สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบควบคุมกล้องวิดีโอผ่านอินเตอร์เน็ต

Video Camera Control Via Internet

จัดทำโดย

นาย สุทธิพล พันธุ์ธาดาพร

นาย สุรศักดิ์ วิทยากรวณิช

นาย เอก สุจินดา

อาจารย์ที่ปรึกษา

รศ.ดร. รัตติกร วรากูลศิริพันธุ์

ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขา อิเล็กทรอนิกส์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2542

เลขหมู่.....
เลขทะเบียน.....36939
วัน, เดือน, ปี.....29 ส.ค. 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์เรื่อง ระบบควบคุมการททำงานกล้องวิดีโอผ่านอินเตอร์เน็ต

Video Camera Operation Control Via Internet

จัดทำโดย นาย สุทธิพล พันธุ์ธาดาพร เลขประจำตัว 40013192

นาย สุรศักดิ์ วิทยากรวณิช เลขประจำตัว 40013194

นาย เอก สุจินดา เลขประจำตัว 40013200

ปริญญานิพนธ์ฉบับนี้ได้ผ่านการตรวจสอบโดยอาจารย์ที่ปรึกษาแล้ว

ลงชื่อ.....อาจารย์ที่ปรึกษา

(รศ.ดร.รัตติกร วรากูลศิริพันธ์)

วันที่ 23 / 3 / 43

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบควบคุมกล้องวิดีโอผ่านอินเตอร์เน็ต

นาย สุทธิพล พันธุ์ธาดาพร 3R/1 รหัส 40013192
นาย สุรศักดิ์ วิทยากรวณิช 3R/1 รหัส 40013194
นาย เอก สุจินดา 3R/1 รหัส 40013200
รศ.ดร.รัตติกร วรากูลศิริพันธุ์ (อาจารย์ที่ปรึกษา)

บทคัดย่อ

ปัจจุบันนี้ อินเตอร์เน็ต เข้ามามีบทบาทในชีวิตประจำวันของเรามากขึ้น โดยเฉพาะการสื่อสารในรูปแบบต่างๆ เช่น การค้าบนอินเตอร์เน็ต การสนทนาผ่านทาง อินเตอร์เน็ต ไม่ว่าเราจะอยู่แห่งใดบนโลกนี้ก็สามารถ เข้าสู่โลกอินเตอร์เน็ตอย่างง่ายดาย จากคุณประโยชน์นี้ผนวกกับความรู้ด้าน ฮาร์ดแวร์ และซอฟต์แวร์ และเน็ตเวิร์ค มาประยุกต์ใช้งานให้เกิดประโยชน์ในด้านการสื่อสาร โครงงานนี้ ได้มุ่งเน้นการควบคุมระยะไกลและการส่งภาพ ซึ่งนำมาใช้ในงานด้านต่างๆ ไม่ว่าจะเป็นระบบรักษาความปลอดภัย การนำไปติดตั้งในที่ไม่ปลอดภัยซึ่งคนไม่สามารถเขาไปได้ เราใช้การควบคุมผ่าน อินเตอร์เน็ต เราอยู่ที่ใดก็สามารถควบคุมได้ อนาคตอินเตอร์เน็ตจะเป็นปัจจัยที่สำคัญต่อเรา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Video Camera Control Via Internet

Mr.Suttipon Pantadaporn 3R/1 code 40013192

Mr.Surasak Vittayakronvanich 3R/1 code 40013194

Mr.Ake Suchinda 3R/1 code 40013200

Assoc.Prof.Dr.Ruttikorn Varakulsiripunth. (adviser)

Abstract

Nowaday the internet extremely become useful in every day life especially in communication such as the chatting commercial on the internet. Wherever we are, we can access the internet world easily. This advantage with the knowledge of network, hardware and software are adapted to create the purpose in communication . This project emphasize the remote and picture transferring Which are used in many ways for example the security system, unsafe area that people can not entrance easily. Which we can control in anywhere. In the future the internet would play an important role in our life

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์เรื่อง ระบบควบคุมกล้องวิดีโอผ่านเครือข่ายอินเตอร์เน็ต โครงงานนี้มีอาจารย์ที่ปรึกษาคือ รศ.ดร. รัตติกร วรากูลศิริพันธุ์ เป็นอาจารย์ที่ปรึกษา ซึ่งต้องขอขอบคุณท่านเป็นอย่างยิ่งสำหรับอุปกรณ์ที่ใช้ในการทำโครงงานครั้งนี้ ขอขอบคุณพี่ตุ๊ก ที่ให้คำแนะนำในเบื้องต้นของโครงงานครั้งนี้ ขอขอบคุณ พี่ พารา ลิมมะณีประเสริฐและพี่ ป.โทที่ Communication Network Lab ทุกท่านที่ให้คำแนะนำและความช่วยเหลือต่างๆ และขอขอบใจเพื่อนๆทุกคน ที่คอยให้กำลังใจด้วยดีตลอดมา

นาย สุทธิพล พันธาดาพร
นาย สุรศักดิ์ วิทยากรวณิช
นาย เอก สุจินดา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทที่ 1 บทนำ	1
บทที่ 2 เครือข่ายอินเตอร์เน็ต	3
2.1 สถาปัตยกรรมอินเตอร์เน็ต	3
2.2 OSI โมเดล	4
2.3 โครงสร้างชั้นเน็ตเวิร์ค	4
2.4 รูปแบบมาตรฐานโปรโตคอลของอินเตอร์เน็ต	7
2.5 Internet IP	7
บทที่3 หลักการเขียนโปรแกรม Winsock ที่ใช้กับ TCP/IP	12
3.1 Process layer	13
3.2 Host-to-Host layer	14
3.3 โปรโตคอล TCP	16
3.4 โปรโตคอล UDP	17
3.5 Internetwork Layer	19
3.6 โปรโตคอล IP	19
3.7 โปรโตคอล ICMP	19
3.8 โปรโตคอล ARP	20
3.9 Network Interface Layer	21
3.10 กลไกของโปรโตคอล IP	23
3.10.1 IP Datagram	24
3.10.2 IP Header	24
3.10.3 IP Host Address and Routing	25
3.10.4 Host Names	26
3.11 รู้จักกับ Winsock	27
บทที่ 4 พื้นฐานไมโครคอนโทรลเลอร์และการสื่อสารแบบอนุกรม	31
4.1 ไมโครคอนโทรลเลอร์ และการสื่อสารแบบอนุกรม	31
4.2 โครงสร้างของหน่วยความจำใน MCS-51	34
4.3 โครงสร้างของพอร์ทอินพุตเอาท์พุต	35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 รีจิสเตอร์ฟังก์ชันพิเศษ 36
4.4.1 Program status Word 36
4.4.2 รีจิสเตอร์ B 37
4.4.3 ตัวชี้แสตก (Stack Pointer) 37
4.4.4 รีจิสเตอร์ Data Pointer (DPTR) 37
4.4.5 รีจิสเตอร์พอร์ท (Port Register) 37
4.4.6 รีจิสเตอร์เวลา (Timer Register) 38
4.4.7 รีจิสเตอร์พอร์ทอนุกรม (Serial Port Register) 38
4.4.8 รีจิสเตอร์อินเตอร์รัพท์ (Interrupt Port Register) 38
4.4.9 Power Control Register (PCON) 38
4.5 การสื่อสารแบบอนุกรม 38
4.5.1 MCS-51 กับการรับข้อมูลแบบอนุกรม 39
บทที่ 5 หลักการออกแบบและการสร้าง 40
5.1 ชุดควบคุมกล้อง (Camera Control) 40
5.1.1 ส่วนฐานจับกล้อง 40
5.1.2 ส่วนไมโครคอนโทลเลอร์ และวงจรขับมอเตอร์ 40
5.2 ชุดโปรแกรมการติดต่อใช้งานระหว่างผู้ควบคุมและผู้ใช้บริการ 47
5.2.1 โปรแกรมควบคุมด้านผู้ใช้ (Client Side Program) 47
5.2.2 โปรแกรมด้านให้บริการ (Server Side Program) 47
บทที่ 6 สรุปผลการทดลอง 50
บรรณานุกรม 56
ภาคผนวก

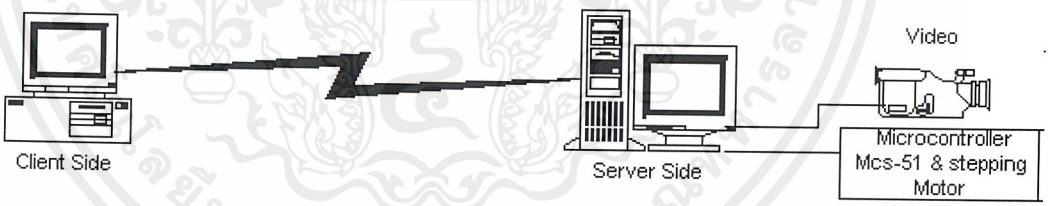
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

ทุกวันนี้ Internet เข้ามามีบทบาทในชีวิตประจำวันเป็นอย่างมากโดยเฉพาะการติดต่อสื่อสารทุกรูปแบบเช่น การสนทนา, การสั่งซื้อสินค้า,การจองห้องพัก,การจองตั๋วเครื่องบินและการควบคุมระยะไกล(Remote Control) เป็นต้น ไม่ว่าเราจะอยู่ที่ใดก็ตามก็จะสามารถเข้าถึงอินเตอร์เน็ตได้ เนื่องจากอินเตอร์เน็ตนั้นครอบคุมทั่วทุกมุมโลก

จากข้อดีของอินเตอร์เน็ตหลายข้อทำให้เกิดแนวความคิดนำการนำระบบอินเตอร์เน็ตมาผสมผสานกับอิเล็กทรอนิกส์มาประยุกต์ใช้ในการควบคุมระยะไกลโดยประกอบด้วยส่วน Hard ware และส่วน Software เพียงเท่านี้ก็สามารถนำประโยชน์ใช้ในที่ต่างที่มีความยากลำบากต่อการทำงานเช่นในสถานที่ มีความร้อนสูง,ในที่มีสารกัมมันตภาพรังสี ที่เป็นอันตรายต่อมนุษย์ และใช้ในการรักษาความปลอดภัยซึ่งจะเป็นประโยชน์มากอีกทั้งทำให้ประหยัดทรัพย์สินด้วย

โครงงานนี้ได้นำเอาแนวความคิดจากตัวอย่างข้างต้นมาทำการศึกษาและสร้างเครื่องมือชุดนี้ขึ้นมาเพื่อให้ได้มาซึ่งวัตถุประสงค์ที่ต้องการ โดยหลักการทำงานอย่างคร่าวๆ สามารถดูได้จากบล็อกรูปที่ 1.1 ดังนี้



รูปที่ 1.1 แสดงการเชื่อมต่ออุปกรณ์ควบคุมกับเครือข่ายอินเตอร์เน็ต

จากบล็อกข้างต้นสามารถที่จะอธิบายได้ดังนี้

1. ส่วนควบคุม (Client Control) จะประกอบไปด้วยคอมพิวเตอร์ที่เชื่อมต่อเข้ากับเครือข่ายอินเตอร์เน็ตและบรรจุโปรแกรมควบคุมการทำงานในส่วนของโครงงานนี้เอาไว้

2. ส่วนประมวลผล (Server Process) จะเป็นคอมพิวเตอร์ที่ทำให้เป็น Server ที่เอาไว้สำหรับติดต่อกับส่วนของส่วนควบคุมโดยผ่านเครือข่ายอินเตอร์เน็ต และนำผลของการประมวลผลไปส่งยังส่วนควบคุมกล้องซึ่งจะได้กล่าวถึงต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ส่วนควบคุมกล้อง (Camera Control Unit) จะเป็นส่วนที่รับคำสั่งควบคุมการทำงานจากส่วนประมวลผลมาปฏิบัติตามแล้วและจะรายงานผลตำแหน่งของกล้องด้วยว่าขณะนี้ที่ตำแหน่งใด

4. กล้องถ่ายภาพ (Video Camera) ทำหน้าที่ แปลงภาพให้เป็นสัญญาณ VDO เพื่อส่งไปให้คอมพิวเตอร์

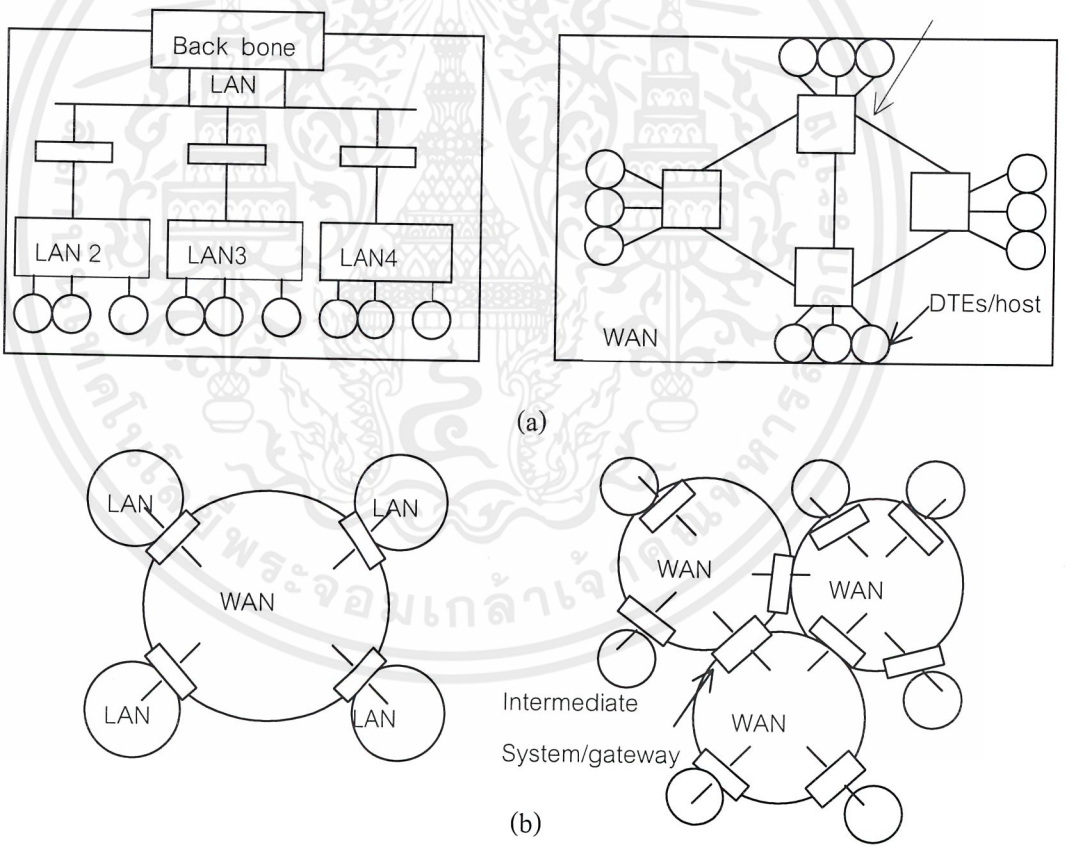
จากการทำงานเบื้องต้นนี้ บทต่อๆ ไปจะกล่าวถึงทฤษฎีที่ใช้ในการออกแบบและหลักการออกแบบในส่วนต่างในบทต่อๆ ไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
เครือข่ายอินเตอร์เน็ต

อินเตอร์เน็ต (Internet) คือการที่เครือข่าย 2 หรือมากกว่า เชื่อมต่อด้วยกัน และการทำงานเสมือนเป็นเครือข่ายเดียวกัน โดย network ที่เป็นส่วนประกอบของ internet คือ Subnetwork (subnet) ซึ่งการจะเป็นเครือข่าย Local Area network (LAN) หรือ Wide area network (WAN) อุปกรณ์ที่ใช้ในการเชื่อมต่อ 2 เครือข่ายเข้าด้วยกันก็คือ intermediate system (IS) หรือ internetworking unit (IWU) การเชื่อมโยงระหว่างระบบที่แตกต่างกัน จำเป็นต้องมีมาตรฐานการติดต่อกัน ซึ่งเรียกเป็นศัพท์เฉพาะว่าโปรโตคอล (protocol)

2.1 สถาปัตยกรรม อินเตอร์เน็ต (Internet Architectures)



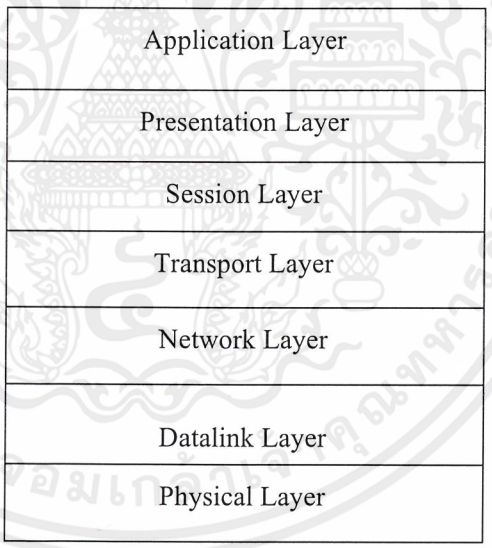
รูปที่ 2.1 แสดงสถาปัตยกรรมของ internet (a) single LAN and WAN(b) Interconnected LAN/WAN

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในรูป (a) แสดงตัวอย่าง 2 ตัวอย่างของเครือข่ายเดี่ยว (Single network) ซึ่งอย่างแรกเป็น site-wide LAN ซึ่งประกอบขึ้นมาจากชุดของ LANs ซึ่งถูกต่อเข้ากับเครือข่ายหลัก(backbone) ซึ่งอุปกรณ์ที่ใช้ต่อ LAN เข้ากับเครือข่ายหลัก ถ้า LAN ทุกเครือข่ายมีระบบเดียวกันก็จะใช้ bridge ถ้าเป็น LAN ที่แตกต่างกันก็จะใช้ router ตัวอย่างที่ 2 เป็นตัวอย่างของ WAN เดี่ยวๆ ในรูป(b) แสดงถึงเครือข่ายอินเตอร์เน็ต ซึ่งประกอบด้วย network ทั้ง 2 ชนิด ข้างต้น

2.2 **OSI โมเดล**

องค์ประกอบมาตรฐานสากล ISO (International Organization for standardization) ได้กำหนดมาตรฐานของเครือข่าย โดยจัดแบ่งกิจกรรมของเครือข่าย ออกเป็นงานย่อย ๆ และกำหนดโมเดลแบ่งเป็นชั้น ๆ ตามลำดับเรียกว่ามาตรฐาน OSI (Open System Interconnection) โดยที่จะแบ่งกิจกรรมที่ซับซ้อนในเครือข่าย ออกเป็นงานย่อยๆ ก็จะช่วยในการออกแบบ และการใช้งานเครือข่ายรวมถึงการเชื่อมโยงกัน เป็นไปได้ด้วยความสะดวก และมีวิธีการทำงานอยู่ในกรอบเดียวกันดังรูปที่ 2.2



รูปที่ 2.2 แสดงการแบ่งการทำงานของเครือข่ายออกเป็น OSI model

ในแต่ละชั้นของ OSI model จะมีการติดต่อสื่อสารกันเป็นชั้น ๆ ตามลำดับลงมาเช่น Application Layer ก็จะติดต่อสื่อสารกับ Presentation Layer ตามลำดับไปจนถึงชั้นแรกสุดคือ Physical Layer

Application Layer เป็นชั้นบนสุดจอง โมเดลเป็นส่วนที่จะทำให้การติดต่อระหว่างเครือข่ายกับผู้ใช้เป็น เป็นไปตามต้องการ ตัวอย่างแอปพลิเคชั่นของเครือข่าย เช่น ระบบ e-mail, การโอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ่ายข้อมูล(File Transfer) ,การขอเข้าใช้ระบบคอมพิวเตอร์ในเครือข่ายเป็นต้นPresentation Layer มีการกำหนดหน้าที่ไม่ชัดเจนและมีการนำไปใช้ไม่มาก ซึ่งหน้าที่หลักก็คือเป็นส่วนที่จัดรูปและนำเสนอข้อมูล ให้เป็นไปตามต้องการ รวมไปถึงการจัดแปลงข้อมูล ในรูปมาตรฐาน ASCII หรือ EBCDIC การลดขนาดข้อมูล (data compression)การเข้ารหัส หรือ session .ให้ระบบ คอมพิวเตอร์ทั้งสองฝั่ง โดยทำหน้าที่ตั้งแต่เริ่มการติดต่อ ดูแลในการส่งผ่านข้อมูล ในการติดต่อครั้งนี้นๆ เป็นไปได้โดยไม่มีปัญหา จนถึงเลิกการติดต่อเมื่อเสร็จงาน

Transport Layer ทำหน้าที่ควบคุมปริมาณ และรายละเอียดการรับส่งข้อมูล ให้เป็นไปตามกำหนดที่ตั้งไว้ และจัดการให้การเชื่อมโยงเครือข่ายเป็นไปด้วยความราบรื่น Transport Layer จะเป็นชั้นสุดท้ายที่จัดการเรื่องเส้นทางในการส่งข้อมูล และจัดการตรวจสอบความผิดพลาดของข้อมูล ซึ่งส่วนของ TCP (Transmission Control Protocol) ในโปรโตคอล TCP/IP ทำงานที่ระดับนี้

Network Layer ทำหน้าที่ควบคุมวิธีการส่งผ่านข้อมูลระหว่างเครือข่ายให้ถูกต้อง และเป็นไปตามเส้นทางที่กำหนดโดยจะจัดการส่งผ่าน packet ข้อมูล ผ่านอุปกรณ์ต่างๆ ไปยังเครือข่ายย่อยได้อย่างถูกต้องตามที่ต้องการ นอกจากนี้จังจัดการดูแลเส้นทางในการส่งจ้อมูล (Routing table) และกั้นกรอง packet ข้อมูลที่ส่งไปยังเครือข่ายเดียวกันไม่ให้ข้ามไปยังเครือข่ายอื่น ซึ่งจะช่วยลดปริมาณข้อมูลที่จะวิ่งบนเครือข่ายได้ส่วนหนึ่ง โปรโตคอล IP.,TCP/IP) และ Ipx เป็นโปรโตคอลที่ทำงานอยู่ใน layer นี้

Data link Layer ทำหน้าที่เรียกใช้หรือกำหนดช่องทาง ในการส่งข้อมูลที่ถูต้อง เช่น Ethernet, Tokenring หรือ FDDI เป็นต้น รวมถึงการลำดับและอัตราการรับส่งข้อมูลหรือ flow control และสถานที่ ที่จะส่งข้อมูลไป (address) ทั้งนี้ Data link layer จะเป็นชั้นแรกที่จัดการแปลงข้อมูลจาก bit ให้เป็น packet โดยจะมีการเพิ่มข้อมูลเพื่อตรวจสอบผ่าน checksum เพื่อดูว่าข้อมูลที่ได้รับมาถูกต้องครบถ้วน และถ้าได้รับ packet ข้อมูลที่ไม่ถูกต้องก็จะไม่เอาข้อมูลนั้นไปใช้งานและบอกให้ต้นทางส่งข้อมูลเดิมมาใหม่

Physical Layer รับผิดชอบดูแลในรายละเอียดในการส่งข้อมูลในด้าน hardware เช่น การควบคุม Network Interface Card การส่งสัญญาณแบบต่าง ๆ การเชื่อมต่อเข้ากับเครือข่ายค่างๆ โดยใช้ Physical Layer จะจัดสร้างสัญญาณทางไฟฟ้า, สัญญาณทางไฟฟ้า,สัญญาณเสียง หรือสัญญาณที่จำเป็นในการสื่อสารโดยตรง

เนื่องจาก Network Layer ในแต่ละ End System (ES) จะเป็นตัวจัดการการติดต่อ แบบ end-to-end ของการบริการ internetwide ไปยังผู้ใช้บริการ(NS-User)

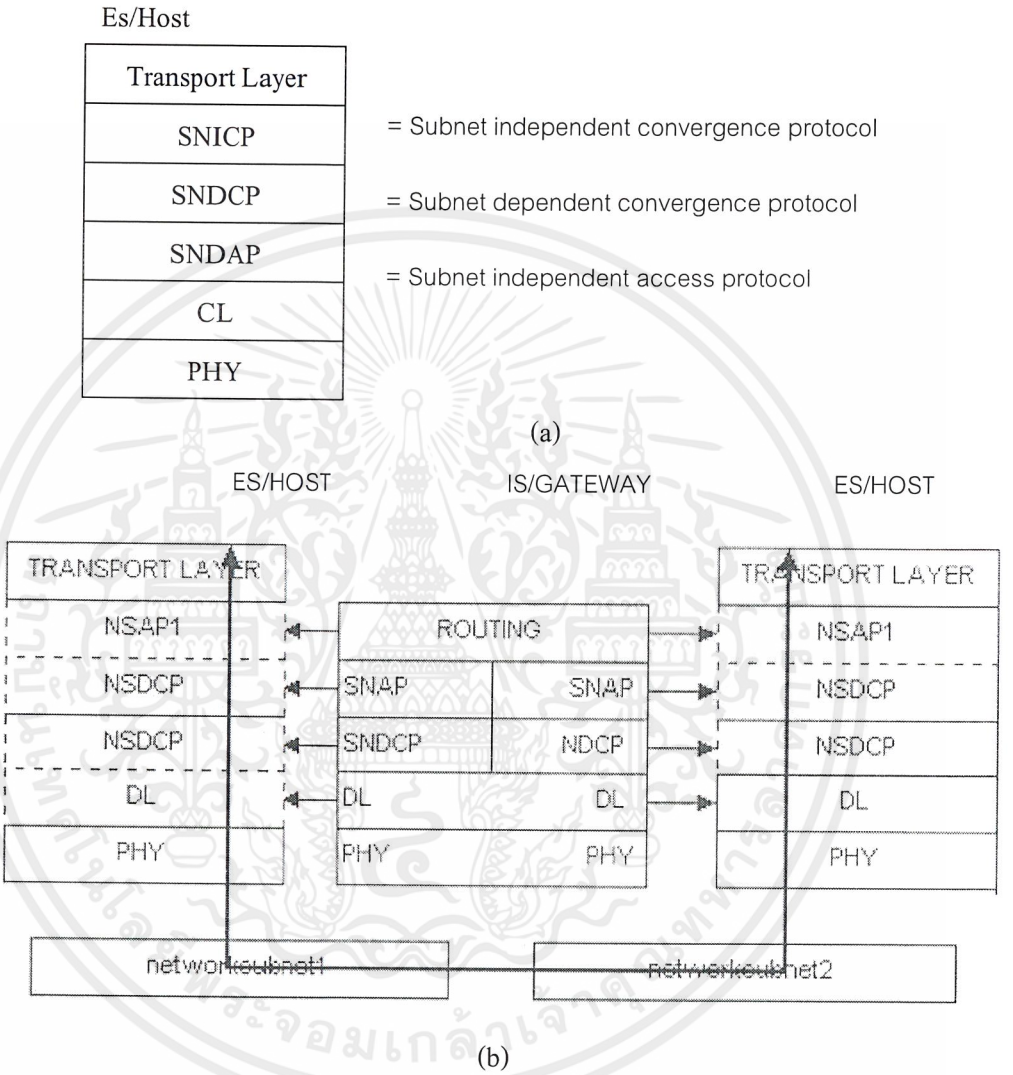
โดย ISO ได้จัด network layer เป็น 3 (sublayer) Protocol ซึ่งทำงานร่วมกัน เพื่อให้บริการใน network layer ได้แก่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Subnetwork independent convergence protocol (SNICP)

Subnetwork dependent convergence protocol (SNDCP)

Subnetwork dependent access protocol (SNDAP)



รูปที่ 2.3 network Layer structure (a) Sub layer Protocol ; (b) IS structure

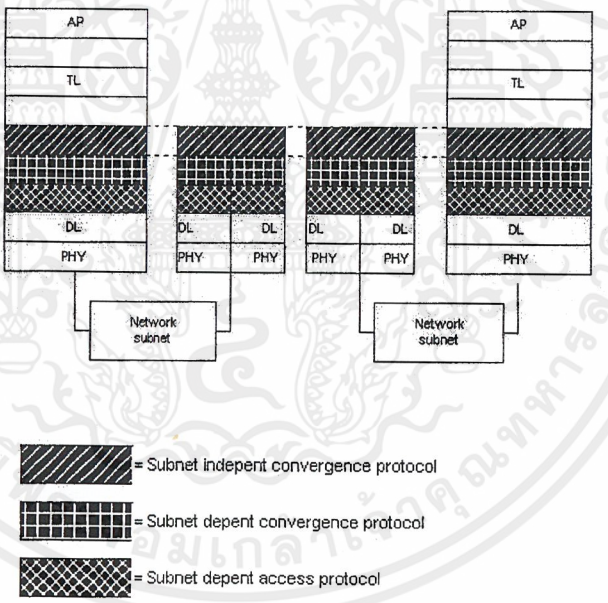
โดยที SNICP จะเป็นตัวสนับสนุนจัดการให้ผู้ใช้บริการ (NS-user) สามารถ interface กับ internetซึ่งมันจะมีหน้าที่เป็นตัวประสานฟังก์ชันต่าง ๆ ที่จำเป็นในการเลือกเส้นทางและถ่ายทอดข้อมูลของผู้ใช้ข้าม Internet ซึ่งการทำงานของมันไม่ขึ้นอยู่กับคุณสมบัติเฉพาะของเครือข่ายย่อย (subnet)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SNDAP จะเป็นตัวโปรโตคอลที่ติดต่อกับเครือข่ายย่อย (subnet) ที่มีลักษณะเฉพาะใน Internet เช่น X.25 packet layer protocol สำหรับเครือข่าย X.25 ซึ่งใช้บ่อยใน LAN เพราะว่าการบริการปละการทำงาน SNDAP แตกต่างจาก network แบบอื่น ๆ sublayer ที่อยู่ตรงกลางคือ SNDCP จะเป็นตัวจัดการะหว่าง SNICP และ SNDAP

2.4 รูปแบบมาตรฐานโปรโตคอลของอินเตอร์เน็ต(Internet Protocol standards)

อินเตอร์เน็ตโปรโตคอลซึ่งถูกใช้ในอินเตอร์เน็ตคือ TCP/IP (Transfer Control Protocol/Internet Protocol) ซึ่งรวมถึง transport และ application โปรโตคอล ซึ่งทั้งหมดของ TCP/IP จะกำหนด ให้เหมาะสมกับการใช้ในเชิงสาธารณะ ซึ่งรูปแบบโดยทั่วไปแสดงดังรูปที่ 2.4 IP เป็น internetwide protocol ซึ่งทำให้สอง transport protocol ที่ต่างสถานที่กันและต่าง Ess / hosts กันสามารถแลกเปลี่ยนหน่วยข้อมูล (NSDUS) กันได้ ซึ่งหมายถึงว่า หลายๆ network / subnet และ ISO/ gateways ที่แตกต่างกันสามารถ ติดต่อสื่อสารกันได้อย่างสมบูรณ์



