Device ID 05H—Parallel Port

Functions

The following are the parallel port functions. The Default Interrupt Handler function (hex 00) and the Return Logical ID Parameters function (hex 01) are described in "Request Block" in the "Transfer Conventions" section.

Note: All reserved input fields must be set to 0.

00H—Default Interrupt Handler

01H—Return Logical ID Parameters

02H—Reserved

03H—Read Device Parameters

- This function returns the device-specific information.
- The Printer Initialization Time to Wait before Resuming Request field contains the wait-time value that is returned by the Reset/Initialize function (hex 05).
- The Printer Interrupt Time-Out field contains the time-out limit for printing. If the print request is performed using programmed I/O, this field represents the time-out limit to print each character. If the print request is performed using direct memory access (DMA), this field represents the time-out limit to print the entire block of characters.
- The value of the Return Code field is hex 0000.

Service-Specific Input

Size	Offset	Description
Word	16H	Reserved

Service-Specific Output

Size	Offset	Description
Byte	16H	Support flags Bit 7 - DMA-support flag = 0 - DMA is not supported = 1 - DMA is supported Bits 6 to 0 - Reserved
Byte	17H	Reserved
DWord	20H	Printer initialization time to wait before resuming request, in microseconds
Byte	29H	Interrupt level
Word	2AH	Printer interrupt time-out Bits 15 to 3 - Time-out, in seconds Bits 2 to 0 - Reserved

04H—Set Device Parameters

- This function sets the device-specific information according to the input parameters.
- The Printer Initialization Time to Wait before Resuming Request field sets the wait-time value that is returned by the Reset/Initialize function (hex 05).
- The Printer Initialization Time to Wait before Resuming Request field must contain a nonzero value. If the field is set to 0, ABIOS sets the Return Code field to hex C006 (Invalid Time to Wait).
- The Printer Interrupt Time-Out field contains the time-out limit for printing. If the print request is performed using programmed I/O, this field represents the time-out limit to print each character. If the print request is performed using direct memory access (DMA), this field represents the time-out limit to print the entire block of characters.
- The Printer Interrupt Time-Out field must contain a nonzero value. If the field is set to 0, ABIOS sets the Return Code field to hex C005 (Invalid Time-Out).
- ABIOS uses these parameters until this function is called to change them.
- The possible values of the Return Code field are hex 0000, 8000, C005, and C006.

Service-Specific Input

Size	Offset	Description
Word	16H	Reserved
DWord	20H	Printer initialization time to wait before resuming request, in microseconds
Word	2AH	Printer interrupt time-out Bits 15 to 3 - Time-out, in seconds Bits 2 to 0 - Reserved

Service-Specific Output

Size	Offset	Description
None		

05H—Reset/Initialize

- This function initializes the printer.
- After performing this function, the printer indicates a busy status while it performs a self-test.
- The Printer Status field is valid only when this function is completed. The status that is returned in the request block is not valid during intermediate stages.
- The possible values of the Return Code field are hex 0000, 0002, and 8000.

Service-Specific Input

Size	Offset	Description
Word	16H	Reserved

Service-Specific Output

Size	Offset	Description
DWord	20H	Time to wait before resuming request, in microseconds
Byte	28H	Printer status Bit 7 = 1 - Busy Bit 6 = 1 - Acknowledge Bit 5 = 1 - End of paper Bit 4 = 1 - Selected Bit 3 = 1 - I/O error Bit 2 = 1 - Interrupt (non-DMA mode only) Bits 1, 0 - Reserved

06H—Enable (Reserved)

07H—Disabled (Reserved)

08H—Read (Reserved)

09H—Print Block

- This function sends a block of characters to the parallel port.
- The Data Pointer 1 field is a logical address to the data that is to be printed. A nonzero value in this field indicates that the pointer is present as input to the Print Block function. When programmed I/O is used, the Data Pointer 1 field must contain a nonzero value. See "Programming Considerations" on page 6-ID05-8 for more details.
- The Data Pointer 2 field is a physical address to the data that is to be printed. A nonzero value in this field indicates that the pointer is present as input to the Print Block function. When direct memory access (DMA) is used, the Data Pointer 2 field must contain a nonzero value. See "Programming Considerations" on page 6-ID05-8 for more details.
- The arbitration-deallocation flag (bit 7) of the Request Flags field enables the device to retain its arbitration level after a DMA transfer is completed. When this flag is set to 0 and a DMA transfer is requested, ABIOS dedicates an arbitration level to the device, and any subsequent DMA requests to this device will use this already-allocated arbitration level. A call to the Cancel Print Block function (hex 0B) can be used to deallocate the dedicated arbitration level. When the arbitration-deallocation flag is set to 1 and a DMA transfer is requested, ABIOS automatically deallocates the arbitration level when the transfer is completed.
- When a Print Block function is in progress, the caller must use the Cancel Print Block function (hex 0B) to cancel it before issuing another Print Block function to the same unit.
- When a print error occurs, when the printer is offline, or when the printer is busy, ABIOS terminates the print-block request. The Number of Characters Printed field indicates which portion of the print block has been printed. To print the unprinted portion of the print block, issue another print-block request when the terminating condition has been corrected.
- The Printer Status field is valid only when the function is successfully completed (return code hex 0000) or when the function is terminated because of an error condition. The status that is returned in the Request Block is not valid in intermediate stages.
- The possible values of the Return Code field are hex 0000, 0001, 0005, 8000, 8001, 8008, 8081, 8082, 8083, 8084, 8085, 8086, 8087, 9000, and C007.

Service-Specific Input

Size	Offset	Description
Word	10H	Reserved
DWord	12H	Data pointer 1
Word	16H	Request flags Bit 7 - Arbitration deallocation flag = 0 - Do not deallocate = 1 - Deallocate Bit 6 - Arbitration allocation flag = 0 - Do not assume that the arbitration level is allocated by the operating system = 1 - Assume that the arbitration level is allocated by the operating system Bits 5 to 0 - Reserved
Byte	17H	Reserved
Word	18H	Reserved
DWord	1AH	Data pointer 2
Word	1EH	Reserved
Word	24H	Number of characters to be printed

Service-Specific Output

Size	Offset	Description
Word	26H	Number of characters printed
Byte	28H	Printer status Bit 7 = 1 - Busy Bit 6 = 1 - Acknowledge Bit 5 = 1 - End of paper Bit 4 = 1 - Selected Bit 3 = 1 - I/O error Bit 2 = 1 - Interrupt (non-DMA mode only) Bits 1, 0 - Reserved

0AH—Additional Data Transfer (Reserved)

0BH—Cancel Print Block

- This function cancels an outstanding Print Block function (hex 09) request and deallocates any arbitration level that is allocated to the device.
- The arbitration-deallocation flag (bit 7) of the Request Flags field enables the device to retain its arbitration level after the Cancel Print Block function is completed. When this flag is set to 1, ABIOS deallocates any arbitration level that is currently allocated to the device. When this flag is set to 0, ABIOS does not deallocate any arbitration level that is currently allocated to the device.
- The Number of Characters Printed field is valid only when this function is used to cancel an outstanding Print Block function (hex

09) request and when the function is called with the same request block that was used to begin the Print Block function request.

- The Printer Status field is valid only when this function is completed. The status that is returned in the request block is not valid during intermediate stages.
- The value of the Return Code field is hex 0000.

Service-Specific Input

Size	Offset	Description
Byte	16H	Request flags Bit 7 - Arbitration-deallocation flag = 0 - Do not deallocate = 1 - Deallocate Bit 6 - Arbitration allocation flag = 0 - Do not assume that the arbitration level is deallocated by the operating system = 1 - Assume that the arbitration level is deallocated by the operating system Bits 5 to 0 - Reserved
Byte	17H	Reserved

Service-Specific Output

Size	Offset	Description
Word	26H	Number of characters printed
Byte	28H	Printer status Bit 7 = 1 - Busy Bit 6 = 1 - Acknowledge Bit 5 = 1 - End of paper Bit 4 = 1 - Selected Bit 3 = 1 - I/O Error Bit 2 = 1 - Interrupt (non-DMA mode only) Bits 1, 0 - Reserved

0CH—Return Printer Status

- This function returns the printer status.
- The Printer Status field is valid only when the function is successfully completed.
- The possible values of the Return Code field are hex 0000 and 8000.

Service-Specific Input

Size	Offset	Description
Word	16H	Reserved

Service-Specific Output

Size	Offset	Description
Byte	28H	Printer status Bit 7 = 1 - Busy Bit 6 = 1 - Acknowledge Bit 5 = 1 - End of paper Bit 4 = 1 - Selected Bit 3 = 1 - I/O error Bit 2 = 1 - Interrupt (non-DMA mode only) Bits 1, 0 - Reserved

Return Codes

Return codes are returned at offset hex 0C.

Value	Description
0000H	Operation Successfully Completed
0001H	Stage on Interrupt
0002H	Stage on Time
0005H	Not My Interrupt, Stage on Interrupt
8000H	Device in Use
8001H	Device Busy
8008H	DMA Not Supported
8081H	Arbitration Level Not Available
8082H	Arbitration Level Not Allocated
8083H	Arbitration Level Disabled
8084H	Transfer in Progress
8085H	No Transfer in Progress
8086H	No DMA Channel Available
8087H	Arbitration Level Not Disabled
9000H	Printer Error
C000H	Invalid Logical ID (ABIOS transfer convention only)
C001H	Invalid Function Number
C003H	Invalid Unit Number
C004H	Invalid Request-Block Length
C005H	Invalid Time-Out
C006H	Invalid Time to Wait
C007H	Invalid Data-Pointer Values

Figure 6-12. Parallel Port Return Codes

Programming Considerations

- The Return Logical ID Parameters function (hex 01) indicates that data pointer 1 and data pointer 2 are expected by the Print Block function (hex 09).
- The Print Block function (hex 09) does not recognize 0 as a valid data pointer. If both data pointer 1 and data pointer 2 are set to 0, the request is denied, and ABIOS sets the Return Code field to hex C007 (Invalid Data-Pointer Values). The following table shows how the Print Block function behaves under all possible combinations of hardware support and data-pointer fields in the request block.

DMA Support Hardware	Nonzero Data Pointer 1	Nonzero Data Pointer 2	Performed Action
N	N	N	Error (C007H)
N	N	Y	Error (8008H)
N	Y	N	PIO
N	Y	Y	PIO
Y	N	N	Error (C007H)
Y	N	Y	DMA*
Y	Y	N	PIO
Y	Y	Y	DMA**

* Indicates that if there are no DMA channels or the arbitration level is not available, the request is denied with an error condition. There is no default to programmed I/O (PIO).

** Indicates that if there are no DMA channels or the arbitration level is not available, the request is performed using programmed I/O (PIO).

Figure 6-13. Print Block Functions

- If the Printer Status field is busy when a Print Block function (hex 09) is requested, ABIOS checks the printer status for a specified period of time, and the function is terminated if the device is still busy after that time. ABIOS sets the Return Code field to hex 8001 (Device Busy). The Number of Characters Printed field indicates the number of characters that were printed.
- If the Print Block function (hex 09) is in programmed I/O mode and the printer is put offline in the middle of a print block, ABIOS checks the printer status for a specified period of time, and the function is terminated if the device is still busy after that time. ABIOS sets the Return Code field to hex 8001 (Device Busy). The caller can issue a new Print Block function to print the remaining characters when the printer is put back online.
- If the Print Block function is in DMA mode and there are any transitions in the printer lines (such as "offline," "out of paper," or "error"), ABIOS terminates the request. An appropriate return code (hex 8000, 8001, or 9000) and the number of characters that were printed are returned in the request block. The caller can issue a new Print Block function to print the remaining characters when the condition has been corrected.
- When the Reset/Initialize function (hex 05) is called, some printers perform a self-test that causes the printer to be busy until the self-test is completed. The busy bit (bit 7) of the Printer Status field indicates this busy condition.
- Parallel port ABIOS supports the parallel port in transmit mode only.

Notes:

Device ID 06H—Asynchronous Communication

Functions

The following are the asynchronous communication functions for the serial port. The Default Interrupt Handler function (hex 00) and the Return Logical ID Parameters function (hex 01) are described in "Request Block" in the "Transfer Conventions" section.

Note: All reserved input fields must be set to 0.

00H—Default Interrupt Handler

01H—Return Logical ID Parameters

02H—Reserved

03H—Read Device Parameters

- This function returns the serial-port information. It has no effect on any other outstanding request, and it does not interact with the hardware.
- The parameters are maintained in the device block as a shadow of the hardware. The shadow is updated when requests are made through ABIOS. If the serial ports are programmed directly or through the BIOS Interrupt 14H functions, the asynchronous parameters in the device block will be incorrect. In this case, the Read Device Parameters function returns the previously-stored values.
- To synchronize these parameters, call the Reset/Initialize function (hex 05).
- During ABIOS initialization, the serial port is initialized to the following default parameters:
  - Baud rate of 1200
  - No parity
  - One stop bit
  - Seven bits per character
  - No break.
- The three Compare Character Match fields enable programming of receive-match characters in receive operations.
- If a character that is received from the controller matches the value in one of the three enabled Compare Character Match

fields, an action is performed as defined by the three Compare Character Function fields.

- The data in a Compare Character Match field is an 8-bit match character. If the word length is less than 8 bits, the match character must be right justified, and any unused bits must be set to 0.
- If null stripping is enabled, the Compare Character Match 3 field is unavailable.
- The three Compare Character Function fields correspond to the three Compare Character Match fields. They enable programming of the controller to start the transmitter, stop the transmitter, delete a match character from an incoming data stream, or to interrupt.
- Multiple Compare Character Match functions per Compare Character function are supported.
- If bit 3 (start transmitter) and bit 2 (stop transmitter) of the Compare Character Function field are both set to 1, bit 2 takes precedence.
- If bit 0 (interrupt) is set to 1, the interrupt occurs as soon as a match with a received data character is detected.
- To disable a Compare Character Match field, set the corresponding Compare Character Function field to 0 before calling the Set Compare Character Match and Function function (hex 1B).
- The Modem Control field returns the active modem-control functions. The Enhanced Function Control field returns additional modem-control functions.
- The baud rate is calculated by the following formula:

  ```
  Baud clock/Scaler/Baud rate = Divisor count
  ```

- The baud clock for the type-3 controller has two internal frequencies: 22.1184 MHz with a scaler of 32, and 1.8432 MHz with a scaler of 16. The baud clock for the type-1 and type-2 controllers has an internal frequency of 1.8432 MHz with a scaler of 16. The frequency is divided internally by a scaler to increase the sample time per character bit.

- The Binary Baud Rate field contains the values that are used to calculate the baud rate when the Communication Baud Rate field is set to hex 0FF. The Binary Baud Rate field consists of a 24-bit value plus an 8-bit fraction that is described in the request block. The divisor count is a 16-bit value that is loaded into the divisor latch of the serial device to generate the final baud-rate clock. Any fractional amount is rounded to the nearest divisor count to get the closest baud rate. This integer is put into the following formula to get the values that are returned as output in the request block:

  ```
  Baud clock/Scaler/Divisor count = Output baud rate
  ```

- The value of the Return Code field is hex 0000.

Service-Specific Input

Size	Offset	Description
Word	18H	Reserved

Service-Specific Output

Size	Offset	Description
Byte	10H	Logical ID interrupt level
Word	18H	Enhanced function status Bit 15 - Binary baud rate support = 0 - Binary baud functions are not supported = 1 - Binary baud functions are supported Bits 14 to 9 - Reserved Bit 8 - 'High data rate' signal status = 0 - 'High data rate' signal is inactive = 1 - 'High data rate' signal is active Bit 7 - Transmit byte pacing = 0 - Not enabled = 1 - Enabled Bit 6 - High-frequency rate = 0 - High-frequency rate not enabled (1.8432 MHz) = 1 - High-frequency rate enabled (22.1184 MHz) Bit 5 - Slow transmit rate (not available when transmit byte pacing is used) = 0 - Transmit at baud rate = 1 - Transmit at 1/16 baud rate Bit 4 - Slow receive rate (not available when transmit byte pacing is used) = 0 - Receive at baud rate = 1 - Receive at 1/16 baud rate

Size	Offset	Description
		Bit 3 - Control receiver via data-set-ready (DSR) = 0 - No receiver control via DSR = 1 - If DSR = 0, the control receiver is turned off; if DSR = 1, the control receiver is turned on; if DSR drops, the character currently being received is discarded. Bit 2 - Control transmitter via data-carrier-detect (DCD) = 0 - No transmitter control via DCD = 1 - If DCD = 0, the control transmitter is turned off; if DCD = 1, the control transmitter is turned on; the transmitter must have initially been turned on by a Start Sending command. Bit 1 - Control transmitter via DSR = 0 - No transmitter control via DSR = 1 - If DSR = 0, the control transmitter is turned off; if DSR = 1, the control transmitter is turned on; the transmitter must have initially been turned on by a Start Sending command. Bit 0 - Control transmitter via clear-to-send (CTS) = 0 - No transmitter control via CTS = 1 - If CTS = 0, the control transmitter is turned off; if CTS = 1, the control transmitter is turned on; the transmitter must have initially been turned on by a Start Sending command.
Byte	1AH	Compare character match 1
Byte	1BH	Compare character function 1 Bits 7 to 4 - Reserved Bit 3 - Start transmitter on a character match = 0 - Transmitter not started on a character match = 1 - Transmitter started on a character match (the transmitter will not start if bit 2 is set to 1) Bit 2 - Stop transmitter on a character match = 0 - Transmitter not stopped on a character match = 1 - Transmitter stopped on a character match (if the transmitter is stopped with a character match, a call to the Start Transmission function (hex 18) can be used to restart the transmitter) Bit 1 - Delete character on a character match = 0 - Character not deleted on a character match = 1 - Character deleted on a character match Bit 0 - Interrupt on character match = 0 - Interrupt not enabled on a character match = 1 - Interrupt enabled on a character match (the interrupt occurs as soon as there is a match with a received data character)
Byte	1CH	Compare character match 2
Byte	1DH	Compare character function 2 (see the description of the Compare Character Function 1 field)
Byte	1EH	Compare character match 3
Byte	1FH	Compare character function 3 (see the description of the Compare Character Function 1 field)

Size	Offset	Description
Byte	28H	Modem control Bits 7 to 2 - Reserved Bit 1 - 'Request to send' (RTS) signal status = 0 - RTS signal is disabled = 1 - RTS signal is enabled Bit 0 - 'Data terminal ready' (DTR) signal status = 0 - DTR signal is disabled = 1 - DTR signal is enabled
Byte	29H	Asynchronous interrupt status word Bit 7 - DMA receive mode = 0 - Disabled = 1 - Enabled Bit 6 - DMA transmit mode = 0 - Disabled = 1 - Enabled Bit 5 - Modem status interrupt = 0 - Disabled = 1 - Enabled Bit 4 - Receive-line status interrupt = 0 - Disabled = 1 - Enabled Bit 3 - Transmit interrupt = 0 - Disabled = 1 - Enabled Bit 2 - Receive interrupt = 0 - Disabled = 1 - Enabled Bit 1 - Receiver character count = 0 - Disabled = 1 - Enabled Bit 0 - 'High data rate' signal selector interrupt = 0 - Disabled = 1 - Enabled
Byte	2AH	Receive trigger-level status = 00H - 1 byte = 01H - 4 bytes = 02H - 8 bytes = 03H - 14 bytes = 04H to FFH - Reserved
Byte	2BH	FIFO-mode status Bit 7 - Dynamic arbitration allocation = 0 - Arbitration levels are fixed = 1 - Arbitration levels are programmable Bits 6 to 4 - Reserved Bit 3 - FIFO support available in ABIOS = 0 - No ABIOS support for FIFO = 1 - ABIOS supports FIFO Bit 2 - DMA transfer available = 0 - No DMA transfer capability = 1 - Device capable of data transfer via DMA Bit 1 - Serial device has FIFO ability = 0 - FIFO not available in serial device = 1 - FIFO available in serial device Bit 0 - Transmit and receive FIFO mode status = 0 - Serial device operates in character mode = 1 - Serial device operates in FIFO mode

Size	Offset	Description
Byte	40H	8-bit fraction of binary baud rate (if offset hex 44 is set to hex FF)
Byte	41H	Least-significant byte of binary baud rate (if offset hex 44 is set to hex FF)
Byte	42H	Next-significant byte of binary baud rate (if offset hex 44 is set to hex FF)
Byte	43H	Most-significant byte of binary baud rate (if offset hex 44 is set to hex FF)
Byte	44H	Communication baud rate = 00H - 110 = 01H - 150 = 02H - 300 = 03H - 600 = 04H - 1200 = 05H - 2400 = 06H - 4800 = 07H - 9600 = 08H - 19200 = 09H - 31418.2 = 0AH - 38400 = 0BH - 57600 = 0CH - 76800 = 0DH - 115200 = 0EH - 138240 = 0FH - 172800 = 10H - 230400 = 11H - 345600 = 12H to FEH - Reserved = FFH - Calculate baud rate from variable at offset hex 40
Byte	45H	Type of parity = 00H - None = 01H - Odd = 02H - Even = 03H - Stick parity odd = 04H - Stick parity even = 05H to FFH - Reserved
Byte	46H	Stop bit = 00H - The stop-bit length is 1 bit = 01H - If the data-bit length is 6, 7, or 8 bits, the stop-bit length is 2 bits; If the data-bit length is 5 bits, the stop-bit length is 1½ bits. = 02H to FFH - Reserved
Byte	47H	Data-bit length = 00H - 5 bits = 01H - 6 bits = 02H - 7 bits = 03H - 8 bits = 04H to 0FFH - Reserved
Byte	48H	Break status = 00H - Disabled = 01H - Enabled = 02H to FFH - Reserved
Byte	49H	Reserved
Byte	4AH	Reserved

04H—Reserved

05H—Reset/Initialize

- This function initializes the serial port according to the input parameters.
- All communication interrupts (receive, transmit, and modem status) are disabled. The caller is responsible for clearing all outstanding request blocks and the interrupt controller, where appropriate. From the ABIOS standpoint, all outstanding request blocks are canceled.
- Any received data that is pending at the serial port is cleared.
- The Reset/Initialize function is used to synchronize the device-block parameters with the current Hardware port values in preparation for a Read Device Parameters function (hex 03) call.
- The Modem Control field enables the setting of modem-control functions. The Enhanced Function Control field enables the setting of additional modem-control functions.
- The baud rate is calculated by the following formula:

  ```
  Baud clock/Scaler/Baud rate = Divisor count
  ```

- The baud clock for the type-3 controller has two internal frequencies: 22.1184 MHz with a scaler of 32, and 1.8432 MHz with a scaler of 16. The baud clock for the type-1 and type-2 controllers has an internal frequency of 1.8432 MHz with a scaler of 16. The frequency is divided internally by a scaler to increase the sample time per character bit.
- The Binary Baud Rate field contains the values that are used to calculate the baud rate when the Communication Baud Rate field is set to hex 0FF. The Binary Baud Rate field consists of a 24-bit value plus an 8-bit fraction that is described in the request block. The divisor count is a 16-bit value that is loaded into the divisor latch of the serial device to generate the final baud-rate clock. Any fractional amount is rounded to the nearest divisor count to get the closest baud rate. This integer is put into the following formula to get the values that are returned as output in the request block:

  ```
  Baud clock/Scaler/Divisor count = Output baud rate
  ```

- The caller is responsible for determining baud-rate tolerances. To determine the maximum or minimum baud rate of the serial device, enter hex 0FFFFFFFF (for the maximum) or hex 0 (for the minimum) into the input parameters. On output, the Binary Baud Rate fields contain the limit for the parameter that was entered.

- For type-3 controllers, baud rates above 19200 bits per second can be selected only from the Communication Baud Rate field (at offset hex 44).
- The FIFO trigger level is programmed only if the FIFO mode is enabled.
- The value of the Return Code field is hex 0000.

