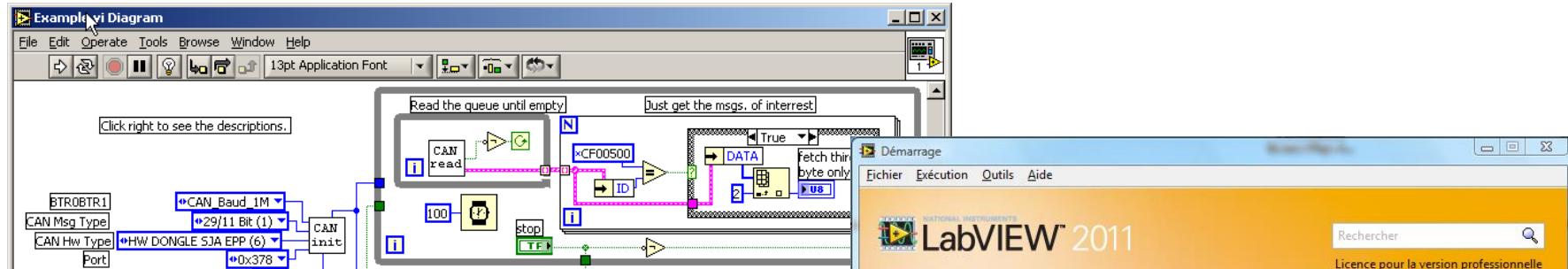




UFR
Sciences
et
Technologies

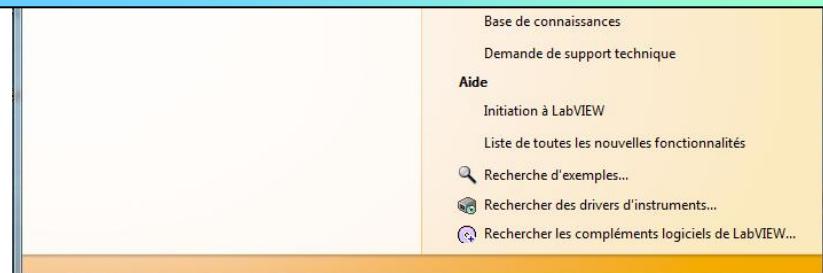
D φ
Département
de Physique

UNIVERSITÉ
BLAISE PASCAL
CLERMONT-FERRAND
UBP



Tutoriel LabVIEW

Des fonctions simples à l'acquisition de données...



Considérations générales sur LabVIEW

Qu'est-ce que LabVIEW ?

LabVIEW :

Laboratory of Virtual Instruments Engineering Workbench

- ⇒ Logiciel de développement d'applications, comparable à la plupart des systèmes de développement en langage C ou BASIC
- ⇒ Logiciel dont la philosophie d'utilisation repose sur la collaboration
 - ⇒ Communauté des développeurs

Considérations générales sur LabVIEW

Qu'est-ce que LabVIEW ?

- ⇒ système de programmation à usage général qui comporte des bibliothèques de fonctions pour toute tache de programmation.
- ⇒ bibliothèques dédiées à l'acquisition de données, le contrôle d'instruments, analyse, traitement et stockage de données
- ⇒ beaucoup de sous-programmes LabVIEW sont développés par les fabricants d'instruments de mesure

⇒ Multiples versions pour applications ciblées

| | |
|-----------|-----------------|
| FPGA | PDA |
| Real Time | Professionnelle |

Distinction des autres logiciels :
permet de faire de la **programmation graphique** !

Considérations générales sur LabVIEW

Le langage G

langage basé sur le principe du flot de données, auquel ont été rajoutées des structures de programmation afin d'obtenir un langage de programmation complet.

⇒ données transitant qu'au moment où elle sont générées par les icônes source.

flots de données



détermine l'ordre d'exécution des traitements du programme



traitements n'échangeant pas de données = libres de s'exécuter en parallèle

Considérations générales sur LabVIEW

Domaines d'application

LabVIEW permet de faire de la programmation graphique pour les mesures et l'automatisation

Domaines d'application traditionnels



Acquisition, traitement et commande à partir d'un PC

LabVIEW permet de développer toute une chaîne d'acquisition



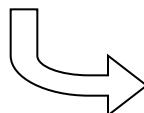
Considérations générales sur LabVIEW

Domaines d'application

H. Sklenarova, A. Svoboda, P. Solich, M. Polasek and R. Karliceck,

Simple laboratory-made automated sequential analysis (SIA) device: SIA operational software based on LABVIEW programming language,

Instrum Sci Technol 3 (2002), pp. 353–360.

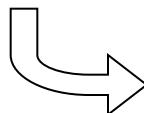


Développement de dispositifs de traitement de l'information

J. Ballesteros, M.A. Hernandez Palop, R.M. Crespo and S.B. Del Pino,

LabView virtual instrument for automatic plasma diagnostic,

Rev Sci Instrum 75 (2003), pp. 90–93.

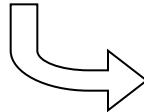


Analyse automatisée de plasma

Chung-hwan Je, Richard Stone, Steven G. Oberg

Development and application of a multi-channel monitoring system for near real-time VOC measurement in a hazardous waste management facility

Science of The Total Environment, Volume 382, Issues 2-3, 1 (2007) pp. 364-374



Réseau de capteurs pour la métrologie de COVs : nez électronique !

Présentation de l'interface

2 fenêtres de travail séparées mais associées !

1^{ère} fenêtre : face avant (*front panel*)

Contient les contrôles et les afficheurs de données

Interrupteurs, variateurs, potentiomètres avec curseurs, boites de dialogue

LEDs, compteurs à aiguilles, vu-mètres, boites de message, graphes



Représentations graphiques



2^{ème} fenêtre : diagramme (*block diagram*)

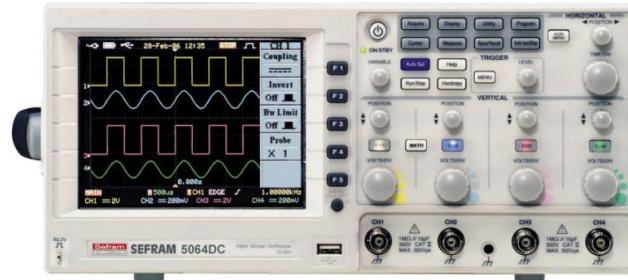
Contient le programme

Terminaux de commandes, nœuds (sous-programme, fonctions, structures, interfaces code), constantes,

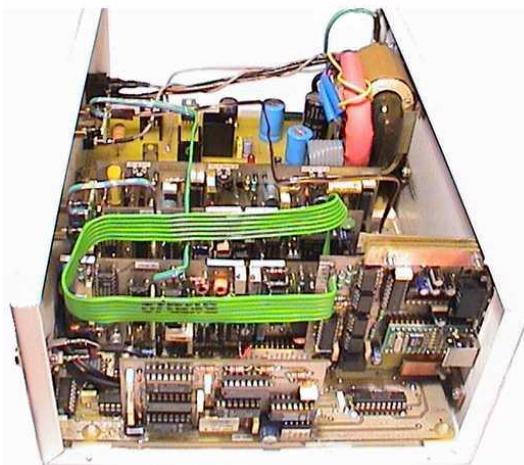
Terminaux d'affichage, des connecteurs, des fils

Présentation de l'interface

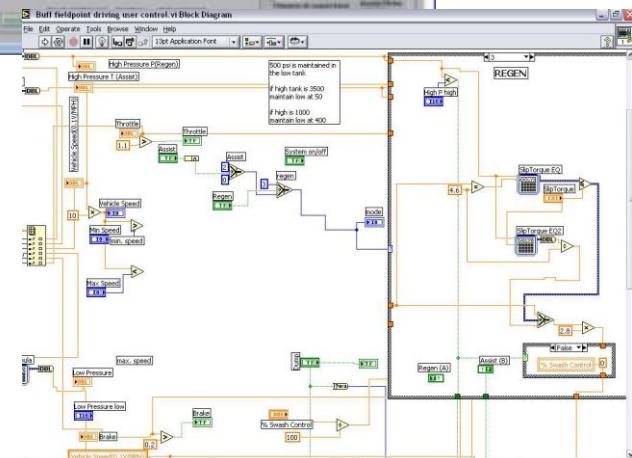
⇒ Développement d'instruments virtuels !



Front panel



Block diagram

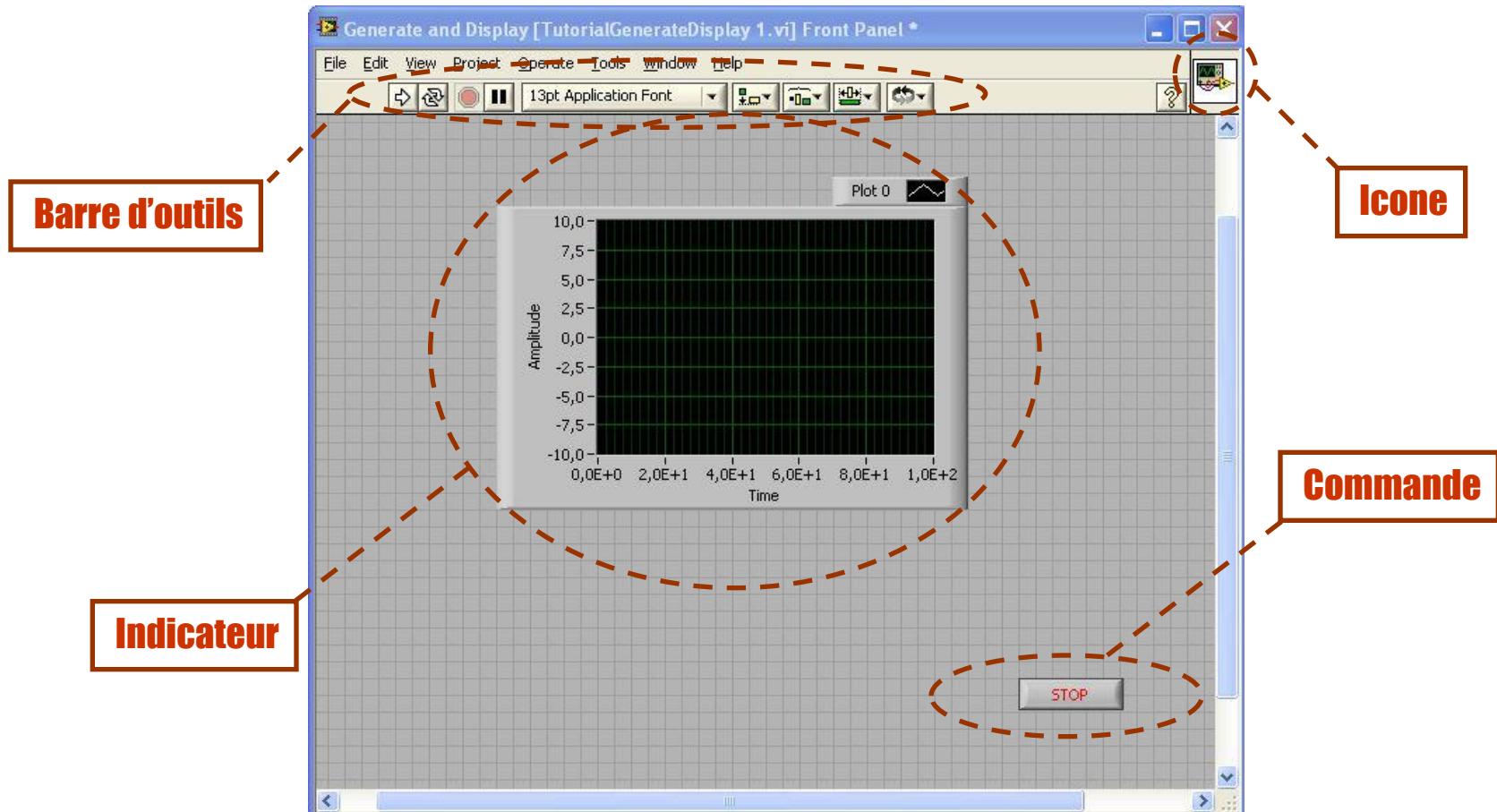


Présentation de l'interface

La face avant (*front panel*)



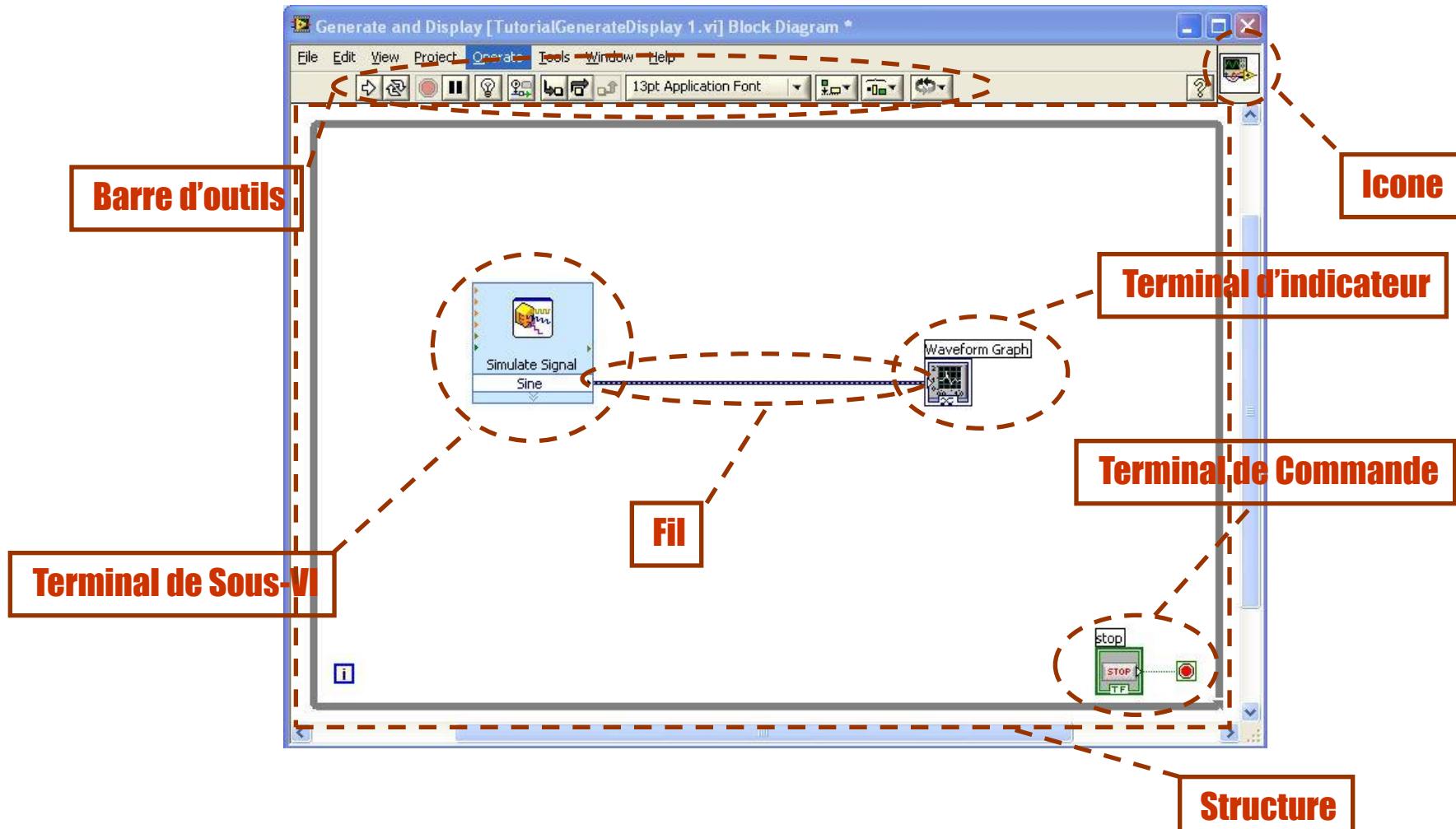
Interface utilisateur



Présentation de l'interface

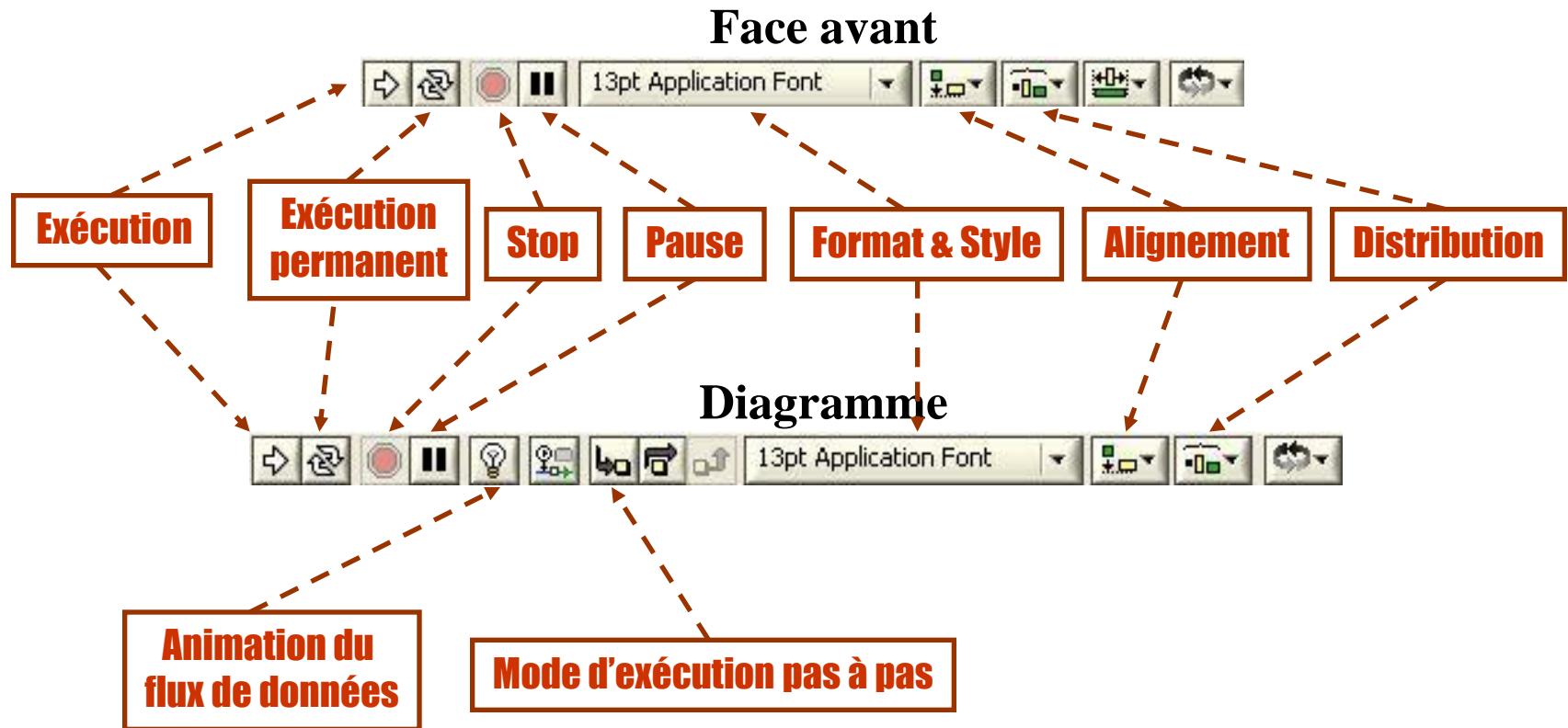
Le diagramme (*block diagram*) ⇒

Interface programmeur



Présentation de l'interface

Les barres d'outils

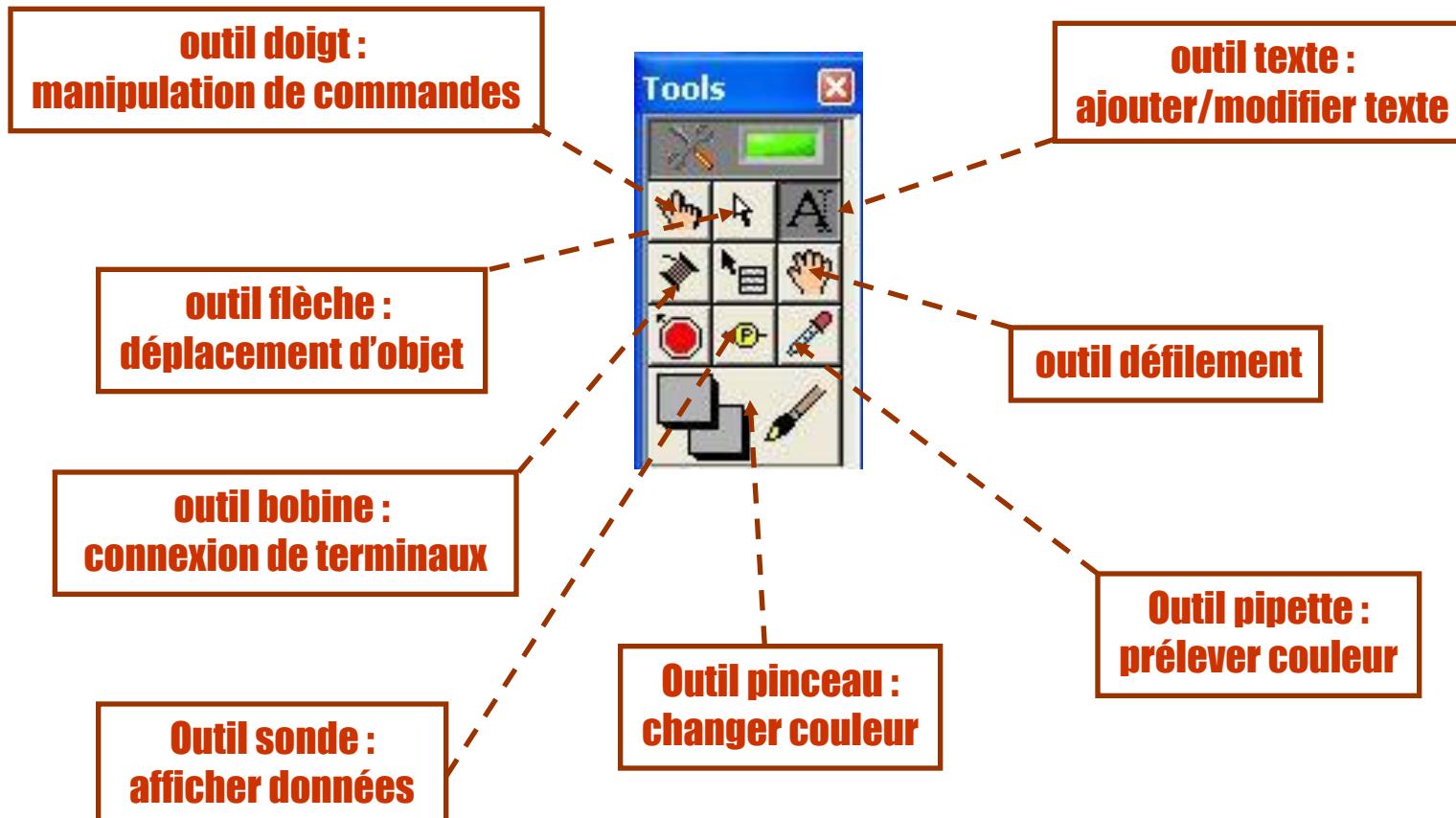


Présentation de l'interface

La palette Outils

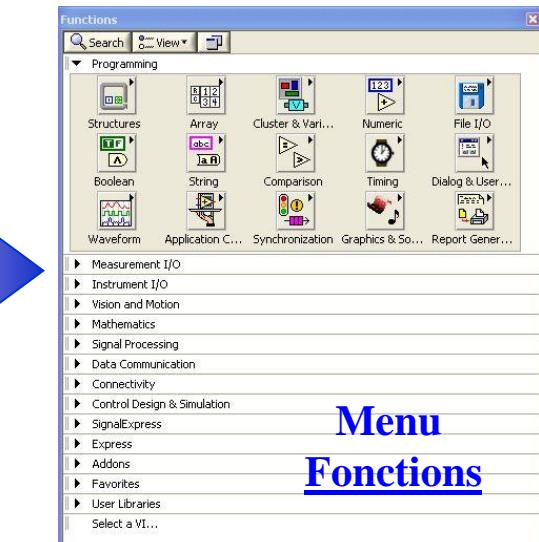
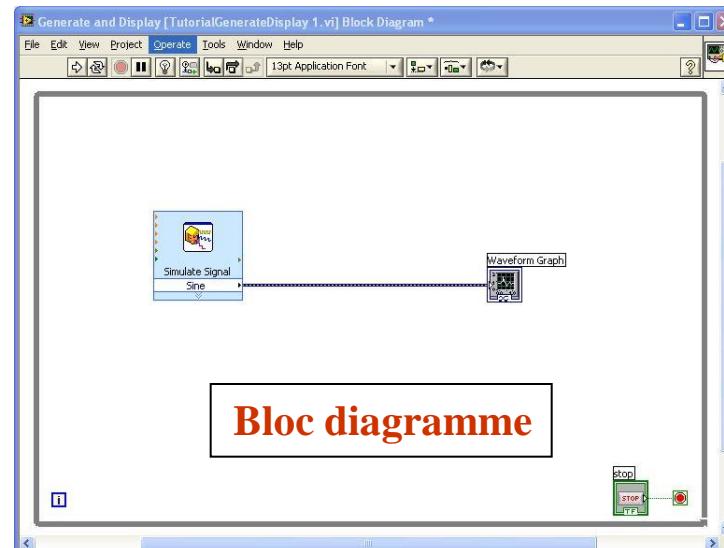
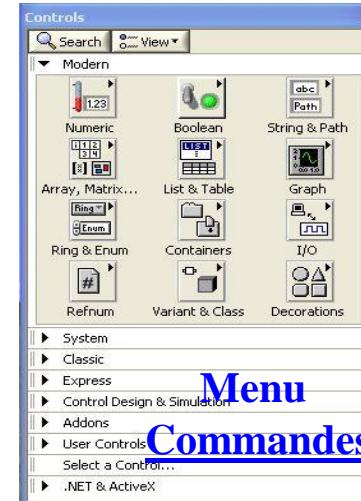
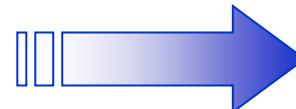
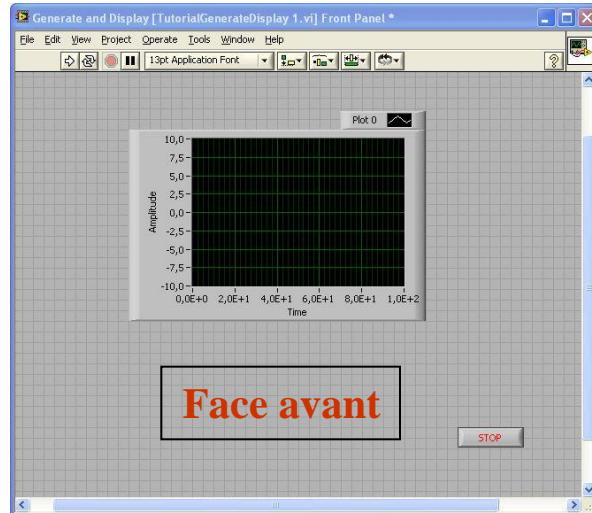
⇒ commune à la face avant et au bloc diagramme !

