข้อมูล และมีระบบการแยกและประกอบดาต้าแกรม (datagram) เพื่อรองรับการส่งข้อมูลระดับ data link ที่มีขนาด MTU (Maximum Transmission Unit) ที่แตกต่างกัน ทำให้สามารถนำ IP ไปใช้บนโปรโตคอลอื่นได้หลากหลาย เช่น Ethernet , Token Ring หรือ Apple Talk การเชื่อมต่อของ IP เพื่อทำการส่งข้อมูล จะเป็นแบบ connectionless หรือเกิดเส้นทางการเชื่อมต่อในทุก ๆ ครั้งของการส่งข้อมูล 1 ดาต้าแกรม โดยจะไม่ทราบถึงข้อมูลดาต้าแกรมที่ส่งก่อนหน้าหรือส่งตามมา แต่การส่งข้อมูลใน 1 ดาต้าแกรม อาจจะเกิดการส่งได้หลายครั้งในกรณีที่มีการแบ่งข้อมูลออกเป็นส่วนย่อย ๆ (fragmentation) และถูกนำไปรวมเป็นดาต้าแกรมเดิมเมื่อถึงปลายทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 IP Header

เฮดเดอร์ของ IP โดยปกติจะมีขนาด 20 bytes ยกเว้นในกรณีที่มีการเพิ่ม option บางอย่าง ฟิลด์ของเฮดเดอร์ IP จะมีความหมายดังนี้

a. Version : เป็นหมายเลขเวอร์ชันของโปรโตคอล ที่ใช้งานในปัจจุบัน คือ เวอร์ชัน 4 (IPv4) และเวอร์ชัน 6 (IPv6)

b. Header Length : ความยาวของเฮดเดอร์ โดยทั่วไปถ้าไม่มีส่วน option จะมีค่าเป็น 5 (5*32 bit)

c. Type of Service (TOS) : ใช้เป็นข้อมูลสำหรับเราเตอร์ในการตัดสินใจเลือกการเราต์ข้อมูลในแต่ละดาต้าแกรม แต่ในปัจจุบันไม่ได้มีการนำไปใช้งานแล้ว

d. Length : ความยาวทั้งหมดจะเป็นจำนวนไบต์ของดาต้าแกรม ซึ่งด้วยขนาด 16 บิตของฟิลด์ จะหมายถึงความยาวสูงสุดของดาต้าแกรม คือ 65535 byte (64k) แต่ในการส่งข้อมูลจริง ข้อมูลจะถูกแยกเป็นส่วนๆตามขนาดของ MTU ที่กำหนดในลิงค์เลเยอร์และนำมารวมกันอีกครั้งเมื่อส่งถึงปลายทางแอพพลิเคชั่นส่วนใหญ่จะมีขนาดของดาต้าแกรมไม่เกิน 512 byte

e. Identification : เป็นหมายเลขของดาต้าแกรมในกรณีที่มีการแยกดาต้าแกรมเมื่อข้อมูลส่งถึงปลายทางจะนำข้อมูลที่มี identification เดียวกันมารวมกัน

f. Flag : ใช้ในกรณีที่มีการแยกดาต้าแกรม

g. Fragment offset : ใช้ในการกำหนดตำแหน่งข้อมูลในดาต้าแกรม ที่มีการแยกส่วน เพื่อให้สามารถนำกลับมาเรียงต่อกันได้อย่างถูกต้อง

h. Time to live(TTL) : กำหนดจำนวนครั้งที่มากที่สุด ที่ดาต้าแกรมจะถูกส่งระหว่าง hop (การส่งผ่านข้อมูลระหว่างเน็ตเวิร์ค) เพื่อป้องกันไม่ให้เกิดการส่งข้อมูลโดยไม่สิ้นสุด โดยเมื่อข้อมูลถูกส่งไป 1 hop จะทำการลดค่า TTL ลง 1 เมื่อค่าของ TTL เป็น 0 และข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ยังไม่ถึงปลายทาง ข้อมูลนั้นจะถูกยกเลิก และเราเตอร์สุดท้ายจะส่งข้อมูล ICMP แจ้งกลับมายังต้นทางว่าเกิด time out ในระหว่างการส่งข้อมูล

i. Protocol : ระบุโปรโตคอลที่ส่งในดาต้าแกรม เช่น TCP, UDP หรือ ICMP

j. Header checksum :ใช้ตรวจสอบความถูกต้องของข้อมูลในเฮดเดอร์

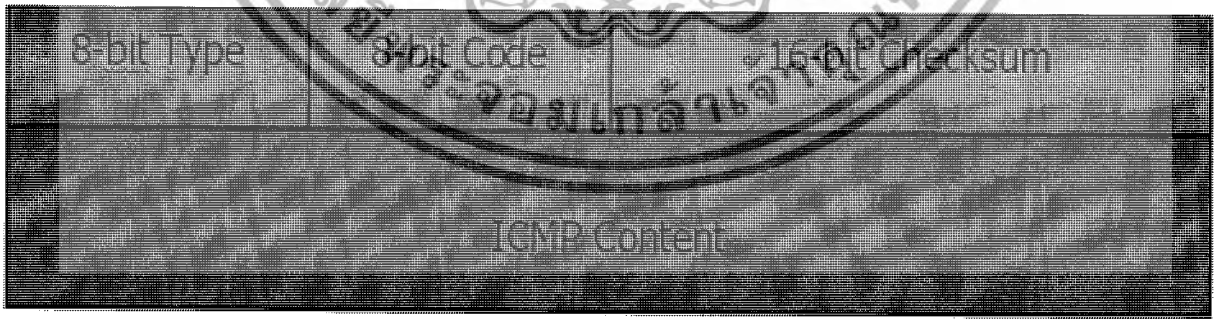
k. Source IP address : หมายเลข IP ของผู้ส่งข้อมูล

l. Destination IP address : หมายเลข IP ของผู้รับข้อมูล

m. Data : ข้อมูลจากโปรโตคอลระดับบน

2.2 ICMP (Internet Control Message Protocol)

ICMP เป็นโปรโตคอลที่ใช้ในการตรวจสอบและรายงานสถานภาพของดาต้าแกรม (Datagram) ในกรณีที่เกิดปัญหากับดาต้าแกรม เช่น เราเตอร์ไม่สามารถส่งดาต้าแกรมไปถึงปลายทางได้ ICMP จะถูกส่งออกไปยังโฮสต้นทางเพื่อรายงานข้อผิดพลาดที่เกิดขึ้น อย่างไรก็ดีไม่มีอะไรรับประกันได้ว่า ICMP Message ที่ส่งไปจะถึงผู้รับจริงหรือไม่ หากมีการส่งดาต้าแกรมออกไปแล้วไม่มี ICMP Message ฟ้อง Error กลับมาก็แปลความหมายได้สองกรณีคือ ข้อมูลถูกส่งไปถึงปลายทางอย่างเรียบร้อยหรืออาจจะมีปัญหา ในการสื่อสารทั้งการส่งดาต้าแกรมและ ICMP Message ที่ส่งกลับมาก็มีปัญหาระหว่างทางก็ได้ ICMP จึงเป็นโปรโตคอลที่ไม่มีความน่าเชื่อถือ (unreliable) ซึ่งจะเป็นหน้าที่ของโปรโตคอลในระดับสูงกว่า Network Layer ในการจัดการให้การสื่อสารนั้น ๆ มีความน่าเชื่อถือ ในส่วนของ ICMP Message จะประกอบด้วย Type ขนาด 8 บิต Checksum ขนาด 16 บิต และส่วนของ Content ซึ่งจะมีขนาดที่แตกต่างกันไปตาม Type และ Code ดังรูป



รูปที่ 2.14 ICMP Header

3. ชั้นสื่อสารนำส่งข้อมูล (Transport Layer)

แบ่งเป็นโพรโตคอล 2 ชนิดตามลักษณะ ลักษณะแรกเรียกว่า Transmission Control Protocol (TCP) เป็นแบบที่มีการกำหนดช่วงการสื่อสารตลอดระยะเวลาการสื่อสาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(connection - oriented) ซึ่งจะยอมให้มีการส่งข้อมูลเป็นแบบ Byte stream ที่ไว้ใจได้โดยไม่มีข้อผิดพลาด ข้อมูลที่มีปริมาณมากจะถูกแบ่งออกเป็นส่วนเล็ก ๆ เรียกว่า message ซึ่งจะถูกส่งไปยังผู้รับผ่านทางชั้นสื่อสารของอินเทอร์เน็ต ทางฝ่ายผู้รับจะนำ message มาเรียงต่อกันตามลำดับเป็นข้อมูลตัวเดิม TCP ยังมีความสามารถในการควบคุมการไหลของข้อมูลเพื่อป้องกันไม่ให้ผู้ส่งส่งข้อมูลเร็วเกินกว่าที่ผู้รับจะทำงานได้ทันอีกด้วย โปรโตคอลการนำส่งข้อมูลแบบที่สองเรียกว่า UDP (User Datagram Protocol) เป็นการติดต่อแบบไม่ต่อเนื่อง (connectionless) มีการตรวจสอบความถูกต้องของข้อมูลแต่จะไม่มีการแจ้งกลับไปยังผู้ส่ง จึงถือได้ว่าไม่มีการตรวจสอบความถูกต้องของข้อมูล อย่างไรก็ตามวิธีการนี้มีข้อดีในด้านความรวดเร็วในการส่งข้อมูล จึงนิยมใช้ในระบบผู้ให้และผู้ใช้บริการ (client/server system) ซึ่งมีการสื่อสารแบบ ถาม/ตอบ (request/reply) นอกจากนั้นยังใช้ในการส่งข้อมูลประเภทภาพเคลื่อนไหวหรือการส่งเสียง (voice) ทางอินเทอร์เน็ต

3.1 UDP : (User Datagram Protocol)

เป็นโปรโตคอลที่อยู่ใน Transport Layer เมื่อเทียบกับโมเดล OSI โดยการส่งข้อมูลของ UDP นั้นจะเป็นการส่งครั้งละ 1 ชุดข้อมูล เรียกว่า UDP datagram ซึ่งจะไม่มีความสัมพันธ์กันระหว่างดาต้าแกรม และจะไม่มีกลไกการตรวจสอบความสำเร็จในการรับส่งข้อมูล กลไกการตรวจสอบโดย checksum ของ UDP นั้นเพื่อเป็นการป้องกันข้อมูลที่อาจจะถูกแก้ไขหรือมีความผิดพลาดระหว่างการส่ง และหากเกิดเหตุการณ์ดังกล่าวปลายทางจะได้รู้ว่ามีข้อผิดพลาดเกิดขึ้น แต่มันจะเป็นการตรวจสอบเพียงฝ่ายเดียวเท่านั้น โดยในข้อกำหนดของ UDP หากพบว่า Checksum Error ก็ให้ผู้รับปลายทางทำการทิ้งข้อมูลนั้นแต่จะไม่มีการแจ้งกลับไปยังผู้ส่ง การรับส่งข้อมูลแต่ละครั้งหากเกิดข้อผิดพลาดในระดับ IP เช่น ส่งไม่ถึง, หมดเวลา ผู้ส่งจะได้รับ Error Message จากระดับ IP เป็น ICMP Error Message แต่เมื่อข้อมูลส่งถึงปลายทางถูกต้อง แต่เกิดข้อผิดพลาดในส่วนของ UDP เอง จะไม่มีการยืนยันหรือแจ้งให้ผู้ส่งทราบ

รูปที่ 2.15 UDP Header

มีรายละเอียด ดังนี้

- Source Port Number : หมายเลขพอร์ตต้นทางที่ส่งดาต้าแกรมนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

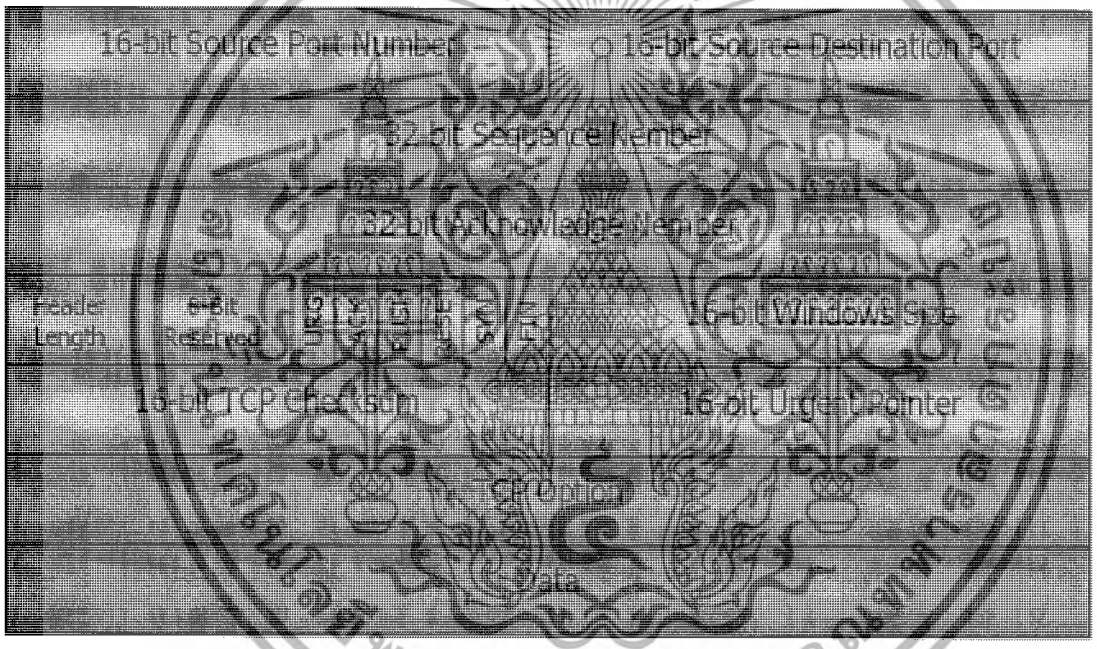
- Destination Port Number : หมายเลขพอร์ตปลายทางที่จะเป็นผู้รับดาต้าแกรม

- UDP Length : ความยาวของดาต้าแกรม ทั้งส่วน Header และ data นั่นหมายความว่าค่าที่น้อยที่สุดในฟิลด์นี้คือ 8 ซึ่งเป็นขนาดของ Header

- Checksum : เป็นตัวตรวจสอบความถูกต้องของ UDP datagram และจะนำข้อมูลบางส่วนใน IP Header มาคำนวณด้วย

3.2 TCP : (Transmission Control Protocol)

อยู่ในชั้นของ Transport Layer เช่นเดียวกับ UDP ทำหน้าที่จัดการและควบคุมการรับส่งข้อมูล ซึ่งมีความสามารถและรายละเอียดมากกว่า UDP โดยดาต้าแกรมของ TCP จะมีความสัมพันธ์ต่อเนื่องกัน และมีกลไกควบคุมการรับส่งข้อมูลให้มีความถูกต้อง (reliable) และมีการสื่อสารอย่างเป็นกระบวนการ (connection-oriented)



รูปที่ 2.16 TCP Header

มีรายละเอียด ดังนี้

- Source Port Number : หมายเลขพอร์ตต้นทางที่ส่งดาต้าแกรมนี้

- Destination Port Number : หมายเลขพอร์ตปลายทางที่จะเป็นผู้รับดาต้าแกรม

- Sequence Number : ฟิลด์ที่ระบุหมายเลขลำดับอ้างอิงในการสื่อสารข้อมูลแต่ละครั้ง เพื่อ

- Acknowledgment Number : ทำหน้าที่เช่นเดียวกับ Sequence Number แต่จะใช้ในการใช้แยกแยะว่าเป็นข้อมูลของชุดใด และนำมาจัดลำดับได้ถูกต้อง

- Header Length : โดยปกติความยาวของเฮดเดอร์ TCP จะมีความยาว 20 ไบต์ แต่อาจจะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มากกว่านั้น ถ้ามีข้อมูลในฟิลด์ option แต่ต้องไม่เกิน 60 ไบต์

- Flag : เป็นข้อมูลระดับบิตที่อยู่ในเฮดเดอร์ TCP โดยใช้เป็นตัวบอกคุณสมบัติของแพ็กเก็ต TCP ขณะนั้น ๆ และใช้เป็นตัวควบคุมจังหวะการรับส่งข้อมูล ซึ่ง Flag มีอยู่ทั้งหมด 6 บิต แบ่งได้ดังนี้

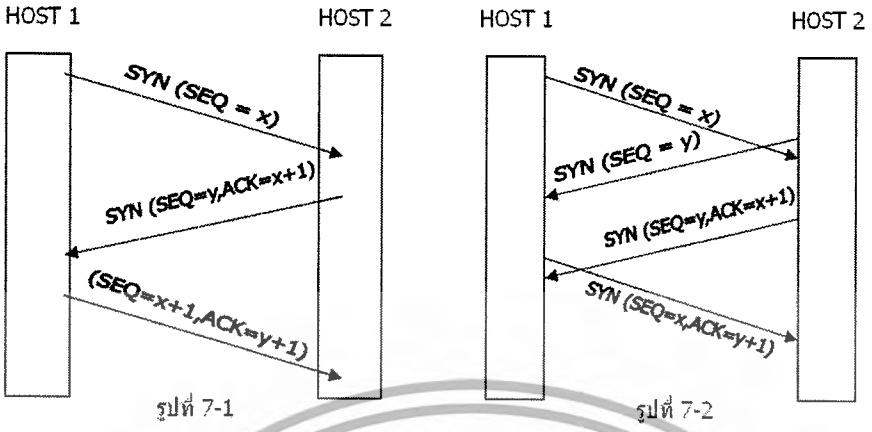
ตารางที่ 2.2 ตาราง Flag

Type	Description
URG	ใช้บอกความหมายว่าเป็นข้อมูลด่วน และมีข้อมูลพิเศษมาด้วย (อยู่ใน Urgent pointer)
ACK	แสดงว่าข้อมูลในฟิลด์ Acknowledge Number นำมาใช้งานได้
DSH	เป็นการแจ้งให้ผู้รับข้อมูลทราบว่าควรจะส่งข้อมูล Segment นี้ไปยัง Application ที่กำลังรออยู่โดยเร็ว
RST	ยกเลิกการติดต่อ (reset) เนื่องจากในกรณีที่เกิดการสับสนขึ้นด้วยเหตุผลต่างๆ เช่น โฮสต์มีปัญหา ให้เริ่มต้นสื่อสารกันใหม่
SYN	ใช้ในการเริ่มต้นขอติดต่อกับปลายทาง
FIN	ใช้ส่งเพื่อแจ้งให้ปลายทางทราบว่ายุติการติดต่อ

Flag ในเฮดเดอร์ของ TCP มีความสำคัญในการกำหนดการทำงานของ TCP segment เนื่องจากข้อมูลในเฮดเดอร์ของ TCP จะมีข้อมูลครบถ้วนทั้งการรับและการส่งข้อมูล ซึ่งในการทำงานแต่ละอย่างจะมีการใช้งานฟิลด์ไม่เหมือนกัน flag จะเป็นตัวกำหนดว่าให้ใช้งานฟิลด์ไหน เช่น ฟิลด์ Acknowledgment number จะไม่ถูกใช้ในขั้นตอนการเริ่มต้นการเชื่อมต่อ แต่จะมีข้อมูลในฟิลด์ ซึ่งเป็นข้อมูลที่ไม่มีความหมายใด ๆ ซึ่งถ้าไม่มี flag เป็นตัวกำหนดก็อาจจะมีการนำข้อมูลมาใช้ และก่อให้เกิดความผิดพลาดได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1 การสื่อสารของ TCP



รูปที่ 2.17 การสื่อสารของ TCP

เมื่อเซกเมนต์ CONNECT (SYN = "1" และ ACK = "0") เดินทางมาถึง Entity TCP ที่โฮสต์ปลายทางจะค้นหาโพรเซสตามหมายเลขพอร์ตที่กำหนดในเขตข้อมูล Destination port ซึ่งถ้าหากไม่พบก็จะตอบปฏิเสธด้วยเซกเมนต์ที่มี RST = "1" กลับไปยังผู้ส่ง เซกเมนต์ CONNECT ของผู้ส่งจะถูกส่งต่อไปยังโพรเซส ตามพอร์ตที่ระบุซึ่งอาจจะตอบรับหรือตอบปฏิเสธก็ได้ ถ้าโพรเซสนั้นต้องการสื่อสารด้วยก็จะส่งเซกเมนต์ตอบรับกลับไป รูปที่ 2.17 แสดงลำดับขั้นตอนการส่ง TCP เซกเมนต์ในการสร้างการเชื่อมต่อในสภาวะปกติระหว่างผู้ส่งและผู้รับ ในกรณีที่โฮสต์สองแห่ง พยายามสร้างการเชื่อมต่อระหว่างซ็อคเก็ตคู่เดียวกันจะเกิดเป็นลำดับขั้นตอนแสดงใน รูปที่ 2.17 ผลสุดท้ายจะมีการเชื่อมต่อเกิดขึ้นเพียงหนึ่งช่องทางเท่านั้นเนื่องจากการเชื่อมต่อในแต่ละช่องทางจะถูกกำหนดขึ้นโดยใช้หมายเลขซ็อคเก็ตผู้ส่งและผู้รับ ถ้าการเชื่อมต่อลำดับแรกสำเร็จก็จะถูกบันทึกไว้ในตารางการสื่อสาร เช่น (x, y) ถ้าการเชื่อมต่อลำดับที่สองสำเร็จในเวลาต่อมา ข้อมูลนี้ก็จะถูกบันทึกไว้ที่เดียวกันคือ (x, y)

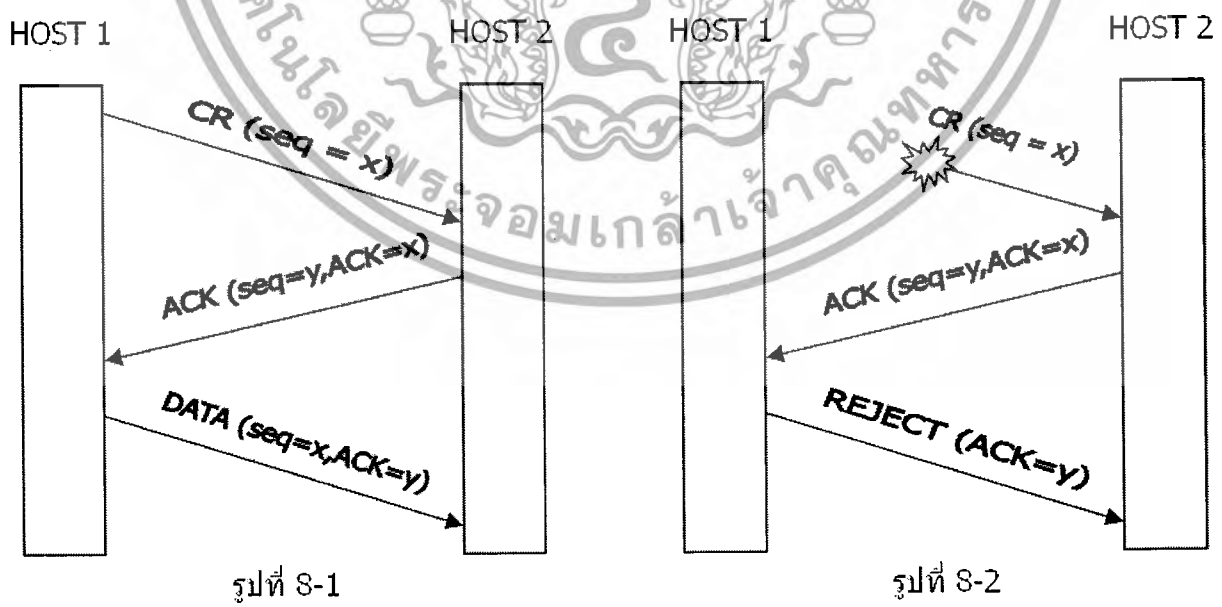
ขั้นตอนในการสร้างการเชื่อมต่อและการยกเลิก สามารถเขียนอธิบายด้วยไฟไนท์สเตทแมชชีนที่มีการทำงาน 11 สถานะ ในแต่ละสถานะจะมีเหตุการณ์บางอย่างที่เป็นไปได้ซึ่งจะได้รับการตอบสนองด้วยการกระทำที่เหมาะสม ในทางตรงกันข้าม เหตุการณ์ที่เป็นไปไม่ได้จะกลายเป็นข้อผิดพลาดที่จะต้องรายงานให้ทราบ

