



smxCD™ User's Guide

August 27, 2013

by David Moore

Table of Contents

1. Introduction	1
2. Configuration.....	1
3. API.....	1
3.1 Disk Open/Close.....	1
3.2 File Open/Close	2
3.3 File Seek and Read	3
3.4 Find First/Next File Matching Path	4
3.5 Default Drive and Current Working Directory.....	5
4. Limitations	7

© Copyright 2000-2013

Micro Digital Associates, Inc.
2900 Bristol Street, #G204
Costa Mesa, CA 92626
(714) 437-7333
support@smxrtos.com
www.smxrtos.com

All rights reserved.

smxCD is a Trademark of Micro Digital Inc.
smx is a Registered Trademark of Micro Digital Inc.

1. Introduction

smxCD is a CD file system that reads ISO-9660 or High Sierra CD-ROMs from a USB CD-ROM drive. It is a read-only file system for CD-ROM, CD-R, and CD-RW media. CD-RW media can be used if written in the ISO-9660 format, using a utility such as Adaptec's Easy CD Creator (www.adaptec.com). Also note that it is capable of reading only the first session of a multi-session CD. These are CDs that have been written to, "closed" to ISO-9660, then added to and closed again. You will not be able to access files added in later sessions. If you are interested in support for reading CDs in other formats, or writing to CDs, please discuss this with us.

The API calls are listed below, with related functions grouped together. Note that the functions that take *userid* parameters are not listed — see #1 in the Limitations section, below.

2. Configuration

Configuration settings are in `cdfs.h`.

3. API

3.1 Disk Open/Close

These mount and unmount the CD drive, initializing data structures required for subsequent accesses.

BOOLEAN **cd_dskopen**(char *path)

Summary Open a CD-ROM drive.

Details Open or mount a CD-ROM drive for default user. Before accessing any files on that drive, you must open it.

Pars path Path of the drive, such as "A:"

Returns TRUE Success
FALSE Drive close failed. CD-ROM drive may not be present or the CD is not ready to be read.

See Also `cd_dskclose`

void **cd_dskclose**(char *path)

Summary Close a CD-ROM drive.

Details Close or un-mount a CD-ROM drive for the default user, when done accessing files on it.

Pars path Path of the drive, such as “A:”

Returns none

See Also **cd_dskopen**

void **cd_dskabort**(char *path)

Summary Close the CD-ROM drive for all users.

Details Close the CD-ROM drive indicated for all users.

Pars path Path of the drive, such as “A:”

Returns none

See Also **cd_dskclose**

void **cd_dskaborta**(void)

Summary Close all CD-ROM drives for all users.

Details Close all CD-ROM drives for all users and free all resources allocated, including block buffer space.

Pars none

Returns none

See Also **cd_dskclose**

3.2 File Open/Close

CDFD **cd_open**(char *path)

Summary Open a file.

Details Attempt to open and initialize a file control data structure and return a file index into the global drive array. Two open routines are included, one for the default user (0) and one for a specified user. If the routine fails a negative index is returned. The path specifier passed in as an argument defines the file to be opened. If a drive is not specified, the default drive for the user will be used. If the path does not include a root specifier, the path is considered relative to the default directory for the drive.

If open failed `cd_errno` will contain one of the following:

<code>CDE_EMFILE</code>	if file block init unsuccessful
<code>CDE_DRIVENO</code>	if drive number invalid or not mounted
<code>CDE_ENOENT</code>	if path entry not found
<code>CDE_INVAL</code>	if invalid path format

Pars path The path of the file, such as “A:\autorun.inf”.

Returns ≥ 0 Index of file control structure in global file array, if open successful.
-1 Open failed.

See Also `cd_close`

void `cd_close`(CDFD fileno)

Summary Close a file.

Details Close the file indicated. Deallocate memory and data structures allocated when opened and remove file from list of open files for the file’s owner drive data structure.

Pars fileno Index returned by `cd_open`()

Returns none

See Also `cd_open`

3.3 File Seek and Read

LONG `cd_lseek`(CDFD fileno, LONG offset, int origin)

Summary Move the file pointer.

Details Move the file pointer 'offset' bytes from the origin specified in argument origin. If the seek attempts to seek past the end of the file, the file length is returned and the pointer is set to 1 byte beyond the end of the file. If the seek attempts to seek before the beginning of the file, 0 is returned and the pointer is set to the beginning of the file. File position pointers are zero based.

