

Spectral Dynamic's
Programmable Function Calculator
Users Manual (V1.79)
Dec 3, 1998

| | |
|---|-----------|
| OVERVIEW | 5 |
| USING THE PROGRAMMABLE FUNCTION CALCULATOR | 7 |
| GETTING STARTED | 8 |
| The PFC Main Menu | 8 |
| The Function List | 8 |
| Equation Section..... | 9 |
| Programming Section..... | 10 |
| Programmable Function Keys | 10 |
| Input/Output Function Assignment | 11 |
| Key Pad..... | 11 |
| USING THE PFC AS A CALCULATOR | 12 |
| PROGRAMMING THE CALCULATOR | 12 |
| Parameter Substitution | 13 |
| Function List..... | 13 |
| Internal Functions | 14 |
| CREATING NEW EU DEFINITIONS..... | 15 |
| Labels..... | 16 |
| <i>SOURCES, SINKS AND STREAMS.....</i> | 17 |
| Opening a Source and a Sink | 17 |
| Channel Definition..... | 18 |
| Defining Input Streams | 19 |
| Defining Output Streams..... | 21 |
| Using Input Streams..... | 21 |
| Using Output Streams | 21 |

| | |
|---|-----------|
| Changing Sources | 22 |
| Summary | 22 |
| RUN MODE | 23 |
| Using Run Mode | 23 |
| Run Mode Processing | 24 |
| PFC CONFIGURATIONS | 25 |
| Settings | 25 |
| Preferences | 26 |
| Configuration | 27 |
| Input | 28 |
| Output | 29 |
| APPENDIX A: SIMPLE ANALYSIS LAYOUT FILE FORMAT | 30 |
| APPENDIX B: GLOSSARY | 31 |
| <u>EXPRESSIONS</u> | 31 |
| <i>STANDARD OPERATORS</i> | 31 |
| Term delimiter | 31 |
| Functions | 31 |
| Predefined Functions | 31 |
| User-defined Functions | 32 |
| Engineering Unit Functions | 32 |
| Function Forms Supported | 32 |
| • Simple Form | 32 |
| Fn = <constant> | 32 |
| • Using parameters in representation | 32 |
| • Nesting functions | 32 |
| Function Name | 32 |
| Function Parameter | 32 |
| Parameter Name | 33 |
| Function Argument | 33 |

| | |
|---|-----------|
| Global Parameters..... | 33 |
| • Predefined Global Parameters..... | 33 |
| • Runtime Global Parameters | 33 |
| This is a parameter that is assigned a value after all calculator functions are defined, but before the processor begins execution. Typically a dialog box will appear for inputting values to all undefined runtime global parameters..... | 33 |
| | |
| <u>PROCESSING</u>..... | 33 |
| | |
| Function definition mode (Setup or Program mode) | 33 |
| | |
| <u>I/O configuration</u> | 33 |
| <i>Output</i> | 33 |
| <i>Input</i> | 34 |
| | |
| Configuration Persistence | 34 |
| | |
| Output channel evaluation (Runtime Mode) | 34 |
| | |
| Output channel evaluation (Calculator Mode) | 34 |
| | |
| APPENDIX C: QUICK START EXAMPLE..... | 35 |

Overview

The Spectral Dynamics **Programmable Function Calculator** is a versatile data handler that operates on input data to create new output data. The input and output data sources are completely configurable and can operate on internal data, data files, or live data.

All data operations are performed by **Functions**. The PFC contains input and output functions, built-in functions, and user created functions.

The use of Functions is the key concept behind the versatility of the PFC. The most important characteristics of Functions are:

- All Functions return data
- Functions can be created from other Functions
- Functions can be assigned to other Functions
- Functions can be assigned to input and output streams
- New Functions can be created
- Functions can be constants
- Labels can be associated with Functions

(Input and Output Functions are special cases of Functions in that they return data associated with a defined input and output source. However, they retain all the common features of Functions and can be treated like any other Function.)

The PFC can be used as a standard providing immediate results on any entered value. Input and Output Functions can be used, but are not required. Complex equations can be evaluated in the manner of a simple calculator, but with the added feature of being able to see the equation as well as the result.

Standard vibration functions such as determining acceleration, velocity and displacement can be quickly and easily calculated using the PFC as a standard calculator. Simply select the desired vibration function and supply the amplitude and frequency values.

You can create new Functions to implement new Engineering Units Definitions. These Functions can then be applied to data to convert input to new output types. For example, InchesToFeet(in) can be defined to convert data (by multiplying by 12) and using a new label called "feet".

The PFC can also be used to process arrays of data. For any size array, every expression that comprises an Output Function will be evaluated for each element of the array. All data from each Output Function can be directed to any desired source (ie a file or live data feed). For Functions that are frequency related (ie, acceleration, velocity and displacement functions), the frequency at each bin (element) is automatically calculated when the array is processed.

Complex expressions can easily be created and tested by combining Functions and constants. Any sub-expression that is used in a complex expression can be view and evaluated apart from the expression in which it is being used. New functions can be created "on the fly" while evaluating a complex equation. This helps to break the problem into smaller, re-usable components and to test intermediate results.

Labels that are associated with Functions help qualify the results. Functions can inherit labels from other Functions (such as g to g pkpk) or replace labels as the function is being evaluated (when you calculate the velocity from acceleration, the results are report as in/sec or m/sec). You can select labels to be in English or Metric units, assign your own labels, and export these labels along with the output data.

Standard Internal Functions are:

- Engineering Units Conversion
 - rms
 - peak
 - peak to peak
 - dB
 - dBV
 - dBRR

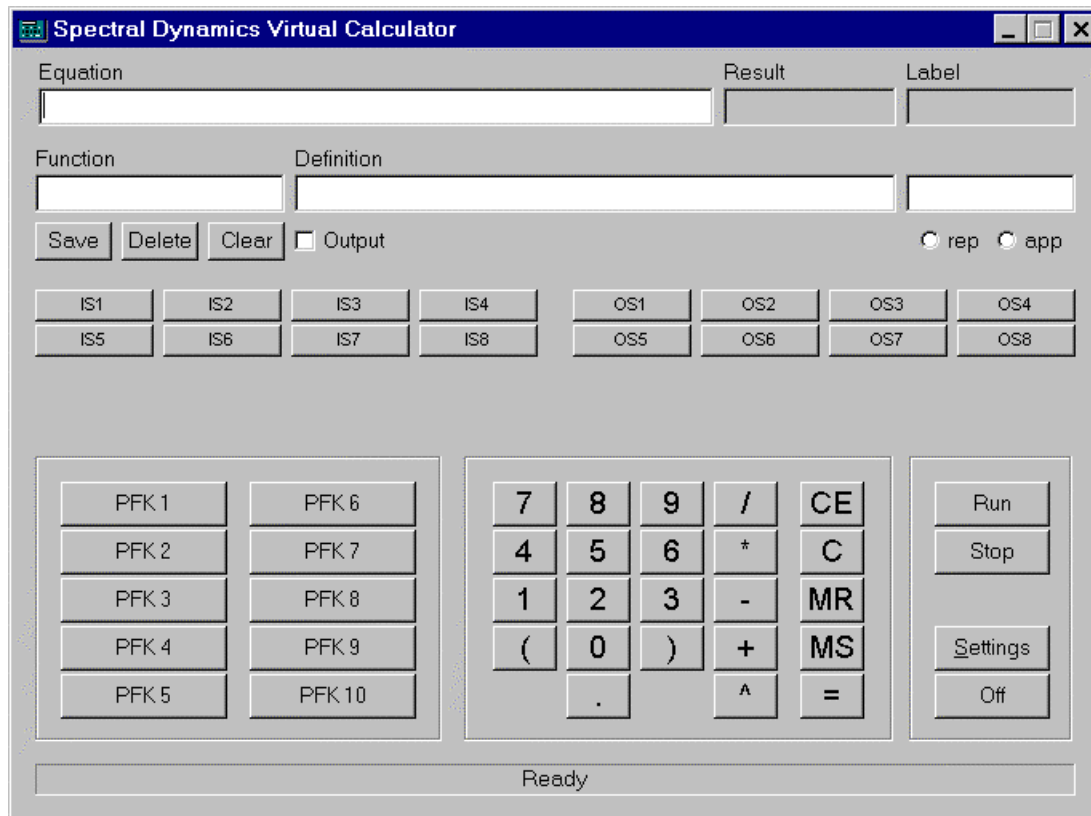
- Vibration
 - Acceleration to Velocity
 - Acceleration to Displacement
 - Velocity to Displacement
 - Displacement to Velocity
 - Displacement to Acceleration
 - Velocity to Acceleration