การเชื่อมต่อเริ่มต้นจากสถานะ CLOSED เมื่อเรียกใช้บริการ LISTEN หรือ CONNECT ก็จะมีการเปลี่ยนสถานะไปจากเดิม และถ้าอีกฝ่ายต้องการเชื่อมต่อด้วย การเชื่อมต่อก็จะเกิดขึ้นและย้ายไปอยู่ในสถานะ ESTABLISHED คือการเชื่อมต่อสมบูรณ์ และเมื่อยกเลิกการติดต่อก็จะกลับไปสู่สถานะ CLOSED อย่างเดิม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเริ่มต้นการสื่อสารของ TCP โดยใช้การบันทึกเวลาแบบ Three-way handshake Three-way Handshake เป็นวิธีการส่งแพ็กเก็ตที่สามารถช่วยแก้ปัญหาในเรื่องแพ็กเก็ตซ้ำซ้อนได้ดี แต่วิธีนี้จำเป็นจะต้องสร้างช่องสื่อสารให้ได้ก่อนที่จะเริ่มรับ-ส่งข้อมูล อย่างไรก็ตามแพ็กเก็ตควบคุมที่ใช้ในการต่อรองค่าตัวแปรสำหรับการสื่อสารต่าง ๆ อาจเกิดการตกค้างอยู่ในระบบได้ ทำให้การกำหนดค่าหมายเลขลำดับมีปัญหาไปด้วย เช่น การสร้างช่องสื่อสารระหว่างโฮสต์ 1 และ โฮสต์ 2 เริ่มจาก โฮสต์ 1 ขอเริ่มการเชื่อมต่อด้วยการส่งแพ็กเก็ต CR (Connection Request) ไปยังโฮสต์ 2 ซึ่งจะมีค่าตัวแปรต่าง ๆ สำหรับการสื่อสารรวมทั้งหมายเลขลำดับและหมายเลขช่องสื่อสารไปด้วย ผู้รับคือโฮสต์ 2 ก็จะส่ง ACK (Acknowledge) กลับมายังโฮสต์ 1 แต่ถ้าแพ็กเก็ตจากผู้ส่งเกิดสูญหายระหว่างทาง และสำเนาแพ็กเก็ตที่ยังตกค้างอยู่ระบบเกิดเดินทางไปถึงผู้รับในภายหลังจะทำให้การสร้างช่องสื่อสารใช้การไม่ได้เนื่องจากมีค่าตัวแปรต่าง ๆ ไม่ตรงกัน

การใช้ Three-way handshake เป็นการไม่บังคับให้ผู้ส่งและผู้รับข้อมูลจะต้องกำหนดค่าเริ่มต้นของหมายเลขลำดับเป็นเลขเดียวกัน ทำให้สามารถนำวิธีนี้มาใช้ร่วมกับวิธีการจัดจังหวะการทำงานให้พร้อมกัน (Synchronization) แบบต่าง ๆ ได้ แทนที่จะเป็นการใช้วิธีการบันทึกเวลา ดังรูปที่ 2.18 แสดงขั้นตอนการเริ่มต้นการทำงานจากโฮสต์ 1 ไปยังโฮสต์ 2 สมมุติให้โฮสต์ 1 เลือกหมายเลขลำดับเป็น “x” และส่งแพ็กเก็ต CONNECTION REQUEST ไปยังโฮสต์ 2 โฮสต์ 2 ตอบรับด้วยแพ็กเก็ต CONNECTION ACCEPTED ซึ่งจะยอมรับหมายเลขลำดับ “x” พร้อมกับประกาศหมายเลขลำดับ “y” ที่เป็นของตนเอง จากนั้นโฮสต์ 1 ก็จะตอบรับค่าตัวเลือกของโฮสต์ 2 ผ่านทางเขตข้อมูลสำหรับการควบคุมในแพ็กเก็ตข้อมูลแรกที่ส่งมา



รูปที่ 2.18 การสูญหายของแพคเก็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สมมติว่า ได้เกิดปัญหาการสูญหายของแพ็กเก็ตในขณะที่สำเนาแพ็กเก็ตที่ค้างในระบบเดินทางไปถึงผู้รับแทน รูปที่ 2.18 แสดงเหตุการณ์ที่แพ็กเก็ต TPDU (ตัวแรกในรูป) เป็นสำเนาแพ็กเก็ตเก่าที่พึ่งจะเดินทางไปถึงโฮสต์ 2 โดยที่โฮสต์ 1 ไม่ทราบ โฮสต์ 2 ก็จะทำงานปกติ คือจะตอบรับด้วยการส่งแพ็กเก็ต CONNECTION ACCEPTED TPDU กลับมาที่โฮสต์ 1ซึ่งโฮสต์ 1 จะสามารถตรวจสอบได้ว่าหมายเลขลำดับโฮสต์ 2 ที่ได้ตอบกลับมานั้นเป็นหมายเลขลำดับที่ได้เลิกใช้ไปแล้ว จึงมีการส่งแพ็กเก็ต REJECT กลับมายังโฮสต์ 2 เพื่อบอกยกเลิกการทำงาน จะเห็นว่าวิธีการนี้อาศัยการสื่อสารผ่านแพ็กเก็ต 3 ตัว ซึ่งเป็นที่มาของคำว่า " การจับมือร่วมสามขั้นตอน " ผลสุดท้ายทั้งโฮสต์ 1 และโฮสต์ 2 ก็จะไม่มีการสร้างช่องสื่อสารขึ้นมาจากข้อมูลในสำเนาแพ็กเก็ตเก่าแต่อย่างใด

4. ชั้นสื่อสารการประยุกต์ (Application Layer)

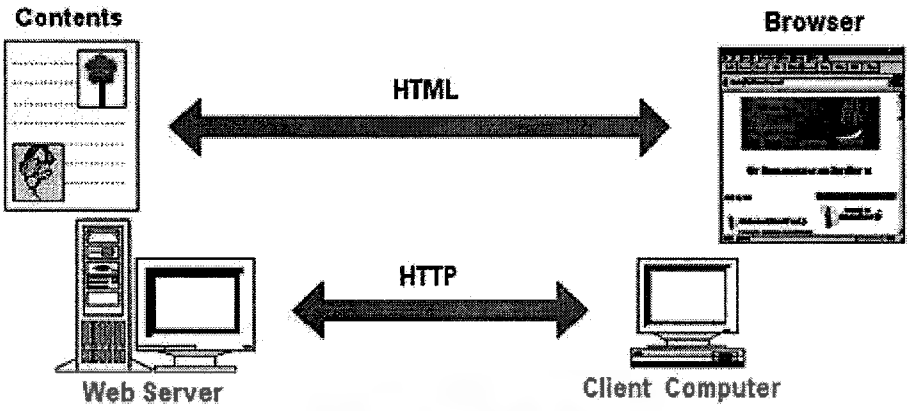
มีโพรโตคอล สำหรับสร้างจอเทอร์มินัลเสมือน เรียกว่า TELNET โพรโตคอลสำหรับการจัดการแฟ้มข้อมูล เรียกว่า FTP และโพรโตคอลสำหรับการให้บริการจดหมายอิเล็กทรอนิกส์ เรียกว่า SMTP โดยโพรโตคอลสำหรับสร้างจอเทอร์มินัลเสมือนช่วยให้ผู้ใช้สามารถติดต่อกับเครื่องโฮสต์ที่อยู่ไกลออกไปโดยผ่านอินเทอร์เน็ต และสามารถทำงานได้เสมือนกับว่ากำลังนั่งทำงานอยู่ที่เครื่องโฮสต์นั้น โพรโตคอลสำหรับการจัดการแฟ้มข้อมูลช่วยในการคัดลอกแฟ้มข้อมูลมาจากเครื่องอื่นที่อยู่ในระบบเครือข่าย หรือส่งสำเนาแฟ้มข้อมูลไปยังเครื่องใดก็ได้ โพรโตคอลสำหรับให้บริการจดหมายอิเล็กทรอนิกส์ช่วยในการจัดส่งข้อความไปยังผู้ใช้ในระบบหรือรับข้อความที่มีผู้ส่งเข้ามา

2.3.3 การติดต่อ server/Web browser

HTTP คือโปรโตคอลที่ใช้สื่อสารระหว่าง client computer กับ server computer ทำให้ทั้งสองเครื่องรู้ว่าจะจัดการส่งข้อมูลไปอย่างไร โปรโตคอล HTTP นี้วิ่งอยู่บน TCP/IP อีกชั้นหนึ่ง รูปแบบการทำงานจะไม่มีการจองสาย โดย client จะเรียกข้อมูลจาก server โดยการส่ง request ไปแล้วจะตัดการติดต่อทันที จากนั้นจะรอจนกระทั่ง server ส่งข้อมูลมาให้ ประโยชน์ของการทำงานแบบไม่จองสายของ HTTP ทำให้ WWW server สามารถให้บริการ client ได้หลาย ๆ คนพร้อม ๆ กัน การสื่อสารของ WWW จึงมีประสิทธิภาพมากขึ้น

ว่าไปแล้ว HTTP กับ HTML นั้นก็เหมือนกาแฟกับคอฟฟี่เมท โดย HTTP คือโปรโตคอลที่ใช้สื่อสารระหว่าง client computer กับ server computer ทำให้ทั้งสองเครื่องรู้ว่าจะจัดการส่งข้อมูลไปอย่างไร ส่วน HTML คือสื่อภาษาที่ทำให้เอกสารหรือ contents ที่อยู่บนเครื่อง server computer เมื่อถูกส่งมาที่ client computer แล้วจะนำไปแสดงได้อย่างไร เราเรียกซอฟท์แวร์ที่ใช้แสดงนี้ว่า Browser

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



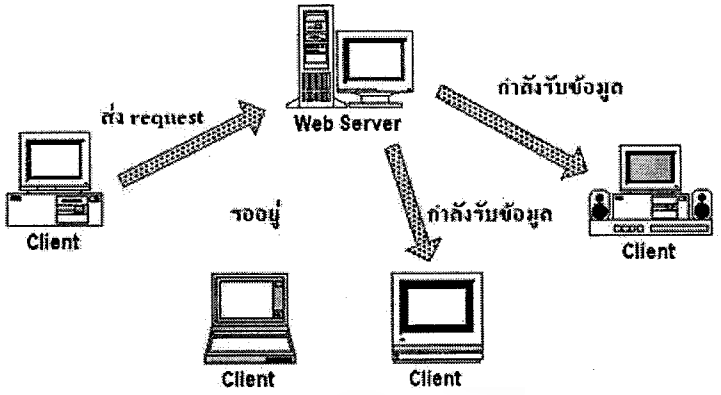
รูปที่ 2.19 แสดงความแตกต่างระหว่าง HTML และ HTTP

โปรโตคอล HTTP นี้วิ่งอยู่บน TCP/IP อีกชั้นหนึ่ง รูปแบบการทำงานจะไม่มีการจองสาย โดย client จะเรียกข้อมูลจาก server โดยการส่ง request ไปแล้วจะตัดการติดต่อทันที จากนั้นจะรอจนกระทั่ง server ส่งข้อมูลมาให้

รูปที่ 2.20 ลักษณะของการส่งของข้อมูล

ประโยชน์ของการทำงานแบบไม่จองสายของ HTTP ทำให้ WWW server สามารถให้บริการ client ได้หลาย ๆ คนพร้อม ๆ กัน การสื่อสารของ WWW จึงมีประสิทธิภาพมากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.21 แสดงการทำงานแบบไม่จองสายของ HTTP

2.4 เว็บแอพพลิเคชัน (Web Application)

เว็บแอพพลิเคชัน คือ ระบบงานที่สร้างขึ้นมาในลักษณะเว็บเพจ (Web Page) แล้วนำไปเก็บไว้ที่เว็บเซิร์ฟเวอร์ โดยสามารถเรียกใช้งานผ่านโปรแกรมบราวเซอร์ (Browser) ซึ่งการพัฒนาเว็บแอพพลิเคชันจะต้องอาศัยเทคโนโลยีด้านระบบเครือข่ายอินเทอร์เน็ต (Internet) และไคลเอนต์เซิร์ฟเวอร์ (Client/Server)

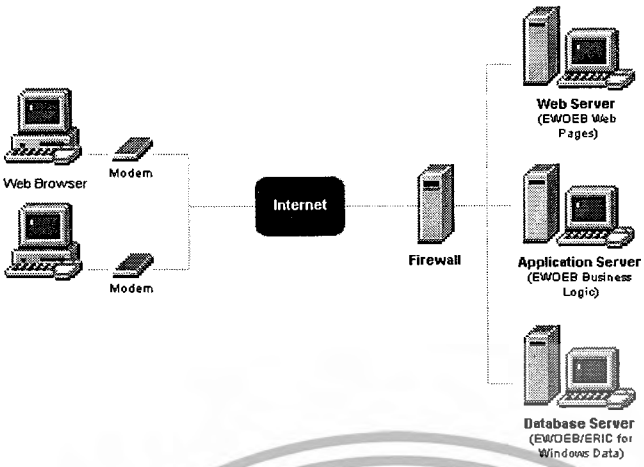
2.4.1 สถาปัตยกรรมไคลเอนต์/เซิร์ฟเวอร์ (Client/Server Architecture)

เครือข่ายคอมพิวเตอร์มีความสำคัญในการเชื่อมโยงภายในองค์กร การทำงานขององค์กรเดิมเน้นการทำงานแบบ Host-based System คือ ในหน่วยงานจะมีเครื่องคอมพิวเตอร์ขนาดใหญ่ซึ่งใช้เป็นทั้งแอพพลิเคชันเซิร์ฟเวอร์และดาต้าเบสหรือไฟล์เซิร์ฟเวอร์ และมีเครื่องคอมพิวเตอร์ขนาดเล็กที่ทำหน้าที่เป็นเทอร์มินอล (Terminal) เชื่อมต่อเข้ากับเครื่องเซิร์ฟเวอร์ โดยในการทำงานเครื่องเทอร์มินอลจะส่งข้อมูลไปประมวลผลที่เครื่องเซิร์ฟเวอร์ แล้วจึงส่งผลลัพธ์มาแสดงที่เครื่องเทอร์มินอล

แนวความคิดแบบ Host-based System นี้ ทำให้ไม่จำเป็นต้องใช้เครื่องเทอร์มินอลที่มีประสิทธิภาพสูงมาก และการเปลี่ยนแปลงแก้ไขแอพพลิเคชันจะสามารถทำได้ง่าย แต่ก็ยังมีปัญหาคือ เมื่อระบบมีขนาดใหญ่มากขึ้น มีปริมาณเทอร์มินอลที่เชื่อมต่อมากขึ้น เครื่องเซิร์ฟเวอร์จะต้องรับภาระหนักในการประมวลผลจนอาจส่งผลต่อเวลาที่ใช้ในการตอบสนองต่อผู้ใช้งาน เพื่อแก้ปัญหาข้างต้นนี้จึงได้เกิดแนวความคิดแบบไคลเอนต์เซิร์ฟเวอร์

แนวความคิดแบบ Client/Server นั้นจะมีเครื่องเซิร์ฟเวอร์ซึ่งทำหน้าที่เป็นเพียงดาต้าเบสเซิร์ฟเวอร์ (Database Server) หรือ ไฟล์เซิร์ฟเวอร์ (File Server) โดยการประมวลผลทั้งหมดจะอยู่ที่เครื่องเทอร์มินอล (Terminal) หรือไคลเอนต์ (Client) และเนื่องจากเครื่องไคลเอนต์ที่ใช้งานในปัจจุบันนั้นมีประสิทธิภาพมากขึ้น ทำให้สามารถสนับสนุนการทำงานในลักษณะนี้ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.22 แสดงการทำงานบนระบบไคลเอนต์/เซิร์ฟเวอร์

เซิร์ฟเวอร์ คือ เครื่องคอมพิวเตอร์บริการที่ให้บริการเมื่อมีผู้ร้องขอ และให้บริการแก่ผู้ใช้หลาย ๆ คน

ไคลเอนต์ คือ เครื่องคอมพิวเตอร์ที่ร้องขอบริการจากเซิร์ฟเวอร์

2.4.1.1 ประเภทของไคลเอนต์เซิร์ฟเวอร์ ตามลำดับชั้นในการทำงาน

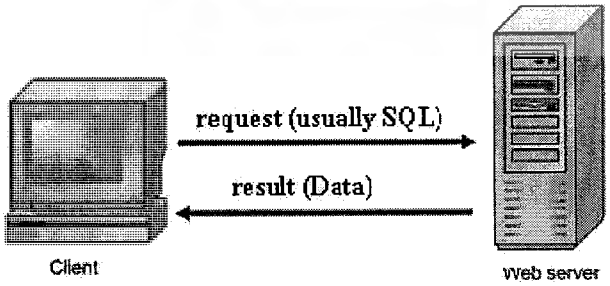
สามารถแบ่งได้เป็น

1) 1 เทียร์ (Tier)

เป็นการรวมการทำงานทุกอย่างภายในเครื่องเดียว เป็นทั้งไคลเอนต์และเซิร์ฟเวอร์ มีลักษณะการทำงานคล้ายกับคอมพิวเตอร์แบบ Stand Alone

2) 2 เทียร์ (Tier)

เป็นการแบ่งการทำงานเป็น 2 ส่วน คือ ส่วนไคลเอนต์ และส่วนเซิร์ฟเวอร์ โดยที่ฝั่งไคลเอนต์จะมีโปรแกรมเกี่ยวกับส่วนติดต่อผู้ใช้ (User Interface) ที่สามารถประมวลผลเบื้องต้นได้ ก่อนที่จะส่งคำสั่งไปขอข้อมูลจากฝั่งเซิร์ฟเวอร์ ซึ่งจะเป็นการลดภาระงานสำหรับเซิร์ฟเวอร์

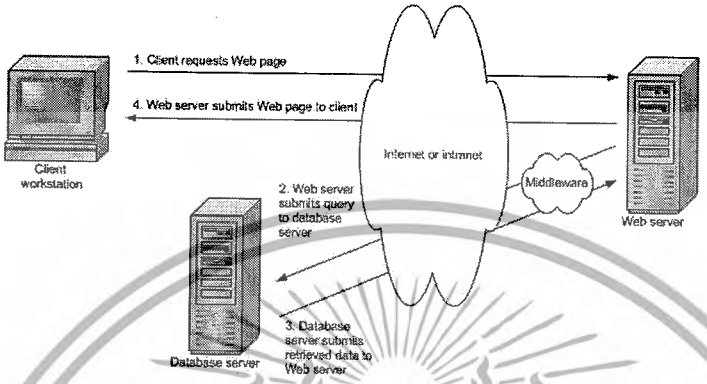


รูปที่ 2.23 แสดงสถาปัตยกรรมแบบ 2 Tier

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) 3 เทียร์ (Tier)

เป็นการแบ่งการทำงานเป็น 2 ส่วน คือ ส่วนไคลเอนต์และส่วนเซิร์ฟเวอร์ คล้ายกับ 2 เทียร์ แต่ว่าจะกระจายการทำงานของเซิร์ฟเวอร์ออกเป็น 2 เทียร์ โดยที่เทียร์หนึ่งทำงานเป็นดาต้าเบสเซิร์ฟเวอร์ และอีกเทียร์หนึ่งทำงานเป็นแอพพลิเคชันเซิร์ฟเวอร์



รูปที่ 2.24 แสดงสถาปัตยกรรมแบบ 3 Tier

2.4.1.2 ประเภทของไคลเอนต์เซิร์ฟเวอร์ ตามประเภทการใช้งาน

สามารถแบ่งได้เป็น

1) ไฟล์เซิร์ฟเวอร์ (File Server)

เป็นเซิร์ฟเวอร์ที่ทำหน้าที่จัดเก็บไฟล์ที่สร้างขึ้น เพื่อให้ไคลเอนต์อื่น ๆ สามารถเข้ามาใช้ไฟล์ร่วมกันได้โดยอาจจะกำหนดสิทธิ์ในการเข้าใช้ตามกลุ่มของผู้ใช้

2) ดาต้าเบสเซิร์ฟเวอร์ (Database Server)

เป็นเซิร์ฟเวอร์ที่ทำหน้าที่จัดการฐานข้อมูล (Database Management System : DBMS) คอยจัดการกับข้อมูลที่เก็บอยู่ในเซิร์ฟเวอร์เดียวกันนี้ ทำให้ผู้ใช้ที่เป็นไคลเอนต์สามารถเข้ามาใช้ข้อมูลร่วมกันได้ และข้อมูลที่ได้มีความทันสมัย และไม่ซ้ำซ้อน

3) แอพพลิเคชันเซิร์ฟเวอร์ (Application Server)

เป็นเซิร์ฟเวอร์ที่ทำหน้าที่จัดการกับโปรแกรมแอพพลิเคชัน ให้เชื่อมต่อระหว่างดาต้าเบสเวิร์ฟเวอร์กับผู้ใช้เซิร์ฟเวอร์ที่เป็นไคลเอนต์ เพื่อให้ทั้งสองส่วนนี้สามารถเชื่อมต่อกันได้อย่างสมบูรณ์

2.4.2 เว็บบราวเซอร์ (Web Browser)

เว็บบราวเซอร์ เป็นโปรแกรมแอพพลิเคชันชนิดหนึ่งที่ให้เราติดต่อหรือเปิดดูเว็บเพจ (WebPage) ที่เก็บอยู่ในบนเว็บไซต์ (Web Site) ใด ๆ และเป็นโปรแกรมบนเครื่องไคลเอนต์ที่ใช้

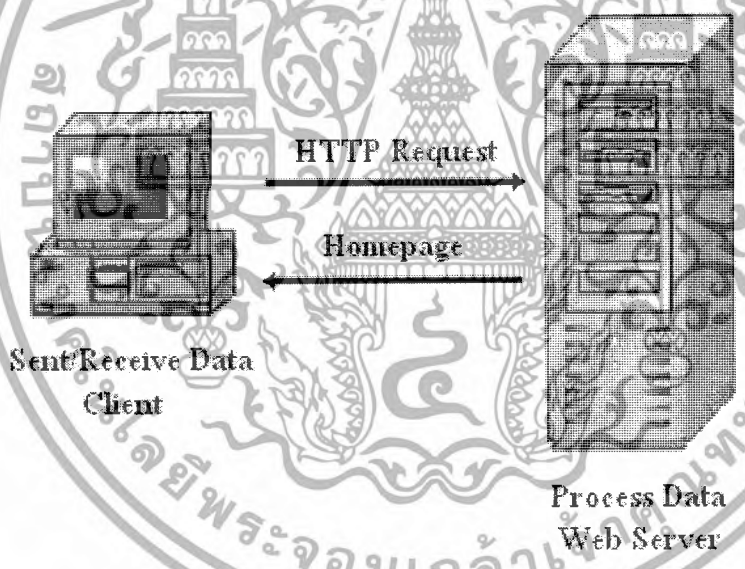
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Hypertext Transfer Protocol (HTTP) ซึ่งเป็นโปรโตคอลที่ใช้ในการร้องขอข้อมูลจากเว็บเซิร์ฟเวอร์ ข้อมูลที่เราสามารถจะเข้าไปดูได้มีทั้งข้อมูลที่อยู่ในรูปข้อความ กราฟฟิก เสียง เป็นต้น ตัวอย่างโปรแกรมเว็บบราวเซอร์ที่นิยมใช้กันทั่วไป เช่น Internet Explorer (IE), Netscape Navigator, Opera, Mozilla Firefox เป็นต้น

นอกจากเว็บบราวเซอร์จะใช้เพื่อดูเว็บเพจจากเว็บไซต์ใด ๆ ได้แล้วยังมีความสามารถอื่น ๆ เช่น บริการส่งจดหมายอิเล็กทรอนิกส์(E-Mail) การค้นหาข้อมูล การโอนถ่ายข้อมูลด้วย FTP เป็นต้น

2.4.3 Common Gateway Interface (CGI)

CGI หรือ Common Gateway Interface สามารถรับเอาข้อมูลจาก Clientไปประมวลผลที่ Server และส่งผลลัพธ์กลับไปให้ Client เช่น Client ต้องการสอบถามข้อมูลว่ามี Website ใดที่มีข้อมูลที่เกี่ยวข้องกับ PHP บ้าง CGI จะรับข้อมูลไปประมวลผลที่ Server และ Server ก็จะตรวจสอบกับ Database ที่มีอยู่ว่ามี Website ใดที่มีข้อมูลดังกล่าวหรือไม่ เสร็จแล้วก็จะส่งกลับไปให้ที่ Client



รูปที่ 2.25 แสดงการทำงานของ CGI

CGI จะใช้ภาษาใดต้องคำนึงถึง Platform ของ Web Server ด้วยดังแสดงในตารางที่ 2.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.3 แสดง CGI ที่ใช้บน Platform ต่าง ๆ

