

[Next](#) [Up](#) [Previous](#) [Contents](#) [FITSIO Home](#)

**Next:** [4.3 HDU-level Routines](#) **Up:** [4 CFITSIO Routines](#) **Previous:** [4.1 Error Reporting](#) [Contents](#)

## 4.2 File Open/Close Routines

```
int fits_open_file( fitsfile **fptr, char *filename, int mode,
int *status)
int fits_open_data( fitsfile **fptr, char *filename, int mode,
int *status)
int fits_open_table(fitsfile **fptr, char *filename, int mode,
int *status)
int fits_open_image(fitsfile **fptr, char *filename, int
mode, int *status)

int fits_create_file(fitsfile **fptr, char *filename, int
*status)
int fits_close_file(fitsfile *fptr, int *status)
```

**These routines open or close a file. The first `fitsfile` parameter in these and nearly every other CFITSIO routine is a pointer to a structure that CFITSIO uses to store relevant parameters about each opened file. You should never directly read or write any information in this structure. Memory for this structure is allocated automatically when the file is opened or created, and is freed when the file is closed.**

**The `mode` parameter in the `fits_open_xxxx` set of routines can be set to either `READONLY` or `READWRITE` to select the type of file access that will be allowed.**

(+)

These symbolic constants are defined in **fitsio.h**.

The **fits\_open\_file** routine opens the file and positions the internal file pointer to the beginning of the file, or to the specified extension if an extension name or number is appended to the file name (see the later section on ``CFITSIO File Names and Filters'' for a description of the syntax). **fits\_open\_data** behaves similarly except that it will move to the first HDU containing significant data if a HDU name or number to open is not explicitly specified as part of the filename. It will move to the first IMAGE HDU with NAXIS greater than 0, or the first table that does not contain the strings `GTI' (a Good Time Interval extension) or `OBSTABLE' in the EXTNAME keyword value. The **fits\_open\_table** and **fits\_open\_image** routines are similar except that they will move to the first significant table HDU or image HDU, respectively if a HDU name or number is not specified as part of the input file name.

When opening an existing file, the **filename** can include optional arguments, enclosed in square brackets that specify filtering operations that should be applied to the input file. For example,

```
myfile.fit[EVENTS][counts > 0]
```

opens the table in the EVENTS extension and creates a virtual table by selecting only those rows where the COUNTS column value is greater than 0. See section 5 for more examples of these powerful filtering capabilities.

In **fits\_create\_file**, the **filename** is simply the

(+)

root name of the file to be created. You can overwrite an existing file by prefixing the name with a `!` character (on the Unix command line this must be prefixed with a backslash, as in ``\!file.fit'`). If the file name ends with `.gz` the file will be compressed using the gzip algorithm. If the filename is `stdout` or `"-"` (a single dash character) then the output file will be piped to the stdout stream. You can chain several tasks together by writing the output from the first task to `stdout` and then reading the input file in the 2nd task from `stdin` or `"-"`.

---

Next	Up	Previous	Contents
------	----	----------	----------

[FITSIO Home](#)

**Next:** [4.3 HDU-level Routines](#) **Up:** [4 CFITSIO Routines](#) **Previous:** [4.1 Error Reporting](#) **[Contents](#)**

(+)