Freescale MQX 0052TOS Example Guide

Mfs_cfcard example

The document explains the mfs_cfcard example, what to expect from the example and a brief introduction to the API.

The example

The application example code is used to demonstrate how to communicate with MQX File System (MFS) on external CF (Compact Flash) card.

The example code opens CF card device and installs MFS. It allows user to perform some basic operation with the CF card through the terminal interface for example write/copy/create/rename. It shows how to work with the driver and how to use shell commands.

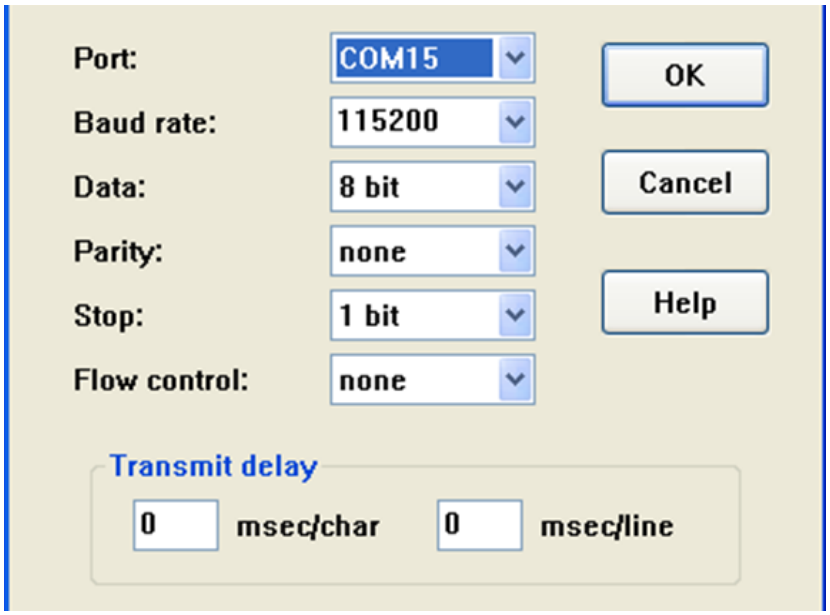
Running example

Check that the SHELLCFG_USES_MFS macro is set to 1 in the <MQX installation folder>/config/<board>/user_config.h. Then rebuild the BSP, PSP, MFS and SHELL projects for the target platform/IDE.

Hardware requirement

MCU board, PRIMARY, TWR-MEM and CF card are needed for running the example.

Start a terminal application on your PC and set the serial connection for 115200 baud, 8 data bits, 1 stop bit, no parity and no flow control.



The Shell function takes an array of commands and a pointer to a file as parameters. The Shell_commands array specifies a list of commands and relates each command to a function.

When a command is entered into the Shell input the corresponding function is executed.

List of shell commands

- "cd": Change the current working directory.
- "copy": Copy a file to another file.
- "create": Create a file.
- "del": Delete a file.
- "disect": Reads a sector of memory.
- "dir": List all files contain in a folder.
- "df": Prints out disk free information for current file system.
- "format": Format folder.
- "help": List all the commands.
- "mkdir": The command creates one or more new directories.
- "pwd": The command is used to output the path of the current working directory.
- "read": Read file.
- "ren": Rename a file.
- "rmdir" Removes the directory entry specified by each directory argument, provided the directory is empty.
- "type": SHELL utility to Ping a host.

Explaining the example

The mfs_cdcard application implements two AUTO_START tasks with different priority: Shell_task and CFCard_task. The objective of the application is show to user an example of communication with MQX file system through terminal interface.

Because of higher priority CFCard_task will become running task and Shell_task is ready task.

CFcard_task installs deriver for CF card and checks the state of CF card.

- CF card driver consists of two parts:
  - Lower layer: Driver can be opened even if the CF card is not inserted.
  - Upper layer: Driver should be opened only when CF card is present in the slot.

- After The task opens PC card device, Task goes into a for loop and check the state of CF card.

  - Previous state is CARD_DETACHED. If card is inserted, task will wait 300 milliseconds for insertion of card is completed. If card is present in the slot, Task will call CF_filesystem_install() for installing file system and change state of card to INTERFACED. If not, State of card will be set to CARD_ERROR.

  - Previous state is INTERFACED. If card still present in the slot, State will nothing changes. If not, state of card will be change to CARD_DETACHED and call CF_filesystem_uninstall() for uninstalling file system.

  - Previous state is CARD_ERROR. If card is not inserted, State will be CARD_DETACHED.

CFcard_task uses CF_filesystem_install() for installing file system.

- Allocate memory
- Open CF card and set block mode for PC flash. If block mode cannot be enabled, Byte mode will be used.
- Try to install partition manager. If partition manager is selected, Task will validate and install MFS file over. If not, Task will install MFS file over CF card driver.
- If MFS is installed, Task will open and format file system if format is required. If MFS cannot be installed, Task will uninstall file system and return NULL.

Note: Partition manager

The partition manager device driver is designed to be installed under the MFS device driver. It lets MFS work independently of the multiple partitions on a disk. It also enforces mutually exclusive access to the disk, which means that two concurrent write operations from two different MFS devices cannot conflict. The partition manager device driver can remove partitions, as well as create new ones. The partition manager device driver is able to work with multiple primary partitions. Extended partitions are not supported.

CFcard_task use CF_filesystem_uninstall() for uninstalling file system.

- Close file system

- Uninstall MFS
- Close and uninstall partition manager.
- Close CF card
- Finally free memory.

At the beginning of for loop, the task waits 200 milliseconds. Because of scheduling, shell_task will become running task and waits for command from user. After 200 milliseconds are expired, Shell_task will stop running and becomes ready task. CFcard_task will run again and so on.