Unix/Linux	Windows NT
C	Perl
C++	PHP
Perl	ASP
PHP	CFM

2.5 Application Server

2.5.1 ระบบฐานข้อมูล (Database System)

2.5.1.1 ความหมาย

ระบบฐานข้อมูล คือ แหล่งเก็บข้อมูลและความสัมพันธ์ระหว่างข้อมูลนั้น โดยถูกออกแบบและควบคุมเป็นพิเศษ ในการจัดเก็บข้อมูลไม่ให้มีความซ้ำซ้อนหรือมีความซ้ำซ้อนน้อยที่สุดเพื่อให้ข้อมูลมีความถูกต้องสูงสุดและสามารถเรียกดูข้อมูลได้อย่างสะดวกและมีประสิทธิภาพ

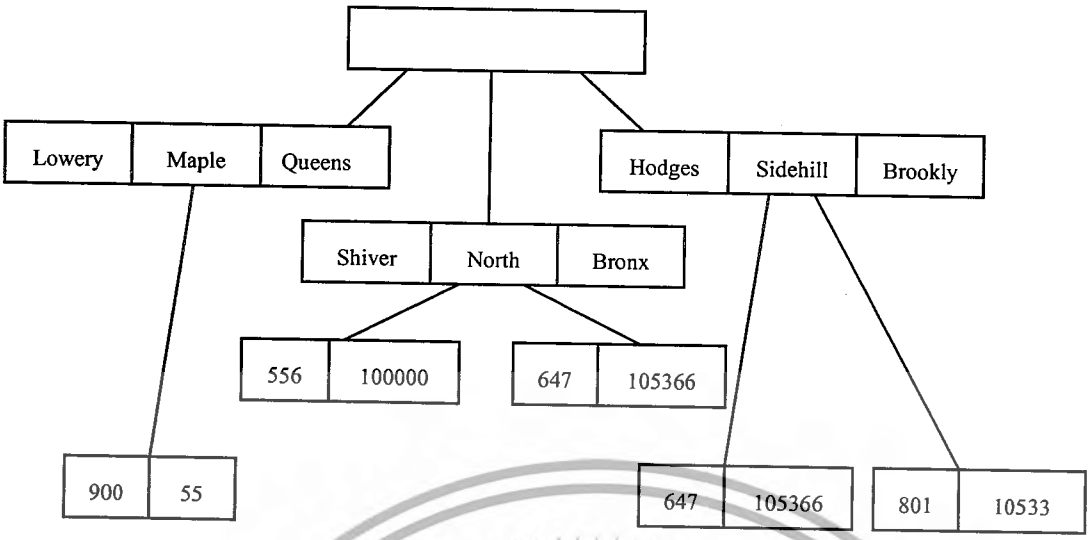
2.5.1.2 ความสัมพันธ์ระหว่างข้อมูลในฐานข้อมูล

ฐานข้อมูลจะอ้างอิงการจัดเก็บข้อมูลเป็นโครงสร้างหรือโมเดลต่าง ๆ ซึ่งทำให้การใช้งานต่างกัน โดยทั่วไปโครงสร้างของฐานข้อมูลที่นิยมใช้อย่างแพร่หลายในปัจจุบันจะมี 3 แบบดังนี้

1. โครงสร้างแบบลำดับชั้นหรือแบบแผนภูมิต้นไม้ (Hierarchical Model)

โครงสร้างนี้จะมีลักษณะใกล้เคียงกับแบบเครือข่าย แต่จะแตกต่างกันที่โครงสร้างความสัมพันธ์ของเรคคอร์ด (record) ที่อยู่ในฐานข้อมูล รูปแบบความสัมพันธ์ของเรคคอร์ดที่อยู่ในฐานข้อมูลเป็นแบบหนึ่งต่อกลุ่ม (one to many) ซึ่งจะมีลักษณะคล้ายต้นไม้กลับหัว และการค้นหาข้อมูลจะเริ่มจากตัวแม่ (root) และลำดับความสัมพันธ์ลงมาตามตัวลูก (child) แต่ละระดับ ตัวอย่างความสัมพันธ์ระหว่างข้อมูลแบบแผนภูมิต้นไม้ แสดงได้ดังรูปที่ 2.26

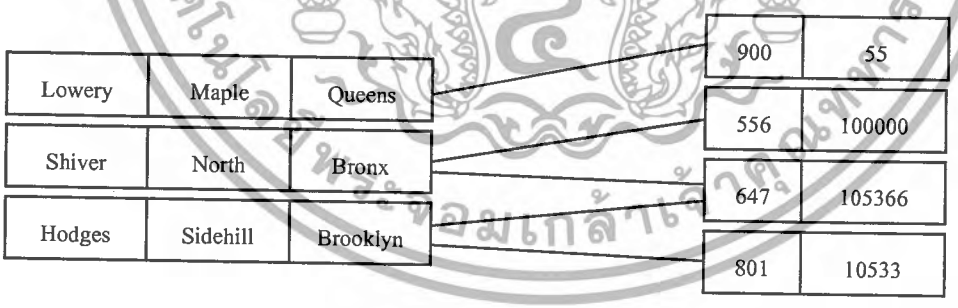
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.26 ตัวอย่างข้อมูลในฐานข้อมูลแบบแผนภูมิต้นไม้

2. โครงสร้างแบบเครือข่าย (Network Model)

การจัดข้อมูลในรูปแบบเครือข่ายนี้จะแสดงผลด้วยกลุ่มเรคคอร์ด ซึ่งจะมีส่วนเชื่อมต่อ (link) หรือตัวชี้ (pointer) แสดงความสัมพันธ์ระหว่างข้อมูล โดยโครงสร้างความสัมพันธ์ของเรคคอร์ดที่อยู่ในฐานข้อมูลนั้นจะมีสภาพเป็นกลุ่มของความสัมพันธ์ที่ไม่มีข้อกำหนดที่แน่นอน (Collection of arbitrary graph) ซึ่งมีความสัมพันธ์เป็นแบบกลุ่มกับกลุ่ม(many to many) ดังแสดงความสัมพันธ์ดังรูปที่ 2.27



รูปที่ 2.27 ตัวอย่างข้อมูลในฐานข้อมูลแบบเครือข่าย

3. โครงสร้างแบบสัมพันธ์ (Relational Model)

จะเป็นการแสดงรายละเอียดของข้อมูลและความสัมพันธ์ระหว่างข้อมูล ซึ่งอยู่ในรูปกลุ่มของตาราง (table) ซึ่งในแต่ละตารางเป็นตาราง 2 มิติ ประกอบด้วยแถว (row) และสดมภ์ (column) โดยที่ข้อมูลในแต่ละแถวต้องไม่ซ้ำกัน และแต่ละสดมภ์นั้นจะต้องมีชื่อไม่ซ้ำกันด้วย แต่ละตารางมีจำนวนแถวได้หลายแถว และจำนวนสดมภ์ได้หลายสดมภ์ ส่วนลำดับของแถวไม่มี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความสำคัญต่อความสัมพันธ์ของข้อมูล และสามารถแสดงความสัมพันธ์ระหว่างข้อมูลอยู่ในรูปของตารางได้ โดยไม่ต้องมีตัวชี้หรือลิงค์ลิสต์ (link list) มาเกี่ยวข้องในการแสดงความสัมพันธ์เพื่อบอกตำแหน่งที่อยู่จริงของข้อมูลในจานแม่เหล็กของอีกเรคคอร์ดหนึ่ง เหมือนโมเดลแบบแผนภูมิต้นไม้หรือแบบเครือข่ายซึ่งเป็นผลให้โมเดลความสัมพันธ์นี้เป็นโมเดลที่ใช้งานง่าย เนื่องจากผู้ใช้ไม่จำเป็นต้องทราบเกี่ยวกับการจัดเก็บข้อมูลในระดับกายภาพแต่สามารถใช้ตัวชี้(index)มาเกี่ยวข้องเพื่อประโยชน์ในการจัดการข้อมูลเท่านั้นซึ่งจะไม่เกี่ยวข้องกับความสัมพันธ์ระหว่างข้อมูล การแสดงความสัมพันธ์ของข้อมูลระหว่างแฟ้มข้อมูลนั้น เราจะสามารถมองเห็นได้จากตัวข้อมูลที่เก็บอยู่ในแฟ้มข้อมูลเลยตัวอย่างการเก็บข้อมูลในฐานข้อมูลเชิงสัมพันธ์แสดงดังตารางที่ 2.4

ตารางที่ 2.4 ตัวอย่างการเก็บข้อมูลในฐานข้อมูลเชิงสัมพันธ์

Name	Street	City	Number
Lowery	Maple	Queens	900
Shiver	North	Bronx	556
Shiver	North	Bronx	647
Hodges	Sidehill	Brooklyn	801
Hodges	Sidehill	Brooklyn	647

number	balance
900	55
556	100000
647	105366
801	10533

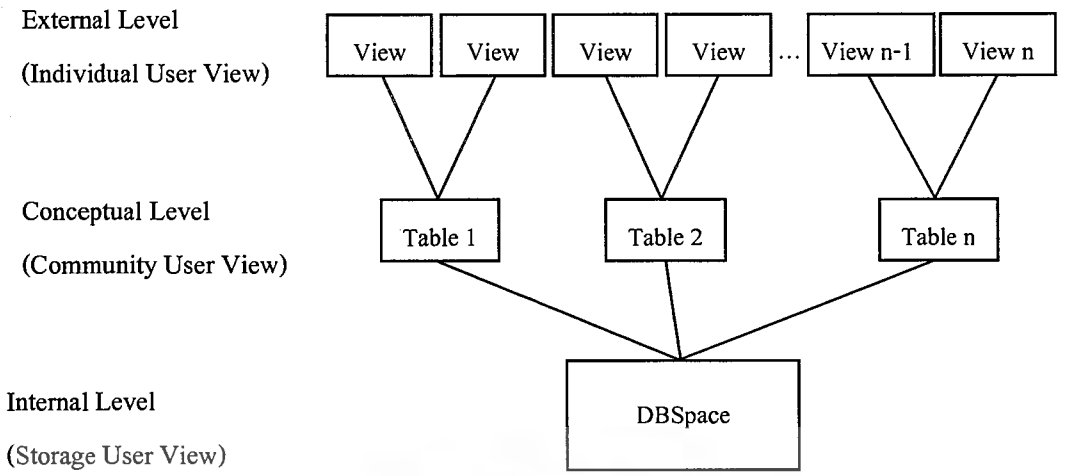
2.5.1.3 สถาปัตยกรรมมาตรฐานของระบบฐานข้อมูล

ได้มีนักวิทยาศาสตร์ของทั้งในยุโรปและอเมริกา พยายามกำหนดสถาปัตยกรรมที่เป็นมาตรฐานของระบบฐานข้อมูลเพื่อให้ฐานข้อมูลก่อประโยชน์ที่ใช้ และหน่วยงานที่ใช้สูงสุด โดยในที่สุด 3 องค์กรหลักคือ

- ISO (International Standard Organization)
- IFIP (International Federation for Information Processing)
- ANSI (American National Standard Institute)

ได้ให้การยอมรับเป็นสถาปัตยกรรมมาตรฐานของระบบฐานข้อมูลเรียกว่าสถาปัตยกรรม 3 นิยามข้อมูล (The 3-schema Architecture) ซึ่งระดับของข้อมูลทั้ง 3 นิยาม ดังแสดงดังรูปที่ 2.28 และมีรายละเอียดดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.28 สถาปัตยกรรม 3 นิยามข้อมูล (The 3-schema Architecture)

2.5.1.3.1 นิยามข้อมูลระดับภายนอก (External Schema)

จะเป็นการกำหนดโครงสร้างข้อมูลของฐานข้อมูลในระดับที่ผู้ใช้มองเห็น ซึ่งอาจเป็นบางส่วนของนิยามข้อมูลระดับแนวคิด เช่น ในระบบฐานข้อมูลเชิงสัมพันธ์นั้น ผู้ใช้บางคนอาจต้องใช้ข้อมูลผ่านวิว (view) ซึ่งจะมีสิทธิใช้ข้อมูลบางแถว หรือบางสดมภ์ของตารางเท่านั้น ดังนั้นผู้ใช้เหล่านี้จะมองเห็นเฉพาะข้อมูลที่ผู้ดูแลและควบคุมฐานข้อมูล (Database Administration: DBA) หรือผู้มีอำนาจกำหนดสิทธิของตารางนั้นๆ กำหนดขอบเขตการใช้ข้อมูลในตารางต่าง ๆ ให้เท่านั้น

2.5.1.3.2 นิยามข้อมูลระดับแนวคิด (Conceptual Schema)

จะเป็นการกำหนดลักษณะรูปแบบข้อมูล ขนาดของข้อมูล และความสัมพันธ์ของข้อมูลทั้งหมดในระบบงาน นั่นคือไม่ว่าฐานข้อมูลจะมีความสัมพันธ์ระหว่างข้อมูลในฐานข้อมูลอยู่ในรูปแบบใด ๆ ก็ตาม จะต้องกำหนดการแทนรูปแบบของข้อมูลในนิยามข้อมูลระดับแนวคิดนี้ เช่น ถ้าความสัมพันธ์ระหว่างข้อมูลอยู่ในรูปแบบเชิงสัมพันธ์ (Relational Model) ในระดับนี้จะแสดงชื่อตาราง (Table Name) ชื่อสดมภ์ (Column Name) ชนิดข้อมูลของแต่ละสดมภ์ตลอดจนชื่อของคีย์หลัก (Primary Key) และชื่อคีย์นอก (Foreign Key) เป็นต้น

2.5.1.3.3 นิยามข้อมูลระดับภายใน (Internal Schema)

จะเป็นการกำหนดลักษณะโครงสร้างขอข้อมูลที่ถูกจัดเก็บจริงในอุปกรณ์การเก็บข้อมูล เช่น ถ้าความสัมพันธ์ระหว่างข้อมูลอยู่ในรูปแบบเชิงสัมพันธ์ ซึ่งในระดับแนวคิดและระดับภายนอกจะแสดงอยู่ในรูปแบบของตาราง แต่เมื่อข้อมูลของตารางนั้น ๆ ถูกจัดเก็บจริง ๆ ในหน่วยความจำสำรอง ข้อมูลอาจถูกจัดเก็บด้วยรูปแบบของโครงสร้างข้อมูลแบบบีทรี (B-Tree)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แซม (C-ISAM) หรือลิงค์ลิสต์ก็ได้ ซึ่งการจัดเก็บข้อมูลในระดับนี้ ระบบการจัดการฐานข้อมูล (Database Management System: DBMS) จะเป็นตัวจัดการให้โดยที่ผู้ใช้ไม่ต้องจัดการเอง

เหตุผลที่มีการแบ่งสถาปัตยกรรมของฐานข้อมูลออกเป็น 3 ระดับ เนื่องจาก

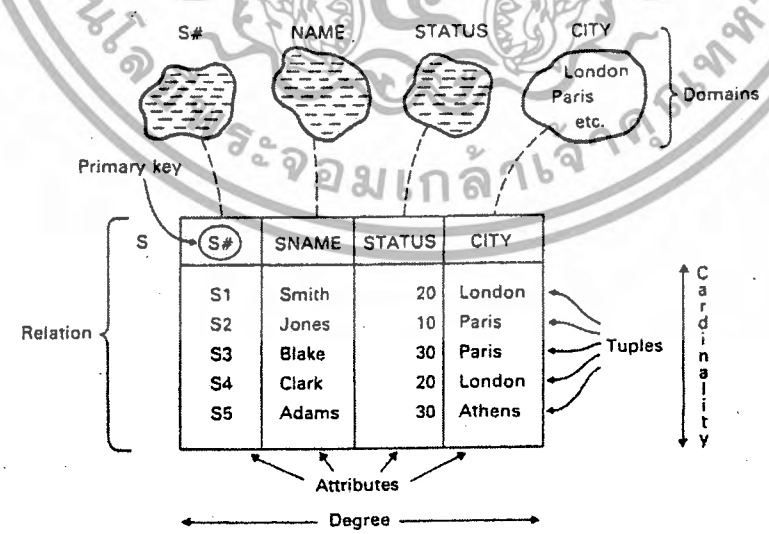
- ข้อมูลชุดเดียวกันสำหรับผู้ใช้แต่ละคนจะมองภาพ (view)
- วิธีการที่ผู้ใช้ต้องการใช้ในการดูข้อมูลอาจเปลี่ยนแปลงได้ตลอดเวลา
- ผู้ใช้ไม่มีความจำเป็นต้องรับรู้ความซับซ้อนของโครงสร้างที่เก็บข้อมูล
- ผู้ดูแลระบบฐานข้อมูล (DBA)นั้น ควรจะสามารถทำการเปลี่ยนแปลงโครงสร้างของข้อมูลได้โดยไม่กระทบมุมมองของผู้ใช้
- โครงสร้างข้อมูลนั้นไม่ควรจะถูกกระทบ โดยการเปลี่ยนแปลงการจัดเก็บข้อมูลทางกายภาพ เช่น เปลี่ยนอุปกรณ์ (device) ในการจัดเก็บข้อมูล

2.5.1.4 ฐานข้อมูลเชิงสัมพันธ์

2.4.1.4.1 ความหมาย

ฐานข้อมูลเชิงสัมพันธ์ เป็นฐานข้อมูลที่รวบรวมความสัมพันธ์ระหว่างข้อมูลไว้ โดยในแต่ละความสัมพันธ์จะมองเห็นลักษณะของตาราง 2 มิติ คือ แถวและสดมภ์ โดยแต่ละแถวจะเรียกว่า Tuple และแต่ละสดมภ์จะเรียกว่า Attribute หรือ Field

2.5.1.4.2 โครงสร้างข้อมูลแบบเชิงสัมพันธ์



รูปที่ 2.29 แสดงรายละเอียดของรีเลชั่น S

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และสามารถเปรียบเทียบที่ใช้ในเชิงวิชาการกับคำศัพท์ที่ใช้ทั่วๆ ไป ได้ดังนี้

ศัพท์ทางวิชาการศัพท์เชิงธุรกิจ

Relation table ที่มี row ไม่ซ้ำกัน

Tuple row หรือ record

Cardinality จำนวนของ row ในแต่ละ table

attributecolumn หรือ field

degree จำนวน column ในแต่ละ table

primary key column อาจหมายถึง 1 column หรือมากกว่า 1 column ที่เมื่อระบุค่าของข้อมูลใน column ที่เป็น primary key 1 ค่า จะได้ข้อมูลใน table นั้น ออกมาเพียง 1 row

domain ค่าของข้อมูลที่เป็นไปได้ทั้งหมดในแต่ละ column ประกอบด้วยสิ่งต่อไปนี้

1) Relation หมายถึง ข้อมูลที่มีความสัมพันธ์กันซึ่งเก็บรวบรวมไว้ด้วยกันในรูปแบบตาราง เรียกว่า Table ซึ่งมีแถว (Row) ไม่ซ้ำกัน

2) Tuple หมายถึง ข้อมูลในแต่ละแถวของ Relation หรือเรียกว่า Row หรือ Record

3) Attribute หมายถึง รายละเอียด หรือคุณลักษณะของข้อมูลที่ประกอบกันขึ้นเป็นตาราง เรียกว่า Field หรือ Column

4) Domain หมายถึง ขอบเขตของค่าข้อมูลที่ควรจะเป็นในแต่ละแอททริบิวต์ (Attribute)

5) Candidate key หมายถึง Attribute ที่มีคุณสมบัติของข้อมูลซึ่งมีค่าเป็นหนึ่งเดียว (Unique) โดยที่แต่ละ Relation อาจมีมากกว่า 1 แอททริบิวต์ที่มีคุณสมบัติดังกล่าว

6) Primary key หมายถึง Candidate key ที่ได้รับเลือกให้เป็นคีย์หลักของ Relation

7) Foreign key หมายถึง Attribute หรือกลุ่มของแอททริบิวต์ในตารางความสัมพันธ์ที่ใช้ในการอ้างถึงแอททริบิวต์เดียวกันในอีกตารางหนึ่ง หรืออาจเป็นตารางเดิมก็ได้ ทั้งนี้แอททริบิวต์นี้ต้องมีคุณสมบัติเป็นคีย์หลักของตารางที่ถูกอ้างอิง

8) Composite key หมายถึง Primary key ที่เกิดจากหลาย ๆ แอททริบิวต์มาประกอบกัน เพื่อที่จะหาค่าที่เป็นหนึ่งเดียว (Unique)

9) Degree หมายถึง จำนวนสดมภ์ในแต่ละตาราง

10) ค่าว่าง (null value) หมายถึง แอททริบิวต์ที่ไม่มีค่าข้อมูลเก็บอยู่ ซึ่งค่าว่างจะไม่ใช่ช่องว่าง หรือค่าศูนย์ แต่จะเป็นค่าที่ไม่ได้อยู่ในกรอบของโดเมน โดยทั่วไปเมื่อมีข้อมูลใหม่เข้ามาและยังไม่ได้กำหนดค่าให้กับบางแอททริบิวต์ระบบจะถือว่าแอททริบิวต์นั้นมีค่าเป็น null

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.1.4.3 คุณสมบัติของ Relation มี 4 ข้อดังต่อไปนี้

1) ไม่มีข้อมูลซ้ำกันในแต่ละแถว (There are no duplicate tuples)

2) ข้อมูลในตารางไม่จำเป็นต้องเรียงลำดับในแต่ละแถว

3) ไม่จำเป็นต้องเรียงลำดับของแอททริบิวต์ (Attribute are unordered)

4) ข้อมูลในแต่ละ Attribute นั้น จะต้องมีเพียงค่าเดียว (All attribute values are atomic)

2.5.1.5 กฎการคงสภาพของข้อมูล (Integrity Rule)

ความคงสภาพของข้อมูล หมายถึง ความถูกต้องและความสมเหตุสมผลของข้อมูล เมื่อมีการเพิ่ม ลบ แก้ไขข้อมูลในฐานข้อมูล DBMS จะทำหน้าที่ให้ข้อมูลมีความคงสภาพอยู่เสมอ

1) กฎควบคุมความถูกต้องของเอนทิตี้ (The Entity Integrity Rule)

มีคำนิยามว่า จะไม่มีแอททริบิวต์ที่เป็นส่วนประกอบของคีย์หลักตัวใดในรีเลชั่นได้รับอนุญาตให้มีค่าเป็นค่าว่าง (No component of the primary key of a base relation is allowed to accept nulls) โดยที่ค่าว่างค่านี้จะเป็นค่าที่ไม่สามารถทราบค่าได้ (Unknown value) ซึ่งจะไม่ใช่ค่าศูนย์ หรือว่องว่าง (Blank) จะเห็นว่ากฎข้อนี้จะใช้ควบคุมความถูกต้องของคีย์หลัก

2) กฎควบคุมความถูกต้องของการการอ้างอิง (The Referential Integrity Rule)

มีคำนิยามว่า ในฐานข้อมูลนั้นจะต้องไม่มีการเก็บค่าของคีย์รองที่ไม่มีความสัมพันธ์กับค่าของคีย์หลัก จะเห็นว่ากฎข้อนี้จะใช้ควบคุมความถูกต้องของคีย์นอก (Foreign Key) ที่มีความสัมพันธ์กับคีย์หลักโดยที่จะไม่ครอบคลุมความสัมพันธ์กับคีย์รอง (Alternate Key)

2.5.2 โปรแกรมฐานข้อมูลที่นิยมใช้

โปรแกรมฐานข้อมูล เป็นโปรแกรมหรือซอฟแวร์ที่ช่วยจัดการข้อมูลหรือรายการต่าง ๆ ที่อยู่ในฐานข้อมูล ไม่ว่าจะเป็นการจัดเก็บ การเรียกใช้ การปรับปรุงข้อมูล

โปรแกรมฐานข้อมูล จะช่วยให้ผู้ใช้สามารถค้นหาข้อมูลได้อย่างรวดเร็ว ซึ่งโปรแกรมฐานข้อมมูลที่นิยมใช้มีอยู่ด้วยกันหลายตัว เช่น Access, FoxPro, Clipper, dBase, FoxBase, Oracle, SQL เป็นต้น โดยแต่ละโปรแกรมจะมีความสามารถต่างกัน บางโปรแกรมใช้ง่ายแต่จะจำกัดขอบเขตการใช้งาน บ่งโปรแกรมใช้งานยากกว่า แต่จะมีความสามารถในการทำงานมากกว่า

2.5.3 Apache

2.5.3.1 ประวัติ

Apache พัฒนามาจาก HTTPD Web Server ที่มีกลุ่มผู้พัฒนาอยู่ก่อนแล้ว โดย ร็อบ แม็คคูล (Rob McCool) ที่ NCSA (National Center for Supercomputing Applications) มหาวิทยาลัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

