Table 99. Scan High or Equal Result Phase *(Continued)*

Byte 6	Number of Data Bytes in Sector

3.5.1.14 Scan Low or Equal Command

3.5.1.14.1 Command Phase

Table 100. Scan Low or Equal Command Phase

	7	6	5	4	3	2	1	0
Byte 0	MT	MFM	SK	1	1	0	0	1
Byte 1	0	0	0	0	0	HD	US	US
Byte 2	Track Number							
Byte 3	Head Address							
Byte 4	Sector Number							
Byte 5	Number of Data Bytes in Sector							
Byte 6	End-of-Track							
Byte 7	Gap Length							
Byte 8	Scan Test							

3.5.1.14.2 Execution Phase

Data transferred from the system to the controller is compared to data read from the disk.

3.5.1.14.3 Result Phase

Table 101. Scan Low or Equal Result Phase

Byte 0	Status Register 0
Byte 1	Status Register 1
Byte 2	Status Register 2
Byte 3	Track Number
Byte 4	Head Address
Byte 5	Sector Number
Byte 6	Number of Data Bytes in Sector

3.5.1.15 Seek Command

3.5.1.15.1 Command Phase

Table 102. Seek Command Phase

	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	1	1	1	1
Byte 1	0	0	0	0	0	HD	US	US

Table 102. Seek Command Phase *(Continued)*

	7	6	5	4	3	2	1	0
Byte 2	New Track Number after Seek							

3.5.1.15.2 .Execution Phase

The disk drive head is stepped in or out to a programmable track.

3.5.1.15.3 Result Phase

This command has no result phase.

3.5.1.16 Sense Drive Status Command

3.5.1.16.1 Command Phase

Table 103. Sense Drive Status Command Phase

	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	0	1	0	0
Byte 1	0	0	0	0	0	HD	US	US

3.5.1.16.2 Result Phase

Table 104. Sense Drive Status Result Phase

Byte 0	Status Register 3

3.5.1.17 Sense Interrupt Status

3.5.1.17.1 Command Phase

Table 105. Sense Interrupt Status Command Phase

	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	1	0	0	0

3.5.1.17.2 Execution Phase

The status of the interrupt is reported.

3.5.1.17.3 Result Phase

Table 106. Sense Interrupt Status Result Phase

Byte 0	Status Register 0
Byte 1	Present Track Number

3.5.1.18 Specify Command

3.5.1.18.1 Command Phase

Table 107. Specify Command Phase

	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	0	0	1	1
Byte 1	Stepping Rate Time				Head Unload Time			
Byte 2	Head Load Time							DMA

3.5.1.18.2 Execution Phase

Internal registers are written.

3.5.1.18.3 Result Phase

This command has no result phase.

3.5.1.19 Verify Command

3.5.1.19.1 Command Phase

Table 108. Verify Command Phase

	7	6	5	4	3	2	1	0
Byte 0	MT	MFM	SK	1	0	1	1	0
Byte 1	EC	0	0	0	0	HD	US	US
Byte 2	Track Number							
Byte 3	Head Address							
Byte 4	Sector Address							
Byte 5	Sector Size							
Byte 6	End-of-Track							
Byte 7	Gap Length							
Byte 8	Byte Transfer Control							

3.5.1.19.2 Execution Phase

Data is read from the disk but not transferred to the system.

3.5.1.19.3 Result Phase

Table 109. Verify Result Phase

Byte 0	Status Register 0
Byte 1	Status Register 1
Byte 2	Status Register 2
Byte 3	Track Number
Byte 4	Head Address
Byte 5	Sector Address
Byte 6	Sector Size

3.5.1.20 Version Command

3.5.1.20.1 Command Phase

Table 110. Version Command Phase

	7	6	5	4	3	2	1	0
Byte 0	0	0	0	1	0	0	0	0

3.5.1.20.2 Result Phase

Table 111. Version Result Phase

	7	6	5	4	3	2	1	0
Byte 0	1	0	0	X	0	0	0	0

X can be a 1 or a 0.

3.5.1.21 Write Data Command

3.5.1.21.1 Command Phase

Table 112. Write Data Command Phase

	7	6	5	4	3	2	1	0
Byte 0	MT	MFM	0	0	0	1	0	1
Byte 1	0	0	0	0	0	HD	US	US
Byte 2	Track Address							
Byte 3	Head Address							
Byte 4	Sector Number							
Byte 5	Number of Data Bytes in Sector							

Table 112. Write Data Command Phase *(Continued)*

	7	6	5	4	3	2	1	0
Byte 6	End-of-Track							
Byte 7	Gap Length							
Byte 8	Data Length							

3.5.1.21.2 Execution Phase

Data is transferred from the system to the controller via DMA or non-DMA modes and written to the disk.

3.5.1.21.3 Result Phase

Table 113. Write Data Result Phase

Byte 0	Status Register 0
Byte 1	Status Register 1
Byte 2	Status Register 2
Byte 3	Track Number
Byte 4	Head Address
Byte 5	Sector Number
Byte 6	Number of Data Bytes in Sector

3.5.1.22 Write Deleted Data Command

3.5.1.22.1 Command Phase

Table 114. Write Deleted Command Phase

	7	6	5	4	3	2	1	0
Byte 0	MT	MFM	0	0	1	0	0	1
Byte 1	0	0	0	0	0	HD	US	US
Byte 2	Track Number							
Byte 3	Head Address							
Byte 4	Sector Number							
Byte 5	Number of Data Bytes in Sector							
Byte 6	End of Track							
Byte 7	Gap Length							
Byte 8	Data Length							

3.5.1.22.2 Execution Phase

Data is transferred from the system to the controller via DMA or non-DMA modes and written to the disk.

3.5.1.22.3 Result Phase

Table 115. Write Deleted Result Phase

Byte 0	Status Register 0
Byte 1	Status Register 1
Byte 2	Status Register 2
Byte 3	Track Number
Byte 4	Head Address
Byte 5	Sector Number
Byte 6	Number of Data Bytes in Sector

3.5.1.23 Invalid Command Status

3.5.1.23.1 Result Phase. The following status byte is returned to the system microprocessor when an invalid command has been received.

Table 116. Invalid Command Status Register

Byte 0	Status Register 0

Bits 6 and 7 in status register 0 are used to indicate command status. When an invalid command is processed, this information is returned to the system microprocessor in the invalid command status byte.

3.5.2 Command Status Registers Provided During Result Phase

This section provides definitions for status registers 0 through 3. These registers are read through the Data Register (FIFO) during the result phase of a floppy disk operation.

Requirements:

3–14. For the device described herein, the Command Status Registers described in Section 3.5.2.1, “Status Register 0 (ST0),” on page 56 through Section 3.5.2.4, “Status Register 3 (ST3),” on page 58 must be implemented as specified.

3.5.2.1 Status Register 0 (ST0)

Table 117. Status Register 0

Bit	Function	Value when OF passes to SW	Value on Software Reset
7, 6	**Interrupt Code:**These bits indicate the command interrupt status as given in Table 118 on page 57.	0	0
5	**Seek End:** This bit is set to 1 when the diskette drive completes the Seek or Recalibrate command, or a read or write operation with an implied Seek command.	0	0

Table 117. Status Register 0 ***(Continued)***

Bit	Function	Value when OF passes to SW	Value on Software Reset
4	**Equipment Check:** This bit is set to 1 if the '-track 0' signal fails to occur after the Recalibrate command is issued or Relative Seek command to step outward beyond track 0	0	0
3	**Reserved:** This bit is always set to 0	0	0
2	**Head Select:** This bit indicates the state of the '-head select' signal after the command was performed. When set to 1, head 1 was selected: when set to 0. head O was selected.	0	0
1, 0	**Drive Select:** These bits indicate the drive that was selected upon command completion	0	0

Table 118. Encodings for the Interrupt Code field (bits 7,6 of ST0)

Bits 7 6	Function
0 0	Normal Termination of Command
0 1	Abnormal Termination of Command
10	Invalid Command Issued
11	Internal drive ready status changed state during the drive polling mode. Only occurs after a hardware or software reset.

Table 119. Drive Select Bits

Bits 1 0	Function
0 0	Drive 0
0 1	Drive 1
1 0	Drive 2
1 1	Drive 3

3.5.2.2 Status Register 1 (ST1)

Table 120. Status Register 1

Bit	Function
7	**End-of-Track**: This bit is set to 1 when the controller tries to gain access to a sector beyond the final sector of a track
6	**Reserved:** This bit is always set to 0.
5	**Cyclic Redundancy Check (CRC) Error:** This bit is set to 1 when a CRC error is detected in the ID or data field.
4	**Overrun/Underrun Error:** This bit is set to 1 if the system does not service the diskette drive controller within an adequate period of time during data transfers.
3	**Reserved:** This bit is always set to 0.

Table 120. Status Register 1 *(Continued)*

Bit	Function
2	**No Data:** This bit is set to 1 when • The controller cannot find the sector specified in the ID register during the execution of a Read Data, Read Deleted Data, or Read ID or Read Track command. • The controller cannot read the ID field without an error during the execution of a Read ID command • The starting sector cannot be found during the execution of a Read Track command
1	**Not Writable:** This bit is set to 1 when the '-write-protect' signal is active during a Write Data, Write Deleted Data, or Format Track command.
0	**Missing Address Mark:** This bit is set to 1 if the controller cannot detect an address mark. When this occurs, bit 0 of Status Register 2 indicates whether the missing address mark is an ID-address mark or a data-address mark.

3.5.2.3 Status Register 2 (ST2)

Table 121. Status Register 2

Bit	Function
7	**Reserved:** This bit is always set to 0.
6	**Control Mark:** This bit is set to 1 when the controller encounters a sector that has a deleted data-address mark during a Read Data or a Read Deleted Data encounters a data address mark.
5	**CRC Error in Data Field:** This bit is set to 1 if the controller detects an error in the data.
4	**Wrong Track:** This bit is set to 1 when the track number on the media is different from the track number issued by the command. When this occurs, bit 2 of Status Register 1 is also set to 1.
3	**Scan Equal Hit:** This bit is set to 1 during the Scan command when the conditions for Equal are satisfied.
2	**Scan Not Satisfied:** This bit is set to 1 during the Scan Command when the scan conditions are not satisfied.
1	**Bad Track:** This bit is set to 1 when the track number on the media is hex FF and the track number value stored in the IDregister is not hex FF. When this occurs, bit 2 of Status Register 1 is also set to 1.
0	**Missing Address Mark in Data Field:** This bit is set to 1 when the controller cannot find a data-address mark. This bit is set to 0 when an ID-address mark cannot be found. Bit 0 in Status Register 0 is also set if either address mark cannot be found.

