

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบควบคุมเครื่องใช้ไฟฟ้าและรักษาความปลอดภัยภายในบ้าน

Appliances and indoor security control system



โดย

นาย ภูมิสรรค์ วงศ์ชื้อศักดิ์

นาย ภูริวิทย์ แสงกาสนีย์

นาย รัฐสรณ์ ทรงศิริ

รฟ.
๑๖๗๑
๑๕๔๙

เลขหมู่.....

เลขทะเบียน..... 72832

วัน,เดือน,ปี 23 ส.ย. 2550

b. 11273066

i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาดมหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบควบคุมเครื่องใช้ไฟฟ้าและรักษาความปลอดภัยภายในบ้าน

Appliances and indoor security control system



ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของวิชา 01044058 PROJECT 2

ของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ประจำภาคเรียนที่ 2 ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายงานเรื่อง ระบบควบคุมเครื่องใช้ไฟฟ้าภายในบ้าน

Home Electric Appliance Control System

จัดทำโดย นาย ภูมิสรรัตน์ วงศ์ชื้อสตัย 46010590
นาย ภูริวิทย์ แสงกาศนีย์ 46010592
นาย รัฐสรณ์ ทรงศิริ 46010635

อาจารย์ที่ปรึกษา อาจารย์ รศ.ดร. มนัส สังวรศิลป์

รายงานฉบับนี้ผ่านการตรวจสอบโดยอาจารย์ที่ปรึกษาแล้ว

ลงชื่อ.....อาจารย์ที่ปรึกษา
(อาจารย์ รศ.ดร. มนัส สังวรศิลป์) .
วันที่/...../.....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบควบคุมเครื่องใช้ไฟฟ้าและรักษาความปลอดภัยภายในบ้าน

ภูมิสรรค์ วงศ์เชื้อสัจย์ 46010590

ภูริวิทย์ แสงกาสนีย์ 46010592

รัฐสรณ์ ทรงศิริ 46010635

บทคัดย่อ

เนื่องจากในปัจจุบันผู้อยู่อาศัยตามบ้านต้องการทั้งความสะดวกสบายและความปลอดภัย จึงทำให้เกิดแนวคิดที่ว่า หากมีระบบควบคุมที่สามารถให้ความสะดวกสบายและความปลอดภัยได้ภายในระบบเดียวกัน โดยสามารถควบคุมเครื่องใช้ไฟฟ้าทั้งหมดได้จากจุดเดียวและมีการแจ้งเตือนเมื่อเกิดไฟไหม้หรือมีการบุกรุกก็จะเป็นสิ่งที่ดี

โครงการนี้เป็นโครงการที่มีจุดประสงค์เพื่อใช้ในการควบคุมการเปิดปิดเครื่องใช้ไฟฟ้า ณ จุดต่าง ๆ ภายในบ้าน และรักษาความปลอดภัยด้วยการเตือนการเกิดไฟไหม้และการถูกบุกรุก โดยอาศัยการเชื่อมต่อ microcontroller หลายตัวและคอมพิวเตอร์ เป็นระบบเครือข่าย เพื่อควบคุมการทำงานของจุดต่าง ๆ ภายในบ้านและส่งข้อความแจ้งเตือนไปยังโทรศัพท์มือถือของผู้ใช้งานหรือเจ้าของบ้าน โดยส่งเป็น SMS ผ่านทางโมดูลโทรศัพท์มือถือ ซึ่งอยู่ในระบบเครือข่ายด้วยเช่นกัน นอกจากนี้ระบบควบคุมนี้ยังสามารถเปิดไฟแบบอัตโนมัติเมื่อถึงเวลากลางคืน และสามารถปิดไฟได้เมื่อถึงเวลาเช้า

Appliances and indoor security control system

Poomsan Wongsuesat 46010590

Phurivit Sangkatsanee 46010592

Ratthasorn Songsiri 46010635

Abstract

This project represents the control system which provides convenience and safety. It controls all appliances and has fire or attack alarm.

The purposes are, to control appliances turn on-turn off anywhere and alarm about its safety. Microcontrollers and computer are network which control and send messaging to alarm the owner. Moreover it can automatic turn on and turn off the light at night and in the morning.

สารบัญ

1 บทนำ	1
2 หลักการ	2
2.1 Visual Studio	2
2.1.1 การพัฒนาอ็พพลิเคชันด้วย Visual Studio	2
2.1.2 จุดเด่นของของ Visual Basic	3
2.2 ความรู้พื้นฐานเกี่ยวกับพอร์ตขนาน	4
2.2.1 รู้จักกับพอร์ตขนาน	4
2.2.2 ลักษณะของพอร์ตขนาน	5
2.2.3 ลักษณะสัญญาณของพอร์ตขนาน	7
2.2.4 STATUS PORT	7
2.2.5 CONTROL PORT	8
2.2.6 การติดตั้งพอร์ตขนาน	9
2.3 การเขียนโปรแกรมติดต่อพอร์ตขนาน	12
2.4 พอร์ตอนุกรมมาตรฐาน RS-232	13
2.4.1 RS-232 serial connector pin assignment	13
2.4.2 รูปแบบของการเชื่อมต่อ	16
2.4.3 ลักษณะสัญญาณของอนุกรม RS-232	20
2.5 พอร์ตอนุกรมมาตรฐาน RS-485	22
2.5.1 รูปแบบของระบบเครือข่าย (Network Topology)	22
2.5.2 SoftwareRS-422 และ RS-485	26
2.5.3 การควบคุมตัวส่งของอุปกรณ์ RS-485 (RS-485 Driver Control)	26
2.5.4 การควบคุมการรับของอุปกรณ์ RS-485 (RS-485 Receiver Control)	27
2.5.5 ระบบแบบตัวแม่และตัวลูก (Master - Slave System)	27
2.5.6 ระบบแบบตัวแม่และตัวลูกที่ต่อกันแบบ 4 สาย (Four-Wire Master - Slave System)	27
2.5.7 ระบบแบบตัวแม่และตัวลูกที่ต่อกันแบบ 2 สาย (Two-Wire Master - Slave System)	27
2.5.8 ระบบ RS-485 ที่มีตัวแม่หลายตัว (Multi Master RS-485 System)	28
2.6 การรายละเอียดของพอร์ตอนุกรมในเครื่องคอมพิวเตอร์	28

สารบัญ

2.7 คุณสมบัติของไมโครคอนโทรลเลอร์ตระกูล MCS 51 อนุกรม AT89XX	30
2.7.1 การจัดขาของไมโครคอนโทรลเลอร์ MCS 51	30
2.7.2 โครงสร้างและการทำงานของพอร์ต	34
2.7.3 การใช้งานเป็นพอร์ตอินพุต	34
2.7.4 การใช้งานเป็นพอร์ตเอาต์พุต	35
2.7.5 การอ่านค่าลอจิกจากพอร์ต	37
2.7.6 รีจิสเตอร์ควบคุมการทำงานของพอร์ตอนุกรมหรือ SCON (Serial port Control Register)	37
2.8 หลักการ SMS	40
2.8.1 หลักการรับส่ง SMS	40
2.8.2 PDU MODE	41
2.8.2.1 การรับข้อมูล SMS	42
2.8.2.2 การส่งข้อมูล SMS	42
2.8.2.3 การถอดรหัสตัวอักษรชนิด 7 บิต	42
2.8.3 AT-COMMAND	43
บทที่ 3 การทำงานของ	45
Visual Basic	45
Microcontroller	46
บทที่ 4 วงจรที่ออกแบบ	47
บทที่ 5 ผลการทดลอง การสื่อสารอนุกรมแบบ RS-485	50
บทที่ 6 วิเคราะห์และสรุปผลการทดลอง	55
ภาคผนวก	56
อ้างอิง	

สารบัญรูปภาพ

รูปที่	หน้า
รูปที่ 2.1 พอร์ตขนานที่ติดอยู่กับเครื่องคอมพิวเตอร์	4
รูปที่ 2.2 การต่อวงจรอิเล็กทรอนิกส์ขนาดเล็กกับพอร์ตขนาน	4
รูปที่ 2.3 ขาสัญญาของพอร์ตขนาน	5
รูปที่ 2.4 แสดงไอคอน System ใน Control Panel	9
รูปที่ 2.5 แสดงการเลือก Device Manager	9
รูปที่ 2.6 แสดงรายละเอียดของพอร์ต	10
รูปที่ 2.7 Printer Port (LPT1) Properties	10
รูปที่ 2.8 RS-232 คอนเน็คเตอร์แบบ DB9	13
รูปที่ 2.9 RS-232 คอนเน็คเตอร์แบบ DB25	14
รูปที่ 2.10 RS-232 คอนเน็คเตอร์แบบ DCE Modify modular jack	14
รูปที่ 2.11 RS-232 คอนเน็คเตอร์แบบ DB9 ตัวผู้ที่อยู่ด้านหลังคอมพิวเตอร์	15
รูปที่ 2.12 ลักษณะการเชื่อมต่อแบบเต็มรูปแบบ	16
รูปที่ 2.13 ลักษณะการเชื่อมต่อแบบ Partial Handshaking	17
รูปที่ 2.14 ลักษณะการเชื่อมต่อแบบ Loop Back Handshaking	18
รูปที่ 2.15 ลักษณะการเชื่อมต่อแบบง่าย	18
รูปที่ 2.16 ลักษณะรูปสัญญาณของอนุกรม RS-232	20
รูปที่ 2.17 ลักษณะสัญญาณของข้อมูลที่มีความสัมพันธ์กับเวลา	21
รูปที่ 2.18 ลักษณะสัญญาณของข้อมูลที่มีความสัมพันธ์กับแรงดันไฟฟ้า	21
รูปที่ 2.19 typical RS-485 Four-wire Multidrop Configuration	23
รูปที่ 2.20 Typical RS-485 Two-wire Multidrop Configuration	23
รูปที่ 2.21 Parallel and AC Termination	25
รูปที่ 2.22 Biasing an RS-485 Network	25
รูปที่ 2.23 แสดงรายละเอียดของพอร์ต	29
รูปที่ 2.24 โครงสร้างพื้นฐานของไมโครคอนโทรลเลอร์ MCS 51	
แบบแฟลชในอนุกรม AT89Cxx	31

สารบัญรูปภาพ

รูปที่	หน้า
รูปที่ 2.25 โครงสร้างพื้นฐานของไมโครคอนโทรลเลอร์ MCS 51	
แบบแฟลชในอนุกรม AT89Sxxx	31
รูปที่ 2.26 รายละเอียดโครงสร้างหลักของไมโครคอนโทรลเลอร์ MCS 51 แบบแฟลชของ Atmel	33
รูปที่ 2.27 การจัดขามาตรฐานของไมโครคอนโทรลเลอร์ MCS 51 แบบแฟลชของ Atmel	34
รูปที่ 2.28 วงจรภายในของพอร์ตทุกพอร์ตในไมโครคอนโทรลเลอร์ MCS 51 แบบแฟลช	36
รูปที่ 2.29 วงจรพูลอัพภายในพอร์ตของไมโครคอนโทรลเลอร์ MCS 51 แบบแฟลช	36
รูปที่ 2.30 หลักการรับ-ส่ง SMS	40
รูปที่ 2.31 ข้อมูลใน SMS ในรูปแบบ PDU MODE	41
รูปที่ 2.32 ตัวอย่างคำสั่งพื้นฐานและ LIST ข้อความใน STO SENT	44
รูปที่ 3.1 รูปแบบการทำงานของโปรแกรม Visual Basic และ User interface	45
รูปที่ 3.2 รูปแบบการทำงานของโปรแกรม Microcontroller	46
รูปที่ 4.1 วงจรควบคุมหลัก	47
รูปที่ 4.2 วงจรควบคุมย่อย	48
รูปที่ 4.3 วงจรสื่อสารอนุกรม	49
รูปที่ 5.1 เมื่อให้เครื่อง 1 เป็นเครื่องส่ง และเครื่อง 2,3 เป็นเครื่องรับ	51
รูปที่ 5.2 เมื่อเปลี่ยนให้เครื่อง 5.2 และ 5.3 เป็นเครื่องส่ง	52
รูปที่ 5.3 เมื่อให้เครื่อง 5, 3 เป็นเครื่องส่งพร้อมกัน	53

สารบัญตาราง

ตารางที่	หน้า
ตารางที่ 2.1 เครื่องมือชนิดต่าง ๆ ที่โปรแกรมเมอร์เลือกใช้	2
ตารางที่ 2.2 รายละเอียดขาสัญญาของพอร์ตขนาน	6
ตารางที่ 2.3 รายละเอียดของ DATA PORT	7
ตารางที่ 2.4 รายละเอียดของ STATUS PORT	7
ตารางที่ 2.5รายละเอียดของ CONTROL PORT	8
ตารางที่ 2.6 Address ต่างของพอร์ตขนาน	11
ตารางที่ 2.7 รายละเอียดขาสัญญาของพอร์ตอนุกรม RS-232	15
ตารางที่ 2.8 หน้าที่การทำงานของการเชื่อมต่อแบบเต็มรูปแบบ	17
ตารางที่ 2.9 หน้าที่การทำงานของการเชื่อมต่อแบบ Partial Handshaking	17
ตารางที่ 2.10 หน้าที่การทำงานของการเชื่อมต่อแบบ Loop Back Handshaking	18
ตารางที่ 2.11 หน้าที่การทำงานของการเชื่อมต่อแบบง่าย	19
ตารางที่ 2.12 ความสัมพันธ์ของ Logic และระดับแรงดัน	20
ตารางที่ 2.13 หน้าที่พิเศษของพอร์ต I ในไมโครคอนโทรลเลอร์ MCS 51 แบบ แฟลชของ Atmel	35
ตารางที่ 2.14 ตารางรีจิสเตอร์ควบคุมการทำงานของพอร์ตอนุกรม	37
ตารางที่ 2.15 ตาราง SM1 (Serial port mode bit 1) : ใช้ร่วมกับบิตSM0ในการกำหนดโหมดการ ทำงานของพอร์ตอนุกรม	38

1 บทนำ

เนื่องจากปัจจุบันนี้มีการใช้ Microcontroller กันอย่างแพร่หลาย และมีการประยุกต์ใช้ในการทำงานหลาย ๆ ด้าน ฉะนั้นทางกลุ่มจึงเล็งเห็นว่าน่าจะนำมาใช้ประโยชน์กับเครื่องใช้ไฟฟ้าภายในบ้าน เพราะว่า เครื่องใช้ไฟฟ้าภายในบ้านเป็นสิ่งที่ต้องใช้งานทุกวัน และแนวคิดที่จะนำ Microcontroller มาใช้ก็จะนำมาใช้ร่วมกัน Visual basic เพื่อให้สั่งงานและประมวลผลบนหน้าจอคอมพิวเตอร์ ซึ่งมีโปรแกรม Microcontroller (MCS 51) เป็นตัวประมวลผลและควบคุมการเปิด ปิด เครื่องใช้ไฟฟ้าภายในบ้าน ดังนั้นเราสามารถที่จะนำไปใช้ในการควบคุมการเปิดปิดไฟฟ้าตามจุดต่าง ๆ ภายในบ้านได้อย่างสะดวกและรวดเร็ว โดยใช้การคอมพิวเตอร์ควบคุมเพียงจุดเดียวเท่านั้น



2 หลักการ

2.1 Visual Studio

2.1.1 การพัฒนาแอปพลิเคชันด้วย Visual Studio

ภาษาโปรแกรมต่าง ๆ นั้นถูกสร้างด้วยวัตถุประสงค์ที่แตกต่างกันซึ่งแต่ละภาษามีจุดเด่นต่างกัน จึงเหมาะกับงานคนละประเภท เช่น ภาษา C เป็นภาษาที่นักเขียนโปรแกรมทั่วโลกนิยมใช้มาก เพราะ ยืดหยุ่นสูงและทำงานได้รวดเร็ว แต่โปรแกรมเมอร์ที่เขียนโปรแกรมเกี่ยวกับฐานข้อมูลกลับไม่นิยมใช้ เพราะผู้ใช้ภาษา Basic ใน Visual Basic ไม่ได้

เพื่อให้เป็นทางเลือกในการสร้างแอปพลิเคชันที่หลากหลายบน Windows ให้ทำงานได้อย่างรวดเร็ว ไมโครซอฟท์ได้แนะนำ Visual Studio ซึ่งเป็นชุดเครื่องมือสำหรับสร้างแอปพลิเคชันบน Window ที่มีเครื่องมือเครื่องมือมากมายมาให้โปรแกรมเมอร์เลือกใช้ได้ตามความเหมาะสมของงาน ดังนี้

เครื่องมือใน Visual Studio	ความสามารถ
Visual Basic	ใช้สำหรับสร้างแอปพลิเคชันเพื่อใช้งานอย่างรวดเร็ว เน้นในส่วนติดต่อผู้ใช้เป็นแบบกราฟฟิก และการสร้างแอปพลิเคชันแบบไคลเอนท์, เซิร์ฟเวอร์, นอกจากนี้ยังสนับสนุนการสร้างแอปพลิเคชันสำหรับอินเทอร์เน็ต
Visual C++	ใช้สำหรับสร้างแอปพลิเคชันที่มีขนาดเล็ก ทำงานได้รวดเร็ว ซึ่ง Visual C++ ใช้อิสระในการพัฒนาแอปพลิเคชันอย่างเต็มที่ ครอบคลุมด้วยเครื่องมือเสริมการทำงานอย่างมากมาย
Visual FoxPro	ใช้สำหรับสร้างแอปพลิเคชันฐานข้อมูลทำงานร่วมกับระบบจัดการฐานข้อมูลแบบต่าง ๆ
MSDN	เป็นแหล่งข้อมูลเอกสารด้านเทคนิค และระบบให้ความช่วยเหลือสำหรับนักพัฒนาที่ใช้งาน Visual Studio

ตารางที่ 2.1 เครื่องมือชนิดต่าง ๆ ที่โปรแกรมเมอร์เลือกใช้

นอกจาก Visual Studio จะมีเครื่องมือสร้างแอปพลิเคชันด้วยภาษาโปรแกรมต่าง ๆ แล้วยังมีเครื่องมือเสริมการทำงานเพื่อสร้าง ทดสอบ และแก้ไขการทำงานของแอปพลิเคชันรูปแบบต่าง ๆ ด้วย ทั้งแอปพลิเคชันที่ทำงานในเครื่องพีซีหรือแอปพลิเคชันที่ทำงานในเครือข่ายภายในองค์กร ไปจนถึงแอปพลิเคชันที่ทำงานผ่านอินเทอร์เน็ตด้วย

แม้ปัจจุบัน ไมโครซอฟท์จะได้แนะนำแนวคิดใหม่ในการพัฒนาแอปพลิเคชันด้วยเทคโนโลยี .NET ซึ่งก็ถือว่าเป็นแนวคิดที่ดี ซึ่งจะเป็นการพัฒนาโดยสามารถเอาเทคนิคโลโลยีจากเครื่องมือชนิดต่าง ๆ มาพัฒนาร่วมกันบนแนวคิดที่ไมโครซอฟท์กำหนดไว้ แต่ปัจจุบันยังถือว่า .NET ต้องการเรียนรู้ให้เกิดความคุ้นเคยซึ่กระยะ ทำให้ผู้ใช้งานส่วนใหญ่ยังคงเริ่มต้นด้วยโปรแกรมแบบ Visual Programming ด้วย Visual Basic 6 อยู่มาก

2.1.2 จุดเด่นของของ Visual Basic

สร้างแอปพลิเคชันได้ง่ายและรวดเร็ว

Visual Basic ได้รับการวางตัวให้เป็นเครื่องมือช่วยสร้างแอปพลิเคชันได้อย่างรวดเร็วและง่ายดาย เพื่อลดเวลาการสร้างแอปพลิเคชันให้สั้นลง

ทั้งนี้เพราะมีการจัดการที่โปรแกรมเมอร์ต้องทำซ้ำๆ ซาก ๆ ออกไป ขจัดสิ่งที่ไม่จำเป็นต้องรู้เกี่ยวกับการควบคุมฮาร์ดแวร์ และการจัดการภายในของบ Windows ออกไปให้เหลือเฉพาะที่ต้องกั้สเกี่ยวกับปัญหาของงานจริง ๆ แล้วเขียนโปรแกรมจัดการปัญหานั้น ๆ ส่วนเรื่องอื่น ๆ ปล่อยให้ Visual Basic จัดการ

ภาษาเขียนโปรแกรมที่ง่ายต่อการเริ่มเรียนรู้

พบว่า Visual Basic นั้นอ่านง่ายคือ อ่านแล้วใกล้เคียงกับภาษาที่ใช้งานปกติ อ่านแล้วสื่อความหมายเข้าใจได้ง่ายกว่าภาษาโปรแกรมอื่น ๆ ทำให้ผู้ที่เริ่มต้นเขียนโปรแกรมทำความเข้าใจกับการเขียนโปรแกรมได้อย่างรวดเร็ว

แม้บรรดานักโปรแกรมทั้งหลาย ๆ คนจะค่อนข้างคิดว่าภาษา Basic ใน Visual Basic นั้นมีโครงสร้างไม่ค่อยดีนักเมื่อเทียบกับภาษา ปลาสคาล หรือ จาวา แต่ต้องไม่ลืมว่าภาษา Basic ใช้เวลาการเรียนรู้ที่สั้นกว่า จึงง่ายต่อการต่อการเรียนรู้และการเข้าใจของคนทั่วไป

2.2 ความรู้พื้นฐานเกี่ยวกับพอร์ตนาน

2.2.1 รู้จักกับพอร์ตนาน

พอร์ตนาน (Parallel Port) หรือรู้จักกันในนามของ Printer Port เพราะใช้งานกับเครื่องพิมพ์เอกสาร (Printer) เป็นส่วนใหญ่ ลักษณะของพอร์ตนานจะเป็น Connector แบบ D-Type 25 Pin (DB25) โดยที่เครื่องคอมพิวเตอร์จะเป็นตัวเมีย



รูปที่ 2.1 พอร์ตนานที่ติดอยู่กับเครื่องคอมพิวเตอร์

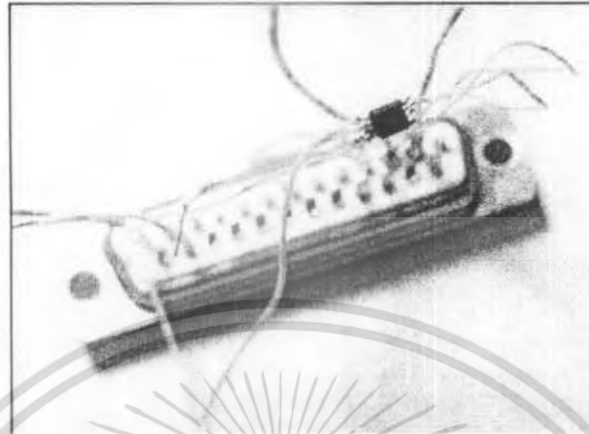
ข้อดีของพอร์ตนาน

- มีความเร็วในการส่งข้อมูลสูง
- มีสัญญาณ Digital TTL ลอจิก "0" มีแรงดันไฟฟ้า 0 Volt และลอจิก "1" มีแรงดันไฟฟ้า 5 Volt ทำให้สามารถต่อกับวงจรรีเลย์หรือทรานซิสเตอร์ขนาดเล็กได้เลย

ข้อเสียของพอร์ตนาน

- ไม่สามารถทำงานในระยะไกลได้
- มีจำนวนสายสัญญาณที่มากเกินไปทำให้ยุ่งยากแก่การใช้งาน

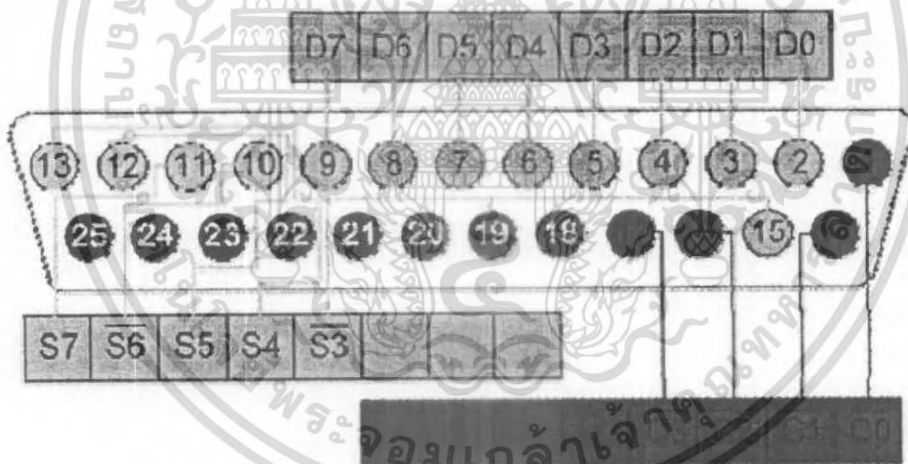
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 การต่อวงจรอิเล็กทรอนิกส์ขนาดเล็กกับพอร์ตขนาน

2.2.2 ลักษณะของพอร์ตขนาน

พอร์ตขนานประกอบด้วยขาสัญญาณ 25 ขาแต่ที่ใช้งานจริงๆ มี 17 ขา มีรายละเอียดดังนี้



รูปที่ 2.3 ขาสัญญาณของพอร์ตขนาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin No.	Signal Name	Bit	Active	Direction (In/Out)
1	nStrobe	C0	LOW	Output
2	Data 0	D0	-	Output
3	Data 1	D1	-	Output
4	Data 2	D2	-	Output
5	Data 3	D3	-	Output
6	Data 4	D4	-	Output
7	Data 5	D5	-	Output
8	Data 6	D6	-	Output
9	Data 7	D7	-	Output
10	nAck	S6	LOW	Output
11	Busy	S7	HIGH	Input
12	PaperEnd	S5	HIGH	Input
13	Select	S4	HIGH	Input
14	nAutoFeed	C1	LOW	Output
15	nError	S3	LOW	Input
16	nInitialize	C2	HIGH	Output
17	nSelectPrinter	C3	LOW	Output
18-25	Ground			

ตารางที่ 2.2 รายละเอียดขาสัญญาณของพอร์ดขนาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3 ลักษณะสัญญาณของพอร์ตขนาน

DADA PORT

เป็นสัญญาณข้อมูลที่จะส่งออกขาพอร์ตขนาน สามารถเขียนได้อย่างเดียว มีทั้งหมด 8-Bit ดังนี้

Name	Read/Write	Bit No.	Signal Name	Pin No.
DADA PORT	Write	7	Data 7	9
		6	Data 6	8
		5	Data 5	7
		4	Data 4	6
		3	Data 3	5
		2	Data 2	4
		1	Data 1	3
		0	Data 0	1

ตารางที่ 2.3 รายละเอียดของ DATA PORT

2.2.4 STATUS PORT

เป็นข้อมูลสถานะของ Hardware ที่ส่งมายังพอร์ตขนานสามารถอ่านได้อย่างเดียว มีอยู่ 8-Bit แต่ใช้งานจริง 5-Bit

Name	Read/Write	Bit No.	Signal Name
STATUS PORT	Read	7	Busy
		6	nAck
		5	PaperEnd
		4	Select
		3	nError
		2	IRQ
		1	Reserved
		0	Reserved

ตารางที่ 2.4 รายละเอียดของ STATUS PORT

โดยมีรายละเอียดของการทำงานแต่ละบิตดังนี้

- Busy เมื่อ Active หมายถึง พรีนเตอร์ไม่พร้อมที่จะรับข้อมูล
- nAck เมื่อ Active หมายถึง พรีนเตอร์พร้อมที่จะทำงาน
- PaperEnd เมื่อ Active หมายถึง พรีนเตอร์ไม่มีกระดาษ
- Select เมื่อ Active หมายถึง เลือกพรีนเตอร์
- nError เมื่อ Active หมายถึง พรีนเตอร์มีข้อผิดพลาด
- IRQ ไม่มีการใช้งาน
- Reserved ไม่มีการใช้งาน

2.2.5 CONTROL PORT

เป็นพอร์ตที่ใช้ควบคุมการทำงานของพรีนเตอร์ สามารถอ่านและเขียนได้ มีทั้งหมด 4- Bit

Name	Read/Write	Bit No.	Signal Name	Pin No.
DADA PORT	Read/Write	3	nESelect	17
		2	nInitialize	16
		1	nAutoFeed	14
		0	nStrobe	1

ตารางที่ 2.5รายละเอียดของ CONTROL PORT

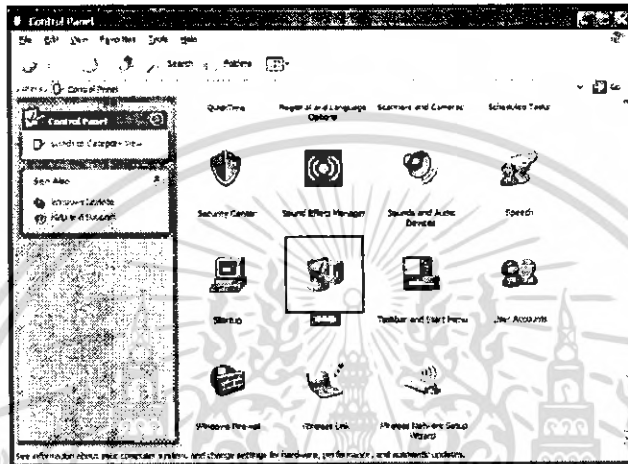
โดยมีรายละเอียดของการทำงานแต่ละบิตดังนี้

- nSelect เมื่อ Active หมายถึง เลือกพรีนเตอร์
- nInitialize เมื่อ Active หมายถึง Reset พรีนเตอร์
- nAutoFeed เมื่อ Active หมายถึง ให้พรีนเตอร์ทำการ Lf: Line Feed
- nStrobe เมื่อ Active หมายถึง ให้พรีนเตอร์รับข้อมูล

2.2.6 การติดตั้งพอร์ตขนาน

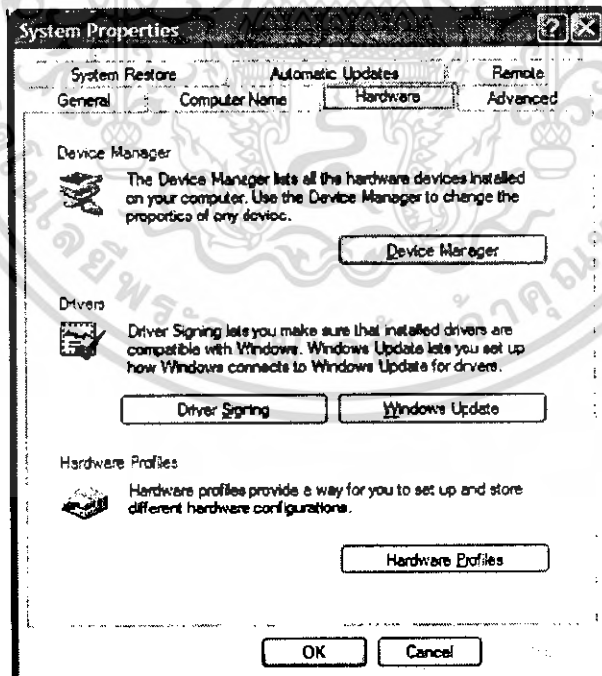
ก่อนที่จะทำการเขียนโปรแกรมติดต่อกับพอร์ตขนานจำเป็นต้องติดตั้งพอร์ตขนาน เพื่อให้ทราบถึง Address ของพอร์ตขนาน โดยในที่นี้จะติดตั้งกับ Windows XP SP2 ซึ่งมีขั้นตอนการติดตั้งดังนี้

1. เข้าไปที่ Control Panel โดยการคลิกที่ “Start -> Setting -> Control Panel”
2. เมื่อเข้าสู่ Control Panel เรียบร้อยแล้วให้ทำการดับเบิลคลิกที่ไอคอน “System”



รูปที่ 2.4 แสดงไอคอน System ใน Control Panel

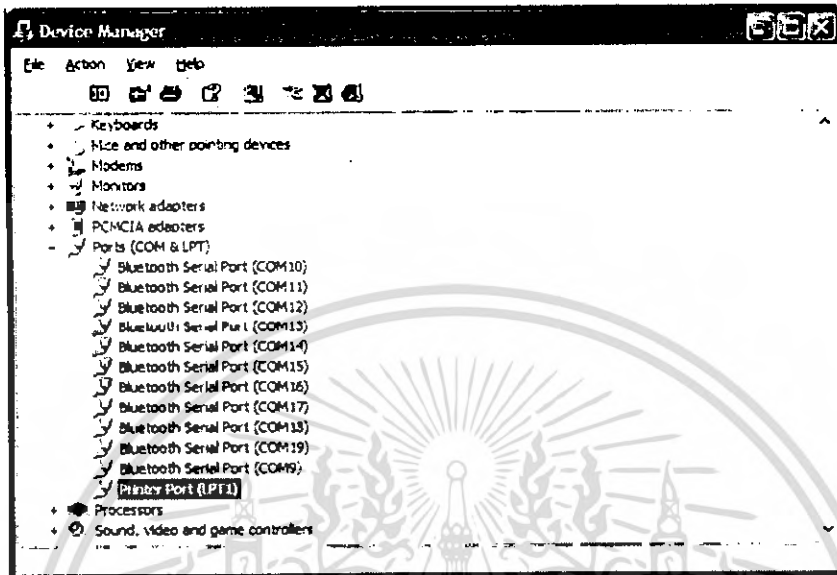
3. เมื่อเข้า “System Properties” แล้ว ทำการเลือกแท็บ “Hardware” คลิก “Device Manager”



รูปที่ 2.5 แสดงการเลือก Device Manager

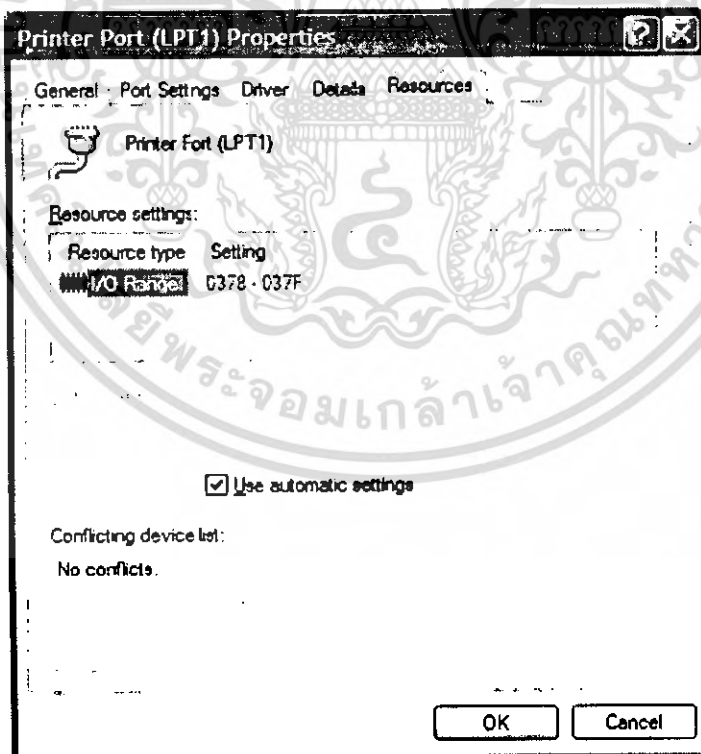
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เมื่อเข้าสู่ “Device Manager” ให้ทำการคลิกเครื่องหมาย + หน้า “Ports (COM & LPT)” เพื่อดูรายละเอียดของพอร์คที่มีอยู่ในเครื่องคอมพิวเตอร์



