

LabView Guide Book

INDEX

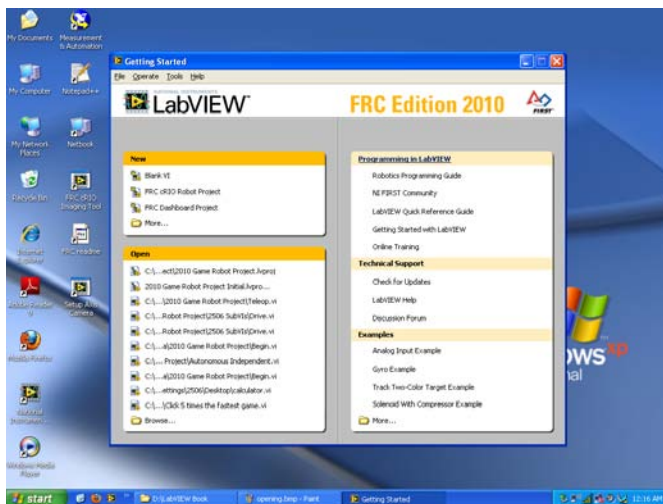
GETTING STARTED.....	
DATA TYPES.....	
CONTROLS TOOL.....	
CONTROLS and INDICATORS.....	
BASIC PROGRAMMING.....	
SELECT STATEMENT.....	
WHILE LOOP.....	
ARRAYS and CLUSTERS.....	
ENUM.....	
CASE STRUCTURE.....	

Getting Started

Click this Icon on the desktop to open Labview

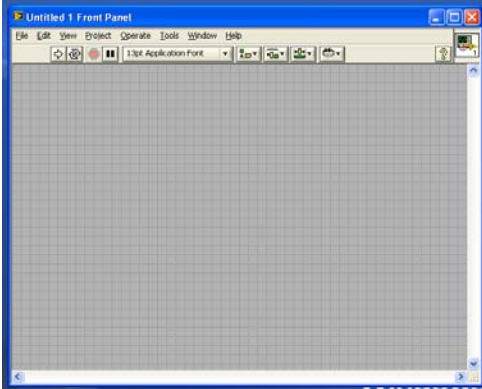


After clicking the icon this will pop up. This is a loading screen and you should just let it finish loading

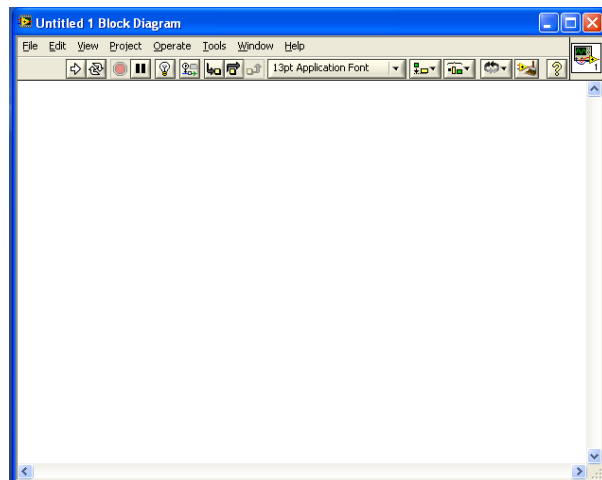


After the program loads it will show this screen. From this screen you can start a new blank VI or a new robot program. Below the new options is a list of programs that have been worked on recently.

This is the Front Panel. This area is used as an interface for a user to use your program (Note that the driver of the robot will not be using this to run the robot).



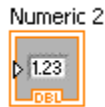
This is where all of the programming is do. Programming in this language is used with icons and “wires”.



Data Types

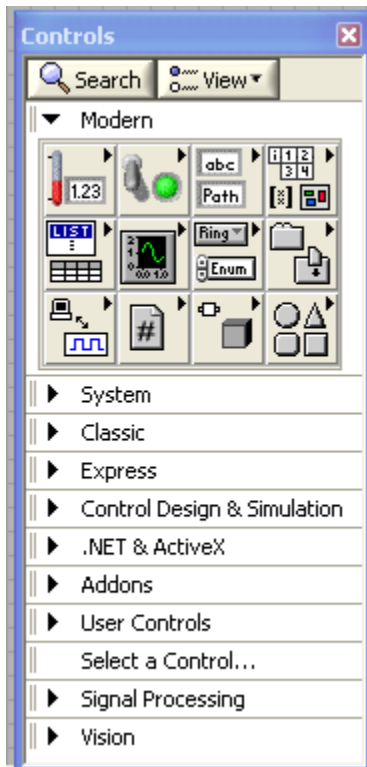


This is a Control form of an icon. This lets the user input data and outputs it to the program. All icons that have an output will have an arrow on the right side of it.



