

VFSdos 2.02

DOS programming environment for PalmOS.

April 9, 2003

(Freeware!).

```
C:\>pause
Press any key to continue...
C:\>dir

Volume in drive C is CF_CARD
Directory of C:\

TRGPRO*1 <DIR>      08-22-00 05:26p
URANTIA  <DIR>      08-22-00 05:27p
PROJECTS <DIR>      08-22-00 05:27p
APR88A*1 <DIR>      08-22-00 05:27p
PALMSOC  <DIR>      08-22-00 05:27p
DATA8A*1 <DIR>      08-22-00 05:28p
E-BOOKS  <DIR>      08-22-00 05:28p
OLDSOU*1 <DIR>      08-22-00 05:28p
REFESE*1 <DIR>      08-22-00 05:28p
SYSTEM   <DIR>      08-22-00 05:28p
SCHEMATI <DIR>      09-15-00 09:39p
TEMP     <DIR>      09-21-00 09:26p
AUTOCEF  <DIR>      12-01-00 14:00d
AUTOEXEC BAT      4 12-31-00 07:57p
14 file(s)          4 bytes
13 dir(s)          350208 bytes free

C:\>
```

Download: [VFSDOS.ZIP](#)

What's new (April 9, 2003): High density support for PalmOS 5 (80x53 display), color support for text output.

What's new (Jan 29, 2003): Bug fixes in the StdioLib library. The VFSdos console was recompiled with the new StdioLib version (2.08) and it now displays the StdioLib version while booting. The VFSdos console source code is now included in the archive.

What's new (Sept 12, 2002): The VFSdos console has been recompiled and is now using the StdioLib library. There is nothing new to the shell itself - all changes are internal. however, there are many new stdio functions available, contributed by Philippe Guillot. Please look at the develop info provided with the VFSdos archive.

* There is also a new **Pascal Compiler for VFSdos**, also written by Philippe, you can check it out [here](#).

INTRODUCTION

VFSdos is a **DOS shell console, text-based environment** for Palm handheld computers equipped with a VFS expansion slot (such as the m505, Clie, Handera and TRGpro). **VFSdos** is very similar to the good old' **COMMAND.COM**. It's not only used for simple DOS file operations - **it's a DOS environment** where **VFSdos applications** and batch files can be executed from SD/MMC/MC/CF cards. It is targeted to DOS programmers, so they can continue to write and test plain DOS code on their Palm computers.

VFSdos should not be confused as a DOS emulator. It will not run the same DOS programs as your

desktop. It will only run *VFS DOS applications* specifically written for it. It's main use is to allow developers to create command-line / console applications such as command-line assemblers (I'm working on a PIC VFSdos assembler at this moment).

VFSdos is not designed to be used as a file management application. There are better solutions for this, such as **McFile** and **TealMover**. It's primary use is to provide a console text-based environment for the use and creation of command-line applications using the provided **STDIOLIB** shared library. It is also for those who miss the old DOS days of the PC (like me :-)

What do you need to run VFSdos: A palm equipped with a VFS expansion slot such as the m505, SONY Clie, or TRGpro (upgraded to OS 3.5.3) and an expansion memory card. **You should also have a good knowledge about DOS on your PC.** VFSdos is mostly targeted to old DOS programmers such as myself :-) Developers should have a look in the included **DEVELOP** subdirectory for information on how to write VFSdos applications.

Important: See end of this document for current limitations. VFSdos has currently more limitations than TRGdos, due to the limitations of the VFS file system in the PalmOS 4.0. For example, wild cards in arguments is not currently supported.

