Transformations Build

This guide walks the reader through building a Virtual Instrument, VI, using LabVIEW. The VI graphs a function, $y = f(x)$, and a transformed function, $y = a * f(b(x - d)) + c$, where the parameters *a* through *d* control the vertical dilation, horizontal dilation, horizontal translation, and vertical translation of the graph.

Joules Webb
Education Service Center, Region 20
julianne.webb@esc20.net
(210) 370-5497

Warm regards,

Eric Mann

Transformation
T-STEM 2013
New Directions | New Pathways | New Outcomes

Building Transformations VI

This VI captures the relationship between algebraic and graphic representations of transformations of functions. For a given parent function, $y = f(x)$, this VI graphs a transformed function,

$$y = a\,f\big(b(x-d)\big) + c,$$

where the parameters *a* through *d* control the vertical dilation, horizontal dilation, horizontal translation, and vertical translation of the graph. We will include controls for each of the parameters as well as indicators for the graph and the transformed equation.

Many common functions can be used for the Parent Function, $y = f(x)$. For example, x, x^2, x^3, abs(x), sqrt(x), sin(x), cos(x), 2^x, 1/x, exp(x), ln(x) can be used. Let's start with adding a control for entering the Parent Function and controls for the transformation parameters.

Transforming the Parent Function Equation

1) Create a new VI by selecting **File»New VI** or pressing **<Ctrl-N>**

The <Ctrl> key in keyboard shortcuts corresponds to the **(Mac OS)** <Command> key or **(Linux)** <Alt> key.

2) Add a **String Control** to the front panel
    a. Right-click on the front panel to open the **Controls** palette
    b. Tack the palette down by clicking on the thumb tack in the upper-left corner of the palette
    c. Select the **String Control** from the **Express»Text Controls** palette
    d. Click on the front panel to place the control
    e. Rename the control, "Parent Function"
3) Add two numeric controls for the vertical transformation parameters, *a* and *c*
    a. Select the **Knob** from **Express»Numeric Controls** palette
    b. Click on the front panel to place the control, and rename the control, "Vertical Dilation, a"
    c. Select the **Vertical Pointer Slide** from **Express»Numeric Controls** palette
    d. Click on the front panel to place the control next to the knob, and rename the control, "Vertical Translation, c"
    e. Set the minimum value for both controls to –10 by double-clicking the 0 and typing the desired value
    f. Right-click on both controls and select **Visible Items»Digital Display**

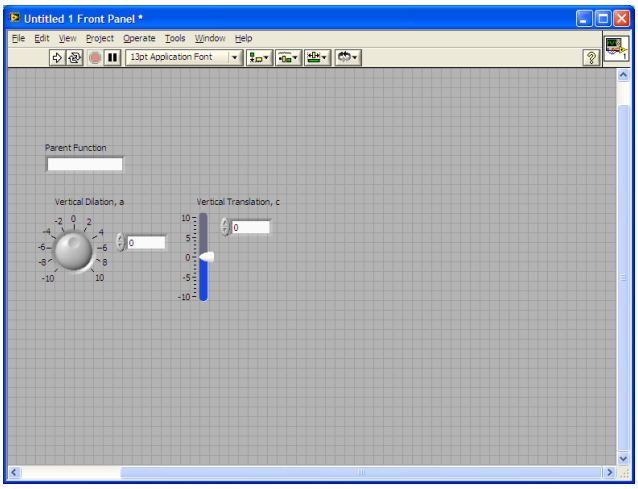

Figure 1. The beginnings of the front panel

4) Below the controls for the vertical parameters, add two controls for the horizontal transformation parameters, *b* and *d*
    a. Select the **Knob** from **Express»Numeric Controls** palette
    b. Click on the front panel to place the control, and rename the control, "Horizontal Dilation, b"
    c. Select the **Horizontal Pointer Slide** from **Express»Numeric Controls** palette
    d. Click on the front panel to place the control next to the knob, and rename the control, "Horizontal Translation, d"
    e. Set the minimum value for both controls to –10
    f. Right-click on each control and select **Visible Items»Digital Display**
5) Save your VI
    a. Select **File»Save** or press **<Ctrl-S>**
    b. Call the VI **myTransformations**