⇒ affichable à partir du menu *VIEW* de l'une ou l'autre fenêtre



Menus et sous-menus

Les menus *Commandes* et *Fonctions*



Menus et sous-menus

Commandes/indicateurs numériques

Le menu *Commande*

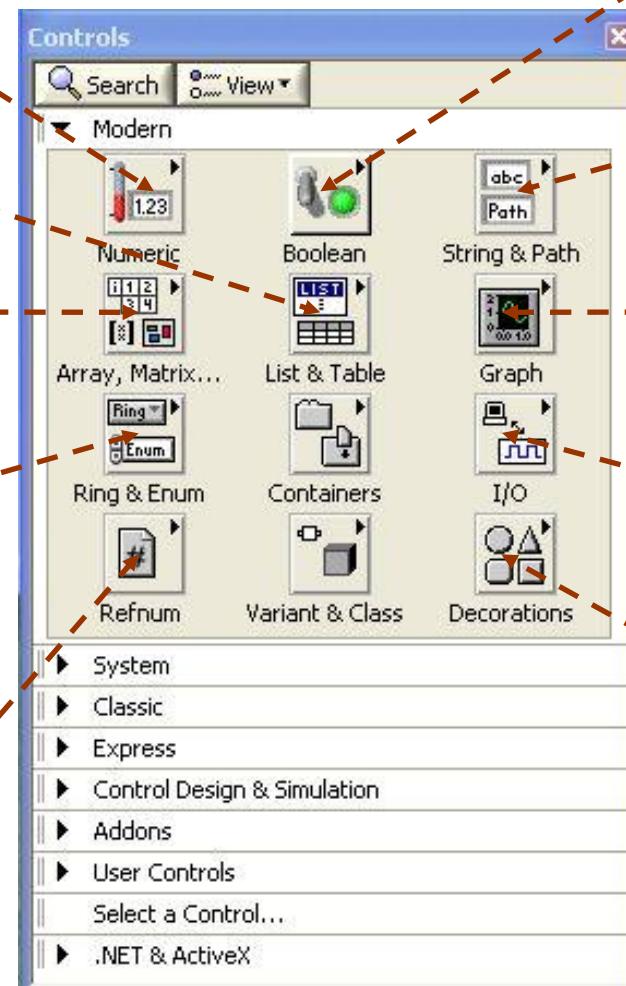
Commandes/Indicateurs booléens

Commandes/indicateurs tableaux

Commandes/indicateurs matrices, réseaux de données

Commandes menus déroulants

Commandes/indicateurs par n° de référence



Commandes/Indicateurs chaînes de caractères

Indicateurs écran (graphes temps réel)

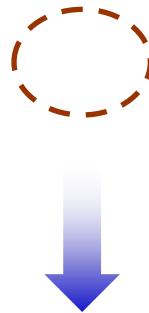
Commandes entrées/sorties

Eléments graphiques pour face avant

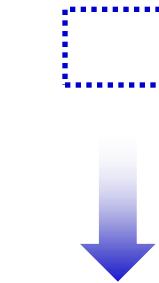
Menus et sous-menus

Exemple de sous-menu du menu *Commande...*

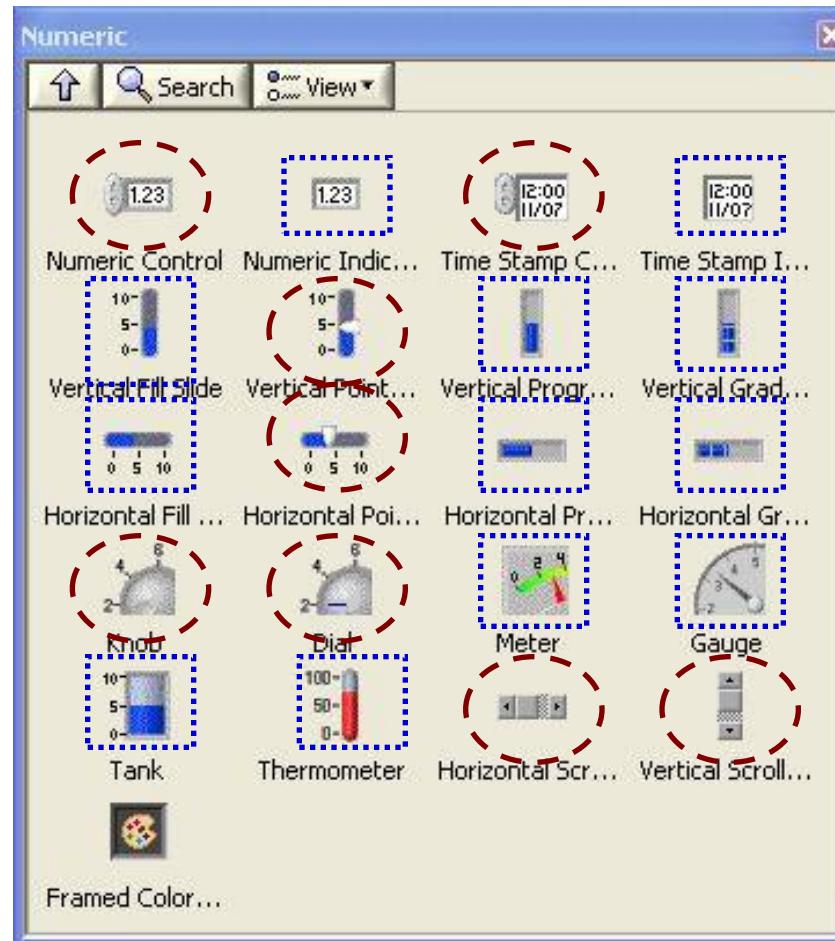
⇒ Bibliothèque Numérique ...



Commandes
Controls

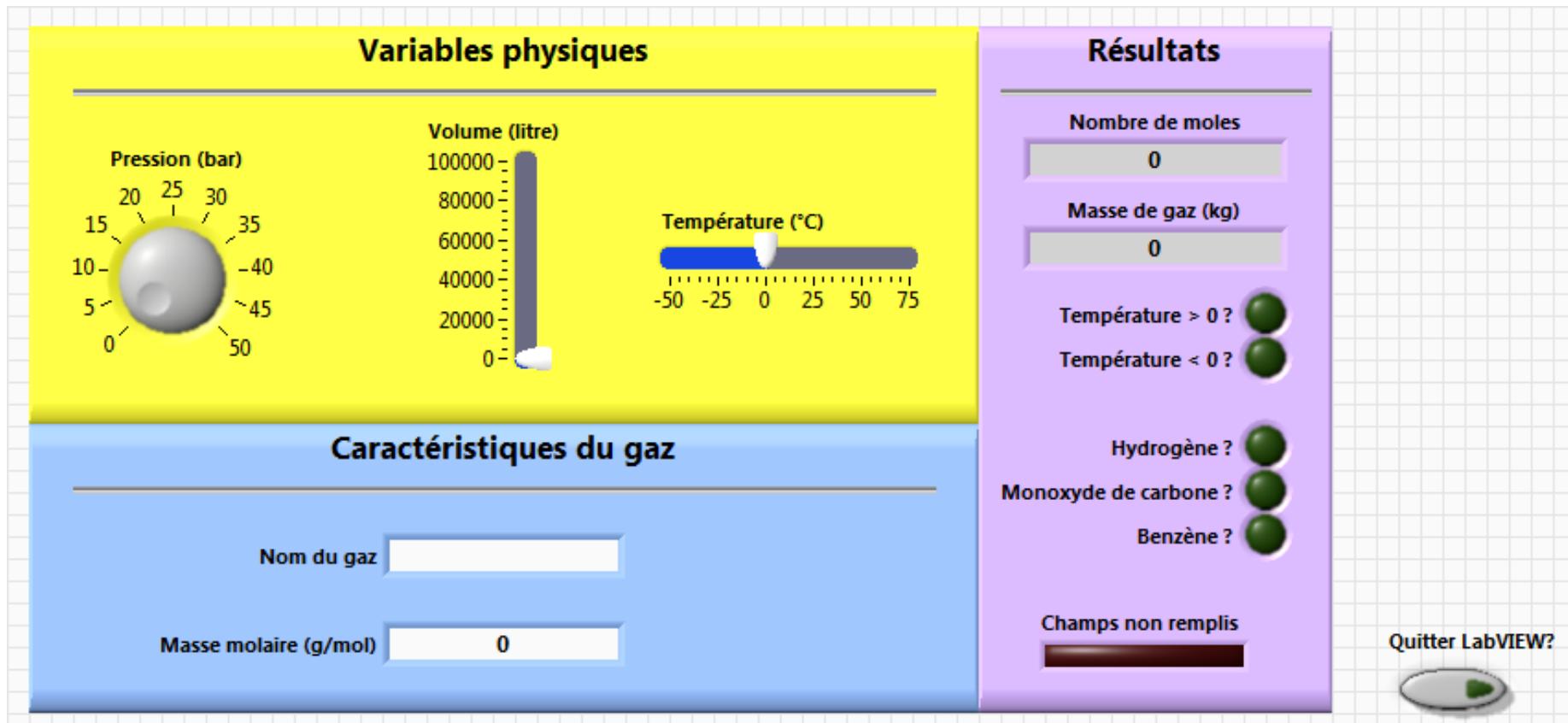


Indicateurs
Indicators



Menus et sous-menus

A l'aide des sous-menus du menu Commande, reproduire la face avant représentée ci-dessous:



Menus et sous-menus

Fonction sur tableaux

Boucles et structures

Fonctions chaînes

Fonctions booléennes

Fonctions
Graphes déroulants

Le menu *Fonctions*

Fonctions numériques

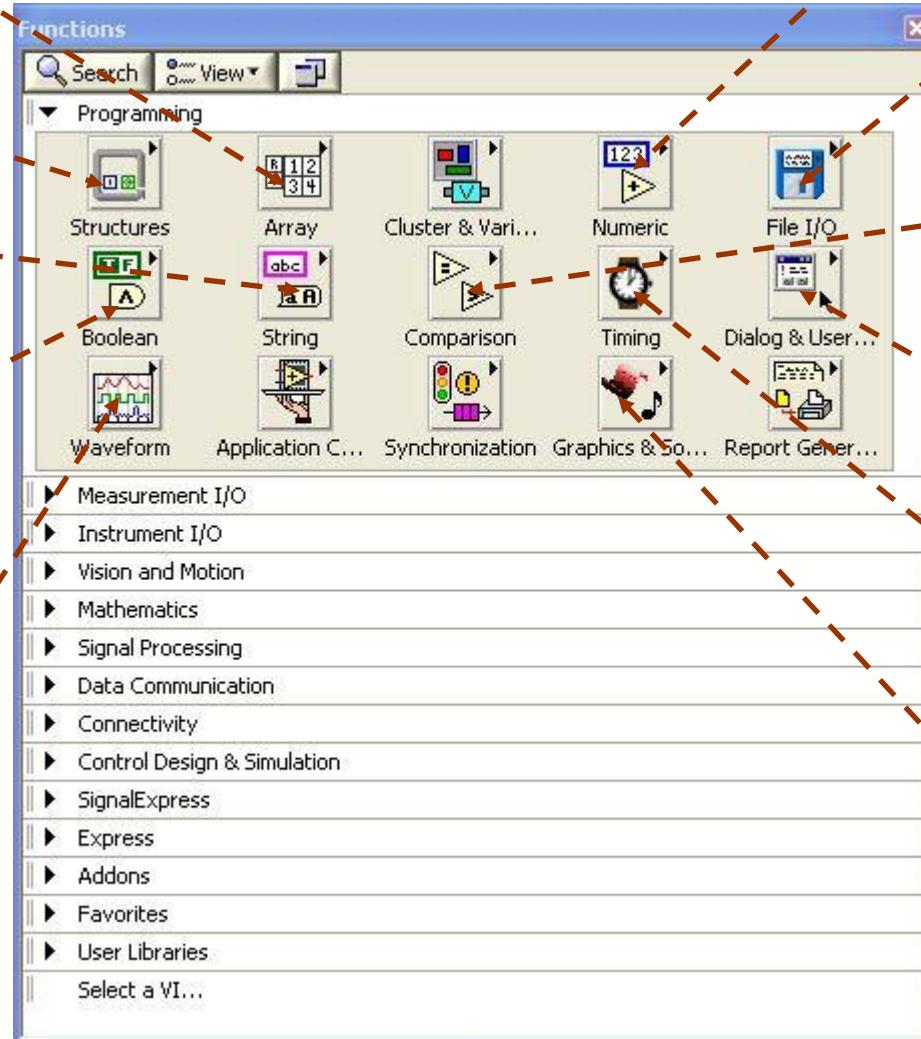
Fonctions sur fichiers

Fonctions comparatives

Fonctions boîtes de dialogue

Fonctions temporelles

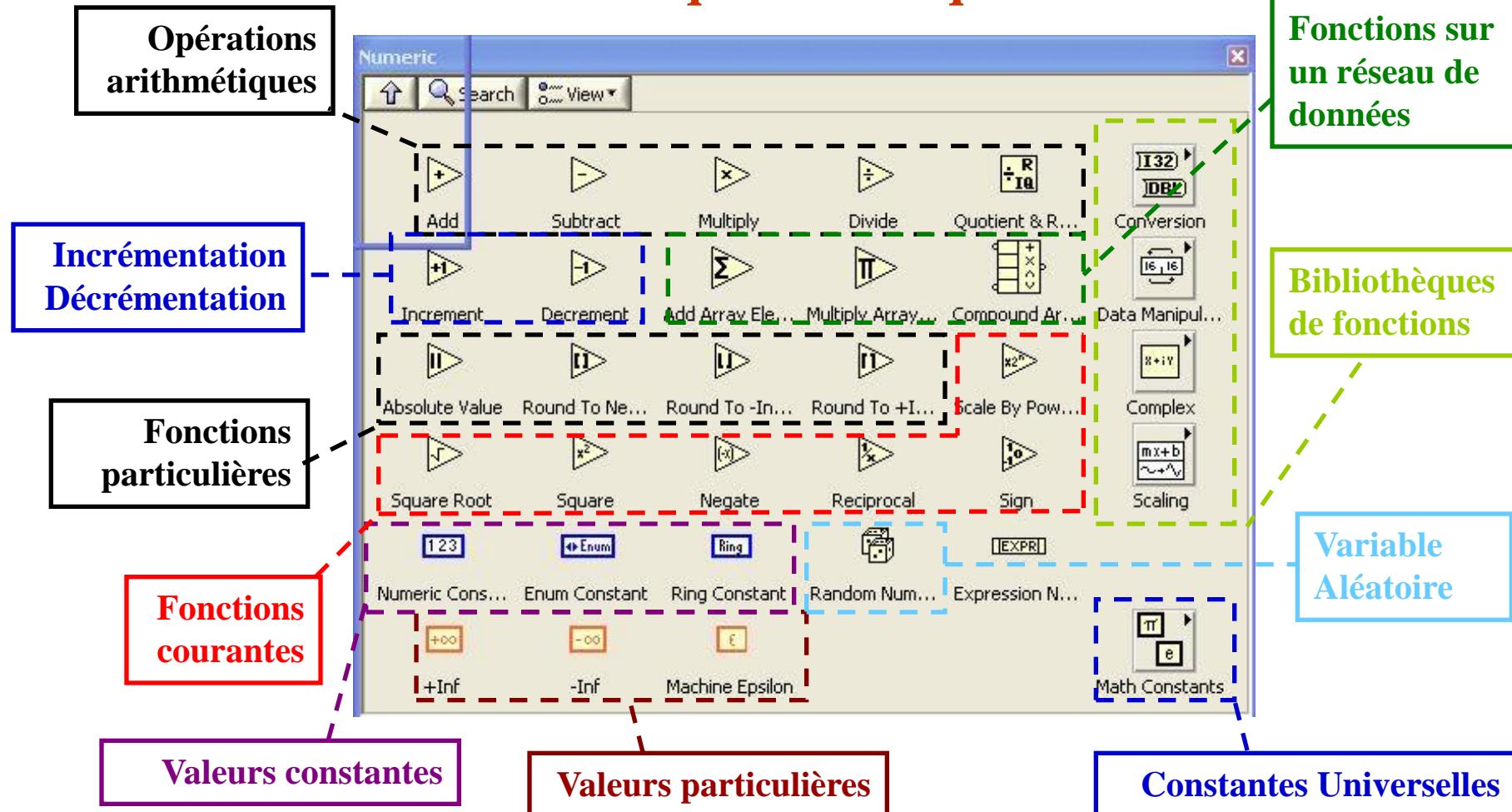
Fonctions sons et images



Menus et sous-menus

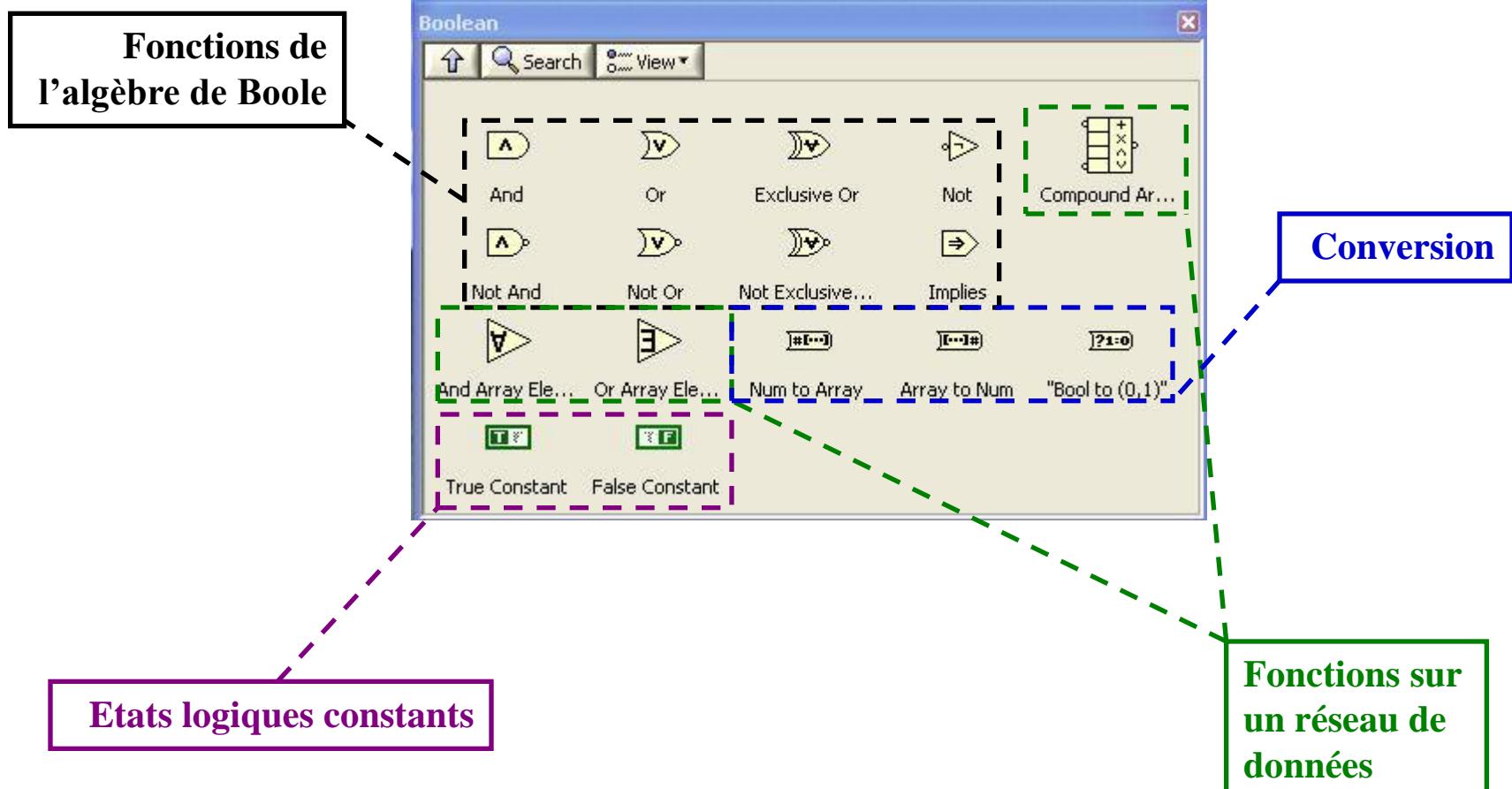
Exemple de sous-menu du menu *Fonctions...*

⇒ Bibliothèque numérique ...

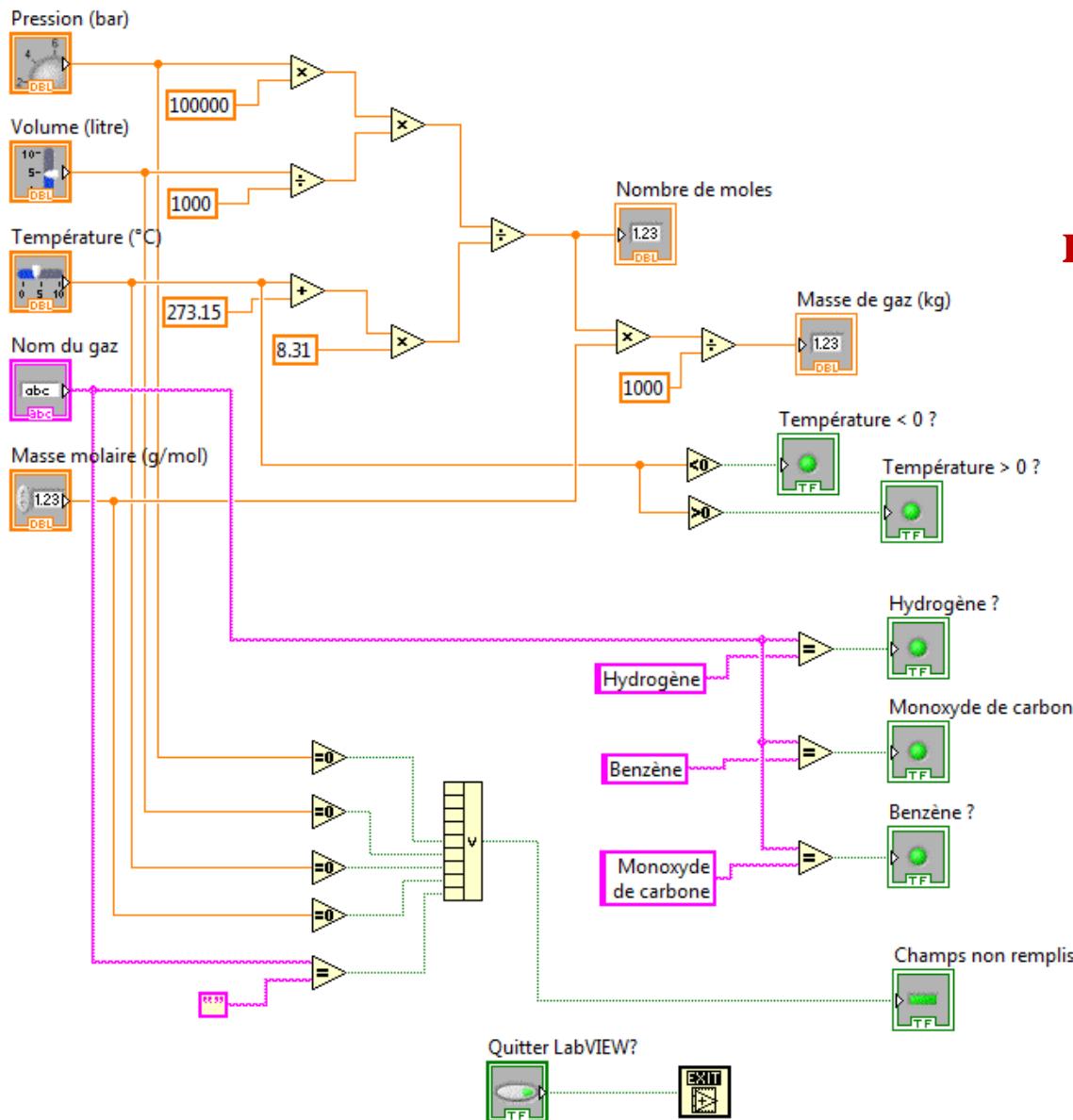


Menus et sous-menus

Exemple de sous-menu du menu *Fonctions...*
⇒ Bibliothèque booléenne ...



Menus et sous-menus



A l'aide des sous-menus
du menu Fonctions,
reproduire le diagramme
représenté ci-dessous

la face avant est celle
réalisée précédemment

Les variables d'entrée

Fils associées aux variables

⇒ Servent à l'acheminement des données entre terminaux vis les nœuds du V.I.

Circulation unidirectionnelle : Commande ⇒ Indicateur

| Type de variable | Représentation | Code couleur |
|---------------------|----------------|--------------|
| Booléenne | | Vert |
| Chaîne de caractère | | Violet |
| Nombre réel | | Orange |
| Nombre entier | | Bleu |

Épaisseur du fil ⇒ nature des données (scalaire, tableau 1D, tableau 2D)

Technique de développement

Flèche brisée dans la barre d'outils = exécution impossible !

Recherche d'erreur ⇒ double-clic sur la flèche brisée !



Mise au point d'un diagramme ⇒ mode pas à pas !

Exécute le diagramme nœud par nœud !



Visualisation séquentielle du déroulement du diagramme ⇒ mode animation !

Permet de suivre à vitesse lente le flux de données au sein du diagramme !



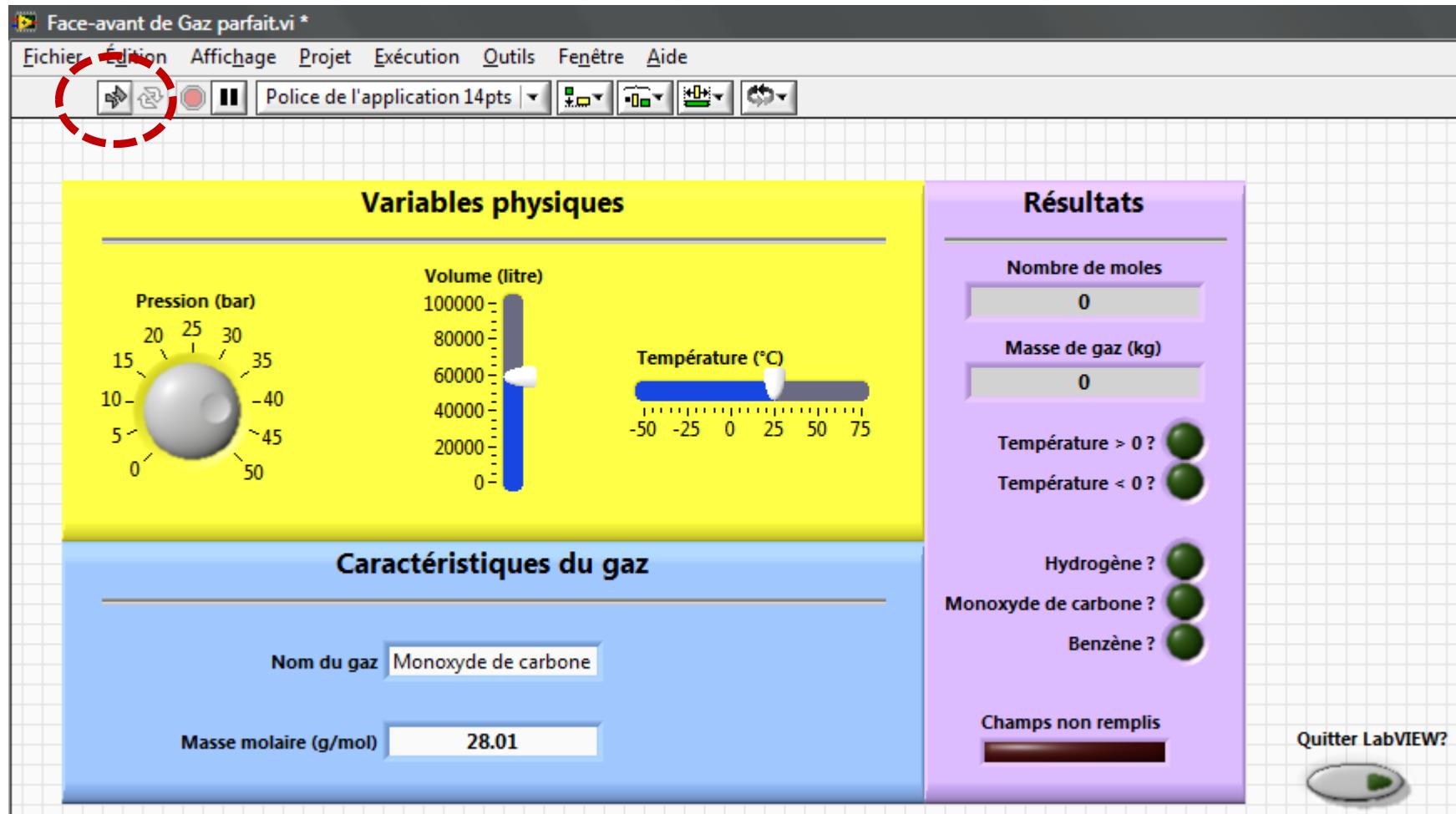
Mise en place d'indicateur locaux ⇒ mode sonde !

Permet de visualiser des données dans un fil au fur et à mesure de leurs acheminement !



Technique de développement

Exécuter le programme, corriger les erreurs si nécessaire et analyser les résultats:



Notions de VI et de sous-VI

Qu'est-ce qu'un VI ?

V.I. : Virtual Instrument

V.I. = programme LabVIEW qui comprend :

- l'interface utilisateur interactive (face avant)
- le diagramme de flux de données (code source)

Le V.I. est construit en langage G !

V.I. ⇒ présente une structure hiérarchique et modulaire

1 V.I. = 1 tache d'exécution !

Notions de VI et de sous-VI

Qu'est-ce qu'un sous-VI ?

V.I. contenu dans un autre V.I. = sous-V.I.

