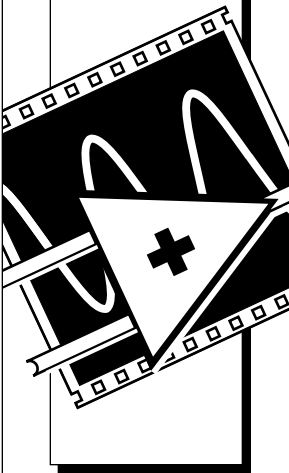


The word "LabVIEW" is written vertically in a large, bold, black serif font. The letters are contained within a thin black rectangular border.

LabVIEW



Guide de Démonstration LabVIEW[®]

Edition de mars 1996
Numéro de série 321292A-01

© Copyright 1993, 1996
National Instruments



National Instruments France

Centre d'Affaires Paris-Nord
BP 217
93153 Le Blanc-Mesnil Cedex
Tél. : (1) 48 14 24 24
Fax : (1) 48 14 24 14
E-mail : info@natinst.com



National Instruments Belgium NV

Leuvensesteenweg 613
B-1930 Zaventem
Tél. : 02/757.03.20
405 120 9 (Luxemburg)
Fax : 02/757.03.11
E-mail : info.belgium@natinst.com



National Instruments Canada

National Instruments Canada
1000 Boulevard St. Jean, Suite 316
Pointe-Claire, Québec H9R 5P1
Tél. : (800) 433-3488
(514) 694-8521
Fax : (514) 694-4399
E-Mail : info@natinst.com



National Instruments Switzerland

National Instruments Switzerland
Sonnenbergstr. 53
CH-5408 Ennetbaden
Tél. : 056/200 51 51
022/980 05 11 (Genève)
Fax : 056/200 51 55
E-mail : ni.switzerland@natinst.com



Filiales

Australia 03 9 879 5166, Austria 0662 45 79 90 0, Belgium 02 757 00 20, Canada (Ontario) 905 785 0085, Canada (Québec) 514 694 8521, Denmark 45 76 26 00, Finland 90 527 2321, France 1 48 14 24 24, Germany 089 741 31 30, Hong Kong 2645 3186, Israel 03 5734815, Italy 02 413091, Japan 03 5472 2970, Korea 02 596 7456, Mexico 95 800 010 0793, Netherlands 0348 433466, Norway 32 84 84 00, Singapore 2265886, Spain 91 640 0085, Sweden 08 730 49 70, Switzerland 056 200 51 51, Taiwan 02 377 1200, U.K. 01635 523545

National Instruments Corporate Headquarters

6504 Bridge Point Parkway Austin, TX 78730-5039 Tél.: (512) 794-0100

Informations supplémentaires

L'objectif de ce manuel est de vous aider à développer des applications en vous servant du logiciel LabVIEW. Si, malgré tout, vous avez besoin d'informations supplémentaires, vous trouverez à votre disposition les supports suivants :

Documentation LabVIEW

LabVIEW vous est fourni avec une documentation complète. Elle comprend des manuels de référence fonctionnels pour toutes les fonctions intégrées, un manuel d'utilisateur illustré d'explications et d'exemples sur les objets des faces-avant, l'environnement LabVIEW, des techniques de programmation ainsi qu'un index général pour vous permettre de localiser des renseignements particuliers.

Répertoire d'exemples

LabVIEW comprend un grand nombre d'exemples tous situés dans le répertoire `EXAMPLES`. Le VI (instrument virtuel) **readme** situé hiérarchiquement à la racine de ce répertoire vous permettra de parcourir des documents en ligne pour comprendre les exemples proposés.

Notes d'application et notes techniques

National Instruments tient gracieusement à votre disposition un très grand nombre de notes techniques et de notes d'application. Ces notes sont regroupées sur le serveur de fichiers BBS et sur le réseau Internet.

Services offerts sur support électronique

Vous avez la possibilité de vous connecter sur le serveur de fichiers BBS de LabVIEW en utilisant un modem ou le réseau Internet pour bénéficier des services suivants :

- support technique et logiciel
- correspondance électronique
- questions des utilisateurs
- publications techniques, notes d'application
- VIs utilitaires pour applications spécifiques
- mises à jour logicielles et nouveaux drivers d'instrument

Support technique en France



Télécopie

Pour bénéficier du support technique par télécopie, veuillez nous communiquer vos nom et prénom, le nom de votre société, la version de LabVIEW et le type de plate-forme sur lesquelles vous travaillez, ainsi que toutes vos questions les plus détaillées possibles.

Composez pour cela le : (33) 1 48 14 24 14.



Téléphone

La possibilité vous est offerte de vous entretenir directement avec nos ingénieurs d'application en composant le : (33) 1 48 14 24 00.



BBS (modem)

Numéro de téléphone	(33) 1 48 65 15 59
Débit en bauds	28800
Bits utiles	8
Bits d'arrêt	1
Parité	aucune
Adresse E-mail	France.support@natinst.com
Site Internet FTP	
Adresse	ftp.natinst.com
Connexion	anonyme
Mot de passe	vosre adresse E-mail

Support technique international



FaxBack

Ce système de consultation d'informations automatique contient des fiches sur les produits, les questions les plus souvent posées ainsi que des notes techniques et d'application. Vous pouvez y accéder sur un combiné téléphonique à boutons-poussoirs. Les documents demandés vous sont ensuite télécopiés. Composez le : 1 (512) 418-1111 ou le 1 (800) 329-7177.



Forums Internet sponsorisés par les utilisateurs

Ces forums permettent aux utilisateurs de communiquer entre eux à propos de LabVIEW

Demandes d'abonnement : info-labview-request@pica.army.mil

Forums : info-labview@pica.army.mil



Adresses E-mail

Il s'agit de boîtes à lettres de support technique gérées par les ingénieurs d'application :

lv.support@natinst.com

lw.support@natinst.com

hiq.support@natinst.com

gpiib.support@natinst.com

daq.support@natinst.com

vxii.support@natinst.com

Informations importantes

Limitations de garantie

Les disquettes sur lesquelles vous recevez les logiciels de National Instruments sont garanties contre les pannes survenant lors de l'exécution de programmes, qui seraient dues à des défauts matériels ou de fabrication. La période de cette garantie est de 90 jours à partir de la date de livraison, attestée par les reçus ou autres documents. Le cas échéant, National Instruments corrigera ou remplacera la disquette qui ne permettrait pas l'exécution normale des programmes, à condition qu'un tel défaut soit stipulé au cours de la période de garantie. National Instruments ne garantit pas que le fonctionnement de ses logiciels ne sera pas interrompu ni se déroulera sans erreur.

Un numéro RMA (Return Material Authorization) d'autorisation de retour de matériel doit être délivré par l'usine et clairement indiqué sur l'emballage du produit afin que celui-ci puisse être accepté pour la réparation sous garantie. National Instruments prendra à sa charge les frais de transport pour renvoyer à son propriétaire les éléments couverts par la garantie.

National Instruments estime avoir fait tout ce qu'il fallait pour que les informations contenues dans ce manuel soient exactes. Ce document a été soigneusement relu pour la précision des informations techniques qu'il contient. Au cas où il resterait malgré tout des erreurs techniques ou des fautes typographiques, National Instruments se réserve le droit d'apporter des modifications aux futures éditions de ce document sans préavis aux détenteurs de cette édition. Le lecteur est prié de prendre contact directement avec National Instruments s'il suspecte des erreurs. En aucun cas, National Instruments ne pourra être tenu responsable des problèmes liés à l'utilisation de ce document ou aux informations qu'il contient.

A L'EXCEPTION DE CE QUI EST SPECIFIE ICI, NATIONAL INSTRUMENTS N'ACCORDE AUCUNE AUTRE GARANTIE, EXPLICITE OU IMPLICITE, ET REJETTE PARTICULIEREMENT TOUTE GARANTIE LIEE A L'ACTE DE VENTE ET A L'ADEQUATION DE SES PRODUITS A UN BESOIN PARTICULIER. LES DROITS DES UTILISATEURS A RECOURIR LES DOMMAGES CAUSES PAR UNE FAUTE OU NEGLIGENCE DE LA PART DE NATIONAL INSTRUMENTS SERONT LIMITEES AUX SOMMES VERSEES PAR L'UTILISATEUR. NATIONAL INSTRUMENTS NE SERA PAS PASSIBLE DE DOMMAGES ET INTERETS A LA SUITE DE PERTES DE DONNEES OU DE PROFITS, OU DE TOUS DOMMAGES (ACCIDENTELS OU NON) LIES A L'UTILISATION DE SES PRODUITS, MEME S'IL EN AVAIT ETE PREALABLEMENT AVERTI. Cette limitation de la responsabilité de National Instruments s'appliquera quelles que soient la nature et l'origine du préjudice, que ce soit à la suite d'un contrat ou la conséquence d'un acte délictueux, y compris par négligence. Toute action contre National Instruments doit être conduite dans l'année qui suit la cause de cette action. National Instruments ne pourra être tenu responsable de tout retard en performance dû à des causes qui iraient au-delà de ce qu'il lui est raisonnablement possible de faire. La garantie fournie ici ne couvre pas les dommages, défauts, dysfonctionnements, ou défauts de service dus à des erreurs commises par l'utilisateur dans l'interprétation des instructions de National Instruments en ce qui concerne l'installation, le fonctionnement et la maintenance. Elle ne couvre pas non plus les négligences, les modifications ou mauvais usages du produit de la part de l'utilisateur, les chutes de tension ou surintensités, le feu, les inondations, les accidents, les agissements de tierces personnes, et tout autre événement incontrôlable.

Copyright

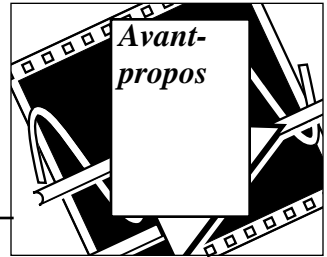
Conformément à la loi sur les droits d'auteurs, ce document ne peut être ni reproduit ni transmis, sous aucune forme que ce soit, informatique ou mécanique, notamment par photocopie, enregistrement, stockage dans un système d'archivage de documentation, ni traduit intégralement ou en partie, sans l'autorisation écrite de National Instruments Corporation.

Marques déposées

LabVIEW[®] et NI-488M[™] sont des marques déposées de National Instruments Corporation. Les produits et noms de sociétés cités sont des marques déposées par leurs propriétaires respectifs.

MISE EN GARDE CONCERNANT L'UTILISATION DES PRODUITS DE NATIONAL INSTRUMENTS DANS LES APPLICATIONS MÉDICALES ET HOSPITALIÈRES

Les produits de National Instruments ne sont pas conçus avec des composants et suivant des méthodes de tests prévus pour assurer un niveau de fiabilité convenant à leur utilisation dans les applications de traitement et de diagnostic sur les personnes. Les applications des produits de National Instruments impliquant des traitements médicaux ou cliniques peuvent potentiellement occasionner des blessures accidentelles à cause d'une panne des produits, ou à cause d'une erreur de la part de l'utilisateur ou du concepteur de l'application. Toute utilisation ou application des produits National Instruments pour ou dans des traitements médicaux ou cliniques doit être effectuée par un personnel médical correctement formé et qualifié, et toutes les garanties médicales d'usage, tous les équipements et toutes les procédures qui sont appropriés à cette situation particulière pour éviter les blessures graves ou la mort, doivent toujours être mis en œuvre lorsque l'on utilise des produits de National Instruments. Les produits de National Instruments N'ONT PAS été conçus pour se substituer à toute forme de procédé, procédure ou équipement utilisée pour la surveillance médicale ou pour garantir la santé publique dans les traitements médicaux et cliniques.



Le *Guide de démonstration LabVIEW* contient les informations dont vous avez besoin pour exploiter les principales fonctions du logiciel LabVIEW (*Laboratory Virtual Instrument Engineering Workbench*). LabVIEW simplifie le développement des applications de traitements scientifiques, de contrôle/commande de processus industriels, de test et mesure électroniques, et de bien d'autres applications en programmation.

Ce guide de démonstration vous offre une brève introduction à LabVIEW en décrivant ses principes de base fondamentaux.

Dans ce manuel, nous partons du principe que vous savez parfaitement vous servir de votre ordinateur personnel et que vous en maîtrisez son système d'exploitation.

Pour une meilleure cohérence entre ce tutorial et le logiciel, certains termes anglais ont été délibérément conservés, notamment ceux qui apparaissent dans les menus et palettes.

Organisation du manuel

Le manuel s'organise de la façon suivante :

- La Préface, *Mise en route de démonstration de LabVIEW*, vous indique comment mettre en route le *Kit de démonstration de LabVIEW* et vous indique les différentes démonstrations qui se trouvent dans le Kit.
- Le chapitre 1, *Introduction à LabVIEW*, décrit ce qu'est LabVIEW, ce qu'est un Instrument Virtuel (VI), comment utiliser l'environnement de LabVIEW (fenêtres, menus, palettes et outils), et vous apprend à faire fonctionner les VIs, les éditer et les créer.
- Le chapitre 2, *La création d'un sous-VI*, traite des sous-VIs, vous apprend à créer leurs icônes et leurs connecteurs et à utiliser les VIs comme des sous-VIs.

- Le chapitre 3, *Boucles et graphes déroulants*, vous présente les boucles *While*, vous apprend à afficher des données sous forme de graphes déroulants, vous renseigne sur les registres à décalage et leur utilisation et vous apprend à utiliser les boucles *For*.
- Le chapitre 4, *Tableaux, clusters et graphes*, traite de la création de tableaux et de l'utilisation des fonctions élémentaires dans les tableaux, les *clusters* et les graphes. Vous y découvrirez également le principe du polymorphisme et la façon d'utiliser les graphes pour afficher des données.
- Le chapitre 5, *Structures Condition, structures Séquence et boîte de calcul*, explique comment utiliser les structures de programmation dites Condition et Séquence et la possibilité d'exploiter les boîtes de calcul (*formula nodes*) classiques.
- Le chapitre 6, *Chaînes de caractères et E/S sur fichier*, vous apprend à créer des commandes et des indicateurs de chaînes de caractères, présente les fonctions de chaînage ainsi que les opérations d'Entrées/Sorties sur fichier. Vous apprendrez aussi à enregistrer les données dans des feuilles de calcul et à entrer et à lire des données dans les fichiers texte.
- Le chapitre 7, *L'acquisition de données* (pour Windows, Macintosh et Sun) *et le contrôle d'instruments*, vous explique comment acquérir des données à partir d'une carte d'acquisition de données, traite de la mise en œuvre de VISA et du GPIB, vous apprend à contrôler une interface de port série à partir de LabVIEW, traite du contrôle d'instrumentation VXI (pour Windows, Macintosh et Sun), explique ce que sont les drivers d'instrument, comment les utiliser et enfin vous explique comment mettre en œuvre un VI **Frequency Response Test**.

Conventions d'écriture et abréviations

La rédaction de ce manuel s'appuie sur les conventions suivantes :

gras

Le texte en gras sert à identifier des éléments des menus, des boutons ou des choix de boîtes de dialogue. Il sert aussi à identifier les paramètres d'entrée et de sortie des VIs.

italique

L'italique sert à mettre en valeur un mot, à présenter une référence croisée, ou un concept essentiel. Certains mots conservés en anglais sont également en italique.

- gras italique*** Le gras italique souligne une remarque, un avertissement ou une mise en garde.
- caractères
courier Une police Courier désigne du texte à taper au clavier. Elle sert aussi à présenter des scripts de programmes, des messages et des réponses que l'ordinateur affiche à l'écran.
- caractères
courier
italique L'italique utilisé dans cette police indique que vous devez fournir les termes ou valeurs appropriés à la place de ces articles.
- < > Les parenthèses angulaires encadrent le nom d'une touche du clavier. Exemple : <Maj>.
- Un trait d'union séparant deux ou plusieurs noms de touches entre parenthèses angulaires signifie qu'il faut appuyer simultanément sur les touches mentionnées. Exemple : <Maj-Suppr> .
- » Le symbole » vous aide à vous déplacer au sein des menus et des boîtes de dialogue jusqu'à l'exécution de la dernière action. La séquence
- File»Page Setup»Options»Substitute Fonts**
- vous invite à ouvrir le menu **File**, à choisir l'option **Page Setup**, à sélectionner **Options** puis, pour finir, à sélectionner l'option **Substitute Fonts** dans la dernière boîte de dialogue.
- chemins Dans ce manuel, les chemins sont indiqués avec des barres obliques inverses (\) pour séparer les noms des lecteurs, répertoires et fichiers, comme dans l'exemple suivant : lecteur\rép1\rép2\monfich

Avertissement : IEEE 488.1 et IEEE 488.1 et IEEE 488.2 se rapportent respectivement aux normes IEEE 488.2ANSI/IEEE 488.1-1987 et ANSI/IEEE 488.2-1987, qui définissent le

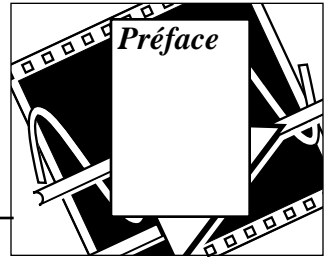


Remarque : *Cette icône située à gauche d'un texte en gras italique introduit une remarque importante.*

Cours de formation

National Instruments vous propose des cours élémentaires de formation ainsi que des cours de perfectionnement pour vous aider à maîtriser rapidement LabVIEW et à développer vos applications avec succès. Le Cours Élémentaire vous enseigne non seulement les principes fondamentaux de LabVIEW, mais il vous apprend également à développer des applications d'acquisition de données et de contrôle d'instruments. Quant au Cours de perfectionnement, il vous apprend à augmenter les performances et l'efficacité de vos applications développées avec LabVIEW. N'hésitez pas à contacter National Instruments pour obtenir le catalogue complet des cours, avec les tarifs et les dates correspondants.

Manuel d'utilisation du Kit de démonstration LabVIEW

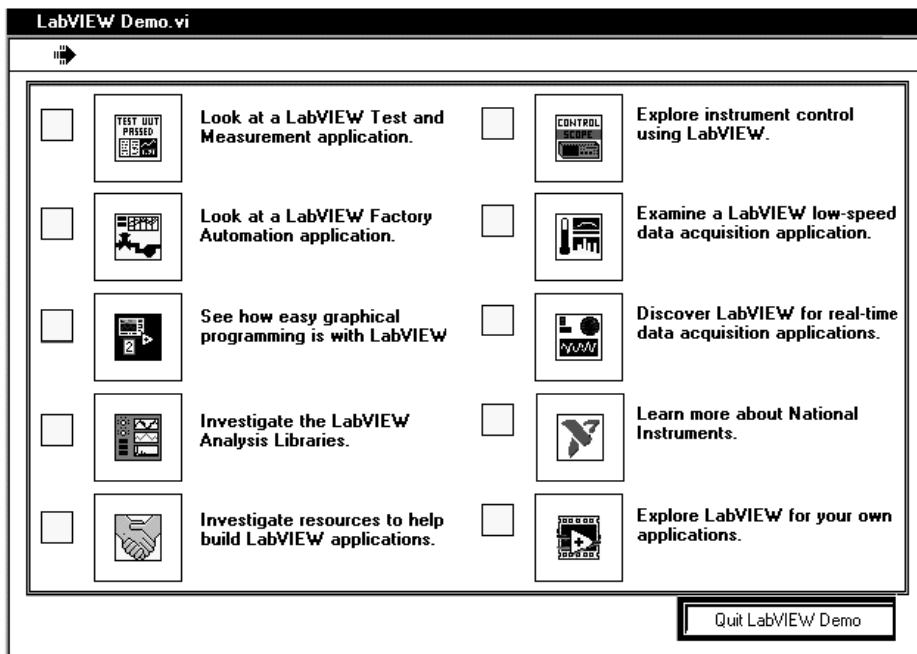


Cette préface vous explique la mise en œuvre du logiciel de démonstration LabVIEW et détaille les différentes démonstrations qu'il contient.



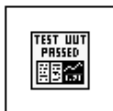
Ouvrez votre logiciel de démonstration LabVIEW en cliquant sur l'icône LabVIEW du dossier démonstration LabVIEW. Lorsque vous ouvrez le logiciel de démonstration LabVIEW, le VI **LabVIEW Demo** apparaît automatiquement à l'écran.

Ce VI de démonstration vous présente rapidement les nombreux moyens mis à votre disposition par LabVIEW pour répondre à vos besoins logiciels. Un clic sur un bouton de catégorie vous conduit à une démonstration spécifique d'un programme exécuté dans LabVIEW. Le programme coche chacune des catégories auxquelles vous avez déjà accédé, à titre de rappel.



Un clic sur une catégorie appelle un exemple d'application entièrement écrite avec LabVIEW. Vous pouvez parcourir ces applications de démonstration et observer leur fonctionnement. Pour un complément d'information sur la présente démonstration, cliquez sur le bouton bleu **More Info...** en bas de l'application, ou appuyez sur la touche <F5>. Pour retourner au menu principal de la démonstration, cliquez sur le bouton rouge **Return** qui se trouve aussi en bas de chaque application, ou appuyez sur la touche <F4>.

La liste ci-dessous décrit chaque catégorie du VI de démonstration.



Observez une application de test et de mesure LabVIEW.

Présente *LabVIEW Test Executive*, une application développée dans LabVIEW que vous pouvez utiliser afin de contrôler les tests pour des applications de production et de fabrication.



Observez une application d'automatisation industrielle LabVIEW.

Simule une application de contrôle/commande de processus, créée dans LabVIEW.



Constatez la facilité de la programmation LabVIEW.

Aborde la programmation graphique et vous présente les bases de la création des VIs (instruments virtuels) dans LabVIEW.



Explorez les bibliothèques d'analyse LabVIEW.

Présente les bibliothèques d'analyse LabVIEW et leur utilisation pour la création de systèmes d'analyse.



Parcourez les ressources d'aide à la création de vos applications LabVIEW.

Vous expose les nombreuses ressources de développement disponibles dans votre logiciel LabVIEW, comme les boîtes à outils, les sources d'information, les formations et les options de support technique de LabVIEW.



Explorez le contrôle d'instruments avec LabVIEW.

Vous indique comment utiliser la souplesse de l'interface LabVIEW pour des applications de contrôle d'instruments.



Examinez une application d'acquisition de données LabVIEW à faible vitesse.

Présente l'utilisation de LabVIEW avec des périphériques d'acquisition de données pour contrôler, collecter et analyser des données.



Découvrez LabVIEW pour les applications d'acquisition de données en temps réel.

Vous démontre pourquoi LabVIEW est particulièrement performant pour des applications d'acquisition des données à haute vitesse, en vous offrant un maximum de puissance et de souplesse avec vos périphériques d'acquisition.



Ayez une meilleure connaissance de National Instruments.

Vous présente rapidement National Instruments et vous indique comment trouver le bureau National Instruments le plus proche de vous.



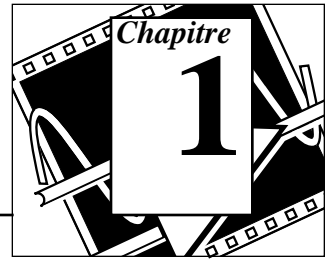
Explorez LabVIEW pour vos propres applications.

Vous voulez en savoir davantage ? Cette option vous indique comment approfondir votre connaissance de LabVIEW grâce au *Guide de démonstration LabVIEW*.

Vous êtes maintenant prêt à vous familiariser avec l'ensemble des possibilités offertes par le logiciel LabVIEW. Pour approfondir votre connaissance de la programmation LabVIEW, reportez-vous au Chapitre 2, *Introduction à LabVIEW*, de ce guide.

Si vous souhaitez explorer d'autres démonstrations complètes de LabVIEW, ouvrez la bibliothèque demos . 11b et choisissez-y un VI de démonstration. Cliquez ensuite sur le bouton Exécution pour lancer le VI.

Introduction à LabVIEW



Ce chapitre décrit ce qu'est LabVIEW, ce qu'est un Instrument Virtuel (VI), comment utiliser l'environnement LabVIEW (les fenêtres, les menus, les palettes et les outils), comment exécuter les VIs, les éditer et les créer.



Avant de commencer à effectuer les opérations associées aux objectifs de ce chapitre, veuillez cliquer sur Explore LabVIEW pour obtenir votre propre option d'application dans le VI de démonstration LabVIEW Demo.

LabVIEW étant un outil de développement de programmes aux possibilités particulièrement étendues, ce guide de démonstration ne prétend pas vous expliquer comment résoudre tous les problèmes liés à la programmation. Il vous enseigne plutôt les principes fondamentaux de LabVIEW, vous familiarise avec les outils de programmation du logiciel et vous fournit brièvement des exemples pratiques d'utilisation, ceux-ci se voulant représentatifs des problèmes de programmation auxquels vous serez probablement confronté.

Si, après avoir étudié ce manuel, vous souhaitez une formation complémentaire, sachez que National Instruments propose des cours de formation pour vous aider à maîtriser rapidement LabVIEW et à développer des applications opérationnelles.

L'assimilation du Cours Élémentaire de LabVIEW vous familiarisera avec les aspects fondamentaux du logiciel, tout en vous offrant la possibilité de développer vous-même des applications concrètes de contrôle d'instruments et d'acquisition de données (pour les plates-formes Windows, Macintosh et Sun). Si vous décidez ensuite de suivre le cours de perfectionnement de LabVIEW, vous apprendrez à optimiser les performances et l'efficacité des applications du logiciel, tout en assimilant ses caractéristiques les plus avancées.

Pour obtenir le catalogue complet des cours de formation, la date des sessions ainsi que les tarifs, veuillez vous reporter à l'adresse figurant sur la deuxième page de couverture de ce manuel pour contacter National Instruments.

Informations relatives au chapitre

Au début de chaque chapitre, vous trouverez un encadré tel que celui-ci, dans lequel sont indiqués les différents points abordés dans le chapitre considéré.

Vous allez apprendre :

- Ce qu'est LabVIEW.
- Ce qu'est un Instrument Virtuel (VI).
- Comment utiliser l'environnement LabVIEW (fenêtres et palettes).
- Comment exécuter des VIs.
- Comment éditer des VIs.
- Comment créer des VIs.

Qu'est-ce que LabVIEW ?

LabVIEW est un logiciel de développement d'applications, comparable à la plupart des systèmes de développement en langage C ou BASIC disponibles sur le marché, ou encore à LabWindows de National Instruments. Cependant, LabVIEW se distingue des autres logiciels sur au moins un point important. En effet, la majorité d'entre eux s'articulent autour de langages à base de texte dont la programmation consiste à empiler des lignes de code, tandis que LabVIEW utilise un langage de programmation graphique, le *langage G*, pour créer un programme sous forme de diagramme.

Inutile d'être expert en programmation pour pouvoir utiliser LabVIEW. La terminologie, les icônes et les principes inhérents à LabVIEW, tous familiers aux ingénieurs et aux scientifiques, font appel à des symboles graphiques pour décrire les opérations de programmation.

LabVIEW offre des bibliothèques étendues de fonctions et de routines (blocs pré-programmés) capables de répondre à la plupart des besoins en programmation. En ce qui concerne les plates-formes Windows, Macintosh et Sun, LabVIEW comprend également des bibliothèques de fonctions spécifiques à l'acquisition de données et au pilotage d'instruments VXI et GPIB, ou encore d'instruments connectés sur une simple liaison série. Il existe aussi des bibliothèques dédiées à la

présentation, à l'analyse et au stockage des données. LabVIEW intègre une panoplie complète d'outils de développement de programme conventionnels, de sorte que vous pouvez définir des points d'arrêt, animer l'exécution du programme en mettant en évidence le cheminement des données et exécuter pas à pas votre programme. Le développement et la mise au point du programme s'en trouvent ainsi facilités.

Comment fonctionne LabVIEW ?

LabVIEW comprend des bibliothèques de fonctions et des outils de développement spécialement conçus pour les applications de contrôle d'instruments. Pour les plates-formes Windows, Macintosh et Sun, LabVIEW intègre également des bibliothèques de fonctions et des outils de développement pour les applications d'acquisition de données. Un programme LabVIEW est appelé *instrument virtuel* (VI) tout simplement parce que sa représentation et son fonctionnement ressemblent à ceux des instruments classiques. Néanmoins, les VIs diffèrent en ce sens qu'ils tirent leur fonctionnalité de la programmation informatique. Ils offrent une interface utilisateur interactive avec l'équivalent en code source, et acceptent les paramètres des VIs de niveau supérieur. Ces points méritent une explication complémentaire.

- Un VI intègre une interface utilisateur interactive appelée *face-avant*, puisqu'elle simule la face-avant d'instruments physiques. La face-avant contient des boutons rotatifs, des boutons-poussoirs, des graphes et autres commandes et indicateurs. Vous saisissez les données à l'aide du clavier ou de la souris, puis vous visualisez les résultats à l'écran.
- Un VI reçoit des instructions de son *diagramme*, que vous construisez en langage graphique G. Le diagramme, qui correspond au code source du VI, réduit ainsi la programmation à une simple manipulation graphique.
- Le VI présente une structure hiérarchique et modulaire. Vous pouvez l'utiliser comme un programme principal ou comme un sous-programme à l'intérieur d'autres programmes ou de sous-programmes. Un VI contenu à l'intérieur d'un autre VI s'appelle un *sous-VI*. Le *cadre icône/connecteur* d'un VI répertorie sous forme graphique tous ses paramètres si bien que d'autres VIs peuvent lui transmettre des données, en le considérant comme un sous-VI.