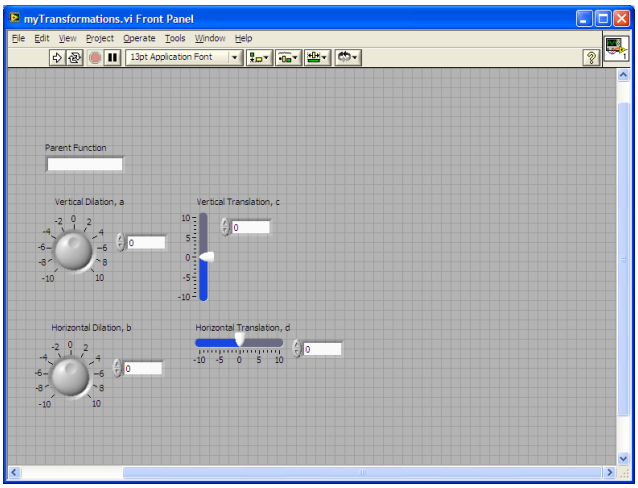

Figure 2. Controls placed on front panel

Next, you will use the values from the numeric controls to transform the equation of the parent function. The **myTransformFnString** VI takes a function of x, $f(x)$, as a string, i.e. text, along with numeric parameters $a - d$ to produce the transformed function, $af\left(b\left(x-d\right)\right)+c$, as a string. The **myTransformFnString** VI is available in the \Transformations Build folder.

6) Select **Window»Show Block Diagram** or press **<Ctrl-E>** to view the block diagram
7) Arrange the icons corresponding to the controls in the order they were added to the VI and as shown below

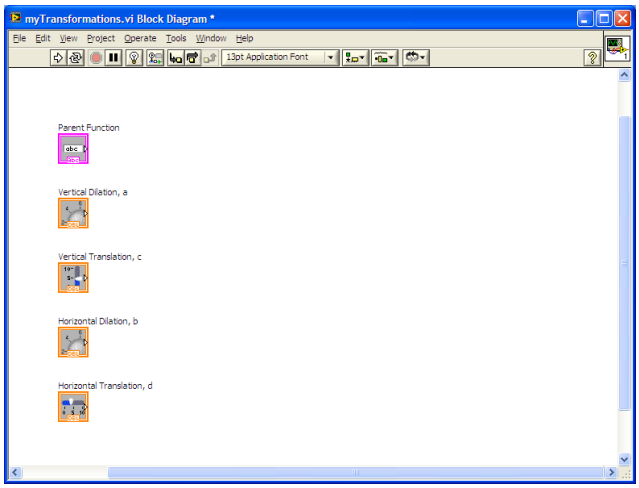

Figure 3. Controls in block diagram

8) Right-click on the block diagram to open the **Functions** palette
9) Tack the palette down by clicking on the thumb tack in the upper-left corner of the palette
10) Add **myTransformFnString** to the block diagram
    a. From the **Functions** palette, choose **Select a VI…**
    b. Browse to **\Transformations Build** folder and select the **myTransformFnString.vi** and click **OK**

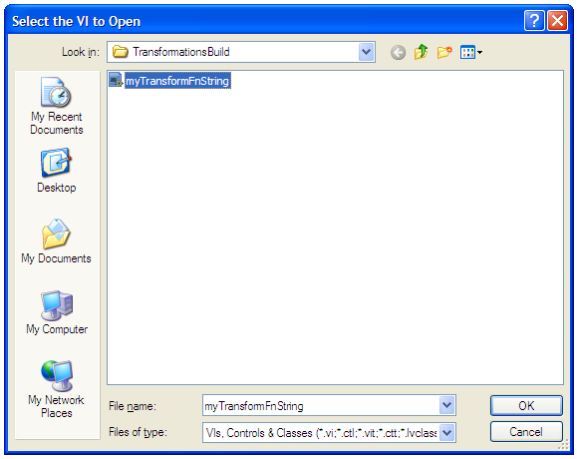

Figure 4. Select the VI to add it to the block diagram

c. Click on the block diagram to place the VI

11) Use the Wiring tool to connect each of the five controls to its corresponding input terminal on the **myTransformFnString** VI

To wire the controls to the subVI, hover the mouse over the terminal of each control until the wiring tool appears. Then, hold down the left mouse button and drag the cursor to the destination terminal on the subVI. When the terminal appears, you can release the mouse button and the wire will be connected.

