

**NAME**

**twe** - 3ware 5000/6000/7000/8000 series PATA/SATA RAID adapter driver

**SYNOPSIS**

To compile this driver into the kernel, place the following lines in your kernel configuration file:

```
device pci  
device twe
```

Alternatively, to load the driver as a module at boot time, place the following line in loader.conf(5):

```
twe_load="YES"
```

**DEPRECATION NOTICE**

The **twe** driver is not present in FreeBSD 14.0.

**DESCRIPTION**

The **twe** driver provides support for AMCC's 3ware 5000/6000/7000/8000 series PATA/SATA RAID adapters. These adapters were formerly known as "3ware Escalade".

These devices support 2, 4, 8, or 12 ATA disk drives and provide RAID0 (striping) and RAID1 (mirroring) functionality.

**HARDWARE**

The **twe** driver supports the following PATA/SATA RAID controllers:

- ⊕ AMCC's 3ware 5000 series
- ⊕ AMCC's 3ware 6000 series
- ⊕ AMCC's 3ware 7000-2
- ⊕ AMCC's 3ware 7006-2
- ⊕ AMCC's 3ware 7500-4LP
- ⊕ AMCC's 3ware 7500-8
- ⊕ AMCC's 3ware 7500-12
- ⊕ AMCC's 3ware 7506-4LP
- ⊕ AMCC's 3ware 7506-8
- ⊕ AMCC's 3ware 7506-12
- ⊕ AMCC's 3ware 8006-2LP
- ⊕ AMCC's 3ware 8500-4LP
- ⊕ AMCC's 3ware 8500-8
- ⊕ AMCC's 3ware 8500-12

- ⊕ AMCC's 3ware 8506-4LP
- ⊕ AMCC's 3ware 8506-8
- ⊕ AMCC's 3ware 8506-8MI
- ⊕ AMCC's 3ware 8506-12
- ⊕ AMCC's 3ware 8506-12MI

## DIAGNOSTICS

### Controller initialisation phase

**twe%d: microcontroller not ready**

The controller's onboard CPU is not reporting that it is ready; this may be due to either a board or system failure. Initialisation has failed.

**twe%d: no attention interrupt**

**twe%d: can't drain AEN queue**

**twe%d: reset not reported**

**twe%d: controller errors detected**

**twe%d: can't drain response queue**

**twe%d: reset %d failed, trying again**

The controller is not responding correctly to the driver's attempts to reset and initialise it. This process is retried several times.

**twe%d: can't initialise controller, giving up**

Several attempts to reset and initialise the controller have failed; initialisation has failed and the driver will not attach to this controller.

### Driver initialisation/shutdown phase

**twe%d: register window not available**

**twe%d: can't allocate register window**

**twe%d: can't allocate parent DMA tag**

**twe%d: can't allocate interrupt**

**twe%d: can't set up interrupt**

**twe%d: can't establish configuration hook**

A resource allocation error occurred while initialising the driver; initialisation has failed and the driver will not attach to this controller.

**twe%d: can't detect attached units**

Fetching the list of attached units failed; initialisation has failed.

**twe%d: error fetching capacity for unit %d**  
**twe%d: error fetching state for unit %d**  
**twe%d: error fetching descriptor size for unit %d**  
**twe%d: error fetching descriptor for unit %d**  
**twe%d: device\_add\_child failed**  
**twe%d: bus\_generic\_attach returned %d**

Creation of the disk devices failed, either due to communication problems with the adapter or due to resource shortage; attachment of one or more units may have been aborted.

### Operational phase

**twe%d: command completed - %s**

A command was reported completed with a warning by the controller. The warning may be one of:

**redundant/inconsequential request ignored**  
**failed to write zeroes to LBA 0**  
**failed to profile TwinStor zones**

**twe%d: command failed - %s**

A command was reported as failed by the controller. The failure message may be one of:

**aborted due to system command or reconfiguration**  
**aborted**  
**access error**  
**access violation**  
**device failure**  
**controller error**  
**timed out**  
**invalid unit number**  
**unit not available**  
**undefined opcode**  
**request incompatible with unit**  
**invalid request**  
**firmware error, reset requested**

The command will be returned to the operating system after a fatal error.

**twe%d: command failed submission - controller wedged**

A command could not be delivered to the controller because the controller is unresponsive.

**twe%d: AEN: <%s>**

The controller has reported a change in status using an AEN (Asynchronous Event Notification). The following AENs may be reported:

**queue empty**

**soft reset**

**degraded mirror**

**controller error**

**rebuild fail**

**rebuild done**

**incomplete unit**

**initialisation done**

**unclean shutdown detected**

**drive timeout**

**drive error**

**rebuild started**

**aen queue full**

AENs are also queued internally for use by management tools.

**twe%d: error polling for signalled AENs**

The controller has reported that one or more status messages are ready for the driver, but attempting to fetch one of these has returned an error.

**twe%d: AEN queue overflow, lost AEN <%s>**

A status message was retrieved from the controller, but there is no more room to queue it in the driver. The message is lost (but will be printed to the console).

**twe%d: missing expected status bits %s****twe%d: unexpected status bits %s**

A check of the controller's status bits indicates an unexpected condition.

**twe%d: host interrupt**

The controller has signalled a host interrupt. This serves an unknown purpose and is ignored.

**twe%d: command interrupt**

The controller has signalled a command interrupt. This is not used, and will be disabled.

**twe%d: controller reset in progress...**

The controller is being reset by the driver. Typically this is done when the driver has determined that the controller is in an unrecoverable state.

**twe%d: can't reset controller, giving up**

The driver has given up on resetting the controller. No further I/O will be handled.

**controller reset done, %d commands restarted**

The controller was successfully reset, and outstanding commands were restarted.

**AUTHORS**

The **twe** driver and manual page were written by Michael Smith <[msmith@FreeBSD.org](mailto:msmith@FreeBSD.org)>.

Extensive work done on the driver by Vinod Kashyap <[vkashyap@FreeBSD.org](mailto:vkashyap@FreeBSD.org)> and Paul Saab <[ps@FreeBSD.org](mailto:ps@FreeBSD.org)>.

**BUGS**

The controller cannot handle I/O transfers that are not aligned to a 512-byte boundary. In order to support raw device access from user-space, the driver will perform alignment fixup on non-aligned data. This process is inefficient, and thus in order to obtain best performance user-space applications accessing the device should do so with aligned buffers.