```
/*    Copyright (c) 1990, 1991, 1992, 1993 UNIX System Laboratories, Inc.    */
/*    Copyright (c) 1988, 1990 AT&T    */
/*      All Rights Reserved      */

/*    THIS IS UNPUBLISHED PROPRIETARY SOURCE CODE OF         */
/*    UNIX System Laboratories, Inc.                         */
/*    The copyright notice above does not evidence any       */
/*    actual or intended publication of such source code.    */

#ident    "@(#)libelf:common/flag.c    1.6"


#ifdef __STDC__
    #pragma weak    elf_flagdata = _elf_flagdata
    #pragma weak    elf_flagehdr = _elf_flagehdr
    #pragma weak    elf_flagelf = _elf_flagelf
    #pragma weak    elf_flagphdr = _elf_flagphdr
    #pragma weak    elf_flagscn = _elf_flagscn
    #pragma weak    elf_flagshdr = _elf_flagshdr
#endif


#include "syn.h"
#include "libelf.h"
#include "decl.h"
#include "error.h"

unsigned
elf_flagdata(data, cmd, flags)
    Elf_Data    *data;
    Elf_Cmd       cmd;
    unsigned    flags;
{
    if (data == 0)
        return 0;
    if (cmd == ELF_C_SET)
        return ((Dnode *)data)->db_uflags |= flags;
    if (cmd == ELF_C_CLR)
        return ((Dnode *)data)->db_uflags &= ~flags;
    _elf_err = EREQ_FLAG;
    return 0;
}

unsigned int
elf_flagehdr(elf, cmd, flags)
    Elf       *elf;
    Elf_Cmd       cmd;
    unsigned    flags;
{
    if (elf == 0)
        return 0;
    if (cmd == ELF_C_SET)
        return elf->ed_ehflags |= flags;
    if (cmd == ELF_C_CLR)
        return elf->ed_ehflags &= ~flags;
    _elf_err = EREQ_FLAG;
    return 0;
}

unsigned
elf_flagelf(elf, cmd, flags)
    Elf       *elf;
    Elf_Cmd       cmd;
    unsigned    flags;
{
    if (elf == 0)
        return 0;
    if (cmd == ELF_C_SET)
        return elf->ed_uflags |= flags;
    if (cmd == ELF_C_CLR)
        return elf->ed_uflags &= ~flags;
    _elf_err = EREQ_FLAG;
    return 0;
}

```



```
/*
flag.c - implementation of the elf_flag*(3) functions.
Copyright (C) 1995 Michael Riepe <riepe@ifwsn4.ifw.uni-hannover.de>

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Library General Public
License as published by the Free Software Foundation; either
version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
Library General Public License for more details.

You should have received a copy of the GNU Library General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA.
*/

#include <private.h>

static unsigned
_elf_flag(unsigned *f, Elf_Cmd cmd, unsigned flags) {
    if (cmd == ELF_C_SET) {
    return *f |= flags;
    }
    if (cmd == ELF_C_CLR) {
    return *f &= ~flags;
    }
    seterr(ERROR_INVALID_CMD);
    return 0;
}

unsigned
elf_flagdata(Elf_Data *data, Elf_Cmd cmd, unsigned flags) {
    Scn_Data *sd = (Scn_Data*)data;

    if (!sd) {
    return 0;
    }
    elf_assert(sd->sd_magic == DATA_MAGIC);
    return _elf_flag(&sd->sd_data_flags, cmd, flags);
}

unsigned
elf_flagehdr(Elf *elf, Elf_Cmd cmd, unsigned flags) {
    if (!elf) {
    return 0;
    }
    elf_assert(elf->e_magic == ELF_MAGIC);
    return _elf_flag(&elf->e_ehdr_flags, cmd, flags);
}

unsigned
elf_flagelf(Elf *elf, Elf_Cmd cmd, unsigned flags) {
    if (!elf) {
    return 0;
    }
    elf_assert(elf->e_magic == ELF_MAGIC);
    return _elf_flag(&elf->e_elf_flags, cmd, flags);
}

unsigned
elf_flagphdr(Elf *elf, Elf_Cmd cmd, unsigned flags) {
    if (!elf) {
    return 0;
    }
    elf_assert(elf->e_magic == ELF_MAGIC);
    return _elf_flag(&elf->e_phdr_flags, cmd, flags);
}

unsigned
elf_flagscn(Elf_Scn *scn, Elf_Cmd cmd, unsigned flags) {
    if (!scn) {
    return 0;
    }
    elf_assert(scn->s_magic == SCN_MAGIC);
    return _elf_flag(&scn->s_scn_flags, cmd, flags);
```


(For readability, not all matches are indicated.)

```
 79:
 80: unsigned
 81: elf_flagphdr(elf, cmd, flags)
 82:     Elf         *elf;
 83:     Elf_Cmd         cmd;
 84:     unsigned    flags;
 85: {
 86:     if (elf == 0)
 87:         return 0;
 88:     if (cmd == ELF_C_SET)
 89:         return elf->ed_phflags |= flags;
 90:     if (cmd == ELF_C_CLR)
 91:         return elf->ed_phflags &= ~flags;
 92:     _elf_err = EREQ_FLAG;
 93:     return 0;
 94: }
 95:
 96:
 97: unsigned
 98: elf_flagscn(scn, cmd, flags)
 99:     Elf_Scn         *scn;
100:     Elf_Cmd         cmd;
101:     unsigned    flags;
102: {
103:     if (scn == 0)
104:         return 0;
105:     if (cmd == ELF_C_SET)
106:         return scn->s_uflags |= flags;
107:     if (cmd == ELF_C_CLR)
108:         return scn->s_uflags &= ~flags;
109:     _elf_err = EREQ_FLAG;
110:     return 0;
111: }
112:
113:
114: unsigned
115: elf_flagshdr(scn, cmd, flags)
116:     Elf_Scn         *scn;
117:     Elf_Cmd         cmd;
118:     unsigned    flags;
119: {
120:     if (scn == 0)
121:         return 0;
122:     if (cmd == ELF_C_SET)
123:         return scn->s_shflags |= flags;
124:     if (cmd == ELF_C_CLR)
125:         return scn->s_shflags &= ~flags;
126:     _elf_err = EREQ_FLAG;
127:     return 0;
128: }
```



```
 79: }
 80:
 81: unsigned
 82: elf_flagshdr(Elf_Scn *scn, Elf_Cmd cmd, unsigned flags) {
 83:     if (!scn) {
 84:     return 0;
 85:     }
 86:     elf_assert(scn->s_magic == SCN_MAGIC);
 87:     return _elf_flag(&scn->s_shdr_flags, cmd, flags);
 88: }
```


(For readability, not all matches are indicated.)