Avec de telles caractéristiques, LabVIEW adhère entièrement au concept de *programmation modulaire*. Ainsi, vous scindez une application en une série de tâches que vous pouvez subdiviser autant de fois que nécessaire jusqu'à ce qu'une application complexe soit ramenée à une série de tâches élémentaires, faciles à mettre en œuvre. Vous construisez un VI pour chaque tâche puis rassemblez les VIs ainsi réalisés dans un diagramme pour leur faire exécuter une tâche plus complexe. Au final, votre VI principal contient un ensemble de sous-VIs qui représentent les fonctions de l'application.

En tant qu'entité opérationnelle à part entière, chaque VI peut fonctionner indépendamment du reste de l'application, ce qui facilite la mise au point de celle-ci. En outre, de nombreux sous-VIs de bas niveau sont souvent réutilisables dans plusieurs applications. Vous pouvez donc développer une gamme de sous-VIs spécialisés, adaptés aux applications que vous envisagez de réaliser.

La palette Tools

LabVIEW utilise une palette **Tools** flottante, que vous pouvez utiliser pour modifier et mettre au point les VIs. Utilisez la touche <Tab> pour passer d'un outil à un autre parmi les plus souvent utilisés. Si vous avez fermé la palette **Tools**, sélectionnez **Windows»Show Tools Palette** pour l'afficher. L'illustration suivante représente la palette **Tools**.



outil Doigt : permet de positionner des éléments des palettes **Controls** et **Functions** sur la face-avant et dans le diagramme



outil Flèche : permet de positionner, redimensionner et sélectionner les objets



outil Texte : permet de modifier du texte et d'en créer



outil Bobine : permet de câbler des objets entre eux dans le diagramme



outil Menu local : fait apparaître un menu local pour un objet



outil Main : fait défiler toute la fenêtre sans avoir recours aux barres de défilement



outil Point d'arrêt : permet de définir des points d'arrêt dans les VIs, les fonctions, les boucles, les séquences et les structures



outil Sonde : permet de créer des sondes sur les fils



outil Pipette : copie les couleurs pour les coller à l'aide de l'outil Pinceau



outil Pinceau : permet de définir la couleur de l'arrière-plan et du premier plan

La palette Controls

La palette **Controls** est une palette graphique flottante qui s'ouvre automatiquement dès que vous lancez LabVIEW. Elle vous permet de placer les commandes et les indicateurs dans la face-avant d'un VI. Chaque icône principale contient des sous-palettes. Si la palette **Controls** n'est pas affichée, vous pouvez la rendre visible en sélectionnant **Windows»Show Controls Palette** dans le menu de la face-avant. Vous pouvez également ouvrir un menu local dans une zone libre de la face-avant pour accéder à une copie temporaire de la palette **Controls**.

L'illustration suivante présente le niveau principal de la palette **Controls**.

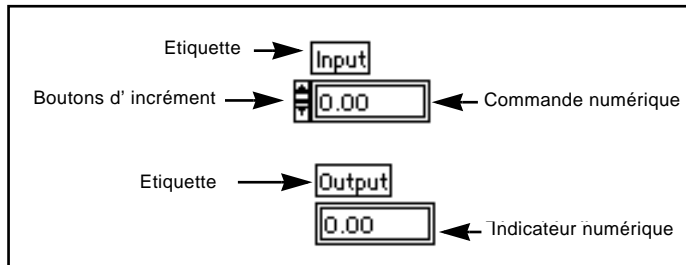


Commandes et indicateurs

Commandes et indicateurs numériques

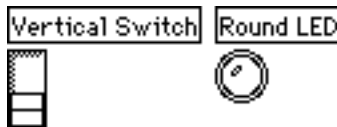
Les commandes numériques servent à entrer des quantités numériques, alors que les indicateurs numériques affichent d'autres quantités

numériques. Les deux objets les plus souvent utilisés sont les *commandes numériques* et les *indicateurs numériques*.



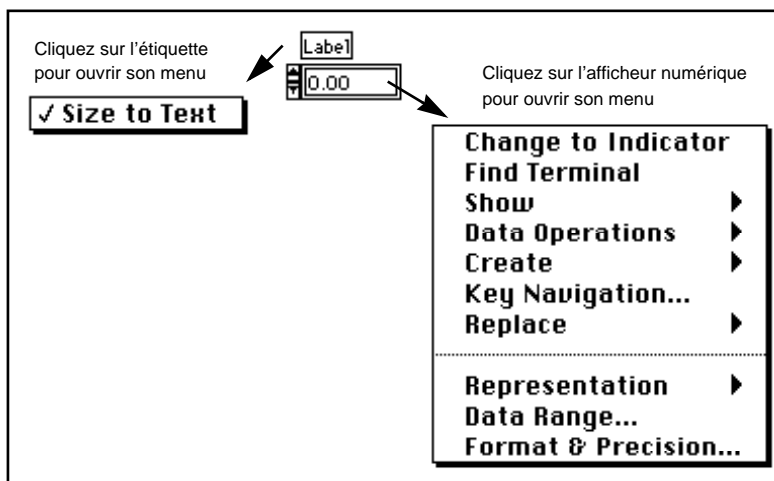
Commandes et indicateurs booléens

Les commandes et les indicateurs booléens servent à saisir et à afficher les valeurs booléennes (TRUE/FALSE). Les objets booléens simulent des interrupteurs, des boutons et des voyants lumineux. Les objets booléens les plus souvent utilisés sont les *interrupteurs verticaux* et les *voyants lumineux ronds*.



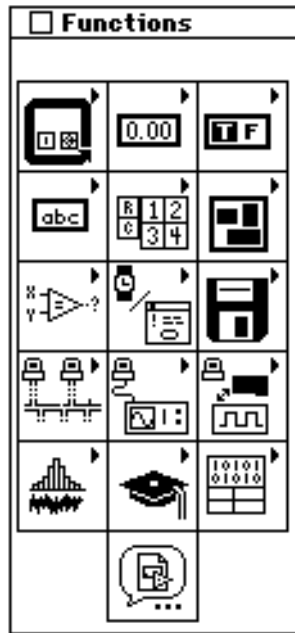
Configuration des commandes et des indicateurs

Vous pouvez configurer pratiquement tous les indicateurs et commandes en utilisant les options de leurs menus locaux respectifs. *Le seul fait d'ouvrir un menu local sur les éléments des commandes et des indicateurs permet d'afficher des menus servant à les personnaliser.* La façon la plus simple d'accéder au menu local consiste à cliquer sur l'outil Menu local, présenté sur la gauche, sur tous les objets dotés d'un menu local. La figure suivante illustre cette méthode d'affichage pour une commande numérique.



La palette Functions

La palette **Functions** est une palette graphique flottante qui s'ouvre automatiquement dès que vous passez dans le diagramme d'un VI. Cette palette sert à placer des nœuds (constantes, indicateurs, VIs, etc.) dans le diagramme d'un VI. Toutes les icônes principales contiennent des sous-palettes. Si la palette **Functions** n'est pas affichée, sélectionnez **Windows»Show Functions Palette** dans le menu du diagramme pour la rendre visible. Vous pouvez également ouvrir un menu local sur une zone libre du diagramme pour accéder à une copie temporaire de la palette **Functions**. La figure suivante est une représentation du niveau principal de la palette **Functions**.



La construction d'un VI

OBJECTIF Construire un VI qui simule l'acquisition de mesure de température.

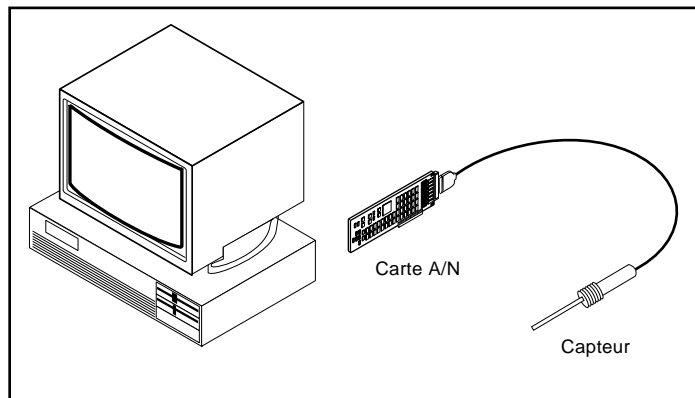


*Avant de commencer cet exercice, veuillez cliquer sur **Explore LabVIEW** pour votre propre option d'application dans le VI de démonstration **LabVIEW Demo**.*

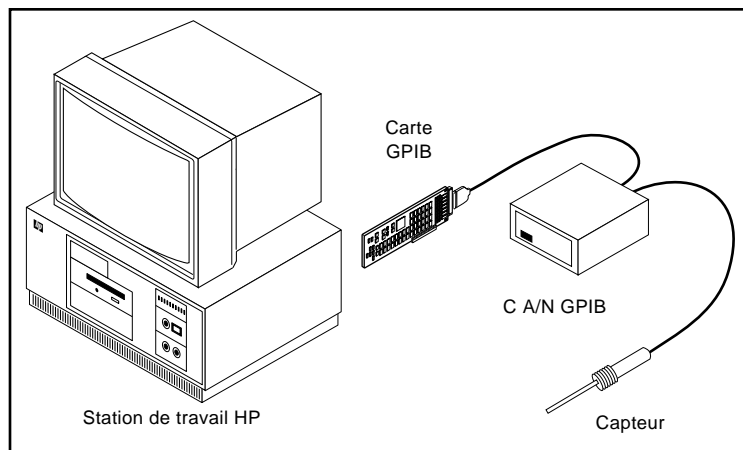
Vous utiliserez le VI **Demo Voltage Read** pour mesurer la tension, puis multipliez le résultat par 100.0 pour convertir la tension en une température exprimée en degrés (Fahrenheit).

Imaginez que vous disposez d'un transducteur ou d'un capteur qui convertit la température en tension.

(Windows, Macintosh et Sun) Ce capteur est connecté à une carte de convertisseur analogique-numérique (C A/N), comme dans l'exemple reproduit ici, qui convertit la tension en données numériques.



(HP-UX) Le capteur pourrait aussi bien être connecté à un convertisseur analogique-numérique connecté à l'ordinateur via une interface GPIB, comme dans l'illustration suivante. Cette méthode permet aussi de convertir la tension en données numériques.



La face-avant

1. Pour ouvrir une nouvelle face-avant, sélectionnez **File»New**. Pour les plates-formes Windows et UNIX, si vous avez fermé tous les VIs, choisissez alors **New VI** dans la fenêtre de dialogue LabVIEW.



Remarque : *si la palette Controls n'est pas affichée, choisissez Windows»Show Controls Palette pour la rendre visible. Vous pouvez également accéder à la palette Controls en ouvrant un menu local dans une zone libre de la face-avant.*



2. Sélectionnez un indicateur **Thermometer** dans **Controls»Numeric**, puis placez-le dans la face-avant.
3. Tapez Temp à l'intérieur de l'étiquette puis cliquez sur le bouton Enter de la barre d'outils.

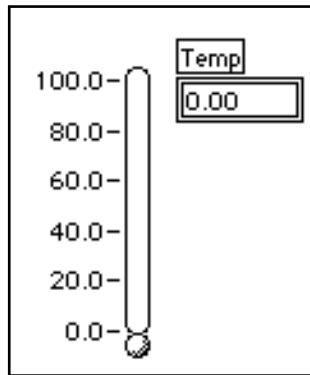


Remarque : *si vous cliquez en dehors de la boîte de dialogue sans entrer de texte, le champ de l'étiquette disparaît. Vous avez la possibilité de le faire réapparaître en ouvrant un menu local sur la commande puis en choisissant Show»Label.*

N'oubliez pas que pour ouvrir un menu local, vous devez cliquer sur le bouton droit de la souris (<command> -click pour Macintosh).



4. Modifiez l'échelle du thermomètre pour afficher la température entre 0.0 et 100.0.
 - a. Pour cela, avec l'outil Texte, double-cliquez sur 10.0 dans l'échelle du thermomètre pour mettre cette valeur en évidence.
 - b. Tapez 100.0 dans l'échelle puis cliquez avec le bouton de la souris n'importe où en dehors de la fenêtre d'affichage. LabVIEW met automatiquement à l'échelle les valeurs intermédiaires. Le thermomètre devrait maintenant ressembler à celui de l'illustration suivante.



Le diagramme

1. Pour ouvrir le diagramme, choisissez les options **Windows»Show Diagram**. Sélectionnez les objets qui vont composer le diagramme dans la palette **Functions**. Pour chaque objet à insérer, sélectionnez l'icône puis l'objet dans le niveau principal de la palette, ou choisissez l'objet dans la sous-palette appropriée. Lorsque vous placez le pointeur de la souris dans le diagramme, LabVIEW affiche les contours de l'objet. C'est à l'utilisateur de choisir l'endroit où l'objet sera placé dans le diagramme.



Remarque : *si la palette Functions n'est pas affichée, sélectionnez Windows»Show Functions Palette pour la rendre visible. Vous pouvez également accéder à la palette Functions en ouvrant un menu local dans une zone libre du diagramme.*



Placez chacun des objets suivants dans le diagramme.

Le VI **Demo Voltage Read (Functions»Tutorial)** simule la mesure d'une tension acquise par une carte d'acquisition de données.



Fonction **Multiply (Functions»Numeric)**. Dans cet exercice, la fonction multiplie par 100.0 la tension donnée par le VI **Demo Voltage Read**.



Numeric constant (Functions»Numeric). Deux constantes numériques sont nécessaires : une pour le facteur d'échelle 100 et une autre pour la constante du périphérique. Pour la première constante numérique, tapez 100.0 dès que la constante s'affiche dans le diagramme.



2. Créez la seconde constante numérique en utilisant une touche de raccourci. Vous créez et câblez ainsi automatiquement la constante au VI **Demo Voltage Read**.
 - a. A l'aide de l'outil Bobine, ouvrez un menu local sur l'entrée Board ID du VI **Demo Voltage Read** puis sélectionnez **Create Constant** dans le menu local. Cette option permet de créer automatiquement une constante numérique et de la câbler au VI **Demo Voltage Read**.
 - b. Tapez 1 lorsque la constante apparaît pour la première fois dans le diagramme. Cette manipulation permet de modifier la valeur par défaut qui passe de 0 à 1.



Remarque : *vous remarquez qu'il n'est pas nécessaire de modifier l'outil Texte pour insérer la valeur en procédant de cette manière.*

- c. A l'aide de l'outil Texte, modifiez le texte de l'étiquette active (Board ID) en Device.

Dans cet exemple, les deux nombres représentent la constante 100.0 et le périphérique pour la fonction **Multiply**.



3. Placez un **String Constant (Functions»String)** dans votre diagramme.
4. Ouvrez un menu local sur l'entrée libellée Channel, en bas à gauche du VI **Demo Voltage Read** puis sélectionnez **Create Constant** dans le menu local. Cette option permet de créer automatiquement une constante chaîne de caractères et de la câbler au VI **Demo Voltage Read**.
5. Tapez 0 dès que la constante apparaît. Cliquez sur la constante et sélectionnez **Show »Label**. Dans cet exemple, vous remarquerez que le terme Channel s'affiche dans l'étiquette active si bien qu'il n'est pas nécessaire de modifier l'étiquette.

Dans cet exemple, vous utilisez une constante chaîne de caractères pour indiquer le numéro du canal.



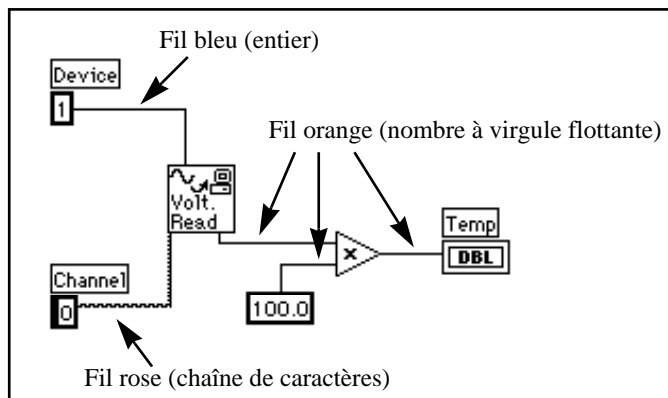
Remarque : *vous pouvez créer et câbler des commandes, des constantes et des indicateurs avec la plupart des fonctions. Si ces options ne sont pas disponibles pour une fonction en particulier, les options **Create Control**, **Create Constant** et **Create Indicator** sont désactivées dans le menu local. Pour plus de détail sur cette caractéristique, veuillez vous reporter à la section *Création et câblage des commandes, constantes et indicateurs ultérieurement dans ce chapitre.**



6. A l'aide de l'outil Bobine, câblez les objets restants conformément aux explications données dans la section intitulée *Techniques de câblage*, plus loin dans ce chapitre.

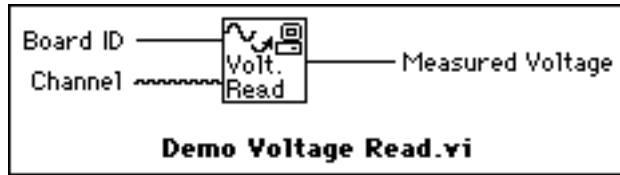


Remarque : *pour déplacer les objets dans le diagramme, cliquez sur l'outil Flèche dans la Palette Tools.*



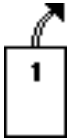
Toutes les connexions LabVIEW sont identifiées par une couleur, laquelle est fonction du type de données acheminées. Ainsi les fils bleus véhiculent des nombres entiers, les fils orange des nombres à virgule flottante, les fils verts des nombres booléens et les fils roses des chaînes de caractères.

Vous pouvez activer la fenêtre d'aide en choisissant **Help»Show Help**. En plaçant n'importe quel outil d'édition sur un nœud, vous visualisez les entrées et les sorties de cette fonction dans la fenêtre d'aide. Au moment où vous passez l'outil d'édition sur l'icône du VI, LabVIEW met en surbrillance les broches de terminaison dans les fenêtres du diagramme et d'aide. Lorsque vous commencerez à câbler vos diagrammes, cette mise en évidence vous aidera à connecter les entrées et les sorties aux bons terminaux.



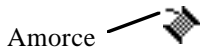
Le VI **Demo Voltage Read** simule la mesure d'une tension acquise sur le canal 0 d'une carte d'acquisition enfichable fournissant des données artificielles à la sortie **Measured Voltage**. Ces données représentent la température réelle divisée par 100. Le VI multiplie alors la tension par 100.0 pour la convertir en une température exprimée en degrés Fahrenheit.

Les techniques de câblage



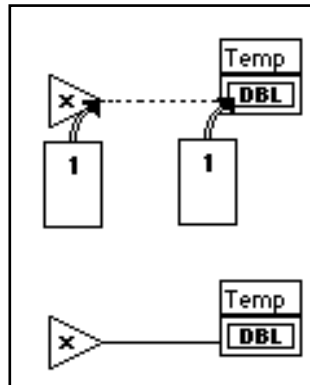
Dans les exemples de câblage reproduits dans cette section, la flèche située à l'extrémité du symbole de la souris indique l'endroit où cliquer et le chiffre figurant sur le bouton de la souris indique le nombre de fois qu'il faut cliquer.

L'*amorce* (ou *point actif*) de l'outil est l'extrémité du morceau de fil dévidé.



Pour connecter un terminal à un autre, cliquez avec l'outil Bobine sur le premier terminal, déplacez la bobine sur le second terminal, puis cliquez dessus. Vous pouvez commencer par n'importe quel terminal, cela n'a aucune importance.

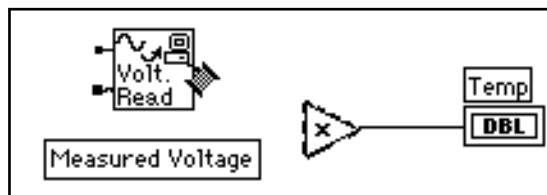
Lorsque l'outil Bobine se trouve au-dessus d'un terminal, la zone concernée clignote, signalant ainsi que vous pouvez cliquer pour établir la connexion. Surtout, ne maintenez *pas* le bouton de la souris enfoncé lorsque vous déplacez l'outil Bobine d'un terminal à un autre. Vous pouvez infléchir la direction d'un fil une seule fois en faisant glisser la souris perpendiculairement à la direction initiale. Pour infléchir la direction du fil plusieurs fois, cliquez sur le bouton de la souris. Pour modifier l'orientation du fil, appuyez sur la barre d'espace. Cliquez sur le bouton de la souris pour *maintenir en place* le fil et faites glisser la souris perpendiculairement.



Info-bulles



Lorsque vous placez l'outil Bobine au-dessus du terminal d'un nœud, une info-bulle apparaît. Il s'agit de fanions jaunes qui reprennent le nom de chaque terminal. Ces info-bulles vous aideront à câbler les terminaux. La figure suivante illustre l'info-bulle (**Measured Voltage**) qui apparaît lorsque vous placez l'outil Bobine au-dessus de la sortie du VI **Demo Voltage Read**.



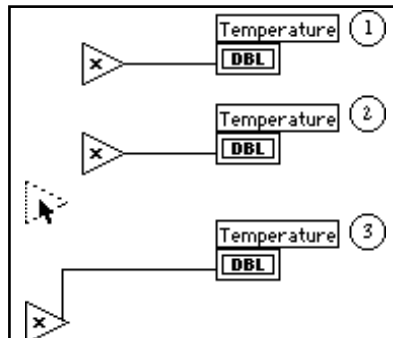
Remarque : lorsque vous placez l'outil Bobine sur un nœud, LabVIEW affiche des mauvaises connexions pour signaler chaque entrée et chaque sortie. Une mauvaise connexion avec un point à son extrémité représente l'entrée à un nœud.

Visualisation des terminaux

Il est important que, pour chaque fonction, les connexions soient effectuées aux bons endroits. Pour vous faciliter le travail, vous pouvez visualiser le connecteur de chaque icône. Pour ce faire, ouvrez le menu local correspondant à la fonction et choisissez **Show Terminals**. Pour revenir à l'icône, ouvrez le menu local de la fonction et sélectionnez de nouveau **Show Terminals**.

Adaptation des liaisons

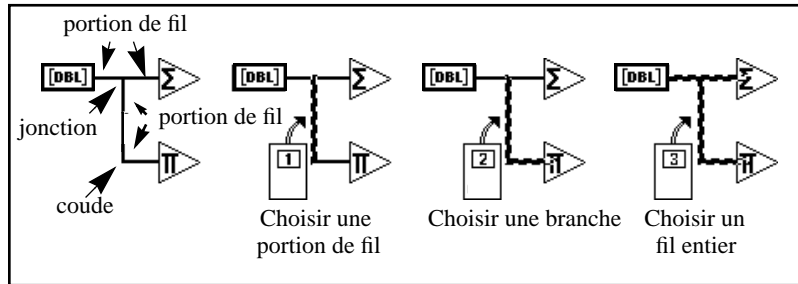
Vous pouvez déplacer les objets câblés soit individuellement soit en groupe en faisant glisser les objets sélectionnés avec l'outil Flèche.



Sélection et suppression des liaisons

Il peut vous arriver de faire une erreur dans le câblage des nœuds. Le cas échéant, sélectionnez le fil à supprimer puis appuyez sur la touche <Suppr>. Une *portion* de fil désigne la partie horizontale ou verticale d'une connexion. Le point où se rejoignent les trois ou quatre portions de fil s'appelle une jonction. Une *branche* rassemble toutes les portions de fil entre les jonctions, d'un terminal à la prochaine jonction, ou d'un terminal à un autre s'il n'existe aucune jonction entre les deux. Pour sélectionner une portion de fil, cliquez dessus avec l'outil Flèche. Un double clic sélectionne une branche entière, un triple clic sélectionne la totalité du fil.



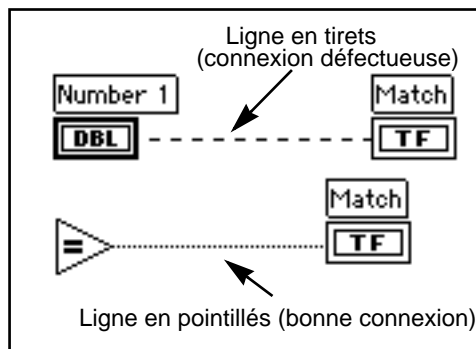


Mauvaises liaisons

Une ligne constituée de tirets représente une connexion défectueuse qui peut avoir plusieurs origines. Il peut s'agir d'une connexion entre deux organes de commande, ou d'une connexion entre un terminal source et un terminal de destination alors que les données sont incompatibles (par exemple, des données numériques et des données booléennes). Vous pouvez supprimer une mauvaise connexion en cliquant dessus à l'aide de l'outil Flèche puis en appuyant sur la touche <Suppr>. En choisissant les options **Edit>Remove Bad Wires**, vous supprimez toutes les connexions défectueuses du diagramme. Vous vous en servirez si votre VI refuse de fonctionner correctement ou s'il renvoie le message d'erreur Signal has loose ends.

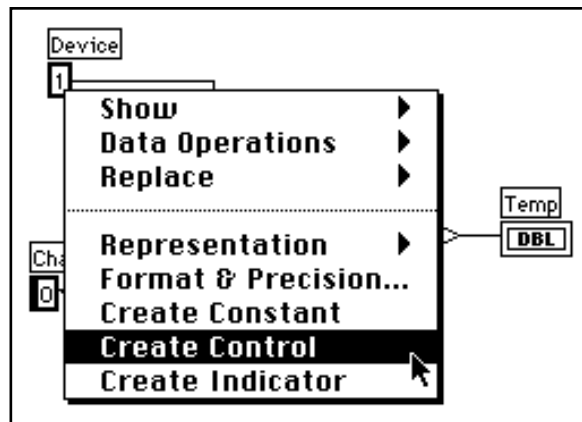


Remarque : *ne confondez pas une ligne en tirets avec une ligne en pointillés. Une ligne en pointillés représente une information de type booléen comme dans l'illustration suivante.*



Création et câblage des commandes, constantes et indicateurs

Pour les terminaux fonctionnant comme des entrées dans le diagramme, LabVIEW offre deux fonctionnalités que vous pouvez utiliser pour créer et câbler une commande ou une constante. Vous accédez à ces fonctionnalités en ouvrant un menu local sur le terminal et en choisissant **Create Control** ou **Create Constant**. LabVIEW crée et câble automatiquement le bon type de commande ou de constante à l'entrée du terminal. L'illustration suivante est un exemple de menu local.



Pour un terminal fonctionnant comme une sortie dans le diagramme, vous pouvez choisir la fonctionnalité **Create Indicator** pour créer puis câbler un indicateur au terminal. Vous accédez à cette fonctionnalité en ouvrant un menu local sur le terminal et en choisissant **Create Indicator**. LabVIEW crée et câble automatiquement le bon type d'indicateur à la sortie du terminal.



Remarque : *Une fois choisi Create Indicator, vous devez passer à la face-avant et utiliser l'outil Flèche pour sélectionner ou supprimer l'indicateur.*

Exécution du VI

1. Activez la face-avant en cliquant sur la barre de titre de la fenêtre ou en choisissant les options **Windows»Show Panel**. Pour les plates-formes Windows et Macintosh, vous pouvez également activer la face-avant en cliquant n'importe où sur celle-ci.



2. Lancez le VI en cliquant sur le bouton Exécution de la barre d'outils de la face-avant.

Vous remarquerez que vous devez relancer le VI à chaque fois. Si vous souhaitez une exécution permanente, vous devez cliquer sur le bouton Exécution permanente.



3. Cliquez sur le bouton Exécution permanente de la barre d'outils.
4. Cliquez une deuxième fois sur le bouton Exécution permanente pour le désactiver. Le VI termine alors l'exécution et se ferme.

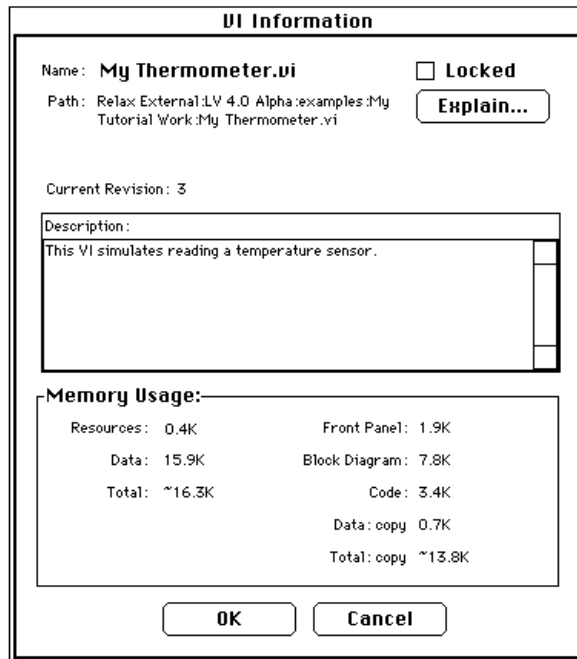


Remarque : *le bouton Exécution permanente n'est pas le meilleur moyen pour renouveler indéfiniment l'exécution du code du diagramme. Il est préférable d'utiliser une structure de bouclage. Cet aspect est abordé dans le chapitre 3, Boucles et graphes déroulants, de ce guide de démonstration.*

Documentation d'un VI

Vous pouvez documenter un VI en choisissant **Windows»Show VI Info...** Tapez le texte de la description du VI dans la boîte de dialogue **Informations VI**. Vous pouvez alors rappeler ce texte en choisissant à nouveau **Windows»Show VI Info...**

1. Documentez le VI. Sélectionnez **Windows»Show VI Info...** Tapez le texte de la description du VI, conformément à l'exemple suivant, puis cliquez sur **OK**.

