Multiprotocol Adapter/A Technical Reference

68X2220

80X0837

The warranty terms and conditions applicable in the country of purchase in respect of an IBM product are available from the supplier. Please retain them with your proof of purchase.

No statements contained in this documentation shall affect the statutory rights of consumers.

First Edition (April, 1987)

The following paragraph does not apply to the United Kingdom or any country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This publication could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time.

It is possible that this publication may contain reference to, or information about, IBM products (machines and programs), programming, or services that are not announced in your country. Such references or information must not be construed to mean that IBM intends to announce such IBM products, programming, or services in your country.

The publication of the information contained herein is not intended to and does not convey any rights or licenses under any IBM patents or copyrights.

Requests for copies of this publication and for technical information about IBM products should be made to your IBM Authorized Dealer or your IBM Marketing Representative.

Contents

Description 1
Control Signal Interface 2
Protocol Controllers 2
Programmable Option Select (POS) 3
POS Registers 0 and 1 3
POS Register 2 4
POS Register 3 5
Asynchronous Communications 6
16550 Description 6
Programming the 16550 7
Transmitter Holding Register 7
Receiver Buffer Register 7
Programmable Baud-Rate Generator 8
Interrupt Enable Register 9
Interrupt Identification Register 10
Line Control Register 11
Modem Control Register 13
Line Status Register 15
Modem Status Register 16
Scratch Pad Register 17
Address and Interrupt Assignments 18
Serial Data Format 20
Binary Synchronous Communications 21
8251A Description 21
Programming the 8251A 22
Mode and Command Instructions 22
Asynchronous Mode 23
Synchronous Mode 25
Command Instructions 28
8254 Description 30
Counter Descriptions 30
Programming the 8254 31
Write Operations 32
Read Operations 34
Simple Read Operation 34
Counter Latch Command 35
Read-Back Command 36
Counter Status Command 37

Mode Definitions 39
Mode 0 - Interrupt on Terminal Count 40
Mode 1 - Hardware Retriggerable One-Shot 40
Mode 2 - Rate Generator 41
Mode 3 - Square Wave Mode 42
Mode 4 - Programmable Triggered Strobe 43
Mode 5 - Hardware Triggered Strobe (Retriggerable) ... 43
Operations Common to All BSC Modes 45
Gate 45
Counter 45
Typical Programming Sequence 46
Address and Interrupt Assignments 48
SDLC/HDLC Communications 49
8273 Description 49
System Interface 50
Modem Interface 55
Programming the 8273 58
8273 Status Register 58
Operating Mode Register 59
Serial I/O Mode Register 61
Data Transfer Mode Register 61
8273 Command Descriptions 66
Initialization/Configuration Commands 66
Receive Commands 67
Transmit Commands 72
Reset Command 74
Modem Control Commands 75
8255 I/O Port Assignment 75
Address, DMA, and Interrupt Assignments 77
DMA Assignment 77
Interrupt Information 77
RS-232-C Interface 78
Modem Control Input Signals 78
Modem Control Output Signals 79
Voltage Interchange Information 80
Connector 81

Index **83**

Figures

1. Multiprotocol Adapter/A Block Diagram 1
2. Register Selection 3
3. POS Register 2 4
4. Mode Selection 4
5. POS Register 3 5
6. Transmitter Holding Register 7
7. Receiver Buffer Register 7
8. Divisor Latch (LSB) 8
9. Divisor Latch (MSB) 8
10. Baud Rates at 1.8432 MHz 9
11. Interrupt Enable Register 9
12. Interrupt Identification Register 10
13. Interrupt Control Functions 11
14. Line Control Register 11
15. Word Length (Stop Bits) 12
16. Word Length 13
17. Modem Control Register 13
18. Line Status Register 15
19. Modem Status Register 16
20. I/O Address and Interrupt Level Assignments (Async) 18
21. Serial Data Format 20
22. Typical Data Block 22
23. Asynchronous Mode Instruction Format 24
24. Number of Stop Bits 25
25. Character Rate Factor 25
26. Baud Rate Factor 25
27. BSC Mode Instruction Format 27
28. Character Length 27
29. Command Instruction Format 28
30. Status Read Format 29
31. Control Word Register 31
32. Select Counter 32
33. Read/Write Options 32
34. Counter Mode Selection 32
35. Sequence Example 1 33
36. Sequence Example 2 33
37. Sequence Example 3 34
38. Sequence Example 4 34

39. Counter Latch Command Format 35
40. Counter Selection 35
41. Read-Back Command Format 37
42. Counter Status Format 37
43. Null Count Operation 38
44. Read-Back Command Example 39
45. GATE Operations Summary 44
46. Minimum and Maximum Initial Counts 44
47. Port A Assignments 46
48. Port B Assignments 47
49. Port C Assignments (Input) 47
50. Port C Assignments (Output) 47
51. I/O Address Assignemnts (BSC) 48
52. SDLC Frame Format 49
53. 8273 Register Selection 50
54. DMA, Interrupt-Driven System 52
55. 8273 Port A (Input) 55
56. 8273 Port B (Output) 56
57. Transmit/Receive Timing 57
58. Status Register Format 58
59. Operating Mode Register 59
60. Serial I/O Mode Register 61
61. Data Transfer Mode Register 61
62. 1-Bit Delay Mode Register 62
63. Command Phase Flowchart 63
64. Command Summary Key 66
65. Initialization/Configuration Command Summary 67
66. Receive Command Summary 68
67. Interrupt Result Code 68
68. Receiver Interrupt Result Codes (RIC) 69
69. Transmit Command Summary 73
70. Transmit Interrupt Result (TIC) Codes 73
71. Port A (Input) Modem Control Command Summary 75
72. Port B (Output) Modem Control Command Summary 75
73. Port A Assignments 76
74. Port B Assignments 76
75. Port C Assignments (Input) 76
76. Port C Assignments (Output) 76
77. I/O Address Assignments (SDLC/HDLC) 77
78. Voltage Levels 80
79. Pin Locations 81
80. EIA/CCITT Pin Assignments 81

Description

The Multiprotocol Adapter/A provides multiple communications features and is designed to work in systems that use the Micro Channel™ architecture. For more information about the channel, refer to the system *Technical Reference* manual.

The adapter provides the following features:

- ISO/CCITT level 2 and level 1 protocol:
  - Level 2: Asynchronous, Bisynchronous (BSC), and Synchronous Data Link Control (SDLC)/High-level Data Link Control (HDLC) protocol
  - Level 1: EIA RS-232C interface.
- One 25-pin, D-type connector
- Internal wrap-back test loop for diagnostics
- Programmable arbitration level for direct DMA transfers on SDLC/HDLC communications.

The following is a block diagram of the adapter:

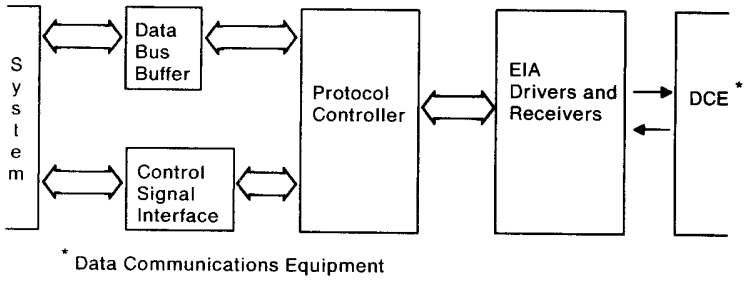

Figure 1. Multiprotocol Adapter/A Block Diagram

Micro Channel is a trademark of the International Business Machines corporation.

Control Signal Interface

The control signal interface performs the following functions:

- I/O read and write signal generation and wait-state insertion
- Programmable Option Select (POS)
- Local Arbiter for DMA operation
- Address decode and protocol mode selection.

Protocol Controllers

Communications protocols are performed by the following controllers and circuits on the adapter:

- NS16550 Asynchronous Communications Element for ASYNC
- 8251A Universal Synchronous/Asynchronous Receiver/Transmitter (USART) for BSC
- 8254 Programmable Interval Timer (provides time-outs and interrupt functions in synchronous communications)
- 8273 SDLC/HDLC Protocol Controller
- 8255-5 Programmable Peripheral Interface (for expanded modem interface and diagnostic aid in synchronous communications)
- Multiplexer.

Programmable Option Select (POS)

Before the adapter can be used it must be initialized by the system. The Programmable Option Select (POS) registers eliminate the need to manually initialize the adapter for a particular configuration.

The adapter contains several configurable options. These options are controlled by the POS registers.

Warning: IBM recommends that programmable options be set only through the System Configuration Utilities. Direct setting of the POS registers and/or CMOS RAM POS parameters can result in multiple assignment of the same system resource, improper operation of the feature, loss of data, or possible damage to the system or options. If application programs use adapter identification (ID) information, compatibility problems between systems or options may result.

The POS registers are only accessible by the system when -CD SETUP on the channel is active. Address lines A0 through A2 select the registers as shown in the following figure:

-CD SETUP	Address Line			Register
	A2	A1	A0	
L	L	L	L	POS Register 0 - Adapter ID
L	L	L	H	POS Register 1 - Adapter ID
L	L	H	L	POS Register 2 - Option Select
L	L	H	H	POS Register 3 - Option Select
H	X	X	X	Normal Operation

Figure 2. Register Selection

POS Registers 0 and 1

These read-only registers contain the hard-wired adapter ID.

When -CD SETUP is active, the hard-wired adapter ID (hex DEFF) is read by the system. POS Register 1 is the most-significant byte and POS Register 0 is the least-significant byte of the adapter ID.

POS Register 2

Programmable adapter options are written to this read/write register. Bit definitions for POS register 2 are shown in the following figure:

Bit	Function
7 - 6	Reserved = 0
5	Reserved = 1
4 - 1	Mode Selection
0	Adapter Enable

Figure 3. POS Register 2

Notes:

1. Bits 0 and 4 are cleared to 0 during power-on.
2. The alternate BSC port has the same I/O address as the SDLC port (0380-038F). The alternate SDLC port has the same I/O address as the BSC port (03A0-03AF).

Bits 7 - 6 Reserved. These bits are always cleared to 0.

Bit 5 Reserved. This bit is always set to 1.

Bits 4 - 1 These bits select modes as shown in the following figure:

4 3 2 1	Mode Selected
0 0 0 0	Serial 1
0 0 0 1	Serial 2
0 0 1 0	Serial 3
0 0 1 1	Serial 4
0 1 0 0	Serial 5
0 1 0 1	Serial 6
0 1 1 0	Serial 7
0 1 1 1	Serial 8
1 0 0 0	SDLC
1 0 0 1	Alternate SDLC
1 0 1 0	Reserved
1 0 1 1	Reserved
1 1 0 0	BSC
1 1 0 1	Alternate BSC
1 1 1 0	Reserved
1 1 1 1	Reserved

Figure 4. Mode Selection

Bit 0 When this bit is set to 1, the adapter is enabled. When this bit is cleared to 0 the adapter is disabled. When the adapter is disabled, it does not respond to any I/O or Memory commands except the Memory Refresh command and POS reads and writes. No interrupt requests are driven. A 'channel reset' clears this bit to 0.

POS Register 3

Programmable adapter options are written to this read/write register. Bit definitions for POS register 3 are shown in the following figure:

Bit	Function
7 - 4	Reserved
3 - 0	Arbitration Level

Figure 5. POS Register 3

Note: All bits in POS Register 3 are cleared to 0 during power-on.

Bits 7 - 4 Reserved. These bits are always cleared to 0.

Bits 3 - 0 These bits specify the arbitration priority level for the adapter. The highest priority is selected with the binary decode of 0000. When competing for the channel, the adapter places this value on the '-arb' lines. If no other device places a higher priority request (lower binary value) on these lines, the adapter wins the arbitration.

Note: Arbitration level 1 is recommended for primary SDLC communications.

Asynchronous Communications

The asynchronous communications portion of the adapter is fully programmable.

The major component of this portion of the adapter is a NS16550 LSI Asynchronous Communications Controller. The NS16550 controller is functionally compatible to the NS16450 controller. To programs, the NS16550 appears to be identical to the IBM Personal Computer AT® Serial/Parallel adapter. Support for the controller on the adapter is restricted to the functions which are identical to the NS16450. Using the controller in the FIFO mode may result in non-detectable data errors.

16550 Description

The NS 16550 adds and removes a start bit, parity bit, and stop bits. A programmable baud-rate generator allows operation from 50 baud to 19,200 baud. The adapter supports 5-, 6-, 7-, and 8-bit characters with 1, 1-1/2, or 2 stop bits. A fully prioritized interrupt system controls transmit, receive, error, line status, and data set interrupts. Diagnostic capabilities provide loopback functions of transmit/receive and input/output signals. Total system load (for example, competing I/O devices) will limit the practical total number of concurrent ports and the speed of each port. Other features include:

- Full double buffering in the character mode, eliminating the need for precise synchronization
- False-start bit detection
- Line-break generation and detection
- Modem control functions.

AT is a registered trademark of the International Business Machines Corporation.

Programming the 16550

The NS16550 controller has a number of accessible registers. The system uses these registers to control adapter operations and to transmit and receive data.

Transmitter Holding Register

The Transmitter Holding register contains the character to be sent. Bit 0 is the least-significant bit and the first bit sent serially as shown in the following figure:

Bit	Function
7	Data Bit 7
6	Data Bit 6
5	Data Bit 5
4	Data Bit 4
3	Data Bit 3
2	Data Bit 2
1	Data Bit 1
0	Data Bit 0

Figure 6. Transmitter Holding Register

Receiver Buffer Register

The Receiver Buffer register contains the received character. Bit 0 is the least-significant bit and the first bit received serially as shown in the following figure:

Bit	Function
7	Data Bit 7
6	Data Bit 6
5	Data Bit 5
4	Data Bit 4
3	Data Bit 3
2	Data Bit 2
1	Data Bit 1
0	Data Bit 0

Figure 7. Receiver Buffer Register

Programmable Baud-Rate Generator

The controller has a programmable baud-rate generator that can divide the clock input (1.8432 MHz) by any divisor from 1 to 65,535. The output frequency of the baud-rate generator is the baud rate multiplied by 16. Two 8-bit divisor latches store the divisor in a 16-bit binary format. These divisor latches must be loaded during adapter initialization to ensure desired operation of the baud-rate generator. When either of the divisor latches is loaded, a 16-bit baud counter is immediately loaded. This prevents long counts on the first load.

The divisor latches are shown in the following figures:

Divisor Latch (Least Significant Byte)

Bit	Function
7	Bit 7
6	Bit 6
5	Bit 5
4	Bit 4
3	Bit 3
2	Bit 2
1	Bit 1
0	Bit 0

Figure 8. Divisor Latch (LSB)

Divisor Latch (Most Significant Byte)

Bit	Function
7	Bit 7
6	Bit 6
5	Bit 5
4	Bit 4
3	Bit 3
2	Bit 2
1	Bit 1
0	Bit 0

Figure 9. Divisor Latch (MSB)

Figure 10 on page 9 illustrates the use of the baud-rate generator with a frequency of 1.8432 MHz. For baud rates of 19,200 and below, the error obtained is minimal.

Note: In no case should the data speed be greater than 19,200 baud.

