

# System Level Laboratory

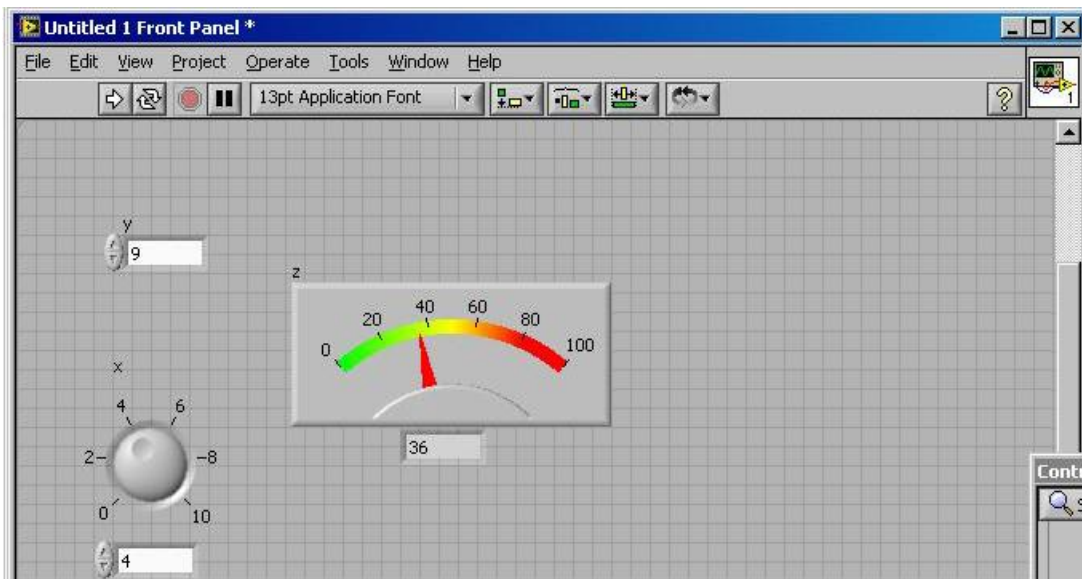
## Introduction of LabVIEW and Virtual Instrument (VI) Construction

### Exercise 1: VI Construction

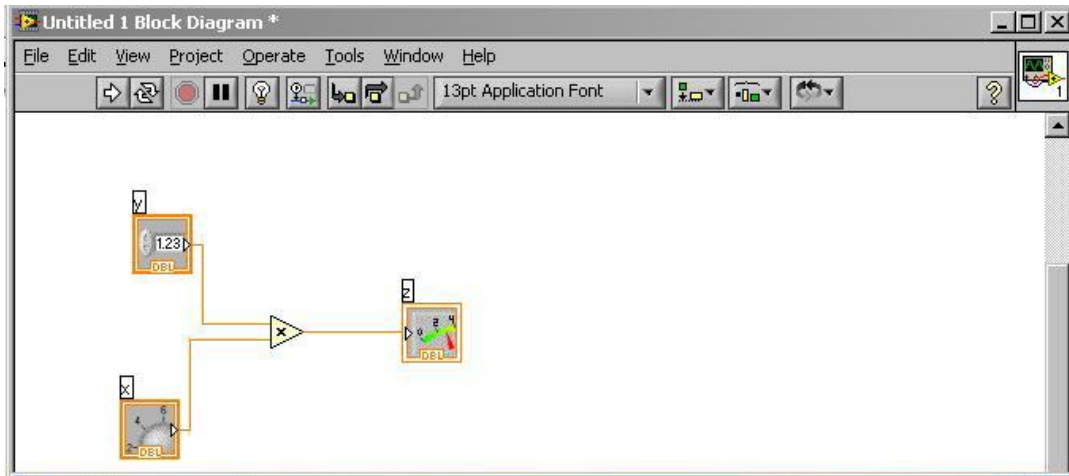
A VI is constructed by building both the front panel and associated block diagram. As an illustration let us develop a VI that will perform the following functions:

- 1). Use **Knob** in **Numeric Controls** on the front panel to input a value,  $x$ , varies from 0 ~ 10
- 2). Use **Numeric Control** in **Numeric Controls** to input another value,  $y$ .
- 3). Compute the product of  $x$  and  $y$  and display the result on a **Meter** on the front panel.

Design a front panel like the one shown below:



The Block Diagram window should look like the one shown below :



Make the front panel active and change the cursor to the operation symbol (the hand) to see the output shown in Z

## Exercise 2: Develop VI using formula node

Last time we "solved" a simple equation by using an input along with associated mathematical operations and then an output. Now we will see how this can be done using a special block into which we put formulas. This can be used to compute multiple outputs and involve complicated functions such as exponentials and trig functions.

Suppose we have an input, say  $x$ , and we want to determine two outputs given by the equations,

$$y = 2x^2 + 4\sin(x) + 5 \quad \text{and} \quad z = 3y + 2xe^{-0.2x}$$

It is easy to solve these two equations using the Formula Node:

**Step 1.** Start a new VI

**Step 2.** On the Front Panel put in an input labeled  $x$  and two outputs (use **Numeric Indicator**) labeled  $y$  and  $z$ .

**Step 3.** On the Block diagram:

- a. On the Functions Tool Box click on **Structures** → **Formula Node** and put it into the window



- b.** Inside the block, type the first equation above. In writing the equation, use “\*” for multiply, “pow(x, 2)” for “ $x^2$ ”, “exp(x)” for “ $e^x$ ”. Use semicolon at end of each equations.
- c.** Point to the left edge of the box of Formula Node, right click and select “add input”. Inside the popped up box, label it as “x”.
- d.** Point to the right edge of Formula Node, right click and select “add output”. In the popped up box, label it “y”.
- e.** Repeat above steps to insert another Formula Node to calculate z.
- f.** "wire" the user screen input block to the input of the Formula Node x, and the two outputs to indicators.