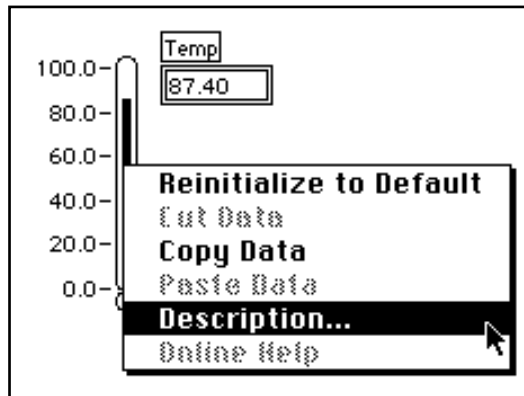


Vous avez la possibilité de visualiser la description des objets sur la face-avant (ou leurs terminaux respectifs dans le diagramme) en ouvrant un menu local sur l'objet et en choisissant **Data Operations»Description...**

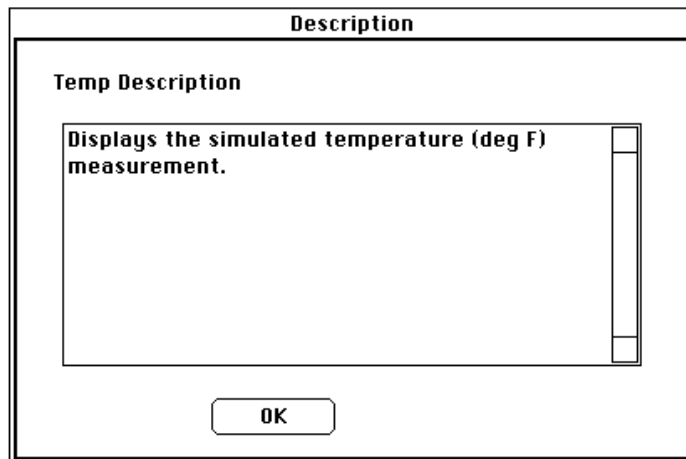


Remarque : *il est impossible de modifier le texte de la description tout en exécutant un VI.*

L'illustration suivante est un exemple de menu local qui s'affiche pendant que vous exécutez un VI. Vous ne pouvez ajouter ni modifier le texte de la description pendant l'exécution du VI. En revanche, vous pouvez visualiser toutes les informations préalablement enregistrées.



2. Documentez l'indicateur thermomètre.
 - a. Dans la face-avant, ouvrez un menu local sur l'indicateur thermomètre et choisissez **Data Operations»Description...**
 - b. Tapez le texte de la description du thermomètre, conformément à l'exemple suivant, puis cliquez sur **OK**.



- Affichez le texte de la description que vous venez de créer en ouvrant un menu local sur l'indicateur thermomètre et en choisissant **Data Operations»Description...**

Enregistrement et chargement des VIs

A l'instar de toutes les autres applications, vous pouvez enregistrer votre VI dans un fichier ou un répertoire traditionnel. Avec LabVIEW, vous pouvez également enregistrer plusieurs VIs dans un fichier unique appelé *bibliothèque de VIs*. La bibliothèque `tutorial.llb` en est un exemple.

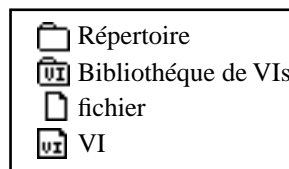
Si vous utilisez Windows 3.1, nous vous recommandons d'enregistrer vos VIs dans les bibliothèques de VIs car vous pouvez utiliser des noms de fichier assez longs, (jusqu'à 255 caractères) en majuscules et en minuscules.

Nous vous déconseillons d'utiliser les bibliothèques de VIs sauf si vous devez transférer vos VIs dans Windows 3.1. Enregistrer les VIs sous la forme de fichiers individuels est une opération plus rentable car vous avez ainsi la possibilité de copier, renommer et supprimer plus facilement les fichiers que si vous utilisiez une bibliothèque de VIs.

Cela étant, il est bon que vous sachiez comment ces bibliothèques fonctionnent. Par conséquent, nous vous invitons à enregistrer tous les VIs que vous allez créer au cours de la lecture de ce guide de démonstration dans des bibliothèques de VIs.

Enregistrez votre VI dans une bibliothèque de VIs.

- Sélectionnez **File»Save As...** Sous UNIX, déplacez-vous dans le système de fichier là où vous avez des privilèges d'écriture. Par exemple, vous pouvez choisir votre répertoire personnel.



2. Attribuez un nom au VI et enregistrez-le dans `mywork.llb`. Vérifiez le nom qui figure dans la commande de type roue codeuse située en haut de la boîte de dialogue. Assurez-vous que le nom correspond bien à `mywork.llb`. Si tel n'est pas le cas, cliquez sur `mywork.llb` dans la liste de répertoires pour être sûr d'avoir bien enregistré votre VI au bon endroit.
 - a. Tapez `My Thermometer.vi` dans la boîte de dialogue.
 - b. Cliquez sur **OK**.
3. Fermez le VI en choisissant **File>Close**.

La création d'un sous-VI



Vous allez apprendre :

- Ce qu'est un sous-VI.
- Comment créer son icône et son connecteur.
- Comment utiliser un VI en tant que sous-VI.

Le concept de hiérarchie

L'un des principes fondamentaux liés à la création d'applications LabVIEW réside dans la compréhension et la mise en œuvre de la nature hiérarchique des VIs. Après avoir créé le VI, vous pouvez l'utiliser comme *sous-VI* dans le diagramme d'un VI de niveau supérieur. Par conséquent, un sous-VI est comparable à un sous-programme en C. De même que le nombre de sous-programmes n'est pas limité pour la programmation en C, il ne l'est pas non plus pour les programmes développés avec LabVIEW. Vous pouvez également appeler un sous-VI dans un autre sous-VI.

Lorsque vous créez une application, vous commencez par le VI principal, puis définissez les entrées et les sorties du système à développer. Vous construisez ensuite les sous-VIs pour mettre en œuvre les opérations sur les données circulant dans le diagramme. Si un diagramme contient un grand nombre d'icônes, regroupez-les dans un VI de niveau inférieur pour simplifier le diagramme. Cette approche modulaire facilite la mise au point, la compréhension et la maintenance des applications.

Création d'un sous-VI

OBJECTIF Générer une icône et un connecteur pour le VI **My Thermometer** que vous avez construit au chapitre 1 et utiliser ce VI comme sous-VI.

Pour utiliser un VI comme sous-VI, vous devez créer une icône pour le représenter dans le diagramme d'un autre VI, ainsi qu'un cadre connecteur auquel vous pourrez connecter les entrées et les sorties.

L'icône



Créez l'icône qui représentera le VI dans le diagramme composé d'autres VIs. Une icône peut être une représentation graphique du VI, ou bien encore une description textuelle du VI ou de ses broches.

1. Si vous avez fermé le VI **My Thermometer**, ouvrez-le de nouveau en sélectionnant **File»Open...** ou en cliquant sur le bouton Open VI dans la boîte de dialogue. Ouvrez `mywork.llb`. Sous Windows, vous pouvez trouver cette bibliothèque dans le répertoire temporaire ou dans `windows\temp`. Sous Macintosh, vous pouvez trouver ce répertoire dans `Temporary Folder` dans le `System Folder`. Sous Unix, `mywork.llb` se trouve dans le répertoire `/tmp`.
2. Choisissez `My Thermometer.vi` dans le répertoire `mywork.llb`.
3. Appelez l'Editeur d'icônes en ouvrant un menu local sur le cadre icône situé en haut à droite de la face-avant et en choisissant **Edit Icon**. En guise de raccourci, vous pouvez également double-cliquer sur le cadre icône pour l'éditer.



Outils et boutons de l'Editeur d'icônes

Les outils situés à gauche de la zone d'édition exécutent les fonctions suivantes :



outil Crayon

Dessine et efface, pixel par pixel.



outil Trait

Trace des lignes droites. Appuyez sur la touche <Ma j> puis faites glisser cet outil pour tracer des lignes horizontales, verticales ou obliques.



outil Pipette

Copie la couleur du premier plan d'un élément de l'icône.



outil Pot de peinture

Remplit une zone délimitée avec la couleur du premier plan.



outil Rectangle

Dessine un rectangle avec la couleur du premier plan. Double-cliquez sur cet outil pour tracer le cadre de l'icône avec la couleur du premier plan.

	outil Rectangle plein	Dessine un rectangle avec la couleur du premier plan et le remplit de la couleur de l'arrière-plan. Double-cliquez pour tracer le cadre de l'icône avec la couleur du premier plan et colorier le fond avec la couleur de l'arrière-plan.
	outil Marquise	Sélectionne une zone de l'icône afin de la déplacer, de la dupliquer ou d'effectuer toute autre modification.
	outil Texte	Saisit du texte à l'intérieur de l'icône.
	outil Plans	Affiche les couleurs actives du premier plan et de l'arrière-plan. Cliquez sur les couleurs pour obtenir la palette des couleurs et choisir de nouvelles couleurs.

Les boutons situés à droite de l'écran d'édition exécutent les fonctions suivantes :

- Undo** Annule la dernière opération effectuée.
- OK** Enregistre votre dessin en tant qu'icône du VI et retourne à la fenêtre de la face-avant.
- Cancel** Retourne à la fenêtre de la face-avant sans tenir compte des modifications.

4. Supprimez l'icône par défaut.
 - a. A l'aide de l'outil Marquise, choisissez la section intérieure de l'icône par défaut, représentée à gauche.
 - b. Appuyez sur la touche <Suppr> pour supprimer l'intérieur de l'icône par défaut.

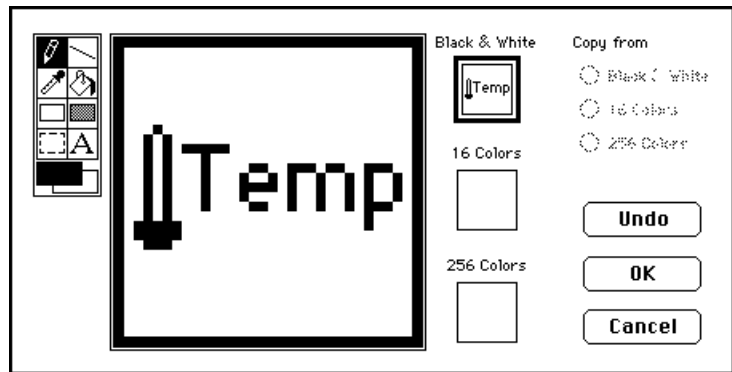


5. Dessinez le thermomètre avec l'outil Crayon.



6. Saisissez le texte avec l'outil Texte. Pour modifier la police, double-cliquez sur l'outil Texte. Exercez-vous à utiliser l'Editeur d'icônes.

Votre icône devrait ressembler à celle de l'illustration suivante.

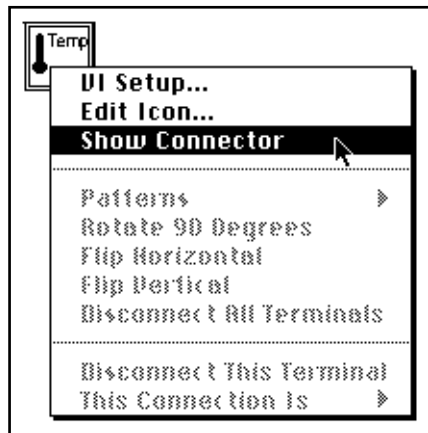


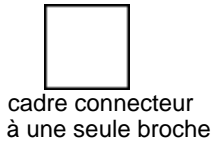
7. Fermez l'Editeur d'icônes en cliquant sur le bouton **OK** une fois l'icône terminée. La nouvelle icône apparaîtra dans le cadre icône en haut à droite de la face-avant.

Le connecteur

Maintenant, vous allez créer le connecteur.

1. Définissez la forme de la broche du connecteur en ouvrant un menu local sur le cadre icône de la face-avant et en choisissant **Show Connector**, conformément à l'illustration suivante.



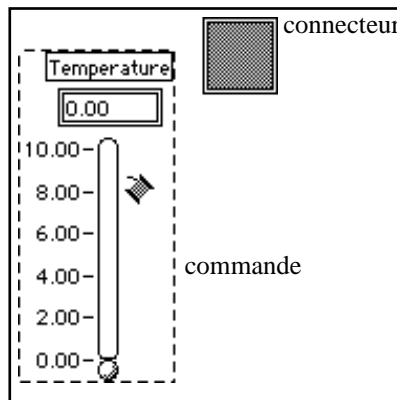


Comme LabVIEW sélectionne automatiquement un type de brochage en fonction du nombre de commandes et d'indicateurs disponibles sur la face-avant, il n'existe qu'une seule broche : l'indicateur thermomètre.



2. Affectez cette broche au thermomètre.

- a. Cliquez sur la broche du connecteur. Le curseur se transforme alors en outil Bobine et la broche s'assombrit.
- b. Cliquez sur le thermomètre. Une ligne en tirets mobile entoure alors l'indicateur, comme dans l'illustration suivante.



Si vous cliquez sur une zone libre de la face-avant, la ligne en tirets disparaît et la broche sélectionnée passe au gris clair pour indiquer que vous avez affecté l'indicateur à cette broche. Une broche qui reste blanche indique une connexion défectueuse. Au besoin, répétez les étapes précédentes.

3. Enregistrez le VI en choisissant **File»Save**. Sur Macintosh, si vous utilisez la boîte de dialogue du fichier source pour enregistrer le VI dans une bibliothèque de VIs, vous devez cliquer sur le bouton **Use LLBs** avant de sélectionner la bibliothèque de VIs.

Ce VI est maintenant terminé et prêt à servir de sous-VI dans d'autres VIs. Son icône va pouvoir le représenter dans les diagrammes de VIs appelants. Son connecteur (mono-broche) donnera en sortie la température mesurée.



Remarque : *le connecteur indique les entrées et les sorties d'un VI lorsque ce dernier est utilisé comme sous-VI. N'oubliez pas que les commandes de la face-avant ne peuvent s'utiliser qu'en entrée, alors que les indicateurs de la face-avant ne peuvent servir qu'en sortie.*

4. Fermez le VI en choisissant **File»Close**.

Mise en œuvre d'un VI en tant que sous-VI

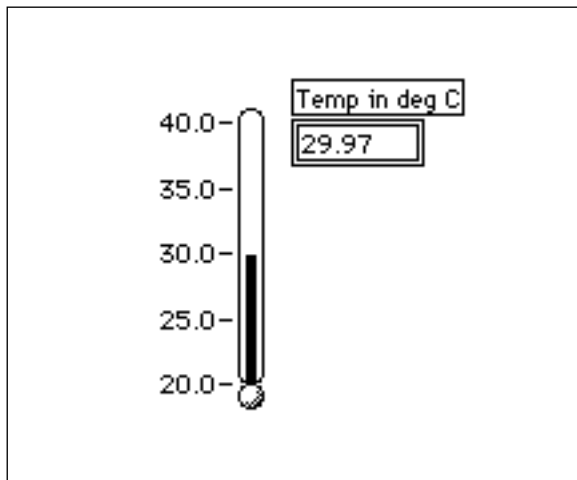
Vous pouvez utiliser n'importe quel VI doté d'une icône et d'un connecteur comme sous-VI dans le diagramme d'un autre VI. Pour choisir les VIs à utiliser comme sous-VIs, il suffit de sélectionner **Functions»Select a VI...** Le fait de choisir cette option ouvre une boîte de dialogue fichier, à partir de laquelle vous pouvez sélectionner n'importe quel VI dans le système. Si vous appelez un VI qui n'a pas d'icône ni de connecteur, un carré blanc s'affiche alors dans le diagramme du VI appelant, que vous ne pouvez pas câbler.

Un sous-VI équivaut à un sous-programme et un nœud de sous-VI (icône/connecteur) à l'appel d'un sous-programme. Le nœud du sous-VI ne constitue pas le sous-VI de même qu'une instruction d'appel de sous-programme dans un programme ne constitue pas le sous-programme lui-même. Un diagramme qui contient plusieurs nœuds identiques appelle plusieurs fois le sous-VI correspondant.

OBJECTIF Construire un VI qui utilise le VI **My Thermometer** comme sous-VI.

Le VI **My Thermometer** que vous avez construit donne une température exprimée en degrés Fahrenheit. Vous utiliserez ce relevé et convertirez la température en degrés Celcius.

La face-avant



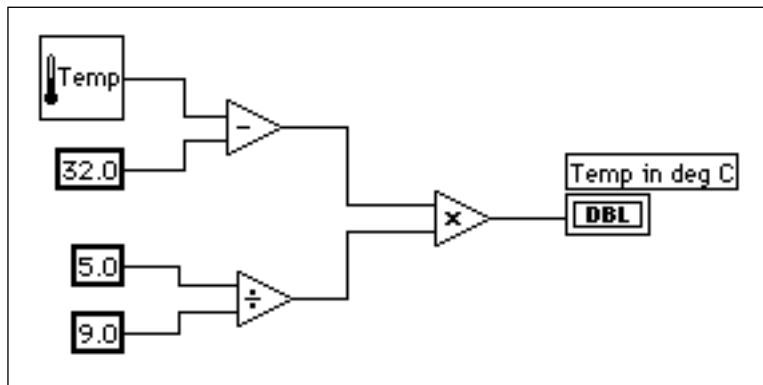
1. Ouvrez une nouvelle face-avant en choisissant **File»New** ou en cliquant sur le bouton New VI dans la boîte de dialogue.
2. Choisissez le thermomètre dans **Controls»Numeric**. Donnez-lui l'étiquette Temp in deg C. Si vous cliquez à l'extérieur du thermomètre avant de saisir l'étiquette, celle-ci disparaîtra. Pour la faire apparaître de nouveau, cliquez sur le thermomètre, choisissez **Show»Label** et saisissez votre étiquette.
3. Modifiez l'échelle du thermomètre pour l'adapter aux valeurs à mesurer. A l'aide de l'outil Doigt, double-cliquez sur la valeur minimale, tapez 20, puis appuyez sur le bouton <Enter> du clavier numérique. Il n'est pas nécessaire de saisir les décimales et les zéros après la virgule. LabVIEW les ajoute automatiquement lorsque vous validez votre saisie. De la même façon, fixez la valeur maximale à 40 et appuyez sur le bouton <Enter> du clavier numérique. LabVIEW ajuste automatiquement les valeurs intermédiaires.

A chaque fois que vous créez une nouvelle commande ou un nouvel indicateur, LabVIEW crée le terminal correspondant dans le diagramme. Les symboles des terminaux rappellent le type de données de la commande ou de l'indicateur. Par exemple, un terminal DBL représente un nombre à virgule flottante en double précision; un terminal TF représente un booléen; un terminal I16 représente un

nombre entier régulier sur 16 \ et un terminal ABC représente une chaîne de caractère.

Le diagramme

1. Choisissez **Windows»Show Diagram**.
2. Ouvrez un menu local dans une zone libre du diagramme et choisissez **Functions»Select a VI...** Une boîte de dialogue s'affiche. Repérez puis ouvrez la bibliothèque `mywork.llb`. (Sous Windows, vous pouvez trouver cette bibliothèque dans le répertoire temporaire ou dans `windows\temp`. Sous Macintosh, ce répertoire se trouve dans `System Folder\Temporary Folder`. Sous Unix, `mywork.llb` se trouve dans le répertoire `/tmp`.) Double-cliquez sur `My Thermometer.vi` ou mettez-le en surbrillance, puis cliquez sur **Open** dans la boîte de dialogue. LabVIEW positionne le VI **My Thermometer** dans le diagramme.
3. Ajoutez les autres éléments du diagramme conformément à l'illustration suivante.



1.23

Numeric constant (Functions»Numeric). Ajoutez trois constantes numériques dans le diagramme en leur affectant les valeurs 32.0, 5.0, et 9.0 avec l'outil Texte.



Remarque : *vous pouvez déterminer le type de constante que le nombre représente par sa couleur. Les constantes numériques bleues sont des nombres entiers et les constantes oranges sont des nombres à double précision. Au besoin, LabVIEW convertit automatiquement les nombres au format adéquat.*



Remarque : *n'oubliez pas que vous pouvez toujours ouvrir un menu local sur les fonctions et choisir la fonction **Create Constant pour créer et câbler automatiquement la bonne constante à une fonction.***



La fonction **Subtract (Functions»Numeric)** permet de retrancher 32 à la valeur de la température exprimée en degrés Fahrenheit pour la convertir en degrés Centigrade.



La fonction **Divide (Functions»Numeric)** calcule la valeur de 5/9 pour permettre la conversion de température.



La fonction **Multiply (Functions»Numeric)** délivre la valeur de la température en degrés Centigrade à partir du processus de conversion.

4. Câblez les éléments du diagramme conformément à l'illustration précédente.



Remarque : *un fil brisé entre l'icône **Thermometer** et le terminal **Temp in deg C** peut indiquer que vous avez mal affecté le connecteur du sous-VI à l'indicateur de sa face-avant. Pour y remédier, veuillez vous reporter aux instructions données dans la section **Création d'un sous-VI au début de ce chapitre**. Lorsque vous aurez modifié le sous-VI, vous aurez peut-être à sélectionner **Relink to SubVI** dans le menu local de l'icône. Si nécessaire, choisissez **Edit»Remove Bad Wires**.*

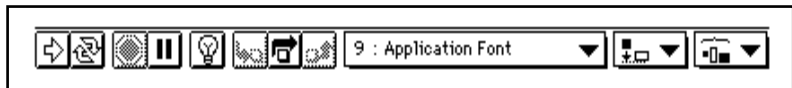


Revenez dans la face-avant puis cliquez sur le bouton **Exécution** dans la barre d'outils.

La barre d'outils du diagramme

Le diagramme contient des options supplémentaires qui ne figurent pas dans la barre d'outils de la face-avant.

Barre d'outils du diagramme :



La barre d'outils du diagramme contient les boutons suivants que vous pouvez utiliser pour mettre au point les VIs.



Ampoule

Affiche les données passant dans les fils de liaison



Exécution détaillée

Exécution pas à pas détaillée (dans les boucles, sous-VIs, etc.)



Exécution semi-détaillée

Exécution pas à pas, chaque boucle, sous-VI, etc., étant considéré comme un seul pas



Sortie

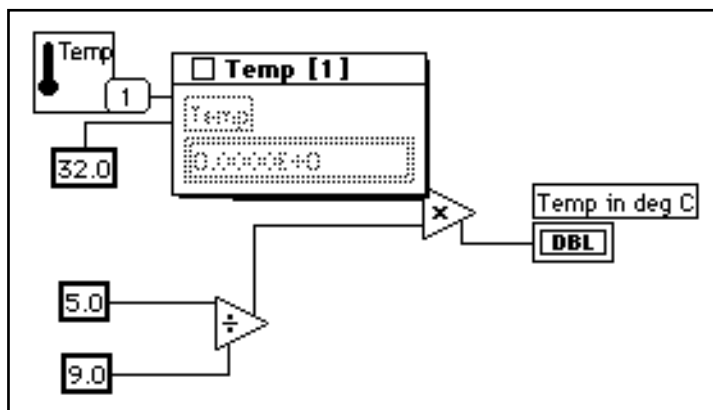
Termine l'exécution en cours de la boucle, du VI, du diagramme, etc.

Quelques techniques de mise au point

1. Le rôle du thermomètre est d'afficher une valeur dans la gamme de mesure choisie. Cependant, si vous souhaitez par exemple, obtenir la valeur en Fahrenheit pour pouvoir faire une comparaison et une mise au point, LabVIEW offre un certain nombre d'outils qui peuvent vous y aider. Dans cet exercice, vous allez étudier la sonde et le mode Animation. Choisissez **Windows»Show Diagram**.



2. Choisissez l'outil Sonde dans la palette **Tools**. A l'aide de l'outil Sonde, cliquez sur la valeur (fil) température du sous-VI **My Thermometer**. Une fenêtre avec une sonde s'affiche alors à l'écran avec le titre Temp 1 et un glyphe jaune avec le numéro de la sonde, conformément à la représentation suivante. La fenêtre Sonde apparaît également dans la face-avant.

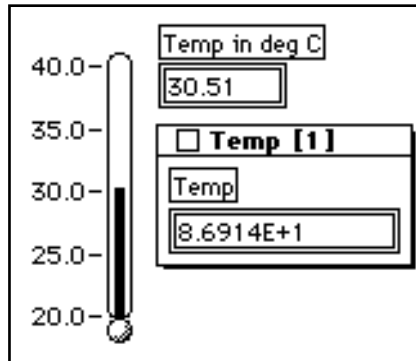


3. Retournez dans la face-avant. Déplacez la fenêtre de la sonde de manière à visualiser les valeurs de la sonde et celles du thermomètre conformément à l'illustration suivante. Lancez le VI.

La température exprimée en degrés Fahrenheit s'affiche dans la fenêtre de la sonde.



Remarque : *les valeurs de température qui apparaissent sur votre écran peuvent être différentes de celles qui apparaissent dans l'illustration suivante. Pour des renseignements supplémentaires, veuillez vous référer à la section intitulée Conversion numérique, dans le chapitre 3, Boucles et graphes déroulant.*



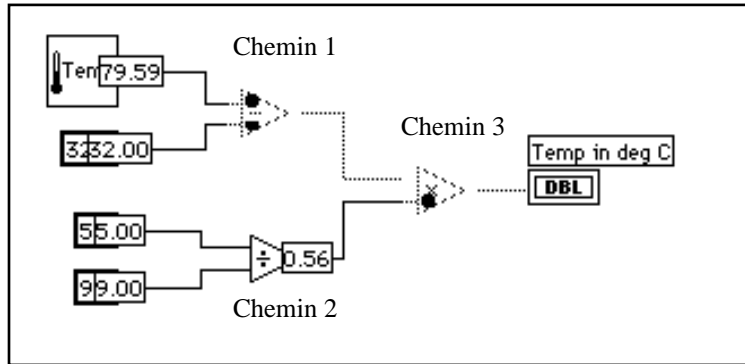
4. Fermez la fenêtre de la sonde en cliquant sur le bouton de fermeture en haut à gauche de sa barre d'outils.

Une autre technique de mise au point très utile consiste à examiner le flux des données en utilisant le mode Animation de LabVIEW.

5. Revenez au diagramme du VI en choisissant **Windows»Show Diagram**.



6. Lancez le mode Animation en cliquant sur le bouton Ampoule de la barre d'outils, représenté en haut à gauche. Le bouton Ampoule se transforme alors en ampoule allumée.
7. Lancez le VI. Vous remarquerez que le diagramme s'anime. Les bulles mobiles représentent le flux des données qui traversent le VI. Vous remarquerez par ailleurs que les valeurs des données transmises apparaissent sur les fils de liaison et affichent en même temps les valeurs contenues dans ces fils, comme si vous aviez sondé le fil de liaison. C'est ce qu'illustre la figure suivante.



Remarquez l'ordre dans lequel s'exécutent les différents nœuds de LabVIEW. Dans un programme écrit en langage textuel classique, les instructions sont exécutées suivant leur ordre d'apparition dans le script du programme. Le logiciel LabVIEW lui, utilise la programmation par *flux des données*. Dans ce type de programmation, un nœud s'exécute uniquement lorsque toutes les données en entrée sont disponibles, et non pas en fonction de la disposition des entrées, de haut en bas ou de gauche à droite.

L'illustration précédente montre comment LabVIEW peut faire du multi-tâche entre les chemins 1 et 2 puisqu'il n'existe aucune dépendance entre les données. C'est-à-dire que rien dans le chemin 1 ne dépend des données qui circulent dans le chemin 2, et réciproquement. Le chemin 3 est le dernier à s'exécuter, car la fonction **Multiply** dépend des données obtenues à partir des fonctions **Subtract** et **Divide**.

Le mode Animation est un outil très utile pour étudier la nature des flux des données. Vous pouvez également utiliser les boutons d'exécution en mode pas à pas si vous souhaitez mieux contrôler le processus de mise au point.



8. Lancez l'exécution en mode pas à pas en cliquant sur le bouton Exécution semi-détaillée de la barre d'outils. Le seul fait de cliquer sur ce bouton affiche la première séquence qui sera exécutée dans le VI. Une fois que LabVIEW a terminé cette partie de la séquence, il met en évidence la prochaine tâche qui s'exécutera dans le VI.



9. Passez à la fonction **Divide** en cliquant sur le bouton Exécution semi-détaillée de la barre d'outils. Le seul fait de cliquer sur ce

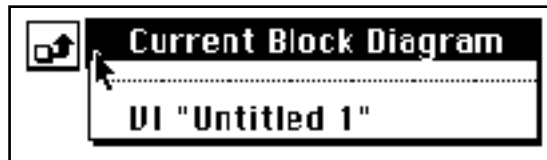
bouton lance la fonction **Divide**. Une fois que LabVIEW a achevé cette tâche, il met en évidence la prochaine à s'exécuter dans le VI.



10. Allez dans le sous-VI **My Thermometer** en cliquant sur le bouton Exécution détaillée de la barre d'outils. Le seul fait de cliquer sur ce bouton ouvre la face-avant et le diagramme de votre sous-VI thermomètre. Vous avez maintenant le choix entre passer en mode Exécution pas à pas ou bien lancer le sous-VI.



