

LabVIEW

LabVIEW™ 評価版

10月1997版
製品番号 350396A-01



本書で使用される取決め

下記は本書で使用される取決めです。

太字 太字のテキストはメニュー名、パレット名、メニュー項目、ダイアログボックスのボタンやオプションを表わします。さらに、ヒントも表わします。

イタリック体

イタリック体のテキストは用語を表わします。

太字のイタリック体

太字のイタリック体のテキストは注意を表わします。

monospace (モノスペース)

このフォントで表わされているテキストや文字はキーボードから入力されるものです。又このフォントはディスクドライブ名、パス名、ディレクトリ名、プログラム名、ファイル名、拡張子名を表わすために使用されま
す。

著作権

著作権法に基づき、ナショナルインスツルメンツ社の書面による事前の許可なく、複写、記録、情報検索システムへの保存及び翻訳を含め、電子的であるか機械的であるかを問わず、いかなる形式であれ本書の一部あるいは全部を複製または伝送することを禁止します。

商標

LabVIEW[®]、NI-DAQ[®]、National Instruments[™] 及び [natinst.com](http://www.natinst.com)[™] はナショナルインスツルメンツ社の商標です。リストされた製品及び会社名は、それぞれ該当する会社の商標または商標名です。



インターネット

E-mail: info@nni.co.jp

FTP Site: [ftp.natinst.com](ftp://ftp.natinst.com)

Web Address: <http://www.natinst.com>

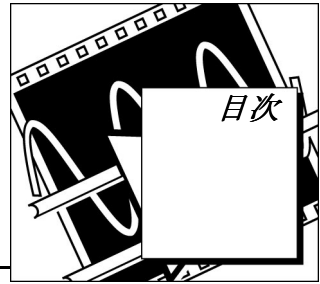
日本語: <http://www.natinst.com/nni>

米国本社

6504 Bridge Point Parkway Austin, TX 78730-5039 Tel: (512) 794-0100 Fax: (512) 794-8411

日本ナショナルインスツルメンツ株式会社

郵便番号 105 東京都港区芝公園 2-4-1 秀和芝パークビル B 館 5 F



第 1 章

LabVIEW の概要

LabVIEW とは.....	1-1
LabVIEW を使用する利点.....	1-3
LabVIEW の動作方法.....	1-4
フロントパネル.....	1-4
ブロックダイアグラム.....	1-5
パレット.....	1-6
ツールパレット.....	1-6
制御器パレット.....	1-6
関数パレット.....	1-7
データフロー.....	1-7

第 2 章

仮想計測器

サンプル VI を実行する.....	2-1
仮想計測器を作成する.....	2-3
ユーザインタフェースを作成する.....	2-3
ブロックダイアグラムを作成する.....	2-7
配線し VI を実行する.....	2-9
VI にタイミングを追加する.....	2-11
解析とファイル I/O を VI に追加する.....	2-13

第 3 章

データ集録

作成した VI にアナログ入力を追加する.....	3-1
---------------------------	-----

第 4 章

計測制御

計測器 I/O.....	4-1
計測器ドライバを使用する.....	4-2
Demo Scope VI を動作させる.....	4-5

第 5 章 デバッグ

実行ハイライトを使用する	5-1
プローブによるシングルステップ動作	5-3

第 6 章 ユーザ支援

オンラインヘルプ	6-1
ヘルプを表示する	6-2
ナショナルインスツルメンツのユーザ支援体制	6-3
顧客の教育	6-3
アライアンスプログラム	6-3
技術サポート	6-3

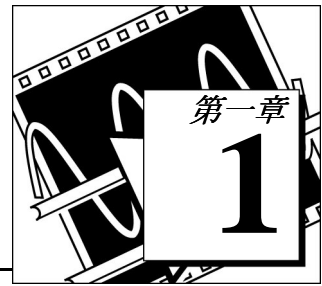
付録

インストール方法

システム構成の条件	A-1
インストール手順	A-1

用語集

LabVIEW の概要



本章では LabVIEW の操作環境の概略を説明します。

システム構成の条件やインストール方法についての詳細は、付録「インストール方法」をご覧ください。

LabVIEW とは

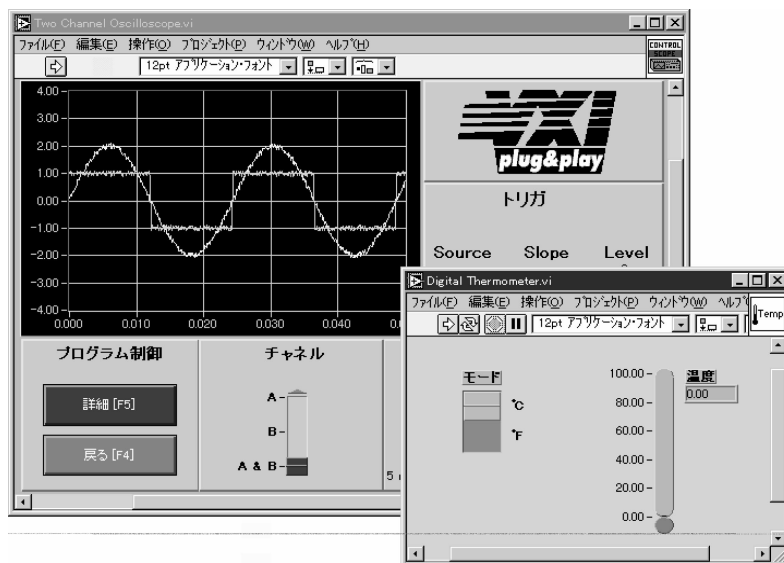
イタリック体のテキストは用語集掲載語句です。

LabVIEW (Laboratory Virtual Instrument Engineering Workbench) は、グラフィカルプログラミング言語 *G* をベースとした開発環境です。LabVIEW は *GPIB*、*VXI*、*RS-232*、*RS-485* はもとより、プラグインデータ集録ボードなど各種ハードウェアとの通信に 100% 統合性があります。また、LabVIEW には付属ライブラリが備わり、TCP/IP ネットワーキングや OLE オートメーションなどのソフトウェア規格が同時に利用できます。

LabVIEW の使用により、32 ビット コンパイル プログラムを作成することができ、ユーザ独自のデータ集録、試験、測定ソリューションに必要な速い実行速度が得られます。さらに、LabVIEW はネイティブ 32 ビットコンパイラであるため、スタンドアロン実行ファイルの作成も可能です。

プログラミング経験に乏しいユーザでも LabVIEW をご使用いただけます。それは LabVIEW がプログラミング動作を記述するために、技術者・科学者・エンジニアが使い慣れた用語、アイコン、および概念を用い、しかもテキスト形式の言語ではなく、視覚的記号に依存しているからです。

実際のアプリケーションに適応される LabVIEW のプログラムは、下図に示すとおり、単純なプログラムから強力なプログラムまで様々です。



従来のプログラム開発ツールに加え、LabVIEW にはデータ集録、解析、表示、保存のための包括的ライブラリが備わっています。ブレークポイントの設定やプログラム実行を動画化して、プログラム実行の確認をしたり、さらにプログラム中にシングルステップを設定してデバッグやプログラムの作成を容易にすることができます。

さらに、LabVIEW は DLL、共有ライブラリ、OLE オートメーションなどを介して、外部のコードやソフトウェアに接続する機構を多数用意しています。加えて、多彩なアドオンツールキットをそろえて、広範なアプリケーションニーズに対応しています。

LabVIEW を使用する利点

LabVIEW のおかげで、ユーザは科学的なシステムやエンジニアリングシステムのためのユーザ独自のソリューションを構築する手段を獲得しました。LabVIEW によりパワフルなプログラミング言語の柔軟性と性能が無理なく容易に利用できます。

LabVIEW は何万人ものユーザに、計測やデータ集録システムのプログラムを作成する迅速な手段を提供してきました。独自の計測システムの試作、設計、試験などに LabVIEW を使用することで、システムの開発時間を短縮し、生産性を 4 倍から 10 倍まで高めることができます。

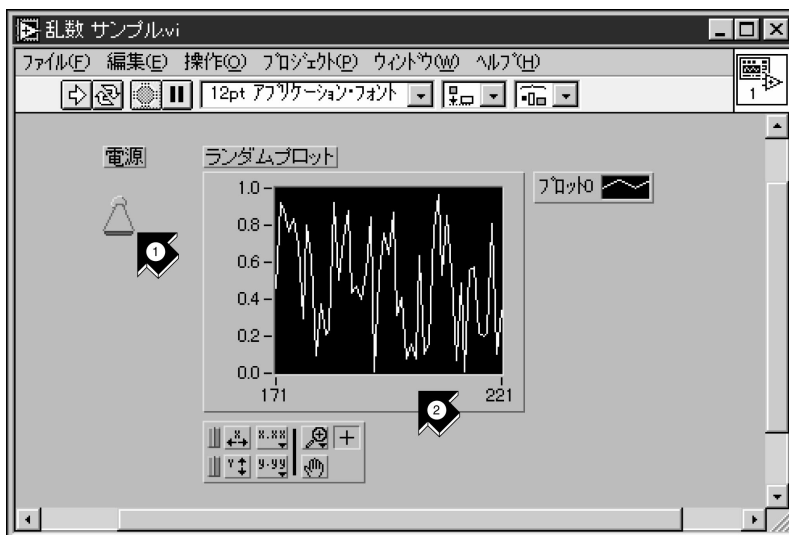
さらに LabVIEW が提供する利点には、大規模にインストールされたユーザベース、長期的な製品フィードバックのほか、パワフルなアドオンツールなどがあります。最終的に、ナショナルインスツルメンツの支援サービスとユーザネットワークにより、ユーザは確実にソリューションの作成ができます。

LabVIEW の動作方法

LabVIEW のプログラム（仮想計測器または VI）にはすべてフロントパネルとブロックダイアグラムがあります。LabVIEW のパレットには VI の作成に使用する選択項目がそろっています。このセクションでは、以下 LabVIEW のフロントパネル、ブロックダイアグラム、パレットの概要と、LabVIEW が順ずるプログラム実行用データフローモデルを説明します。

フロントパネル

フロントパネルは LabVIEW の VI に備わったユーザグラフィカルインタフェースです。このインタフェースはユーザからの入力を集録し、プログラム出力を表示します。フロントパネル内にはノブ、押しボタン、グラフなどを入れることができます。



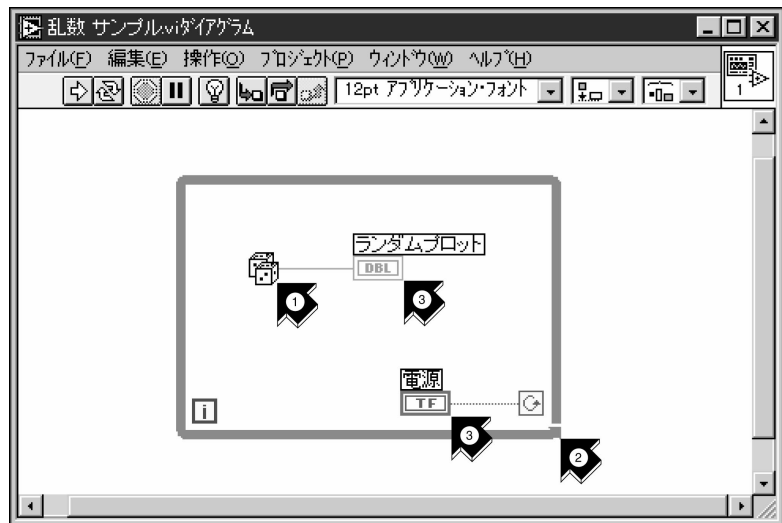
1. 制御器（入力）

2. 表示器（出力）

ブロックダイアグラム

ブロックダイアグラムにはユーザ VI のグラフィカルソースコードがあります。ブロックダイアグラムで VI をプログラムし、フロントパネルで作成した入出力に対する制御や関数を実行します。

