

VB 81 XuR.

VB81 XuR HELP Welcome to the simple ZX81 world...



VB81 personal release.

Update: May, 2022

Page 2

Files menu :

Specity tape directory - Files explorer - BASIC text format

81 XuR.

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

	/B81 XuR			<u>(199</u> 7)	×
File	Options Tools Help				
	Specify tape & directory		2x81"		
	ASM Breaker (Escape key)		F		
	Reset ZX81 (software)	F12			
	Reset ZX81 (Hardware)	Ctrl+F12			
	Capture ZX81 screen x1 (32 col) [Auto.]	F11			
	Capture ZX81 screen to ASCII format.	Maj+F11			
	Capture VB81 screen	Ctrl+F11			
	Load Z81 file.				
	Save Z81 File.				
	Exit VB81.				
			_		
B					



Specify tape directory:

🧰 Specify Tape Directory, a tape file or "load & break" selected file.	×
Select a directory to load tape images from. Type LOAD "filename" to load a program from the selected folder (or do	uble-click on the file to load it with auto-launch feature)
🔄 SNAP 🗾 🚺 🔀 🏢 🔛	
Tape Images in this Folder:	Z81 Memory snap.
3DMAZE.Z81 ADVENT~1.Z81 ESSAI.Z81 P and 81 files (Pure binary bloc) T81 files (Merged binary bloc) TZX files (Advance tape file) SBB files (Stream of Binary Blocks)	SCORE
Wav files (Sampled PCM sounds) Memory dump files (Text Z81 memory sna	Screen memory. Tools
Rom, UDG or codes (specials Binary bloc Text Basic file (Ascii Basic programs) Image: All files All files	OK Cancel Cancel Load & Break Set Ram to 32Kb.

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 :

decode		
Tape Images in this Fold	er:	- OSM SCREENSHOT
💯 Asm_Debug.bin	Text compilator.	
IIII ASM_TEST.txt IECODE_BIN DECODE_ZX81 BA Decoder.p DEMO.P Horse.BAS HORSE.P OnePeace.BAS OnePeace.BAS IIII OP_IZX81 BASIC TI	Started. Program length:3152 Bytes Success.	5 BY GCH GREE;G 5 BY GCH GREE;G 2,00,01,05,0 5,77,06,87,0 4,58,00,00,0 3,16,77,0 3,18,00,00,0 3,16,18,81,80,01 0,10,18,83,80,01 0,10,18,83,80,01
	Save P file. Return to the Basic Editor. Cancel	Exit

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:

	\ '	\ # # 🗱
\:'	\ .	$\backslash \sim \sim$
\':	\ ' '	,
\setminus :.	\setminus .	🔪 ଜ ଜ 🎆
\setminus .:	\setminus :	∖!!
\ '	\ :	\;;; <mark></mark>
\.	\setminus . '	\backslash ::
	\'.	

<u>F</u> ont:		Font style:	Size:
ZX_81_riched		Regular	12
Trebuchet MS Verdana Webdings Wingdings	*	REGULAR <i>Calennus</i> B illi B illi	6 🔺
ZX 81 RICHED			
Zx81_Riched_ZxBasic	+		T
		ABBBY	ĭ₩Z⊠
		ABBY	MzZ
		ABBY Script: Western	°nze

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



# directory.	© DIREC	TCRY -		
#	8			
# Have Fun /XavSnap/	0 7X81.			
# 7X81 Vb81@free fr	Ö Žíonis 💼 E		881 XOR BRO	JECTS.
# ZyBasic_ZyTokon_Vb81 XuB projects	• •			
	10 RE	M DEC/HX:ZXT	EXT2P:ZXBAS:	IC (SDOCE)
#	10 RC 20 DF	M 001-01 · 0	· · · · · · · · · · · · · · · · · · ·	SHTET1
0 REM DEC/HX:ZXTEX12P:ZXBASIC	30 RE	M 002=02: 0	· · •	" SHTET +2
10 REM 000=00: " " : " " < SPACE>	40 RE	M 003=03: M	•••	" SHIFT+7
20 REM 001=01: \' : "¶" SHIFT+1	50 RE	M 004=04: 🕑.	· · · · · · · · · · · · · · · · · · ·	" SHIFT+4
30 REM 002=02·\' · "." SHIFT+2	60 RE	M 005=05: 🕑:		" SHIFT+5
40 DEM 002-02: \"	70 RE	M 006=06: M.		SHIFT+T
40 REM 003-03.1 . , SHIFT+7	80 RE	M 007=07: M: M 009-09: G0	vo	5H1F1+E
50 REM 004=04: \. : """ SHIF1+4	100 DE	M 000-00. 00	······································	" SHIFTID
60 REM 005=05: \: : "" SHIFT+5	110 RE	M 010=0A: P	· · · · · · · · · · · · · · · · · · ·	" SHIFT+S
70 REM 006=06: \.' : "»" SHIFT+T	120 RE	M 011=0B: "	: "@ØB'	
80 REM 007=07: \.' . "1/" SHIFT+E	130 RE	M 012=0C: £	: "Ē"	
00 DEM 009-09: \## . "1/" CHIET: A	140 RE	M 013=0D: "4	• " \$ "	
90 REM 000-00. WH . 72 SHIFT A	150 RE	M 014=0E: ":		
100 REM 009=09: , :"%" SHIF1+D	100 KC	M 015-0F· :		
110 REM 010=0A: \~~ : "¿" SHIFT+S	180 RE	M 017=11: "1	'n n'n	
120 REM 011=0B: " : "@0B"	190 RE	M 018=12: ">		
130 RFM 012=0C: £ "£"	200 RE	M 019=13: "«	:" : " < "	
140 DEM 013-0D- "\$" . "\$"	210 RE	M 020=14: "=	= '' : '' = ''	
	220 RE	M 021=15: "4	·:: : : : : : : : : : : : : : : : : : :	
	200 RE	M 093-17 "		
160 REM 015=0F: "?" : "?"	240 RL		с · ж	