11. Terminez l'exécution du diagramme en cliquant sur le bouton Sortie de la barre d'outils. Le seul fait de cliquer sur ce bouton termine toutes les séquences restant à mettre en œuvre dans le diagramme. Une fois que LabVIEW a terminé cette partie de la séquence, il met en évidence la prochaine tâche devant s'exécuter dans le VI. Vous pouvez également maintenir enfoncé le bouton de la souris lorsque vous cliquez sur le bouton Sortie pour accéder à un menu local. Dans ce menu local, vous pouvez limiter l'exécution du VI. L'illustration suivante affiche les options de fin d'exécution dans le menu local du bouton Sortie.



12. Sélectionnez **File»Save as** et enregistrez le VI dans le répertoire `mywork.llb`. Nommez-le `Using My Thermometer.vi`, puis fermez-le.

Ouverture, exécution et modification des sous-VIs

Vous pouvez ouvrir un VI comme sous-VI à partir du diagramme du VI appelant. Pour ce faire, double-cliquez sur l'icône du sous-VI ou choisissez **Project»This VI's SubVIs**. Ouvrez ensuite le diagramme en sélectionnant **Windows»Show Diagram**.

Sachez que toute modification effectuée sur un sous-VI n'affecte que la version en cours d'utilisation (en mémoire vive) jusqu'à ce que vous enregistrez le sous-VI. Vous remarquerez cependant que les modifications sont prises en compte pour tous les appels du sous-VI et qu'elles ne se limitent pas au nœud sur lequel vous étiez pour ouvrir le VI.

La fenêtre Hiérarchie

La fenêtre Hiérarchie (**Project»Show VI Hierarchy**) sert à visualiser les liens de dépendance des VIs en donnant des informations sur les VIs appelants et les sous-VIs. Cette fenêtre contient une barre d'outils qui va vous servir à configurer plusieurs types de paramètres pour les objets affichés. La figure ci-après présente la barre d'outils Hiérarchie du VI.



Vous pouvez utiliser les boutons de la barre d'outils de la fenêtre Hiérarchie ou le menu **VIEW**, ou bien encore ouvrir un menu local sur un espace vide de la fenêtre pour accéder aux options suivantes :



- **Dessin** Répartit les nœuds après des manipulations successives sur les nœuds de hiérarchies, pour réduire les croisements de lignes et privilégier l'aspect symétrique. S'il existe un nœud principal, vous pouvez alors parcourir la fenêtre de manière à ce que la première racine affichant des sous-VIs soit visible.



- **Présentation de gauche à droite** Présente les nœuds de haut en bas, les racines en haut.



- **Présentation de haut en bas** Présente les nœuds de gauche à droite, les racines sur la gauche.



- **Inclusion/Exclusion des VIs dans les bibliothèques de VIs** Permute l'affichage du graphe déroulant pour masquer/afficher les VIs des bibliothèques de VIs.



- **Inclusion/Exclusion des variables globales** Permute l'affichage du graphe déroulant pour masquer/afficher les variables globales.



- **Inclusion/Exclusion des types** Permute l'affichage du graphe déroulant pour masquer/afficher les définitions de type.

De plus, le menu **View** et les menus locaux proposent les options **Show all VIs** et **Full VI Path in Label** auxquelles vous n'avez pas accès depuis la barre d'outils.



Au fur et à mesure que vous déplacez l'outil Doigt sur les objets dans la fenêtre Hiérarchie, LabVIEW affiche le nom du VI sous son icône.

Utilisez la touche de commutation <Tab> située entre les outils Flèche et Fenêtre de défilement. Cette manipulation vous permettra de déplacer plus facilement les nœuds de la fenêtre Hiérarchie vers le diagramme.

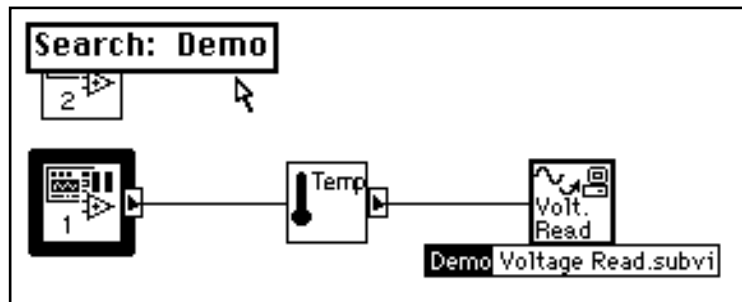
Vous pouvez faire glisser le nœud d'un VI ou sous-VI dans un diagramme ou le copier dans le presse-papiers en cliquant sur le nœud. Tout en appuyant sur la touche <Maj>, cliquez sur le nœud d'un VI ou d'un sous-VI pour sélectionner plusieurs choix à recopier dans d'autres diagrammes ou faces-avant. Le fait de double-cliquer sur un nœud de VI ou de sous-VI suffit à ouvrir sa face-avant.

A tous les VIs ayant des sous-VIs correspond un bouton en forme de flèche qui permet de les afficher ou de les masquer. Le fait de cliquer sur la flèche rouge ou de double-cliquer sur le VI lui-même suffit à ouvrir les sous-VIs du VI. Une flèche noire sur un nœud d'un VI indique que tous les sous-VIs existants sont affichés. Vous pouvez également ouvrir un menu local sur le nœud d'un VI ou d'un sous-VI pour accéder à un menu déroulant avec des options comme celles permettant d'afficher ou de masquer les sous-VIs, d'ouvrir la face-avant d'un VI ou d'un sous-VI, d'éditer l'icône d'un VI, etc.

Recherche dans la hiérarchie

Vous pouvez aussi rechercher les nœuds visibles dans la fenêtre Hiérarchie par nom. Vous démarrez la recherche en tapant le nom du nœud n'importe où dans la fenêtre. Lorsque vous saisissez le texte, une fenêtre de recherche apparaît qui affiche le texte au fur et à mesure que

vous le tapez et qui effectue parallèlement une recherche au sein de la hiérarchie. La figure suivante en est une illustration.

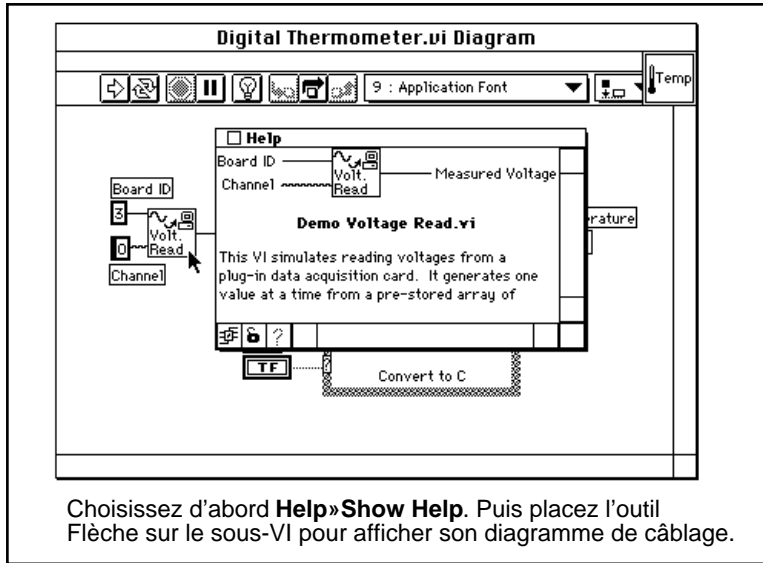


Lorsque vous avez trouvé le nœud qui convient, appuyez sur le bouton <Enter> pour rechercher le nœud suivant correspondant aux critères de recherche, ou appuyez sur les touches <Maj-Enter> (Windows) ; <Maj-return> (Macintosh) ; <Maj-Return> (Sun) ; ou <Maj-Enter> (HP-UX) pour trouver le nœud précédent correspondant aux critères de recherche.

L'aide en ligne pour les nœuds des sous-VIs

Lorsque vous placez l'un des outils sur le nœud du sous-VI, la fenêtre d'aide affiche l'icône du sous-VI avec les fils rattachés à chaque broche. L'illustration suivante vous présente un exemple d'aide en ligne. Il s'agit du VI **Digital Thermometer** de la palette **Functions»Tutorial**. Votre VI **Thermometer** contient également le

texte que vous avez saisi dans la boîte de dialogue relative aux informations du VI.



Aide simple/détaillée

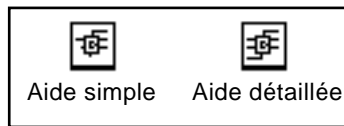
Dans la fenêtre d'aide, vous avez la possibilité d'opter pour une représentation simple ou, à l'inverse, détaillée des objets du diagramme.



Remarque : *lorsque vous ouvrez la fenêtre d'aide, LabVIEW affiche par défaut la représentation simple.*

Dans le cas de la représentation simple, LabVIEW n'affiche que les entrées obligatoires et recommandées pour les VIs et les fonctions. Dans la représentation détaillée, LabVIEW affiche aussi les entrées dites optionnelles, plus le nom complet du chemin du VI. Pour accéder à une représentation simple, appuyez sur le bouton Aide simple/détaillée ou choisissez **Help»Simple Diagram Help**. La figure

suivante est l'illustration des deux interrupteurs à utiliser pour obtenir une représentation simple ou détaillée.

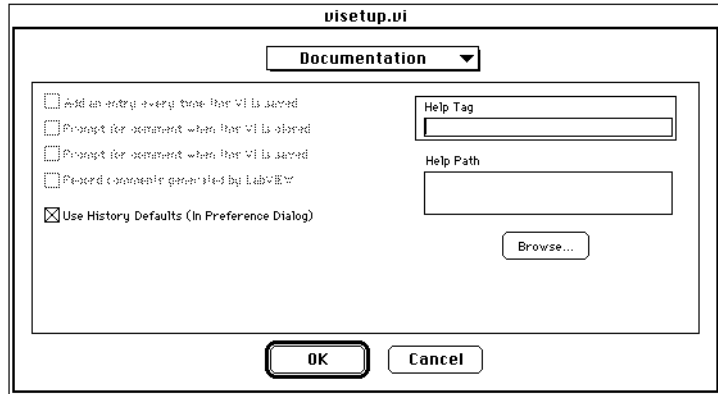


Dans la fenêtre d'aide, les entrées obligatoires figurent en gras, les entrées recommandées en texte normal et les données optionnelles en gris. Lorsque vous développez vos propres VIs, vous avez la possibilité de déterminer quelles entrées seront obligatoires, recommandées ou optionnelles en ouvrant un menu local sur une entrée ou sortie du cadre connecteur et en choisissant l'option qui convient dans le sous-menu **This Connection is**.



Liens vers les fichiers d'aide en ligne

Dans la fenêtre d'aide, vous pouvez cliquer sur le bouton d'aide en ligne pour accéder à l'aide en ligne de LabVIEW ainsi qu'aux fichiers d'aide que vous avez créés avec le compilateur d'aide. Si vous souhaitez créer votre propre fichier d'aide, vous devez en indiquer le lien en cliquant sur le cadre icône et en choisissant **VI Setup...** Lorsque la boîte de dialogue **VI Setup** s'ouvre, choisissez **Documentation** dans la commande de type roue codeuse située en haut de la boîte, puis entrez le chemin correspondant au fichier d'aide dans la zone **Help Path**. L'exemple suivant présente les options disponibles dans la boîte de dialogue **VI Setup**.



Vous choisissez **Browse...** pour associer le fichier d'aide et l'objet à votre VI.

Boucles et graphes déroulants



Vous allez apprendre :

- Comment utiliser une boucle *While*.
- Comment afficher des données dans un graphe déroulant.
- Ce qu'est un registre à décalage et comment l'utiliser.
- Comment utiliser une boucle *For*.

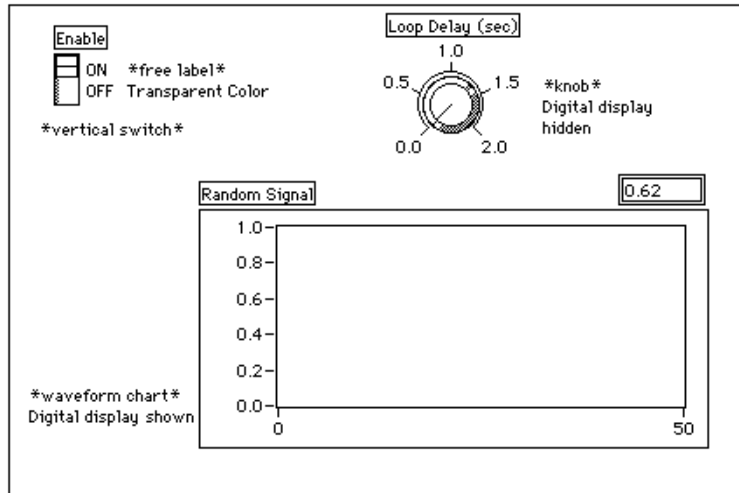
Les structures contrôlent le flux des données dans les VIs. LabVIEW s'articule autour de quatre structures différentes : la boucle *While*, la boucle *For*, la structure Condition et la structure Séquence. Ce chapitre aborde les boucles *While* et *For* ainsi que les graphes déroulants et les registres à décalage. Les structures Condition et Séquence sont détaillées au chapitre 5, *Structures Condition, structures Séquence et boîte de calcul*.

Mise en œuvre d'une boucle *While* et d'un graphe déroulant

OBJECTIF Utiliser une boucle *While* et un graphe déroulant pour acquérir et présenter les données en temps réel.

Vous allez construire un VI qui génère des données aléatoires et les restitue ensuite dans un graphe déroulant. Sur sa face-avant, le VI disposera d'un bouton rotatif pour régler le temps de scrutation de la boucle entre 0 et 2 secondes et d'un interrupteur pour arrêter le VI. Vous apprendrez à modifier le comportement mécanique de l'interrupteur pour ne plus avoir à l'actionner à chaque mise en route du VI. Inspirez-vous de la face-avant de l'illustration suivante pour construire votre VI.

La face-avant



1. Ouvrez une nouvelle face-avant en sélectionnant **File»New** ou en cliquant sur le bouton New VI dans la boîte de dialogue.
2. Placez un interrupteur vertical (**Controls»Boolean**) dans la face-avant. Nommez-le **Enable**. Il vous servira à interrompre le processus d'acquisition.
3. Utilisez l'outil Texte pour créer les textes libres ON et OFF. Vous pouvez créer ces étiquettes en cliquant sur l'outil Texte puis sur votre face-avant et en saisissant le texte de l'étiquette. Utilisez l'outil Pinceau pour effacer les bords des textes libres. La lettre T en bas à gauche de la palette de couleurs permet de rendre un objet transparent.
4. Placez un graphe déroulant (**Controls»Graph**) dans la face-avant. Nommez-le **Random Signal**. Ce graphe servira à afficher les données aléatoires en temps réel.



Remarque : assurez-vous de sélectionner un graphe déroulant et non pas un graphe déroulant multicourbes. Dans la palette Graph, le graphe déroulant apparaît le plus près du côté gauche de la palette.



5. Ouvrez un menu local sur le graphe, puis choisissez **Show»Digital Display**. Cet afficheur numérique contient la dernière valeur.
6. A l'aide de l'outil Texte, double-cliquez sur la valeur 10.0 dans le graphe déroulant et tapez 1.0, puis cliquez en dehors de l'étiquette.

Le seul fait de cliquer valide la valeur saisie. Vous pouvez également appuyer sur la touche <Enter> (Windows), <return> (Macintosh), <Return> (Sun) ou <Enter> (HP-UX) pour valider le changement d'échelle.

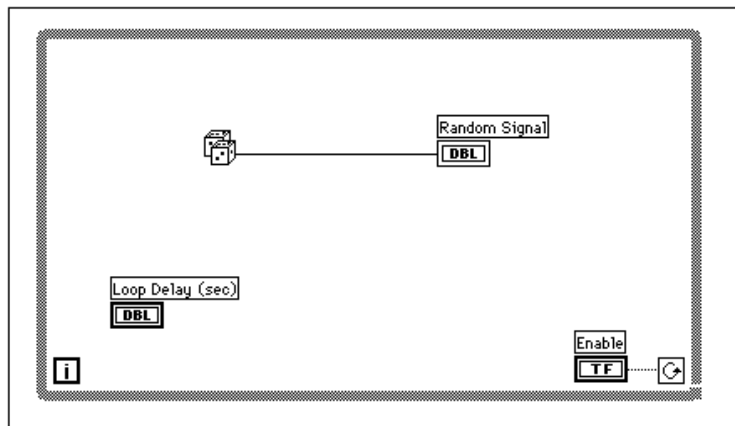


- Placez un bouton rotatif (**Controls»Numeric**) dans la face-avant. Nommez-le `Loop delay (sec)`. Ce bouton servira à contrôler le temps de cycle de la boucle *While* comme nous le verrons dans la suite de cet exercice. Ouvrez un menu local sur le bouton rotatif et désactivez **Show»Digital Display** pour faire disparaître l'afficheur numérique affiché par défaut.



- A l'aide de l'outil Texte, double-cliquez sur la valeur 10.0 dans l'échelle qui entoure le bouton rotatif, tapez 2.0, puis cliquez en dehors de l'étiquette pour valider la nouvelle valeur.


Le diagramme




- Ouvrez la fenêtre du diagramme.
- Placez-y la boucle *While*, en la choisissant dans la palette **Functions»Structures**. Cette boucle *While* est une boîte réajustable qui n'est pas immédiatement déposée dans le diagramme. Au contraire, vous avez ici la possibilité de la positionner et de la redimensionner à votre guise. Pour ce faire, cliquez dans une zone en haut à gauche de tous les terminaux. Continuez, tout en maintenant le bouton de la souris enfoncé, et dessinez un rectangle de manière à entourer tous ces terminaux.

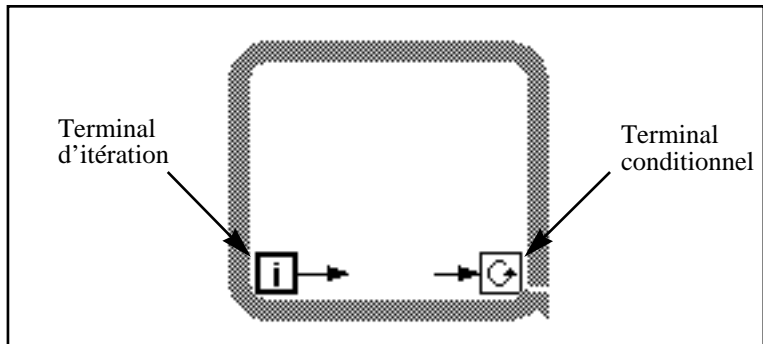


Une boucle *While* est alors créée à l'emplacement et au format voulus.

Terminal conditionnel 

Terminal d'itération 

- La boucle *While*, reproduite dans l'exemple suivant, est une boîte réajustable que vous utilisez pour exécuter le diagramme qu'elle contient, jusqu'à ce que la valeur booléenne transmise au *terminal conditionnel* (un terminal d'entrée) prenne la valeur FALSE. Le VI vérifie la valeur fournie à la fin de chaque itération de la boucle *While*, si bien que la boucle *While* s'exécute toujours au moins une fois. Le *terminal d'itération* est une sortie numérique qui contient le nombre de fois que la boucle s'est exécutée. Cependant, le comptage des itérations démarre toujours à zéro, moyennant quoi, lorsque la boucle ne s'exécute qu'une seule fois, le terminal d'itération de sortie vaut 0.



La boucle *While* équivaut au pseudo-code suivant :

Faire

Exécuter le diagramme à l'intérieur de la boucle (ce qui définit la condition)

Tant que la condition While est VRAIE



- Choisissez la fonction **Random Number (0-1)** dans la palette **Functions»Numeric**.
- Câblez le diagramme conformément à l'illustration de la section *Diagramme*, en connectant la fonction **Random Number (0-1)** au terminal du graphe déroulant **Random Signal**, et l'interrupteur Enable au terminal conditionnel de la boucle *While*. Ne câblez pas pour l'instant le terminal Loop Delay.
- Retournez dans la face-avant et positionnez l'interrupteur vertical sur MARCHE en cliquant dessus avec l'outil Doigt. Lancez le VI.

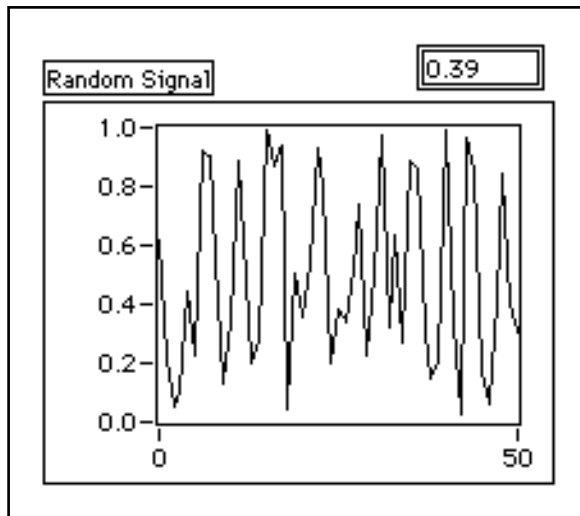


La boucle *While* est une structure de bouclage indéfinie. Le diagramme qu'elle contient s'exécute tant que la condition indiquée reste vraie. Dans cet exemple, tant que l'interrupteur est sur la position MARCHE (TRUE), le diagramme continue à générer des nombres aléatoires et à les afficher dans le graphe déroulant.

7. Pour arrêter la boucle, cliquez sur l'interrupteur vertical. Il passe alors en position ARRÊT et transmet la valeur FALSE au terminal conditionnel de la boucle qui s'arrête.
8. Le graphe déroulant dispose d'une mémoire tampon d'affichage dans laquelle sont conservés un certain nombre de points, une fois sortis de la zone d'affichage de l'écran. Affectez au graphe déroulant une barre de défilement en ouvrant un menu local sur celui-ci et en choisissant **Show»Scrollbar**. Vous pouvez utiliser l'outil Flèche pour ajuster la taille et la position de la barre de défilement.

Pour vous déplacer dans le graphe déroulant, cliquez et maintenez enfoncé le bouton de la souris sur l'une des flèches de la barre de défilement.

Pour effacer le contenu de la mémoire tampon d'affichage et réinitialiser le graphe déroulant, ouvrez un menu local sur le graphe et choisissez **Data Operations»Clear Chart**.





Remarque : *par défaut, la taille de la mémoire tampon d'affichage est de 1024 points. Vous pouvez augmenter ou diminuer la taille de la mémoire en ouvrant un menu local sur le graphe déroulant et en choisissant l'option **Chart History Length...***

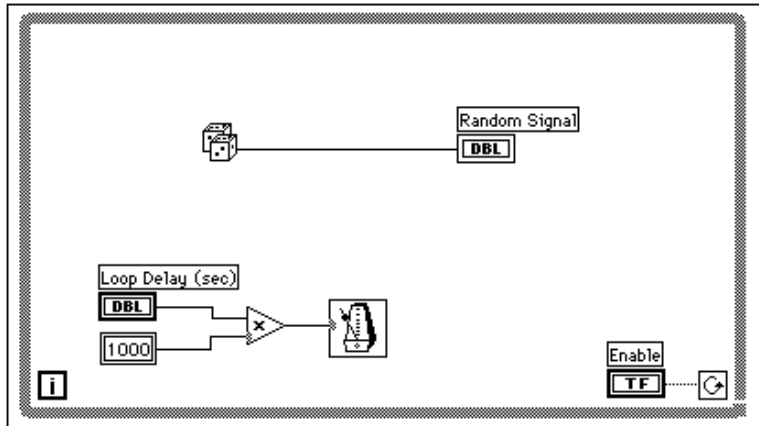
Temps de cycle

Lorsque vous demandez l'exécution du VI, la boucle *While* se déroule le plus rapidement possible. Cependant, vous avez peut-être besoin d'acquérir des données à intervalles réguliers, comme toutes les secondes ou toutes les minutes.

Le cas échéant, vous disposez de fonctions de temps de cycle de LabVIEW qui fournissent le temps en millisecondes (ms), ce qui ne signifie pas pour autant que votre système doit maintenir ce temps de cycle. Vous trouverez dans la liste suivante les instructions à suivre pour déterminer la précision des fonctions des temps de cycle de LabVIEW sur votre système.

- **(Windows 3.1)** L'horloge a une résolution par défaut de 55 ms. Vous pouvez configurer LabVIEW pour avoir une résolution de 1 ms en choisissant **Edit»Preferences...**, en choisissant **Performance** et **Disk** du menu **Chemins**, et en désactivant la case **Use Default Timer**. LabVIEW ne prend pas la résolution de 1 ms par défaut car cela implique une charge plus importante à votre système d'exploitation.
- **(Windows 95/NT)** L'horloge a une résolution de 1 ms. Néanmoins, cela dépend du matériel utilisé. Par conséquent, dans des systèmes plus lents, comme un 80386, il y a de fortes chances que le temps de cycle de la résolution soit plus faible.
- **(Macintosh)** Pour les systèmes 68K qui ne disposent pas de l'extension QuickTime, l'horloge a une résolution de 16.66 ms (1/60ème de seconde). Si vous disposez d'un Power Macintosh ou si vous avez installé QuickTime dans votre système, la résolution de l'horloge est de 1 ms.
- **(UNIX)** L'horloge a une résolution de 1 ms.

Vous pouvez contrôler le temps de cycle de la boucle en utilisant la fonction **Wait Until Next ms Multiple (Functions»Time & Dialog)**. Cette fonction garantit qu'aucune itération ne durera moins longtemps que le nombre de millisecondes spécifié.



1. Modifiez le VI pour que la génération des nombres aléatoires se fasse à intervalles de temps réguliers. On en règle la durée au moyen du bouton rotatif, comme l'illustre le schéma précédent.



Fonction **Wait Until Next ms Multiple (Functions»Time & Dialog)**. Dans cet exercice, cette fonction multiplie le terminal du bouton rotatif par 1000 pour convertir la valeur, exprimée en secondes, de ce bouton, en millisecondes. Reprenez cette valeur comme entrée à la fonction **Wait Until Next ms Multiple**.



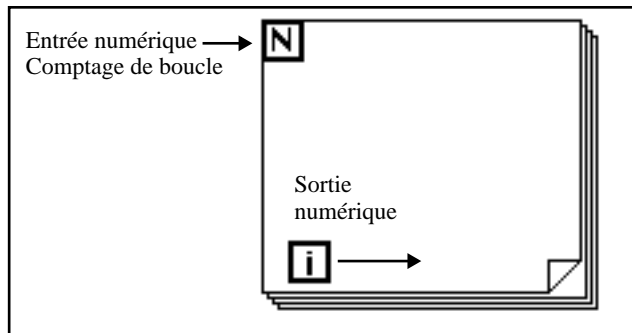
Fonction **Multiply (Functions»Numeric)**. Dans cet exercice, la fonction **Multiply** permet de multiplier la valeur du bouton rotatif par 1000 pour convertir les secondes en millisecondes.



Fonction **Numeric Constant (Functions»Numeric)**. La constante numérique contient la constante utilisée pour multiplier la valeur du bouton rotatif en vue d'obtenir une quantité exprimée en millisecondes. Ainsi donc, si le bouton affiche une valeur égale à 1.0, la boucle s'exécute toutes les 1000 millisecondes (c'est-à-dire une fois par seconde).

2. Exécutez le VI. Tournez le bouton afin d'obtenir des valeurs différentes pour le nombre de secondes.
3. Enregistrez et fermez le VI dans la bibliothèque `mywork.llb`. Nommez-le `My Random Signal.vi`.

La boucle *For*



Placez la boucle *For* dans le diagramme en la choisissant dans **Functions»Structures**. Une boucle *For* (voir l'illustration précédente) est une boîte réajustable, comme une boucle *While*. Comme la boucle *While*, elle n'est pas immédiatement déposée sur le diagramme. A la place, apparaît une petite icône qui représente la boucle *For* dans le diagramme. Vous pouvez la redimensionner et la déplacer à votre guise. Pour se faire, cliquez dans une zone à gauche de tous les terminaux. Tout en maintenant le bouton de la souris enfoncé, dessinez un rectangle qui englobe tous les terminaux que vous souhaitez faire apparaître à l'intérieur de la boucle *For*. Lorsque vous relâchez le bouton de la souris, LabVIEW crée une boucle *For* dans la taille et à l'emplacement voulus.

La boucle *For* exécute le diagramme dans les limites de sa bordure un certain nombre de fois préétabli. La boucle *For* compte deux terminaux :



le *terminal de comptage* (terminal d'entrée). Le terminal de comptage précise le nombre de fois que la boucle doit s'exécuter.



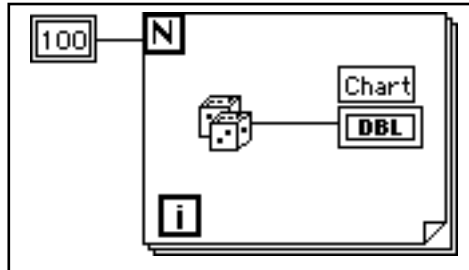
le *terminal d'itération* (terminal de sortie). Le terminal d'itération contient le nombre de fois que la boucle s'est déjà exécutée.

La boucle *For* équivaut au pseudo-code suivant :

Pour $i = 0$ jusqu' à $N-1$

Exécuter le diagramme à l'intérieur de la boucle

L'exemple suivant présente une boucle *For* qui génère 100 nombres aléatoires et affiche les points sur un graphe déroulant.



