HOSTESS

Hostess™ Series
Hostess™ *550* Series

Programming Guide

COMTROL
Powerful Choices

First Edition, September 6, 1994

Copyright © 1994. Comtrol Corporation. All Rights Reserved.

Comtrol Corporation makes no representations or warranties with regard to the contents of this guide or to the suitability of the Comtrol Hostess or Hostess *550* controllers for any particular purpose.

Trademarks

The Comtrol logo and the Hostess or Hostess *550* series are trademarks of Comtrol Corporation.

Comtrol is a registered trademark of Comtrol Corporation.

Product names mentioned herein may be trademarks and/or registered trademarks of their respective companies.

Document Number: 6231

Before You Begin

Scope

This *Programming Guide* discusses configuration and programming issues for the following Comtrol controllers:

- Hostess 2-port
- Hostess *550* 2-port
- Hostess 4-port with a 100-pin connector or the RJ45 connector
- Hostess *550* 4-port with a 100-pin connector or the RJ45 connector
- Hostess 8-port with a 100-pin connector or the RJ11 connector
- Hostess *550* 8-port with a 100-pin connector or the RJ11 connector
- Hostess 16-port

Typically, you receive this *Programming Guide* if you did not order a Comtrol device driver with the controller.

If you plan on using a Hostess or Hostess *550* 2-port controller in COM mode, use the documentation that came with the controller to configure and install the controller.

If you ordered a Comtrol device driver, use the documentation that came with the device driver to configure and install the controller.

Purpose

This *Programming Guide* explains configuration, controller installation, and programming issues for Hostess series and Hostess *550* series controllers.

Audience

The primary audience for this *Programming Guide* includes people who plan to program using the Hostess or Hostess *550* series. This *Programming Guide* assumes that you are somewhat familiar with installing equipment in your computer.

The secondary audience includes people who require additional information about the Hostess or Hostess *550* series.

Prerequisites

The following are required to install Hostess series or Hostess *550* series controllers:

- An applicable Comtrol controller (see the *Scope* discussion)
- Optionally, an interface box (depending upon the controller model you purchased)
- Cabling for peripheral devices

For information about configuring and installing the interface box, see the *Interface Reference Card* that came with the interface box.

Organization

This manual is organized in the following sections:

Section 1. Installation Overview
Discusses general configuration and installation issues.

Section 2. Configuring 2-Port Controllers
Provides you with specific configuration and installation information about 2-port controllers.

Section 3. Configuring 4-Port and 8-Port Controllers
Provides you with specific configuration and installation information about 4-port and 8-port controllers.

Section 4. Configuring 16-Port Controllers
Provides you with specific configuration and installation information about 16-port controllers.

Section 5. Technical Overview
Provides you with specific information about accessible registers for each controller type.

Appendix A.Troubleshooting and Technical Support
Provides you with troubleshooting and technical support information for your controller.

Appendix B.Warranty
Discusses the warranty for Hostess series and Hostess *550* series controllers.

Bibliography

Data Transmission Circuits Line Drivers, Receivers, Transceivers, UARTs. Texas Instruments: U.S.A., 1993.

Table of Contents

Table of Contents

List of Figures, Flowcharts, and Tables

Figures

Flowcharts

Tables

Section 1. Installation Overview

1.1. Introduction

This section provides you with an overview of technical information about Hostess series or Hostess *550* series controllers that may affect how you configure the controller. This information includes:

- Selecting I/O base addresses.
- Setting the hardware interrupt and using the mask and poll registers.
- Daisy-chaining controllers together to share interrupts.

If you know what I/O address and interrupt (IRQ) you want to use, go to one of the following subsections to configure your controller:

- Section 2 for 2-port controllers
- Section 3 for 4-port or 8-port controllers
- Section 4 for 16-port controllers

1.2. Introduction to Base I/O Port Addresses

Choosing an I/O port base address varies for each particular application. Your operating system may require a specific base I/O address or it may allow you to select your own base I/O address. Table 1-1 shows the default base I/O addresses for the Hostess or Hostess *550* series and the name of the switch on the controller.

Table 1-1. Default Base I/O Addresses

Controller Type	Address in Hexadecimal	Switch Identification
2-Port	0280	ADDRESS
4-Port	02E0	ADDRESS* or SW1 (RJ45 model)
8-Port	02C0	ADDRESS* or ADDRESS SELECT (RJ11 model)
16-Port	0500	S1

* Older 4-port and 8-port models are labeled SW1.

Base I/O addresses have the following characteristics:

- The 2-port controllers use nine address bits that correspond to binary bit values. All other controllers use eight address bits that correspond to binary bit values.
- They are four-digit hexadecimal (hex) values.
- To determine the switch setting, you may need to translate the hex value into a binary value.

  Note: *A bit which equals 0 means that the corresponding switch position is ON. A bit which equals 1 is OFF.*

The following list provides information that you may need to determine the base I/O address for the controller:

- Each I/O port of the controller requires eight consecutive addresses (see Table 1-2).

Table 1-2. Required I/O Address Block Sizes

Controller Type	Consecutive Address Block Size
2-Port	16
4-Port	32
8-Port	64
16-Port	128

Note: *Refer to Table A-1 or an I/O address map in your computer's manual to determine which addresses are in use.*

- Each I/O address uses a four-digit hex value. For example, address 0280h contains:
  - The first digit of 0
  - The second digit of 2
  - The third digit of 8
  - The fourth digit of 0

  When you choose a base I/O address, the digit you choose must agree with the value parameters in Table 1-3.

Table 1-3. Hexadecimal Value Parameters

Controller Type	1st Digit	2nd Digit	3rd Digit	4th Digit
2-Port	0 or 1	0 through F	0 through F	0
4-Port	0 or 1	0 through F	0, 2, 4, 6, 8, A, C, or E	0
8-Port	0 or 1	0 through F	0, 4, 8, or C	0
16-Port	0 through 7	0 through F	0 or 8	0

Table 1-4 lists the equivalent hexadecimal and binary numbers.

Table 1-4. Hexadecimal and Binary Equivalents

Hex	Binary	Hex	Binary
0	0000	8	1000
1	0001	9	1001
2	0010	A	1010
3	0011	B	1011
4	0100	C	1100
5	0101	D	1101
6	0110	E	1110
7	0111	F	1111

Each controller model uses special rules in calculating the base I/O address. Use the following sections for specific calculation methods for each controller:

- Section 2 for 2-port controllers
- Section 3 for 4-port and 8-port controllers
- Section 4 for 16-port controllers

1.3. How to Determine an Individual Port's Address

To find the I/O address of a particular port, you must determine the offset from the base address. To determine the base address of a specific port, add the offset to the base address. The required block size of consecutive addresses for each controller is listed in Table 1-2.

Table 1-5. Hexadecimal Offsets

Port	Add base address plus hexadecimal offset	Hex Range
1	base address + 0h	0–7
2	base address + 8h	8–F
3	base address + 10h	10–17
4	base address + 18h	18–1F
5	base address + 20h	20–27
6	base address + 28h	28–2F
7	base address + 30h	30-37
8	base address + 38h	38–3F
9	base address + 40h	40–47
10	base address + 48h	48–4F
11	base address + 50h	50–57
12	base address + 58h	58–5F
13	base address + 60h	60-67
14	base address + 68h	68–6F
15	base address + 70h	70–77
16	base address + 78h	78–7F

For example,

- The base address of port 1 is equal to the base address of the controller.
- The base address of the second port is the base address plus 8h.
- The base address of the ninth port is the base address plus 40h and so forth. If the base address was 280h, then:

  Port 1 + 0 = 280
  Port 2 + 8 = 288
  Port 9 + 40 = 2C0

1.4. Introduction to Hardware Interrupts

Hostess series and Hostess *550* series controllers support interrupt driven serial communications. Each I/O port on a controller uses a shared (or common) interrupt line to the CPU.

You should determine which interrupt lines are already being used by your computer or other devices. Consult your computer's technical reference manual or other references concerning your peripherals. Once you know which interrupt line is free, you can use it for the controller. Each controller uses a single interrupt line except when you use the 2-port models in COM mode. COM mode supports different IRQs for each port.

The following table lists the default IRQ setting and switch name for each controller type.

Table 1-6. Default IRQ Settings

Controller Type	Default IRQ	Switch Name
2-Port	3	INTERRUPT
4-Port	3	INTERRUPT * or SW2 (RJ45 model)
8-Port	3	INTERRUPT * or IRQ SELECT (RJ11 model)
16-Port	3	S2

* Older 4-port and 8-port models are labeled SW2.

Interrupts can be further controlled by using the mask and poll registers on the controller.

- The mask register is an 8-bit register that allows you to disable interrupts for each port individually or in any combination.

  Any port interrupt can be masked or disabled by writing a 0 to the corresponding bit in the mask register. The interrupt is enabled by writing a 1.

  If position 1 of the interrupt switch is ON, the mask register is enabled. This allows you to individually mask the interrupts received from the I/O ports.

  If position 1 is OFF, the mask register is disabled and the interrupts can not be masked individually.

- A poll register is an 8-bit register that allows you to determine which ports have interrupts pending.

  Reading a 1 in the poll register indicates that an interrupt is pending on the port corresponding to the bit set in the poll register.

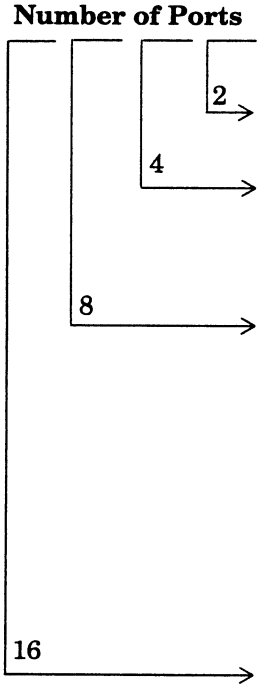
Mask and poll registers may be accessed at:

- I/O BASE + 7 hex (for 2-, 4-, and 8-port controllers)
- I/O BASE + 7 hex and I/O BASE + 47 hex (for 16-port controllers)

The mask register is a write-only register and the poll register is a read-only register. The mask/poll register for ports 1 through 8 is located at I/O BASE + 7h which is the scratch register. Therefore, the scratch register for port 1 is unavailable. The mask/poll register for ports 9 through 16 are located at I/O BASE + 47h which is the scratch register. Therefore, the scratch register for ports 1 and 8 are unavailable for the 16-port controller.

Tables 1-7 and 1-8 show the bit assignments for each port.

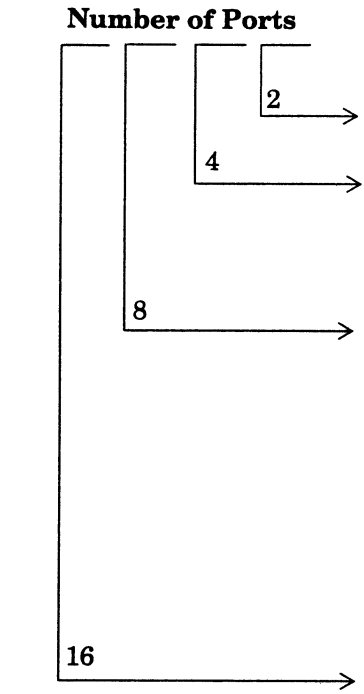
Table 1-7. Bit Assignments for the Mask Register



Mask Register WRITE Base Address +7h			
Bit 0	Port 1	0 Disables	1 Enables
Bit 1	Port 2	0 Disables	1 Enables
Bit 2	Port 3	0 Disables	1 Enables
Bit 3	Port 4	0 Disables	1 Enables
Bit 4	Port 5	0 Disables	1 Enables
Bit 5	Port 6	0 Disables	1 Enables
Bit 6	Port 7	0 Disables	1 Enables
Bit 7	Port 8	0 Disables	1 Enables
Mask Register WRITE Base Address +47h			
Bit 0	Port 9	0 Disables	1 Enables
Bit 1	Port 10	0 Disables	1 Enables
Bit 2	Port 11	0 Disables	1 Enables
Bit 3	Port 12	0 Disables	1 Enables
Bit 4	Port 13	0 Disables	1 Enables
Bit 5	Port 14	0 Disables	1 Enables
Bit 6	Port 15	0 Disables	1 Enables
Bit 7	Port 16	0 Disables	1 Enables

Note: *Switch 1 of the INTERRUPT, IRQ SELECT, SW2, or S2 enables or disables the mask register.*

Table 1-8. Bit Assignments for the Poll Register



Poll Register READ Base Address +7h			
Bit 0	Port 1	0 Clear	1 Pending
Bit 1	Port 2	0 Clear	1 Pending
Bit 2	Port 3	0 Clear	1 Pending
Bit 3	Port 4	0 Clear	1 Pending
Bit 4	Port 5	0 Clear	1 Pending
Bit 5	Port 6	0 Clear	1 Pending
Bit 6	Port 7	0 Clear	1 Pending
Bit 7	Port 8	0 Clear	1 Pending
Poll Register READ Base Address +47h			
Bit 0	Port 9	0 Clear	1 Pending
Bit 1	Port 10	0 Clear	1 Pending
Bit 2	Port 11	0 Clear	1 Pending
Bit 3	Port 12	0 Clear	1 Pending
Bit 4	Port 13	0 Clear	1 Pending
Bit 5	Port 14	0 Clear	1 Pending
Bit 6	Port 15	0 Clear	1 Pending
Bit 7	Port 16	0 Clear	1 Pending

Note: *Switch 1 of the INTERRUPT, IRQ SELECT, SW2, or S2 enables or disables the mask register.*

1.5. Daisy-Chaining 4-, 8-, and 16-Port Controllers

If you have more than one controller installed in your system, you can choose a different interrupt line for each controller by making a different selection on each INTERRUPT, IRQ SELECT, SW2, or S2 switch.

Note: *You cannot daisy-chain 2-port controllers.*

However, if you want to use the same interrupt line for all of the controllers in your system, perform the following steps:

1. Enable the interrupt switch on only one controller.
2. Connect the controllers together using daisy-chain connections (see Figures 1-1 or 1-2). The daisy-chain connections are made from ribbon cable, available on request from Comtrol. You can connect up to four Hostess series or Hostess *550* series controllers (excluding the 2-port model) together in this manner.

Note: *Only the first controller in the daisy chain configuration has an interrupt enabled.*

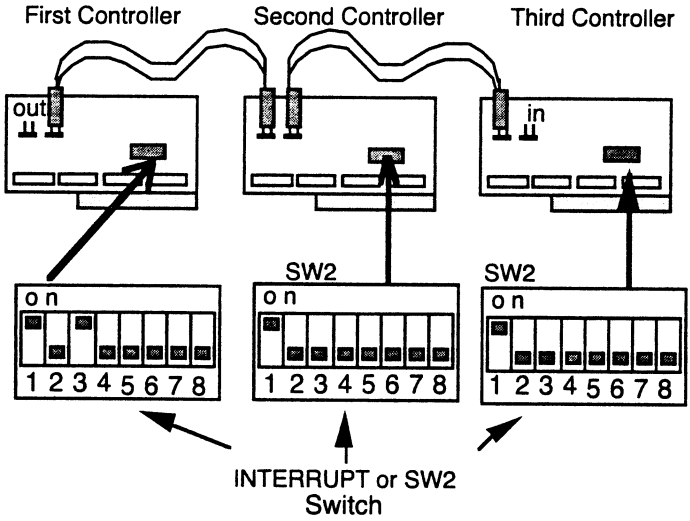


Figure 1-1. Daisy-Chaining Three Hostess or Hostess*550* Controllers (4- or 8-Port)

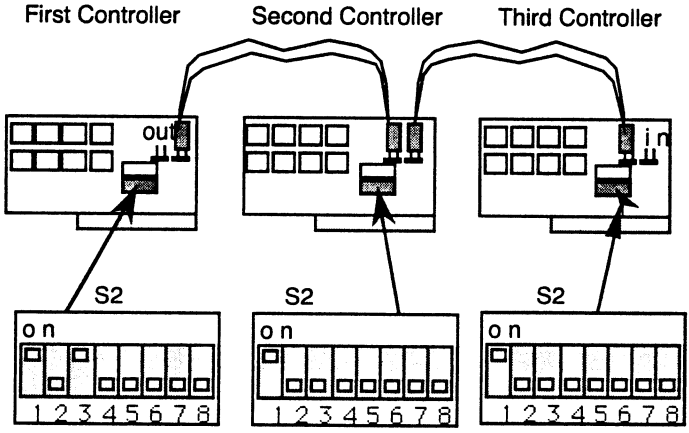


Figure 1-2. Daisy-Chaining Three Hostess or Hostess *550* Controllers (16-Port)

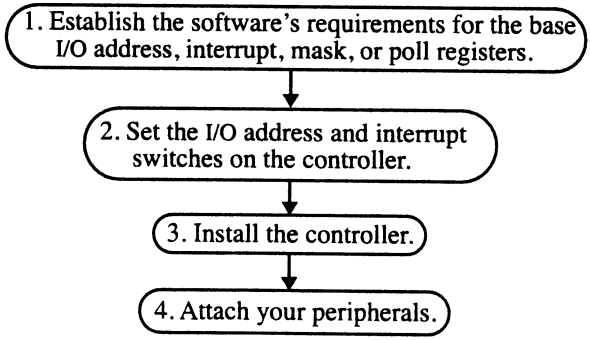
Section 2. Configuring and Installing 2-Port Controllers

If you want to use the controller in COM mode, use the documentation that came with your controller to set the MODE SELECT, ADDRESS, and INTERRUPT switches.

