How to Use Case Structures Tutorial

Functions -> Programming -> Structures -> Case Structure

A Case Structure is a branching control mechanism that allows different executions depending on the value of the label. The Case Structure is analogous to the Case block in Java or C++ in which, based on what case value the input variable matched, the case structure will choose the correct cases for execution.

To create a Case Structure, open the Functions palette and open the Structures palette . Under Structures select the Case Structure .

Once you have selected the Case Structure in the palette, your cursor will change to a box with a question mark in it. Click and drag a box of whatever size you will need for the functions that you're going to put in the case structure and you will get a Case Structure as shown in Figure 1.

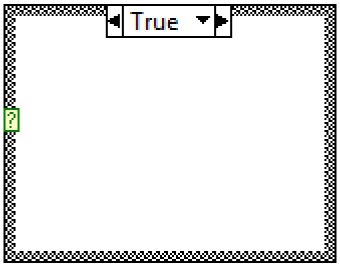

Figure 1

You can resize the Case Structure later by selecting it and dragging the handles.

Now that the Case Structure has been created, you will notice that on the left side, the Case Structure has an input called a case selector . The case selector is the value used to determine which case to execute and is roughly equivalent to the following pseudocode.

Given respective case selector values A, B, and C, with a case selector value X,

```
if(X == A)
        <execute the case for A>
if(X == B)
        <execute the case for B>
if(X == C)
        <execute the case for C>
else
        <execute the default case>
```

In addition, you will also notice that the case selector defaults to an undefined data type so the user must wire the desired data type to the terminal of the case selector.

You will also notice that at the top of the Case Structure there is menu indicated in Figure 2.

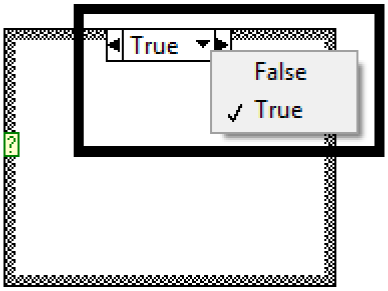

Figure 2

This menu is where the case values should be defined. Each case has its own function design in the case structure which can be displayed by selecting the desired case in the menu.

After you have defined the data type for the cases, you may need to add cases to the Case Structure. To do this, Right click on the Case Structure and select the "Add Case Before" or "Add Case After" option as shown in Figure 1.

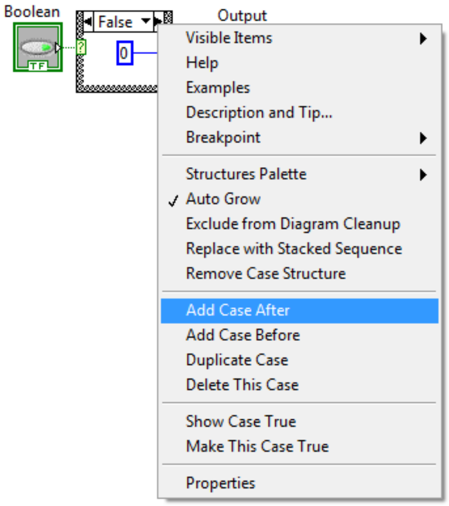

Figure 1

If the input value matches any of the values in the menu, the Case Structure will only execute the functions for that case.

For example, consider the following Case Structure which outputs 1 if the input value is True (Figure 2) and 0 if the input value is False (Figure 3).

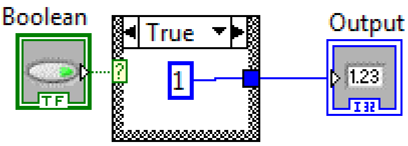

Figure 2

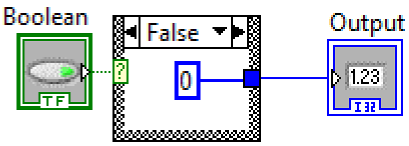

Figure 1

You will notice that there are two different function layouts embedded in this structure. It is important to define the function for each case before running a VI with a Case Structure in it (LabView compiler will throw an error if you don't).

To handle cases that do not match any of the inputs, the Case Structure will automatically have a "Default" option listed as one of the pre-programmed options which must be completed for all data types except for the Boolean data type.

For example, consider the following Case Structure shown in Figures 3, 4, and 5, that takes in a numeric input value. The case structure will output 2 (1 + 1) if the input is 1, 10 (2 x 5) if the input is 2, and 0 if the input is anything else (the 0 case and Default case).

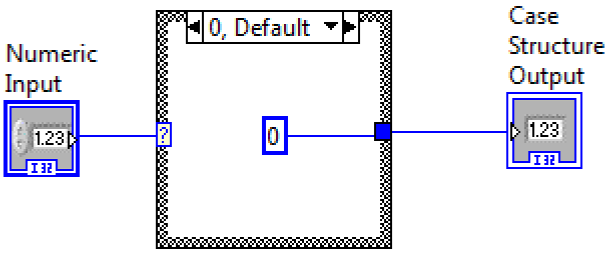

Figure 4

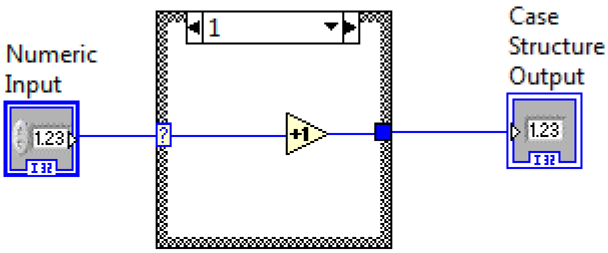

Figure 5

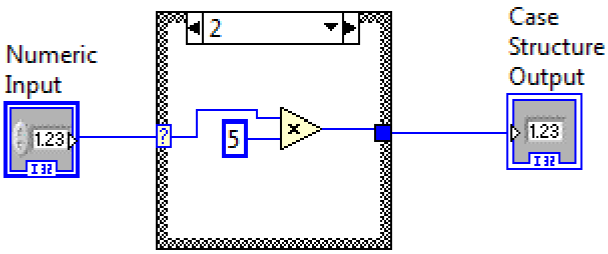

Figure 6