รูปที่ 2.6 แสดงรายละเอียดของพอร์ค

5. ทำการดับเบิลคลิกที่ “Printer Port (LPT1)” เพื่อเข้าสู่ “Printer Port (LPT1) Properties”



รูปที่ 2.7 Printer Port (LPT1) Properties

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. เมื่อเข้าสู่ “Printer Port (LPT1) Properties” ทำการเลือกแท็บ “Resource” แล้วดูในส่วน
ของ “Resource System” ที่ “Resource Type” จะมี “I/O Range” ซึ่งจะมีค่าของ “Setting”
เป็น “0378-037F” ทำให้เราทราบถึง Address ต่างของพอร์ตขนานดังนี้

Port	LPT1	LPT2
DATA PORT	0378H	037CH
STATUS PORT	0379H	037DH
CONTROL PORT	037AH	037EH

ตารางที่ 2.6 Address ต่างของพอร์ตขนาน



2.3 การเขียนโปรแกรมติดต่อพอร์ตนาน

การเขียนโปรแกรมติดต่อพอร์ตนานการที่จะติดต่อพอร์ตนานได้นั้นจำเป็นที่จะต้องมียิงก์ชันสำเร็จ รูปในการอ่านและเขียนพอร์ตนานซึ่งเป็นไฟล์แบบ .dll ชื่อ "Inpout32.dll" ทำการติดตั้งไฟล์โดย Copy ไฟล์ Inpout32.dll ไปไว้ที่ C:\Windows\System32 ส่วนการเรียกใช้งานนั้น จำเป็นต้องทำการประกาศฟังก์ชันเพื่อเรียกใช้งานไฟล์ ซึ่งมีรายละเอียดดังนี้

- การประกาศฟังก์ชันรับข้อมูล Private Declare Function Input Lib "inpout32.dll" Alias "Inp32" (ByVal PortAddress As Integer) As Integer
- การประกาศฟังก์ชันส่งข้อมูล Private Declare Sub Output Lib "inpout32.dll" Alias "Out32" (ByVal PortAddress As Integer, ByVal Value As Integer)

การเรียกใช้งานฟังก์ชัน

- การรับข้อมูล Data = Input(Port Address)
- การส่งข้อมูล Output(Port Address, Data)

โดยที่

Data หมายถึง ข้อมูลที่รับ ได้จากพอร์ตหรือข้อมูลที่ส่งยังพอร์ต

Port Address หมายถึง Address ของพอร์ตที่จะทำการรับหรือส่งข้อมูล

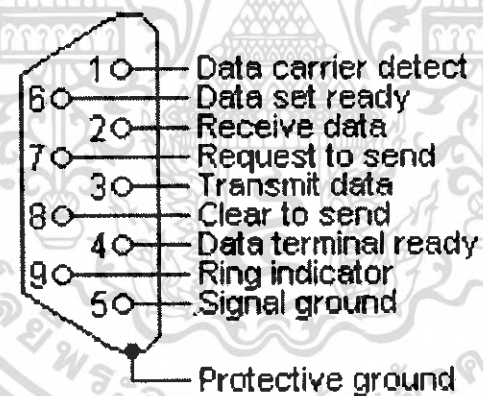
เช่น DATA PORT = 0378 เป็นต้น

2.4 พอร์ตอนุกรมมาตรฐาน RS-232

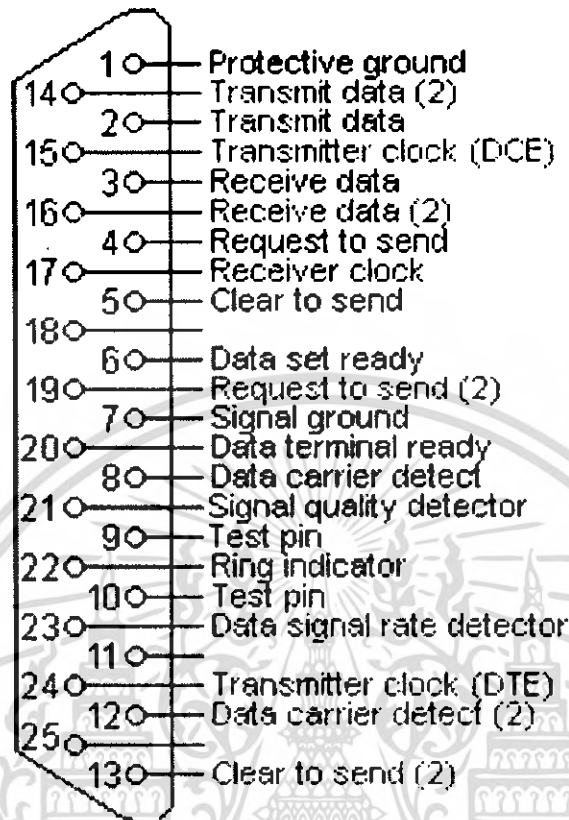
ปัจจุบันการเชื่อมต่อคอมพิวเตอร์ด้วย RS-232 นั้นได้รับความนิยมสูง เพราะมีความง่ายตายไม่ยุ่งยากในการใช้งาน ในบทนี้จะกล่าวถึงรายละเอียดของการเชื่อมต่อคอมพิวเตอร์ด้วยการสื่อสารแบบอนุกรมมาตรฐาน RS-232

2.4.1 RS-232 serial connector pin assignment

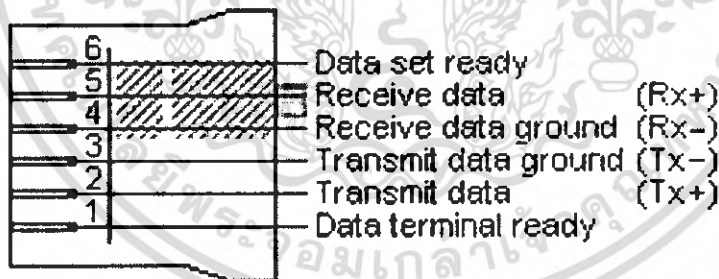
RS-232 คอนเน็คเตอร์ (Connector) แต่เดิมนั้นใช้คอนเน็คเตอร์แบบ 25 ขาหรือเรียกว่าคอนเน็คเตอร์แบบ DB25 ออกแบบมาสำหรับการสื่อสารแบบอนุกรม RS-232 ถึง 2 ช่องการสื่อสาร (Channel) โดยมีช่องการสื่อสารช่องหนึ่งที่มีระบบตรวจสอบความถูกต้อง (Handshake) ติดมาด้วย และมีคอมพิวเตอร์บางรุ่นที่มีระบบตรวจสอบความถูกต้องทั้ง 2 ช่องการสื่อสารเลย ส่วนอีกช่องการสื่อสารเป็นการสื่อสารแบบโมเด็ม (Teletype Modem) ซึ่งสามารถเช็คสถานะของโมเด็มว่าพร้อมหรือไม่พร้อมในการสื่อสาร สำหรับเครื่องคอมพิวเตอร์ส่วนบุคคลทั่วไป (Personal Computer) ในปัจจุบัน RS-232 คอนเน็คเตอร์จะเป็นแบบเล็ก 9 ขาหรือเรียกว่า DB9 ดังแสดงในรูปที่ 1 – 3 โดยขาสี่ขาจะเป็นการสื่อสารแบบโมเด็มส่วนสี่ขาจะเป็นการสื่อสารแบบมีระบบตรวจสอบความถูกต้อง ขา Protective Ground ใน DB25 จะเป็นส่วนเปลือกของคอนเน็คเตอร์แบบ DB9 เพื่อเป็นการป้องกันสัญญาณลบกวน



รูปที่ 2.8 RS-232 คอนเน็คเตอร์แบบ DB9



รูปที่ 2.9 RS-232 คอนเน็คเตอร์แบบ DB25



รูปที่ 2.10 RS-232 คอนเน็คเตอร์แบบ DCE Modify modular jack

นอกจากนั้นยังมีคอนเน็คเตอร์อีกแบบหนึ่งเรียกว่า DCE Modify modular jack ดังรูปที่ 3 การสื่อสารแบบอนุกรมมาตรฐาน RS-232 มีระดับแรงดันที่เป็นบวกและลบ (ไม่เป็น 0) ในการเชื่อมต่อนั้นจะต้องใช้กับอุปกรณ์เฉพาะที่มีระดับสัญญาณเหมือนกันเท่านั้น กำหนดให้อุปกรณ์ต้นทางคือ DTE: data terminal equipment เช่น คอมพิวเตอร์, ปริ้นเตอร์ เป็นต้น และอุปกรณ์ปลายทางคือ DCE: data communication equipment เช่น โมเด็ม เป็นต้น



รูปที่ 2.11 RS-232 คอนเน็คเตอร์แบบ DB9 ตัวผู้ที่อยู่ด้านหลังคอมพิวเตอร์

สำหรับเครื่องคอมพิวเตอร์ส่วนบุคคลทั่วไป (Personal Computer) ในปัจจุบันจะเป็น RS-232 คอนเน็คเตอร์แบบเล็ก DB9 ที่หลังคอมพิวเตอร์จะเป็นตัวผู้ ซึ่งมีรายละเอียดของขาดังตารางที่ 1

DB25	DB9	Signal	Name	Direction (In/Out)
2	3	TD	Transmit Data	Output
3	2	RD	Receive Data	Input
4	7	RTS	Request to Send	Output
5	8	CTS	Clear to Send	Input
6	6	DSR	Data Set Ready	Input
7	5	SG	Signal Ground	Ground
8	1	CD	Carrier Detect	Input
20	4	DTR	Data Terminal Ready	Output
22	9	RI	Ring Indicator	Input

ตารางที่ 2.7 รายละเอียดขาสัญญาณของพอร์ตอนุกรม RS-232

โดยมีรายละเอียดหน้าที่การทำงานดังต่อไปนี้

- TD: Transmit Data ใช้สำหรับส่งข้อมูลออก
- RD: Receive Data ใช้สำหรับรับข้อมูลเข้า
- RTS: Request to Send บอกให้ปลายทางเตรียมรับข้อมูล
- CTS: Clear to Send สำหรับตรวจสอบความพร้อมของอุปกรณ์ปลายทางที่จะรับข้อมูล
- DSR: Data Set Ready ใช้ตรวจสอบการเชื่อมต่อ
- SG: Signal Ground กราวด์
- CD: Carrier Detect จะ Active เมื่อมีสัญญาณ Carrier จากโมเด็ม
- DTR: Data Terminal Ready ใช้บอกอุปกรณ์ปลายทางว่าต้องการติดต่อ
- RI: Ring Indicator จะ Active เมื่อ โมเด็ม ได้รับสัญญาณเรียกเข้าจากโทรศัพท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

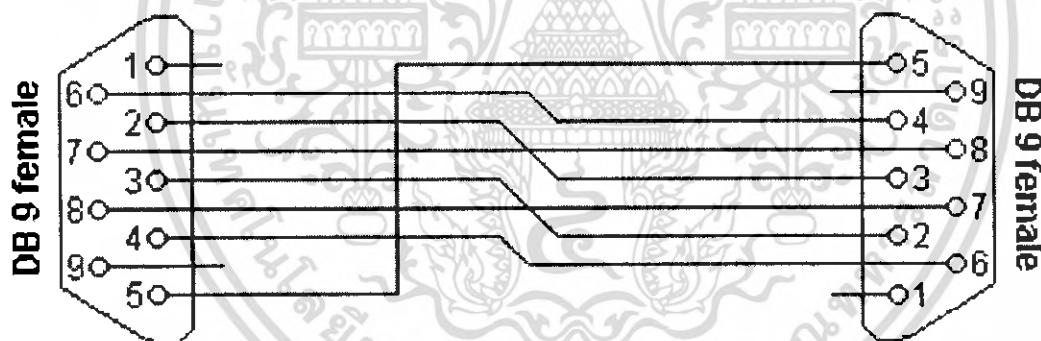
2.4.2 รูปแบบของการเชื่อมต่อ

สำหรับ RS-232 ที่เป็นการสื่อสารแบบโมเด็มนั้น สามารถที่จะประยุกต์การเชื่อมต่อระหว่างอุปกรณ์ 2 ตัวโดยใช้สาย RS-232 ซึ่งสามารถรองรับรูปแบบการเชื่อมต่อมากมาย เช่น สามารถใช้สายสัญญาณเพียง 3 เส้น คือ RD, TD และ SG ก็เพียงพอสำหรับการเชื่อมต่อแล้ว โดยรูปแบบการเชื่อมต่อที่ขึ้นอยู่กับ Software ที่ใช้ ซึ่งบาง Software อาจจะไม่จำเป็นต้องมีการตรวจสอบความผิดพลาดเลย หรือการสื่อสารระหว่างคอมพิวเตอร์ (Windows 95, 98, ME) การเชื่อมต่อแบบ Loop Back Handshaking นั้นเป็นแบบที่เหมาะสมที่สุด

การเชื่อมต่อระหว่างอุปกรณ์ 2 ตัวนั้นมีหลากหลายรูปแบบ เช่น Loop Back Handshaking เหมาะสำหรับการเชื่อมต่อคอมพิวเตอร์ หรือ แบบเต็มรูปแบบเหมาะสำหรับระบบใหญ่ 2 ระบบ เป็นต้น ในการเรียนรู้ในบทนี้เราจะใช้การเชื่อมต่อแบบง่ายในการสื่อสารข้อมูล ซึ่งรูปแบบการเชื่อมต่อที่ไปรับความนิยมมีดังนี้

- Full Handshaking

เป็นการใช้งาน RS-232 แบบเต็มรูปแบบ

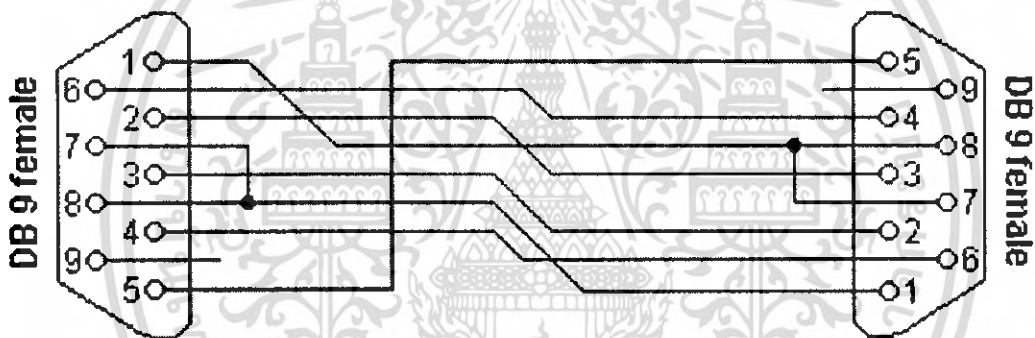


รูปที่ 2.12 ลักษณะการเชื่อมต่อแบบเต็มรูปแบบ

Connector 1	Connector 2	Function
2	3	Rx ⌚ Tx
3	2	Tx ⌚ Rx
4	6	DTR ⌚ DSR
5	5	Signal ground
6	4	DSR ⌚ DTR
7	8	RTS ⌚ CTS
8	7	CTS ⌚ RTS

ตารางที่ 2.8 หน้าที่การทำงานของการทำงานเชื่อมต่อแบบเต็มรูปแบบ

• Partial Handshaking



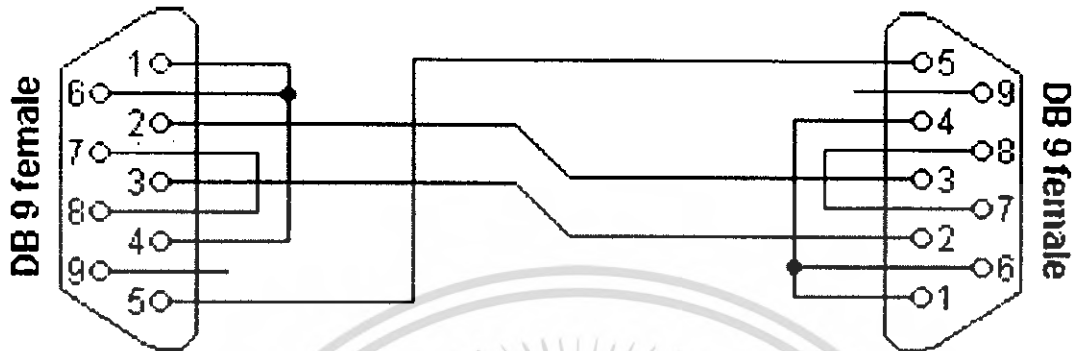
รูปที่ 2.13 ลักษณะการเชื่อมต่อแบบ Partial Handshaking

Connector 1	Connector 2	Function
1	7 + 8	RTS2 ⌚ CTS2 + CD1
2	3	Rx ⌚ Tx
3	2	Tx ⌚ Rx
4	6	DTR ⌚ DSR
5	5	Signal ground
6	4	DSR ⌚ DTR
7 + 8	1	RTS1 ⌚ CTS1 + CD2

ตารางที่ 2.9 หน้าที่การทำงานของการทำงานเชื่อมต่อแบบ Partial Handshaking

72832

• Loop Back Handshaking

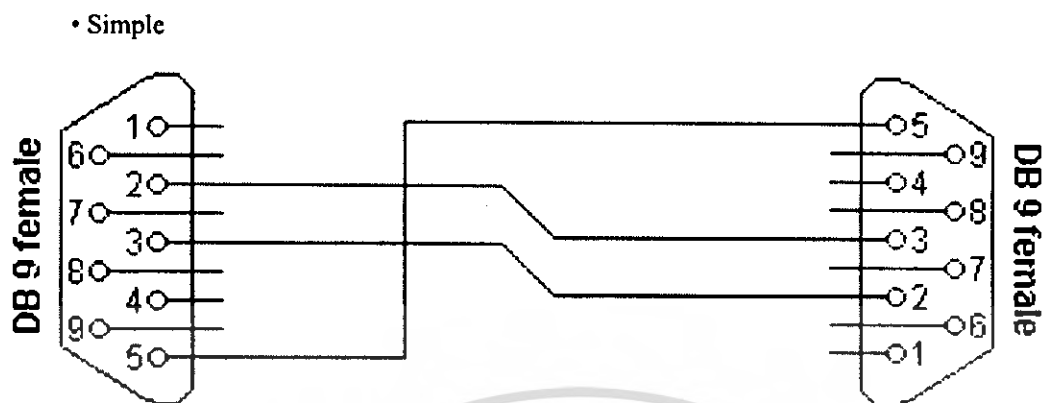


รูปที่ 2.14 ลักษณะการเชื่อมต่อแบบ Loop Back Handshaking

Connector 1	Connector 2	Function
2	3	Rx \odot Tx
3	2	Tx \odot Rx
5	5	Signal ground
1 + 4 + 6	-	DTR \odot CD + DTR
-	1 + 4 + 6	DTR \odot CD + DTR
7 + 8	-	RTS \odot CTS
-	7 + 8	RTS \odot CTS

ตารางที่ 2.10 หน้าทีการทำงานของการเชื่อมต่อแบบ Loop Back Handshaking

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



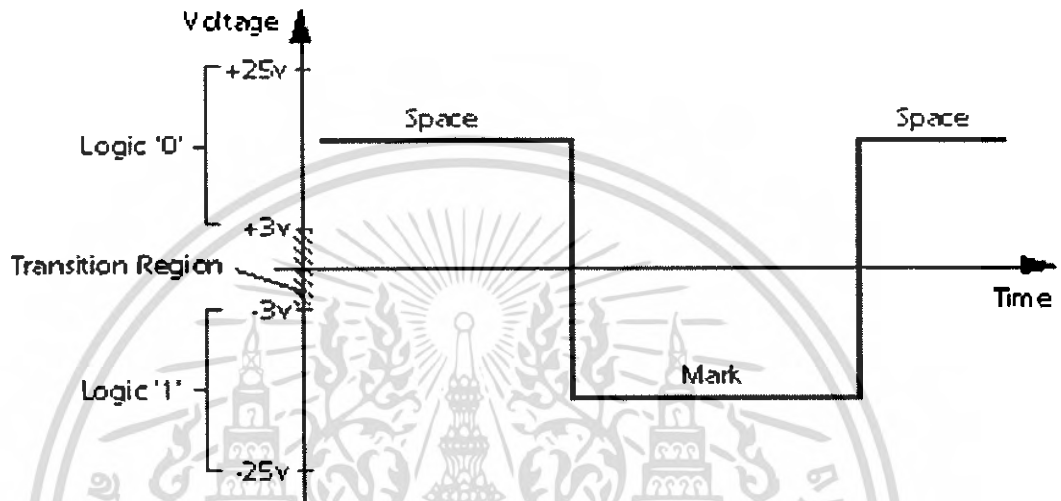
รูปที่ 2.15 ลักษณะการเชื่อมต่อแบบง่าย

Connector 1	Connector 2	Function
2	3	Rx ⌚ Tx
3	2	Tx ⌚ Rx
5	5	Signal ground

ตารางที่ 2.11 หน้าที่การทำงานของการทำงานเชื่อมต่อแบบง่าย

2.4.3 ลักษณะสัญญาณของอนุกรม RS-232

การสื่อสารแบบอนุกรมมาตรฐาน RS-232 มีระดับแรงดันที่เป็นบวกและลบ (ไม่เป็น 0) ดังแสดงในรูปที่ 9 ในการเชื่อมต่อนั้นจะต้องใช้กับอุปกรณ์เฉพาะที่มีระดับสัญญาณเหมือนกันเท่านั้น



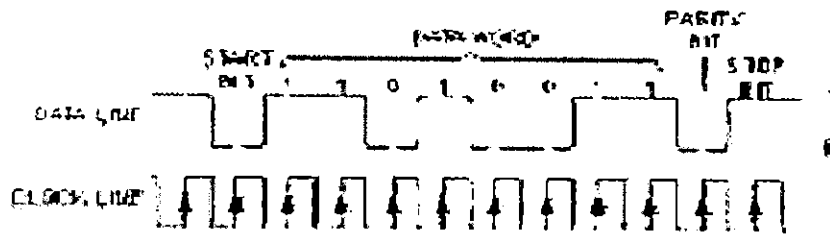
รูปที่ 2.16 ลักษณะรูปสัญญาณของอนุกรม RS-232

Logic	Voltage
"0"	+3 to +25
"1"	-3 to -25
Unknown	+3 to -3

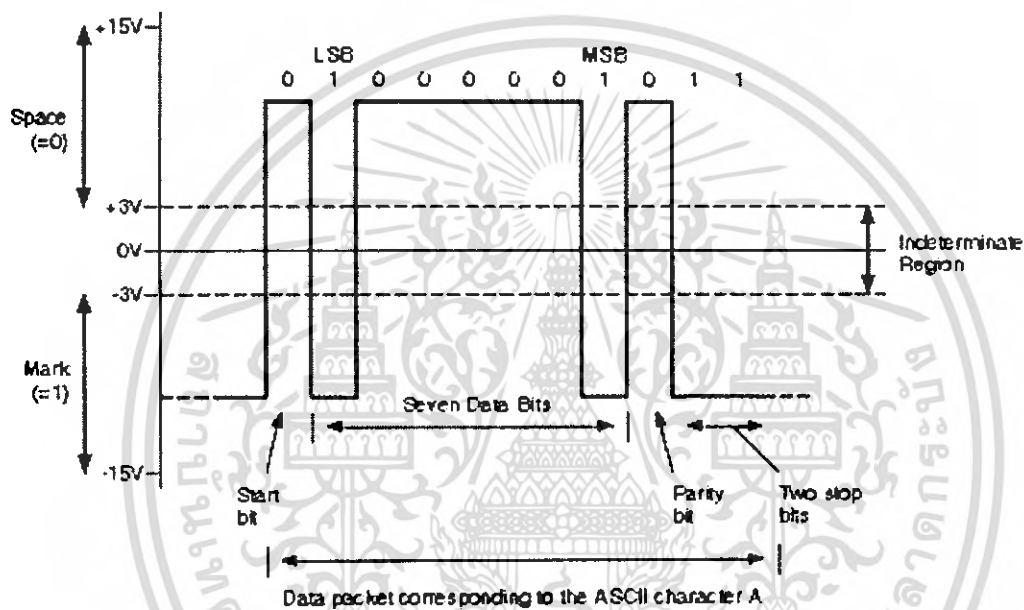
ตารางที่ 2.12 ความสัมพันธ์ของ Logic และระดับแรงดัน

เนื่องจากสัญญาณรับกวนคือสัญญาณที่ไม่พึงประสงค์ในระบบ โดยทั่วไปสัญญาณรบกวนจะมีระดับแรงดันที่น้อย เพราะฉะนั้น RS-232 จึงเพิ่มระดับแรงดันให้มากกว่าระดับแรงดันของสัญญาณรบกวนเพื่อจุดประสงค์ในการกำจัดสัญญาณรบกวนนั่นเอง

สำหรับสัญญาณของข้อมูลนั้นจะเป็นลักษณะข้อมูลแบบอนุกรมทีละบิต โดยแต่ละบิตจะแสดงตามระยะเวลาที่กำหนด ซึ่งเป็นตัวกำหนดอัตราความเร็วในการส่งข้อมูล (Baud Rate) คือ จำนวนบิตต่อวินาที เช่น 110, 150, 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200, 256000 ตามมาตรฐาน RS-232 ดังแสดงในรูปที่ 10



รูปที่ 2.17 ลักษณะสัญญาณของข้อมูลที่มีความสัมพันธ์กับเวลา



รูปที่ 2.18 ลักษณะสัญญาณของข้อมูลที่มีความสัมพันธ์กับแรงดันไฟฟ้า

รายละเอียดของสัญญาณแต่ละบิตมีดังต่อไปนี้

- Start Bit เป็นบิตเริ่มต้นของชุดบิตข้อมูล
- Data Bit เป็นชุดบิตข้อมูล มีขนาด 7 หรือ 8 บิต
- Parity Bit เป็นบิตที่ใช้ตรวจสอบความถูกต้องของข้อมูล
 - Even Parity เป็น "1" เมื่อจำนวนบิต "1" เป็นเลขคู่
 - Odd Parity เป็น "1" เมื่อจำนวนบิต "1" เป็นเลขคี่
 - None จะไม่มีการตรวจสอบความถูกต้อง
- Stop Bit เป็นบิตปิดท้าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 พอร์ตอนุกรมมาตรฐาน RS-485

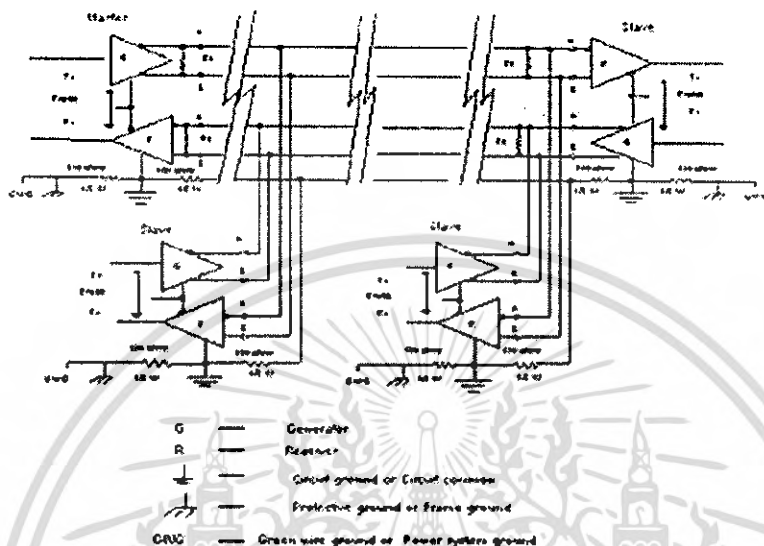
ระบบเครือข่าย RS-485

- ติดต่อกันแบบ Half-Duplex หรือ Full-Duplex
- ระบบสมดุลย์ (Balanced system) ใช้สายสัญญาณ 1 คู่(Half-Duplex) หรือ 2 คู่ (Full-Duplex) และ สายกราวด์ 1 เส้น
- ติดต่อกันระยะทางไม่เกิน 1,200 เมตร หรือ 4,000 ฟุต
- ความยาวของระบบและจำนวนโหนดสามารถขยายได้โดยตัวทวนสัญญาณหรือRepeater

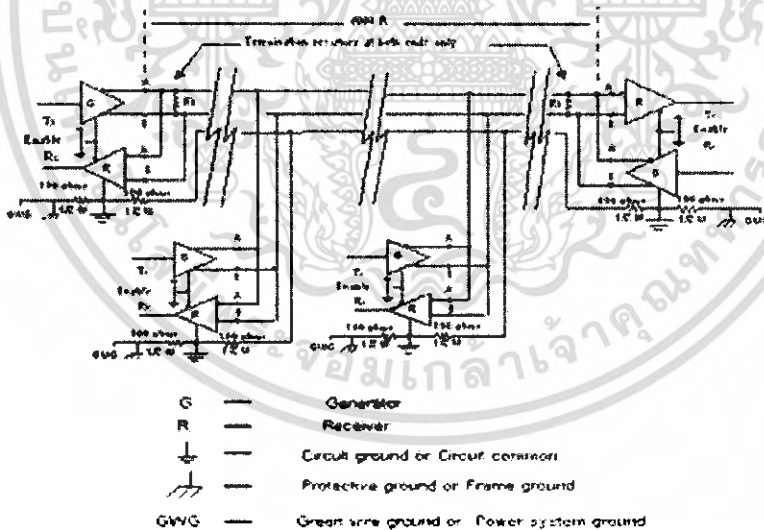
2.5.1 รูปแบบของระบบเครือข่าย (Network Topology)

ในระบบเครือข่าย RS-232 และ RS-485 ไม่มีข้อกำหนดในหัวข้อรูปแบบการเชื่อมต่อ การออกแบบการเชื่อมต่อมักขึ้นอยู่กับความต้องการทางกายภาพของระบบ ระบบ 2 สาย และ 4 สาย (Two-wire or Four-wire System)

ระบบ RS-422 ใช้ 1 คู่สายสำหรับแต่ละสัญญาณ เช่น 1 คู่สายสำหรับสายส่ง , 1 คู่สายสำหรับสายรับ และในบางกรณีอาจมี 1 คู่สายทำหน้าที่ควบคุมการทำ tri-state (handshake line) ในการทำ tri-stated ของ RS-485 มีความสามารถทำให้การรับ-ส่งข้อมูลผ่านสายสัญญาณคู่เดียวกันได้หรือมีลักษณะการสื่อสารแบบHalf-duplexแบบการเชื่อมต่อในลักษณะ 2 สายจะช่วยลดค่าใช้จ่ายในการเชื่อมต่อระบบได้เนื่องจากใช้สายน้อยกว่า(สายสัญญาณ 2 สายและสายกราวด์ 1 สาย) ในการต่อแบบ 2 สายของอุปกรณ์ RS-485 จะมีข้อกำหนดการเชื่อมต่อทั้งแบบภายในและภายนอก ในการเชื่อมต่อภายในจะมีขั้วต่อ A และ B (ในบางครั้งแทนด้วย - และ +) ให้ทำการเชื่อมต่อเข้ากับสายส่งการเชื่อมต่อแบบ 4 สายจะมีขั้ว A และ B ทั้งที่ตัวส่งและตัวรับ ผู้ใช้สามารถส่งข้อมูลให้กับตัวรับโดยผ่านสาย 2 เส้นและส่งข้อมูลกลับผ่านสายอีก 2 สาย สายกราวด์ยังเป็นสายที่มีความสำคัญต่อระบบจำเป็นต้องใช้เป็นจุดอ้างอิงในการรักษาระดับแรงดันคอมมอนหรือVcm (แรงดันที่วัดระหว่างสายสัญญาณแต่ละเส้นเทียบกับสายกราวด์) ระบบสามารถทนแรงดันคอมมอนได้ที่ค่าหนึ่ง ดังนั้นจำเป็นต้องรักษาแรงดันดังกล่าวให้อยู่ในช่วงที่อุปกรณ์รับได้ ถ้าไม่มีสายกราวด์ในระบบจะทำให้เกิดสัญญาณรบกวน (noise) และระบบจะไม่มี ความมั่นคง รูปที่2.26 แสดงการต่อแบบ 4 สาย รูปที่2.27 แสดงการต่อแบบ 2 สาย



รูปที่ 2.26 Typical RS-485 Four-wire Multidrop Configuration



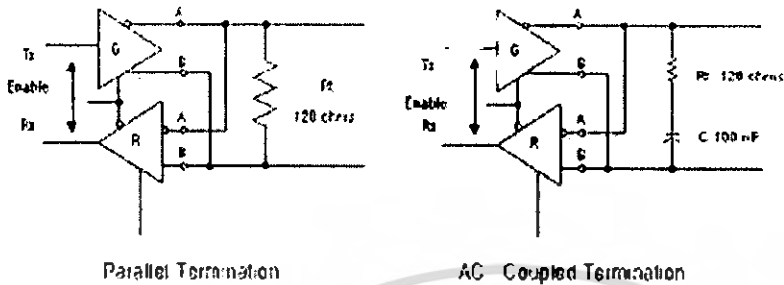
รูปที่ 2.27 Typical RS-485 Two-wire Multidrop Configuration

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การต่อจ็วปลาย(Termination)