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: OA,OB,OC]" = 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 " 1 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
- $\ll VB_XuR/ScreenShot/Ascii_Screen.txt \ \ \ \ \ file. \ All \ values \ are \ added \ in \ the \ same \ file.$
 - 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...:

VB8	81 XuR					<u></u>	×
File	Options Tools Help						
	Display size	>	LO	MY	ZX81"		
	Show toolbar	F2					
	Show Disk infos	F3					
	Show Zx Printer	F4					
	Show Output sound	F5					
	Show Tape Manager	F6					
	Colors & capture						
	Hardware Setting						
	Centronics monitor.						
	Reset AY-8912 Card.						
К							

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 photorealisme 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.





Colors:



To change ZX81's screen colours.

81 XuR.

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:

VB81 Setting :		x í
VB81		
Memory. 48 Kb Ram Pack ▼ ✓ Allow Writes to Shadow ROM (8-16K area) Actual Rom: 2x81a.ROM Audit ROMs	Static Roms Allocations (click on a free segment to install R	OM) (2 k
Rom Hack. Allow auto-setup 32k memory. Allow 32x24 'COPY' function Load wav file. Save to wav file.	16k4 AGB_Monitor@3800.rom @3000h/Start@3000h/ 12288	<u>1</u> 8k
Reset all values.	Update now Clos	e.

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:

81 XuR.

VB81 Setting :				×	
VB81					
Memory. 48 Kb Ram ✓ Allow <u>W</u> rites Actual Rom: 2x8	Pack to Shadow ROM (8-16K an 1a.ROM Audit ROM	ea)	llocations segment to install ROM		
Rom Hack.	» Rom audit.	m or		×	
Load wav fil	Files: _COD.ADR-F024 _COD.KEY TRO F	length: Crc32 2048 FE129D43 2048 AA489E	: Comments:	Î	
Reset all values.	CE85.bin CE85.bin CE85.bin CE85.bin AGB_Monitor@	4096 EB6D4C8 2048 F02C1420 2048 9974D3D	3 FLM @ \$2000 (USR C A.G.B. Centronics RC 3 AGB ROM Card "Mo	1) nit	
Add or refresh a file in the	current VB81 Database.				– 🗆 X
Main Name : [4GB_Mo Main Comment : [AGB RO]	nitor@3000.rom Sec 4 Card ''Monitor''@3000	condary Name: AGB_M/ Secondary comment:	onitor@3000.rom M	S Name: AG CRC32: le lenght:	B_Monitor@3000.rom 9974D3D9 2048 Bytes.
File comment : Merge File :					Cancel

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.

Setting :	— 🗆 X
Graphic Card.	
No H	RG CARD
None C Memotech HRG	C G007 HRG C QuickSilva HRG
UDG card. QuickSilva UDG card. AGB UDG card.	Force screen refresh mode. Inverted screen mode.
Printer output.	1 Vinder in Cal
Print to the Memotech C. I/F Centronics.	View the printer file
[10:\$2F/\$3F] ROM @ \$2800	

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.



- 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.

VB81 Setting :	8		×
			^
VB81			
Franking The Street	2-12 - IN PETRILY R	THE PARTY OF	646
Disk interface. None C Larken Dos controler Card. Auto-Update disk image.	へ Mageco controller Card. ROM Release : つ 2.6 0	3,0	
– Larken Disk (LkDos)		in i	
Drive 0 : Disabled 📃 No Drives.			
Drive 1 : Disabled 📃 No Drives.			
Speed		14 16	0.0%
* *	to 600%. But this change can	speed from 10 cause typing	JU/6
	problems. Use it only during the phase whitout using keyboard.	e calculation	
Development			
Heset all Values.	Update now	Close.	

