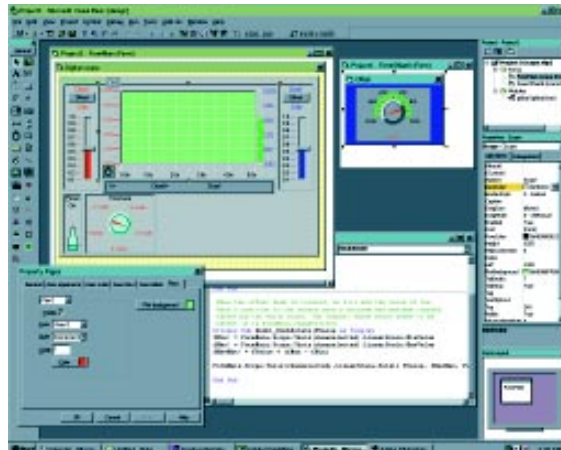# VIX Components - Virtual Instrumentation Software Tools

Virtual Instrumentation is the software representation of traditional and exciting new measurement instruments on a personal computer. With virtual instrumentation software tools and a data acquisition board you can construct a custom instrument. Your virtual instrument will have the controls and displays you want to see and use. It will accept the ranges you choose and will analyze measurement data in the manner you prescribe. No wonder Virtual Instrumentation has swept the measurement world by storm!

To create a virtual instrument you need easy to use software tools. Virtual Instrumentation tools should have the visual and control aspects you need and be simple to integrate with hardware and other software such as browsers, spread sheets, Visual Basic programs and C/C++ programs. One set of tools should work in all environments uniformly. ComputerBoards' VIX Components 32 bit and 16 bit components does all this and more, bundled together into one product that combines the best of the programming and ActiveX control environments.

## VIX Components Includes Both 32 bit for Win/NT & Win95 and 16 bit for Win3.x

## VIX-32 bit support *NEW!*

32 Bit Virtual Instrumentation ActiveX Graphical Controls and Dynamic Link Libraries for Display, Analysis and Control.

### Specifications
32 Bit Architecture
Visual C/C++, Visual Basic 5.x, Visual Basic for Applications, ANY ActiveX Container such as WEB pages, Labview, HP-VEE and other modern software programs.

### VIX Components Features
VIX-Components is the newest virtual instrumentation software tool kit. It captures the best visual aspects of earlier OCX and VBX virtual instrumentation software tools in a new paradigm; AcitveX. The ActiveX architecture is a universally accepted standard for software component objects. This means that you learn how to use an object in one environment, and that knowledge is applicable to all. If you learn to use a Bar Meter in Visual Basic, it is the same object and exhibits the same behavior in C++, Visual Basic For Applications and on a WEB page using JAVA or VB Script.

### On Line Documentation, Help and Examples
An extensive set of context sensitive help documents provide complete information on methods, events and functions. Graphics showing the locations of user programmable features within the Active X controls are located in the help file along with easy to understand explanations. Within the help file are example programs that you can cut and paste to your application.

An extensive set of example programs for Visual Basic 5, C++ and HTML are included. Many simple applications may be constructed from these examples.

## VI-16 bit support

Windows 3.x and its programming languages are widely used still, and it is important for you to know that ComputerBoards continues to support all your legacy platforms. 16 Bit Virtual Instrumentation Software Tool include VBX and Dynamic Link Libraries for Graphics, Analysis & Control

### Specifications
16 Bit Architecture for Windows 3.x and 16 Bit programming languages. Includes VBX controls and a DLL with functions which may be programmed to create graphics, analyze data and implement real-time control.

# A Wide Array of Display, Analysis and Control Tools

Virtual Instrumentation software tools may be grouped into three categories; display, analysis and control. Display tools are used to create beautiful and functional user interfaces and to visually display measurements and analyzed data. Analysis ActiveX controls and API functions transform and analyze data after acquisition, or in real time.