ブロックダイアグラムには LabVIEW の付属 VI ライブラリに収納されている関数やストラクチャを入れることもできます。さらにフロントパネルで作成した制御器や表示器に関連する端子が含まれています。



1. 関数
2. ストラクチャ
3. フロントパネルで作成した端子

パレット

フロントパネルやブロックダイアグラムの作成と編集に必要な項目を LabVIEW のパレットで選択することができます。

ツールパレット

ツールパレットはフロントパネルとブロックダイアグラムの両方で使用します。ツールパレットには、フロントパネルオブジェクトのほかブロックダイアグラムオブジェクトの編集やデバッグに使うツールがそろっています。



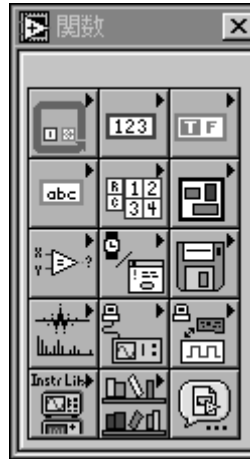
制御器パレット

制御器パレットはフロントパネルで使用します。制御器パレットには、ユーザインタフェースの作成時に使用するフロントパネルの制御器や表示器があります。



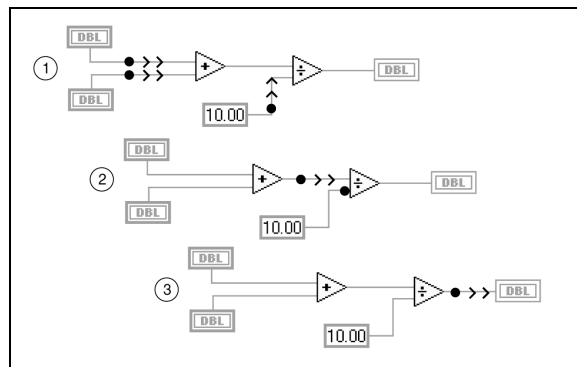
関数パレット

関数パレットはブロックダイアグラムで使用します。関数パレットには、演算、計測器 I/O、ファイル I/O のほか、データ集録操作など、VI のプログラミングに使用するオブジェクトがそろっています。



データフロー

LabVIEW の VI はデータフローモデルに従いプログラムを実行します。ブロックダイアグラムは VI、ストラクチャ、フロントパネル上で作成した端子などの接点から構成されています。これらの接点はワイヤで結ばれ、これがプログラムのデータフローを定義します。接点のすべての入力に値が到達した時点で実行を行い、実行完了後、その接点の出力すべてをデータパス上の次の接点に引き継ぎます。



仮想計測器



本章では LabVIEW でアプリケーションを作成する方法を手順にそって説明します。

本章の内容：

- ・ 新規プログラムを LabVIEW で作成します。
 - LabVIEW に付属のサンプル VI を開いて実行の確認を行います。
 - 構築した VI でデータを生成・解析し、ファイルへの書き込みを行います。

サンプル VI を実行する

LabVIEW のサンプル VI を実行することによって LabVIEW の動作を確認します。

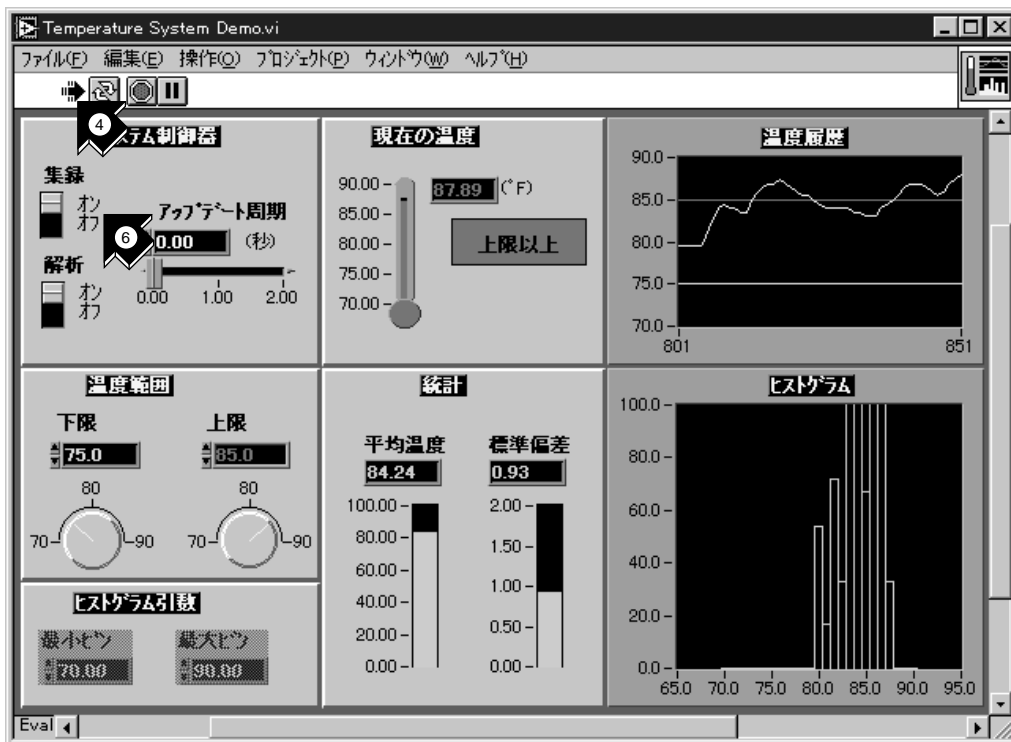


このセクションの終了時間の目安は約 5 分です。



1. 既存のサンプル VI を開くには、LabVIEW ダイアログボックスから **VI を開く** を選択してください。

2. ファイルダイアログで Examples ディレクトリの中にある Apps ディレクトリ 内の tempsys.llb を選択してください。
3. Temperature System Demo.vi という デモ用 VI をクリックし、開いてください。



 実行ボタン

4. フロントパネルのツールバー上の**実行**ボタンをクリックし、VIの動作を確認してください。
5. スライド、ノブ、および制御器をそれぞれクリックし、それがデータに与える影響を確認してください。
6. **集録**スイッチをオフ側にクリックし、プログラムを停止させます。

仮想計測器を作成する

LabVIEW でのチャート作成、解析、およびデータ保存がいかに容易であるかを、以下の操作で実際に試してみます。



この操作の終了時間の目安は約 30 分です。

ユーザインタフェースを作成する

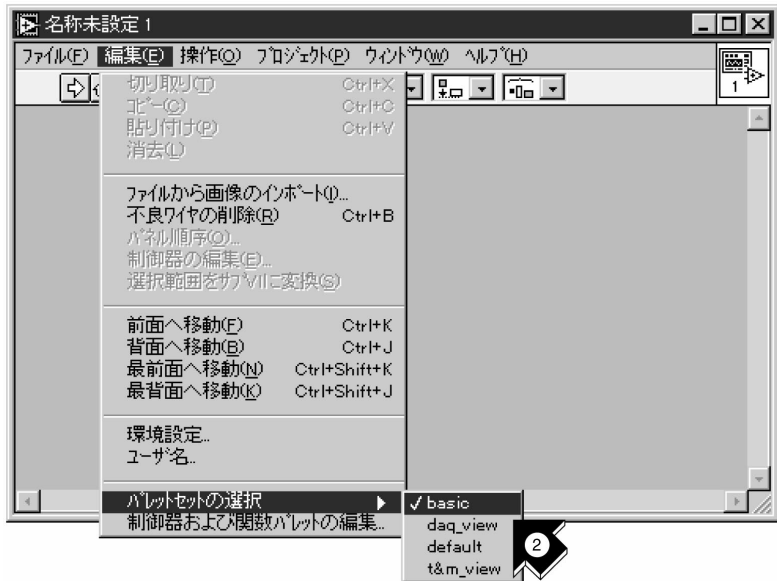
コントロールパレットから選択した制御器や表示器を使い、フロントパネル上にグラフィカルユーザインタフェースを作成します。



このセクションの終了時間の目安は約 5 分です。



1. LabVIEW ダイアログボックスの**新規 VI** をクリックして新しい VI を開きます。

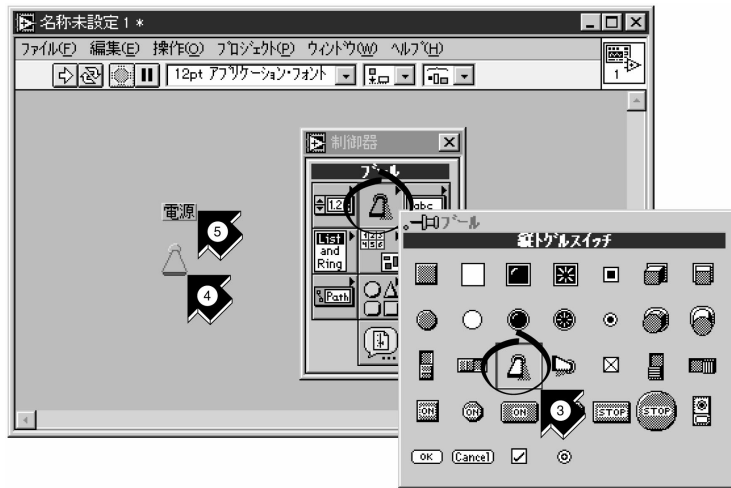


2. フロントパネルで**編集 >> パレットセットの選択 >> basic** を選択してください。



ヒント 「>>>」シンボル重ねあったメニューとパレットのアイテムから最終動作を導きます。編集 >> パレットセットの選択 >> basic では編集メニューを開き、パレットセットの選択をクリックし最後に basic をパレットから選択します。

basic パレットで選択した操作は LabVIEW ライブラリの小さなサブセットのライブラリです。LabVIEW のすべての機能を見るには default パレットを選択して見るすることができます。



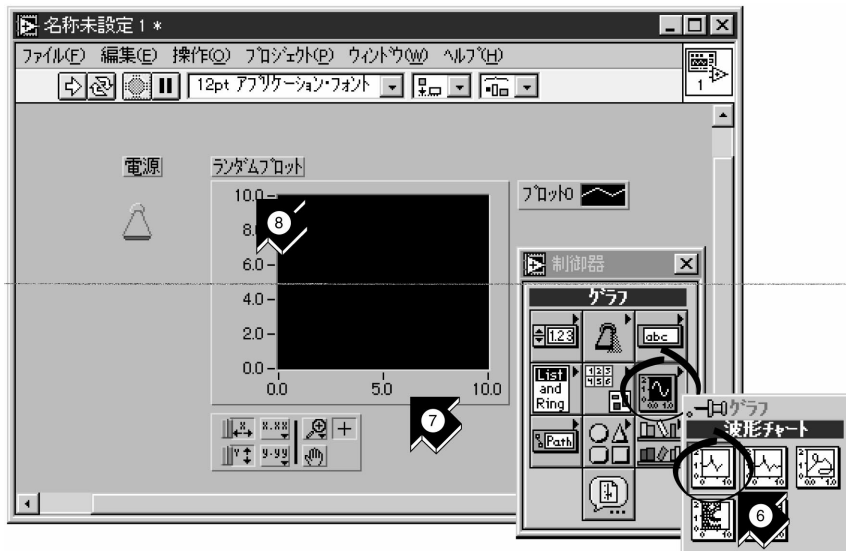
3. 縦トグルスイッチを制御器 >> ボールの順にサブパレットから選択してください。メニューから項目を選択する場合と同様に、アイコンをパレットから選択してください。
4. ポインタをフロントパネル上に移動し、クリックしてトグルスイッチをフロントパネルに配置してください。
5. グルスイッチのラベルには「電源」とキー入力してください。ラベルが表示されていない場合には、トグルスイッチのポップアップメニューから**表示 >> ラベル**を選んでください。ポップアップメニューへアクセスするには、トグルスイッチを右クリックしてください。(Macintosh の場合、トグルスイッチをコマンドークリックしてください。)



