

LabVIEW[®]

***Statistical Process Control Toolkit
Reference Manual***

National Instruments Corporate Headquarters

6504 Bridge Point Parkway

Austin, TX 78730-5039

(512) 794-0100

Technical support fax: (800) 328-2203

(512) 794-5678

Branch Offices:

Australia (03) 879 9422, Austria (0662) 435986, Belgium 02/757.00.20, Canada (Ontario) (519) 622-9310,

Canada (Québec) (514) 694-8521, Denmark 45 76 26 00, Finland (90) 527 2321, France (1) 48 14 24 24,

Germany 089/741 31 30, Italy 02/48301892, Japan (03) 3788-1921, Netherlands 03480-33466, Norway 32-848400,

Spain (91) 640 0085, Sweden 08-730 49 70, Switzerland 056/20 51 51, U.K. 0635 523545

Limited Warranty

The media on which you receive National Instruments software are warranted not to fail to execute programming instructions, due to defects in materials and workmanship, for a period of 90 days from date of shipment, as evidenced by receipts or other documentation. National Instruments will, at its option, repair or replace software media that do not execute programming instructions if National Instruments receives notice of such defects during the warranty period. National Instruments does not warrant that the operation of the software shall be uninterrupted or error free.

A Return Material Authorization (RMA) number must be obtained from the factory and clearly marked on the outside of the package before any equipment will be accepted for warranty work. National Instruments will pay the shipping costs of returning to the owner parts which are covered by warranty.

National Instruments believes that the information in this manual is accurate. The document has been carefully reviewed for technical accuracy. In the event that technical or typographical errors exist, National Instruments reserves the right to make changes to subsequent editions of this document without prior notice to holders of this edition. The reader should consult National Instruments if errors are suspected. In no event shall National Instruments be liable for any damages arising out of or related to this document or the information contained in it.

EXCEPT AS SPECIFIED HEREIN, NATIONAL INSTRUMENTS MAKES NO WARRANTIES, EXPRESS OR IMPLIED, AND SPECIFICALLY DISCLAIMS ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. CUSTOMER'S RIGHT TO RECOVER DAMAGES CAUSED BY FAULT OR NEGLIGENCE ON THE PART OF NATIONAL INSTRUMENTS SHALL BE LIMITED TO THE AMOUNT THERETOFORE PAID BY THE CUSTOMER. NATIONAL INSTRUMENTS WILL NOT BE LIABLE FOR DAMAGES RESULTING FROM LOSS OF DATA, PROFITS, USE OF PRODUCTS, OR INCIDENTAL OR CONSEQUENTIAL DAMAGES, EVEN IF ADVISED OF THE POSSIBILITY THEREOF. This limitation of the liability of National Instruments will apply regardless of the form of action, whether in contract or tort, including negligence. Any action against National Instruments must be brought within one year after the cause of action accrues. National Instruments shall not be liable for any delay in performance due to causes beyond its reasonable control. The warranty provided herein does not cover damages, defects, malfunctions, or service failures caused by owner's failure to follow the National Instruments installation, operation, or maintenance instructions; owner's modification of the product; owner's abuse, misuse, or negligent acts; and power failure or surges, fire, flood, accident, actions of third parties, or other events outside reasonable control.

Copyright

Under the copyright laws, this publication may not be reproduced or transmitted in any form, electronic or mechanical, including photocopying, recording, storing in an information retrieval system, or translating, in whole or in part, without the prior written consent of National Instruments Corporation.

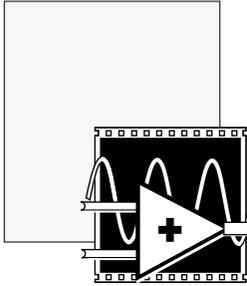
Trademarks

LabVIEW® is a trademark of National Instruments Corporation.

Product and company names listed are trademarks or trade names of their respective companies.

WARNING REGARDING MEDICAL AND CLINICAL USE OF NATIONAL INSTRUMENTS PRODUCTS

National Instruments products are not designed with components and testing intended to ensure a level of reliability suitable for use in treatment and diagnosis of humans. Applications of National Instruments products involving medical or clinical treatment can create a potential for accidental injury caused by product failure, or by errors on the part of the user or application designer. Any use or application of National Instruments products for or involving medical or clinical treatment must be performed by properly trained and qualified medical personnel, and all traditional medical safeguards, equipment, and procedures that are appropriate in the particular situation to prevent serious injury or death should always continue to be used when National Instruments products are being used. National Instruments products are NOT intended to be a substitute for any form of established process, procedure, or equipment used to monitor or safeguard human health and safety in medical or clinical treatment.



Contents

About This Manual	ix
Organization of This Manual	ix
Conventions Used in This Manual.....	x
Related Documentation	xii
Customer Communication	xii

Chapter 1

Introduction to Statistical

Process Control in LabVIEW	1-1
Installation	1-1
Windows	1-1
SPARCstation	1-2
Macintosh	1-3
Requirements for Using the SPC Toolkit	1-3
SPC Toolkit Organization	1-4
VI Libraries	1-4
Custom Controls	1-6
LabVIEW SPC Toolkit Examples	1-7
Implementing SPC Applications in LabVIEW	1-8
Representation of Process Data in LabVIEW	1-8
Viewing Raw Process Data	1-9
Creating Control Charts and	
Determining Whether the Process Is in Control	1-13
Detecting Out-of-Control Points and Process Shift	1-15
Process Capability Analysis	1-19
Pareto Analysis	1-20

Chapter 2

Control Chart VIs	2-1
Calculating Control Chart Limits and Points	2-1
Variables Chart VIs	2-2
Attributes Charts VIs	2-3
VIs for Drawing Charts	2-4
VIs for Plotting Control Chart Points and Limits	2-5
VIs for Creating Graphs of Raw Process Data	2-5
Rule Checker VIs for Testing	
Out of Limits, Run Rules, and Process Shift	2-7

Variables Chart VIs	2-7
X-Bar & s Chart	2-7
X-Bar & R Chart	2-11
X & mR Chart	2-14
mX-bar & mR Chart	2-18
Single Point X-Bar & R/S	2-22
Single Point x/mX-bar & mR	2-23
Attributes Charts VIs	2-24
p chart	2-24
np Chart	2-27
c Chart	2-29
u Chart	2-32
Draw Control Chart VIs	2-35
Draw Control Chart	2-35
Draw Chart with Zones	2-36
Draw Chart with Var Limits	2-37
Draw Run Chart	2-38
Draw Tier Chart	2-40
Rule Checker VIs	2-42
Check Control Limits	2-42
Rule Checker (AT&T/WE)	2-43
Rule Checker (Nelson)	2-44
Process Shift Detector	2-46
Sequence Checker	2-47

Chapter 3

Process Statistics VIs	3-1
Process Mean and Sigma	3-1
Compute Process Capability	3-3
Sample Statistics VI	3-5
General Histogram	3-6
Fit Nrml PDF to Histogram	3-8
Normal PDF Graph with Limits	3-9
Plot Normal PDF	3-10
Vertical Bar Graph with Limits	3-12
Vertical Bar Graph	3-13
Rotate Graph	3-14

Chapter 4

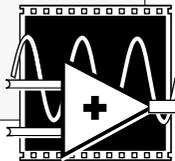
Pareto Analysis VIs	4-1
Pareto Counter	4-2
Pareto Chart	4-3
Cause Code Lookup	4-4
Array to Bar/Comb Graph	4-5

Appendix
Customer Communication..... A-1

Glossary G-1

Figures

Figure 1-1. Basic Run Chart 1-10
 Figure 1-2. Diagram for Basic Run Chart 1-10
 Figure 1-3. Basic Histogram Plot 1-11
 Figure 1-4. Diagram for Basic Histogram Plot 1-12
 Figure 1-5. Basic Tier Chart 1-12
 Figure 1-6. Diagram for Basic Tier Chart 1-13
 Figure 1-7. X-bar and S Chart Example 1-14
 Figure 1-8. Diagram for X-bar and S Chart Example 1-14
 Figure 1-9. p Chart with Variable Limits VI Example 1-15
 Figure 1-10. Diagram for p Chart with Variable Limits VI
 Example 1-15
 Figure 1-11. X-bar and R Chart Check Limits Example 1-16
 Figure 1-12. Diagram for X-bar and R Chart Check Limits
 Example 1-17
 Figure 1-13. Zone Rule Test (AT&T/WE) Example 1-18
 Figure 1-14. Diagram for Zone Rule Test (AT&T/WE) Example... 1-18
 Figure 1-15. Process Capability Example 1 1-20
 Figure 1-16. Diagram for Process Capability Example 1 1-20
 Figure 1-17. Pareto Chart Example 1-21
 Figure 1-18. Diagram for Pareto Chart Example 1-21



About This Manual

The *LabVIEW Statistical Process Control Toolkit Reference Manual* describes the LabVIEW add-on package you can use for implementing statistical process control functions.

Organization of This Manual

This manual is organized as follows:

- Chapter 1, *Introduction to Statistical Process Control in LabVIEW*, contains installation instructions, gives an overview of Statistical Process Control (SPC), and discusses the LabVIEW SPC Toolkit VIs and examples.
- Chapter 2, *Control Chart VIs*, describes the control chart VIs, which include the variables charts, attributes charts, chart drawing, and rule checking VIs. The control chart VIs compute control limits for control charts, create control chart graphs, and apply rules to control chart data that detect out-of-control conditions.
- Chapter 3, *Process Statistics VIs*, describes the process statistics VIs, which are useful for process capability analysis and for viewing and measuring process distribution.
- Chapter 4, *Pareto Analysis VIs*, describes the Pareto analysis VIs, which include the Pareto Counter VI, the Pareto Chart VI, and the Cause Code Lookup VI. The Array to Bar/Comb VI, which the Pareto Analysis VIs use as a subVI, is also included.
- The Appendix, *Customer Communication*, contains forms you can use to request help from National Instruments or to comment on our products and manuals.
- The *Glossary* contains an alphabetical list and description of terms used in this manual, including abbreviations, acronyms, metric prefixes, mnemonics, and symbols.

Conventions Used in This Manual

The following conventions are used in this manual:

- | | |
|---------------------------|---|
| bold | Bold text denotes menus, menu items, and VI input and output parameters. |
| <i>italic</i> | Italic text denotes emphasis, a cross reference, or an introduction to a key concept. Italic text also denotes a variable such as <i>filename</i> or <i>N</i> when it appears in a text passage. |
| <i>bold italic</i> | Bold italic text denotes a note, caution, or warning. |
| monospace | Monospace font denotes text or characters that you enter using the keyboard. File names, directory names, drive names, sections of code, programming examples, syntax examples, and messages and responses that the computer automatically prints to the screen also appear in this font. |



Warning:

This icon to the left of bold italicized text denotes a warning, which alerts you to the possibility of damage to you or your equipment.



Caution:

This icon to the left of bold italicized text denotes a caution, which alerts you to the possibility of data loss or a system crash.



Note:

This icon to the left of bold italicized text denotes a note, which alerts you to important information.

LabVIEW Data Types

Each VI description gives a data type picture for each input and output parameter, as illustrated in the following table.

Control	Indicator	Data Type
		Signed 8-bit integer
		Signed 16-bit integer
		Signed 32-bit integer
		Unsigned 8-bit integer
		Unsigned 16-bit integer
		Unsigned 32-bit integer
		Single-precision floating-point number
		Double-precision floating-point number
		Extended-precision floating-point number
		String
		Boolean
		Array of signed 32-bit integers
		Cluster
		File Refnum

Abbreviations, acronyms, metric prefixes, mnemonics, symbols, and terms are listed in the *Glossary*.

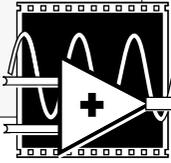
Related Documentation

The following documents contain information that you may find helpful as you read this manual:

- Your LabVIEW tutorial
- Your LabVIEW user manual
- American Society for Quality Control. *American National Standard. Definitions, Symbols, Formulas, and Tables for Control Charts*, 1987. Publication number: ANSI/ASQC A1-1987.
- Breyfogle, Forest W., *Statistical Methods for Testing, Development, and Manufacturing*, John Wiley and Sons, 1992.
- Montgomery, Douglas C., *Introduction to Statistical Quality Control*, J. Wiley and Sons, 2nd edition, 1991.
- Wheeler, Donald J. and Chambers, David S., *Understanding Statistical Process Control*, SPC Press, 2nd edition, 1992.

Customer Communication

National Instruments wants to receive your comments on our products and manuals. We are interested in the applications you develop with our products, and we want to help if you have problems with them. To make it easy for you to contact us, this manual contains comment and technical support forms for you to complete. These forms are in the appendix, *Customer Communication*, at the end of this manual.



Introduction to Statistical Process Control in LabVIEW

This chapter contains the installation procedure, gives an overview of Statistical Process Control (SPC), and discusses the LabVIEW SPC Toolkit VIs and examples.

Installation

The following sections contain instructions for installing the SPC Toolkit on Windows, Sun SPARCstation, and Macintosh. The SPC Toolkit comes in compressed form on floppy disks. Installing the SPC Toolkit requires approximately 4 MB.

Windows

You can install the SPC Toolkit from the DOS prompt, the Windows File Manager, or with the **Run...** command from the **File** menu of the Program Manager.

1. Insert the first SPC Toolkit disk into the 3.5-in. disk drive and run the `SETUP . EXE` program using one of the following three methods.
 - a. From Windows, select **Run...** from the **File** menu of the Program Manager. A dialog box appears. Type `X: \SETUP` (where X is the proper drive designation). Press `<enter>` or select **OK**.
 - b. From Windows, launch the File Manager. Click on the drive icon that contains the installation disk. Find `SETUP . EXE` in the list of files on that disk and double-click on it.
2. After you choose an installation option, follow the instructions that appear on the screen. The installer will prompt you to name the directory that contains LabVIEW and its associated files.

After you install the LabVIEW SPC Toolkit, your LabVIEW directory should contain a new SPC directory, and the LabVIEW **Functions** and **Controls** menus will contain SPC entries the next time you launch LabVIEW.

SPARCstation

You can install the SPC Toolkit as shown in the following steps. You do not need root privileges to install the SPC Toolkit, but you must be able to write to the LabVIEW directory where the SPC Toolkit will be installed.

On systems running Solaris 2.2 or later you must determine whether your system is running the volume manager, by entering the following command:

```
ps -a | fgrep vold
```

The following message usually appears to tell you that the volume manager is running:

```
14818 pts/9 S 0:01 /usr/sbin/vold
```

If volume manager is running, install the SPC toolkit as follows:

1. Use the `cd` command to change to a directory where you have write permission, such as `/var/tmp` or your home directory.
2. Insert the first SPC Toolkit disk into the 3.5 in. disk drive.
3. Type `volcheck`.
4. Type `tar xf /vol/dev/aliases/floppy0 INSTALL` to extract the installation script.
5. To run the installation script, type `./INSTALL`. Follow the instructions on the screen. The installer will prompt you to name the directory that contains LabVIEW and its associated files.

If volume manager is not running or if your system runs Solaris 1, install the SPC toolkit as follows:

1. Use the `cd` command to change to a directory where you have write permission, such as `/var/tmp` or your home directory.
2. Insert the first SPC Toolkit disk into the 3.5 in. disk drive.
3. Type `tar xf /dev/rfd0c INSTALL` to extract the installation script.
4. To run the installation script, type `./INSTALL`. Follow the instructions on the screen. The installer will prompt you to name the directory that contains LabVIEW and its associated files.

After you install the LabVIEW SPC Toolkit, your LabVIEW directory should contain a new SPC directory, and the LabVIEW **Functions** and **Controls** menus will contain SPC entries the next time you launch LabVIEW.

Macintosh

1. Insert the first SPC Toolkit disk into the 3.5 in. disk drive and double-click on the LabVIEW SPC Toolkit Installer icon.
2. After you select the **Install** button, you are prompted to select a destination directory. Select your LabVIEW folder.
3. Follow the instructions on the screen.

After you install the LabVIEW SPC Toolkit, your LabVIEW directory should contain a new SPC directory, and the LabVIEW **Functions** and **Controls** menus will contain SPC entries the next time you launch LabVIEW.

Requirements for Using the SPC Toolkit

Some of what you need to build an SPC application is already part of the LabVIEW programming environment. The SPC Toolkit package adds the missing pieces you need to complete your application. The SPC Toolkit consists of a set of VI libraries that implement key SPC functions such as control charts, process statistics, and Pareto analysis. The SPC Toolkit also contains several subVIs that generate the typical SPC graphical presentations.

To use Statistical Process Control effectively, you must be trained in SPC methods. SPC training is necessary because success in an SPC program depends on educated judgment and experience. Rote application of pre-existing templates is no substitute for this judgment. The SPC Toolkit package is a way to use LabVIEW to create SPC applications. If you are using this package to analyze and improve your process, you must receive training in SPC methods or have access to someone who has SPC expertise.

Two good sources on Statistical Process Control methods are the Wheeler and Chambers work and the Montgomery work cited in the *Related Documentation* section of *About This Manual*. The first reference can help you understand how to apply SPC methods, and the

second reference provides a good theoretical and mathematical basis for SPC.

You must have LabVIEW programming experience to use this package. You can explore the simple examples included in the `SPC_EXMP` library after going through Chapter 1 in both the LabVIEW user and tutorial manuals which cover basic LabVIEW principles. To modify the more advanced SPC application examples successfully, however, you must be an advanced LabVIEW user.

In the next section you will take a brief look at the organization of the SPC VIs. Then the following section guides you through some of the LabVIEW programming techniques you will use in statistical processing.

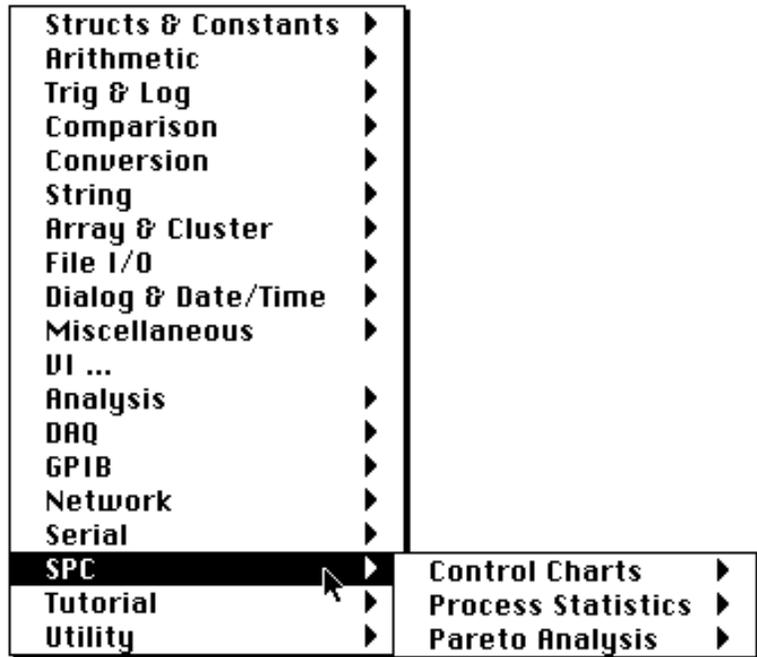
SPC Toolkit Organization

The SPC Toolkit is organized into three sections: VI Libraries, Custom Controls, and Examples.

VI Libraries

After you have read this chapter, you are ready to begin using the SPC Toolkit VIs. Click on the block diagram to activate it and select **SPC** under the **Functions** menu. You see the menu and submenu shown in the following illustration. Then select the VI you want; the icon corresponding to that VI will appear in the block diagram, ready for you to wire it.

► **Note:** *The screens illustrated in this manual were taken on the Macintosh. If you are using Sun or Windows, your screens will look slightly different, but the information on the screens is the same across all three platforms.*



The Control Chart VIs include VIs for calculating control chart limits for both attributes and variables charts, drawing control chart graphs, and applying run rules to control charts.

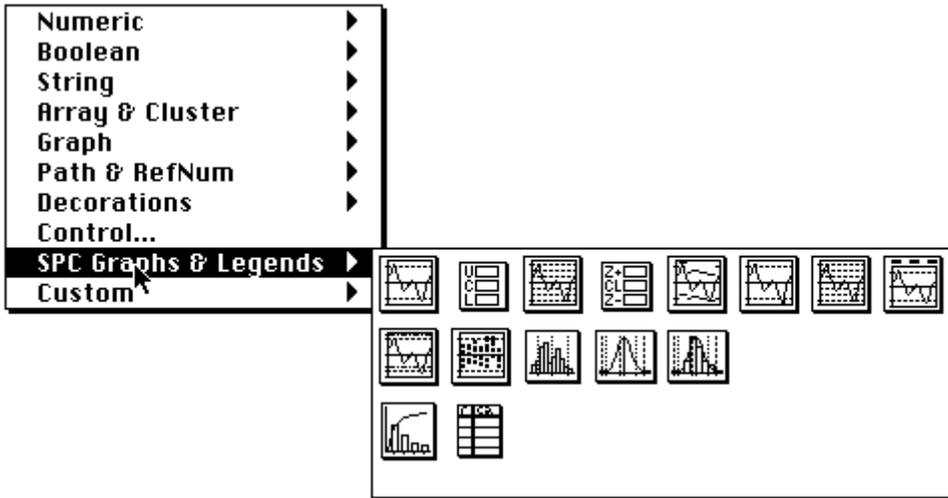
The Process Statistics VIs include VIs for estimating process distribution and capability, calculating and plotting histograms, and functions for plotting and fitting normal probability distribution functions to histograms.

The Pareto Analysis VIs include VIs for counting and sorting assigned causes and for creating Pareto charts.

When you view the VIs from your block diagram using the help window, notice that some of the input parameters are labeled in bold typeface. Bold typeface identifies parameters that should be wired to make the VI operate properly. Plain typeface identifies input parameters that are optional. Optional parameters can help you take advantage of advanced modes of operation, but are not necessary for the VI to work. When you do not wire the optional input parameters they are automatically set to reasonable default values.

Custom Controls

A set of custom controls for SPC graphs and legends are also installed as part of the LabVIEW front panel **Controls** menu. These include XY graphs specially preformatted to match the multiplot XY graphs output by the various SPC subVIs. The following illustration shows the SPC Graphs & Legends palette with a set of custom controls for use with the SPC Toolkit VIs.



The custom controls are installed as part of SPC Toolkit. These custom controls are pre-formatted and labeled X-Y graphs and legends for wiring directly to the outputs of the drawing VIs for control charts, process statistics and Pareto analysis. They are as follows:



Basic Control Chart. A pre-formatted X-Y graph for use with the Draw Control Chart VI.



Control Chart Lines cluster. A cluster displaying values for the control chart lines for use as a legend with all the drawing VIs for control charts.



Control Chart with Zones. A pre-formatted X-Y graph for use with the Draw Control Chart with Var Limits VI.



Control Chart Zones cluster. A cluster displaying values for the control chart zones A, B and C, for use as a legend with the Draw Control Chart with Zones VI.



Control Chart with Var Limits. A pre-formatted X-Y graph for use with the Draw Control Chart with Var Limits VI.



Control Chart (show pts not in control). A pre-formatted X-Y graph useful for highlighting out of control points. See the example VI X-bar & S Chart correct limits in SPC_EXMP . 11b for a demonstration of how to use this type of graph format.



Control Chart with Zones (show pts). A pre-formatted X-Y graph useful for highlighting out of control points on a Control Chart with Zones. See the example VI Zone Rule Test (Nelson) Example in SPC_EXMP . 11b for a demonstration of how to use this type of graph format.



Control Chart and Limits cluster. A cluster containing a pre-formatted X-Y graph and three numeric indicators for the control chart lines. This is a useful organization and grouping for a control chart and limit values.



Run Chart with Limits. A pre-formatted X-Y graph for use with the Draw Run Chart VI.



Tier Chart. A pre-formatted X-Y graph for use with the Draw Tier Chart VI.



Histogram Bar Graph with Limits. A pre-formatted X-Y graph for use with the Draw Vertical Bar Graph with Limits VI.



Normal PDF Graph with Limits. A pre-formatted X-Y graph for use with the Normal PDF Graph with Limits VI.



Histogram and Normal PDF Plot. A pre-formatted X-Y graph for use with histogram and superimposed normal PDF plot with limits. See the example VI Proc Cap Example 2 in SCP_EXMP . 11b for a demonstration of how to use this type of graph format.



Pareto Chart. A pre-formatted X-Y graph for use with either Pareto Chart output of the Pareto Chart VI.



Pareto Chart legend. A pre-formatted table indicator for use with the legend output of the Pareto Chart VI.

LabVIEW SPC Toolkit Examples

There are two libraries of examples with the SPC Toolkit. The SPC_EXMP . 11b library contains basic to intermediate SPC examples.

These examples are useful for getting started and learning how to group the SPC VIs to perform typical SPC calculations and presentations.

The `SPC_DEMO.llb` (SPC demonstration library) contains an example application, the `Real-time SPC Demo`, that analyzes process data acquired point by point. This is a more advanced VI that you could modify once you are more proficient at using the SPC Toolkit.

These libraries of examples are contained in the SPC directory in your LabVIEW folder or directory.

Implementing SPC Applications in LabVIEW

This section discusses the main components that make up an SPC application and guides you through some of the programming techniques you can use in your statistical processing. These programming techniques include representation of process data, viewing raw process data, creating control charts and determining whether your process is in control, detecting out-of-control points, and using process capability and Pareto analysis.

This section also directs you to the relevant standard LabVIEW features or the additional SPC Toolkit features to use when implementing an application. Definitions of the SPC terms used in this overview appear in the *Glossary* at the end of this manual. All examples that appear in this section are located in the `SPC_EXMP.llb` library.

Representation of Process Data in LabVIEW

In SPC applications, some key characteristics of the process are measured or counted, and then tracked. In this manual, measurements of these processes are referred to as *individual observations* or *individuals*. These measurements are often grouped into *samples* or *subgroups*. The number of observations in a sample is referred to as the *sample size* (also known as subgroup size). Deciding which measurements to make, how many and how often to make them, and how they are grouped is beyond the scope of this manual. See “rational subgrouping” in the sources cited in the *Related Documentation* section of *About This Manual* for more information on this topic.

In the SPC VIs for calculations on variable (measured) data, samples consisting of a number of individual observations are handled as 2D arrays. The arrays are set up where each row is a sample, and the columns

contain the observations. To use these VIs, group your measured process data into appropriate 1D array samples (subgroups), and then group the samples together to form a 2D array. All samples in a 2D array must be the same size. The control chart VIs automatically calculate sample size by measuring the width of the 2D array. You can use the LabVIEW Reshape Array function to convert a 1D array to a 2D array.

If you have a sample (subgroup) size of one, you can keep your data in 1D arrays. In this case, you will be limited to using the X & moving Range chart or \bar{x} -bar & moving Range VIs. Attribute data, such as number of defects per unit, are handled as 1D arrays.

There are two ways you can graphically present your measured data in LabVIEW—as you acquire each data point or sample, and after you have acquired a collection of samples. LabVIEW has several standard methods for viewing process data. Three basic graph types—the waveform chart, the waveform graph, and the XY graph—are all useful to you.

You can implement a run chart (a plot of the individual observations plotted in time order) by wiring a 1D array containing your observations to the standard waveform graph. If you want to monitor your incoming data one point at a time, use a waveform chart. If you are plotting all the points at once, you can use a waveform graph.

SPC charts typically plot process data against reference lines, which may be specification limits, control chart limits, or some other useful reference. In LabVIEW, you can use an XY graph to plot a set of points and reference lines by specifying the reference lines as X-Y pairs. The LabVIEW SPC Toolkit automatically generates these types of XY graphs for you.

The SPC Toolkit includes a set of custom SPC controls, including XY graphs that are preformatted for various types of SPC charts and chart legends. These charts are preformatted to work with the SPC VIs that create SPC graphs.

If you are updating a waveform chart one point at a time, you can group each point into a cluster with the reference points, and wire the cluster to your waveform chart.

Viewing Raw Process Data

It is useful to view your raw process data before calculating control limits and plotting control charts. The SPC VIs provide three methods for

statistics library, estimates the process mean and sigma from the process samples.

For viewing the distribution of your data, a histogram is useful. The General Histogram VI computes a histogram, automatically estimating a reasonable number of bins based on Sturges' rule. You can also choose the number of bins, or specify bin sizes. LabVIEW then plots the histogram using the Vertical Bar Graph VI and an XY graph.

You can superimpose the specification limits on the histogram, which the Vertical Bar Graph with Limits VI does for you. Figure 1-3 shows a basic histogram plot of the individual observations in the 2D samples array plotted against both the natural process limits calculated by the Process Mean and Sigma VI, and the specification limits. Figure 1-4 illustrates the block diagram for the Basic Histogram Plot VI example.

Figure 1-3. Basic Histogram Plot

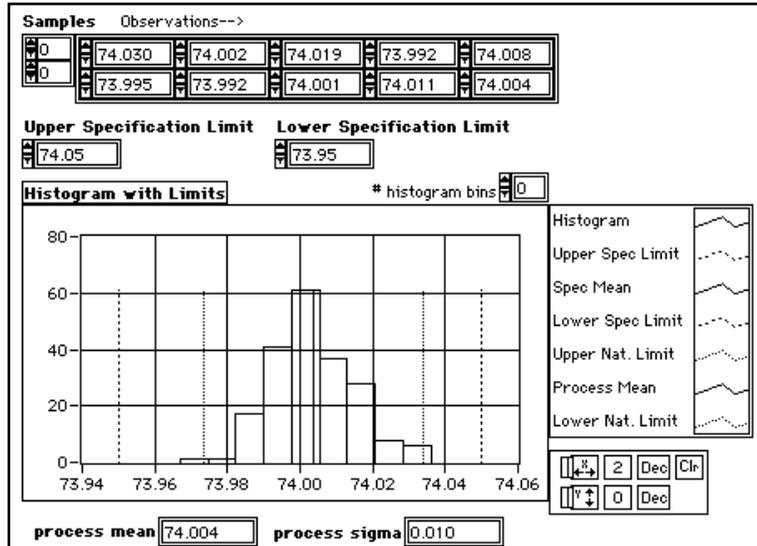
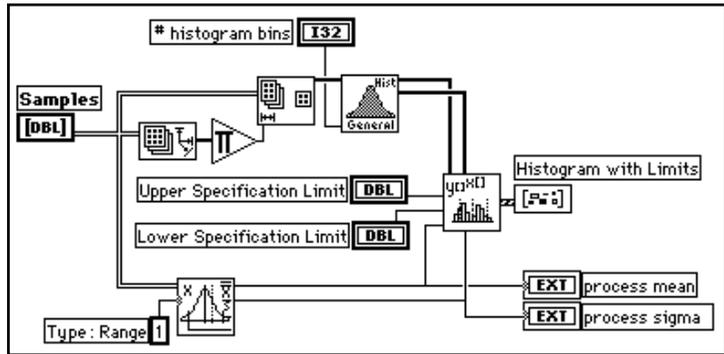


Figure 1-4. Diagram for Basic Histogram Plot



Another useful way to view the raw process data is on the tier chart, also known as a tolerance diagram. This plot charts the observations in each sample in a straight, vertical line. With this vertical line plot, you can visualize the spread and location of the observations in each sample. The Draw Tier Chart VI generates the tier chart for you, as shown in Figure 1-5. Figure 1-6 illustrates the block diagram for the Basic Tier Chart VI.

Figure 1-5. Basic Tier Chart

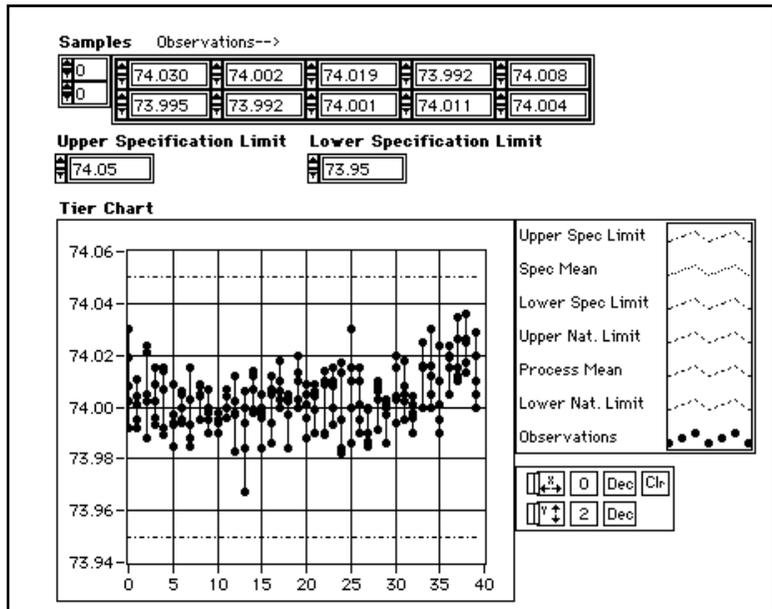
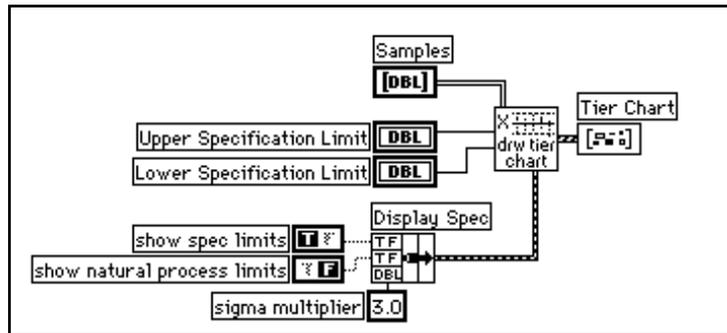


Figure 1-6. Diagram for Basic Tier Chart



Creating Control Charts and Determining Whether the Process is in Control

You use control charts to determine if a process is in control. The LabVIEW SPC Toolkit VIs generate the following standard types of control charts.

- Variables charts:
 - X-bar and standard deviation (X-bar & s Chart VI)
 - X-bar and range (X-bar & R Chart VI)
 - X and moving range (x & mR Chart VI)
 - moving average and moving range (mX-bar & mR Chart VI)
- Attributes Charts:
 - p (p Chart VI)
 - np (np Chart VI)
 - u (u Chart VI)
 - c (c Chart VI)

The control chart VIs calculate the control limits for a control chart. Normally, the control chart VIs use the process data to calculate the control limits. You must choose the set of samples from which to calculate the control limits. Variables charts typically use the first 20 to 30 samples of sample size four or five, for a total of about 100 individual observations of the process. The control chart VIs can also calculate control limits from standard values.

Once the VI calculates the limits, there are several ways to plot the control charts with corresponding VIs that will generate the XY graphs

for the different chart styles. The most common presentation is a control chart that draws the data against the three standard error control limits illustrated in Figure 1-7, in which the Draw Control Chart VI does the graphing. Figure 1-8 shows the block diagram for this VI example.

Figure 1-7. X-bar and S Control Chart Example

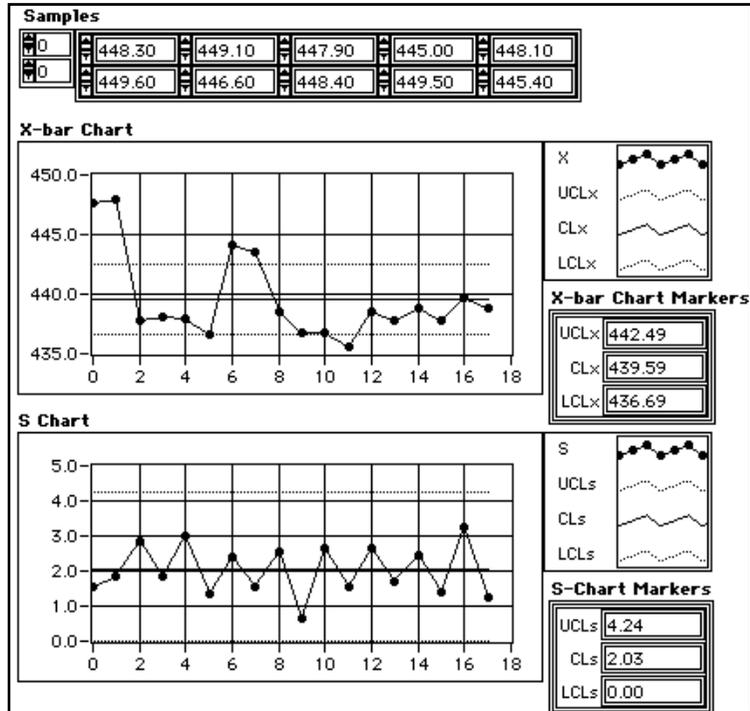
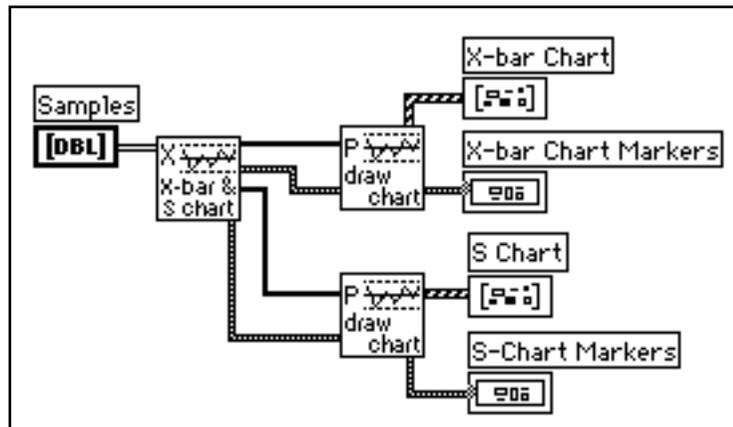


Figure 1-8. Diagram for X-bar and S Control Chart Example



The Draw Chart with Zones VI divides the area between the three sigma control limits into six zones that are one sigma wide, and draws the zones against the control chart points. This presentation is useful when you want to apply rules to the chart to detect out-of-control points. This use of a zones chart is illustrated in the next section, *Detecting Out-of-Control Points* (Figures 1-13 and 1-14).

Some of the attributes charts calculate variable control limits, which are plotted by the Draw Chart with Var Limits VI. The front panel and block diagram of the p Chart with Variable Limits VI Example, which uses the Draw Chart with Var Limits VI, are shown in Figures 1-9 and 1-10.

Figure 1-9. p Chart Example, Var Limits

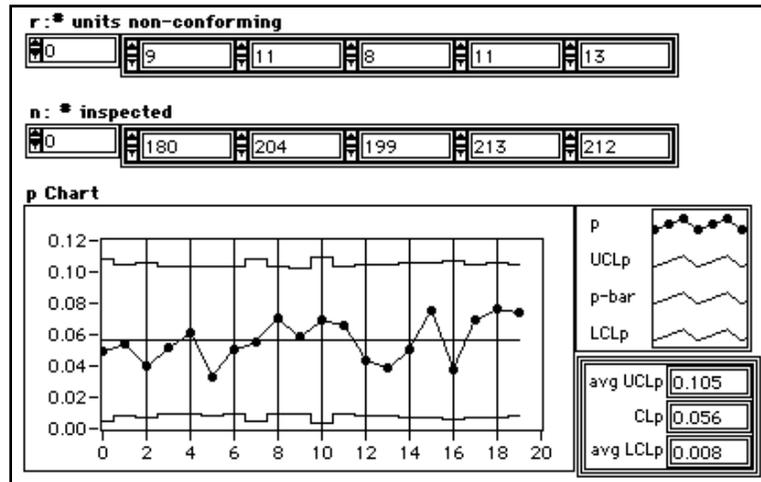
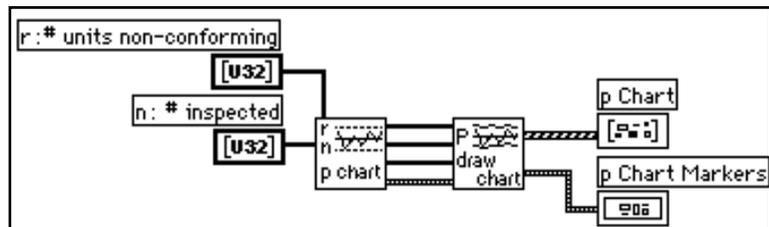


Figure 1-10. Diagram for p Chart Example, Var Limits



Detecting Out-of-Control Points and Process Shift

After a variable or attribute chart VI calculates the control limits, you can determine if the process is in control. The most basic way to determine if a process is in control is to observe which points exceed the upper and

lower control limits. The Check Limits VI identifies the index of each sample that exceeds the process limits.

Figure 1-11 shows the Check Limits VI applied to the X-bar chart in the X-bar & Range Chart Check Limits example; its block diagram is illustrated in Figure 1-12. Notice that, out of the given 40 samples, 25 samples (index zero to 24) are selected for calculating the control limits. The VI calculates the points of the remaining samples for the graph, but does not include them in the control limit calculation.

Figure 1-11. X-bar & Range Chart Check Limits Example

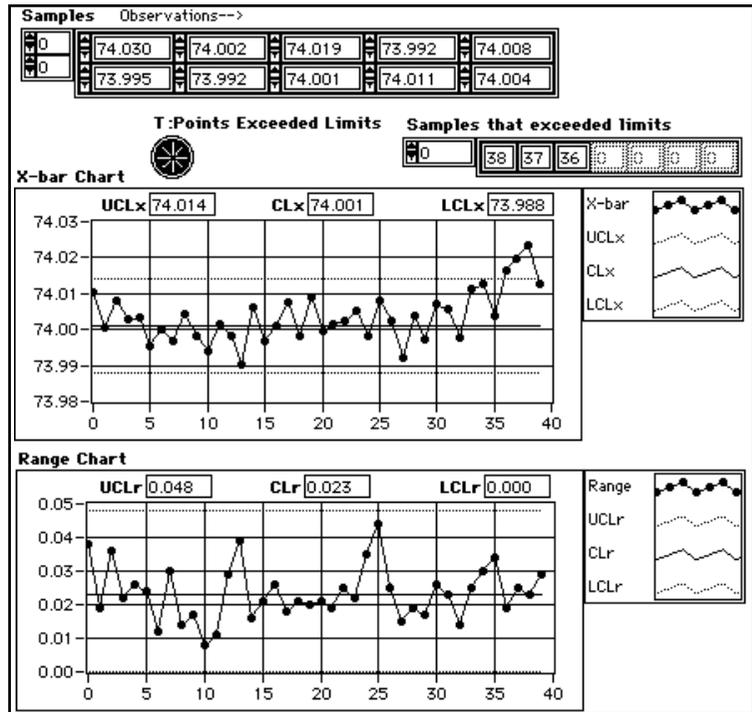
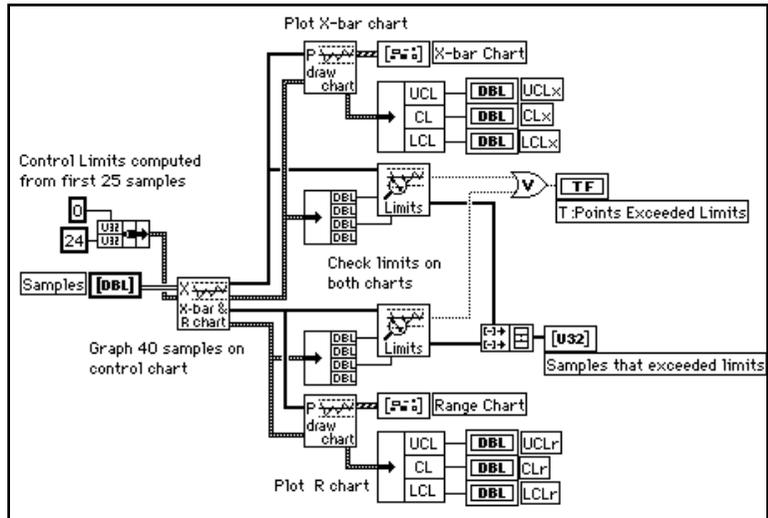


Figure 1-12. Diagram for X-bar & Range Chart Check Limits Example



Control points calculated from a process can stay within the control limits but still exhibit nonrandom behavior such as repeated patterns in the data. To detect such patterns you can use the rule checker VIs to apply run rules to the control chart array. The run rules included in the SPC Toolkit are AT&T/Western Electric and Nelson rules. The rule checker VIs identify the indices of samples that violate the run rules. You can individually enable run rules. Figure 1-13 shows the Zone Rule Test (AT&T/WE) Example, which applies the AT&T/Western Electric rules to an X-bar chart. Figure 1-14 shows the block diagram for this example.

Figure 1-13. Zone Rule Test (AT&T/WE) Example

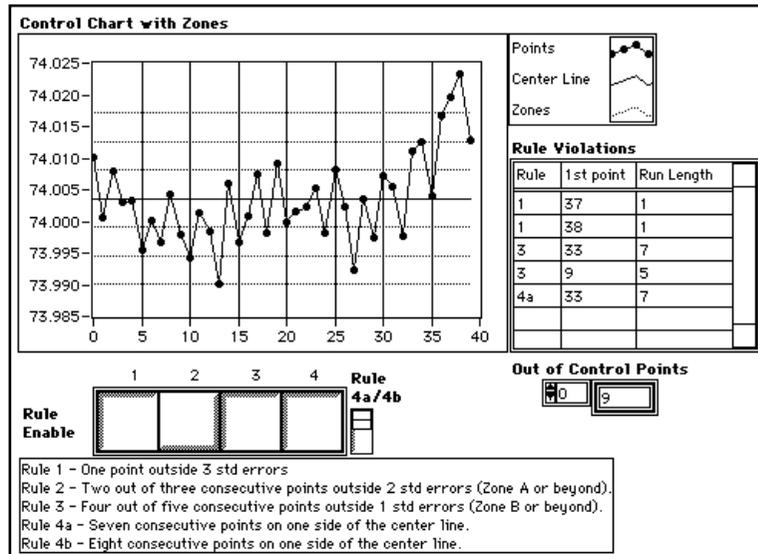
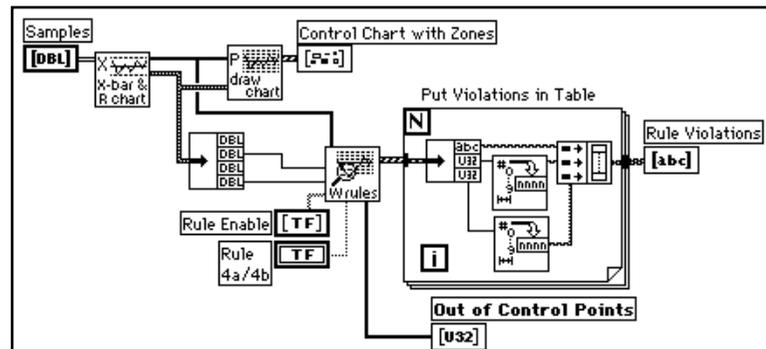


Figure 1-14. Diagram for Zone Rule Test (AT&T/WE) Example



After you have identified samples that have violated run rules, you can recalculate the control limits by calling the Control Chart VI again, and pass in the list of sample indices to ignore.

Note: *Before ignoring a sample in a control limit calculation, you must know what caused the sample to be out of control (that is, you need to know the assignable cause).*

You can also apply run rules to detect process shift, which indicates that control chart limits should be recalculated because the process has changed (shifted with respect to the center line). The Process Shift

Detector VI uses four rules to detect process shift and identifies the first point of the process shift.

Process Capability Analysis

Using process capability analysis, you can quantify the ability of your process to create product within specification. Once your process is in control, you can calculate its capability, which is a predictor of the process performance, as long as the process remains in control. It is misleading to perform these computations unless your process is in control. If it is not in control, process capability analysis is no longer predictive, but can still characterize the past performance of your process.

Two common measures of process capability are the process capability index (PCI or C_p), which measures the process variability with respect to the specification limits, and the centered capability index, or C_{pk} , which measures how centered the process is with respect to the specification limits. The Compute Process Capability VI performs these calculations.

If your process is normally distributed, you can estimate the process fraction non-conforming in parts per million. The Compute Process Capability VI performs this computation, but is invalid unless the process is normally distributed. One method for determining whether your process is normally distributed is to view a histogram of the observations against a normal curve fitted to the histogram.

It is useful to visualize the distribution of the process relative to the specification limits. Figure 1-15 shows a histogram of the process observations against the specification limits and natural process limits. A normal distribution curve is fitted to the histogram. The process capability measures, C_p , C_{pk} , and reject rate, are also calculated and displayed. Figure 1-16 illustrates the block diagram for this example.

Figure 1-15. Process Capability Example 1

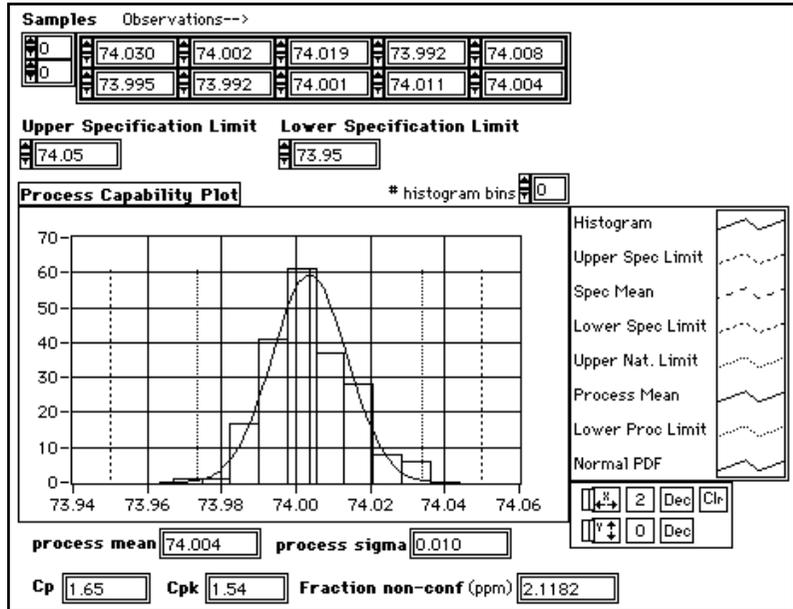
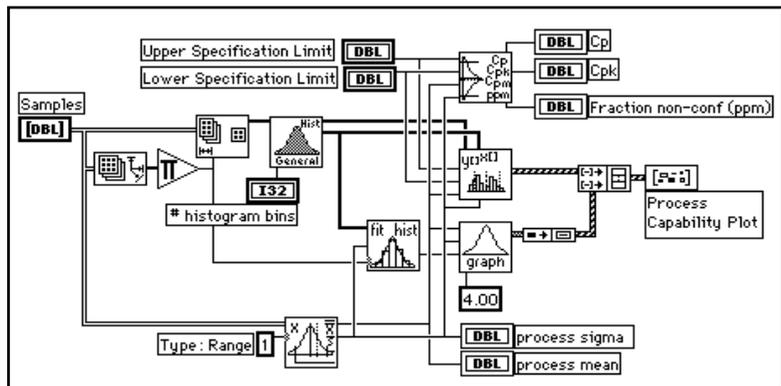


Figure 1-16. Diagram for Process Capability Example 1



Pareto Analysis

In SPC applications, you often need to quantify and prioritize assignable causes that prevent a process from being in control or otherwise prevent a product from conforming to specifications. You can assign causes to a sample when you detect samples being out of control from a control chart. There are other things that can prevent a product from conforming to specifications that need to be analyzed such as tabulated results from product inspection. You can totalize, order, and present causes using the

Pareto VI. Figure 1-17 shows the Pareto analysis and presentation example, Pareto Chart Example. Figure 1-18 illustrates the block diagram for this example.

Figure 1-17. Pareto Chart Example

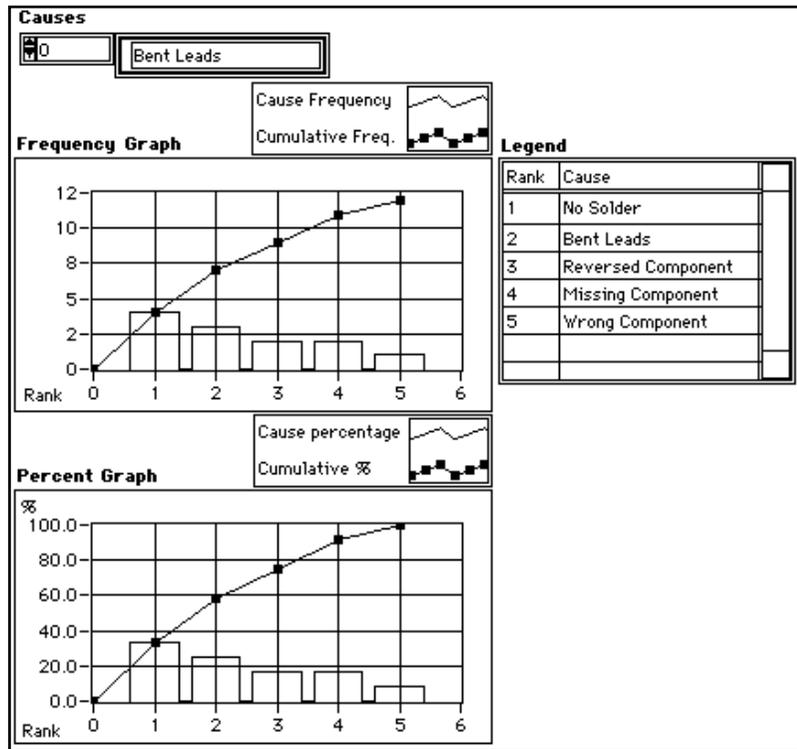
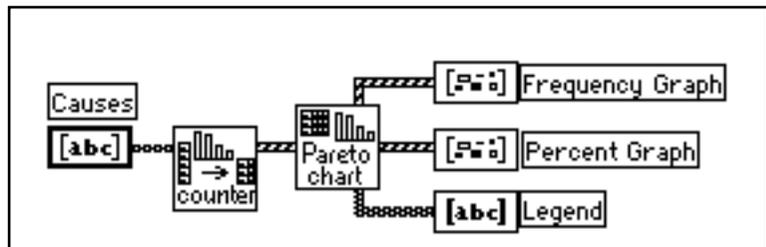
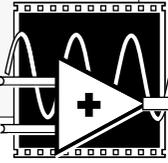


Figure 1-18. Diagram for Pareto Chart Example





Control Chart VIs

This chapter describes the control chart VIs which include the variables charts, attributes charts, chart drawing, and rule checking VIs. The control chart VIs compute control limits for control charts, create control chart graphs, and apply rules to control chart data that detect out-of-control conditions.

Calculating Control Chart Limits and Points

The variables and attributes chart VIs compute the points to be plotted on the control charts, as well as the center line and control limits for the control chart. The process data input to the chart VIs is a one- or two-dimensional array of samples. The control chart VIs pass output arrays and chart limits clusters to one of the chart drawing VIs to create the desired control chart graph.

The chart limits cluster contains the upper control limit (UCL), center line (CL), lower control limit (LCL), and the standard error from which the upper and lower control limits are calculated. The limits are center line ± 3 standard errors by default.

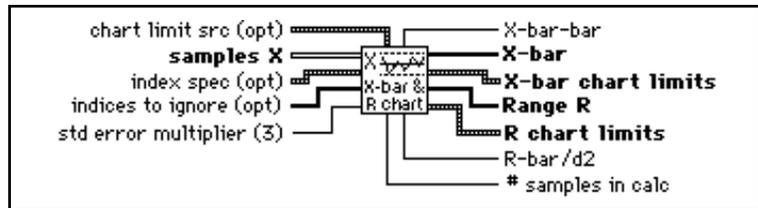
To compute the control limits from the input sample data, you select a subset of the array input to the Control Chart VI by wiring an index specifier. The index specifier designates the start and end index of the samples the control chart limit calculations use. You can also exclude specific samples from the control limit calculation by wiring an array of the sample indices to the **indices to ignore** input of the VI. Doing this is useful when samples are detected to be out of control by one of the rule checking VIs. The **# samples in calc** output returns the actual number of samples the VI used to calculate the control limits. If you do not wire either input, the VI calculates the control limits from the entire input array.

Normally the control limits are calculated from the input sample data, however the control chart VIs will calculate control limits based on standard values if you wire the **chart limit src** input cluster.

The standard error multiplier input specifies the multiplier for the VI to use when calculating the upper and lower control limits, normally three. You do not need to wire this input unless you are using upper and lower control limits that are not at ± 3.0 standard errors.

Variables Chart VIs

You use variables charts to detect out-of-control conditions on measured process values. The VIs for creating variables charts generate outputs for two control charts—sample mean and variation. The chart for sample mean tracks variation in the mean of each sample against control limits. The chart for sample variation tracks the variation in the distribution of each sample against control limits. A typical variables control chart VI, the X-bar & R Chart VI, is shown in the following illustration.



The variables chart VIs, whose names appear in the following list, are described in more detail in this chapter.

- X-bar & s Chart
- X-bar & R Chart
- x & mR Chart
- mX-bar & mR Chart
- Single Point X-bar & R/S
- Single Point x/mX-bar & mR

The X-bar & s Chart and X-bar & R Chart VIs take a two-dimensional input array of samples, where each column contains an individual observation within a sample, and each row is a sample. The sample size is the number of columns in the 2-D array. The X-bar & R Chart VI is limited to sample sizes of 25 or less (25 columns). The X-bar & s Chart VI has no limit on the sample size.

The \bar{x} & mR Chart VI and \bar{mX} -bar & mR Chart VI take a one-dimensional input array of individual observations. The VIs calculate the moving average range from n consecutive observations, where n is sample size input. By default, n is set to two.

The Single Point \bar{X} -bar & R/S VI calculates points for sample mean and variation control charts one sample at a time and uses both the range and sample standard deviation calculations. This VI is useful for calculating individual points for a control chart when generating control charts in real time. It is still necessary to use the \bar{X} -bar & s or \bar{X} -bar & R VI for calculating the control limits.

The Single Point \bar{x} / \bar{mX} -bar & mR VI calculates the individual points for an \bar{X} and moving range or moving average and moving range control chart. This VI is useful for calculating individual points for a control chart when generating control charts in real time. It is still necessary to use the \bar{X} & mR or \bar{mX} -bar & mR VI for calculating the control limits.

The variables control chart VIs are each described in more detail later in this chapter.

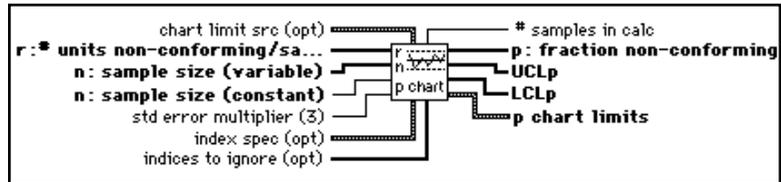
Attributes Chart VIs

You use attributes charts to detect out-of-control conditions on process data that is counted, such as the number of parts defective in a sample of n units inspected. The attributes charts included in the SPC Toolkit are the following.

- p chart
- np chart
- c chart
- u chart

The attributes chart VIs take one or more one-dimensional arrays as the input data. The p chart and u chart can handle both a fixed sample size or variable sample sizes. If the sample sizes are variable, the VI calculates the variable control limits.

A typical attributes chart VI, the p Chart VI, is shown in the following illustration.



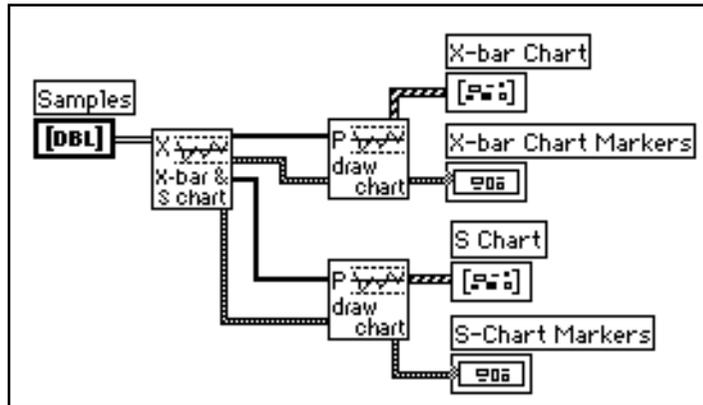
The attributes chart VIs generate outputs for a single control chart. Inputs are one or more 1D arrays that contain values counted from the process. The output includes an array of points for the control chart and the chart limits. In the case of the p chart (shown in the preceding illustration) and the u chart, the sample size inspected may vary for each value of # units non-conforming, or the sample size may be constant. So, you can choose one of the following two inputs: a scalar input for a constant number inspected, n , or an array input for a variable number inspected, n . You should use only one of these two inputs.

The output arrays UCL and LCL are the variable control limits (p and u charts only). The chart limits cluster contains the average upper control limit (UCL), center line (CL), average lower control limit (LCL), and the standard error from which the VI calculates the upper and lower control limits.

The attributes chart VIs are described in more detail later in this chapter.

VIs for Drawing Charts

The control chart VI library contains several VIs for graphing control charts and raw process data. You can also use the built-in LabVIEW waveform chart and XY graphs to present SPC data. The VIs in this library use the XY graph to draw limits against control chart points, a format that is typical of SPC graph presentations. The following illustration shows a typical diagram using the control charts and draw control charts VIs.



Custom controls already preformatted for use with the VIs for drawing charts are available in the SPC Graphs & Legends Control Menu.

VIs for Plotting Control Chart Points and Limits

The control chart VIs calculate control chart limits and points. The VIs in the following list generate a graph of center lines, the upper and lower limit lines, and the computed points from the control chart.

- Draw Control Chart—to use with constant control limits—this is your basic control chart graph
- Draw Chart with Zones—draws zones or warning limits (from constant control limits)—useful for testing run rules
- Draw Control with Var Limits—to use with variable control limits (p and u charts)

These VIs are described in more detail later in this chapter.

VIs for Creating Graphs of Raw Process Data

The Draw Run Chart and Draw Tier Chart VIs create graphs that are independent of the type of control chart you use, and are convenient for viewing the individual observations that make up your samples. This class of graphs will optionally plot your data against specification limits or natural process limits.

Specification limits are user-defined tolerances for the process output. Natural process limits are computed from the samples and represent the process mean and 3 sigma. The natural process limits are not control limits, but are a statistic of the variability in your raw data. The Draw Run Chart and Draw Tier Chart VIs are described as follows.

- Draw Run Chart VI—plots a run chart of the individuals within each sample in order of occurrence. This VI optionally displays specification limits and/or natural process limits (process mean and 3 sigma) against the data.
- Draw Tier Chart VI (variables charts only)—plots all observations (individuals) within each sample. This VI optionally displays specification limits and/or natural process limits (process mean and 3 sigma) against the data.

These VIs have a display mode specifier that you can use to turn on and off drawing of the specification limits or the natural process limits. The display specifier also designates the sigma multiplier for the VI to use for the natural process limits (default 3). You can leave the display mode input unwired, in which case the graphing VI will use the defaults. The defaults are not the same for all the VIs.

The x-axis on all the graphs in the control chart VI library is labeled by sample number (the default starting sample number is zero). You can wire a different number to suit your needs. Notice that array index counting in LabVIEW is zero-based; therefore, numbering samples starting from zero is the least confusing method to use. The control chart VIs use simple (X,Y) pairs to define horizontal limit lines drawn on the XY graph.

An easy way to use these graphing VIs is to copy the graph on the front panel of the VI, and paste it on the front panel of your application. The graph already has appropriate labels, colors, and patterns selected for all the graph lines. You can then size and customize the graph display to fit your needs.

You can also use waveform charts (sometimes called strip charts) to plot your control charts, in which case the VI passes information to the chart one sample at a time. The SPC Toolkit does not provide VIs for strip chart presentation. To draw control chart, natural process, or specification limits against your control chart points, cluster the limit values with your point, and wire the cluster to your waveform chart.

The VIs for drawing charts are described in more detail later in this chapter.

Rule Checker VIs for Testing Out of Limits, Run Rules and Process Shift

The SPC Toolkit contains the following rule checker VIs to test whether points exceed the control limits or whether any of the run rules are violated, and to detect process shift.

- Check Control Limits—identifies samples that exceed the upper and lower control limits.
- Rule Checker (AT&T/WE)—identifies samples that violate one or more of the selected AT&T/Western Electric run rules.
- Rule Checker (Nelson)—identifies samples that violate one or more of the selected Nelson run rules.
- Process Shift Detector—detects process shift with respect to center line.
- Check Sequence—identifies samples violating a generic n out of m sequence.

After a VI has identified out-of-control points, you can wire the rule checker VI output to the indices input of the control chart VIs and exclude these samples from the control limit calculation.

The rule checker VIs are described in more detail later in this chapter.

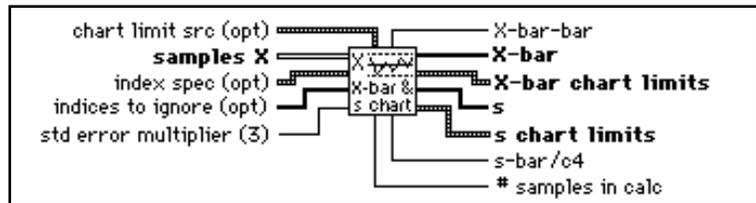
Variables Chart VIs

As described earlier in this chapter, you use the variables chart VIs to detect out-of-control conditions on measured process characteristics. These VIs generate outputs for control charts for sample mean and variation.

X-Bar & s Chart

Computes points and limits for X-bar and s control charts. These are control charts for showing mean and sample standard deviation. Optionally, you can choose a range of indices for samples to use and indices of samples to ignore in the control limit calculation. The VI outputs the actual number of samples used to calculate the control limits. By default, control limits are calculated from the input samples. You can

also calculate control limits from standard values by wiring the **chart limit src** input.



samples X. Samples on which to compute control limits and points for the **X-bar** and s charts. Each row is a sample with n observations, where n is the number of columns in the 2D array. N is also known as the subgroup size. The minimum sample/subgroup size is 2.



index spec. Use this cluster to choose a range of samples to use for the control limits calculation.



start index. Index of the first sample to include in the control limit calculation. Default is zero, which is the index of the first sample in **samples X** array.



end index. Index of the last sample to include in the control limit calculation. Default is zero which selects the last sample in **samples X** array.

If you do not wire this cluster, the VI includes all samples in the **samples X** input array in the control limit calculation.



indices to ignore. Indices of samples to exclude from the control limit calculation. This is useful for eliminating out-of-control points from the control limit calculation. By default, this is an empty array.



std error multiplier. The multiplier to use for the upper and lower control limits. Normally you leave **std error multiplier** unwired, and the VI sets the control chart limits to ± 3.0 standard errors. The default is 3.0.



X-bar-bar. This is an estimate of the process mean based on the grand average of the samples included in the control limit calculation. If the control limits are calculated from standard values, this is set to the standard mean.



X-bar. The mean of each input sample. This is the array plotted on the **X-bar** control chart.



X-bar chart limits. This cluster contains the limits for the **X-bar** chart.



UCL_x. The upper control limit for the **X-bar** chart.



CL_x. The center line for the **X-bar** chart.

CL_x = X-bar-bar or standard mean and is also the estimated process mean, if calculated from the input samples.



LCL_x. The lower control limit for the **X-bar** chart.



standard error. The standard error associated with **CL_x**.



s. The sample standard deviation of each input sample. This is the array plotted on the s control chart.



s chart limits. This cluster contains the limits for the s chart.



UCL_s. The upper control limit for the s chart.



CL_s. The center line for the s chart. CL_s = s-bar if calculated from the input samples, or is standard s₀ or standard sigma *c₄.



LCL_s. The lower control limit for the s chart.



standard error. The standard error associated with CL_s.



s-bar/c₄. This is an estimate of the process sigma (standard deviation) based on the average standard deviation of the samples included in the control limit calculation.



samples in calc. The number of samples the VI used in the control limits calculation. If the control limits are calculated from standard values, this is set to standard s₀/c₄ or sigma.



chart limit src. Specifies whether or not to use standard values for the chart limit calculations. If unwired, this cluster defaults to the common case where the chart limits are calculated from the data in the input sample array. You do not have to wire this cluster unless you want the chart limits to be calculated from standard values. When using standard values, the center line for the X-bar control chart, X-bar-bar, is set to std mean, and the center line for the s control chart, s-bar, is set to std s₀, or std sigma*c₄.



source. Selects one of three sources for chart limits calculation:

0: from data—chart limits are calculated from the data in

the samples X array as qualified by the index spec and indices to ignore inputs (default selection).

- 1: use std mean, s0—chart limits are calculated from standard values for mean and sample standard variation
- 2: use std mean, sigma—chart limits are calculated from standard values for mean and sigma.



std mean. The standard mean value to use when calculating chart limits from standard values.



std s0/sigma. The standard s0 or sigma value to use in calculating chart limits from standard values.

The control limit calculations for the chart limits are as follows.

If standard values are used for the control limit calculations, then

$\bar{\bar{X}}$ (X-bar-bar) is set to std mean

\bar{s} (s-bar) is set to std s0 or std sigma * c4, otherwise, X and s are calculated from the input samples as specified by the index spec and indices to ignore inputs.

$$UCL_X = \bar{\bar{X}} + \text{stderr} \cdot \text{mult} \left(\frac{\bar{s}}{c_4 \sqrt{n}} \right)$$

$$CL_X = \bar{\bar{X}}$$

$$LCL_X = \bar{\bar{X}} - \text{stderr} \cdot \text{mult} \left(\frac{\bar{s}}{c_4 \sqrt{n}} \right)$$

$$\text{standard error} = \frac{\bar{s}}{c_4 \sqrt{n}}$$

$$UCL_s = \bar{s} + \text{stderr} \cdot \text{mult} \left(\frac{\bar{s} \sqrt{1-c_4^2}}{c_4} \right)$$

$$CL_s = \bar{s}$$

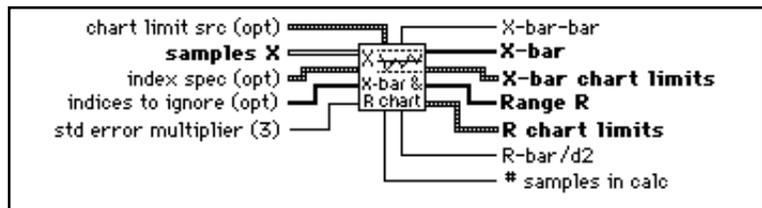
$$LCL_s = \bar{s} - \text{stderr} \cdot \text{mult} \left(\frac{\bar{s} \sqrt{1-c_4^2}}{c_4} \right)$$

$$\text{standard error} = \frac{\bar{s} \sqrt{1 - c_4^2}}{c_4}$$

X-Bar & R Chart

Computes points and limits for X-bar and Range control charts. These are control charts for charting mean and range. Optionally, you can choose a range of indices for samples to use and indices of samples to ignore in the control limit calculation. The VI outputs the actual number of samples it uses to calculate the control limits.

Note: *Use the Range Chart for samples or subgroups of 25 or less, because for sample size of ten or more, the sample range loses efficiency of sample variance. For sample sizes of ten or larger, you should use the X-bar & s Chart VI instead of Range Chart.*



samples X. Samples on which to compute control limits and points for the X-bar and Range charts. Each row is a sample with n observations where n is the number of columns in the 2D array. N is also known as the subgroup size. The maximum sample/subgroup size permitted is 25, the minimum is 2.



index spec. Use this cluster to choose a range of samples to use for the control limits calculation.



start index. Index of the first sample to include in the control limit calculation. Default is zero, which is the index of the first sample in **samples X** array.



end index. Index of the last sample to include in the control limit calculation. Default is zero, which selects the last sample in **samples X** array.

If you do not wire this cluster, the VI includes all the samples in the **samples X** input array in the control limit calculation.



indices to ignore. Indices of samples to exclude from the control limit calculation. Excluding samples is useful for eliminating out-of-control points from the control limit calculation. By default, this is an empty array.



std error multiplier. The standard error multiplier to use for the upper and lower control limits. Normally you leave **std error multiplier** unwired, and the VI sets the control chart limits to +/- 3.0 standard errors. The default is 3.0.



X-bar-bar. This is an estimate of the process mean based on the grand average of the samples included in the control limit calculation. If the control limits are calculated from standard values, this is set to the standard mean.



X-bar. The mean of each input sample. This is the array plotted on the **X-bar** control chart.



X-bar chart limits. This cluster contains the limits for the **X-bar** chart.



UCLx. The upper control limit for the **X-bar** chart.



CLx. The center line for the **X-bar** chart.

$CLx = \mathbf{X-bar-bar}$ or standard mean and is also the estimated process mean if calculated from the input samples.



LCLx. The lower control limit for the **X-bar** chart.



standard error. The standard error associated with CLx .



range R. The range of each input sample. This is the array plotted on the R control chart.



R chart limits. This cluster contains the limits for the Range chart.



UCLr. The upper control limit for the R chart.



CLr. The center line for the R chart. $CLr = R\text{-bar}$ if calculated from the input samples, otherwise this is standard $R0$ or standard σ^2 .



LCLr. The lower control limit for the R chart.



standard error. The standard error associated with CLr.



R-bar/d2. This is an estimate of the process sigma (standard deviation) based on the average range of the samples included in the control limit calculation. If the control limits are calculated from standard values, this is set to standard R0/d2 or standard sigma.



samples in calc. The number of samples the VI used in the control limits calculation.



chart limit src. Specifies whether or not to use standard values for the chart limit calculations. If unwired this defaults to the common case where the chart limits are calculated from the data in the input sample array. You do not have to wire this cluster unless you want the chart limits to be calculated from standard values. When using standard values the center line for the X-bar control chart, X-bar-bar is set to std mean, and the center line for the Rcontrol chart, R-bar, is set to std R0, or std sigma*d2.



source. Selects one of three sources for chart limits calculation:

0: from data—chart limits are calculated from the data in the samples X array as qualified by the index spec and indices to ignore inputs (default selection).

1: use std mean, R0—chart limits are calculated from standard values for mean and sample standard variation

2: use std mean, sigma—chart limits are calculated from standard values for mean and sigma.



std mean. The standard mean value to use when calculating chart limits from standard values.



std R0/sigma. The standard R0 or sigma value to use in calculating chart limits from standard values.

The control limit calculations for the chart limits are as follows.

If standard values are used for the control limit calculations, then

$\bar{\bar{X}}$ (X-bar-bar) is set to std mean

\bar{R} (R-bar) is set to std R0 or std sigma * d2, otherwise, X and R are calculated from the input samples as specified by the index spec and indices to ignore inputs.

$$UCLx = \bar{\bar{X}} + \text{stderr mult} \left(\frac{\bar{R}}{d_2 \sqrt{n}} \right) = \bar{\bar{X}} + A_2 \bar{R} \text{ for stderr mult} = 3.0$$

$$CLx = \bar{\bar{X}}$$

$$LCLx = \bar{\bar{X}} - \text{stderr mult} \left(\frac{\bar{R}}{d_2 \sqrt{n}} \right) = \bar{\bar{X}} - A_2 \bar{R} \text{ for stderr mult} = 3.0$$

$$\text{standard error} = \frac{\bar{R}}{d_2 \sqrt{n}}$$

$$UCLr = \bar{R} + \text{stderr mult} \frac{d_3 \bar{R}}{d_2} = D_4 \bar{R} \text{ for stderr mult} = 3.0$$

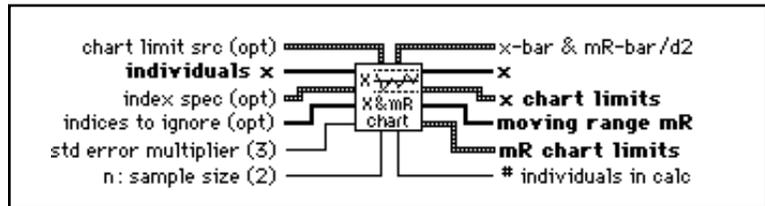
$$CLr = \bar{R}$$

$$LCLr = \bar{R} - \text{stderr mult} \left(\frac{d_3 \bar{R}}{d_2} \right) = D_3 \bar{R} \text{ for stderr mult} = 3.0$$

$$\text{standard error} = \frac{d_3 \bar{R}}{d_2}$$

X& mR Chart

Computes points and limits for individuals x and moving Range charts. These are control charts for charting mean and moving range. This VI computes for a moving range from n consecutive observations. By default, n is two. Optionally, you can choose a range of indices for individuals to use and indices of individuals to ignore in the control calculation. The actual number of individuals the VI uses to calculate the control limits is also output. By default, control limits are calculated from the input samples. You can also calculate control limits from standard values by wiring the **chart limit src** input.



individuals x. Individual observations (or samples of subgroup size 1) on which to compute control limits.



index spec. Use this cluster to choose a range of individuals to use for control limits calculation.



start index. Index of the first individual to include in the control limit calculation. Default is zero or index of the first individual in **individuals x** array.



end index. Index of the last individual to include in the control limit calculation. Default is zero which selects the last sample in **individuals x** array.

If you do not wire this cluster, all values in **individuals x** input array are included in the control limit calculation.



indices to ignore. Indices of individuals to exclude from the control limit calculation. This is useful for eliminating out-of-control points from the control limit calculation. By default, this is an empty array.



std error multiplier. The multiplier to use for the upper and lower control limits. Normally you leave **std error multiplier** unwired, and the VI sets the control chart limits to +/- 3.0 standard errors. The default is 3.0.



x-bar. This is an estimate of the process mean based on the average of the individuals included in the control limit calculation. If the control limits are calculated from standard values, this is set to the standard mean.



x. The individual observations. This is the array plotted on the x control chart.



x chart limits. This cluster contains the limits for the x chart.



UCLx. The upper control limit for the x chart.

-  **CLx.** The center line for the x control chart. $CLx = \bar{x}$ or standard mean and is also the estimated process mean if calculated from the input individuals.
-  **LCLx.** The lower control limit for the x chart.
-  **standard error.** The standard error associated with CLx.
-  **moving range mR.** The moving range of individuals $\{x[i], \dots, x[i-n+1]\}$, where n is the sample size. This is the array plotted on the mR control chart.
-  **mR chart limits.** This cluster contains the limits for the mR chart.
-  **UCLmr.** The upper control limit for the mR chart.
-  **CLmr.** The center line for the mR control chart. $CLmr = \overline{mR}$ if calculated from the input data individuals, or standard $R0$ or standard $\sigma \cdot d2$.
-  **LCLmr.** The lower control limit for the mR chart.
-  **standard error.** The standard error associated with CLmr.
-  **mR-bar/d2.** This is an estimate of the process sigma (standard deviation) based on the average of the moving range between individuals included in the control limit calculation.
-  **# individuals in calc.** The number of individuals the VI uses in the control limits calculation.
-  **chart limit src.** Specifies whether or not to use standard values for the chart limit calculations. If unwired this defaults to the common case where the chart limits are calculated from the data in the input individuals array. You do not have to wire this cluster unless you want the chart limits to be calculated from standard values. When using standard values the center line for the x control chart, \bar{x} is set to std mean, and the center line for the mR control chart, \overline{mR} , is set to std $R0$, or std $\sigma \cdot d2$.
-  **source.** Selects one of three sources for chart limits calculation:
 0: from data—chart limits are calculated from the data in the individuals x array as qualified by the index spec and indices to ignore inputs (default selection).
 1: use std mean, $R0$ —chart limits are calculated from

standard values for mean and range

2: use std mean, sigma—chart limits are calculated from standard values for mean and sigma.



std mean. The standard mean value to use when calculating chart limits from standard values.



std R0/sigma. The standard R0 or sigma value to use in calculating chart limits from standard values.



n: sample size. The number of consecutive individual values to use in the moving range calculation. **n: sample size** may range from 2 to 25. By default, **n: sample size** is 2.



x-bar & mR-bar/d2. This cluster contains estimates on process mean and sigma based on individuals included in the control limit calculation, or on standard values.



x-bar. This is an estimate of the process mean based on the average of the individuals included in the control limit calculation. If the control limits are calculated from standard values, this is set to the standard mean.



mR-bar/d2. This is an estimate of the process sigma based on the average of the moving ranges included in the control limit calculation. If the control limits are calculated from standard values, this is set to the standard R0/d2, or the standard sigma.

The control limit calculations are as follows.

If standard values are used for the control limit calculations, then

\bar{x} (x-bar) is set to std mean

\overline{mR} (mR-bar) is set to std R0 or std sigma * d2,
otherwise, X and mR are calculated from the input samples as specified by the index spec and indices to ignore inputs.

$$UCL_x = \bar{x} + \text{stderr mult} \left(\frac{\overline{mR}}{d_2} \right) = \bar{x} + E_2 \overline{R} \text{ for stderr mult} = 3.0$$

$$CL_x = \bar{x}$$

$$\text{LCL}_x = \bar{x} - \text{stderr mult} \left(\frac{\overline{mR}}{d_2} \right) = \bar{x} - E_2 \overline{R} \text{ for stderr mult} = 3.0$$

$$\text{standard error} = \frac{\overline{mR}}{d_2}$$

$$\text{UCL}_{mr} = \overline{mR} + \text{stderr mult} \left(\frac{d_3 \overline{mR}}{d_2} \right) = D_4 \overline{mR} \text{ for stderr mult} = 3.0$$

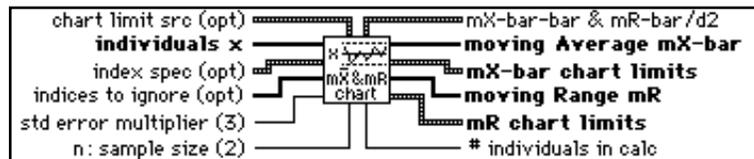
$$\text{CL}_{mr} = \overline{mR}$$

$$\text{LCL}_{mr} = \overline{mR} - \text{stderr mult} \left(\frac{d_3 \overline{mR}}{d_2} \right) = D_3 \overline{mR} \text{ for stderr mult} = 3.0$$

$$\text{standard error} = \left(\frac{d_3 \overline{mR}}{d_2} \right)$$

mX-bar & mR Chart

Computes points and limits for mX-bar and moving Range charts. These are control charts for charting moving average and moving range. This VI computes moving average and range over n observations. By default, n is 2. Optionally, you can choose a range of indices for individuals to use and indices of individuals to ignore in the control calculation. The actual number of individuals the VI uses to calculate the control limits is also output. By default, control limits are calculated from the input samples. You can also calculate control limits from standard values by wiring the **chart limit src** input.



individuals x. Individual observations (or samples of subgroup size 1) on which to compute control limits.



index spec. Use this cluster to choose a range of individuals to use for control limits calculation.



start index. Index of the first individual to include in the control limit calculation. Default is zero or index of the first individual in **individuals x** array.



end index. Index of the last individual to include in the control limit calculation. Default is zero which selects the last sample in **individuals x** array.

If you do not wire this cluster, all values in **individuals x** input array are included in the control limit calculation.



indices to ignore. Indices of individuals to exclude from the control limit calculation. This is useful for eliminating out-of-control points from the control limit calculation. By default, this is an empty array.



std error multiplier. The multiplier to use for the upper and lower control limits. Normally you leave **std error multiplier** unwired, and the VI sets the control chart limits to +/- 3.0 standard errors. The default is 3.0.



x-bar-bar. This is an estimate of the process mean based on the moving average of the individuals included in the control limit calculation. If the control limits are calculated from standard values, this is set to the standard mean.



moving Average mX-bar. The moving average of individuals $\{x[i], \dots, x[i-n+1]\}$, where n is the sample size. This is the array plotted on the x control chart.



mX-bar chart limits. This cluster contains the limits for the moving average chart.



UCLmx. The upper control limit for the moving average chart.



CLmx. The center line for the x control chart. $CLmx = x\text{-bar-bar}$ or standard mean and is also the estimated process mean if calculated from the input individuals.



LCLmx. The lower control limit for the moving average chart.



standard error. The standard error associated with CLmx.



moving range mR. The moving range of individuals $\{x[i], \dots, x[i-n+1]\}$, where n is the sample size. This is the array plotted on the mR control chart.



mR chart limits. This cluster contains the limits for the mR chart.



UCLmr. The upper control limit for the mR chart.



CLmr. The center line for the mR control chart.
CLmr = mR-bar if calculated from the input data individuals, or standard R0 or standard $\sigma \cdot d2$.



LCLmr. The lower control limit for the mR chart.



standard error. The standard error associated with CLmr.



individuals in calc. The number of individuals the VI uses in the control limits calculation.



chart limit src. Specifies whether or not to use standard values for the chart limit calculations. If unwired this defaults to the common case where the chart limits are calculated from the data in the input individual array. You do not have to wire this cluster unless you want the chart limits to be calculated from standard values. When using standard values, the center line for the mX-bar control chart, X-bar-bar, is set to std mean, and the center line for the mR control chart, mR-bar, is set to std R0, or std $\sigma \cdot d2$.



source. Selects one of three sources for chart limits calculation:

0: from data—chart limits are calculated from the data in the individuals x array as qualified by the index spec and indices to ignore inputs (default selection).

1: use std mean, R0—chart limits are calculated from standard values for mean and range

2: use std mean, sigma—chart limits are calculated from standard values for mean and sigma.



std mean. The standard mean value to use when calculating chart limits from standard values.



std R0/sigma. The standard R0 or sigma value to use in calculating chart limits from standard values.



n: sample size. The number of consecutive individual values to use in the moving range calculation. **n: sample size** may range from 2 to 25. By default, **n: sample size** is 2.



mX-bar & mR-bar/d2. This cluster contains estimates on process mean and sigma based on individuals included in the control limit calculation, or on standard values.



mX-bar-bar. This is an estimate of the process mean based on the average of the moving averages included in the control limit calculation. If the control limits are calculated from standard values, this is set to the standard mean.



mR-bar/d2. This is an estimate of the process sigma based on the average of the moving ranges included in the control limit calculation. If the control limit \s are calculated from standard values, this is set to the standard R0/d2, or the standard sigma.

The control limit calculations are as follows.

If standard values are used for the control limit calculations, then

\bar{X} (X-bar) is set to std mean

\overline{mR} (mR-bar) is set to std R0 or std sigma * d2, otherwise, X and mR are calculated from the input samples as specified by the index spec and indices to ignore inputs.

$$UCL_x = \bar{X} + \text{stderr mult} \left(\frac{\bar{R}}{d_2 \sqrt{n}} \right) = \bar{X} + A_2 \bar{R} \text{ for stderr mult} = 3.0$$

$$CL_x = \bar{X}$$

$$LCL_x = \bar{X} - \text{stderr mult} \left(\frac{\bar{R}}{d_2 \sqrt{n}} \right) = \bar{X} - A_2 \bar{R} \text{ for stderr mult} = 3.0$$

$$\text{standard error} = \frac{\bar{R}}{d_2 \sqrt{n}}$$

$$UCL_{mr} = \overline{mR} + \text{stderr mult} \left(\frac{d_3 \overline{mR}}{d_2} \right) = D_4 \overline{mR} \text{ for stderr mult} = 3.0$$

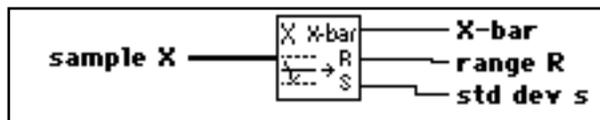
$$CL_{mr} = \overline{mR}$$

$$LCL_{mr} = \overline{mR} - \text{stderr mult} \left(\frac{d_3 \overline{mR}}{d_2} \right) = D_3 \overline{mR} \text{ for stderr mult} = 3.0$$

$$\text{standard error} = \left(\frac{d_3 \overline{mR}}{d_2} \right)$$

Single Point X-Bar & R/S

Computes single points for both X-bar and Range or X-bar and s control charts. Computing single points is useful for plotting control charts one point at a time while you are collecting samples. You must continue to use the X-bar & R or X-bar & s Chart VIs to calculate the control limits for the chart.



sample X. Sample on which to compute the control chart points. **sample X** contains two or more individual observations.



X-bar. The mean of the input sample. Wire this output to the waveform chart to create an **X-bar** control chart.



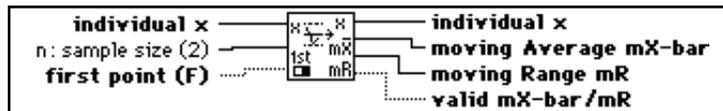
range R. The range of the input sample. Wire this output to the waveform chart to create an **range R** control chart.



std dev S. The sample standard deviation of the input sample. Wire this output to the waveform chart to create an **s** control chart.

Single Point $\bar{x}/m\bar{X}$ -bar & mR

Computes single points for the \bar{x} and moving range or moving average and moving range control charts. Computing single points is useful for plotting control charts one point at a time while you are collecting samples. You must still use the \bar{x} & mR Chart or $m\bar{X}$ -bar & mR Chart VIs to calculate the control limits for the chart. This VI is reentrant so that you can use multiple copies. Each time it is called, this VI saves a copy of the previous $n-1$ individuals to use in calculating the next moving Average and Range.



DBL

individual x. Individual on which to compute the control chart points.

U32

n: sample size. The sample size to use for the moving average and moving range calculations. **n: sample size** ranges from 2 to 25. By default, sample size is 2.

TF

first point(F). Use this input to initialize the VI. **first point** should be TRUE for the first point, FALSE afterwards. The VI will calculate the first point **moving range mR** as zero.

DBL

individual x. Wire this output to the waveform chart to create an X control chart.

DBL

moving Average $m\bar{X}$ -bar. The average of the current **individual x** and the previous **sample size $n-1$** individual x values. Wire this output to the waveform chart to create a moving average control chart.

DBL

moving Range mR . The range of the current **individual x** and the previous **sample size $n-1$** individual x values. Wire this output to the waveform chart to create a moving range control chart.

TF

valid $m\bar{X}$ -bar/ mR . This is set to TRUE when the VI has accumulated sample size n points to calculate the moving Average and Range. For example, for sample size $n = 2$, the VI would have to be called twice in order to have accumulated 2 individual x values from which it calculates the moving average and range. Until sample size n values have been

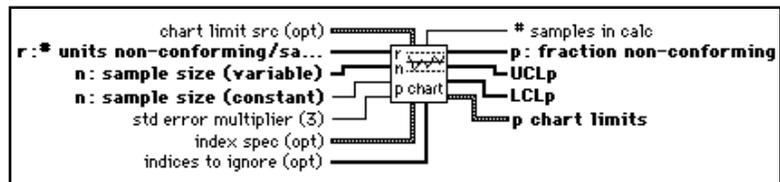
accumulated, the moving average and range are calculated from a smaller sample size, and valid $m\bar{X}$ -bar/mR is set to FALSE.

Attributes Chart VIs

As described earlier in this chapter, you use the attributes chart VIs to detect out-of-control conditions on counted process data.

p chart

Computes points and limits for a p chart, a control chart for fraction non-conforming. Number of units inspected n can be constant for all r or vary for each element of r . Calculates both variable and constant chart limits. Optionally, you can choose a range of indices for samples to use and indices of samples to ignore in the control calculation. The actual number of samples the VI uses to calculate the control limits is also output. By default, control limits are calculated from the input samples. You can also calculate control limits from standard values by wiring the **chart limit src** input.



r: # units non-conforming/sample unit. The number of units non-conforming in each sample inspected.



n: sample size (variable). The number of units inspected per sample for each value of r . Use this array if a variable number of units were inspected for each entry in r . The array length must be the same the length of the array r .



n: sample size (constant). Use this input if the same number of units was inspected for each entry in r . Otherwise, if you are providing an array for n , leave this input unwired.



std error multiplier. The multiplier to use for the upper and lower control limits. Normally you leave **std error multiplier** unwired, and the VI sets the control chart limits to +/- 3.0 standard errors. The default is 3.0.



index spec. Use this cluster to choose a range of samples to use in *r* for the control limits calculation.



start index. Index of the first element of *r* to include in the control limit calculation. Default is zero, which is the index of the first sample in *r* array.



end index. Index of the last element of *r* to include in the control limit calculation. Default is zero which selects the last sample in *r* array.

If you do not wire this cluster, the VI includes all elements in the *r* input array in the control limit calculation.



indices to ignore. Indices of *r* to exclude from the control limit calculation. This input is useful for eliminating out-of-control points from the control limit calculation. By default, this is an empty array.



samples in calc. The number of samples used in the control limits calculation.



p: fraction non-conforming. The number of units non-conforming divided by the number inspected. This is the fraction non-conforming in each input sample. This is the array plotted on the *p* control chart.



UCL_p. The variable limit for the upper control limits. If **std error multiplier** is three, this will be $\bar{p} + 3$ standard errors. The standard error calculation varies with *n*.



LCL_p. The variable limit for the lower control limits. If **std error multiplier** is three, this will be $\bar{p} - 3$ standard errors. The standard error calculation varies with *n*.



p chart limits. This cluster contains the constant limits for the *p* chart.



average UCL_p. The average value of the variable upper control limit for the *p* chart. If the number inspected is constant, use this for the upper control limit.



CL_p. The center line for the *p* chart, \bar{p} or standard p_0 . \bar{p} is the estimated fraction non-conforming for the process if calculated from the input data.



average LCLp. The average value of the variable lower control limit for the p chart. If the number inspected is constant, use this for the lower control limit.



standard error. The standard error associated with CLp.



chart limit src. Specifies whether or not to use standard values for the chart limit calculations. If unwired this defaults to the most common case where the chart limits are calculated from the data in the input array(s). You do not have to wire this cluster unless you want the chart limits to be calculated from standard values. When using standard values the center line for the control chart, p-bar, is set to p0.



source. Selects one of two sources for chart limits calculation:

0: from data—chart limits are calculated from the data in the input arrays as qualified by the index spec and indices to ignore inputs (default selection).

1: use std p0—chart limits are calculated from the standard value p0.



std p0. The standard p0 value to use when calculating chart limits from standard values. The center line for the control chart p-bar will then be set to p0.

The control limit calculations are as follows:

If standard values are used for the control limit calculations, then p-bar is set to std p0, otherwise, p-bar is calculated from the input samples as specified by the index spec and indices to ignore inputs.

$$\bar{p} = \frac{\sum r}{\sum n} \text{ if calculated from input data, otherwise } \bar{p} = p0$$

$$p(i) = \frac{r(i)}{n(i)}$$

$$UCLp = \bar{p} + \text{stderr mult} \sqrt{\frac{\bar{p}(1-\bar{p})}{n}}$$

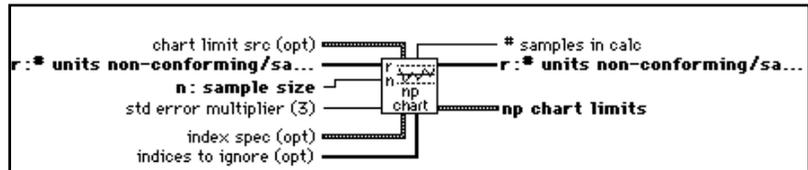
$$CLp = \bar{p}$$

$$LCLp = \bar{p} - \text{stderr mult} \sqrt{\frac{\bar{p}(1 - \bar{p})}{n}}$$

$$\text{standard error} = \sqrt{\frac{\bar{p}(1 - \bar{p})}{n}}$$

np Chart

Computes points and limits for an np chart, a control chart for number non-conforming. Optionally, you can choose a range of indices of samples to use and indices of samples to ignore in the control calculation. The actual number of samples the VI uses to calculate the control limits is also output. By default, control limits are calculated from the input samples. You can also calculate control limits from standard values by wiring the **chart limit src** input.



r: # units non-conforming/sample unit. The number of units non-conforming per sample inspected.



n: sample size. Number of units in each sample.



std error multiplier. The multiplier to use for the upper and lower control limits. Normally you leave **std error multiplier** unwired, and the VI sets the control chart limits to +/- 3.0 standard errors. The default is 3.0.



index spec. Use this cluster to choose a range of samples in r to use for the control limit calculation.



start index. Index of the first element of r to include in the control limit calculation. Default is zero, which is the index of the first sample in samples X array.



end index. Index of the last element of *r* to include in the control limit calculation. Default is zero, which selects the last sample in *r* array.

If you do not wire this cluster, the VI includes all elements in *r* input array in the control limit calculation.



indices to ignore. Indices of *r* to exclude from the control limit calculation. This is useful for eliminating out-of-control points from the control limit calculation. By default, this is an empty array.



samples in calc. The number of samples used in the control limits calculation.



r: # units non-conforming/sample unit. The number of units non-conforming found per sample unit. This is identical to the input array *r*. These are the points plotted on the np control chart.



np chart limits. This cluster contains the limits for the np chart.



UCLnp. The upper control limit for the np chart. If Std error multiplier is three, this will be $\bar{np} + 3$ standard errors.



CLnp. The center line for the np chart, \bar{np} or standard $p_0 * n$. \bar{np} is the estimated number non-conforming for the process if calculated from the input data.



LCLnp. The lower control limit for the np chart. If Std error multiplier is three, this will be $\bar{np} - 3$ standard errors.



standard error. The standard error associated with CLnp.



chart limit src. Specifies whether or not to use standard values for the chart limit calculations. If unwired this defaults to the most common case where the chart limits are calculated from the data in the input array. You do not have to wire this cluster unless you want the chart limits to be calculated from standard values. When using standard values, the center line for the control chart, \bar{np} , is set to $n * p_0$.



source. Selects one of two sources for chart limits calculation:

0: from data—chart limits are calculated from the data in the input arrays as qualified by the index spec and indices to ignore inputs (default selection).

1: use std p0—chart limits are calculated from the standard value p0.



std p0. The standard p0 value to use when calculating chart limits from standard values. The center line for the control chart $n \cdot \bar{p}$ will then be set to $n \cdot p0$.

The control limit calculations are as follows:

If standard values are used for the control limit calculations, then $n \cdot \bar{p}$ is set to $std \ n \cdot p0$, otherwise, $n \cdot \bar{p}$ is calculated from the input samples as specified by the index spec and indices to ignore inputs.

$$\bar{p} = \frac{\sum r}{n(\text{r array length})}, \text{ if calculated from input data,}$$

otherwise $\bar{p} = p0$

$$UCL_{np} = n \times \bar{p} + \text{stderr} \text{ mult} \sqrt{(n\bar{p}(1-\bar{p}))}$$

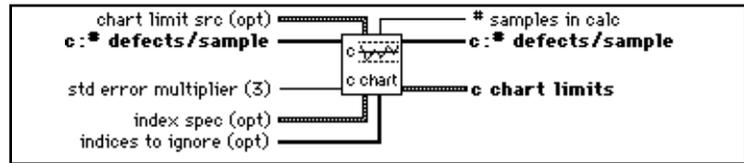
$$CL_{np} = n \times \bar{p}$$

$$LCL_{np} = n \times \bar{p} - \text{stderr} \text{ mult} \sqrt{(n\bar{p}(1-\bar{p}))}$$

$$\text{standard error} = \sqrt{n\bar{p}(1-\bar{p})}$$

c Chart

Computes points and limits for a c chart, a control chart for number of non-conformities or defects. Optionally, you can choose a range of indices for samples to use and indices of samples to ignore in the control calculation. The actual number of samples the VI uses to calculate the control limits is also output. By default, control limits are calculated from the input samples. You can also calculate control limits from standard values by wiring the **chart limit src** input.



c: # defects/ sample. The number of non-conformities or defects per sample inspected.



std error multiplier. The multiplier to use for the upper and lower control limits. Normally you leave **std error multiplier** unwired, and the VI sets the control chart limits to +/- 3.0 standard errors. The default is 3.0.



index spec. Use this cluster to choose a range of samples in **c** to use for the control limit calculation.



start index. Index of the first element of **c** to include in the control limit calculation. The default is zero, which is the index of the first sample in samples X array.



end index. Index of the last element of **c** to include in the control limit calculation. The default is zero, which selects the last sample in **c** array.

If you do not wire this cluster, the VI includes all elements in **c** input array in the control limit calculation.



indices to ignore. Indices of **c** to exclude from the control limit calculation. This is useful for eliminating out-of-control points from the control limit calculation. By default, this is an empty array.



samples in calc. The number of samples used in the control limits calculation.



c: # non-conformities/inspection unit. The number of non-conformities or defects found per sample unit. This is identical to the input array **c**. These are the points plotted on the **c** control chart.



c chart lines. This cluster contains the limits for one **c** chart.



UCLc. The upper control limit for the **c** chart. If **std error multiplier** is three, this will be $\bar{c} + 3$ standard errors.



CLc. The center line for the c chart, c-bar or standard c0. c-bar is the estimated number of non-conformities for the process if calculated from the input data.



LCLc. The lower control limit for the c chart. If **std error multiplier** is three, this will be np-bar - 3 standard errors.



standard error. The standard error associated with CLc.



chart limit src. Specifies whether or not to use standard values for the chart limit calculations. If unwired this defaults to the most common case where the chart limits are calculated from the data in the input array. You do not have to wire this cluster unless you want the chart limits to be calculated from standard values. When using standard values, the center line for the control chart, c-bar, is set to c0.



source. Selects one of two sources for chart limits calculation:

0: from data—chart limits are calculated from the data in the input arrays as qualified by the index spec and indices to ignore inputs (default selection).

1: use std c0—chart limits are calculated from the standard value c0.



std c0. The standard c0 value to use when calculating chart limits from standard values. The center line for the control chart c-bar will then be set to c0.

The control limit calculations are as follows:

If standard values are used for the control limit calculations, then c-bar is set to std c0, otherwise, c-bar is calculated from the input samples as specified by the index spec and indices to ignore inputs.

$$\bar{c} = \frac{\sum c}{c \text{ array length}} \text{ if calculated from the input data,}$$

$$\text{otherwise } \bar{c} = c0$$

$$UCLc = \bar{c} + \text{stderr mult } \sqrt{\bar{c}}$$

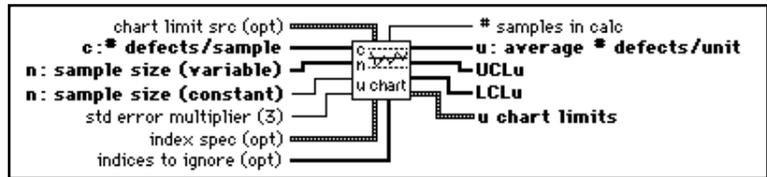
$$CLc = \bar{c}$$

$$LCL_{np} = \bar{c} - \text{stderr mult} \sqrt{\bar{c}}$$

$$\text{standard error} = \sqrt{\bar{c}}$$

u Chart

Computes points and limits for a u chart, a control chart for fraction of non-conformities or defects. Number of units inspected n can be constant for all c or vary for each element in c . Calculates both variable and constant chart limits. Optionally, you can choose a range of indices for samples to use and indices of samples to ignore in the control calculation. The actual number of samples the VI uses to calculate the control limits is also output. By default, control limits are calculated from the input samples. You can also calculate control limits from standard values by wiring the **chart limit src** input.



c: # defects/sample. The number of non-conformities or defects found for each number inspected.



n: sample size (variable). The number of units inspected for each value of c . Use this array if a variable number of units were inspected for each entry in c . The array length must be the same as the length of the array c .



n: sample size (constant). Use this input if the same number of units was inspected for each entry in c . Otherwise, if you are providing an array for n , leave this input unwired.



std error multiplier. The multiplier to use for the upper and lower control limits. Normally you leave **std error multiplier** unwired, and the VI sets the control chart limits to ± 3.0 standard errors. The default is 3.0.



index spec. Use this cluster to choose a range of samples in u to use for the control limit calculation.



start index. Index of the first element of u to include in the control limit calculation. The default is zero, which is the index of the first sample in samples X array.



end index. Index of the last element of u to include in the control limit calculation. The default is zero, which selects the last sample in c array.

If you do not wire this cluster, the VI includes all elements in u input array in the control limit calculation.



indices to ignore. Indices of u to exclude from the control limit calculation. This is useful for eliminating out-of-control points from the control limit calculation. By default, this is an empty array.



u : average # defects/unit. The number of defects divided by the number inspected. This is the average number of non-conformities or defects per unit inspected. This array is plotted on the u control chart.



UCL u . The variable limit for the upper control limits. If **std error multiplier** is three, this output will be $\bar{u} + 3$ standard errors.



LCL u . The variable limit for the lower control limits. If **std error multiplier** is three, this will be $\bar{u} - 3$ standard errors. The standard error calculation varies with n .



u chart limits. This cluster contains the constant limits for the u chart.



average UCL u . The average value of the variable upper control limit for the u chart. If the number inspected is constant, use this for the upper control limit.



CL u . The center line for the u -chart, \bar{u} or standard u_0 . \bar{u} is the estimated fraction of non-conformities or defects for the process if calculated from the input data.



average LCL u . The average value of the variable lower control limit for the chart. If the number inspected is constant, use this for the lower control limit.



standard error. The standard error associated with CL u .



chart limit src. Specifies whether or not to use standard values for the chart limit calculations. If unwired this defaults to the most common case where the chart limits are calculated from the data in the input array(s). You do not have to wire this cluster unless you want the chart limits to

be calculated from standard values. When using standard values, the center line for the control chart, \bar{u} , is set to u_0 .

I16

source. Selects one of two sources for chart limits calculation:

0: from data—chart limits are calculated from the data in the input arrays as qualified by the index spec and indices to ignore inputs (default selection).

1: use $std\ u_0$ —chart limits are calculated from the standard value u_0 .

DBL

std u_0 . The standard u_0 value to use when calculating chart limits from standard values. The center line for the control chart \bar{u} will then be set to u_0 .

$$\bar{u} = \frac{\sum c}{\sum n}, \text{ if calculated from input data,}$$

otherwise $\bar{u} = u_0$

$$\bar{u}(i) = \frac{c(i)}{n(i)}$$

$$UCL_u = \bar{u} + \text{stderr mult} \sqrt{\frac{\bar{u}}{n}}$$

$$CL_u = \bar{u}$$

$$LCL_u = \bar{u} - \text{stderr mult} \sqrt{\frac{\bar{u}}{n}}$$

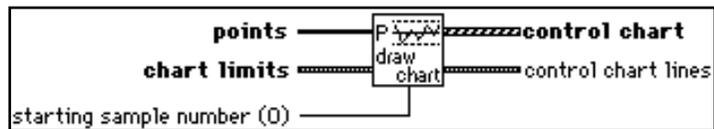
$$\text{standard error} = \sqrt{\frac{\bar{u}}{n}}$$

Draw Control Chart VIs

As described earlier in this chapter, you use the draw control chart VIs for graphing control charts and raw process data.

Draw Control Chart

Given an array of control chart points and the chart limits cluster as created by the chart VIs, creates a control chart (XY graph indicator) with points plotted against upper and lower control limits and center line.



points. The points to be plotted on the control chart, normally the output of one of the control chart VIs, such as an X-bar array.



chart limits. This cluster is output by the control chart VIs.



UCL. The upper control limit for the control chart.



CL. The center line for the control chart.



LCL. The lower control limit for the control chart.



standard error. The standard error associated with CL.



starting sample number. The sample number of the first sample the VI uses to label the chart x-axis. If you do not wire this input, sample labeling on the x-axis will start at zero.



control chart. A control chart XY graph with points and limit lines drawn. You can wire this to the Basic Control Chart custom SPC control.



control chart lines. This cluster is useful as a legend for the limit lines. You can wire this to the Control Chart Lines custom SPC control.



UCL. The upper control limit value for the control chart.



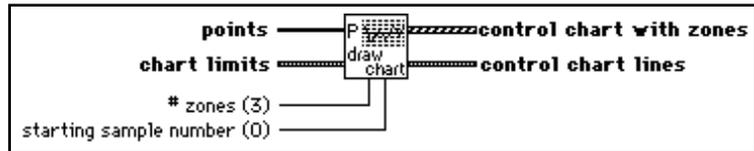
CL. The center line value for the control chart.



LCL. The lower control limit value for the control chart.

Draw Chart with Zones

Given an array of control chart points and the chart limits cluster as created by the chart VIs, creates a control chart graph (XY graph indicator) with points plotted against zones. This VI is useful when visually applying run rules to a control chart.



points. The points to be plotted on the control chart. (Normally the output of one of the control chart VIs, such as an X-bar array).



chart limits. This cluster is output by the control chart VIs.



UCL. The upper control limit for the control chart.



CL. The center line for the control chart.



LCL. The lower control limit for the control chart.



standard error. The standard error associated with CL.



zones. The number of zones to mark on the control chart. If you do not wire this input, three zones—zone A, B, and C—are marked on the chart.



starting sample number. The sample number of the first sample the VI uses to label the chart x-axis. If you do not connect this input, sample labeling on the x-axis will start at zero.



control chart with zones. A control chart XY graph with points and horizontal zone lines drawn. You can wire this to the Control Chart with Zones custom SPC control.



control chart zones. This cluster is useful as a legend for the zone lines. You can wire this to the Control Chart Zones Cluster custom SPC control.



zone A+. The upper 3 standard error control limit value for the control chart.



zone B+. The upper 2 standard error control limit value for the control chart.



zone C+. The upper 1 standard error control limit value for the control chart.



CL. The center line value for the control chart.



zone C-. The lower 1 standard error control limit value for the control chart.



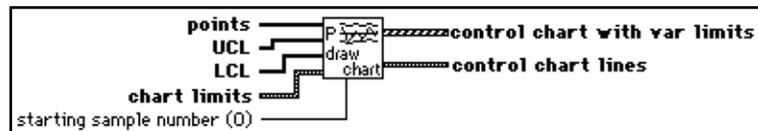
zone B-. The lower 2 standard error control limit value for the control chart.



zone A-. The lower 3 standard error control limit value for the control chart.

Draw Chart with Var Limits

Given an array of control chart points, arrays for the upper and lower control limits, and the control chart limits cluster as created by the chart VIs, creates a variable limits control chart (XY graph indicator). Use this VI with the u chart and p chart if you have a variable number n per unit inspected.



points. The points to be plotted on the control chart, normally the output of one of the control charts, such as an u chart array.



UCL. The variable upper limit to be drawn on the control chart.



LCL. The variable lower limit to be drawn on the control chart.



chart limits. This cluster is output by control chart VIs.



average UCL. The average upper control limit for the control chart.



CL. The center line for the control chart.



average LCL. The average lower control limit for the control chart.



standard error. The standard error associated with CL.



starting sample number. The sample number of the first sample the VI uses to label the chart x-axis. If you do not wire this input, sample labeling on the x-axis will start at one.



control chart. A control chart XY graph with points and variable limit lines drawn. You can wire this to the Control Chart with Var Limits custom SPC control.



control chart lines. This cluster is useful as a legend for the constant limit lines. You can wire this to the Control Chart Lines custom SPC control.



UCL. The average upper control limit value for the control chart.



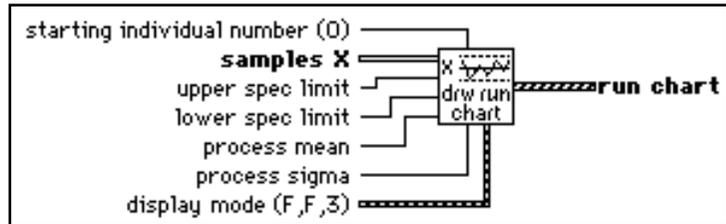
CL. The center line value for the control chart.



LCL. The average lower control limit value for the control chart.

Draw Run Chart

Draws a run chart of the individuals making up the samples X array in time order. The VI interprets the individual observations in the samples array to be in time order by row, and the individuals within each row by column. Optionally, the VI plots the specification limits and/or the natural process limits against the observations.



You can turn on and off drawing of the specification limits or the natural process limits, and choose the sigma multiplier to use for the natural process limits (default three), by wiring in the display mode. If you do not wire the display mode, the VI does not draw the specification limits and natural process limits by default.



samples X. The points to be plotted on the run chart. Each row is a sample with n observations where n is the number of columns in the 2D array. N is also known as the subgroup size. You can generate a run chart from a 1D array by wiring it to a build array primitive before wiring it to the **samples X** input.



starting individual number. The individual number of the first individual the VI uses to label the chart x-axis. If you do not wire this input, sample labeling on the x-axis will start at zero.



upper spec limit. The upper specification limit of the process.



lower spec limit. The lower specification limit of the process.



process mean. The estimated **process mean**. See the description of the Process Mean and Sigma VI in Chapter 3, *Process Statistics VIs*, for an explanation of how the **process mean** and **process sigma** values are calculated.



process sigma. The estimated **process sigma**.



display mode. Optional display mode specifier. If unwired, the VI will not draw the specification and natural process limits by default. This cluster contains the following.



show spec limits. If TRUE, the VI will draw the specification limits on the chart. The default is FALSE.



show natural process limits. If TRUE, the VI will draw the natural process limits on the chart. The default is FALSE.



NP limit sigma. The sigma multiplier for the natural process limits. By default, this is 3.0. The natural process limits drawn on the chart will then be **process mean +/- NP limit sigma*process sigma**.



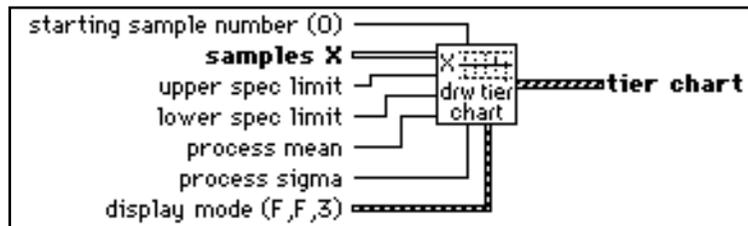
Note: *If you wire the display mode cluster to change one of the default settings, you must specify all three elements.*



run chart. An XY graph with the individual observations in each sample plotted in order of occurrence. Optionally the VI plots the upper and lower specification limits and the mean specification value, and/or the natural process limits and process mean against the sample plots. You can wire this to the Run Chart with Limits custom SPC control.

Draw Tier Chart

Draws a tier chart (also known as a tolerance diagram). Plots the spread of the observations in each sample. Optionally, the specification limits and/or the natural process limits are also plotted against the observations.



You can turn on and off drawing of the specification limits or the natural process limits, and choose the sigma multiplier to use for the natural process limits (default three), by wiring in the **display mode**. If you do not wire the **display mode**, the specification limits and natural process limits are not drawn.



samples X. The values to be plotted on the tolerance chart. Each row is a sample with n observations where n is the number of columns in the 2D array. n is also known as the subgroup size.



starting sample number. The sample number of the first sample the VI uses to label the chart x-axis. If unwired, sample labeling on the x-axis will start at zero.



upper spec limit. The upper specification limit of the process.



lower spec limit. The lower specification limit of the process.



process mean. The estimated **process mean**. See the description of the Process Mean and Sigma VI in Chapter 3, *Process Statistics VIs*, for an explanation of how the **process mean** and **process sigma** values are calculated.



process sigma. The estimated **process sigma**.



display mode. Optional display mode specifier. If unwired, the VI will not draw the specification limits and the natural process limits by default. This cluster contains the following.



show spec limits. If TRUE, the VI will draw the specification limits. The default is FALSE.



show natural process limits. If TRUE, the VI will draw the natural process limits on the plot. The default is FALSE.



NP limit sigma. The sigma multiplier for the natural process limits. By default, this is 3. The natural process limits drawn on the plot will then be **process mean +/- NP limit sigma*process sigma**.



Note: *If you wire the display mode cluster to change one of the default settings, you must select all three elements.*



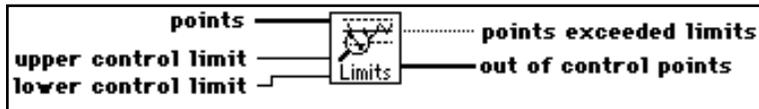
tier chart. A graph in which the VI plots the observations in each sample vertically. Optionally the VI plots the upper and lower specification limits and the mean specification value, and/or the natural process limits and process mean against the sample plots. You can wire this to the Tier Chart custom SPC control.

Rule Checker VIs

As described earlier in this chapter, you use the rule checker VIs to test whether points exceed the control limits or whether any of the run rules are violated.

Check Control Limits

Given a set of control chart points, and the upper and lower control limits from the control chart VIs, this VI checks for points that exceed the control limits. If such points exist, the VI lists the index of each point that exceeds the limits, in the order of how much each point exceeds the control limits.



points. The points plotted on the control chart, normally the output of one of the control chart VIs, such as an X-bar array.



upper control limit. The upper limit line from the control chart VI limits cluster.



lower control limit. The lower limit line from the control chart VI limits cluster.



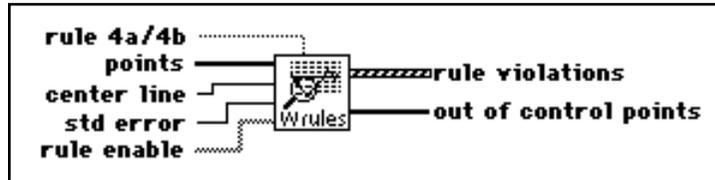
points exceeded limits. TRUE if any of the points exceeded the limits given.



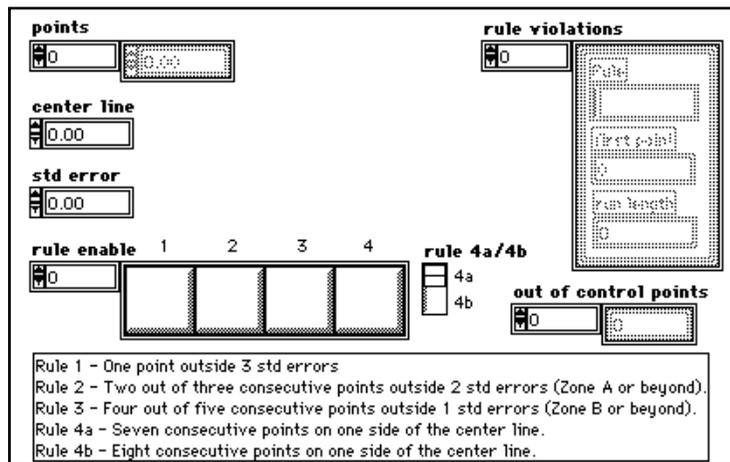
out of control points. Lists the index of each point in the points array that exceeded the limits. The **out of control points** are listed in order of how much each point exceeds the control limits. The most extremely out-of-control point is listed first, followed by the next most out-of-control point, and so forth.

Rule Checker (AT&T/WE)

Applies AT&T/Western Electric run rules to the input points array. Given a set of control chart **points**, and the **center line** and **std error** from the control chart VIs, this VI checks whether points are out of control (or non-randomly distributed) according to the rules enabled.



The front panel in the following illustration lists which rules you can enable. The VI outputs the rules that are violated along with the first point and length of the sequence of points violating the rule, and a list of all points violating any of the enabled rules.



points. The points plotted on the control chart, normally the output of one of the control chart VIs, such as an X-bar array.



center line. The **center line** from the control chart VI limits cluster.



std error. The standard error associated with CL from the control chart VI limits cluster.



rule enable. The run rules to apply to the **points** array. Each element of the array corresponds to a rule 1, 2, 3, 4 from element 0 to 3, in that order

(see front panel for rule explanation). Set the corresponding element to TRUE for each rule you want to apply. (An easy way to do this from the diagram is to use an unsigned 8-bit integer, set the appropriate bits, and then use the number for binary array conversion.)



rule 4a/4b. Selects rule 4a or 4b, when rule 4 is enabled.



rule violations. This cluster lists the rules being violated and which sequence of points violated the rule.



rule. The rule violated (1, 4a, and so forth).



first point. The index of the first point of the sequence of points violating the given rule.



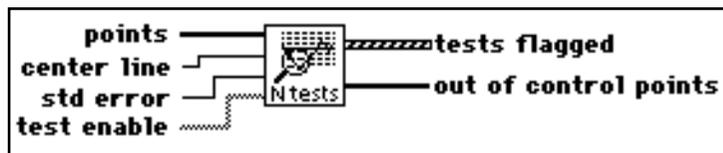
run length. The length of the sequence of points violating the given rule.



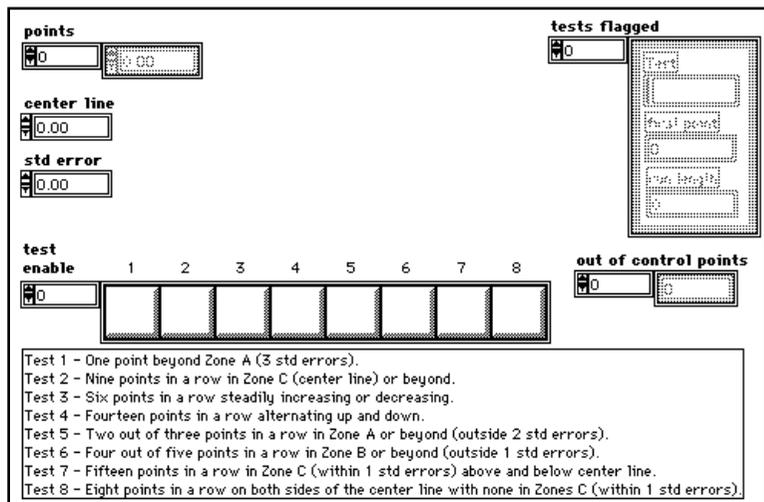
out of control points. Lists the index of each point in the points array that the VI identified as out of control according to the applied rules.

Rule Checker (Nelson)

Applies Nelson tests to the input **points** array. Given a set of control chart points, and the **center line** and **std error** from the control chart VIs, this VI will check whether points are out of control (or non-randomly distributed) according to the enabled Nelson tests.



The front panel in the following illustration lists which tests you can enable. The VI outputs the rules that are violated along with the first point and length of the sequence of points violating the tests, and a list of all points violating any of the enabled tests.



points. The points plotted on the control chart, normally the output of one of the control chart VIs, such as an X-bar array.



center line. The center line from the control chart VI limits cluster.



std error. The standard error associated with CL from the control chart VI limits cluster.



test enable. The run rules to apply to the **points** array. Each element of the array corresponds to a test 1, 2, 3, ... 8 from element 0 to 7, in that order (see front panel for test explanation). Set the corresponding element to TRUE for each test you want to apply. (An easy way to do this from the diagram is to use an unsigned 8-bit integer, set the appropriate bits, and then use the number to binary array conversion.)



tests flagged. This cluster lists the tests violated and which sequence of points violated the test.



test. The test violated (1, 4, and so forth).



first point. The index of the first point of the sequence of points violating the given test.



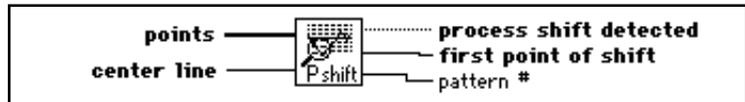
run length. The length of the sequence of points violating the given test.



out of control points. Lists the index of each point in the **points** array that the VI identified as out of control according to the applied tests.

Process Shift Detector

Uses run rules to detect a process shift in a control chart with respect to the center line. Returns whether or not a process shift was detected, and if so, the index of the first point of shift and the pattern detected are output. This VI can be used to determine when to recalculate control limits.



The VI searches for the following patterns that signal a process shift:

Pattern 1: At least 10 out of 11 consecutive data points are on the same side of the center line.

Pattern 2: At least 12 out of 14 consecutive data points are on the same side of the center line.

Pattern 3: At least 14 out of 17 consecutive data points are on the same side of the center line.

Pattern 4: At least 16 out of 20 consecutive data points are on the same side of the center line.



points. The points plotted on the control chart, normally the output of one of the control chart VIs, such as an X-bar array.



center line. The center line from the control chart VI limits cluster.



process shift detected. If true, the VI detected one or more of the four patterns listed above in the control chart points with respect to the center line, indicating a process shift.



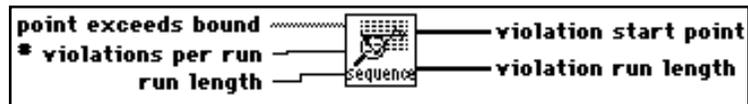
first point of shift. If process shift detected is true, this outputs the index of the first point in the detected process shift. This point can be used as the start index for recalculating control limits.



pattern #. If process shift detected is true, this outputs the number of the first pattern [1 to 4] the VI found.

Sequence Checker

Searches Boolean input array **point exceeds bound** for **# violations per run** set to TRUE within **run length**. Values in the **point exceeds bound** array are set to TRUE if they violate some rule limit. Returns arrays with the index of each violation point and run length associated with that point. You use this VI as a generic building block in the rule checker VIs.



For example, to test the rule four out of five points above two standard errors, set **# violations per run** to four and **run length** to five, test whether each of the control chart points is greater than center line plus two standard errors, and wire resulting boolean array to **point exceeds bound** input.



point exceeds bound. Set each element to TRUE if point exceeds test bound.



violations per run. The number of violations per **run length** that indicate a rule has been violated.



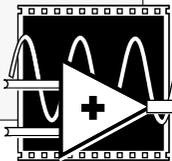
run length. The **run length** for the rule.



violation start point. Start point of each sequence violating the rule.



violation run length. The length of each sequence violating the rule.



Process Statistics VIs

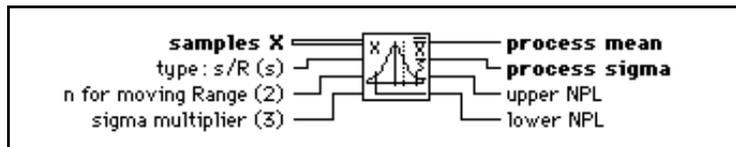
This chapter describes the process statistics VIs, which are useful for process capability analysis and for viewing and measuring process distribution.

The process statistics VIs perform the following operations.

- compute process mean and sigma
- compute process capability ratios and reject rates
- create and graph histograms
- plot normal probability distribution functions against histograms and process specification limits

Process Mean and Sigma

Computes process mean and sigma and upper and lower natural process limits from process samples. You can estimate the process sigma in several ways. If the sample size is greater than one, you can use either sample standard deviation s or range R to estimate the process sigma. You select s or R by the type input. If the sample size is one, the VI automatically uses the moving range to estimate the process sigma.



samples X. Samples on which to compute **process mean** and **process sigma**. Each row is a sample with n observations where n is the number of columns in the 2D array. To calculate process mean and sigma on a 1D array, use the array builder primitive to create a 2D array and then wire to the **samples X** input.



type: s/R. Type of **process sigma** computation to perform.

s: base **process sigma** estimate on mean sample standard deviation (default)

R: base **process sigma** estimate on mean sample range



Note: *If sample size is 1, process sigma estimate is based on mean moving range.*



sigma multiplier. The sigma multiplier for calculating **upper NPL** and **lower NPL**. By default, sigma is multiplied by 3.0.



process mean. Mean of the process, estimated from X-bar-bar (or x-bar if sample size is one).



process sigma. The sigma of the process, estimated from

$$\frac{\bar{s}}{c_4} \text{ if type is S,}$$

$$\frac{\bar{R}}{d_2} \text{ if type is r, or}$$

$$\frac{\overline{mR}}{d_2} \text{ if sample size is 1.}$$



upper NPL. The upper natural process limit of the process computed as **process mean + sigma multiplier * process sigma**. By default, this is **process mean + 3.0 * process sigma**.



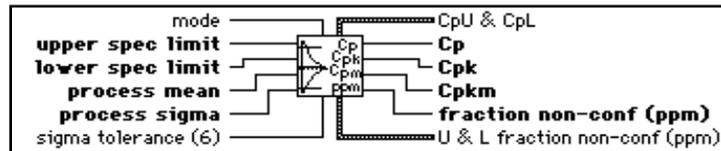
lower NPL. The lower natural process limit of the process computed as **process mean - sigma multiplier * process sigma**. By default, this is **process mean - 3.0 * process sigma**.



n for moving Range. This input is ignored unless the sample size (number of columns) in the input array samples X is one, in which case the moving range is used as a basis for estimating process mean and sigma. **n for moving Range** ranges from 2 to 25. By default, **n for moving Range** is set to 2.

Compute Process Capability

Given the specification limits and the **process mean** and **process sigma**, computes the process capability ratios **Cp**, **Cpk**, and **Cpkm** as well as the estimated process fraction non-conforming in parts per million (ppm). Notice that the fraction non-conforming is valid only if the process is normally distributed. Computes one-sided upper and one-sided lower in addition to two-sided process capability ratios and fraction non-conforming.



[DBL]

upper spec limit. The upper specification limit of the process.

[DBL]

lower spec limit. The lower specification limit of the process.

[DBL]

process mean. The estimated **process mean**. See the description of the Process Mean and Sigma VI in this chapter, for an explanation of how the values for **process mean** and **process sigma** are calculated.

[DBL]

process sigma. The estimated **process sigma**.

[U16]

mode. Selects whether to calculate two-sided, one-sided upper only or one-sided lower only process capability and fraction non-conforming. By default **mode** is two-sided.

0: two-sided—**Cpk** and **fraction non-conf** are calculated with respect to both the upper and lower specification limits.

1: upper only—**Cpk** and **fraction non-conf** are calculated with respect to the upper specification limit only. In this case, **Cpk** = **CpU** and **fraction non-conf** = **fraction** > **USL**.

2: lower only—**Cpk** and **fraction non-conf** are calculated with respect to the lower specification limit only. In this case, **Cpk** = **CpL** and **fraction non-conf** = **fraction** < **LSL**.

The values **Cp** and **Cpkm** are always calculated from both the upper and lower specification limits, regardless of the value of **mode**.

[DBL]

sigma tolerance. The sigma multiplier to use in the process capability calculations. If **sigma tolerance** is left unwired, a value of 6.0 is used by

default, which is the tolerance spread most commonly used for process capability calculations.



CpU & CpL. This cluster contains the one-sided upper and lower process capability ratios.



CpU. The one-sided upper process-capability ratio.

$$CpU = \frac{\text{upper spec limit} - \text{process mean}}{0.5 \text{ sigma tolerance} * \text{process sigma}}$$

By default, 0.5 **sigma tolerance**=3.0.



CpL. The one-sided lower process-capability ratio.

$$CpL = \frac{\text{process mean} - \text{lower spec limit}}{0.5 \text{ sigma tolerance} * \text{process sigma}}$$

By default, 0.5 **sigma tolerance**=3.0.



Cp. The process capability index (also known as PCR or PCI).

$$Cp = \frac{\text{upper spec limit} - \text{lower spec limit}}{\text{sigma tolerance} * \text{process sigma}}$$

Cp is always calculated with respect to both the upper and lower specification limits. By default, **sigma tolerance**=6.0.



Cpk. The centered capability ratio also known as the Cpk index.

$$Cpk = \text{MIN}(CpU, CpL).$$



Cpkm. A centered process capability ratio.

$$CpU = \frac{\text{upper spec limit} - \text{lower spec limit}}{\text{sigma tolerance} * \tau}$$

where tau=

$$\sqrt{\text{process sigma}^2 + \left(\text{process mean} - \left(\frac{\text{upper spec limit} - \text{lower spec limit}}{2} \right) \right)^2}$$



fraction non-conf. The estimated fraction non-conforming in parts per million based on a normal process distribution.



U & L fraction non-conf. This cluster contains the estimated fraction non-conforming in parts per million above and below the specification limits.



fraction>USL. The estimated fraction non-conforming in parts per million above the **upper spec limit** based on a normal process distribution.



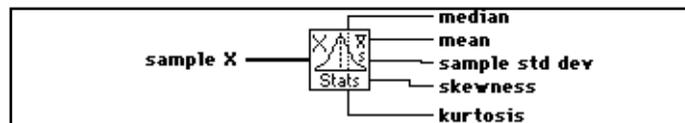
fraction<LSL. The estimated fraction non-conforming in parts per million below the **lower spec limit** based on a normal process distribution.



Note: *If your process distribution is not normal, the fraction non-conforming estimates are invalid.*

Sample Statistics VI

Computes statistics on the input array **sample X**.



sample X. Individual observations on which to compute sample statistics.



median. The center value of the points in sample X when sorted in ascending order. If the number of points in sample X is even, the median is the average of the center pair of points.



mean. The average of the points in sample X.



sample std dev. sample standard deviation computed as:

$$\sqrt{\frac{\text{sum}(x[i] - \bar{x})^2}{n - 1}}$$

where n is the number of points in X.



skewness. degree of asymmetry in the distribution of the points in sample X around the mean. A normal distribution has a skewness of 0. Skewness is computed as

$$\frac{\text{sum}(x[i]-\bar{x})^3}{n \cdot \text{sample std dev}^3}$$

where n is the number of points in X.



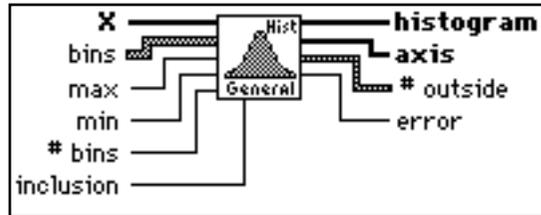
kurtosis. The relative peakedness or flatness of the distribution of the points in sample X. A normal distribution has a kurtosis of 3. Kurtosis is computed as

$$\frac{\text{sum}(x[i]-\bar{x})^4}{n \cdot \text{sample std dev}^4}$$

where n is the number of points in X.

General Histogram

Finds the discrete histogram of the input sequence X based on the given bin specifications.



X. The input data.



bins specifies the boundaries of each bin of the histogram.

The input Bins is an array of clusters where each cluster defines the range of values for a bin. The cluster includes the following elements.



lower specifies the lower boundaries of the bin.



upper specifies the upper boundaries of the bin.



inclusion specifies how the boundaries of each bin are treated. The acceptable values for inclusion are listed as follows.

- 0: lower
- 1: upper
- 2: both

Choosing 0 causes the lower boundary to be part of the bin but not the upper boundary. Choosing 1 is exactly opposite. Both boundaries can be included by choosing 2.

If no bin specifications are provided in the input **bins**, the inputs **max**, **min**, **# bins**, and **inclusion** will be used to specify a set of uniformly spaced bins.



max specifies the maximum value to include in the histogram. This parameter is optional as explained below.



min specifies the minimum value to include in the histogram. This parameter is optional as explained below.

If you do not wire the inputs **max** and **min**, the VI will use the maximum and minimum values in the input sequence **Xd**.



bins specifies the number of bins in the histogram. This parameter is optional as explained below.

If **# bins** is left unwired, the number of bins will be determined according to Sturges' Rule (number of bins = $1 + 3.3\log(\text{sizeof}(X))$).



inclusion specifies how the boundaries of each bin are handled. The valid values for inclusion are:

- 0: include the lower boundary
- 1: include the upper boundary
- 2: include both boundaries



histogram specifies the resulting histogram.



axis specifies the center values for each bin of Histogram.

The centers of each bin are set according to the following equation and returned in the output array **axis**.

$$\text{center}[i] = \frac{\text{lower} + \text{upper}}{2}$$



outside is the output cluster. # **outside** contains three elements:



total. Upon successful execution, the element **total** will contain the total number of points in **X** not falling in any bin.

The elements **above** and **below** have meaning only if Bins are specified such that $\text{Bin}[0].\text{upper} \leq \text{Bin}[1].\text{lower} < \text{Bin}[1].\text{upper}$, and so on.



above represents the number of values in **X** above $\text{Bin}[\text{sizeof}(\text{Bins})-1].\text{upper}$.



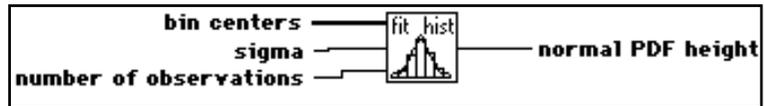
below represents the number of values in **X** below $\text{Bin}[0].\text{lower}$.



error. No errors are returned. This output maintains compatibility with the Advanced Analysis Library General Histogram VI.

Fit Nrml PDF to Histogram

Given the bin centers from a histogram (**axis** values output from the General HistogramVI), the estimated sigma of the observations for the histogram, and the total number of observations in the histogram, calculates the height for a normal probability distribution function (PDF) that fits the histogram.



bin centers. The x-axis values for the center of each histogram bin. Connect the x values output of the histogram VI to this input.



sigma. The estimated sigma of the observations from which the histogram was created. See the description of the Process Mean and Sigma VI or the Sample Statistics VI in this chapter for an explanation of how this value is calculated.



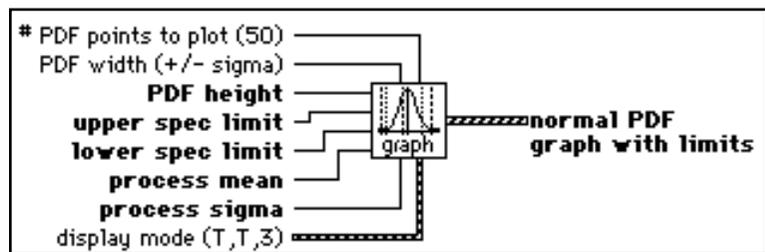
number of observations. The number of observations from which the histogram was created.



normal PDF height. The height of the normal PDF that will fit the histogram. Use this as an input to the Plot Normal PDF VI or Normal PDF Graph with Limits VI.

Normal PDF Graph with Limits

Given the specification limits and the **process mean** and **process sigma**, creates a graph of a normal probability distribution function (PDF) of the process against the specification limits and process mean and sigma.



You can turn on and off drawing of the specification limits or the natural process limits, and specify the sigma multiplier to use for the natural process limits (default 3), by wiring in the display mode. If you do not wire the display mode, the VI automatically draws the specification limits and natural process limits.



upper spec limit. The upper specification limit of the process.



lower spec limit. The lower specification limit of the process.



process mean. The estimated **process mean**. See the description of the Process Mean and Sigma VI in this chapter for an explanation of how the **process mean** and **process sigma** is calculated.



process sigma. The estimated **process sigma**.



PDF points to plot. The number of points to create for the **normal PDF** plot. In most cases, you can leave this unwired. By default 50 points are plotted.



PDF height. The height to draw the PDF. If zero, the VI will draw the PDF with a height proportional to $1/(\text{process sigma})$, corresponding to an area of one under the PDF curve. When fitting a normal PDF to a

histogram, use the **PDF height** output from the Fit Normal PDF to Histogram VI.



PDF width. The width to draw the PDF in terms of +/- sigma. If unconnected, the VI draws the PDF from -3 sigma to +3 sigma by default.



display mode. Optional display mode specifier. If you do not wire this cluster, the specification limits and the natural process limits will be drawn by default. This cluster contains the following inputs.



show spec limits. If true, the specification limits will be drawn.



show natural process limits. If true, the natural process limits will be drawn on the plot.



NP limit sigma. The sigma multiplier for the natural process limits. By default, this is three. The natural process limits drawn will then be **process mean +/- NP limit sigma * process sigma.**



Note: *If you wire the display mode cluster to change one of the default settings, you must specify all three elements.*

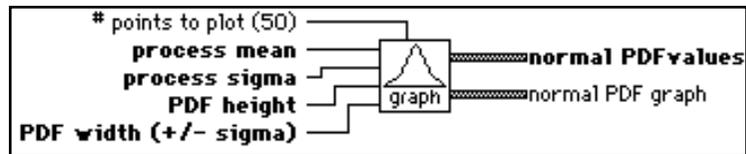


normal PDF graph with limits. An XY graph with the normal PDF of the process plotted against the specification limits and process mean and standard deviation. You can wire this output to the normal PDF graph with Limits custom SPC control.

Plot Normal PDF

Given the process mean and sigma, creates a graph of a normal probability distribution function (PDF) for the process. Also returns an array of XY values for the normal distribution. This graph is useful for

viewing the estimated distribution of a sample or group of samples or for plotting a normal PDF against a histogram.



process mean. The estimated **process mean**. See the Process Mean and Sigma VI or the Sample Mean and Std Dev VI to calculate this value.



process sigma. The estimated **process sigma**. See the Process Mean and Sigma VI or the Sample Statistics VI to calculate this value.



points to plot. The number of points to create for the PDF plot. Normally, you can leave this unwired. By default, 50 points are plotted.



PDF height. The height to draw the PDF. If 0, the VI will draw the PDF with a height proportional to $1/\text{sigma}$, corresponding to an area of 1 under the PDF curve. When fitting a normal PDF to a histogram, use the **PDF height** output from the Fit Normal PDF to Histogram VI.



PDF width. The width to draw the PDF in terms of \pm sigma. By default, the width is 3, and the VI draws the PDF from -3 sigma to +3 sigma.



normal PDF values. XY values making up the PDF plot. An array of clusters made up of the following inputs.



X. The X value of each normal PDF plot point.



Y. The Y axis value of each normal PDF plot point.

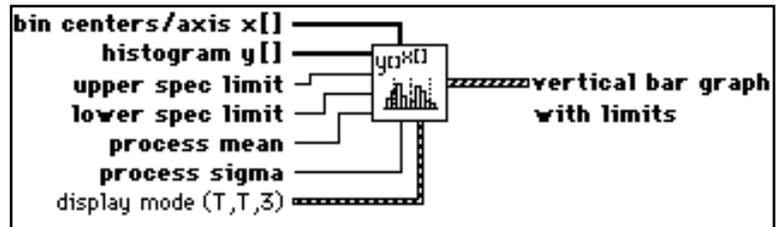
This output can also be treated as an XY graph output; you can wire it directly to an XY graph.



normal PDF graph. An XY graph of a **normal PDF** with the specified height, width, and number of points.

Vertical Bar Graph with Limits

Creates a vertical bar graph centered at each of the bin centers $x[]$ and plots the specification limits and natural process limits as vertical lines against the bar graph. This type of bar graph is useful for viewing the results of a histogram against relevant limits.



You can turn on and off drawing of the specification limits or the natural process limits, and specify the sigma multiplier to use for the natural process limits (default 3), by wiring in the display mode. If you do not wire the display mode, the VI will draw the specification limits and natural process limits (at ± 3 sigma) by default.



histogram $y[]$. Each element of this array is arranged so that the number of observations in each bin of the histogram is from left to right. This is an output from the Histogram VI or the General Histogram VI.



bin centers/axis $x[]$. The x-axis values for the center of each histogram bin. Connect the x values output of the Histogram or General Histogram VI to this input.



upper spec limit. The upper specification limit of the process.



lower spec limit. The lower specification limit of the process.



process mean. The estimated **process mean**. See the description of the Process Mean and Sigma VI in this chapter for an explanation of how the **process mean** and **process sigma** are calculated.



process sigma. The estimated **process sigma**.



display mode. Optional display mode specifier. If unwired, the VI will draw the specification limits and the natural process limits by default. Cluster of the following inputs.



show spec limits. If set to TRUE, the VI draws the specification limits. By default, this is TRUE.



show natural process limits. If set to TRUE, the VI draws natural process limits on the plot. By default, this is FALSE.



NP limit sigma. The sigma multiplier for the natural process limits. The natural process limits are drawn at **process mean** +/- **NPLimit sigma*****process sigma**. By default, this is 3.0.



Note: *If you wire the display mode cluster to change one of the default settings, you must specify all three elements.*



vertical bar graph with limits. Centered vertical bar graph, in which each bar is the height given in the **histogram** input and is centered at the values given in the **bin centers** input. You can wire this output to the Histogram Bar Graph with Limits custom SPC control.

Vertical Bar Graph

Creates a centered vertical bar graph. Bars are centered at the **x[]** values with height **y[]**. Can be used to plot a histogram.



histogram. Each element of this array is arranged so that the number of observations in each bin of the histogram is from left to right. This is an output from the Histogram VI.



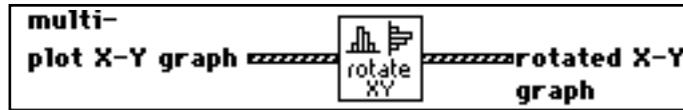
bin centers. The x-axis values for the center of each histogram bin. Connect the x values output of the histogram VI to this input.



bar graph. Centered vertical bar graph. Each bar is of the height given in the **histogram** input and centered at the values given in the **bin centers** input.

Rotate Graph

Given a multiplot XY graph, transposes the X and Y values so that the graph is rotated clockwise 90°.



You can use this general purpose VI to rotate any XY graph. It is particularly useful for plotting horizontal histograms and normal distributions when comparing them against the Y axis of control charts.



multi-plot X-Y graph. An array of clusters of an array of clusters containing:



X. X value of point.



Y. Y value of point.

This array is equivalent to an X-Y multi-plot graph.



rotated X-Y graph. An array of clusters of an array of clusters containing:

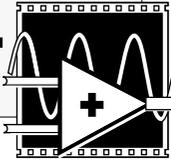


X. X value of point.



Y. Y value of point.

This array is equivalent to a multi-plot graph in which the input X and Y values have been transposed.



Pareto Analysis VIs

This chapter describes the Pareto analysis VIs, which include the Pareto Counter VI, the Pareto Chart VI, and the Cause Code Lookup VI. The Array to Bar/Comb VI, which the Pareto Analysis VIs use as a subVI, is also included.

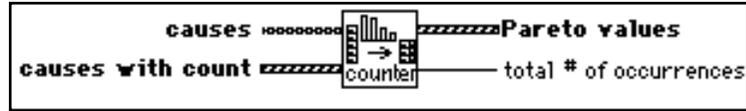
- Pareto Counter—given either an unsorted list of causes or a list of causes and corresponding count, produces a set of Pareto totals for each cause, including frequency, cumulative frequency, percentage of total, and cumulative percentage of total for each cause. The VI sorts the list from the cause with the highest occurrence to the cause with the lowest occurrence.
- Pareto Chart—given a set of Pareto values, creates a Pareto chart of the frequency of occurrence of each cause, and a Pareto chart of the percentage contribution of each cause.
- Cause Code Lookup—given an unsorted list of numeric cause codes, and a list of the cause for each cause code, produces a list of causes with the count for each cause, sorted from the cause with the highest count to the cause with the lowest count.

See Figure 1-17 in Chapter 1 for an example of a Pareto chart application.

The Pareto Counter VI will accept two alternative inputs. You can either pass in an unsorted list of causes (an array of strings) or an array of clusters with each cluster containing a cause string and the corresponding total number of occurrences of that cause. You may be using numeric cause codes instead of strings. In this case, you can use the Cause Code Lookup VI to count the number of occurrences of cause codes and generate a list of cause strings with the count for each cause to be passed to the Pareto Counter VI. If no cause string is given for a cause code, the code itself will be put in string form.

Pareto Counter

Given an unsorted list of causes and the number of occurrences for each cause, sorts the list from the cause with the largest number of occurrences to the smallest and computes Pareto statistics for each cause.



causes with count. An array of clusters for which the number of occurrences for each cause is already counted. If the cause input is empty, the VI calculates the Pareto values from the causes with the count input.



causes with count. The number of occurrences of **cause**.



cause. The name of the assigned cause.

The **causes with count** array has the number of occurrences for each cause counted. If the **causes** input is empty, the Pareto values are calculated from the **causes with count** input.



Pareto Values. An array of clusters consisting of the following outputs.



cause.



frequency. The number of occurrences of **cause**.



cumulative frequency. The total of occurrences at the time this value was read.



percent of total. The percent contribution of **cause**.



cumulative percent. The total of percentages at the time this value was read, sorted by **frequency** of occurrence of reach **cause**.



total # of occurrences. The total number of occurrences of all causes.

Pareto Chart

Given a set of Pareto values (output of the Pareto Counter VI), creates two Pareto charts and the associated legend. One is a bar chart of the frequency of occurrence of each cause. The other is a bar chart of the percentage contribution of each cause. The legend is a list of cause codes with their rank in a table (2D array of strings) format.



Pareto Values. An array of clusters consisting of the following.



cause. The name of the assigned cause.



frequency. The number of occurrences of **cause**.



cumulative frequency. The total number of occurrences at the time this value was read.



percent of total. The percent contribution of **cause**.



cumulative percent. The total of percentages at the time this value was read.

Entries are sorted by **frequency** of occurrence of each **cause**.



Pareto chart (freq). Bar graph of frequency of occurrence of each **cause** with superimposed cumulative plot of frequency. You can wire this output to the Pareto Chart custom SPC control.



Pareto chart (percent). Bar graph of percentage contribution of occurrence of each **cause** with superimposed cumulative plot of frequency. You can wire this output to the Pareto Chart custom SPC control.



Legend. A table (2D array of strings) listing the cause (string) for each bar in the Pareto chart from the rank of 1 (cause with highest frequency of occurrence) on. You can wire this output to the Pareto Chart Legend custom SPC control.

Cause Code Lookup

Given an unsorted list of numeric cause codes (an array of numbers), and a cause code lookup list containing the cause string for each cause code, returns a sorted list of causes with the count of the number of occurrences of each cause.



cause codes. Array of 32-bit integers.



code lookup table. An array of clusters consisting of the following.



cause code. The code for **cause**.



cause name. String associated with the **cause code**.



causes with count. An array of clusters consisting of the following.



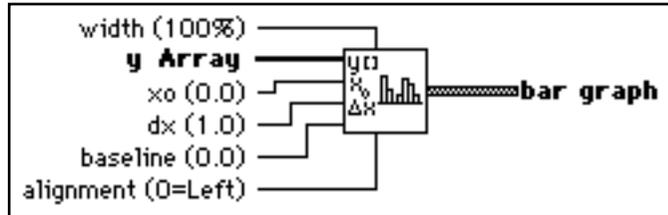
causes on count. The number of occurrences of **cause**.



cause. Name of cause.

Array to Bar/Comb Graph

Makes a vertical bar graph from a single array of numbers. The X-axis values begin at x_0 with increments of dx . The width of the bars can be between 0 (single line “comb plot”) to 100 percent (space between bars). The Pareto Graph VI uses this general purpose bar graph as a subVI.



width. Width of bar in percent. Default is 100 percent.



y Array. An array of heights for each bar.



x_0 . The reference point of first bar. Default is 0.0.



dx. Bar spacing. Default is 1.0.



baseline. Baseline of bar graph. Default is 0.0.



alignment. Alignment of bar from reference point x .

0 = Left

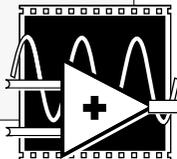
1 = Center

2 = Right

Default is 0 = Left.



bar graph. An XY graph.



Customer Communication

For your convenience, this appendix contains forms to help you gather the information necessary to help us solve technical problems you might have as well as a form you can use to comment on the product documentation. Filling out a copy of the Technical Support Form before contacting National Instruments helps us help you better and faster.

National Instruments provides comprehensive technical assistance around the world. In the U.S. and Canada, applications engineers are available Monday through Friday from 8:00 a.m. to 6:00 p.m. (Central time). In other countries, contact the nearest branch office. You may fax questions to us at any time.

Corporate Headquarters

(512) 795-8248

Technical Support fax: (800) 328-2203
(512) 794-5678

Branch Offices	Phone Number	Fax Number
Australia	(03) 879 9422	(03) 879 9179
Austria	(0662) 435986	(0662) 437010-19
Belgium	02/757.00.20	02/757.03.11
Denmark	45 76 26 00	45 76 71 11
Finland	(90) 527 2321	(90) 502 2930
France	(1) 48 14 24 00	(1) 48 14 24 14
Germany	089/741 31 30	089/714 60 35
Italy	02/48301892	02/48301915
Japan	(03) 3788-1921	(03) 3788-1923
Netherlands	03480-33466	03480-30673
Norway	32-848400	32-848600
Spain	(91) 640 0085	(91) 640 0533
Sweden	08-730 49 70	08-730 43 70
Switzerland	056/20 51 51	056/20 51 55
U.K.	0635 523545	0635 523154

Technical Support Form

Photocopy this form and update it each time you make changes to your software or hardware, and use the completed copy of this form as a reference for your current configuration. Completing this form accurately before contacting National Instruments for technical support helps our applications engineers answer your questions more efficiently.

If you are using any National Instruments hardware or software products related to this problem, include the configuration forms from their user manuals. Include additional pages if necessary.

Name _____

Company _____

Address _____

Fax (____) _____ Phone (____) _____

Computer brand _____ Model _____ Processor _____

Operating system (include version number) _____

Clock speed _____MHz RAM _____MB Display adapter _____

Mouse ___yes ___no Other adapters installed _____

Hard disk capacity _____MB Brand _____

Instruments used _____

Boards installed (include revision and configuration) _____

LabVIEW Version ____ VI Libraries installed (other than standard libraries) _____

The problem is: _____

List any error messages: _____

The following steps reproduce the problem: _____

Documentation Comment Form

National Instruments encourages you to comment on the documentation supplied with our products. This information helps us provide quality products to meet your needs.

Title: **LabVIEW® Statistical Process Control Toolkit Reference Manual**

Edition Date: **September 1994**

Part Number: **320753A-01**

Please comment on the completeness, clarity, and organization of the manual.

If you find errors in the manual, please record the page numbers and describe the errors.

Thank you for your help.

Name _____

Title _____

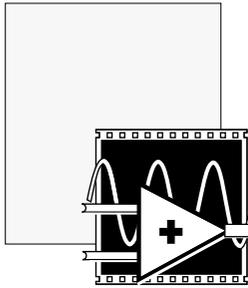
Company _____

Address _____

Phone (____) _____

Mail to: Technical Publications
National Instruments Corporation
6504 Bridge Point Parkway, MS 53-02
Austin, TX 78730-5039

Fax to: Technical Publications
National Instruments Corporation
MS 53-02
(512) 794-5678



Glossary

Prefix	Meaning	Value
μ-	micro-	10 ⁻⁶
m-	milli-	10 ⁻³
k-	kilo-	10 ³

A

- Assignable Cause** A cause that can be detected and identified as contributing to process variation.
- Attribute Data** Data from counting process results such as number of units non-conforming per inspected sample, or number of defects per inspected sample containing n units. As opposed to measured data, attribute data has a discrete number of possible values. p, np, u and c control charts are used to track attribute data.
- Attributes Chart** A control chart that tracks whether a process is in control by tracking attribute data—data counted from the process. p, np, u, c charts are attributes charts.

B

- block diagram** Pictorial description or representation of a program or algorithm. In LabVIEW, the block diagram, which consists of executable icons called nodes and wires that carry data between the nodes, is the source code for the VI. The block diagram resides in the Diagram window of the VI.
- Boolean controls and indicators** Front panel objects used to manipulate and display or input and output Boolean (TRUE or FALSE) data. Several styles are available, such as switches, buttons and LEDs.

C

- c Chart** Control chart that uses number of defects per sample to monitor the stability of the process. Each sample contains n units, and contains c defects per sample (zero or more defects per unit). The value of n must be constant from sample to sample.

Glossary

center line	A line on a control chart representing average long-term value of the statistic plotted on the control chart.
cluster	A set of ordered, unindexed data elements of any data type including numeric, Boolean, string, array, or cluster. The elements must be all controls or all indicators.
connector	Part of the VI or function node that contains its input and output terminals, through which data passes to and from the node.
control	Front panel object for entering data to a VI interactively or to a subVI programmatically.
control chart	A charting method for determining the stability of a process; that is, whether or not a process is in a state of statistical control. Shewart control charts monitor process stability by plotting sample statistics against a center line and control limits.
control limits	Upper and lower limits on a control chart represent the amount of variation about a center line that can be attributed to chance causes for a given process characteristic. Control chart points that fall outside the control limits signal that the process is not in control and that some action should be taken. Control limits may be calculated from process data or from standard values.
Cp process capability index	A measure in sigma units of the process capability that is a ratio of the spread between the specification limits over the m sigma spread of the process variation, where m is normally 6.
Cpk centered capability ratio	A measure in sigma units of the process capability with respect to how well the process is centered relative to the specification limits, also known as distance to nearest specification.
current VI	VI whose Panel window, Diagram window, or icon editor window is the active window.

D

Diagram window	VI window that contains the block diagram code.
Defect	A measured characteristic of a specific unit of product or service that prevents the unit from meeting a specification requirement or is otherwise undesirable. In a unit, one or more defects are possible, and many different types of defects are possible. Also known as a non-conformity, if the characteristic prevents the unit from meeting a specification requirement.

F

fraction non-conforming	the number of units in a sample not conforming to specification divided by the total number of units in the sample.
front panel	The interactive user interface of a VI. Modeled from the front panel of physical instruments, it is composed of switches, slides, meters, graphs, charts, gauges, LEDs, and other controls and indicators.

H

Help window	Special window that displays the names and locations of the terminals for a function or subVI, the description of controls and indicators, the values of universal constants, and descriptions and data types of control attributes.
Histogram	A graphical summary of data in which the individual values are sorted by the range of values into which they fall (also known as bins), and in which the number of individuals falling within a each bin is counted. The data is plotted by showing the number of values (frequency) in each bin.

I

icon	Graphical representation of a node on a block diagram.
in-control process	A stable process whose variation is due only to chance causes; that is, no assignable causes of variation are present.
indicator	Front panel object that displays output.
individual observation	A single measurement of a process characteristic.

M

matrix	Two-dimensional array.
measurement data	Data that is a result of observations or measurements of a characteristic that has a continuous range. As opposed to attribute data, measurement data is not discrete; that is, any discreteness in measurement data is due to the resolution of the measuring device, not the characteristic.
moving Average chart	Control chart that uses the average of n successive individual observations from a process to track the stability of the process mean. This type of control chart is typically used when the sample size is one.
moving Range Chart	Control chart that uses the range between n successive individual observations from a process to track the stability of the process

variation. This type of control chart is typically used when sample size is one.

N

natural process limits	The limits which contain a stated fraction of the individual observations in a population. For a normally distributed population, the stated fraction is typically the process mean ± 3.0 sigma.
non-conforming unit	A unit of product or service that does not meet a specification requirement.
normally distributed	If a process is normally distributed, expected values for individuals within the process population fall on a normal or bell-shaped curve.
np Chart	Control chart that uses the number of non-conforming units in a sample to monitor the stability of the process. The sample contains n units, and zero or more units may be non-conforming. The value n be constant from sample to sample.
numeric controls and indicators	Front panel objects used to manipulate and display or input and output numeric data.

O

observation	A measurement of a process characteristic.
-------------	--

P

p chart	Control chart that uses the fraction of non-conforming units in a sample to monitor the stability of the process. The sample contains n units, and zero or more units may be non-conforming. The value n may vary from sample to sample.
Panel window	VI window that contains the front panel, the execution palette, and the icon/connector pane.
plot	A graphical representation of an array of data shown either on a graph or a chart.
pop up	To call up a special menu by clicking on an object with the right mouse button.
pop-up menus	Menus accessed by command-clicking, usually on an object. Menu options pertain to that object specifically.
Pareto chart	A chart of a set of counted or totaled characteristics in which the characteristics are ranked in order of their frequency of occurrence. In SPC, Pareto charts are normally used to evaluate relative contribution of assignable causes and prioritize corrective action.

process capability	A measure of the ability of a process to produce product within specification, assuming the process is stable. Specifically the variability that can be expected for a given process characteristic with respect to the specification limits for the characteristic.
pull-down menus	Menus accessed from a menu bar. Pull-down menu options are usually general in nature.

R

R (Range) chart	Control chart that uses the sample range R to track the stability of the variation in the process. Sample range is max sample value minus min sample value. Sample size must be two or more.
representation	Subtype of the numeric data type, of which there are signed and unsigned byte, word, and long integers, as well as single-, double-, and extended-precision floating-point numbers, both real and complex.
run chart	A chart of the individual observations in a set of samples plotted in time order of occurrence.
run rules	Rules applied to a consecutive set of points on a control chart, that are used to detect changes in the process such as out-of-control conditions, or process shift.

S

s (std dev) chart	Control chart that uses the sample standard deviation s to determine the stability of variation in the process. Sample size must be two or more.
sample (or subgroup) size	For measurement data, this is the number of observations (or individual measurements) making up the sample. For attribute data, this is the number of units n inspected for the counted characteristic.
sample	a set of measurements (observations) or units (from which counted data is taken) used as a basis for evaluating the process.
specification limits	Limits that define the range within which a product or characteristic conforms to specification or user requirements (also known as tolerance limits).
standard values	known standard values for process range, sample standard deviation, mean or sigma from which control limits can be calculated.
subdiagram	Block diagram within the border of a structure.
subgroup	a set of measurements (observations) taken from a larger set.
subVI	VI used in the block diagram of another VI; comparable to a subroutine.

Glossary

T

- tier chart A chart in which the individual observations in each sample are plotted vertically. It is a useful means of visualizing the variation or spread in each sample.
- top-level VI VI at the top of the VI hierarchy. This term distinguishes the VI from its subVIs.

U

- u chart Control chart that uses the average number of defects per sample to monitor the stability of the process. The sample contains n units, and contains c defects per sample (zero or more defects per unit). The value n may vary from sample to sample.

V

- variables chart A pair of control charts that track whether a process is in control by tracking mean and variability of samples of measured data. Variables charts include X-bar and s, X-bar and R, \bar{x} and mR charts, and mX-bar and mR charts.
- VI library Special file that contains a collection of related VIs for a specific use.

W

- waveform chart A numeric plotting indicator modeled after a paper strip chart recorder, which scrolls as it plots data.
- wire Data path between nodes.
- Wiring tool Tool used to define data paths between source and sink terminals.

X

- x (individual) chart Control chart that uses the individual observations from a process to track the stability of the process mean. This type of control chart is typically used when sample size is one.
- X-bar chart Control chart that uses the sample mean, X-bar, to track the stability of the process mean. Sample size must be two or more.