Identification

Summary of Calls within the Basic File System
R. C. Daley

Purpose

This section summarizes all of the calls to the major modules of the basic file system. The calls outlined in this section are fully specified in other sections of this manual. The intent of this section is to provide a working document which can be updated quickly with a minimum of effort.

Segment Control

The calls to segment control are outlined below. These calls are specified in detail in section BG.3 of this manual.

1. The System Interface Module

makeknown(name, id, mode, ptlist, dirsw, dtbm, dp, slot, dhs, rsw, segptr, slotlist)

Parameters: name char(*), id bit(70), mode bit(5), ptlist(*) bit(18), dirsw bit(1), dtbm bit(52), dp ptr, slot fixed bin(17), (dhs, rsw) bit(1), segptr ptr, slotlist(*) bit (18)

Called by: estblseg, deletry, finddir, appendb

Calls: sum$nsrchkst, sum$idsrchkst

sim1$getdirseq(name, segptr, mode)

Parameters: name char(*), segptr ptr, mode bit(5)

Called by: findentry, readacl, movefile, list_dir, getentry, putentry, appendb

Calls: sum$nsrchkst, finddir
MULTICS SYSTEM-PROGRAMMERS' MANUAL

SECTION BG.14

PAGE 2

makeunknown(segptr)

Parameters: segptr ptr

Called by: list_dir, appendb, delentry, readacl, writeacl, movefile, getentry, putentry, traphan, administrative-ring procedures

Calls: sum$searchast, setfaults, checkentry, sum$nsrchkst, sum$idsrchkst

segfault(scuptr,dbr,ringno)

Parameters: scuptr ptr, dbr bit(36), ringno fixed bin(17)

Called by: system fault interceptor, boundfault

Calls: boundfault, sum$searchast, getastentry, refindb, maketrailer, pcreadseg, checkentry

sim1$moveseg(segptr,did)

Parameters: segptr ptr, did bit(17)

Called by: multilevel processes

Calls: getastentry, pcreadseg, checkentry

boundfault(scuptr,dbr,ringno)

Parameters: scuptr ptr, dbr bit(36), ringno fixed bin(17)

Called by: segfault, system fault interceptor

Calls: sum$searchast, checkentry, segfault

initialize_kst

Parameters: none

Called by: loadsegs

Calls: none
sim2$dirmod(dp)
Parameters: dp ptr
Called by: chname, writeacl, set$bc, setml, set$copy, set$rd, appendl, appendb, movefile, setsystrap, setlimits, removeb
Calls: getastentry, updates, checkentry

sim2$branchmod(id,dtbm)
Parameters: id bit(70), dtbm bit(52)
Called by: writeacl, setsystrap
Calls: sum$searchast, setfaults, checkentry

sim2$updateb(id,dp)
Parameters: id bit(70), dp ptr
Called by: list_dir, status, getentry
Calls: sum$searchast, activinfo$wrbranch, checkentry

sim2$unloadseg(id,deactsw)
Parameters: id bit(70), deactsw bit(1)
Called by: setml, movefile
Calls: sum$searchast, setfaults, pcfreecore, getastentry$delastentry

sim2$deleteseg(segptr)
Parameters: segptr ptr
Called by: delentry
Calls: getastentry, setfaults, pctruncate, sim2$unloadseg
changing(oldring, newring, dbr, callptr)
Parameters: (oldring newring) fixed bin(17), dbr bit(36),
callptr ptr
Called by: ring-crossing procedures
Calls: groupstat, alloc_sst, assign

sim2$transuse(segptr, tus)
Parameters: segptr ptr, tus bit(1)
Called by: backup and multilevel processes
Calls: none