ヒント オブジェクトやワイヤの配置のし直しや、またはサイズ変更には、



位置決めツール



6. **制御器** >> **グラフ** >> **波形チャート** の順に選択し、**波形チャート** を作成してください。このチャートは一度に一点データをプロットします。
7. チャートをフロントパネル上に配置し、「ランダムプロット」というラベルをつけてください。
8. 波形チャートのスケールを変更するためには、ツールパレットから**操作ツール**を選択してください。ツールパレットが表示されていない場合には、**ウィンドウ** >> **ツールパレットを表示**の順に選択してください。表示器のY軸上にある**10.0**をダブルクリックし、「1.0」と入力してください。



操作ツール

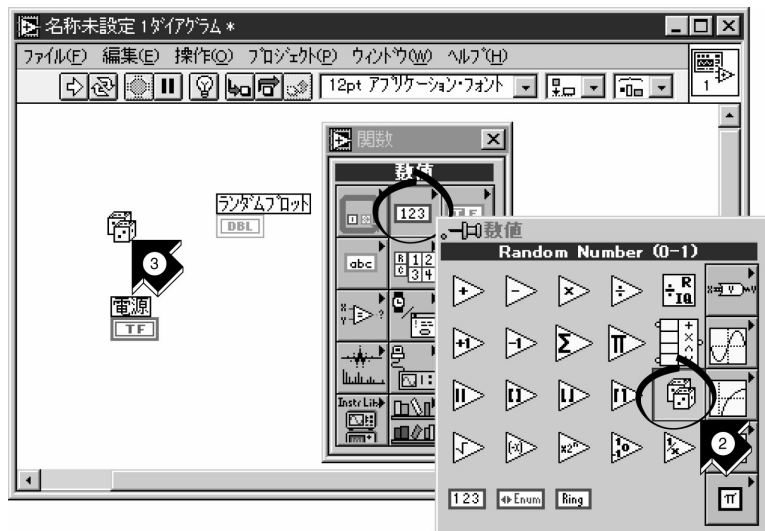
ブロックダイアグラムを作成する

次に、作成したVIのブロックダイアグラムでソースコードを作成してみます。

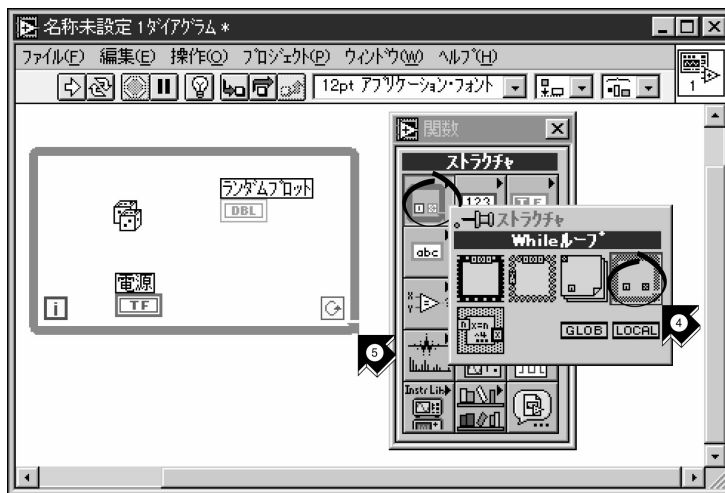


このセクションの終了時間の目安は約5分です。

1. **ウィンドウメニューからダイアグラムを表示**を選択するか、またはブロックダイアグラムウィンドウをクリックして、ブロックダイアグラムの画面を表示します。ブロックダイアグラムの2つの端子はそれぞれフロントパネル上の電源トグルスイッチとランダムプロット波形チャートに対応しています。



2. **関数 >> 数値**サブパレットから **Random Number (0-1)** を選択してください。関数パレットが表示されていない場合には、**ウィンドウ >> 関数パレットを表示**を選択してください。この操作では、乱数ジェネレータを使いVI用の入力データを作成します。
3. ブロックダイアグラム上に Random Number 関数を配置してください。このVIは0から1までの乱数を作成します。



条件端子

4. **関数 >> ストラクチャ**の順でサブパレットを選び、**While ループ**を選択してください。While ループは条件端子の値が TRUE 状態にある限り境界線内にあるコードすべてを繰り返し動作させます。この値が FALSE になると、While ループは終了します。
5. ループの左上の角を固定したいブロックダイアグラムの位置にポインタを配置してください。ループを斜めにドラッグし、Random Number 関数、電源トグルスイッチ、ランダムプロットチャートを囲んでください。



ヒント While ループがコードすべてを囲みきれない場合には、ループの端を右ボタンクリックし、ポップアップメニューにアクセスしてください。(Macintosh ではループの端をコマンドクリックしてください。) このポップアップメニューから While ループを削除を選び、再度上記の 4 と 5 の手順を繰り返してもう一つループを作成し、コード全部がループで囲まれるようにしてください。

配線し VI を実行する

配線を追加して VI のデータの流を定義する必要があります。VI の完成後、フロントパネルから VI を動作させ、作成したチャートにデータがプロットされたかどうかを確認してください。

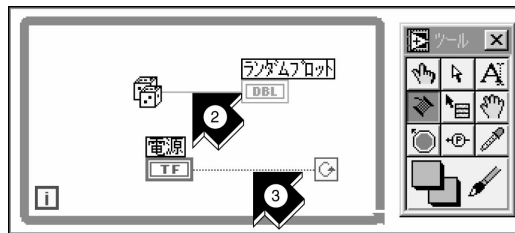


このセクションの終了時間の目安は約 5 分です。



配線ツール

1. ツールパレットから **配線ツール** を選択してください。ツールパレットが表示されていない場合には、**ウィンドウ >> ツールパレットを表示** を選択してください。



2. Random Number 関数をランダムプロットチャート端子に配線してください。ワイヤを作成するには、Random Number 関数をクリックし、ランダムプロットチャートの位置まで移動させ、もう一度クリックするとワイヤがつながります。



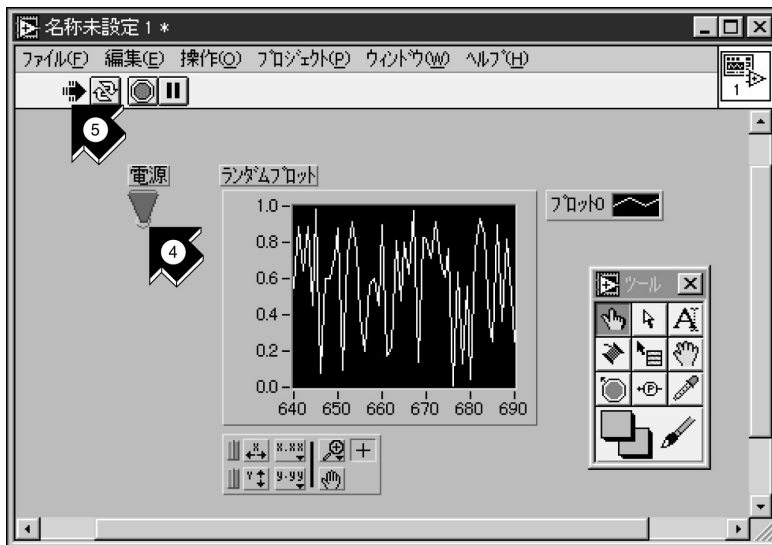
ヒント 配線ツールを端子上に配置すると、端子が点滅し端子ラベルのヒントラベルが現れます。正しい端子が点滅した時にクリックし、その端子からまたは端子にワイヤを配線してください。

黒い破線のワイヤが現れた場合には、編集メニューから不良ワイヤ削除を選択してください。



条件端子

3. 電源トグルスイッチ端子と While ループ上の条件端子とを接続してください。



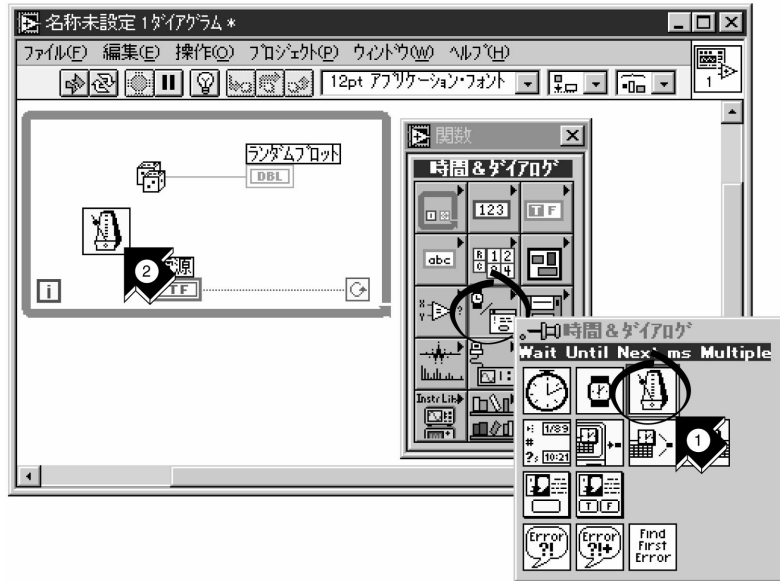
4. フロントパネルに戻り、**操作ツール**を選択してください。電源トグルスイッチをクリックし、TRUE 位置に倒してください。
5. ツールバーの**実行**ボタンをクリックし、VI を実行させてください。
6. VI 動作を停止させるためには、電源トグルスイッチをもう一度クリックし、FALSE 位置にもどしてください。While ループは条件端子が TRUE 位置にある場合にのみ実行されるため、トグルスイッチの値が FALSE に変更されるとループは停止します。

VIにタイミングを追加する

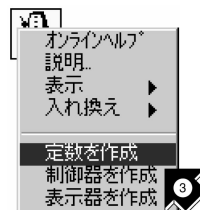
作成したVIにタイミングディレーを追加し、波形チャート上の各ポイントをゆっくりとプロットします。



このセクションの終了時間の目安は約5分です。



1. ブロックダイアグラムの画面を表示します。関数 >> 時間 & ダイアログの順でサブパレットを選択し、**Wait Until Next ms Multiple** を選択してください。
2. Wait Until Next ms Multiple 関数を While ループ内に配置してください。

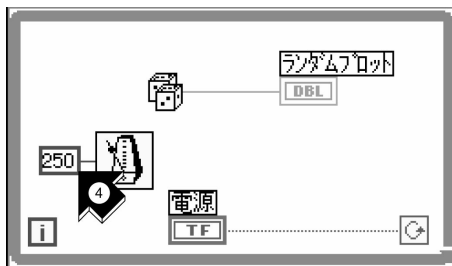


3. Wait Until Next ms Multiple 関数の左側のポップアップメニューから**定数を作成**を選んでください。関数のポップアップメニューをアクセスする際には、関数の左側にポインタが

あることを確認してください。左側にポインタがない場合には、作成した定数は関数に配線されません。



ヒント LabVIEW オブジェクトのポップアップメニューにアクセスするには、オブジェクトを右ボタンクリックしてください。Macintosh ではコマンドクリックをしてください。