12) Add an indicator to display the transformed function
    a. Hove the cursor over the **Transformed Function** output terminal of **myTransformFnString** until the Wiring tool appears
    b. Right-click and select **Create»Indicator** from the context menu
    c. The indicator will be named "Transformed Function"
13) Save your VI, **<Ctrl-S>**

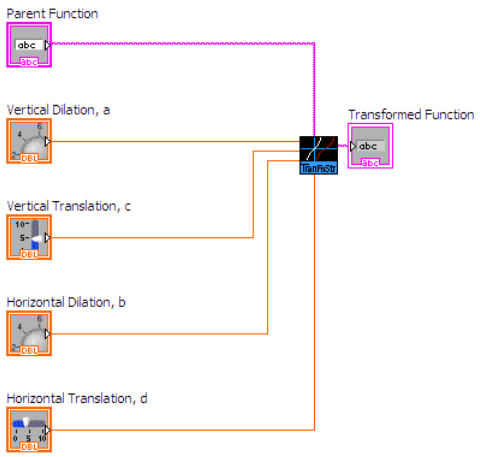

Figure 5. Controls wired to the myTransformFnString VI

Many common functions can be used for the parent function. For example, x, x^2, x^3, abs(x), sqrt(x), sin(x), cos(x), 2^x, 1/x, exp(x), ln(x) can be used. Test out your VIs ability to transform one of these parent functions.

14) Select **Window»Show Front Panel** or press **<Ctrl-E>** to view the front panel
15) Enter "x^2" in the **Parent Function** control
16) Use the other controls to set the values of the transformation parameters
17) Click the **Run** button or press **<Ctrl-R>** to run the VI
18) If necessary, resize the **Transformed Function** indicator to display the entire transformed expression

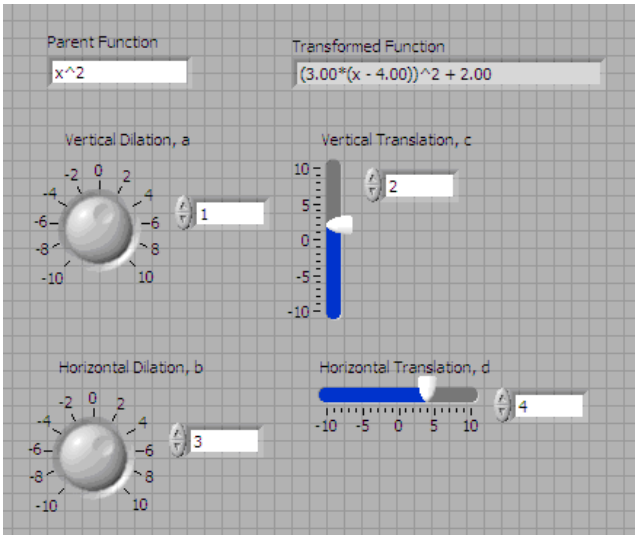

Figure 6. Transformed function displayed in front panel

Graphing the Transformed Function

The algebraic representation of the function is being successfully transformed. Next, you will add an **XY Graph** to the front panel. This indicator will be used to display the graph of the transformed function.

19) Add an **XY Graph** to the front panel
    a. Click on the double arrows at the bottom of the **Controls** palette
    b. Select **XY Graph** from **Modern»Graph**

Be sure to use the **XY Graph** from the **Modern** subpalette. The **Express XY Graph** in the **Express»Graph Indicators** palette is not the same indicator.

    c. Click on the front panel to place the graph
    d. If necessary, resize the front panel to accommodate the **XY Graph**
    e. Change the labels of the axes from “Time” and “Amplitude” to “x” and “y”
        i. Double-click on each word to select it and then type the desired text

