The LabVIEW Style Book

Peter A. Blume
President, Bloomy Controls, Inc.

Color Gallery

PRENTICE HALL

Upper Saddle River, NJ • Boston • Indianapolis • San Francisco
New York • Toronto • Montreal • London • Munich • Paris • Madrid
Capetown • Sydney • Tokyo • Singapore • Mexico City

Library of Congress Cataloging-in-Publication Data:

Blume, Peter A.
The labVIEW style book / Peter A. Blume.
p. cm.
Includes bibliographical references and index.
ISBN 0-13-145835-3 (hardback : alk. paper) 1. Scientific apparatus and instruments—Computer simulation. 2. LabVIEW. I. Title.
Q185.B568 2007
005.4'38—dc22

2006035871

ISBN 0-13-145835-3
Text printed in the United States at RR Donnelly in Crawfordsville, Indiana.
Fourth Printing, December 2010, corrections made March 2011

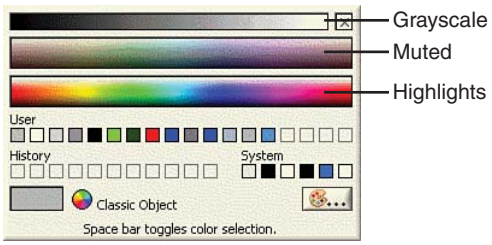

Figure 3-12
The LabVIEW color picker arranges colors into grayscale, muted, and highlight colors.

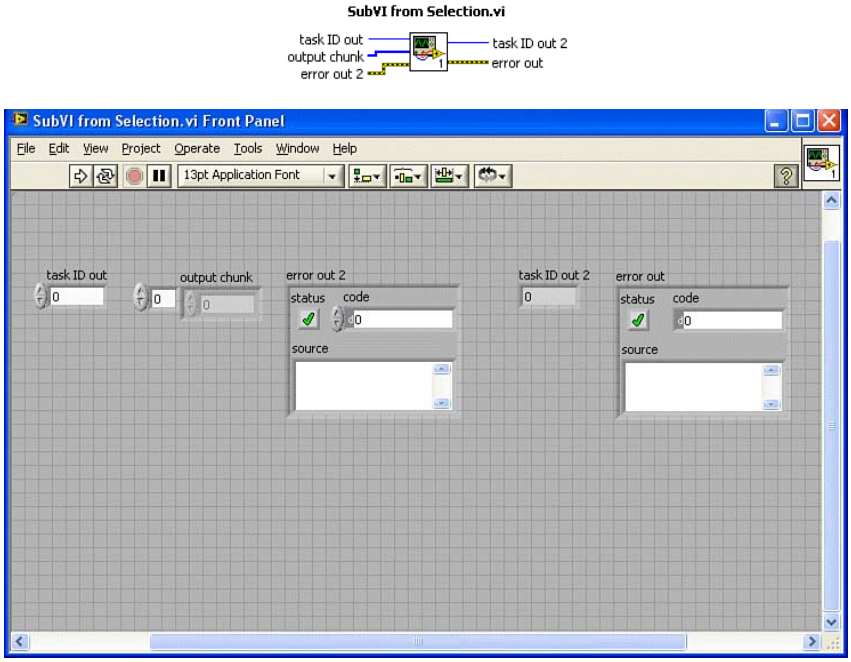

Figure 3-14A
SubVI from Selection VI violates rules 3.15, 3.16, 3.21, and 3.40.

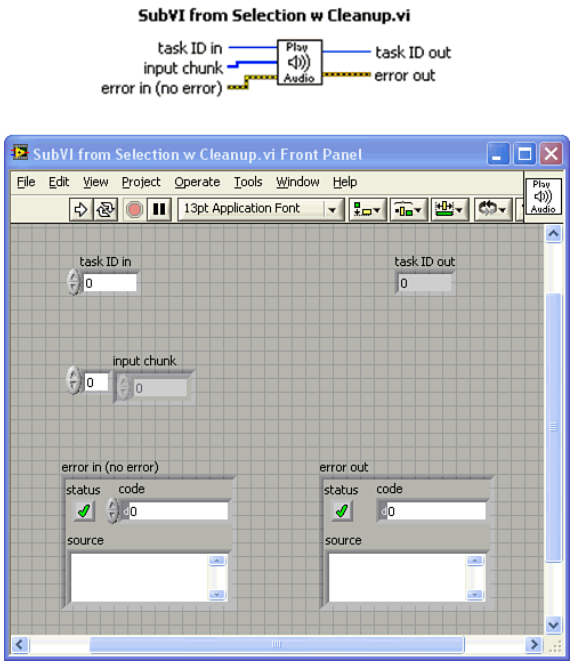

Figure 3-14B
SubVI from Selection w Cleanup VI conforms to the rules for subVI front panels.

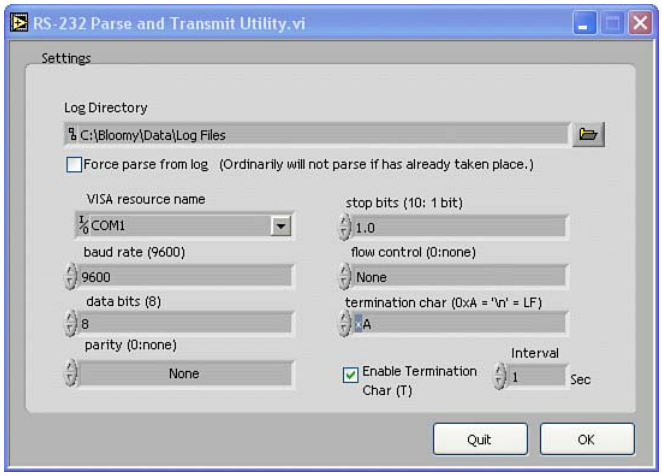

Figure 3-15A
RS-232 Parse and Transmit Utility VI does not have a native operating system appearance.

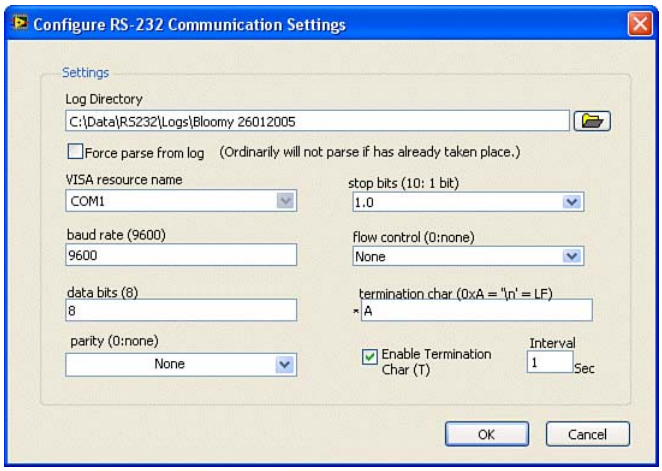

Figure 3-15B
This revision uses Dialog window title and appearance, Dialog fonts, and System controls conforming to the conventions for good dialog VI style.

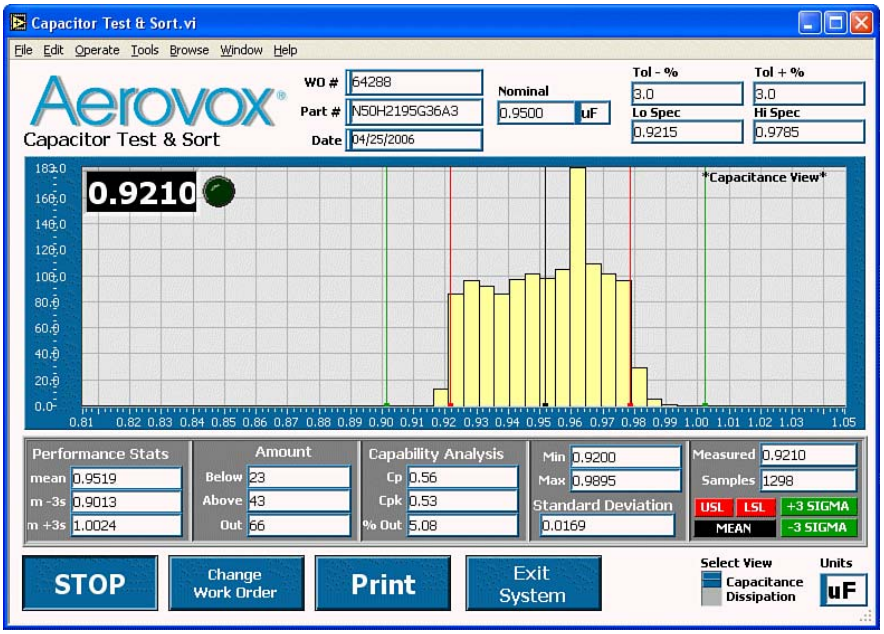

Figure 3-16A
The main GUI VI panel of a Capacitor Test & Sort application has a logical and intuitive layout, but it violates several rules regarding text, menu bars, and overlapping controls.

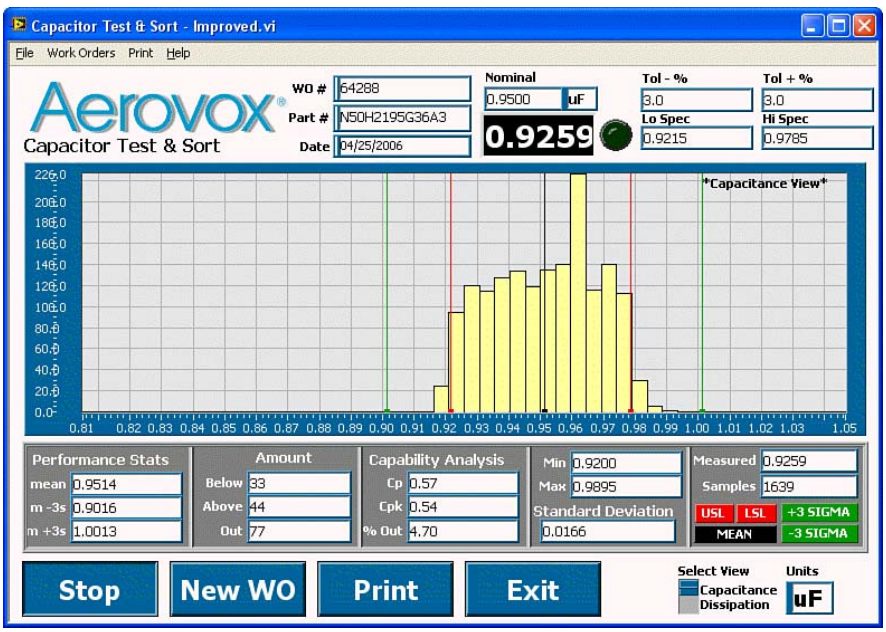

Figure 3-16B
The panel is improved, with succinct Boolean commands and consistent fonts, a custom menu bar for user navigation, and nonoverlapping indicators.

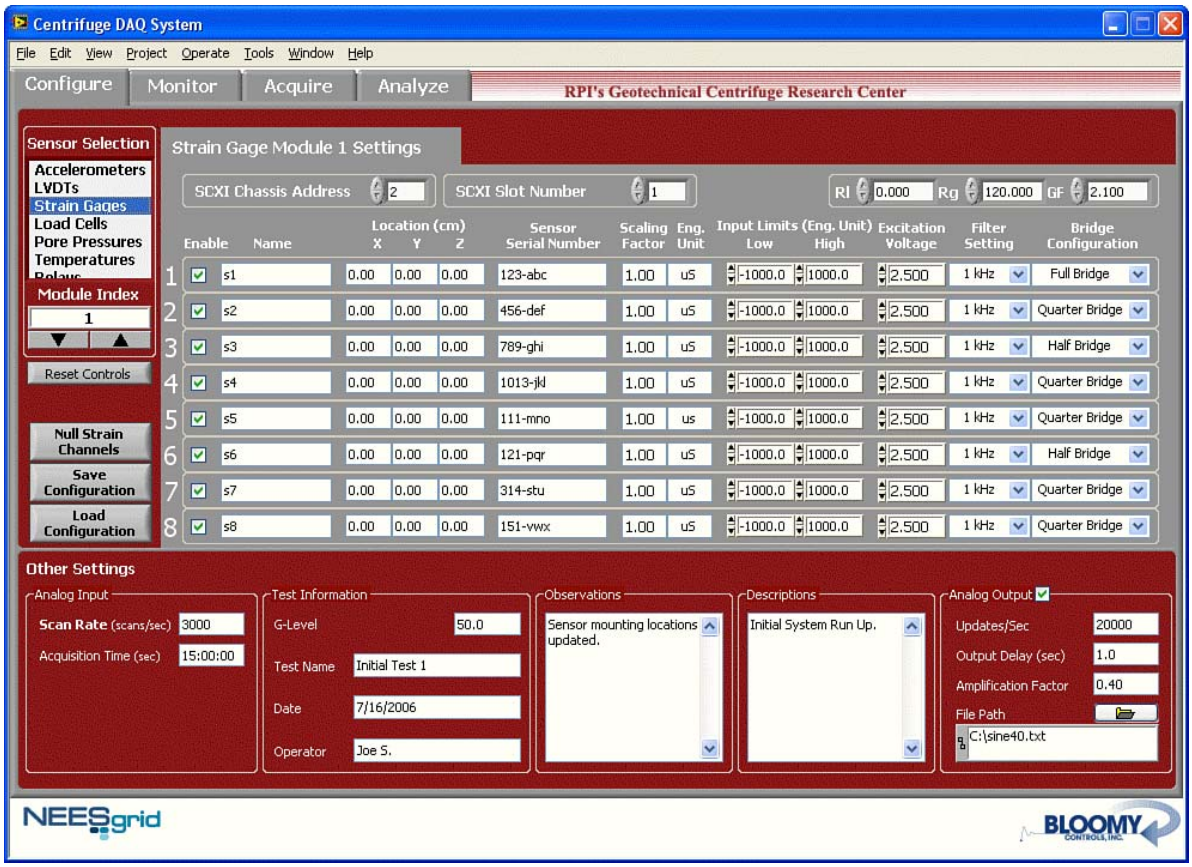

Figure 3-17
Centrifuge DAQ research application for seismic event simulation

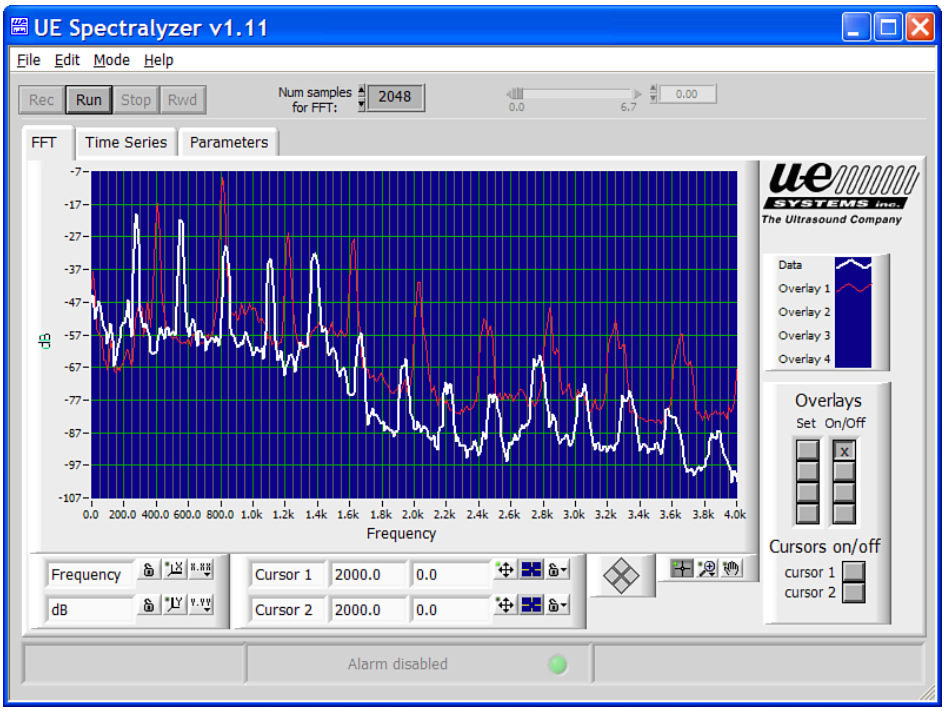

Figure 3-18
Spectralyzer is a commercial application for diagnosing mechanical wear of industrial equipment. The GUI conforms to most of the rules and appears very intuitive.

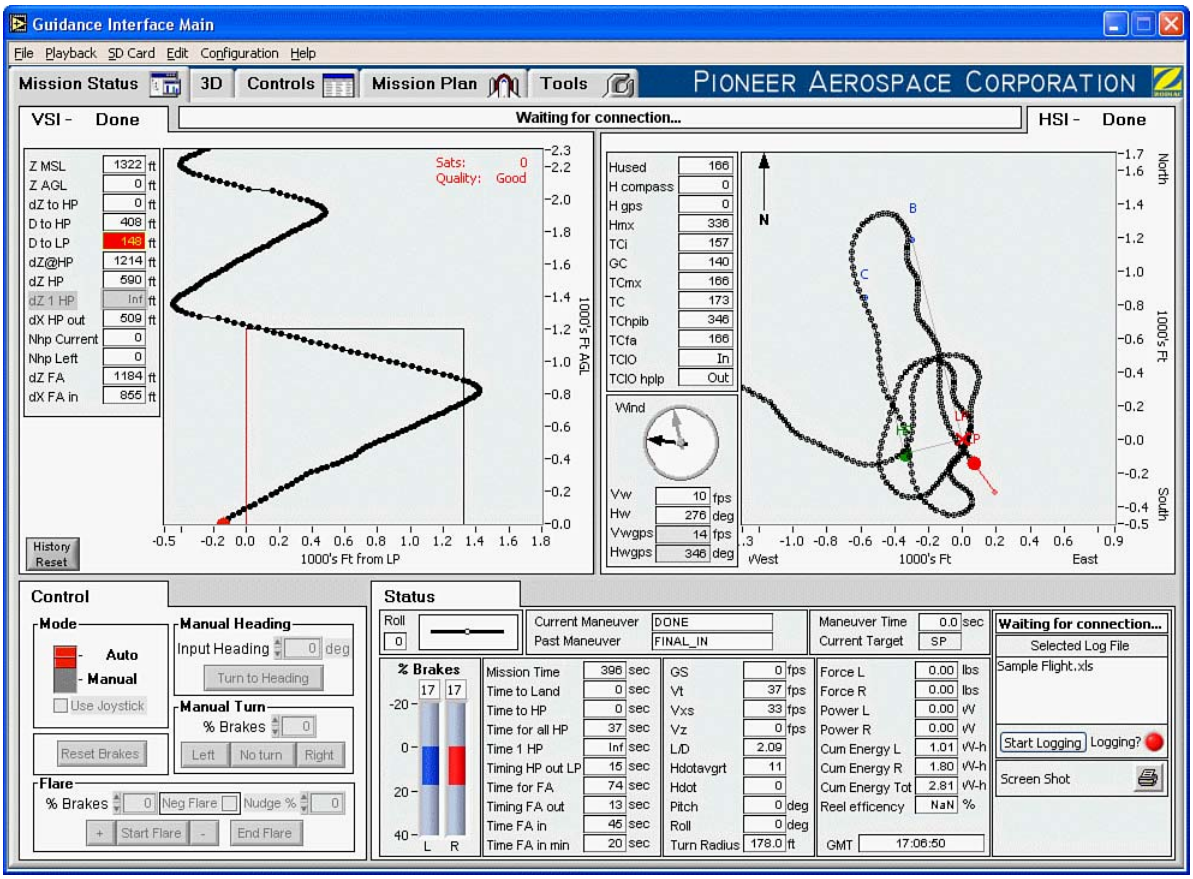

Figure 3-19
Parafoil Guidance Interface is a virtual cockpit. It contains a high density of data while maintaining good readability.

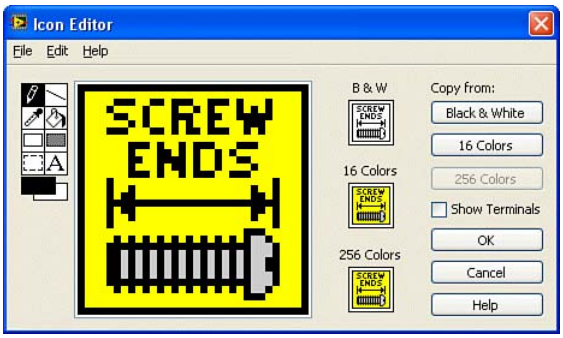

Figure 5-3
The icon of Find Screw Ends VI combines a glyph of a screw with two words created using 8-point small fonts.

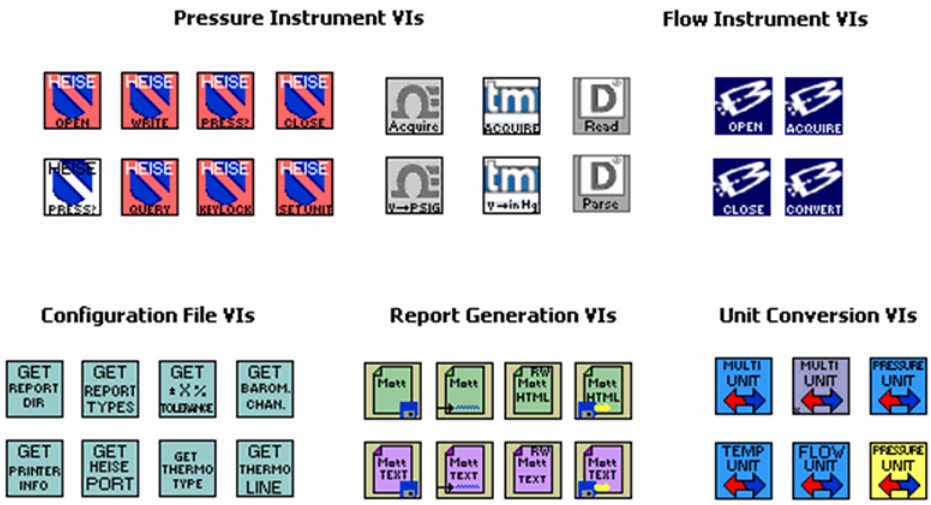

Figure 5-4
An icon convention with unified styles for several sets of related VIs

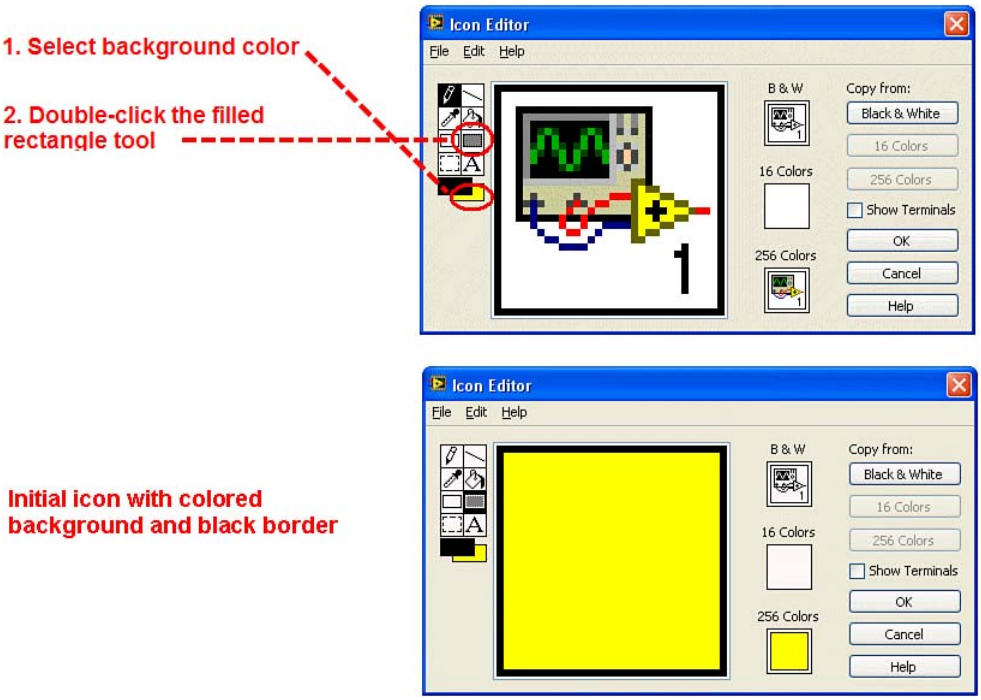

Figure 5-5
The top illustration is a new VI containing a default icon. Select the desired background color and double-click the **Filled Rectangle** tool to initiate an icon with a colored background and black border.

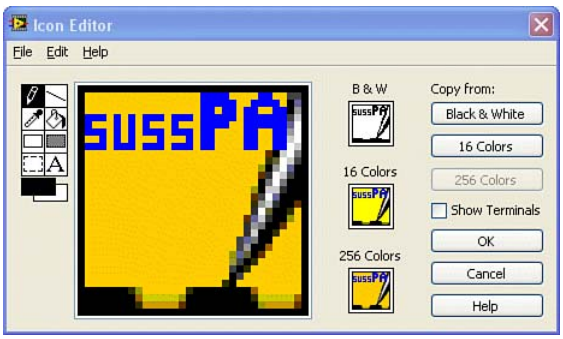

Figure 5-6A
The icon template for the Suss Interface Toolkit contains a glyph of a probe tip contacting a semiconductor device, and an acronym describing the instrument manufacturer and model.

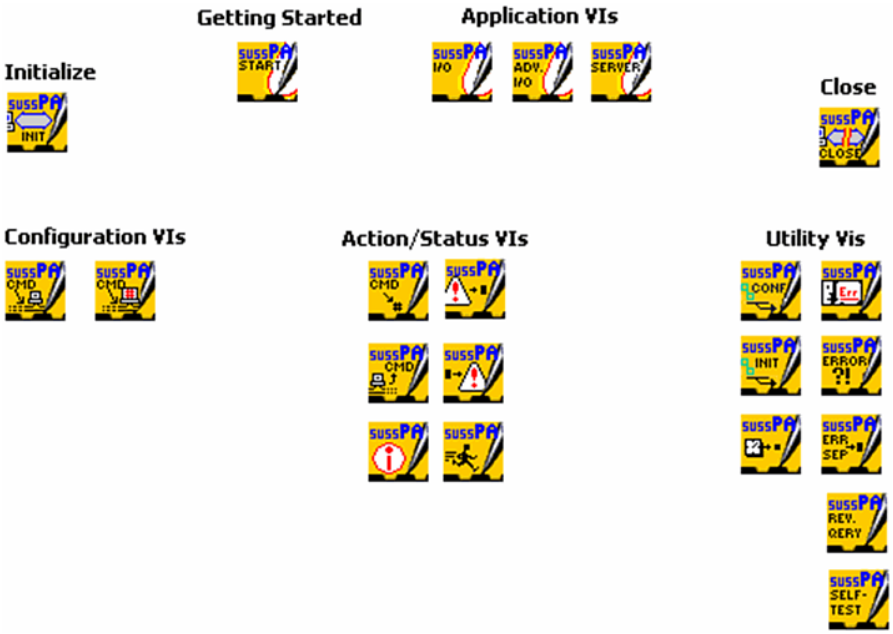

Figure 5-6B
All VIs contained in the toolkit are based on the template, as shown in the VI tree.

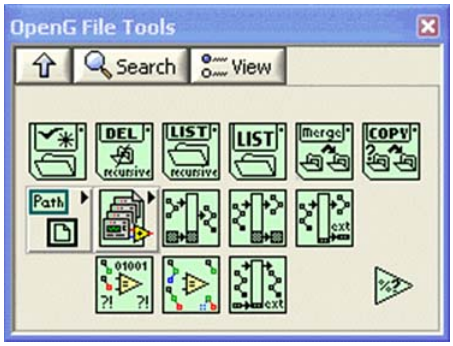

Figure 5-7
The **OpenG File Tools** palette contains icons that resemble functions from the LabVIEW **File I/O** palette. The OpenG icons are shaded green, to distinguish them from the LabVIEW functions.

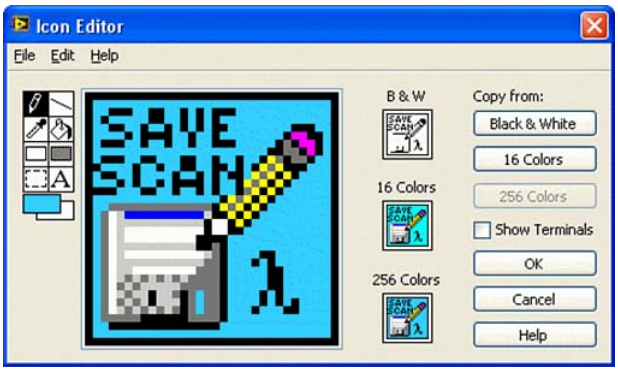

Figure 5-8
The floppy disk and pencil graphics are copied from an Express VI on the **File I/O** palette. The background color and text are customized, for distinction.

Figure 5-14A
SubVI from Selection VI violates multiple style rules, including default icon and counterintuitive terminal labels.

Figure 5-14B
SubVI from Selection w Cleanup VI contains a meaningful icon, intuitive terminal labels, and the standard 4×2×2×4 connector pattern.

Figure 5-15A
Icon contains text drawn freehand, and connector assignments are unconventional.

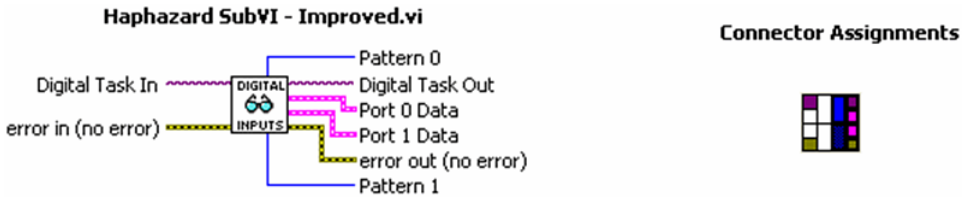

Figure 5-15B
The subVI icon has been improved using 8-point small fonts and a glyph of a pair of glasses. The DAQmx task passes through the top left and right terminals of the 4×2×2×4 connector pattern.

Figure 5-16A
This icon illustrates Bob's bold font.

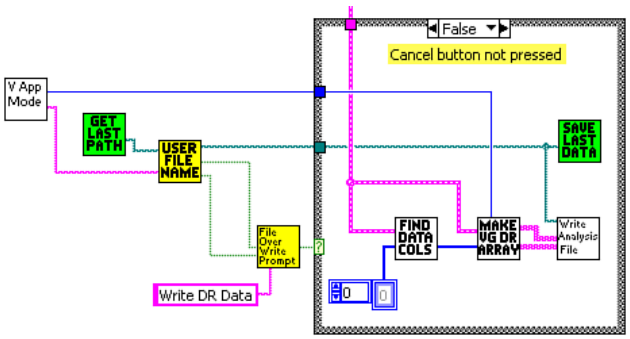

Figure 5-16B
Bob's bold font on several icons in a diagram section appears to shout.

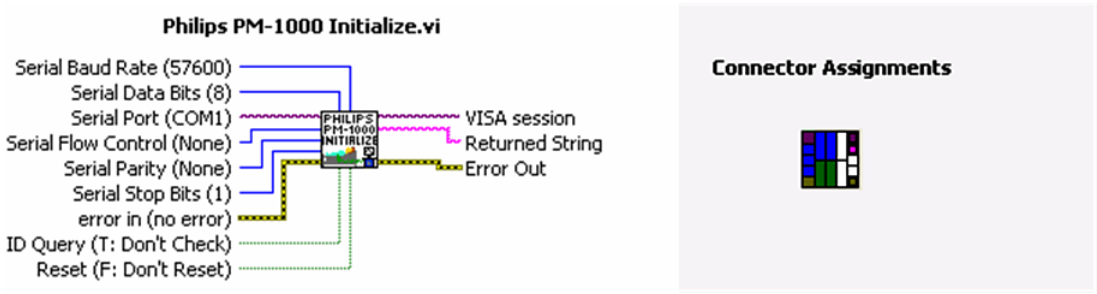

Figure 5-17A
Instrument driver VI for a fictitious medical instrument has a text-heavy icon and too many terminals.

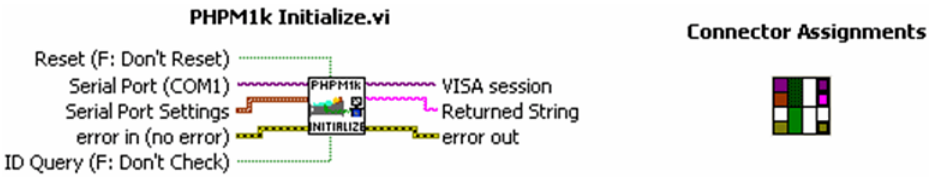

Figure 5-17B
The VI has been improved with a banner containing the instrument prefix, a centered glyph, cluster input, and fewer terminals.

Figure 5-17C
The icon and connector of the Initialize VI generated automatically using the Create New Instrument Driver Project utility. The utility uses a template to create a standard glyph, connector assignments, and terminal labels.

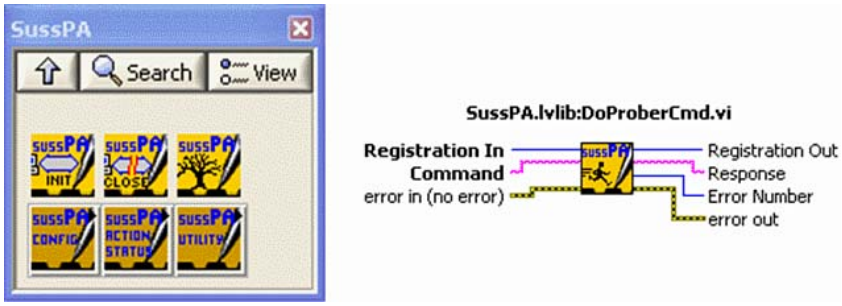

Figure 5-18A
The **Functions** palette for the Suss Interface Toolkit for LabVIEW, a commercial product that controls a semiconductor wafer probe system. All icons share a common glyph describing the instrument. DoProberCmd VI is a subVI from the **Action/Status** subpalette containing an artful silhouette of a runner and common glyph.

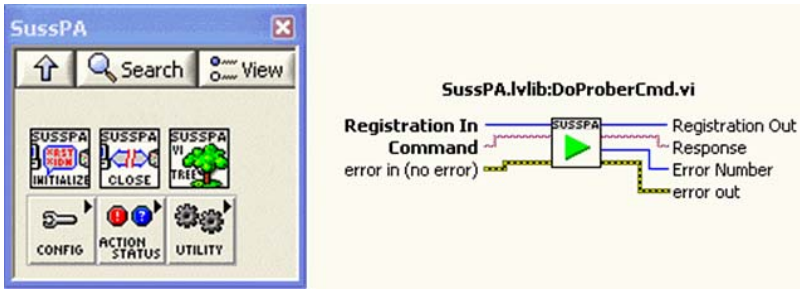

Figure 5-18B
An alternate icon convention consists of the standard glyphs from the instrument driver template. The icon for DoProberCmd VI contains a sideways triangle, a standard glyph for Run, Start, and Initiate. This convention weakens the association with the wafer prober.

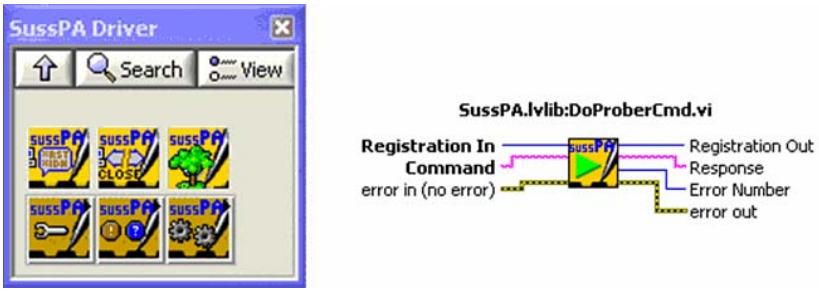

Figure 5-18C
The standard glyphs from the instrument driver template are merged with the instrument-specific glyphs to maintain strong association to both the wafer prober and instrument driver icon conventions.

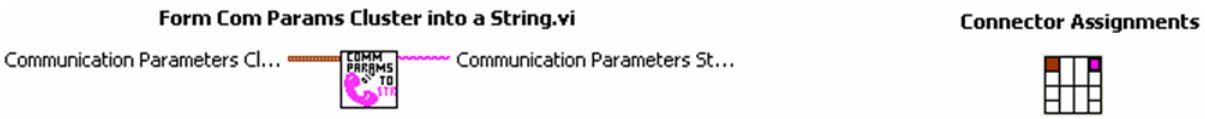

Figure 5-19A
The lengthy terminal labels of Form Com Params Cluster Into a String VI are truncated by the Context Help window.

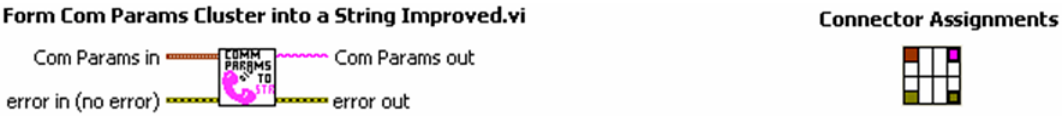

Figure 5-19B
The label lengths have been reduced, and error terminals have been applied to promote proper data flow.

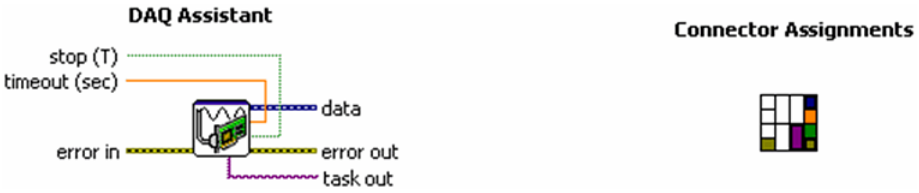

Figure 5-20
DAQ Assistant Express VI has inputs assigned to terminals on the right, causing wire crossovers on the calling VI diagrams.

Figure 5-21A
Confirm Quit VI has a simple graphic that has been copied from a Windows system prompt. The type of confirmation is not identified by the icon.

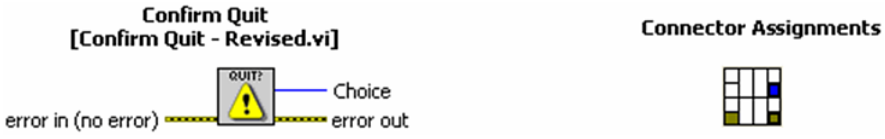

Figure 5-21B
The icon has been modified to indicate the type of confirmation. The standard 4×2×2×4 connector pattern has been applied, as well as the assignment of error terminals to facilitate execution ordering.

Figure 5-22
Print VI in Landscape Mode VI is composed of three-quarters demonstrative graphic and one-quarter text.

Figure 5-23
Dynamic VI Path Builder VI is composed of two-thirds demonstrative graphic and one-third text.

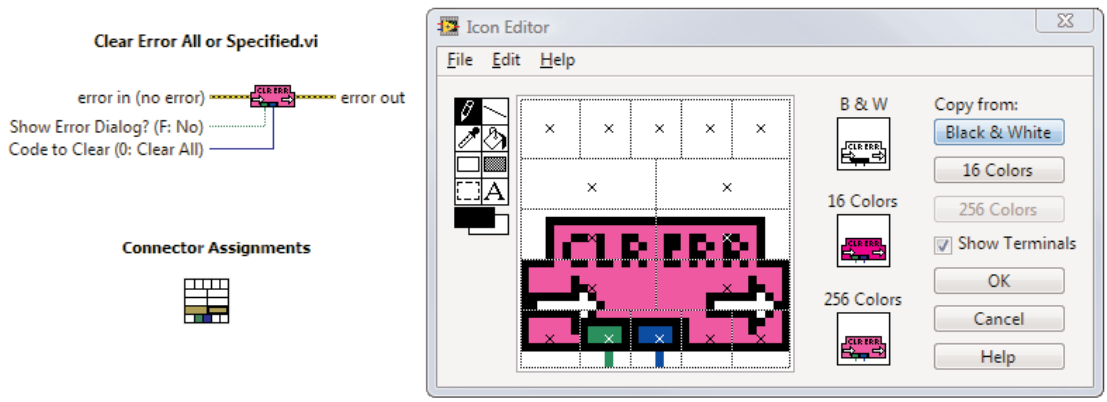

Figure 5-24
Clear Error All or Specified VI has a custom border providing a unique shape, and a 5×3×3×5 connector pattern rotated 90 degrees.

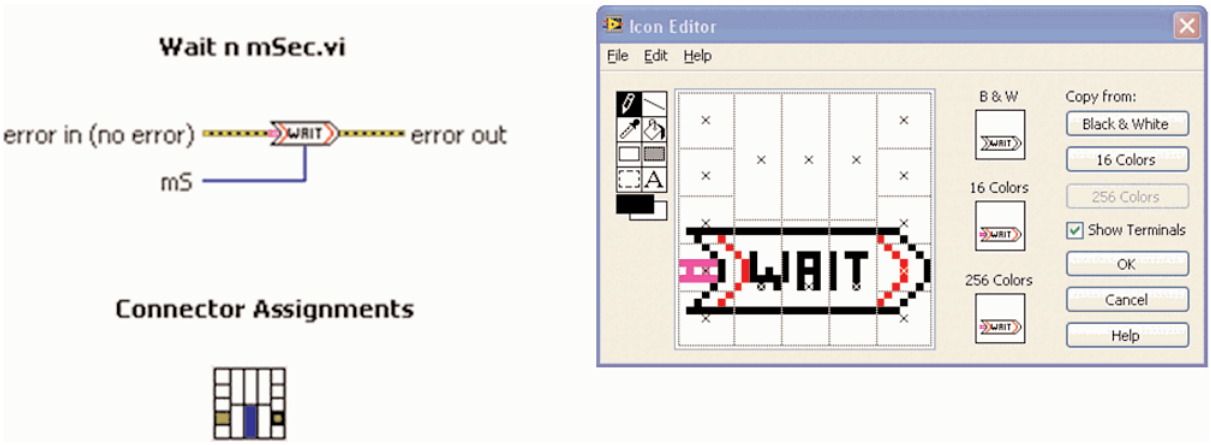

Figure 5-25
Wait n mSec VI has a custom border and promotes good data flow style. It has a 5×3×3×5 connector pattern with three terminals assigned.

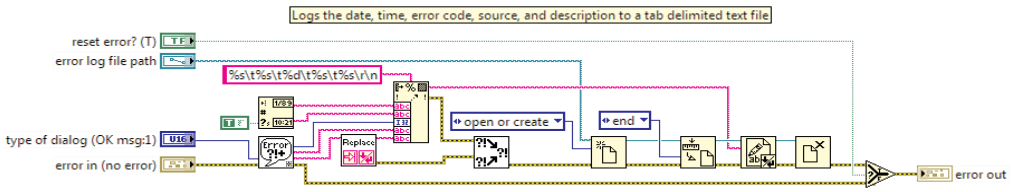

Figure 7-9A
The date and time are combined with the error message, source, and code, and are logged to a tab-delimited text file. Readability is maximized using this approach.

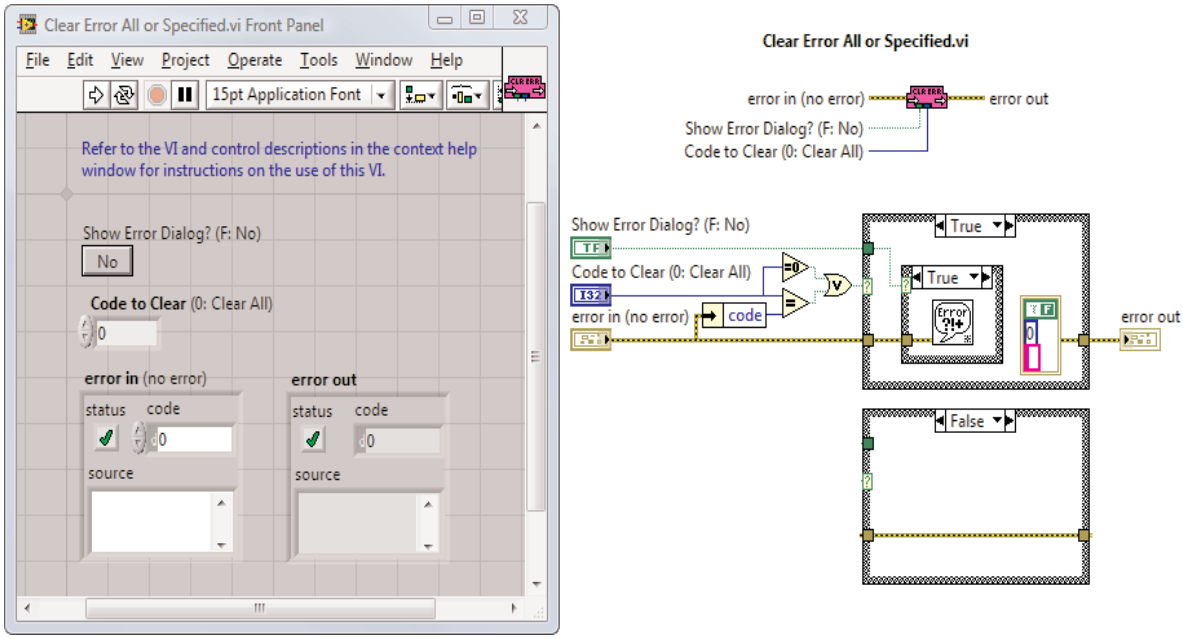

Figure 7-16
Clear Error All or Specified VI is a utility that selectively clears errors and warnings according to the value of **Code to Clear**.

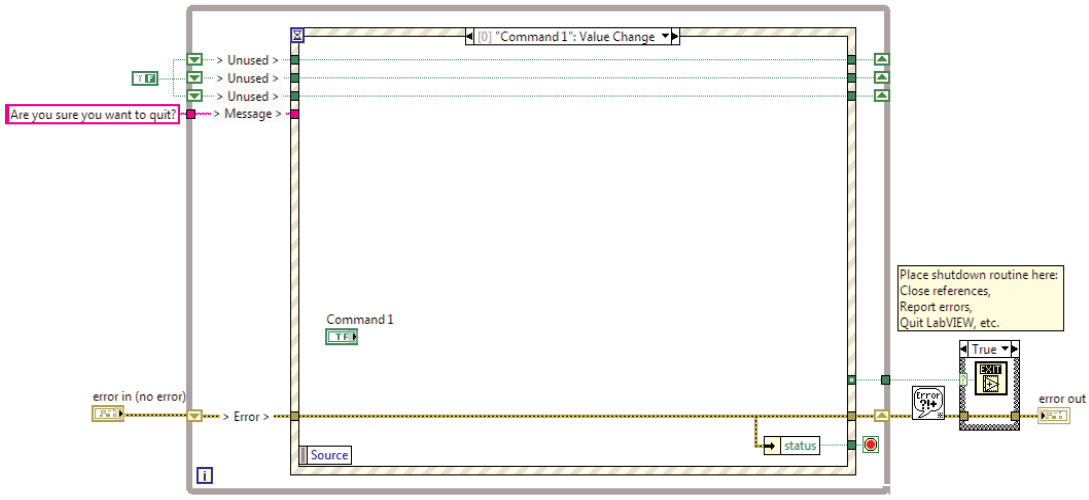

Figure 8-6B
The diagram consists of an Event structure within a While Loop containing shift registers, and a shutdown routine outside the loop. The control terminals are in their respective **Value Change** event cases.

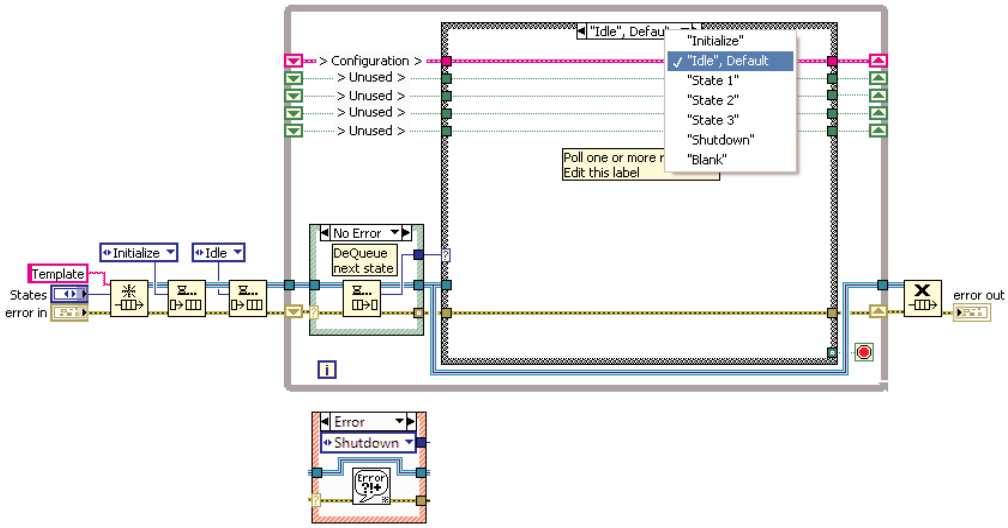

Figure 8-12
A template for the Queued State Machine design pattern, which maintains multiple states in a buffer.

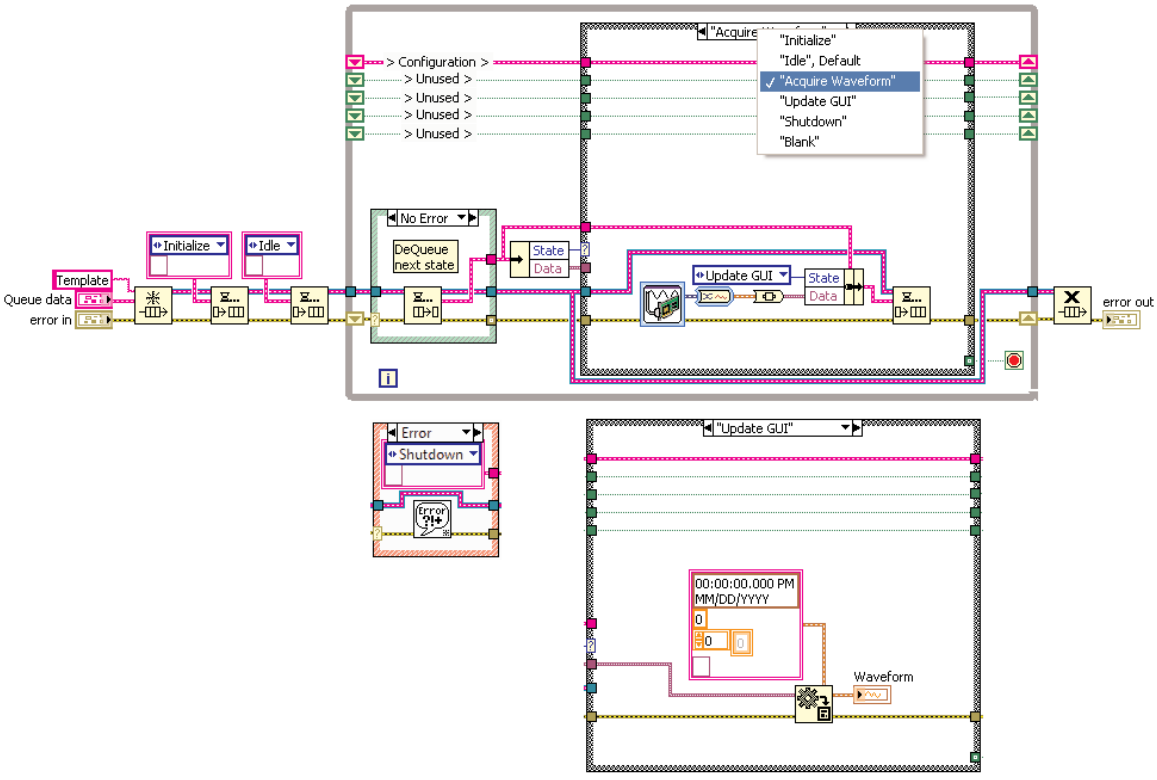

Figure 8-13
The queue is used to pass data between states via a cluster containing a variant and state enumeration. This reduces wires and shift registers, but the data is not accessible in subsequent states after dequeuing.

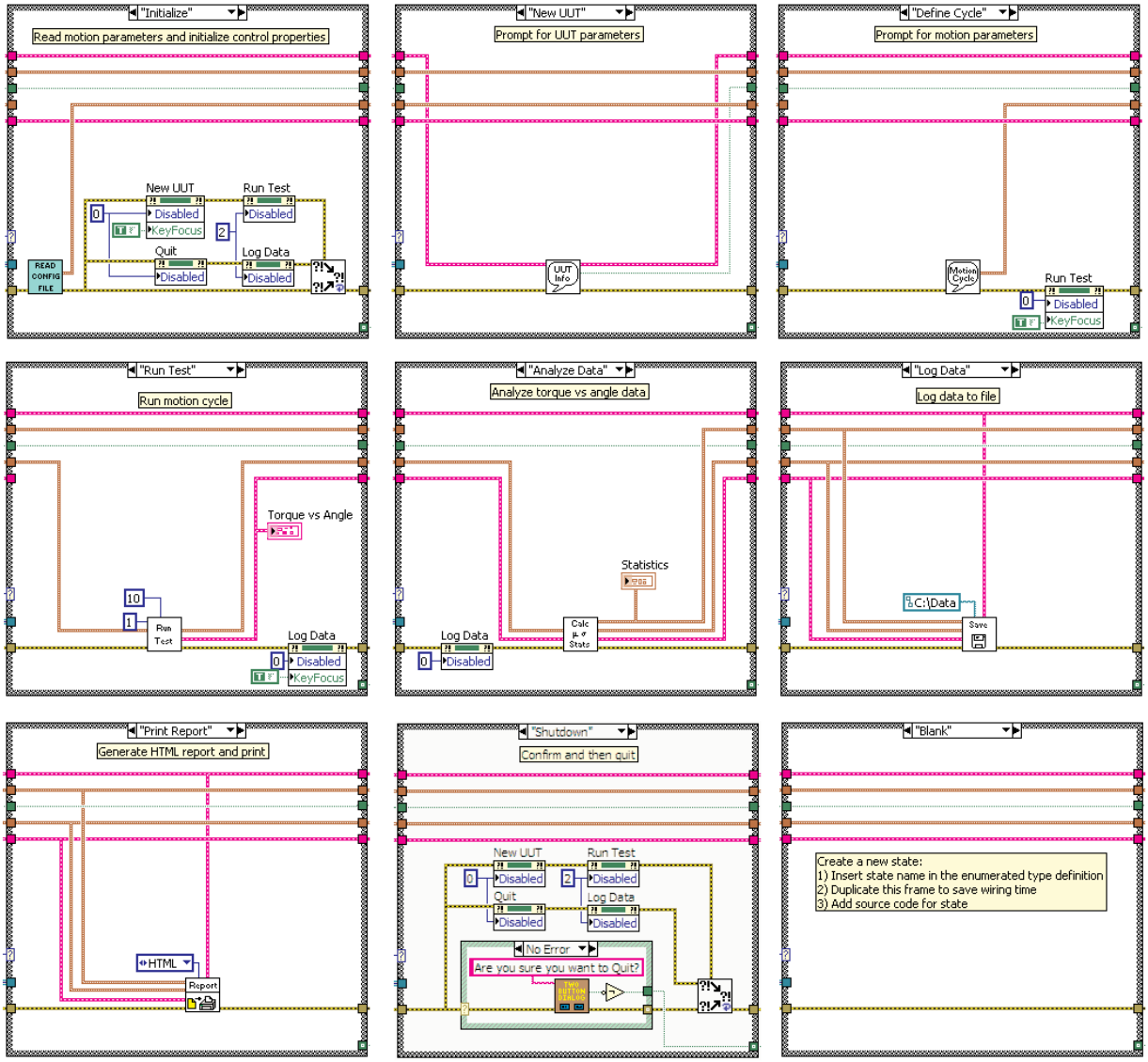

Figure 8-14C
A separate state is delineated for each of the application's primary tasks, in addition to **Initialize**, **Idle**, **Shutdown**, and **Blank**.