If you plan on using a Comtrol device driver, use the documentation that came with the device driver to configure and install the controller or controllers.

2.1. Installation Overview

Use the following flowchart as a guide to installing your Hostess or Hostess *550* 2-port controller.



Flowchart 2-1. Installing 2-Port Controllers

2.2. Setting the MODE SELECT and ADDRESS Switches

The MODE SELECT switch sets the controller to Hostess mode or COM mode. Figure 2-1 shows how to set the MODE SELECT switch for Hostess mode.

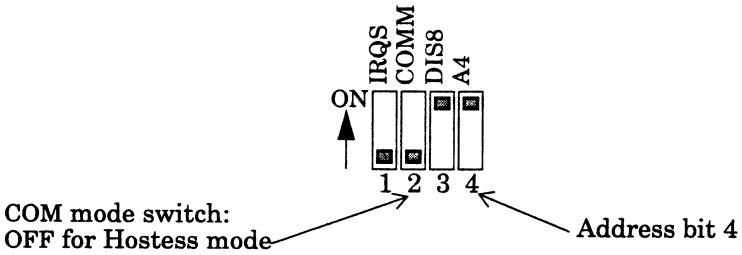


Figure 2-1. MODE SELECT Switch Settings

Use Table 2-1 to set the ADDRESS switch for your system and software. If you want to use an address that is not illustrated in the table, use Figure 2-2 to calculate your ADDRESS switch setting.

Table 2-1. Hostess Mode Base I/O Addresses

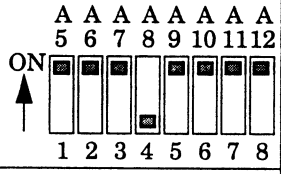
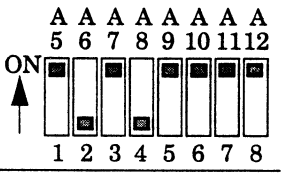
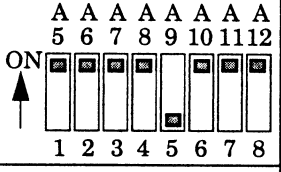
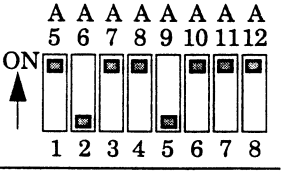
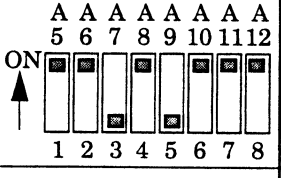
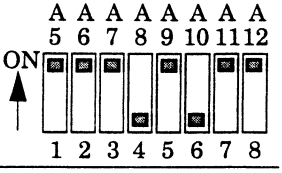
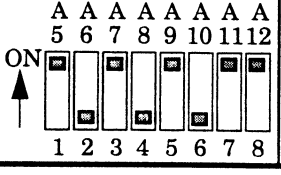
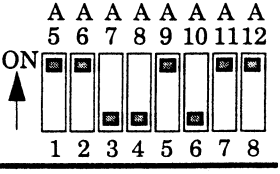
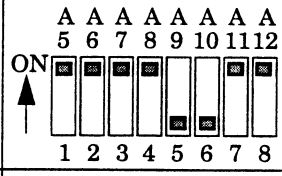
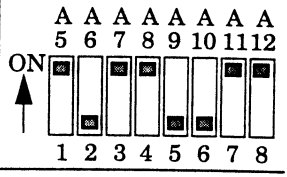
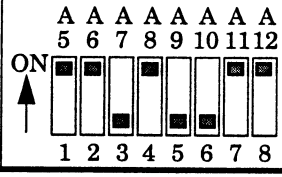
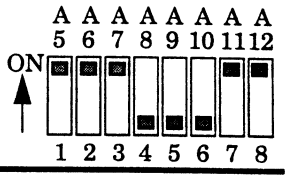
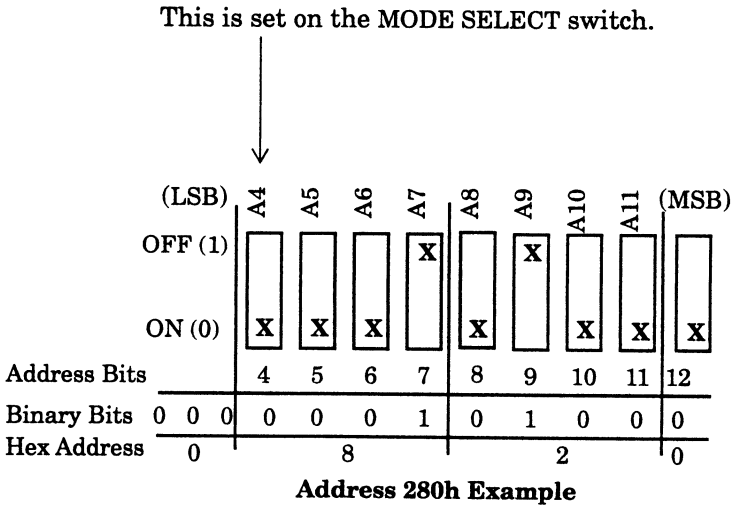
Base I/O Address	ADDRESS Switch	Base I/O Address	ADDRESS Switch
100h		140h	
200h		240h	
280h		500h	
540h		580h	

Table 2-1. Hostess Mode Base I/O Addresses (Continued)

Base I/O Address	ADDRESS Switch	Base I/O Address	ADDRESS Switch
600h		640h	
680h		700h	

You can use the following method to calculate the switch settings, if you require a base I/O address that is not listed in the table.



Address 280h Example

Figure 2-2. Calculating Additional 2-Port I/O Addresses

2.3. Setting the Interrupt

Use Table 2-2 to set the interrupt on the controller for your system and software. You can only set the controller for one interrupt in Hostess mode. The table illustrates interrupt settings for enabling or disabling the mask register.

Table 2-2. IRQ Settings - Hostess Mode

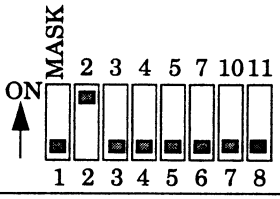
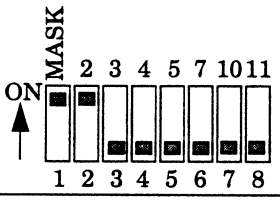
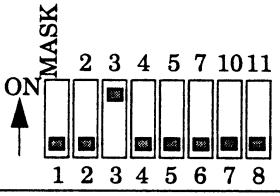
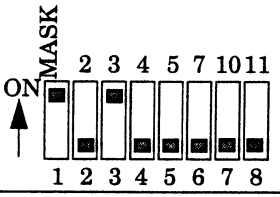
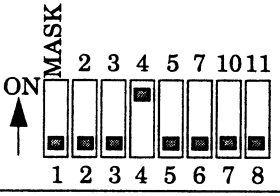
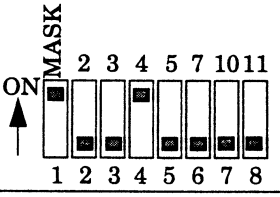
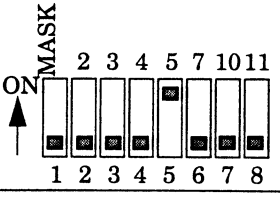
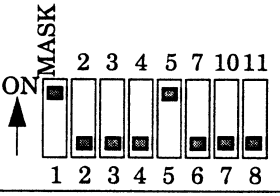
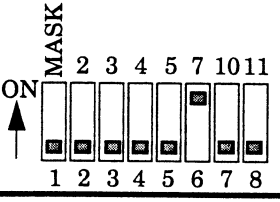
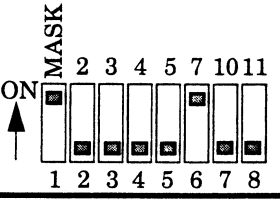
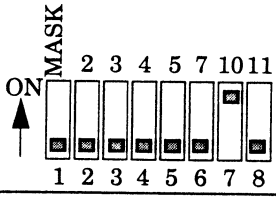
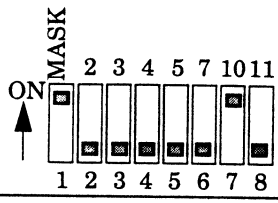
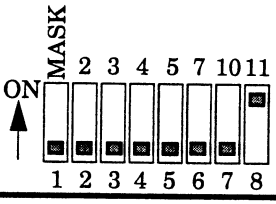
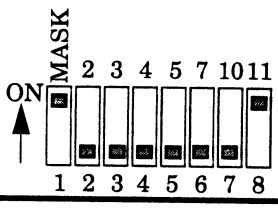
IRQ	INTERRUPT Switch (Mask Off)	INTERRUPT Switch (Mask On)
2		
3		
4		
5		
7		

Table 2-2. IRQ Settings - Hostess Mode (Continued)

IRQ	INTERRUPT Switch (Mask Off)	INTERRUPT Switch (Mask On)
10		
11		

2.4. Installing the Controller

If you have not done so already, set the switches on the controller. If the switches have already been set, use the following steps to install the controller.

Warning *Static electricity may damage the controller. When touching and installing the controller, wear a grounding strap. Hold the controller only by its edges or the mounting bracket.*

1. Turn the power switch for the system unit to the OFF position.
2. Remove the system unit cover.
3. Select a slot to install the controller.
4. Remove the expansion slot cover.
5. Insert the controller in the expansion slot, making sure that it is properly seated.
6. Attach the controller to the chassis with the expansion slot screw. Repeat steps 3 through 5 for each controller.
7. Replace the cover on the system unit.

Once the controller or controllers are installed, you can connect your peripherals.

Section 3. Configuring and Installing 4-Port and 8-Port Controllers

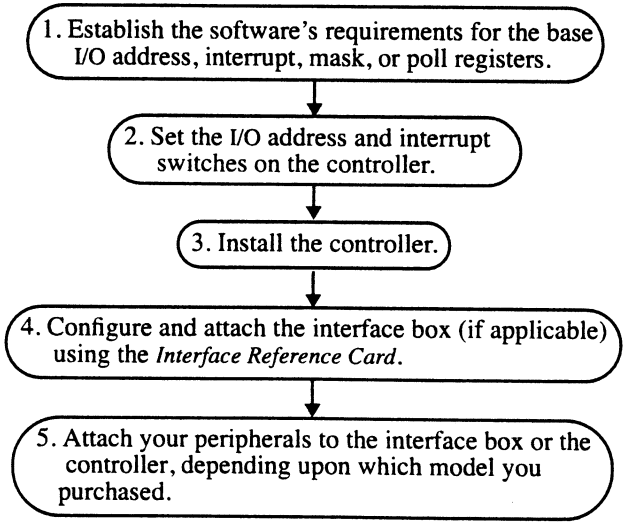
This section shows you how to set the I/O address and interrupt switches for the following controllers:

- Hostess or Hostess *550* 4- and 8-port series with a 100-pin connector
- Hostess or Hostess *550* 4-port series with an RJ45 connector
- Hostess or Hostess *550* 8-port series with an RJ11 connector

If you plan on using a Comtrol device driver, use the documentation that came with it to configure and install the controller.

3.1. Installation Overview

The following flowchart shows the steps to install a Hostess or Hostess *550* controller.



Flowchart 3-1. Installing 4- or 8-Port Controllers

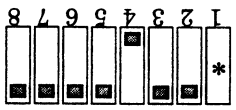
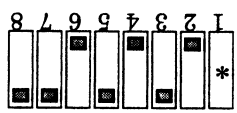
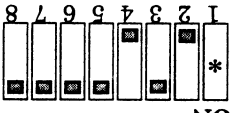
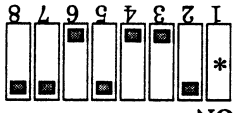
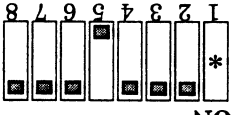
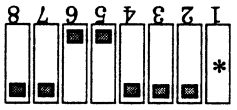
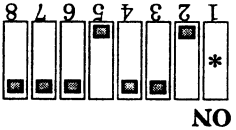
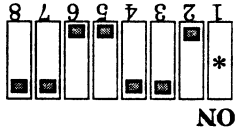
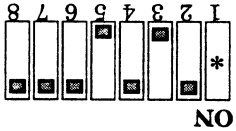
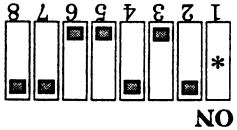
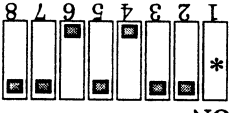
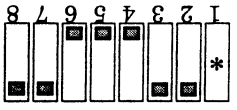
3.2. Setting the ADDRESS or SW1 Switch

Use the following to set the ADDRESS or SW1 switch for your system and software.

- Table 3-1 for 4-port and 8-port controllers with a 100-pin connector.
- Table 3-2 for 4-port controllers with the RJ45 connector.
- Table 3-3 for 8-port controllers with the RJ11 connector.

If you want to use an I/O address that is not illustrated in the following tables, you can calculate an address using the figures after the tables.

Table 3-1. Base I/O Addresses (4- and 8-Port)

Address	ADDRESS Switch	Address	ADDRESS Switch
100h		540h	
140h		580h	
200h		600h	
240h		640h	
280h		680h	
500h		700h	

Note: *Switch 1 should be turned ON for a 4-port controller and OFF for an 8-port controller.*

Table 3-2. RJ45 Base I/O Addresses (4-Port Only)

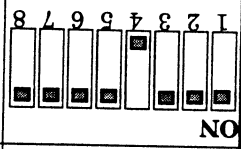
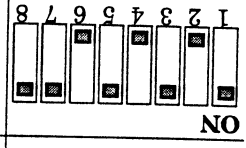
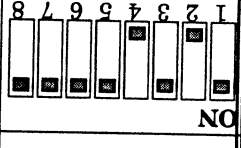
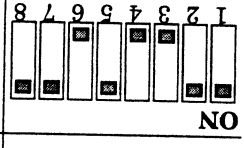
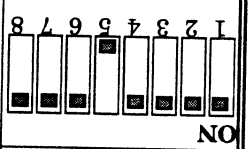
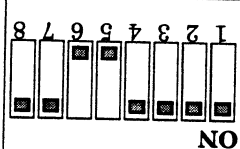
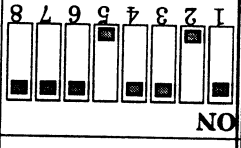
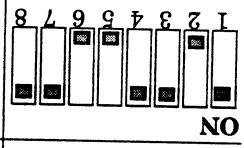
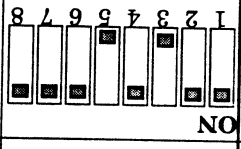
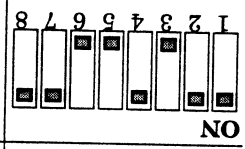
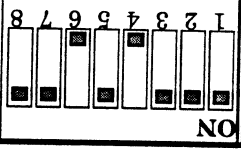
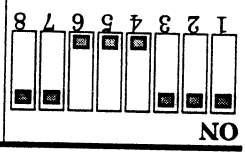
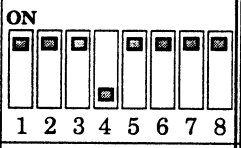
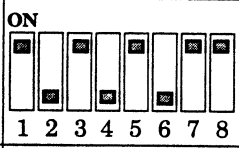
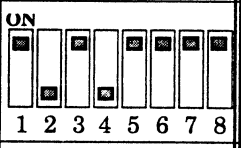
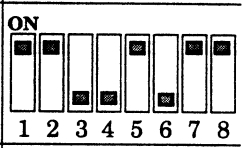
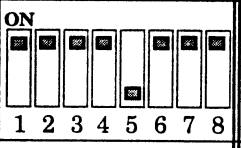
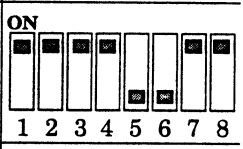
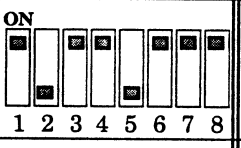
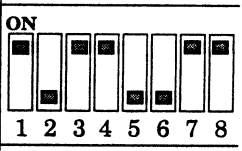
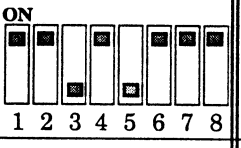
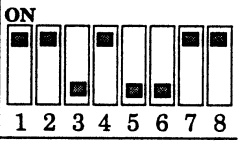
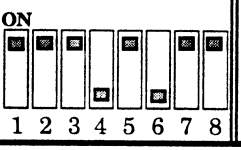
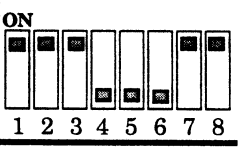
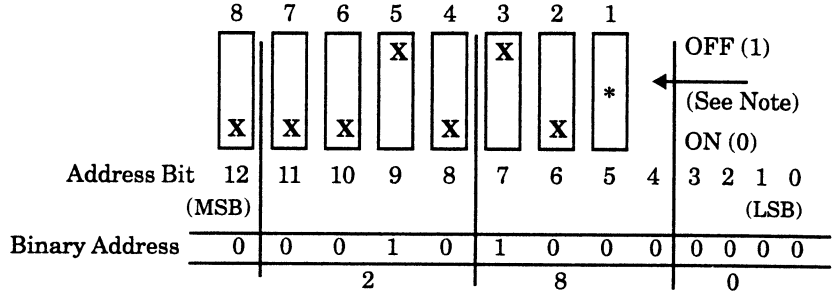
Address	SW1 Switch	Address	SW1 Switch
100h		540h	
140h		580h	
200h		600h	
240h		640h	
280h		680h	
500h		700h	

Table 3-3. RJ11 Base I/O Addresses (8-Port Only)

Address	ADDRESS SELECT Switch	Address	ADDRESS SELECT Switch
100h		540h	
140h		580h	
200h		600h	
240h		640h	
280h		680h	
500h		700h	

You can use the following method to calculate the switch settings if you require a base I/O address that is not listed in the previous tables.