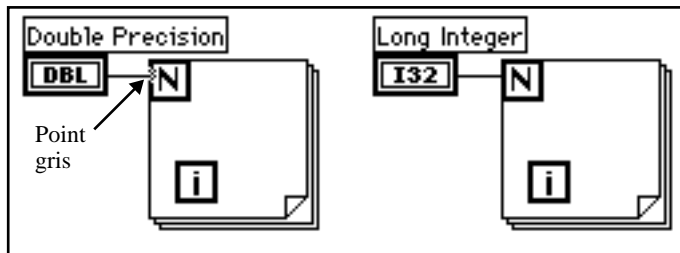
Conversion numérique

Jusqu'à présent, tous les indicateurs et commandes numériques que vous avez utilisés étaient des nombres à virgule flottante en double précision représentés sur 32 bits. Cependant, LabVIEW peut aussi traiter des nombres entiers (octets, mots ou entiers longs) ou des nombres à virgule flottante (simple précision, double précision ou précision étendue). Par défaut, les nombres sont traités comme des nombres à virgule flottante en double précision.

Si vous connectez deux terminaux de types différents, LabVIEW convertit l'un des terminaux dans le type de l'autre. A titre de rappel, LabVIEW place un point gris appelé point de coercition, sur le terminal objet de la conversion.



Par exemple, prenez le cas du terminal de comptage de la boucle *For*. La variable qui y figure est un entier long. Si vous connectez un nombre à virgule flottante en double précision à ce terminal, LabVIEW convertit ce nombre en entier long. Vous remarquerez qu'il y a un point gris sur le terminal de comptage de la boucle *For*.





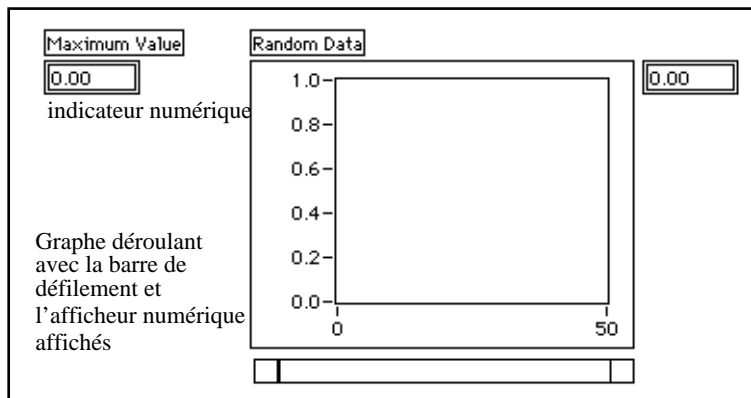
Remarque : lorsque le VI convertit un nombre à virgule flottante en un nombre entier, il l'arrondit à l'entier le plus proche. Si le nombre se situe entre deux entiers, il est arrondi à l'entier pair le plus proche. Par exemple, le VI arrondit 6.5 à 6, mais arrondit 7.5 à 8. Cette méthode correspond à la norme IEEE utilisée pour la lecture des nombres. Pour en savoir plus, veuillez vous reporter à la norme IEEE 754.

Mise en œuvre d'une boucle For

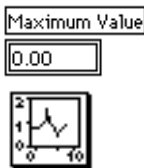
OBJECTIF

Utiliser une boucle *For* et des registres à décalage pour calculer la valeur maximale dans une série de nombres aléatoires. Vous allez mettre en œuvre une boucle *For* ($N = 100$) à la place de la boucle *While*.

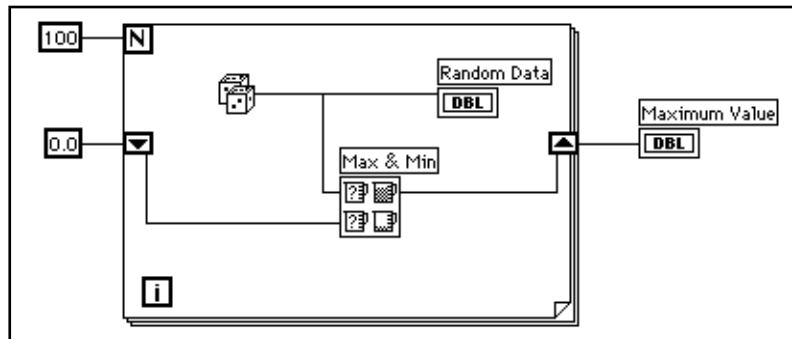
La face-avant



1. Ouvrez une nouvelle face-avant et ajoutez-y les objets présentés dans l'illustration précédente.
 - a. Placez un indicateur numérique dans la face-avant et nommez-le Maximum Value.
 - b. Placez un graphe déroulant dans la face-avant et nommez-le Random Data. Modifiez l'échelle du graphe déroulant afin d'avoir des valeurs allant de 0.0 à 1.0.
 - c. Ouvrez un menu local dans le graphe déroulant et choisissez **Show»Scrollbar** puis **Show»Digital Display**.



Le diagramme



1. Ouvrez la fenêtre du diagramme.
2. Ajoutez la boucle *For* (**Functions»Structures**).
3. Ajoutez le registre à décalage en plaçant le curseur sur la bordure droite ou gauche de la boucle *For* et en choisissant **Add Shift Register**. Consultez la section suivante pour plus d'informations sur les registres à décalage.
4. Ajoutez les autres éléments dans le diagramme.



Utilisez la fonction **Random Number (0-1)** (**Functions»Numeric**) pour générer les nombres aléatoires.



Numeric Constant (Functions»Numeric). La boucle *For* doit savoir combien d'itérations elle doit exécuter. Dans le cas présent, la boucle *For* s'exécutera 100 fois de suite.



Numeric Constant (Functions»Numeric). Sachant que la sortie du générateur de nombres aléatoires est comprise entre 0,0 et 1,0, vous allez régler le registre à décalage sur zéro.

Vous devez avoir un minimum de connaissances sur les données à acquérir pour initialiser un registre à décalage. Par exemple, il ne serait pas judicieux d'initialiser le registre à décalage sur 1,0, cette valeur étant déjà supérieure à toutes celles que vous allez acquérir. Si vous n'initialisez pas le registre à décalage, alors il devra contenir la valeur maximale de l'exécution précédente du VI. Par conséquent, vous pourriez très bien obtenir une valeur maximale en sortie qui n'ait aucun rapport avec les valeurs de l'acquisition en cours de collecte.



La fonction **Max & Min (Functions»Comparison)** prend deux valeurs numériques et restitue la plus grande en haut à droite et la plus petite en bas à droite. Comme dans cet exercice, seule la valeur la plus grande nous intéresse : ne connectez que la valeur la plus grande et ignorez la plus petite.

5. Câblez les terminaux conformément à la représentation suivante. Si le terminal Maximum Value se trouvait à l'intérieur de la boucle *For*, vous verriez qu'il serait continuellement mis à jour, mais comme il se trouve à l'extérieur, il ne contient que la dernière valeur maximale connue.



Remarque : *la mise à jour des indicateurs à chaque itération de la boucle prend beaucoup de temps. C'est pourquoi il est préférable de l'éviter à chaque fois que possible pour améliorer la vitesse d'exécution.*

6. Lancez le VI.
7. Enregistrez le VI. Nommez-le My Calculate Max.vi.

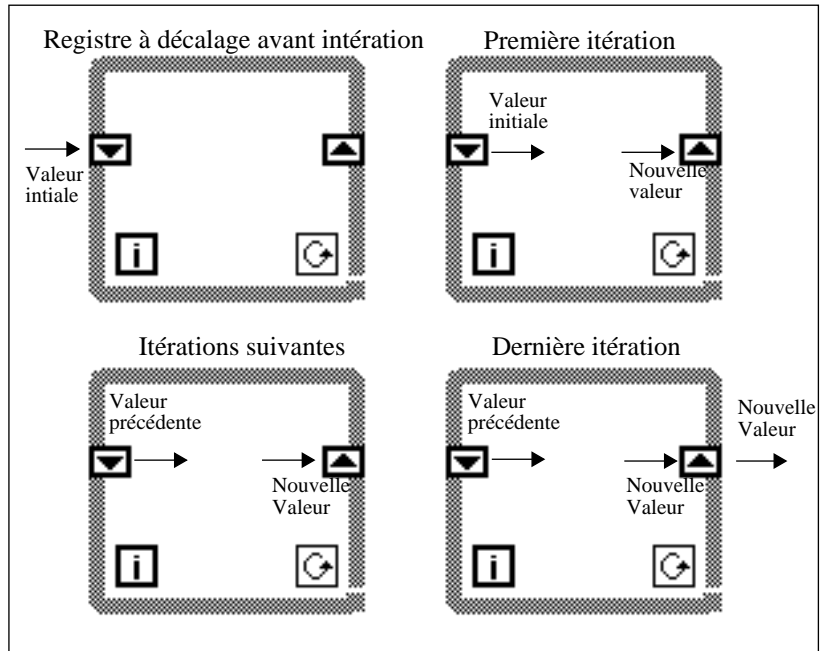
Les registres à décalage

Les registres à décalage (disponibles pour les boucles *While* et *For*) permettent de transférer les valeurs d'une itération de la boucle à une autre. Pour créer un registre à décalage, ouvrez un menu local sur la bordure droite ou gauche d'une boucle puis choisissez **Add Shift Register**.

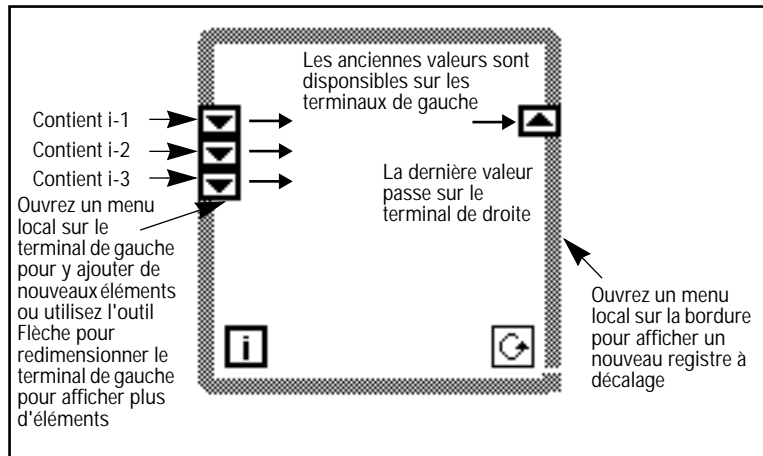


Le registre à décalage contient deux terminaux situés de part et d'autre des bords verticaux du cadre de la boucle. Le terminal *de droite* stocke les données pendant toute l'itération. A la fin de l'itération, elles se décalent et apparaissent dans le terminal *de gauche* au début de la

prochaine itération (voir l'illustration suivante). Un registre à décalage peut contenir n'importe quel type de données : numérique, booléen, chaîne de caractères, tableau, etc. Le registre à décalage s'adapte automatiquement au type de données du premier objet connecté au registre à décalage.



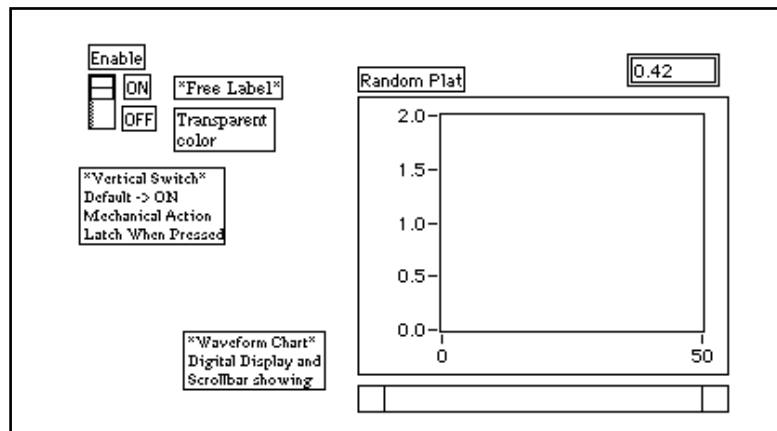
Vous pouvez configurer les registres à décalage de manière à ce qu'ils conservent les données d'itérations précédentes. Cette caractéristique est très utile pour établir la moyenne des points acquis lors d'itérations successives. Pour ce faire, il suffit de créer des terminaux supplémentaires en ouvrant un menu local sur le terminal de *gauche* ou de *droite* et en choisissant l'option **Add Element**. Par exemple, si un registre à décalage contient trois éléments dans son terminal de gauche, vous pouvez accéder à ces valeurs à partir des trois dernières itérations.



Mise en œuvre des registres à décalage

OBJECTIF Construire un VI qui affiche deux tracés aléatoires dans un graphe déroulant. Les deux tracés seront ceux d'un nombre aléatoire et de la moyenne des quatre derniers points du tracé aléatoire.

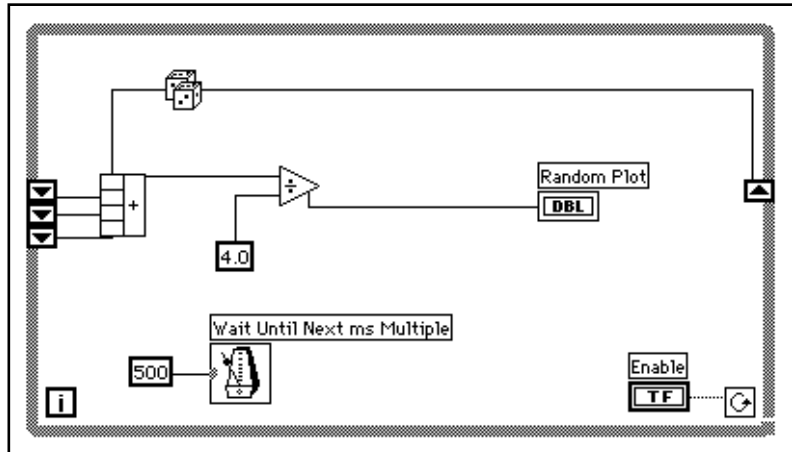
La face-avant



1. Ouvrez une nouvelle fenêtre pour y créer la face-avant reproduite dans l'illustration précédente.

2. Après avoir ajouté le graphe déroulant dans la face-avant, modifiez l'échelle pour que les valeurs soient comprises entre 0.0 et 2.0.
3. Après avoir ajouté l'interrupteur vertical, définissez l'état ON comme étant l'état par défaut puis configurez l'action mécanique en **Latch When Pressed**.

Le diagramme



1. Ajoutez **While Loop (Functions»Structures)** dans le diagramme et créez le registre à décalage.
 - a. Ouvrez un menu local sur le bord droit ou gauche de la boucle *While* puis choisissez **Add Shift Register**.
 - b. Ajoutez un autre élément en ouvrant un menu local sur le terminal de *gauche* du registre à décalage et en choisissant **Add Element**. Ajoutez un troisième élément en procédant de la même manière.
2. Construisez le diagramme de l'illustration précédente.



La fonction **Random Number (0-1) (Functions»Numeric)** génère des données brutes.

Fonction **Compound Arithmetic (Functions»Numeric)**. Dans cet exercice, la fonction **Compound Arithmetic** renvoie la somme des nombres aléatoires de deux itérations. Pour ajouter plus de données en entrée, ouvrez un menu local et choisissez **Add Input** dans celui-ci.



Fonction **Divide (Functions»Numeric)**. Dans cet exercice, la fonction **Divide** calcule la moyenne des quatre derniers nombres aléatoires.



Numeric Constant (Functions»Numeric). Lors de chaque itération de la boucle *While*, la fonction **Random Number (0-1)** génère une valeur aléatoire. Le VI ajoute cette valeur aux trois dernières valeurs conservées par les terminaux de gauche du registre à décalage. La fonction **Random Number (0-1)** divise le résultat par quatre pour calculer la moyenne des valeurs (la valeur actuelle et les trois précédentes). La moyenne est ensuite affichée dans le graphe déroulant.



Fonction **Wait Until Next ms Multiple (Functions»Time & Dialog)**. Cette fonction permet de garantir que la durée de chaque itération de la boucle ne sera pas inférieure au nombre de millisecondes spécifié en entrée, soit 500 millisecondes dans le cas qui nous intéresse. Si vous ouvrez un menu local sur l'icône et choisissez **Show»Label**, l'étiquette **Wait Until Next ms Multiple** apparaît à l'écran.

3. Ouvrez un menu local sur l'entrée de la fonction **Wait Until Next ms Multiple** puis choisissez **Create Constant**. Une constante numérique apparaît alors à l'écran qui est automatiquement connectée à la fonction.
4. Entrez 500 dans l'étiquette. La constante numérique connectée à la fonction **Wait Until Next ms Multiple** indique un temps d'attente de 500 millisecondes (une demi-seconde), ce qui veut dire que la boucle s'exécutera toutes les demi-secondes.



Vous remarquerez que le VI initialise les registres à décalage avec un nombre aléatoire. Si vous n'initialisez pas le terminal de registre à décalage, il prend la valeur par défaut ou la dernière valeur connue de la précédente exécution. Le cas échéant, les premières moyennes ne sont pas significatives.

5. Exécutez le VI et observez ce qui se passe. LabVIEW ne trace que la moyenne sur le graphe.

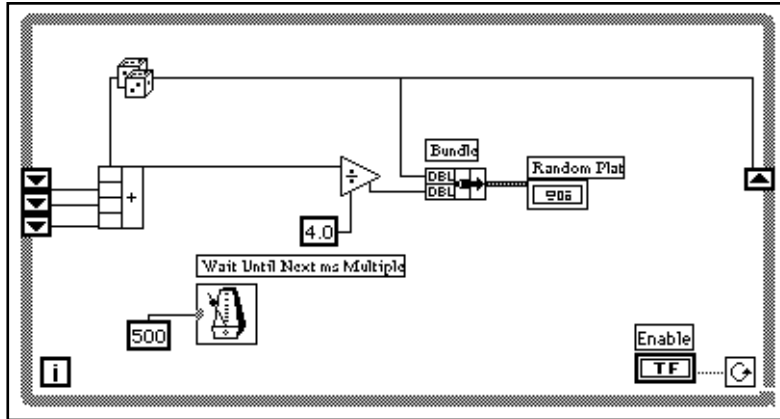


Remarque : *n'oubliez pas d'initialiser les registres à décalage afin d'éviter d'y intégrer d'anciennes valeurs ou bien encore des valeurs par défaut dans les mesures des données en cours.*

Les graphes déroulants multicourbes

Les graphes déroulants peuvent contenir plusieurs tracés. Lorsqu'il y a plusieurs entrées scalaires, il faut les assembler.

Vous allez modifier le diagramme afin d'afficher la moyenne et la valeur aléatoire courante sur le même graphe déroulant.



1. Modifiez le diagramme conformément à l'illustration précédente.

Fonction **Bundle** (**Functions**»**Cluster**). Dans cet exercice, la fonction **Bundle** intègre, ou regroupe, la valeur moyenne et la valeur courante à tracer sur le graphe déroulant. Conformément à l'illustration, le nœud d'assemblage apparaît à gauche lorsque vous le placez dans le diagramme. Si vous ouvrez un menu local sur ce nœud et choisissez **Show**»**Label**, le mot **Bundle** s'affiche dans l'étiquette. Vous pouvez agrandir ce nœud en lui ajoutant des éléments supplémentaires. Pour ce faire, utilisez le curseur de redimensionnement (que vous obtenez en plaçant l'outil Flèche dans le coin de la fonction).

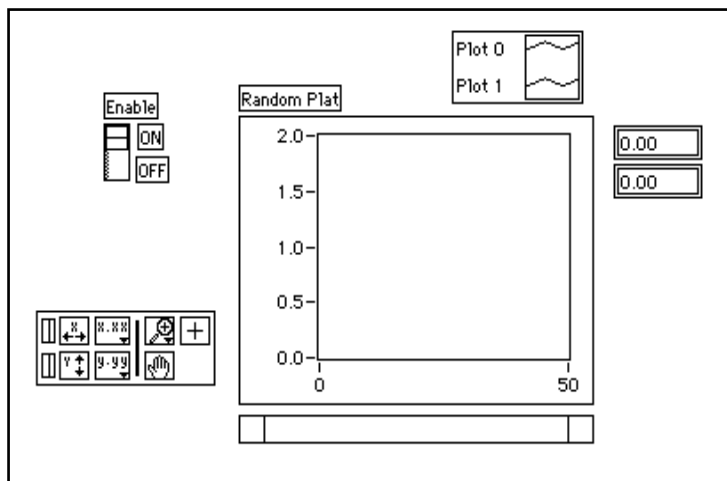


Remarque : *l'ordre des données en entrée de la fonction Bundle détermine l'ordre dans lequel sera effectué le tracé des points sur le graphe déroulant. Par exemple, si vous connectez la variable brute à la donnée principale du nœud d'assemblage et la moyenne à la dernière donnée, le premier tracé correspondra aux données brutes et le second à la moyenne.*

2. Exécutez le VI. Le VI affiche deux tracés sur le même graphe déroulant. Les tracés sont superposés, c'est-à-dire qu'ils partagent la même échelle verticale. Essayez d'exécuter le VI en mode Animation pour voir comment se comportent les données dans les registres à décalage. Lorsque vous aurez terminé, n'oubliez pas d'arrêter cette fonction pour que le VI puisse s'exécuter à pleine vitesse.

Personnalisation des graphes déroulants

Vous pouvez personnaliser les graphes déroulants pour répondre à des besoins d'affichage particuliers et recevoir des informations complémentaires comme : des barres de défilement, des légendes, des palettes et des afficheurs numériques.



Dans le graphe déroulant, l'afficheur numérique a été activé. Vous remarquerez que chaque tracé du graphe a son propre afficheur numérique.

1. Si la barre de défilement est visible, masquez-la en cliquant dessus pour ouvrir un menu local et désactivez l'option **Show»ScrollBar**.
2. Personnalisez l'axe des Y.
 - a. Utilisez l'outil Texte pour double-cliquer sur 2.0 dans l'échelle des Y. Tapez 1 . 2 puis appuyez sur <Enter> (Windows); <return> (Macintosh); <Return> (Sun); ou <Enter> (HP-UX).



- b. Toujours avec l'outil Texte, cliquez sur l'avant-dernier nombre de l'axe des Y. Remplacez ce nombre par 0.2, ou 0.5 par exemple. Ce nombre détermine le pas de graduation de l'axe des Y.



Remarque : *la taille du graphe a une incidence directe sur la graduation des axes. Agrandissez la taille du graphe déroulant si vous avez des difficultés à personnaliser les axes.*

3. Affichez la légende en ouvrant un menu local sur le graphe déroulant et en choisissant **Show»Legend**. Au besoin, déplacez la légende.



Vous pouvez placer la légende n'importe où dans le graphe déroulant. Agrandissez-la pour y inclure deux tracés en utilisant le curseur de redimensionnement. L'outil Flèche se transforme en curseur de redimensionnement pour indiquer que vous pouvez redimensionner la légende. Remplacez 0 par `Current Value` en double-cliquant sur le texte avec l'outil Texte et en tapant le nouveau texte. Vous pouvez pareillement remplacer le tracé 1 par `Running Avg`. Si le texte disparaît, élargissez le rectangle réservé au texte de légende en le redimensionnant à partir de l'angle *gauche* de la légende avec le curseur de redimensionnement. Vous pouvez aussi définir le style des tracés, avec des lignes et des points, en cliquant sur la légende pour ouvrir un menu local.

Vous pouvez définir l'épaisseur du tracé en ouvrant un menu local dans la légende. En utilisant ce menu local, vous pouvez modifier les paramètres par défaut de la ligne en affectant une valeur supérieure à 1 pixel. Vous avez aussi la possibilité de sélectionner une épaisseur égale à celle d'un cheveu, invisible à l'écran mais qui est en revanche imprimée si votre imprimante supporte l'impression d'une ligne d'une telle finesse.



Si vous avez un moniteur couleur, vous pouvez également colorier le fond du graphe et des tracés en cliquant avec l'outil Pinceau sur l'élément que vous souhaitez modifier dans la légende pour faire apparaître un menu local. Choisissez la couleur qui vous convient parmi les couleurs présentées.

4. Affichez la palette locale du graphe déroulant en ouvrant son menu local, et choisissez l'option **Show»Palette**.

Avec la palette, vous pouvez modifier la présentation du graphe déroulant pendant que le VI s'exécute. Vous pouvez réinitialiser le graphe déroulant, modifier l'échelle de l'axe des X ou des Y et redimensionner l'affichage à tout moment. Vous pouvez également parcourir le contenu d'autres zones ou faire des zooms sur des zones du graphe ou du graphe déroulant. Comme pour la légende, vous pouvez placer la palette n'importe où dans le graphe déroulant.

5. Lancez le VI. Pendant l'exécution du VI, utilisez les boutons de la palette pour modifier le graphe déroulant.



Vous pouvez utiliser les boutons X et Y pour rééchelonner respectivement les axes des X et des Y. Si vous voulez que le graphe déroulant assure une mise à l'échelle automatique et permanente de l'un des axes, cliquez sur l'interrupteur de verrouillage situé à gauche de chacun des boutons pour verrouiller la mise à l'échelle automatique.



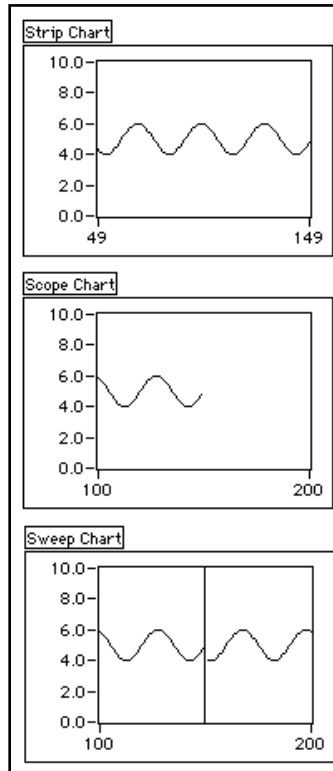
Vous pouvez utiliser les autres boutons pour modifier le contenu du texte sur les axes ou pour contrôler le mode de fonctionnement du graphe déroulant. Exercez-vous à utiliser ces boutons afin de bien assimiler leur fonctionnement, parcourir la zone affichée ou faire des gros plans des zones du graphe déroulant.



Remarque : *la modification du format du texte des axes nécessite bien souvent plus d'espace par rapport à l'espace initialement prévu pour les axes. Si vous les modifiez, le texte risque de dépasser la taille maximale autorisée. Pour y remédier, utilisez le curseur de redimensionnement pour réduire la zone du graphe déroulant réservée à l'affichage.*

Les différents modes d'affichage

La figure suivante présente les trois modes d'affichage disponibles à partir de la palette **Data Operations**»**Update Mode** : le graphe déroulant, l'oscillographe et le graphe à balayage. Le mode d'affichage par défaut est le graphe déroulant. (Si le VI est toujours en cours d'exécution, le sous-menu **Data Operations** correspond au menu local du graphe.)



L'affichage en mode *graphe déroulant* s'apparente à un enregistreur papier. En effet, lorsque le VI reçoit une nouvelle valeur, il trace à droite le point correspondant à cette valeur et déplace les anciennes valeurs sur la gauche.

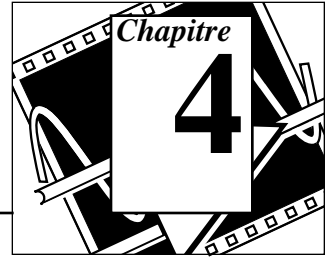
1. Assurez-vous que le VI est toujours en cours d'exécution, ouvrez un menu local sur le graphe déroulant puis choisissez **Data Operations»Update Mode»Scope Chart**.

En mode *oscillographe*, le tracé se rafraîchit comme sur un oscilloscope. Au fur et à mesure que le VI reçoit une nouvelle valeur, il trace le point correspondant à cette valeur à droite du dernier point. Lorsque le tracé atteint la bordure droite de la zone d'affichage de la courbe, le VI efface le tracé et recommence son affichage à partir de la bordure gauche. L'oscillographe est beaucoup plus rapide que le graphe déroulant puisqu'il n'est pas nécessaire de redessiner toute la courbe à l'arrivée de chaque nouveau point (ce qui se traduit par un glissement de l'ensemble de la courbe vers la gauche).

2. Assurez-vous que le VI est toujours en cours d'exécution, ouvrez un menu local sur le graphe déroulant, puis choisissez **Data Operations»Update Mode»Sweep Chart**.

Le mode *graphe à balayage* se comporte sensiblement de la même façon que le mode oscillographe à la différence que l'écran du graphe ne s'efface pas lorsque le point arrive sur la bordure droite. Au contraire, une ligne verticale mobile de séparation se déplace dans la zone d'affichage au fur et à mesure que le VI ajoute de nouvelles données.

3. Arrêtez le VI, et enregistrez-le sous le nom `My Random Average.vi`.



Tableaux, clusters et graphes

Vous allez apprendre :

- Ce que sont les tableaux.
- Comment générer des tableaux sur le pourtour des boucles.
- Ce qu'est le polymorphisme.
- Ce que sont les *clusters*.
- Comment utiliser les graphes déroulants pour afficher les données.
- Comment utiliser quelques-unes des fonctions de base dans les tableaux.

Les tableaux

Un tableau se définit comme un ensemble de données du même type. Il peut être à une ou plusieurs dimensions pouvant comporter jusqu'à $2^{31} - 1$ éléments à condition bien sûr que la capacité mémoire le permette. Dans LabVIEW, les tableaux peuvent comporter tout type d'élément (à l'exception des tableaux, graphes déroulants ou graphes). Vous accédez à chaque élément du tableau grâce à son indice. L'indice est un chiffre compris entre 0 et $n-1$, où n correspond au nombre d'éléments contenus dans le tableau. Le tableau de valeurs numériques reproduit ci-après est un exemple de tableau à une dimension. Vous remarquerez que l'indice du premier élément est égal à 0, celui du deuxième à 1, et ainsi de suite.

indice	0	1	2	3	4	5	6	7	8	9
tableau de 10 éléments	1.2	3.2	8.2	8.0	4.8	5.1	6.0	1.0	2.5	1.7

Les commandes, constantes et indicateurs de tableaux

Vous créez des commandes, constantes et indicateurs de tableau sur la face-avant ou dans le diagramme en associant une *constante de tableau* à une valeur numérique, booléenne, une chaîne de caractères ou un

cluster. Seule restriction : un élément du tableau ne peut pas être un autre tableau, un graphe ou un graphe déroulant.

