Assembler

CHIPPER oder c8asm

- org, end nicht schreiben
- beide verstehen die Standard-Syntax nach http://devernay.free.fr/hacks/chip8/C8TECH10.HTM

syntax.asm

```
; This program just demonstrates and tests all the instructions in
the assembler.
; Don't try to run it. It will probably do strange things.

; Comments start with semicolons
start:              ; labels are identifiers followed by colons
    CLS
    JP    start
    JP    #123        ; Hexadecimal numbers are preceded by #
symbols
    JP    v0, #123
    JP    v0, end
    call end
    call #203
    se V1, #AA
    se V2, v3
    sne V1, #AA
    sne V2, v3
end:
    RET
    add V1, #AA
    add V2, v3
    ld V1, #AA
    ld V2, v3
    or V2, v3
    and VA, vb
    xor VA, vb
    shr VA, vb
    shr VA
    subn VA, vb
    shl VA, vb
    shl VA
    rnd VD, #FF
    drw VE, VF, #4
    skp VE
    sknp VA
    add I, V8
    ld I, #AAA
    ld V5, DT
    ld V5, K
```

```
    ld DT, V5
    ld ST, V5
    ld F, V5
    ld B, V5
    ld [I], VA
    ld VA, [I]

; "define" can be used to define constants
define aaa #222
    jp aaa

; "define" can also be used to define aliases for registers
define bbb vd
    ld bbb, %01010101    ; Binary literals start with % symbols
    JP %101001010101
    JP x
    LD I, x

; SCHIP instructions are supported
    SCD    #4
    SCL
    SCR
    EXIT
    HIGH
    LOW
    DRW V1, V2, 0
    LD HF, V5
    LD R, V6
    LD V7, R

; Offset moves the location where output is generated
offset #280

; This is how you can define sprites:
; "db" emits raw bytes, separated by commas.
; "dw" can emit 16-bit words.

x: db #11, #22, #33, #44
y: db
    %00100100,
    %11111111,
    %01011010,
    %00111100,
    %00100100
    CLS
```

From:
https://hc-ddr.hucki.net/wiki/ - **Homecomputer DDR**

Permanent link:
https://hc-ddr.hucki.net/wiki/doku.php/homecomputer/chip8/assembler

Last update: **2021/02/22 16:22**