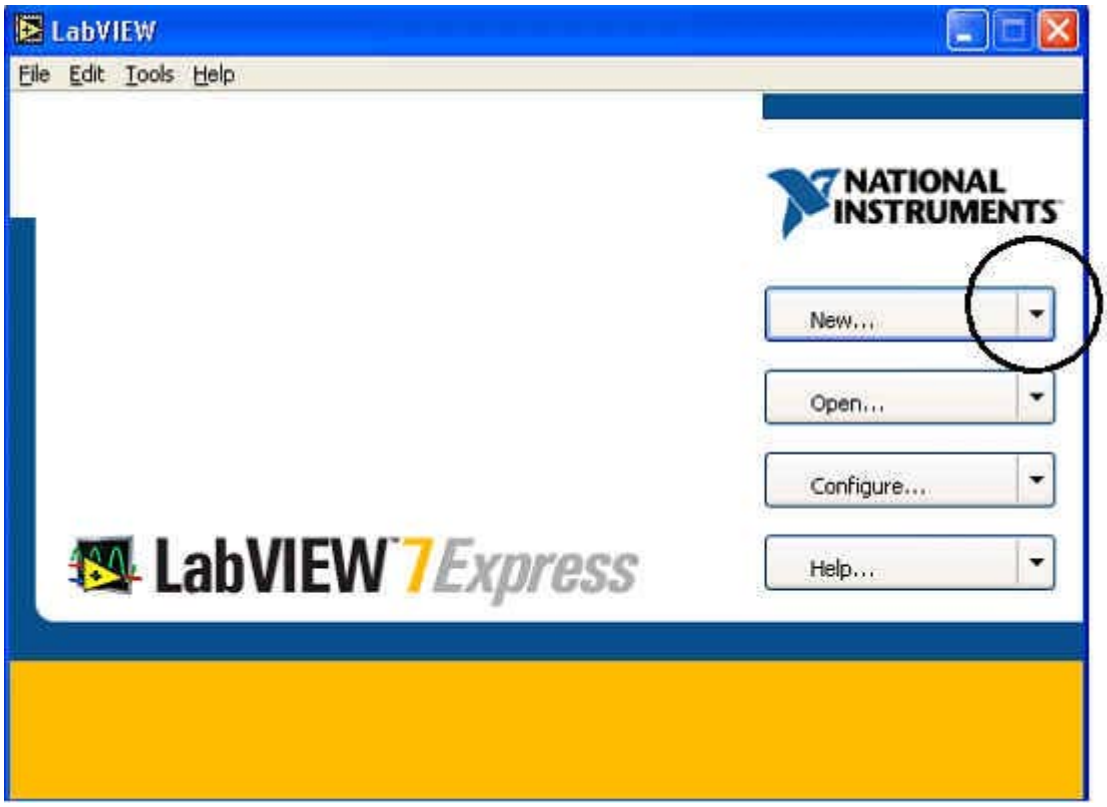
First Bytes Programming Lab 2

This lab is available online at www.cs.utexas.edu/users/scottm/firstbytes.

Introduction: In this lab you will investigate the properties of colors and how they are displayed on computers. In the realm of light, colors are defined by the intensity of red, green, and blue light. The combination of these three colors creates all other colors that are seen on computer screens.

1. Start LabVIEW. Select **Start >> Programming >> National Instruments** (the top choice on the programs menu) >> **LabVIEW.** It will take a few moments for the program to start. An introductory window will open. Click on the **Continue** button when this pops up to go on to the main LabVIEW start menu.

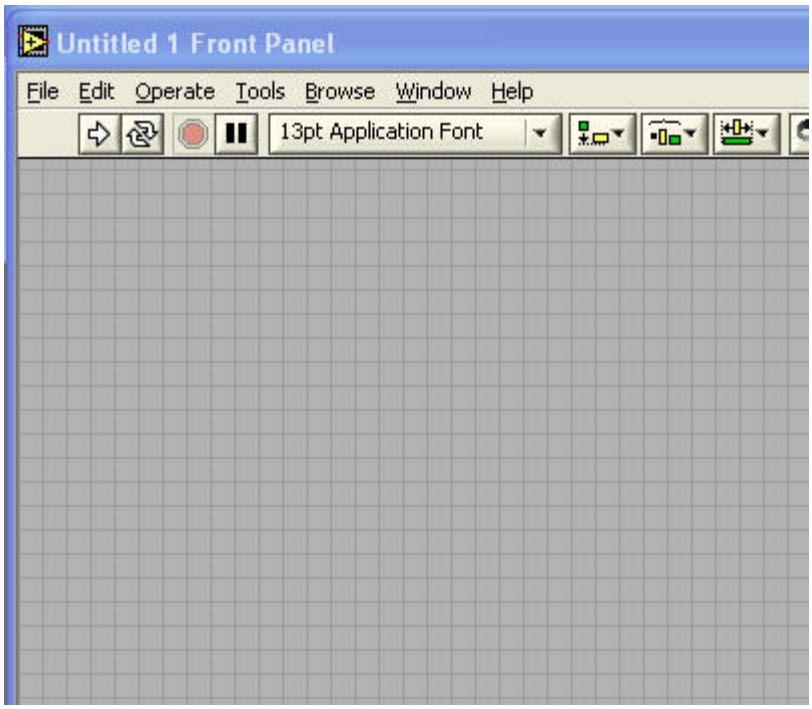
2. When the LabVIEW start window is open select the arrow on **New** and select **Blank VI**.