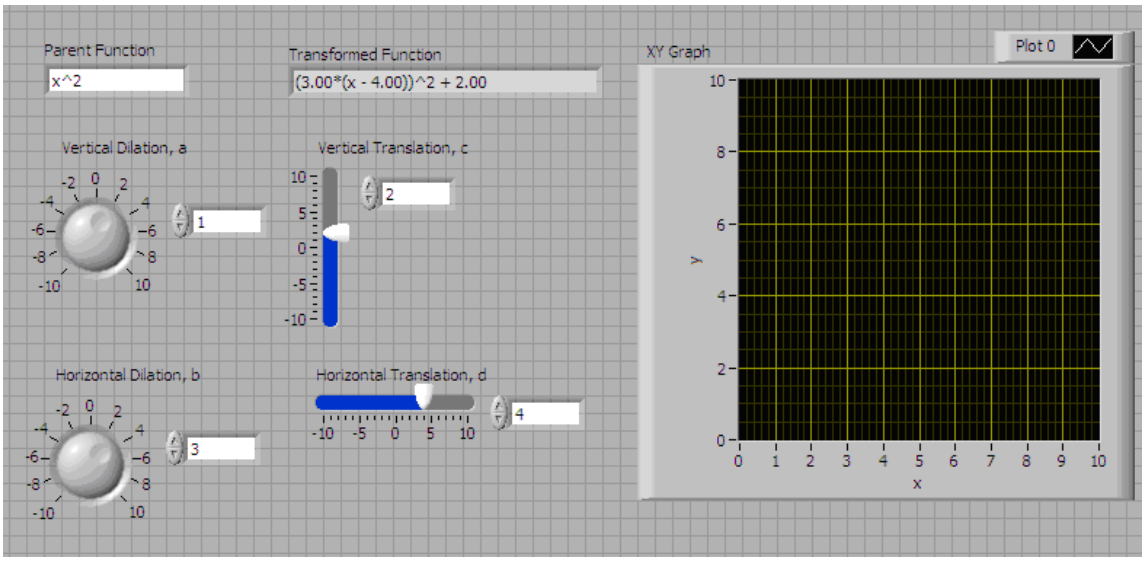

Figure 7. XY Graph added to front panel

LabVIEW has built-in VIs for evaluating a function entered as text. You can use these VIs to evaluate the transformed function for given values of *x*. To graph the transformed function, you will want a set of *xy*-pairs.

20) Select **Window»Show Block Diagram** or press **<Ctrl-E>** to view the block diagram
21) Add **Eval y=f(x).vi** to the block diagram
    a. Make room for the VI by resizing the block diagram and moving the **XY Graph** off to the right
    b. Select **Eval y=f(x).vi** from **Mathematics»Scripts & Formulas»1D & 2D Evaluation** on the **Functions** palette

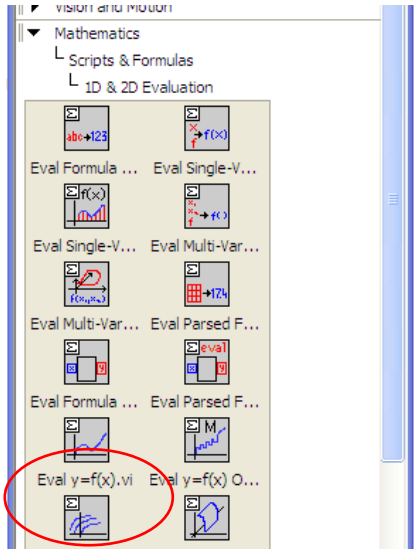

Figure 8. 1D & 2D Evaluation palette

c. Click on the block diagram to place the VI to the left of the **XY Graph**

22) Use **Context Help** to understand the **Eval y=f(x).vi**
   a. Hover the cursor over the **Eval y=f(x).vi**
   b. Press **<Ctrl-H>** to open the **Context Help**
   c. Use the **Detailed Help** link for more information about the VI
23) Use the Wiring tool to connect the **Transformed Function** output of the **myTransformFnString** VI to the **formula** input of **Eval y=f(x).vi**
24) Connect other inputs to **Eval y=f(x).vi**
   a. Right-click on the **start** terminal of the VI, select **Create»Constant**, and set the value of the constant to −100
   b. Right-click on the **end** terminal of the VI, select **Create»Constant**, and set the value of the constant to 100
   c. Right-click on the **number of points** terminal of the VI, select **Create»Constant**, and set the value of the constant to 800

The **Eval y=f(x).vi** evaluates the **formula** for a set of *x*-values between the **start** and **end**. The VI outputs both the set of *x*-values and *y*-values. These sets are limited by the use of −100 and 100 for the **start** and **end** values. (It is possible to capture the X Scale minimum and maximum and use those for **start** and **end**. To do so, requires the use of **Property Nodes**, an advanced topic left for another time.)