3.5.2.4 Status Register 3 (ST3)

Table 122. Status Register 3

Bit	Function
7	**Reserved:** This bit is always set to 0
6	**Write Protect:** This bit indicates the status of the '-write-protect' signal from the diskette drive. When this bit is set to 1, the'-write-protect' signal is active.
5	**Reserved:** This bit is always set to 1.
4	**Track 0:** This bit indicates the status of the '-track O' signal from the diskette drive. When this bit is set to 1, the '-track 0 signal is active.
3	**Reserved:** This bit is always set to 1.

Table 122. Status Register 3 *(Continued)*

Bit	Function
2	**Head Address:** This bit indicates the status of the '-head 1 select' signal from the diskette drive. When this bit is set to 1, the '-head 1 select' signal is active.
1, 0	**Drive Select:** These bits indicate the current selected drive.

3.6 Media Sense

Requirements:

3–15. The disk drive must be configured in a manner such that the Disk in Place signal, on the drive, is functionally a state indicator of the presence of media. The signal must assert an active level when a disk is present in the drive and an inactive level when there is no disk in the drive.

3.7 Floppy Drive Signal Connector Pin Assignment

Table 123. Signal Connector Pin Assignment

PIN	Signal Description	PIN	Signal Description
1	AUTO EJECT *	2	N. C.
3	Key	4	HD OUT
5	Ground	6	DISK IN PLACE *
7	Ground	8	INDEX
9	Ground	10	DRIVE SELECT 0
11	Ground	12	DRIVE SELECT 1
13	Ground	14	N. C.
15	Ground	16	MOTOR ON
17	Ground	18	DIRECTION
19	Ground	20	STEP
21	Ground	22	WRITE DATA
23	Ground	24	WRITE GATE
25	Ground	26	TRACK 00
27	Ground	28	WRITE PROTECT
29	Ground	30	READ DATA
31	Ground	32	HEAD SELECT
33	Ground	34	DISK CHANGE

Hardware Implementation Note: Signal pins AUTOEJECT, signal pin 1, DISK IN PLACE, signal pin 6, and HD OUT, signal pin 4, must be gated by Drive Select, if the drive is used in platforms that support multiple drives.

3.8 References

1. *National Semiconductor PC87308VUL SuperI/O Enhanced Sidewinder Lite Plug and Play Compatible Chip, with a Floppy Disk Controller, a Keyboard Controller, a Real-Time Clock, Two UARTs, Full Infrared Support and an IEEE1284 Parallel Port* [24].

2. *National Semiconductor PC87332VLJ (3.3V/5V) and PC87332VLJ-5 (5V) (SuperI/O III Premium Green) with Floppy Disk Controller, Dual UARTs, IEEE 1284 Parallel Port, and IDE Interface* [26].

Legacy Interrupt Controller

4

4.1 Overview and General Requirements

Table 124 on page 61 is an excerpt from *PowerPC Microprocessor Common Hardware Reference Platform: A System Architecture*, showing the minimum requirements of the Legacy Interrupt Controller on CHRP platforms.

Table 124. Minimum Platform Requirements for the Legacy Interrupt Controller

Subsystem	Specification	Portable	Personal	Server	Description
Interrupt Controller	8259 tree	R	R	R	8259 required for ISA compatibility.

The ISA Legacy Interrupt Controller is actually a set of two programmable interrupt controllers (PICs) referred to as PIC1 (master) and PIC2 (slave) in this book. Their interrupt lines may be set as edge or level sensitive. There are a number of possible modes for determining the order in which the priority of interrupts is resolved. The two controllers are cascaded in a master/slave relationship as shown in Figure 3.

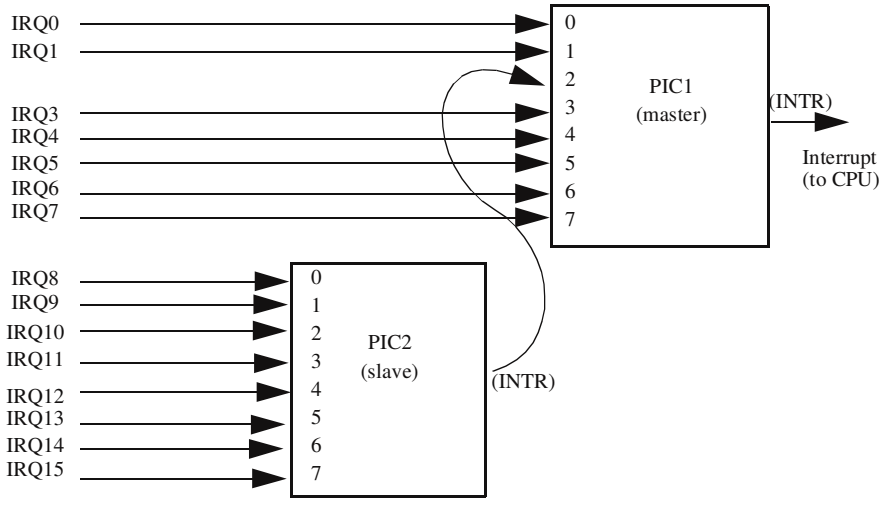

Figure 3. PIC1 (master) and PIC2 (slave) Interconnection

Requirements:

4–1. The Legacy Interrupt Controller must be represented by the Open Firmware properties “name=interrupt-controller, device_type=interrupt-controller, compatible=chrp,iic”.

4–2. The Legacy Interrupt Controller registers specified herein must be implemented precisely as defined, including the following characteristics and bit definitions:

- **a.** Table 129, “Initialization Command Word 1 Register (ICW1) Characteristics,” on page 68
- **b.** Table 130, “Initialization Command Word 1 Register (ICW1),” on page 68
- **c.** Table 131, “Initialization Command Word 2 Register (ICW2) Characteristics,” on page 68
- **d.** Table 132, “Initialization Command Word 2 Register (ICW2),” on page 68
- **e.** Table 133, “Initialization Command Word 3 Register (ICW3) Characteristics,” on page 69.
- **f.** Table 134, “Initialization Command Word 3 Register (ICW3),” on page 69
- **g.** Table 135, “Initialization Command Word 4 Register (ICW4) Characteristics,” on page 69
- **h.** Table 136, “Initialization Command Word 4 Register (ICW4),” on page 70
- **i.** Table 137, “Operation Command Word 1 Register (OCW1) Characteristics,” on page 70
- **j.** Table 138, “Operation Command Word 1 Register (OCW1),” on page 70
- **k.** Table 140, “Operation Command Word 2 Register (OCW2) Characteristics,” on page 71
- **l.** Table 141, “Operation Command Word 2 Register (OCW2),” on page 71
- **m.** Table 144, “Operation Command Word 3Register (OCW3) Characteristics,” on page 72
- **n.** Table 145, “Operation Command Word 3 Register (OCW3),” on page 73
- **o.** Table 149, “Interrupt Request Register (IRR) Characteristics,” on page 74
- **p.** Table 143, “Interrupt Level Select Encodings,” on page 72
- **q.** Table 151, “In-Service Register (ISR) Characteristics,” on page 75
- **r.** Table 143, “Interrupt Level Select Encodings,” on page 72
- **s.** Table 153, “Edge/Level Interrupt Control Register 1 (ELI1) Characteristics,” on page 76
- **t.** Table 154, “Edge/Level Interrupt Control 1 Register (ELI1),” on page 76
- **u.** Table 155, “Edge/Level Interrupt Control 2 Register (ELI2) Characteristics,” on page 76
- **v.** Table 156, “Edge/Level Interrupt Control 2 Register (ELI2),” on page 76

4–3. This device must be addressed purely with ISA I/O addresses.

Hardware and Software Implementation Note: The first 256 bytes of I/O space is not aliased (that is, all of the bits of the address are decoded).

4.2 Open Firmware Requirements

Requirements:

4–4. The registers shown in Table 125 on page 63 must all be present in the device defined in this chapter. The Open Firmware “reg” property information given in Table 125 on page 63 must be provided in the Open Firmware device tree for this device.

4–5. This device must be represented as an ISA device and a child of the ISA bridge node in the Open Firmware device tree.

4–6. This device must be represented in the OF device tree as specified in the *CHRP ISA Interrupt Controller Device Binding to: IEEE 1275-1994 Standard for Boot (Initialization, Configuration) Firmware.*

4–7. All the reg property fields specified in table Table 125 on page 63 must comply with the format given in the ISA (or PCI) bus binding to IEEE Std 1275-1994, as modified by the CHRP binding to IEEE Std 1275-1994.

4–8. The following events must occur during the OF initialization sequence:

a. The level sense circuit must be set to the Edge-sensitive mode (following the initialization procedure, interrupts are generated by a low-to-high transition high level on the interrupt request input.)

b. IRQ7 must be assigned priority 7.

c. PIC2 (slave) mode address must be set to 7.

d. The Special Mask mode must be cleared and the controller must be set to read the Interrupt Request register.

e. All Legacy Interrupt Controller registers must be initialized to the values indicated in the “Value in Register When OF Passes Control to OS” column of the characteristics table corresponding to the specific register.

4–9. PIC1 (master) interrupt controller must be initialized before PIC2 (slave) interrupt controller. Failure to do so will cause unexpected results.

A synopsis of the registers for the ISA Interrupt Controller is given in Table 125. The positions in the reg property (denoted by reg[1], reg[2], etc.) and the value of the size fields are stated in Table 125 on page 63. The format is described in the referenced Open Firmware bindings.

Table 125. Legacy Interrupt Controller Registers

Offset	Register	Open Firmware Reg Property
0	ICW1,OCW2,OCW3, IRR, ISR (PIC1 (master))	reg[1], size=2
1	ICW2,ICW3,ICW4,OCW1 (PIC1 (master))	
0	ICW1,OCW2,OCW3, IRR, ISR (PIC2 (slave))	reg[2], size=2
1	ICW2,ICW3,ICW4,OCW1 (PIC2 (slave))	

Table 125. Legacy Interrupt Controller Registers *(Continued)*

Offset	Register	Open Firmware Reg Property
0	Edge Level Control register ELI1 (PIC1 (master))	reg[3], size=2
1	Edge Level Control register ELI2 (PIC2 (slave))	

4.3 Modes of Operation

The interrupt controller can be programmed to operate in a variety of modes through the Initialization Command Word registers (ICW) and the Operation Command Word registers (OCW).

A typical interrupt request occurs in the following manner:

1. One or more 'interrupt request' lines are set active, causing the corresponding bits in the Interrupt Request register (IRR) to be set to 1.
2. The interrupt controller evaluates the requests and sends an interrupt to the system microprocessor, if appropriate.
3. The system microprocessor receives the interrupt and enters the interrupt handler. The interrupt handler responds by reading 1 byte from the Interrupt Acknowledge address (at an address as specified by the *8259-interrupt-acknowledge* property of the OF device tree). This generates an 'interrupt acknowledge' cycle to the interrupt controller. (Refer to the *8259-interrupt-acknowledge* property in the CHRP System Binding for Open Firmware for information on how to perform the interrupt acknowledge.)
4. The controller prioritizes the unmasked bits in the Interrupt Request register and strobes the bit with the highest priority into the corresponding bit of the In-Service register (ISR). No data is sent to the system microprocessor.