รูปที่ 2.4 Internetwide IP schematic

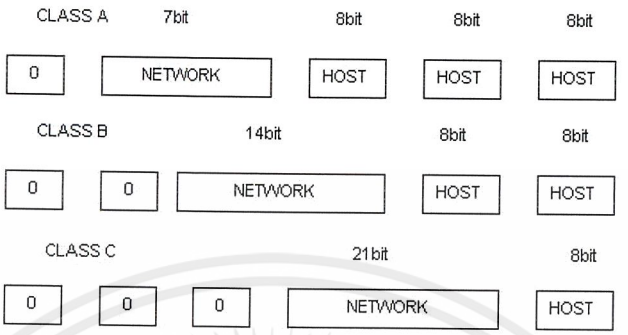
2.5 Internet IP

2.5.1 address structure

ในศัพท์ของ ISO เมื่อ2 network ติดต่อกันด้วย host /ES ที่ต่อกับ อินเตอร์เน็ต network เหล่านี้ ติดต่อกันได้โดยใช้ network service access point (NSAP) address และ subnet point of attachment (SNPA) สำหรับใน TCP/IP ก็จะมี IP address และ NPA address ตามลำดับ โดย NPA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

address จะแตกต่างกันในแต่ละชนิดของ และชนิดของ network / subnet ขณะที่ IP address ตามลำดับ โดย NPA address จะเป็นรูปแบบเดียวกัน โครงสร้างของ IP address แสดงดังรูปที่ 2.5



รูปที่ 2.5 โครงสร้างของ Address ที่ใช้ใน class ต่างๆ ของเครือข่ายทั้งหมดความยาว 32 bit

IP address นี้มีการจัดแบ่งงออกเป็นทั้งหมด 5 ระดับ (class) แต่ที่ใช้งานทั่วไปจะมีเพียง 3 ระดับคือคอมพิวเตอร์ที่เชื่อมต่ออยู่มาก จะมีหมายเลขอยู่ใน Class A และลดหลั่นกันมาใน Class B และ Class C ตามลำดับ

จากรูปจะเห็นว่าหมายเลข IP ของ Class A มีคัวแรกเป็น 0 และหมายเลขเครือข่าย (network number) ขนาด 7 bit และมีหมายเลขเครื่องคอมพิวเตอร์ (Host number) ขนาด 24 bit ทำให้ในหนึ่งเครือข่ายของ Class A สามารถ มีคอมพิวเตอร์เชื่อมต่ออยู่ในเครือข่ายได้ถึง $2^{24} = 16$ ล้านเครื่องแต่ใน Class A จะมีหมายเลขเครือข่ายได้ 128 ตัวเท่านั้นทั่วโลก ซึ่งก็คือจะมีเครือข่ายใหญ่แบบนี้เพียง 128 เครือข่ายเท่านั้น

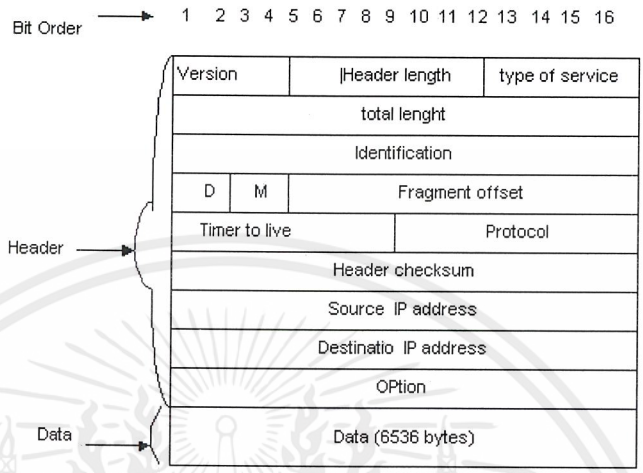
สำหรับ Class B จะมีหมายเลขเครือข่ายแบบเครื่องคอมพิวเตอร์แบบ 16 bit (ส่วนอีก 2 bit ที่เหลือบังคับว่าต้องขึ้นต้นด้วย 10_2) ดังนั้นจึงสามารถมีคอมพิวเตอร์เชื่อมต่อในเครือข่าย Class B แต่ละเครือข่ายได้ถึง 2^{16} กว่า 65000 เครื่อง และสุดท้ายคือ Class C ซึ่งมีหมายเลขคอมพิวเตอร์แบบ 8 bit และมีหมายเลขเครือแบบ21 bit ส่วน 3 bit แรกบังคับว่าต้องเป็น 110_2 ดังนั้นในแต่ละเครือข่าย($2^8 = 256$ แต่หมายเลข 0 และ 255 จะไม่ถูกใช้งานจึงเหลือเพียง 254)

จะเห็นได้ว่าเมื่อเครือข่ายและเครื่องคอมพิวเตอร์ที่ต่ออยู่ในอินเตอร์เน็ตมีหมายเลข IP address ให้ใช้อ้างอิงได้ไม่ซ้ำกันและมีความหมายให้ทรายถึงขนาดเครือข่ายแล้ว การติดต่อส่งผ่านข้อมูลจึงกระทำได้ไม่สับสน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.2 รูปแบบของข้อมูล(Datagrams)

รูปแบบของ IP data unit ก็คือ datagrams ซึ่งโครงสร้างของ datagrams เป็นดังรูปที่ 2.6



รูปที่ 2.6 Internet datagrams format and contents

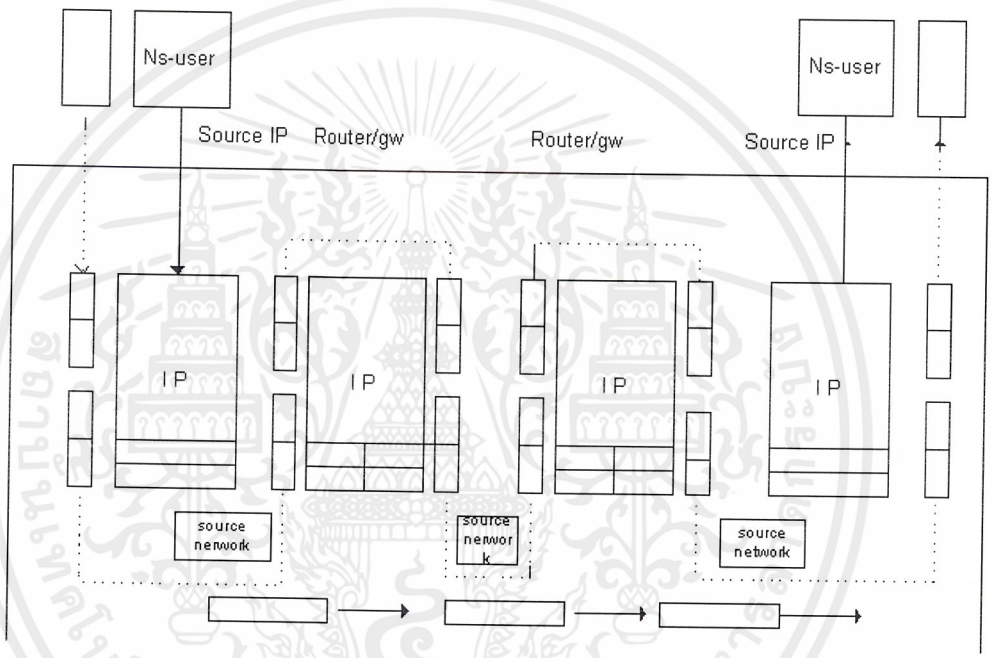
หน่วยข้อมูล IP (IP datagrams) แต่ละหน่วยจะประกอบด้วย ส่วนของข้อมูลที่รับมาจากส่วนของงาน TCP หรือ UDP และส่วนของข้อมูลนำทาง (Header) ซึ่งมีรายละเอียดดังนี้

- Version หมายถึงรุ่นของข้อกำหนด IP
- Header Length ความยามของข้อมูลนำทาง
- Type of service วิธีการจัดการกับข้อมูล
- Total Length ความยาวของหน่วยข้อมูล
- Identification ,Flags และ Fragment offset รายและเอียดที่เกี่ยวกับการแบ่งย่อยข้อมูลซึ่งจะถูนำมาใช้ในการรวบรวมข้อมูล
- Time to live เวลาสูงสุดที่ใช้ในการเดินทาง ซึ่งกำหนดมาจากต้นทาง เวลานี้จะลดลงเรื่อยในระหว่างทางถ้าลดลงไปถึงศูนย์ หน่วยข้อมูลนั้นจะถูกกำจัดไป
- Protocol ชนิดของข้อมูลเป็น UDP หรือ TCP
- Header Checksum ค่าตรวจสอบข้อมูลนำทาง
- IP address หมายเลข internetwide IP (NSP) ของเครื่องต้นทางและปลายทาง
- Option ข้อมูลอื่น ๆ เช่นข้อมูลเกี่ยวกับการรักษาความปลอกภัย บันทึกเส้นทางเดินของข้อมูล และเวลาที่ข้อมูลเดินทางมาถึง เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.3 **การแบ่งส่วนของข้อมูลและการประกอบขึ้นใหม่ (Fragmentation and Reassembly)**

ขนาดข้อมูลของผู้ใช้ซึ่งอ้างอิง NSDU มีความจุ ได้ถึง 64 k หรือ 65,536 bytes ต่อขนาดของหน่วยข้อมูล (packet size) ที่สามารถติดต่อกันในระบบที่ต่างกัน สามารถมีได้ตั้งแต่ 128 byte สำหรับระบบ X.25 packet switching จนถึง 8,000 bytes สำหรับบาง LAN ดังนั้นกระบวนการ Fragmentation และ Reassembly จึงถูกนำมาใช้เพื่อ ทำให้ขนาดของข้อมูลเล็กลง และสามารถส่งไปในระบบได้ และเมื่อถึงปลายทาง IP ก็จะทำการประกอบข้อมูล (reassembly) ขึ้นมาใหม่ก้อนที่จะส่งผ่านไปยังผู้ใช้ปลายทาง ดังรูปที่ 2.7



รูปที่ 2.7 internet fragmentation and reassembly

อันดับแรก IP ใน Host ต้นทางจะแยกข้อมูลของผู้ใช้ (NS-User) , NSDU เป็น Datagram ซึ่งมี Address กำกับเป็นเฉพาะส่วน ๆ ไปซึ่งจะถูกออกคำสั่งโดย Network ที่ในติดต่ออยู่ด้วยและการส่ง Datagram ไปยัง IP ใน Gateway ตัวแรก โดยที่ IP ใน Gateway จะไม่ Reassemble NSDU แต่จะปรับปรุงในขอบเขตที่เหมาะสม และส่ง Datagram ที่ได้รับตรงไปยัง Network ที่สอง (ถ้า Network ที่สองสามารถรองรับ ขนาด Datagram นี้) หรือทำการ Fragment datagram ให้มีขนาดเล็กลง ซึ่งขั้นตอนนี้จะถูกทำซ้ำที่ Gateway ตัวต่อไป Network ตัวสุดท้ายสามารถรองรับขนาดของ packet ได้มากกว่า packet ที่มันได้รับข้อมูลจึงถูกส่งได้โดยตรง โดยที่จะมีการปรับปรุงในบางส่วน Header ของ Datagram เท่านั้น จากนั้น IP ใน Host ปลายทางจะทำการประกอบข้อมูล (Reassemble) ที่มันได้รับขึ้นมาใหม่และส่งผลที่ได้รับก็คือ NSDU ไปยังผู้ใช้ (NS-user)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการคิดค่าเวลาสูงสุดที่ Host ต้นทางกำหนดให้ Gateway รอ Datagram(NSDU) ระหว่างแต่ละการ Assembly ซึ่งก็คือ time-to-live ซึ่งจะถูกตั้งค่าโดย IP ใน Host ต้นทาง ซึ่งจะทีค่าลดลงเรื่อย ๆ ในแต่ละขั้นตอนของการ Process datagram ถ้า Datagram ถูก Fragment ค่าปัจจุบันจะถูกนำไปใส่ในส่วน Header ชอง Datagram ตัวใหม่ ถ้ามันถึงค่า 0 ที่จุดใด ๆ ระหว่างการ Process ใน Gateway (หรือ Host) การ Reassembly ก็จะล้มเหลวและทุก ๆ การ Fragment ที่เกี่ยวกับ NSDU ก็จะถูกตัดทิ้ง

ค่า Time-to-live ในแต่ละ Datagram จะเป็นจำนวนเท่าของวินาที โดยที่จำนวนของมันจะถูกลดลงโดยแต่ละ IP ซึ่งจะเปลี่ยนแปลงไปตามค่าเวลาจริงในการส่งถ่ายข้อมูลของ Network ที่ติดต่อด้วย

2.5.4 การเลือกเส้นทาง (Routing)

ในแต่ละ Network (หรือ subnet) ใน Internet จะมีชนิดของ PA address ที่แตกต่างกัน ซึ่งระบบ (system Host) หรือ gateway ที่ถูต่อเข้ากับ network จะสามารถส่ง datagram ไปยังระบบอื่นได้โดยตรงเฉพาะ network ที่เหมือนกันเท่านั้น ในการเลือกเส้นทาง (routing) ให้ datagram ข้ามไปยังหลาย ๆ network IP ในแต่ละ internetwork gateway ต้องรู้ PA address ของ host ปลายทาง

ซึ่งมี 2 วิธีการพื้นฐานที่ถูกใช้ในการหาเส้นทางภายใน Internet 8nv centralized และ distributed ด้วยวิธีการ centralized routing ข้อมูลเกี่ยวกับการเลือกเส้นทาง ที่เกี่ยวข้องกับแต่ละ gateway จะถูก download จาก site ส่วนกลางโดยใช้ข้อมูล network และ special network management จะพยายามตรวจสอบ network และ host ที่ถูกเพิ่มเข้า และถอดออก และข้อบกพร่องที่จะถูกวินิจฉัยและถอกออกปละข้อบกพร่องที่จะถูกวินิจฉัยและตรวจสอบ

ด้วยวิธีการ Distributed routing ทุก ๆ host ปละ gateway จะร่วมกันในการแบ่งปัน วิธีการในการรับประกันว่า ข้อมูลเกี่ยวกับการเลือกเส้นทางจะถูกจดจำไว้โดยแต่ละระบบ ในรูปของ routing table ซึ่งจะมี NPA address ไว้ให้ในการส่งแต่ละ datagarm ซึ่ง Internet จะใช้วิธีการแบบนี้

ขั้นตอนการ Routing ที่เกี่ยวกับ IP ขั้นตอนแรกจะอ่าน IP address (NSAP) ปลายทางจากภายใน datagram และใช้มันในการหาการตอบสนอง PA address ของ host หรือ gateway จาก routing table ในส่วนที่เพิ่มเติมชุดของ routing protocol จะถูใช้เพิ่มและรักษาส่วนที่อยู่ในแต่ละ routing table ในแบบของ distributed ซึ่งรูปแบบทั่วไปที่ถูกใช้ภายใน host IP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
หลักการเขียนโปรแกรม Winsock ที่ใช้กับ TCP/IP

ในช่วง 5 ปีที่ผ่านมามี Programmer จำนวนไม่มากที่ได้รับข่าวของ TCP/IP ซึ่งเป็นชุด Protocol ไว้ใช้ในการส่งผ่านเครือข่าย Internet สำหรับงาน Internet Communication และไม่มีระบบปฏิบัติการใดที่จะสามารถเอาใจใส่ทำให้สมบูรณ์ในเรื่องของ Protocolแม้แต่ระบบปฏิบัติการ Windows ก็ไม่ละเว้น

Application Programming ที่ส่วนมากใช้กันอย่างกว้างขวางสำหรับ TCP/IP Programmer คือ Windows Sockets Library ซึ่งได้ถูกใช้ในการส่งออกที่ Internet สำหรับการใช้สิ่งแวดล้อม TCP/IP Application และเป็นธรรมดาอยู่แล้วเมื่อเวลาผ่านมา Microsoft ได้คิดค้น API สำหรับ TCP/IP ภายใต้ระบบปฏิบัติการของ Windows

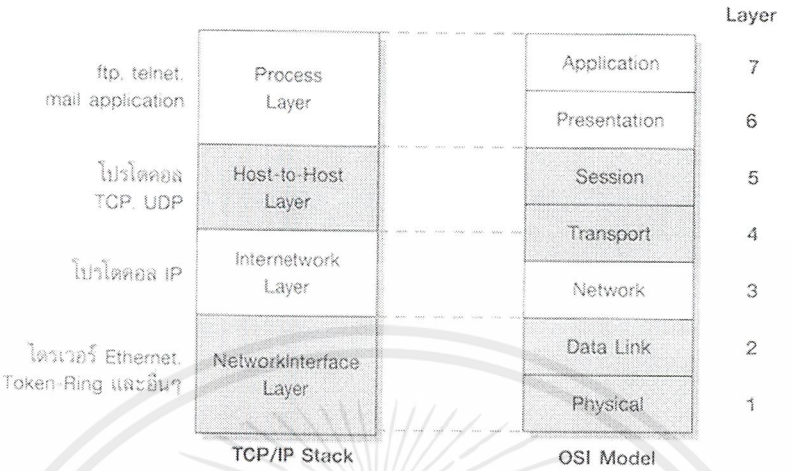
ในการใช้งานของ Windows Sockets จะไม่ถูกจำกัดในเรื่องของ TCP/IP Programming Windows Sockets จะแสดงให้เห็นว่าแม้การติดต่อที่เป็นงานหนัก เมื่อใช้ TCP/IP หรือ Protocol อื่นๆ สำหรับการส่งค่าจริง ๆ ของข้อมูลอย่างไรก็ตามในปัจจุบัน Winsock Implementation จะจำกัดการใช้งานของ Windows Sockets ในการติดต่อกับ TCP/IP

โปรโตคอล TCP/IP มีกลไกการทำงานเป็นชั้นหรือ Layer เรียงต่อกัน โดยในแต่ละ Layer จะมีการทำงานเทียบได้กับ OSI model มาตรฐาน แต่บาง Layer ของโปรโตคอล TCP/IP จะทำงานเทียบกับ OSI หลาย Layer ปนกัน ซึ่งในแต่ละ Layer ของโปรโตคอล TCP/IP จะประกอบด้วย

- Process layer
- Host-to-Host layer
- Internetwork layer
- Network Interface layer

โดยเมื่อเทียบกับมาตรฐาน OSI model แล้วจะเป็นดังรูปที่ 3.1 ซึ่งเราจะเห็นว่าบางกลไกของโปรโตคอล TCP/IP เทียบได้กับมาตรฐาน OSI model สองชั้น หรือบางกลไกก็จะทำงานคาบเกี่ยวกันระหว่างบางชั้นของ OSI model ตัวอย่างเช่น กลไกการทำงานของโปรโตคอล TCP/IP ในส่วน Network Interface layer เมื่อเทียบกับมาตรฐาน OSI model จะเทียบได้กับ Data Link layer และ Physical layer 2 ชั้นรวมกัน เป็นต้น ในแต่ละกลไกของโปรโตคอล TCP/IP จะมีโปรโตคอลอื่นๆ ในชุดของ TCP/IP ร่วมทำงานอยู่ด้วย ซึ่งจะกล่าวโดยละเอียดต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1 แสดง TCP/IP stack เปรียบเทียบกับมาตรฐาน OSI

3.1 Process layer

จากรูปแสดงลำดับชั้นการทำงานของโปรโตคอล TCP/IP เทียบกับมาตรฐาน OSI model นั้น ในชั้นบนสุดเรียกว่า Process layer ทำงาน 2 หน้าที่เทียบได้กับ Application layer และ Presentation layer ในชั้นนี้จะรองรับการทำงานของแอพพลิเคชันต่างๆ ที่ทำงานเป็นโปรเซส อยู่ในเครื่องเซิร์ฟเวอร์ให้บริการและเครื่องที่ขอใช้บริการ หรือ ไคลเอนต์ (client) ซึ่งจะติดต่อกันผ่านโปรโตคอลเฉพาะแอพพลิเคชั่นอีกทีหนึ่ง ตัวอย่างเช่น เมื่อผู้ใช้งานอินเตอร์เน็ตต้องการโอนถ่ายไฟล์หรือ download ข้อมูลจากเครื่องเซิร์ฟเวอร์ที่ให้บริการ โดยอาจจะเรียกใช้โปรแกรม ftp client ทั่วไป เช่นโปรแกรม WS_ftp ติดต่อกับโปรเซส ftp ที่กำลังให้บริการอยู่ที่เครื่องเซิร์ฟเวอร์ จากนั้นตัวโปรเซส ftp ก็จะเรียกใช้โปรโตคอล FTP (ftp Transfer Protocol) เพื่อทำการโอนถ่ายไฟล์นี้ หรือถ้าผู้ใช้ต้องการเรียกใช้งานคอมพิวเตอร์เครื่องที่อยู่ห่างไกลออกไปด้วยการใช้โปรแกรม telnet ที่เครื่องเซิร์ฟเวอร์ให้บริการ ตัวโปรเซส telnet ที่อยู่ก็จะเรียกใช้โปรโตคอล Telnet เพื่อติดต่อกันหรือในกรณีที่มีการเรียกใช้โปรแกรม web browser เช่น Netscape Navigator เพื่อเรียกดูเว็บเพจในเว็บไซท์ CNN ที่เครื่องซึ่งให้บริการเว็บของ CNN ก็จมีโปรเซส HTTP (HyperText Tranfer Protocol) ทำงานอยู่และจะติดต่อกับผู้ใช้ผ่านโปรโตคอล HTTP เป็นต้น

การทำงานของแอพพลิเคชั่นต่างๆ จะอยู่ที่ Process layer นี้ และมีการติดต่อกันตามแต่โปรโตคอลเฉพาะแล้วแต่แอพพลิเคชั่นที่ใช้งาน จากการที่ Process layer ของ TCP/IP รองรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ให้โปรโตคอลอื่นทำงานได้หลายโปรเซสและหลายโปรโตคอลได้พร้อมกันนั้น ทำให้ผู้ใช้สามารถเปิดโปรแกรมใช้งานได้หลายๆอย่างพร้อมกัน เช่น เปิดโปรแกรม Internet Explorer เพื่อเรียกดูเว็บเพจ พร้อมกับใช้งานโปรแกรม Outlook Express เพื่อรับส่งอีเมล์ไปพร้อมกันได้โดยไม่ต้องรอให้ทำงานอย่างหนึ่งอย่างใดให้เสร็จก่อนหรือในปัจจุบันมีการพัฒนาโปรแกรม web browser ให้สามารถเรียกใช้งานโปรโตคอลอื่นๆ ได้มากขึ้น ทำให้เราสามารถใช้โปรแกรม web browser โอนถ่ายไฟล์ข้อมูลที่ใช้โปรโตคอล FTP ได้โดยไม่ต้องไปหาโปรแกรมอื่นมาใช้

โปรโตคอลหลักๆที่ทำงานใน Process layer ซึ่งผู้ใช้มักจะคุ้นเคยกันดีได้แก่ FTP (File Transfer Protocol) ,Telnet,HTTP (HyperText Transfer Protocol) และ SMTP (Simple Mail Transfer Protocol) นอกจากนี้ยังมีโปรโตคอลอื่นมีอยู่เบื้องหลัง ซึ่งทำงานโดยที่ผู้ใช้ไม่สามารถมองไม่เห็นได้จากโปรแกรมหรือไม่ได้มีการใช้งานโดยตรง เช่น

- โปรโตคอล DNS (Domain Name System) ที่ทำหน้าที่แปลงข้อมูลชื่อ domain name หรือชื่อเว็บไซท์ทั้งหลายให้เป็นหมายเลข IP address
- โปรโตคอล SNMP (Simple Network Management Protocol) ใช้ในการควบคุมและตรวจสอบอุปกรณ์ที่อยู่ในเครือข่าย
- โปรโตคอล DHCP (Dynamic Host Configuration Protocol) ทำหน้าที่แจกจ่ายข้อมูลพารามิเตอร์ของเครือข่ายให้กับเครื่องลูกข่ายที่เชื่อมต่ออยู่
-

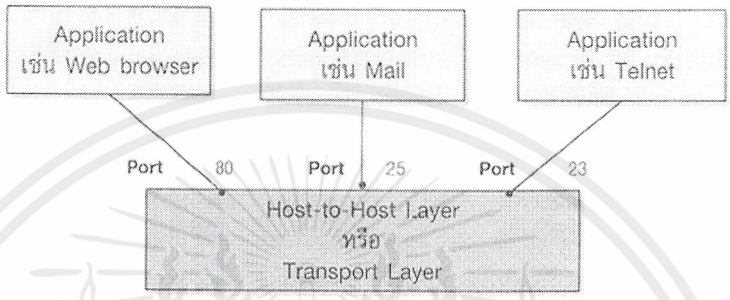
3.2 Host-to-Host layer

ผู้ที่ใช้งานอินเตอร์เน็ตเคยสงสัยหรือไม่ว่าเครื่องเซิร์ฟเวอร์ที่ให้บริการต่างๆ เช่น เว็บเซิร์ฟเวอร์นั้น เมื่อมีผู้เข้ามาเรียกใช้บริการพร้อมกันหลายคน จะมีวิธีการส่งข้อมูลกลับไปยังต้นทางได้อย่างไรโดยไม่ผิดพลาด ซึ่งบางครั้งผู้ใช้รายหนึ่งอาจจะเปิดโปรแกรม web browser ซ้อนกันเพื่ออ่านข้อมูลจากเว็บเพจอื่นๆ พร้อมกันไปได้ ดังนั้นระบบจะทราบได้อย่างไรถึงการจัดส่งข้อมูลจากเว็บเพจอื่นๆพร้อมกันไปได้ ดังนั้นระบบจะทราบได้ไรอย่างถึงการจัดส่งข้อมูลได้อย่างถูกต้อง

การทำงานที่ชั้นของ Host-to-Host layer นี้จะมีบทบาทในการจัดการต่อจาก Process layer บางครั้งเรามีกเรียกชั้น Host - to -Host ว่าเป็น Transport layer ซึ่งไม่ใช่ชั้นของ Transport layer ในมาตรฐาน OSI model การทำงานของ Host-to-Host layer นี้จะมีการสร้าง connection หรือการเชื่อมต่อกันระหว่างแอพพลิเคชั่น กับ Host-to-Host layer โดยจุดที่เชื่อมกันเพื่อรับส่งข้อมูลนี้เรียกว่า port หรือ socket (คำว่า port ในที่นี้ไม่ได้หมายถึง port ทางฮาร์ดแวร์) และในแต่ละ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แอพพลิเคชั่นก็จะสร้างการเชื่อมต่อผ่าน port ได้พร้อมกันหลายแอพพลิเคชั่น ซึ่งการใช้งาน port ของแต่ละแอพพลิเคชั่นที่อยู่ในชั้น Process layer จะแตกต่างกันตามหมายเลขที่กำหนดไว้ และแต่ละโปรโตคอลจะมีการใช้งาน port หมายเลขต่างๆไม่ซ้ำกัน ตามรูปที่3.2



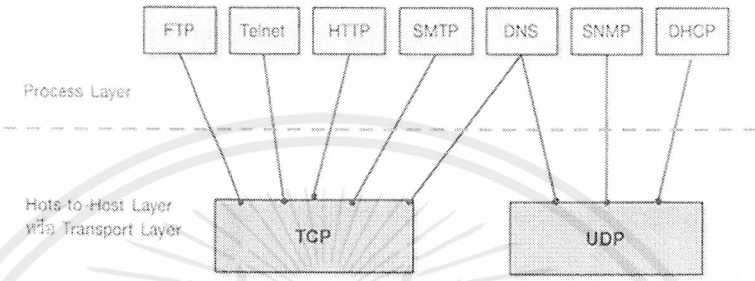
รูปที่ 3.2 แอพพลิเคชั่นหรือโปรเซสต่างๆ สื่อสารกับ Host-to-Host layer ผ่านจุดเชื่อต่อหรือ port ส่วนหมายเลขในรูปคือหมายเลข port ที่โปรเซสใช้งานเช่น เว็บหรือโปรเซส http ใช้งาน port 80 ในการส่งผ่านข้อมูล เป็นต้น

เมื่อแอพพลิเคชั่นทำงานผ่านโปรโตคอลในชั้น Process layer จะมีการส่งผ่านข้อมูลไปยัง Host-to-Host layer ที่ชั้นนี้จะมีการเชื่อมต่อผ่าน port ที่กำหนด ทำให้การรับส่งข้อมูลในแต่ละโปรโตคอลทำได้ถูกต้อง ถึงแม้ว่าในเครื่องเซิร์ฟเวอร์ที่ให้บริการจะมีการทำงานอยู่หลายโปรเซสที่แตกต่างกันก็ตาม หรือมีผู้ใช้บริการเข้ามาใช้งานพร้อมกันจำนวนมากและหลายแอพพลิเคชั่นในเวลาเดียวกัน ในชั้น Host-to-Host หรือ Transport layer ของ TCP/IP นี้จะมีโปรโตคอลทำงานอยู่ 2 โปรโตคอลที่แตกต่างกันคือ โปรโตคอล TCP และโปรโตคอล UDP (User Datagram Procotol) ในการส่งผ่านข้อมูลลงไปที่ชั้นถัดๆไป เราจะเห็นว่าโปรโตคอล TCP และ UDP (User Datagram Protocol) ในการส่งผ่านข้อมูลลงไปที่ชั้นถัดๆไป เราจะเห็นว่าโปรโตคอล TCP และ UDP จะถูกผนึกเข้าไปในโปรโตคอล IP อีกทีหนึ่งและส่งต่อไปยังเครือข่ายอินเตอร์เน็ตต่อไป

ตัวโปรโตคอล TCP และโปรโตคอล UDP จะมีแอพพลิเคชั่น เฉพาะเพื่อเรียกใช้งานแยกกันคือ แอพพลิเคชั่นที่ใช้โปรโตคอล FTP,Telnet,HTTP และ SMTP จะมีการส่งผ่านข้อมูลโดยเรียกใช้โปรโตคอล TCP ส่วนแอพพลิเคชั่นที่ใช้โปรโตคอล SNMP และ DHCP จะส่งผ่านข้อมูลโดยเรียกใช้โปรโตคอล UDP และสำหรับโปรโตคอล DNS นั้นจะสามารถเรียกใช้งานได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทั้ง TCP และ UDP ดังรูป ซึ่งเหตุผลที่มีการเรียกใช้โปรโตคอล TCP และ UDP แตกต่างกัน ก็เนื่องจากวิธีการทำงานของทั้งสองโปรโตคอลต่างกันนั่นเอง



รูปที่ 3.3 โปรเซสต่างๆที่เรียกใช้ Transport layer เพื่อส่งผ่านข้อมูลโดยอาศัย port ซึ่งในแต่ละโปรเซสจะเรียกใช้งาน port เฉพาะแตกต่างกัน ยกเว้น DNS ที่สามารถใช้งานได้ทั้ง TCP และ UDP

3.3 โปรโตคอล TCP

โปรโตคอล TCP (Transmission Control Protocol) เป็นโปรโตคอลที่มีการรับส่งข้อมูลแบบ stream oriented protocol หมายความว่า การรับส่งข้อมูลจะไม่คำนึงถึงปริมาณข้อมูลที่จะส่งไป แต่จะแบ่งข้อมูลเป็นส่วนย่อยๆ ก่อน แล้วจึงจะส่งไปยังปลายทางอย่างต่อเนื่องเป็นลำดับข้อมูล ในกรณีที่ข้อมูลส่วนใดส่วนหนึ่งสูญหายไป ก็จะส่งข้อมูลส่วนนั้นใหม่อีกครั้ง สำหรับปลายทางก็จะทำหน้าที่จัดเรียงส่วนของข้อมูล datagram ใหม่ให้ต่อเนื่องกันและประกอบกลับเป็นข้อมูลทั้งหมดได้ ซึ่งจะแยกข้อมูลที่ไม่ถูกต้องออกดังนั้นแอพพลิเคชั่นหรือโปรเซสเซอร์ใดที่อาศัยการส่งผ่านข้อมูลด้วยโปรโตคอล TCP จะต้องใช้หน่วยความจำและขนาดของช่องสัญญาณ (bandwidth) มากกว่า UDP