ระบบควรมีการต่อจ็วปลายหรือการต่ออิมพีแดนซ์(Rt)เข้าระหว่างเทอร์มินอล เพื่อทำให้เกิดอิมพีแดนซ์สมดุลกันระหว่างอิมพีแดนซ์ของโหลดกับอิมพีแดนซ์ของสายส่ง ถ้าอิมพีแดนซ์ไม่สมดุลจะทำให้โหลดไม่ได้รับสัญญาณที่สมบูรณ์เนื่องจากสัญญาณบางส่วนเกิดการสะท้อนกลับภายในสายส่ง ถ้าอิมพีแดนซ์ของตัวกำเนิด (source) , อิมพีแดนซ์ของสายส่ง (transmission line) และอิมพีแดนซ์ของโหลดมีค่าเท่ากันจะไม่เกิดการสะท้อนกลับในระบบ แต่การต่อจ็วปลายนั้นก็ถือว่าเป็นการเพิ่มโหลดให้กับตัวส่ง ทำให้การติดตั้งซับซ้อนมากขึ้น , จำเป็นต้องเปลี่ยนแปลงการไบอัส และการปรับปรุงหรือแก้ไขระบบจะทำได้ยากขึ้น การตัดสินใจว่าควรทำการต่อจ็วปลายหรือไม่ขึ้นอยู่กับความยาวของสายเคเบิลและอัตราข้อมูลในระบบ หรืออีกทางหนึ่งอาจดูจาก Propagation delay ถ้ามีค่าน้อยกว่าความยาว 1 บิท ไม่จำเป็นต้องมีการต่อจ็วปลายเนื่องจากสัญญาณจะสะท้อนกลับไปกลับมาจนมีขนาดน้อยลงและหายไปในที่สุด(damp out)จึงไม่มีผลต่อการรับข้อมูล การรับของ "UART" จะเอาค่าที่อยู่ตรงกลางของบิทซึ่งถือว่าเป็นค่าที่มีความถูกต้องที่สุด การหาค่า Propagation delay คำนวณได้จากผลคูณระหว่างความยาวสายเคเบิลกับ Propagation velocity ของสายเคเบิล โดยจะถูกกำหนดจากผู้ผลิตซึ่งมีค่าอยู่ในช่วง 66%-75% ของความเร็วแสง จะเห็นได้ว่า Propagation delay มีค่าน้อยกว่าครึ่งหนึ่งของความยาว 1 บิท ดังนั้นการสะท้อนกลับไม่มีผลต่อข้อมูล จึงไม่จำเป็นต้องมีการต่อจ็วปลาย การต่อจ็วปลายมีหลายวิธีแต่ B&B แนะนำวิธีการต่อขนาน หรือ Parallel termination ความต้านทานจะต่อขนานเข้ากับจ็ว A และ B ของตัวรับ ถ้าเป็นระบบที่ไม่มีการต่อตัวทวนสัญญาณ หรือ Repeater ควรต่อจ็วปลายเพียง 2 แห่งคือที่ปลายทั้งสองด้านของสายส่ง สำหรับการเลือกค่าความต้านทานจ็ว หรือ Rt นั้นขึ้นอยู่กับค่าอิมพีแดนซ์ภายในสายส่งซึ่งขึ้นอยู่กับการผลิตของผู้ผลิตไม่ได้ขึ้นอยู่กับความยาวของสาย ซึ่งโดยมากแล้วผู้ผลิตมักกำหนดการผลิตสายเคเบิลให้มีค่าความต้านทานอยู่ที่ 120 โอห์ม อย่างไรก็ตามสามารถเลือกค่าความต้านทานได้ตั้งแต่ 90 โอห์มขึ้นไปข้อเสียของการต่อจ็วปลายแบบขนานคือเปรียบเสมือนการเพิ่มโหลด DC ให้กับระบบและอาจทำให้ตัวเปลี่ยน RS-232 เป็น RS-485 (RS-232 to RS-485 Converter) เกิดการ overload นอกจากนี้ยังมีการต่อจ็วปลายอีกวิธีหนึ่งเรียกว่า AC Coupled Termination ทำได้โดยการต่อคาปาซิเตอร์ขนาดเล็กๆอนุกรมเข้ากับความต้านทานจ็ว (Rt) เพื่อกำจัดผลกระทบจาก DC loading การเลือกค่าคาปาซิเตอร์ขึ้นอยู่กับคุณสมบัติของระบบ ผู้ออกแบบควรศึกษาจาก National Semiconductor Application Note รูปที่2.28 แสดงการต่อจ็วปลายแบบ Parallel

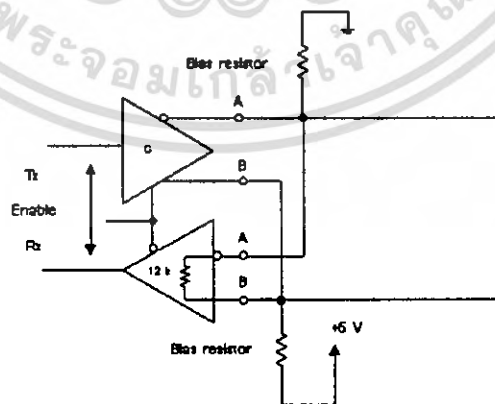
termination และ AC Coupled Termination ในระบบ RS-485 แบบ 2 สาย สำหรับในระบบแบบ 4 สาย ความต้านทานขั้ว หรือ R_t จะถูกต่อคร่อมที่ตัวรับ



รูปที่ 2.28 Parallel and AC Termination

การไบอัสในระบบเครือข่าย RS-485 (Biasing an RS-485 Network)

เมื่อทุกโหนดอยู่ในโหมดของการรอรับข้อมูล หรืออยู่ในสถานะ idle (idle state) จะไม่มีตัวส่งใดทำงาน (active) หรือเรียกว่าอยู่ในสถานะ tri-state สายส่งจะอยู่ในลักษณะเรียกว่า Unknown ถ้าระดับแรงดันของอินพุตที่ ตัวรับ (A และ B) มีค่าอยู่ในช่วง ± 200 mV. ระดับลอจิกที่เอาท์พุทของตัวรับจะคงค่าของบิตสุดท้ายที่รับ เพื่อรักษาแรงดันidleให้เหมาะสมจำเป็นต้องมีการต่อความต้านทานที่เรียกว่า ความต้านทานไบอัส เข้าระหว่างขั้ว B กับ 5 V. หรือเรียกว่า Pull up resistor และใส่ความต้านระหว่างขั้ว A กับกราวด์หรือเรียกว่า Pull down resistor รูปที่ 2.29 แสดงการต่อความต้านทานไบอัสเข้ากับตัวรับในระบบแบบ 2 สาย สำหรับระบบแบบ 4 สายจะต่อความต้านทานเข้ากับตัวรับ ค่าความต้านทานไบอัสขึ้นอยู่กับ การต่อขั้วปลายและจำนวนโหนดในระบบ การสร้างกระแสไบอัสให้มีค่าเพียงพอที่จะรักษาแรงดันระหว่างสายส่ง A และ B ให้มีค่ามากกว่า 200 mV. ดูได้จากตัวอย่างที่ 2 และ 3 ซึ่งแสดงการคำนวณค่าความต้านทานไบอัส



รูปที่ 2.29 Biasing an RS-485 Network

2.5.2 Software RS-422 และ RS-485

เป็นข้อกำหนดทางฮาร์ดแวร์ แต่ไม่รวมถึงข้อกำหนดทางซอฟต์แวร์โปรโตคอล ซึ่งจะขึ้นอยู่กับผู้ออกแบบระบบที่จะตัดสินใจเลือกใช้โปรโตคอลที่เหมาะสมกับระบบของผู้ออกแบบเอง บทนี้จะไม่กล่าวถึงพื้นฐานของโปรโตคอล แต่จะกล่าวถึงหลักที่ต้องพิจารณาในการเลือกใช้โปรโตคอลไม่ว่าผู้ออกแบบจะซื้อหรือเขียนเองก็ตามระบบ RS-422 (RS-422 System)ซอฟต์แวร์ของระบบ RS-422 แตกต่างจากระบบสื่อสาร RS-232 Point - to - Point เล็กน้อย RS-422 มักถูกใช้ขยายระยะทางระหว่างโหนดที่ระบบ RS-232 ไม่สามารถส่งได้ นอกจากนี้ RS-422 ยังสื่อสารในลักษณะโหนดแม่และโหนดลูกในการเชื่อมต่อแบบ 4 สายได้ (Four - Wire Master - Slave Network) ในการเลือกซื้อหรือเขียนซอฟต์แวร์ ผู้ออกแบบจะต้องเข้าใจถึงสัญญาณที่จำเป็นต่อการทำงานของฮาร์ดแวร์ในระบบ บ่อยครั้งที่ระบบ RS-422 ที่ต่ออยู่ในระบบ RS-232 ไม่มีการต่อฮาร์ดแวร์บางชนิดเช่น handshake line อาจมีเหตุผลมาจากต้องมีการเดินสายเพิ่มซึ่งหมายถึงการลงทุนที่สูงขึ้นด้วย

2.5.3 การควบคุมตัวส่งของอุปกรณ์ RS-485 (RS-485 Driver Control)

ข้อแตกต่างหลักระหว่าง RS-422 และ RS-485 คือการทำ High Impedance หรือ tri-state คือการยินยอมให้ตัวส่งอื่นๆสามารถส่งข้อมูลผ่านสายคู่เดียวกันได้ มี 2 วิธีในการทำ tri-state วิธีแรกคือการใช้สายควบคุม (Control line) หรือเรียกว่า RTS handshake line วิธีนี้ต้องมีซอฟต์แวร์หลักทำหน้าที่ส่งสัญญาณให้ตัวส่งเริ่มส่งและทำการตัดตัวส่งออกเมื่อส่งเสร็จ ข้อสำคัญในการทำงานนี้ก็คือตัวส่งต้องตัดตัวเองออกอย่างรวดเร็วที่สุด เพื่อหลีกเลี่ยงการส่งของตัวส่งพร้อมกัน 2 ตัว หรือเกิดการแย่งสายกันส่งข้อมูล (Contention) ซึ่งในระบบที่ใช้วิธีการควบคุมการส่งแบบ handshake line นี้เป็นการยากที่จะหลีกเลี่ยงเหตุการณ์ดังกล่าว

วิธีที่สองคือการใช้วงจรควบคุมการส่งข้อมูลแบบอัตโนมัติ เมื่อวงจรได้รับข้อมูลที่จะส่ง วงจรจะทำการส่งสัญญาณ Enable ให้ตัวส่งต่อตัวเองเข้ากับสายส่งและเมื่อส่งเสร็จภายในระยะเวลา 1 ความยาวตัวอักษร ตัวส่งจะตัดตัวเองออกทันที ซึ่งวิธีนี้เป็นที่นิยมใช้มากกว่าเนื่องจากลดการทำงานของซอฟต์แวร์หรือข้อบกพร่องที่เกิดจากการเขียนซอฟต์แวร์ของโปรแกรมเมอร์

2.5.4 การควบคุมการรับของอุปกรณ์ RS-485 (RS-485 Receiver Control)

ตัวรับในระบบ RS-485 มีสัญญาณควบคุมหรือ Enable signal ข้อดีของระบบ RS-485 ที่มีการเชื่อมต่อระหว่างตัวส่งและตัวรับเป็นแบบ 2 สาย ในลักษณะวงจรปิด หรือ loopback ก็สามารถตัดตัวรับออกทันทีในกรณีที่เกิดการสะท้อนกลับของข้อมูล (echo) และสามารถตัดตัวเองออกได้ในกรณีที่ตรวจพบความผิดพลาดของข้อมูลที่วนกลับมา ซึ่งอาจแสดงให้เห็นว่าเกิดการแย่งสายส่งกันหรือไม่ และแม้ว่าสัญญาณที่กลับมานั้นจะไม่สามารถบอกรายละเอียดเกี่ยวกับความถูกต้องของข้อมูลทั้งหมดได้แต่ก็เป็น การตรวจสอบความถูกต้องในเบื้องต้นได้ในระดับหนึ่ง

2.5.5 ระบบแบบตัวแม่และตัวลูก (Master - Slave System)

ระบบแบบตัวแม่และตัวลูกจะมีโหนดหลักอยู่โหนดหนึ่งซึ่งเป็นโหนดที่ทำหน้าที่ส่งคำสั่งไปยังตัวลูกอื่นๆในระบบและประมวลสัญญาณที่ได้รับกลับมา โหนดลูกจะไม่สามารถส่งข้อมูลได้ถ้าปราศจากคำสั่งที่ตัวแม่ส่งมาเพื่อขอข้อมูลและโหนดลูกไม่สามารถสื่อสารกันเองได้ โหนดลูกแต่ละโหนดจะมีตำแหน่งเฉพาะของตัวเองไม่ซ้ำกัน ระบบในลักษณะนี้สามารถต่อแบบ 2 สาย หรือ 4 สายก็ได้ ซึ่งโดยส่วนมากแล้วในระบบแบบ 4 สายจะใช้อุปกรณ์ RS-422 เป็นโหนดหลักเนื่องจากตัวส่งถูกต่อเข้ากับสายส่งเสมอและใช้อุปกรณ์ RS-485 เป็นโหนดลูกเพื่อลดความซับซ้อนของระบบ

2.5.6 ระบบแบบตัวแม่และตัวลูกที่ต่อกันแบบ 4 สาย (Four-Wire Master - Slave System)

ในการเชื่อมต่อแบบ 4 สายที่มีการทำงานแบบโหนดแม่และโหนดลูกนั้นสามารถลดความซับซ้อนของซอฟต์แวร์ที่โหนดแม่ได้เนื่องจากตัวส่งและตัวรับของตัวแม่จะต่อเข้ากับสายส่งตลอดเวลา ตัวแม่จะส่งคำสั่งไปยังตัวลูกโดยระบุเป็นตำแหน่งของตัวลูกที่ต้องการจะติดต่อ และเนื่องจากการทำงานของโหนดแม่ในลักษณะดังกล่าวจึงไม่สามารถส่งคำสั่งขอข้อมูล (request) ไปยังโหนดลูกพร้อมกันหลายๆโหนดได้ และเนื่องจากโหนดลูกใช้สายคู่เดียวกันในการส่งสัญญาณข้อมูล ดังนั้นจึงไม่เกิดปัญหาเกี่ยวกับการสะท้อนกลับ (echo) หรือการชนกันของข้อมูล (Collision) และทำให้ไม่จำเป็นต้องคำนึงถึง turn around delay

2.5.7 ระบบแบบตัวแม่และตัวลูกที่ต่อกันแบบ 2 สาย (Two-Wire Master - Slave System)

การต่อแบบ 2 สายจะเพิ่มความซับซ้อนให้กับระบบขึ้นเล็กน้อย ตัวส่ง RS-485 จะตัดตัวเองออกจากสายส่งหรือเมื่อไม่ส่งข้อมูลและยอมให้โหนดอื่นๆทำการส่งข้อมูลผ่านสายส่งคู่เดียวกัน ระยะเวลา หรือ Time delay ระหว่างการสิ้นสุดการส่งและการทำ tri-state นั้นเป็นสิ่งสำคัญมากใน

ระบบแบบนี้ ถ้าตัวลูกตอบก่อนที่ตัวแม่จะทำ tri-state หรือตัดตัวเองออกจากสายส่ง อาจทำให้เกิดการชนกันของข้อมูลและข้อมูลเกิดการสูญหายได้ ผู้ออกแบบระบบต้องมั่นใจว่าตัวแม่จะทำการตัดตัวส่งออกภายในระยะเวลา tum around delay

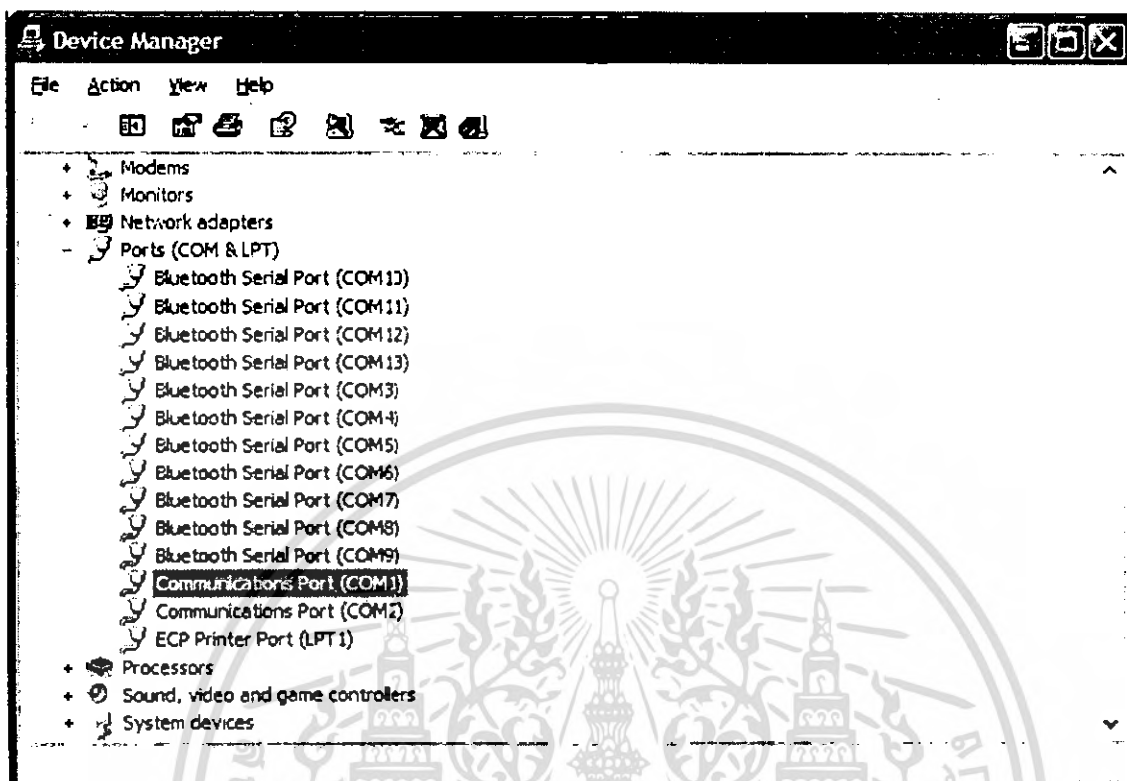
2.5.8 ระบบ RS-485 ที่มีตัวแม่หลายตัว (Multi Master RS-485 System)

ในระบบ RS-485 ที่มีตัวแม่หลายตัวอาจทำให้เกิดการชนกันของข้อมูล ในระบบชนิดนี้ต้องมีการออกแบบวิธีการตรวจจับความผิดพลาดที่มีลักษณะพิเศษสามารถตรวจสอบความผิดพลาดที่ดีมากๆ รวมทั้งการตรวจจับการแย่งสายส่ง หรือเกิดการที่ตัวส่ง 2 ตัวต่อเข้ากับสายส่งพร้อมกัน นอกจากนี้ควรสามารถตรวจจับความผิดพลาดในข้อมูลที่ส่งได้และควรมีวิธีการในการส่งข้อมูลที่ถูกต้องแทนอีกครั้ง

2.6 การดูรายละเอียดของพอร์ตอนุกรมในเครื่องคอมพิวเตอร์

ก่อนที่จะทำการเขียนโปรแกรมติดต่อกับพอร์ตอนุกรมจำเป็นต้องรู้รายละเอียดของพอร์ตอนุกรม เพื่อให้ทราบถึง Address ของพอร์ตอนุกรม ซึ่งมีขั้นตอนดังนี้

1. เข้าไปที่ Control Panel โดยการคลิกที่ “Start -> Setting -> Control Panel”
2. เมื่อเข้าสู่ Control Panel เรียบร้อยแล้วให้ทำการดับเบิลคลิกที่ไอคอน “System”
3. เมื่อเข้า “System Properties” แล้ว ทำการเลือกแท็บ “Hardware” คลิก “Device Manager”
4. เมื่อเข้าสู่ “Device Manager” ให้ทำการคลิกเครื่องหมาย + หน้า “Port(COM & LPT)” เพื่อดูรายละเอียดของพอร์ตที่มีอยู่ในเครื่องคอมพิวเตอร์



รูปที่ 2.19 แสดงรายละเอียดของพอร์ต

จากรูปที่ 12 จะเห็นว่า มีพอร์ตอนุกรม Communications Port (COMx) อยู่ 2 พอร์ตคือ COM1 และ COM2 โดยจะนำไปอ้างอิงเพื่อเขียน โปรแกรมควบคุมต่อไป

2.7 คุณสมบัติของไมโครคอนโทรลเลอร์ตระกูล MCS 51 อนุกรม AT89XX

- เป็นไมโครคอนโทรลเลอร์ที่ใช้ซีพียูขนาด 8 บิต
- ภายในมีหน่วยความจำโปรแกรมเป็นแบบแฟลชมาสามารถลบและเขียนใหม่ได้พันครั้ง
- หน่วยความจำข้อมูลพื้นฐานเป็นแบบแรม ในบางเบอร์มีหน่วยความจำอีพ롬เพิ่มเติม
- มีวงจรสื่อสารอนุกรมแบบฟูลดูเพล็กซ์
- ไทเมอร์ เคนต์เนอร์ขนาด 16 บิต อย่างน้อย 2 ตัว
- สามารถรับแหล่งกำเนิดอินเตอร์รัปต์ได้อย่างน้อย 6 ประเภท
- สามารถขยายหน่วยความจำภายนอกเพิ่มเติมได้สูงสุด 64 กิโลไบต์
- มีวงจรกำเนิดสัญญาณพิกายู่ภายในชิป
- มีวงจรสื่อสารอนุกรมแบบ SPI สำหรับในอนุกรม AT89xx
- มีวอตช์ดอกไทเมอร์ในตัว สำหรับในอนุกรม AT89xx

2.7.1 การจัดขาของไมโครคอนโทรลเลอร์ MCS 51

ไมโครคอนโทรลเลอร์ MCS 51 ทุกเบอร์จะมีสถาปัตยกรรมและขาใช้งานพื้นฐานเหมือนกัน

ขา Vcc ใช้สำหรับต่อไฟเลี้ยง +5V

ขา GND เป็นขากราวด์ สำหรับต่อกับกราวด์ของระบบ

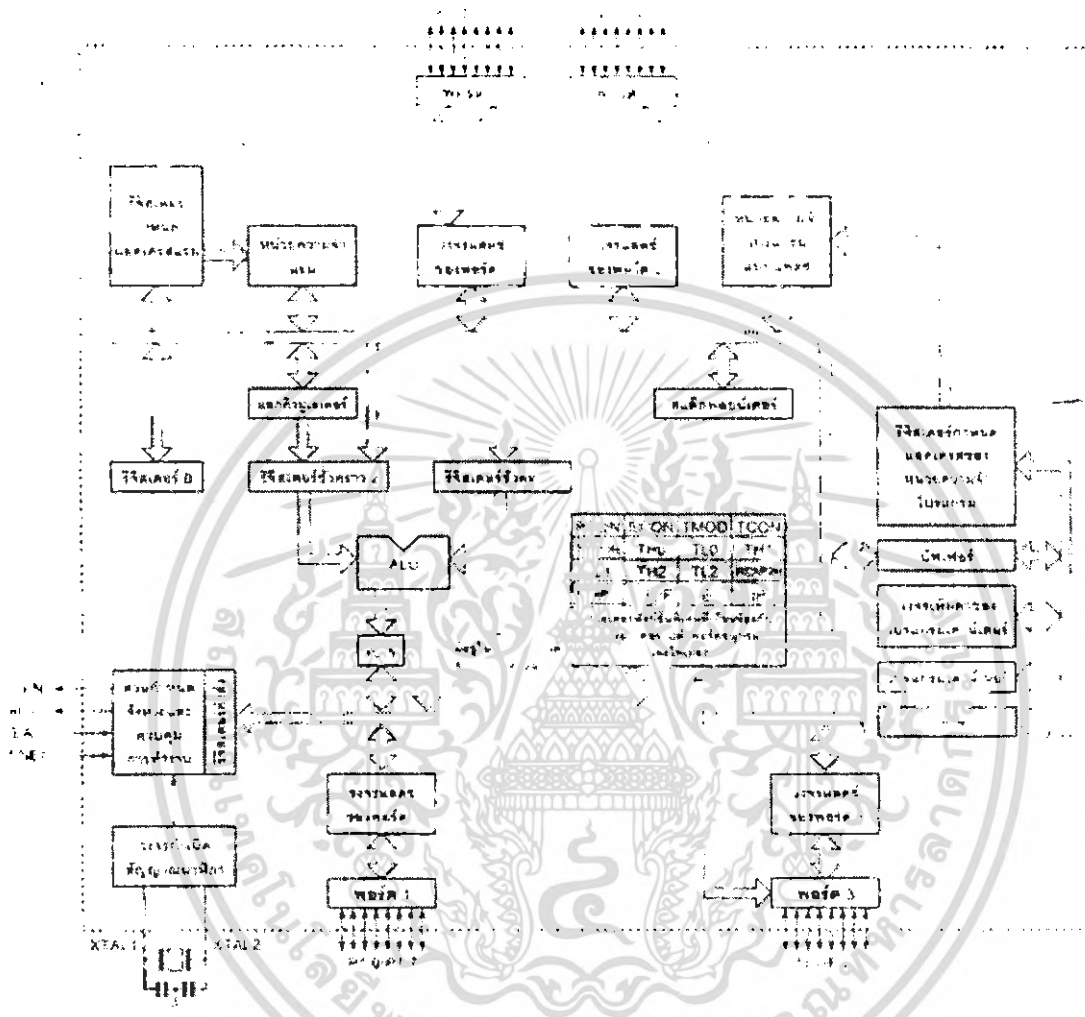
ขาพอร์ต 0(P0.0-P0.7) มี 8 ขา แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุตและเอาต์พุต สำหรับใช้งานทั่วไป ถ้าหากต้องการกำหนดให้ขาพอร์ต 0 ขาใดขาหนึ่งเป็นอินพุตสามารถทำได้โดยการเขียนข้อมูล 1 ไปยังแต่ละบิตของพอร์ตที่ต้องการต่อด้วย ส่งผลให้ขาพอร์ตนั้นมีสถานะปล่อยลอย จึงมีอินพุตอิมพีแดนซ์สูง สามารถใช้งานเป็นขาพอร์ตอินพุตได้ นอกจากนี้ขาพอร์ตนี้ยังถูกใช้งานในการติดต่อกับขาแอดเดรสไบต์ต่ำของหน่วยความจำภายนอก และขาข้อมูล โดยใช้กระบวนการมัลติเพล็กซ์เข้าช่วย เพื่อสลับการทำงานเป็นได้ทั้งขาติดต่อกับอิดเรสและขาข้อมูล

ขาพอร์ต 1(P1.0-P1.7) มี 8 ขา แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุตและเอาต์พุต สำหรับใช้งานทั่วไป ถ้าหากต้องการกำหนดให้ขาพอร์ตใดเป็นอินพุตสามารถทำได้โดยการเขียนข้อมูล 1 ไปยังแต่ละบิตของพอร์ตที่ต้องการต่อด้วยนอกจากนั้นในอนุกรม AT89Sxx จะใช้ขา P1.0 เป็นขาอินพุตสำหรับนับค่าของไทเมอร์ 2 และ P1.1 เป็นขาอินพุตทริกเกอร์ของไมเมอร์ 2 ในขณะที่ขา P1.4 ถึง P1.7 เป็นขาสำหรับเชื่อมต่อแบบ SPI เพื่อโปรแกรมข้อมูลในหน่วยความจำแฟลช ภายใน

ขาพอร์ต 2(2.0-P2.7) มี 8 ขา แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุตและเอาต์พุต สำหรับใช้งานทั่วไป ถ้าหากต้องการกำหนดให้ขาพอร์ต 0 ขาใดขาหนึ่งเป็นอินพุตสามารถทำได้โดยการเขียนข้อมูล 1 ไปยังแต่ละบิตของพอร์ตที่ต้องการต่อด้วย ส่งผลให้ขาพอร์ตนั้นมีสถานะปล่อยลอย จึงมีอินพุตอิมพีแดนซ์สูง สามารถใช้งานเป็นขาพอร์ตอินพุตได้ นอกจากนี้ขาพอร์ตนี้ยังถูกใช้งานในการติดต่อกับขาแอดเดรสไบต์สูงของหน่วยความจำภายนอก

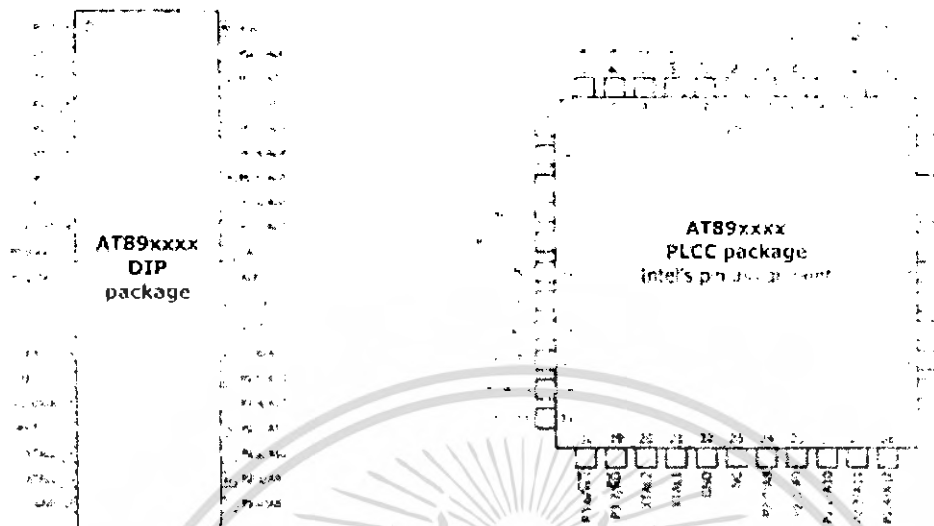
ขาพอร์ต 3(P3.0-P3.7) มี 8 ขา แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุตและเอาต์พุต สำหรับใช้งานทั่วไป ถ้าหากต้องการกำหนดให้ขาพอร์ต 0 ขาใดขาหนึ่งเป็นอินพุตสามารถทำได้โดยการเขียนข้อมูล 1 ไปยังแต่ละบิตของพอร์ตที่ต้องการต่อด้วย ส่งผลให้ขาพอร์ตนั้นมีสถานะปล่อยลอย จึงมีอินพุตอิมพีแดนซ์สูง สามารถใช้งานเป็นขาพอร์ตอินพุตได้ นอกจากนี้ขาพอร์ตนี้ยังมีหน้าที่การใช้งานพิเศษ ดังมีรายละเอียดข้างต้นดังนี้

- P3.0 เป็นขาอินพุตสำหรับข้อมูลจากการสื่อสารแบบอนุกรม หรือขา RxD
- P3.1 เป็นขาอินพุตสำหรับข้อมูลจากการสื่อสารแบบอนุกรม หรือขา TxD
- P3.2 เป็นขาอินพุตรับสัญญาณอินเทอร์รัปต์จากภายนอกช่อง 0
- P3.3 เป็นขาอินพุตรับสัญญาณอินเทอร์รัปต์จากภายนอกช่อง 1
- P3.4 เป็นขาอินพุตรับสัญญาณ ไทเมอร์จากภายนอกช่อง 0
- P3.5 เป็นขาอินพุตรับสัญญาณอินเทอร์รัปต์จากภายนอกช่อง 1
- P3.6 เป็นขาสัญญาณ WR ในกรณีที่ใช้เชื่อมต่อกับหน่วยความจำภายนอก
- P3.7 เป็นขาสัญญาณ RD ในกรณีที่ใช้เชื่อมต่อกับหน่วยความจำภายนอก



รูปที่ 2.22 รายละเอียดโครงสร้างหลักของไมโครคอนโทรลเลอร์ MCS 51 แบบแฟลชของ Atmel

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.23 การจัดขามาตรฐานของไมโครคอนโทรลเลอร์ MCS 51 แบบแฟลชของ Atmel

2.7.2 โครงสร้างและการทำงานของพอร์ต

ไมโครคอนโทรลเลอร์ MCS 51 แบบแฟลชมีพอร์ตให้ใช้งานทั้งสิ้น 4 พอร์ต แต่ละพอร์ตมีขนาด 8 บิตเป็นพอร์ตแบบ 2 ทิศทาง กล่าวคือ สามารถเป็นได้ทั้งอินพุตสำหรับรับสัญญาณของข้อมูลและเอาต์พุตของข้อมูลออก ทุกพอร์ตของไมโครคอนโทรลเลอร์ แบบแฟลชมีวงจรถ่ายและวงจรถับตลอดจนบัฟเฟอร์อินพุต

2.7.3 การใช้งานเป็นพอร์ตอินพุต

เนื่องด้วยพอร์ตทั้งหมดของไมโครคอนโทรลเลอร์ MCS 51 แบบแฟลชสามารถเป็นได้ทั้งอินพุตและเอาต์พุต ดังนั้นจึงมีความจำเป็นอย่างยิ่งที่ต้องทำความเข้าใจถึงการกำหนดลักษณะการทำงานให้แก่พอร์ตของไมโครคอนโทรลเลอร์ MCS 51 แบบแฟลช

ในการกำหนดให้เป็นพอร์ตอินพุต ต้องเริ่มต้นด้วยการเขียนข้อมูล 1 มาที่แต่ละบิตของพอร์ตที่ต้องการใช้งานเป็นอินพุต เพื่อหยุดการทำงานของเฟตที่ใช้ในการขับสัญญาณเอาต์พุตของบิตนั้น ๆ ทำให้ขาของสัญญาณของพอร์ตเชื่อมต่อกับวงจรพูลอัพภายในโดยตรง ส่งผลให้ขoport นั้นมีลอจิกเป็น 1 สามารถรับสัญญาณลอจิก 0 จากอุปกรณ์ภายนอกได้ง่าย สัญญาณข้อมูลจากอุปกรณ์ภายนอกจะถูกส่งเข้ามาเก็บไว้ในวงจรบัฟเฟอร์ภายในพอร์ต แล้วรอให้ซีพียูมาอ่านเข้าไป เมื่อเป็นเช่นนี้ อุปกรณ์

