VB81 XuR HELP

Welcome to the simple ZX81 world...

VB81 personal release.

Update: May , 2022

Files menu :

Specity tape directory
- Files explorer
- BASIC text format

ASM breaker.
Soft reset
Hard reset
Screen capture (screenshot)
Load Snapshot
Save Snapshot
Exit

Options menu :

Display size

Show toolbar

Show Larken/Mageco infos

Show printer window
Main window
Multi-column picture generator
Screen copy
Stylus colour

Show ouput sound

Colors
Auto-Save pictures

Settings :
Memory
Easy custom ROM installer
& ROM audit system
Graphic cards & Printers
Floppy disk interfaces
Z80 Speed overclocking
IO setting/Sound Cards
- Cursors
- IO cards
- Sound Cards

Tools menu :

Disassembler window
Advance user and exprot menu

Tiny TASM comiler window

Exernal Processor

Binary editor window

ARTIC assembler window

Dfile tools.

Reset memory to 32kb

Memory tools.

Compare memory window

BASIC tools window

BASIC variables window

Type-in window

UDG tools window

Easy draw window

Digitalize a bitmap picture

Easy Keys window

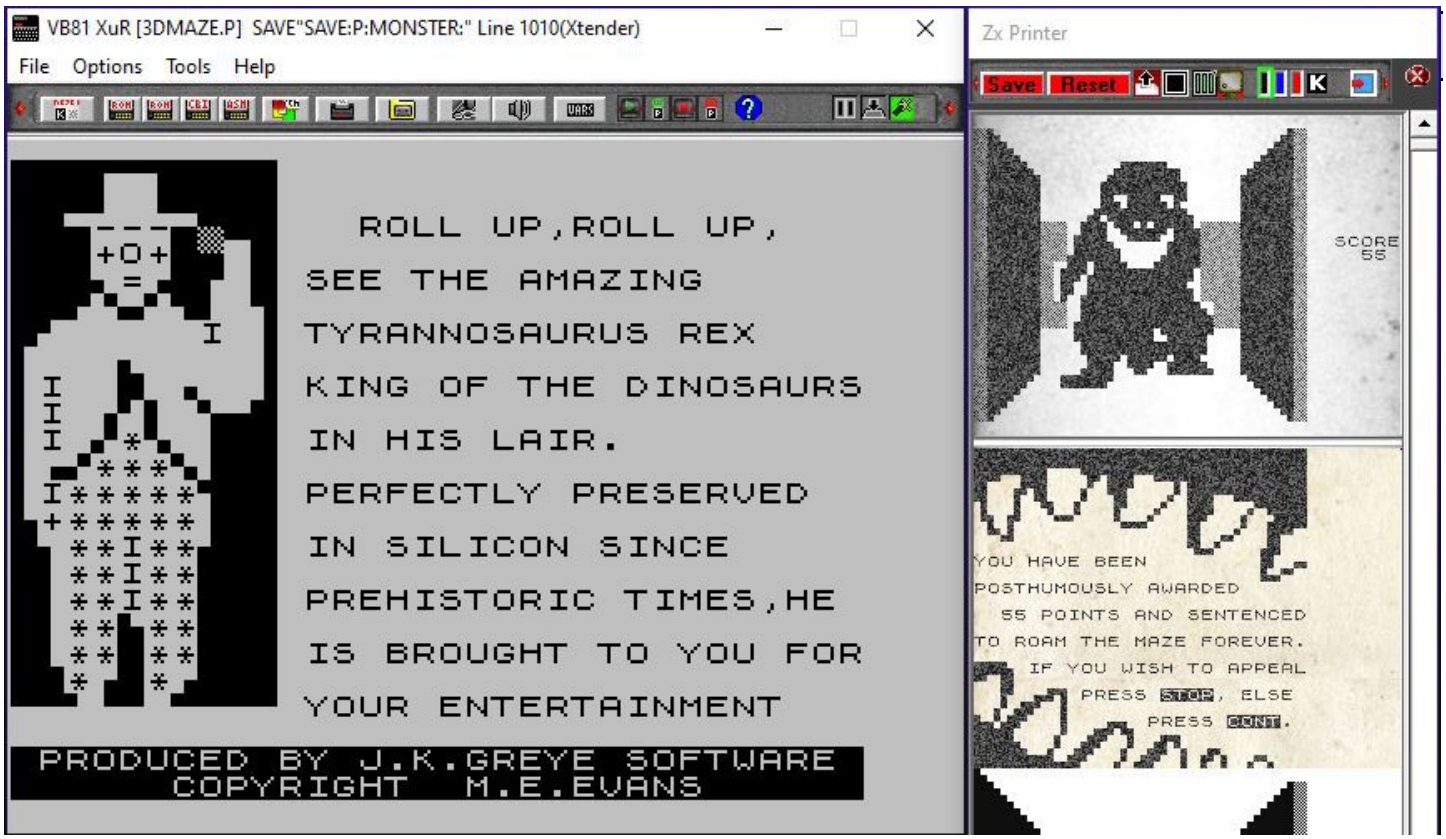
Help menu :

- Keyboard
- ZX81 ASCII table
- ASM key table
- About VB81 XuR

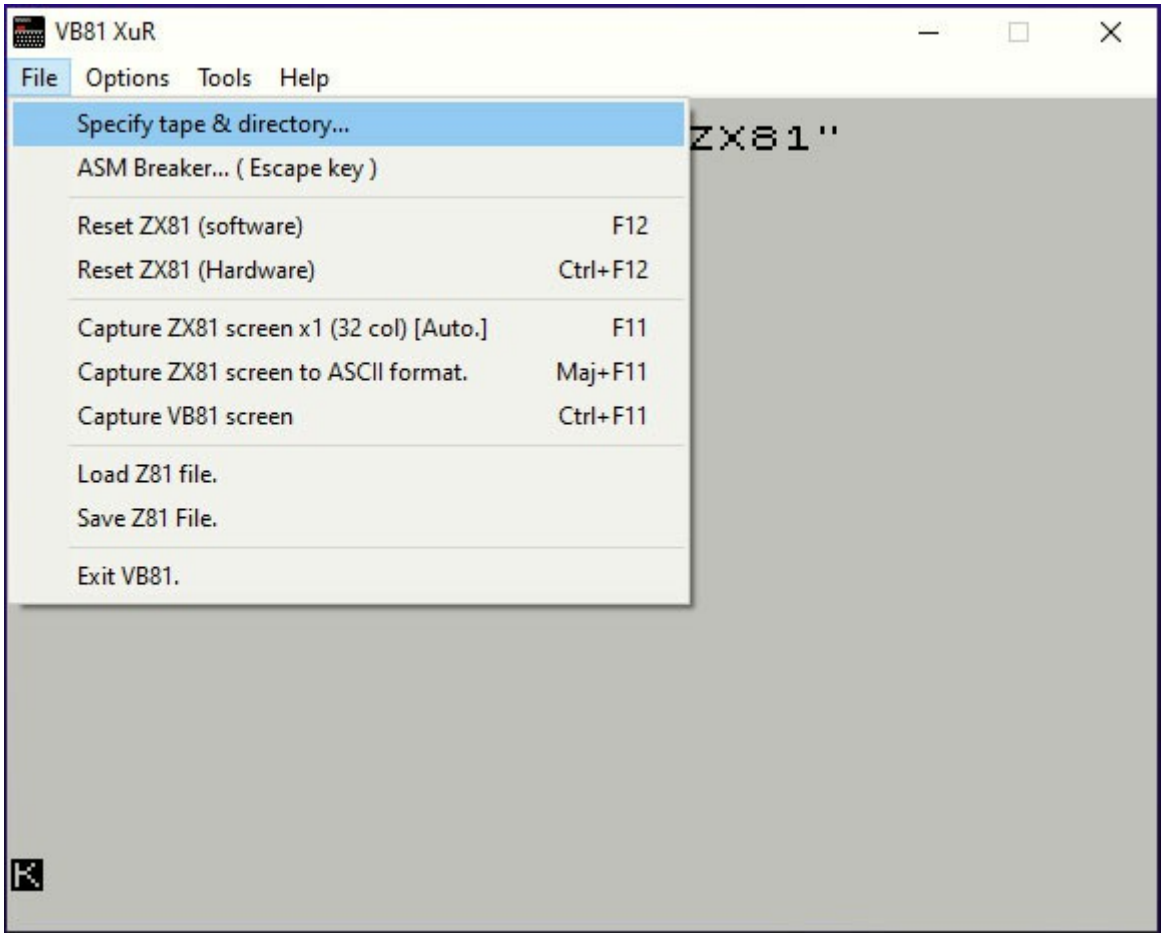
Vb81 shortcuts

ZX81 keyboard

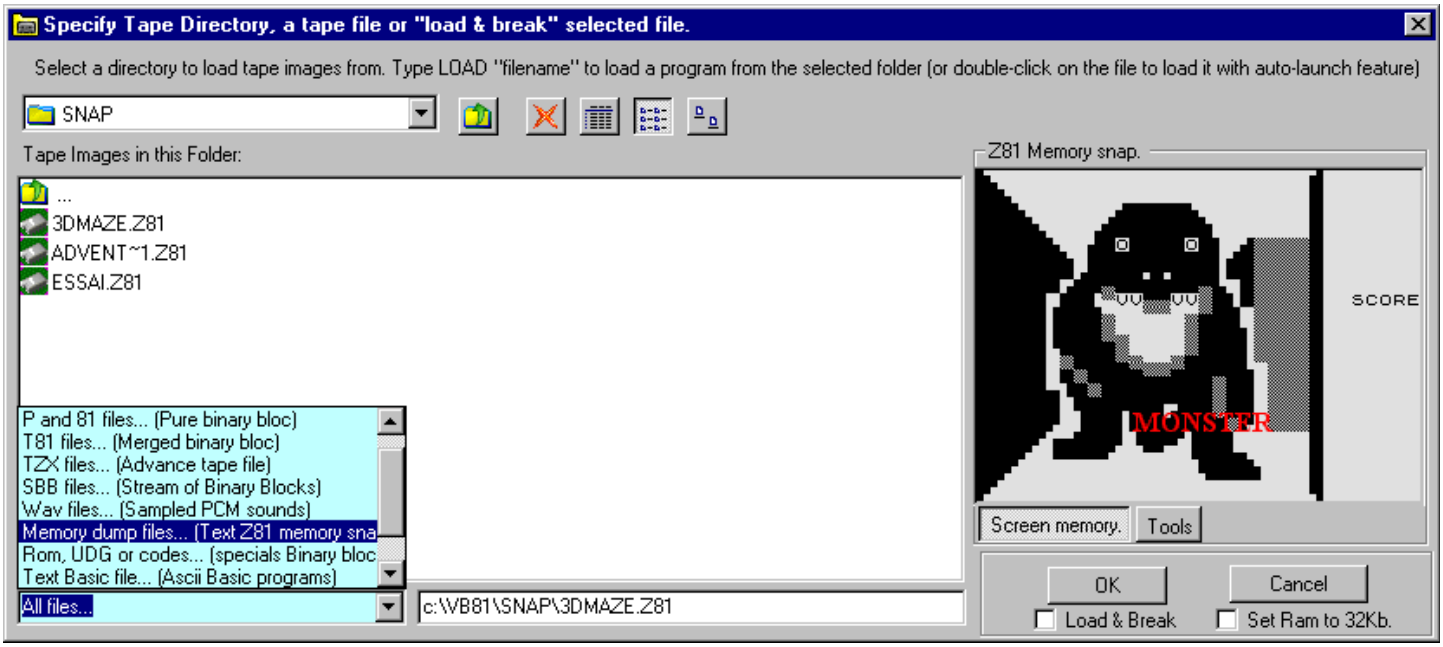
How to install VB81



3D Monster Maze - J.K. Greye Software



Specify tape directory:



Change the tape directory.
- Select a working and the default directory
- Launch a tape, load a ROM, compile a BASIC text file or launch a memory dump.

You can select a file in the list:

> a P or 81 files. (display the saved screen)
> T81,SBB , TZX digitals and virtual tape.(display all files included in the tape)
> a text BASIC file (display text, like « 10 PRINT "HELLO" »)
> Wav sound samples files.(display sample infos)
> Z81 memory dump.(display the saved screen)
> ROMs and UDG files.(display the memory map)

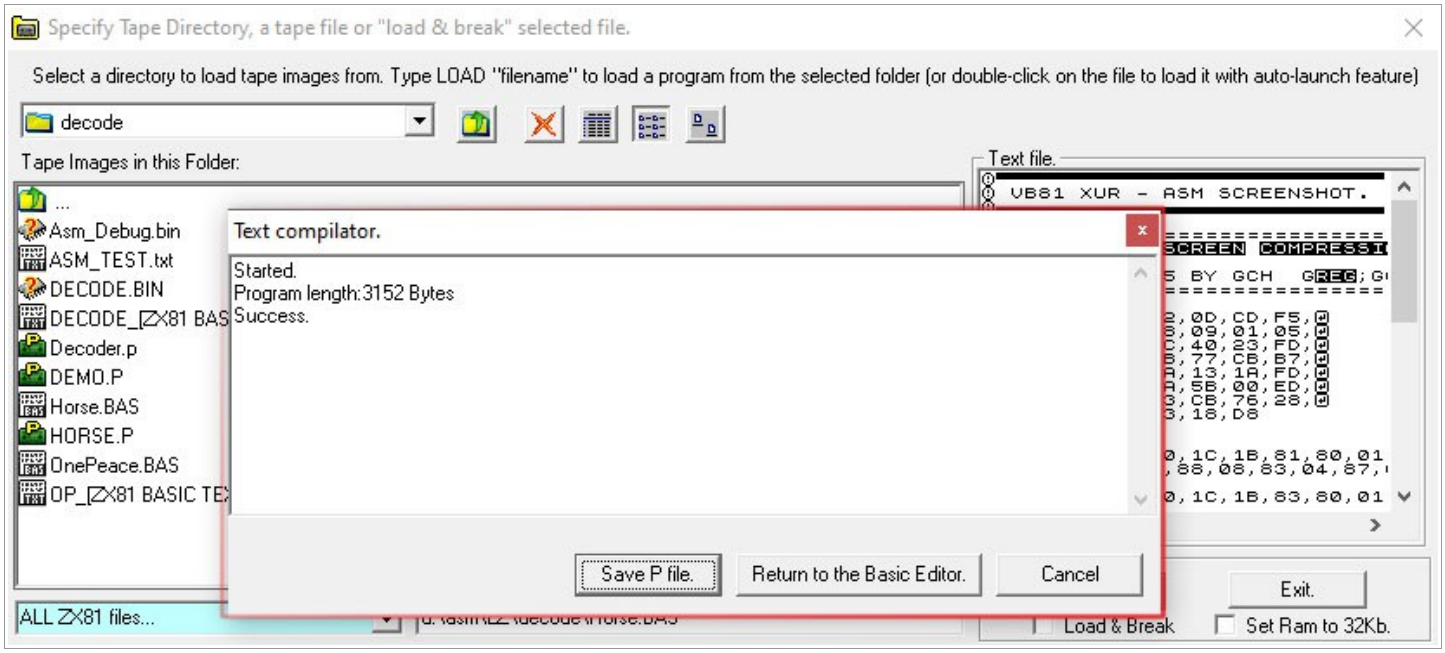
Load & break: Stop a BASIC file saved using the SAVE function in a BASIC program.

Set RAM to 32kb: If the 32 or 64 RAM pack is selected, it sets the BASIC memory to 32kb before file loading.

Tip:

One click select a file and display file infos. / A double click launch the file.

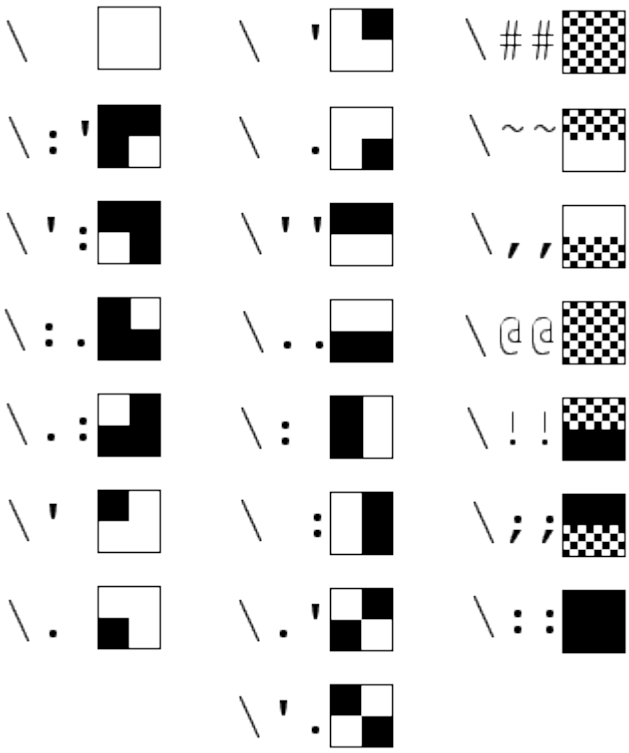
BASIC text format :



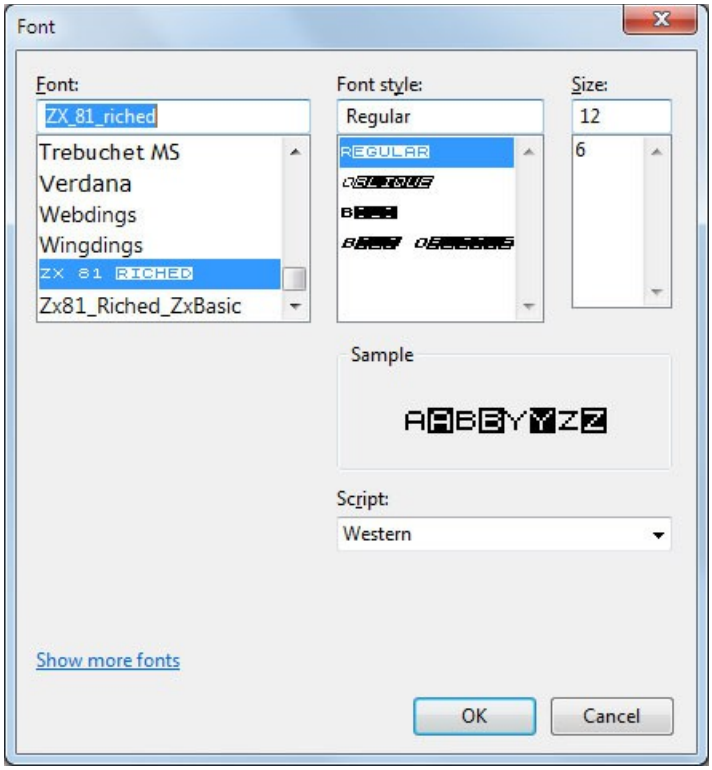
The file explorer is designed to load ZX81 text listings, to avoid to type ins your own programs.
This feature can be use to convert and feed OCR text in the ZX81 BASIC memory.

This converter is base on the « Zxtext2P » Characters codes (by Chris Cowley) format, and the
« ZX81_Riched_V3[XavSnap].ttf » Windows fonte.
Both codes can be use in the text listing.