2. The User Interface Module

uim$read_seg(addr_ptr, nwords)
Parameters: addr_ptr ptr, nwords fixed bin(18)
Called by: user
Calls: getastentry, pcreadseg, checkentry

write_seg(addr_ptr, array, nwords)
Parameters: addr_ptr ptr, array ptr, nwords fixed bin(18)
Called by: user
Calls: getastentry, refrindb, getloaded, checkentry

uim$free_core(addr_ptr, nwords)
Parameters: addr_ptr ptr, nwords fixed bin(18)
Called by: user
Calls: sum$searchast, pcfreecore, checkentry
uim$truncate_seg(addr_ptr)
Parameters: addr_ptr ptr
Called by: user
Calls: getastentry, pctruncate, checkentry

uim$core_test(addr_ptr, nwords) value= percent
Parameters: addr_ptr ptr, nwords fixed bin(18), percent fixed bin(17)
Called by: user
Calls: sum$searchast, pctestcore, checkentry

uim$check_access(segptr, ringno) value= mode
Parameters: segptr ptr, ringno fixed bin(17), mode bit(5)
Called by: user
Calls: none

3. The Process Load Module

actproc(dirname, processid) value= pstptr
Parameters: dirname char(*), processid bit(36), pstptr ptr
Called by: hard-core-ring procedures
Calls: estblseg, getastentry, checkentry, alloc_sst, makeunknown

deactproc(pstptr)
Parameters: pstptr ptr
Called by: hard-core-ring procedures
Calls: checkentry
**loadproc**(pstptr)
Parameters:  pstptr  ptr  
Called by:  hard-core-ring procedures  
Calls:  getloaded, checkentry, alloc_sst  

**loadsegs**(pstptr)
Parameters:  pstptr  ptr  
Called by:  hard-core-ring procedures  
Calls:  getastentry, pcreadseg, checkentry, alloc_sst, changering, initialize_kst  

**unloadproc**(pstptr)
Parameters:  pstptr  ptr  
Called by:  hard-core-ring procedures  
Calls:  groupstat, checkentry  

**createseg**(size, maxl, priority, descr)
Parameters:  (size maxl) fixed bin(18), priority fixed bin(17), descr bit(36)  
Called by:  hard-core-ring procedures  
Calls:  assign  

**killseg**(descr)
Parameters:  descr bit(36)  
Called by:  hard-core-ring procedures  
Calls:  unassign
4. The Segment Utility Module

**sum$searcbast**(id, found, hsi, astptr)

Parameters: id bit(70), found bit(1), hsi fixed bin(17), astptr ptr

Called by: makeunknown, boundfault, sim2$branchmod, sim2$updateb, sim2$unloadseg, uim$free_core, uim$core_test, getastentry

Calls: none

**getastentry**(kstptr, did) value= astptr

Parameters: kstptr ptr, did bit(17), astptr ptr

Called by: segfault, sim1$moveseg, sim2$dirmod, sim2$deleteseg, uim$readseg, write_seg, uim$truncate_seg, actproc, loadsegs, getastentry

Calls: sum$searcbast, getastentry, checkentry, alloc_sst, activinfo$rdbranch, multilevel move control, accounting routines

**maketrailer**(astptr, segno)

Parameters: astptr ptr, segno bit(18)

Called by: segfault

Calls: alloc_sst, checkentry

**getastentry$delastentry**(astptr, hsi)

Parameters: astptr ptr, hsi fixed bin(17)

Called by: sim2$unloadseg, alloc_sst

Calls: cleanup, getloaded, activinfo$wrbranch, checkentry, alloc_sst$free_sst, accounting routines
**alloc_sst(type, ptr)**

Parameters: type fixed bin(17), ptr ptr

Called by: changering, actproc, loadproc, loadsegs, getastentry, maketrailer

Calls: sum$searchast, getastentry$delastentry

**alloc_sst$free_sst(type, ptr)**

Parameters: type fixed bin(17), ptr ptr

Called by: deactproc, unloadproc, getastentry$delastentry

Calls: none

**sum$nsrchkst(name, found, hsi, segno, ep)**

Parameters: name char(*), found bit(1), hsi fixed bin(17), segno bit(18), ep ptr

Called by: makeknown, sim1$getdirseg, makeunknown

Calls: none

**sum$idsrchkst(id, found, hsi, segno, ep)**

Parameters: id bit(70), found bit(1), hsi fixed bin(17), segno bit(18), ep ptr

Called by: makeknown, makeunknown

Calls: none

**sum$searchhst(id, found, hsi, segno)**

Parameters: id bit(70), found bit(1), hsi fixed bin(17), segno bit(18)

Called by: makeknown

Calls: none
The calls to page control are outlined below. These calls are specified in detail in section BG.4 of this manual.