Floppy disk interfaces:

81 XuR.

- 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 $\ll \text{DIRE} \gg \text{for DIREctory}$

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:

<u>Site AbandonMagazine d'Ordi-5.- Test de la carte contôleur MAGECO pour votre ZX81 et VP100</u> - (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.

V <mark>B8</mark> 1				
Cursors. • 5,6,7 and 8	- 0 5,6,7 and 8 [Shifted]	C User defin	edDefi	ne
-Joysticks AGB dual Joysti Kempston joysti -Light-Pen □ VTR/MAGI	ck @ 10: EFh_DFh. (cursors+'0') ck @ 10: 1Fh. (cursors+'Ctrl') ECO Light-Pen. [10 IN h7D ; 1N h7F]		
- Input/Output c 1/0 OFF - No In	ards. put/ouput device installed			-
Sound Device. VTR/ZON-X	(Set to mute : Sound error detected Sound Card.	IP [AY-3-8912 / IO:DFI @ IN/OUT Ports. Registers: &hDF	n-OFh] C PEEK/PC Datas: &hOF	IKE.
Reset all values.	Γ	Lindate now	Clo	se.

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.

81 XuR.

Sound card:

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



Tools... :

VB81 XuR					3-8		×
File Options	Tools Help						
BEZE I	Disassemble			8 💌 🖥	?	ША	<i>#</i>
	Tiny "TASM" Z80 Assembler						CONTRACTOR CONTRACTOR
	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.

81 XuR.



- (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.
- (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 :

🏙 Tiny Inboard TASM Z80 assembler.		-		Disasser	nble:			- 🗆 X
Load Save Paste clipboard Help [][] ∧ Text ASM opcodes :	Translated opcodes :	Labels :		Option HEX: 8000	to 81F4) 🖡 🗾 🔼 🛛		en e
Compile this file using: ************************************	Line: Opcodes:	Label PUTBYTE GETBYTE DISPLAY DISPLAY2 DISPLAY3 DISPLAY3 DISPLAY4 DISPLAY5	Values: 0 8 16 19 28 32 45	Address 8000 8003 8005 8007 8008 8008 8000 8010 8010 8013 8014 8016 8014 8016 8014 8012 8020	Opp.Code(H) FD;7E;7B D3;FD 18;09 C9 DB;FD FD;77;7C 18;01 C9 01;00;08 1F 38;06 FD;36;3C;B4 18;04 18;04 FD;36;3C;F4	Mnemonics LD A. (IY+123) OUT (\$FD).A JR +10 RET IN A.(\$FD) LD (IY+124).A JR +2 RET LD BC,\$0800 RRA JR C.+7 LD (IY+60).\$E4 JR +5 LD (IY+60).\$97 INC IY DUN7_16	Comments ZX81 NM [\$8010: User port. [\$8010: [\$8010: [\$8020:	T 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
Error log. : 51 bytes Lenght. *** Code compiled. *** Assemble codes Send to Disassembler offset : \$80	000 Send to: USER			8022 8028 8028 802D 802E 8030 8032 8033 8034 → HL (HL)	10, pr FD; 21; 00; 40 21; 43; 40 06; 08 7E D7 2B 10; FB C9 76 BE DE BC (BC) (BC)	LD IY, \$4000 LD IY, \$4000 LD HL, \$4043 LD B, \$08 LD A, (HL) RST 10 DEC HL DJNZ -4 RET HALT CP (HL) AF	Display [\$802D: c IR X SP	

External processors:

Output text: [HEX:] ▼ Start: \$4082 End: \$4008 USER Pause: 1 sec. Execut 65-128 0 0 -	PuCrunch	 pucrunch.exe 	Pucrunch ©1997	-2008 by Pasi 'Alberi	ť Ojala.		
55-128 0 0 0	Output text : [HEX:]	✓ Start: \$4082	End: \$40D8 USEF	•	Pause : 1 sec.	Execut	e j
Z/7 rescan gained 0 bytes Iash Checks 29 (29, 100.00% equal), RLE/LZ compares 840 Compressed 87 bytes in 0.00 seconds (84.96 kB/sec) # VB81 XuR [] # VB81 XuR [] # Empty Basic Segment. # Length : 26 # External: PuCrunch REM [HEX:\ 10,02,70,75,B9,01,01,58,02,01,08,80,00,FF,FF,01,\ 10,BF,59,DA,EB,F5,58,F9,FF,F8]	5-128 0 0 0 29-256 0 0 -	į.					-
# VB81 XuR [] # Empty Basic Segment. # Length : 26 # External: PuCrunch REM [HEX:\ ;0,02,70,75,B9,01,01,58,02,01,08,80,00,FF,FF,01,\ ;0,BF,59,DA,EB,F5,5B,F9,FF,F8]	∠// rescan gained 0 by lash Checks 29 (29, 10)	ytes 0.00% equal), RLE/LZ c 0.00	compares 840				
# VB81 XuR [] # Empty Basic Segment. # Length : 26 # External: PuCrunch REM [HEX:\ 30,02,70,75,B9,01,01,58,02,01,08,80,00,FF,FF,01,\ 30,02,70,75,B9,01,01,58,02,01,08,80,00,FF,FF,01,\	compressed 87 bytes in 1	U.UU SECONOS (84.36 KB	/secj				
# External: PuCrunch REM [HEX:\)0,02,70,75,89,01,01,58,02,01,08,80,00,FF,FF,01,\)0,BF,59,DA,EB,F5,58,F9,FF,F8]	#VB81 XuR [] #EmptyBasic Segment. #Length : 26						1
REM [HEX:\)0,02,70,75,B9,01,01,58,02,01,08,80,00,FF,FF,01,\)0,BF,59,DA,EB,F5,5B,F9,FF,F8]	# External: PuCrunch						
	REM [HEX:\ 10,02,70,75,89,01,01,58,02,1 10,8F,59,DA,EB,F5,58,F9,FF	01,08,80,00,FF,FF,01,\ ;F8]					



Binary editor:

Index	00 01 02 03 04 05 06 07	08 09 OA OB OC OD OE OF	ASCII
4000	FF.01.FC.BF.00.C0.00.00	00.00.0A.00.92.40.AB.43	0 0 0 0 0 0 0 0 0 0
4010	AB.43.00.00.AC.43.AE.43	00.00.AE.43.AE.43.FF.5D	
4020	40.00.02.00.00.FF.FF.OF	37.00.00.00.00.00.00.00	• • • • ?R
4030	6B.0C.00.00.CB.D4.00.00	BC.21.00.C0.00.00.00.00	of 00 .5 0
4040	00.00.00.00.00.00.00.00	00.00.00.00.00.00.00.00	6m, 6m, 6m, 100
4050	00.00.00.00.00.00.00.00	00.00.00.00.76.84.20.00	₽ 4
4060	00.00.00.84.00.00.00.84	A0.00.00.00.00.00.00.00	- 4
4070	00.00.00.00.00.00.00.00	00.00.00.00.00.00.0A.11	****)
4080	00.F5.0B.2D.2A.31.31.34	00.32.3E.00.3F.3D.24.1D	•"HELLO MY ZX81
4090	0B.76.76.00.00.1D.1C.92	35.37.2E.33.39.00.0B.2D	" 44 10>PRINT "H
40A0	2A.31.31.34.00.32.3E.00	3F.3D.24.1D.0B.00.00.00	ELLO MY ZX81"
40B0	00.00.00.76.00.00.00.00	00.00.00.00.00.00.00.00	e e
40C0	00.00.00.00.00.00.00.00	00.00.00.00.00.00.00.00	
40D0	00.00.00.00.76.00.00.00	00.00.00.00.00.00.00.00	ų
40E0	00.00.00.00.00.00.00.00	00.00.00.00.00.00.00.00	
40F0	00.00.00.00.00.76.00.00	00.00.00.00.00.00.00.00	4
4100	00.00.00.00.00.00.00.00	00.00.00.00.00.00.00.00	Cash.
4110	00.00.00.00.00.00.76.00	00.00.00.00.00.00.00	ų
4120	00.00.00.00.00.00.00.00	00.00.00.00.00.00.00	1000 at 100 a
4130	00.00.00.00.00.00.00.76	00.00.00.00.00.00.00.00	÷.
0	→ 400Ah/16394	: 0Ah / 10	- 0000b 2000b 4000b D. File VABS 🏥

ZX81 memory in hexadecimal/decimal values.