Pour des exemples de tableaux, veuillez vous reporter au répertoire `exemples\general\arrays.llb`.

Les graphes

Un *indicateur de graphe* est un affichage à deux dimensions d'un ou de plusieurs tableaux de données que l'on appelle *tracés*. LabVIEW propose trois types de graphes différents : les *graphes XY*, les *graphes oscilloscopiques* et les *graphes d'intensité* (veuillez vous reporter à la section *Quelques informations supplémentaires* à la fin de ce chapitre pour en savoir plus sur les graphes d'intensité).

La différence entre un graphe et un graphe déroulant (point abordé dans le chapitre 3, *Boucles et graphes déroulants*, de ce guide de démonstration) réside dans le fait que le premier trace des données en bloc tandis que le second trace les données point par point ou tableau par tableau.

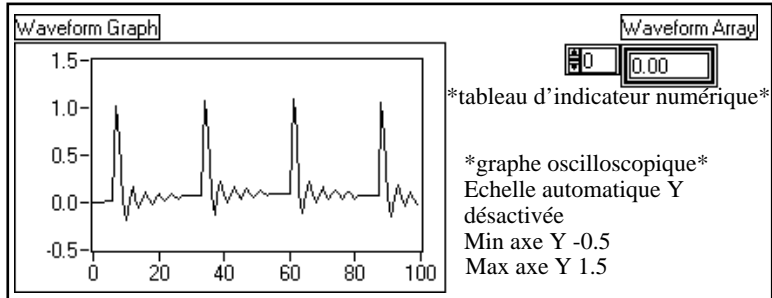
Création d'un tableau par auto-indexation

OBJECTIF

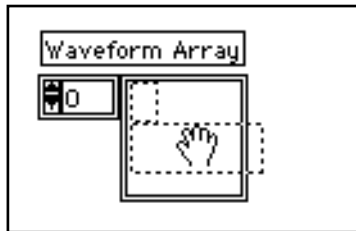
Créer un tableau en utilisant la fonction d'auto-indexation d'une boucle *For* et tracer le tableau sous la forme d'un graphe oscilloscopique.

Vous allez construire un VI qui va d'abord générer un tableau en utilisant le VI **Generate Waveform** puis tracer le tableau sous la forme d'un graphe oscilloscopique. Vous allez également modifier le VI pour obtenir plusieurs tracés.

La face-avant



1. Ouvrez une nouvelle face-avant.
2. Sélectionnez un tableau vierge dans **Controls»Array & Cluster** et placez-le dans la face-avant. Donnez-lui l'étiquette **Waveform Array**.
3. Sélectionnez un indicateur numérique dans **Controls»Numeric** et positionnez-le dans la zone d'affichage des éléments du tableau vierge, conformément à l'illustration suivante. Cet indicateur affiche le contenu du tableau.



Conformément à ce que nous avons vu précédemment, un *indicateur de graphe* est un affichage à deux dimensions d'un ou de plusieurs tableaux de données appelés *tracés*. LabVIEW offre trois types de graphes : les *graphes XY*, les *graphes oscilloscopiques* et les *graphes d'intensité*.

4. Sélectionnez un graphe oscilloscopique dans **Controls»Graph** et placez-le dans la face-avant. Donnez-lui l'étiquette **Waveform Graph**.

Ce graphe oscilloscopique trace des tableaux avec des points à intervalle régulier, comme dans le cas des courbes variant avec le temps.



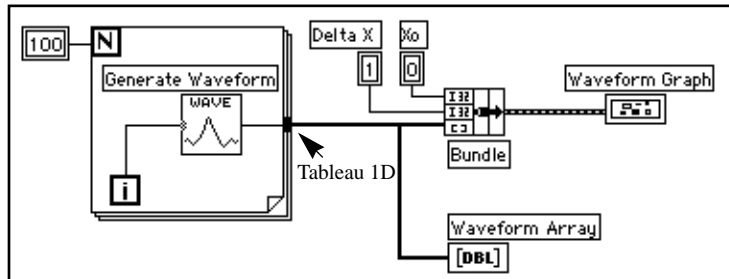
5. Agrandissez ce graphe en faisant glisser l'un de ses coins à l'aide du curseur de redimensionnement.

Par défaut, les graphes *mettent automatiquement à l'échelle* leurs entrées. C'est-à-dire qu'ils ajustent automatiquement les limites des axes X et Y pour visualiser l'intégralité des données.



6. Désactivez la fonction de mise à l'échelle automatique. Pour ce faire, ouvrez un menu local sur le graphe et désactivez **Y Scale»Autoscale Y**.
7. Modifiez les limites de l'axe Y en double-cliquant dessus à l'aide de l'outil Texte et en entrant les nouvelles valeurs. Tapez pour la valeur la plus basse de l'axe des Y -0.5 et pour la valeur la plus haute 1.5.

Le diagramme



1. Construisez le diagramme reproduit ci-dessus.

Le VI **Generate Waveform (Functions»Tutorial)** restitue un point d'une courbe. Ce VI nécessite une entrée d'indice scalaire ; câblez le terminal d'itération de la boucle à cette entrée. Le fait d'ouvrir un menu local sur le VI et de sélectionner **Show»Label** affiche le mot `Generate Waveform` dans l'étiquette.

Vous remarquerez que le fil du VI **Generate Waveform** s'épaissit en se transformant en tableau à la bordure de la boucle.

La boucle *For* empile automatiquement les tableaux sur son pourtour. Ce phénomène s'appelle l'*auto-indexation*. Dans ce cas, la constante

numérique 100 câblée à l'entrée numérique de comptage de la boucle permet à la boucle *For* de créer un tableau de 100 éléments (indexés de 0 à 99).



La fonction **Bundle (Fonctions»Cluster)** rassemble les composantes du tracé en un *cluster*. Vous devez redimensionner l'icône de la fonction **Bundle** pour pouvoir la câbler correctement. Pour ce faire, positionnez l'outil Flèche en bas à droite de l'icône. L'outil se transforme alors en curseur de redimensionnement. A ce moment-là, cliquez dessus et faites-le glisser jusqu'à ce qu'une troisième broche apparaisse. Vous pouvez alors continuer à câbler le diagramme en vous inspirant de l'illustration précédente.

Un *cluster* peut contenir des éléments de types différents. Par exemple, dans le diagramme que vous êtes en train de construire, le *cluster* regroupe des éléments provenant de plusieurs endroits différents du diagramme pour limiter l'enchevêtrement du câblage. Lorsque vous utilisez des *clusters*, il vous faut moins de broches pour câbler vos sous-VIs. Un *cluster* s'apparente à un enregistrement en Pascal ou à une structure en C. Vous pouvez le comparer à un bouquet de fils, un peu comme ceux des centraux téléphoniques, où chaque fil correspond à un élément différent du *cluster*. Les composantes de ce *cluster* comprennent la valeur initiale de X (0), la valeur de delta-X (1) et le tableau des Y (données de la courbe, fournies par les constantes numériques du diagramme). Dans LabVIEW, utilisez la fonction **Bundle** pour former un *cluster*.

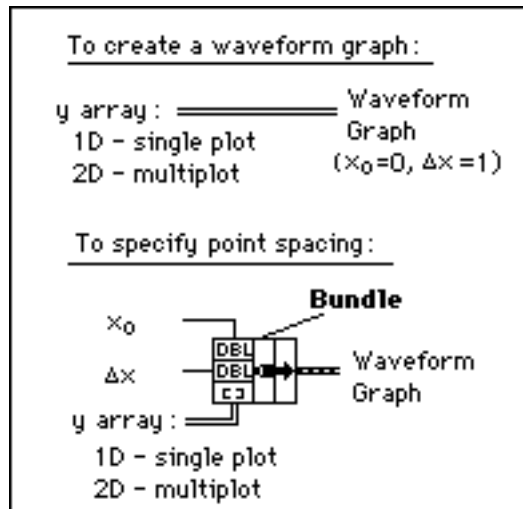


Remarque : *assurez-vous de ne construire que des types de données reconnus par les graphes et les graphes déroulants.*

Au fur et à mesure que vous construisez votre diagramme, vérifiez bien les types de données en procédant de la manière suivante :



- Ouvrez la fenêtre d'aide en choisissant **Help»Show Help**.
- Déplacez l'outil Bobine au-dessus du terminal du graphe.
- Vérifiez l'exactitude des informations qui apparaissent dans la fenêtre d'aide. A titre d'exemple, reportez-vous à l'illustration suivante.



Numeric Constant (Functions»Numeric). Trois constantes numériques permettent de fixer le nombre d'itérations de la boucle *For*, la valeur initiale de X et la valeur de delta-X. Vous remarquerez que vous avez toujours la possibilité d'ouvrir un menu local sur le terminal de comptage de la boucle *For*, présenté à gauche, puis de sélectionner **Create Constant** pour ajouter et câbler automatiquement une constante numérique à ce terminal.

Chaque itération de la boucle *For* génère un point de la courbe que le VI stocke dans le tableau de courbe créé automatiquement sur la bordure de la boucle. Une fois que la boucle a terminé de s'exécuter, la fonction **Bundle** rassemble la valeur initiale de X (X_0), la valeur de delta-X et le tableau pour les tracer sur le graphe.

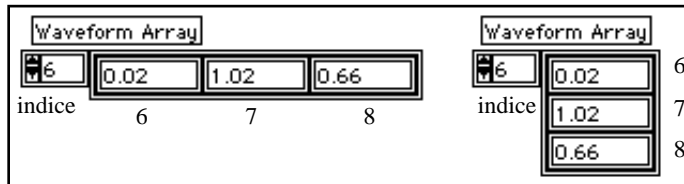
2. Revenez dans la face-avant et exécutez le VI. Il trace le tableau auto-indexé de la courbe dans le graphe déroulant. La valeur initiale de X est égale à 0 et la valeur de delta-X est égale à 1.
3. Modifiez la valeur de delta-X en 0.5 et la valeur initiale de X en 20. Lancez de nouveau le VI.

Vous remarquerez que le graphe affiche maintenant les mêmes 100 points de données avec une valeur de départ de 20 et un delta-X de 0.5 entre chaque point (voir l'axe des X). Dans un test de synchronisation, le graphe correspondrait à un enregistrement de 50 secondes à 20

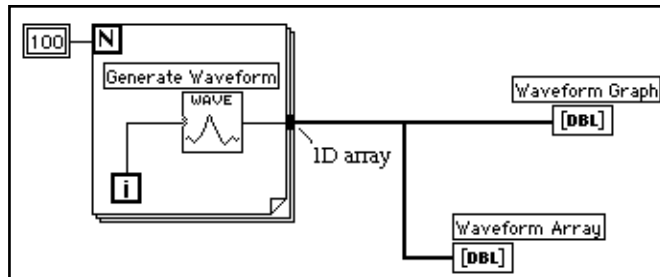
secondes du début. Exercez-vous en modifiant la valeur initiale et le delta-X.

- Vous avez la possibilité de visualiser n'importe quel élément du tableau en entrant l'indice de cet élément dans l'affichage d'indice. Si vous saisissez un nombre supérieur à la taille du tableau, l'afficheur s'assombrit, indiquant par là qu'aucune valeur n'a été définie pour cet indice.

Si vous souhaitez visualiser plusieurs éléments à la fois, vous pouvez redimensionner l'indicateur du tableau. Positionnez l'outil Flèche en bas à droite du tableau. L'outil se transforme alors en curseur de redimensionnement. Lorsque l'outil se transforme, faites-le glisser vers la droite ou vers le bas. Le tableau affiche alors plusieurs éléments dans l'ordre croissant d'indexation en commençant par l'élément correspondant à l'indice spécifié, conformément à l'illustration suivante.



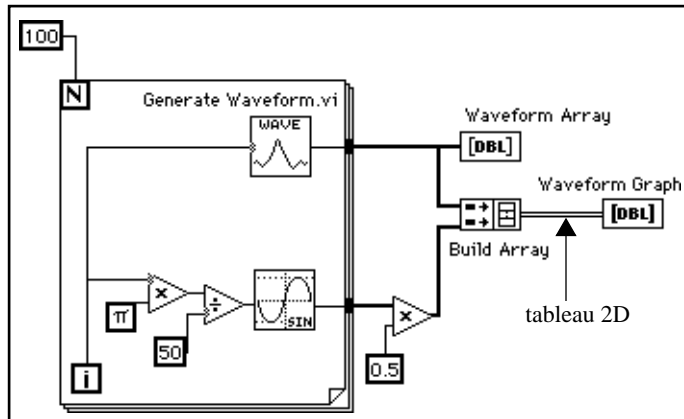
Dans le diagramme précédent, vous avez indiqué une valeur initiale de X et une valeur de delta-X pour la courbe. Sachez qu'il est fréquent que la valeur initiale de X soit égale à zéro et que la valeur de delta-X soit égale à 1. Dans ces cas-là, vous pouvez câbler le tableau de la courbe directement au terminal du graphe oscilloscopique comme le montre l'illustration ci-après.



5. Revenez dans la fenêtre du diagramme. Effacez la fonction **Bundle** ainsi que les constantes numériques qui lui sont affectées. Pour ce faire, choisissez la fonction et les constantes à l'aide de l'outil Flèche, puis appuyez sur la touche <Suppr>. Choisissez **Edit»Remove Bad Wires**. Finissez de câbler le diagramme comme indiqué dans l'illustration précédente.
6. Lancez le VI. Vous remarquerez qu'il trace la courbe avec une valeur initiale de X égale à 0 et une valeur de delta-X égale à 1.

Les graphes multicourbes

Vous pouvez créer des graphes multicourbes en construisant un tableau avec des données du même type que celles d'un graphe à un tracé unique.



1. Poursuivez la construction de votre diagramme jusqu'à ce qu'il ressemble au diagramme de la figure précédente.



Fonction **Sine (Functions»Numeric»Trigonometric)**. Dans cet exercice, cette fonction, utilisée dans une boucle *For*, va vous permettre de construire un tableau de points représentant une période de sinusoïde.



Fonction **Build Array (Functions»Array)**. Dans cet exercice, cette fonction va vous permettre de créer la structure de données qui convient le mieux pour tracer deux tableaux sur un graphe oscilloscopique, à savoir, un tableau à deux dimensions. Agrandissez

la fonction **Build Array** pour créer deux entrées en tirant sur un coin à l'aide de l'outil Flèche.



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

Petit rappel : les fonctions **Multiply** et **Divide** se trouvent dans la palette **Functions»Numeric**.

2. Revenez dans la face-avant puis lancez le VI.

Vous remarquerez que les deux courbes sont bien tracées dans le même graphe oscilloscopique. La valeur initiale de X est prise par défaut, soit 0, de même que la valeur de delta-X, soit 1, pour les deux ensembles de données.



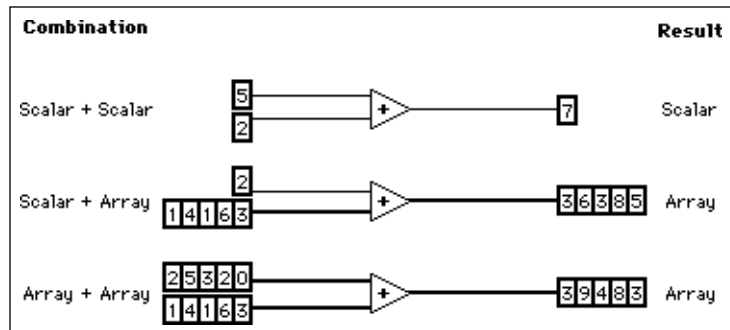
Remarque : *vous avez la possibilité de modifier le style du graphe. Pour ce faire, ouvrez un menu local en cliquant dans la légende correspondant au tracé voulu. Vous pouvez par exemple obtenir un graphe sous la forme de barres à la place de lignes, en choisissant les options **Common Plots»Bar Graph**.*

3. Enregistrez et fermez le VI. Nommez-le My Graph Waveform Arrays.vi. Assurez-vous que vous avez bien enregistré votre travail dans le répertoire mywork.llb.

Le polymorphisme

On appelle polymorphisme la capacité que possède une fonction à s'adapter à des données d'entrée de types, dimensions ou représentations différents. La plupart des fonctions de LabVIEW sont dotées de cette capacité. Le diagramme précédent est un exemple de polymorphisme. Vous remarquerez que la fonction **Multiply** est utilisée à deux endroits, à l'intérieur et à l'extérieur de la boucle *For*. A l'intérieur de la boucle *For*, la fonction multiplie deux valeurs scalaires. A l'extérieur, elle multiplie un tableau par une valeur scalaire.

L'exemple suivant présente quelques-unes des combinaisons polymorphes de la fonction **Add**.



Dans la première combinaison, les deux valeurs scalaires sont additionnées, le résultat obtenu étant une autre valeur scalaire. Dans le deuxième cas de figure, la valeur scalaire est ajoutée séparément à chaque élément du tableau, ce qui donne un nouveau tableau. Dans la troisième combinaison, chaque élément du tableau est ajouté à l'élément correspondant d'un autre tableau. Vous avez également la possibilité d'utiliser d'autres combinaisons, comme les *clusters* de numériques, les tableaux de *clusters*, et ainsi de suite.

Ces principes peuvent également s'appliquer à d'autres types de données et de fonctions de LabVIEW. Les fonctions LabVIEW sont chacune plus ou moins polymorphes. Certaines acceptent des entrées numériques et booléennes, d'autres une combinaison de n'importe quel type de données.

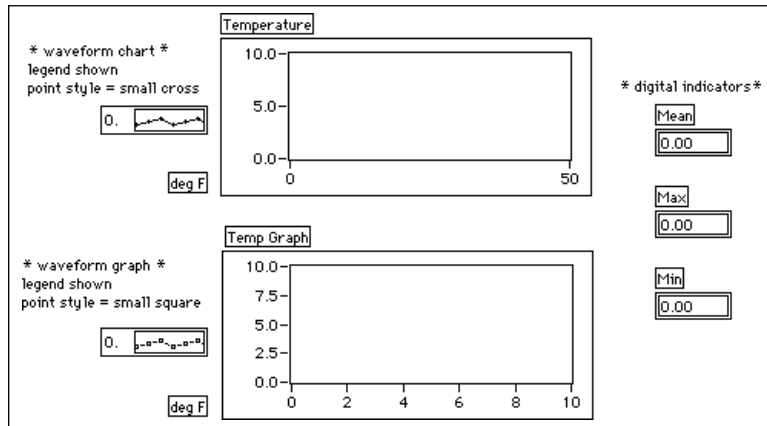
Mise en œuvre des VIs d'analyse et de graphes

OBJECTIF

Créer un VI qui mesure une température toutes les 0,25 secondes pendant 10 secondes. En cours d'acquisition, le VI visualise les mesures en temps réel sur un graphe déroulant. Lorsque le processus d'acquisition est terminé, le VI trace un graphe mettant en surbrillance les températures moyenne, maximale et minimale.

Pour des exemples de VIs d'analyse, reportez-vous au répertoire `examples\analysis`.

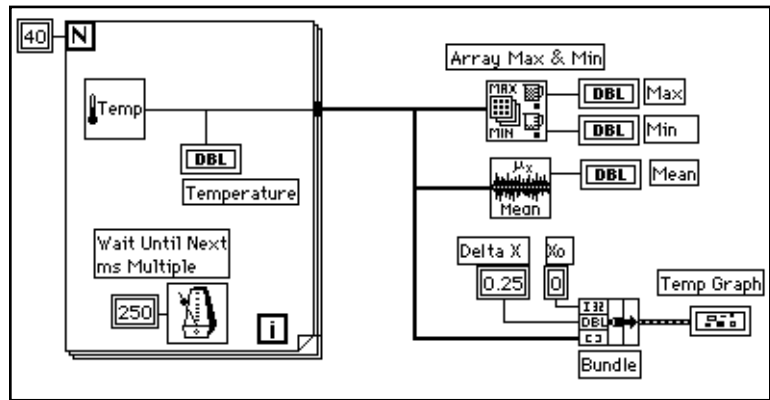
La face-avant



1. Ouvrez une nouvelle fenêtre puis construisez la face-avant reproduite dans l'illustration précédente. Vous pouvez modifier les styles des points des graphes déroulants et des graphes oscilloscopiques en ouvrant un menu local sur leurs légendes.

Le graphe déroulant *Temperature* affichera l'évolution de la température au fur et à mesure de l'acquisition. En fin d'acquisition, le VI tracera les données dans *Temp Graph*. Les indicateurs numériques *Mean*, *Max* et *Min* affichent les températures moyenne, maximale et minimale.

Le diagramme



1. Construisez le diagramme de l'illustration précédente à l'aide des éléments suivants :



Vous pouvez utiliser le VI **Digital Thermometer**

(**Functions»Tutorial**), ou bien encore le VI que vous avez construit au chapitre 2 en choisissant **Functions»Select a VI...** et en sélectionnant My Thermometer VI. Celui-ci effectue un relevé de température.



Fonction **Wait Until Next ms Multiple** (**Functions»Time & Dialog**).

Dans cet exercice, cette fonction garantit que la boucle *For* s'exécutera toutes les 0,25 secondes (250 millisecondes).



Numeric Constant (**Functions»Numeric**). Vous avez également la possibilité d'ouvrir un menu local sur la fonction **Wait Until Next ms Multiple** et de choisir **Create Constant** afin de créer et de câbler automatiquement la constante numérique.



Fonction **Array Max & Min** (**Functions»Array**). Dans cet exercice, cette fonction permet de retenir les minimum et maximum des températures relevées au cours de l'acquisition.



Le VI **Mean** (**Functions»Analysis»Probability and Statistics**) permet d'obtenir la moyenne des relevés de température.



Fonction **Bundle** (**Functions»Cluster**). Elle rassemble les différents éléments du tracé dans un *cluster*. Parmi les éléments en question, figurent la valeur initiale de X, égale à 0, la valeur de delta-X, égale à

0,25, et le tableau Y des données de température. Utilisez l'outil Flèche pour redimensionner la fonction en tirant sur l'un de ses coins.

La boucle *For* s'exécute 40 fois de suite. La fonction **Wait Until Next ms Multiple** déclenche une itération toutes les 250 millisecondes. Le VI conserve les relevés de température dans chacun des tableaux créés en bordure de la boucle *For* (auto-indexation). Lorsque l'exécution de la boucle *For* est terminée, le tableau exécute plusieurs nœuds.

La fonction **Array Max & Min** retourne les températures minimale et maximale. Le VI retourne la moyenne des relevés de température.

Le VI complet regroupe les données dans un tableau en prenant comme valeur initiale de X, 0 et comme valeur de delta-X, 0,25. Cette valeur de delta-X égale à 0,25 est nécessaire pour que le VI trace sous la forme d'un graphe oscilloscopique l'évolution de la température toutes les 0,25 secondes.

2. Revenez dans la face-avant et exécutez le VI.
3. Enregistrez le VI dans `mywork.llb` sous le nom de `My Temperature Analysis.vi`.

Mise en œuvre des tableaux

LabVIEW offre de nombreuses fonctions permettant de manipuler les tableaux dans la palette **Functions»Array**. Quelques-unes des fonctions les plus courantes sont abordées ici.

Création et initialisation des tableaux

Si vous avez besoin d'un tableau comme source de données dans votre diagramme, vous pouvez choisir la fonction **Functions»Array** pour y sélectionner une constante tableau vierge que vous positionnerez dans votre diagramme. Vous pouvez aussi choisir à l'aide de l'outil Doigt une constante numérique, booléenne ou chaîne de caractères que vous placerez dans le tableau vide. L'illustration suivante reprend un exemple de constante tableau vierge avec une constante numérique intégrée dans le tableau vierge.

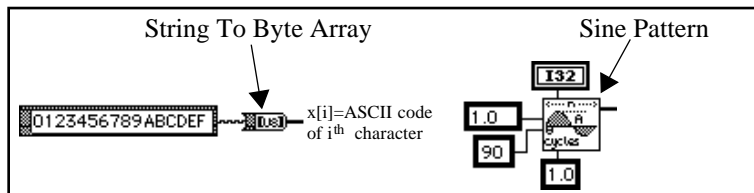




Remarque : *vous pouvez aussi créer un tableau et ses commandes sur la face-avant puis copier ou faire glisser la commande de tableau à l'intérieur du diagramme pour reproduire la constante correspondante.*

Pour plus d'informations sur la manière de créer des commandes et des indicateurs de tableaux sur la face-avant, veuillez vous reporter au chapitre 15, *Les commandes et les indicateurs de clusters et de tableaux*, du *Manuel de l'utilisateur LabVIEW*.

Il existe plusieurs façons de créer et d'initialiser des tableaux dans le diagramme. Vous savez déjà comment créer des tableaux au pourtour des boucles et comment utiliser la fonction **Initialize Array**. Certaines fonctions du diagramme permettent aussi de produire des tableaux comme le montre l'illustration suivante.



Structures Condition, structures Séquence et boîte de calcul



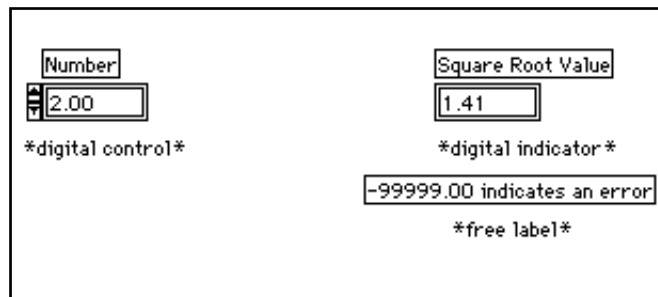
Vous allez apprendre :

- Comment utiliser la structure Condition.
- Comment utiliser la structure Séquence.
- Ce que sont les variables locales de séquence et comment les utiliser.
- Ce qu'est une boîte de calcul (*formula node*) et comment l'utiliser.

Mise en œuvre d'une structure Condition

OBJECTIF Construire un VI qui teste un nombre pour voir s'il est positif ou non. S'il est positif, le VI calcule alors sa racine carrée, dans le cas contraire, il signale une erreur.

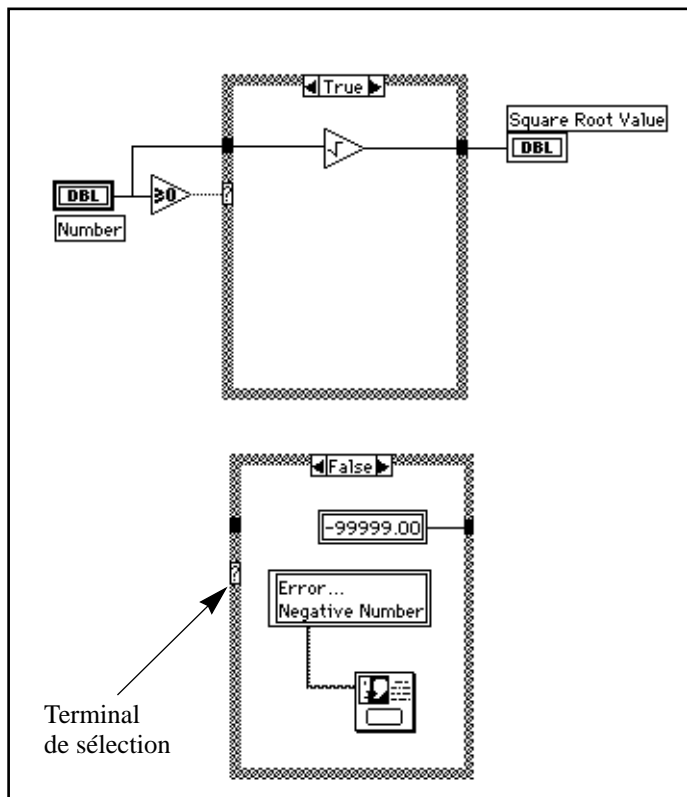
La face-avant



1. Ouvrez une nouvelle fenêtre de face-avant et construisez la face-avant de l'illustration précédente.

La commande **Number** fournit le nombre. L'indicateur **Square Root Value** affiche la racine carrée de ce nombre. Le texte libre fait office de remarque à l'attention de l'utilisateur.

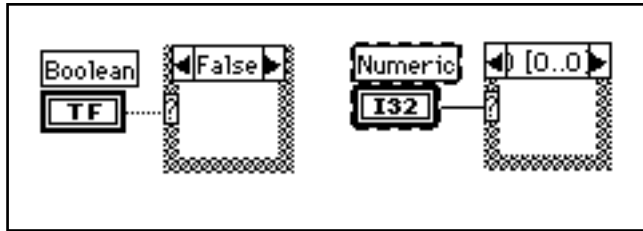
Le diagramme



1. Ouvrez la fenêtre du diagramme.
2. Placez une structure Condition (**Functions»Structures**) dans le diagramme. Agrandissez la structure Condition en tirant sur l'un de ses coins à l'aide du curseur de redimensionnement.

Par défaut, la structure Condition est booléenne et ne peut présenter que deux conditions, à savoir : TRUE ou FALSE. Une structure Condition booléenne s'apparente aux instructions if-then-else des langages de programmation à base de texte. Elle passe

automatiquement en numérique lorsque vous câblez une commande numérique sur le terminal de sélection.