4. ミリ秒倍数制御器に「250」とキー入力し、250ms の生成ポイント間のディレイを作成し、チャートをプロットしてください。
5. フロントパネル画面にもどり、電源トグルスイッチを TRUE 位置に倒し、VI を動作させ、ディレイの影響を確認してください。電源トグルスイッチをクリックし、FALSE 位置にもどし VI を停止させます。このプログラムは乱数 サンプル .vi という名前で、LabVIEW¥vi.lib¥tutorial.11b ディレクトリに保存してください。



実行ボタン



注意： 本マニュアルでは、パスはバックスラッシュ (¥) を使って表示し、ドライブ名、ディレクトリ、およびファイルと区別しています。

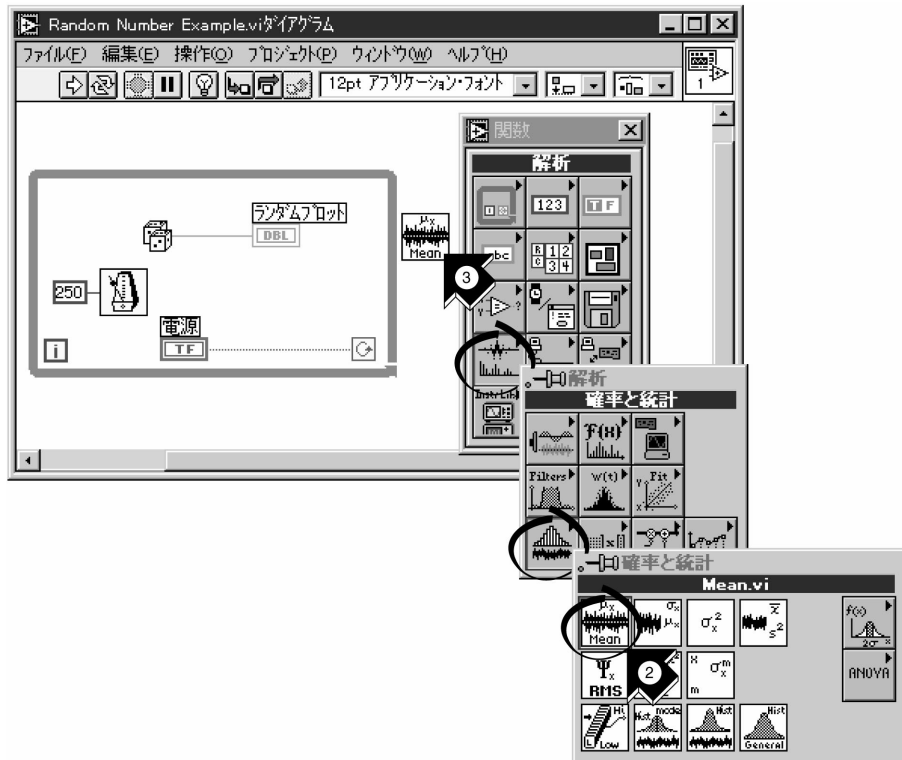
解析とファイル I/O を VI に追加する

集録した乱数データポイントを平均化し、データをスプレッドシートファイルに保存します。



このセクションの所要時間の目安は約 10 分です。

- すでに作成してある乱数 サンプル VI のブロックダイアグラム画面を表示してください。VI が開かれていない場合には、LabVIEW¥vi.lib¥tutorial.11b ディレクトリから Random Number Example.vi を選んでください。

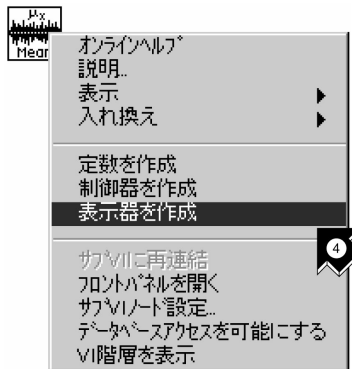


- 関数 >> 解析 >> 確立と統計の順にサブパレットを選び、最後に **Mean.vi** を選択してください。

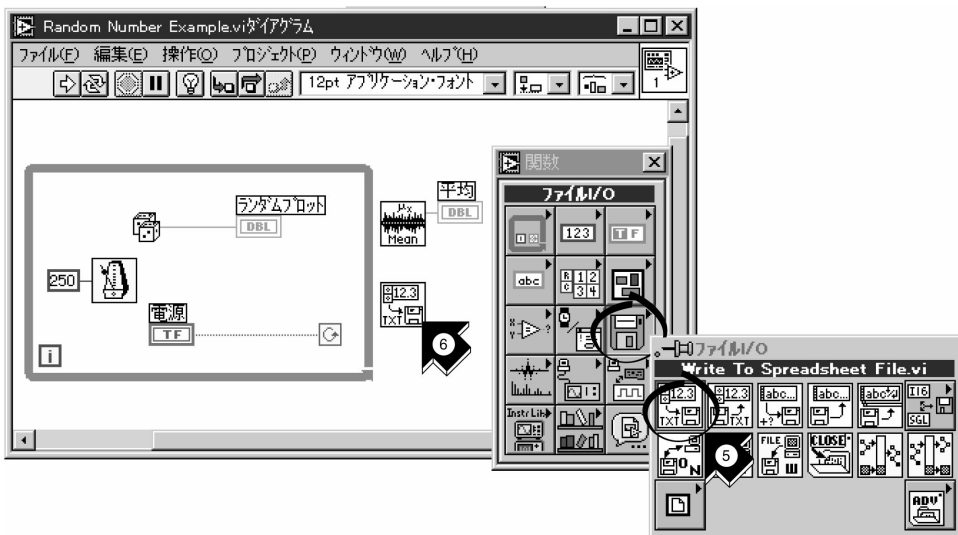


ヒント 関数パレットが表示されていない場合には、ウィンドウ >> 関数パレットの表示を選択してください。

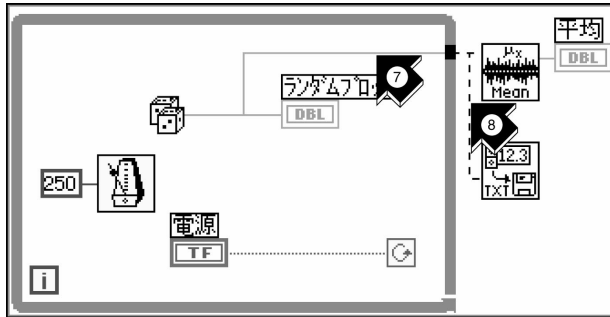
- ブロックダイアグラム内の Mean VI を While ループの外側に配置してください。



4. Mean vi の上部右角のポップアップメニューから**表示器を作成**を選んでください。こうしてフロントパネル上に乱数データの平均を表示する数値表示器が作成されました。



5. **関数 >> ファイル I/O** サブパレットの順で選び、**Write To Spreadsheet File.vi** を選択します。
6. ブロックダイアグラムにある Write To Spreadsheet File VI を While ループの外側に配置します。



配線ツール

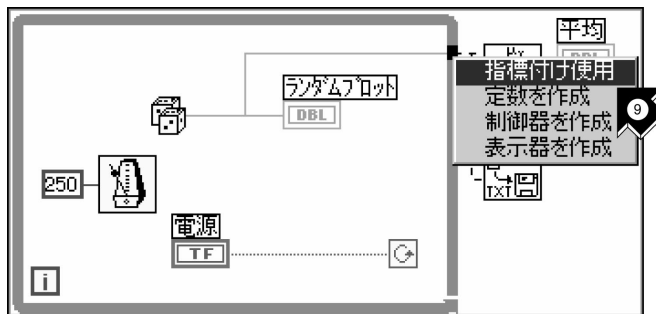
7. **配線** ツールを使用し、Random Number 関数を Mean VI の X 入力端子につなぎます。この分岐ワイヤは既存のワイヤセグメントから配線して作成してください。



ヒント 配線ツールが正確に位置決めされ、既存のセグメントから新規ワイヤに固定されると、ワイヤセグメントが点滅します。

配線ツールにより、ブロックダイアグラム内の節点上に端子用 **ヒント** ラベルを表示させることができます。

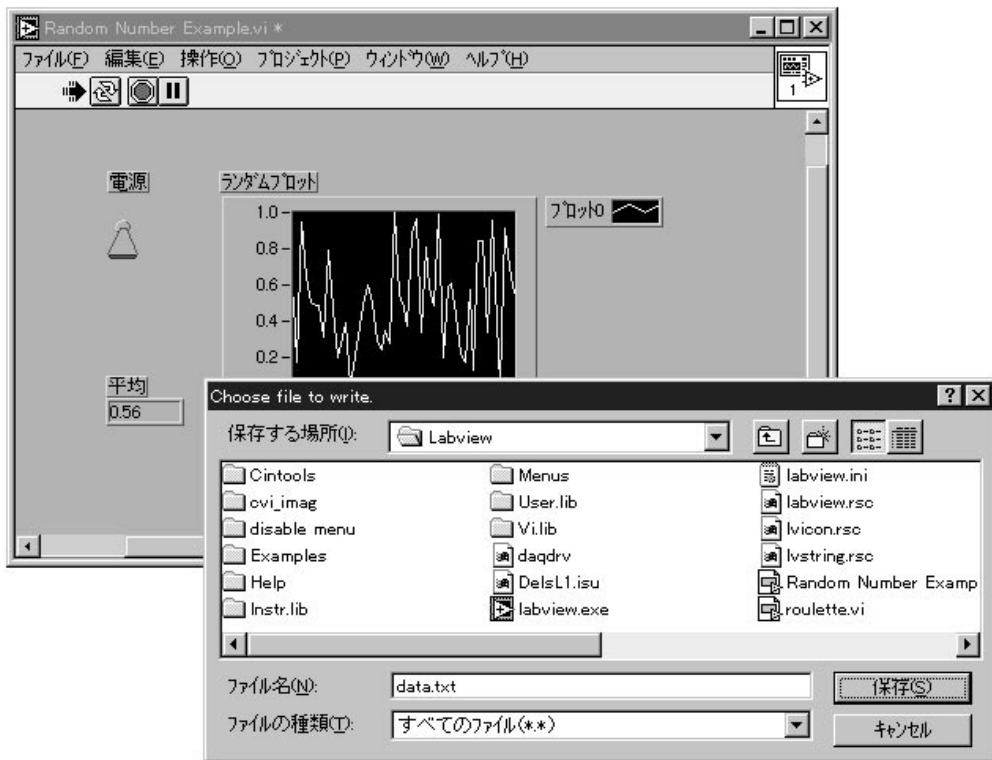
8. 前回のステップで作成した分岐ワイヤから、別の分岐ワイヤを作成してください。この新しい分岐ワイヤを Write to Spreadsheet File VI の **1次元データ** 入力と接続してください。1次元データ入力を使うのは、While ループがすでに作成した乱数値から1次元列データを作成するからです。



9. While ループ上の黒色トンネルはループのデータ排出端子です。黒色トンネルのポップアップメニューから **指標付け使用** を選んでください。破線のワイヤがオレンジ色の実線ワイヤに変化します。**指標付け使用** により While ループがループ終了時にデータを蓄積し、データセットとして、Mean VI に渡すことができます。



10. フロントパネル画面を出してください。操作ツールを操作し、電源トグルスイッチを TRUE 位置に倒し、VI を動作させてください。



11. 電源を切ると、データの平均値とともに、乱数ファイル名を要求するファイルダイアログボックスが表示されます。data.txt と入力し、保存をクリックしてください。