Sous-V.I.s = V.I.s de plus bas niveaux ou sous-programme

V.I. principal = ensemble de sous-V.I. représentant une tâche spécifique

Avantage : modularité de programmation !

Chaque sous-V.I. peut fonctionner indépendamment du reste de l'application

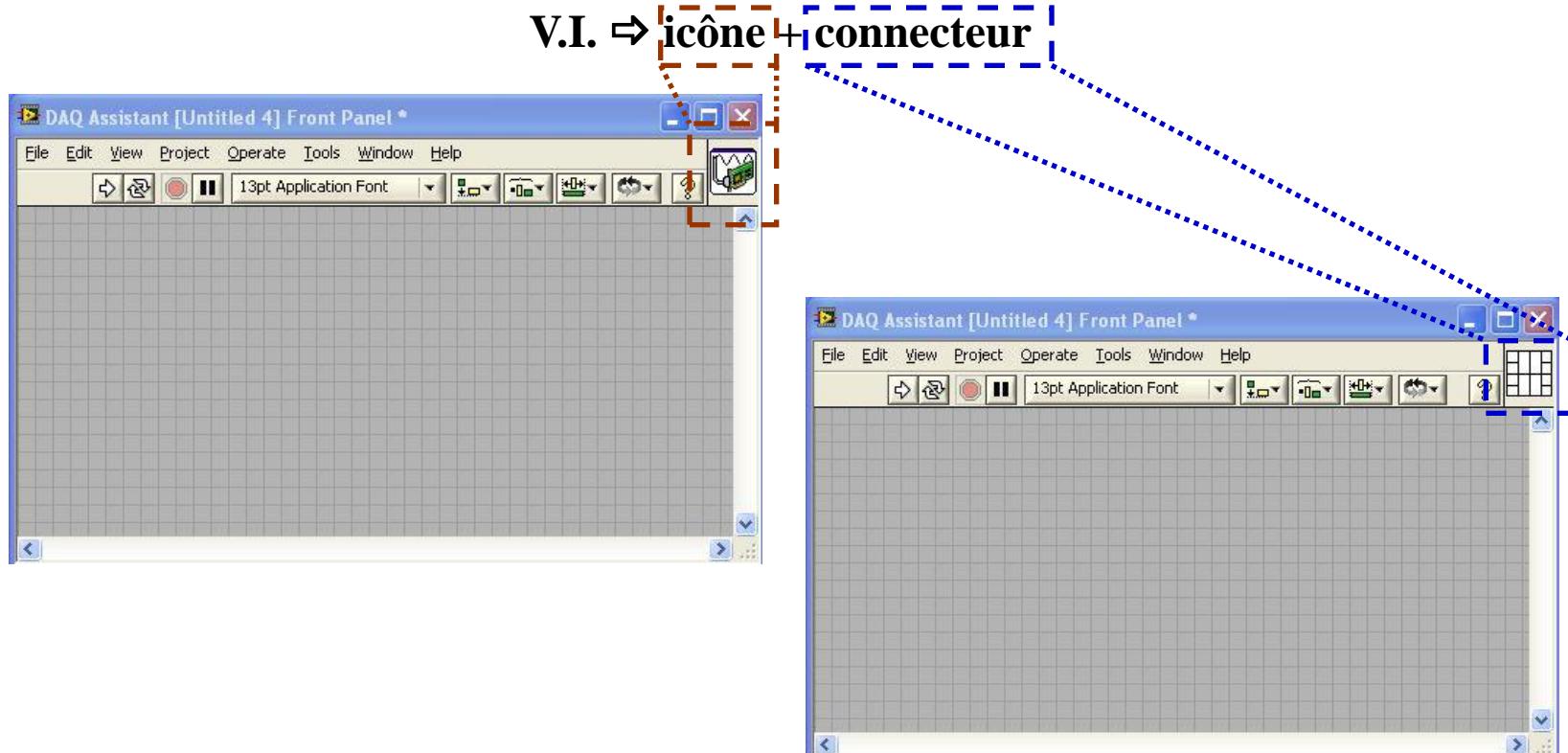
Sous-V.I. = V.I. de bas niveaux, que l'on peut enregistrer dans des bibliothèques spécifiques

Notions de VI et de sous-VI

Qu'est-ce qu'un sous-VI ?

Nœud du sous-V.I. = appel du sous-programme.

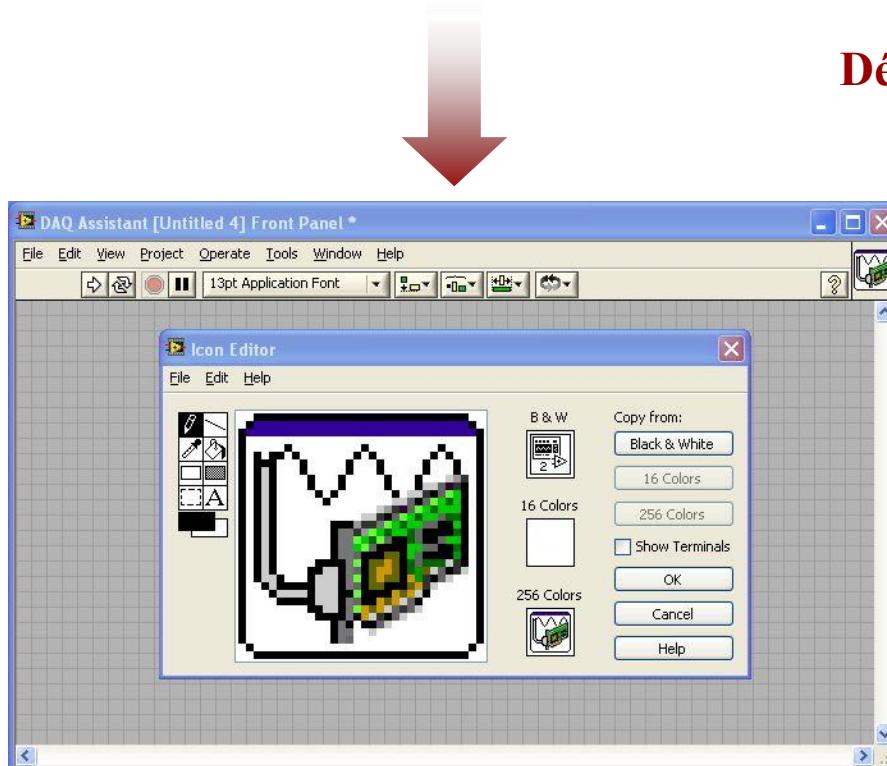
Développement du sous-V.I. : double-clic sur le nœud



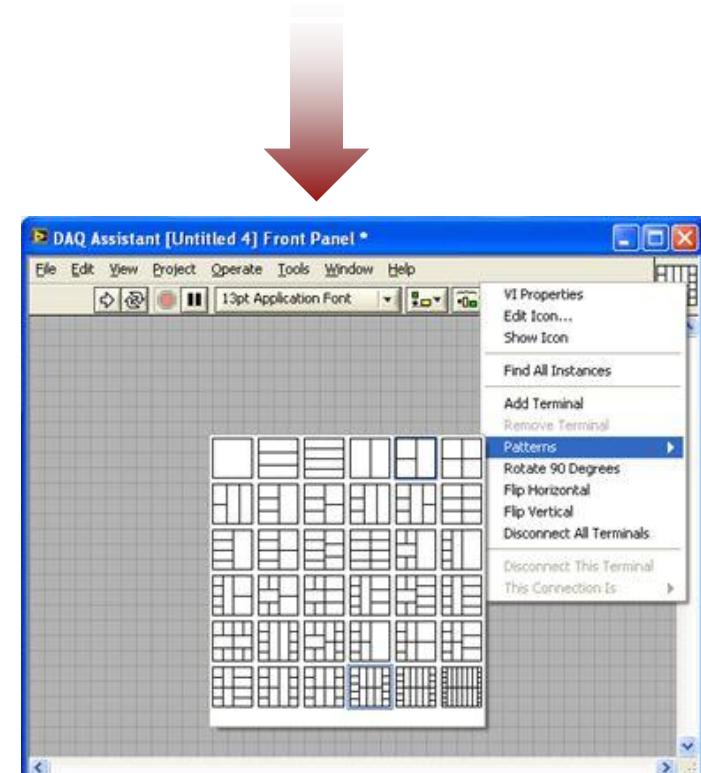
Notions de VI et de sous-VI

Configuration du sous-V.I.

Personnalisation graphique de l'icône



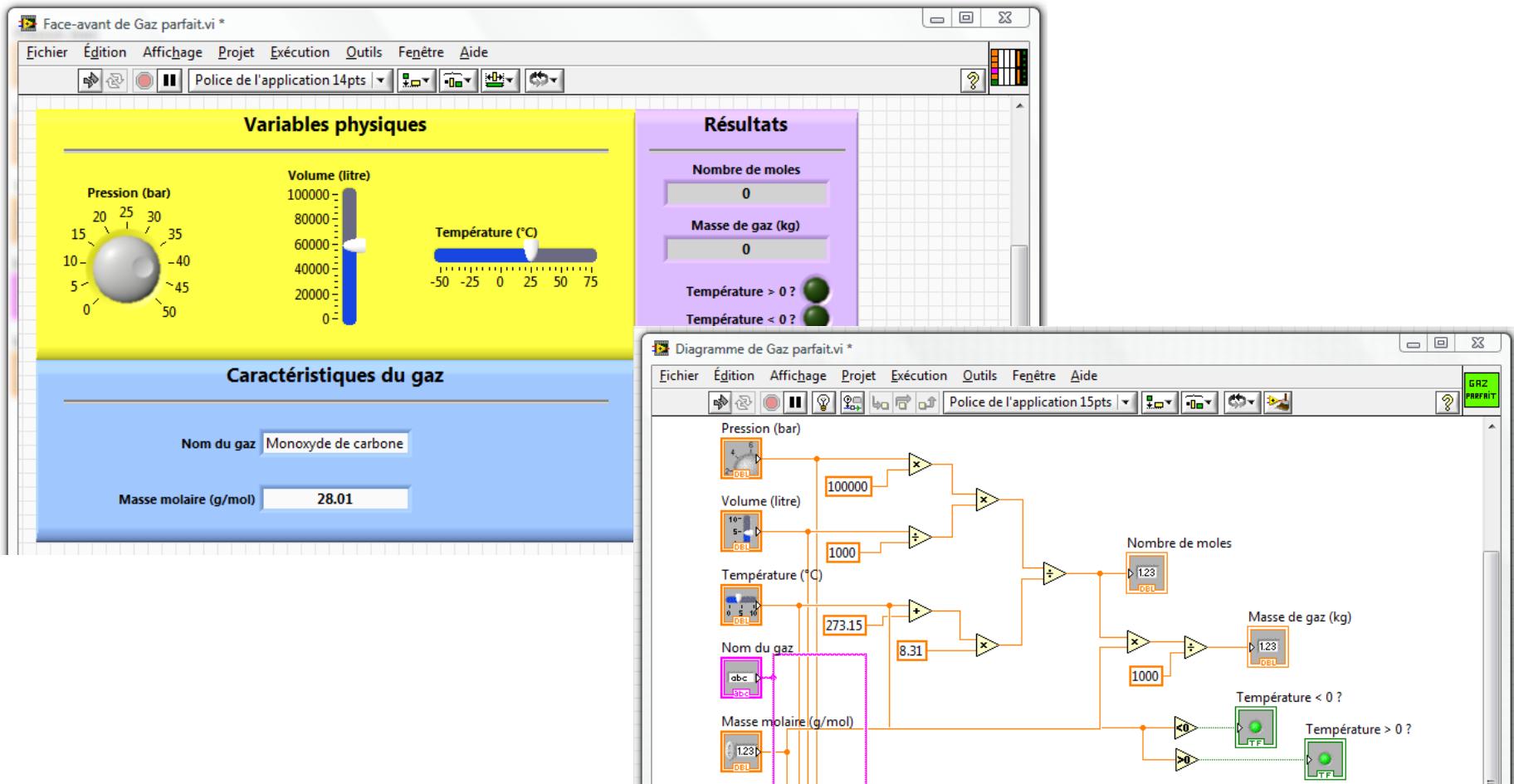
Interface de liaison
commandes / indicateur
Définis le nombre d'entrées et de sorties



⇒ Enregistrement des sous-V.I. dans les
librairies *User Libraries*

Notions de VI et de sous-VI

Configurer le VI précédemment réalisé pour pouvoir le réutiliser comme sous-VI:



Boucles et structures

Concept de boucles et de structures

Boucles ⇒ permettent l'exécution d'un programme, d'un sous-programme ou d'une partie de programme jusqu'à une action ou une valeur définie par l'opérateur.

Structures ⇒ contrôlent le flux de données dans un V.I.

Structures ⇒ permettent d'organiser, de séquencer ou de conditionner des éléments d'un V.I. ou un V.I. complet

Boucles ⇒ identique à celles mises en jeu en programmation en langage C ou en langage BASIC

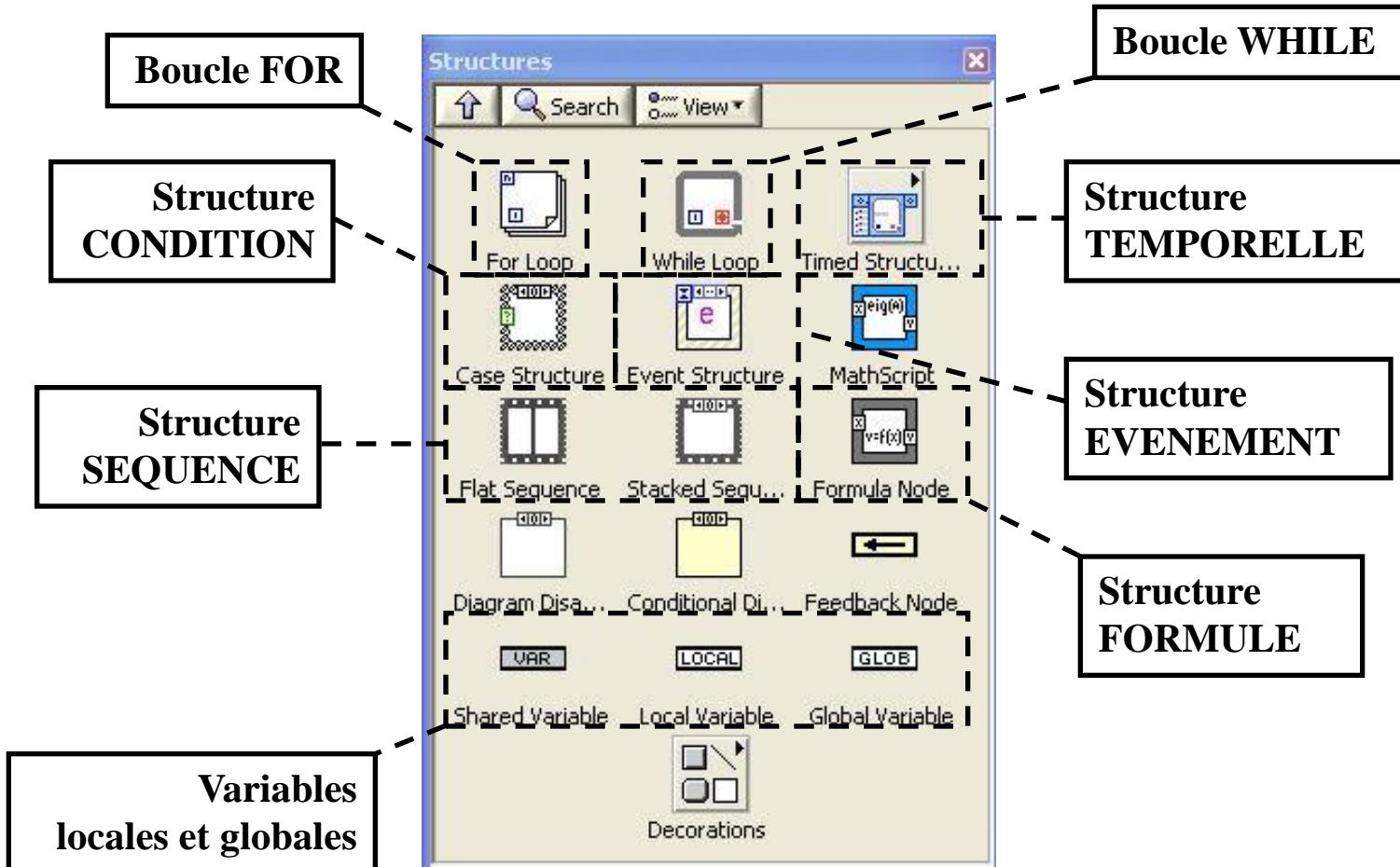
Structures

⇒ conditions = if, then, else...

⇒ temporelle = déroulement du programme (pas de flux de données)...

Boucles et structures

⇒ Bibliothèque boucles et structures ...



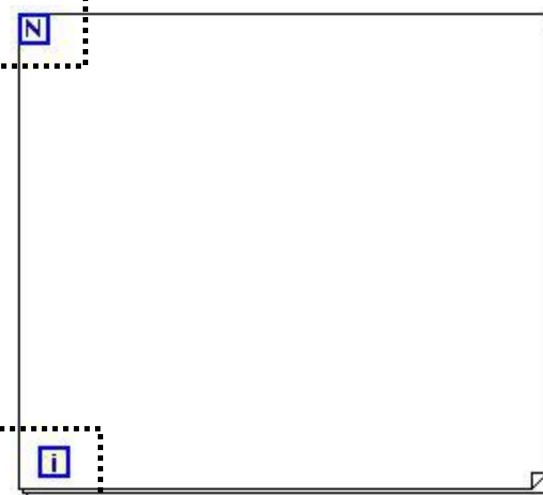
Boucles et structures

Boucle FOR

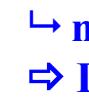
⇒ Répète une partie du code diagramme un nombre déterminé de fois, ce nombre étant définissable par l'opérateur

Terminal de comptage

⇒ définit par 2 terminaux



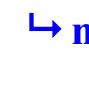
Entrée numérique



↪ nbre total d'exécution
⇒ Défini par l'utilisateur



Sortie numérique



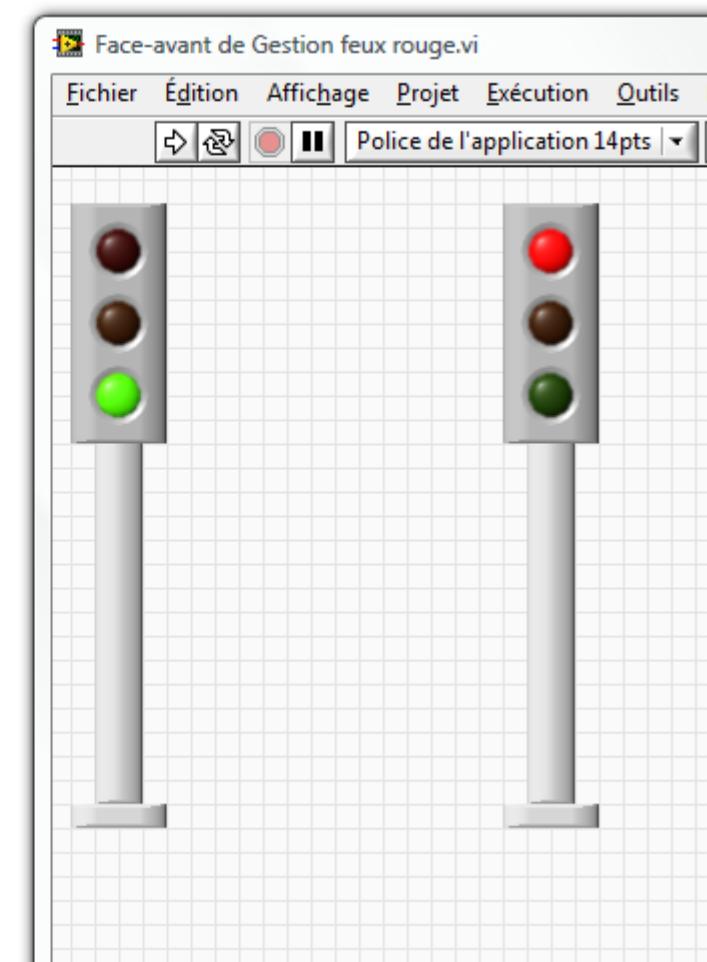
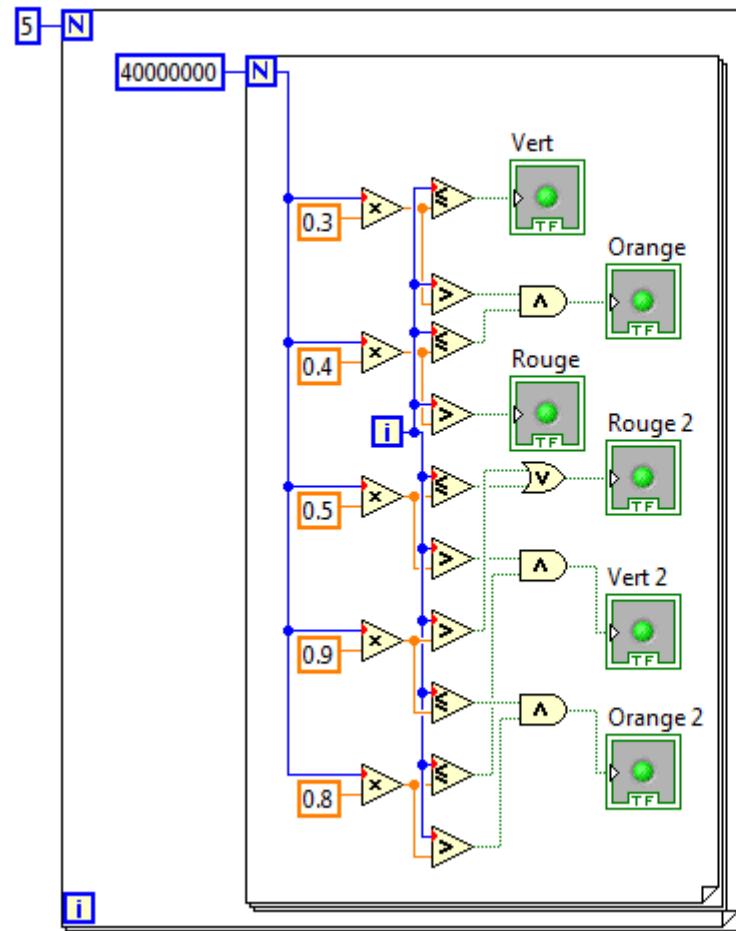
↪ nbre d'itération effectuée

⇒ Boucle comparable à la boucle FOR en programmation en langage C ou BASIC

Boucles et structures

Application d'une boucle FOR:

Reproduire le VI ci-dessous, identifier les fonctions et observer les résultats

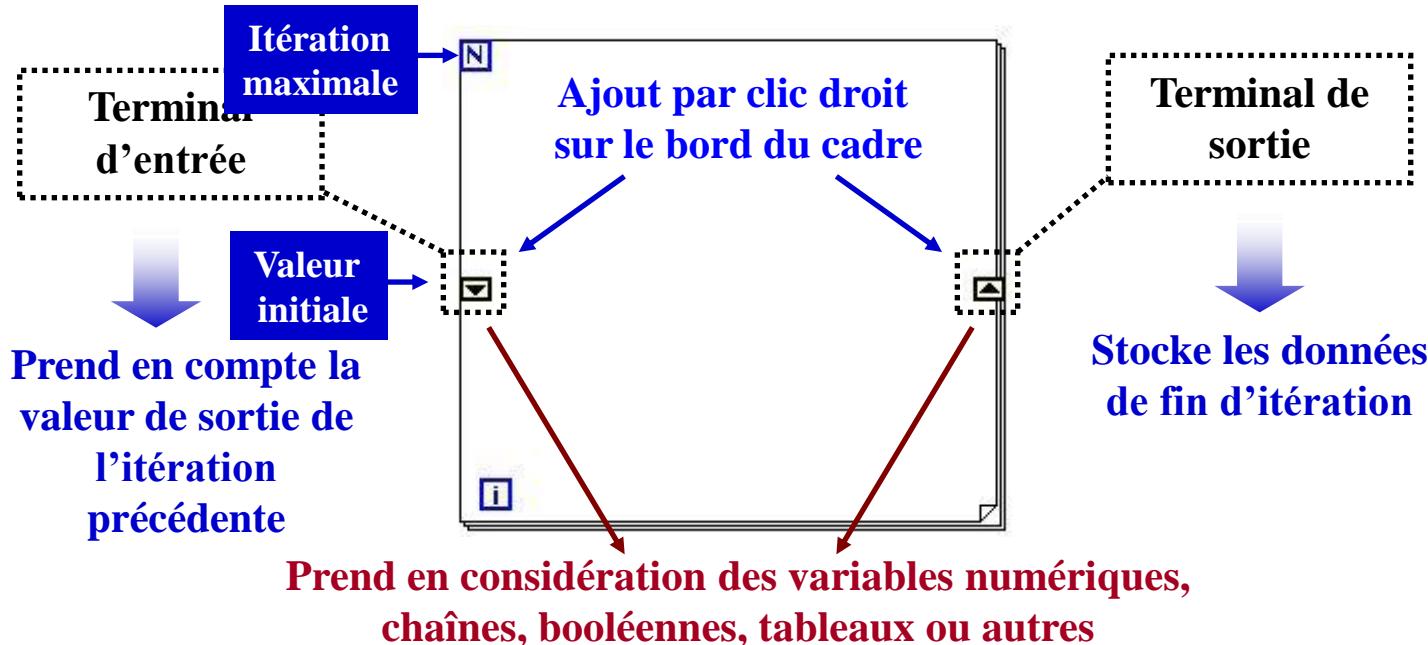


Boucles et structures

Structure Registre à décalage

⇒ Variables locales qui transfèrent des données entre deux itérations consécutives d'une même boucle

⇒ 2 terminaux placés au bord droit et gauche de la boucle

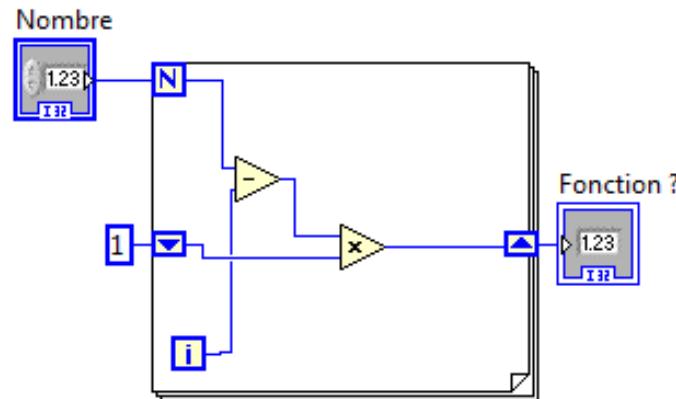


Application représentative

⇒ calcul d'une moyenne flottante

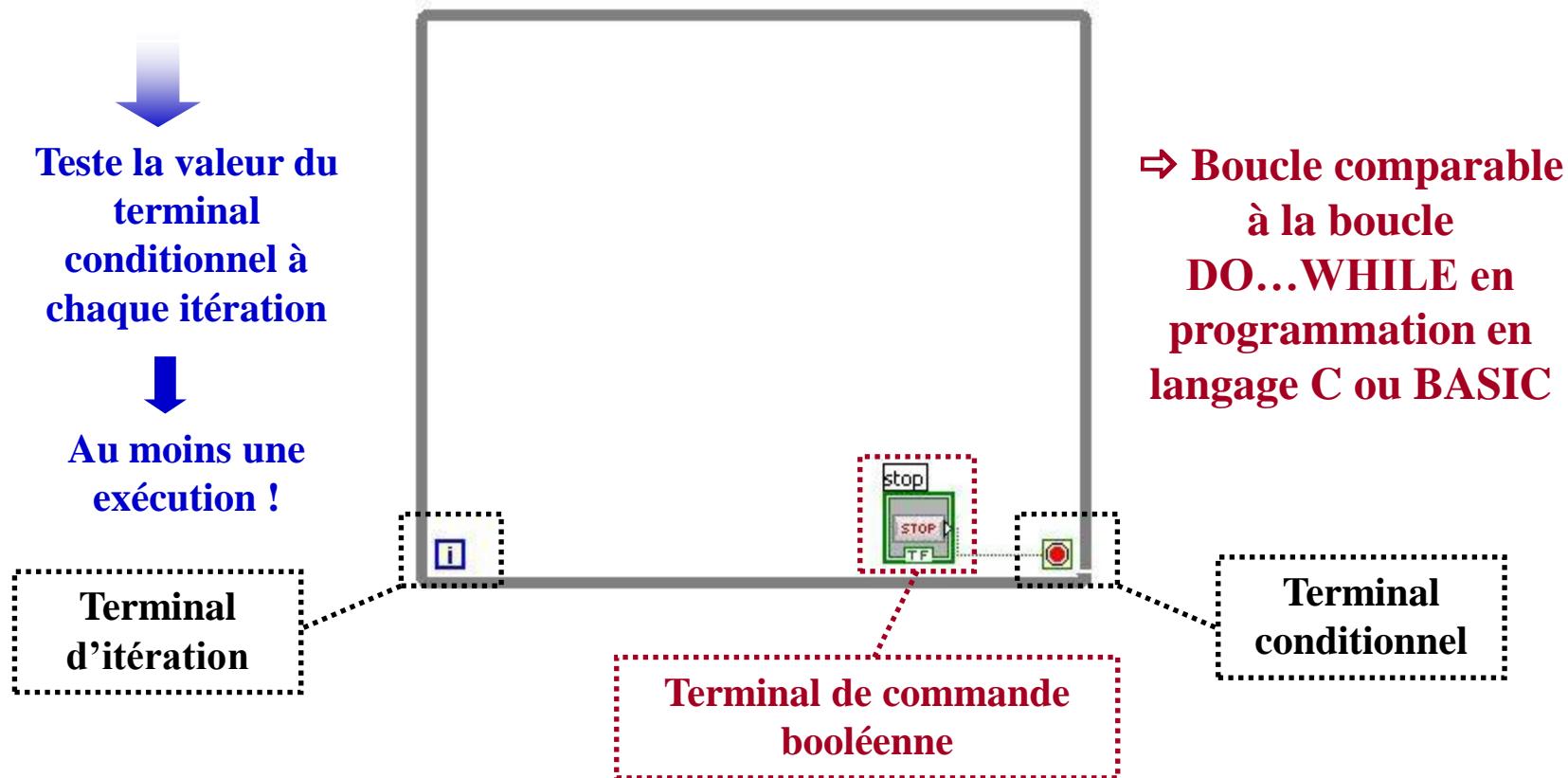
Application d'un registre à décalage

Quelle est la fonction réalisée par un VI dont le diagramme est représenté ci-dessous?



