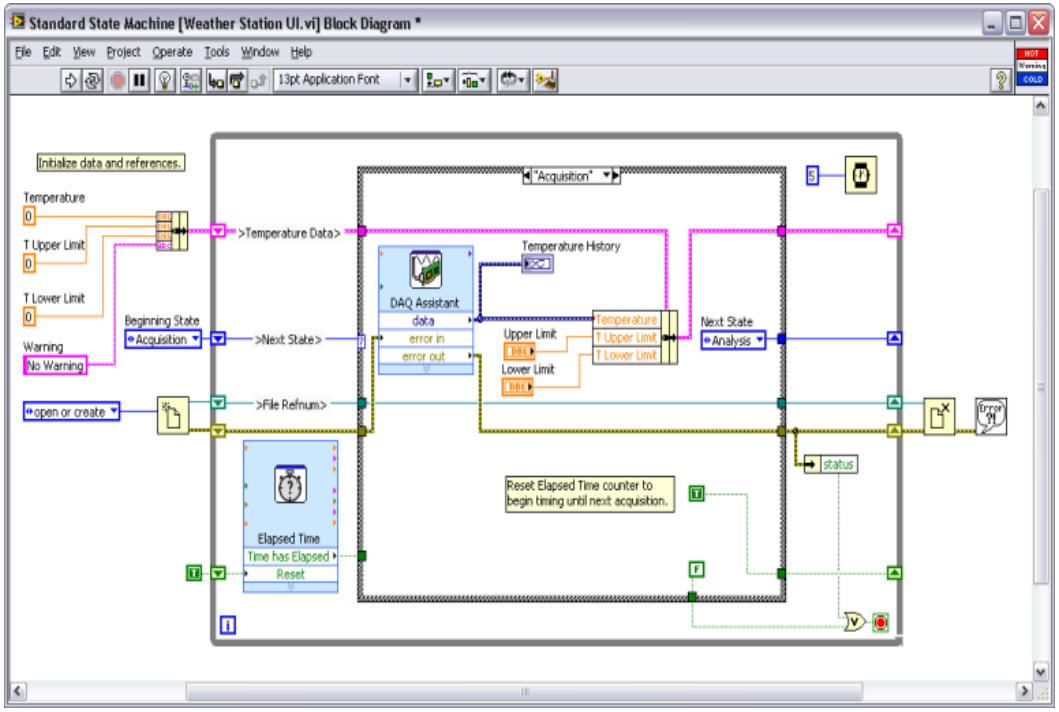
Lecture 1
Introduction to the Concept of Virtual Instrument (VI) and Graphical Programming Language

吳文中

What Is LabVIEW?

— A graphical programming environment used to develop sophisticated measurement, test, and control systems.

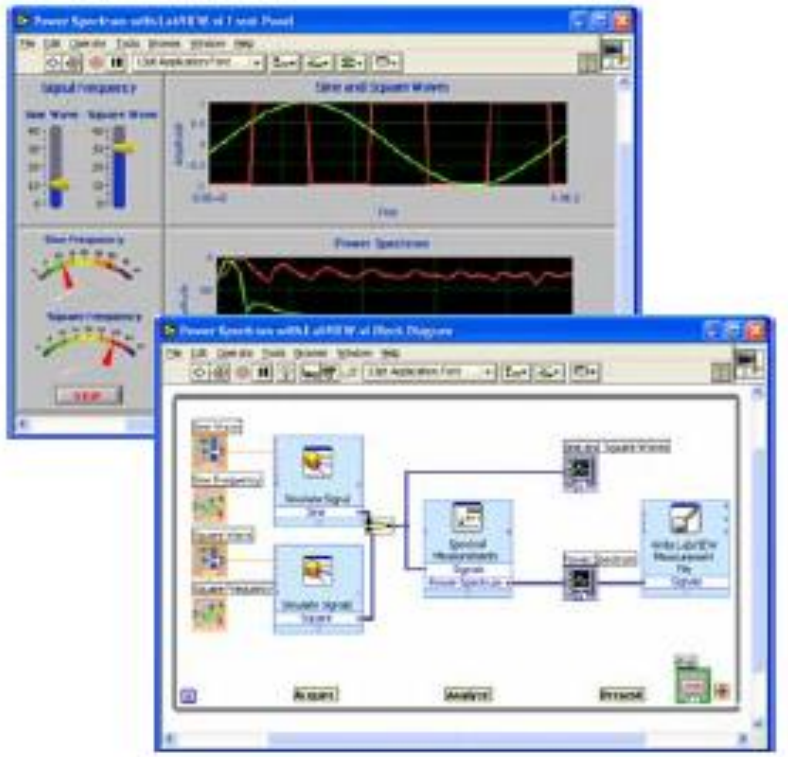


LabVIEW:

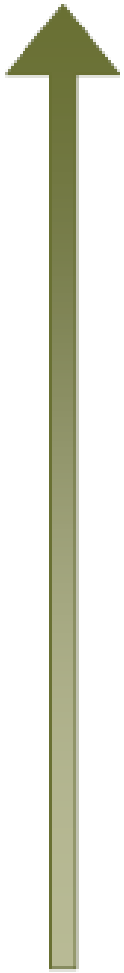
- Interfaces with wide variety of hardware
- Scales across different targets and OSs
- Provides built-in analysis libraries

Graphical Language

- **Why NI LabVIEW?**
  - **Graphical Programming for Test, Measurement, and Control**
  - Rapid application development with Express VIs and easy-to-use graphical environment
  - Interactive measurement assistants and powerful redesigned DAQ interface for connecting to all types of I/O
  - Expanded targeting options from Real-Time to FPGA
- **Competitors? HP(Agilent) VEE**