ヒント 電源が切られた際、この平均値はデータが集録されるまで表示されません。

12. テキストエディタを使い、data.txt を開き、データを確認してください。



注意 : LabVIEW¥vi.lib¥tutorial.11b¥Random Number Example Solution.VI で、この練習の解答が得られます。


データ集録



本章では LabVIEW を使って、データ集録用ハードウェアからデータ集録をする方法を説明します。スタンドアロン計測器のみを使い、その制御を GPIB、VXI あるいはシリアル規格で行う場合には、第 4 章、「計測制御」をご参照ください。

本章の内容：

- ・ 第 2 章、*仮想計測器*で作成した VI にアナログ入力を追加します。


 **注意：** データ集録ハードウェアのインストールと構成手順の詳細は、お手持のハードウェアマニュアルあるいは NI-DAQ オンラインヘルプファイルをご参照ください。

作成した VI にアナログ入力を追加する

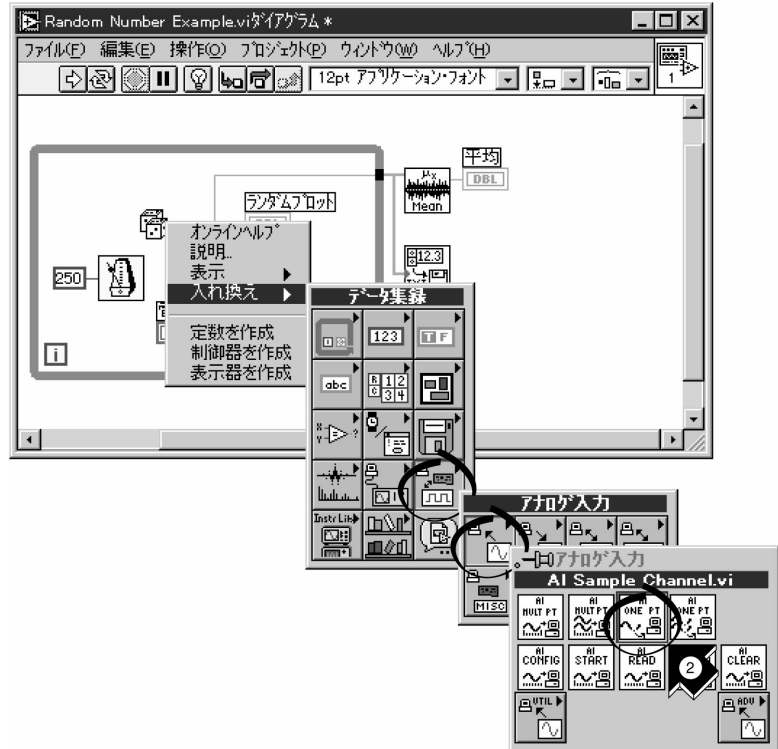
これまで、乱数を生成し、これらの数値をストリップチャートにプロットする乱数 サンプル VI を作成しました。次は、乱数ジェネレータをアナログ入力 VI と交換して、DAQ ボードからデータの集録やストリップチャートへのプロットのほか、ファイルへの書き込みをします。



この操作の終了時間の目安は約 30 分です。

 **注意：** 乱数 サンプル VI をまだ作成していない場合でも、ソリューション VI が `LabVIEW\vi.lib\tutorial.llb\Random Number Example Solution.vi` に入っています。

1. `LabVIEW\vi.lib\tutorial.llb` ディレクトリから、乱数サンプル VI を開いてください。



2. ブロックダイアグラム画面を表示してください。Random Number 関数のポップアップメニューから、**入れ換え >> データ集録 >> アナログ入力 >> AI Sample Channel.vi** の順に選んでください。

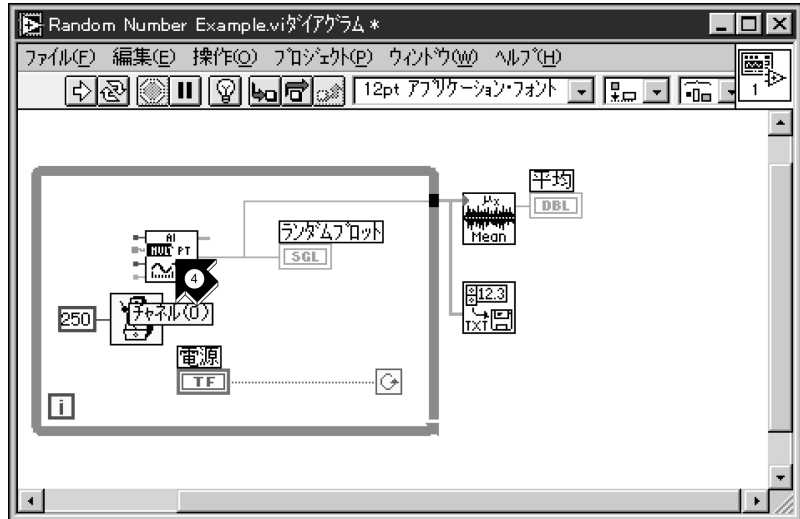


ヒント AI Sample Channels.vi ではなく、必ず AI Sample Channel.vi を選択してください。

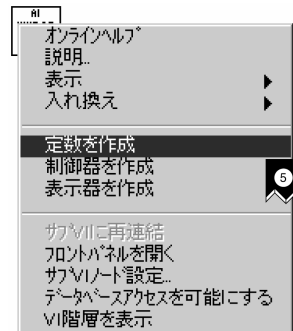


配線ツール

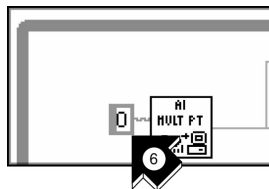
3. **配線** ツールをツールパレットから選択してください。配線ツールではブロックダイアグラム内の接点の端子に対応するヒントラベルが表示されます。



4. ポインタを AI Sample Channel VI の **チャンネル (0)** 端子上に移動させます。端子の名前を表示をしたヒントラベルが表示されます。



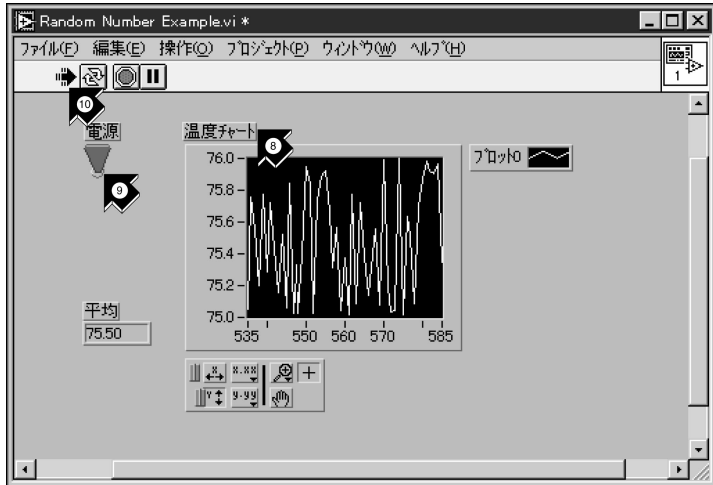
5. ハイライトされた**チャンネル (0)** 端子上のポップアップメニューから、**定数を作成**を選んでください。ポップアップメニューにアクセスするためには、端子上で右クリックしてください。(Macintosh の場合、端子上でコマンドークリックしてください。)



6. 作成された定数の中に「0」と入力してください。



7. フロントパネルを表示してください。グラフのポップアップメニューから、**Y 軸 >> Y 軸オートスケール**の順に選んでください。



ラベリングツール



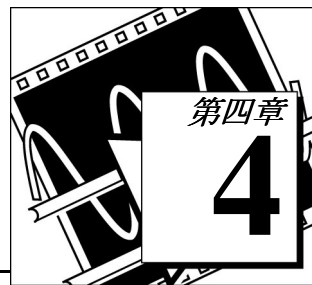
操作ツール



実行ボタン

8. **ラベリングツール**を使用し、チャートラベルをハイライトにし、ラベルを「温度チャート」と変更してください。
9. **操作ツール**を使用し、電源トグルスイッチを TRUE 位置にクリックしてください。
10. **実行ボタン**をクリックし、データチャートを見てください。
11. 電源トグルスイッチをクリックし、VI を停止させてください。
12. ファイルダイアログボックスに、data.txt などの名前を入力し、**保存**をクリックしてください。こうして、温度データはスプレッドシートファイルに保存されます。
13. この VI を「集録温度.vi」として
LabVIEW¥vi.lib¥tutorial.11b ディレクトリに保存してください。

計測制御



本章では GPIB、VXI、RS-232、およびその他のハードウェア規格で制御されるスタンドアロン型計測器から LabVIEW を使いデータを集録する方法を説明します。

本章の内容：

- ・ 計測器ドライバを使用して、お持ちの計測器と通信を行います。
- ・ 計測器用 I/O ハードウェアがインストールされていない場合、デモスコープ VI を動作させます。

計測器 I/O

LabVIEW は計測器ドライバを使い数々の計測器と通信を行います。計測器ドライバはプログラマブル計測器を制御する VI のライブラリです。計測器ごとの低レベルなプログラミングプロトコルを学習する必要性を省くことにより、LabVIEW の計測器ドライバは計測器制御を簡易化し、試験開発時間を短縮します。

計測器は一連のコマンドに従って遠隔制御に応答し、データを要求します。LabVIEW 計測器ドライバを使用すると、デジタルマルチメータ用 Read DC 電圧 VI やデジタルオシロスコープ用 Configure Time Axis VI など、直感的で高レベルな VI を動作させることができます。呼び出された計測器ドライバ VI は適切な計測器専用コマンドストリングを自動的に計測器に送信します。

LabVIEW 計測器ドライバの基礎は VISA (仮想計測器ソフトウェアアーキテクチャ) VI ライブラリで、これは GPIB、VXI、RS-232 のほか、他の種類の計測器タイプを制御する唯一のインタフェースライブラリです。VISA を使用する計測器ドライバはその計測器 I/O インタフェースでもスケール可能です。

