

# LabVIEW Lesson 6 – File I/O and Plotting

## Lesson Overview

- Why use file I/O?
  - How to incorporate file I/O's into a VI.
  - What file operations are available?
  - What types of plots are available and when to use them.
  - What plotting options are available/How to edit plots?
- 

### I. Why use file I/O?

- a. File I/O is used to pass data between your VI and certain data files.
- b. The file I/O operations allow the user to perform any of the following:
  - i. Opening and closing data files
  - ii. Reading from or writing to files including spreadsheets
  - iii. Managing files (moving, renaming, or changing file characteristics)
  - iv. Managing configuration files (creating, modifying, and reading)
- c. Two types of file I/O VIs exist:
  - i. **High-Level** – performs common I/O operations all in one VI (i.e. opens, reads, and closes file).
    1. These VIs can be accessed in the Functions Palette:
      - **Functions Palette** → **File I/O**
    2. These VIs can perform operations from the following data types:
      - a. Characters to and from text files
      - b. Lines from text files
      - c. Arrays of single-precision numerics to and from spreadsheet text files
      - d. Arrays of single-precision numerics or signed 16-bit integers to and from binary files
    3. Note: Avoid inserting high-level VIs in loops as the designated operation will execute during each iteration.
    4. High-level VIs always need a file path input and if one is not assigned, an error message will appear prompting for the file path.
  - ii. **Low-Level** – performs only one I/O operation (i.e. reads file)

- d. File I/O formats depend on the type of files being used. 3 types of formats exist: text, binary, and datalog. Some basic guidelines for choosing a data type are:
- i. Text files are the most common and should be used if the data will be used in programs such as Word and Excel.
  - ii. Binary files are more efficient than text files with respect to disk space and speed and should be used if memory and speed is crucial.
  - iii. Datalog files are the best way to store data if the data will only be accessed in LabVIEW and if the data structure is complex.

II. How to incorporate file I/O's into a VI.

- a. Access File I/O operations by:

• **Functions Palette** → **All-Functions** → **File I/O** 

- b. In addition to the File I/O operations, the user will need to identify the File Path so that the VI knows where the file can be accessed. Insert a File Path Control or Indicator by:

• **Controls Palette** → **All-Controls** → **String & Path** → **File Path Control or File Path Indicator**

- c. **Example 6.1:** Modifying the Temperature.VI to Incorporate File I/O

- i. **PROBLEM:** The user is measuring the temperature of a cup of hot water (100 deg F) as it is left to cool in 20 deg F air, probably in a refrigerator. In addition to the experimental temperature, the user has created a file with the theoretical temperatures from time 0 to 100 seconds and would like to compare these values to the experimental values while the experiment is executing. Lastly, the user would like to create a file that stores all the temperature measurement values so that he/she may load them into Excel or another spreadsheet to analyze the data further.
- ii. This example will modify the previously created temperature reading VI to allow the user to access a file with the theoretical temperature values and compare them to the experimental temperature values. Once the measurements are complete, the VI will then append the experimental temperature values to a copy of the accessed file.
- iii. Open "**TempForLoop.VI**" and go to the **Front Panel**.
  1. Insert **2 File Path Controls**, one will be used to identify the file that contains the theoretical temperature data and the other will identify the temperature file which the experimental temperature data will be appended, so name them appropriately.

- **Controls Palette** → **All-Controls** → **String & Path** → **File Path Control** 
2. Insert **2 File Path Indicators**; these will indicate the file that is actually being retrieved by the File I/O operations, so name them appropriately.
    - **Controls Palette** → **All-Controls** → **String & Path** → **File Path Indicator** 
  3. Insert 2 additional **Numerical Indicators**, so the VI can display the theoretical temperature as well as the temperature difference between the experimental and theoretical temperatures. Again, name them accordingly.
  4. Expand the **Top-Right Box** on the **Waveform Chart** so that it includes two plots. This will serve as the legend for the experimental and theoretical temperature plots, so name the two accordingly. *Note: The names should follow the same order as data will be connected in a Bundle Function.*
  5. Arrange the **Front Panel** in some logical orientation. Refer to **Figure 6.1** for some guidance, if necessary.