Desired Baud Rate	Divisor Used to Generate 16x Clock (Decimal)	(Hex)	Percent Error Difference Between Desired and Actual
50	2304	0900	--
75	1536	0600	--
110	1047	0417	0.026
134.5	857	0359	0.058
150	768	0300	--
300	384	0180	--
600	192	00C0	--
1200	96	0060	--
1800	64	0040	--
2000	58	003A	0.69
2400	48	0030	--
3600	32	0020	--
4800	24	0018	--
7200	16	0010	--
9600	12	000C	--
19200	6	0006	--

Figure 10. Baud Rates at 1.8432 MHz

Interrupt Enable Register

This 8-bit register allows the four types of controller interrupts to separately activate the 'chip-interrupt' output signal. The interrupt system can be totally disabled by clearing bits 0 through 3 of the Interrupt Enable register. Similarly, by setting the appropriate bits of this register to 1, selected interrupts can be enabled. Disabling the interrupts will inhibit the 'chip-interrupt' output signal from the controller. All other system functions operate normally, including the setting of the Line Status and Modem Status registers.

Bit	Function
7 - 4	Reserved = 0
3	Modem-Status Interrupt
2	Receiver-Line-Status Interrupt
1	Transmitter-Holding-Register-Empty Interrupt
0	Received Data Available Interrupt

Figure 11. Interrupt Enable Register

Bits 7 - 4 Reserved. These bits are always cleared to 0.

Bit 3 When set to 1, this bit enables the modem-status interrupt.

Bit 2 When set to 1, this bit enables the receiver-line-status interrupt.

Bit 1 When set to 1, this bit enables the transmitter-holding-register-empty interrupt.

Bit 0 When set to 1, this bit enables the received-data-available interrupt.

Interrupt Identification Register

In order to minimize programming overhead during data character transfers, the controller prioritizes interrupts into four levels:

- Priority 1 - Receiver-line-status
- Priority 2 - Received-data-available
- Priority 3 - Transmitter-holding-register-empty
- Priority 4 - Modem status.

Information about a pending interrupt is stored in the Interrupt Identification register. When the Interrupt Identification register is addressed, the pending interrupt with the highest priority is held and no other interrupts are acknowledged until the system microprocessor services that interrupt.

Bit	Function
7 - 3	Reserved = 0
2	Interrupt ID, Bit 1
1	Interrupt ID, Bit 0
0	Interrupt Pending = 0

Figure 12. Interrupt Identification Register

Bits 7 - 3 Reserved. These bits are always cleared to 0.

Bits 2 - 1 These two bits identify the pending interrupt with the highest priority as shown in Figure 13 on page 11.

Bit 0 When this bit is set to 1, no interrupt is pending and polling (if used) continues. When this bit is cleared to 0, an interrupt is pending and the contents of this register can be used as a pointer to the appropriate interrupt service routine.

This bit can be used in either hard-wired, prioritized, or polled conditions to indicate if an interrupt is pending.

Bits 2 - 0 select Interrupt Control Functions as shown in the following figure:

Bits 2 1 0	Priority	Type	Cause	Interrupt Reset Control
0 0 1	-	None	None	-
1 1 0	Highest	Receiver Line Status	Overrun, Parity, or Framing Error or Break Interrupt	Read the Line Status Register
1 0 0	Second	Received Data Available	Data in Receiver Buffer	Read the Receiver Buffer Register
0 1 0	Third	Transmitter Holding Register Empty	Transmitter Holding Register is Empty	Read Interrupt Identification Register or Write to Transmitter Holding Register
0 0 0	Fourth	Modem Status	Change in Signal Status from Modem	Read the Modem Status Register

Figure 13. Interrupt Control Functions

Line Control Register

The format of asynchronous communications is programmed through the Line Control register.

Bit	Function
7	Divisor Latch Access Bit
6	Set Break
5	Stick Parity
4	Even Parity Select
3	Parity Enable
2	Number of Stop Bits
1	Word Length Select, Bit 1
0	Word Length Select, Bit 0

Figure 14. Line Control Register

Bit 7 This bit must be set to 1 during a read or write operation to gain access to the divisor latches of the baud-rate generator. It must be cleared to 0 to gain access to the Receiver Buffer, Transmitter Holding, or Interrupt Enable registers.

Bit 6 When this bit is set to 1, set-break is enabled and serial output is forced to the spacing state and remains there regardless of other transmitter activity. When this bit is cleared to 0, set-break is disabled.

Bit 5 When bits 5, 4, and 3 are set to 1, the parity bit is sent and checked as a logical 0. When bits 5 and 3 are set to 1, and bit 4 is cleared to 0, the parity bit is sent and checked as a logical 1.

Bit 4 When this bit and bit 3 are set to 1, an even number of logical 1's are transmitted and checked in the data word bits and parity bit. When this bit is cleared to 0, and bit 3 is set to 1, an odd number of logical 1's are transmitted and checked in the data word bits and parity bit.

Bit 3 When set to 1, a parity bit is generated (transmit data) or checked (receive data) between the last data word bit and stop bit. (The parity bit is used to produce an even or odd number of 1's when the data-word bits and the parity bit are summed.)

Bit 2 This bit, along with bits 0 and 1, specifies the number of stop bits in each serial character that is sent or received as shown in the following figure:

Bit 2	Word Length *	Number of Stop Bits
0	N/A	1
1	5-Bits	1-1/2
1	6-Bits	2
1	7-Bits	2
1	8-Bits	2

* Word Length is specified by bits 0 - 1 in this register.

Figure 15. Word Length (Stop Bits)

Bits 0 - 1 These two bits specify the number of bits in each serial character that is sent or received. Word length is selected as shown in the following figure:

Bits 1 0	Word Length
0 0	5-Bits
0 1	6-Bits
1 0	7-Bits
1 1	8-Bits

Figure 16. Word Length

Modem Control Register

This 8-bit register controls the data exchange with the modem, data set, or peripheral device emulating a modem.

Bit	Function
7 - 5	Reserved = 0
4	Loop
3	Out 2
2	Out 1
1	Request-to-Send
0	Data-Terminal-Ready

Figure 17. Modem Control Register

Bits 7 - 5 Reserved. These bits are always cleared to 0.

Bit 4 This bit provides a loopback feature for diagnostic testing of the adapter. When bit 4 is set to 1:

- Transmitter-serial-output is set to the marking state
- Receiver-serial-input is disconnected
- Output of the Transmitter Shift register is "looped back" to the Receiver Shift register input

  Note: The Transmitter and Receiver Shift registers are not accessible NS16550 registers.

- The modem control inputs (CTS, DSR, DCD, and RI) are disconnected

- The modem control outputs (DTR, RTS, OUT 1, and OUT 2) are internally connected to the four modem control inputs
- The modem control output pins are forced inactive.

When the adapter is in the diagnostic mode, transmitted data is immediately received. This feature allows the system microprocessor to verify the transmit-data and receive-data paths of the adapter.

When the adapter is in the diagnostic mode, the receiver and transmitter interrupts are fully operational. The modem control interrupts are also operational, but their sources are the lower four bits of the Modem Control register instead of the four modem control input signals. The interrupts are still controlled by the Interrupt Enable register.

Bit 3 This bit controls the 'output 2' (OUT 2) signal which is an auxiliary user-designated interrupt enable signal. OUT 2 is used to control the interrupt signal to the channel. Setting this bit to 1 enables the interrupt. Clearing this bit to 0 disables the interrupt.

Bit 2 This bit controls the 'output 1' (OUT 1) signal which is an auxiliary user-designated output signal. When set to 1, OUT 1 is forced active. When cleared to 0, OUT 1 is forced inactive.

Bit 1 This bit controls the 'request-to-send' (RTS) controller output signal. When set to 1, RTS is forced active. When cleared to 0, RTS is forced inactive.

Bit 0 This bit controls the 'data-terminal-ready' (DTR) controller output signal. When set to 1, DTR is forced active. When cleared to 0, DTR is forced inactive.

Line Status Register

This 8-bit register provides the system microprocessor with status information about the data transfer.

Bit	Function
7	Reserved = 0
6	Transmitter Shift Register Empty
5	Transmitter Holding Register Empty
4	Break Interrupt
3	Framing Error
2	Parity Error
1	Overrun Error
0	Data Ready

Figure 18. Line Status Register

Bit 7 Reserved. This bit is always cleared to 0.

Bit 6 This bit is set to 1 when the Transmitter Holding register and the Transmitter Shift register are both empty. This bit is cleared to 0 when either the Transmitter Holding register or the Transmitter Shift register contains a data character.

Bit 5 This bit indicates that the adapter is ready to accept a new character for transmission. This bit is set to 1 when a character is transferred from the Transmitter Holding register into the Transmitter Shift register. This bit is cleared to 0 when the system microprocessor loads the Transmitter Holding register.

This bit also causes the adapter to issue an interrupt to the system microprocessor when bit 1 in the Interrupt Enable register is set to 1.

Bit 4 This bit is set to 1 when the received data input is held in the spacing state for longer than a fullword transmission (the total time of the start bit + data bits + parity + stop bits).

Note: Bits 1 through 4 are the error conditions that produce a receiver line-status interrupt whenever any of the corresponding conditions are detected.

Bit 3 This bit is set to 1 when the stop bit following the last data bit or parity bit is at a spacing level. This indicates that the received character did not have a valid stop bit.

Bit 2 This bit is set to 1 when a parity error is detected (the received character does not have the correct even or odd parity, as selected by the even-parity-select bit). This bit is cleared to 0 when the system microprocessor reads the contents of the Line Status register.

Bit 1 When it is set to 1, this bit indicates that data in the Receiver Buffer register was not read by the system microprocessor before the next character was transferred into the Receiver Buffer register, destroying the previous character. This bit is cleared to 0 when the system microprocessor reads the contents of the Line Status register.

Bit 0 This bit is set to 1 when a complete incoming character has been received and transferred into the Receiver Buffer register. This bit is cleared to 0 by reading the Receiver Buffer register.

Modem Status Register

This 8-bit register provides the current state of the control lines from the modem (or external device) to the system microprocessor. In addition, bits 3 - 0 of the this register provide change information. These four bits are set to logical 1 whenever a control input from the modem changes state. They are reset to logical 0 whenever the system microprocessor reads this register.

Bit	Function
7	Data-Carrier-Detect
6	Ring Indicator
5	Data-Set-Ready
4	Clear-to-Send
3	Delta-Data-Carrier-Detect
2	Trailing Edge Ring Indicator
1	Delta-Data-Set-Ready
0	Delta-Clear-to-Send

Figure 19. Modem Status Register

Bit 7 This bit is the inverted 'data-carrier-detect' (DCD) modem control input signal. If bit 4 of the Modem Control register is set to 1, this bit is equivalent to bit 3 in the Modem Control register.

Bit 6 This bit is the inverted 'ring-indicator' (RI) modem control input signal. If bit 4 of the Modem Control register is set to 1, this bit is equivalent to bit 2 in the Modem Control register.

Bit 5 This bit is the inverted 'data-set-ready' (DSR) modem control input signal. If bit 4 of the Modem Control register is set to 1, this bit is equivalent to bit 0 in the Modem Control register.

Bit 4 This bit is the inverted 'clear-to-send' (CTS) modem control input signal. If bit 4 of the Modem Control register is set to 1, this bit is equivalent to bit 1 in the Modem Control register.

Bit 3 When set to 1, this bit indicates that the 'data-carrier-detect' modem control input signal has changed state since the last time it was read by the system microprocessor.

Note: Whenever bit 0, 1, 2, or 3 is set to 1, a modem status interrupt is generated.

Bit 2 When set to 1, this bit indicates that the 'ring-indicator' modem control input signal has changed from an active condition to an inactive condition.

Bit 1 When set to 1, this bit indicates that the 'data-set-ready' modem control input signal has changed state since the last time it was read by the system microprocessor.

Bit 0 When set to 1, this bit indicates that the 'clear-to-send' modem control input signal has changed state since the last time it was read by the system microprocessor.

Scratch Pad Register

This register does not control the adapter in any way. It can be used by the system microprocessor to temporarily hold data.

Address and Interrupt Assignments

The controller register I/O addresses and interrupt levels are:

Register	Serial Mode 1	2	3	4	5	6	7	8
RBR (DLAB = 0)	03F8	02F8	3220	3228	4220	4228	5220	5228
THR (DLAB = 0)	03F8	02F8	3220	3228	4220	4228	5220	5228
IER (DLAB = 0)	03F9	02F9	3221	3229	4221	4229	5221	5229
IIR	03FA	02FA	3222	322A	4222	422A	5222	522A
LCR	03FB	02FB	3223	322B	4223	422B	5223	522B
MCR	03FC	02FC	3224	322C	4224	422C	5224	522C
LSR	03FD	02FD	3225	322D	4225	422D	5225	522D
MSR	03FE	02FE	3226	322E	4226	422E	5226	522E
SCR	03FF	02FF	3227	322F	4227	422F	5227	522F
DLL (DLAB = 1)	03F8	02F8	3220	3228	4220	4228	5220	5228
DLM (DLAB = 1)	03F9	02F9	3221	3229	4221	4229	5221	5229
Interrupt Level	4	3	3	3	3	3	3	3

Figure 20. I/O Address and Interrupt Level Assignments (Async)

Definitions of the abbreviations in Figure 20 are as follows:

DLAB: Divisor Latch Access Bit
DLL: Divisor Latch (LSB)
DLM: Divisor Latch (MSB)
IER: Interrupt Enable Register
IIR: Interrupt Identification Register (Read Only)
LCR: Line Control Register
LSR: Line Status Register (Read Only)
MCR: Modem Control Register
MSR: Modem Status Register
RBR: Receiver Buffer Register (Read Only)
SCR: Scratch Pad Register
THR: Transmitter Holding Register (Write Only)

Interrupt Assignments

Two interrupt lines are provided to the system. Interrupt level 4 (IRQ4) is for serial mode 1 and interrupt level 3 (IRQ3) is for serial modes 2 through 8. To allow the adapter to send interrupts to the system, bit 3 of the Modem Control register must be set to 1. At this point, any interrupts allowed by the Interrupt Enable register will cause an interrupt.

Hardware Interrupts

Hardware interrupts are level-sensitive for systems using the Micro Channel architecture while systems using the Personal Computer type I/O channel design have edge-sensitive hardware interrupts. On edge-sensitive interrupt systems, the interrupt controller clears its internal interrupt-in-progress latch when the interrupt routine sends an End-of-Interrupt (EOI) command to the controller. The EOI is sent whether the incoming interrupt request to the controller is active or inactive.

In level-sensitive systems, the interrupt-in-progress latch is readable at an I/O address bit position. This latch is read during the interrupt service routine and may be reset by the read operation or may require an explicit reset.

Note: Designers may wish to limit the number of devices sharing an interrupt level for performance and latency considerations.

The interrupt controller on level-sensitive systems requires the interrupt request to be inactive at the time the EOI is sent; otherwise, a "new" interrupt request will be detected and another microprocessor interrupt caused.

To avoid this problem, a level-sensitive interrupt handler must clear the interrupt condition (usually by a Read or Write to an I/O port on the device causing the interrupt). After clearing the interrupt condition, a JMP $+2 should be executed prior to sending the EOI to the interrupt controller. This ensures that the interrupt request is removed prior to re-enabling the interrupt controller. Another JMP $+2 should be executed after sending the EOI, but prior to enabling the interrupt through the Set Interrupt Enable Flag (STI) command.

Serial Data Format

The serial data format is as follows:

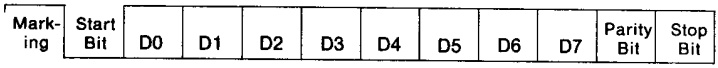

Figure 21. Serial Data Format

Data bit 0 is the first bit to be transmitted or received. The adapter automatically inserts the start bit, the correct parity bit, if programmed to do so, and the stop bits (1, 1-1/2, or 2 depending on the command in the Line Control register).