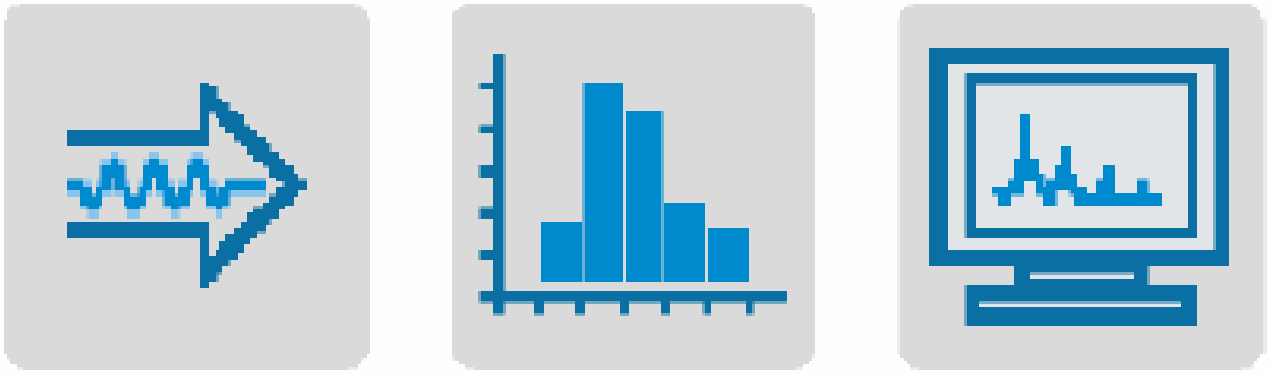


A History of LabVIEW



- **Modern LabVIEW ~**

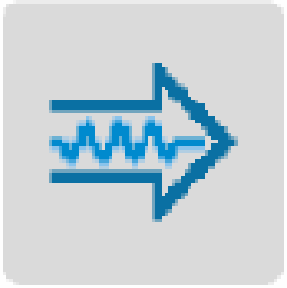
August 2005	• LabVIEW 8 Project-based developments, distributed intelligent
May 2003	• LabVIEW 7 Express VIs, I/O Assistants, FPGA/PDA targets
January 2002	• LabVIEW 6.1 Enhanced networking capabilities, analysis
August 2000	• LabVIEW 6/Internet-ready measurement intelligence
March 1998	• LabVIEW 5.0 ActiveX, Multithreading
February 1996	• LabVIEW 4.0 Added professional tools, improved debugging
August 1993	• LabVIEW 3.0 Multiplatform version of LabVIEW
September 1992	• LabVIEW for Windows
January 1990	• LabVIEW 2.0 for Macintosh
October 1986	• LabVIEW 1.0 for Macintosh
April 1983	• LabVIEW project begins



Acquire, Analyze, and Present

Nearly all test, measurement, and control applications can be divided into 3 main components: the ability to *acquire*, *analyze*, and *present* data. LabVIEW is the easiest, most powerful tool for acquiring, analyzing, and presenting real-world data.

Acquire with LabVIEW

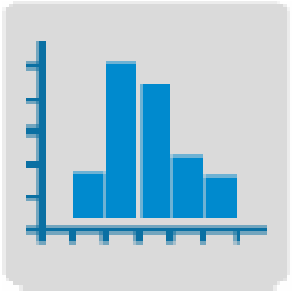


LabVIEW is tightly integrated with all NI hardware, in addition to connecting to thousands of I/O devices from hundreds of different vendors.

LabVIEW can acquire data using the following devices and more:

- GPIB, Serial, Ethernet, VXI, PXI Instruments
- Data Acquisition (DAQ)
- PCI eXtensions for Instrumentation (PXI)
- Image Acquisition (IMAQ)
- Motion Control
- Real-Time (RT) PXI
- PLC (through OPC Server)
- PDA
- Modular Instruments

Analyze with LabVIEW



Powerful measurement analysis is built in to the LabVIEW development environment.

LabVIEW includes the following tools to help you analyze your data:

- More than 400 measurement analysis functions for Differential Equations, Optimization, Curve Fitting, Calculus, Linear Algebra, Statistics, etc.
- 12 new Express VIs specifically designed for measurement analysis, including filtering and spectral analysis
- Signal Processing VIs for Filtering, Windowing, Transforms, Peak Detection, Harmonic Analysis, Spectrum Analysis, etc.

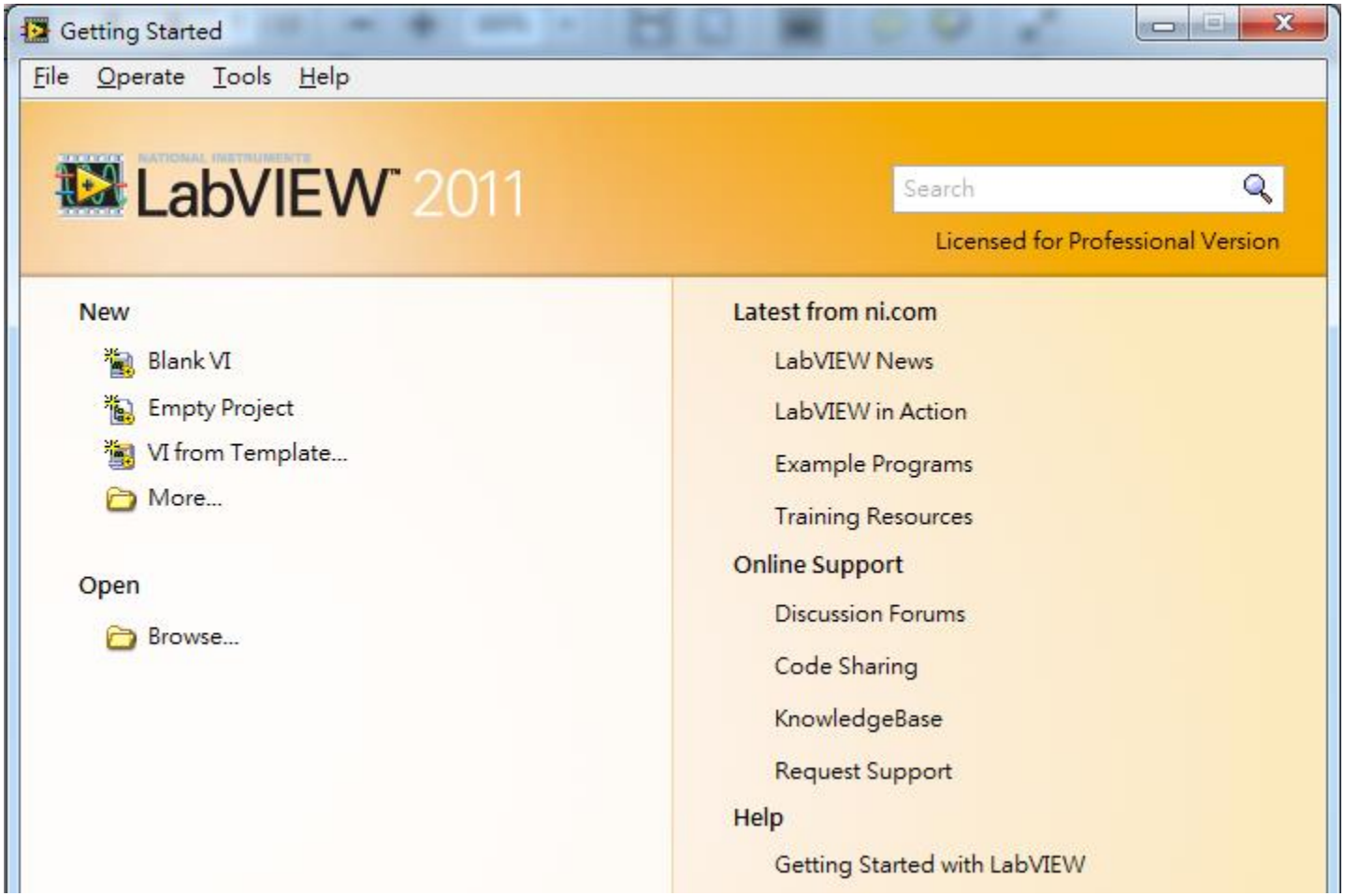
Present with LabVIEW

Presentation with LabVIEW can be done on your PC or over a network, or you can take advantage of additional applications such as DIAdem.