Pars fileno Index returned by `cd_open`()

offset Number of bytes to seek from origin.
origin Three origins are defined as follows:
 PSEEK_SET - offset from beginning of file
 PSEEK_CUR - offset from current file position
 PSEEK_END - offset from end of file(backwards)

Returns The new file position after the seek operation

See Also **cd_open**

int **cd_read**(CDFD fileno, u8 *buf, u16 count)

Summary Read a specified number of bytes from a file into the specified buffer.

Details Attempt to read a specified number of bytes from a file into the specified buffer. If the read is successful, the number of bytes read is returned. If the end of file is encountered before the request is complete, the number of bytes returned will be less than the request. If an error occurs, the returned value will be less than 0 and will contain an error code. The file position pointer is updated.

Pars fileno Index returned by cd_open()
 buf Read buffer
 count Length of read request

Returns >=0 Length of read if successful
 -1 Error

See Also **cd_open**

3.4 Find First/Next File Matching Path

BOOLEAN **cd_gfirst**(CD_DSTAT *stat, char *path)

Summary Retrieve the first directory entry that matches a pattern.

Details Attempt to retrieve the first directory entry that matches a pattern. The path/pattern is parsed to the lowest directory level. A file is opened for the directory using one of the global file slots. The directory pointed to by this file is then searched for the match pattern. The file pointer is updated for subsequent calls. Since the drive must be open by a user before any file open operations are performed, a user specific version of this routine is provided. If the default user (0) is assumed but the default user (0) has not opened the drive, the match will return false.

Pars stat Directory statistic entry returned if match found
 path Path/match specifier

Returns TRUE if match found, file index of directory file in arg “stat.fileno” and generic format directory entry in arg “stat.dir”
FALSE if match not found or some error encountered

See Also `cd_gnext`

BOOLEAN `cd_gnext`(CD_DSTAT *stat, char *path)

Summary Retrieve the next directory entry that matches a pattern.

Details Attempt to retrieve the next directory entry that matches a pattern. File pointer is updated for subsequent calls.

Pars stat Directory statistic entry returned if match found
path Path/match specifier

Returns TRUE If match found, file index of directory file in arg “stat.fileno” and generic format directory entry in arg “stat.dir”
FALSE If match not found or some error encountered

See Also `cd_gfirst`

void `cd_gdone`(CD_DSTAT *stat)

Summary Finish a directory retrieval operation.

Details Finish a directory retrieval operation by closing the file allocated for the operation.

Pars stat Directory statistic entry returned if match found

Returns none

See Also `cd_gfirst`

3.5 Default Drive and Current Working Directory

(See Limitation 1)

int `cd_getdefltdrvno`(void)

Summary Get default drive number for default (0) user.

Details Get default drive number for default (0) user.

Pars none

Returns Default drive number for default (0) user

See Also `cd_setdefltdrvno`

BOOLEAN `cd_setdefltdrvno`(int driveno)

Summary Set default drive number for default (0) user.

Details Set default drive number for default (0) user

Pars driveno New default drive number

Returns TRUE Success
FALSE Fail

See Also `cd_getdefltdrvno`

BOOLEAN `cd_set_default_drive`(char *path)

Summary Set default drive for default (0) user.

Details Set default drive for default (0) user

Pars path New default drive path, such as "A:"

Returns TRUE Success
FALSE Fail

See Also `cd_setdefltdrvno`

BOOLEAN `cd_gcwd`(char *path)

Summary Get current working directory for default user.

Details Get current working directory for default user.

Pars path Current working directory

Returns TRUE Success
FALSE Fail

See Also `cd_scwd`

BOOLEAN **cd_scwd**(char *path)

Summary Set current working directory for default user.

Details Set current working directory for default user.

Pars path New working directory, such as “A:**”

Returns TRUE Success
FALSE Fail

See Also **cd_gcwd**

4. Limitations

1. Default drive and path: Currently, there can only be one default drive and path used by all tasks. The functions in *cdapi.c* that take *userid* parameter are intended to handle this, but some work is necessary to finish it. It is best, then, to explicitly specify the full drive and path for each operation.
2. *smxCD* is a read-only file system.
3. *smxCD* can access only the first session of a multi-session recordable CD (CD-R).
4. Reading from CD-RW discs is not supported, since these are created using proprietary software that does not write the data in ISO-9660 or High Sierra format.