QuickCMD user guide

written by Arnaud Storq

July 17, 2011

http://www.norecess.net/quickcmd.html

Figure 1: Using QuickCMD

Contents

1 Introduction

QuickCMD is a new shell-like user interface for disc-based Amstrad CPC computers.

The initial goal of this software is to provide a faster way to browse disc content and launch programs. Usually, an Amstrad CPC user tends to use BASIC prompt to first display the disc content to retrieve a filename then manually launch a program. While this process is OK and functional, this can quickly become a boring task when there is a need to browse multiple discs.

Additionally to that, this program also features some interesting commands to manage disc content that are usually not accessible from BASIC prompt. This is only possible thanks to the low-level management of a disc content, that directly read and writes sectors instead of using functionalities provided by default with the firmware. Many formats are supported, including DATA, SYSTEM and PARADOS. Due to that, disc operations are faster than the ones accessible from BASIC.

Finally, QuickCMD is really respectful regarding the system. It has a very small memory footprint, and can be easily extended with already-existing RSXs (Resident System eXtension). It also features a char display routine that is faster than the standard one provided by the firmware, so commands generally performs faster than their equivalents under BASIC.

2 Installation

This program requires :

- **an Amstrad CPC 6128** : both classic and Plus models are supported. Amstrad CPC 464s with external disc drive are currently not supported.

- **a ROM box** : such devices are RAMCARD, ROMBOX, etc. Because QuickCMD is provided as a ROM, you need to be able to extend your CPC with custom ROMs. More information about those devices can be found here : `http://www.cpcwiki.eu/index.php/Peripherals#Memory_expansions_.2F_ROM_boxes`

    You can install this ROM at any place. Just make sure to not override the installed foreground ROM (BASIC ROM is default, at rom 0) or the ROM responsible of drive management (CPM ROM is default, at rom 7).

Once installed, QuickCMD will only use &6 bytes in memory for ROM persistent data. This is possible because QuickCMD is treated as a real ROM, meaning the code is executed from ROM, and not copied back to RAM at initialization time.

When launched, QuickCMD will dynamically allocate &328 bytes of data under HIMEM address, used as temporary buffer managed during run-time. As a note, it's interesting to mention it actually takes less memory to use QuickCMD than using the regular BASIC's CAT command (that internally use &800 bytes for its temporary catalog buffer under HIMEM address).

3 Commands

When user enters QuickCMD's user interface, a command-line prompt is visible to the user. People familiar with any DOS-based systems will recognize the workflow. User has to manually type

commands in order to send orders to be executed by the computer. Commands are not case-sensitive.

This section of this document describes all available commands.

3.1 A: or B:

By typing A: or B:, the user can select the corresponding drive (drive A or B). All forthcoming disc operations will be using the selected drive.

3.2 ABOUT

This gives the credits of the program. It also reports memory usage (address and length) required for QuickCMD's execution.

3.3 CAT

This command behaves exactly like the regular CAT command does under BASIC. Its aim is for compatibility purpose, other programs such as BonnyDOS rely on this to display hard-drive content. Hidden files from catalog are not displayed. This command is not recommended to browse a disc, prefer the DIR command instead.

3.4 CATART

This command behaves exactly like the regular CAT command does under BASIC, excepted it also reset the screen to default's startup one (blue background with yellow letters, using graphical MODE 1). This allows to display "cat arts", which represent graphical screens displayed with CAT command. Once displayed, user can press a key to return back to default QuickCMD's screen setup. This command is internally called by DIR command when a "cat art" is automatically detected.

3.5 CRTC

This command reports the CRTC type. CRTC is the graphical chip of the Amstrad CPC. The CRTC test is based on Madram's implementation. It may not be the best implementation out there, but does the work as expected. It's useful to know installed CRTC type when dealing with demos that are dedicated to a single CRTC type.

3.6 DEL

This command delete a file on disc. Be aware, no confirmation is asked to the user !

3.7 DIR

This command gives the list of files on disc. It shows files in the current user only. Available size in Kb is also reported. When this command requires multiple pages to display a long list of files, it will actually ask the user to press a key at each page-scroll. There is no filename sorting, files get displayed in order of apparition when parsing catalog information. The command supports Read-Only and Hidden files. It's interesting to mention that Hidden files are also treated and shown. If catalog appears to be a "cat art", then this command will be internally redirected to CATART command.

3.8 DIRALL

This command performs like the DIR command, excepted it lists files for ALL users (from user 0 to 15). This is interesting to find files that are not under the current user. User of file is displayed as a prefix to the filename.

3.9 DIRDEL

This command performs like the DIR command, excepted it lists deleted files. If a file has been recently deleted and that no write access on disc have been done, it's possible to retrieve original file content using the UNDEL command.

3.10 DISCINFO

This gives information about disc format. It reports the name of the format, its capacity in Kb, the number of maximum files that can be managed, the single- or double-sided info, as also sectors and tracks identifiers. Many formats are supported, including those managed by PARADOS.

3.11 HIDE

This command allows to hide a file from catalog. Of course, it will be still shown under QuickCMD (flagged as a Hidden file). When using CAT command, this file will be not reported (invisible) to the user. Files hidden can be shown back to catalog by using the SHOW command.

3.12 INFO

This gives information about specified file. It reports its type (BASIC, PROTECTED BASIC, BINARY and ASCII). If the type allows it, it also reports the Load Address, the length of the file and its Execution Address.

3.13 PUTROM

This command is only compatible with RAMCARD devices. First parameter is the ROM's filename, second parameter is the ROM number to write to. Specified file will be put onto the rom box. User will be asked to switch on/off the corresponding write DIP switch. It makes use of default implementation provided by Ram7 (the original author of the RAMCARD). It's advised to use ROMS command before using PUTROM to make sure to not overwrite a wanted and already installed ROM.

3.14 REN

This command is used to rename a file. The first parameter is the new filename, the second parameter is the current filename to rename. Be aware ! REN command under BASIC use a reverse convention (old filename before new one). Such a reversion is used for DOS-like compatibility.

3.15 RO

This command is used to set Read-Only attribute to a specified file. Files set as Read-Only can not be deleted till this attribute got removed.

3.16   RW

This command is used to set Read-Write attribute to a specified file (which is the default for a file). Files set as Read-Write can be deleted or overwritten.

3.17   ROMS

This lists the installed ROMs on the system. On an unexpanded CPC, ROM 0 (BASIC) and ROM 7 (CPM ROM) are listed by default. This is useful to execute this command before putting a ROM on a rom box, to know which rom number should be used. The command shows, for each installed ROM, its internal name, its version and its type (foreground, background or extension).

3.18   RSX

This command allows the execution of a RSX. First parameter is the RSX's name, other parameters are optional parameters to be sent to RSX. Optional parameters are always treated as string types. QuickCMD offers the functionality to execute a RSX as a command, so this command is used to override QuickCMD's internal commands. This is an example to make it clear : let's say the Arnor's UTOPIA ROM has a RSX called HELP. In QuickCMD, if user types HELP, the internal QuickCMD's HEP command will be executed. User typing RSX HELP will then bypass the QuickCMD's HELP command and execute the Arnor's UTOPIA's HELP RSX.

3.19   RUN

This command allows the execution of a file on disc. Supported file types are BASIC, PRO-TECTED BASIC or BINARY. It internally makes use of the BASIC's RUN command, so it's fully compliant with all existing programs. Like the RSX command, QuickCMD offers the functionality to type a filename as a command, so this command is used to override QuickCMD's internal commands. This is an example to make it clear : let's say we have a file on disc called HELP.BAS. In QuickCMD, if user types HELP, the internal QuickCMD's HEP command will be executed. User typing RUN HELP will then bypass the QuickCMD's HELP command and execute the file HELP.BAS located on disc.

3.20   SETUSER

This command is used to set user for a specified file. First parameter is filename, second parameter is the user.

3.21   SHOW

This command is used to show (unhide) a file from catalog. Shown files can be hidden back from catalog by using the HIDE command.

3.22   UNDEL

This command is used to undelete a file. This command will fail if a disc write access has been performed since the file has been deleted. Deleted files can be shown through the DIRDEL command.

3.23 USER

This command is used to change current user. Default user is USER 0. Note that current user is also the one used in BASIC, so if you leave QuickCMD back to BASIC, then current user will remain active.

4 Tips & Tricks

QuickCMD also offers some additional features that can make user's activity easier.

4.1 Shortcuts

In order to maximize user workflows, some shortcuts are provided to enhance productivity :

- **ENTER** : if you press the ENTER key while the input command line is empty, then the screen will be cleared.

- **TAB** : this is probably one of the most interesting feature from QuickCMD. When you start typing a filename, if you press TAB key, QuickCMD will parse the catalog and try to autocomplete filename.

- **TAB** : if TAB key is pressed while the input command line is empty, then DIR command will be auto-typed (and executed).

- **ESC** : if ESC key is pressed, then the input command line's content will be cleared.

- **ESC or CTRL+TAB** : if ESC key or CTRL+TAB keys are pressed, then QuickCMD will exit back to BASIC.

4.2 Expected workflows

The above shortcuts actually draws what should be one of the top uses of QuickCMD. It should be specifically interesting for new comers. If you find yourself a common workflow that is not listed here, feel free to report it to the author of this program !

- **the "fast DIR"** : when your Amstrad CPC got started, user just has to press CTRL+TAB to enter QuickCMD. Then, TAB key is expected to execute DIR command. Once the user chose a file to execute, the first characters composing the filename can be typed and then the filename can be auto-completed using the TAB key. Finally, press ENTER key to execute file. This workflow is ultra-fast, you will get quickly used to the CTRL+TAB then TAB scheme ! Way faster than typing CAT command under BASIC then using SHIFT + arrow keys to select a filename then pressing COPY key to copy/paste filename then pressing CTRL+left arrow then pressing CTRL+ENTER to RUN" the file.

- **BASIC is still your friend** : QuickCMD is not a replacement for BASIC. BASIC has important features that QuickCMD does not support, and vice versa. It has been made easy to the user to quickly switch from BASIC to QuickCMD (and from QuickCMD to BASIC) by using the CTRL+TAB keys. Since QuickCMD is a ROM, you can deal with large BASIC programs or data stored in BANKs without any kind of alteration.

- **Recommended third-party ROMs to be used with QuickCMD** : QuickCMD is very well integrated with Arnor's ROMs. If you need a sector editor, use UTOPIA's DEDIT command. Use UTOPIA's MEDIT command for a memory editor. Use PROTEXT to edit your

source files. Use MAXAM to program your assembly source-codes. By using all these power-ful tools conjointly, you will be get an efficient manner to be creative with your CPC. This is a completely safe workflow, there have been many tests during QuickCMD implementation to make sure memory was not overwritten regarding various usage of HIMEM.

- **Rediscover your library !** : One of the obvious strength of QuickCMD is the ability to show Hidden files on disc, as also quickly get information about a file or disc format. From a developer point of view, it's fun to be curious about how "proper" is a program, cf. is it composed by one or two files or is it composed by multiple small hidden files ? etc. Those things generally does not appear when using regular CAT command.

5   History

This is version 2.0 of the software. This version is a complete rewrite of version 1.0 . While the initial version was featuring the interesting concepts featured back in version 2.0, it also pointed out some negative points :

- **too big for what it does** : most of the program was written in C language. While being more productive that way, the program actually took too much memory.

- **a ROM that isn't a ROM** : the program was stored in a ROM. When executed, the content of the ROM was actually copied into RAM to make use of dynamic memory (it's impossible to write into a ROM). That was pretty bad, as the program was launched from BANK &C7 as also some parts of memory from fixed &8000 location.

- **poor implementation** : the program performed poorly. Most of disc operations were based on CAS CATALOG (the CAT command), that is slow and requires a 2Kb memory buffer.

- **custom charset** : the program made use of a custom integrated charset. Char output routine was efficient but took lots of memory space.

- **and more...** : many little details that were also polluting the code base.

Instead of maintaining a program with lots of lacks, decision has been took to completely rewrite the program with a different approach. Here are the most interesting points :

- **asm including C** : instead of writing a C program with some assembly code inside, it has been more efficient to write the whole application using assembly code and implementing some high-level parts in C programming language. One of the best example for such a usage is the sector parsing part, that use a C's struct to iterate through the file entries.

- **a ROM that is a ROM** : writing a real ROM forced to use dynamic memory management. While this technic was interesting to implement, the most interesting part to point out is that QuickCMD's memory management got actually optimized to the maximum. The list of dynamic variables has to be kept as small as possible.

- **and more...** : many little details that, when written a second time, get suddenly clearer and much more maintenable.

6 Credits

Program written by Arnaud Storq (www.norecess.net, 2011) With the help of Kevin Thacker and Richard Wilson, CRTC type detection by Madram, RAMCARD management based on implementation from Ram7.

Many thanks to every person that reported me issues with version 1. This helped a lot making a better implementation for version 2 !