LabVIEW includes the following tools to help you present your data:

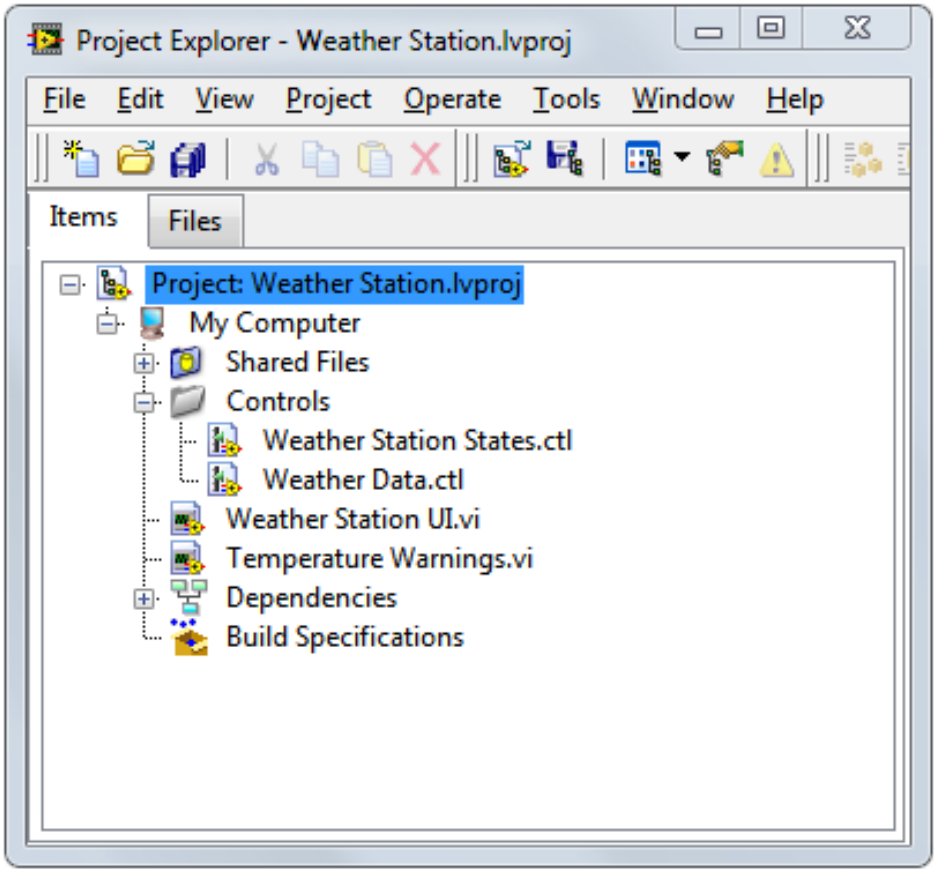
- On your machine — Graphs, Charts, Tables, Gauges, Meters, Tanks, 3D Controls, Picture Control, 3D Graphs (Windows Only), Report Generation (Windows Only)
- Over the Internet — Web Publishing Tools, Datasocket (Windows Only), TCP/IP, VI Server, Remote Panels, Email
- Enterprise Connectivity Toolset — SQL Tools (Databases), Internet Tools (FTP, Telnet, HTML)

LabVIEW Dialog Box



Project Explorer

- Find, access, and organize project files
- Prevent, detect, and resolve incorrect links
- Deploy or download files to targets
- Manage code for build options
  - Executables, installers, and zip files
- Integrate with source code control providers



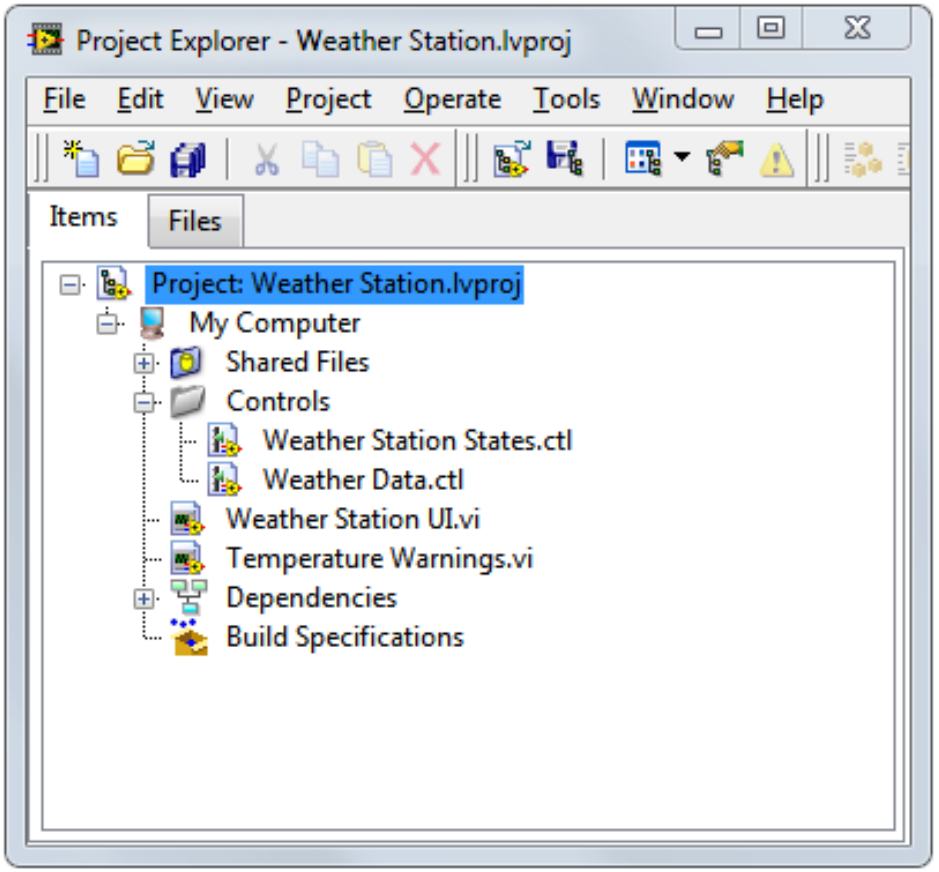
LabVIEW Files

•Common LabVIEW file extensions:

LabVIEW project —.lvproj

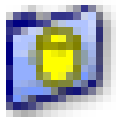
Virtual instrument (VI) — .vi

Custom control — .ctl

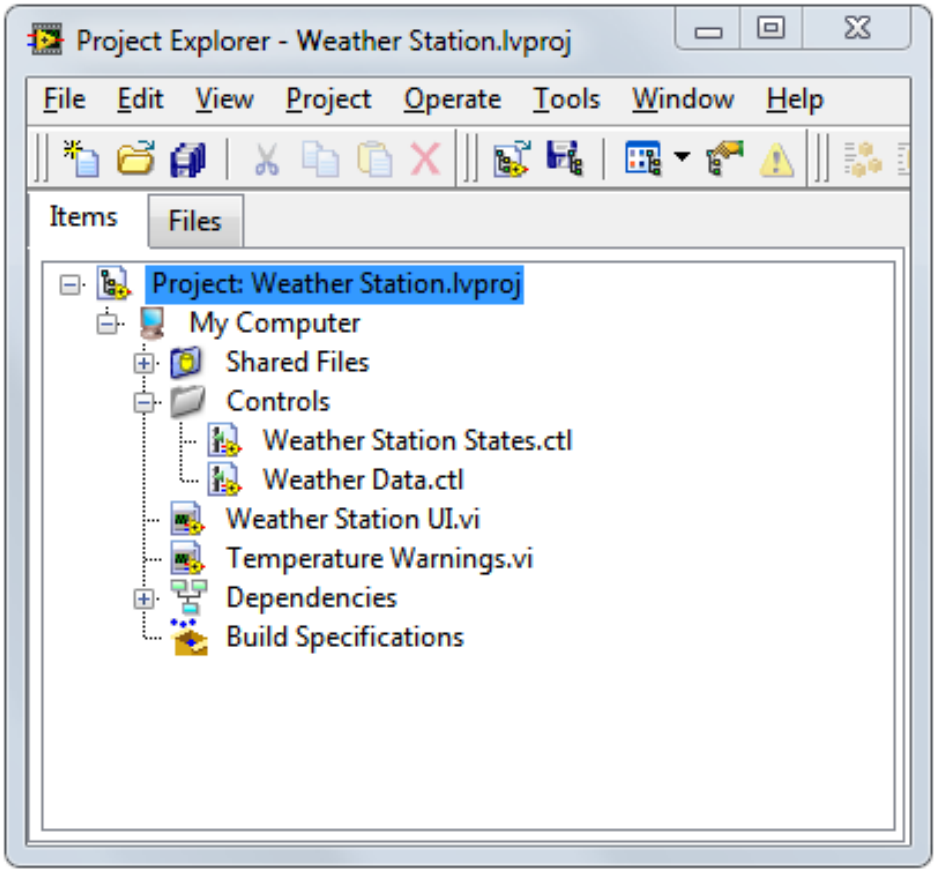


Adding Folders to a Project

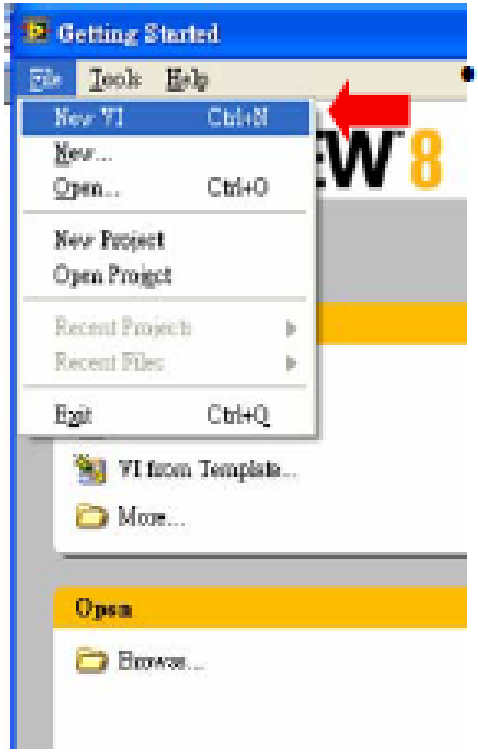
- Virtual folder
  - Organizes project items and does not represent files on disk



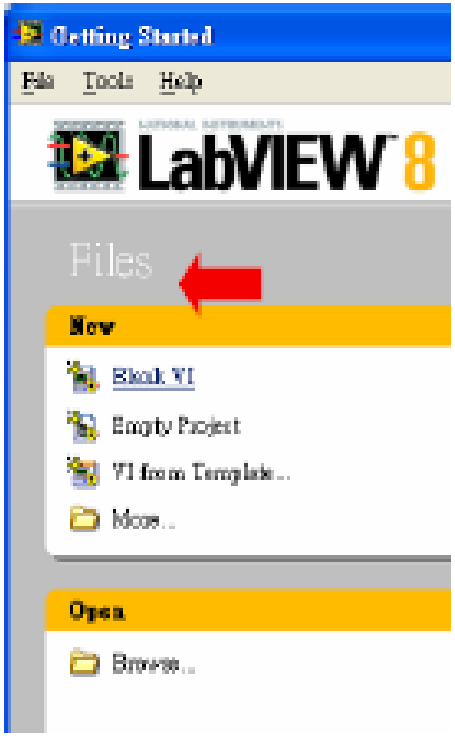
- Auto-populating folder
  - Adds a directory on disk to the project
  - LabVIEW continuously monitors and updates the folder according to changes made in the project and on disk



