

HP - 4122USB

82143A Simulator



## **Contents**

Introduction.....	:	3
Hardware description.....	:	4
Software description.....	:	5
Port configuration.....	:	6
Controls.....	:	7
Printing commands output.....	:	8
Apéndice A.		
Interface info for developers:.....	:	10

## Introduction:

Although many devices have been developed along the years for the HP-41 system, most of them were dedicated to RAM and/or ROM memory extensions. From the original MLDL to the later MLDL2000 (by Mendert Kuipers) and my own Clonix/NoV's series.

Special mention to the development of the 41CL (by Monte Dalrymple) which not only includes a wide range of memory enhancements and expansions but also a serial interface.

Apart from this, previous interfacing with HP-41 system have been based on proprietary devices or the HP-IL loop protocol and its interconnecting devices: HP-IB, RS-232, 82973A (ISA), etc. Which limited the type of devices and PC's allowed to operate in such environment.

PIL-Box (by J-F Garnier) is a later development much more improved which allows USB to HP-IL interconnection.

The HP41<>USB interface presented here is intended to provide users with a direct link between the HP-41 and any current PC, without the need of intermediate devices nor dedicated interface ROM image.

The first application available for this new interface device is the 82143A Peripheral Printer simulation. However, the USB interface has been designed as a tool to create a whole range of Input/Output capabilities between current PC's and our beloved vintage 41's systems.

One of the keystones in this developments is the use of the standard ROM's for the simulated task. So, the 82143A Printer simulation presented in here, only needs of the -PRINTER-.ROM to work, and such -PRINTER- ROM image is also included in the HP41<>USB interface.

The microcontroller inside the module has therefore two function: first it behaves (and can be used like) a Clonix module, reserving page #6 for the -PRINTER- ROM; and second, it traps any communication between 82143A Printer and the HP-41 system, simulating that a real printer is present, and sends relevant data to the USB interface in serial mode.

Current printer implementation has some known limitations:

- It is permanently set to MAN mode. TRACE or NORM mode printing is not allowed.
- Graphical features are not included, only text printing is simulated.
- Some of the special characters in the 82143A set has been replaced, see Table 1.

Internal design of the HP41<>USB interface allows development of future applications including bidirectional data transfer. So far only unidirectional transfer is implemented; page transfer from HP-41 to PC .ROM file has also been successfully tested using the COPYROM command from HEPAX module.

## **Hardware description:**

From a HW point of view the HP41<>USB interface module is divided into two main subsystems:

- HP I/O interface & Clonix (ROM) simulation.
- USB serial interface.

A modified version of the Clonix-D device has been chosen to provide the ROM simulation part of the interface.

It can be re-configured using the ConixConfig.exe utility (ver. 3.0)\* like any other Clonix or NoV module.

In its current status, only 82143A emulation is developed so, there will be no other available configurations when 41<>USB module is selected.

User can select up to 8 more pages to be included into the module so that the whole range of port addressable memory plus 82143A simulation can be filled with a single module port usage.

On the USB part of the device, care has been taken in choosing the same RS-232<>USB driver already used by the USB Programmer (Prolific interface) therefore there will be no need for the users to install new drivers on their PC's

Serial interface for this simulation has been set to the following specs:

- Speed: 115200 bps
- Data bits: 8
- Parity bits: None
- Stop bits: 1
- Handshake: None

Make sure that your PC port used to interconnect with the HP41<>USB interface is set to the above configuration.

You can find the Serial Port list of your PC in: Control Panel > Device Manager. Once there just plug the USB connector to your PC and you'll see a new Prolific COMx port under Ports (COM & LPT). Please take down the number "x" assigned to your port as it may be necessary to configure the software interface.

\*Current ClonixConfig.exe (ver. 2.1) does not provide reconfiguration tools for the HP41<>USB module. Ver. 3.0 will be released along with the HP41<>USB interface.

## Software description:

Since it's a personal project and hence constrained by my limited skills, the software utility provided inhere, USB-82143A.exe, is a simple GUI written in .NET Visual Basic. Appendix A provides details for any skilled programmer to build suitable solutions on their preferred platform.