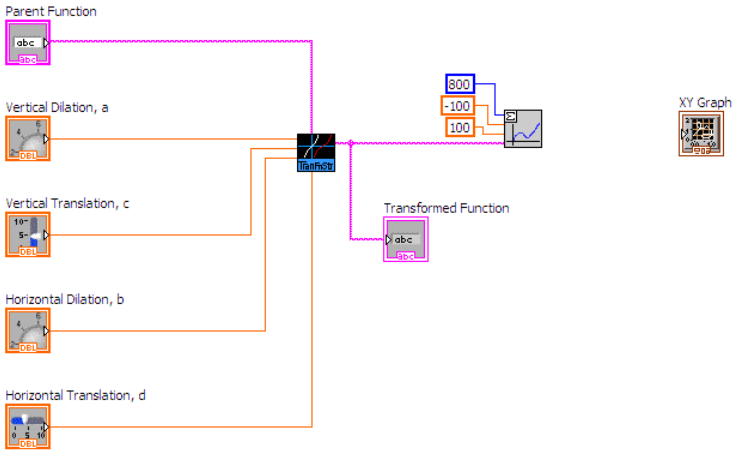

Figure 9. Evaluation VI added to block diagram

The **Eval y=f(x).vi** creates an array of *y*-values corresponding to an array of *x*-values. The **XY Graph** uses both of these arrays of values to graph the function. To connect the two arrays to the **XY Graph**, you will bundle the arrays together as a cluster.

A cluster is like an array in that it can hold multiple items of value. The two types differ in that a cluster can hold multiple data types, while an array holds multiple elements of a single data type.

25) Bundle the *x* and *y* arrays into a cluster
   a. Select **Programming»Cluster, Class, & Variant»Bundle** and click on the block diagram to place the VI

b. Connect the **X values** output of the **Eval y=f(x).vi** to the first element of the **Bundle** VI
c. Connect the **Y values** output of the **Eval y=f(x).vi** to the second element of the **Bundle** VI

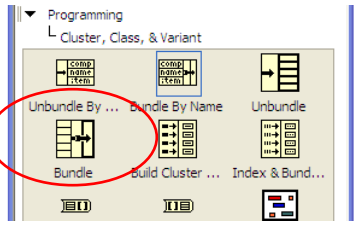

Figure 10. Bundle VI on the Cluster, Class, & Variant palette

26) Wire the output of the **Bundle** VI to the **XY Graph**
    a. The XY Graph will change from brown to pink to denote its data type as a cluster of numeric arrays
27) Save your VI, **<Ctrl-S>**

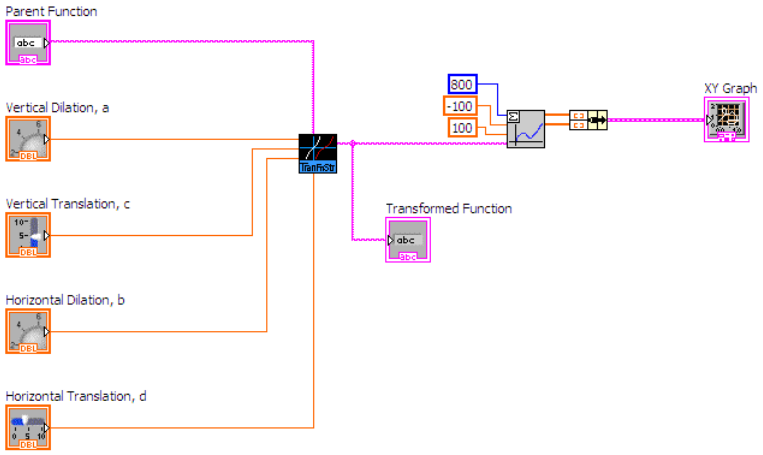

Figure 11. Data wired to XY Graph

Your VI is now ready to display the graph of the transformed function.