- Math
 - Pi
 - Pow
 - Omega
 - Integrate
 - Double Integrate
 - Differentiate
 - Double Differentiate

- Functions
 - Ramp
 - Random

Using the Programmable Function Calculator

The User Interface to the PFC is very similar to a standard calculator:



You will note the major differences are

- Input and Output Function Buttons
- Labels
- Function Programming / Definition area
- Programmable Function Keys
- Run/Stop for Stream processing

The PFC also has an additional operating mode that allows arrays (streams) to be processed and an extensive configurable settings option.

Unlike a standard calculator, additional windows and dialogs become available when you select operations that require more information than can be displayed on the calculator.

You can use the mouse to press buttons on the PFC or simply use the keyboard to enter values and Functions. Using the right mouse button brings up additional windows for displaying, selecting and entering more information. There are also keyboard "shortcuts" for many PFC operations.

There are two main input sections at the top of the calculator; the equation line and the Function/Definition line. When the input focus (a blinking vertical bar) is in the desired window, all keyboard, mouse, and button actions are directed to that window. You will use these two areas extensively when calculating equations and programming the PFC.

Getting Started

You can use the mouse and/or keyboard to access all functions of the PFC. A single click of the left button of the mouse is used to set focus to a window, depress a button or select an option. Double clicking the left button will select a choice from a list. Pressing the right mouse button when in the **Equation** or **Programming** windows will bring up the PFC **Function List**. Using the right mouse button when over a programmable function key (**PFK**) or input/output button will bring up an Assign Menu. Pressing the right mouse button any other time will bring up the PFC Main Menu.

You can also use the <**TAB**> key to move among the Equation or Programming windows and function buttons. The active window (the window with the input focus) is surrounded by a highlighted border. All keyboard, mouse, and button actions will go to that window.

You may also use the <**Delete**> key whenever a Delete function is available. This is typically when you want to delete an item from a list. You would first right click on a choice (or use the up/down arrows to select a choice), then delete it. The selected choice is always highlighted.

The PFC Main Menu

The Main Menu has four options:

| | |
|----------------------|--------------|
| Function List | Alt-F |
| Monitor | Alt-M |
| Settings | Alt-S |
| Quit | |

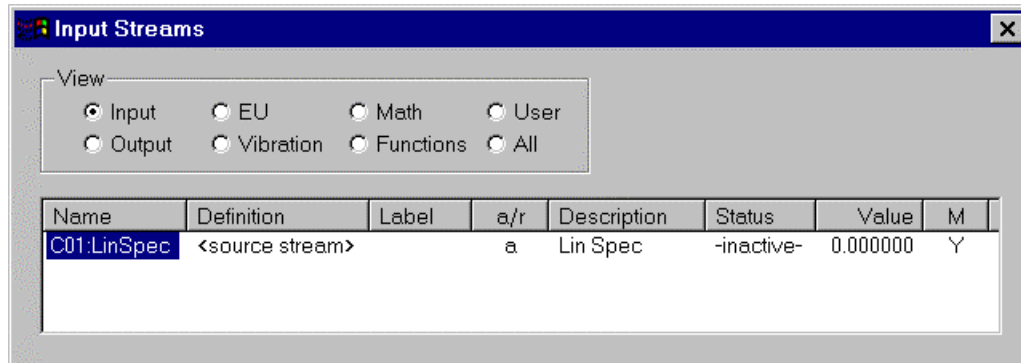
Selecting Function List will present the Function List; Monitor will bring up the monitor window, if it has been enabled; Settings will bring up the Settings Dialog; and Quit will exit the PFC. You can use the Alt-x keyboard short cuts instead of using the Main Menu.

The Function List

The Function List is the easiest way to view and select any PFC function. The Function List groups functions into seven categories:

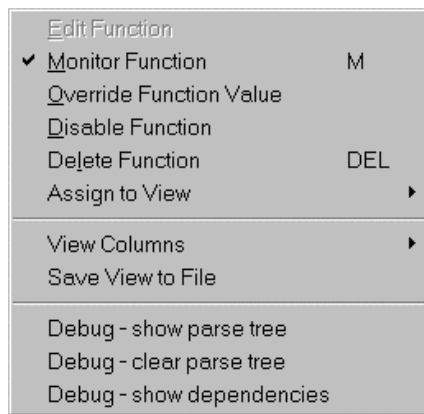
- Input
- Output
- Engineering Units
- Vibration
- Math
- Function
- User Defined

You select the desired category (or All), and the function name, definition, label, description and other information for each function in the category is displayed on the List. You can select a function by clicking with the left mouse button or using the up/down arrow keys. You can insert a function into the Equation or Programming windows by double clicking on the desired function.

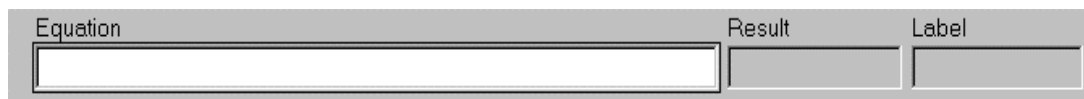


Using the right mouse button while in the Function List will bring up the Function List Menu. From this menu you can:

- Select which properties of the function to display in the list
- Move a function from one category to another
- Delete the function
- Add the Function to the Monitor window
- Set a specific override value for the function



Equation Section



The Equation Section is the top line of the calculator and is used to enter equations for immediate evaluation. The Result and Results Label (if appropriate) are shown to the right of the Equation input window.

Programming Section



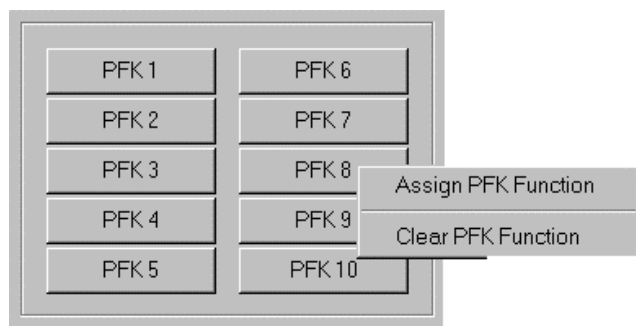
Below the Equation Section is the Programming Section. Here is where you can define new Functions, review the definition of existing functions, and evaluate results of functions.

When you double click on a sub-expression in the Equation Section, the sub-expression will appear in the Programming **Function** input window if it is a defined function. You can review its definition and label, and make changes to the function.

When you create a new Function, you type the function name and arguments into the Function input window, then press <TAB> or click the mouse in the **Definition** window and use the calculator to create a definition equation for the function. You can also assign a label to the new function, and test the function immediately by evaluating the function as you create it.

You can also modify an existing Function by right clicking in the **Function** window to bring up a list of functions. Double click on the desired function name to enter this function into the Function window, or highlight the function name using the up/down arrow keys and press <Enter>.