**pagefault** (scuptr, dbr)
Parameters: scuptr ptr, dbr bit(36)
Called by: system fault interceptor
Calls: assign, updates, new_io, accounting routines

**pcreadseg** (astptr, addr, nwords, blocksw)
Parameters: astptr ptr, (addr nwords) fixed bin(18), blocksw bit(1)
Called by: segfault, sim$moveseq, uim$read_seg, loadsegs
Calls: getloaded, assign, updates, new_io, accounting routines

**pctruncate** (astptr, addr)
Parameters: astptr ptr, addr fixed bin(18)
Called by: sim2$deleteseg, uim$truncate_seg
Calls: setfaults, getloaded, unassign, new_io, updates, accounting routines

**removepage** (sstptr, dssw, pageno, retarg)
Parameters: sstptr ptr, dssw bit(1), pageno fixed bin(17), retarg fixed bin(17)
Called by: assign
Calls: new_io, updates, assign, checkentry

**pcfreecore** (sstptr, dssw, addr, nwords, blocksw)
Parameters: sstptr ptr, dssw bit(1), (addr nwords) fixed bin(18), blocksw bit(1)
Called by: uim$free_core, getastentry$delastentry
Calls: setfaults, unassign, new_io, updates, assign
pctestcore(astptr, addr, nwords) value= percent
Parameters: astptr ptr, (addr, nwords) fixed bin(18), percent
fixed bin(17)
Called by: uim$core_test
Calls: none

cleanup(astptr)
Parameters: astptr ptr
Called by: getastentry$delastentry
Calls: pcfreecore, new_io

updates(astptr, modsw)
Parameters: astptr ptr, modsw bit(1)
Called by: sim2$dirmod, pagefault, pcreadseg, pctruncate, removepage, pcfreecore
Calls: none

setfaults(entryptr, all)
Parameters: entryptr ptr, all bit(1)
Called by: makeunknown, sim2$branchmod, sim2$deleteseg, getastentry$delastentry, checkentry
Calls: none

getloaded(astptr)
Parameters: astptr ptr
Called by: write_segment, loadproc, getastentry$delastentry, pcreadseg, pctruncate
Calls: assign
checkentry (sstptr, dssw)
Parameters: sstptr ptr, dssw bit(1)

Called by: makeunknown, segfault, sim1$moveseq, boundfault, sim2$dirmod, sim2$branchmod, updateb, uim$read_seg, write_seg, uim$free_core, uim$truncate_seg, uim$core_test, actproc, deactproc, loadproc, loadsegs, unloadproc, getastentry, removepage, iodone

Calls: setfaults, unassign

iodone (astptr, aftptr, opcode, stateword, iocode)
Parameters: (astptr, aftptr) ptr, opcode bit(3), stateword bit(36), iocode bit(18)

Called by: file system device interface modules

Calls: groupstat, unassign, checkentry
Core Control

The calls to core control are outlined below. These calls are specified in detail in section BG.6 of this manual.

\[ \text{assign}(\text{type}, \text{size}, \text{status}, \text{pagsiz}, \text{pageno}, \text{pool}, \text{dssw}, \text{procsw}, \text{sstptr}, \text{threshold}, \text{retopt}) \quad \text{value} = \text{coreloc} \]

Parameters: type bit(2), size fixed bin(17), status bit(2), pagsiz bit(1), (pageno pool) fixed bin(17), (dssw, procsw) bit(1), sstptr ptr, threshold fixed bin(17), retopt bit(3), coreloc bit(18)

Called by: changering, createseg, getloaded, pagefault, pcreadseg, removepage, pcfreecore

Calls: removepage

\[ \text{unassign}(\text{coreloc}, \text{ifstatus}) \]

