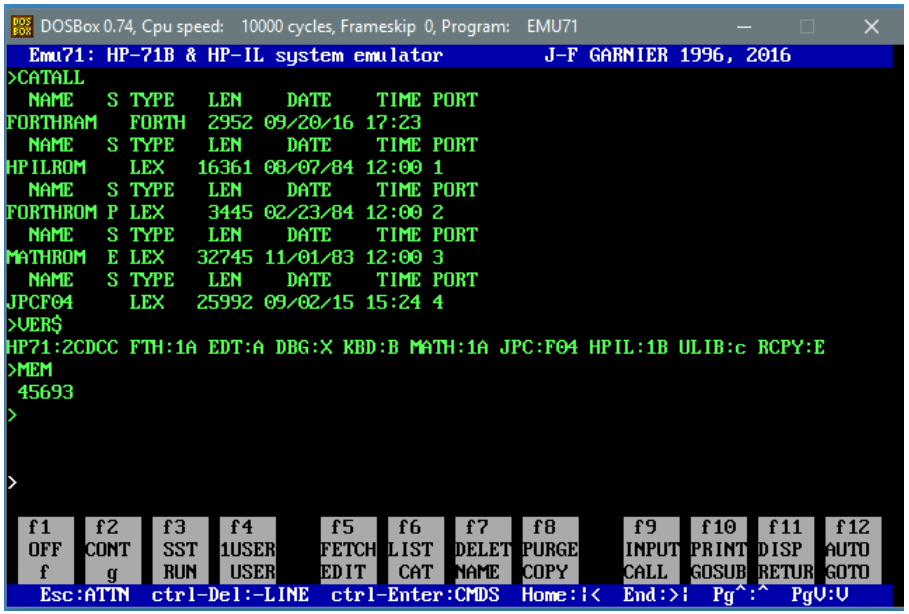
# Emu71/DOS: 20th Anniversary!


Emu71 v1.1 (1996) on a Windows-98 computer




Emu71 v2.45 (2016) in DOSBox on a Windows-10 computer

# Emu71/DOS history notes

```
; ********************************************************************
;                émulation du processeur Saturn par le 8086
;                      J-F Garnier 1993, 2004
; ------------------------------------------------------------------
; Module saturn.asm : définitions des registres, moteur d'exécution
; ------------------------------------------------------------------
; Version 0.0 16/03/93
;   prototype, faisabilité
; Version 1.0 04/11/95
; Version 2.0 07/09/01 extensions pour HP42
; Version 2.1 08/09/02 maj
; Version 3.0 03/10/02 12 bits jump table
```

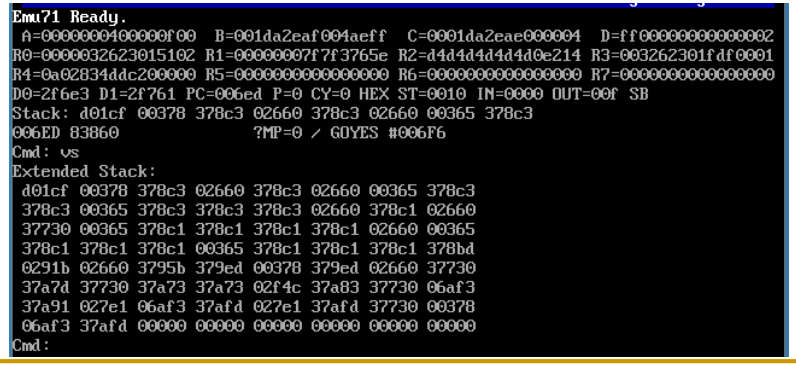
- The idea of Emu71 started in … 1993
but work really started only on end of 1995

- Motivations:
  - build a HP-71B environment with the ability to run modules that I only owned as ROMCOPY images,
  - have lot of memory (at the time my configuration was a HP-71B + a HHP 32k RAM/EPROM module – not enough!),
  - and possibly be faster that the real thing.