The table below shows the functions available as ActiveX controls, VBX Controls and API function calls. A large 'X' means ActiveX 32 bit control. A small 'x' means VBX 16 bit Control. A large 'A' means 32 bit API function while a small 'a' means that function is available in the 16 bit package only. Win95 and Win/NT can take advantage of 32 bit ActiveX controls and API functions *ONLY*. Win3.x can use the 16 bit VBX controls and functions only.

| GUI objects: | 32 Bit | 16 Bit | Data analysis (continued) | 32 Bit | 16 Bit |
|---|---|---|---|---|---|
| Analog meter | X | x,a | DSP | | |
| Annunciator | X | x,a | FFT | A | a |
| Bar meter | X | x,a | Inverse FFT | A | a |
| Circular knob | X | x,a | Magnitude of Harmonic | A | a |
| Digital meter | X | x,a | Power Spectrum | A | a |
| LED | X | x,a | Convolution | A | a |
| Oscilloscope | X | x,a | Correlation | A | a |
| Scrolling strip chart | X | x,a | FIR Filters (Real time) | | |
| Slide switch | X | x,a | Low-pass | X | a |
| Slider control | X | x,a | High-pass | X | a |
| Spin bar | | x,a | Band-pass | X | a |
| Toggle switch | X | x,a | Band-stop | X | a |
| XY Chart | X | x,a | Coeffiecients & | X | a |
| XY Plot | | x,a | Response | X | a |
| | | | Integration | | |
| **Data analysis and manipulation:** | | | Any array using extended | A | a |
| Calculation | | | trapezoidal and Simpson's | A | a |
| Z=(A,Y) Op (BX +C) | | a | Matrix | | |
| Calibration | | | Addition | A | a |
| Y = AY + B | | a | Subtraction | A | a |
| Complex Numbers | | | Multiplication | A | a |
| (Two Arrays) | A | a | Transpose | A | a |
| Additon | A | a | Determinant | A | a |
| Subtraction | A | a | Inverse | A | a |
| Division | A | a | Signal Generation | | |
| Multiplication | A | a | Sine | | a |
| Rect. to Polar | A | a | Pulse | | a |
| Curve Fit | | | Triangular | | a |
| Linear | A | a | Sawtooth | | a |
| Polynomial | A | a | Smoothing filters | | |
| Derivative | | | Moving average & | | a |
| Digital Windows | | | Median | | a |
| Blackman | A | a | Statistics | | |
| Hamming | A | a | Mean | A | a |
| Welch | A | a | Standard Deviation | A | a |
| Hanning | A | a | Variance | A | a |
| Bartlett | A | a | Absolute Deviation | A | a |
| Parzen | A | a | **Real time control:** | | |
| Rectangular | A | a | Alarms with four setpoints | | a |
| | | | Proportional Integral/Derivative | | a |

X = 32 bit ActiveX, A = 32 bit API
x = 16 bit VBX, a = 16 bit API

## GUI Objects

Input controls like circular knobs and sliders allow you to interact and change the value of a variable. Output controls like analog meters, bar meters, strip charts etc. are used to display the value of a variable in a graphical format that is easier to interpret and use. Any control can be printed, saved as a bitmap or copied to the clipboard. Here is a brief description of the GUI controls.