28) Return to the front panel by pressing **<Ctrl-E>**
29) Enter a desired **Parent Function** and set the transformation parameters, $a - d$, to non-zero values
30) Click the **Run** button or press **<Ctrl-R>** to run the VI

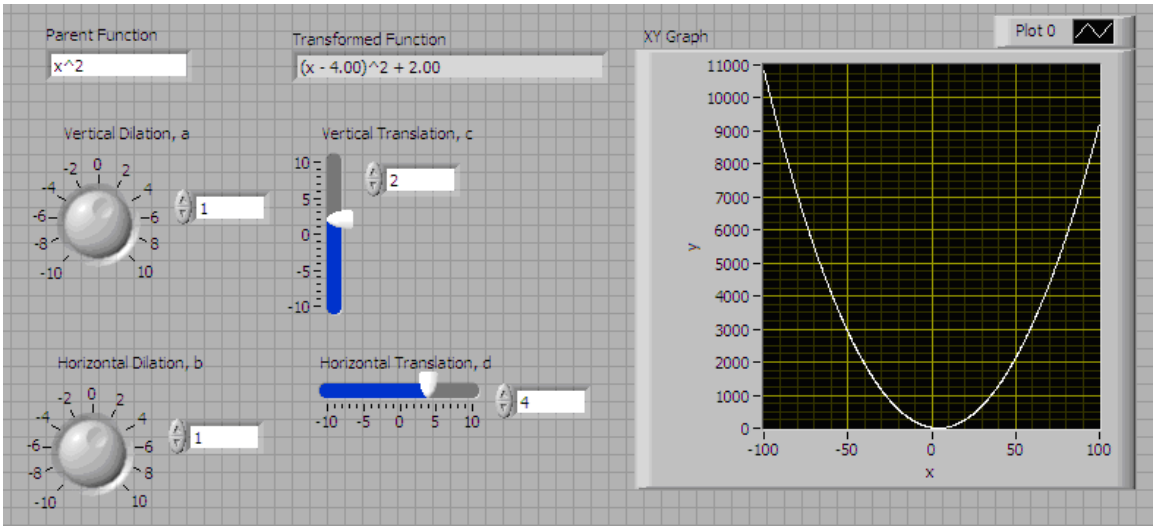

Figure 12. Graph of transformed function

Notice the x and *y*-values in the X and Y Scales are automatically adjusting. This is called autoscaling. Autoscaling can be turned off and the graph's window can be controlled using the values displayed on the X and Y Scales.

31) Configure the **XY Graph**
   a. Turn off autoscaling
      i. Right-click on the graph and select **X Scale»AutoScale X** to deselect autoscaling
      ii. Right-click on the graph and select **Y Scale»AutoScale Y** to deselect autoscaling
   b. Adjust the limits on the X and Y axes by clicking on the upper and lower limits on the axes and setting them to −10 and 10 respectively