計測器ドライバを使用する

以下の操作では、計測器ドライバを使いユーザの計測器と通信を行います。

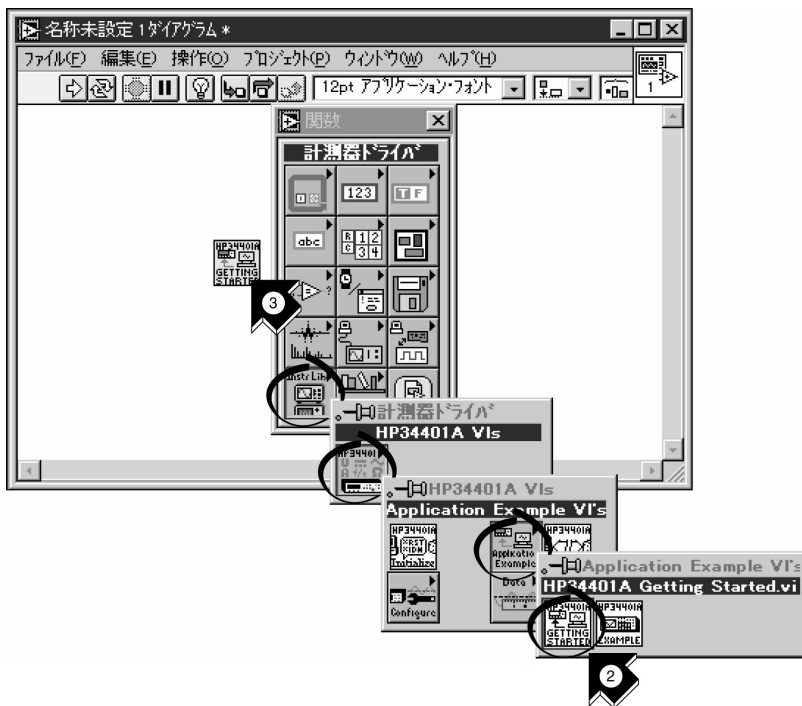


この操作の終了時間の目安は約 5 分です。



ヒント 計測器 I/O ハードウェアがインストールされていない場合には、**Demo Scope VI** を実行してください。

1. 新規 VI を作成し、ブロックダイアグラムを表示させてください。



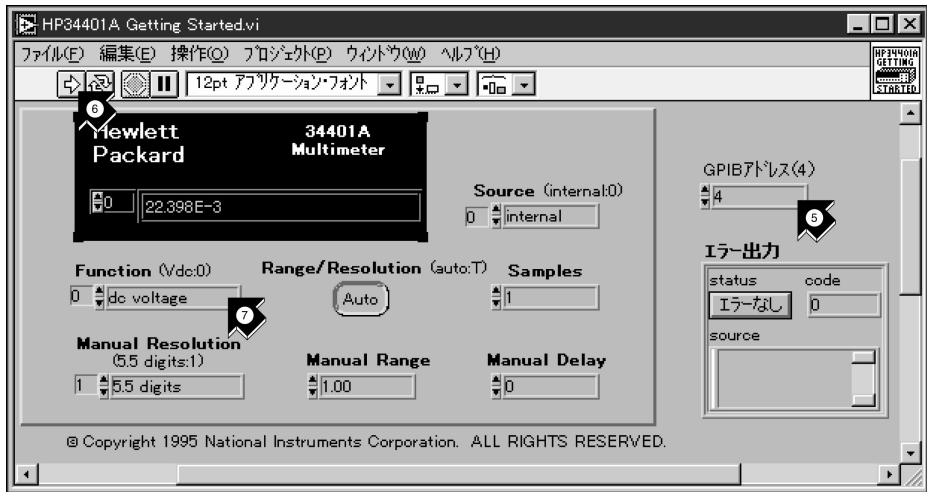
2. **関数 >> 計測器ドライバ >> HP34401A VI >> Application Example VI >> HP34401A Getting Started.vi** の順に選択するか、あるいは計測器名とその Getting Started VI を選択してください。



ヒント ここで計測器が見つからない場合には、LabVIEW CD またはナショナルインストルメンツ製計測器ドライバライブラリ CD を確認し、お

持ちの計測器用のドライバを LabVIEW ルートディレクトリの instr.lib フォルダにインストールしてください。

3. VI をブロックダイアグラム上に配置してください。
4. この HP34401A Getting Started VI をダブルクリックし、フロントパネルとブロックダイアグラムを見てください。この VI は HP34401A デジタルマルチメータを制御します。Getting Started VI はより高レベルの計測器ドライバサブ VI を呼び出し計測器を制御する高レベルプログラムの一例です。これらのサブ VI は計測器ドライバのパレット内にあります。



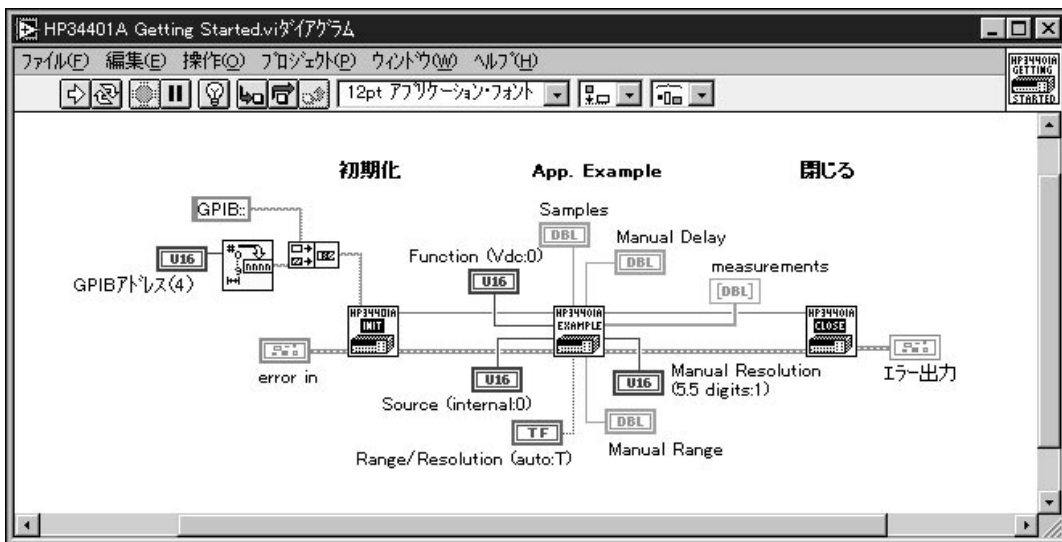
5. フロントパネル上で HP34401A Getting Started VI の GPIB アドレスを確認してください。デフォルトアドレスは 4 です。
6. VI を実行し、DC 電圧を測定してください。
7. **操作**ツールを使い、**Function** 入力を変更し、VI を再度動作させて、他の測定値を確認してください。



実行ボタン



操作ツール



8. ブロックダイアグラムに注目してください。まず **HP34401 A Initialize** サブ VI が呼び出されます。続いて計測器制御用サブ VI が、最後に **HP 34401A Close** サブ VI が呼び出されます。



ヒント 計測器ドライバのサブ VI は、関数 >> 計測器ドライバ >> お持ちの計測器の順に選択して見つかります。計測器ドライバを使い独自にプログラムの書き込みを行う場合には、最初に初期化を呼び出し、次に計測器を制御するために使用する一連のサブ VI を、最後に終了サブ VI を選択し計測器との通信を終了します。

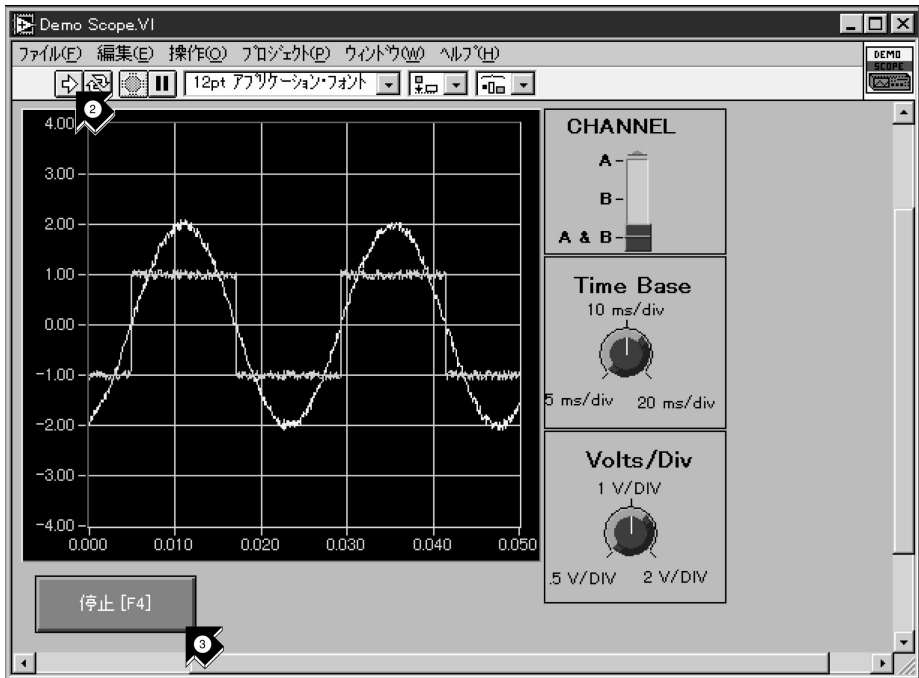
Demo Scope VI を動作させる

計測器 I/O ハードウェアがインストールされていない場合には、デモスコープ VI を動作させ、計測器 I/O の例をみるることができます。デモスコープ VI は実際の計測器ドライバ用 Getting Started VI のデモ版に相当します。



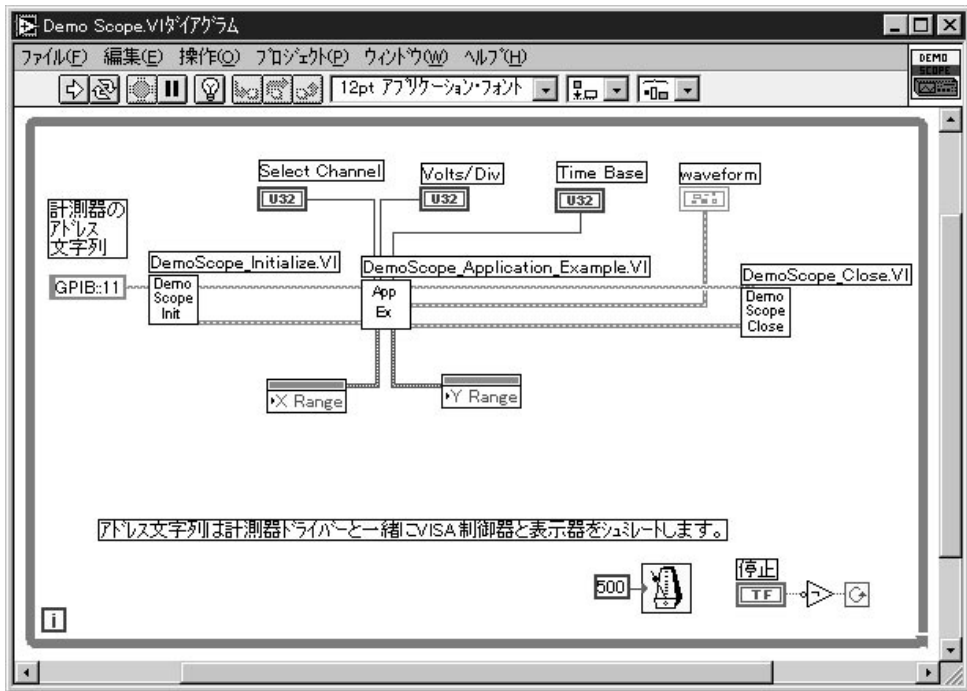
この操作の終了時間の目安は約 5 分です。

1. LabVIEW\vi.lib\tutorial.11b にある Demo Scope VI 計測器ドライバを開きます。



実行ボタン

2. プログラムを動作させ、オシロスコープのチャンネルの 1 つあるいは 2 つからデータを集録してください。タイムベースと 1 区分の圧力の設定を変更し、その結果を確認してください。
3. 矩形の**停止 [F4]** ボタンをクリックし、VI を停止させてください。



4. ブロックダイアグラム画面に戻ってください。最初に **Initialize** が呼び出され、続いて Application Example VI で計測器にコマンドが送られることに注意してください。そして **Close** VI が計測器との通信を完了します。LabVIEW 計測器ドライバを使いプログラムをする場合には、まずこの計測器初期化モデルに従い、次に関数を呼び出して計測器を制御し、最後に計測器の通信を完了してください。