Parameters: coreloc bit(18), ifstatus bit(2)

Called by: killseg, pctruncate, pcfreecore, checkentry, iodone

Calls: none

\[ \text{groupstat}(\text{coreloc}, \text{changestat}, \text{newstat}) \quad \text{value} = \text{oldstat} \]

Parameters: coreloc bit(18), (changestat newstat oldstat) bit(2)

Called by: changering, unloadproc, iodone

Calls: none

\[ \text{poolsizes}(\text{pool}, \text{poolmax}) \]

Parameters: (pool poolmax) fixed bin(17)

Called by: use of this call is as yet unspecified

Calls: none
Directory Control

The calls to directory control are outlined below. These calls are specified in detail in section BG.8 of this manual.

1. The Directory Supervisor

`list_dir` (dirname, areap, branchp, branchct, linkp, linkct)

Parameters: `dirname` char(*), (areap, branchp, linkp) ptr, (branchct, linkct) fixed bin(17)

Called by: user

Calls: `sim1$getdirseg`, `sim2$updateb`, `makeunknown`, `packer$packb`, `packer$pack1`

`status` (dirname, entry, type, areap, entryp)

Parameters: `dirname` char(*), `entry` char(*), `type` bit(1), (areap, entryp) ptr

Called by: user

Calls: `findentry`, `sim2$updateb`, `packer$packb`, `packer$pack1`

`chname` (dirname, entry, oldname, newname)

Parameters: `(dirname entry oldname newname)` char(*)

Called by: user

Calls: `findentry`, `hash$in`, `hash$out`, `sim2$dirmod`

`dele` (dirname, entry, csw)

Parameters: `(dirname, entry)` char(*), `csw` bit(2)

Called by: user

Calls: `findentry`, `effmode`, `removel`, `makeknown`, `sim2$deleteseg`, `makeunknown`, `removeb`
readacl (dirname, -entry-, areap, accessp)
Parameters: (dirname, entry) char(*), (areap, accessp) ptr
Called by: user
Calls: findbranch, sim1$getdirseg, makeunknown

writeacl (dirname, -entry-, accessp)
Parameters: (dirname, entry) char(*), accessp ptr
Called by: user
Calls: findbranch, sim1$getdirseg, sim2$dirmod, sim2$branchmod, makeunknown

appendb (dirname, entry, type, mode, optsw, max)
Parameters: (dirname, entry) char(*), type bit(1), mode bit(5), optsw bit(2), max bit(8)
Called by: user
Calls: findentry, hash$in, makeknown, makeunknown, sim2$dirmod

appendl (dirname, entry, pathname)
Parameters: (dirname, entry, pathname) char(*)
Called by: user
Calls: findentry, hash$in, sim2$dirmod

movefile (dirname, entry, csw, newdir, newname)
Parameters: (dirname entry newdir newname) char(*), csw bit(2)
Called by: user
Calls: findbranch, effmode, sim2$unloadseg, sim1$getdirseg, sim2$dirmod, makeunknown
estblseg (dirname, entry, segptr, optionsw, slotlist)
Parameters: (dirname, entry) char(*), segptr ptr, optionsw bit(3), slotlist(*) fixed bin(17)
Called by: actproc, traphan, administrative-ring procedures
Calls: findbranch, effmode, makeknown

setml (dirname, entry, max)
Parameters: (dirname, entry) char(*), max bit(8)
Called by: user
Calls: findbranch, sim2$dirmod, sim2$unloadseg

set$bc (dirname, entry, bitct)
Parameters: (dirname, entry) char(*), bitct bit(24)
Called by: user
Calls: findbranch, effmode, sim2$dirmod

set$copy (dirname, entry, copysw)
Parameters: (dirname, entry) char(*), copysw bit(1)
Called by: user
Calls: findbranch, sim2$dirmod

set$rd (dirname, entry, retdate)
Parameters: (dirname, entry) char(*), retdate bit(72)
Called by: user
Calls: findbranch, effmode, sim2$dirmod
**set$constsw** (dirname, entry, constsw, dtbm)