**Analog meter:** These display a value by positioning a needle along a circular scale. You can specify the size, diameter, number of needles and their type, arc and color of the meter. In addition, the meter's scale can be divided into three separate colored regions to represent low alarm, normal and high alarm ranges.
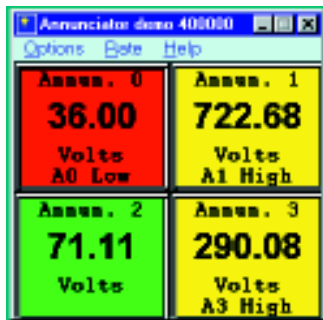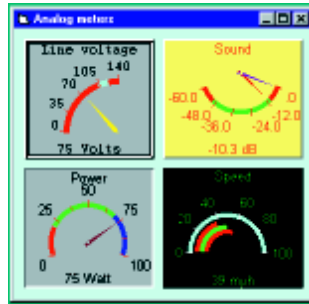
**Annunciator:** An annunciator is used to display the current value and alarm status of data in text form. Typically, an application would have a matrix of these annunciators in which each would be monitoring a variable. Specify the label of each annunciator and the unit of the monitored variable as well as the color for the annunciator and the text message to be displayed when a low alarm, normal and high alarm condition is detected.

**Bar meter**: A bar meter displays a value by changing the size of a bar. The bar can be chosen to be either vertical or horizontal and have a scale with tick marks  Specify the range of the scale and the number of ticks. The bar can be set  to change its color when the   variable falls below a low alarm point, or is in the normal operating region or exceeds a high alarm point.
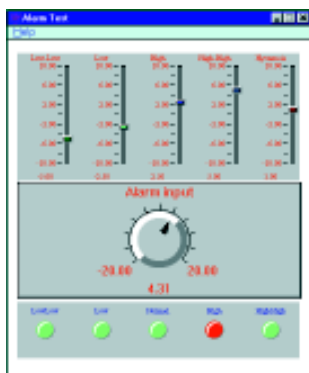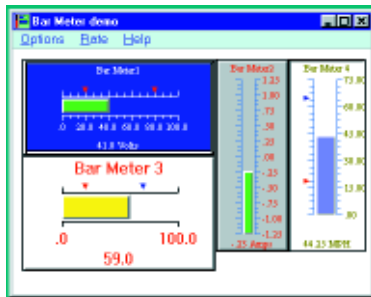
**Circular knob:** A circular knob is an input control which changes the value of some variable. You can specify the color of the knob, the circular arc over which it should move and the range for the input. The circular knob can be rotated by clicking on it and then moving the mouse. This control notifies the application  program by sending messages about the increase or decrease in value.

