

NAME

sysdecode_enum, **sysdecode_acltype**, **sysdecode_atfd**, **sysdecode_extattrnamespace**, **sysdecode_fadvice**, **sysdecode_fcntl_cmd**, **sysdecode_getfsstat_mode**, **sysdecode_getrusage_who**, **sysdecode_idtype**, **sysdecode_ipproto**, **sysdecode_itimer**, **sysdecode_kldsym_cmd**, **sysdecode_kldunload_flags**, **sysdecode_lio_listio_mode**, **sysdecode_madvise**, **sysdecode_minherit_flags**, **sysdecode_msgctl_cmd**, **sysdecode_nfssvc_flags**, **sysdecode_pathconf_name**, **sysdecode_prio_which**, **sysdecode_procctl_cmd**, **sysdecode_ptrace_request**, **sysdecode_rlimit**, **sysdecode_rtprio_function**, **sysdecode_scheduler_policy**, **sysdecode_sctp_pr_policy**, **sysdecode_sctp_sinfo_flags**, **sysdecode_semctl_cmd**, **sysdecode_shmctl_cmd**, **sysdecode_shutdown_how**, **sysdecode_sigbus_code**, **sysdecode_sigchld_code**, **sysdecode_sigfpe_code**, **sysdecode_sigill_code**, **sysdecode_signal**, **sysdecode_sigprocmask_how**, **sysdecode_sigsegv_code**, **sysdecode_sigtrap_code**, **sysdecode_sockaddr_family**, **sysdecode_socketdomain**, **sysdecode_sockettype**, **sysdecode_sockopt_level**, **sysdecode_sysarch_number**, **sysdecode_umtx_op**, **sysdecode_vmresult**, **sysdecode_whence** - lookup name of various enumerated values

LIBRARY

System Argument Decoding Library (libsysdecode, -lsysdecode)

SYNOPSIS

```
#include <sysdecode.h>
```

```
const char *
```

```
sysdecode_acltype(int type);
```

```
const char *
```

```
sysdecode_atfd(int fd);
```

```
const char *
```

```
sysdecode_extattrnamespace(int namespace);
```

```
const char *
```

```
sysdecode_fadvice(int advice);
```

```
const char *
```

```
sysdecode_fcntl_cmd(int cmd);
```

```
const char *
```

```
sysdecode_getfsstat_mode(int mode);
```

```
const char *
```

sysdecode_getrusage_who(*int who*);

*const char **

sysdecode_idtype(*int idtype*);

*const char **

sysdecode_ipproto(*int protocol*);

*const char **

sysdecode_itimer(*int which*);

*const char **

sysdecode_kldsym_cmd(*int cmd*);

*const char **

sysdecode_kldunload_flags(*int flags*);

*const char **

sysdecode_lio_listio_mode(*int mode*);

*const char **

sysdecode_madvise(*int advice*);

*const char **

sysdecode_minherit_flags(*int inherit*);

*const char **

sysdecode_msgctl_cmd(*int cmd*);

*const char **

sysdecode_nfssvc_flags(*int flags*);

*const char **

sysdecode_pathconf_name(*int name*);

*const char **

sysdecode_prio_which(*int which*);

*const char **

sysdecode_procctl_cmd(*int cmd*);

*const char **
sysdecode_ptrace_request(*int request*);

*const char **
sysdecode_rlimit(*int resource*);

*const char **
sysdecode_rtprio_function(*int function*);

*const char **
sysdecode_scheduler_policy(*int policy*);

*const char **
sysdecode_sctp_pr_policy(*int policy*);

*const char **
sysdecode_semctl_cmd(*int cmd*);

*const char **
sysdecode_shmctl_cmd(*int cmd*);

*const char **
sysdecode_shutdown_how(*int how*);

*const char **
sysdecode_sibus_code(*int si_code*);

*const char **
sysdecode_sigchld_code(*int si_code*);

*const char **
sysdecode_sigfpe_code(*int si_code*);

*const char **
sysdecode_sigill_code(*int si_code*);

*const char **
sysdecode_signal(*int sig*);

*const char **

sysdecode_sigprocmask_how(*int how*);

*const char **

sysdecode_sigsegv_code(*int si_code*);

*const char **

sysdecode_sigtrap_code(*int si_code*);

*const char **

sysdecode_sockaddr_family(*int sa_family*);

*const char **

sysdecode_socketdomain(*int domain*);

*const char **

sysdecode_sockettype(*int type*);

*const char **

sysdecode_sockopt_level(*int level*);

*const char **

sysdecode_sysarch_number(*int number*);

*const char **

sysdecode_umtx_op(*int op*);

*const char **

sysdecode_vmresult(*int result*);

*const char **

sysdecode_whence(*int whence*);

DESCRIPTION

The **sysdecode_enum** functions return a text description of an integer value. The text description matches the name of a C macro with the same value as the sole function argument. NULL is returned if there is no matching C macro name.

