

## Re: IOCTL\_SCSI\_GET\_ADDRESS on SAS devices?

---

*Source:*

<http://www.tech-archive.net/Archive/Development/microsoft.public.development.device.drivers/2008-04/msg00283.1>

---

- *From:* "Alexander Grigoriev" <[alegr@xxxxxxxxxxxxxx](mailto:alegr@xxxxxxxxxxxxxx)>
  - *Date:* Wed, 9 Apr 2008 07:57:53 -0700
- 

SAS is just a transport (as much as "old" SCSI, iSCSI, FCP, etc).  
SCSI\_GET\_ADDRESS is handled on the higher layer. Properly written application should not care what transport is used. It should even be able to work with ATAPI devices. No "porting" should be required.

"Peter Karlsson" <[reply.to.newsgroup@xxxxxxxxxxxxxx](mailto:reply.to.newsgroup@xxxxxxxxxxxxxx)> wrote in message [news:op.t9chyb1hx32khr@xxxxxxxxxxxxxx](mailto:news:op.t9chyb1hx32khr@xxxxxxxxxxxxxx)

Hi!

I have been given the task of porting an old SCSI application to support SAS. The underlying libraries are really old, as they were originally written to use ASPI for DOS, later ported to ASPI for Win16, then to ASPI for Win32, and through the `aspiemu.c` that was published in Dr. Drobb's in 1997, to use SPTI.

The libraries still think they are using ASPI (through `SendASPI32Command`) to a device identified by a host adapter number, a device number and a LUN.

Now, how does SAS enter in the equation? To map the SCSI id/LUN back to a device name, the `aspiemu.c` uses `IOCTL_SCSI_GET_ADDRESS` to find the device node that corresponds to the SCSI id I give it. Is it at all possible to use this scheme for SAS devices, and if so, what kind of SCSI id should I hand it?

Any tips and pointers in the right direction would be highly appreciated.

--  
\\// Peter Karlsson – [peter.karlsson@xxxxxxxxxxxxxx](mailto:peter.karlsson@xxxxxxxxxxxxxx)