



**Full credit is given to the above companies including the Operating System (OS) that this PDF file was generated!**

### ***Rocky Enterprise Linux 9.2 Manual Pages on command 'dsp56k.4'***

***\$ man dsp56k.4***

DSP56K(4)                      Linux Programmer's Manual                      DSP56K(4)

#### **NAME**

dsp56k - DSP56001 interface device

#### **SYNOPSIS**

```
#include <asm/dsp56k.h>

ssize_t read(int fd, void *data, size_t length);

ssize_t write(int fd, void *data, size_t length);

int ioctl(int fd, DSP56K_UPLOAD, struct dsp56k_upload *program);

int ioctl(int fd, DSP56K_SET_TX_WSIZE, int wsize);

int ioctl(int fd, DSP56K_SET_RX_WSIZE, int wsize);

int ioctl(int fd, DSP56K_HOST_FLAGS, struct dsp56k_host_flags *flags);

int ioctl(int fd, DSP56K_HOST_CMD, int cmd);
```

#### **CONFIGURATION**

The dsp56k device is a character device with major number 55 and minor number 0.

#### **DESCRIPTION**

The Motorola DSP56001 is a fully programmable 24-bit digital signal processor found in Atari Falcon030-compatible computers. The dsp56k special file is used to control the DSP56001, and to send and receive data using the bidirectional handshaked host port. To send a data stream to the signal processor, use write(2) to the device, and read(2) to receive processed data. The data can be sent or received in 8, 16, 24, or 32-bit quantities on the host side, but will always be seen as 24-bit quantities in the DSP56001.

The following ioctl(2) calls are used to control the dsp56k device:

DSP56K\_UPLOAD

resets the DSP56001 and uploads a program. The third `ioctl(2)` argument must be a pointer to a struct `dsp56k_upload` with members `bin` pointing to a DSP56001 binary program, and `len` set to the length of the program, counted in 24-bit words.

#### DSP56K\_SET\_TX\_WSIZE

sets the transmit word size. Allowed values are in the range 1 to 4, and is the number of bytes that will be sent at a time to the DSP56001. These data quantities will either be padded with bytes containing zero, or truncated to fit the native 24-bit data format of the DSP56001.

#### DSP56K\_SET\_RX\_WSIZE

sets the receive word size. Allowed values are in the range 1 to 4, and is the number of bytes that will be received at a time from the DSP56001. These data quantities will either truncated, or padded with a null byte ('\0') to fit the native 24-bit data format of the DSP56001.

#### DSP56K\_HOST\_FLAGS

read and write the host flags. The host flags are four general-purpose bits that can be read by both the hosting computer and the DSP56001. Bits 0 and 1 can be written by the host, and bits 2 and 3 can be written by the DSP56001.

To access the host flags, the third `ioctl(2)` argument must be a pointer to a struct `dsp56k_host_flags`. If bit 0 or 1 is set in the `dir` member, the corresponding bit in `out` will be written to the host flags. The state of all host flags will be returned in the lower four bits of the `status` member.

#### DSP56K\_HOST\_CMD

sends a host command. Allowed values are in the range 0 to 31, and is a user-defined command handled by the program running in the DSP56001.

## FILES

`/dev/dsp56k`

## SEE ALSO

`linux/include/asm-m68k/dsp56k.h`, `linux/drivers/char/dsp56k.c`, [?http://dsp56k.nocrew.org/?](http://dsp56k.nocrew.org/),  
DSP56000/DSP56001 Digital Signal Processor User's Manual

## COLOPHON

This page is part of release 5.10 of the Linux man-pages project. A description of the project, information about reporting bugs, and the latest version of this page, can be found at <https://www.kernel.org/doc/man-pages/>.