Vous ne pouvez afficher qu'une seule condition à la fois. Pour passer d'une condition à l'autre, il suffit de cliquer sur les flèches qui figurent en haut de la structure Condition.

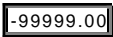
3. Choisissez les autres objets présents dans le diagramme et câblez-les en vous inspirant du diagramme de l'illustration précédente.



Fonction **Greater Or Equal To 0? (Functions»Comparison)**. Dans cet exercice, cette fonction va nous permettre de déterminer si l'entrée est négative. La fonction retourne l'état TRUE si le nombre en entrée est supérieur ou égal à 0.



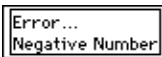
Fonction **Square Root (Functions»Numeric)**. Dans cet exercice, cette fonction calcule la racine carrée du nombre en entrée.



Numeric constant (Functions»Numeric).



Fonction **One Button Dialog (Functions»Time & Dialog)**. Dans cet exercice, cette fonction affiche une boîte de dialogue qui contient le message `Error...Negative Number`.



String Constant (Functions»String). Vous entrez du texte à l'intérieur de la boîte avec l'outil Texte.

Dans cet exercice, l'exécution du VI conduit à deux choix possibles : TRUE ou FALSE. Si le nombre est supérieur ou égal à zéro, le VI est alors dans la situation TRUE et calcule la racine carrée du nombre. Dans la situation FALSE, la sortie indique -99999.00 et la boîte de dialogue affiche le message suivant : `Error...Negative Number`.



Remarque : *vous devez définir le tunnel de sortie pour chaque cas. Lorsque vous créez un tunnel de sortie pour un cas particulier, vous créez automatiquement un tunnel au même endroit dans les autres cas. Lorsqu'un tunnel n'est pas câblé, il prend la forme d'un carré blanc.*

Veillez à câbler le tunnel de sortie sur chaque cas en cliquant sur le tunnel de sortie lui-même à chaque fois. Dans cet exercice, vous allez affecter une valeur au tunnel de sortie du cas FALSE puisque le cas TRUE a déjà un tunnel de sortie. Si vous ne souhaitez pas affecter une valeur de sortie à tous les cas, vous devez placer l'indicateur dans ce cas précis ou bien encore utiliser une variable locale ou une variable globale.

4. Revenez sur la face-avant et lancez le VI. Prenez un nombre supérieur à zéro et un nombre inférieur à zéro en modifiant la valeur de la commande numérique que vous avez nommée Number. Vous remarquerez que lorsque la commande numérique prend une valeur négative, LabVIEW affiche le message d'erreur que vous avez attribué au cas FALSE de la structure Condition.
5. Enregistrez et fermer le VI. Nommez-le My Square Root.vi.

Logique du VI

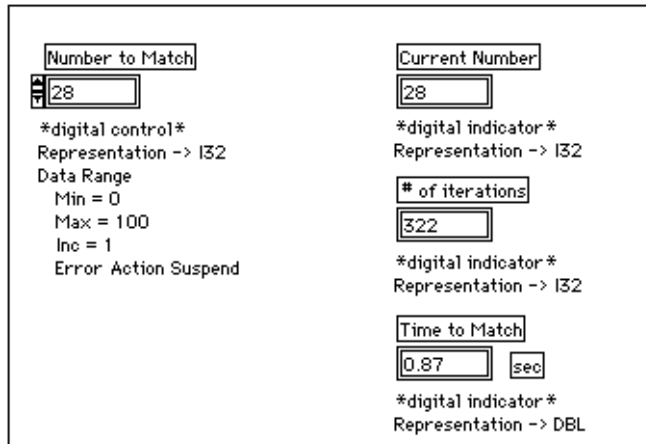
```

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


Mise en œuvre d'une structure Séquence

OBJECTIF Construire un VI qui calcule le temps nécessaire pour générer un nombre aléatoire qui corresponde à un nombre donné.

La face-avant



1. Ouvrez et construisez une nouvelle face-avant en vous inspirant de l'illustration suivante. Veillez à bien respecter les consignes données suivantes pour modifier les commandes et les indicateurs.

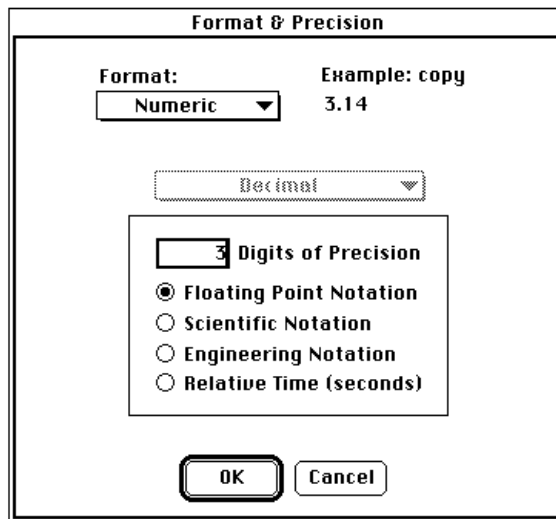
La commande **Number to Match** contient le nombre que vous souhaitez atteindre. L'indicateur **Current Number** affiche la valeur aléatoire générée. L'indicateur **# of iterations** fournit le nombre d'itérations nécessaires à la recherche. Enfin, l'indicateur **Time to Match** indique combien de secondes il a fallu pour obtenir le nombre correspondant.

Modification du format numérique

Par défaut, LabVIEW affiche les valeurs des commandes numériques au format décimal, avec deux chiffres après la virgule (symbolisée ici par un point), comme par exemple 3.14. Vous pouvez aussi utiliser l'option **Format & Precision...** d'une commande ou d'un indicateur d'un menu local pour modifier la précision ou afficher les commandes et les indicateurs numériques conformément aux usages des

scientifiques et des ingénieurs. Vous pouvez aussi utiliser l'option **Format & Precision...** pour horodater les valeurs numériques.

1. Modifiez la précision de l'indicateur Time to Match.
 - a. Accédez au menu local de l'indicateur numérique Time to Match puis choisissez **Format & Precision...** Vous devez être dans la face-avant pour pouvoir accéder à ce menu.
 - b. Entrez la valeur 3 dans le paramètre Digits of Precision, puis cliquez sur **OK**.



2. Modifiez la représentation de la commande numérique et transformez deux des indicateurs numériques en entiers longs.
 - a. Ouvrez un menu local sur la commande numérique Number to Match, puis choisissez **Representation»Long**.
 - b. Répétez l'étape précédente pour les indicateurs numériques Current Number et # of iterations.


Délimitation de la gamme de données

Avec l'option **Data Range...**, vous pouvez éviter qu'un opérateur ne positionne la valeur d'une commande ou d'un indicateur en dehors d'une gamme ou d'une incrémentation prédéfinie. Vous pouvez soit ignorer la valeur, la forcer à rester à l'intérieur de la gamme prédéfinie ou bien encore interrompre définitivement l'exécution. Le symbole d'erreur de gamme remplace le bouton Exécution dans la barre d'outils, lorsqu'une erreur de gamme interrompt l'exécution. De plus, un cadre sombre et épais entoure la commande incriminée.



1. Fixez la gamme entre 0 et 100 avec un incrément de 1.
 - a. Ouvrez un menu local sur l'indicateur Time to Match puis choisissez **Data Range...**
 - b. Complétez la boîte de dialogue comme indiqué dans l'illustration suivante, puis cliquez sur **OK**.

Representation



132
31 0

Long

Minimum

Maximum

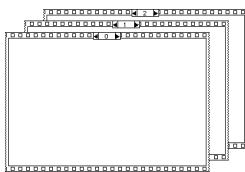
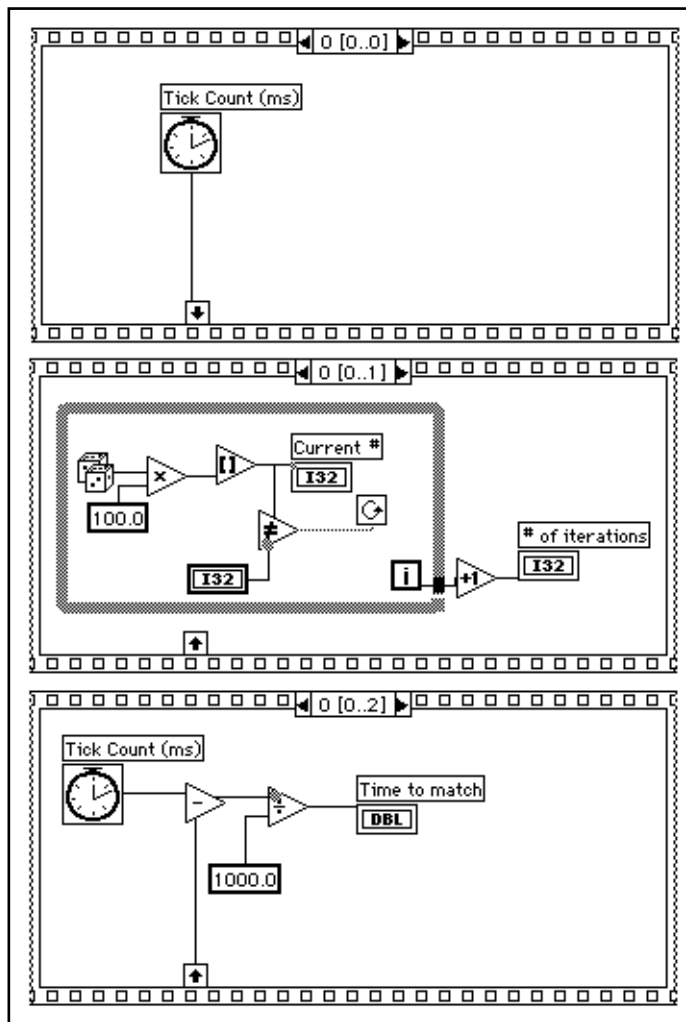
Increment

Default

If Value is Out of Range:

▼

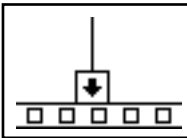
Le diagramme



1. Ouvrez la fenêtre du diagramme.
2. Placez la structure Séquence (**Functions»Structures**) dans la fenêtre du diagramme.

La structure Séquence qui se présente un peu comme les trames successives d'un film, exécute séquentiellement des diagrammes. Dans les langages de programmation classiques, les instructions s'exécutent suivant leur ordre d'apparition. Alors que dans la programmation par flux des données, un nœud ne s'exécute que lorsque toutes les données sont disponibles à ses entrées. Toutefois, il est parfois nécessaire d'exécuter un nœud avant un autre. Le cas échéant, LabVIEW utilise la structure Séquence comme une méthode permettant de contrôler l'ordre d'exécution des nœuds. LabVIEW place le diagramme que le VI exécute en premier à l'intérieur du cadre 0, puis place le diagramme qu'il exécute en deuxième à l'intérieur du cadre 1, et ainsi de suite. Comme dans le cas de la structure Condition, vous ne pouvez voir qu'un seul cadre à la fois.

3. Agrandissez la structure en tirant sur l'un de ses coins à l'aide du curseur de redimensionnement.
4. Pour créer un nouveau cadre, ouvrez un menu local en bordure du cadre et choisissez **Add Frame After**. Procédez de même pour créer le cadre 2.



Le cadre 0 de l'illustration précédente contient un petit carré avec une flèche à l'intérieur. Ce petit carré représente une variable locale qui achemine les données entre les cadres d'une structure Séquence. Vous pouvez créer des variables locales sur la bordure des cadres. Les données câblées à la variable locale d'un cadre sont alors disponibles dans les cadres suivants. Néanmoins, il convient de noter que vous ne pouvez pas accéder aux données des cadres précédant la création du cadre contenant la variable locale.

5. Créez cette variable locale en ouvrant un menu local sur la bordure inférieure du cadre 0 puis en choisissant **Add Sequence Local**.

La variable locale se présente sous la forme d'un carré vide. La flèche à l'intérieur du carré apparaît automatiquement dès que vous câblez une fonction à la variable locale.

6. Terminez le diagramme jusqu'à ce qu'il ressemble à celui de l'illustration reproduite dans la section *Diagramme*.



Fonction **Tick Count (ms)** (**Functions**»**Time & Dialog**). Cette fonction donne le nombre de millisecondes qui se sont écoulées depuis la mise sous tension. Pour cet exercice, vous aurez besoin de deux fonctions **Tick Count**.



Fonction **Random Number (0-1) (Functions»Numeric)**. Cette fonction donne un nombre aléatoire compris entre 0 et 1.



Fonction **Multiply (Functions»Numeric)**. Dans cet exercice, cette fonction multiplie le nombre aléatoire par 100. C'est-à-dire qu'elle génère un nombre aléatoire compris entre 0.0 et 100.0.



Fonction **Numeric Constant (Functions»Numeric)**. Dans cet exercice, la constante numérique représente le nombre maximum pouvant être multiplié.



Fonction **Round to Nearest (Functions»Numeric)**. Dans cet exercice, cette fonction arrondit le nombre aléatoire compris entre 0 et 100 au nombre entier le plus proche.



Fonction **Not Equal? (Functions»Comparison)**. Dans cet exercice, cette fonction compare le nombre aléatoire au nombre indiqué dans la face-avant et donne l'état TRUE si les deux nombres sont différents. Dans le cas contraire, cette fonction donne l'état FALSE.



Fonction **Increment (Functions»Numeric)**. Dans cet exercice, cette fonction incrémente le comptage de la boucle *While* de 1.



Fonction **Subtract (Functions»Numeric)**. Dans cet exercice, cette fonction fournit le temps écoulé entre le cadre 2 et le cadre 0 en millisecondes.



Fonction **Divide (Functions»Numeric)**. Dans cet exercice, cette fonction divise le nombre de millisecondes écoulées par 1000 pour convertir le nombre en secondes.



Fonction **Numeric Constant (Functions»Numeric)**. Dans cette exercice, cette fonction convertit le nombre exprimé en millisecondes en secondes.

Dans le cadre 0, la fonction **Tick Count (ms)** fournit le temps en millisecondes. Cette valeur est câblée à la variable locale, pour être transmise aux séquences suivantes. Dans le cadre 1, le VI exécute la boucle *While* tant que le nombre indiqué ne correspond pas au nombre délivré par la fonction **Random Number (0-1)**. Dans le cadre 2, la fonction **Tick Count (ms)** donne le nouveau temps exprimé en millisecondes. Le VI soustrait le temps enregistré dans la séquence 0 (transmis par la variable locale) au nouveau temps, de façon à calculer le temps écoulé lors de la séquence 1.

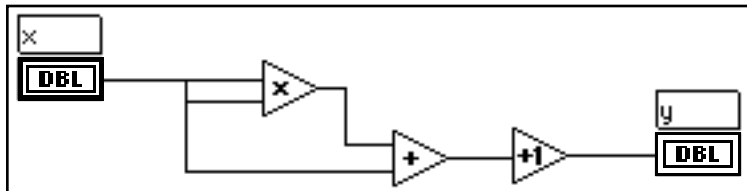
7. Revenez dans la face-avant puis entrez un nombre à l'intérieur de la commande `Number to Match` puis lancez le VI.
8. Enregistrez et fermez le VI. Nommez-le `My Time to Match.vi`.

La boîte de calcul

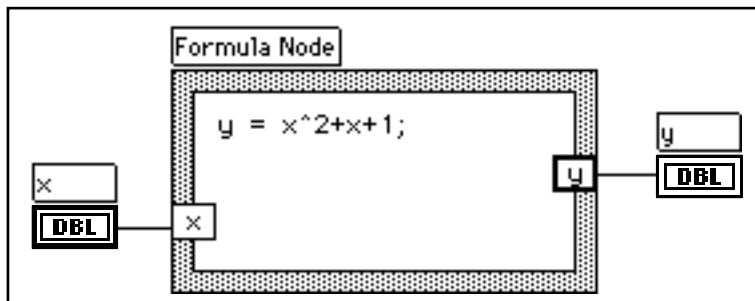
La boîte de calcul (*formula node*) est une fenêtre réajustable qui permet de saisir des formules mathématiques directement dans un diagramme. Pour ce faire, vous placez la boîte de calcul dans le diagramme à partir de la palette **Function»Structures**. Cette méthode est très pratique lorsque l'équation contient plusieurs variables ou qu'elle est relativement compliquée. Par exemple, considérez l'équation suivante :

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

Si vous effectuez cette équation en utilisant les fonctions arithmétiques habituelles de LabVIEW, vous obtiendrez un diagramme semblable à celui reproduit ci-dessous.



Vous pouvez effectuer cette même équation en utilisant une boîte de calcul, identique à celle de l'illustration suivante.



Grâce à la boîte de calcul, vous pouvez entrer directement une formule complexe au lieu de créer plusieurs sous-sections de diagrammes. Vous entrez les formules avec l'outil Texte. Vous créez les terminaux d'entrée et de sortie de la boîte de calcul en ouvrant un menu local en bordure du nœud et en choisissant **Add Input (Add Output)**. Tapez ensuite le nom de la variable dans le carré. Sachez que les variables

sont différentes selon que vous utilisez des majuscules ou des minuscules pour les saisir. Vous entrez la ou les formule(s) mathématique(s) à l'intérieur du carré. Chaque élément de la formule mathématique doit impérativement se terminer par un point-virgule (;).

La liste des opérateurs et des fonctions disponibles dans la boîte de calcul figure dans la fenêtre d'aide correspondante, comme le montre l'illustration suivante. Un point-virgule se trouve à la fin de chaque formule.

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

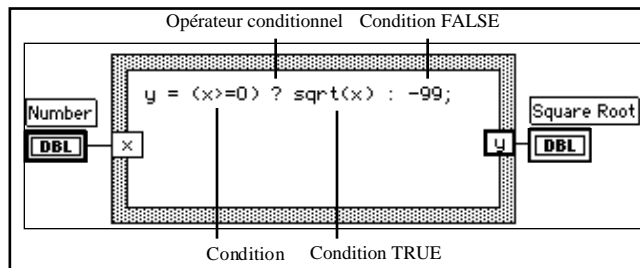
Formula Node functions:
abs acos acosh asin asinh atan atanh ceil
cos cosh cot csc exp expm1 floor getexp getman
int intrz ln ln1 log log2 max min mod rand
rem sec sign sin sinc sinh sqrt tan tanh
```

L'exemple suivant vous explique comment réaliser un branchement conditionnel à l'intérieur d'une boîte de calcul.

Imaginez que vous vouliez calculer la racine carrée de x . Si x est positif, le résultat du calcul est attribué à y . Si x est négatif, le code -99 est attribué à y .

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

Vous pouvez parvenir au même résultat en utilisant une boîte de calcul, comme celle reproduite dans le diagramme ci-après.



Mise en œuvre d'une boîte de calcul

OBJECTIF Construire un VI qui utilise un *Formula Node* pour calculer les équations suivantes :

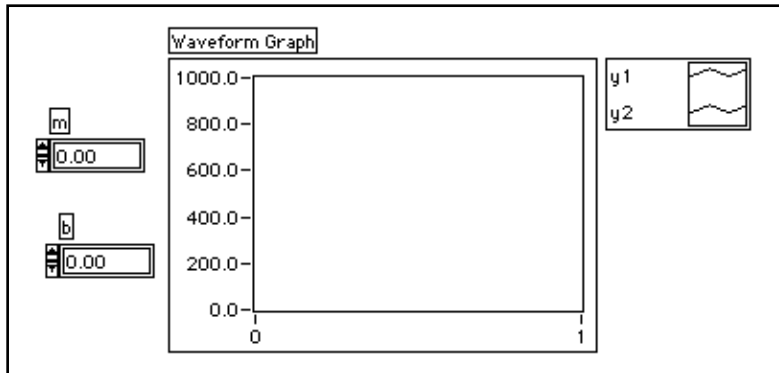
$$y1 = x^3 - x^2 + 5$$

$$y2 = m * x + b$$

où x varie entre 0 et 10.

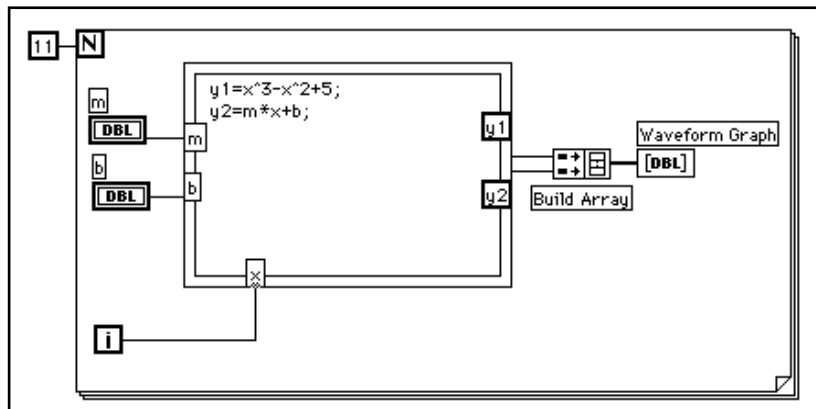
Vous n'utiliserez qu'une seule boîte de calcul pour résoudre les deux équations en superposant les résultats dans un seul graphe.

La face-avant



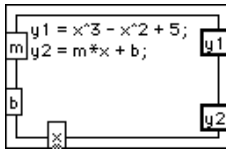
1. Ouvrez une nouvelle fenêtre de face-avant puis construisez la face-avant présentée dans l'illustration précédente. L'indicateur du graphe oscilloscopique affiche les tracés de l'équation. Le VI utilise les deux commandes numériques pour entrer les valeurs à attribuer à m et à b .

Le diagramme



1. Reproduisez le diagramme de l'illustration précédente.
2. Placez la boucle *For* (**Functions»Structures**) dans le diagramme et tirez sur le coin pour agrandir la boucle.





Formula Node (Functions»Structures). Avec cette boîte de calcul, vous pouvez entrer directement une ou plusieurs formule(s). Créez trois terminaux d'entrée en ouvrant un menu local en bordure du cadre et en choisissant **Add Input**. De même, vous créez le terminal de sortie en choisissant **Add Output** à partir du menu local.

Lorsque vous créez un terminal d'entrée ou de sortie, vous devez lui donner un nom de variable. Ce nom doit exactement correspondre au nom que vous utilisez dans la formule. Des noms apparemment identiques seront pourtant différents s'ils sont en majuscules ou en minuscules. Par exemple, si vous utiliser un a minuscule pour nommer le terminal, vous devez obligatoirement utiliser ce même a minuscule dans la formule. Pour cela, vous pouvez entrer les noms de variables et de formules avec l'outil Texte.



Remarque : *même si les noms de variables ne sont pas limités en longueur, ne perdez pas de vue que des noms trop longs vont occuper beaucoup d'espace dans le diagramme. Un point-virgule (;) doit être placé à la fin de chaque formule.*



Numeric Constant (Functions»Numeric). Vous pouvez également ouvrir un menu local sur le terminal de comptage et choisir **Create Constant** pour créer et câbler automatiquement la constante numérique. Celle-ci indique le nombre d'itérations de la boucle *For*. Si x varie entre 0 et 10 inclus, vous devez câbler 11 au terminal de comptage.



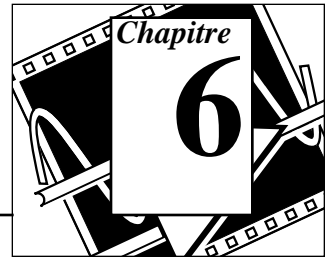
Comme le terminal d'itération compte de 0 à 10, il sert à contrôler la valeur X dans la boîte de calcul.



Build Array (Functions»Array) rassemble les deux entrées du tableau en un graphe multicourbes. Créez les deux terminaux d'entrée en utilisant le curseur de redimensionnement pour tirer sur l'un des coins.

3. Revenez dans la face-avant et lancez le VI avec différentes valeurs pour m et b .
4. Enregistrez et fermez le VI. Nommez-le `My Equations.vi`.

Chaînes de caractères et E/S sur fichier



Vous allez apprendre :

- Comment créer des indicateurs et des commandes de type chaînes de caractères.
- Comment utiliser les fonctions de manipulation de chaînes de caractères.
- A manipuler les entrées et sorties sur fichier.
- Comment enregistrer des données dans un fichier au format tableur.
- Comment écrire ou lire des données de fichiers texte.

Les chaînes de caractères

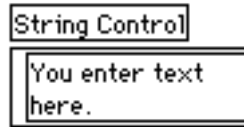
Une chaîne de caractères est un ensemble de caractères ASCII. Leur utilisation va bien au-delà du cadre des messages textuels. Dans le domaine du contrôle d'instruments, vous pouvez transmettre des données numériques sous la forme de chaînes de caractères, puis convertir ces chaînes en chiffres. Le stockage de données numériques sur disque peut se faire sous la forme de chaînes de caractères. Pour stocker des nombres dans un fichier ASCII, il vous faut d'abord convertir ces nombres en chaînes de caractères avant de les écrire dans un fichier sur le disque.

Création des indicateurs et commandes de type chaînes de caractères



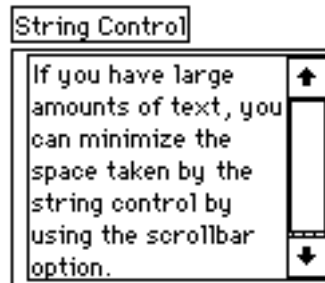
Vous retrouverez les indicateurs et commandes de type chaînes de caractères présentés à gauche dans la palette **Controls»String & Table**. Vous pouvez entrer ou bien encore modifier du texte à l'intérieur d'une commande de chaînes de caractères avec l'outil Doigt ou l'outil Texte.

Pour agrandir les indicateurs et commandes, cliquez sur un coin et faites-le glisser à l'aide de l'outil Doigt.



Les chaînes de caractères et E/S sur fichier

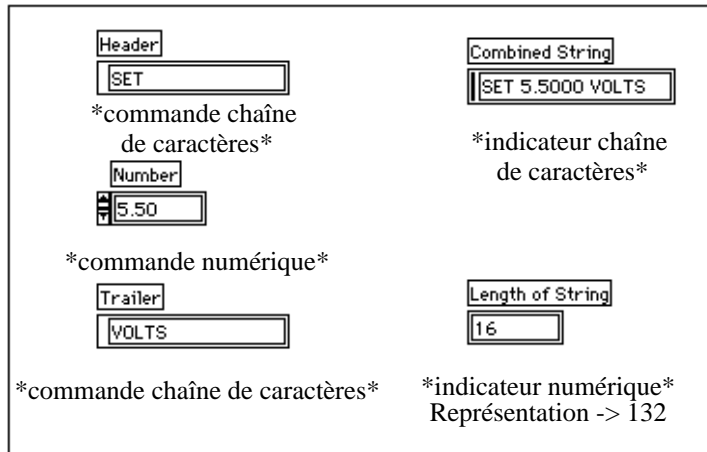
Si vous souhaitez réduire au minimum l'espace occupé par une commande ou un indicateur de chaînes sur la face-avant, choisissez l'option **Show»Scrollbar**. Si cette option est grisée, il vous faut agrandir la fenêtre pour la rendre visible.



Mise en œuvre des fonctions de chaînage

OBJECTIF LabVIEW offre de nombreuses fonctions permettant de manipuler les chaînes de caractères. Ces fonctions sont disponibles dans la palette **Functions»String**. Vous allez construire un VI qui convertit un nombre en chaîne de caractères et qui va ensuite concaténer cette chaîne à d'autres chaînes de caractères pour ne donner qu'une seule chaîne de caractères. Le VI détermine également la longueur de la chaîne obtenue.

La face-avant

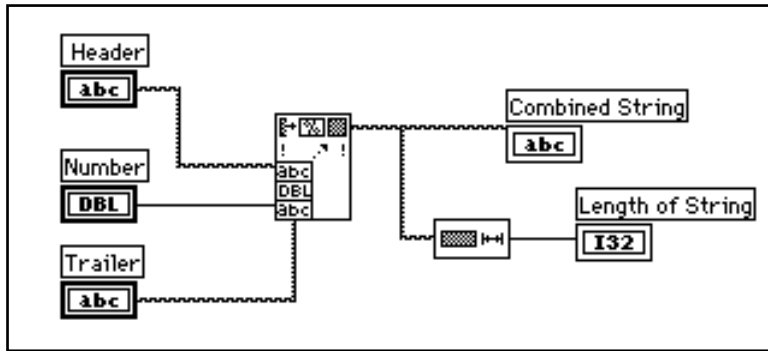


1. Ouvrez et construisez une nouvelle face-avant en vous inspirant de celle de l'illustration précédente. Veillez à modifier correctement les indicateurs et les commandes comme indiqué.

Les deux commandes de chaînes de caractères et la commande numérique peuvent parfaitement se combiner pour ne former qu'une seule chaîne de caractères de sortie, et s'afficher dans l'indicateur de chaîne de caractères. L'indicateur numérique sert à afficher la longueur de la chaîne.

Dans cet exercice, la sortie **Combined String** présente un format comparable aux chaînes des commandes utilisées pour communiquer avec les instruments GPIB (IEEE 488) et série (RS-232 ou RS-422). Pour en savoir plus sur les chaînes utilisées dans les commandes d'instruments, veuillez vous reporter au chapitre 7, intitulé *L'acquisition de données et le contrôle d'instruments*, de ce même guide de démonstration.

Le diagramme



1. Construisez le diagramme de l'illustration précédente.



La fonction **Format Into String (Functions»String)** permet de concaténer et de formater les nombres et les chaînes pour ne former qu'une seule chaîne de sortie. Utilisez le curseur de redimensionnement et placez-le sur l'icône pour ajouter trois entrées supplémentaires.