Service-Specific Input

Size	Offset	Description
Word	18H	Enhanced function control Bits 15 to 6 - Reserved Bit 5 - Slow transmit rate (not available when transmit byte pacing is used) = 0 - Transmit at baud rate = 1 - Transmit at 1/16 baud rate Bit 4 - Slow receive rate (not available when transmit byte pacing is used) = 0 - Receive at baud rate = 1 - Receive at 1/16 baud rate Bit 3 - Control receiver via DSR = 0 - No receiver control via DSR = 1 - If DSR = 0, the control receiver is turned off; if DSR = 1, the control receiver is turned on; if DSR drops, the character currently being received is discarded. Bit 2 - Control transmitter via DCD = 0 - No transmitter control via DCD = 1 - If DCD = 0, the control transmitter is turned off; if DCD = 1, the control transmitter is turned on; the transmitter must have initially been turned on by a Start Sending command. Bit 1 - Control transmitter via DSR = 0 - No transmitter control via DSR = 1 - If DSR = 0, the control transmitter is turned off; if DSR = 1, the control transmitter is turned on; the transmitter must have initially been turned on by a Start Sending command. Bit 0 - Control transmitter via CTS = 0 - No transmitter control via CTS = 1 - If CTS = 0, the control transmitter is turned off; if CTS = 1, the control transmitter is turned on; the transmitter must have initially been turned on by a Start Sending command.
Byte	28H	Modem control Bits 7 to 2 - Reserved Bit 1 - RTS signal status = 0 - RTS signal is disabled = 1 - RTS signal is enabled Bit 0 - DTR signal status = 0 - DTR signal is disabled = 1 - DTR signal is enabled

Size	Offset	Description
Byte	29H	FIFO-mode control = 00H - Disable = 01H - Enable and reset FIFO = 02H - Enable without resetting FIFO = 03H to FFH - Reserved
Byte	2AH	Receive trigger level = 00H - 1 byte = 01H - 4 bytes = 02H - 8 bytes = 03H - 14 bytes = 04H to FFH - Reserved
Byte	40H	8-bit fraction of binary baud rate (if offset hex 44 is set to hex FF)
Byte	41H	Least-significant byte of binary baud rate (if offset hex 44 is set to hex FF)
Byte	42H	Next-significant byte of binary baud rate (if offset hex 44 is set to hex FF)
Byte	43H	Most-significant byte of binary baud rate (if offset hex 44 is set to hex FF)
Byte	44H	Communication baud rate = 00H - 110 = 01H - 150 = 02H - 300 = 03H - 600 = 04H - 1200 = 05H - 2400 = 06H - 4800 = 07H - 9600 = 08H - 19200 = 09H - 31418.2 = 0AH - 38400 = 0BH - 57600 = 0CH - 76800 = 0DH - 115200 = 0EH - 138240 = 0FH - 172800 = 10H - 230400 = 11H - 345600 = 12H to FEH - Reserved = FFH - Calculate baud rate from variable at offset hex 40
Byte	45H	Type of parity = 00H - None = 01H - Odd = 02H - Even = 03H - Stick parity odd = 04H - Stick parity even = 05H to FFH - Reserved
Byte	46H	Stop bit = 00H - The stop-bit length is 1 bit = 01H - If the data-bit length is 6, 7, or 8 bits, the stop-bit length is 2 bits; If the data-bit length is 5 bits, the stop-bit length is 1½ bits. = 02H to FFH - Reserved

Size	Offset	Description
Byte	47H	Data-bit length = 00H - 5 bits = 01H - 6 bits = 02H - 7 bits = 03H - 8 bits = 04H to 0FFH - Reserved
Byte	48H	Break status = 00H - Disable = 01H - Enable = 02H to FFH - Reserved

Service-Specific Output

Size	Offset	Description
Byte	40H	8-bit fraction of binary baud rate (if offset hex 44 is set to hex FF)
Byte	41H	Least-significant byte of binary baud rate (if offset hex 44 is set to hex FF)
Byte	42H	Next-significant byte of binary baud rate (if offset hex 44 is set to hex FF)
Byte	43H	Most-significant byte of binary baud rate (if offset hex 44 is set to hex FF)
Byte	49H	Line status Bit 7 - Error in receiver FIFO = 0 - No error in FIFO = 1 - Error in receiver FIFO Bit 6 - Transmitter empty = 0 - Inactive = 1 - Active Bit 5 - Transmitter holding register empty = 0 - Inactive = 1 - Active Bit 4 - Break interrupt = 0 - Inactive = 1 - Active Bit 3 - Framing error = 0 - Inactive = 1 - Active Bit 2 - Parity error = 0 - Inactive = 1 - Active Bit 1 - Overrun error = 0 - Inactive = 1 - Active Bit 0 - Data ready = 0 - Inactive = 1 - Active

Size	Offset	Description
Byte	4AH	Modem status Bit 7 - Data-carrier detect = 0 - Inactive = 1 - Active Bit 6 - Ring indicator = 0 - Inactive = 1 - Active Bit 5 - Data set ready = 0 - Inactive = 1 - Active Bit 4 - Clear to send = 0 - Inactive = 1 - Active Bit 3 - Delta data-carrier detect = 0 - Inactive = 1 - Active Bit 2 - Trailing-edge ring indicator = 0 - Inactive = 1 - Active Bit 1 - Delta data set ready = 0 - Inactive = 1 - Active Bit 0 - Delta clear to send = 0 - Inactive = 1 - Active

06H—Enable (Reserved)

07H—Disable (Reserved)

08H—Read (Reserved)

09H—Write (Reserved)

0AH—Additional Data Transfer (Reserved)

0BH—Set Modem Control

- This function sets the modem control according to the input parameters. This function does not affect the interrupt state of any other stage-on-interrupt requests.
- The Modem Control field enables the setting of modem-control functions. The Enhanced Function Control field enables the setting of additional modem-control functions.
- The value of the Return Code field is hex 0000.

Service-Specific Input

Size	Offset	Description
Word	18H	Enhanced function control Bits 15 to 4 - Reserved Bit 3 - Control receiver via DSR = 0 - No receiver control via DSR = 1 - If DSR = 0, the control receiver is turned off; if DSR = 1, the control receiver is turned on; if DSR drops, the character currently being received is discarded. Bit 2 - Control transmitter via DCD = 0 - No transmitter control via DCD = 1 - If DCD = 0, the control transmitter is turned off; if DCD = 1, the control transmitter is turned on; the transmitter must have initially been turned on by a Start Sending command. Bit 1 - Control transmitter via DSR = 0 - No transmitter control via DSR = 1 - If DSR = 0, the control transmitter is turned off; if DSR = 1, the control transmitter is turned on; the transmitter must have initially been turned on by a Start Sending command. Bit 0 - Control transmitter via CTS = 0 - No transmitter control via CTS = 1 - If CTS = 0, the control transmitter is turned off; if CTS = 1, the control transmitter is turned on; the transmitter must have initially been turned on by a Start Sending command.
Byte	28H	Modem control Bits 7 to 2 - Reserved Bit 1 - RTS signal status = 0 - RTS signal is disabled = 1 - RTS signal is enabled Bit 0 - DTR signal status = 0 - DTR signal is disabled = 1 - DTR signal is enabled

Service-Specific Output

Size	Offset	Description
None		

0CH—Set Line Control

- This function sets the line control according to the input parameters. It does not affect the interrupt state of any other stage-on-interrupt requests.
- The value of the Return Code field is hex 0000.

Service-Specific Input

Size	Offset	Description
Word	18H	Reserved
Byte	45H	Type of parity = 00H - None = 01H - Odd = 02H - Even = 03H - Stick parity odd = 04H - Stick parity even = 05H to FFH - Reserved
Byte	46H	Stop bit = 00H - The stop-bit length is 1 bit = 01H - If the data-bit length is 6, 7, or 8 bits, the stop-bit length is 2 bits; If the data-bit length is 5 bits, the stop-bit length is 1½ bits. = 02H to FFH - Reserved
Byte	47H	Data-bit length = 00H - 5 bits = 01H - 6 bits = 02H - 7 bits = 03H - 8 bits = 04H to 0FFH - Reserved
Byte	48H	Break status = 00H - Disabled = 01H - Enabled = 02H to FFH - Reserved

Service-Specific Output

Size	Offset	Description
None		

0DH—Set Baud Rate

- This function sets the baud rate according to the input parameter. It does not affect the interrupt state of any other stage-on-interrupt requests.
- The baud rate is calculated by the following formula:

  ```
  Baud clock/Scaler/Baud rate = Divisor count
  ```

- The baud clock for the type-3 controller has two internal frequencies: 22.1184 MHz with a scaler of 32, and 1.8432 MHz with a scaler of 16. The baud clock for the type-1 and type-2 controllers has an internal frequency of 1.8432 MHz with a scaler of 16. The frequency is divided internally by a scaler to increase the sample time per character bit.

- The Binary Baud Rate field contains the values that are used to calculate the baud rate when the Communication Baud Rate field is set to hex 0FF. The Binary Baud Rate field consists of a 24-bit value plus an 8-bit fraction that is described in the request block. The divisor count is a 16-bit value that is loaded into the divisor latch of the serial device to generate the final baud-rate clock. Any fractional amount is rounded to the nearest divisor count to get the closest baud rate. This integer is put into the following formula to get the values that are returned as output in the request block:

  ```
  Baud clock/Scaler/Divisor count = Output baud rate
  ```

- The caller is responsible for determining baud-rate tolerances. To determine the maximum or minimum baud rate of the serial device, enter hex 0FFFFFFFF (for the maximum) or hex 0 (for the minimum) into the input parameters. On output, the Binary Baud Rate fields contain the limit for the parameter that was entered.
- For type-3 controllers, baud rates above 19200 bits per second can be selected only from the Communication Baud Rate field (at offset hex 44).
- The value of the Return Code field is hex 0000.

Service-Specific Input

Size	Offset	Description
Word	18H	Enhanced function control Bits 15 to 6 - Reserved Bit 5 - Slow transmit rate (not available when transmit byte pacing is used) = 0 - Transmit at baud rate = 1 - Transmit at 1/16 baud rate Bit 4 - Slow receive rate (not available when transmit byte pacing is used) = 0 - Receive at baud rate = 1 - Receive at 1/16 baud rate Bits 3 to 0 - Reserved
Byte	40H	8-bit fraction of binary baud rate (if offset hex 44 is set to hex FF)
Byte	41H	Least-significant byte of binary baud rate (if offset hex 44 is set to hex FF)
Byte	42H	Next-significant byte of binary baud rate (if offset hex 44 is set to hex FF)
Byte	43H	Most-significant byte of binary baud rate (if offset hex 44 is set to hex FF)

Size	Offset	Description
Byte	44H	Communication baud rate = 00H - 110 = 01H - 150 = 02H - 300 = 03H - 600 = 04H - 1200 = 05H - 2400 = 06H - 4800 = 07H - 9600 = 08H - 19200 = 09H - 31418.2 = 0AH - 38400 = 0BH - 57600 = 0CH - 76800 = 0DH - 115200 = 0EH - 138240 = 0FH - 172800 = 10H - 230400 = 11H - 345600 = 12H to FEH - Reserved = FFH - Calculate baud rate from variable at offset hex 40

Service-Specific Output

Size	Offset	Description
Byte	40H	8-bit fraction of binary baud rate (if offset hex 44 is set to hex FF)
Byte	41H	Least-significant byte of binary baud rate (if offset hex 44 is set to hex FF)
Byte	42H	Next-significant byte of binary baud rate (if offset hex 44 is set to hex FF)
Byte	43H	Most-significant byte of binary baud rate (if offset hex 44 is set to hex FF)
Byte	44H	Communication baud rate (if hex FF is selected, the value in offset hex 40 is returned; otherwise, no values are changed)

0EH—Transmit Interrupt

- This function provides several modes for transferring data to the transmitter in a serial port.
- Data pointers that describe a buffer area are passed in a request block to ABIOS. The data in the buffer is loaded into the transmitter in an asynchronous device. Each time the transmitter is emptied, it is reloaded. For a system that supports DMA mode, several options are available for reloading the transmitter. Output parameters are returned to inform the program of transmit-operation progress.
- There are two modes for transmitting data: programmed input/output (PIO) mode and direct memory access (DMA) mode.

- A variety of transmit options are available to support PIO mode and DMA mode. These options are controlled through bit settings in the Enhanced Mode field (offset hex 18) and the Transmit Control field (offset hex 44). When a DMA-transmit operation is requested, options can be selected to assist the program in completing the data transfer. If DMA mode is selected but is not available, an option is available to default to PIO mode or to return an error. Depending on the mode that is selected, logical pointers, physical pointers, or both are required. An option is also available to deallocate the DMA arbitration level and channel when the data transfer is complete.
- The caller must initialize the Operation Status field to 0 before calling the Start routine. None of the bits in the Operation Status field takes precedence over any other bits in this field.
- If the value of the Number of Bits Per Character field (offset hex 47) is less than 8, 8 bits are sent to the serial port, but only the number of bits that is specified in the Number of Bits Per Character field are transmitted.
- The Additional Operation Status field is defined only in systems that support DMA mode. Therefore, do not use the Additional Operation Status field if bit 2 of the FIFO-Mode Status field (offset hex 2B on output) is set to 0.
- The possible values of the Return Code field are hex 0000, 0001, 0005, 0081, 8000, 8001, 8081, 8082, 8083, 8084, 8085, 8086, 8087, and 9000.

Programmed Input/Output (PIO) Transmit

- Programmed input/output is a technique in which serial ports transfer data through interrupts.
- Each time the transmitter is emptied, an interrupt is generated, causing a branching to the Transmit Interrupt function (hex 0E). The Transmit Interrupt function fills the transmitter, using the processing-unit I/O instructions. This process is repeated until all the data in the buffer has been transmitted.
- In early serial ports, the transmitter holds only one character at a time. In more advanced serial ports, the transmitter holds a 16-character, first-in, first-out (FIFO) buffer, allowing more characters to be sent from the data buffer to the transmitter each time an interrupt occurs.
- When bit 0 of the Enhanced Mode field is set to 0, PIO mode is enabled.

- ABIOS enables the Transmit interrupt but does not transmit any data until a Transmit interrupt occurs.
- The Transmit Tail Pointer field points to the first byte of data that is to be transmitted. The Transmit Head Pointer field points to one byte logically beyond the last byte that is to be transmitted.
- The values of the Transmit Head Pointer field and the Transmit Tail Pointer field are relative to the beginning of the transmit buffer, where a value of 0 indicates the first physical byte of the buffer and the value of the Transmit-Buffer Length field minus 1 indicates the last physical byte of the buffer. The values of the Transmit Head Pointer field and the Transmit Tail Pointer field must never be out of that range.
- The maximum number of characters that the transmit buffer can indicate to be sent at any time is the value of the Transmit-Buffer Length field minus 1.
- When a Transmit interrupt occurs, it increases the value in the Transmit Tail Pointer field by the number of bytes that were transmitted. Bit 1 of the Operation Status field (offset hex 4B on output) is set to 1. Because of the possibility that the buffer will be checked asynchronously by the Transmit interrupt, the data is written to the transmit buffer before the value of the Transmit Tail Pointer field is increased.
- As interrupts occur, the value of the Transmit Tail Pointer field approaches the value of the Transmit Head Pointer field. If the value of the Transmit Tail Pointer field reaches the end of the transmit buffer, the Transmit Tail Pointer field is set to 0.
- A transmit-buffer-empty condition occurs when the value of the Transmit Tail Pointer field equals the value of the Transmit Head Pointer field. During processing of a Transmit interrupt, if a transmit-buffer-empty condition occurs after data is sent to the serial port, ABIOS stops sending data to the serial port and informs the caller of the condition. Bit 6 of the Operation Status field is set to 1, indicating that the transmit buffer is empty but the Transmit interrupt is still enabled. On any subsequent Transmit interrupt, if the transmit-buffer-empty condition exists, ABIOS disables the Transmit interrupt, the request block is considered to be complete, and bit 7 of the Operation Status field is set to 1. Bits 6 and 7 of the Operation Status field are mutually and exclusively set.
- A transmit-buffer-full condition occurs when the value of the Transmit Head Pointer field equals the value of the Transmit Tail Pointer field minus 1. A transmit-buffer-full condition occurs also when the value of the Transmit Tail Pointer field is 0 and the

value of the Transmit Head Pointer field equals the value of the Transmit-Buffer Length field minus 1.

- The values of the Transmit-Buffer Segment field, the Transmit-Buffer Offset field, and the Transmit-Buffer Length field can be altered during calls to the Transmit interrupt. Because ABIOS removes the data from the transmit buffer before changing the value of the Transmit Tail Pointer field, the caller can put additional data into the buffer and logically increase the value of the Transmit Head Pointer field during processing of a Transmit interrupt. The caller must not allow the value of the Transmit Head Pointer field to equal the value of the Transmit Tail Pointer field, because this indicates a transmit-buffer-empty condition.
- The following fields must be initialized and passed for a PIO-transmit operation:
  - Transmit-Buffer Offset field
  - Transmit-Buffer Segment field
  - Enhanced Mode field
  - Transmit-Buffer Length field
  - Transmit-Buffer Head field
  - Transmit-Buffer Tail field
  - Transmit Byte Pacing Rate field (if bit 2 of the Enhanced Mode field is set to 1).
- Transmit byte pacing is available only in PIO mode.
- The transmit-byte-pacing bit (bit 2 of the Enhanced Mode field) enables pacing of the transmitter. If this bit is set to 1, the serial port is programmed to wait a specified length of time before sending the next character. The delay is specified in the Transmit Byte Pacing Rate field. The value in this field represents the number of 16-bit packets that are contained within the delay between characters. Therefore, the delay time (in seconds) between characters is calculated as follows:

$$\frac{\text{Transmit byte pacing rate} \times 16}{\text{Baud rate}}$$

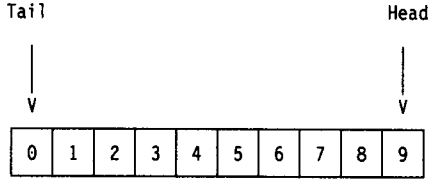
  Each time the transmit-byte-pacing-rate counter decreases to 0, an interrupt occurs to enable ABIOS to reload the counter.
- If a transmit operation is enabled by bit 2 of the Enhanced Mode field after the Receive Interrupt function (hex 0F) is enabled by bit 3 of the Enhanced Mode field (in function hex 0F), the interrupt-on-receiver-character-count=0 option is disabled. The

functions that are associated with bit 2 of the Enhanced Mode field (in function hex 0E) and bit 3 of the Enhanced Mode field (in function hex 0F) are mutually exclusive at the serial port.

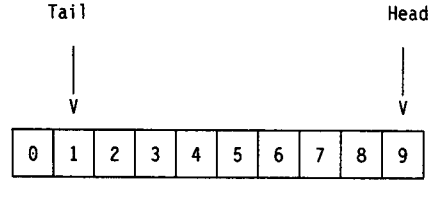
- In PIO mode, if the Transmit-Buffer Length field (offset hex 2C) is set to 0 when the Start routine is called, no action is performed, and the Return Code field is set to hex 0000 (Operation Successfully Completed).

The following is an example of a PIO-mode transmit operation:

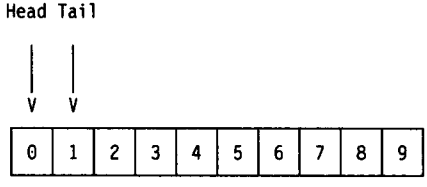
1. The caller initializes the transmit request block and calls the Start routine. The Transmit Head Pointer field and Transmit Tail Pointer field must be set within the range from 0 to the value of the Transmit-Buffer Length field, inclusive. In this example, the buffer length is set to 10, and the transmit buffer is full when the value of the Transmit Head Pointer field is 9. The maximum number of characters that can be sent without the caller changing the value of the Transmit Head Pointer field is 9 (bytes 0 to 8).



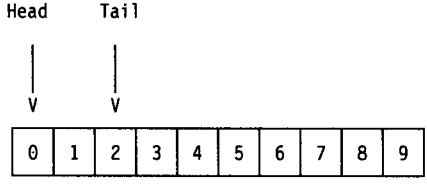
2. The Start routine enables the Transmit interrupt but does not send any data to the serial port.
3. Because the Transmitter Holding register is empty, an interrupt is generated.
4. The Transmit interrupt is called. A character is sent to the serial port, and the value of the Transmit Tail Pointer field is increased by 1. In this example, byte 0 is sent, and the Transmit Tail Pointer field is set to 1.



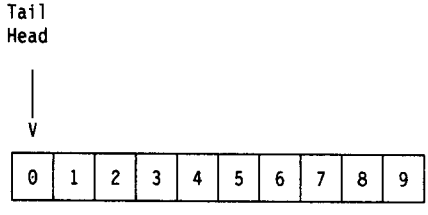
5. Eight bytes are left to be sent (bytes 1 to 8). To increase the number of bytes in the buffer from 8 to 9, the caller can reset the value of the Transmit Head Pointer field to 0 (the beginning of the buffer).



6. Again, when the Transmitter Holding register is empty, an interrupt is generated to the processor. The Interrupt routine is called, and a character is sent to the serial port. The value of the Transmit Tail Pointer field is increased by 1. In this example, byte 1 is sent, and the Transmit Tail Pointer field is set to 2.



7. Assuming that the caller does not change the value of the Transmit Head Pointer field, the process repeats as interrupts occur. Byte 2 is sent, and the Transmit Tail Pointer field is set to 3, and so forth, until byte 9 is sent and the Transmit Tail Pointer field is reset to 0. At this point, the value of the Transmit Tail Pointer field equals the value of the Transmit Head Pointer field, and the caller is informed of the transmit-buffer-empty condition (by the Operation Status field). If the transmit-buffer-empty condition still exists when the next Transmit interrupt is generated, ABIOS disables the Transmit interrupt and sets the Return Code field to hex 0000 (Operation Successfully Completed).



Direct Memory Access (DMA) Transmit

- In a system that supports DMA mode, the transmitter can be loaded through a DMA device. Each time the transmitter is emptied, the DMA device (when properly initialized) loads the transmitter by sending data directly from the buffer memory. This technique requires no processing-unit action and allows data to be sent to serial ports at higher baud rates than those allowed by programmed input/output (PIO) mode.
- To determine whether DMA mode is supported, call the Read Device Parameters function (hex 03). Bit 2 of the FIFO-Mode Status field indicates whether DMA mode is available.
- When bit 0 of the Enhanced Mode field is set to 1, DMA mode is enabled. ABIOS attempts to initialize the DMA device with the values that are in the Transmit Physical Address Buffer Pointer field and the Transmit-Buffer Transfer Length field.

- The following fields must be initialized and passed for a DMA-transmit operation:
  - Transmit-Buffer Offset field (if bit 0 of the Transmit Control field is set to 1)
  - Transmit-Buffer Segment field (if bit 0 of the Transmit Control field is set to 1)
  - Enhanced Mode field
  - Transmit-Buffer Length field (if bit 0 of the Transmit Control field is set to 1)
  - Transmit Head Pointer field (if bit 0 of the Transmit Control field is set to 1)
  - Transmit Tail Pointer field (if bit 0 of the Transmit Control field is set to 1)
  - Transmit Control field (if bit 0 of the Enhanced Mode field is set to 1)
  - Transmit Byte Pacing Rate field (if bit 2 of the Enhanced Mode field is set to 1)
  - Transmit Physical Address Buffer Pointer field
  - Transmit-Buffer Transfer Length field.
- An arbitration level and a DMA channel must be allocated to the DMA device to accomplish a data transfer. The system operator assigns arbitration levels during setup. Asynchronous communication ABIOS automatically allocates DMA channels before any DMA transmit operations are started. If the arbitration level and DMA channel are available, the DMA device is initialized with the parameters that are passed in the request block, and the serial port immediately begins transmitting data.
- If bit 0 of the Transmit Control field is set to 1 and either the arbitration level or the necessary DMA channel is not available, the serial port is initialized to operate in PIO mode, and any DMA-transmit operation will default to PIO mode. If this occurs, ABIOS sets bit 0 of the Enhanced Mode field to 0 to indicate that the request has defaulted to PIO mode, and operation continues as if the request had been made in PIO mode. If bit 0 of the Transmit Control field is set to 0 and the arbitration level or DMA channel is not available, an error is returned, and no action is taken.
- If bit 15 of the Transmit Control field is set to 1, the arbitration level is deallocated when the data transfer is complete. If bit 15 of the Transmit Control field is set to 0, the arbitration level remains allocated to the serial port, and any subsequent transmit requests to that serial port will use that arbitration level. The Cancel function (hex 12) can be used to deallocate the arbitration level.

- If bit 14 of the Enhanced Mode field is set to 1, ABIOS assumes that another process (such as an external resource manager) will allocate and deallocate DMA arbitration-level resources, and it starts the DMA-transmit operation without allocating the arbitration level. When this bit is set to 1, ABIOS does not deallocate the arbitration level when the transmit operation is complete, regardless of whether bit 15 of the Transmit Control field is set to 1. If bit 14 of the Enhanced Mode field is set to 0, ABIOS allocates the arbitration level before starting the DMA-transmit operation.
- In DMA mode, if the Transmit-Buffer Length field (offset hex 2C) is set to 0 when the Start routine is called, a DMA channel is allocated, but no data transfers occur, and the Return Code field is set to hex 0000 (Operation Successfully Completed).