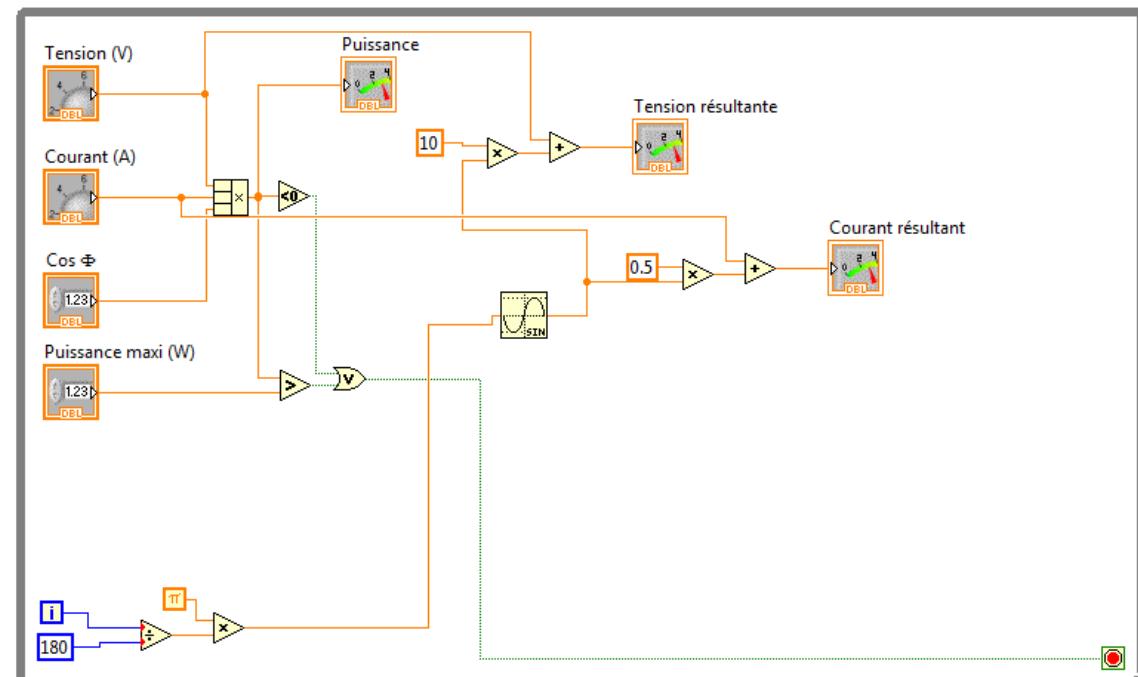
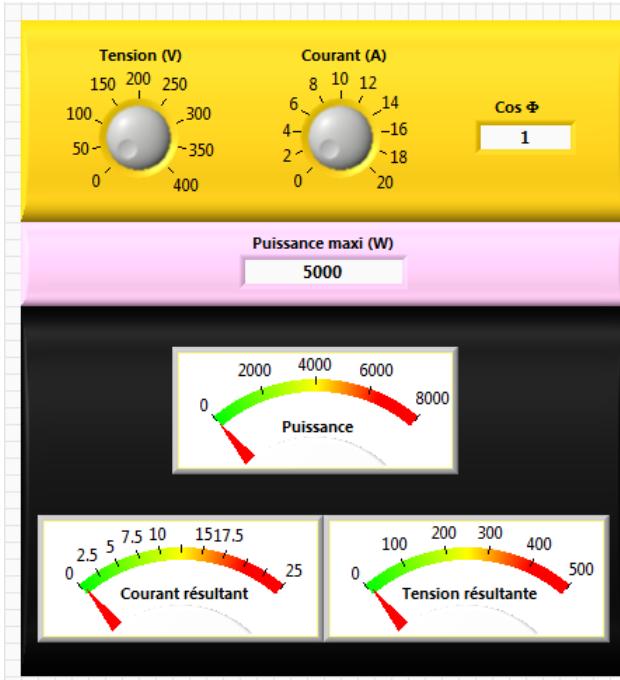
Boucle WHILE

⇒ Répète le code diagramme contenu à l'intérieur de la boucle jusqu'à un changement d'état de la variable booléenne associé au terminal conditionnel



Boucles et structures

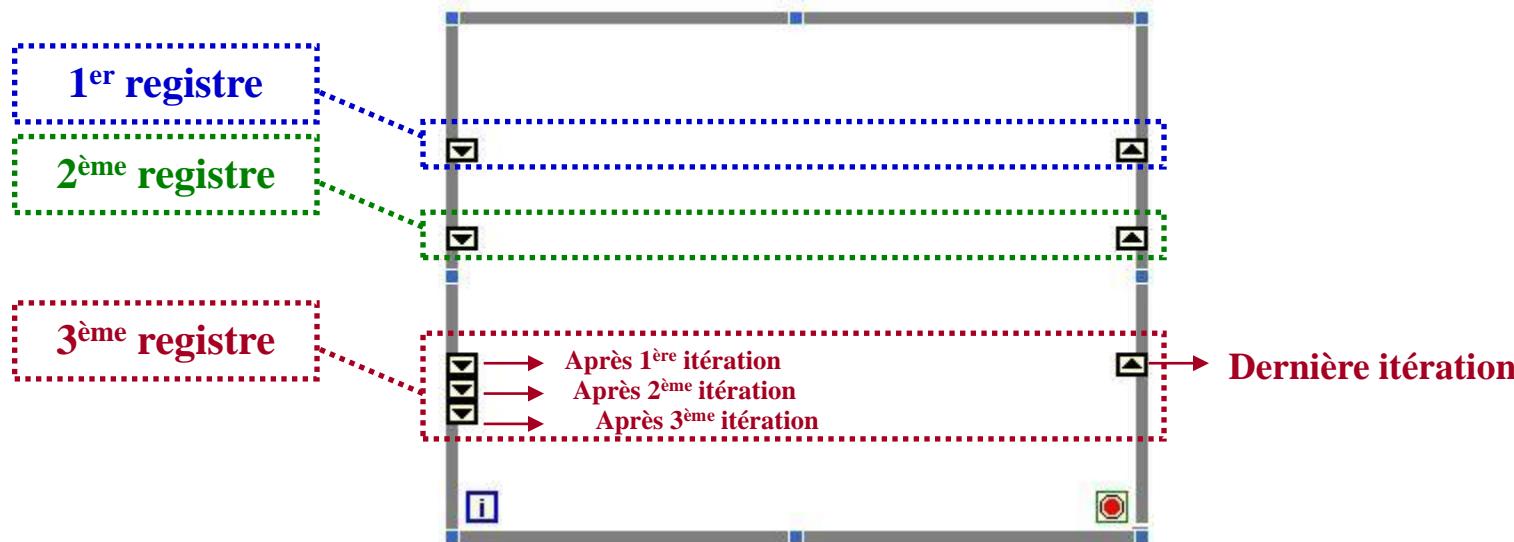
Application d'une boucle WHILE:



Structure Registre à décalage

⇒ Structure disponible également sur les boucles WHILE

⇒ terminaux ajoutés aux bords latéraux de la boucle



Valeur initiale du registre par défaut

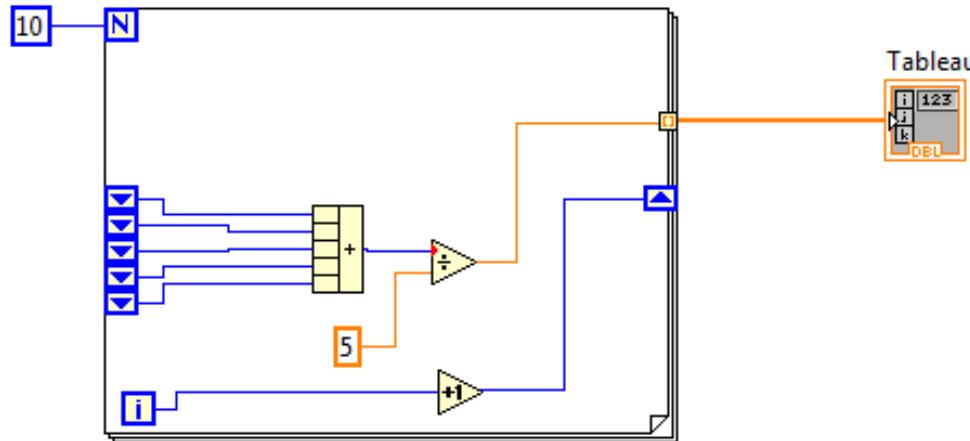
⇒ 0 en numérique

⇒ FAUX en booléen

Boucles et structures

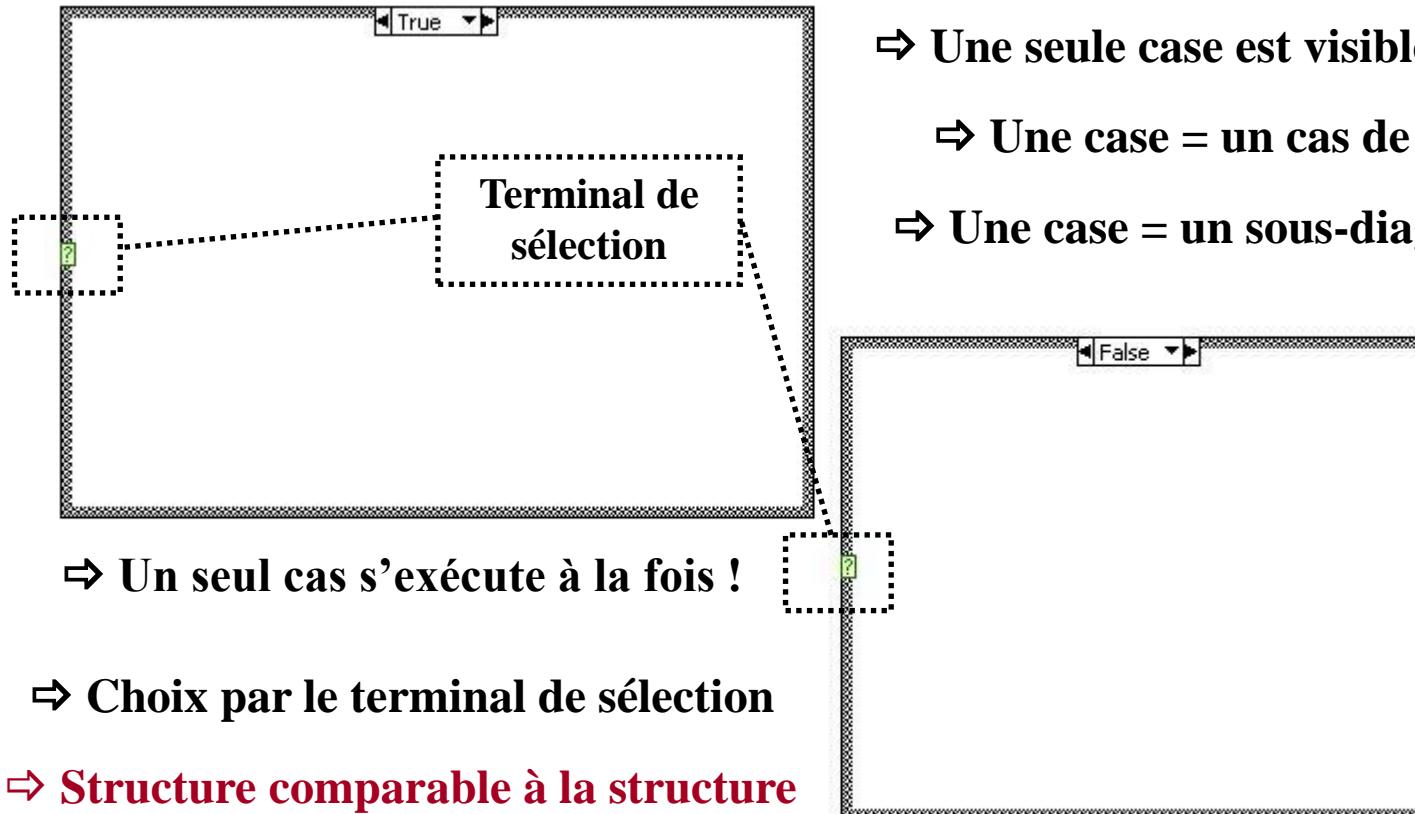
Application d'un registre à décalage à entrées multiples:

Quelle est la fonction réalisée par un VI dont le diagramme est représenté ci-dessous?



Structure CONDITION

⇒ Structure organisée sous forme de fenêtres associées



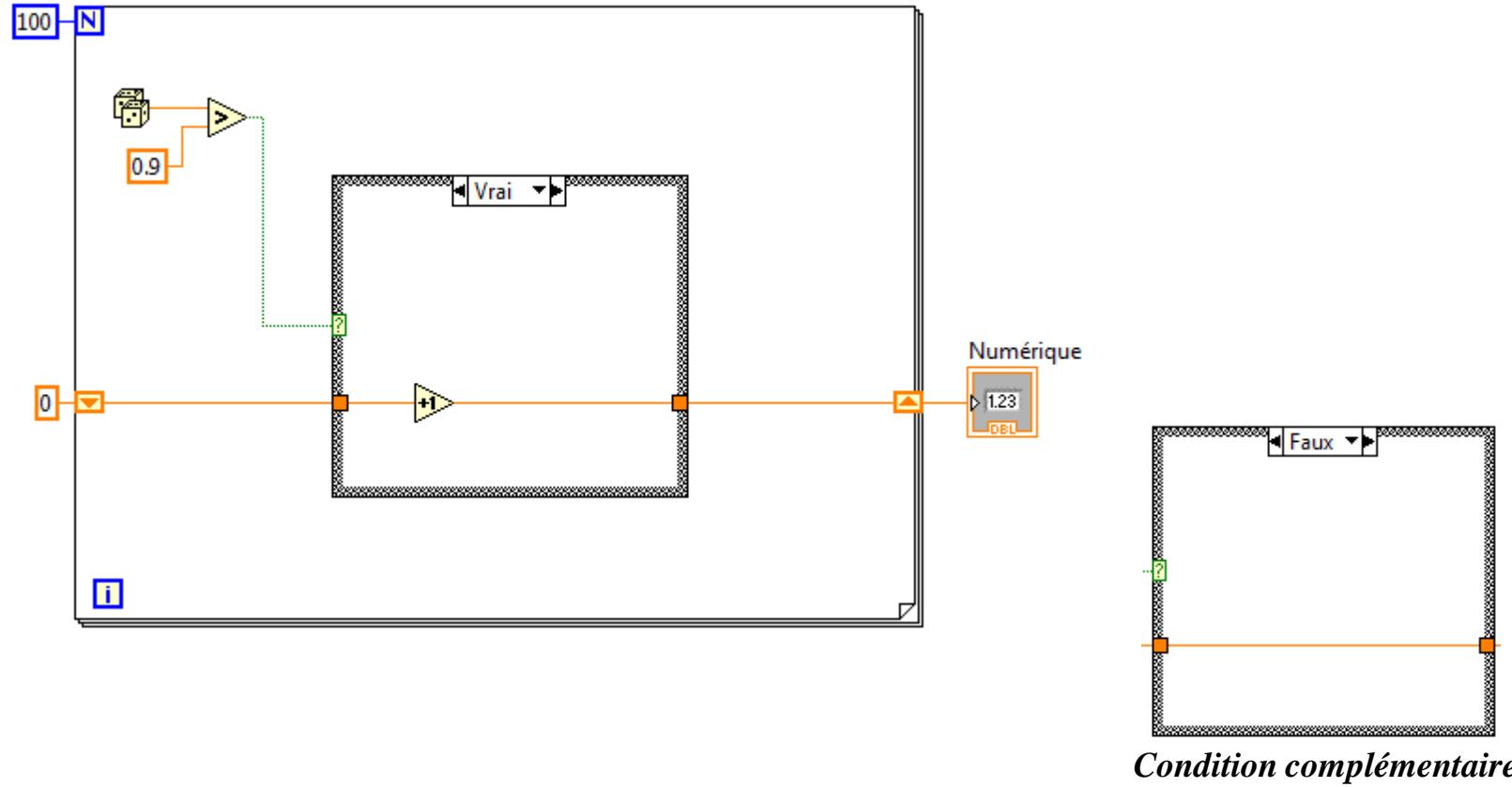
en programmation en langage C ou BASIC

- ⇒ Une seule case est visible à la fois !
- ⇒ Une case = un cas de figure !
- ⇒ Une case = un sous-diagramme !

Boucles et structures

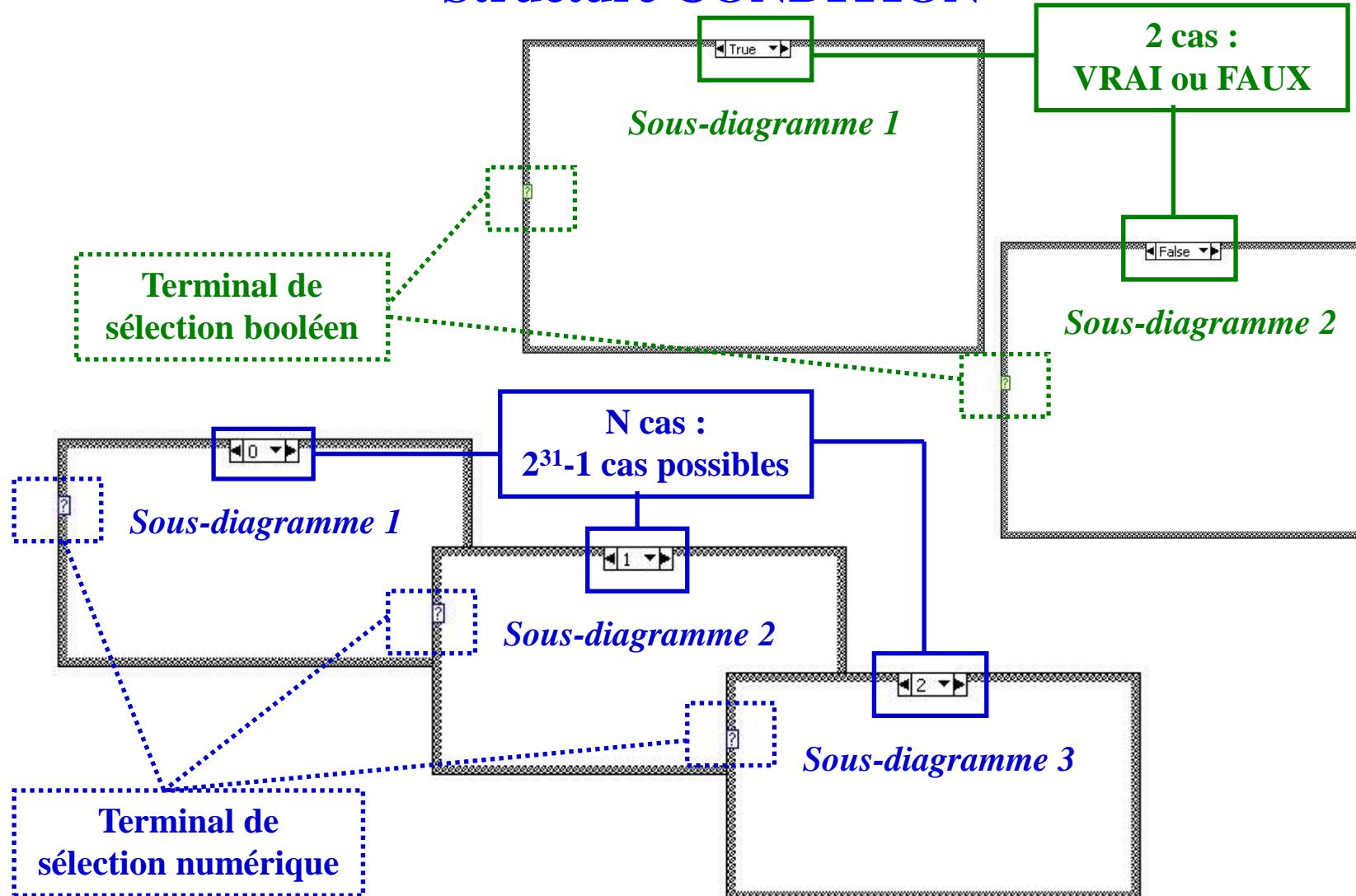
Application d'une structure condition à sélection booléenne:

Quelle est la fonction réalisée par un VI dont le diagramme est représenté ci-dessous?



Boucles et structures

Structure CONDITION

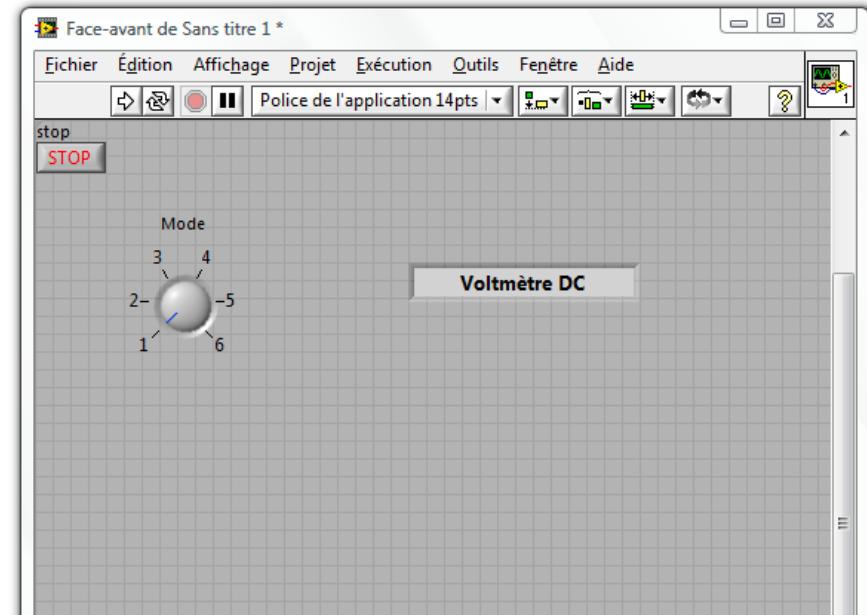
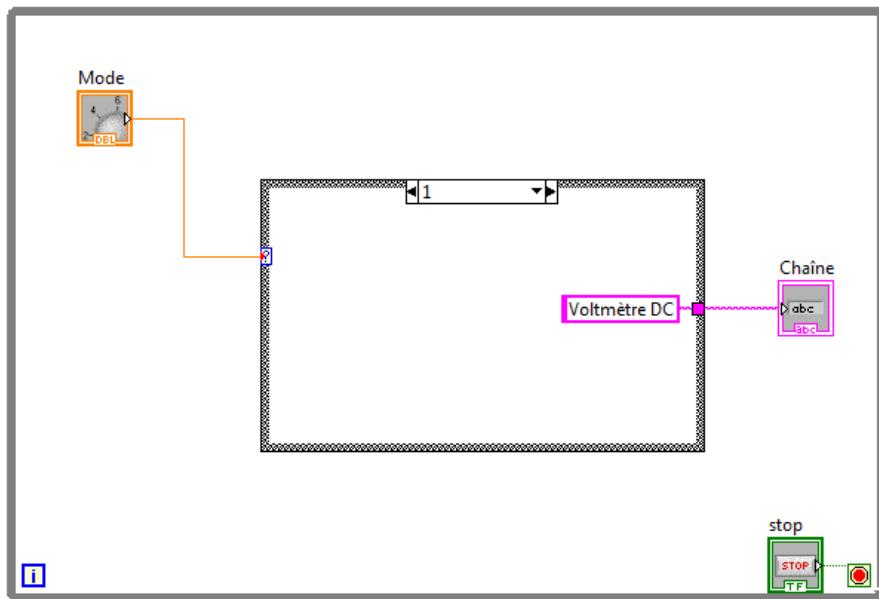


Boucles et structures

Application d'une structure condition à sélection numérique:

Reproduire le VI ci-dessous avec les différents mode suivants:

| Choix | 0, par défaut | 1 | 2 | 3 | 4 | 5 | 6 |
|-------|------------------|-----------------|-----------------|-------------------|-------------------|---------|-----------|
| Mode | Mode inconnu | Voltmètre DC | Voltmètre AC | Ampèremètre DC | Ampèremètre AC | Ohmètre | Wattmètre |



Boucles et structures

Structure SEQUENCE

Programmation en langage textuel



Chronologie d'exécution

=

Chronologie écriture

Programmation en langage G



Chronologie d'exécution

=

Chronologie de disponibilité des données aux entrées

Hiérarchisation d'exécution des nœuds

⇒ développement de structures SEQUENCE

Application représentative

⇒ Initialisation et configuration d'un appareil de mesure avant la collecte et le traitement les données

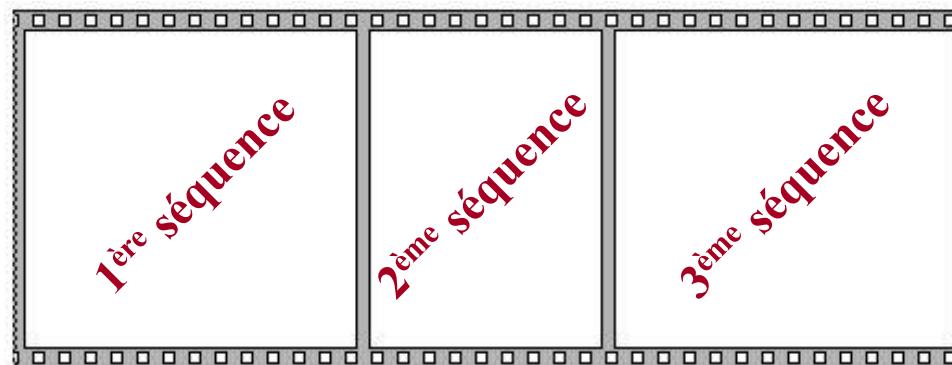
Boucles et structures

Structure SEQUENCE

⇒ Structure qui permet l'exécution de sous-diagrammes de manière séquentielle

⇒ Structuration chronologique des actions

Cet outil permet à LabVIEW de contrôler l'ordre dans lequel les nœuds du VI doivent s'exécuter



Graphiquement comparable à une séquence cinématographique

1^{er}
diagramme
à exécuter

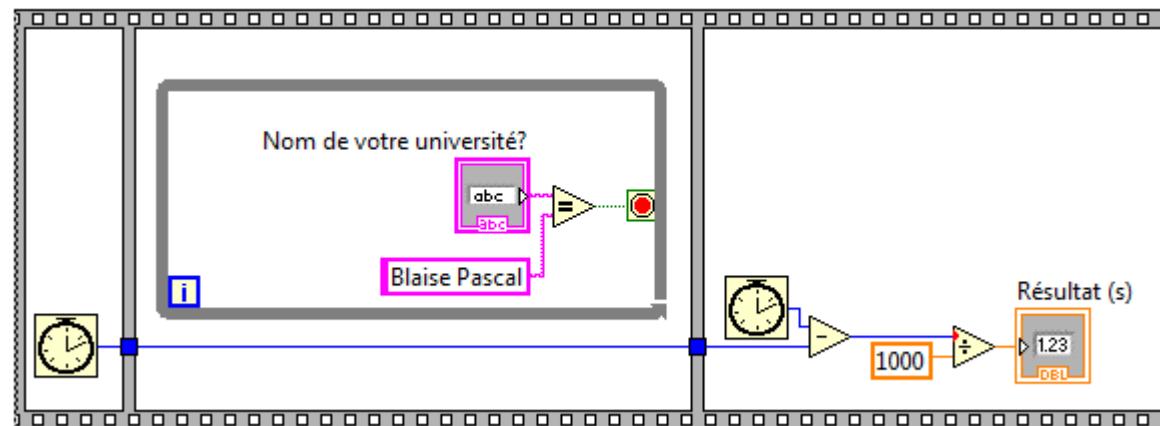
2^{ème}
diagramme
à exécuter

3^{ème}
diagramme
à exécuter

Boucles et structures

Application d'une structure séquence:

Quelle est la fonction réalisée par un VI dont le diagramme est représenté ci-dessous?