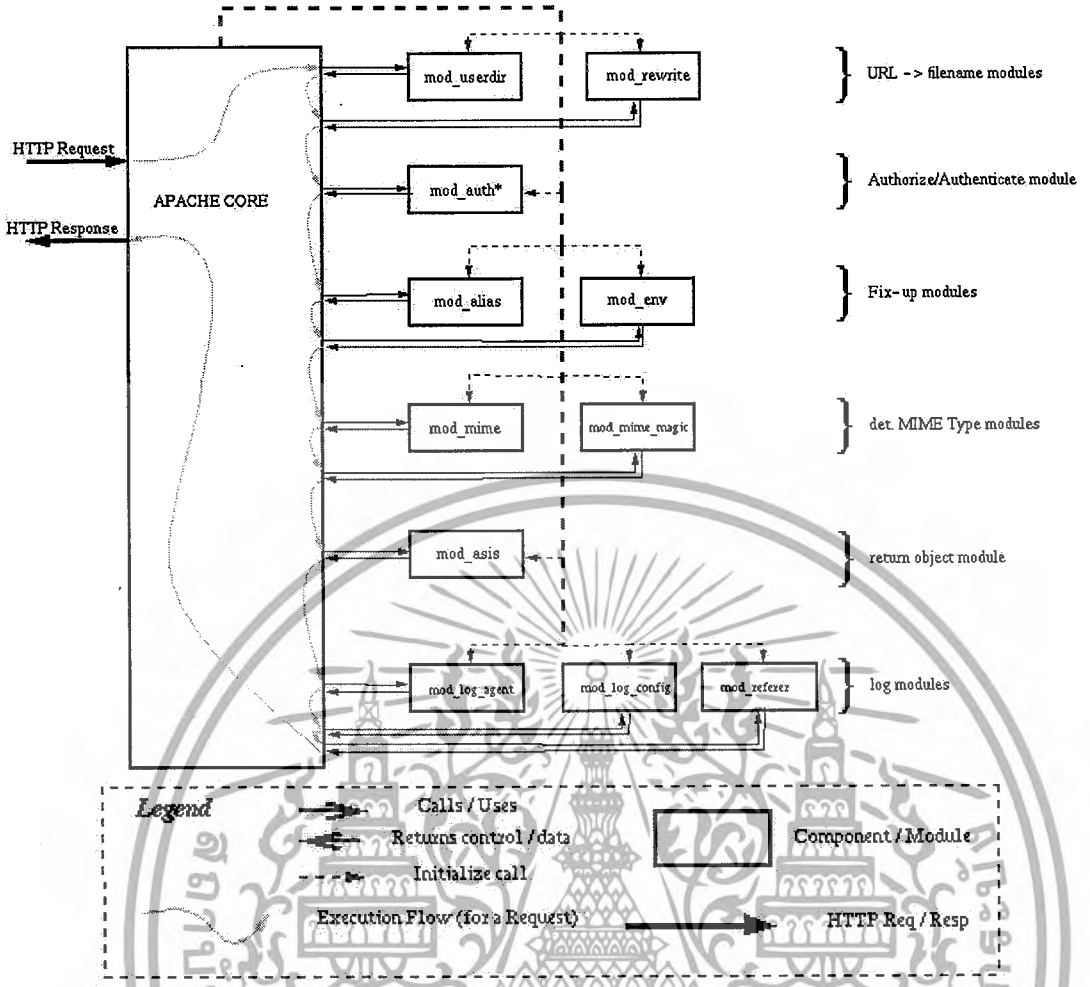
อิลลินอยส์ เออร์แบนา-แชมเปญจน์ สหรัฐอเมริกา แต่หลังจากที่ แม็คคูล ออกจาก NCS และหันไปให้ความสนใจกับโครงการอื่น ๆ มากกว่าทำให้ HTTPD เว็บเซิร์ฟเวอร์ ถูกปล่อยทิ้งไม่มีผู้พัฒนาต่อ แต่เนื่องจากเป็นซอร์ฟแวร์ที่อยู่ภายใต้ลิขสิทธิ์กนู คือ ทุกคนมีสิทธิ์ที่จะนำเอาซอร์สโค้ดไปพัฒนาต่อได้ ทำให้มีผู้ใช้กลุ่มหนึ่งได้พัฒนาโปรแกรมขึ้นมาเพื่ออุดช่องโหว่ที่มีอยู่เดิม (หรือ แพช) และยังได้รวบรวมเอาข้อมูลการพัฒนาและการแก้ไขต่าง ๆ แต่ข้อมูลเหล่านี้อยู่ตามที่ต่าง ๆ ไม่ได้รวมอยู่ในที่ที่เดียวกัน จนในที่สุด ไบอัน บีเลนดอร์ฟ (Brian Behlendorf) ได้สร้างจดหมายกลุ่ม (mailing list) ขึ้นมาเพื่อนำเอาข้อมูลเหล่านี้เข้าไว้เป็นกลุ่มเดียวกัน เพื่อให้สามารถเข้าถึงข้อมูลเหล่านี้ได้ง่ายยิ่งขึ้นและในที่สุด กลุ่มผู้พัฒนาได้เรียกตัวเองว่า กลุ่มอะแพชี(Apache Group) และได้ปล่อยซอฟต์แวร์ HTTPD เว็บเซิร์ฟเวอร์ที่พัฒนาโดยการนำเอาแพชหลาย ๆ ตัวที่ผู้ใช้ได้พัฒนาขึ้นเพื่อปรับปรุงการการทำงานของซอฟ์แวร์ตัวเดิมให้มีประสิทธิภาพมากยิ่งขึ้น

ตั้งแต่ ปี พ.ศ. 2539 Apache ได้รับความนิยมขึ้นเรื่อย ๆ จนปัจจุบันได้รับความนิยมเป็นอันดับหนึ่ง มีผู้ใช้งานอยู่ประมาณ 65% ของเว็บเซิร์ฟเวอร์ที่ให้บริการอยู่ทั้งหมด

2.5.3.2 สถาปัตยกรรมของของ Apache

Apache คือ Project ที่ทำหน้าที่เป็น web server ที่มีผู้ใช้ทั่วโลกมากกว่า 60 % หน้าที่ในการจัดเก็บ Home Page และส่ง Home Page ไปยัง Browser ที่มีการเรียกเข้ายัง Web server ที่เก็บHome Page นั้นอยู่ซึ่งปัจจุบันจัดได้ว่าเป็น web server ที่มีความน่าเชื่อถือมาก ซึ่งสามารถหามา Download ได้จาก website www.apache.org

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.30 สถาปัตยกรรมของ Apache

ฟังก์ชั่นของเว็บเซิร์ฟเวอร์ คือ การให้บริการตามที่ร้องขอผ่านทางโปรโตคอล HTTP โดยปกติเซิร์ฟเวอร์จะรับ request เพื่อระบุแหล่งที่มาและคืนกลับไปก็คือ response Client อาจจะอ้างอิงในไฟล์ที่ร้องขอมาแล้วไฟล์นั้นก็จะถูกส่งกลับ สรุปก็คือ เว็บเซิร์ฟเวอร์จะนำ request ไปถอดรหัสแล้วเก็บแหล่งที่มา จากนั้นถึงส่งคืน client นอกจากนี้ยังมีความสัมพันธ์เกี่ยวกับการควบคุมสิทธิในการเข้าถึง client ซึ่งเป็นหน้าที่ของเว็บเซิร์ฟเวอร์

2.5.3.3 ความสามารถ

การที่อาปาเช่เป็นซอฟต์แวร์ที่อยู่ในลักษณะของโอเพ่นซอร์ส ที่เปิดให้บุคคลทั่วไปสามารถเข้ามาร่วมพัฒนาส่วนต่าง ๆ ของอาปาเช่ได้ ซึ่งทำให้เกิดเป็นโมดูลที่เกิดประโยชน์มากมายเช่น mod_perl, mod_python หรือ mod_php ซึ่งเป็นโมดูลที่ทำให้อาปาเช่สามารถใช้ประโยชน์และทำงานร่วมกับภาษาอื่นได้ แทนที่จะเป็นเพียงเซิร์ฟเวอร์ที่ให้บริการเพียงแค่ HTML ได้เพียงอย่างเดียว นอกจากนี้อาปาเช่เองยังมีความสามารถอื่น ๆ ด้วย เช่น การยืนยันตัวบุคคล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(mod_auth,mod_access, mod_digest) หรือเพิ่มความปลอดภัยในการสื่อสารผ่าน โปรโตคอล https (mod_ssl)นอกจากนี้ ก็ยังมีโมดูลอื่น ๆ ที่ได้รับความนิยมใช้ เช่น mod_vhost ทำให้สามารถสร้างโฮสท์เสมือน www.sample.com, wiki.sample.com, mail.sample.com หรือ www.ilovewiki.org ภายในเครื่องเดียวกันได้ หรือ mod_rewrite เป็นเครื่องมือที่จะช่วยให้ url ของเว็บนั้นอ่านง่ายขึ้น

2.5.4 PHP

PHP เกิดในปี 1994 โดย Rasmus Lerdorf โปรแกรมเมอร์ชาวสหรัฐอเมริกาได้คิดค้นสร้างเครื่องมือที่ใช้ในการพัฒนาเว็บส่วนตัวของเขา โดยใช้ข้อดีของภาษา C และ Perl เรียกว่า Personal Home Page และได้สร้างส่วนติดต่อกับฐานข้อมูลชื่อว่า Form Interpreter (FI) รวมทั้งสองส่วนเรียกว่า PHP/FI ซึ่งก็เป็นจุดเริ่มต้นของ PHP มีคนที่เข้ามาเยี่ยมชมเว็บไซต์ของเขาแล้วเกิดชอบจึงติดต่อขอเอาโค้ดไปใช้บ้าง และนำไปพัฒนาต่อ ในลักษณะของ Open Source ภายหลังมีความนิยมขึ้นเป็นอย่างมากภายใน 3 ปีมีเว็บไซต์ที่ใช้ PHP/FI ในติดต่อฐานข้อมูลและแสดงผลแบบ ไดนามิก และอื่น ๆ มากกว่า 50000 ไซต์

PHP เป็นภาษาสคริปต์ที่ประมวลผลที่ฝั่งเซิร์ฟเวอร์ แล้วส่งผลลัพธ์ไปแสดงผลที่ฝั่งไคลเอ็นต์ผ่านบราวเซอร์เช่นเดียวกับ CGI และ ASP ต่อมาเมื่อมีผู้ใช้มากขึ้นจึงมีการร้องขอให้มีการพัฒนาประสิทธิภาพของ PHP/FI ให้สูงขึ้น Rasmus Lerdorf ก็ได้ผู้ที่มาช่วยพัฒนาอีก 2 คนคือ Zeev Suraski และ Andi Gutmans ชาวอิสราเอล ซึ่งปรับปรุงโค้ดของ Lerdorf ใหม่โดยใช้ C++ ต่อมาก็มีเพิ่มเข้ามาอีก 3 คน คือ Stig Bakken รับผิดชอบความสามารถในการติดต่อ Oracle, Shane Caraveo รับผิดชอบดูแล PHP บน Window 9x/NT, และ Jim Winstead รับผิดชอบการตรวจ ความบกพร่องต่าง ๆ และได้เปลี่ยนชื่อเป็น Professional Home Page

PHP3 ได้ออกสู่สายตาของนักโปรแกรมเมอร์เมื่อ มิถุนายน 1998 ที่ผ่านมาในเวอร์ชั่นนี้มีคุณสมบัติเด่นคือสนับสนุนระบบปฏิบัติการทั้ง Window 95/98/ME/NT, Linux และเว็บเซร์ฟเวอร์อย่าง IIS, PWS, Apache, OmniHTTPd สนับสนุน ระบบฐานข้อมูลได้หลายรูปแบบเช่น SQL Server, MySQL, mSQL, Oracle, Informix, ODBC

เวอร์ชั่นล่าสุดในปัจจุบันคือ PHP4 ซึ่งได้เพิ่ม Functions การทำงานในด้านต่าง ๆ ให้มากและง่ายขึ้นโดย Zend ซึ่งมี Zeev และ Andi Gutmans ได้ร่วมก่อตั้งขึ้น (http://www.zend.com) ในเวอร์ชั่นนี้จะเป็น compile script ซึ่งในเวอร์ชั่นหน้านี้จะเป็น embed script interpreter ในปัจจุบันมีคนใช้ PHP สูงกว่า 5,100,000 sites แล้วทั่วโลก ผู้พัฒนาได้ตั้งชื่อของ PHP ใหม่ว่า PHP: Hypertext Preprocessor ซึ่งหมายถึงมีประสิทธิภาพระดับโปรเฟสเซอร์สำหรับไฮเปอร์เท็กซ์ ความสามารถของ PHP นั้นในความสามารถพื้นฐานที่ภาษาสคริปต์ทั่ว ๆ ไปมีนั้น PHP ก็มีความสามารถทำได้ทัดเทียมเช่นเดียวกันเช่น การรับข้อมูลจากฟอร์ม, การสร้าง Content ในลักษณะ Dynamic, รับส่ง Cookies, สร้าง, เปิด, อ่าน และปิดไฟล์ในระบบ, การรองรับระบบจัดการฐานข้อมูลมากมายดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.5 PHP

Adabas D	Ingres	Oracle (OCI7 and OCI8)
Dbase	InterBase	Ovrimos
Empress	FrontBase	PostgreSQL
FilePro (read-only)	mSQL	Solid
Hyperwave	Direct MS-SQL	Sybase
IBM DB2	MySQL	Velocis
Informix	ODBC	Unix dbm

แต่ตัวจัดการฐานข้อมูลที่ทาง NINETO E-MAGAZINE ONLINE เลือกมาใช้ในบทความนี้คือ MySQL เหตุที่เลือกตัวนี้คือ เป็นที่นิยมกว้างขว้างและประเด็นหนึ่งที่จะต้องพิจารณาคือ Free เพราะ MySQL จัดเป็น Software ประเภท Freeware รองรับ OS ได้หลายระบบด้วยกัน

Protocol Support ความสามารถในการรองรับโปรโตคอลหลายแบบทั้ง IMAP, SNMP, NNTP, POP3, HTTP และยังมีไลบารีสำหรับติดต่อ กับแอพพลิเคชั่นได้มากมาย มีความยืดหยุ่นสูงสามารถนำไปสร้างแอพพลิเคชั่นได้หลากหลาย และอีกข้อดีหนึ่งที่โดเด่นคือของ PHP ก็คือสามารถแทรกลงในแท็ก HTML ในตำแหน่งใดก็ได้

2.5.4.1 การใช้ PHP

เนื่องจากว่า PHP ไม่ได้เป็นส่วนหนึ่งของตัว Web Server ดังนั้นถ้าจะใช้ PHP ก็จะต้องดูก่อนว่า Web server นั้นสามารถใช้สคริปต์ PHP ได้หรือไม่ ยกตัวอย่างเช่น PHP สามารถใช้ได้กับ Apache WebServer และ Personal Web Server (PWP) สำหรับระบบปฏิบัติการ Windows 95/98/NT

ในกรณีของ Apache เราสามารถใช้ PHP ได้สองรูปแบบคือ ในลักษณะของ CGI และ Apache Module ความแตกต่างอยู่ตรงที่ว่า ถ้าใช้ PHP เป็นแบบโมดูล PHP จะเป็นส่วนหนึ่งของ Apache หรือเป็นส่วนขยายในการทำงานนั้นเอง ซึ่งจะทำงานได้เร็วกว่าแบบที่เป็น CGI เพราะว่า ถ้าเป็น CGI แล้ว ตัวแปลชุดคำสั่งของ PHP ถือว่าเป็นแค่โปรแกรมภายนอก ซึ่ง Apache จะต้องเรียกขึ้นมาทำงานทุกครั้ง ที่ต้องการใช้ PHP ดังนั้น ถ้ามองในเรื่องของประสิทธิภาพในการทำงาน การใช้ PHP แบบที่เป็นโมดูลหนึ่งของ Apache จะทำงานได้มีประสิทธิภาพมากกว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.4.2 รูปแบบการเขียน PHP

การเขียนโค้ดเราสามารถเขียนได้จากโปรแกรม Editor ทั่วไปเช่น Notepad หรือ Editplus แน่นอนที่สะดวกที่สุดคงจะไม่พ้น Notepad เพราะแถมมากับ window อยู่แล้ว แต่ถ้าต้องการความสามารถและ Options ที่เพิ่มขึ้นก็แนะนำว่าโปรแกรม Editplus ใช้ได้ดีทีเดียว

รูปแบบการเขียน PHP เขียนได้ 4 แบบดังตัวอย่าง ที่นิยมคือแบบที่ 1 และ 2 แบบที่ 3 ใช้งานคล้ายกับ Java script ส่วนแบบที่ 4 ตัว tag <% จะเหมือนกับ ASP โดยเมื่อรันจะได้ผลลัพธ์เหมือนกัน และสามารถแทรกลงในส่วนของภาษา HTML ส่วนใดก็ได้

1.การเขียนโค้ดในรูปแบบภาษา SGML จะมีรูปแบบดังนี้

```
<?
คำสั่งในภาษาPHP;
?>
```

2. การเขียนโค้ดเพื่อใช้ร่วมกับภาษา XHTML หรือ XML (แต่สามารถใช้ใน HTML แบบปกติได้) จะมีรูปแบบดังนี้

```
<?php
คำสั่งในภาษาPHP;
?>
```

3. การเขียนโค้ดในรูปแบบ JavaScript จะมีรูปแบบดังนี้

```
<ScriptLanguage="php">
คำสั่งในภาษาPHP;
</Script>
```

4. การเขียนโค้ดในรูปแบบ ASP จะมีรูปแบบดังนี้

```
<%
คำสั่งในภาษาPHP;
%>
```

สำหรับรูปแบบที่ 4 จะใช้ได้กับ PHP 3.0.4 ขึ้นไป และจะต้องไปแก้ไฟล์ php.ini ในโฟลเดอร์ C:\WINDOWS เสียก่อนโดยให้ asp_tags มีค่าเป็น On

การเขียนสคริปต์ PHP ในรูปแบบใดก็ตามจะต้องมีเครื่องหมาย semicolon (;) ลงท้ายคำสั่งเสมอเหมือนกับการเขียนภาษา C กับภาษา Perl และคำสั่งหรือฟังก์ชั่นในภาษา PHP จะเขียนด้วยตัวพิมพ์เล็กหรือพิมพ์ใหญ่ก็ได้ (case-insensitive) การจบ statement หรือสิ้นสุด script เราจะปิดท้ายสคริปต์ด้วยแท็ก (?>) และคำสั่งสุดท้ายในสคริปต์นั้นจะลงท้ายด้วย semicolon (;) หรือไม่ก็ได้เพราะจะถูกปิดด้วยแท็ก (?>) อยู่แล้ว