During the transmission of data, the marking condition will be used to denote the binary state 1, and the spacing condition is used to denote the binary state 0.

Binary Synchronous Communications

The Binary Synchronous Communications (BSC) portion of the adapter uses:

- An Intel 8251A Universal Synchronous/Asynchronous Receiver/Transmitter communications controller
- An Intel 8255A-5 Programmable Peripheral Interface (PPI) for expanded modem operation
- An Intel 8254 Programmable Interval Timer for time-outs and generating interrupts.

8251A Description

The 8251A communications controller accepts data characters from the system in parallel format and converts them into a continuous serial data stream for transmission. While accepting data characters it can also receive external serial data streams and convert them into parallel data characters for the system microprocessor. When the controller can accept a new character for transmission or when it has received a character for the system microprocessor it signals the system. The system microprocessor can read the complete status of the controller at any time.

Once the controller is programmed, the 'transmitter ready' (TxRDY) signal is raised to an active level to signal the system microprocessor that the controller is ready to receive a data character from the system microprocessor. TxRDY is reset automatically when a character is written to the controller. The controller also receives serial data from the modem or I/O device. Upon receiving an entire character, the 'receiver ready' (RxRDY) signal is raised to an active level to an active level to signal the system microprocessor that the controller has a character ready. RxRDY is reset automatically by a data read operation. The controller cannot begin transmission until the transmitter enable bit is set in the Command instruction and it has received a 'clear-to-send' (CTS) modem control input signal. The 'transmit data' output signal is held in the marking state upon reset.

Programming the 8251A

Prior to starting data transmission or reception, the 8251A communications controller must be loaded with a set of control words generated by the system microprocessor. These control words define the complete functional definition of the controller and must immediately follow an internal or external reset operation.

Control words are split into two formats:

- Mode Instructions
- Command Instructions.

Mode and Command Instructions

The Mode and Command instructions define the general operational characteristics of the controller. Once the Mode instruction has been written into the controller by the system microprocessor, sync characters (usually hex 32 in BSC) or Command instructions may be inserted.

The following represents a typical data block and shows the primary SDLC Mode instruction and Command instruction:

Primary Hex Address	Data Block Information
389	Mode Instruction
3A9	Sync Character 1 (Sync Mode Only)
3A9	Sync Character 2 (Sync Mode Only)
3A9	Command Instruction
3A8	Data
3A9	Command Instruction
3A8	Data
3A9	Command Instruction

Figure 22. Typical Data Block

Note: The second Sync Character is skipped if the Mode instruction has programmed the controller to single-character internal sync mode. Both Sync Characters are skipped if the Mode instruction has programmed the controller to the asynchronous mode.

Mode Instruction

The controller can be used for either asynchronous or binary synchronous data communication. The format can be changed only after a master chip reset.

Note: When parity is enabled, it is not considered as one of the data bits for the purpose of programming the word length. The actual parity bit received on the 'receive data' line cannot be read on the data bus. In the case of a programmed character length of less than 8 bits, the least significant data bus bits hold the data; unused bits are "don't care bits" when writing data to the controller, and are 0's when reading the data from the controller.

Asynchronous Mode

Transmit

When a data character is sent by the system microprocessor, the controller automatically adds a start bit (low level) followed by the data bits (least-significant bit first), and the programmed number of stop bits to each character. Also, an even or odd parity bit is inserted prior to the stop bits, as defined by the Mode instruction. The character is then transmitted as a serial data stream on the 'transmit data' line. The serial data is shifted out on the falling edge of the 'transmit character' (TxC) signal at a rate equal to 1/16 of TxC as defined by the Mode instruction. Break characters can be continuously sent to 'transmit data' if commanded to do so.

When no data characters have been loaded into the controller, the 'transmit data' output signal remains high (marking) unless a Break (continuously low) has been programmed.

Receive

The 'receive data' line is normally high. A falling edge on this line triggers the beginning of a start bit. The validity of this start bit is checked by strobing this bit a second time at its nominal center (16X or 64X mode only). If a low is detected again, it is a valid start bit, and the bit counter starts counting. The bit counter locates the center of the data bits, the parity bit and the stop bits. If a parity error occurs, the parity error flag is set.

Data and parity bits are sampled on the 'receive data' line with the rising edge of the 'receive character' (RxC) signal. If a low level is detected as the stop bit, the framing error flag is set. The stop bit signals the end of a character. The receiver requires only one stop bit, regardless of the number of stop bits programmed. This character is then loaded into the parallel I/O buffer of the controller. RxRDY is set high to signal the system that a character is ready. If a previous character has not been received by the system microprocessor, the present character replaces it in the I/O buffer, and the overrun error flag is raised and the previous character is then lost. All of the error flags can be reset by an Error Reset instruction. The occurrence of any of these errors does not affect the operation of the controller.

The Asynchronous Mode Instruction Format is shown in the following figure:

Bit	Name	Function
7 - 6	S2, S1	Stop Bits
5	EP	Even Parity Generation/Check
4	PEN	Parity enable
3 - 2	L2, L1	Character Rate Factor
1 - 0	B2, B1	Baud Rate Factor

Figure 23. Asynchronous Mode Instruction Format

Bits 7 - 6 Bits 7 (S2) and 6 (S1) select the number of stop bits used for transmit as shown in the following figure:

	Invalid	1 Bit	1.5 Bits	2 Bits
S1	0	1	0	1
S2	0	0	1	1

Receive operations never require more than one stop bit.

Figure 24. Number of Stop Bits

Bit 5 When this bit is set to 1, even parity is generated or checked. When this bit is cleared to 0, odd parity is generated or checked.

Bit 4 When this bit is set to 1, parity is enabled. When this bit is cleared to 0, parity is disabled.

Bits 3 - 2 Bits 3 (L2) and 2 (L1) select the character rate factor as shown in the following figure:

	5 Bits	6 Bits	7 Bits	8 Bits
L1	0	1	0	1
L2	0	0	1	1

Figure 25. Character Rate Factor

Bits 1 - 0 Bits 1 (B2) and 0 (B1) select the Baud Rate Factor as shown in the following figure:

	External Clock	(1X)	(16X)	(64X)
B1	0	1	0	1
B2	0	0	1	1

If character length is defined as 5, 6 or 7 bits, the unused bits are cleared to 0.

Figure 26. Baud Rate Factor

Synchronous Mode

Transmit

The 'transmit data' output signal is normally high until the system sends its first character to the 8251A controller. When CTS goes low, the first character is serially transmitted. All characters are shifted out on the falling edge of the 'transmit clock' signal. Data is shifted out at the transmit clock rate.

Once transmission has started, the data stream at the 'transmit data' output signal must continue at the transmit clock rate. If the system does not provide the controller with a data character before the controller transmitter buffers become empty, the sync characters are automatically inserted in the 'transmit data' data stream. In this case, the transmitter empty bit in the Status Read Format is set to 1 to signal that the controller is empty and sync characters are being sent out. The transmitter empty bit is not cleared to 0 when the sync is being shifted out. The transmitter empty bit is internally reset by a data character being written into the controller.

Receive

In this mode, character synchronization is achieved internally. If the sync mode has been programmed, the Enter Hunt command should be included in the first Command instruction word written. Data on the 'receive data' line is sampled on the rising edge of the 'receive character' (RxC) signal. The content of the receiver buffer is compared at every bit boundary with the first sync character until a match occurs. If the controller has been programmed for two sync characters, the next character received is also compared; when both sync characters have been detected, the controller ends the Hunt mode and is in Character Synchronization mode. The synchronization detect (SYNDET) bit in the Status Read Format is then set to 1, and is reset automatically by a Status Read. If parity is programmed, the synchronization detect bit is not set until the middle of the parity bit instead of the middle of the last data bit. An Enter Hunt command has no effect in the asynchronous mode of operation.

Parity errors and overrun errors are both checked the same way as in the asynchronous receive mode. Parity is checked when not in the Hunt mode.

The system microprocessor can command the receiver to enter the Hunt mode if synchronization is lost. This also sets all used character bits in the buffer to 1, preventing a possible false synchronization detect caused by data that happens to be in the receiver buffer at Enter Hunt time. When in the Sync mode, but not in Hunt, sync detection is still functional, but only occurs at the known word boundaries. If one status read indicates sync detect and a second status read also indicates sync detect, then the programmed sync detect characters have been received since the previous status read. (If double character sync has been programmed, then both

sync characters have been contiguously received to gate a synchronization detect indication.)

The BSC Mode Instruction Format is shown in the following figure:

Bit	Name	Function
7	SCS	Single Character Sync
6	ESD	External Sync Detect
5	EP	Even Parity Generation/Check
4	PEN	Parity Enable
3 - 2	L2, L1	Character Length
1 - 0	--	Reserved = 0

Figure 27. BSC Mode Instruction Format

Bit 7 When it is set to 1, this bit selects a single sync character. When it is cleared to 0, this bit selects a double sync character.

Bit 6 When it is set to 1, this bit sets the 'syndet' signal on the 8251 as an output. When it is cleared to 0, this bit sets the 'syndet' signal as an input. This bit is normally cleared to 0.

Bit 5 When this bit is set to 1, even parity is generated or checked. When this bit is cleared to 0, odd parity is generated or checked.

Bit 4 When this bit is set to 1, parity is enabled. When this bit is cleared to 0, parity is disabled.

Bits 3 - 2 Bits 3 (L2) and 2 (L1) select the character length as shown in the following figure:

	Character Length			
	5 Bits	6 Bits	7 Bits	8 Bits
L1	0	1	0	1
L2	0	0	1	1

Figure 28. Character Length

Bits 1 - 0 Reserved. These bits are always cleared to 0.

Command Instructions

Once the functional definition of the controller has been programmed by the Mode instruction and the sync characters are loaded (if in sync mode), the device is ready to be used for data communication. The Command instruction controls the actual operation of the selected format. Functions such as Enable Transmit/Receive, Error Reset and Modem Controls are provided by the Command instruction.

Once the Mode instruction has been written into the controller and if necessary, sync characters inserted, then all further "control writes" (3A9 primary, 389 alternate) will load a Command instruction. A reset operation (internal or external) returns the controller to the Mode Instruction Format.

The Command Instruction Format is shown in the following figure:

Bit	Name	Function
7	EH	Enter Hunt Mode*
6	IR	Internal Reset
5	RTS	Request-To-Send
4	ER	Error Reset
3	SBRK	Send-Break-Character
2	RxE	Receiver Enable
1	DTR	Data-Terminal-Ready
0	TxEN	Transmitter Enable

* An Error Reset must be performed when Receiver Enable and Enter Hunt mode are programmed.

Figure 29. Command Instruction Format

Bit 7 When this bit is set to 1, the adapter enters the Hunt Mode.

Bit 6 When this bit is set to 1, the adapter returns to the Mode Instruction Format.

Bit 5 When this bit is set to 1, RTS output is forced low.

Bit 4 When this bit is set to 1, the adapter resets all error flags.

Bit 3 When it is set to 1, this bit forces the 'transmit data' signal inactive. When this bit is cleared to 0, the adapter is set for normal operation.

Bit 2 When it is set to 1, this bit enables 'receiver ready' (RxRDY). When it is cleared to 0, this bit disables RxRDY.

Bit 1 When this bit is set to 1, DTR output is forced low.

Bit 0 When this bit is set to 1, transmit is enabled. When this bit is cleared to 0, transmit is disabled. This bit has no effect in the asynchronous mode.

Status Reads: The status of the device can be read at any time during the functional operation. Status updates are inhibited during status read.

A normal read command is issued by the system with an input from port (3A9 primary, 389 alternate).

Some of the bits in the Status Read Format have identical meanings to external output pins so that the controller can be used in a completely polled environment or in an interrupt-driven environment. TxRDY is an exception.

The Status Update can have a maximum delay of 28 clock periods (12 µsec) from the actual event affecting the status.

The Status Read Format is shown in the following figure:

Bit	Name	Function
7	DSR	Data-Set-Ready
6	SYNDET	Sync Detect
5	FE	Framing Error (Async Only)
4	OE	Overrun Error
3	PE	Parity Error
2	TxEMPTY	Transmitter Empty
1	RxRDY	Receiver Ready
0	TxRDY	Transmitter Ready

Figure 30. Status Read Format

Bit 7 When this bit is set to 1, 'data-set-ready' is at a low level.

Bit 6 This bit is set to 1 when a synchronous character is detected.

Bit 5 When this bit is set to 1, a framing error occurred. The framing error indicates that a valid stop bit was not detected at the end of every character. This bit is cleared to 0 by the Error bit of the command instruction.

A framing error does not inhibit the operation of the controller.

Bit 4 This bit is set to 1 when a character is not read before the next one becomes available. It is cleared to 0 by the Error bit of the command or instruction. An overrun error does not inhibit operation of the controller, but the previously overrun character is lost.

Bit 3 This bit is set to 1 when a parity error is detected. This bit is cleared to 0 by the Error bit of the Command instruction. A parity error does not inhibit operation of the controller.

Bit 2 When this bit is set to 1, the transmitter is empty.

Bit 1 When this bit is set to 1, the adapter is ready to receive data.

Bit 0 When this bit is set to 1, the adapter is ready to accept a character from the system.

8254 Description

The 8254 is a programmable interval timer and counter. The 8251A communications controller uses the 8254 for inactivity time-outs in the synchronous mode to interrupt the system after a preselected amount of time has elapsed from the start of a communication operation. Counter 0 is not used for synchronous operation. Counters 1 and 2 connect to interrupt-level 4 and, being programmed to terminal-count values, provide the desired time delay before generating a level-4 interrupt. These interrupts signal the system that a predetermined amount of time has elapsed without a TxRDY (level 4) or an RxRDY (level 3) interrupt being sent to the system.

Counter operation is determined by programming. Each counter must be initialized before it can be used. Unused counters do not need to be programmed.

After power-up, the state of the 8254 is undefined. The mode, count value, and output of all counters are undefined.

Counter Descriptions

Three counters are used: Counter 0, Counter 1, and Counter 2.

Counter 0: This counter is used to generate the baud rate for asynchronous communication mode. It is set to 16 times (16X) the baud rate so the controller can check the validity of a start bit when in receive mode. Counter 0 output is also connected to port-C, pin-5 of the 8255.

Counter 1: Counter 1 output is connected to port-A, pin-7 of the 8255 and interrupt level 4.

Counter 2: Counter 2 output is connected to port-A, pin-6 of the 8255 and interrupt level 4.

Programming the 8254

Counters are programmed by writing a control word and then an initial count.

All control words are written into the Control Word register (3A7 primary, 387 alternate). The control word itself specifies which counter is being programmed.