デバッグ

本章では LabVIEW の包括的デバッグユーティリティの使い方を説明します。

実行ハイライトを使用する

実行ハイライトは実行中の VI のデータの流れを追います。

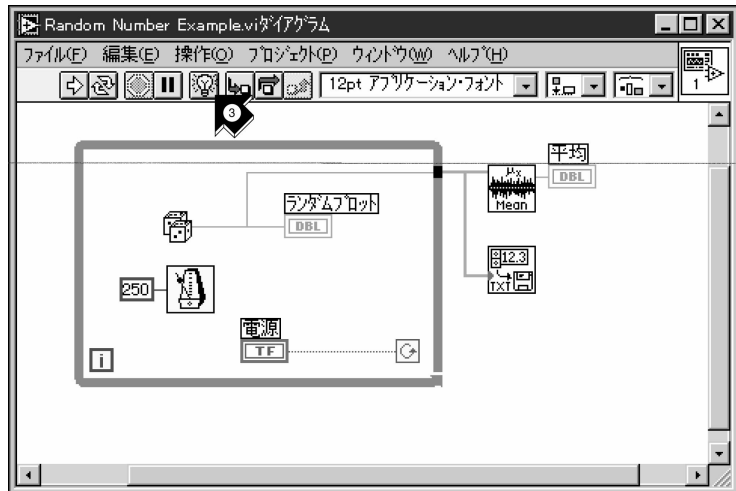


この操作の終了時間の目安は約 5 分です。



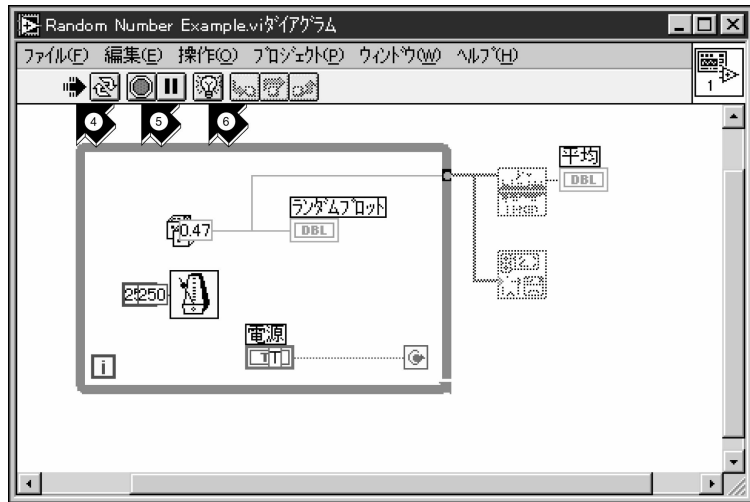
操作ツール

1. LabVIEW¥vi.lib¥tutorial.11b ディレクトリから、以前に作成した乱数サンプル VI を開いてください。
2. **操作**ツールを使い、電源トグルスイッチをオンにしてください。



実行ハイライト

3. ブロックダイアグラムを表示し、ツールバーの**実行ハイライト**ボタンをクリックしてください。



実行ボタン



停止ボタン



実行ハイライト

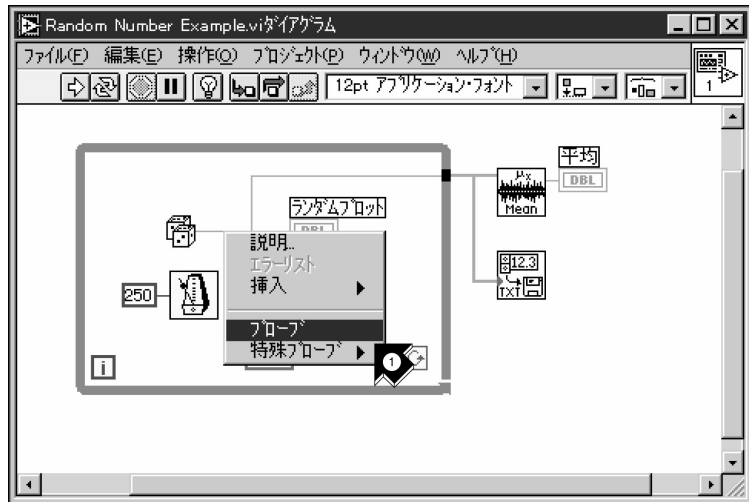
4. ブロックダイアグラムの VI を動作させてください。プログラムはゆっくりと点で実行の流れをハイライトするとともに、データが VI で利用可能になりしだい、そのデータも表示します。
5. **停止** ボタンをクリックして VI を停止してください。
6. 再度実行ハイライトボタンをクリックし、実行ハイライトを停止してください。

プローブによるシングルステップ動作

VIにブレークポイントを設定して、ブロックダイアグラムにワンステップで実行したり、あるいは部分的に飛び越すこともできます。さらに実行中のプログラムにプローブを挿入し、データの値を詳細に調べることも可能です。次の例ではVIでのシングルステップの実行方法とプローブを挿入し、VIを実行してデータをモニタする方法を示します。



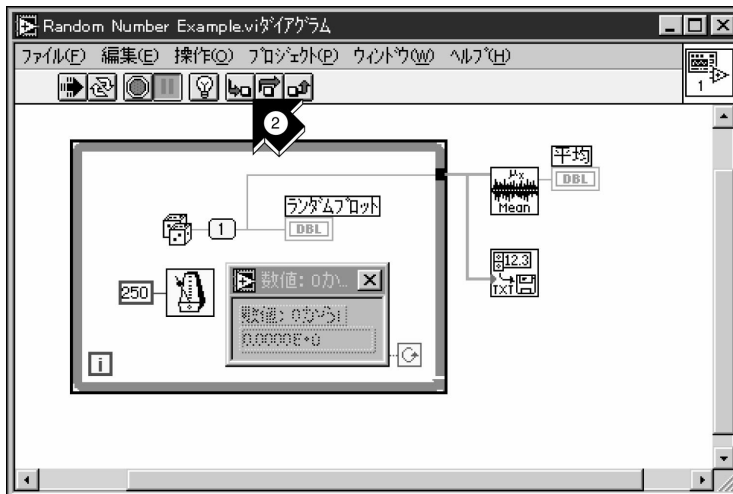
この操作の終了時間の目安は約5分です。



1. Random Number 関数の出力ワイヤのポップアップメニューから、**プローブ**を選んでください。小さな**数値: ...** ウィンドウが表示され、そのポイントでのデータの数値が表示されます。

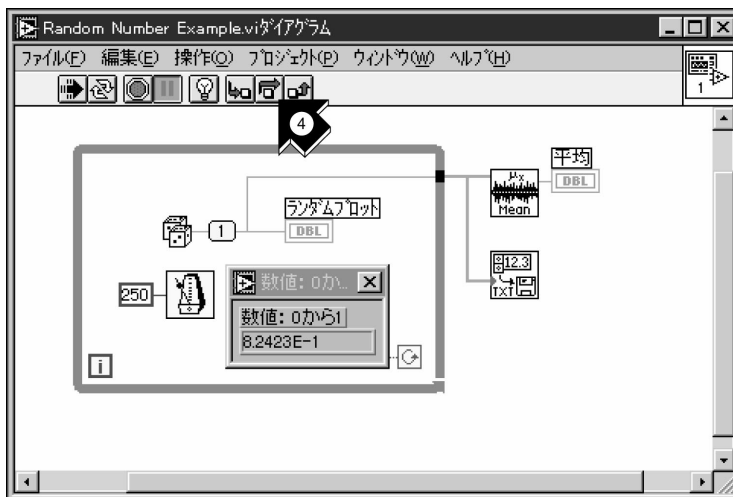


ヒント フロントパネルの電源トグルスイッチがオンになっていることを確認してください。



中に入る

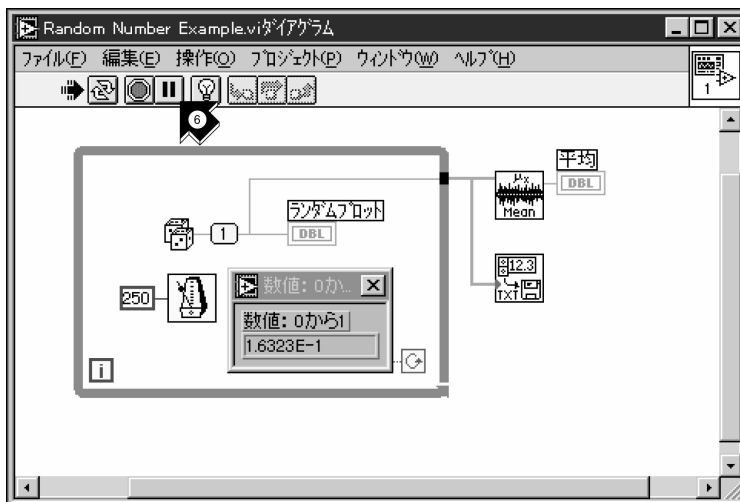
2. ツールバーの**中に入る**ボタンをクリックしてください。While ループが点滅し、プログラムがシングルステップモードで動作中であることを示します。
3. **中に入る**ボタンをもう一度クリックしてください。Random Number 関数が点滅します。



飛び越える

4. **飛び越える**ボタンをクリックして Random Number 関数を飛び越えてください。Random Number 関数が実行され、出力がプロープに表示されます。

5. **中に入る**ボタンと**飛び越える**ボタンを使い、ループで数回シングルステップ動作を行い、VIが作成した各乱数の状態を確認してください。



6. 通常のモードで VI を実行するためには、**一時停止**ボタンの選択を解除してください。
7. フロントパネル画面にもどり、電源トグルスイッチを切り、VIを終了します。集録データを data.txt に保存してください。

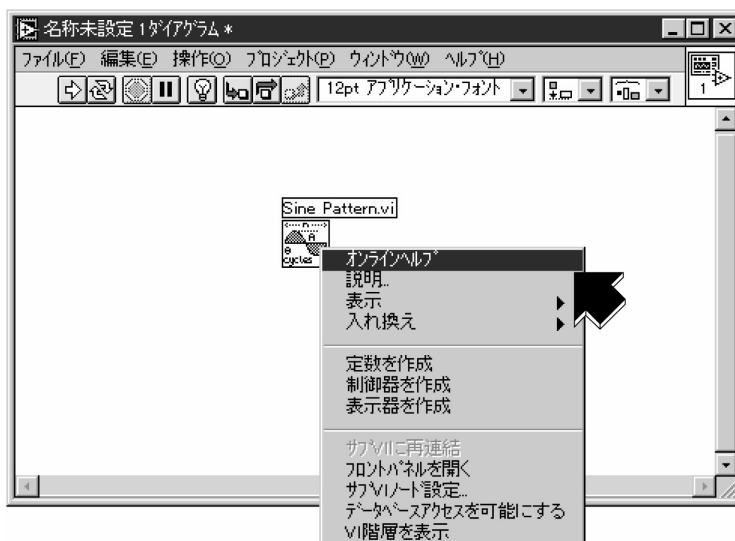
ユーザ支援



本章では LabVIEW のオンラインヘルプの使用方法を説明します。加えて、ユーザに成功をもたらすための弊社の努力と、ユーザが利用できるその他のリソースの概略を説明します。LabVIEW のプログラミング技術をさらに詳しく学習するためには、LabVIEW 開発システムに付属する *LabVIEW チュートリアルマニュアル* および *LabVIEW ユーザマニュアル* をご参照ください。