« Zxtext2P » specials char codes:



« Riched ZX81 font » codes (ZxToken/ZxBASIC):



```
# directory.
#
# Have Fun. /XavSnap/
# ZX81.Vb81@free.fr
# ZxBasic-ZxToken-Vb81 XuR projects.
#
 0  REM DEC/HX:ZXTEXT2P:ZXBASIC
 10  REM 000=00: " "    : " " <SPACE>
 20  REM 001=01: \'      : "¶" SHIFT+1
 30  REM 002=02: \ '     : "·" SHIFT+2
 40  REM 003=03: \"      : "¸" SHIFT+7
 50  REM 004=04: \.      : "¹" SHIFT+4
 60  REM 005=05: \:      : "º" SHIFT+5
 70  REM 006=06: \.'     : "»" SHIFT+T
 80  REM 007=07: \:'     : "¼" SHIFT+E
 90  REM 008=08: \##     : "½" SHIFT+A
100  REM 009=09: \,,     : "¾" SHIFT+D
110  REM 010=0A: \~~     : "¿" SHIFT+S
120  REM 011=0B: "       : "@0B"
130  REM 012=0C: £       : "£"
140  REM 013=0D: "$"     : "$"
150  REM 014=0E: ":"     : ":"
160  REM 015=0F: "?"     : "?"
```

```
0 DIRECTORY.
0
0 HAVE FUN. /XAVSNAP/
0 ZX81.VB81@FREE.FR
0 ZXBASIC-ZXTOKEN-VB81 XUR PROJECTS.
0
  0   REM DEC/HX:ZXTEXT2P:ZXBASIC
 10   REM 000=00: " "     : " "  <SPACE>
 20   REM 001=01: ▝       :  "▗"  SHIFT+1
 30   REM 002=02: ▘       :  "▄"  SHIFT+2
 40   REM 003=03: ▀       :  "▀"  SHIFT+7
 50   REM 004=04: ▝.      :  "▖"  SHIFT+4
 60   REM 005=05: ▘:      :  "▌"  SHIFT+5
 70   REM 006=06: ▀.      :  "▙"  SHIFT+T
 80   REM 007=07: ▀:      :  "▛"  SHIFT+E
 90   REM 008=08: ▀00     :  "▒"  SHIFT+A
100   REM 009=09: ▝,,     :  "▚"  SHIFT+D
110   REM 010=0A: ▝       :  "▒"  SHIFT+S
120   REM 011=0B: "       : "@0B"
130   REM 012=0C: £       : "£"
140   REM 013=0D: "$"     : "$"
150   REM 014=0E: ":"     : ":"
160   REM 015=0F: "?"     : "?"
170   REM 016=10: "("     : "("
180   REM 017=11: ")"     : ")"
190   REM 018=12: ">"     : ">"
200   REM 019=13: "<"     : "<"
210   REM 020=14: "="     : "="
220   REM 021=15: "+"     : "+"
230   REM 022=16: "-"     : "-"
240   REM 023=17: "*"     : "*"
```

Text rules:

“#” = Ignore this line, it's a comment.

This is a comment .

“\” = In the end of the line, to cut and jump on the next line.

10 REM HELLO WORLD
equal ...
10 REM HE\
LLO \
WOR\
LD”

“@” = Is a tag to PRINT or set a special character in hexadecimal.

10 PRINT “@0C” equal 10 PRINT “£”
10 PRINT “@E3” equal 10 PRINT “STOP ”

“%” = Is a tag to invert the next character (equal minus characters.

10 PRINT “Hello@8F”equal 10 PRINT “H%E%L%L%O%?”

In a REM :

“REM [HEX: 0A,0B,0C]” = Enter the hexadecimals specified bytes.
“REM [DEC: 0,1,255]” = Enter the decimals specified bytes.
“REM [BIN: 0,101,10101010]” = Enter the binary specified bytes.

“REM [SPD: 1024]” = Enter a blank line with 1024 bytes in decimal.
“REM [SPH: 800]” = Enter a blank line with 1024 bytes in hexadecimal.

“REM [TX6: my text...£...\~~]” = Enter a text string to be code in 6bits (4 characters in 3 bytes).
Where « £ » is a New/line and « \ ~~» (chr$(10)) is the end of text line.

“REM [ASM: my TAsm Mnemonics]” = Enter a machine code text to be assemble with the “tiny assembler” feature.
Use **“|\”** at the end of each lines to set the “Enter” tag, and link the next line.

```
10 REM [ASM : \
,ORG  16514|\
LD HL,$4000 ; COMMENT |\
DEC HL |\
RET ]
```

To install and use the « ZX81_Riched_V3 font [XavSnap].ttf », have look to the « \ZX_FONTS » directory.

Asm breaker (« esc » key):

Extra ZX81's feature, stops a machine code routine when it's running.

Soft Reset :

Soft reset set the regPC to 0 (RST 0H), clear the RAM up to RAMTOP and reset the Z80 registers.

Hard Reset :

Reset the computer and all installed features... power off the computer.

Capture screen:

Default directory : “/ScreenShot” in VB81_XuR directory, or the working project direcory.

- Capture the screen display (x1) in a BMP format (1bit format).
- Capture the screen in text mode and save it in ascii format in the « VB_XuR/ScreenShot/Ascii_Screen.txt » file. All values are added in the same file.
  Example : Allow to retrieve a database listing.
- Capture the screen display as is in a BMP format (24bits format).

Load Snapshot:

Load a memory image (*.z81) in RAM.

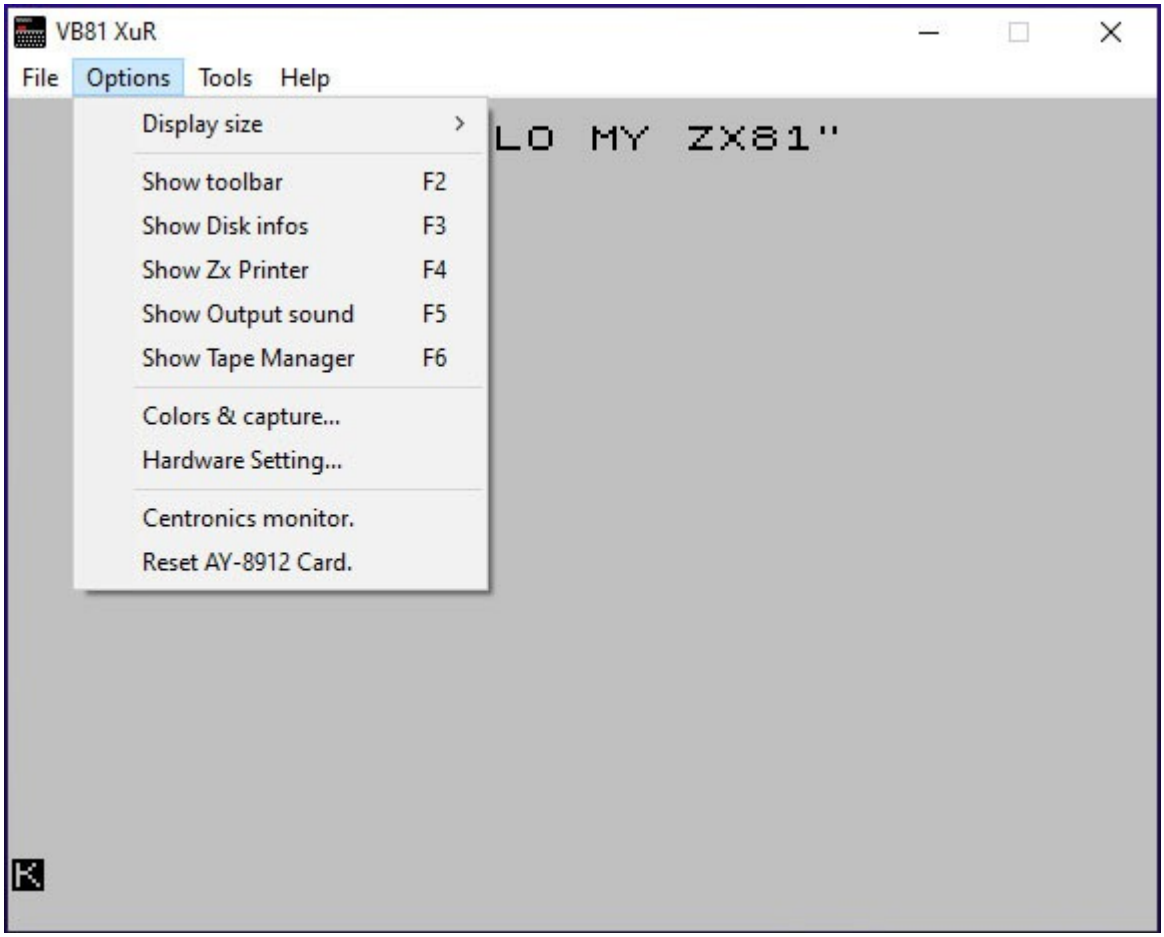
Save Snapshot:

Save a memory image to a file.

Exit:

Close all opened windows and ...Exit VB81...

Options... :



Display size:

To change emulator display size.

Show toolbar:

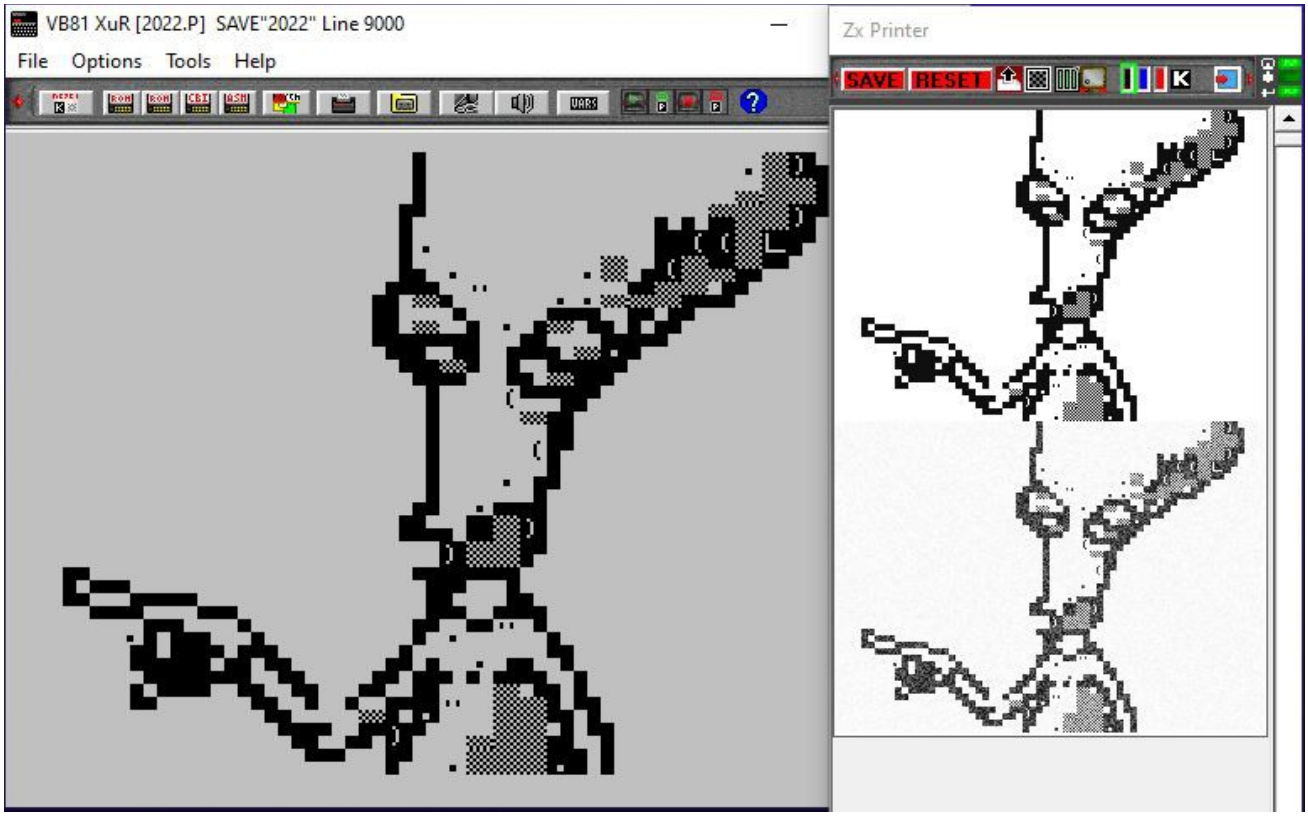
Display the main toolbar.

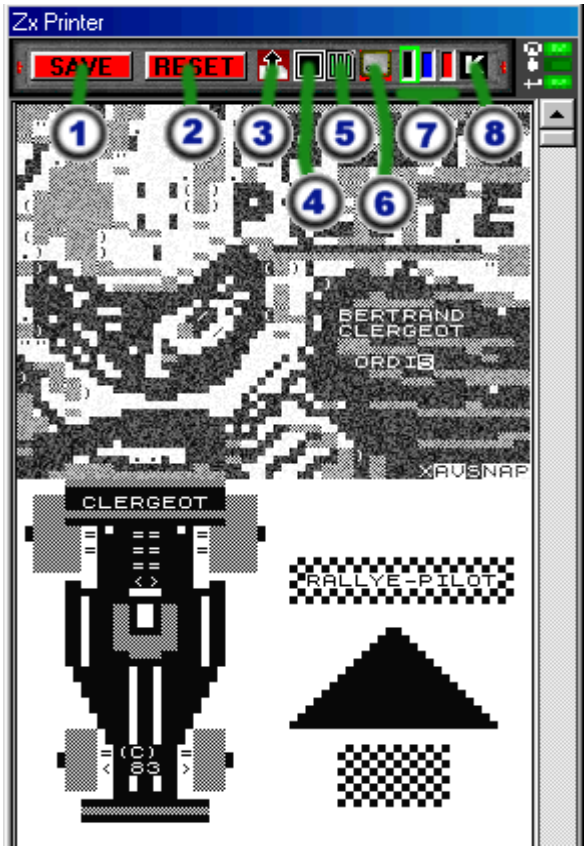
Show Larken info:

Display the Larken controller toolbar.

Show ZxPrinter:

Open the ZxPrinter emulator.



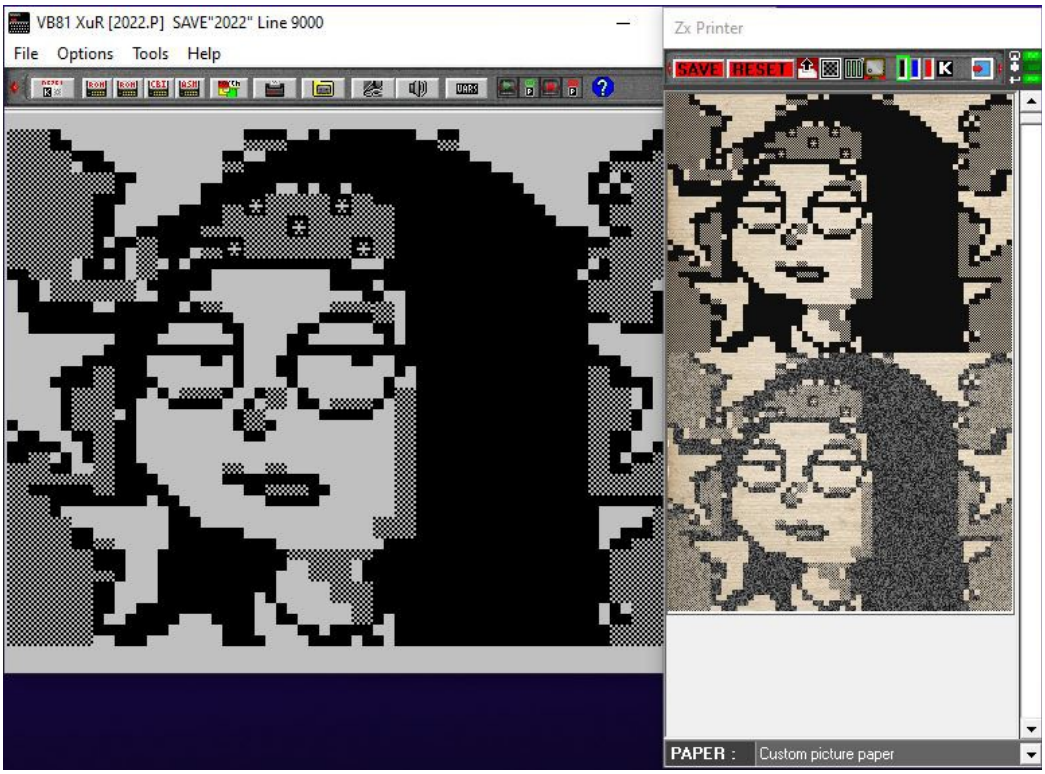


It able to :

(1) Save the paper sheet output.
(2) Reset the printer (clear the paper output).
(3) Feed paper output (add a blank line) by clicking on the left button, the right button will cut the paper.
(4) Select the printing mode:
black and white / gray (photo-realistic rendering).
(5) Past the printer output on a single page. This feature create a new page and put your listing using four columns.
Note: To improve printer quality, don't use the photo-realisme feature.
(6) Take a screen snapshot without a “Copy” function. Able to get a screen while BASIC or ASM running program.
(7) Choose the printer “ink” (colour matrix printer 3 ink rubbon... (yellow not implemented)
(8) Reverse the ploted dot color.