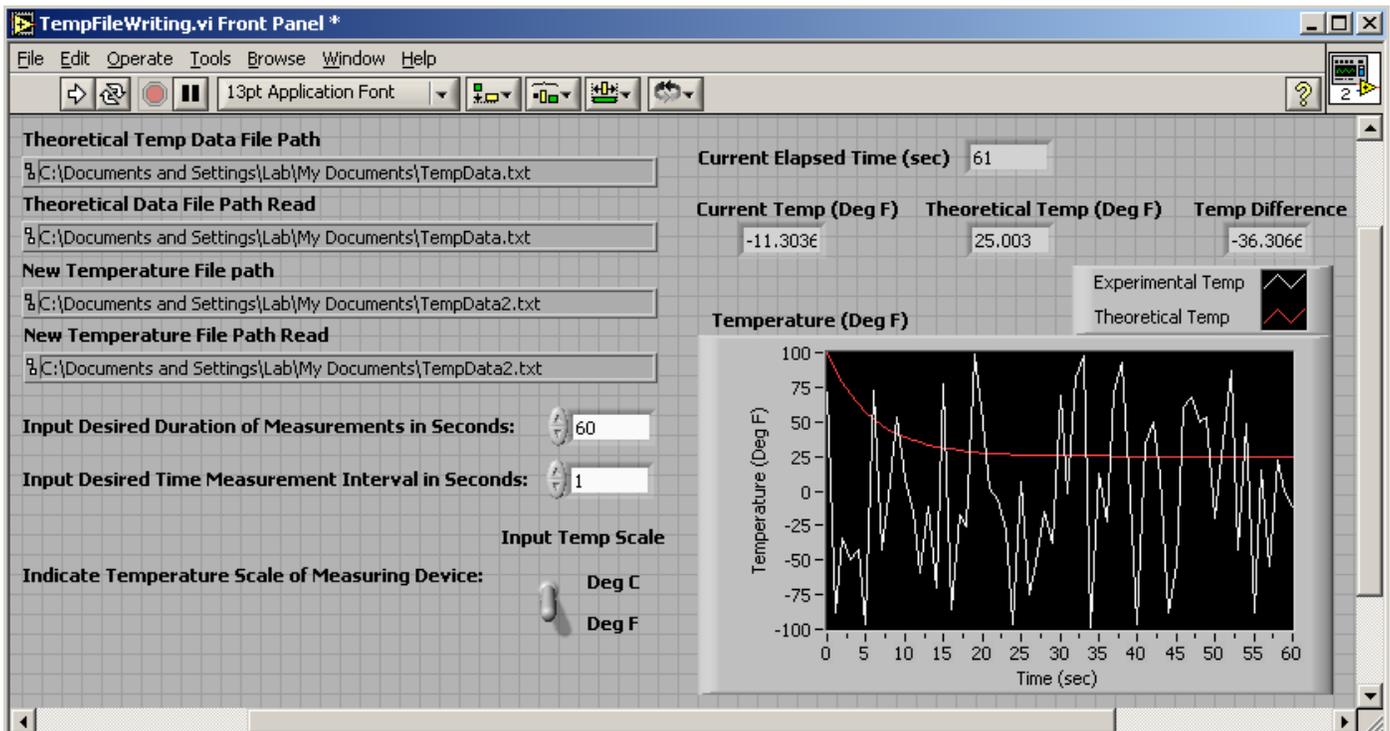


Figure 6.1: Front Panel Configuration for Modified Temperature VI

- iv. Switch to the **Block Diagram**.
  1. Setup the **Theoretical Temperature Reading File Input**.