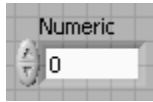
This is the icon for an Indicator. It takes out puts and displays them or shows them on the Front Panel. Anything that needs an input will have an arrow on the left.

Controls Tool



By right clicking on the Front Panel the Controls Menu will pop up. The menu is where all of the Controls and Indicator are located.(Note by having the mouse sit on top of an icon it will give you the name of the icon) To put a Control or Indicator on to the Front Panel click the icon and drag it to the desired place on the Front Panel and then release the click. To bring up the Controls Menu again just right click again.

Controls and Indicators

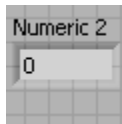


A Numeric Control will look something like this on the Front Panel. The Numeric Control lets you input number values into the program

Numeric



The Numeric control will show up on the Block Diagram.



A Numeric Indicator will display the output of a numerical input on the Front Panel.

Numeric 2



The Numeric Indicator will show up on the Block Diagram.

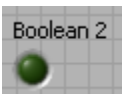


Boolean Controls act as to let the program make decisions based on a single comparison or a series of comparisons. (a single Boolean represents 1/0 or true/false)

Boolean



The Boolean Control will look like this on the Block Diagram.

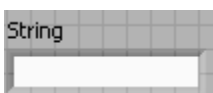


The Boolean Indicator will show the output of a Boolean Control after it has run though the comparison.

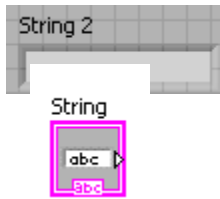
Boolean 2



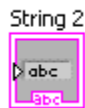
The Boolean Indicator shows up on the Block Diagram like this.



String Controls let you input Word based information into the program
The String Control will show up like this on the Block Diagram



The String Indicator will show the output of a string value on the Front Panel



The String Indicator will show up like this on the Block Diagram

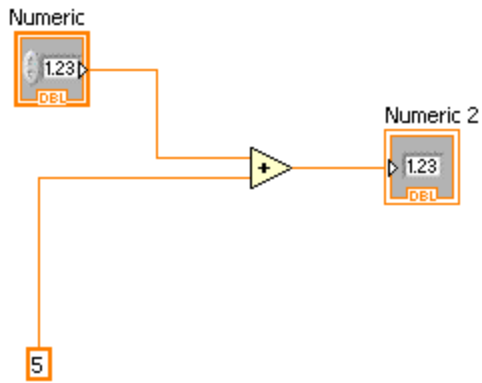
Basic Programming



this is how to connect anything in Labview by using a “Wire”. The Wire connects the Input arrow to the Output arrow.

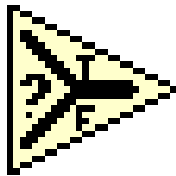


By holding the mouse over an arrow this symbol will appear. Then you click once and drag it to what you want to connect and click again, then the wire will appear and they will be connected.