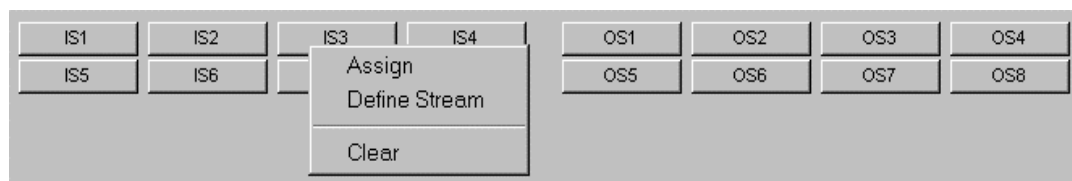
Programmable Function Keys



You can assign frequently used functions to 12 Programmable Function Keys (PFKs) by clicking the right mouse button over the desired PFK and selecting **Assign PFK Function**. Select the desired function from the **Function List** and the function name will appear on the PFK. You may then press the PFK to insert the assigned function into an equation.

You may replace any function assignment by selecting another function for the desired **PFK**, or remove a function by selecting **Clear PFK Function**.

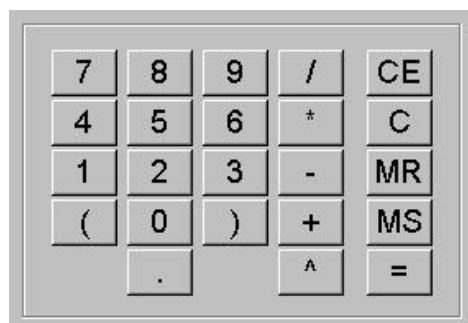
Input/Output Function Assignment



Input and Output streams can also be assigned to buttons. Input streams are assigned to buttons by right clicking over the desired button and assigning a defined input stream using the drop down menu **Assign** option. Output streams are similarly assigned. Input streams can also be defined at the time of assignment by right clicking over the desired button and selecting **Define** from the menu. You may then use the input or output button to insert the input or output stream into any equation.

You may reassign any input or output button at any time by selecting another input or output stream, or clear the button using the **Clear** option from the drop down menu.

Key Pad



The Calculator has a standard Key Pad with all numerals, decimal point, and common arithmetic operators. The Key Pad also has open and close parentheses to group sub-expressions with explicit precedence. Memory Store (**MS**) and Memory Recall (**MR**) will store and recall the most recent result. Clear (**C**) will clear the Equation completely and Clear Entry (**CE**) will clear only the most recent operand, operator, or function.

The = button will evaluate the equation and place the results in the results window. If any functions are used that have a label, the label will be placed in the equation label window. You may evaluate the equation as you are building the equation to view intermediate results.

Using the PFC as a Calculator

You can use the PFC as you would use any standard calculator to evaluate any equation that you create. Simply place the focus on the Equation Window and press the calculator Key Pad buttons or type the equation from the keyboard. The PFC displays the complete equation so that you can review the sub-expressions in complex equations. Press = at any time to evaluate the expression.

You would not typically use Input and Output streams in these equations, unless you are building a complex function that you wish to test before you save it as a user defined Function. In this case, you would elect to use the Internal Source for the Input and Output Streams, and you can assign values to the Input Streams for testing and the Output Streams would hold the output results.

You can use any Function you wish when using the PFC as a calculator. When using Functions, be aware of the number of arguments that the functions require. You may omit arguments for some internal Functions, in which case global values will be used. Also be aware of Functions that you have created that use Input and Output Streams; you will be warned if they are not available when you use the PFC in Calculator mode.

Programming the Calculator

You can easily create new functions by specifying a function name in the **Function Window** and associating a constant or any other defined function in the **Definition Window**. You may also optionally add or change the label and the label rule (append or replace).



The screenshot shows a software interface for defining functions. It features two main input fields: 'Function' and 'Definition'. Below the 'Function' field are three buttons: 'Save', 'Delete', and 'Clear'. To the right of these buttons is a checkbox labeled 'Output'. On the far right, there are two radio buttons labeled 'rep' and 'app', representing 'replace' and 'append' label rules respectively.

If the function you are defining has arguments, you denote the argument(s) by a preceding dollar sign (\$) and use the argument in the definition. Eg:

| Function | Definition | Label |
|-----------------|---------------|-------------|
| MyFunction(arg) | pk(\$arg) + 2 | Bigger Peak |

In this example, MyFunction is created with the parameter 'arg'. The definition uses the internal function **pk()** (which also takes a parameter) and the constant '2'. 'arg' is the symbolic name of the argument, which will be replaced by a value when you use the function. MyFunction(2) evaluates to $pk(2) + 2$ or 4.818.

Note: you can not use operands (+, -, *, /, ^) in a function name, ie Acc->Disp is illegal because it contains the – symbol).

You may optionally assign a new label to the function and specify the label rule as append this label to existing labels, or replace the existing label with this function's label.

To save this function, press **save**. The Function will be added to the *Function List*. You can then use this function in any equation or another function, and you can assign it to a **PFK**. Unless you are specifically creating a Function that will be written to an output stream, be sure that the **Output** radio button is off.

Parameter Substitution

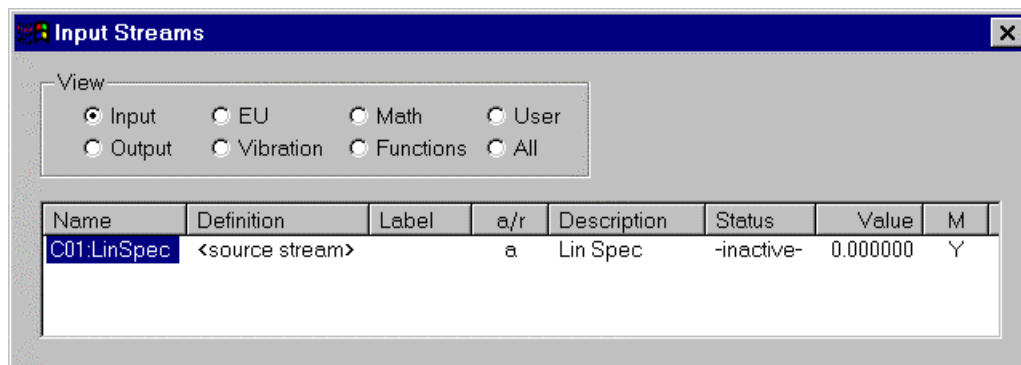
When you select a function that has parameter(s), the parameter(s) are symbolic representation for the functions parameter(s). To replace the symbolic representation with a value or another function, highlight the parameter by double clicking on it, then enter the desired value. The symbolic representation will be replaced with the actual value. For example:

AccelToVel([value @ freq](#))

value and frequency are replaceable symbolic parameters. When you double click on either symbol, it will highlight to indicate that it can be replaced.

Function List

All internal and user defined functions can be accessed from the Function List. The Function List can be viewed by pressing the right mouse button over the body of the calculator (any part of the calculator that isn't a button or window) and you can select from the Function List when the focus is in the Equation Window, Function Window or Definition Window, or when you **select Assign PFK Function** from a programmable function key.



The Function List allows you to view and select functions by category by selecting the corresponding radio button:

- Input
- Output
- Engineering Units
- Vibration
- Math
- Function
- User Defined
- All

The list shows:

- Function Name with parameters
- Definition
- Label
- Label Rule (append or replace)
- Description
- Status (active/inactive)
- Override Value (if set)
- Monitor state (on/off)

You can order the list by any list heading by clicking on the heading. You can also select which properties appear on the list by using the Function List Menu...List Columns option.

When in the Equation Window, Function Window or Definition Window, you can select a function to use by pressing clicking on the function name. If you want to select a function to place into an Equation or Programming window, double click on the function. If you only wish to view the Functions, you may then dismiss the Function List by pressing the close window (x) icon.

When you right click the mouse in the Function List window, the Function List Menu will appear. In addition to selecting which properties to display, you can also

- Move a function from one category to another
- Delete the function (provided it is not an internal function)
- Add the Function to the Monitor window
- Set a specific override value for the function. This allows you to easily debug an equation with a known value.