   Hardware and Software Implementation Note: If an interrupt request is not present (for example, the duration of the request was too short), the interrupt controller issues an interrupt 7.

5. The interrupt controller responds by releasing the interrupt vector (as defined in Table 126 on page 65) on the data bus, where it is read by the system microprocessor.
6. The highest priority in-service bit remains set to 1 until the proper End of Interrupt (EOI) command is issued by the interrupt subroutine. If the source of the interrupt request is PIC2 (slave) interrupt controller, the EOI command must be issued twice, once for PIC1 (master) and once for PIC2 (slave). When the in-service bit is set to 1, all other interrupts with the same or lower priority are inhibited; interrupts with a higher priority cause an interrupt, but the interrupt is acknowledged only if the system microprocessor interrupt input has been re-enabled by software.

The EOI command has two forms, specific and non-specific. The controller responds to a Non-specific EOI command by resetting the highest in-service bit of those set. In a mode that uses a fully-nested interrupt structure, the highest in-service bit set is the level that was just acknowledged and serviced. In a mode that can use other than the fully-nested interrupt structure, a Specific EOI command is required to define which in-service bit to reset.

Software Implementation Note: An in-service bit masked by an Interrupt Mask register (see OCW1 for information about the Interrupt Mask register) bit cannot be reset by a Non-specific EOI command when in the Special Mask mode. See-Section 4.4.2.2, “Operation Command Word 2 Register (OCW2),” on page 71 for more information.

Requirements:

4–10. The Interrupt Controller must respond to a Non-specific EOI command by resetting the highest in-service bit of those set.

4–11. The Interrupt Controller must respond to a Specific EOI command by resetting the in-service bit that corresponds to the one specified by the command, except that an in-service bit masked by an Interrupt Mask register bit cannot be reset by a Non-specific EOI command when in the Special Mask mode.

4–12. The Interrupt Vector which is returned by the Interrupt Controller on an Interrupt Acknowledge cycle must be as defined in Table 126 on page 65.

Table 126. Interrupt Vector

Bit	Description
7-3	**Interrupt Vector Base Address:** This field defines the base address of the interrupt vector table which gives the base addresses for the interrupt routines associated with each IRQ on the controller.
2-0	**Interrupt:** Interrupt level that is currently requiring service and has been deemed to have the highest priority by this interrupt controller. IRQ0 is represented by 0b000, IRQ1 by 0b001, and so on.

4.3.1 Fully-Nested Mode

In the Fully-nested mode, interrupts are prioritized from 0 (highest priority) to 7 (lowest priority). This mode is automatically entered after initialization unless another mode has been programmed.

Software Implementation Note: The priorities can be changed by rotating the priorities through OCW2.

4.3.2 Special Fully-Nested Mode

The Special Fully-nested mode is used when the priority in PIC2 (slave) interrupt controller must be preserved. This mode is similar to the normal Nested mode with the following exceptions:

- When PIC2's (slave's) interrupt request is in service, PIC2 (slave) can still generate additional interrupt requests of a higher priority that are recognized by PIC1 (master), and initiate interrupts to the system microprocessor.
- Upon completion of the interrupt service routine, software must send a Non-specific EOI (see 4.4.2.2, "Operation Command Word 2 Register (OCW2)," on page 71) command to PIC2 (slave) and read PIC2 (slave)'s In-Service register to ensure that the interrupt just serviced was the only one generated by PIC2 (slave). If the register is not empty, additional interrupts are pending, and an EOI command must not be sent to PIC1 (master). If the register is empty, a Non-specific EOI command can be sent to PIC1 (master).

The Special Fully-nested mode is selected through ICW4.

4.3.3 Automatic Rotation Mode

The Automatic Rotation mode accommodates multiple devices having the same interrupt priority. After a device is serviced, it is assigned the lowest priority and must wait until all other devices requesting an interrupt are serviced once before the first device is serviced again.

The following example shows the status and priorities of the In-Service register bits before and after bit 4 of the Interrupt Request register is serviced by a Rotation on Non-specific EOI command.

Table 127. Automatic Rotation Mode

In-Service Register Bits	Status Before Service	Priority Before Rotate	Status After Service	Priority After Rotate
7	0	7(Lowest)	0	2
6	1(Pending)	6	1(Pending)	1
5	0	5	0	0
4	1(Pending)	4	0(Serviced)	7
3	0	3	0	6
2	0	2	0	5
1	0	1	0	4
0	0	0(Highest)	0	3

The Automatic Rotation mode is selected by issuing a Rotation on Non-specific EOI command through OCW2.

4.3.4 Specific Rotation Mode

The Specific Rotation mode allows the application programs to change the priority levels by assigning the lowest priority to a specific interrupt level. Once the lowest level priority is selected, all other priority levels change. The following example compares the normal Nested mode to the Specific Rotation mode with bit 5 of the Interrupt Request register set to the lowest priority.

Table 128. Specific Rotation Mode When IRQ5 Has the Lowest Priority

Interrupt Request Register Bits	Nested Mode Priority Level	Specific Rotation Mode Priority Level
7	7 (Lowest)	1
6	6	0 (Highest)
5	5	7 (Lowest)
4	4	6
3	3	5

Table 128. Specific Rotation Mode When IRQ5 Has the Lowest Priority *(Continued)*

Interrupt Request Register Bits	Nested Mode Priority Level	Specific Rotation Mode Priority Level
2	2	4
1	1	3
0	0 (Highest)	2

The Specific Rotation mode is selected by issuing a Rotate on Specific EOI command or a Set Priority command through OCW2.

4.3.5 Special Mask Mode

The Special Mask mode allows application programs to selectively enable and disable any interrupt or combination of interrupts at any time during its execution. The Special Mask mode is selected through OCW3 register. Once the controller is in the Special Mask mode, setting a bit in OCW1 sets a corresponding bit in the Interrupt Mask register (see OCW1 for information about the Interrupt Mask register). Each bit set in the Interrupt Mask register masks the corresponding interrupt channel. Interrupt channels above and below a masked channel are not affected.

4.3.6 Poll Mode

Devices are serviced by software issuing a Poll command through OCW3. The first 'read' pulse following a Poll command is interpreted by the controller as an 'interrupt acknowledge' pulse; the controller sets the appropriate in-service bit and reads the priority level. The byte placed on the data bus during a 'read' pulse is shown in Table 148 on page 73.

4.4 Programming the Interrupt Controller

Before the system can be used, the interrupt controller must be programmed with four sequential writes to the four ICW registers. When PIC1 (master) is written at address reg[1]+0x00 or PIC2 (slave) at address reg[2]+0x00) with data bit 4 set to 1, the command is recognized as a write to ICW1. ICW1 is the first of four writes to the ICW registers that are required to program the interrupt controller. Once the interrupt controller is initialized by the ICW registers, the controller can be further programmed by the OCW registers to operate in other modes.

4.4.1 Initialization Command Word Registers

Requirements:

4–13. For the device described herein, all the registers defined in sections 4.4.1.1 through 4.4.1.4 must be implemented precisely as described herein.

4–14. During the initialization sequence ICW1 must be written before writing ICW2, ICW2 must be written before writing ICW3, and ICW3 must be written before writing ICW4.

4.4.1.1 Initialization Command Word 1 Register (ICW1)

Table 129. Initialization Command Word 1 Register (ICW1) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF Passes Control to OS	SW Initiated Reset Value
1 byte	Sequenced	reg[1],reg[2]	0x00	Write Only	0x01	N/A

This is the first register written in a four register sequence. This byte must be set for each PIC. Each PIC has a different base address. The other three ICW registers are written consecutively at offset 0x1.

Table 130. Initialization Command Word 1 Register (ICW1)

Bit	Function
7 - 5	**Reserved:** Must be set to 0.
4	ICW/OCW Select: Must be set to 1 to select ICW1, ICW2, ICW3 and ICW4. 0 for OCW1, OCW2, OCW3.
3	**Edge/Level Bank Select (LTIM):** Must be set to 0
2-1	**Reserved:** Must be set to 0.
0	**ICW4 Write Required:** Must be set to 1.

4.4.1.2 Initialization Command Word 2 Register (ICW2)

Table 131. Initialization Command Word 2 Register (ICW2) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF Passes Control to OS	SW Initiated Reset Value
1 byte	Sequenced	reg[1],reg[2]	0x01	Write Only	0x00 for PIC1 0x08 for PIC2	N/A

This byte defines the address returned as the Interrupt Vector. Bits 7 through 3 define the five high-order bits of the interrupt vector address as shown in Table 132. See also Table 126 on page 65 for a complete definition of the Interrupt Vector.

Table 132. Initialization Command Word 2 Register (ICW2)

Bit	Function
7-3	**Interrupt Vector Base Address:** This field defines the base address of the interrupt vector table which gives the base addresses for the interrupt routines associated with each IRQ on the controller. Must be set to 0b00000 for PIC1 (master) and to 0b00001 for PIC2 (slave)

Table 132. Initialization Command Word 2 Register (ICW2) ***(Continued)***

Bit	Function
2-0	**Reserved:** Must be set to 0b000.

4.4.1.3 Initialization Command Word 3 Register (ICW3)

Table 133. Initialization Command Word 3 Register (ICW3) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF Passes Control to OS	SW Initiated Reset Value
1 byte	Sequenced	reg[1], reg[2]	0x01	Write Only	0x04 for PIC1 0x02 for PCI2	N/A

In PIC1 (master), this register has a value of 0x04 to tell PIC1 (master) that hardware interrupt IRQ2 indicates an interrupt is pending in the PIC2 (slave) controller. If IRQ2 wins the interrupt selection in PIC1 (master) then PIC1 (master) will assert the INT line to the processor. When the processor issues an acknowledge, PIC1 (master) will tell PIC2 (slave) to respond to the acknowledge cycle.

In PIC2 (slave), this register has a value of 0x02 to tell PIC2 (slave) that it is using hardware interrupt 2 to communicate with PIC1 (master). PIC2 (slave) compares this identification code which is broadcast from the PIC1 (master) controller; if they are equal, PIC2 (slave) releases the interrupt vector address on the data bus.

Table 134. Initialization Command Word 3 Register (ICW3)

Bit	Function
7-0	**Reserved:** Must be set to 0x04 for PIC1 and 0x02 for PIC2

4.4.1.4 Initialization Command Word 4 Register (ICW4)

Table 135. Initialization Command Word 4 Register (ICW4) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF Passes Control to OS	SW Initiated Reset Value
1 byte	Sequenced	reg[1],reg[2]	0x01	Write Only	0x01	N/A

This byte is written to either PIC1 (master) or PIC2 (slave).