Present utility it's a modified sample program from Windows VB library. Some functions have been added to allow file saving and hardcopy printing.

Main Window (Fig. 1): Shows “CHARS” program listing and DISASMBling of “PRA” command.

The screenshot shows a Windows application window titled "USB 82143A". At the top, there are four buttons: "Port Settings", "Clear", "Save", and "Print". Below these buttons is a scrollable text area containing assembly-like code. The code starts with a label "01♦LBL "CHARS"" and continues with various instructions and data. At the bottom of the text area, there is a status bar displaying "COM1 115200 N & 1 Handshake:None".

```
01♦LBL "CHARS"
02 1.127
03 STO 00
04 FIX 0
05♦LBL 00
06 RCL 00
07 ACX
08 ACCHR
09 ADV
10 ISG 00
11 GTO 00
12 FIX 4
13 END

6295 081 "A"
6296 012 "R"
6297 010 "P"
6298 259 ?NC XQ
6299 1B4 ->6D96
629A 2D1 ?NC XQ
629B 188 ->62B4
629C 3B8 READ 14(d)
629D 358 ST=C XP
629E 31D ?NC XQ
629F 1A4 ->69C7
62A0 09D ?NC GO
62A1 1B6 ->6D27
62A2 215 ?NC XQ
62A3 1B4 ->6D85
62A4 3E0 RTN
62A5 2C9 ?NC XQ
62A6 1B4 ->6DB2
62A7 3E0 RTN
62A8 393 JNC 629A -0E
62A9 279 ?NC XQ
62AA 1B4 ->6D9E
62AB 37B JNC 629A -11
```

## Port configuration:

Make sure you have your HP41<>USB connected to your computer before USB-82143A.exe is launched.

If no COM ports are available you'll see an error message just below the text window indicating that circumstance. If this happens, please close the application, connect your USB interface and run USB-82143A.exe again.

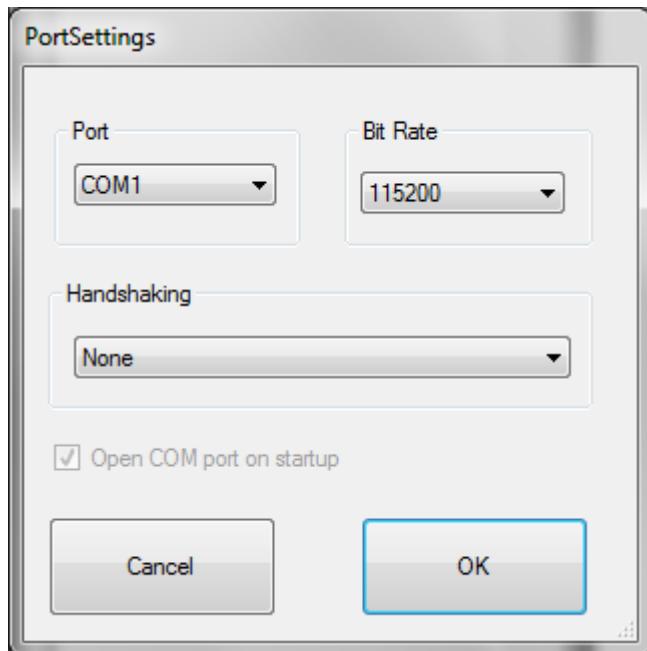
By default the lower numbered available COM port is assigned to the USB-82143A application. You can check which port number is actually selected by looking at the status bar at the bottom of the main form. In the example above COM1 has been selected.

In case the designated COM port number does not fit the one used by your Prolific USB adapter, click the “Port settings” button to change it. See Fig. 2 below.

If HP41<>USB is the only serial interface attached or the first numbered one then no further configuration is required, and you can start using your system

See Hardware description above for details on how to find out the COM port number assigned to your interface.

Changing default COM port number dialog (Fig. 2):



Once you have set the appropriate default COM port click OK. Please do not modify “Bit Rate” nor “Handshaking”.