Service-Specific Input

Size	Offset	Description
Word	10H	Reserved
Word	12H	Transmit-buffer offset
Word	14H	Transmit-buffer segment
Word	18H	Enhanced mode Bit 15 - Reserved Bit 14 - Assume transmit arbitration level = 0 - ABIOS does not assume that the arbitration level is assigned = 1 - ABIOS assumes that the arbitration level is assigned Bits 13 to 3 - Reserved Bit 2 - Transmit byte pacing = 0 - Disable = 1 - Enable Bit 1 - Reserved Bit 0 - DMA-transmit mode = 0 - Disable = 1 - Enable
DWord	1AH	Reserved
Word	2CH	Transmit-buffer length, in bytes
Word	2EH	Reserved
Word	30H	Transmit head pointer
Word	32H	Reserved
Word	34H	Transmit tail pointer
Word	36H	Reserved

Size	Offset	Description
Word	44H	Transmit control Bit 15 - Transmit arbitration level deallocation = 0 - Do not deallocate arbitration level when transfer is complete = 1 - Deallocate arbitration level when transfer is complete Bits 14 to 3 - Reserved Bit 2 - Transmit search arbitration level direction = 0 - Increasing, starting with the lowest arbitration level = 1 - Decreasing, starting with the highest arbitration level Bit 1 - Transmit request arbitration level = 0 - No arbitration level was passed = 1 - Arbitration level was passed Bit 0 - Transmit default to PIO = 0 - Return error if DMA is not available = 1 - Use PIO mode if DMA is not available
Byte	4AH	Transmit arbitration level (if dynamic arbitration is supported)
Word	4BH	Reserved
Word	54H	Transmit-byte-pacing rate (maximum value = hex FF)
DWord	5EH	Transmit physical address buffer pointer
DWord	62H	Transmit-buffer transfer length

Service-Specific Output

Size	Offset	Description
Word	16H	Additional operation status Bits 15 to 2 - Reserved Bit 1 - Transmit Byte Pacing interrupt = 0 - Did not occur = 1 - Occurred Bit 0 - Additional interrupts pending = 0 - No additional interrupts pending = 1 - Additional interrupts pending
Word	18H	Enhanced mode Bits 15 to 1 - Reserved Bit 0 - DMA-transmit mode = 0 - Disable = 1 - Enable
Word	34H	Transmit tail pointer
Word	36H	Reserved
Byte	4AH	Transmit arbitration level (if dynamic arbitration is supported)

Size	Offset	Description
Word	4BH	Operation status; return only from Interrupt routine Bits 15 to 8 - Reserved Bit 7 - Transmit buffer empty; Transmitter Holding register empty = 0 - Inactive = 1 - Active Bit 6 - Transmit buffer empty = 0 - Inactive = 1 - Active Bits 5 to 2 - Reserved Bit 1 - Transmit interrupt in progress = 0 - Inactive = 1 - Active Bit 0 - Reserved

0FH—Receive Interrupt

- This function provides several modes for transferring data from the receive buffer.
- Data pointers that describe a buffer area are passed in a request block to ABIOS. The serial port receives characters and moves them into the buffer area. Each time the receive buffer is filled, a signal is sent to request service. Output parameters are returned to inform the program of receive-operation progress.
- There are two modes for receiving data: programmed input/output (PIO) mode and direct memory access (DMA) mode.
- A variety of receive options are available to support PIO mode and DMA mode. These options are controlled through bit settings in the Enhanced Mode field (offset hex 18) and the Receive Control field (offset hex 46). When a DMA-receive operation is requested, options can be selected to assist the program in completing the data transfer. If DMA mode is selected but is not available, an option is available to default to PIO mode or to return an error. Depending on the mode that is selected, logical pointers, physical pointers, or both are required. An option is also available to help service Receive interrupts at very high baud rates without overruns.
- The caller must initialize the Operation Status field to 0 before calling the Start routine. None of the bits in the Operation Status field takes precedence over any other bits in this field.
- When the Reset/Initialize function (hex 05) is executed, if the value of the Number of Bits Per Character field (offset hex 47) is less than 8, the high-order bits of each byte are set to 0 when data is received.

- If bit 5 (stop transmitter on any line error) of the Enhanced Mode field is set to 1, the transmitter is stopped after the shift register is emptied on any line error.
- If bit 4 (received-data status) of the Enhanced Mode field is set to 1, for every byte if data that is received, a byte of status is received in the succeeding byte in memory. The status of this bit is independent of the status of bit 1 (DMA-receive mode) of the Enhanced Mode field. If bit 0 (error/break) of the status byte is set to 1, it remains set for each character until the Reset Error/Break function (hex 19) is called. The received-data status byte has the following format:

  Bit 7 - Data-carrier detect
  Bit 6 - Clear to send
  Bit 5 - Data set ready
  Bit 4 - Break
  Bit 3 - Framing
  Bit 2 - Parity
  Bit 1 - Overrun
  Bit 0 - Error/break

- If bit 3 (interrupt on receiver-character-count = 0) of the Enhanced Mode field is set to 1, the serial port is programmed to interrupt every time the value of the Receive Character Count field (offset hex 52) decreases to 0. When a Receive Character Count interrupt occurs, ABIOS rewrites the value that is in the Receive Character Count field to the serial port. The caller can modify this value across calls to the Receive interrupt.
- Receive Character Count interrupts and Pre-Terminal Count interrupts both use the Receive Character Count field; therefore, only one of these interrupts can be selected at a time. If bit 3 of the Enhanced Mode field and bit 15 (receive pre-terminal count) of the Receive Control field are both set to 1 during a Receive Interrupt function (hex 0F) or a Combined Interrupts function (hex 10), bit 15 of the Receive Control field takes precedence.
- Receive Character Count interrupts and Pre-Terminal Count interrupts are not enabled if transmit byte pacing has been enabled by the Transmit Interrupt function (hex 0E). The functions that are associated with Bit 3 (interrupt on receiver-character-count = 0) and bit 2 (transmit byte pacing) of the Enhanced Mode field are mutually exclusive at the serial port.
- The Null-Stripping Indicator field is independent of the status of bit 1 (DMA-receive mode) of the Enhanced Mode field. In DMA mode, ABIOS requires the use of the Compare Character Match 3 field and the Compare Character Function 3 field. However, when null stripping and DMA mode are both enabled, the values

in these fields are ignored. If null stripping and PIO mode are both enabled, each data character of 0 that is received is not stored in the receive buffer. If a null data byte causes an overrun error, ABIOS discards the null data byte. Bit 12 of the Operation Status field indicates whether this condition occurred.

- The Additional Operation Status field is defined only in systems that support DMA mode. Therefore, do not use the Additional Operation Status field if bit 2 of the FIFO-Mode Status field (offset hex 2B on output) is set to 0.
- The possible values of the Return Code field are hex 0000, 0001, 0005, 0081, 8000, 8001, 8081, 8082, 8083, 8084, 8085, 8086, 8087, and 9000.

Programmed Input/Output (PIO) Receive

- Programmed input/output is a technique in which the receiver in a serial port is emptied through interrupts.
- Each time the receiver is filled, the serial port generates an interrupt, causing the processing unit to service the serial port and empty the receiver.
- In early serial ports, the receiver holds only one character at a time. In more advanced serial ports, the receiver holds 16 bytes of data, allowing more time for the processing unit to service the serial port and allowing more characters to be read each time the serial port is serviced.
- A trigger level can be specified through the Reset/Initialize function (hex 05) or the Set Receive Trigger Level function (hex 16) to cause an interrupt to be generated each time the receiver is filled to a specific level. When characters are received but do not fill the receiver to the trigger level, a Receiver Time-Out interrupt is generated if no more characters are received within four character time frames. This enables the serial port to empty the receiver when no more characters remain to be sent.
- When bit 0 of the Enhanced Mode field is set to 0, PIO mode is enabled.
- ABIOS enables the Receive interrupt, and data is read from the serial port when a Receive interrupt is generated.
- The Receive Head Pointer field points to the first character position that is to be filled. The Receive Tail Pointer field points to the first received character that is to be removed by the caller.
- The values of the Receive Head Pointer field and the Receive Tail Pointer field are relative to the beginning of the receive buffer,

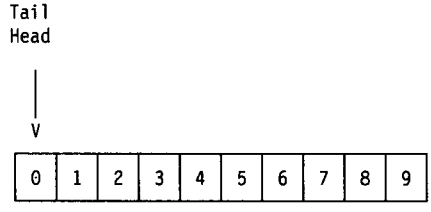
where a value of 0 indicates the first physical byte of the buffer, and the value of the Receive-Buffer Length field minus 1 indicates the last physical byte of the buffer.

- The maximum number of characters that the receive buffer can indicate to be received at any time is the value of the Receive-Buffer Length field minus 1.
- When a Receive interrupt occurs, it increases the value in the Receive Head Pointer field by 1. Bit 0 of the Operation Status field (offset hex 4B on output) is set to 1. Because of the possibility that the buffer will be checked asynchronously by the Receive interrupt, the character is written to the receive buffer before the value of the Receive Head Pointer field is increased.
- As interrupts occur, the value of the Receive Head Pointer field approaches the value of the Receive Tail Pointer field. If the value of the Receive Head Pointer field reaches the end of the receive buffer, the Receive Head Pointer field is set to 0.
- A receive-buffer-full condition occurs when the value of the Receive Head Pointer field equals the value of the Receive Tail Pointer field minus 1. A receive-buffer-full condition occurs also when the value of the Receive Tail Pointer field is 0 and the value of the Receive Head Pointer field equals the value of the Receive Buffer Length field minus 1. Bit 4 of the Operation Status field is set to 1 to indicate a receive-buffer-full condition. If a Receive interrupt occurs while a receive-buffer-full condition exists, the current byte is lost, and bit 5 of the Operation Status field is set to 1. Bits 4 and 5 of the Operation Status field are mutually and exclusively set.
- A receive-buffer-empty condition occurs when the value of the Receive Head Pointer field equals the value of the Receive Tail Pointer field. ABIOS never sets the value of the Receive Head Pointer field equal to the value of the Receive Tail Pointer field; however, the caller can set the value of the Receive Head Pointer field equal to the value of the Receive Tail Pointer field as the receive buffer is emptied and the value of the Receive Tail Pointer is logically increased.
- The values of the Receive-Buffer Segment field, the Receive-Buffer Offset field, the Receive Head Pointer field, and the Receive Tail Pointer field can be altered during calls to the Receive interrupt. Because ABIOS puts data into the receive buffer before changing the value of the Receive Head Pointer field, the caller can remove received data from the receive buffer and logically increase the value of the Receive Tail Pointer field during processing of a Receive interrupt.

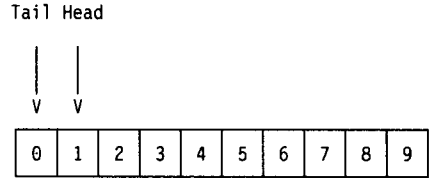
- A Receive interrupt can be terminated only by a call to the Cancel function (hex 12).
- If an error condition occurs at the serial port, ABIOS stores the current data byte in the receive buffer and returns to the caller. Bits 8 to 11 of the Operation Status field indicate the type of error that occurred. If an overrun error occurs, the overrun character is lost, and the data byte contains the valid character that caused the overrun. If a parity error occurs, the data byte contains the character that has the incorrect parity. If a framing error occurs, the data byte contains the character that does not have a valid stop bit. If a break error occurs, the data byte is set to 0.
- When Received Data Status mode is used in PIO mode, FIFO mode must be enabled.
- The following fields must be initialized and passed for a PIO-receive operation:
  - Enhanced Mode field
  - Compare Character Match 1 field (if the Compare Character Function 1 field is not set to 0)
  - Compare Character Function 1 field
  - Compare Character Match 2 field (if the Compare Character Function 2 field is not set to 0)
  - Compare Character Function 2 field
  - Compare Character Match 3 field (if the Compare Character Function 3 field is not set to 0)
  - Compare Character Function 3 field
  - Receive-Buffer Offset field
  - Receive-Buffer Segment field
  - Null-Stripping Indicator field
  - Receive-Buffer Length field
  - Receive Head Pointer field
  - Receive Tail Pointer field
  - Receive Character Count field (if bit 3 of the Enhanced Mode field is set to 1).
- In PIO mode, if the Receive-Buffer Length field (offset hex 38) is set to 0 when the Start routine is called, no action is performed, and the Return Code field is set to hex 0000 (Operation Successfully Completed).

The following is an example of a PIO-mode receive operation:

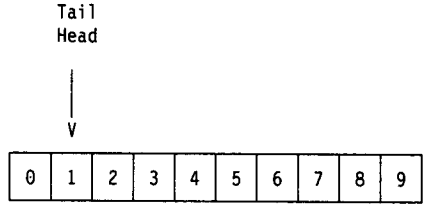
1. The caller initializes the receive request block and calls the Start routine. The Receive Head Pointer field and Receive Tail Pointer field must be set within the range from 0 to the value of the Receive Buffer Length field, inclusive. In this example, the buffer length is set to 10, and the receive buffer is full when the value of the Receive Head Pointer field is logically 1 byte less than the value of the Receive Tail Pointer field. The maximum number of characters that can be received without the caller changing the value of the Receive Tail Pointer field is 9 (bytes 0 to 8).



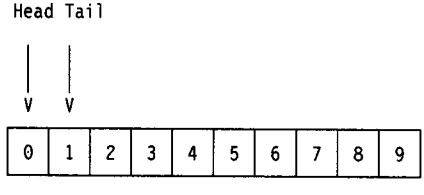
2. The Start routine enables the Receive interrupt but does not read any data.
3. When data is available at the serial port, an interrupt is generated.
4. The Receive interrupt is called. A character is read from the serial port, placed at location 0, and the value of the Receive Head Pointer field is increased by 1.



5. Eight locations are available in the buffer to receive data (bytes 1 to 8). To increase the number of bytes that are available in the buffer to receive data, the caller can read data from the buffer and logically increase the value of the Receive Tail Pointer field up to (but not exceeding) the value of the Receive Head Pointer field. If the value of the Receive Head Pointer field equals the value of the Receive Tail Pointer field, the buffer is empty.



6. Nine locations (bytes 1 to 9) are now available in the receive buffer. When the next Receive interrupt is generated, the Interrupt routine is called, and a character is read from the serial port. The value of the Receive Head Pointer field is increased by 1. Assuming that the caller does not change the value of the Receive Tail Pointer field, the process repeats as interrupts occur. ABIOS receives a character, places the character at location 1, and increases the value of the Receive Head Pointer field by 1. This process is repeated until ABIOS receives a character and places that character at location 9. Because the value of the Receive Head Pointer field, when increased by 1, is outside the buffer range, ABIOS resets the value of the Receive Head Pointer field to 0.



7. The receive buffer is full when the value of the Receive Head Pointer field is 1 less than the value of the Receive Tail Pointer field, or if the value of the Receive Tail Pointer field is 0 and the value of the Receive Head Pointer field is 1 less than the value of the Receive-Buffer Length field. If the receive-buffer-full condition still exists when the Interrupt routine is called, the character that caused the interrupt is lost.

Direct Memory Access (DMA) Receive

- In a system that supports DMA mode, data can be received through a DMA device. The DMA device transfers data from the receiver to memory until the receiver is empty or until the DMA device reaches the terminal count. This technique requires little processing-unit intervention.
- To determine whether DMA mode is supported, call the Read Device Parameters function (hex 03). Bit 2 of the FIFO-Mode Status field indicates whether DMA mode is available.
- When bit 0 of the Enhanced Mode field is set to 1, DMA mode is enabled. ABIOS attempts to initialize the DMA device with the values that are in the Receive Physical Address Buffer Pointer field and the Receive-Buffer Transfer Length field.
- In DMA mode, first-in, first-out (FIFO) mode is automatically enabled and is always active.
- DMA-transmit operations are controlled through the trigger level that is specified through the Reset/Initialize function (hex 05) or the Set Receive Trigger Level function (hex 16). When characters are received but do not fill the receiver to the trigger level, a Receiver Time-Out interrupt is generated if no more characters are received within four character time frames. This enables the serial port to empty the receiver when no more characters remain to be sent.
- The following fields must be initialized and passed for a DMA-receive operation:
  - Enhanced Mode field
  - Compare Character Match 1 field (if the Compare Character Function 1 field is not set to 0)
  - Compare Character Function 1 field
  - Compare Character Match 2 field (if the Compare Character Function 2 field is not set to 0)
  - Compare Character Function 2 field
  - Compare Character Match 3 field (if the Compare Character Function 3 field is not set to 0)
  - Compare Character Function 3 field
  - Receive-Buffer Offset field (if bit 0 of the Receive Control field is set to 1)
  - Receive-Buffer Segment field (if bit 0 of the Receive Control field is set to 1)
  - Null-Stripping Indicator field
  - Receive-Buffer Length field (if bit 0 of the Receive Control field is set to 1)

- Receive Head Pointer field (if bit 0 of the Receive Control field is set to 1)
- Receive Tail Pointer field (if bit 0 of the Receive Control field is set to 1)
- Receive Control field (if bit 1 of the Enhanced Mode field is set to 1)
- Operation Control field
- Receive Character Count field (if bit 3 of the Enhanced Mode field is set to 1).
- Receive-Buffer Switch Threshold field (if bit 15 of the Receive Control field is set to 1)
- Receive Physical Address Buffer Pointer field
- Receive-Buffer Transfer Length field.

- An arbitration level and a DMA channel must be allocated to the DMA device to accomplish a data transfer. The system operator assigns arbitration levels during setup. Asynchronous communication ABIOS automatically allocates DMA channels before any DMA receive operations are started. If the arbitration level and DMA channel are available, the DMA device is initialized with the parameters that are passed in the request block, and the serial port immediately begins receiving data.

- If bit 0 of the Receive Control field is set to 1 and either the arbitration level or the necessary DMA channel is not available, the serial port is initialized to operate in PIO mode, and any DMA-receive operation will default to PIO mode. If this occurs, ABIOS sets bit 0 of the Enhanced Mode field to 0 to indicate that the request has defaulted to PIO mode, and operation continues as if the request had been made in PIO mode. When the receive operation defaults to PIO mode, the receiver receives data in FIFO mode, using the current trigger-level value. If bit 0 of the Transmit Control field is set to 0 and the arbitration level or DMA channel is not available, the Return Code field is set to hex 8081 (Arbitration Level Not Available), and no action is taken.

- If bit 15 of the Receive Control field is set to 1, pre-terminal-count buffer transfer support is enabled. Pre-terminal-count buffer transfer support allows data to be transferred at higher baud rates without overruns, and it provides a way to receive an interrupt, before the terminal count is reached, while a DMA-receive operation is in progress. ABIOS determines the rate at which interrupts occur, and interrupts occur continuously until the end of the buffer is within reach of the number of paragraphs (16 bytes) that are specified in the Receive-Buffer Switch Threshold field. The pre-terminal-count threshold must be less than the receive-buffer length divided by 16.

- ABIOS toggles the buffer by reloading the DMA device with the values that are in the Receive Physical Address Buffer Pointer field and the Receive-Buffer Transfer Length field. Bit 3 of the Operation Status field is set to 1 to indicate that the buffer has been toggled. Any residual data that is received while the Pre-Terminal Count interrupt is being serviced is contained in the original buffer. Any data that is received after the DMA device has been reloaded is contained in the new buffer.
- If a receive request reaches terminal count and the caller has not allowed enough slack in the Receive-Buffer Switch Threshold field, the buffer is toggled, and the Return Code field is set to hex 0001 (Stage on Interrupt).
- When bit 15 of the Receive Control field and bit 3 of the Operation Status field are both set to 1, the caller should update the Receive Physical Address Buffer Pointer field and the Receive-Buffer Transfer Length field immediately after calling the Start routine.
- If bit 5 of the Receive Control field is set to 1, ABIOS is forced to toggle the buffer at the next Receive interrupt. ABIOS initiates another transfer operation by reloading the DMA device with the values that are in the Receive Physical Address Buffer Pointer field and the Receive-Buffer Transfer Length field.
- If a receive request reaches terminal count, ABIOS toggles the buffer.
- The Receive Current Transfer Count field returns the number of transfers that remain before the terminal count will be reached. This field is valid only when all the following conditions exist:
  - The request was successfully started in DMA mode.
  - The request did not default to PIO mode.
  - The arbitration level was successfully allocated.
  - A receive-buffer toggle occurred.
- If bit 5 (stop transmitter on any line error) of the Enhanced Mode field is set to 1, the transmitter is stopped after the shift register is emptied on any received line error. In DMA mode, ABIOS programs the serial port to interrupt on line-status errors. The Operation Status field returns the results of these interrupts. Note that a Transmit Interrupt function (hex 0E) must be outstanding.
- If bit 15 of the Enhanced Mode field is set to 1, ABIOS assumes that another process (such as an external resource manager) will allocate and deallocate DMA arbitration-level resources, and it starts the DMA-receive operation without allocating the

arbitration level. When this bit is set to 1, ABIOS does not deallocate the arbitration level when the receive operation is complete. If bit 14 of the Enhanced Mode field is set to 0, ABIOS allocates the arbitration level before starting the DMA-receive operation.

- In DMA mode, if the Receive-Buffer Length field (offset hex 38) is set to 0 when the Start routine is called, a DMA channel is allocated, but no data transfers occur, and the Return Code field is set to hex 0000 (Operation Successfully Completed).

