

**NAME**

dd\_freup - release access to data distributor control file

**SYNOPSIS**

```
dd_freup(oid)
int oid;
```

```
dd_wakup(ptr)
char *ptr;
```

**DESCRIPTION**

dd\_freup

This subroutine is intended to be used by programs which access the .ddcntl files. This subroutine, when called, will check a file (protect file) which contains an access slot for each office. Each access slot is either empty, in which case no process is currently accessing the .ddcntl file for that office, or the access slot contains the process ID of the process currently having access to the ddcntl file for that office.

The argument oid is the office ID of the office to be accessed.

This subroutine, when called, will check to see if the process ID in the access slot of the office of interest matches the process ID of the calling process. If it does, it will remove the process ID from the access slot and zero the access slot, indicating that the calling process no longer has access to that particular .ddcntl file. It will then post a semaphore, causing any process which has been waiting for access to be awakened. This subroutine should thus be called only after a program has finished with the .ddcntl file for an office.

Subroutine dd\_freup returns one of the following values to the calling program:

- 5 Bad Protect file.
- 4 Protection file semaphore permanently locked.
- 3 System error occurred.
- 2 Illegal office ID passed to this subroutine.
- 1 This process ID was not in the office slot for the desired office; access slot not zeroed.
- 0 Free up worked O.K.

The Protect file is a temporary file and is rebuilt, by the first call to the subroutine dd\_wnfree or subroutine dd\_ifree, each time the system is rebooted.

dd wakeup

This routine will execute the program which sends a signal to a process. The program to be executed runs as super-user and can therefore send a signal to any process.

The argument ptr is a pointer to a string which is the process ID of the process to be awakened.

The return value indicates the success of the signal attempt:

- 0 -- process successfully signalled
- 1 -- process to be awakened did not exist
- 2 -- another type of error occurred

**LIBRARY**

/lib/lib1.a

**SEE ALSO**

dd\_ifree(3L) dd\_wnfree(3L)