

How to use PAST for data acquisition

Pierre DIEUMEGARD
Lycée Pothier
2 bis rue Marcel Proust
45000 Orléans (France)
pierre.dieumegard@ac-orleans-tours.fr

PAST¹ is a software for data analysis : first we have to get data, and second we can use PAST for statistics and graphics.

But there is an under-used menu option, in the right part of the menu : « script ». In fact, this « scripting language » is a programming language, a Pascal language.

In comparison to big programming languages as Delphi or Python, this scripting language is very poor, but new version of PAST (3.11) has a very powerful ability : it can use dynamic libraries (.dll of Windows).

Dynamic libraries are pieces of executable programs, with special functions. Dynamic libraries cannot be used alone, they must be called by a main program : here the main program is PAST. There are dynamic libraries for a lot of purpose : here we discuss only for measurement, in order to give data to PAST. Because there are a lot of measurement devices, this paper will describe only for a few devices easily found on every PC : internal clock and soundcard, but the receipt is the same one for serious measurement devices : pHmeters, thermometers, scales, spectrophotometers, or other devices. Only we need the good dynamic library, and we need to know the good functions of this dll..

Dynamic libraries can be programmed by usual programming languages : FreeBasic, PureBasic, Delphi, FreePascal, C, C++... You can have dll for many devices on <http://sciencexp.free.fr> ; this website is mainly in french, but there is an explanation about programming dynamic libraries in english at http://sciencexp.free.fr/documents/programmation/bibdyncourt_en.pdf

Because scripting language of PAST is Pascal, the examples of dynamic libraries here are also in Pascal (Delphi or FreePascal) : if you can understand this scripting language, you can also understand the source code of the dynamic library.

1 An example with internal clock (function stdeadouble)

Dynamic library is « bibdysyst_delphi2016.dll » (source : bibdysyst_delphi2016.dpr).

There is a function « stdeadouble », for the analog input, and stdeadouble(0) gives the number of seconds in a minute, from the internal clock. This function has a parameter (« double » : a float number with double precision) which is the channel number, and gives a result, also a float number with double precision.

The first line loads a function named stdeadouble from library « bibdysyst_delphi2016.dll ».

```
function stdeadouble(n:double):double;stdcall;external  
'bibdysyst_delphi2016.dll';
```

The loop between lines 11 to 25 sends data to the text tab, the graphic tab and the table tab. X-value is the time, in seconds, from function time of Past. The Y-value is also the time in seconds in the

¹ Hammer, Ø., Harper, D.A.T., and P. D. Ryan, 2001. PAST: Paleontological Statistics Software Package for Education and Data Analysis. Palaeontologia Electronica 4(1): 9pp.

minute, from dynamic library (because channel 0 gives this value). Line 24 gives a pause of 1000 milliseconds.