Service-Specific Input

Size	Offset	Description
Word	18H	Enhanced mode Bit 15 - Assume receive arbitration level = 0 - ABIOS does not assume that the arbitration level is assigned = 1 - ABIOS assumes that the arbitration level is assigned Bits 14 to 6 - Reserved Bit 5 - Stop transmitter on any line error = 0 - Disable = 1 - Enable Bit 4 - Received-data status = 0 - Disable = 1 - Enable Bit 3 - Interrupt on receiver-character-count = 0 = 0 - Disable = 1 - Enable Bit 2 - Reserved Bit 1 - DMA-receive mode = 0 - Disable = 1 - Enable Bit 0 - Reserved
Byte	1AH	Compare character match 1

Size	Offset	Description
Byte	1BH	Compare character function 1 Bits 7 to 4 - Reserved Bit 3 - Start transmitter on a character match = 0 - Transmitter not started on a character match = 1 - Transmitter started on a character match (the transmitter will not start if bit 2 is set to 1) Bit 2 - Stop transmitter on a character match = 0 - Transmitter not stopped on a character match = 1 - Transmitter stopped on a character match (if the transmitter is stopped with a character match, a call to the Start Transmission function (hex 18) can be used to restart the transmitter) Bit 1 - Delete character on a character match = 0 - Character not deleted on a character match = 1 - Character deleted on a character match Bit 0 - Interrupt on character match = 0 - Interrupt not enabled on a character match = 1 - Interrupt enabled on a character match (the interrupt occurs as soon as there is a match with a received data character)
Byte	1CH	Compare character match 2
Byte	1DH	Compare character function 2 (see the description of the Compare Character Function 1 field)
Byte	1EH	Compare character match 3
Byte	1FH	Compare character function 3 (see the description of the Compare Character Function 1 field)
Word	20H	Reserved
Word	22H	Receive-buffer offset
Word	24H	Receive-buffer segment
Word	28H	Null-stripping indicator = 00H - Disable = 01H - Enable = 02H to FFH - Reserved
Byte	29H	Receive-buffer switch threshold
Word	38H	Receive-buffer length, in bytes
Word	3AH	Reserved
Word	3CH	Receive head pointer
Word	3EH	Reserved
Word	40H	Receive tail pointer
Word	42H	Reserved

Size	Offset	Description
Word	46H	Receive control Bit 15 - Receive pre-terminal count = 0 - Disable = 1 - Enable Bits 14 to 6 - Reserved Bit 5 - Buffer toggle = 0 - Do not force toggle = 1 - Force toggle Bits 4, 3 - Reserved Bit 2 - Receive search arbitration level direction = 0 - Increasing, starting with the lowest arbitration level = 1 - Decreasing, starting with the highest arbitration level Bit 1 - Receive request arbitration level = 0 - No arbitration level was passed = 1 - Arbitration level was passed Bit 0 - Receive default to PIO = 0 - Return error if DMA is not available = 1 - Use PIO mode if DMA is not available
Byte	49H	Receive arbitration level (if dynamic arbitration is supported)
Word	4BH	Reserved
Word	52H	Receive character count (maximum value = hex FF)
DWord	66H	Receive physical address buffer pointer
DWord	6AH	Receive-buffer transfer length

Service-Specific Output

Size	Offset	Description
Word	16H	Additional operation status Bits 15 to 4 - Reserved Bit 3 - Receive Line-Status Error interrupt = 0 - Did not occur = 1 - Occurred Bit 2 - Receive Time-Out interrupt = 0 - Did not occur = 1 - Occurred Bit 1 - Reserved Bit 0 - Additional interrupts pending = 0 - No additional interrupts pending = 1 - Additional interrupts pending
Word	18H	Enhanced mode Bits 15 to 2 - Reserved Bit 1 - DMA-receive mode = 0 - Disabled = 1 - Enabled Bit 0 - Reserved
Word	3CH	Receive head pointer
Word	3EH	Reserved

Size	Offset	Description
Word	46H	Receive control Bits 15 to 6 - Reserved Bit 5 - Buffer toggle = 0 - Toggle not forced = 1 - Toggle forced Bits 4, 3 - Reserved Bit 2 - Receive search arbitration level direction = 0 - Increasing, starting with the lowest arbitration level = 1 - Decreasing, starting with the highest arbitration level Bit 1 - Receive request arbitration level = 0 - No arbitration level was passed = 1 - Arbitration level was passed Bit 0 - Reserved
Byte	49H	Receive arbitration level (if dynamic arbitration is supported)
Word	4BH	Operation status; return only from Interrupt routine Bit 15 - Compare Character Match 3 interrupt = 0 - Did not occur = 1 - Occurred Bit 14 - Compare Character Match 2 interrupt = 0 - Did not occur = 1 - Occurred Bit 13 - Compare Character Match 1 interrupt = 0 - Did not occur = 1 - Occurred Bit 12 - Overrun error with null data byte found = 0 - Inactive = 1 - Active only if null data was found and discarded Bit 11 - Break detected = 0 - Inactive = 1 - Active Bit 10 - Framing error = 0 - Inactive = 1 - Active Bit 9 - Parity error = 0 - Inactive = 1 - Active Bit 8 - Overrun error = 0 - Inactive = 1 - Active Bits 7, 6 - Reserved Bit 5 - Receive buffer full; data discarded = 0 - Inactive = 1 - Active Bit 4 - Receive buffer full = 0 - Inactive = 1 - Active Bit 3 - Buffer toggle = 0 - Did not occur = 1 - Occurred

Size	Offset	Description
		Bit 2 - Receive Character Count/ Pre-Terminal-Count Buffer Transfer interrupt = 0 - Did not occur = 1 - Occurred
		Bit 1 - Reserved
		Bit 0 - Receive interrupt in progress = 0 - Inactive = 1 - Active
DWord	5AH	Receive current transfer count

10H—Combined (Transmit and Receive) Interrupts

- This function is used to perform a transmit operation and a receive operation in a single request. It can be used also to perform a receive-only operation.
- PIO-transmit and PIO-receive operations are defined as requests that are initiated in PIO mode or DMA requests that default to PIO mode. DMA-transmit and DMA-receive operations are defined as requests that are initiated in DMA mode and do not default to PIO mode. See "Programmed Input/Output (PIO) Transmit" on page 6-ID06-16 for an explanation of PIO-transmit operations. See "Programmed Input/Output (PIO) Receive" on page 6-ID06-27 for an explanation of PIO-receive operations. See "Direct Memory Access (DMA) Transmit" on page 6-ID06-21 for an explanation of DMA-transmit operations. See "Direct Memory Access (DMA) Receive" on page 6-ID06-32 for an explanation of DMA-receive operations.
- A receive operation can be initiated by the Start routine with only the receive parameters set (the Transmit-Buffer Length field is set to 0 for a PIO-receive operation, and the Transmit-Buffer Transfer Length field is set to 0 for a DMA-receive operation). If this function was initially called with only the receive operation enabled, the transmit operation can be enabled on a subsequent call with the Transmit-Buffer Length field (for PIO mode) or the Transmit-Buffer Transfer Length field (for DMA mode) set to a nonzero value. Both fields must be set to nonzero values when bit 0 (transmit default to PIO) of the Transmit Control field is set to 1.
- A transmit operation cannot be initiated on the first call to the Start routine.
- When the Start routine is called to enable a receive operation, the Return Code field must be set to hex FFFF (Return Code Not Valid). However, if the Start routine is called for a second time to enable the transmit operation, the previously-stored value of the Return Code field must not be changed. On return from the

second call to the Start routine, the value of the Return Code field is undefined and should be ignored.

- During receive operations in PIO mode, the transmit operation is never disabled; no test is performed to determine whether the value of the Transmit Head Pointer field is equal to the value of the Transmit Tail Pointer field.
- To optimize performance after a receive operation is serviced, ABIOS checks for an interrupt-pending condition. If a transmit interrupt is pending, the interrupt is serviced, and the Return Code field is set to hex 0009 (Attention, Stage on Interrupt). If a second receive interrupt is pending, ABIOS does not service the pending interrupt, and it returns to the caller.
- If the Return Code field is set to hex 0009 (Attention, Stage on Interrupt), the Operation Status of Previous Interrupt field contains the status that was returned when the first interrupt was serviced. The Return Code of Previous Interrupt field contains the return-code value that was returned from the first interrupt. The Operation Status field and the Return Code field contain the values from the second interrupt. The Operation Status of Previous Interrupt field and the Return Code of Previous Interrupt field must be initialized to 0 by the caller before the Start routine is called.
- In a combined operation, if the receive operation is completed, the transmit operation is not allowed to start, and the Return Code field is set to hex C020 (No Receive Active in Combined Interrupt). However, if the transmit operation is in DMA mode and the transmit-buffer length is 0, the transmit operation is allowed to start, to allocate the transmit arbitration level.
- When this function is called by the Start routine for the first time, if the value of the Transmit-Buffer Length field (in PIO mode) or the Transmit-Buffer Transfer Length field (in DMA mode) is 0, and the value of the Receive-Buffer Length field (in PIO mode) or the Receive-Buffer Transfer Length field (in DMA mode) is nonzero, this function acts as a Receive Interrupt function (hex 0F). When this function is called by the Start routine for the second time, if the value of the Transmit-Buffer Length field (in PIO mode) or the Transmit-Buffer Transfer Length field (in DMA mode) is 0, no action is performed, and the Return Code field is set to hex 0001 (Stage on Interrupt).
- If a Transmit interrupt and a Receive interrupt are both enabled and bit 3 (interrupt on receiver-character-count=0) and bit 2 (transmit byte pacing) of the Enhanced Mode field are set to 1, bit 3 takes precedence over bit 2. The functions that are associated

with bit 3 and bit 2 are mutually exclusive at the serial port. In the same way, bit 15 (receive pre-terminal count) of the Receive Control field takes precedence over bit 2 (transmit byte pacing) of the Enhanced Mode field.

- If bit 0 (transmit default to PIO) of the Transmit Control field and bit 0 (receive default to PIO) of the Receive Control field are both set to 0 and the necessary DMA channels are not available, the Return Code field is set to hex 8086 (No DMA Channel Available), and neither transfer is initiated.
- The possible values of the Return Code field are hex 0000, 0001, 0005, 0009, 0081, 8000, 8001, 8081, 8082, 8083, 8084, 8085, 8086, 8087, 9000, and C020.

Service-Specific Input

Size	Offset	Description
Word	10H	Reserved
Word	12H	Transmit-buffer offset
Word	14H	Transmit-buffer segment
Word	18H	Enhanced mode Bit 15 - Assume receive arbitration level = 0 - ABIOS does not assume that the arbitration level is assigned = 1 - ABIOS assumes that the arbitration level is assigned Bit 14 - Assume transmit arbitration level = 0 - ABIOS does not assume that the arbitration level is assigned = 1 - ABIOS assumes that the arbitration level is assigned Bits 13 to 6 - Reserved Bit 5 - Stop transmitter on any line error = 0 - Disable = 1 - Enable Bit 4 - Received-data status = 0 - Disable = 1 - Enable Bit 3 - Interrupt on receiver-character-count = 0 = 0 - Disable = 1 - Enable Bit 2 - Transmit byte pacing = 0 - Disable = 1 - Enable Bit 1 - DMA-receive mode = 0 - Disable = 1 - Enable Bit 0 - DMA-transmit mode = 0 - Disable = 1 - Enable
Byte	1AH	Compare character match 1

Size	Offset	Description
Byte	1BH	Compare character function 1 Bits 7 to 4 - Reserved Bit 3 - Start transmitter on a character match = 0 - Transmitter not started on a character match = 1 - Transmitter started on a character match (the transmitter will not start if bit 2 is set to 1) Bit 2 - Stop transmitter on a character match = 0 - Transmitter not stopped on a character match = 1 - Transmitter stopped on a character match (if the transmitter is stopped with a character match, a call to the Start Transmission function (hex 18) can be used to restart the transmitter) Bit 1 - Delete character on a character match = 0 - Character not deleted on a character match = 1 - Character deleted on a character match Bit 0 - Interrupt on character match = 0 - Interrupt not enabled on a character match = 1 - Interrupt enabled on a character match (the interrupt occurs as soon as there is a match with a received data character)
Byte	1CH	Compare character match 2
Byte	1DH	Compare character function 2 (see the description of the Compare Character Function 1 field)
Byte	1EH	Compare character match 3
Byte	1FH	Compare character function 3 (see the description of the Compare Character Function 1 field)
Word	20H	Reserved
Word	22H	Receive-buffer offset
Word	24H	Receive-buffer segment
Byte	28H	Null-stripping indicator = 00H - Disabled = 01H - Enabled = 02H to FFH - Reserved
Byte	29H	Receive-buffer switch threshold/ receive demand-allocation threshold
Word	2CH	Transmit-buffer length, in bytes
Word	2EH	Reserved
Word	30H	Transmit head pointer
Word	32H	Reserved
Word	34H	Transmit tail pointer
Word	36H	Reserved
Word	38H	Receive-buffer length, in bytes
Word	3AH	Reserved
Word	3CH	Receive head pointer
Word	3EH	Reserved
Word	40H	Receive tail pointer
Word	42H	Reserved

Size	Offset	Description
Word	44H	Transmit control Bit 15 - Transmit arbitration level deallocation = 0 - Do not deallocate arbitration level when transfer is complete = 1 - Deallocate arbitration level when transfer is complete Bits 14 to 3 - Reserved Bit 2 - Transmit search arbitration level direction = 0 - Increasing, starting with the lowest arbitration level = 1 - Decreasing, starting with the highest arbitration level Bit 1 - Transmit request arbitration level = 0 - No arbitration level was passed = 1 - Arbitration level was passed Bit 0 - Transmit default to PIO = 0 - Return error if DMA is not available = 1 - Use PIO mode if DMA is not available
Word	46H	Receive control Bit 15 - Receive pre-terminal count = 0 - Disable = 1 - Enable Bits 14 to 6 - Reserved Bit 5 - Buffer toggle = 0 - Do not force toggle = 1 - Force toggle Bits 4 to 1 - Reserved Bit 0 - Receive default to PIO = 0 - Return error if DMA is not available = 1 - Use PIO mode if DMA is not available
Byte	49H	Receive arbitration level (if dynamic arbitration is supported)
Byte	4AH	Transmit arbitration level (if dynamic arbitration is supported)
Word	4BH	Operation control (reserved)
Word	52H	Receive character count (maximum value = hex FF)
Word	54H	Transmit-byte-pacing rate (maximum value = hex FF)
DWord	5EH	Transmit physical address buffer pointer
DWord	62H	Transmit-buffer transfer length
DWord	66H	Receive physical address buffer pointer
DWord	6AH	Receive-buffer transfer length

Service-Specific Output

Size	Offset	Description
Word	16H	Additional operation status Bits 15 to 4 - Reserved Bit 3 - Receive Line-Status Error interrupt = 0 - Did not occur = 1 - Occurred Bit 2 - Receive Time-Out interrupt = 0 - Did not occur = 1 - Occurred Bit 1 - Transmit Byte Pacing interrupt = 0 - Did not occur = 1 - Occurred Bit 0 - Additional interrupts pending = 0 - No additional interrupts pending = 1 - Additional interrupts pending
Word	18H	Enhanced mode Bits 15 to 2 - Reserved Bit 1 - DMA-receive mode = 0 - Disabled = 1 - Enabled Bit 0 - DMA-transmit mode = 0 - Disabled = 1 - Enabled
Word	34H	Transmit tail pointer
Word	36H	Reserved
Word	3CH	Receive head pointer
Word	3EH	Reserved
Word	46H	Receive control Bits 15 to 6 - Reserved Bit 5 - Buffer toggle = 0 - Toggle not forced = 1 - Toggle forced Bits 4 to 0 - Reserved
Byte	49H	Receive arbitration level (if dynamic arbitration is supported)
Byte	4AH	Transmit arbitration level (if dynamic arbitration is supported)
Word	4BH	Operation status; return only from Interrupt routine Bit 15 - Compare Character Match 3 interrupt = 0 - Did not occur = 1 - Occurred Bit 14 - Compare Character Match 2 interrupt = 0 - Did not occur = 1 - Occurred Bit 13 - Compare Character Match 1 interrupt = 0 - Did not occur = 1 - Occurred Bit 12 - Overrun error with null data byte found = 0 - Inactive = 1 - Active only if null data was found and discarded

Size	Offset	Description
		Bit 11 - Break detected = 0 - Inactive = 1 - Active Bit 10 - Framing error = 0 - Inactive = 1 - Active Bit 9 - Parity error = 0 - Inactive = 1 - Active Bit 8 - Overrun error = 0 - Inactive = 1 - Active Bit 7 - Transmit buffer empty; Transmitter Holding register empty = 0 - Inactive = 1 - Active Bit 6 - Transmit buffer empty = 0 - Inactive = 1 - Active Bit 5 - Receive buffer full; data discarded = 0 - Inactive = 1 - Active Bit 4 - Receive buffer full = 0 - Inactive = 1 - Active Bit 3 - Buffer toggle = 0 - Did not occur = 1 - Occurred Bit 2 - Receive Character Count/ Pre-Terminal-Count Buffer Transfer interrupt = 0 - Did not occur = 1 - Occurred Bit 1 - Transmit interrupt in progress = 0 - Inactive = 1 - Active Bit 0 - Receive interrupt in progress = 0 - Inactive = 1 - Active
Word	4DH	Operation status of previous interrupt
Word	4FH	Return code of previous interrupt
DWord	56H	Reserved
DWord	5AH	Receive current transfer count

11H—Modem Status Interrupt

- This function enables the Modem Status interrupt. The Modem Status field is returned to the caller on exit from the Start and Interrupt routines.
- In PIO mode, if the modem-status-interrupt request block is processed before the receive or transmit request block at interrupt time, the ABIOS routine detects a change in the modem status, even if the higher-priority interrupts in the Interrupt Identification register are still pending. This enables an interrupt handler to detect a change in modem status before any data is received or transmitted.
- The 'high-data-rate signal selector' signal from dual-range asynchronous data-communication equipment is used to select between two ranges of data-signaling rates. When bit 15 of the Enhanced Function Control field is set to 1, an interrupt occurs when the 'high-data-rate signal selector' signal changes state. The current state of the 'high-data-rate signal selector' signal is returned in bit 8 of the Enhanced Function Status field.
- The possible values of the Return Code field are hex 0001, 0005, 0081, 8000, and 8001.

Service-Specific Input

Size	Offset	Description
Word	18H	Enhanced function control Bit 15 - 'High data rate' signal interrupt = 0 - Disable = 1 - Enable Bits 14 to 0 - Reserved
DWord	1AH	Reserved

Service-Specific Output

Size	Offset	Description
Word	18H	Enhanced function status Bits 15 to 9 - Reserved Bit 8 - 'High data rate' signal status = 0 - 'High data rate' signal is inactive = 1 - 'High data rate' signal is active Bits 7 to 0 - Reserved

Size	Offset	Description
Byte	4AH	Modem status Bit 7 - Data-carrier detect = 0 - Inactive = 1 - Active Bit 6 - Ring indicator = 0 - Inactive = 1 - Active Bit 5 - Data set ready = 0 - Inactive = 1 - Active Bit 4 - Clear to send = 0 - Inactive = 1 - Active Bit 3 - Delta data-carrier detect = 0 - Inactive = 1 - Active Bit 2 - Trailing-edge ring indicator = 0 - Inactive = 1 - Active Bit 1 - Delta data set ready = 0 - Inactive = 1 - Active Bit 0 - Delta clear to send = 0 - Inactive = 1 - Active

12H—Cancel

- This function cancels a requested interrupt operation. Any combinations of interrupts can be canceled. Associated interrupts are disabled on return from this function. If an outstanding request is associated with the interrupt type that is canceled, that request block is considered canceled and must be appropriately deallocated.
- DMA transmit and receive operations can be canceled individually within a Combined Interrupt function (hex 10).
- 'High Data Rate' Signal interrupts are canceled when the Modem Status Interrupt function (hex 11) is canceled.
- If a Receive interrupt and a Transmit interrupt are both enabled in a Combined Interrupts function (hex 10) and the Cancel function is called to cancel both interrupts, the combined request block is considered canceled.
- If a Receive interrupt and a Transmit interrupt are both enabled in a Combined Interrupts function (hex 10) and the Cancel function is called to cancel either the Transmit interrupt or the Receive interrupt, but not both, the request block is considered active.
- If a Receive interrupt and a Transmit interrupt are both enabled in a Combined Interrupts function (hex 10) and the Cancel function

is called to cancel the Transmit interrupt, a call to the Start routine to reenable the Transmit interrupt is permitted.

- If a Receive interrupt and a Transmit interrupt are both enabled in a Combined Interrupts function (hex 10) and the Receive interrupt has been cancelled through the Receive interrupt, to reenable the Receive interrupt through the Combined Interrupts function (hex 10), the caller must first cancel the Transmit interrupt, then call the Combined Interrupts function (hex 10) with the Receive interrupt enabled.
- If only the Receive interrupt is enabled in a Combined Interrupts function (hex 10) and the Cancel function is called to cancel the Receive interrupt, the combined request block is considered canceled.
- If the Receive Interrupt function is canceled, the Receive interrupt and the Line Status interrupt are disabled. Canceling the Receive interrupt also disables the Interrupt On Receiver-Character-Count = 0 interrupt and the three Compare Character functions.
- If the Transmit Interrupt function is canceled, the Transmit interrupt and the Transmit Byte Pacing interrupt are disabled. Canceling the Transmit Interrupt function also cancels the Stop Transmission and Send Data function (hex 17).
- The Transmit Current Transfer Count field and the Receive Current Transfer Count field return the number of transfers that remain before DMA terminal count will be reached.
- See page 6-ID06-23 for an explanation of bit 14 (assume transmit arbitration level) of the Enhanced Mode field. See page 6-ID06-34 for an explanation of bit 15 (assume receive arbitration level) of the Enhanced Mode field.
- The possible values of the Return Code field are hex 0000, 8000, 8001, 8081, 8082, 8083, 8084, 8085, 8086, and 8087.

Service-Specific Input

Size	Offset	Description
Word	18H	Enhanced mode Bit 15 - Assume receive arbitration level = 0 - ABIOS does not assume that the arbitration level is assigned = 1 - ABIOS assumes that the arbitration level is assigned Bit 14 - Assume transmit arbitration level = 0 - ABIOS does not assume that the arbitration level is assigned = 1 - ABIOS assumes that the arbitration level is assigned Bits 13 to 0 - Reserved
Byte	51H	Interrupt type to be canceled Bit 7 - Receive arbitration level deallocation = 0 - Do not deallocate = 1 - Deallocate Bit 6 - Transmit arbitration level deallocation = 0 - Do not deallocate = 1 - Deallocate Bit 5 - Modem status = 0 - Do not disable = 1 - Disable Bit 4 - Reserved Bit 3 - Transmit = 0 - Do not disable = 1 - Disable Bit 2 - Receive = 0 - Do not disable = 1 - Disable Bit 1 - Stop transmission and send data = 0 - Do not disable = 1 - Disable Bit 0 - 'High data rate' signal = 0 - Do not cancel request = 1 - Cancel request

Service-Specific Output

Size	Offset	Description
DWord	56H	Transmit current transfer count
DWord	5AH	Receive current transfer count

13H—Return Line Status

- This function returns the line status.
- The value of the Return Code field is hex 0000.

Service-Specific Input

Size	Offset	Description
None		

Service-Specific Output

Size	Offset	Description
Byte	49H	Line status Bit 7 - Error in receiver FIFO = 0 - No error in FIFO = 1 - Error in receiver FIFO Bit 6 - Transmitter empty = 0 - Inactive = 1 - Active Bit 5 - Transmitter holding register empty = 0 - Inactive = 1 - Active Bit 4 - Break interrupt = 0 - Inactive = 1 - Active Bit 3 - Framing error = 0 - Inactive = 1 - Active Bit 2 - Parity error = 0 - Inactive = 1 - Active Bit 1 - Overrun error = 0 - Inactive = 1 - Active Bit 0 - Data ready = 0 - Inactive = 1 - Active

14H—Return Modem Status

- This function returns the modem status. If this function is called and the Modem Status interrupt is enabled, the Return Code field is set to hex 8000 (Device Busy).
- The possible values of the Return Code field are hex 0000 and 8000.

Service-Specific Input

Size	Offset	Description
None		

Service-Specific Output

Size	Offset	Description
Word	18H	Enhanced function status Bits 15 to 9 - Reserved Bit 8 - 'High data rate' signal status = 0 - 'High data rate' signal is inactive = 1 - 'High data rate' signal is active Bits 7 to 0 - Reserved

Size	Offset	Description
Byte	4AH	Modem status Bit 7 - Data-carrier detect = 0 - Inactive = 1 - Active Bit 6 - Ring indicator = 0 - Inactive = 1 - Active Bit 5 - Data set ready = 0 - Inactive = 1 - Active Bit 4 - Clear to send = 0 - Inactive = 1 - Active Bit 3 - Delta data-carrier detect = 0 - Inactive = 1 - Active Bit 2 - Trailing-edge ring indicator = 0 - Inactive = 1 - Active Bit 1 - Delta data set ready = 0 - Inactive = 1 - Active Bit 0 - Delta clear to send = 0 - Inactive = 1 - Active

15H—Reserved

16H—Set Receive Trigger Level

- This function sets the first-in, first-out (FIFO) trigger level and enables the FIFO mode for the serial port, according to the input parameters.
- The possible values of the Return Code field are hex 0000 and 8001.

Service-Specific Input

Size	Offset	Description
Word	18H	Enhanced function control Bits 15 to 4 - Reserved Bit 3 - Reset transmitter FIFO = 0 - Take no action = 1 - Discard all characters in transmitter FIFO Bit 2 - Reset receiver FIFO = 0 - Take no action = 1 - Discard all characters in receiver FIFO Bits 1, 0 - FIFO enable/disable control = 00 - No action taken = 10 - Reserved = 01 - Disable FIFO = 11 - Enable FIFO

Size	Offset	Description
Byte	28H	Receive trigger level = 00H - 1 byte = 01H - 4 bytes = 02H - 8 bytes = 03H - 14 bytes = 04H to FFH - Reserved

Service-Specific Output

Size	Offset	Description
None		

17H—Stop Transmission and Send Data

- This function empties the shift register and stops transmitting, regardless of the number of characters that remain in the FIFO buffer. When the transmitter is stopped, a string of characters, a single character, or no character is transmitted, as specified in bits 2, 1, and 0 of the Function Control field. The string of characters to which the Logical String Pointer field points is loaded into the shift register, and each character is transmitted individually.
- The Transmitter Holding Register Empty interrupt is enabled to signal to the system when the single-character transmission is complete. Each time the interrupt occurs, a call to the Interrupt routine must be executed with the Stop Transmission and Send Data function request block as a parameter. This process is repeated for each character until the value of the Character Count to Be Transmitted field is exhausted. If multiple characters are sent, multiple interrupts occur. Calling the Interrupt routine to pass the suspended Transmit interrupt function (hex 0E) request block or the suspended Combined Interrupts function (hex 10) request block is not necessary.
- Before the Start Transmission (hex 18) function is issued to restart the DMA transfer, each character to which the Logical String Pointer field points must have been transmitted, and the interrupt must have been serviced.
- If this function is executed with the Function Control field set to hex 00, the transmitter is stopped, no characters are sent to the Transmitter Holding Register, and the Return Code field is set to hex 0000 (Operation Successfully Completed).
- If this function is executed with the Function Control field set to hex 01, the transmitter is stopped, ABIOS writes a single character to the Transmitter Holding register, and the Return Code field is set to hex 0001 (Stage on Interrupt). When the

Transmitter Holding register is emptied, the interrupt occurs. A call to the Interrupt routine causes the character to be written to the Transmitter Holding register, and the Return Code field is set to hex 0000 (Operation Successfully Completed).

- If this function is executed with the Function Control field set to hex 02, the transmitter is stopped, ABIOS sends a string of characters to the Transmitter Holding register, and the Return Code field is set to hex 0001 (Stage on Interrupt) after the first character and any subsequent characters are written to the Transmitter Holding register. When the entire character string has been sent, the Return Code field is set to hex 0000 (Operation Successfully Completed).
- This function can be stopped by setting bit 1 of the Interrupt Type to Be Canceled field in the Cancel function (hex 12) to 1 and executing the Cancel function.
- A transmit operation that is interrupted with this function should be restarted through the Start Transmission function (hex 18).
- The possible values of the Return Code field are hex 0000, 0001, 0005, 0081, 8000 and 8001.

Service-Specific Input

Size	Offset	Description
DWord	12H	Logical string pointer
Word	18H	Reserved
Byte	1AH	Character to be transmitted
Byte	1BH	Function control = 00H - Stop transmitter = 01H - Stop transmitter and send a character = 02H - Stop transmitter and send a string
Word	2CH	Character count to be transmitted
Word	2EH	Reserved

Service-Specific Output

Size	Offset	Description
Word	2CH	Number of characters that were transmitted
Word	2EH	Reserved

18H—Start Transmission

- This function is used to restart or continue a transmit request that has been interrupted by the Stop Transmission and Send Data function (hex 17) or by a stop-transmitter-on-a-character-match condition in the Receive Interrupt function (hex 0F) or in the Combined Interrupts function (hex 10).
- No attempt is made to verify that a transmit request is outstanding.
- If this function is called while a request to the Stop Transmission and Send Data function is outstanding, the Return Code field is set to hex 8000 (Device Busy, Request Refused).
- The possible values of the Return Code field are hex 0000, 8000, and 8001.

Service-Specific Input

Size	Offset	Description
Word	18H	Reserved

Service-Specific Output

Size	Offset	Description
None		

19H—Reset Error/Break

- This function resets bit 0 (error/break) in the received-data status byte, which is optionally stored with received data. The Reset command affects the next status byte, which is stored in the FIFO buffer.
- A program should check each byte in which the error/break bit is set to 1 and issue a Reset command each time an error is found. This procedure will minimize the overhead that is associated with error detection.
- This function can be used with the Receive Interrupt function (hex 0F) or with the Combined Interrupts function (hex 10) to determine error status with greater accuracy. When bit 4 (received-data status) of the Enhanced Mode function is set to 1 in the Receive Interrupt function or the Combined Interrupts function, a byte of status is received with each character. If a line-status error occurs, the error/break bit of the received-data status byte of each received character is set to 1 until the Reset Error/Break function is called.

- The possible values of the Return Code field are hex 0000 and 8001.

Service-Specific Input

Size	Offset	Description
Word	18H	Reserved

Service-Specific Output

Size	Offset	Description
None		

1AH—Return Current Transfer Counts

- This function returns the current transfer counts for transmit operations and receive operations.
- On return, the Return Code field (offset hex 0C) contains the status of the receive transfer counts. The Return Code of Transmit Transfer Count field (offset hex 4F) contains the status of the transmit transfer counts. Both return-code fields should be checked before the Transmit Current Transfer Count field (offset hex 56) and the Receive Current Transfer Count field (offset hex 5A) are read.
- If this function is called with no transmit or receive requests outstanding, the Transmit Current Transfer Count field and the Receive Current Transfer Count field are set to 0 on return.
- The possible values of the Return Code fields are hex 0000, 8001, 8082, and 8083.

Service-Specific Input

Size	Offset	Description
Word	18H	Reserved

Service-Specific Output

Size	Offset	Description
Word	4FH	Return code of transmit transfer count
DWord	56H	Transmit current transfer count
DWord	5AH	Receive current transfer count

1BH—Set Compare Character Match and Function

- This function sets the three Compare Character registers and their associated Compare Character functions.
- A Receive Interrupt function (hex 0F) or a Combined Interrupts function (hex 10) with the receive operation enabled should be outstanding before this function is called. If neither of these functions is outstanding, this function writes the Character Compare registers and Character Compare functions to the hardware.
- The three Compare Character Match fields enable programming of receive-match characters in receive operations.
- If a character that is received from the controller matches the value in one of the three enabled Compare Character Match fields, an action is performed as defined by the three Compare Character Function fields.
- The data in a Compare Character Match field is an 8-bit match character. If the word length is less than 8 bits, the match character must be right justified, and any unused bits must be set to 0.
- If null stripping is enabled, the Compare Character Match 3 field is unavailable.
- The three Compare Character Function fields correspond to the three Compare Character Match fields. They enable programming of the controller to start the transmitter, stop the transmitter, delete a match character from an incoming data stream, or to interrupt.
- Multiple Compare Character Match functions per Compare Character function are supported.
- If bit 3 (start transmitter) and bit 2 (stop transmitter) of the Compare Character Function field are both set to 1, bit 2 takes precedence.
- If bit 0 (interrupt) is set to 1, the interrupt occurs as soon as a match with a received data character is detected.
- To disable a Compare Character Match field, set the corresponding Compare Character Function field to 0 before calling the Set Compare Character Match and Function function (hex 1B).
- The possible values of the Return Code field are hex 0000 and 8001.

Service-Specific Input

Size	Offset	Description
Word	18H	Reserved
Byte	1AH	Compare character match 1
Byte	1BH	Compare character function 1 Bits 7 to 4 - Reserved Bit 3 - Start transmitter on a character match = 0 - Transmitter not started on a character match = 1 - Transmitter started on a character match (the transmitter will not start if bit 2 is set to 1) Bit 2 - Stop transmitter on a character match = 0 - Transmitter not stopped on a character match = 1 - Transmitter stopped on a character match (if the transmitter is stopped with a character match, a call to the Start Transmission function (hex 18) can be used to restart the transmitter) Bit 1 - Delete character on a character match = 0 - Character not deleted on a character match = 1 - Character deleted on a character match Bit 0 - Interrupt on character match = 0 - Interrupt not enabled on a character match = 1 - Interrupt enabled on a character match (the interrupt occurs as soon as there is a match with a received data character)
Byte	1CH	Compare character match 2
Byte	1DH	Compare character function 2 (see the description of the Compare Character Function 1 field)
Byte	1EH	Compare character match 3
Byte	1FH	Compare character function 3 (see the description of the Compare Character Function 1 field)

Service-Specific Output

Size	Offset	Description
None		

Return Codes

Return codes are returned at offset hex 0C.

Value	Description
0000H	Operation Successfully Completed
0001H	Incomplete – Stage on Interrupt
0005H	Incomplete – Not My Interrupt, Stage on Interrupt
0009H	Attention, Stage on Interrupt
0081H	Unexpected Interrupt Reset, Stage on Interrupt
8000H	Device Busy, Request Refused
8001H	Async Hardware Level Not Available
8081H	Arbitration Level Not Available
8082H	Arbitration Level Not Allocated
8083H	Arbitration Level Disabled
8084H	DMA Transfer in Progress
8085H	No DMA Transfer in Progress
8086H	DMA Channel Not Available
8087H	Arbitration Level Not Disabled
9000H	Bad Communication Port
C000H	Invalid Logical ID (ABIOS transfer convention only)
C001H	Invalid Function
C003H	Invalid Unit Number
C004H	Invalid Request-Block Length
C020H	No Receive Active in Combined Interrupt

Figure 6-14. Asynchronous Communication Return Codes

Programming Considerations

- Asynchronous communication ABIOS can be initialized and operated in PIO mode without data dependencies other than what can be obtained in ABIOS. For DMA operation, additional data is expected to be returned from system services. The following data is required to support DMA operation:
  - Interrupt level
  - Transmit and receive arbitration levels
  - Base port pairs
  - Enhanced port pairs
  - Dynamic arbitration allocation ports
  - Fixed/dynamic arbitration information
  - Interface description information.

  This information is used to initialize the machine state for the asynchronous-communication logical ID that is being initialized. If this information is not available, the serial port is initialized to operate as a character device. Serial channel 1 is initialized to operate on interrupt level 4. All others operate on interrupt level 3.

- Asynchronous DMA operation also requires the use of the DMA device in the system. During initialization for DMA transfers, information from the DMA device is required. For this information to be available, the DMA device must be initialized before asynchronous communication ABIOS is initialized. If either the receive arbitration level or the transmit arbitration level is disabled for a serial port, ABIOS configures the serial port in the FIFO mode.

Notes:

Device ID 07H—System Timer

Functions

The following are the system timer functions. The Default Interrupt Handler function (hex 00) and the Return Logical ID Parameters function (hex 01) are described in "Request Block" in the "Transfer Conventions" section.

00H—Default Interrupt Handler

01H—Return Logical ID Parameters

02H—Reserved

03H—Read Device Parameters (Reserved)

04H—Set Device Parameters (Reserved)

05H—Reset/Initialize (Reserved)

06H—Enable (Reserved)

07H—Disable (Reserved)

08H—Read (Reserved)

09H—Write (Reserved)

0AH—Additional Data Transfer (Reserved)

Return Codes

Return codes are returned at offset hex 0C.

Value	Description
0000H	Operation Successfully Completed
0001H	Stage on Interrupt
0005H	Not My Interrupt, Stage on Interrupt
C000H	Invalid Logical ID (ABIOS transfer convention only)
C001H	Invalid Function Number
C003H	Invalid Unit Number
C004H	Invalid Request-Block Length

Figure 6-15. System Timer Return Codes

Programming Consideration

The system timer interrupt is handled through the default interrupt handler.

Device ID 08H—Real-Time Clock

Functions

The following are the real-time clock functions. The Default Interrupt Handler function (hex 00) and the Return Logical ID Parameters function (hex 01) are described in "Request Block" in the "Transfer Conventions" section.

Note: All reserved input fields must be set to 0.

00H—Default Interrupt Handler

01H—Return Logical ID Parameters

02H—Reserved

03H—Read Device Parameters

- This function returns the current settings of the real-time clock.
- If the alarm interrupts and periodic interrupts are enabled, this function returns the previously-set values for these interrupts.
- The Periodic Interrupt Rate field is valid only when the periodic interrupt function is enabled.
- The Alarm Hours, Alarm Minutes, and Alarm Seconds fields are valid only when the alarm interrupt function is enabled.
- If the real-time clock is in a clock-update cycle, the Return Code field is set to hex 8000 (Device in Use).
- If the real-time clock is not started, the Return Code field is set to hex 8001 (Real-Time Clock Not Started), and no other output fields are valid.
- The possible values of the Return Code field are hex 0000, 8000, and 8001.

Service-Specific Input

Size	Offset	Description
Word	16H	Reserved

Service-Specific Output

Size	Offset	Description
Byte	10H	Periodic interrupt rate Bits 7 to 4 - Reserved Bits 3 to 0 - Rate value set = 0000 - None = 0001 - 30.517 microseconds = 0010 - 61.035 microseconds = 0011 - 122.070 microseconds = 0100 - 244.141 microseconds = 0101 - 488.281 microseconds = 0110 - 976.562 microseconds = 0111 - 1.953125 milliseconds = 1000 - 3.90625 milliseconds = 1001 - 7.8125 milliseconds = 1010 - 15.625 milliseconds = 1011 - 31.250 milliseconds = 1100 - 62.500 milliseconds = 1101 - 125.00 milliseconds = 1110 - 250.00 milliseconds = 1111 - 500.00 milliseconds
Byte	11H	Real-time-clock status byte Bit 7 - Set bit status = 0 - Clock started = 1 - Clock not started Bit 6 - Periodic interrupt = 0 - Disabled = 1 - Enabled Bit 5 - Alarm interrupt = 0 - Disabled = 1 - Enabled Bit 4 - Update-ended interrupt = 0 - Disabled = 1 - Enabled Bits 3, 2 - Reserved Bit 1 - Clock mode = 0 - 12-hour mode = 1 - 24-hour mode Bit 0 - Reserved
Byte	12H	Alarm hours, in binary coded decimal (00 to 23D)
Byte	13H	Alarm minutes, in binary coded decimal (00 to 59D)
Byte	14H	Alarm seconds, in binary coded decimal (00 to 59D)

04H—Set Device Parameters

- This function sets the real-time clock to its different modes.
- The value of the Return Code field is hex 0000.

Service-Specific Input

Size	Offset	Description
Word	16H	Reserved
Byte	19H	Real-time-clock mode settings Bits 7 to 2 - Reserved Bit 1 - Hour mode = 0 - 12-hour mode = 1 - 24-hour mode Bit 0 - Reserved

Service-Specific Output

Size	Offset	Description
None		

05H—Reset/Initialize (Reserved)

06H—Enable (Reserved)

07H—Disable (Reserved)

08H—Read (Reserved)

09H—Write (Reserved)

0AH—Additional Data Transfer (Reserved)

0BH—Set Alarm Interrupt

- This function sets the alarm time according to the input values.
- The interrupt is continuous until it is canceled.
- If the interrupt is already enabled when this function is called, the request is refused, and the Return Code field is set to hex 8002 (Interrupt Already Enabled). The request must be canceled by the Cancel Alarm Interrupt function (hex 0C) before it can be restarted.
- If the real-time clock is in a clock-update cycle, the Return Code field is set to hex 8000 (Device in Use).
- If more than one request is returned with the Return Code field set to hex 0001 (Stage on Time), the caller is required to call only the first request. Return code hex 0005 (Not My Interrupt, Stage on Interrupt) indicates that the interrupt is not from the real-time-clock device. Return code hex 0001 (Stage on Interrupt) indicates that the interrupt is from the real-time-clock device but is not necessarily associated with the Request Block function.

- The Interrupt-Pending Status field indicates which interrupts occurred on return from the Interrupt routine when the Return Code field is set to hex 0001 (Stage on Interrupt).
- The real-time clock must be started through the Write Time and Date function (hex 12) before the Set Alarm Interrupt function (hex 0B) is called.
- A single real-time clock interrupt can indicate the occurrence of multiple interrupt types (alarm, periodic, or update-ended interrupt). These interrupt types are activated individually through the Set Alarm Interrupt function (hex 0B), Set Periodic Interrupt function (hex 0D), or Set Update-Ended Interrupt function (hex 0F). When a real-time clock interrupt occurs with multiple interrupt-type requests active, the caller must make a single call with any one of the outstanding request blocks to service the multiple outstanding requests. On return, the Interrupt-Pending Status field indicates which interrupts occurred and were serviced.
- The Cancel Alarm Interrupt function (hex 0C) can be used to cancel an outstanding Set Alarm Interrupt function request and must be called before a previously-set alarm time is changed.
- The possible values of the Return Code field are hex 0001, 0005, 8000, 8001, and 8002.

Service-Specific Input

Size	Offset	Description
Byte	12H	Alarm hours, in binary coded decimal (00 to 23D)
Byte	13H	Alarm minutes, in binary coded decimal (00 to 59D)
Byte	14H	Alarm seconds, in binary coded decimal (00 to 59D)
DWord	16H	Reserved

Service-Specific Output

Size	Offset	Description
Byte	1AH	Interrupt-pending status Bit 7 - Reserved Bit 6 - Periodic interrupt = 0 - Disabled = 1 - Enabled Bit 5 - Alarm interrupt = 0 - Disabled = 1 - Enabled Bit 4 - Update-ended interrupt = 0 - Disabled = 1 - Enabled Bits 3 to 0 - Reserved

0CH—Cancel Alarm Interrupt

- This function disables the alarm interrupt.
- The value of the Return Code field is hex 0000.

Service-Specific Input

Size	Offset	Description
Word	16H	Reserved

Service-Specific Output

Size	Offset	Description
None		

0DH—Set Periodic Interrupt

- This function sets the interval for the periodic interrupt.
- The interrupt is continuous until it is canceled.
- If the interrupt is already enabled when this function is called, the request is refused, and the Return Code field is set to hex 8002 (Interrupt Already Enabled). The request must be canceled by the Cancel Alarm Interrupt function (hex 0C) before it can be restarted.
- The real-time clock must be started through the Write Time and Date function (hex 12) before the Set Periodic Interrupt function (hex 0D) is called.
- If more than one request is returned with the Return Code field set to hex 0001 (Stage on Time), the caller is required to call only the first request. Return code hex 0005 (Not My Interrupt, Stage on Interrupt) indicates that the interrupt is not from the real-time-clock device. Return code hex 0001 (Stage on Interrupt) indicates that the interrupt is from the real-time-clock device but is not necessarily associated with the Request Block function.
- The Interrupt-Pending Status field indicates which interrupts occurred on return from the Interrupt routine when the Return Code field is set to hex 0001 (Stage on Interrupt).
- A single real-time clock interrupt can indicate the occurrence of multiple interrupt types (alarm, periodic, or update-ended interrupt). These interrupt types are activated individually through the Set Alarm Interrupt function (hex 0B), Set Periodic Interrupt function (hex 0D), or Set Update-Ended Interrupt function (hex 0F). When a real-time clock interrupt occurs with multiple interrupt-type requests active, the caller must make a single call

with any one of the outstanding request blocks to service the multiple outstanding requests. On return, the Interrupt-Pending Status field indicates which interrupts occurred and were serviced.

- The Cancel Periodic Interrupt function (hex 0E) can be used to cancel an outstanding Set Periodic Interrupt function request and must be called before a previously-set periodic interrupt interval is changed.
- The possible values of the Return Code field are hex 0001, 0005, 8001, and 8002.

Service-Specific Input

Size	Offset	Description
Byte	10H	Periodic interrupt rate Bits 7 to 4 - Reserved Bits 3 to 0 - Rate value set = 0000 - None = 0001 - 30.517 microseconds = 0010 - 61.035 microseconds = 0011 - 122.070 microseconds = 0100 - 244.141 microseconds = 0101 - 488.281 microseconds = 0110 - 976.562 microseconds = 0111 - 1.953125 milliseconds = 1000 - 3.90625 milliseconds = 1001 - 7.8125 milliseconds = 1010 - 15.625 milliseconds = 1011 - 31.250 milliseconds = 1100 - 62.500 milliseconds = 1101 - 125.00 milliseconds = 1110 - 250.00 milliseconds = 1111 - 500.00 milliseconds
DWord	16H	Reserved

Service-Specific Output

Size	Offset	Description
Byte	1AH	Interrupt-pending status Bit 7 - Reserved Bit 6 - Periodic interrupt = 0 - Disabled = 1 - Enabled Bit 5 - Alarm interrupt = 0 - Disabled = 1 - Enabled Bit 4 - Update-ended interrupt = 0 - Disabled = 1 - Enabled Bits 3 to 0 - Reserved

0EH—Cancel Periodic Interrupt

- This function disables the periodic interrupt.
- The value of the Return Code field is hex 0000.

Service-Specific Input

Size	Offset	Description
Word	16H	Reserved

Service-Specific Output

Size	Offset	Description
None		

0FH—Set Update Ended Interrupt

- This function enables the Update Ended interrupt.
- The interrupt is continuous until it is canceled.
- If the interrupt is already enabled when this function is called, the request is refused, and the Return Code field is set to hex 8002 (Interrupt Already Enabled). The request must be canceled by the Cancel Alarm Interrupt function (hex 0C) before it can be restarted.
- The real-time clock must be started through the Write Time and Date function (hex 12) before the Set Update-Ended Interrupt function (hex 0D) is called.
- If more than one request is returned with the Return Code field set to hex 0001 (Stage on Time), the caller is required to call only the first request. Return code hex 0005 (Not My Interrupt, Stage on Interrupt) indicates that the interrupt is not from the real-time-clock device. Return code hex 0001 (Stage on Interrupt) indicates that the interrupt is from the real-time-clock device but is not necessarily associated with the Request Block function.
- The Interrupt-Pending Status field indicates which interrupts occurred on return from the Interrupt routine when the Return Code field is set to hex 0001 (Stage on Interrupt).
- A single real-time clock interrupt can indicate the occurrence of multiple interrupt types (alarm, periodic, or update-ended interrupt). These interrupt types are activated individually through the Set Alarm Interrupt function (hex 0B), Set Periodic Interrupt function (hex 0D), or Set Update-Ended Interrupt function (hex 0F). When a real-time clock interrupt occurs with multiple interrupt-type requests active, the caller must make a single call

with any one of the outstanding request blocks to service the multiple outstanding requests. On return, the Interrupt-Pending Status field indicates which interrupts occurred and were serviced.

- The Cancel Update Ended Interrupt function (hex 10) can be used to cancel an outstanding Set Update Ended Interrupt function request and must be called before an Update Ended interrupt is restarted.
- The possible values of the Return Code field are hex 0001, 0005, 8001, and 8002.

Service-Specific Input

Size	Offset	Description
DWord	16H	Reserved

Service-Specific Output

Size	Offset	Description
Byte	1AH	Interrupt-pending status Bit 7 - Reserved Bit 6 - Periodic interrupt = 0 - Disabled = 1 - Enabled Bit 5 - Alarm interrupt = 0 - Disabled = 1 - Enabled Bit 4 - Update-ended interrupt = 0 - Disabled = 1 - Enabled Bits 3 to 0 - Reserved

10H—Cancel Update Ended Interrupt

- This function disables the Update Ended interrupt.
- The value of the Return Code field is hex 0000.

Service-Specific Input

Size	Offset	Description
Word	16H	Reserved

Service-Specific Output

Size	Offset	Description
None		

11H—Read Time and Date

- This function reads the current setting of the real-time clock.
- The real-time clock must be started through the Write Time and Date function (hex 12) before the Read Time and Date function (hex 11) is called.
- The Time and Date fields are valid only when the Return Code field is set to hex 0000 (Operation Successfully Completed).
- The possible values of the Return Code field are hex 0000, 8000, and 8001.

Service-Specific Input

Size	Offset	Description
Word	10H	Reserved

Service-Specific Output

Size	Offset	Description
Byte	12H	Hours, in binary coded decimal (00 to 23D)
Byte	13H	Minutes, in binary coded decimal (00 to 59D)
Byte	14H	Seconds, in binary coded decimal (00 to 59D)
Byte	15H	Century, in binary coded decimal (19D or 20D)
Byte	16H	Year, in binary coded decimal (00 to 99D)
Byte	17H	Month, in binary coded decimal (01D to 12D)
Byte	18H	Day, in binary coded decimal (01D to 31D)

12H—Write Time and Date

- This function starts the clock, if it is not already started, and sets the time and date information according to the input parameters.
- The value of the Return Code field is hex 0000.

Service-Specific Input

Size	Offset	Description
Word	10H	Reserved
Byte	12H	Hours, in binary coded decimal (00 to 23D)
Byte	13H	Minutes, in binary coded decimal (00 to 59D)
Byte	14H	Seconds, in binary coded decimal (00 to 59D)
Byte	15H	Century, in binary coded decimal (19D or 20D)
Byte	16H	Year, in binary coded decimal (00 to 99D)
Byte	17H	Month, in binary coded decimal (01D to 12D)
Byte	18H	Day, in binary coded decimal (01D to 31D)

Service-Specific Output

Size	Offset	Description
None		

Return Codes

Return codes are returned at offset hex 0C.

Value	Description
0000H	Operation Successfully Completed
0001H	Stage on Interrupt
0005H	Not My Interrupt, Stage on Interrupt
8000H	Device in Use
8001H	Real-Time Clock Not Started
8002H	Interrupt Already Enabled
C000H	Invalid Logical ID (ABIOS transfer convention only)
C001H	Invalid Function Number
C003H	Invalid Unit Number
C004H	Invalid Request-Block Length

Figure 6-16. Real-Time Clock Return Codes

Device ID 09H—System Services

Functions

The following are the system services functions. The Default Interrupt Handler function (hex 00) and the Return Logical ID Parameters function (hex 01) are described in "Request Block" in the "Transfer Conventions" section.

Note: All reserved input fields must be set to 0.

00H—Default Interrupt Handler

01H—Return Logical ID Parameters

02H—Reserved

03H—Used Internally by ABIOS

04H—Set Device Parameters (Reserved)

05H—Reset/Initialize (Reserved)

06H—Enable (Reserved)

07H—Disable (Reserved)

08H—Read (Reserved)

09H—Write (Reserved)

0AH—Additional Data Transfer (Reserved)

0BH—Switch to Real Mode

- This function switches the processor into real mode and disables address line 20.
- All interrupts, including the nonmaskable interrupt (NMI), are disabled. ABIOS gives control back to the caller at the location pointed to by the Resume Pointer field. The caller must reenable all interrupts, including the NMI.

- For 80386, 80386 SX, 80486, and 80486 SX systems, the selector to a Dummy Descriptor field in the caller's global descriptor table must have its segment limit set to the maximum (hex 0FFFF). The base address can be any value, and the access-rights byte is set to:
  - Expand upward (E = 0)
  - Writable (W = 1)
  - Present (P = 1).
- For all calls to this function, all registers must be saved and restored in the proper mode. A call to this function modifies all registers, including the FS and GS segment registers, in an 80386-, 80386 SX-, 80486-, or 80486 SX-compatible environment. Other ABIOS services depend on the value of the FS and GS segment registers. An 80286 operating system that uses ABIOS must dynamically determine the presence of the 80386 or compatible processor and save and restore the FS and GS segment registers.
- The Return Code field is not updated in this function unless a parameter error occurs. The AH register should be set to a value other than 0 so that it can be checked on return.