La fonction **String Length (Functions»String)** indique le nombre de caractères contenus dans la chaîne concaténée.

2. Lancez le VI. Vous remarquerez que la fonction **Format Into String** a concaténé les deux commandes de chaînes de caractères et la commande numérique en une seule chaîne de caractères en sortie.
3. Enregistrez le VI sous le nom *My Build String.vi*. Vous reprendrez ce VI au cours du prochain exercice.
- 4.

E/S sur fichier

Les fonctions d'entrées/sorties (E/S) sur fichier de LabVIEW (**Functions»File I/O**) constituent un ensemble d'outils à la fois puissants et simples d'utilisation pour travailler avec des fichiers de données. En plus d'assurer la lecture et l'écriture des données, ces fonctions permettent également de déplacer et de renommer des fichiers et des répertoires, de créer des fichiers au format tableur avec

du texte lisible au format ASCII, et d'écrire des données en format binaire pour gagner de l'espace et de la vitesse.

Vous pouvez stocker et extraire des données des fichiers selon trois formats différents.

- Format ASCII. Vous devez stocker des données dans le format ASCII lorsque vous prévoyez d'y accéder à partir d'un autre logiciel tel qu'un traitement de texte ou un tableur. Pour ce faire, vous devez convertir toutes les données en chaînes de caractères ASCII.
- Format Datalog. Ces fichiers au format binaire ne sont accessibles que par LabVIEW uniquement. Les fichiers Datalog de LabVIEW sont similaires aux fichiers de bases de données en ce sens qu'ils offrent la possibilité de stocker plusieurs types de données différents dans un seul enregistrement (log) de fichier.
- Format binaire. Avec ce format de fichiers, vous disposez de la méthode la plus compacte et la plus rapide pour stocker les données. Vous devez convertir les données en chaînes binaires et savoir précisément à quels types de données vous avez affaire pour sauvegarder et extraire les données de ces fichiers.

Cette section aborde en particulier les fichiers au format ASCII, qui reste le plus courant des formats.

Les fonctions E/S sur fichier

La plupart des opérations d'E/S sur fichier supposent trois étapes de base, à savoir : l'ouverture d'un fichier existant ou la création d'un nouveau fichier, l'écriture ou la lecture du fichier, et enfin, la fermeture du fichier. A ce titre, LabVIEW contient de nombreux VIs utilitaires dans la palette **Functions** » **File I/O**. Dans cette section, nous allons aborder les neuf utilitaires de haut niveau. Ces fonctions s'articulent autour de VIs de niveau intermédiaire qui prévoient la gestion d'erreurs parmi les fonctions d'E/S sur fichier.

Vous pouvez également définir un séparateur ou une chaîne de séparateurs, comme les marques de tabulations, les virgules, etc. dans votre tableur. Cette opération vous évitera d'avoir à analyser le tableur si vous avez utilisé un séparateur différent du séparateur défini par défaut, en l'occurrence la marque de tabulation, lorsque vous avez configuré le tableur.



Le VI **Write Characters To File** écrit une chaîne de caractères dans un nouveau fichier ou l'ajoute dans un fichier existant. Ce VI ouvre ou crée le fichier, écrit les données puis ferme le fichier.



Le VI **Read Characters From File** lit un nombre précis de caractères dans un fichier à partir d'une position spécifiée. Ce VI ouvre au préalable le fichier puis le ferme.



Le VI **Read Lines From File** lit un nombre de lignes précis dans un fichier à partir d'une position spécifiée. Ce VI ouvre au préalable le fichier puis le ferme.



Le VI **Write To Spreadsheet File** convertit un tableau 1D ou 2D de nombres en simple précision en une chaîne de texte, puis écrit cette chaîne dans un nouveau fichier ou l'ajoute dans un fichier existant. Il est également possible de transposer les données. Ce VI ouvre ou crée au préalable le fichier puis le ferme. Vous pouvez utiliser ce VI pour créer des fichiers texte lisibles par la plupart des tableurs.



Le VI **Read From Spreadsheet File** lit un nombre précis de lignes ou de colonnes dans un fichier alphanumérique à partir d'une position spécifiée. Il convertit ensuite les données en un tableau 2D de nombres en simple précision. Vous pouvez également transposer le tableau. Ce VI ouvre au préalable le fichier puis le ferme. Vous pouvez utiliser ce VI pour lire des fichiers au format tableur sauvegardés au format texte.

Écriture de données dans un fichier tableur

Une façon très courante de sauvegarder des données dans un fichier consiste à formater le fichier texte de façon à pouvoir l'ouvrir avec un tableur. Dans la plupart des tableurs, des tabulations séparent les colonnes et des caractères de fin de ligne (EOL) séparent les lignes, comme dans la figure suivante :

0.00	↵	0.4258	¶	↵ = Tabulation
1.00	↵	0.3073	¶	¶ = Séparateur de lignes
2.00	↵	0.9453	¶	
3.00	↵	0.9640	¶	
4.00	↵	0.9517	¶	

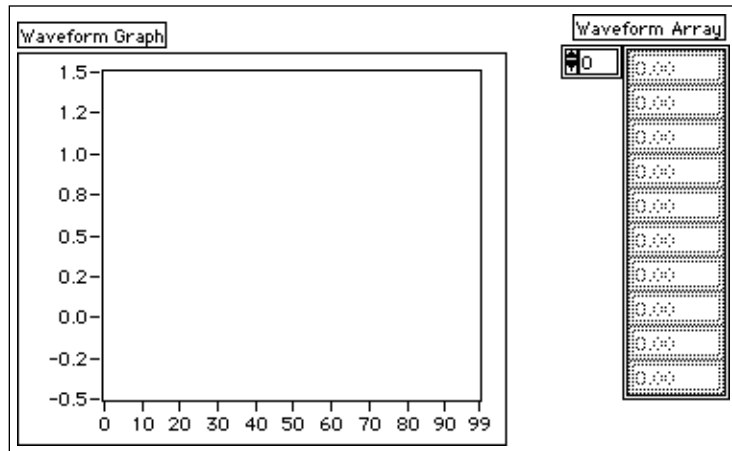
Lorsque vous ouvrez un fichier en utilisant un tableur, vous obtenez le tableau suivant :

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

OBJECTIF

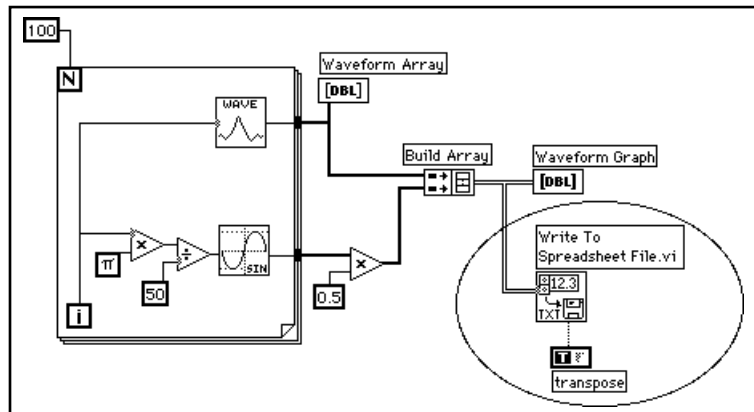
Modifier un VI existant pour y intégrer une fonction d'E/S sur fichier, de façon à enregistrer les données dans un nouveau fichier au format ASCII. Vous pourrez alors accéder à ce fichier à partir d'un tableur.

La face-avant



1. Ouvrez My Graph Waveform Arrays.vi que vous avez construit au chapitre 4 de ce guide de démonstration. Ce VI génère deux tableaux de données qu'il trace sur un même graphe. Vous allez modifier ce VI pour écrire les deux tableaux dans un fichier où chaque colonne contient un tableau de données.

Le diagramme



- Ouvrez le diagramme de My Graph Waveform Arrays puis modifiez le VI en y ajoutant les éléments présents dans l'ovale, comme indiqué ci-dessus.



Le VI **Write To Spreadsheet File (Functions»File I/O)** convertit le tableau à deux dimensions en chaîne de caractères au format tableur et l'écrit dans un fichier. Si vous n'avez pas indiqué le nom du chemin, une boîte de dialogue apparaît qui vous invite à entrer un nom de fichier. **Write To Spreadsheet File** écrit un tableau à une ou deux dimension(s) dans un fichier. Comme dans notre exemple, le tableau final étant déjà un tableau de données 2D, vous n'avez pas à le câbler à l'entrée 1D. Avec ce VI, vous pouvez utiliser un séparateur ou une chaîne de séparateurs de tableur, comme les marques de tabulation ou les virgules dans vos données.



Boolean Constant (Functions»Boolean). Cette constante permet de savoir si LabVIEW peut transposer ou non le tableau 2D avant de l'écrire dans le fichier. Pour que l'état de la valeur soit dans la situation TRUE, cliquez sur la constante à l'aide de l'outil Doigt. Dans ce cas, vous voulez que les données soient transposées parce que les tableaux de données sont relatifs aux lignes (chaque ligne du tableau à deux dimensions est un tableau de données). Comme chaque colonne du fichier tableur doit contenir un tableau de données, le tableau 2D doit être transposé en premier.

- Retournez dans la face-avant et lancez le VI. Une fois les tableaux de données générés, une boîte de dialogue de fichier vous invite à donner le nom du nouveau fichier que vous êtes en train de créer. Entrez un nom de fichier puis cliquez sur **OK**.



Attention : *n'essayez pas d'écrire des données dans des bibliothèques de VIs telles que mywork.llb. Vous risqueriez d'écrire par dessus votre bibliothèque et donc de perdre tout votre travail.*

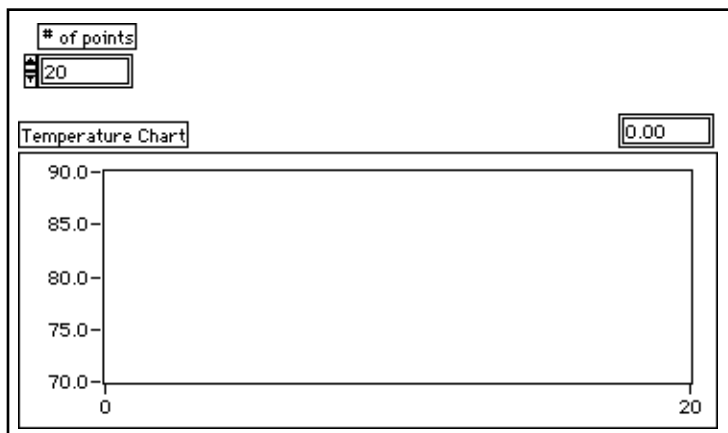
- Enregistrez le VI sous le nom My Waveform Arrays to File.vi, puis fermez-le.
- Vous pouvez dès à présent utiliser un tableur ou un éditeur de texte pour ouvrir et visualiser le fichier que vous venez de créer. Vous devriez donc voir apparaître deux colonnes de 100 éléments.

Dans cet exemple, les données ne sont pas converties ou écrites dans le fichier avant de disposer de l'intégralité des tableaux de données. Si vous êtes en train d'acquérir de gros paquets de données ou si vous souhaitez écrire les valeurs des données sur le disque au fur et à mesure de l'acquisition, alors il vous faudra utiliser un autre VI d'E/S sur fichier.

Ajout de données à un fichier

OBJECTIF Créer un VI pour ajouter des données de température à un fichier ASCII. Ce VI utilise une boucle *For* pour générer les valeurs de température et les stocker dans un fichier. Au cours de chaque itération, vous allez convertir les données en chaîne de caractères, ajouter une virgule comme séparateur, puis ajouter la chaîne de caractères dans un fichier.

La face-avant



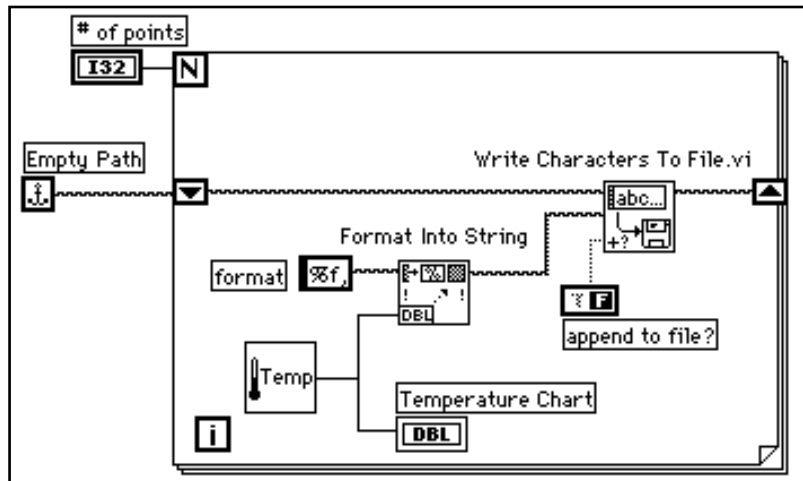
1. Ouvrez une nouvelle face-avant et placez-y les objets comme indiqué ci-dessus.

Cette face-avant contient une commande numérique et un graphe déroulant. Choisissez l'option **Show»Digital Display**. La commande # of points indique le nombre de valeurs de température à acquérir et à écrire dans le fichier. Le graphe déroulant restitue la courbe de température. Modifiez l'échelle de l'axe des Y comme suit, de 70.0 à 90.0, et celle de l'axe des X comme suit, de 0 à 20.



2. Ouvrez un menu local sur la commande numérique # of points et choisissez **Representation»Long**.

Le diagramme



1. Ouvrez le diagramme.
2. Ajoutez la boucle *For* et agrandissez-la. Ce VI génère le nombre de valeurs de température spécifié par la commande # of Points.
3. Ajoutez un registre à décalage sur la boucle en ouvrant un menu local sur sa bordure. Le registre à décalage contient le nom du chemin d'accès au fichier.
4. Finissez de câbler les objets.



Constante **Empty Path (Functions»File I/O»File Constants)**. Cette fonction permet d'initialiser le registre à décalage de sorte qu'à la première tentative d'écriture dans le fichier, le chemin d'accès soit vide. Une boîte de dialogue de fichier vous invite alors à entrer le nom de fichier.



Le VI **My Thermometer** que vous avez construit au chapitre 2 (**Functions»Select a VI...**) ou le VI **Digital Thermometer (Functions»Tutorial)** simule un relevé de température réalisé par un capteur de température.



La fonction **Format Into String (Functions»String)** convertit la mesure de température (un nombre) en chaîne de caractères et effectue la concaténation de la virgule qui suit.



Constante **String Constant (Functions»String)**. Ce format de chaîne indique que vous souhaitez convertir un nombre en une chaîne au format fractionnaire, suivie d'une virgule.



Le VI **Write Characters To File (Functions»File I/O)** écrit une chaîne de caractères dans un fichier.



Boolean Constant (Functions»Boolean) règle l'entrée `append to file?` du VI **Write Characters To File VI** sur la valeur TRUE de sorte que les nouvelles valeurs de température acquises lors de l'itération de la boucle soient ajoutées au fichier sélectionné. Avec l'outil Doigt, cliquez sur la constante pour régler sa valeur sur TRUE.

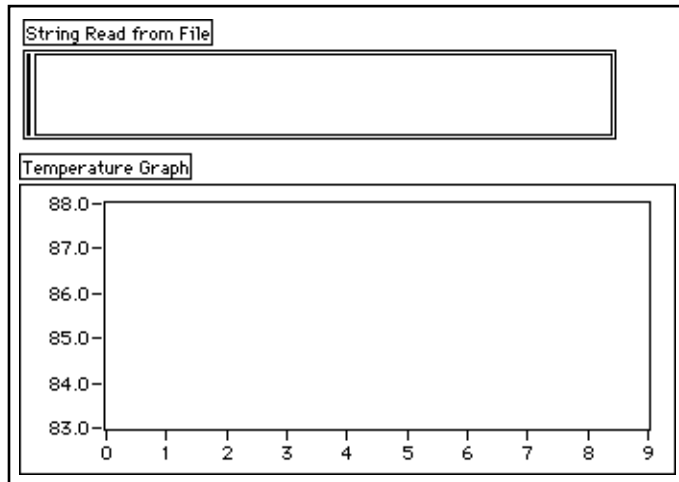
5. Retournez dans la face-avant et exécutez le VI en réglant le paramètre `# of points` sur 20. Une boîte de dialogue de fichier vous invite à entrer un nom de fichier. Lorsque c'est chose faite de fichier, le VI commence à écrire les valeurs de température dans celui-ci au fur et à mesure qu'il génère les points.
6. Enregistrez le VI sous le nom `My Write Temperature to File.vi`, puis fermez-le.
7. Utilisez n'importe quel logiciel de traitement de texte comme *Write* pour Windows, *Teach Text* pour Macintosh, et *Text Editor in Open Windows* pour UNIX pour ouvrir ce fichier de données et étudier son contenu. Vous devriez obtenir un fichier de vingt valeurs (avec trois chiffres après la virgule) séparées par des virgules.

Lecture de données en provenance d'un fichier

OBJECTIF

Créer un VI capable de lire les données que vous venez d'écrire dans l'exemple précédent et d'afficher les données sous la forme d'un graphe oscilloscopique. Attention : vous devez lire les données dans le même format que celui dans lequel vous les avez enregistrées. Par conséquent, puisque vous les avez sauvegardées au départ dans un format ASCII en utilisant des données du type chaînes de caractères, vous devez les retrouver en chaîne lors de la lecture avec un des VIs d'E/S sur fichier.

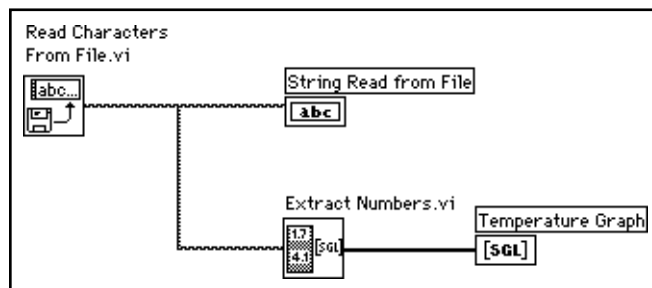
La face-avant



1. Ouvrez une nouvelle fenêtre de face-avant et construisez la face-avant reproduite ci-dessus.

La face-avant contient un indicateur de chaîne de caractères et un graphe oscilloscopique. L'indicateur **String Read from File** affiche les valeurs de températures séparées par des virgules, obtenues dans l'exemple précédent. Le graphe oscilloscopique restitue la courbe de température.

Le diagramme



1. Construisez le diagramme reproduit ci-dessus.



Le VI **Read Characters From File** (**Functions**»**File I/O**) lit les données du fichier et restitue les informations obtenues sous forme de chaîne de caractères. Si vous n'avez pas spécifié de nom de chemin, alors une boîte de dialogue vous invite à entrer un nom de fichier. Dans cet exemple, vous n'avez pas besoin d'indiquer le nombre de caractères à lire puisque celui qui figure dans le fichier est inférieur à la quantité prévue par défaut, soit 512.

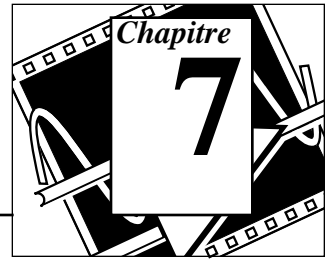
Vous devez savoir de quelle manière les données ont été stockées dans un fichier pour pouvoir les relire. Si vous connaissez la longueur d'un fichier, vous pouvez utiliser le VI **Read Characters From File** pour spécifier le nombre de caractères à lire.



Le VI **Extract Numbers** (`examples\general\strings.llb`) prend une chaîne de caractères ASCII contenant des nombres séparés par des virgules, des retours à la ligne ou tout autre caractère non numérique, puis la convertit en un tableau de valeurs numériques.

2. Retournez dans la face-avant et lancez le VI. Choisissez le fichier de données que vous venez d'écrire sur le disque lorsque la boîte de dialogue vous le demande. Vous devriez obtenir les mêmes valeurs affichées sur le graphe que celles de l'exemple du VI **My Write Temperature to File**.
3. Enregistrez le VI sous le nom `My Temperature from File.vi`, puis fermez-le.

L'acquisition de données et le contrôle d'instruments



Vous allez apprendre :

- A acquérir des données au moyen de cartes d'acquisition (**Windows, Macintosh et Sun**).
- Ce que sont les fonctions VISA.
- Ce que sont les fonctions GPIB.
- A utiliser le port de communication série de votre ordinateur.
- A utiliser un port série pour communiquer avec d'autres ports série.
- A utiliser le VXI pour l'acquisition de données (**Windows, Macintosh et Sun**).
- Ce qu'est un driver d'instrument.
- A utiliser un VI de test de réponse en fréquence.

Mise en œuvre de LabVIEW pour acquérir des données

L'une des caractéristiques les plus intéressantes de LabVIEW réside dans sa capacité à acquérir des données à partir de toutes sortes d'équipements. C'est pourquoi, LabVIEW dispose de VIs pour le contrôle des cartes et instruments suivants :

- Cartes d'acquisition de données (**Windows, Macintosh et Sun**)
- Instruments GPIB (IEEE 488)
- Instruments à port de communication série
- Instruments VXI (**Windows, Macintosh et Sun**)

Ces VIs utilisent les logiciels drivers de National Instruments, devenus depuis des standards de facto, pour vous offrir un contrôle complet de vos acquisitions de données et du matériel utilisé pour gérer votre système d'instrumentation.

Ce guide de démonstration se limite aux caractéristiques et fonctions de base de LabVIEW.

Les cartes d'acquisition de données (Windows, Macintosh et Sun)

National Instruments fabrique tous les composants nécessaires à la construction des systèmes d'acquisition de données. Les cartes enfichables sont disponibles pour tous les environnements, IBM PC/AT, EISA, IBM PS/2 MicroChannel, Macintosh NuBus Series, Macintosh LC/LCII, et les ordinateurs SPARCstation SBus.

Toutes ces cartes offrent un large éventail de combinaisons d'entrées et de sorties analogiques, numériques et d'horloge. Vous pouvez utiliser en amont des modules SCXI multiplexeurs pour le conditionnement des signaux, ce qui vous permet d'augmenter le nombre de voies d'entrées analogiques pour un prix avantageux. Une grande variété de modules de conditionnement de signaux pour thermocouples, sondes de température à résistance (RTD), entrées en tension et en courant, entrées/sorties numériques de haute puissance, complètent la gamme des produits d'acquisition.

Les VIs d'acquisition de données **DAQ** de LabVIEW permettent de contrôler l'ensemble des matériels de National Instruments.

Le contrôle d'instrumentation VISA

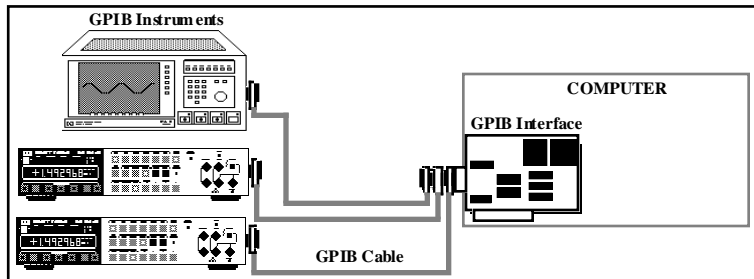
VISA est une bibliothèque d'interfaces qui permet de contrôler à la fois des instruments GPIB, VXI et d'autres types. Grâce aux fonctions VISA, vous pouvez construire un VI de driver d'instrument, qui contrôle à lui seul un modèle particulier d'instrument au moyen de différents supports d'E/S. Une chaîne de caractères est transmise à la fonction **Open** de VISA afin de choisir le type d'E/S à utiliser pour communiquer avec l'instrument en question. Une fois que la session avec l'instrument est ouverte, les fonctions VISA, telles que **VISA Read** et **VISA Write**, exécutent les activités d'E/S de l'instrument de manière générique. Ce qui signifie donc que le programme n'est pas lié aux fonctions GPIB ou VXI. Le driver d'instrument VISA se veut autonome et indépendant de l'interface et peut servir dans plusieurs systèmes différents.

Les drivers d'instrument qui utilisent les fonctions VISA s'articulent autour des activités propres à l'instrument, et non pas au support de

communication. Cette particularité offre de nombreuses possibilités quant à l'utilisation d'un driver d'instrument avec un certain nombre de programmes.

Le contrôle d'instrumentation GPIB

Le bus GPIB (*General Purpose Interface Bus*), également appelé IEEE 488, permet de communiquer avec des instruments autonomes comme les multimètres ou les oscilloscopes. National Instruments fabrique de nombreux produits pour le contrôle d'appareils via le bus GPIB. La méthode la plus simple consiste à installer une carte d'interface GPIB dans votre ordinateur et d'y connecter vos instruments au moyen d'un câble GPIB.



Les fonctions GPIB de LabVIEW contrôlent les interfaces GPIB de National Instruments. LabVIEW utilise le logiciel standard NI-488.2 de National Instruments, livré avec votre interface GPIB.

La bibliothèque GPIB (**Functions»Instrument I/O**) contient à la fois des fonctions traditionnelles GPIB et 488.2. Les fonctions GPIB 488.2 apportent à LabVIEW la compatibilité IEEE 488.2. Ces fonctions contiennent des fonctions spécifiées par la norme IEEE 488.2 et ressemblent aux routines du logiciel NI-488.2 de National Instruments.



Remarque : *dans la mesure du possible, préférez la fonction VISA à un VI GPIB, compte tenu de sa capacité d'adaptation.*

Les ports série

La liaison série est un moyen de communication très répandu pour la transmission des données entre un ordinateur et un périphérique comme une imprimante, un traceur ou un instrument programmable. La liaison série utilise un émetteur pour envoyer les données, bit par bit, sur une ligne de communication unique à destination d'un récepteur. Cette méthode de communication sert habituellement aux transferts de données à faible vitesse ou sur de longues distances. Les données peuvent ainsi être transférées via des modems sur des lignes téléphoniques traditionnelles.

Le succès de la liaison série s'explique du fait que la plupart des ordinateurs sont dotés d'un ou de deux ports série. Il convient de rappeler toutefois, qu'un port série ne peut communiquer qu'avec un seul périphérique à la fois, d'où ses limites. Pour pouvoir adapter à votre ordinateur plusieurs périphériques, il vous faut utiliser une carte dotée de plusieurs ports série ou d'un module de multiplexage du port série.

Avant de commencer à utiliser LabVIEW pour les communications série, nous vous conseillons de vérifier d'abord que l'instrument est correctement raccordé à l'ordinateur. Sous Windows, pensez également à contrôler les problèmes d'interruption. L'un des moyens d'y parvenir (Windows et Macintosh) consiste à utiliser un logiciel de gestion de terminaux standard tel que Terminal ou ZTerm de Microsoft Windows. Une fois la liaison établie avec l'instrument, vous pouvez alors utiliser les VIs de liaison série de LabVIEW qui se trouvent dans la palette **Fonctions»Instrument I/O»Serial**.

LabVIEW propose cinq VIs pour les communications série : **Serial Port Init**, **Serial Port Write**, **Serial Port Read**, **Bytes at Serial Port** et **Serial Port Break**.

Le contrôle d'instrumentation VXI pour Windows, Macintosh et Sun

Le bus VXI est une plate-forme qui se développe très rapidement, pour les systèmes d'instrumentation. Il utilise un châssis offrant une capacité maximale de treize emplacements capables de recevoir des instruments modulaires sur des cartes enfichables. Les instruments et les châssis sont disponibles en plusieurs tailles chez tous les fabricants. Vous pouvez par ailleurs utiliser des instruments de tailles différentes

dans un même châssis. Plusieurs méthodes différentes sont à votre disposition pour contrôler un châssis VXI.

LabVIEW dispose de VIs pour le contrôle de haut et de bas niveau des systèmes VXI. Vous pouvez accéder à ces VIs à partir de **Functions»Instrument I/O»VISA**.

Les drivers d'instrument

Un driver d'instrument est un logiciel qui permet de contrôler un instrument en particulier. LabVIEW convient tout particulièrement à la création de drivers d'instrument. En effet, la face-avant de LabVIEW simule le fonctionnement d'une véritable face-avant d'instrument. Le diagramme, quant à lui, envoie les commandes nécessaires pour que l'instrument effectue les opérations spécifiées dans la face-avant. Lorsque vous avez terminé la construction d'un driver d'instrument, inutile de retenir les commandes de contrôle de cet instrument. Il vous suffit de définir les valeurs de réglage sur la face-avant. Non pas que cette face-avant logicielle soit d'un grand intérêt en soi pour le contrôle de l'instrument. Là où elle devient intéressante, c'est lorsque vous utilisez le driver d'instrument en tant que sous-VI avec d'autres sous-VIs dans un VI de niveau supérieur pour contrôler un système complet.

LabVIEW dispose d'une bibliothèque composée de plus de 550 drivers d'instrument pour les appareils à interface GPIB, série, CAMAC et VXI (pour Windows, Macintosh et Sun). Compte tenu de la variété des types d'instruments, il nous est impossible de présenter les techniques de création de drivers pour tous les types d'instruments. Sachez cependant que chaque driver construit une *chaîne de commande* qu'il envoie ensuite à l'instrument pour qu'il exécute l'opération dans la face-avant virtuelle. La chaîne de commande comprend des commandes spécifiques à l'appareil (généralement en ASCII) qui le contrôlent à distance. C'est pourquoi, les drivers d'instrument contiennent davantage de fonctions réservées à la manipulation de chaînes qu'à l'interfaçage spécifique.