Internal Functions

Standard Internal Functions are:

- Engineering Units Conversion

| | |
|--------------|-------------------------------|
| Rms | $1 * x$ (useful for labeling) |
| Peak | $1.414 * x$ |
| peak to peak | $2.82 * x$ |
| dB | |
| dBV | |
| dBR | |
- Vibration
 - Acceleration to Velocity
 - Acceleration to Displacement
 - Velocity to Displacement
 - Displacement to Velocity
 - Displacement to Acceleration
 - Velocity to Acceleration

Vibration units (English or Metric) are determined from the configuration as defined in **Settings...Preferences**.

- Math
 - Pi 3.159 (constant)
 - Pow x^y
 - Omega $2 * \text{Pi} * \text{freq}$
 - Integrate
 - Double Integrate
 - Differentiate
 - Double Differentiate

- Functions
 - Ramp
 - Random

Creating new EU Definitions

The PFC can be used to define and label Engineering Units Functions. Once an EU is defined, this value (and label) can be used in an equation or programmed function by selecting the desired EU from the EU Function List.

The EU is defined like any other function: by typing a name into the **Function** window, then creating a definition for this function. It is also very useful to supply a label for this EU function, and selecting either the Append or Replace label rule. When saving this function, be sure that the **Output** radio button is Off.

All newly created functions first appear in the User category on the **Function List**. You may then move this function to the EU category.

For example, if you create an EU definition to be:

| Function | Definition | Label | |
|-----------|------------|-------|-------------|
| Intoft(x) | $\$x / 12$ | ft | (o) replace |

You can select Intoft from the Fuction List and supply any value, equation or stream as the argument, eg

Intoft(12) = 1

or

Intoft(C01:time)

Defined EUs can be used for further definitions, eg

| Function | Definition | Label |
|---------------|---------------------------------------|--------|
| InToMeters(x) | $\text{InToFt}(\$x) / (39/12) * 1000$ | meters |

Labels

Labels are used for annotation. They are useful in the PFC when performing calculations as a check that the result units are correct. They are also used by PUMA for graph annotation.

All input Channels have labels that are determined from the channel descriptor when an input source is defined. (If the source is PUMA or File, the channel descriptors are predefined by the source. For Internal source, the user must define the channels.) All internal functions have 'built-in' labels. When new functions are created, the PFC creates a 'best guess' output label by either appending or replacing the input label(s) by the replace or append rule assigned to the label. You may either use this label, or provide a new label when saving the function.

Sources, Sinks and Streams

The PFC has the ability to process streams of data from a number of different sources. Streams are essentially data arrays that often correspond to a 'block' of time data or spectral data. The source is where the data is coming from – internally generated, from a file, or from a live feed from PUMA.

The input data has a description indicating its channel and function type, ie Channel 1, Time or Channel 15, Mag Spec. It also has other characteristics such as its blocksize or resolution, Time or Frequency Span, and Engineering units.

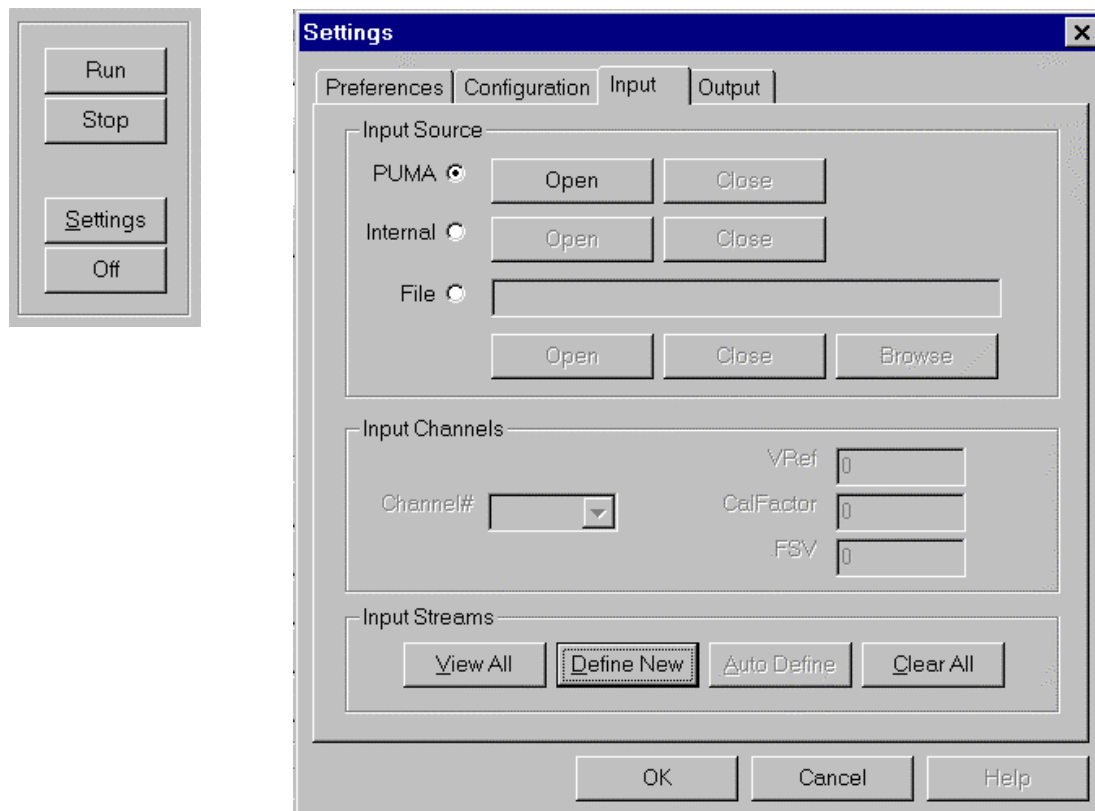
Input data originates from a channel measurement. Its data format may be raw ADC counts, volts, Engineering Units in the time or frequency domain. At the time of measurement, each measurement is assigned a channel number (typically corresponding to its ADC input channel) and provided with calibration information such as its Full Scale Voltage and Sensitivity. This information is available for review when an Input Source is opened.

Output Streams are a particular type of Function that can be written to a Sink. Sinks are the output counterpart to Source and can be internal, to a file, or to a live feed into PUMA. The output data has the same characteristics as the input data, but its description is different.

Streams are processed when **Run** is selected.

Opening a Source and a Sink

In order to use streams, a source and a sink must be opened by selecting the appropriate tabs on the **settings** page. You may select Internal, File or PUMA as the Source and Sink.



When you select File as a Source or Sink, you must also provide the file name to use.

After selecting the Source and Sink, you must **open** them in order to use input and output streams when **Run** is pressed. You may **close** Source and Sink when they are no longer needed, or close them and open new Sources and Sinks. Sources and Sinks must be closed to define and modify streams.

PFC V1.0 Source and Sinks must be the same.

- When a PUMA source is opened, the PUMA sink is automatically opened.
- When a File source is opened, you must specify a file name. Opening a File Sink, is optional, but if a File Sink is opened, you must provide an output file name.
- Internal Source does not have a Sink.

Channel Definition

Physical input channels must be defined before a Source can be used.

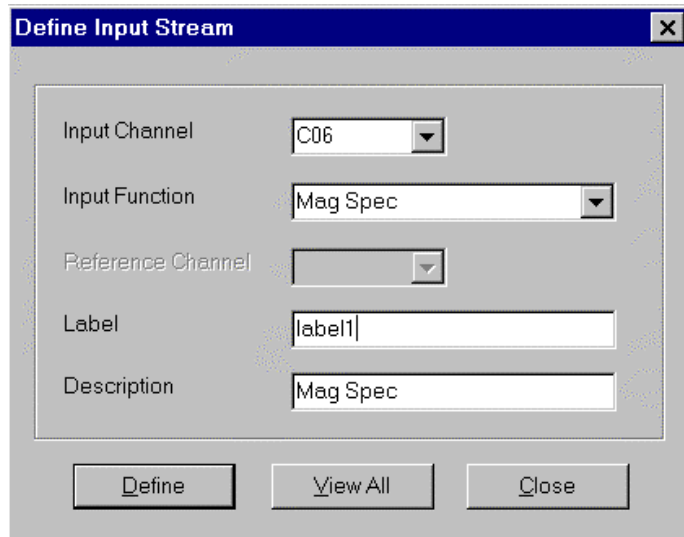
- For Internal Source, channels are initially defined when the Source is opened.
- For Source File, the channel definitions are read from the file when the source is opened.
- For Source Puma, channel definitions are not available until after the first frame becomes available from PUMA (ie after the Source is opened and the **Run** button pressed).