Creating a new VI



- Blank VI or
  File»New VI to open a blank VI

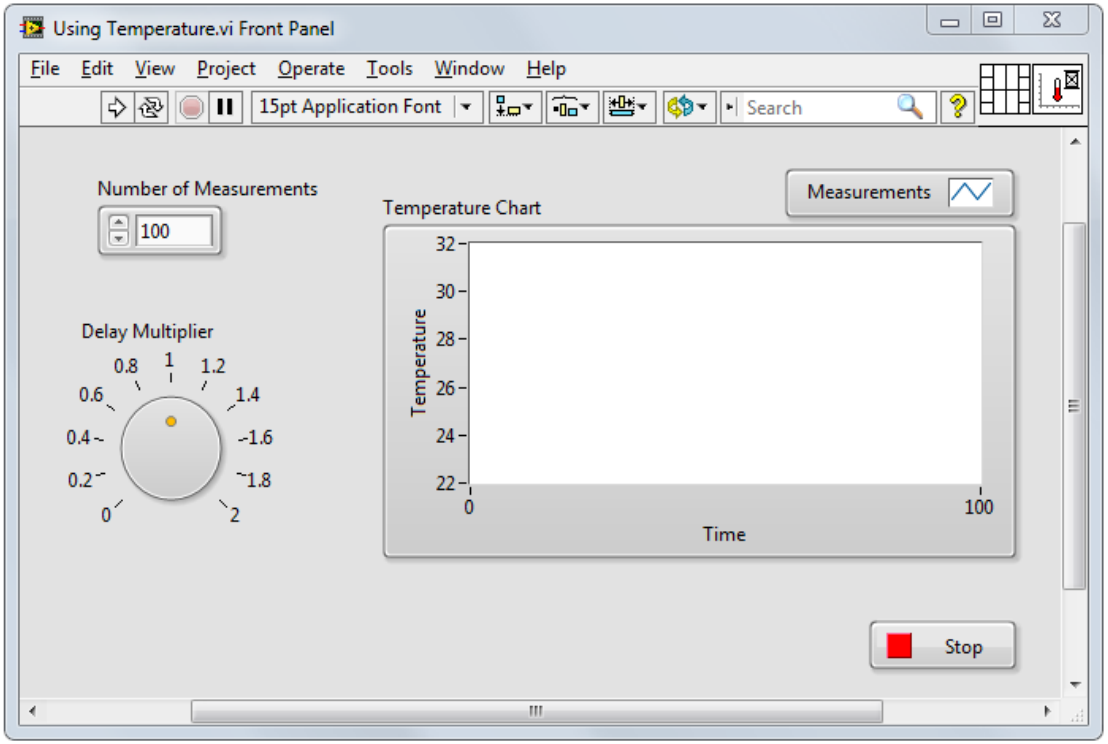


- VI from Template... or File»New... to open the Template Browser ,and configure a VI template, global variable, control, etc…

Parts of a VI – Front Panel

Front Panel – User interface for the VI

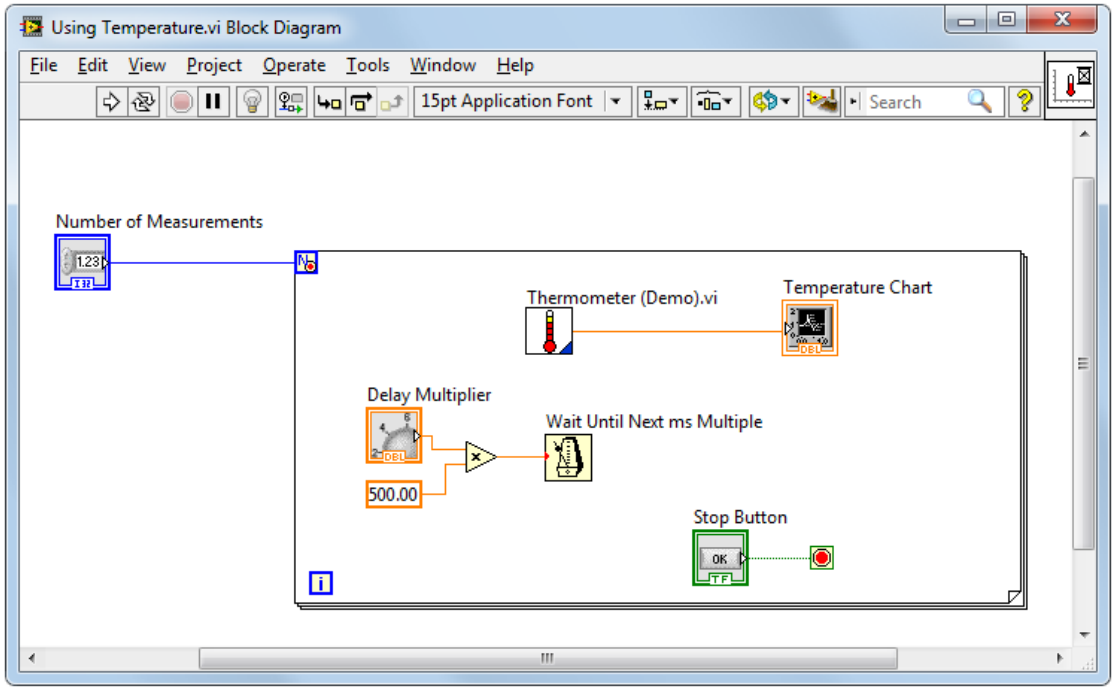
You build the front panel with controls (inputs) and indicators (outputs).



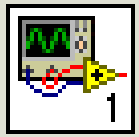
Parts of a VI – Block Diagram

Block Diagram – Contains the graphical source code

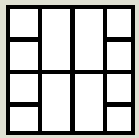
Front panel objects appear as terminals on the block diagram.



Parts of a VI – Icon/Connector Pane



Icon – Graphical representation of a VI



Connector Pane – Map of the inputs and outputs of a VI

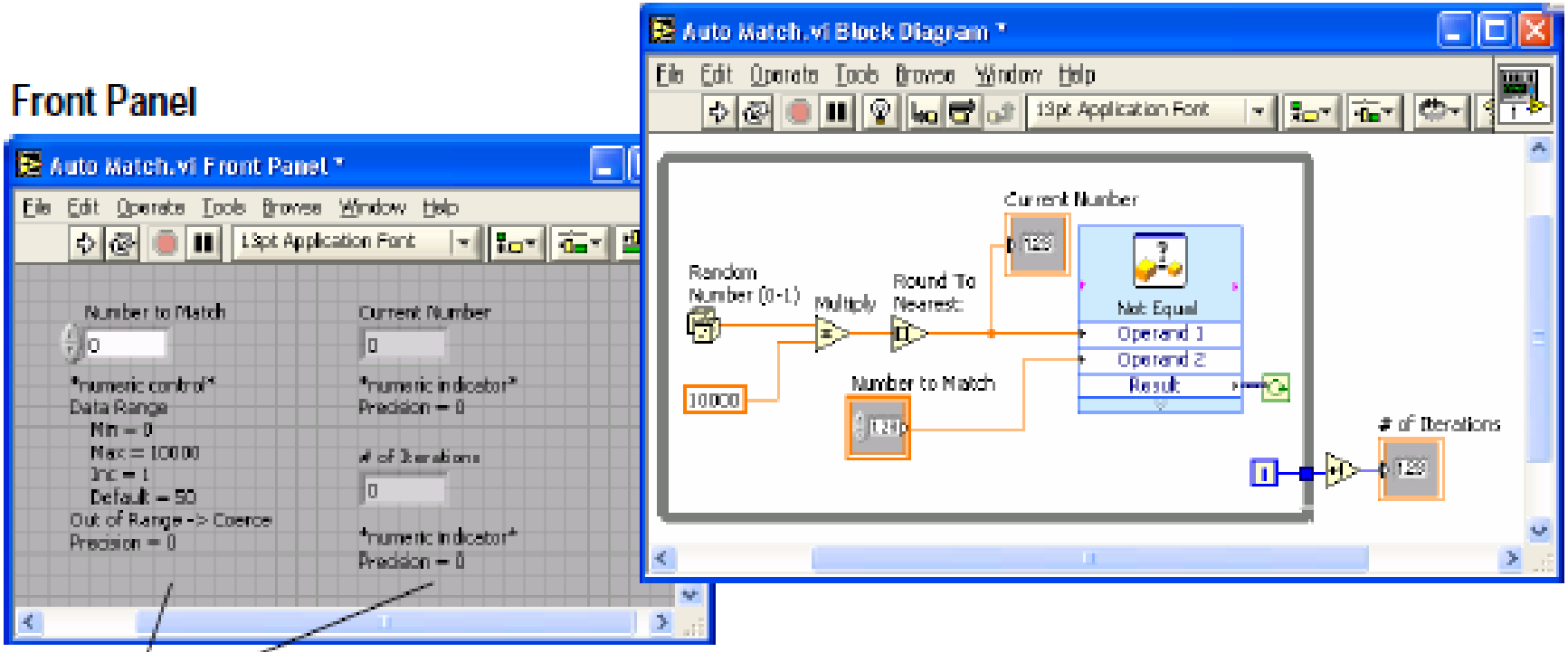
Icons and connector panes are necessary to use a VI as a subVI.