- First stable, usable version in **1996**: single-line display emulation, but already full ROM and RAM module emulation.
About 3x the native HP-71B speed on a 386SX at 25 MHz

# The Saturn emulation core

- Hand-coded, highly-optimized assembly code
- Heavy use of tables, macros, unrolled loops to get the highest speed. My PC of the time was a 386SX, I wasn't sure to get at least the HP-71B speed.
- Use the packed BCD format (2 nibbles per byte) and the Intel x86 native BCD support provided by the DAA/DAS opcodes, for speed (again).
- Emulation core source file (before macro expansions): ~ 200 kB
- Emulation core object code: ~ 30 kB,
- many hours (I didn't count…) of optimizing, testing, benchmarking on different machines and host CPU, …
- bonus:
extra R5-R7 registers and 64-level hardware stack:




```
; fonctions d'addition décimal

addd_byte       macro
                   mov  al,[di]
                   adc  al,[si]
                   daa
                   mov  [di],al
                   inc  si
                   inc  di
                 endm

_addd_hinib:     ;
                 mov  al,[di]
                 mov  ah,[si]
                 mov  dl,al      ; copie
                 and  ax,0f0f0h  ; masque
                 add  al,ah
                 daa
                 lahf
                 and  dl,0fh
                 or   al,dl
                 sahf
                 mov  [di],al
                 ret

_addd_16:        addd_byte
_addd_14:        addd_byte
_addd_12:        addd_byte
_addd_10:        addd_byte
_addd_8:         addd_byte
_addd_6:         addd_byte
_addd_4:         addd_byte
_addd_2:         addd_byte
                 update_carry
                 ret

_addd_15:        addd_byte
_addd_13:        addd_byte
_addd_11:        addd_byte
_addd_9:         addd_byte
_addd_7:         addd_byte
_addd_5:         addd_byte
_addd_3:         addd_byte
_addd_1:
_addd_lonib:
                 mov  al,[di]
                 mov  bh,al
                 adc  al,[si]
                 daa
                 update_halfcarry
                 mov  ah,bh
                 and  ax,0f00fh
                 or   al,ah
                 mov  [di],al
                 ret
```

# Emu71/DOS specificities…

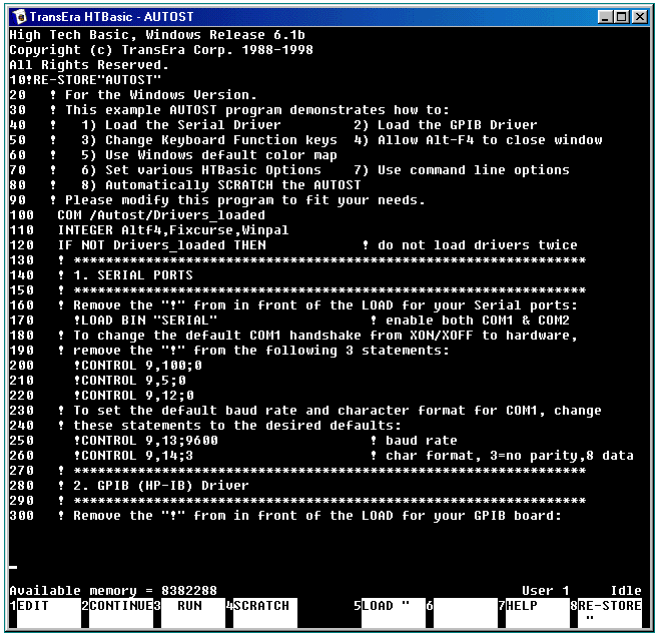
- Since speed was the top priority, some choice have been made since the beginning, especially for the I/O system emulation.

- As a result, Emu71/DOS is not an exact, accurate emulation at register level, especially for the display and keyboard management (the Diagnostic ROM fails). A benefit of this implementation is the 78-char display line instead of the 22-char LCD.

- Emu71/DOS also has some limitations on RAM modules (e.g. no support of 4 kB RAM modules, RAM ports other than port 0 not usable as main memory).

- Emu71/DOS is a text-mode, DOS-style application, and this will not change! The model of the Emu71/DOS user interface was the HP Basic aka RMB (still existing as HT Basic for Windows).