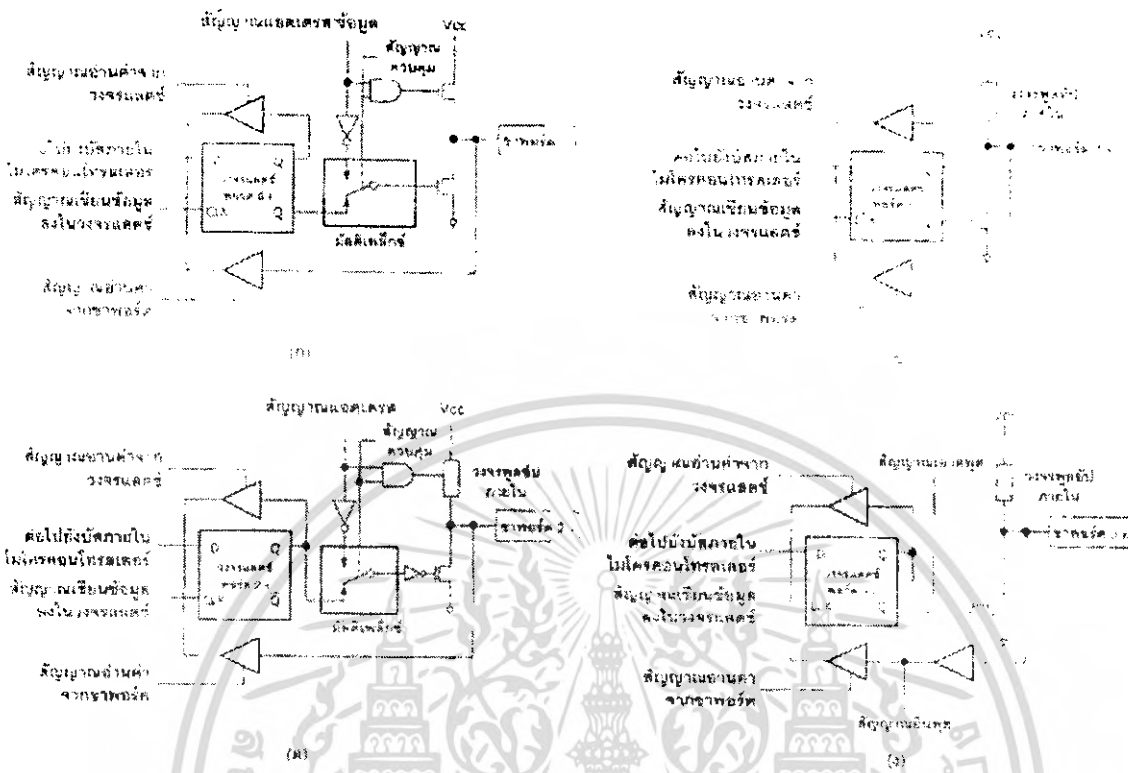
ภายนอกที่เชื่อมต่อกับพอร์ตอินพุตของไมโครคอนโทรลเลอร์ แบบแฟลชควรถูกกำหนดให้ทำงานในสถานะลอจิก 0

ขา	เบอร์ของไมโครคอนโทรลเลอร์	หน้าที่พิเศษ
P1.0	AT89C52/AT89Sxx	อินพุตนับค่าของไทเมอร์/เคาน์เตอร์ 2
P1.1	AT89C52/AT89Sxx	ควบคุมทิศทางของสัญญาณ
P1.4	AT89Sxx	เป็นอุปกรณ์เลฟในระบบการติดต่อแบบ SPI
P1.5	AT89Sxx	ใช้ในการติดต่อกับพอร์ต SPI
P1.6	AT89Sxx	ใช้ในการติดต่อกับพอร์ต SPI
P1.7	AT89Sxx	ขาสัญญาณนาฬิกาของการติดต่อกับพอร์ต SPI

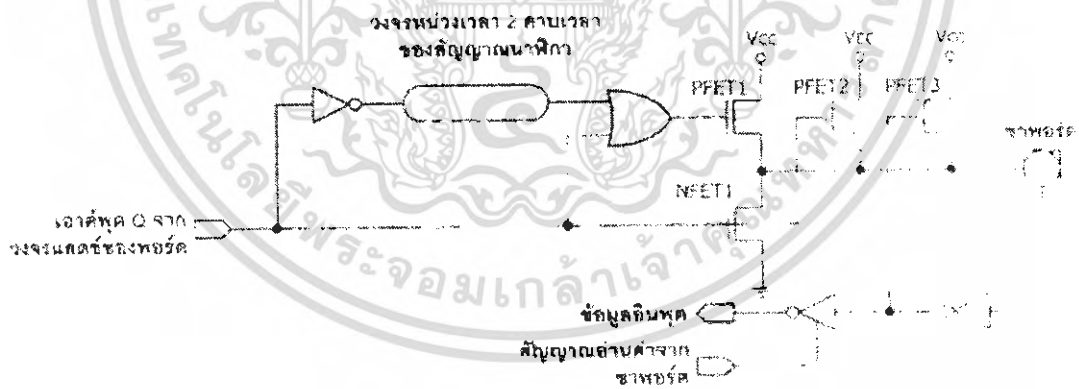
ตารางที่ 2.13 หน้าที่พิเศษของพอร์ต 1 ในไมโครคอนโทรลเลอร์ MCS 51 แบบ แฟลชของ Atmel

2.7.4 การใช้งานเป็นพอร์ตเอาต์พุต

โดยปกติแล้ว ขาพอร์ตจะกำหนดให้มีลักษณะกำหนดให้มีลักษณะเป็นเอาต์พุตอยู่แล้ว ดังนั้นจึงสามารถส่งข้อมูลได้อย่างง่ายดาย และตรงไปตรงมา กล่าวคือเมื่อต้องการส่งข้อมูล 0 ออกไปทางเอาต์พุตก็ให้เขียนข้อมูล 0 ไปยังวงจรวอร์เรตซ์ ซึ่งก็จะสามารถส่งต่อไปจับเฟด ทำให้เฟดทำงานที่ขาพอร์ตที่กำหนดให้ทำงานก็จะเกิดลอจิก 0 ขึ้น ในทางที่ขาพอร์ตเชื่อมต่อกับวงจรวอร์เรตซ์ในเกิดเป็นลอจิก 1 ไปยังวงจรวอร์เรตซ์ วงจรจับก็จะสามารถหยุดทำงาน ทำให้ที่ขาพอร์ตเชื่อมต่อกับวงจรวอร์เรตซ์นอกเกิดเป็นลอจิก 1 ที่ขาพอร์ตนั้น ซึ่งจะคล้ายกับการกำหนดให้เป็นขาอินพุตบัฟเฟอร์ แต่ถ้าเป็นเอาต์พุตไม่มีการอ่านข้อมูลที่บัฟเฟอร์แต่อย่างใด เว้นแต่ในกรณีที่ต้องการตรวจสอบข้อมูลที่ส่งออกมาทางเอาต์พุต



รูปที่ 2.24 วงจรภายในของพอร์ตทุกพอร์ตในไมโครคอนโทรลเลอร์ MCS 51 แบบแฟลช



วงจรพัลส์ประกอบด้วยเฟตชนิดที่สามชนิด 3 ตัวคือ PFET1, PFET2 และ PFET3 โดย NFET1 จะทำงานเมื่อได้รับพัลส์ 0 จากขาพอร์ต 0 ของไมโครคอนโทรลเลอร์ เมื่อได้รับพัลส์ 1 จากขาพอร์ต 0 ของไมโครคอนโทรลเลอร์ จะทำให้ NFET1 ทำงานตามไปด้วย ทำให้เกิดพัลส์ที่ขาพอร์ต 0 ของไมโครคอนโทรลเลอร์ เป็นเวลา 2 ไมโครวินาที หลังจากนั้นเกิดพัลส์ที่ขาพอร์ต 0 ของไมโครคอนโทรลเลอร์ เป็นเวลา 2 ไมโครวินาที ทำให้เกิดพัลส์ที่ขาพอร์ต 0 ของไมโครคอนโทรลเลอร์ เป็นเวลา 2 ไมโครวินาที

รูปที่ 2.25 วงจรพัลส์ภายในพอร์ตของไมโครคอนโทรลเลอร์ MCS 51 แบบแฟลช

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.5 การอ่านค่าลอจิกจากพอร์ต

ในกรณีที่พอร์ตต่อกับขาเบสทรานซิสเตอร์ชนิด NPN และขาอิมิตเตอร์ของทรานซิสเตอร์ตัวนั้นต่อจากราวด์ หากมีการส่งข้อมูล 1 ไปยังทรานซิสเตอร์ จะทำให้ทรานซิสเตอร์ทำงานสถานะลอจิกที่ขาพอร์ตจะเป็น 0 เนื่องจากเมื่อทรานซิสเตอร์ทำงาน จะเสมือนว่าขาพอร์ตนั้นถูกต่อลงกราวด์ทำให้หากอ่านค่าลอจิกที่ขาพอร์ตจะได้ผลตรงข้ามกับที่ส่งมา แต่หากทำงานอ่านค่าลอจิกที่วงจรเลขจะได้อ่านค่าที่ตรงกับค่าที่ต้องการส่งจริงดังนั้น ในการอ่านค่าลอจิกจากพอร์ตจึงต้องเลือกวิธีการให้เหมาะสมกับอุปกรณ์ที่นำมาต่อด้วย

2.7.6 รีจิสเตอร์ควบคุมการทำงานของพอร์ตอนุกรมหรือ SCON (Serial port Control Register)

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
SM0	SM1	SM2	REN	TB8	RB8	TI	RI

ตารางที่ 2.14 ตารางรีจิสเตอร์ควบคุมการทำงานของพอร์ตอนุกรม

SM0/FE (Serial port mode bit 0/Framing error bit) : ปกติจะใช้ร่วมกับบิต SM1 เพื่อกำหนดโหมดการทำงานของพอร์ตอนุกรม การเข้าถึงบิตนี้จะเกิดขึ้นได้ก็ต่อเมื่อมีการเคลียร์บิต SMOD ซึ่งก็คือบิต 6 ของรีจิสเตอร์ PCON ในกรณีที่ใช้ความสามารถตรวจจับความผิดพลาดของเฟรมข้อมูล บิตนี้จะใช้แจ้งความผิดพลาดที่เกิดขึ้น โดยจะเซตเป็น 1 ทันทีที่พบว่าไม่สามารถตรวจจับบิตหยุดหรือบิตปิดท้ายของข้อมูลของพอร์ตอนุกรมได้ การเอนเอเบิลความสามารถนี้ทำได้โดยการเซตบิต SMOD ในรีจิสเตอร์ PCON การเคลียร์บิตนี้ต้องกระทำทางซอฟต์แวร์เท่านั้น

SM1 (Serial port mode bit 1) : ใช้ร่วมกับบิตSM0ในการกำหนดโหมดการทำงานของพอร์ตอนุกรม

SM0	SM1	โหมด	รายละเอียด	อัตราบอด
0	0	0	ชิพรีจิสเตอร์	ความถี่สัญญาณนาฬิกา/6
0	1	1	UART 8 บิต	ปรับค่าได้
1	0	2	UART 9 บิต	ความถี่สัญญาณนาฬิกา/32
1	1	3	UART 9 บิต	ปรับค่าได้

ตารางที่ 2.14 ตาราง SM1 (Serial port mode bit 1) : ใช้ร่วมกับบิตSM0ในการกำหนดโหมดการทำงานของพอร์ตอนุกรม

SM1 (Serial port mode bit 2) : ใช้ในการเอ็นเอเบิลความสามารถการรับรู้แอดเดรสในการติดต่ออัตโนมัติเมื่อมีการเชื่อมต่อไมโครคอนโทรลเลอร์หลายตัวเข้าด้วยกัน โดยความสามารถนี้จะใช้ได้ก็ต่อเมื่อวงจรพอร์ตอนุกรมของไมโครคอนโทรลเลอร์ทำงานในโหมด2หรือ3 ถ้าบิตSM2เป็น1 บิตRIจะไม่เซต เว้นแต่ข้อมูลบิตที่9ที่รับเข้ามาเป็น1จะแจ้งว่า สามารถติดต่อได้และข้อมูลที่รับเข้ามาคือค่าแอดเดรสที่ต้องการติดต่อด้วย

ในกรณีที่พอร์ตอนุกรมทำงานอยู่ในโหมด1 ถ้าบิตSM2เซต บิตRIจะไม่เปลี่ยนแปลงจนกว่าจะได้รับข้อมูลบิตหยุดหรือบิตปิดท้าย และข้อมูลที่รับได้จะเป็นข้อมูลแอดเดรสที่ต้องการติดต่อด้วย

ในกรณีที่วงจรพอร์ตอนุกรมทำงานในโหมด0 บิตSM2นี้เป็น0

REN (Receive enable bit) : ใช้เอ็นเอเบิลความสามารถในการรับรู้ข้อมูลของวงจรพอร์ตอนุกรม ถ้าเป็น1คือเอ็นเอเบิล 0คือดิสเอเบิล การเซตหรือเคลียร์บิตนี้ต้องการทำด้วยซอฟต์แวร์

TB8 (Transmit data bit 8) : ใช้กับข้อมูล 8 หรือบิตที่ 9 ที่ต้องการส่งออกทางพอร์ตอนุกรมเมื่อทำงานในโหมด 2 และ 3

RB8(Receive data bit 8) : ใช้เก็บข้อมูลบิต 8 ของข้อมูลที่รับเข้ามาของพอร์ตอนุกรมเมื่อทำงานในโหมด 2 และ 3 โหมด 1 ถ้าบิต SM = 0 ข้อมูลของบิตหยุดจะเก็บไว้ที่บิตนี้ ไม่ใช้งานบิตนี้ในโหมด 0

TI(Transmit interrupt falg) : บิตแสดงการเกิดอินเตอร์รัปต์จากการส่งข้อมูลออกทางพอร์ตอนุกรมเมื่อทำงานในโหมด 0 บิตนี้จะเซตเมื่อมีการส่งข้อมูลบิตที่ 7 หรือบิตที่ 8 ออกไป แต่ถ้าทำงานในโหมด

อื่น บิตนี้จะเซตเมื่อมีการเริ่มต้นส่งบิตหยุดหรือบิตปิดท้าย การเคลียร์บิตนี้ต้องกระทำด้วยกระบวนการทางซอฟต์แวร์

RI(Receive interrupt flag) : บิตแสดงการเกิดอินเทอร์รัปต์เนื่องจากการรับข้อมูลเข้ามาของพอร์ตอนุกรม เมื่อทำงานในโหมด 0 บิตนี้จะเซตเมื่อรับข้อมูลบิต 7 หรือบิต 8 เสร็จสมบูรณ์ แต่ถ้าทำงานในโหมดอื่น บิตนี้จะเซตเมื่อการรับบิตปิดท้ายดำเนินไปได้ครึ่งทาง นอกจากนี้การเซตบิตนี้ยังมีเงื่อนไขที่กำหนดโดยบิต SM 2 ร่วมด้วย การเคลียร์บิตนี้ต้องกระทำด้วยกระบวนการทางซอฟต์แวร์

โดยส่วนใหญ่แล้วมักจะตั้งค่าของ SCON ดังนี้

SCON = 0 x 40; พอร์ตอนุกรมทำงานในโหมด 1 สามารถส่งข้อมูลได้อย่างเดียว

SCON = 0 x 50; พอร์ตอนุกรมทำงานในโหมด 1 สามารถรับและส่งข้อมูลได้



2.8 หลักการ SMS

2.8.1 หลักการรับส่ง SMS

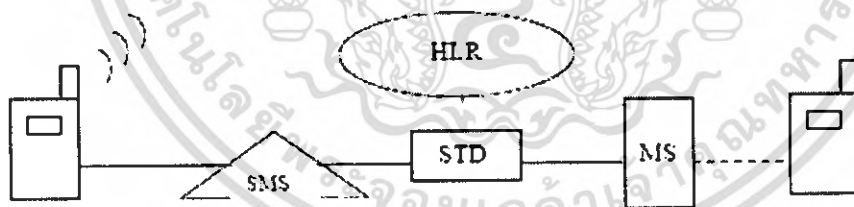
SMS : Short Message Service คือ การให้บริการส่งข้อความสั้นไปยังโทรศัพท์มือถือแบบดิจิทัลโดยแต่ละข้อความสามารถบรรจุตัวอักษรสูงสุดได้ 160 ตัวอักษร (อักษรภาษาอังกฤษ) นอกจากนี้ยังสามารถส่งข้อความไปที่เครื่อง Fax, PC หรือ Internet address อีกด้วย

ระบบ SMS ในระบบเครือข่ายโทรศัพท์มือถือ รองรับโดยระบบ GSM (Global System for Mobile Communication), TDMA (time division multiple access) และ CDMA (code division multiple access)

เมื่อ SMS ถูกส่งจากโทรศัพท์มือถือเครื่องหนึ่ง ข้อความนั้นจะถูกส่งไปที่ Short Message Service Center (SMSC) จากนั้นจึงจะส่งไปยังโทรศัพท์มือถือเครื่องรับอีกทอดหนึ่ง โดยมีกระบวนการดังนี้

1. SMSC จะส่ง SMS Request ไปยัง Home location register (HLR) เพื่อหาตำแหน่งของผู้รับ
2. เมื่อ HLR ได้รับสัญญาณ Request ก็จะต้องส่งสถานะของผู้รับ (subscriber's status) กลับมายัง SMSC คือ

- 1) สถานะของเครื่องรับ Inactive หรือ Active
- 2) ตำแหน่งของเครื่องรับ



รูปที่ 2.1 หลักการรับ-ส่ง SMS

ถ้าสถานะของเครื่องรับเป็น Inactive แล้ว SMSC จะเก็บข้อความไว้ช่วงเวลาหนึ่ง และเมื่อใดที่เครื่องรับมีสถานะ Active แล้ว HLR จะส่ง SMS Notification ไปยัง SMSC และ SMSC ก็จะตอบรับข้อความนั้นไว้ จากนั้น SMSC จะส่งผ่านข้อความไปในรูปแบบ Short Message Delivery Point-to-Point ไปยังระบบบริการ โดยระบบจะทำการเรียกไปยังเครื่องรับ และถ้าเครื่องรับมีการตอบรับกลับมา ข้อความก็จะถูกส่งตามไปและ SMSC จะได้รับการตอบยืนยันว่า ข้อความได้ถูกรับโดยปลายทางเรียบร้อยแล้ว หลังจากนั้นข้อความจะมีสถานะเป็น SENT และจะไม่ถูกส่งอีก

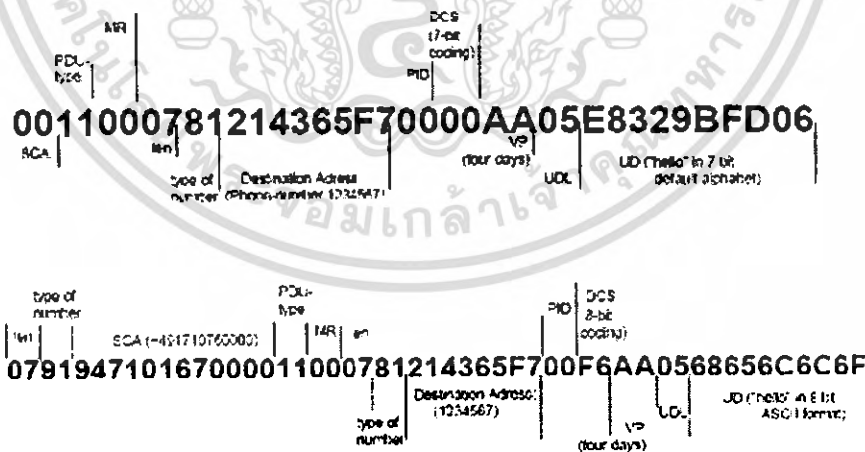
การรับส่งข้อมูล SMS มี 2 โหมด คือ Text Mode และ PDU Mode (Protocol Description Unit Mode) การส่งข้อมูล Text Mode นั้นจะเป็นการนำข้อความที่ต้องการส่งมาเข้ารหัสก่อน แล้วค่อยส่งข้อมูลใน PDU Mode

2.8.2 PDU MODE

PDU : Protocol Description Unit คือ รูปแบบของข้อมูลที่ใช้ในการส่งข้อความสั้น ซึ่งเกิดจากการนำข้อมูลดิจิทัลมาเข้ารหัสเพื่อใช้ในการส่งผ่านข้อมูลผ่าน Air Interface โดยการรับและส่งข้อมูล SMS ในแบบ

PDU MODE จะมีส่วนประกอบและโครงสร้างที่แตกต่างกัน

PDU CODE ข้อมูล SMS ที่อยู่ใน PDU MODE ประกอบด้วยเลขฐาน 10 และเลขฐาน 16 โดยตัวเลขแต่ละคู่เรียกว่า Octet



รูปที่ 2.2 ข้อมูลใน SMS ในรูปแบบ PDU MODE

2.8.2.1 การรับข้อมูล SMS

ข้อมูลที่ได้รับจะประกอบด้วยข้อมูลของผู้ส่ง, ข้อมูล SMS Service Center (SMSC), Time Stamp และอื่นๆ ตามด้วยส่วนของข้อความซึ่งจะอยู่ท้ายสุด

2.8.2.2 การส่งข้อมูล SMS

ข้อมูลที่ส่งจะประกอบด้วย ข้อมูลของ SMSC, หมายเลขผู้รับ, ช่วงเวลาหมดอายุของข้อความ, และอื่นๆ ปิดท้ายด้วยข้อมูลในส่วนของข้อความสั้น

2.8.2.3 การถอดรหัสตัวอักษรชนิด 7 บิต

ในส่วนของข้อมูลที่เป็นข้อความสั้นในกรณีที่เป็นตัวอักษรภาษาอังกฤษ อักษรแต่ละตัวจะใช้รหัสขนาด 7 บิต (7 bits default alphabet) และสำหรับอักษรภาษาอื่นๆ จะใช้ตัวอักษรชนิดอื่น เช่น 8 บิต หรือ 16 บิตสำหรับข้อความสั้นภาษาไทยนั้น ใช้รหัสตัวอักษรแบบ UNICODE ในที่นี้เราจะกล่าวถึงการถอดรหัสข้อความสั้นภาษาอังกฤษเท่านั้น โดยมีขั้นตอนการถอดรหัส ดังนี้

- 1) นำ PDU CODE ในส่วนของข้อความสั้น (TP-UD) ซึ่งเป็นเลขฐาน 16 มาเขียนเป็นเลขฐาน 2 ทีละ Octet
- 2) ตัวอักษรแรกเกิดจากบิตที่ 0 ถึง 6 ของ Octet แรก ตัวอักษรถัดมาเกิดจากบิตที่ 0 ถึง 5 ของ Octet ที่ 2 และนำบิตที่ 7 ของ Octet ที่ 1 มาต่อท้าย ตัวอักษรถัดไปเกิดจากบิตที่ 0 ถึง 4 ของ Octet ที่ 3 และนำบิตที่ 7 และ 6 ของ Octet ที่ 2 มาต่อท้าย ทำเช่นนี้ไปเรื่อยๆ
- 3) นำรหัสตัวอักษร 7 บิต จากข้อสอง เพิ่มบิตที่ 8 ด้วย '0' นำไปเทียบกับตารางรหัส ASCII โปรแกรม Microcontroller ที่เกี่ยวข้องกับการถอดรหัส PDU เป็น ASCII ได้แสดงไว้ที่ภาคผนวก

2.8.3 AT-COMMAND

AT-COMMAND คือ ชุดคำสั่งมาตรฐาน ที่สามารถใช้ติดต่อสื่อสารกับอุปกรณ์สื่อสารต่างๆ เช่น โมเด็ม หรือ อุปกรณ์ DTE (Data Terminal Equipment) เพื่อโต้ตอบตั้งค่าหรือสั่งงานอุปกรณ์เหล่านั้น ให้ทำงานตามที่ต้องการ และสำหรับการติดต่อกับโทรศัพท์มือถือ จะใช้ชุดคำสั่งที่เรียกว่า GSM AT COMMAND

ตัวอย่างคำสั่งที่เป็น BASIC AT COMMAND

AT	ตรวจสอบความพร้อมของอุปกรณ์ ถ้าสามารถติดต่อกันได้ อุปกรณ์จะตอบกลับมาว่า OK
ATDT phone number;	โทร ไปยังเลขหมายปลายทาง (phone number)
ATH	วางสาย
ATA	รับสาย

AT COMMAND ที่เกี่ยวข้องกับการรับ-ส่ง SMS

ชุดคำสั่ง AT COMMAND ที่ใช้กับโทรศัพท์มือถือ ได้มีอยู่มากมาย ทั้งการอ่านรุ่นโทรศัพท์มือถือ, ตรวจสอบระดับแบตเตอรี่, ตรวจสอบระดับสัญญาณ แต่ในที่นี้จะกล่าวถึงเพียงคำสั่งที่เกี่ยวข้องกับการรับ-ส่ง SMS เท่านั้น

1) Message Format (AT+CMGF) เป็นคำสั่งกำหนดรูปแบบของข้อความที่จะให้แสดงออกมา โดย

AT+CMGF = 1 คือ แสดงข้อความในรูปแบบ TEXT

AT+CMGF = 0 คือ แสดงข้อความในรูปแบบ PDU CODE

2) List Message (AT+CMGL) เป็นคำสั่งที่ให้แสดงข้อความในสถานะต่างๆ โดยจะแสดงข้อความทั้งหมด นั้นสถานะนั้น ดังนี้

AT+CMGL=0 คือ แสดงข้อความที่ได้รับแต่ยังไม่ได้อ่าน ("REC UNREAD")

AT+CMGL=1 คือ แสดงข้อความที่ได้รับและอ่านแล้ว ("REC READ")

AT+CMGL=2 คือ แสดงข้อความที่เก็บไว้และยังไม่ได้ส่ง ("STO UNSENT")

AT+CMGL=3 คือ แสดงข้อความที่เก็บไว้และส่งออกไปแล้ว ("STO SENT")

AT+CMGL=4 คือ แสดงข้อความทั้งหมด ("ALL")

หมายเหตุ .. หากกำหนด Message Format เป็น PDU CODE จะต้องเลือกสถานะ โดยใช้ตัวเลข 0 ถึง 4 แต่หากกำหนด Message Format เป็น Text จะต้องเลือกสถานะ โดยใช้ตัวอักษรที่วงเล็บด้านหลัง

```

example - HyperTerminal
File Edit View Call Transfer Help
[Icons]
AT
OK
ATDT "+6618766188"
NO DIALTONE
AT+CMGF=?
+CMGF: (0-1)

OK
AT+CMGF=0
OK
AT+CMGL=?
+CMGL: 0,1,2,3,4

OK
AT+CMGL=2
+CMGL: 19,2,58
004100098110942787F700182E050003088202003A00440020007400680061006E006B0020006100
20006C006F00740020006E00610020006A0061

OK

connected 0:01:04 Auto detect 19200 B-N-1 CAPS NUM

```

รูปที่ 2.3 ตัวอย่างคำสั่งพื้นฐานและ LIST ข้อความใน STO SENT

3) Read Message (AT+CMGR) เป็นคำสั่งที่ใช้อ่านข้อความที่เฉพาะเจาะจงได้โดยระบุตำแหน่งที่ข้อความนั้นถูกเก็บไว้

```

AT
OK
AT+CMGF=1
OK
AT+CMGR=5
+CMGR: "REC READ","+6614972787",,"04/05/02 20:37 02-28".0
It's ok wh. he understands it when it drunk it lol 555;0 he is a nice guy now.

OK

```

รูปที่ 2.4 ตัวอย่างการอ่านข้อความ

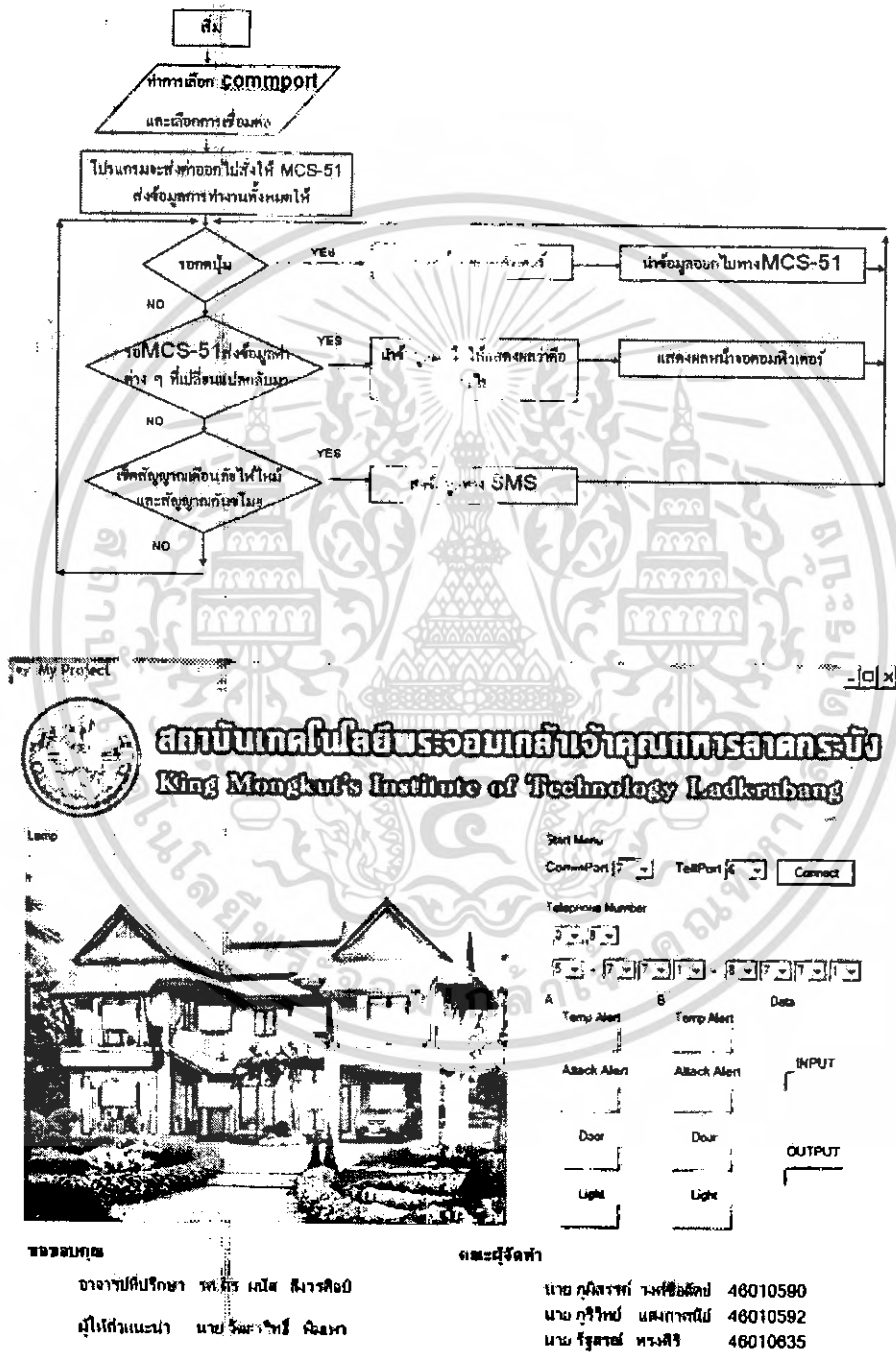
4) Set the Message center number (AT+CSCA="XXX") เป็นคำสั่งที่ใช้ระบุหมายเลขของ SMSC เพื่อจะทำการส่ง SMS ต่อไปได้

5) Send Message (AT+CMGS="YYY") เป็นคำสั่งที่ใช้ระบุเลขหมายปลายทางที่ต้องการส่ง SMS ไปถึง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3 การทำงานของ

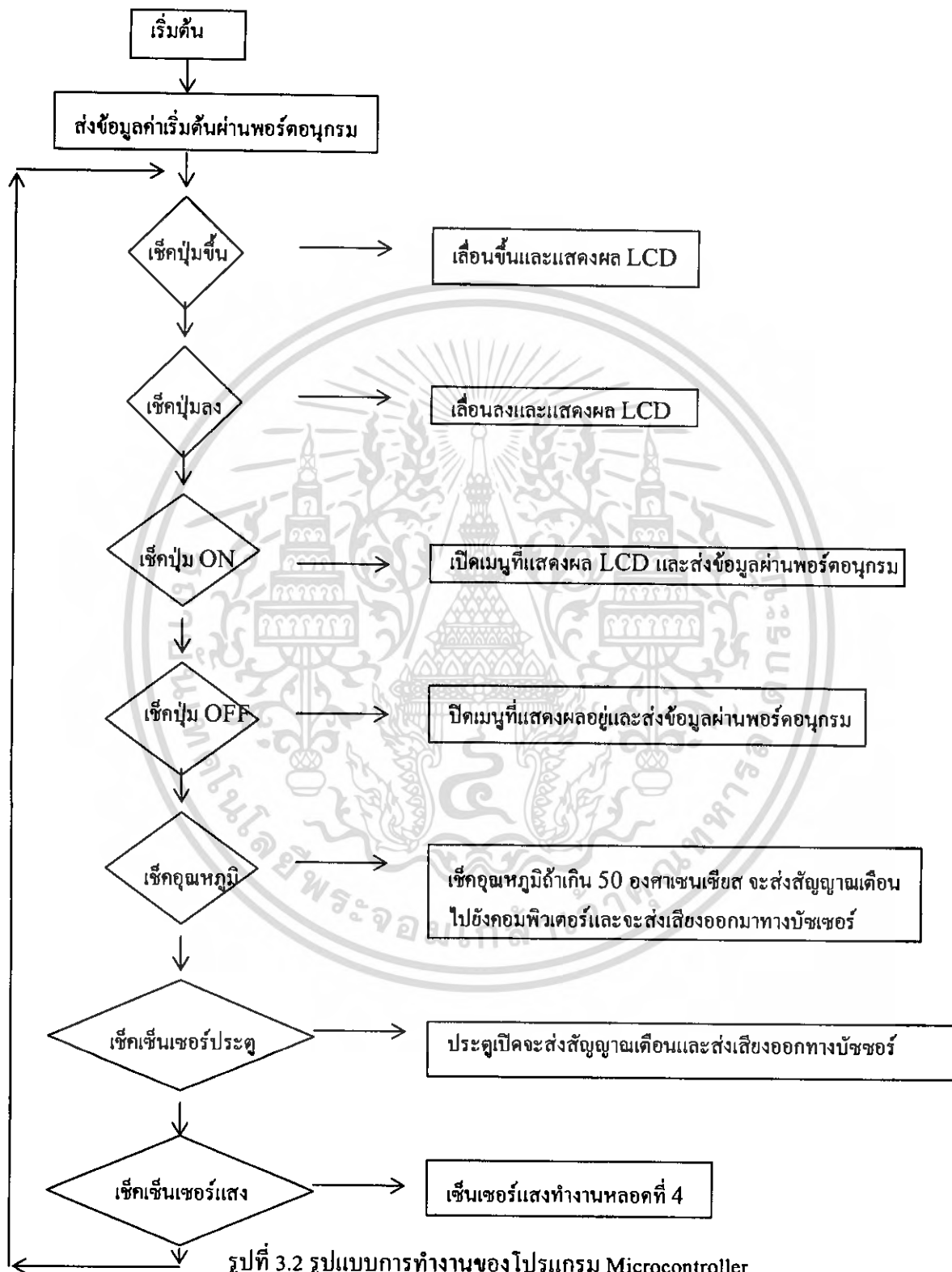
Visual Basic



รูปที่ 3.1 รูปแบบการทำงานของ โปรแกรม Visual Basic และ User interface

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Microcontroller

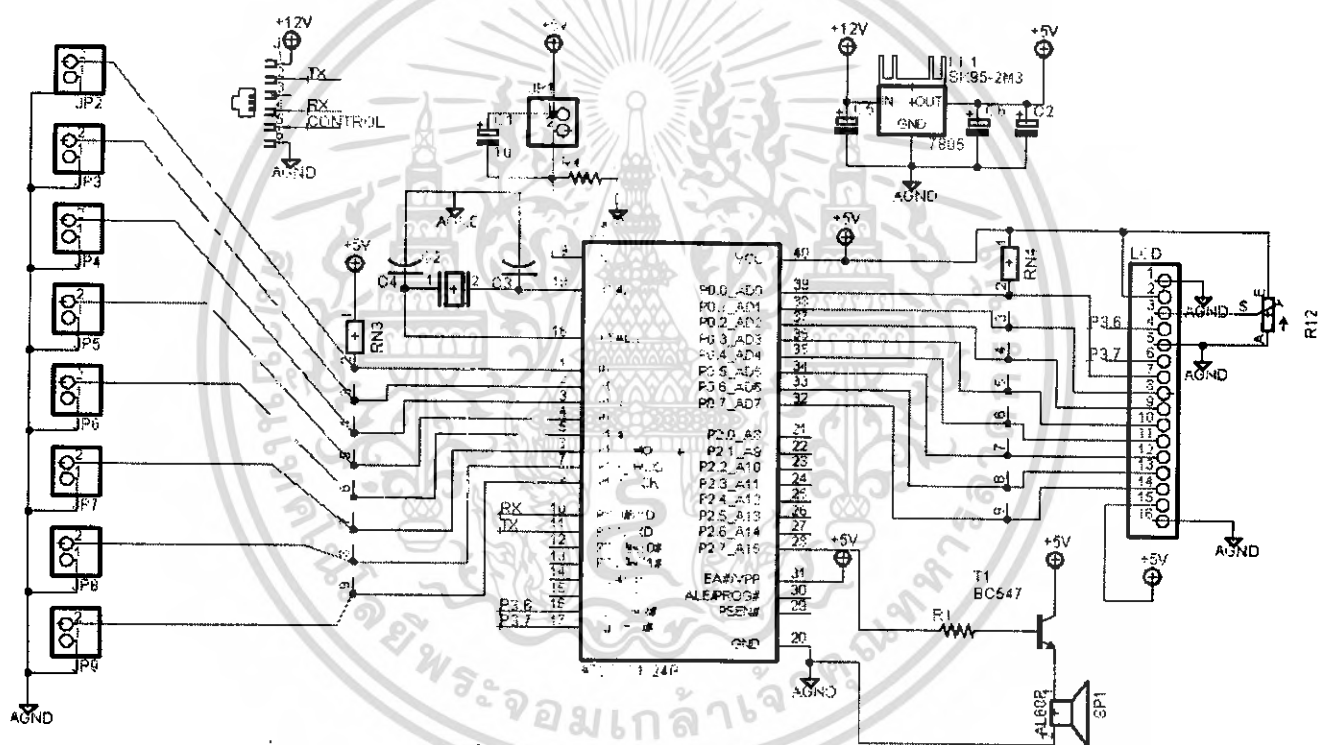


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4 วงจรที่ออกแบบวงจรควบคุมหลัก (Master)