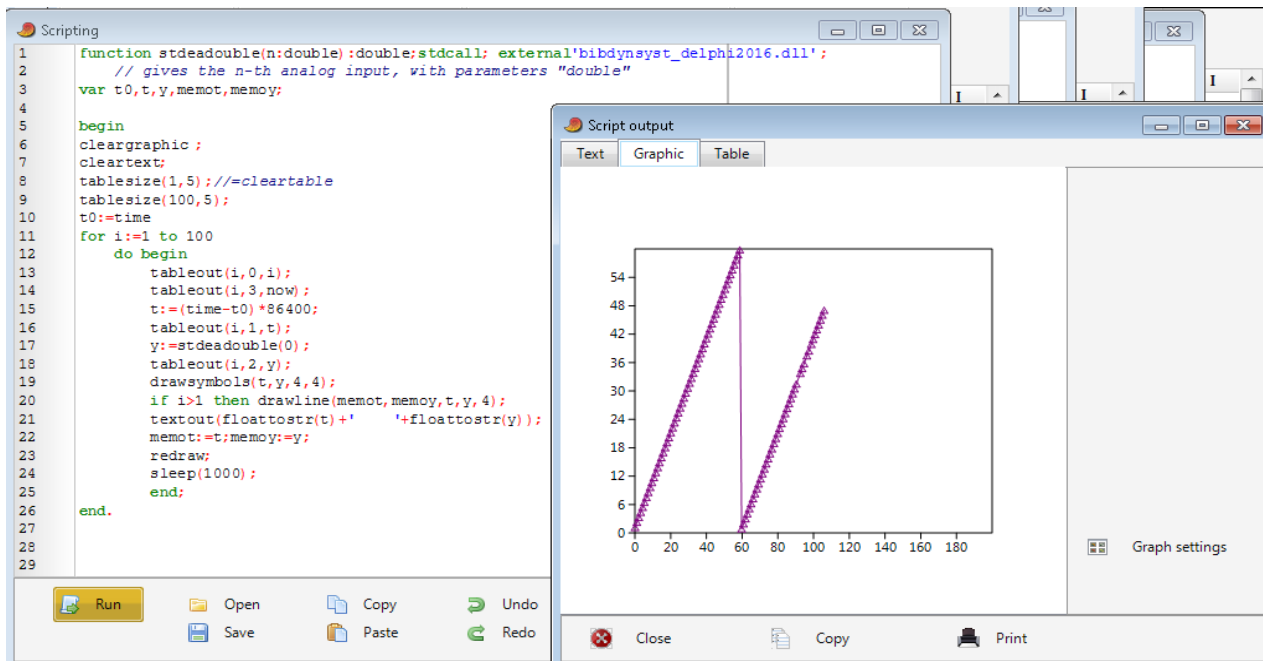


Illustration 1: Measuring by bibdynsyst_Delphi2016.dll : seconds in the minute

Exercise : from the table (third tab), copy the data to the clipboard, and now paste them to the main spreadsheet of PAST. Plot the graph of columns B and C.

Exercise : copy the data from the table to the clipboard, and then paste them to your spreadsheet (LibreOffice, Excel, PlanMaker, Gnumeric...). Plot the graph of these values.

2 How to choose the channels : function stdnead

In Mensurasoft system, all numerical functions have a name (a string). If the name is empty (length of the string equals zero), the numerical function does not exist.

The second little program :

- shows the second of the minute (stddeadouble(0), as the first program).
- shows twice the detail of this driver (function stddetail). The first one is directly stddetail, but with ideograms, because stddetail gives an ASCII-ANSI string, and PAST uses Unicode-strings. The second is decoded by « decode_unicode », and gives a string with latin characters.
- shows six channels for analog input. Each line begins with the number (0 to 5), gives the name of the channel, and the numerical value of this analog input. Channel 0 is second in the minute, channel 1 is date and time (as « now » in PAST language), channel 2 is the second in the day, channel 3 is X-position of the mouse, channel 4 is Y-position of the mouse. Because channel 5 does not exist, the name is a string whose length is zero, and the value is -777.