**Digital meter:** A digital meter is used to display the exact value of a variable in numeric format. The library functions allow you to specify the size, color and label of the meter and the unit of the variable. Additionally, the color and font for the display can be selected.
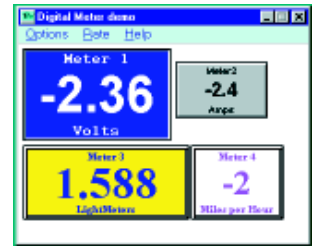
**LED:** An LED can change color if the value of a variable is above or below a specified threshold. When the value is below the threshold, the LED is off and is displayed your choice of 'off' color. When above, it changes to your 'on' color. LEDs may be square or round and respond to mouse clicks to perform as push button switches.
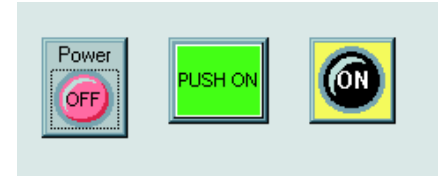
**Oscilloscope:** The oscilloscope displays the values of one or more variables (or channels) as waveforms which change over time. The vertical axis is used to depict the amplitude of the waveform and the horizontal axis for the period. Specify the size of the display, its background color, colors for all the monitored channels, color and line types for the axes and grids and many other visual attributes. The waveforms are displayed starting at the left edge of the display and progressing to the right. When the right edge is reached, the display is cleared and the waveform starts all over again from the left edge. The oscilloscope also has options for triggering and displays the values of the waveforms when the mouse is clicked in the plot.
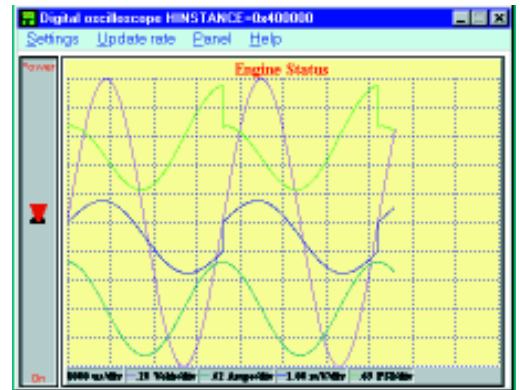
**Scrolling strip chart:** A strip chart provides all the features of the oscilloscope but the waveforms are scrolled from the right to the left as they change over time. Therefore at any instant, the oldest values are to the left edge of the display and the most recent values are at the right edge of the display. You can choose several visual attributes of the scrolling strip chart. This control has an important property in that it stores data in an internal buffer. You can specify the size of this buffer and historical data can be replayed as desired.
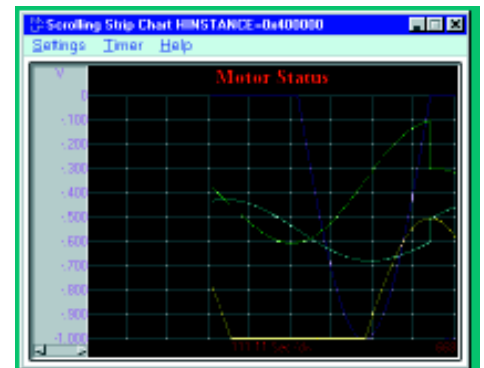
**Slider control & Slide switch:** A slider control is a scroll bar used to change the value of a variable. Specify the color of the control, the scale over which it should move and the range for the input. The thumb tack in the slider can be moved by clicking on it and then moving the mouse. This control notifies the application program by sending messages about the increase or decrease in value. A slider <u>switch</u> is limited to discrete positions, like 1, 2, 3...
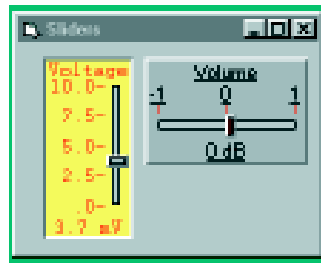
**Spin bar:** Spin bars have an edit line with an up and down arrow to the right. Entering a value in the edit line sends the value to your program (values may be validated). Pressing the arrows will increment or decrement the value in the edit area. There is not a spin bar control in the 32 bit ActiveX control group because VB5 added the spin bar as a standard control.
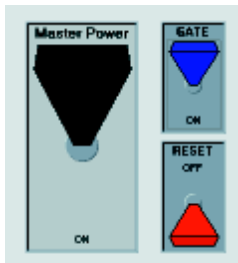
**Toggle switch:** A toggle switch is an input control which is used to change the value of some variable from ON to OFF or vice versa. Like the other controls, various visual attributes of the switch can be selected by the program. This control notifies the application program by sending messages when the operator clicks on the switch and toggles it.

**XY Chart:** An XY Chart displays a set of Y values plotted against a common X axis. You can specify the Y scale of each data set and the extent of the X axis. Choose the font, grid type, line type, background color and much more. Add legends and markers.
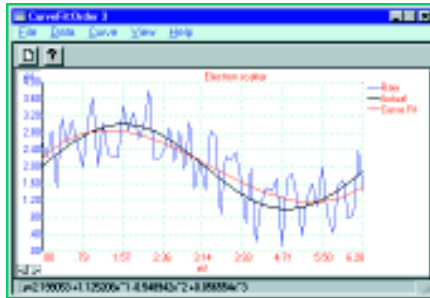
**XY Plot:** This a dynamic display of sets of X and Y values. Specify the X and Y scales of each data set and several visual properties.
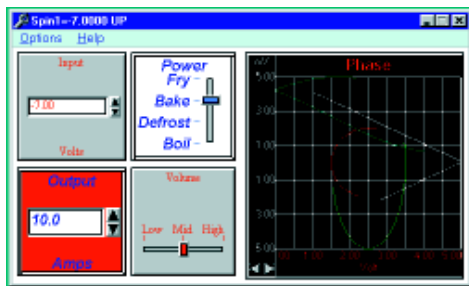
## Programming Considerations