- A subVI is a VI that appears on the block diagram of another VI.
- A subVI is similar to a subroutine or function in a text-based programming language.

Course Exercises Notation

Block Diagram

Front Panel



** Comments **
Do not enter these

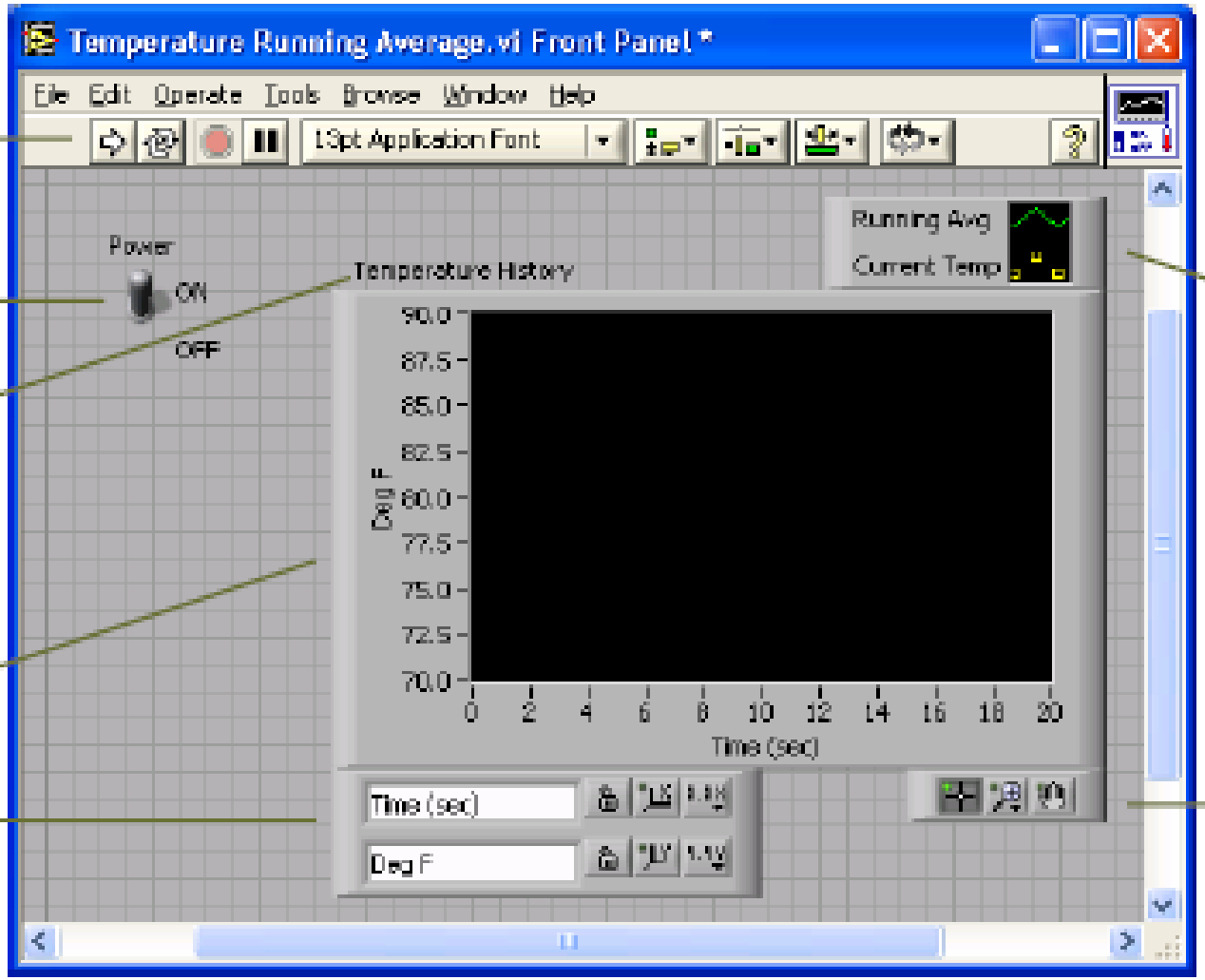
Name of object

Where it can be found

Random Number (0-1) (Numeric subpalette)
Returns a random number between 0 and 1.