การติดต่อระหว่างกันจะต้องเป็นแบบ connection-oriented คือต้องมีการสร้างการติดต่อกันเป็น session ทั้ง 2 ด้านเสียก่อน แล้วจึงจะรับส่งข้อมูลไปได้พร้อมกัน (full duplex) เหมือนกับการใช้โทรศัพท์ติดต่อกัน เมื่อผู้ติดต่อต้นทางเรียกให้ฝ่ายตรงข้ามรับสายแล้ว จึงเริ่มสนทนา เช่น พูดคำว่า สวัสดี หรือ ฮัลโหล กันก่อนเพื่อให้แน่ใจว่าฝ่ายตรงข้ามพร้อมจะติดต่อด้วย จากนั้นจึงเริ่มต้นติดต่อกัน และเมื่อต้องการจะเลิกการติดต่อก็จะมีการพูดคำว่า “สวัสดี” .ให้ฝ่ายตรงข้ามทราบว่าจะเลิกการติดต่อและวางสายไป ซึ่งในระหว่างการติดต่อกันนั้น แม้ว่าฝ่ายใด

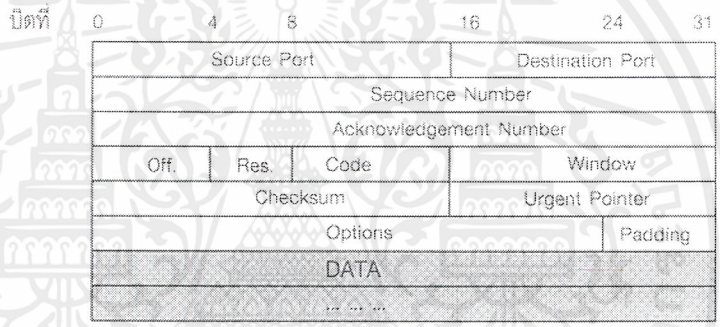
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ฝ่ายหนึ่งหรือทั้งสองฝ่ายจะเงียบไป คือไม่พูดอะไรเป็นเวลานานๆ แต่การเชื่อมโยงระหว่างทั้งสองด้านยังคงมีอยู่ไม่ขาดไปจนกว่าฝ่ายหนึ่งจะวางสาย เช่นเดียวกับการติดต่อกันด้วยกลไกโปรโตคอล TCP เมื่อแอพพลิเคชั่นต้องการส่งผ่านข้อมูลจะใช้โปรโตคอลที่เหมาะสมในชั้น Process layer ติดต่อไปและมีการสร้างช่องส่งข้อมูลผ่าน port ที่กำหนดเพื่อส่งผ่านข้อมูลไปยังโปรโตคอล TCP

ในระหว่างการรับส่งข้อมูลนี้ โปรโตคอล TCP จะเพิ่มขบวนการสอบทานข้อมูลเพื่อให้ข้อมูลมีความถูกต้องไม่ผิดพลาดไปจากเดิมโดยการส่งสัญญาณสอบทานข้อมูล (acknowledgement) และส่งข้อมูลให้ใหม่อีกครั้ง ถ้าปลายทางไม่ได้รับหรือเกิดความผิดพลาดขึ้น

ความน่าเชื่อถือของการส่งผ่านข้อมูลโดยโปรโตคอล TCP จะมีมากกว่า แต่ก็ต้องอาศัยทรัพยากรของระบบมากว่าในการทำงานเช่นกัน



รูป 3.4 รูปแบบของ TCP packet จะเห็นว่ามีฟิลด์ Acknowledgement Number และข้อมูล Checksum เพื่อใช้ตรวจสอบการเดินทางของข้อมูล ส่วน header มีข้อมูลมากทำให้ต้องอาศัยทรัพยากรของระบบทำงานมาก

3.4 โปรโตคอล UDP

ใน Host-to-Host layer นอกจากจะมีโปรโตคอล TCP ทำงานแล้ว ก็ยังมีโปรโตคอล UDP (User Datagram Protocol) ที่มีคุณสมบัติแตกต่างกันอยู่ด้วยในการรับส่งข้อมูลผ่านโปรโตคอล UDP จะเป็นแบบที่ทั้งสองด้านไม่จำเป็นต้องอาศัยการสร้างช่องทางเชื่อมต่อกัน (connectionless) ระหว่างเครื่องเซิร์ฟเวอร์ให้บริการกับเครื่องที่ขอใช้บริการ โดยไม่ต้องแจ้งให้ฝ่ายรับข้อมูลเตรียมรับข้อมูลเหมือนโปรโตคอล TCP และไม่มีการตรวจสอบความถูกต้องครบถ้วนในการรับส่งข้อมูลนั้นๆ ด้วย เนื่องจากโปรโตคอล UDP ไม่มีสัญญาณสอบทานข้อมูล (acknowledgement) ในการส่งข้อมูลแต่ละครั้งและไม่มีการส่งข้อมูลใหม่อีกในกรณีที่เกิดความผิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
36939

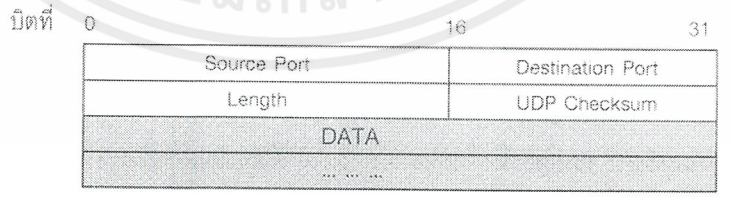
พลาดของการส่งข้อมูล เมื่อเป็นเช่นนี้แอพพลิเคชั่นหรือโปรเซสใดที่ต้องอาศัยโปรโตคอล UDP ในการส่งผ่านข้อมูลก็อาจจะต้องสร้างขบวนการตรวจสอบข้อมูลขึ้นมาเอง

ตามรูป 3.3 จะเห็นว่าโปรโตคอลชั้นบนขึ้นไป ที่ใช้การส่งผ่านข้อมูลโดยโปรโตคอล UDP เช่น โปรโตคอล SNMP (ใช้ควบคุมและจัดการอุปกรณ์ในเครือข่าย) หรือโปรโตคอล DHCP (ใช้ส่งข้อมูลพารามิเตอร์ของเครือข่ายให้กับเครื่องลูกข่ายได้ใช้งาน) การส่งข้อมูลเหล่านั้นไม่ต้องรับทราบหรือตรวจสอบว่าข้อมูลไปถึงปลายทางถูกต้องหรือไม่ แต่กลไกการตรวจสอบข้อมูลที่มีการรับส่งจะไปทำในขั้นตอนของโปรโตคอลชั้นที่สูงกว่าแทน

ตัวอย่างขั้นตอนกลไกการทำงานโดยใช้โปรโตคอล UDP มีดังต่อไปนี้

1. ในขั้นของ Process layer เมื่อโปรแกรมควบคุมอุปกรณ์เครือข่ายเช่น โปรแกรม Network management ต้องการส่งข้อมูลไปยังอุปกรณ์ที่ต้องการ แอพพลิเคชั่นนั้นจะติดต่อผ่านโปรโตคอล SNMP ในชั้น Process layer
2. โปรโตคอล SNMP จะติดต่อกับโปรโตคอล UDP ในชั้นถัดไป เพื่อขอติดต่อผ่าน port ที่กำหนด
3. โปรโตคอล SNMP เตรียมข้อมูลที่จะส่ง รวมทั้งที่อยู่ปลายทาง
4. โปรโตคอล SNMP ส่งผ่านข้อมูลให้โปรโตคอล UDP ที่อยู่ในชั้น Host-to-Host layer
5. โปรโตคอล UDP ทำหน้าที่ผนึกข้อมูลหรือ datagram นั้น ไปกับโปรโตคอล IP ในชั้นถัดลงไป เพื่อส่งข้อมูลออกจากเครื่อง

ซึ่งจะเห็นว่ามีกลไกที่ต่างจากการส่งข้อมูลด้วยโปรโตคอล TCP ซึ่งจะต้องมีการติดต่อกันก่อน และทั้งสองฝ่ายรับทราบการรับส่งข้อมูลของช่องการส่งข้อมูลนั้น



รูปที่ 3.5 รูปแบบของ UDP packet จะมีฟิลด์ข้อมูลส่วน header น้อยมากและไม่มีข้อมูลส่วนการตรวจสอบข้อมูล ทำให้ UDP packet มีขนาดเล็กและใช้หน่วยความจำหรือทรัพยากรของระบบน้อย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 Internetwork Layer

ในระดับล่างต่อมาในชั้น Internetwork layer มีหน้าที่ส่งผ่านข้อมูลในระหว่างเครือข่ายโดยมีโปรโตคอลที่ทำงานเป็นกลไกสำคัญในการส่งผ่านข้อมูลไปยังเครือข่ายใดๆบนอินเตอร์เน็ต คือ โปรโตคอล IP (Internet Protocol) นอกจากนี้ในชั้น Internetwork layter ยังมีโปรโตคอลทำงานอยู่ด้วยอีก 2 ชนิดคือ โปรโตคอล Internet control Message Protocol (ICMP) และโปรโตคอล Address Resolution Protocol (ARP)

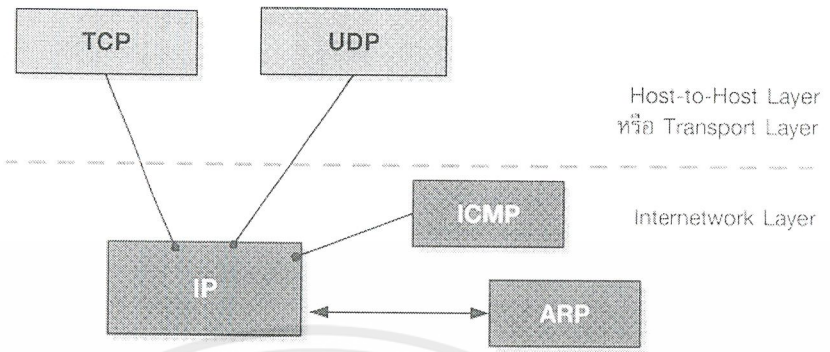
3.6 โปรโตคอล IP

โปรโตคอล IP ทำหน้าที่ให้บริการส่งผ่านข้อมูลที่มาจาก Host-to-Host layer เพื่อส่งข้ามไปยังเครือข่ายใดๆได้อย่างถูกต้อง แม้ว่าจะมีเครือข่ายเชื่อมต่อกันอยู่ในอินเตอร์เน็ตเป็นล้านๆ เครือข่ายก็ตาม เนื่องจากโปรโตคอล IP มีข้อมูลตำแหน่ง IP ปลายทางที่จะส่งข้อมูลไปให้โดยทำงานร่วมกับอุปกรณ์ Router เพื่อส่งข้อมูลข้ามเครือข่ายออกไปได้ ตัวโปรโตคอล IP จะทำงานแบบ packet switching คือมีการส่งข้อมูลผ่านสวิทช์ (switch) ไปยังปลายทาง ตัววงจรผ่านหรือ switch นี้อาจเป็น Gateway หรือ Router ในระบบเครือข่ายก็ได้ ซึ่งในข้อมูลของโปรโตคอล IP จะมีข้อมูลของหมายเลข IP ปลายทางที่จะส่งข้อมูลไปและเมื่อถึงเครือข่ายปลายทางแล้ว จะมีกลไกแปลงหมายเลข IP ให้เป็นหมายเลขฮาร์ดแวร์ ประจำเครื่องที่ถูกต้องอีกทีหนึ่งด้วยโปรโตคอล ARP ตามรูป 3.6 ที่จะแสดงการติดต่อกันระหว่างโปรโตคอลในชั้นของ Host-to-Host layer และ Internetwork layer

3.7 โปรโตคอล ICMP

หน้าที่หลักของโปรโตคอล ICMP (Internet Control Message Protocol) คือการแจ้งหรือแสดงข้อความจากระบบ เพื่อบอกให้ผู้ใช้ทราบว่าเกิดอะไรขึ้นในการส่งผ่านข้อมูลนั้น ซึ่งปัญหาส่วนมากที่พบคือส่งไปไม่ได้ หรือปลายทางรับข้อมูลไม่ได้ เป็นต้น นอกจากนี้โปรโตคอล ICMP ยังถูกเรียกใช้งานจากเครื่องเซิร์ฟเวอร์และ Router อีกด้วยเพื่อแลกเปลี่ยนข้อมูลที่ใช้ควบคุม ส่วนรูปแบบการทำงานของโปรโคคอล ICMP นั้นจะทำควบคู่กับโปรโตคอล IP ในระดับเดียวกัน และข้อความต่างๆ ที่แจ้งให้ทราบจะถูกผนึกอยู่ภายในข้อมูลของ IP (IP datagram) อีกทีหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 โปรโตคอล TCP และ UDP อาศัยโปรโตคอล IP ที่อยู่ชั้นล่างเพื่อส่งผ่านข้อมูลระหว่างเครื่องข่าย และในชั้น Internetwork Protocol ยังมีโปรโตคอล ICMP ทำหน้าที่ส่งข้อความแจ้งเตือนและโปรโตคอล ARP ทำหน้าที่แปลงเลขหมาย IP ไปเป็นเลขหมายของฮาร์ดแวร์จริง

ข้อความที่โปรโตคอล ICMP ส่งนั้นแบ่งออกได้ 2 แบบคือ ICMP error message หรือข้อความแจ้งข้อผิดพลาด และ ICMP query หรือข้อความเรียกขอข้อมูลเพิ่ม ตัวอย่างกลไกการทำงานขอโปรโตคอล ICMP เช่นเมื่อ มีการส่งผ่านข้อมูลจากผู้ใช้ไปยังปลายทางที่ไม่ถูกต้อง หรือขณะนั้นเครื่องปลายทางเกิดปัญหาจนไม่สามารถรับข้อมูลได้ที่ Router จะส่งข้อความแจ้งเป็น ICMP message ที่ชื่อ destination unreachable ให้กับผู้ส่งข้อมูล นอกจากนี้ตัวข้อมูลที่แจ้งข้อความก็จะมีส่วนข้อมูล IP datagram ที่เกิดปัญหาด้วย ดังนั้นเมื่อผู้ส่งข้อมูลได้รับข้อความแจ้งแล้วก็จะทราบได้ว่าจุดที่เกิดปัญหานั้นอยู่ที่ใด

ดังนั้นโปรโตคอล ICMP จึงกลายมาเป็นเครื่องมืออย่างหนึ่งในการช่วยทดสอบเครือข่าย เช่น คำสั่ง ping ที่เรามักใช้ทดสอบว่าเครื่องเซิร์ฟเวอร์ที่ให้บริการหรืออุปกรณ์ที่ต่ออยู่ในเครือข่ายอินเตอร์เน็ตนั้นยังทงานเป็นปกติหรือไม่ แล้วคำสั่ง ping มีการเรียกใช้งานโปรโตคอล ICMP แจ้งเป็นข้อความให้ทราบอีกต่อหนึ่ง

3.8โปรโตคอล ARP

โปรโตคอล ARP (Address Resolution Protocol) ถูกเรียกใช้งานโดยโปรโตคอล IP เพื่อช่วยแปลงหมายเลข IP ไปเป็นหมายเลขฮาร์ดแวร์ปลายทาง ตัวอย่างเช่น เว็บเซิร์ฟเวอร์เครื่องหนึ่งเชื่อมต่ออยู่ในเครือข่ายอินเตอร์เน็ต และในการเชื่อมต่อนี้ต้องอาศัย Network Interface Card

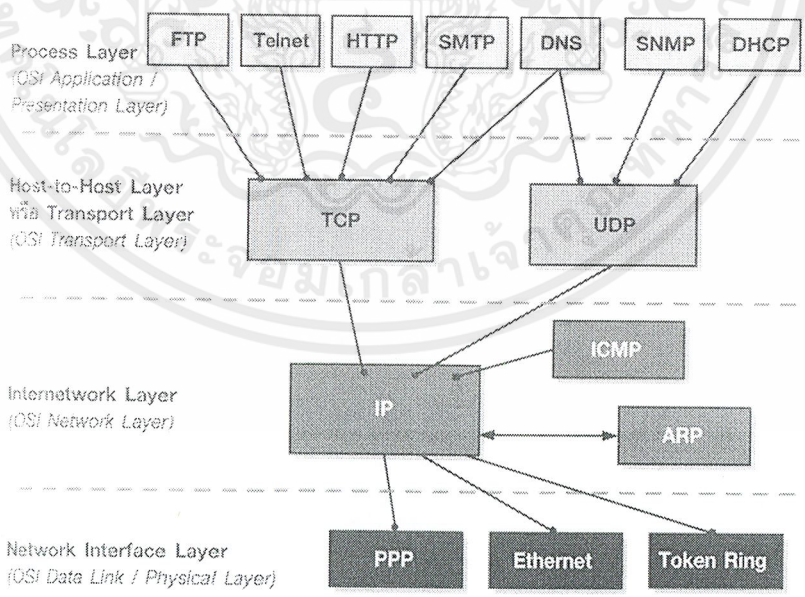
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(NIC) หรือ LAN card ติดตั้งอยู่ ที่ LAN card นี้เองจะมีหมายเลขเฉพาะประจำฮาร์ดแวร์ที่ไม่ซ้ำกับใคร เพื่อใช้อ้างอิงการส่งข้อมูลในเครือข่าย แต่เมื่อมาใช้งานในโปรโตคอล TCP/IP ก็จะต้องมีการกำหนดหมายเลข IP address ประจำตัวเพื่อใช้อ้างอิงกัน และโปรโคคอล ARP จะทำหน้าที่แปลงค่าหมายเลข IP ให้เป็นหมายเลขฮาร์ดแวร์จริงให้ในระดับการทำงานที่ Internetwork layer นี้ซึ่งกลไกการแปลงนี้เรียกว่า address resolution

3.9Network Interface Layer

เนื่องจากในด้านกายภาพของเครือข่ายนั้น มีหลายวิธีการและหลายรูปแบบในการเชื่อมต่อระบบให้เป็นเครือข่าย แต่อย่างไรก็ตามในเครือข่ายอินเตอร์เน็ตนี้ ข้อมูลหรือ IP datagram จะถูกถ่ายทอดและส่งผ่านไปยังปลายทางโดยไม่คำนึงถึงรูปแบบการเชื่อมต่อทางกายภาพ ไม่ว่าจะเป็นการใช้เครือข่ายใยแก้วนำแสงหรือเครือข่ายสาย Unshielded Twist Pair (UTP) เชื่อมต่อเป็นแบบเครือข่าย Ethenet ธรรมดาหรือเครือข่าย Token Ring ,ATM,ISDN ฯลฯ ก็ตาม

การทำงานระดับล่างสุดต่อจาก Internetwork layer จะเป็นการแปลงข้อมูล IP datagram ให้อยู่ในรูปที่เหมาะสม และแปลงเป็นสัญญาณไฟฟ้าส่งไปยังเครือขายต่อไปซึ่งในชั้น Network Interface layer นี้เมื่อเทียบกับมาตรฐาน OSI model แล้วจะเป็นการรวม 2 layer เข้าด้วยกันคือ Data link layer และ Physical layer กล่าวโดยสรุปคือการทำงานในชั้นต่างๆ ตามโครงสร้างของโปรโตคอล TCP/IP จะมีลักษณะดังรูปที่3.7



รูปที่ 3.7 โครงสร้างของโปรโตคอล TCP/IP ในแต่ละชั้นหรือ layer จะมีโปรโตคอลหลักทำหน้าที่ต่างๆและส่งผ่านข้อมูลไปยังเครือข่ายและออกสู่อินเตอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กล่าวโดยสรุปก็คือ โปรโตคอล TCP/IP ทำงานโดยแบ่งเป็นชั้นเทียบกับ OSI model ได้ กลไกในการทำงานของโปรโตคอล TCP/IP มี 4 ชั้น ซึ่งในขั้นแรก คือ Process layer ทำหน้าที่ติดต่อกับแอพพลิเคชั่นและโปรโตคอลที่แอพพลิเคชั่นนั้นๆใช้งาน และส่งต่อมาให้ชั้น Host-to-Host layer เพื่อติดต่อกันระหว่างเครื่องเซิร์ฟเวอร์ให้บริการกับเครื่องผู้ขอใช้บริการ ในชั้นนี้จะมีการสร้าง Session หรือการเชื่อมต่อระหว่างระบบขึ้นตามแต่ละโปรโตคอลที่ต้องการ ต่อมาเป็นการผนึกข้อมูลไปเป็น IP datagram ที่ชั้น Internetwork layer โดยอาศัยโปรโตคอล IP เพื่อให้สามารถติดต่อส่งข้อมูลข้ามเครือข่ายไปยังเครือข่ายและเครื่องที่ถูกต้องได้ และสุดท้ายการส่งข้อมูลออกสู่โลกภายนอกต้องอาศัยกลไกในชั้น Network Interface layer เพื่อแปลงข้อมูลใหม่ เพิ่มข้อมูลใหม่เพิ่มข้อมูลที่จำเป็นในการอ้างอิงตำแหน่งและแปลงข้อมูลเป็นสัญญาณไฟฟ้าส่งออกไปเครือข่าย และอาจจะออกไปยัง Gateway หรือ Router เพื่อข้ามเครือข่ายออกไปยังเส้นทางที่กำหนดไว้ในอินเตอร์เน็ตต่อไปคอลหรือมากกว่า ในแต่ละโปรโตคอลเหล่านี้ก็จะรับผิดชอบหน้าที่ของตนเพื่อผ่านข้อมูลลงไปยังระดับล่าง และออกสู่เครือข่ายอินเตอร์เน็ตในที่สุด

ตารางสรุปหมายเลขบางส่วนของ port ที่ใช้งานโดย TCP และ UDPเราจะเห็นว่าในแต่ละชั้นของโครงสร้าง TCP/IP Stack มีการใช้งานโปรโตคอลต่างอยู่หนึ่งโปรโตคอล

โปรโตคอลที่ใช้งาน	Port หรือ socket เชื่อมต่อ (เลขฐาน 10)	โปรโตคอลในระดับ Host-to-Host	รายละเอียด
BootP	67	UDP	BOOTstrap Protocol ด้านเซิร์ฟเวอร์
BootP	68	UDP	BOOTstrap Protocol ด้านไคลเอนต์
DHCP	67	UDP	Dynamic Host Configuration Protocol ด้านเซิร์ฟเวอร์
DHCP	68	UDP	Dynamic Host Configuration Protocol ด้านไคลเอนต์
DNS	53	UDP/TCP	Domain Name System
FTP	21	TCP	File Transfer Protocol ด้านเซิร์ฟเวอร์ที่ควบคุม
FTP	20	TCP	File Transfer Protocol ด้านเซิร์ฟเวอร์ที่ส่งข้อมูล
HTTP	80	TCP/UDP	Hyper Text Transfer Protocol ด้านเซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรโตคอลที่ใช้งาน	Port หรือ socket เชื่อมต่อ (เลขฐาน 10)	โปรโตคอลในระดับ Host-to-Host	รายละเอียด
NetBT	138	UDP	NetBIOS datagram service
NetBT	139	TCP	NetBIOS session service
SMTP	25	TCP	Simple Mail Transfer Protocol ด้านเซิร์ฟเวอร์
SNMP	161	UDP	Simple Network Management Protocol ด้าน agent
SNMP	162	UDP	SNMP trap manager
Telnet	23	TCP	Teletype Network Protocol
TFTP	69	UDP	Trivial File Transfer Protocol
WINS	137	UDP	Windows Internet Name Service

3.10 กลไกของโปรโตคอล IP

ในการส่งผ่านข้อมูล หรือ IP datagram ไปยังเครือข่ายอินเตอร์เน็ตนั้นโปรโตคอล IP จะทำหน้าที่พิจารณาว่าปลายทางในการส่ง IP datagram นั้นจะเป็นภายในเครือข่ายของตนเอง หรือจะต้องส่งขอ้มูลข้ามเครือข่ายไปอีก โดยการพิจารณานี้โปรโตคอล IP address ปลายทางว่าส่วนที่เป็นค่าหมายเลขเครือข่าย (network address) จะเหมือนกับค่าหมายเลขเครือข่ายของ IP address ต้นทางหรือไม่ ถ้าค่าตรงกันแสดงว่าการส่งขอ้มูลอยู่ภายในเครือข่ายเดียวกัน แต่ถ้าค่าต่างกันแสดงว่าต้องส่งข้อมูลไปยังปลายทางที่อยู่คนละเครือข่ายกัน

การส่งข้อมูลภายในเครือข่ายเดียวกัน มีกลไกดังนี้

1.โปรโตคอล IP จะเรียกใช้บริการโปรโตคอล ARP (Address Resolution Protocol) เพื่อแปลงหมายเลข IP ปลายทางให้เป็นค่าหมายเลขฮาร์ดแวร์ เช่น MAC address

2.เมื่อโปรโตคอล IP ได้รับค่าหมายเลขฮาร์ดแวร์แล้ว ก็จะส่งข้อมูลนั้นไปยังฮาร์ดแวร์ที่ระบุไว้

การส่งข้อมูลข้ามเครือข่าย มีกลไกดังนี้

1. โปรโตคอล IP ตรวจสอบพบว่าหมายเลข IP address ปลายทางอยู่คนละเครือข่ายกันโดยโปรโตคอล IP จะอ่านค่า IP address ของ Router เพื่อเตรียมส่งข้อมูลไปที่ Router แทน ซึ่งในที่นี้จะมีการกำหนดเป็น default router
2. โปรโตคอล IP จะเรียกใช้บริการโปรโตคอล ARP เพื่อแปลงค่า IP Address ของ Router ให้เป็นค่าหมายเลขฮาร์ดแวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. โปรโตคอล IP ส่งข้อมูล IP datagram ไปยัง Router ที่กำหนดไว้ จากนั้น Router ส่งข้อมูลข้ามเครือข่ายไปตามขั้นตอน

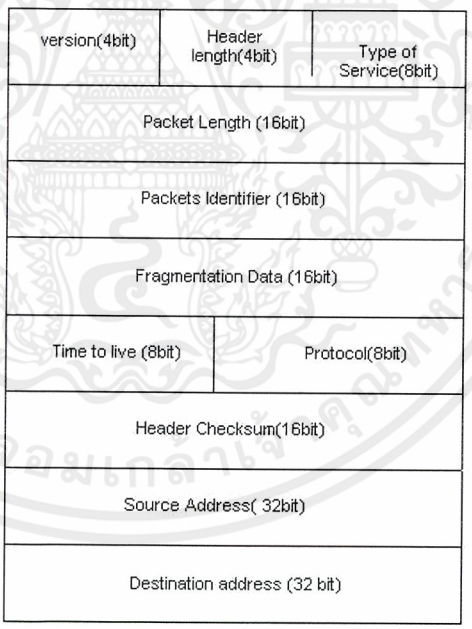
3.10.1 **IP Datagram**

หัวใจหลักของการส่งข้อมูล TCP/IP คือ IP Datagram ซึ่งจะเป็นชุดของข้อมูลซึ่งประกอบด้วย Source ,Address ปลายทาง, Type of Service Information ,User data และ Error Correction information

IP Datagram จะประกอบด้วยส่วนของ Header Block ของข้อมูลซึ่งข้อมูลสามารถที่จะกำหนดขึ้นอยู่กับชนิดของ Service และความต้องการของผู้ใช้ ในส่วน Header จะประกอบด้วยชุดของ well-Defined Field

3.10.2 **IP Header**

ในส่วนหัวของ IP Datagram หรือ IP Header โดยส่วนมากจะประกอบไปด้วย 20 bytes ตามรูปที่ 2.2 จะแสดง IP Header



รูปที่ 3.8 แสดง IP Header

ตามรูปที่ 2.1 ครั้งแรกจะพิจารณาสิ่งที่น่าสนใจในส่วนของ Protocol , Source, Address และ Destination Address Field ในส่วนของ Protocol จะถูกกำหนดวิธีของ IP Packet จะถูกแปล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทั่วๆ ไปค่าต่างๆ จะถูกกำหนดสำหรับส่วนนี้ ในส่วนของ Address จะแสดงให้เห็นส่วนของ Host Address ซึ่ง Address นี้จะเพียงหมายเลขเดียวเท่านั้นที่เหมือนกันในระบบ Internet

3.10.3 **IP Host Address and Routing**

IP Host Address จะเป็นส่วนข้อมูล 32 bits ซึ่งมีเพียงหมายเลขเดียวเท่านั้นที่เหมือนกันซึ่งใช้ในการติดต่อกับ Internet Host ส่วนของ Gateway (เป็นตัวทำหน้าที่ Interface Network ที่มีมากกว่า 1 Network) จะมี Host อยู่มากมาย

ตามหลักที่ถูกต้องคือ 4 bytes ของ Internet host address มักเขียนให้อยู่ในรูปของเลขฐานสิบ ตัวอย่างเช่น 121.0.0.1

Internet host address ส่วนมากจะถูกแบ่งออกเป็น 2 ส่วน คือ Network address และ Actual host address ถ้าเรียงตามความยาวทั้งสองตาม address โดยมากจะขึ้นอยู่กับการกำหนด bytes ใน address เสียเป็นส่วนใหญ่

Class A Network address จะสังเกตุได้ว่า byte แรกจะอยู่ระหว่าง 0 ถึง 127 และจะประกอบไปด้วย 8 bit network จะประกอบด้วย 24 bit host address เนื่องจาก address เริ่มต้นที่ 0 และ 127 จะถูกสงวนไว้ ซึ่งสามารถมีค่าสูงสุดคือ 126 Class A Subnets ใน Class A นี้สามารถมีได้ทั้งหมด 16,777,214 hosts (Address อยู่ในรูป nnn.0.0.0 และ nnn.255.255.255 จะถูกสงวนเอาไว้

Class B Netowork address จะสังเกตุได้จากตัวเลขหลักแรกสุดอยู่ระหว่าง 128 และ 191 ,Class B Network จะประกอบไปด้วย 16 bit Network address และ 16 bit host address และสามารถมี 16,383 Subnets Class B Subnet สามารถที่จะบรรจุได้ถึง 65,534 hosts (address อยู่ในรูป nnn.mmm.0.0 และ nnn.mmm.255.255 จะถูสงวนเอาไว้)

Class C Network address จะสังเกตได้จากตัวเลขหน้าสุดอยู่ระหว่าง 192 และ 223 ซึ่งจะมีได้ 2,097,152 Subnets Class C Subnets สามารถที่จะบรรจุได้ 254 hosts (โดยเริ่มที่ address nnn.mmm.bbb.0 และ nnn.mmm.bbb.255 จะถูกสงวนเอาไว้)

Class D Network address (โดยทั่วไปตัวเลข Byte แรกจะอยู่ระหว่าง 224 ถึง255) จะถูกสงวนเอาไว้เพื่อ IP Multitasking ที่ถูกจำกัดจากในส่วนของ Ip broadcastion คือ จะไม่ไปยุ่งเกี่ยวกับ Winsock Programmer

3.10.4 Host Names

ในระหว่างการคิดค้นประดิษฐ์ Internet ซึ่งตัวเลขที่จะนำมาแทน Host address ซึ่งจะมีไม่เพียงพอและอีกอย่างหนึ่งคือตัวเลขเหล่านี้เป็นการจากที่จะจดจำ และอีกอย่างถ้าเกิด host address มีการเปลี่ยนแปลงอันเนื่องมาจากหลาย ๆ สาเหตุ จึงทำให้เกิดความยุ่งยาก ดังนั้นระบบ Naming

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

System จึงได้ถูกสร้างขึ้นมาซึ่งใช้ในการ Map ตัวเลขของ IP Address ไปเป็น memories host Names

