CP/M 2.2 COMPACT FLASH NOTES Alan Paton

From results found during development in 2016

1. No need for a separate format program.

The CF card only requires an E5H byte at every location, because CP/M will allocate all required data fields. CP/M will read E5H as available space.

Use a PC with a Hex Editor program to fill complete CF card with E5H. Then install CP/M system on track zero.

2. Use 8K allocation blocks - this will save RAM space when building allocation tables and make access faster when reading or writing large files and data.

For an 8MB disk approx. 256 bytes are required for each disk when using 4K allocation blocks compared with 128 bytes with 8K blocks.

With 16 disks on the compact flash card this means that 4096 bytes have to be reserved in RAM if using 4K blocks but only 2048 for 8K blocks. (a saving of 2K in the BIOS)

If all files were 4K or less then it would be possible to store nearly 2000 files on an 8MB disk - This reduces to just under 1000 files at 8K each.

In practice, I have found that the majority of files are over 4K and usually 2 or 3 hundred files are plenty for one disk. (as there is only one directory per disk user with CP/M 2.2)

3. To simplify and save additional space in the BIOS reserve 1 Track on all disks.

1 Track is equal to 16K bytes - which is only 2 files at 8K each.

The last disk can still be allocated less space because of the CF card limits.

4. To read or write files the following Z80 code is the fastest and most efficient - after setting up the Logical Block Addressing etc.

```
          CPORT   EQU   0058H
          RAMDAT  EQU   6000H
          CMD     EQU   5FH          ;I/O address (as required)
-------------------------------------------------------------------------
0225        ; Read data from CF drive
0225        ; The data is transferred to store address.
0225
0225 015800  RDCF:LD    BC,CPORT     ;Count=00, Port=58H
0228 210060       LD    HL,RAMDAT    ;store address
022B CD3F02       CALL  WAITBSY
022E 3E20         LD    A,20H        ;read sector
0230 D35F         OUT   (CMD),A
0232 CD4D02       CALL  WAITDRQ      ;wait for Data Request
0235 EDB2         INIR
0237 EDB2         INIR               ;read 512 data bytes (2x 256)
0239 CD3F02       CALL  WAITBSY
023C DB5F         IN    A,(CMD)      ;check final drive status
023E C9           RET                ;on errors, return with A=1
```

```
------------------------------------------------------------------------
            ;write data to CF drive
0203        ;from store address RAMDAT
0203
0203 015800  WRCF:LD    BC,CPORT    ;Count=00, Port=58H
0206 210080       LD    HL,RAMDAT   ;store address
0209 CD3F02       CALL  WAITBSY
020C 3E30         LD    A,30H       ;write sector
020E D35F         OUT   (CMD),A
0210 CD4D02       CALL  WAITDRQ     ;wait for Data Request
0213 EDB3         OTIR
0215 EDB3         OTIR              ;write 512 data bytes (2x256)
0217 CD3F02       CALL  WAITBSY
021A DB5F         IN    A,(CMD)     ;check final drive status
021C C9           RET               ;on errors, return with A=1
021D

------------------------------------------------------------------------
023F
023F        ;wait until BUSY bit = 0
023F DB5F WAITBSY:IN    A,(CMD)     ;status
0241 CB7F         BIT   7,A
0243 C8           RET   Z           ;BSY bit is clear
0244 18F9         JR    WAITBSY
0246
024D        ;wait until DRQ bit = 1
024D DB5F WAITDRQ:IN    A,(CMD)     ;status
024F CB5F         BIT   3,A
0251 28FA         JR    Z,WAITDRQ
0253 C9           RET               ;DRQ bit is set
0254        ;
```

5. It is very important to keep the wiring from Z80 CPU to Compact Flash as short as possible. This makes the difference between a CF drive that gives random errors to one that has no errors.