

**NAME**

**devctl** - device event reporting and device control interface

**SYNOPSIS**

The **devctl** driver is automatically included in the kernel.

**DESCRIPTION**

The **devctl** device is used to report device events from the kernel. Future versions will allow for some device control as well.

**IMPLEMENTATION NOTES**

This design allows only one reader for */dev/devctl*. This is not desirable in the long run, but will get a lot of hair out of this implementation. Maybe we should make this device a clonable device.

Also note: we specifically do not attach a device to the *device\_t* tree to avoid potential chicken and egg problems. One could argue that all of this belongs to the root node. One could also further argue that the *sysctl(3)* interface that we have now might more properly be an *ioctl(2)* interface.

SIGIO support is included in the driver. However, the author is not sure that the SIGIO support is done correctly. It was copied from a driver that had SIGIO support that likely has not been tested since FreeBSD 3.4 or FreeBSD 2.2.8!

The read channel for this device is used to report changes to userland in realtime. We return one record at a time. If you try to read this device a character at a time, you will lose the rest of the data. Listening programs are expected to cope.

The *sysctl hw.bus.devctl\_queue* can be used to control queue length. It is set to 0 to disable **devctl** when no *devd(8)* is running.

**PROTOCOL**

The **devctl** device uses an ASCII protocol. The driver returns one record at a time to its readers. Each record is terminated with a newline. The first character of the record is the event type.

<i>Type</i>	<i>Description</i>
!	A notify event, such as a link state change.
+	Device node in tree attached.
-	Device node in tree detached.
?	Unknown device detected.

## Message Formats

Except for the first character in the record, attach and detach messages have the same format.

*Tdev at parent on location*

<i>Part</i>	Description
<i>T</i>	+ or -
<i>dev</i>	The device name that was attached/detached.
<i>parent</i>	The device name of the parent bus that attached the device.
<i>location</i>	Bus specific location information.

The nomatch messages can be used to load devices driver. If you load a device driver, then one of two things can happen. If the device driver attaches to something, you will get a device attached message. If it does not, then nothing will happen.

The attach and detach messages arrive after the event. This means one cannot use the attach message to load an alternate driver. The attach message driver has already claimed this device. One cannot use the detach messages to flush data to the device. The device is already gone.

All values passed back are of the form 'key=value' or 'key="value"'. When the latter, the string "value" must have any internal backslashes doubled. It must also have any internal double quote characters ' preceded by a backslash. All other characters should be passed through.

## SEE ALSO

devd(8)