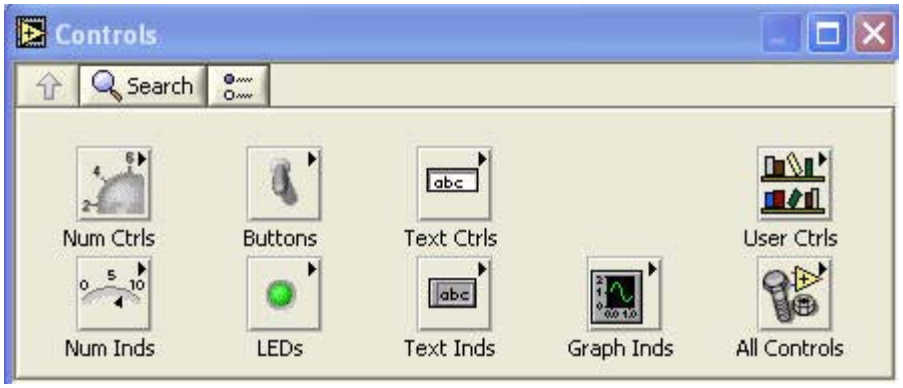
LabVIEW programs are known as VIs, which stands for virtual instrument. The VI consist of two parts, the *Front Panel* which is the part someone using the program interacts with and the *Block Diagram* which is the part the programmer works with to create the inner workings of the program. When creating a VI you, the programmer, can add parts to either the block diagram or the front panel.

3. Arrange the block diagram and front panel so they are easier to work with. At the top of either window select **Window** and then **Tile Left and Righ**t or **Tile Up and Down** depending on you preference. Or your preference may be to only display one window at a time and minimize the other. **Note, if you close the front panel display the program will be closed.**

4. Now we are ready to add some components to our program. Select the front panel window that will have a gray background with a grid on it.

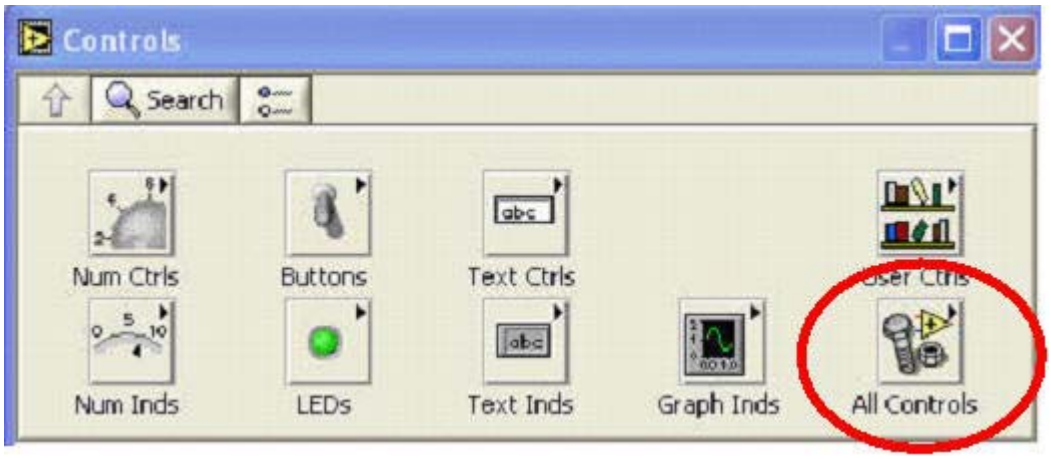


5. When you select the Front Panel window you may see a window entitled **Controls** or **Numeric Controls** as shown below.

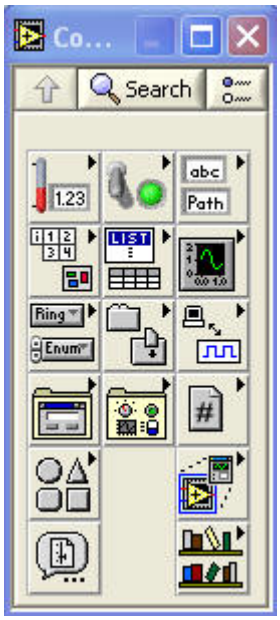


If you do not see one of the windows right click in the front panel display and the window will appear. Click the push pin to keep the controls window open.

Change the view of the Controls menu. To do this click on the **All Controls** button which is circled below.



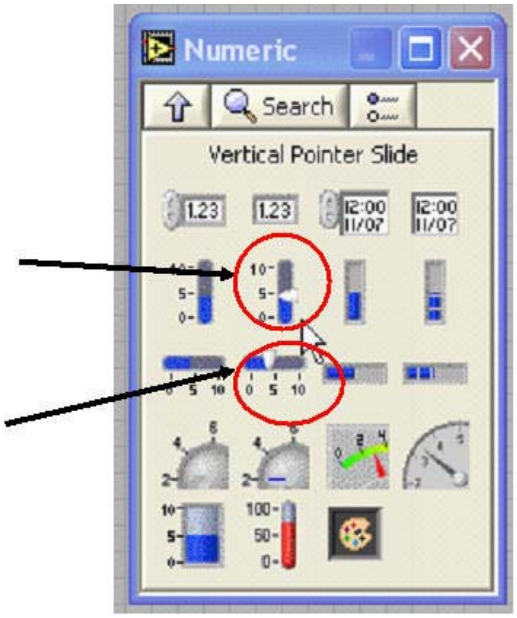
After clicking the button the **Controls** menu should now look like this:



6. Once the Controls Window is open and in the advanced view press the **Num Ctrls** button to open the Numeric Controls button.