Initial counts are written into the counters, not the Control Word register. The format of the initial count is determined by the control word used. The 8254 control word format is shown in the following figure:

Bit	Name	Function
7	SC1	Select Counter 1
6	SC0	Select Counter 0
5	RW1	Read/Write 1
4	RW0	Read/Write 0
3	M2	Mode 2
2	M1	Mode 1
1	M0	Mode 0
0	BCD	Binary Coded Decimal

Figure 31. Control Word Register

Bits 7 - 6 These bits are used to select the counter as shown in the following figure:

SC1	SC0	
0	0	Select Counter 0
0	1	Select Counter 1
1	0	Select Counter 2
1	1	Read-Back Command (See Read Operations)

Figure 32. Select Counter

Bits 5 - 4 These bits select the Read and Write options as shown in the following figure:

RW1	RW0	
0	0	Counter Latch Command
1	0	Read/Write (Most-Significant Byte Only)
0	1	Read/Write (Least-significant Byte Only)
1	1	Read/Write (Least-Significant Byte first, then Most-Significant Byte)

Figure 33. Read/Write Options

Bits 3 - 1 These bits are used for counter mode selection, as shown in the following figure:

Bit 3	Bit 2	Bit 1	Mode Selected
0	0	0	Mode 0
0	0	1	Mode 1
0	1	0	Mode 2
0	1	1	Mode 3
1	0	0	Mode 4
1	0	1	Mode 5

Figure 34. Counter Mode Selection

Bit 0 When this bit is set to 1, the Binary Coded Decimal counter (4 Decades) is selected. When this bit is cleared to 0, the 16-bit Binary Counter is selected.

Write Operations

When programming write operations to the counter:

1. For each counter, the control word must be written before the initial count is written.
2. The initial count must follow the count format specified in the

control word (least-significant byte and then most-significant byte).

Since the Control Word register and the three counters have separate addresses and each control word specifies the counter it applies to (bits SC0, SC1), no special instruction sequence is required.

A new initial count may be written to a counter at any time without affecting the counter programmed mode. Counting is affected as described in the mode definitions. The new count must follow the programmed count format.

If a counter is programmed to read or write 2-byte counts, a program must not transfer control words between writing the first and second byte to another routine that also writes into that same counter. Otherwise, the counter is loaded with an incorrect count.
In the following programming examples, all counters are programmed to read or write counts. Four of the possible programming sequences are shown in the following figures:

Programming Sequence	A1	A0
Control Word - Counter 0	1	1
LSB of Count - Counter 0	0	0
MSB of Count - Counter 0	0	0
Control Word - Counter 1	1	1
LSB of Count - Counter 1	0	1
MSB of Count - Counter 1	0	1
Control Word - Counter 2	1	1
LSB of Count - Counter 2	1	0
MSB of Count - Counter 2	1	0

Figure 35. Sequence Example 1

Programming Sequence	A1	A0
Control Word - Counter 2	1	1
Control Word - Counter 1	1	1
Control Word - Counter 0	1	1
LSB of Count - Counter 2	1	0
MSB of Count - Counter 2	1	0
LSB of Count - Counter 1	0	1
MSB of Count - Counter 1	0	1
LSB of Count - Counter 0	0	0
MSB of Count - Counter 0	0	0

Figure 36. Sequence Example 2

Programming Sequence	A1	A0
Control Word - Counter 0	1	1
Control Word - Counter 1	1	1
Control Word - Counter 2	1	1
LSB of Count - Counter 2	1	0
LSB of Count - Counter 1	0	1
LSB of Count - Counter 0	0	0
MSB of Count - Counter 0	0	0
MSB of Count - Counter 1	0	1
MSB of Count - Counter 2	1	0

Figure 37. Sequence Example 3

Programming Sequence	A1	A0
Control Word - Counter 1	1	1
Control Word - Counter 0	1	1
LSB of Count - Counter 1	0	1
Control Word - Counter 2	1	1
LSB of Count - Counter 0	0	0
MSB of Count - Counter 1	0	1
LSB of Count - Counter 2	1	0
MSB of Count - Counter 0	0	0
MSB of Count - Counter 2	1	0

Figure 38. Sequence Example 4

Read Operations

There are three possible methods for reading the counters: Simple Read operations, Counter Latch commands, and Read-Back commands.

Simple Read Operation

To read the counter, which is selected with the A1, A0 signals on the channel, the clock input of the selected counter must be inhibited by using either the 'gate' input signal or external logic. Otherwise, the count may be in the process of changing when it is read, giving an undefined result.

Counter Latch Command

This command is written to the Control Word register as shown in the following figure.

Bit	Name	Function
7	SC1	Select Counter 1
6	SC0	Select Counter 0
5 - 4	--	Reserved = 0
3 - 0	--	Reserved = 0

Figure 39. Counter Latch Command Format

Bits 7 - 6 These bits are used to select one of the three counters as shown in the following figure:

SC1	SC0	COUNTER
0	0	0
0	1	1
1	0	2
1	1	Read-Back Command

Figure 40. Counter Selection

Bits 5 - 4 Reserved. These bits distinguish this command from a control word. These bits are always cleared to 0.

Bits 3 - 0 Reserved. These bits should always be cleared to 0.

The selected counter Output Latch latches the count at the time the Counter Latch command is resolved. This count is held in the latch until it is read by the system or until the counter is reprogrammed. The count is then unlatched automatically and the Output Latch returns to following the counting element. This allows reading the contents of the counters without affecting any counting currently in progress. Multiple Counter-Latch commands are used to latch more than one counter. Each latched counter Output Latch holds its count until it is read. Counter Latch commands do not affect the programmed mode of the counter.

If a counter is latched and then latched a second time before the count is read, the second Counter Latch command is ignored. The count read indicates the count available the time the first Counter Latch command was issued.

If the counter is programmed for two byte counts, two bytes must be read. The two bytes do not have to be read one right after the other. Read, write, or programming operations of other counters may be inserted between them.

Read and write operations of the same counter may be interleaved. For example, if the counter is programmed for two byte counts, the following sequence is valid:

1. Read least-significant byte
2. Write new least-significant byte
3. Read most-significant byte
4. Write new most-significant byte.

If a counter is programmed to read or write 2-byte counts, a program must not transfer control between reading the first and second byte to another routine that also reads from that same counter. Otherwise, an incorrect count is read.

Read-Back Command

This command allows the program to check the count value, programmed mode, and current states of the 'out' signal and Null Count Flag of the selected counters. This command is written into the Control Word register.

The read-back command is functionally equivalent to several Counter Latch commands, one for each counter latched. Each counter latched count is held until it is read (or the counter is reprogrammed). That counter is automatically unlatched when read, but other counters remain latched until they are read. If multiple count Read-Back commands are issued to the same counter without reading the count, all but the first are ignored. For example, the count that is read is the count at the time the first Read-Back command was issued.

Status must be latched to be read; status of a counter is accessed by a read from that counter. The Read Back Command format is shown in the following figure:

Bit	Name	Function
7 - 6	--	Reserved = 1
5	COUNT	0 = Latch Count Of Selected Counters
4	STATUS	0 = Latch Status Of Selected Counters
3	CNT 2	1 = Select Counter 2
2	CNT 1	1 = Select Counter 1
1	CNT 0	1 = Select Counter 0
0	--	Reserved = 0

Figure 41. Read-Back Command Format

Bits 7 - 6 Reserved. These bits must always be set to 1.

Bit 5 When this bit is cleared to 0, the counter output latches of the counters selected by bits 3 - 1 are latched.

Bit 4 When this bit is cleared to 0, the status information of the counters selected by bits 3 - 1 are latched.

Bit 3 When this bit is set to 1, Counter 2 is selected.

Bit 2 When this bit is set to 1, Counter 1 is selected.

Bit 1 When this bit is set to 1, Counter 0 is selected.

Bit 0 Reserved. This bit must always be cleared to 0.

Counter Status Command

Counter status reflects the counter programmed mode exactly as written in the last Mode Control Word. The Counter Status format is shown in the following figure:

Bit	Name	Function
7	OUTPUT	Monitor Out Signal
6	NULL COUNT	Null Count Operations
5	RW1	Counter Programmed Mode
4	RW0	Counter Programmed Mode
3	M2	Counter Programmed Mode
2	M1	Counter Programmed Mode
1	M0	Counter Programmed Mode
0	BCD	Counter Programmed Mode

Figure 42. Counter Status Format

Bit 7 When this bit is set to 1, the 'out' signal is high. When this bit is cleared to 0, the 'out' signal is low.

Bit 6 When this bit is set to 1, a Null Count is indicated. When this bit is cleared to 0, the count is available to be read.

This bit indicates when the last count written to the Counter register has been loaded into the counting element. The exact time this happens depends on the mode of the counter and is described in the mode definitions on page 39, but until the count is loaded into the counting element, it cannot be read from the counter. If the count is latched or read before this time, the count value will not reflect the new count just written.

The operation of null count is shown in the following figure:

This Action:	Causes:
A. Write to the Control Word Register *	Null Count = 1
B. Write to the Count Register (CR) **	Null Count = 1
C. New Count is Loaded into Counting Element (CR→CE)	Null Count = 0

* Only the counter specified by the control word will have its null count set to 1; null count bits of other counters are unaffected.

** If the counter is programmed for 2-byte counts (least-significant byte, then most-significant byte), the null count goes to 1 when the second byte is written.

Figure 43. Null Count Operation

Bits 5 - 0 These bits select the counter mode as shown in the figures on page 32.

If multiple status latch operations of the counters are performed without reading the status, all but the first are ignored. For example, the status that is read is the status of the counter at the time the first status Read-Back command was issued.

Both count and status of the selected counters may be latched simultaneously by clearing bits 5 and 4 to 0. This is functionally the

same as issuing two separate Read-Back commands at once. If multiple counts or status Read-Back commands are issued to the same counters without any intervening read operations, all but the first are ignored.
This is illustrated in the following figure:

Command 7 6 5 4 3 2 1 0	Description	Result
1 1 0 0 0 0 1 0	Read-back count and status of counter 0.	Count and status latched for counter 0.
1 1 1 0 0 1 0 0	Read-back status of counter 1.	Status latched for counter 1.
1 1 1 0 1 1 0 0	Read-back status of counters 2, 1.	Status latched for counter 2, but not counter 1.
1 1 0 1 1 0 0 0	Read-back count of counter 2.	Count latched for counter 2.
1 1 0 0 0 1 0 0	Read-back count and status of counter 1.	Count latched for counter 1, but not status.
1 1 1 0 0 0 1 0	Read-back status of counter 1.	Command ignored, status already latched for counter 1.

Figure 44. Read-Back Command Example

If both count and status of a counter are latched, the first read operation of that counter returns latched status, regardless of which was latched first. The next one or two read operations return the latched count (depending on whether the counter is programmed for one or two type counts). Subsequent reads return unlatched count.

Mode Definitions

The following are defined for use in describing the operation of the 8254:

Clock (CLK) pulse: A rising edge, then a falling edge, in that order, of a counter clock input.

Trigger: A rising edge of a counter 'gate' input signal.

Counter Loading: The transfer of a count from the counter register to the counting element.

Mode 0 - Interrupt on Terminal Count

Mode 0 is typically used for event counting. After the control word is written, the out bit is initially low, and remains low until the counter reaches 0. The out bit then goes high and remains high until a new count or a new mode 0 control word is written into the counter.

When the gate bit is set to 1, counting is enabled; when the gate bit is cleared to 0, counting is disabled. The gate bit has no effect on the out bit.

After the control word and initial count are written to a counter, the initial count is loaded on the next clock pulse. This clock pulse does not decrement the count; for an initial count of N, OUT does not go high until N + 1 clock pulses after the initial count is written.

If a new count is written to the Counter, it is loaded on the next clock pulse and counting continues from the new count. If a 2-byte count is written:

1. Writing the first byte disables counting. The out bit is set low immediately (no clock pulse is required).
2. Writing the second byte allows the new count to be loaded on the next clock pulse.

These steps allow the counting sequence to be synchronized by programming. The out bit does not go high until N + 1 clock pulses after the new count of N is written.

If an initial count is written while the gate bit is cleared to 0, it is loaded on the next clock pulse. When the gate bit is set to 1, the out bit is set to 1 N clock pulses later. No clock pulse is needed to load the counter as this has already been done.

Mode 1 - Hardware Retriggerable One-Shot

OUT is initially high. OUT goes low on the clock pulse following a trigger to begin the one-shot pulse, and remains low until the counter reaches 0. OUT then goes high and remains high until the clock pulse after the next trigger.

After writing the control word and initial count, the counter is armed. A trigger results in loading the counter and setting OUT low on the

next clock pulse, starting the one-shot pulse. An initial count of N results in a one-shot pulse that is N clock cycles in duration. The one-shot is retriggerable; OUT remains low for N clock pulses after any trigger. The one-shot pulse can be repeated without rewriting the same count into the counter. GATE has no effect on OUT.

If a new count is written to the counter during a one-shot pulse, the current one-shot is not affected unless the counter is retriggered. In that case, the counter is loaded with the new count and the one-shot pulse continues until the new count expires.

Mode 2 - Rate Generator

This mode is typically used to generate a real time clock interrupt. It functions like a divide-by-N counter. OUT is initially high. When the initial count has decremented to 1, OUT goes low for one clock pulse. OUT then goes high again, the counter reloads the initial count, and the process is repeated. Mode 2 is periodic; the same sequence is repeated indefinitely. For an initial count of N, the sequence repeats every N clock cycles.

When the gate bit is set to 1, counting is enabled; when the gate bit is cleared to 0, counting is disabled. If GATE goes low during an output pulse, OUT is set high immediately. A trigger reloads the counter with the initial count on the next clock pulse; OUT goes low N clock pulses after the trigger. The GATE input can be used to synchronize the counter.

After writing a control word and initial count, the counter is loaded on the next clock pulse. OUT goes low N CLK pulses after the initial count is written. This allows the counter to be synchronized by programming.

Writing a new count during counting does not affect the current counting sequence. If a trigger is received after writing a new count but before the end of the current sequence, the counter is loaded with the new count on the next clock pulse and counting continues from the new count. Otherwise, the new count is loaded at the end of the current counting sequence. In mode 2, a count of 1 is illegal.

Mode 3 - Square Wave Mode

Mode 3 is typically used for baud rate generation. Mode 3 is similar to mode 2 except for the duty cycle of OUT. OUT initially is high. When half the initial count has expired, OUT goes low for the remainder of the count. Mode 3 is periodic; the sequence above is repeated indefinitely. An initial count of N results in a square wave with a period of N clock cycles.

When the gate bit is set to 1, counting is enabled; when the gate bit is cleared to 0, counting is disabled. If GATE goes low while OUT is low, OUT is set high immediately; no clock pulse is required. A trigger reloads the counter with the initial count on the next clock pulse. The 'gate' input signal can be used to synchronize the counter.

After writing a control word and initial count, the counter is loaded on the next clock pulse. This allows the counter to be synchronized by programming.

Writing a new count while counting does not affect the current counting sequence. If a trigger is received after writing a new count but before the end of the current half-cycle of the square wave, the counter is loaded with the new count on the next clock pulse and counting continues from the new count. Otherwise, the new count loaded at the end of the current half-cycle.

Even Counts: OUT is initially high. The initial count is loaded on a clock pulse and then is decremented by two on each succeeding clock pulse. When the count expires, OUT changes value and the counter is reloaded with the initial count. This process is repeated indefinitely.

Odd Counts: OUT is initially high. The initial count minus one (an even number) is loaded on a clock pulse and then is decremented by two on each succeeding clock pulse. One clock pulse after the count expires, OUT goes low and the counter is reloaded with the initial count minus one. Succeeding clock pulses decrement the count by two. When the count expires, OUT goes high again and the counter is reloaded with the initial count minus one. This process is repeated indefinitely. So for odd count, OUT is high for $(N+1)/2$ counts and low for $(N-1)/2$ counts.

Mode 4 - Programmable Triggered Strobe

OUT is initially high. When the initial count expires, OUT goes low for one clock pulse and then goes high again. The counting sequence is triggered by writing the initial count.

When the gate bit is set to 1, counting is enabled; when the gate bit is cleared to 0, counting is disabled. GATE has no effect on OUT.

After writing a control word and initial count, the counter is loaded on the next clock pulse. This clock pulse does not decrement the count, so for an initial count of N, OUT does not strobe low until N + 1 clock pulses after the initial count is written.

If a new count is written during counting, it is loaded on the next clock pulse and counting continues from the new count. If a 2-byte count is written:

1. Writing the first byte has no effect on counting.
2. Writing the second byte allows the new count to be loaded on the next clock pulse.

This allows the sequence to be retriggered by programming. OUT strobes low N + 1 clock pulses after the new count of N is written.

Mode 5 - Hardware Triggered Strobe (Retriggerable)

OUT is initially high. Counting is triggered by a rising edge of the 'gate' signal. When the initial count expires, OUT goes low for one clock pulse and then goes high again.

After writing the control word and initial count, the counter is not loaded until the clock pulse after a trigger. This clock pulse does not decrement the count, so for an initial count of N, OUT does not strobe low until N + 1 clock pulses after a trigger.

A trigger results in the counter being loaded with the initial count on the next clock pulse. The counting sequence is retriggerable. OUT does not strobe low for N + 1 clock pulses after any trigger. GATE has no effect on OUT.

If a new count is written during counting, the current counting sequence is not affected. If a trigger occurs after the new count is

written but before the current count expires, the counter is loaded with the new count on the next clock pulse and counting continues from there.

A summary of 'gate' signal operations is shown in the following figure:

	Gate Signal Status		
Modes	**Low or Going Low**	**Rising**	**High**
0	Disables Counting	--	Enables Counting
1	--	1) Initiates Counting 2) Resets Output After Next Clock	--
2	1) Disables Counting 2) Immediately Sets Output High	Initiates Counting	Enables Counting
3	1) Disables Counting 2) Immediately Sets Output High	Initiates Counting	Enables Counting
4	Disables Counting	--	Enables Counting
5	--	Initiates Counting	--

Figure 45. GATE Operations Summary

The minimum and maximum initial counts are shown in the following figure:

Mode	Minimum Count	Maximum Count
0	1	0
1	1	0
2	2	0
3	2	0
4	1	0
5	1	0

Figure 46. Minimum and Maximum Initial Counts

Note: 0 is equivalent to 2^{16} for binary counting and 10^4 for binary coded decimal counting.

Operations Common to All BSC Modes

When a control word is written to a counter, all control logic is immediately reset and OUT goes to a known initial state; no clock pulses are required.

The following are descriptions of operations common to all BSC modes.

Gate

The 'gate' input signal is always sampled on the rising edge of CLK. In Modes 0, 2, 3, and 4, the 'gate' input signal is level sensitive, and the logic level is sampled on the rising edge of CLK. In Modes 1, 2, 3, and 5, the 'gate' input signal is rising-edge sensitive. In these modes, a rising edge of GATE (trigger) sets an edge-sensitive flip-flop in the counter. This flip-flop is then sampled on the next rising edge of CLK; the flip-flop is reset immediately after it is sampled. A trigger is detected no matter when it occurs; a high logic level does not have to be maintained until the next rising edge of clock. In Modes 2 and 3, the 'gate' input signal is both edge- and level-sensitive. In Modes 2 and 3, if a clock source other than the system clock is used, GATE should be pulsed immediately following write of a new count value.

Counter

New counts are loaded and counters are decremented on the falling edge of CLK.

The largest possible initial count is 0; this is equivalent to 2^{16} for binary counting and 10^4 for binary coded decimal counting.

The counter does not stop when it reaches 0. In Modes 0, 1, 4, and 5 the counter "wraps around" to the highest count, either hex FFFF for binary counting or 9999 for binary coded decimal counting, and continues counting. Modes 2 and 3 are periodic; the counter reloads itself with the initial count and continues counting from there.

Typical Programming Sequence

Before starting data transmission or reception, the system programs the BSC portion of the adapter to define control and gating ports, counter functions, and the communications environment.

The 8255A-5 Programmable Peripheral Interface (PPI) is set for the proper mode by selecting the mode select address of the PPI (3A3 primary, 383 alternate) and writing the control word, hex 98. This defines port A as an input, port B as an output for modem control and gating, and port C for 4-bit input and 4-bit output. An output to port C sets the adapter to the wrap mode, disallows interrupts, and gates external clocks (address = PPI port C select, data = hex 0D). The adapter is now isolated from the communication interface, and setup continues.

Bit 4 of port B brings the 8251A 'reset' signal high, holds it, then drops it. This resets the internal registers of the 8251A.

Note: Provide a time delay of 850 ns (minimum) between read and/or write operations to the same 8255A-5 PPI.

Port assignments for the 8255 are shown in the following figures:

Port A Bit	Assigned Function (3A0 Primary, Alternate 380)
PA7	1 = Counter 1 Output Active
PA6	1 = Counter 2 Output Active
PA5	1 = Tx Ready Active
PA4	Oscillating = Receive Clock Active
PA3	0 = Clear-To-Send is on from interface
PA2	Oscillating = Transmit Clock Active
PA1	0 = Data Carrier Detect is on from interface
PA0	0 = Ring Indicator is on from interface

Figure 47. Port A Assignments

Port B Bit	Assigned Function (3A1 Primary, 381 Alternate)
PB7	1 = Gate Counters 1 and 2 to Interrupt level 4
PB6	1 = Gate Counter 1
PB5	1 = Gate Counter 2
PB4	1 = Reset 8251A
PB3	Not Used
PB2	0 = Turn on Test
PB1	0 = Turn on Select Standby
PB0	0 = Turn on Speed Select

Figure 48. Port B Assignments

Port C Bit	Assigned Function (3A2 Primary, 382 Alternate)
PC7	1 = SDLC Adapter 0 = BSC Adapter
PC6	0 = Test Indicate Active (Electronic Wrap Mode)
PC5	Oscillating = Counter 0 Output
PC4	Oscillating = Receive Data

Figure 49. Port C Assignments (Input)

Port Address	Assigned Function (3A2 Primary, Alternate 382)
PC3	0 = Enable Transmit Interrupt 4 and RVC Interrupt 3
PC2	1 = Electronic Wrap
PC1	1 = Gate External Clock
PC0	1 = Gate Internal Clock

Figure 50. Port C Assignments (Output)

Address and Interrupt Assignments

The following figure shows the I/O address assignments:

Hex Address Pri/Alt	Function
	- 8255 -
3A0/380	Port A Select
3A1/381	Port B Select
3A2/382	Port C Select
3A3/383	Mode Select
	- 8254 -
3A4/384	Counter 0 Select
3A5/385	Counter 1 Select
3A6/386	Counter 2 Select
3A7/387	Control Word
	- 8251A -
3A8/388	Data Select
3A9/389	Control Select

Figure 51. I/O Address Assignemnts (BSC)

Interrupt Information

The BSC portion of the adapter contains twelve I/O address spaces. Two interrupt levels are assigned for 'receiver ready' and 'transmitter ready' signals of the 8251A. The output of counter 1 and counter 2 of the 8254 share interrupt level 4 with the 'transmitter ready' signal.

Interrupt Level 3: Receiver Ready

Interrupt Level 4: Transmitter Ready
Counter 1
Counter 2

SDLC/HDLC Communications

The Synchronous Data Link Control (SDLC) and the High-level Data Link Control (HDLC) portion of the adapter uses an Intel 8273 Data Communications Protocol Controller.

8273 Description

Each 8273 controller operation, whether it is a frame transmission, reception or port read, consists of three phases: the command, execution, and result phases. A discussion of the transmit operation follows.

When transmitting a frame, the system issues a Transmit Frame command to the controller. It is not sufficient to just instruct the controller to transmit. The frame level command structure sometimes requires more information such as frame length and address and control field content. Once this additional information is supplied, the command phase is complete and the execution phase is entered.

During the execution phase the frame transmission takes place. The controller transmits the opening flag, A and C fields, the specified number of information field bytes, inserts the Frame Check Sequence (FCS), and the closing flag. Once the closing flag is transmitted, the controller leaves the execution phase and begins the result phase.

The SDLC frame format is shown in the following figure:

Opening Flag	Address Field(A)	Control Field(C)	Information Field(I)	Frame Check Sequence (FCS)	Closing Flag
01111110	8-Bits	8-Bits	Any Length 0 to N Bits	16-Bits	01111110

Figure 52. SDLC Frame Format

During the result phase, the controller notifies the system of the outcome of the command by supplying interrupt results. In this case, either the frame is complete or some error condition causes the transmission to be aborted. Once the system reads all of the results, the operation is complete (there is only one result phase for the Transmit Frame command).

System Interface

The system interface consists of four major blocks: Control/Read/Write logic (C/R/W), internal registers, data transfer logic, and data bus buffers.

Control Read Write Logic (C/R/W): The system uses the C/R/W logic to issue commands to the controller. Once the controller receives a command and executes it, the controller returns the results (good/bad completion) of the command by way of the C/R/W logic. The C/R/W logic is supported by seven registers that are addressed through address lines A0 and A1 and the 'I/O read' (-RD), 'I/O write' (-WR), and -CS signals. The 'A0' and 'A1' lines are generally derived from the two low order bits of the system address while -RD and -WR are the normal I/O Read and Write signals. The following figure shows the address of each register using the C/R/W logic:

Address Inputs		Control Inputs	
A1	**A0**	**-CS/-RD**	**-CS/-WR**
0	0	Status	Command
0	1	Result	Parameter
1	0	TxI/R	Test Mode
1	1	RxI/R	—

Figure 53. 8273 Register Selection

Internal Registers: The following details the function of each register using the C/R/W logic:

- **Command Register** - Controller operations are initiated by writing the appropriate command byte into this register.
- **Parameter Register** - Many commands require more information than found in the command itself. This additional information is provided through the parameter register.
- **Immediate Result (Result) Register** - The completion information (result) for commands that execute immediately are provided in this register.
- **Transmit Interrupt Results (TxI/R) Register** - Results of transmit operations are passed to the system in this register.
- **Receiver Interrupt Result (RxI/R) Register** - Receive operation results are passed to the system through this register.

- **Status Register** - The general status of the controller is provided in this register. The status register supplies the handshaking necessary during various phases of controller operation.
- **Test Mode Register** - This register provides a program reset function for the controller.

The commands, parameters, and bit definitions of these registers are discussed in "Programming the 8273" on page 58. Notice that there are no specific transmit or receive data registers. This feature is explained in the data transfer logic discussion that follows.

The final elements of the C/R/W logic are the the transmit and receive interrupt lines (TxINT and RxINT). These lines notify the system that either the transmitter or the receiver requires service. For example, results are read from the appropriate interrupt result register or a data transfer is required. The interrupt request remains active until all the associated interrupt results have been read or the data transfer is performed. By using the interrupt lines to relieve the system of the task of polling the controller to check if service is needed, the state of each interrupt line is reflected by a bit in the status register. Non-interrupt driven operation is possible by periodically examining the contents of these bits.

Data Transfer Logic: The controller supports two independent interfaces through the data transfer logic: receive data and transmit data. These interfaces are programmable for either DMA or non-DMA data transfers. While the choice of the configuration is up to the program, it is based on the intended maximum data rate of the communications channel. Figure 54 on page 52 illustrates the transfer rate of data bytes that are acquired by the controller based on link data rate. Duplex data rates above 9600 baud usually require DMA. Slower speeds may or may not require DMA, depending on the task load and interrupt response time of the processor.

The following figure shows the controller in a typical DMA environment:

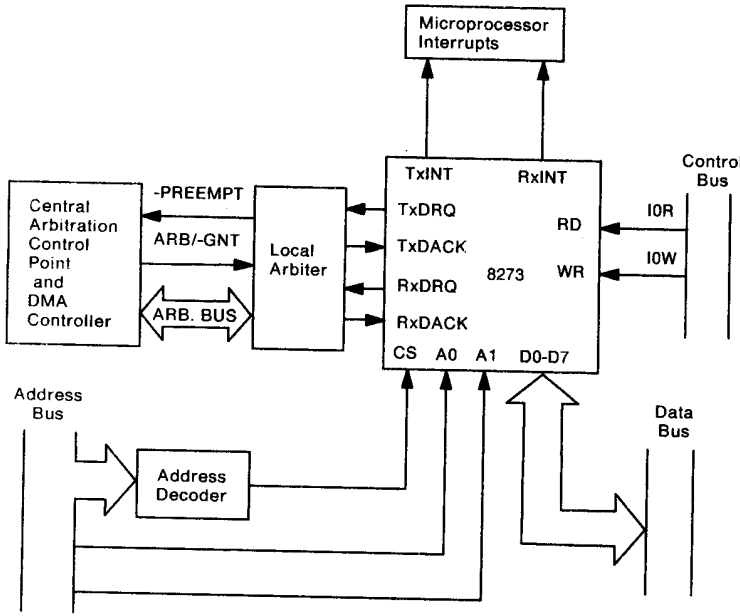

Figure 54. DMA, Interrupt-Driven System

Notice that a separate DMA controller is required. The DMA controller supplies the timing and addresses for the data transfers while the controller manages transfer requests and the actual data block length count. In this case, elements of the data transfer interface are:

TxDRQ: Transmit DMA Request - This line requests a DMA transfer from memory to the controller for transmit.

TxDACK: Transmit DMA Acknowledge - Returned by the Local Arbiter in response to TxDRQ, this line notifies the controller that a request has been granted, and provides access to the Transmitter Data register.

RxDRQ: Receiver DMA Request - This line requests a DMA transfer from the controller to memory for a receive operation.

RxDACK: Receiver DMA Acknowledge - Returned by the Local Arbiter, it notifies the controller that a receive DMA cycle has been granted, and provides access to the Receiver Data register.

RD: Read - This indicates that data is to be read from the controller and placed in memory.

WR: Write - This indicates that data is to be written to the controller from memory.

-PREEMPT If PC3 of the 8255-5 is cleared to 0, and TxDRQ or RxDRQ is asserted by the controller, the adapter activates -PREEMPT to cause an arbitration cycle to occur. The adapter drives the -PREEMPT low continuously until it wins the control of the channel.

ARB/-GNT The rising edge of this signal initiates an arbitration cycle. The adapter, assuming -PREEMPT is active, monitors arbitration and presents the arbitration priority levels to the channel if no other higher priority participant is simultaneously requesting bus service. The arbitration bits are withdrawn from the arbitration bus if another higher priority participant is requesting the channel. The falling edge of ARB/-GNT terminates this arbitration cycle. The Central Arbitration Control point activates and deactivates this signal.

DMA Transfers

To request a DMA transfer, the controller raises the appropriate DMA request line. Assume it is a transmitter request (TxDRQ): once the DMA Controller obtains control of the channel through its 'hold' and 'hold acknowledge' (HLDA) lines, the local arbiter notifies the controller that TxDRQ has been granted by returning TxDACK and -WR. The TxDACK and -WR signals transfer data to the controller for a transmit, independent of the controller chip select (CS) signal. A similar sequence occurs for receiver requests. This hard select of data into the transmitter or out of the receiver alleviates the need for the normal transmit and receive data registers addressed by a combination of address lines, -CS, and -WR or -RD. (The SDLC controller sees both the addresses and control signals supplied by the DMA controller during DMA cycles.) Typical frame transmit and

frame receive sequences show how the controller truly manages the DMA data transfer.

Frame Transmit

Before a frame can be transmitted, the system supplies the starting address for the desired information field to the DMA controller. See "Programming the 8273" on page 58 for more information. The controller is then commanded to transmit a frame. After the command, but before transmission begins, the controller needs more information (parameters). Four parameters are required for the transmit frame command: the address field byte, the control field byte, and two bytes that are the least-significant and most-significant bytes of the information field byte length. Once all four parameters are loaded, the controller makes 'request-to-send' (RTS) active and waits for 'clear-to-send' (CTS) to go active. Once CTS is active, the controller starts the frame transmission. While the controller is transmitting the opening flag, address field, and control field, it starts making transmitter DMA requests. These requests continue at character (byte) boundaries until the pre-loaded number of bytes of information field have been transmitted. At this point the requests stop, the FCS and closing flag are transmitted, and TxINT is raised, signaling the system that the frame transmission is complete. After the initial command and parameter loading, no system intervention is required (since DMA is used for data transfers) until the entire frame is transmitted.

Frame Receive

The receiver operation is similar to the transmit operation. Like the initial transmit sequence, the DMA controller is loaded with a starting address for a receiver data buffer and the controller is commanded to receive. Unlike the transmitter, there are two different receive commands: General Receive (refer to page 67), where all received frames are transferred to memory, and Selective Receive (refer to page 67), where only frames having an address field matching one of two pre-programmed controller address fields are transferred to memory.

After the Receive command, two parameters are required before the receiver becomes active: the least-significant and most-significant bytes of the receiver buffer length. Once these bytes are loaded, the receiver is active and the system may return to other tasks. The next

frame appearing at the receiver input is transferred to memory using receiver DMA requests. When the closing flag is received, the controller checks the frame check sequence and raises RxINT. The system can then read the results that indicate if the frame was error-free or not. If the received frame had been longer than the pre-loaded buffer length, the system would have been notified of that occurrence earlier with a receiver error interrupt. The command description section on page 66 contains a complete description of error conditions. Like the transmit example, after the initial command, the system is free for other tasks until a frame is completely received. These examples have illustrated the controller management of both the receiver and transmitter DMA channels.

Modem Interface

The modem interface of the controller consists of two major blocks: the Modem Control Block and the Serial Data Timing Block.

Modem Control Block: The modem control block provides both dedicated and user-defined modem control functions. Port A is a modem control input port; the following figure shows the bit definitions for Port A.

Bit	Name	Function
7 - 5	--	Reserved = 1
4	PA4	Data-Set-Ready Changed
3	PA3	Clear-To-Send Changed
2	PA2	Data-Set-Ready
1	CD	Carrier Detect
0	CTS	Clear-To-Send

Figure 55. 8273 Port A (Input)

Bits 7 - 5 Reserved. These bits are always set to 1.

Bit 4 This bit reflects the logical state of the 'data-set-ready' signal.

Bit 3 This bit reflects the logical state of the 'clear-to-send' signal.

Bit 2 This bit reflects the logical state of the 'data-set-ready' signal.

Bit 1 This bit reflects the logical state of the 'carrier detect' (CD) signal. CD is used to condition the start of a frame reception. CD must be active in time for a frame address field. If CD is lost (goes inactive) while receiving a frame, an interrupt is generated with a CD failure result.

Note: CD is tied low (active).

Bit 0 This bit reflects the logical state of the 'clear-to-send' signal. This signal is used to condition the start of a transmission. The controller waits until CTS is active before it starts transmitting a frame. While transmitting, if CTS goes inactive, the frame is aborted and the system is interrupted. When the system reads the interrupt results, a CTS failure is indicated.

Port B is a modem control output port; the following figure shows the bit definitions for Port B.

Bit	Name	Function
7 - 6	--	Reserved = 1
5	PB5	Flag Detect
4	PB4	Reserved = 1
3	PB3	Reserved = 1
2	PB2	Data-Terminal-Ready
1	--	Reserved = 1
0	RTS	Request-To-Send

Figure 56. 8273 Port B (Output)

As in Port A, the bit values represent the logical condition of the signals.

Bit 7 - 6 Reserved. These bits are always set to 1.

Bit 5 This bit reflects the state of the 'flag detect' signal. This signal goes active when an active receiver sees a flag character.

Bit 4 Reserved. This bit is always set to 1.

Bit 3 - 1 These bits are used defined outputs. PB1, PB3, and PB4 reflect the logical state of these bits.

Bit 0 This bit reflects the logical state of the 'request-to-send' signal. RTS is normally used to notify the modem that the controller wishes to transmit. This function is handled automatically by the controller. If RTS is inactive (high) when the controller is commanded to transmit, the controller makes it active and then waits for CTS before transmitting the frame. One byte time after the end of the frame, the controller returns RTS to its inactive state. If RTS was active when a Transmit command is issued, the controller leaves it active when the frame is complete.

Serial Timing Data Block: Elements of the serial data logic section are the data signals, TxD (transmit data output) and RxD (receive data input), and the respective transmit and receive data clocks. The transmit and receive data is synchronized by the transmit and receive character clocks. The following figure shows the timing for these signals:

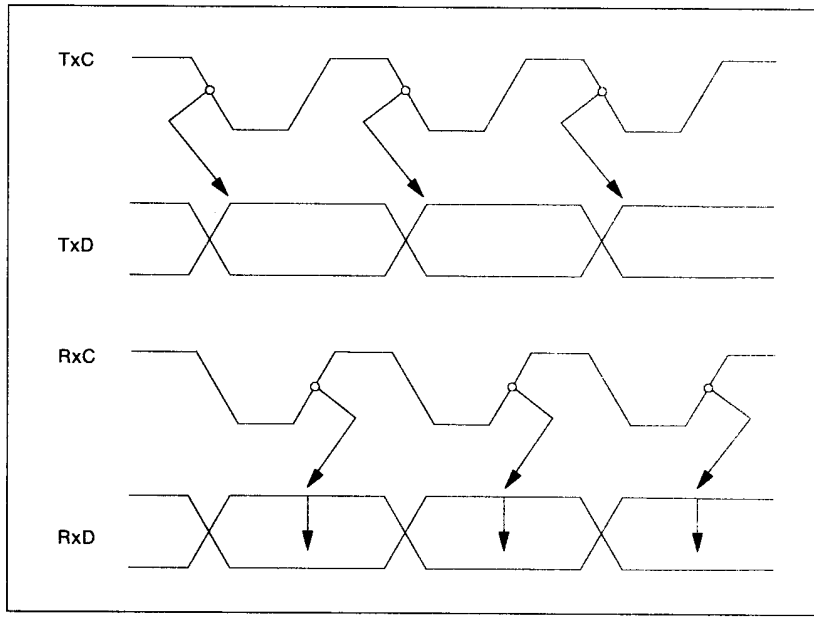

Figure 57. Transmit/Receive Timing

The leading edge (negative transition) of TxC generates new transmit data and the trailing edge (positive transition) of RxC is used to capture the receive data.

Programming the 8273

Programming the 8273 controller involves the communication of both commands from the system to the controller and the return of results of those commands from the controller to the system. This section describes the registers and commands used by the 8273 controller.

8273 Status Register

Handshaking protocol between the controller and the system is handled by the Status register. The bit definition of this register is shown in the following figure:

Bit	Name	Function
7	CBSY	Command Busy
6	CBF	Command Buffer Full
5	CPBF	Command Parameter Buffer Full
4	CRBF	Command Result Buffer Full
3	RxINT	Receive Interrupt
2	TxINT	Transmit Interrupt
1	RxIRA	Receive Interrupt Result Available
0	TxIRA	Transmit Interrupt Result Available

Figure 58. Status Register Format

Bit 7 This bit indicates when the controller is in the command phase. This bit is set to 1 when the system writes a command into the Command register, starting the command phase. It is cleared to 0 when the last parameter is placed in the Parameter register and accepted by the controller, completing the command phase.

Bit 6 When it is set to 1, this bit indicates that a byte is present in the Command register. This bit is normally not used.

Bit 5 This bit indicates that the Parameter register contains a parameter. It is set to 1 when the system places a parameter in the Parameter register. It is cleared to 0 when the controller accepts the parameter.

Bit 4 This bit is set to 1 when the controller places a result from an immediate type command in the Result register. It is cleared to 0 when the system reads the result from the Result register.

Bit 3 This bit reflects the state of the 'RxInt' signal. RxINT is set by the controller whenever the receiver needs servicing. RxINT is cleared to 0 when the system reads the results or performs the data transfer.

Bit 2 This bit is identical to RxINT except action is initiated based on transmitter interrupt sources.

Bit 1 This bit is set to 1 when the controller places an interrupt result byte into the Receiver Interrupt Result Available register register. This bit is cleared to 0 when the system reads the Receiver Interrupt Result Available register.

Bit 0 This bit is set to 1 when the controller places an interrupt result byte in the Transmit Interrupt Available register and cleared to 0 when the system reads the register.

Operating Mode Register

Bit	Function
7 - 6	Reserved -- Do Not Change
5	HDLC Abort Enable
4	EOP Interrupt Enable
3	Early Transmit Interrupt Enable
2	Buffered Mode
1	Pre-frame Sync Mode
0	Flat Stream Mode

Figure 59. Operating Mode Register

Bits 7 - 6 Reserved. These bits must be 0 for the Set command and 1 for the Reset command.

Bit 5 When this bit is set to 1, the controller interrupts when seven 1's (HDLC Abort) are received by an active receiver. When this bit is cleared to 0, eight 1's (SDLC Abort) causes an interrupt.

Bit 4 When this bit is set to 1, reception of an EOP character (0 followed by seven 1's) causes the controller to interrupt the system. When this bit is cleared to 0, no EOP interrupt is generated.

Bit 3 This bit specifies when the transmitter should generate an end-of-frame interrupt. When this bit is set to 1, an interrupt is generated when the last data character has been passed to the controller. If the program issues another transmit command within two byte times, the final flag interrupt does not occur and the new frame is transmitted with only one flag of separation. If this restriction is not met, more than one flag separates the frames and a frame-complete interrupt is generated after the closing flag. When set to 1, this bit allows a single flag to separate consecutive frames. When this bit is cleared to 0, only the frame-complete interrupt occurs.

Bit 2 When this bit is set to 1, the address and control fields of received frames are buffered in the controller and passed to the system as results after a received frame interrupt (the results are not transferred to memory with the information field). On transmit, the A and C fields are passed to the controller as parameters. This mode simplifies buffer management. When this bit is cleared to 0, the A and C fields are passed to and from memory as the first two data transfers.

Bit 1 When this bit is set to 1, the controller prefaces each transmitted frame with two characters before the opening flag. These two characters provide 16 transitions to allow synchronization of the opposing receiver. To guarantee 16 transitions, the two characters are hex-55, hex-55 for the non-NRZI (non-return to zero) mode or hex-00, hex-00 for NRZI mode. When this bit is cleared to 0, no pre-frame characters are transmitted.

Bit 0 When this bit is set to 1, the transmitter starts sending flag characters as soon as it is idle. When this bit is cleared to 0, the transmitter starts sending idle characters on the next character boundary if it is already idle, or at the end of a transmission if active.

Serial I/O Mode Register

Bit	Function
7 - 3	Reserved -- Do Not Change
2	Data Loopback
1	Clock Loopback
0	NRZI Mode

Figure 60. Serial I/O Mode Register

Bits 7 - 3 Reserved. These bits must be 0 for the Set command and 1 for the Reset command.

Bit 2 When this bit is set to 1, transmitted data (TxD) is internally routed to the receive data circuitry; when cleared to 0, TxD and RxD are independent.

Bit 1 When this bit is set to 1, TxC is internally routed to RxC; when cleared to 0, the clocks are independent.

Bit 0 When this bit is set to 1, the controller assumes the received data is NRZI encoded, and NRZI encodes the transmitted data. When this bit is cleared to 0, the received and transmitted data are treated as normal positive logic bit streams.

Data Transfer Mode Register

Bit	Function
7 - 1	Reserved -- Do Not Change
0	Interrupt Data Transfer

Figure 61. Data Transfer Mode Register

Bits 7 - 1 Reserved. These bits must be 0 for the Set command and 1 for the Reset command.

Bit 0 When this bit is set to 1, the controller interrupts the system when data transfers are required (the corresponding IRA Status register bit is 0 to signify a data transfer interrupt rather than a result phase interrupt). When this bit is cleared to 0, controller data transfers are performed through DMA requests on the DRQ line without interrupting the system.

1-Bit Delay Mode Register

Bit	Function
7	1-Bit Delay
6 - 0	Reserved

Figure 62. 1-Bit Delay Mode Register

Bit 7 When this bit is set to 1, the controller retransmits the received data stream 1-bit delayed. This mode is entered and exited at a received character boundary. When cleared to 0, the transmitted and received data are independent.

Bits 6 - 0 Reserved. These bits must be 0 for the Set command and 1 for the Reset command.

The mask that sets or clears the desired bits is treated as a single parameter. These commands do not interrupt or provide results during the result phase. After reset, these bits are cleared to 0.

Command Phase Programming

The system starts the command phase by writing a command byte into the controller Command register. If further information about the command is required by the controller, the system writes this information into the Parameter register.

The following figure is a flowchart of the command phase:

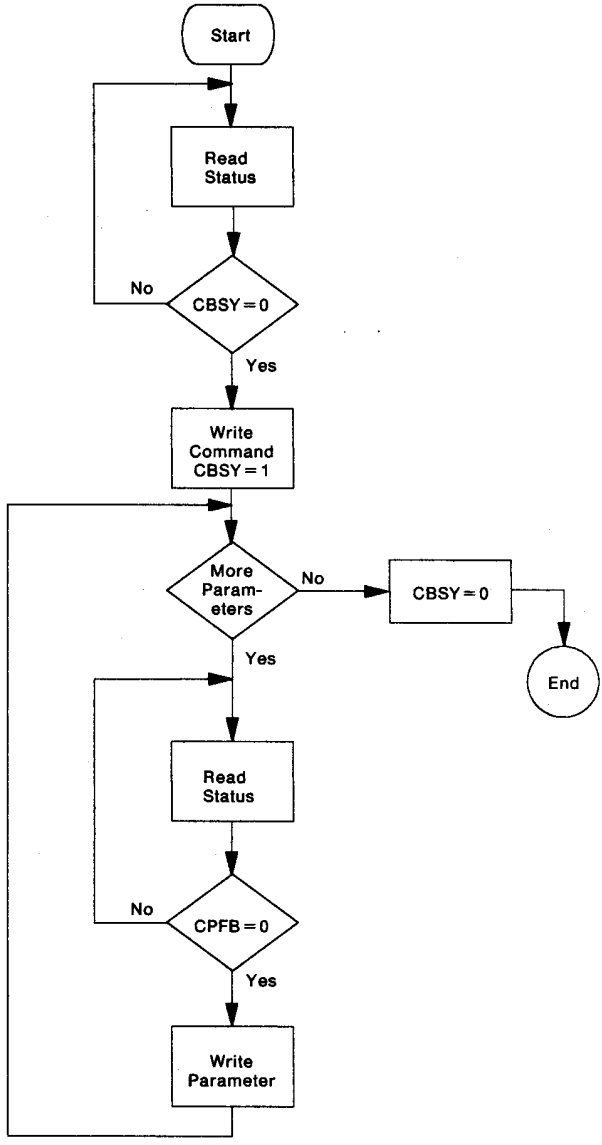

Figure 63. Command Phase Flowchart

Bits 7 and 5 of the Status register are used to handshake the command and parameter bytes. The Command Phase Flowchart shows that a command may not be issued if the Status register indicates the controller is busy, bit 7 set to 1. If a command is issued while bit 7 is set to 1, the original command is overwritten and lost. The flowchart also includes a parameter buffer full check. The system must wait until bit 5 of the Status register is cleared to 0 before writing a parameter to the Parameter register. If a parameter is issued while bit 5 is set to 1, the previous parameter is overwritten and lost. This assumes that a command buffer exists in memory. The Command Buffer Address register points to the buffer. Commands are entered into the buffer followed by the number of required parameters.

The controller transmitter and receiver may be executing commands or passing interrupt results at any given time. (Separate receive and transmit interrupt signals and result registers are provided for this reason.) However, there is only one Command register. The Command register must be used for only one command sequence at a time and the transmitter and receiver may never be simultaneously in a command phase. A detailed description of the commands and their parameters is presented in "8273 Command Descriptions" on page 66.

Execution Phase Programming

During the execution phase, the operation specified by the command phase is performed. If the system utilizes DMA for data transfers, there is no system involvement during this phase; no program support is required.

For interrupt-driven transfers, the controller raises the appropriate interrupt signal. When responding to the interrupt, the system must determine whether it is a data transfer request or an interrupt signaling that an operation is complete and results are available. The system determines the cause by reading the status register and interrogating the associated Interrupt Result Available (IRA) bit (TxIRA for TxINT and RxIRA for RxINT). If the IRA bit is cleared to 0, the interrupt is a data transfer request. If the IRA bit is set to 1, an operation is complete and the associated Interrupt Result register must be read to determine the completion status. An interrupt handler implementing the above sequence is presented as part of the result phase programs.

Result Phase Programming

During the result phase, the controller notifies the system of the outcome of a command. The result phase is initiated by either a successful completion of an operation or an error detected during execution. Some commands such as reading or writing the I/O ports provide immediate results, Other commands, such as frame transmit, take time to complete. Separate result registers are provided to distinguish these two types of commands and to avoid interrupt handling for simple results.

Immediate results are provided in the Result register. Validity of information in this register is indicated to the system through the bit 4 in the Status register. When the system completes the command phase of an immediate command, it polls the Status register, waiting until the bit 4 bit is set to 1. When this occurs, the system reads the Result register to obtain the immediate result. The Result register provides only the results from immediate commands.

All non-immediate commands deal with either the transmitter or receiver. Results from these commands are provided in the Transmit Interrupt Result and Receive Interrupt Result registers respectively. Results in these registers are conveyed to the system by the bits 0 and 1 of the Status register. Results of non-immediate commands consist of one byte result interrupt code indicating the condition for the interrupt and, if required, one or more bytes supplying additional information. The interrupt codes and the meaning of the additional results are described on page 68.

Non-immediate results are passed to the system in response to either interrupts or polling of the Status register.

8273 Command Descriptions

In this section, each command is discussed in detail. To shorten the notation, refer to the following table:

B0, B1	LSB and MSB of Receive Buffer Length
R0, R1	LSB and MSB of Received Frame Length
L0, L1	LSB and MSB of Transmit Frame Length
A1, A2	Match Addresses for Selective Receive
RIC	Receiver Interrupt Result Code
TIC	Transmitter Interrupt Result Code
A	Address Field of Received Frame
C	Control Field of Received Frame

Figure 64. Command Summary Key

The 8273 controller uses five different command types: initialization/configuration, receive, transmit, reset, and modem control.

Initialization/Configuration Commands

The initialization/configuration commands change internal controller registers that define the various operating modes. These commands either set or reset specified bits in the registers depending on the type of command. One parameter is required. Set commands perform a logical OR operation of the parameter (mask) and the internal register. This mask contains 1's where register bits are to be set. A 0 in the mask causes no change in the corresponding register bit. Reset commands perform a logical AND operation of the parameter (mask) and the internal register; for example, the mask is 0 to clear a register bit and a 1 to cause no change.

The following figure shows the Set and Reset commands associated with the register descriptions that follow:

Register	Command	Hex Code	Parameter
1-Bit Delay Mode	Set	A4	Set Mask
	Reset	64	Reset Mask
Data Transfer Mode	Set	97	Set Mask
	Reset	57	Reset Mask
Operating Mode	Set	91	Set Mask
	Reset	51	Reset Mask
Serial I/O Mode	Set	A0	Set Mask
	Reset	60	Reset Mask

Figure 65. Initialization/Configuration Command Summary

Receive Commands

Receive commands are discussed in the following section.

General Receive

When issued a General Receive command, the controller passes all frames to memory (DMA mode) regardless of the contents of the frame address field. This command is used for primary stations. Two parameters are required: B0 and B1. These parameters are the LSB and MSB of the receiver buffer size. Giving the controller these parameters alleviates the system of the burden of checking for buffer overflow. The controller interrupts the system if the received frame attempts to overfill the allotted buffer space.

Selective Receive

For Selective Receive commands, two additional parameters besides B0 and B1 are required: A1 and A2. These parameters are two address match bytes. When issued the Selective Receive command, the controller passes to memory or the system only those frames having an address field matching either A1 or A2. This command is usually used for secondary stations. A1 is the secondary address and A2 is the "All Parties" address. If only one match byte is needed, A1 and A2 should be equal. As for the General Receive command, the controller counts the incoming data bytes and interrupts the system if the receiver buffer size (B0, B1) is exceeded.

Receive Disable

The Receive Disable command terminates any receive operation immediately. No parameters are required and no results are returned.

The Receive command summary is shown in the following figure:

Command	Hex Code	Parameters	Results* RxI/R
General Receive	C0	B0,B1	RIC,R0,R1,A,C
Selective Receive	C1	B0,B1,A1,A2	RIC,R0,R1,A,C
Selective Loop Receive	C2	B0,B1,A1,A2	RIC,R0,R1,A,C
Disable Receiver	C5	None	None

* A and C are passed as results only in the buffered mode.

Figure 66. Receive Command Summary

Interrupt Result Codes

The interrupt result code key is shown in the following figure:

RIC D7-D0	Receiver Interrupt Result Code	Receive Status After Interrupt
* 00000	A1 Match or General Receive	Active
* 00001	A2 Match	Active
000 00011	CRC Error	Active
000 00100	Abort Detected	Active
000 00101	Idle Detected	Disabled
000 00110	EOP Detected	Disabled
000 00111	Frame < 32 Bits	Active
000 01000	DMA Overrun	Disabled
000 01001	Memory Buffer Overflow	Disabled
000 01010	Carrier Detect Failure	Disabled
000 01011	Receiver Interrupt Overrun	Disabled

* See following table (Receiver Interrupt Result Codes).

Figure 67. Interrupt Result Code

The receiver interrupt result codes are shown in the following figure:

D7-D5	Partial Byte Received
111	All 8 Bits of Last Byte
000	D0
100	D1-D0
010	D2-D0
110	D3-D0
001	D4-D0
101	D5-D0
011	D6-D0

Figure 68. Receiver Interrupt Result Codes (RIC)

The interrupt result code is the first byte passed to the system in the Receiver Interrupt Results register during the Result phase. Bits D4-D0 define the cause of the receiver interrupt. Since each result code has specific implications, they are discussed separately.

A1/A2 Match: The first two result codes result from the error-free reception of a frame. If the frame is received correctly after a General Receive command, the first result is returned. If either Selective Receive command was used, a match with A1 generates the first result code and a match with A2 generates the second. In either case, the receiver remains active after the interrupt; however, the internal buffer size counters are not reset. If the Receive command indicated 100 bytes were allocated to the receive buffer (B0,B1) and an 80-byte frame was received correctly, the maximum next frame size that could be received without resetting the receiver buffer (B0 and B1) is 20 bytes. It is common practice to reset the receiver buffer after each frame reception. DMA or memory pointers are usually updated at this time. (If the controller buffer management features are not desired, write hex 0FF to B0 and B1 for each Receive command. This allows frames of 65K bytes to be received without buffer overflow errors.)

CRC Error: This code indicates that a frame was received in the correct format; however, the received frame check sequence was not the same as the internally generated frame check sequence. The frame should be discarded. The receiver remains active. (Even though an error condition has been detected, all frame information up until that error has either been transferred to memory or passed to the system. This information should be invalidated and applies to all receiver error conditions.) Note that the FCS, either transmitted or received, is never available to the system.

Abort Detect: This code occurs whenever the receiver sees an SDLC (eight 1's) or an HDLC (seven 1's). However, the intervening abort character between a closing flag and an idle does not generate an interrupt. If an abort character (seen by an active receiver within a frame) is not preceded by a flag and is followed by an idle, an interrupt is generated for the abort, followed by an idle interrupt one character time interval later.

Idle Detect: This code occurs whenever 15 consecutive 1's are received. After an abort detect interrupt, the receiver remains active. After an idle detect interrupt, the receiver is disabled and must be reset before further frames may be received.

EOP Detected: If the EOP interrupt bit is set in the Operating Mode register, the EOP detect result is returned whenever an EOP character is received. The receiver is disabled, so the idle following the EOP does not generate an idle detect interrupt.

Frame Less Than 32 Bits: The minimum number of bits in a valid frame between the flags is 32. Fewer than 32 bits indicates an error. If buffered mode is selected, such frames are ignored (no data transfers or interrupts are generated). In non-buffered mode, a less-than-32-bit frame generates an interrupt with the less-than-32-bit frame result since data transfers may already have disturbed the interrupt handler. The receiver remains active.

DMA Overrun: This code occurs when the DMA controller is too slow extracting data from the controller. For example, the 'rxdack' signal is not returned before the next received byte is ready for transfer. The receiver is disabled if this error condition occurs.

Memory Buffer Overflow: This code occurs when the number of received bytes exceeds the receiver buffer length supplied by the B0 and B1 parameters in the receive command. The receiver is disabled.

Carrier Detect Failure: This code occurs when CD goes high (inactive) during reception of a frame. CD is used to qualify reception and must be active by the time the address field starts to be received. If CD is lost during the frame, a CD failure interrupt is generated and the receiver is disabled. No interrupt is generated if CD goes inactive between frames.

Receiver Interrupt Overrun: If a condition occurs that requires a second interrupt to be generated before the system has finished reading the previous interrupts results, the second interrupt is generated after the current result phase is complete (RxINT and the status bit go low, then high). The interrupt result for the second interrupt is a receive interrupt overrun. The actual cause of the second interrupt is lost. This may occur at the end of a received frame where the line goes idle. The controller generates a received frame interrupt after the closing flag and then 15 bit-times later, generates an idle detect interrupt. If the interrupt service routine is slow in reading the first interrupt results, the internal Receiver Interrupt Results register still contains result information when the idle detect interrupt occurs. Rather than deleting the previous results, the controller adds a receive interrupt overrun result as an extra result. If the system interrupt structure is such that the second interrupt is not acknowledged (interrupts are still disabled from the first interrupt), the receive interrupt overrun result is read as an extra result, after those from the first interrupt. If the second interrupt is serviced, the receive interrupt overrun is returned as a single result. In general, it is possible to have interrupts occurring at one character time intervals. The interrupt handling program must have at least that much response and service time.

The occurrence of receive interrupt overruns is an indication of marginal program design; the system interrupt response and servicing time is not sufficient for the data rates being attempted. It is advisable to configure the interrupt handling program to read the interrupt results, place them into a buffer, and clear the interrupt as quickly as possible. The program can then examine the buffer for new results, and take appropriate action. This can easily be accomplished by using a result buffer flag that indicates when new results are available. The interrupt handler sets the flag and the main program resets it once the results are retrieved.

SDLC allows frames that are of arbitrary length (greater than 32 bits). The controller handles this N-bit reception through the high-order bits (D7-D5) of the result code. These bits code the number of valid received bits in the last received information field byte. This indicates which of the high-order bits of the received partial byte are indeterminate. The address, control, and information fields are transmitted least-significant bit (A0) first. The FCS is complemented and transmitted most-significant bit first.

Transmit Commands

The Transmit are described in the following section.

Transmit Frame

The Transmit Frame command transmits a frame. Four parameters are required when buffered mode is selected and two are required when it is not. In either case, the first two parameters are the least- and the most-significant bytes of the desired frame length (L0 and L1 specify the number of data transfers to be performed). In buffered mode, the address and control fields are presented to the transmitter as the third and fourth parameters respectively. In non-buffered mode, the A and C fields must be passed as the first two data transfers.

When the Transmit Frame command is issued, the controller makes RTS active (low) if it was not already. It then waits until CTS goes active (low) before starting the frame. If the pre-frame sync bit in the Operating Mode register is set, the transmitter prefaces two characters (16 transitions) before the opening flag. If the flag stream bit is set in the Operating Mode register, the frame (including pre-frame sync if selected) is started on a flag boundary. Otherwise the frame starts on a character boundary.

At the end of the frame, the transmitter interrupts the system (the interrupt results are discussed on page 73) and returns to either idle or flag stream, depending on the flag stream bit of the Operating Mode register. If RTS was active before the transmit command, the controller does not change it. If it was inactive, the controller deactivates it within one character time interval.

Transmit Transparent

The Transmit Transparent command enables the controller to transmit a block of raw data. This data is without SDLC protocol; for example, no 0 bit insertion, flags, or FCS. It is possible to construct and transmit a bisynchronous message for front-end processor switching or to construct and transmit an SDLC message with incorrect FCS for diagnostic purposes. Only the L0 and L1 parameters are used since there are no fields in this mode. (The controller does not support a Receive Transparent command.)

Abort Commands

Each of the above transmit commands has an associated abort command. The Abort Frame Transmit command causes the transmitter to send eight contiguous 1's immediately (no 0 bit insertion) and then revert to either idle or flag streaming based on the flag stream bit (eight 1's is an abort character).

The Abort Transmit Transparent command causes the transmitter to revert to either idles or flags as a function of the flag stream mode specified. The abort commands require no parameters; however, they do generate an interrupt and return a result when complete.

A summary of the Transmit commands is shown in the following figure:

Command	Hex Code	Parameters *	Results Txl/R	
Transmit Frame Abort	C8	L0, L1, A, C	TIC	TIC
	CC	None		
Loop Transmit Abort	CA	L0, L1, A, C	TIC	TIC
	CE	None		
Transmit Transparent Abort	C9	L0, L1	TIC	TIC
	CD	None		

* A and C are passed as parameters in buffered mode only.

Figure 69. Transmit Command Summary

Transmit Interrupt Result Codes

The following figure shows the various transmitter interrupt result codes:

Bits 7-0	Transmit Interrupt Result Code	Transmit Status After Interrupt
000 01100	Early Transmit Interrupt	Active
000 01101	Frame Transmit Complete	Idle or Flags
000 01110	DMA Underrun	Abort
000 01111	Clear to Send Error	Abort
000 10000	Abort Complete	Idle or Flags

Figure 70. Transmit Interrupt Result (TIC) Codes

As in the receiver operation, the transmitter generates interrupts based on either good completion of an operation or an error condition to start the result phase.

Early Transmit Interrupt: This code occurs after the last data transfer to the controller if the early transmit interrupt bit is set in the Operating Mode register. If the controller is commanded to transmit again within two character time intervals, a single flag separates the frames. (Buffered mode must be used for a single flag to separate the frames. If non-buffered mode is selected, three flags separate the frames.) If this time constraint is not met, another interrupt is generated and multiple flags or idles separate the frames. The second interrupt is the normal frame transmit complete interrupt.

Frame Transmit Complete: This code occurs at the closing flag to signify a good completion.