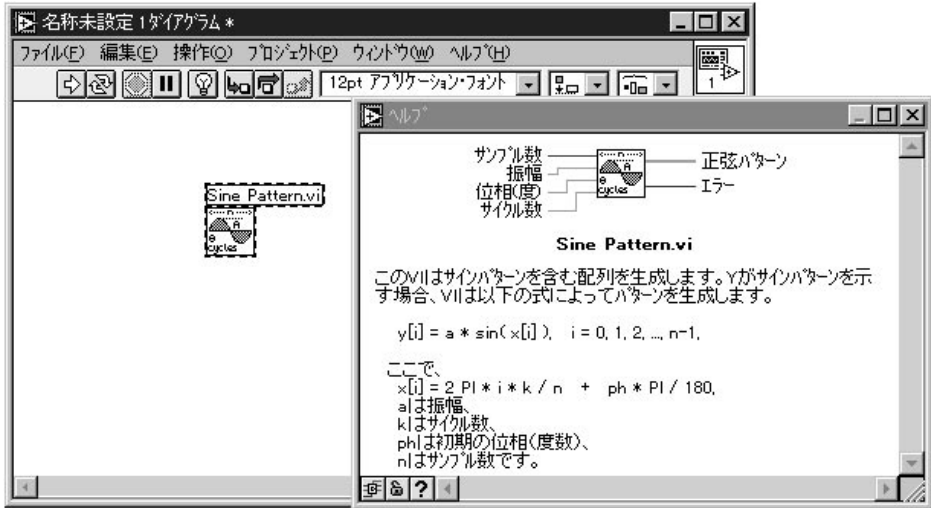
オンラインヘルプ

LabVIEW に搭載の VI にはすべて**オンラインリファレンス**がついています。見慣れない VI がある場合には、ブロックダイアグラム上にその VI を配置し、VI のポップアップメニューから**オンラインヘルプ**を選びます。オンラインヘルプで VI の機能とパラメータ全般についての説明を見ることができます。

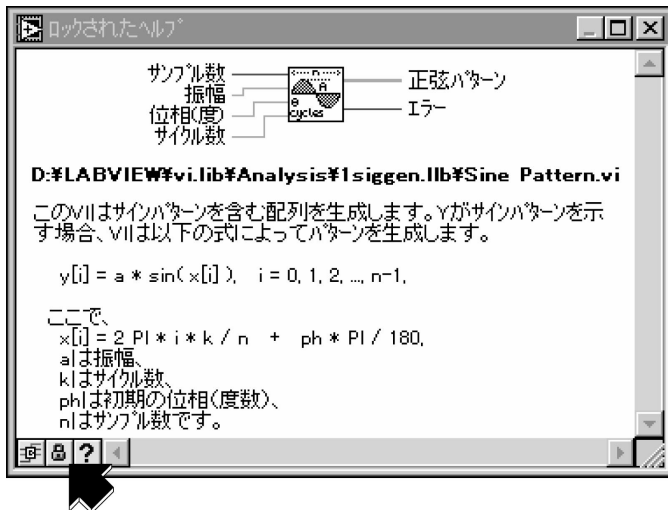


ヘルプを表示する

クイックリファレンスでVIの機能あるいは入出力パラメータを確認する必要がある場合、**ヘルプ >> ヘルプを表示**を選択してください。



ヘルプウィンドウは文脈に反応しているので、VIの情報が必要な時にはポインタをVIアイコンの上に合わせてください。ヘルプウィンドウ下側の**ロック**ボタンをクリックし、現在表示されているヘルプウィンドウの内容をロックしてください。



ナショナルインスツルメンツのユーザ支援体制

LabVIEW は、計測システム開発を簡易化するツールをユーザに提供するという弊社の長年にわたる努力を表しています。開発環境として LabVIEW を選んだユーザは、パワフルなグラフィカルプログラミングを利用する何万人もの科学者やエンジニアの仲間入りをしたこととなります。

顧客の教育

トレーニングを希望するユーザのために、対話式の CD、ビデオ、本、および LabVIEW コースを用意し、ユーザが短時間で LabVIEW をマスターし、アプリケーションを上手に開発するのに役立っています。

ナショナルインスツルメンツの提供する LabVIEW の総合トレーニングコースは、短時間でアプリケーション構築を学ぶのに有効です。3 日間の基本コースと上級コースでは、LabVIEW の開発環境について弊社のエンジニアが開発のこつを教えるほか、集中講座を行います。これらのコースは東京と大阪で開講しています。さらに、GPIB、DAQ、SCXI などその他特定の題目に関するコースも開講し、ユーザのシステム全体の開発に役立っています。

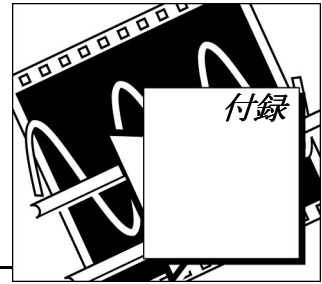
アライアンスプログラム

アライアンスプログラムは、LabVIEW とその他ナショナルインスツルメンツ製商品のエキスパートである開発者やコンサルタントの第三者間のネットワークです。弊社の *Solution* ディレクトリにはこうしたアライアンスメンバによって開発されたライブラリやユーティリティが別途に一覧化されており、LabVIEW の利用に役立っています。また、*Solution* ディレクトリにはカスタムアプリケーション開発に有効な LabVIEW 専門のコンサルタント名も掲載されています。

技術サポート

ナショナルインスツルメンツは充実した技術サポートを提供しています。まず、インターネットのサイト（WEB または FTP）、BBS、あるいはファックス応答サービスを利用し、重要な情報や製品事例、質問・回答文書、および技術開発のヒントなどをダウンロードできます。また、技術フォーラムがインターネットで利用でき、他の LabVIEW ユーザと意見の交換が可能です。さらに、ナショナルインスツルメンツの経験豊富なエンジニアが世界各地に駐在し、ユーザのサポートを行っています。

インストール方法



この付録では LabVIEW のためのシステム条件とインストール手順を説明します。

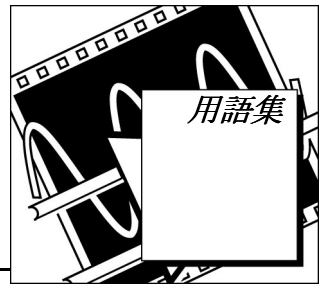
システム構成の条件

LabVIEW が Windows プラットフォームに対応するためには次のようなシステム構成条件があります：

- Windows 3.1
 - LabVIEW は最低 8MB までの RAM を搭載した 386 高性能モードで動作します。80486CPU 搭載コンピュータで最速動作を行います。80386 ベースのパソコンでも動作できます。Windows 対応の LabVIEW にはコプロセッサが必要となります。
- Windows 95
 - LabVIEW は最低 8MB から 12MB までの RAM を搭載した Windows95 をサポートするシステムすべてで動作し、効率の高い操作が可能です。
- Windows NT
 - LabVIEW が効率よく動作するためには、最低 12MB から 16MB までの RAM 搭載 Windows NT バージョン 3. 5. 1 以上が必要です。Windows 対応の LabVIEW は Windows NT 80X86 コンピュータでのみ動作します。

インストール手順

- LabVIEW 開発システム
 - LabVIEW のインストール方法については、*LabVIEW リリースノート*を参照。
- 評価ソフトウェア
 - *LabVIEW* 評価パッケージからインストールする場合の手順は、*setup.exe* を実行してください。



G

G LabVIEW のアプリケーション作成に使用するグラフィカルプログラミング言語。

GPIB GPIB(General Purpose Interface Bus: 汎用インタフェース)は、ANSI/IEEE 規格集 488.1-1987 と、ANSI/IEEE 規格集 488. 2 - 1987 で定義されている通信インタフェースシステムの通称です。このバスを考案したヒューレットパッカード社では、HP-IB と呼んでいます。

L

LabVIEW Laboratory Virtual Instrument Engineering Workbench ーラボラトリー仮想計測エンジニアリングワークベンチ

R

RS-232 推奨規格 232、シリアルインタフェースバス規格。

RS-485 推奨規格 485、シリアルインタフェースバス規格。

V

VI 仮想計測器を参照。

VISA バーチャル計測器ソフトウェアアーキテクチャ。GPIB、VXI、RS-232 およびその他の種類の計測器を制御する単独インタフェースライブラリ。

VXI VME eXtension for Instrumentation (バス)

W

While ループ ある条件が満たされるまでコードを部分的に繰り返すループ構造。従来のプログラミング言語の Do ループや Repeat Until ループに相当します。

あ

位置決めツール オブジェクトを移動、選択、サイズ変更をするために使われるツール。

か

仮想計測器 実際の計測器の概観や機能を模倣したグラフィカルプログラミング言語 G のプログラム。

関数 内蔵されている実行要素で、従来の言語の演算子、関数または文に相当します。

関数パレット ブロックダイアグラムストラクチャ、定数、通信機能および VI を表示するパレット。

さ

サブパレット パレットに含まれている下層のパレット。

サブ VI 異なる VI のブロックダイアグラムで使用される VI で、サブルーチンに相当します。

実行ハイライト VI のデータフローを明確にするため、VI の実行を画像的に示す機能。

指標付け使用 まとまったデータ一式を While ループの端子で解放されるように構築できるオプション。インデックスを無効にすると、While ループが解放するのはループで生成された最終データポイントのみになります。

条件端子 While ループの端子で、この中にはいつているブール値によって VI が次の繰り返しを実行するかどうかが決まります。

ストラクチャ While ループなどのプログラム制御要素。

制御器	対話形式で VI へ、またはプログラムでサブ VI にデータを入力するためのフロントパネルオブジェクト。
制御器パレット	フロントパネルの制御器および表示器を備えたパレット。
接点	関数、ストラクチャおよびサブ VI から構成されるブロックダイアグラムの実行要素。データフロー、ワイヤを参照。
操作ツール	データ操作のほか、データを制御器に入力するために使用するツール。

た

端子	データの受け渡しが行われる接点上のオブジェクトあるいは領域。
ツールパレット	フロントパネルとブロックダイアグラムオブジェクトの編集やデバッグに使用するツールの入ったパレット。
データフロー	実行可能な接点で構成されるプログラミングシステム。接点は必要な入力データをすべて受け取ってからのみ実行され、実行終了後、自動的に出力を生成します。LabVIEW はデータフローシステムです。
トンネル	ストラクチャ上のデータ入力または出力端子。

は

波形チャート	データポイントを一定レートでプロットするインジケータ。
配線ツール	端子間のデータパスを定義するために使用するツール。巻きワイヤに似ています。
パレット	使用可能なオプションを表示する画像メニュー。制御パレット、関数パレット、サブパレット、ツールパレットを参照。
表示器	出力を表示するフロントパネルオブジェクト。
ヒントラベル	オブジェクト、制御器、あるいは端子名を表示する文章見出し。
ブロックダイアグラム	プログラムまたはアルゴリズムを図式的に表記または説明したもの。LabVIEW のブロックダイアグラムは、接点と呼ばれ

る実行可能なアイコンと、接点間のデータを転送するワイヤで構成され、VIのソースコードとなるブロックダイアグラムはブロックのVIダイアグラムウィンドウに常駐しています。

プローブ

VIの中間値をチェックするデバッグ機能。

フロントパネル

VIの対話型ユーザインタフェース。実際の計測器のフロントパネル形状を模倣し、スイッチ、スライド、ゲージ、LEDおよびその他制御器や表示器で構成されます。

分岐ワイヤ

間に接合部がない場合、ワイヤセグメントから接合部まで、ある端子から次の接合部まで、あるいはある端子から別の端子までワイヤセグメント全体が入ったワイヤの部分。

ら

ラベリングツール

ラベルを作成しウィンドウにテキストを入力するために使用するツール。

わ

ワイヤ

接点間のデータパス。データフローの項目を参照。

ワイヤセグメント

1本の縦方向または横方向のワイヤ。

ワイヤ接合部

3本以上のワイヤセグメントの結合点。