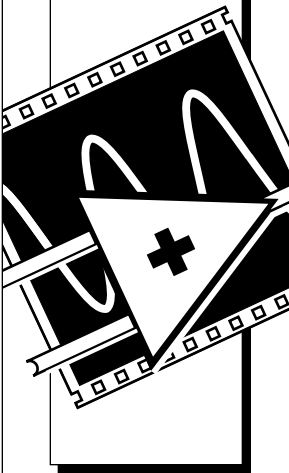


LabVIEW



LabVIEW® Demonstrationshandbuch

Ausgabe March 1996
Teilenummer 321293A-01

© Copyright 1993, 1996 National Instruments Corporation.
Alle Rechte vorbehalten.



Internet-Unterstützung

GPIB: gpiib.support@natinst.com
 DAQ: daq.support@natinst.com
 VXI: vx1.support@natinst.com
 LabVIEW: lv.support@natinst.com
 LabWindows: lw.support@natinst.com
 HiQ: hiq.support@natinst.com

E-Mail Deutschland: nig.cs@natinst.com
 FTP-Standort: <ftp.natinst.com>
 Web-Adresse: www.natinst.com



Bulletin Board Unterstützung

BBS USA: +(512) 794-5422
 BBS Großbritannien: 01635 551422
 BBS Deutschland: (8-N-1) 089 712 561



FaxBack-Unterstützung (USA)

+(512) 418-1111



Telefon-Unterstützung (USA)

**National Instruments
 Germany GmbH**
 Konrad-Celtis-Str. 79
 81369 München
 Tel: 089/741 31 30
 Fax: 089/714 60 35

**National Instruments
 Ges.m.b.H.**
 Plainbachstr. 12
 5101 Salzburg-Bergheim
 Tel: 0662/45 79 90-0
 Fax: 0662/45 79 90-19

**National Instruments
 Switzerland**
 Sonnenbergstr. 53
 Ch-5408 Ennetbaden
 Tel: 056/200 51 51
 022/980 05 11 (Genf)
 Fax: 056/200 51 55



Internationale Zweigstellen

Australien 03 9 879 9422, Belgien 02 757 00 20, Dänemark 45 76 26 00, Finnland 90 527 2321,
 Frankreich 1 48 14 24 24, Großbritannien 01635 523545, Hongkong 2645 3186, Italien 02
 48301892, Japan 03 5472 2970, Kanada (Ontario) 519 622 9310, Kanada (Québec) 514 694 8521,
 Korea 02 596 7456, Mexiko 5 202 2544, Niederlande 03480 33466, Norwegen 32 84 84 00,
 Schweden 08 730 49 70, Singapur 2265886, Spanien 91 640 0085, Taiwan 02 377 1200

National Instruments Firmenhauptsitz

6504 Bridge Point Parkway Austin, Texas 78730-5039 Telefon: +(512) 794-0100

Wichtige Informationen

Garantie

National Instruments garantiert, daß die Medien, auf denen Sie die National Instruments Software erhalten, nicht aufgrund fehlerhaften Materials oder fehlerhafter Ausführung bei der Ausführung der Programmieranweisungen versagen. Dies gilt für 90 Tage vom Versandsdatum an, wie durch Quittungen oder andere Unterlagen belegt. National Instruments wird die Software-Medien, die die Programmieranweisungen nicht ausführen, nach seiner Wahl reparieren oder ersetzen, falls National Instruments innerhalb der Garantiefrist eine Benachrichtigung zu solchen Fehlern erhält. National Instruments garantiert nicht, daß der Betrieb der Software ununterbrochen oder fehlerfrei ist.

Es muß eine "Return Material Authorization" (RMA) Nummer (Berechtigungsnummer für Warenrücksendung) vom Werk angefordert und deutlich auf der Außenseite der Verpackung vermerkt werden, bevor ein Gerät für Garantiearbeiten angenommen wird. National Instruments übernimmt die Versandkosten für die Rücksendung derjenigen Teile an den Eigentümer, die durch die Garantie abgedeckt sind.

National Instruments ist davon überzeugt, daß die Informationen in diesem Handbuch genau sind. Die Unterlage wurde sorgfältig auf technische Genauigkeit geprüft. Im Falle von technischen Fehlern oder Druckfehlern behält sich National Instruments das Recht vor, ohne vorherige Bekanntgabe gegenüber Besitzern dieser Ausgabe Änderungen an nachfolgenden Ausgaben vorzunehmen. Der Leser sollte sich mit National Instruments in Verbindung setzen, wenn Fehler vermutet werden. National Instruments ist unter keinen Umständen für Schäden, die aufgrund dieser Unterlage oder der darin enthaltenen Informationen entstehen oder mit ihnen zusammenhängen, haftbar.

VORBEHALTLICH WIE HIER ANGEGEBEN, GIBT NATIONAL INSTRUMENTS KEINERLEI EXPLIZITE ODER IMPLIZITE GARANTIE UND LEHNT INSBESONDERE JEGLICHE GARANTIE AUF HANDELSÜBLICHE QUALITÄT UND EIGNUNG FÜR EINEN BESTIMMTEN ZWECK AB. DER ANSPRUCH DES KUNDEN AUF ERSATZ FÜR SCHÄDEN, DIE DURCH FEHLER ODER FAHRLÄSSIGKEIT VON NATIONAL INSTRUMENTS VERURSACHT WURDEN, IST DAHER AUF DEN ZUVOR VOM KUNDEN GEZÄHLTEN BETRAG BESCHRÄNKT. NATIONAL INSTRUMENTS IST NICHT HAFTBAR FÜR AUS DATENVERLUST, GEWINNVERLUST, ANWENDUNG VON PRODUKTEN ENTSTEHENDE SCHÄDEN ODER FÜR BEILÄUFIGE ODER MITTELBARE SCHÄDEN, AUCH WENN ES AUF DIESE MÖGLICHKEIT HINGEWIESEN WIRD. Diese Haftungsbeschränkung von National Instruments gilt unabhängig von der Art der Klage, ob als Vertragsklage oder Klage wegen unerlaubter Handlung, einschließlich Fahrlässigkeit. Jede Klage gegen National Instruments muß innerhalb eines Jahres, nachdem der Klagegrund entstanden ist, vorgebracht werden. National Instruments ist nicht für verspätete Erfüllungen aufgrund von Ursachen jenseits seiner angemessenen Kontrolle haftbar. Die hierin geleistete Garantie deckt keine Schäden, Fehler, Störungen oder Wartungsversagen ab, die dadurch entstehen, daß der Eigentümer die Installations-, Bedienungs- und Wartungsanweisungen von National Instruments nicht befolgt; daß der Eigentümer Änderungen am Produkt vornimmt; die durch falschen Gebrauch, Mißbrauch oder fahrlässige Handlungen des Eigentümers entstehen; oder die durch Stromausfall oder Stromstöße, Feuer, Überflutung, Unfall, Handlungen Dritter oder andere Ereignisse jenseits der angemessenen Kontrolle entstehen.

Urheberrecht

Laut Urheberrecht darf diese Unterlage nicht ohne vorherige schriftliche Genehmigung von National Instruments Corporation auf irgendeine Art elektronisch oder mechanisch – einschließlich Fotokopien, Aufnahmen, Speichern auf einem Datenwiedergewinnungssystem oder Übersetzungen – ganz oder in Teilen vervielfältigt oder übertragen werden.

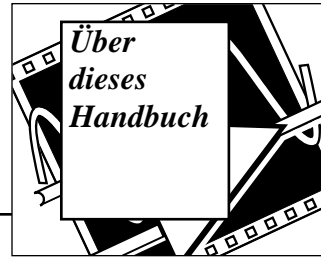
Warenzeichen

LabVIEW® und NI-488M™ sind Warenzeichen von National Instruments Corporation.

Die aufgelisteten Produkt- und Firmennamen sind die Warenzeichen oder Handelsbezeichnungen der jeweiligen Unternehmen.

WARNUNG HINSICHTLICH MEDIZINISCHER UND KLINISCHER VERWENDUNG VON NATIONAL INSTRUMENTS PRODUKTEN

National Instruments Produkte sind nicht mit Hilfe von Komponenten und Tests entworfen, die ein Maß an Zuverlässigkeit vorsehen, durch das sie zur Verwendung bei Behandlung und Diagnose von Menschen geeignet sind. Anwendungen von National Instruments Produkten, in die eine medizinische oder klinische Behandlung einbezogen wird, können eine potentielle Verletzungsgefahr aufgrund von Produktversagen oder Fehlern seitens des Benutzers oder Anwendungs-Entwicklers darstellen. Jegliche Verwendung oder Anwendung von National Instruments Produkten für oder im Zusammenhang mit medizinischer oder klinischer Behandlung darf nur von ordnungsgemäß geschultem und qualifiziertem medizinischen Personal erfolgen, und alle herkömmlichen medizinischen Schutzeinrichtungen, Geräte und Verfahren, die in der jeweiligen Situation zur Vermeidung von schweren Verletzungen oder Tod angemessen sind, sollten auch weiterhin bei Einsatz von National Instruments Produkten angewendet werden. National Instruments Produkte sollen KEINEN Ersatz darstellen für irgendeine Form eines herkömmlichen Vorgangs, Verfahrens oder Geräts, das zu Überwachung oder Schutz menschlicher Gesundheit und Sicherheit bei medizinischer oder klinischer Behandlung dient.



Über dieses Handbuch

Das LabVIEW Demonstrations enthält alle Informationen, die Sie zum Arbeiten mit dem Softwarepaket Laboratory Virtual Instrument Engineering Workbench (LabVIEW) benötigen. LabVIEW dient der Vereinfachung von wissenschaftlichen Berechnungen, der Prozeßsteuerung sowie Test- und Meßanwendungen; ferner kann es auch für eine ganze Reihe weiterer Programmieranwendungen eingesetzt werden.

Aufbau dieses Handbuchs

Dieses Handbuch ist folgendermaßen aufgebaut:

- In der Einführung in die LabVIEW-Demonstration werden die ersten Schritte mit dem LabVIEW-Demonstrationspaket sowie die verschiedenen, in ihm enthaltenen Demonstrationen erläutert.
- In Kapitel 1, *Einführung in LabVIEW*, wird beschrieben, was LabVIEW ist, was ein virtuelles Instrument (VI) ist, wie die LabVIEW-Umgebung (Fenster, Menüs, Paletten und Tools) zu verwenden ist und wie VIs erstellt, bedient und bearbeitet werden.
- In Kapitel 2, *Erstellung eines SubVIs*, wird beschrieben, was ein SubVI ist, wie man ein Icon und einen Connector erstellt, und wie man ein VI als SubVI einsetzt.
- In Kapitel 3, *Schleifen und Charts*, werden While-Schleifen, die Darstellung von Daten in einem Chart, die Bedeutung und Verwendung von Shift Registers sowie die Verwendung von For-Schleifen erläutert.
- In Kapitel 4, *Arrays, Clusters und Graphs*, wird die Erstellung von Arrays sowie die Verwendung von grundlegenden Array-Funktionen, Clusters und Graphs erläutert. Hier erfahren Sie ebenfalls, was Polymorphismus ist, und wie man Graphs zur Darstellung von Daten verwendet.
- In Kapitel 5, *Case- und Sequence-Strukturen und der Formula Node*, wird die Verwendung von Case- und Sequence-Strukturen sowie Sequence Locals und Formula Nodes erläutert.

- In Kapitel 6, *Strings und File I/O*, lernen Sie die Erstellung von String Controls und Indicators und erfahren näheres über die Verwendung von String-Funktionen, Vorgänge bei der Datei Ein-/Ausgabe, das Sichern von Daten in Tabellenkalkulationsdateien sowie das Schreiben und Lesen von Daten aus Textdateien.
- In Kapitel 7, *Datenerfassung* (für Windows, Macintosh und Sun) *und Gerätesteuerung*, wird erläutert, wie Einsteckkarten zur Datenerfassung verwendet werden. Ferner erfahren Sie hier näheres zu VISA, GPIB, der Steuerung einer Serial-Port-Schnittstelle von LabVIEW, VXI (für Windows, Macintosh und Sun), Gerätetreiber und deren Verwendung sowie die Verwendung eines Frequency Response Test VIs.

Konventionen in diesem Handbuch

	In diesem Handbuch werden die folgenden Konventionen verwendet:
Fettdruck	Fettgedruckter Text dient der Kennzeichnung von Menüs, Menüpunkten und Dialogfeldschaltflächen bzw. -optionen sowie VI-Ein-/Ausgabeparametern.
<i>Kursivdruck</i>	Kursiv gedruckter Text dient der Kennzeichnung einer Betonung, eines Querverweises oder der Einführung eines Schlüsselkonzepts.
fett/kursiv	Fett- und kursiv gedruckter Text dient der Kennzeichnung eines Hinweises, einer Mahnung zur Vorsicht oder einer Warnung.
einschritt	Einschriftschrift dient der Kennzeichnung von Text bzw. Zeichen, deren Eingabe über die Tastatur erfolgt. Code-Abschnitte, Programmier- und Syntaxbeispiele sowie Meldungen und Antworten, die vom Computer automatisch am Bildschirm ausgegeben werden, sind ebenfalls in dieser Schriftart angezeigt.
<i>kursiv/</i> <i>einschritt</i>	Kursivschrift in dieser Schriftart gibt an, daß anstelle dieser Platzhalter die entsprechenden Wörter bzw. Werte einzugeben sind.
<>	Eckige Klammern dienen der Kennzeichnung einer Taste auf der Tastatur, z.B. <Shift>.
-	Ein Bindestrich zwischen mehreren, in eckigen Klammern gesetzten Tastennamen bedeutet, daß die genannten Tasten gleichzeitig zu drücken sind, z.B. <Shift-Entf>.

» Mit dem Symbol » werden Sie durch verschachtelte Menüpunkte und Dialogfeldoptionen zum letztendlich auszuführenden Punkt geleitet. Durch die Folge

File»Page Setup»Options»Substitute Fonts

werden Sie wie folgt angewiesen: Das Menü File wählen, den Punkt Page Setup wählen, anschließend Options wählen, und zum Abschluß im letzten Dialogfeld die Option Substitute Fonts wählen.

Pfade In diesem Handbuch werden Pfade mit einem negativen Schrägstrich (\) gekennzeichnet. Laufwerksnamen, Verzeichnisse und Dateien werden folgendermaßen getrennt: Laufwerksname\Verz1Name\Verz2Name\meineDatei.

IEEE 488.1 und IEEE 488.2 IEEE 488.1 und IEEE 488.2 beziehen sich auf den ANSI/IEEE-Standard 488.1-1987 sowie den ANSI/IEEE-Standard 488.2-1987, von denen GPIB definiert wird.

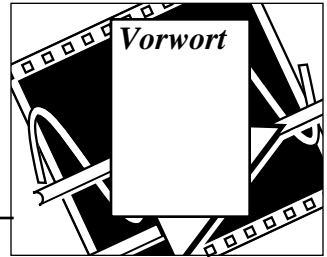


Hinweis: *Dieses links von fett- und kursivgedrucktem Text abgebildete Symbol dient der Kennzeichnung eines Hinweises, mit dem Sie auf wichtige Informationen aufmerksam gemacht werden.*

Kundenschulung

National Instruments bietet Kurse, in denen Teilnehmer in die Grundlagen (Basics) und fortgeschrittene Anwendungen (Advanced) von LabVIEW eingeführt werden, damit sie schnell mit LabVIEW vertraut werden und erfolgreiche Anwendungen entwickeln können. Im umfassenden Basics-Kurs lernen Sie nicht nur die Grundlagen von LabVIEW, Sie haben auch die Gelegenheit, praktische Erfahrungen bei der Entwicklung von Anwendungen zur Datenerfassung und Gerätesteuerung zu sammeln. Im darauf aufbauenden Advanced-Kurs erfahren Sie, wie die Leistungsfähigkeit und Effizienz von LabVIEW-Anwendungen gesteigert werden können. Einen umfassenden Katalog der Kursinhalte, -gebühren und -termine erhalten Sie auf Wunsch von National Instruments.

Einführung in die LabVIEW-Demonstration

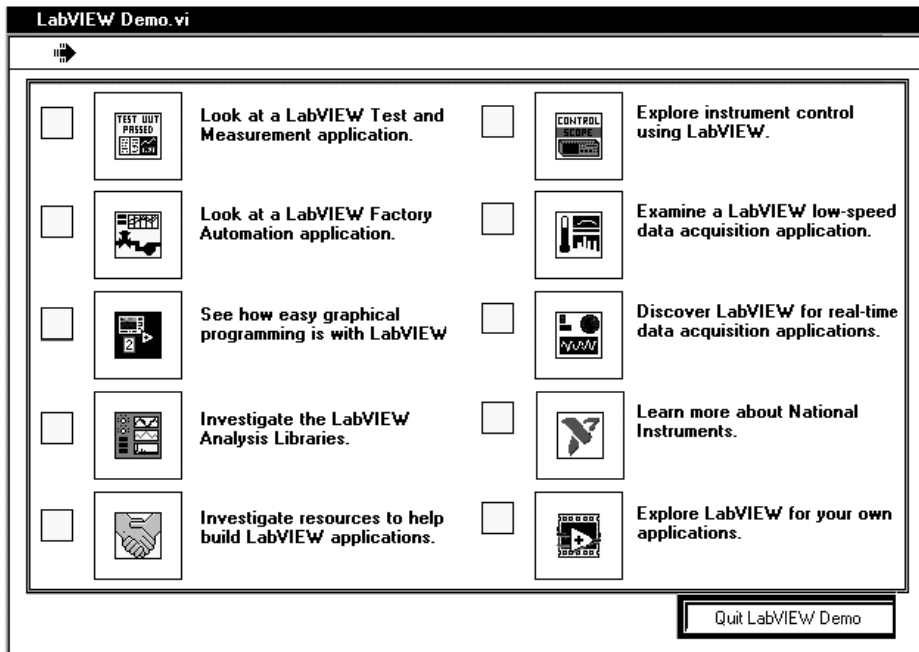


Dieses Vorwort gibt eine Einführung in das LabVIEW-Demonstrationspaket und erklärt die verschiedenen Demonstrationen, die Sie in diesem Paket aufrufen können.



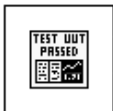
Um die LabVIEW-Demonstrationspaketanwendung aufzurufen, klicken Sie auf das LabVIEW-Symbol im LabVIEW-Demonstrationsordner. Das LabVIEW-Demo-VI erscheint automatisch auf dem Bildschirm, wenn Sie das LabVIEW-Demonstrationspaket öffnen.

Dieses Demonstrations-VI ermöglicht einen kurzen Blick auf die vielen Möglichkeiten, mit denen LabVIEW Ihren Softwarebedürfnissen gerecht werden kann. Durch Anklicken einer Kategorie-Taste erhalten Sie eine Demonstration über ein bestimmtes Programm, das in LabVIEW ausgeführt wird. Jede aufgerufene Kategorie wird markiert, damit Sie den Überblick über das behalten, was bereits eingesehen wurde.



Wenn Sie auf eine der Kategorien klicken, erhalten Sie ein in LabVIEW geschriebenes Applikations-Beispiel. Sie können sich jede dieser Demo-Anwendungen anschauen, um zu sehen, wie sie funktionieren. Wenn Sie mehr über die Demonstration, die Sie gerade sehen, wissen möchten, klicken Sie auf die blaue Taste "Weitere Informationen" im unteren Teil der Anwendung, oder drücken Sie die <F5>-Taste. Um zum Hauptmenü der Demonstrationen zurückzukehren, klicken Sie auf die rote "Zurück"-Taste, die sich ebenfalls unten in der Anwendung befindet, oder drücken Sie die Taste <F4>.

Die folgende Liste beschreibt jede einzelne Kategorie im Demo-VI.



Blick auf LabVIEW-Test- und -Messungsanwendung.

Demonstriert den LabVIEW Test Executive. Dies ist eine in LabVIEW

entwickelte Anwendung zur Kontrolle von Produktions- und Herstellungs-Testanwendungen.



Blick auf eine LabVIEW-Betriebsanlagen-Automationsanwendung.

Simuliert eine in LabVIEW erstellte Anwendung zur Prozeßvisualisierung und -steuerung.



Grafisches Programmieren mit LabVIEW leicht gemacht.

Einführung in grafisches Programmieren und Vermittlung von Grundkenntnissen über den Aufbau virtueller Instrumente (VIs) in LabVIEW.



Erforschen Sie die LabVIEW-Analysebibliotheken.

Demonstriert die LabVIEW-Analysebibliotheken und gibt eine Einführung zu deren Gebrauch beim Aufbau von Analysesystemen.



Ressourcen erkunden, mit denen Sie LabVIEW-Anwendungen aufbauen können.

Zeigt Ihnen den Weg zu den zahlreichen verfügbaren Entwicklungsressourcen in Ihrem LabVIEW-Paket: Toolkits, Informationsquellen, angebotene Kurse und Möglichkeiten zur technischen Unterstützung mit LabVIEW.



Erforschen Sie Instrumentensteuerung mit LabVIEW.

Erklärt, wie Sie LabVIEWs flexible Schnittstelle für Anwendungen zur Instrumentensteuerung verwenden können.



Überwachen Sie eine langsame Datenerfassungsanwendung.

Demonstriert, wie LabVIEW mit Datenerfassungsgeräten zur Überwachung, Sammlung und Analyse von Daten verwendet werden kann.



Entdecken Sie LabVIEW für Echtzeit-Datenerfassungsanwendungen.

Demonstriert, wie gut LabVIEW mit Hochgeschwindigkeits-Datenerfassungsanwendungen zusammenarbeitet, indem es Ihnen maximale Leistung und Flexibilität mit Ihren DAQ-Geräten bietet.



Mehr über National Instruments.

National Instruments wird kurz vorgestellt, und Sie erfahren, wie Sie den Standort Ihrer nächstgelegenen National Instruments-Niederlassung finden.



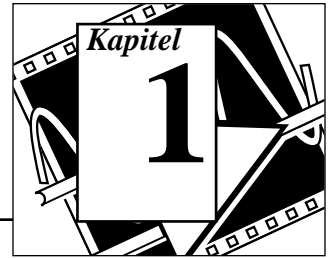
Erforschen Sie LabVIEW für Ihre eigenen Anwendungen.

Möchten Sie mehr erfahren? Diese Option erklärt, wie Sie mit dem LabVIEW-Demonstrationshandbuch einen tieferen Einblick in LabVIEW erhalten können.

Sie können jetzt einen Blick auf die Möglichkeiten des LabVIEW-Pakets werfen. Genauer über Programmieren in LabVIEW erfahren Sie in Kapitel 2, "Einführung in LabVIEW", in diesem Handbuch.

Wenn Sie weitere vollständige LabVIEW-Demonstrationen erkunden möchten, öffnen Sie die demos.lib-Bibliothek und wählen Sie ein Demonstrations-VI aus. Klicken Sie anschließend auf "Ausführen", um das VI einzusehen.

Einführung in LabVIEW



In diesem Kapitel wird beschrieben, was LabVIEW ist, was ein virtuelles Instrument (VI) ist, wie die LabVIEW-Umgebung (Fenster, Menüs, Paletten und Tools) einzusetzen ist und wie VIs erstellt, bedient und bearbeitet werden.

Bevor Sie damit beginnen, an den Zielen in diesem Kapitel zu arbeiten, sollten Sie auf die Option “Erforschen Sie LabVIEW für Ihre eigenen Anwendungen” im LabVIEW Demo-VI klicken.

Da es sich bei LabVIEW um ein Programmentwicklungssystem mit einer Vielzahl von Leistungsmerkmalen handelt, würde es den Rahmen dieses Demonstrations-/Lernhandbuchs sprengen, für alle denkbar möglichen Programmierprobleme Lösungen aufzuzeigen. Stattdessen dient dieses Demonstrations-/Lernhandbuchs dazu, Ihnen die theoretischen Grundlagen von LabVIEW zu erläutern; es enthält Übungen, anhand deren Sie mit den LabVIEW-Programmierwerkzeugen vertraut gemacht werden und führt Sie kurz durch den praktischen Einsatz der LabVIEW-Leistungsmerkmale in ihrer Anwendung auf konkrete Programmieraufgaben.

Falls Sie nach dem Durcharbeiten dieses Handbuchs an weiterer Schulung interessiert sind, bietet National Instruments praxisnahe LabVIEW-Kurse, in denen Sie in kurzer Zeit mit LabVIEW vertraut gemacht werden und erfolgreiche Anwendungen entwickeln lernen.

Im umfassenden LabVIEW Basics-Kurs lernen Sie nicht nur die Grundlagen von LabVIEW, Sie haben auch die Gelegenheit, praktische Erfahrungen bei der Entwicklung von Anwendungen zur Datenerfassung (für Windows, Macintosh und Sun) und Gerätesteuerung zu sammeln. Im darauf aufbauenden LabVIEW Advanced-Kurs erfahren Sie, wie die Leistungsfähigkeit und Effizienz von LabVIEW-Anwendungen gesteigert werden kann, während Sie mit den fortgeschrittenen Leistungsmerkmalen von LabVIEW vertraut gemacht werden.

Einen umfassenden Katalog der Kursinhalte, -gebühren und -termine erhalten Sie auf Wunsch von National Instruments. Die entsprechenden

Kontaktinformationen finden Sie auf der Adressenseite auf der inneren Umschlagsseite vorne in diesem Handbuch.

Kapitelinformationen

Jedes Kapitel beginnt mit einem dem folgenden ähnlichen Abschnitt, in dem die Lernziele des Kapitels aufgeführt sind.

Sie werden lernen:

- was LabVIEW ist
- was ein virtuelles Instrument (VI) ist
- wie man die LabVIEW-Umgebung (Fenster und Paletten) einsetzt
- wie man VIs erstellt
- wie man VIs bearbeitet
- wie man VIs verwendet

Was ist LabVIEW?

LabVIEW ist eine Anwendung zur Programmentwicklung, ganz ähnlich verschiedenen kommerziellen C- oder BASIC-Entwicklungssystemen oder LabWindows von National Instruments. LabVIEW unterscheidet sich jedoch in einer wichtigen Hinsicht von diesen Anwendungen. Andere Programmiersysteme setzen *textbasierende* Sprachen zur Erstellung von Codezeilen ein, wohingegen LabVIEW eine *grafische* Programmiersprache, G, zur Erstellung von Programmen in der Form eines Blockdiagramms verwendet.

Sie können LabVIEW auch mit geringer Programmiererfahrung einsetzen. LabVIEW verwendet Terminologie, Symbole und Konzepte, die Wissenschaftlern und Ingenieuren vertraut sind; die Anwendung stützt sich zur Beschreibung von Programmiervorgängen nicht auf eine Textsprache, sondern auf grafische Symbole.

LabVIEW verfügt über ausführliche Bibliotheken von Funktionen und Subroutinen für die meisten Programmieraufgaben. Für Windows, Macintosh und Sun enthält LabVIEW anwendungsspezifische Bibliotheken zur Datenerfassung und VXI-Gerätesteuerung. LabVIEW enthält auch anwendungsspezifische Bibliotheken für GPIB und serielle Gerätesteuerung, Datenanalyse, -darstellung und

-speicherung. LabVIEW enthält herkömmliche Programmierungswerkzeuge, die Ihnen ermöglichen, Breakpoints zu setzen, die Ausführung als Animation darzustellen, um den Datenfluß durch das Programm zu veranschaulichen und zur Erleichterung von Debugging und Programmentwicklung das Programm im Einzelschrittmodus auszuführen.

Wie funktioniert LabVIEW?

LabVIEW enthält Bibliotheken von speziell zur Gerätesteuerung konstruierten Funktionen und Entwicklungswerkzeugen. Für Windows, Macintosh und Sun enthält LabVIEW auch Bibliotheken von Funktionen und Entwicklungswerkzeugen zur Datenerfassung. LabVIEW-Programme werden *virtuelle Instrumente (VIs)* genannt, weil ihr Aussehen und ihre Funktion tatsächlichen Instrumenten ähnelt. Sie sind jedoch analog zu Funktionen aus herkömmlichen Programmiersprachen. VIs besitzen sowohl eine interaktive Benutzerschnittstelle als auch eine Entsprechung im Quellcode, und sie akzeptieren Parameter von VIs auf höherer Ebene. Im folgenden führen wir die Beschreibungen dieser drei Leistungsmerkmale von VIs auf.

- VIs verfügen über eine interaktive Benutzerschnittstelle, *Front Panel (Bedieneroberfläche)* genannt, weil sie die Bedieneroberfläche eines realen Geräts simuliert. Das Front Panel kann Knöpfe, Tasten, Diagramme und andere Controls und Indicators aufweisen. Daten werden über Tastatur und Maus eingegeben und die Ergebnisse auf dem Computerbildschirm angezeigt.
- Das VI erhält Befehle von einem *Block Diagram (Blockdiagramm)*, welches in G konstruiert wird. Das Blockdiagramm stellt die grafische Lösung eines Programmierproblems dar und enthält gleichzeitig den Quellcode für das VI.
- VIs sind hierarchisch und modular aufgebaut. Sie können als Programme auf höchster Ebene oder als Unterprogramme innerhalb anderer Programme bzw. Unterprogramme eingesetzt werden. Ein VI, das sich innerhalb eines anderen VIs befindet, wird *SubVI* genannt. Die Icons und Connectors eines VIs funktionieren wie eine grafische Parameterliste, so daß es als SubVI von anderen VIs Daten empfangen kann.

Mit diesen Leistungsmerkmalen fördert und hält LabVIEW am Konzept des *modularen Programmierens* fest. Man unterteilt eine Anwendung in eine Serie von Aufgaben, die wiederum unterteilt werden kann, bis aus einer komplizierten Anwendung eine Serie einfacher Teilaufgaben wird. Zur Erledigung jeder Teilaufgabe erstellt man ein VI und verbindet daraufhin diese VIs zur Erledigung der übergeordneten Aufgabe in einem weiteren Blockdiagramm. Das VI der höchsten Ebene enthält schließlich eine Ansammlung von SubVIs, die Anwendungsfunktionen darstellen.

Da jedes SubVI unabhängig vom Rest der Anwendung ausgeführt werden kann, wird das Debugging stark erleichtert. SubVIs können auf unteren Ebenen von mehreren Anwendungen aufgerufen werden, ob daß man möglichst universell verwendbare SubVIs entwickeln sollte.

Tools-Palette

LabVIEW bedient sich einer bewegbaren **Tools**-Palette, mit der man VIs bearbeiten und debuggen kann. Mittels der Taste <Tab> können Sie durch die am häufigsten verwendeten Tools in der Palette blättern. Falls Sie die **Tools**-Palette ausgeblendet haben, kann sie durch Auswahl von **Windows»Show Tools Palette** erneut angezeigt werden. Die **Tools**-Palette ist im folgenden abgebildet.



Operating Tool – zum Platzieren von Objekten aus den **Controls**- und **Functions**- Paletten in das Front Panel und das Blockdiagramm



Positioning Tool – zur Positionierung, Größenänderung und Auswahl von Objekten



Labeling Tool – zur Bearbeitung von Text und Erstellung von leeren Etiketten



Wiring Tool – zur Verdrahtung von Objekten im Blockdiagramm



Object Pop-up Menü Tool – zum Aufrufen eines Pop-up Menüs für ein Objekt



Scroll Tool – zum Blättern ohne Verwendung der Rollbalken



Breakpoint Tool – zur Einstellung von Breakpoints innerhalb VIs, Funktionen, Schleifen, Sequences und Cases



Probe Tool – Erlaubt das Einsehen der aktuellen Daten in Diagramm.



Color Copy Tool – zum Kopieren von Farben, um diese mit dem Color Tool einfügen zu können



Color Tool – zur Einstellung von Vorder- und Hintergrundfarben

Controls-Palette

Die **Controls**-Palette ist eine grafische, schwebende Palette, die beim Start von LabVIEW automatisch geöffnet wird. Mit dieser Palette platziert man Controls und Indicators in das Front Panel eines VIs. Alle Symbole der höchsten Ebene enthalten Unterpalletten. Ist die **Controls**-Palette nicht sichtbar, kann sie über den Befehl **Windows»Show Controls Palette** im Menü des Front Panels eingeblendet werden. Sie können auch eine beliebige freie Stelle im Front Panel mit der rechten Maustaste anklicken (Pop-up), um eine temporäre Kopie der **Controls**-Palette zu erhalten.

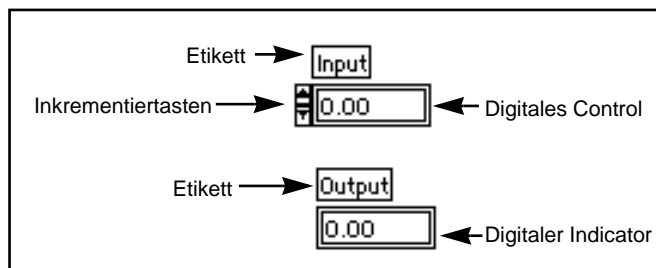
In der folgenden Abbildung ist die höchste Ebene der **Controls**-Palette dargestellt.



Controls und Indicators

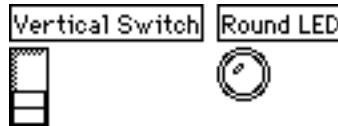
Numerische Controls und Indicators

Numerische Controls dienen zur Eingabe und numerische Indikatoren der Anzeige von numerischen Größen. Die beiden am häufigsten verwendeten numerischen Objekte sind das *digitale Control* und der *digitale Indicator*.



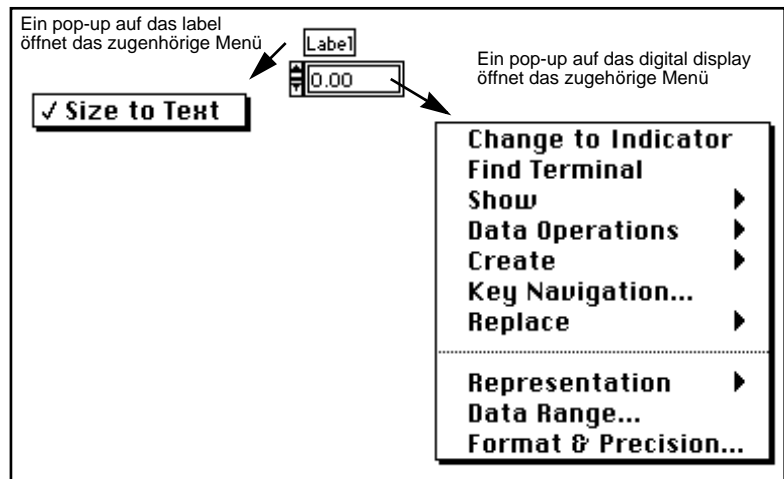
Boolesche Controls und Indicators

Boolesche Controls und Indicators werden zur Eingabe und zum Anzeigen von Booleschen Werten (TRUE/FALSE) verwendet. Boolesche Objekte simulieren Schalter, Tasten und LEDs. Die am häufigsten verwendeten Booleschen Controls und Indicators sind der *vertikale Schalter* und die *runde LED*.



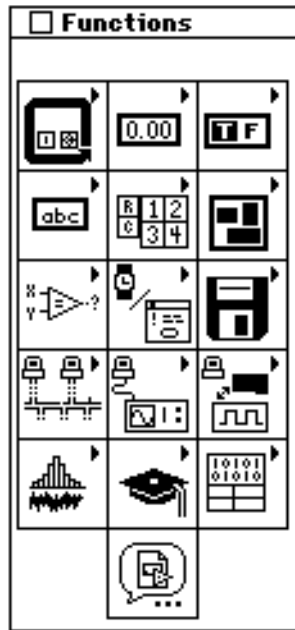
Konfiguration von Controls und Indicators

Fast alle Controls und Indicators können über Optionen aus ihren entsprechenden Pop-up Menüs konfiguriert werden. *Beim Durchführen eines Pop-ups auf einzelnen Komponenten von Controls und Indicators werden die Menüs zur Anpassung dieser Komponenten angezeigt.* Eine einfache Methode des Zugriffs auf das Pop-up Menü besteht darin, mit dem Object Pop-up Menü Tool (siehe Abbildung links) ein beliebiges Objekt anzuklicken, das über ein Pop-up Menü verfügt. In der folgenden Abbildung wird dieses Anzeigeverfahren für ein digitales Control dargestellt.



Functions-Palette

Die **Functions**-Palette ist eine grafische, schwebende Palette, die beim Wechsel zum Blockdiagramm automatisch geöffnet wird. Mit dieser Palette platziert man Nodes (Constants, Indicators, VIs usw.) in das Blockdiagramm eines VIs. Alle Symbole der höchsten Ebene enthalten Unterpaletten. Ist die **Functions**-Palette nicht sichtbar, kann sie über den Befehl **Windows»Show Functions Palette** im Menü des Blockdiagramms eingeblendet werden. Sie können auch eine beliebige freie Stelle im Blockdiagramm anklicken, um eine temporäre Kopie der **Functions**-Palette zu erhalten. Die folgende Abbildung zeigt die höchste Ebene der **Functions**-Palette.



Erstellung eines VIs

ZIEL

Die Erstellung eines VIs, das die Erfassung einer Temperaturmessung simuliert.

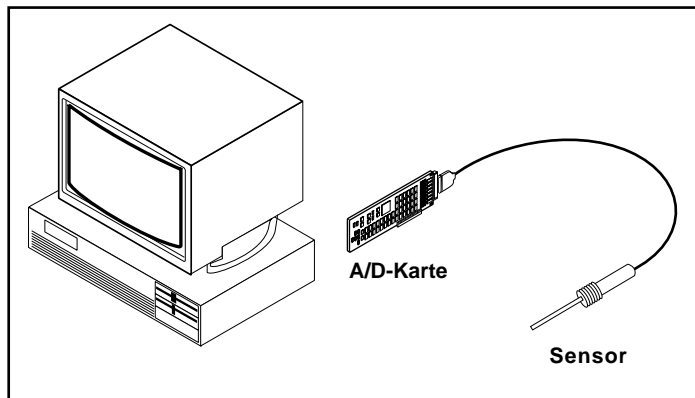
Vergewissern Sie sich, daß Sie vor Beginn dieser Übung auf die Option “Erforschen Sie LabVIEW für Ihre eigenen Anwendungen” im LabVIEW Demo-VI geklickt haben.

Dazu verwenden Sie das VI “Demo Voltage Read” zur Spannungsmessung und multiplizieren die Messung mit 100.0, um die Spannung in eine in Fahrenheit angegebene Temperatur umzuwandeln.

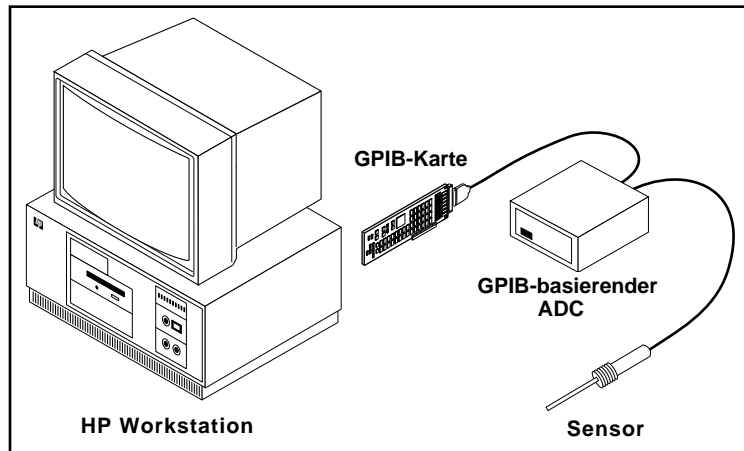
Nehmen Sie an, Sie verfügen über einen Meßwertwandler oder Sensor, der Temperatur in Spannung umwandelt.



(Windows, Macintosh und Sun) Der Sensor ist an eine Analog-Digital-Wandler-Karte angeschlossen, die Spannung in digitale Daten umwandelt (siehe folgende Abbildung).



(HP-UX) Der Sensor kann auch an einen Analog-Digital-Wandler angeschlossen sein, der über GPIB mit dem Rechner verbunden ist (siehe folgende Abbildung). Auf diese Weise wird ebenfalls die Umwandlung von Spannung in digitale Daten erreicht.



Front Panel

1. Öffnen Sie mit dem Befehl **File»New** oder durch Auswahl der Taste New VI im Dialogfeld ein neues Front Panel.



Hinweis: *Ist die Controls-Palette ausgeblendet, wählen Sie **Windows»Show Controls Palette**, um die Palette anzuzeigen. Alternativ dazu kann die Controls-Palette durch Anklicken einer freien Stelle im Front Panel eingeblendet werden. Klicken Sie dazu auf die rechte Maustaste (<command>-Klick auf Macintosh).*

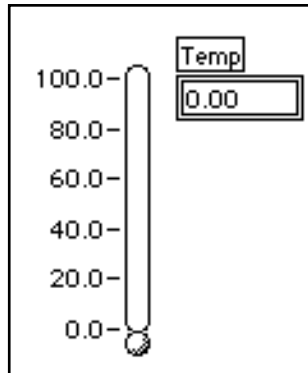


Hinweis: *Klicken Sie, ohne Text eingegeben zu haben, außerhalb eines Textfelds, verschwindet das Etikett. Sie können das Etikett erneut anzeigen, indem Sie auf dem Control einen Pop-up durchführen und den Befehl **Show»Label** wählen.*

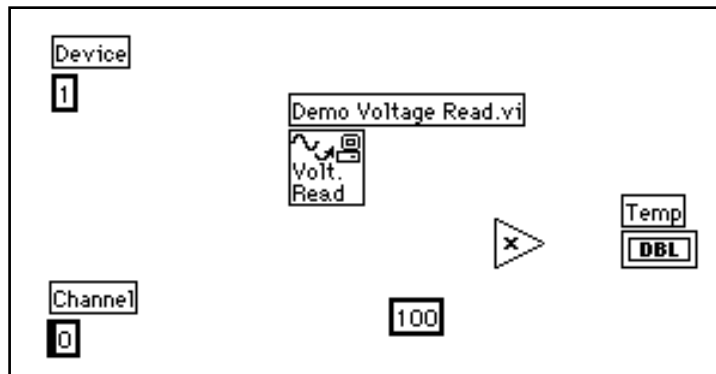


2. Wählen Sie einen Thermometer-Indicator aus **Controls»Numeric**, und plazieren Sie ihn im Front Panel, indem Sie den Indicator auf das Panel ziehen.
3. Geben Sie in das Etiketten-Textfeld Temp ein und klicken Sie im Toolbar auf die Enter-Taste.
4. Ändern Sie den Maßstab des Thermometer-Controls, um die Temperatur zwischen 0.0 und 100.0 anzuzeigen.
 - a. Doppelklicken Sie mit dem Labeling Tool auf die 10.0 in der Thermometerskala, um den Wert hervorzuheben.

- b. Geben Sie in der Skala 100.0 ein, und klicken Sie mit der Maustaste an beliebiger Stelle außerhalb des Anzeigefensters. Die zwischen den beiden Werten liegenden Abstufungen werden von LabVIEW automatisch skaliert. Das Temperatur-Control sollte nun der folgenden Abbildung ähneln.



Blockdiagramm



- Öffnen Sie das Blockdiagramm durch die Wahl des Befehls **Windows»Show Diagram**. Wählen Sie die unten besprochenen Blockdiagramm-Objekte aus der **Functions**-Palette. Wählen Sie für jedes einzusetzende Objekt das Symbol und dann das Objekt aus der höchsten Palettenebene, oder wählen Sie das Objekt aus der entsprechenden Unterpalette. Wenn Sie den Mauscursor auf

dem Blockdiagramm positionieren, zeigt LabVIEW einen Umriß des Objekts an.



Hinweis: *Ist die Functions-Palette nicht sichtbar, wählen Sie Windows»Show Functions Palette, um die Palette einzublenden. Alternativ dazu kann die Functions-Palette eingeblendet werden, indem Sie an einer offenen Stelle im Blockdiagramm einen Pop-up durchführen.*



Plazieren Sie jedes der folgenden Objekte in das Blockdiagramm.

Das VI "Demo Voltage Read" (**Functions»Tutorial**) simuliert die Spannungsmessung von einer Einsteckkarte für Datenerfassung.



Multiply Function (**Functions»Numeric**). In dieser Übung wird die vom VI "Demo Voltage Read" zurückgegebene Spannung mit 100.0 multipliziert.



Numeric Constant (**Functions»Numeric**). Es sind zwei numerische Konstanten erforderlich: eine für den Skalierfaktor 100 und eine für die Gerätekonstante. Geben Sie für die erste numerische Konstante 100.0 ein, wenn die Konstante zum ersten Mal im Blockdiagramm erscheint.

2. Erstellen Sie die zweite numerische Konstante mittels einer Abkürzung; so wird die Konstante automatisch erstellt und mit dem VI "Demo Voltage Read" verdrahtet.



a. Führen Sie mit dem Wiring Tool einen Pop-up auf dem mit Board ID gekennzeichneten Eingang des VIs "Demo Voltage Read" durch, und wählen Sie aus dem Pop-up Menü die Option **Create Constant**. Dadurch wird automatisch die numerische Konstante erstellt und mit dem VI "Demo Voltage Read" verdrahtet.

b. Geben Sie 1 ein, wenn die Konstante zum ersten Mal im Blockdiagramm erscheint; dadurch wird der Vorgabewert von null auf eins umgestellt.

Hinweis: *Beim Einsetzen des Wertes auf diese Weise ist ein Wechsel zum Labeling Tool nicht erforderlich, da sich der Cursor bereits an der richtigen Stelle befindet.*

c. Führen Sie einen Pop-up auf der Konstanten durch, und wählen Sie Show»Label. Ändern Sie das Vorgabe-Etikett (Board ID-Nummer) mittels des Labeling Tools in Device um.

In diesem Beispiel stellen die zwei numerischen Werte die Konstante 100.0 sowie das Gerät für die Multiplikationsfunktion dar.



3. Platzieren Sie eine String-Konstante (Functions»String) in Ihr Blockdiagramm.
4. Führen Sie mit dem Wiring Tool auf dem mit Channel bezeichneten Eingang einen Pop-up durch (unten links im VI “Demo Voltage Read”) und wählen Sie aus dem Pop-up Menü die Option **Create Constant**. Damit wird automatisch eine String-Konstante erstellt und mit dem VI “Demo Voltage Read” verdrahtet.
5. Geben Sie 0 ein, wenn die Konstante zum ersten Mal im Blockdiagramm erscheint. Führen Sie einen Pop-up auf der Konstanten durch, und wählen Sie Show»Label. In diesem Fall erscheint Channel als Vorgabe-Etikett, d.h. ein Ändern des Etiketts ist nicht erforderlich.

In diesem Beispiel kann die String-Konstante zur Darstellung des Kanalnamens verwendet werden.



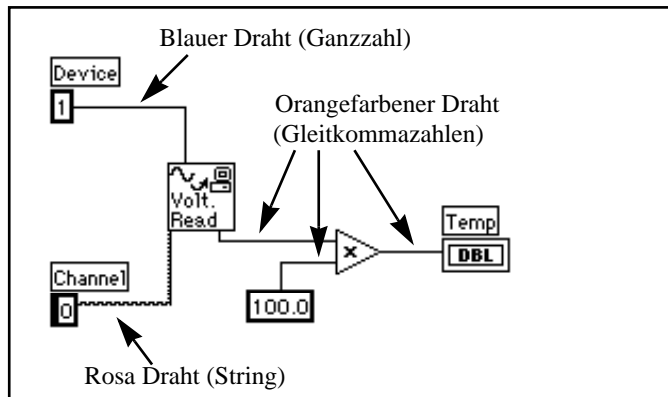
Hinweis: *Controls, Constants und Indicators können mit den meisten Funktionen erstellt und verdrahtet werden. Wenn diese Optionen für eine bestimmte Funktion nicht verfügbar sind, erscheinen die Optionen Create Control, Create Constant und Create Indicator im Pop-up Menü als deaktiviert. Näheres zu diesem Leistungsmerkmal finden Sie im Abschnitt Erstellen und Verdrahten von Controls, Constants und Indicators weiter unten in diesem Kapitel.*

Sie sollten jetzt alle in der folgenden Abbildung gezeigten Objekte auf Ihr Blockdiagramm heruntergezogen haben.



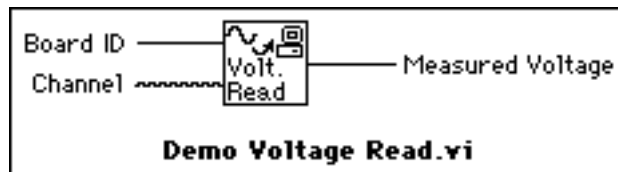
6. Verdrahten Sie die verbleibenden Objekte unter Verwendung des Wiring Tools miteinander, wie im Abschnitt *Verdrahtungstechniken* weiter unten in diesem Kapitel beschrieben.

Hinweis: Zum Verschieben von Objekten im Blockdiagramm klicken Sie auf das Positioning Tool in der Palette Tools.



Je nach Art der von ihnen übermittelten Daten werden Drähte in LabVIEW mit der entsprechenden Farbe gekennzeichnet. In blauen Drähten werden Ganzzahlen, in orangefarbenen Drähten Gleitkommazahlen, in grünen Drähten Boolesche Werte und in rosa Drähten Strings übermittelt.

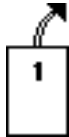
Mit dem Befehl **Help»Show Help** können Sie das Hilfe-Fenster aktivieren. Wenn Sie ein beliebiges Editier-Tool auf einem Node platzieren, werden die Ein- bzw. Ausgänge der entsprechenden Funktion im Hilfe-Fenster angezeigt. Wenn Sie ein Editing Tool auf einem VI-Symbol platzieren, werden die Verdrahtungsterminals sowohl im Blockdiagramm als auch im Hilfe-Fenster hervorgehoben. Wenn Sie mit der Verdrahtung Ihrer eigenen Diagramme beginnen, erleichtert Ihnen diese blinkende Markierung das Verbinden der Ein- bzw. Ausgänge mit den richtigen Terminals.



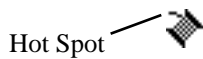
Das VI "Demo Voltage Read" simuliert die Spannungsmessung am Kanal 0 einer Einsteckkarte und liefert somit künstliche Daten an den

Ausgang Measured Voltage. Diese Daten stellen die echte Temperatur, dividiert durch 100, dar. Das VI multipliziert die Spannung dann mit 100.0, um sie in eine in Fahrenheit angegebene Temperatur umzuwandeln.

Verdrahtungstechniken



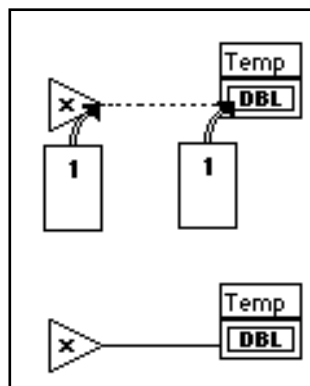
In den in diesem Abschnitt gezeigten Abbildungen zur Verdrahtung gibt der Pfeil am Ende des Maus-Symbols an, auf welche Stelle zu klicken ist; die auf der Maustaste abgebildete Ziffer gibt die erforderliche Anzahl von Mausklicks an.



Der *Hot Spot* des Tools ist die Spitze des abwickelnden Drahtsegments.

Um ein Terminal mit einem anderen zu verdrahten, klicken Sie mit dem Wiring Tool auf das erste Terminal, bewegen das Tool zum zweiten Terminal und klicken darauf. Es spielt keine Rolle, bei welchem Terminal Sie beginnen.

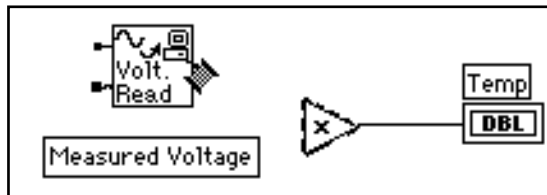
Wenn sich das Wiring Tool über einem Terminal befindet, blinkt der Terminalbereich, um anzuzeigen, daß der Draht durch ein Anklicken mit dem Terminal verbunden wird. Beim Bewegen des Wiring Tools von einem Terminal zum anderen ist es *nicht* notwendig, die Maustaste gedrückt zu halten. Sie können einen Draht einmal biegen, indem Sie die Maus im rechten Winkel zur aktuellen Richtung bewegen. Um den Draht an weiteren Stellen zu biegen, klicken Sie auf die Maustaste. Um die Richtung des Drahtes zu ändern, drücken Sie die Leertaste. Klicken Sie mit der Maustaste, um den Draht *zu befestigen*, und bewegen Sie die Maus im rechten Winkel.



Tip Strips



Wenn Sie das Wiring Tool auf dem Terminal eines Nodes plazieren, erscheint für das Terminal ein Tip Strip. Tip Strips sind kleine, gelbe Textstreifen, in denen der Name des jeweiligen Terminals angegeben ist, wodurch die Verdrahtung von Terminals erleichtert wird. Die folgende Abbildung zeigt das Tip Strip (Measured Voltage), das erscheint, wenn Sie das Wiring Tool auf der Ausgabe des VIs "Demo Voltage Read" plazieren.



Hinweis: *Wenn Sie das Wiring Tool auf einem Node plazieren, werden von LabVIEW Drahtenden angezeigt, die Ein- bzw. Ausgang darstellen. Verfügt das Drahtende über einen Punkt, handelt es sich um eine Eingabe des Nodes.*

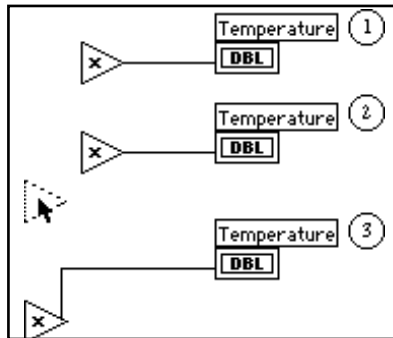
Darstellung von Terminals

Es ist wichtig, die richtigen Terminals einer Funktion zu verdrahten. Zur Erleichterung der Verdrahtung kann der Icon Connector eingeblendet werden. Führen Sie dazu auf der Funktion einen Pop-up durch und wählen Sie **Show Terminals**. Um erneut das Symbol anzuzeigen, führen Sie auf der Funktion erneut einen Pop-up durch und wählen wiederum **Show Terminals**.

Draht-Dehnung

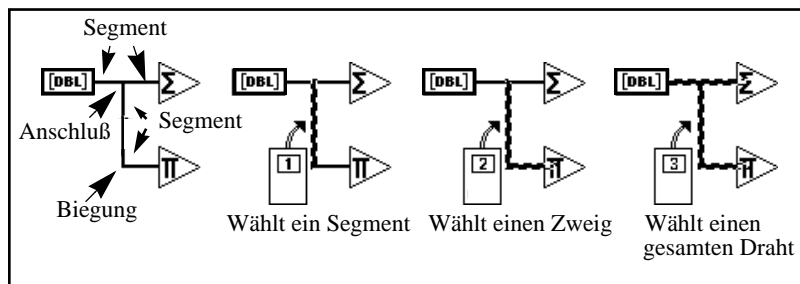


Verdrahtete Objekte können einzeln oder gruppenweise bewegt werden, indem die ausgewählten Objekte mit dem Positioning Tool an eine neue Stelle verschoben werden.



Auswählen und Löschen von Drähten

Es kann vorkommen, daß Sie Nodes versehentlich falsch verdrahten. Ist dies der Fall, wählen Sie den zu löschenden Draht und drücken anschließend <Delete>. Ein Drahtsegment ist ein einziges, horizontal oder vertikal verlegtes Drahtstück. Der Punkt, an dem drei oder vier Drahtsegmente zusammenlaufen, wird ein *Anschluß* genannt. Ein Drahtzweig enthält alle Drahtsegmente, die von einem Anschluß zum nächsten, von einem Terminal zum nächsten Anschluß oder von einem Terminal zum nächsten verlaufen, wenn sich zwischen ihnen keine Anschlüsse befinden. Die Auswahl eines Drahtsegments erfolgt durch Anklicken mit dem Positioning Tool. Durch Doppelklicken wählen Sie einen Zweig, und durch dreimaliges Klicken den gesamten Draht.



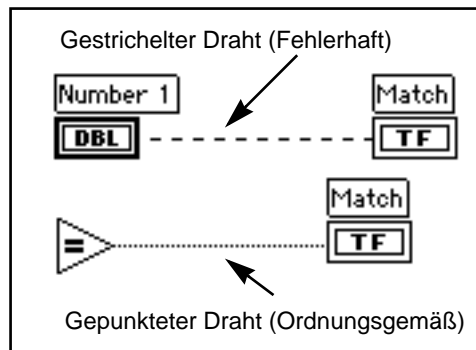
Fehlerhafte Drähte



Ein fehlerhafter Draht wird durch eine gestrichelte Linie dargestellt. Ein fehlerhafter Draht kann auf mehrere Weisen zustandekommen, z.B. die Verbindung von zwei Controls, oder das Verbinden einer Datenquelle mit einem Zielterminal, wenn die Datentypen nicht miteinander vereinbar sind (beispielsweise numerische und Boolesche Daten). Einen fehlerhaften Draht entfernt man durch Anklicken mit dem Positioning Tool und Drücken der Taste <Delete>. Durch die Wahl des Befehls **Edit>Remove Bad Wires** werden alle fehlerhaften Drähte im Blockdiagramm gelöscht. Dieses Verfahren bietet schnelle Abhilfe, falls sich das VI nicht ausführen läßt oder die Fehlermeldung `Signal has loose ends` ausgegeben wird.



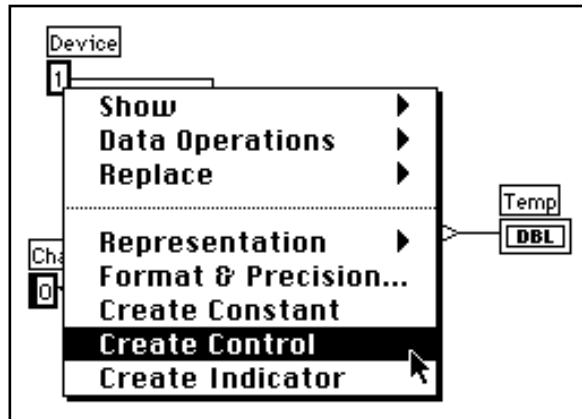
Hinweis: *Ein schwarzer, gestrichelter Draht sollte nicht mit einem gepunkteten Draht verwechselt werden. Wie aus der folgenden Abbildung ersichtlich ist, dient ein gepunkteter Draht zur Darstellung des Booleschen Datentyps.*



Erstellen und Verdrahten von Controls, Constants und Indicators

Für Terminals, die im Blockdiagramm zur Eingabe dienen, bietet LabVIEW zwei Leistungsmerkmale, die sich zur Erstellung und Verdrahtung von Controls oder Constants verwenden lassen. Sie können auf diese Optionen zugreifen, indem Sie auf dem Terminal einen Pop-up durchführen und den Befehl **Create Control** bzw. **Create Constant** wählen. Das korrekte Control bzw. der korrekte Konstanten-Typ wird von LabVIEW automatisch erstellt und mit dem

Terminaleingang verdrahtet. Die folgende Abbildung zeigt ein Beispiel eines Pop-up Menüs.



Für ein Terminal, das im Blockdiagramm als Ausgang dient, können Sie die Option **Create Indicator** wählen, um einen Indicator zu erstellen und mit dem Terminal zu verdrahten. Auf diese Option können Sie zugreifen, indem Sie auf dem Terminal einen Pop-up durchführen und den Befehl **Create Indicator** wählen. Der korrekte Indicator-Typ wird erstellt und mit dem Ausgang eines Terminals verdrahtet.

Hinweis: *Nachdem Sie Create Indicator gewählt haben, müssen Sie auf das Front Panel umschalten und den Indikator mit Hilfe des Positioning Tools auswählen und löschen*

Ausführen eines VIs

1. Aktivieren Sie das Front Panel, indem Sie auf die Titelleiste des Fensters klicken bzw. den Befehl **Windows»Show Panel** wählen. Unter Windows und Macintosh können Sie das Front Panel auch durch Klicken an einer beliebigen Stelle aktivieren.
2. Führen Sie das VI aus, indem Sie im Toolbar auf dem Front Panel die Taste "Run" anklicken.



Das VI muß jedesmal neu ausgeführt werden. Soll das VI mehrfach ausgeführt werden, muß die Taste "Continuous Run" angeklickt werden.



3. Klicken Sie im Toolbar auf die Taste “Continuous Run”.
4. Durch nochmaliges Anklicken der Taste “Continuous Run” wird diese Option deaktiviert. Das VI stoppt die Ausführung und wird beendet.

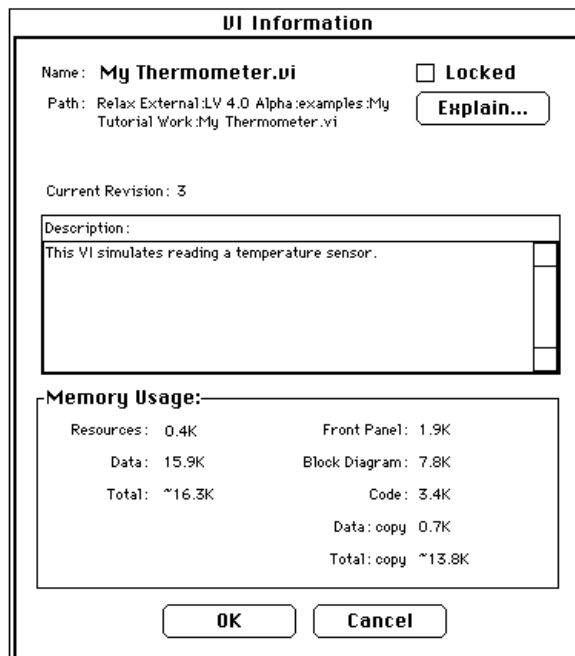


Hinweis: *Die Taste “Continuous Run” ist nicht die bevorzugte Methode zur Wiederholung eines Blockdiagramm-Codes. Stattdessen sollte eine Schleifen-Struktur verwendet werden. Näheres hierzu finden Sie in Kapitel 3, Schleifen und Charts, in diesem Demonstrations-/Lernhandbuch.*

Dokumentieren eines VIs

Durch die Wahl des Befehls **Windows»Show VI Info...** kann das VI dokumentiert werden. Geben Sie die Beschreibung des VIs in das Dialogfenster “VI Information” ein. Diese Beschreibung kann über den Befehl **Windows»Show VI Info...** erneut eingeblendet werden.

1. Dokumentieren Sie das VI. Wählen Sie den Befehl **Windows»Show VI Info...** Geben Sie wie in der folgenden Abbildung gezeigt die gewünschte Beschreibung für das VI ein und klicken Sie abschließend auf **OK**.

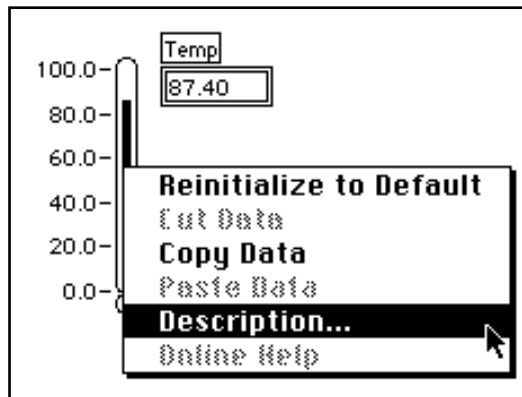


Sie können die Beschreibungen von Objekten im Front Panel (bzw. deren entsprechenden Terminals im Blockdiagramm) einblenden, indem Sie auf dem Objekt einen Pop-up durchführen und den Befehl **Description...** wählen.



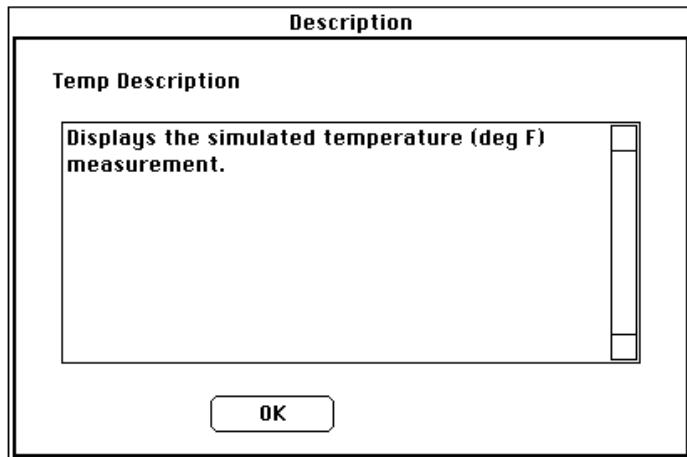
Hinweis: *Die Beschreibung kann während der Ausführung eines VIs nicht geändert werden.*

Die folgende Abbildung ist ein Beispiel eines Pop-up Menüs, das bei der Ausführung eines VIs erscheint. Während der Ausführung eines VIs kann die Beschreibung weder ergänzt noch geändert werden; bisher eingegebene Informationen können jedoch angezeigt werden.



2. Dokumentieren Sie den Thermometer-Indicator.
 - a. Führen Sie im Front Panel auf dem Thermometer-Indicator einen Pop-up durch, und wählen Sie den Befehl **Data Operations»Description...**

- b. Geben Sie wie in der folgenden Abbildung dargestellt die Beschreibung des Indicators ein, und klicken Sie auf **OK**.



3. Zeigen Sie erneut die soeben erstellte Beschreibung an, indem Sie auf dem Thermometer-Indicator einen Pop-up durchführen und den Befehl **Data Operations» Description...** wählen.

Sichern und Laden von VIs

Wie bei anderen Anwendungen auch können Sie ein VI in einem gewöhnlichen Verzeichnis als Datei speichern. Mit LabVIEW können auch mehrere VIs in einer Datei gespeichert werden, die eine *VI-Bibliothek* genannt wird. Die Bibliothek `tutorial.llb` ist ein Beispiel für eine derartige Datei.

Falls Sie mit Windows 3.1 arbeiten, sollten Sie Ihre VIs in VI-Bibliotheken speichern, da in der Bibliothek lange Dateinamen (bis zu 255 Zeichen) mit Groß- und Kleinschreibung verwendet werden können.

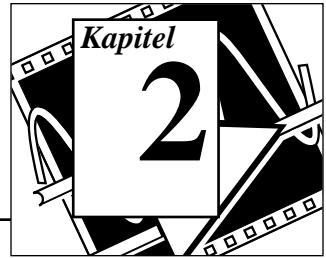
VI-Bibliotheken sollten nur verwendet werden, wenn Sie Ihre VIs in Windows 3.1 übertragen müssen. Das Speichern von VIs in Einzeldateien ist effektiver, da sie leichter kopiert, neu benannt und gelöscht werden können, als in VI-Bibliotheken.

Auch wenn Sie nicht Ihre eigenen VIs in VI-Bibliotheken speichern, sollten Sie mit deren Funktionsweise vertraut sein. Wir empfehlen

deshalb, alle im Verlauf dieses Demonstrations-/Lernhandbuchs erstellten VIs in VI-Bibliotheken zu speichern.

Speichern Sie Ihr VI in einer VI-Bibliothek.

1. Wählen Sie **File»Save As...** Wenn Sie mit UNIX arbeiten, geben Sie einen Bereich im Dateisystem an, für den Sie schreibbefugt sind, beispielsweise in Ihrem Home Directory.
2. Benennen Sie das VI und sichern Sie es in Ihrer neuen Bibliothek. Überprüfen Sie den Namen im Ringfeld oben im Dialogfenster, um sicherzustellen, daß er `mywork.llb` lautet. Ist dies nicht der Fall, klicken Sie auf `mywork.llb` in der Verzeichnisliste, um sicherzustellen, daß das VI an der richtigen Stelle gesichert wird.
 - a. Geben Sie im Dialogfenster `My Thermometer.vi` ein.
 - b. Klicken Sie auf **OK**.
3. Schließen Sie das VI mit dem Befehl **File»Close**.



Erstellung eines SubVIs

Was Sie lernen werden:

- Was ein SubVI ist.
- Wie ein Icon und ein Connector erstellt werden.
- Wie ein VI als SubVI verwendet wird.

Näheres zur Hierarchie

Einer der wichtigsten Aspekte zum Erstellen von LabVIEW-Anwendungen besteht in dem Verständnis und der Verwendung der hierarchischen Struktur des VIs. Wenn Sie ein VI erstellt haben, können Sie es als *SubVI* im Blockdiagramm eines höheren VIs verwenden. Ein SubVI ist daher analog zu einem Unterprogramm in C. Genauso wie es für die Anzahl der Unterprogramme, die Sie in einem C-Programm verwenden können, keine Begrenzung gibt, gibt es keine Begrenzung für die Anzahl der SubVIs, die Sie in einem LabVIEW-Programm verwenden können. Sie können auch ein Sub-IV innerhalb eines anderen Sub-IVs aufrufen.

Bei der Erstellung einer Anwendung beginnen Sie mit dem VI der höchsten Ebene und definieren die Eingaben und Ausgaben für die Anwendung. Dann konstruieren Sie die SubVIs, die die erforderlichen Operationen an den Daten durchführen, während sie das Blockdiagramm durchlaufen. Hat ein Blockdiagramm viele Icons, gruppieren Sie diese in ein VI tieferer Ebene, um die Einfachheit des Blockdiagramms beizubehalten. Dieser modulare Ansatz erleichtert das Debugging, Verständnis und die Pflege von Anwendungen.

Erstellung eines SubVIs

Ziel Für das VI “My Thermometer”, das Sie in Kapitel 1 erstellt haben, ein Icon und einen Connector zu erstellen und das VI als SubVI zu verwenden.

Um ein VI als SubVI verwenden zu können, müssen Sie ein Icon erstellen, das das VI auf dem Blockdiagramm eines anderen VIs darstellt, und anschließend ein Connector Pane (Anschluß-Plan), mit dem Sie die Eingaben und Ausgaben verknüpfen können.

Icon



Erstellen Sie das Icon, das das VI in dem Blockdiagramm anderer VIs darstellen soll. Ein Icon kann eine graphische Darstellung des Zwecks eines VIs oder eine Textbeschreibung des VIs oder seiner Terminals (Anschlüsse) sein.

1. Wenn Sie das VI "My Thermometer" geschlossen haben, öffnen Sie es durch Wählen von **File»Open...**, oder indem Sie auf die Taste "Open VI" im Dialogfeld drücken. Öffnen Sie **mywork.11b**. Unter Windows können Sie diese Bibliothek im temporären Verzeichnis oder in **windows/temp** finden. Unter Macintosh können Sie dieses Verzeichnis im **temporären Ordner** im **Systemordner** finden. Unter Unix befindet sich **mywork.11b** im Verzeichnis **/tmp**.



2. Wählen Sie My Thermometer.vi aus mywork.11b.
3. Rufen Sie den Icon Editor auf, indem Sie in der Icon-Belegung in der rechten oberen Ecke des Fensters für die Bedienoberfläche einen Pop-up durchführen und **Edit Icon** wählen. Sie können zur Bearbeitung des Icons auch einen Kurzbefehl verwenden und auf die Icon-Belegung doppelklicken.

Icon Editor Tools und Buttons

Die Tools links neben dem Bearbeitungsbereich führen folgende Funktionen durch:



Pencil Tool

Zeichnet und löscht Pixel für Pixel.



Line Tool

Zeichnet gerade Linien. Drücken Sie <Shift>, und ziehen Sie dann dieses Tool zum Einzeichnen von horizontalen, vertikalen und diagonalen Linien.



Dropper Tool

Kopiert die Vordergrundfarbe von einem Element in das Icon.



Fill Bucket Tool

Füllt einen umrissenen Bereich mit der Vordergrundfarbe aus.

	Rectangle Tool	Zeichnet eine rechteckige Umrandung in der Vordergrundfarbe ein. Doppelklicken Sie auf dieses Tool, um das Icon in der Vordergrundfarbe zu umranden.
	Filled Rectangle Tool	Zeichnet ein mit der Vordergrundfarbe umrandetes Rechteck ein und füllt es mit der Hintergrundfarbe aus. Doppelklicken Sie auf dieses Tool, um das Icon in der Vordergrundfarbe zu umranden und mit der Hintergrundfarbe auszufüllen.
	Select Tool	Wählt einen Bereich des Icons zum Verschieben, Klonen und anderen Änderungen.
	Text Tool	Dient zur Eingabe von Text in das Icon-Design.
	Vordergrund/Hintergrund	Zeigt die aktuellen Vordergrund- und Hintergrundfarben an. Klicken Sie auf Vordergrund oder Hintergrund, um eine Farbpalette anzuzeigen, aus der Sie neue Farben wählen können.

Die Tasten (Buttons) rechts neben dem Bearbeitungsbereich führen folgende Funktionen durch:

- Undo** Macht den letzten von Ihnen durchgeführten Befehl rückgängig.
- OK** Sichert Ihre Zeichnung als VI-Icon und kehrt zum Fenster der Bedienoberfläche zurück.
- Cancel** Kehrt zum Fenster der Bedienoberfläche zurück, ohne daß Änderungen gesichert werden.

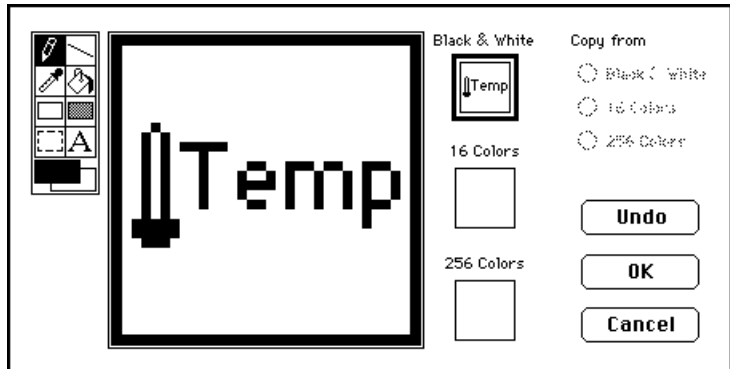
4. Löschen Sie das Standard-Icon.
 - a. Wählen Sie mit dem Select Tool den Innenbereich des Standard-Icons (siehe links).
 - b. Drücken Sie <Delete>, um den Innenbereich des Standard-Icons zu löschen.
5. Zeichnen Sie das Thermometer mit dem Pencil Tool ein.





- Erstellen Sie den Text mit dem Text Tool. Doppelklicken Sie zum Ändern der Schriftart auf das Text Tool. Experimentieren Sie mit dem Editor.

Ihr Icon sollte der folgenden Abbildung ähnlich sehen.



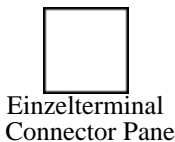
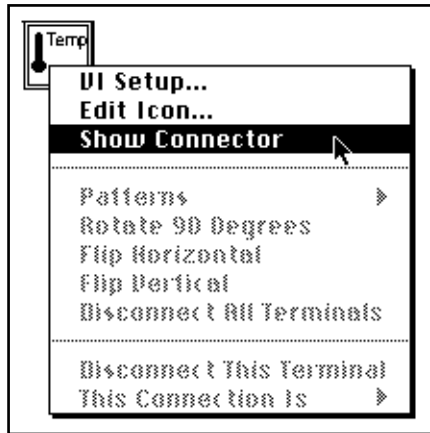
- Schließen Sie nach Fertigstellung Ihres Icons den Icon Editor, indem Sie auf **OK** klicken. Das neue Icon erscheint auf der Icon-Belegung in der oberen rechten Ecke des Fensters der Bedienoberfläche.

Connector

Sie können nun den Connector erstellen.

- Definieren Sie das Connector-Terminal-Muster, indem Sie in der Icon-Belegung auf der rechten oberen Ecke des Front Panels einen

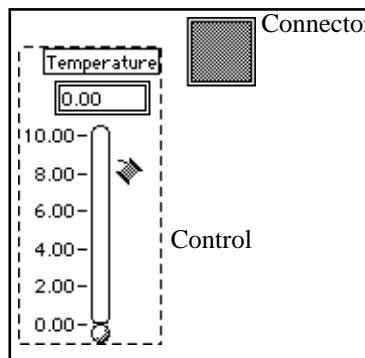
Pop-up durchführen und Show **Connector** wählen (siehe folgende Abbildung).



Da LabVIEW ein Terminalmuster aufgrund der Anzahl der Controls und Indicators auf dem Front Panel wählt, gibt es nur ein Terminal—den Thermometer Indicator.

2. Ordnen Sie das Terminal dem Thermometer zu.

- a. Klicken Sie im Terminal auf den Connector. Der Cursor wird automatisch zum Wiring Tool, und das Terminal wird schwarz.
- b. Klicken Sie auf den Thermometer Indicator. Der Indicator wird von einer sich bewegenden Linie umrandet (siehe folgende Abbildung).



Wenn Sie im Front Panel einen offenen Bereich anklicken, verschwindet die gestrichelte Linie und das gewählte Terminal verblaßt; dadurch wird angezeigt, daß Sie den Indicator dem betreffenden Terminal zugeordnet haben. Ist das Terminal weiß, haben Sie die Verbindung nicht richtig hergestellt. Wiederholen Sie, falls erforderlich, die vorhergehenden Schritte.

3. Sichern Sie das VI durch Wählen von **File»Save**. Wenn Sie am Macintosh zum Sichern von Dateien in der VI-Bibliothek das Dialogfenster für systemeigene Dateien verwenden, müssen Sie auf die Taste **Use LLBs** klicken, bevor Sie die VI-Bibliothek wählen.

Das VI ist nun fertig und kann als SubVI bei anderen VIs verwendet werden. Das Icon stellt das VI in dem Blockdiagramm des Aufruf-VIs dar. Der Connector (mit einem Terminal) gibt die Temperatur aus.



Hinweis: *Der Connector gibt die Eingaben und Ausgaben eines VIs an, wenn Sie es als SubVI verwenden. Vergessen Sie nicht, daß die Controls am Front Panel nur als Eingaben verwendet werden können; Indicators am Front Panel können nur als Ausgaben verwendet werden.*

4. Schließen Sie das VI durch Wählen von **File»Close**.

Verwendung eines VIs als SubVI

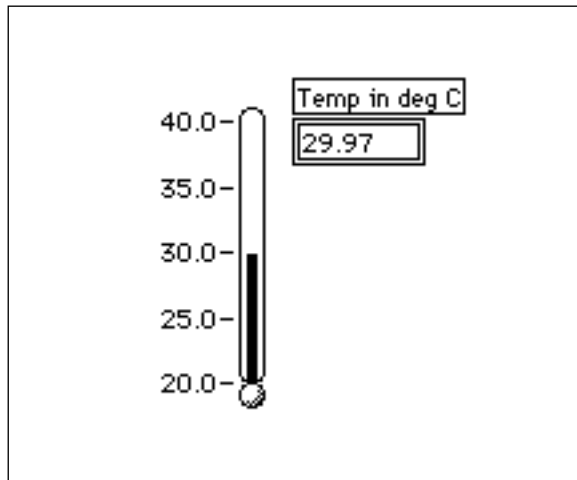
Sie können jedes VI, das ein Icon und einen Connector hat, als SubVI im Blockdiagramm eines anderen VIs verwenden. Sie wählen VIs, die als SubVIs verwendet werden sollen, mit **Functions»Select a VI...** Durch Wählen dieser Option wird ein Datei-Dialogfenster erstellt, aus dem Sie ein beliebiges VI im System wählen können. Wenn Sie ein VI öffnen, das kein Icon und keinen Connector hat, erscheint im Blockdiagramm des Aufruf-VIs ein leeres Viereck. Sie können diesen Node nicht verdrahten.

Ein SubVI ist analog zu einem Unterprogramm. Ein SubVI-Node (Icon/Connector) ist analog zu einem Unterprogrammaufruf. Der SubVI-Node ist nicht identisch mit dem SubVI an sich, genauso wie eine Unterprogramm-Aufrufanweisung in einem Programm nicht identisch mit dem Unterprogramm an sich ist. Ein Blockdiagramm mit mehreren identischen SubVI-Nodes ruft dasselbe SubVI mehrere Male auf.

ZIEL

Konstruktion eines VIs, das das VI “My Thermometer” als SubVI verwendet.

Das von Ihnen konstruierte VI “My Thermometer” gibt die Temperatur in Grad Fahrenheit aus. Sie werden den Wert ablesen und die Temperatur in Grad Celsius umsetzen.

Front Panel

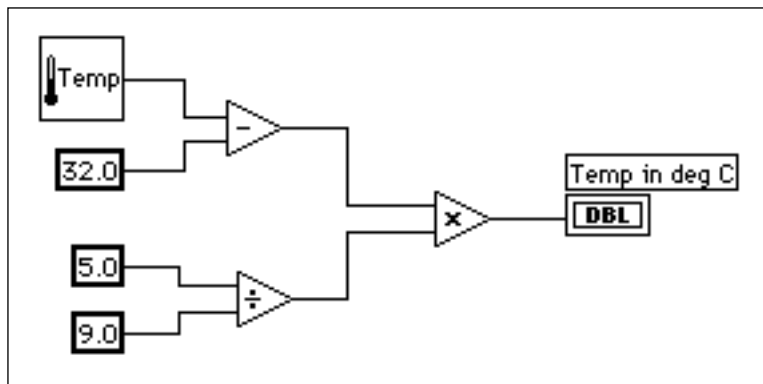
1. Öffnen Sie ein neues Front Panel durch Wählen von **File»New**, oder indem Sie auf die Taste “New VI” im Dialogfeld klicken.
2. Wählen Sie das Thermometer aus **Controls»Numeric**. Beschriften Sie es mit Temp in deg C. Falls Sie vor dem Eingeben des Etiketts außerhalb des Thermometers geklickt haben, verschwindet dieses. Zum Wiedereinblenden des Etiketts führen Sie einen Pop-up auf dem Thermometer durch; wählen Sie Show»Label, und geben Sie dann das Etikett ein.
3. Verändern Sie den Meßbereich des Thermometers entsprechend der Temperaturwerte. Doppelklicken Sie mit dem Operating Tool auf die untere Grenze, geben Sie 20 ein, und drücken Sie <Enter> auf dem Tastenfeld. Sie brauchen keine Dezimalzeichen und abschließenden Nullen einzugeben. Wenn Sie den Wert eingeben, fügt LabVIEW diese automatisch hinzu. Ändern Sie ebenso die obere Grenze des Thermometers auf 40,

und drücken Sie <Enter> auf dem Tastenfeld. LabVIEW stellt automatisch die Zwischenwerte ein.

Jedesmal wenn Sie ein neues Control oder einen neuen Indicator erstellen, erstellt LabVIEW das entsprechende Terminal im Blockdiagramm-Fenster. Die Terminalsymbole verweisen auf den Datentyp des Controls oder Indicators. Zum Beispiel stellt ein DBL-Terminal eine Gleitkommazahl mit double-precision dar. Ein TF-Terminal ist ein Boolesches Control; ein I16-Terminal stellt eine normale, 16-Bit-Ganzzahl dar, und ein ABC-Terminal stellt einen String dar.

Blockdiagramm

1. Wählen Sie **Windows»Show Diagram**.
2. Führen Sie in einem freien Bereich des Blockdiagramms einen Pop-up durch, und wählen Sie **Functions»Select a VI...** Ein Dialogfenster erscheint. Ermitteln und öffnen Sie die Bibliothek mywork.11b. (Unter Windows können Sie diese Bibliothek im temporären Verzeichnis oder in **windows/temp** finden. Unter Macintosh können Sie dieses Verzeichnis im **Systemordner/temporären Ordner** finden. Unter Unix befindet sich **mywork.11b** im Verzeichnis **/tmp**.) Doppelklicken Sie auf My Thermometer.vi, oder heben Sie es hervor, und klicken Sie im Dialogfenster auf **Open**. LabVIEW plaziert das VI "My Thermometer" im Blockdiagramm.
3. Fügen Sie die anderen Objekte, wie in der folgenden Abbildung gezeigt, dem Blockdiagramm hinzu.



1.23

Numerische Konstante (**Functions»Numeric**). Fügen Sie dem Blockdiagramm drei numerische Konstanten hinzu. Ordnen Sie den Konstanten mit dem Labeling Tool die Werte 32.0, 5.0 und 9.0 zu.

Hinweis: *Sie können anhand ihrer Farbe feststellen, um welche Konstante es sich bei der Zahl handelt. Blaue numerische Konstanten sind Ganzzahlen, und orangefarbige Konstanten sind Gleitkommazahlen. LabVIEW konvertiert Zahlen bei Bedarf automatisch in das richtige Format.*



Hinweis: *Beachten Sie, daß Sie auf den Funktionen einen Pop-up durchführen und Create Constant wählen können, um automatisch die korrekte Konstante zu erstellen und mit einer Funktion zu verdrahten.*



Die Funktion “Subtract” (**Functions»Numeric**) subtrahiert zur Konvertierung in Celsius 32 vom Fahrenheit-Wert.



Die Funktion “Divide” (**Functions»Numeric**) berechnet für die Temperaturkonvertierung den Wert 5/9.



Die Funktion “Multiply” (**Functions»Numeric**) gibt den Celsius-Wert nach der Konvertierung aus.

4. Verdrahten Sie die Diagrammobjekte wie in der vorhergehenden Blockdiagrammabbildung dargestellt.



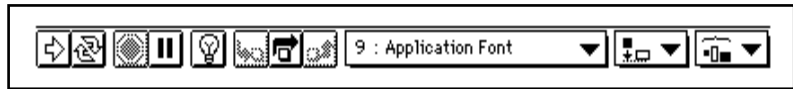
Hinweis: *Ein defekter Draht zwischen dem Thermometer-Icon und dem Terminal “Temp in deg C” weist eventuell darauf hin, daß Sie das SubVI Connector Terminal dem Front Panel Indicator falsch zugeordnet haben. Lesen Sie die Anweisungen im Abschnitt Erstellung eines SubVIs weiter vorn in diesem Kapitel. Wenn Sie das SubVI modifiziert haben, müssen Sie eventuell aus dem Icon Pop-up Menü Relink to SubVI wählen. Falls erforderlich, wählen Sie Edit»Remove Bad Wires.*

5. Kehren Sie zum Front Panel zurück, und klicken Sie in der Toolbar auf die Taste “Run”.

Blockdiagramm Toolbar (Werkzeugleiste)

Das Blockdiagramm-Fenster enthält zusätzliche Optionen, die nicht im Front Panel Toolbar enthalten sind.

Blockdiagramm Toolbar:



Das Blockdiagramm Toolbar enthält die folgenden Tasten, die Sie zum Debuggen von VIs verwenden können.



Taste “Hilite execute”–Zeigt den augenblicklichen Datenfluß an.



Taste “Step into”–Eintritt in Schleifen, SubVIs usw im Einzelschrittbetrieb.



Taste “Step over”–Beginnt mit Einzelschrittbetrieb, überspringt eine Schleife, ein SubVI usw.



Taste “Step out”–Beendet die Ausführung von Schleifen, VIs, Blockdiagrammen usw.

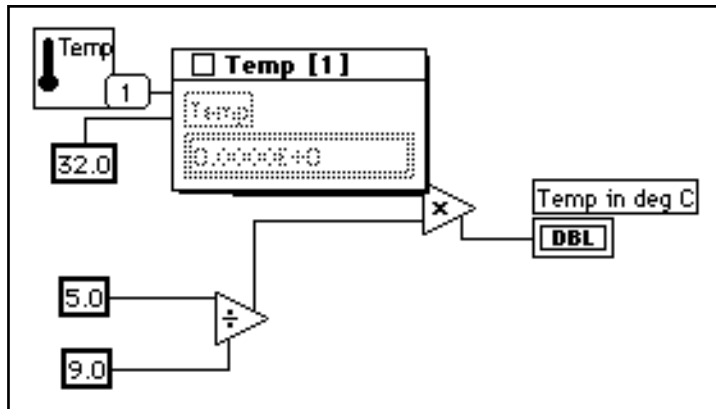
Einige Debugging-Techniken

Das Thermometer sollte einen Wert in dem gewählten Bereich anzeigen. Angenommen, Sie möchten zum Vergleich und Debugging den Fahrenheitwert sehen. LabVIEW verfügt über einige Tools, die Ihnen dabei behilflich sein können. In diesem Übungsbeispiel untersuchen Sie das Probe-Tool und die Leistungsmerkmale von Execution Highlighting (Visualisierung des Programmablaufs).

1. Wählen Sie **Windows»Show Diagram**.
2. Wählen Sie das Probe Tool aus der **Tools**-Palette. Klicken Sie mit dem Probe Tool auf den Temperaturwert (Draht), der aus dem SubVI “My Thermometer” austritt. Ein Sondenfenster mit dem Titel Temp 1 und ein gelbes Symbol mit der Sondenziffer werden

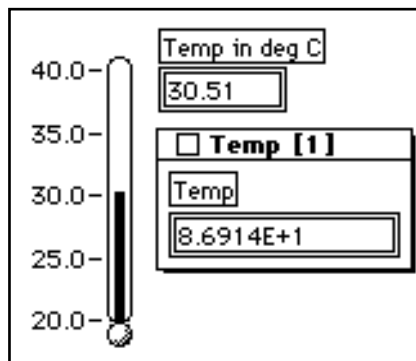


eingebildet (siehe folgende Abbildung). Das Sondenfenster erscheint außerdem auf dem Front Panel.



3. Kehren Sie zum Front Panel zurück. Verschieben Sie das Sondenfenster, so daß Sie sowohl die Sonden- als auch die Thermometerwerte sehen können (siehe folgende Abbildung). Führen Sie das VI aus. Die Temperatur erscheint in Grad Fahrenheit im Sondenfenster.

Hinweis: *Die auf Ihrem Bildschirm erscheinenden Temperaturwerte können sich von den Angaben auf dieser Abbildung unterscheiden. Zusätzliche Informationen entnehmen Sie bitte dem Abschnitt Numerische Konvertierung in Kapitel 3, Schleifen und Charts.*



- Schließen Sie das Probe-Fenster, indem Sie auf das Schließfeld in der oberen linken Ecke der Titelleiste des Sondenfensters klicken.

Das Überprüfen des Datenflusses in dem Blockdiagramm anhand des LabVIEW-Leistungsmerkmals Execution Highlighting ist eine weitere nützliche Debugging-Technik.

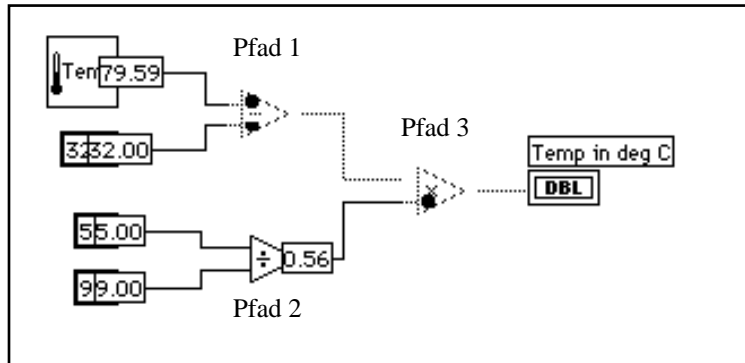
- Kehren Sie durch Wählen von **Windows»Show Diagram** zum Blockdiagramm des VIs zurück.



- Beginnen Sie mit dem Execution Highlighting, indem Sie im Toolbar auf die Taste "Hilite execute" (siehe links) klicken. Die Taste "Hilite execute" verwandelt sich in eine erleuchtete Glühbirne.



- Führen Sie das VI aus, und beobachten Sie, wie Execution Highlighting die Ausführung des VI-Blockdiagramms in Animation darstellt. Der Datenfluß durch das VI wird durch sich bewegende Punkte dargestellt. Beachten Sie außerdem, daß die Datenwerte auf den Drähten erscheinen und die zu dem Zeitpunkt in den Drähten vorhandenen Werte angezeigt werden (siehe folgendes Blockdiagramm), als ob Sie den Draht mit einer Meßsonde geprüft hätten.



Beachten Sie die Reihenfolge, in der die verschiedenen Nodes in LabVIEW ausgeführt werden. Bei konventionellen textorientierten Sprachen werden die Programmanweisungen in der Reihenfolge ihres Auftretens ausgeführt. LabVIEW verwendet jedoch *Datenflußprogrammierung*. Bei der Datenflußprogrammierung wird ein Node ausgeführt, wenn an allen Node-Eingaben Daten vorhanden sind, und nicht zwangsläufig von oben nach unten oder links nach rechts.

Die vorhergehende Abbildung stellt dar, daß LabVIEW zwischen Pfad 1 und Pfad 2 *Multitasking* durchführen kann, da keine Datenabhängigkeit besteht; nichts in Pfad 1 hängt von Daten in Pfad 2 ab, und nichts in Pfad 2 hängt von Daten in Pfad 1 ab. Pfad 3 muß jedoch zuletzt ausgeführt werden, da die Multiplikationsfunktion von den Daten der Subtraktions- und Divisionsfunktionen abhängt.

Execution Highlighting ist ein nützliches Tool zum Überprüfen der Datenflußbeschaffenheit von LabVIEW .

Sie können auch die Einzelschritt-Tasten verwenden, wenn Sie mehr Kontrolle über den Debugging-Prozeß haben möchten.



8. Beginnen Sie durch Klicken auf die Taste “Step over” im Toolbar mit dem Einzelschrittbetrieb. Durch Klicken auf diese Taste wird die erste Ausführungssequenz im VI angezeigt. Wenn LabVIEW diesen Teil der Sequenz abgeschlossen hat, visualisiert es das nächste Element, das im VI ausgeführt wird.



9. Greifen Sie durch Klicken auf die Taste “Step over” in der Toolbar auf die Funktion “Divide” zu. Durch Klicken auf diese Taste wird die Funktion “Divide” ausgeführt. Wenn LabVIEW diesen Teil der Sequenz abgeschlossen hat, visualisiert es das nächste Element, das im VI ausgeführt wird.



10. Beginnen Sie mit dem SubVI “My Thermometer”, indem Sie in der Toolbar auf die Taste “Step into” klicken. Durch Klicken auf diese Taste werden das Front Panel und das Blockdiagramm Ihres Thermometer-SubVIs geöffnet. Sie können nun das SubVI in Einzelschritten durchlaufen oder es ausführen.



11. Beenden Sie die Ausführung des Blockdiagramms, indem Sie in der Toolbar auf die Taste “Step out” klicken. Durch Klicken auf diese Taste werden alle verbleibenden Sequenzen im Blockdiagramm abgeschlossen. Wenn LabVIEW diesen Teil der Sequenz abgeschlossen hat, visualisiert es das nächste Element, das im VI ausgeführt wird. Sie können beim Klicken auf die Taste “Step out” auch die Maustaste gedrückt halten, um so auf ein Pop-up Menü zuzugreifen. In diesem Pop-up Menü können Sie wählen, wie weit das VI ausgeführt werden soll, bevor es unterbrochen

wird. Die folgende Abbildung zeigt die Optionen im Pop-up Menü der Taste "Step out" zur Beendigung der Ausführung.



12. Wählen Sie **File»Save as** und sichern Sie das VI in `mywork.llb`. Nennen Sie es `Using My Thermometer.vi`, und schließen Sie das VI.

Öffnen, Bedienen und Ändern von SubVIs

Sie können ein als SubVI verwendetes VI vom Blockdiagramm des Aufruf-VIs öffnen. Sie können das Fenster der Bedienoberfläche eines SubVIs öffnen, indem Sie auf das SubVI doppelklicken oder indem Sie **Project»This VI's SubVIs** wählen. Sie öffnen dann das Blockdiagramm, indem Sie **Windows»Show Diagram** wählen.

Jegliche von Ihnen an einem SubVI vorgenommenen Änderungen ändern nur die Version im Speicher, bis Sie das SubVI sichern. Beachten Sie, daß die Änderungen alle Aufrufe an das SubVI ändern, und nicht nur den Node, den Sie zum Öffnen des SubVIs verwendet haben.

Hierarchy-Fenster

Sie verwenden das Hierarchy-Fenster (**Project»Show VI Hierarchy**), zur Visualisierung von Abhängigkeitsbeziehungen von VIs, indem Sie Informationen zu VI-Aufrufbefehlen und SubVIs eingeben. Dieses Fenster enthält eine Toolbar, die Sie zur Konfigurierung verschiedener Einstellungstypen für angezeigte Elemente verwenden können. Die folgende Abbildung ist ein Beispiel für die VI Hierarchy-Toolbar.



Zum Zugreifen auf die folgenden Optionen können Sie die Tasten in der Toolbar des Hierarchy-Fensters oder das **VIEW**-Menü verwenden oder in einem freien Fensterbereich einen Pop-up durchführen.



- “Redraw” – Ordnet die Nodes nach aufeinanderfolgenden Operationen an Hierarchy-Nodes neu an, wenn Sie die Zeilenüberschneidungen minimieren und die Symmetrieästhetik maximieren müssen. Ist ein Fokus-Node vorhanden, durchblättern Sie dann das Fenster, bis das erste Wurzelsegment mit SubVIs sichtbar ist.



- “Switch to vertical layout” – Ordnet die Nodes von oben nach unten an, wobei die Wurzelsegmente oben platziert werden.



- “Switch to horizontal layout” – Ordnet die Nodes von Links nach rechts an, wobei die Wurzelsegmente links platziert werden.



- “Include/Exclude VIs in VI libraries” – Schaltet den Hierarchy-Graph zwischen Einschließen und Ausschließen von VIs in VI-Bibliotheken hin und her.



- “Include/Exclude global variables” – Schaltet den Hierarchy-Graph zwischen Einschließen und Ausschließen von globalen Variablen hin und her.



- “Include/Exclude typedefs” – Schaltet den Hierarchy-Graph zwischen Einschließen und Ausschließen von Typvereinbarungen hin und her.

Außerdem schließen das Menü **View** und die Pop-up Menüs die Optionen **Show all VIs** und **Full VI Path in Label** ein, auf die Sie in der Toolbar nicht zugreifen können.



Wenn Sie das Operating Tool auf Objekte im Hierarchy-Fenster verschieben, zeigt LabVIEW den Namen des VIs unterhalb des VI-Icons an.

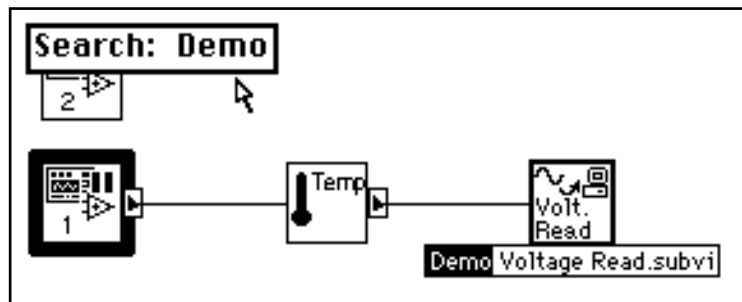
Verwenden Sie die Taste <Tab>, um zwischen den Positioning und Scroll Fenstertools hin- und herzuschalten. Diese Funktion ist zum Verschieben von Nodes vom Hierarchy-Fenster zum Blockdiagramm nützlich.

Durch Klicken auf den Node können Sie einen VI- oder SubVI-Node auf das Blockdiagramm ziehen oder auf die Zwischenablage kopieren. Durch <Shift>-Klicken auf einen VI- oder SubVI-Node können Sie mehrere Elemente zum Kopieren auf andere Blockdiagramme oder Front Panels wählen. Durch Doppelklicken auf einen VI- oder SubVI-Node wird das Front Panel des betreffenden Nodes geöffnet.

Jedes SubVIs enthaltende VI hat eine Pfeil-Taste neben dem VI, die Sie zum Ein- und Ausblenden der SubVIs des VIs verwenden können. Die SubVIs des VIs werden durch Klicken auf die Taste mit rotem Pfeil oder durch Doppelklicken auf das VI selbst geöffnet. Eine Taste mit schwarzem Pfeil bei einem VI-Node bedeutet, daß alle SubVIs angezeigt sind. Sie können auf einem VI- oder SubVI-Node auch durch Durchführen eines Pop-ups auf ein Menü mit verschiedenen Optionen zugreifen, z.B. Einblenden und Ausblenden von SubVIs, Öffnen des Front Panels des VIs oder SubVIs, Bearbeiten des VI-Icons usw.

Suche im Hierarchy-Fenster

Sie können außerdem momentan angezeigte Nodes im Hierarchy-Fenster nach Namen durchsuchen. Sie lösen die Suche aus, indem Sie den Namen des Nodes an einer beliebigen Stelle im Fenster eingeben. Wenn Sie den Text eingeben, erscheint ein Suchfenster, das den von Ihnen eingegebenen Text anzeigt und gleichzeitig die Hierarchie durchsucht. Die folgende Abbildung zeigt die Such-Hierarchie.

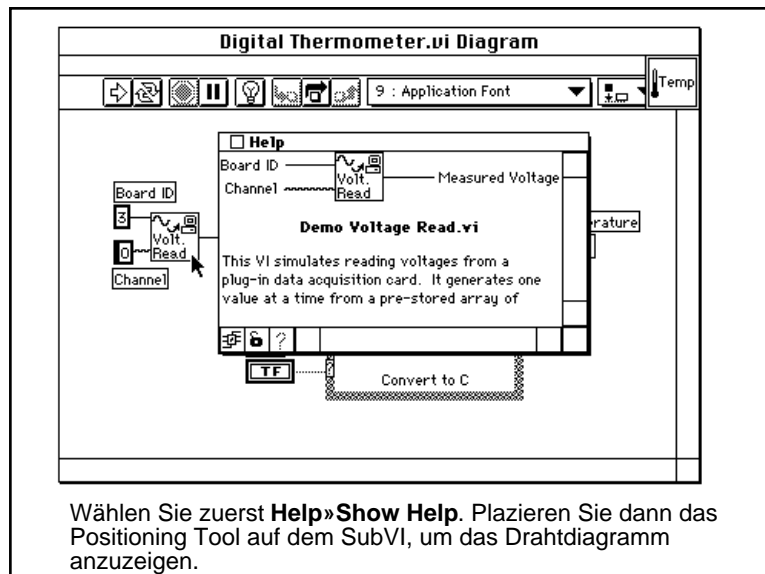


Wenn Sie den korrekten Node gefunden haben, können Sie <Enter> drücken, um nach dem nächsten mit dem Such-String übereinstimmenden Node zu suchen, oder Sie können <Shift-Enter> (Windows); <shift-return> (Macintosh); <Shift-Return> (Sun) oder <Shift-Enter> (HP-UX) drücken, um den vorherigen mit dem Such-String übereinstimmenden Node zu suchen.

Online-Hilfe für SubVI Nodes

Wenn Sie eines der Tools auf einem SubVI-Node plazieren, zeigt das Fenster "Help" das Icon für das SubVI einschließlich Drähten an jedem Terminal an. Die folgende Abbildung zeigt ein Beispiel für Online-Hilfe, das VI "Digital Thermometer" aus **Functions»Tutorial**. Ihr VI

“Thermometer” enthält außerdem den Text, den Sie im Dialogfenster “VI Information” eingegeben haben.



Einfache/Komplexe Hilfe-Ansicht

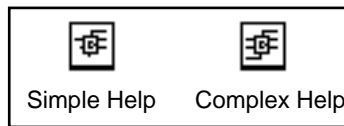
Sie können im Fenster “Help” angeben, ob Sie eine einfache oder komplexe Hilfeansicht für Blockdiagrammobjekte anzeigen möchten.



Hinweis: *Wenn Sie das Fenster “Help” öffnen, zeigt LabVIEW standardmäßig die einfache Hilfe-Ansicht an.*

In der einfachen Hilfe-Ansicht zeigt LabVIEW nur die für VIs und Funktionen erforderlichen und empfohlenen Eingaben an. In der komplexen Hilfe-Ansicht zeigt LabVIEW die für VIs und Funktionen erforderlichen, empfohlenen und optionalen Eingaben an. Außerdem wird der volle Pfadname eines VIs angezeigt. Zum Zugreifen auf die einfache Hilfe-Ansicht drücken Sie auf den Schalter “Simple/Complex Diagram Help”, oder wählen Sie **Help»Simple Diagram Help**. Die

folgende Abbildung zeigt beide Ansichten des Schalters “Simple/Complex Diagram Help”.

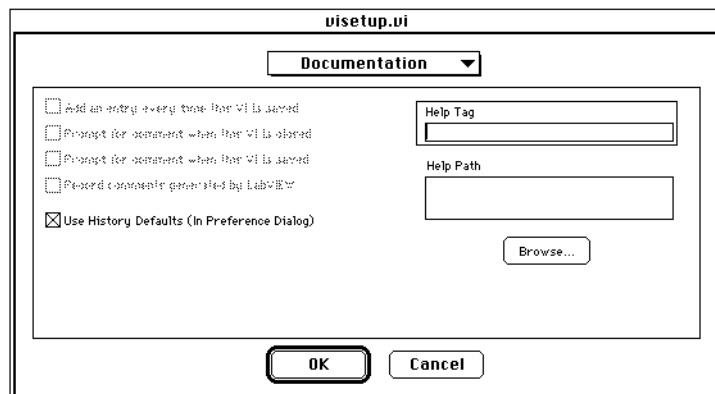


Im Fenster “Help” erscheinen erforderliche Eingaben fettgedruckt, empfohlene Eingaben normalgedruckt und optionale Eingaben grauedruckt. Wenn Sie Ihre eigenen VIs entwerfen, können Sie angeben, welche Eingaben erforderlich, empfohlen oder optional sind; öffnen Sie dazu durch Ausführen eines Pop-ups über einer Eingabe oder Ausgabe in dem Connector Pane das Untermenü **This Connection is**, und wählen Sie die korrekte Option.

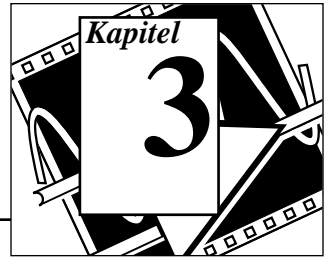


Verknüpfungen mit Online-Hilfe-Dateien

Sie können im Fenster “Help” auf die Taste “Online help” klicken, um auf die Online-Hilfe von LabVIEW sowie auf Hilfedateien, die Sie mit einem Hilfe-Kompilierer erstellt haben, zuzugreifen. Wenn Sie Ihre eigene Hilfedatei erstellen wollen, müssen Sie die Verknüpfung zur Hilfedatei angeben; klicken Sie dazu auf die Icon-Belegung, und wählen Sie **VI Setup...** Wenn das Dialogfenster “VI Setup” geöffnet wird, wählen Sie “Documentation” aus dem Ring-Control oben im Dialogfenster, und geben Sie dann den Pfad der Hilfedatei im Feld “Help Path” ein. Die folgende Abbildung zeigt die im Dialogfenster “VI Setup” erscheinenden Optionen.



Wählen Sie **Browse...**, um Hilfedatei und-thema mit Ihrem VI zu verknüpfen.



Schleifen und Charts

Sie werden lernen:

- Wie eine While-Schleife verwendet wird.
- Wie Daten in einem Chart angezeigt werden.
- Was ein Shift Register (Schieberegister) ist und wie es verwendet wird.
- Wie eine For-Schleife verwendet wird.

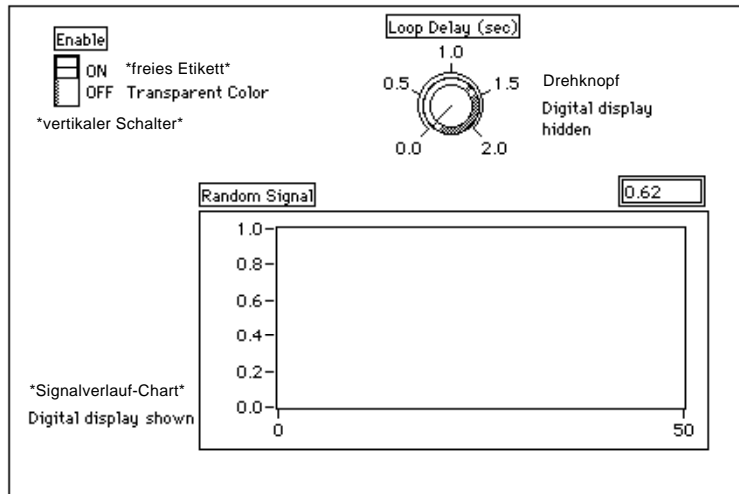
Der Datenfluß in einem VI wird von Strukturen (Structures) gesteuert. LabVIEW hat vier Strukturen: die While-Schleife, die For-Schleife, die Case-Struktur und die Sequence-Struktur. In diesem Kapitel werden die Strukturen der While-Schleife und For-Schleife sowie das Chart und das Shift Register eingeführt. Die Case- und Sequence-Strukturen werden in Kapitel 5, *Case- und Sequence-Strukturen und der Formula Node*, erläutert.

Verwendung von While-Schleifen und Charts

Ziel Verwendung einer While-Schleife und eines Charts zum Erfassen und Anzeigen von Daten in Echtzeit.

Sie werden ein VI konstruieren, das Zufallsdaten generiert und in einem Chart anzeigt. Ein Knopf auf dem Front Panel stellt die Schleifengeschwindigkeit zwischen 0 und 2 Sekunden ein, und ein Schalter hält das VI an. Sie werden lernen, wie man die mechanische Aktion des Schalters ändert, so daß Sie nicht bei jeder Ausführung des VIs den Schalter einschalten müssen. Verwenden Sie zum Start das Front Panel in der folgenden Abbildung.

Front Panel



1. Öffnen Sie ein neues Front Panel durch Wählen von File»New, oder indem Sie auf die Taste “New VI” im Dialogfeld klicken.
2. Platzieren Sie einen vertikalen Schalter (**Controls»Boolean**) auf dem Front Panel. Beschriften Sie den Schalter mit Enable. Sie verwenden diesen Schalter zum Anhalten der Datenerfassung.
3. Verwenden Sie das Labeling Tool zum Erstellen des freien Etiketts für ON und OFF. Sie können diese Etikette durch Klicken auf das Labeling Tool und dann auf das Front Panel und durch das anschließende Eingeben des Etikettentextes erstellen. Verwenden Sie das Color Tool, um die Umrandung des freien Etiketts transparent zu gestalten. Durch Klicken auf das T in der unteren linken Ecke der Farbpalette wird das Etikett transparent.
4. Platzieren Sie ein Signalverlauf-Chart (**Controls»Graph**) auf dem Fenster der Bedienoberfläche. Beschriften Sie das Chart mit Random Signal. Das Chart zeigt Zufallsdaten in Echtzeit an.



Hinweis: *Vergewissern Sie sich, daß Sie einen Waveform-Chart, und keinen Waveform-Graph ausgewählt haben. In der Graph-Palette erscheint der Waveform-Chart der linken Seite der Palette am nächsten.*

5. Führen Sie auf dem Chart einen Pop-up durch, und wählen Sie **Show»Digital Display**. Das Digital Display (digitale Anzeige) zeigt den letzten Wert an.



6. Doppelklicken Sie mit dem Labeling Tool im Chart auf 10.0, geben Sie 1.0 ein, und klicken Sie außerhalb des Beschriftungsbereichs. Der Klick gibt den Wert ein. Sie können auch <Enter> (Windows); <return> (Macintosh); <Return> (Sun) oder <Enter> (HP-UX) drücken, um die Änderung der Skala einzugeben.

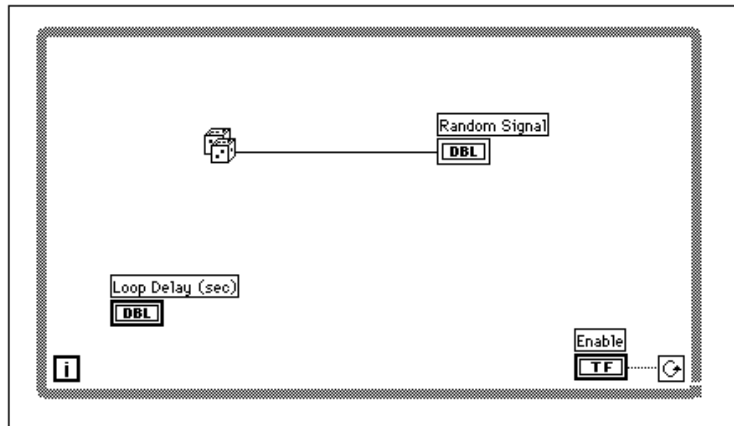


7. Plazieren Sie einen Drehknopf (**Controls»Numeric**) auf dem Front Panel. Beschriften Sie den Knopf mit `Loop Delay (sec)`. Dieser Knopf steuert später in diesem Übungsbeispiel das Timing der While-Schleife. Führen Sie auf dem Knopf einen Pop-up durch, und deaktivieren Sie **Show»Digital Display**, um das standardmäßig angezeigte Digital Display auszublenden.



8. Doppelklicken Sie mit dem Labeling Tool auf 10.0 in der Skala um den Knopf, geben Sie 2.0 ein, und klicken Sie zur Eingabe des neuen Werts außerhalb des Beschriftungsbereichs.


Blockdiagramm



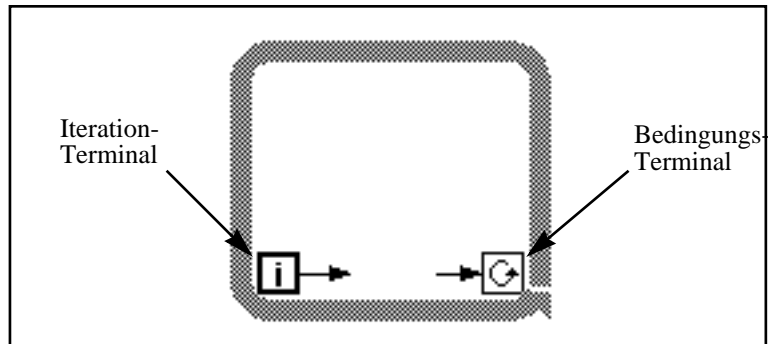
1. Öffnen Sie das Blockdiagramm.
2. Plazieren Sie die While-Schleife auf dem Blockdiagramm, indem Sie sie aus **Functions»Structures** wählen. Die While-Schleife ist ein umformbares Viereck, das nicht sofort auf dem Diagramm abgelegt wird. Sie haben stattdessen die Möglichkeit, es zu positionieren und umzuformen. Klicken Sie dazu auf den Bereich über und links neben allen Terminals. Halten Sie die Maustaste weiter gedrückt, und ziehen Sie die Maus so, daß die Terminals

von einem Rechteck eingeschlossen werden. Es wird dann eine While-Schleife mit der angegebenen Position und Größe erstellt.

Bedingungs-Terminal 

Iteration-Terminal 

Die in der folgenden Abbildung dargestellte While-Schleife ist ein umformbares Viereck, das Sie zum Ausführen des darin enthaltenen Diagramms verwenden können, bis der Boolesche Wert, der an das *Bedingungs-Terminal* (ein Eingabe-Terminal) übergeben wird, FALSE ist. Das VI überprüft das Bedingungs-Terminal am Ende jeder Wiederholung; daher *wird die While-Schleife immer mindestens einmal ausgeführt*. Das *Iteration-Terminal* ist ein numerisches Ausgabe-Terminal und gibt an, wie oft die Schleife ausgeführt wurde. Da die Wiederholungszählung immer bei null startet, gibt das Iteration-Terminal 0 aus, wenn die Schleife einmal ausgeführt wurde.



Die While-Schleife entspricht dem folgenden Pseudocode:

Do

Execute Diagram Inside the Loop (which sets the condition)

While Condition is TRUE



3. Wählen Sie die Funktion "Random Number (0-1)" aus **Functions»Numeric**.
4. Verdrahten Sie das Diagramm, wie in der ersten Abbildung dieses Abschnitts, *Blockdiagramm*, dargestellt; verbinden Sie dabei die Funktion "Random Number (0-1)" mit dem Chart-Terminal "Random Signal" und den Schalter "Enable" mit dem Bedingungs-Terminal der While-Schleife. Verdrahten Sie das Loop-Delay-Terminal jetzt noch nicht.
5. Kehren Sie zum Front Panel zurück, und schalten Sie den vertikalen Schalter ein, indem Sie mit dem Operating Tool darauf klicken. Führen Sie das VI aus.

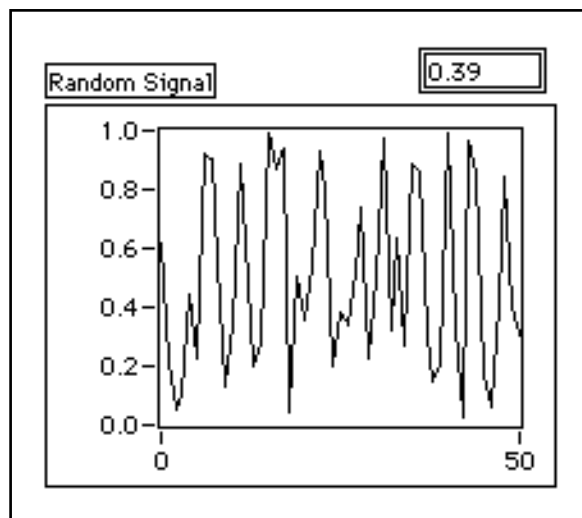


Die While-Schleife ist eine unendliche Schleifenstruktur. Das Diagramm innerhalb ihrer Umrandung wird solange ausgeführt, wie die angegebene Bedingung TRUE ist. Solange in diesem Beispiel der Schalter eingeschaltet (TRUE) ist, generiert das Diagramm Zufallszahlen und zeigt sie auf dem Chart an.

6. Klicken Sie zum Anhalten der Schleife auf den vertikalen Schalter. Durch Ausschalten des Schalters wird der Wert FALSE an das Bedingungs-Terminal der Schleife gesendet und die Schleife angehalten.
7. Der Chart hat einen Anzeige-Puffer, der eine gewisse Anzahl von Punkten enthält, nachdem sie nicht mehr auf der Anzeige zu sehen sind. Versehen Sie den Chart mit einem Rollbalken, indem Sie auf dem Chart einen Pop-up durchführen und **Show»Scrollbar** wählen. Sie können das Positioning Tool zum Einstellen der Größe und Positionieren des Rollbalkens verwenden.

Um den Chart durchzublättern, klicken und halten Sie den Maustaste über einem der Pfeile auf dem Rollbalkens gedrückt.

Um die Daten im Anzeige-Puffer zu löschen und den Chart rückzusetzen, führen Sie auf dem Chart einen Pop-up durch, und wählen Sie **Data Operations»Clear Chart**.





Hinweis: *Die Standardgröße des Anzeige-Puffers beträgt 1024 Punkte. Sie können die Puffergröße ändern, indem Sie auf dem Chart einen Pop-up durchführen und Chart History Length... wählen.*

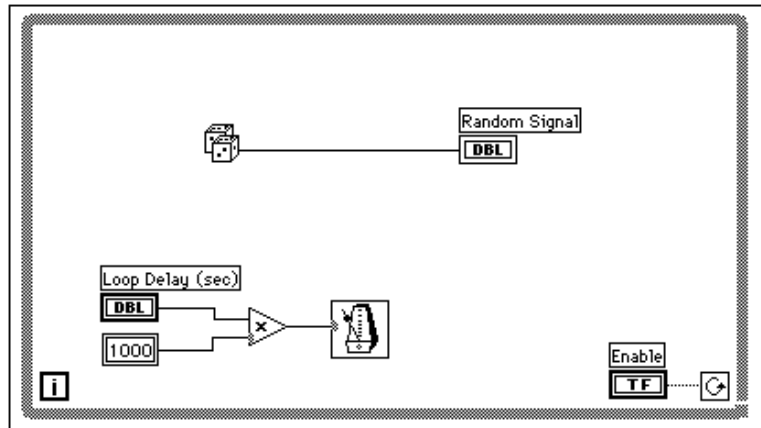
Hinzufügen von Timing

Die While-Schleife wurde beim Ablaufen des VIs so schnell wie möglich ausgeführt. Unter Umständen möchten Sie jedoch Daten zu bestimmten Intervallen aufnehmen, zum Beispiel einmal pro Sekunde oder einmal pro Minute.

Die Timing-Funktionen von LabVIEW drücken Zeit in Millisekunden (ms) aus; Ihr Betriebssystem hält eventuell diesen Grad der Timing-Genauigkeit nicht aufrecht. Die folgende Auflistung enthält Richtlinien zum Feststellen der Genauigkeit der LabVIEW-Timing-Funktionen bei Ihrem System.

- **(Windows 3.1)** Der Timer (Zeitgeber) hat eine Standardauflösung von 55 ms. Sie können LabVIEW auf eine Auflösung von 1 ms konfigurieren, indem Sie **Edit»Preferences...** wählen, aus dem Ring "Path" "Performance and Disk" wählen und dann das Kontrollkästchen "Use Default Timer" deaktivieren. LabVIEW verwendet die 1 ms Auflösung nicht als Standard, da dies eine größere Belastung des Betriebssystems bedeutet.
- **(Windows 95/NT)** Der Timer hat eine Auflösung von 1 ms. Dies ist jedoch hardwareabhängig, so daß Sie auf langsameren Systemen, zum Beispiel 80386-Systemen, u. U. ein niedrigeres Auflösungs-Timing haben.
- **(Macintosh)** Bei 68K-Systemen ohne QuickTime-Erweiterung hat der Timer eine Auflösung von $16 \frac{2}{3}$ ms (1/60 einer Sekunde). Wenn Sie einen Power Macintosh oder QuickTime installiert haben, beträgt die Timer-Auflösung 1 ms.
- **(UNIX)** Der Timer hat eine Auflösung von 1 ms.

Sie können das Schleifen-Timing durch Verwendung der Funktion "Wait Until Next ms Multiple" steuern (**Functions»Time & Dialog**). Diese Funktion stellt sicher, daß keine Wiederholung kürzer ist als die angegebene Anzahl von Millisekunden.



1. Modifizieren Sie das VI, um eine neue Zufallszahl zu dem von dem Knopf angegebenen Zeitintervall zu erstellen (siehe vorheriges Diagramm).



Funktion “Wait Until Next ms Multiple” (**Functions»Time & Dialog**). In diesem Übungsbeispiel multiplizieren Sie das Drehknopf-Terminal mit 1000, um dessen Wert von Sekunden in Millisekunden umzurechnen. Verwenden Sie diesen Wert als Eingabe für die Funktion “Wait Until Next ms Multiple”.



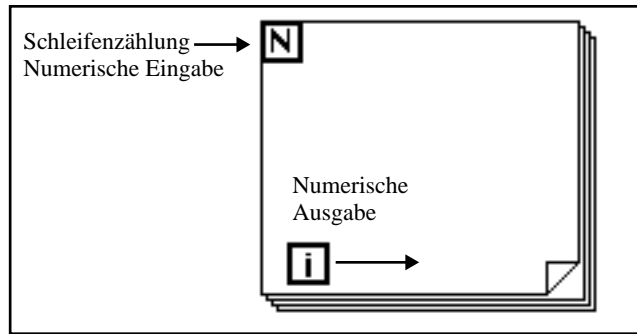
Funktion “Multiply” (**Functions»Numeric**). In diesem Übungsbeispiel multiplizieren Sie den Wert des Drehknopfes mit 1000, um von Sekunden in Millisekunden umzurechnen.



Numerische Konstante (**Functions»Numeric**). Sie enthält die Konstante, mit der Sie den Wert des Drehknopfes multiplizieren, um ihn in Millisekunden zu erhalten. Hat der Drehknopf einen Wert von 1.0, wird die Schleife alle 1000 Millisekunden einmal ausgeführt (einmal pro Sekunde).

2. Führen Sie das VI aus. Drehen Sie den Drehknopf, um andere Werte für die Anzahl der Sekunden zu erhalten.
3. Sichern und schließen Sie das VI unter mywork.11b. Nennen Sie es My Random Signal.vi.

For-Schleife



Sie platzieren die *For-Schleife* auf dem Blockdiagramm, indem Sie sie aus **Functions»Structures** wählen. Wie die While-Schleife ist die For-Schleife (siehe vorherige Abbildung) ein umformbares Viereck. Wie die While-Schleife wird sie nicht sofort auf dem Diagramm abgelegt. Stattdessen erscheint im Blockdiagramm-Fenster ein kleines, die For-Schleife darstellendes Icon, und Sie haben die Gelegenheit, es umzuformen und zu positionieren. Klicken Sie dazu zuerst in den Bereich oberhalb und links von allen Terminals. Während Sie die Maustaste gedrückt halten, ziehen Sie die Maus, so daß alle Terminals, die Sie innerhalb der For-Schleife plazieren wollen, von einem Rechteck umfaßt werden. Wenn Sie die Maustaste loslassen, erstellt LabVIEW eine For-Schleife der korrekten Größe und an der von Ihnen gewählten Position.

Die For-Schleife führt das Diagramm innerhalb ihrer Umrandung mit einer bestimmten Anzahl von Wiederholungen aus. Die For-Schleife hat zwei Terminals:



das *Zähl-Terminal* (ein Eingabe-Terminal). Das Zähl-Terminal gibt an, wie oft die Schleife ausgeführt werden soll.



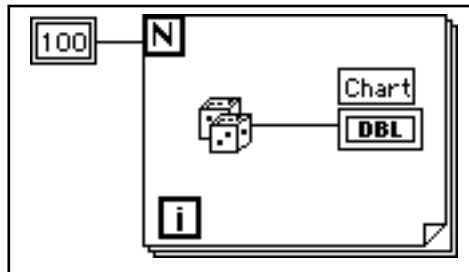
das *Iteration -Terminal* (ein Ausgabe-Terminal). Das Iteration-Terminal gibt an, wie oft die Schleife ausgeführt wurde.

Die For-Schleife entspricht dem folgenden Pseudocode:

```
For i = 0 to N-1
```

```
Execute Diagram Inside The Loop
```

Das Beispiel in der folgenden Abbildung zeigt eine For-Schleife, die 100 Zufallszahlen generiert und die Punkte auf dem Chart anzeigt.



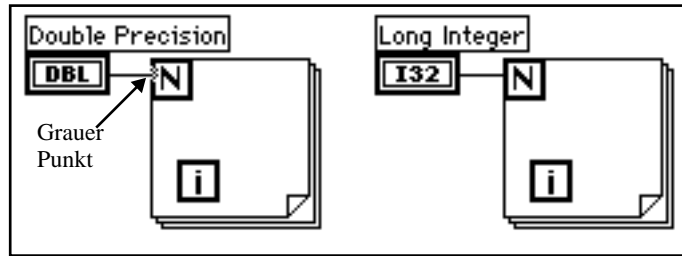
Numerische Konvertierung

Bisher waren alle von Ihnen verwendeten numerischen Controls und Indicators mit 32 Bit dargestellte Gleitkommazahlen mit doppelter Genauigkeit. LabVIEW kann jedoch Ziffern als Ganzzahlen (Byte, Word oder Long) oder als Gleitkommazahlen (single-, double- oder extended precision) darstellen. Standarddarstellung einer Ziffer ist Gleitkommazahl mit double-precision.

Wenn Sie zwei Terminals unterschiedlicher Datentypen miteinander verdrahten, setzt LabVIEW eines der Terminals auf dieselbe Darstellung wie das andere Terminal um. Als Gedächtnisstütze platziert LabVIEW einen grauen Punkt, den sogenannten Coercion Dot (Kennzeichen für Formatumwandlung), auf das Terminal, an dem die Konvertierung stattfindet.



Sehen Sie sich zum Beispiel das Zähl-Terminal der For-Schleife an. Die Darstellung des Terminals ist eine Ganzzahl vom Typ Long. Wenn Sie eine Gleitkommazahl mit double-precision mit dem Zählterminal verdrahten, setzt LabVIEW die Zahl in eine Ganzzahl vom Typ Long um. Beachten Sie den grauen Punkt auf dem Zähl-Terminal der ersten For-Schleife.



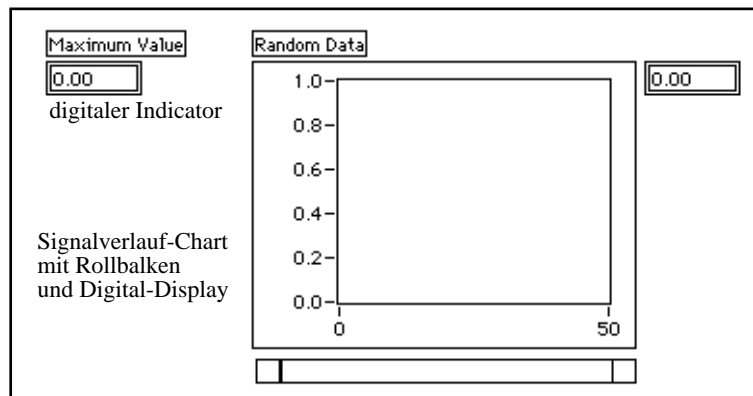
Hinweis: Wenn das VI Gleitkommazahlen in Ganzzahlen umsetzt, rundet es auf die nächste Ganzzahl. Liegt eine Zahl genau zwischen zwei Ganzzahlen, rundet es auf die nächste gerade Ganzzahl. Zum Beispiel rundet das VI 6.5 auf 6, aber 7.5 auf 8. Dies ist eine IEEE-Standardmethode zum Ablesen von Zahlen. Siehe IEEE Standard 754 für weitere Einzelheiten.

Verwendung einer For-Schleife

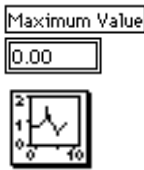
Ziel

Verwendung einer For-Schleife und von Shift Registern zur Berechnung des Maximalwerts in einer Reihe von Zufallszahlen. Sie werden eine For-Schleife (N=100) anstatt einer While-Schleife verwenden.

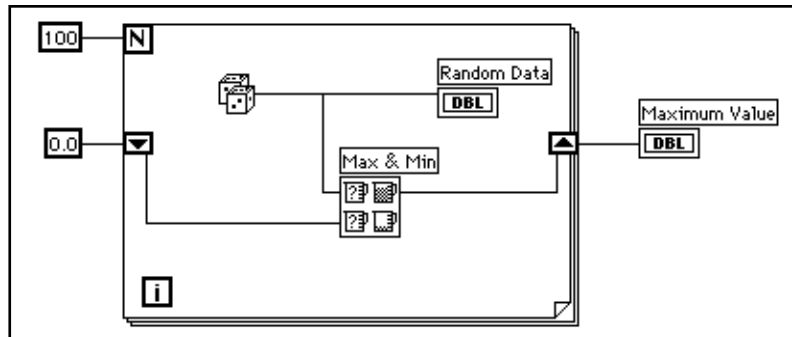
Front Panel



1. Öffnen Sie ein neues Front Panel, und fügen Sie die in der vorherigen Abbildung gezeigten Objekte hinzu.
 - a. Plazieren Sie einen digitalen Indicator auf dem Front Panel, und beschriften Sie ihn mit **Maximum Value**.
 - b. Plazieren Sie einen Waveform-Chart auf dem Front Panel, und beschriften Sie es mit **Random Data**. Ändern Sie die Skala des Charts in den Bereich 0.0 bis 1.0.
 - c. Führen Sie auf dem Chart einen Pop-up durch, und wählen Sie **Show»Scrollbar** und **Show»Digital Display**. Klicken Sie die Option **Show»Palette** an und deaktivieren Sie sie, falls sie ausgewählt wurde.



Blockdiagramm



1. Öffnen Sie das Blockdiagramm-Fenster.
2. Fügen Sie die **For-Schleife** hinzu (**Functions»Structures**).
3. Fügen Sie das Shift Register hinzu, indem Sie auf der linken oder rechten Umrandung der For-Schleife einen Pop-up ausführen und **Add Shift Register** wählen. Im nächsten Abschnitt können Sie mehr über Shift Register erfahren.
4. Fügen Sie die anderen Objekte dem Blockdiagramm hinzu.



Funktion "Random Number (0-1)" (**Functions»Numeric**). Erzeugt Zufallsdaten.



Numerische Konstante (**Functions»Numeric**). Die For-Schleife muß wissen, wie viele Wiederholungen sie durchführen muß. In diesem Fall führen Sie die For-Schleife 100 mal aus.



Numerische Konstante (**Functions»Numeric**). In diesem Übungsbeispiel legen Sie den Ausgangswert des Shift Registers auf null fest, da Sie wissen, daß die Ausgabe des Zufallszahl-Generators zwischen 0.0 und 1.0 liegt.

Sie müssen etwas über die von Ihnen erfaßten Daten wissen, damit Sie ein Shift Register initialisieren können. Wenn Sie zum Beispiel das Shift Register mit 1.0 initialisieren, ist der Wert bereits größer als alle erwarteten Datenwerte und somit immer der Maximalwert. Wenn Sie das Shift Register nicht initialisieren, enthielte es den Maximalwert einer früheren Ausführung des VIs. Sie könnten dann einen maximalen Ausgabewert erhalten, der nicht mit dem aktuellen Satz erfaßter Daten in Zusammenhang steht.



Die Funktion “Max & Min” (**Functions»Comparison**) nimmt zwei numerische Eingaben und gibt den Maximalwert der beiden Eingaben in der oberen rechten Ecke und den Minimalwert der beiden Eingaben in der unteren rechten Ecke aus. Da Sie in diesem Übungsbeispiel nur am Maximalwert interessiert sind, verdrahten Sie nur die Maximalausgabe und beachten den Minimalwert nicht.

5. Verdrahten Sie die Terminals wie dargestellt. Wäre der Maximalwert innerhalb der For-Schleife, würden Sie sehen, wie er fortlaufend aktualisiert wird; da er jedoch außerhalb der Schleife liegt, enthält er nur den letzten berechneten Maximalwert.



Hinweis: *Das Aktualisieren von Indicators bei jeder Wiederholung einer Schleife ist zeitaufwendig und möglichst zu vermeiden, um die Ausführungsgeschwindigkeit zu erhöhen.*

6. Führen Sie das VI aus.
7. Sichern Sie das VI. Nennen Sie das VI `My Calculate Max.vi`.

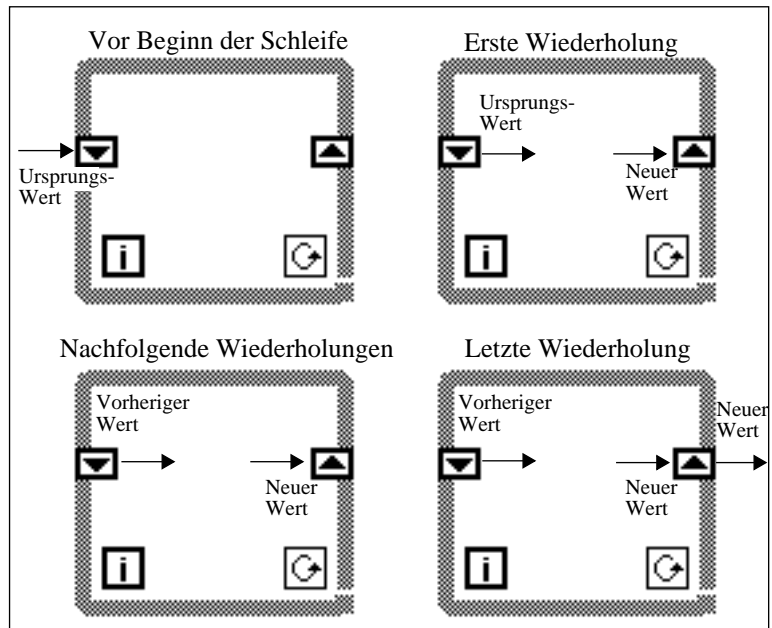
Shift Register

Shift Register (bei While-Schleifen und For-Schleifen verfügbar) übertragen Werte von einer Schleifenwiederholung zur nächsten. Sie erstellen ein Shift Register, indem Sie auf der rechten oder linken

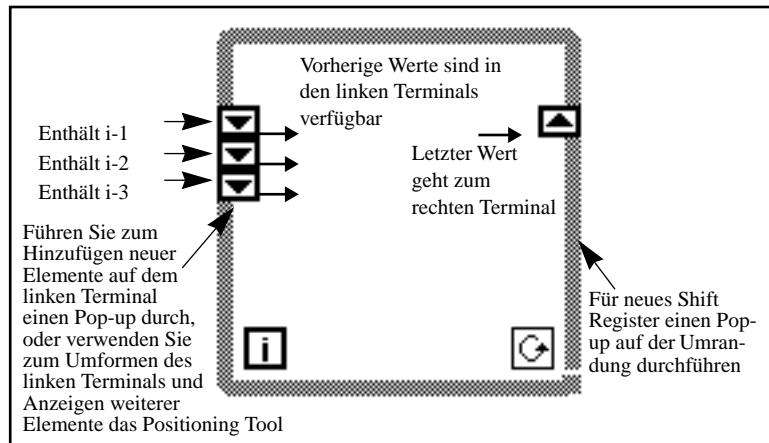
Umrandung einer Schleife einen Pop-up durchführen und **Add Shift Register** wählen.



Das Shift Register enthält ein Paar von Terminals, die direkt einander gegenüber auf den vertikalen Seiten der Schleifenumrandung angeordnet sind. Das *rechte* Terminal speichert die Daten nach Abschluß einer Wiederholung. Die Daten verschieben sich am Ende der Wiederholung und erscheinen zu Beginn der nächsten Wiederholung im *linken* Terminal (siehe folgende Abbildung). Ein Shift Register kann einen beliebigen Datentyp enthalten – Ziffer, Boolesches Element, String, Array usw. Das Shift Register paßt sich automatisch dem ersten Datentyp an, den Sie mit dem Shift Register verdrahten.



Sie können ein Shift Register so konfigurieren, daß es sich an Werte mehrerer vorhergehender Wiederholungen "erinnert". Dieses Leistungsmerkmal ist zur Durchschnittsbildung von Datenpunkten nützlich. Sie erstellen zusätzliche Terminals zum Zugriff auf Werte vorheriger Wiederholungen, indem Sie auf dem *linken* oder *rechten* Terminal einen Pop-up durchführen und **Add Element** wählen. Enthält ein Shift Register zum Beispiel drei Elemente im linken Terminal, können Sie auf die Werte der letzten drei Wiederholungen zugreifen.

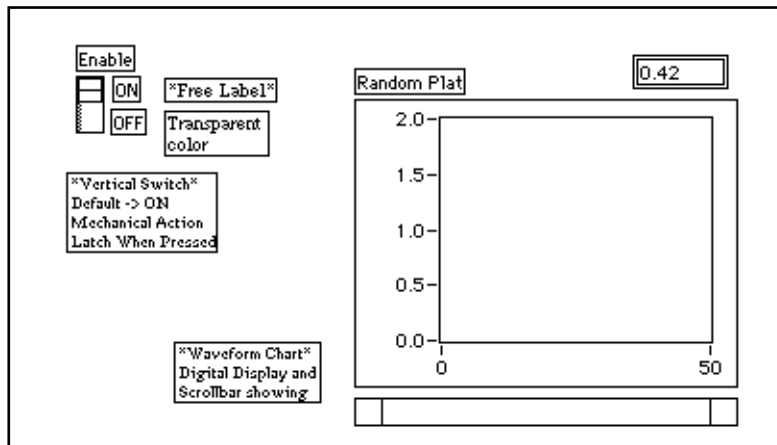


Verwendung von Shift Registers

ZIEL

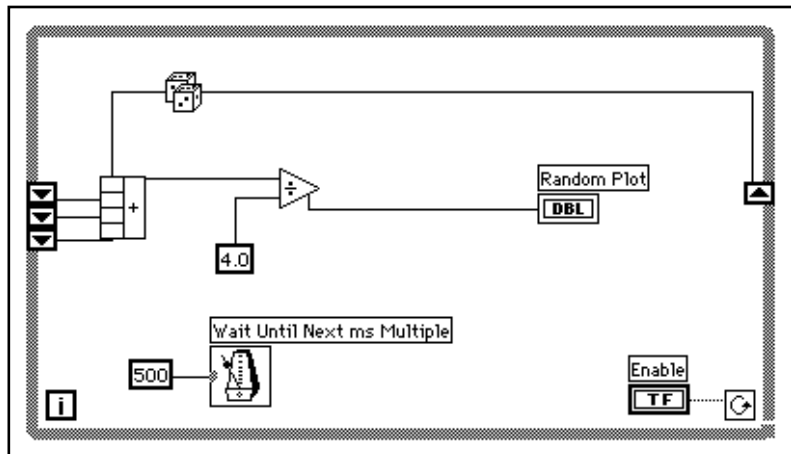
Sie werden ein VI konstruieren, das zwei Zufalls-Plots auf einem Chart anzeigt. Die zwei Plots sollten aus einem Zufalls-Plot und einem gleitenden Mittelwert von mindestens vier Punkten des Zufalls-Plots bestehen.

Front Panel



1. Öffnen Sie ein neues Front Panel, und erstellen Sie das in der vorherigen Abbildung dargestellte Front Panel.
2. Wenn Sie den Waveform-Chart zum Front Panel hinzugefügt haben, ändern Sie den Bereich der Skala von 0.0 auf 2.0.
3. Wenn Sie den vertikalen Schalter hinzugefügt haben, führen Sie einen Pop-up auf der Taste auf dem Front Panel durch, und wählen Sie **Mechanical Action»Latch When Pressed**; stellen Sie dann den Status ON als Standard ein, indem Sie **Operate»Make Current Values Default** wählen.

Blockdiagramm



1. Fügen Sie die **While-Schleife (Functions»Structures)** im Blockdiagramm-Fenster hinzu, und erstellen Sie das Shift Register.
 - a. Führen Sie auf der rechten oder linken Umrandung der While-Schleife einen Pop-up durch, und wählen Sie **Add Shift Register**.
 - b. Fügen Sie ein zusätzliches Element hinzu, indem Sie auf dem *linken* Terminal des Shift Registers einen Pop-up durchführen und **Add Element** wählen. Fügen Sie auf dieselbe Art ein drittes Element hinzu.
2. Konstruieren Sie das in der vorherigen Abbildung dargestellte Blockdiagramm.



Die Funktion “Random Number (0-1)” (**Functions»Numeric**) erstellt Rohdaten.

Funktion “Compound Arithmetic” (**Functions»Numeric**). In diesem Übungsbeispiel gibt die Funktion “Compound Arithmetic” die Summe der Zufallszahlen von zwei Wiederholungen aus. Führen Sie zum Hinzufügen weiterer Eingaben einen Pop-up auf einer Eingabe durch, und wählen Sie **Add Input** aus dem Pop-up Menü.



Funktion “Divide” (**Functions»Numeric**). In diesem Übungsbeispiel gibt die Funktion “Divide” den Durchschnitt der letzten vier Zufallszahlen aus.



Numerische Konstante (**Functions»Numeric**). Die Funktion “Random Number (0-1)” generiert während jeder Wiederholung der While-Schleife einen Zufallswert. Das VI fügt diesen Wert den letzten drei in den linken Terminals des Shift Registers gespeicherten Werten hinzu. Die Funktion “Random Number (0-1)” ermittelt durch Division durch vier (den aktuellen Wert plus die letzten drei Werte) den Durchschnitt der Werte. Der Durchschnitt wird dann auf dem Chart angezeigt.



Funktion “Wait Until Next ms Multiple” (**Functions»Time & Dialog**) stellt sicher, daß jede Schleifen-Wiederholung nicht schneller abläuft als die Millisekundeneingabe. Für dieses Übungsbeispiel ist die Eingabe 500 Millisekunden. Wenn Sie einen Pop-up auf dem Icon durchführen und **Show»Label** wählen, erscheint das Etikett Wait Until Next ms Multiple.

3. Führen Sie auf der Eingabe der Funktion “Wait Until Next ms Multiple” einen Pop-up durch, und wählen Sie **Create Constant**. Ein numerisches Constant erscheint und wird automatisch mit der Funktion verdrahtet.



4. Geben Sie im Etikett 500 ein. Das mit der Funktion “Wait Until Next ms Multiple” verdrahtete numerische Constant gibt eine Wartezeit von 500 Millisekunden (eine halbe Sekunde) an. Somit wird die Schleife jede halbe Sekunde ausgeführt.

Beachten Sie, daß das VI die Shift Register mit einer Zufallszahl initialisiert. Wenn Sie ein Shift Register Terminal nicht initialisieren, enthält es den Standardwert oder den letzten Wert der vorherigen Ausführung. In diesem Fall wären die ersten Durchschnittswerte bedeutungslos.

5. Führen Sie das VI aus, und beobachten Sie den Ablauf. LabVIEW stellt auf dem Graph nur den Durchschnitt graphisch dar.

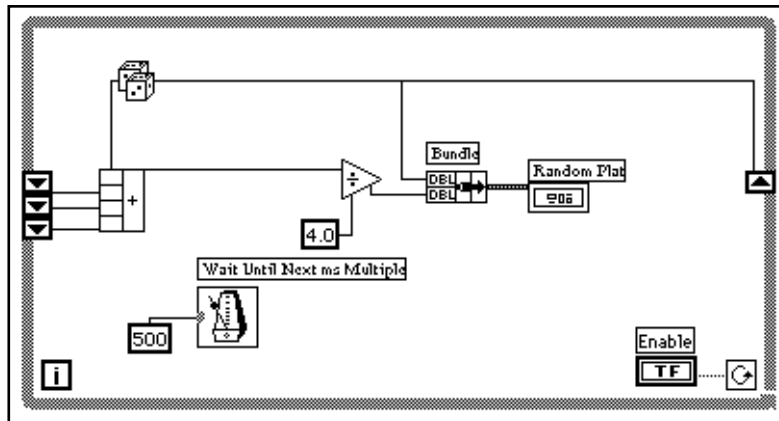


Hinweis: *Denken Sie daran, die Shift Register zu initialisieren. So vermeiden Sie, daß bei Ihren aktuellen Datenmessungen alte Daten oder Standardwerte benutzt werden.*

Multiplot-Charts

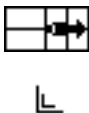
Charts können mehrere Plots aufnehmen. Im Fall mehrerer skalarer Eingänge müssen Sie die Daten gruppieren.

Sie sollten das Blockdiagramm so modifizieren, daß es den Durchschnitt und die aktuelle Zufallszahl auf demselben Chart anzeigt.



1. Modifizieren Sie das Blockdiagramm, wie in der Abbildung dargestellt.

Funktion “Bundle” (**Functions»Cluster**). In diesem Übungsbeispiel *bündelt* oder gruppiert die Funktion “Bundle” den durchschnittlichen und den aktuellen Wert zur graphischen Darstellung auf dem Chart. Der Bundle Node (Bundle-Knoten) erscheint an der linken Seite (siehe Abbildung), wenn Sie ihn im Blockdiagramm-Fenster platzieren. Wenn Sie auf dem Bundle Node einen Pop-up durchführen und **Show»Label** wählen, erscheint das Wort `Bundle` auf dem Etikett. Sie können weitere Elemente hinzufügen, indem Sie den Resizing Cursor zur Vergrößerung des Nodes verwenden (Zugriff auf den Cursor erfolgt durch Platzierung des Positioning Tools auf der Ecke der Funktion).



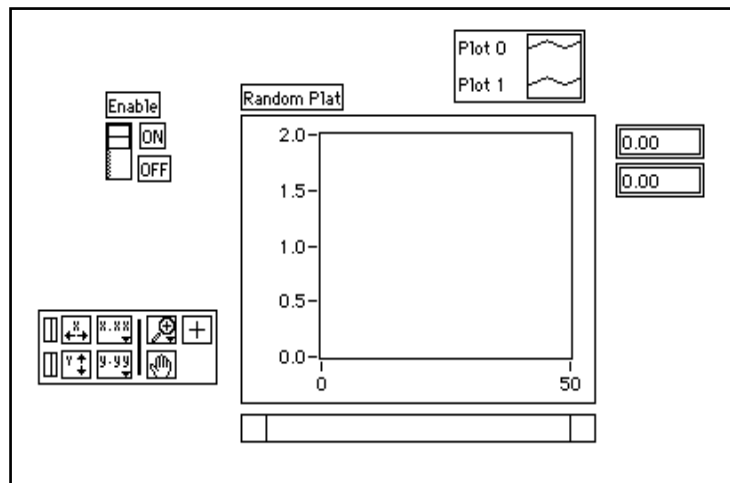
Hinweis: *Die Anordnung der Eingaben in die Funktion “Bundle” bestimmt die Anordnung der Plots auf dem Chart. Wenn Sie zum Beispiel die generierten Zufallszahlen mit der oberen Eingabe der Funktion “Bundle” und den gleitenden Mittelwert mit der unteren Eingabe verdrahten, entspricht der erste Plot den Zufallszahlen und der zweite Plot dem Durchschnitt.*

2. Führen Sie das VI aus. Das VI zeigt zwei Plots auf dem Chart an. Die Plots sind einander überlagert; d.h. sie haben eine gemeinsame vertikale Skala. Versuchen Sie, das VI mit aktiviertem Execution Highlighting auszuführen, um die Daten in den Shift Registers zu

sehen. Denken Sie daran, die Taste "Hilite execute" bei Abschluß wieder in der Toolbar zu deaktivieren, damit das VI bei voller Geschwindigkeit ausgeführt werden kann.

Anpassung von Charts

Sie können die Charts so anpassen, daß sie mit Ihren Datenanzeigeanforderungen übereinstimmen oder mehr Information anzeigen. Zu den für Charts verfügbaren Leistungsmerkmalen gehören: ein Rollbalken, eine Legende, eine Palette und ein Digital Display.



Das Digital Display wurde auf dem Chart aktiviert. Beachten Sie, daß für jede Ablaufverfolgung ein getrenntes digitales Display vorhanden ist.

1. Ist der Rollbalken vorhanden, blenden Sie ihn aus, indem Sie einen Pop-up auf dem Chart durchführen und **Show»ScrollBar** deaktivieren.
2. Passen Sie die Y-Achse an.
 - a. Doppelklicken Sie mit dem Labeling Tool auf 2.0 in der Y-Skala. Geben Sie 1.2 ein, und drücken Sie <Enter> (Windows); <return> (Macintosh); <Return> (Sun) oder <Enter> (HP-UX).
 - b. Klicken Sie mit dem Labeling Tool die zweite Zahl von unten auf der Y-Achse an. Ändern Sie diese Zahl auf 0.2, 0.5 oder



eine andere Zahl, die sich von der aktuellen Zahl unterscheidet. Diese Zahl bestimmt den numerischen Abstand der Unterteilungen auf der Y-Achse.



Hinweis: *Die Chart-Größe hat auf die Anzeige der Achsenskalen eine direkte Auswirkung. Vergrößern Sie den Chart, wenn Sie Probleme beim Anpassen der Achse haben.*

3. Blenden Sie die Legende ein, indem Sie auf dem Chart einen Pop-up durchführen und **Show»Legend** wählen. Verschieben Sie die Legende nach Bedarf.



Sie können die Legende an einer beliebigen Stelle plazieren. Dehnen Sie die Legende mittels Resizing Cursor, so daß sie zwei Plots umfaßt. Das Positioning Tool wird zum Resizing Cursor und zeigt so an, daß Sie die Legende umformen können. Benennen Sie 0 in *Current Value* um, indem Sie auf das Labeling Tool doppelklicken und den neuen Text eingeben. Sie können auf dieselbe Art Plot 1 in *Running Avg* umbenennen. Sollte der Text verschwinden, vergrößern Sie das Textfeld der Legende, indem Sie es an der *linken* Ecke der Legende mit dem Resizing Cursor umformen. Sie können den Linien- und Punktstil des Plots ändern, indem Sie auf dem Plot in der Legende einen Pop-up durchführen.

Sie können die Plotlinienbreite festlegen, indem Sie auf dem Plot in der Legende einen Pop-up durchführen. Anhand des Pop-up Menüs können Sie die Standardlinieneinstellung auf "größer als 1 Pixel" ändern. Sie können auch Haarstrichbreite wählen; diese Breite wird zwar nicht am Computerbildschirm angezeigt, aber von Druckern, die diesen Druckmodus unterstützen, gedruckt.



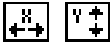
Wenn Sie einen Farbmonitor haben, können Sie außerdem die Farbe des Plot-Hintergrunds, der Ablaufverfolgungen und des Punktstils einstellen, indem Sie mit dem Color Tool auf dem zu ändernden Objekt einen Pop-up durchführen. Wählen Sie aus der dann erscheinenden Farbpalette die gewünschte Farbe.

4. Blenden Sie die schwebende Palette des Charts ein, indem Sie auf dem Chart einen Pop-up durchführen und **Show»Palette** wählen.

Sie können die Chart-Anzeige mit der Palette modifizieren, während das VI ausgeführt wird. Sie können jederzeit den Chart rücksetzen, die X- oder Y-Achse skalieren oder das Anzeigeformat ändern. Sie können außerdem mit dem Rollbalken andere Bereiche eines Graphs oder

Charts anzeigen oder diese Bereiche vergrößern. Sie können die Palette wie die Legende an einer beliebigen Stelle relativ zum Chart plazieren.

5. Führen Sie das VI aus. Verwenden Sie während der Ausführung des VIs die Tasten in der Palette, um das Chart zu modifizieren.



Sie können die Tasten “X” und “Y” verwenden, um die X- und Y-Achsen neu zu skalieren. Soll der Graph bei einer der beiden Skalen eine kontinuierliche Autoskalierung durchführen, klicken Sie auf den Sperrschalter links neben jeder Taste; so wird die Autoskalierung gesperrt.



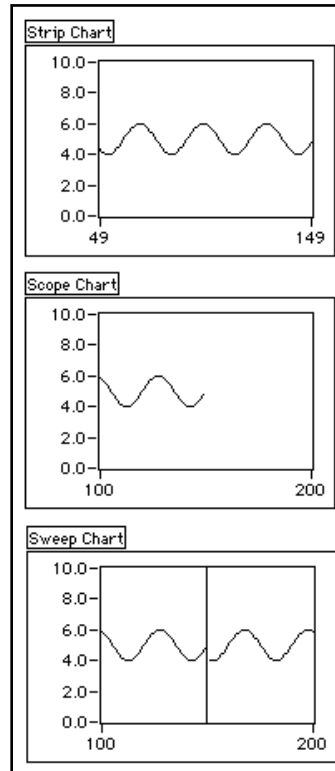
Sie können die anderen Tasten zur Modifikation der Genauigkeit des Achsentextes oder zur Steuerung der Betriebsart für das Chart verwenden. Experimentieren Sie mit diesen Tasten, um ihre Funktionsweise zu erkunden, durch den dargestellten Bereich zu blättern oder Bereiche des Charts zu vergrößern.



Hinweis: *Bei der Modifikation des Achsentextformats ist oft mehr Platz erforderlich, als ursprünglich für die Achse bereitgestellt wurde. Wenn Sie die Achse ändern, wird der Text eventuell größer als die Maximalgröße, die vom Waveform-Chart korrekt angezeigt werden kann. Verwenden Sie zur Korrektur dieses Problems den Resizing Cursor, um den Anzeige-Bereich des Charts zu verkleinern.*

Verschiedene Chart-Modi

Die folgende Abbildung stellt die drei verschiedenen Anzeigeoptionen für Charts dar, die über **Data Operations**»**Update Mode** verfügbar sind: Streifenschreiber (Strip Chart), Oszilloskop-Diagramm (Scope Chart) und Sweep-Chart. Der Standardmodus ist Streifenschreiber. (Wenn das VI noch ausgeführt wird, ist das Untermenü **Data Operations** das Pop-up Menü des Charts.)



Die Rollbalken-Anzeige des *Streifenschreiber*-Modus ist einem Streifenschreiber mit Papierband ähnlich. Wenn das VI einen neuen Wert empfängt, stellt er den Wert an der rechten Kante dar und verschiebt die alten Werte nach links.

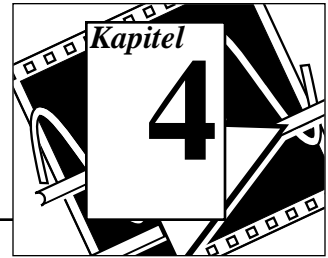
1. Stellen Sie sicher, daß das VI noch ausgeführt wird, führen Sie auf dem Chart einen Pop-up durch, und wählen Sie **Data Operations»Update Mode»Scope Chart**.

Der *Oszilloskop-Diagramm*-Modus hat eine Nachzieh-Anzeige, die einem Oszilloskop ähnlich ist. Wenn das VI einen neuen Wert empfängt, stellt es den Wert rechts neben dem letzten Wert graphisch dar. Erreicht das Plot die rechte Umrandung des Darstellungsbereichs, löscht das VI den Plot und beginnt erneut mit der graphischen Darstellung an der linken Seite. Das Oszilloskop-Diagramm ist erheblich schneller als der Streifenschreiber, da es nicht mit dem durch das Blättern entstehenden Verarbeitungsaufwand belastet ist.

2. Stellen Sie sicher, daß das VI noch ausgeführt wird, führen Sie auf dem Chart einen Pop-up durch, und wählen Sie **Data Operations»Update Mode»Sweep Chart**.

Der *Sweep-Chart*-Modus ist dem Oszilloskop-Diagramm sehr ähnlich, wird jedoch nicht gelöscht, wenn die Daten die rechte Umrandung erreichen. Stattdessen markiert eine sich bewegende vertikale Linie den Beginn der neuen Daten und bewegt sich über die Anzeige, während das VI neue Daten hinzufügt.

3. Halten Sie das VI an, und sichern Sie es. Nennen Sie es *My Random Average.vi*.



Arrays, Clusters und Graphs

Sie werden lernen:

- Näheres zu Arrays.
- Wie man Arrays mittels Schleifen generiert.
- Was Polymorphismus ist.
- Näheres zu Clusters.
- Wie man Graphen zur Datenanzeige verwendet.
- Wie man einige der grundlegenden Array-Funktionen verwendet.

Arrays

Ein Array besteht aus einer Sammlung von Datenelementen desselben Typs. Ein Array hat eine oder mehrere Dimensionen und bis zu $2^{31} - 1$ Elemente pro Dimension, vorausgesetzt, daß genügend Speicherkapazität vorhanden ist. Arrays in LabVIEW können von einem beliebigen Typ sein (außer Array, Chart oder Graph). Sie greifen auf jedes Array-Element über seinen Index zu. Der Index liegt im Bereich 0 bis $n-1$, wobei n die Anzahl der Elemente im Array ist. Das folgende eindimensionale Array numerischer Werte veranschaulicht diese Struktur. Beachten Sie, daß das erste Element den Index 0 hat, das zweite Element den Index 1 usw.

Index	0	1	2	3	4	5	6	7	8	9
10-Elemente Array	1.2	3.2	8.2	8.0	4.8	5.1	6.0	1.0	2.5	1.7

Array-Controls, Constants und Indicators

Sie erstellen Array-Controls, Constants und Indicators auf dem Front Panel oder Blockdiagramm, indem Sie eine Array-Shell mit einer Ziffer oder einem Booleschen Element, String oder Cluster verbinden. Das Array-Element kann kein anderes Array, Chart oder Graph sein.

Graphs

Ein *Graph Indicator* besteht aus einer zweidimensionalen Anzeige eines oder mehrerer Daten-Arrays, die *Plots* genannt werden. LabVIEW hat drei Graph-Typen: *XY -Graphs*, *Waveform-Graphs* und *Intensity-Graphs* (siehe Abschnitt *Weitere Themen* am Ende dieses Kapitels für Informationen zu Intensity-Graphs).

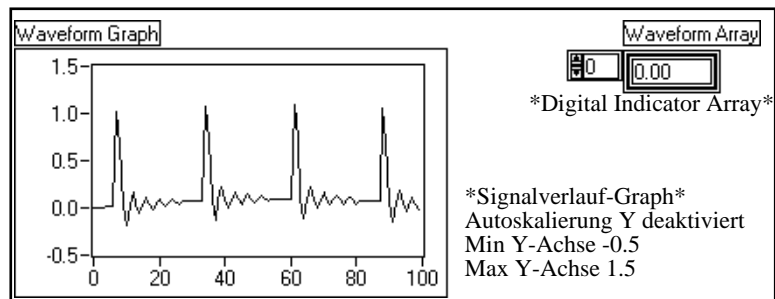
Der Unterschied zwischen einem Graph und einem Chart (in Kapitel 3, *Schleifen und Charts*, dieses Demonstrations-/Lernhandbuchs erläutert) besteht darin, daß ein Graph die Daten als Block graphisch darstellt, während ein Chart Daten Punkt für Punkt oder Array für Array graphisch darstellt.

Erstellung eines Arrays mit Auto-Indexing

Ziel Erstellung eines Graphen mit Auto-Indexingfunktion einer For-Schleife und graphischer Darstellung des Arrays in einem Waveform-Graph.

Sie werden ein VI konstruieren, das das VI "Generate Waveform" verwendet und das Array in einem Waveform-Graph graphisch darstellt. Sie werden außerdem das VI modifizieren, so daß es in der Lage ist, mehrere Plots graphisch darzustellen.

Front Panel

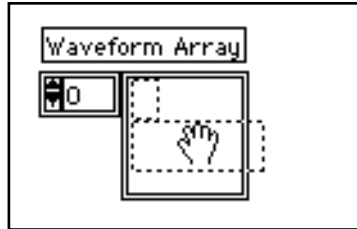


1. Öffnen Sie ein neues Front Panel.
2. Plazieren Sie eine Array Shell aus **Controls»Array & Cluster** auf dem Front Panel. Nennen Sie das Array-Constant Waveform Array.



1.23

- Plazieren Sie einen digitalen Indicator aus **Controls»Numeric** innerhalb der Element-Anzeige des Array-Shells (siehe folgende Abbildung). Der Indicator zeigt den Array-Inhalt an.



Wie zuvor erwähnt, ist ein *Graph Indicator* eine zweidimensionale Anzeige eines oder mehrerer Daten-Arrays, die *Plots* genannt werden. LabVIEW hat drei Graph-Typen: *XY -Graphs*, *Waveform-Graphs* und *Intensity-Graphs*.

- Plazieren Sie einen Waveform-Graph aus **Controls»Graph** auf dem Fenster der Bedienoberfläche. Nennen Sie den Graph *Waveform Graph*.

Der Waveform-Graph stellt Arrays mit gleichmäßig voneinander entfernten Punkten graphisch dar, zum Beispiel erfaßte zeitvariable Signalverläufe.



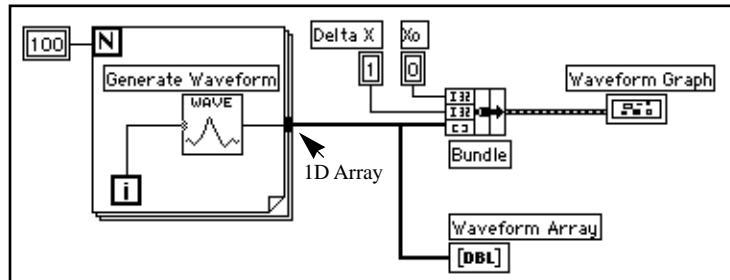
- Vergrößern Sie den Graph, indem Sie mit dem Resizing Cursor an einer Ecke ziehen.

Graphs *autoskalieren* ihre Eingabe standardmäßig. Das bedeutet, sie stellen automatisch die Skalierungsgrenzen der X- und Y-Achse ein, um den gesamten Eingabe-Datensatz anzuzeigen.

- Deaktivieren Sie die Autoskalierung, indem Sie auf dem Graph einen Pop-up durchführen und **Y Scale»Autoscale Y** deaktivieren.
- Modifizieren Sie die Y-Achsendaten, indem Sie mit dem Labeling Tool auf die Skalengrenzen doppelklicken und die neuen Zahlen eingeben. Ändern Sie das Y-Achsen-Minimum in -0.5 und das Maximum in 1.5.



Blockdiagramm



1. Konstruieren Sie das in der vorherigen Abbildung dargestellte Blockdiagramm.

Das VI “Generate Waveform” (**Functions»Tutorial**) gibt einen Punkt des Signalverlaufs aus. Das VI erfordert eine skalare Index-Eingabe; verdrahten Sie daher das Iteration-Terminal der Schleife mit dieser Eingabe. Durch Durchführen eines Pop-ups auf dem VI und Wählen von **Show»Label** wird das Wort Generate Waveform im Etikett angezeigt.

Beachten Sie, daß der Draht vom VI “Generate Waveform” dicker wird, wenn er zu einem Array an der Schleifen-Umrandung wechselt.

Die For-Schleife sammelt die skalaren Werte an ihrem Rahmen und bildet auf diese Weise ein Array. Dies wird *Auto-Indexing* genannt. In diesem Fall veranlaßt die mit der numerischen Eingabe der Schleifenzählung verdrahtete numerische Konstante die For-Schleife zur Erstellung eines 100-Elemente-Arrays (mit Index 0 bis 99).



Die Funktion “Bundle” (**Functions»Cluster**) stellt die Plot-Komponenten in einem Cluster zusammen. Sie müssen das Icon der Funktion “Bundle” umformen, bevor Sie es korrekt verdrahten können. Platzieren Sie das Positioning Tool auf der rechten unteren Ecke des Icons. Das Tool wird zum Resizing Cursor, wie in der linken Spalte dargestellt. Wenn sich das Tool verändert, klicken Sie es an, und ziehen Sie es nach unten, bis ein Drittel des Eingabe-Terminals erscheint. Sie können nun mit dem Verdrahten Ihres Blockdiagramms, wie in der ersten Abbildung dieses Abschnitts dargestellt, fortfahren.

Ein Cluster besteht aus einem Datentyp, der Datenelemente verschiedener Typen enthalten kann. Der Cluster im Blockdiagramm,

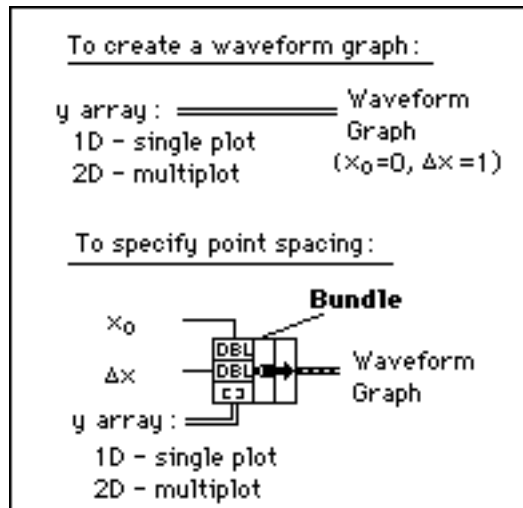
das Sie in diesem Übungsbeispiel konstruieren, gruppiert verwandte Datenelemente von verschiedenen Stellen auf dem Diagramm; dadurch wird eine unübersichtliche Verdrahtung vermieden. Wenn Sie Clusters verwenden, erfordern Ihre SubVIs weniger Verbindungsterminals. Ein Cluster ist analog zu einem Record in Pascal oder einem Struct in C. Sie können sich einen Cluster wie ein Drahtbündel, ähnlich einem Telefonkabel, vorstellen. Jeder Draht in dem Kabel stellt dann ein anderes Element des Clusters dar. Die Komponenten umfassen den Ursprungs-X-Wert (0), den Delta-X-Wert (1) und das Y-Array (von den numerischen Konstanten im Blockdiagramm bereitgestellte Signalverlauf-Daten). Verwenden Sie bei LabVIEW zur Zusammenstellung eines Clusters die Funktion "Bundle".



Hinweis: Stellen Sie sicher, daß Sie Datentypen konstruieren, die von Graphs und Charts akzeptiert werden.

Überprüfen Sie bei der Konstruktion Ihres Blockdiagramms Ihre Datentypen anhand folgender Schritte:

- Öffnen Sie das Fenster "Help" durch Wählen von **Help»Show Help**.
- Verschieben Sie das Wiring Tool auf das Graph-Terminal.
- Überprüfen Sie die Datentyp-Informationen, die im Fenster "Help" erscheinen. Siehe zum Beispiel folgende Abbildung:





Numeric Constant (**Functions»Numeric**). Drei numerische Konstanten legen die Anzahl der Wiederholungen der For-Schleife, den Ursprungs-X-Wert und den Delta-X-Wert fest. Beachten Sie, daß Sie auf dem Zähl-Terminal der For-Schleife (siehe linke Spalte) einen Pop-up durchführen und **Create Constant** wählen können, um automatisch eine numerische Konstante für das Terminal hinzuzufügen und zu verdrahten.

Bei jedem Schleifendurchlauf wird ein Punkt der Kurve generiert und am Schleifenrand zu einem Array zusammengefaßt. Nachdem die Ausführung der Schleife beendet ist, gruppiert die Funktion "Bundle" den Ursprungs-X-Wert (X_0), den Delta-X-Wert und das Array zur graphischen Darstellung auf dem Graph.

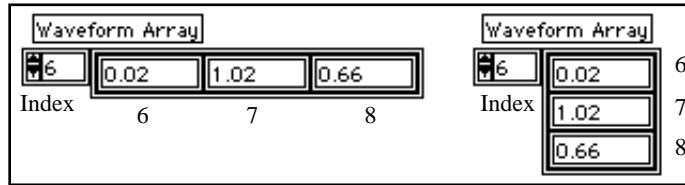
2. Kehren Sie zum Front Panel zurück, und führen Sie das VI aus. Das VI stellt das mit Auto-Indexing erstellte Signalverlauf-Array auf dem Signalverlauf-Graph graphisch dar. Der Ursprungs-X-Wert ist 0, der Delta-X-Wert ist 1.
3. Ändern Sie den Delta-X-Wert in 0.5 und den Ursprungs-X-Wert in 20. Führen Sie das VI erneut aus.

Beachten Sie, daß der Graph nun dieselben 100 Datenpunkte mit einem Ausgangswert von 20 und einem Delta-X-Wert von 0.5 für jeden Punkt darstellt (siehe X-Achse). In einem Zeittest entspräche dieser Graph 50 Sekunden von Daten, beginnend bei 20 Sekunden. Experimentieren Sie mit verschiedenen Kombinationen für die Ursprungs- und die Delta-X-Werte.

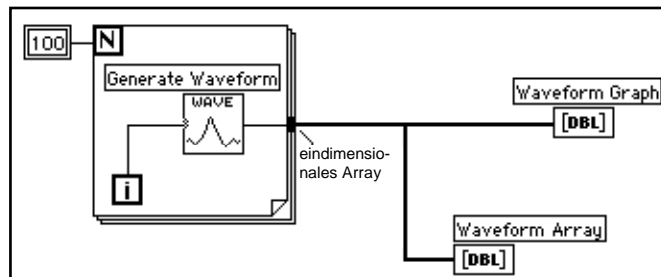
4. Sie können die einzelnen Array-Elemente betrachten, indem Sie den Index der Elemente in die Index-Anzeige eingeben. Wenn Sie eine Zahl eingeben, die größer als die Array-Größe ist, verblaßt die Anzeige, was darauf hinweist, daß Sie keinen Wert für diesen Index definiert haben.



Wenn Sie mehr als ein Element auf einmal betrachten wollen, können Sie den Array-Indicator umformen. Plazieren Sie das Positioning Tool auf der rechten unteren Ecke des Arrays. Das Tool verwandelt sich in den Resizing Cursor (siehe linke Spalte). Wenn sich das Tool ändert, ziehen Sie den Cursor nach rechts oder gerade nach unten. Das Array zeigt nun verschiedene Elemente in aufsteigender Index-Ordnung an; dabei wird mit dem Element begonnen, das dem angegebenen Index entspricht (siehe folgende Abbildung).



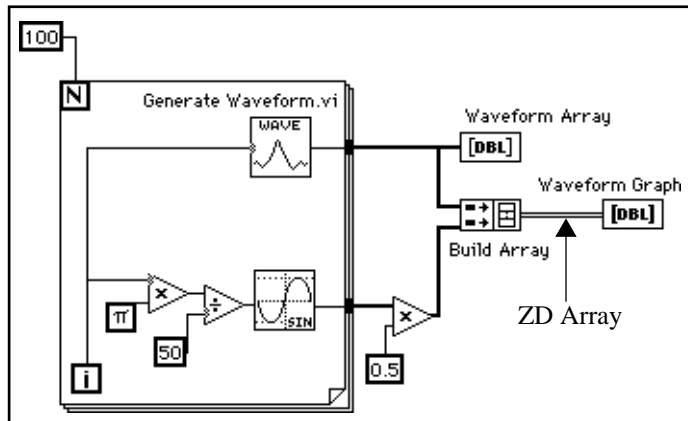
Im vorherigen Blockdiagramm haben Sie einen Ursprungs-X-Wert und einen Delta-X-Wert für den Signalverlauf angegeben. Häufig ist jedoch der Ursprungs-X-Wert null und der Delta-X-Wert 1. In diesen Fällen können Sie das Signalverlauf-Array direkt mit dem Signalverlauf-Graph-Terminal verdrahten (siehe folgende Abbildung).



5. Kehren Sie zum Blockdiagramm zurück. Löschen Sie die Funktion "Bundle" und die mit ihr verdrahteten numerischen Konstanten. Zum Löschen der Funktion und Konstanten wählen Sie die Funktion und Konstanten mit dem Positioning Tool, und drücken dann <Delete>. Wählen Sie **Edit>Remove Bad Wires**. Beenden Sie das Verdrahten des Blockdiagramms, wie in der vorherigen Abbildung dargestellt.
6. Führen Sie das VI aus. Beachten Sie, daß das VI den Signalverlauf mit einem Ursprungs-X-Wert von 0 und einem Delta-X-Wert von 1 graphisch darstellt.

Multiplot-Graphs

Sie können Multiplot-Waveform-Graphen erstellen, indem Sie ein Array des Datentyps konstruieren, der normalerweise an einen Einzel-Plot-Graph übergeben wird.



1. Fahren Sie mit der Konstruktion Ihres Blockdiagramms, wie in der vorherigen Abbildung dargestellt, fort.



Funktion “Sine” (**Functions»Numeric»Trigonometric**). In diesem Übungsbeispiel verwenden Sie die Funktion in einer For-Schleife, um ein Array von Punkten zu konstruieren, das den Zyklus einer Sinuskurve darstellt.



Funktion “Build Array” (**Functions»Array**). In diesem Übungsbeispiel verwenden Sie die Funktion zur Erstellung der korrekten Datenstruktur, um zwei Arrays auf einem Waveform-Graph darzustellen; der Graph ist in diesem Fall ein zweidimensionales Array. Vergrößern Sie zur Erstellung von zwei Eingaben die Funktion “Build Array”, indem Sie eine Ecke mit dem Positioning Tool ziehen.



Pi Constant (**Functions»Numeric»Additional Numeric Constants**).

Denken Sie daran, daß Sie die Funktionen “Multiply” und “Divide” unter **Functions»Numeric** finden.

2. Wechseln Sie zum Front Panel. Führen Sie das VI aus.

Beachten Sie, daß die beiden Signalverläufe auf demselben Waveform-Graph graphisch dargestellt werden. Für beide Datensätze ist der Ausgangsparameter für den X-Wert gleich 0, für den Delta-X-Wert gleich 1.



Hinweis: *Sie können das Aussehen der graphischen Darstellung auf dem Graphen ändern, indem Sie auf der Legende eines bestimmten Plots einen Pop-up*

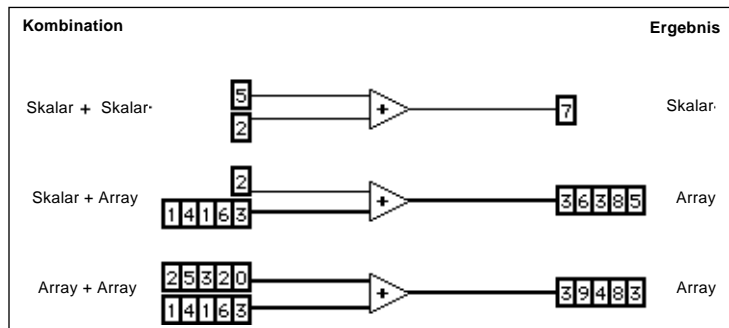
durchführen. Sie können zum Beispiel einen Linien-Graph in einen Balken-Graph ändern, indem Sie Common Plots»Bar Graph wählen.

3. Sichern und schließen Sie das VI. Nennen Sie es My Graph Waveform Arrays.vi. Stellen Sie sicher, daß Sie Ihre Daten unter mywork.llb sichern.

Polymorphismus

Polymorphismus ist die Fähigkeit einer Funktion, sich Eingabedaten verschiedener Typen, Dimensionen und Darstellungen anzupassen. Die meisten LabVIEW-Funktionen sind polymorph. Das vorherige Blockdiagramm ist ein Beispiel für Polymorphismus. Beachten Sie, daß Sie die Funktion “Multiply” an zwei Stellen verwenden, innerhalb und außerhalb der For-Schleife. Innerhalb der For-Schleife multipliziert die Funktion zwei Skalarwerte; außerhalb der For-Schleife multipliziert die Funktion ein Array mit einem Skalarwert.

Das folgende Beispiel zeigt einige der polymorphen Kombinationen der Funktion “Add”.



In der ersten Kombination werden zwei Skalare addiert, und das Ergebnis ist ein Skalar. In der zweiten Kombination wird ein Skalar zu den einzelnen Array-Elementen hinzuaddiert, und das Ergebnis ist ein Array. In der dritten Kombination werden die einzelnen Array-Elementen den entsprechenden Elementen eines anderen Arrays hinzuaddiert. Sie können auch andere Kombinationen verwenden, wie zum Beispiel Ziffern-Clusters, Cluster-Arrays usw.

Diese Prinzipien können auf andere LabVIEW-Funktionen und -Datentypen angewendet werden. LabVIEW-Funktionen können in unterschiedlichen Maßen polymorph sein. Einige Funktionen können

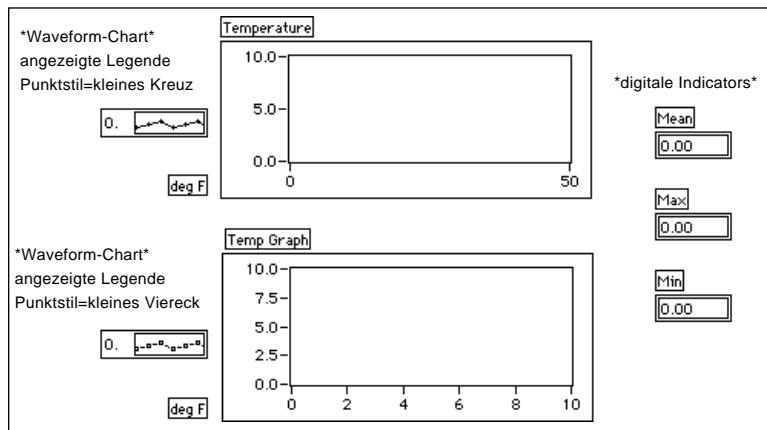
numerische und Boolesche Eingaben akzeptieren, andere können eine Kombination beliebiger Datentypen akzeptieren.

Verwendung der Graph- und Analyse-VIs

ZIEL

Sie werden ein VI konstruieren, das alle 0,25 Sekunden eine 10 Sekunden dauernde Temperaturmessung durchführt. Während der Datenerfassung zeigt das VI die Messungen in Echtzeit auf einem Streifenschreiber (Strip Chart) an. Nach Durchführung der Datenerfassung stellt das VI die Daten graphisch dar und berechnet die Durchschnitts-, Maximal- und Minimaltemperaturen.

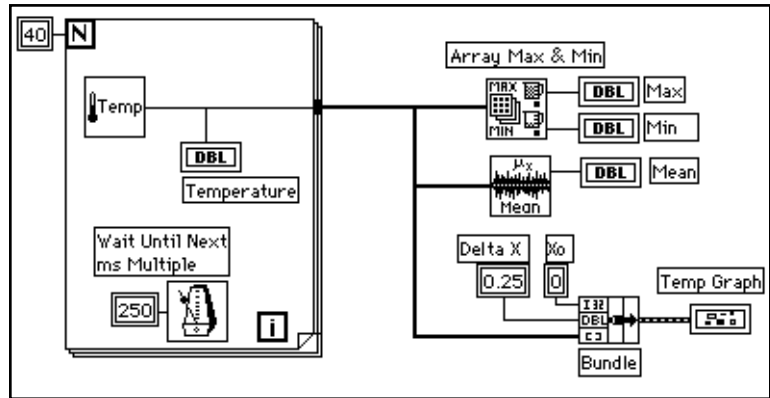
Front Panel



1. Öffnen Sie ein neues Front Panel, und konstruieren Sie das in der vorherigen Abbildung dargestellte Front Panel. Sie können die Punktarten des Waveform-Charts und des Signalverlauf-Graphen ändern, indem Sie auf ihren Legenden einen Pop-up durchführen.

Das Waveform-Chart “Temperature” zeigt die Temperatur an, während sie erfaßt wird. Nach der Erfassung stellt das VI die Daten auf dem “Temp Graph” graphisch dar. Die digitalen Indicators “Mean”, “Max” und “Min” zeigen die Durchschnitts-, Maximal- und Minimaltemperaturen an.

Blockdiagramm



1. Konstruieren Sie unter Verwendung der folgenden Elemente das in der vorherigen Abbildung dargestellte Blockdiagramm:



Das VI “Digital Thermometer” (**Functions»Tutorial**); Sie können auch das von Ihnen in Kapitel 2 erstellte VI verwenden, indem Sie **Functions»Select a VI...** und dann das VI “My Thermometer” wählen. Es liefert einen Temperaturwert.



Funktion “Wait Until Next ms Multiple” (**Functions»Time & Dialog**). In diesem Übungsbeispiel stellt diese Funktion sicher, daß die For-Schleife alle 0,25 Sekunden (250 Millisekunden) ausgeführt wird.



Numeric Constant (**Functions»Numeric**). Sie können auch einen Pop-up auf der Funktion “Wait Until Next ms Multiple” durchführen und **Create Constant** wählen, um automatisch eine numerische Konstante zu erstellen und zu verdrahten.



Funktion “Array Max & Min” (**Functions»Array**). In diesem Beispiel gibt diese Funktion die während der Erfassung gemessenen Maximal- und Minimaltemperaturen aus.



Das VI “Mean” (**Functions»Analysis»Probability and Statistics**) gibt den Durchschnittswert der Temperaturmessungen aus.



Funktion “Bundle” (**Functions»Cluster**) setzt die Plot-Komponenten zu einem Cluster zusammen. Die Komponenten umfassen den Ursprungs-X-Wert (0), den Delta-X-Wert (0.25) und das Y-Array

(Temperaturdaten). Verwenden Sie das Positioning Tool zur Umformung der Funktion, indem Sie an einer Ecke ziehen.

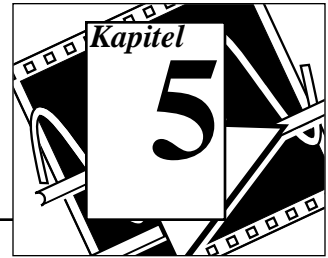
Die For-Schleife wird 40 mal ausgeführt. Durch die Funktion “Wait Until Next ms Multiple” findet jede Wiederholung alle 250 Millisekunden statt. Das VI speichert die Temperaturmessungen in einem Array, das auf der Umrandung der For-Schleife erstellt wurde (Auto-Indexing). Wenn die For-Schleife die Ausführung beendet, wird das Array an verschiedene Nodes übergeben.

Die Funktion “Array Max & Min” gibt die Maximal- und Minimaltemperatur aus. Das VI “Mean” errechnet den durchschnittlichen Temperaturwert.

Ihr fertiggestelltes VI gruppiert das Daten-Array mit einem Ursprung-X-Wert von 0 und einem Delta-X-Wert von 0.25. Für das VI ist ein Delta-X-Wert von 0.25 erforderlich, so daß das VI die Temperatur-Array-Punkte alle 0,25 Sekunden auf dem Signalverlauf-Graph darstellt.

2. Kehren Sie zum Front Panel zurück, und führen Sie das VI aus.
3. Sichern Sie das VI in mywork.llb als My Temperature Analysis.vi.

Case- und Sequence-Strukturen und der Formula Node



Sie werden lernen:

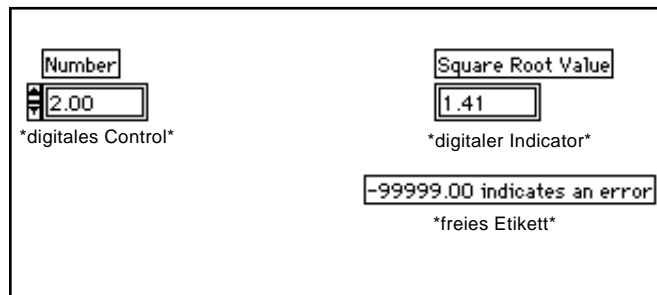
- Wie man die Case-Struktur verwendet.
- Wie man die Sequence-Struktur verwendet.
- Was Sequence Locals sind und wie sie verwendet werden.
- Was ein Formula Node ist und wie er verwendet wird.

Verwendung der Case-Struktur

ZIEL

Sie werden ein VI konstruieren, das eine Zahl darauf prüft, ob sie positiv ist. Ist die Zahl positiv, berechnet das VI ihre Quadratwurzel; andernfalls gibt das VI einen Fehler aus.

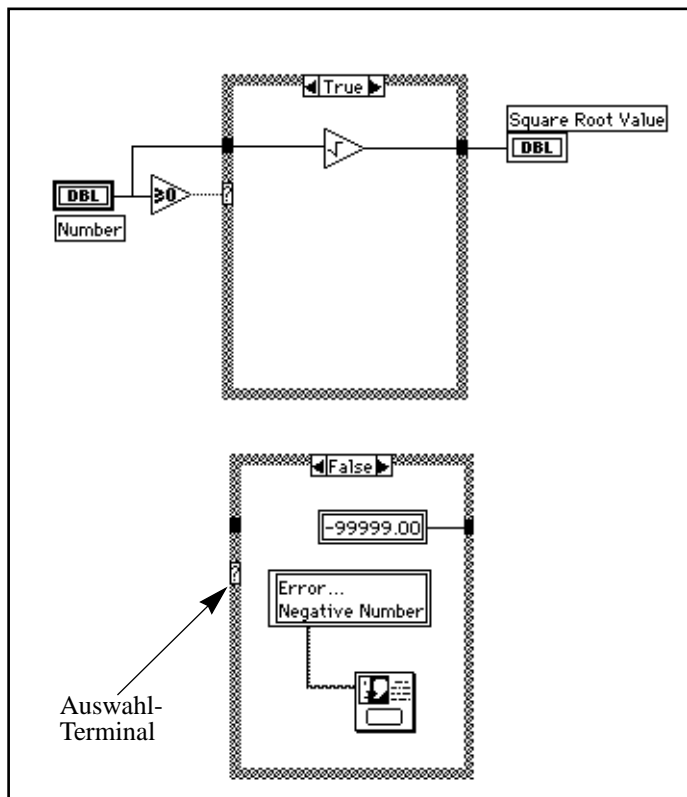
Front Panel



1. Öffnen Sie ein neues Front Panel, und konstruieren Sie das in der vorherigen Abbildung dargestellte Front Panel.

Das Control "Number" liefert die Zahl. Der Indikator "Square Root Value" zeigt die Quadratwurzel dieser Zahl an. Das freie Etikett dient als Hinweis für den Benutzer.

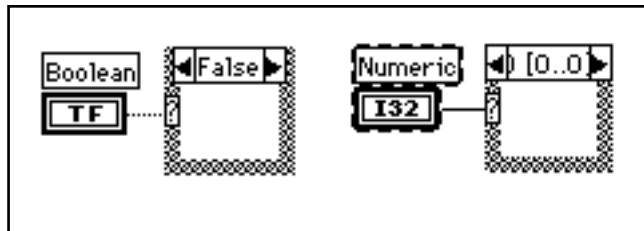
Blockdiagramm



1. Öffnen Sie das Blockdiagramm.
2. Plazieren Sie eine Case-Struktur (**Functions»Structures**) auf dem Blockdiagramm-Fenster. Vergrößern Sie die Case-Struktur, indem Sie mit dem Resizing Cursor an einer Ecke ziehen.

Die Case-Struktur hat standardmäßig nur zwei Cases (Fälle): TRUE und FALSE. Eine Boolesche Case-Struktur ist analog zu einer Wenn-Dann-Aussage in textorientierten Programmiersprachen. Sie wird

automatisch auf numerisch geändert, wenn Sie ein numerisches Control mit dem Auswahl-Terminal verdrahten.



Sie können nur jeweils einen Case anzeigen. Zum Ändern von Cases klicken Sie auf die Pfeile oben in der Case-Struktur.

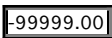
3. Wählen Sie andere Blockdiagrammobjekte, und verdrahten Sie sie wie in der Abbildung des Blockdiagramm-Fensters dargestellt.



Funktion “Greater Or Equal To 0?” (**Functions»Comparison**). In diesem Übungsbeispiel bestimmt die Funktion, ob die Zahleneingabe negativ ist. Die Funktion gibt TRUE aus, wenn die Zahleneingabe gleich oder größer 0 ist.



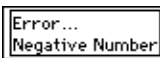
Funktion “Square Root” (**Functions»Numeric**). In diesem Übungsbeispiel gibt die Funktion die Quadratwurzel der Eingabezahl aus.



Numerische Konstante (**Functions»Numeric**).



One Button Dialog Funktion (**Functions»Time & Dialog**). In diesem Übungsbeispiel zeigt die Funktion ein Dialogfenster mit der Meldung Error...Negative Number an.



String Constant (**Functions»String**). Geben Sie mit dem Labeling Tool Text in dieses Feld ein.

In diesem Übungsbeispiel führt das VI entweder die TRUE oder die FALSE Bedingung aus. Ist die Zahl größer oder gleich null, führt das VI die TRUE Bedingung aus und gibt die Quadratwurzel der Zahl aus. Die FALSE Bedingung gibt -99999.00 aus und zeigt ein Dialogfenster mit der Meldung Error...Negative Number an.



Hinweis: *Sie müssen für jeden Case den Ausgabe-Tunnel definieren. Wenn Sie bei einem Case einen Ausgabe-Tunnel erstellen, erscheinen bei allen anderen*

Cases in derselben Position ebenfalls Tunnel. Nicht verdrahtete Tunnel erscheinen als weiße Quadrate.

Um sicherzustellen, daß jeder nicht verdrahtete Case mit dem Ausgabe-Tunnel verdrahtet wird, klicken Sie jedesmal auf den Tunnel selbst. In diesem Übungsbeispiel ordnen Sie dem Ausgabe-Tunnel bei der FALSE Bedingung einen Wert zu, da die TRUE Bedingung einen Ausgabe-Tunnel hat. Wenn Sie die Ausgabe nicht bei allen Cases einem Wert zuordnen wollen, müssen Sie den Indicator auf diesem Case positionieren oder eine globale oder lokale Variable verwenden.

4. Kehren Sie zum Front Panel zurück, und führen Sie das VI aus. Probieren Sie eine Zahl größer als null und eine Zahl kleiner als null aus, indem Sie den Wert im digitalen Control, das Sie mit `number` beschriftet haben, ändern. Beachten Sie, daß LabVIEW die von Ihnen für die FALSE Bedingung eingerichtete Fehlermeldung der Case-Struktur anzeigt, wenn Sie das digitale Control in eine negative Zahl ändern.
5. Sichern und schließen Sie das VI. Nennen Sie es `My Square Root.vi`.

VI-Logik

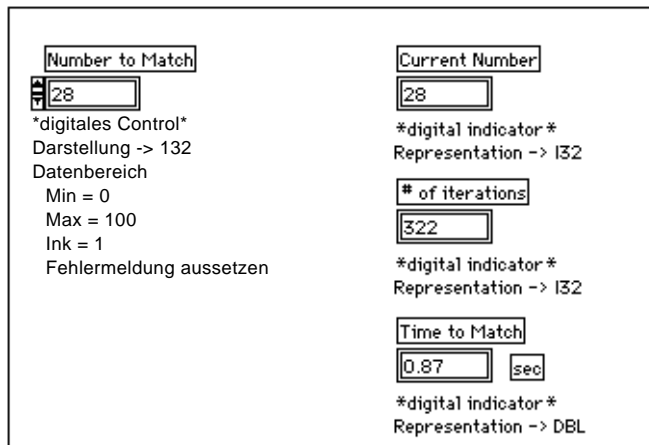
```
if (Number >= 0) then
Square Root Value = Sqrt(Number)
else
Square Root Value = -99999.00
Display Message "Error...Negative Number"
end if
```

Verwendung der Sequence-Struktur

ZIEL

Sie werden ein VI konstruieren, das die Zeit berechnet, die zur Generierung einer mit einer gegebenen Zahl übereinstimmenden Zufallszahl erforderlich ist.

Front Panel



- Öffnen Sie ein neues Front Panel, und konstruieren Sie das in der vorherigen Abbildung dargestellte Front Panel. Stellen Sie sicher, daß Sie die Controls und Indicators entsprechend der auf die Abbildung folgenden Anweisungen modifizieren.

Das Control `Number to Match` enthält die Zahl, für die Sie eine Übereinstimmung finden wollen. Der Indicator `Current Number` zeigt die aktuelle Zufallszahl an. Der Indicator `# of iterations` zeigt die Zahl der Wiederholungen vor der Übereinstimmung an. `Time to Match` zeigt an, wieviele Sekunden es gedauert hat, bevor eine übereinstimmende Zahl gefunden wurde.

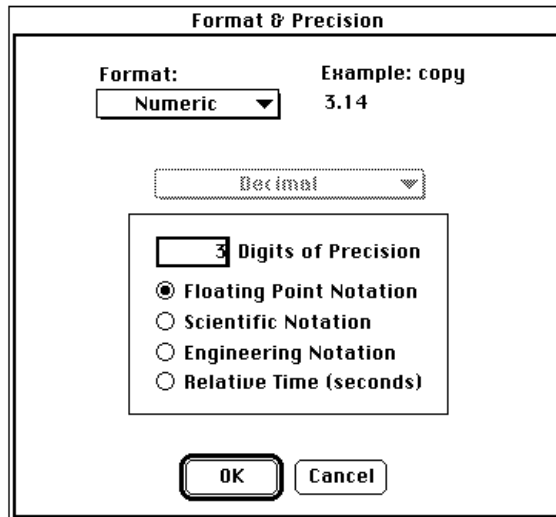
Modifizierung des numerischen Formats

LabVIEW zeigt die numerischen Controls standardmäßig als Dezimalzahlen mit zwei Dezimalstellen an (zum Beispiel 3.14). Sie können die Option **Format & Precision...** eines Controls oder das Pop-up Menü eines Indicators verwenden, um die Genauigkeit zu ändern oder die numerischen Controls in wissenschaftlicher Schreibweise anzuzeigen. Sie können die Option **Format & Precision...** auch zur Anzeige von Zeit- und Datumsformaten für Ziffern verwenden.

- Ändern Sie die Genauigkeit im Indicator `Time to Match`.
 - Führen Sie auf dem digitalen Indicator `Time to Match` einen Pop-up durch, und wählen Sie **Format & Precision...** Sie

müssen sich im Front Panel befinden, um auf das Menü zugreifen zu können.

- b. Geben Sie 3 in “Digits of Precision” ein, und klicken Sie auf **OK**.



2. Ändern Sie die Darstellung des digitalen Controls und zwei der digitalen Indicators in Ganzzahlen vom Typ Long.
 - a. Führen Sie auf dem digitalen Control “Number to Match” einen Pop-up durch, und wählen Sie **Representation»Long**.
 - b. Wiederholen Sie den vorherigen Schritt für die digitalen Indicators “Current Number” und “# of iterations digital”.

Einstellung des Datenbereichs

Sie können mit der Option **Data Range...** einen Benutzer daran hindern, einen Control- oder Indicator-Wert außerhalb eines bestimmten Bereichs oder Inkrements einzustellen. Sie können den Wert ignorieren, ihn in den festgelegten Bereich zwingen oder die Ausführung aussetzen. Anstelle der Taste “Run” erscheint das Bereichsfehler-Symbol auf dem Toolbar, wenn die Ausführung durch einen Bereichsfehler ausgesetzt wird. Außerdem wird das Control, das sich außerhalb des Bereichs befindet, von einer dunklen Umrandung eingefasst.



1. Stellen Sie den Datenbereich zwischen 0 und 100 mit einem Inkrement von 1 ein.
 - a. Führen Sie auf dem Indicator “Time to Match” einen Pop-up durch, und wählen Sie **Data Range...**
 - b. Füllen Sie das Dialogfenster wie in der folgenden Abbildung dargestellt aus, und klicken Sie auf **OK**.

Representation

Long

If Value is Out of Range:

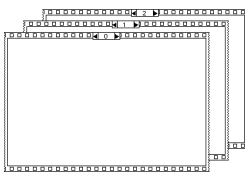
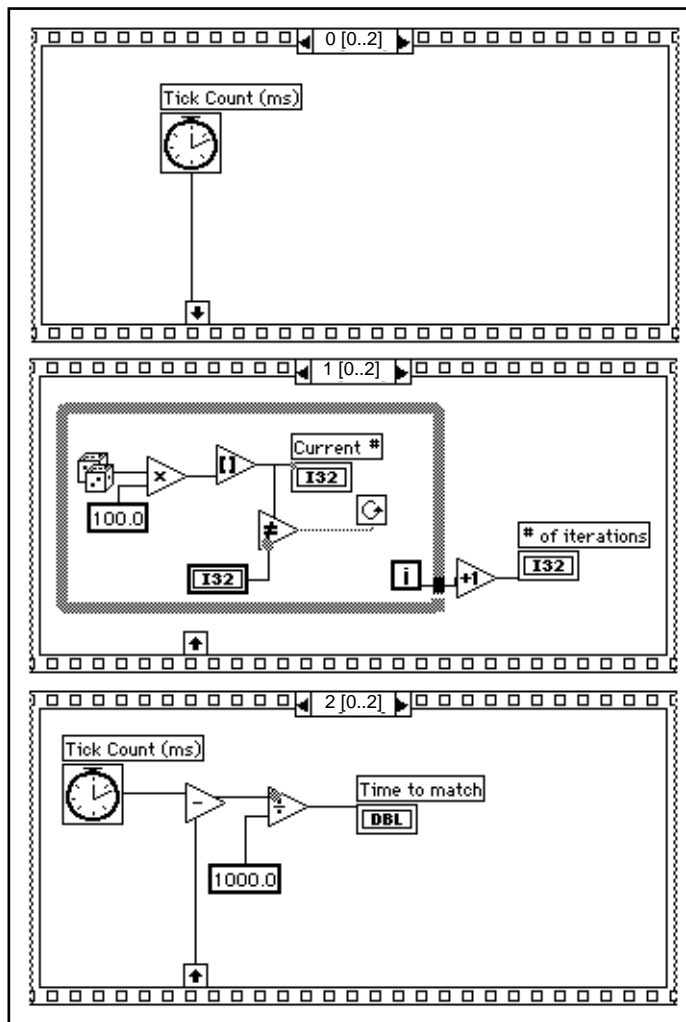
Minimum

Maximum

Increment

Default

Blockdiagramm

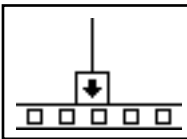


1. Öffnen Sie das Blockdiagramm-Fenster.
2. Plazieren Sie die Sequence-Struktur (**Functions»Structures**) auf dem Blockdiagramm-Fenster.

Die Sequence-Struktur, die wie Einzelbilder (Frames) eines Films aussieht, führt Blockdiagramme sequentiell aus. Bei konventionellen

Programmiersprachen werden die Programmanweisungen in der Reihenfolge ausgeführt, in der sie erscheinen. Bei Datenflußprogrammierung wird ein Node dann ausgeführt, wenn Daten an allen Node-Eingaben vorhanden sind, obwohl manchmal ein Node vor einem anderen ausgeführt werden muß. LabVIEW verwendet die Sequence-Struktur zur Steuerung der Reihenfolge, in der Nodes ausgeführt werden. LabVIEW plaziert das Diagramm, das als erstes vom VI ausgeführt wird, innerhalb von Frame 0, das Diagramm, das als zweites ausgeführt wird, in Frame 1 usw. Wie bei der Case-Struktur ist nur jeweils ein Frame auf einmal sichtbar.

3. Vergrößern Sie die Struktur, indem Sie eine Ecke mit dem Resizing Cursor ziehen.
4. Erstellen Sie ein neues Frame, indem Sie auf der Frame-Umrandung einen Pop-up durchführen und **Add Frame After** wählen. Wiederholen Sie diesen Schritt zur Erstellung von Frame 2.



Frame 0 in der vorherigen Abbildung enthält ein kleines Viereck mit einem Pfeil. Dieses Viereck ist eine Sequence Local Variable, die Daten zwischen den Frames einer Sequence-Struktur weiterleitet. Sie können auf der Umrandung eines Frames Sequence Locals erstellen. Die mit dem Sequence Local eines Frames verdrahteten Daten sind dann in nachfolgenden Frames verfügbar. Sie können jedoch nicht auf Daten zugreifen, die sich in Frames befinden, die dem Frame vorausgehen, in dem Sie das Sequence Local erstellt haben.

5. Erstellen Sie ein Sequence Local, indem Sie auf der unteren Umrandung von Frame 0 einen Pop-up durchführen und **Add Sequence Local** wählen.

Das Sequence Local erscheint als leeres Viereck. Der Pfeil im Viereck erscheint automatisch, wenn Sie eine Funktion mit dem Sequence Local verdrahten.

6. Vollenden Sie das Blockdiagramm, wie in der ersten Abbildung des Abschnitts *Blockdiagramm* dargestellt.



Funktion "Tick Count (ms)" (**Functions»Time & Dialog**). Gibt die Anzahl der Millisekunden an, die seit dem Einschalten verstrichen sind. Für dieses Übungsbeispiel brauchen Sie diese Funktion zweimal.



Funktion "Random Number (0-1)" (**Functions»Numeric**). Gibt eine Zufallszahl zwischen 0 und 1 aus.



Funktion “Multiply” (**Functions»Numeric**). In diesem Übungsbeispiel multipliziert die Funktion die Zufallszahl mit 100. Anders ausgedrückt: die Funktion gibt eine Zufallszahl zwischen 0.0 und 100.0 aus.



Funktion “Numeric Constant” (**Functions»Numeric**). In diesem Übungsbeispiel stellt die numerische Konstante die Maximalzahl dar, mit der multipliziert werden kann.



Funktion “Round to Nearest” (**Functions»Numeric**). In diesem Übungsbeispiel rundet die Funktion die Zufallszahl zwischen 0 und 100 auf die nächste Ganzzahl.



Funktion “Not Equal?” (**Functions»Comparison**). In diesem Übungsbeispiel vergleicht die Funktion die Zufallszahl mit der auf dem Front Panel angegebenen Zahl und gibt TRUE aus, wenn die Zahlen nicht gleich sind. Andernfalls gibt die Funktion FALSE aus.



Funktion “Increment” (**Functions»Numeric**). In diesem Übungsbeispiel inkrementiert die Funktion die While-Schleifen-Zählung um 1.



Funktion “Subtract” (**Functions»Numeric**). In diesem Übungsbeispiel gibt die Funktion die zwischen Frame 2 und Frame 0 verstrichene Zeit (in Millisekunden) aus.



Funktion “Divide” (**Functions»Numeric**). In diesem Übungsbeispiel dividiert die Funktion die Anzahl der verstrichenen Millisekunden durch 1000, d.h. sie rechnet sie in Sekunden um.



Numerische Konstante (**Functions»Numeric**). In diesem Übungsbeispiel rechnet die Funktion Millisekunden in Sekunden um.

In Frame 0 gibt die Funktion “Tick Count (ms)” die aktuelle Zeit in Millisekunden aus. Dieser Wert wird mit dem Sequence Local verdrahtet, wo der Wert in nachfolgenden Frames verfügbar ist. Das VI führt bei Frame 1 die While-Schleife so lange aus, wie die angegebene Zahl nicht mit der Zahl, die von der Funktion “Random Number (0-1)” ausgegeben wird, übereinstimmt. Die Funktion “Tick Count (ms)” gibt eine neue Zeit in Millisekunden aus. Das VI subtrahiert die alte Zeit (von Frame 1 durch das Sequence Local geleitet) von der neuen Zeit und berechnet so die verstrichene Zeit.

7. Kehren Sie zum Front Panel zurück, geben Sie im Control “Number to Match” eine Zahl ein, und führen Sie das VI aus.

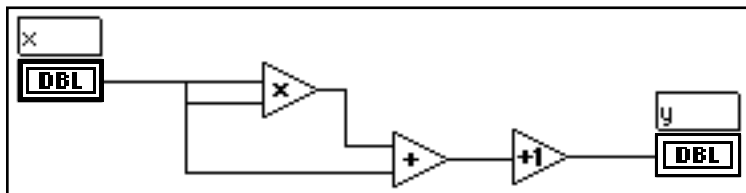
8. Sichern und schließen Sie das VI. Nennen Sie es My Time to Match.vi.

Formula Node

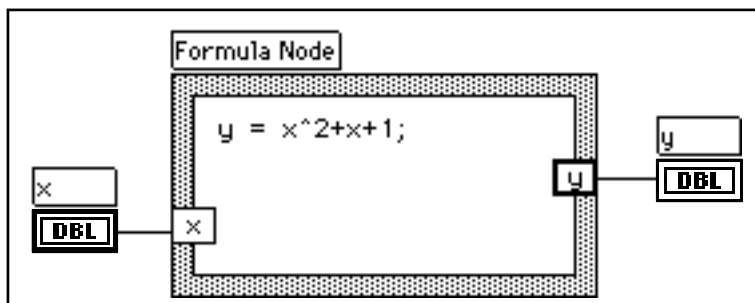
Der *Formula Node* ist ein umformbares Viereck, das Sie zur direkten Eingabe von Formeln in ein Blockdiagramm verwenden können. Sie platzieren den Formula Node auf dem Blockdiagramm, indem Sie ihn aus **Function»Structures** wählen. Es sollte dann verwendet werden, wenn eine Gleichung viele Variablen hat oder anderweitig kompliziert ist. Sehen Sie sich zum Beispiel folgende Gleichung an:

$$y = x^2 + x + 1.$$

Wenn Sie diese Gleichung anhand der normalen Arithmetikfunktionen von LabVIEW implementieren, sieht das wie folgt aus.



Sie können dieselbe Gleichung auch durch ein Formula Node implementieren (siehe folgende Abbildung).



Mit dem Formula Node können Sie komplizierte Formeln direkt eingeben, anstatt Blockdiagramm-Unterabschnitte zu erstellen. Sie geben diese Formeln mit dem Labeling Tool ein. Sie erstellen die Eingabe- und Ausgabe-Terminals des Formula Nodes, indem Sie auf der Umrandung des Nodes einen Pop-up durchführen und **Add Input**

(**Add Output**) wählen. Geben Sie den Variablennamen in das Viereck ein. Variablen reagieren auf Groß- und Kleinschreibung. Sie geben die Formel(n) innerhalb des Vierecks ein. Jede Formelanweisung muß mit einem Semikolon (;) abgeschlossen werden.

Die innerhalb des Formula Nodes verfügbaren Operationszeichen und Funktionen sind im Fenster "Help" des Formula Nodes aufgelistet (siehe folgende Abbildung). Ein Semikolon beendet jede Formelanweisung.

```

Formula Node operators, lowest precedence first:
=                assignment
? :             conditional
|| &&           logical
== != > < >= <= relational
+ - * / ^       arithmetic
+ - !           unary

Formula Node functions:
abs acos acosh asin asinh atan atanh ceil
cos cosh cot csc exp expm1 floor getexp getman
int intrz ln lnp1 log log2 max min mod rand
rem sec sign sin sinc sinh sqrt tan tanh
    
```

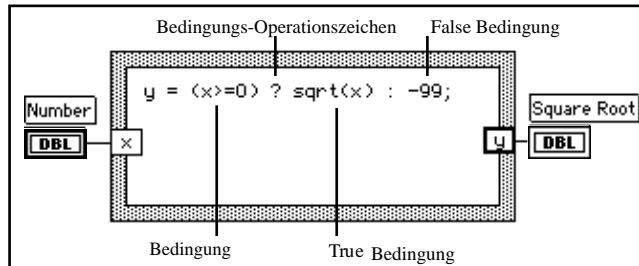
Folgendes Beispiel zeigt, wie Sie innerhalb des Formula Nodes eine bedingte Zuordnung durchführen können.

Sehen Sie sich das folgende Code-Fragment an, das die Quadratwurzel von x berechnet, wenn x positiv ist, und das Ergebnis y zuordnet. Ist x negativ, ordnet der Code y -99 zu.

```

if (x >= 0) then
y = sqrt(x)
else
y = -99
end if
    
```

Sie können das Code-Fragment mit einem Formula Node implementieren (siehe folgendes Diagramm),



Verwendung des Formula Node

ZIEL

Sie werden ein VI konstruieren, das den Formula Node zur Berechnung folgender Gleichungen verwendet.

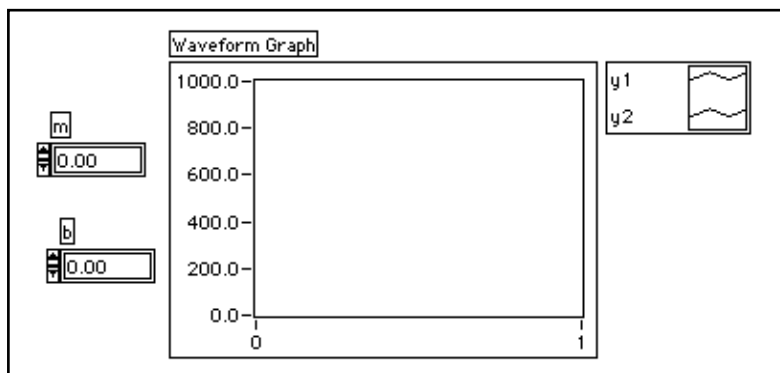
$$y1 = x^3 - x^2 + 5$$

$$y2 = m * x + b$$

wobei x zwischen 0 und 10 liegt.

Sie werden für beide Gleichungen nur einen Formula Node verwenden und die Ergebnisse auf demselben Graph darstellen.

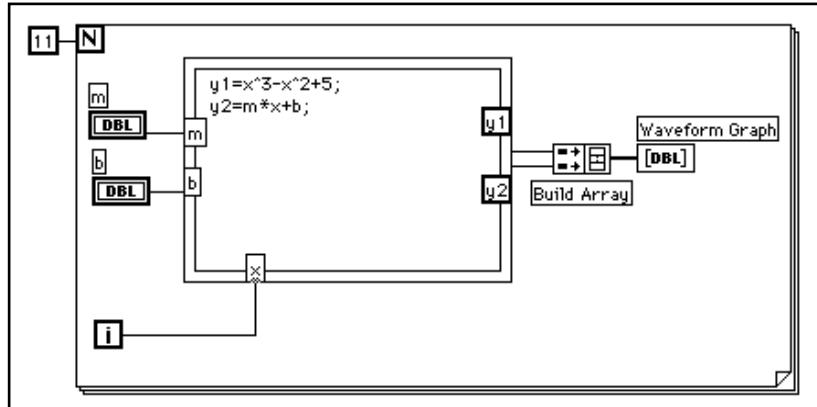
Front Panel



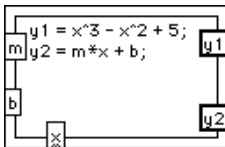
1. Öffnen Sie ein neues Front Panel, und gestalten es wie oben abgebildet. Der Indicator "Waveform Graph" zeigt die Plots der

Gleichung an. Das VI verwendet zur Eingabe der Werte für m und b die beiden digitalen Controls.

Blockdiagramm



1. Konstruieren Sie das in der vorherigen Abbildung dargestellte Blockdiagramm.
2. Plazieren Sie die For-Schleife (**Functions»Structures**) im Blockdiagramm-Fenster, und ziehen Sie die Ecke, um die Schleife zu vergrößern.



Formula Node (**Functions»Structures**). Sie können mit diesem Node direkt Formeln eingeben. Erstellen Sie die drei Eingabe-Terminals, indem Sie auf der Umrandung einen Pop-up durchführen und **Add Input** wählen. Sie erstellen das Ausgabe-Terminal, indem Sie **Add Output** aus dem Pop-up Menü wählen.

Wenn Sie ein Eingabe- oder Ausgabe-Terminal erstellen, müssen Sie es mit einem Variablennamen versehen. Dieser Variablenname muß genau mit dem von Ihnen in der Formel verwendeten übereinstimmen. Die Namen reagieren auf Groß- und Kleinschreibung. Das heißt, wenn Sie bei der Benennung des Terminals ein kleines a eingeben, müssen Sie auch bei der Formel ein kleines a eingeben. Sie können diese Variablennamen und Formeln mit dem Labeling Tool eingeben.



Hinweis: *Obwohl Variablennamen nicht in ihrer Länge beschränkt sind, beachten Sie, daß lange Namen viel Platz im Diagramm einnehmen. Ein Semikolon (;) beendet die Formelanweisung.*



Numerische Konstante (**Functions»Numeric**). Sie können auch einen Pop-up auf dem Zähl-Terminal durchführen und **Create Constant** wählen, um so automatisch die numerische Konstante zu erstellen und zu verdrahten. Die numerische Konstante gibt die Anzahl der For-Schleifen-Wiederholungen an. Liegt der x-Bereich zwischen 0 und 10, einschließlich 10, müssen Sie 11 mit dem Zähl-Terminal verdrahten.



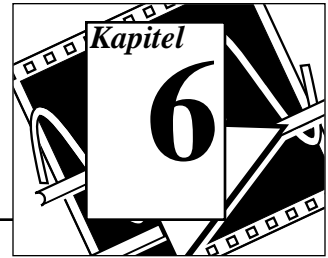
Da das Iteration-Terminal von 0 bis 10 zählt, verwenden Sie es zur Steuerung des X-Werts im Formula Node.



Die Funktion “Build Array” (**Functions»Array**) fügt zwei Arrays zu einem Multiplot-Graph zusammen. Erstellen Sie die zwei Eingangsterminals, indem Sie mit dem Resizing Cursor an einer der Ecken ziehen.

3. Kehren Sie zum Front Panel zurück, und führen Sie die verschiedenen Werte für m und b aus.
4. Schließen und sichern Sie das VI. Nennen Sie das VI My Equations.vi.

Strings und File I/O



Sie werden lernen:

- Wie man String Controls und Indicators erstellt.
- Wie man String-Funktionen verwendet.
- Näheres zu Datei-Eingabe- und -Ausgabe-Operationen.
- Wie man Dateien im Tabellenkalkulationsformat speichert.
- Wie man Daten in Textdateien schreibt und aus ihnen liest.

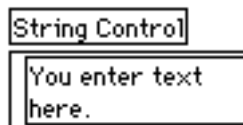
Strings

Ein String ist eine Sammlung von ASCII-Zeichen. Sie können Strings für mehr als nur einfache Texteingaben einsetzen. Bei der Gerätesteuerung können Sie numerische Daten als Zeichen-Strings weiterleiten und dann diese Strings in Zahlen umsetzen. Das Speichern numerischer Daten auf Platte kann ebenfalls unter Verwendung von Strings geschehen. Zum Speichern von Zahlen in einer ASCII-Datei müssen Sie zuerst Zahlen in Strings umsetzen, bevor Sie die Zahlen in eine Festplattendatei schreiben.

Erstellung von String Controls und Indicators

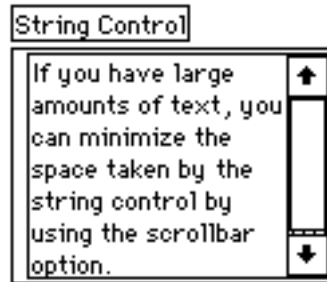


Sie finden String Controls und Indicators (siehe linke Spalte) unter **Controls»String & Table**. Sie können innerhalb eines String Controls Text eingeben oder ändern, indem Sie das Operating Tool oder das Labeling Tool verwenden. Vergrößern Sie String Controls und Indicators, indem Sie eine Ecke mit dem Positioning Tool ziehen.



Strings und File I/O

Wenn Sie den von einem String Control oder Indicator auf dem Front Panel beanspruchten Platz minimieren wollen, wählen Sie **Show»Scrollbar**. Ist diese Option schwach dargestellt, müssen Sie das Fenster in vertikaler Richtung vergrößern, damit sie verfügbar wird.

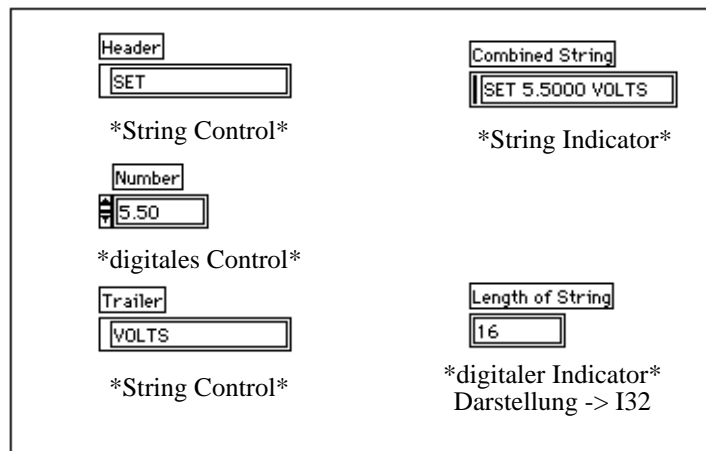


Verwendung von String-Funktionen

ZIEL

In LabVIEW gibt es vielfältige String-Bearbeitungsfunktionen, die Sie unter **Functions»String** finden. Sie werden ein VI erzeugen, das eine Zahl in einen String konvertiert, der wiederum mit anderen Strings verkettet wird und somit einen gemeinsamen Ausgangs-String bildet. Das VI ermittelt außerdem die Gesamtlänge des Strings.

Front Panel

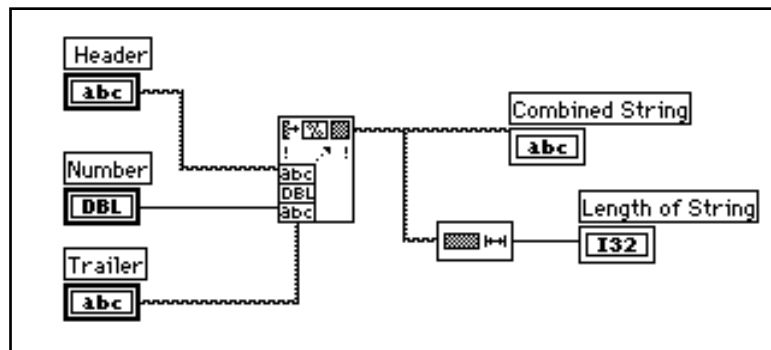


1. Öffnen Sie ein neues Front Panel, und konstruieren Sie das in der vorherigen Abbildung dargestellte Front Panel. Stellen Sie sicher, daß Sie die Controls und Indicators wie dargestellt modifizieren.

Die beiden String Controls und das digitale Control können in einem Ausgabe-String kombiniert und im String Indicator angezeigt werden. Der digitale Indicator zeigt die Länge des Strings an.

Die Ausgabe “Combined String” in diesem Übungsbeispiel hat ein Format, das Befehls-Strings ähnlich ist, die zur Kommunikation mit GPIB (IEEE 488) und seriellen (RS-232 oder RS-422) Geräten eingesetzt werden. Siehe Kapitel 7, *Datenerfassung und Gerätesteuerung*, dieses Demonstrations, um mehr über für Gerätebefehle verwendete Strings zu erfahren.

Blockdiagramm



1. Konstruieren Sie das in der vorherigen Abbildung dargestellte Blockdiagramm.



Die Funktion “Format Into String” (**Functions»String**) verkettet und formatiert Zahlen und Strings zu einem einzigen Ausgabe-String. Verwenden Sie den Resizing Cursor am Icon, um drei Argument-Eingaben hinzuzufügen.



Die Funktion “String Length” (**Functions»String**) gibt die Anzahl der Zeichen in dem verketteten String aus.

2. Führen Sie das VI aus. Beachten Sie, daß die Funktion “Format Into String” die beiden String Controls und das digitale Control zu einem einzigen Ausgabe-String verkettet.

3. Sichern Sie das VI als `My Build String.vi`. Sie werden dieses VI im nächsten Übungsbeispiel verwenden.

File I/O (Datei Ein-/Ausgabe)

LabVIEW enthält einen leistungsstarken und flexiblen Satz an File I/O Funktionen. Damit sind Sie in der Lage, neben Durchführung allgemeiner Lese- und Schreiboperationen, auch Dateien bewegen und umbenennen zu können sowie Verzeichnisstrukturen zu erstellen. Sie können aus lesbarem ASCII-Text Tabellenkalkulationsdateien erzeugen und zwecks Kompaktheit und Schnelligkeit Dateien im Binärformat erzeugen und verarbeiten.

Sie können Daten in drei verschiedenen Formaten speichern oder lesen.

- ASCII Bytestream-Files. Sie sollten Daten im ASCII-Format speichern, wenn Sie diese mit einem anderen Softwareprogramm, zum Beispiel einem Textverarbeitungs- oder Tabellenkalkulationsprogramm, weiterverarbeiten wollen. Zum Speichern von Daten in diesem Format müssen Sie alle Daten in ASCII-Strings umsetzen.
- Datalog Files. Auf Dateien in diesem Binärformat kann nur mit LabVIEW zugegriffen werden. Datalog Files sind Datenbankdateien ähnlich, da Sie verschiedene Datentypen in einem (Log) Record einer Datei speichern können.
- Binäre Bytestream-Files. Diese Files stellen die kompakteste und schnellste Methode zur Datenspeicherung dar. Sie müssen die Daten in das binäre String Format umsetzen und genau wissen, welche Datentypen Sie verwenden, um Daten in Dateien zu speichern oder aus ihnen zurückzuholen.

Dieser Abschnitt erläutert ASCII Bytestream-Files, da sie das gebräuchlichste Daten-File-Format darstellen.

File I/O Funktionen

Die meisten File I/O Operationen umfassen drei grundlegende Schritte: Öffnen einer vorhandenen Datei (File) oder Erstellen einer neuen Datei; Schreiben in oder Lesen aus der Datei; und Schließen der Datei. LabVIEW enthält daher in **Functions»File I/O** eine Vielzahl von Utilities, die im folgenden näher beschrieben werden. Diese

Dienstprogramm-Funktionen sind auf VIs auf Zwischenebene aufgebaut, die eine Fehlerüberprüfung und -bearbeitung mit File I/O-Funktionen einschließen.

Sie können diesen Utilities den zu verwendenden Zeichentrenner (Tabulator, Komma usw.) vorschreiben. Dies ist dann hilfreich, wenn Ihre Tabellenkalkulationsdatei einen Zeichentrenner verwendet, der vom Standard-Tabulator abweicht.



Das VI “Write Characters To File” schreibt einen Zeichen-String in ein neues Bytestream-File oder fügt den String einem vorhandenen Bytestream-File hinzu. Dieses VI öffnet oder erstellt die Datei, schreibt die Daten und schließt dann die Datei.



Das VI “Read Characters From File” liest, bei einem angegebenen Zeichen-Offset beginnend, eine angegebene Anzahl von Zeichen aus einem Bytestream-File. Dieses VI öffnet die Datei vorher und schließt sie hinterher.



Das VI “Read Lines From File” liest, bei einem angegebenen Zeichen-Offset beginnend, eine angegebene Anzahl von Zeilen aus einem Bytestream-File. Dieses VI öffnet die Datei vorher und schließt sie hinterher.



Das VI “Write To Spreadsheet File” setzt ein ein- oder zweidimensionales Array von Single-Precision Werten in einen Text-String um oder fügt den String einer vorhandenen Datei hinzu. Sie können die Daten, falls gewünscht, auch austauschen. Dieses VI öffnet die Datei vorher und schließt sie hinterher. Sie können dieses VI zum Erstellen von Textdateien verwenden, die in den meisten Tabellenkalkulationsprogrammen gelesen werden können.



Das VI “Read From Spreadsheet File” liest eine angegebene Anzahl von Zeilen oder Reihen aus einer numerischen Textdatei, wobei es bei einem angegebenen Zeichen-Offset beginnt, und setzt die Daten in ein zweidimensionales Array von Single-Precision Werten um. Sie können die Daten, falls gewünscht, auch austauschen. Dieses VI öffnet die Datei vorher und schließt sie hinterher. Sie können dieses VI zum Lesen von Tabellenkalkulationsprogrammen verwenden, die in Textformaten gespeichert sind.

Eine Tabellenkalkulationsdatei schreiben

Das Sichern von Daten in einer Datei wird sehr häufig dazu verwendet, die Textdatei so zu formatieren, daß sie in einer Tabellenkalkulation geöffnet werden kann. In den meisten Tabellenkalkulationen werden Spalten durch Tabulatoren und Zeilen durch EOL-Zeichen (Zeilenende-Zeichen) getrennt (siehe folgende Abbildung).

0.00	↠	0.4258	¶	↠ = Tabulator
1.00	↠	0.3073	¶	¶ = Zeilentrenner
2.00	↠	0.9453	¶	
3.00	↠	0.9640	¶	
4.00	↠	0.9517	¶	

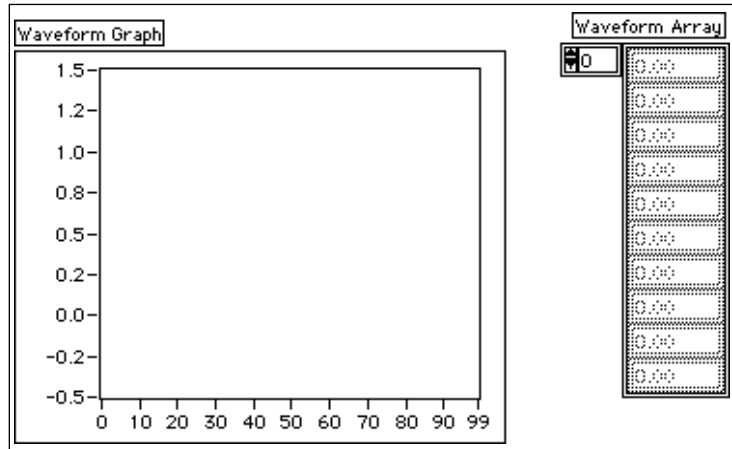
Wird die Datei in einem Tabellenkalkulationsprogramm geöffnet, erhält man folgende Tabelle.

	A	B	C
1	0	0.4258	
2	1	0.3073	
3	2	0.9453	
4	3	0.964	
5	4	0.9517	
6			

ZIEL

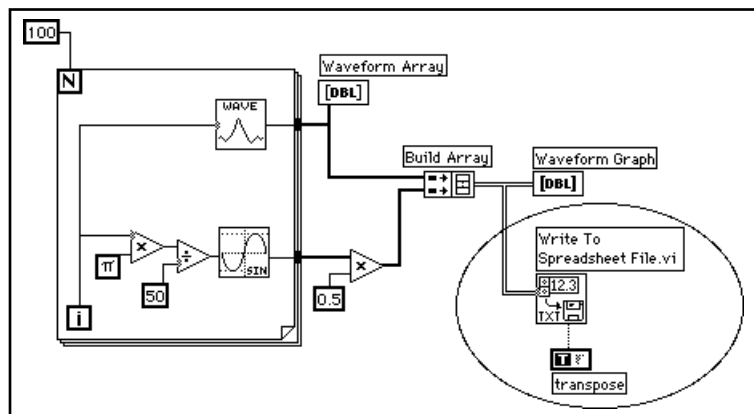
Sie werden ein vorhandenes VI so modifizieren, daß es eine File I/O-Funktion verwendet, mit der Sie Daten in einer neuen Datei im ASCII-Format sichern können. Sie können dann später auf diese Datei von einer Tabellenkalkulationsanwendung aus zugreifen.

Front Panel



- Öffnen Sie das VI `My Graph Waveform Arrays.vi`, das Sie in Kapitel 4 dieser Demonstrations erstellt haben. Wie Sie bereits wissen, generiert dieses VI zwei Daten-Arrays und stellt diese auf einem Graph dar. Sie modifizieren dieses VI so, daß es zwei Arrays in eine Datei schreibt, wobei jede Spalte ein Daten-Array enthält.

Blockdiagramm



2. Öffnen Sie das Blockdiagramm von My Graph Waveform Arrays, und modifizieren Sie das VI, indem Sie die in der vorherigen Abbildung im Oval dargestellten Blockdiagramm-Funktionen hinzufügen.



Das VI “Write To Spreadsheet File” (**Functions»File I/O**) setzt das zweidimensionale Array in einen Tabellenkalkulations-String um und schreibt es in eine Datei. Wenn Sie keinen Pfadnamen angegeben haben, erscheint ein Dialogfenster und fordert Sie zur Eingabe eines Dateinamens auf. “Write To Spreadsheet File” schreibt entweder ein ein- oder zweidimensionales Array in die Datei. Da Sie in diesem Beispiel mit einem zweidimensionalen Daten-Array arbeiten, brauchen Sie die eindimensionale Eingabe nicht zu verdrahten. Sie können bei diesem VI einen Zeichentrenner, zum Beispiel Tabulatoren oder Kommata, in Ihren Daten verwenden.



Die Boolesche Konstante (**Functions»Boolean**) steuert, ob LabVIEW das zweidimensionale Array transponiert, bevor es in die Datei geschrieben wird. Klicken Sie mit dem Operating Tool auf die Konstante, um den Wert in TRUE zu ändern. In diesem Fall müssen Sie die Daten transponieren, da die Daten-Arrays zeilenspezifisch sind (jede Zeile des zweidimensionalen Arrays ist ein Daten-Array). Da jede Spalte der Tabellenkalkulationsdatei ein Daten-Array enthält, muß das zweidimensionale Array zuerst transponiert werden.

3. Kehren Sie zum Front Panel zurück, und führen Sie das VI aus. Nach Generierung der Daten-Arrays erscheint ein Dialogfenster und fordert Sie zur Eingabe eines Dateinamens für die von Ihnen erstellte neue Datei auf. Geben Sie einen Dateinamen ein, und klicken Sie auf **OK**.



Vorsicht: *Versuchen Sie nicht, Daten in VI-Bibliotheken, zum Beispiel mywork.llb, zu schreiben, da Sie sonst eventuell Ihre Bibliothek überschreiben und Ihre frühere Arbeit löschen.*

4. Sichern Sie das VI, nennen Sie es My Waveform Arrays to File.vi, und schließen Sie das VI.
5. Sie können nun ein Tabellenkalkulations- oder Textverarbeitungsprogramm verwenden, um die soeben von Ihnen erstellte Datei zu öffnen und anzuschauen. Sie sollten zwei Spalten mit je 100 Elementen sehen.

In diesem Beispiel wurden die Daten erst nach kompletter Erfassung formatiert und in die Datei geschrieben. Wenn Sie große Datenpuffer

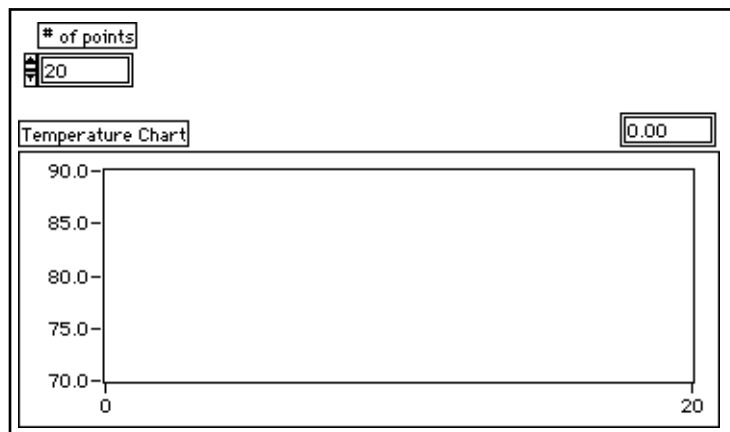
erfassen oder die Datenwerte auf die Festplatte schreiben wollen, während sie generiert werden, müssen Sie ein anderes Datei Ein-/Ausgabe-VI verwenden.

Einem File Daten hinzufügen

ZIEL

Sie werden ein VI erstellen, um einer Datei im ASCII-Format Temperaturdaten hinzuzufügen. Dieses VI verwendet eine For-Schleife, um Temperaturwerte zu generieren und sie in einer Datei zu speichern. Sie werden während jeder Iteration die Daten in einem String umsetzen, ein Komma als Zeichentrenner hinzufügen und die Daten einer Datei hinzufügen.

Front Panel



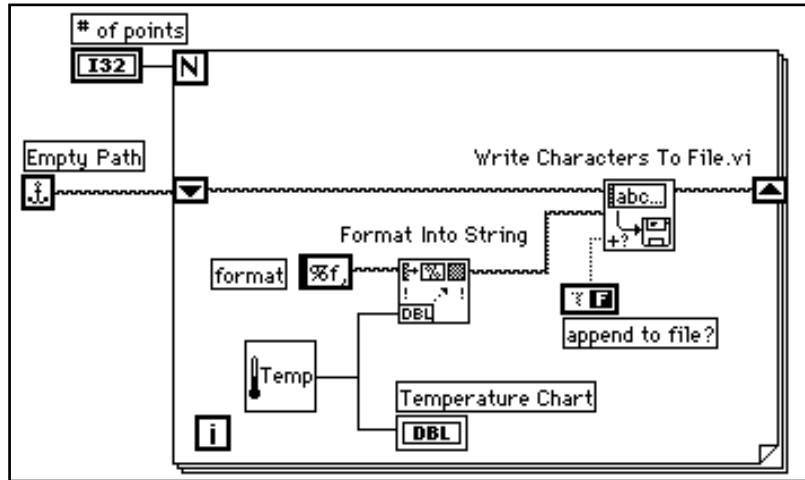
1. Öffnen Sie ein neues Front Panel, und plazieren Sie die Objekte wie in der vorherigen Abbildung dargestellt.

Das Front Panel enthält ein digitales Control und einen Waveform-Chart. Wählen Sie **Show»Digital Display**. Das Control # of points gibt an, wie viele Temperaturwerte erfasst und in die Datei geschrieben werden müssen. Das Chart zeigt die Temperaturkurve an. Skalieren Sie die y-Achse des Charts auf den Bereich 70.0 bis 90.0 um, und skalieren Sie die x-Achse des Charts auf den Bereich 0 bis 20 um.

2. Führen Sie auf dem digitalen Control “# of points” einen Pop-up durch, und wählen Sie **Representation»Long**.



Blockdiagramm



1. Öffnen Sie das Blockdiagramm.
2. Fügen Sie die For-Schleife hinzu, und vergrößern Sie sie. Das VI generiert die im Control “# of Points” angegebene Anzahl an Temperaturwerten.
3. Fügen Sie der Schleife ein Shift Register hinzu, indem Sie auf der Schleifen-Umrandung einen Pop-up durchführen. Dieses Shift Register enthält den Pfadnamen der Datei.
4. Beenden Sie das Verdrahten der Objekte.



Empty Path Constant (**Functions»File I/O»File Constants**). Die Funktion “Empty Path” initialisiert das Shift Register, so daß der Pfad leer ist, wenn Sie zum ersten Mal versuchen, einen Wert in die Datei zu schreiben. Ein Datei-Dialogfenster fordert Sie zur Eingabe eines Dateinamens auf.



Das VI “My Thermometer”, das Sie in Kapitel 2 konstruiert haben, (**Functions»Select a VI...**) oder das VI “Digital Thermometer” (**Functions»Tutorial**) gibt eine simulierte Temperaturmessung von einem Temperatursensor aus.



Die Funktion “Format Into String” (**Functions»String**) setzt die Temperaturmessung (eine Zahl) in einen String um und verkettet das darauffolgende Komma.



String Constant (**Functions»String**). Dieser Format String gibt an, daß Sie die Zahl in einen Fractional Format String umsetzen und nach dem String ein Komma einfügen wollen.



Das VI “Write Characters To File” (**Functions»File I/O**) schreibt einen Zeichen-String in eine Datei.



Die Boolesche Konstante (**Functions»Boolean**) stellt die Eingabe “append to file?” des VIs “Write Characters To File” auf TRUE ein, so daß die neuen Temperaturwerte der gewählten Datei hinzugefügt werden, während die Schleife wiederholt wird. Klicken Sie mit dem Operating Tool auf das Constant, um seinen Wert auf TRUE einzustellen.

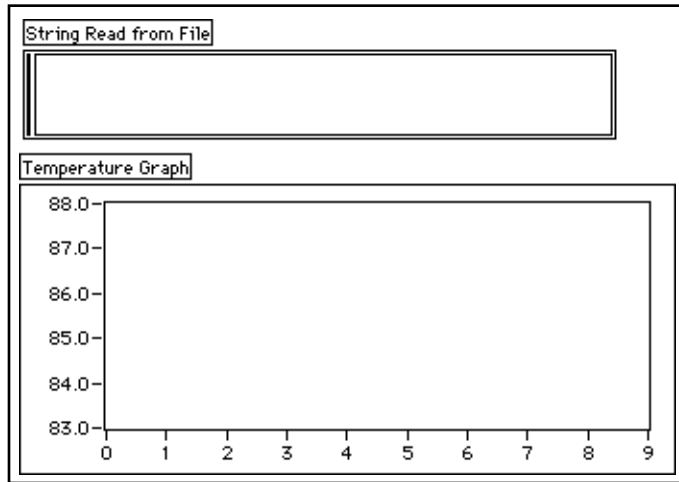
5. Kehren Sie zum Front Panel zurück und führen Sie das VI aus, wobei “# of points” auf 20 eingestellt ist. Ein Datei-Dialogfenster fordert Sie zur Eingabe eines Dateinamens auf. Wenn Sie einen Dateinamen eingeben, beginnt das VI mit dem Schreiben der Temperaturwerte in die Datei, während gleichzeitig die einzelnen Punkte generiert werden.
6. Sichern Sie das VI, nennen Sie es My Write Temperature to File.vi, und schließen Sie das VI.
7. Verwenden Sie eine beliebige Textverarbeitungs-Software, zum Beispiel Write for Windows, Teach Text for Macintosh und Text Editor in Open Windows for UNIX, um die Datendatei zu öffnen und ihren Inhalt anzuschauen. Sie sollten eine Datei mit zwanzig, durch Kommata getrennten Datenwerten erhalten (mit einer Genauigkeit von drei Stellen nach dem Dezimalpunkt).

Daten aus einem File lesen

ZIEL

Sie werden ein VI erstellen, das die Daten-Datei, die Sie im vorherigen Beispiel geschrieben haben, liest und die Daten auf einem Waveform Graph anzeigt. Sie müssen die Daten in demselben Format lesen, in dem Sie sie gesichert haben. Da Sie die Daten ursprünglich im ASCII-Format anhand von String-Datentypen gesichert haben, müssen Sie sie nun mit einem der Datei Ein-/Ausgabe-VIs als String-Daten lesen.

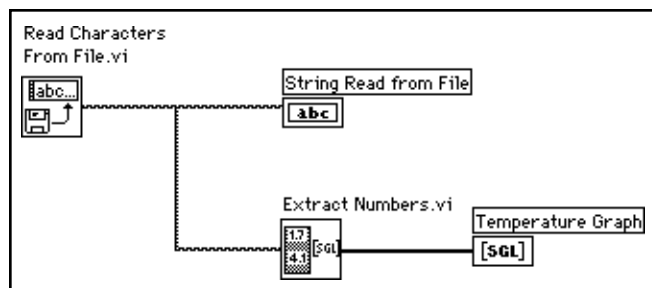
Front Panel



1. Öffnen Sie ein neues Front Panel, und konstruieren Sie das in der Abbildung dargestellte Front Panel.

Das Front Panel enthält einen String Indicator und einen Waveform-Graph. Der Indicator "String Read from File" zeigt die durch Kommata begrenzte Temperaturdaten der Datei an, die Sie im letzten Beispiel erstellt haben. Der Graph zeigt die Temperaturkurve an.

Blockdiagramm



1. Konstruieren Sie das in der vorherigen Abbildung dargestellte Blockdiagramm.



Das VI “Read Characters From File” (**Functions»File I/O**) liest die Daten aus der Datei und gibt die Informationen in einem String aus. Ist kein Pfadname angegeben, fordert ein Datei-Dialogfenster zur Eingabe eines Dateinamens auf. In diesem Beispiel brauchen Sie die Anzahl der zu lesenden Zeichen nicht zu bestimmen, da weniger als die standardmäßigen Zeichen (512) in der Datei vorhanden sind.

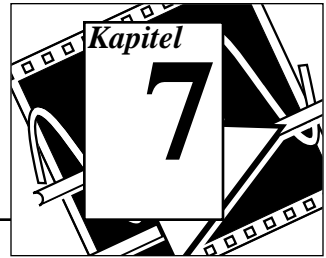
Sie müssen wissen, wie die Daten in einer Datei gespeichert wurden, damit die Daten wieder gelesen werden können. Wenn Sie wissen, wie lang eine Datei ist, können Sie das VI “Read Characters From File” verwenden, um die bekannte Anzahl von zu lesenden Zeichen festzulegen.



Das VI “Extract Numbers” (`examples\general\strings.llb`) verwendet einen ASCII-String mit durch Kommata, Zeilenvorschüben oder anderen nicht-numerischen Zeichen getrennten Zahlen und setzt sie in ein Ziffern-Array um.

2. Kehren Sie zum Front Panel zurück, und führen Sie das VI aus. Wählen Sie die soeben von Ihnen auf Festplatte geschriebene Daten-Datei, wenn Sie das Datei-Dialogfenster dazu auffordert. Sie sollten dieselben Werte auf dem Graph angezeigt sehen, die Sie beim Beispiel-VI “My Write Temperature to File” gesehen haben.
3. Sichern Sie das VI, nennen Sie es `My Temperature from File.vi`, und schließen Sie das VI.

Datenerfassung und Gerätesteuerung



Sie werden lernen:

- Wie man mit einer Datenerfassungskarte arbeitet (**Windows, Macintosh und Sun**).
- Näheres zu VISA-Funktionen.
- Näheres zu GPIB-Funktionen.
- Näheres zur Serial-Port-Kommunikation.
- Wie man einen Serial-Port zur Kommunikation mit anderen Serial-Ports verwendet.
- Wie man VXI zur Datenerfassung verwendet (**Windows, Macintosh und Sun**).
- Was ein Gerätetreiber ist.
- Wie man ein Frequency Response Test VI verwendet.

Verwendung von LabVIEW zur Datenerfassung

Eines der wertvollsten Leistungsmerkmale von LabVIEW ist seine Fähigkeit, Daten aus fast jeder Quelle zu erfassen. LabVIEW verfügt über VIs zur Steuerung folgender Geräte:

- Einsteckkarten für Datenerfassung (**Windows, Macintosh und Sun**)
- GPIB (IEEE 488)-Geräte
- Serial-Port-Geräte
- VXI-Geräte (**Windows, Macintosh und Sun**)

Diese VIs verwenden auf Treiberebene die dem Industriestandard entsprechende Software von National Instruments; sie bieten so eine vollkommene Steuerung Ihrer Datenerfassungs- und Gerätesteuerungshardware.

Dieses Demonstrations-/Lernhandbuch konzentriert sich auf die grundlegenden LabVIEW-Leistungsmerkmale und -Funktionen.

Näheres zu Einsteckkarten für Datenerfassung (Windows, Macintosh und Sun)

National Instruments fertigt alle Komponenten, die Sie zur Konstruktion eines vollständigen Datenerfassungssystems brauchen. Einsteckkarten sind für IBM PC/AT, EISA, IBM PS/2 MicroChannel, Macintosh PCI, Macintosh NuBus Series, Macintosh LC/LCII und SPARCstation SBus Computer erhältlich.

Diese Karten haben verschiedene Kombinationen von analogen, digitalen und Timing Eingaben und Ausgaben. Sie können Front-end SCXI-Signalkonditionierungs-Multiplexer verwenden, um die Anzahl der analogen Eingabekanäle kostengünstig zu erhöhen. Die Datenerfassungs-Hardwareprodukte werden durch eine Vielzahl von Signalkonditionierungsmodulen für Thermoelemente, Widerstandstemperaturfühler (RTD), Spannungs- und Stromeingaben sowie digitale Ein- und Ausgänge für hohe Ströme vervollständigt.

Die LabVIEW Datenerfassungs-(DAQ)-VIs steuern die gesamte Datenerfassungs-Hardware von National Instruments.

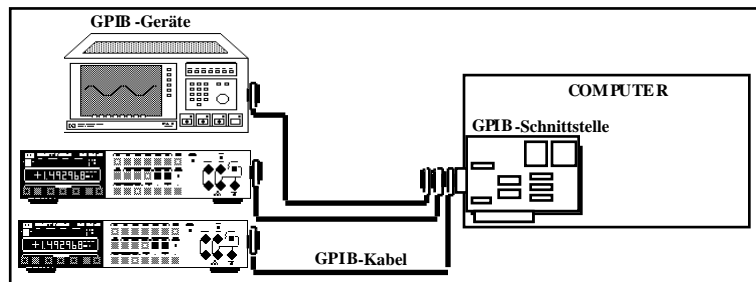
Näheres zu VISA

VISA ist eine Schnittstellen-Bibliothek zur Steuerung von GPIB, VXI und anderen Gerätetypen. Sie können mit den VISA-Funktionen ein einzelnes Gerätetreiber-VI konstruieren, das ein bestimmtes Gerätemodell über verschiedene I/O-Medien hinweg steuert. Ein String wird an die Funktion "VISA Open" übergeben, um den I/O-Typ, der zur Kommunikation mit den Geräten verwendet werden soll, zu wählen. Wenn eine Sitzung mit einem Gerät eröffnet ist, führen die VISA-Funktionen, zum Beispiel "VISA Read" und "VISA Write", die I/O-Aktivitäten des Geräts in allgemeiner Art durch. Das Programm ist daher nicht an bestimmte GPIB- oder VXI-Funktionen gebunden. Der VISA-Gerätetreiber ist schnittstellenunabhängig und kann bei mehreren verschiedenen Systemen eingesetzt werden.

Gerätetreiber, die VISA-Funktionen verwenden, erfassen die dem Gerät und nicht die dem Kommunikationsmedium spezifischen Aktivitäten. Dies kann weitere Möglichkeiten zur Wiederverwendung eines Gerätetreibers bei einer Anzahl von Programmen eröffnen.

Näheres zu GPIB

Der General Purpose Interface Bus (GPIB), auch IEEE 488 genannt, ist eine Methode zur Kommunikation mit unabhängigen Geräten, wie zum Beispiel Multimetern und Oszilloskopen. National Instruments fertigt viele Produkte zur Steuerung von Geräten mit dem GPIB. Die direkteste Methode besteht in der Installation einer GPIB-Einsteckkarte in Ihren Computer und dem Anschluß Ihres Gerätes mit einem GPIB-Kabel direkt an diese Karte.



Die LabVIEW GPIB-Funktionen steuern die GPIB-Schnittstellen von National Instruments. Der Abschnitt *GPIB Overview* in Kapitel 1, *Introduction*, im *LabVIEW Instrument I/O VI Reference Manual* enthält eine Beschreibung und Hintergrundinformationen zum GPIB bzw. IEEE 488. LabVIEW verwendet die National Instruments

NI-488.2 Standard-Treibersoftware, die mit Ihrer GPIB-Schnittstelle geliefert wird.

Die GPIB-Bibliothek (**Functions»Instrument I/O**) enthält sowohl die traditionellen GPIB-Funktionen als auch 488.2-Funktionen. Die GPIB 488.2-Funktionen geben LabVIEW IEEE 488.2-Kompatibilität. Diese Funktionen implementieren Aufrufe, die in der Norm IEEE 488.2 spezifiziert sind, und ähneln den Unterprogrammen der National Instruments NI-488.2 Software.

Hinweis: *Aufgrund der Vielseitigkeit von VISA sollten Sie anstelle von GPIB nach Möglichkeit VISA-Funktionen verwenden.*

Näheres zu Serial-Ports

Serielle Kommunikation ist eine gebräuchliche Methode der Datenübertragung zwischen einem Computer und einem Peripheriegerät, wie zum Beispiel einem Drucker, einem Plotter oder einem programmierbaren Gerät. Bei der seriellen Kommunikation wird ein Sender zur bitweisen Datenübertragung über eine einzelne Kommunikationsleitung an einen Empfänger eingesetzt. Diese Kommunikationsmethode ist bei der Übertragung von Daten mit niedrigen Übertragungsgeschwindigkeiten oder über große Entfernungen gebräuchlich. Serielle Daten können zum Beispiel mit Modems über normale Telefonleitungen übertragen werden.

Die serielle Kommunikation ist eine weit verbreitete Methode, da die meisten Computer einen oder zwei serielle Ports haben. Eine Beschränkung der seriellen Kommunikation besteht jedoch darin, daß über eine serielle Schnittstelle nur mit einem Gerät kommuniziert werden kann. Um mit mehreren Geräten kommunizieren zu können, müssen Sie eine Karte mit mehreren Serial-Ports oder eine Serial-Port-Multiplexeinrichtung haben.

Bevor Sie LabVIEW für die serielle Kommunikation verwenden, sollten Sie sicherstellen, daß das Gerät richtig an den Computer angeschlossen ist. Bei Windows müssen Sie außerdem sicherstellen, daß keine Interruptkonflikte vorhanden sind. Windows- und Macintosh-Benutzer können dies anhand eines allgemeinen Terminal-Softwareprogramms, z.B. Microsoft Windows Terminal oder ZTerm, feststellen. Nachdem Sie die Kommunikation mit einem Gerät hergestellt haben, können Sie die LabVIEW Serial-Port-VIs verwenden, die sich in **Functions»Instrument I/O»Serial** befinden.

LabVIEW enthält fünf VIs für serielle Kommunikation – “Serial Port Init”, “Serial Port Write”, “Serial Port Read”, “Bytes at Serial Port” und “Serial Port Break”.

Näheres zu VXI für Windows, Macintosh und Sun

Der VXI-Bus ist eine sehr stark wachsende Plattform für Instrumentierungssysteme. Ein VXI-System besteht aus einem Mainframe-Chassis mit max. 13 Slots, die modular und flexibel VXI-Meßgeräte (Instrument on the Card) aufnehmen. Für den VXI-Bus ist eine große Vielzahl an Instrumenten und Mainframes von unterschiedlichsten Herstellern verfügbar, wobei auch

unterschiedliche Gerätegrößen in demselben Mainframe betrieben werden können. Ein VXI-System kann auf verschiedene Arten gesteuert werden.

LabVIEW hat VXI-VIs zur Steuerung eines VXI-Systems auf höherer und niedriger Ebene. Sie greifen auf diese VIs von **Functions»Instrument I/O»VISA** aus zu.

Näheres zu Gerätetreibern

Ein Gerätetreiber ist eine Software, die ein bestimmtes Gerät steuert. LabVIEW ist ideal zur Erstellung von Gerätetreibern geeignet. Das LabVIEW Front Panel kann die Funktionsweise der Bedienoberfläche (Front Panel) eines Geräts simulieren. Das Blockdiagramm kann die erforderlichen Befehle an das Gerät senden, um die vom Front Panel angegebene Operation durchzuführen. Wenn Sie mit der Konstruktion eines Gerätetreibers fertig sind, brauchen Sie sich nicht mehr an die Befehle erinnern, die zur Steuerung des Geräts notwendig sind. Sie brauchen nur noch das Front Panel zu bedienen. Ein Software Panel zur Steuerung des Geräts hat an sich wenig Wert. Der eigentliche Wert besteht darin, daß Sie den Gerätetreiber als SubVI zusammen mit anderen SubVIs in einem größeren VI verwenden können, um ein gesamtes System zu steuern.

LabVIEW hat eine Bibliothek mit über 550 Gerätetreibern für GPIB, serielle, CAMAC und VXI (für Windows, Macintosh und Sun) Geräte. Da es viele verschiedene Gerätetypen gibt, ist es nicht möglich, die Techniken zur Erstellung von Treibern für alle Gerätetypen zu demonstrieren; bei einem nachrichten-basierenden (message-based) Gerät konstruieren jedoch alle Treiber einen *Befehls-String* und senden ihn an das Gerät, das dann die vom simulierten Front Panel angegebene Operation durchführt. Der Befehls-String besteht aus gerätespezifischen Befehlen (normalerweise in ASCII), die das Gerät fernsteuern. Die Gerätetreiber enthalten daher mehr Funktionen zur String-Bearbeitung als spezifische Schnittstellen-Befehle.