ในทุกวันนี้จะมีอยู่ 5 hosts บน Internet ทุก ๆ hosts จะถูกจับเปลี่ยนให้อยู่ในรูปของFiles กับข้อมูลของ Internet host ทุกตัวและ address อย่างไรก็ตามอินเตอร์เนตก็ได้เจริญเติบโตอย่างต่อเนื่องทำให้ดูว่าจะไม่เพียงพอ อย่างแรกคือการตั้งชื่อ internet host จะทำได้มีมาตรฐาน อย่างที่ 2 คือทางด้าน hardware สมารถที่จะแยกแยะความต้องการที่จะติดต่อกับทุกเครื่องคอมพิวเตอร์ทุกตัวที่ต่ออินเคตอร์เนตอยู่

คำตอบของปัญหานี้คือการกำหนด Domain Name System (DNS) ซึ่งจะเป็นตัวจัดระเบียบของการตั้งชื่อ (Naming System) รูปแบบของ Domain Names คือจะคล้ายกันทุกคนที่ใช้อินเตอร์เน็ต โดยปกติการตั้งชื่อจะอยู่ในรูปแบบคือ Host Subdomain Domain

ในส่วนนี้จะกำหนดโดยหน่วยงานหนึ่ง (ส่วนมากเป็นหน่วยงานในสหรัฐ) หรือ โดยแต่ละประเทศ โดยจะมีดังต่อไปนี้

GOV : Government Bodies

EDU : Education Institutions

COM : Commercial enterprises

MIL : Milliary organization

ORG : Other organization

Top –level Domain โดยใช้ชื่อประเทศ มักจะใช้ตามมาตรฐาน ISO 3166 โดยจะใช้อักษรย่อ 2 ตัวของชื่อประเทศ

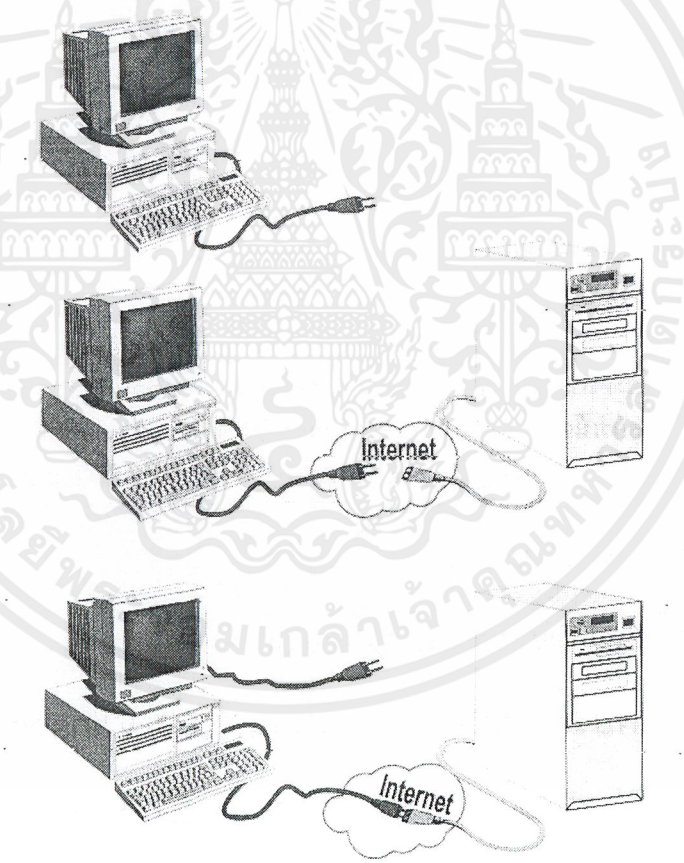
การทำงานเกี่ยวกับเครือข่ายคอมพิวเตอร์นั้นเป็นเรื่องค่อนข้างซับซ้อน จนทำให้เราสร้างข้อกำหนดต่างๆ ขึ้นมาเพื่อให้วิธีการสื่อสารระหว่างกันเป็นไปได้โดยราบรื่น แล้วก็ทำให้เกิดเป็นโพรโตคอล (Protocol) มากมายหลายชนิดมาใช้งานร่วมกับเครือข่าย ซึ่งเรามีการแบ่งโพรโตคอลต่างๆ ออกเป็นระดับชั้น เพื่อให้ง่ายต่อการใช้งานและง่ายต่อการทำความเข้าใจ นานวันเข้าโพรโตคอลที่เราคิดว่ามันทำให้เราทำงานสะดวก นั้นก็มีเพิ่มมากขึ้นจนกลายเป็นความยุ่งยาก ต่างบริษัทก็ต่างมีผลิตภัณฑ์ที่สนับสนุนโพรโตคอลต่างๆ ไม่เหมือนกัน ทำให้การใช้งานร่วมกันทำได้ยากขึ้น ทำให้เราต้องมาหาวิธีใช้งานโพรโตคอลมาตรฐานเหล่านั้นสำหรับ Visual Basic แล้วการสื่อสารที่ใช้งานในอินเตอร์เน็ตเป็นเรื่องที่น่าสนใจ เพราะหมายถึงเราสามารถแลกเปลี่ยนข้อมูลระหว่างคอมพิวเตอร์ต่างๆ ที่เชื่อมต่อกัน ในอินเตอร์เน็ตได้อย่างไม่ยากเย็น โดยการใช้ ActiveX Control ที่ทรงประสิทธิภาพที่เตรียมไว้ให้ คือ Winsock Control ซึ่งบทนี้เราจะสร้างแอพพลิเคชั่นเหล่านั้นดูบ้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.11 รู้จักกับ Winsock

ระบบปฏิบัติการยูนิกซ์จะมีความสามารถอย่างหนึ่งที่เรียกว่า Socketซึ่งก็คือการที่โปรแกรมต่างๆสามารถสื่อสารข้อมูลระหว่างกันได้โดยไม่จำเป็นว่าโปรแกรมนรันจะต้องรันอยู่บนคอมพิวเตอร์เครื่องเดียวกัน

Socket เองก็มาจากแนวคาามคิดไคลเอ็นต์/เซิร์ฟเวอร์อันแสนจะคลาสสิก โดยส่วนที่ทำหน้าที่เป็นเซิร์ฟเวอร์จะสร้าง Socket (ซึ่งถ้าแปลตามตัวก็จะหมายถึงช่องเสียบ หรือรูเสียบ) จะมีจำนวนเท่าใดก็แล้วแต่ จากนั้นเมื่อมีไคลเอ็นต์ที่ต้องการสื่อสารด้วยก็จะติดต่อมาที่ Socket ที่เซิร์ฟเวอร์ได้เตรียมไว้ให้ เมื่อเชื่อมต่อกันเสร็จก็พร้อมจะแลกเปลี่ยนข้อมูลระหว่างกันได้ โดยเป็นการสื่อสารแบบ 2 ทางชนิดเต็มรูปแบบ (2 Way Full-Duplex)



รูป3.9หลักการทำงานของ Socket ในยูนิกซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

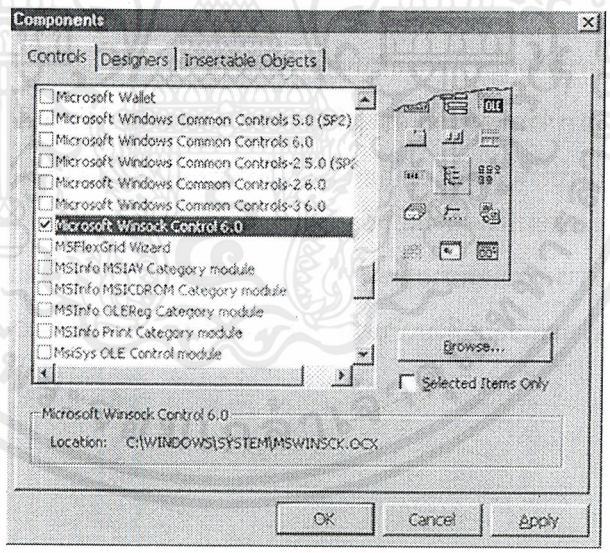
การใช้งาน Winsock กับระบบปฏิบัติการตระกูล Windows

ระบบปฏิบัติการตระกูล Windows อันได้แก่ Windows95 /98 และ WindowsNT ก็ได้เตรียมการทำงานในลักษณะSocket ไว้ให้นักพัฒนาแอพพลิเคชั่นได้ใช้กันโดยตั้งชื่อว่า Winsock โดยมีหลักการคล้าย คลังกับ Socket แม้จะไม่ Compatible กัน 100% ก็ตาม

ในมุมองของนักพัฒนาแอพพลิเคชั่นแล้ว Winsock เป็น API (Application Program Interface) หรือชุดฟังก์ชันมาตรฐาน ซึ่งออกแบบมาเพื่อทำให้การเขียนโปรแกรมกับอินเตอร์เน็ตเป็นเรื่องง่าย อีกทั้ง Winsock ยังถูกกำหนดให้เป็น API ที่มีลักษณะเปิดกว้าง และสามารถ port ไปยัง Socket ของยูนิกซ์ได้อย่างไม่ยากนัก

การใช้งาน Winsock Control

Visual Basic ได้เตรียม ActiveX Control ที่มีชื่อว่า Winsock Control ไว้ให้เราใช้งานซึ่งพร้อมให้เราใช้งานเพื่อสร้างการเชื่อมต่อระหว่างคอมพิวเตอร์ 2 เครื่องในเครือข่าย ทำให้เราสามารถแลกเปลี่ยนข้อมูลระหว่างกันได้ สำหรับ Winsock Control จะถูกเก็บไว้ในไฟล์ MSWINSCK.OCX ในโฟลเดอร์ \Windows\System



รูปที่ 3.10 เรียกใช้งาน Winsock Control

โหมดการทำงานของ Winsock Control

การทำงานของ Winsock Control กับโพรโตคอล TCP/IP นี้แบ่งการทำงานออกเป็น 2 โหมดคือ TCP กับ UDP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TCP (Transmission Control Protocol)

TCP เป็นการทำงานในลักษณะ Connection – Based เปรียบเหมือนกับการทำงานของโทรศัพท์ที่จะมีการตรวจสอบการทำงานจากทั้งสองฝั่งมีการโต้ตอบกันระหว่างกัน ทำให้ได้การสื่อสารที่มีประสิทธิภาพ

UDP(User Datagram Protocol)

UDP เป็นการทำงานในลักษณะตรงกันข้ามกับ TCP นั่นคือมีการทำงานในลักษณะ Connectionless เปรียบเหมือนกับการกระจายเสียงวิทยุ

ควรเลือกใช้งานในโหมดใด

TCP จะเหมาะกับการส่งข้อมูลที่ต้องการการทำงานที่ถูกต้องแม่นยำของข้อมูล เช่นการส่งภาพหรือเสียงผ่านเครือข่าย ซึ่งจะมีการตรวจสอบความถูกต้องของข้อมูลที่มีการส่งถึงกัน ทำให้มีการใช้ทรัพยากรของระบบค่อนข้างสูง

UDP จะเหมาะกับการส่งข้อมูลที่ไม่สำคัญนัก เช่น ส่งข้อมูลแสดงสถานะการทำงาน ซึ่งก็มักจะเป็นข้อมูลขนาดเล็ก ซึ่งทำให้ UDP กินทรัพยากรของระบบน้อยกว่า TCP

สำหรับตัวอย่างในบทนี้จะขอเน้นเฉพาะการทำงานร่วมกับ TCP เท่านั้น เพราะจะให้การทำงานที่มีประสิทธิภาพกว่า UDP

พรอพเพอร์ตี้สำคัญของ Winsock Control

Potocol เป็นการเลือกโพรโตคอลสำหรับการทำงาน

LocalPort เป็นการกำหนดหมายเลขพอร์ตของคอมพิวเตอร์ที่จะใช้งานกับ Winsock

RemoteHost เป็นการกำหนดหมายเลขพอร์ตของคอมพิวเตอร์ที่เราจะติดต่อด้วย อาจจะเป็น IP Address หรือเป็นชื่อคอมพิวเตอร์ที่เป็นชื่อที่ง่ายต่อการจดจำ

ByteReceive เป็นจำนวนข้อมูลที่รับเข้ามาเก็บในบัฟเฟอร์(จากเมธอด GetData) มีหน่วยเป็นไบต์

เมธอดสำคัญของ Winsock Control

Listen เป็นเมธอดที่ใช้สร้าง Socket ทำให้คอมพิวเตอร์เครื่องอื่นๆ สามารถติดต่อเข้ามาได้

Connect เป็นเมธอดที่ใช้สร้างการติดต่อแบบ Socket ไปยังคอมพิวเตอร์เครื่องอื่น โดยจะต้องระบุ

Socket Address (IP Address กับ หมายเลขพอร์ตที่กำหนดให้ใช้กับ Socket)

Accept เป็นเมธอดที่ใช้รับ Request จากคอมพิวเตอร์ที่ติดต่อเข้ามา

SendData เป็นเมธอดที่ใช้ส่งข้อมูลจากบัฟเฟอร์เข้ามาเก็บในตัวแปรที่เรากำหนดให้ โดยสามารถกำหนดชนิดตัวแปร และความยาวของข้อมูลที่จะนำมาเก็บได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Close เป็นเมธอดที่ใช้ยกเลิกการติดต่อแบบ Socket

อีเวนต์สำคัญของ Winsock Control

ConnectionRequest เป็นเหตุการณ์ที่เกิดเมื่อ คอมพิวเตอร์เครื่องอื่น มีการ Request เข้ามา ซึ่งจะมีการกำหนด ID ให้กับแต่ละ Request ที่เข้ามา

DataArrival เป็นเหตุการณ์ที่เกิดเมื่อมีข้อมูลชุดใหม่เข้ามาเก็บในบัฟเฟอร์ ซึ่งเราสามารถตรวจสอบขนาดข้อมูลได้จากพรอพเพอร์ตี้ ByteReceive

SendProgress เป็นเหตุการณ์ที่เกิดขณะที่กำลังมีการส่งข้อมูลระหว่างกัน ซึ่งจะมีพารามิเตอร์แสดงจำนวนข้อมูลที่ส่งมาแล้ว และข้อมูลที่ยังคงเหลือ

SendComplete เป็นเหตุการณ์ที่เกิดเมื่อการส่งข้อมูลเสร็จสิ้นสมบูรณ์

Error เป็นเหตุการณ์ที่เกิดเมื่อมีความผิดพลาดเกิดขึ้น ซึ่งจะแสดงหมายเลขของความผิดพลาด,คำอธิบาย , และรายละเอียดอื่นๆ สำหรับการจัดการกับข้อผิดพลาดที่เกิดขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
พื้นฐานไมโครคอนโทรลเลอร์ และการสื่อสารแบบอนุกรม

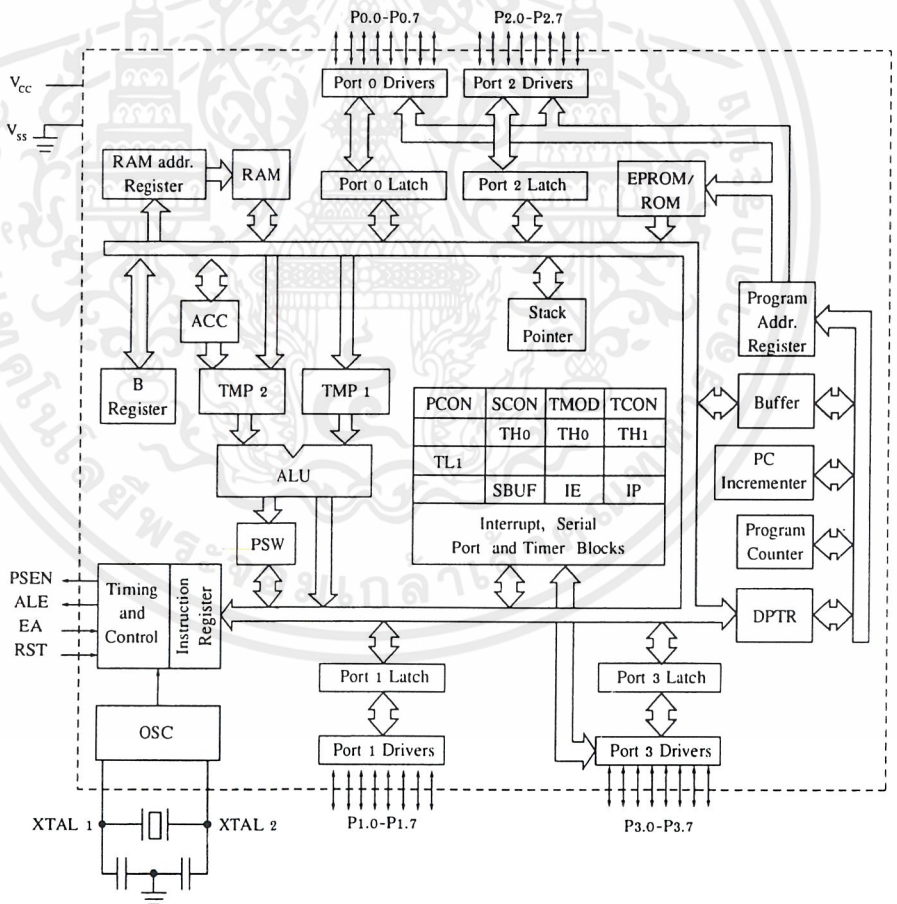
4.1 ไมโครคอนโทรลเลอร์

เป็นส่วนควบคุมการทำงานของชุดควบคุมมอเตอร์ ซึ่งมีหน้าที่

1. รับคำสั่งควบคุมจาก Computer มาประมวลผลเพื่อกำหนดตำแหน่งให้กับมอเตอร์
2. จัดเก็บข้อมูลตำแหน่งของมอเตอร์เพื่อส่งให้ Computer

โครงสร้างของ MCS-51

ไมโครคอนโทรลเลอร์ที่เลือกใช้ในครั้งนี้จะเป็นไมโครคอนโทรลเลอร์ในตระกูล MCS-51 ซึ่งมีอยู่ด้วยกันหลายเบอร์ขึ้นกับโครงสร้างภายในของมัน



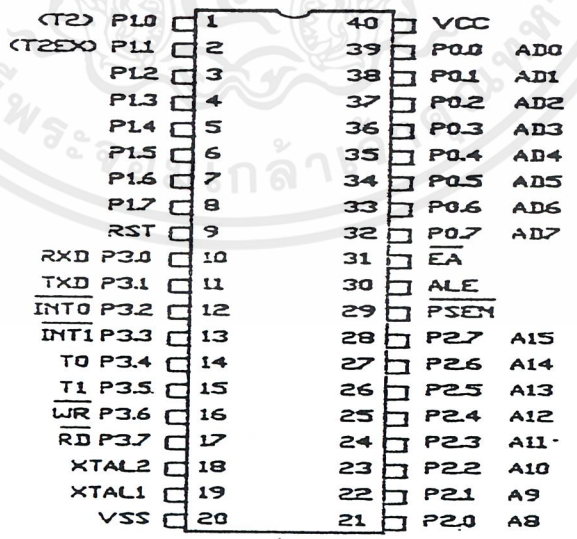
รูปที่ 4.1 แสดงโครงสร้างภายในของ MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คุณสมบัติที่สำคัญของ MCS-51 มีดังนี้

- มีหน่วยความจำ ROM 4K byte
- มีหน่วยความจำ RAM 128 byte
- มี Port I/O ขนาด 8 บิต 4 Port
- มี Timer 16 บิต 2 ตัว
- สามารถ Interrupt ได้ 5 แหล่ง
- มีวงจรออสซิลเลเตอร์และวงจรนาฬิกาบนชิพ
- มี Port อนุกรมที่สามารถรับส่งข้อมูลแบบ Full Duplex ความเร็วสูง
- อ้างหน่วยความจำโปรแกรมภายนอกได้ 64K
- อ้างหน่วยความจำข้อมูลภายนอกได้ 64K
- สามารถประมวลผลทีละบิตได้
- สามารถอ้างหน่วยความจำแบบบิตได้ 210 ตำแหน่งฯ
- หนึ่งวัฏจักรคำสั่งกินเวลาประมาณ 1 ไมโครวินาที ขณะทำงานด้วย Clock 12 MHz

เบอร์พื้นฐานของไมโครคอนโทรลเลอร์ MCS-51 ประกอบด้วย 8051,8031,8751 ซึ่งแตกต่างกันที่ชนิดและหน่วยความจำภายใน



รูปที่ 4.2 แสดงตำแหน่งขาของไมโครคอนโทรลเลอร์ตระกูล MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าที่การใช้งานของแต่ละขาของ MCS-51

- ขา Vss (ขา 20) สำหรับต่อลงกราวด์
- ขา Vcc (ขา 40) สำหรับต่อแหล่งจ่ายไฟตรง +5 V
- ขาพอร์ต 0 (ขา 32-39) มี 8 ขา ใช้เป็นพอร์ต 0 (P0.0-P0.7)เป็นได้ทั้งเอาต์พุตและอินพุตพอร์ตและใช้การติดต่อหน่วยความจำเป็นไบต์ต่ำ(A0-A7)
- ขาพอร์ต 1 (ขา 1-8) มี 8 ขา ใช้เป็นพอร์ต 1 (P1.0-P1.7)เป็นได้ทั้งเอาต์พุตและอินพุตพอร์ตทั่วไปโดยมีวงจรพูลอัพภายใน
- ขาพอร์ต 2 (ขา 21-28) มี 8 ขา ใช้เป็นพอร์ต 2 (P2.0-P2.7)เป็นได้ทั้งเอาต์พุตและอินพุตพอร์ตและใช้การติดต่อหน่วยความจำเป็นไบต์สูง(A8-A15)
- ขาพอร์ 3 (ขา 10-17) มี 8 ขา ใช้เป็นพอร์ต 3 (P3.0-P3.7)เป็นได้ทั้งเอาต์พุตและอินพุตพอร์ตทั่วไปโดยมีวงจรพูลอัพภายในและยังใช้งานในหน้าที่พิเศษต่างๆดังนี้
- ขา P3.0 ใช้รับข้อมูลภายนอกแบบอนุกรม
- ขา P3.1 ใช้ส่งข้อมูลออกภายนอกแบบอนุกรม
- ขา P3.2 ใช้เป็นอินพุตอินเตอร์รัปต์ชนิดที่ 0
- ขา P3.3 ใช้เป็นอินพุตอินเตอร์รัปต์ชนิดที่ 1
- ขา P3.4 สัญญาณอินพุตให้เคาน์เตอร์ของไทเมอร์ 0
- ขา P3.5 สัญญาณอินพุตให้เคาน์เตอร์ของไทเมอร์ 1
- ขา P3.6 ใช้เป็นสัญญาณควบคุมการเขียนข้อมูลไปยังหน่วยความจำข้อมูลภายนอก
- ขา P3.7 ใช้เป็นสัญญาณควบคุมการอ่านข้อมูลไปยังหน่วยความจำข้อมูลภายนอก

การใช้งานพอร์ต 3 ในหน้าที่พิเศษนี้จะต้องโหลดค่า 1 ไปยังแต่ละบิตที่ต้องการใช้งานก่อนทุกครั้ง

- ขา RST (ขา 9) ใช้ในการรีเซ็ตค่าภายในทั้งหมด
- ขา ALE/PROG (ขา 30) เป็นขาที่ใช้ในการส่งสัญญาณควบคุมการแลตช์ค่าแอดเดรสไบต์ต่ำ จาก P0 ในการติดต่อกับหน่วยความจำภายนอก
- ขา PSEN (ขา 29) ใช้ส่งสัญญาณสโตรบเพื่ออ่านคำสั่งจากโปรแกรม
- ขา EA/Vpp (ขา 31) เป็นขาสำหรับใช้เลือกให้ MCS-51 ทำงานจากโปรมแกรมภายในหรือภายนอก
- ขา XTAL (ขา 19) ใช้ต่อคริสตอลภายนอก โดยเป็นขาอินพุต
- ขา XTAL (ขา 18) ใช้ต่อคริสตอลภายนอก โดยเป็นขาเอาต์พุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

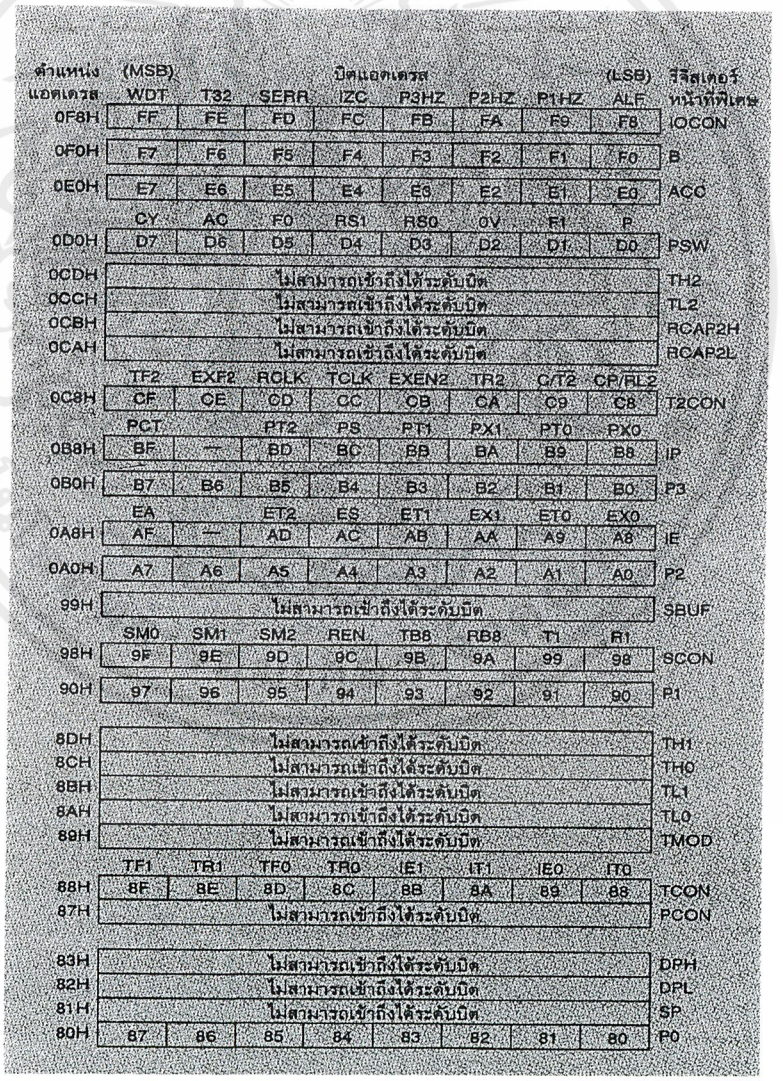
4.2 โครงสร้างของหน่วยความจำใน MCS-51

แบ่งออกเป็น 2 ส่วนคือ

1. หน่วยความจำโปรแกรม
2. หน่วยความจำข้อมูล

MCS-51 ทุกเบอร์จะมีหน่วยความจำสำหรับเก็บข้อมูลภายในอย่างน้อย 128 ไบต์ไปจนถึง 256ไบต์ หน่วยความจำสำหรับเก็บข้อมูลภายในบริเวณ 128 ไบต์แรกมีชื่อเรียกว่า Lower 128 และในบริเวณ 128 ไบต์หลังที่มีเพิ่มในบางเบอร์มีชื่อเรียกว่า Upper 128 ดังแสดงในรูปที่ 4.3

หน่วยความจำสำหรับเก็บข้อมูลภายในบริเวณ 128 ไบต์แรกและหลังจะมีวิธีการเข้าถึงข้อมูลในหน่วยความจำทั้งสองส่วนไม่เหมือนกัน

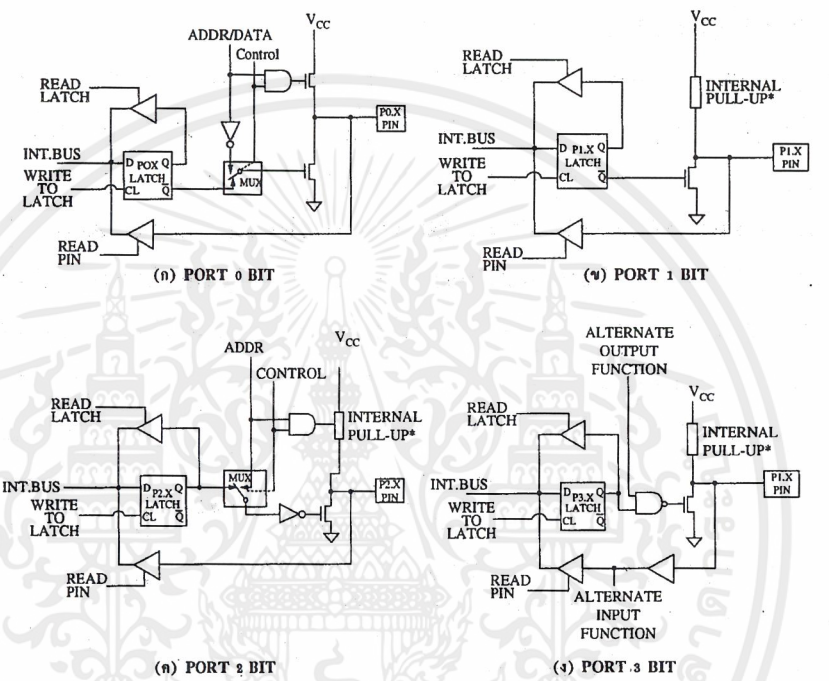


รูปที่ 4.3 ตำแหน่งหน่วยความจำของ MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 โครงสร้างของพอร์ทอินพุตเอาต์พุต

ขาของพอร์ทจะมีโครงสร้างเป็น Field-Effect Transistor ต่ออยู่กับขาภายนอก และมีตัวต้านทานต่อ Pull-up อยู่สำหรับพอร์ท 1,2,3 แต่ถ้าเป็นพอร์ท 0 จะไม่มีตัวต้านทาน Pull-up ภายใน เพราะว่าต้องใช้เป็นขา Address Bus และ Data Bus



รูปที่ 4.4 แสดงโครงสร้างพอร์ททั้ง 4 ของ MCS-51

พอร์ทนี้สามารถใช้เป็นอินพุทเอาต์พุทกับอุปกรณ์ภายนอกได้ ในการอ่านข้อมูลจากพอร์ทจะอ่านได้ 2 แบบคือ Read Latch และ Read Pin โดย Read Latch หมายถึงการอ่านข้อมูลที่ถูก Latch เอาไว้เข้าสู่ระบบบัสภายในของ MCS-51 แต่ถ้าเป็นการ Read Pin จะเป็นการใช้พอร์ทเป็นอินพุท โดยจะอ่านค่าจากขาของไอซีเข้าสู่บัสภายใน

4.4 รีจิสเตอร์ฟังก์ชันพิเศษ

ใน MCS-51 รีจิสเตอร์จะใช้หน่วยความจำ RAM ภายในชิพ โดยส่วนหนึ่งเป็น รีจิสเตอร์พิเศษ ซึ่งมีทั้งหมด 21 ตัว โดยรีจิสเตอร์พิเศษต่างๆ จะเริ่มที่หน่วยความจำตั้งแต่ 80H ถึง FFH ซึ่งมีทั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หมด 128 ตำแหน่ง แต่จะเป็นรีจิสเตอร์ฟังก์ชันพิเศษเพียง 21 ตำแหน่ง แต่ถ้าเป็น 8032 / 8051 จะใช้ 26 ตำแหน่งหรือมี SFR 26 ตัว

4.4.1 Program Status Word

รีจิสเตอร์ตัวนี้เรียกย่อๆ ว่า PWS จะอยู่ที่ตำแหน่ง D0H ซึ่งสามารถเข้าถึงข้อมูลระดับบิตได้ โดยรีจิสเตอร์ตัวนี้จะเป็นตัวบอกสภาวะต่างๆ ของไมโครคอลโทรเลอร์แยกเป็นบิตคือ