Parameters: (dirname, entry) char(*), constsw bit(2), dtbm bit(72)

Called by: user, backup processes

Calls: findbranch, sim2$dirmod

**set_base_dir**(basename)

Parameters: basename char(*)

Called by: privileged user

Calls: none

**finddir**(dirname, segptr)

Parameters: dirname char(*), segptr ptr

Called by: sim1$getdirseg

Calls: findbranch, effmode, makeknown

**refindb**(dp, slot, uid, dtbm, mode, plist)

Parameters: dp ptr, slot fixed bin(17), uid bit(70), dtbm bit(52), mode bit(5), (1 plist, 2 size fixed bin(17), 2 array(plistmax) fixed bin(18))

Called by: segfault

Calls: effmode

**setlimits**(dirname, entry, hlim, llim)

Parameters: (dirname, entry) char(*), (hlim, llim) fixed bin(17)

Called by: administrative-ring procedures

Calls: findbranch, sim2$dirmod
setstrap (dirname, entry, flag, trapsw, trap)
Parameters: (dirname, entry) char(*), flag bit(2), trapsw bit(1), trap char(*)
Called by: administrative-ring procedures
Calls: findbranch, sim2$dirmod, sim2$branchmod

setretrievel (dirname, entry, flag, trapsw, trap)
Parameters: (dirname, entry) char(*), flag bit(2), trapsw bit(1), trap char(*)
Called by: backup and multilevel processes
Calls: findbranch, sim2$dirmod

setdd (dirname, entry, dtd)
Parameters: (dirname, entry) char(*), dtd bit(72)
Called by: backup and multilevel processes
Calls: findentry

ggetentry (dirname, entry, count, cdate, getsw, space, entryp)
Parameters: (dirname, entry) char(*), count fixed bin(17), cdate bit(72), getsw bit(1), space area(*), entryp ptr
Called by: backup and multilevel processes
Calls: findentry, sim1$getdirseg, makeunknown

putentry (dirname, entry, count, cdate, putsw, entryp)
Parameters: (dirname, entry) char(*), count fixed bin(17), cdate bit(72), putsw bit(1) entryp ptr
Called by: backup and multilevel processes
Calls: findentry, hash$in, hash$out, sim1$getdirseg, makeunknown
setusage (dirname, entry, olduse, newuse, blocksw)

Parameters: (dirname, entry) char(*), (olduse newuse) bit(2), blocksw bit(1)

Called by: administrative-ring procedures

Calls: effmode

activinfo$rdbranch (dp, slot, uid, actsw, itemsptr)

Parameters: dp ptr, slot fixed bin(17), uid bit(70), actsw bit(1), itemsptr ptr

Called by: getastentry

Calls: none

activinfo$wrbranch (dp, slot, uid, actsw, itemsptr)

Parameters: dp ptr, slot fixed bin(17), uid bit(70), actsw bit(1), itemsptr ptr

Called by: sim2$updateb, getastentry$delastentry

Calls: none

2. The Directory Maintainer

packer$packb (j, branchp, areap, i, entryp)

Parameters: (j, i) fixed bin(17), (branchp, areap, entryp) ptr

Called by: list_dir, status

Calls: appmode

packer$packl (j, linkp, areap, i, entryp)

Parameters: (j, i) fixed bin(17), (linkp, areap, entryp) ptr

Called by: list_dir, status

Calls: none
findentry(dirname, entry, slot, mode, ep)

Parameters: (dirname, entry) char(*), slot fixed bin(17), mode bit(5), ep ptr

Called by: status, chname, appendb, appendl, setdtd, getentry, putentry, findbranch

Calls: sim1$getdirseg, makeunknown

findbranch(dirname, entry, slot, mode, ep)

Parameters: (dirname, entry) char(*), slot fixed bin(17), mode bit(5), ep ptr

Called by: readacl, writeacl, set$bc, setml, set$copy, set$rd, movefile, estblseg, finddir, setlimits, setsystrap, setretrieve, setusage