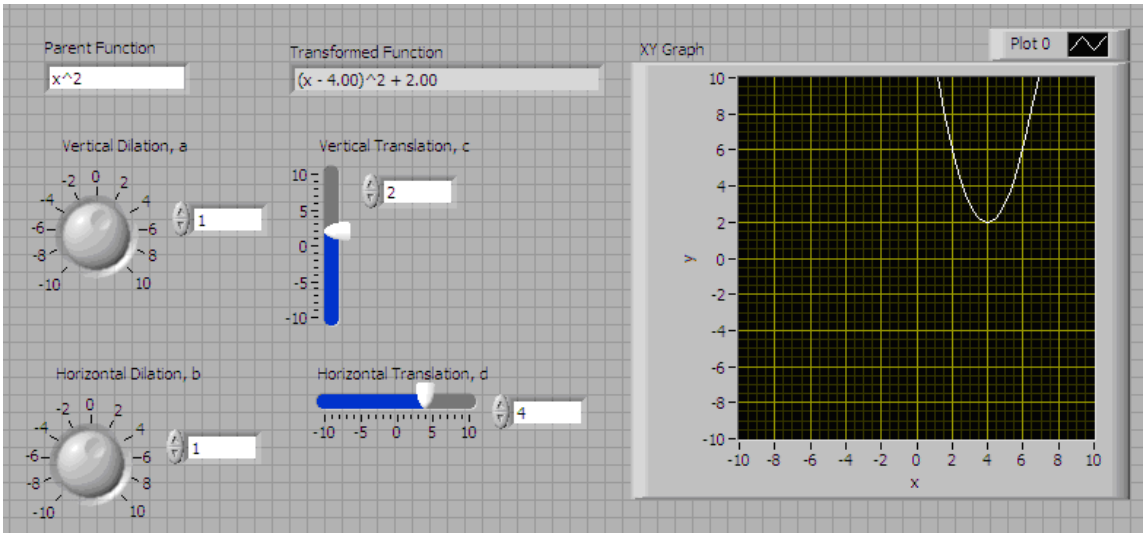

Figure 13. XY Graph with autoscaling turned off

Now, the window shown in the graph can be controlled by editing the values shown in the scales. This window has the transformation be more evident. You can change the window anytime, even while the VI is running.

You can run the VI using the **Run** button or the **Run Continuously** button. Running continuously will have the code in the block diagram executed over and over again. This has the VI be interactive while it is running. To stop the execution, use the **Abort Execution** button.

32) Run the VI continuously, play with the transformation parameters, and notice the changes in the graph and the transformed function.

Graphing the Parent Function

In addition to displaying the graph of the transformed function, it would be nice to display the graph of the parent function also. You simply need to evaluate the parent function over a set of *x*-values and pass the *xy*-pairs to the **XY Graph**.

33) Add a second **Eval y=f(x).vi** to the block diagram
    a. Click on the **Eval y=f(x).vi** icon and select **Edit»Copy**, or press **<Ctrl-C>**
    b. Click on the block diagram above the copied one and select **Edit»Paste**, or press **<Ctrl-V>**, to place the VI
34) Use the Wiring tool to connect inputs to the new **Eval y=f(x).vi**
    a. Connect the **Parent Function** control to the **formula** input
    b. Connect the three constants used before, **number of points**, **start**, and **end** to the corresponding inputs

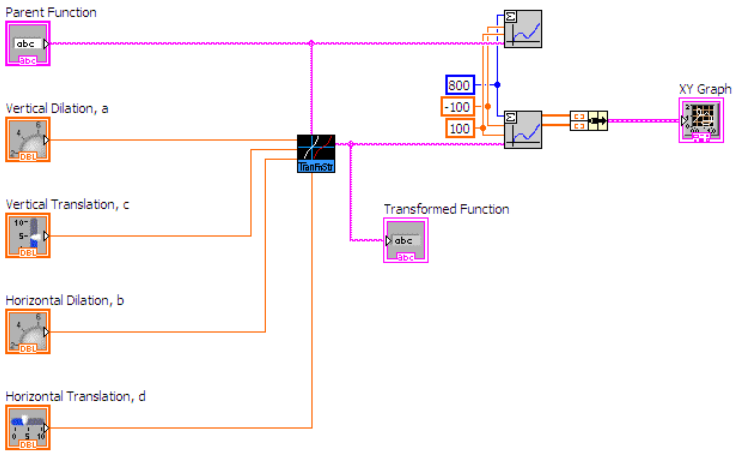

Figure 14. Setting up for graphing parent function

You will follow some of the same steps as before to connect the new set of *xy*-pairs to the XY Graph.

35) Bundle the *x* and *y* arrays into a cluster
    a. Copy and paste the **Bundle** VI above the previous **Bundle**
    b. Connect the **X values** output of the **Eval y=f(x).vi** to the first element of the new **Bundle** VI
    c. Connect the **Y values** output of the **Eval y=f(x).vi** to the second element of the new **Bundle** VI
36) Delete the wire connecting the first **Bundle** VI to the **XY Graph**
    a. Select the wire
    b. Press **Delete**
37) If necessary, select and move the **XY Graph** to the right

The two clusters of *x* and *y* values have to be put together so that there is a single wire carrying all of the data to the **XY Graph**.