- a. Insert **Read Characters from File SubVI**  inside the **For Loop**; this will read in the theoretical temperature one value at a time.
    - **Functions Palette** → **All-Functions** → **File I/O**  → **Read Characters from File**
  - b. Add a **Set of Shift Registers**, which will keep track of the character position in the file. (*The SubVI will read a certain length of characters and after it reads those characters, the shift registers will store the placement and the SubVI will start reading from that position in the next iteration.*)
  - c. Create a **Numerical Constant** connected to the **Left Shift Register** and input the value of **295**. (*This indicates where the reading will start; it will skip the first 295 characters, which is the time data.*)
  - d. Wire the **Left Shift Register** to the **“Start of Read Offset” Terminal** (*Bottom Terminal on Left Side*).
  - e. Wire the **“Mark after Read” Terminal** (*Bottom Terminal on Right Side*) to the **Right Shift Register**.
  - f. Create a **Numerical Constant** connected to the **“Number of Characters” Terminal** (*Middle Terminal on Left Side*) and input the value of **7** (*This tells the SubVI to read 7 characters per iteration, since each temperature value contains 6 numbers and a decimal.*)
  - g. Connect the **“Theoretical Temp” File Path Control** to the **“File Path” Terminal** (*Top Terminal on Left Side*) and the **“Theoretical Temp” File Path Indicator** to the **“New File Path” Terminal** (*Top Terminal on Right Side*).
  - h. Insert a **Fract/Exp String to Number Function**  and wire the **File Read Function’s “Character String” Terminal** to **“String” Terminal** of this function, *since the output of the SubVI is a String of Characters and it needs to be in a numeric format, this function will do the needed conversion.*
2. Setup **Waveform Chart to Plot Both Sets of Data**
    - a. Insert a **Bundle Function** that will collect both sets of data into one cluster which can be connected to the Waveform Chart.
    - b. Wire the **“Number” Terminal** from the **String/Number Converter Function** to the **Bottom Terminal** of the **Bundle Function** and the **Experimental “Output Temp” Terminal** to the **Top Terminal** of the **Bundle Function**.

- c. Wire the **Bundle Output Terminal** to the “**Temperature**” **Waveform Chart**.
  - d. Branch a wire from the wire going into the **Bottom Terminal** of the **Bundle Function** and connect it to the “**Theoretical Temp**” **Numerical Indicator**.
3. Setup the **Temperature Difference Numerical Indicator**
  - a. Insert a **Subtraction Function** and branch wires from both the **Experimental Temp Numerical Indicator** and the **Theoretical Temp Numerical Indicator** and connect them to the **Input Terminals** of the **Subtraction Function**.
  - b. Wire the **Subtraction Function’s Output Terminal** to the **Temp Difference Numerical Indicator**.
4. Setup the **Write to Spreadsheet File SubVI**
  - a. Insert **Write to Spreadsheet File SubVI**  outside the **For Loop**, this will append the experimental temperatures to the file specified.
  - b. Branch a wire from the **Experimental Temp Numerical Indicator Wire** and connect it to the “**1D Data**” **Terminal** on the **SubVI**.
  - c. Insert a **Boolean True Constant** and connect it to “**Append to File?**” **Terminal** on the **SubVI**, this indicates that the data will just be appended to the file and nothing will be replaced.
  - d. Connect the “**New Temp**” **File Path Control** to the “**File Path**” **Terminal** (*Top Terminal on Left Side*) and the “**New Temp**” **File Path Indicator** to the “**New File Path**” **Terminal** (*Top Terminal on Right Side*).
5. Arrange the **Block Diagram** in some logical orientation. Refer to **Figure 6.2** for some guidance, if necessary.
6. **Save** as “**TempFileWriting.VI**”. Test your VI using the data files provided (“**TempData.txt**” as the input file and “**TempData2.txt**” as the output file). *Note: If you use different files, you may have to alter the parameters as to how many characters to read and how many to skip.*

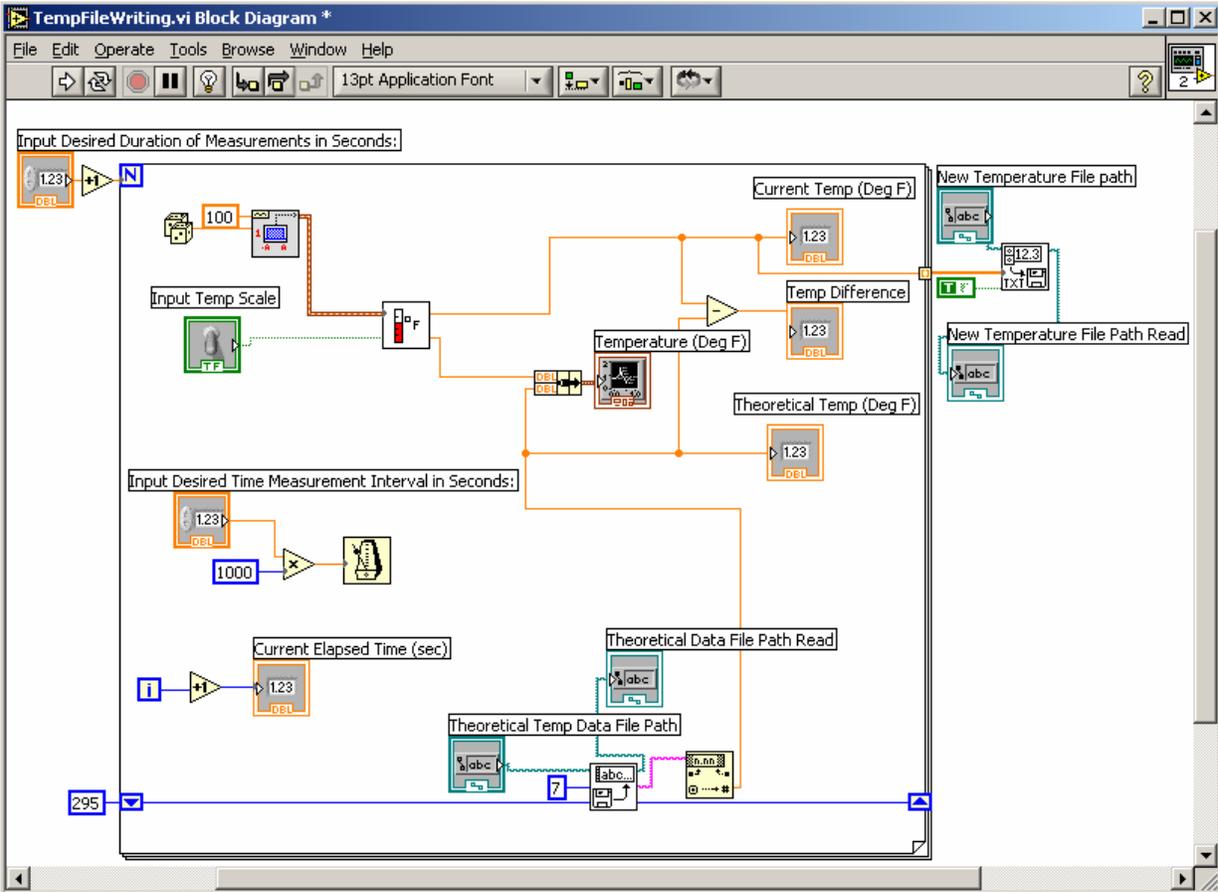


Figure 6.2: Block Diagram Configuration for Modified Temperature Reading VI

### III. What File Operations are Available?

- a. Many different file operations are available. Figure 6.3 below shows a few of the available functions/subVIs as more advanced ones are also available.

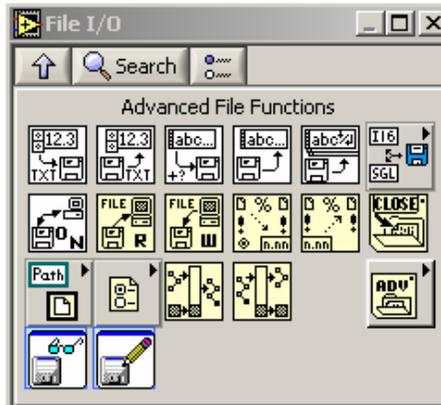


Figure 6.3: Available File I/O Functions/SubVIs

- b. The previous example demonstrated how to use 2 of these functions: **Read Characters from File.VI** and **Write to Spreadsheet File.VI** as well as the **Fract/Exp String to Number Function**.