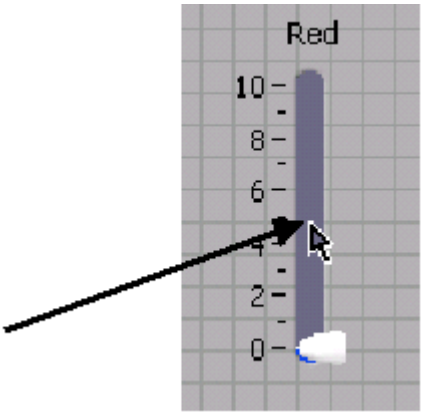
On the front panel add three pointer slide controls from the numeric controls menu to the front panel itself. These can be horizontal or vertical sliders.

(on Front Panel)

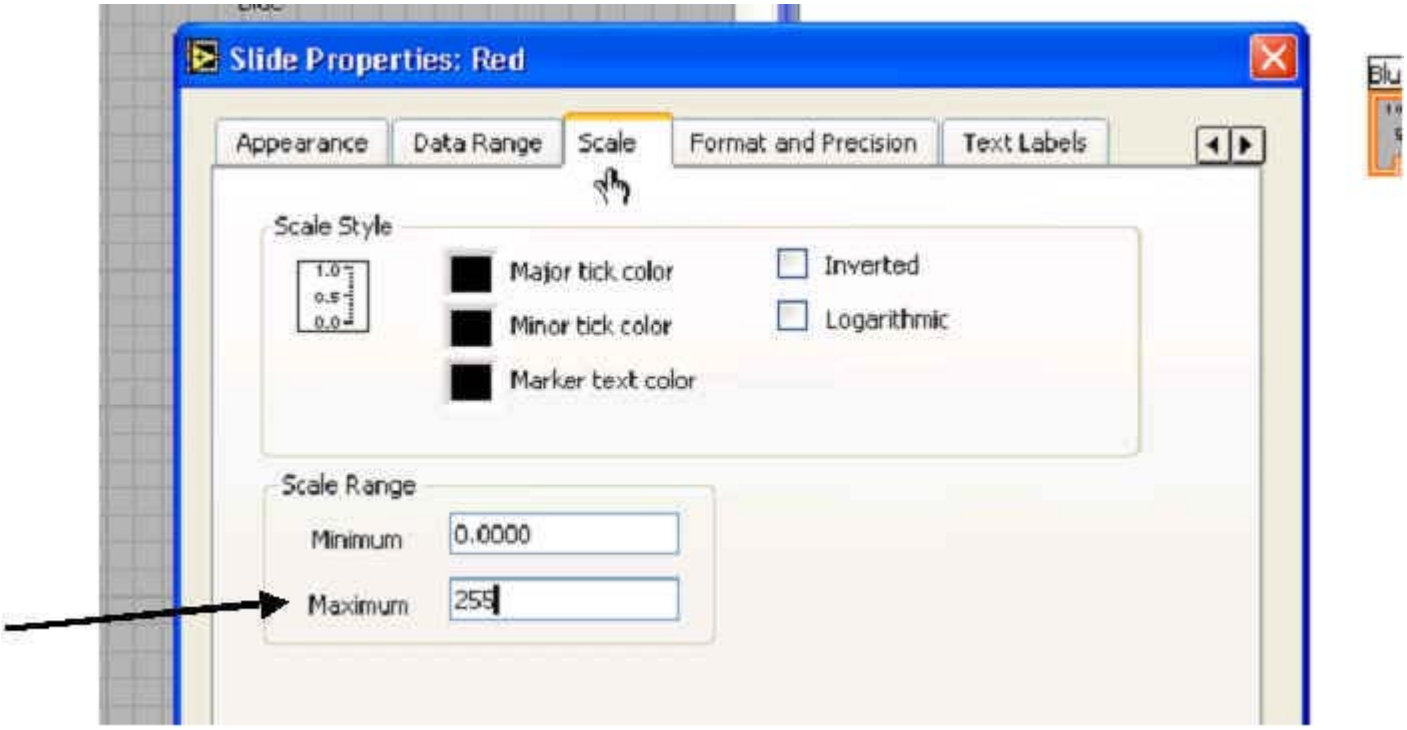


Alter the name of the slides so that one is labeled **Red**, one **Green**, and one **Blue**.