Service-Specific Input

Size	Offset	Description
DWord	10H	Reserved
DWord	60H	Resume pointer
Word	64H	Selector to a dummy descriptor field

Service-Specific Output

Size	Offset	Description
None		

0CH—Used Internally by ABIOS

0DH—Enable Address Line 20

- The possible value of the Return Code field is hex 0000.

Service-Specific Input

Size	Offset	Description
DWord	10H	Reserved

Service-Specific Output

Size	Offset	Description
None		

0EH—Disable Address Line 20

- The possible value of the Return Code field is hex 0000.

Service-Specific Input

Size	Offset	Description
DWord	10H	Reserved

Service-Specific Output

Size	Offset	Description
None		

0FH—Speaker

- This function enables the system speaker with the input frequency and duration.
- If the value of the Frequency Divisor field is 0, or the value of the Duration Counter field is 0, no action is performed, and the Return Code field is set to hex 0000 (Operation Successfully Completed).
- The possible value of the Return Code field is hex 0000.

Service-Specific Input

Size	Offset	Description
DWord	10H	Reserved
Word	60H	Frequency divisor (1.19 MHz/freq. div. = frequency freq. div. = 1331 for 886 Hz freq.)
Byte	66H	Duration counter, in 1/64 seconds

Service-Specific Output

Size	Offset	Description
None		

13H—Processor Functions

- This function enables or disables processor functions for testing and recovery.
- The possible values of the Return Code field are hex 0000, 80FE, 9000, 9001, 9002, and C001.

Service-Specific Input

Note: For an 80486 processor, disable any external caches when the on-chip cache (L1) is disabled.

Size	Offset	Description
Word	10H	Reserved
Byte	12H	Reserved
Byte	13H	Reserved
Word	14H	Processor functions = 0001H - Disable L1 cache = 0002H - Enable L1 cache = 0003H - Disable L2 cache = 0004H - Enable L2 cache = 0005H - Disable both caches = 0006H - Enable both caches = 0007H - Status of both caches = 0008H to FFFFH - Reserved

Service-Specific Output

Size	Offset	Description
Word	0CH	Return status
Byte	10H	Status of L1 cache, returned for function code hex 0007; the value is one of the following: = 00H - L1 cache is enabled = 01H - L1 cache is disabled = 02H - L1 cache is disabled because of a cache test error; the cache cannot be enabled
Byte	11H	Status of L2 cache, returned for function code hex 0007; the value is one of the following: = 00H - L2 cache is enabled = 01H - L2 cache is disabled = 02H - L2 cache is disabled because of a cache test error; the cache cannot be enabled

Return Codes

Return codes are returned at offset hex 0C.

Value	Description
0000H	Operation Successfully Completed
80FEH	Bad NVRAM
9000H	Cannot Perform Operation Because of State of Other Level of Cache
9001H	Cache Test Error
9002H	No Level 2 Cache Present
C000H	Invalid Logical ID (ABIOS transfer convention only)
C001H	Invalid Function Number
C003H	Invalid Unit Number
C004H	Invalid Request-Block Length

Figure 6-17. System Services Return Codes

Device ID 0AH—Nonmaskable Interrupt (NMI)

The nonmaskable interrupt (NMI) handler is used to process severe errors. In most cases, the fault is in the hardware, but it is also possible for a software error to force an NMI to occur.

Functions

The following are the nonmaskable interrupt (NMI) functions. The Default Interrupt Handler function (hex 00) and the Return Logical ID Parameters function (hex 01) are described in "Request Block" in the "Transfer Conventions" section.

Note: All reserved input fields must be set to 0.

00H—Default Interrupt Handler (Reserved)

01H—Return Logical ID Parameters

02H—Reserved

03H—Read Device Parameters (Reserved)

04H—Set Device Parameters (Reserved)

05H—Reset/Initialize (Reserved)

06H—Enable

- This function unmasks the NMI.
- The value of the Return Code field is hex 0000.

Service-Specific Input

Size	Offset	Description
Word	21H	Reserved

Service-Specific Output

Size	Offset	Description
None		

07H—Disable

- This function disables the NMI for memory parity and I/O channel checks.
- The DMA bus time-out NMI and the watchdog time-out NMI cannot be disabled through ABIOS.
- The value of the Return Code field is hex 0000.

Service-Specific Input

Size	Offset	Description
Word	21H	Reserved

Service-Specific Output

Size	Offset	Description
None		

08H—Continuous Read

- This function starts the monitoring of NMI status.
- If the NMI is caused by a direct memory access (DMA) arbitration bus time-out, the arbitration level that caused the time-out is returned in the request block.
- If the NMI is caused by a channel check, the slot number of the adapter that caused the NMI is returned in the request block.
- When the update of this request block is completed, the NMI is disabled. The caller should reenable the NMI through the Enable function (hex 06) if nonmaskable interrupts are desired.
- The possible values of the Return Code field are hex 0001, 0005, and 0009. Return code hex 0009 (Attention, Stage on Interrupt) indicates that an update has occurred and needs to be handled. In preparation for the next NMI, the caller should change the value of the Return Code field from hex 0009 to hex 0001 (Stage on Interrupt) after the request block has been reviewed.

Service-Specific Input

Size	Offset	Description
Word	21H	Reserved

Service-Specific Output

Size	Offset	Description
Word	10H	Type of NMI = 00H - Reserved = 01H - Parity = 02H - Channel check = 03H - DMA bus time-out = 04H - Watchdog time-out = 05H to FFFFH - Reserved
Byte	1EH	DMA arbitration level that caused the DMA bus time-out
Byte	1FH	Slot number that caused the I/O channel check

09H—Write (Reserved)

0AH—Additional Data Transfer (Reserved)

Return Codes

Return codes are returned at offset hex 0C.

Value	Description
0000H	Operation Successfully Completed
0001H	Stage on Interrupt
0005H	Not My Interrupt, Stage on Interrupt
0009H	Attention, Stage on Interrupt
C000H	Invalid Logical ID (ABIOS transfer convention only)
C001H	Invalid Function Number
C003H	Invalid Unit Number
C004H	Invalid Request-Block Length

Figure 6-18. Nonmaskable Interrupt Return Codes

Notes:

Device ID 0BH—Pointing Device

Functions

The following are the pointing device functions. The Default Interrupt Handler function (hex 00) and the Return Logical ID Parameters function (hex 01) are described in "Request Block" in the "Transfer Conventions" section.

Note: All reserved input fields must be set to 0.

00H—Default Interrupt Handler

01H—Return Logical ID Parameters

02H—Reserved

03H—Read Device Parameters

- This function returns the current pointing-device status and package-size setting.
- The value of the Return Code field is hex 0000.

Service-Specific Input

Size	Offset	Description
Word	1CH	Reserved

Service-Specific Output

Size	Offset	Description
Byte	10H	Interface status Bits 7 to 6 - Reserved Bit 5 - Interface enablement = 0 - Disabled = 1 - Enabled Bits 4 to 0 - Reserved
Byte	11H	Data-package size = 00H - Reserved = 01H - 1 byte = 02H - 2 bytes = 03H - 3 bytes = 04H - 4 bytes = 05H - 5 bytes = 06H - 6 bytes = 07H - 7 bytes = 08H - 8 bytes = 09H to FFH - Reserved

Size	Offset	Description
Word	12H	Current pointing-device status Bits 15 to 7 - Reserved Bit 6 - Mode = 0 - Stream mode = 1 - Remote/poll mode Bit 5 - Status = 0 - Disabled = 1 - Enabled Bit 4 - Scaling = 0 - Scaling set to 1:1 = 1 - Scaling set to 2:1 Bit 3 - Reserved Bit 2 - Left-button status = 0 - Not pressed = 1 - Pressed Bit 1 - Reserved Bit 0 - Right-button status = 0 - Not pressed = 1 - Pressed
Word	14H	Current resolution = 00H - 1 count per millimeter = 01H - 2 counts per millimeter = 02H - 4 counts per millimeter = 03H - 8 counts per millimeter = 04H to FFFFH - Reserved
Word	16H	Current sample rate = 0AH - 10 reports per second = 14H - 20 reports per second = 28H - 40 reports per second = 3CH - 60 reports per second = 50H - 80 reports per second = 64H - 100 reports per second = C8H - 200 reports per second
DWord	18H	Time to wait before resuming request, in microseconds

04H—Set Device Parameters (Reserved)

05H—Reset/Initialize Pointing Device

- This function resets the pointing device. On completion of this function, the pointing device is initialized as follows:
  - The package size remains unchanged.
  - Resolution = 4 counts per millimeter.
  - Sample rate = 100 reports per second.
  - Scaling = 1:1.
  - The pointing device is disabled.
- The possible values of the Return Code field are hex 0000, 0001, 0002, 0005, 8000, 8001, 8002, and 8003.

Service-Specific Input

Size	Offset	Description
Word	1CH	Reserved

Service-Specific Output

Size	Offset	Description
Byte	10H	Pointing-device completion code
Byte	11H	Pointing-device identification code
DWord	18H	Time to wait before resuming request, in microseconds

06H—Enable

- This function enables the pointing device.
- The possible values of the Return Code field are hex 0000, 0001, 0002, 0005, 8000, 8001, 8002, and 8003.

Service-Specific Input

Size	Offset	Description
Word	1CH	Reserved

Service-Specific Output

Size	Offset	Description
DWord	18H	Time to wait before resuming request, in microseconds

07H—Disable

- This function disables the pointing device.
- The possible values of the Return Code field are hex 0000, 0001, 0002, 0005, 8000, 8001, 8002, and 8003.

Service-Specific Input

Size	Offset	Description
Word	1CH	Reserved

Service-Specific Output

Size	Offset	Description
DWord	18H	Time to wait before resuming request, in microseconds

08H—Read

- This function is called to read the pointing device; it leaves the pointing device disabled. This function must be called before any other pointing-device function except the Return Logical ID Parameters function (hex 01). After the Read function (hex 08) has been called, the pointing device can be enabled through the Enable function (hex 06).
- The possible values of the Return Code field are hex 0001, 0002, 0005, 0009, 8000, 8003, and C005.

Service-Specific Input

Size	Offset	Description
Byte	10H	Package size = 00H - Reserved = 01H - 1 byte = 02H - 2 bytes = 03H - 3 bytes = 04H - 4 bytes = 05H - 5 bytes = 06H - 6 bytes = 07H - 7 bytes = 08H - 8 bytes = 09H to FFH - Reserved
DWord	12H	Reserved

Service-Specific Output

Size	Offset	Description
Byte	1CH	First byte from the pointing device – status Bit 7 - Y-data overflow = 0 - No overflow = 1 - Overflow Bit 6 - X-data overflow = 0 - No overflow = 1 - Overflow Bit 5 - Y-data sign = 0 - Positive = 1 - Negative Bit 4 - X-data sign = 0 - Positive = 1 - Negative Bit 3 - Reserved (set to 1) Bit 2 - Reserved (set to 0) Bit 1 - Right-button status = 0 - Not pressed = 1 - Pressed Bit 0 - Left-button status = 0 - Not pressed = 1 - Pressed
Byte	1DH	Second byte from the pointing device – X data
Byte	1EH	Third byte from the pointing device – Y data

Size	Offset	Description
5 bytes	1FH to 23H	Fourth byte to eighth byte from the pointing device (raw data)
4 bytes	24H to 27H	Ninth byte to twelfth byte from the pointing device (reserved)

09H—Write (Reserved)

0AH—Additional Data Transfer (Reserved)

0BH—Set Sample Rate

- This function sets the pointing-device sample rate.
- The possible values of the Return Code field are hex 0000, 0001, 0002, 0005, 8000, 8001, 8002, and 8003.

Service-Specific Input

Size	Offset	Description
Word	12H	Sample rate = 0AH - 10 reports per second = 14H - 20 reports per second = 28H - 40 reports per second = 3CH - 60 reports per second = 50H - 80 reports per second = 64H - 100 reports per second = C8H - 200 reports per second

Service-Specific Output

Size	Offset	Description
DWord	18H	Time to wait before resuming request, in microseconds

0CH—Set Resolution

- This function sets the pointing-device resolution.
- The possible values of the Return Code field are hex 0000, 0001, 0002, 0005, 8000, 8001, 8002, and 8003.

Service Specific Input

Size	Offset	Description
Word	12H	Resolution = 00H - 1 count per millimeter = 01H - 2 counts per millimeter = 02H - 4 counts per millimeter = 03H - 8 counts per millimeter = 04H to FFFFH - Reserved
Word	1CH	Reserved

Service-Specific Output

Size	Offset	Description
DWord	18H	Time to wait before resuming request, in microseconds

0DH—Set Scaling

- This function sets the pointing-device scaling value.
- The possible values of the Return Code field are hex 0000, 0001, 0002, 0005, 8000, 8001, 8002, and 8003.

Service-Specific Input

Size	Offset	Description
Byte	10H	Scaling value = 00H - Reserved = 01H - Set scaling to 1:1 = 02H - Set scaling to 2:1 = 03H to FFH - Reserved
Word	1CH	Reserved

Service-Specific Output

Size	Offset	Description
DWord	18H	Time to wait before resuming request, in microseconds

0EH—Read Pointing-Device Identification Code

- This function returns the pointing-device identification code.
- The possible values of the Return Code field are hex 0000, 0001, 0002, 0005, 8000, 8001, 8002, and 8003.

Service-Specific Input

Size	Offset	Description
Word	1CH	Reserved

Service-Specific Output

Size	Offset	Description
Byte	10H	Pointing-device identification code
DWord	18H	Time to wait before resuming request, in microseconds

Return Codes

Return codes are returned at offset hex 0C.

Value	Description
0000H	Operation Successfully Completed
0001H	Stage On Interrupt
0002H	Stage On Time
0005H	Not My Interrupt, Stage on Interrupt
0009H	Attention, Stage on Interrupt, Data Available
8000H	Device in Use
8001H	Resend
8002H	Two Consecutive Resends Found
8003H	System Lock
8004H	Nonacknowledgment Response
9100H	Controller Failure
C000H	Invalid Logical ID (ABIOS transfer convention only)
C001H	Invalid Function Number
C003H	Invalid Unit Number
C004H	Invalid Request-Block Length
C005H	Invalid Pointing-Device Parameter

Figure 6-19. Pointing Device Return Codes

Programming Consideration

The Read Pointing-Device Identification Code function (hex 0E) returns the device ID that is read from the auxiliary device interface. The IBM pointing device ID is hex 00.

Notes:

Device ID 0EH—Nonvolatile Random Access Memory (NVRAM)

Functions

The following are the NVRAM functions. The Default Interrupt Handler function (hex 00) and the Return Logical ID Parameters function (hex 01) are described in "Request Block" in the "Transfer Conventions" section.

Note: All reserved input fields must be set to 0.

00H—Default Interrupt Handler

01H—Return Logical ID Parameters

02H—Reserved

03H—Read Device Parameters (Reserved)

04H—Set Device Parameters (Reserved)

05H—Reset/Initialize (Reserved)

06H—Enable (Reserved)

07H—Disable (Reserved)

08H—Read NVRAM

- This function returns the data that is currently stored in the specified data buffer of the specified RAM (64-byte RAM or extended RAM).
- If the value of the Number of Bytes to Be Transferred field is 0, no action is performed, and the Return Code field is set to hex 0000 (Operation Successfully Completed).
- If the value of the Number of Bytes to Be Transferred field plus the value of the Starting RAM Address field is greater than the maximum amount of RAM, no action is performed, and the Return Code field is set to hex C005 (Invalid NVRAM Parameter).
- Because the Read NVRAM function (hex 08) allows access to real-time-clock data, the real-time clock must be checked to determine whether it is in a clock-update cycle before it can

return valid data. If the real-time clock is in a clock-update cycle, the Return Code field is set to hex 9101 (Retryable Device Error).

- The possible values of the Return Code field are hex 0000, 80FE, 80FF, 9101, C004, and C005.

Service-Specific Input

Size	Offset	Description
Word	10H	Reserved
DWord	12H	Pointer to data buffer
Word	16H	Reserved
DWord	1AH	Reserved
Word	20H	Flag word Bit 15 - NMI state on exit = 0 - NMI enabled = 1 - NMI disabled Bits 14 to 1 - Reserved Bit 0 - RAM type = 0 - 64-byte RAM = 1 - Extended RAM
Word	22H	Starting RAM address
Word	24H	Number of bytes to be transferred

Service-Specific Output

Size	Offset	Description
None		

09H—Write NVRAM

- This function writes the data that is in the specified data buffer to the specified location of the specified RAM (64-byte RAM or extended RAM).
- If the value of the Number of Bytes to Be Transferred field is 0, no action is performed, and the Return Code field is set to hex 0000 (Operation Successfully Completed).
- If the value of the Number of Bytes to Be Transferred field plus the value of the Starting RAM Address field is greater than the maximum amount of RAM, no action is performed, and the Return Code field is set to hex C005 (Invalid NVRAM Parameter).
- Because the Write NVRAM function (hex 09) allows access to real-time clock data, the real-time clock must be checked to determine whether it is in a clock-update cycle before it can return valid data. If the real-time clock is in a clock-update cycle, the Return Code field is set to hex 9101 (Retryable Device Error).
- The possible values of the Return Code field are hex 0000, 80FE, 80FF, 9101, C004, and C005.

Service-Specific Input

Size	Offset	Description
Word	10H	Reserved
DWord	12H	Pointer to data buffer
Word	16H	Reserved
DWord	1AH	Reserved
Word	20H	Flag word Bit 15 - NMI state on exit = 0 - NMI enabled = 1 - NMI disabled Bits 14 to 1 - Reserved Bit 0 - RAM type = 0 - 64-byte RAM = 1 - Extended RAM
Word	22H	Starting RAM address
Word	24H	Number of bytes to be transferred

Service-Specific Output

Size	Offset	Description
None		

0AH—Additional Data Transfer Function (Reserved)

0BH—Recompute Checksum

- This function recomputes the checksum for the specified RAM (64-byte or extended RAM).
- The possible values of the Return Code field are hex 0000, 80FE, C004, and 80FF.

Service-Specific Input

Size	Offset	Description
Word	16H	Reserved
Word	20H	Flag word Bit 15 - NMI state on exit = 0 - NMI enabled = 1 - NMI disabled Bits 14 to 1 - Reserved Bit 0 - RAM type = 0 - 64-byte RAM = 1 - Extended RAM

Service-Specific Output

Size	Offset	Description
None		

Return Codes

Return codes are returned at offset hex 0C.

Value	Description
0000H	Operation Successfully Completed
80FEH	NVRAM Checksum Invalid
80FFH	NVRAM Battery Bad
9101H	Retryable Device Error
C000H	Invalid Logical ID (ABIOS transfer convention only)
C001H	Invalid Function Number
C003H	Invalid Unit Number
C004H	Invalid Request-Block Length
C005H	Invalid NVRAM Parameter

Figure 6-20. Nonvolatile Random Access Memory (NVRAM) Return Codes

Device ID 0FH—Direct Memory Access (DMA)

Functions

The following are the direct memory access (DMA) functions. The Return Logical ID Parameters function (hex 01) is described in "Request Block" in the "Transfer Conventions" section.

Note: All reserved input fields must be set to 0.

00H—Reserved

01H—Return Logical ID Parameters

02H—Reserved

03H—Read Device Parameters

- This function returns the DMA device parameters.
- The DMA functional enhancements that are provided when the functional-support bit in the Parameter Flags field is set to 1 are as follows:
  - The Allocate Arbitration Level function (hex 0B) can allocate an arbitration level with or without allocating a channel.
  - Bits 0 and 1 of the Abort Arbitration Level Information field (offset hex 16) of the Abort Arbitration Level function (hex 0F) indicates whether an arbitration level is to be aborted, disabled, and deallocated; whether an arbitration level is to be aborted or disabled; or whether an arbitration level is only to be aborted.
  - The Read from Memory and Write to I/O function (hex 10), the Read from I/O and Write to Memory function (hex 11), and the Verify function (hex 12) each have 2 bits that select whether to search for an available arbitration level on which to perform the transfer and at what priority level the search is to start.
  - The bit numbers in the bit maps correspond to arbitration levels or channels. For example, bit 0 at offset hex 18 corresponds to arbitration level 0; bit 0 at offset hex 1C corresponds to DMA channel 0. If a bit in the Bit Map of Virtual Arbitration Levels field (offset hex 18) is set to 1, the corresponding arbitration level is programmable. If a bit in the Bit Map of Virtual DMA Channels field (offset hex 1C) is set to 1, the corresponding DMA channel is programmable.

- The value of the Return Code field is hex 0000.

Service-Specific Input

Size	Offset	Description
Word	16H	Reserved

Service-Specific Output

Size	Offset	Description
Word	10H	Maximum address, in 1MB (the actual value is 1 byte less than this value)
Word	12H	Maximum direct memory access transfer size, in 1KB
Byte	14H	Number of arbitration levels
Byte	15H	Number of direct memory access channels
Byte	16H	Parameter flags Bits 7 to 1 - Reserved Bit 0 - Functional support = 0 - DMA functional enhancements not supported = 1 - DMA functional enhancements supported
DWord	18H	Bit map of virtual (programmable) arbitration levels
DWord	1CH	Bit map of virtual (programmable) DMA channels

04H—Set Device Parameters (Reserved)

05H—Reset/Initialize (Reserved)

06H—Enable for Interrupts (Reserved)

07H—Disable for Interrupts (Reserved)

08H—Read (Reserved)

09H—Write (Reserved)

0AH—Additional Data Transfer (Reserved)

0BH—Allocate Arbitration Level

- This function allocates an arbitration level.
- Bit 0 of the Allocate Arbitration Bit Level Information field determines whether a channel is to be allocated for the arbitration level. This is useful for bus masters that need an arbitration level but not a channel.
- The possible values of the Return Code field are hex 0000, 8001, 8006, and C006.

Service-Specific Input

Size	Offset	Description
Byte	16H	Allocate arbitration bit level information Bits 7 to 1 - Reserved Bit 0 - Channel allocation = 0 - Allocate channel for requested arbitration level = 1 - No channel needed for this arbitration level
Byte	17H	Reserved
Byte	1FH	Arbitration level to be allocated

Service-Specific Output

Size	Offset	Description
None		

0CH—Deallocate Arbitration Level

- This function makes available a previously-allocated arbitration level. It also deallocates the channel that is associated with the arbitration level, if one had been allocated.
- The possible values of the Return Code field are hex 0000, 8002, 8004, 8007, and C006.

Service-Specific Input

Size	Offset	Description
Word	16H	Reserved
Byte	1FH	Arbitration level to be deallocated

Service-Specific Output

Size	Offset	Description
None		

0DH—Disable Arbitration Level

- This function disables the arbitration level in the DMA controller.
- The possible values of the Return Code field are hex 0000, 8002, 8004, and C006.

Service-Specific Input

Size	Offset	Description
Word	16H	Reserved
Byte	1FH	Arbitration level to be disabled

Service-Specific Output

Size	Offset	Description
None		

0EH—Transfer Status

- This function returns the number of transfers that are left to be completed, as read from the DMA controller.
- The possible values of the Return Code field are hex 0000, 8002, 8003, and C006.

Service-Specific Input

Size	Offset	Description
Word	16H	Reserved
Byte	1FH	Arbitration level to be checked

Service-Specific Output

Size	Offset	Description
DWord	18H	Number of transfers left

0FH—Abort

- This function accesses the DMA controller to disable the channel and to read both the physical address and the number of DMA transfers that were left when the Abort function was issued.
- The Abort function has additional capabilities available through the Abort Arbitration Level Information field (offset hex 16). If bit 1 is set to 1, the transfer is aborted, the channel is disabled, and the arbitration level is deallocated. If bit 1 is set to 0 and bit 0 is set to 1, the transfer is aborted, and the channel is disabled. If both bit 1 and bit 0 are set to 0, the transfer that is in progress is aborted. The physical address and the number of transfers that remained when the transfer was aborted are returned.
- The possible values of the Return Code field are hex 0000, 8002, 8003, 8005, and C006.

Service-Specific Input

Size	Offset	Description
Byte	16H	Abort arbitration level information Bits 7 to 2 - Reserved Bit 1 - Abort, disable, and deallocate arbitration level = 0 - Do not deallocate the arbitration level; refer to bit 0 for channel action = 1 - Abort transfer, disable channel, and deallocate arbitration level (not dependent on bit 0) Bit 0 - Abort or disable channel = 0 - Abort transfer that is in progress = 1 - Abort or disable with no contingency on a transfer being in progress
Byte	17H	Reserved
Byte	1FH	Arbitration level on which to abort operation

Service-Specific Output

Size	Offset	Description
DWord	10H	Physical address when abort was issued
DWord	18H	Number of transfers remaining when abort was issued

10H—Read from Memory and Write to I/O

- This function starts a DMA transfer from memory to an I/O device by programming the DMA controller with the indicated values from the request block.
- The Number of DMA Transfers field specifies the number of transfers that are to be executed by the DMA channel. The number of transfers is always 1 more than the specified count. For example, if the count is 0, one transfer is executed.