1. แฟลกตัวทด Carry Flag เป็นบิตที่ 7 ของ PSW บิตนี้จะมีความสำคัญหากมีการกระทำทางคณิตศาสตร์ โดยบิตนี้จะ set เมื่อเกิดการทดของบิตที่ 7 ขณะทำการบวกเลข หรือ set เมื่อเกิดการยืมของบิตที่ 7 เมื่อเกิดการลบเลข
2. แฟลกตัวทดช่วย Auxiliary Carry Flag เป็นบิตที่ 6 เมื่อมีการบวกแบบ Binary Code Decimal (BCD) บิต AC หรือบิตตัวทดช่วยจะถูก set เมื่อการทดจากบิตที่ 3 ไปบิตที่ 4 หรือถ้าใน Lowe Nibble มีค่าระหว่าง 0AH-0FH เนื่องจากรหัส BCD มีค่าได้มากที่สุดแค่ 9 ถ้าหากมีการบวกเลขแบบ BCD จะต้องตามด้วยคำสั่ง DAA เพื่อปรับค่าที่เกิน 9 โดยบวกเพิ่มด้วย 6 เข้าไป จะทำให้เป็นรหัส BCD ได้
3. แฟลกศูนย์ Flag 0 เป็นบิตที่ 5 ของ PSW
4. บิตเลือกรีจิสเตอร์แบงก์ บิตที่ 4,3 ทำให้เลือกใช้ รีจิสเตอร์ R0-R7 ได้ 4 ชุด
5. แฟลกโอเวอร์โฟลว์ บิตที่ 2 จะถูก set เมื่อกระทำทางคณิตศาสตร์ แล้วเกิด Overflow คือจำนวนที่ได้จากการกระทำมีค่าเกินกว่าจำนวนไบต์ที่จะเป็นได้คือ มากกว่า 128 หรือน้อยกว่า –128
6. บิตพาริตี้ Parity Bit เป็นบิตที่บอกค่าพาริตี้ของรีจิสเตอร์ A โดยเป็นบิตที่ 0 ของ PSW

4.4.2 รีจิสเตอร์ B

จะอยู่ที่ตำแหน่ง F0H เป็นรีจิสเตอร์ที่สามารถใช้งานได้ทั่วไปได้ แต่โดยส่วนใหญ่จะใช้รีจิสเตอร์นี้คูณหรือหาร กับรีจิสเตอร์ A

4.4.3 ตัวชี้สแตก (Stack Pointer)

SP เป็นรีจิสเตอร์ขนาด 8 บิต อยู่ที่ตำแหน่ง 81H

4.4.4 รีจิสเตอร์ Data Pointer (DPTR)

DPTR เป็นรีจิสเตอร์ที่ใช้สำหรับชี้ตำแหน่งรหัสโปรแกรมหรือข้อมูลในหน่วยความจำ โดยเป็นรีจิสเตอร์ขนาด 16 บิต ซึ่งประกอบด้วยรีจิสเตอร์ 2 ตัว คือ DPL ตำแหน่งที่ 82H โดยจะเก็บเป็น 8 บิตต่ำและ DPH ตำแหน่งที่ 83H โดยจะเก็บค่า 8 บิตสูง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4.5 รีจิสเตอร์พอร์ท (Port Register)

ใน MCS-51 ค่าของพอร์ทจะหมายถึงค่าของหน่วยความจำด้วย หากต้องการส่งข้อมูลออกพอร์ท ก็เพียงแต่เขียนข้อมูลไปที่หน่วยความจำตำแหน่งที่พอร์ทนั้นอยู่ และถ้าหากต้องการจะอ่านพอร์ท ก็เพียงอ่านค่าจากตำแหน่วที่หน่วยความจำนั้นอยู่ ใน MCS-51 พอร์ท 0 จะอยู่ที่ตำแหน่ง 80H พอร์ท 1 จะอยู่ที่ตำแหน่ง 90H พอร์ท 2 อยู่ที่ตำแหน่ง A0H และพอร์ท 3 อยู่ที่ตำแหน่ง B0H พอร์ท 0,2,3

โดยทั่วไปแล้วจะไม่มีการใช้ถ้าหากมีการติดต่อกับหน่วยความจำภายนอกหรือใช้เป็นพอร์ทพิเศษ โดยปกติแล้วจะใช้ พอร์ท 1 ในการติดต่อกับอุปกรณ์ภายนอกและพอร์ททุกพอร์ทสามารถอ้างข้อมูลระดับบิตได้

4.4.6 รีจิสเตอร์เวลา (Timer Register)

ใน MCS-51 เบอร์ 8051 จะมีรีจิสเตอร์ที่ใช้นับและจับเวลาขนาด 16 บิต อยู่ 2 ตัวคือ Timer 0 อยู่ที่ตำแหน่ง 8AH และ 8CH โดยหมายถึง TL0 และ TH0 ตามลำดับ รีจิสเตอร์อีกตัวคือ Timer 1 ตำแหน่ง 8BH เป็น TL1 และ 8DH เป็น TH1 การใช้งาน Timer จะต้องกำหนดการทำงานในรีจิสเตอร์ TMOD ซึ่งอยู่ที่ตำแหน่ง 88H ก่อน

4.4.7 รีจิสเตอร์พอร์ทอนุกรม(Serial Port Register)

MCS –51 จะมีพอร์ทสื่อสารอนุกรมอยู่ภายในชิพ ซึ่งสามารถจะรับส่งข้อมูลได้โดยติดต่อผ่านรีจิสเตอร์ SBUF ซึ่งอยู่ที่ตำแหน่ง 99H และต้องกำหนดการใช้งานผ่านรีจิสเตอร์ SCON

4.4.8 รีจิสเตอร์อินเตอร์รัพท์ (Interrupt Port Register)

MCS-51 สามารถ Interrupt ได้ 5 ตำแหน่งโดยมี 2-Priority ตัว Interrupt นี้จะถูก Disable หลังจากระบบถูก Reset และจะ Enable หลังจากที่เขียนข้อมูลไปที่รีจิสเตอร์ IE หรือตำแหน่ง A8H

4.4.9 Power Control Register (PCON)

รีจิสเตอร์ PCON อยู่ที่ตำแหน่ง 87H ใช้หยุดการทำงานของ MCS-51 โดยจะหยุดจ่ายสัญญาณนาฬิกาให้ระบบ ทำให้ข้อมูลต่างๆ ภายใน MCS-51 ไม่ม่การเปลี่ยนแปลงและทำให้ลดพลังงานไฟฟ้าที่จ่ายให้กับ MCS-51 ลงด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 การสื่อสารแบบอนุกรม

ลักษณะของการส่งข้อมูลแบบอนุกรมนั้น ข้อมูลจะส่งออกมาทีละบิต จากตัวส่งไปตัวรับข้อมูล ช่องสัญญาณในการส่งข้อมูลอาจใช้เพียง 1 หรือ 2 ช่องสัญญาณเท่านั้น ทำให้ค่าใช้จ่ายในการสื่อสารจะถูกกว่าแบบขนาน แต่อัตราการรับ-ส่งข้อมูลจะช้ากว่าแบบขนาน ในการส่งข้อมูลแบบอนุกรม ข้อมูลที่ต้องการส่งจะอยู่ในลักษณะเป็นไบต์จะทยอยส่งทีละบิต และทางตัวรับจะต้องรับข้อมูลเข้ามาทีละบิต แล้วมารวมกันเป็นไบต์ซึ่งทางตัวรับจะต้องคอยตรวจสอบว่าบิตใดเป็นบิตเริ่มต้นหรือบิตสุดท้ายของข้อมูล การตรวจสอบขึ้นอยู่กับรูปแบบของรหัสของบิตที่ใช้ ซึ่งในการรับส่งข้อมูลแบบอนุกรมระหว่างไมโครคอมพิวเตอร์กับอุปกรณ์ภายนอกนั้น จำเป็นจะต้องมีมาตรฐานในการรับส่งข้อมูล ซึ่งมาตรฐานที่นิยมมากที่สุดก็คือ มาตรฐาน RS-232

เพื่อที่จะทำให้อุปกรณ์จากผู้ผลิตต่างกันทำงานร่วมกันได้ มาตรฐานหลายชนิดจึงได้รับการออกแบบขึ้น มาตรฐานที่ใช้กันอย่างกว้างขวางที่สุดคือ RS-232C ซึ่งโดยปรกติไมโครคอมพิวเตอร์จะมีพอร์ตที่เป็นแบบอนุกรมอยู่ในตัวอยู่แล้ว และจะทำหน้าที่รับส่งข้อมูลในแบบอนุกรม

ตามจุดประสงค์ของมาตรฐาน RS-232C นั้นเพื่อจะสามารถเชื่อมต่อกันระหว่างอุปกรณ์รับส่งปลายทาง (Data Terminal Equipment : DTE) เช่น พอร์ตของคอมพิวเตอร์หลักหรืออุปกรณ์ปลายทางกับอุปกรณ์สื่อสาร

4.5.1 MCS-51 กับการรับส่งข้อมูลแบบอนุกรม

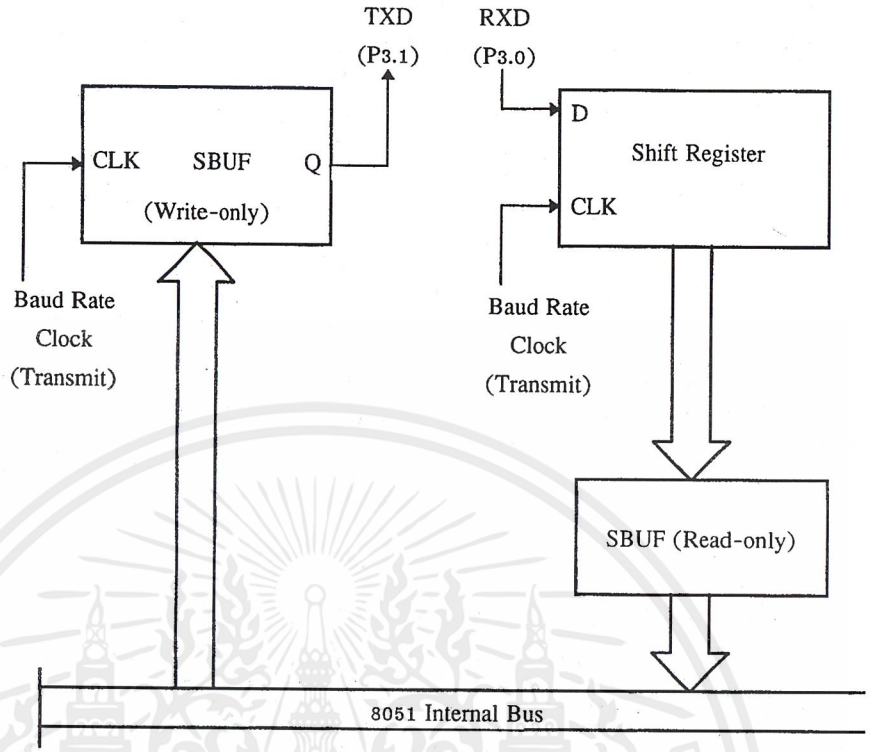
พอร์ทอนุกรมของ MCS-51 จะใช้ขา TXD และ RXD ในการรับส่งข้อมูล โดยขาทั้งสองจะอยู่ในพอร์ท 3 คือ P3.1 เป็น TXD และ P3.0 เป็น RXD พอร์ทอนุกรมของ MCS-51 สามารถทำงานแบบ Full Duplex ได้ คือสามารถส่งและรับข้อมูลในเวลาเดียวกันได้ โดยในการรับและส่งข้อมูลจะมีบัฟเฟอร์สำหรับเก็บข้อมูลให้ใช้

รีจิสเตอร์ที่สำคัญในการรับส่งข้อมูลคือ SBUF และ SCON ซึ่งเป็นรีจิสเตอร์ที่อยู่ใน Special Function Register โดยรีจิสเตอร์ SBUF จะอยู่ในตำแหน่ง 99H ถ้าเขียนข้อมูลไปที่ตำแหน่งนี้ จะเป็นการส่งข้อมูลออกทางพอร์ทอนุกรม และถ้าอ่านข้อมูลจากตำแหน่งนี้จะเป็นการรับข้อมูลจากพอร์ทอนุกรม โดยใน SBUF จะประกอบด้วยบัฟเฟอร์ 2 ตัว สำหรับส่งและรับข้อมูล

สำหรับ SCON ซึ่งอยู่ที่ตำแหน่ง 98H จะเป็นรีจิสเตอร์ที่สามารถเข้าถึงข้อมูลระดับบิตได้ รีจิสเตอร์นี้จะทำหน้าที่ควบคุมและบอกสถานะต่างๆ ของการรับส่งข้อมูลแบบอนุกรม

ส่วนความเร็วของการส่งข้อมูลสามารถหาได้จากการหารสัญญาณนาฬิกาที่ใช้กับ MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 แสดงการรับส่งข้อมูลระหว่างรีจิสเตอร์กับบัสภายใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

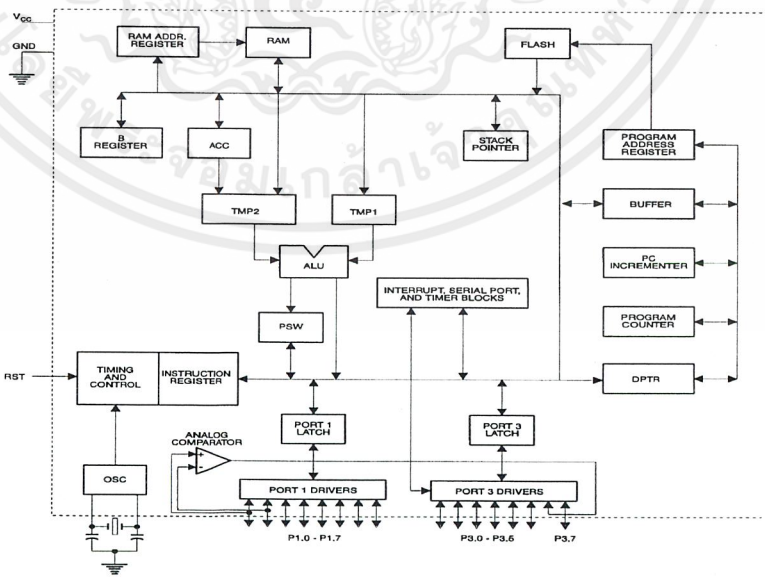
บทที่ 5
หลักการออกแบบและการสร้าง

การออกแบบและสร้างจะประกอบไปด้วย ชุดควบคุมกล้อง ส่วนของประมวลผล(Server Process) และส่วนควบคุม (Client)จะได้อธิบายดังต่อไปนี้

5.1 **ชุดควบคุมกล้อง (Camera Control)** โดยจะแบ่งเป็น 2 ส่วนคือ

5.1.1 ส่วนฐานจับกล้อง ได้ออกแบบตามลักษณะกล้องที่ใช้งานคือคำนึงถึงความสะดวกกระทัดรัด ซึ่งเป็นกล้อง Videoconference ที่นิยมใช้ในปัจจุบัน มีการอินเตอร์เฟสกับคอมพิวเตอร์โดยผ่านทางช่องรับสัญญาณ Video ของการ์ดจับภาพ คุณภาพการแสดงภาพ จะเป็นภาพสีคุณภาพจัดว่าใช้ได้เมื่อมาใช้กับโครงงานนี้และในส่วนของ Stepping Motor ที่ใช้จะมีขนาดเล็กและหาซื้อได้ง่ายราคาถูก ซึ่งมีคุณสมบัติคือ ใช้ไฟเลี้ยง 12 Volt และมี 1.8 ดีกรีต่อ 1 สเต็ป จำนวนสองตัว เพื่อทำงานในการหมุนทางด้านแนวซ้ายขวา กับ แนวบนล่าง

5.1.2 ส่วนไมโครคอนโทรลเลอร์และวงจรขับมอเตอร์ โดยจะมีอยู่สองส่วนด้วยกันคือส่วนที่เป็นไมโครคอนโทรลเลอร์ซึ่งโครงงานนี้ได้เลือกใช้ Mcs-51 Microcontroller เบอร์ AT89C2051ซึ่งเป็น Microcontroller ขนาดเล็กใช้งานง่าย โครงสร้างภายในของ AT89C2051 จะคล้ายกับโครงสร้างหลักของ MCS-51 แต่จะลดจำนวนพอร์ตลงทำให้ได้เป็น Microcontroller ขนาดเล็กขนาด 20 ขาดังรูปที่ 5.1

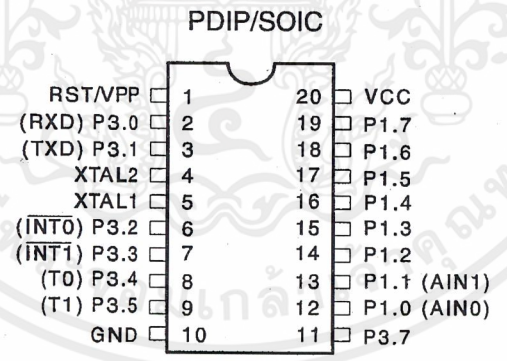


รูปที่ 5.1 แสดงโครงสร้างภายในของ AT89C2051

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดของ AT89C2051

- มีโครงสร้างและชุดคำสั่งเหมือนกันกับไมโครคอนโทรลเลอร์ในตระกูล MCS-51
- มีหน่วยความจำโปรแกรมชนิด Flash Memory ขนาด 2 Kbyte สามารถโปรแกรมซ้ำได้ 1000 ครั้ง
- มีหน่วยความจำแบบ RAM 8-bit ขนาด 128 Byte (Internal Ram)
- ทำงานที่แรงดัน 2.7 – 6 โวลต์
- Run ที่ความเร็วนาฬิกา 0Hz-24MHz
- มีพอร์ต I/O 15 บิต (พอร์ต 1 และ 3)
- พอร์ตสามารถ Sink กระแสได้ 20 mA และใช้เป็นเอาต์พุตขับ LED ได้โดยตรง
- มีพอร์ตสื่อสารแบบอนุกรม 1 Channel (UART)
- สามารถโปรแกรมข้อมูลเพื่อป้องกันการอ่าน-เขียนได้ 2 ระดับ
- มีวงจรไทเมอร์และวงจรนับขนาด 16 บิต จำนวน 2 Channel
- มีสัญญาณอินเตอร์รัปต์ 5 แหล่ง แบ่งระดับความสำคัญได้ 2 ระดับ
- มีวงจรเปรียบเทียบสัญญาณอะนาลอก 1 Channel
- มีระบบประหยัดพลังงาน



รูปที่ 5.2 แสดงตำแหน่งขาต่างๆของ AT89C2051

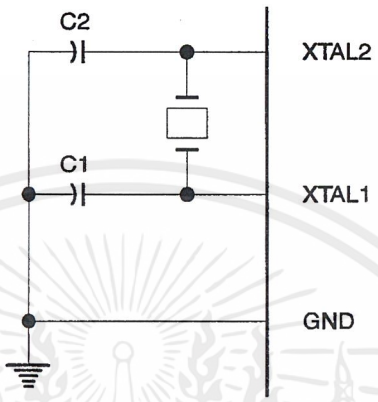
การออกแบบวงจรส่วนของ AT89C2051

AT89C2051 มี I/O พอร์ตทั้งหมด 15 บิต คือ พอร์ต 1 (P1.0-P1.7)และ พอร์ต 3 (P3.0 – P3.5 และ P3.7)แต่ในโครงงานนี้จะใช้เพียง 8 บิต0จึงใช้ P1.0-P1.7 เป็น Output Port สำหรับการควบคุมมอเตอร์สองตัว จากรูปที่ 5.1 จะเห็นว่า P1.0 และ P1.1 จะใช้เป็นขาร่วมของ Input ของวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Comparator ด้วยจึงไม่มี Internal Pull-Up ทำให้เมื่อเรานำ P1.0 , P1.1 มาใช้เป็น Output Port จะต้องทำการเพิ่ม Pull-Up จากภายนอกให้ด้วย

ส่วนของการต่อ XTAL ให้กับ AT89C2051 เราจะใช้ XTAL ค่า 11.059 Mhzเป็นตัวกำหนดสัญญาณนาฬิกาให้กับคอนโทรลเลอร์



รูปที่ 5.3 แสดงการต่อคริสตอลให้กับ AT89C2051

C1, C2 ที่เห็นในรูปที่ 5.3 จะใช้ค่าในช่วง 20pF – 40pF สำหรับการใช้คริสตอล และ 30pF – 50pF สำหรับการใช้ Ceramic Resonators

ส่วนพอร์ต 3 นอกจากจะเป็น I/O พอร์ตแล้วยังทำงานในหน้าที่อื่นด้วยตามตารางที่ 5.1

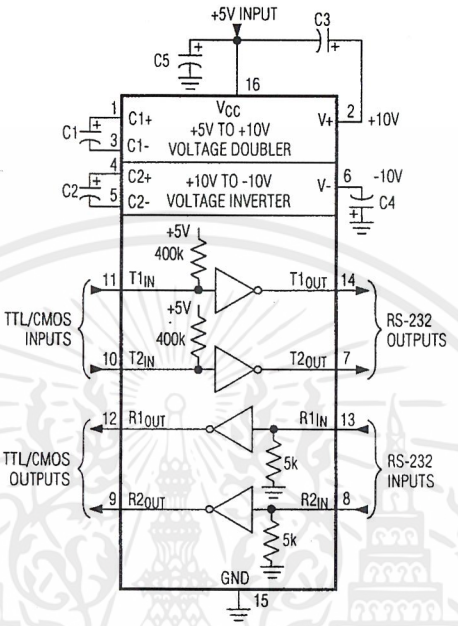
Port Pin	Alternate Functions
P3.0	RXD (serial input port)
P3.1	TXD (serial output port)
P3.2	$\overline{\text{INT0}}$ (external interrupt 0)
P3.3	$\overline{\text{INT1}}$ (external interrupt 1)
P3.4	T0 (timer 0 external input)
P3.5	T1 (timer 1 external input)

ตารางที่ 5.1 แสดงหน้าที่พิเศษของพอร์ต 3

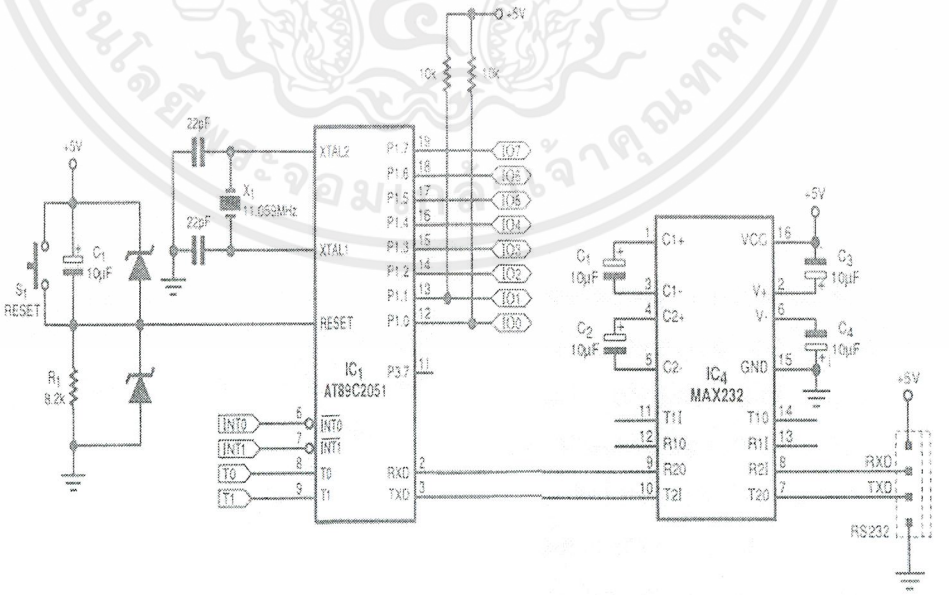
ซึ่งในโครงงานนี้จะใช้หน้าที่พิเศษของพอร์ต 3 นี้ด้วยคือ P3.0 และ P3.1 โดยจะนำมาเป็นส่วนของการส่งข้อมูลระหว่าง ไมโครคอนโทรลเลอร์กับคอมพิวเตอร์ในลักษณะการสื่อสารแบบอนุกรมตามแบบ RS-232 ซึ่งจะต้องเพิ่มส่วนของ Line Driver เพื่อให้แรงดันในการสื่อสารเป็นไปตาม RS-232 โดยจะเลือกใช้ IC เบอร์ MAX232 ในการทำหน้าที่นี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 5.4 จะเห็นได้ว่าสามารถใช้งานได้โดยง่ายเพียงแค่ต่อ C เพิ่ม 4-5 ตัวก็จะได้ Line Driver 2 ชุด โดยเราจะใช้ค่า C1,C2 เท่ากับ 4.7 uF , C3,C4 เท่ากับ 10uF และ C5 เท่ากับ 0.1 uF



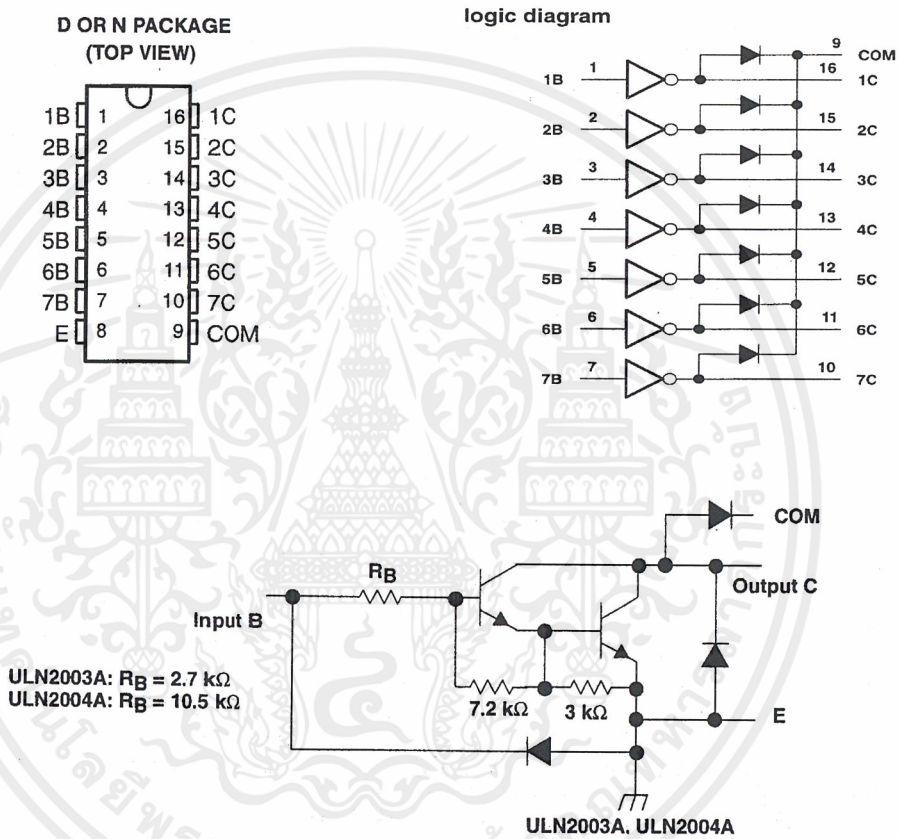
รูปที่ 5.4 แสดงโครงสร้างของ MAX232



รูปที่ 5.5 แสดงวงจรที่ทำการออกแบบในส่วนของคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนของวงจรขับมอเตอร์ เนื่องจากมอเตอร์ที่ใช้เป็นแบบสเต็บสามารถควบคุมได้โดยการป้อน Pulse .ให้แต่ละเฟสของมอเตอร์ แต่ Output ของพอร์ตไม่สามารถที่จะไปขับมอเตอร์ได้โดยตรง และเป็นการป้องกันความเสียหายของตัวไมโครคอนโทรลเลอร์จึงจำเป็นต้องมีชุดขับกระแสต่อระหว่างพอร์ตของไมโครคอนโทรลเลอร์กับมอเตอร์ ในโครงงานนี้จะใช้ IC ULN2003A เป็นตัวขับกระแสและแรงดันให้กับมอเตอร์



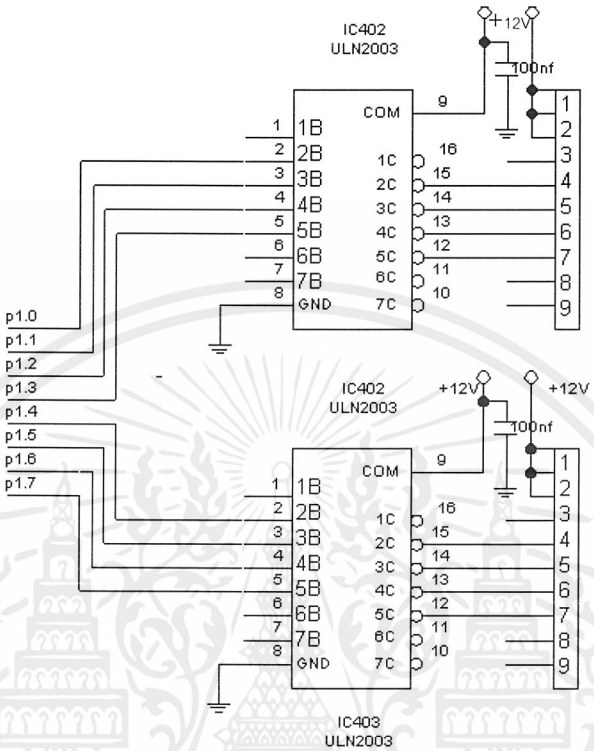
รูปที่ 5.6 แสดงรายละเอียดของ ULN 2003A

ULN 2003A แต่ละตัวสามารถใช้เป็นวงจรขับได้ 7 ตัว ทำให้เราต้องใช้ 2 ตัวในการใช้ขับวงจร 8 บิตโดยแบ่งเป็นตัวละ 4 บิต บน – ล่าง ทำให้แยกขับออกเป็นมอเตอร์แต่ละตัวได้โดยไม่จำเป็นต้องต่ออุปกรณ์เพิ่มจากภายนอกเลย

คุณสมบัติที่สำคัญของ ULN2003A

- มี Output Voltage สูงได้ถึง 50 V
- จ่ายกระแส Output สูงสุดได้ถึง 500 mA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