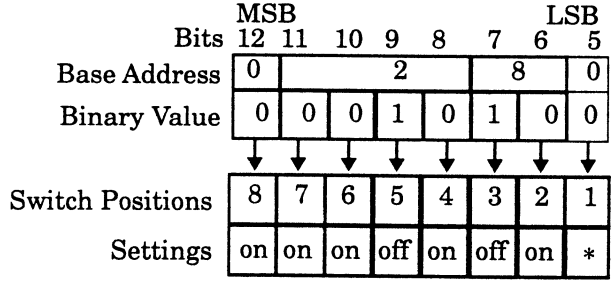
Note: *Switch 1 must be off for an 8-port controller with the 100-pin connector (it is not decoded). You must decode switch 1 for any 4-port controller.*

Figure 3-1. Calculating 4- or 8-Port I/O Addresses

The following example shows how to determine the switch setting for the address you chose, and demonstrates how the binary equivalent corresponds to positions eight through one of the switch. The following example uses 0280h address.

- Translate the 0280h base address into four binary values. (Each hex value has four bits.)
- Switch positions eight through two correspond to binary bit values. Position one is ON or off for a 4-port controller (depending upon the base I/O address that you chose) and OFF for an 8-port controller.

  The following figure shows that the switch reads bits 12 through 5. The last bit of the binary value 0000 for 0 of the 0280h address corresponds to switch 8, which is ON. The four bits of 2 correspond to positions 7, 6, 5, and 4, in which 7, 6, and 4 are ON and position 5 is OFF.
- The first three digits of the binary value 1000 are equivalent to 8 of the address 0280h correspond to positions 3, 2, and 1, in which position 3 is OFF and positions 2 and 1 are ON. The last binary value of 0000 is equivalent to 0 of address 0280h, does not affect the position settings.

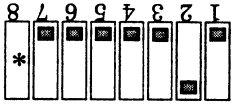
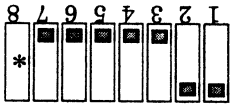
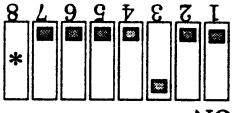
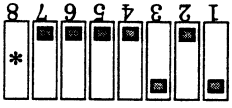
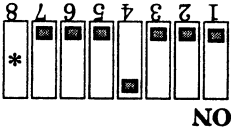
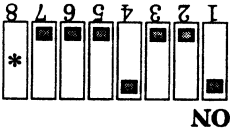
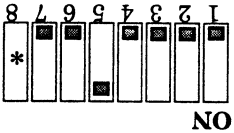
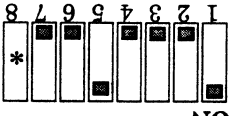
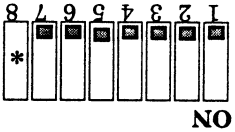
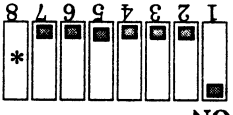
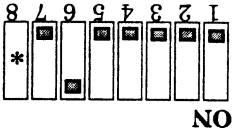
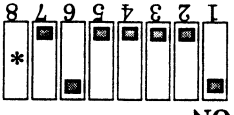


* *Position 1 can be set to ON or OFF for a 4-port controller with a 100-pin connector (depending on the selected base I/O address and OFF for an 8-port controller.*

3.3. Setting the Interrupt

Use Tables 3-4, 3-5, or 3-6 to set the interrupt on the controller for your system and software. You can only set one interrupt per controller.

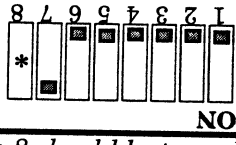
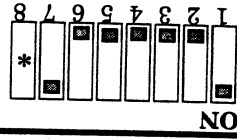
Table 3-4. IRQ Settings (4- and 8-Port)

IRQ	INTERRUPT Switch (Mask Enable OFF)	INTERRUPT Switch (Mask Enable ON)
2		
3		
4		
5		
7**		
10		

* *Switch 8 should be turned ON for a 4-port controller and OFF for an 8-port controller.*

** *You must also move the jumper on JP2 to pins 2 and 3 (see the* Hardware Reference Card *for pin locations). Interrupt 7 is only available on models whose switch is labeled INTERRUPT.*

Table 3-4. IRQ Settings (4- and 8-Port) (Continued)

IRQ	INTERRUPT Switch (Mask Enable OFF)	INTERRUPT Switch (Mask Enable ON)
11		

* *Switch 8 should be turned ON for a 4-port controller and OFF for an 8-port controller.*

** *You must also move the jumper on JP2 to pins 2 and 3 (see the* Hardware Reference Card *for pin locations). Interrupt 7 is only available on models whose switch is labeled INTERRUPT.*

Table 3-5. RJ45 IRQ Settings (4-Port Only)

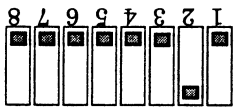
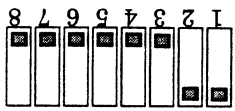
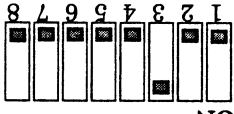
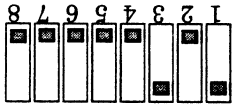
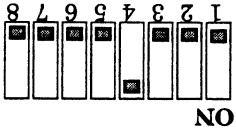
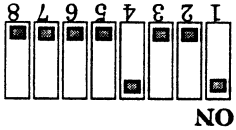
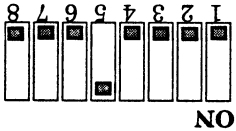
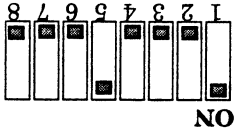
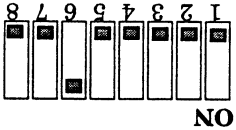
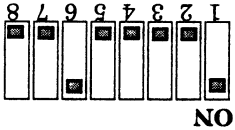
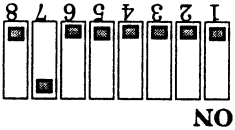
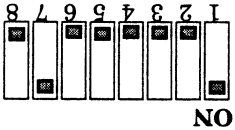
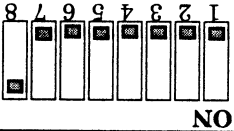
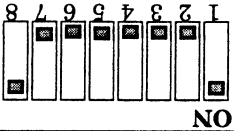
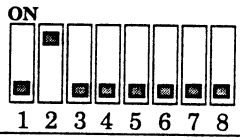
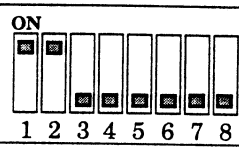
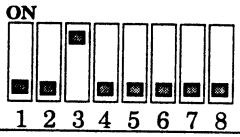
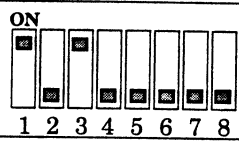
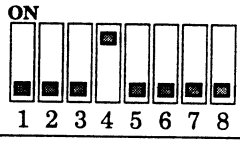
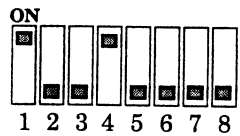
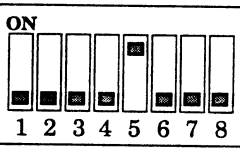
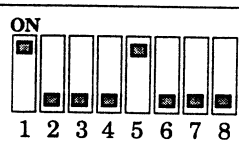
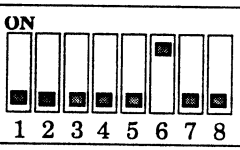
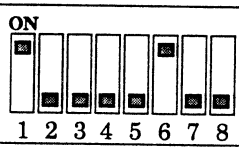
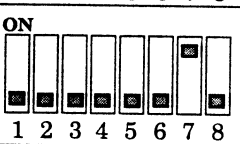
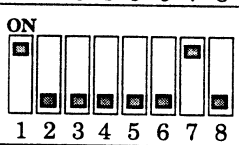
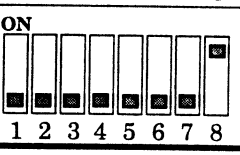
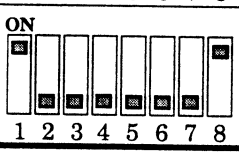
IRQ	SW2 Switch (Mask Enable OFF)	SW2 Switch (Mask Enable ON)
2	 NO	 NO
3	 NO	 NO
4	 NO	 NO
5	 NO	 NO
10	 NO	 NO
11	 NO	 NO
12	 NO	 NO

Table 3-6. RJ11 IRQ Settings (8-Port Only)

IRQ	IRQ SELECT Switch (Mask Enable OFF)	IRQ SELECT Switch (Mask Enable ON)
2		
3		
4		
5		
7		
10		
11		

3.4. Installing the Controller

If you have not done so already, set the switches on the controller. If the switches have already been set, use the following steps to install the controller.

Warning: *Static electricity may destroy the controller. When touching or installing the controller, wear a grounding strap. Hold the controller only by its edges or the mounting bracket.*

1. Turn the power switch for the system unit to the OFF position.
2. Remove the system unit cover.
3. Select a slot to install the controller.
4. Remove the expansion slot cover.
5. Insert the controller in the expansion slot, making sure that it is properly seated.
6. Attach the controller to the chassis with the expansion slot screw. Repeat steps 3 through 5 for each controller.
7. Replace the cover on the system unit.

Once the 100-pin controller or controllers are installed you can attach the interface box. See the *Interface Reference Card* if you need information about configuring the interface box. After the interface box is connected, you can connect your peripherals.

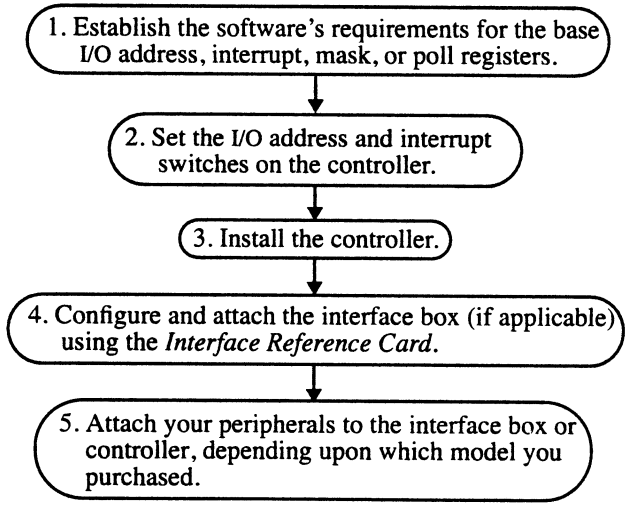
If you installed an RJ45 or RJ11 model controller, you are ready to attach your peripherals.

Section 4. Configuring and Installing 16-Port Controllers

If you plan on using a Comtrol device driver, use the documentation that came with it to configure and install the controller.

4.1. Installation Overview

Use the following flowchart as a guide to installing your Hostess *550* controller or controllers.

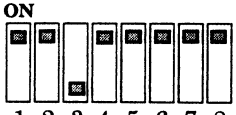
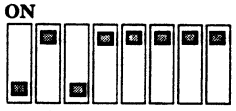
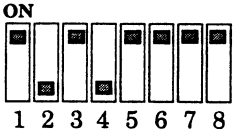
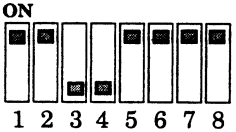


Flowchart 4-1. Installing Hostess *550* 16-Port Controllers

4.2. Setting the S1 Switch

Use Table 4-1 to set the S1 I/O address switch for your system and software. If you want to use an I/O address that is not illustrated in the table, use Figures 4-1 and 4-2 (after the table).

Table 4-1. Common Base I/O Addresses

Base I/O Address	S1 Switch	Base I/O Address	S1 Switch
200h		280h	
500h		580h	
600h		680h	
700h			

You can use the following method to calculate the switch settings if you require a base I/O address that is not listed in the table.

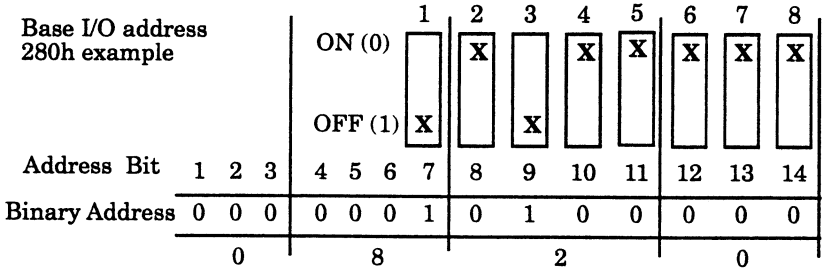


Figure 4-1. Calculating Additional I/O Addresses

The following example shows how to determine the S1 switch settings for the address you chose, and demonstrates how the binary equivalent corresponds to positions eight through one of S1. This example uses the 0280h address.

- Translate the 0280h base address into four binary values. Use Figure 4-1 if you need help determining these values.
- Positions eight through one of switch S1 correspond to binary bit values. A bit that equals 0 corresponds to an ON switch position. A bit that equals 1 is set to OFF.

- Figure 4-2 shows that S1 switch reads bits 7 through 14. The last three bits of the binary value 000 for the hexadecimal value 0 correspond to switches 6, 7, and 8.

  Note: *Read the address from right to left.*

- The four bits of 2 correspond to positions 2, 3, 4, and 5.
- The first bit of 8 corresponds to position 1. The last binary value is equivalent to 0 of address 0280h and does not affect the position settings.
- The switch positions for the address 0280h are: positions 2, 4, 5, 6, 7, and 8 are set to ON, positions 1 and 3 are set to OFF.

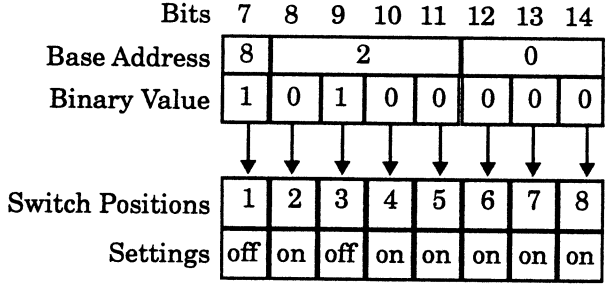


Figure 4-2. Example of Setting the I/O Address

4.3. Setting the Interrupt

Use Table 4-2 to set the S2 interrupt switch on the controller for your system and software. You can only set one interrupt for the controller.

Table 4-2. IRQ Settings

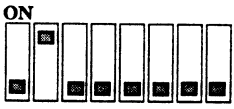
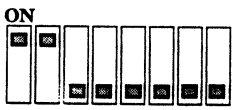
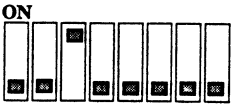
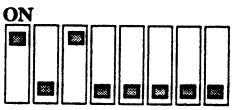
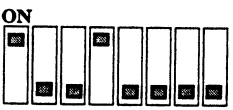
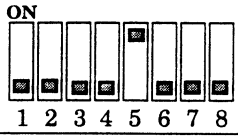
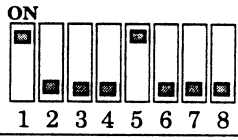
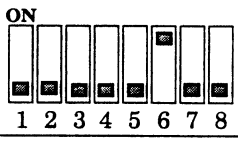
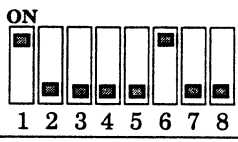
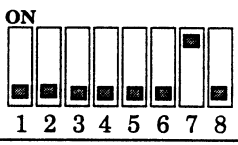
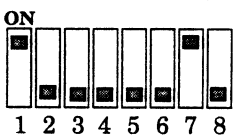
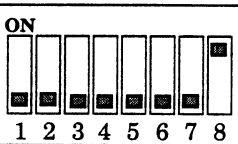
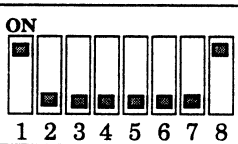
IRQ	S2 Switch (Mask Enable OFF)	S2 Switch (Mask Enable ON)
2		
3		
4		

Table 4-2. IRQ Settings (Continued)

IRQ	S2 Switch (Mask Enable OFF)	S2 Switch (Mask Enable ON)
5		
10		
11		
12		

4.4. Installing the Controller

If you have not done so already, set the switches on the controller. If the switches have already been set, use the following steps to install the controller.

Warning: *Static electricity may destroy the controller. When touching or installing the controller, wear a grounding strap. Hold the controller only by its edges or the mounting bracket.*

1. Turn the power switch for the system unit to the OFF position.
2. Remove the system unit cover.
3. Select a slot to install the controller.
4. Remove the expansion slot cover.
5. Insert the controller in the expansion slot, making sure that it is properly seated.
6. Attach the controller to the chassis with the expansion slot screw. Repeat steps 3 through 5 for each controller.
7. Replace the cover on the system unit.