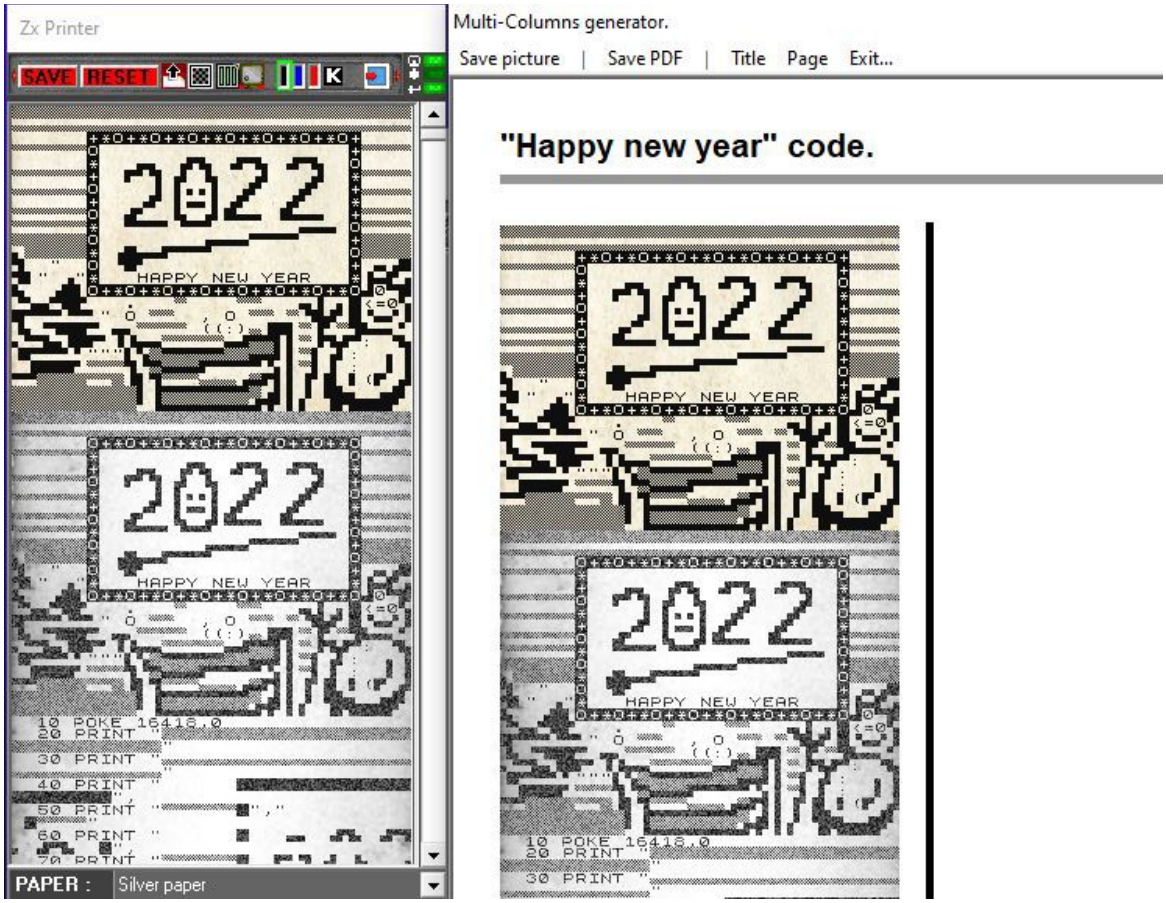
(Bottom off the window) Change the background paper texture.

- Ultra bright.
- Silver Zxprinter paper.
- User defined background (Stored in the « \PRINTER_BACKGROUND » directoy.



The green LED panel is used to close « On/Off » the ZxPrinter window...

Multi-column generator :



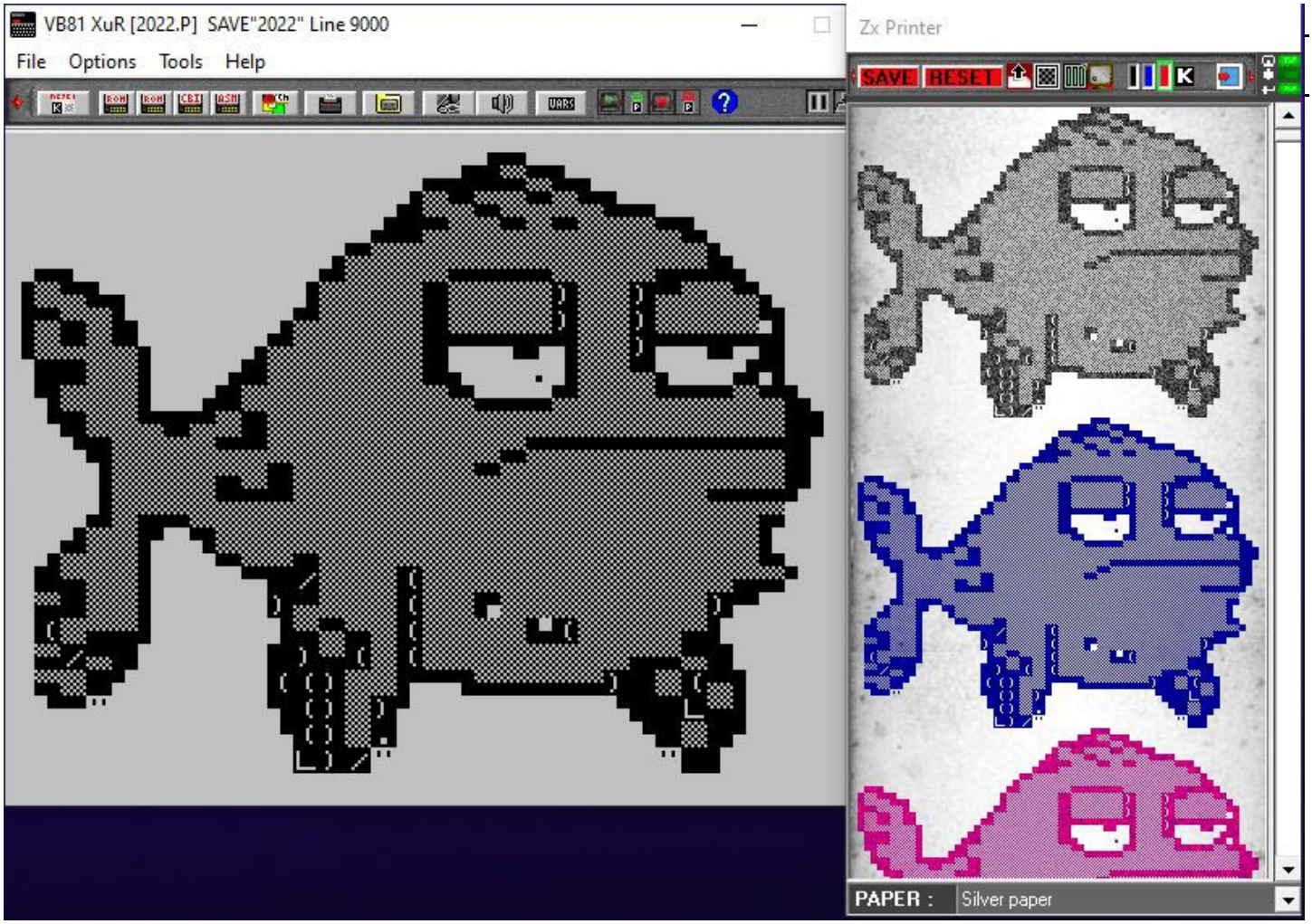
This feature able to print a ZxPrinter roll on a A4 (21x29,7 cm) page.
It able to save it in variours picture format (BMP,JPG or PNG) or in an embedded PDF format.
In case of multiple listing pages, you had to save each picture (page), but in the PDF format, all pictures will be save.

You can customize each page with a title and a different page number.
This window have a dynamic display, and will be refresh while ZxPrinter running.

Screen copy :

Just print the screen with the specified background.

Stylus color:



Set a different color to the printer (4 ink ribbon : black, cyan, blue and yellow), but the yellow was unable to see on the white paper.

Show Output sound:

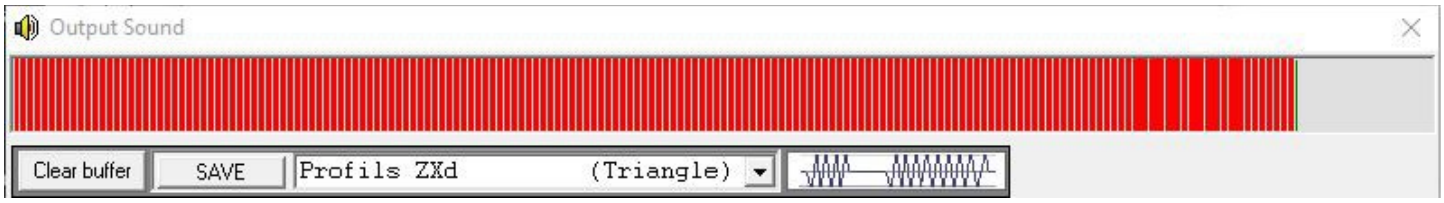
[input sound] [output sound]
LOAD SAVE

P= P files = ROM patched. (down statement)
> or O = Interrupt chached. (up statement)

In case of "Save to Wav" patched ROM, all sound output are redirect to this window. (Right Switch in the tool bar, the red one)

If this switch is on the "P" stat, all files will be saved in P files (default directory) or in the wav fast-Load mode using the BASIC function **SAVE"MYPROG.FL"**, a regular WAV forme **SAVE"MYPROG.WAV"**.

(the green switch is use to load a "wav" file in the current directory)



The sound will be buffered in memory and can be saved in a wav file with several harmonics:

1) Eighty-One profile: pure sinusoidal harmonic.

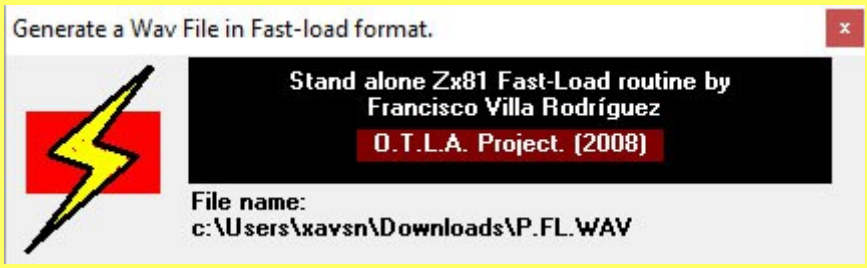
2) Zxd profile: triangle harmonic.

3) Zxtape profile: rectangle harmonic.

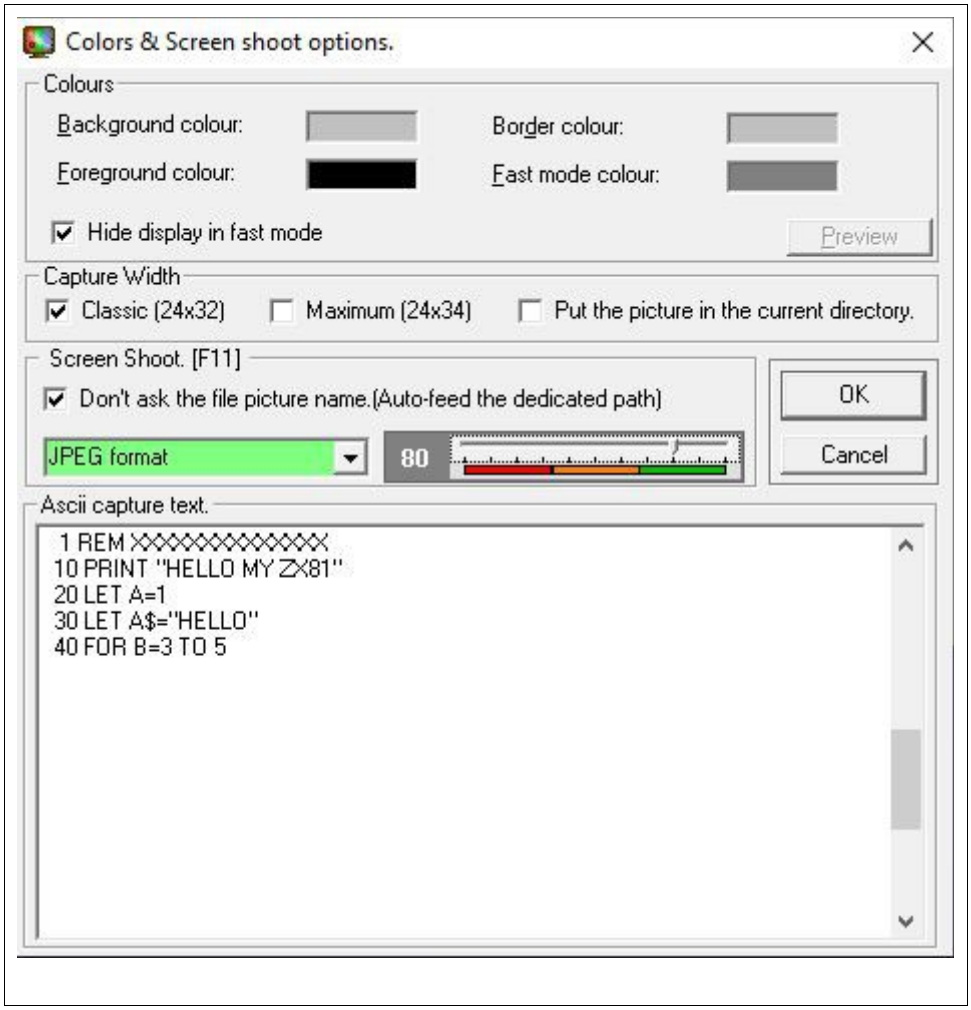
4) Fast load profile: rectangle harmonic, must be retrieved with a specific loader.

NOTE: This feature is an asynchronous buffer !
You can't directly save a sample like specifics Fast-Loads or beepers redirect to the "mic" output.

Tip:
The "real" sample is not available in this windows, there's too many different profiles!
But, the sinus, use ***SAVE "MYPRG.WAV"*** *to save the more representative sample.*
SAVE "MYPRG.FL" *will write a WAV file in a OTLA Fast-Load Format.*



Colors:



To change ZX81's screen colours.

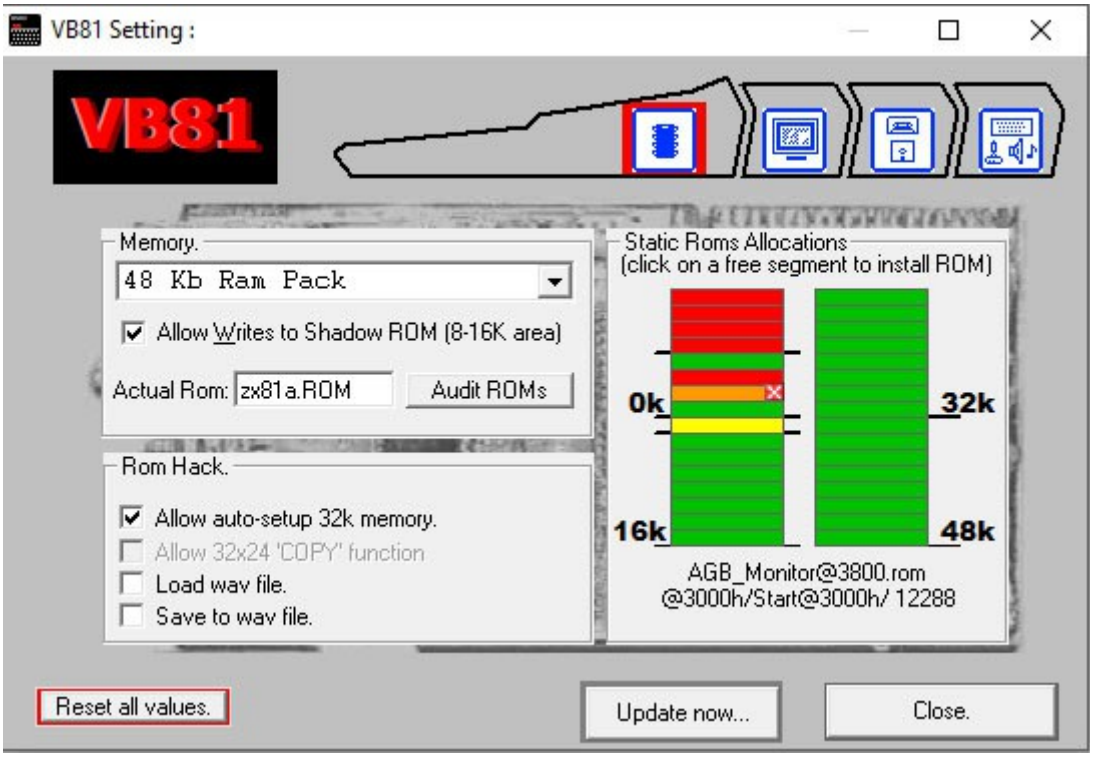
Auto-Save picture system:

In case of bitmap pictures, it able to save the picture with the windows explorer interface while using « F11 ».
If the « **[]Don't ask...**» checkbox is selected the screenshot image will be saved in the « screenshot » directory using the next empty « Capxxxx.bmp » in this directory. If « Cap0000.bmp » exist, the file name will be « Cap0001.bmp »,
If the picture format is specified, all pictures will be save in this format.

To save the image in the working/project directory select the "**[] Put picture in the current directory.**" checkbox. Keep in mind this option to avoid to search yours pictures everywhere on your computer hard-disk!

The "ASCII capture text" is the ASCII screenshot buffer.(Shifted F11 key) located in the "\screenshot \ASCII_Screen.txt" text file.

Settings:



Setup the memory 1, 2, 16, 32 or 64kb.

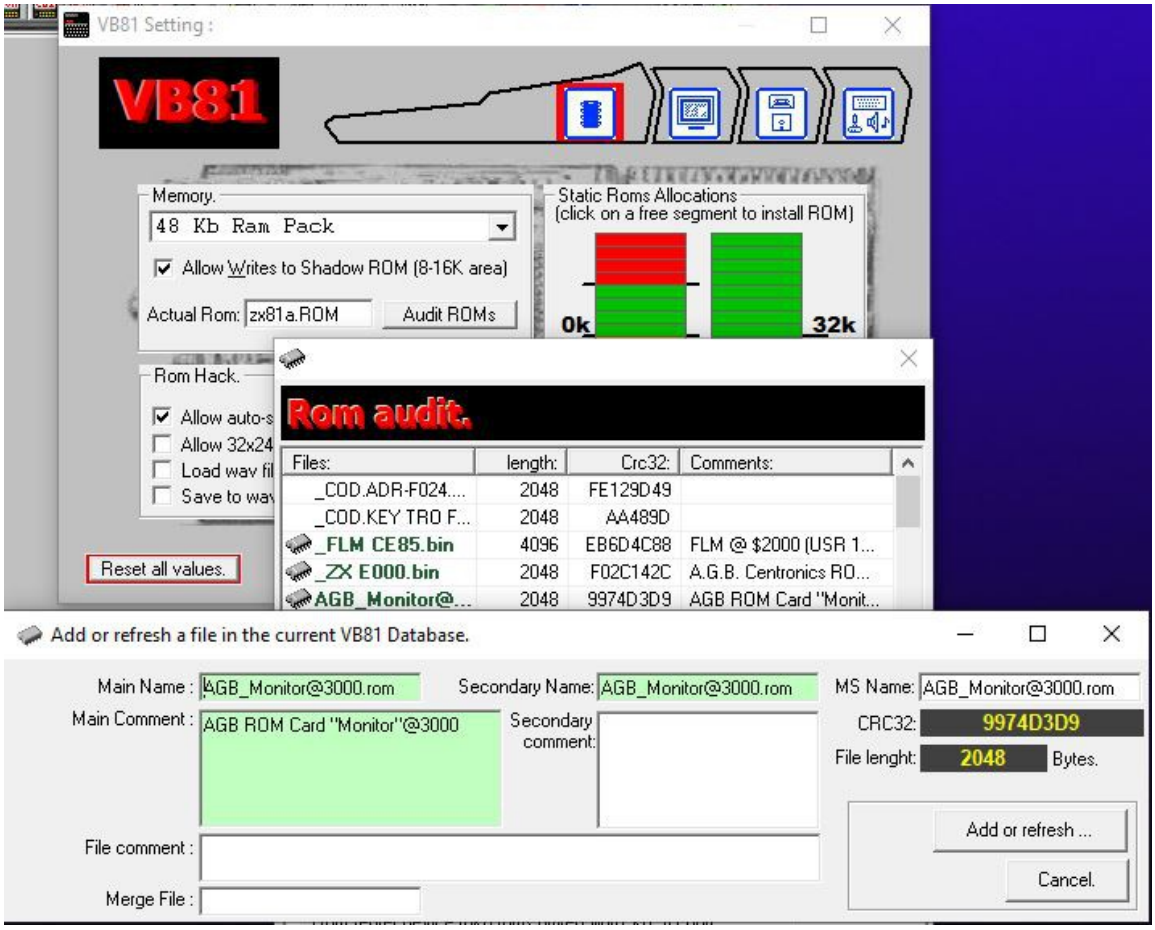
"**Allow writes to shadow ROM**" activates the 8k RAM segment between the ROM (0-2000h) and BASIC RAM (4000h).