นอกจากรูปแบบแล้ว การวาง code ผสมกับ HTML ก็เป็นวิธีหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
<html>
<head>
<title>Example</title>
</head>
<body>
<?php
echo " Hi, I' ma PHP script!";
?>
</body>
```

2.5.4.3 การรองรับ PHP

คำสั่งของพีเอชพี สามารถสร้างผ่านทางโปรแกรมแก้ไขข้อความทั่วไป เช่น โน้ตแพด หรือ vi ซึ่งทำให้การทำงานพีเอชพี สามารถทำงานได้ในระบบปฏิบัติการหลักเกือบทั้งหมด โดยเมื่อเขียนคำสั่งแล้วนำมาประมวลผล Apache, Microsoft Internet Information Services (IIS) , Personal Web Server, Netscape และ iPlanet servers, Oreilly Website Pro server, Caudium, Xitami, OmniHTTPd, และอื่น ๆ อีกมากมาย สำหรับส่วนหลักของ PHP ยังมี Module ในการรองรับ CGI มาตรฐาน ซึ่ง PHP สามารถทำงานเป็นตัวประมวลผล CGI และ PHP มีอิสรภาพในการเลือกระบบปฏิบัติการและเว็บเซิร์ฟเวอร์ นอกจากนี้ยังสามารถใช้สร้างโปรแกรมโครงสร้าง สร้างโปรแกรมเชิงวัตถุ (OOP) หรือสร้างโปรแกรมที่รวมทั้งสองอย่างเข้าด้วยกัน แม้ว่าความสามารถของคำสั่ง OOP มาตรฐานในเวอร์ชันนี้ยังไม่สมบูรณ์ แต่ตัวไลบรารีทั้งหลายของโปรแกรมและตัวโปรแกรมประยุกต์ (รวมถึง PEAR library) ได้ถูกเขียนขึ้นโดยใช้รูปแบบการเขียนแบบ OOP เท่านั้น

พีเอชพีสามารถทำงานร่วมกับฐานข้อมูลได้หลายชนิด ซึ่งฐานข้อมูลส่วนหนึ่งที่รองรับได้แก่ ออราเคิล dBase PostgreSQL IBM DB2 MySQL Informix ODBC โครงสร้างของฐานข้อมูลแบบ DBX ซึ่งทำให้พีเอชพีใช้กับฐานข้อมูลอะไรก็ได้ที่รองรับรูปแบบนี้ และ PHP ยังรองรับ ODBC (Open Database Connection) ซึ่งเป็นมาตรฐานการเชื่อมต่อฐานข้อมูลที่ใช้กันแพร่หลายอีกด้วย สามารถเชื่อมต่อกับฐานข้อมูลต่าง ๆ ที่รองรับมาตรฐานโลกนี้ได้

พีเอชพียังสามารถรองรับการสื่อสารกับการบริการในโปรโตคอลต่าง ๆ ได้ เช่น LDAP IMAP SNMP NNTP POP3 HTTP COM (บนวินโดวส์) และอื่น ๆ อีกมากมาย คุณสามารถเปิด Socket บนเครื่อข่ายโดยตรง และ ตอบโต้โดยใช้ โปรโตคอลใด ๆ ก็ได้ PHP มีการรองรับสำหรับการแลกเปลี่ยนข้อมูลแบบ WDDX Complex กับ Web Programming อื่น ๆ ทั่วไปได้ พูดถึง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในส่วน Interconnection พีเอชพีมีการรองรับสำหรับ Java objects ให้เปลี่ยนมันเป็น PHP Object แล้วใช้งาน ยังสามารถใช้รูปแบบ CORBA เพื่อเข้าสู่ Remote Object ได้เช่นกัน

2.5.5 My SQL

MySQL (มายเอสคิวแอล) เป็นระบบจัดการฐานข้อมูลโดยใช้ภาษา SQL แม้ว่า MySQL เป็นซอฟต์แวร์โอเพนซอร์ส แต่แตกต่างจากซอฟต์แวร์โอเพนซอร์สทั่วไป โดยมีการพัฒนาภายใต้บริษัท MySQL AB ในประเทศสวีเดน โดยจัดการ MySQL ทั้งในแบบที่ให้ใช้ฟรีและแบบที่ใช้ในเชิงธุรกิจ

MySQL สร้างขึ้นโดยชาวสวีเดน 2 คนและชาวฟินแลนด์ ชื่อ David Axmark, Allan Larsson และ Michael "Monty" Widenius.

ปัจจุบันบริษัทซันไมโครซิสเต็มส์ (Sun Microsystems, Inc.) เข้าซื้อกิจการของ MySQL AB เรียบร้อยแล้ว ฉะนั้นผลิตภัณฑ์ภายใต้ MySQL AB ทั้งหมดจะตกเป็นของซัน

2.5.5.1 ชนิดของข้อมูลที่สนับสนุน

ชนิดข้อมูลที่ MySQL สนับสนุนแบ่งเป็นสามประเภทหลักใหญ่ๆ

- ชนิดข้อมูลที่เป็นตัวเลข

BIT (มีใช้ได้กับ MyISAM, InnoDB, Memory)

TINYINT

SMALLINT

MEDIUMINT

INT

BIGINT

- ชนิดข้อมูลที่เกี่ยวกับวันที่และเวลา

DATETIME

DATE

TIMESTAMP

TIME

YEAR

- ชนิดข้อมูลที่เกี่ยวกับตัวอักษร

CHAR

VARCHAR

BINARY

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

VARBINARY

BLOB

TEXT

ENUM

SET

2.5.5.2 การใช้งาน

MySQL เป็นที่นิยมใช้กันมากสำหรับฐานข้อมูลสำหรับเว็บไซต์ เช่น มีเดียวิกิ และ phpBB และนิยมใช้งานร่วมกับภาษาโปรแกรม PHP ซึ่งมักจะได้ชื่อว่าเป็นคู่ จะเห็นได้จากคู่มือคอมพิวเตอร์ต่าง ๆ ที่จะสอนการใช้งาน MySQL และ PHP ควบคู่กันไป นอกจากนี้ หลายภาษาโปรแกรมที่สามารถทำงานร่วมกับฐานข้อมูล MySQL ซึ่งรวมถึง ภาษาซี ซีพลัสพลัส ปาสคาล ซีชาร์ป ภาษาจาวา ภาษาเพิร์ล พีเอชพี ไพทอน รูบี และภาษาอื่น ใช้งานผ่าน API สำหรับโปรแกรมที่ติดต่อผ่าน ODBC หรือ ส่วนเชื่อมต่อกับภาษาอื่น (database connector) เช่น เอเอสพี สามารถเรียกใช้ MySQL ผ่านทาง MyODBC,ADO,ADO.NET เป็นต้น

2.5.5.3 ส่วนเชื่อมต่อกับภาษาการพัฒนาอื่น (database connector)

มีส่วนติดต่อ (interface) เพื่อเชื่อมต่อกับภาษาในการพัฒนาอื่น ๆ เพื่อให้เข้าถึงฟังก์ชันการทำงานกับฐานข้อมูล MySQL ได้ เช่น ODBC (Open Database Connector) อันเป็นมาตรฐานกลางที่กำหนดมาเพื่อให้ใช้เป็นสะพานในการเชื่อมต่อกับโปรแกรมหรือระบบอื่น ๆ เช่น MyODBC อันเป็นไดรเวอร์เพื่อใช้สำหรับการเชื่อมต่อในระบบปฏิบัติการวินโดว์, JDBC คลาสส่วนเชื่อมต่อสำหรับ Java เพื่อใช้ในการติดต่อกับ MySQL และมี API (Application Programming Interface) ต่าง ๆ มีให้เลือกใช้มากมายในการที่เข้าถึง MySQL โดยไม่ขึ้นอยู่กับภาษาการพัฒนาใดภาษาหนึ่งนอกเหนือจากตัวเชื่อมต่อกับภาษาอื่น (Connector) ที่ได้กล่าวมาแล้วยังมีAPI ที่สนับสนุนในขณะนี้คือ DBI สำหรับการเชื่อมต่อกับภาษา perl

Ruby สำหรับการเชื่อมต่อกับภาษา ruby

Python สำหรับการเชื่อมต่อกับภาษา python

.NET สำหรับการเชื่อมกับภาษา .NET framework

MySQL++ สำหรับเชื่อมต่อกับภาษา C++

Ch สำหรับการเชื่อมต่อกับ Ch (C/C++ interpreter)

ยังมีโปรแกรมอีกตัวซึ่งเป็นโปรแกรมบริหารพัฒนาโดยผู้อื่น ซึ่งใช้กันอย่างแพร่หลายและนิยมกันเขียนในภาษาพีเอชพี เป็นโปรแกรมเว็บแอปพลิเคชัน ชื่อ phpMyAdmin

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.5.4 โครงสร้างของภาษา SQL

โครงสร้างของภาษา SQL ประกอบไปด้วย 2 ส่วนใหญ่ คือ

1. Data Definition Language (DDL) เป็นกลุ่มคำสั่งในภาษา SQL ที่ใช้สำหรับจัดการโครงสร้างของฐานข้อมูล เช่น การสร้างฐานข้อมูล, ปรับปรุงโครงสร้าง ของฐานข้อมูล เป็นต้น ตัวอย่างการใช้งานกลุ่มคำสั่ง DDL นี้ก็คือ การสร้างฐานข้อมูลด้วย MS SQL Server 7.0 ก็จะมีการใช้งานคำสั่งในกลุ่มDDLเป็นหลัก

2. Data Manipulation Language (DML) เป็นกลุ่มคำสั่งในภาษา SQL ที่ใช้สำหรับจัดการข้อมูลในฐานข้อมูล เช่น การแสดงข้อมูลแบบมีเงื่อนไข, การลบข้อมูล, การเพิ่มข้อมูล และการแสดงข้อมูลที่มาจากตารางหลายตาราง เป็นต้น

กลุ่มคำสั่ง DML ที่มีความสำคัญ และใช้กันอยู่เสมอ ประกอบไปด้วย 4 คำสั่งคือ

1.คำสั่ง DELETE เป็นคำสั่งที่ใช้สำหรับลบข้อมูล หรือลบ record ใด ๆ ออกจากตาราง มีรูปแบบการใช้งาน 2 ลักษณะ ดังนี้

รูปแบบที่ 1 DELETE FROM tablename WHERE criteria

รูปแบบที่ 2 DELETE * FROM tablename

ตัวแปร tablename หมายถึง ชื่อของตารางที่ต้องการลบ

ตัวแปร criteria หมายถึง เงื่อนไขในการลบข้อมูลหรือลบ record

เครื่องหมาย * หมายถึง ข้อมูลใด ๆ

2. คำสั่ง INSERT เป็นคำสั่งที่ใช้สำหรับเพิ่มข้อมูล หรือเพิ่ม record เข้าไปในตาราง มีรูปแบบการใช้งาน 2 ลักษณะ ดังนี้

รูปแบบที่ 1 INSERT INTO tablename(field1,field2,field3...)VALUES(value1,value2,value3,...)

รูปแบบที่ 2 INSERT INTO tablename1 SELECT * FROM tablename2 WHERE criteria

ตัวแปร tablename หมายถึง ชื่อตารางที่ต้องการเพิ่ม record เข้าไป

ตัวแปร field1-fieldn หมายถึง ชื่อของฟิลด์ต่าง ๆ ในตาราง tablename ซึ่งจะต้องเรียงตามลำดับของฟิลด์ในตารางดังกล่าวด้วย

ตัวแปร value1-valuen หมายถึง ค่าของฟิลด์นั้น ๆ

ตัวแปร tablename1 หมายถึง ชื่อของตารางที่ต้องการเพิ่มข้อมูลเข้าไป

ตัวแปร tablename2 หมายถึง ชื่อของตารางที่ต้องการดึงข้อมูลออกมา แล้วเพิ่มเข้าไปในตาราง tablename1

ตัวแปร criteria หมายถึง เงื่อนไขในการดึงข้อมูลจากตาราง tablename2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. คำสั่ง SELECT ใช้สำหรับเลือกหรือดึงข้อมูลที่เราต้องการจากฐานข้อมูล เป็นคำสั่งที่มีความยืดหยุ่นสูงมาก เพราะว่าเงื่อนไขในการนำข้อมูลออกมาจากตารางมีมากมาย แต่มีรูปแบบการใช้งานหลัก ๆ อยู่ 2 ลักษณะ

รูปแบบที่ 1 SELECT * FROM tablename

รูปแบบที่ 2 SELECT fieldname1, fieldname2, ...,fieldname-n FROM tablename WHERE criteria

เครื่องหมาย * หมายถึง ข้อมูลใด ๆ

ตัวแปร tablename หมายถึง ชื่อตารางที่ต้องการดึงข้อมูล

ตัวแปร fieldname1-fieldname-n หมายถึง ชื่อฟิลด์ที่ต้องการดึงข้อมูล ถ้ามีมากกว่า 1 ฟิลด์จะใช้เครื่องหมาย , คั่นระหว่างฟิลด์

ตัวแปร criteria หมายถึง เงื่อนไขในการดึงข้อมูล

4. คำสั่ง UPDATE ใช้สำหรับแก้ไขหรือเปลี่ยนแปลงข้อมูลใน record ที่มีอยู่แล้วในตาราง มีรูปแบบการใช้งาน ดังนี้

UPDATE tablename SET fieldname = value WHERE criteria

ตัวแปร tablename หมายถึง ชื่อตารางที่ต้องการแก้ไขหรือเปลี่ยนแปลงข้อมูล

ตัวแปร fieldname หมายถึง ชื่อฟิลด์ที่ต้องการแก้ไข

ตัวแปร value หมายถึง ค่าที่กำหนดให้กับตัวแปร fieldname

ตัวแปร criteria หมายถึง เงื่อนไขในการแก้ไข

2.6 Web page

2.6.1 Appserv

สำหรับโปรแกรม AppServ นี้ไม่ได้เกิดการสนับสนุนจากหน่วยงานรัฐบาล หรือหน่วยงานเอกชน หรือองค์กรอิสระ ใดๆเลยทั้งสิ้น แต่โปรแกรม AppServ ได้กำเนิดจากแรงบันดาลใจจากเพื่อนของผู้พัฒนาคนหนึ่งที่ได้เริ่มศึกษาภาษา PHP และฐานข้อมูล MySQL และมีปัญหาทุกครั้งในการติดตั้ง กว่าจะติดตั้งได้ก็ใช้เวลาไม่น้อยกว่า 3 ชั่วโมง บางทีทำได้บ้างไม่ได้บ้าง และทุกครั้งที่ติดตั้งไม่ได้ก็จะมาขอความช่วยเหลือจากผู้พัฒนาเป็นประจำทุกครั้ง จึงทำให้ผู้พัฒนาได้สร้างโปรแกรมที่สะดวกในการติดตั้งเพื่อให้เพื่อนของผู้พัฒนาสามารถนำไปใช้งานได้ทันที โดยไม่ต้องมาปวดหัวกับการติดตั้งที่ยุ่งยากอีกต่อไป ในช่วงแรกที่แจกจ่ายนั้น ผู้พัฒนาได้แจกจ่ายในเว็บไซต์ที่เป็นภาษาอังกฤษ ผู้ใช้งานต่างประเทศให้ความสนใจและมีการใช้งานเป็นจำนวนมาก และในปัจจุบันได้เพิ่มเติมในส่วนของเว็บไซต์ภาษาไทย ในอนาคตผู้พัฒนาจะจัดทำเว็บไซต์สามารถรองรับทุกภาษา และเข้าถึงผู้ใช้งานทุกคนทั่วโลก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AppServ คือโปรแกรมที่รวบรวมเอา Open Source Software หลาย ๆ อย่างมารวมกัน โดยมีPackageหลักดังนี้

- Apache
- PHP
- MySQL
- phpMyAdmin

โปรแกรมต่าง ๆ ที่นำมารวบรวมไว้ทั้งหมดนี้ได้ทำการดาวน์โหลดจาก Official Release ทั้งสิ้น โดยเฉพาะตัว AppServ จึงให้ความสำคัญว่าทุกสิ่งทุกอย่างจะต้องให้เหมือนกับต้นฉบับ จึงไม่ได้ตัดทอนหรือเพิ่มเติมอะไรที่แปลกไปกว่า Official Release แต่อย่างใด เพียงแต่มีบางส่วนเท่านั้นที่เราได้เพิ่มประสิทธิภาพการติดตั้งให้สอดคล้องกับการทำงาน โดยที่การเพิ่มประสิทธิภาพนี้ไม่ได้ไปยุ่งในส่วนของ Original Package เลยแม้แต่น้อย เพียงแต่เป็นการกำหนดค่า Config เท่านั้น เช่น Apache ก็จะเป็นในส่วนของ httpd.conf PHP ก็จะเป็นในส่วนของ php.ini, MySQL ก็จะเป็นในส่วนของ my.ini ดังนั้นเราจึงรับประกันได้ว่าโปรแกรม AppServ สามารถทำงานและความเสถียรของระบบ ได้เหมือนกับ Official Release ทั้งหมด

จุดประสงค์หลักของการรวมรวบ Open Source Software เหล่านี้เพื่อทำให้การติดตั้งโปรแกรมต่าง ๆ ที่ได้กล่าวมาให้ง่ายขึ้น เพื่อลดขั้นตอนการติดตั้งที่แสนจะยุ่งยากและใช้เวลานาน โดยผู้ใช้งานเพียงดับเบิ้ลคลิก setup ภายในเวลา 1 นาที ทุกอย่างก็ติดตั้งเสร็จสมบูรณ์ระบบต่าง ๆ ก็พร้อมที่จะทำงานได้ทันทีทั้ง Web Server, Database Server เหตุผลนี้จึงเป็นเหตุผลหลักที่หลาย ๆ คนทั่วโลก ได้เลือกใช้โปรแกรม AppServ แทนการที่จะต้องมาติดตั้งโปรแกรมต่าง ๆ ที่ละส่วนไม่ว่าจะเป็นผู้ที่ความชำนาญในการติดตั้ง Apache, PHP, MySQL ก็ไม่ได้เป็นเรื่องง่ายเสมอไป เนื่องจากการติดตั้งโปรแกรมที่แยกส่วนเหล่านี้ให้มารวมเป็นชิ้นอันเดียวกัน ก็ใช้เวลาค่อนข้างมากพอสมควร แม้แต่ตัวผู้พัฒนา AppServ เอง ก่อนที่จะ Release แต่ละเวอร์ชั่นให้ดาวน์โหลด ต้องใช้ระยะเวลาในการติดตั้งไม่น้อยกว่า 2 ชั่วโมง เพื่อทดสอบความถูกต้องของระบบ ดังนั้นจึงจะเห็นว่าเราเองนั้นเป็นมือใหม่หรือมือเก่า ย่อมไม่ใช่เรื่องง่ายเลยที่จะติดตั้ง Apache, PHP, MySQL ในพริบตาเดียว

มีบางคำถามที่พบบ่อยว่า AppServ สามารถนำไปเป็น Web Server หรือ Database Server ได้ทันทีหรือไม่ ข้อนี้ต้องตอบว่าได้แน่นอน 100% แต่ทางผู้พัฒนาเองขอแนะนำว่า ระบบจัดการ Memory และ CPU บน Windows ที่ทำงานเกี่ยวกับ Web Server หรือ Database Server ไม่เหมาะกับการใช้งานหนัก ๆ เป็นอย่างยิ่ง เพราะ Windows นั้นจะกลืนกินทรัพยากรอันมหาศาล และหากเทียบอัตรารองรับระบบงานกับ OS ตัวอื่น เช่น Linux/Unix จะยิ่งเห็นได้ชัดว่า OS ที่เป็น Windows ที่มีขนาด Memory และ CPU ที่เท่า ๆ กัน OS ที่เป็น Linux/Unix นั้น จะรองรับงานได้น้อยกว่ามาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พอสมควร เช่น Windows รับได้ 1000 คนพร้อมๆ กัน แต่ Linux/Unix อาจรับได้ถึง 5000 พร้อม ๆ กัน หากท่านต้องทำงานหนัก ๆ ทางผู้พัฒนาแนะนำให้เลือกใช้ Linux/Unix OS จึงจะเหมาะสมกว่า

2.6.2 Macromedia Dreamweaver 8

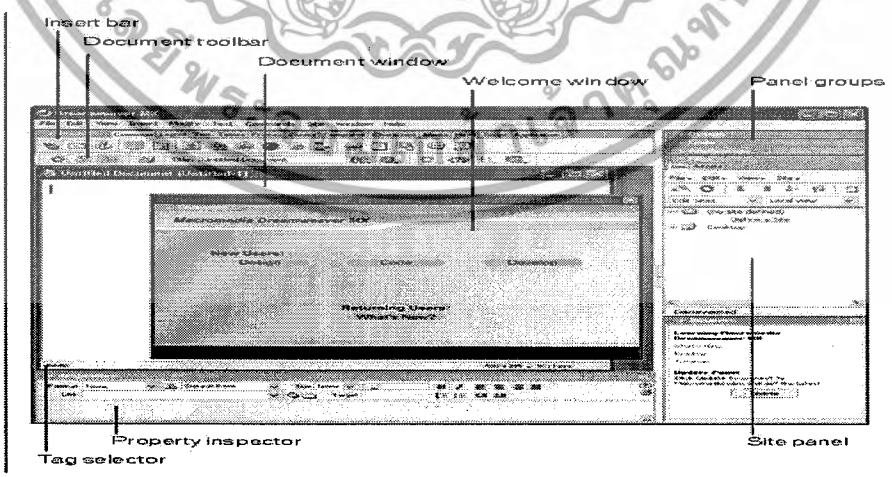
Macromedia Dreamweaver 8 คือโปรแกรมหรือเครื่องมือที่ใช้ในการพัฒนาเว็บไซต์ระดับมืออาชีพ มีความสามารถในการใช้สร้าง ออกแบบ เขียนโค้ด เว็บเพ็จ บริหารจัดการเว็บไซต์ และเว็บแอพพลิเคชั่นได้อย่างมีประสิทธิภาพ ช่วยลดงาน ลดเวลาในการพัฒนาเว็บเพ็จ โดยสามารถสร้างโค้ดได้หลายภาษา เช่น HTML, PHP, ASP, JSP ฯ และสามารถติดต่อกับฐานข้อมูลได้หลายฐานข้อมูล เช่น MySQL, PostgreSQL, MS Access, MS SQL Server ฯ โดยที่ผู้ออกแบบเว็บเพ็จไม่จำเป็นต้องมีความความรู้ด้านภาษาและการจัดการฐานข้อมูล หรือมีความรู้เพียงเล็กน้อยก็สามารถสร้างเว็บเพ็จได้อย่างรวดเร็ว

การสร้างเว็บเพ็จด้วย Dreamweaver 8 เครื่องคอมพิวเตอร์ต้องมีโปรแกรมเหล่านี้เป็นอย่างน้อย

1. Operating System (OS) เช่น Windows, Linux (SQL Database Server)
2. Web Server คือ Apache
3. PHP
4. Dreamweaver MX

2.6.2.1 ส่วนประกอบของ Dreamweaver 8

ส่วนประกอบต่างๆ บนหน้าจอของ Dreamweaver 8 ประกอบด้วย



รูปที่ 2.31 แสดงหน้าต่างของโปรแกรม Dreamweaver 8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. Welcome window จะปรากฏครั้งเดียวจากการที่เรียกใช้งาน Dreamweaver 8 ครั้งแรกหลังจากการติดตั้ง ให้เลือก Design เพราะการสร้างเว็บเพ็จ จะใช้การหน้าจอ Design (ออกแบบ) เป็นหลัก

2. Insert bar ประกอบด้วยไอคอน Object ต่าง ๆ แต่ละ Object จะมีคุณสมบัติเฉพาะ ผู้พัฒนาต้องรู้คุณสมบัติของแต่ละ Object ซึ่งผู้พัฒนาสามารถแทรก Object ต่าง ๆ ลงในเว็บเพ็จ เช่น ข้อความ รูปภาพ ตาราง ฟอร์ม ฯลฯ ซึ่ง Object เหล่านี้ เมื่อแทรกลงในเว็บเพ็จก็คือการแทรกแท็ก HTML ลงไปนั่นเอง ผู้พัฒนาสามารถใช้เมนู Insert แทนการคลิกปุ่ม Object ต่าง ๆ บน Insert bar ได้เช่นเดียวกัน

3. Document toolbar ประกอบด้วยไอคอนที่เกี่ยวข้องกับ Document window เช่น ไอคอน การออกแบบ (Show Design view) การเขียนโค้ด (Show Code view)

4. Document window เป็นพื้นที่ในการสร้างเว็บเพ็จ ประกอบด้วย Show Design view ใช้ในการออกแบบเว็บเพ็จ และ Show Code view ใช้ในการเขียนโค้ด หน้าจอทั้งสองนี้สำคัญมาก เพราะการสร้างเว็บเพ็จจะสร้างจากหน้าจอทั้งสองนี้เป็นหลัก

5. Property inspector ดูหรือแก้ไขคุณสมบัติของ Object บนเว็บเพ็จ

6. Panel groups คือกลุ่มของแถบ (Panel) ซึ่งเป็นกลุ่มที่รวบรวมการทำงานที่เกี่ยวข้องกันไว้ที่เดียวกัน ผู้พัฒนาจะใช้แถบเหล่านี้ในการสร้างเว็บเพ็จ

7. File panel คือ กลุ่มแถบไฟล์ประกอบ Site ที่เก็บเว็บเพ็จต่าง ๆ ที่อยู่ระหว่างการพัฒนา

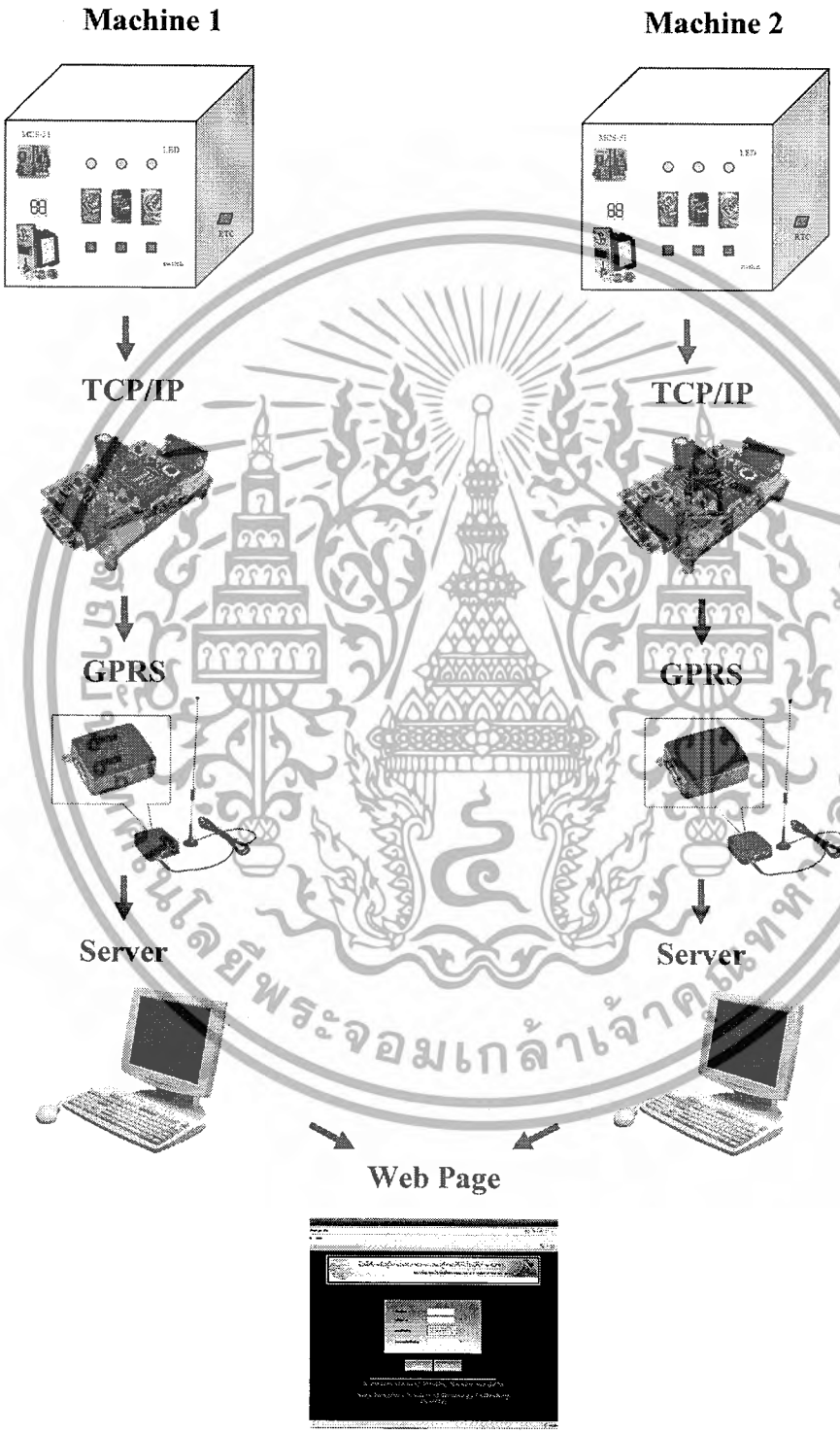
8. Tag selector สำหรับคลิกเพื่อเลือกแท็กในตำแหน่งที่ต้องการบนเว็บเพ็จ

นอกเหนือจากนี้ Dreamweaver 8 ยังมีแถบ (Panel) และเครื่องมืออื่น ๆ ที่ไม่ได้แสดงให้เห็นในหน้าจอนี้ เช่น Design Panel, Code Panel และ Application Panel เป็นต้น แต่สามารถเรียกใช้ได้ผ่านเมนูของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบและทดลองระบบ



รูปที่ 3.1 แสดงการทำงานของระบบทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1 ส่วนของฮาร์ดแวร์ (Hardware)

3.1.1 ไมโครคอนโทรลเลอร์

ในโครงงานนี้จะใช้ไมโครคอนโทรลเลอร์ตระกูล MCS-51 ทั้งสองตัวถูกออกแบบไว้เพื่อควบคุมการทำงานของเครื่องขายสินค้าอัตโนมัติทั้งสอง เพื่อทำการส่งข้อมูลไปยัง TCP/IP Stack ในการส่งข้อมูลจะส่งผ่านทางพอร์ทอนุกรม RS-232 รูปแบบของข้อมูลจะเป็น ASCII code อัตราการรับส่งข้อมูลของทั้งระบบอยู่ที่ 9600 bit/sec

3.1.2 TCP/IP Stack Chip

ใช้เบอร์ TNS010i โดยที่ TCP/IP Stack จะรับคำสั่งซึ่งเป็น AT-command ซึ่งได้ทำการเขียนโปรแกรมไว้ในไมโครคอนโทรลเลอร์และทำการส่งข้อมูลจากตู้ขายน้ำอัตโนมัติและส่งไปยัง TCP/IP Stack ข้อมูลที่ถูกส่งมาจะอยู่ในรูปของ ASCII code มาประมวลผลรูปแบบคำสั่งและทำการส่งต่อไปยัง GPRS Module

3.1.3 GPRS Module

ใช้รุ่น Sony Ericsson gm 29 เป็นอุปกรณ์ในการเชื่อมต่อเครือข่าย GPRS เพื่อทำการส่งค่าที่ได้รับมาจาก TCP/IP Stack ไปยังเครื่อง Server ตาม IP ที่กำหนดและรับค่าจาก Server เพื่อส่งต่อไปยังไมโครคอนโทรลเลอร์เพื่อใช้ในการตรวจสอบความถูกต้องของการส่งข้อมูล

3.2 ส่วนของซอฟท์แวร์ (Software)

3.2.1 KEIL µVision3

เป็นซอฟท์แวร์ที่ใช้ในการเขียนโปรแกรมให้กับไมโครคอนโทรลเลอร์โดยใช้ภาษา C และดาวน์โหลดโปรแกรมที่เขียนขึ้นลง CPU ของไมโครคอนโทรลเลอร์ผ่านทางโปรแกรม Flip ซึ่งมีวิธีการดาวน์โหลดสองแบบคือแบบ Manual ผ่านทางพอร์ทอนุกรม RS-232 และแบบ Auto ผ่านทางพอร์ท ET-Download

3.2.2 Apache

เป็นซอฟแวร์สำหรับสร้าง Server โดยจะให้บริการ Server บนโปรโตคอล HTTP สามารถทำงานได้บนหลายระบบปฏิบัติการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3 My SQL, PHP

My SQL เป็นโปรแกรมที่ใช้ในการจัดเก็บข้อมูล และสร้างระบบฐานข้อมูลที่ถูกส่งมาจากไมโครคอนโทรลเลอร์ผ่านทาง GPRS โดยมีภาษา PHP เป็น API ในการใช้งาน My SQL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

4.1 การออกแบบการทดลอง

1. ระบบที่ใช้ในการทดลองนี้ เป็นการจำลองการส่งข้อมูลเพื่อแสดงการขายสินค้าของตู้ขายน้ำอัตโนมัติ ซึ่งจะใช้ไมโครคอนโทรลเลอร์เป็นอุปกรณ์ส่งข้อมูลออกจากตู้ขายน้ำอัตโนมัติ

2. ระบบจะทำการส่งข้อมูลจากตู้ขายน้ำอัตโนมัติ โดยใช้ไมโครคอนโทรลเลอร์ทำหน้าที่ส่งข้อมูลไปยัง GPRS

3. ทำการเขียนโปรแกรมภาษาซีให้กับไมโครคอนโทรลเลอร์ ให้สามารถควบคุมการทำงานของตู้ขายน้ำอัตโนมัติและส่งข้อมูลเข้าสู่ GPRS Module โดยพอร์ท RS-232 ทำให้ GPRS Module ส่งแพคเกจข้อมูลเข้าสู่ SQL Server โดยเครือข่าย GPRS

