

## HIGHPOINT ROCKETRAID 3xxx/4xxx ADAPTER DRIVER (hptiop)

## Controller Register Map

For RR44xx Intel IOP based adapters, the controller IOP is accessed via PCI BAR0 and BAR2:

BAR0 offset	Register
0x11C5C	Link Interface IRQ Set
0x11C60	Link Interface IRQ Clear
BAR2 offset	Register
0x10	Inbound Message Register 0
0x14	Inbound Message Register 1
0x18	Outbound Message Register 0
0x1C	Outbound Message Register 1
0x20	Inbound Doorbell Register
0x24	Inbound Interrupt Status Register
0x28	Inbound Interrupt Mask Register
0x30	Outbound Interrupt Status Register
0x34	Outbound Interrupt Mask Register
0x40	Inbound Queue Port
0x44	Outbound Queue Port

For Intel IOP based adapters, the controller IOP is accessed via PCI BAR0:

BAR0 offset	Register
0x10	Inbound Message Register 0
0x14	Inbound Message Register 1
0x18	Outbound Message Register 0
0x1C	Outbound Message Register 1
0x20	Inbound Doorbell Register
0x24	Inbound Interrupt Status Register
0x28	Inbound Interrupt Mask Register
0x30	Outbound Interrupt Status Register
0x34	Outbound Interrupt Mask Register
0x40	Inbound Queue Port
0x44	Outbound Queue Port

For Marvell IOP based adapters, the IOP is accessed via PCI BAR0 and BAR1:

BAR0 offset	Register
0x20400	Inbound Doorbell Register
0x20404	Inbound Interrupt Mask Register
0x20408	Outbound Doorbell Register
0x2040C	Outbound Interrupt Mask Register
BAR1 offset	Register
0x0	Inbound Queue Head Pointer
0x4	Inbound Queue Tail Pointer
0x8	Outbound Queue Head Pointer
0xC	Outbound Queue Tail Pointer
0x10	Inbound Message Register
0x14	Outbound Message Register
0x40-0x1040	Inbound Queue

0x1040-0x2040      Outbound Queue

## I/O Request Workflow

---

All queued requests are handled via inbound/outbound queue port.  
A request packet can be allocated in either IOP or host memory.

To send a request to the controller:

- Get a free request packet by reading the inbound queue port or allocate a free request in host DMA coherent memory.

The value returned from the inbound queue port is an offset relative to the IOP BAR0.

Requests allocated in host memory must be aligned on 32-bytes boundary.

- Fill the packet.
- Post the packet to IOP by writing it to inbound queue. For requests allocated in IOP memory, write the offset to inbound queue port. For requests allocated in host memory, write  $(0x80000000 | (bus\_addr \gg 5))$  to the inbound queue port.
- The IOP process the request. When the request is completed, it will be put into outbound queue. An outbound interrupt will be generated.

For requests allocated in IOP memory, the request offset is posted to outbound queue.

For requests allocated in host memory,  $(0x80000000 | (bus\_addr \gg 5))$  is posted to the outbound queue. If IOP\_REQUEST\_FLAG\_OUTPUT\_CONTEXT flag is set in the request, the low 32-bit context value will be posted instead.

- The host read the outbound queue and complete the request.

For requests allocated in IOP memory, the host driver free the request by writing it to the outbound queue.

Non-queued requests (reset/flush etc) can be sent via inbound message register 0. An outbound message with the same value indicates the completion of an inbound message.

## User-level Interface

---

The driver exposes following sysfs attributes:

NAME	R/W	Description
driver-version	R	driver version string
firmware-version	R	firmware version string

---

Copyright (C) 2006-2009 HighPoint Technologies, Inc. All Rights Reserved.

This file is distributed in the hope that it will be useful,  
but WITHOUT ANY WARRANTY; without even the implied warranty of  
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the  
GNU General Public License for more details.

linux@highpoint-tech.com  
<http://www.highpoint-tech.com>