Calls: findentry

hash$in(dp, entry, slot)

Parameters: dp ptr, entry char(*), slot fixed bin(17)

Called by: chname, appendb, appendl, putentry

Calls: hash$search

hash$out(dp, entry)

Parameters: dp ptr, entry char(*)

Called by: chname, putentry, removeb, removeb

Calls: hash$search

hash$search(dp, entry, found, hloc, slot, ep)

Parameters: dp ptr, entry char(*), found bit(1), (hloc, slot) fixed bin(17), ep ptr

Called by: hash$in, hash$out

Calls: none
rehash (hp, size)
Parameters: hp ptr, size fixed bin(17)
Called by: hash$in, hash$out
Calls: none

remval (ep, slot)
Parameters: ep ptr, slot fixed bin(17)
Called by: delentry
Calls: hash$out, sim2$dirmod

remvec (ep, slot)
Parameters: ep ptr, slot fixed bin(17)
Called by: delentry
Calls: hash$out, sim2$dirmod
Access Control

The calls to access control are outlined below. These calls are specified in detail in section BG.9 of this manual.

1. Access Control (The Hard-Core Section)

appmod (ep, dp, plistptr, entrylistptr, amode)
Parameters: (ep dp plistptr entrylistptr) ptr, amode bit(5)
Called by: list_dir, status
Calls: none

effmode (ep, dp, slot, opname, vacant, plistptr, emode)
Parameters: (ep dp ptr, slot fixed bin(17), opname char(*), vacant label, plistptr ptr, emode bit(5)
Called by: delentry, set$bc, set$rd, estblseg, fiddir, refindb, movefile
Calls: traphan

2. Trap Control (The Administrative Section)

traphan (trapargs, varargs, emode, ringno)
Parameters: (trapargs, varargs) char(*), emode bit(5), ringno fixed bin(17)
Called by: effmode
Calls: specified trap procedure
File System Device Interface Modules

The calls to the various procedures of the file system device interface modules are outlined below. These calls are specified in detail in section BG.10 of this manual. The Procedure names marked with an asterisk "*" indicate procedures which are called indirectly by means of the "push/pop" mechanism of DIM control (see section BG.10.3).

1. The Device Strategy Section

**new_io**(astptr, aftptr, op, stateword, relrec, reccount, memloc)

Parameters: (astptr, aftptr) ptr, op bit(3), stateword bit(36), (relrec, reccount, memloc) bit(18)

Called by: pagefault, pcreadseg, pctruncate, removepage, pcfreecore, cleanup

Calls: dim$control, device_control

**dim_driver**(dimid)

Parameters: dimid fixed bin(17)

Called by: DIM driving daemons

Calls: dim_control, device_control

**dim$control**(dimid)

Parameters: dimid fixed bin(17)

Called by: new_io, dim_driver

Calls: iodone, accounting routines, issues all indirect module calls

**dim$push**(iogx, modx, rtnx)

Parameters: iogx bit(17), modx fixed bin(17), rtnx bit(5)

Called by: request_initiator, dim_read, dim_write, dim_truncate, dim_cleanup, isolate_history_entry, fetch_map_sector, expunge_history_entries

Calls: none
**dim$pop**(ioqx,commsw)

Parameters: ioqx bit(17), commsw bit(1)

Called by: request_initiator, dim_read, dim_write, dim_truncate, dim_cleanup, isolate_history_entry, fetch_map_sector, expunge_history_entries

Calls: none

**dim$wait**(ioqx,setready,rtnx)

Parameters: ioqx bit(17), setready bit(1), rtnx bit(5)

Called by: isolate_history_entry, fetch_map_sector, expunge_history_entries

Calls: none

**request_initiator**(dimid,ioqx,ap,rtnx) (*)

Parameters: dimid fixed bin(17), ioqx bit(17), ap ptr, rtnx bit(5)

Called by: dim$control

Calls: dim$push, dim$pop, dim_read(*), dim_write(*), dim_truncate(*), dim_cleanup(*)