All of the GUI controls are implemented as custom controls which extend the set of standard controls available in Windows and ease application development tremendously. Like standard controls, these custom controls are simply child windows dedicated to a single purpose. A custom control communicates with the operator, the application program, and the system. The control's visual describes
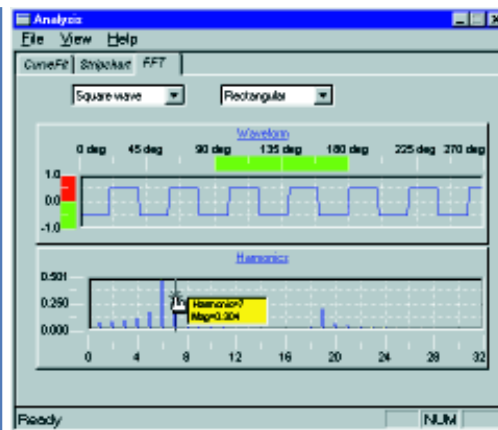
the complete operator interface, while the control's messages and exported functions make up its API. Applications can communicate with the control by calling its API functions or sending it messages. In addition, applications can also use any of the Windows API functions to manipulate the custom controls. This allows you to customize any aspect of the control using standard Windows techniques including subclassing.

## Data Analysis and Manipulation Functions

VIX-COMponents for Programmers enables the application developer to perform sophisticated analysis on data being collected. The analysis is performed in real-time with minimum overhead. A brief description of each of the data analysis functions is given here.

**Calculation:** Calculates Z = (AY) operation (BX+C), where Z is the output data set, X and Y are input data sets, A, B and C are constants. The operation can be addition, subtraction, multiplication or division.

**Calibration:** Computes Y=AY+B, where Y is an input data set, A and B are constants.

**Complex Number:** Add, subtract, multiply and divide complex numbers.
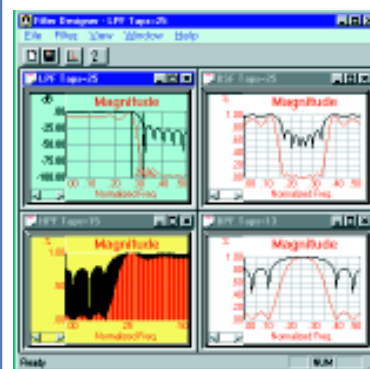
**Curve Fit:** Perform linear and polynomial curve fit on a set of data

**Derivative:** Finds the derivative of an input data set in real time. You can select the number of data points used in the computation.

**Digital Window:** Reduce spectral leakage in FFT calculations and design FIR filters using Blackman, Hamming, Welch, Hanning, Bartlett, Parzen and Rectangular windows.
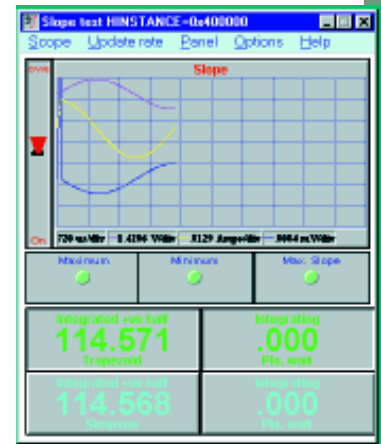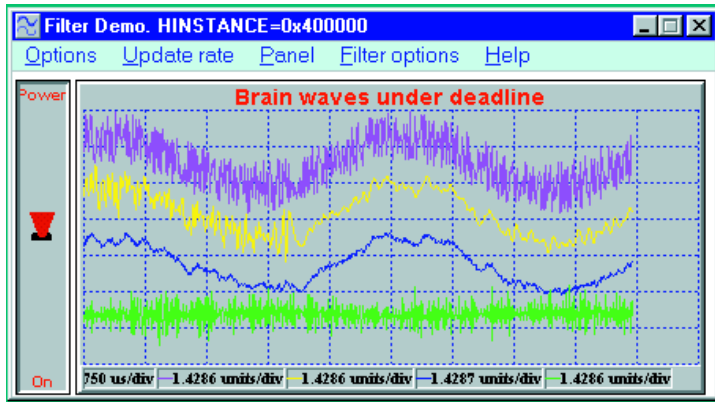
**DSP:** Calculates the FFT of real data, magnitudes of any specified harmonic and the inverse FFT.