7. Change the range of the input for each slider. There are many ways to indicate the intensity of red, green, and blue in a given color. One of the most popular, termed true color uses a scale from 0 to 255 for the intensity of each color. Right click on the fill portion of a slider. From the menu that appears select **Data Range**

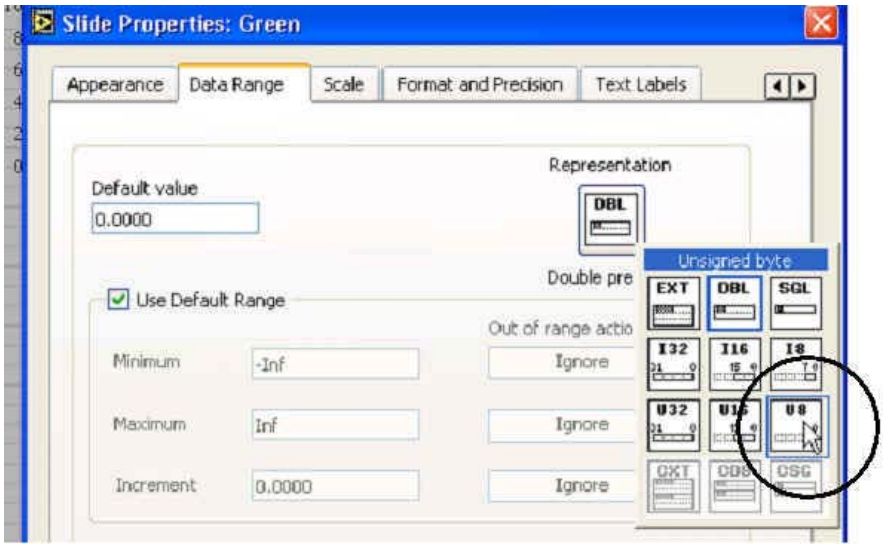


A window entitled **Slide Properties** appears. First select the **Scale** tab. Alter the scale of the maximum of this slide to 255.



Now select the tab entitled **Data Range**. Change the representation of the input from **Double Precision** to **Unsigned byte**. Representation refers to how the input from the slider is treated. Computers store all information as 1s and 0s, but those 1s and 0s can be interpreted many, many ways. Treating the input as double precision means using 62 1s and 0s to represent the number. (a single 1 or 0 is known as a bit, 8 bits are a byte. A double precision number uses 8 bytes or 62 bits to represent the number.) In this application we only need to represent the integers from 0 to 255. To do this requires 8 bits or a single byte. The term unsigned refers to the fact that this representation does

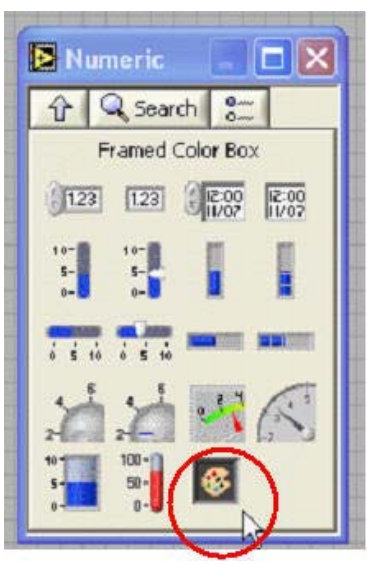
not allow negative numbers. This makes sense for choosing the intensity of a color, because it is not possible to have a negative intensity of red, blue, or green.



8. Alter the range and representation for the Green and Blue sliders. If you want to make the display more intuitive you may want to change the fill color for the Red and Green sliders to the appropriate color. This can be done by selecting the **Appearance** tab on the Slide Properties window and left clicking in the fill color box. This allows you to select whatever color you want for the fill color of the slider.

9. Add a box to display the resulting color. On the front panel drag a **Color Box** icon from the Numeric Control window to the front panel itself, below your three sliders.

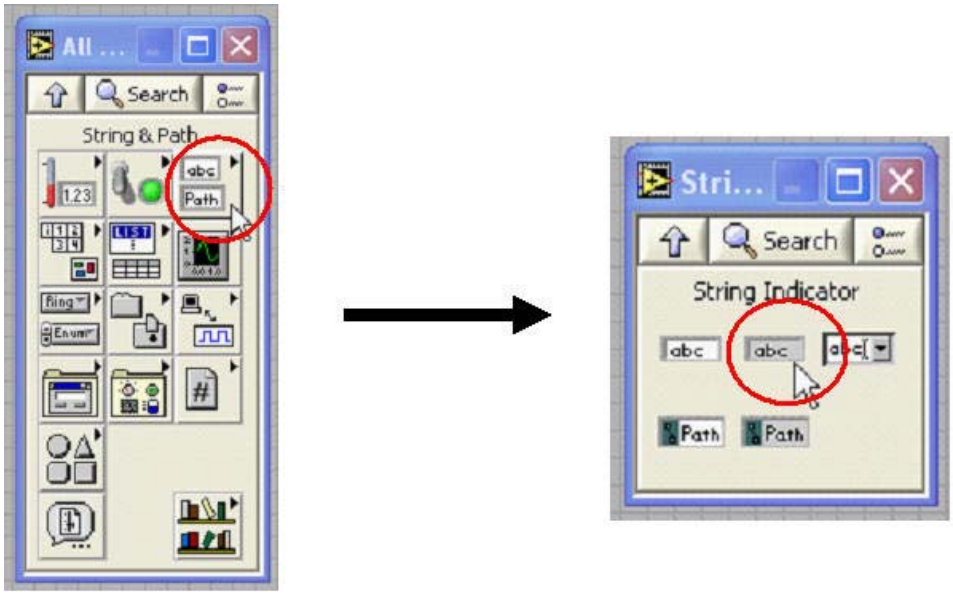
   (on Front Panel)



Click on the black color box and make the box larger by expanding it. Change the name of the color box to **Actual Color**.