Description

Front Panel Window



Front Panel Toolbar

Boolean Control

Waveform Graph Owned Label

Waveform Graph

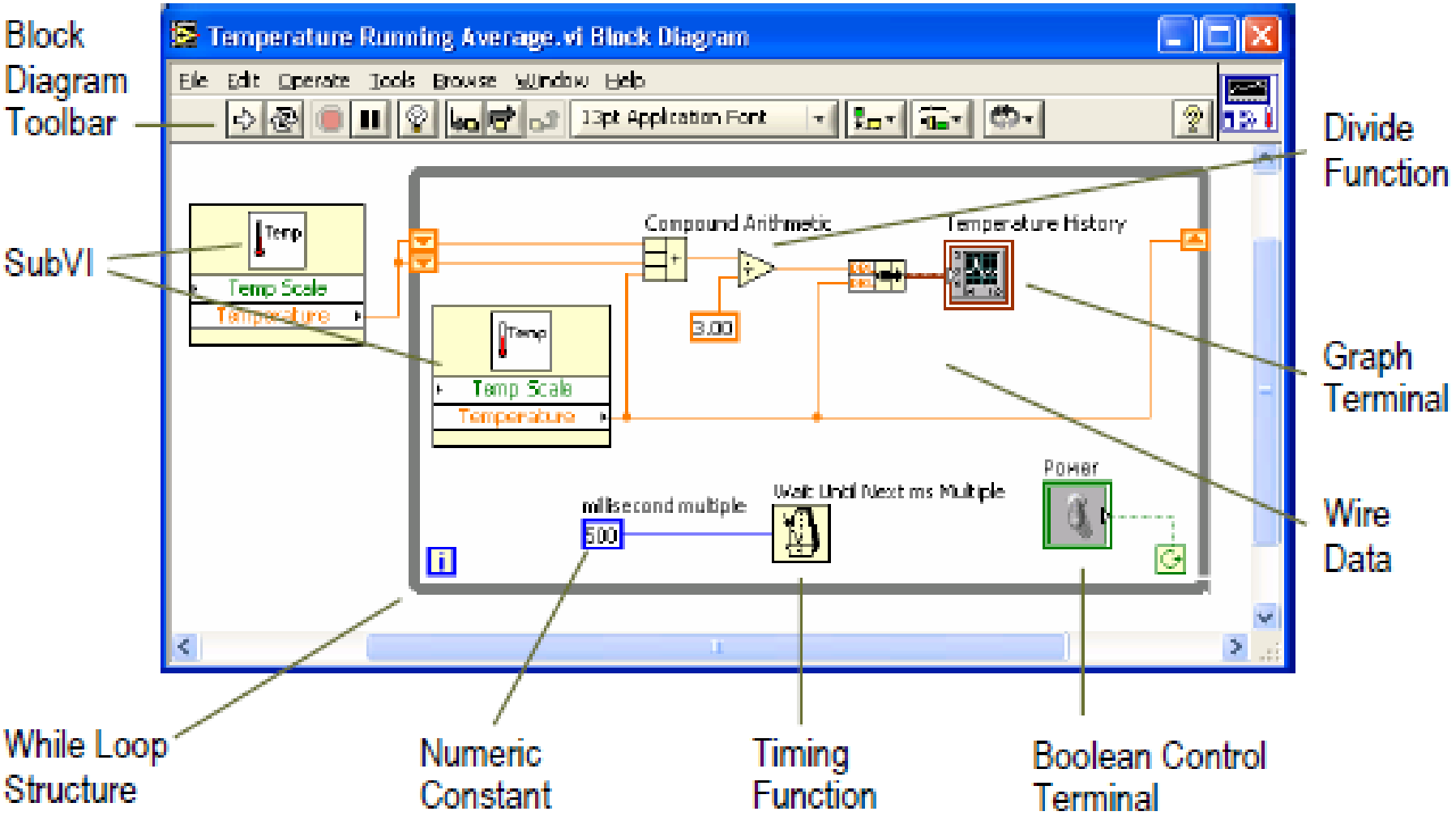
Plot Legend

Icon

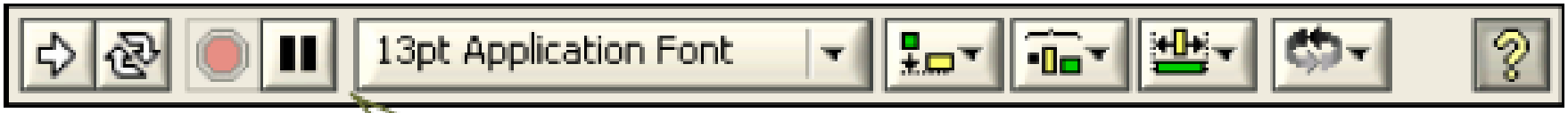
Graph Legend

Scale Legend

Block Diagram Window



Front Panel and Block Diagram Toolbars

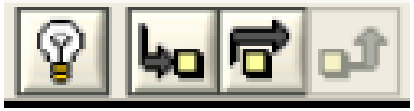


Run button

Continuous Run button

Abort button

Pause/Continue button



Additional Buttons on the Block Diagram Toolbar

- Execution Highlighting button
- Step Into button
- Step Over button
- Step Out button

Font ring

Alignment ring

Distribution ring

Resize ring

Reorder ring

Context Help Button

Warning indicator

Enter button

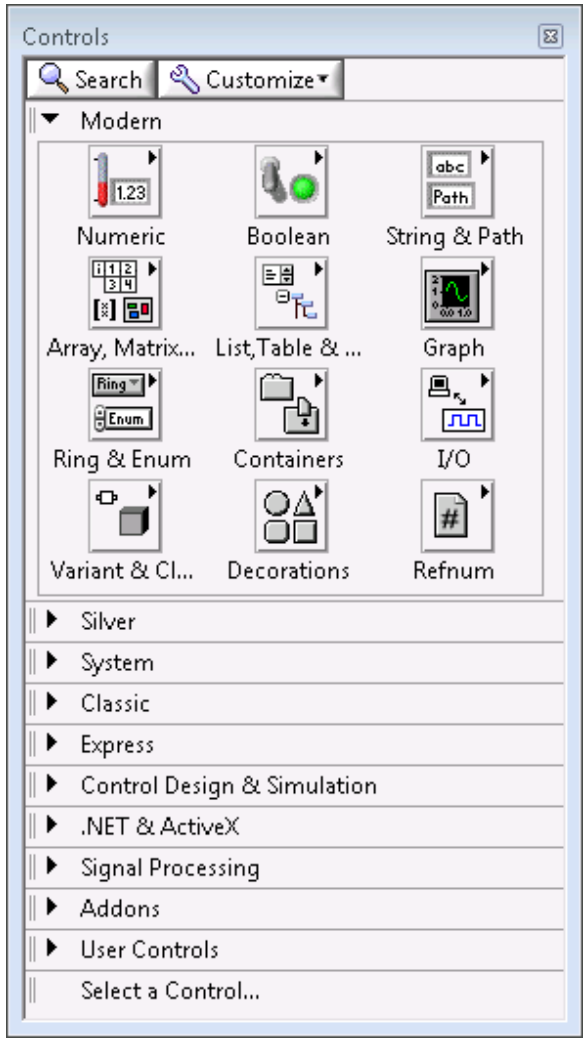
Broken Run button

Tools Palette

- LabVIEW automatically selects the tool needed
- Available on the front panel and the block diagram
- A tool is a special operating mode of the mouse cursor
- Use the tools to operate and modify front panel and block diagram objects
- To show the tools palette, select
  View » Tools Palette