Input channels can be reviewed and/or redefined after a Source is opened. The **Input Channels** section of the **settings... source** page becomes active and you may review or modify channel parameters by selecting a **Channel Number**. The following parameters are then displayed for the selected channel and can be modified if required:

| | |
|-----------|--|
| FSV | Full Scale Voltage |
| Vref | Reference Voltage |
| CalFactor | Calibration factor in units per mVolts |

Defining Input Streams

Input Streams are the data that is available from the Source. Input Streams must be defined before they can be used. They can be defined individually, using **Define New**, or all streams from the Source can automatically be defined, used **Auto Define**. The Source must be closed in order to define Input Streams.



Defined Input Streams are always available to a Source. However, the source may not produce the defined Input Stream, therefore any particular Input Stream may not be available as an input function and any dependent output function. After a source is opened, you may view the status of the Input Streams using the View All option. The status will appear as +mapped+ if the stream is available to the source or +unmapped+ if it is not.

You may save various Input (and Output) stream configurations as named configuration files using the **Settings...Configuration Save** function.

Input Streams have the following nomenclature:

Cnn:Function

Where nn is the channel number, and Function is the function type available from the source. Eg:

C01:MagSpec

C02:Time

C15:Xfer(C01) - this is the nomenclature when a function requires a reference channel

Individual Input Streams can be defined or modified using the **Define** button, which brings up the following dialog:

Input Channel:
Input Function:
Reference Channel: (if function required)
Label:
Description:

You can also define an input stream and assign it to an Input Button by right clicking on an Input Button, which presents the **Assign/Define/Clear** menu so you can define a new input stream and assign it to an Input button in one step.

(The Input Buttons are programmable function keys, but are restricted to Input Streams. The **Settings...Preferences** allows you to set the number of Input Buttons to display if Show is enabled.)

You may use Auto Define to define all streams that are associated with a source. The source must first be opened, then use the Auto Define option. This option cannot be used when the Source is PUMA.

- When the Source is internal, the default function MagSpec will be set for all Input streams. The number of Input Streams is set by the number of channels (as defined on the **Settings...Configuration** tab). Any default can be changed using the **Define** Input Stream Dialog. When using Internal Source, an initial value can be supplied by using the Function List **Override Value** menu option.
- When the Source is FILE, the Input Stream defaults will be read from the file.
- When the Source is PUMA, the Input Stream list is initially empty, and input streams **MUST** be individually defined.

You can use **view All** to review all defined streams. Individual streams may be removed from the list by highlighting the desired stream, then pressing **<Delete>** or right clicking and selecting **Delete** from the menu. You may also remove all streams by pressing **Clear All** from the **Input Settings** page.

View All shows the following information for each defined Input Stream:

| | |
|-------------------|-----------------------------|
| Input Stream Name | |
| Definition | |
| Label | |
| Label Rule | a for append, r for replace |
| Description | |
| Status | mapped or inactive |
| Value | its current value |
| Monitored | Yes or No |

You may select which attributes are displayed in the list by right clicking in the list to bring up the List menu, and selecting **Columns**. A checkmark next to the attribute indicates that it will be displayed on the List. You may also print the list by selecting **Print**.

You can add any input stream to the Monitor by highlighting the desired input stream name and selecting **Monitor** from the List menu.

Defining Output Streams

Output Streams are like any other defined function, except that they can be written to a Sink and associated with an Output Button. Output streams can have any name.

Output Streams are created by specifying a Function Name and a definition, then selecting the **Output** radio button below the programming window before you **save** the Function.



When the Output radio button is off, the Function is saved as a user defined function; When Output is selected, the Function is saved as an output stream.

You can associate an Output button with an Output Stream by right clicking on the button and selecting the **Assign** menu option, then selecting the desired output stream from the Function List. You can also remove an assignment by using the **Clear** menu option.

The **settings...Preferences** allows you to select the number of Output Function Keys to display if Show is enabled.

Using Input Streams

Input Streams can be used when creating new user defined functions or Output Streams. They can also be using in Equations. They are used by right click the mouse and selected the desired input stream function from the Function List, by pressing, the associated input button, or by typing the Input Stream name into the Function Definition Window, or Equation Window. The input appears as its full name, eg

C01:MagSpec

Using Output Streams

Output Streams can be used when creating new user defined functions or other Output Streams. They can also be using in Equations. They are used by right click the mouse and selected the desired Output stream function from the Function List, by pressing, the associated output button, or by typing the Output Stream name into the Function Definition Window, or Equation Window.

Changing Sources

If a source is changed when input and output streams are defined, the Function List will show which streams are available as status `+mapped+` or `+inactive+`. Inactive sources will be disregarded in any dependent function and the dependent function results will be undefined (or all zeros). If an input stream is not available from the source due to an upstream restriction (read error, disabled channel), the output written to the sink will be also be undefined. Therefore, be sure to check the status of the input streams when changing sources.

You will receive a "Parse Error" warning message for each dependent output function that references an inactive Input Stream. The missing Input Stream will be named in the warning message.

Summary

The following steps are needed to define and use Input and Output Streams:

1. Define the Input Streams
2. Create Output Functions to be written to the Sink
3. Select a Source
4. Open the Source
5. Review the Input Channel Descriptions, if desired
6. Select A Sink (only if using File as a Source)
7. Open the Sink (only if using File as a Source)

Once the streams are defined, they can be saved in a named configuration, and recalled to be used with any Source or Sink. If a different Source or Sink is used that may not contain all the defined Streams, these streams will have a status of "Inactive" and will not be used.

Run Mode



Input and Output Streams are processed in Run Mode. The Input Streams are read from the opened Source and the Output Streams written to the Opened Sink. Streams are processed one Frame at a time. A Frame consists of all the mapped Input Streams. Output streams will be added to the Frame. Stream data can be Frequency or Time domain data. Each Stream of frequency data contains a number of data points as indicated on the **Settings...Configuration** page as **Resolution**. Time data streams contain **Resolution** * 2.56 points.

A digital Monitor is available in Run Mode. Input and Output Functions can be added to the Monitor to view the data in the streams as the streams are processed. The Function List Monitor menu option is used to select input and output functions that you want to monitor. The **Settings...Preferences Update** option allows you to set the update rate for the monitor.

A Step Mode is available when processing Internal or File Sources, which allows single stepping through every n points of each Frame. This mode is set using Run Mode parameters on the **Settings...Preferences** dialog.

Using Run Mode

Using the Calculator in Run Mode is like using any standard calculator, with the main difference being that Input streams and Output Streams can be specified in the equations, and that vector operations will be performed when input streams are used in Run mode.

The Input Streams you wish to use are defined using **Settings...Source**. You may create equations using these input streams and create functions or output streams from these equations. Any Output Stream that is saved will be processed in Run mode.

An Output Stream is created by giving the stream a name in the Function Window, then using the calculator to create an equation by keying the appropriate functions and constants. Intermediate functions can be created to hold intermediate results for other calculations by saving them as Functions. Input and Output Streams and any other functions can be used in the equation by pressing the appropriate programmable function keys or selecting the desired function from the appropriate function list.

An example equation window would look like:

$$Fn2 = (C01:MagSpec * 1.414) + Fn1$$

Fn2 would then be saved as an Output Stream. This Output Stream is then available to the defined Output Source, such as live plotting in PUMA.

Defined EU Functions can also be used by selecting the desired EU conversion Function from the Function List. For example:

$$\text{Fn3} = \text{pk}(\text{C01:MagSpec}) + \text{Fn1}$$

A Label for any Function can be assigned by typing any text into the label window, or using the label has been previously assigned to that function.