Once the controller or controllers are installed, refer to your *Interface Reference Card* to attach the interface box.

After the interface box is connected, you can connect your peripherals to the interface box.

Section 5. Technical Overview

5.1. Introduction

This section provides you with an overview of technical information about Hostess series and Hostess *550* series controllers. This section is divided into several subsections that discuss the following topics:

- An overview that discusses Asynchronous Communications Elements (ACEs). This is a general subsection that is applicable to all ACE types.
- Principles of operation of the 16450 UART (see Subsection 5.3).
- Principles of operation of the 16550 UART (see Subsection 5.4).
- Principles of operation of the 16552 DUART (see Subsection 5.5).

5.2. ACE Overview

This series of Comtrol controllers use one of two Asynchronous Communications Elements (ACEs). The ACE is often referred to as a Universal Asynchronous Receiver/transmitter (UART) or a Dual Universal Asynchronous Receiver/transmitter (DUART). Table 5-1 lists the controller and ACE types.

Table 5-1. Controller and ACE Types

Controller Type	Number of Ports	ACE Type
Hostess	2	16450 (UART)
Hostess *550*	2	16550 (UART)
Hostess	4	16450 (UART)
Hostess *550*	4	16550 (UART)
Hostess	8	16450 (UART)
Hostess *550*	8	16550 (UART)
Hostess *550*	16	16552 (DUART)

The ACE performs serial to parallel conversion on data characters received from a peripheral device or modem, and parallel to serial conversion of data characters received from the CPU. The CPU can read the complete status of the ACE at any time during the functional operation. Status information reported includes the type and condition of the transfer operations being performed by the ACE, as well as any error conditions (parity overrun, framing, or break interrupt).

The 16550 UART and 16552 DUART includes a programmable baud generator that is capable of dividing the timing reference clock input by divisors of:

1 to 2^{16} - 1

In addition, the 16550 UART and 16552 DUART can be put into another mode, called the First-in, First-out (FIFO) mode. This mode relieves the CPU of excessive software overhead. Internal FIFOs are activated which allow 16 bytes (plus three bits per byte of error data in the RCVR FIFO) to be stored in both receive and transmit modes. This buffering mode reduces the number of interrupts presented to the CPU.

For detailed information about a specific ACE, see the appropriate subsection.

5.3. Principles of Operation of the 16450 UART

This complete subsection is an excerpt of the *Data Transmission Circuits Data Book: Line Drivers, Receivers, Transceivers, UARTs* provided by Texas Instruments (TI). If you want more information about the 16450 UART, contact your nearest TI Field Sales Office, local authorized TI distributor, or by writing directly to:

Texas Instruments Incorporated
LITERATURE RESPONSE CENTER
P.O. Box 809066
Dallas, Texas 75380-9066

You can also call the TI Literature Response number at: (800) 477-8924

5.3.1. Accessible Register Overview

Table 5-2 summarizes the ACE registers that you can access and control. These registers are used to control ACE operations, receive data, and transmit data.

The following subsections discuss the following topics:

- Accessible register overview
- Receiver buffer register (RBR)
- Transmitter holding register (THR)
- Interrupt enable register (IER)
- Interrupt identification register (IIR)
- Line control register (LCR)
- Modem control register (MCR)
- Line status register (LSR)
- Modem status register (MSR)
- Scratch register (SCR)
- Programmable baud generator

Table 5-2. Register Summary for the 16450 UART

Bit Number	Register Address				
	0 DLAB=0	0 DLAB=0	1 DLAB=0	2	3
	Receive Buffer Register (RBR) Read-only	Transmit Holding Register (THR) Write-only	Interrupt Enable Register (IER)	Interrupt ID Register (IIR) Read-only	Line Control Register (LCR)
0	data bit 0*	data bit 0	enable receive data available interrupt (ERBF)	0 if interrupt pending	word length select bit 0 (WLS0)
1	data bit 1	data bit 1	enable transmitter holding register empty interrupt (ETBE)	interrupt ID bit(0)	word length select bit 1 (WLS1)
2	data bit 2	data bit 2	enable receive line status interrupt (ELSI)	interrupt ID bit(1)	Number of stop bits (STB)
3	data bit 3	data bit 3	enable modem status interrupt (EDSSI)	0	parity enable (PEN)
4	data bit 4	data bit 4	0	0	even parity select (EPS)
5	data bit 5	data bit 5	0	0	stick parity
6	data bit 6	data bit 6	0	0	set break
7	data bit 7	data bit 7	0	0	divisor latch access bit (DLAB)

** Bit 0 is the least significant bit. It is the first bit serially transmitted or received.*

Source: Texas Instruments

Continued

Table 5-2. Register Summary for the 16450 UART (cont.)

Bit Number	Register Address					
	4	5	6	7	0 DLAB=1	1 DLAB=0
	Modem Control Register (MCR)	Line Status Register (LSR)	Modem Status Register (MSR)	Scratch Register (SCR)	Divisor Latch (DDL) LSB	Latch (DLM) MSB
0	data terminal ready (DTR)	data ready (DR)	delta clear to send (DCTS)	bit 0	bit 0	bit8
1	request to send (RTS)	overrun error (OE)	delta data set ready (DDSR)	bit 1	bit 1	bit 9
2	out 1	parity error (PE)	trailing edge ring indicator (TERI)	bit 2	bit 2	bit 10
3	out 2	framing error (FE)	delta data carrier detect (DDCD)	bit 3	bit 3	bit 11
4	loop	break interrupt (BI)	clear to send (CTS)	bit 4	bit 4	bit 12
5	0	transmitter holding register (THRE)	data set ready (DSR)	bit 5	bit 5	bit 13
6	0	transmitter empty (TEMT)	ring indicator (RI)	bit 6	bit 6	bit 14
7	0	0	data carrier detect (DCD)	bit 7	bit 7	bit 15

** Bit 0 is the least significant bit (LSB). It is the first bit serially transmitted or received.*

Source: Texas Instruments

5.3.2. Receiver Buffer Register (RBR)

The ACE receiver section consists of a receiver shift register and a receiver buffer register. Timing is supplied by the 16 X receiver clock (RCLK). Receiver section control is a function of the ACE line control register.

The ACE receiver shift register receives serial data from the serial input (SIN) pin. The receiver shift register then converts the data to a parallel from and loads it into the receiver buffer register. When a character is placed in the receiver buffer register and the received data available interrupt is enabled, an interrupt is generated. This interrupt is cleared when the data is read out of the receiver buffer register.

5.3.3. Transmitter Holding Register (THR)

The ACE transmitter section consists of a transmitter holding register and a transmitter shift register. Timing is supplied by the Baud Out ($\overline{\text{BAUDOUT}}$) clock signal. Transmitter section control is a function of the ACE line control register.

The ACE transmitter holding register receives data off the internal data bus and, when the shift register is idle, moves it into the transmitter shift register. The transmitter shift register serializes the data and outputs it at the serial output (SOUT). If the transmitter holding register is empty and the transmitter holding register empty (THRE) interrupt is enabled, an interrupt is generated. This interrupt is cleared when a character is loaded into the register.

5.3.4. Interrupt Enable Register (IER)

The interrupt enable register enables each of the four types of interrupts (see Table 5-3) and INTRPT output signal in response to an interrupt generation. The interrupt enable register can also be used to disable the interrupt system by setting bit 0 through 3 to logic 0. Table 5-2 summarizes the contents of this register and the following describes it in detail:

Bit 0 — This bit, when set to logic 1, enables the received data available interrupt.

Bit 1 — This bit, when set logic 1, enables the transmitter holding register empty interrupt.

Bit 2 — This bit, when set to logic 1, enables the receiver line status interrupt.

Bit 3 — This bit, when set to logic 1, enables the modem status interrupt.

Bits 4 – 7 — Bits 4 through 7 in the interrupt enable register are not used and are always set to logic 0.

5.3.5. Interrupt Identification Register (IIR)

The ACE has an on-chip interrupt generation and prioritization capability that permits a flexible interface with most microprocessors.

The ACE provides four prioritized levels of interrupts:

Priority 1 Receiver line status (highest priority)

Priority 2 Receiver data ready

Priority 3 Transmitter holding register empty

Priority 4 Modem status (lowest priority)

When an interrupt is generated, the interrupt identification register indicates that an interrupt is pending and the type of that interrupt in its three least significant bits (bits 0, 1, and 2). Table 5-4 summarizes the contents of this register. Table 5-3 provides a detailed description of the interrupt control functions.

Bit 0 This bit can be used either in a hardwire prioritized or polled interrupt system. If this bit is logic 0, an interrupt is pending. When bit 0 is a logic 1, no interrupt is pending.

Bits 1 & 2 These two bits are used to identify the highest priority interrupt pending as indicated in Table 5-3.

Bits 3 – 7 Bits 3 through 7 in the interrupt identification register are not used and are always set to logic 0.

Table 5-3. Interrupt Control Functions

Interrupt Identification Register			Priority Level	Interrupt Type	Interrupt Source	Interrupt Reset Method
Bit 2	Bit 1	Bit 0				
0	0	1	None	None	None	Not applicable
1	1	0	1	Receiver line status	Overrun error, parity error, framing error or break interrupt	Reading the line status register
1	0	0	2	Receiver data available	Receiver data available	Reading the receiver buffer register

Table 5-3. Interrupt Control Functions (Continued)

Interrupt Identification Register			Priority Level	Interrupt Type	Interrupt Source	Interrupt Reset Method
Bit 2	Bit 1	Bit 0				
0	1	0	3	Transmitter holding register empty	Transmitter holding register empty	Reading the interrupt identification register (if source of interrupt) or writing into the transmitter holding register
0	0	4	Modem status	Clear to send, data set ready, ring indicator, or data carrier detect	Reading the modem status register	

5.3.6. Line Control Register (LCR)

You control the format of the asynchronous data communications exchange through the line control register. In addition, you are able to retrieve, inspect, and modify the contents of the line control register; this eliminates the need for separate storage of the line characteristics in system memory. Table 5-2 summarizes the contents of this register and the following discusses it in detail.

Bits 0 &1 These two bits specify the number of bits in each transmitted or received serial character. These bits are encoded as follows.

Bit 1	Bit 0	Word Length
0	0	5 bits
0	1	6 bits
1	0	7 bits
1	1	8 bits

Bit 2 This bit specifies either one, one and one-half, or two stop bits in each transmitted character. If bit 2 is a logic 0, one stop bit is generated in the data. If bit 2 is a logic 1, the number of stop bits generated is dependent on the word length selected with bits 0 and 1. The receiver checks the first stop bit only, regardless of the number of stop bits selected. The number of stop bits generated, in relation to word length and bit 2, is shown in the following table.

Bit 2	Word Length Selected by Bits 1 and 2	Number of Stop Bits Generated
0	Any word length	1
1	5 bits	1 1/2
1	6 bits	2
1	7 bits	2
1	8 bits	2

Bit 3 This bit is the parity enable bit. When bit 3 is logic 1, a parity bit is generated in transmitted data between the last data word bit and the first stop bit. In received data, if bit 3 is logic 1, parity is checked. When bit 3 is a logic 0, no parity is generated or checked.

Bit 4 Bit 4 is the even parity select bit. When parity is enabled by bit 3: a logic 1 in bit 4 produces even parity (an even number of logic is in the data and parity bits) and a logic 0 in bit 4 produces odd parity (an odd number of logic 1s).

Bit 5 This is the stick parity bit. When bits 3, 4, and 5 are logic 1s, the parity bit is transmitted and checked as a logic 0. When bits 3 and 5 are logic 1s and bit 4 is a logic 0, the parity bit is transmitted and checked as a logic 1.

Bit 6 This bit is the break control bit. Bit 6 is set to a logic 1 to force a break condition, that is, a condition where the serial output (SOUT) pin is forced to the spacing (logic 0) state. When bit 6 is set to a logic 0, the break condition is disabled. The break condition has no effect on the transmitter logic, it only effects the serial output.

Bit 7 This bit is the divisor latch access bit (DLAB) Bit 7 must be set to a logic 1 to access the divisor latches of the baud generator during a read or write. Bit 7 must be set to a logic 0 during a read or write to access the receiver buffer, the transmitter holding register, or the interrupt enable register.

5.3.7. Modem Control Register (MCR)

The modem control register is an 8-bit register that controls an interface with a modem, data set, or peripheral device that is emulating a modem. Table 5-2 summarizes the contents of this register and the following describes it in detail.

Bit 0 — Bit 0 (DTR) controls the data terminal ready ($\overline{DTR}$) output. Setting bit 0 to a logic 1, forces the $\overline{DTR}$ output to its active state (low). When bit 0 is set to a logic 0, $\overline{DTR}$ goes high.

Bit 1 — Bit 1 (RTS) controls the request to send ($\overline{RTS}$) output in a manner identical to Bit 0's control over the $\overline{DTR}$ output.

Bit 2 — Bit 2 (OUT 1) controls the output 1 ($\overline{OUT\ 1}$) signal, a user designated output signal, in a manner identical to Bit 0's control over the $\overline{DTR}$ output.

Bit 3 — Bit 3 (OUT 2) controls the output 2 ($\overline{OUT\ 2}$) signal, a user designated output signal, in a manner identical to Bit 0's control over the $\overline{DTR}$ output.

Bit 4 — Bit 4 provides a local loopback feature for diagnostic testing of the ACE. When this bit is set to a logic high, the following occurs:

1. The transmitter serial output (SOUT) is set high.
2. The receiver serial input (SIN) is disconnected.
3. The output of the transmitter shift register is looped back into the receiver shift register input.
4. The four modem control inputs ($\overline{CTS}$, $\overline{DSR}$, $\overline{DCD}$, and $\overline{RI}$) are disconnected.
5. The four modem control outputs ($\overline{DTR}$, $\overline{RTS}$, $\overline{OUT\ 1}$, and $\overline{OUT\ 2}$) are internally connected to the four modem control inputs.
6. The four modem control output pins are forced to their inactive states (high).

In diagnostic mode, data that is transmitted is immediately received. This allows the processor to verify the transmit- and receive-data paths to the ACE. The receiver and transmitter interrupts are fully operation. The modem control interrupts are also operational but the modem control interrupt sources are now the lower four bits of the modem control register instead of the four modem control inputs. All interrupts are still controlled by the interrupt enable register.

Bits 5 - 7 — These bits are set to logic 0.

5.3.8. Line Status Register (LSR)[1]

The line status register provides information to the CPU concerning the status of data transfers. Table 5-2 summarizes the contents of this register and the following discusses it in detail.

Bit 0 — Bit 0 is the data ready (DR) indicator for the receiver. This bit is set to a logic 1 condition whenever a complete incoming character has been received and transferred into the receiver buffer register and is reset to logic 0 by reading the receiver buffer register.

Bit 1[2] — Bit 1 is the overrun error (OE) indicator. When this bit is set to logic 1, it indicates that before the character in the receiver buffer register was read, it was overwritten by the next character transferred into the register. The OE indicator is reset every time the CPU reads the contents of the line status register.

Bit 2[2] — Bit 2 is the parity error (PE) indicator. When this bit is set to logic 1, it indicates that the parity of the received data character does not match the parity selected in the line control register (bit 4). The PE bit is reset every time the CPU reads the contents of the line status register.

Bit 3[2] — Bit 3 is the framing error (FE) indicator. When this bit is set to logic 1, it indicates that the received character did not have a valid (logic 1).stop bit. The FE bit is reset every time the CPU reads the contents of the line status register.

Bit 4[2] — Bit 4 is the break interrupt (BI) indicator. When this bit is set to logic 1, it indicates that the received data input was held in the logic 0 state for longer than a full-word transmission time. A full-word transmission time is defined as the total time of the start, data, parity, and stop bits. The B1 bit is reset every time the CPU reads the contents of the line status register.

Bit 5 — Bit 5 is the transmitter holding register empty (THRE) indicator. This bit is set to a logic 1 condition when the transmitter holding register is empty, indicating that the ACE is ready to accept a new character. If the THRE interrupt is enabled when the THRE bit is a logic 1, then an interrupt is generated. THRE is set to a logic 1 when the contents of the transmitter holding register are transferred to the transmitted shift register. This bit is reset to logic 0 concurrent with the loading of the transmitter holding register by the CPU.

---

1. The line status register is intended for read operations only; writing to this register is not recommended outside of a factory testing environment.
2. Bits 1 through 4 are the error conditions that produce a receiver line status interrupt.

Bit 6 Bit 6 is the transmitter empty (TEMT) indicator. This bit is set to a logic 1 when the transmitter holding register and the transmitter shift register are both empty. When either the transmitter holding register or the transmitter shift register contains a data character, the TEMT bit is reset to logic 0.

Bit 7 This bit is always reset to logic 0.

5.3.9. Modem Status Register (MSR)

The modem status register is an 8-bit register that provides information about the current state of the control lines from the modem, data set, or peripheral device to the CPU. Additionally, four bits of this register provides change information; when a control input from the modem changes state the appropriate bit is set to logic 1. All four bits are reset to logic 0 when the CPU reads the modem status register. Table 5-2 discusses the contents of this register and the following discusses it in detail.

Bit 0 — Bit 0 is the delta clear to send (DCTS) indicator. This bit indicates that the $\overline{CTS}$ input has changed state since the last time it was read by the CPU. When this bit is a logic 1 and the modem status interrupt is enabled, a modem status interrupt is generated.

Bit 1 — Bit 1 is the delta data set ready (DDSR) indicates that the $\overline{DSR}$ input has changed state since the last time it was read by the CPU. When this bit is a logic 1 and the modem status interrupt is enabled, a modem status interrupt is generated.

Bit 2 — Bit 2 is the trailing edge of ring indicator (TERI) detector. This bit indicates that the $\overline{RI}$ input to the chip has changed from a low to a high state. When this bit is a logic 1 and the modem status interrupt is enabled, a modem status interrupt is generated.

Bit 3 — Bit 3 is the delta data carrier detect (DDCD) indicator. This bit indicates that the $\overline{DCD}$ input to the chip has changed state since the last time it was read by the CPU. When this bit is a logic 1 and the modem status interrupt is enabled, a modem status interrupt is generated.

Bit 4 — Bit 4 is the complement of the clear to send ($\overline{CTS}$) input. If bit 4 (loop) of the modem control register is set to a logic 1, this bit is equivalent to the modem control register bit 1 (RTS).

Bit 5 — Bit 5 is the complement of the data set ready ($\overline{DSR}$) input. If bit 4 (loop) of the modem control register is set to logic 1, this bit is equivalent to the modem control register bit 0 (DTR).

Bit 6 — Bit 6 is the complement of the ring indicator ($\overline{RI}$) input. If bit 4 (loop) of the modem control register is set to a logic 1, this bit is equivalent to the modem control registers bit 2 (OUT 1).

Bit 7 — Bit 7 is the complement of the data carrier detect ($\overline{DCD}$) input. If bit 4 (loop) of the modem control register is set to a logic 1, this bit is equivalent to the modem control register bit 3 (OUT 2).

5.3.10. Scratch Register (SCR)

The scratch register is an 8-bit register that is intended for you to use as a scratch pad. It will temporarily hold your data without affecting any other ACE operation.

The Mask/Poll register is located at I/O BASE + 7, which is the scratch register for port 1. Therefore, the scratch register for port 1 is not available.

5.3.11. Programmable Baud Generator

The ACE contains a programmable baud generator that takes a clock input in the range between DC and 9 MHz and divides it by a divisor in the range between 1 and 2^{16}-1. The output frequency of the baud generator is sixteen time (16X) the baud rate. The formula for the divisor is:

divisor # = XTAL1 frequency input /(desired baud rate x 16)

Two 8-bit registers, called divisor latches, are used to store the divisor in a 16-bit binary format. These divisor latches must be loaded during initialization of the ACE in order to ensure desired operation of the baud generator. When either of the divisor latches is loaded, a 16-bit baud counter is also loaded to prevent long counts on initial load.

5.4. Principals of Operation of the 16550 UART

This complete subsection is an excerpt of the *Data Transmission Circuits Data Book: Line Drivers, Receivers, Transceivers, UARTs* provided by Texas Instruments (TI). If you want more information about the 16550 UART, contact your nearest TI Field Sales Office, local authorized TI distributor, or by writing directly to:

Texas Instruments Incorporated
LITERATURE RESPONSE CENTER
P.O. Box 809066
Dallas, Texas 75380-9066

You can also call the TI Literature Response number at:(800) 477-8924

5.4.1. Accessible Register Overview

The following subsections discuss the following topics:

You have access to and control over any of the ACE registers through the CPU. Table 5-4 provides you with an overview of the ACE registers.

The following subsections discuss the following topics:

- Receiver buffer register (RBR)
- Transmitter holding register (THR)
- Interrupt Enable Register (IER)
- FIFO control register
- Interrupt identification register (IIR)
- Line control register (LCR)
- Modem control register (MCR)
- Line status register (LSR)
- Modem status register (MSR)
- Scratch register (SCR)
- Programmable baud generator
- FIFO interrupt-mode operation
- FIFO polled-mode operation

Table 5-4. Register Summary - 16550 Programmable UART

Bit Number	Register Addresses					
	0 DLAB=0	0 DLAB=0	1 DLAB=0	2	2	3
	Receive Buffer Register (RBR) Read-only	Transmitter Holding Register (THR) Write-only	Interrupt Enable Register (IER)	Interrupt ID Register (IIR) Read-only	FIFO Control Register (FCR) Write-only	Line Control Register (LCR)
0	data bit 0*	data bit 0	enable receive data available interrupt (ERBF)	0 if interrupt pending	FIFO enable	word length select bit 0 (WLS0)
1	data bit 1	data bit 1	enable transmitter holding register empty interrupt (ETBEI)	interrupt ID bit(0)	receiver FIFO reset	word length select bit 1 (WLS1)
2	data bit 2	data bit 2	enable receiver line status interrupt (ELSI)	interrupt ID bit(1)	transmitter FIFO reset	Number of stop bits (STB)
3	data bit 3	data bit 3	enable modem status interrupt (EDSSI)	interrupt ID bit(2)**	DMA mode select	parity enable (PEN)
4	data bit 4	data bit 4	0	0	reserved	even parity select (EPS)
5	data bit 5	data bit 5	0	0	reserved	stick parity
6	data bit 6	data bit 6	0	0	receiver trigger (LSB)	set break
7	data bit 7	data bit 7	0	0	receiver trigger (MSB)	divisor latch access bit (DLAB)

* *Bit 0 is the LSB. It is the first bit serially transmitted or received.*
** *These bits are always in the TL16C450 mode.*

Source: Texas Instruments

Table 5-4. Register Summary - 16550 Programmable UART (Continued)

Bit Number	Register Addresses					
	4	5	6	7	0 DLAB=1	1 DLAB=0
	Modem Control Register (MCR)	Line Status Register (LSR)	Modem Status Register (MSR)	Scratch Register (SCR)	Divisor Latch (DLL) LSB	Latch (DLM) MSB
0	data terminal ready (DTR)	data ready (DR)	delta clear to send (DCTS)	bit 0	bit 0	bit8
1	request to send (RTS)	overrun error (OE)	delta data set ready (DDSR)	bit 1	bit 1	bit 9
2	out 1	parity error (PE)	trailing edge ring indicator (TER)	bit 2	bit 2	bit 10
3	out 2	framing error (FE)	delta data carrier detect (DDCD)	bit 3	bit 3	bit 11
4	loop	break interrupt (BI)	clear to send (CTS)	bit 4	bit 4	bit 12
5	0	transmitter holding register THRE	data set ready (DSR)	bit 5	bit 5	bit 13
6	0	transmitter empty TEMT	ring indicator (RI)	bit 6	bit 6	bit 14
7	0	Error in RCVR FIFO **	data carrier detect (DCD)	bit 7	bit 7	bit 15

* *Bit 0 is the LSB. It is the first bit serially transmitted or received.*
** *These bits are always in the TL16C450 mode.*

Source: Texas Instruments

5.4.2. Receiver Buffer Register (RBR)

The ACE's receiver section consists of a receiver shift register (RSR) and a receiver buffer register (RBR). The RBR is actually a 16-byte FIFO. Timing is supplied by the 16 X Receiver Clock (RCLK). Receiver section control is a function of the ACEs line control register.

The ACE's RSR receives serial data from the serial input (SIN) pin. The RSR then deserializes the data and moves it into the RBR FIFO. In TL16C450 mode, when a character is placed in the receiver buffer register and the received data available interrupt is enabled, an interrupt is generated. This interrupt is cleared when the data is read out of the receiver buffer register. In FIFO mode, the interrupts are generated based on the control setup in the FIFO control register.

5.4.3. Transmitter Holding Register (THR)

The ACE's transmitter section consists of a transmitter holding register (THR) and a transmitter shift register (TSR).The THR is actually a 16-byte FIFO. Timing is supplied by the baud out ($\overline{\text{BAUDOUT}}$) clock signal. Transmitter section control is a function of the ACE's line control register.

The ACE's THR receives data off the internal data bus and, when the shift register is idle, moves it into the TSR. The TSR serializes the data and outputs it at the serial output (SOUT). In TL16C450 mode, if the THR is empty and the transmitter holding register empty (THRE) interrupt is enabled, and interrupt is generated. This interrupt is cleared when a character is loaded into the register. In FIFO mode, the interrupts are generated based on the control setup in the FIFO control register.

5.4.4. Interrupt Enable Register (IER)

The interrupt enable register enables each of the five types of interrupts (Table 5-5) and the INTRPT output signal in response to an interrupt generation. The interrupt enable register can also be used to disable the interrupt system by setting bits 0 through 3 to logic 0. Table 5-4 summarizes the contents of this register and the following discusses it in detail.

Bit 0 — This bit, when set to logic 1, enables the received data available interrupt.

Bit 1 — This bit, when set to logic 1, enables the transmitter holding register empty interrupt.

Bit 2 — This bit, when set to logic 1, enables the receiver line status interrupt.

Bit 3 — This bit, when set to logic 1, enables the modem status interrupt.

Bits 4 - 7 Bits 4 through 7 in the interrupt enable register are not used and are always set to logic 0.

5.4.5. FIFO Control Register (FCR)

The FIFO control register (FCR) is a write-only register at the same location as the IIR, which is a read-only register. The FCR is used to enable the FIFOs, clear the FIFOs, set the receiver FIFO trigger level, and select the type of DMA signaling.

Bit 0 FCR0, when set to logic 1, enables the transmit and receive FIFOs. This bit must be a 1 when other FCR bits are written to or they will not be programmed. Changing this bit clears the FIFOs.

Bit 1 FCR1, when set to logic 1, clears all bytes in the receiver FIFO and resets its counter logic to 0. The shift register is not cleared. The 1 that is written to this bit position is self-clearing.

Bit 2 FCR2, when set to logic 1, clears all bytes in the transmit FIFO and resets its counter to 0. The shift register is not cleared. The 1 that is written to this bit position is self-clearing.

Bit 3 If FCR0 is a 1, setting FCR3 to a 1 causes the $\overline{\text{RXRDY}}$ and $\overline{\text{TXRDY}}$ to change from mode 0 to mode 1.

Bits 4 & 5 FCR4 and FCR5 are reserved for future use.

Bits 6 & 7 FCR6 and FCR7 are used to set the trigger level for the receiver FIFO interrupt.

Bit 7	Bit 6	Receiver FIFO Trigger Level (Bytes)
0	0	01
0	1	04
1	0	08
1	1	14

5.4.6. Interrupt Identification Register (IIR)

The ACE has an on-chip interrupt generation and prioritization capability that permits a flexible interface with most popular microprocessors.

The ACE provides four prioritized levels of interrupts:

Priority 1 Receiver line status (highest priority)

Priority 2 Receiver data ready or receiver character timeout

Priority 3 Transmitter holding register empty

Priority 4 Modem status (lowest priority)

When an interrupt is generated, the interrupt identification register indicates that an interrupt is pending and the type of that interrupt in its three least significant bits (bits 0,1, and 2). Table 5-4 summarizes the contents of this register and Table 5-5 discusses it in detail. Each bit is as follows:

Bit 0 This bit can be used either in a hardwire-prioritized, or polled interrupt system. If this bit is a logic 0, an interrupt is pending. When bit 0 is a logic 1, no interrupt is pending.

Bits 1 & 2 These two bits are used to identify the highest priority interrupt pending, as indicated in Table 5-5.

Bits 4 – 5 These two bits are not used and are always set at logic 0.

Bits 6 & 7 These two bits are always 0 in the TL16C450 mode. They are set when bit 0 of the FIFO control register is equal to 1.

Table 5-5. Interrupt Control Functions

Interrupt Identification Register				Priority Level	Interrupt Type	Interrupt Source	Interrupt Reset Method
Bit 3	Bit 2	Bit 1	Bit 0				
0	0	0	1	None	None	None	–
0	1	1	0	1	Receiver line status	Overrun error, parity error, framing error or break interrupt	Reading the line status register
0	1	0	0	2	Receiver data available	Receiver data available in TL16C450 mode or trigger level reached in the FIFO mode.	Reading the receiver buffer register

Table 5-5. Interrupt Control Functions (Continued)

Interrupt Identification Register				Priority Level	Interrupt Type	Interrupt Source	Interrupt Reset Method
Bit 3	Bit 2	Bit 1	Bit 0				
1	1	0	0	2	Character timeout indication	No characters have been removed from or input to the receiver FIFO during the last four character times and there is a least one character in it during this time.	Reading the receiver buffer register
0	0	1	0	3	Transmitter holding register empty	Transmitter holding register empty	Reading the IIR (if source of interrupt) or writing into the transmitter holding register
0	0	0	0	4	Modem status	Clear to send, data set ready, ring indicator, or data carrier detect	Reading the modem status register

5.4.7. Line Control Register (LCR)

You can control the format of the asynchronous data communications exchange through the line control register. In addition, you can retrieve, inspect, and modify the contents of the line control register. This eliminates the need for separate storage of the line characteristics in system memory. Table 5-4 summarizes the contents of this register and the following discusses it in detail.

Bits 0 & 1 These two bits specify the number of bits in each transmitted or received serial character. These bits are encoded as follows:

Bit 1	Bit 0	Word Length
0	0	5 bits
0	1	6 bits
1	0	7 bits
1	1	8 bits

Bit 2 This bit specifies either one, one and one-half, or two-stop bits in each transmitted character. If bit 2 is a logic 0, one stop bit is generated in the data. If bit 2 is logic 1, the number of stop bits generated is dependent on the word length selected with bits 0 and 1. The receive clocks the first stop bit only, regardless of the number of stop bits selected. The number of stop bits generated, in relation to word length and bit 2, is shown in the following table:

Bit 2	Word Length Selected by Bits 1 and 2	Number of Stop Bits Generated
0	Any word length	1
1	5 bits	1 1/2
1	6 bits	2
1	7 bits	2
1	8 bits	2

Bit 3 This bit is the parity enable bit. When bit 3 is a logic 1, a parity bit is generated in transmitted data between the last data word bit and the first stop bit. In received data, if bit 3 is a logic 1, parity is checked. When bit 3 is a logic 0, no parity is generated or checked.

5.4.7. Line Control Register (LCR)

You can control the format of the asynchronous data communications exchange through the line control register. In addition, you can retrieve, inspect, and modify the contents of the line control register. This eliminates the need for separate storage of the line characteristics in system memory. Table 5-4 summarizes the contents of this register and the following discusses it in detail.

Bits 0 & 1 These two bits specify the number of bits in each transmitted or received serial character. These bits are encoded as follows:

Bit 1	Bit 0	Word Length
0	0	5 bits
0	1	6 bits
1	0	7 bits
1	1	8 bits

Bit 2 This bit specifies either one, one and one-half, or two-stop bits in each transmitted character. If bit 2 is a logic 0, one stop bit is generated in the data. If bit 2 is logic 1, the number of stop bits generated is dependent on the word length selected with bits 0 and 1. The receive clocks the first stop bit only, regardless of the number of stop bits selected. The number of stop bits generated, in relation to word length and bit 2, is shown in the following table:

Bit 2	Word Length Selected by Bits 1 and 2	Number of Stop Bits Generated
0	Any word length	1
1	5 bits	1 1/2
1	6 bits	2
1	7 bits	2
1	8 bits	2

Bit 3 This bit is the parity enable bit. When bit 3 is a logic 1, a parity bit is generated in transmitted data between the last data word bit and the first stop bit. In received data, if bit 3 is a logic 1, parity is checked. When bit 3 is a logic 0, no parity is generated or checked.

Bit 4 Bit 4 is the even parity select bit. When parity is enabled by bit 3; a logic 1 in bit 4 produces even parity (an even number of logic 1's in the data and parity bits) and a logic 0 in bit 4 produces odd parity (an odd number of logic 1's).

Bit 5 This is the stick parity bit. When bits 3, 4, and 5 are logic 1's, the parity bit is transmitted and checked as a logic 0. When bits 3 and 5 are logic 1's and bit 4 is a logic 0, the parity bit is transmitted and checked as a logic 1. If bit 5 is a logic 0, stick parity is disabled.

Bit 6 This bit is the break control bit. Bit 6 is set to a logic 1 to force a break condition, that is, a condition where the serial output (SOUT) pin is forced to the spacing (logic 0) state. When bit 6 is set to a logic 0, the break condition is disabled. The break condition has no effect on the transmitter logic; it only effects the serial output.

Bit 7 This bit is the divisor latch access bit (DLAB). Bit 7 must be set to a logic 1 to access the divisor latches of the baud generator during a read or write. Bit 7 must be set to a logic 0 during a read or write to access the receiver buffer, the transmitter holding register, or the interrupt enable register.

5.4.8. Modem Control Register (MCR)

The modem control register is an 8-bit register that controls an interface with a modem, data set, or peripheral device that is emulating a modem. Table 5-4 summarizes the contents of this register and the following discusses it in detail.

Bit 0 Bit 0 (DTR) controls the data terminal ready ($\overline{DTR}$) output. Setting this bit to a logic 1 forces the $\overline{DTR}$ output to its low state. When bit 0 is set to a logic 0, $\overline{DTR}$ goes high.

Bit 1 Bit 1 (RTS) controls the request to sent ($\overline{RTS}$) output in a manner identical to bit 0's control over the $\overline{DTR}$ output.

Bit 2 Bit 2 (OUT 1) controls the output 1 ($\overline{OUT\ 1}$) signal, a user-designated output signal, in a manner identical to bit 0's control over the $\overline{DTR}$ output.

Bit 3 Bit 3 (OUT 2) controls the output 2 ($\overline{OUT\ 2}$) signal, a user-designated output signal, in a manner identical to bit 0's control over the $\overline{DTR}$ output.

Bit 4 Bit 4 provides a local loopback feature for diagnostic testing of the ACE. When this bit is set to a logic high, the following occurs:

1. The transmitter serial output (SOUT) is set high.
2. The receiver serial input (SIN) is disconnected.
3. The output of the transmitter shift register is looped back into the receiver shift register input.

4. The four modem control inputs ($\overline{CTS}$, $\overline{DSR}$, $\overline{DCD}$, and $\overline{RI}$) are disconnected.
5. The four modem control outputs ($\overline{DTR}$, $\overline{RTS}$, $\overline{OUT\ 1}$, and $\overline{OUT\ 2}$) are internally connected to the our modem control inputs.
6. The four modem control output pins are forced to their inactive states (high).

In diagnostic mode, data that is transmitted is immediately received. This allows the processor to verify the transmit- and receive-data paths to the ACE. The receiver and transmitter interrupts are fully operational. The modem control interrupts are also operational but the modem control interrupt's sources are now the lower four bits of the modem control register instead of the four modem control inputs. All interrupts are still controlled by the interrupt enable register.

Bits 5 – 7 These bits are permanently set to logic 0.

5.4.9. Line Status Register (LSR)[3]

The line status register provides information to the CPU concerning the status of data transfers. Table 5-4 summarizes the contents of this register and the following discusses it in detail.

Bit 0 Bit 0 is the data ready (DR) indicator for the receiver. This bit is set to a logic 1 condition whenever a complete incoming character has been received and transferred into the receiver buffer register or the FIFO and is reset to logic 0 by reading all of the data in the receiver buffer register or the FIFO.

Bit 1[4] Bit 1 is the overrun error (OE) indicator. When this bit is set to logic 1, it indicates that before the character in the receiver buffer register was read, it was overwritten by the next character transferred into the register. The OE indicator is reset every time the CPU reads the contents of the line status register. If FIFO mode data continues to fill the FIFO beyond the trigger level, an overrun error will occur only after the FIFO is full and the next character has been completely received in the shift register. OE is indicated to the CPU as soon as it happens. The character in the shift register is overwritten but is not transferred to the FIFO.

3. The line status register is intended for read operations only; writing to this register in not recommended outside of a factory environment.
4. Bits 1 through 4 are the error conditions that produce a receiver line status interrupt.

Bit 2[5] Bit 2 is the parity error (PE) indicator. When this bit is set to logic 1, it indicates that the parity of the received data character does not match the parity selected in the line control register (bit 4). The PE bit is reset every time the CPU reads the contents of the line status register. In FIFO mode, this error is associated with the particular character in the FIFO to which it applies. This error is revealed to the CPU when its associated character is at the top of the FIFO.

Bit 3[5] Bit 3 is the framing error (FE) indicator. When this bit is set to logic 1, it indicates that the received character did not have a valid (logic 1) stop bit. The FE bit is reset every time the CPU reads the contents of the line status register. In FIFO mode, this error is associated with the particular character in the FIFO to which it applies. This error is revealed to the CPU when its associated character is at the top of the FIFO. The ACE will try to resynchronize after a framing error. To accomplish this, it is assumed that the framing error is due to the next start bit. The ACE then samples this start bit twice and then accepts the input data.

Bit 4[5] Bit 4 is the break interrupt indicator. When this bit is set to logic 1, it indicates that the received data input was held in the logic 0 state for longer than a full-word transmission time. A full-word transmission time is defined as the total time of the start, data, parity, and stop bits. The B1 bit is reset every time the CPU reads the contents of the line status register. In FIFO mode, this error is associated with the particular character in the FIFO to which it applies. This error is revealed to the CPU when its associated character is at the top of the FIFO. When break occurs, only one 0 character is loaded into the FIFO. The next character transfer is enabled after SIN goes to the marking state and receives the next valid start bit.

Bit 5 Bit 5 is the transmitter holding register empty (THRE) indicator. This bit is set to logic 1 when the transmitter holding register is empty, indicating that the ACE is ready to accept a new character. If the THRE interrupt is enabled when the THRE bit is a logic 1, then an interrupt is generated. THRE is set to a logic 1 when the contents of the transmitter holding register are transferred to the transmitter shift register. This bit is reset to logic 0 concurrent with the loading of the transmitter holding register by the CPU. In FIFO mode, this bit is set when the transmit FIFO is empty; it is cleared when at least 1 byte is written to the transmit FIFO.

---

5. Bits 1 through 4 are the error conditions that produce a receiver line status interrupt.

Bit 6 Bit 6 is the transmitter empty (TEMT) indicator. This bit is set to a logic 1 when the transmitter holding register and the transmitter shift register are both empty. When either the transmitter holding register or the transmitter shift register contains a data character, the TEMT bit is reset to logic 0. In FIFO mode, this bit is set to a 1 when the transmitter FIFO and shift register are both empty.

Bit 7 In the TL16C550A, this bit is always reset to logic 0. In TL16C450 mode, this bit is always a 0. In FIFO mode, LSR7 is set when there is at least one parity, framing, or break error in the FIFO. It is cleared when the microprocessor reads the LSR and there are no subsequent errors in the FIFO.

5.4.10. Modem Status Register (MSR)

The modem status register is an 8-bit register that provides information about the current state of the control lines from the modem, data set, or peripheral device to the CPU. Additionally, four bits of this register provides change information; when a control input from the modem changes state, the appropriate bit is set to logic 1. All four bits are reset to logic 0 when the CPU reads the modem status register. Table 5-4 summarizes the contents of this register and the following discusses it in detail.

Bit 0 Bit 0 is the change in clear to send (DCTS) indicator. This bit indicates that the $\overline{CTS}$ input has changed state since the last time it was read by the CPU. When this bit is a logic 1 and the modem status interrupt is enabled, a modem status interrupt is generated.

Bit 1 Bit 1 is the change in data set ready (DDSR) indicator. This bit indicates the $\overline{DSR}$ input has changed state since the last time it was read by the CPU. When this bit is a logic 1 and the modem status interrupt is enabled, a modem status interrupt is generated.

Bit 2 Bit 2 is the trailing edge of ring indicator (TERI) detector. This bit indicates that the $\overline{RI}$ input to the chip has changed from a low to high state. When this bit is a logic 1 and the modem status interrupt is enabled, a modem status interrupt is generated.

Bit 3 Bit 3 is the change in data carrier detect (CDDC) indicator. This bit indicates that the $\overline{DCD}$ input to the chip has changed state since the last time it was read by the CPU. When this bit is a logic 1 and the modem status interrupt is enabled, a modem status interrupt is generated.

Bit 4 Bit 4 is the compliment of the clear to send ($\overline{CTS}$) input. If bit 4 (loop) of the modem control register is set to a logic 1, this bit is equivalent to the modem control register bit 1 (RTS).

Bit 5 Bit 5 is the compliment of the data set ready ($\overline{DSR}$) input. If bit 4 (loop) of the modem control register is set to a logic 1, this bit is equivalent to the modem control register bit 1 (DTR).

Bit 6 Bit 6 is the compliment of the ring indicator ($\overline{RI}$) input., If bit 4 (loop) of the modem control register is set to a logic 1, this bit is equivalent to the modem control registers bit 2 (OUT 1).

Bit 7 Bit 7 is the compliment of the data carrier detect ($\overline{DCD}$) input. If bit 4 (loop) of the modem control register is set to a logic 1, this bit is equivalent to the modem control registers bit 3 (OUT 2).

5.4.11. Scratch Register (SCR)

The scratch register is an 8-bit register that is intended for you to use a scratch pad. It temporarily holds the your data without affecting any other ACE operation.

The Mask/Poll register is located at I/O BASE + 7, which is the scratch register for port 1. Therefore, the scratch register for port 1 is not available.

5.4.12. Programmable Baud Generator

The ACE contains a programmable baud generator that takes a clock input in the range between DC and 8 MHz and divides it by a divisor in the range between 1 and 2^{16}-1. The output frequency of the baud generator is sixteen times (16X) the baud rate. The formula for the divisor is:

divisor # = XIN frequency input /(desired baud rate x 16)

Two 8-bit registers, called divisor latches are used to store the divisor in a 16-bit binary format. These divisor latches must be loaded during initialization of the ACE in order to ensure desired operation of the baud generator. When either of the divisor latches is loaded, a 16-bit baud counter is also loaded to prevent long counts on initial load.

For baud rates of 38.4 kilobits per second and below, the error obtained is very small. The accuracy of the selected baud rate is dependent on the selected crystal frequency.

5.4.13. FIFO Interrupt-Mode Operation

When the receiver FIFO and receiver interrupts are enabled (FCR0=1,IERO=1) receiver interrupts will occur as follows:

1. The receive data available interrupt will be issued to the microprocessor when the FIFO has reached its programmed trigger level. It will be cleared as soon as the FIFO drops below its programmed trigger level.
2. The IIR receive data available indication also occurs when the FIFO trigger level is reached, and, like the interrupt, it is cleared when the FIFO drops below the trigger level.
3. The receiver line status interrupt (IIR=06), as before, has higher priority than the received data available (IIR=04) interrupt.
4. The data ready bit (LSR0) is set as soon as a character is transferred from the shift register to the receiver FIFO. It is reset when the FIFO is empty.

When the receiver FIFO and receiver interrupts are enabled, receiver FIFO timeout interrupts will occur as follows:

1. FIFO timeout interrupt will occur if the following conditions exist:
   a. At least one character is in the FIFO.
   b. The most recent serial character received was longer than four continuous character times ago (if 2 stop bits are programmed, the second one is included in this time delay).
   c. The most recent microprocessor read of the FIFO was longer than four continuous character times ago. This causes a maximum character received to interrupt issued delayed of 160 ms at 300 baud with 12-bit character.
2. Character times are calculated by using the RCLK input for a clock signal (this makes the delay proportional to the baud rate).
3. When a timeout interrupt has occurred, it is cleared and the timer resets when the microprocessor reads one character from the receiver FIFO.
4. When a timeout interrupt has not occurred, the timeout timer is reset after a new character is received or after the microprocessor reads the receiver FIFO.

When the transmit FIFO and transmitter interrupts are enabled (FCR0=1, IER1=1), transmit interrupts will occur as follows:

1. The transmitter holding register interrupt (02) occurs when the transmit FIFO is empty. It is cleared as soon as the transmitter holding register is written to (1 to 16 characters may be written to the transmit FIFO while servicing this interrupt) or the IIR is read.

2. The transmit FIFO empty indications will be delayed one character time minus the last stop bit time when the following occurs:
   - THRE=1 and there have not been at least two bytes at the same time in the transmit FIFO since the last THRE=1.
   - The first transmitter interrupt after changing FCR0 will be immediate, if it is enabled.

Character timeout and receiver FIFO trigger level interrupts have the same priority as the current received data available interrupt; transmit FIFO empty has the same priority as the current transmitter holding register empty interrupt.

5.4.14. FIFO Polled-Mode Operation

With FCR0=1, resetting IER0, IER1, IER2, IER3, or all four put the ACE in the FIFO polled mode of operation. Since the receiver and transmitter are controlled separately, either one or both can be in the polled mode of operation.

In this mode, the user program will check receiver and transmitter status through the LSR:

1. LSR0 will be set as long as there is one byte in the receiver FIFO.
2. LSR1 through LSR4 will specify which error or errors have occurred. Character error status is handled the same way as when in the interrupt mode, the IIR is not affected since IER2=0.
3. LSR5 will indicate when the transmit FIFO is empty.
4. LSR6 will indicate that both the transmit FIFO and shift registers are empty.
5. LSR7 will indicate whether there are any errors in the receiver FIFO.

There is no trigger level reached or timeout conditions indicated in the FIFO polled mode. However, the receiver and transmit FIFOs are still fully capable of holding characters.

This page was intentionally left blank.

5.5. Principles of Operation of the 16552 DUART

This complete subsection is an excerpt of the *Data Transmission Circuits Data Book: Line Drivers, Receivers, Transceivers, UARTs* provided by Texas Instruments (TI). If you want more information about the 16552 UART, contact your nearest TI Field Sales Office, local authorized TI distributor, or by writing directly to:

Texas Instruments Incorporated
LITERATURE RESPONSE CENTER
P.O. Box 809066
Dallas, Texas 75380-9066

You can also call the TI Literature Response number at:(800) 477-8924

The following subsections contain information about these topics:

- Detailed description of internal registers
- Line control register (LCR)
- Line status register (LSR)
- FIFO control register (FCR)
- Modem control register (MCR)
- Modem status register (MSR)
- Divisor latches
- Scratchpad register (SCR)
- Interrupt identification register (IIR)
- Interrupt enable register (IER)
- Receiver
- Master reset
- Programming
- FIFO-interrupt-mode operation

5.5.1. Detailed Description of Internal Registers

Three types of information are stored in the internal registers used in the ACE:

- Control
- Status
- Data

The following table shows the mnemonic abbreviations for the registers.

Control	Mnemonic
Line control register	LCR
FIFO control register	FCR
Modem control register	MCR
Divisor latch LSB	DLL
Divisor latch MSB	DLM
Interrupt enable register	IER
Status	**Mnemonic**
Line status register	LSR
Modem status register	MSR
Data	**Mnemonic**
Receiver buffer register	RBR
Transmitter holding register	THR

The address, read, and write inputs are used with the divisor latch access bit (DLAB) in the line control register (bit 7) to select the register to be written or read (see Table 5-6).

Table 5-6. Serial Channel Internal Registers

DLAB	A2	A1	A0	Mnemonic	Register
L	L	L	L	RBR	Receiver buffer register (read only)
L	L	L	L	THR	Transmitter holding register (write only)
L	L	L	H	IER	Interrupt enable register
X	L	H	L	IIR	Interrupt identification register (read only)
X	L	H	L	FCR	FIFO control register (write only)
X	L	H	H	LCR	Line control register

Table 5-6. Serial Channel Internal Registers

DLAB	A2	A1	A0	Mnemonic	Register
X	H	L	L	MCR	Modem control register
X	H	L	H	LSR	Line status register
X	H	H	L	MSR	Modem status register
X	H	H	H	SCR	Scratch register
H	L	L	L	DLL	Divisor latch (LSB)
H	L	L	H	DLM	Divisor latch (MSB)

X = Irrelevant, L = Low level, H = High level

Note: *The serial channel is accessed when either* $\overline{CS0}$ *or* $\overline{CS1}$ *is low.*

Individual bits within the registers are referred to by the register mnemonic and the bit number in parenthesis. As an example, LCR (7) refers to the line control register bit 7.

The transmitter buffer register and receiver buffer register are data registers that hold from five to eight bits of data. If less than eight data bits are transmitted, data is right justified to the LSB. Bit 0 of a data word is always the first serial data bit received and transmitted. The ACE data registers are double buffered so that read and write operations may be performed when the ACE is performing the parallel-to-serial or serial-to-parallel conversion.

Table 5-7. Register Summary for the 16552 Programmable DUART

Address	Register Mnemonic	Registers Bit Number			
		Bit 7	Bit 6	Bit 5	Bit 4
0	RBR read-only	data bit 7 (MSB)	data bit 6	data bit 5	data bit 4
0	THR write-only	data bit 7	data bit 6	data bit 5	data bit 4
0*	DDL	bit 7	bit 6	bit 5	bit 4
1°	DLM	bit 15	bit 14	bit 13	bit 12
1	IER	0	0	0	0
2	FCR write-only	RCVR trigger level (MSB)	RCVR trigger level (LSB)	Reserved	Reserved
2	IIR read-only	FIFOs enabled**	FIFOs enabled**	0	0
3	LCR	divisor latch access bit (DLAB)	set break	stick parity	even parity select (EPS)
4	MCR	0	0	0	loop
5	LSR	error in RCVR FIFO **	transmitter empty (TEMT)	transmitter holding register empty (THRE)	break interrupt (BI)
6	MSR	data carrier detect (DCD)	ring indicator (RI)	data set ready (DSR)	clear to send (CTS)
7	SCR	Bit 7	Bit 6	Bit 5	Bit 4

* *DLAB=1*

** *These bits are always 0 when FIFOs are disabled.*

Source: Texas Instruments

Table 5-7. Register Summary for the 16552 Programmable DUART (Continued)

Address	Register Mnemonic	Register Bit Number			
		Bit 3	Bit 2	Bit 1	Bit 0
0	RBR read-only	data bit 3	data bit 2	data bit 1	data bit 0 (LSB)
0	THR write-only	data bit 3	data bit 2	data bit 1	data bit 0
0*	DLL	bit 3	bit 2	bit 1	bit 0
1°	DLM	bit 11	bit 10	bit 9	bit 8
1	IER	enable modem status interrupt (EDSSI)	enable receiver line status interrupt (ERRLSI)	enable transmitter holding register empty interrupt	enable received data available interrupt (ERBFI)
2	FCR write-only	DMA mode select	XMIT FIFO reset	RCVR FIFO reset	FIFO enable
2	IIR read-only	interrupt ID bit(2)**	interrupt ID bit(1)	interrupt ID bit 0	0 if interrupt pending
3	LCR	parity enable (PEN)	number of stop bits (STB)	word length select bit (WLSB)	word length select bit 0 (WLSB0)
4	MCR	enable external interrupt (INT0 or INT1)	OUT1 (an unused internal signal	request to send (RTS)	data terminal ready (DTR)
5	LSR	framing error (FE)	parity error (PE)	overrun error (OE)	data ready (DR)
6	MSR	delta data carrier detect (DDCD)	trailing edge ring indicator (TERI)	delta data set ready (DDSR)	delta data clear to send (DCTS)
7	SCR	Bit 3	Bit 2	Bit 1	Bit 0

* *DLAB=1*

** *These bits are always 0 when FIFOs are disabled.*

Source: Texas Instruments

5.5.2. Line Control Register (LCR)

The format of the data character is controlled by the line control register, the LCR may be read. Its contents are described below and shown in Figure 5-1.