Most of these functions decode an argument passed to a system call:

| Function | System Call | Argument |
|----------|-------------|----------|
|----------|-------------|----------|

| | | |
|-------------------------------------|----------------------------|----------------------|
| sysdecode_acltype() | acl_get_file(3) | <i>type</i> |
| sysdecode_atfd() | openat(2) | <i>fd</i> |
| sysdecode_extattrnamespace() | extattr_get_fd(2) | <i>attrnamespace</i> |
| sysdecode_fadvice() | posix_fadvise(2) | <i>advice</i> |
| sysdecode_fcntl_cmd() | fcntl(2) | <i>cmd</i> |
| sysdecode_getfsstat_mode() | getfsstat(2) | <i>mode</i> |
| sysdecode_idtype() | procctl(2), waitid(2) | <i>idtype</i> |
| sysdecode_itimer() | getitimer(2), setitimer(2) | <i>which</i> |
| sysdecode_kldsym_cmd() | kldsym(2) | <i>cmd</i> |
| sysdecode_kldunload_flags() | kldunloadf(2) | <i>flags</i> |
| sysdecode_lio_listio_mode() | lio_listio(2) | <i>mode</i> |
| sysdecode_madvise() | madvise(2) | <i>advice</i> |
| sysdecode_minherit_inherit() | minherit(2) | <i>inherit</i> |
| sysdecode_msgctl_cmd() | msgctl(2) | <i>cmd</i> |
| sysdecode_nfssvc_flags() | nfssvc(2) | <i>flags</i> |
| sysdecode_pathconf_name() | pathconf(2) | <i>name</i> |
| sysdecode_prio_which() | getpriority(2) | <i>which</i> |
| sysdecode_procctl_cmd() | procctl(2) | <i>cmd</i> |
| sysdecode_ptrace_request() | ptrace(2) | <i>request</i> |
| sysdecode_rlimit() | getrlimit(2) | <i>resource</i> |
| sysdecode_rtprio_function() | rtprio(2) | <i>function</i> |
| sysdecode_getrusage_who() | getrusage(2) | <i>who</i> |
| sysdecode_scheduler_policy() | sched_setscheduler(2) | <i>policy</i> |
| sysdecode_semctl_cmd() | semctl(2) | <i>cmd</i> |
| sysdecode_shmctl_cmd() | shmctl(2) | <i>cmd</i> |
| sysdecode_shutdown_how() | shutdown(2) | <i>how</i> |
| sysdecode_sigprocmask_how() | sigprocmask(2) | <i>how</i> |
| sysdecode_sockopt_level() | getsockopt(2) | <i>level</i> |
| sysdecode_sysarch_number() | sysarch(2) | <i>number</i> |
| sysdecode_umtx_op() | _umtx_op(2) | <i>op</i> |
| sysdecode_whence() | lseek(2) | <i>whence</i> |

These functions decode signal-specific signal codes stored in the *si_code* field of the *siginfo_t* object associated with an instance of signal:

| Function | Signal |
|---------------------------------|---------------|
| sysdecode_sigbus_code() | SIGBUS |
| sysdecode_sigchld_code() | SIGCHLD |
| sysdecode_sigfpe_code() | SIGFPE |
| sysdecode_sigill_code() | SIGILL |

sysdecode_sigsegv_code() SIGSEGV
sysdecode_sigtrap_code() SIGTRAP

Other functions decode the values described below:

sysdecode_ipproto() An IP protocol.

sysdecode_sctp_pr_policy() A PR-SCTP policy.

sysdecode_signal() A process signal.

sysdecode_sockaddr_family() A socket address family.

sysdecode_socketdomain() A socket domain.

sysdecode_vmresult() The return value of a function in the virtual memory subsystem of the kernel indicating the status of the associated request.

RETURN VALUES

The **sysdecode_enum** functions return the name of a matching C macro or NULL if no matching C macro was found.

SEE ALSO

sysdecode(3), sysdecode_mask(3), sysdecode_sigcode(3)