**Step 4.** Test the formula node by enter numbers for x, run the simulation, and see the y and z change accordingly.

### Exercise 3: Programming Structures: FOR and WHILE loop

#### 1. Use of the FOR loop

The For loop is also found under structures. Everything inside the For block will be executed for the specified number of times.

**Step 1.** Start a new VI (or just erase everything from the previous one)

**Step 2.** On the Front Panel put an **Numeric Indicator** as output **BUT** change it from **DBL** to integer readout **I32**. This is done by right clicking on the output icon in the Block Diagram and selecting **Representation** and then **I32**.

**Step 3.** On the Block Diagram screen:

- a. Go to **Structures** → **FOR** and drag this into the window
- b. Drag the icon for the output inside the block
- c. Drag a **Numeric Constant** (from **Mathematics** → **Numeric**) into the window but outside of the FOR blocks and inserts ‘50’ (this will be the number of times we will execute the FOR)
- d. Add a time delay so the FOR does not execute too fast to see. Go to the **Timing** of the functions block and drag the symbol for a **Metronome** names “Wait until Next mms Multiple” into the FOR block.

- e. Drag a **Numeric Constant** into the FOR and insert a 500, wire this to the metronome.
- f. Wire the FOR loop increment bloc (i) to the output.

Now as the loop executes, you will see the incrementing count, delayed by the count to the increment in milliseconds.

## 2. Use of the While Loop

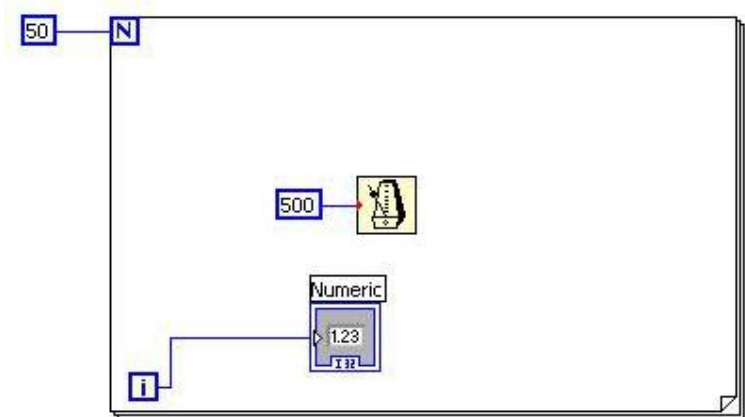
This is similar to the FOR but everything in the block executes until a BOOLEAN condition becomes false. You can use a simple switch to control the loop in this example

You find the While loop under structures. You can see the “i” which represents the number of times the loop has been executed and the little half circle which represents the BOOLEAN condition. This must be true for the loop to execute.

**Step 1.** Start a new VI.

**Step 2.** On the Front Panel put the following items:

- a. a knob from **Numeric** to control delay, a **Push Button** from **Boolean** to start and stop the loop, and a **Numeric Indicator** for the output.



**Step 3.** On the Block Diagram screen:

- a. Go to **Structures** → **While** and drag this into the window.
- b. Put everything inside.
- c. Add a time delay. Drag the **Metronome** into the While block.
- d. Wire the **Knob** to the **Metronome**.
- e. Wire the While loop increment bloc (i) to the output.

Now as the loop executes, you will see the incrementing count, the knob can control the speed of the counter to the increment in milliseconds.

## Exercise 4: Build a VI that will compute and display on a chart the function:

$$y(t) = 5(3.14*t-1) \text{ for } t \text{ varying from } 0 \text{ to } 30 \text{ seconds.}$$

You can use either FOR or WHILE loop. Show your work to TA.

### Exercise 5: Build a VI that will calculate following equation:

$$\text{Result} = 0.5 * X + \sin (Y) + Z^2$$

Consider X, Y and Z as input value, display the Result by Numeric Indicator. Use the VI to build a new VI, and show it to TA.

### Exercise 6: Displaying Scaled Random Numbers within defined limit on a Chart

**Step1:** Write a Lab VIEW virtual instrument (VI) to read voltages and display them.

**Hint:** Build a VI that works like a graphing digital multimeter. Your multimeter must be able to set the range of values for the readings (resolution) and display the reading on a digital readout and a chart. You may need a While loop, a Random Number Generator from “Numeric” and Waveform Chart, and a metronome (set the delay factor to 100).

Run the VI several times. Does the VI run? How do you know? How could you determine the number of times the while loop executes each time you run the program? Hint: What does the other small square with an "i" do in the bottom left corner of the While Loop? Answer these questions on your lab report.

**Step 2:** Add parameters to your program to provide a scaled random number between user defined values of X and Y.

Implement the following equation using Multiply, Subtract, and Add function:

$$\text{Output} = \text{Random} \times (\text{Upper} - \text{Lower}) + \text{Lower}$$



Run the VI with five different sets of values of Upper Limit and Lower Limit. Record a few of the values you get after pressing the stop button each time. Did you get any values outside the limits? Write the numbers and your answer on your data sheet.

## II. Post lab Exercise and Question (for individual student):

### 1. Create a Calculator VI

Using the “Case Structure” that write a VI that adds, subtracts, multiplies, divides, and averages two input numbers and displays the results on the front panel. Save your VI as Calculator.vi. Turn in your snap shot of front panel when running the simulation and block diagram.

## 2. Create a VI to:

Read in a .wav file, display the waveform, and smooth it using averaging of five adjacent values. GUI is similar to this:

