

Geräte: 1 Computer mit LabView, 1 Bewegungsmesswandler, grosses Stativ, 1 Schwingfeder, 1 Schwingmasse, Faden

Auftrag:

Der s-Ausgang des Bewegungsmesswandlers ist mit dem Eingang des Digital-Analog-Wandlers verbunden. Ein Messprogramm in LabView zeichnet den Vorgang auf einem Plotter auf. Analysieren Sie das Messprogramm (s. Arbeitsblatt).

LabView

LabVIEW is a program development application that uses a graphical programming language, G, to create programs in block diagram form.

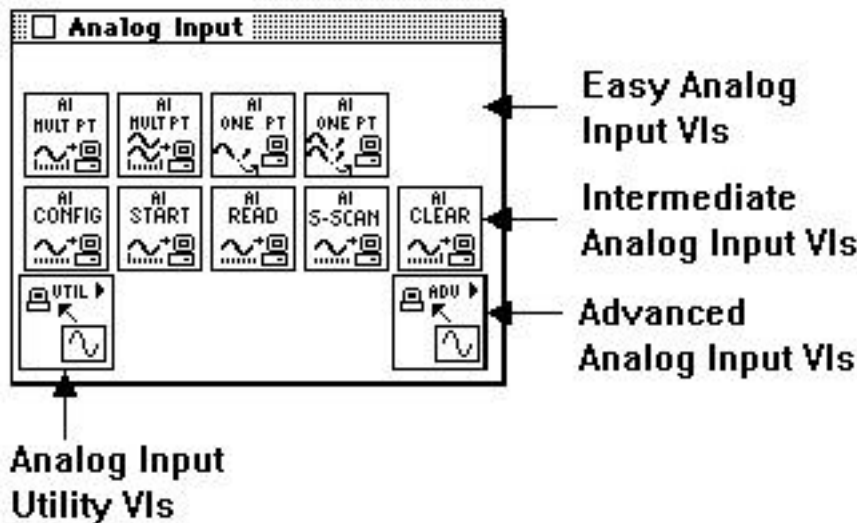
What is a Virtual Instrument?

A *virtual instrument* (VI) is a program in the graphical programming language G. Virtual instrument front panels often have a user interface similar to physical instruments. G also has built-in functions that are similar to VIs, but do not have front panels or block diagrams as VIs do. Function icons always have a yellow background.

Analog Input

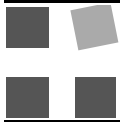
This chapter explains the Analog Input VIs and shows where you can find them in LabVIEW.

There are four classes of Analog Input VIs found in the **Analog Input** palette. The Easy Analog Input VIs, Intermediate Analog Input VIs, Analog Input Utility VIs, and Advanced Analog Input VIs. The following illustration shows these different VI classes. Click on the label next the VIs for a brief explanation of the class of VIs or click on an actual icon to go to that particular VI.



For examples of how to use the analog input VIs, see examples\daq\anlogin.llb or examples\daq\anlog_io.llb.

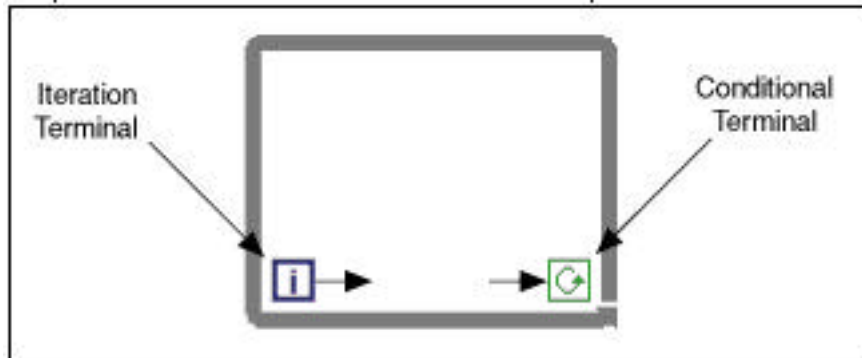
Note: Use only the inputs that you need on each VI. LabVIEW sets all unwired inputs to their default values. Many of the DAQ function inputs are optional and do not appear in the Simple Diagram Help window. These inputs typically specify rarely-used options. If an input is required, your VI wiring remains “broken” until a value is wired to the input. Required inputs appear in bold in the Help window, recommended inputs appear in plain text, and optional inputs are in gray text. The default values for inputs appear in parentheses beside the input name in the Help window.



While Loops

A While Loop is a structure that repeats a section of code until a condition is met. It is comparable to a Do Loop or a Repeat-Until loop in traditional programming language.

The While Loop, shown in the following illustration, is a resizable box you use to execute the diagram inside it until the Boolean value passed to the conditional terminal (an input terminal) is FALSE. The VI checks the conditional terminal at the end of each iteration; therefore, the While Loop always executes at least once. The iteration terminal is an output numeric terminal that outputs the number of times the loop has executed. However, the iteration count always starts at zero, so if the loop runs once, the iteration terminal outputs 0.



The While Loop is equivalent to the following pseudocode:

Do

Execute Diagram Inside the Loop (which sets the condition)

While Condition is TRUE

Hilfe

Sie können weitere Anmerkungen unter Online reference im Help-Menü erhalten.
Konsultieren sie auch die bereitgestellten Handbücher.