วงจรประกอบด้วย ไมโครคอนโทรลเลอร์เบอร์ AT89S52 โดย

- พอร์ต 0 เชื่อมต่อกับจอแสดงผล LCD16x2
- พอร์ต 1 ใช้งานเป็นอินพุทของปุ่มกดสั่งงาน
- พอร์ต 2.0 ใช้เป็นขาสัญญาณควบคุมทิศทางของการรับ-ส่ง
- พอร์ต 2.7 ต่อกับทรานซิสเตอร์เพื่อขับกระแสให้ขั้วเซอร์ทำงาน
- พอร์ต 3.0 และ 3.1 ใช้เป็นขาสัญญาณสื่อสารแบบอนุกรม



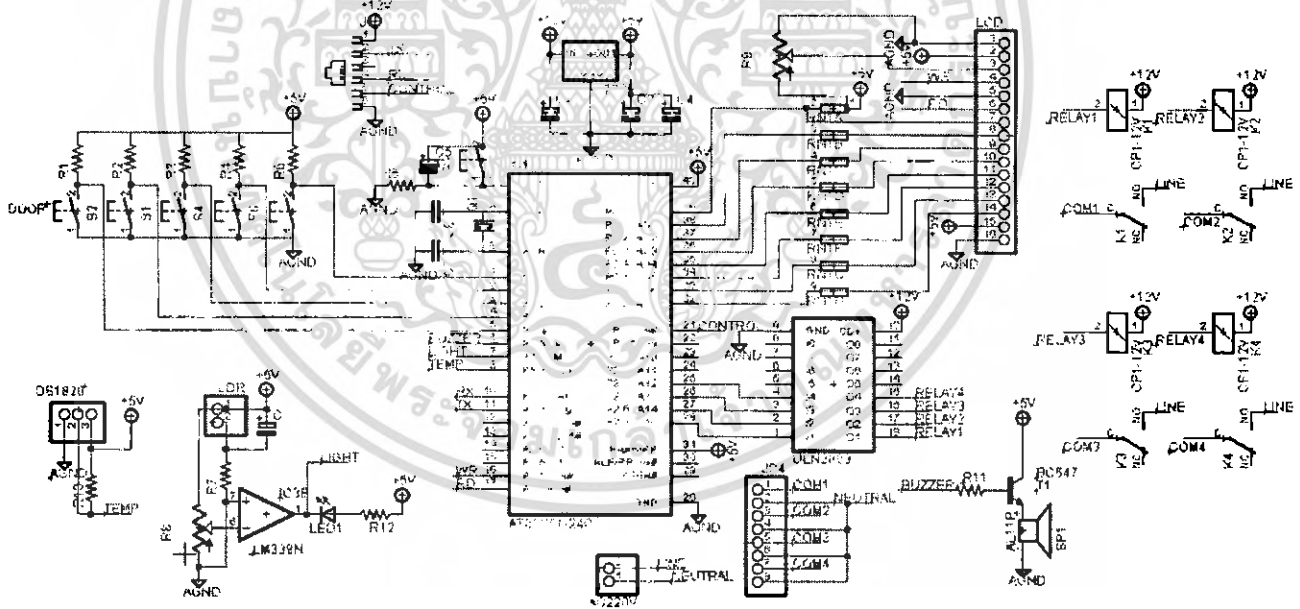
รูปที่ 4.1... วงจรควบคุมหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรควบคุมย่อย (Slave)

วงจรประกอบด้วย ไมโครคอนโทรลเลอร์เบอร์ AT89S52 โดย

- พอร์ต 0 เชื่อมต่อกับจอแสดงผล LCD 16x2
- พอร์ต 1.0-1.3 ใช้งานเป็นอินพุตของปุ่มกดสั่งงาน
- พอร์ต 1.4 ต่อกับลิมิตสวิตช์เพื่อเป็นเซนเซอร์ติดที่ประตู
- พอร์ต 1.5 ต่อกับบัสเซอร์
- พอร์ต 1.6 ต่อกับวงจรเซนเซอร์ตรวจจับแสง โดยใช้ LDR ต่อกับ comparator เบอร์ LM339
- พอร์ต 1.7 ต่อกับไอซีวัดอุณหภูมิเบอร์ DS1820
- พอร์ต 2.0 ใช้เป็นขาสัญญาณควบคุมทิศทางของการรับ-ส่ง
- พอร์ต 2.4-2.7 ต่อกับ ไอซีเบอร์ ULN2803 เพื่อขับกระแสให้ขดลวดของรีเลย์ทำงาน
- พอร์ต 3.0 และ 3.1 ใช้เป็นขาสัญญาณสื่อสารแบบอนุกรม

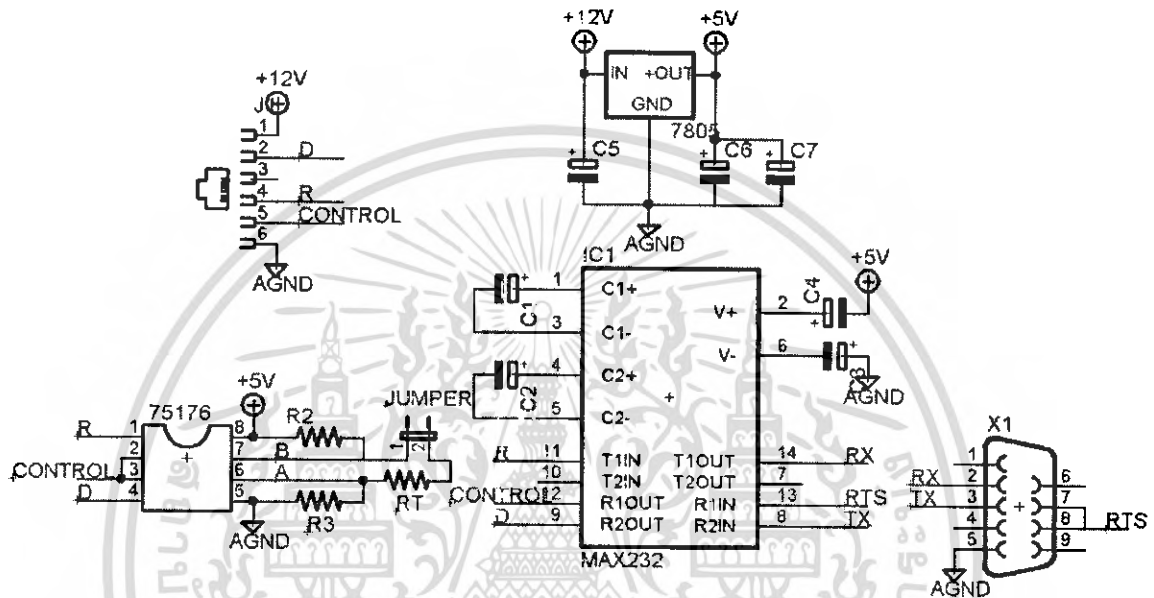


รูปที่ 4.2.. วงจรควบคุมย่อย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรถ่ายโอนข้อมูล (RS-485 Network)

ประกอบด้วยสองส่วนหลักคือ ไอซีเบอร์ SN75176 ซึ่งเป็นตัวแปลงสัญญาณข้อมูลอนุกรมแบบ TTL ให้เป็นแบบ Differential Mode และไอซีเบอร์ MAX232 ทำหน้าที่แปลงระดับสัญญาณแบบ RS-232 จากคอมพิวเตอร์ให้เป็นแบบ TTL เพื่อใช้กับ 75176



รูปที่ 4.3.. วงจรถ่ายโอนข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5 ผลการทดลอง การสื่อสารอนุกรมแบบRS-485

การทดลองนี้ใช้คอมพิวเตอร์ในการทดลอง โดยใช้โปรแกรมไฮเปอร์เทอร์มินอลในการแสดงผลและรับส่งข้อมูล ส่วนวงจรที่ใช้สื่อสารแบบRS-485นั้น ใช้การต่อพอร์ตอนุกรมจากคอมพิวเตอร์มาที่โปรโตบอร์ดซึ่งมีวงจรแปลงสัญญาณโดยใช้ MAX232และ SN75176

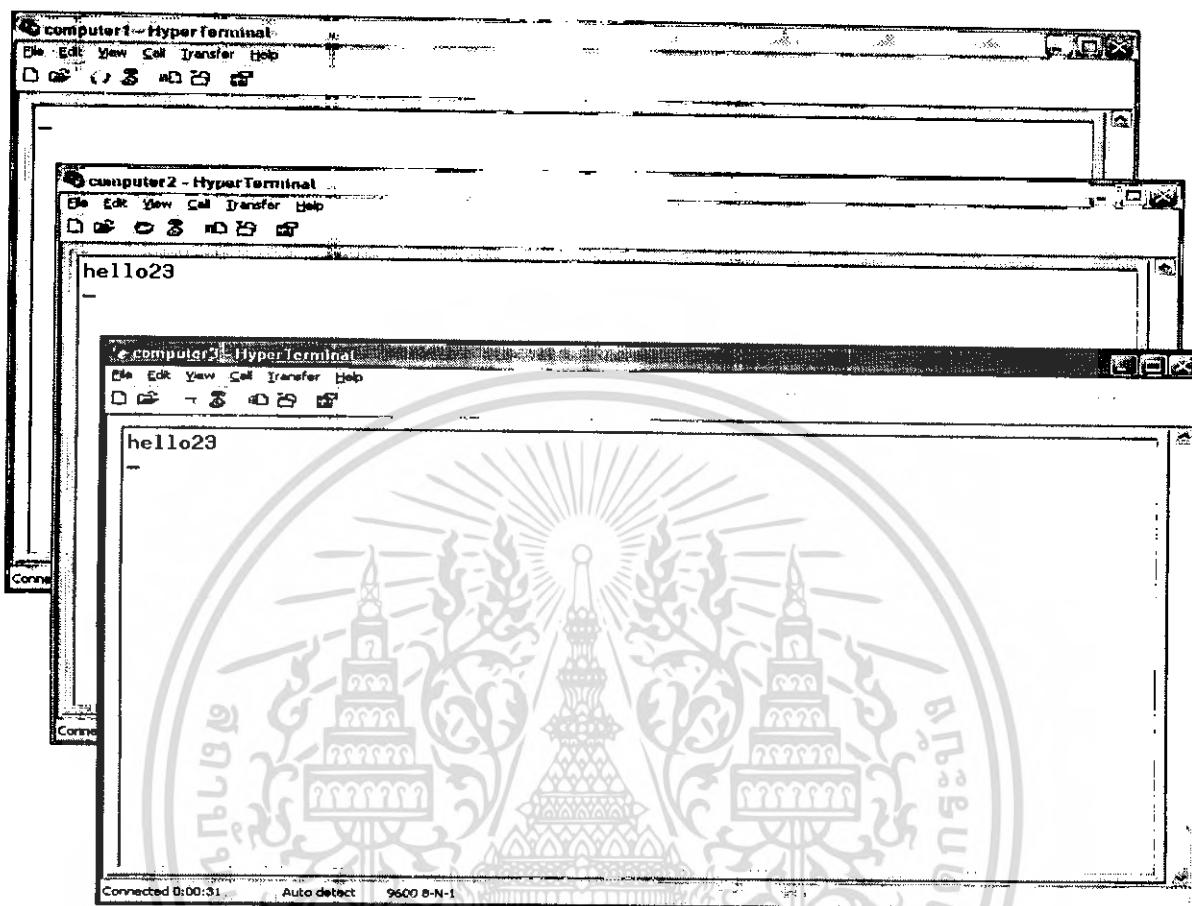
การสื่อสารแบบ1ต่อ1

ทิศทางของข้อมูลจะเป็นไปตามที่ได้ตั้งไว้ คอมพิวเตอร์เครื่องที่ตั้งให้ขา/REและDEเป็น 5Vจะส่งข้อมูลได้อย่างเดียว แต่รับข้อมูลไม่ได้ ส่วนเครื่องที่ให้ขา/REและDEเป็น0Vจะรับข้อมูลได้อย่างเดียว แต่ส่งไม่ได้

การสื่อสารแบบNetwork

ใช้คอมพิวเตอร์3เครื่อง โดยมีขั้นตอนการทดลอง ดังนี้

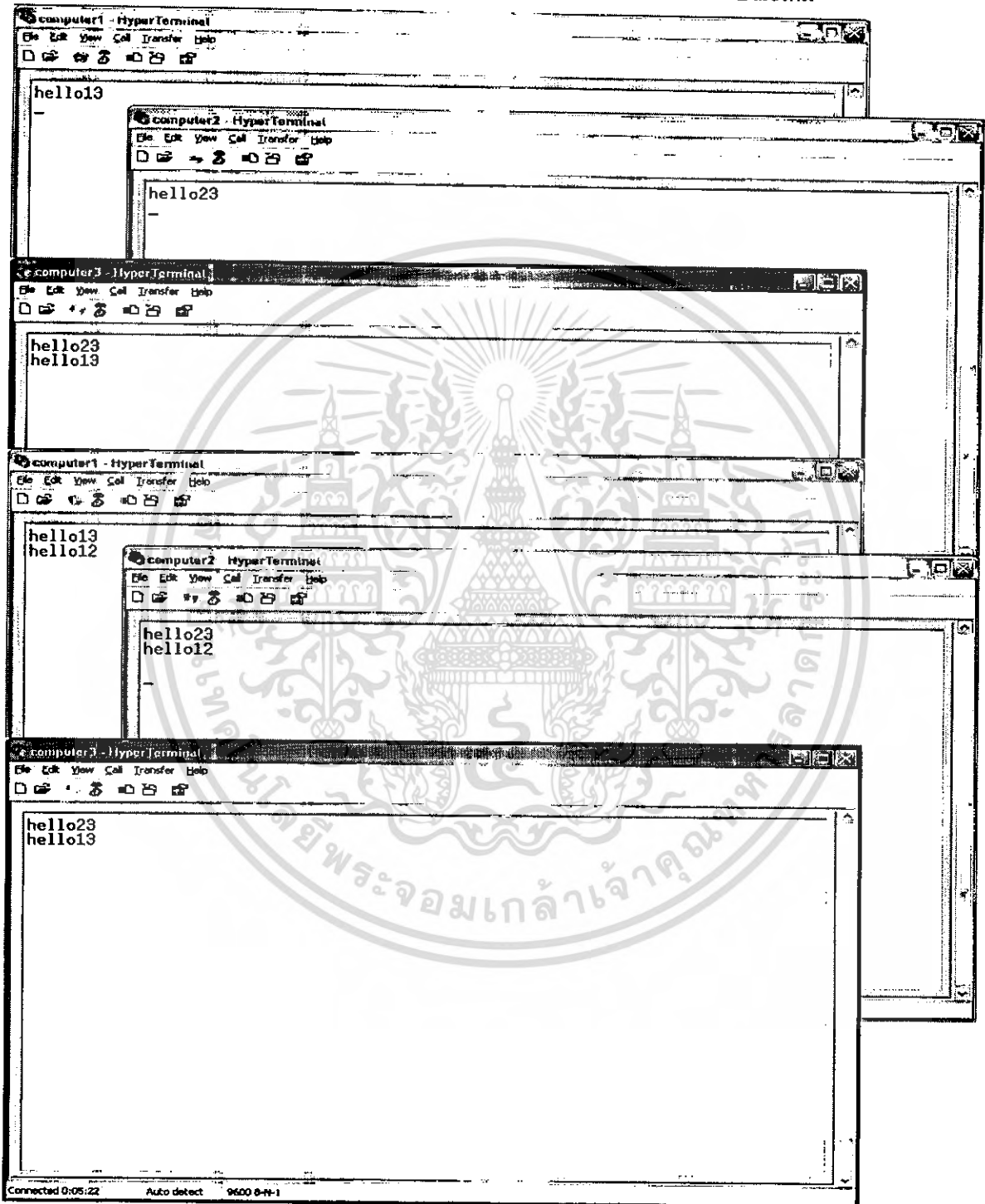
1.Computer1 ตั้ง/REและDEเป็น5Vซึ่งจะเป็นตัวส่ง Computer2และ3 ให้/REและDEเป็น0Vซึ่งจะเป็นตัวรับ ปรากฏว่า เครื่อง1 ส่งข้อมูลได้อย่างเดียว ส่วนเครื่อง2และ3รับข้อมูลได้อย่างเดียวและเป็นข้อมูลจากเครื่อง1เหมือนกัน ดังรูป



รูปที่.5.1... เมื่อให้เครื่อง1เป็นเครื่องส่ง และเครื่อง2, 3เป็นเครื่องรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

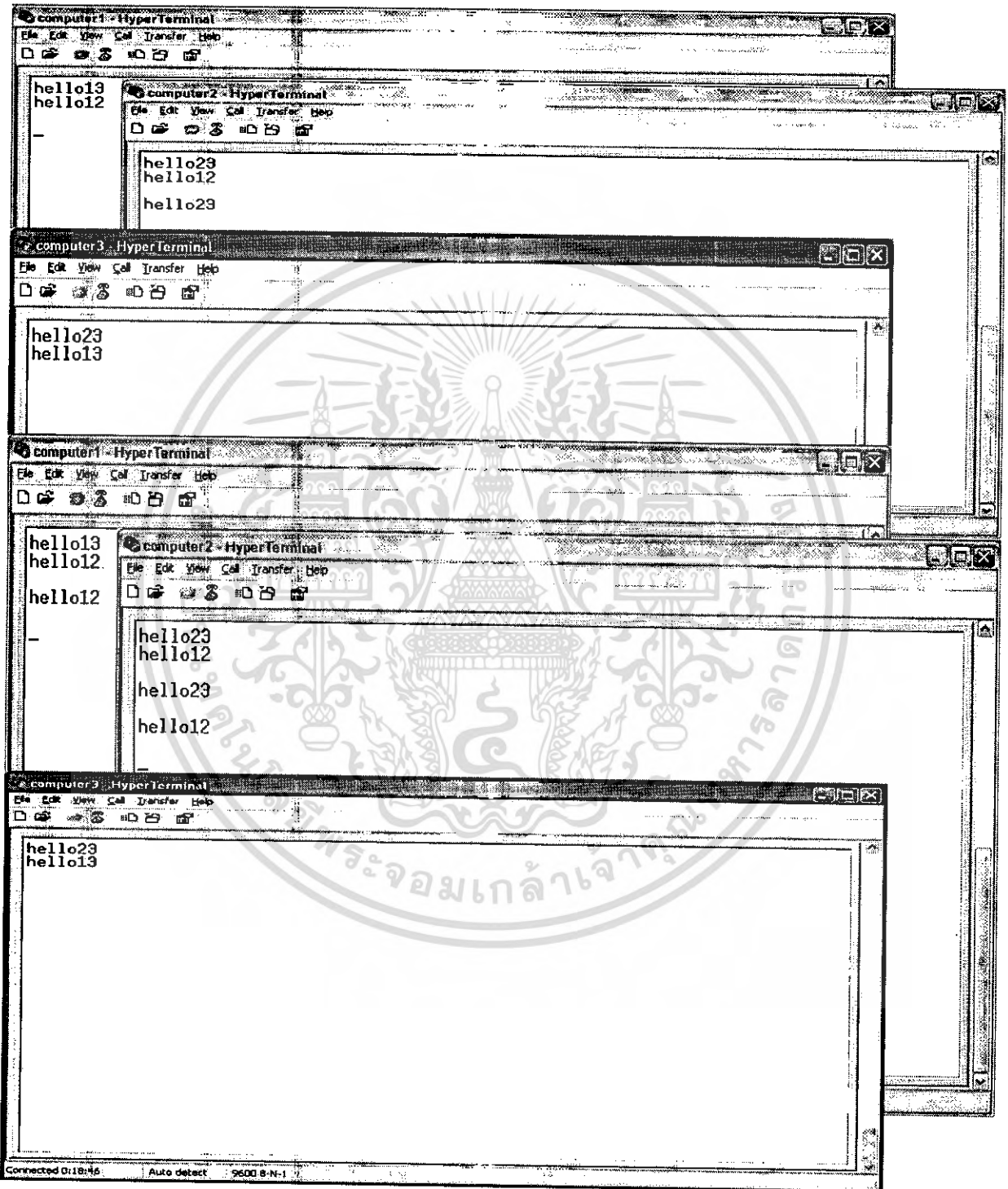
2. ทดลองเปลี่ยนให้เครื่อง 2 เป็นตัวส่ง เครื่อง 1, 3 เป็นตัวรับ ได้ผลในแนวทางเดิม คือ เครื่อง 2 ส่งได้ รับไม่ได้ เครื่อง 1, 3 รับได้ ส่งไม่ได้ ซึ่งเมื่อให้เครื่อง 3 เป็นตัวส่งก็ยังคงได้ผลในลักษณะเดิม



รูปที่.5.2... เมื่อเปลี่ยนให้เครื่อง 2 และ 3 เป็นเครื่องส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. แต่เมื่อกำหนดให้เป็นเครื่องส่งพร้อมกัน 2 เครื่อง จะมีเครื่องหนึ่งที่ไม่สามารถส่งข้อมูลได้ เสมือนถูกตัดขาดไปจากระบบ ซึ่งไม่สามารถคาดการณ์ได้ว่า ตัวใดจะถูกตัดออก



รูปที่...5.3.. เมื่อให้เครื่อง1,3เป็นเครื่องส่งพร้อมกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. และยังพบอีกว่า หากยกสายไฟที่ต่อกับขา/RE และ DE ระหว่างทำการส่งข้อมูล จะทำให้การส่งข้อมูลหยุดลงทันที นั่นหมายความว่า การสื่อสารจะถูกระงับหากขา/RE และ DE มีสถานะเป็น high impedance หรือไม่ต่อกับวงจร



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6 วิเคราะห์และสรุปผลการทดลอง

Visual Basic

ได้ทดสอบการส่งข้อมูลผ่านทางพอร์ตอนุกรมแล้วได้ผลตามที่ได้เขียนโปรแกรมไว้อย่างถูกต้อง

Microcontroller

ได้ทดสอบโดยกดปุ่มควบคุมของ Microcontroller แล้ว ผลที่ออกทาง LCD และ Relay ไม่มีการผิดเพี้ยนจากที่เขียนโปรแกรมไว้

SMS

สามารถเชื่อมข้อมูลกับพอร์ต RS 232 ได้ถูกต้อง

หนังสืออ้างอิง

1. ชีรบูลย์ หล่อวิเชียรรุ่ง , นคร ภัคดีชาติ , ชัยวัฒน์ ลีัมพรจิตวิไล , “ ปฏิบัติการไมโครคอนโทรลเลอร์ MCS 51 ด้วยโปรแกรมภาษา C ” , Innovative Experiment
2. วรพจน์ กรแก้ววัฒนากุล , ชัยวัฒน์ ลีัมพรจิตวิไล , “ เรียนรู้และปฏิบัติการไมโครคอนโทรลเลอร์ MCS 51 ” , ฉบับ AT89C5x/AT89Sxxx , Innovative Experiment
3. ศิวพงศ์ เพ็ชรสงค์ , “ ถอดรหัส SMS เรื่องลับที่ไม่ลับอีกต่อไป PC MAGAZINE กันยายน 2546 หน้า 176-181 ”
4. ปริมาภร์ เนตรวิกรม , “ เครื่องส่ง SMS อัดโน้มนัดผ่านมือถือ ” , เซมิคอนดักเตอร์ ฉบับ 246 , กรกฎาคม 2547
5. www.alldatashcet.com
6. www.google.com



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

โค้ดโปรแกรมภาษาซี ที่ใช้กับ AT89S52

โค้ดในส่วนของ Master

```
#include<reg51.h>
#include<intrins.h>

sfr data_bus=0x80;
sbit rs=P3^6;
sbit e=P3^7;
sbit control=P3^5;
unsigned char menu1=0,
menu2=0,set_serial=0x00;

sbit input11=P1^0;
sbit input12=P1^1;
sbit input13=P1^2;
sbit input14=P1^3;
sbit input21=P1^4;
sbit input22=P1^5;
sbit input23=P1^6;
sbit input24=P1^7;
sbit voice=P2^7;

bit output11;
bit output12;
bit output13;
bit output14;
bit set_door1;
bit set_bright1;

bit output21;
bit output22;
bit output23;
bit output24;
bit set_door2;
bit set_bright2;

void initlo(void);
void delay(unsigned char k);
static void enablelcd(int t);
void clearscreen(void);
void putchar(char c);
void showlcd(void);
void setcon(char c);

void button_up1(void);
void button_on1(void);
void button_down1(void);
void button_off1(void);
void showmenu1(void);
void ck_button1(void);

void button_up2(void);
void button_on2(void);
void button_down2(void);
void button_off2(void);
void showmenu2(void);
void ck_button2(void);

void ck_serial2(void);
void ck_serial1(void);
void con_serial(void);

void sevice_serial () interrupt 4
{
    if(RI==1)
    {
        RI=0;
        set_serial=SBUF;
        ck_serial1();
        ck_serial2();
    }
    if(TI==1)
    {
        TI=0;
        control = 0;
    }
}

void con_serial(void)
{
    TMOD = 0x21;
    PCON = 0x00;
    SCON = 0x50;
    EA=1;
    ES=1;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    TH1 = 0xFD;
    TL1 = 0xFD;
    TR1 = 1;
}

void ck_serial1(void)
{
    if(set_serial==0x21)
    {
        output11=0;
        menu1=1;
        showmenu();
    }
    if(set_serial==0x22)
    {
        output11=1;
        menu1=1;
        showmenu();
    }
    if(set_serial==0x23)
    {
        output12=0;
        menu1=2;
        showmenu();
    }
    if(set_serial==0x24)
    {
        output12=1;
        menu1=2;
        showmenu();
    }
    if(set_serial==0x25)
    {
        output13=0;
        menu1=3;
        showmenu();
    }
    if(set_serial==0x26)
    {
        output13=1;
        menu1=3;
        showmenu();
    }
    if(set_serial==0x27)
    {
        if(set_bright1==0)
        {
            output14=0;
            menu1=4;
            showmenu();
        }
    }
    if(set_serial==0x28)
    {
        if(set_bright1==0)
        {
            output14=1;
            menu1=4;
            showmenu();
        }
    }
    if(set_serial==0x29)
    {
        set_bright1=0;
        menu1=5;
        showmenu();
    }
    if(set_serial==0x2A)
    {
        set_bright1=1;
        menu1=5;
        showmenu();
    }
    if(set_serial==0x2B)
    {
        set_door1=0;
        menu1=6;
        showmenu();
    }
    if(set_serial==0x2C)
    {
        set_door1=1;
        menu1=6;
        showmenu();
    }
    if(set_serial==0x2D)
    {
        voice=1;
    }
    if(set_serial==0x2E)
    {
        voice=0;
    }
    if(set_serial==0x2F)
    {
        voice=1;
    }
    if(set_serial==0x30)
    {
        voice=0;
    }
}

void ck_serial2(void)
{
    if(set_serial==0x41)
    {
        output21=0;
        menu2=1;
        showmenu2();
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(set_serial==0x42)
{
    output21=1;
    menu2=1;
    showmenu2();
}
if(set_serial==0x43)
{
    output22=0;
    menu2=2;
    showmenu2();
}
if(set_serial==0x44)
{
    output22=1;
    menu2=2;
    showmenu2();
}
if(set_serial==0x45)
{
    output23=0;
    menu2=3;
    showmenu2();
}
if(set_serial==0x46)
{
    output23=1;
    menu2=3;
    showmenu2();
}
if(set_serial==0x47)
{
    if(set_bright2==0)
    {
        output24=0;
        menu2=4;
        showmenu2();
    }
}
if(set_serial==0x48)
{
    if(set_bright2==0)
    {
        output24=1;
        menu2=4;
        showmenu2();
    }
}
if(set_serial==0x49)
{
    set_bright2=0;
    menu2=5;
    showmenu2();
}
if(set_serial==0x4A)
{
    set_bright2=1;
    menu2=5;
    showmenu2();
}
if(set_serial==0x4B)
{
    set_door2=0;
    menu2=6;
    showmenu1();
}
if(set_serial==0x4C)
{
    set_door2=1;
    menu2=6;
    showmenu2();
}
if(set_serial==0x4D)
{
    voice=1;
}
if(set_serial==0x4E)
{
    voice=0;
}
if(set_serial==0x4F)
{
    voice=1;
}
if(set_serial==0x50)
{
    voice=0;
}
}
void begin(void)
{
    input11 = 1;
    input12 = 1;
    input13 = 1;
    input14 = 1;
    input21 = 1;
    input22 = 1;
    input23 = 1;
    input24 = 1;
    output11=0;
    output12=0;
    output13=0;
    output14=0;
    set_door1=0;
    set_bright1=0;
    output21=0;
    output22=0;
    output23=0;
    output24=0;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

set_door2=0;
set_bright2=0;
voice = 1;
menu1=1;
menu2=1;
}

void main()
{
con_serial();
begin();
initlo();
showmenu1();
showmenu2();
control = 1;
SBUF=0x20;
delay(1000);
control = 1;
SBUF=0x40;
delay(1000);

while(1)
{
ck_button1();
ck_button2();
}

void delay(unsigned char k)
{
unsigned char i,j;
for(i=0;i<k;i++)
for(j=0;j<200;j++);
}

static void enablelcd(int t)
{
unsigned char i;
e=1;
for(i=0;i<t;i++)i=i;
e=0;
for(i=0;i<t;i++)i=i;
}

void initlo(void)
{
rs=0;
data_bus=0x38;enablelcd(255);
;
data_bus=0x38;enablelcd(255);
;
data_bus=0x38;enablelcd(255);
;
}

data_bus=0x0E;enablelcd(255);
;
data_bus=0x04;enablelcd(255);
;
data_bus=0x01;enablelcd(255);
rs=1;
}

void clearscreen(void)
{
rs=0;
data_bus=0x01;enablelcd(255);
;
rs=1;
}

void setcon(char c)
{
rs=0;
data_bus=c;enablelcd(255);
rs=1;
}

void putchar(char c)
{
data_bus=c;
enablelcd(45);
}

void button_up1(void)
{
if (menu1 < 6)
{
menu1 = menu1+1;
showmenu1();
delay(1000);
}
}

void button_up2(void)
{
if (menu2 < 6)
{
menu2 = menu2+1;
showmenu2();
delay(1000);
}
}

void button_down1(void)
{
if (menu1 > 1)
{

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        menu1 = menu1-1;
        showmenu1();
        delay(1000);
    }
}
void button_down2(void)
{
    if (menu2 > 1)
    {
        menu2 = menu2-1;
        showmenu2();
        delay(1000);
    }
}

void button_on1(void)
{
    if (menu1 == 1)
    {
        output11=1;
        showmenu1();
        control = 1;
        SBUF=0x22;
        delay(1000);
    }
    if (menu1 == 2)
    {
        output12=1;
        showmenu1();
        control = 1;
        SBUF=0x24;
        delay(1000);
    }
    if (menu1 == 3)
    {
        output13=1;
        showmenu1();
        control = 1;
        SBUF=0x26;
        delay(1000);
    }
    if (menu1 == 4)
    {
        output14=1;
        showmenu1();
        control = 1;
        SBUF=0x28;
        delay(1000);
    }
}

    if (menu1 == 5)
    {
        set_bright1=1;
        showmenu1();
        control = 1;
        SBUF=0x2A;
        delay(1000);
    }
    if(menu1 == 6)
    {
        set_door1=1;
        showmenu1();
        control = 1;
        SBUF=0x2C;
        delay(1000);
    }
}

void button_on2(void)
{
    if (menu2 == 1)
    {
        output21=1;
        showmenu2();
        control = 1;
        SBUF=0x42;
        delay(1000);
    }
    if (menu2 == 2)
    {
        output22=1;
        showmenu2();
        control = 1;
        SBUF=0x44;
        delay(1000);
    }
    if (menu2 == 3)
    {
        output23=1;
        showmenu2();
        control = 1;
        SBUF=0x46;
        delay(1000);
    }
    if (menu2 == 4)
    {
        output24=1;
        showmenu2();
        control = 1;
        SBUF=0x48;
        delay(1000);
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    }
    if (menu2 == 5)
    {
        set_bright2=1;
        showmenu2();
        control = 1;
        SBUF=0x4A;
        delay(1000);
    }
    if(menu2 == 6)
    {
        set_door2=1;
        showmenu2();
        control = 1;
        SBUF=0x4C;
        delay(1000);
    }
}

void button_off1(void)
{
    if (menu1 == 1)
    {
        output11=0;
        showmenu1();
        control = 1;
        SBUF=0x21;
        delay(1000);
    }
    if (menu1 == 2)
    {
        output12=0;
        showmenu1();
        control = 1;
        SBUF=0x23;
        delay(1000);
    }
    if (menu1 == 3)
    {
        output13=0;
        showmenu1();
        control = 1;
        SBUF=0x25;
        delay(1000);
    }
    if (menu1 == 4)
    {
        output14=0;
        showmenu1();
        control = 1;
        SBUF=0x27;
        delay(1000);
    }
}

    }
    if (menu1 == 5)
    {
        set_bright1=0;
        showmenu1();
        control = 1;
        SBUF=0x29;
        delay(1000);
    }
    if(menu1 == 6)
    {
        set_door1=0;
        showmenu1();
        control = 1;
        SBUF=0x2B;
        delay(1000);
    }
}

void button_off2(void)
{
    if (menu2 == 1)
    {
        output21=0;
        showmenu2();
        control = 1;
        SBUF=0x41;
        delay(1000);
    }
    if (menu2 == 2)
    {
        output22=0;
        showmenu2();
        control = 1;
        SBUF=0x43;
        delay(1000);
    }
    if (menu2 == 3)
    {
        output23=0;
        showmenu2();
        control = 1;
        SBUF=0x45;
        delay(1000);
    }
    if (menu2 == 4)
    {
        output24=0;
        showmenu2();
        control = 1;
        SBUF=0x47;
        delay(1000);
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        button_down2();
    }

    if (menu2 == 5)
    {
        set_bright2=0;
        showmenu2();
        control = 1;
        SBUF=0x49;
        delay(1000);
    }

    if(menu2 == 6)
    {
        set_door2=0;
        showmenu2();
        control = 1;
        SBUF=0x4B;
        delay(1000);
    }
}

void ck_button1(void)
{
    if(input11==0)
    {
        button_up1();
    }

    if(input12==0)
    {
        button_down1();
    }

    if(input13==0)
    {
        button_on1();
    }

    if(input14==0)
    {
        button_off1();
    }
}

void ck_button2(void)
{
    if(input21==0)
    {
        button_up2();
    }

    if(input22==0)
    {
        button_down2();
    }

    if(input23==0)
    {
        button_on2();
    }

    if(input24==0)
    {
        button_off2();
    }
}

void showmenu1(void)
{
    if (menu1==1)
    {
        setcon(0x80);
        putchar('1');
        putchar('.');
        putchar('1');
        putchar(' ');
        putchar('L');
        putchar('I');
        putchar('G');
        putchar('H');
        putchar('T');
        putchar('1');
        putchar(' ');
        putchar(' ');
        putchar(' ');

        if (output11==1)
        {
            putchar('O');
            putchar('N');
            putchar(' ');
        }

        if (output11==0)
        {
            putchar('O');
            putchar('F');
            putchar('F');
        }
    }

    if (menu1==2)
    {
        setcon(0x80);
        putchar('1');
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

        putchar('4');
        putchar(' ');
        putchar('L');
        putchar('I');
        putchar('G');
        putchar('H');
        putchar('T');
        putchar('4');

if (output22==1)
{
    putchar('O');
    putchar('N');
    putchar(' ');
}
if (output22==0)
{
    putchar('O');
    putchar('F');
    putchar('F');
}
}
if (menu2==3)
{
    setcon(0xc0);
    putchar('2');
    putchar('.');
    putchar('3');
    putchar(' ');
    putchar('L');
    putchar('I');
    putchar('G');
    putchar('H');
    putchar('T');
    putchar('3');
    putchar(' ');
    putchar(' ');
    putchar(' ');
    putchar(' ');

    if (output23==1)
    {
        putchar('O');
        putchar('N');
        putchar(' ');
    }
    if (output23==0)
    {
        putchar('O');
        putchar('F');
        putchar('F');
    }
}
if (menu2==4)
{
    setcon(0xc0);
    putchar('2');
    putchar('.');

    if (set_bright2==1)
    {
        putchar(' ');
    }
    if (set_bright2==1)
    {
        putchar('s');
    }
    putchar(' ');
    putchar(' ');
    if (output24==1)
    {
        putchar('O');
        putchar('N');
        putchar(' ');
    }
    if (output24==0)
    {
        putchar('O');
        putchar('F');
        putchar('F');
    }
}
    if (menu2==5)
    {
        setcon(0xc0);
        putchar('2');
        putchar('.');
        putchar('5');
        putchar(' ');
        putchar('S');
        putchar('e');
        putchar('n');
        putchar('s');
        putchar('o');
        putchar('r');
        putchar(' ');
        putchar(' ');
        putchar(' ');
    }
    if (set_bright2==1)
    {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Codeในส่วนของ Slave