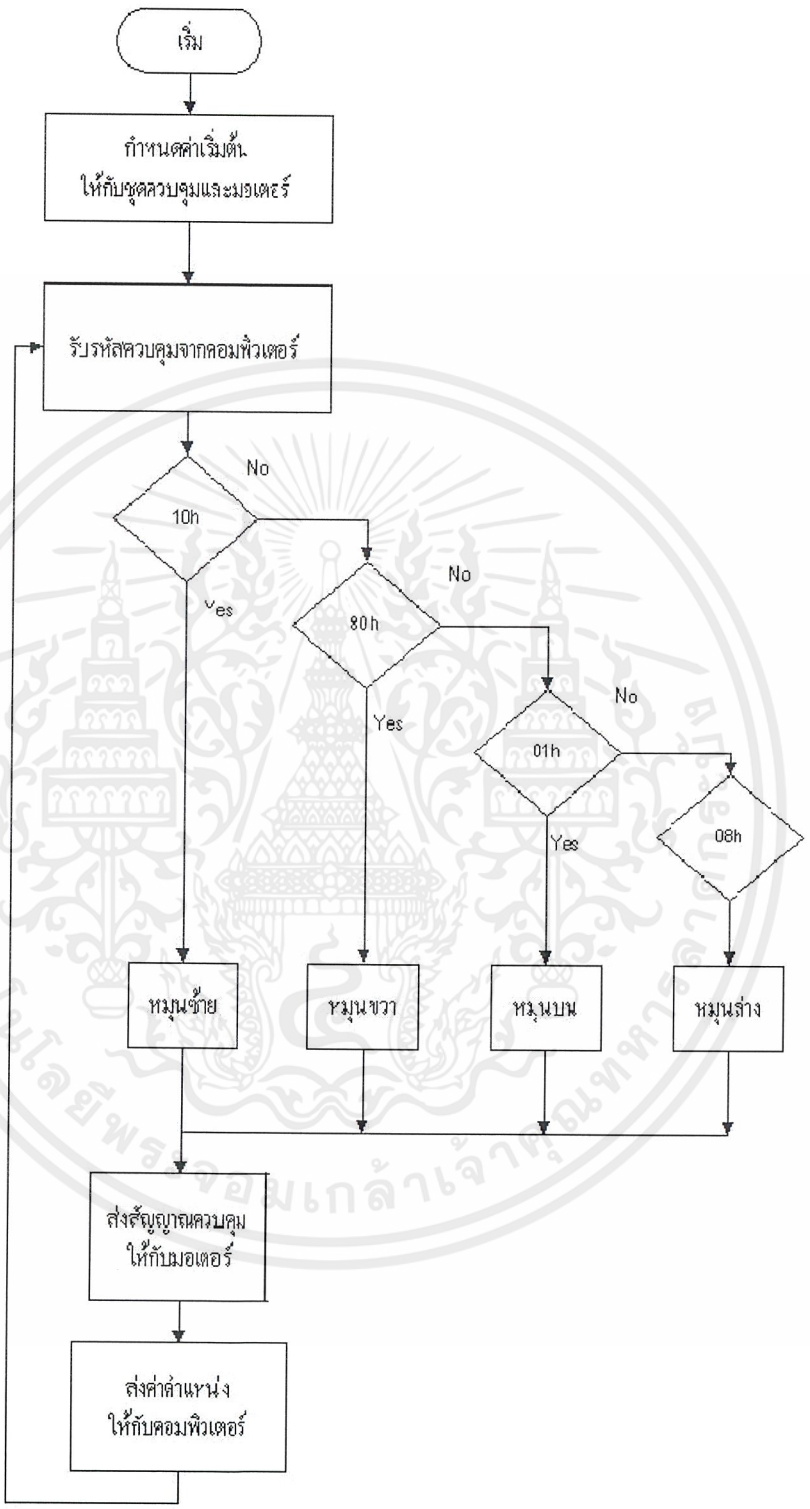


รูปที่ 5.7 แสดงวงจรขับกระแสของชุดควบคุมกล้อง

สำหรับโปรแกรมสำหรับควบคุมการทำงานของสเต็ปปิ้งมอเตอร์ทั้งสองตัวนี้ ใช้ภาษา Assembly ซึ่งจะเก็บไว้ภายใน Flash Memory ของ AT89C2051 ที่มีรายละเอียดการทำงานของฟังก์ชันตาม Flowchart ดังรูปที่ 5.8

รายละเอียดการทำงานของชุดควบคุมกล้องเริ่มจากการกำหนดและ Set ค่าต่างให้ไมโครคอนโทรลเลอร์พร้อมที่จะทำการสื่อสารกับเครื่องคอมพิวเตอร์ที่ทำตัวเองเป็น Server หลังจากนั้นจะทำการรอรหัสการควบคุมจาก Server แล้วนำมาประมวลผลค่าตำแหน่งของมอเตอร์ที่ไมโครคอนโทรลเลอร์ซึ่งทำหน้าที่รอรับค่าจากคอมพิวเตอร์จะรอรับเพียง Byte เดียว โดยผ่าน RS232 ซึ่งต่ออนุกรมกับเครื่องคอมพิวเตอร์ ซึ่งจะมีโค้ดในการสั่งงานให้ Motor นั้นทำงานคือ โค้ดในการสั่งให้ Motor หมุนซ้ายโดยรับค่า 10H, หมุนขวา80H, หมุนขึ้นบน08H, หมุนลงล่าง01H ซึ่งโค้ดเหล่านี้จะมาจาก Server อีกที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.8 แสดง FlowChart ของโปรแกรมที่อยู่ในชุดควบคุมกล้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 ชุดโปรแกรมการติดต่อใช้งานระหว่างผู้ควบคุมและผู้ใช้บริการ

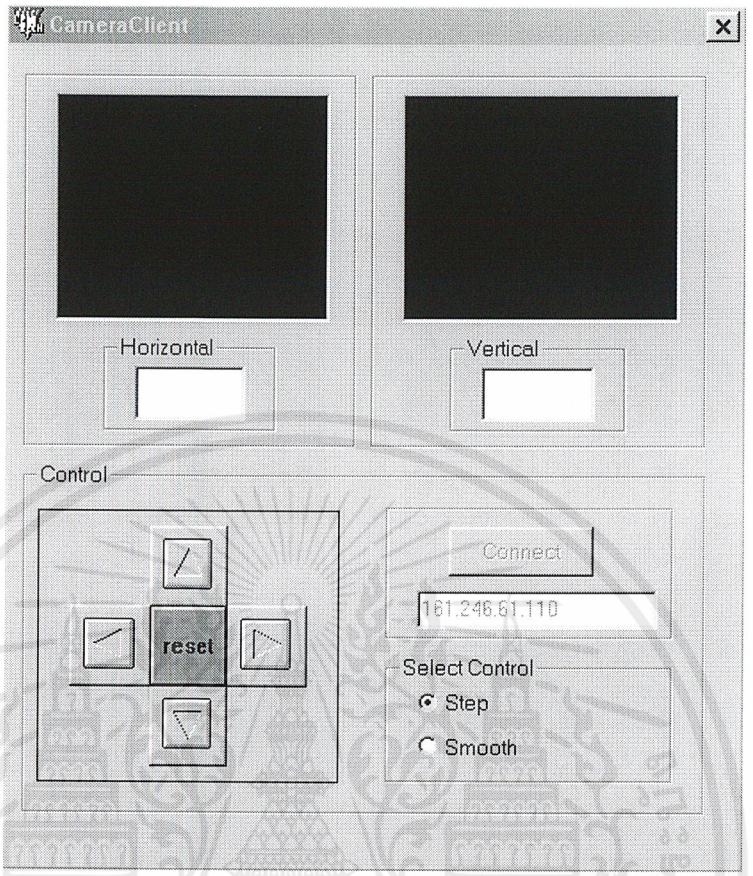
สำหรับในส่วนของคอมพิวเตอร์ที่เป็น Server Control ซึ่งจะใช้ PC ธรรมดาที่มีคุณสมบัติพอใช้ได้ ที่ใช้ระบบปฏิบัติการ Windows 9X เพื่อที่ความสามารถที่จะรันโปรแกรม Visual Basic 6 ได้อย่างไม่มีปัญหา การควบคุมการทำงานของการ หมุน Stepping Motor และอีกส่วนหนึ่งคือ การนำภาพมาแสดงผลที่ Client เริ่มจากการการหาโปรแกรมที่สามารถเขียนได้บน Window 95 & 98 ได้ ซึ่งได้พิจารณาความเหมาะสมแล้ว ใช้ Visual Basic 6 ซึ่งตรงกับความต้องการเบื้องต้นต้องการและได้ทำการเขียนโปรแกรมที่ใช้สำหรับเทอมนี้

ในการทำงานของโปรแกรมนี้หลักที่สำคัญคือรับข้อมูลจาก Mcs-51 มาทำการประมวลผลและแสดงผล และรับคำสั่งจากผู้ใช้งาน (User) ไปควบคุมการหมุนของ มอเตอร์อีกที และที่สำคัญคือการแสดงผลจากภาพจากกล้องวิดีโอ ซึ่งรายละเอียดปลีกย่อยจะอยู่ในภาคผนวกแล้ว

จากที่ได้อธิบายมาจะเป็นส่วนของการติดต่อระหว่างเครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็น Server กับตัวชุดจับกล้องโดยจะผ่านทางพอร์ตอนุกรม RS 232 ต่อไปก็จะเป็นการออกแบบในส่วนของการติดต่อระหว่างผู้ใช้งานคือ โปรแกรมด้านผู้ควบคุม(ผู้ใช้งาน) และโปรแกรมทางด้านผู้ให้บริการ(Server) โดยจะเริ่มจากออกแบบโปรแกรมในส่วนที่เป็นผู้ใช้ก่อนโดยจะมีรายละเอียดดังต่อไปนี้

5.2..1 โปรแกรมควบคุมด้านผู้ใช้ (Client side Program) เนื่องจากโปรแกรมที่ทางผู้จัดทำได้เขียนบนระบบปฏิบัติการ Window 95 ดังนั้นโปรแกรมที่เรานำมาสร้างโปรแกรมควบคุมจะใช้โปรแกรม Visual Basic 6 ซึ่งสามารถทำงานได้ดีบนระบบนี้ได้ เริ่มจากได้สร้างโปรแกรม โดยจะทำการออกแบบโครงสร้างโปรแกรมนี้ก่อน โดยติดจากฟังก์ชั่นการทำงานทั้งหมดที่จำเป็นในการทำงานของการควบคุมทิศทางการหมุนของกล้องที่อยู่ด้านปลายทางอีกที โดยมีฟังก์ชันการทำงานที่เห็นคือ ส่วนการควบคุมทิศทางการหมุน โดยจะควบคุมการหมุนไปทางด้ายซ้าย 90 องศา ด้านขวา 90 องศา ด้านบน 45 องศา ด้านล่าง 45 องศา จากตำแหน่งจุดศูนย์กลางทั้งสองด้าน และส่วนที่โปรแกรมที่ใช้สำหรับติดต่อกับชุดจับกล้อง โดยชุดจับกล้องตัวนี้จะอยู่คนละที่กับด้านของผู้ใช้นี้ซึ่งจะต้องติดต่อผ่านเครือข่ายอินเตอร์เน็ตโดยการโปรแกรมก็จะเป็นการส่งรหัสควบคุมการหมุนไปยังชุดควบคุมกล้องซึ่งอยู่ไกลออกไป และจะคอยรับข้อมูลที่จะบอกตำแหน่งปัจจุบันของมอเตอร์ที่เราเมื่อได้กำหนดการทำงานทั้งหมดแล้วก็จะเป็นการเริ่มสร้างส่วนของโปรแกรมชุดนี้ดังรูปที่ 5.3 รูปแสดงการวางตำแหน่ง component ต่าง ๆ โดยการจัดรูปแบบจะมีตำแหน่งของทิศทางของกล้องบอกว่าตอนนี้อยู่ที่องศาใดและยังมีภาพการเคลื่อนที่ของกล้องแสดงยังด้านClientSideอีกด้วย แล้วเขียนโปรแกรมต่าง ๆ ที่ได้กำหนดไว้ตามหน้าที่การใช้งานและ เพื่อใช้ในการรับภาพจากด้านServer โดย Component ที่สำคัญในส่วนจะทำงานบนอินเตอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.9 แสดงการวางตำแหน่ง Component ต่างๆของClient Control

5.2.2 โปรแกรมด้านให้บริการ(Server Side Program) โดยในส่วนนี้จะเป็นส่วนที่เป็นตัวจัดการเกี่ยวกับการควบคุม การรับส่งข้อมูล การจัดการเกี่ยวกับภาพที่จะส่งไปยังจุดหมายปลายทาง โดยจะมีชุดจับกล้องรวมอยู่ที่ส่วนนี้ด้วย แต่เราจะไม่พูดถึงในส่วนนี้เพราะได้กล่าวมาแล้วในตอนต้นโดยจะกล่าวถึงโปรแกรมที่ใช้ควบคุมในส่วนนี้ จะเป็นภาพจากกล้องวิดีโอโดยผ่านทางพอร์ตอนุกรม และสุดท้ายจะเป็นการรับคำสั่งควบคุมจากผู้ใช้แล้วรายงานผลตำแหน่งของมอเตอร์ รวมทั้งส่งภาพไปยังปลายทางด้วย การโปรแกรมจะเริ่มจากการกำหนดลักษณะของหน้าที่การทำงานทั้งหมด อาจแบ่งหน้าที่การทำงานภายในโปรแกรมนี้ออกได้เป็น 3 ส่วนใหญ่ได้แก่

1) ส่วนการติดต่อกับชุดจับกล้อง โดยที่ชุดจับกล้องที่ได้ออกแบบจะติดต่อโดยผ่านทางพอร์ตอนุกรม RS 232 แต่ในการเขียนโปรแกรมโดยใช้ Visual Basic ซึ่งโปรแกรมจะมี Component มาตรฐานอยู่แล้วซึ่งนำมาใช้ได้

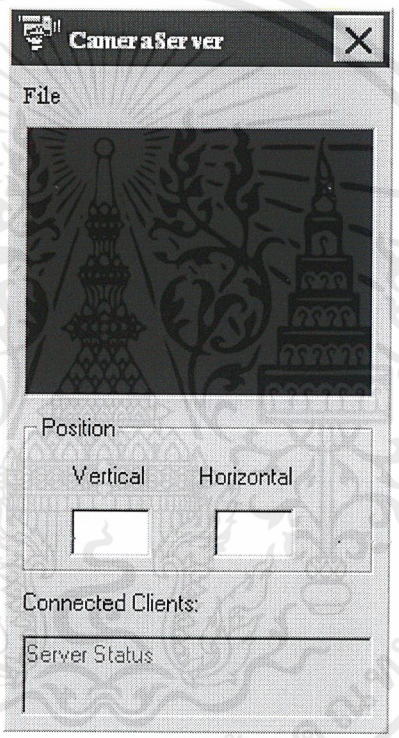
2) ส่วนในการรับภาพและบันทึกภาพจากกล้องวิดีโอ ในส่วนนี้ถือว่าเป็นส่วนที่สำคัญอย่างหนึ่งในการจะส่งภาพ ซึ่งในVisual Basic ไม่มีComponentทางด้านนี้จึงต้องทำการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Down Load จาก web site ที่เกี่ยวข้องซึ่งใช้ในการ Capture Frame ภาพเพื่อใช้ในการส่งผ่านไปฝั่ง Client side

3) ส่วนการติดต่อผ่านทางอินเตอร์เน็ต ก็จะใช้ Component ที่ได้มีให้คือ Winsock Control เป็นตัวรับคำสั่งในการควบคุม และส่งข้อมูลไปยังผู้สั่งงาน การทำงานในส่วนนี้ก็สามารถทำงานได้ถูกต้อง

เมื่อได้เขียนโปรแกรมทั้งหมดแล้วในส่วนนี้จะมีการวางรูปแบบดังรูปต่อไปนี้และซอสโค้ดโปรแกรามได้แสดงไว้ในส่วนของภาคผนวกแล้ว



รูปที่ 5.10 แสดงการวางรูปแบบของโปรแกรมด้าน Server Control

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

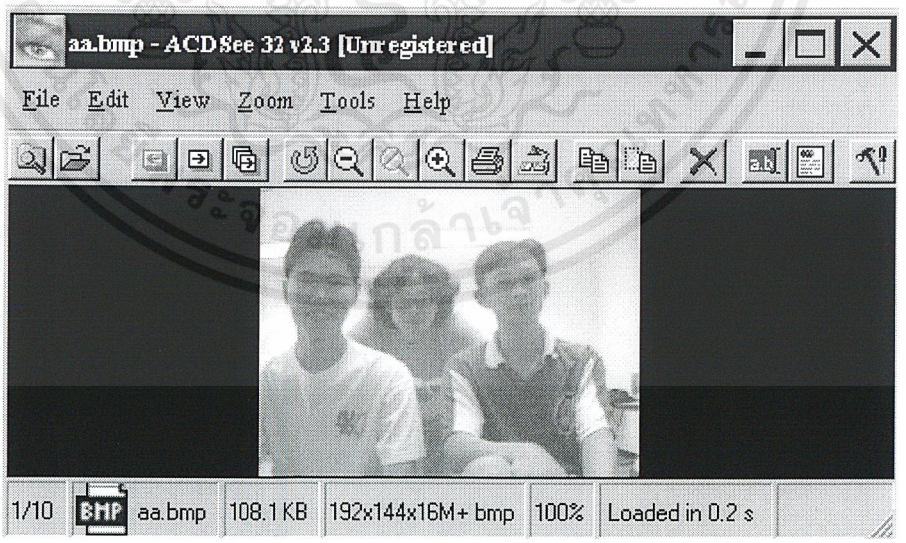
บทที่ 6
สรุปผลการทดลอง

ในการทดลองรับ-ส่งภาพและการควบคุมผ่านทางอินเตอร์เน็ตสามารถแสดงผลการทดลองได้ดังต่อไปนี้

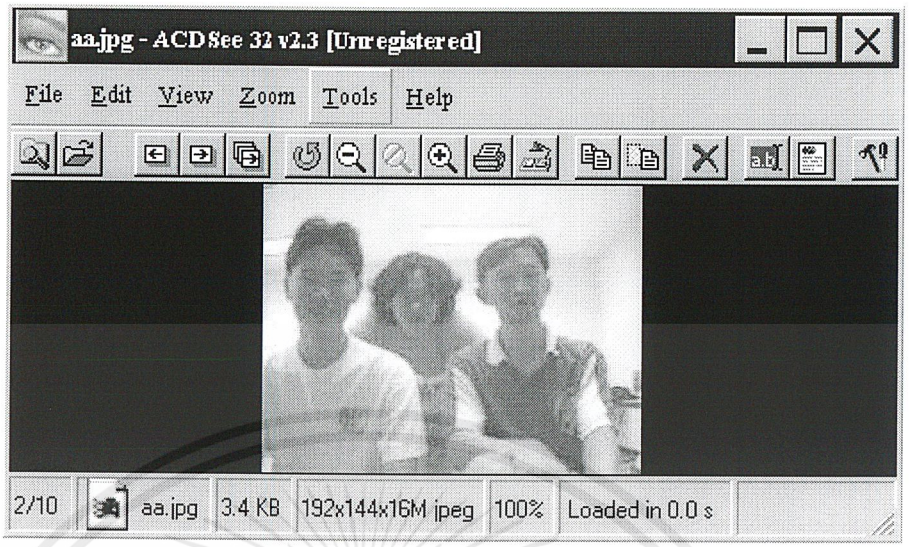
6.1 การทดลองในการส่งภาพ

6.1.1 การทดลองแปลงFile Format ภาพ

ในการส่งภาพนั้นภาพที่ได้จากกล้อง Digital เมื่อรับภาพเข้ามาในฝั่งServerนั้นมี File Format เป็น Bmp(Bitmap) เนื่องจาก File Bmp นั้นมีขนาดใหญ่จึงไม่เหมาะสมที่จะใช้ในการส่งผ่านระบบอินเตอร์เน็ต เราจึงต้องแปลง File Bmp ให้มีขนาดเล็กลงโดยให้มีคุณภาพของภาพใกล้เคียงกับภาพเดิมมากที่สุด ในโครงงานนี้ได้ทำการแปลง File Bmp เป็น JPEG (jpg) แสดงได้ดังภาพด้านล่างนี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.1 แสดงรูปภาพเปรียบเทียบระหว่าง Bitmap กับ JPEG

6.1.2 การทดลองในการส่งภาพ

โดยการใช้ทดสอบการ Capture หน่วย Frame per Second แล้วใช้ในส่งผ่านอินเตอร์เน็ต ได้ผลดังตารางดังนี้

ตารางที่ 6.1 ผลการเซ็ตค่า เฟรมที่จะใช้ส่ง

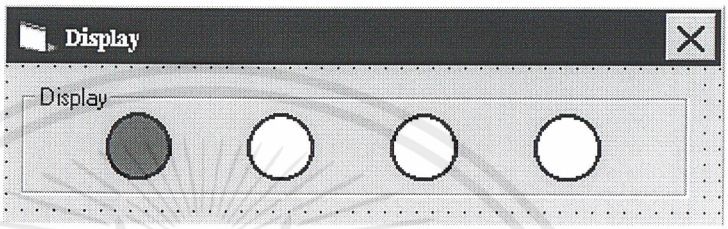
เฟรม/วินาที	คุณภาพ
1	ภาพละ 1 วินาที แสดงเป็นเฟรม ไม่ต่อเนื่อง
2	ภาพที่ได้เร็วขึ้น กว่าแต่ยังไม่ต่อเนื่อง
3	ภาพที่ได้เริ่มมีความต่อเนื่อง
4	ภาพที่ได้มีความต่อเนื่องมากขึ้น
5	ภาพที่ได้มีความต่อเนื่องมากขึ้น
6	ภาพต่อเนื่องพอรับได้
7	ภาพต่อเนื่องพอรับได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.2 การควบคุมกล้องระยะไกล

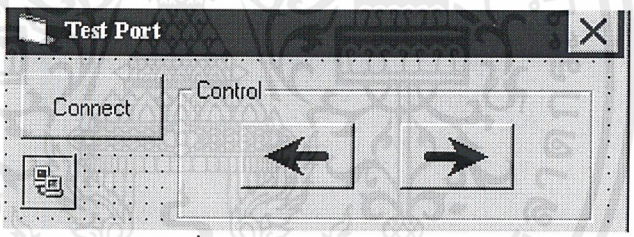
โดยในตอนแรกเราทำการทดลองเบื้องต้น โดยการเขียนโปรแกรมสำหรับ ทดสอบการส่งข้อมูลควบคุมระยะไกลบนเครื่อง PC ก่อนที่จะส่งออกพอร์ตจริงโดยโปรแกรมจะแยกเป็น 2ส่วน คือ

1. โปรแกรมทางด้าน แสดงผล (Server) จะมีหน้าตาดังรูปที่ 6.2



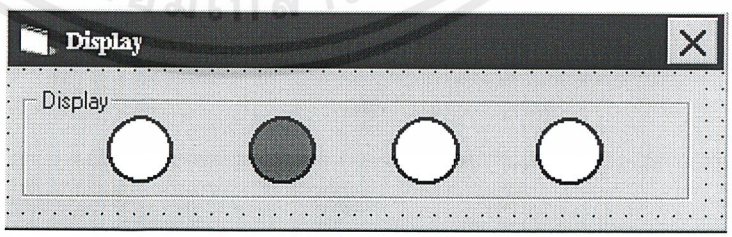
รูปที่ 6.2 ภาพทางด้านแสดงผล

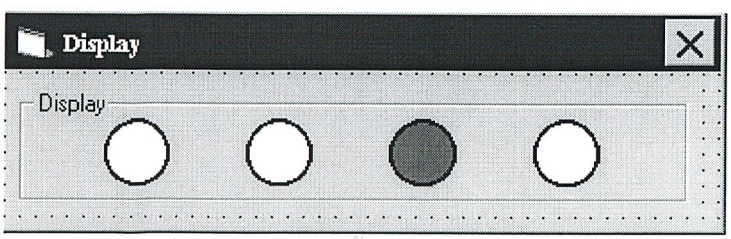
2. โปรแกรมทางด้าน ควบคุมระยะไกล (Client) จะมีหน้าตาดังรูปที่ 6.3



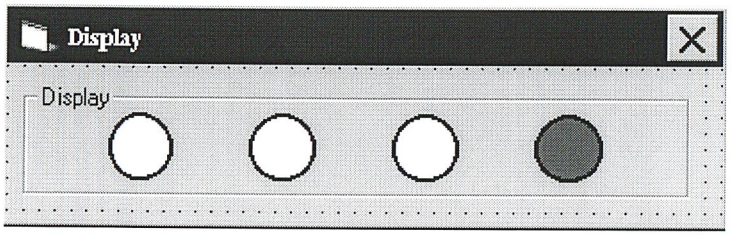
รูปที่ 6.3 ภาพหน้าจอควบคุม

จากโปรแกรมเราใช้ Winsock Control ควบคุมข้ามเครือข่ายโดยส่งข้อมูลควบคุมให้ไฟ LED ที่เขียนเสมือนไว้ บนเครื่อง PC ด้าน Display ผลที่ได้จะเป็นดังรูปที่ 6.4





เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.4 แสดงการ Display โดยกดคอนโทรปุ่มขวามือจากด้าน Client

จากการทดลองการควบคุมข้างต้น เราสามารถควบคุม LED เสมือนที่สร้างจากโปรแกรมในระยะไกลได้ (โดยทดลองในเครือข่ายอินเตอร์เน็ต) และในด้านปุ่มซ้ายก็ให้ผลได้เช่นเดียวกัน ดังนั้นเราจึงนำไปประยุกต์กับ โปรแกรมของเราได้โดย เพิ่มส่วนในการส่งข้อมูลทางด้าน Serial Port ด้วย โดยเทียบค่าที่ส่งได้ดังนี้

ตารางที่ 6.2 แสดงค่าในการส่งเพื่อทำการคอนโทรลระยะไกล

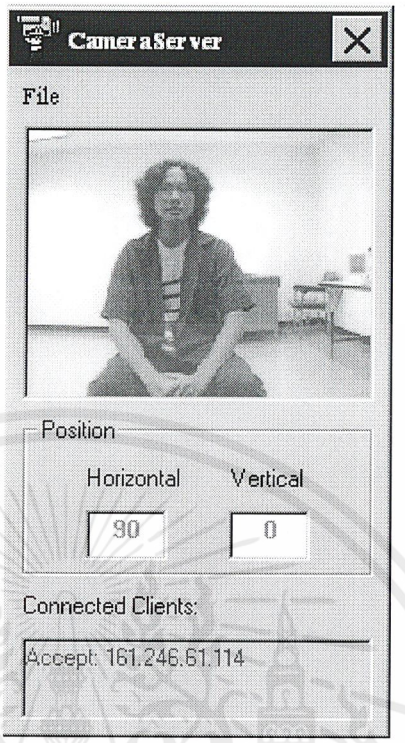
ค่าที่ส่งด้าน Client	ค่าที่ส่งด้าน Server	ค่าที่ MCS-51 ได้รับ	ผลที่ได้จากการส่งค่า
1	1	01H	บน
2	8	08H	ล่าง
3	16	10H	ซ้าย
4	128	80H	ขวา

6.3 การทดลองส่งภาพผ่านเครือข่ายแบบต่างๆ

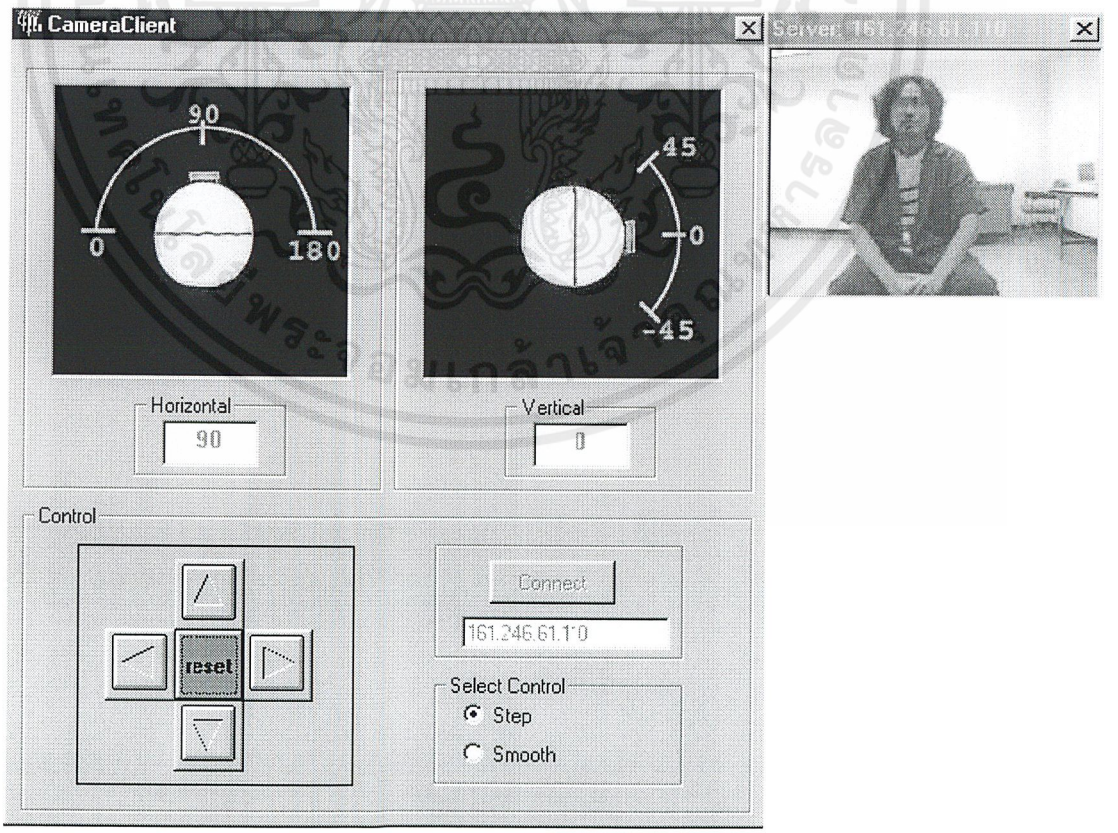
6.3.1 การส่งผ่านภาพผ่านเครือข่ายภายในสถาบัน

โดยการทดลองเปิดเครื่องทางด้าน Server Online ไว้ที่ห้องปฏิบัติการวิจัยโครงการวิจัย Reccit ชั้น 10 ส่วนเครื่องทางด้าน Client นั้น ได้ไปทดลองที่ตึก B โดยการทดลองด้าน Client ต้องทราบ IP Address ของทางด้าน Server ซึ่งเปิดรอไว้ก่อน ส่วนทางด้าน Server ไม่จำเป็นต้องทราบ เพียงแต่เปิดเครื่องรอไว้เพื่อรับบริการ ผลปรากฏเป็นที่น่าพอใจคือภาพที่ได้รับมาจากฝั่ง Sever มาทางด้าน Client ค่อยข้างจะต่อเนื่องใกล้เคียงกับทางด้าน Server

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.5 ฝั่ง Server ด้านที่รับและส่งข้อมูลไปให้ Client ที่เข้ามาขอใช้บริการ



รูปที่ 6.6 ฝั่ง Client ที่ขอใช้บริการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3.2 การทดลองส่งภาพผ่านเครือข่ายภายนอกสถาบัน

โดยการทดลองเราจะควบคุมระยะไกลโดยการเชื่อมต่อเราจะใช้ Modem ในการเข้ามาควบคุม Server ซึ่งมีชุดควบคุมกล้องซึ่ง Online อยู่ภาพที่ได้ทางฝั่ง Client จะไม่ต่อเนื่องเท่าไรนักมีการกระตุกเป็นช่วงๆ รวมทั้งส่วนการควบคุมทิศทางกล้องก็จะช้าลงเล็กน้อย แต่โดยรวมก็ยังสามารถส่งภาพมาแสดงได้ และสามารถควบคุมได้ ได้ผลเป็นที่น่าพอใจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. สมศักดิ์ อัศวกุลไพบูลย์ , "การเขียนโปรแกรมบนวินโดวส์ด้วย Visual Basic 4.0 ภาคปฏิบัติ" , ซีเอ็ดยูเคชั่น , 612 หน้า ,2540
2. สัจจะ จรัสรุ่งรวีวร , "Internet Programming ด้วย Visual Basic 6.0 และ ASP" , info PRESS , 284 หน้า , 2542
3. สุวัฒน์ ปุณณะชัยยะ , "เปิดโลกของ TCP/IP และโปรโตคอลของอินเตอร์เน็ต" , โปรวิชั่น , 312 หน้า, 2543
4. ธีรวัฒน์ ประกอบผล , " การประยุกต์ใช้งานไมโครคอนโทรลเลอร์" ,สมาคมส่งเสริมทคโนโลยี (ไทย – ญี่ปุ่น) , 235 หน้า, 2542
5. ไกรวุฒิ โรจน์ประเสริฐสุด , " เข้าใจ/ สร้าง/เล่น ไมโครโปรเซสเซอร์ 2", ซีเอ็ดยูเคชั่น, 200 หน้า, 2539
6. Steve Brown , "Visual Basic 6 In Record Time" , SYBEX , 660 p,1998

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Features

- Compatible with MCS-51™ Products
- 2K Bytes of Reprogrammable Flash Memory
  - Endurance: 1,000 Write/Erase Cycles
- 2.7V to 6V Operating Range
- Fully Static Operation: 0 Hz to 24 MHz
- Two-level Program Memory Lock
- 128 x 8-bit Internal RAM
- 15 Programmable I/O Lines
- Two 16-bit Timer/Counters
- Six Interrupt Sources
- Programmable Serial UART Channel
- Direct LED Drive Outputs
- On-chip Analog Comparator
- Low-power Idle and Power-down Modes

8-bit Microcontroller with 2K Bytes Flash

AT89C2051

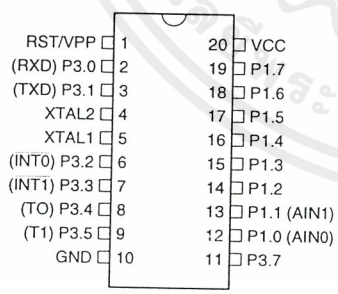
Description

The AT89C2051 is a low-voltage, high-performance CMOS 8-bit microcomputer with 2K bytes of Flash programmable and erasable read only memory (PEROM). The device is manufactured using Atmel's high-density nonvolatile memory technology and is compatible with the industry-standard MCS-51 instruction set. By combining a versatile 8-bit CPU with Flash on a monolithic chip, the Atmel AT89C2051 is a powerful microcomputer which provides a highly-flexible and cost-effective solution to many embedded control applications.