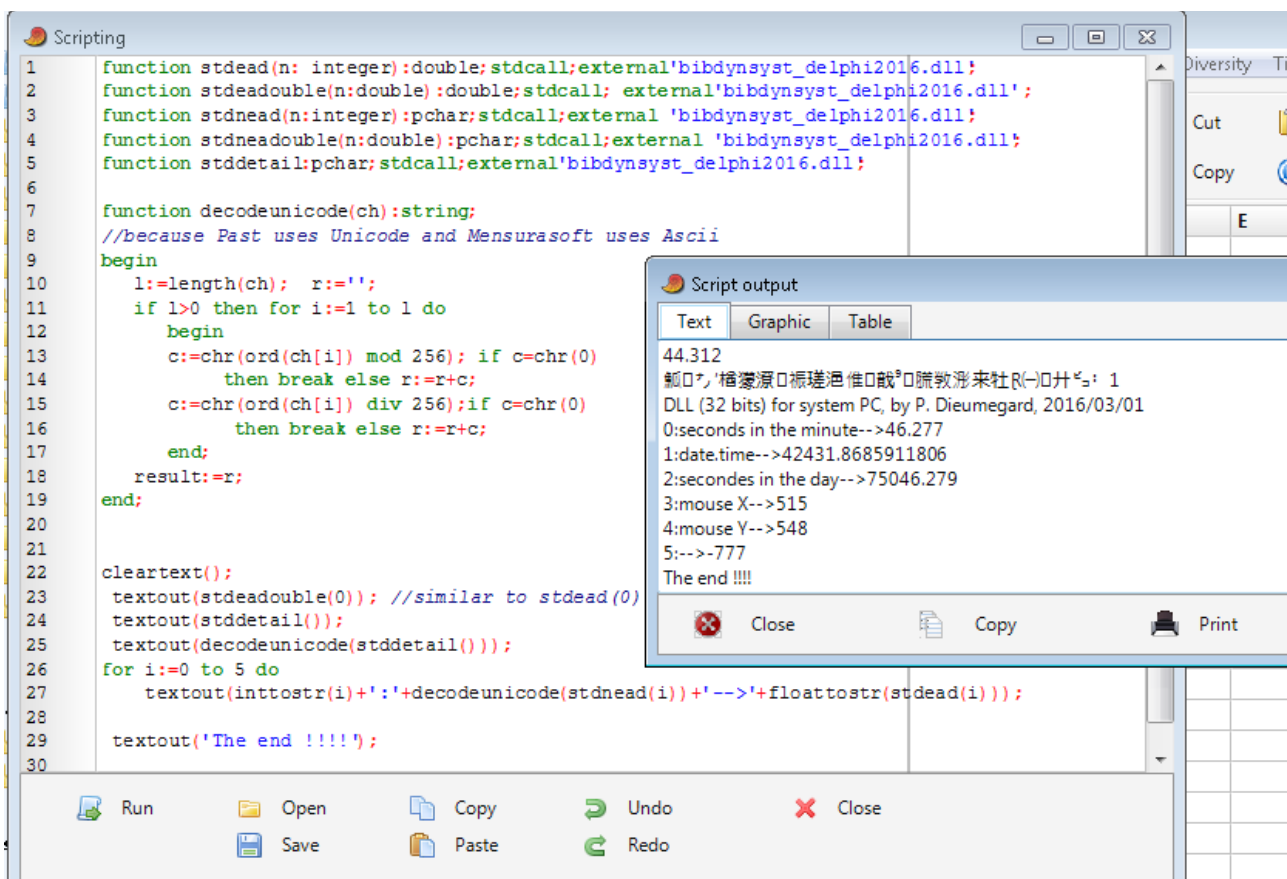


Illustration 2: A little program to explore analog inputs : `stdnead(n)` gives the name of input function `stdead(n)`. There are 5 real functions (0 to 4), and the next one does not exist. The first lines loads functions from library. « pchar » is a string (null-terminated).

Exercise : modify the first program, to draw the Y-position of the mouse against the time.

3 Exploring the other functions : analog outputs, binary inputs and outputs

`stdsad(n, value)` can set the value to the n-th channel for analog output (n is an integer, and value is a double ; this function gives a double)

`stdeb(n)` gives the status of the n-th channel for digital (binary) input (n is an integer, and this function gives an integer)

`stdsb(n, value)` can set the value to the n-th channel for digital (binary) output. Parameters n and value are integer. This function gives an integer, but this is not important.

This library `bibdynsyst_delphi2016.dll` does not have analog output. Binary inputs are the buttons of the mouse, and binary outputs are lock keys of the keyboard, and the CD : with this program, the CD-tray will be open.

We can get these functions by :

```

function stdsad(n:integer ; v:double):double;stdcall;external
'bibdynsyst_delphi2016.dll';

function stdeb(n:integer):integer;stdcall;external
'bibdynsyst_delphi2016.dll';

function stdsb(n:integer;v:integer):integer;stdcall;external