LCR Bit	Description
LCR(0) and LCR(1) word length select bit 1	The number of bits in each serial character is programmed as shown in Figure 5-1.
LCR(2) stop bit select bit 2	LCR(2) specifies the number of stop bits in each transmitted character as shown in Figure 5-1. The receiver always checks for one stop bit.
LCR(3) parity enable bit 3	When LCR(3) is high, a parity bit between the last data word bit and stop bit is generated and checked.
LCR(4) even parity select bit 4	When enabled, a logic one selects even parity.
LCR(5) stick parity bit 5	When parity is enabled (LCR(3)=1), LCR(5)=1 causes the transmission and reception of a parity bit to be in the opposite state from the value of LCR(4). This forces parity to a known state and allows the receiver to check the parity bit in a known state.
LCR(6) break control bit 6	When LCR(6) is set to logic 1, the serial output (SOUT1/SOUT0) is forced to the spacing state (low). The break control bit acts only on the serial output and does not affect the transmitter logic. If the following sequence in used, no invalid characters will be transmitted because of the break: 1. Load a zero byte in response to the transmitter holding register empty (THRE) status indication. 2. Set the break in response to the next THRE status information. 3. Wait for the transmitter to be idle when transmitter empty status signal is set high (TEMT=1). Then clear the break when the normal transmission has to be restored.
LCR(7) divisor latch access bit (DLAB) bit 7	Bit 7 must be set high (logic 1) to access the divisor latches DLL and DLM of the baud rate generator during a read or write operation. LCR(7) must be input low (logic 0) to access the receiver buffer register, the transmitter holding register or the interrupt enable register.

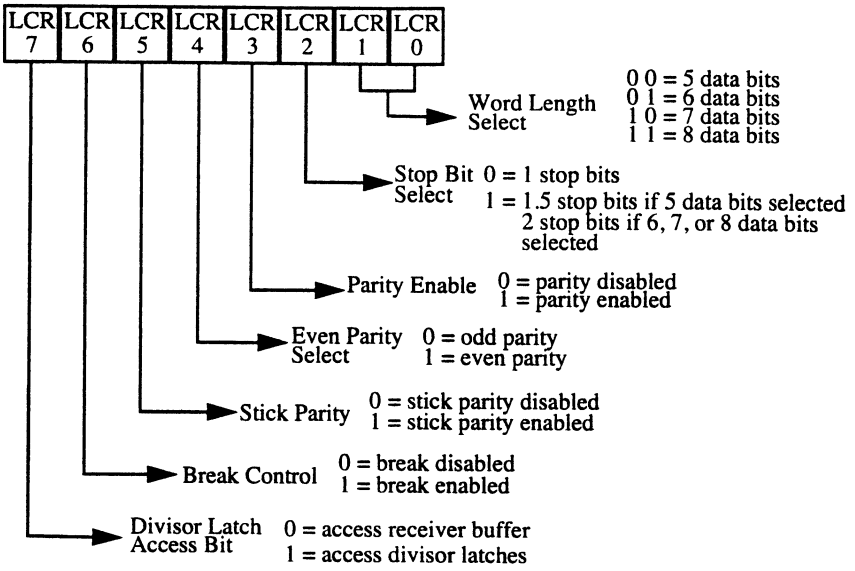


Figure 5-1. Line Control Register

5.5.3. Line Status Register (LSR)

The line status register (LSR) is a single register that provides status indications. Table 5-7 summarizes the contents of this register and the following discusses it in detail.

Table 5-8. Line Status Register Descriptions

LSR Bit	Description
LSR(0) data ready (DR) bit 0	Data ready is set high when an incoming character has been received and transferred into the receiver buffer register or the FIFO. LSR(0) is reset low by a CPU read of the data in the receiver buffer register or the FIFO.
LSR(1) overrun error (OE) bit 1	Overrun error indicates that data in the receiver buffer register was not read by the CPU before the next character was transferred into the receiver buffer register overwriting the previous character. The OE indicator is reset whenever the CPU reads the contents of the line status register. An overrun error will occur in FIFO mode after the FIFO is full and the next character is completely received. The overrun error is detected by the CPU on the first LSR read after it happens. The character in the shift register is not transferred to the FIFO but it is overwritten.

Table 5-8. Line Status Register Descriptions (Continued)

LSR Bit	Description
LSR(2) parity error (PE) bit 2	Parity error indicates that the received data character does not have the correct parity as selected by LCR(3) and LCR(4). The PE bit is set high upon detection of a parity error and is reset low when the CPU reads the contents of the LSR. In FIFO mode, the parity error is associated with a particular character in the FIFO. LSR(2) reflects the error when the character is at the top of the FIFO.
LSR(3) framing error (FE) bit 3	Framing error indicates that the received character did not have a valid stop bit. LSR(3) is set high when the stop bit following the last data bit or parity bit is detected as a zero bit (spacing level). The FE indicator is reset low when the CPU reads the contents of the LSR. In FIFO mode, the framing error is associated with a particular character in the FIFO. LSR(3) reflects the error when the character is at the top of the FIFO.
LSR(4) break interrupt (BI) bit 4	Break interrupt is set high when the received data input is held in the spacing (logic 0) state for longer than a full word transmission time (start bit + data bits + parity + stop bits). The BI indicator is reset when the CPU reads the contents of the line status register. In FIFO mode, this is associated with a particular character in the FIFO. LSR(2) reflects the BI when the break character is at the top of the FIFO. The error is detected by the CPU when its associated character is at the top of the FIFO during the first LSR read. Only one zero character is loaded into the FIFO when BI occurs. LSR(1) through LSR(4) are the error conditions that produce a receiver line status interrupt (priority 1 interrupt in the interrupt identification register (IIR)) when any of the conditions are detected. This interrupt is enabled by setting IER(2)=1 in the interrupt enable register.
LSR5(5) transmitter holder register empty (THRE) bit 5	THRE indicates that the ACE is ready to accept a new character for transmission. The THRE bit is set high when a character is transferred from the transmitter holding register into the transmitter shift register. LSR(5) is reset low by the loading of the transmitter holding register by the CPU. LSR(5) is not reset by a CPU read of the LSR. In FIFO mode, when the XMIT FIFO is empty, this bit is set. It is cleared when one byte is written to the XMIT FIFO. When the THRE interrupt is enabled by IER(1), THRE causes a priority 3 interrupt in the IIR. If THRE is the interrupt source indicated in IIR, INTRPT is cleared by a read of the IIR.

Table 5-8. Line Status Register Descriptions (Continued)

LSR Bit	Description
LSR(6) transmitter empty (TEMT) bit 6	TEMT is set high when the transmitter holding register (THR) and the transmitter shift register (TSR) are both empty. LSR(6) is reset low when a character is loaded into the THR and remains low until the character is transferred out of SOUT. TEMT is not reset low by a CPU read of the LSR. In the FIFO mode, when both the transmitter FIFO and shift register are empty, this bit is set to one.
LSR(7) RCVR FIFO error bit 7	The LSR(7) bit is always 0 in the TL16C450 mode. In FIFO mode, it is set when at least one of the following data errors is in the FIFO: parity error, framing error, or break interrupt indication. It is cleared when the CPU reads the LSR if there are no subsequent errors in the FIFO.

Note: *The line status register may be written. However, this function is intended only for factory test. It should be considered as read only by applications software.*

Table 5-9. Line Status Register Bits

LSR Bits	1	0
LSR0 data ready (DR)	Ready	Not ready
LSR1 overrun error (OE)	Error	No error
LSR2 parity error (PE)	Error	No error
LSR3 framing error (FE)	Error	No error
LSR4 break interrupt (BI)	Break	No break
LSR5 transmitter holding register (THRE)	Empty	Not empty
LSR6 transmitter empty (TEMT)	Empty	Not empty
LSR7 RCVR FIFO error	Error in FIFO	No error in FIFO

5.5.4. FIFO Control Register (FCR)

This write-only register is at the same location as the IIR. It is used to enable and clear the FIFOs, set the trigger level of the RCVR FIFO, and select the type of DMA signaling.

FCR Bit	Description
FCR(0)	Enables both the XMIT and RCVR FIFOs. All bytes in both FIFOs can be cleared by resetting FCR(0). Data is cleared automatically from the FIFOs when changing from FIFO mode to TL16C450 mode and vice versa. Programming of other FCR bits is enabled by setting FCR(0)=1.
FCR(1)=1	Clears all bytes in the RCVR FIFO and resets the counter logic to 0. This does not clear the shift register.
FCR(2)=1	Clears all bytes in the XMIT FIFO and resets the counter logic 0. This does not clear the shift register.
FCR(3)=1	Changes the $\overline{\text{RXRDY}}$ and $\overline{\text{TXRDY}}$ pins from mode 0 to mode 1 if FCR(0)=1.
FCR(4) - FCR(5)	These two bits are reserved for future use.
FCR(5) - FCR(7)	These two bits are used for setting the trigger level for the RCVR FIFO interrupt as follows: **RCVR FIFO** **Bit 7 / Bit 6 / Trigger Level (Bytes)** 0 0 01 0 1 04 1 0 08 1 1 14

5.5.5. Modem Control Register (MCR)

The modem control register (MCR) controls the interface with the modem or data set as described in Figure 5-2. MCR can be written and read. The $\overline{\text{RTS}}$ and $\overline{\text{DTR}}$ outputs are directly controlled by their control bits in this register. A high input asserts a low (true) at the output pins 0, 1, 2, 3, and 4 are shown as follows.

MCR Bit	Description
MCR(0)	When MCR(0) is set high, the $\overline{\text{DTR}}$ output is forced low. When MCR(0) is reset low, the $\overline{\text{DTR}}$ output is forced high. The $\overline{\text{DTR}}$ output of the serial channel may be input into an inverting line driver in order to obtain the proper polarity input at the modem or data set.
MCR(1)	When MCR(1) is set high, the $\overline{\text{RTS}}$ output is forced low. When MCR(1) is reset low, the $\overline{\text{RTS}}$ output is forced high. The $\overline{\text{RTS}}$ output of the serial channel may be input into an inverting line driver to obtain the proper polarity input at the modem or data set.

MCR Bit	Description
MCR(2)	When MCR(2) is set high, $\overline{OUT1}$ is forced low.
MCR(3)	When MCR(3) is set high, the $\overline{OUT2}$ output is forced low.
MCR(4)	MCR(4) provides a local loopback feature for diagnostic testing of the channel. When MCR(4) is set high, serial output (SOUT) is set to the marking (logic 1) state, and the receiver data input serial input (SIN) is disconnected. The output of the transmitter shift register is looped back into the receiver shift register input. The four modem control inputs ($\overline{CTS}$, $\overline{DSR}$, $\overline{DCD}$, and $\overline{RI}$) are disconnected. The modem control outputs ($\overline{DTR}$, $\overline{RTS}$, $\overline{OUT1}$, and $\overline{OUT2}$) are internally connected to the four modem control inputs. The modem control output pins are forced to their inactive state (high) on the TL16C552. In diagnostic mode, data transmitted is immediately received. This allows the processor to verify the transmit and receive data paths of the selected serial channel. Interrupt control is fully operational. However, interrupts are generated by controlling the lower four MCR bits internally. Interrupts are not generated by activity on the external pins represented by those four bits.
MCR(5) – MCR(7)	Permanently set to logic 0.

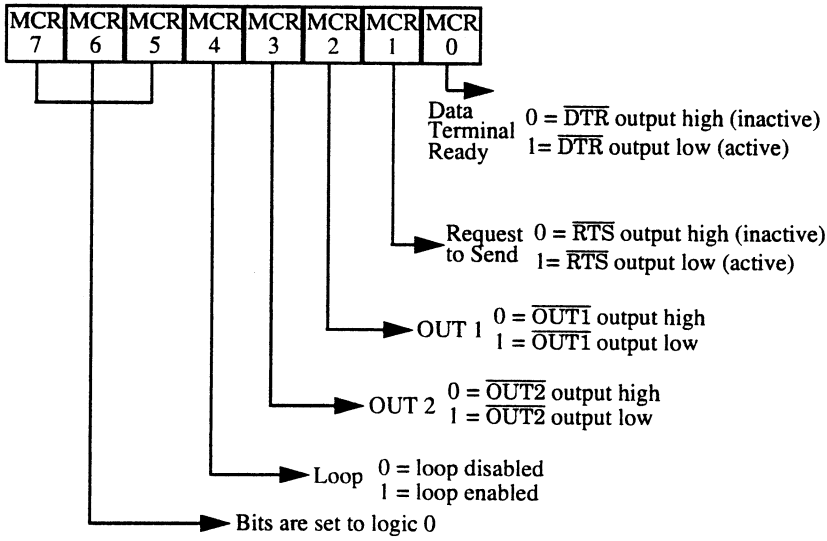


Figure 5-2. Modem Control Register Contents

5.5.6. Modem Status Register (MSR)

The MSR provides the CPU with status of the modem input lines from the modem or peripheral devices. The MSR allows the CPU to read the serial channel modem signal inputs by accessing the data bus interface of the ACE in addition to the current status of four bits of the MSR that indicate whether the modem inputs have changed since the last reading of the MSR. The delta status bits are set high when a control input from the modem changes state and reset low when the CPU reads the MSR.

The modem input lines are $\overline{CTS}$, $\overline{DSR}$, $\overline{RI}$, and $\overline{DCD}$. MSR(4) – MSR(7) are status indications of these lines. A status bit = 1 indicates the input is a low. A status bit = 0 indicates the input is high. If the modem status interrupt in the interrupt enable register is enabled (IER(3)), an interrupt is generated whenever MSR(0) - MSR(3) is set to a one. The MSR is a priority 4 interrupt. The contents of the modem status register are described in Table 5-10.

MSR Bit	Description
MSR(0) delta clear to send (DCTS) bit 0	DCTS displays that the $\overline{CTS}$ input to the serial channel has changed state since it was last read by the CPU.
MSR(1) delta data set ready (DDSR) bit 1	DDSR indicates that the $\overline{DSR}$ input to the serial channel has changed state since the last time it was read by the CPU.
MSR(2) trailing edge of ring indicator (TERI) bit 2	TERI indicates that $\overline{RI}$ input to the serial channel has changed state from low to high since the last time it was read by the CPU. High-to-low transitions on RI do not activate TERI.
MSR(3) delta data carrier detect (DDCD) bit 3	DDCD indicates that the $\overline{DCD}$ input to the serial channel has changed state since the last time it was read by the CPU.
MSR(4) clear to send (CTS) bit 4	CTS is the complement of the $\overline{CTS}$ input from the modem indicating to the serial channel that the modem is ready to receive data from the serial channel's transmitter output (SOUT). If the serial channel is in loop mode ((MCR(4)=1), MSR(4) reflects the value to RTS in the MCR.
MSR(5) data set ready (DSR) bit 5	DSR is the complement of the $\overline{DSR}$ input from the modem to the serial channel that indicates that the modem is ready to provide received data to the serial channel receiver circuitry. If the channel is in loop mode (MCR(4)=1), MSR(5) reflects the value of DTR in the MCR.

MSR Bit	Description
MSR(6) ring indicator (RI) bit 6	RI is the complement of the RI input. If the channel is in loop mode (MCR(4)=1), MSR(6) reflects the value of $\overline{OUT1}$ in the MCR.
MSR(7) data carrier detect (DCD) bit 7	Data carrier detect indicates the status of the data carrier detect ($\overline{DCD}$) input. If the channel is in loop mode (MCR(4)=1), MSR(7) reflects the value of OUT2 in the MCR. Reading the MSR register clears the delta modem status indications but has no effect on the other status bits. For LSR and MSR, the setting of status bits is inhibited during status register read operations. If a status condition is generated during a read $\overline{IOR}$ operation, the status bit is not set until the trailing edge of the read. If a status bit is set during a read operation, and the same status condition occurs, that status bit will be cleared at the trailing edge on the read instead of being set again. In loopback mode, when modem status interrupts are enabled, the $\overline{CTS}$, $\overline{DSR}$, $\overline{RI}$ and $\overline{DCD}$ input pins are ignored. However, a modem status interrupt may still be generated by writing to MCR3 – MCR0. Applications software should not write to the modem status register.