The AT89C2051 provides the following standard features: 2K bytes of Flash, 128 bytes of RAM, 15 I/O lines, two 16-bit timer/counters, a five vector two-level interrupt architecture, a full duplex serial port, a precision analog comparator, on-chip oscillator and clock circuitry. In addition, the AT89C2051 is designed with static logic for operation down to zero frequency and supports two software selectable power saving modes. The Idle Mode stops the CPU while allowing the RAM, timer/counters, serial port and interrupt system to continue functioning. The power-down mode saves the RAM contents but freezes the oscillator disabling all other chip functions until the next hardware reset.

Pin Configuration

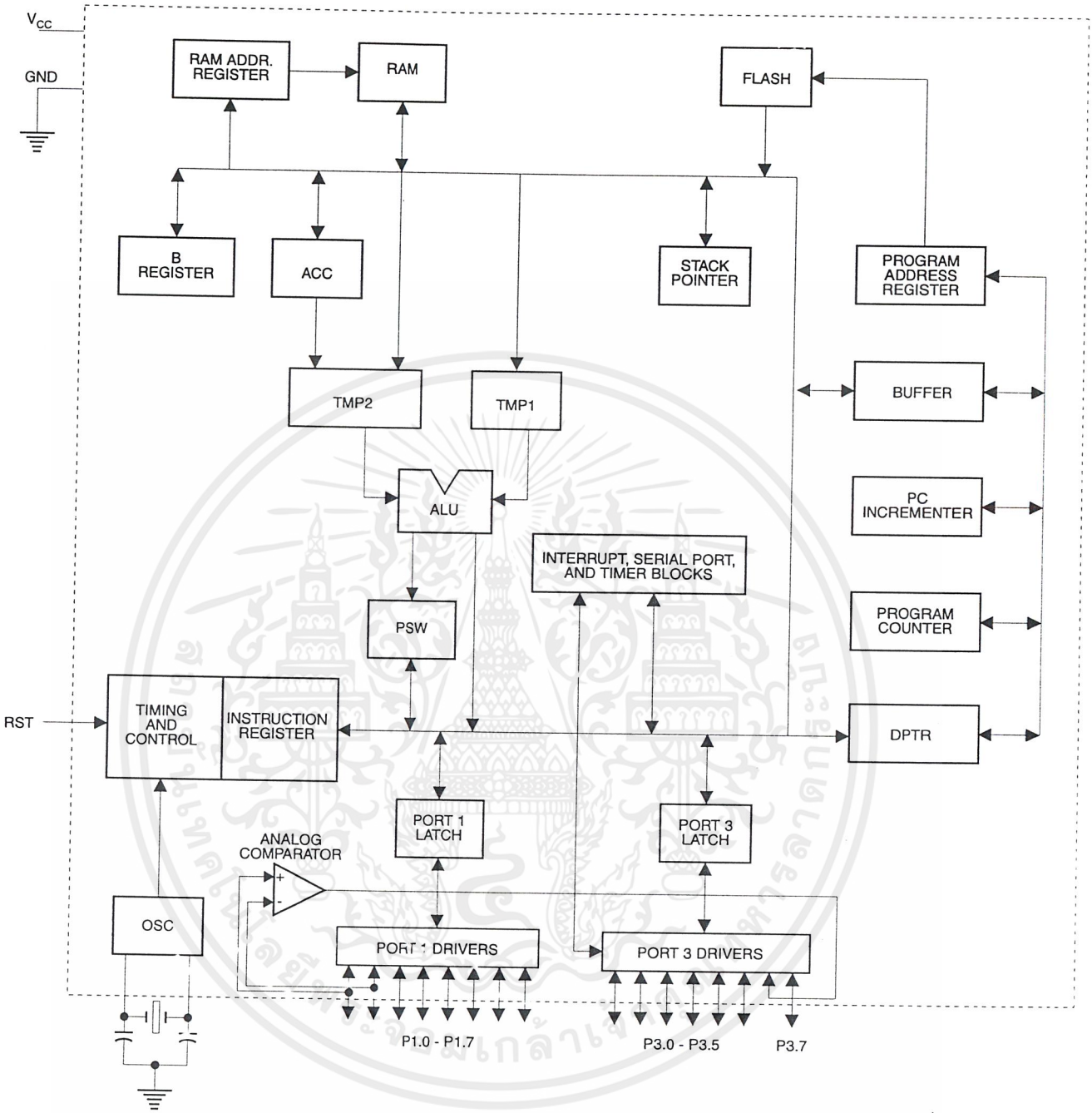
PDIP/SOIC



Rev. 0368E–02/00

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Block Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Description

VCC

Supply voltage.

GND

Ground.

Port 1

Port 1 is an 8-bit bi-irectional I/O port. Port pins P1.2 to P1.7 provide internal pullups. P1.0 and P1.1 require external pullups. P1.0 and P1.1 also serve as the positive input (AIN0) and the negative input (AIN1), respectively, of the on-chip precision analog comparator. The Port 1 output buffers can sink 20 mA and can drive LED displays directly. When 1s are written to Port 1 pins, they can be used as inputs. When pins P1.2 to P1.7 are used as inputs and are externally pulled low, they will source current (I_{IL}) because of the internal pullups.

Port 1 also receives code data during Flash programming and verification.

Port 3

Port 3 pins P3.0 to P3.5, P3.7 are seven bi-irectional I/O pins with internal pullups. P3.6 is hard-wired as an input to the output of the on-chip comparator and is not accessible as a general purpose I/O pin. The Port 3 output buffers can sink 20 mA. When 1s are written to Port 3 pins they are pulled high by the internal pullups and can be used as inputs. As inputs, Port 3 pins that are externally being pulled low will source current (I_{IL}) because of the pullups.

Port 3 also serves the functions of various special features of the AT89C2051 as listed below:

Port Pin	Alternate Functions
P3.0	RXD (serial input port)
P3.1	TXD (serial output port)
P3.2	$\overline{INT0}$ (external interrupt 0)
P3.3	$\overline{INT1}$ (external interrupt 1)
P3.4	T0 (timer 0 external input)
P3.5	T1 (timer 1 external input)

Port 3 also receives some control signals for Flash programming and verification.

RST

Reset input. All I/O pins are reset to 1s as soon as RST goes high. Holding the RST pin high for two machine cycles while the oscillator is running resets the device.

Each machine cycle takes 12 oscillator or clock cycles.

XTAL1

Input to the inverting oscillator amplifier and input to the internal clock operating circuit.

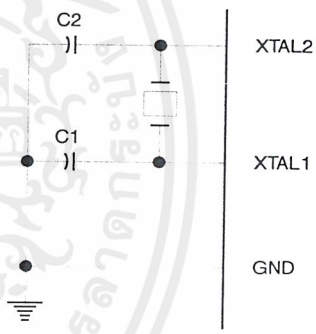
XTAL2

Output from the inverting oscillator amplifier.

Oscillator Characteristics

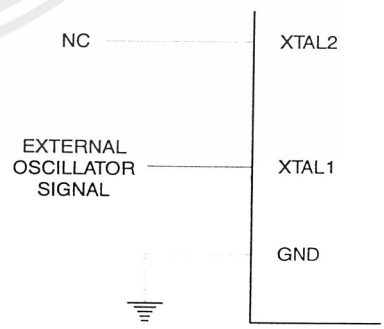
XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier which can be configured for use as an on-chip oscillator, as shown in Figure 1. Either a quartz crystal or ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected while XTAL1 is driven as shown in Figure 2. There are no requirements on the duty cycle of the external clock signal, since the input to the internal clocking circuitry is through a divide-by-two flip-flop, but minimum and maximum voltage high and low time specifications must be observed.

Figure 1. Oscillator Connections



Note: C1, C2 = 30 pF ± 10 pF for Crystals
= 40 pF ± 10 pF for Ceramic Resonators

Figure 2. External Clock Drive Configuration



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Special Function Registers

A map of the on-chip memory area called the Special Function Register (SFR) space is shown in the table below.

Note that not all of the addresses are occupied, and unoccupied addresses may not be implemented on the chip. Read accesses to these addresses will in general return random data, and write accesses will have an indeterminate effect.

User software should not write 1s to these unlisted locations, since they may be used in future products to invoke new features. In that case, the reset or inactive values of the new bits will always be 0.

Table 1. AT89C2051 SFR Map and Reset Values

0F8H									0FFH
0F0H	B 00000000								0F7H
0E8H									0EFH
0E0H	ACC 00000000								0E7H
0D8H									0DFH
0D0H	PSW 00000000								0D7H
0C8H									0CFH
0C0H									0C7H
0B8H	IP XXX00000								0BFH
0B0H	P3 11111111								0B7H
0A8H	IE 0XX00000								0AFH
0A0H									0A7H
98H	SCON 00000000	SBUF XXXXXXXX							9FH
90H	P1 11111111								97H
88H	TCON 00000000	TMOD 00000000	TL0 00000000	TL1 00000000	TH0 00000000	TH1 00000000			8FH
80H		SP 00000111	DPL 00000000	DPH 00000000				PCON 0XXX0000	87H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Restrictions on Certain Instructions

The AT89C2051 and is an economical and cost-effective member of Atmel's growing family of microcontrollers. It contains 2K bytes of flash program memory. It is fully compatible with the MCS-51 architecture, and can be programmed using the MCS-51 instruction set. However, there are a few considerations one must keep in mind when utilizing certain instructions to program this device.

All the instructions related to jumping or branching should be restricted such that the destination address falls within the physical program memory space of the device, which is 2K for the AT89C2051. This should be the responsibility of the software programmer. For example, LJMP 7E0H would be a valid instruction for the AT89C2051 (with 2K of memory), whereas LJMP 900H would not.

1. Branching instructions:

LCALL, LJMP, ACALL, AJMP, SJMP, JMP @A+DPTR

These unconditional branching instructions will execute correctly as long as the programmer keeps in mind that the destination branching address must fall within the physical boundaries of the program memory size (locations 00H to 7FFH for the 89C2051). Violating the physical space limits may cause unknown program behavior.

CJNE [...], DJNZ [...], JB, JNB, JC, JNC, JBC, JZ, JNZ With these conditional branching instructions the same rule above applies. Again, violating the memory boundaries may cause erratic execution.

For applications involving interrupts the normal interrupt service routine address locations of the 80C51 family architecture have been preserved.

2. MOVX-related instructions, Data Memory:

The AT89C2051 contains 128 bytes of internal data memory. Thus, in the AT89C2051 the stack depth is limited to 128 bytes, the amount of available RAM. External DATA memory access is not supported in this device, nor is external PROGRAM memory execution. Therefore, no MOVX [...] instructions should be included in the program.

A typical 80C51 assembler will still assemble instructions, even if they are written in violation of the restrictions mentioned above. It is the responsibility of the controller user to know the physical features and limitations of the device being used and adjust the instructions used correspondingly.

Program Memory Lock Bits

On the chip are two lock bits which can be left unprogrammed (U) or can be programmed (P) to obtain the additional features listed in the table below:

Lock Bit Protection Modes[1]

Program Lock Bits			
	LB1	LB2	Protection Type
1	U	U	No program lock features.
2	P	U	Further programming of the Flash is disabled.
3	P	P	Same as mode 2, also verify is disabled.

Note: 1. The Lock Bits can only be erased with the Chip Erase operation.

Idle Mode

In idle mode, the CPU puts itself to sleep while all the on-chip peripherals remain active. The mode is invoked by software. The content of the on-chip RAM and all the special functions registers remain unchanged during this mode. The idle mode can be terminated by any enabled interrupt or by a hardware reset.

P1.0 and P1.1 should be set to "0" if no external pullups are used, or set to "1" if external pullups are used.

It should be noted that when idle is terminated by a hardware reset, the device normally resumes program execution, from where it left off, up to two machine cycles before the internal reset algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to the port pins is not inhibited. To eliminate the possibility of an unexpected write to a port pin when Idle is terminated by reset, the instruction following the one that invokes Idle should not be one that writes to a port pin or to external memory.

Power-down Mode

In the power down mode the oscillator is stopped, and the instruction that invokes power down is the last instruction executed. The on-chip RAM and Special Function Registers retain their values until the power down mode is terminated. The only exit from power down is a hardware reset. Reset redefines the SFRs but does not change the on-chip RAM. The reset should not be activated before V_{CC} is restored to its normal operating level and must be held active long enough to allow the oscillator to restart and stabilize.

P1.0 and P1.1 should be set to "0" if no external pullups are used, or set to "1" if external pullups are used.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Programming The Flash

The AT89C2051 is shipped with the 2K bytes of on-chip PEROM code memory array in the erased state (i.e., contents = FFH) and ready to be programmed. The code memory array is programmed one byte at a time. *Once the array is programmed, to re-program any non-blank byte, the entire memory array needs to be erased electrically.*

Internal Address Counter: The AT89C2051 contains an internal PEROM address counter which is always reset to 000H on the rising edge of RST and is advanced by applying a positive going pulse to pin XTAL1.

Programming Algorithm: To program the AT89C2051, the following sequence is recommended.

1. Power-up sequence:
   Apply power between V_{CC} and GND pins
   Set RST and XTAL1 to GND
2. Set pin RST to "H"
   Set pin P3.2 to "H"
3. Apply the appropriate combination of "H" or "L" logic levels to pins P3.3, P3.4, P3.5, P3.7 to select one of the programming operations shown in the PEROM Programming Modes table.

To Program and Verify the Array:

4. Apply data for Code byte at location 000H to P1.0 to P1.7.
5. Raise RST to 12V to enable programming.
6. Pulse P3.2 once to program a byte in the PEROM array or the lock bits. The byte-write cycle is self-timed and typically takes 1.2 ms.
7. To verify the programmed data, lower RST from 12V to logic "H" level and set pins P3.3 to P3.7 to the appropiate levels. Output data can be read at the port P1 pins.
8. To program a byte at the next address location, pulse XTAL1 pin once to advance the internal address counter. Apply new data to the port P1 pins.
9. Repeat steps 5 through 8, changing data and advancing the address counter for the entire 2K bytes array or until the end of the object file is reached.
10. Power-off sequence:
    set XTAL1 to "L"
    set RST to "L"
    Turn V_{CC} power off

$\overline{\text{Data}}$ Polling: The AT89C2051 features $\overline{\text{Data}}$ Polling to indicate the end of a write cycle. During a write cycle, an attempted read of the last byte written will result in the complement of the written data on P1.7. Once the write cycle has been completed, true data is valid on all outputs, and the next cycle may begin. $\overline{\text{Data}}$ Polling may begin any time after a write cycle has been initiated.

Ready/$\overline{\text{Busy}}$: The Progress of byte programming can also be monitored by the RDY/$\overline{\text{BSY}}$ output signal. Pin P3.1 is pulled low after P3.2 goes High during programming to indicate BUSY. P3.1 is pulled High again when programming is done to indicate READY.

Program Verify: If lock bits LB1 and LB2 have not been programmed code data can be read back via the data lines for verification:

1. Reset the internal address counter to 000H by bringing RST from "L" to "H".
2. Apply the appropriate control signals for Read Code data and read the output data at the port P1 pins.
3. Pulse pin XTAL1 once to advance the internal address counter.
4. Read the next code data byte at the port P1 pins.
5. Repeat steps 3 and 4 until the entire array is read.

The lock bits cannot be verified directly. Verification of the lock bits is achieved by observing that their features are enabled.

Chip Erase: The entire PEROM array (2K bytes) and the two Lock Bits are erased electrically by using the proper combination of control signals and by holding P3.2 low for 10 ms. The code array is written with all "1"s in the Chip Erase operation and must be executed before any non-blank memory byte can be re-programmed.

Reading the Signature Bytes: The signature bytes are read by the same procedure as a normal verification of locations 000H, 001H, and 002H, except that P3.5 and P3.7 must be pulled to a logic low. The values returned are as follows.

(000H) = 1EH indicates manufactured by Atmel
(001H) = 21H indicates 89C2051

Programming Interface

Every code byte in the Flash array can be written and the entire array can be erased by using the appropriate combination of control signals. The write operation cycle is self-timed and once initiated, will automatically time itself to completion.

All major programming vendors offer worldwide support for the Atmel microcontroller series. Please contact your local programming vendor for the appropriate software revision.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Flash Programming Modes

Mode		RST/VPP	P3.2/PROG	P3.3	P3.4	P3.5	P3.7
Write Code Data[1][3]		12V		L	H	H	H
Read Code Data[1]		H	H	L	L	H	H
Write Lock	Bit - 1	12V		H	H	H	H
	Bit - 2	12V		H	H	L	L
Chip Erase		12V	(2)	H	L	L	L
Read Signature Byte		H	H	L	L	L	L

Notes:
1. The internal PEROM address counter is reset to 000H on the rising edge of RST and is advanced by a positive pulse at XTAL 1 pin.
2. Chip Erase requires a 10 ms $\overline{PROG}$ pulse.
3. P3.1 is pulled Low during programming to indicate RDY/$\overline{BSY}$.

Figure 3. Programming the Flash Memory

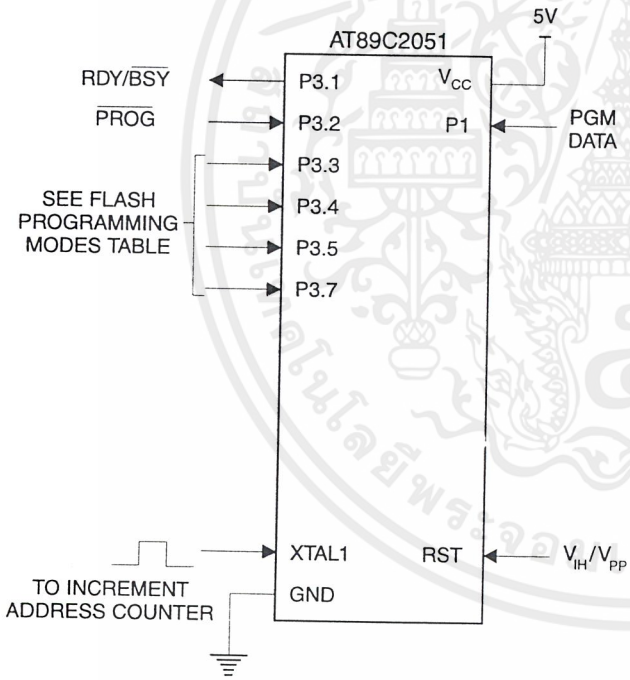
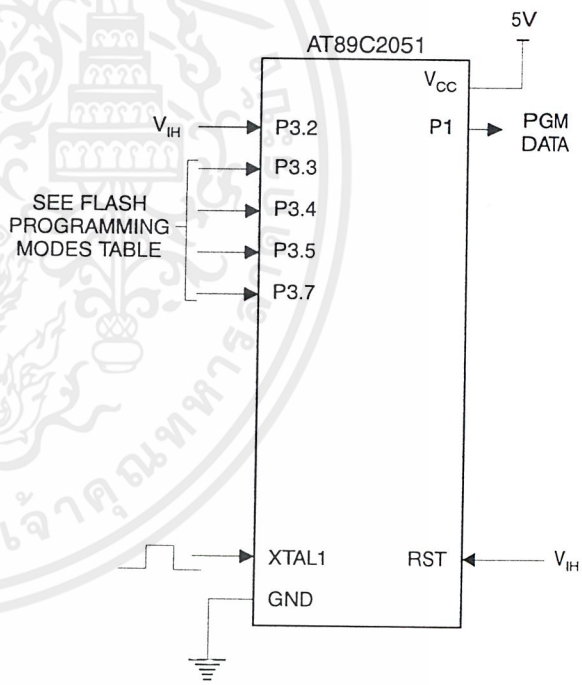


Figure 4. Verifying the Flash Memory



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

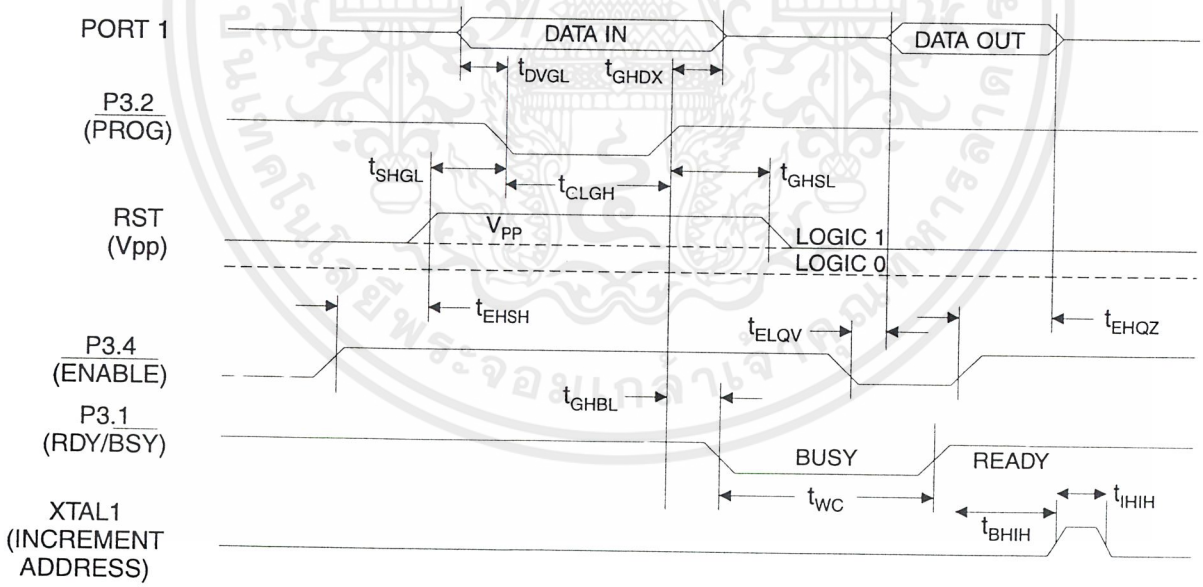
Flash Programming and Verification Characteristics

T_A = 0°C to 70°C, V_{CC} = 5.0 ± 10%

Symbol	Parameter	Min	Max	Units
V_{PP}	Programming Enable Voltage	11.5	12.5	V
I_{PP}	Programming Enable Current		250	µA
t_{DVGL}	Data Setup to $\overline{PROG}$ Low	1.0		µs
t_{GHDX}	Data Hold after $\overline{PROG}$	1.0		µs
t_{EHSH}	P3.4 ($\overline{ENABLE}$) High to V_{PP}	1.0		µs
t_{SHGL}	V_{PP} Setup to $\overline{PROG}$ Low	10		µs
t_{GHSL}	V_{PP} Hold after $\overline{PROG}$	10		µs
t_{GLGH}	$\overline{PROG}$ Width	1	110	µs
t_{ELQV}	$\overline{ENABLE}$ Low to Data Valid		1.0	µs
t_{EHQZ}	Data Float after $\overline{ENABLE}$	0	1.0	µs
t_{GHBL}	$\overline{PROG}$ High to $\overline{BUSY}$ Low		50	ns
t_{WC}	Byte Write Cycle Time		2.0	ms
t_{BHIH}	RDY/$\overline{BSY}$\ to Increment Clock Delay	1.0		µs
t_{IHIL}	Increment Clock High	200		ns

Note: 1. Only used in 12-volt programming mode.

Flash Programming and Verification Waveforms



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Absolute Maximum Ratings*

Operating Temperature	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-1.0V to +7.0V
Maximum Operating Voltage	6.6V
DC Output Current	25.0 mA

*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

DC Characteristics

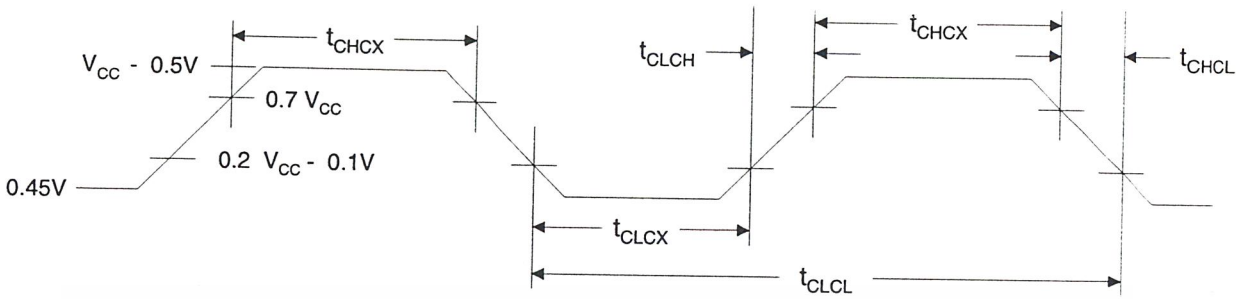
T_A = -40°C to 85°C, V_{CC} = 2.0V to 6.0V (unless otherwise noted)

Symbol	Parameter	Condition	Min	Max	Units
V_{IL}	Input Low-voltage		-0.5	0.2 V_{CC} - 0.1	V
V_{IH}	Input High-voltage	(Except XTAL1, RST)	0.2 V_{CC} + 0.9	V_{CC} + 0.5	V
V_{IH1}	Input High-voltage	(XTAL1, RST)	0.7 V_{CC}	V_{CC} + 0.5	V
V_{OL}	Output Low-voltage[1] (Ports 1, 3)	I_{OL} = 20 mA, V_{CC} = 5V I_{OL} = 10 mA, V_{CC} = 2.7V		0.5	V
V_{OH}	Output High-voltage (Ports 1, 3)	I_{OH} = -80 µA, V_{CC} = 5V ± 10%	2.4		V
		I_{OH} = -30 µA	0.75 V_{CC}		V
		I_{OH} = -12 µA	0.9 V_{CC}		V
I_{IL}	Logical 0 Input Current (Ports 1, 3)	V_{IN} = 0.45V		-50	µA
I_{TL}	Logical 1 to 0 Transition Current (Ports 1, 3)	V_{IN} = 2V, V_{CC} = 5V ± 10%		-750	µA
I_{LI}	Input Leakage Current (Port P1.0, P1.1)	0 < V_{IN} < V_{CC}		±10	µA
V_{OS}	Comparator Input Offset Voltage	V_{CC} = 5V		20	mV
V_{CM}	Comparator Input Common Mode Voltage		0	V_{CC}	V
RRST	Reset Pull-down Resistor		50	300	KΩ
C_{IO}	Pin Capacitance	Test Freq. = 1 MHz, T_A = 25°C		10	pF
I_{CC}	Power Supply Current	Active Mode, 12 MHz, V_{CC} = 6V/3V		15/5.5	mA
		Idle Mode, 12 MHz, V_{CC} = 6V/3V P1.0 & P1.1 = 0V or V_{CC}		5/1	mA
	Power-down Mode[2]	V_{CC} = 6V P1.0 & P1.1 = 0V or V_{CC}		100	µA
		V_{CC} = 3V P1.0 & P1.1 = 0V or V_{CC}		20	µA

Notes:
1. Under steady state (non-transient) conditions, I_{OL} must be externally limited as follows:
   Maximum I_{OL} per port pin: 20 mA
   Maximum total I_{OL} for all output pins: 80 mA
   If I_{OL} exceeds the test condition, V_{OL} may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
2. Minimum V_{CC} for Power-down is 2V.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

External Clock Drive Waveforms



External Clock Drive

Symbol	Parameter	V_{CC} = 2.7V to 6.0V		V_{CC} = 4.0V to 6.0V		Units
		Min	Max	Min	Max	
$1/t_{CLCL}$	Oscillator Frequency	0	12	0	24	MHz
t_{CLCL}	Clock Period	83.3		41.6		ns
t_{CHCX}	High Time	30		15		ns
t_{CLCX}	Low Time	30		15		ns
t_{CLCH}	Rise Time		20		20	ns
t_{CHCL}	Fall Time		20		20	ns

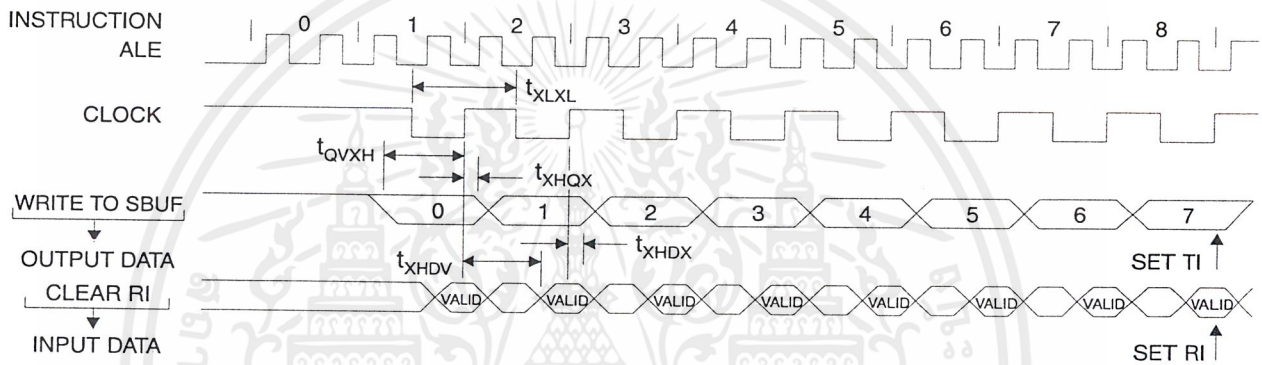
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Serial Port Timing: Shift Register Mode Test Conditions

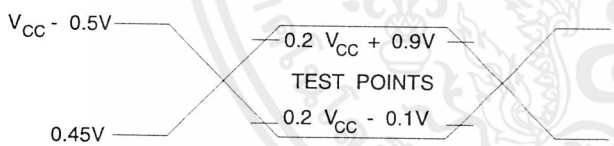
V_{CC} = 5.0V ± 20%; Load Capacitance = 80 pF

Symbol	Parameter	12 MHz Osc		Variable Oscillator		Units
		Min	Max	Min	Max	
t_{XLXL}	Serial Port Clock Cycle Time	1.0		$12t_{CLCL}$		µs
t_{QVXH}	Output Data Setup to Clock Rising Edge	700		$10t_{CLCL}$-133		ns
t_{XHQX}	Output Data Hold after Clock Rising Edge	50		$2t_{CLCL}$-117		ns
t_{XHDX}	Input Data Hold after Clock Rising Edge	0		0		ns
t_{XHDV}	Clock Rising Edge to Input Data Valid		700		$10t_{CLCL}$-133	ns

Shift Register Mode Timing Waveforms

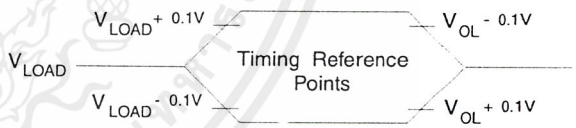


AC Testing Input/Output Waveforms[1]



Note: 1. AC Inputs during testing are driven at V_{CC} - 0.5V for a logic 1 and 0.45V for a logic 0. Timing measurements are made at V_{IH} min. for a logic 1 and V_{IL} max. for a logic 0.