*** VFSdos was written and developed on a m505 using OnBoardC as a C-compiler, and QEd as an editor.**

VFSdos has the following features:

*** True DOS commands:** *CLS, DIR, REN, MD, RD, CD, TYPE, ECHO, PAUSE, VER, COPY, DEL, ...*

*** Minimal support for simple batch (.BAT) files, including AUTOEXEC.BAT.**

*** Support for VFSdos applications** (A demo utility is included in the ZIP file as a sample).

*** DOS application shared library** available to developers for developing DOS applications for VFSdos.

*** Launches VFSdos applications with command-line parameters from the command prompt.**

*** No need of unique PalmOS registration IDs:** All DOS applications use the same ID and operate under VFSdos.

*** VFSdos applications are of small size, since there's no graphical user interface overhead.**

*** DOS command-line parameter and batch-control power.**

*** Scrollable 40x26 or 80x53 (PalmOS 5 only) output console.**

*** Easy to port existing DOS (text) applications to VFSdos:** The VFSdos programmer's library is using standard **STDIO** functions: *printf(), gets(), getch(), open(), close(), read(), write(), malloc(), main(int argc, char *argv[]), etc...*

What is a VFSdos application:

It's a DOS application that is written to run under the **VFSdos environment**. It has no graphical user interface - it uses a **40x26 or 80x53 (OS5) scrollable screen** as output and has **command-line input** (flashing cursor, input via graffiti or external keyboard). VFSdos applications are easy to develop: they are **using standard C-library functions** such as *printf()*. It's easy for developers to write **VFSdos applications**, and source code can be ported to the PC with no modifications, recompiled and run on the PC.

VFSdos was primarily designed for VFS equipped handhelds. However, it works on other Palm handhelds as well, but since some Palms have no expansion slot, all file/directory DOS operations are not available. You can use VFSdos on such Palms to only execute VFSdos applications. Since PalmOS uses long filenames for its application databases, directory output on those handhelds differs from the standard output.

Here's the source code of a very simple VFSdos application. It simply displays the command line parameters that are passed to it from the VFSdos command prompt:

```
// -----  
  
//      VFSDOS SAMPLE APPLICATION  
  
//      Version 1.0  
  
//      by: Harry Konstas Jan 5,2002  
  
// -----  
  
#include "stdio.h"  
  
// -----  
  
//      M A I N  
  
// -----  
  
int main(int argc, char *argv[])  
{  
    int f;  
  
    printf("DOS sample application.\n\n");  
  
    printf("Number of parameters = %d\n\n",argc);
```

```

    for(f=0;f<=argc;f++)

        printf("parameter #d = %s\n",f,argv[f]);

    printf("\nPress any key to continue...");

    getch();

    return 0;

}

```

As you can see, the above sample code can be also compiled 'as-is' on your desktop computer using standard C compilers as TurboC. That's where VFSdos becomes usefull, you can easily port text-based DOS applications to VFSdos! The included developer's StdioLib library also provides R/W file and streaming operations as well.

VFSdos shell USAGE:

We won't go into the details on how to use the DOS commands since VFSdos is very similar to MSDOS. Assuming that you already have used DOS in the past, I will only list the current limitations/differences of VFSdos.

First, simply launch VFSdos like any other application from your application launcher screen. You will immediatly find the familiar **c:> prompt** on your screen. **To EXIT VFSdos** simply tap the **APPS** icon or press any of the four **HARDWARE** buttons. The first time you launch VFSdos, you will get the version information of VFSdos, and the available disk drives. You can always get this information by using the command **VER.:**

Available disk drives:

A: (virtual) This is your handheld's RAM, where all your Palm applications reside. You cannot switch (log) onto this drive - you can only use the command **dir a:** to view its contents. Between "press any key" prompts, if you press 'C' (use graffiti) it reacts as the familiar **CTRL-C** in DOS, which halts the current operation.

B: (virtual) Again your RAM, but this time only VFSdos applications will be displayed. You cannot switch (log) into this drive - use the **dir b:** command to view its contents. This drive is only used to group all the VFSdos applications. Between "press any key" prompts, if you press 'C' (use graffiti) it reacts as the familiar **CTRL-C** in DOS, which halts the current operation.

You can delete any installed VFSdos DOS application from the C: prompt as follows:

```
c:/>del b:filename
```

where *filename* is an installed VFSdos application.

