NAME

trace − event-tracing driver

DESCRIPTION

Trace is a special file that allows event records generated within the UNIX kernel to be passed to a user program so that the activity of a driver or other system routines can be monitored for debugging purposes.

An event record is generated from within a kernel driver or system routine by invoking the *trsave* function:

```
trsave(dev, chno, buf, cnt)
char   dev, chno, *buf, cnt;
```

Dev is a minor device number of the trace driver; *chno* is an integer between 0 and 15 inclusive that identifies the data stream (channel) to which the record belongs; *buf* is a buffer containing the bytes that make up a single event record, and *cnt* is the number of bytes in *buf*. Calls to *trsave* will result in data being placed on a queue, provided that some user program has opened the trace minor device *dev* and has enabled channel *chno*. Event records prefaced by *chno* and *cnt* are stored on a queue until a system-defined maximum (TRQMAX) is reached; an event record is discarded if there is not sufficient room on the queue for the entire record. This implies that event records with *cnt* > TRQMAX − 2 are discarded. The queue is emptied by a user program reading the trace driver. Each *read* returns an integral number of event records; the read count must, therefore, be at least equal to the size of a record plus two.

The *trace* driver supports *open*, *close*, *read*, and *ioctl* system calls. The *ioctl* system call is invoked as follows:

```
#include <sys/vpm.h>
int fildes, cmd, arg;
ioctl(fildes, cmd, arg);
```

The *trace ioctl* commands are:

VPMSETC Enable trace channels. This command enables the channels indicated by a 1 in the bit mask found in *arg*. The low-order bit (bit 0) corresponds to channel zero, the next bit (bit 1) corresponds to channel 1, etc..

VPMGETC Get enabled channels. This command returns in *arg* a bit mask containing a 1 for each channel that is currently enabled.

VPMCLRC Disable channels. This command disables the channels indicated by a 1 in the bit mask found in *arg*.

SEE ALSO

vpmstart(1C), vpm(4).