```c
#ifndef _LINUX_MINIX_FS_H
#define _LINUX_MINIX_FS_H

#include <linux/types.h>
#include <linux/magic.h>

/*
 * The minix filesystem constants/structures
 */

/*
 * Thanks to Kees J Bot for sending me the definitions of the new
 * minix filesystem (aka V2) with bigger inodes and 32-bit block
 * pointers.
 */

#define MINIX_ROOT_INO 1

/* Not the same as the bogus LINK_MAX in <linux/limits.h>. Oh well. */
#define MINIX_LINK_MAX 250
#define MINIX2_LINK_MAX 65530

#define MINIX_I_MAP_SLOTS       8
#define MINIX_Z_MAP_SLOTS       64
#define MINIX_VALID_FS          0x0001          /* Clean fs. */
#define MINIX_ERROR_FS          0x0002          /* fs has errors. */

#define MINIX_INODES_PER_BLOCK ((BLOCK_SIZE)/(sizeof (struct minix_inode)))

/*
 * This is the original minix inode layout on disk.
 * Note the 8-bit gid and atime and ctime.
 */
struct minix_inode {
        __u16 i_mode;
        __u16 i_uid;
        __u32 i_size;
        __u32 i_time;
        __u8 i_gid;
        __u8 i_nlinks;
        __u16 i_zone[9];
};

/*
 * The new minix inode has all the time entries, as well as
 * long block numbers and a third indirect block (7+1+1+1
 * instead of 7+1+1). Also, some previously 8-bit values are
 * now 16-bit. The inode is now 64 bytes instead of 32.
 */
struct minix2_inode {
        __u16 i_mode;
        __u16 i_nlinks;
        __u16 i_uid;
        __u16 i_gid;
        __u32 i_size;
        __u32 i_atime;
        __u32 i_mtime;
        __u32 i_ctime;
        __u32 i_zone[10];
};

/*
 * minix super-block data on disk
 */
```

```
 */
struct minix_super_block {
        __u16 s_ninodes;
        __u16 s_nzones;
        __u16 s_imap_blocks;
        __u16 s_zmap_blocks;
        __u16 s_firstdatazone;
        __u16 s_log_zone_size;
        __u32 s_max_size;
        __u16 s_magic;
        __u16 s_state;
        __u32 s_zones;
};

/*
 * V3 minix super-block data on disk
 */
struct minix3_super_block {
        __u32 s_ninodes;
        __u16 s_pad0;
        __u16 s_imap_blocks;
        __u16 s_zmap_blocks;
        __u16 s_firstdatazone;
        __u16 s_log_zone_size;
        __u16 s_pad1;
        __u32 s_max_size;
        __u32 s_zones;
        __u16 s_magic;
        __u16 s_pad2;
        __u16 s_blocksize;
        __u8 s_disk_version;
};

struct minix_dir_entry {
        __u16 inode;
        char name[0];
};

struct minix3_dir_entry {
        __u32 inode;
        char name[0];
};
#endif
```