

**NAME**

**elf\_cntl** - control an elf file descriptor

**LIBRARY**

ELF Access Library (libelf, -lelf)

**SYNOPSIS**

```
#include <libelf.h>
```

*int*

```
elf_cntl(Elf *elf, Elf_Cmd cmd);
```

**DESCRIPTION**

Function **elf\_cntl()** controls the ELF library's subsequent use of the file descriptor used to create ELF descriptor *elf*.

Argument *cmd* informs the library of the action to be taken:

**ELF\_C\_FDDONE** This value instructs the ELF library not to perform any further I/O on the file descriptor associated with argument *elf*. For ELF descriptors opened with mode *ELF\_C\_WRITE* or *ELF\_C\_RDWR* subsequent **elf\_update()** operations on the descriptor will fail.

**ELF\_C\_FDREAD** This value instructs the ELF library to read in all necessary data associated with ELF descriptor *elf* into memory so that the underlying file descriptor can be safely closed with command **ELF\_C\_FDDONE**.

Argument *elf* must be an ELF descriptor associated with a file system object (e.g., an ar(1) archive, an ELF file, or other data file).

**IMPLEMENTATION NOTES**

Due to use of `mmap(2)` internally, this function is a no-op for ELF objects opened in **ELF\_C\_READ** mode.

**RETURN VALUES**

Function **elf\_cntl()** returns 0 on success, or -1 if an error was detected.

**ERRORS**

[**ELF\_E\_ARCHIVE**] Argument *elf* is a descriptor for an archive member.

[ELF\_E\_ARGUMENT]

Argument *elf* was NULL.

[ELF\_E\_ARGUMENT]

Argument *cmd* was not recognized.

[ELF\_E\_MODE]

An ELF\_C\_FDREAD operation was requested on an ELF descriptor opened for writing.

**SEE ALSO**

elf(3), elf\_begin(3), elf\_end(3), elf\_next(3), elf\_update(3), gelf(3)