```
#include<reg51.h>
#include<intrins.h>
#include<stdio.h>

sbit onewire=P1^7;
sbit control = P2^0;
sfr data_bus=0x80;
sbit rs=P3^6;
sbit e=P3^7;

sbit input1=P1^0;
sbit input2=P1^1;
sbit input3=P1^2;
sbit input4=P1^3;

sbit output1=P2^4;
sbit output2=P2^5;
sbit output3=P2^6;
sbit output4=P2^7;

sbit voice=P1^5;
sbit bright=P1^6;
sbit door=P1^4;

bit set_door;
bit set_bright;
bit temp_over;
bit op_door;
bit save_op4;
bit over_door;
bit over_temp;
bit backup_temp;
bit backup_door;
unsigned char
temp=0x30,s3=0x30,s2=0x30,s1=0x30,
s0=0x30,menu=0,set_serial=0x00;

unsigned int i;

void ds1820_reset();
void ds1820_ans(void);
bit readbit(void);
unsigned char ds1820_read();
void ds1820_write(unsigned char
com);
void delay_convert(unsigned int
count);
void r_temp();

void initlo(void);
void delay(unsigned char k);
static void enablelcd(int t);
void clearscreen(void);
void putchar0(char c);
void showlcd(void);
void setcon(char c);

void button_up(void);
void button_on(void);
void button_off(void);
void button_down(void);
void showmenu(void);
void ck_button(void);
void begin(void);
void ck_bright(void);
void ck_door(void);

void ck_serial(void);
void returndata(void);

void con_serial(void)
{
    TMOD = 0x21;
    PCON = 0x00;
    SCON = 0x50;
    EA=1;
    ES=1;
    TH1 = 0xFD;
    TL1 = 0xFD;
    TR1 = 1;
}
void sevice_serial () interrupt 4
{
    if(RI==1)
    {
        RI=0;
        set_serial=SBUF;
        ck_serial();
    }
    if(TI==1)
    {
        TI=0;
        control = 0;
    }
}
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

void ck_serial(void)
{
    if(set_serial==0x40)
    {
        returndata();
    }
    if(set_serial==0x41)
    {
        output1=0;
        menu=1;
        showmenu();
    }
    if(set_serial==0x42)
    {
        output1=1;
        menu=1;
        showmenu();
    }
    if(set_serial==0x43)
    {
        output2=0;
        menu=2;
        showmenu();
    }
    if(set_serial==0x44)
    {
        output2=1;
        menu=2;
        showmenu();
    }
    if(set_serial==0x45)
    {
        output3=0;
        menu=3;
        showmenu();
    }
    if(set_serial==0x46)
    {
        output3=1;
        menu=3;
        showmenu();
    }
    if(set_serial==0x47)
    {
        if(set_bright==0)
        {
            output4=0;
            menu=4;
            showmenu();
        }
    }
    if(set_serial==0x48)
    {
        if(set_bright==0)
        {
            output4=1;
            menu=4;
        }
    }
}

showmenu();
}
if(set_serial==0x49)
{
    set_bright=0;
    menu=5;
    showmenu();
}
if(set_serial==0x4A)
{
    set_bright=1;
    menu=5;
    showmenu();
}
if(set_serial==0x4B)
{
    set_door=0;
    menu=6;
    showmenu();
}
if(set_serial==0x4C)
{
    set_door=1;
    menu=6;
    showmenu();
}
}

void begin(void)
{
    menu = 1;
    showmenu();

    input1 = 1;
    input2 = 1;
    input3 = 1;
    input4 = 1;

    voice = 1;
    bright=1;
    door = 1;
    menu = 1;

    output1=0;
    output2=0;
    output3=0;
    output4=0;
    set_door=0;
    set_bright=0;
    over_door=0;
    over_temp=0;
    backup_temp=0;
    control = 0;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

void main()
{
    begin();
    con_serial();
    initlo();
    showlcd();
    showmenu();
    returndata();

    while(1)
    {
        r_temp();
        ck_button();
        ck_bright();
        ck_door();
    }
}

void ds1820_reset()
{
    onewire=0;
    for(i=0;i<100;i++)
        _nop_();

    onewire=1;
    for(i=0;i<2;i++)
        _nop_();
}

void ds1820_ans(void)
{
    while(onewire);
    while(~onewire);
    for(i=0;i<2;i++)
        _nop_();
}

bit readbit(void)
{
    bit dat;
    onewire=0;
    _nop_();
    _nop_();

    onewire=1;
    _nop_();
    _nop_();
    _nop_();
    _nop_();

    dat=onewire;

    for(i=0;i<8;i++)
        _nop_();
    return(dat);
}

unsigned char ds1820_read()
{
    unsigned char i,j,dat;
    dat=0;
    for(i=0;i<8;i++)
    {
        j=readbit();
        dat=(j<<7)|(dat>>1);
    }
    return(dat);
}

void ds1820_write(unsigned char com)
{
    unsigned char j;
    bit send;
    for (j=0;j<8;j++)
    {
        send=com&0x01;
        com=com>>1;
        if(send)
        {
            onewire = 0;
            _nop_();
            _nop_();
            _nop_();
            onewire = 1;
            for(i=0;i<8;i++)
                _nop_();
        }
        else
        {
            onewire=0;
            for(i=0;i<8;i++)
                _nop_();
            onewire=1;
            _nop_();
            _nop_();
            _nop_();
            _nop_();
        }
    }
}

void delay_convert(unsigned int count)
{
    do
    {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        count--;
    }
    while(count>0);
}

void r_temp()
{
    ds1820_reset();
    ds1820_ans();
    ds1820_write(0xCC);
    ds1820_write(0x44);
    delay_convert(10000);
    ds1820_reset();
    ds1820_ans();
    ds1820_write(0xCC);
    ds1820_write(0xBE);
    temp=ds1820_read();

    s0=temp&0x01;
    if(s0) s0=0x35;
    else s0=0x30;

    temp=temp>>1;
    temp=temp&0x7F;
    s2=temp/10|0x30;
    s1=temp%10|0x30;

    if(s2 > 0x33)
    {
        over_temp=1;

        voice!=(over_temp |
over_door);
    }
    else
    {
        over_temp=0;

        voice!=(over_temp |
over_door);
    }

    if(!(over_temp ==
backup_temp))
    {
        if(over_temp==1)
        {
            control = 1;
            SBUF = 0x4E;
        }
        if(over_temp==0)
        {
            control = 1;
            SBUF = 0x4D;
        }
    }

    backup_temp =
over_temp;

    setcon(0x8C);
    putchar0(s2);
    setcon(0x8D);
    putchar0(s1);
    setcon(0x8F);
    putchar0(s0);
}

void delay(unsigned char k)
{
    unsigned char i,j;
    for(i=0;i<k;i++)
        for(j=0;j<200;j++);
}

static void enablelcd(int t)
{
    unsigned char i;
    e=1;
    for(i=0;i<t;i++)i=i;
    e=0;
    for(i=0;i<t;i++)i=i;
}

void initlo(void)
{
    rs=0;
    data_bus=0x38;enablelcd(255);
    ;
    data_bus=0x38;enablelcd(255);
    ;
    data_bus=0x38;enablelcd(255);
    ;
    data_bus=0x01;enablelcd(255);
    ;
    data_bus=0x0E;enablelcd(255);
    ;
    data_bus=0x04;enablelcd(255);
    ;
    rs=1;
}

void clearscreen(void)
{
    rs=0;
    data_bus=0x01;enablelcd(255);
    ;
    rs=1;
}

void setcon(char c)
{
    rs=0;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    data_bus=c;enablelcd(255);
    rs=1;
}
void putchar0(char c)
{
    data_bus=c;
    enablelcd(45);
}
void showlcd(void)
{
    clearsreen();
    putchar0('T');
    putchar0('e');
    putchar0('m');
    putchar0('p');
    putchar0('e');
    putchar0('r');
    putchar0('a');
    putchar0('t');
    putchar0('u');
    putchar0('r');
    putchar0('e');
    putchar0(' ');
    putchar0(s2);
    putchar0(s1);
    putchar0('.');
    putchar0(s0);
}
void ck_button(void)
{
    if(input1==0)
    {
        button_up();
    }
    if(input2==0)
    {
        button_down();
    }
    if(input3==0)
    {
        button_on();
    }
    if(input4==0)
    {
        button_off();
    }
}
void button_up(void)
{
    if (menu < 6)
    {
        data_bus=c;enablelcd(255);
        rs=1;
        menu = menu+1;
        showmenu();
        delay(100);
    }
}
void button_down(void)
{
    if (menu > 1)
    {
        menu = menu-1;
        showmenu();
        delay(100);
    }
}
void button_on(void)
{
    if (menu == 1)
    {
        output1=1;
        showmenu();
        control = 1;
        SBUF = 0x42;
        delay(100);
    }
    if (menu == 2)
    {
        output2=1;
        showmenu();
        control = 1;
        SBUF = 0x44;
        delay(100);
    }
    if (menu == 3)
    {
        output3=1;
        showmenu();
        control = 1;
        SBUF = 0x46;
        delay(100);
    }
    if (menu == 4)
    {
        if(set_bright==0)
        {
            output4=1;
            showmenu();
            control = 1;
            SBUF = 0x48;
            delay(100);
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}
if (menu == 5)
{
    if(set_bright==0)
    {
        set_bright=1;
        showmenu();
        control = 1;
        SBUF = 0x4A;
        save_op4=output4;
        showmenu();
        delay(100);
    }
}

if(menu == 6)
{
    set_door=1;
    showmenu();
    control = 1;
    SBUF = 0x4C;
    over_door=0;
    backup_door=0;
    delay(100);
    control = 1;
    SBUF = 0x4E;
}

void button_off(void)
{
    if (menu == 1)
    {
        output1=0;
        showmenu();
        control = 1;
        SBUF = 0x41;
        delay(100);
    }
    if (menu == 2)
    {
        output2=0;
        showmenu();
        control = 1;
        SBUF = 0x43;
        delay(100);
    }
    if (menu == 3)
    {
        output3=0;
        control = 1;
    }
}

SBUF = 0x45;
showmenu();
delay(100);
}
if (menu == 4)
{
    if(set_bright==0)
    {
        output4=0;
        showmenu();
        control = 1;
        SBUF = 0x47;
        delay(100);
    }
}

if (menu == 5)
{
    if (set_bright==1)
    {
        set_bright=0;
        control = 1;
        SBUF = 0x49;
        output4=save_op4;
        showmenu();
        delay(100);
    }
}
if(menu == 6)
{
    set_door=0;
    showmenu();
    over_door=0;
    backup_door=0;
    control = 1;
    SBUF = 0x4B;
    delay(100);
    control = 1;
    SBUF = 0x4F;
}
}

void showmenu(void)
{
    if (menu==1)
    {
        setcon(0xc0);
        putchar('1');
        putchar('.');
        putchar('L');
        putchar('I');
        putchar('G');
        putchar('H');
        putchar('T');
        putchar('1');
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

putchar0(' ');
putchar0(' ');
putchar0(' ');
putchar0(' ');
putchar0(' ');

if (output1==1)
{
    putchar0('O');
    putchar0('N');
    putchar0(' ');
}
if (output1==0)
{
    putchar0('O');
    putchar0('F');
    putchar0('F');
}
}

if (menu==2)
{
    setcon(0xc0);
    putchar0('2');
    putchar0('.');
    putchar0('L');
    putchar0('I');
    putchar0('G');
    putchar0('H');
    putchar0('T');
    putchar0('2');
    putchar0(' ');
    putchar0(' ');
    putchar0(' ');
    putchar0(' ');
    putchar0(' ');

    if (output2==1)
    {
        putchar0('O');
        putchar0('N');
        putchar0(' ');
    }
    if (output2==0)
    {
        putchar0('O');
        putchar0('F');
        putchar0('F');
    }
}

if (menu==3)
{
    setcon(0xc0);
    putchar0('3');
    putchar0('.');
    putchar0('L');
    putchar0('I');
    putchar0('G');
    putchar0('H');
    putchar0('T');
    putchar0('3');
    putchar0(' ');
    putchar0(' ');
    putchar0(' ');
    putchar0(' ');

    if (output3==1)
    {
        putchar0('O');
        putchar0('N');
        putchar0(' ');
    }
    if (output3==0)
    {
        putchar0('O');
        putchar0('F');
        putchar0('F');
    }

    if (menu==4)
    {
        setcon(0xc0);
        putchar0('4');
        putchar0('.');
        putchar0('L');
        putchar0('I');
        putchar0('G');
        putchar0('H');
        putchar0('T');
        putchar0('4');
        putchar0(' ');
        putchar0(' ');

        if (set_bright==0)
        {
            putchar0(' ');
        }
        if (set_bright==1)
        {
            putchar0('s');
        }
        putchar0(' ');
        putchar0(' ');
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

void ck_bright(void)
{
    if (set_bright==1)
    {
        if (bright==1)
        {
            output4=0;
        }
        if (bright==0)
        {
            output4=1;
        }
    }
}

void ck_door(void)
{
    if(set_door==1)
    {
        if(door==0)
        {
            over_door=1;
            voice=(over_door |
over_temp);
        }
        if(!(over_door ==
backup_door))
        {
            control = 1;
            SBUF=0x50;
        }
        backup_door =
over_door;
    }
}

void returndata(void)
{
    if(output1 == 0)
    {
        control = 1;
        SBUF=0x41;
        delay(100);
    }
    if(output1 == 1)
    {
        control = 1;
        SBUF=0x42;
        delay(100);
    }
    if(output2 == 0)
    {
        control = 1;
        SBUF=0x43;
        delay(100);
    }
    if(output2 == 1)
    {
        control = 1;
        SBUF=0x44;
        delay(100);
    }
    if(output3 == 0)
    {
        control = 1;
        SBUF=0x45;
        delay(100);
    }
    if(output3 == 1)
    {
        control = 1;
        SBUF=0x46;
        delay(100);
    }
    if (set_bright==0)
    {
        if(output4 == 0)
        {
            control = 1;
            SBUF=0x47;
            delay(100);
        }
        if(output4 == 1)
        {
            control = 1;
            SBUF=0x48;
            delay(100);
        }
    }
    if (set_bright==1)
    {
        if(save_op4 == 0)
        {
            control = 1;
            SBUF=0x47;
            delay(100);
        }
        if(save_op4 == 1)
        {
            control = 1;
            SBUF=0x48;
            delay(100);
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

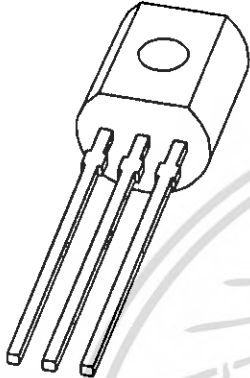
    }
    delay(100);
}
if(set_bright == 0)
{
    control = 1;
    SBUF=0x49;
    delay(100);
}
if(set_bright == 1)
{
    control = 1;
    SBUF=0x4A;
    delay(100);
}
if(set_door == 0)
{
    control = 1;
    SBUF=0x4B;

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DATA SHEET



BC556; BC557 PNP general purpose transistors

Product specification
Supersedes data of 1997 Mar 27

1999 Apr 15

Philips
Semiconductors



PHILIPS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PNP general purpose transistors

BC556; BC557

FEATURES

- Low current (max. 100 mA)
- Low voltage (max. 65 V).

APPLICATIONS

- General purpose switching and amplification.

DESCRIPTION

PNP transistor in a TO-92; SOT54 plastic package.
NPN complements: BC546 and BC547.

PINNING

PIN	DESCRIPTION
1	emitter
2	base
3	collector

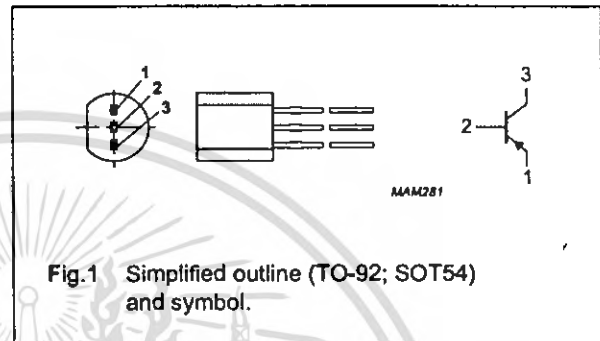


Fig.1 Simplified outline (TO-92; SOT54) and symbol.

LIMITING VALUES

In accordance with the Absolute Maximum Rating System (IEC 134).

SYMBOL	PARAMETER	CONDITIONS	MIN.	MAX.	UNIT
V _{CB0}	collector-base voltage	open emitter	-	-80	V
	BC556		-	-50	V
V _{CE0}	collector-emitter voltage	open base	-	-65	V
	BC557		-	-45	V
V _{EB0}	emitter-base voltage	open collector	-	-5	V
I _C	collector current (DC)		-	-100	mA
I _{CM}	peak collector current		-	-200	mA
I _{BM}	peak base current		-	-200	mA
P _{tot}	total power dissipation	T _{amb} ≤ 25 °C	-	500	mW
T _{stg}	storage temperature		-65	+150	°C
T _J	junction temperature		-	150	°C
T _{amb}	operating ambient temperature		-65	+150	°C

PNP general purpose transistors

BC556; BC557

THERMAL CHARACTERISTICS

SYMBOL	PARAMETER	CONDITIONS	VALUE	UNIT
$R_{th\ j-a}$	thermal resistance from junction to ambient	note 1	250	K/W

Note

1. Transistor mounted on an FR4 printed-circuit board.

CHARACTERISTICS

$T_j = 25\text{ }^\circ\text{C}$ unless otherwise specified.

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
I_{CBO}	collector cut-off current	$I_E = 0; V_{CB} = -30\text{ V}$	-	-1	-15	nA
		$I_E = 0; V_{CB} = -30\text{ V}; T_j = 150\text{ }^\circ\text{C}$	-	-	-4	μA
I_{EBO}	emitter cut-off current	$I_C = 0; V_{EB} = -5\text{ V}$	-	-	-100	nA
h_{FE}	DC current gain BC556 BC557 BC556A BC556B; BC557B BC557C	$I_C = -2\text{ mA}; V_{CE} = -5\text{ V};$ see Figs 2, 3 and 4	125	-	475	
			125	-	800	
			125	-	250	
			220	-	475	
			420	-	800	
V_{CEsat}	collector-emitter saturation voltage	$I_C = -10\text{ mA}; I_B = -0.5\text{ mA}$	-	-60	-300	mV
		$I_C = -100\text{ mA}; I_B = -5\text{ mA}$	-	-180	-650	mV
V_{BEsat}	base-emitter saturation voltage	$I_C = -10\text{ mA}; I_B = -0.5\text{ mA};$ note 1	-	-750	-	mV
		$I_C = -100\text{ mA}; I_B = -5\text{ mA};$ note 1	-	-930	-	mV
V_{BE}	base-emitter voltage	$I_C = -2\text{ mA}; V_{CE} = -5\text{ V};$ note 2	-600	-650	-750	mV
		$I_C = -10\text{ mA}; V_{CE} = -5\text{ V};$ note 2	-	-	-820	mV
C_c	collector capacitance	$I_E = I_C = 0; V_{CB} = -10\text{ V}; f = 1\text{ MHz}$	-	3	-	pF
C_e	emitter capacitance	$I_C = I_E = 0; V_{EB} = -0.5\text{ V}; f = 1\text{ MHz}$	-	10	-	pF
f_T	transition frequency	$I_C = -10\text{ mA}; V_{CE} = -5\text{ V}; f = 100\text{ MHz}$	100	-	-	MHz
F	noise figure	$I_C = -200\text{ }\mu\text{A}; V_{CE} = -5\text{ V}; R_S = 2\text{ k}\Omega;$ $f = 1\text{ kHz}; B = 200\text{ Hz}$	-	2	10	dB

Notes

1. V_{BEsat} decreases by about -1.7 mV/K with increasing temperature.
2. V_{BE} decreases by about -2 mV/K with increasing temperature.

Features

- Compatible with MCS™-51 Products
- 12K Bytes of In-System Programmable (ISP) Flash Program Memory
 - SPI Serial Interface for Program Downloading
 - Endurance: 10,000 Write/Erase Cycles
- 2K Bytes EEPROM Data Memory
 - Endurance: 100,000 Write/Erase Cycles
- 64-byte User Signature Array
- 2.7V to 5.5V Operating Range
- Fully Static Operation: 0 Hz to 24 MHz
- Three-level Program Memory Lock
- 256 x 8-bit Internal RAM
- 32 Programmable I/O Lines
- Three 16-bit Timer/Counters
- Nine Interrupt Sources
- Enhanced UART Serial Port with Framing Error Detection and Automatic Address Recognition
- Enhanced SPI (Double Write/Read Buffered) Serial Interface
- Low-power Idle and Power-down Modes
- Interrupt Recovery from Power-down Mode
- Programmable Watchdog Timer
- Dual Data Pointer
- Power-off Flag
- Flexible ISP Programming (Byte and Page Modes)
 - Page Mode: 64 Bytes/Page for Code Memory, 32 Bytes/Page for Data Memory
- Four-level Enhanced Interrupt Controller
- Programmable and Fuseable x2 Clock Option
- Internal Power-on Reset
- 42-pin PDIP Package Option for Reduced EMC Emission
- Green (Pb/Halide-free) Packaging Option

1. Description

The AT89S8253 is a low-power, high-performance CMOS 8-bit microcontroller with 12K bytes of In-System Programmable (ISP) Flash program memory and 2K bytes of EEPROM data memory. The device is manufactured using Atmel's high-density non-volatile memory technology and is compatible with the industry-standard MCS-51 instruction set and pinout. The on-chip downloadable Flash allows the program memory to be reprogrammed in-system through an SPI serial interface or by a conventional nonvolatile memory programmer. By combining a versatile 8-bit CPU with downloadable Flash on a monolithic chip, the Atmel AT89S8253 is a powerful microcontroller which provides a highly-flexible and cost-effective solution to many embedded control applications.



**8-bit
Microcontroller
with 12K Bytes
Flash and 2K
Bytes EEPROM**

AT89S8253

Preliminary

3286C-MICRO-1/05



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The AT89S8253 provides the following standard features: 12K bytes of In-System Programmable Flash, 2K bytes of EEPROM, 256 bytes of RAM, 32 I/O lines, programmable watchdog timer, two data pointers, three 16-bit timer/counters, a six-vector, four-level interrupt architecture, a full duplex serial port, on-chip oscillator, and clock circuitry. In addition, the AT89S8253 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 external interrupt or hardware reset.

The on-board Flash/EEPROM is accessible through the SPI serial interface. Holding RESET active forces the SPI bus into a serial programming interface and allows the program memory to be written to or read from, unless one or more lock bits have been activated.

2. Pin Configurations

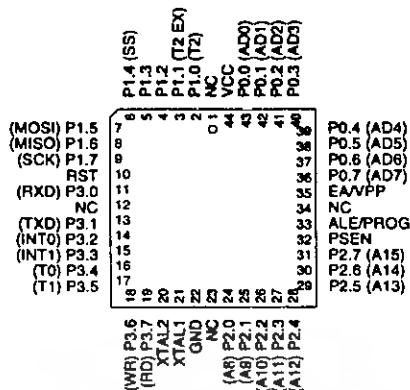
2.1 40P6 – 40-lead PDIP

(T2) P1.0	1	40	VCC
(T2 EX) P1.1	2	39	P0.0 (AD0)
P1.2	3	38	P0.1 (AD1)
P1.3	4	37	P0.2 (AD2)
(SS) P1.4	5	35	P0.3 (AD3)
(MOSI) P1.5	6	35	P0.4 (AD4)
(MISO) P1.6	7	34	P0.5 (AD5)
(SCK) P1.7	8	33	P0.6 (AD6)
RST	9	32	P0.7 (AD7)
(RXD) P3.0	10	31	EA/VPP
(TXD) P3.1	11	30	ALE/PROG
(INT0) P3.2	12	29	PSEN
(INT1) P3.3	13	28	P2.7 (A15)
(T0) P3.4	14	27	P2.6 (A14)
(T1) P3.5	15	26	P2.5 (A13)
(WR) P3.6	18	25	P2.4 (A12)
(RD) P3.7	17	24	P2.3 (A11)
XTAL2	18	23	P2.2 (A10)
XTAL1	19	22	P2.1 (A9)
GND	20	21	P2.0 (A8)

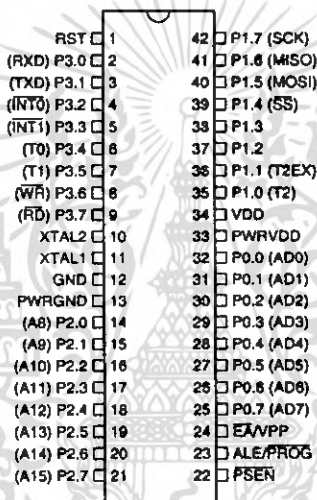
2.2 44A – 44-lead TQFP

(MOSI) P1.5	1	33	P0.4 (AD4)
(MISO) P1.6	2	32	P0.5 (AD5)
(SCK) P1.7	3	31	P0.6 (AD6)
RST	4	30	P0.7 (AD7)
(RXD) P3.0	5	29	EA/VPP
NC	6	28	NC
(TXD) P3.1	7	27	ALE/PROG
(INT0) P3.2	8	26	PSEN
(INT1) P3.3	9	25	P2.7 (A15)
(T0) P3.4	10	24	P2.6 (A14)
(T1) P3.5	11	23	P2.5 (A13)
(WR) P3.6	12	22	P2.4 (A12)
(RD) P3.7	13	21	P2.3 (A11)
XTAL2	14	20	P2.2 (A10)
XTAL1	15	19	P2.1 (A9)
GND	16	18	P2.0 (A8)
GND	17	17	P2.0 (A8)
(A8) P2.0	18	16	P2.0 (A8)
(A9) P2.1	19	15	P2.1 (A9)
(A10) P2.2	20	14	P2.2 (A10)
(A11) P2.3	21	13	P2.3 (A11)
(A12) P2.4	22	12	P2.4 (A12)
(A13) P2.5	23	11	P2.5 (A13)
(A14) P2.6	24	10	P2.6 (A14)
(A15) P2.7	25	9	P2.7 (A15)
PSEN	26	8	PSEN
ALE/PROG	27	7	ALE/PROG
NC	28	6	NC
EA/VPP	29	5	EA/VPP
P0.7 (AD7)	30	4	P0.7 (AD7)
P0.6 (AD6)	31	3	P0.6 (AD6)
P0.5 (AD5)	32	2	P0.5 (AD5)
P0.4 (AD4)	33	1	P0.4 (AD4)
P0.3 (AD3)	34	34	P0.3 (AD3)
P0.2 (AD2)	35	35	P0.2 (AD2)
P0.1 (AD1)	36	36	P0.1 (AD1)
P0.0 (AD0)	37	37	P0.0 (AD0)
VCC	38	38	VCC
NC	39	39	NC
P1.0 (T2)	40	40	P1.0 (T2)
P1.1 (T2 EX)	41	41	P1.1 (T2 EX)
P1.2	42	42	P1.2
P1.3	43	43	P1.3
(SS) P1.4	44	44	(SS) P1.4

2.3 44J – 44-lead PLCC



2.4 42PS6 – PDIP



3. Pin Description

3.1 VCC

Supply voltage (all packages except 42-PDIP).

3.2 GND

Ground (all packages except 42-PDIP; for 42-PDIP GND connects only the logic core and the embedded program/data memories).

3.3 VDD

Supply voltage for the 42-PDIP which connects only the logic core and the embedded program/data memories.

3.4 PWRVDD

Supply voltage for the 42-PDIP which connects only the I/O Pad Drivers.

The application board **must** connect both VDD and PWRVDD to the board supply voltage.

3.5 PWRGND

Ground for the 42-PDIP which connects only the I/O Pad Drivers. PWRGND and GND are weakly connected through the common silicon substrate, but not through any metal links. The application board **must** connect both GND and PWRGND to the board ground.

3.6 Port 0

Port 0 is an 8-bit open drain bi-directional I/O port. As an output port, each pin can sink six TTL inputs. When 1s are written to port 0 pins, the pins can be used as high-impedance inputs.

Port 0 can also be configured to be the multiplexed low-order address/data bus during accesses to external program and data memory. In this mode, P0 has internal pull-ups.

Port 0 also receives the code bytes during Flash programming and outputs the code bytes during program verification. **External pull-ups are required during program verification.**

3.7 Port 1

Port 1 is an 8-bit bi-directional I/O port with internal pull-ups. The Port 1 output buffers can sink/source six TTL inputs. When 1s are written to Port 1 pins, they are pulled high by the weak internal pull-ups and can be used as inputs. As inputs, Port 1 pins that are externally being pulled low will source current (I_{IL} , 150 μ A typical) because of the weak internal pull-ups.

Some Port 1 pins provide additional functions. P1.0 and P1.1 can be configured to be the timer/counter 2 external count input (P1.0/T2) and the timer/counter 2 trigger input (P1.1/T2EX), respectively.