Run Mode Processing

Input and Output Streams must first be defined, and Input and Output (optional) Sources opened. **run** is then pressed to start processing. As long as data is available from the Source, the PFC will continue to process the data and write the Output Streams to the Sink. **stop** can be pressed to stop the processing of data.

The PFC can be configured to step through data in each frame instead of continuously processing the data. This may be useful in debugging or looking at data from files, but would not typically be used when the Source is a live data feed. You may set **step** operation and a Step increment from the **Settings...Preference** page.

Input and Output Streams can be monitored by adding them from the Function List to the Monitor List. This is accomplished from the Function List Menu using the **monitor** option. You may view the Monitor List by selecting **monitor** from the PFC Menu or **Alt-M** keyboard shortcut. Details of Individual Functions can viewed from the Monitor List by right clicking on the Function in the Monitor List, or you may remove a function from the Monitor by highlighting it and pressing the **<Delete>** key. The Monitor List will only appear if it is enabled on the **Settings...Preference** page and there is at least one function on the Monitor List.

When Run Mode is active, Equations that use Input and Output Streams will be evaluated. The results from the immediate position within the Frame will be shown. The monitor shows the Total number of Frames Processed and the current point within the frame.

Note: The Internal Source is provided primarily as a programming aide. The default size of the Input Stream is one data point. The value for this data point can be specified when defining the Stream. When in Run (or Step) Mode, the same value will be used for each Frame.

PFC Configurations

The PFC can be customized for different applications and different users by saving and recalling configurations. A PFC configuration consists of:

- Preferences
- Global Parameters
- Source Settings, including:
 - * Source Type
 - * Input Stream Definitions
 - * Channel Definitions
- Sink Settings, including:
 - * Sink Type
 - * Output Stream Definitions
- User Defined Functions
- Programmable Function Key associations

If a configuration is recalled, all input and output stream definitions are replaced with those in the configuration.

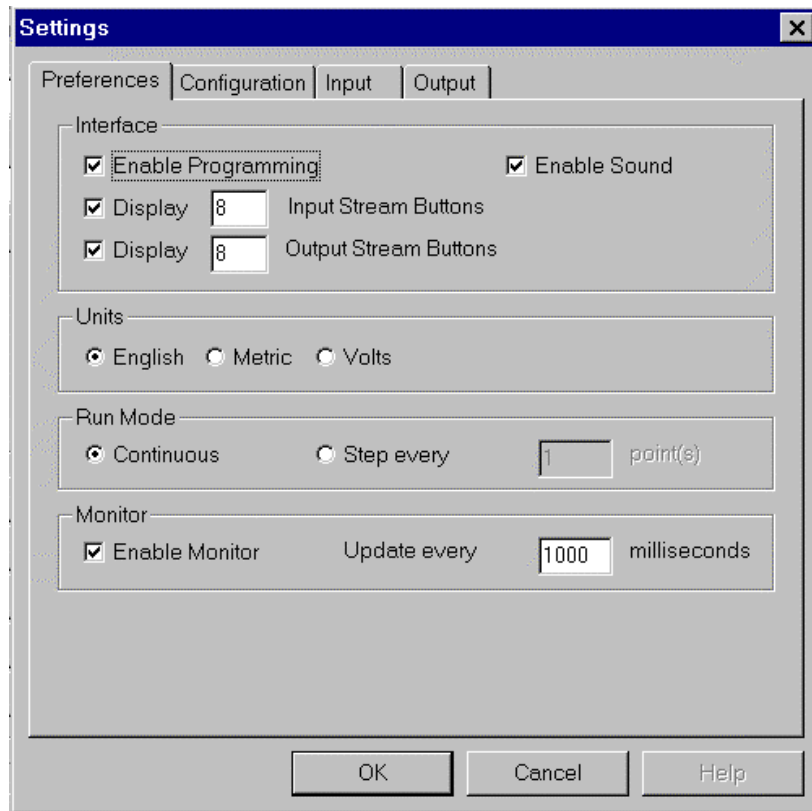
On startup, the last used configuration file is loaded, or if it cannot be found, the default configuration is used. You may use **Settings...Configuration** to load a new configuration, clear the current configuration, or save a new configuration.

Settings

Preferences, Configuratins, Input sources and Output sources can be modified from the Settings dialog by pressing the Settings button, using the Settings option of the main menu, or using the Alt-S keyboard shortcut. You can then select the appropriate tab for the changes you want to make. After you make the changes, press **OK** to save the changes, or **Cancel** if you don't want to save them.

Preferences

These settings control the general appearance and functionality of the Calculator.



Interface Options

Enable Programming – when this is checked, you can use the PFC for programming new functions. If it is not checked, the programming function buttons do not appear on the Calculator, and you cannot save new functions. You can, however, review the definitions for existing functions.

Enable Sound – sound is enabled if this is checked and you have a sound card. If it is not checked, the Calculator will operate in quiet mode.

Display Input Streams – when this is checked, the input stream buttons appears. You can also specify the number of buttons to display. If this is not checked, the input buttons will not appear on the Calculator.

Display Output Streams – when this is checked, the output stream buttons appears. You can also specify the number of buttons to display. If this is not checked, the output buttons will not appear on the Calculator.

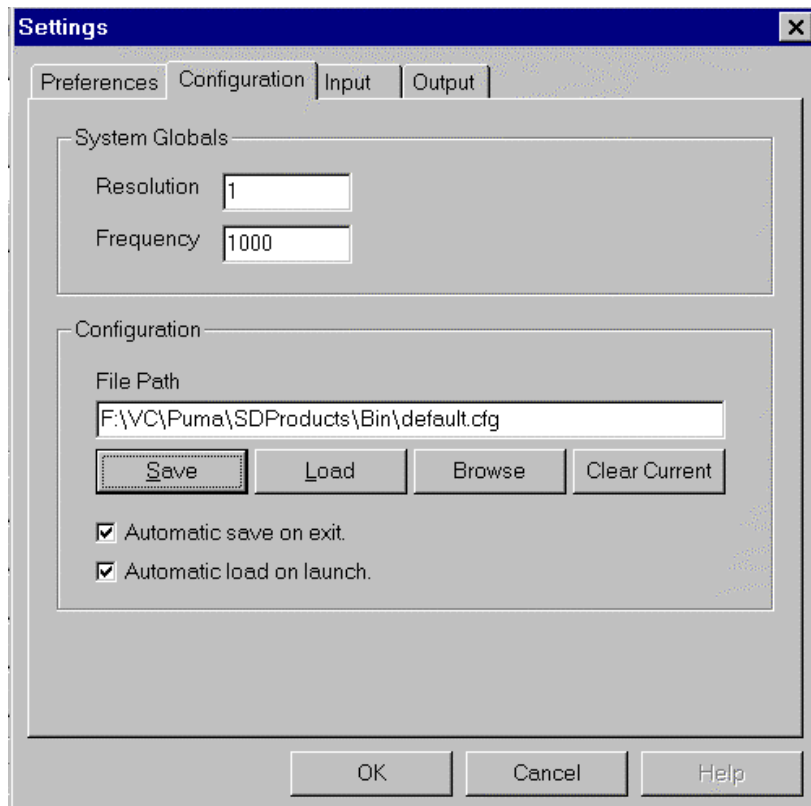
Units - You may select **English**, **Metric**, or **Volts** to be the default units. These units are used to for unit dependent constants, such as acceleration.

Run Mode – When **Continuous** is selected, data will be processed as continuous when Run is pressed. Otherwise, the Calculator will **Step** through the data at an increment of the specified number of points. Note: it is not advisable to step through data when the Source is PUMA as frames of data will be held by the Calculator until all data in the frame is processed.

Monitor – When **Enable Monitor** is checked, the Monitor List can be shown by selecting Monitor from the Main Menu or using the Alt-M keyboard shortcut. You can also specify the **update** rate in milliseconds. When Enable Monitor is not checked, the Monitor will not be available.

Configuration

These settings contain global parameters and configuration options.



System Globals

Resolution

Frequency – These are the system wide frequency resolution (number of bins or lines) and the Frequency bandwidth values. Delta F is calculated from Frequency / Resolution and is used in any functions that use Bin Frequency (ie, Omega and Vibration functions). If a Frequency parameter is omitted from any function, the System value is used. Delta T and the number of data points for time traces are also calculated from these values.

$$\text{Delta T} = 1 / (\text{Frequency} * 2.56)$$

$$\text{Npts} = \text{Resolution} * 2.56$$

Configuration

File Path – The full filename of the configuration file

Save – save the current configuration as the file specified in File Path

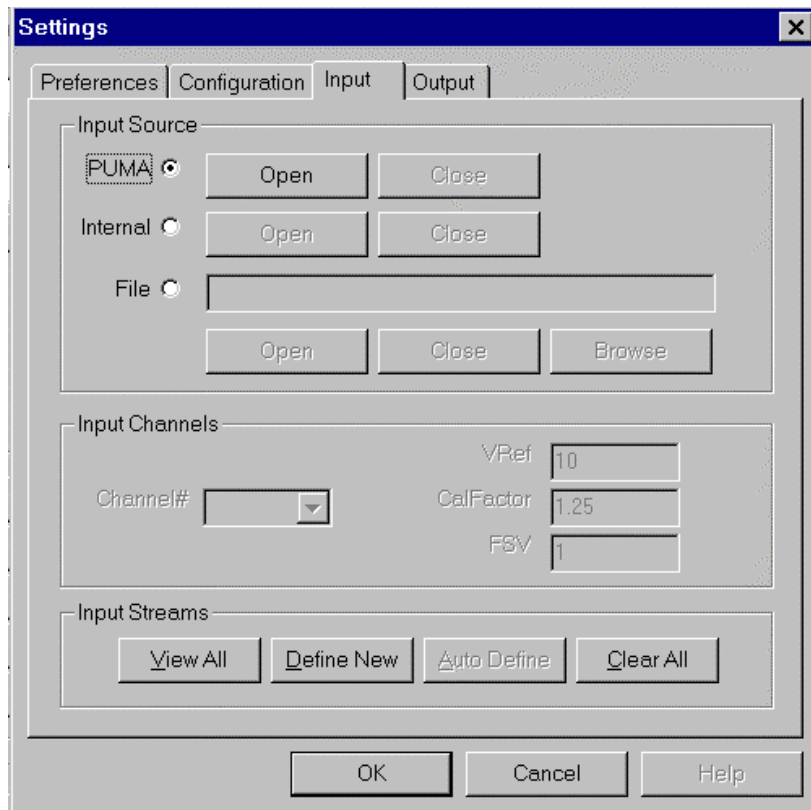
Load - load the configuration as specified in File Path

Browse – search for a configuration file or directory

Clear Current – reset the current configuration to system defaults.

Input

These functions have been previously described in detail. Following is a summary.



Input Source

Puma – This selects PUMA as an input source, and as an output sink.

Internal – Select this option if you wish to use VPC as a standalone calculator.

File – You may specify a file to **Open** or **Close**, or **Browse** all files.

Input Channels

Channel – This defines the name of the input streams

Vref – References voltage for this channel

Cal – Calibration Factor (EU/V) for this channel

FSV – Full Scale Voltage

Input Streams

View All – review all defined input streams

Define New – allows defining a new input stream

Auto Define – automatically define all input streams (for **File** only)

Clear All – remove all input streams

Output

These functions have been previously described in detail. Following is a summary.



File – You may specify a file to **Open** or **Close**, or **Browse** all files. This is only available if a **File** is opened as an **Input** stream.

View All – review all defined output streams

Clear All – remove all output streams

(Note: **Puma** output is automatically **Opened** and **Closed** when the Puma Source is opened and closed. There is no output for **Internal** input).

APPENDIX A: Simple Analysis Layout File Format

The input and output file sources are in Simple Analysis Layout File Format. This format consists of a header, followed by one or more frames of data. A frame contains one or more records of data, with each record corresponding to a channel. All data (in engineering units) for each channel comprises a record.

Each frame must contain the same number of records, and all frames must contain the same type of data, ie all data must be PSD data or all data must be Time Data. The SAL header is:

```
#define MAX_CHANS 16

typedef struct tagCHAN_DESC
{
    float rCal;
    float rFSV;
    float rDBRef;
    char cLabel[40];
} CHAN_DESC;

typedef struct tagDATA_HEADER
{
    int iDataType; // Time, PSD, MagSpec (see unit_id.h)
    int iNChans; // number of channels in each frame
    int iNVChans; // not used
    float rFreq; // Frequency Range
    int iResolution; // number of lines (or npts for time)
    int nFrames; // number of frames in the file
    CHAN_DESC ChanDesc[MAX_CHANS];
} DATA_HEADER;
```

The input SAL file is:

```
Data Header
Data format...
Frame 1
float chan1, bin0 ... bin iResolution
....
float chanN, bin0 ... bin iResolution
Frame N...
```

The output SAL file is:

```
Modified Data Header with number of output functions
Data format...
Frame 1
float OutFunction1, bin0 ... bin iResolution
...
float OutFunctionN, bin0 ... bin iResolution
Frame N...
```

APPENDIX B: Glossary

Physical Channel (or Channel) - data channel of acquired data. A Channel is described by its number, Sensitivity and other parameters.

Input Stream – A function defined in the PFC which contains data provided to the PFC from a designated source. Denoted as Cnn:Func

Output Stream – A function created by the PFC which contains output data.

Equation Generator - the PFC will create an equation involving Input Stream(s), Output Streams(s) and or constants and operands or functions. This can be considered the programmin step in using the PFC. These equations will then be processed when input data streams are available and create output data streams (the processing (RUN) step).

Simple Equation - a one or two term equation, involving a single operand, e.g. $A + B$ or $1/C$

Complex Equation - an equation containing more than one operand, e.g. $A + B * C$. Each operand is computed in the order discovered, e.g. $1 + 2 * 3 = 9$.

Complex Equation with precedence - a Complex equation that is parsed by algebraic precedence and may contain parenthesis for grouping, e.g. $1 + 2 * 3 = 7$, or $(1 + 2) * 3 = 9$

Input/Output Stream - an abstract stream of data. The stream will contain a data header followed by a block of data.

Data Header - a description of the data to include at a minimum:

Channel Indicator (i.e. C1 or VC12)

BlockSize - number of points in the data

Data Type - Time, Frequency, Complex

Sample Rate or Frequency Range

Process Server - PUMA's implementation of a routine to dispatch Processing Clients, one of which will be the VC. The Process Server will describe its API.

Expressions

Standard operators

Standard operators join terms within a function definition. Currently +, -, *, and /. Also known as conjunction operators.

Term delimiter

Term delimiters end parsing of a function name or parameter name. Currently <space>, :, (,), and any *standard operator*.

Functions

An object that contains terms and can be evaluated to perform a transform on one or more data items sources from input channels or the results of previously evaluated functions. A function may contain parameters.

Predefined Functions

- C1 - Cn, evaluate to input channel data.
- V1 - Vn, evaluate to output channel data (virtual channels)
- mathematical constants, e.g. PI, evaluates to 1.414.

- mathematical functions, e.g. SQRT(), SQR(), LOG(), Invert() etc.

User-defined Functions

- Functions can be created by a user for specific applications by using the keyboard and the calculator pad and mouse.
- Functions can use any other previously defined function in their definition, combined using standard operators.
- Functions can take parameters.

Engineering Unit Functions

- These functions are a form of User Defined Functions.

Function Representation

The displayed definition of a function, i.e. the right hand side of the expression:

<FunctionName> = <FunctionRepresentation>.

Function Forms Supported

- **Simple Form**

Fn = <constant>

example: PI = 1.414

This is essentially a named constant

Fn = <channel>

example: Accelerometer = C1

This would alias an input channel

Fn = <channel operand channel>

example: SumC1C2 = C1 + C2

- **Using parameters in representation**

- example 1: OffsetC1By(offset) = C1 + :offset

- example 2: OffsetChannelByGlobal(channelNum) = :channelNum + ::globalOffset

- example 3: Difference(channel1,channel2) = :channel1 - :channel2

- **Nesting functions**

- example 1: V1 = (C1 + C2) - offset

- example 2: V1 = Difference(SumC1C2, OffsetC1by(::globalOffset))

Function Name

The function identifier, e.g., “C1”, “Invert”, “MyTransform”. The function name is recognized during parsing of a function definition.

Function Parameter

A term that serves as a substitution variable within a function definition, identified by a leading colon, e.g., “Fn = C1 + :k”.

Parameter Name

The text identifying a specific parameter. Any non-delimiter text can be used to define a parameter name. Parameter names begin with a leading colon.

Function Argument

- A parameter name contained in a Function Name, e.g. Fn(param1,param2). “param1” and “param2” are function arguments. Function arguments can be the evaluated result of other functions, e.g. Fn(Fx(Fy),PI).
- Function argument values are assigned to function parameters from left to right in the function definition. For example, for the function definition “Fn = C1 + :a/:b” called with function arguments as in Fn(1,2), :a will be assigned the value 1 and :b will be assigned the value 2. A missing parameter value will be detected during syntax checking after the user selects “store”.

Global Parameters

Global parameters are parameters used in function definitions which do not require or correspond to a function argument. They are identified by a double colon preceding the parameter name, e.g., “::samp_freq”.

- **Predefined Global Parameters**

This is a parameter built in to the VC and initialized from data header information, or an initialization file (INI), or registry value.

- **Runtime Global Parameters**

This is a parameter that is assigned a value after all calculator functions are defined, but before the processor begins execution. Typically a dialog box will appear for inputting values to all undefined runtime global parameters.

Processing

Function definition mode (Setup or Program mode)

- This mode allows the user to define custom functions for each output channel.
- A function name edit box allows the user to enter unique function names or choose from a dropdown combo box one of the predefined functions already supplied by the application or which were previously loaded from a configuration file.
- If a predefined function is chosen, the function name appears in the function name box and the current definition appears in the function definition edit box where it can be modified.
- If a new function name is entered, the function definition edit box is cleared and the user enters the new definition.
- A listbox of currently defined functions is always available to choose from for inserting known functions as a subterm into the current function definition.
- When the definition is complete, the user presses a “store” button. At this point, the function definition is parsed for syntax errors, undefined function references, or missing parameter arguments. On error, a message is displayed and, when possible, the offending token is highlighted. On successful parsing, the function becomes part of the known function list and can then be inserted into subsequent function definitions.

I/O configuration

Output

- A method is supplied for assigning virtual output channels to physical output channels (actual channel numbers in the output sink).

- A method is supplied for enabling and disabling output channels without removing their function definitions (a toggle).
- A method is supplied for assigning output labels to each output channel.

Input

- A method is supplied for examining all parameters of an input channel (read from the source header).
- A method is supplied for examining overall source parameters (read from the source header).

Configuration Persistence

Named files containing function definitions, I/O configurations, and other preference sets can be loaded from and saved to disk.

Output channel evaluation (Runtime Mode)

- For each output (virtual) channel defined, there exists a function (a term) to evaluate it's value. Default definition for all output channels will simply be the assignment of the input channel value, e.g., $V1 = C1$, $V2 = C2$, etc. ("pass through" mode).
- The read process is started by a "EOFrame" semaphore or by selecting "run" (in debug or evaluate mode).
- A value is read in from the CInputSSource class for each input channel ($C1 \dots Cn$).
- Values are read as fast as they can be evaluated or manually one at a time (single step debug mode).
- For each read cycle, all output channels have their evaluation method called.
- If the output channel term has subterms, the evaluation method calls the evaluation method for each subterm and combines the results (using each term's conjunction operation) into an accumulator. Any subterm that also contains subterms is evaluated recursively in the same manner. The accumulator becomes the return value for each term.
- Finally, the channel term passes it's accumulator value to the COutputSink class as it's output value.
- Possible variation: Instead of running from the VCProcessor, the user selects "single step" which reads only one row of input channel data and allows inspection of output values as described below in "Calculator Mode". Or, a method exists for stepping "n" times (defaulting to 1).

Output channel evaluation (Calculator Mode)

- Output channel function definition is identical to Runtime Mode.
- In calculator mode, the input channel values are supplied by the user and stored in static registers. The user presses an input channel button and the current value for that channel is displayed, e.g. $C1 = 2.345$. If the user presses number keys and then "store", the newly entered value is used for that channel in subsequent operations.
- To evaluate an output channel in calculator mode, the user presses a button (one each appears for all defined output channels) for that channel, and the channel's function is evaluated using the current input channel register values and the result printed in the display window, e.g. $V1 = 0.2221$.
- Possible variation: Perhaps calculator mode is just a method for displaying individual output channel values using the "current" input channel values. The "current" input channel values can be changed by the user (with "store"), or read from the input source (with "single step").

APPENDIX C: Quick Start Example

As an example, the following steps will create a mag spec input stream for channels one and two, create an output stream of ch1 + ch2, and display the new waveform in the Random application.

- 1) Create a short cut for the Virtual Calculator on the Windows desk top.
- 2) Start the Virtual Calculator.
- 3) Right Mouse click on the IS1 button and select Define Stream with the left mouse button.
- 4) When the define input stream box appears, use the input channel drop down menu to select C01.
- 5) Under label, enter ch1mag followed by a <CR>.
- 6) Click OK to the input stream created notification. You have just created a input data stream.
- 7) Verify that C01:MagSpec appears in place of IS1 on the button.
- 8) Right mouse click on the IS2 button and repeat steps 4 through 6 to define a channel 2 Mag Spec input stream (substitute C02 for C01 and ch2 for ch1 in steps 4 through 6).
- 9) Verify that the IS2 button now displays C02:MagSpec.
- 10) Left mouse click in the function box and enter ch1+ ch2. This will be the name of the new function.
- 11) Left mouse click in the Definition Box. With the cursor flashing in the definition box, left click the input stream box C01:MagSpec. C01:MagSpec should appear in the Definition box.
- 12) Enter a + sign at the end of C01:MagSpec in the definition box, then left click the C02:MagSpec input stream box. The Definition box should now display C01:MagSpec+ C02:MagSpec. This entry defines the function of ch1 + ch2.
- 13) Left click in the white box to the right of the definition box and enter c1 + c2 mag. This will be the function label that will be displayed when the function is displayed later in Random.
- 14) Click the output box beneath the definition box and when a ✓ appears, click the save button.
- 15) Click OK to the output stream created notification box. The newly defined output stream ch1 + ch2 has just been created.
- 16) Exit the Virtual Calculator application.
- 17) Start the Random application.
- 18) Setup channel one as a control channel and channel two as a measurement channel. Note: All channels that are defined as input streams in the Virtual Calculator must be enabled before re-starting the Virtual Calculator.
- 19) Start the Virtual Calculator.
- 20) In the Virtual Calculator, click the settings button in the lower right corner.
- 21) Select the input tab.
- 22) Select Puma as the input source and click open. This selects the source for the input data streams to be live data from Puma.
- 23) When the open button is ghosted, click OK.
- 24) Click Run on the Virtual Calculator.
- 25) Verify that the word Running is displayed at the bottom of the Virtual Calculator. The new data stream is now being passed to Puma.

- 26) In the Random application, select a plot to display the plot definition table.
- 27) Scroll to the bottom of the plot definition function box and select ch1+ch2. The new function is now selected.
- 28) Select channel 1 as the display channel (all output data streams from the Virtual Calculator are displayed on channel 1).
- 29) The new wave form function $ch1 + ch2$ $c1 + c2$ mag should now appear. The function $ch1 + ch2$ is the function name, similar to the function PSD, and the function label $c1 + c2$ mag would be equivalent to G^2/Hz .