```

```
'bibdynsyst_delphi2016.dll';
```

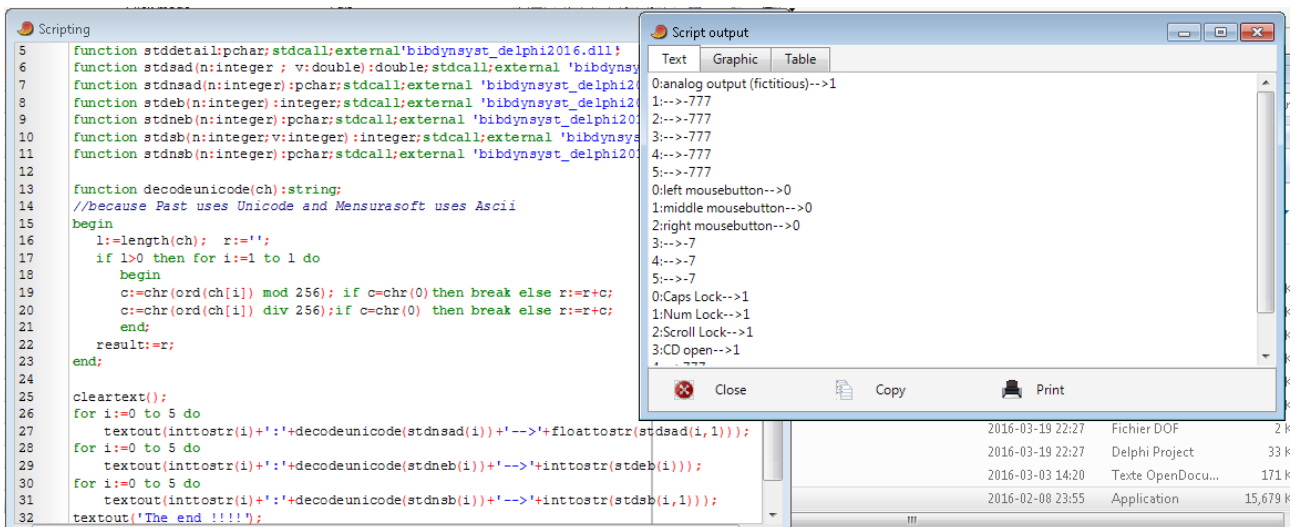


Illustration 3 : use of analog output, binary input and binary output.

Exercise : you are unhappy because this program open the CD-tray, and locks your keyboard. Modify it !

4 How to study sound, by a microphone on soundcard

Until now, this library is only for a test : it cannot do a true measurement.

On the internet, you can have « port.dll » (<http://www.b-kainka.de/port.zip>). It can be used with serial port (for a lot of pHmeters, scales, spectrophotometers, and others), and also with the microphone of a soundcard : all laptop computers can use it.

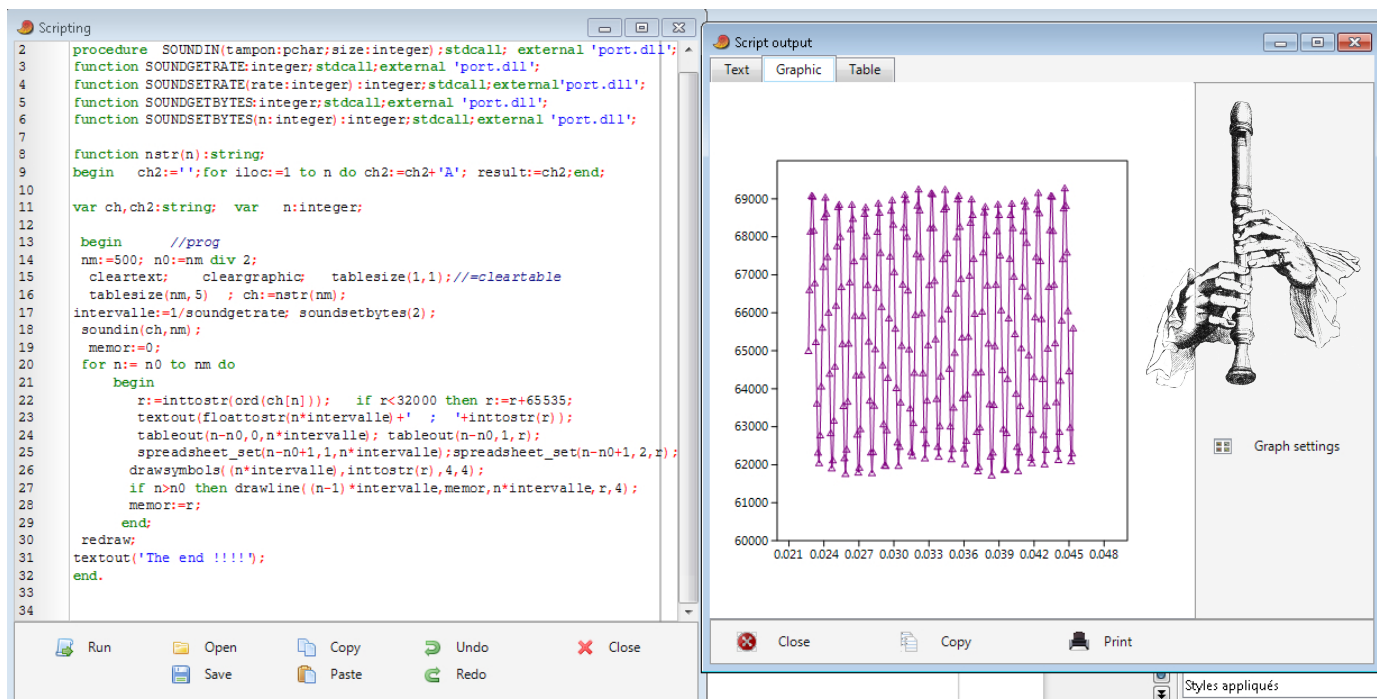


Illustration 4: Recording the recorder (soprano) : A (= la)

This little program send data to the main spreadsheet of Past. With it, we can use « Time series / Spectral analysis / Simple periodogram », and get the frequency of this sound.

