

```
#include <stdint.h>
```

```
typedef struct fat_BS {
```

```
    unsigned char    bootjmp[3];
    unsigned char    oem_name[8];
    unsigned short   bytes_per_sector;
    unsigned char    sectors_per_cluster;
    unsigned short   reserved_sector_count;
    unsigned char    table_count;
    unsigned short   root_entry_count;
    unsigned short   total_sectors_16;
    unsigned char    media_type;
    unsigned short   table_size_16;
    unsigned short   sectors_per_track;
    unsigned short   head_side_count;
    unsigned int     hidden_sector_count;
    unsigned int     total_sectors_32;
```

```
    //this will be cast to it's specific type once the driver actually knows what
    type of FAT this is.
```

```
    unsigned char    extended_section[54];
```

```
} __attribute__((packed)) fat_BS_t;
```

```
typedef struct dir_ent {
```

```
    uint8_t dir_name[11];           // short name
    uint8_t dir_attr;               // File sttribute
    uint8_t dir_NTRes;              // Set value to 0, never chnage this
    uint8_t dir_crtTimeTenth;       // millisecond timestamp for file creation time
    uint16_t dir_crtTime;           // time file was created
    uint16_t dir_crtDate;           // date file was created
    uint16_t dir_lstAccDate;        // last access date
    uint16_t dir_fstClusHI;         // high word fo this entry's first cluster
    number
    uint16_t dir_wrtTime;           // time of last write
    uint16_t dir_wrtDate;           // dat eof last write
    uint16_t dir_fstClusLO;        // low word of this entry's first cluster number
    uint32_t dir_fileSize;         // 32-bit DWORD hoding this file's size in bytes
```

```
} __attribute__((packed)) dirEnt;
```

```
int OS_cd(const char *path);
```

```
int OS_open(const char *path);
```

```
int OS_close(int fd);
```

```
int OS_read(int fildes, void *buf, int nbyte, int offset);
```

```
dirEnt *OS_readDir(const char *dirname);
```

```
int OS_mkdir(const char *path);
```

```
int OS_rmdir(const char *path);
```

```
int OS_rm(const char *path);
```

```
int OS_creat(const char *path);
```

```
int OS_write(int fildes, const void *buf, int nbytes, int offset);
```