Furthermore, P1.4, P1.5, P1.6, and P1.7 can be configured as the SPI slave port select, data input/output and shift clock input/output pins as shown in the following table.

Port Pin	Alternate Functions
P1.0	T2 (external count input to Timer/Counter 2), clock-out
P1.1	T2EX (Timer/Counter 2 capture/reload trigger and direction control)
P1.4	\overline{SS} (Slave port select input)
P1.5	MOSI (Master data output, slave data input pin for SPI channel)
P1.6	MISO (Master data input, slave data output pin for SPI channel)
P1.7	SCK (Master clock output, slave clock input pin for SPI channel)

Port 1 also receives the low-order address bytes during Flash programming and verification.

3.8 Port 2

Port 2 is an 8-bit bi-directional I/O port with internal pull-ups. The Port 2 output buffers can sink/source six TTL inputs. When 1s are written to Port 2 pins, they are pulled high by the weak internal pull-ups and can be used as inputs. As inputs, Port 2 pins that are externally being pulled low will source current (I_{IL} , 150 μ A typical) because of the weak internal pull-ups.

Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @ DPTR). In this application, Port 2 uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOVX @ RI), Port 2 emits the contents of the P2 Special Function Register.

Port 2 also receives the high-order address bits and some control signals during Flash programming and verification.

3.9 Port 3

Port 3 is an 8-bit bi-directional I/O port with internal pull-ups. The Port 3 output buffers can sink/source six TTL inputs. When 1s are written to Port 3 pins, they are pulled high by the weak internal pull-ups and can be used as inputs. As inputs, Port 3 pins that are externally being pulled low will source current (I_{IL} , 150 μ A typical) because of the weak internal pull-ups.

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

Port 3 also serves the functions of various special features of the AT89S8253, as shown in the following table.

Port Pin	Alternate Functions
P3.0	RXD (serial input port)
P3.1	TXD (serial output port)
P3.2	$\overline{INT0}$ (external interrupt 0) ⁽¹⁾
P3.3	$\overline{INT1}$ (external interrupt 1) ⁽¹⁾
P3.4	T0 (timer 0 external input)
P3.5	T1 (timer 1 external input)
P3.6	\overline{WR} (external data memory write strobe)
P3.7	\overline{RD} (external data memory read strobe)

Note: 1. All pins in ports 1 and 2 and almost all pins in port 3 (the exceptions are P3.2 $\overline{INT0}$ and P3.3 $\overline{INT1}$) have their weak internal pull-ups disabled in the Power-down mode. Port pins P3.2 ($\overline{INT0}$) and P3.1 ($\overline{INT1}$) are active even in Power-down mode (to be able to sense an interrupt request to exit the Power-down mode) and as such still have their weak internal pull-ups turned on.

3.10 RST

Reset input. A high on this pin for at least two machine cycles while the oscillator is running resets the device.

3.11 ALE/ $\overline{\text{PROG}}$

Address Latch Enable. ALE/ $\overline{\text{PROG}}$ is an output pulse for latching the low byte of the address (on its falling edge) during accesses to external memory. This pin is also the program pulse input ($\overline{\text{PROG}}$) during Flash programming.

In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency and may be used for external timing or clocking purposes. Note, however, that one ALE pulse is skipped during each access to external data memory.

If desired, ALE operation can be disabled by setting bit 0 of the AUXR SFR at location 8EH. With the bit set, ALE is active only during a MOVX or MOVC instruction. Otherwise, the pin is weakly pulled high. Setting the ALE-disable bit has no effect if the microcontroller is in external execution mode.

3.12 $\overline{\text{PSEN}}$

Program Store Enable. $\overline{\text{PSEN}}$ is the read strobe to external program memory (active low).

When the AT89S8253 is executing code from external program memory, $\overline{\text{PSEN}}$ is activated twice each machine cycle, except that two $\overline{\text{PSEN}}$ activations are skipped during each access to external data memory.

3.13 $\overline{\text{EA/VPP}}$

External Access Enable. $\overline{\text{EA}}$ must be strapped to GND in order to enable the device to fetch code from external program memory locations starting at 0000H up to FFFFH. Note, however, that if lock bit 1 is programmed, $\overline{\text{EA}}$ will be internally latched on reset.

$\overline{\text{EA}}$ should be strapped to V_{CC} for internal program executions. This pin also receives the 12-volt programming enable voltage (V_{PP}) during Flash programming when 12-volt programming is selected.

3.14 XTAL1

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

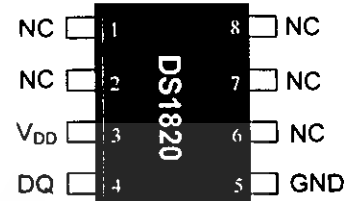
3.15 XTAL2

Output from the inverting oscillator amplifier.

FEATURES

- Unique 1-wire interface requires only one port pin for communication
- Each device has a unique 64-bit serial code stored in an on-board ROM
- Multi-drop capability simplifies distributed temperature sensing applications
- Requires no external components
- Can be powered from data line. Power supply range is 3.0V to 5.5V
- Measures temperatures from -55°C to $+125^{\circ}\text{C}$ (-67°F to $+257^{\circ}\text{F}$)
- $\pm 0.5^{\circ}\text{C}$ accuracy from -10°C to $+85^{\circ}\text{C}$
- 9-bit thermometer resolution
- Converts temperature in 750 ms (max.)
- User-definable nonvolatile alarm settings
- Alarm search command identifies and addresses devices whose temperature is outside of programmed limits (temperature alarm condition)
- Applications include thermostatic controls, industrial systems, consumer products, thermometers, or any thermally sensitive system

PIN ASSIGNMENT



8-pin 150-mil SOIC
 (DS18S20Z)

PIN DESCRIPTION

- GND - Ground
- DQ - Data In/Out
- V_{DD} - Power Supply Voltage
- NC - No Connect

DESCRIPTION

The DS18S20 Digital Thermometer provides 9-bit centigrade temperature measurements and has an alarm function with nonvolatile user-programmable upper and lower trigger points. The DS18S20 communicates over a 1-wire bus that by definition requires only one data line (and ground) for communication with a central microprocessor. It has an operating temperature range of -55°C to $+125^{\circ}\text{C}$ and is accurate to $\pm 0.5^{\circ}\text{C}$ over the range of -10°C to $+85^{\circ}\text{C}$. In addition, the DS18S20 can derive power directly from the data line ("parasite power"), eliminating the need for an external power supply.

Each DS18S20 has a unique 64-bit serial code, which allows multiple DS18S20s to function on the same 1-wire bus; thus, it is simple to use one microprocessor to control many DS18S20s distributed over a large area. Applications that can benefit from this feature include HVAC environmental controls, temperature monitoring systems inside buildings, equipment or machinery, and process monitoring and control systems.

DETAILED PIN DESCRIPTIONS Table 1

8-PIN SOIC*	TO-92	SYMBOL	DESCRIPTION
5	1	GND	Ground.
4	2	DQ	Data Input/Output pin. Open-drain 1-wire interface pin. Also provides power to the device when used in parasite power mode (see "Parasite Power" section.)
3	3	V _{DD}	Optional V_{DD} pin. V_{DD} must be grounded for operation in parasite power mode.

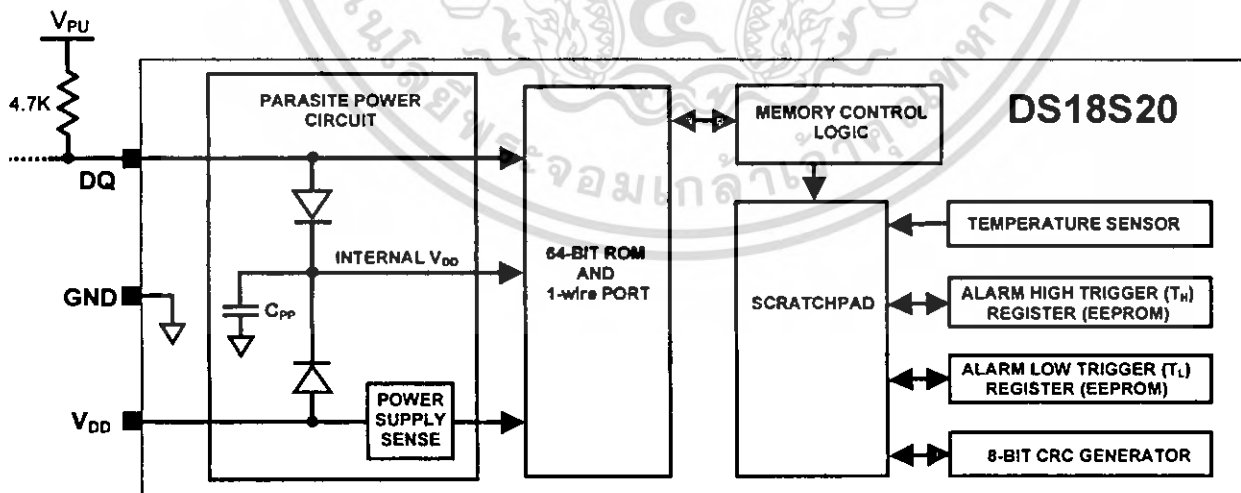
*All pins not specified in this table are "No Connect" pins.

OVERVIEW

Figure 1 shows a block diagram of the DS18S20, and pin descriptions are given in Table 1. The 64-bit ROM stores the device's unique serial code. The scratchpad memory contains the 2-byte temperature register that stores the digital output from the temperature sensor. In addition, the scratchpad provides access to the 1-byte upper and lower alarm trigger registers (T_H and T_L). The T_H and T_L registers are nonvolatile (EEPROM), so they will retain data when the device is powered down.

The DS18S20 uses Dallas' exclusive 1-wire bus protocol that implements bus communication using one control signal. The control line requires a weak pullup resistor since all devices are linked to the bus via a 3-state or open-drain port (the DQ pin in the case of the DS18S20). In this bus system, the microprocessor (the master device) identifies and addresses devices on the bus using each device's unique 64-bit code. Because each device has a unique code, the number of devices that can be addressed on one bus is virtually unlimited. The 1-wire bus protocol, including detailed explanations of the commands and "time slots," is covered in the 1-WIRE BUS SYSTEM section of this datasheet.

Another feature of the DS18S20 is the ability to operate without an external power supply. Power is instead supplied through the 1-wire pullup resistor via the DQ pin when the bus is high. The high bus signal also charges an internal capacitor (C_{PP}), which then supplies power to the device when the bus is low. This method of deriving power from the 1-wire bus is referred to as "parasite power." As an alternative, the DS18S20 may also be powered by an external supply on V_{DD}.

DS18S20 BLOCK DIAGRAM Figure 1

OPERATION – MEASURING TEMPERATURE

The core functionality of the DS18S20 is its direct-to-digital temperature sensor. The temperature sensor output has 9-bit resolution, which corresponds to 0.5°C steps. The DS18S20 powers-up in a low-power idle state; to initiate a temperature measurement and A-to-D conversion, the master must issue a Convert T [44h] command. Following the conversion, the resulting thermal data is stored in the 2-byte temperature register in the scratchpad memory and the DS18S20 returns to its idle state. If the DS18S20 is powered by an external supply, the master can issue “read time slots” (see the 1-WIRE BUS SYSTEM section) after the Convert T command and the DS18S20 will respond by transmitting 0 while the temperature conversion is in progress and 1 when the conversion is done. If the DS18S20 is powered with parasite power, this notification technique cannot be used since the bus must be pulled high by a strong pullup during the entire temperature conversion. The bus requirements for parasite power are explained in detail in the POWERING THE DS18S20 section of this datasheet.

The DS18S20 output data is calibrated in degrees centigrade; for Fahrenheit applications, a lookup table or conversion routine must be used. The temperature data is stored as a 16-bit sign-extended two’s complement number in the temperature register (see Figure 2). The sign bits (S) indicate if the temperature is positive or negative: for positive numbers S = 0 and for negative numbers S = 1. Table 2 gives examples of digital output data and the corresponding temperature reading.

Resolutions greater than 9 bits can be calculated using the data from the temperature, COUNT REMAIN and COUNT PER °C registers in the scratchpad. Note that the COUNT PER °C register is hard-wired to 16 (10h). After reading the scratchpad, the TEMP_READ value is obtained by truncating the 0.5°C bit (bit 0) from the temperature data (see Figure 2). The extended resolution temperature can then be calculated using the following equation:

$$TEMPERATURE = TEMP_READ - 0.25 + \frac{COUNT_PER_C - COUNT_REMAIN}{COUNT_PER_C}$$

Additional information about high-resolution temperature calculations can be found in Application Note 105: “High Resolution Temperature Measurement with Dallas Direct-to-Digital Temperature Sensors”.

TEMPERATURE REGISTER FORMAT Figure 2

	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
LS Byte	2 ⁶	2 ⁵	2 ⁴	2 ³	2 ²	2 ¹	2 ⁰	2 ⁻¹
	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8
MS Byte	S	S	S	S	S	S	S	S

TEMPERATURE/DATA RELATIONSHIP Table 2

TEMPERATURE	DIGITAL OUTPUT (Binary)	DIGITAL OUTPUT (Hex)
+85.0°C*	0000 0000 1010 1010	00AAh
+25.0°C	0000 0000 0011 0010	0032h
+0.5°C	0000 0000 0000 0001	0001h
0°C	0000 0000 0000 0000	0000h
-0.5°C	1111 1111 1111 1111	FFFFh
-25.0°C	1111 1111 1100 1110	FFCEh
-55.0°C	1111 1111 1001 0010	FF92h

*The power-on reset value of the temperature register is +85°C

OPERATION – ALARM SIGNALING

After the DS18S20 performs a temperature conversion, the temperature value is compared to the user-defined two's complement alarm trigger values stored in the 1-byte T_H and T_L registers (see Figure 3). The sign bit (S) indicates if the value is positive or negative: for positive numbers $S = 0$ and for negative numbers $S = 1$. The T_H and T_L registers are nonvolatile (EEPROM) so they will retain data when the device is powered down. T_H and T_L can be accessed through bytes 2 and 3 of the scratchpad as explained in the MEMORY section of this datasheet.

T_H AND T_L REGISTER FORMAT Figure 3

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
S	2^6	2^5	2^5	2^5	2^2	2^1	2^0

Only bits 8 through 1 of the temperature register are used in the T_H and T_L comparison since T_H and T_L are 8-bit registers. If the result of a temperature measurement is higher than T_H or lower than T_L , an alarm condition exists and an alarm flag is set inside the DS18S20. This flag is updated after every temperature measurement; therefore, if the alarm condition goes away, the flag will be turned off after the next temperature conversion.

The master device can check the alarm flag status of all DS18S20s on the bus by issuing an Alarm Search [ECh] command. Any DS18S20s with a set alarm flag will respond to the command, so the master can determine exactly which DS18S20s have experienced an alarm condition. If an alarm condition exists and the T_H or T_L settings have changed, another temperature conversion should be done to validate the alarm condition.

POWERING THE DS18S20

The DS18S20 can be powered by an external supply on the V_{DD} pin, or it can operate in “parasite power” mode, which allows the DS18S20 to function without a local external supply. Parasite power is very useful for applications that require remote temperature sensing or that are very space constrained. Figure 1 shows the DS18S20's parasite-power control circuitry, which “steals” power from the 1-wire bus via the DQ pin when the bus is high. The stolen charge powers the DS18S20 while the bus is high, and some of the charge is stored on the parasite power capacitor (C_{PP}) to provide power when the bus is low. When the DS18S20 is used in parasite power mode, the V_{DD} pin must be connected to ground.

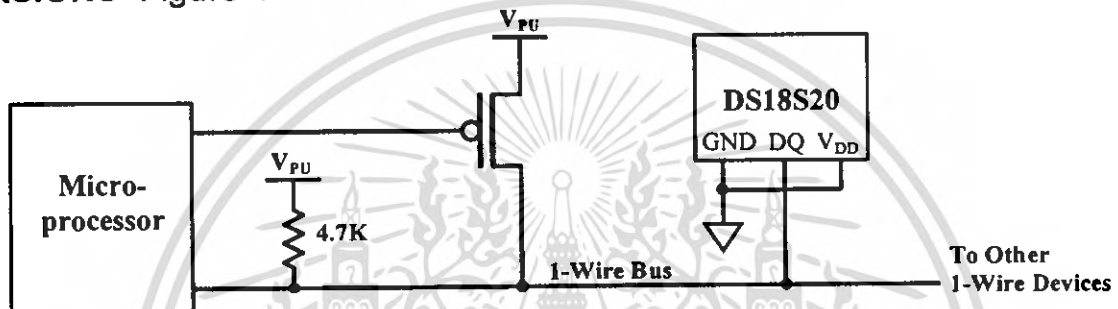
In parasite power mode, the 1-wire bus and C_{PP} can provide sufficient current to the DS18S20 for most operations as long as the specified timing and voltage requirements are met (refer to the DC ELECTRICAL CHARACTERISTICS and the AC ELECTRICAL CHARACTERISTICS sections of this data sheet). However, when the DS18S20 is performing temperature conversions or copying data from the scratchpad memory to EEPROM, the operating current can be as high as 1.5 mA. This current can cause an unacceptable voltage drop across the weak 1-wire pullup resistor and is more current than can be supplied by C_{PP} . To assure that the DS18S20 has sufficient supply current, it is necessary to provide a strong pullup on the 1-wire bus whenever temperature conversions are taking place or data is being copied from the scratchpad to EEPROM. This can be accomplished by using a MOSFET to pull the bus directly to the rail as shown in Figure 4. The 1-wire bus must be switched to the strong pullup within 10 μ s (max) after a Convert T [44h] or Copy Scratchpad [48h] command is issued, and the bus must be held high by the pullup for the duration of the conversion (t_{conv}) or data transfer ($t_{wr} = 10$ ms). No other activity can take place on the 1-wire bus while the pullup is enabled.

The DS18S20 can also be powered by the conventional method of connecting an external power supply to the V_{DD} pin, as shown in Figure 5. The advantage of this method is that the MOSFET pullup is not required, and the 1-wire bus is free to carry other traffic during the temperature conversion time.

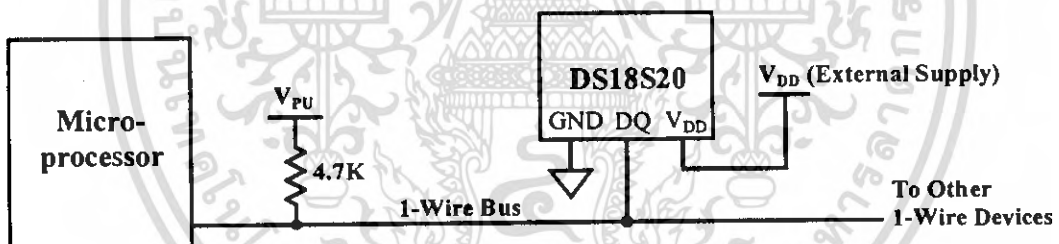
The use of parasite power is not recommended for temperatures above 100°C since the DS18S20 may not be able to sustain communications due to the higher leakage currents that can exist at these temperatures. For applications in which such temperatures are likely, it is strongly recommended that the DS18S20 be powered by an external power supply.

In some situations the bus master may not know whether the DS18S20s on the bus are parasite powered or powered by external supplies. The master needs this information to determine if the strong bus pullup should be used during temperature conversions. To get this information, the master can issue a Skip ROM [CCh] command followed by a Read Power Supply [B4h] command followed by a “read time slot”. During the read time slot, parasite powered DS18S20s will pull the bus low, and externally powered DS18S20s will let the bus remain high. If the bus is pulled low, the master knows that it must supply the strong pullup on the 1-wire bus during temperature conversions.

SUPPLYING THE PARASITE-POWERED DS18S20 DURING TEMPERATURE CONVERSIONS Figure 4



POWERING THE DS18S20 WITH AN EXTERNAL SUPPLY Figure 5



64-BIT LASERED ROM CODE

Each DS18S20 contains a unique 64-bit code (see Figure 6) stored in ROM. The least significant 8 bits of the ROM code contain the DS18S20's 1-wire family code: 10h. The next 48 bits contain a unique serial number. The most significant 8 bits contain a cyclic redundancy check (CRC) byte that is calculated from the first 56 bits of the ROM code. A detailed explanation of the CRC bits is provided in the CRC GENERATION section. The 64-bit ROM code and associated ROM function control logic allow the DS18S20 to operate as a 1-wire device using the protocol detailed in the 1-WIRE BUS SYSTEM section of this datasheet.

64-BIT LASERED ROM CODE Figure 6



MEMORY

The DS18S20's memory is organized as shown in Figure 7. The memory consists of an SRAM scratchpad with nonvolatile EEPROM storage for the high and low alarm trigger registers (T_H and T_L). Note that if the DS18S20 alarm function is not used, the T_H and T_L registers can serve as general-purpose memory. All memory commands are described in detail in the DS18S20 FUNCTION COMMANDS section.

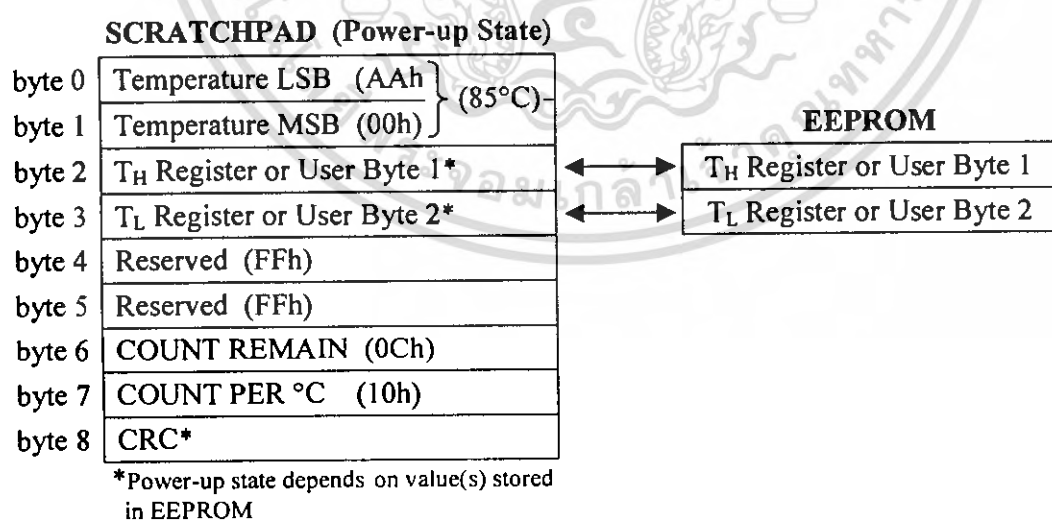
Byte 0 and byte 1 of the scratchpad contain the LSB and the MSB of the temperature register, respectively. These bytes are read-only. Bytes 2 and 3 provide access to T_H and T_L registers. Bytes 4 and 5 are reserved for internal use by the device and cannot be overwritten; these bytes will return all 1s when read. Bytes 6 and 7 contain the COUNT REMAIN and COUNT PER °C registers, which can be used to calculate extended resolution results as explained in the OPERATION – MEASURING TEMPERATURE section.

Byte 8 of the scratchpad is read-only and contains the cyclic redundancy check (CRC) code for bytes 0 through 7 of the scratchpad. The DS18S20 generates this CRC using the method described in the CRC GENERATION section.

Data is written to bytes 2 and 3 of the scratchpad using the Write Scratchpad [4Eh] command; the data must be transmitted to the DS18S20 starting with the least significant bit of byte 2. To verify data integrity, the scratchpad can be read (using the Read Scratchpad [BEh] command) after the data is written. When reading the scratchpad, data is transferred over the 1-wire bus starting with the least significant bit of byte 0. To transfer the T_H and T_L data from the scratchpad to EEPROM, the master must issue the Copy Scratchpad [48h] command.

Data in the EEPROM registers is retained when the device is powered down; at power-up the EEPROM data is reloaded into the corresponding scratchpad locations. Data can also be reloaded from EEPROM to the scratchpad at any time using the Recall E² [B8h] command. The master can issue “read time slots” (see the 1-WIRE BUS SYSTEM section) following the Recall E² command and the DS18S20 will indicate the status of the recall by transmitting 0 while the recall is in progress and 1 when the recall is done.

DS18S20 MEMORY MAP



1-WIRE BUS SYSTEM

The 1-wire bus system uses a single bus master to control one or more slave devices. The DS18S20 is always a slave. When there is only one slave on the bus, the system is referred to as a “single-drop” system; the system is “multi-drop” if there are multiple slaves on the bus.

All data and commands are transmitted least significant bit first over the 1-wire bus.

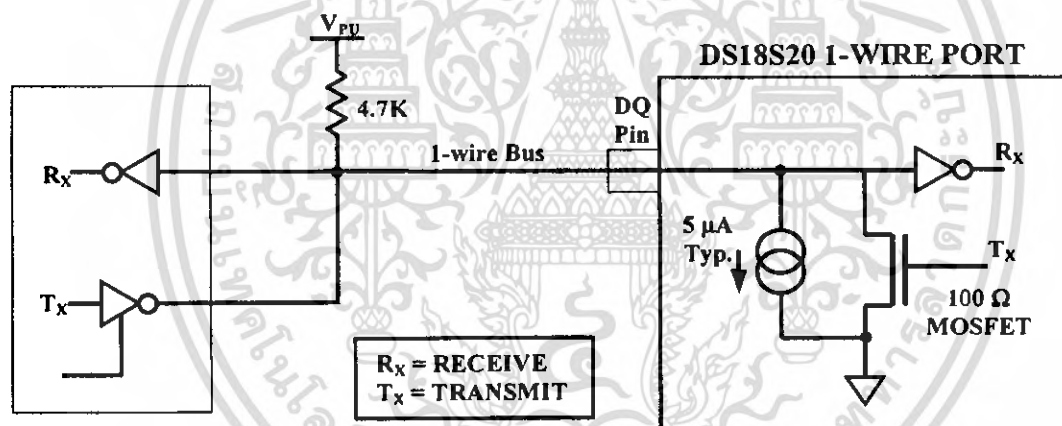
The following discussion of the 1-wire bus system is broken down into three topics: hardware configuration, transaction sequence, and 1-wire signaling (signal types and timing).

HARDWARE CONFIGURATION

The 1-wire bus has by definition only a single data line. Each device (master or slave) interfaces to the data line via an open drain or 3-state port. This allows each device to “release” the data line when the device is not transmitting data so the bus is available for use by another device. The 1-wire port of the DS18S20 (the DQ pin) is open drain with an internal circuit equivalent to that shown in Figure 9.

The 1-wire bus requires an external pullup resistor of approximately 5 k Ω ; thus, the idle state for the 1-wire bus is high. If for any reason a transaction needs to be suspended, the bus MUST be left in the idle state if the transaction is to resume. Infinite recovery time can occur between bits so long as the 1-wire bus is in the inactive (high) state during the recovery period. If the bus is held low for more than 480 μ s, all components on the bus will be reset.

HARDWARE CONFIGURATION Figure 9



TRANSACTION SEQUENCE

The transaction sequence for accessing the DS18S20 is as follows:

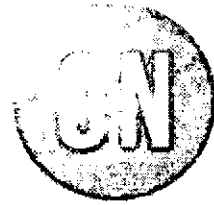
Step 1. Initialization

Step 2. ROM Command (followed by any required data exchange)

Step 3. DS18S20 Function Command (followed by any required data exchange)

It is very important to follow this sequence every time the DS18S20 is accessed, as the DS18S20 will not respond if any steps in the sequence are missing or out of order. Exceptions to this rule are the Search ROM [F0h] and Alarm Search [ECh] commands. After issuing either of these ROM commands, the master must return to Step 1 in the sequence.

LM339, LM239, LM2901, LM2901V, NCV2901, MC3302



ON Semiconductor®

<http://onsemi.com>

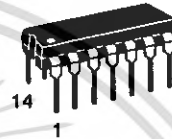
These comparators are designed for use in level detection, low-level sensing and memory applications in consumer, automotive, and industrial electronic applications.

Features

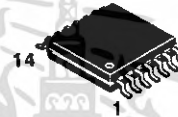
- Single or Split Supply Operation
- Low Input Bias Current: 25 nA (Typ)
- Low Input Offset Current: ± 5.0 nA (Typ)
- Low Input Offset Voltage
- Input Common Mode Voltage Range to GND
- Low Output Saturation Voltage: 130 mV (Typ) @ 4.0 mA
- TTL and CMOS Compatible
- ESD Clamps on the Inputs Increase Reliability without Affecting Device Operation
- NCV Prefix for Automotive and Other Applications Requiring Site and Control Changes
- Pb-Free Packages are Available



SOIC-14
D SUFFIX
CASE 751A

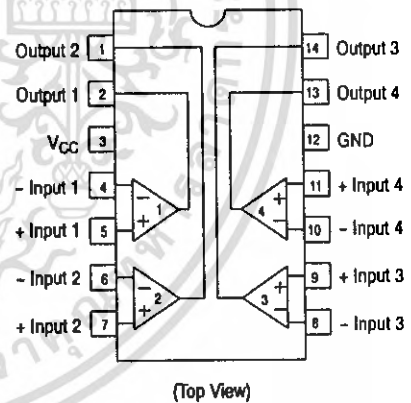


PDIP-14
N, P SUFFIX
CASE 646



TSSOP-14
DTB SUFFIX
CASE 948G

PIN CONNECTIONS



ORDERING INFORMATION

See detailed ordering and shipping information in the package dimensions section on page 6 of this data sheet.

DEVICE MARKING INFORMATION

See general marking information in the device marking section on page 8 of this data sheet.

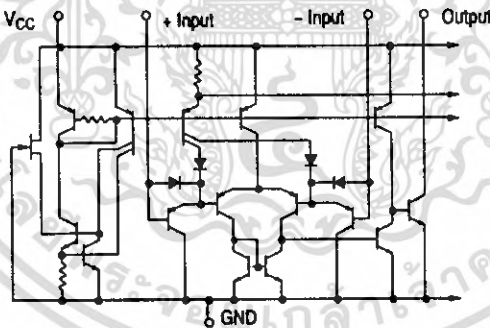
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Power Supply Voltage LM239/LM339/LM2901, V MC3302	V_{CC}	+36 or ± 18 +30 or ± 15	Vdc
Input Differential Voltage Range LM239/LM339/LM2901, V MC3302	V_{IDR}	36 30	Vdc
Input Common Mode Voltage Range	V_{ICMR}	-0.3 to V_{CC}	Vdc
Output Short Circuit to Ground (Note 1)	I_{SC}	Continuous	
Power Dissipation @ $T_A = 25^\circ\text{C}$ Plastic Package Derate above 25°C	P_D $1/R_{\theta JA}$	1.0 8.0	W mW/ $^\circ\text{C}$
Junction Temperature	T_J	150	$^\circ\text{C}$
Operating Ambient Temperature Range LM239 MC3302 LM2901 LM2901V, NCV2901 LM339	T_A	-25 to +85 -40 to +85 -40 to +105 -40 to +125 0 to +70	$^\circ\text{C}$
Storage Temperature Range	T_{stg}	-65 to +150	$^\circ\text{C}$
ESD Protection at any Pin (Note 2) Human Body Model Machine Model	V_{ESD}	1500 200	V

Stresses exceeding Maximum Ratings may damage the device. Maximum Ratings are stress ratings only. Functional operation above the Recommended Operating Conditions is not implied. Extended exposure to stresses above the Recommended Operating Conditions may affect device reliability.

1. The maximum output current may be as high as 20 mA, independent of the magnitude of V_{CC} . Output short circuits to V_{CC} can cause excessive heating and eventual destruction.
2. V_{ESD} rating for NCV/SC devices is: Human Body Model - 2000 V; Machine Model - 200 V.



NOTE: Diagram shown is for 1 comparator.

Figure 1. Circuit Schematic

LM339, LM239, LM2901, LM2901V, NCV2901, MC3302

ELECTRICAL CHARACTERISTICS ($V_{CC} = +5.0$ Vdc, $T_A = +25^\circ\text{C}$, unless otherwise noted)

Characteristic	Symbol	LM239/339			LM2901/2901V/ NCV2901			MC3302			Unit
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
Input Offset Voltage (Note 4)	V_{IO}	-	± 2.0	± 5.0	-	± 2.0	± 7.0	-	± 3.0	± 20	mVdc
Input Bias Current (Notes 4, 5) (Output in Analog Range)	I_{IB}	-	25	250	-	25	250	-	25	500	nA
Input Offset Current (Note 4)	I_{IO}	-	± 5.0	± 50	-	± 5.0	± 50	-	± 3.0	± 100	nA
Input Common Mode Voltage Range	V_{ICMR}	0	-	$V_{CC} - 1.5$	0	-	$V_{CC} - 1.5$	0	-	$V_{CC} - 1.5$	V
Supply Current $R_L = \infty$ (For All Comparators) $R_L = \infty, V_{CC} = 30$ Vdc	I_{CC}	-	0.8 1.0	2.0 2.5	-	0.8 1.0	2.0 2.5	-	0.8 1.0	2.0 2.5	mA
Voltage Gain $R_L \geq 15$ k Ω , $V_{CC} = 15$ Vdc	A_{VOL}	50	200	-	25	100	-	25	100	-	V/mV
Large Signal Response Time $V_I =$ TTL Logic Swing, $V_{ref} = 1.4$ Vdc, $V_{RL} = 5.0$ Vdc, $R_L = 5.1$ k Ω	-	-	300	-	-	300	-	-	300	-	ns
Response Time (Note 6) $V_{RL} = 5.0$ Vdc, $R_L = 5.1$ k Ω	-	-	1.3	-	-	1.3	-	-	1.3	-	μs
Output Sink Current $V_I(-) \geq +1.0$ Vdc, $V_I(+)=0$, $V_O \leq 1.5$ Vdc	I_{sink}	6.0	16	-	6.0	16	-	6.0	16	-	mA
Saturation Voltage $V_I(-) \geq +1.0$ Vdc, $V_I(+)=0$, $I_{sink} \leq 4.0$ mA	V_{sat}	-	130	400	-	130	400	-	130	500	mV
Output Leakage Current $V_I(+)\geq +1.0$ Vdc, $V_I(-)=0$, $V_O = +5.0$ Vdc	I_{OL}	-	0.1	-	-	0.1	-	-	0.1	-	nA

3. (LM239) $T_{low} = -25^\circ\text{C}$, $T_{high} = +85^\circ\text{C}$
 (LM339) $T_{low} = 0^\circ\text{C}$, $T_{high} = +70^\circ\text{C}$
 (MC3302) $T_{low} = -40^\circ\text{C}$, $T_{high} = +85^\circ\text{C}$
 (LM2901) $T_{low} = -40^\circ\text{C}$, $T_{high} = +105^\circ\text{C}$
 (LM2901V & NCV2901) $T_{low} = -40^\circ\text{C}$, $T_{high} = +125^\circ\text{C}$
NCV2901 is qualified for automotive use.
4. At the output switch point, $V_O = 1.4$ Vdc, $R_S \leq 100 \Omega$ 5.0 Vdc $\leq V_{CC} \leq 30$ Vdc, with the inputs over the full common mode range (0 Vdc to $V_{CC} - 1.5$ Vdc).
5. The bias current flows out of the inputs due to the PNP input stage. This current is virtually constant, independent of the output state.
6. The response time specified is for a 100 mV input step with 5.0 mV overdrive. For larger signals, 300 ns is typical.