With an original ZX81, this segment is a ROM mirroring and can't be written. But, this address segment can be use to address 8k RAM or add-on ROMs (like Larken RAM and Larken buffer).
64kb Memotech interface able to add a RAM segment in 2000h-4000h.
This segment is usually used to store machine-code programs or data, it isn't alter by a LOAD"" or NEW command so be used to store UDG data with a new IR offset (I register points to the new font definition).

« Audit ROM » click will change the BIOS BASIC monitor located in $0000 to $1FFF,
If you had to place a ROM in the memory, just click on the free (green) memory segment by $800/2048 byte blocks.

ROM Hacks:
"**Allow auto-setup 32k**" is a ROM patch to change the original memory scan ($7FFF) to $BFFF.
The RAMTOP will be automatically setup at the hardware/software reset !
Change ROM to allow advanced functions:
"Allow 32x24 'copy' function" : Patch the ROM to print a full screen in BASIC mode.(instead of 32x22)
"Save to WAV" : Use a Patch to redirect SAVE BASIC function to your hard disk or in a separate WAV file.

Easy custom ROM installer / ROM audit system:



To change the BIOS BASIC Monitor using « Audit ROMs » or set another location with a custom ROM. Click on the ROM file to install it in the memory. (if the new BASIC Monitor work, you had to confirm it!) The original BIOS ROM will be reinstalled in case of boot fault.

All ROMs are located in the « \ROM » folder.

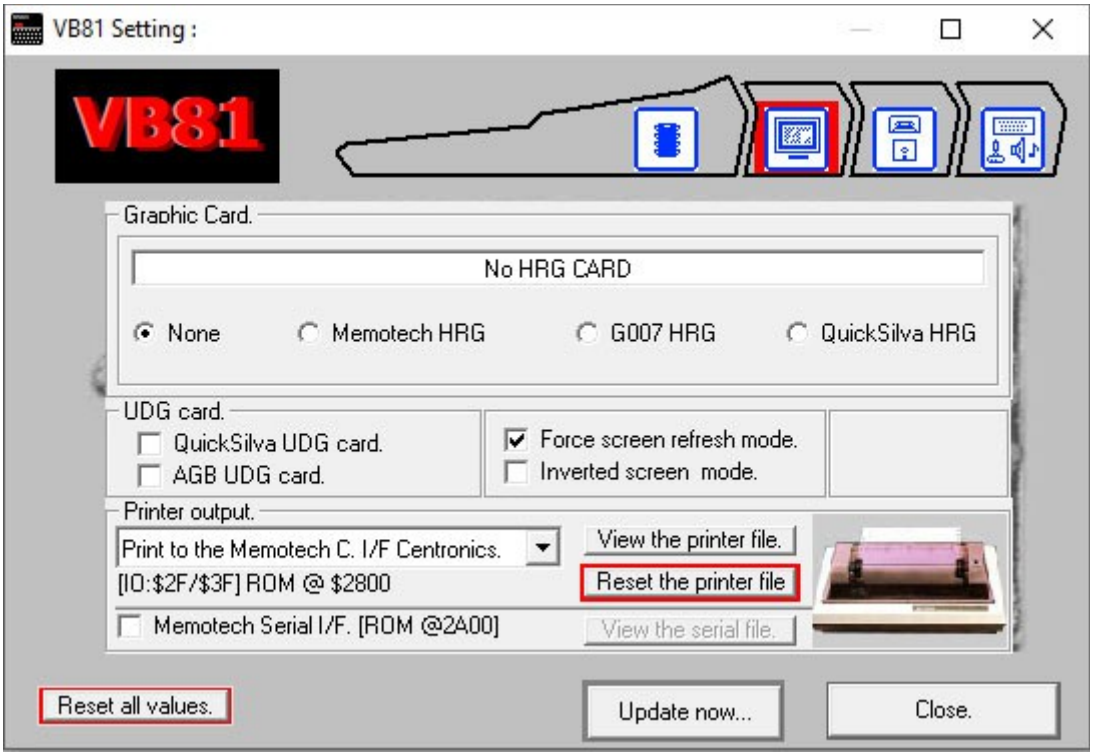
Use the database to add comments and retrieve specifications.
You can add your own Database by clicking on the file, and add address specifications and manufacturer.

This feature doesn't target the file name, but it's CRC32 (checksum) identity, and can put the comment on several files with different names (same content).

In case of custom ROMs, click on the first ROM block location, It will feed the right memory block room, and tag it in the orange colour, with the selected file.

All blocks tagged in red, can't be removed. These blocks are used by a card or an internal feature, like graphics cards, Memotech ROMs or CIF/SIF cards.

The custom ROMs, can easily be removed by clicking on the “close [X]” icon in the orange bloc.



Graphic cards:

- **None :** Use the classic ZX81 display.

- **Memotech :** Emulate the HRG Memotech HRG graphic card.
  Load card's ROM at 8192 and allows 0-1024 RAM support and redirect screen display RAM segment.

> ***Tip:***
> *Have a look to HRG Memotech demos in "HRG-CARDs-DEMO" folder.*

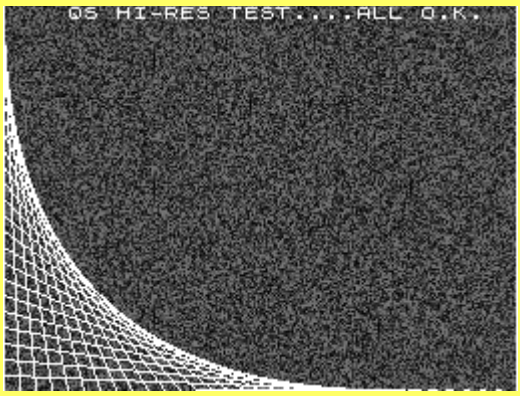
- G007 : Emulate the G007 HRG graphic card.
Load card's ROM at 10240 and redirect screen display RAM segment.
The ROM is called with classic BASIC functions (CLS 1...)

> ***Tip:***
> *Have a look to HRG G007 demos in "HRG-CARDs-DEMO" folder.*

- QuickSilva HRG : Emulate the QuickSilva HRG graphic card.
Load card's ROM at 10240 and redirect screen display RAM segment.
Have a look to the QuickSilva directory in the “HRG-CARDs-DEMO” folder.

Tip:

Type RAND USR 11036 to Launch card's auto-test .



Type RAND USR 11321 to print the screen on the ZxPrinter .

- QuickSilva UDG card:

- AGB UDG card :

- « Force screen refresh » : Display all screen frames.

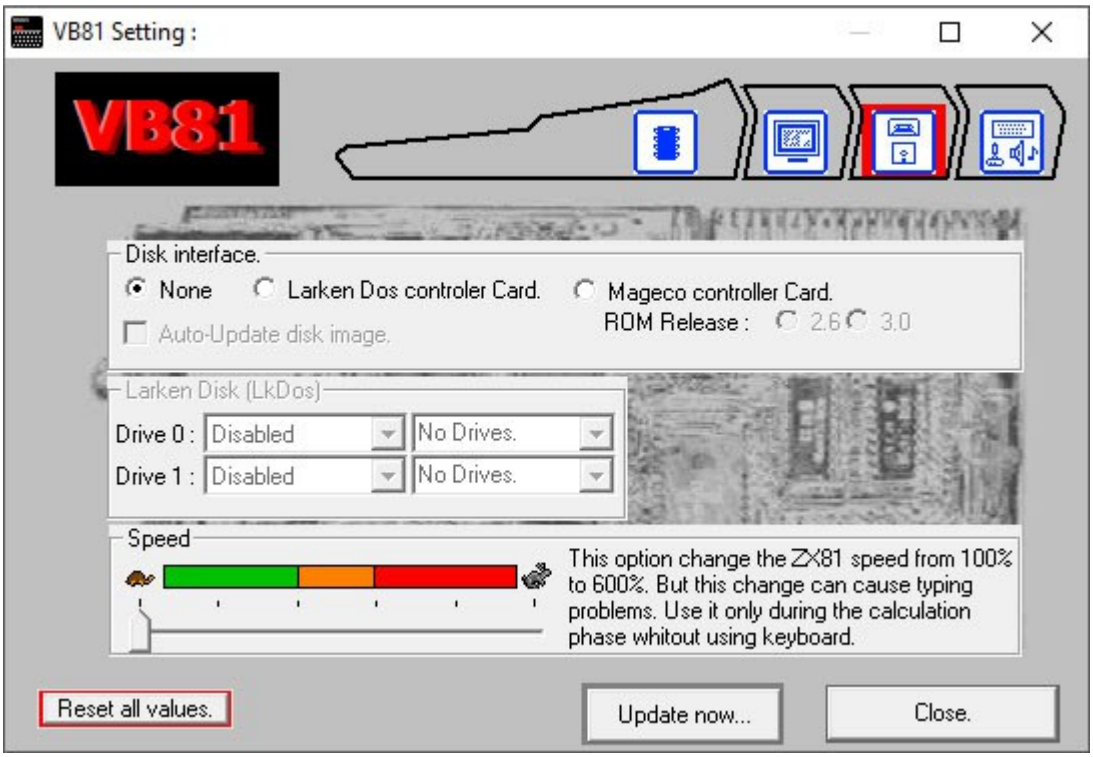
- « Inverted screen mode » : All printed character will be in inverted mode or normal, if reversed.

- Printer out :

1. **Select the ZxPrinter output.**
2. **Select the Memotech C.IF (Centronics interface) and install the inboard ROM.**
3. **Select the A.G.B. Centronics card and install the inboard ROM.**

4. **Select the Memotech S.IF (Serial interface) and install the inboard ROM. (can be use with the C.IF card)**

Note : 2,3 and 4 options are designed to send ASCII characters to the printer.



Floppy disk interfaces:

- Larken disk controller card:

Load card's ROM at 14336 and active virtual drives.

In Larken Disk panel, select a drive (or both drives).

Your drive is now active, and you can insert a new disk in your drive.

In the main VB81's frame, select “Larken Controller” pop-up menu, select “mount disk” and load a disk image in your virtual drive.

Default directory : “/Disk” in VB81_XuR directory.

Type: RAND USR 14336

and type « DIRE » for DIREctory

If you check the Auto-Update, the disk image will be saved if a sector is changed. If unchecked, the disk image is only saved if the disk is ejected or if the main frame is closed.

Tips:

1) LOAD"+" also display the current active drive's disk directory.

2) If you had to create a new disk, you had to initialize your disk image with "INIT" in the Larken editor.

3) If you had to change the active drive, you had to load a BASIC program in P_TOOLS directory named "CHDRIVE.P".

For more details, have a look to English Larken documentation available at http://ZX81.ordi5.free.fr/larken

Tips:

(1) - You had to select and active a drive in "Larken Dive" box !
(2) - The Larken or Mageco controller can't be used with a graphic card.
both use the same ROM/RAM memory segments.

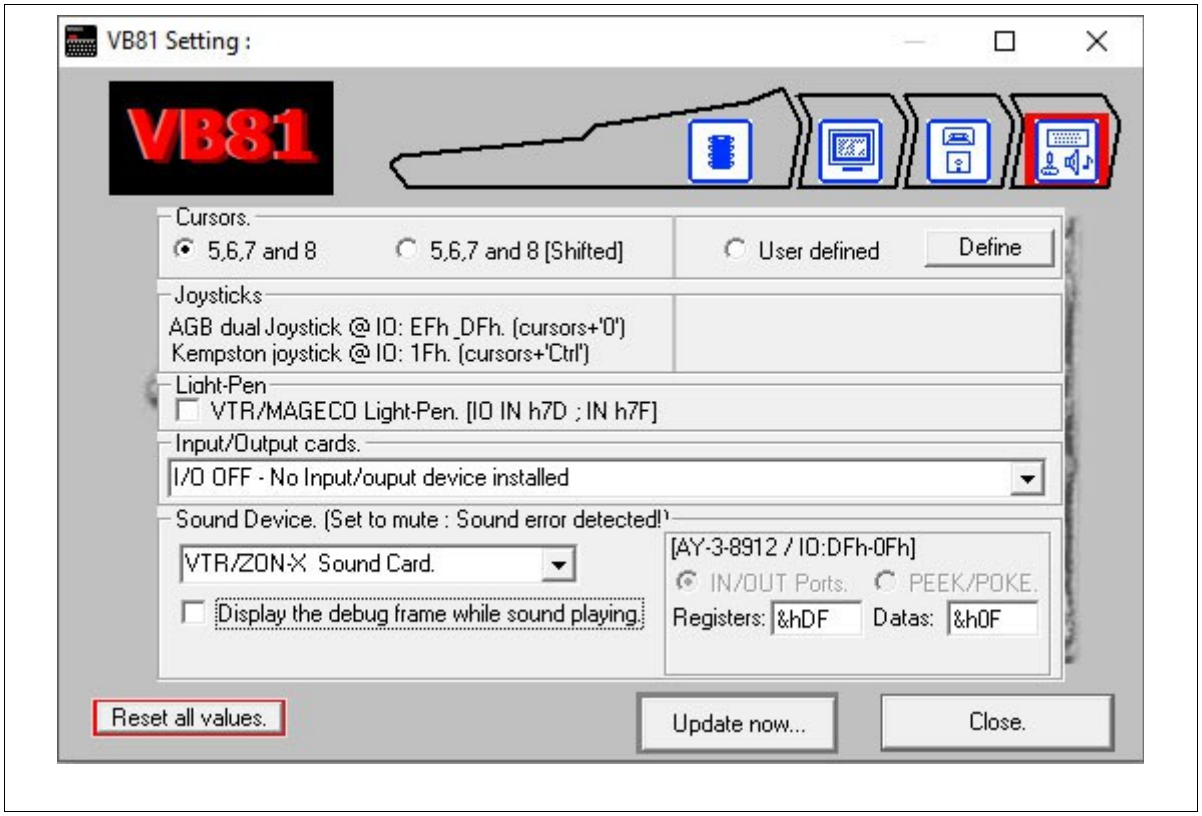
- **Mageco disk controller card:**

  Site AbandonMagazine d'Ordi-5.- Test de la carte contôleur MAGECO pour votre ZX81 et VP100 - (free.fr)

  http://ZX81.ordi5.free.fr/vtr/download/DOC_floppy1.pdf
  http://ZX81.ordi5.free.fr/vtr/download/DOC_floppy2.pdf

Z80 Speed overclocking:

Speed: Change the z80 clock.



Cursors:

Set the PC cursor to unshifted/shifted ZX81 cursor.
You also can assign specifics keys (Shifted or not)... Just select the key on the displayed keyboard!

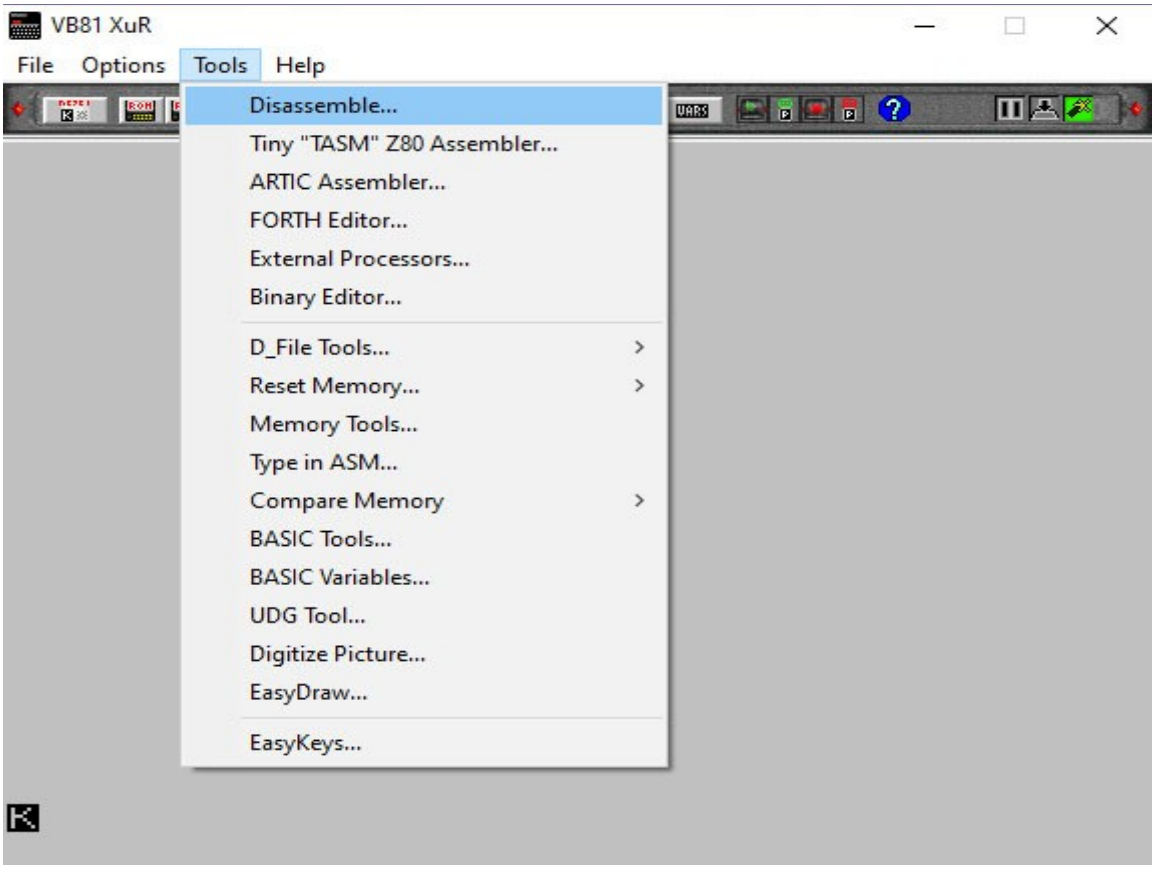
Input-Output cards:

Active IO interfaces on the computer.