- The Mode Control field provides the following functions:
  - Bit 3 is used only if bit 1 is set to 1. Bit 3 determines whether the search for an available arbitration level starts with the lowest-priority or the highest-priority arbitration level. When bit 3 is set to 0, the search starts with the highest priority.
  - Bit 2 indicates that the I/O address will be programmed to the DMA controller. When bit 2 is set to 1, the DMA controller drives the I/O address that is specified at offset hex 14 of the request block onto the address bus during the DMA cycles, instead of driving an I/O address of 0.
  - Bit 1 specifies whether to use the arbitration level that is already allocated or to search for an available level. When bit 1 is set to 1, ABIOS searches for an available level, starts the transfer, and returns the allocated arbitration level at offset hex 1F. When bit 1 is set to 0, ABIOS uses the arbitration level that is specified at offset hex 1F. Bit 3 determines whether the search starts with the highest priority or the lowest priority.
  - Bit 0 causes the DMA controller to automatically initialize the transfer count and memory address to their previously-programmed values after the terminal count is reached.
- The Transfer Control field provides the following functions:
  - Bits 1 and 0 indicate whether the transfer size is 8 bits or 16 bits. The transfer-count size that is returned is in bytes if the device size is 8 bits; it is in words if the device size is 16 bits.
  - Bit 2 specifies whether the physical address of memory is increased or decreased during the transfer.
- The possible values of the Return Code field are hex 0000, 8001, 8002, 8004, 8006, C005, and C006.

Service-Specific Input

Size	Offset	Description
DWord	10H	Physical address of memory
DWord	14H	Physical address of I/O
DWord	18H	Number of DMA data transfers
Byte	1CH	Mode control Bits 7 to 4 - Reserved Bit 3 = 1 - Start search with highest priority Bit 2 = 1 - Program I/O address Bit 1 = 1 - Allocate any arbitration level for transfer Bit 0 = 1 - Autoinitialization

Size	Offset	Description
Byte	1DH	Transfer control Bits 7 to 3 - Reserved Bit 2 - Count control = 0 - Increase = 1 - Decrease Bits 1, 0 - Device size = 00 - 8 bits = 01 - 16 bits = 10 - Reserved = 11 - Reserved
Byte	1FH	Arbitration level to be used; input only if bit 1 of the Mode Control field is set to 0

Service-Specific Output

Size	Offset	Description
Byte	1FH	Allocated arbitration level

11H—Read from I/O and Write to Memory

- This function starts a DMA transfer from an I/O device to memory by programming the DMA controller with the indicated values from the request block.
- The Number of DMA Transfers field specifies the number of transfers that are to be executed by the DMA channel. The number of transfers is always 1 more than the specified count. For example, if the count is 0, one transfer is executed.
- The Mode Control field provides the following functions:
  - Bit 3 is used only if bit 1 is set to 1. Bit 3 determines whether the search for an available arbitration level starts with the lowest-priority or the highest-priority arbitration level. When bit 3 is set to 0, the search starts with the highest priority.
  - Bit 2 indicates that the I/O address will be programmed to the DMA controller. When bit 2 is set to 1, the DMA controller drives the I/O address that is specified at offset hex 14 of the request block onto the address bus during the DMA cycles, instead of driving an I/O address of 0.
  - Bit 1 specifies whether to use the arbitration level that is already allocated or to search for an available level. When bit 1 is set to 1, ABIOS searches for an available level, starts the transfer, and returns the allocated arbitration level at offset hex 1F. When bit 1 is set to 0, ABIOS uses the arbitration level that is specified at offset hex 1F. Bit 3 determines whether the search starts with the highest priority or the lowest priority.

- Bit 0 causes the DMA controller to automatically initialize the transfer count and memory address to their previously-programmed values after the terminal count is reached.

- The Transfer Control field provides the following functions:
  - Bits 1 and 0 indicate whether the transfer size is 8 bits or 16 bits. The transfer-count size that is returned is in bytes if the device size is 8 bits; it is in words if the device size is 16 bits.
  - Bit 2 specifies whether the physical address of memory is increased or decreased during the transfer.

- The possible values of the Return Code field are hex 0000, 8001, 8002, 8004, 8006, C005, and C006.

Service-Specific Input

Size	Offset	Description
DWord	10H	Physical address of memory
DWord	14H	Physical address of I/O
DWord	18H	Number of DMA data transfers
Byte	1CH	Mode control Bits 7 to 4 - Reserved Bit 3 = 1 - Start search with highest priority Bit 2 = 1 - Program I/O address Bit 1 = 1 - Allocate any arbitration level for transfer Bit 0 = 1 - Autoinitialization
Byte	1DH	Transfer control Bits 7 to 3 - Reserved Bit 2 - Count control = 0 - Increase = 1 - Decrease Bits 1, 0 - Device size = 00 - 8 bits = 01 - 16 bits = 10 - Reserved = 11 - Reserved
Byte	1FH	Arbitration level to be used; input only if bit 1 of the Mode Control field is set to 0

Service-Specific Output

Size	Offset	Description
Byte	1FH	Allocated arbitration level

12H—Verify

- This function sets up the DMA controller, using the indicated values from the request block, to perform a read verification. The DMA controller performs a memory read without actually performing the transfer, until the number of DMA data transfers is reached.
- The Number of DMA Transfers field specifies the number of transfers that are to be executed by the DMA channel. The number of transfers is always 1 more than the specified count. For example, if the count is 0, one transfer is executed.
- The Mode Control field provides the following functions:
  - Bit 3 is used only if bit 1 is set to 1. Bit 3 determines whether the search for an available arbitration level starts with the lowest-priority or the highest-priority arbitration level. When bit 3 is set to 0, the search starts with the highest priority.
  - Bit 2 indicates that the I/O address will be programmed to the DMA controller. When bit 2 is set to 1, the DMA controller drives the I/O address that is specified at offset hex 14 of the request block onto the address bus during the DMA cycles, instead of driving an I/O address of 0.
  - Bit 1 specifies whether to use the arbitration level that is already allocated or to search for an available level. When bit 1 is set to 1, ABIOS searches for an available level, starts the transfer, and returns the allocated arbitration level at offset hex 1F. When bit 1 is set to 0, ABIOS uses the arbitration level that is specified at offset hex 1F. Bit 3 determines whether the search starts with the highest priority or the lowest priority.
  - Bit 0 causes the DMA controller to automatically initialize the transfer count and memory address to their previously-programmed values after the terminal count is reached.
- The Transfer Control field provides the following functions:
  - Bits 1 and 0 indicate whether the transfer size is 8 bits or 16 bits. The transfer-count size that is returned is in bytes if the device size is 8 bits; it is in words if the device size is 16 bits.
  - Bit 2 specifies whether the physical address of memory is increased or decreased during the transfer.
- The possible values of the Return Code field are hex 0000, 8001, 8002, 8004, 8006, C005, and C006.

Service-Specific Input

Size	Offset	Description
DWord	10H	Physical address of memory
DWord	14H	Physical address of I/O
DWord	18H	Number of DMA data transfers
Byte	1CH	Mode control Bits 7 to 4 - Reserved Bit 3 = 1 - Start search with highest priority Bit 2 = 1 - Program I/O address Bit 1 = 1 - Allocate any arbitration level for transfer Bit 0 = 1 - Autoinitialization
Byte	1DH	Transfer control Bits 7 to 3 - Reserved Bit 2 - Count control = 0 - Increase = 1 - Decrease Bits 1, 0 - Device size = 00 - 8 bits = 01 - 16 bits = 10 - Reserved = 11 - Reserved
Byte	1FH	Arbitration level to be used; input only if bit 1 of the Mode Control field is set to 0

Service-Specific Output

Size	Offset	Description
Byte	1FH	Allocated arbitration level

13H—Abort and Start New DMA Transfer

- This function accesses the DMA controller to disable the channel and to read both the physical address and the number of DMA transfers that were left when the abort request was issued. Then the DMA controller is programmed for the next DMA operation with the indicated values from the request block.
- The Number of DMA Transfers field specifies the number of transfers that are to be executed by the DMA channel. The number of transfers is always 1 more than the specified count. For example, if the count is 0, one transfer is executed.

- The Mode Control field provides the following functions:
  - Bit 3 is used only if bit 1 is set to 1. Bit 3 determines whether the search for an available arbitration level starts with the lowest-priority or the highest-priority arbitration level. When bit 3 is set to 0, the search starts with the highest priority.
  - Bit 2 indicates that the I/O address will be programmed to the DMA controller. When bit 2 is set to 1, the DMA controller drives the I/O address that is specified at offset hex 14 of the request block onto the address bus during the DMA cycles, instead of driving an I/O address of 0.
  - Bit 1 specifies whether to use the arbitration level that is already allocated or to search for an available level. When bit 1 is set to 1, ABIOS searches for an available level, starts the transfer, and returns the allocated arbitration level at offset hex 1F. When bit 1 is set to 0, ABIOS uses the arbitration level that is specified at offset hex 1F. Bit 3 determines whether the search starts with the highest priority or the lowest priority.
  - Bit 0 causes the DMA controller to automatically initialize the transfer count and memory address to their previously-programmed values after the terminal count is reached.
- The Transfer Control field provides the following functions:
  - Bits 1 and 0 indicate whether the transfer size is 8 bits or 16 bits. The transfer-count size that is returned is in bytes if the device size is 8 bits; it is in words if the device size is 16 bits.
  - Bit 2 specifies whether the physical address of memory is increased or decreased during the transfer.
- The possible values of the Return Code field are hex 0000, 8001, 8002, 8004, 8006, C005, and C006.

Service-Specific Input

Size	Offset	Description
DWord	10H	Physical address of memory
DWord	14H	Physical address of I/O
DWord	18H	Number of DMA data transfers
Byte	1CH	Mode control Bits 7 to 4 - Reserved Bit 3 = 1 - Start search with highest priority Bit 2 = 1 - Program I/O address Bit 1 = 1 - Allocate any arbitration level for transfer Bit 0 = 1 - Autoinitialization

Size	Offset	Description
Byte	1DH	Transfer control Bits 7 to 3 - Reserved Bit 2 - Count control = 0 - Increase = 1 - Decrease Bits 1, 0 - Device size = 00 - 8 bits = 01 - 16 bits = 10 - Reserved = 11 - Reserved
Byte	1FH	Arbitration level to be used; input only if bit 1 of the Mode Control field is set to 0

Service-Specific Output

Size	Offset	Description
DWord	10H	Physical address when abort was issued
DWord	18H	Number of transfers left when abort was issued

14H—Get Current Allocation Status

- This function returns the current allocation status for DMA channels and arbitration levels.
- The bit numbers in the bit maps correspond to arbitration levels or channels. For example, bit 0 at offset hex 10 corresponds to arbitration level 0; bit 0 at offset hex 14 corresponds to DMA channel 0. If a bit in the Bit Map of Allocated Arbitration Levels field (offset hex 10) is set to 1, the corresponding arbitration level is allocated. If a bit in the Bit Map of Allocated DMA Channels field (offset hex 14) is set to 1, the corresponding DMA channel is allocated. There are 15 arbitration levels and eight channels; undefined bits in the field are set to 0.
- The value of the Return Code field is hex 0000.

Service-Specific Input

Size	Offset	Description
DWord	1AH	Reserved

Service-Specific Output

Size	Offset	Description
DWord	10H	Bit map of allocated arbitration levels
DWord	14H	Bit map of allocated DMA channels
Byte	18H	Number of arbitration levels not currently allocated
Byte	19H	Number of DMA channels not currently allocated

Return Codes

Return codes are returned at offset hex 0C.

Value	Description
0000H	Operation Successfully Completed
0005H	Not My Interrupt, Stage on Interrupt
8000H	Device in Use
8001H	Arbitration Level Not Available
8002H	Arbitration Level Not Allocated
8003H	Arbitration Level Disabled
8004H	Transfer in Progress
8005H	No Transfer in Progress
8006H	No Channel Available
8007H	Arbitration Level Not Disabled
C000H	Invalid ID (ABIOS transfer convention only)
C001H	Invalid Function Number
C003H	Invalid Unit Number
C004H	Invalid Request-Block Length
C005H	Invalid DMA Parameters
C006H	Invalid DMA Arbitration Level Specified

Figure 6-21. Direct Memory Access (DMA) Return Codes

Programming Considerations

- DMA channels are either physical or virtual. A physical channel can have only one arbitration level assigned to it. A virtual channel can be programmed to use any arbitration level that is not currently assigned to a different channel.
- There is no difference in function between physical and virtual channels. Priority of the channels is determined by the arbitration level; arbitration level hex 00 has the highest priority, and arbitration level hex 0E has the lowest priority. Arbitration level hex 0F is reserved.
- To perform a DMA transfer operation, a caller performs the following steps:
  1. Request an arbitration level.
  2. Set up a transfer to a device.
  3. Disable the arbitration level.
  4. Deallocate the arbitration level.
- Direct reading or writing of the DMA controller ports can cause unpredictable results.

Notes:

Device ID 10H—Programmable Option Select (POS)

Functions

The following are the POS functions. The Default Interrupt Handler function (hex 00) and the Return Logical ID Parameters function (hex 01) are described in "Request Block" in the "Transfer Conventions" section.

Note: All reserved input fields must be set to 0.

00H—Default Interrupt Handler

01H—Return Logical ID Parameters

02H—Reserved

03H—Read Device Parameters (Reserved)

04H—Set Device Parameters (Reserved)

05H—Reset/Initialize (Reserved)

06H—Enable (Reserved)

07H—Disable (Reserved)

08H—Read (Reserved)

09H—Write (Reserved)

0AH—Additional Data Transfer Function (Reserved)

0BH—Read Stored POS Data to Memory

- This function returns the programmable option select data that is currently stored in 64-byte RAM or extended RAM for the specified slot.
- For system-board option select data, the output buffer contains the system-board option select byte.
- For adapter option select data, the output buffer contains 4 bytes of data (adapter option select bytes 1, 2, 3, and 4).

- If the value of the Slot Number field is greater than the maximum number of slots, no action is performed, and the Return Code field is set to hex C005 (Invalid POS Parameter).
- The possible values of the Return Code field are hex 0000, 80FE, 80FF, and C005.

Service-Specific Input

Size	Offset	Description
Byte	10H	Slot number Bits 7 to 4 - Reserved Bits 3 to 0 - Slot number (values in binary) = 0000 - System board (planar) = 0001 - Slot 1 = 0010 - Slot 2 = 0011 - Slot 3 = 0100 - Slot 4 = 0101 - Slot 5 = 0110 - Slot 6 = 0111 - Slot 7 = 1000 - Slot 8
Byte	11H	Reserved
Word	14H	Reserved
DWord	16H	Pointer to data buffer
Word	1CH	Reserved

Service-Specific Output

Size	Offset	Description
Word	12H	Adapter ID

Output Buffer (System-Board POS)

Size	Offset	Description
Byte	00H	System-board POS data

Output Buffer (Adapter POS)

Size	Offset	Description
Byte	00H	Adapter option-select byte 1
Byte	01H	Adapter option-select byte 2
Byte	02H	Adapter option-select byte 3
Byte	03H	Adapter option-select byte 4

0CH—Write Stored POS Data from Memory

- This function writes the programmable option select data to the specified slot locations of the appropriate RAM (64 byte RAM or extended RAM).
- For system-board option select data, the output buffer contains the system-board option select byte.

- For adapter option select data, the output buffer contains 4 bytes of data (adapter option select bytes 1, 2, 3, and 4).
- If the value of the Slot Number field is greater than the maximum number of slots, no action is performed, and the Return Code field is set to hex C005 (Invalid POS Parameter).
- The possible values of the Return Code field are hex 0000, 80FE, 80FF, and C005.

Service-Specific Input

Size	Offset	Description
Byte	10H	Slot number Bits 7 to 4 - Reserved Bits 3 to 0 - Slot number (values in binary) = 0000 - System board (planar) = 0001 - Slot 1 = 0010 - Slot 2 = 0011 - Slot 3 = 0100 - Slot 4 = 0101 - Slot 5 = 0110 - Slot 6 = 0111 - Slot 7 = 1000 - Slot 8
Byte	11H	Reserved
Word	12H	Adapter ID
Word	14H	Reserved
DWord	16H	Pointer to data buffer
Word	1CH	Reserved

Service-Specific Output

Size	Offset	Description
None		

Output Buffer (System-Board POS)

Size	Offset	Description
Byte	00H	System-board POS data

Output Buffer (Adapter POS)

Size	Offset	Description
Byte	00H	Adapter option-select byte 1
Byte	01H	Adapter option-select byte 2
Byte	02H	Adapter option-select byte 3
Byte	03H	Adapter option-select byte 4

0DH—Read Dynamic POS Data to Memory

- This function reads the supplied programmable option select data to the adapter in the specified slot.

- For system-board option select data, the output buffer contains the system-board option select byte.
- For adapter option select data, the output buffer contains 4 bytes of data (adapter option select bytes 1, 2, 3, and 4).
- If the value of the Slot Number field is greater than the maximum number of slots, no action is performed, and the Return Code field is set to C005 (Invalid POS Parameter).
- The possible values of the Return Code field are hex 0000 and C005.

Service-Specific Input

Size	Offset	Description
Byte	10H	Slot number (values in binary) Bits 7 to 4 - Reserved Bits 3 to 0 - Slot number = 0000 - System board (planar) = 0001 - Slot 1 = 0010 - Slot 2 = 0011 - Slot 3 = 0100 - Slot 4 = 0101 - Slot 5 = 0110 - Slot 6 = 0111 - Slot 7 = 1000 - Slot 8
Byte	11H	Reserved
Word	14H	Reserved
DWord	16H	Pointer to data buffer
Word	1CH	Reserved

Service-Specific Output

Size	Offset	Description
Word	12H	Card ID

Output Buffer (System-Board POS)

Size	Offset	Description
Byte	00H	System-board POS data

Output Buffer (Adapter POS)

Size	Offset	Description
Byte	00H	Adapter option-select byte 1
Byte	01H	Adapter option-select byte 2
Byte	02H	Adapter option-select byte 3
Byte	03H	Adapter option-select byte 4

0EH—Write Dynamic POS Data from Memory

- This function writes the supplied programmable option select data to the adapter in the specified slot.
- For system-board option select data, the output buffer contains the system-board option select byte.
- For adapter option select data, the output buffer contains 4 bytes of data (adapter option select bytes 1, 2, 3, and 4).
- If the value of the Slot Number field is greater than the maximum number of slots, no action is performed, and the Return Code field is set to C005 (Invalid POS Parameter).
- The possible values of the Return Code field are hex 0000 and C005.

Service-Specific Input

Size	Offset	Description
Byte	10H	Slot number (values in binary) Bits 7 to 4 - Reserved Bits 3 to 0 - Slot Number = 0000 - System board (planar) = 0001 - Slot 1 = 0010 - Slot 2 = 0011 - Slot 3 = 0100 - Slot 4 = 0101 - Slot 5 = 0110 - Slot 6 = 0111 - Slot 7 = 1000 - Slot 8
Byte	11H	Reserved
Word	14H	Reserved
DWord	16H	Pointer to data buffer
Word	1CH	Reserved

Service-Specific Output

Size	Offset	Description
None		

Input Buffer (System-Board POS)

Size	Offset	Description
Byte	00H	System-board POS data

Input Buffer (Adapter POS)

Size	Offset	Description
Byte	00H	Adapter option-select byte 1
Byte	01H	Adapter option-select byte 2
Byte	02H	Adapter option-select byte 3
Byte	03H	Adapter option-select byte 4

Return Codes

Return codes are returned at offset hex 0C.

Value	Description
0000H	Operation Successfully Completed
80FEH	NVRAM Checksum Invalid
80FFH	NVRAM Battery Bad
C000H	Invalid Logical ID (ABIOS transfer convention only)
C001H	Invalid Function Number
C003H	Invalid Unit Number
C004H	Invalid Request-Block Length
C005H	Invalid POS Parameter

Figure 6-22. Programmable Option Select (POS) Return Codes

Device ID 16H—Keyboard Security

Functions

The following are the keyboard security functions. The Default Interrupt Handler function (hex 00) and the Return Logical ID Parameters function (hex 01) are described in "Request Block" in the "Transfer Conventions" section.

Note: All reserved input fields must be set to 0.

00H—Default Interrupt Handler (Reserved)

01H—Return Logical ID Parameters

02H—Reserved

03H—Read Device Parameters

- This function returns the maximum password length.
- The possible value of the Return Code field is hex 0000.

Service-Specific Input

Size	Offset	Description
Byte	11H	Reserved

Service-Specific Output

Size	Offset	Description
Byte	10H	Maximum password length

04H—Set Device Parameters (Reserved)

05H—Reset/Initialize (Reserved)

06H—Enable

- This function enables password security.
- The possible values of the Return Code field are hex 0000, 8000, and 8003.

Service-Specific Input

Size	Offset	Description
Byte	11H	Reserved

Service-Specific Output

Size	Offset	Description
None		

07H—Disable (Reserved)

08H—Read (Reserved)

09H—Write (Reserved)

0AH—Additional Data Transfer (Reserved)

0BH—Write Password

- This function changes the password.
- Password scan codes are written to the keyboard controller. A password can consist of from 1 to 7 bytes of keyboard-make scan codes. It can consist of letters, numbers, and other characters; keys that are tagged with a null byte (such as Alt, Caps Lock, Ctrl, Num Lock, Shift, and Scroll Lock) are not valid.
- If the value of the Password Length field is 0 or is greater than the maximum password length, no action is performed, and the Return Code field is set to hex C005 (Invalid Keyboard-Security Parameter).
- The maximum password length is returned in the Read Device Parameters function (hex 03).
- The possible values of the Return Code field are hex 0000, 8000, and 8003.

Service-Specific Input

Size	Offset	Description
Byte	10H	Password length (bytes)
Byte	11H	Reserved
Byte	12H	First scan code
Byte	13H	Second scan code
Byte	14H	Third scan code
Byte	15H	Fourth scan code
Byte	16H	Fifth scan code
Byte	17H	Sixth scan code
Byte	18H	Seventh scan code

Service-Specific Output

Size	Offset	Description
None		

0CH—Write Invocation Byte

- This function changes the invocation-byte scan code, which is used to signal to the system that keyboard security has been activated with a valid password. After keyboard security is activated, the system sends this byte (by using the Keyboard interrupt) to the operating system as if the byte were a scan code. If the invocation byte is set to 0, the system does not send this byte after keyboard security has been activated.
- The possible values of the Return Code field are hex 0000, 8000, and 8003.

Service-Specific Input

Size	Offset	Description
Byte	10H	Invocation-byte scan code
Byte	11H	Reserved

Service-Specific Output

Size	Offset	Description
None		

0DH—Write Match Byte

- This function changes the match-byte scan code, which is used to signal to the system that keyboard security has been deactivated with the correct password. After the correct sequence is typed, the system sends this byte (by using the Keyboard interrupt) to the operating system as if the byte were a scan code. If the match byte is set to 0, the system does not send this byte after keyboard security has been deactivated.
- The possible values of the Return Code field are hex 0000, 8000, and 8003.

Service-Specific Input

Size	Offset	Description
Byte	10H	Match-byte scan code
Byte	11H	Reserved

Service-Specific Output

Size	Offset	Description
None		

0EH—Write Filter Byte 1

- This function changes filter byte 1. Filter bytes are scan codes that are ignored during password validation. For example, it might be preferable to ignore the scan code for the shift keys.
- The possible values of the Return Code field are hex 0000, 8000, and 8003.

Service-Specific Input

Size	Offset	Description
Byte	10H	Filter byte 1
Byte	11H	Reserved

Service-Specific Output

Size	Offset	Description
None		

0EH—Write Filter Byte 2

- This function changes filter byte 2. Filter bytes are scan codes that are ignored during password validation. For example, it might be preferable to ignore the scan code for the shift keys.
- The possible values of the Return Code field are hex 0000, 8000, and 8003.

Service-Specific Input

Size	Offset	Description
Byte	10H	Filter byte 2
Byte	11H	Reserved

Service-Specific Output

Size	Offset	Description
None		

Return Codes

Return codes are returned at offset hex 0C.

Value	Description
0000H	Operation Successfully Completed
8000H	Device Busy
8003H	Device Inhibited
C000H	Invalid Logical ID (ABIOS transfer convention only)
C001H	Invalid Function Number
C003H	Invalid Unit Number
C004H	Invalid Request-Block Length
C005H	Invalid Keyboard-Security Parameter

Figure 6-23. Keyboard-Security Return Codes

Notes:

Device ID 17H—SCSI Subsystem Interface

Functions

The functions that are described in this section control the small computer system interface (SCSI) subsystem. Some of these functions affect all devices that are attached to the SCSI subsystem; therefore, the programmer should have a thorough understanding of the operation of the SCSI bus and subsystem before using these functions (see the technical reference manual for the subsystem).