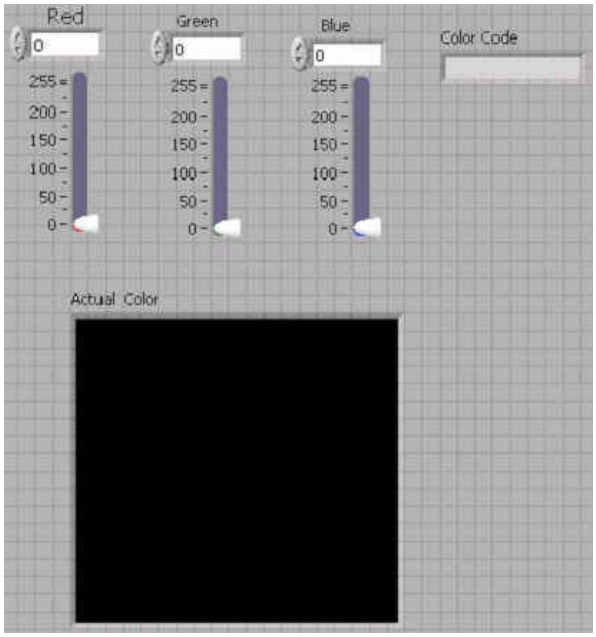
10. Finally add an indicator for the numeric description of the color selected with the sliders. Back up to the main Controls window. Select the **Text Inds** icon.

    (on Front Panel)

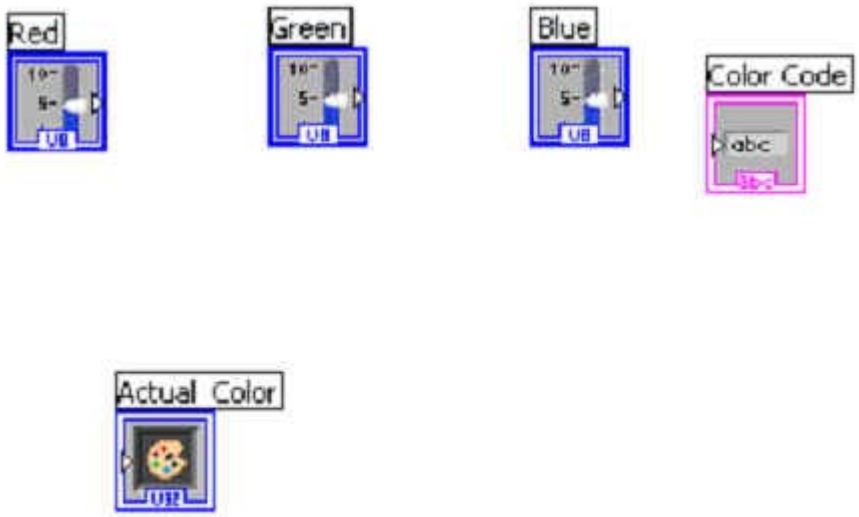


Click and drag a **String Ind** control from the window to the front panel. A string is a collection of 0 or more characters. In order to put the numbers representing the amount of red, green, and blue together we will convert them to strings, piece them together, and then convert the final result back to a number.

Change the name of the String Indicator to **Color Code**. Your front panel should now look something like this:

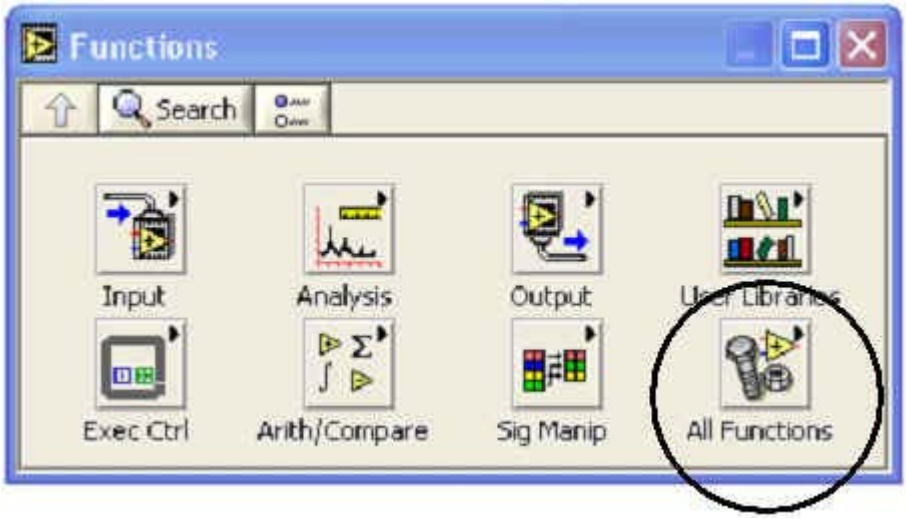


11. Right now our program doesn't do anything. We must make the connections inside the block diagram to take the inputs, convert them to strings, connect them together, convert the result back to a number and feed that number to the color box. Your block diagram should look something like this:



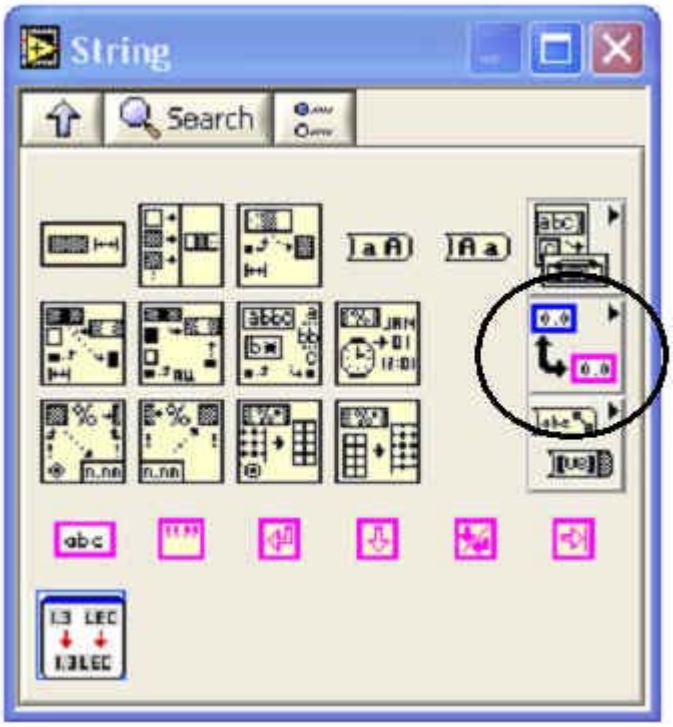
12. We need a component to convert a number to a string. On the main Functions window select the **All Functions** icon.

(on Block Diagram)



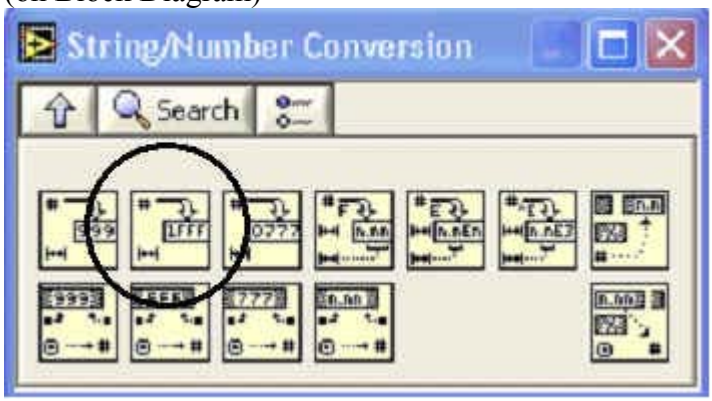
On the All Functions window select the String icon. Then on the **String** window select the S**tring Number Conversion Icon**.

(on Block Diagram)



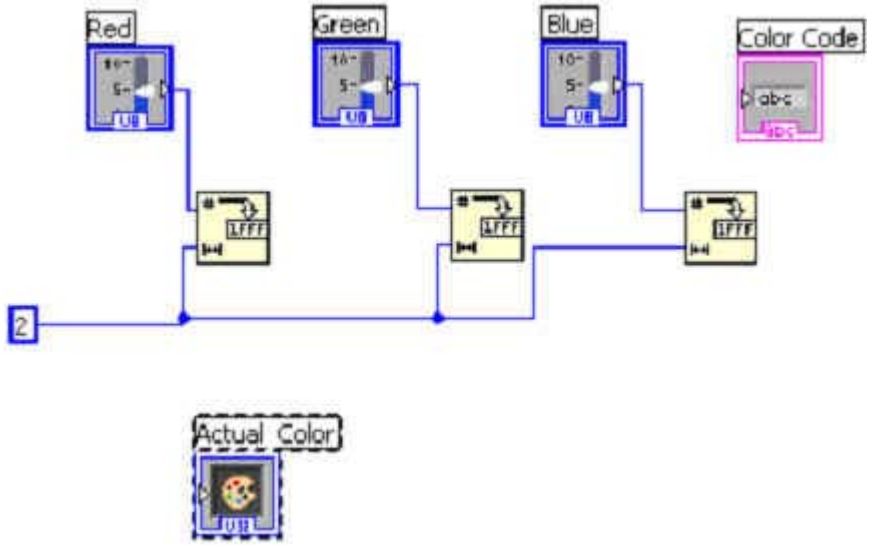
On the **String/Number Conversion** window select and drag 3 **Number to Hexadecimal String** icons and place them on the block diagram. A Hexadecimal number is a number that is represented in a base 16 system. Most people are familiar with the base 10 numbering system also known as the decimal system.

(on Block Diagram)



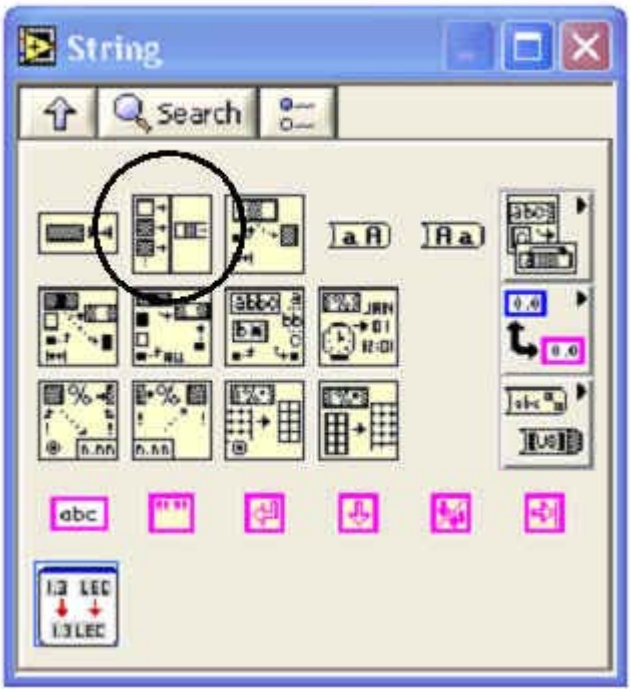
13. Connect each of the outputs of the red, blue, and green controls to a separate conversion icon. The proper spot to connect to is the upper left terminal on the conversion icon. It is also necessary to indicate the proper width of the input. Navigate on the controls to insert a numeric constant. Drag this numeric constant to the block diagram and set its value to 2. Wire the output of this constant to all three of the width inputs on the conversion icons. The width terminal is the lower left terminal on the

conversion icon. Your block diagram should look something like this:

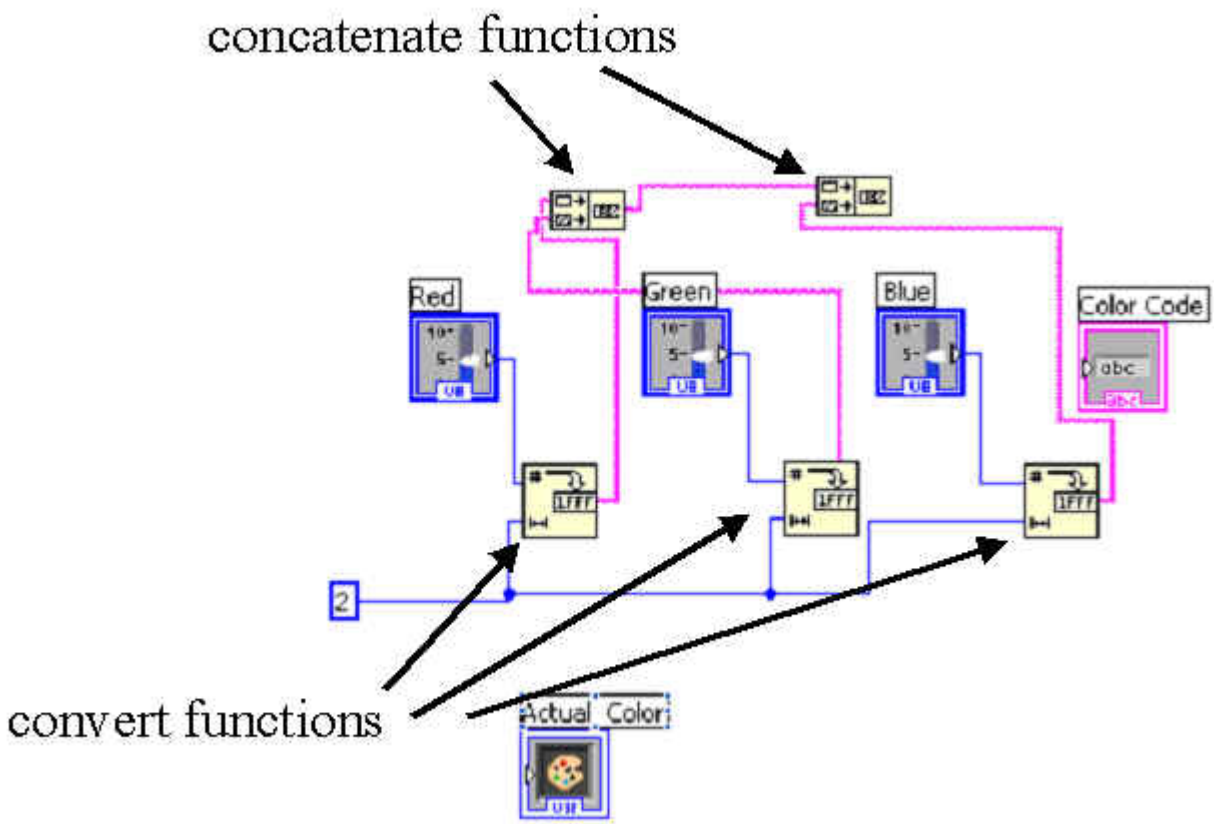


14. Now the output of the icons that convert the input numbers to string must be put together. *Concatenation* is the term usually used for connecting strings together. Navigate back to the String menu and add 2 concatenate string icons.

    (on Block Diagram)



15. Connect the converted Red value to the top terminal of one of the concatenate icons and the converted Green value to the bottom of the icon. Connect the output of the 1st concatenate icon to the top terminal of the 2nd concatenate icon and finally connect the converted Blue value to the bottom terminal of the 2nd concatenate icon. You block diagram should now look something like this:

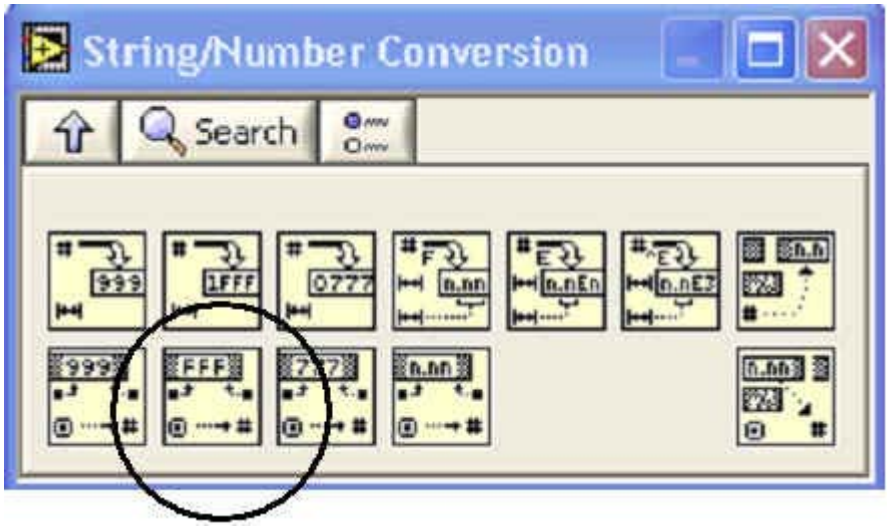


Note, the order of the operands for the concatenate function matters, just like the order of operands in subtraction or division. In division 20 / 5 = 4, but 5 / 20 = 0.25. The order matters. This is true of concatenation as well. "Hi" concat " There" = "HiThere", but " There" concat "Hi" = " ThereHi". The String we are creating for the color code expects the intensity of Red first, then Green, then Blue. RRGGBB. The left hand operand to the concatenate function is the top input. The right hand side operand is the bottom input.

16. Wire the result of the 2nd concatenate icon to the result indicator.

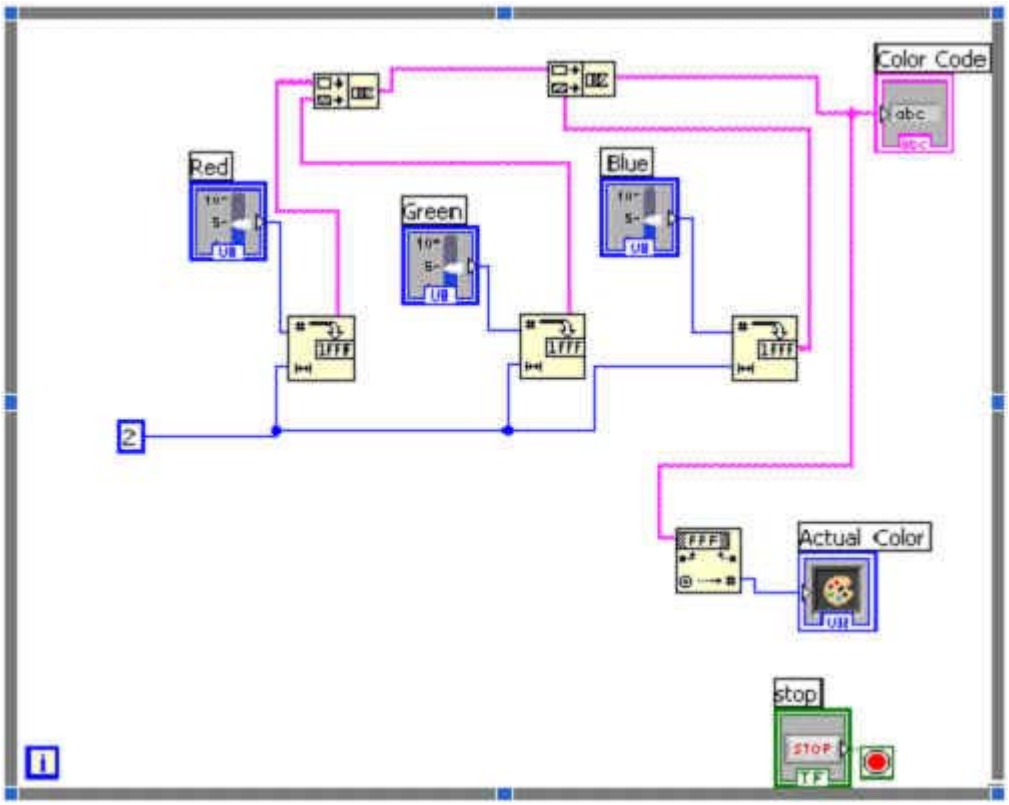
17. Now that the String has been created that represents a color it must be converted back to a number so that it can be fed to the result color box. Navigate back to the **String/Number Conversion** Window and drag a **Hexadecimal String to Number** icon to the block diagram.

    (on Block Diagram)



Wire the output of the 2nd concatenate icon to the top left terminal of the **Hexadecimal String to Number Converter**.

18. Right click on the color box to change it to an indicator. Wire the output of the converter (terminal on the bottom right side) to the input of the color box.
19. Finally add a **while loop** control around the entire block diagram. Your final program should look something like this:



20. Run your program. Adjust the sliders and notice the output color. How does one make "Pink", "Purple", "Orange", and "Yellow"? This format for colors is used in html, the description language for web pages. Go to a web page and open up the html source for the page. This is usually done by selecting **View** and then **Source** or **Page Source** under the **View** menu. Search for the word "color"

in the html code. You can search rapidly using the **find** option. Press the control key and the f key at the same time to open **find**. Type in color and search for that word. You will often see a code for the color. Such as #662626 or #B28B12 or #F1F1FD. (Those are just a few of the colors on Yahoo!) Enter the color code you find on a web page into your color picker and see if you can spot the color on the web page.

21. What next?
    A. Add the ability for a user to enter a color code directly instead of using the sliders
    B. add loops to create a display of many shades of colors. Don't try and display them all, but try every 10th or 20th color code.