Table 136. Initialization Command Word 4 Register (ICW4)

Bit	Function
7-5	**Reserved:** Must be set to 0.
4	**Special Fully-Nested Mode (SFNM):** If 1, then the PIC is in Special Fully-nested mode.
3-2	**Reserved:** Must be set to 0b00.
1	**Automatic EOI:** Must be set to 0.
0	**80286/80386 Microprocessor Mode:** Must be set to 1.

4.4.2 Operation Command Word Registers

Requirements:

4–15. For the device described herein, all the registers defined in sections 4.4.2.1 through 4.4.4 must be implemented precisely as described herein.

4–16. Writing of OCW1 must be performed before the OCW2 and OCW3 registers are accessed, and following the ICW1-4 initialization sequence.

4.4.2.1 Operation Command Word1 Register (OCW1)

Table 137. Operation Command Word 1 Register (OCW1) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF Passes Control to OS	SW Initiated Reset Value
1 byte	Sequenced	reg[1], reg[2]	0x01	Read/Write	0xFF	N/A

This byte controls the individual bits in the Interrupt Mask register. Reading this register will return the value in the Interrupt Mask register.

Table 138. Operation Command Word 1 Register (OCW1)

Bit	Function
7-0	**Interrupt Mask:** When a 1 is written to any bit in this register, the corresponding IRQ line is masked. See Table 139 on page 71 for the bit-to-interrupt level correspondence.

.

Table 139. OCW1 Bit Definitions

Bit	PIC1 (master)	PIC2 (slave)
7	Interrupt Level 7	Interrupt Level 15
6	Interrupt Level 6	Interrupt Level 14
5	Interrupt Level 5	Interrupt Level 13
4	Interrupt Level 4	Interrupt Level 12
3	Interrupt Level 3	Interrupt Level 11
2	Interrupt Level 2	Interrupt Level 10
1	Interrupt Level 1	Interrupt Level 9
0	Interrupt Level 0	Interrupt Level 8

Software Implementation Note: Masking of interrupt level 2 on PCI1 (master) will effectively mask off all the interrupts which come into PIC2 (slave).

4.4.2.2 Operation Command Word 2 Register (OCW2)

Table 140. Operation Command Word 2 Register (OCW2) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF Passes Control to OS	SW Initiated Reset Value
1 byte	Direct	reg[1],reg[2]	0x00	Write Only	0x00	N/A

This register controls the interrupt priority and EOI command.

Table 141. Operation Command Word 2 Register (OCW2)

Bit	Function
7-5	**Rotate and EOI Codes (R,SL, EOI):** These bits define the Rotate mode, EOI mode, or a combination of the two, as shown in Table 142
4-3	**OCW2 Select:** Must be 0b00 to select OCW2
2- 0	**Interrupt Level Select:** When bit 6, SL, is set to 1, the encoding of this field as shown in Table 143 determines which interrupt channel the command acts on

Table 142. Rotate and EOI Code

Bit 7	Bit 6	Bit 5	Function
0	0	0	Reserved
0	0	1	Non-specific EOI Command
0	1	0	No Operation
0	1	1	Specific EOI Command. Interrupt channel number determined by setting of bits 2-0 (see Table 143)
1	0	0	Reserved
1	0	1	Rotate on Non-specific EOI Command
1	1	0	Set Priority Command. Interrupt channel number determined by setting of bits 2-0 (see Table 143)
1	1	1	Rotate on Specific EOI Command. Interrupt channel number determined by setting of bits 2-0 (see Table 143)

Table 143. Interrupt Level Select Encodings

Bit 2	Bit 1	Bit 0	PIC1 (master) Function	PIC2 (slave) Function
0	0	0	Interrupt Level 0	Interrupt Level 8
0	0	1	Interrupt Level 1	Interrupt Level 9
0	1	0	Interrupt Level 2	Interrupt Level 10
0	1	1	Interrupt Level 3	Interrupt Level 11
1	0	0	Interrupt Level 4	Interrupt Level 12
1	0	1	Interrupt Level 5	Interrupt Level 13
1	1	0	Interrupt Level 6	Interrupt Level 14
1	1	1	Interrupt Level 7	Interrupt Level 15

4.4.2.3 Operation Command Word 3 Register (OCW3)

Table 144. Operation Command Word 3Register (OCW3) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF Passes Control to OS	SW Initiated Reset Value
1 byte	Direct	reg[1], reg[2]	0x00	Read/Write	0x00	N/A

This byte is written to either PIC1 (master) or PIC2 (slave).

Table 145. Operation Command Word 3 Register (OCW3)

Bit	Function
7	**Reserved:** Must be set to 0
6	**Special Mask Mode (SMM):** See Table 146 on page 73
5	**Enable Special Mask Mode (ESMM):** See Table 146 on page 73
4-3	**OCW3 Select:** Must be set to 0b01
2	**Poll Command:** When set to a 0, the Poll Command mode is disabled, and when set to a 1, the next read to one of the Interrupt Controller registers will be treated as an Interrupt Acknowledge cycle, with the data being returned as shown in Table 148 on page 73
1,0	**Register Read Command:** After programming these bits with the appropriate values as shown in Table 147 on page 73, the selected register (IRR or ISR) can be read repeatedly without having to write OCW3 each time

Table 146. Interpretation of SMM and ESMM Bits

Bit 6 (SMM)	Bit 5 (ESMM)	Function
0	0	No Action
0	1	Normal Mask Mode
1	0	No Action
1	1	Special Mask Mode

Table 147. Register Read Command Decoding

Bits 1-0	Function
0x00	No Action
0x01	No Action
0x02	**Read Interrupt Request Register (IRR):** See Table 149 on page 74
0x03	**Read In-Service Register (ISR):** See Table 151 on page 75

Table 148. Interrupt Controller Response to a Read Pulse In Poll Mode

Bit	Function
7	**Interrupt Present:** This bit is set to 1 if an interrupt is present.

Table 148. Interrupt Controller Response to a Read Pulse In Poll Mode *(Continued)*

Bit	Function
6-3	**Undefined:** These bits are not used and may be set to either 0 or 1.
2-0	**Highest Priority Level:** These bits contain the binary code of the highest priority level requesting service

4.4.3 Interrupt Request Register (IRR)

The IRR indicates interrupts which are pending and which have not yet been acknowledged.

Table 149. Interrupt Request Register (IRR) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF Passes Control to OS	SW Initiated Reset Value
1 byte	Indirect (OCW3)	reg[1], reg[2]	0x00	Read Only	Undefined	N/A

A 1 in any bit position indicates that there is an interrupt pending, a 0 indicates that there is no interrupt pending.

Table 150. Interrupt Request Register (IRR)

Bit	PIC1 (master)	PIC2 (slave)
7	Interrupt Level 7	Interrupt Level 15
6	Interrupt Level 6	Interrupt Level 14
5	Interrupt Level 5	Interrupt Level 13
4	Interrupt Level 4	Interrupt Level 12
3	Interrupt Level 3	Interrupt Level 11
2	Interrupt Level 2	Interrupt Level 10
1	Interrupt Level 1	Interrupt Level 9
0	Interrupt Level 0	Interrupt Level 8

4.4.4 In-Service Register (ISR)

Requirements:

4–17. When an in-service bit is set to 1, the Interrupt Controller must inhibit all other interrupts with the same or lower priority.

4–18. When an in-service bit is set to 1, the Interrupt Controller must allow interrupts with a higher priority to cause an interrupt.

The ISR indicates interrupts which have been acknowledged but have not had an EOI issued for them.

Table 151. In-Service Register (ISR) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF Passes Control to OS	SW Initiated Reset Value
1 byte	Indirect (OCW3)	reg[1], reg[2]	0x00	Read Only	Undefined	N/A

If the bit is a 1, then there is an interrupt that has been acknowledged but not yet EOI'ed.

Table 152. In-Service Register (ISR)

Bit	PIC1 (master)	PIC2 (slave)
7	Interrupt Level 7	Interrupt Level 15
6	Interrupt Level 6	Interrupt Level 14
5	Interrupt Level 5	Interrupt Level 13
4	Interrupt Level 4	Interrupt Level 12
3	Interrupt Level 3	Interrupt Level 11
2	Interrupt Level 2	Interrupt Level 10
1	Interrupt Level 1	Interrupt Level 9
0	Interrupt Level 0	Interrupt Level 8

4.4.5 Edge/Level Interrupt Control Registers (ELI)

These registers determine whether an interrupt is edge sensitive or level sensitive. After reset all registers are set to edge sensitive (positive edge; that is, low to high transition).

Requirements:

4–19. For the device described herein, all the registers defined in sections 4.4.5.1 through 4.4.5.2 must be implemented precisely as described herein.

4.4.5.1 Edge/Level Interrupt Control Register for PIC1 (master) (ELI1)

Table 153. Edge/Level Interrupt Control Register 1 (ELI1) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF Passes Control to OS	SW Initiated Reset Value
1 byte	Direct	reg[3]	0x00	Read/Write	Undefined	N/A

Table 154. Edge/Level Interrupt Control 1 Register (ELI1)

Bit	Function
7	**INT7:** When this bit is 0, INT7 is edge sensitive, when this bit is a 1, INT7 is level-sensitive
6	**INT6:** When this bit is 0, INT6 is edge sensitive, when this bit is a 1, INT6 is level-sensitive
5	**INT5:** When this bit is 0, INT5 is edge sensitive, when this bit is a 1, INT5 is level-sensitive
4	**INT4:** When this bit is 0, INT4 is edge sensitive. when this bit is a 1, INT4 is level-sensitive
3	**INT3:** When this bit is 0, INT3 is edge sensitive, when this bit is a 1, INT3 is level-sensitive
2-0	**Reserved:** Always 0 (INT2, INT1, and INT0 are edge sensitive)

4.4.5.2 Edge/Level Interrupt Control Register for PIC2 (slave) (ELI2)

Table 155. Edge/Level Interrupt Control 2 Register (ELI2) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF Passes Control to OS	SW Initiated Reset Value
1 byte	Direct	reg[3]	0x01	Read/Write	Undefined	N/A

Table 156. Edge/Level Interrupt Control 2 Register (ELI2)

Bit	Function
7	**INT15:** When this bit is 0, INT15 is edge sensitive,.when this bit is a 1, INT15 is level-sensitive
6	**INT14:** When this bit is 0, INT14 is edge sensitive when this bit is a 1, INT14 is level-sensitive
5	**Reserved:** Always 0 (INT13 is edge sensitive)
4	**INT12:** When this bit is 0, INT12 is edge sensitive, when this bit is a 1, INT12 is level-sensitive
3	**INT11:** When this bit is 0, INT11 is edge sensitive, when this bit is a 1, INT11 is level-sensitive

Table 156. Edge/Level Interrupt Control 2 Register (ELI2) ***(Continued)***

Bit	Function
2	**INT10:** When this bit is 0, INT10 is edge sensitive, when this bit is a 1, INT10 is level-sensitive
1	**INT9:** When this bit is 0, INT9 is edge sensitive, when this bit is a 1, INT9 is level-sensitive
0	**Reserved:** Always 0 (INT8 is edge sensitive)

4.5 References

1. *Intel 82378ZB System I/O (SIO), Intel Corporation* [20].