Table 5-10. Modem Status Register Bits

MSR Bit	Mnemonic	Description
MSR(0)	DCTS	Delta clear to send
MSR(1)	DDSR	Delta clear set ready
MSR(2)	TERI	Trailing edge of ring indicator
MSR(3)	DDCD	Delta data carrier detect
MSR(4)	$\overline{CTS}$	Clear to send
MSR(5)	$\overline{DSR}$	Data set ready
MSR(6)	$\overline{RI}$	Ring indicator
MSR(7)	$\overline{DCD}$	Data carrier detect

5.5.7. Divisor Latches

The ACE serial channel contains a programmable baud-rate generator (BRG) that divides the clock (DC to 8 MHz) by any divisor from 1 to 2^{16}-1. The output frequency of the baud generator is:

16x the data rate (divisor#=clock/(baud rate x 16)

This is referred to in this document as RCLK. Two 8-bit divisor latch registers store the divisor in a 16-bit binary format. These divisor latch registers must be loaded during initialization. Upon loading either of

the divisor latches, a 16-bit baud counter is immediately loaded. This prevents long counts on initial load. The BRG can use any of three different popular frequencies to provide standard baud rates. These frequencies are 1.8432 MHz, 3.072 MHz, and 8 MHz. With these frequencies, standard bit rates from 50 to 512K bps are available.

5.5.8. Scratchpad Register (SCR)

The scratchpad register is an 8-bit read/write register that has no effect on either channel in the ACE. It is intended to be used by you to hold data temporarily.

The Mask/Poll register for ports 1 through 8 is located at I/O BASE + 7, which is the scratch register for port 1. The Mask/Poll register for ports 9 through 16 is located at I/O BASE + 47, which is scratchpad register for port 8. Therefore, the scratchpad registers for ports 1 and 8 are not available.

5.5.9. Interrupt Identification Register (IIR)

In order to minimize software overhead during data character transfers, the serial channel prioritizes interrupts into four levels. The four levels of interrupt conditions are as follows:

1. Receiver line status (priority 1)
2. Received data ready (priority 2) or character timeout
3. Transmitter holding register empty (priority 3)
4. Modem status (priority 4)

Information indicating that a prioritized interrupt is pending and the type of interrupt is stored in the interrupt identification register (IIR). The IIR indicates the highest priority interrupt pending The contents of the IIR are indicated in Table 5-11.

Table 5-11. Interrupt Identification Register

FIFO Mode Only	Interrupt Identification Register			Interrupt Set and Reset Functions			
Bit 3	Bit 2	Bit 1	Bit 0	Priority Level	Interrupt Type	Interrupt Source	Interrupt Reset Control
0	0	0	1	-	None	None	-
0	1	1	0	1st	Receiver line status	OE, PE, FE, or BI	LSR read

Table 5-11. Interrupt Identification Register

FIFO Mode Only	Interrupt Identification Register			Interrupt Set and Reset Functions			
Bit 3	Bit 2	Bit 1	Bit 0	Priority Level	Interrupt Type	Interrupt Source	Interrupt Reset Control
0	1	0	0	2nd	Received data available	Receiver data available or trigger level reached	RBR read until FIFO drops below the trigger level
1	1	0	0	2nd	Character timeout indication	No characters have been removed from or input to the receiver FIFO during the last four character times and there is at least one character in it during this time.	RBR read
0	0	1	0	3rd	THRE	THRE	IIR read if THRE is the interrupt source or THR write
0	0	0	0	4th	Modem status	$\overline{CTS}$, $\overline{DSR}$, $\overline{RI}$, or $\overline{DCD}$	MSR read

IIR(0) Can be used to indicate whether an interrupt is pending. When IIR(0) is low, an interrupt is pending.

IIR(1) & IIR(2)
Are used to identify the highest priority interrupt pending as indicated in Table 5-11.

IIR(3 This bit is always logic 0 when in the TL16C450 mode. This bit is set along with bit 2 when in FIFO mode and a trigger change level interrupt is pending.

IIR(4) & IIR(5)
These two bits are always set to logic 0.

IIR(6) & IIR(7)
FCR(0)=1 sets these two bits.

5.5.10. Interrupt Enable Register (IER)

The interrupt enable register (IER) is used to independently enable the four serial channel interrupt sources that activate the interrupt (INT0 or INT1) output. All interrupts are disabled by resetting IER(0)-IER(3) of the interrupt enable register. Interrupts are enabled by setting the appropriate bits of the IER high. Disabling the interrupt system inhibits the interrupt identification register and the active (high) interrupt output. All other system functions operate in their normal manner, including the setting of the line status and modern status registers. The contents of the interrupt enable register are described below.

IER(0) When set to one, IER(0) enables the received data available interrupt and the timeout interrupts in FIFO mode.

IER(1) When set to one, IER(1) enables the transmitter holding register empty interrupt.

IER(2) When set to one, IER(2) enables the receiver line status interrupt.

IER(3) When set to one, IER(3) enables the modem status interrupt.

IER(4)& IER(7)
These four bits of the IER are logic 0.

5.5.11. Receiver

Serial asynchronous data is input into the SIN pin. The ACE continually searches for a high-to-low transition from the idle state. When the transmission is detected, a counter is reset, and counts the 16X clock to 7 1/2, which is the center of the start bit. The start bit is valid if the SIN is still low. Verifying the start bits prevents the receiver from assembling a false data character due to a low-going noise spike on the SIN input.

The line control register determines the number of data bits in a character [LCR(0), LCR(1)]. If parity is used, LCR(3) and the polarity of parity LCR(4) are needed. Status for the receiver is provided in the line status register. When a full character is received, including parity and stop bits, the data received indication in LSR(0) is set high. The CPU reads the receiver buffer register, which resets LSR(0). If the character is not read prior to a new character transfer from the RSR to the RBR, the overrun error status indication is set in LSR(1). If there is a parity error, the parity error is set in LSR(2). If a stop bit is not detected, a framing error indication is set in LSR(3).

If the data into SIN is a symmetrical square wave, the center of the data cells will occur within –+ 3.125% of the actual center, providing an error margin of 46.875%. The start bit can begin as much as one 16x clock cycle prior to being detected.

5.5.12. Master Reset

After power up, the ACE $\overline{\text{RESET}}$ input should be held low for one microsecond to reset the ACE circuits to an idle mode until initialization. A low on $\overline{\text{RESET}}$ causes the following:

1. Initializes the transmitter and receiver clock counters.
2. Clears the line status register(LSR), except for transmitter shift register empty (TEMT) and transmit holding register empty (THRE), which are set. The modem control register (MCR) is also cleared. All of the discrete lines, memory elements, and miscellaneous logic associated with these register bits are also cleared or turned off. The line control register (LCR), divisor latches, receiver buffer register, and transmitter buffer register are not affected.

Following the removal of the reset condition ($\overline{\text{RESET}}$ high), the ACE remains in idle mode until programmed.

A hardware reset of the ACE sets the THRE and TEMT status bit in the LSR. When interrupts are subsequently enabled, an interrupt occurs due to THRE.

5.5.13. Programming

The serial channel of the ACE is programmed by the control registers:

- LCR
- IER
- DLL
- DLM
- MCR
- FCR

These control words define the following

- Character length
- Number of stop bits
- Parity
- Baud rate
- Modem interface

While the control registers can be written in any order, the IER should be written last, because it controls the interrupt enables. Once the serial channel is programmed and operational, these registers can be updated any time the ACE serial channel is not transmitting or receiving data.

5.5.14. FIFO-Interrupt-Mode Operation

The following RCVR status occurs when the RCVR FIFO and receiver interrupts are enabled:

1. LSR(0) is set when a character is transferred from the shift register to the RCVR FIFO. When the FIFO is empty, it is reset.
2. IIR=06 receiver line status interrupt has higher priority than the received data available interrupt IIR=04.
3. Received data available interrupt will be issued to the CPU when the programmed trigger level is reached by the FIFO. As soon as the FIFO drops below its programmed trigger level, it will be cleared.
4. IIR=04 (received data available indication) also occurs when the FIFO reaches its trigger level. It is cleared when the FIFO drops below the programmed trigger level.

The following RCVR FIFO character timeout status occurs when RCVR FIFO and receiver interrupts are enabled.

1. If the following conditions exist, a FIFO character timeout occurs
   - Minimum of one character in FIFO.
   - Last received serial character was longer than 4 continuous previous character times ago (if two stop bits are programmed, the second one is included in the time delay).
   - The last CPU read of the FIFO was more than 4 continuous character times earlier. At 300 baud and 12-bit characters, the FIFO timeout interrupt causes a latency of 160 ms maximum from received character to interrupt issued.
2. By using the RCLK input for a clock signal, the character times can be calculated. The delay is proportional to the baud rate.
3. The timeout timer is reset after the CPU reads the RCVR FIFO or after a new character is received, when there has been no timeout interrupt.
4. A timeout interrupt is cleared and the timer is reset when the CPU reads a character from the RCVR FIFO.

XMIT interrupts occur as follows when the transmitter and XMIT FIFO interrupts are enabled (FCRO=1, IER=1).

1. When the transmitter FIFO is empty, the transmitter holding register interrupt (IIR=02) occurs. The interrupt is cleared as soon as the transmitter holding register is written to or the IIR is read. One to sixteen characters may be written to the transmit FIFO when servicing this interrupt.
2. The XMIT FIFO empty indications will be delayed one character time minus the last stop bit time whenever the following occurs:
   - THRE=1 and there has not been a minimum of two bytes at the same time in XMIT FIFO, since the last THRE=1.
   - The first transmitter interrupt after changing FCR0 will be immediate, however, assuming it is enabled.

RCVR FIFO trigger level and character timeout interrupts have the same priority as the received data available interrupt. The transmitter holding register empty interrupt has the same priority as the transmitter FIFO empty interrupt.

Appendix A. Troubleshooting and Support

A.1. Troubleshooting

If installation fails or you are trying to resolve a problem, you should verify the following before calling Comtrol technical support line:

- Reinstall the controller selecting a different I/O address range (see Table A-1 for common I/O addresses).
- Check the signals between your peripherals and the interface box to verify that they match (if applicable).
- Check to make sure the cables are connected properly.
- Reseat the controller in the slot (the power should be off).
- Reboot the system.

If you have not been able to get the controller operating:

1. Turn off your PC and insert the diagnostic diskette.
2. Boot the PC and follow the instructions provided by the diagnostic diskette.

Table A-1. System I/O Addresses

Address Block	Addresses Used	Description
000 – 03F		Reserved for Motherboard
040 – 07F		Reserved for Motherboard
080 – 0BF		Reserved for Motherboard
0C0 – 0FF		Reserved for Motherboard
100 – 13F		
140 – 17F		
180 – 1BF		
1C0 – 1FF	1F0 – 1F8	Fixed Disk
200 – 23F	218 – 21B	Comtrol controllers
240 – 27F	278 – 27F	LPT2
280 – 2BF		
2C0 – 2FF	2E8 – 2EF 2F8 – 2FF	COM4 COM2
300 – 33F	318 – 31B	Comtrol controllers

Table A-1. System I/O Addresses (Continued)

Address Block	Addresses Used	Description
340 – 37F	378 – 37F	LPT1
380 – 3BF	3B0 – 3BF	Monochrome display and LPT3
3C0 – 3FF	3D0 – 3DF 3E8 – 3EF 3F0 – 3F7 3F8 – 3FF	Graphics Monitor Adapter COM3 Floppy Disk Controller COM1

A.2. Placing a Support Call

Before you place a technical support call to Comtrol, please make sure that you have the following information. The following table provides you with an area to enter the required information.

Table A-2. Support Call Information

Item	Your System Information
Controller type	
Interface type	
I/O address	
Interrupt (IRQ) number	
Operating system type and release	
Device driver release number	
PC make, model, and speed	
List of other devices in the PC and their addresses	
List any error messages from the system, device driver, or diagnostic software	

After you have gathered this information, contact Comtrol using one of the following methods:

- email: **support@Comtrol.com**
- FAX: (612) 631-8117 (US) or (44) 869-323-211 (UK)
- Toll free: (800) 926-6876 (US)
- Phone: (612) 631-7654 (US) or (44) 869-323-220

Appendix B. Warranty

Comtrol Corporation provides:

- A 30-day money-back guarantee
- A limited five (5) year warranty* (US and Canada)
- Support for your Comtrol controller for 5 years from the date of purchase.

*Check with your distributor for guarantee conditions in countries other than the U.S.A. and Canada.

B.1. Limited Warranty

Comtrol Corporation, Inc. ("the Company") and its affiliate (Comtrol Europe, Ltd.) make no representations or warranties, expressed or implied including warranties of merchantability, noninfringement, and fitness for a particular purpose except as provided below.

B.2. Hardware

Comtrol warrants to the original purchaser that its controller is free of defect in design, materials and workmanship for five years from the date of delivery of a new controller. Comtrol (or its authorized repair center), at its option, will repair or replace, at the business location of Comtrol each part of the controller which is proven to the satisfaction of Comtrol to have been defective in design, material or workmanship.

This warranty shall not apply to any part of the controller which, in the judgment of Comtrol, has been subjected to misuse, negligence, alteration, accident, improper maintenance, or damage by excessive physical or electrical stress. Adjustment of the controller, where warning labels and operation manuals warn against such adjustments, will void this warranty.

This warranty is void if the serial number of the Comtrol controller has been defaced, altered or removed. This warranty does not apply to expendable components such as fuses or bulbs. Repair and replacement parts will be furnished on an exchange basis and may be either reconditioned or new. All replaced parts or controllers become the property of Comtrol.

The sole remedy for breach of warranty shall be repair, replacement, or refund, at the option of Comtrol, of the defective product provided as follows.

B.3. Software

Comtrol warrants that for a period of ninety (90) days from the date of delivery to you as evidenced by a copy of your receipt, the disks on which the program is furnished will under normal use be free from defects in materials and workmanship and the program under normal use will perform substantially in accordance with the documentation without significant errors that make it unusable.

Comtrol's entire liability and your exclusive remedy under this warranty (which is subject to you returning the program to Comtrol or an authorized dealer with a copy of your receipt) will be, at Comtrol's option, to attempt to correct or help you around errors with efforts that Comtrol believes suitable to the problem, to replace the program or disks with functionally equivalent software or disks, as applicable, or to refund the purchase price and terminate this Agreement.

No Comtrol dealer, distributor, agent or employee is authorized to modify this warranty.

Comtrol does not warrant that the functions contained in the programs will meet your requirements or that the operation of the programs will be uninterrupted or error-free. You assume the responsibility for the selection of the programs and hardware to achieve your intended results and for the installation, use and results obtained from the programs.

Some programs contained on disk are specifically for and have been optimized to run with Comtrol products. Therefore, the programs on these disks will not run effectively and will cause errors in data or operation when this software is attempted to be used with non-Comtrol products.

This warranty shall not apply if the serial number has been defaced, altered or removed, or if the software has been altered in any fashion.

B.4. Return Procedure

To qualify for the previously discussed warranty, the original purchaser must follow the procedure outlined below:

1. Comtrol must be notified in writing within thirty (30) days of the date that the defect is discovered. Comtrol will then issue a Return Material Authorization (RMA) Number which the purchaser must include with all correspondence and display on the outside of the shipping container when returning the controller.
2. All Comtrol controllers must be shipped freight and insurance prepaid, in the original shipping container, or in a container providing equal or better protection, with the Return Material Authorization (RMA) Number displayed on the outside of the container in a prominent manner.

3. A written description of the defect together with a copy of your receipt or other proof of purchase, and the name of the dealer which sold you the Comtrol product, must be shipped with the controller. All defects must be reproducible at Comtrol's location to qualify for this limited warranty. Ship the controller to:

   Comtrol Corporation
   2675 Patton Road, Dock D
   Saint Paul, MN 55113

Comtrol will return a controller which qualifies under this warranty freight and insurance prepaid. Comtrol will repair or replace the controllers that do not qualify under the terms of this warranty at the option of the purchaser, in which case the purchaser will pay the cost of repair or replacement, and return freight and insurance.

This limited warranty is in lieu of all other warranties and conditions expressed, implied or statutory including merchantability, fitness for purpose, non-infringement, course of dealing, trade or performance and all other liabilities of Comtrol all of which are hereby disclaimed.

In no event will Comtrol be liable for damages, including lost profits, lost savings or other special, punitive, incidental, or consequential damages arising out of the use of or inability to use the Comtrol controller, even if Comtrol or an authorized dealer has been advised of the possibility of such damages, or for any claim by any other party. This warranty gives you specific legal rights and you may also have other rights that vary from state to state (U.S.) or in your home country.

B.5. Limited Liability

Independent of the warranty or any other agreement between you and Comtrol, regardless of the basis for any claim, neither Comtrol nor anyone else who has been involved in the creation, production, or delivery of this software or hardware shall be liable for any direct, indirect, consequential or incidental damages; Comtrol's maximum liability shall be limited to refund of the purchase price. Some consumer laws may not allow the limitation or exclusion of incidental or consequential damages for consumer products, so the above limitations or exclusions may not apply to you. The price of the materials and programs reflects this allocation of risk.

B.6. Technical Support

If you have questions about your controller or need assistance, contact Comtrol by email, FAX , or phone.

email: support@Comtrol.com

FAX: (612) 631-8117 (US) or (44) 869-323-211 (UK)

Toll free: (800) 926-6876 (US)

Phone: (44) 869-323-220 (UK) or (612) 631-7654 (US)

BBS: (612) 631-8310

Note: *The BBS supports modem speeds up to 28.8 Kbps with 8 bits and no parity.*

Comtrol has a staff of hardware and software engineers, technicians, and managers available to help you.

Index

S