CF Storage card and CF-Monitor (Memory Mode Addressing)

Connect the PIC to the serial, power it up, and after some seconds of a blinking led you'll see:

```
CF-mon v 0.1


> _
```

CF-Monitor now is ready to accept commands. Commands used in Memory Mode operation (v0.1) are:

```
 - pek  Read a byte from PIC ram.           Format "pek fff"
 - pok  Write a byte to PIC ram.            Format "pok fff ff"
 - amr  CF, Attibute Memory Read.           Format "amr fff"
 - crw  CF, Configuration Register Write.   Format "crw fff ff"
 - cmr  CF, Common Memory Read.             Format "cmr fff"
 - cmw  CF, Common Memory Write.            Format "cmw fff ff"
 - rst  CF, Soft Reset.                     Format "rst"
 - cis  CF, Dump CIS registers.             Format "cis"
 - dup  CF, Dump current ATA sector.        Format "dup"
```

CF cards have an special registers managed by commands *AMR* and *CRW* (signal REG=0), for our purposes only *AMR* will be used. In CF storage cards, CIS registers are placed in even addresses in the range 0x000..0x1FF of these special registers. To access these CIS registers, called tuples, just write the command *CIS*. You must find what's mean these bytes, i.e. follow this link: http://www.linkclub.or.jp/~clover/cis/

```
> cis
000 |01 03 D9 01 FF 1C 04 03 D9 01 FF 18 02 DF 01 20 |.............. |
010 |04 01 4E 00 02 15 2B 04 01 4B 4F 44 41 4B 20 50 |..N...+..KODAK P|
020 |69 63 74 75 72 65 20 43 61 72 64 20 20 20 20 20 |icture Card     |
030 |20 20 00 4B 4F 44 41 4B 20 20 00 56 34 58 30 31 |  .KODAK  .V4X01|
040 |00 FF 21 02 04 01 22 02 01 01 22 03 02 04 5F 1A |..!..."..."..._.|
050 |05 01 03 00 02 0F 1B 0B C0 40 A1 27 55 4D 5D 75 |.........@.'UM]u|
060 |08 00 21 1B 06 00 01 21 B5 1E 4D 1B 0D C1 41 99 |..!....!..M...A.|
070 |27 55 4D 5D 75 64 F0 FF FF 21 1B 06 01 01 21 B5 |'UM]ud...!....!.|
080 |01 03 D9 01 FF 1C 04 03 D9 01 FF 18 02 DF 01 20 |.............. |
090 |04 01 4E 00 02 15 2B 04 01 4B 4F 44 41 4B 20 50 |..N...+..KODAK P|
0A0 |69 63 74 75 72 65 20 43 61 72 64 20 20 20 20 20 |icture Card     |
0B0 |20 20 00 4B 4F 44 41 4B 20 20 00 56 34 58 30 31 |  .KODAK  .V4X01|
0C0 |00 FF 21 02 04 01 22 02 01 01 22 03 02 04 5F 1A |..!..."..."..._.|
0D0 |05 01 03 00 02 0F 1B 0B C0 40 A1 27 55 4D 5D 75 |.........@.'UM]u|
0E0 |08 00 21 1B 06 00 01 21 B5 1E 4D 1B 0D C1 41 99 |..!....!..M...A.|
0F0 |27 55 4D 5D 75 64 F0 FF FF 21 1B 06 01 01 21 B5 |'UM]ud...!....!.|

200 OK
>
```

Now we know that we have a "Kodak Picture Card" attached to PIC, in fact it's a 16MB one :). CF cards can be addressed using 3 different methods, we'll use the so called "Memory mode". To see this we have more 'special registers', specially at address 0x200.

0x200 Configuration Option Register

```
> amr 200
00
200 OK
>
```

```
    B7.- SRESET.   To make a soft reset, set it, wait reset time, and clear it.
                   It's equivalent to a hard reset.
    B6.- LevIREQ.  Interrupt mode. 1=Level 0=Pulse.
    B5..B0.- Conf. Operation Mode.
                   B5 B4 B3 B2 B1 B0  Mode
                   0  0  0  0  0  0   Memory Mapped
                   0  0  0  0  0  1   IO Mapped. Any 16bit boundary.
                   0  0  0  0  1  0   IO Mapped. 0x1F0..0x1F7/0x3F6..0x3F7
                   0  0  0  0  1  1   IO Mapped. 0x170..0x177/0x376..0x377
```

Once we have checked that CF card is working in "Memory Mapped Mode", the CF Sotrage Card Registers (signal REG=1) are accessed in a 2K window. We will manage them with the commands *CMR* & *CMW*, here's a brief table:

```
      Address   Read      Write
      ----------------------
      001       Error  |  Features
      002         Sector Count
      003         Sector number
      004         Cylinder Low
      005         Cylinder High
      006         LBA/Drive/Head
      007       Status  | Command
      008         Even byte data
      009         Odd byte data

   0x001. Error Register (read only).
      B7.- BBK. Bad Block.
      B6.- UNC. Uncorrectable error.
      B5.- 0
      B4.- IDNF. Sector ID cannot be found
      B3.- 0
      B2.- Abort. Command aborted.
      B1.- 0
      B0.- AMNF. General error.

   0x003. Sector Number
          Also LBA<7:0>

   0x004. Cylinder Low
          Also LBA<15:8>

   0x005. Cylinder High
          Also LBA<23:16>

   0x006. LBA/Drive/Head
      B7.- 1
      B6.- LBA. 1=LBA mode, 0=CHS mode.
      B5.- 1
      B4.- DRV. Drive, for our purposes 0
      B3..B0. Head. Also LBA<27:24>
```

```
   0x007. Status (read only).
      B7.- BUSY. Set by CF card while it's working
      B6.- RDY.  Ready to accept commands, cleared at powerup.
      B5.- DWF.  Write fault.
      B4.- DSC.  CF is ready
      B3.- DRQ.  Data Request.
      B2.- CORR. Correctable error.
      B1.- 0
      B0.- ERR.  Error in previous command, see ERROR register.
```

Now we're capable to send ATA commands to a CF Storage card. See 6.2.1 section of CF specs for a detailed description of them.

Checking Power ATA drive:

```
> cmw 007 98
200 OK
> cmr 001
00
200 OK
> cmr 007
50
200 OK
>
```

After the "Check Power Mode" command (0x98), we'll see that we've no errors (*cmr 001*) and ATA drive is ready to accept commands (*cmr 007*).

Identify Device :

```
> cmw 006 e0
200 OK
> cmw 007 ec
200 OK
> cmr 001
00
200 OK
> cmr 007
58
200 OK
```

This command fill the data buffer with the drive information. STATUS reports that it's busy, we must get the buffer data. We'll clear the BUSY bit by reading the data with the *DUP* command. The DUP command store the full sector (512 bytes) in PIC RAM range 0x200..0x3FF and then is sended to serial:

```
> dup
000 |8A 84 FB 00 00 00 04 00 00 40 00 02 20 00 00 00 |.........@.. ...|
010 |80 7D 00 00 30 30 30 30 30 30 30 30 30 39 30 30 |.}..000000000900|
020 |30 30 33 31 36 32 39 37 02 00 02 00 04 00 2E 56 |00316297.......V|
030 |58 34 31 30 20 20 4F 4B 41 44 20 4B 54 41 5F 41 |X410  OKAD KTA_A|
040 |4C 46 53 41 20 48 20 20 20 20 20 20 20 20 20 20 |LFSA H          |
050 |20 20 20 20 20 20 20 20 20 20 20 20 20 20 01 00 |              ..|
060 |00 00 00 02 00 00 00 03 00 00 01 00 FB 00 04 00 |................|
070 |20 00 80 7D 00 00 00 01 80 7D 00 00 00 00 00 00 | ..}.....}......|
080 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
090 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
0A0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
```

```
0B0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
0C0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
0D0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
0E0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
0F0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
100 |00 00 45 42 48 52 4E 41 20 55 4D 49 4E 41 20 20 |..EBHRNA UMINA  |
110 |20 20 49 48 4F 52 54 20 48 41 52 41 20 41 20 20 |  IHORT HARA A  |
120 |4D 20 20 2E 53 41 41 4E 53 41 41 48 49 52 20 20 |M  .SAANSAAHIR  |
130 |20 20 4D 20 4B 49 20 45 53 41 41 53 20 52 1B 8B |  M KI ESAAS R..|
140 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
150 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
160 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
170 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
180 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
190 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
1A0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
1B0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
1C0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
1D0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
1E0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
1F0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|

200 OK
> cmr 007
50
200 OK
>
```

The meaning of all these bytes is in section 6.2.1.6 of CF specs. By now we'll identify some of them:

Offset	Value	Meaning
(000) 000	848A	Default (by now), Note the LSByte order in dump.
(002) 001	00FB	Cylinders (251)
(006) 003	0004	Heads (4)
(00C) 006	0020	Sectors/Track (32)
(00E) 007	00007D80	Total Sectors (32128). 32128x512=16MB.

Getting Partition Table :

```
> cmw 002 01
200 OK
> cmw 003 01
200 OK
> cmw 004 00
200 OK
> cmw 005 00
200 OK
> cmw 006 A0
200 OK
> cmw 007 20
200 OK
```

The partition table is in Drive=0, Head=0, Cyl=0, Sect=1. We'll get the hole sector by invoking the *read sector* ATA command (0x20) and dumping the sector. This sector will end with the magic bytes '55AA':

```
> dup
000 |FA BE 00 7C BF 00 7A B9 00 01 FC 0E 1F 0E 07 F3 |...|..z.........|
010 |A5 EA 16 7A 00 00 BB BE 7B 33 C9 80 3F 80 75 06 |...z....{3..?.u.|
020 |FE C5 8B F3 EB 07 80 3F 00 75 02 FE C1 83 C3 10 |.......?.u......|
030 |81 FB FE 7B 72 E5 83 F9 04 74 0B 81 F9 03 01 74 |...{r....t.....t|
040 |0A BB A6 7A EB 2C BB 87 7A EB 27 8B 4C 02 8B 14 |...z.,..z.'.L...|
050 |B8 01 02 BB 00 7C CD 13 73 05 BB BE 7A EB 13 2E |.....|..s...z...|
060 |A1 FE 7D 3D 55 AA 74 05 BB BE 7A EB 05 EA 00 7C |..}=U.t...z....||
070 |00 00 2E 8A 07 3C 00 74 0C 53 BB 07 00 B4 0E CD |.....<.t.S......|
080 |10 5B 43 EB ED EB FE 4E 6F 20 62 6F 6F 74 61 62 |.[C....No bootab|
090 |6C 65 20 70 61 72 74 69 74 69 6F 6E 20 69 6E 20 |le partition in |
0A0 |74 61 62 6C 65 00 49 6E 76 61 6C 69 64 20 50 61 |table.Invalid Pa|
0B0 |72 74 69 74 69 6F 6E 20 74 61 62 6C 65 00 49 6E |rtition table.In|
0C0 |76 61 6C 69 64 20 6F 72 20 64 61 6D 61 67 65 64 |valid or damaged|
0D0 |20 42 6F 6F 74 61 62 6C 65 20 70 61 72 74 69 74 | Bootable partit|
0E0 |69 6F 6E 00 53 79 73 74 65 6D 53 6F 66 74 20 42 |ion.SystemSoft B|
0F0 |6F 6F 74 20 50 61 72 74 69 74 69 6F 6E 20 53 65 |oot Partition Se|
100 |63 74 6F 72 00 00 00 00 00 00 00 00 00 00 00 00 |ctor............|
110 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
120 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
130 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
140 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
150 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
160 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
170 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
180 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
190 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
1A0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
1B0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
1C0 |01 01 04 00 E0 EB 20 00 00 00 60 7D 00 00 00 00 |...... ...`}....|
1D0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
1E0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
1F0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 55 AA |..............U.|

200 OK
>
```

The partition table have 4 entries of 16 bytes each one, starting at offset 0x1BE:

```
Entry 0 :  00 00 01 01 04 00 E0 EB 20 00 00 00 60 7D 00 00
Entry 1 :  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Entry 2 :  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Entry 3 :  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Each entry have the following structure (i.e. Entry 0):

```
 Offset   Value  Meaning
 00       00     0x00=Not bootable, 0x80=Bootable. Also Drive=0
 01       00     Start Head (0)
 02       0101   Start Cyl/Sect INT13H format: (c7..c0) (c9,c8,sect5..0)
                 Cyl=1 Sect=1
 04       04     SO type. (FAT16<32MB)
 05       00     End Head (1)
 06       EBE0   End Cyl/Sect INT13H format: (c7..c0) (c9,c8,sect5..0)
                 Cyl=1003 Sect=32
 08       0000
          0020   Start sector in LBA Mode (32)
 0C       0000
          7D60   Total sectors in partition (32096). ~16MB.
```

Getting Boot Sector of a Partition :

In LBA mode, first sector of our partition is at 0x00000020 (see offset 08 of partition table), so we'll get this sector. Also we'll check that ends with the magic bytes '55AA'.

```
> cmw 002 01
200 OK
> cmw 003 20
200 OK
> cmw 004 00
200 OK
> cmw 005 00
200 OK
> cmw 006 E0
200 OK
> cmw 007 20
200 OK
> dup
000 |EB 3C 90 6D 6B 64 6F 73 66 73 00 00 02 04 01 00 |.<.mkdosfs......|
010 |02 00 02 60 7D F8 20 00 20 00 01 00 00 00 00 00 |...`}. . .......|
020 |00 00 00 00 00 00 29 B4 3A 19 41 20 20 20 20 20 |......).:.A     |
030 |20 20 20 20 20 20 46 41 54 31 36 20 20 20 0E 1F |      FAT16   ..|
040 |BE 5B 7C AC 22 C0 74 0B 56 B4 0E BB 07 00 CD 10 |.[|.".t.V.......|
050 |5E EB F0 32 E4 CD 16 CD 19 EB FE 54 68 69 73 20 |^..2.......This |
060 |69 73 20 6E 6F 74 20 61 20 62 6F 6F 74 61 62 6C |is not a bootabl|
070 |65 20 64 69 73 6B 2E 20 20 50 6C 65 61 73 65 20 |e disk.  Please |
080 |69 6E 73 65 72 74 20 61 20 62 6F 6F 74 61 62 6C |insert a bootabl|
090 |65 20 66 6C 6F 70 70 79 20 61 6E 64 0D 0A 70 72 |e floppy and..pr|
0A0 |65 73 73 20 61 6E 79 20 6B 65 79 20 74 6F 20 74 |ess any key to t|
0B0 |72 79 20 61 67 61 69 6E 20 2E 2E 2E 20 0D 0A 00 |ry again ... ...|
0C0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
0D0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
0E0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
0F0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
100 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
110 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
120 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
130 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
140 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
150 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
160 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
170 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
180 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
190 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
1A0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
1B0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
1C0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
1D0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
1E0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
1F0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 55 AA |..............U.|

200 OK
>
```

Usefull data we can find in Boot Sector.

Offset	Value	Meaning
0B	0200	Bytes/Sector (512)
0D	04	Sectors/Cluster
0E	0001	Reserved Sectors
10	02	FAT copies
11	0200	Root Directory Entries (512)
13	7D60	Total sectors (32096)
15	F8	DOS media type (HD)
16	0020	Sectors/FAT (32)
18	0020	Sectors/Cyl (32)
1A	0001	Heads
1C	0000	Hidden sectors (0)

From current data in Boot Sector, 1rst FAT starts at 0x00000021 (Boot Sector + 1).

```
> cmw 003 21
200 OK
> cmw 007 20
200 OK
> dup
000 |F8 FF FF FF FF FF 04 00 05 00 06 00 07 00 08 00 |................|
010 |09 00 0A 00 0B 00 0C 00 0D 00 FF FF 0F 00 10 00 |................|
020 |FF FF 12 00 13 00 14 00 15 00 16 00 17 00 18 00 |................|
030 |19 00 1A 00 1B 00 1C 00 FF FF FF FF 00 00 00 00 |................|
040 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
050 |29 00 2A 00 2B 00 2C 00 2D 00 2E 00 2F 00 30 00 |).*.+.,.-.../.0.|
060 |31 00 32 00 33 00 34 00 35 00 36 00 FF FF 38 00 |1.2.3.4.5.6...8.|
070 |FF FF 3A 00 3B 00 3C 00 3D 00 3E 00 3F 00 40 00 |..:.;.<.=.>.?.@.|
080 |41 00 42 00 43 00 44 00 45 00 46 00 47 00 48 00 |A.B.C.D.E.F.G.H.|
090 |FF FF 4A 00 4B 00 4C 00 4D 00 4E 00 4F 00 50 00 |..J.K.L.M.N.O.P.|
0A0 |51 00 52 00 53 00 54 00 55 00 56 00 57 00 58 00 |Q.R.S.T.U.V.W.X.|
0B0 |59 00 5A 00 5B 00 5C 00 5D 00 5E 00 5F 00 FF FF |Y.Z.[.\.].^._...|
0C0 |61 00 62 00 63 00 64 00 65 00 66 00 67 00 68 00 |a.b.c.d.e.f.g.h.|
0D0 |FF FF 6A 00 6B 00 6C 00 6D 00 6E 00 6F 00 70 00 |..j.k.l.m.n.o.p.|
0E0 |71 00 72 00 73 00 74 00 75 00 76 00 FF FF 00 00 |q.r.s.t.u.v.....|
0F0 |00 00 7A 00 7B 00 7C 00 7D 00 7E 00 7F 00 80 00 |..z.{.|.}.~.....|
100 |81 00 82 00 83 00 84 00 85 00 86 00 FF FF 88 00 |................|
110 |89 00 8A 00 8B 00 8C 00 8D 00 8E 00 8F 00 90 00 |................|
120 |91 00 92 00 93 00 94 00 95 00 96 00 97 00 98 00 |................|
130 |99 00 9A 00 9B 00 9C 00 9D 00 9E 00 9F 00 A0 00 |................|
140 |A1 00 A2 00 A3 00 A4 00 A5 00 A6 00 A7 00 A8 00 |................|
150 |A9 00 AA 00 AB 00 AC 00 AD 00 AE 00 AF 00 B0 00 |................|
160 |B1 00 B2 00 B3 00 B4 00 B5 00 B6 00 B7 00 B8 00 |................|
170 |B9 00 BA 00 BB 00 BC 00 BD 00 BE 00 FF FF C0 00 |................|
180 |FF FF C2 00 C3 00 C4 00 C5 00 C6 00 C7 00 C8 00 |................|
190 |C9 00 CA 00 CB 00 CC 00 CD 00 CE 00 CF 00 D0 00 |................|
1A0 |D1 00 D2 00 D3 00 FF FF D5 00 D6 00 D7 00 D8 00 |................|
1B0 |D9 00 DA 00 DB 00 DC 00 FF FF FF FF 00 00 00 00 |................|
1C0 |00 00 00 00 00 00 00 00 E5 00 E6 00 E7 00 E8 00 |................|
1D0 |E9 00 FF FF 00 00 00 00 00 00 00 00 EF 00 F0 00 |................|
1E0 |F1 00 FF FF 00 00 00 00 F5 00 FF FF 00 00 00 00 |................|
1F0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|

200 OK
>
```

Note that FAT fills 32 sectors. Here's only the first one.

2nd FAT (a copy) is at 0x00000021+20 (1rst FAT + FAT size)

```
> cmw 003 41
200 OK
> cmw 007 20
200 OK
> dup
000 |F8 FF FF FF FF FF 04 00 05 00 06 00 07 00 08 00 |................|
010 |09 00 0A 00 0B 00 0C 00 0D 00 FF FF 0F 00 10 00 |................|
020 |FF FF 12 00 13 00 14 00 15 00 16 00 17 00 18 00 |................|
030 |19 00 1A 00 1B 00 1C 00 FF FF FF FF 00 00 00 00 |................|
040 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
050 |29 00 2A 00 2B 00 2C 00 2D 00 2E 00 2F 00 30 00 |).*.+.,.-.../.0.|
060 |31 00 32 00 33 00 34 00 35 00 36 00 FF FF 38 00 |1.2.3.4.5.6...8.|
070 |FF FF 3A 00 3B 00 3C 00 3D 00 3E 00 3F 00 40 00 |..:.;.<.=.>.?.@.|
080 |41 00 42 00 43 00 44 00 45 00 46 00 47 00 48 00 |A.B.C.D.E.F.G.H.|
090 |FF FF 4A 00 4B 00 4C 00 4D 00 4E 00 4F 00 50 00 |..J.K.L.M.N.O.P.|
0A0 |51 00 52 00 53 00 54 00 55 00 56 00 57 00 58 00 |Q.R.S.T.U.V.W.X.|
0B0 |59 00 5A 00 5B 00 5C 00 5D 00 5E 00 5F 00 FF FF |Y.Z.[.\.].^._...|
0C0 |61 00 62 00 63 00 64 00 65 00 66 00 67 00 68 00 |a.b.c.d.e.f.g.h.|
0D0 |FF FF 6A 00 6B 00 6C 00 6D 00 6E 00 6F 00 70 00 |..j.k.l.m.n.o.p.|
0E0 |71 00 72 00 73 00 74 00 75 00 76 00 FF FF 00 00 |q.r.s.t.u.v.....|
0F0 |00 00 7A 00 7B 00 7C 00 7D 00 7E 00 7F 00 80 00 |..z.{.|.}.~.....|
100 |81 00 82 00 83 00 84 00 85 00 86 00 FF FF 88 00 |................|
110 |89 00 8A 00 8B 00 8C 00 8D 00 8E 00 8F 00 90 00 |................|
120 |91 00 92 00 93 00 94 00 95 00 96 00 97 00 98 00 |................|
130 |99 00 9A 00 9B 00 9C 00 9D 00 9E 00 9F 00 A0 00 |................|
140 |A1 00 A2 00 A3 00 A4 00 A5 00 A6 00 A7 00 A8 00 |................|
150 |A9 00 AA 00 AB 00 AC 00 AD 00 AE 00 AF 00 B0 00 |................|
160 |B1 00 B2 00 B3 00 B4 00 B5 00 B6 00 B7 00 B8 00 |................|
170 |B9 00 BA 00 BB 00 BC 00 BD 00 BE 00 FF FF C0 00 |................|
180 |FF FF C2 00 C3 00 C4 00 C5 00 C6 00 C7 00 C8 00 |................|
190 |C9 00 CA 00 CB 00 CC 00 CD 00 CE 00 CF 00 D0 00 |................|
1A0 |D1 00 D2 00 D3 00 FF FF D5 00 D6 00 D7 00 D8 00 |................|
1B0 |D9 00 DA 00 DB 00 DC 00 FF FF FF FF 00 00 00 00 |................|
1C0 |00 00 00 00 00 00 00 00 E5 00 E6 00 E7 00 E8 00 |................|
1D0 |E9 00 FF FF 00 00 00 00 00 00 00 00 EF 00 F0 00 |................|
1E0 |F1 00 FF FF 00 00 00 00 F5 00 FF FF 00 00 00 00 |................|
1F0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|

200 OK
>
```

... and Root Dir Entry at 0x00000041+20 (2nd FAT + FAT size). Root Dir. fills 32 sectors (512 entries x 32bytes/each), here's only first and second one (using consecutive *DUP*s):

```
> cmw 003 61
200 OK
> cmw 007 20
200 OK
> dup
000 |41 61 00 62 00 6F 00 75 00 74 00 0F 00 2C 2E 00 |Aa.b.o.u.t...,..|
010 |68 00 74 00 6D 00 6C 00 00 00 00 00 FF FF FF FF |h.t.m.l.........|
020 |41 42 4F 55 54 7E 31 20 48 54 4D 20 00 00 63 AA |ABOUT~1 HTM ..c.|
030 |0A 31 0A 31 00 00 7B 86 E9 30 02 00 33 07 00 00 |.1.1..{..0..3...|
040 |E5 68 00 61 00 72 00 64 00 32 00 00 00 71 2E 00 |.h.a.r.d.2...q..|
050 |68 00 74 00 6D 00 6C 00 7E 00 00 00 00 00 FF FF |h.t.m.l.~.......|
060 |E5 41 52 44 32 7E 32 20 48 54 4D 20 00 00 9B 80 |.ARD2~2 HTM ....|
070 |0B 31 0B 31 00 00 4C 98 EC 30 DE 00 32 2B 00 00 |.1.1..L..0..2+..|
080 |41 64 00 6F 00 63 00 73 00 2E 00 0F 00 B3 68 00 |Ad.o.c.s......h.|
```

```
090 |74 00 6D 00 6C 00 00 00 FF FF 00 00 FF FF FF FF |t.m.l...........|
0A0 |44 4F 43 53 7E 31 20 20 48 54 4D 20 00 00 63 AA |DOCS~1  HTM ..c.|
0B0 |0A 31 0A 31 00 00 24 8C EE 30 0E 00 AA 14 00 00 |.1.1..$..0......|
0C0 |E5 68 00 61 00 72 00 64 00 33 00 00 00 6D 2E 00 |.h.a.r.d.3...m..|
0D0 |68 00 74 00 6D 00 6C 00 7E 00 00 00 00 00 FF FF |h.t.m.l.~.......|
0E0 |E5 41 52 44 33 7E 32 20 48 54 4D 20 00 00 AF 80 |.ARD3~2 HTM ....|
0F0 |0B 31 0B 31 00 00 D3 9B ED 30 EA 00 AD 1E 00 00 |.1.1.....0......|
100 |41 68 00 61 00 72 00 64 00 2E 00 0F 00 97 68 00 |Ah.a.r.d......h.|
110 |74 00 6D 00 6C 00 00 00 FF FF 00 00 FF FF FF FF |t.m.l...........|
120 |48 41 52 44 7E 31 20 20 48 54 4D 20 00 00 63 AA |HARD~1  HTM ..c.|
130 |0A 31 0A 31 00 00 A0 9B ED 30 1D 00 C6 07 00 00 |.1.1.....0......|
140 |41 68 00 61 00 72 00 64 00 32 00 0F 00 61 2E 00 |Ah.a.r.d.2...a..|
150 |68 00 74 00 6D 00 6C 00 00 00 00 00 FF FF FF FF |h.t.m.l.........|
160 |48 41 52 44 32 7E 31 20 48 54 4D 20 00 00 9B 80 |HARD2~1 HTM ....|
170 |0B 31 0A 31 00 00 9B 80 0B 31 E4 00 47 2B 00 00 |.1.1.....1..G+..|
180 |41 68 00 61 00 72 00 64 00 33 00 0F 00 5D 2E 00 |Ah.a.r.d.3...]..|
190 |68 00 74 00 6D 00 6C 00 00 00 00 00 FF FF FF FF |h.t.m.l.........|
1A0 |48 41 52 44 33 7E 31 20 48 54 4D 20 00 00 AF 80 |HARD3~1 HTM ....|
1B0 |0B 31 0A 31 00 00 AF 80 0B 31 EE 00 BB 1E 00 00 |.1.1.....1......|
1C0 |E5 70 00 69 00 63 00 74 00 2E 00 00 00 79 68 00 |.p.i.c.t.....yh.|
1D0 |74 00 6D 00 6C 00 7E 00 00 00 00 00 FF FF FF FF |t.m.l.~.........|
1E0 |E5 49 43 54 7E 32 20 20 48 54 4D 20 00 00 D0 80 |.ICT~2  HTM ....|
1F0 |0B 31 0B 31 00 00 D0 80 0B 31 00 00 00 00 00 00 |.1.1.....1......|

200 OK
> dup
000 |E5 4E 5F 42 52 45 7E 31 4A 50 47 20 00 00 63 AA |.N_BRE~1JPG ..c.|
010 |0A 31 0A 31 00 00 46 92 EC 30 28 00 55 75 00 00 |.1.1..F..0(.Uu..|
020 |41 69 00 6E 00 64 00 65 00 78 00 0F 00 33 2E 00 |Ai.n.d.e.x...3..|
030 |68 00 74 00 6D 00 6C 00 00 00 00 00 FF FF FF FF |h.t.m.l.........|
040 |49 4E 44 45 58 7E 31 20 48 54 4D 20 00 00 63 AA |INDEX~1 HTM ..c.|
050 |0A 31 0A 31 00 00 00 43 F8 30 37 00 8D 0B 00 00 |.1.1...C.07.....|
060 |E5 6A 00 70 00 67 00 00 00 FF FF 00 00 92 FF FF |.j.p.g..........|
070 |FF FF FF FF FF FF FF FF FF FF 00 00 FF FF FF FF |................|
080 |E5 69 00 73 00 61 00 5F 00 70 00 00 00 92 69 00 |.i.s.a._.p....i.|
090 |6E 00 2D 00 68 00 65 00 61 00 00 00 64 00 2E 00 |n.-.h.e.a...d...|
0A0 |E5 53 41 5F 50 49 7E 31 4A 50 47 20 00 00 63 AA |.SA_PI~1JPG ..c.|
0B0 |0A 31 0A 31 00 00 4A 92 EC 30 39 00 F6 7C 00 00 |.1.1..J..09..|..|
0C0 |E5 6A 00 61 00 6C 00 5F 00 68 00 00 00 00 74 00 |.j.a.l._.h....t.|
0D0 |74 00 70 00 2E 00 6A 00 70 00 00 00 67 00 00 00 |t.p...j.p...g...|
0E0 |E5 41 4C 5F 48 54 54 50 4A 50 47 20 00 00 63 AA |.AL_HTTPJPG ..c.|
0F0 |0A 31 0A 31 00 00 7D 86 E9 30 49 00 C7 B3 00 00 |.1.1..}..0I.....|
100 |42 6C 00 00 00 FF FF FF FF FF FF 0F 00 17 FF FF |Bl..............|
110 |FF FF FF FF FF FF FF FF FF FF 00 00 FF FF FF FF |................|
120 |01 6A 00 65 00 5F 00 74 00 68 00 0F 00 17 65 00 |.j.e._.t.h....e.|
130 |6F 00 72 00 79 00 2E 00 68 00 00 00 74 00 6D 00 |o.r.y...h...t.m.|
140 |4A 45 5F 54 48 45 7E 31 48 54 4D 20 00 00 64 AA |JE_THE~1HTM ..d.|
150 |0A 31 0A 31 00 00 2A 4A EA 30 60 00 65 45 00 00 |.1.1..*J.0`.eE..|
160 |E5 2E 00 70 00 6E 00 67 00 00 00 00 00 45 FF FF |...p.n.g.....E..|
170 |FF FF FF FF FF FF FF FF FF FF 00 00 FF FF FF FF |................|
180 |E5 70 00 69 00 63 00 6E 00 69 00 00 00 45 63 00 |.p.i.c.n.i...Ec.|
190 |62 00 72 00 6F 00 77 00 73 00 00 00 65 00 72 00 |b.r.o.w.s...e.r.|
1A0 |E5 49 43 4E 49 43 7E 31 50 4E 47 20 00 00 64 AA |.ICNIC~1PNG ..d.|
1B0 |0A 31 0A 31 00 00 76 86 E9 30 69 00 71 6A 00 00 |.1.1..v..0i.qj..|
1C0 |41 70 00 69 00 63 00 74 00 2E 00 0F 00 B1 68 00 |Ap.i.c.t......h.|
1D0 |74 00 6D 00 6C 00 00 00 FF FF 00 00 FF FF FF FF |t.m.l...........|
1E0 |50 49 43 54 7E 31 20 20 48 54 4D 20 00 00 D0 80 |PICT~1  HTM ....|
1F0 |0B 31 0A 31 00 00 D0 80 0B 31 F4 00 3C 0A 00 00 |.1.1.....1..<...|

200 OK
>
```

Each directory entry consist of 32 bytes, here's an example:

```
44 4F 43 53 7E 31 20 20 48 54 4D 20 00 00 63 AA |DOCS~1  HTM ..c.|
0A 31 0A 31 00 00 24 8C EE 30 0E 00 AA 14 00 00 |.1.1..$..0......|
```

Offset	Size	Value	Meaning
00	08	DOCS~1	File Name
08	03	HTM	File ext
0B	01	20	Attributes (file)
0C	10		Reserved
16	02	8C24	Time
18	02	30EE	Date
1A	02	000E	Start Cluster (15)
1C	04	000014AA	Size (5290)

File_name and File_ext should be filled with 0x20s to complete the size. If filename starts with 0xE5, the entry is erased.

Attibutes:

```
B5.- File
B4.- Subdir
B3.- Volume name
B2.- System
B1.- Hidden
B0.- Read only
```

Time, packed in a word using the following formula:

```
Time = ( Hour * 2048 ) + ( Min * 32 ) + ( Sec / 2 )
```

Date, packed in a word using the following formula:

```
Date = ( (Year-1980) * 512 ) + ( Month * 32 ) + Day
```

Start Cluster is the entry point in the FAT table, and the data stored in FAT points to next cluster. To get the sector(s) of a given FAT cluster:

(Cluster- 2) *(sec/cluster)+(2ndFAT+FATsize)=start sector

(0x0E- 0x02)*4+81=0xB1

```
> cmw 003 b1
200 OK
> cmw 007 20
200 OK
> dup
000 |3C 21 44 4F 43 54 59 50 45 20 48 54 4D 4C 20 50 |<!DOCTYPE HTML P|
010 |55 42 4C 49 43 20 22 2D 2F 2F 57 33 43 2F 2F 44 |UBLIC "-//W3C//D|
020 |54 44 20 48 54 4D 4C 20 33 2E 32 2F 2F 45 4E 22 |TD HTML 3.2//EN"|
030 |3E 0A 3C 48 54 4D 4C 3E 0A 3C 48 45 41 44 3E 0A |>.<HTML>.<HEAD>.|
040 |09 3C 4D 45 54 41 20 48 54 54 50 2D 45 51 55 49 |.<META HTTP-EQUI|
050 |56 3D 22 43 4F 4E 54 45 4E 54 2D 54 59 50 45 22 |V="CONTENT-TYPE"|
060 |20 43 4F 4E 54 45 4E 54 3D 22 74 65 78 74 2F 68 | CONTENT="text/h|
070 |74 6D 6C 3B 20 63 68 61 72 73 65 74 3D 69 73 6F |tml; charset=iso|
080 |2D 38 38 35 39 2D 31 22 3E 0A 09 3C 54 49 54 4C |-8859-1">..<TITL|
090 |45 3E 4A 41 4C 20 45 74 68 65 72 6E 65 74 20 2D |E>JAL Ethernet -|
0A0 |20 44 6F 63 75 6D 65 6E 74 73 20 3C 2F 54 49 54 | Documents </TIT|
0B0 |4C 45 3E 0A 09 3C 4D 45 54 41 20 4E 41 4D 45 3D |LE>..<META NAME=|
```

```
0C0 |22 47 45 4E 45 52 41 54 4F 52 22 20 43 4F 4E 54 |"GENERATOR" CONT|
0D0 |45 4E 54 3D 22 4F 70 65 6E 4F 66 66 69 63 65 2E |ENT="OpenOffice.|
0E0 |6F 72 67 22 3E 0A 09 3C 4D 45 54 41 20 4E 41 4D |org">..<META NAM|
0F0 |45 3D 22 43 52 45 41 54 45 44 22 20 43 4F 4E 54 |E="CREATED" CONT|
100 |45 4E 54 3D 22 32 30 30 34 30 37 30 38 3B 31 37 |ENT="20040708;17|
110 |32 31 32 32 30 30 22 3E 0A 09 3C 4D 45 54 41 20 |212200">..<META |
120 |4E 41 4D 45 3D 22 43 48 41 4E 47 45 44 22 20 43 |NAME="CHANGED" C|
130 |4F 4E 54 45 4E 54 3D 22 32 30 30 34 30 37 30 38 |ONTENT="20040708|
140 |3B 31 38 33 31 35 39 30 30 22 3E 0A 3C 2F 48 45 |;18315900">.</HE|
150 |41 44 3E 0A 3C 42 4F 44 59 3E 0A 3C 54 41 42 4C |AD>.<BODY>.<TABL|
160 |45 20 42 4F 52 44 45 52 3D 30 20 43 45 4C 4C 50 |E BORDER=0 CELLP|
170 |41 44 44 49 4E 47 3D 34 20 43 45 4C 4C 53 50 41 |ADDING=4 CELLSPA|
180 |43 49 4E 47 3D 30 3E 0A 09 09 3C 54 52 3E 0A 09 |CING=0>...<TR>..|
190 |09 09 3C 54 48 20 57 49 44 54 48 3D 31 36 35 20 |..<TH WIDTH=165 |
1A0 |48 45 49 47 48 54 3D 34 35 3E 0A 09 09 09 09 3C |HEIGHT=45>.....<|
1B0 |50 3E 3C 41 20 48 52 45 46 3D 22 68 74 74 70 3A |P><A HREF="http:|
1C0 |2F 2F 73 6F 75 72 63 65 66 6F 72 67 65 2E 6E 65 |//sourceforge.ne|
1D0 |74 2F 70 72 6F 6A 65 63 74 73 2F 6A 61 6C 65 74 |t/projects/jalet|
1E0 |68 65 72 6E 65 74 22 3E 3C 55 3E 3C 46 4F 4E 54 |hernet"><U><FONT|
1F0 |20 53 49 5A 45 3D 35 3E 3C 42 3E 4A 41 4C 45 54 | SIZE=5><B>JALET|

200 OK
>
```

A subdirectory entry:

```
49 4D 41 47 45 53 20 20 20 20 20 10 00 00 DD 80 |IMAGES     .....|
0B 31 0B 31 00 00 DD 80 0B 31 DD 00 00 00 00 00 |.1.1.....1......|
```

The offset 0x0B have the subdir bit enabled (0x10, instead of file 0x20). Searching the start sector in the same way (note the first 2 entries):

```
> cmw 003 ed
200 OK
> cmw 004 03
200 OK
> cmw 007 20
200 OK
> dup
000 |2E 20 20 20 20 20 20 20 20 20 20 10 00 00 3C 80 |.          ...<.|
010 |0B 31 0B 31 00 00 3C 80 0B 31 DD 00 00 00 00 00 |.1.1..<..1......|
020 |2E 2E 20 20 20 20 20 20 20 20 20 10 00 00 3C 80 |..         ...<.|
030 |0B 31 0B 31 00 00 3C 80 0B 31 00 00 00 00 00 00 |.1.1..<..1......|
040 |41 63 00 6F 00 6D 00 70 00 5F 00 0F 00 3B 73 00 |Ac.o.m.p._...;s.|
050 |69 00 64 00 65 00 2E 00 6A 00 00 00 70 00 67 00 |i.d.e...j...p.g.|
060 |43 4F 4D 50 5F 53 7E 31 4A 50 47 20 00 00 63 AA |COMP_S~1JPG ..c.|
070 |0A 31 0B 31 00 00 48 92 EC 30 03 00 39 53 00 00 |.1.1..H..0..9S..|
080 |41 65 00 61 00 73 00 79 00 77 00 0F 00 E9 62 00 |Ae.a.s.y.w....b.|
090 |2E 00 6A 00 70 00 67 00 00 00 00 00 FF FF FF FF |..j.p.g.........|
0A0 |45 41 53 59 57 42 20 20 4A 50 47 20 00 00 63 AA |EASYWB  JPG ..c.|
0B0 |0A 31 0B 31 00 00 66 93 ED 30 11 00 56 59 00 00 |.1.1..f..0..VY..|
0C0 |42 2E 00 6A 00 70 00 67 00 00 00 0F 00 48 FF FF |B..j.p.g.....H..|
0D0 |FF FF FF FF FF FF FF FF FF FF 00 00 FF FF FF FF |................|
0E0 |01 69 00 6E 00 5F 00 62 00 72 00 0F 00 48 65 00 |.i.n._.b.r...He.|
0F0 |61 00 64 00 62 00 6F 00 61 00 00 00 72 00 64 00 |a.d.b.o.a...r.d.|
100 |49 4E 5F 42 52 45 7E 31 4A 50 47 20 00 00 63 AA |IN_BRE~1JPG ..c.|
110 |0A 31 0B 31 00 00 46 92 EC 30 28 00 55 75 00 00 |.1.1..F..0(.Uu..|
...
1E0 |41 70 00 69 00 63 00 79 00 6E 00 0F 00 41 69 00 |Ap.i.c.y.n...Ai.|
1F0 |63 00 2E 00 6A 00 70 00 67 00 00 00 00 00 FF FF |c...j.p.g.......|

200 OK
>
```

CF Storage card and CF-Monitor (IO Mode Addressing)

You can work with CompactFlash either in MEM mode or IO mode, but not both at same time. You'll work with ATA registers & commands in the same way, but using new types of addressing. To put CompactFlash in IO mode just type:

```
> crw 200 03
200 OK
> imd 3
200 OK
```

You must set Configuration Option Register with the proper value, as we seen previously. CF & CF-monitor works by default in Memory mode, so you must tell the program to work in IO mode with the instruction *IMD:*

```
imd 1          IO mode using any 16bit address.
imd 2          IO mode using 0x1F0..0x1F7
imd 3          IO mode using 0x170..0x177
mmd            Memory Mode.
```

CF-Monitor now is ready to accept commands. Commands used in IO Mode operation (v0.1) are:

```
- pek  Read a byte from PIC ram.           Format "pek fff"
- pok  Write a byte to PIC ram.            Format "pok fff ff"
- amr  CF, Attibute Memory Read.           Format "amr fff"
- crw  CF, Configuration Register Write.   Format "crw fff ff"
- rst  CF, Soft Reset.                     Format "rst"
- cis  CF, Dump CIS registers.             Format "cis"
- dup  CF, Dump current ATA sector.        Format "dup"
- ior  CF, IO Read.                        Format "ior fff"
- iow  CF, IO Write.                       Format "iow fff ff"
```

Here's an example of polling ATA drive parameters. You can check that the returned sector is the same that previously returned in Memory mode.

```
> crw 200 03
200 OK
> imd 3
200 OK
> iow 172 01
200 OK
> iow 173 01
200 OK
> iow 174 00
200 OK
> iow 175 00
200 OK
> iow 176 A0
200 OK
> iow 177 ec
200 OK
```

```
> dup
000 |8A 84 FB 00 00 00 04 00 00 40 00 02 20 00 00 00 |.........@.. ...|
010 |80 7D 00 00 30 30 30 30 30 30 30 30 30 39 30 30 |.}..000000000900|
020 |30 30 33 31 36 32 39 37 02 00 02 00 04 00 2E 56 |00316297.......V|
030 |58 34 31 30 20 20 4F 4B 41 44 20 4B 54 41 5F 41 |X410  OKAD KTA_A|
040 |4C 46 53 41 20 48 20 20 20 20 20 20 20 20 20 20 |LFSA H          |
050 |20 20 20 20 20 20 20 20 20 20 20 20 20 20 01 00 |              ..|
060 |00 00 00 02 00 00 00 03 00 00 01 00 FB 00 04 00 |................|
070 |20 00 80 7D 00 00 00 01 80 7D 00 00 00 00 00 00 | ..}.....}......|
080 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
090 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
0A0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
0B0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
0C0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
0D0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
0E0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
0F0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
100 |00 00 45 42 48 52 4E 41 20 55 4D 49 4E 41 20 20 |..EBHRNA UMINA  |
110 |20 20 49 48 4F 52 54 20 48 41 52 41 20 41 20 20 |  IHORT HARA A  |
120 |4D 20 20 2E 53 41 41 4E 53 41 41 48 49 52 20 20 |M  .SAANSAAHIR  |
130 |20 20 4D 20 4B 49 20 45 53 41 41 53 20 52 1B 8B |  M KI ESAAS R..|
140 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
150 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
160 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
170 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
180 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
190 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
1A0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
1B0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
1C0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
1D0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
1E0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|
1F0 |00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |................|

200 OK
>
```

That's all folks!

Javier Martínez
2004- August- 12