38) Combine the two clusters of *xy*-pairs in an array
    a. Select **Build Array** from **Programming»Array** in the **Controls** palette
    b. Click on the block diagram to place **Build Array** to the right of the **Bundle** VIs
    c. Resize **Build Array** to allow for two inputs
    d. Connect the output of each **Bundle** VI to an input of **Build Array**
39) Connect the output of **Build Array** to the **XY Graph**
40) Save your VI, **<Ctrl-S>**

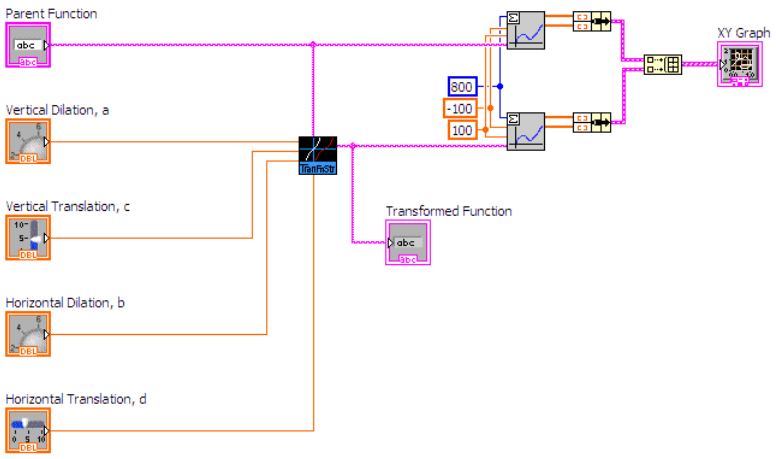

Figure 15. Parent function graph set to display

Now, when you run your VI, both the parent and transformed function will be graphed. You can customize the **Plot Legend** to label each graph. As it is, the **Plot Legend** shows the white graph as "Plot 0".

41) Run the VI
42) Select **Window»Show Front Panel** or press **<Ctrl-E>** to view the front panel
43) Resize the **Plot Legend** so that "Plot 0" (white) and "Plot 1" (red) are both visible
44) Triple-click each label to change them to "Parent" (white) and "Transformed" (red)

45) Save your VI, **<Ctrl-S>**

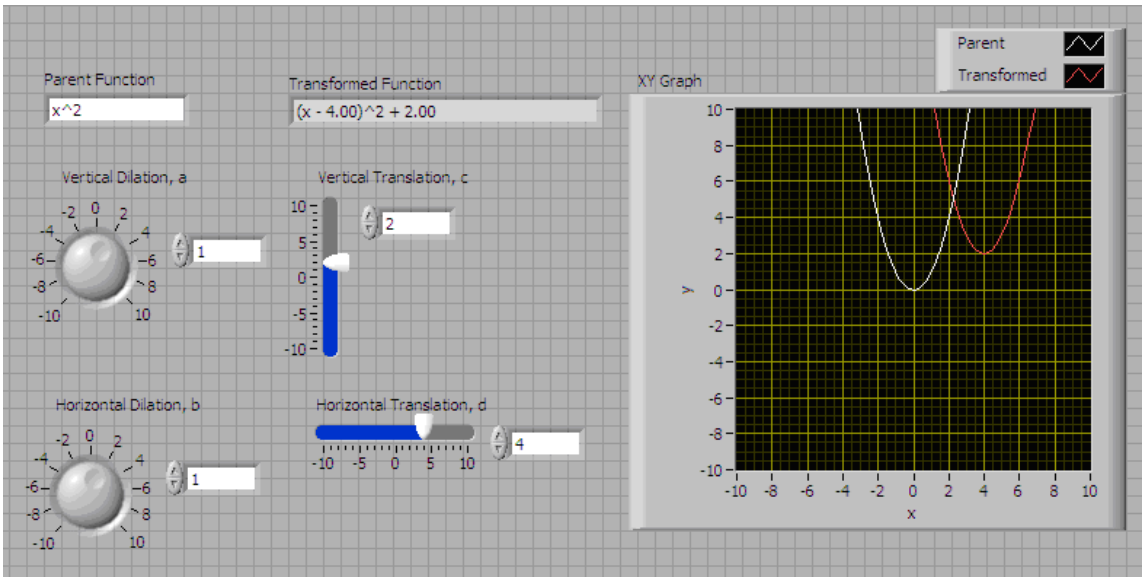

Figure 16. Parent and transformed functions graphed

The Transformations VI is complete. Of course, more customization is possible. Right-click on any of the controls or indicators to see what else could be made visible or what properties could be changed. Many display options are configurable.