Sound card:

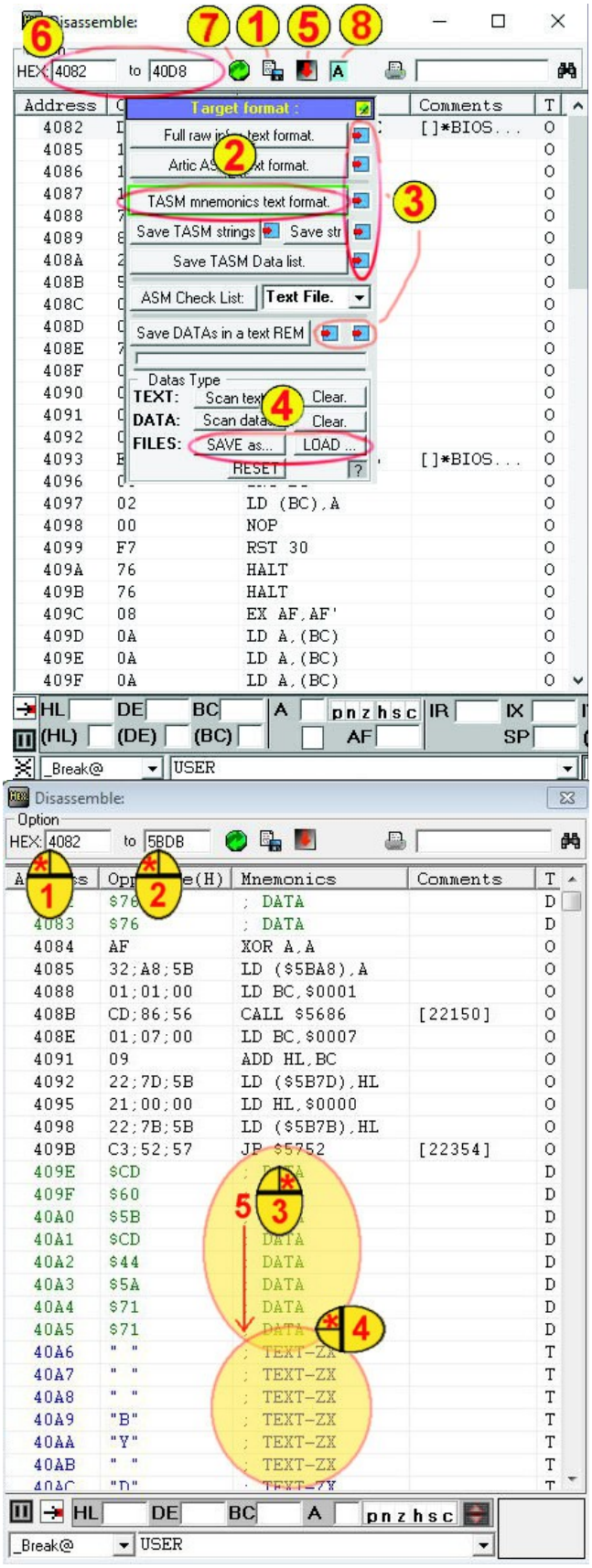
Debugging feature to display all data assigned to address 0h to 4000h.

Tools... :


VB81 XuR
File
Options
Tools
Help
Disassemble...
Tiny "TASM" Z80 Assembler...
ARTIC Assembler...
FORTH Editor...
External Processors...
Binary Editor...
D_File Tools...
Reset Memory...
Memory Tools...
Type in ASM...
Compare Memory
BASIC Tools...
BASIC Variables...
UDG Tool...
Digitize Picture...
EasyDraw...
EasyKeys...

Disassemble and Debugger window:

Open the inboard disassembler window.



(1) Text Export window.
(2) Export to labelled text TASM mnemonics.
(3) Copy all text in the clipboard.
(4) Save or Load user data set in memory.
(5) Import or export binary to a file.
Load : The binary will be loaded between the low and high address range set in the address box (6).
If the binary is too large, it will be trimmed to suite to the right length.
Save : The binary data between the low and high address rang will be saved.
(6) Set the disassembler address location.
Default in hexadecimal, you can switch to decimal by clicking the address column header.
(7) Used to refresh the dissembled listing, and refresh the REM listing at the bottom of the window.
(8) Tiny assembler text export.
This button will load the inboard “Tiny Tasm” text box.

To set memory offset, click on offset text-box and enter the start offset or select the REM line number in the bottom combo box.

To set hexadecimal to decimal display:
(1) - Address: click on address column header.
(2) - Op. Codes: click on Op.Code header.
(1 click=Decimal, 2 clicks=ZX81 ASCII)

Mnemonics (standard Z80 codes):
JP NZ,$0000 = JP NZ 0000 in ARTIC ASM2
JR Z,+33 and LD C,(IY+40) are always in decimal.

Refresh green icon:
Refresh the listing.

Export icon:
This icon open an export feature to text format.
It able to write disassembled opcodes in a standard 'TASM' text file.

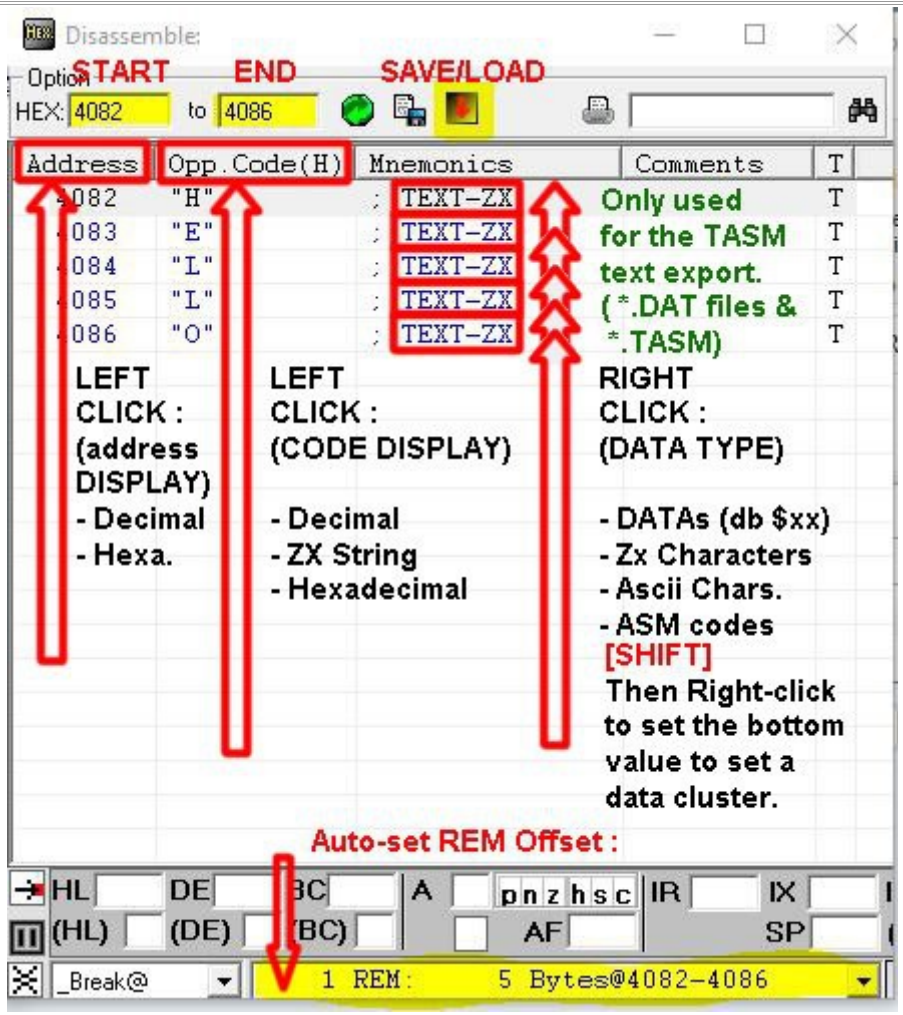
Print icon:
Print opcodes list to your PC printer.

Find Opcode icon: Retrieve an ASCII string in the mnemonic column.

Asynchronous debugging :
Just click on a row to add a debugging break point.
If the ZX PC register reaches this point, a new icon will be displayed to inform the user about register stat (A register) , jumps and running conditions.

In this window, the right click is dedicated to the « *Tasm Advance user disassembler* ».
The right click sets the data type display in the TASM like text file (3).
Another right click will switch the data tag (right column in the window) to the next data type (4)
To select a data block, just right click on the first mnemonic, release the mouse button, and press the Shift key.
The mouse icon will display the data type. Now, perform a right click at the end of the block (5).

Advance user disassembler and export menu:

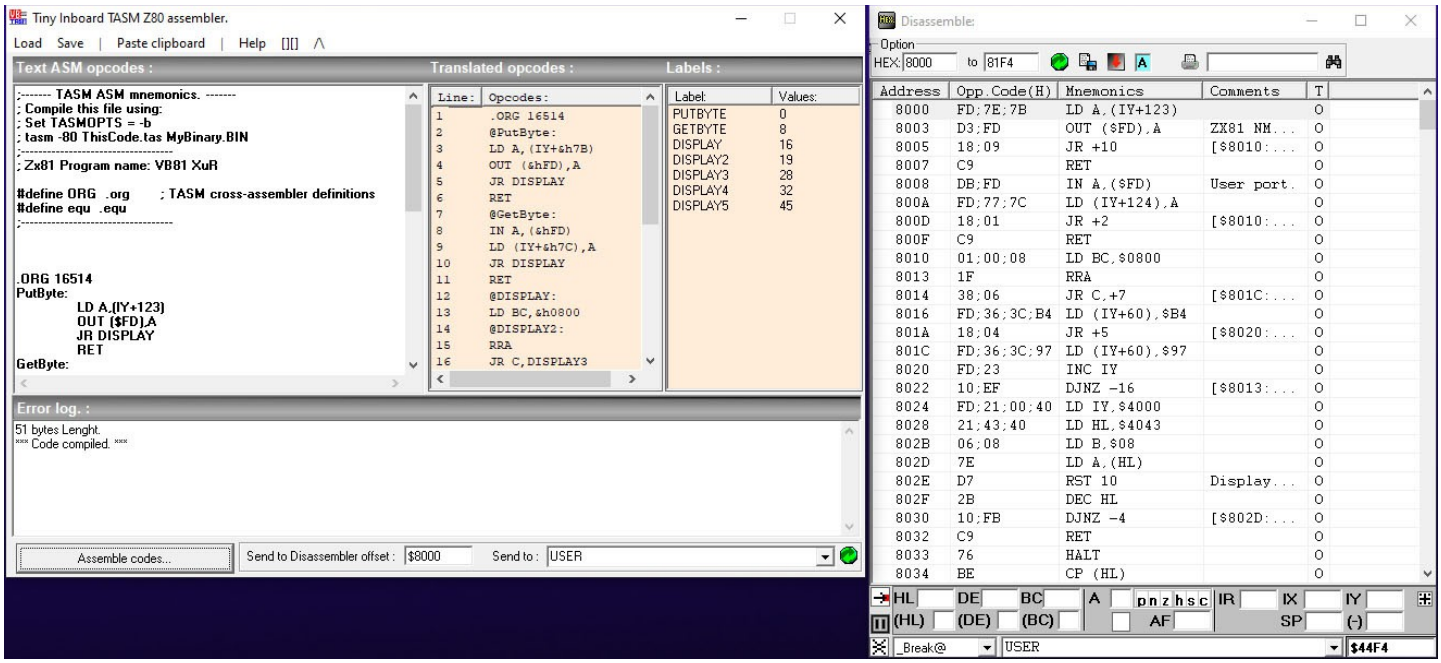


Select the REM number dropdown, and load the binary between the low and high address rang shown.
The higher or lower value can be changed [USER].

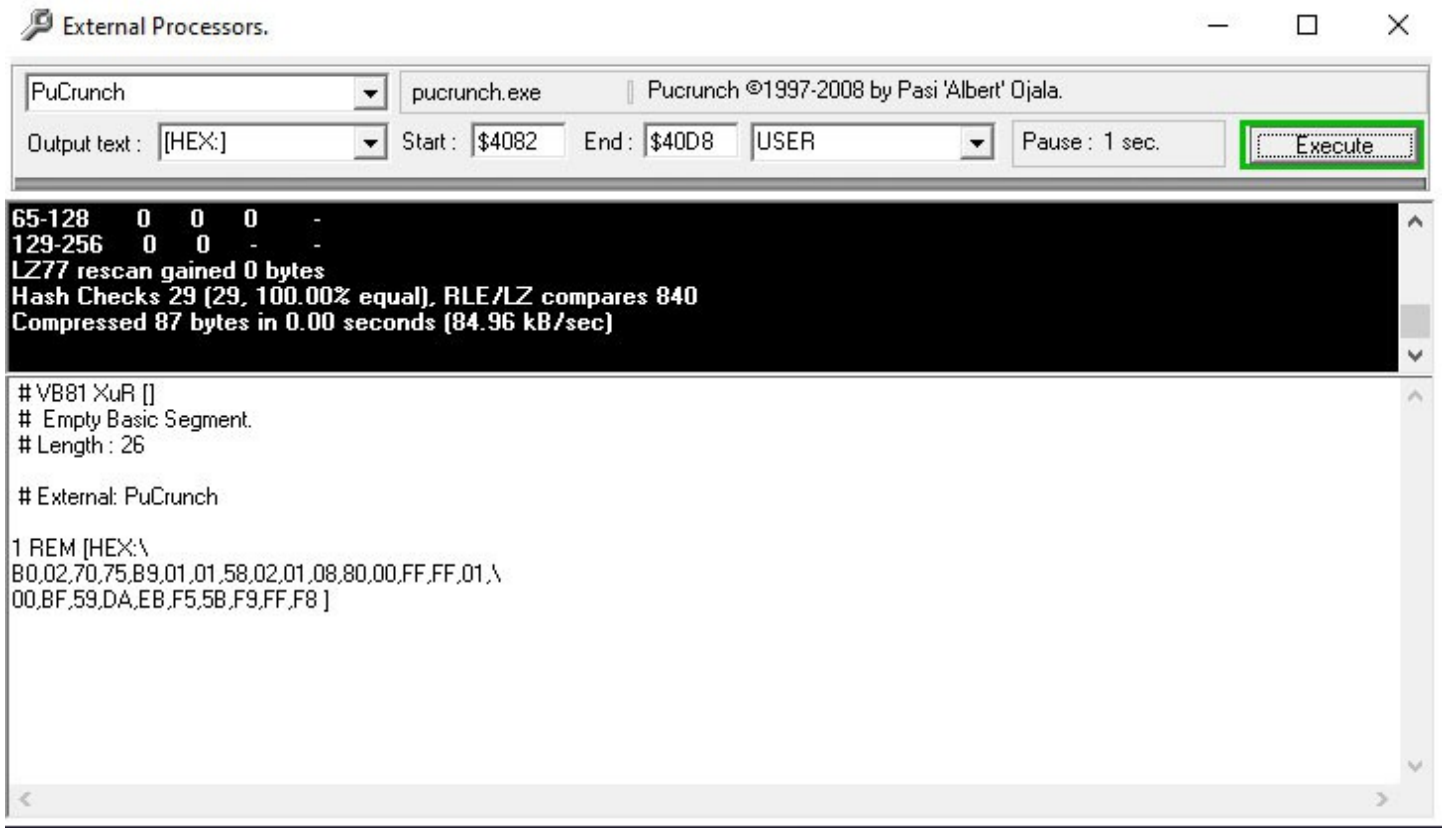
To load a binary, select a REM, the binary will be filled in the free REM memory room… and trimmed if the binary is too big.

If the selected address range is larger than the binary, it will be loaded starting at the lower address selected.

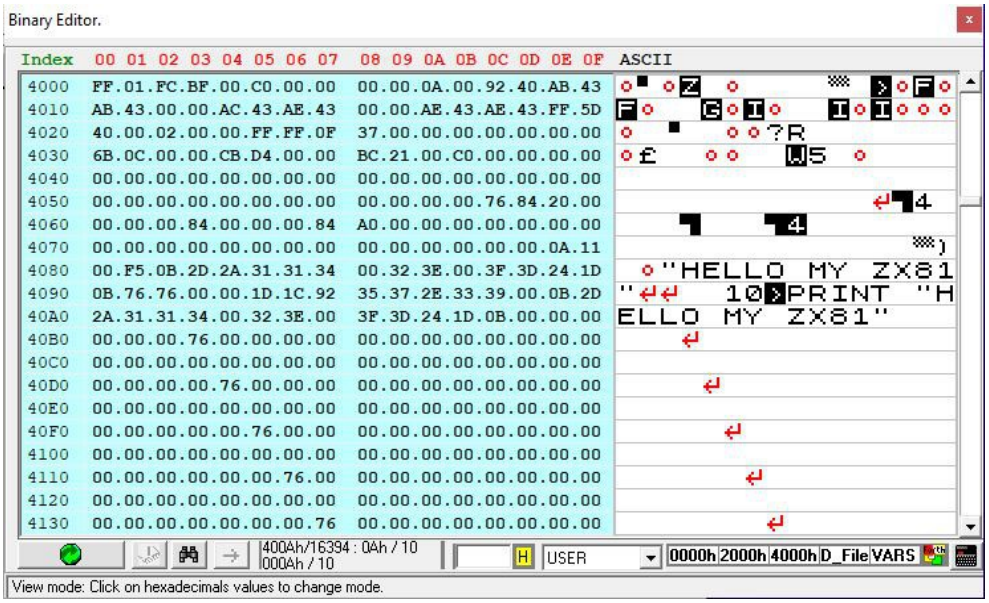
Tiny TASM format compiler :



External processors:

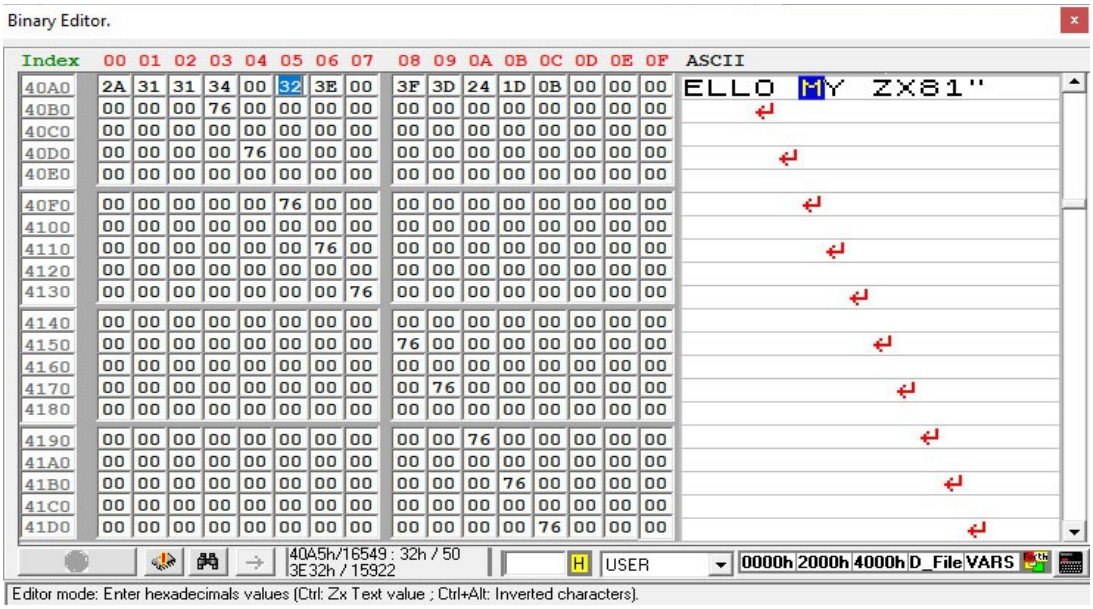


Binary editor:



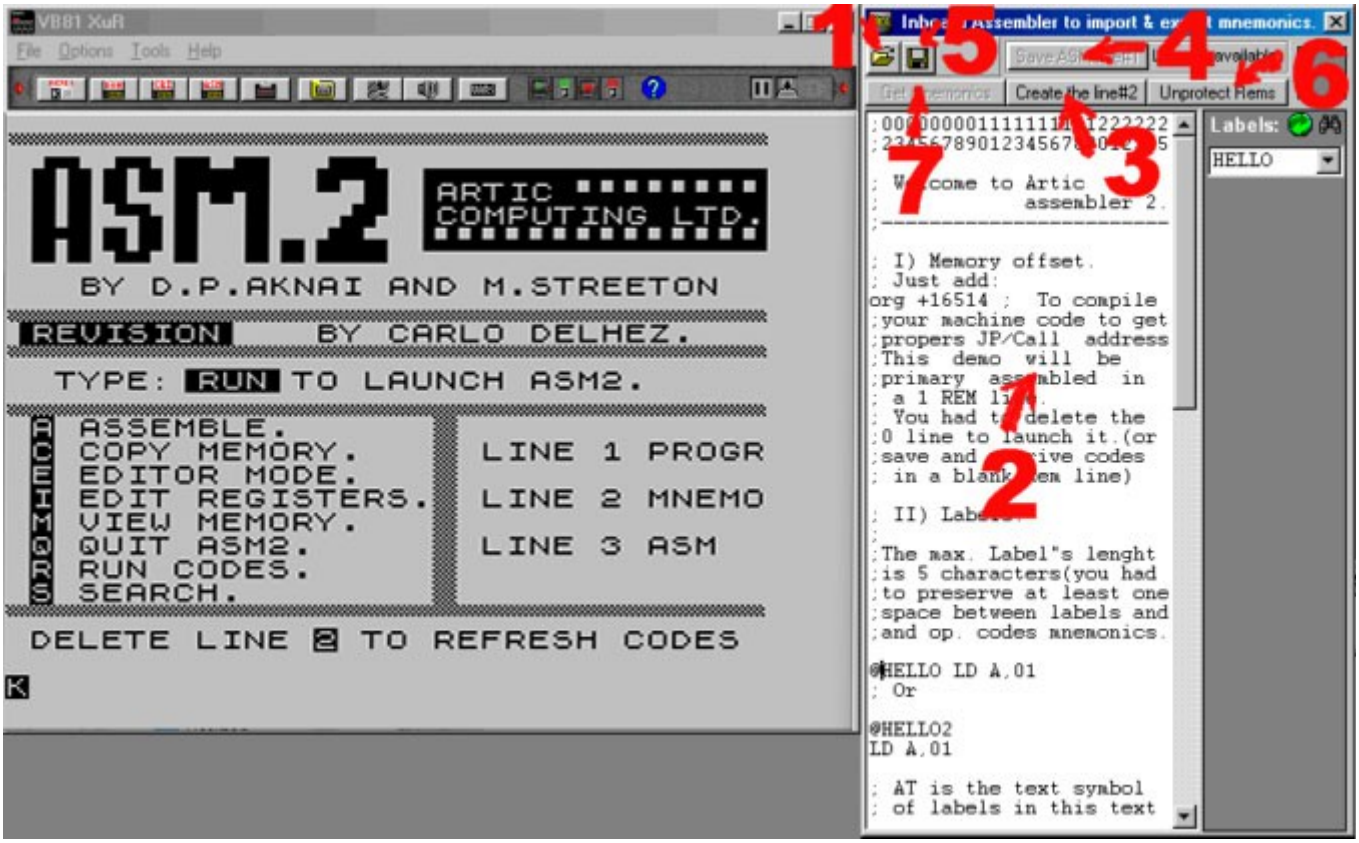
ZX81 memory in hexadecimal/decimal values.

Click on the left panel (hexadecimal values) to change the memory.



You can enter values in hexadecimal and press « Enter » to validate.
Press the « Ctrl » while a character to send a Zx text, shifted « Ctrl » to send an inverted characters.

The Zx memory will be updated when the value is set. To return to the editor window, click on the second bottom left button (the ROM chip).



ARTIC Assembler 2:

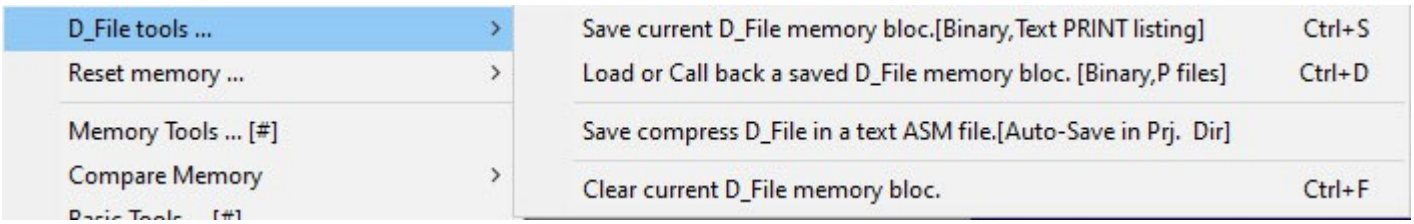
(1) « Open file » icon: Retrieve a text file in editor text-box.
(2) Text to send to the ASM2 compiler.
(3) Create the Rem line 2 and fill with the contents of editor text-box.
(4) Save ASM in REM line 1: Save ASM Codes include in the compiled line #1.
(5) « Save file » icon: Save a text file with the text in editor text-box.
(6) Remove the ASM2 header bytes.
(7) Get mnemonics in the REM line 2, if any codes are already typed/changed on the compiler.

Tip:

For more informations, have a look to the http://ZX81.vb81.free.fr/ASM2.html

D_file tools:

The D_file is the display memory segment.



This segment is located below the BASIC program and its offset depends on the length of BASIC program 4000h (empty BASIC program) to the top of the ZX81 RAM.

The D_File is include in your saved program.(“Save” function in BASIC mode)

If you need to retrieve the D_File offset, just click on the “system info” icon in the main tool bar, or LET DFILE= peek(16397)*256+peek(16396)

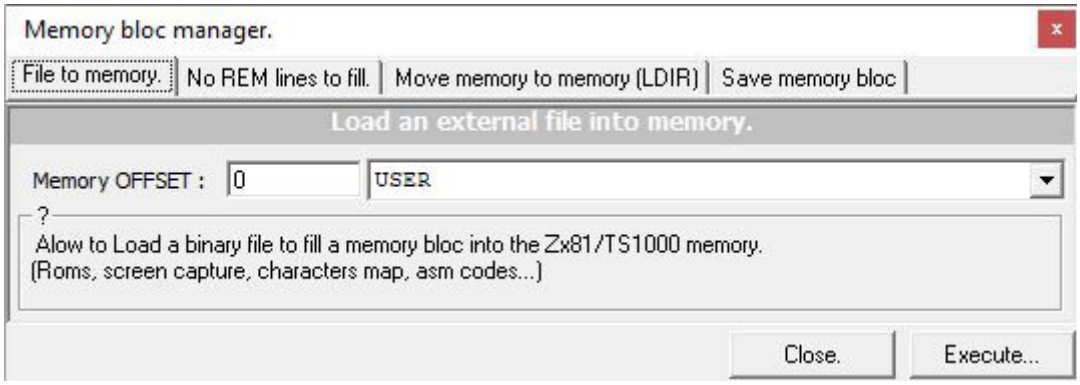
This toolbox can save, load or clear this memory segment.

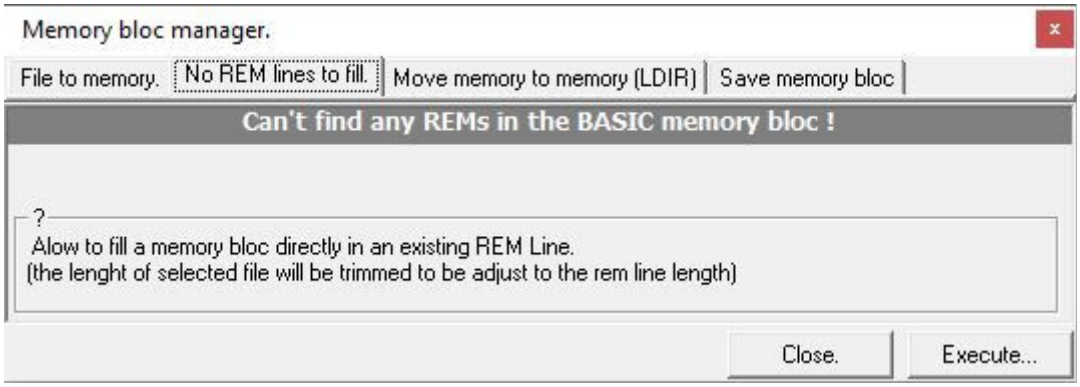
Reset memory to 32kb:

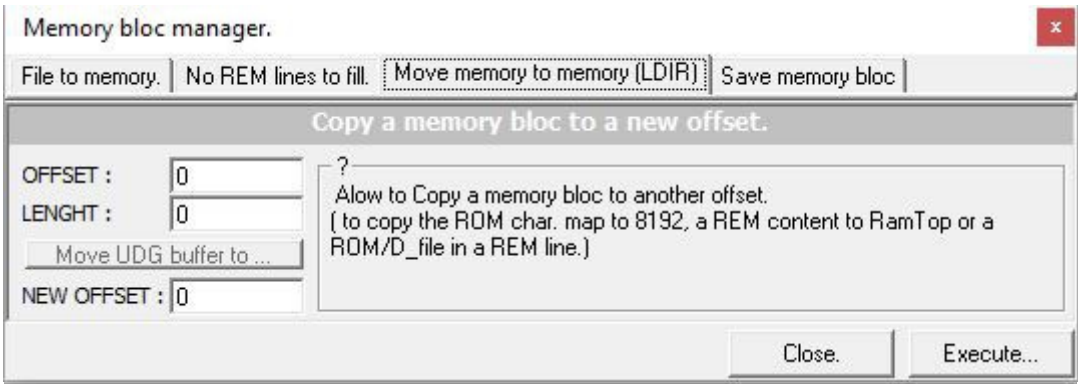
Set RAMTOP to 32kb.

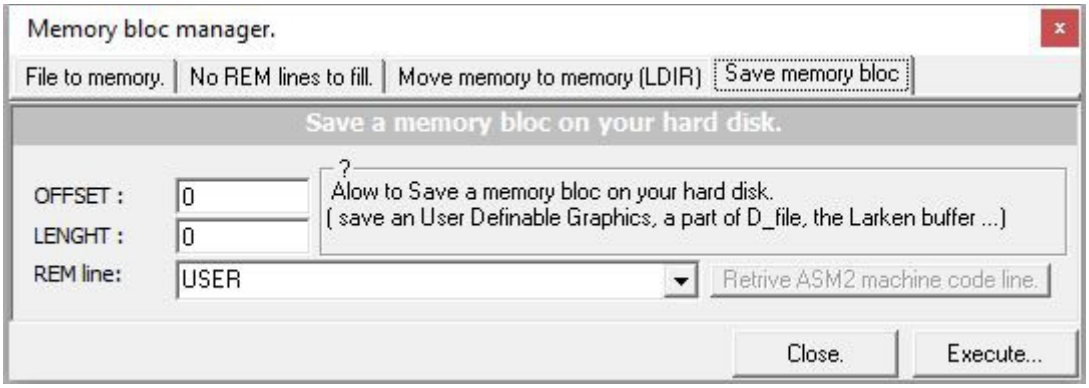
> ***Tip:***
>
> *You can use this feature to launch the ASM2 program in 32kb mode.*

Memory Tools:









Allow to import or export binary codes to ZX memory or your hard disk.

File to memory: loads a file, ROM or data, to ZX memory.

REM filler: Fill a REM line with data.

Move memory: Move memory block to another memory area.

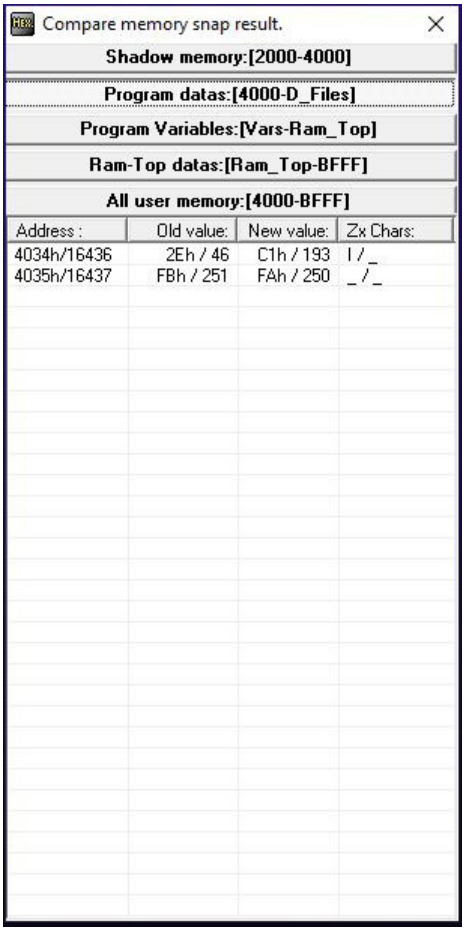
Save memory: Save memory area to file or REM line.

To retrieve an ASM2 segment (the Line “1 REM”), the mnemonic list must be assembled and the “1 REM” line must be present.

Just click on “retrieve ASM2 machine code” to extract ASM codes without header and footer.

Press execute to load or save binary file.

Compare memory:



Save a memory snapshot for later comparison.

Example: During a game your lives left credit is “3”.

select “Get memory image”

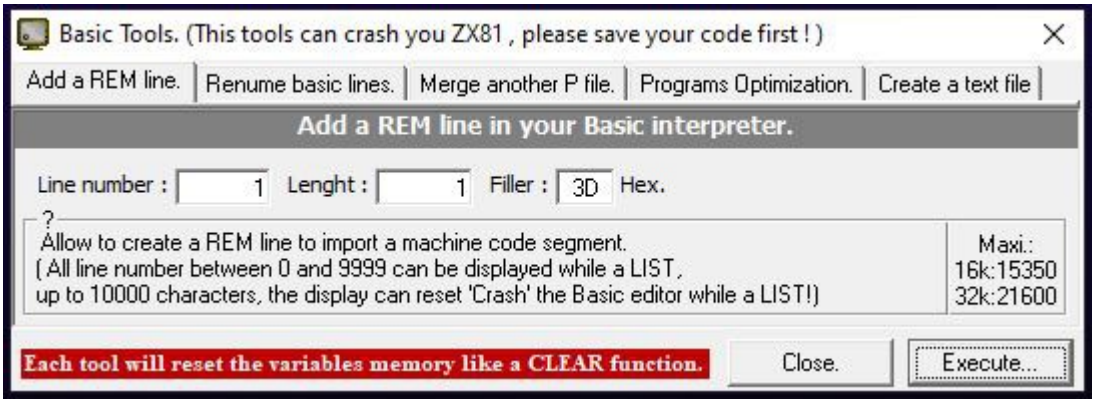
If your lives left credit is “2”, select “Compare...” and “Lower bytes”.

All bytes decreased or lower than your memory image will be displayed.

Note the “2” memory offset in this listing, and change this byte in the memory editor with “3” (or another value) to retrieve the old value.

Note: Can work with assembler programs with temporary data in REMs lines.
In case of BASIC programs, it recommended to open the “Vars” window, and type in LET LIVE=3 ... for example.

BASIC Tools:

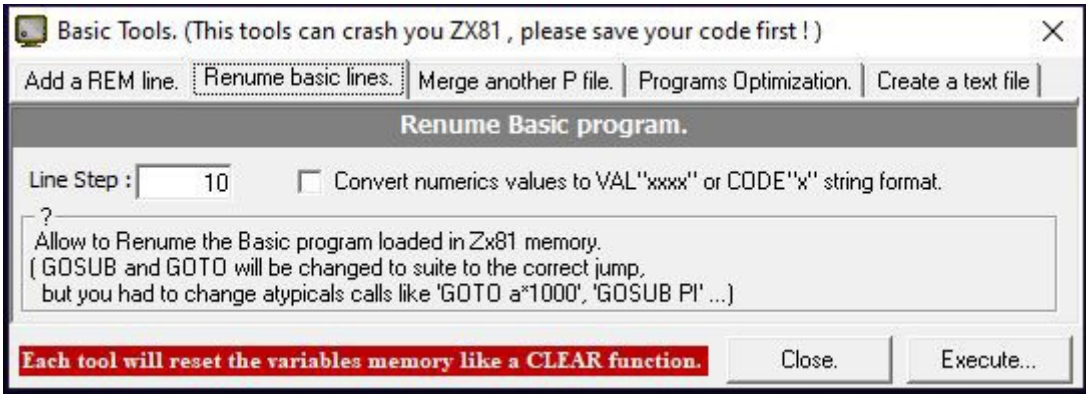


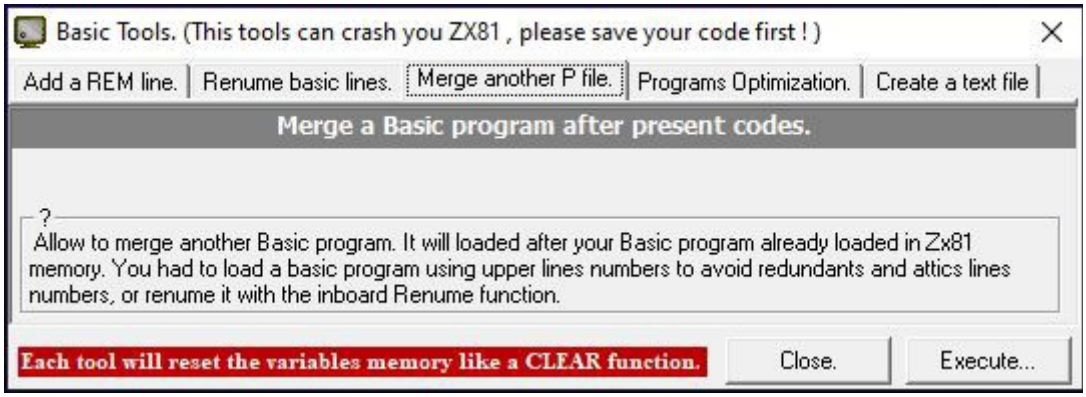
Add a REM line to your program to hold machine code or other data.