Tableaux et graphes

Les tableaux

⇒ ensemble de données du même type à plusieurs dimensions

⇒ peuvent contenir jusqu'à 2^{31} éléments par dimension

Les données peuvent être de tout type !

Cas impossibles : tableaux de tableaux ou tableaux de graphes

Identification des données par son indice entre 0 et N-1
pour un tableau à N éléments

Tableaux à 1 dimension ⇒ 1 indice

Tableaux à 2 dimension ⇒ 2 indices (1 pour la ligne et 1 pour la colonne)

Tableaux et graphes

Tableau vierge

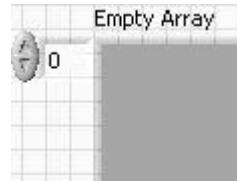


Tableau 1D

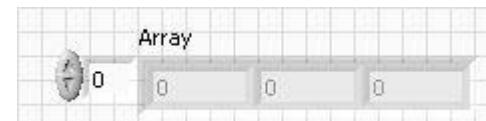
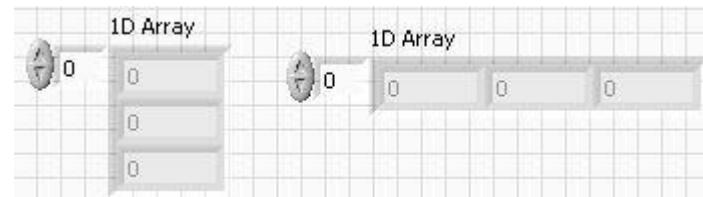
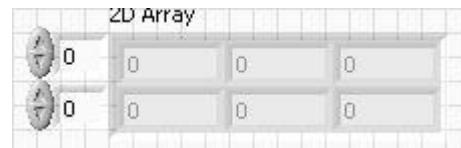


Tableau 2D



Remplissage du tableau \Rightarrow glisser une commande dans le cadre du bloc tableau

Tableaux et graphes

Créations automatiques de tableaux

- ⇒ à l'aide de boucles FOR ou WHILE
- ⇒ possibilité d'indexer et de générer des tableaux automatiquement à leur bordure

⇒ AUTO-INDEXATION !

⇒ activée par défaut pour les boucles FOR mais pas pour les boucles WHILE

une itération = une donnée du tableau !

Création de tableau 1D : boucle FOR auto-indexée !

Création de tableau 2D : 2 boucles FOR auto-indexées imbriquées!

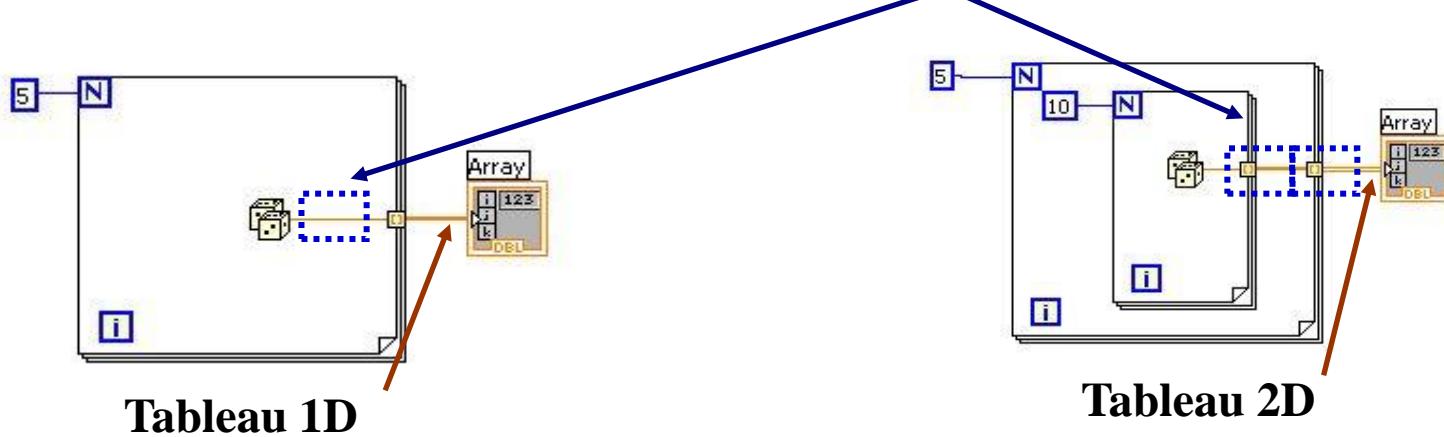
Boucle interne = création des colonnes

Boucle externe = création des lignes

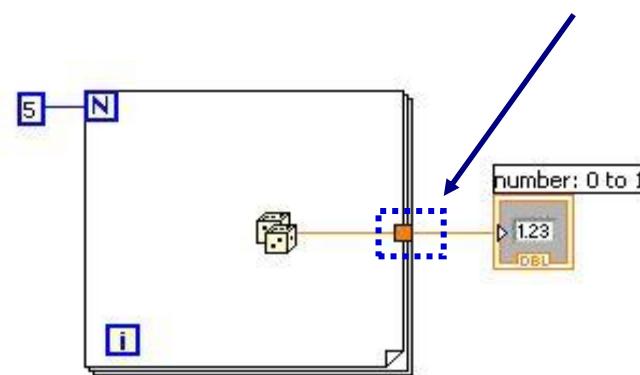
Tableaux et graphes

Créations automatiques de tableaux

AUTO-INDEXATION activée !



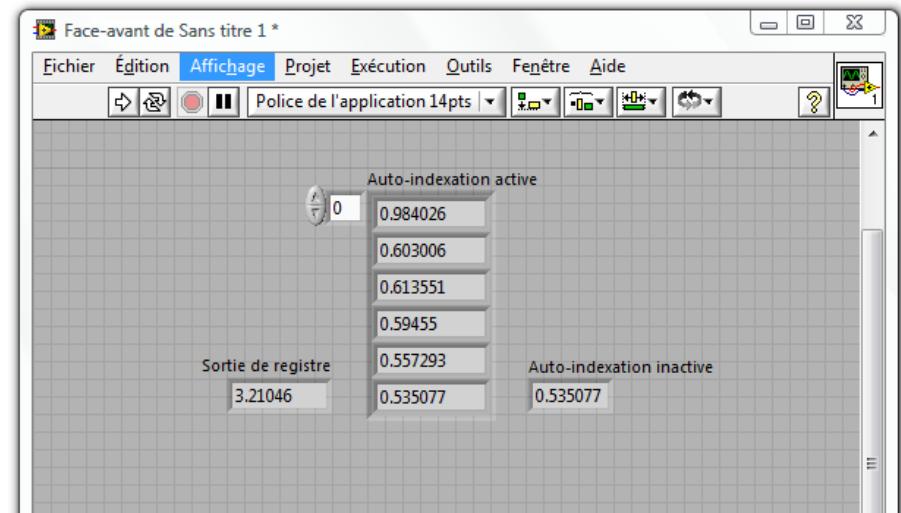
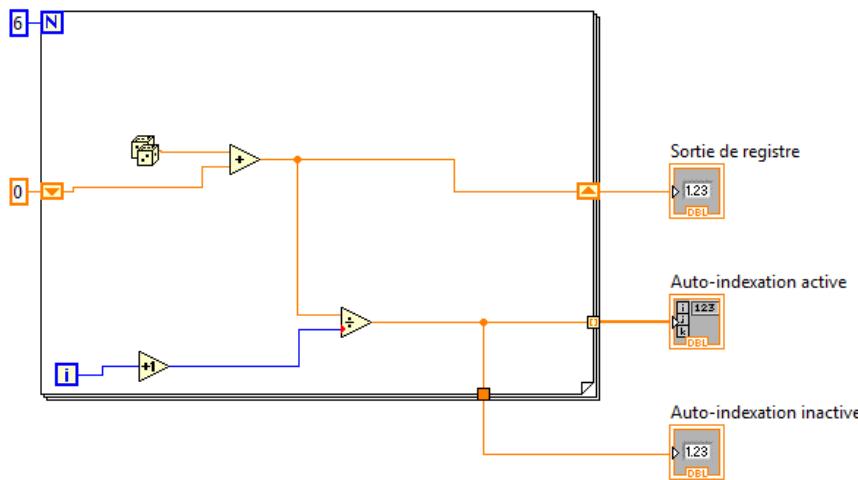
AUTO-INDEXATION désactivée !



Tableaux et graphes

Tableaux et auto-indexation:

Reproduire le VI ci-dessous et interpréter les résultats



Tableaux et graphes

Les graphes

⇒ représentation bidimensionnelle d'un ou plusieurs tableaux de données appelés *tracés*

⇒ 2 types de graphes:

Les graphes XY

Les graphes simples

Graphes XY : indicateurs graphiques permettant de tracer des points répartis à intervalles réguliers

Ex : tracé de tableaux de données régulièrement espacées

Graphes simples : objet graphique d'usage général, idéal pour tracer des fonctions multivariées

Ex : tracé de forme circulaire ou courbes variant avec le temps

Les graphes

Manipulations correctes des graphes \Rightarrow maîtrise des *clusters* (autre structure de LabVIEW)

Clusters \Rightarrow structure particulière qui regroupe des données de nature différentes (contrairement aux tableaux)

Comment se représenter un cluster ?

\Rightarrow Comparable à un câble téléphonique où chacun des fil représente un élément du cluster

Autres graphes particuliers \Rightarrow graphes déroulants

Les graphes déroulants

⇒ Indicateurs numériques particuliers qui affichent un ou plusieurs tracés

⇒ Défilement similaire à un enregistrement à papier type sismographe
(tracé de données suivant une vitesse de défilement définie)

⇒ 3 modes de rafraîchissement:

Graphe déroulant

Oscillographe

Graphe à balayage

Graphe déroulant ⇒ données cumulées à vitesse de défilement donnée

Oscillographe ⇒ données effacées dès le plein écran puis retracées

Graphe à balayage ⇒ conservation de toutes les données mais retracé et création d'une barre de positionnement

Rapidité

Quelques précisions sur les graphes

- ⇒ Sur un graphe déroulant peut être directement câblé un scalaire (entier ou réels)
- ⇒ Sur un graphe déroulant peuvent être affichés simultanément plusieurs tracé
Utilisation de la fonction *Bundle*

Fonction *Bundle* ⇒ permet la combinaison de plusieurs données en cluster

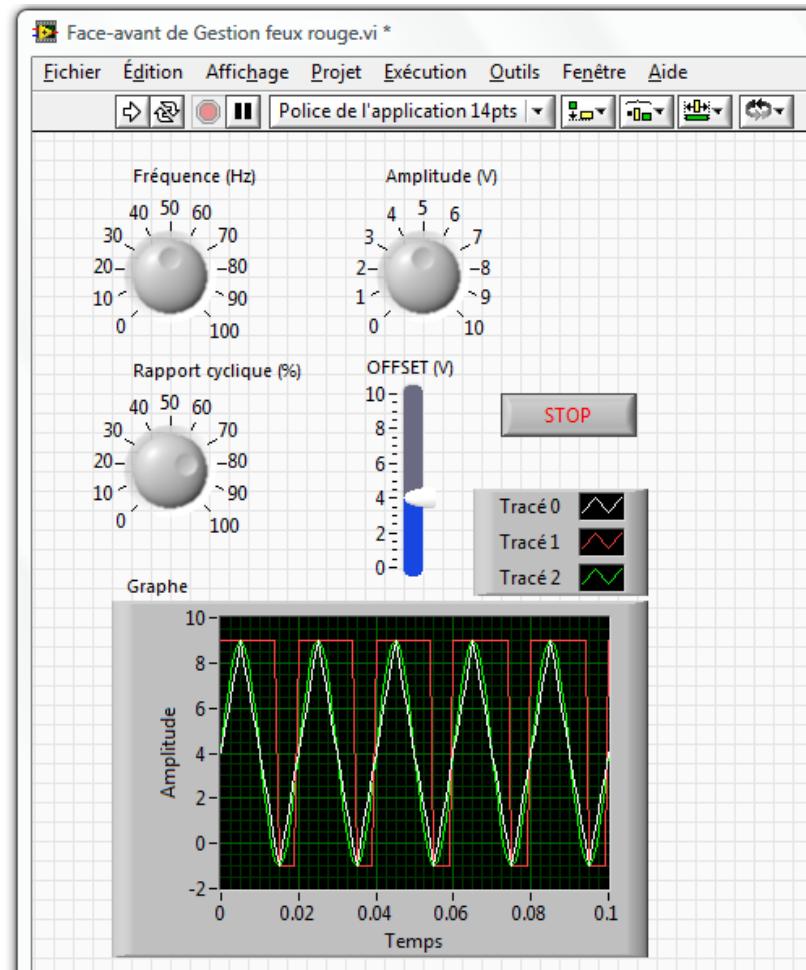
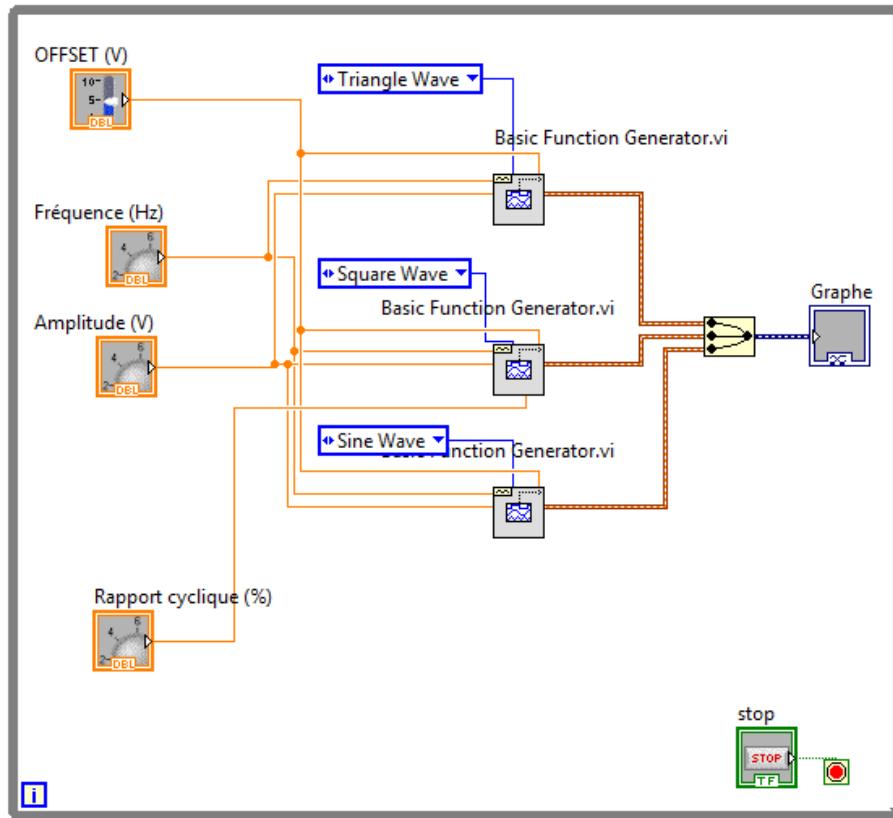
Exemple de combinaison avec la Fonction *Bundle*

2 tableaux (données X et Y)
Sortie connectées au graphe XY
Graphe XY = cluster

Tableaux et graphes

Application avec affichage graphique:

Reproduire le VI ci-dessous et interpréter les résultats



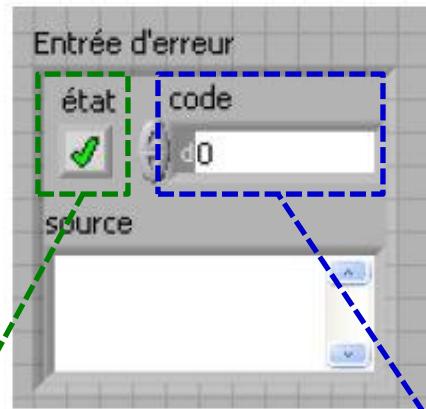
Autres applications des clusters

Gestion des erreurs par les clusters

Cluster d'erreur \Rightarrow permet la visualisation, la gestion et la transmission des erreurs entre terminaux au sein des V.I.s

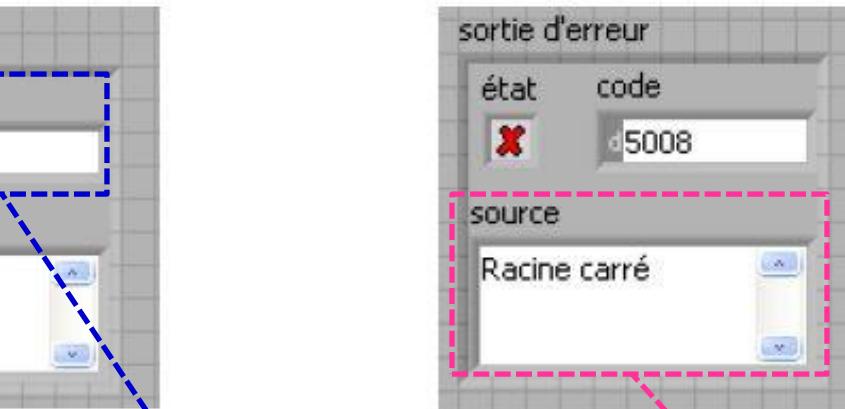
Exemple :

Indicateur face avant



Pas d'erreur

Information d'état
Variable type
booléen



Erreur

Nature de l'erreur
Variable type chaîne

Autres applications des clusters

Gestion des erreurs par les clusters

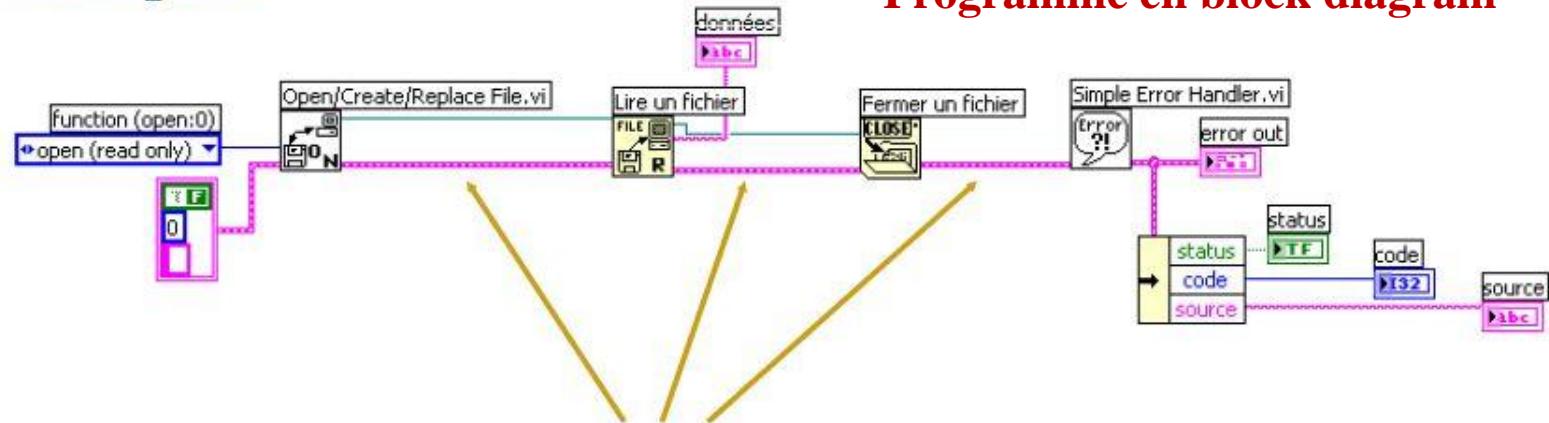
⇒ Transit de l'information erreur entre V.I. et sous-V.I.s

Si erreur dans un sous V.I. ⇒ exécution inhabituelle des sous-V.I.s suivants

⇒ Possibilité d'une gestion automatique des erreurs

Exemple :

Programme en block diagram



Clusters d'erreurs

Chaînes de caractères

Chaîne de caractères = suite de caractères ASCII

ASCII = American Standard Code for Information Interchange

ASCII = code le plus répandu pour tous les caractères alphanumériques

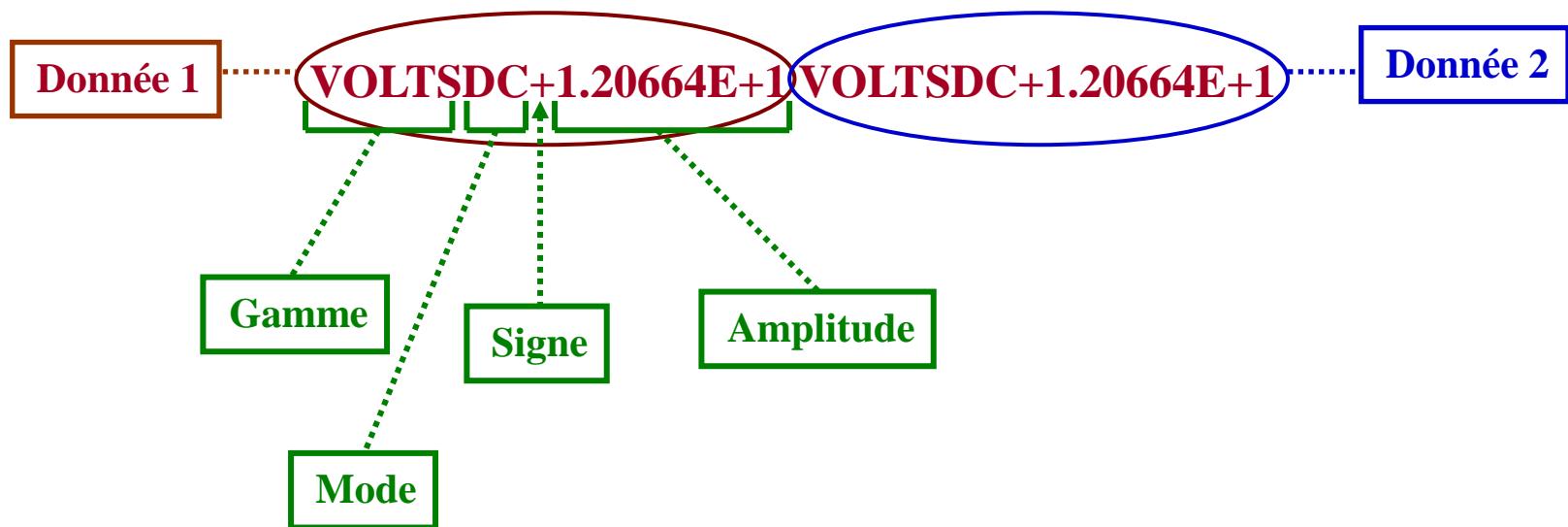
Chaînes de caractères ⇒ non restreint à la transmission et ou le traitement de messages textuels

Contrôle d'instrument ⇒ données = chaînes de caractères comprenant entre autre la valeur numérique

Conversion numérique de la chaîne de caractère obligatoire pour extraire l'amplitude de la donnée

Chaînes de caractères

Exemple de données enregistrées et transmises
par un multimètre KEITHLEY 2000



LabVIEW \Rightarrow nombreuses fonctions sur les chaînes de caractères

Exemple : Match Pattern \Rightarrow permet de décomposer les chaînes en divers éléments

Entrée = chaîne complète

Sortie = gamme + mode + amplitude

E/S sur fichiers

Opérations Entrées/Sorties sur fichiers

⇒ Enregistrer ou lire des informations dans des fichiers d'un lecteur

LabVIEW ⇒ nombreuses fonctions intégrées et différents V.I.s pour le traitement des entrées/sorties sur fichiers

⇒ Hiérarchie sur trois niveaux:

Fonctions E/S sur fichiers de bas niveau

V.I.s de fichiers intermédiaires

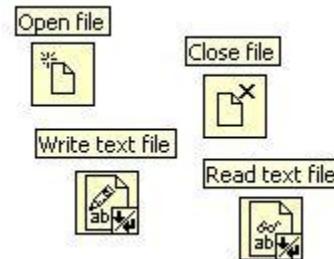
V.I.s de fichiers de haut niveau

Préférez les V.I.s de fichiers de haut niveau car ils gèrent de manière transparent les opérations d'ouverture et de fermeture de fichiers !