4. สร้างเว็บเพจเพื่อแสดงข้อมูลการขายสินค้าของตู้ขายน้ำอัตโนมัติ

4.2 ขั้นตอนการทดลอง

1. ทำการเชื่อมต่ออุปกรณ์ตามที่ได้ออกแบบไว้

2. อัพโหลดเว็บไซท์และระบบฐานข้อมูลขึ้นไปใช้งานจริงบน Internet

3. ไมโครคอนโทรลเลอร์ทำการเปิด TCP/IP Stack และสั่งงานให้เปิดพอร์ท GPRS

4. เมื่อมีการกด Switch เลือกซื้อสินค้าจะทำให้เกิดส่งข้อมูลจากไมโครคอนโทรลเลอร์ในตู้ไปยังไมโครคอนโทรลเลอร์ตัวกลางเพื่อส่งสัญญาณไปยัง GPRS Module และเข้าสู่ Server

5. Server จะทำการรับข้อมูลและแสดงข้อมูลเมื่อมีการเรียกดูจาก Internet

6. การแสดงผลสามารถเรียกดูผ่านทาง Web Browser ได้จากเครือข่ายอินเตอร์เนต

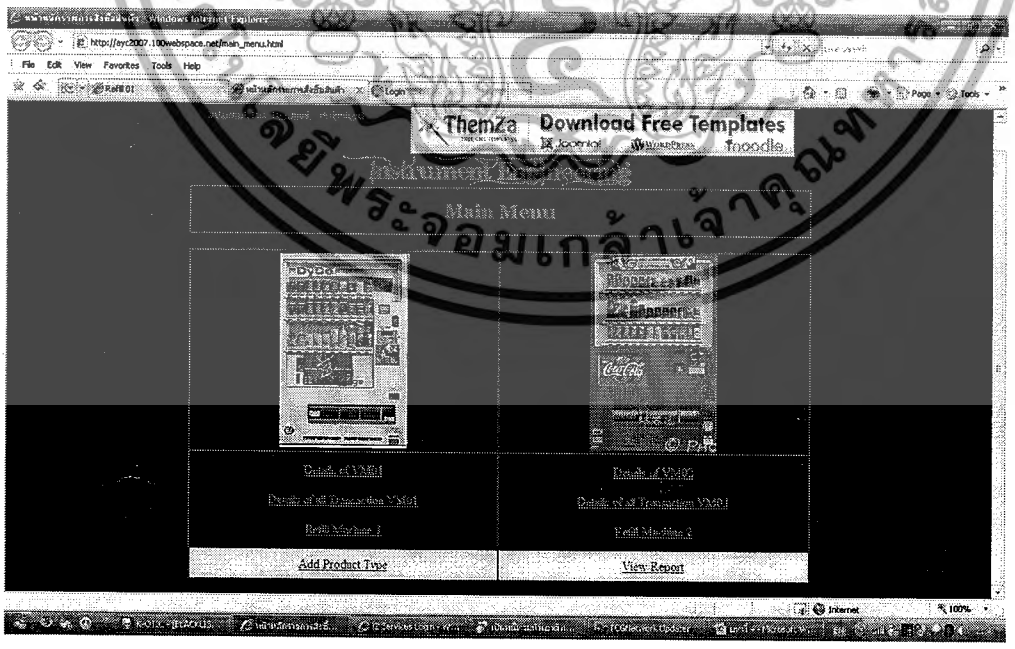
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 บันทึกผลการทดลอง

1. เมื่อเปิด Web Browser ขึ้นมา จะมีหน้า Login เพื่อรักษาความปลอดภัยแสดงผลดังรูป

รูปที่ 4.1 แสดงหน้า Login เข้าสู่หน้าหลัก

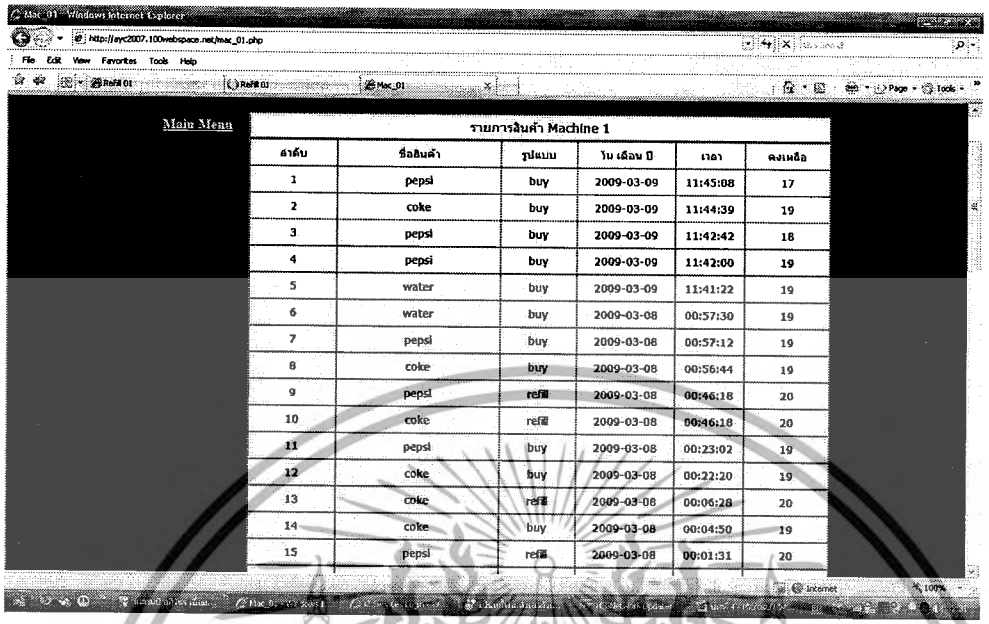
2. เมื่อทำการ Login แล้วจะทำการเข้าสู่หน้าหลักเพื่อเลือกดูข้อมูลที่ต้องการ โดยจะมีให้เลือกดูได้สองแบบคือ 50 ข้อมูลล่าสุดและข้อมูลทั้งหมดของแต่ละเครื่อง



รูปที่ 4.2 แสดงหน้าหลักเพื่อเลือกดูข้อมูลของแต่ละเครื่อง

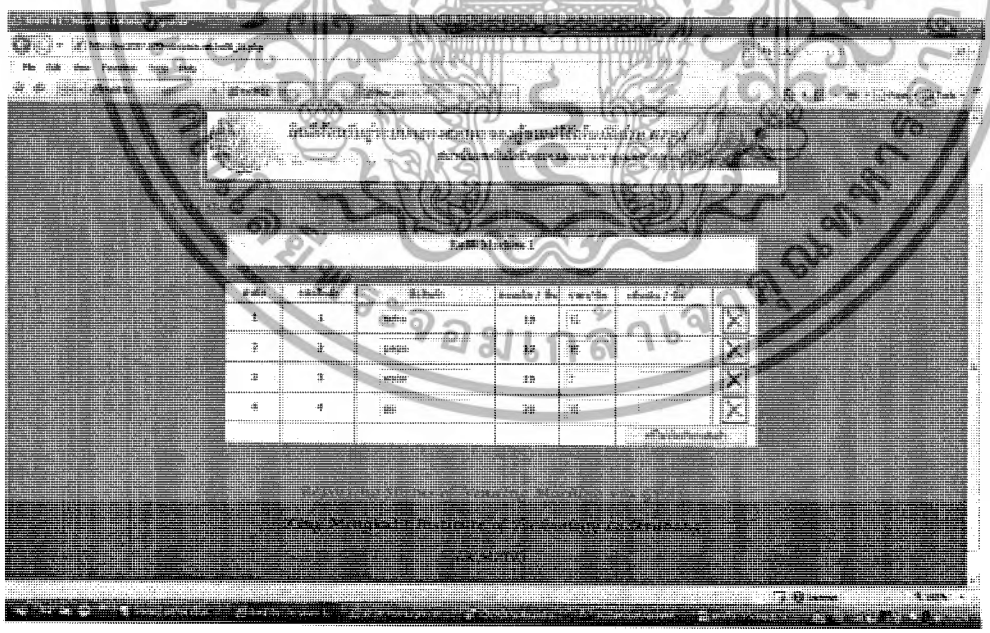
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เว็บเพจแสดงข้อมูลของแต่ละตู้ ซึ่งบอกถึงรายละเอียดของการขายสินค้าของตู้นั้น ๆ



รูปที่ 4.3 หน้าแสดงผลข้อมูลการซื้อสินค้า

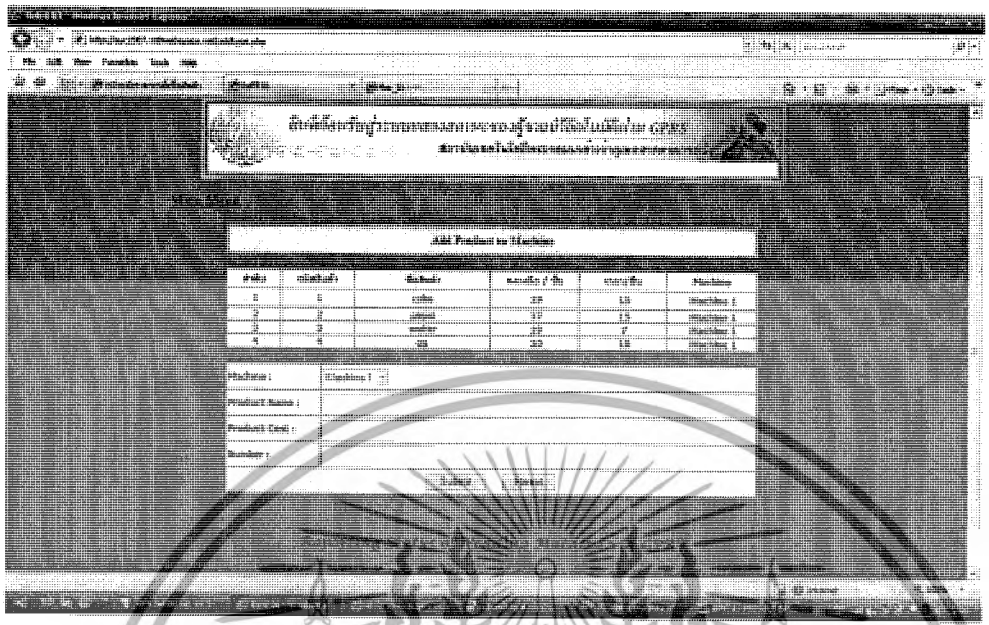
4. เป็นหน้าสำหรับแก้ไขข้อมูลหลังจากมีการเติมสินค้าและข้อมูลทั่วไปของสินค้า



รูปที่ 4.4 หน้าเว็บเพจการเติมสินค้าและแก้ไขข้อมูลสินค้า

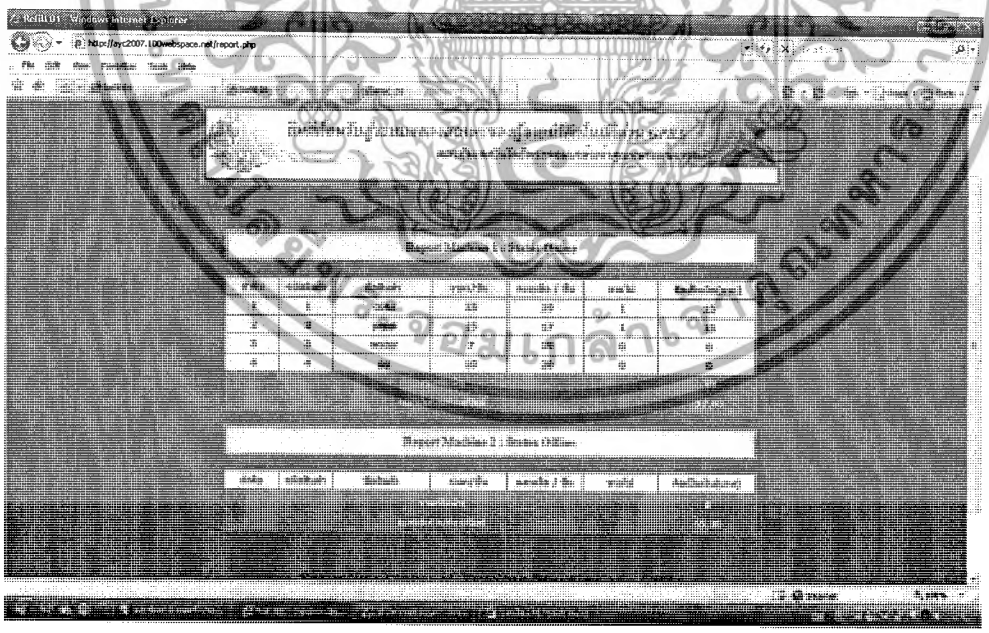
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. หน้าสำหรับการเพิ่มข้อมูลสินค้า



รูปที่ 4.5 หน้าเพิ่มสินค้า

6. หน้ารายงานการซื้อขายสินค้า



รูปที่ 4.6 หน้ารายงานการซื้อขายสินค้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 สรุปผลการทดลอง

1. จากการทดลอง GPRS Module สามารถรับข้อมูลจากไมโครคอนโทรลเลอร์แล้วส่งค่าไปบันทึกลงฐานข้อมูลที่ติดตั้งบน Server

2. Web Browser สามารถแสดงผลข้อมูลได้และระบบฐานข้อมูลสามารถทำการ Reset ยอดคงเหลือให้เต็มเช่นเดิมได้ หลังการเติมสินค้า

3. สามารถเพิ่มและแก้ไขข้อมูลสินค้าได้ทางหน้าเว็บเพจ และมีการแสดงยอดขายสินค้าของแต่ละตู้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลการวิจัยและข้อเสนอแนะ

5.1 สรุป

จากการทดลอง สามารถสร้างชุดจำลองเครื่องขายน้ำอัตโนมัติโดยนำเอาไมโครคอลโทรลเลอร์เข้ามาควบคุมการทำงาน ซึ่งมีหลักการทำงานคือ เมื่อผู้ใช้บริการเครื่องขายน้ำอัตโนมัติทำการซื้อสินค้า ไมโครคอลโทรลเลอร์จะทำการรายงานสถานการณ์การซื้อสินค้าผ่านทางอินเตอร์เนต โดยในขั้นตอนนี้ไมโครคอลโทรลเลอร์จะส่งค่าผ่านทาง GPRS Module ไปยังฐานข้อมูลในเครื่องเซิฟเวอร์ และเวบเพจก็จะทำการดึงค่าในฐานข้อมูลขึ้นมาแสดงผลทางหน้าเวบ

5.2 ปัญหาทางเทคนิค

1. เรื่องของสัญญาน GPRS ที่บางครั้งไม่สามารถเชื่อมต่อได้เนื่องจากความหนาแน่นของเครือข่าย GPRS ณ เวลานั้น

2. สายสัญญาณต่าง ๆ อาจเกิดการเสียหายภายใน การเข้าหัวสายที่ไม่มีความหนาแน่นทำให้เกิดปัญหาการเชื่อมต่อสื่อสาร

3. อุปกรณ์บางรุ่นเป็นของที่ยังมีการรับประกันสินค้าอยู่ ถ้าเสียก็ยังสามารถนำไปซ่อมได้ แต่ยังมีปัจจัยทางด้านระยะเวลา ที่ศูนย์แต่ละศูนย์นั้นมีระยะเวลาการซ่อมแซมที่มากน้อยไม่เท่ากัน ถ้าระยะเวลาที่ต้องทำการซ่อมใช้เวลาทำให้เกิดปัญหาทั้งระบบได้

4. เนื่องจากไม่สามารถทำการส่งชุดคำสั่ง AT-command ให้ทำการติดต่อกับ TCP/IP Stack เพื่อส่งข้อมูลให้กับ GPRS Module ได้โดยตรง

5.3 แนวทางการพัฒนาต่อ

1. พัฒนาตัวเครื่องขายสินค้าให้สามารถรับเงินที่เป็นธนบัตร และสามารถทำการทอนเงินที่เหลือได้

2. เขียนชุดคำสั่ง AT-command ให้ทำการติดต่อกับ TCP/IP Stack โดยตรงได้

3. พัฒนาให้หน้าเวบเพจมีโปรแกรมประมวลผลทางการตลาดแบบสำเร็จรูป

4. หาแนวทางในการลดต้นทุนของตัวเครื่องขายสินค้าอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

- ธีรวัฒน์ ประกอบผล. การประยุกต์ใช้งานไมโครคอนโทรลเลอร์. กรุงเทพฯ : สมาคมส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น), 2540
- ดอนสัน ปงผาบ. การเขียนโปรแกรมภาษาซีในการควบคุม. กรุงเทพฯ : สมาคมส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น), 2548
- ดวงพร เกี๋ยงคำ , วงศ์ประชา จัทร์สมวงศ์. คู่มือสร้างเว็บไซค์ด้วยตนเอง ฉบับสมบูรณ์. กรุงเทพฯ: โปรวิชั่น , 2547
- วรรณวิภา ติตถะสิริ คู่มือเรียน SQL ด้วยตนเอง. กรุงเทพฯ ; โปรวิชั่น , 2545
- พร้อมเลิศ หล่อวิจิตร. คู่มือการเรียน PHP และ SQL สำหรับผู้เริ่มต้น, กรุงเทพฯ ; โปรวิชั่น , 2550
- สงกรานต์ ทองสว่าง. My SQL ระบบฐานข้อมูล สำหรับอินเตอร์เนต, สำนักพิมพ์ซีเอ็ด ยูเคชั่น, กรุงเทพฯ
- http://www.ett.co.th
- http://www.keil.com
- http://www.flip.com
- http://www.wara.co.th
- http://www.tcpipchip.com
- http://www.mysql.com
- http://www.thaiwbi.com

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

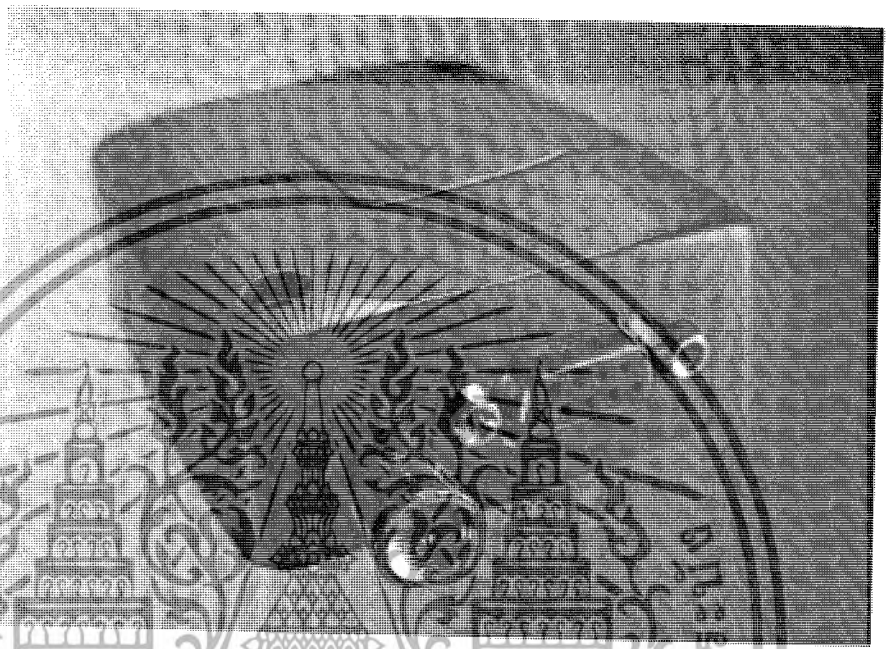
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Sony Ericsson

GM28/GM29 GSM/GPRS Terminal

Features:

- Self Contained
- 900+1800 850+1900 Dual Band
- GSM/GPRS Class 8
- Integrated TCP/IP Stack
- GSM Ph2+ FCC/PTCRB Fully Type Approved



Benefits

- Short time to market due to minimal hardware design
- Wireless GPRS Internet access
- Reduce cost of testing business concepts

The GM28/GM29 GSM/GPRS Terminal is a powerful, compact and self-contained unit designed for system integrators to quickly test application and product concepts based on or using GSM/GPRS communications. With standard connector interfaces and an integral SIM card reader the GM28/GM29 offers a ready and cost-effective GSM communications solution for integration into a range of business, commercial and industrial applications – requiring minimal modification to the existing hardware and software platform to which it is being integrated.

The GM28/GM29 is a natural fit for integrators and application service providers to evolve to GPRS from legacy non-universal wireless data communication standards like CDPD and Mobitex, with minimal re-design burden. The GM28/GM29 can be used as a solid alternative to a second land-based phone line for deal-up backup communications, e.g. for POS terminals. When communicating data through GPRS, the GM28/GM29 will provide the customer application a data rate comparable to that of high-speed dial-up modems. In addition, with its high data rate the GM28/GM29 is an ideal alternative to dial-up backup lines for broadband and dedicated data lines.

Applications can include the monitoring and control of any electronically controlled system, industrial processes, electronic payment terminals and meter readings in areas such as gas, water, electricity and power networks. They also extend into service and maintenance in areas such as lifts, escalators, vending machines and point-of-sale equipment.

m2m
Expanding Wireless Possibilities

Expanding Wireless Possibilities

สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

GM28/GM29 GSM/GPRS Terminal

Product Features

- Dual Band GSM/GPRS Terminal (EGSM 850/1900MHz [GM28] or GSM 900/1800MHz [GM29])
- Compliant with 3GPP GSM Phase 2+ (GCF-CC v3.8.1 [GM29]/NAPRD v.2.7.2 [GM28] standard) (Small MS)
- Output power:
  - Class 4 (2W) @ 850/900MHz
  - Class 1 (1W) @ 1800/1900MHz
- Power supply:
  - 5–32VDC
- Operating Temperature:
  - -30°C to +75°C
- Storage Temperature:
  - -30°C to +85°C
- Full Specification:
  - -10°C to +55°C
- Overall dimensions (excluding connectors): 77 x 67 x 26mm
- Weight: < 110g

Interfaces

- Data: RS232 9–way (V.28)
- Power and Extended I/Os: 5 – 32VDC (RJ11)
- Audio 4-wire Handset Interface (RJ9)
- Antenna: 50Ω (FME male)
- SIM card reader: 3V/5V interface with SIM detection

Data Features

- CSD asynchronous – non transparent up to 9.6kbps
- HSCSD (2+1), up to 19.2kbps
- MS Class B, GPRS Class 8, up to 85.6kbps
  - P Channel Support
  - Coding schemes CS1 – CS4
- GSM 07.10 Multiplexing Protocol

Short Message Service (SMS) Features

- Text and PDU
- MT/MO & CBM
- Cell Broadcast
- Concatenation – up to 6 SMS

Voice Features

- Telephony
- Emergency calls
- Full Rate, Enhanced Full Rate and Half Rate (FR/EFR/HR)
- Dual Tone Multi Frequency function (DTMF)
- GM28 supports AMR

Fax Features

- Fax Group 3, Class 1 & 2[†]

TCP/IP

- AT Command access to TCP/IP stack
- Universal Datagram Protocol (UDP)

GSM Supplementary Services

- Call forwarding
- Call hold, waiting and multiparty
- Calling/called number identification
- Advice of charge
- USSD
- Alternate line service
- Customer service profile
- Preferred networks
- Operator selection
- Network registration
- Call barring
- Call transfer

Other Features

- ME + SIM phone book management – read/write/find, call screening, groups
- SIM Application Toolkit Class 2
- Real Time Clock
- Software upgradeable
- Audio control
- Fixed dialling number
- UCS2 16 bit data supported
- Commercially available accessories

[†] US Carriers may require compliance with AMR or EOTD standards

Sony Ericsson

Email: m2minfo@SonyEricsson.com
Website: www.SonyEricsson.com/m2m

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

© 2005 Sony Ericsson Mobile Communications AB.
All rights reserved. The marble logo is a trademark of Sony Ericsson Mobile Communications AB.
Sony is a trademark of Sony Corporation. Ericsson is a trademark of Telefonaktiebolaget LM Ericsson.

Subject to change without prior notice.
LZT 123 7287 R1G

Expanding Wireless Possibilities

DPY 102 230 (GM28) DPY 102 229 (GM29)

TNS010i TCP/IP Stack Chip

PRELIMINARY

Data Sheet and User Manual

Rev. 010-17-04-2004

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

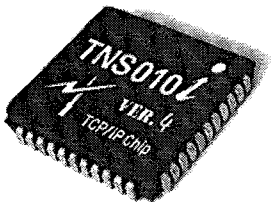
Checkout:

http://www.tcpipchip.com

for details and updates

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TNS010i *TCP/IP* stack Chip



Description

TNS010i TCP/IP stack chip is a complete single turnkey product that enables the Internet connection. Not only does it reduce development time to market, the TNS010i allows several real-time efficient transmissions of data and commands between an 8/16/32-bit micro controller (MCU) and the core TCP/IP software engine. With no priori information about the Internet protocol or networking, this device allows the MCU to send and receive web pages, data, and commands in a form of scripting phrases through its built in TCP/IP protocol engine.

Executing only a few simple commands from an MCU through serial pins of TNS010i and you will be able to transmit and receive data to/from web application program. Each command sent across the TNS010i will invoke TNS010i to response back to an MCU with a message. This debugging information helps users keep track of every step during the connection attempt.

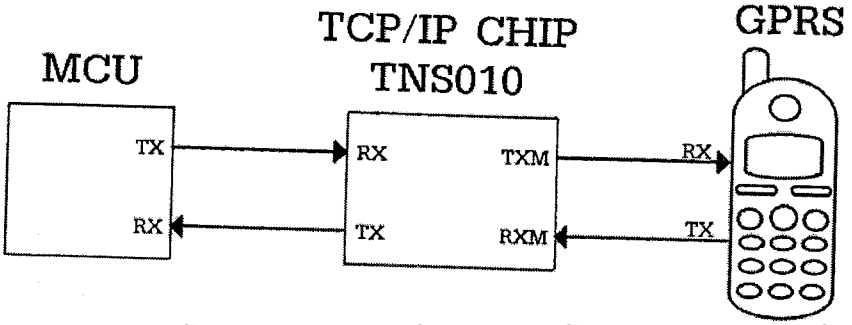
This device can connect to the GPRS phone directly with no additional components. Even the MCU needs to set-up or read out the GPRS parameters, this device automatically bypasses those AT commands to the phone thus eliminating

- The need for two serial-port hardware on the MCU side. and
- A complex circuit for switching among the three modules: MCU, TNS010i, and GPRS.

Features

- Support basic protocols including TCP/IP, PPP, DNS ,HTTP,SMTP,POP3 and TCP/IP socket.
- Easy interface to GPRS phone or Modem.
- Input commands length as long as 1,000 bytes.
- No need for external RAM or ROM.
- Minimal components. Requires only one 18.432 MHz Xtal, and 3 Capacitors .
- 19,200 BPS Command communicates with micro controller (MCU)
- Information exchanges rate with GPRS is variable from 1,200 BPS to 57,600 BPS
- low power consumption
- 44-pin PLCC and LQFP package

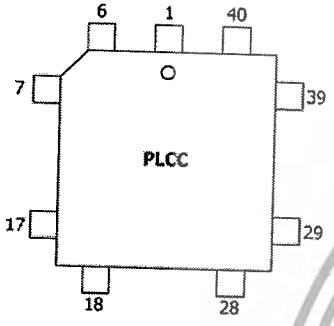
The Connection Diagram

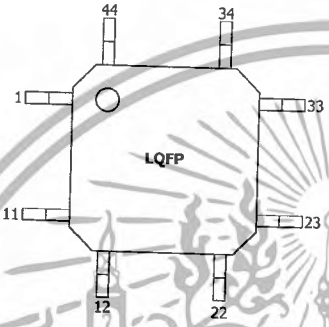


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Electrical Characteristics

Symbol	Description	Min	Type	Max	Unit
VIL	Input Low Voltage	-0.5		0.9	V.
VIH	Input High Voltage	1.9		5.5	V.
VIH_	Input High Voltage XTAL1,RST	3.5		5.5	V.
Icc	Power supply current			60	mA
Vcc	Power supply voltage	4.75		5.25	V.





Pin Description

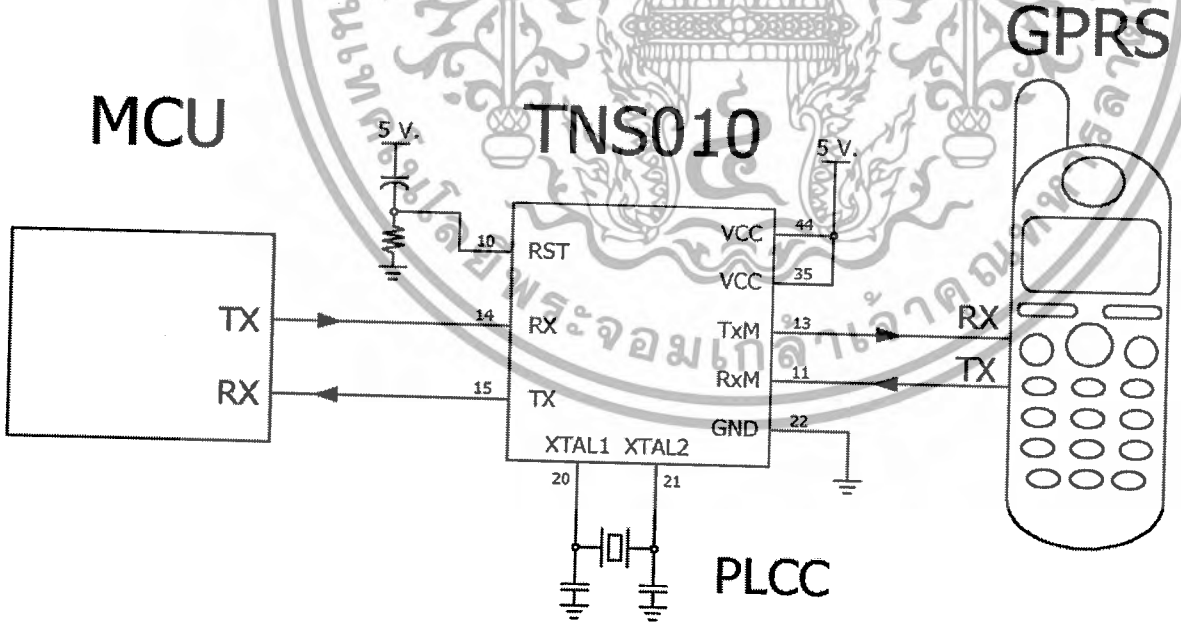
Symbol	PLCC	LQFP	In/Out	Name and Function
Vcc	44,35	38,29	I	Power Supply : power supply voltage for operation
Gnd	22	16	I	Ground : 0 V. reference
XTAL1	21	15	I	Crystal 1: 18.432 MHz. Input oscillator amplifier.
XTAL2	20	14	O	Crystal 2: 18.432 MHz. Output oscillator amplifier.
GPRS/modem	18	12	I	GPRS / Land line modem time out select
CD	2	40	I	Carrier detect
DTR	3	41	O	Data terminal ready
DSR	4	42	I	Data set ready
RTS	5	43	O	Request to send
CTS	6	44	I	Clear to send
RST	10	4	I	Reset : hold high at least 0.22 mS
TxM	13	7	O	Transmit to modem/GPRS phone
RxM	11	5	I	Receive from modem/GPRS phone
Tx	15	9	O	Transmit to controller
Rx	14	8	I	Receive from controller
LED-Connect	9	3	O	LED status shows connected to net
LED-Data trans	19	13	O	LED status shows data transferring

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TCP/IP Software Specifications

Maximum at+i command length (include at+i itself)	1000 bytes
Maximum TCP segment size	512 bytes
Serial RX buffer length (Ring buffer)	2,500 bytes
Serial TX buffer length	1,000 bytes
PAP	Support
CHAP	Not support
DNS	Support
HTTP	Support
FTP	Not support
SMTP	Support
POP3	Support
TCP/IP socket	Support

Typical testing circuit



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Summary of Commands

- 19,200 BPS command baud rate (MCU (controller) ←→ TCP/IP Chip) N-8-1
- 1,200-57,600 BPS GPRS/modem baud rate. (TCP/IP Chip ←→ GPRS/Modem) N-8-1
- Dark printing characters are main commands and the gray are option commands

Chip test

Command: at+i<CR>
Response: **+I_OK**

Open a PPP connection

Command: at+iopen<CR>
Response:
- Success **+I_OK**

- Not success
+I_ERROR 1 PPP_NOT_ESTABLISH *(can't connect to PPP server)*
+I_ERROR 3 DIAL_NOT_SET *(no dial number set up)*
+I_ERROR 4 MODEM_NOT_RESPONSE *(no modem response)*
+I_ERROR 7 PASSWORD_NOT_GOOD *(invalid password)*

Close a PPP connection

Command: at+iclose<CR>
Response:
- Success **+I_OK**

- Not success
+I_ERROR 4 MODEM_NOT_RESPONSE *(can't disconnect modem)*

Display a current assigned IP address

Command: at+iip<CR>
Response: **CLIENT_IP=x.x.x.x**
SERVER_IP=x.x.x.x
+I_OK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Set up a dialing number

Command:	at+isetd=< ISP dialing number><CR>
Response:	**+I_OK**
Example:	at+isetd=*99***1#<CR> **+I_OK**

Display a dialing number

Command:	at+isetd=?<CR>
Response:	**<DIAL NUMBER>** **+I_OK**
Example:	at+isetd=? <CR> ***99**1#** **+I_OK**

Set up DNS server's IP address (in case of ISP not automatic DNS assigned)

Command:	at+isetdns=<ip address><CR>
Response:	**+I_OK**
Example:	at+isetdns=203.155.33.1<CR> **+I_OK**

Display DNS server's IP address

Command:	at+isetdns=?<CR>
Response:	**<DNS SERVER IP ADDRESS>** **+I_OK**
Example:	at+isetdns=? <CR> **203.155.33.1** **+I_OK**

Set up PPP user name

Command:	at+isetuser=<username><CR>
Response:	**+I_OK**
Example:	at+isetuser=David<CR> **+I_OK**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Display PPP user name

Command: at+isetuser=?<CR>
Response: **<USER NAME>**
+I_OK

Example: at+isetuser=? <CR>
David
+I_OK

Set up PPP password

Command: at+isetpass=<password><CR>
Response: **+I_OK**

Example: at+isetpass=David password<CR>
+I_OK

Display PPP password

Command: at+isetpass=?<CR>
Response: **<PASSWORD>**
+I_OK

Example: at+isetpass=?<CR>
David password
+I_OK

HTTP Request

Command:
at+ihttp://<url>[<path>][:<port>][<space></h></l></d>]<CR>
Response:
- Success **+I_OK**
<WEB PAGE CONTENT.....
.....
.....
.....>

- Not success
+I_ERROR 2 DNS_IP_NOT_SET
(If specify domain name, must set a DNS IP address first : at+isetdns)
+I_ERROR 5 TCP_TIMEOUT *(server is busy)*
+I_ERROR 0 PPP_CLOSED *(PPP is not established yet, must open with at+iopen)*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Note:
1. /h = display all **header** field.
2. /l = display only **length** field
3. /d = display only **date** field
4. default port number is 80

Example:

```
at+ihttp://www.google.com /l<CR>
+I_OK
Content-Length:2998
<html><head><meta http-equiv="content-type"
content="text/html; charset=windows-
874"><title>Google</title><style><!--
body,td,a,p,.h{font-family:;}
.h{font-size: 20px;}
.q{text-decoration:none; color:#0000cc;}
.....................
.....................
</html>
```

Example:

```
at+ihttp://www.myserver.com:123/doc.htm/d/l<CR>
+I_OK
Date: Mon, 29 Dec 2003 12:29:09 GMT
Content-Length: 3203
<HTML>.....................
.....................
.....................
</HTML>
```

How to request the next HTTP ?

After successfully retrieving the first web page, the successive request needs no further initial set-up commands. Just send

```
Command:
at+ihttp://<url>[<path>][:<port>][<space></h></l></d>]<CR>
```

In case a connection is broken

1. Close the existing connection first with

```
Command:    at+iclose<CR>
```

2. Then re-open PPP connection with

```
Command:    at+iopen<CR>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. Followed by requesting a web page

```
Command:
at+ihttp://<url>[<path>][:<port>][<space></h></l></d>]<CR>
```

TCP/IP SOCKET

1.OPEN TCP/IP SOCKET

```
Command:    at+isockopen=<url>:<port><CR>
Response:   +I_OK
            <RESPONSE/NONE>
            Note: <port> is require , no default value

Example:    at+sockopen=mail.server.com:25<CR>
            +I_OK
            220 SMTP Service ready
```

2.SEND DATA ON TCP/IP SOCKET

```
Command:    at+isock=<data><CR>
Response:   +I_OK
            <RESPONSE/NONE>

Example:    at+sock= HELO\r\n<CR>
            +I_OK
            250 HELLO <CR>
```

3.CLOSE TCP/IP SOCKET

```
Command:    at+isockclose<CR>
Response:   +I_OK
```

EMAIL SEND/RECEIVE (SMTP & POP3)

1.OPEN -> SEND -> CLOSE Mail

```
Command:
at+ismtpsend=<mailserver>[:<port>],<from>,<to>,<subject>,<data><CR>
Response:   +I_OK
             or
            +I+ERROR 10 MAIL_ERROR
```

Note:

- <port> default value is 25.

Example:

```
at+ismtpsend=mail.server.com,sender@aaa.com,
recv@bbb.com,My subject,Hello how are you?<CR>
+I_OK
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.READ EMAIL(POP3)

2.1 OPEN -> CHECK MAIL

Command:
at+ipop3open=<mailserver>[:<port>],<user_name>,<password><CR>

Response: **< mail number list >**
+I_OK
or
+I+ERROR 10 MAIL_ERROR

Note:
- <port> default value is 110.

Example:

at+ipop3open=pop3.server.com,david,1234<CR>
1,2,3,4,5
+I_OK

2.2 READ EMAIL

Command:
at+ipop3read=<mail_number><CR>

Response: **< mail content >**
+I_OK
or
+I+ERROR 10 MAIL_ERROR

Example:

at+ipop3read=4<CR>
From: sender@aaa.com
Subject: Test

Hello , How are you?
+I_OK

2.3 CHECK MAIL

Command: at+ipop3check<CR>

Response: **< mail number list >**
+I_OK
or
+I+ERROR 10 MAIL_ERROR

Example:

at+ipop3check<CR>
1,2,3,4,5
+I_OK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 DELETE MAIL

Command: at+ipop3delete=<mail number><CR>

Response: **< mail number >**
+I_OK
or
+I+ERROR 10 MAIL_ERROR

Note: mail number

"all" = delete all messages
num1, num2, num3 = delete specific messages
num1-num2 = delete range of messages

Example:

at+ipop3delete=1,2,6<CR>
3,4,5 **//display undelete message**
+I_OK

at+ipop3delete=all<CR>

+I_OK

at+ipop3delete=4-10<CR>
1,2,3 **//display undelete message**
+I_OK

2.5 CLOSE READING EMAIL SERVER

Command: at+ipop3close<CR>

Response: **+I_OK**
or
+I+ERROR 10 MAIL_ERROR

Read software version

Command: at+iver<CR>

Response: **VERx.xx**
+I_OK

Example: at+iver <CR>
VER0.07
+I_OK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Bypass to modem's AT commands mode

1st Step :
type **+++** and **<cr>** for entering modem's AT commands mode

Command:	+++<CR>
Response:	**+I_OK**

2nd step :

Case1 : modem is not connecting to internet
Type **at<cr>** to initiate and enter modem's commands mode
OK → response from modem

Case2 : modem is connecting to internet
Type +++ and wait for 2-3 sec. and then type
at<cr> to initiate and enter modem's commands mode
OK → response from modem

From this state you can communicate to modem with it's AT Commands

To quit from modem's AT commands mode and enter TNS010i commands mode

Case1 : modem is not connecting to internet
Type **at+i<cr>** to enter TNS010i's commands mode
+I_OK → response from TNS010i

Case2 : modem is connecting to internet
Type **ato** to enter modem's internet connecting state
CONNECT → response from modem
and then type
at+i<cr> to enter TNS010i's commands mode
+I_OK → the response from TNS010i

Turn on DEBUG mode

Debug mode on = print out some more information when chip is in Processing.

Command:	at+idebug1<CR>
Response:	**+I_OK**

Turn off DEBUG mode

Command:	at+idebug0<CR>
Response:	**+I_OK**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Set up TCP time out

Command: at+isettcpto=<TCP time out value ><CR>
Response: **+I_OK**

Example: at+isettcpto=8<CR> (1-255 Sec.)
+I_OK

Display TCP time out value

Command: at+isettcpto=?<CR>
Response: **<Time-out value>**
+I_OK

Example: at+isettcpto=?<CR>
8
+I_OK

Set up DNS time out

Command: at+isetdnsto=<DNS time out value ><CR>
Response: **+I_OK**

Example: at+isetdnsto=8<CR> (1-255 Sec.)
+I_OK

Display DNS time out value

Command: at+isetdnsto=?<CR>
Response: **<Time-out value>**
+I_OK

Example: at+isetdnsto=?<CR>
8
+I_OK

Set up Modem time out

Command: at+isetmodemto=<time out value ><CR>
Response: **+I_OK**
Example: at+isetmodemto=8<CR> (1-255 Sec.)
+I_OK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Display Modem time out

Command:	at+isetmodemto=?<CR>
Response:	**<Time-out value>**
	+I_OK

Example:

at+isetmodemto=?<CR>
8
+I_OK

Detect and automatic set up GPRS/modem interfacing baud rate.

Command:	at+idetect<CR>
Response:	**xxxx**
	+I_OK

Example:

at+idetect<CR>
9600
+I_OK

Manual set up GPRS/modem interfacing baud rate

Command:	at+isetbaud=<baud rate><CR>
Response:	**+I_OK**

Baud rate value list:

576 = 57600 BPS.(default)
384 = 38400
192 = 19200
096 = 9600
048 = 4800
024 = 2400
012 = 1200

Example:

at+isetbaud=096<CR>
+I_OK

Display GPRS/modem interfacing baud rate

Command:	at+isetbaud=?<CR>
Response:	**<baud rate value>**
	+I_OK

Example:

at+isetbaud=?<CR>
9600
+I_OK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Data counter

Check amount of total data transfer

Command:	at+icount=?<CR>
Response:	**<data transfer counter>** Bytes **+I_OK**
Example:	at+isetbaud=?<CR> **1371** **+I_OK**

Reset data transfer counter

Command:	at+icount=0<CR>
Response:	**+I_OK**
Example:	at+isetbaud=0<CR> **+I_OK**

note: the data counter command available only in ver 4.03 up

Modem's flow Control

Setting Modem's flow contrl

Command:	at+icom1ctrl=hard/none<CR>
Response:	**+I_OK**
Example:	at+icom1ctrl=none<CR> **+I_OK**

Read Modem's flow control status

Command:	at+icom1ctrl=?<CR>
Response:	**HARD/NONE** **+I_OK**
Example:	at+icom1ctrl=?<CR> **NONE** **+I_OK**

note : default = none
Dev-010i-B's modem connection is null modem type

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

GRPS phone / Modem selection

When power on reset. TNS010i will load a modem time out default value according to the logic level below.

GPRS sel. = high(or no connect)
LAND LINE MODEM sel. = low (connect to GND.)

Note:
GPRS modem time out default value = 10 Sec.
Land line modem time out default value = 60 Sec.

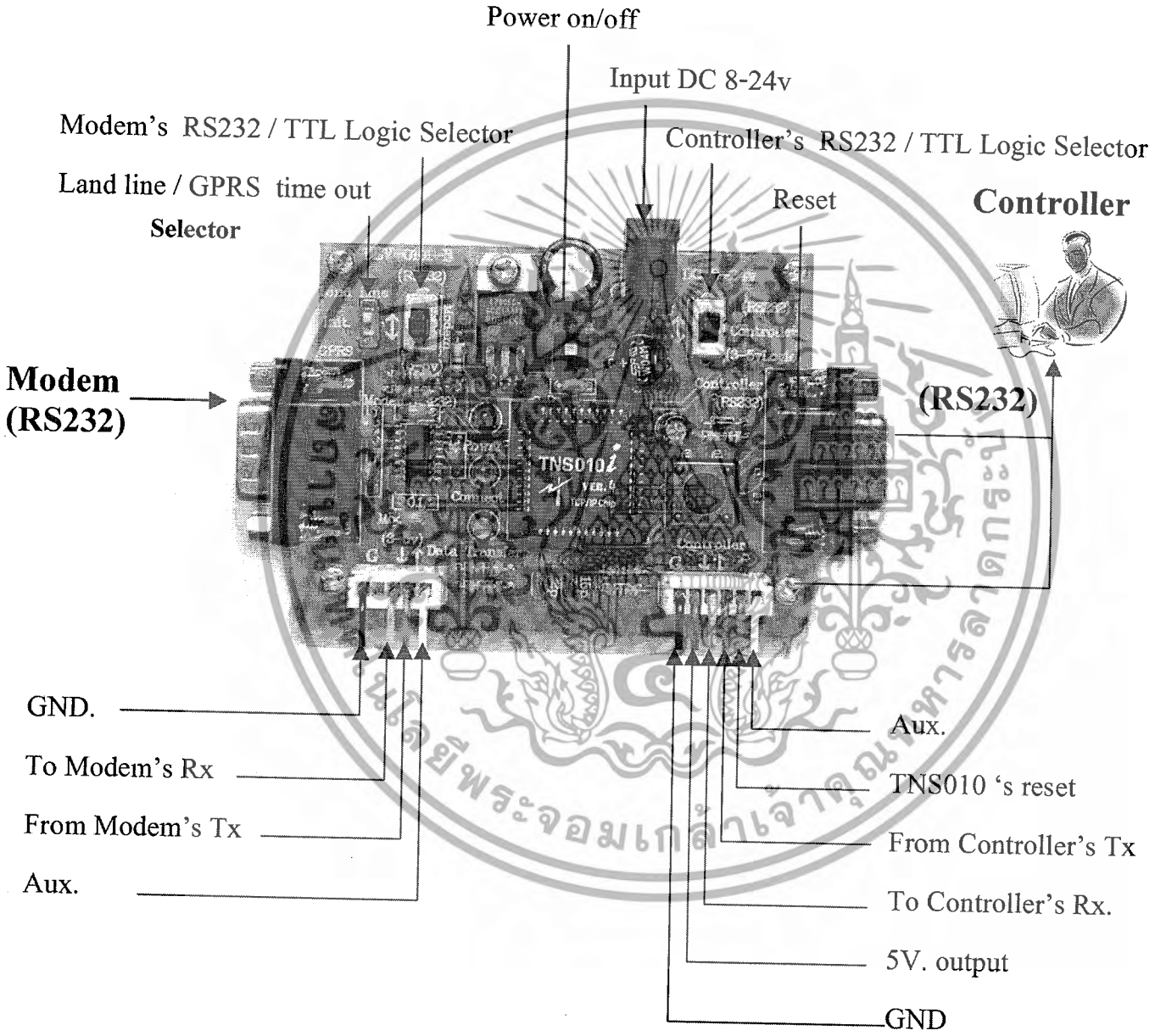
you can change these default value by issue
at+isetmodemto=xxx

Error message response