2. *SL82C565 System I/O Controller with PCI Arbiter*, Symphony Laboratories Publication Number 2565; Version A.2. [21]

Parallel Port Controller

5

Printers have been attached to PCs, workstations, and servers for years with a parallel port which was developed several decades ago. Over time, this port has been enhanced to increase the data rates and add additional capabilities such as bi-directional capabilities, addressing, data transfer by DMA, and so on. This chapter describes the requirements for the Parallel Port Controller for platforms based on the Common Hardware Reference Platform architecture.

The following is a general description of the various modes in which the Parallel Port Controller can operate:

- *Standard Parallel Port (SPP) mode* is a combination of what the IEEE 1284 Specification [25] calls Compatibility mode and Nibble mode. This is one of the oldest modes and therefore the most common. Data transfer in this mode is mainly in the forward (host to device) direction, however four control lines can be used for transfer of data from the device to the host in the Nibble mode. All IEEE 1284 compatible devices are required to support the Nibble mode. In this mode there is no FIFO or DMA and all of the data transfer is done a byte at a time (or a nibble at a time in the Nibble mode) by the software, with software also directly controlling the control signals to the printer.
- *SPP Extended mode* adds to the SPP mode the capability of transferring data in the reverse direction (device to host) on the eight data lines. The IEEE 1284 Specification calls this the Byte mode.
- *Enhanced Parallel Port (EPP) mode* provides an interface from the host to the device which is similar to a typical microprocessor bus, with an address being provided followed by one or more data cycles. Although there are two modes detailed in some specifications (versions 1.7 and 1.9), only the more recent version, 1.9, is required by this architecture. In this mode, the address and data strobe signals to the device are controlled by the Parallel Port Controller hardware. This mode is tied to the microprocessor bus timing, with the bus being held off for the duration of the read or write to the bus (there is no First-In-First-Out (FIFO) buffer to speed-match the incoming data to the outgoing data).
- *Extended Capabilities Port (ECP) mode* has channel-like characteristics where there are 128 channels that can be addressed between the host and the device. All signalling is controlled by the Parallel Port Controller hardware and there is a FIFO to match the speed of the system to the speed of the peripheral device, so this is one of the faster modes of operation. In addition, filling and emptying of the FIFO by DMA is supported in this mode.
- *Parallel Port FIFO mode* is like the SPP mode except there is hardware control of the interface signals, there is a speed matching FIFO, and DMA is supported for FIFO fill. Note that the direction for this mode is always the forward direction.
- *Test FIFO mode* is a mode for testing of ECP device driver code and for diagnostic purposes. Data written to the FIFO can be read back. Data goes out on the data lines to the attached peripheral device, but the device does not see the data because the control signals are not activated.

The capabilities of these modes are summarized in Table 157 on page 80.

Table 157. Summary of the Parallel Port Controller Modes

Mode	Reverse Data Path	Control of Interface Signals by Hardware or Software?	FIFO Support	DMA Support	Device Address Control
SPP	4 bits via interface control signals	Software	No	No	None
SPP Extended	8 bit via data lines	Software	No	No	None
EPP	8 bit via data lines	Hardware	No	No	8 bit address
ECP	8 bit via data lines	Hardware	Yes	Yes	7 bit channel address
Parallel Port FIFO	Forward direction only (reverse support via SPP mode)	Hardware	Yes	Yes	None
Test FIFO	Reverse direction gives data written to FIFO in forward direction	N/A	Yes	Yes	N/A

5.1 General Requirements

Table 158 on page 80 is an excerpt from *PowerPC Microprocessor Common Hardware Reference Platform: A System Architecture*, showing the minimum requirements of the Parallel Port Controller on CHRP platforms.

Table 158. Minimum Platform Requirements for the Parallel Port

Subsystem	Specification	Portable	Personal	Server	Description
Parallel Port	P1284 +ECP Mode	O	R	O	[20]

Requirements:

5–1. The Parallel Port Controller must be represented by the Open Firmware properties “name=parallel, device_type=parallel, compatible=chrp,ecp”. The compatible property may also specify the compatible property for the National Semiconductor 87308 in addition to “chrp,ecp”.

5–2. The Parallel Port Controller must protect against damage to the Parallel Port Controller electronics in the case where the device attached to the Parallel Port has its power on and the Parallel Port Controller has its power off.

5–3. The device described herein must support the following modes as defined by the IEEE 1284 Specification: Compatibility mode, Nibble mode (Compatibility mode along with Nibble mode are called Standard Parallel Port, SPP, in this specification), Byte mode (called SPP Extended mode in this specification), Enhanced Parallel Port (EPP) mode (version 1.9), and Extended Capabilities Port (ECP) mode.

5–4. The device described herein must also support the following modes as described herein: Parallel Port FIFO mode and Test FIFO mode.

5–5. This device must be addressed purely with ISA I/O addresses which can be aliased or non-aliased.

5–6. DMA transfers fro the Parallel Port Controller must be 8-bit DMA slave and must support at least the ISA Compatible Timing mode.

5–7. Interrupts generated by the Parallel Port Controller must be high true edge sensitive.

5–8. The Parallel Port Controller registers specified herein must be implemented precisely as defined, including the following characteristics and bit definitions:

- **a.** Table 160, “Data Register (DTR) Characteristics,” on page 84
- **b.** Table 161, “Data Register (DTR),” on page 84
- **c.** Table 162, “Status Register (STR or DSR) Characteristics,” on page 85
- **d.** Table 163, “Status Register (STR or DSR),” on page 85
- **e.** Table 164, “Control Register (CTR or DCR) Characteristics,” on page 85
- **f.** Table 165, “Control Register (CTR or DCR),” on page 86
- **g.** Table 166, “Extended Control Register (ECR) Characteristics,” on page 87
- **h.** Table 167, “Extended Control Register (ECR),” on page 87
- **i.** Table 168, “EPP Address Register (ADDR) Characteristics,” on page 88
- **j.** Table 169, “EPP Data Port 0 Register (DTR) Characteristics,” on page 88
- **k.** Table 170, “EPP Data Port 1 Register (DATA1) Characteristics,” on page 88
- **l.** Table 171, “EPP Data Port 2 Register (DATA2) Characteristics,” on page 89
- **m.** Table 172, “EPP Data Port 3 Register (DATA1) Characteristics,” on page 89
- **n.** Table 173, “ECP Address FIFO (AFIFO) Characteristics,” on page 90
- **o.** Table 174, “Address FIFO Register (AFIFO),” on page 90
- **p.** Table 175, “ECP Data FIFO Register (DFIFO) Characteristics,” on page 90
- **q.** Table 176, “ECP Data FIFO Register (DFIFO),” on page 90
- **r.** Table 177, “Parallel Port FIFO Register (CFIFO) Characteristics,” on page 91
- **s.** Table 178, “Parallel Port FIFO Register (CFIFO),” on page 91
- **t.** Table 179, “Test FIFO Register (TFIFO) Characteristics,” on page 91
- **u.** Table 180, “Test FIFO Register, TFIFO,” on page 91

Hardware Implementation Note: IEEE Std 1284-1994 specification recommends the use of the IEEE 1284-C connectors in all new designs (host side as well as peripheral side).

Software Implementation Note: Note that the operating system is expected to extract the DMA channel number from the node’s *dma* property. Requirement 5–6 means that the operating system need not extract the DMA transfer size, DMA count width, or master versus slave capability from the node’s *dma* property. Firmware must set these fields to the values that match the above requirement. Operating systems are recommended to extract the DMA timing from the node’s *dma* property, though they are free to default to the ISA Compatibility Timing mode.

Software Implementation Note: Note that the operating system is expected to extract the interrupt number from the node's *interrupts* property. Requirement 5–7 means that the operating system need not extract the interrupt type from the node's *interrupts* property. Firmware must set this field to the value that matches the above requirement.

5.2 Open Firmware Requirements

Requirements:

5–9. The Open Firmware "reg" property information given in Table 159 on page 83 must be provided in the Open Firmware device tree for this device.

5–10. This device must be represented as an ISA device and a child of an ISA bridge node in the Open Firmware device tree.

5–11. This device must be represented in the OF device tree as specified in the *CHRP ISA Parallel Port Device Binding to: IEEE 1275-1994 Standard for Boot (Initialization, Configuration) Firmware*. [23]

5–12. All the reg property fields specified in Table 159 on page 83 must comply with the format given in the ISA (or PCI) bus binding to IEEE Std 1275-1994, as modified by the CHRP binding to IEEE Std 1275-1994.

5–13. The following events must occur during the OF initialization sequence:

a. The Parallel Port Controller must be enabled and activated.

b. If there is I/O Range checking of the Parallel Port Controller it must be disabled.

c. The parallel Port Controller address space must be set up, enabled.

d. The interrupt level and type (edge versus level) must be setup, enabled, and reported in the OF device tree.

e. Eight-bit DMA for the Parallel Port Controller must be: set up, enabled, and reported in the OF device tree.

f. Demand DMA must be enabled

g. Both ECP and EPP 1.9 modes must be enabled.

h. The Parallel Port Clock must be enabled (that is, clock must be operational).

i. The Parallel Port Controller must be set up to disable its outputs when the Parallel Port is disabled.

j. All Parallel Port Controller registers must be initialized to the values indicated in the "Value in Register When OF Passes Control to OS" column of the characteristics table corresponding to the specific register.

The Parallel Port Controller reg properties are listed below in Table 159 on page 83. The positions in the reg property (denoted by reg[1], reg[2], etc.) and the value of the size fields are stated in Table 159 on page 83. The format is described in the referenced Open Firmware bindings.

Table 159. Parallel Port Controller Registers and Offsets

Name	Offset	Type	Mode	Reg Properties for Open Firmware
DTR	0x00	Read/Write	SPP, SPP Extended, EPP	reg[1], size=8
DATAR	0x00	Read Only	ECP	
AFIFO	0x00	Write Only	ECP	
STR	0x01	Read Only	SPP, SPP Extended, EPP Parallel Port FIFO, Test FIFO	
DSR			ECP	
CTR	0x02	Read/Write	SPP, SPP Extended, EPP Parallel Port FIFO, Test FIFO	
DCR			ECP	
ADDR	0x03	Read/Write	EPP	
DATA0	0x04	Read/Write	EPP	
DATA1	0x05	Read/Write	EPP	
DATA2	0x06	Read/Write	EPP	
DATA3	0x07	Read/Write	EPP	
CFIFO	0x00	Write Only	Parallel Port FIFO	reg[2], size=6
DFIFO	0x00	Read/Write	ECP	
TFIFO	0x00	Read/Write	Test FIFO	
Reserved	0x01			
ECR	0x02	Read/Write	All	
Reserved	0x03			
Reserved	0x04			
Reserved	0x05			

Hardware Implementation Note: The Parallel Port Controller cannot be located at the legacy LPT1 addresses because of the size of 8 bytes would cause it to overlap with other allocated I/O addresses.