Port number default configuration is not interactive, therefore, you'll need to close the application and run it again to get your port number effectively changed. It will be kept as long as you don't change it again. It is recommended that you use the same USB port everytime you use your HP41<>USB interface module.

## Controls:

A few controls have been added to the top of the Main Form in order to help users in handling the printing output from their USB-82143A simulation.

- **Clear**: This button erases all contents present in the Text window.
- **Save**: This button opens a standard Save File Dialog box, allowing the user to enter an appropriate name to the file to be saved. Files are saved using a .rtf (Rich Text File) format.
- **Print**: This button opens a standard Print Dialog box. User may set usual default printing parameters: Printer, Paper size, etc. Once finished, just click Print and you'll get a hardcopy of the contents in the Text window.

Both Text window and printed output use “Lucida Console” font. A translation look-up table has been set in order to produce an output as similar as possible to the original HP-82143A character set.

A detailed character set output (0 to 127) is shown below (Table 1).

	0	1	2	3	4	5	6	7
0	◆	Θ	space	0	@	P	`	p
1	(¤)	Ω	!	1	A	Q	a	q
2	(Ж)	δ	"	2	B	R	b	r
3	←	Å	#	3	C	S	c	s
4	α	å	\$	4	D	T	d	t
5	β	Ä	%	5	E	U	e	u
6	Γ	ä	&	6	F	V	f	v
7	↓	Ö	'	7	G	W	g	w
8	Δ	ö	(	8	H	X	h	x
9	σ	Ü	)	9	I	Y	y	y
A	◆	ü	*	:	J	Z	j	z
B	λ	Æ	+	;	K	[	k	π
C	μ	æ	,	<	L	\	l	
D	(Д)	≠	-	=	M	]	m	→
E	Τ	£	.	>	N	↑	n	Σ
F	Φ	⊗	/	?	O	-	o	†

Characters in parenthesis have been chosen for their similitudes with the original HP-82143A characters.

## **Printing commands output:**

This chapter shows the comparative behavior of different printing commands listed as they're presented in the HP-82143A catalog. Please keep in mind that only printing functions working on MAN mode are available.

Other printing modes (NORN & TRACE) or graphical features are not functional in the USB-82143A simulation.

The printing Flags setting (12 & 13) have no effect in the output of the USB-82143A

### **ACA: Accumulate Alpha.**

Behavior: Same as HP-82143A, accumulates the contents of ALPA register into printer buffer for later printing.

Note: in case of a series of ACA commands to be sent to the USB simulator and no printing command (ADV or PRBUF) is issued in less than 5 seconds, a Time Out warning will show up below the Text Box window. Other than this it will behave and continue working as expected.

### **ACCHR: Accumulate Character.**

Behavior: Same as HP-82143A.

### **ACCOL: Accumulate Column.**

Graphical command not available in USB-82143A simulation.

### **ACSPEC: Accumulate Special Character.**

Graphical command not available in USB-82143A simulation.

### **ACX: Accumulate X-Register.**

Behavior: Same as HP-82143A.

### **BLDSPEC: Build Special Character.**

Graphical command not available in USB-82143A simulation.

### **LIST: List Program Lines.**

Behavior: Same as HP-82143A.

### **PRA: Prints ALPHA contents.**

Behavior: Same as HP-82143A.

### **'PRAXIS: Print Plotting Axis.**

Graphical command not available in USB-82143A simulation.

### **PRBUF: Print buffer contents.**

Behavior: Same as HP-82143A for text printing.

Note: This command is also used to print graphical accumulated info, which is not available in USB-82143A simulation.

### **PRFLAGS: Print Status and Flags.**

Behavior: Same as HP-82143A.

**PRKEYS: Print Assigned Keys.**

Behavior: Same as HP-82143A.

**PRP: Print Complete Program Listing.**

Behavior: Same as HP-82143A.

**'PRPLOT: Print Plotting Curve of a given function.**

Graphical command not available in USB-82143A simulation.

**'PRPLOTP: Print Plotting Curve of a given function. (Non Interactive)**

Graphical command not available in USB-82143A simulation.