```
+I_ERROR 0 PPP_CLOSED
+I_ERROR 1 PPP_NOT_ESTABLISH
+I_ERROR 2 DNS_IP_NOT_SET
+I_ERROR 3 DIAL_NOT_SET
+I_ERROR 4 MODEM_NOT_RESPONSE
+I_ERROR 5 TCP_TIME_OUT
+I_ERROR 6 DNS_TIME_OUT
+I_ERROR 7 PASSWORD_NOT_GOOD
+I_ERROR 8 CONNECT_LOST
```

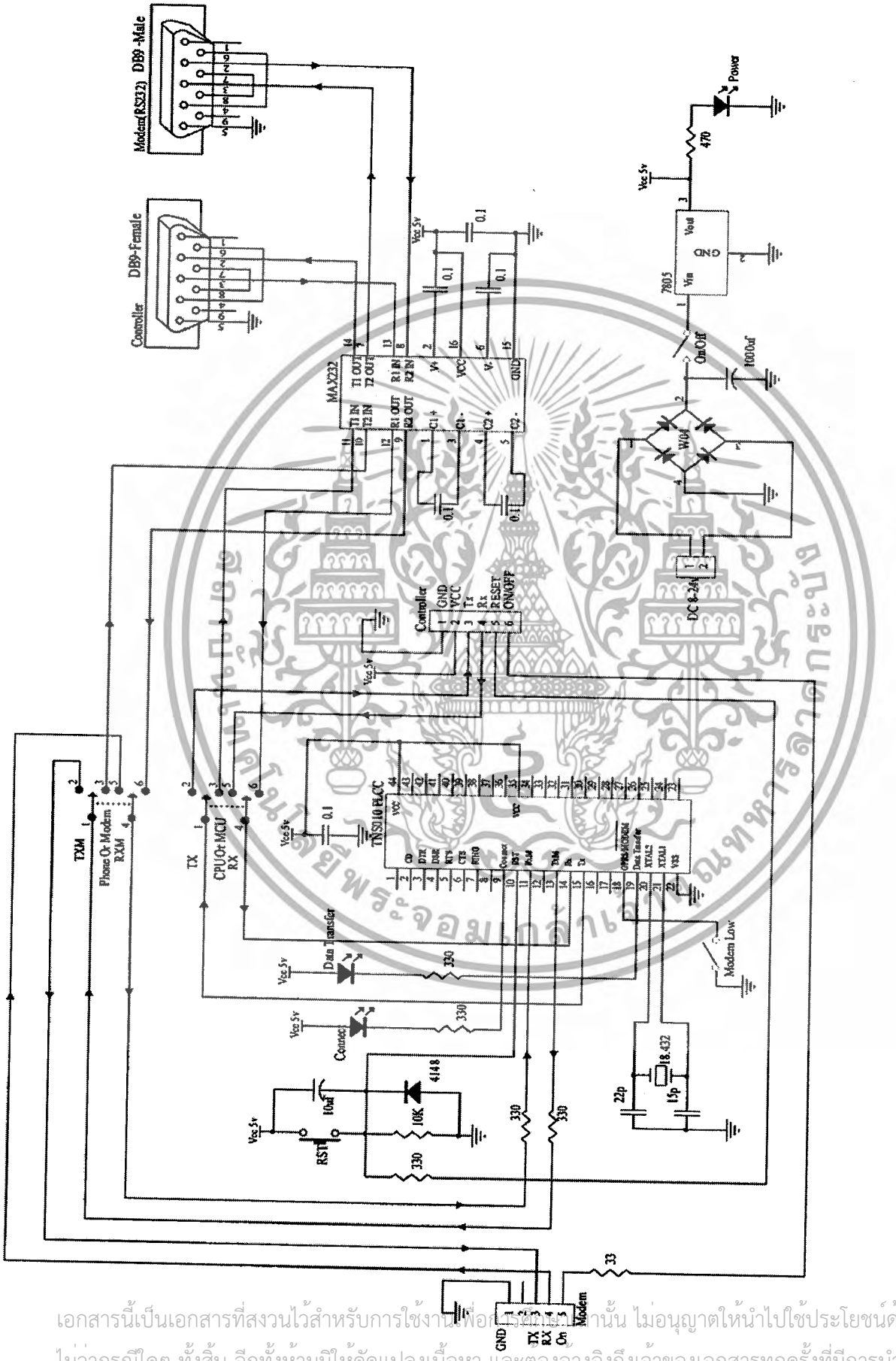
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Development board DEV-010i-B wiring guide



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

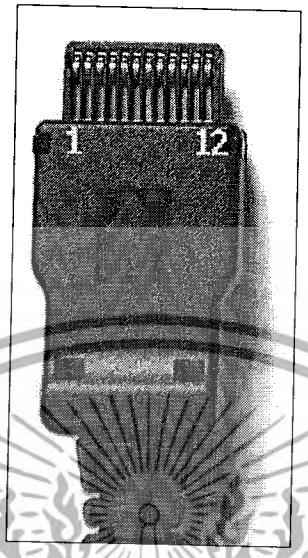
DEV-010i-B schematic



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Popular GPRS phone data link details

Siemens S45 connector's pin out

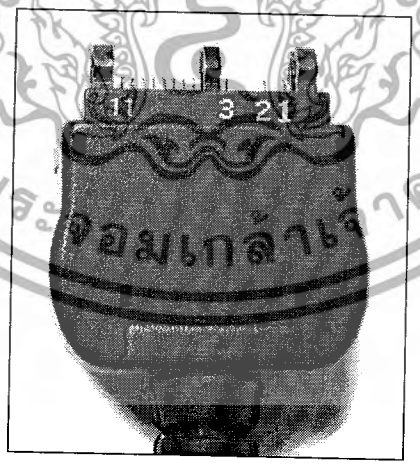


1. Gnd
2. –
3. Charge
4. –
5. Tx (out from phone)
6. Rx (into phone)

Siemens C55

2. Gnd.
3. Tx
4. Rx

Ericsson T65/68/200 connector's pin out



1. Charge
2. Gnd
3. –
4. Gnd
5. –
6. –
7. TX (out from phone)
8. RX (into phone)
9. Power ON (TRIG LOW) **** ONLY ERICSSON T65/T68 ****

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Quick Start Guide

TNS010i works well with any GPRS phone or GPRS modem module.
Such as

- Ericsson T65/T68/T200
- Siemens S45/ C55
- Wavecom Integra-GPRS (GPRS module)
- Etc.

** Recommendation for Wavecom integra **

1. Set GPRS modem to baudrate = 57600
   by send AT+IPR=57600 and
2. Save this setting value in it's EEPROM
   by send AT&W
3. Send AT+WOPEN=0

Following is the sample how to start TNS010i with Ericsson/Siemens phone

Testing phone parameters

1. Ensure that the GPRS phone have been activated and tested to verify the connection.

   In case of Ericsson, and Siemens C55 you need to purchase a data link cable (built in RS232 IC.) available from it's dealer.
2. Set the phone parameter according to the recommendations from your GPRS service provider. You may consult a sample web page of how to set at
   http://www.mobilelife.co.th/mobilelife/t/customertools/mobilesetting/manual/gprs/index.htm
   or from your local GRPS 'provider web site
   Note that the parameters may be vary from country to another, and from one provider to another.
3. Add "New Hardware" GPRS phone/modem to your PC (you will need GPRS phone driver that should come with the phone).
4. Add "New Internet Connection". Choose your GPRS phone as a modem
5. Connect the GPRS phone via its data link cable (built in RS232 IC) and connect to Internet by choosing a GPRS modem.
6. Try surfing the web page via your GPRS phone

If you are successful in surfing the web page with the GPRS phone then proceed to the next step.

Testing the TNS010i chip on Window's Hyperterminal program

As an alternative to reduce the overall manufacturing cost and the package size, a data link cable with no RS232IC in its cable can be used given that there is the same logic voltage level to and from the MCU and the GPRS phone.

7. Close all the Internet connections in PC including any web browser. Then remove the data link cable.
8. Open Hyper terminal program in the Windows (N-8-1 19,200 BPS)
9. Connect the GPRS phone to the a new development board 's data link cable(no RS232 IC) and turn the power on.
10. As soon as the development board is powered up, the "I_READY" sign should appear on the Hyper terminal screen.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11. Type the following sequence of commands in its order

→ at+i <ENTER> /* TCP/IP chip response testing */
← I_OK

→ at+idetect<ENTER> /* detect/set modem baudrate */
← 57600 /* modem's baud rate */
← I_OK

Skip the following gray colour steps if your GPRS modem's parameter was set correctly

---

→ +++<ENTER> /* quit TNS010's commands mode */
← I_OK

wait 0-3 Sec. (depend on brand /model of modem)

→ at<ENTER> /* Initiate modem's commands mode */
← OK

→ at+cgdcont=xxx /* xxx = GPRS parameter setting Example = 1,"ip","internet" */
← OK

→ at+i <ENTER> /* Enter TNS010i's commands mode */
← I_OK

→ at+isetd=xxx /* xxx = ISP dial number Example = *99***1# */
← I_OK

→ at+iopen /* connect to ISP and open the TCP/IP socket */
← I_OK

→ at+ihttp://xxx /* xxx=url IP / url domain name */
Example =203.130.155.66/test.php?content1=123&content2=345 </l , /d , /h>
← I_OK
← (Data's length or date/month/year or full header or nothing (default) depend on the type of command's suffix (/l,/d,/h))
← <html> .....<html> /* HTTP response data*/

เอกสารนี้ใช้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Kingbright

14.22mm (0.56INCH) DUAL DIGIT NUMERIC DISPLAYS

DA56-11EWA/SRWA/YWA/GWA
DC56-11EWA/SRWA/YWA/GWA

Features

- 0.56 INCH DIGIT HEIGHT.
- LOW CURRENT OPERATION.
- EXCELLENT CHARACTER APPEARANCE.
- EASY MOUNTING ON P.C . BOARDS OR SOCKETS.
- TWO DIGIT PACKAGE SIMPLIFIES ALIGNMENTS & ASSEMBLY.
- I.C. COMPATIBLE.
- CATEGORIZED FOR LUMINOUS INTENSITY, YELLOW AND GREEN CATEGORIZED FOR COLOR.
- MECHANICALLY RUGGED.
- STANDARD : GRAY FACE, WHITE SEGMENT.

Description

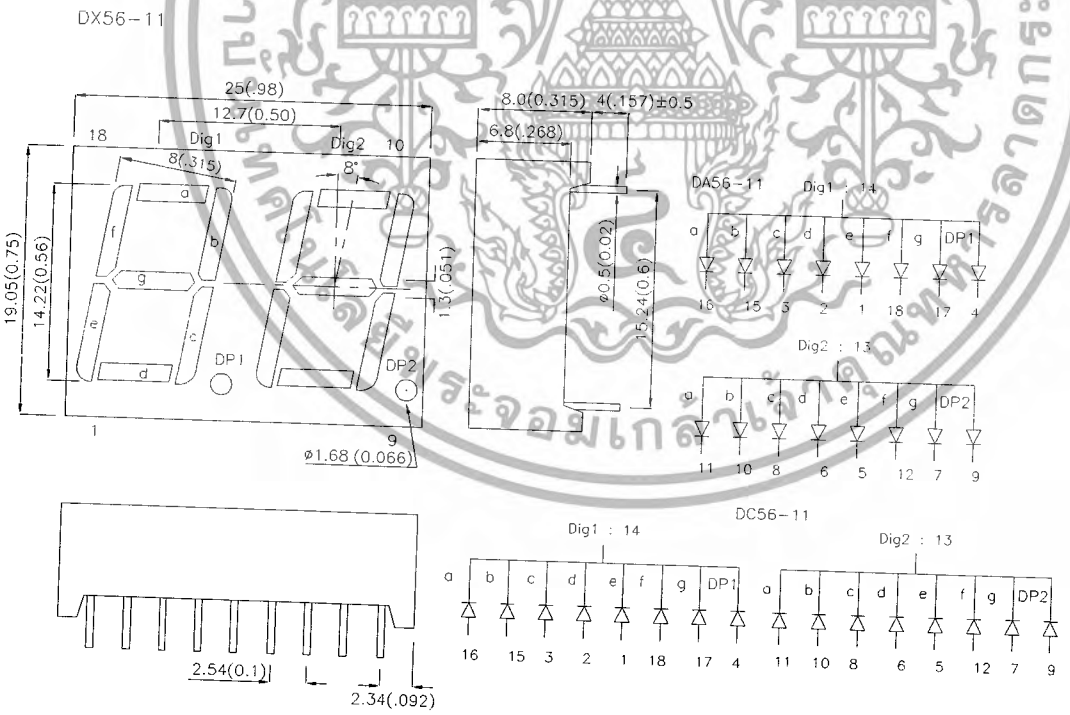
The High Efficiency Red source color devices are made with Gallium Arsenide Phosphide on Gallium Phosphide Orange Light Emitting Diode.

The Super Bright Red source color devices are made with Gallium Aluminum Arsenide Red Light Emitting Diode.

The Yellow source color devices are made with Gallium Arsenide Phosphide on Gallium Phosphide Yellow Light Emitting Diode.

The Green source color devices are made with Gallium Phosphide GreenLight Emitting Diode.

Package Dimensions & Internal Circuit Diagram



Notes:
1. All dimensions are in millimeters (inches), Tolerance is ±0.25(0.01")unless otherwise noted.
2. Specificaions are subject to change whitout notice.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Selection Guide

Part No.	Dice	Lens Type	Iv (ucd) @ 10 mA		Description
			Min.	Typ.	
DA56-11EWA	HIGH EFFICIENCY RED (GaAsP/GaP)	WHITE DIFFUSED	1900	8000	Common Anode.Rt.Hand Decimal
DC56-11EWA					Common Cathode.Rt.Hand Decimal
DA56-11SRWA	SUPER BRIGHT RED (GaAlAs)	WHITE DIFFUSED	8000	24000	Common Anode.Rt.Hand Decimal
DC56-11SRWA					Common Cathode.Rt.Hand Decimal
DA56-11YWA	YELLOW (GaAsP/GaP)	WHITE DIFFUSED	1900	4700	Common Anode.Rt.Hand Decimal
DC56-11YWA					Common Cathode.Rt.Hand Decimal
DA56-11GWA	GREEN (GaP)	WHITE DIFFUSED	3000	10500	Common Anode.Rt.Hand Decimal
DC56-11GWA					Common Cathode.Rt.Hand Decimal

Electrical / Optical Characteristics at T_A=25°C

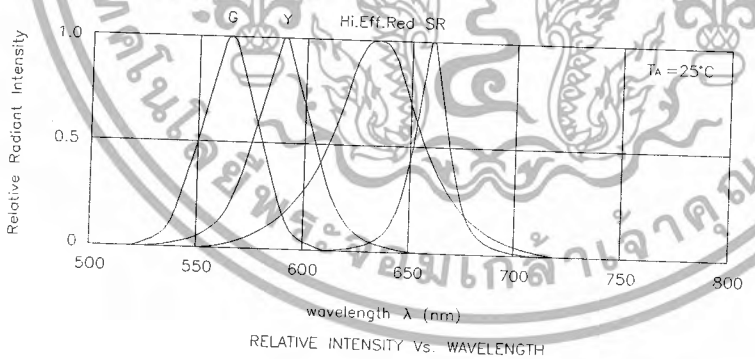
Symbol	Parameter	Device	Typ.	Max.	Units	Test Conditions
λpeak	Peak Wavelength	High Efficiency Red Super Bright Red Yellow Green	627 660 590 565		nm	I_F=20mA
λ D	Dominate Wavelength	High Efficiency Red Super Bright Red Yellow Green	625 640 588 568		nm	I_F=20mA
Δλ1/2	Spectral Line Half-width	High Efficiency Red Super Bright Red Yellow Green	45 20 35 30		nm	I_F=20mA
C	Capacitance	High Efficiency Red Super Bright Red Yellow Green	15 45 20 15		pF	V_F=0V;f=1MHz
V_F	Forward Voltage	High Efficiency Red Super Bright Red Yellow Green	2.0 1.85 2.1 2.2	2.5 2.5 2.5 2.5	V	I_F=20mA
I_R	Reverse Current	All		10	uA	V_R = 5V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Absolute Maximum Ratings at T_A=25°C

Parameter	High Efficiency Red	Super Bright Red	Yellow	Green	Units
Power dissipation	105	100	105	105	mW
DC Forward Current	30	30	30	25	mA
Peak Forward Current [1]	160	155	140	140	mA
Reverse Voltage	5	5	5	5	V
Operating/Storage Temperature	-40°C To +85°C				
Lead Solder Temperature [2]	260°C For 5 Seconds				

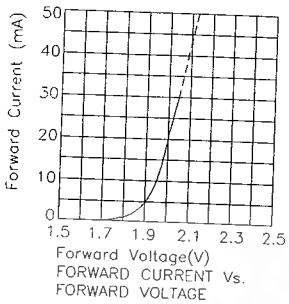
Notes:
1. 1/10 Duty Cycle, 0.1ms Pulse Width.
2. 2mm below package base.



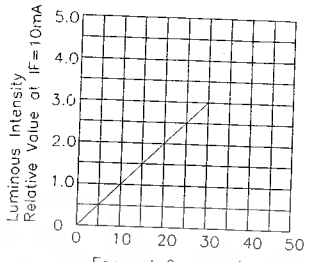
RELATIVE INTENSITY Vs. WAVELENGTH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

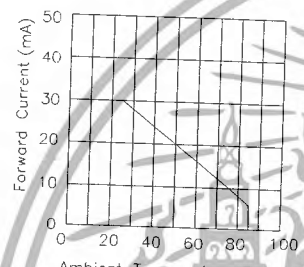
High Efficiency Red



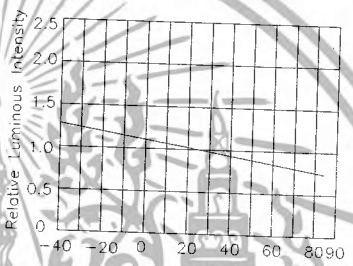
FORWARD CURRENT Vs.
FORWARD VOLTAGE



LUMINOUS INTENSITY Vs.
FORWARD CURRENT

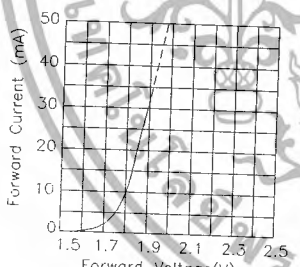


FORWARD CURRENT
DERATING CURVE

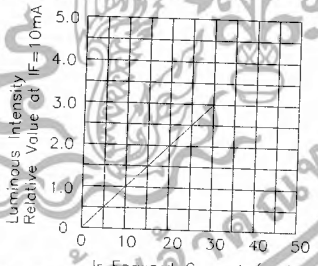


LUMINOUS INTENSITY Vs.
AMBIENT TEMPERATURE

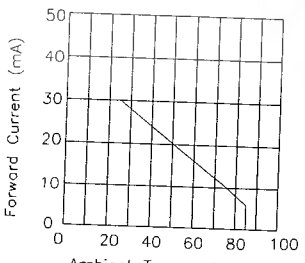
Super Bright Red



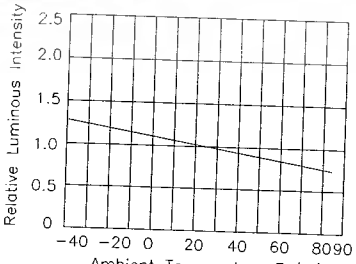
FORWARD CURRENT Vs.
FORWARD VOLTAGE



LUMINOUS INTENSITY Vs.
FORWARD CURRENT



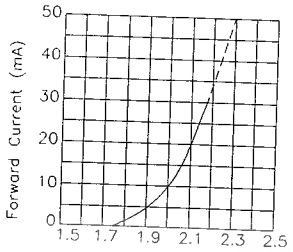
FORWARD CURRENT
DERATING CURVE



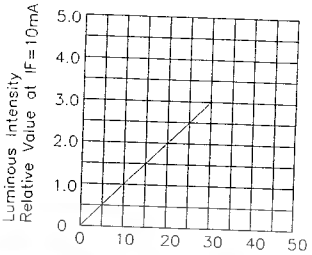
LUMINOUS INTENSITY Vs.
AMBIENT TEMPERATURE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

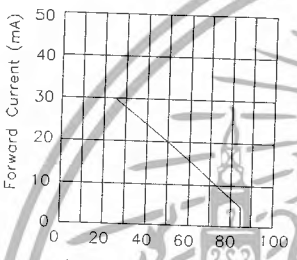
Yellow



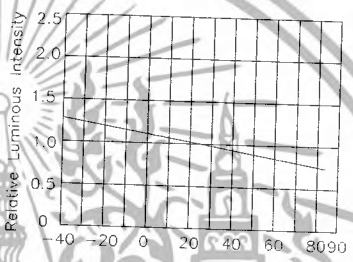
FORWARD CURRENT Vs.
FORWARD VOLTAGE



LUMINOUS INTENSITY Vs.
FORWARD CURRENT

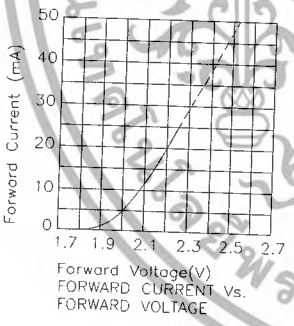


FORWARD CURRENT
DERATING CURVE

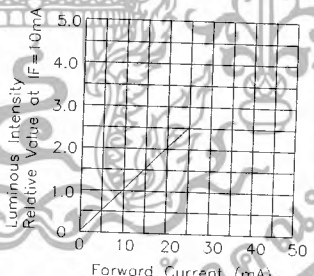


LUMINOUS INTENSITY Vs.
AMBIENT TEMPERATURE

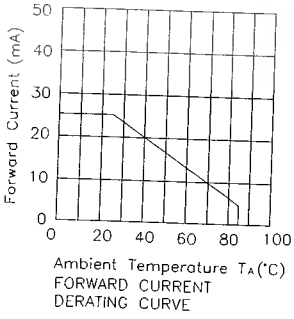
Green



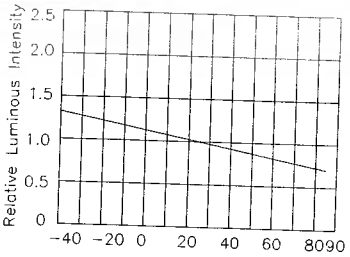
FORWARD CURRENT Vs.
FORWARD VOLTAGE



LUMINOUS INTENSITY Vs.
FORWARD CURRENT



FORWARD CURRENT
DERATING CURVE



LUMINOUS INTENSITY Vs.
AMBIENT TEMPERATURE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้