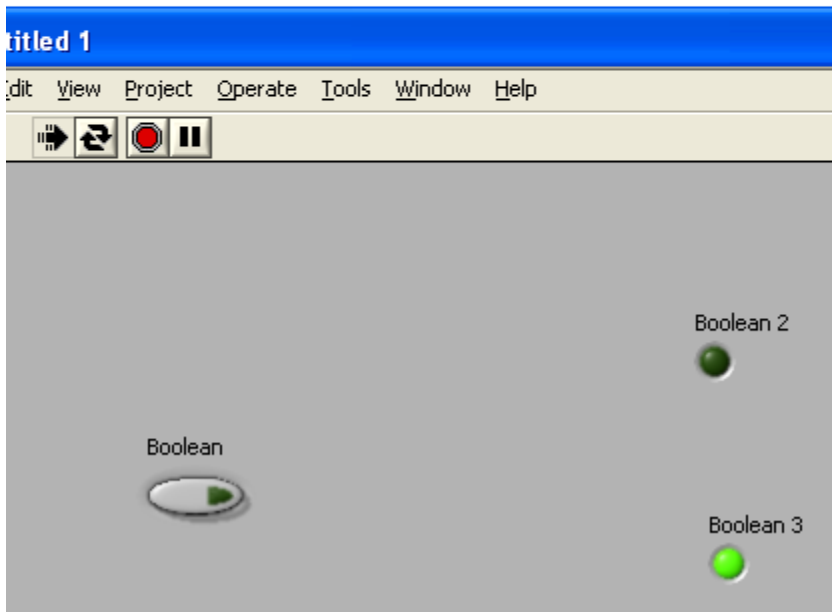
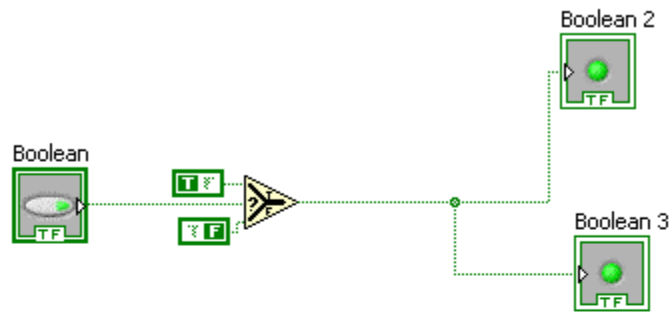


Here is a basic program. There is a Numeric Indicator, a Numeric Control, and a Numeric Constant. In the middle is the Logic or what the program does. Most programs are set-up by having the Inputs on the left side, the Logic in the middle, and the Outputs on the right. In this program it take a Numeric Control's Input and add it to a Numeric Constant that is set to five and then display the Output in the Numeric Indicator.

Select Statement

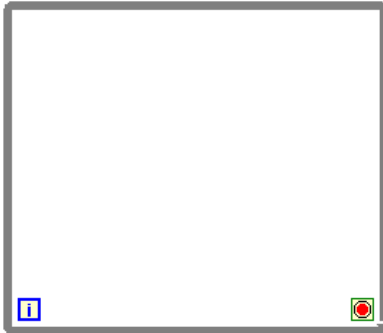


This is a Select statement. It is used to Output one of two opinions based on a statement (an example is if $X > 2$. X being a Numeric Control).



You Wire the Output of the select statement that you want if the statement is True goes into the T or the top Input. The Statement that you want to use as a comparison goes into the middle Input.). You Wire the Output of the select statement that you want if the statement is False to the F or bottom Input.

While Loop

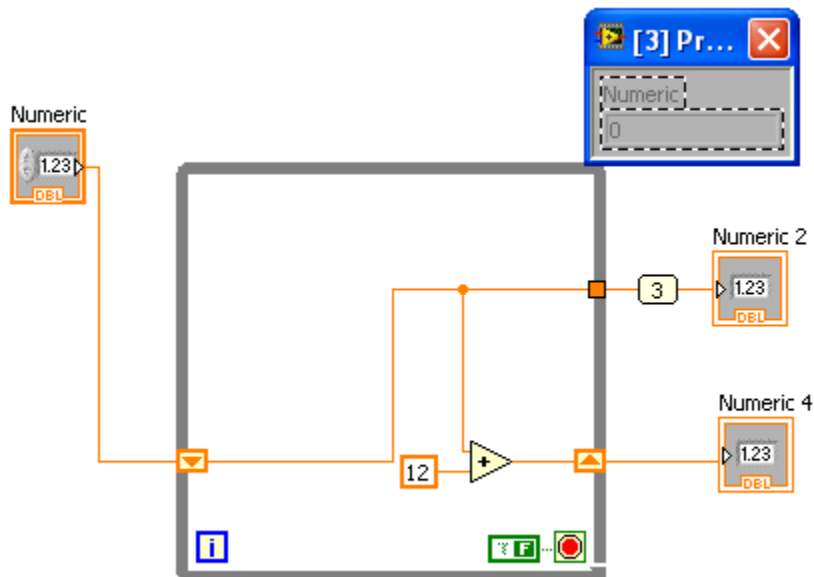


This is a While Loop. The function of a While Loop is to repeat a step over a certain number of times, infinitely, or until a condition is met.

A While Loop will only run what is inside of the gray lined box.

The 'i' surrounded by a blue box takes a Numeric Constant to tell how many times to run the loop. (If you want to have it run infinitely or to run until a condition is met, this box should remain blank.)

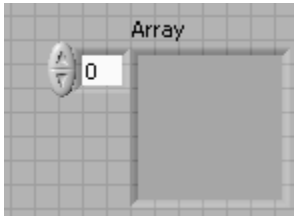
The stop sign sets the condition in which the loop will stop (the Output of a Select statement can go here or a comparison)



To have Data from the While Loop carry over into the next step in the While Loop a Shift Register is needed to do so. To Make a Shift Register click the box on the While Loop (the box will look like the one left of the 3.) you will get the option to replace with Shift Register select it and it will change.

Shift Register takes the value from the arrow on the right of the While Loop (facing up) and sends it to the arrow on the left of the While Loop (facing down). Then it will take the value and run it through the program again.

Arrays and Clusters



Arrays are a grouping of the same type of Data (all String, Numeric, or Boolean). To set the Data type of the Array, drag a Control or Constant into the array. The number in the white box is an Array Index Number that refers to the Control in sequence order. On the Block Diagram you can create an Array Constant That the User or Programmer can not change while the program is running.

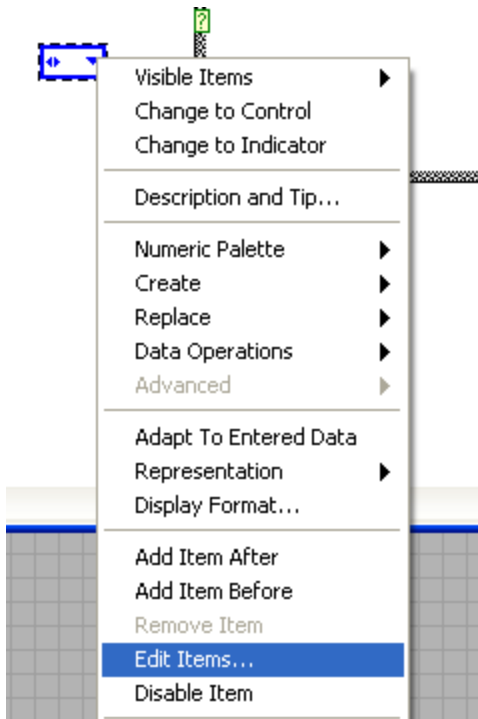


Clusters are used to put more than one Data Type in. (Ex: one String and two Numeric) To get the cluster to have the different Data Types simply drag them inside the box. Clusters are mainly used to keep the Front Panel organized.

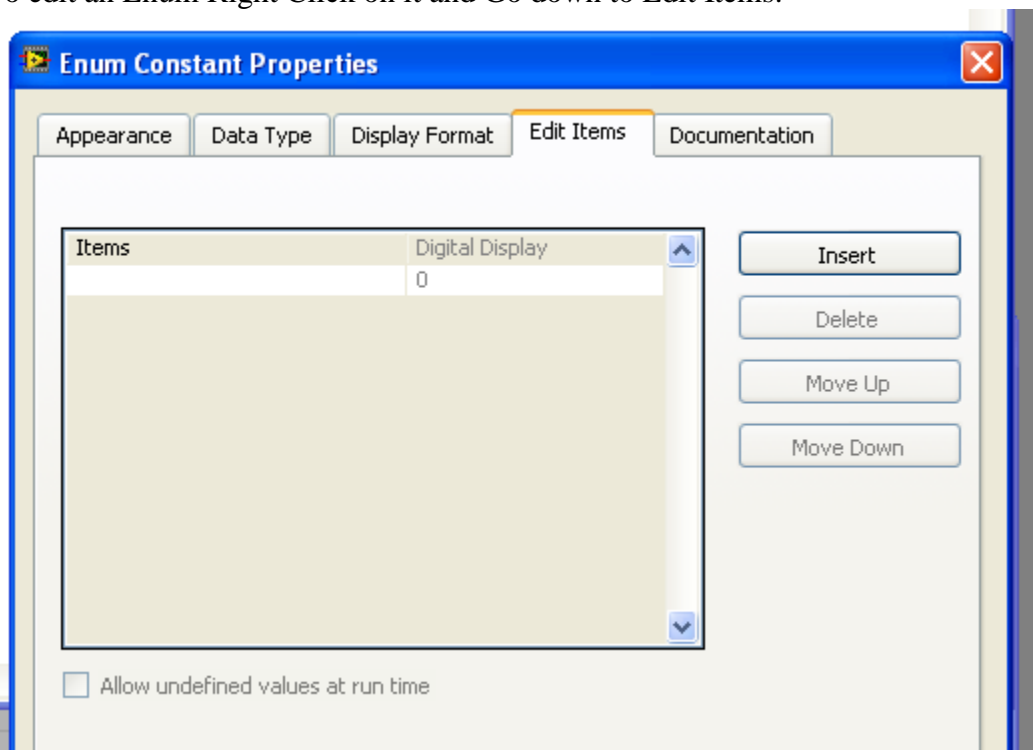
Enum



This is an Enum Constant. An Enum assigns a name to a number. The purpose of an Enum is to give a meaningful name to a number that is used in a Case Structure. Enum Constant is made on the Block Diagram.

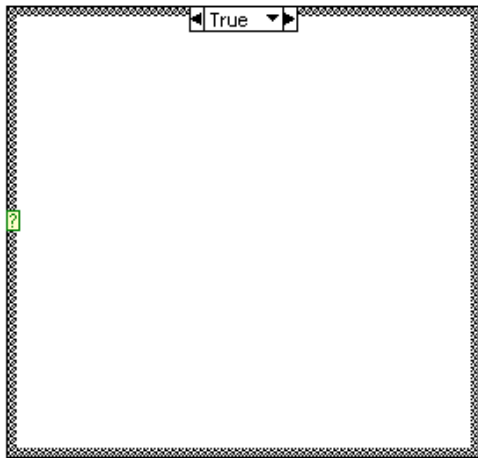


To edit an Enum Right Click on it and Go down to Edit Items.



Then a screen will pop-up click Edit Items Tab if it is not already selected. To add an item to the list click Insert. To get rid of an item select the item and click Delete. The numbers are assigned in the order from top to bottom. 0 is the Default or the starting State of the Case Structure.

Case Structure

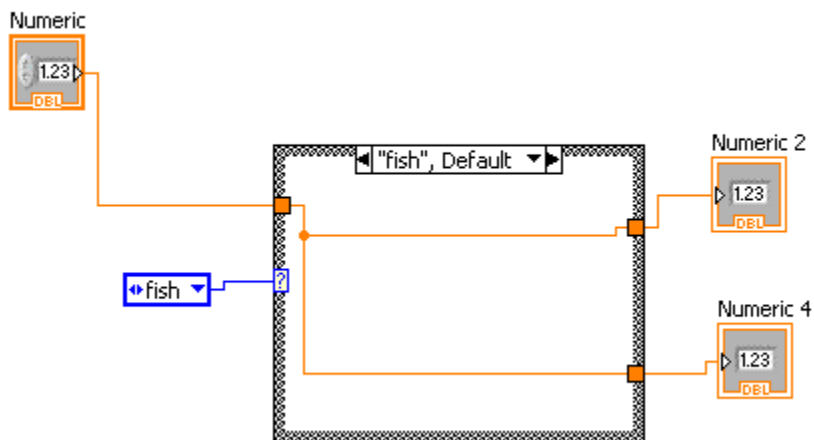


This is a Case Structure. The function of a Case Structure is to make a selection based on given information. Case Structures a Boolean Control, Numeric Control, Enum Control, or and Enum Consent.