5.3 Parallel Port Register Definitions

For the device described herein, all the registers are defined in sections 5.3.1.1 through 5.3.1.4. The purpose and operation of some of these registers changes depending on the mode in which the parallel controller is operating at the time.

5.3.1 Registers Used in Multiple Modes

The first of the registers in this section is defined for all modes except ECP. The rest of the registers in this section are defined for all modes.

5.3.1.1 DATA Register (DTR or DATAR)

This register is accessed in SPP, SPP Extended, and EPP 1.9 mode as a Read/Write register, and in ECP mode as a Read Only register. In all modes except ECP, this register is abbreviated DTR. In ECP mode this register is abbreviated DATAR. This register is for writing data to or reading data from the parallel interface.

Table 160. Data Register (DTR) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF Passes Control to OS	SW Initiated Reset Value
1 byte	Direct - Mode Selected	reg[1]	0x00	Read/Write	0x00	N/A

This register is defined as shown in Table 161.

Table 161. Data Register (DTR)

Bit	Description
7-0	**Data:** The action performed on a read or write depends on the mode in which the Parallel Port Controller is operating. In all modes, when the direction is in the forward direction (bit 5 of the Extended Control Register (ECR) is a 0), on a read the data returned is either the data in the Data register or the data on the input lines to the chip (signal lines PD7-0), depending on the implementation. In SPP mode on a write, the data in this register is driven directly to the output signal pins PD7-0. In SPP Extended mode: • On a write, data written to this register is latched and is gated to the output depending on the direction bit in the Control register (CTR). If the direction bit (bit 5) is a 0, then data transfer is from the Parallel Port to the device (forward direction) and therefore the data written is latched and gated to signal lines PD7-0. If the direction bit (bit 5) is a 1, then data transfer is from the device to the Parallel Port (reverse direction) and therefore the data written is latched into the Data register but not gated out onto the signal lines. • On a read with the direction bit a 1, then data transfer is from the device to the Parallel Port (reverse direction) and the data read is the data on the input lines to the chip (signal lines PD7-0). In EPP 1.9 mode: same as SPP mode except that in EPP mode, the busy bit of the STR must be checked to make sure it is a 1 (not busy) before writing to the DTR, in order to avoid possible corruption of an EPP cycle that is in progress. In ECP mode: For reads, same as SPP Extended mode. For writes, undefined.

Software Implementation Note: Damage to the drivers in the Parallel Port Controller interface or the attached device can occur if the controller and the attached device are driving data onto the data lines at the same time.

5.3.1.2 Status Register (STR or DSR)

This register returns the interrupt-pending status of the interface, and the real-time status of the connector pins. In all modes except ECP, this register is abbreviated STR. In ECP mode this register is abbreviated DSR.

Table 162. Status Register (STR or DSR) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF Passes Control to OS	SW Initiated Reset Value
1 byte	Direct	reg[1]	0x01	Read/Write	Undefined	N/A

The Status register is defined as shown in Table 163.

Table 163. Status Register (STR or DSR)

Bit	Description
7	**$\overline{\text{Busy}}$:** This bit represents the inverse of the state of the busy signal. When this bit is 0, the printer is busy and cannot accept data. In the Nibble mode, this is bit 3 and then bit 7 of the data. This bit is Read Only.
6	**$\overline{\text{Acknowledge}}$ ($\overline{\text{ACK}}$):** This bit represents the current state of the device acknowledge ($\overline{\text{ACK}}$) signal. When this bit is pulsed from a 1 to a 0 to a 1, the device has received a character and is ready to accept another. This bit is Read Only.
5	**Paper End (PE):** This bit represents the current state of the device paper end (PE) signal. When this bit is 1, the printer detected the end of the paper. In the Nibble mode, this is bit 2 and then bit 6 of the data. This bit is Read Only.
4	**Select (SLCT):** This bit represents the current state of the select (SLCT) signal. When this bit is 1, the printer has been selected. In the Nibble mode, this is bit 1 and then bit 5 of the data. This bit is Read Only.
3	**$\overline{\text{Error}}$:** This bit represents the current state of the device $\overline{\text{ERR}}$ signal. When this bit is 0, the printer has encountered an error condition. In the Nibble mode, this is bit 0 and then bit 4 of the data. This bit is Read Only.
2-1	**Reserved:** On a read, the returned data is undefined.
0	**Time-Out:** EPP Time-out bit. • When not in EPP mode: On a read, the returned data is undefined. • In EPP mode: A 0 indicates that no timeout has occurred and a 1 indicates that a time-out has occurred on an EPP cycle (minimum 10 microseconds). This bit is cleared to 0 after the STR is either read (some implementations) or written (other implementations). If cleared on a read of the bit, hardware must ignore writes to the bit. Software must write this bit to a 0 to guarantee that it gets reset.

5.3.1.3 Control Register (CTR or DCR)

In all modes except ECP, this register is abbreviated CTR. In ECP mode this register is abbreviated DCR. This register provides access to all the outbound control lines to the printer. All bits except bit 5 are Read/Write.

Table 164. Control Register (CTR or DCR) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF Passes Control to OS	SW Initiated Reset Value
1 byte	Direct	reg[1]	0x02	Read/Write	0xC0	N/A

The CTR (DCR) is defined as shown in Table 165.

Table 165. Control Register (CTR or DCR)

Bit	Description
7-6	**Reserved:** These bits are reserved and must be set to 0b11. On a read, the returned data is undefined.
5	**Direction:** When this bit is a 0, the Parallel Port Controller is in the forward direction (data transfer in the Parallel Port Controller to device direction). When this bit is a 1, the Parallel Port Controller is in the backward direction (data transfer in the device to Parallel Port Controller direction) and the Parallel Port Controller must not be driving the data lines. How this bit affects operations depends on which mode the Parallel Port Controller is operating. See also Table 161 on page 84. The Parallel Port Controller must be in the SPP Extended mode when this bit is changed from a 0 to a 1. • In SPP mode: This bit is Write Only. Must always be written to 0. Read as 0. • In SPP Extended mode: This bit is a Read/Write bit. • In EPP modes: Same as SPP except that this bit is a Read/Write bit and when there is an on-going EPP cycle, $\overline{RD}$ and $\overline{WR}$ signals on the bus determine the direction. This bit must not be changed from a 0 to a 1 in this state. • In ECP mode: This bit is a Read/Write bit. This bit must not be changed from a 0 to a 1 in this state. • In Parallel Port FIFO mode: This bit is Write Only. Must always be written to 0. • In Test FIFO mode: This bit is a Read/Write bit. This bit must not be changed from a 0 to a 1 in this state.
4	**Interrupt Enable:** This bit enables the $\overline{ACK}$ interrupt event (1=enable, 0=mask). An interrupt occurs when this bit is a 1 and either the $\overline{acknowledge}$ signal changes from active to inactive (non-DMA mode only) or when the EPP timeout bit (bit 0 of the Status register) gets set to 1 in the EPP mode. In all modes except the ECP mode, this bit floats (high-impedance state) the IRQ pin when this bit is 0.
3	**SLIN:** This bit controls the select in signal. When this bit is 1, the printer is selected. This bit must be set to a 1 before switching to the ECP mode. • In all modes except ECP mode: If the current state of the actual signal from the device is to be obtained by reading this register, this bit should be set to its inactive state (0). • In ECP mode: On a read, this bit will always reflect the value in the Parallel Port Controller's DCR register. • In EPP modes: When this bit is a 0, the $\overline{SLIN}$ signal is controlled by ECP hardware.
2	**$\overline{INIT}$:** This bit controls the initialize signal. When this bit is 0, the printer will be initialized. this bit must be set to a 1 before switching to the ECP mode. • In all modes except ECP mode: If the current state of the actual signal from the device is to be obtained by reading this register, this bit should be set to its inactive state (1). • In ECP mode: On a read, this bit will always reflect the value in the Parallel Port Controller's DCR register.
1	**AFD:** This bit controls the automatic feed XT signal. When this bit is 1, the $\overline{AFD}$ signal is driven low and the printer automatically spaces the paper up one line for every carriage return. • In all modes except ECP mode: If the current state of the actual signal from the device is to be obtained by reading this register, this bit should be set to its inactive state (0). • In ECP mode: On a read, this bit will always reflect the value in the Parallel Port Controller's DCR register. • In ECP and EPP modes: When this bit is a 0, the $\overline{AFD}$ signal is controlled by ECP hardware.
0	**STB:** This bit controls the $\overline{strobe}$ ($\overline{STB}$) signal to the printer. When this bit is 1 the $\overline{STB}$ line is activated, causing data is clocked into the attached device's data register. • In all modes except ECP mode: If the current state of the actual signal from the device is to be obtained by reading this register, this bit should be set to its inactive state (0). • In ECP mode: On a read, this bit will always reflect the value in the Parallel Port Controller's DCR register. • In Parallel Port FIFO mode and ECP mode: When this bit is 0, $\overline{STB}$ is controlled by Parallel Port Controller hardware.

5.3.1.4 Extended Control Register (ECR)

Table 166. Extended Control Register (ECR) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF Passes Control to OS	SW Initiated Reset Value
1 byte	Direct	reg[2]	0x02	Read/Write	0x15	N/A

The ECR register is defined as shown in Table 167.

Table 167. Extended Control Register (ECR)

Bit	Description
7-5	**Parallel Port Mode Bits:** These bits determine the mode in which the Parallel Port is operating. • 000 - SPP (Compatibility and Nibble modes together): In this mode, write cycles are controlled by the software. In this mode, the direction bit in the DCR is not used and PD7-0 are driven. The FIFO is reset and is empty. • 001 - SPP Extended (Byte mode): Read and Write are under software control, the FIFO is reset, and the direction bit is active. • 010 - Parallel Port FIFO: Write cycles are controlled by hardware with STB being generated by the hardware. The direction bit of the DCR is assumed to be 0 (this mode is defined for the forward direction only) and PD7-0 are driven. • 011 - ECP: The FIFO direction is controlled by the direction bit in the DCR. $\overline{STB}$ and $\overline{AFD}$ are controlled by hardware. • 100 - EPP (1.7 and 1.9). • 101 - Reserved. • 110 - Test FIFO: The FIFO is accessible via the TFIFO register. The hardware does not automatically fill and empty the FIFO. • 111 - Reserved.
4	**Interrupt Mask Bit:** When this bit is 0, an interrupt is generated on $\overline{ERR}$ assertion from the parallel interface (the high to low edge of $\overline{ERR}$). An interrupt is also generated if $\overline{ERR}$ is asserted when this bit is changed from 1 to 0. When this bit is a 1, no interrupt will be generated when $\overline{ERR}$ is activated.
3	**DMA Enable Bit:** When this bit is a 0, the DMA request signal (DMA0, 1, 2 or 3) that serves the selected DMA request signal is put into tri-state (high-impedance) state. When this bit is a 1, the DMA is enabled and the DMA starts when bit 2 of this register (ECP Service) is 0.
2	**Service Bit:** When the ECP clock is frozen, this bit is read as 0 regardless of its actual value (even though this bit may be modified by software when the ECP clock is frozen). When one of the following interrupt events occurs, an interrupt will be generated and this bit will be set to a 1 by the hardware: • Bit 3 of the ECR is a 1 and the terminal count is reached during DMA • Bit 3 of the ECR is 0 and bit 5 of the DCR (CTR) is 0 and there are eight or more bytes free in the FIFO • Bit 3 of the ECR is a 0 and bit 5 of the DCR (CTR) is a 1 and there are eight or more bytes to be read from the FIFO. When this bit is a 1, no further DMA requests or interrupt requests are generated. Writing a a 1 to this bit does not cause an interrupt.
1	**FIFO Full:** This bit is Read Only. A write to the ECR will not change its value. This bit continuously reflects the FIFO state. When the ECP clock is frozen this bit is read as 1, regardless of the actual FIFO state. When this bit is a 0, the FIFO has at least one free byte. When this bit is a 1, the FIFO is full.
0	**FIFO Empty:** This is a Read Only bit that continuously reflects the FIFO state. When this bit is a 0, the FIFO has at least one byte of data. When this bit is a 1, the FIFO is empty. If the ECP clock is frozen, this bit is a 1 regardless of the actual FIFO state.