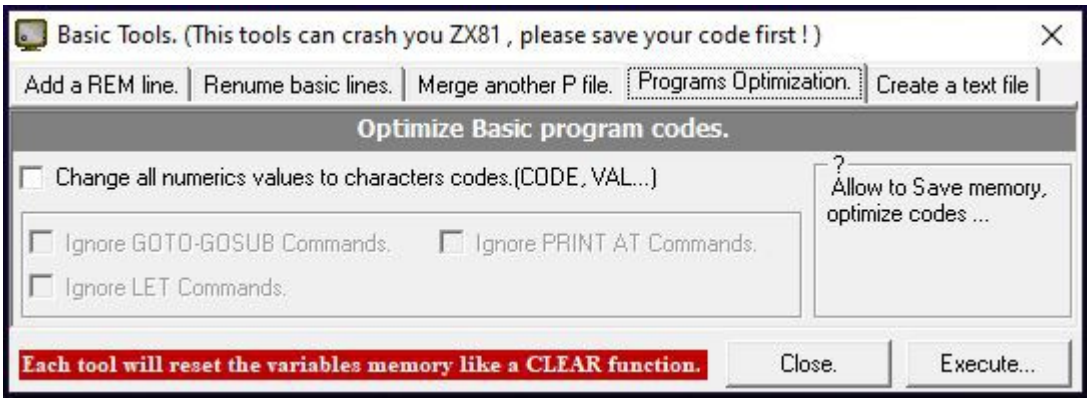
Note: With merge function, the BASIC compiler may crash after the “execute” command. It's due to several internal BASIC update in system variables. In fact, this process had to erase all variables and you had to reinit them to avoid problems.

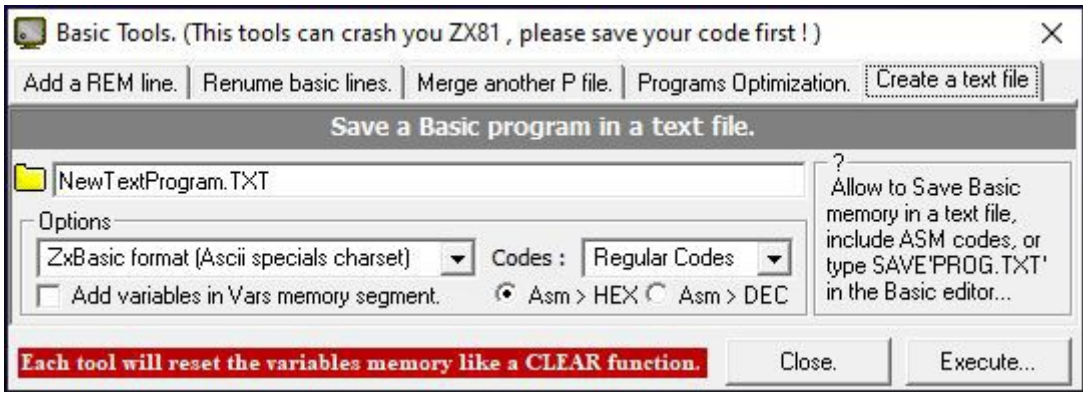
This window can convert your running BASIC program in a text file.

But, you also can use the BASIC SAVE function : SAVE"MYPROG.TXT"
A BASIC text file will be generated in the default directory.

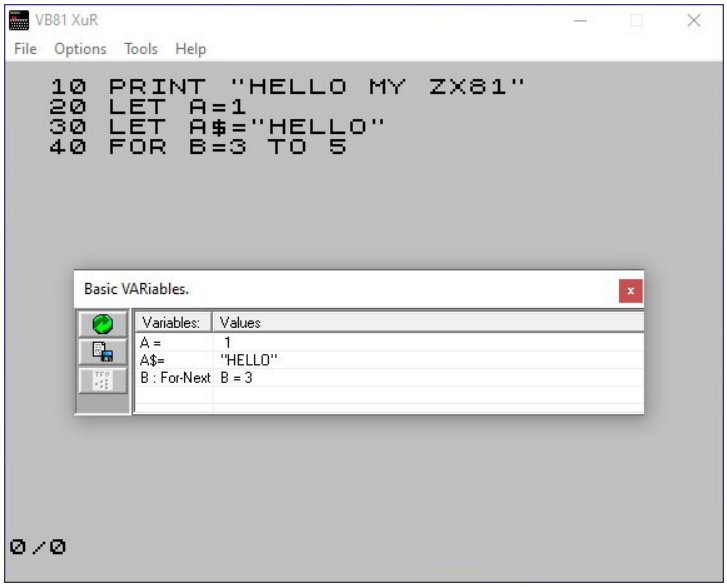






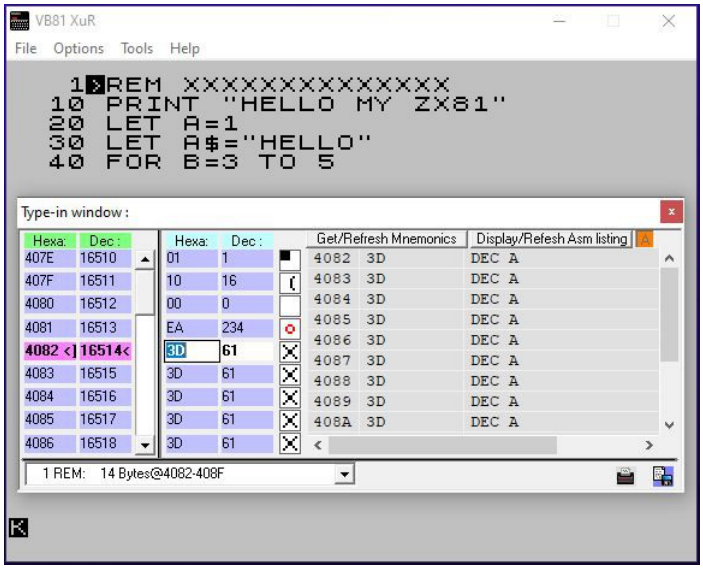


BASIC variables window:

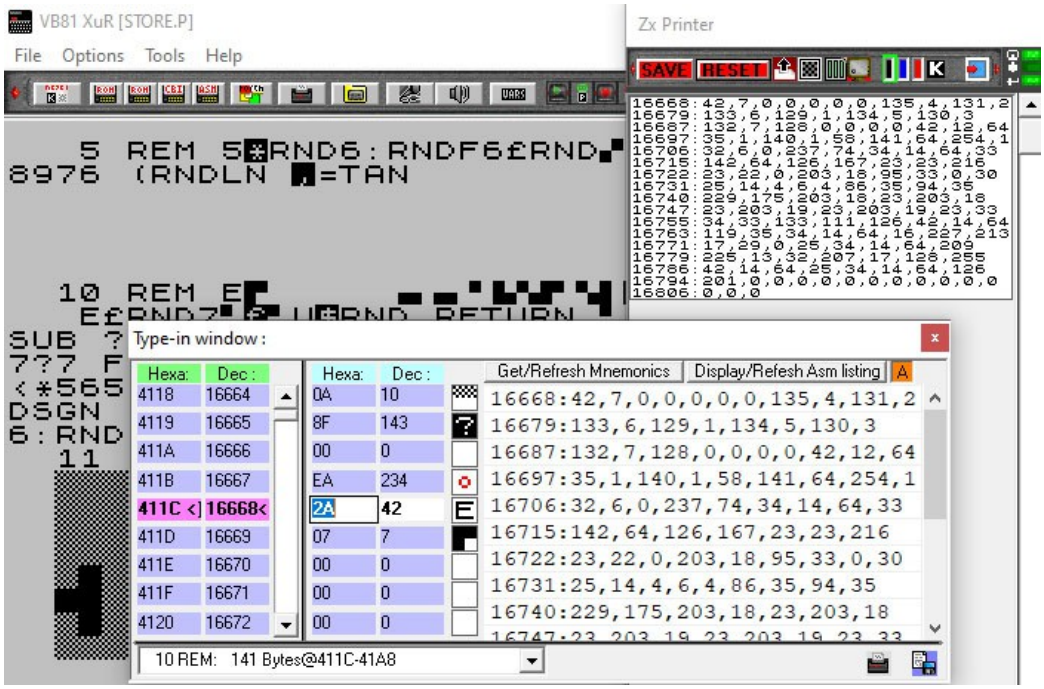


Display a window showing all BASIC variables and their values.

Type-in window:



Open the « Inboard type- in » window to change your REMs content.

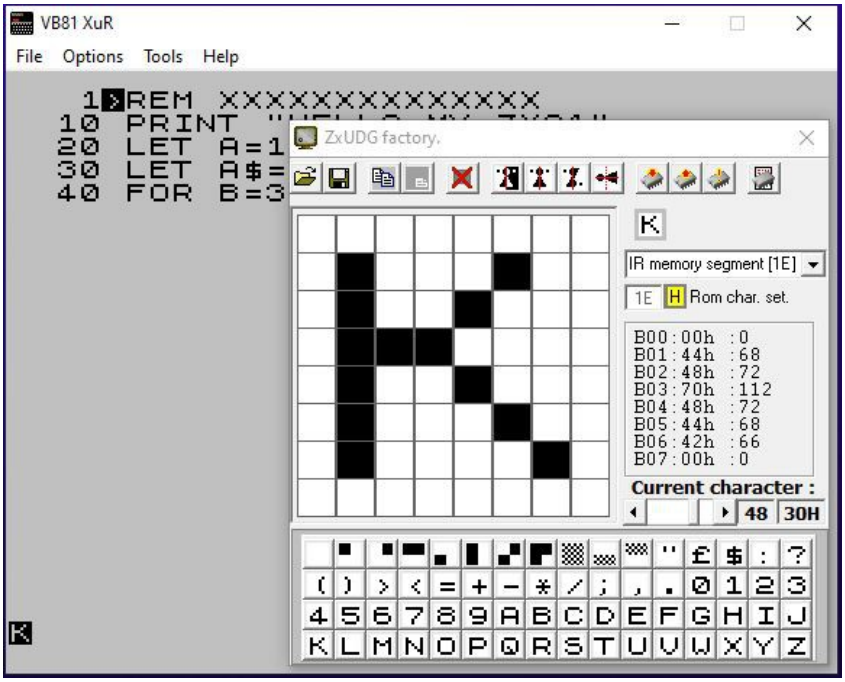


You can display the disassembled mnemonics or the decimal values listing:

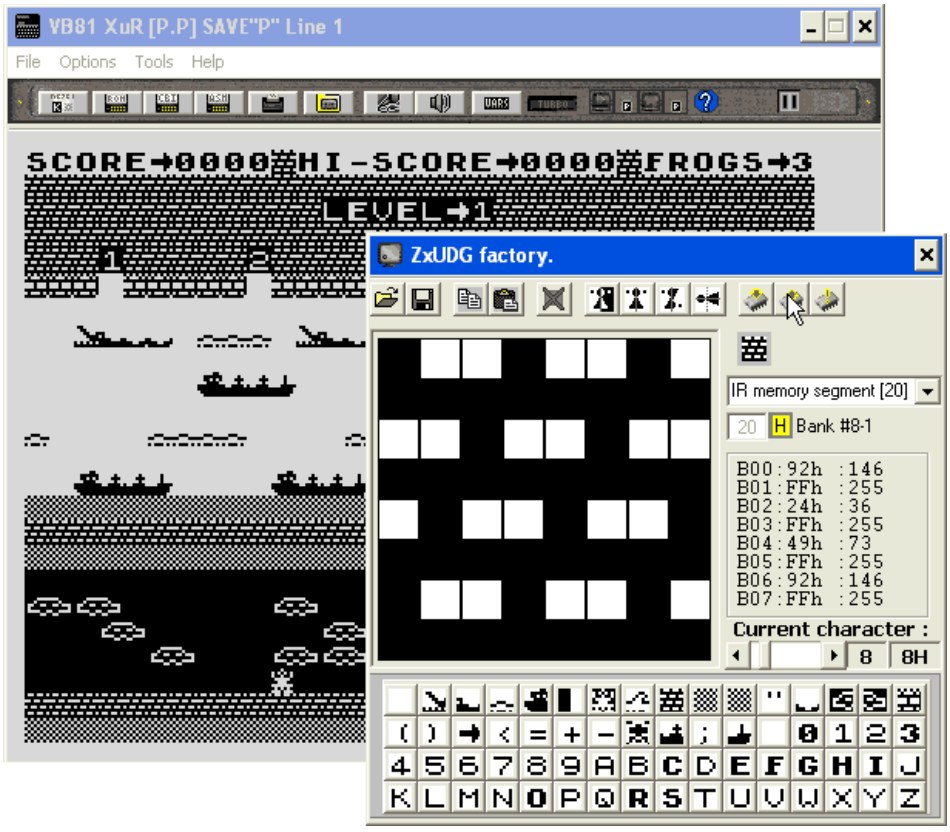
Address check list (orange button at the bottom right side):

[A]: Full decimal rows Zx printer listing : With 32 characters and cut the last value the the next line if the length is too large.
[B]: Display a line with 6 decimal values.
[C]: Display a line with 6 or 7 decimal values. : With 32 characters and cut the last value the the next line if his length is too large.

UDG TOOL:

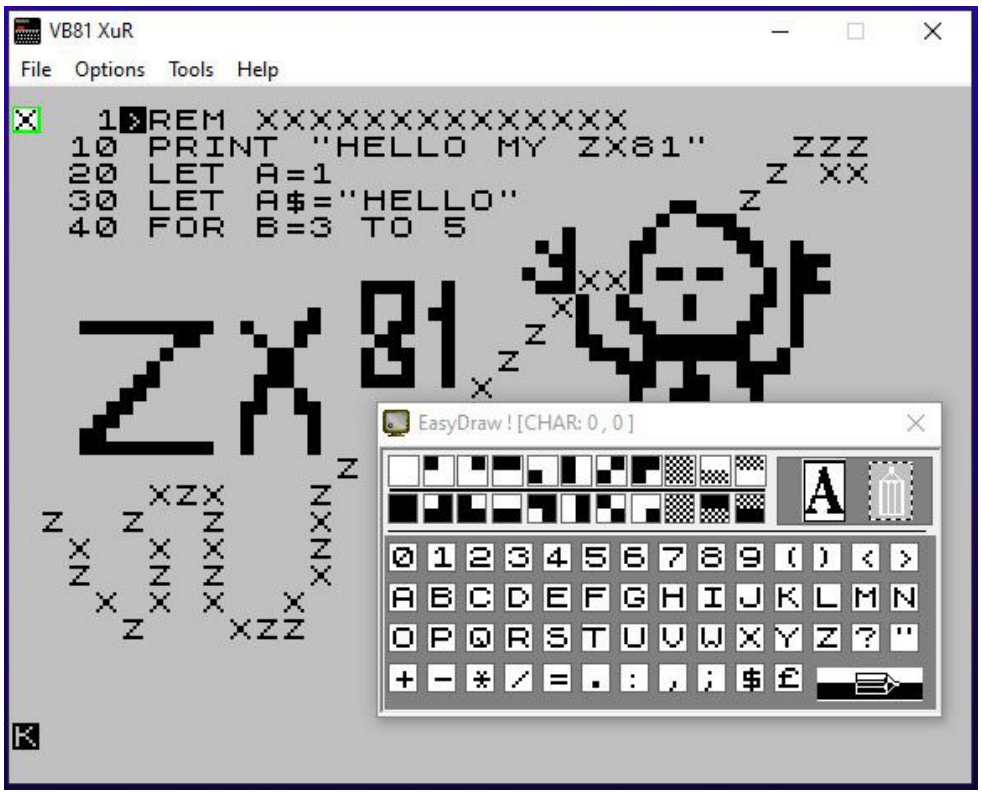


Display a window to change the ZX81 character set on the IR register and the Quicksilva & AGB UDG card.



Note : The ZX81 can't display UDG without screen drivers or dedicated cards.

EasyDraw:



This tool allows you to draw pictures to the D_File in real time. A minimum of 16k of RAM is required as an uncompressed D_File is needed. Drawing can only be done when the display is in a static, unchanging state such as “10 GOTO 10”,

Your drawing can be saved as binary, a BASIC or an ASM program using the “Tools>D_File Tools” menu.

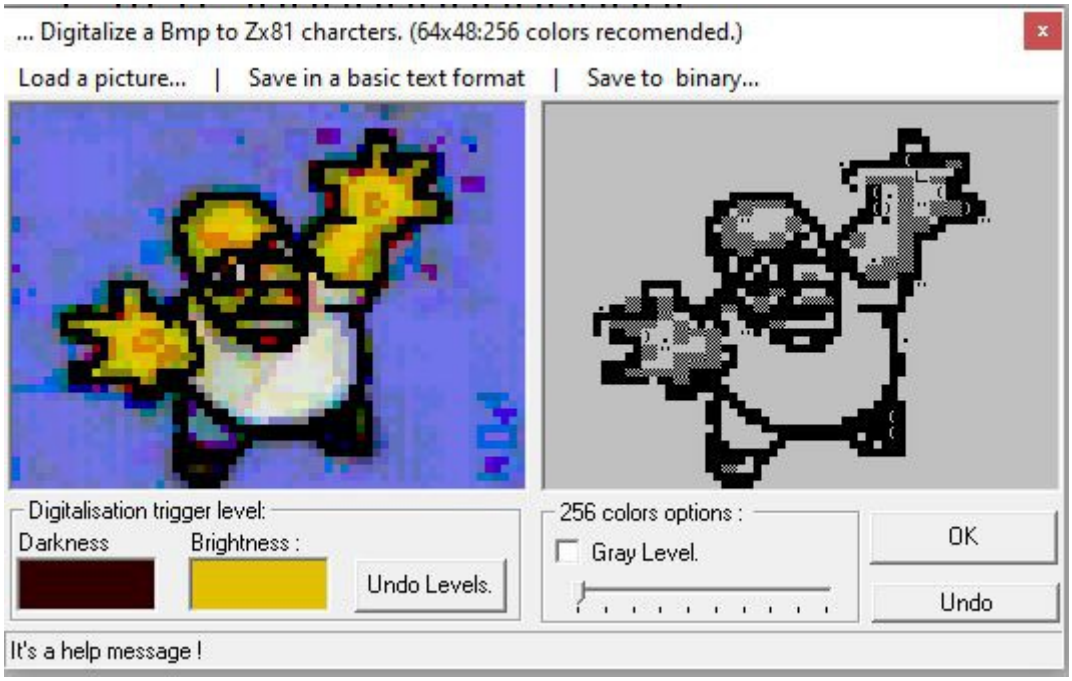
To select pixel drawing mode click the “Pencil” icon. Left click draws pixels, right click erases.

To select character mode click the “A” icon.
Select a character and left click on an area of the screen will put it at that location.
Right click on the screen sets the repeat mode, the selected character will be repeated while holding down the right button, left click again to end repeating.