DMA Underrun: This code is analogous to the DMA overrun result in the receiver. Since SDLC does not support intraframe time-fill, if the DMA controller or system does not supply the data in time, the frame must be aborted. The action taken by the transmitter on this error is automatic. It aborts the frame just as if an abort command had been issued.

Clear-to-Send Error: This code is generated if CTS goes inactive during a frame transmission, the frame is aborted. The abort complete result is self-explanatory. No abort complete interrupt is generated when an automatic abort occurs. The next command type consists of only one command.

Reset Command

The Reset command provides a programmable reset function for the controller. It is a special case and does not use the normal command interface. The reset facility is provided in the Test Mode register. The controller is reset by sending a hex 01 followed by a hex 00 to the Test Mode register. Writing the hex 01 followed by the hex 00 mimics the action required by the hardware reset. Since the controller requires time to process the reset internally, at least 10 cycles of CLK must occur between the writing of hex 01 and hex 00. The Reset command results in the following:

1. The modem control outputs are forced high (inactive).
2. The 8273 controller Status register is cleared.
3. Any commands in progress cease.
4. The 8273 controller enters an idle state until the next command is issued.

Modem Control Commands

The commands used to change the modem control ports are shown in the following figures:

Command	Hex Code	Parameter	Register Result
Read	22	None	Port Value

Figure 71. Port A (Input) Modem Control Command Summary

Command	Hex Code	Parameter	Register Result
Read	23	None	Port Value
Set	A3	Set Mask	None
Reset	63	Reset Mask	None

Figure 72. Port B (Output) Modem Control Command Summary

Read Port A and B: The Read Port A and Read Port B commands are immediate. The bit definition for the returned byte is shown beginning on page 55. The returned value represents the logical condition of the signal; for example, signal active (low) = bit set.

Set and Reset Port B: The Set and Reset Port B commands are similar to the initialization commands in that they use a mask parameter that defines the bits to be changed. Set Port B uses a logical OR mask and Reset Port B uses a logical AND mask. Setting a bit makes the signal active (low). Resetting the bit deactivates the signal (high).

8255 I/O Port Assignment

The 8255 contains three 8-bit ports (A, B, C). Port A is configured as an input port and port B as an output port. Port C is configured with PC0 to PC3 as outputs and PC4 to PC7 as inputs. Port A senses external modem signals and internal signals.

The 8255 port assignments are shown in the following figures:

Port A Bit	Assigned Function
PA7	1 = Timer 1 Output Active
PA6	1 = Timer 2 Output Active
PA5	1 = Modem Status Changed
PA4	Oscillating = Receive Clock Active
PA3	0 = Clear To Send is on from interface
PA2	Oscillating = Transmit Clock Active
PA1	0 = Data Carrier Detect is on from interface
PA0	0 = Ring Indicator is on from interface

Figure 73. Port A Assignments

Port B Bit	Assigned Function
PB7	1 = Enable Level 4 Interrupt
PB6	1 = Gate Timer 1
PB5	1 = Gate Timer 2
PB4	1 = Reset 8273
PB3	0 = Reset Modem Status Changed Logic
PB2	0 = Turn on Test
PB1	0 = Turn on Select Standby
PB0	0 = Turn on Speed Select

Figure 74. Port B Assignments

Port C Bit	Assigned Function
PC7	1 = SDLC Adapter 0 = BSC Adapter
PC6	0 = Test Indicate Active
PC5	Oscillating = Timer 0 Output
PC4	Oscillating = Receive Data

Figure 75. Port C Assignments (Input)

Port C Bit	Assigned Function
PC3	0 = Enable Interrupts 3, 4 and DMA Register
PC2	1 = Electronic Wrap
PC1	1 = Gate External Clock
PC0	1 = Gate Internal Clock

Figure 76. Port C Assignments (Output)

Address, DMA, and Interrupt Assignments

The following figure shows the I/O address assignments:

Hex Address Prl/Alt	Function
	- 8255 -
380/3A0	Port A Select
381/3A1	Port B Select
382/3A2	Port C Select
383/3A3	Mode Select
	- 8254 -
384/3A4	Counter 0 Select
385/3A5	Counter 1 Select
386/3A6	Counter 2 Select
387/3A7	Control Word
	- 8273 -
388/3A8	Out Command/In Status
389/3A9	Out Parameter/In Result Register
38A/3AA	Out Test Mode/In Transmit I/R
38B/3AB	In Receive I/R
38C/3AC	Out Data/In Data

Figure 77. I/O Address Assignments (SDLC/HDLC)

DMA Assignment

To select a proper DMA channel, the proper arbitration level should be programmed (see "Programmable Option Select (POS)" on page 3 for details).

In order to run the existent application, DMA channel 1 should be programmed for transmit and receive data transfers. The arbitration level 1 should be programmed for primary SDLC. The arbitration level 7 is recommended for ALT SDLC.

Interrupt Information

Interrupt Level 3: Transmit/Receive Interrupt

Interrupt Level 4: Counter 1 Interrupt
Counter 2 Interrupt
Clear-to-Send Changed
Data-Set-Ready Changed

RS-232-C Interface

One 25-pin, D-shell, male connector is provided to attach various peripheral devices. This connector provides an EIA RS-232C interface as described in the specifications of RS-232C and CCITT V-24.

The adapter converts signals to or from TTL levels and EIA voltage levels. These signals are sampled or generated by the communications control chip. These signals can then be sensed by the system program to determine the state of the interface or peripheral device.

Additional Adapter Signal

The adapter provides one additional signal for use with World Trade modems, the 'connect-data-set-to-line' (CDSTL) signal (connect to DCE).

This signal is not standardized by EIA RS-232C Specifications. CDSTL is normally used with World Trade modems (CCITT 108.1). This signal is used the same as the DTR uses the same pin (pin-20) as DTR. The On condition causes the data communications equipment (DCE) to connect the modem to the line. The Off condition causes the DCE to remove the modem from the line.

Modem Control Input Signals

The following are input signals from the modem or external device to the controller.

Clear-to-Send (CTS): When active, this signal indicates that the modem is ready for the adapter to transmit data.

Data-Set-Ready (DSR): When active, this signal indicates the modem or data set is ready to establish the communications link and transfer data with the controller.

Ring Indicator (RI): When active, this signal indicates the modem or data set detected a telephone ringing signal.

Data-Carrier-Detect (DCD): When active, this signal indicates that the modem or data set detected a data carrier.

Modem Control Output Signals

The following are controller output signals. They are all set inactive upon a master reset operation. These signals are controlled by bits 3 - 0 in the Modem Control register.

Data-Terminal-Ready (DTR): When active, this signal informs the modem or data set that the controller is ready to communicate.

Request-to-Send (RTS): When active, this signal informs the modem or data set that the controller is ready to send data.

Output 1 (OUT 1): This signal is pulled high.

Output 2 (OUT 2): User-designated output. This signal controls interrupts to the system.

Voltage Interchange Information

The signal is considered in the *marking* condition when the voltage on the interchange circuit, measured at the interface point, is more negative than -3 Vdc with respect to signal ground. The signal is considered in the *spacing* condition when the voltage is more positive than +3 Vdc with respect to signal ground. The region between +3 Vdc and -3 Vdc is defined as the transition region, and considered an invalid level. The voltage that is more negative than -15 Vdc or more positive than +15 Vdc will also be considered an invalid level.

Interchange Voltage	Binary State	Signal Condition	Interface Control Function
Positive Voltage	Binary 0	Spacing	On
Negative Voltage	Binary 1	Marking	Off

Figure 78. Voltage Levels

Connector

The following figures shows the connector pin locations and pin assignments:

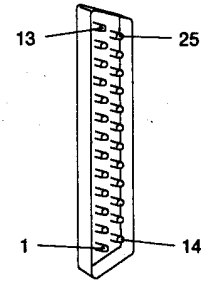

Figure 79. Pin Locations

Pin No.	Signal Flow	Lead ID RS232C/ CCITT24	Signal Name
7	NA	AB/102	Signal Ground
2	O	BA/103	Transmitted Data
3	I	BB/104	Received Data
4	O	CA/105	Request-To-Send
5	I	CB/106	Clear-To-Send
6	I	CC/107	Data-Set-Ready
20	O	CD/108.2	Data-Terminal-Ready
20	O	**/108.1	Connect-Data-Set-to-Line
8	I	CF/109	Data Carrier Detect
23*	O	CH/111	Rate Select
15*	I	DB/114	Transmit-Signal-Element-Timing
17*	I	DD/115	Receive Signal Element Timing
22	I	CE/125	Ring Indicator

* Not used in asynchronous communications mode.
** Not standardized by EIA (Electronics Industry Association).

Figure 80. EIA/CCITT Pin Assignments

Notes:

Index

A

abort commands 73
adapter description 1
additional adapter signal 78
address and interrupt assignments (Async) 18
address and interrupt assignments, (BSC) 48
address assignments
  async 18
  BSC 48
  SDLC/HDLC 77
address, DMA, and interrupt assignments (SDLC/HDLC) 77
async 6
asynchronous communications 6
asynchronous mode 23

B

baud-rate generator 8, 9
binary synchronous communications 21
break interrupt 15
BSC 21

C

C/R/W logic 50
codes, interrupt result 68
command descriptions, 8273 66
command phase programming 62
commands
  abort 73
  counter latch 35
  counter status 37
  general receive 67
  initialization/configuration 66
  modem control 75
  read port A and B 75
  read-back 36
  receive 67
  receive disable 68
  Reset 74
  selective receive 67
  set and reset port B 75
  transmit 72
  transmit frame 72
  transmit transparent 72
common operations, BSC modes 45
  counter operations 45
  gate operations 45
connector 81
control commands 75
control read write logic 50
control register, modem 13
control word register 31
controller, NS16550 6
counter descriptions 30
counter latch command 35
counter status command 37
counter, BSC 45

D

data ready 16
data speed 9
data transfer logic 51
data transfer mode register 61
data-terminal-ready (DTR) 14
descriptions
  adapter 1
  commands, 8273 66
  counter 30
  16550 6
  8251A 21
  8254 30
  8273 49
diagnostic capabilities 6, 13
diagnostic mode (Async) 14
divisor latch access bit (DLAB) 12
DMA assignment 77

DMA transfers 53

E

errors (Async)
framing 15, 16
overrun 15
parity 15
errors (BSC)
framing 29
overrun 29
parity 29
even counts, mode 3 42
execution phase programming 64

F

frame receive 54
frame transmit 54
framing error 15, 16

G

gate input signal 44, 45
generator, baud-rate 8

H

hardware interrupts 19
hardware retriggerable one-shot, mode 1 40
hardware triggered strobe (retriggerable), mode 5 43

I

initialization/configuration commands 66
input signals
CTS (clear-to-send) 78
DCD (data-carrier-detect) 79
DSR (data-set-ready) 78
DTR (data-terminal-ready) 79
modem control 78
OUT 1 (output 1) 79
OUT 2 (output 2) 79
RI (ring indicator) 78
RTS (request-to-send) 79
serial 13
input/output signals
instructions
command 28
mode 23
mode and command 22
instructions, command
enter hunt mode 28
error reset 28
internal reset 28
receiver enable 28
request-to-send 28
send-break-character 28
status read 29
transmitter enable 28
instructions, mode
asynchronous receive 24
asynchronous transmit 23
synchronous receive 26
synchronous transmit 25
interchange information, voltage 80
interface, RS-232-C 78
internal registers 8273 50
interrupt assignments
async 18, 19
bsc 48
sdlc 77
interrupt assignments (Async) 19
interrupt enable register 9
interrupt identification register 10
interrupt information 48, 77
interrupt on terminal count, mode 0 40
interrupt result codes 68
interrupts, hardware 19, 48, 77

L

line control register 11
line status register 15

M

mode and command instructions 22
mode definitions 39
mode instruction 23
mode 0 - interrupt on terminal count 40
mode 1 - hardware retriggerable one-shot 40
mode 2 - rate generator 41
mode 3 - square wave mode 42
mode 4 - programmable triggered strobe 43
mode 5 - hardware triggered strobe (retriggerable) 43
modem
- control block 55
- control inputs 13
- control interrupts 14
- control output port 56
- control outputs 14
- control register 13
- interface 55
- modem control 75
- status register 16

modem control block 55
modem control input signals 78
modem control output signals 79
modem interface 55

N

NS16550 controller 6

O

odd counts, mode 3 42
operating mode register 59
operations, common BSC modes 45
operations, read (BSC) 34
- counter latch 35
- read-back command 36
- simple read 34

operations, write (BSC) 32
output signals
- DTR (data-terminal-ready) 14
- modem control 79
- OUT 1 (output 1) 14
- OUT 2 (output 2) 14
- RTS (request-to-send) 14

overrun error 16

P

parity error 16
pin assignments, connector 81
pin locations, connector 81
port assignments, 8255
- port A 46
- port B 47
- port C 47

programmable baud-rate generator 8
programmable option select (POS) 3
programmable triggered strobe, mode 4 43
programming
- command phase 62
- execution phase 64
- result phase 65
- sequence (BSC) 33

programming sequence (BSC) 46
programming the 16550 7
programming the 8273 58

R

rate generator, mode 2 41
read operations 34
read-back command 36
receive commands 67
receive disable 68
receive, frame 54
receiver buffer register 7
registers, Async
- interrupt enable 9
- interrupt identification 10
- line control 11
- line status 15
- modem control 13
- receiver buffer 7
- scratch pad 17

transmitter holding 7
registers, BSC
control word 31, 35
registers, POS
register 2 4
register 3 5
registers 0 and 1 3
registers, SDLC/HDLC
command 50
data transfer mode 61
immediate result 50
operating mode 59
parameter 50
receiver interrupt results 50
serial I/O mode 60
status 51, 58
test mode 51
transmit interrupt results 50
1-bit delay mode 62
request-to-send (RTS) 14
reset command 74
result codes
abort detect 70
A1/A2 match 69
carrier detect failure 70
clear-to-send error 74
CRC Error 69
DMA overrun 70
DMA underrun 74
early transmit interrupt 74
EOP Detected 70
frame less than 32 bits 70
frame transmit complete 74
idle detect 70
memory buffer overflow 70
receiver interrupt overrun 71
transmitter interrupt result codes 73
result phase programming 65
RS-232-C interface 78

S

scratch pad register 17
SDLC/HDLC communications 49
selective receive command 67
serial data format 20
serial I/O mode register 60
serial input (SIN) 13
serial output 13
serial timing data block 57
simple read operation 34
square wave mode, mode 3 42
status register, modem 16
status register, 8273 58
synchronous mode 25
system interface 50

T

transfers, DMA 53
transmit commands 72
transmit frame command 72
transmit interrupt result codes 73
transmit transparent command 72
transmit, frame 54
transmitter holding register 7
transmitter-empty 15

V

voltage interchange information 80

W

write operations 32

Numerics

1-bit delay mode register 62
16550 description 6
8251A description 21
8251A, programming 22
8254 counter descriptions
counter 0 31
counter 1 31
counter 2 31
8254 description 30

8254 modes
counter operations 45
gate operations 45
Mode 0 40
Mode 1 40
Mode 2 41
Mode 3 42
Mode 4 43
Mode 5 43
8254, programming 31
8255 I/O port assignment 75
8255 port assignments
port A 46
port B 47
port C 47
8273 description 49
8273 port A 55
8273 port B 56
8273 registers
command 50
data transfer mode 61
immediate result 50
operating mode 59
parameter 50
receiver interrupt results 50
serial I/O mode 60
status 51, 58
test mode 51
transmit interrupt results 50
1-bit delay mode 62