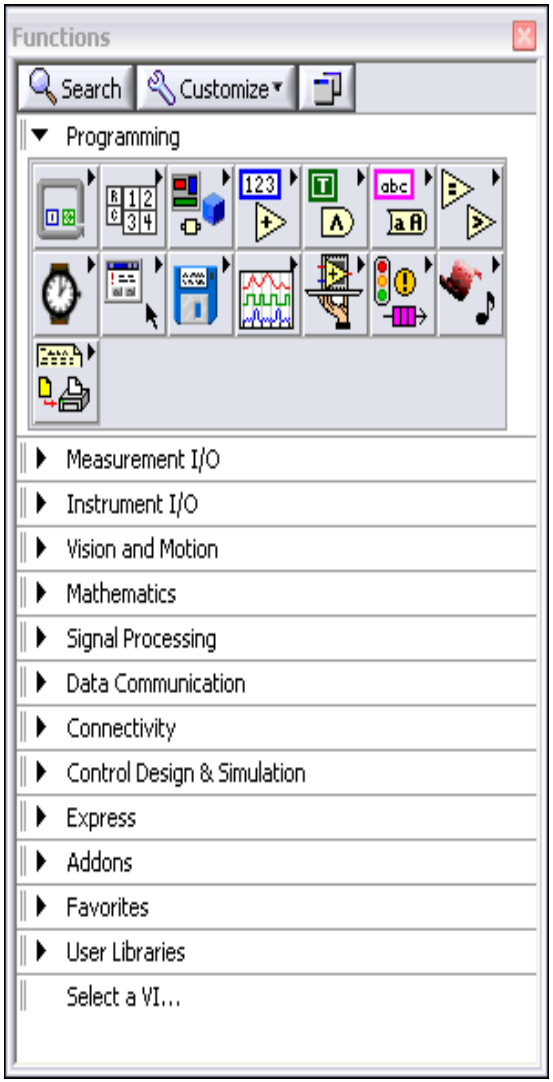
Controls Palette

- Contains the controls and indicators you use to create the front panel.
- Navigate the subpalettes or use the **Search** button to search the Controls palette.



Functions Palette

- Contains the VIs, functions, and constants you use to create the block diagram.
- Navigate the subpalettes or use the **Search** button to search the Functions palette.



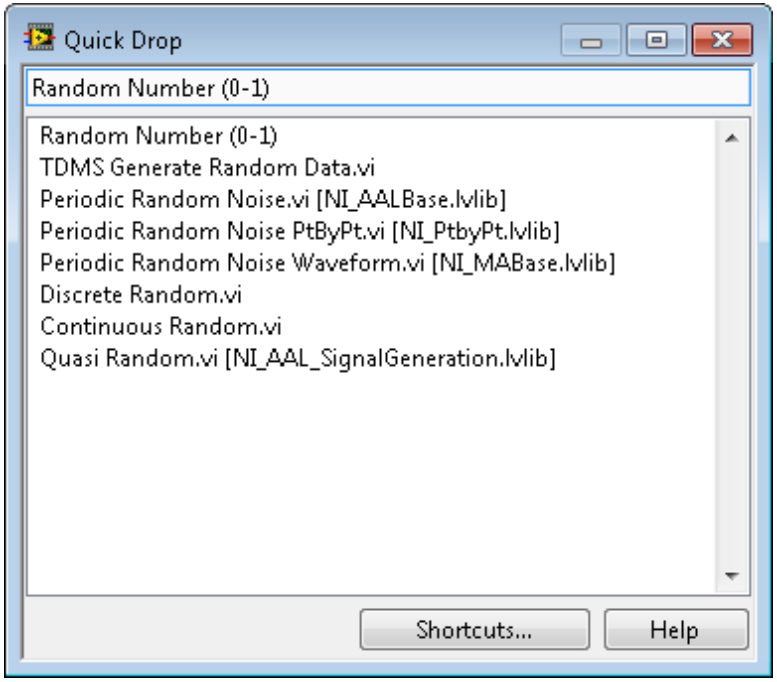
Searching for Controls, VIs, and Functions

Ways to find controls, VIs, and functions:

- Search or navigate the palettes.
  - Controls palette
  - Functions palette
- Search by name of object.
  - Quick Drop dialog box
- Search palettes, *LabVIEW Help*, and `ni.com`.
  - Search text box in toolbar

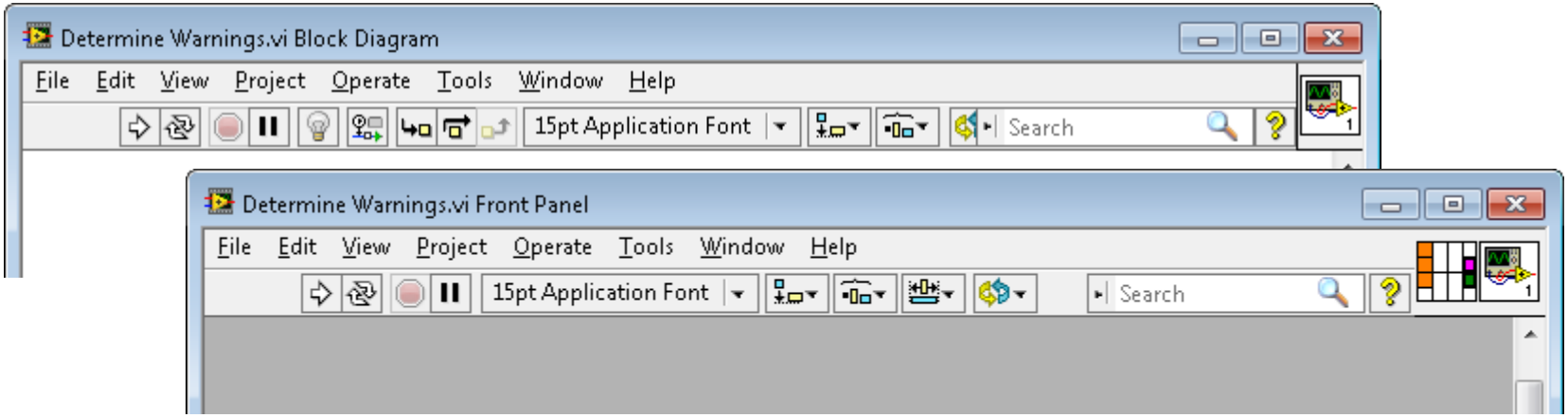
Searching with Quick Drop

- Lets you quickly find controls, functions, VIs, and other items by name.
- Press the <Ctrl-Space> keys to display the Quick Drop dialog box.



Global Search

Use the Search bar in the top right of the front panel and block diagram windows to search palettes, *LabVIEW Help*, and `ni.com`.



Summary

- Virtual instruments (VIs) have three main parts — the front panel, the block diagram, and the icon and connector pane
- The front panel is the user interface of a LabVIEW program and the block diagram is the executable code
- The block diagram contains the graphical source code composed of nodes, terminals, and wires
- Floating Palettes: Tools Palette, Controls Palette (only when Front Panel Window is active), and Functions Palette (only when Block Diagram Window is active)
- There are help utilities including the Context Help Window and LabVIEW Help