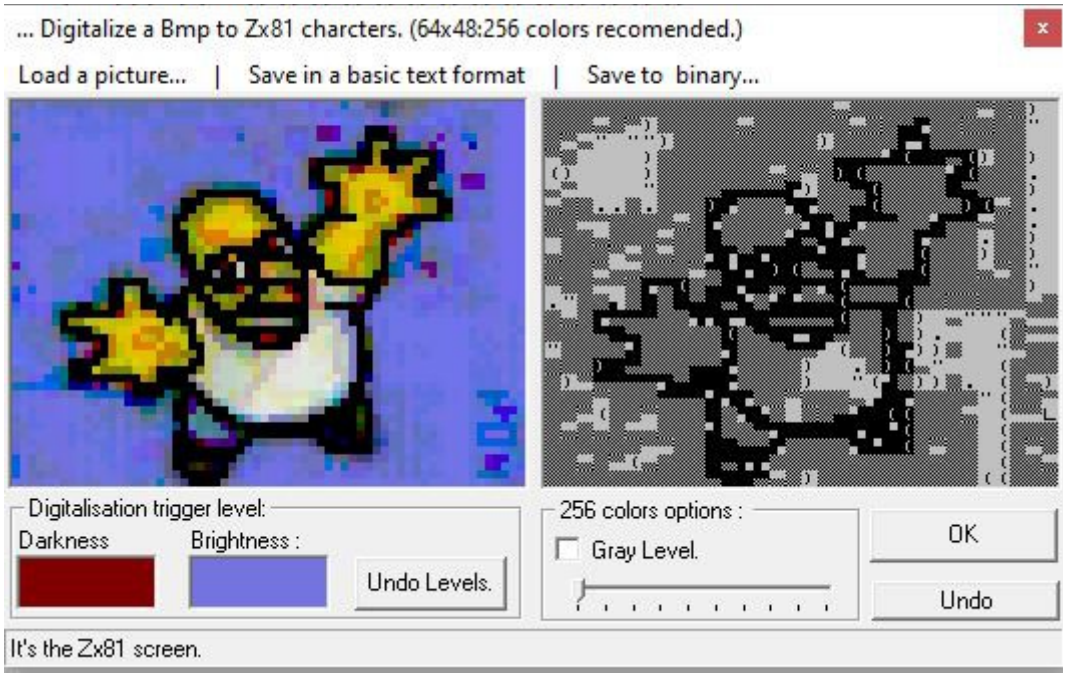
Now, you will be able to retrieve your drawing in BASIC text files or in an ASM compressed file.

Digitize a bitmap picture:

The ZX81 RAM had to be set to 16kb or more to load the 32x24 character screen.



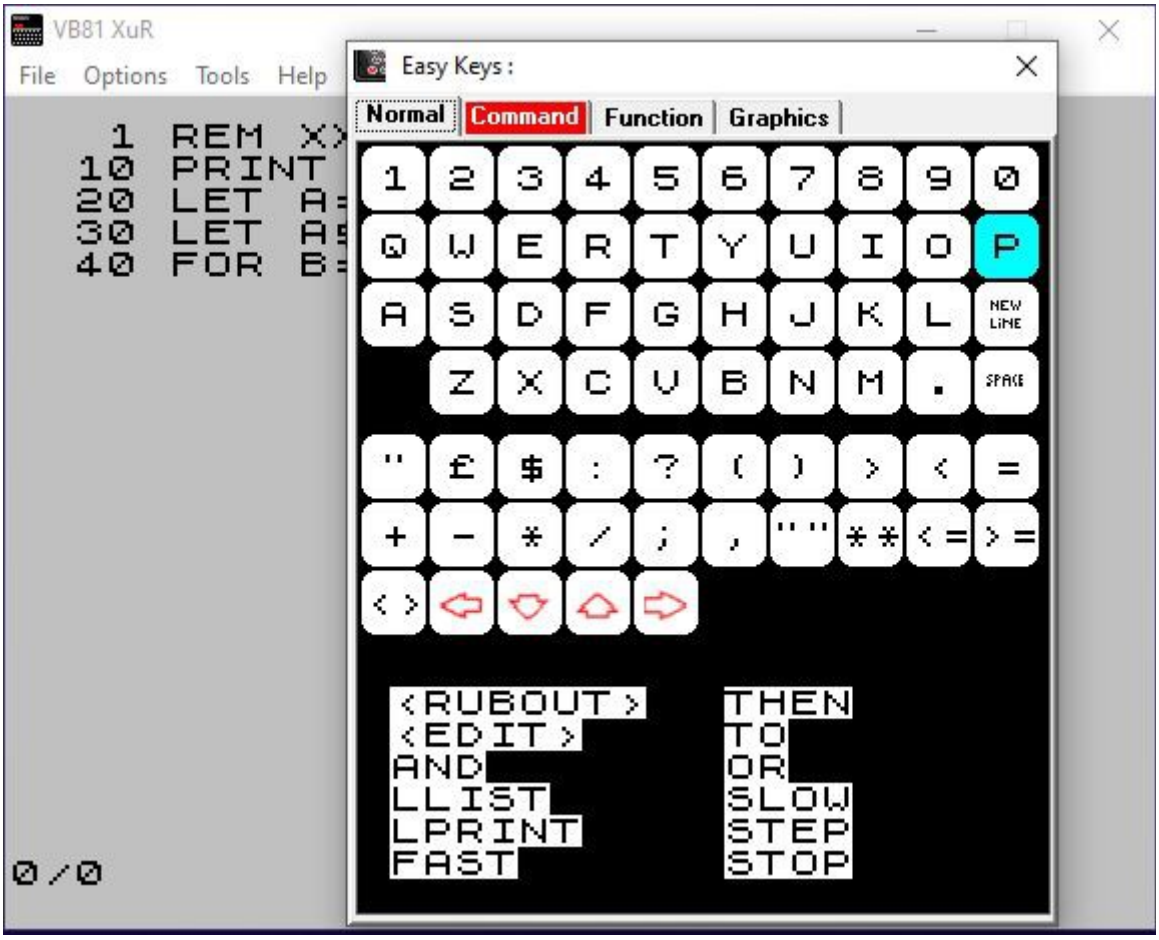
This tool can load bitmap, and retrieve a 4x4 Zx character to get an ASCII Art rendering.
Only two values are required, the upper brigh color (white or grey) and the lower contrast level.
Right click to set the lower trigger, and the Left click to set the upper trigger.



[Ok] will send the picture to the D_File, or select the saving process to keep the binary or the BASIC listing file.
You can try to select different colours by clicking left and right to improve the picture quality, and use the "EasyDraw" in the main emulator screen.

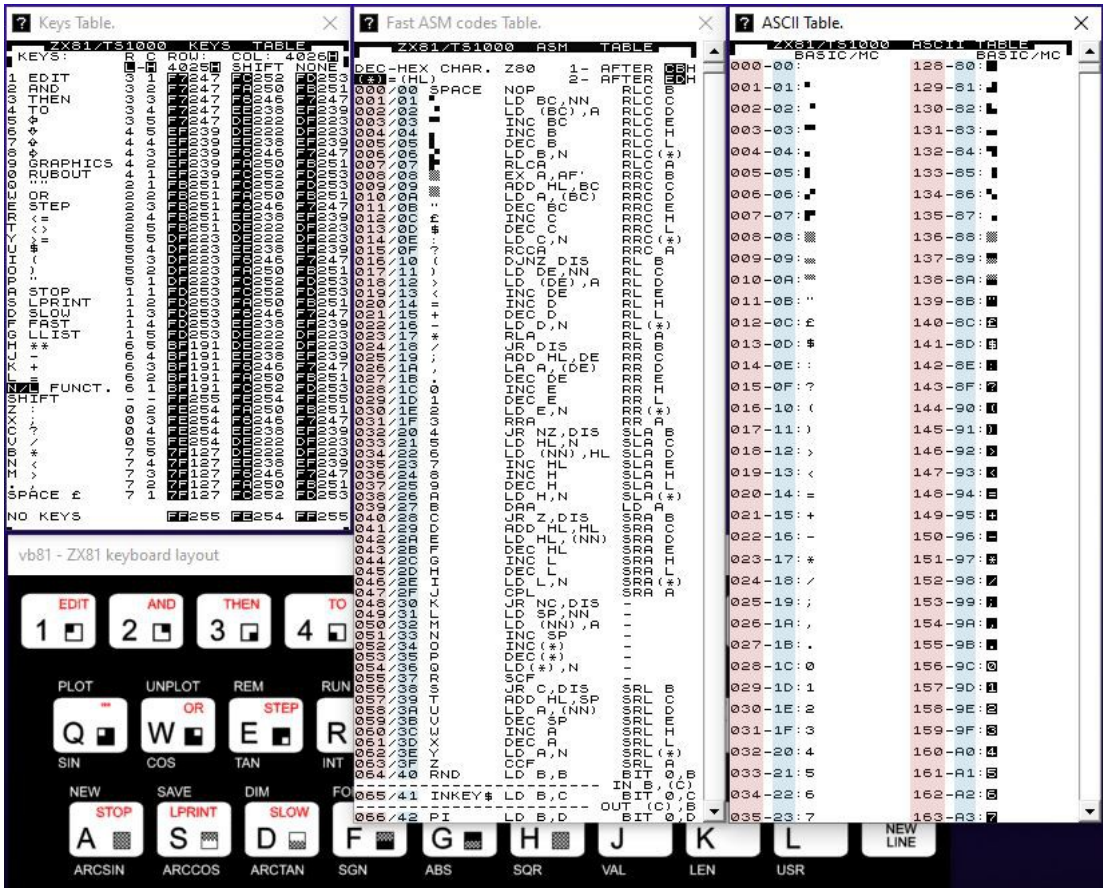
Easy Keys:

Display a keyboard with Functions, graphics chars and BASIC commands.



PS : A red tab label will be display if the table content is lock.

Help... :



ZX81 Keyboard:

Display the ZX81 keyboard.

Upper : “K” BASIC commands. (GOTO ; GOSUB ...)

Black : direct keys.

Red : shifted keys.

Bottom : functions keys, Press “shift” and “New Line” (enter) to access to this functions.

Graphics: “G” cursor, Press “shift” and “9”

To edit a BASIC line, select the line with « LIST 10 » and press SHIFT+1.
Use the cursors (left and right) to move the editing prompt.

The inverted “>” after the line number is the “READY TO EDIT” tag. It can be move with the up and down cursors keys. Note, the tag will be printed in case of a “LLIST 10”command, has “10>REM”.

ZX81 ASCII table:

Display the ASCII table used with “CHR$”.

ZX81 ASM keys table:

Display the key code table used in machine code.

About VB81... :

« About window » information about VB81 and XuR Release,

The release was fixed to « 1,30,3 » to keep Chris Cowley's original project number.

The XavSnap revision is the date (DD/MM/YY french format).

In fact, this « XuR » (XavSnap unofficial Release) was primary designed to add a screenshot in 2006!
It wasn't my intention to revise this project, but I want to preserve the excellent work of Chris Cowley.

And if this release become an “Official Release”, it will be “VB XOR”... an upside down release.

XuR release : XavSnap (Xavier)

Original release : Chris Cowley.

Special thanks to Greg (GCHarder)

Mail (XavSnap) : ZX81.VB81 § FREE.FR

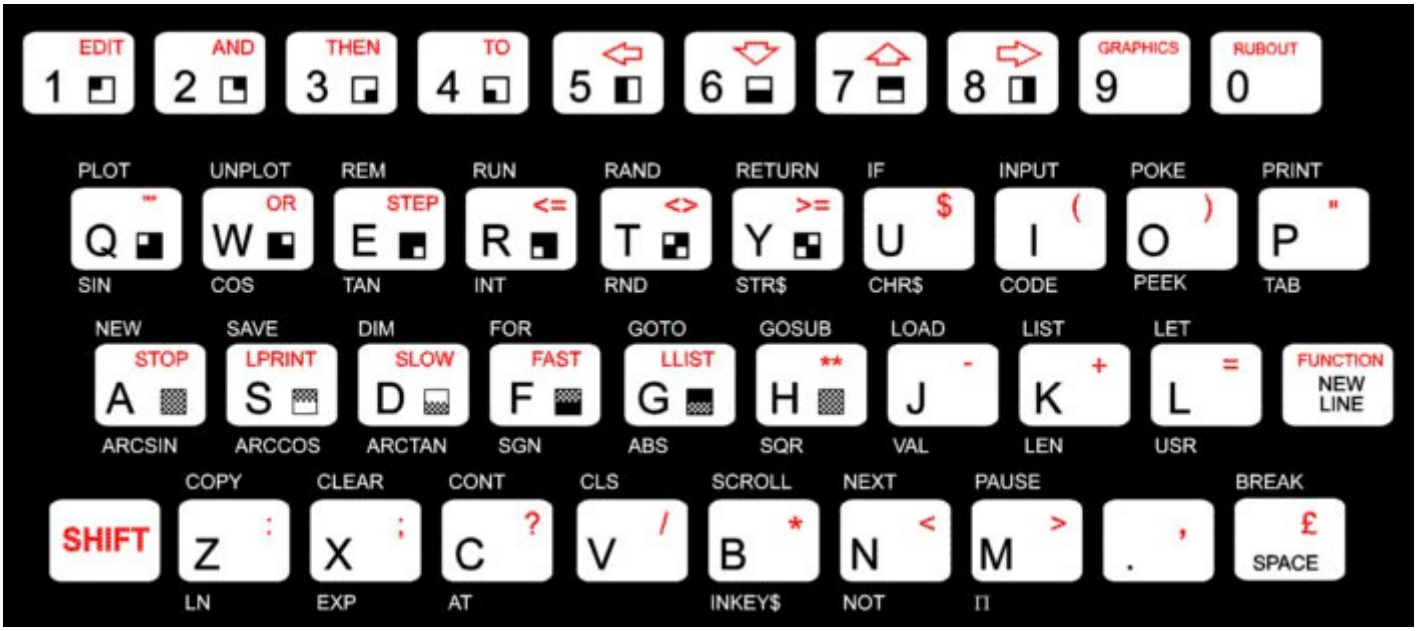
VB81's shortcut Keys :

F1 : Original ZX81 keyboard,
F2 : Show toolbar
F3 : Show Disk infos
F4 : Show Zx Printer
F5 : Show Output sound
F6 : Show Tape Manager

Display :
F5+Shift : Standard (256 x 192)
F6+Shift : Double Size (512 x 384)
F7+Shift : Triple Size (768 x 576)

Screenshot :
F11 : Capture ZX81 screen (x1)
F11+Shift : Capture ZX81 screen to ASCII format
F11+CTRL : Capture VB81 screen (GUI)
F12 : Soft Reset
F12+CTRL : Hard Reset

ZX81 Keys :



Example the « R » key:
BASIC prompt « K » = « RUN » commande.
Shifted « R » = « <= »
BASIC prompt « L » = « R »
BASIC prompt « F » (shifted New/line) = « INT »
BASIC prompt « G» (shifted « 9 »)= « \' : » (chr$(132))

How to install VB81... :

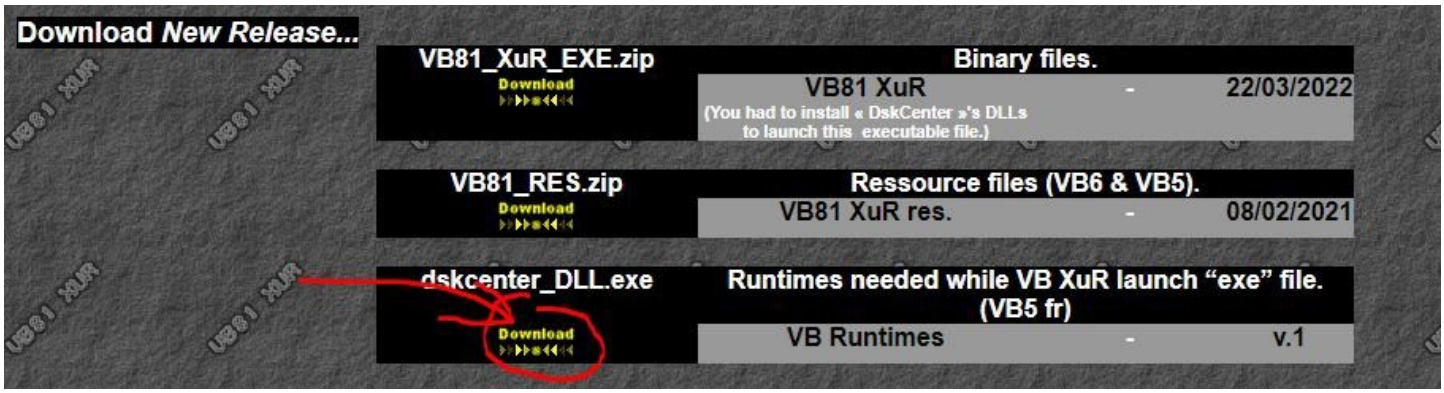
VB81 is designed to work on a 200Mhz PC under Windows 95.
It's coded in Microsoft Visual BASIC V5/6.

Tested on Windows XP-7-10-11 in 32/64bit mode.

Executables compiled on Visual BASIC need DLLs Run-time setup for all Visual BASIC stuff.

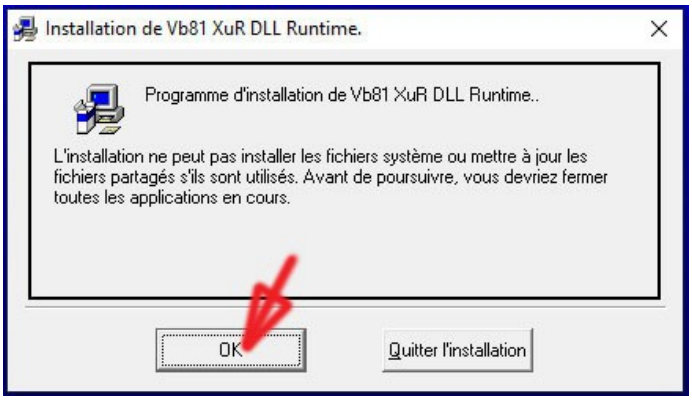
If the DLLs are not installed a message box will appear on the screen, and will point the “DLL missing”.

The DLLs pack is available on the VB81 XuR download web site.

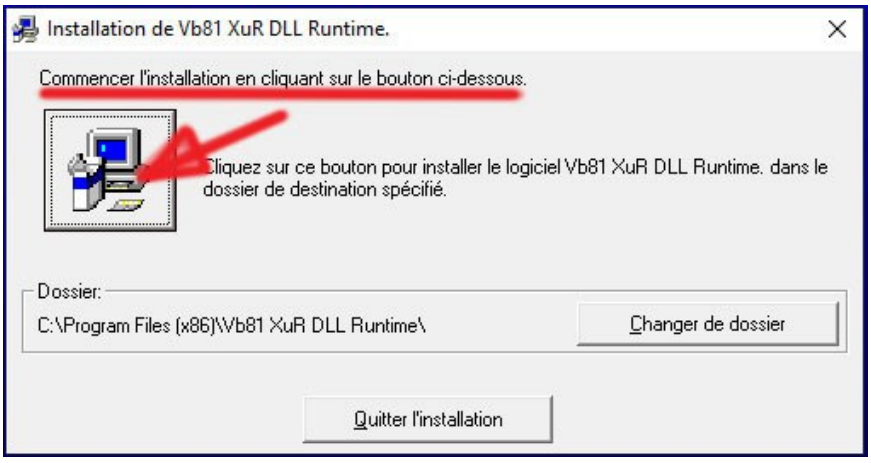


It's currently a French Visual BASIC V5 release and some VB6 OCX updates.

The installation file is also available in the “\Vb81_Setup” directory.



Click on “OK” to start the DLLs setup.



Click on the install icon to copy DLLs files in the system folders.

“Quitter installation” =” EXIT “

If DLLs are already installed, all DLLs will be preserved in case of canceled action.