**dim_read**(dimid,ioqx,ap,rtnx) (*)

Parameters: dimid fixed bin(17), ioqx bit(17), ap ptr, rtnx bit(5)

Called by: dim$control, request_initiator(*)

Calls: dim$push, dim$pop, fetch_map_sector(*), put_command(*), release_history_entry

**dim_write**(dimid,ioqx,ap,rtnx) (*)

Parameters: dimid fixed bin(17), ioqx bit(17), ap ptr, rtnx bit(5)

Called by: dim$control, request_initiator(*)

Calls: dim$push, dim$pop, dim$wait, allocate_hypersector(*), isolate_history_entry(*), fetch_map_sector(*), put_command(*), release_history_entry
**dim_truncate(dimid,iogx,ap,rtnx)** (*)

Parameters: dimid fixed bin(17), iogx bit(17), ap ptr, rtnx bit(5)

Called by: dim$control, request_initiator(*)

Calls: dim$push, dim$pop, dim$wait, delete_and_fetch(*), put_command(*), deallocate_hypersector(*), release_history_entry, delete_history_entry

**dim_cleanup(dimid,iogx,ap,rtnx)** (*)

Parameters: dimid fixed bin(17), iogx bit(17), ap ptr, rtnx bit(5)

Called by: dim$control, request_initiator(*)

Calls: dim$push, dim$pop, expunge_history_entries(*)

**isolate_history_entry(dimid,iogx,ap,rtnx)** (*)

Parameters: dimid fixed bin(17), iogx bit(17), ap ptr, rtnx bit(5)

Called by: dim$control, dim_write(*)

Calls: dim$push, dim$pop, dim$wait, link_command

**fetch_map_sector(dimid,iogx,ap,rtnx)** (*)

Parameters: dimid fixed bin(17), iogx bit(17), ap ptr, rtnx bit(5)

Called by: dim$control, dim_read(*), dim_write(*), dim_truncate(*)

Calls: dim$push, dim$pop, dim$wait, link_command

**delete_and_fetch(dimid,iogx,ap,rtnx)** (*)

Parameters: dimid fixed bin(17), iogx bit(17), ap ptr, rtnx bit(5)

Called by: dim$control, dim_truncate(*)

Calls: none
expunge_history_entries (dimid, iogx, ap, rtnx) (*)
Parameters: dimid fixed bin(17), iogx bit(17), ap ptr, rtnx bit(5)
Called by: dim$control, dim_cleanup(*)
Calls: dim$push, dim$pop, dim$wait, link_command

release_history_entry (dimid, ap, histx)
Parameters: dimid fixed bin(17), ap ptr, histx bit(17)
Called by: dim_read, dim_write, dim_truncate
Calls: none

delete_history_entry (dimid, ap, histx)
Parameters: dimid fixed bin(17), ap ptr, histx bit(17)
Called by: dim_truncate
Calls: none

2. The Device Control Section

device_control (dimid, recall)
Parameters: dimid fixed bin(17), recall bit(1)
Called by: new_io, dim_driver
Calls: none

allocate_hypersector (dimid, iogx, ap, rtnx) (*)
Parameters: dimid fixed bin(17), iogx bit(17), ap ptr, rtnx bit(5)
Called by: dim$control, dim_write(*)
Calls: link_command
deallocate_hypersector(dimid, iogx, ap, rtnx) (*)
Parameters: dimid fixed bin(17), iogx bit(17), ap ptr, rtnx bit(5)
Called by: dim$control, dim_truncate(*)
Calls: link_command

put_command(dimid, iogx, ap, rtnx) (*)
Parameters: dimid fixed bin(17), iogx bit(17), ap ptr, rtnx bit(5)
Called by: dim$control, dim_read(*), dim_write(*),
dim_truncate(*)
Calls: none

link_command(dimid, comx, topsw)
Parameters: dimid fixed bin(17), comx bit(17), topsw bit(1)
Called by: isolate_history_entry, fetch_map_sector,
expunge_history_entries, allocate_hypersector,
deallocate_hypersector
Calls: none