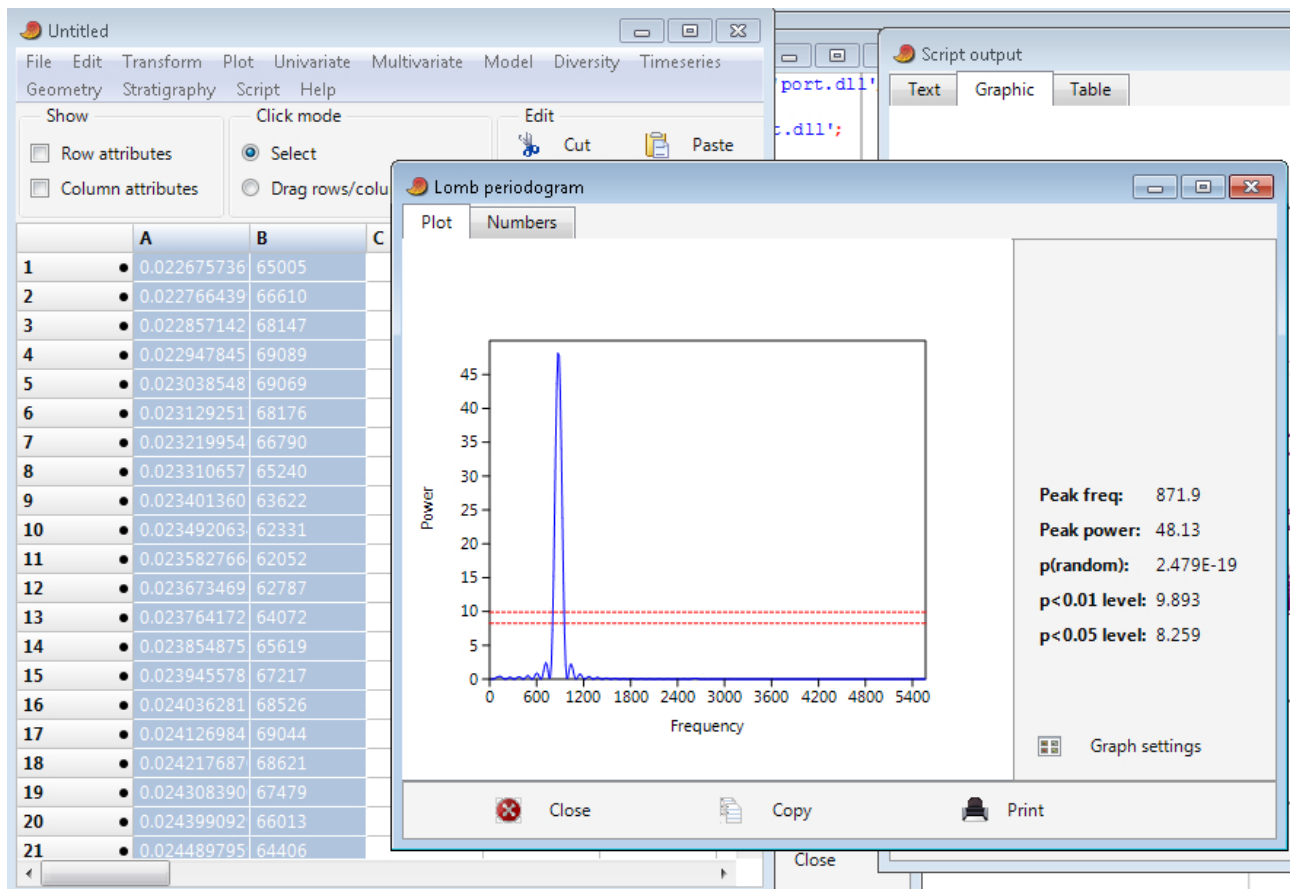


Illustration 5: Periodogram of sound "A" (la) of a soprano recorder

We can see that the frequency of this sound is 871.9 Hz (theoretically 880 Hz).

Exercise : speak to the microphone during a run of this program, and test the option "periodogram" to see the main frequencies of your voice. Compare voice of a man to a voice of a woman.

5 Conclusion

These little programs show that Past can use functions of dynamic libraries (DLL).

The main job of Past is to plot and analyse data. Its scripting language, using dynamic libraries, can make a bridge to physical world.

Library port.dll is useful for a lot of devices.

At <http://sciencexp.free.fr>, you can get libraries for many devices.

You can also use your DLL, home-made by a compiled language (FreeBasic, PureBasic, C/C++, Pascal/Delphi).