LM339, LM239, LM2901, LM2901V, NCV2901, MC3302

PERFORMANCE CHARACTERISTICS ($V_{CC} = +5.0$ Vdc, $T_A = T_{low}$ to T_{high} [Note 7])

Characteristic	Symbol	LM239/339			LM2901/2901V/ NCV2901			MC3302			Unit
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
Input Offset Voltage (Note 8)	V_{IO}	-	-	± 9.0	-	-	± 15	-	-	± 40	mVdc
Input Bias Current (Notes 8, 9) (Output in Analog Range)	I_{IB}	-	-	400	-	-	500	-	-	1000	nA
Input Offset Current (Note 8)	I_{IO}	-	-	± 150	-	-	± 200	-	-	± 300	nA
Input Common Mode Voltage Range	V_{ICMR}	0	-	$V_{CC} - 2.0$	0	-	$V_{CC} - 2.0$	0	-	$V_{CC} - 2.0$	V
Saturation Voltage $V_{I(-)} \geq +1.0$ Vdc, $V_{I(+)} = 0$, $I_{sink} \leq 4.0$ mA	V_{sat}	-	-	700	-	-	700	-	-	700	mV
Output Leakage Current $V_{I(+)} \geq +1.0$ Vdc, $V_{I(-)} = 0$, $V_O = 30$ Vdc	I_{OL}	-	-	1.0	-	-	1.0	-	-	1.0	μ A
Differential Input Voltage All $V_i \geq 0$ Vdc	V_{ID}	-	-	V_{CC}	-	-	V_{CC}	-	-	V_{CC}	Vdc

7. (LM239) $T_{low} = -25^\circ\text{C}$, $T_{high} = +85^\circ\text{C}$
 (LM339) $T_{low} = 0^\circ\text{C}$, $T_{high} = +70^\circ\text{C}$
 (MC3302) $T_{low} = -40^\circ\text{C}$, $T_{high} = +85^\circ\text{C}$
 (LM2901) $T_{low} = -40^\circ\text{C}$, $T_{high} = +105^\circ\text{C}$
 (LM2901V & NCV2901) $T_{low} = -40^\circ\text{C}$, $T_{high} = +125^\circ\text{C}$
NCV2901 is qualified for automotive use.
8. At the output switch point, $V_O \approx 1.4$ Vdc, $R_S \leq 100 \Omega$, 5.0 Vdc $\leq V_{CC} \leq 30$ Vdc, with the inputs over the full common mode range (0 Vdc to $V_{CC} - 1.5$ Vdc).
9. The bias current flows out of the inputs due to the PNP input stage. This current is virtually constant, independent of the output state.
10. The response time specified is for a 100 mV input step with 5.0 mV overdrive. For larger signals, 300 ns is typical.

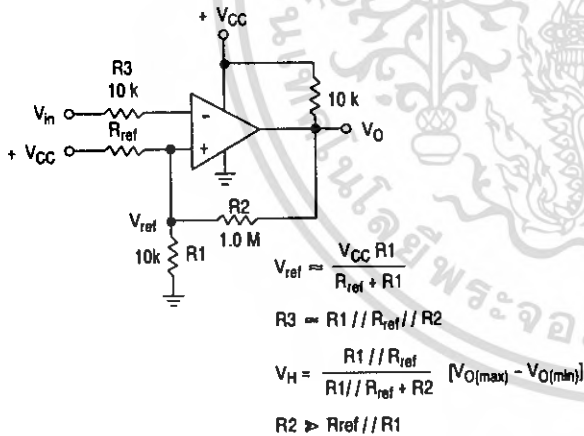


Figure 2. Inverting Comparator with Hysteresis

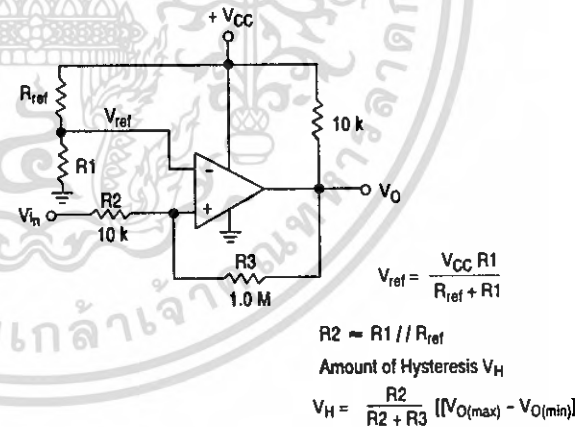


Figure 3. Noninverting Comparator with Hysteresis



Octal High Voltage, High Current Darlington Transistor Arrays

The eight NPN Darlington connected transistors in this family of arrays are ideally suited for interfacing between low logic level digital circuitry (such as TTL, CMOS or PMOS/NMOS) and the higher current/voltage requirements of lamps, relays, printer hammers or other similar loads for a broad range of computer, industrial, and consumer applications. All devices feature open-collector outputs and free wheeling clamp diodes for transient suppression.

The ULN2803 is designed to be compatible with standard TTL families while the ULN2804 is optimized for 6 to 15 volt high level CMOS or PMOS.

MAXIMUM RATINGS ($T_A = 25^\circ\text{C}$ and rating apply to any one device in the package, unless otherwise noted.)

Rating	Symbol	Value	Unit
Output Voltage	V_O	50	V
Input Voltage (Except ULN2801)	V_I	30	V
Collector Current - Continuous	I_C	500	mA
Base Current - Continuous	I_B	25	mA
Operating Ambient Temperature Range	T_A	0 to +70	$^\circ\text{C}$
Storage Temperature Range	T_{stg}	-55 to +150	$^\circ\text{C}$
Junction Temperature	T_J	125	$^\circ\text{C}$

$R_{\theta JA} = 55^\circ\text{C/W}$

Do not exceed maximum current limit per driver.

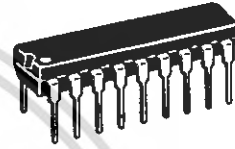
ORDERING INFORMATION

Device	Characteristics		
	Input Compatibility	$V_{CE}(\text{Max})/I_C(\text{Max})$	Operating Temperature Range
ULN2803A	TTL, 5.0 V CMOS	50 V/500 mA	$T_A = 0 \text{ to } +70^\circ\text{C}$
ULN2804A	6 to 15 V CMOS, PMOS		

ULN2803 ULN2804

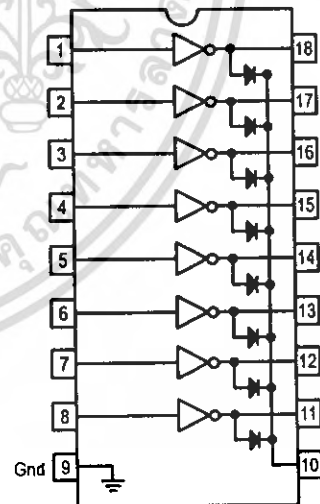
OCTAL PERIPHERAL DRIVER ARRAYS

SEMICONDUCTOR TECHNICAL DATA



A SUFFIX
PLASTIC PACKAGE
CASE 707

PIN CONNECTIONS



ULN2803 ULN2804

ELECTRICAL CHARACTERISTICS ($T_A = 25^\circ\text{C}$, unless otherwise noted)

Characteristic		Symbol	Min	Typ	Max	Unit
Output Leakage Current (Figure 1) ($V_O = 50\text{ V}$, $T_A = +70^\circ\text{C}$) ($V_O = 50\text{ V}$, $T_A = +25^\circ\text{C}$) ($V_O = 50\text{ V}$, $T_A = +70^\circ\text{C}$, $V_I = 6.0\text{ V}$) ($V_O = 50\text{ V}$, $T_A = +70^\circ\text{C}$, $V_I = 1.0\text{ V}$)	All Types All Types ULN2802 ULN2804	I_{CEX}	- - - -	- - - -	100 50 500 500	μA
Collector-Emitter Saturation Voltage (Figure 2) ($I_C = 350\text{ mA}$, $I_B = 500\text{ }\mu\text{A}$) ($I_C = 200\text{ mA}$, $I_B = 350\text{ }\mu\text{A}$) ($I_C = 100\text{ mA}$, $I_B = 250\text{ }\mu\text{A}$)	All Types All Types All Types	$V_{CE(sat)}$	- - -	1.1 0.95 0.85	1.6 1.3 1.1	V
Input Current – On Condition (Figure 4) ($V_I = 17\text{ V}$) ($V_I = 3.85\text{ V}$) ($V_I = 5.0\text{ V}$) ($V_I = 12\text{ V}$)	ULN2802 ULN2803 ULN2804 ULN2804	$I_{I(on)}$	- - - -	0.82 0.93 0.35 1.0	1.25 1.35 0.5 1.45	mA
Input Voltage – On Condition (Figure 5) ($V_{CE} = 2.0\text{ V}$, $I_C = 300\text{ mA}$) ($V_{CE} = 2.0\text{ V}$, $I_C = 200\text{ mA}$) ($V_{CE} = 2.0\text{ V}$, $I_C = 250\text{ mA}$) ($V_{CE} = 2.0\text{ V}$, $I_C = 300\text{ mA}$) ($V_{CE} = 2.0\text{ V}$, $I_C = 125\text{ mA}$) ($V_{CE} = 2.0\text{ V}$, $I_C = 200\text{ mA}$) ($V_{CE} = 2.0\text{ V}$, $I_C = 275\text{ mA}$) ($V_{CE} = 2.0\text{ V}$, $I_C = 350\text{ mA}$)	ULN2802 ULN2803 ULN2803 ULN2803 ULN2804 ULN2804 ULN2804 ULN2804	$V_{I(on)}$	- - - - - - - -	- - - - - - - -	13 2.4 2.7 3.0 5.0 6.0 7.0 8.0	V
Input Current – Off Condition (Figure 3) ($I_C = 500\text{ }\mu\text{A}$, $T_A = +70^\circ\text{C}$)	All Types	$I_{I(off)}$	50	100	-	μA
DC Current Gain (Figure 2) ($V_{CE} = 2.0\text{ V}$, $I_C = 350\text{ mA}$)	ULN2801	h_{FE}	1000	-	-	-
Input Capacitance		C_i	-	15	25	pF
Turn-On Delay Time (50% E_I to 50% E_O)		t_{on}	-	0.25	1.0	μs
Turn-Off Delay Time (50% E_I to 50% E_O)		t_{off}	-	0.25	1.0	μs
Clamp Diode Leakage Current (Figure 6) ($V_R = 50\text{ V}$)	$T_A = +25^\circ\text{C}$ $T_A = +70^\circ\text{C}$	I_R	-	-	50 100	μA
Clamp Diode Forward Voltage (Figure 7) ($I_F = 350\text{ mA}$)		V_F	-	1.5	2.0	V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

M75176P, M75177P M75178P, M75179P

RS-485 TRANSCEIVER

DESCRIPTION

Each of the M75176 to M75179 is a semiconductor IC with a built-in differential driver and a built-in differential receiver both of which meet the EIA standards RS-422A and RS-485.

FEATURES

Common

- 5V single power supply
- With output control input (except M75179P)

Driver

- 54Ω terminal resistor connectable between outputs.
- High output impedance at power-OFF time
- Built-in output current limit circuit

Receiver

- High input sensitivity ($\pm 200\text{mV}$ max.)
- Hysteresis input (50mV typ.)
- High input impedance (12kΩ min.)
- "H" emitted when input is open (failsafe function)

APPLICATION

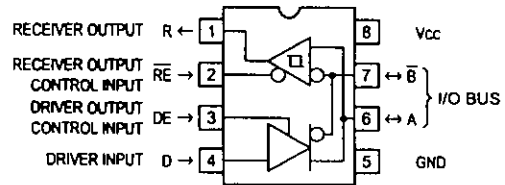
High-speed data transmission interface for digital equipment

DESCRIPTION OF FUNCTION

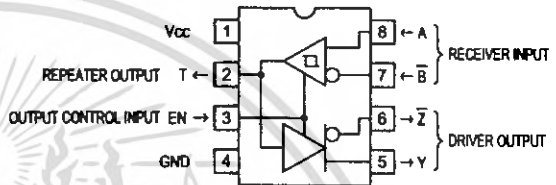
The M75176P to M75179P are line interface ICs which meet EIA standards RS-422A and RS-485, and are suitable for long-distance, high-speed data transmission.

The M75176P is designed to connect the driver circuit output and the receiver input to each other internally, and serves as a line transceiver. The M75177P/M75178P is designed to connect the receiver circuit output and the driver circuit input internally, and serves as a line repeater. The M75177P and M75178P enter the output enabled state with active "H" and active "L", respectively. The M75179P is equipped with a built-in differential driver and a built-in differential receiver. It does not have an output control input pin but its output impedance at the power-OFF time goes high.

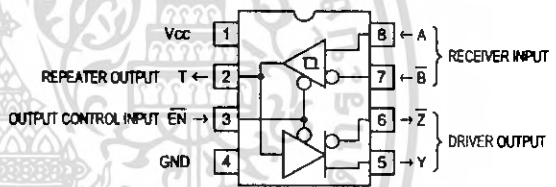
PIN CONFIGURATION (TOP VIEW)



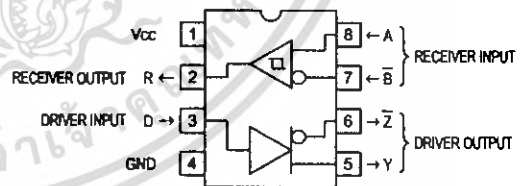
M75176P



M75177P



M75178P



M75179P

Outline 8P4



FUNCTION TABLE (Note 1)

(a) M75176P

Driver

INPUTS		OUTPUTS	
D	DE	A	\bar{B}
H	H	H	L
L	H	L	H
X	L	Z	Z

Receiver

INPUTS		OUTPUTS
V _{id}	RE	R
V _{id} > 0.2V	L	H
-0.2V < V _{id} < 0.2V	L	*
V _{id} < -0.2V	L	L
X	H	Z

(b) M75177P/M75178P

V _{id}	INPUTS		OUTPUTS		
	EN (M75177P)	\bar{EN} (M75178P)	T	Y	\bar{Z}
V _{id} > 0.2V	H	L	H	H	L
-0.2V < V _{id} < 0.2V	H	L	*	*	*
V _{id} < -0.2V	H	L	L	L	H
X	L	H	Z	Z	Z

(c) M75179P

Driver

INPUTS	OUTPUTS	
D	Y	Z
H	H	L
L	L	H

Receiver

INPUTS	OUTPUTS
V _{id}	R
V _{id} > 0.2V	H
-0.2V < V _{id} < 0.2V	*
V _{id} < -0.2V	L

Note 1: V_{id}: (A applied voltage) – (\bar{B} applied voltage)
 X : Either "L" or "H"
 * : Output state is not defined.
 Z : High impedance state.

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Conditions	Ratings	Unit
V _{CC}	Supply voltage		-0.5 ~ +7	V
V _{ID}	Differential input voltage	75177/178/179	-25 ~ +25	V
V _{IE}	Enable input voltage	75176/177/178	-0.5 ~ +5.5	V
P _d	Power dissipation (Note 2)	When T _a =25°C	925	mW
T _{stg}	Storage temperature		-65 ~ 150	°C

Note 2: When T_a ≥ 25°C, do derating according to the attached thermal derating.

RECOMMENDED OPERATING CONDITIONS

Symbol	Parameter	Limits			Unit	
		Min.	Typ.	Max.		
V _{CC}	Supply voltage	4.75	5	5.25	V	
V _I , V _{IC}	Bus pin voltage (each pin voltage, in-phase input voltage)	-7		+12	V	
V _{ID}	Differential input voltage	-12		+12	V	
I _{OH}	"H" output current	Driver	0	-60	mA	
		Receiver	0	-400	μA	
I _{OL}	"L" output current	Driver	0	60	mA	
		Receiver	V _{OL} < 0.45V	0	8	mA
			V _{OL} < 0.5V	0	16	mA
T _{opr}	Operating ambient temperature	-20		75	°C	

ELECTRICAL CHARACTERISTICS (Driver) (V_{CC} = 5V ± 5%, T_a = -20 ~ 75°C, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit	
			Min.	Typ.*	Max.		
V _{IH}	"H" input voltage		2			V	
V _{IL}	"L" input voltage				0.8	V	
V _{IK}	Input clamp voltage	I _I = -18mA			-1.5	V	
V _{OD1}	Differential output voltage 1	I _O = 0mA	1.5		6	V	
V _{OD2}	Differential output voltage 2	R _L = 100Ω See Fig. 1.	2 (Note 3)	2.42		V	
		R _L = 54Ω See Fig. 1.	1.5	2.18	5		
Δ V _{OD}	Differential output voltage variance width				±0.2	V	
V _{OL}	In-phase output voltage	R _L = 54/100Ω See Fig. 1.	-1	2.08	3	V	
Δ V _{OC}	In-phase output voltage variance width				±0.2	V	
I _O	Output current	75176	Output disable	V _O = 12V	1	mA	
		75177/178/179	V _{CC} = 0V, V _O = -7 ~ +12V	V _O = -7V	-0.8		
I _{OZ}	Off-state output current	75177/178	V _O = -7 ~ +12V		±100	μA	
I _{IH}	"H" input current	V _I = 2.4V			±300	μA	
I _{IL}	"L" input current	V _I = 0.4V			20	μA	
I _{OS}	Output short-circuit current	V _O = -7V			-250	mA	
		V _O = 0V			-150		
		V _O = V _{CC}			250		
		V _O = 12V			250		
I _{CC}	Supply voltage	No output load	Output enable		29.5	48	mA
			Output disable			32.5	

*: The standard value is given on the condition of V_{CC} = 5V and T_a = 25°C.

Note 3: The standard is 1.9V (min.) when T_a ≤ 0°C.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SWITCHING CHARACTERISTICS (Driver) (Vcc = 5V, Ta=25°C)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
tDD	Differential output delay time	RL=54Ω	See Fig. 2.	13	25	ns
tTD	Differential output transition time	CL=50pF		11	25	ns
tPLH	Output "L - H" propagation delay time	RL=27Ω	See Fig. 3.	15	25	ns
tPHL	Output "H - L" propagation delay time	CL=50pF		12	25	ns
tPZH	"H" enable time	RL=110Ω CL=50pF	See Figs. 4 & 5.	23	35	ns
tPZL	"L" enable time			16	35	ns
tPHZ	"H" disable time			11	25	ns
tPLZ	"L" disable time			22	35	ns

ELECTRICAL CHARACTERISTICS (Receiver) (Vcc = 5V ± 5%, Ta = -20 ~ 75°C, unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit	
			Min.	Typ.*	Max.		
VTH	High threshold voltage	Vo=2.7V, Io=-0.4mA			0.2	V	
VTL	Low threshold voltage	Vo=0.5V, Io=16mA	-0.2			V	
VT+ - VT-	Hysteresis width (Note 4)			50		mV	
Vik	Enable input clamp voltage	Ii=-18mA			-1.5	V	
VOH	"H" output voltage	VIo=0.2V, IOH=-0.4mA	2.7	3.6		V	
VOL	"L" output voltage	VIo=-0.2V	IoL=8mA	0.23	0.45	V	
			IoL=16mA	0.30	0.5		
IoZ	Off-state output current	75176	Vo=0.4 ~ 2.4V		±20	µA	
		75177/178	Vo=0.4V		-400		
			Vo=2.4V		20		
Ii	Line input current	Other input 0V	Vo=12V		1	mA	
			Vo=-7V		-0.8		
IiH	"H" enable input current	75176/177/178	VIH=2.7V		20	µA	
IiL	"L" enable input current	75176	VIL=0.4V		-100	µA	
		75177/178			-400		
ri	Input resistance			12		kΩ	
IOS	Output short-circuit current	Vo=0V			-15	-85	mA
ICC	Supply voltage	No output load	Output enable			48	mA
			Output disable				

*: The standard value is given on the condition of Vcc = 5V and Ta = 25°C.

Note 4: The hysteresis width is the difference between positive threshold voltage VT+ and negative threshold voltage VT-.

SWITCHING CHARACTERISTICS (Receiver) (Vcc = 5V, Ta=25°C)

Symbol	Parameter	Test conditions	Limits			Unit
			Min.	Typ.	Max.	
tPLH	Output "L - H" propagation delay time	CL=15pF	See Fig. 6.	22	35	ns
tPHL	Output "H - L" propagation delay time			19	35	ns
tPZH	"H" enable time	CL=15pF	See Fig. 7.	9	20	ns
tPZL	"L" enable time			11	20	ns
tPHZ	"H" disable time			17	35	ns
tPLZ	"L" disable time	CL=15pF	See Fig. 7.	22	35	ns



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MAXIM**+5V-Powered, Multichannel RS-232 Drivers/Receivers****General Description**

The MAX220–MAX249 family of line drivers/receivers is intended for all EIA/TIA-232E and V.28/V.24 communications interfaces, particularly applications where $\pm 12V$ is not available.

These parts are especially useful in battery-powered systems, since their low-power shutdown mode reduces power dissipation to less than 5 μ W. The MAX225, MAX233, MAX235, and MAX245/MAX246/MAX247 use no external components and are recommended for applications where printed circuit board space is critical.

Applications

Portable Computers
Low-Power Modems
Interface Translation
Battery-Powered RS-232 Systems
Multidrop RS-232 Networks

Features**Superior to Bipolar**

- ◆ Operate from Single +5V Power Supply (+5V and +12V—MAX231/MAX239)
- ◆ Low-Power Receive Mode in Shutdown (MAX223/MAX242)
- ◆ Meet All EIA/TIA-232E and V.28 Specifications
- ◆ Multiple Drivers and Receivers
- ◆ 3-State Driver and Receiver Outputs
- ◆ Open-Line Detection (MAX243)

Ordering Information

PART	TEMP. RANGE	PIN-PACKAGE
MAX220CPE	0°C to +70°C	16 Plastic DIP
MAX220CSE	0°C to +70°C	16 Narrow SO
MAX220CWE	0°C to +70°C	16 Wide SO
MAX220C/D	0°C to +70°C	Dice*
MAX220EPE	-40°C to +85°C	16 Plastic DIP
MAX220ESE	-40°C to +85°C	16 Narrow SO
MAX220EWE	-40°C to +85°C	16 Wide SO
MAX220EJE	-40°C to +85°C	16 CERDIP
MAX220MJE	-55°C to +125°C	16 CERDIP

Ordering information continued at end of data sheet.

*Contact factory for dice specifications.

Selection Table

Part Number	Power Supply (V)	No. of RS-232 Drivers/Rx	No. of Ext. Caps	Nominal Cap. Value (μ F)	SHDN & Three-State	Rx Active in SHDN	Data Rate (kbps)	Features
MAX220	+5	2/2	4	0.1	No	—	120	Ultra-low-power, industry-standard pinout
MAX222	+5	2/2	4	0.1	Yes	—	200	Low-power shutdown
MAX223 (MAX213)	+5	4/5	4	1.0 (0.1)	Yes	✓	120	MAX241 and receivers active in shutdown
MAX225	+5	5/5	0	—	Yes	✓	120	Available in SO
MAX230 (MAX200)	+5	5/0	4	1.0 (0.1)	Yes	—	120	5 drivers with shutdown
MAX231 (MAX201)	+5 and +7.5 to +13.2	2/2	2	1.0 (0.1)	No	—	120	Standard +5/+12V or battery supplies; same functions as MAX232
MAX232 (MAX202)	+5	2/2	4	1.0 (0.1)	No	—	120 (64)	Industry standard
MAX232A	+5	2/2	4	0.1	No	—	200	Higher slew rate, small caps
MAX233 (MAX203)	+5	2/2	0	—	No	—	120	No external caps
MAX233A	+5	2/2	0	—	No	—	200	No external caps, high slew rate
MAX234 (MAX204)	+5	4/0	4	1.0 (0.1)	No	—	120	Replaces 1488
MAX235 (MAX205)	+5	5/5	0	—	Yes	—	120	No external caps
MAX236 (MAX206)	+5	4/3	4	1.0 (0.1)	Yes	—	120	Shutdown, three state
MAX237 (MAX207)	+5	5/3	4	1.0 (0.1)	No	—	120	Complements IBM PC serial port
MAX238 (MAX208)	+5	4/4	4	1.0 (0.1)	No	—	120	Replaces 1488 and 1489
MAX239 (MAX209)	+5 and +7.5 to +13.2	3/5	2	1.0 (0.1)	No	—	120	Standard +5/+12V or battery supplies; single-package solution for IBM PC serial port
MAX240	+5	5/5	4	1.0	Yes	—	120	DIP or flatpack package
MAX241 (MAX211)	+5	4/5	4	1.0 (0.1)	Yes	—	120	Complete IBM PC serial port
MAX242	+5	2/2	4	0.1	Yes	✓	200	Separate shutdown and enable
MAX243	+5	2/2	4	0.1	No	—	200	Open-line detection simplifies cabling
MAX244	+5	8/10	4	1.0	No	—	120	High slew rate
MAX245	+5	8/10	0	—	Yes	✓	120	High slew rate, int. caps, two shutdown modes
MAX246	+5	8/10	0	—	Yes	✓	120	High slew rate, int. caps, three shutdown modes
MAX247	+5	8/9	0	—	Yes	✓	120	High slew rate, int. caps, nine operating modes
MAX248	+5	8/8	4	1.0	Yes	✓	120	High slew rate, selective half-chip enables
MAX249	+5	6/10	4	1.0	Yes	✓	120	Available in quad flatpack package

MAXIM

Maxim Integrated Products 1

For pricing, delivery, and ordering information, please contact Maxim/Dallas Direct! at 1-888-629-4642, or visit Maxim's website at www.maxim-ic.com.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MAX220-MAX249

+5V-Powered, Multichannel RS-232 Drivers/Receivers

ABSOLUTE MAXIMUM RATINGS—MAX220/222/232A/233A/242/243

Supply Voltage (V _{CC})	-0.3V to +6V	20-Pin Plastic DIP (derate 8.00mW/°C above +70°C)	440mW
Input Voltages		16-Pin Narrow SO (derate 8.70mW/°C above +70°C)	696mW
T _{IN}	-0.3V to (V _{CC} - 0.3V)	16-Pin Wide SO (derate 9.52mW/°C above +70°C)	762mW
R _{IN} (Except MAX220)	±30V	18-Pin Wide SO (derate 9.52mW/°C above +70°C)	762mW
R _{IN} (MAX220)	±25V	20-Pin Wide SO (derate 10.00mW/°C above +70°C)	800mW
T _{OUT} (Except MAX220) (Note 1)	±15V	20-Pin SSOP (derate 8.00mW/°C above +70°C)	640mW
T _{OUT} (MAX220)	±13.2V	16-Pin CERDIP (derate 10.00mW/°C above +70°C)	800mW
Output Voltages		18-Pin CERDIP (derate 10.53mW/°C above +70°C)	842mW
T _{OUT}	±15V	Operating Temperature Ranges	
R _{OUT}	-0.3V to (V _{CC} + 0.3V)	MAX2__AC__, MAX2__C__	0°C to +70°C
Driver/Receiver Output Short Circuited to GND	Continuous	MAX2__AE__, MAX2__E__	-40°C to +85°C
Continuous Power Dissipation (T _A = +70°C)		MAX2__AM__, MAX2__M__	-55°C to +125°C
16-Pin Plastic DIP (derate 10.53mW/°C above +70°C)	842mW	Storage Temperature Range	-65°C to +160°C
18-Pin Plastic DIP (derate 11.11mW/°C above +70°C)	889mW	Lead Temperature (soldering, 10sec)	+300°C

Note 1: Input voltage measured with T_{OUT} in high-impedance state, $\overline{\text{SHDN}}$ or V_{CC} = 0V.

Note 2: For the MAX220, V₊ and V₋ can have a maximum magnitude of 7V, but their absolute difference cannot exceed 13V.

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

ELECTRICAL CHARACTERISTICS—MAX220/222/232A/233A/242/243

(V_{CC} = +5V ±10%, C1–C4 = 0.1μF, MAX220, C1 = 0.047μF, C2–C4 = 0.33μF, T_A = T_{MIN} to T_{MAX}, unless otherwise noted.)

PARAMETER	CONDITIONS		MIN	TYP	MAX	UNITS
RS-232 TRANSMITTERS						
Output Voltage Swing	All transmitter outputs loaded with 3kΩ to GND		±5	±8		V
Input Logic Threshold Low				1.4	0.8	V
Input Logic Threshold High	All devices except MAX220		2	1.4		V
	MAX220: V _{CC} = 5.0V		2.4			
Logic Pull-Up/Input Current	All except MAX220, normal operation			5	40	μA
	SHDN = 0V, MAX222/242, shutdown, MAX220			±0.01	±1	
Output Leakage Current	V _{CC} = 5.5V, SHDN = 0V, V _{OUT} = ±15V, MAX222/242			±0.01	±10	μA
	V _{CC} = SHDN = 0V, V _{OUT} = ±15V			±0.01	±10	
Data Rate				200	116	kb/s
Transmitter Output Resistance	V _{CC} = V ₊ = V ₋ = 0V, V _{OUT} = ±2V		300	10M		Ω
Output Short-Circuit Current	V _{OUT} = 0V		±7	±22		mA
RS-232 RECEIVERS						
RS-232 Input Voltage Operating Range					±30	V
RS-232 Input Threshold Low	V _{CC} = 5V	All except MAX243 R _{2IN}	0.8	1.3		V
		MAX243 R _{2IN} (Note 2)	-3			
RS-232 Input Threshold High	V _{CC} = 5V	All except MAX243 R _{2IN}		1.8	2.4	V
		MAX243 R _{2IN} (Note 2)		-0.5	-0.1	
RS-232 Input Hysteresis	All except MAX243, V _{CC} = 5V, no hysteresis in shdn.		0.2	0.5	1	V
	MAX243			1		
RS-232 Input Resistance			3	5	7	kΩ
TTL/CMOS Output Voltage Low	I _{OUT} = 3.2mA			0.2	0.4	V
TTL/CMOS Output Voltage High	I _{OUT} = -1.0mA		3.5	V _{CC} - 0.2		V
TTL/CMOS Output Short-Circuit Current	Sourcing V _{OUT} = GND		-2	-10		mA
	Sinking V _{OUT} = V _{CC}		10	30		

+5V-Powered, Multichannel RS-232 Drivers/Receivers

MAX220-MAX249

ELECTRICAL CHARACTERISTICS—MAX220/222/232A/233A/242/243 (continued)

(V_{CC} = +5V ±10%, C₁–C₄ = 0.1μF, MAX220, C₁ = 0.047μF, C₂–C₄ = 0.33μF, T_A = T_{MIN} to T_{MAX}, unless otherwise noted.)

PARAMETER	CONDITIONS		MIN	TYP	MAX	UNITS
TTL/CMOS Output Leakage Current	SHDN = V _{CC} or EN = V _{CC} (SHDN = 0V for MAX222), 0V ≤ V _{OUT} ≤ V _{CC}			±0.05	±10	μA
EN Input Threshold Low	MAX242			1.4	0.8	V
EN Input Threshold High	MAX242		2.0	1.4		V
Operating Supply Voltage			4.5		5.5	V
V _{CC} Supply Current (SHDN = V _{CC}), Figures 5, 6, 11, 19	No load	MAX220		0.5	2	mA
		MAX222/232A/233A/242/243		4	10	
	3kΩ load both inputs	MAX220		12		
		MAX222/232A/233A/242/243		15		
Shutdown Supply Current	MAX222/242	T _A = +25°C		0.1	10	μA
		T _A = 0°C to +70°C		2	50	
		T _A = -40°C to +85°C		2	50	
		T _A = -55°C to +125°C		35	100	
SHDN Input Leakage Current	MAX222/242				±1	μA
SHDN Threshold Low	MAX222/242			1.4	0.8	V
SHDN Threshold High	MAX222/242		2.0	1.4		V
Transition Slew Rate	C _L = 50pF to 2500pF, R _L = 3kΩ to 7kΩ, V _{CC} = 5V, T _A = +25°C, measured from +3V to -3V or -3V to +3V	MAX222/232A/233A/242/243	6	12	30	V/μs
		MAX220	1.5	3	30	
Transmitter Propagation Delay TLL to RS-232 (normal operation), Figure 1	t _{PHLT}	MAX222/232A/233A/242/243		1.3	3.5	μs
		MAX220		4	10	
	t _{PLHT}	MAX222/232A/233A/242/243		1.5	3.5	
		MAX220		5	10	
Receiver Propagation Delay RS-232 to TLL (normal operation), Figure 2	t _{PHLR}	MAX222/232A/233A/242/243		0.5	1	μs
		MAX220		0.6	3	
	t _{PLHR}	MAX222/232A/233A/242/243		0.6	1	
		MAX220		0.8	3	
Receiver Propagation Delay RS-232 to TLL (shutdown), Figure 2	t _{PHLS}	MAX242		0.5	10	μs
	t _{PLHS}	MAX242		2.5	10	
Receiver-Output Enable Time, Figure 3	t _{ER}	MAX242		125	500	ns
Receiver-Output Disable Time, Figure 3	t _{DR}	MAX242		160	500	ns
Transmitter-Output Enable Time (SHDN goes high), Figure 4	t _{ET}	MAX222/242, 0.1μF caps (includes charge-pump start-up)		250		μs
Transmitter-Output Disable Time (SHDN goes low), Figure 4	t _{DT}	MAX222/242, 0.1μF caps		600		ns
Transmitter + to - Propagation Delay Difference (normal operation)	t _{PHLT} - t _{PLHT}	MAX222/232A/233A/242/243		300		ns
		MAX220		2000		
Receiver + to - Propagation Delay Difference (normal operation)	t _{PHLR} - t _{PLHR}	MAX222/232A/233A/242/243		100		ns
		MAX220		225		

Note 3: MAX243 R_{2OUT} is guaranteed to be low when R_{2IN} is ≥ 0V or is floating.

MAXIM

3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้