5.3.2 Registers Used Only in EPP Mode

The registers in this section are only defined in EPP modes.

5.3.2.1 EPP Address Register (ADDR)

Table 168. EPP Address Register (ADDR) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF Passes Control to OS	SW Initiated Reset Value
1 byte	Direct - Mode Selected	reg[1]	0x03	Read/Write	Undefined	N/A

Requirements:

5–14. Writing a Parallel Port device address to this register must initiate an EPP device/register selection operation.

5.3.2.2 EPP Data Port 0 Register (DATA0)

Table 169. EPP Data Port 0 Register (DTR) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF Passes Control to OS	SW Initiated Reset Value
1 byte	Direct - Mode Selected	reg[1]	0x04	Read/Write	Undefined	N/A

Requirements:

5–15. Accesses to this register must initiate device read or write operations to bits 7-0 of the 32 bit word to be transferred.

5.3.2.3 EPP Data Port 1 Register (DATA1)

Table 170. EPP Data Port 1 Register (DATA1) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF Passes Control to OS	SW Initiated Reset Value
1 byte	Direct - Mode Selected	reg[1]	0x05	Read/Write	Undefined	N/A

Requirements:

5–16. Accesses to this register must initiate device read or write operations to bits 15-8 of the 32 bit word to be transferred.

5.3.2.4 EPP Data Port 2 Register (DATA2)

Table 171. EPP Data Port 2 Register (DATA2) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF Passes Control to OS	SW Initiated Reset Value
1 byte	Direct - Mode Selected	reg[1]	0x06	Read/Write	Undefined	N/A

Requirements:

5–17. Accesses to this register must initiate device read or write operations to bits 23-16 of the 32 bit word to be transferred.

5.3.2.5 EPP Data Port 3 Register (DATA3)

Table 172. EPP Data Port 3 Register (DATA1) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF Passes Control to OS	SW Initiated Reset Value
1 byte	Direct - Mode Selected	reg[1]	0x07	Read/Write	Undefined	N/A

Requirements:

5–18. Accesses to this register must initiate device read or write operations to bits 31-24 of the 32 bit word to be transferred.

5.3.3 Registers Used Only in ECP Mode

The registers in this section are only defined in the ECP mode.

Software Implementation Note: When the OF compatible property includes the compatible name of the National Semiconductor 87308 or when the OF property *dma-308* is present, then the capability to do demand DMA gets disabled on exit from the ECP mode. In this case, to re-enable this capability, software should do the following:

1. Make sure that the Parallel Port Controller is quiesced
2. Write 0x05 to reg[1]+0x03
3. Read a byte from reg[1]+0x04
4. Set bit 6 of the byte just read to a 1
5. Write this byte to reg[1]+0x04

5.3.3.1 ECP Address FIFO Register (AFIFO)

In the ECP mode, when the Parallel Port's state is in the forward direction (bit 5 of the DCR is 0), a byte written into this register is pushed into the FIFO and tagged as a command. Writes to this register when the state of the Parallel Port is in the backward direction have no effect and the data is ignored.

Table 173. ECP Address FIFO (AFIFO) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF Passes Control to OS	SW Initiated Reset Value
1 byte	Direct - Mode Selected	reg[1]	0x00	Write Only	Undefined	N/A

The AFIFO is defined as shown in Table 174.

Table 174. Address FIFO Register (AFIFO)

Bit	Description
7-0	**Command:** These bits represent data that will be tagged as a command in the AFIFO register

5.3.3.2 ECP Data FIFO Register (DFIFO)

DFIFO is accessible in ECP mode. In the forward direction (bit 5 of the DCR is 0) a byte written to this register is pushed into the FIFO and tagged as data. In ECP mode, the DMA automatically writes to this register. Other write operations must address this register specifically. Reading this register has no effect and the data read is undefined.

In the backward direction (bit 5 of the DCR is 1) the ECP automatically issues ECP read cycles to fill the FIFO. Reading this register pops a byte from the FIFO. Writing this register has no effect and the data written is ignored.

Table 175. ECP Data FIFO Register (DFIFO) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF Passes Control to OS	SW Initiated Reset Value
1 byte	Direct - Mode Selected	reg[2]	0x00	Read/Write	Undefined	N/A

The DFIFO register is defined as shown in Table 176.

Table 176. ECP Data FIFO Register (DFIFO)

Bit	Description
7-0	**Data:** These bits represent the data that will be tagged as data when pushed onto or popped from the FIFO.

5.3.4 Registers Used Only in the Parallel Port FIFO Mode

CFIFO is accessible in Parallel Port FIFO mode. A byte written to CFIFO is pushed into the FIFO and tagged as data. In Parallel Port FIFO mode the DMA automatically writes to this register. Other write operations must address this register specifically. This mode is defined for the forward direction only, therefore this register is Write Only.

Table 177. Parallel Port FIFO Register (CFIFO) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF Passes Control to OS	SW Initiated Reset Value
1 byte	Direct - Mode Selected	reg[2]	0x00	Write Only	Undefined	N/A

The CFIFO register is defined as shown in Table 178.

Table 178. Parallel Port FIFO Register (CFIFO)

Bit	Description
7-0	**Data:** These bits represent the data that will be tagged as data when pushed onto the FIFO.

5.3.5 Registers Used Only in the Test FIFO Mode

TFIFO is accessible in Test FIFO mode. A byte written into this register is pushed into the FIFO. A byte read from this register is popped from the FIFO. The ECP does not issue a ECP cycle to transfer the data to or from the peripheral device. The TFIFO is readable and writable in both directions. In the forward direction (bit 5 of the DCR (CTR) is 0) PD0-7 are driven, but the data is undefined. The FIFO does not stall when overwritten or under-run (access is ignored). Bytes are always read from the top of the FIFO, regardless of the direction bit (bit 5 of the CTR or DCR).

Table 179. Test FIFO Register (TFIFO) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF Passes Control to OS	SW Initiated Reset Value
1 byte	Direct - Mode Selected	reg[2]	0x00	Read/Write	Undefined	N/A

The TFIFO register is defined as shown in Table 180.

Table 180. Test FIFO Register, TFIFO

Bit	Description
7-0	**Data:** These bits represent the data that will be tagged as data when pushed onto or popped from the FIFO.

5.4 References

1. *National Semiconductor PC87308VUL SuperI/O™ Enhanced Sidewinder Lite Plug and Play Compatible Chip, with a Floppy Disk Controller, a Keyboard Controller, a Real-Time Clock, Two UARTs, Full Infrared Support and an IEEE1284 Parallel Port* [24]
2. *CHRP ISA Parallel Port Device Binding to: IEEE 1275-1994 Standard for Boot (Initialization, Configuration) Firmware.* [23]
3. *IEEE Standard Signaling Method for a Bidirectional Parallel Peripheral Interface for Personal Computers*, also called *IEEE Std 1284-1994 Specification* [25]

UART Controller

6

6.1 CHRP Requirements

Table 181 is an excerpt from *PowerPC Microprocessor Common Hardware Reference Platform: A System Architecture*, showing the minimum requirements for a serial port on CHRP platforms.

Table 181. Summary of Minimum Platform Requirements (excerpt)

Subsystem	Specification	Portable	Personal	Server	Description
Serial Port		O	R	O	It is acceptable to provide the PC style interface or the Apple style interface by means of a plug-in card to achieve certification.
	16550	O	R	O	
	SCC	O	R	O	

Requirements:

6–1. The UART serial device must be NS16550 compatible, as specified in this chapter.

6–2. The UART serial device must be addressed only with ISA I/O addresses.

6–3. The UART serial device must generate only low-to-high edge sensitive interrupts.

Software Implementation Note: Requirements 6–2 and 6–3 relieve the operating system from having to extract address type or interrupt type from the Open Firmware device tree.

6.2 UART Controller Open Firmware Requirements

Requirements:

6–4. The Open Firmware device tree node for the UART serial device described in this chapter must follow the specifications given in the *CHRP ISA Serial Port Device Binding* [28].

6–5. The Open Firmware "**`reg`**" property information given in Table 182, "UART Registers," on page 94 must be provided in the Open Firmware device tree for the UART serial device.

6–6. The UART serial device must be represented as an ISA device and a child of an ISA bridge node in the Open Firmware device tree.

6.3 UART Registers

Requirements:

6–7. All the "**reg**" property fields specified in Table 182, "UART Registers," on page 94 must comply with the format given in the ISA/EISA/ISA-PnP binding to IEEE Std 1275-1994, as modified by the CHRP binding to IEEE Std 1275-1994.

6–8. The UART serial device must precisely implement the registers defined in Section 6.3, "UART Registers," on page 94.

6–9. The selection of the TDR/IER versus the DLL/DLH for the UART serial device must be exclusively controlled by the DLAB bit (bit 7) in the Line Control Register.

The positions in the "**reg**" property (denoted by reg[1], reg[2], etc.) and the value of the size fields are stated in Table 182 on page 94. The format is described in the referenced Open Firmware bindings.