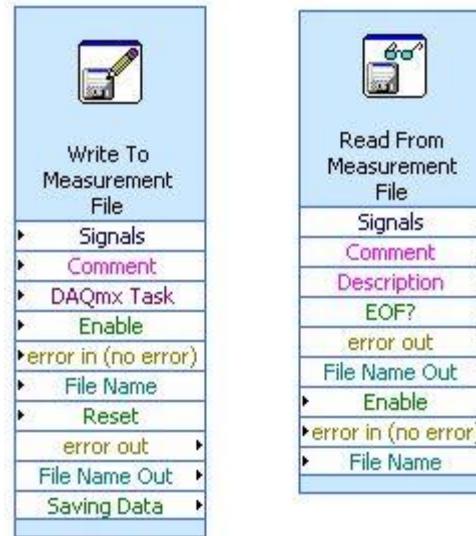
⇒ Font appel aux V.I.s de fichiers intermédiaires

E/S sur fichiers

V.I.s E/S sur fichiers de bas niveau



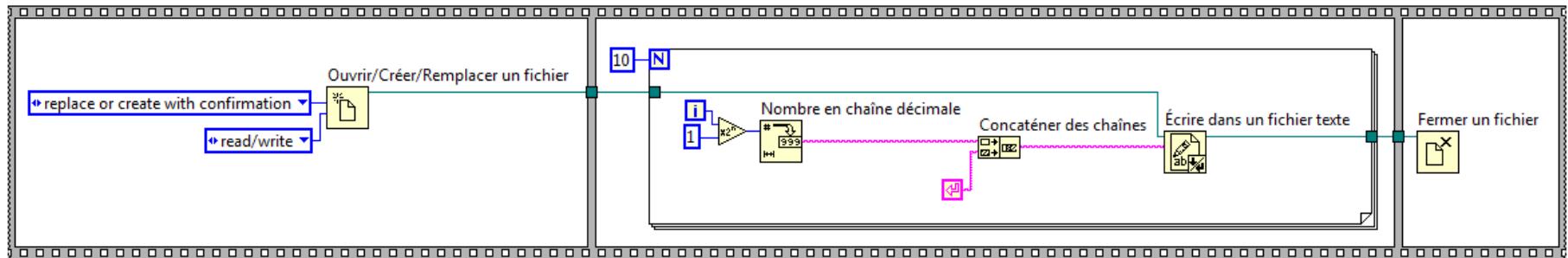
V.I.s E/S sur fichiers de haut niveau



E/S sur fichiers

Quelques opérations sur les fichiers:

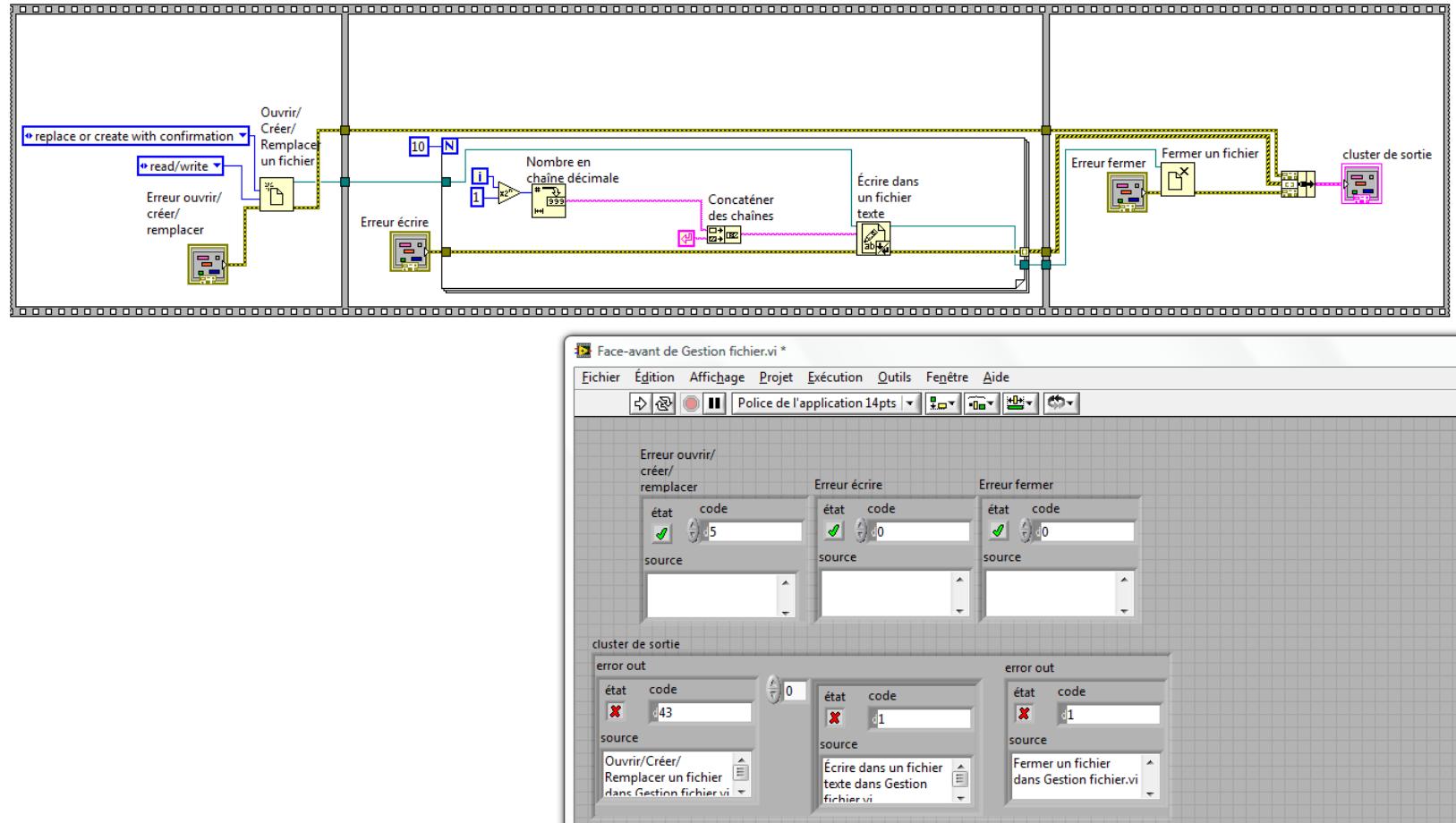
Reproduire le VI ci-dessous et interpréter les résultats



Autres applications des clusters

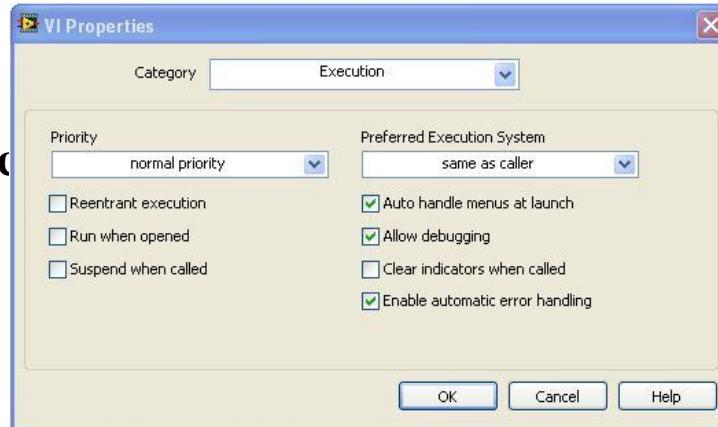
Complément : la gestion d'erreur

Compléter le diagramme précédent en incluant la gestion d'erreur



Configuration des V.I.s

Accès



��置 du VI possibles

du VI en face avant puis menu *VI setup*

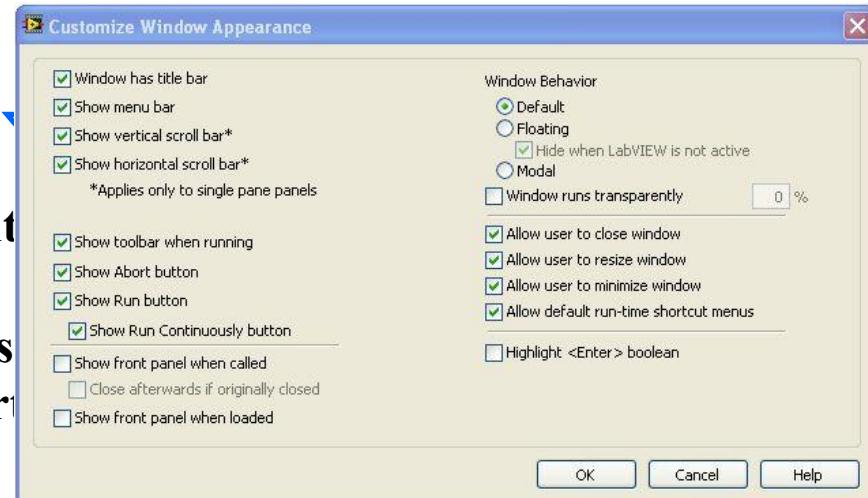
s options

n de l'exécution

⇒ configuration de l'apparence de la fenêtre

⇒ les options de fenêtrage s'appliquent

Possibilité ⇒ limiter les actions possibles
l'accès à cer



Configuration des V.I.s

⇒ Plusieurs options de configuration des sous-VI possibles également

Accès à ces options ⇒ clic sur l'icône du sous-VI dans le diagramme du VI appelant puis menu *subVI Node Setup* dans le menu *local*

Toutes commandes en face avant ⇒ option *Key Navigation* disponible



⇒ Associe une combinaison de touches à une commande en face avant qui permet en mode exécution de mettre en évidence la commande associée à cette combinaison

Commande numérique ou textuelle ⇒ *surbrillance*

Commande booléenne ⇒ *basculement de son état*

Quelques astuces pour simplifier les diagrammes

Les variables locales

Définition ⇒ duplications de terminaux dans un même diagramme

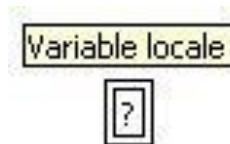
Fonctionnalités ⇒ pouvoir écrire (même pour les terminaux commande) ou lire (même pour les terminaux indicateur) en n'importe quel point du diagramme

Intérêts ⇒ initialisation de valeurs en face avant ou asservissement de commandes

Comment la créer ⇒ dans le diagramme, menu *Structures* dans la palette *Fonctions*
mais elle n'est associée à aucun terminal !

Associer la variable ⇒ clic droit sur la variable puis sélectionner un élément pour associer un terminal

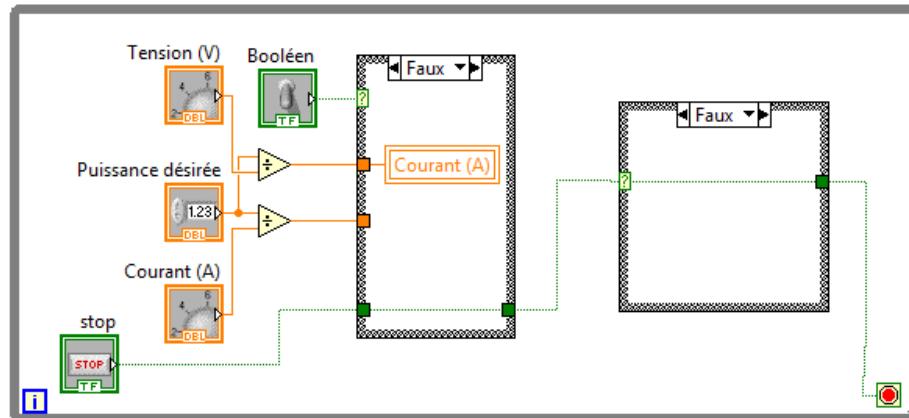
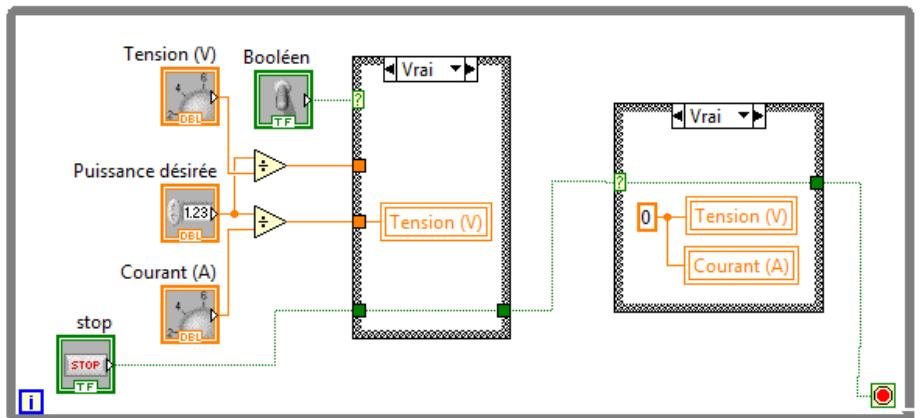
Alternative ⇒ clic droit sur le terminal puis créer la variable locale



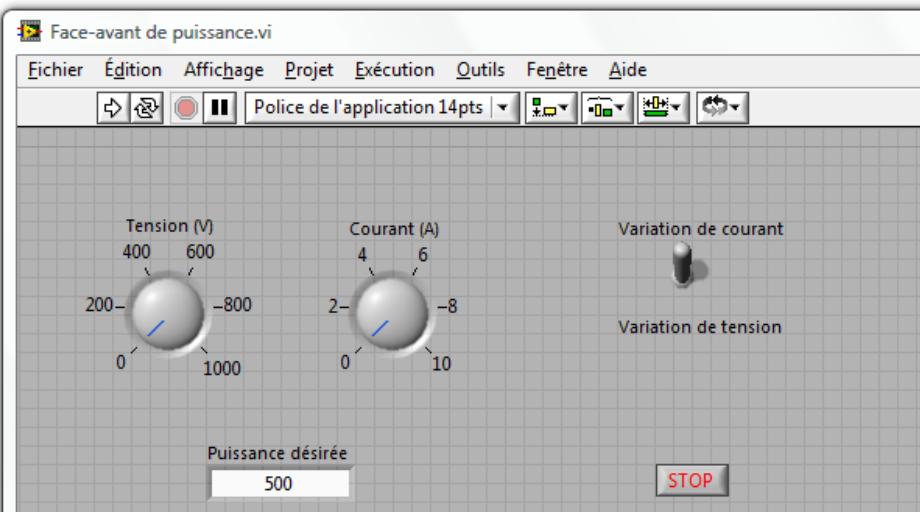
Quelques astuces pour simplifier les diagrammes

Utilisation des variables locales:

Reproduire le VI ci-dessous et interpréter les résultats



Conditions complémentaires



Quelques astuces pour simplifier les diagrammes

Les nœuds de propriétés

Définition ⇒ accès aux propriétés de terminaux

Fonctionnalités ⇒ changer la valeur, redimensionner, rendre visible, faire clignoter, changer la couleur... d'éléments de la face avant

Intérêts ⇒ pouvoir modifier par programmation l'apparence des objets de la face avant en réponse à certaines commandes

Comment la créer ⇒ dans le diagramme, menu *Contrôles d'application* dans la palette *Fonctions*
mais elle n'est associée à aucun terminal !

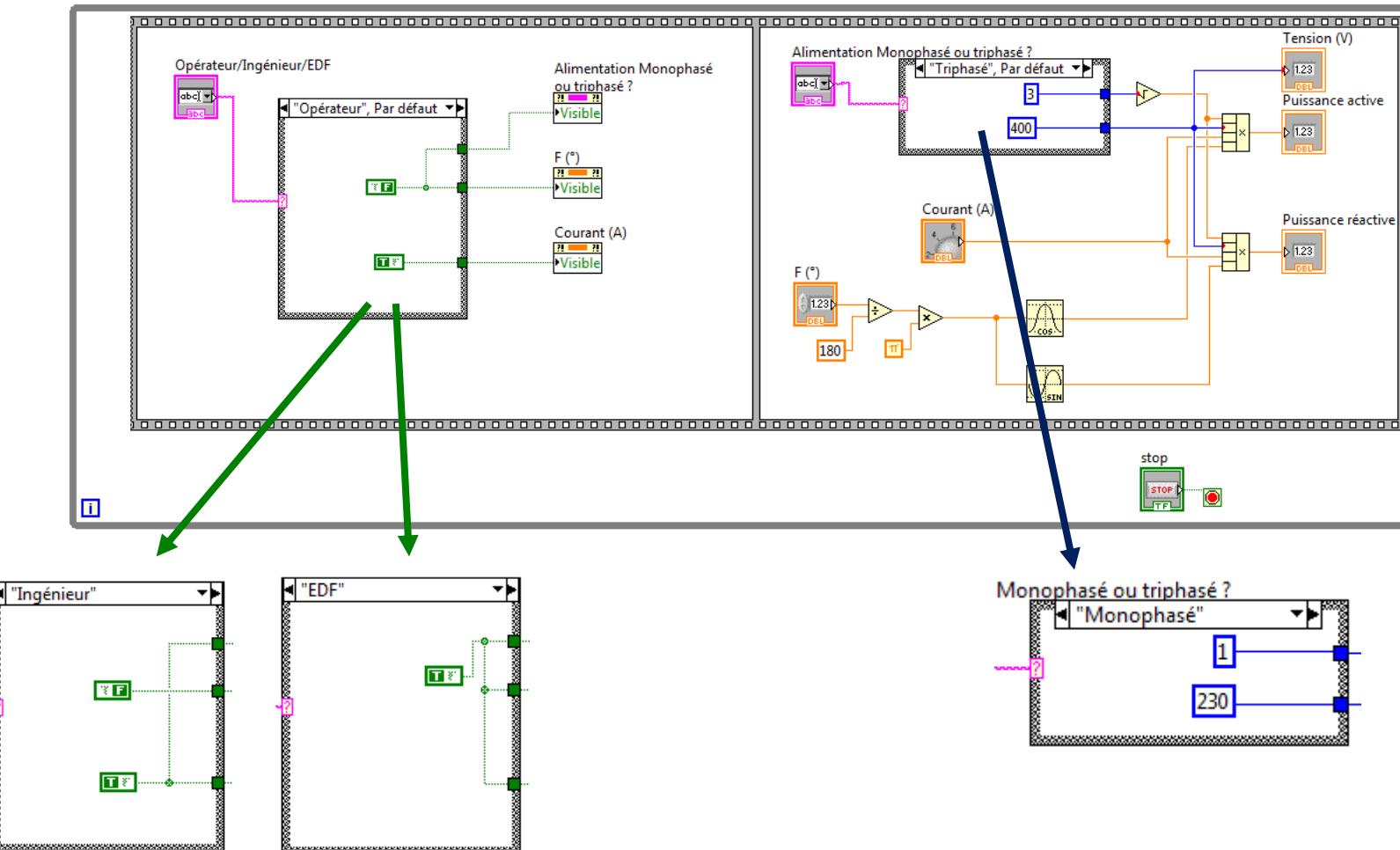
Associer la variable ⇒ clic droit sur le nœud de propriété puis *lier à* puis *sélectionner une propriété*

Alternative ⇒ clic droit sur le terminal puis créer le nœud de propriété puis choisir la propriété

Quelques astuces pour simplifier les diagrammes

Utilisation des nœuds de propriétés:

Reproduire le VI ci-dessous et interpréter les résultats



Quelques astuces pour simplifier les diagrammes

Les variables globales

Définition ⇒ duplications de terminaux utilisables dans n'importe quel programme

Présentation ⇒ sous-VI représenté uniquement par une face avant

Fonctionnalités ⇒ partage de données

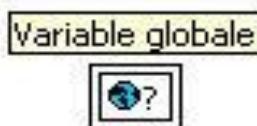
Comment la créer ⇒ dans le diagramme, menu *Structures* dans la palette *Fonctions*
mais c'est alors une structure vide !

Créer les données composant la variable globale

⇒ double clic droit sur l'icône de la variable pour ouvrir sa face avant

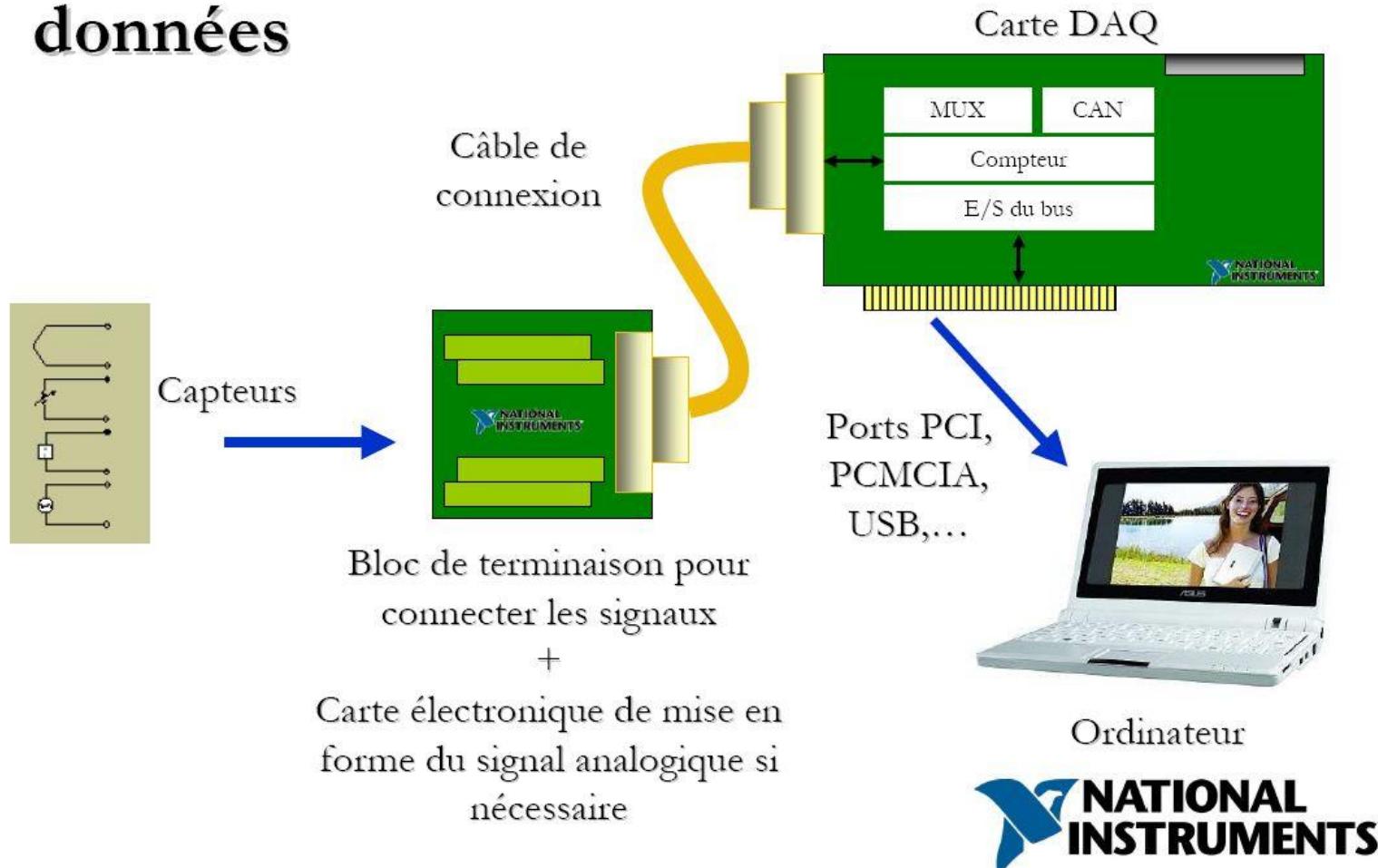
⇒ Créer les différentes données (numériques, booléennes, tableaux, cluster,...)

⇒ Sauvegarder la variable globale avec l'extension *.gbl*



Acquisition de données

Qu'est-ce qu'une chaîne d'acquisition de données?
Exemple de chaîne d'acquisition de données



Acquisition de données

Qu'est-ce que les NI-DAQmx ?

- ⇒ Ensemble de V.I.s permettant la gestion de signaux analogiques et numériques ainsi que leur gestion et leur traitement par l'environnement LabVIEW
- ⇒ Permettent de traiter plusieurs portions de code dans des threads différents (parties du microprocesseur) et ainsi d'optimiser l'utilisation du processeur

2 familles de V.I.s DAQmx:

V.I.s icône, polymorphes

V.I.s DAQ Express

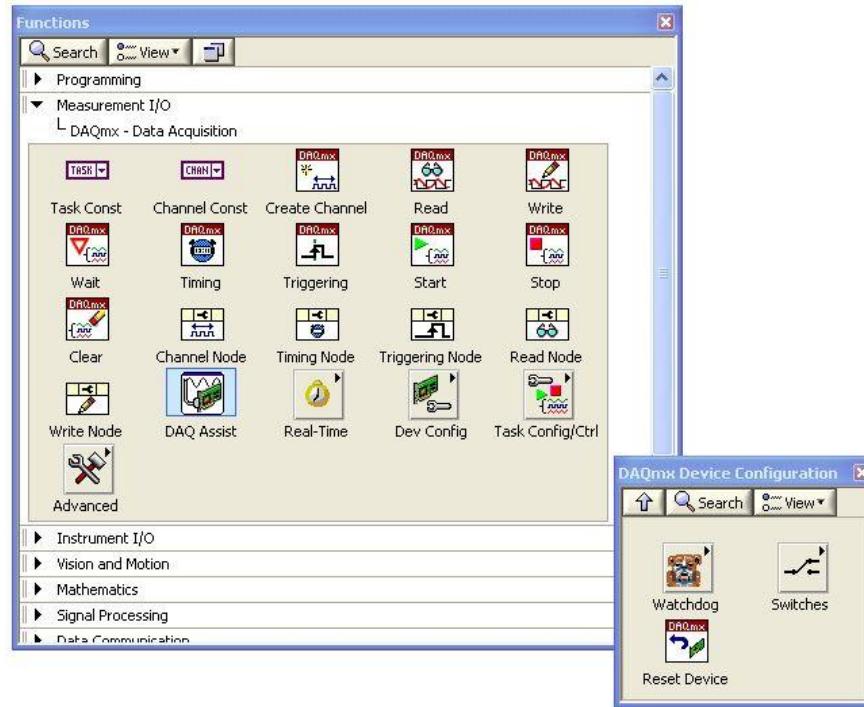
Intérêts

- ⇒ programme de moindre densité
- ⇒ codage multithread donc vitesses d'exécution différentes compatibles

Acquisition de données

Le menu NI-DAQmx

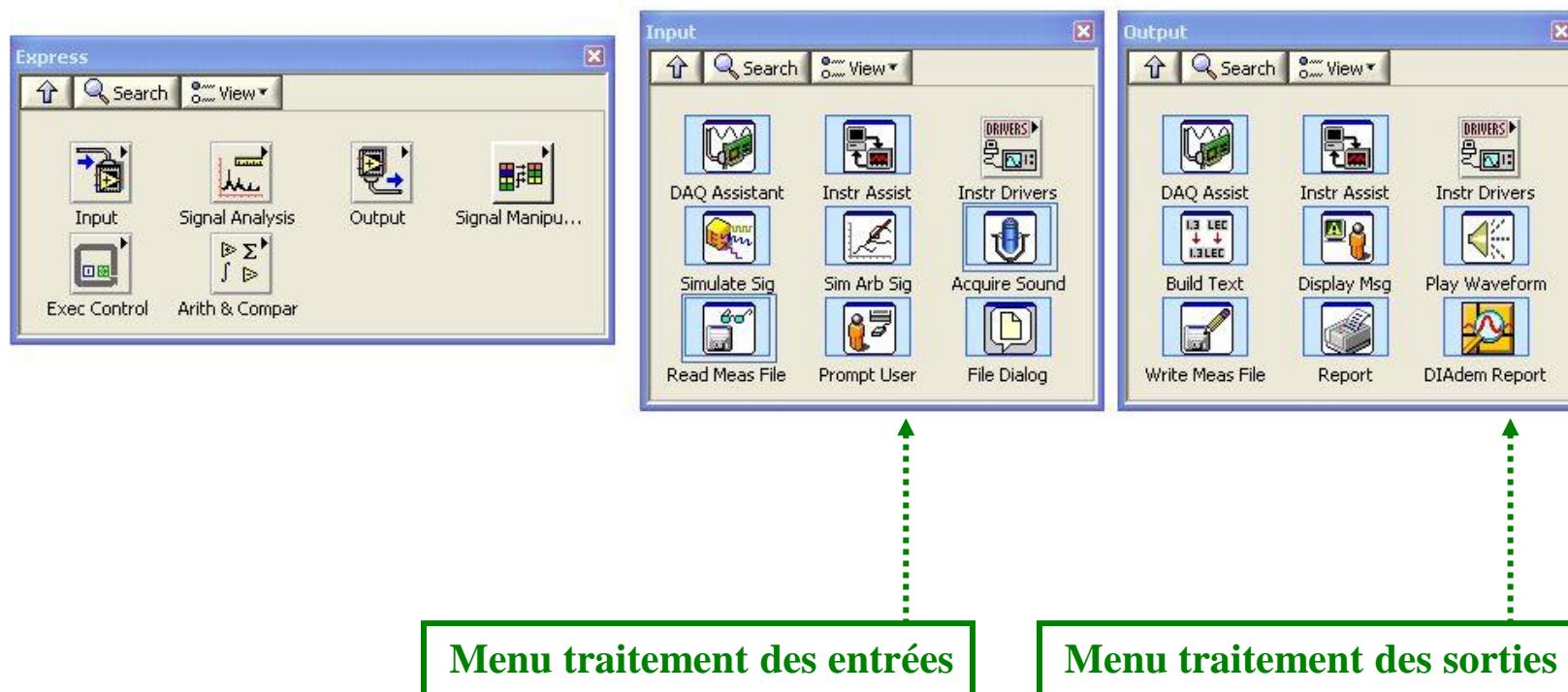
⇒ Bibliothèques de fonctions de traitement et de configuration de signaux acquis par des cartes périphériques développées par NI