Float Waveforms[1]

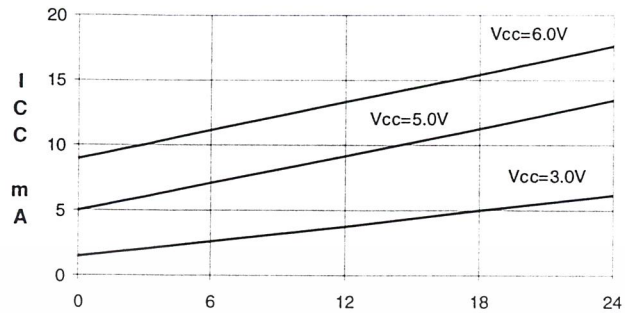


Note: 1. For timing purposes, a port pin is no longer floating when a 100 mV change from load voltage occurs. A port pin begins to float when 100 mV change frothe loaded V_{OH}/V_{OL} level occurs.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

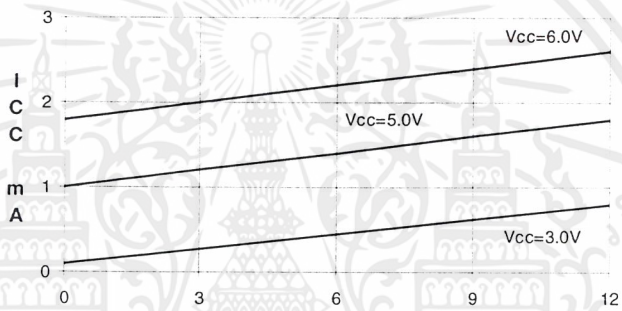
AT89C2051

TYPICAL ICC - ACTIVE (85°C)



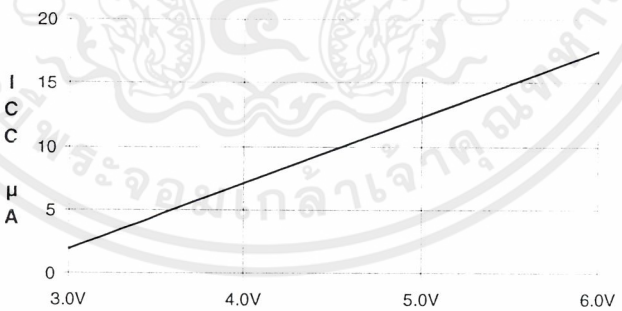
AT89C2051

TYPICAL ICC - IDLE (85°C)



AT89C2051

TYPICAL ICC vs. VOLTAGE- POWER DOWN (85°C)



Notes:
1. XTAL1 tied to GND for I_{CC} (power-down)
2. P.1.0 and P1.1 = V_{CC} or GND
3. Lock bits programmed

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Ordering Information

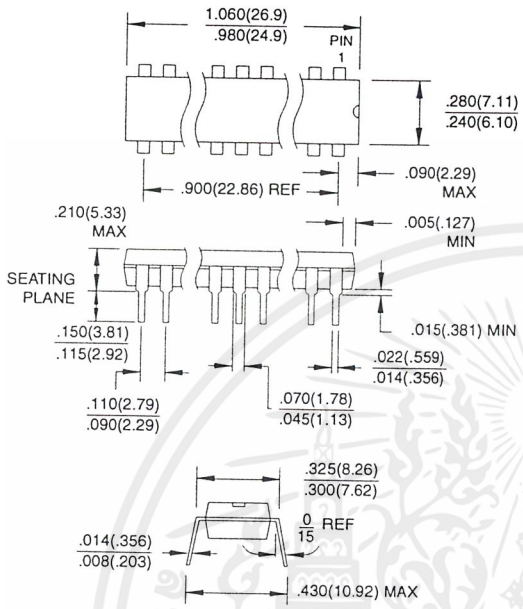
Speed (MHz)	Power Supply	Ordering Code	Package	Operation Range
12	2.7V to 6.0V	AT89C2051-12PC AT89C2051-12SC	20P3 20S	Commercial (0°C to 70°C)
		AT89C2051-12PI AT89C2051-12SI	20P3 20S	Industrial (-40°C to 85°C)
24	4.0V to 6.0V	AT89C2051-24PC AT89C2051-24SC	20P3 20S	Commercial (0°C to 70°C)
		AT89C2051-24PI AT89C2051-24SI	20P3 20S	Industrial (-40°C to 85°C)

Package Type	
20P3	20-lead, 0.300" Wide, Plastic Dual In-line Package (PDIP)
20S	20-lead, 0.300" Wide, Plastic Gull Wing Small Outline (SOIC)

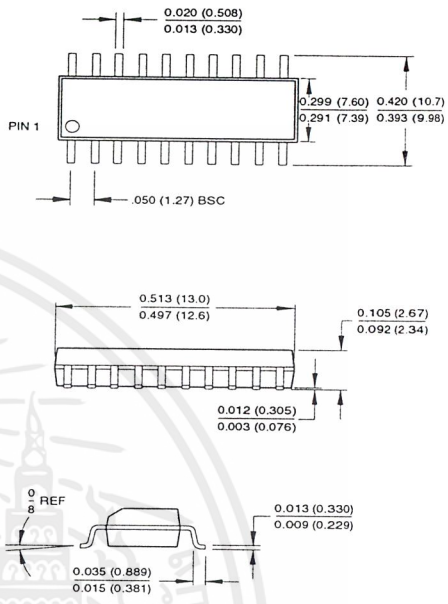
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Packaging Information

20P3, 20-lead, 0.300" Wide, Plastic Dual Inline Package (PDIP)
Dimensions in Inches and (Millimeters)
JEDEC STANDARD MS-001 AD



20S, 20-lead, 0.300" Wide, Plastic Gull WIng Small Outline (SOIC)
Dimensions in Inches and (Millimeters)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Atmel Headquarters

Corporate Headquarters
2325 Orchard Parkway
San Jose, CA 95131
TEL (408) 441-0311
FAX (408) 487-2600

Europe
Atmel U.K., Ltd.
Coliseum Business Centre
Riverside Way
Camberley, Surrey GU15 3YL
England
TEL (44) 1276-686-677
FAX (44) 1276-686-697

Asia
Atmel Asia, Ltd.
Room 1219
Chinachem Golden Plaza
77 Mody Road Tsimhatsui
East Kowloon
Hong Kong
TEL (852) 2721-9778
FAX (852) 2722-1369

Japan
Atmel Japan K.K.
9F, Tonetsu Shinkawa Bldg.
1-24-8 Shinkawa
Chuo-ku, Tokyo 104-0033
Japan
TEL (81) 3-3523-3551
FAX (81) 3-3523-7581

Atmel Operations

Atmel Colorado Springs
1150 E. Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906
TEL (719) 576-3300
FAX (719) 540-1759

Atmel Rousset
Zone Industrielle
13106 Rousset Cedex
France
TEL (33) 4-4253-6000
FAX (33) 4-4253-6001

Fax-on-Demand
North America:
1-(800) 292-8635
International:
1-(408) 441-0732

e-mail
literature@atmel.com

Web Site
http://www.atmel.com

BBS
1-(408) 436-4309

© Atmel Corporation 2000.
Atmel Corporation makes no warranty for the use of its products, other than those expressly contained in the Company's standard warranty which is detailed in Atmel's Terms and Conditions located on the Company's web site. The Company assumes no responsibility for any errors which may appear in this document, reserves the right to change devices or specifications detailed herein at any time without notice, and does not make any commitment to update the information contained herein. No licenses to patents or other intellectual property of Atmel are granted by the Company in connection with the sale of Atmel products, expressly or by implication. Atmel's products are not authorized for use as critical components in life support devices or systems.

Marks bearing ® and/or ™ are registered trademarks and trademarks of Atmel Corporation.

Terms and product names in this document may be trademarks of others.

Printed on recycled paper.

0368E–02/00/xM

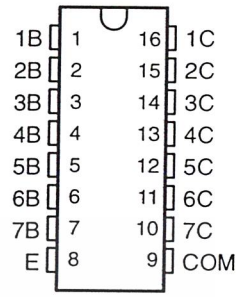
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SLRS027 – DECEMBER 1976 – REVISED APRIL 1993

HIGH-VOLTAGE HIGH-CURRENT DARLINGTON TRANSISTOR ARRAYS

- 500-mA Rated Collector Current (Single Output)
- High-Voltage Outputs . . . 50 V
- Output Clamp Diodes
- Inputs Compatible With Various Types of Logic
- Relay Driver Applications
- Designed to Be Interchangeable With Sprague ULN2001A Series

D OR N PACKAGE
(TOP VIEW)

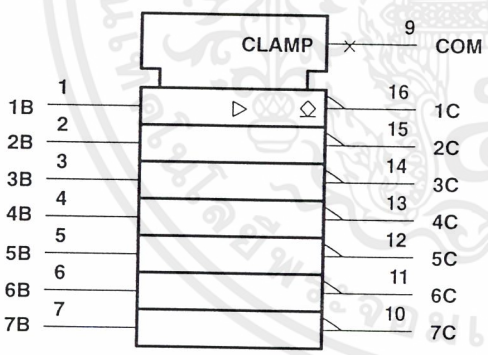


description

The ULN2001A, ULN2002A, ULN2003A, and ULN2004A are monolithic high-voltage, high-current Darlington transistor arrays. Each consists of seven npn Darlington pairs that feature high-voltage outputs with common-cathode clamp diodes for switching inductive loads. The collector-current rating of a single Darlington pair is 500 mA. The Darlington pairs may be paralleled for higher current capability. Applications include relay drivers, hammer drivers, lamp drivers, display drivers (LED and gas discharge), line drivers, and logic buffers. For 100-V (otherwise interchangeable) versions, see the SN75465 through SN75469.

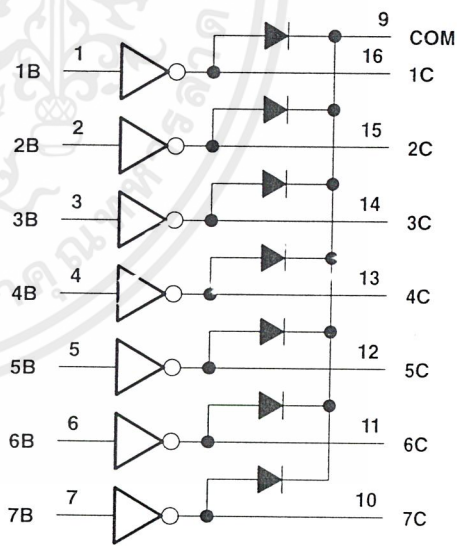
The ULN2001A is a general-purpose array and can be used with TTL and CMOS technologies. The ULN2002A is specifically designed for use with 14- to 25-V PMOS devices. Each input of this device has a zener diode and resistor in series to control the input current to a safe limit. The ULN2003A has a 2.7-kΩ series base resistor for each Darlington pair for operation directly with TTL or 5-V CMOS devices. The ULN2004A has a 10.5-kΩ series base resistor to allow its operation directly from CMOS devices that use supply voltages of 6 to 15 V. The required input current of the ULN2004A is below that of the ULN2003A, and the required voltage is less than that required by the ULN2002A.

logic symbol†



† This symbol is in accordance with ANSI/IEEE Std 91-1984 and IEC Publication 617-12.

logic diagram



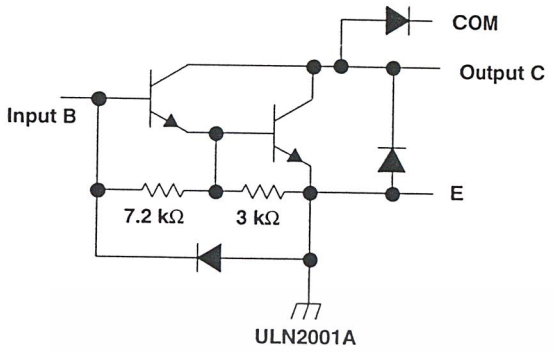
PRODUCTION DATA information is current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

Copyright © 1993, Texas Instruments Incorporated

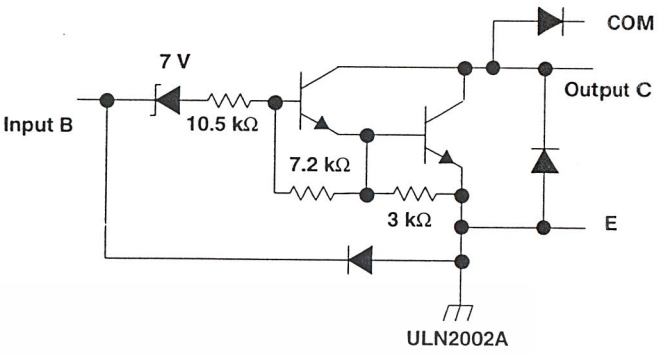
TEXAS INSTRUMENTS
POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

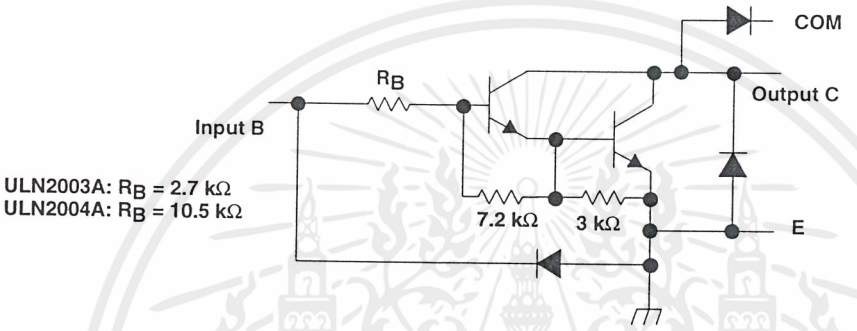
schematics (each Darlington pair)



ULN2001A



ULN2002A



ULN2003A, ULN2004A

All resistor values shown are nominal.

absolute maximum ratings at 25°C free-air temperature (unless otherwise noted)

Collector-emitter voltage . . . . . 50 V
Input voltage, V_I (see Note 1) . . . . . 30 V
Peak collector current (see Figures 14 and 15) . . . . . 500 mA
Output clamp current, I_{OK} . . . . . 500 mA
Total emitter-terminal current . . . . . −2.5 A
Continuous total power dissipation . . . . . See Dissipation Rating Table
Operating free-air temperature range, T_A . . . . . −20°C to 85°C
Storage temperature range, T_{stg} . . . . . −65°C to 150°C
Lead temperature 1,6 mm (1/16 inch) from case for 10 seconds . . . . . 260°C

NOTE 1: All voltage values are with respect to the emitter/substrate terminal E, unless otherwise noted.

DISSIPATION RATING TABLE

PACKAGE	T_A = 25°C POWER RATING	DERATING FACTOR ABOVE T_A = 25°C	T_A = 85°C POWER RATING
D	950 mW	7.6 mW/°C	494 mW
N	1150 mW	9.2 mW/°C	598 mW

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

electrical characteristics, T_A = 25°C (unless otherwise noted)

PARAMETER		TEST FIGURE	TEST CONDITIONS		ULN2001A			ULN2002A			UNIT
					MIN	TYP	MAX	MIN	TYP	MAX	
$V_{I(on)}$	On-state input voltage	6	V_{CE} = 2 V,	I_C = 300 mA						13	V
$V_{CE(sat)}$	Collector-emitter saturation voltage	5	I_I = 250 µA,	I_C = 100 mA		0.9	1.1		0.9	1.1	V
			I_I = 350 µA,	I_C = 200 mA		1	1.3		1	1.3	
			I_I = 500 µA,	I_C = 350 mA		1.2	1.6		1.2	1.6	
V_F	Clamp forward voltage	8	I_F = 350 mA			1.7	2		1.7	2	V
I_{CEX}	Collector cutoff current	1	V_{CE} = 50 V,	I_I = 0			50			50	µA
		2	V_{CE} = 50 V, T_A = 70°C	I_I = 0			100			100	
				V_I = 6 V						500	
$I_{I(off)}$	Off-state input current	3	V_{CE} = 50 V, T_A = 70°C	I_C = 500 µA,	50	65		50	65		µA
I_I	Input current	4	V_I = 17 V						0.82	1.25	mA
I_R	Clamp reverse current	7	V_R = 50 V,	TA = 70°C			100			100	µA
			V_R = 50 V				50			50	
h_{FE}	Static forward current transfer ratio	5	V_{CE} = 2 V,	I_C = 350 mA	1000						
C_i	Input capacitance		V_I = 0,	f = 1 MHz		15	25		15	25	pF

electrical characteristics, T_A = 25°C (unless otherwise noted)

PARAMETER		TEST FIGURE	TEST CONDITIONS		ULN2003A			ULN2004A			UNIT
					MIN	TYP	MAX	MIN	TYP	MAX	
$V_{I(on)}$	On-state input voltage	6	V_{CE} = 2 V	I_C = 125 mA						5	V
				I_C = 200 mA			2.4			6	
				I_C = 250 mA			2.7				
				I_C = 275 mA						7	
				I_C = 300 mA			3				
				I_C = 350 mA						8	
$V_{CE(sat)}$	Collector-emitter saturation voltage	5	I_I = 250 µA,	I_C = 100 mA		0.9	1.1		0.9	1.1	V
			I_I = 350 µA,	I_C = 200 mA		1	1.3		1	1.3	
			I_I = 500 µA,	I_C = 350 mA		1.2	1.6		1.2	1.6	
I_{CEX}	Collector cutoff current	1	V_{CE} = 50 V,	I_I = 0			50			50	µA
		2	V_{CE} = 50 V, T_A = 70°C	I_I = 0			100			100	
				V_I = 1 V						500	
V_F	Clamp forward voltage	8	I_F = 350 mA			1.7	2		1.7	2	V
$I_{I(off)}$	Off-state input current	3	V_{CE} = 50 V, T_A = 70°C	I_C = 500 µA,	50	65		50	65		µA
I_I	Input current	4	V_I = 3.85 V			0.93	1.35				mA
			V_I = 5 V						0.35	0.5	
			V_I = 12 V						1	1.45	
I_R	Clamp reverse current	7	V_R = 50 V				50			50	µA
			V_R = 50 V,	T_A = 70°C			100			100	
C_i	Input capacitance		V_I = 0,	f = 1 MHz		15	25		15	25	pF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TYPICAL CHARACTERISTICS

COLLECTOR-EMITTER SATURATION VOLTAGE vs COLLECTOR CURRENT (ONE DARLINGTON)

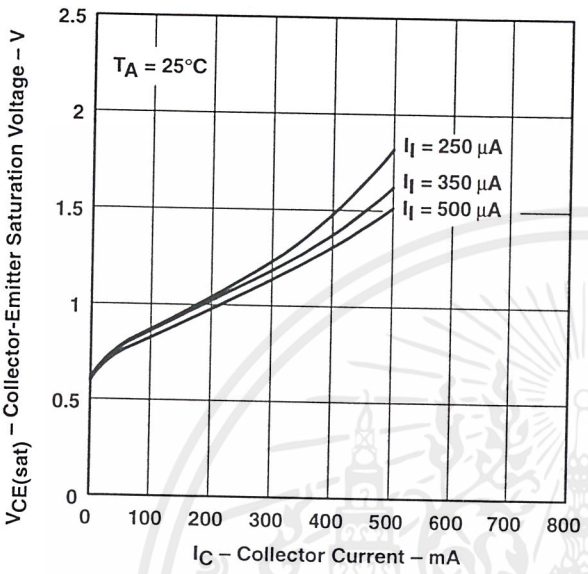


Figure 11

COLLECTOR-EMITTER SATURATION VOLTAGE vs TOTAL COLLECTOR CURRENT (TWO DARLINGTONS PARALLELED)

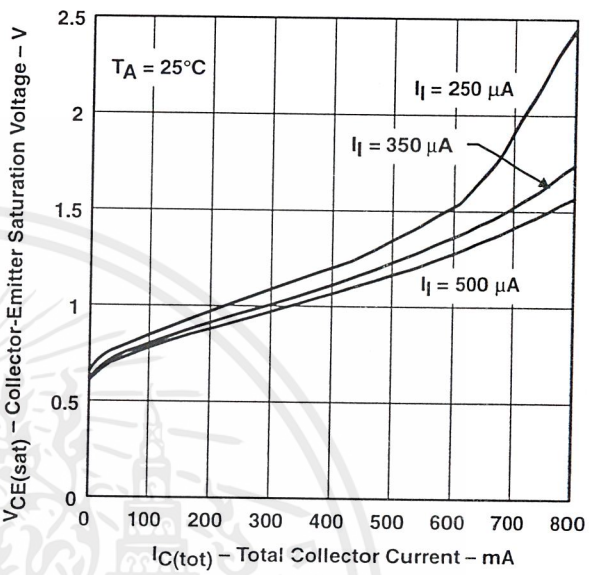


Figure 12

COLLECTOR CURRENT vs INPUT CURRENT

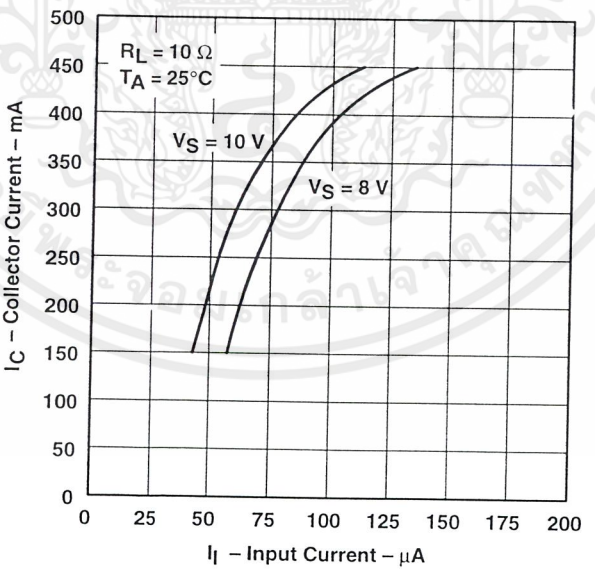


Figure 13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

THERMAL INFORMATION

D PACKAGE
MAXIMUM COLLECTOR CURRENT
vs
DUTY CYCLE

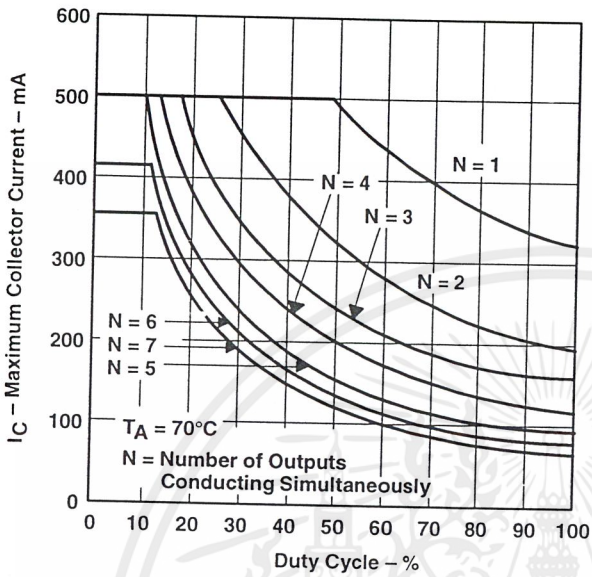


Figure 14

N PACKAGE
MAXIMUM COLLECTOR CURRENT
vs
DUTY CYCLE

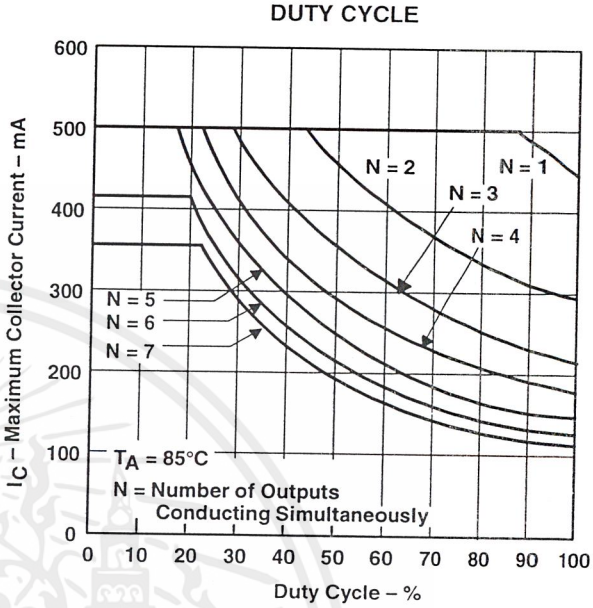


Figure 15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

APPLICATION INFORMATION

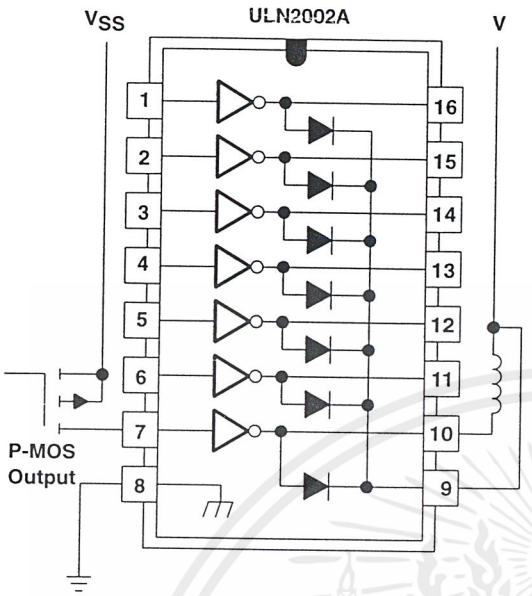


Figure 16. P-MOS to Load

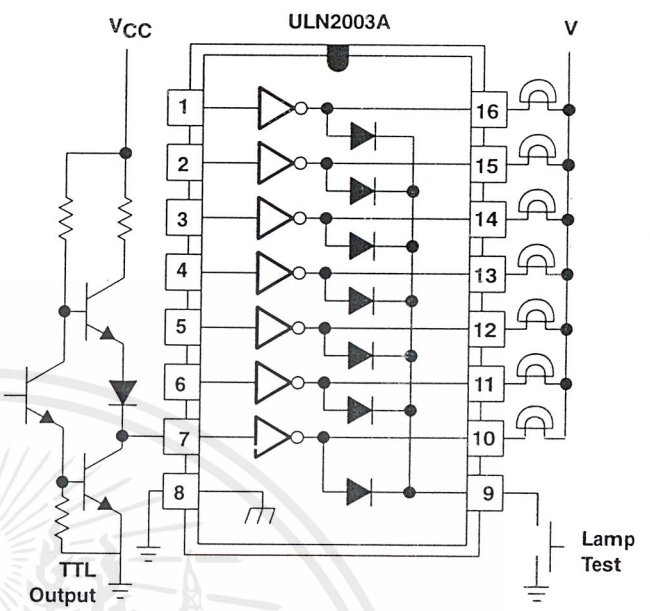


Figure 17. TTL to Load

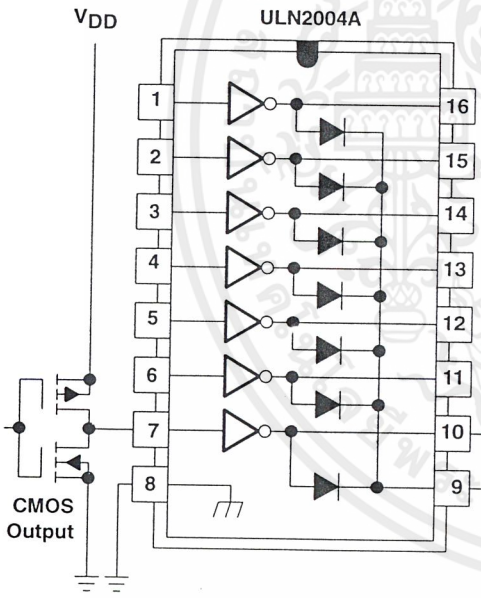


Figure 18. Buffer for Higher Current Loads

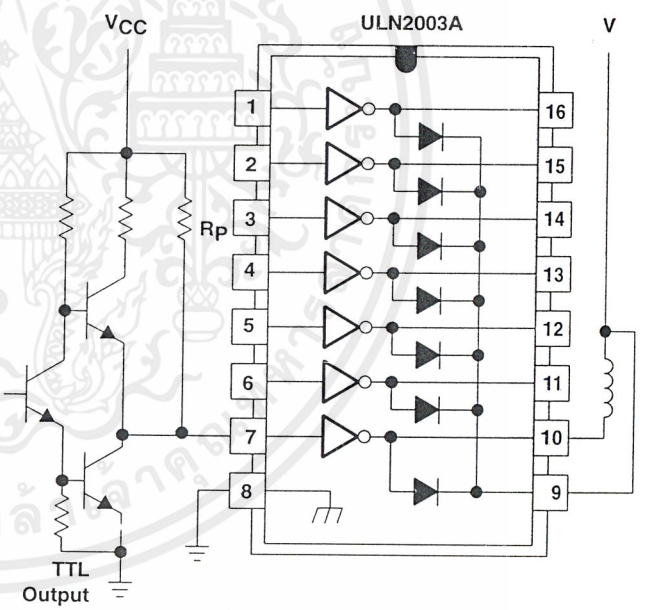


Figure 19. Use of Pullup Resistors to Increase Drive Current

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

IMPORTANT NOTICE

Texas Instruments and its subsidiaries (TI) reserve the right to make changes to their products or to discontinue any product or service without notice, and advise customers to obtain the latest version of relevant information to verify, before placing orders, that information being relied on is current and complete. All products are sold subject to the terms and conditions of sale supplied at the time of order acknowledgement, including those pertaining to warranty, patent infringement, and limitation of liability.

TI warrants performance of its semiconductor products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

CERTAIN APPLICATIONS USING SEMICONDUCTOR PRODUCTS MAY INVOLVE POTENTIAL RISKS OF DEATH, PERSONAL INJURY, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE ("CRITICAL APPLICATIONS"). TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS. INCLUSION OF TI PRODUCTS IN SUCH APPLICATIONS IS UNDERSTOOD TO BE FULLY AT THE CUSTOMER'S RISK.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards must be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance or customer product design. TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used. TI's publication of information regarding any third party's products or services does not constitute TI's approval, warranty or endorsement thereof.

Copyright © 1998, Texas Instruments Incorporated

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้