**PRREG: Print Registers.**

Behavior: Same as HP-82143A.

**PRREGX: Print Registers by X.**

Behavior: Same as HP-82143A.

**PRΣ: Print Statistics Registers.**

Behavior: Same as HP-82143A.

**PRSTK: Print Stack.**

Behavior: Same as HP-82143A.

**PRX: Print X-Register.**

Behavior: Same as HP-82143A.

**REGPLOT: Plot Single line using data in registers.**

Graphical command not available in USB-82143A simulation.

**SKPCHR: Skip characters.**

Behavior: Same as HP-82143A.

**SKPCOL: Skip Columns X-Register.**

Graphical command not available in USB-82143A simulation.

**STKPLOT: Plot Single line using data in stack.**

Graphical command not available in USB-82143A simulation.

## Appendix A:

The information below is intended to provide programmers with the required information to write their own application on their favourite platform.

The requirements for this utility are very simple. The Clonix function of the interface simulates a printer connected to the calculator; it handles the status and flags requests from the NUT CPU. So the HP-41 'sees' a physical HP-82143A connected to one of its ports.

The serial part of the interface sends every byte received from the NUT CPU to the USB serial port.

Allowed range of received bytes are as follows:

- Values from 0 to 127 (h'00 – h'7F) are the printable characters. See Table 1 above.
- Values from 161 to 183 (h'A1 to h'B7) inserts 1 to 23 spaces in the printing buffer.
- Value 224 (h'E0) causes End of Line left justified. Prints and sets new line.
- Value 232 (h'E8) causes End of Line right justified. Adds as many spaces as needed to fill a 24 character line, then prints and sets new line.

Any other value is disregarded. Print width is fixed to 24 characters.

Four “If...Then” constructions are used in the VB implementation to filter the received bytes and print them as required. Below you'll find the Sub construction in VB to produce the required output.

```
Dim newReceivedData As Byte
Dim PrtTxt As String
Dim HP82143ACharMap As String = "♦¤Ж←αβΓ↓Δσ♦λμπτΦΘΩδÅåÄäÖöÜüÆæ≠£¤!" + Chr(34) + "#$%
%&'()*+,-./0123456789;:<=>?@ABCDEFGHIJKLMNPQRSTUVWXYZ[`]_`abcdefghijklmnopqrstuvwxyz|→Σ|"
```

```
Public Sub DataReceived(ByVal sender As Object, ByVal e As SerialDataReceivedEventArgs)
    Try
        newReceivedData = 0
        'Initializes received data variable
        While newReceivedData <> 240
            newReceivedData = SelectedPort.ReadByte
            'Receiving loop active for every line
            '240 is set as “print-done” indicator
            'Gets byte from serial port
            If newReceivedData < 128 Then
                PrtTxt = PrtTxt + HP82143ACharMap(newReceivedData)
                ' If it's a printable character
                'Translate according to look-up table
            End If
            If newReceivedData > 160 And newReceivedData < 184 Then
                PrtTxt = PrtTxt + Space(newReceivedData - 160)
                'If it's a “skip-chr” byte
                'append spaces as required
            End If
            If newReceivedData = 224 And PrtTxt <> "" Then
                PrtTxt = PrtTxt + Chr(10)
                'If it's an EoL (left justified)
                'add a LF (h'10) character
            End If
            If newReceivedData = 232 And PrtTxt <> "" Then
                PrtTxt = Right(Space(24) + PrtTxt, 24) + Chr(10)
                'If it's an EoL (right justified)
                'add spaces as required and LF
            End If
        End While
    End Try
End Sub
```

```

If Right(PrtTxt, 1) = Chr(10) Then
    RaiseEvent UserInterfaceData("AppendToMonitorTextBox", PrtTxt, Color.Black) 'Proceed with printing
    PrtTxt = ""
    newReceivedData = 240
End If
End While

Catch ex As Exception
    DisplayException(ModuleName, ex)

End Try

End Sub

```

Hope this info may be useful to anyone willing to write applications for different platforms. Please do not hesitate in mailing me if you want further details, or find any errors in this document.