Table 182. UART Registers

Address Offsets	Registers		Reg Properties
	Read	Write	
0x0	Receive Data Register (RDR)	Transmitter Data Register (TDR) OR[1] Divisor Latch Low Byte (DLL)	reg[1],size=8
1	Interrupt Enable Register (IER)	Interrupt Enable Register (IER) OR[1] Divisor Latch High Byte (DLH)	
2	Interrupt Identification Register (IIR)	FIFO Control Register (FCR)	
3	Line Control Register (LCR)	Line Control Register (LCR)	
4	Modem Control Register (MCR)	Modem Control Register (MCR)	
5	Line Status Register (LSR)	Writing the LSR is not recommended	
6	Modem Status Reg (MSR)	Modem Status Reg (MSR)	
7	Scratch Register (SCR)	Scratch Register (SCR)	

[1]These address offsets have dual register definitions based on the state of the DLAB bit in the Line Control Register (LCR). DLAB=0 references the TDR and IER; DLAB=1 references DLL and DLH

6.3.1 Transmitter Data Register (TDR)

Table 183. Transmitter Data Register (TDR) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF passes control to OS	SW Initialized Reset Value
1 byte	Direct, LCR(7)=0	irba[0]	0x00	Write Only	Undefined	N/A

Table 184. Transmitter Data Register Bit Definition

Bit	Description
7-0	**Data Bits 7-0:** The Transmitter Data Register contains the character to be sent. Bit 0 is the least-significant bit and is the first bit sent serially.

6.3.2 Receiver Data Register (RDR)

Table 185. Receiver Data Register (RDR) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF passes control to OS	SW Initialized Reset Value
1 byte	Direct, LCR(7)=0	irba[0]	0x00	Read Only	Undefined	N/A

Table 186. Receiver Data Register Bit Definition

Bit	Description
7-0	**Data Bits 7-0:** The Receiver Data Register contains the received character. Bit 0 is the least-significant bit and is the first bit received serially**.**

6.3.3 Interrupt Enable Register (IER)

The Interrupt Enable Register allows the four types of controller interrupts to separately activate the chip interrupt output signal. The interrupt system can be completely disabled by setting bits 0 through 3 of the Interrupt Enable register to 0. Similarly, by setting the appropriate bits of this register to 1, selected interrupts can be enabled. Disabling prevents the controller from generating the external interrupt to the system. All other system functions operate normally, including the setting of the Line Status and Modem Status registers.

Table 187. Interrupt Enable Register (IER) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF passes control to OS	SW Initialized Reset Value
1 byte	Direct, LCR(7)=0	irba[0]	0x01	Read/Write	0x00	N/A

Table 188. Interrupt Enable Register Bit Definition

Bit	Description
7-4	**Reserved** = 0
3	**Enable Modem Status Interrupt:** When set to 1, this bit enables the Modem Status Interrupt.
2	**Enable Receiver Line Status Interrupt:** When set to 1, this bit enables the Receiver Line Status Interrupt.
1	**Enable Transmitter Holding Register Empty Interrupt:** When set to 1, this bit enables the Transmitter Holding Register Empty Interrupt.
0	**Enable Received Data Available Interrupt (Character and FIFO Mode) and Time-Out Interrupts (FIFO Mode Only):** When set to 1, this bit enables the Received Data Available Interrupt. In the FIFO mode, this bit also enables the time-out interrupts.

6.3.4 FIFO Control Register (FCR)

The FIFO Control Register is a write-only register at the same location as the read-only Interrupt Identification Register. The FIFO Control Register enables the FIFO registers, clears the FIFO registers, and sets the Receiver FIFO register trigger level.

Software Implementation Note: The Transmitter and Receiver FIFO registers are not accessible serial controller registers.

Table 189. FIFO Control Register (FCR) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF passes control to OS	SW Initialized Reset Value
1 byte	Direct	irba[0]	0x02	Write Only	Undefined	N/A

Table 190. FIFO Control Register Bit Definition

Bit	Description
7, 6	**Receiver FIFO Register Trigger:** These bits select the trigger level for the receiver-register interrupt, as shown in Table 191. When the number of bytes in the RCVR FIFO reaches the selected interrupt trigger level, a Receiver Data Available Interrupt is activated.
5,3	**Reserved** = 0

Table 190. FIFO Control Register Bit Definition *(Continued)*

Bit	Description
2	**Transmitter FIFO Register Reset:** When this bit is set to 1, all bytes in the Transmitter FIFO register are cleared and its counter logic is reset to 0. The Transmitter Shift register is not cleared. This bit is self-clearing.
1	**Receiver FIFO Register Reset:** When this bit is set to 1, all bytes in the Receiver FIFO register are cleared and its counter logic is reset to 0. The Transmitter Shift register is not cleared. This bit is self-clearing.
0	**FIFO Mode Enable:** When this bit is set to 1, the FIFO mode is enabled. When this bit is changed, the transmit and receive (XMIT and RCV) FIFOs are cleared. When writing to any other FIFO Control register bits, this bit must be a 1.

Table 191. Receiver Trigger Level Bit Decodes

Bits 7,6	Receiver Trigger Level
0 0	1 Byte
0 1	4 Bytes
1 0	8 Bytes
1 1	14 Bytes

6.3.5 Interrupt Identification Register (IIR)

To minimize programming overhead during character mode transfers, the controller prioritizes interrupts into four levels:

- Priority 1 - Receiver-line-status
- Priority 2 - Received-data-available
- Priority 2 - Time-out (FIFO mode)
- Priority 3 - Transmitter-holding-register-empty
- Priority 4 - Modem status.

Information about a pending interrupt is stored in the Interrupt Identification Register (IIR). When the IIR is read, only the highest priority pending interrupt is indicated. After this interrupt condition is reset, other lower priority interrupts can be identified by re-reading the IIR.

Table 192. Interrupt Identity Register (IIR) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF passes control to OS	SW Initialized Reset Value
1 byte	Direct	irba[0]	0x02	Read Only	0x01	N/A

Table 193. Interrupt Identification Register Bit Definition

Bit	Description
7, 6	**FIFO Registers Enabled:** These two bits are set to 1 when FCR0=1 (FIFO mode enabled)
5, 4	**Reserved** = 0
3-0	**Interrupt ID (2-0), Interrupt Not Pending:** The interpretation of these bits is given in Table 194

Table 194. Interrupt Control Functions

Bits 3210	Priority	Type	Cause	Interrupt Reset Control
0001	n/a	none	no interrupt pending	n/a
0110	Highest	Receiver Line Status	Overrun, Parity or Framing error or Break Interrupt	Read the Line Status Register
0100	Second	Received data available	Data in the Receiver Buffer or the FIFO trigger level has been reached or a receive FIFO timeout occurs.	Read the Receiver Data Register or FIFO Register drops below the FIFO trigger level.
1100[1]	Second	Character Time-Out Indication	No characters have been removed from or put into the receiver FIFO register during the last four character times, and at least 1 character is in it at this time.	Read the Receiver Data Register.
0010	Third	Transmitter holding register empty	Transmitter holding register is empty	Read the Interrupt Identification Register or write to Transmitter Data Register
0000	Fourth	Modem status	Change in signal status from modem	Read the Modem Status Register.

[1] FIFO Mode only.

6.3.6 Line Control Register (LCR)

Asynchronous communications are programmed through the Line Control Register.

Table 195. Line Control Register (LCR) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF passes control to OS	SW Initialized Reset Value
1 byte	Direct	irba[0]	0x03	Read/Write	Undefined	N/A

Table 196. Line Control Register Bit Definition

Bit	Description
7	**Divisor Latch Access Bit (DLAB):** This bit is set to 1 to gain access to the divisor latches of the baud-rate generator. It is set to 0 to gain access to the Receiver Buffer, Transmitter Holding, or Interrupt Enable registers.
6	**Break Control:** When this bit is a 1, a break condition is transmitted to the receiving UART. A break condition is when the serial output is forced to the spacing state and remains there regardless of other transmitter activity. When this bit is set a 0, the break condition is not active.
5,4	**Parity Selection:** Bits 5 and 4 specify the parity selection as defined in Table 197. These bits have no effect if bit 3 is set to 0
3	**Parity Enable:** When bit 3 is set to 1, a parity bit is calculated based on the parity selection value in bits 5 and 4. This parity bit is generated on transmit and checked on receive. The parity bit appears between the last data bit and the first stop bit of the serial data. When bit 3 is set to 0, no parity bit is transmitted or checked.
2	**Number of Stop Bits:** The value of this bit, along with the value of bits 1 and 0, specify the number of stop bits (stop bit duration specified in terms of bit times). Table 198 gives the stop bit duration based on word length selection.
1,0	**Word Length Select, Bit 1,0:** These bits specify the number of data bits in each serial character that is sent or received as represented in Table 198

Table 197. Parity Selection Bit Description

Bits 5,4	Description
00	**Odd Parity Selected**
01	**Even Parity Selected**
10	**Mark Parity Selected:** A parity bit of logical 1 is always transmitted and checked.
11	**Space Parity Selected:** A parity bit of logical 0 is always transmitted and checked.

Table 198. Stop Bit Duration and Word Length

Bits 210	Stop Bit Duration (in bit times)	Word Length
000	1	5 Bits
001	1	6 Bits
010	1	7 Bits
011	1	8 Bits
100	1.5	5 Bits
101	2	6 Bits
110	2	7 Bits
111	2	8 Bits

6.3.7 Modem Control Register (MCR)

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

Table 199. Modem Control Register (MCR) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF passes control to OS	SW Initialized Reset Value
1 byte	Direct	irba[0]	0x04	Read/Write	0x00	N/A

Table 200. Modem Control Register Bit Description

Bit	Description
7-5	**Reserved** = 0
4	**Loop Mode:** This bit provides a loopback feature for diagnostic testing of the serial port. When bit 4 is set to 1: • Transmitter-serial-output is set to the marking state. • Receiver-serial-input is disconnected. • Output of the Transmitter Shift register is "looped back" to the Receiver Shift register input. • The modem control inputs (CTS, DSR, DCD, AND Rl) are disconnected. • The modem control outputs (DTR, RTS, OUT 1, and OUT 2) are internally connected to the (DSR, CTS, RI, and DCD) modem status inputs, respectively. • The modem control output pins are forced inactive.
3	**Out 2 (IRQ Output Control):** This bit is set = 1 to enable interrupts.
2	**Out 1:** This bit is not used in a normal mode. In loop mode, its status is reported to bit 6 (Rl) of the Modem Status register.
1	**Request-to-Send:** This bit controls the '-request to send' signal (-RTS) modem control output. When this bit is set to 1, -RTS is active. When this bit is set to 0, -RTS is inactive.
0	**Data-Terminal-Ready:** This bit controls the '-data terminal ready' signal (-DTR) modem control output. When this bit is set to1, -DTR is active. When this bit is set to 0, -DTR is inactive.

6.3.8 Line Status Register (LSR)

This 8-bit read-only register provides the system microprocessor with ongoing information about the data transfer.

Software Implementation Note: Writing to this register can produce unpredictable results.

Table 201. Line Status Register (LSR) Characteristics

Register Size	Register Addressing	Associated Reg Property	Offset Address	Read/Write Characteristics	Value in Register when OF passes control to OS	SW Initialized Reset Value
1 byte	Direct	irba[0]	0x05	Read Only	0xC0	N/A