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

Binary Edit	tor.																	¢
Index	00	01	02	03	04	05	06	07	08	09	0A	OB	OC	OD	OE	OF	ASCII	
40A0	2A	31	31	34	00	32	3E	00	3F	3D	24	1D	0B	00	00	00	ELLO MY ZX81"	-
40B0	00	00	00	76	00	00	00	00	00	00	00	00	00	00	00	00	4	
40C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	the little second se	
40D0	00	00	00	00	76	00	00	00	00	00	00	00	00	00	00	00	4	
40E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
40F0	00	00	00	00	00	76	00	00	00	00	00	00	00	00	00	00	4	_
4100	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
4110	00	00	00	00	00	00	76	00	00	00	00	00	00	00	00	00	4	
4120	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
4130	00	00	00	00	00	00	00	76	00	00	00	00	00	00	00	00	4	
4140	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
4150	00	00	00	00	00	00	00	00	76	00	00	00	00	00	00	00	÷.	
4160	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
4170	00	00	00	00	00	00	00	00	00	76	00	00	00	00	00	00	ų.	
4180	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
4190	00	00	00	00	00	00	00	00	00	00	76	00	00	00	00	00	4	
41A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
41B0	00	00	00	00	00	00	00	00	00	00	00	76	00	00	00	00	4	
41C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	10 P.	
41D0	00	00	00	00	00	00	00	00	00	00	00	00	76	00	00	00	÷.	-
			à 6	H	÷	40 3E	A5h/ 32h /	16549 / 1592	1: 32h 2	/ 50		Γ		H	USE	R	👻 0000h 2000h 4000h D_File VARS 踏	

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:

81 XuR.

- (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



Page 29

D_file tools:

The D_file is the display memory segment.

D_File tools	>	Save current D_File memory bloc.[Binary,Text PRINT listing]	Ctrl+S
Reset memory	>	Load or Call back a saved D_File memory bloc. [Binary,P files]	Ctrl+D
Memory Tools [#]		Save compress D_File in a text ASM file.[Auto-Save in Prj. Dir]	
Compare Memory	>	Clear current D_File memory bloc.	Ctrl+F

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.



You can use this feature to launch the ASM2 program in 32kb mode.



Memory Tools:

	Load an external file into memory.
Memory OFFSET: 0	USER
Alow to Load a binary file Roms, screen capture, ch) fill a memory bloc into the Zx81/TS1000 memory. racters map, asm codes)

Can't find any REMs in the BASIC memory bloc !

2		
? Alow to fill a memory blo (the lenght of selected fil	c directly in an existing REM Line. le will be trimmed to be adjust to the rem line length)	
ow to fill a memory blo e lenght of selected fil	oc directly in an existing REM Line. le will be trimmed to be adjust to the rem line length)	

Close. Execute...

OFFSET : 0 ? LENGHT : 0 ? Move UDG buffer to ? Alow to Copy a memory bloc to another offset. NEW OFFSET : 0 ?	File to memory. No R	f lines to fill. Move memory to memory (LDIR)	ave memory bloc	c
OFFSET : 0 ? LENGHT : 0 ? Move UDG buffer to (to copy the ROM char. map to 8192, a REM content to RamTop or a ROM/D_file in a REM line.) NEW OFFSET : 0		Copy a memory bloc to a new offse	et.	
	OFFSET : 0 LENGHT : 0 Move UDG buffer NEW OFFSET : 0	Alow to Copy a memory bloc to another offs (to copy the ROM char. map to 8192, a RE ROM/D_file in a REM line.)	et. M content to R	amTop or a

File to memor	y. No REM	l lines to fill. Hove memory to memory (LDIR)	Save memory bloc	
		Save a memory bloc on your hard	disk.	
OFFSET :	0	Alow to Save a memory bloc on your har	d disk.	
LENGHT :	0	(save an Oser Derinable Graphics, a par	t of D_file, the Larki	en burrer j
REM line:	USER	<u> </u>	Retrive ASM2 mad	chine code line.
			Close	Evecute

VB 81 XuR.

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:

			10	
Compare m	iemory snap re	esult.	×	
5na Des	adow memory	(2000-400)		Save a memory snapshot for later comparison
Progra	gram uatas.(« m Variable«·I	Yare-Ram	Topl	Suve a memory snapshot for later comparison.
Bam.	Top datas:[B	am Top-RF	- CPJ	Example: During a game your lives left credit is "3"
All		am_10p-br		Example. During a game your rives left creat is 5.
Address : 4034h/16436 4035h/16437	Old value: 2Eh / 46 FBh / 251	New value: C1h / 193 FAh / 250	Zx Chars:	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.	Renume basic lines.	Merge another P file.	Programs Optimization.	Create a text file
	Add a R	EM line in your Bas	ic interpreter.	
	1 Lenght :	1 Filler : 3D	Hex.	
Allow to create (All line number	a REM line to import a between 0 and 9999 o racters, the display ca	machine code segment an be displayed while a n reset 'Crash' the Basic	LIST, :editor.while a LISTI)	Maxi.: 16k:15350 32k:21600

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 Tools. (This tools can crash you ZX81, please save your	code first !)	×
Add a REM line. Renume basic lines. Merge another P file. Progra	ams Optimization. Create a	text file
Renume Basic program.		
Line Step : 10 Convert numerics values to VAL''xxxx'	" or CODE"x" string format.	
Allow to Renume the Basic program loaded in Zx81 memory. (GOSUB and GOTO will be changed to suite to the correct jump, but you had to change atypicals calls like 'GOTO a*1000', 'GOSUB P	ť)	
Each tool will reset the variables memory like a CLEAR function	Close. E:	kecute