Acquisition de données

Les V.I.s DAQ express

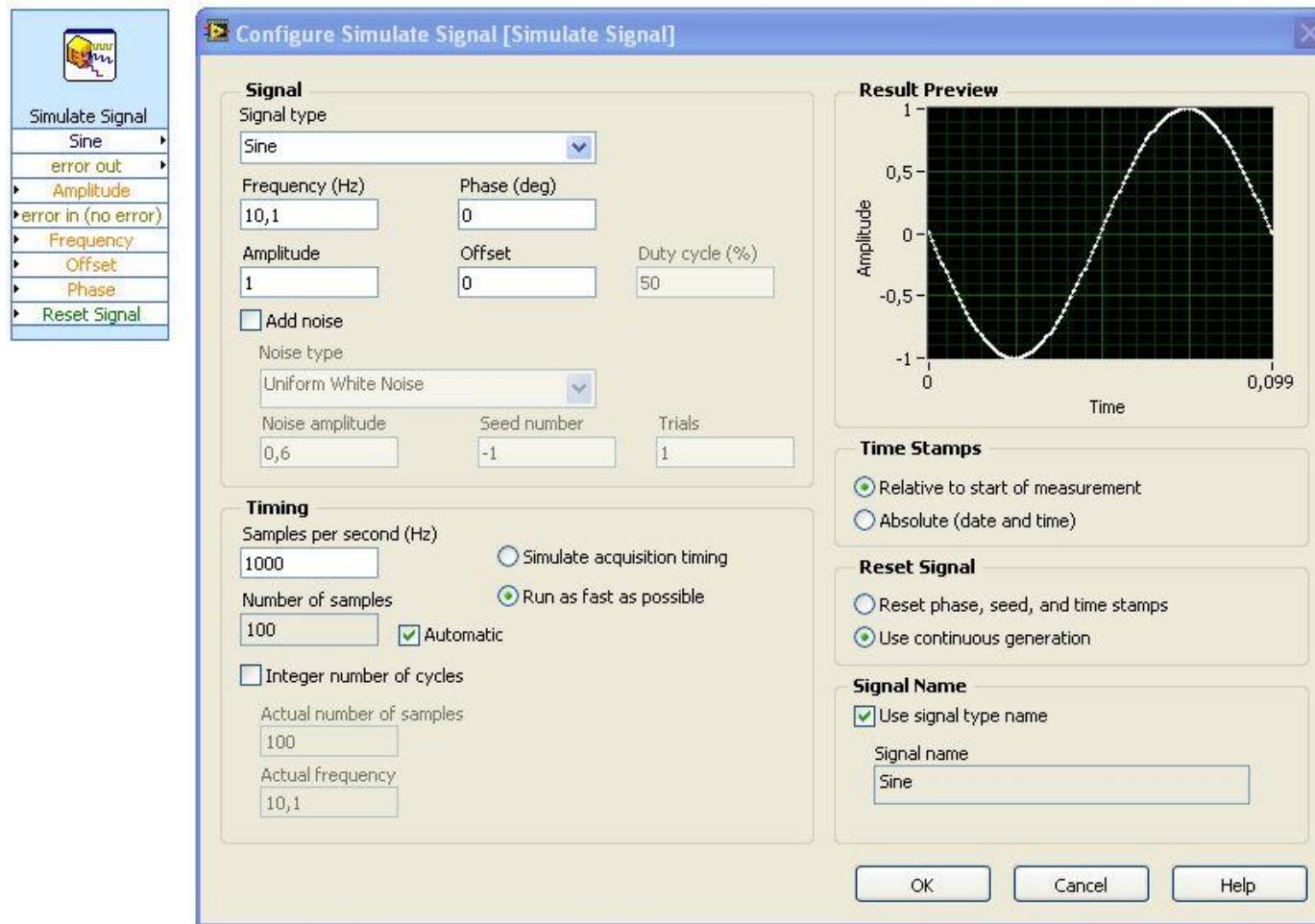
- ⇒ Bibliothèques de sous-VIs dont les paramètres sont accessibles par un menu
- ⇒ sous-VIs permettant de gérer, de configurer et de traiter les entrées/sorties



Acquisition de données

Quelques exemples de VI express

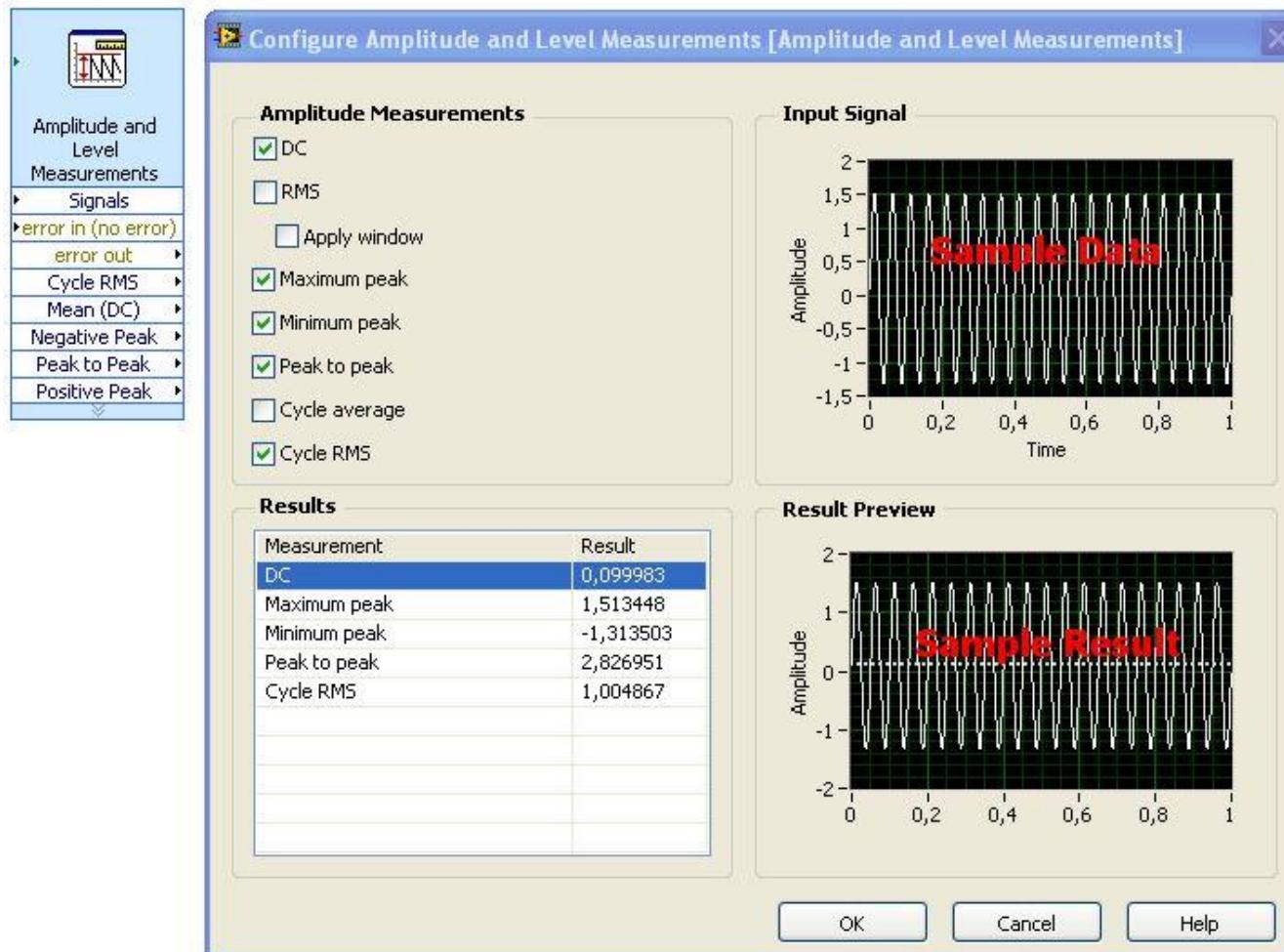
⇒ Générer un signal électrique paramétrable...



Acquisition de données

Quelques exemples de VI express

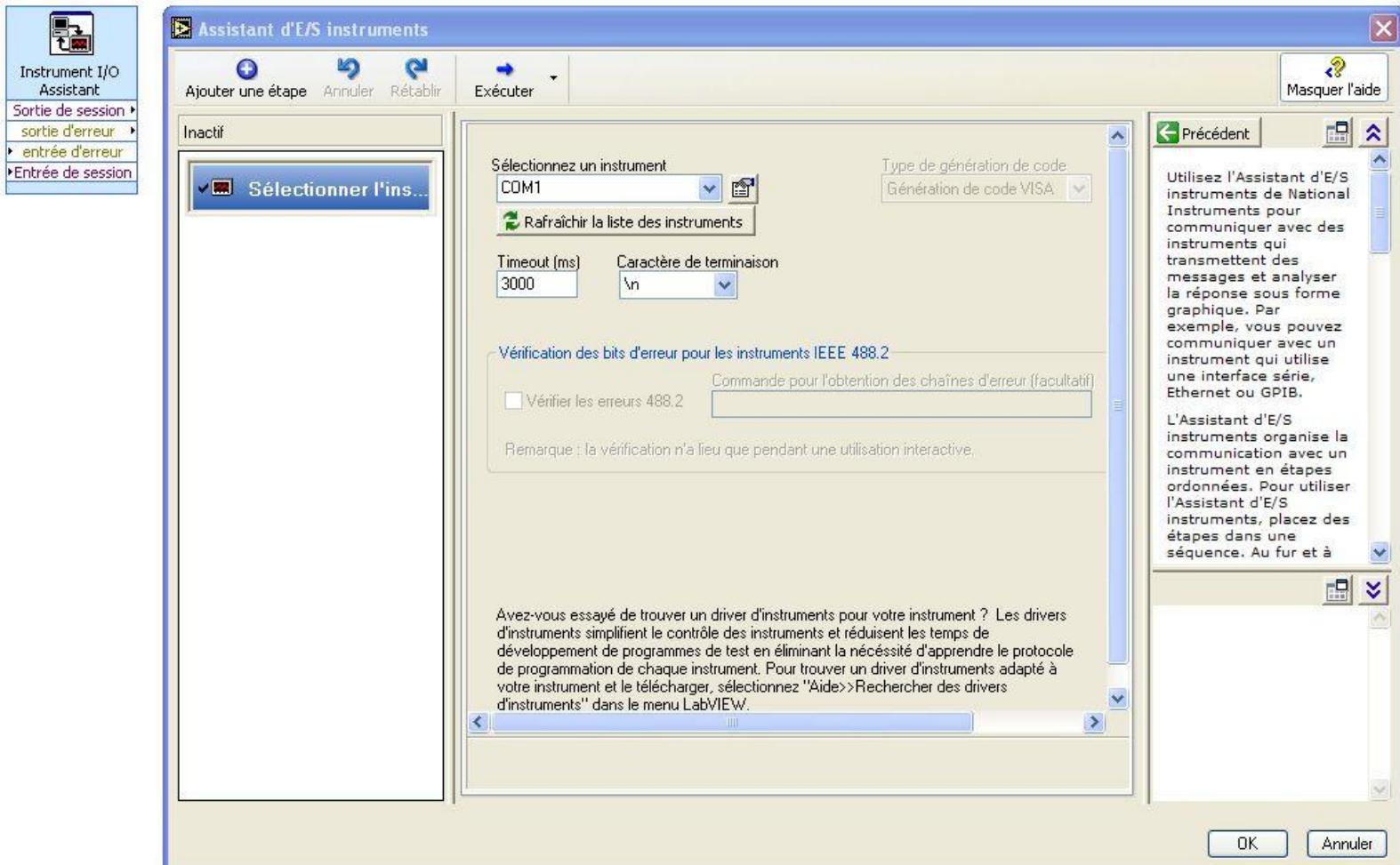
⇒ Calculer les grandeurs électriques représentatives d'un signal...



Acquisition de données

Quelques exemples de VI express

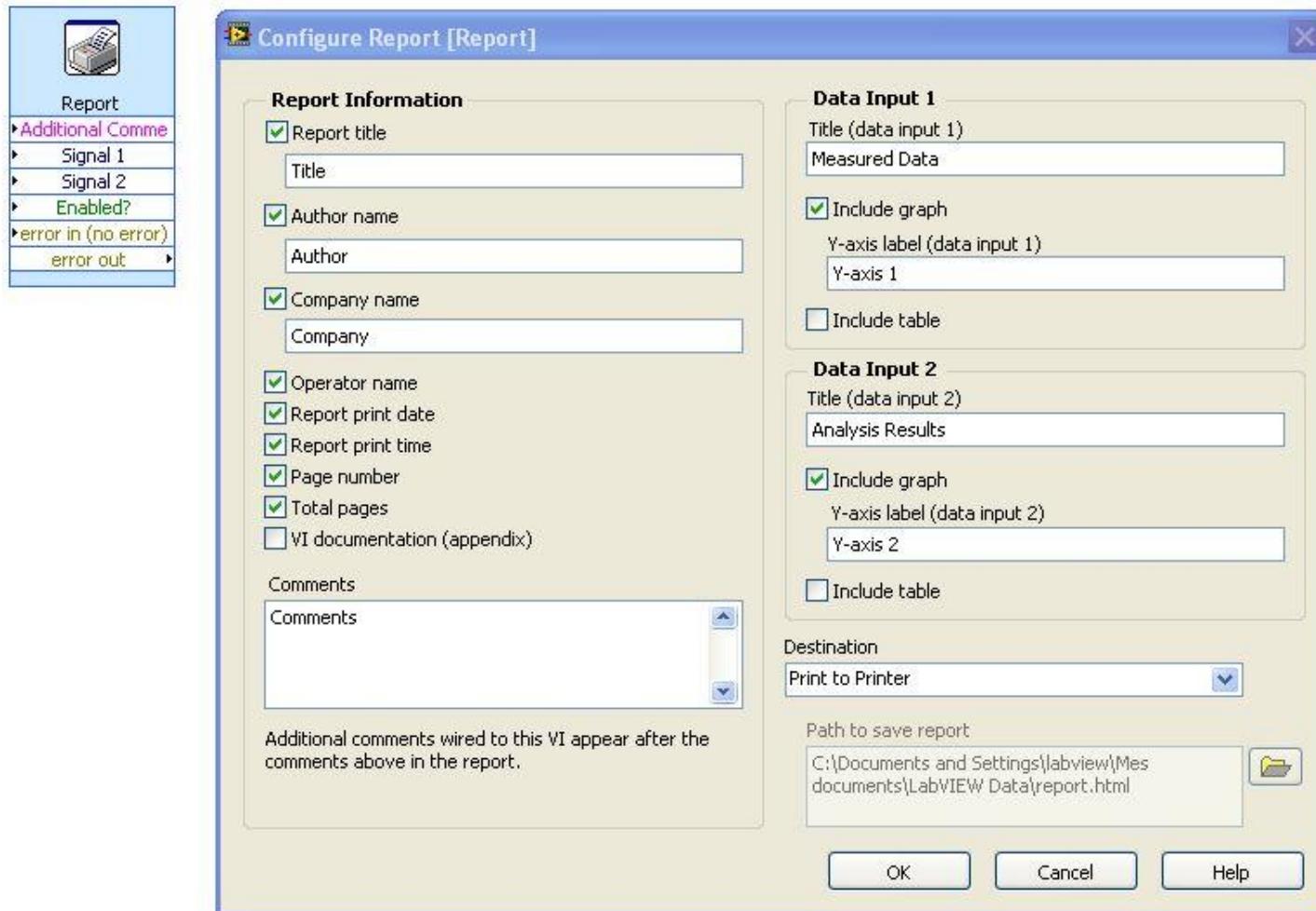
⇒ Communiquer avec un instrument commercial de mesure...



Acquisition de données

Quelques exemples de VI express

⇒ Générer un rapport de mesures...

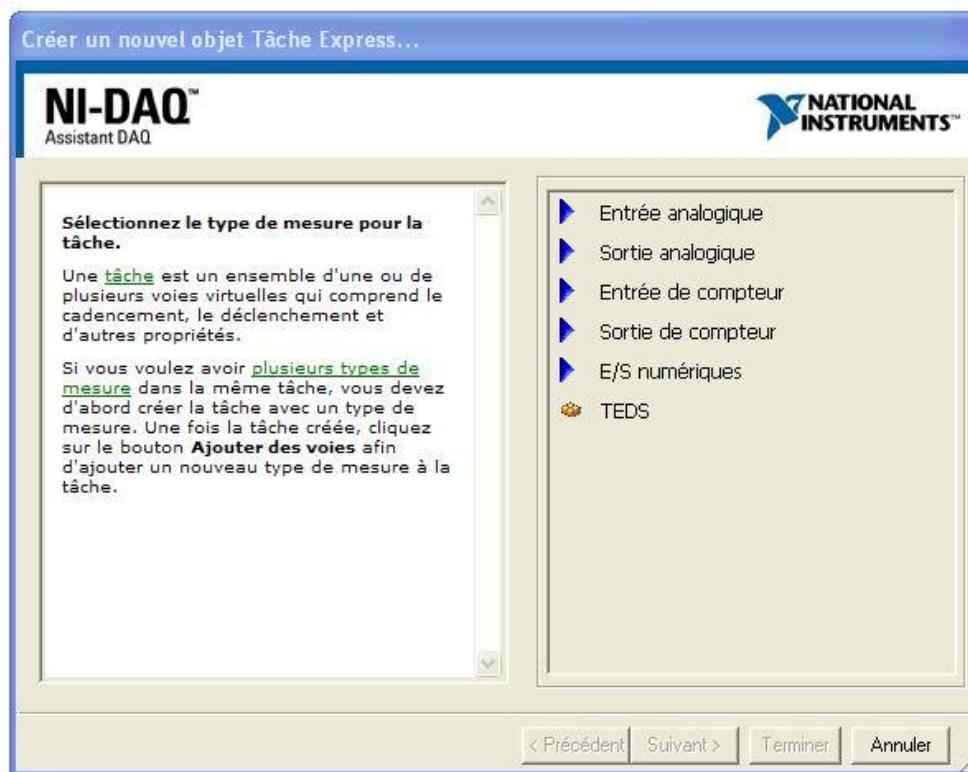


Acquisition de données

Le DAQ ASSISTANT

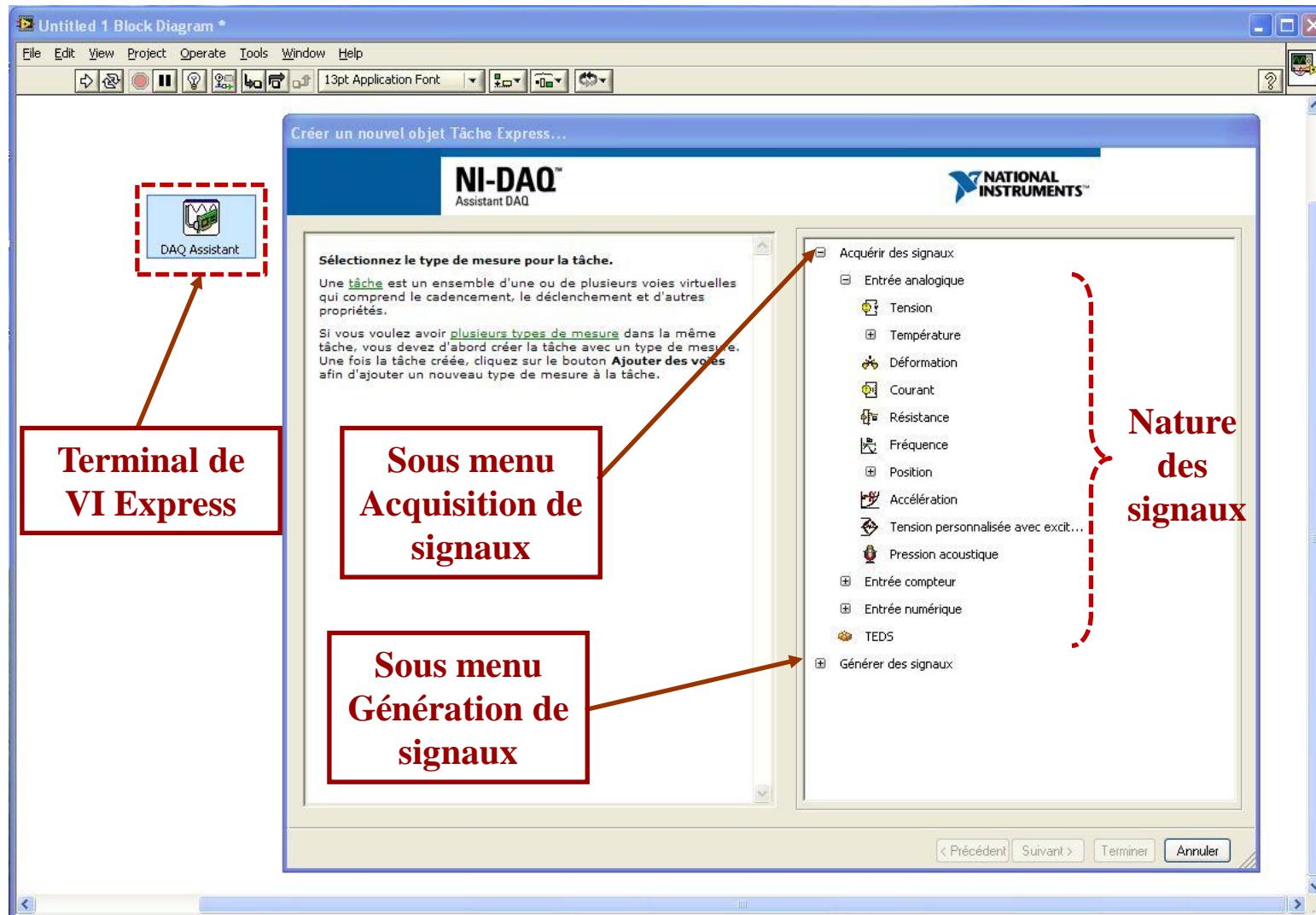
⇒ Assistant à l'acquisition de signaux

⇒ Prise en compte des diverses formes d'entrées/sorties possibles



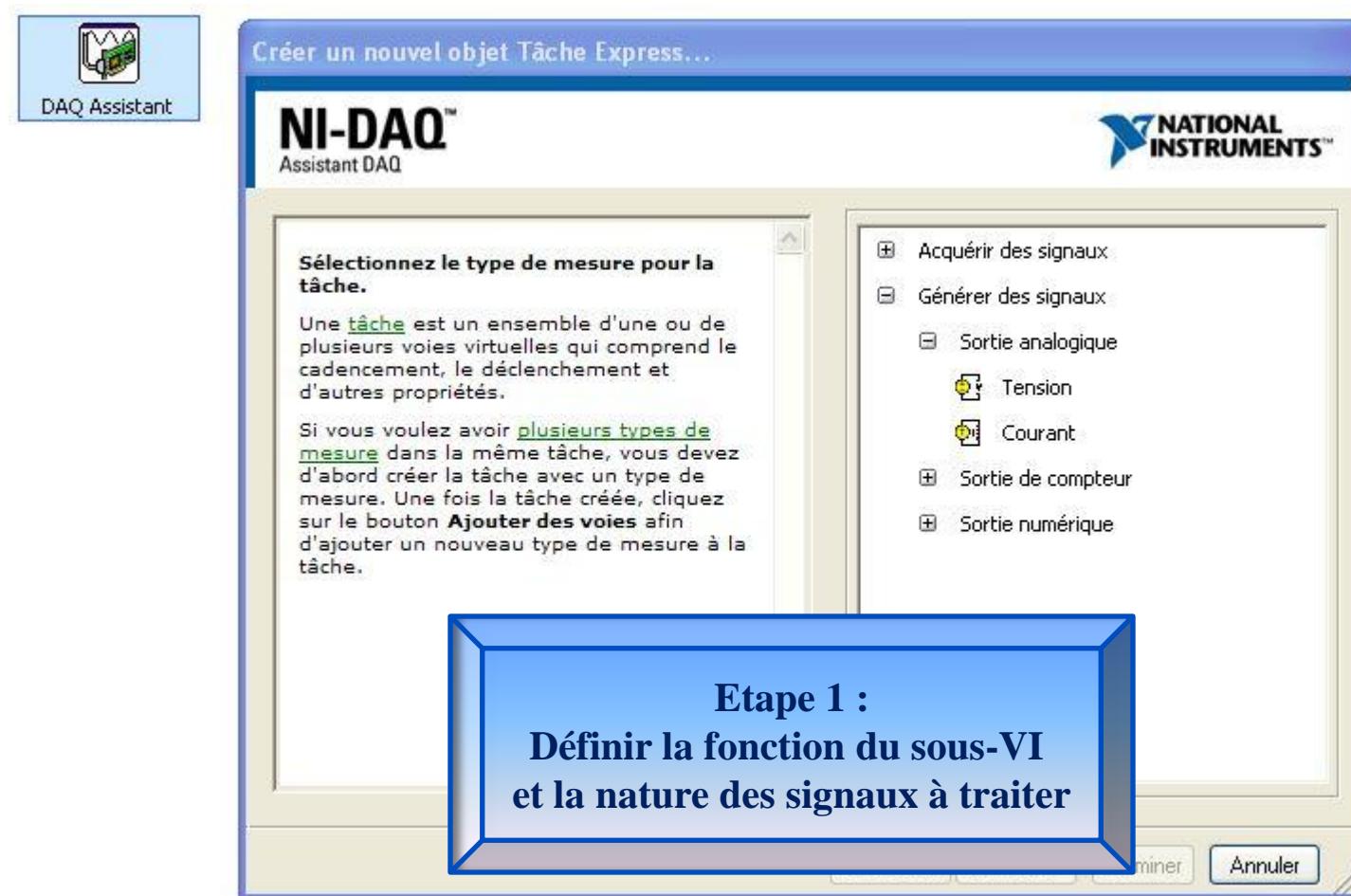
Acquisition de données

Interface DAQ ASSISTANT



Acquisition de données

Interface DAQ ASSISTANT



Acquisition de données

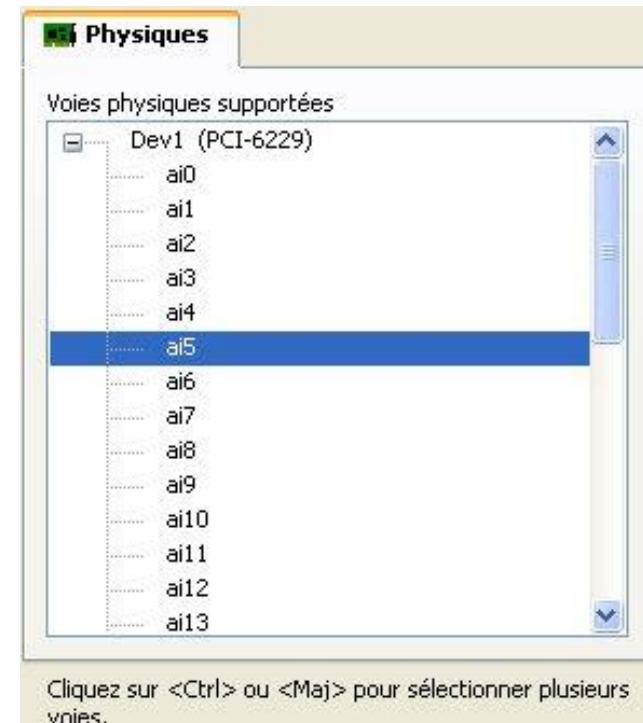
Interface DAQ ASSISTANT



Carte PCI d'acquisition multifonction



Etape 2 :
Définir les voies physiques
de la carte interface d'acquisition



Acquisition de données

Cartes d'acquisition de données

Quelques cartes d'acquisition de données



Carte NI PCI-6229



Boîtier NI USB-6008



MyNI

Contactez NI

Produits et Services

Solutions

Support

NI Developer Zone

Enseignement

Événements

Société

Page d'accueil > Produits et services > Matériaux d'acquisition de données (DAQ) > Acquisition de données (DAQ) multifonction > NI PCI-6229

De base (jusqu'à 250 kéch./s, 32 voies, 16 bits) > PCI, 32 AI, 48 DIO, 4 AO

NI PCI-6229

32 entrées, 16 bits, 250 kéch./s

[+] Envoyer cette page par e-mail

Configurer la page pour ::

[+] Imprimer

[+] PDF

[+] Rich Text

4 sorties analogiques 16 bits (833 kéch./s)

48 E/S numériques ; compteurs 32 bits ; déclenchement numérique

E/S numériques corrélées (32 lignes cadencées, 1 MHz)

Certificat d'étalonnage du NIST et plus de 70 options de conditionnement de signaux

Sélectionnez les matériaux de la Série M haute vitesse pour des fréquences d'échantillonage 5 fois plus élevées ou de la Série M haute précision pour une résolution 4 fois plus élevée.

Driver NI-DAQmx et logiciel interactif d'enregistrement de données NI LabVIEW SignalExpress

[+] Comparez à des produits similaires

[+] Afficher la fiche technique et les spécifications (angl.)

Description

Spécifications

Prix

Services

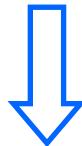
Ressources

Besoin d'aide ?