**Filters:** Design and implement FIR and IIR digital filters.

**Integration:** Computes the integral of a given data set using the extended trapezoidal rule and extended Simpson's rule.

**Interpolation:** Perform polynomial interpolation and extrapolation.

**Matrix Operations:** Add, subtract and multiply matrices. Compute the transpose, determinant and inverse of a matrix.

**Signal Generation:** Generate sine, pulse, triangular and sawtooth waveforms of any period and duty cycle.



**Smoothing filters:** Eliminates noise from a given data set using moving average, median and Savitzky-Golay filters. You can select various options for these filters.

**Statistics:** Determines the mean, standard deviation, variance and mean absolute deviation of a given data set.

## Real-Time Control Functions

The 16 bit API has functions to test for alarms and PID loops. Each operates in real-time and is based on the timer you select.

**Alarms:** Monitor up to four alarm thresholds and hysteresis.

**PID:** Proportional control based on linear or nonlinear functions.
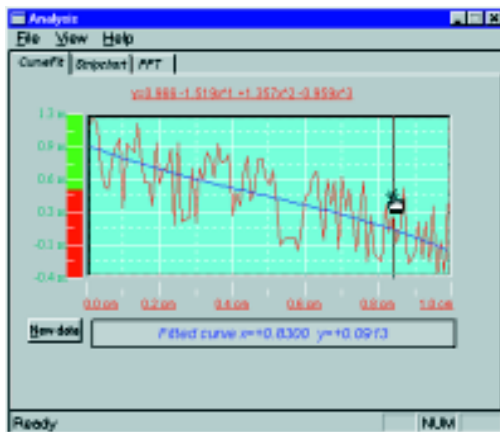
## MORE ACTIVE X ON THE WAY!

We are aggressively adding to the ActiveX group. All of the math functions in the API, as well as many new functions are being added to VIX-Components. Check our www.computerboards.com site to find out about the current list and what they can do for you.

## Royalty Free Distribution

Programs written with VIX-Components may be distributed royalty free. VIX-Components is based on several binary files which must be present on each CPU running a program written with the components. When you purchase VI-Components, your license is for use on one CPU at a time. For a one time charge you may purchase the license to make unlimited copies of the binary files for distribution with your application program.

Simply order part number VI-COMPONENTS-RUN. You will receive a distributable copy of the necessary files and a license agreement. From then on you may distribute your application royalty free.

## Source Code License

VIX-Components is written in C and C++ using MFC and you may purchase the source code. For expert programmers this is an opportunity to save hours developing what you see here, as well as lots of supporting code. Source code may be purchased for one time use, or for redistribution in a product.

Call and ask for the technical sales manager for specifics. Sale of source code is restricted to noncompetitive applications and entities

---

**ORDERING GUIDE**

**VIX-COMPONENTS**
ActiveX, VBX and Dynamic Link Libraries
Ships on CD-ROM with card to request 3.5" floppy

**VIX-COMPONENTS-RUN**
Run time license for unlimited distribution of applications.

**VIX-COMPONENTS-SRC**
Source code in C. Call for ordering & license form.