Basic Tools. (This tools can crash you ZX81, please save your code	first!)	×
Add a REM line. Renume basic lines. Merge another P file. Programs Op	otimization. Cr	reate a text file
Merge a Basic program after present co	odes.	
Allow to merge another Basic program. It will loaded after your Basic program memory. You had to load a basic program using upper lines numbers to avoid numbers, or renume it with the inboard Renume function.	already loaded redundants an	d in Zx81 id attics lines
Each tool will reset the variables memory like a CLEAR function.	Close.	Execute

Page 3	33
--------	----

Optimize Basic program code	5.
Change all numerics values to characters codes.(CODE, VAL)	Allow to Save memory
Ignore GOTO-GOSUB Commands. I Ignore PRINT AT Comma I Ignore LET Commands.	inds.

Add a REM line. Renume basic lines. Merge another P file. Programs 0	ptimization. Create a text file
Save a Basic program in a text file	
NewTextProgram.TXT Options ZxBasic format (Ascii specials charset) ✓ Codes : Regular Codes	Allow to Save Basic memory in a text file, include ASM codes, or type SAVE'PROG.TXT'

BASIC variables window:

VB 81 XuR.

e Options	Tools Help		
10 F 20 L 30 F	RINT "HELLO MY ZX81" ET A=1 ET A\$="HELLO" OR B=3 TO 5		
Basic	/ARiables.	x	
	Variables: Values		
	A = 1 A\$= "HELLO" B : For-Next B = 3		
	A = 1 As= "HELLO" B : For-Next B = 3	_	
	A = 1 As= "HELLO" B : For-Next B = 3	-	

Display a window showing all BASIC variables and their values.



Type-in window:

	otions To	ols	Help		en elema						
1		EM	$ X \times X$	(XX)	\mathbb{C}	$\propto \propto$	XXXXX MV 4Ve				
ź	0 LE	ΞŤ	A=	=1		-0	nr 270	э т			
- 3	ΘĒĒ	ĒŤ	i As	₅⊒…⊢	IEL	LLO.					
4	0 F0	DR	: B=	•з т	0	5					
ype-in	window :										
Hexa:	Dec:		Hexa:	Dec:		Get/Re	fresh Mnemonics	Disp	lay/Refesh Asm	listing	Ą
407E	16510	-	01	1		4082	3D	DEC	A		
407F	16511		10	16		4083	3D	DEC	A		
4080	16512		00	0	F	4084	3D	DEC	A		
4081	16513		EA	234	0	4085	3D	DEC	A		
4082	116514<		30	61	H	4086	ЗD	DEC	A		
1083	16515		30	61	H	4087	ЗD	DEC	A		
4003	10510		30	C1	님	4088	3D	DEC	A		
1084	16516		30	61		4089	3D	DEC	A		
4085	16517		3D	61		408A	3D	DEC	A		
4086	16518	-	3D	61	\times	<					>
	M 14 Bu	tes@	4082-408	3F		-					100
1 RE											

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

VB81 XuR [5	STORE.P]				Zx Printer	
File Options	Tools Help				SAVE RESET	•
8976	REM 5 (RNDLN	RND6) : RNI AN	()) ())) ()))	5£RND 16578 : 123,7,0,0,0,0,0,0,1,135,14,1 16579 : 133,6,129,11,134,15,136 16579 : 133,6,129,11,135,14,1 16597 : 132,7,1128,0,0,0,0,0,141,164,1 16570 : 132,7,1128,0,0,0,0,0,0,0,0,0,0 16706 : 35,11,140,1,55,141,164,1 16706 : 35,11,140,1,55,141,164,1 16706 : 35,11,140,1,55,141,164,1 16706 : 32,16,0,20,167,23,34,14,64,23,1 16707 : 142,64,126,00,167,23,10,34,14,16,16,20 16707 : 147,29,0,25,14,14,164,16,20 16779 : 126,14,14,64,20,147,14,14,14,14,14,14,14,14,14,14,14,14,14,	L31,2 ,3 ,4 ,2 ,4 ,3 ,3 ,3 ,3 ,3 ,3 ,3 ,3 ,3 ,3 ,3 ,3 ,3
SUB ?	Type-in window	- 1 1636 :		DE.		
7?7 E	Hexa: Dec :	Hex	a: Dec:		Get/Refresh Mnemonics Display/Refesh Asm listing A	
DSGN	4118 16664	A 04	10		16668:42,7,0,0,0,0,0,135,4,131,2 ^	
6 RND	4119 16665	81	143		16679:133,6,129,1,134,5,130,3	
11	411A 16667	FΔ	234	H	16697:35.1.140.1.58.141.64.254.1	
	411C <1 16668	< 2A	42	F	16706:32,6,0,237,74,34,14,64,33	
	411D 16669	07	7		16715:142,64,126,167,23,23,216	
	411E 16670	00	0		16722:23,22,0,203,18,95,33,0,30	
	411F 16671	00	0		16731:25,14,4,6,4,86,35,94,35	
	4120 16672	– 00	0		16747.23 203 19 23 203 19 23 23	
	10 REM: 141	Bytes@411C	-41A8		▼	