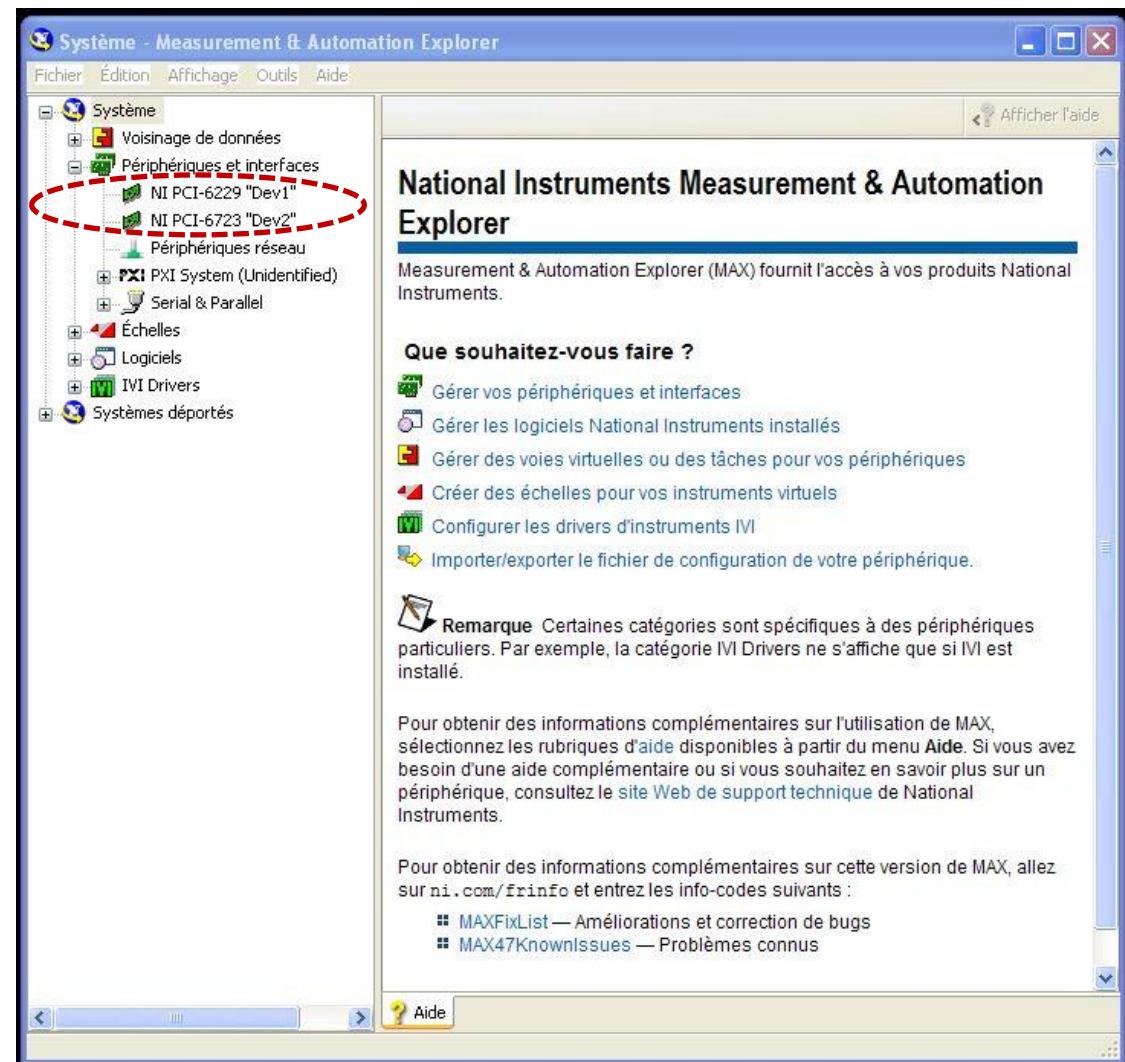


Acquisition de données

Measurement & Automation explorer

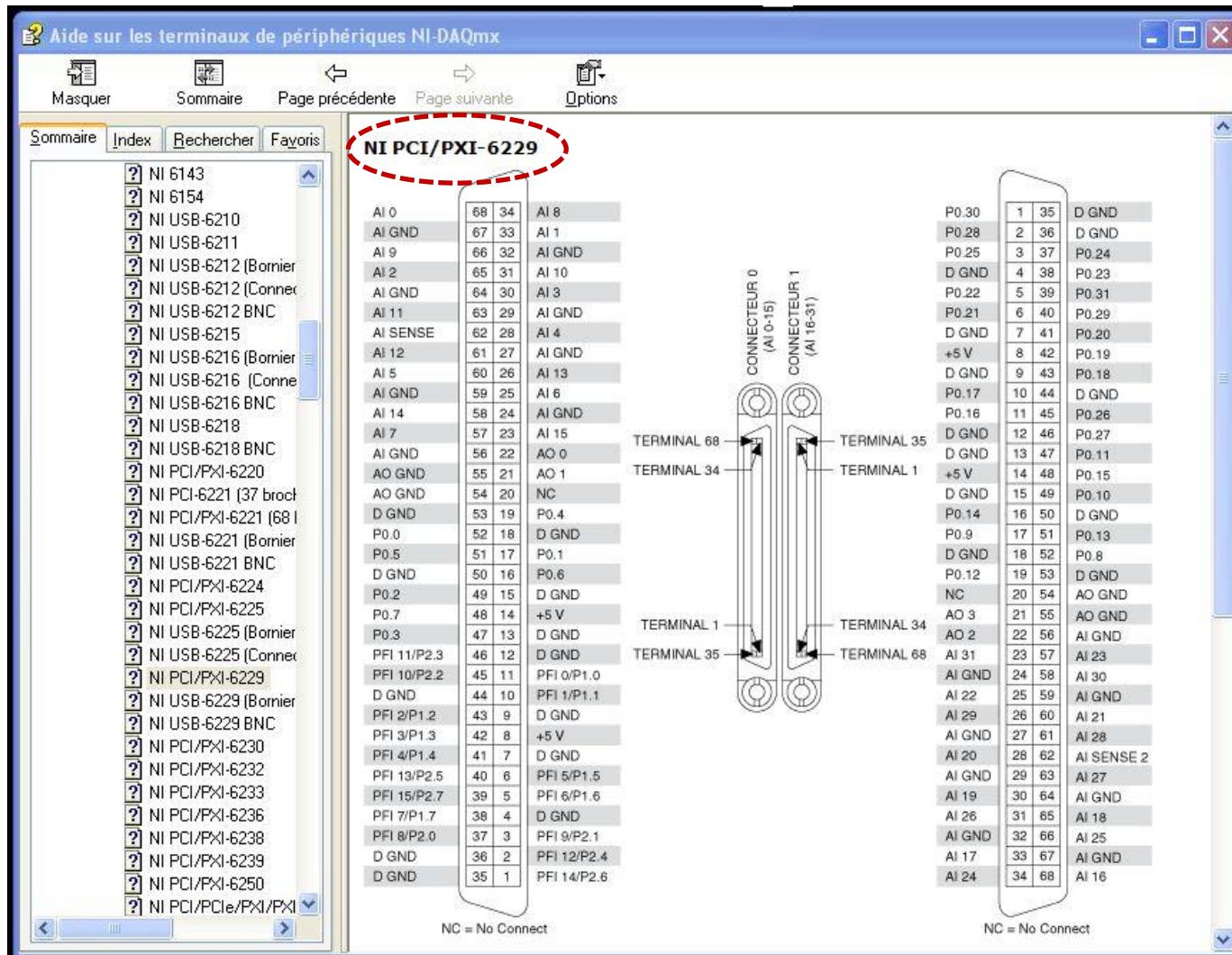


Accès aux ressources matérielles
Détection et communication
avec les appareils GPIB
Test des cartes d'acquisition
et de génération
Recalibration



Acquisition de données

Measurement & Automation explorer



Acquisition de données

Interface DAQ ASSISTANT

The screenshot shows the DAQ Assistant software interface. A red dashed box highlights the main workspace, which includes a graph area and a configuration panel. A blue box labeled "Etape 3 : Paramétrage des voies de traitement des données" is overlaid on the right side. Three red callout boxes point to specific features: "Affichage du signal" points to the graph area; "Configuration des voies d'acquisition" points to the configuration panel; and "Option d'échantillonnage" points to the sampling parameters.

Affichage du signal

Configuration des voies d'acquisition

Option d'échantillonnage

Etape 3 : Paramétrage des voies de traitement des données

de mesure sont conçus pour mesurer ou lire la tension. Les deux [mesures de tension](#) courantes sont celles de CC et CA.

Les tensions CC sont utiles pour mesurer des phénomènes qui varient lentement avec le temps, comme la température, la pression ou la déformation.

Les tensions CA, en revanche, sont des waveforms qui augmentent, réduisent ou inversent constamment la polarité. La plupart des lignes électriques acheminent une tension CA.

Ce graphique affiche les signaux analogiques acquis ou générés par le périphérique.

Acquisition de données

Interface DAQ ASSISTANT

Etape 4 :
Identification des associations
voies physiques / connections
sur bloc de terminaison

Diagramme des connexions

Voies de la tâche

Tension

Liste des connexions

| Point 1 | Point 2 |
|-------------|------------|
| Voltage/CH+ | CB-68LP/60 |
| Voltage/CH- | CB-68LP/26 |

Enregistrer en HTML...

NATIONAL INSTRUMENTS

CB-68LP

Connexion à CH+ et CH- pour la tension

Votre périphérique de mesure supporte la connexion à plusieurs accessoires. Utilisez l'accessoire connecté au câble/connecteur marqué 'Connector 0'.

Utilisez l'outil Diagramme des connexions pour afficher les connexions de signaux requises entre votre capteur et le bloc de connexion. Pour utiliser cet outil, vous devrez d'abord spécifier un accessoire pour votre périphérique. Faites un clic droit sur votre périphérique sous Périphériques et interfaces et sélectionnez Propriétés afin de spécifier un accessoire. Cet outil supporte actuellement les mesures de type entrées analogiques et la plupart des borniers à vis M10 et SXI.

Pour enregistrer un diagramme de connexions, cliquez sur Enregistrer en HTML et entrez un nom pour le fichier HTML. Un sous-répertoire images est créé, renfermant toutes

Sélectionnez chacune des Voies de la tâche pour afficher les connexions de la voie virtuelle.

OK Annuler



Blocs de terminaison

Terminal résultant



Acquisition et génération de signaux:

Créé un VI qui permet:

- l'acquisition d'une tension sur l'entrée analogique 0 da la carte d'acquisition;
 - de mesurer la valeur efficace et la valeur moyenne de ce signal;
 - retranche au signal d'entrée sa valeur moyenne;
 - écrête ce signal à +/- sa valeur efficace;
- génère ce signal sur l'une des sorties analogiques de la carte d'acquisition.

Acquisition de données

Matériels dédiés à l'acquisition sous LabVIEW

Exemples de matériels dédiés à l'acquisition

BNC-2090

Connectivité directe des capteurs pour une mise en oeuvre rapide.



22 connecteurs BNC pour les signaux analogiques, numériques, de déclenchement et de compteurs/timers.



SCB-68

Bloc de connexion d'E/S blindé pour interfaçer les signaux d'E/S avec des matériaux enfichables DAQ équipés de connecteurs 68 broches.

SCXI



Acquisition de données

Matériels dédiés à l'acquisition sous LabVIEW

Exemples de matériels dédiés à l'acquisition

ELVIS



Ensemble d'instruments virtuels : oscilloscope, multimètre numérique, générateur de fonctions,... pour le prototypage en laboratoire et l'enseignement.

CompactDAQ



Acquisition par USB
(Avantages du plug-and-play et de la connectique unique universelle).

Châssis modulaire



PXI/CompactPCI

Permet de répondre à un grand nombre d'applications de tests et de mesures.



Pilotes d'instruments

Les ports de communication LabVIEW-instruments



Port parallèle IEEE 488



Port série

IEEE 488 : 1 Mo/s
HS 488 : 8 Mo/s

Interface
GPIB/USB HS
de National
Instruments



Carte GPIB
Câble GPIB



Connectiques reliées au PC sur port PCI ou USB



Ports de communications divers

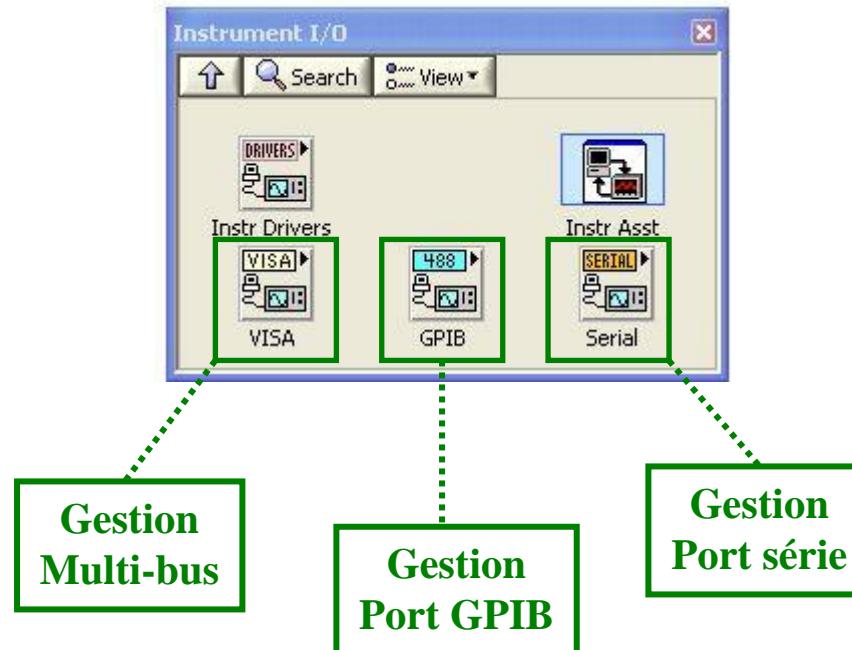
Pilotes d'instruments

Que sont les pilotes d'instruments ?

Pilote d'instrument ⇒ ensemble de routines qui contrôle un instrument programmable

Routine ⇒ opération simple de programmation

Menu pilote d'instrument:



Pilotes d'instruments

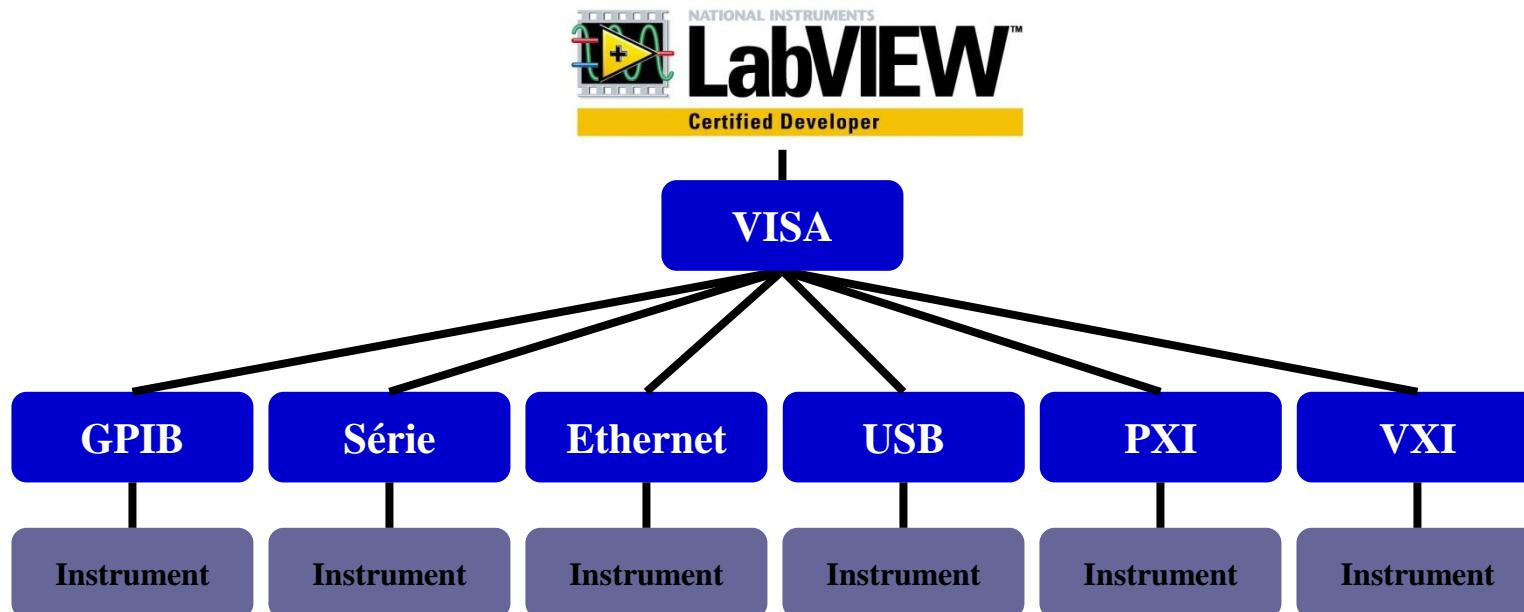
Qu'est-ce que VISA ?

VISA :

Virtual Instrumentation Software Architecture

VISA ⇒ pilote générique qui permet une forte adaptabilité / port de communication

Intérêt : interfaçage successif par plusieurs bus de liaison d'un appareil



Pilotes d'instruments

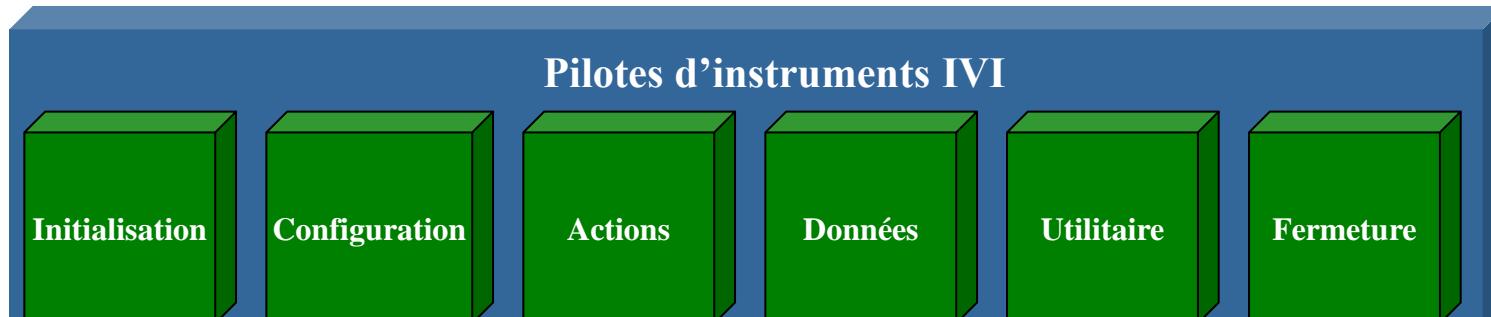
Qu'est-ce que IVI ?

IVI :

Interchangable Virtual Instrument

IVI \Rightarrow pilote d'instrumentation qui permet une forte interchangeabilité / type d'instrument

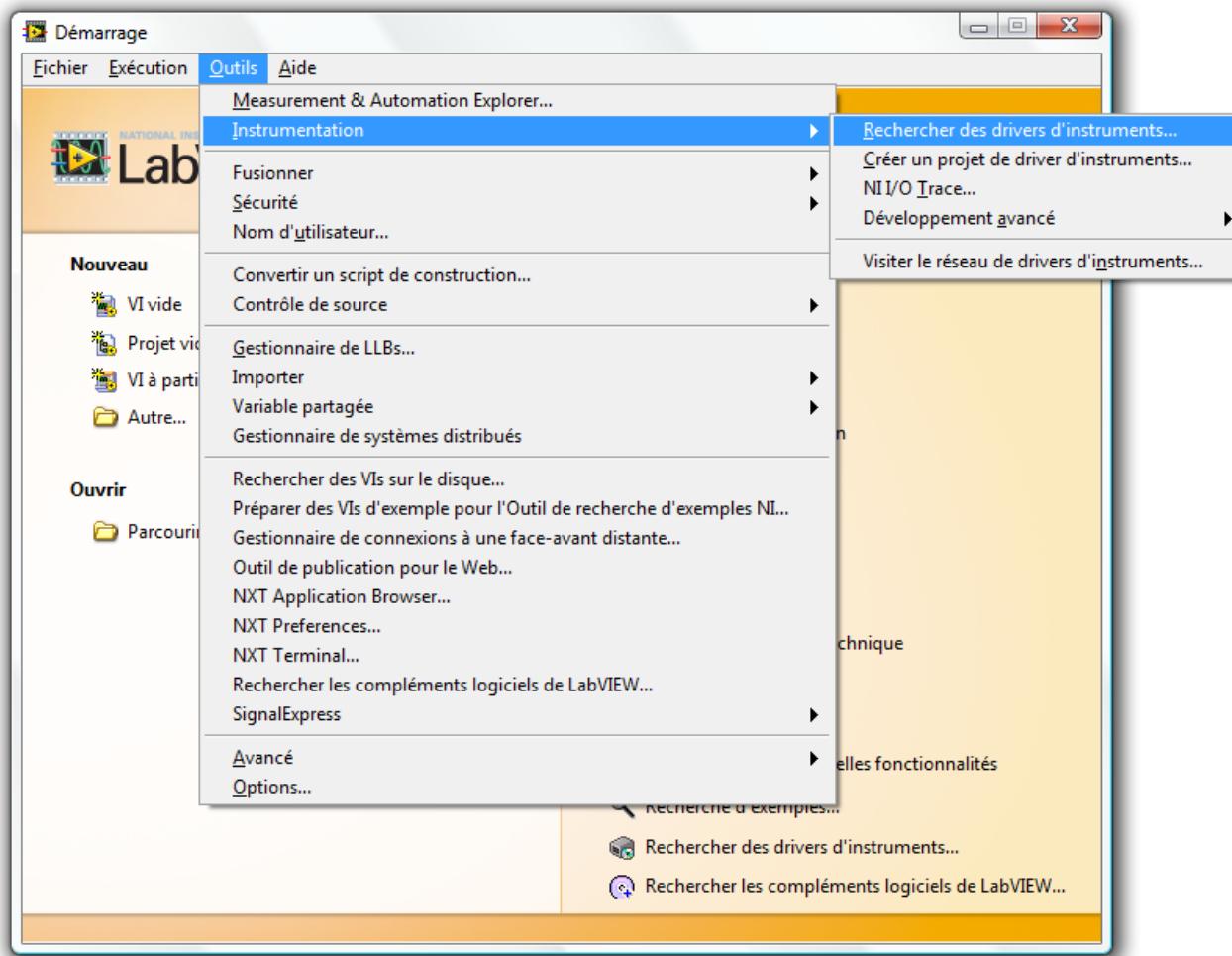
Intérêt : remplacement aisé d'un appareil par un autre à condition qu'ils aient des pilotes IVI certifiés



Fonction nécessairement présentes dans les pilotes IVI

Pilotes d'instruments

Y-a-t-il un pilote dans LabVIEW?



Pilotes d'instruments

Y-a-t-il un pilote dans LabVIEW?

The screenshot shows two windows from the NI Instrument Driver Search Assistant.

Left Window: "Assistant de recherche de drivers d'instruments NI - Configurer la recherche". It displays a tree view of connected instruments and installed drivers. Under "Instruments connectés", there is a folder for "Agilent 34401". Under "Drivers d'instruments installés", there is a folder for "Agilent Technologies". A message in the center says: "Vous n'avez pas ouvert de session. Les téléchargements de drivers d'instruments sont disponibles gratuitement aux utilisateurs enregistrés sur ni.com. Un message vous invitera à créer un nouveau profil ou à ouvrir une session lorsque vous sélectionnez le driver à installer." Below this are buttons for "Ouvrir une session" and "Rechercher les instruments". A dropdown menu for "Fabricant" is set to "Agilent Technologies". A search bar contains "53131" and a checkbox for "Drivers certifiés par NI uniquement" is unchecked. At the bottom, it says: "*Vous pouvez double-cliquer sur le nom d'un instrument connecté dans cette arborescence pour vous aider à créer votre demande, ou sur un driver installé pour commencer à l'utiliser." Buttons at the bottom are "< Précédent", "Rechercher >", "Fermer", and a close button.

Right Window: "Assistant de recherche de drivers d'instruments NI - Résultats de la recherche". This window lists search results. There are two entries:

| Driver | Certifié par N |
|----------------------------|----------------|
| ag5313xa Instrument Driver | Oui |
| hp5313xa Instrument Driver | Non |

Details for the first entry (ag5313xa Instrument Driver):

- 2.7 ★★★★☆ 6 évaluations
- Logiciels de développement pour le driver : LabVIEW Version min - 8.0
- Technologie du driver : Plug and Play (project-style)
- Logiciel de support nécessaire : NI-VISA Version min - 3.2
- Révision du driver : 1.0
- Fabricant : Agilent Technologies
- Modèles supportés : 53131A, 53132A

*Utilisez ces informations pour déterminer quel est le téléchargement de driver approprié pour votre instrument et environnement de développement.

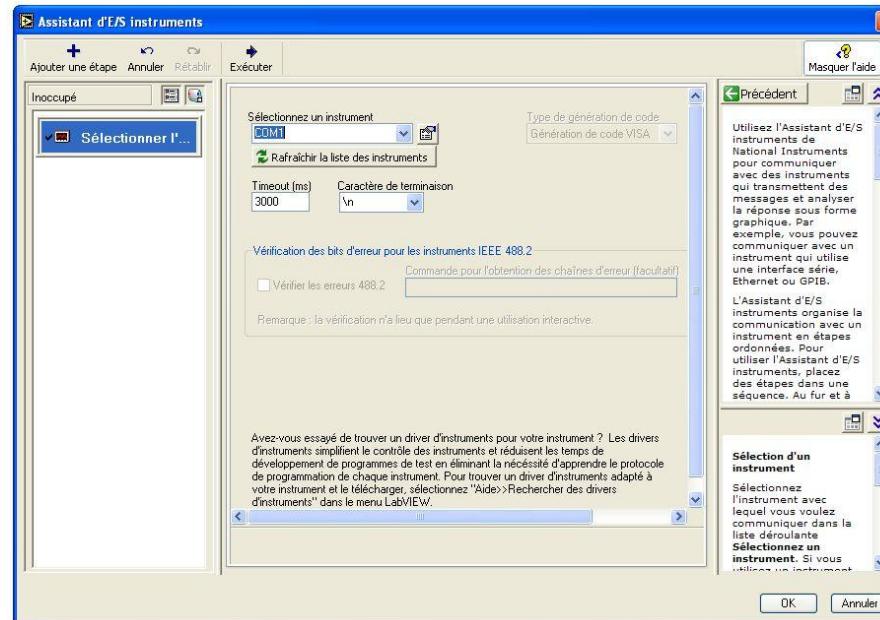
Buttons at the bottom of the right window are "< Précédent", "Installer", "Fermer", and "Aide".

Pilotes d'instruments

Y-a-t-il un pilote dans LabVIEW?

Assistant de configuration des E/S instruments

- ⇒ A utiliser lorsqu'il n'existe aucun pilote pour un instrument
- ⇒ Construction pas à pas de code GPIB ou VISA



⇒ Étapes successives

Sélectionner
l'instrument

Ecrire

Requérir
et analyser

Lire
et analyser

Pilotes d'instruments

Pilotes d'instruments préconfigurés

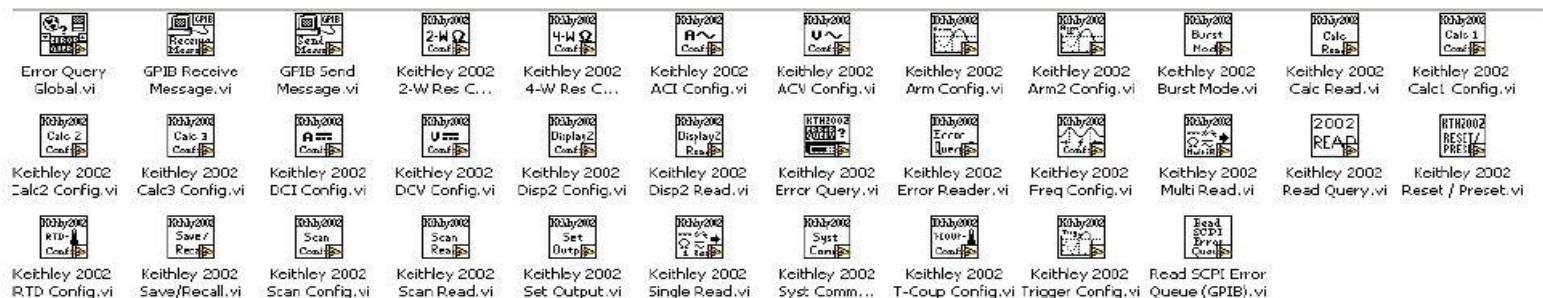
Drivers d'instruments

Les drivers d'instruments sont des bibliothèques de VI qui permettent de contrôler un instrument programmable.

Accessible sur le site de NI à l'adresse suivante : www.ni.com/idnet

Ou par labVIEW à partir de l'arborescence suivante : « Outils > Instrumentation > Base de drivers d'instruments ».

Exemple : Multimètre KEITHLEY modèle 2002 :



Communication avec un instrument:

Créé un VI qui permet:

- de récupérer toutes les 5 secondes pendant 1 minute la tension mesurée par un multimètre KEITHLEY à votre disposition via le port IEEE;
 - de calculer la valeur moyenne de ces mesures;
 - d'enregistrer les résultats sous forme de fichier texte.

A vous de programmer ...