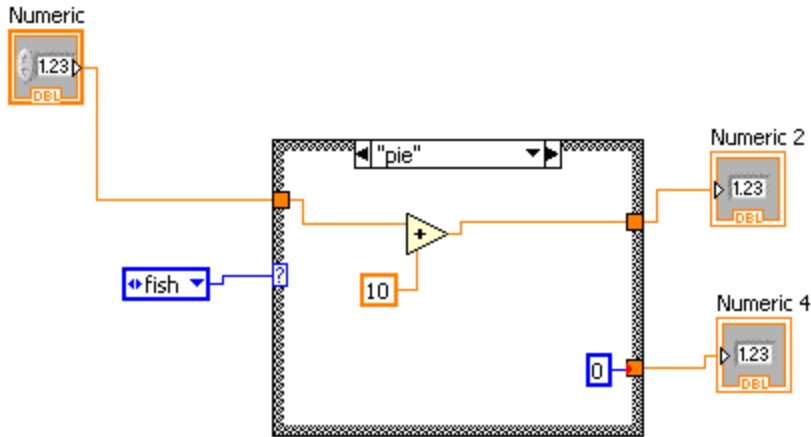
Boolean Control Works just as a Select works

The Numeric control would change what state the Selector is in. (the Selector is the box located on top of the Case Structure.)

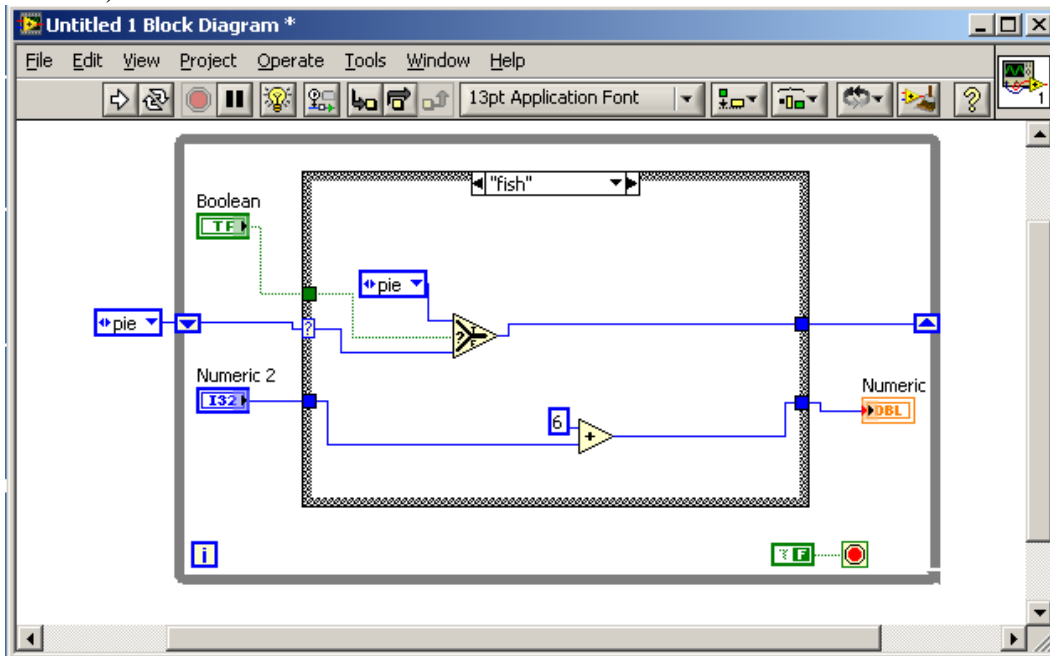
Enum Controls would work the same as Numeric Controls.



Here is an example of a Case Structure that uses Enum Consents. In this first State called "Fish" which is the Default as shown in the Selector. In this State the program just displays the input two times in two Different Outputs.



Here is the second State in the Case Structure. Here the Input is taken and add to 10. It is only displayed in one Output while, the other Output is Displaying 0. (in each State every Output need an Input. That is why Numeric 4 is displaying 0 instead of just leaving it blank.)



Here is a full working Case Structure that runs infinitely due to the use of a While Loop. First the only thing that we want on the While Loop is the Enum to Control the Case Structure. The Enum runs through a Shift Register so that each time it runs through a step it carries that value over. The Controls that their values don't want to be carried over to the next step should be on the inside of the While Loop. To get the Enum to change the state of the Case Structure an Enum Constant outside of the While Loop. Then connect it to the question mark on the Case Structure and going through the While Loop on the left side. Make a Select Statement. Wire the Question Mark to the False Input and make a Enum Constant of the state you want to go into. Use a Boolean Data Type in the statement part (a button or a comparison). Wire the Output to all the way to the right side of the While Loop. Then Change the two blue Squares in to Shift Registers. Repeat for each of the States.