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 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 next line if his length is too large.



UDG TOOL:

81 XuR.



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

File Options Tools Help SCORE +88989 #HI - SCORE +8889 #FROGS +33 LEUEL +1 State State <t< th=""><th>B81 XuR [P.P] SAVE"P" Line 1</th><th>_ 🗆 🗙</th></t<>	B81 XuR [P.P] SAVE"P" Line 1	_ 🗆 🗙
SCORE→88888#HI-SCORE→88888#FROGS→3 LEUEL→1 1 1 2 2 2 2 2 2 3 3 2 3 4 4 5 4 5 3 4 4 5 4 5	File Options Tools Help	
SCORE→60000 HI - SCORE→60000 HI ILEUEL→11		
1 2 XUDG factory. X X X X X X	SCORE→0000円HI-S	CORE→0000ĂFROGS→3
1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 1 2 2 1 1 1 2 1		
Image: Second		ZxUDG factory.
★★★★★★★ ★★★★★★★ ★★★★★★★ ★★★★★★★ ★★★★★★↓ ★★★★★↓ ★★★★★↓ ★★★★★↓ ★★★★★↓ ★★★★★↓ ★★★★★↓ ★★★★★↓ ★★★★★↓ ★★★★↓ ★★★★★↓ ★★★★↓ ★★★★↓ ★★★★↓ ★★★★↓ ★★★★↓ ★★★★↓ ★★★★↓ ★★★★↓ ★★★★↓ ★★★★↓ ★★★↓ ★★★★↓ ★★★↓↓ ★★★↓ ★★★↓↓ ★★★↓↓ ★★↓↓↓ ★★↓↓↓ ★↓↓↓↓ ★★↓↓↓ ★↓↓↓↓ ★↓↓↓↓ ★↓↓↓↓ ★↓↓↓↓ ★↓↓↓↓↓ ★↓↓↓↓ ★↓↓↓↓↓ ★↓↓↓↓↓ ↓↓↓↓↓ ★↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓↓		
Statut IR memory segment [20] ▼ Statut Statut Statut <thstatut< th=""></thstatut<>	- Level and level	
C: C: <t< th=""><th><u>44.44</u></th><th>IR memory segment [20]</th></t<>	<u>44.44</u>	IR memory segment [20]
\$\frac{1}{2}\$	~ ~ ~ ~ ~ ~	20 H Bank #8-1
B07:FFh::255 Current character: Curren	***** *****	B00:92h:146 B01:FFh:255 B02:24h:36 B03:FFh:255 B04:49h:73 B05:FFh:255 B06:92b:146
Current character : ·	<i></i>	B07:FFh :255
● ●		Current character :
○ ○	දියාංක	<u> </u>
456789ABCDEFGHIJ	*****	
4007096060616610		
KLMNOPQRSTUVUXYZ		(LMNOPQRSTUVUXYZ

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... :