The Default Interrupt Handler function (hex 00) and the Return Logical ID Parameters function (hex 01) are described in "Request Block" in the "Transfer Conventions" section.

Note: All reserved input fields must be set to 0.

00H—Default Interrupt Handler

01H—Return Logical ID Parameters

02H—Reserved

03H—Read Device Parameters

- This function returns information about the SCSI subsystem.
- All return-code values are possible.

Service-Specific Input

Size	Offset	Description
Word	28H	Reserved

Service-Specific Output

Size	Offset	Description
Word	10H	Length of device configuration table, in bytes (see function hex 0B)
Byte	12H	Subsystem control block (SCB) architecture card compatibility level
Word	14H	Reserved

04H—Set Device Parameters (Reserved)

05H—Reset/Initialize

- This function issues either a hard reset or a soft reset to the subsystem. The Reset Type field selects the type of reset that will be issued.
- The subsystem and its devices must be in a stable state throughout this function.
- This function affects the SCSI and disk BIOS routines.
- If the reset type is not within the range that is defined in the Reset Type field (offset hex 10), the Return Code field is set to hex C005 (Invalid Parameter), and no action is taken.

Hard Reset

- The subsystem stops all current activity, including any active requests that the subsystem is maintaining for attached devices. However, the devices can continue to process the requests, and the programs that are controlling the devices can appear to still have active requests, even though these requests will not be successfully completed. This results in an apparent time-out condition for each logical ID that has an outstanding request. Invoke the Time-Out routine for those logical IDs.
- For reset type 3, ABIOS restores the DMA pacing value to the value that was in effect before the reset. Also, the device time-out states are restored (on a device-by-device basis) to the states that were in effect before the reset.
- For reset type 2, the parameters remain at their default values.
- The assignments are always reestablished.
- The subsystem-retry state remains at the default setting (enabled).
- All return-code values are possible.

Soft Reset

- The subsystem stops all current activity and activates the SCSI reset signal.
- A soft reset does not change the adapter DMA pacing and time-out states.
- All return-code values are possible.

Service-Specific Input

Size	Offset	Description
Byte	10H	Reset type = 00H to 01H - Reserved = 02H - Hard reset; leaves subsystem states at default values = 03H - Hard reset; restores subsystem states to their pre-reset values = 04H - Soft reset; issues 'SCSI bus reset' signal = 05H to FFH - Reserved
Word	16H	Reserved

Service-Specific Output

Size	Offset	Description
DWord	28H	Time to wait before resuming request, in microseconds
Byte	32H	Status Bits 7 to 2 - Reserved Bit 1 - Command-complete status indicator (see "Programming Considerations" on page 6-ID17-9) = 0 - Command-complete status not required = 1 - Command-complete status should be requested for more information Bit 0 - Reserved

06H—Enable (Reserved)

07H—Disable (Reserved)

08H—Read (Reserved)

09H—Write (Reserved)

0AH—Additional Data Transfer (Reserved)

0BH—Return Device Configuration Table

- This function builds a device configuration table at the specified address. This table describes the physical devices that are attached to the SCSI subsystem.
- The first entry in the device configuration table is one word long. It specifies the number of different peripheral types that are in the system.

- For each peripheral type, there is a 2-byte entry in the table that indicates the peripheral type and the number of physical devices of that type that are attached to the SCSI subsystem. Each physical-device entry has the following format:

Number of Devices	Peripheral Type

  If a peripheral type is not present in the system, it does not have an entry in the table. However, if a peripheral type is present in the system but is not attached to the SCSI subsystem, the device count is 0. This helps a controlling program establish the SCSI configuration across all SCSI subsystems.

- The entire length of this table (in bytes) can be obtained through the Read Device Parameters function (hex 03).
- The value of the Return Code field is hex 0000.

Service-Specific Input

Size	Offset	Description
Word	10H	Reserved
DWord	12H	Logical pointer to table-build area
Word	16H	Reserved

Service-Specific Output

The following table is built at the specified address:

Size	Offset	Description
Word	0000H	Number of peripheral-device-type entries (*m*)
Word	0002H	Peripheral-device-type entry 1
Word	0004H	Peripheral-device-type entry 2
Word	(*m*×2)H	Peripheral-device-type entry *m*

Figure 6-24. Device Configuration Table Format

0CH—Return Interrupting Logical ID

- This function returns the interrupt-pending status of the SCSI subsystem. If there is an interrupt pending, this function also returns the logical ID that is associated with the interrupt. If no interrupt is pending, the Interrupting Logical ID field is undefined.
- The value of the Return Code field is hex 0000.

Service-Specific Input

Size	Offset	Description
Word	16H	Reserved

Service-Specific Output

Size	Offset	Description
Byte	10H	Interrupt status Bits 7 to 1 - Reserved Bit 0 - Status = 0 - Interrupt not pending = 1 - Interrupt pending
Word	12H	Interrupting logical ID; valid only if an interrupt is pending

10H—Set SCSI Subsystem DMA Pacing Factor

- This function programs the SCSI subsystem with the specified pacing value. All devices that are attached to the SCSI subsystem are affected.
- The pacing value is expressed as a percentage in the range from 25 to 100, inclusive.
- BIOS does not check the range of the pacing value. Using a value outside the specified range can cause an adapter error.
- All devices must be inactive at the time of this request.
- All return-code values are possible.

Service-Specific Input

Size	Offset	Description
Byte	10H	Pacing factor (from 25% to 100%)
Word	16H	Reserved

Service-Specific Output

Size	Offset	Description
DWord	28H	Time to wait before resuming request, in microseconds
Byte	32H	Status Bits 7 to 2 - Reserved Bit 1 - Command-complete status indicator (see "Programming Considerations" on page 6-ID17-9) = 0 - Command-complete status not required = 1 - Command-complete status should be requested for more information Bit 0 - Reserved

11H—Return Device DMA Pacing Factor

- This function returns the current pacing factor for the SCSI subsystem that is specified by the logical ID.
- The pacing value is expressed as a percentage in the range from 25 to 100, inclusive.
- The value of the Return Code field is hex 0000.

Service-Specific Input

Size	Offset	Description
Word	16H	Reserved

Service-Specific Output

Size	Offset	Description
Byte	10H	Current pacing factor

12H—Transfer SCB

- This function programs the SCSI subsystem to process the subsystem control block (SCB) that is pointed to by the Physical Pointer to SCB field.
- ABIOS does not check the validity of the SCB.
- The SCB chain header has the following format:

Size	Offset	Description
Word	00H	Reserved (set to 0)
DWord	02H	Logical pointer to next SCB header in chain, or chain-ending indicator (0)
Word	06H	Reserved
Word	08H	Reserved
DWord	0AH	Logical pointer to termination status block (TSB) that is associated with this SCB
Word	0EH	Reserved

- A logical pointer of 0 ends the SCB chain.
- The chain must have an ending.
- If the Logical Pointer to SCB Chain Header field (offset hex 16 on input) is set to 0, ABIOS does not initiate the SCB transfer, and the Return Code field is set to hex 0000 (Operation Successfully Completed).
- The chain header must immediately precede the SCB.
- All return-code values are possible.

- See the "Chain Example" diagram in "Transfer SCB Request Block" in "Device ID 02H—Fixed Disk."

Service-Specific Input

Size	Offset	Description
DWord	10H	Physical pointer to SCB
Word	14H	Reserved
DWord	16H	Logical pointer to SCB chain header
Word	1CH	Reserved
Word	26H	Reserved
Word	2CH	Reserved
Byte	2EH	Flags Bits 7 to 1 - Reserved (set to 0) Bit 0 - Length = 0 - Normal-length SCB = 1 - Long SCB

Service-Specific Output

Size	Offset	Description
DWord	28H	Time to wait before resuming request, in microseconds
Byte	32H	Status Bits 7 to 2 - Reserved Bit 1 - Command-complete status indicator (see "Programming Considerations" on page 6-ID17-9) = 0 - Command-complete status not required = 1 - Command-complete status should be requested for more information Bit 0 - Reserved

13H—Reserved

Return Codes

Return codes are returned at offset hex 0C.

Value	Description
0000H	Operation Successfully Completed
0001H	Incomplete – Stage on Interrupt
0002H	Incomplete – Stage on Time
0005H	Incomplete – Not My Interrupt, Stage on Interrupt
8000H	Device Busy, Request Refused
8003H	LID Not Associated with This Adapter
8100H	Device Busy, Request Refused
9000H	Operation Ended in Error
900CH	Command Completed with Failure
900EH	Command Error
900FH	Sequence Error
9020H	Bad Controller
9100H	Operation Ended in Error
910CH	Command Completed with Failure
910EH	Command Error
910FH	Sequence Error
9120H	Bad Controller
A000H	Operation Ended in Time-Out
A020H	Bad Controller
A100H	Operation Ended in Time-Out
A120H	Bad Controller
B000H	Time-Out Routine Failed
B020H	Bad Controller
B100H	Time-Out Routine Failed
B120H	Bad Controller
C000H	Invalid Logical ID (ABIOS transfer convention only)
C001H	Invalid Function
C003H	Invalid Unit Number
C004H	Invalid Request-Block Length
C005H	Invalid Parameter
FFFFH	Return Code Not Valid

Figure 6-25. SCSI Subsystem Interface Return Codes

Programming Considerations

- Some error conditions do not result in a termination status block (TSB) being automatically stored. When a TSB is not stored, bit 1 of the Status field (offset hex 32 on output) is set to 1 to indicate that a Get Command-Complete Status SCB should be built and sent through the Transfer SCB function (hex 12) for more information. If the command does not result in an error, this bit is set to 0. The controlling program can still request command-complete status. This bit is defined for return codes hex 9000 to hex BFFF only.
- The SCSI-adapter device ID and the DMA device ID must be initialized before the SCSI peripheral-type device ID is initialized.
- BIOS calls must not be made in an ABIOS environment.
- During ABIOS initialization, the following BIOS services are used:
  - Interrupt 13H (the BIOS hardware interrupt service must be in place)
  - Interrupt 15H.

Notes:

Device ID 18H—SCSI Peripheral Type

Functions

Logical IDs are allocated to specific small computer system interface (SCSI) devices on demand. Through the Allocate SCSI Peripheral Device function (hex 15), the device driver specifies a SCSI peripheral type, a removable-media indicator, and a relative unit number. If a device exists and is unallocated, ABIOS assigns a logical ID to that device. The controlling program can then use the logical ID to make device requests through the SCSI adapter. The structure for input to the SCSI adapter is the subsystem control block (SCB). The Transfer SCB function (hex 12) transfers SCBs to the adapter.

The Deallocate SCSI Peripheral Device function (hex 14) removes the association between a specific device and a logical ID. The controlling program uses this function to release a device that it does not support. The program issues the Deallocate SCSI Peripheral Device function (hex 14), allocates another device to the logical ID, and determines whether it supports that device. Only one device can be allocated to a logical ID, and only one logical ID can be associated with a device at any time.

Note: All reserved input fields must be set to 0.

00H—Default Interrupt Handler

- This is a standard ABIOS function call. It requires device allocation before it can be executed. If this function is called without device allocation, the return code is hex 0005, not hex 8003.
- This function is described fully in "Request Block" in the "Transfer Conventions" section.

01H—Return Logical ID Parameters

- This is a standard ABIOS function call. If a device has not been allocated to this logical ID, the Hardware Interrupt Level field (offset hex 10) and the Arbitration Level field (offset hex 11) are set to hex 0FD to indicate that the information is not currently available. When an allocation is successful, this function is called to obtain the values for the Hardware Interrupt Level field and the Arbitration Level field.
- The value of the Return Code field is hex 0000.
- This function is described fully in "Request Block" in the "Transfer Conventions" section.

02H—Reserved

03H—Read Device Parameters

- This function returns information about the SCSI device.
- This function requires device allocation before it can be executed.
- The device-power status and device-defective status are set during POST and are returned at offset hex 14 of the request block. ABIOS does not attempt to reset these flags or make any determinations on the basis of the state of these flags.
- All return-code values are possible.

Service-Specific Input

Size	Offset	Description
Word	28H	Reserved

Service-Specific Output

Size	Offset	Description
Word	10H	Reserved
Byte	12H	SCB architecture card compatibility level
Byte	13H	Adapter index (0 based)
Word	14H	Device flags Bits 15 to 2 - Reserved Bit 1 - Device power during POST = 0 - Power on = 1 - Power off Bit 0 - Device-defective error during POST = 0 - Device not defective = 1 - Device defective
Byte	16H	Logical unit number (LUN)
Byte	17H	Physical unit number (PUN)

04H—Set Device Parameters (Reserved)

05H—Reset/Initialize

- This function issues a Reset command to the physical device. All logical units on the device are affected.
- This function requires device allocation before it can be executed.
- All return-code values are possible.

Service-Specific Input

Size	Offset	Description
Word	16H	Reserved

Service-Specific Output

Size	Offset	Description
DWord	28H	Time to wait before resuming request, in microseconds
Byte	32H	Status Bits 7 to 2 - Reserved Bit 1 - Command-complete status indicator (see "Programming Considerations" on page 6-ID18-11) = 0 - Command-complete status not required = 1 - Command-complete status should be requested for more information Bit 0 - Reserved

06H—Enable (Reserved)

07H—Disable (Reserved)

08H—Read (Reserved)

09H—Write (Reserved)

0AH—Additional Data Transfer (Reserved)

0BH to 0FH—Reserved

10H—Set Device Time-Out

- This function programs the adapter time-out value for a device. It does not affect any other device on the adapter.
- This function requires device allocation before it can be executed.
- If the time-out count is set to 0, the adapter is programmed not to time out for commands to the device. This allows for operations that can take longer than the 127-minute maximum that the interface provides. On all subsequent stage-on-interrupt returns, the Time-Out Value field is set to 0 to show that there is no time-out for the operation. The controlling program must call the time-out entry point to terminate the request, if the interrupt has not occurred after an appropriate length of time.
- All return-code values are possible.

Service-Specific Input

Size	Offset	Description
Byte	10H	Time-out value Bit 7 - Granularity = 0 - Time-out count is expressed in seconds = 1 - Time-out count is expressed in minutes Bits 6 to 0 - Time-out count
Byte	11H	Reserved
Word	16H	Reserved

Service-Specific Output

Size	Offset	Description
DWord	28H	Time to wait before resuming request, in microseconds
Byte	32H	Status Bits 7 to 2 - Reserved Bit 1 - Command-complete status indicator (see "Programming Considerations" on page 6-ID18-11) = 0 - Command-complete status not required = 1 - Command-complete status should be requested for more information Bit 0 - Reserved

11H—Read Device Time-Out

- This function returns the current time-out value for a device. It does not affect any other device on the adapter.
- This function requires device allocation before it can be executed.
- The possible values of the Return Code field are hex 0000 and 8003.

Service-Specific Input

Size	Offset	Description
Byte	11H	Reserved
Word	16H	Reserved

Service-Specific Output

Size	Offset	Description
Byte	10H	Time-out value Bit 7 - Granularity = 0 - Time-out count is expressed in seconds = 1 - Time-out count is expressed in minutes Bits 6 to 0 - Time-out count

12H—Transfer SCB

- This function transfers the subsystem control block (SCB) that is pointed to by the Physical Pointer to SCB field to the adapter that controls the specified device.
- This function requires device allocation before it can be executed.
- ABIOS does not check the validity of the SCBs.
- The SCB chain header has the following format:

Size	Offset	Description
Word	00H	Reserved (set to 0)
DWord	02H	Logical pointer to next SCB header in chain, or chain-ending indicator (0)
Word	06H	Reserved
Word	08H	Reserved
DWord	0AH	Logical pointer to termination status block (TSB) that is associated with this SCB
Word	0EH	Reserved

- A logical pointer of 0 ends the SCB chain.
- The chain must have an ending.
- If the Logical Pointer to SCB Chain Header field (offset hex 16 on input) is set to 0, ABIOS does not initiate the SCB transfer, and the Return Code field is set to hex 0000 (Operation Successfully Completed).
- The chain header must immediately precede the SCB.
- All return-code values are possible.
- See the "Chain Example" diagram in "Transfer SCB Request Block" in "Device ID 02H—Fixed Disk."

Note: The Read and Write SCB commands are the primary interface to devices such as IBM-type fixed disk drives. Do not use the

Send Other SCSI command to read from or write to these devices, because it can delete data from the disk.

Service-Specific Input

Size	Offset	Description
DWord	10H	Physical pointer to SCB
Word	14H	Reserved
DWord	16H	Logical pointer to SCB chain header
Word	1CH	Reserved
Word	26H	Reserved
Word	2CH	Reserved
Byte	2EH	Flags Bits 7 to 1 - Reserved Bit 0 - Length = 0 - Normal-length SCB = 1 - Long SCB

Service-Specific Output

Size	Offset	Description
DWord	28H	Time to wait before resuming request, in microseconds
Byte	32H	Status Bits 7 to 2 - Reserved Bit 1 - Command-complete status indicator (see "Programming Considerations" on page 6-ID18-11) = 0 - Command-complete status not required = 1 - Command-complete status should be requested for more information Bit 0 - Reserved

14H—Deallocate SCSI Peripheral Device

- This function deallocates the SCSI peripheral-device type that is assigned to a logical ID.
- If the device type is not currently allocated, the Return Code field is set to hex 8003 (Device Not Allocated to This Logical ID).
- The possible values of the Return Code field are hex 0000, 8000, and 8003.

Service-Specific Input

Size	Offset	Description
Word	16H	Reserved

Service-Specific Output

Size	Offset	Description
None		

15H—Allocate SCSI Peripheral Device

- This function allocates the *n*th available device of the specified SCSI device type in the request block to a logical ID, where $0 \leq n \leq$ the device count (see the Return Peripheral-Type Count function (hex 16)). When *n* is 0, the next available device of the specified type is allocated. Any other value of *n* causes the *n*th device of the specified peripheral type to be allocated.
- This function must be executed, because no device is associated with a logical ID after ABIOS initialization. The controlling program fills in the request-block fields that are needed to describe the requested device. These fields are:
  - Peripheral-device type (offset hex 10)
  - Removable-media indicator (bit 7 of offset hex 11)
  - *n*th device of this type (offset hex 12).
- Some BIOS functions can be called without a device being allocated to a logical ID. (See Figure 6-27 on page 6-ID18-11 to determine which functions require device allocation.)
- Only one SCSI device can be allocated to a logical ID.
- If the requested device does not exist, the Return Code field is set to hex 8002 (Device Not Available).
- If a device is already allocated to a logical ID, the Return Code field is set to hex 8004 (A Device Is Already Allocated to This Logical ID).
- If the requested device exists but is currently allocated to another logical ID, the Return Code field is set to hex 8005 (Requested Device Allocated to Another Logical ID).
- If the requested device exists, but its associated arbitration level could not be allocated, the Return Code field is set to hex 8006 (Arbitration Level Could Not Be Allocated for Requested Device).
- The possible values of the Return Code field are hex 0000, 8002, 8004, 8005, and 8006.

Service-Specific Input

Size	Offset	Description
Byte	10H	Peripheral-device type
Byte	11H	Device-type flags Bit 7 - Removable-media indicator = 0 - Device media is not removable = 1 - Device media is removable Bits 6 to 0 - Reserved
Word	12H	*n*th device of this type
Word	16H	Reserved

Service-Specific Output

Size	Offset	Description
None		

16H—Return Peripheral-Type Count

- This function returns the number of SCSI devices that are available for the specified type. The controlling program fills in the request-block fields that are needed to describe the requested device. These fields are:
  - Peripheral-device type (offset hex 10)
  - Removable-media indicator (bit 7 of offset hex 11).
- A count of 0 indicates that no devices of the specified type were found.
- The value of the Return Code field is hex 0000.

Service-Specific Input

Size	Offset	Description
Byte	10H	Peripheral-device type
Byte	11H	Device-type flags Bit 7 - Removable-media indicator = 0 - Device media is not removable = 1 - Device media is removable Bits 6 to 0 - Reserved
Word	16H	Reserved

Service-Specific Output

Size	Offset	Description
Word	14H	Count of requested device type

17H—Abort

- This function issues an Abort command to the specified device.
- This function requires allocation of the device to the logical ID.
- This function can be issued regardless of whether another request is outstanding.
- To abort an existing request, a new request block for the Abort function is created; the pending request block must be kept active. If the Abort command cannot be issued, the Return Code field is set to hex 8000 (Device Busy, Request Refused). If the Return Code field is set to hex FFFF or 8000 and an interrupt occurs for the device, the pending request block must be passed to BIOS to enable BIOS to handle the interrupt. If the Return Code field is not set to hex FFFF or 8000, the request block is

used to answer the interrupt, and the previous request block is considered to be complete.

- All return-code values are possible.

Service-Specific Input

Size	Offset	Description
Word	16H	Reserved

Service-Specific Output

Size	Offset	Description
DWord	28H	Time to wait before resuming request, in microseconds
Byte	32H	Status Bits 7 to 2 - Reserved Bit 1 - Command-complete status indicator (see "Programming Considerations" on page 6-ID18-11) = 0 - Command-complete status not required = 1 - Command-complete status should be requested for more information Bit 0 - Reserved

Return Codes

Return codes are returned at offset hex 0C.

Value	Description
0000H	Operation Successfully Completed
0001H	Incomplete – Stage on Interrupt
0002H	Incomplete – Stage on Time
0005H	Incomplete – Not My Interrupt, Stage on Interrupt
8000H	Device Busy, Request Refused
8002H	Device Not Available
8003H	Device Not Allocated to This Logical ID
8004H	A Device Is Already Allocated to This Logical ID
8005H	Requested Device Allocated to Another Logical ID
8006H	Arbitration Level Could Not Be Allocated for Requested Device
8100H	Device Busy, Request Refused
9000H	Operation Ended in Error
900CH	Command Failure
900EH	Invalid Adapter Command/Parameter
900FH	Sequence Error
9020H	Bad Controller
9100H	Operation Ended in Error
910CH	Command Failure
910EH	Unsupported Adapter Command/Parameter
910FH	Sequence Error
9120H	Bad Controller
A000H	Operation Ended in Time-Out, No Additional Status
A080H	Operation Ended in Time-Out
A100H	Operation Ended in Time-Out, No Additional Status
A180H	Operation Ended in Time-Out
B000H	Time-Out Routine Failed
B00CH	Time-Out Routine Completed with Failure
B00EH	Time-Out Routine Command Error
B00FH	Time-Out Routine Sequence Error
B020H	Bad Controller
B080H	Time-Out
B100H	Time-Out Routine Failed
B10CH	Time-Out Routine Completed with Failure
B10EH	Time-Out Routine Command Error
B10FH	Time-Out Routine Sequence Error
B120H	Bad Controller
B180H	Time-Out
C000H	Invalid Logical ID (ABIOS transfer convention only)
C001H	Invalid Function
C003H	Invalid Unit Number
C004H	Invalid Request-Block Length
C005H	Invalid ABIOS Parameter
C008H	Cache Buffer Not Supported
FFFFH	Return Code Not Valid

Figure 6-26. SCSI Peripheral Type Return Codes

Programming Considerations

- The following figure identifies the calling restrictions that are on the SCSI peripheral-type functions.

Function Number	Requires Device Allocation
0000H	Yes*
0001H	No**
0003H	Yes
0005H	Yes
0010H	Yes
0011H	Yes
0012H	Yes
0014H	Yes
0015H	No
0016H	No
0017H	Yes

* Returns hex 0005 (Not My Interrupt – Stage on Interrupt) if device is not allocated

** Returns hex FD for arbitration and interrupt level if device is not allocated

Figure 6-27. Function Restrictions

- Some error conditions do not result in a termination status block (TSB) being automatically stored. When a TSB is not stored, bit 1 of the Status field (offset hex 32 on output) is set to 1 to indicate that a Get Command-Complete Status SCB should be built and sent through the Transfer SCB function (hex 12) for more information. If the command does not result in an error, this bit is set to 0. The controlling program can still request command-complete status. This bit is defined for return codes hex 9000 to hex BFFF only.
- The SCSI-adapter device ID and the DMA device ID must be initialized before the SCSI peripheral-type device ID is initialized.
- BIOS calls must not be made in an ABIOS environment.
- The activity LED is defined for fixed-disk devices only. If a device is a fixed disk drive (512-byte blocks, nonremovable media, peripheral type 0), the activity LED is used.
- During ABIOS initialization, the following BIOS services are used:
  - Interrupt 13H (the BIOS hardware interrupt service must be in place)
  - Interrupt 15H.

Notes: