

# Writing to a file byte-by-byte

BASIC code:

```

10 FS=8192:FE=16384
20 OPEN 2,8,2,"JUST A FILENAME,P,W"
30 IF ST<>0 THEN GOTO 70
40 A=PEEK(FS):FS=FS+1
50 PRINT#2,CHR$(A);
60 IF FE>FS THEN GOTO 30
70 CLOSE 2

```

Assembler code:

```

file_start = $2000    ; example addresses
file_end   = $4000

        LDA #fname_end-fname
        LDX #<fname
        LDY #>fname
        JSR $FFBD      ; call SETNAM

        LDA #$02       ; file number 2
        LDX $BA        ; last used device number
        BNE .skip
        LDX #$08       ; default to device 8
.skip   LDY #$02       ; secondary address 2
        JSR $FFBA      ; call SETLFS

        JSR $FFC0      ; call OPEN
        BCS .error     ; if carry set, the file could not be opened

        ; check drive error channel here to test for
        ; FILE EXISTS error etc.

        LDX #$02       ; filename 2
        JSR $FFC9      ; call CHKOUT (file 2 now used as output)

        LDA #<file_start
        STA $AE
        LDA #>file_start
        STA $AF

        LDY #$00
.loop   JSR $FFB7      ; call READST (read status byte)
        BNE .werror    ; write error
        LDA ($AE),Y    ; get byte from memory
        JSR $FFD2      ; call CHROUT (write byte to file)
        INC $AE

```

```
        BNE .skip
        INC $AF
.skip
        LDA $AE
        CMP #<file_end
        LDA $AF
        SBC #>file_end
        BCC .loop      ; next byte
.close
        LDA #$02      ; filename 2
        JSR $FFC3     ; call CLOSE

        JSR $FFCC     ; call CLRCHN
        RTS

.error
        ; Akkumulator contains BASIC error code

        ; most likely errors:
        ; A = $05 (DEVICE NOT PRESENT)

        ... error handling for open errors ...
        JMP .close    ; even if OPEN failed, the file has to be closed
.werror
        ; for further information, the drive error channel has to be read

        ... error handling for write errors ...
        JMP .close

fname:  .TEXT "JUST A FILENAME,P,W" ; ,P,W is required to make this an
output file!
fname_end:
```

You may open more than one file if you use different file numbers and secondary addresses for them. File numbers and secondary addresses should be in the range of 2 to 14. It's usually a good idea to use the same number for both to keep confusion low.

From:

<https://codebase64.org/> - **Codebase 64 wiki**

Permanent link:

[https://codebase64.org/doku.php?id=base:writing\\_a\\_file\\_byte-by-byte](https://codebase64.org/doku.php?id=base:writing_a_file_byte-by-byte)

Last update: **2015-04-17 04:34**