C: (real) This is your expansion card drive. All DOS operations can be used with this drive, since it's the **only** drive that points to a true file-system (RAM has no file system - only databases). **This drive is not available on non-VFS equipped Palms.**

D: (virtual) Another RAM portion which groups together **ALL DOC** (document reader) files. You cannot switch (log) into this drive, you can only access it by using the **dir d:** command to view only all the DOC files installed into your memory.

* **All DOS file operations such as:** *del,ren,md,cd, etc...* **can only be used on the C: drive.**

The **REBOOT** command can be used to "reboot" VFSdos and re-launch any available autoexec.bat file.

VFSdos Limitations:

Wildcards (ex: dir *.pdb) are not currently supported by VFSdos, due to the lack of support by the VFS file system. The TRGdos had support for it because the TRGpro file system had support for wildcards.

Long filenames are not supported by VFSdos. Directories having long names will not be shown. Filenames with long names will be shown truncated, and will not be accessible for file operations. This is because the VFS file system does not support short filenames.

You can only log into the **C:** drive (Palms with no VFS support log only into drive B:). Virtual disk drives **A: B: and D:** are only accessible via the **DIR drive:** command. This is because drive C: has access to the only available file system of your PALM, the VFS expansion slot.

The **COPY** command is limited to copying one file at a time (no wildcard specs). Parameters are:

copy <source filename> <destination filename>

The **DIR** command has only the **/p** switch option (paged output). Between "press any key" prompts, if you press 'C' (use graffiti) it reacts as the familiar **CTRL-C** in DOS, which halts the current operation.

Batch file support has no stack, so launching an application within a batch file, the control returns to DOS. Available batch language commands are: *echo* and *pause*.

Currently there is no piping or input redirection support in VFSdos.

Output screen is erased by PalmOS when a VFSdos application is launched.

Currently available VFSdos functions on VFS equipped handhelds:

?, reboot,help, exit, ver, cls, dir [/p], cd, md/mkdir, rd/rmdir, del/delete, ren/rename, type,echo and pause.

Currently available VFSdos functions on Palms with no VFS support:

?, reboot,help, exit, ver, cls, dir [/p], del/delete (drive b: only), echo and pause.

* The **REBOOT** command is used to reboot VFSdos.

Other important notes:

At the DOS prompt, pressing the **UP hardware button** has the same effect as the **F3** key of DOS: it repeats the last command that was entered at the DOS prompt.

VFSdos remembers and stays at the last directory path between launches.

While booting, VFSdos checks for a **MEMO** having **AUTOEXEC.BAT** as a title (1st line). If found, the batch file will be executed. If not, VFSdos launches **AUTOEXEC.BAT** file - if available at the root of drive C:. Please note that first line of the autoexec memo **MUST BE IN CAPITALS** like this: **AUTOEXEC.BAT** so VFSdos can recognize it and launch it during boot time.

VFSdos applications have all the same (already registered) unique ID: '**pdos**'. They are not visible under the Palm's native application launcher since they are of type '**pdos**'. They are only visible and executable from drive B: under VFSdos.

There's a VFSdos 'demo' file included with VFSdos. To launch any VFSdos application, simply type its name and command line parameters from the c:\> prompt.

You can delete any installed VFSdos DOS application from the C: prompt as follows:

```
c:/>del b:filename
```

where *filename* is an installed VFSdos application. **You must include the drive 'b:' followed by filename.**

VFSdos was written by [Harry Konstas](#) on a m505 handheld, using [OnBoardC](#).

The **StdioLib** shared library was ported from the first VFSdos version by [Philippe Guillot](#) using OnBoardC.

There are no warranties given with the use of VFSdos. User assumes all responsibilities - use at your own risk. VFSdos source code available on demand.

Acknowledgements:

I would like to thank [Philippe Guillot](#) for porting the VFSdos developer's library into the StdioLib shared library file, and for adding many new functions making the library more usefull and easier to use. Without his work, the library would have remained as a bunch of source code files. Thank you Philippe, your work is very much appreciated!

I would also like to thank [Laurent Thaler](#) for providing the VFSmgr.h header file.

Harry Konstas

www.total.net/~hkonstas

[Home page](#)