- Emu71/DOS includes a few special features:
support of the 8 ports (0-7) managed by the HP-71B OS,
support of the DEBUG command of the Forth/Assembler ROM,
support of the EEPROM module type.



HTBasic - a model for the Emu71/DOS user interface

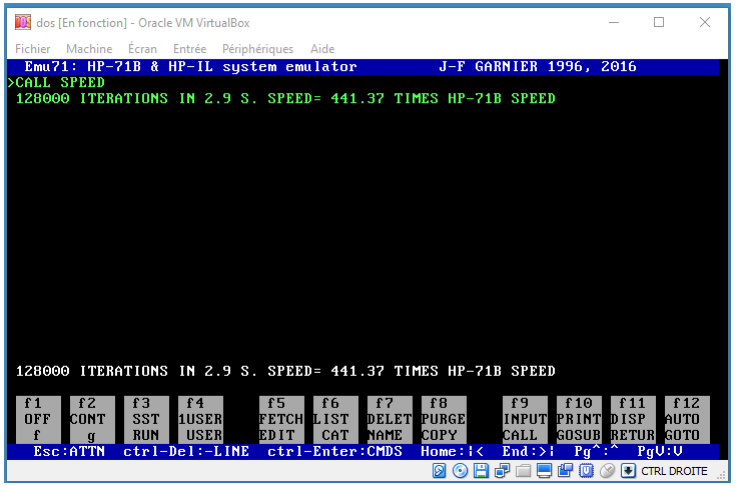
# Emu71/DOS status in 2016

- Still usable:

Despite its 16-bit DOS nature, Emu71/DOS can still be run on modern 64-bit OS. DOSBox is the simplest solution. VirtualBox gives higher performance but is more complex.

- Still powerful:

x5 the HP-71B speed with DOSBox (2 emulation layers), x400+ the HP-71B speed with VirtualBox [1].



Emu71/DOS v2.45 in VirtualBox

- The HP-IL board and the PIL-Box are no more supported natively due to the modern OS constraints. ILPer is now the natural gateway between Emu71/DOS and the real HP-71B.

- Still maintained:

The last update was rev 2.44 on dec. 2014, to support large HP-IL drives up to 16 MB.

- And for the first time, the full Emu71/DOS source files including the Saturn emulation core in assembly language are available (under GNU license).



[1] Tested on a Intel core i3 1.8GHz computer.
Benchmark: number of empty loops done in 10s ( FOR I=1 TO 1E6 @ NEXT I )

```
1 19171 6F4A                            _sto_C_D0_n:      exec_mem store C D0 n
2 19172 6F4A  BB 217Cr                                    mov   bx, offset _regD0
2 19173 6F4D  BF 2116r                                    mov   di, offset _regC
2 19174 6F50  E9 FD1D                                     jmp   exec_store_n
1 19175 6F53                            _lod_A_D1_n:      exec_mem load A D1 n
2 19176 6F53  BB 2180r                                    mov   bx, offset _regD1
2 19177 6F56  BF 2102r                                    mov   di, offset _regA
2 19178 6F59  E9 F6F6                                     jmp   exec_load_n
1 19179 6F5C                            _lod_C_D1_n:      exec_mem load C D1 n
2 19180 6F5C  BB 2180r                                    mov   bx, offset _regD1
2 19
2 19
1 19183 6F65                            _sto_A_D1_n:      exec_mem store A D1 n
2 19184 6F65  BB 2180r                                    mov   bx, offset _regD1
2 19185 6F68  BF 2102r                                    mov   di, offset _regA
2 19186 6F6B  E9 FD02                                     jmp   exec_store_n
1 19187 6F6E                            _sto_C_D1_n:      exec_mem store C D1 n
2 19188 6F6E  BB 2180r                                    mov   bx, offset _regD1
2 19189 6F71  BF 2116r                                    mov   di, offset _regC
2 19190 6F74  E9 FCF9                                     jmp   exec_store_n
  19191
  19192
  19193                                                   end
Turbo Assembler        Version 4.1        23/07/07 16:19:23       Page 338
Symbol Table
```

The Emu71/DOS page: http://www.jeffcalc.hp41.eu/emu71/index.html

The end…