? Keys Table. X	? Fast ASM codes Table.	? ASCII Table.	×
ZX81/TS1000 KEYS TABLE KEYS: B C BON: COL: 4026H.	ZX81/TS1000 ASM TABLE	AX81/151000 HSCH HBLE BASIC/MC BASIC/M	10
1 EDIT 3 1 19247 19252 19253	DEC-HEX CHAR. Z80 1- AFTER BH	000-00: 128-80:	- F
2 AND 3 2 7247 74250 78251 3 THEN 3 3 7247 68246 7247	000/00 SPACE NOP RLC B	001-01: 129-81:	
4 TO 3 4 7247 5238 54239 5 4 3 5 7247 5222 5523	1002/02 LD (BC),A RLC D	002-02: 130-82:	
6 4 5 5 239 5 222 5 223 7 4 4 5 239 5 238 5 239		003-03: 131-83:	
8 ¢ 4 3 5 239 5 246 7 247 9 Graphics 4 2 5 239 5 250 5 251	006/06 LD B,N RLC(*)	004-04: 132-84:	
0 RUBOUT 4 1 EF239 F0252 F0253 0 "" 2 1 F8251 F0252 F0253	008/08 EX A, AF' RRC B	005-05: 133-85:	
W OR 2 2 58251 58250 58251 E STEP 2 3 58251 56246 57247	010/0A H LD A, (BC) RRC D 011/08 DEC BC RRC E	006-06: 134-86:	
R (= 24 8251 8238 8239 T <> 25 8251 8222 8823	012/0C £ INC C RRC H	007-07: 135-87:	
Y)= 55 DF223 DF222 DF223 U \$ 54 DF223 EF238 EF239	014/0E: LD C,N RRC(*) 015/0F ? RCCA RRC A	008-08: 136-88: 1	
I (5 3 DF223 F6246 F7247 O) 5 2 DF223 F6250 F6251	016/10 (DJNZ DIS RL B 017/11) LD DE,NN RL C	009-09:*** 137-89:**	
Р " 5 1 07223 10252 10253 А STOP 1 1 10253 10252 10253	018/12 > LD (DE),A RL D 019/13 < INC DE RL E	138-8H:	
S LPRINT 1 2 60253 60250 68251 D SLOW 1 3 60253 66246 67247	020/14 = INC D RL H 021/15 + DEC D RL L	011-08:" 139-88:	
F FAST 1 4 10253 10236 10239 G LLIST 1 5 FD253 DE222 DF223	022/16 - LD D,N RL(*) 023/17 * RLA RLA	012-00:£ 140-80:B	
H ** 6 5 67191 07222 07223 J - 6 4 67191 07238 07239	024/18 / JR DIS RR B 025/19 ; ADD HL,DE RR C	013-0D:\$ 141-8D:E	
K + 6 3 BF191 F8246 F7247 L = 6 2 BF191 F8250 F8251	026/1A, LA A, (DE) RR D 027/1B. DEC DE RR E		
HIFT FR255 FE254 FF255	028/10 0 INC E RR H 029/10 1 DEC E RR L	015-0F: ? 143-6F: M	
Z: 021254125013251 X: 03122541624617247	030/1E 2 LD E,N RR(*) 031/1F 3 RRA RR A		
	032/20 4 JR NZ,DIS SLA B 033/21 5 LD HL,N SLA C		
	034/22 6 LD (NN),HL SLA D 035/23 7 INC HL SLA E		
7 2 7 127 FA250 FB251	036/24 8 INC H SLA H 037/25 9 DEC H SLA L		
PHCE E / 1 Mai2/ 20252 20253	038/26 A LD H,N SLA(*) 039/27 B DAA LD A		
	041/29 D ADD HL,HL SRA C		
vb81 - 7X81 keyboard layout	042/2H E LD HL, (NN) SRH D 043/2B F DEC HL SRA E		
1561 Elor Reyboard Rybac	044/20 G INCL SRH H 045/20 H DECL SRA L		
	046/2E I LD L,N SRH(*) 047/2F J CPL SRA A		
EDIT AND THEN TO	049/31 L LD SP,NN -	205 10: 154 00:	
1 . 2 . 3 . 4 .	051/33 N INC SP -	207 1P: 154-9A.M	
	052/34 0 INC(*) - 053/35 P DEC(*) -		
PLOT LINPLOT PEM PLI	055/37 R SCF -		
	057/39 T ADD HL,SP SRL C	029-10.1 157-90.M	
	059/38 V DEC SP SRL E	000-12:2 150-52:0	
	061/30 X DEC A SRL L	032-20:4 160-00:5	
SIN COS TAN INT	063/3F Z CCF SRL A	033_01'5 161_01'F	
NEW SAVE DIM FO	004/40 RND LD D,D D11 0,B	034_22:6 162_02:B	
STOP LPRINT SLOW	000/#1 INKET\$ LO B,C BIT 0,C	035_03-7 163_02-8	
		NEW NEW	
		L	
ARCSIN ARCCOS ARCTAN S	GN ABS SQR VAL LEN	USR	

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)



Page 41

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.

🛃 Installation de Vb81 XuR DLL Runtime. 🛛 🗙	Click on "OK" to start the DLLs setup.
Programme d'installation de Vb81 XuR DLL Runtime L'installation ne peut pas installer les fichiers système ou mettre à jour les fichiers partagés s'ils sont utilisés. Avant de poursuivre, vous devriez fermer toutes les applications en cours.	
🛃 Installation de Vb81 XuR DLL Runtime.	×
Commencer l'installation en cliquant sur le bouton ci-dessous.	. Runtime. dans le Click on the install icon to copy DLLs files in the system folders. "Quitter installation" =" EXIT "
Dossier: C:\Program Files (x86)\Vb81 XuR DLL Runtime\hang	er de dossier If DLLs are already installed, all DLLs will be preserved in case of canceled action.
Quitter l'installation	