

PERSISTOR[®] CF2

Managing CF2 Behavior

With PicoDOS[®]



Overview of PBM an PicoDOS	4
The CF2 Prompts	4
PBM details	5
PBM Operation	5
PBM Persistent Variables	5
PBM Timeout	5
PBM Flash Writes Protection	5
PBM BOOT manager	5
When does the CF2 go into PBM mode?	6
PicoDOS Details	6
PicoDOS and ToPico Projects	6
PicoDOS and Memory Cards	7
Power Cycled Memory Card Detect is Necessary	7
PicoDOS Commands Depending on Memory Card	7
PicoDOS and the Real Time Clock	8
The Backup Battery	8
The Virtual EEPROM	8
What is VEE and How It Works	8
User Variables	9
System Variables	9
Loading & Running CF2 Applications	9
.APP, .RUN and .PXE files	9
Loading .APP and .RUN files with MotoCross	9
Loading PicoDOS with MotoCross	10
Saving .RUN files as .PXE files on CompactFlash	10
Placing .RUN files directly on CompactFlash	11
Loading .APP Files with RESTORE2	11
Loading Application Files by Modem	11
BACKROM and RESTORE2	12
Batch and AUTOEXEC.BAT files	12
Loading and Off-Loading Data Files	14
Using a Memory Card Reader	14
Loading data files with MotoCross	14
Using XModem or YModem	14
Using USB offloads	14
PBM commands	15
PicoDOS commands	19
System Commands	19
Launch Commands	23
Drive (Memory Card) Commands	24
Directory Commands	25
File Commands	26
Memory Commands	28
Port Test Commands	29
Modem Commands	30
Appendix A - File Limits	32

Purpose of this Guide

This guide is primarily for beginning users of the Persistor CF2. We will describe the basic operating system, setting program variables and boot preferences, and summarize the PicoDOS commands.

Other Documents

CF2GSG.pdf is the CF2 Getting Started Guide. It is the first manual a new user should review.

CF2PRG.pdf is the CF2 Programmers Manual.

CF2API.pdf is the CF2 API guide. It describes the hardware specific library of functions and macros.

Overview of PBM and PicoDOS

Your CF2 is delivered with a DOS like operating system installed, called PicoDOS. It provides a set of utility commands which will be described ahead. The PicoDOS sign on message looks like this;

```
-----  
Persistor CF21M    SN 08358    PicoDOS V4.03r1    PBM V4.00  
(C) 1998-2007 Persistor Instruments Inc. - www.persistor.com  
-----  
  
C:\>
```

The CF2 also has a lower level monitor program called "Permanent Boot Monitor", commonly referred to by the abbreviation PBM. The PBM determines whether the CF2 will boot into your loaded application, or into PicoDOS, or into its own minimal monitor. The PBM also negotiates with Motcross to receive an application or file transfer and place it into RAM, or a program space in FLASH or on the CompactFlash card. If a program load is initiated while the CF2 is in PicoDOS, the PBM monitor handles the application load in the background. The PBM sign on message looks like this;

```
-----  
CF2 Permanent Boot Monitor  
www.persistor.com    V4.00  
-----P  
== FLASH WRITES ENABLED ==  
  
PBM>
```

The CF2 Prompts

There are three normal prompts presented by PicoDOS and PBM, and possibly other prompts.

CF2> this is the default prompt from PicoDOS when there is no memory card detected.

C:\> this is the default prompt from PicoDOS when a memory card has been detected. C:\ is the root directory of the memory card.

PBM> this is the PBM prompt.

ANY OTHER PROMPT> an installed application may also present a unique prompt, or a user can change the prompt (semi-permanently or until next reset) with the PROMPT command.

PBM details

PBM Operation

Users seldom need to interact directly with PBM, but may find themselves in PBM if something has gone wrong.

PBM resides in the bottom 16KB of flash. A tiny portion of PBM is the first bit of software to execute from a power-up or reset. It determines what program should be launched; PicoDOS or a user application (or resume a user application returning from suspend mode). If there is a problem, it boots its own monitor. It accepts standard Motorola S records and MotoCross binary load files which can be stored to RAM or flash memory, and configured to automatically run at startup.

PBM executes its principle reset vectoring function directly from flash. When required to do anything more challenging than this, it moves itself to RAM and runs from 32KB of RAM at the base of what is normally considered the PicoDOS private data area (0x4000 - 0x7FFF). PBM is entirely self-contained and can accomplish all of its maintenance tasks with no references to external driver software.

PBM Persistent Variables

PBM has access to non-volatile memory for variables which are saved through a power cycle. These include the BOOT setting.

PBM Timeout

PBM can be enabled with or without a one minute timeout. When the timeout is enabled, when the timeout counter reaches zero, the PBM will either execute a CF2 reset or drop into a low power mode. The PBM monitor displays the countdown when enabled.

PBM Flash Writes Protection

PBM can enable or disable flash writes. When flash writes are disabled, you will not be able to load an application, set system variables, or change the boot selection. The PBM sign on message will tell you whether flash writes are enabled or disabled and you can re-enable flash writes using the WREN command, described later. The default setting for flash writes on the CF2 is enabled, but some PBM activities can disable flash writes.

PBM BOOT manager

PBM decides what will be run by the CF2 on power up or reset. Unless there is a problem, PBM will vector to the selection made by the BOOT command (from PBM or PicoDOS) or resume a user application returning from suspend mode.

When does the CF2 go into PBM mode?

The PBM sign on indicates what vector caused it to boot. The indicator is a single character at the end of the second line of dashes in the sign on message, highlighted in red below.

```
-----  
CF2 Permanent Boot Monitor  
www.persistor.com      V4.00  
-----P  
== FLASH WRITES ENABLED ==  
  
PBM>
```

The CF2 will go into PBM mode under these conditions;

1. If the CF2 is set to automatically boot PBM without timeout. There is no vector character.
2. If the CF2 is set to automatically boot PBM with timeout. The vector character will be T.
3. If you type PBM <enter> while in PicoDOS. The vector character will be P.
4. If IRQ5 is grounded when the CF2 is reset or powered up. Persistor base boards like the R2 and MRCP provide a PBM button for this purpose. A button is also provided for RESET. Hold the PBM button in and push the RESET button and the CF2 will boot PBM. The vector character will be I.
5. If PicoDOS is missing or has a 'fatal error' when tried to boot. The vector character will be X.
6. If an application is exited by BIOSResetToPBM();

PicoDOS Details

PicoDOS and ToPico Projects

All of the commands that will be described as belonging to PicoDOS can be included, selectively, in your application if you use a ToPico project for your application. By using ToPico, you incorporate the Table Driven Command Processor into your application. You can then easily include whichever PicoDOS commands you would like your users to have.

You can easily permit them to set the clock, for instance, or get a directory of files on the CompactFlash card. By leaving out certain commands you can prevent the user from performing more dangerous actions. For example, you might not want your users to have the FORMAT command. Typically, when you write a ToPico program, you expect your user to activate your specific application code by typing a command line. For instance you might use CALIBRATE as the command to launch code to calibrate your sensor and RECORD as the command to start recording data. The command processor can parse a command line so that the necessary parameters for a command can be included after it.

Two examples are provided in a CF2 starter kit. The CommLogger and PicoDAQ are compiled from ToPico projects and demonstrate command line parsing.

When a user application runs, the "real" PicoDOS will still be available in its space.

PicoDOS and Memory Cards

PicoDOS supports a FAT file system and supports standard volume management commands for the on board CompactFlash card. The memory card is treated as a DOS drive in PicoDOS. Typically a memory card purchased from Persistor Instruments will just show up as drive C:\> at the PicoDOS prompt. The file system is self-selective from FAT16 to FAT32, depending on the volume size, and supports directories. File naming is limited, however, to 8.3 standards. For instance MYLOGGER.DAT is 8 characters followed by a 3 character extension.

There are practical limitations on the number and size of files possible on a CF2 memory card. See appendix A for a list of these limitations.

Power Cycled Memory Card Detect is Necessary

The CF2 hardware can only detect the presence of a memory card at power up. The PicoDOS sign on message will declare that the memory card is missing if not detected.

```
-----  
Persistor CF21M   SN 08358   PicoDOS V4.03r1   PBM V4.00  
(C) 1998-2007 Persistor Instruments Inc. - www.persistor.com  
-----  
  
CompactFlash card missing, most features disabled
```

If a memory card is not installed at power up, then PicoDOS will not recognize a memory card installed until the next power cycle.

NOTE reset does NOT cause the CF2 to perform a card detect.

You can, however, ask PicoDOS to manage its own power cycle by using the CCC (Cooperative Card Change) command. The CCC command executes a SUSPEND mode, in which the CF2 power manager cycles power to the 68332. Note, in the example below, how the CF2> prompt is presented when the CF2 does not recognize a memory card and C:\> is presented when a memory card is recognized.

```
CF2>CCC  
You have 30 seconds to insert a CompactFlash card...  
  
C:\>CCC  
You have 30 seconds to remove the CompactFlash card...  
You have 30 seconds to insert a CompactFlash card...  
  
C:\>
```

Similarly You will want to incorporate the CCC command, or equivalent into your application if you want your users to be able to change memory cards without removing power from the CF2.

NOTE the CCC command will NOT work properly unless a backup battery supplies power to VBBK.

PicoDOS Commands Depending on Memory Card

While in PicoDOS, typing HELP<enter> will produce a list of available commands. Be aware that, if there is no memory card detected, PicoDOS will only display the commands that are possible without a memory card installed.

PicoDOS and the Real Time Clock

The CF2 has a real time clock which can be accessed by PicoDOS. At sign on, if the time does not seem to make sense, PicoDOS will declare so and suggest setting the clock. The unset time does not have a default value and will vary from CF2 to CF2. The clock is run in the CF2 power manager, which is a separate co-processor that remains powered when the main battery supply is removed. The CF2 Power manager must be supplied by a backup battery input to VBBK in order for the clock to be run when main power is cycled. Persistor's R2 and MRCP base boards provide a CR2032 backup battery for this purpose.

```
-----  
Persistor CF21M   SN 08358   PicoDOS V4.03r1   PBM V4.00  
(C) 1998-2007 Persistor Instruments Inc. - www.persistor.com  
-----
```

```
The clock needs setting if it's not July 31, 1937!
```

```
C:\>DATE 09/16/11 11:49:11
```

```
Clock reads: Friday, September 16, 2011 11:49:11 am
```

```
C:\>
```

The Backup Battery

The CF2 uses an MSP430 and some voltage detection circuitry to manage SUSPEND mode. In this mode, the MSP430 shuts everything down except itself and the RAM. This requires the CF2 to have a backup battery supplying VBBK typically with 3 volts. Persistor base boards provide a CR2032 backup battery for this purpose. The MSP430 also maintains the RTC for the CF2. Time will be kept while the CF2 is powered by the main battery input but will be lost when depowered if the backup battery is not installed.

The CCC command, used to change a memory card in a powered CF2, will not work without the backup battery.

The Virtual EEPROM

What is VEE and How It Works

The CF2 has a "Virtual EEPROM" which is actually a section of the program flash reserved for persistent variables managed transparently by PicoDOS. You have full access to the VEE through your application (see the CF2 programmer's manual) but you can also access VEE from PicoDOS. The SET command will be described later but this section provides a brief description of the VEE.

The VEE is meant to hold small variables that are not changed often. This is because flash life is limited to the number of erase/write cycles.

The VEE is constructed from two 8KB flash pages, either one of which is valid at any time. 500bytes are reserved for the PBM variables and 7.5KB are available to the user. Entries are written, with 16bit CRC, sequentially on to one page. If a new value is given to an already written variable, the re-entry is appended. This is because you cannot simply rewrite a small location in flash. It must be erased (zeroed) by blocks and then it can be written (ones overwritten). When a page is filled, PicoDOS reads the page, eliminates the old unused values, rewrites the updated entries to the

second page, and then erases the page previously written. When PicoDOS signs on, it checks the VEE and will report an error, if any is found. A fetch from the VEE also verifies the CRC. PicoDOS limits the VEE to two page erases between resets.

VEE entries can be longs, floats and C strings, as well as binary data for custom types and structures.

User Variables

Examples of user variables might be calibration data for a sensor, or parameters affecting the operation of your application. For example, Persistor's CommLogger application keeps run time values in VEE for baud rate, system clock speed, low power mode, memory buffer size, and I/O mirror pins for low power setup.

System Variables

In addition to user variables, the VEE also holds system variables for PBM and PicoDOS. For instance the PBM boot preference and flash write privilege are stored there and PicoDOS SYS.NAME system variables, that are changed from the default values, are also stored in VEE. As an example, the default baud rate for PicoDOS is 9600, but SYS.BAUD may be set to another baud rate.

The system variables or "Persistent Environment Variables" and their default values are defined in `_cfx_globals.h`.

Loading & Running CF2 Applications

Applications can be loaded to the CF2 using MotoCross, with a utility program from a CompactFlash card, or by modem if necessary. They can be run from flash, RAM or CompactFlash. They can be launched by the PBM boot preference, by PicoDOS with an `autoexec.bat` file, or directly by the user.

.APP, .RUN and .PXE files

Among the files produced, when a CodeWarrior project is compiled, are an `.APP` file and a `.RUN` file.

The `.APP` file is, by default, targeted to load to the user application space in flash memory starting at E40000. It will reside in the program space until it is CLEARED or overwritten by a new load. Flash programs run from flash, but the non-constant static and initialized data is loaded from flash into ram by startup code that executes before calling `main()`.

The `.RUN` file is targeted to load to RAM, from which it can be run once.

The CF2 can also execute a "Persistor Executable" with the extension `.PXE` from the CompactFlash card. The `.PXE` file is not compiled by CodeWarrior, rather it is created by PicoDOS from a `.RUN` file, after loading it with MotoCross. The `.PXE` file will be loaded to and run from RAM.

Loading .APP and .RUN files with MotoCross

MotoCross is the PC based application that the user should use to interact with the CF2 and to load compiled applications to it. Motocross negotiates with PBM to place a compiled application into the proper location and ramps the baud rate up for the duration of the load. MotoCross also calculates and embeds a CRC into the file to ensure the load is valid.

Managing CF2 Behavior With PicoDOS

Simply choose "Load..." from the Transfer menu in Motocross and navigate to the .APP or .RUN file in the bin folder located in your CodeWarrior project folder. PBM will be called on the CF2 and MotoCross will negotiate the entire transfer.

After the file is loaded, a "G" will appear in the MotoCross monitor window. Hit the <enter> key and the application will launch, either from RAM if it was a .RUN load or from flash if it was a .APP load.

```
C:\>
###MX Load: CRC E40000 E4FFFF = 9CA76FA5
G
-->"??" @ E40000

hello, world

C:\>
```

Loading PicoDOS with MotoCross

It is nearly impossible to delete the PBM monitor from the CF2 but PicoDOS can sometimes become corrupted by a failed application load, or sometimes the user CLEARs PicoDOS by accident. PicoDOS can be installed as an .APP file and is loaded the same way as any other .APP. The file is already targeted to the proper flash location.

All modern PicoDEV installations include the CF2 PicoDOS application file here;

[C:\Program Files\Persistor\MotoCross Support\CFX\bin\CF2\CF2R4031.APP](#)

NOTE if you are loading a different version of PicoDOS than you already have, you should use the RESTORE2 utility with a memory card, to insure the proper version of PBM is installed, which will NOT be updated when loaded with MotoCross.

Saving .RUN files as .PXE files on CompactFlash

To load an application to the CompactFlash with MotoCross, load the .RUN file and place an "S" after the "G" presented by the CF2, followed by a space and a name of 8 characters or less. The file will be written to CompactFlash with a .PXE extension. The .PXE and .RUN files are slightly different due to the means of creation, but are effectively the same, when stored on the CompactFlash card.

Once the .PXE file has been created, it can be invoked from PicoDOS by simply typing its name.

```
C:\>
###MX Load: CRC 010000 0159BD = 1D722EA8
GS HI_EARTH
Saving 22974 bytes to HI_EARTH.PXE...

C:\>DIR

Volume in drive C has no label
Volume Serial Number is 18EC-16A1
Directory of C:\

HI_EARTH.PXE                22,974   01-01-98   6:43p
    1 file(s)                22,974 bytes
    0 dir(s)                 1,024,622,592 bytes free

C:\>HI_EARTH
```

```
hello, world
```

```
C:\>
```

Placing .RUN files directly on CompactFlash

A .RUN file can be launched (into RAM) from the CompactFlash card by typing its name while in PicoDOS. The .RUN file would be placed on the CF card using a Windows PC card reader.

Loading .APP Files with RESTORE2

RESTORE2.RUN is a utility application that will locate .APP files on the CompactFlash and load the selected .APP file to the program space in the CF2 onboard flash. All modern PicoDEV installations include the RESTORE2.RUN application file here;

[C:\Program Files\Persistor\MotoCross Support\CFX\bin\CF2\RESTORE2](#)

Simply place RESTORE2.RUN and one or more .APP files on your CompactFlash, install it in the CF2, and type RESTORE2 <enter> in PicoDOS.

```
C:\>dir
```

```
Volume in drive C has no label
Volume Serial Number is 18EC-16A1
Directory of C:\
```

```
RESTORE2.RUN          153,914   07-20-05   5:02a
HELLOCFX.APP          23,010   11-19-10  12:57p

                2 file(s)             176,924 bytes
                0 dir(s)       1,024,458,752 bytes free
```

```
C:\>RESTORE2
```

```
CF2 Restore to ROM
Version 2.20 (using PicoDOS 4.02r1)
```

```
Searching for *.APP files on C: ...
```

```
Restore HELLOCFX.APP [N] ? Y
Erasing flash from E40000 to E459BD ...
Writing flash .....
Complete, resetting
```

```
-----
Persistor CF21M   SN 08358   PicoDOS V4.02r1   PBM V4.00
(C) 1998-2005 Persistor Instruments Inc. - www.persistor.com
-----
```

```
The clock needs setting if it's not January 01, 1970!
```

```
C:\>APP
hello, world
```

```
C:\>
```

Loading Application Files by Modem

Sometimes a user ends up needing to replace an application in a CF2 which is only accessible by modem. You cannot load files with MotoCross through a modem, but you can X or Y modem RESTORE2.RUN and your new .APP file to the CF2. The CF2 will place the received files on the CompactFlash card. Then you can invoke RESTORE2 in PicoDOS through the modem to load the .APP file into flash. MotoCross does not perform X or Y modem transfers, so you will have to use another terminal program for that.

BACKROM and RESTORE2

Sometimes a user would like to clone a CF2, including the application, versions of BIOS, PBM, PicoDOS, and the contents of VEE including system variables. Or perhaps just save an image of the whole system. Or perhaps make a copy of an inherited file for which the project has been lost. This is possible using the BACKROM command, which creates a fully reloadable .APP file.

An example is the ComLogger application provided with the CF2 starter kit. The .APP file contains not only the application, but also an older version of PicoDOS and an image of the VEE with ComLogger launch time variables.

BACKROM .APP files should be loaded onto other CF2s using the RESTORE2.RUN utility to insure the PBM copy is also executed, if PBM is part of the file. (Motocross will not install the PBM portion.)

```
C:\>BACKROM /?
Backup application in flash memory to CPM68K file

BACKROM [drive:][path]filename [/S /A /V /P /I]
filename      Specifies the file to save to
/S            System only (PBM, VEE, BIOS & PicoDOS)
/A            Application only, exclude system
/V            Exclude VEE (Virtual EEPROM)
/P            Exclude PBM
/I            Binary image only (no CPM68K header)

C:\>BACKROM TODAY
Condensing VEE ...
Checking for non-erased from E40000 to EFFFFFF ...
Creating TODAY.APP ...
Computing CRCs:  PBM:400=0x487265A1,  BIOS:402=0xBA4B855C
                  PICO:402=0x49FA1A72,  USER:???[64KB]=0xC4C7E1B7
Writing 36 byte header into TODAY.APP ...
... appending 311296 bytes flash from E04000 to E4FFFF ...
... appending 16384 byte PBM for RESTORE ...
Complete

Note: The PBM portion will only be installed by the RESTORE program.
      Loads from MotoCross will ignore the PBM block.
C:\>DIR

Volume in drive C has no label
Volume Serial Number is 18EC-16A1
Directory of C:\

TODAY.APP                327,716   01-01-98   7:55p
                        1 file(s)          327,716 bytes
```

```
0 dir(s)      1,024,311,296 bytes free

C:\>
```

Batch and AUTOEXEC.BAT files

PicoDOS supports batch files. A batch file is a text file with the extension .BAT that contains commands the system (PicoDOS in this case) can execute.

If the CF2 is set to boot PicoDOS, and if an AUTOEXEC.BAT file exists on the root of the first partition of a detected memory card, or an alternately defined path, then PicoDOS will execute the AUTOEXEC.BAT file on power up or reset.

The AUTOEXEC file can launch any .RUN or .PXE file on the CompactFlash card or execute the APP command to launch the application installed in on-board flash. You could have several applications then, installed on your memory card, and have the applications erase the existing autoexec file and replace it with a new autoexec file before they complete and then exit to PicoDOS, which would then read the new autoexec file and follow those new instructions, which might include launching another application.

The CF2 programmers guide describes batch files in more detail, but here is an example of a very simple autoexec file that sets the clock to midnight, deletes a configuration file, and launches a program in flash. For the example, we will create the autoexec file and configuration file using the CAPTURE command.

```
C:\>DIR

Volume in drive C has no label
Volume Serial Number is 18EC-16A1
Directory of C:\

0 file(s)      0 bytes
0 dir(s)      1,024,655,360 bytes free

C:\>CAPTURE CONFIG.CFG
Ready, <CTRL-C or BREAK> to end
THIS IS MY CONFIGURATION FILE (use F8 to create a break from MotoCross)

Complete

C:\>CAPTURE AUTOEXEC.BAT
Ready, <CTRL-C or BREAK> to end
TIME 00:00:00
ERASE CONFIG.CFG
APP (use F8 to create a break from MotoCross)

Complete

C:\>DIR

Volume in drive C has no label
Volume Serial Number is 18EC-16A1
Directory of C:\
```

```
CONFIG.CFG                29  09-21-11  4:21p
AUTOEXEC.BAT              42  09-21-11  4:22p

      2 file(s)              71 bytes
      0 dir(s)    1,024,622,592 bytes free

C:\>BOOT
Boots:
PicoDOS

C:\>RESET
Resetting...

TIME 00:00:00

Clock reads: Wednesday, September 21, 2011  12:00:00 am

ERASE CONFIG.CFG

1 file(s) erased
APP

      hello, world

C:\>DIR

Volume in drive C has no label
Volume Serial Number is 18EC-16A1
Directory of C:\

AUTOEXEC.BAT              42  09-21-11  4:22p

      1 file(s)              42 bytes
      0 dir(s)    1,024,638,976 bytes free

C:\>
```

Loading and Off-Loading Data Files

Using a Memory Card Reader

Inexpensive memory card readers for your windows PC are commonly available. If you can eject your CF card from the CF2 and use a memory card reader to copy files, that will be the most sensible and fastest way to offload data files.

Be aware that there are a few cautions involved with using a memory card reader.

1. Always format your CF2 memory cards with PicoDOS; never with your PC.
2. Erase your files with PicoDOS; not with your PC.
3. Moving a file from the CF card onto a PC, moving a file from PC to PC, or directory to directory, and then moving it back onto the CF2 may cause file attributes to change, or even change the extension. For example, some versions of WIN7 will rename autoexec.bat as autoexec.bak and set the archive, system and hidden attributes.

Loading data files with MotoCross

Non-application files can be loaded to the CompactFlash with MotoCross by using the Send Binary File... selection in the Transfer menu. Remember that you can change both the CF2 and the MotoCross baud rates, so bumping them up to 115,200 will speed up large transfers.

Using XModem or YModem

PicoDOS can send and receive files by XModem or YModem. MotoCross, however, does not offer XModem or YModem receive or send, so another terminal program would have to be used. In general, the receiving end is set up and initiated, and then the sending end is initiated. See XR, XS, YR and YS in PicoDOS commands.

Using USB offloads

There is an add-on board for the CF2, called the USB-2, which is manufactured by Oceanographic Embedded Systems. The user installs one application on the CF2 and a separate application on his PC. When both applications are running, the PC application will display the CF2 directory and the user can select files and download them to the PC. The USB-2 software is only capable of offloading files from the CF2. The CF2 cannot initiate the offload. The data rate is limited by the CF2 hardware but is about 200Kbytes per second, which is the fastest offload possible without removing the memory card from the CF2.

PBM commands

While in PBM, typing ? <enter> or HELP <enter> will bring up a list of ten available commands, but there are actually a few more. None of the commands display syntax or summary information.

Commands may be typed in lower or upper case. The <enter> key activates the command.

HELP – get a list of commands

Produces a list of commands

RES - reset

Forces a complete system reset equivalent to grounding the /RESET signal.

PICO – reset into PicoDOS

Exits PBM and starts PicoDOS, if possible.

WREN – enable writes to flash

The CF2 defaults to writes to flash enabled, but certain activities may trigger disabling of writes. The PBM sign-on message will indicate whether flash writes are enabled or disabled. If a command that requires writes to be enabled is issued when writes are disabled, the PBM will reply with "can't !"

The WREN command must be the first command issued after booting PBM and IRQ5 (the PBM button) must be grounded when the enter key is hit. Finally WREN needs to be confirmed by entering Y.

```
CF2 Permanent Boot Monitor
www.persistor.com    V4.00
-----T
== FLASH WRITES DISABLED ==
=== TIMEOUTS ACTIVATED ===

PBM>60>WREN
Can't (pin 39 +) (this message given because IRQ5 was not grounded)
PBM>WREN
Too late (this message given because the new WREN is now the second command issued)
PBM>RES
Resetting...

-----T
CF2 Permanent Boot Monitor
www.persistor.com    V4.00
-----T
== FLASH WRITES DISABLED ==
=== TIMEOUTS ACTIVATED ===

PBM>60>WREN (this is the first command and IRQ5 (pin 39) is grounded)
Release 39 and confirm (Y/N) ? Y
== FLASH WRITES ENABLED ==

PBM>
```

BOOT – choose or view what boots next

Typing BOOT will present all possible choices and allow the user to choose one.

```
PBM>BOOT
Options:
  9 - PicoDOS at E10000
=> A - Program at E40000
  B - Program at E80000
  C - Program at EC0000
  D - PBM (LPSTOP timeout)
  E - PBM (LPHALT timeout)
  F - PBM (no timeout)
Select (9,A-F) [A] ? 9
Next: => 9 - PicoDOS at E10000
```

Simply typing BOOT and your choice will also set the default boot.

```
PBM>BOOT PICO
Next: => 9 - PicoDOS at E10000

PBM>BOOT PBM
Next: => F - PBM (no timeout)

PBM>BOOT APP
Next: => A - Program at E40000

PBM>bOOT D
Next: => D - PBM (LPSTOP timeout)
```

A is the default location for the user application.

Managing CF2 Behavior With PicoDOS

Selection D and E will boot PBM with a one minute timeout. The PBM sign-on message will state TIMEOUTS ACTIVATED if true. The PBM monitor will count down the seconds by tens, until the last 10 seconds, which it counts down by ones. If a program load is initiated during the minute, the timeout will be cancelled. If a command is entered, completion of the command will restart the timeout.

For boot selection D, The CF2 will drop into LPSTOP mode, after timeout, until an incoming character arrives on the main SCI UART. In that event the PBM will be reinitiated.

For boot selection E, The CF2 will drop into LPSTOP mode, after timeout, until the next reset or power cycle.

CHECK & CLEAR – check for erased flash & erase flash

CHECK an address range to see if it is erased.

Normally, CLEARing flash is not necessary. Loading an application or PicoDOS automatically erases the target space. CLEAR without a memory range will erase everything except PBM, so PicoDOS will then have to be reloaded.

```
PBM>CHECK E40000 E4FFFF
Checking E40000-E4FFFF ...
E40000-E4FFFF not erased

PBM>CLEAR E40000 E4FFFF
Erase E40000-E4FFFF (Y/N) ? Y
Erasing ... ok

PBM>CHECK E40000 E4FFFF
Checking E40000-E4FFFF ...
All erased
```

BAUD – view or change the PBM baud rate

PBM's default baud rate is 9600. You can change it to any other baud rate from 300 to 230,400 but the other parameters remain fixed as 1 stop bit, 8 data bits and no parity. The baud change only remains in effect until the next reset. The actual baud rate produced, based on the system clock frequency, is calculated and returned.

```
PBM>BAUD
9583 BAUD
PBM>BAUD 19200
Setting 19200 BAUD
Hit <enter> .. . <enter> after changing MotoCross baud rate
19166 BAUD
```

LOAD & LO – load binary files & load S records (hex files)

Normally, a user will load application files or binary data with MotoCross; the monitor/development program provided by Persistor Instruments. MotoCross automatically initiates the proper loading command.

LOAD instructs the PBM to receive a binary load file (CPM68K) from Motocross. The user does not need to use the LOAD command.

Managing CF2 Behavior With PicoDOS

LO instructs the PBM to prepare to receive standard Motorola S records. The monitor stops echoing characters and expects to receive a stream of S records, and does not return until it sees a final valid S record signaling the end. The user does not need to use the LO command.

G – Go execute the last or the specified load

The G command launches a loaded application. Normally PBM places a G after loading an application and the user just needs to hit the <enter> key. See "Saving .RUN files as .PXE files on CompactFlash" for using the GS command.

MD – Display memory

Display memory with [address:length] or [address address] or next line with <enter.>

```
PBM>MD E40000:20
00E40000 6000 0006 0000 0000 4BFA FFF6 206D 0068 ~.....K... m.h
00E40010 4EE8 0000 206D 006C 4EE8 0000 4E7A 8801 N... m.lN...Nz..
PBM>MD E40000 E40020
00E40000 6000 0006 0000 0000 4BFA FFF6 206D 0068 ~.....K... m.h
00E40010 4EE8 0000 206D 006C 4EE8 0000 4E7A 8801 N... m.lN...Nz..
PBM><enter>
00E40020 2068 0004 4EE8 0000 00E4 0000 0000 59BE h..N.....Y.
PBM><enter>
00E40030 00E4 0000 0000 526E 000C 0000 0001 0000 .....Rn.....
PBM>
```

ERR – Display previous error message

ERR re-displays the most recent error and clears the error flags.

The four fields in the response are: numeric error code, error count, and two optional error dependent parameters. Only the first error code encountered is logged as some kinds of errors can result in meaningless cascading errors, and the error count lets you know if this has happened.

Below are error codes and decoding:

Code	Description	Parameter 1	Parameter 2
0	No error	none (zero)	none (zero)
1	UART receive buffer overflow (probably during load)	queue head	queue tail
2	Flash write or erase failed (timed out)	failed address	data word
3	Attempted to burn flash at odd address	failed address	data word
4	Attempted to burn a zeroed bit to one	failed address	data word
5	Attempted to burn or erase an invalid flash address	top address	bottom address
6	Attempted to erase a block more than one time	top address	bottom address
7	S-Record (hex) load error	failed address	error code

```
PBM>F
Huh?
PBM>ERR
Error: "Huh?" 0:0, 0x0, 0x0
PBM>ERR
```

None
PBM>

PicoDOS commands

While in PicoDOS, typing ? or HELP <enter> will bring up a list of available commands. If the CF2 did not detect a memory card at the last power cycle, then PicoDOS will not display the commands that require the presence of a memory card.

PicoDOS commands can be entered as lower or upper case and are executed when the <enter> key is hit. Generally, command parameters are entered after the command name separated by space/.

Most PicoDOS commands will return a summary, options and syntax if the command is entered with a /?. For example COMMAND /?.

This section is not meant to repeat the information gotten from the PicoDOS monitor, but rather to enhance the information where necessary.

*/ vs *

In case you have forgotten, or never learned, \ is used for directory paths and / is used for command parameter indicators.

This is a directory path	C:\BANANA\CREAM\PIE\
This is a command structure	CHLDSK /B /I
This is a command with a path	PATH C:\BANANA\CREAM\PIE /P

System Commands

BAUD – view or set the baud temporarily or permanently

The default baud rate for PicoDOS is 9600. You can change this just for your picoDOS session or you can change the default by setting the sys.baud in VEE with the /P modifier. The actual baud produced is a function of the system clock and PicoDOS may suggest changing the system clock for some baud rates. If done, the sys.clock setting will also be set in VEE. The actual baud (assuming the system clock is perfect) is displayed. In the example below, we look at the two system variables afterward with the SET command.

```
CF2>BAUD
BAUD = 9615,N

CF2>BAUD 115200 /P
Requesting 115200 BAUD,N (actual 115000, 0.1% error) with 14720kHz clock
Change terminal baud rate now --- hit <Enter> when ready ...
... also change SYS.CLOCK to make this permanent [Y] ? Y
BAUD = 115000,N

CF2>SET
===== BINARY DATA =====
SYS.BAUD= (LONG) 115200 [0x1C200]
SYS.CLOCK= (LONG) 14720 [0x3980]
```

Managing CF2 Behavior With PicoDOS

CF2>

You can enter any baud rate. If you forget the value, or make a mistake, you can recover this way;

1. Boot into PBM by grounding IRQ5 (the PBM button) and resetting. PBM always boots at 9600 baud.
2. Now that you are in PBM at 9600 baud, while still grounding IRQ5 (the PBM button), type PICO and <enter>. This will force PicoDOS to boot at 9600 baud too.
3. Use the baud command to properly set the baud rate.

BACKROM –backup all or part of the CF2 contents of flash

Backrom allows you to save any combination of the CF2 application, BIOS, PBM, PicoDOS and VEE to a reloadable .APP file.

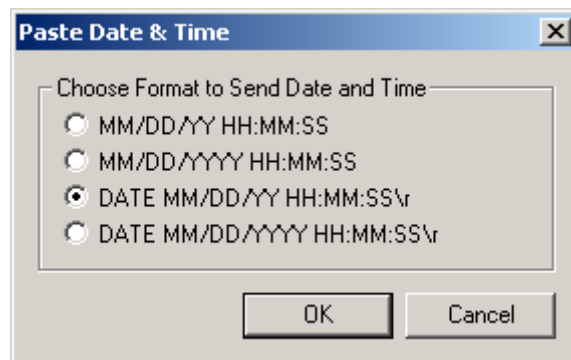
See "**BACKROM and RESTORE2**" in "**Loading & Running CF2 Applications**" for more details.

CCC – change or insert a new memory card

The "Cooperative Card Change" command executes a power cycle to detect a new memory card. The CF2 backup battery must be in place for this command to work. A CF2 reset will NOT cause the CF2 to detect a new or changed memory card, so CCC or a manual power cycle is required to change or install a memory card.

DATE and TIME – set the real time clock

These commands can be entered manually, but the simplest method to set the CF2 clock is to paste the date and time in from the MotoCross Edit menu. Choose the third option.



```
C:\>DATE 09/23/11 15:44:10
Clock reads: Friday, September 23, 2011 03:44:10 pm
C:\>
```

The DATE command offers three formats for order, and 12 or 24 hour display.

The CF2, seated in a persistor base board, with a backup battery, will maintain the clock when the main battery power is removed.

LO – load executable files

The average user will never need this command. Just load your .RUN or .APP files with the MotoCross Load... choice from the Transfer menu and Motocross will execute this command in the background. Here are the details for the rare instance they are needed;

Managing CF2 Behavior With PicoDOS

Syntax LO [offset] [Bx[+]] [G]

Offset specifies the location to write the file.

B controls the baud rate of the transfer.

0 = 9600 1 = 19200 2 = 38400 3 = 57600 4 = 76800 6 = 115200

G tells the Persistor to "go" to the code for execution immediately after loading.

+ allows overwriting PBM; a highly dangerous activity.

Examples:

```
C:\> LO B3                    ; Loads at 57600 baud
C:\> LO B1G                  ; Loads at 19200 baud and executes
C:\> LO 00010000 B2        ; Loads to offset 10000 at 38400 baud
```

PATH – set PicoDOS search path for executable files

This command sets the search path, which PicoDOS will search when it boots. The default path is C:\ but you can change it to another directory. The /P modifier is required to set it as a system variable named SYS.PATH. In the example we change the path to an already created subdirectory, check it in VEE, make the autoexec.bat file, and then test a reset.

```
C:\>PATH C:\BANANA\CREAM\PIE /P
C:\>SET
SYS.PATH=C:\BANANA\CREAM\PIE

C:\>CHDIR C:\BANANA\CREAM\PIE
\BANANA\CREAM\PIE

C:\BANANA\CREAM\PIE\>CAPTURE AUTOEXEC.BAT
Ready, <CTRL-C or BREAK> to end
APP (F8 to break)

Complete

C:\BANANA\CREAM\PIE\>CHDIR C:\

C:\>BOOT
Boots:
PicoDOS

C:\>RES
Resetting...

APP

hello, world

C:\>
```

PROMPT – set the PicoDOS prompt

Change the standard prompt. Make it permanent with the /P modifier.

```
C:\>PROMPT ENTER$Q$G /P
ENTER=>SET
SYS.PROMPT=ENTER$Q$G
```

Managing CF2 Behavior With PicoDOS

ENTER=>

Prompt can be made up of normal characters and the following special codes:

- \$Q = (equal sign)
- \$ \$ (dollar sign)
- \$T Current time
- \$D Current date
- \$P Current drive and path
- \$V PicoDOS version number
- \$N Current drive
- \$G > (greater-than sign)
- \$L < (less-than sign)
- \$H Backspace (erases previous character)
- \$E Escape code (ASCII code 27)
- \$_ Carriage return and linefeed
- \$S Space character (ASCII code 32)
- \$C Comma character (ASCII code 44)
- \$Xhh Any hexadecimal character

RES or RESET – execute a hard reset

Equivalent to grounding the RESET pin.

SET – set a user or system variable into VEE

SET is the PicoDOS gateway to the VEE variables. See “The Virtual EEPROM” in this document for a description of the VEE.

With no modifier, SET displays the contents of VEE.

With the /E modifier, SET erases the contents of VEE. A RESET is necessary after erasing the VEE.

Variables may be SET directly or the SET command can query for the name and value. SYS.NAME variables may also be SET by other PicoDOS commands with the /P modifier.

```
C:\>SET /?
Displays, sets, or removes PicoDOS environment variables

SET [variable=[string]] [/SLFE?]
variable  Specifies the environment-variable name
string    Specifies a series of characters to assign to the variable
/S        Query for variable name and string data
/L        Query for variable name and long integer data
/F        Query for variable name and floating point data
/E        Erase all of the environment variables

Type SET without parameters to display the current environment variables

C:\>SET
CMLG.BAUD=9600
CMLG.CLOCK=14720

C:\>SET SYS.BAUD=19200
```

```
C:\>SET /F
VEE variable name ? DEMONSTRATION
Floating point data ? 123.45

C:\>SET
CMLG.BAUD=9600
CMLG.CLOCK=14720
SYS.BAUD=19200
DEMONSTRATION= (FLOAT) 123.45 [0x42F6E666]
```

VER – get system version information

Use VER /I to get information on all installed Persistor firmware as well as CRC verification on those and the user application space.

```
C:\>VER /I
CF21M    SN 08358
PicoDOS  4.02r1
BIOS     4.02r1
PBM      4.00
SPV      5.20-P
TPU      P.01
TLC      1.03

Dev:CF2   Code:PBM   Ver:400   CRC:487265A1...Verified
Dev:CF2   Code:BIOS  Ver:402   CRC:BA4B855C...Verified
Dev:CF2   Code:PICO  Ver:402   CRC:49FA1A72...Verified
Dev:CF2   APP:E40000-E4FFFF  CRC:FBE5DA44...Verified
C:\>
```

Launch Commands

APP –immediately launch application at E40000

If there is no application loaded, the command returns application error 2015.

If the application can take command line modifiers, they can be passed after the APP command. Here is an example of launching PicoDAQ with modifiers;

```
C:\>APP -FTEST001 -H500.500.500.500.500.0.0.0 -D1000 -S2000 -V
This program: PicoDAQ.c: Apr  5 2002 17:23:08

Command line arguments: argc = 6
  argv[0] = <APP>
  argv[1] = <-FTEST001>
  argv[2] = <-H500.500.500.500.500.0.0.0>
  argv[3] = <-D1000>
  argv[4] = <-S2000>
  argv[5] = <-V>

Command line interpreted as:
  filename:      'TEST001'
  cmdline:      'APP -FTEST001 -H500.500.500.500.500.0.0.0 -D1000 -S2000 -V'
  startDelay:    1000
  sampleDuration: 2000
  bipolar:       0
  differential:   0
```

```
validParams:      1
ioStartStopPin:   0
breakStartStop:   0
motoBinFormat:    0
asciiFormat:      0
verboseDiag:      1
```

```
Finishing up...
```

```
C:\>
```

BOOT – set the boot vector for reset or power cycle

BOOT without modifier returns the current boot vector.

BOOT PICO or BBOT PBM or BOOT APP will happen immediately and become the default vector.

G or GO – launch application at address

The G command is automatically invoked by MotoCross after downloading a new application. Normally the user only need hit <enter> after the G prompt placed in the MotoCross terminal window.

```
G or GO [addr] [arg1] [argN...] [/A] [/F[n]]
Addr      Absolute hex address for /A switch.
argN      Command line arguments to pass.
/A        First argument is an absolute hex address.
/Fn       Target is standard flash application at 0xE40000
          or 0xEn0000 where n is typically 4, 8, or C
```

PBM or MON – launch PBM

Launches the PBM monitor immediately.

Drive (Memory Card) Commands

The CompactFlash memory card appears, by default, as the C:\ drive in PicoDOS. It is managed with traditional DOS commands.

CCC – change or insert a new memory card

The CF2 hardware can only detect the presence of a memory card at power up. If a memory card is not installed at power up, then PicoDOS will not recognize a memory card installed until the next power cycle. The CCC command executes a SUSPEND mode, in which the CF2 power manager cycles power to the 68332. Note, in the example below, how the CF2> prompt is presented when the CF2 does not recognize a memory card and C:\> is presented when a memory card is recognized.

```
CF2>CCC
You have 30 seconds to insert a CompactFlash card...

C:\>CCC
You have 30 seconds to remove the CompactFlash card...
You have 30 seconds to insert a CompactFlash card...

C:\>
```


NOTE the CCC command will NOT work properly unless a backup battery supplies power to VBBK.

CHKDSK – check drive (memory card) information

CHKDSK is not normally required by the average user.

CHKDSK /B /I will display the DOS Boot Record and the Identify Device parameters.

CHKDSK without a modifier will identify physical sector errors and the /F modifier will attempt to fix them.

FDISK – display or modify a drive (memory card) partitions

FDISK is not normally required by the average user, but it can be used to see or modify the partitions available on a drive (memory card).

FORMAT – format a drive (memory card) for PicoDOS

A new memory card, not purchased from Persistor, should ALWAYS be formatted using the PicoDOS FORMAT command. Normally the card should just be formatted without any command modifiers.

FORMAT is also a quick way to erase the card. The normal FORMAT does not actually destroy files; rather it erases the pointers to the files, so the data is still on the card after a simple FORMAT.

FORMAT /E will take more time, dependent on the size of the card, but it will fill the entire partition with FFs which can be handy if a data recording session goes badly. It may help in data recovery to know the entire card started as FFs before the recording began.

In PicoDOS 4.x and up, the card is formatted by as FAT16 or FAT32 governed by the card size (FAT32 if > 2GB) but this can be changed with the SYS.F32MBCUT system variable.

The /D modifier should not be needed by a modern PicoDOS user. If a very small memory card is used, the /D modifier changes the maximum number of files allowed on the card by FAT16 rules.

There are practical limitations on the number and size of files possible on a CF2 memory card. See appendix A for a list of these limitations.

MOUNT – mount or dismount a drive (memory card)

MOUNT is not normally required by the average user.

Directory Commands

DIR – display directory contents

There are many DIR options. Use DIR /? To see them.

CHDIR – change directory

MKDIR – make directory

RMDIR – remove directory

```
C:\>DIR
```

```
Volume in drive C has no label  
Volume Serial Number is 18EC-6600
```

```
Directory of C:\

0 file(s)                0 bytes
0 dir(s)                1,024,655,360 bytes free

C:\>MKDIR BANANA

C:\>MKDIR BANANA\SPLIT

C:\>CHDIR BANANA\SPLIT
\BANANA\SPLIT

C:\BANANA\SPLIT>DIR

Volume in drive C has no label
Volume Serial Number is 18EC-6600
Directory of C:\BANANA\SPLIT\

.                <DIR>                09-26-11   4:52p
..               <DIR>                09-26-11   4:52p

0 file(s)                0 bytes
2 dir(s)                1,024,622,592 bytes free

C:\BANANA\SPLIT>CHDIR C:\

C:\>RMDIR BANANA\SPLIT

C:\>
```

File Commands

ATTRIB – display or change file attributes

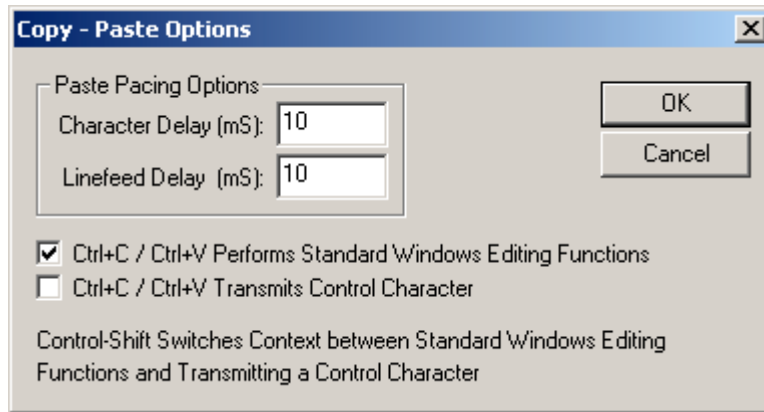
This command sets or clears the read-only, archive, system and hidden file attributes.

CAPTURE – create a file with the keyboard

CAPTURE is used to save incoming characters to a file, on the CompactFlash card. A large file could be written by pasting it into the monitor window preceded by the CAPTURE command and followed by a break.

Using the CAPTURE and TYPE commands with a text editor program permits you to easily get batch or configuration files, copy them to your editor, make changes, and paste them back into the terminal to the CAPTURE command.

If you have Copy-Paste Options set in MotoCross for Ctrl+C / Ctrl+V perform edit functions, then you will need to use a break to conclude a CAPTURE. Use F8 to send a break from MotoCross.



The option /N switch causes capture to append linefeed characters '\n' to incoming carriage returns '\r'. If your captures don't display correctly when you TYPE them back, try using the /N switch.

```
C:\>CAPTURE RECIPE.TXT
Ready, <CTRL-C or BREAK> to end
FLOUR WATER EGGS SALT SUGAR (used F8 here)
```

Complete

```
C:\>TYPE RECIPE.TXT
FLOUR WATER EGGS SALT SUGAR
```

C:\>

COPY – copy a file

```
C:\>COPY RECIPE.TXT C:\INGREDIA.TXT
```

RECIPE.TXT

1 file(s) copied

```
C:\>DIR
```

Volume in drive C has no label
Volume Serial Number is 18EC-6600
Directory of C:\

```
INGREDIA.TXT          27  09-27-11 11:15a
RECIPE.TXT            27  09-27-11 11:11a
```

```
2 file(s)              54 bytes
0 dir(s)      1,024,622,592 bytes free
```

C:\>

DEL or ERASE – erase a file

USE * to indicate anything when erasing multiple files. For instance ERASE *.TXT erases every file with the TXT extension. ERASE *.* erases everything. Of course, if you have more than a few files, FORMAT might be a faster choice to erase the entire partition.

```
C:\>ERASE *.TXT /P
INGREDIA.TXT,      Delete (Y/N)?Y
```

Managing CF2 Behavior With PicoDOS

```
RECIPE.TXT,      Delete (Y/N)?Y
2 file(s) erased

C:\>
```

DUMP – display a binary file in hexadecimal and ASCII

```
C:\>DUMP C:\TEXT.TXT 20 4F
00000020  36 37 38 39 41 42 43 44 45 46 47 48 49 4A 4B 4C  6789ABCDEFGHIJKL
00000030  4D 4E 4F 50 51 52 53 54 55 56 57 58 59 5A 5A 59  MNOPQRSTUVWXYZ
00000040  58 57 56 55 54 53 52 51 50 4F 4E 4D 4C 4B 4A 49  XWVUTSRQPONMLKJI
```

REN or RENAME – rename a file

```
C:\>DIR

Volume in drive C has no label
Volume Serial Number is 18EC-6600
Directory of C:\

THISISIT.TXT                40   08-15-05   8:03p
                            1 file(s)             40 bytes
                            0 dir(s)       1,024,638,976 bytes free

C:\>REN THISISIT.TXT THISISNT.BAT

C:\>DIR

Volume in drive C has no label
Volume Serial Number is 18EC-6600
Directory of C:\

THISISNT.BAT                40   08-15-05   8:03p
                            1 file(s)             40 bytes
                            0 dir(s)       1,024,638,976 bytes free

C:\>
```

SAVE or S– save a block of memory as a file on CompactFlash

With start and end addresses, SAVE can be used to save contents of any valid address space as a file on CompactFlash.

```
C:\>SAVE VEEFILE.SAV E04000 E07FFF
Saving 16384 bytes to VEEFILE.SAV...

C:\>DUMP VEEFILE.SAV 200 22F
00000200  16 53 64 05 43 4D 4C 47 2E 42 41 55 44 00 39 36  .Sd.CMLG.BAUD.96
00000210  30 30 00 00 62 8B 68 06 43 4D 4C 47 2E 43 4C 4F  00..b.h.CMLG.CLO
00000220  43 4B 00 00 31 34 37 32 30 00 6F 18 70 02 43 4D  CK..14720.o.p.CM

C:\>
```

Managing CF2 Behavior With PicoDOS

Without start and end addresses, the addresses from the last LO command (typically Load... from the MotoCross Transfer Menu) will be inferred. As an example, typing S filename, after a .RUN file is loaded with MotoCross, will save the file from its location in RAM to a .PXE file on the Compactflash card.

TYPE – display a text file in ASCII

```
C:\>TYPE THISISIT.TXT
THIS IS A TEXT FILE NAMED THISISIT.TXT
```

```
C:\>
```

Memory Commands

MD – display contents of memory

As of this writing, only the MD [address] [address] works in PicoDOS.

```
C:\>MD E40000 E40020
00E40000  6000  0006  0000  0000  4BFA  FFF6  206D  0068  `.....K... m.h
00E40010  4EE8  0000  206D  006C  4EE8  0000  4E7A  8801  N... m.lN...Nz..
C:\>
```

ML – disassemble contents of memory

```
C:\>ML E40000
00e40000: 6000 0006                bra.w    0xe40008
00e40004: 0000 0000                ori.b    #0,d0
00e40008: 4bfa fff6                lea      0xe40000(pc),a5
00e4000c: 206d 0068                movea.l  0x68(a5),a0
00e40010: 4ee8 0000                jmp      0(a0)
C:\>
```

MM – modify contents of memory

This command only makes sense in RAM. Use MM /? For more detail.

```
C:\>MM 10000
00010000  0000  ?  6000
00010002  0000  ?  0006
00010004  0000  ?  0000
00010006  0000  ?  0000
00010008  00E4  ?  4BFA
0001000A  0000  ?  FFF6
0001000C  0000  ?  . <use period to escape>
```

MR and MW– read and write memory location

These two commands are not intended for manipulating data; rather they are used for troubleshooting hardware by observation with an oscilloscope.

SD – display drive sectors in hexadecimal

```
C:\>SD 0

C: Sector 0 [0x0] (Abs 0x3F)
```

```
00000000 EB3C 9050 6963 6F44 4F53 2000 0220 0100 .<.PicoDOS .. ..
00000010 0200 0200 00F8 F500 3F00 1000 3F00 0000 .....?...?...
00000020 A18B 1E00 8000 2900 66EC 184E 4F20 4E41 .....).f..NO NA
00000030 4D45 2020 2020 4641 5431 3620 2020 5E01 ME FAT16 ^.
Etc . . .
```

Port Test Commands

PicoDOS uses the C connector pin number for the port commands. For instance, to toggle TPU1 type PT 22. The port commands are handy as simple tests to confirm the ports are operating properly, or for immediate manipulation of attached hardware. The following pins are accessible by the PicoDOS port commands, shown in their relative C connector positions;

NAME	POSITION		NAME
DS	1	2	
	3	4	
	5	6	
	7	8	
	9	10	
	11	12	
	13	14	
PCS2	15	16	SCK
PCS3	17	18	MOSI
PCS1	19	20	MISO
PCS0	21	22	TPU1
TPU2	23	24	TPU3
TPU4	25	26	TPU5
TPU6	27	28	TPU7
TPU8	29	30	TPU9
TPU10	31	32	TPU11
TPU12	33	34	TPU13
TPU14	35	36	
TPU15	37	38	
IRQ5	39	40	IRQ7
IRQ2	41	42	MODCLK

Note that the CF2 highlights every 5th pin with a white silkscreen square, to help you find the pins more easily.

PR – read port

Displays the current logic value and leaves the port configured as an input.

PC – set port low

Sets the port as an output low and leaves the port configured thus.

PS – set port high

Sets the port as an output high and leaves the port configured thus.

PM – mirror port

Inputs the logic level of the port and then drives the port at the same logic level.

PT – toggle port

Drives the port at the opposite of its current value. Hold the <enter> key down to continuously toggle the port.

Modem Commands

XR XS YR YS modem send or receive files

Blah blah

Appendix A – File Limits

- 1 The maximum file size on any FAT file system is the lesser of 4GB or the limit imposed by the volume size.
- 2 The maximum volume size for any FAT16 file system is 2GB, so the maximum file size will be somewhat less.
- 3 The maximum file size for PicoDOS using a memory card formatted as FAT32 is also 2GB because PicoDOS uses signed 32-bit integers for certain file operations.
- 4 The maximum volume size for a CF1/CF2 4.xx FAT32 formatted volume is limited to 128GB (137,438,953,472 actual bytes (or 137GB where manufacturer uses $*1,000,000,000 = 1\text{GB}$))
- 4a PicoDOS has only been tested to 60GB.
- 5 If the card is formatted as FAT16, then the root directory is limited to 512 files but you can override the 512 limit with the /D option.
- 6 If the card is formatted as FAT32, then the root directory is treated like a subdirectory without an artificial limit on the number of files.
- 7 If the card is formatted with PicoDOS 2.x then the card will be formatted as FAT16 (and 2GB is the limit for a FAT16 volume).
- 8 If the card is formatted with PicoDOS 3.x or 4.x, and if the card is 2GB or less then the card will be formatted as FAT16.
- 9 If the card is formatted with PicoDOS 3.x or 4.x, and if the card is greater than 2GB then the card will be formatted as FAT32.
- 10 Regardless of the FAT format, there is no artificial limit on the number of files within a directory, and a directory counts as a file when it is in the root directory.
- 10a However, the number of files in either FAT format is limited to (bytes free / cluster size) because a file cannot be smaller than a cluster. The cluster size is found by typing `chkdsk /b` in PicoDOS. $\text{BPB_BytsPerSec} * \text{BPB_SecPerClus} = \text{cluster size}$. For instance, a 2GB card formatted as FAT16 has a cluster size of $(64 * 512) = 32\text{KB}$ and so the 2GB card cannot have more than 64,030 files (including directories).
- 11 Beware that changing the format from FAT16 to FAT32 will increase all the file operation times because the FAT tables and directory file pages are now larger and the CF2 has to look through them to find start of file, end of file and the connecting data. See also the "FAT32Considerations" note on your PC located here;

C:\Program Files\Persistor\MotoCross Support\CFX\Docs\Release Notes\FAT32Considerations.htm