#### IV. What Types of Plots are Available and When to use them?

- a. The following plots are available under **Controls Palette** → **Graph Inds** and are the most widely used:
  - i. **Waveform Chart** – A numerical indicator that is used to display single or multiple plots and receives numerical data one value at a time. These types of plots are good to use inside loops.
  - ii. **Waveform Graph** – Also a numerical indicator that is capable of displaying single or multiple plots; however, it receives data in an array and then plots it. These types of plots are good outside of loops and if you are not concerned with the data until all the data is collected.
  - iii. **XY Graph** – Another numerical indicator that accepts clusters that contain an x array and a y array. These types of plots are good for comparing data that is not based on time. (*i.e. force versus distance or experimental versus theoretical*)
- b. All available plots can be accessed under **Controls Palette** → **All-Controls** → **Graph**, which include the following:
  - i. **Intensity Charts/Graphs**
  - ii. **Digital Waveform Graph**
  - iii. **3D Graphs**

#### V. What Plotting Options are Available/How to Edit Plots?

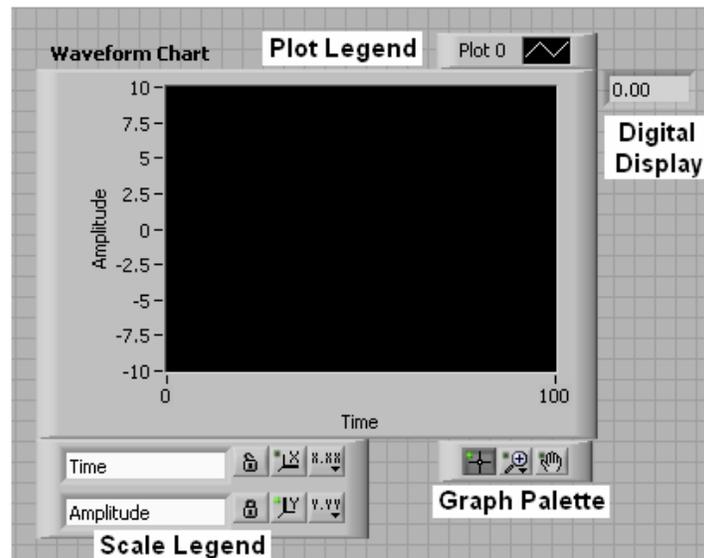


Figure 6.4: Waveform Chart Plotting Controls

- a. All of the controls labeled in **Figure 6.4** can be displayed for a Waveform Chart, if not already, by **Right-Clicking on the Graph** → **Visible** → **Select Component**

- i. The **Plot Legend** shows the different plots being displayed on the graph or chart.
  1. The defaults are Plot 0, Plot 1, Plot 2, and so on; which can be changed by **Double Left-Clicking** on the **Name** (*Plot #*).
  2. **Right-Click** on the **Plot Legend** to change the type of plot, the color of the plot lines, the type of data points (useful when plotting multiple sets of data), and many other options. **Figure 6.5** shows the drop-down menu and some of the available options.

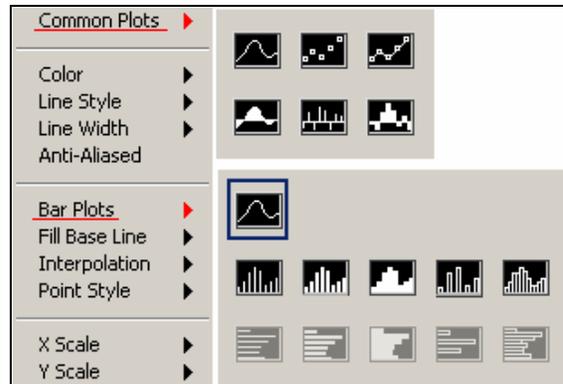


Figure 6.5: Plot Legend Options

- ii. The **Digital Display** is available only for **Waveform Charts** as it will display the current data value being plotted.
- iii. The **Scale Legend** allows the user to change the title of each axis, lock the scales, autoscale each axis, and change the format of each scale. Refer to **Figure 6.6** for option locations.

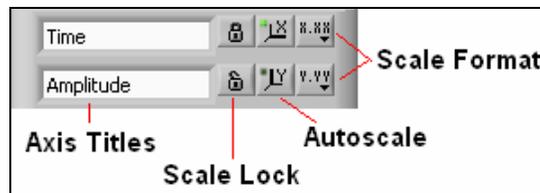


Figure 6.6: Scale Legend Options

- iv. The **Graph Palette** allows the user to place a cursor on the plot with the **Cursor Movement Tool**, zoom in or out on the data with the **Zoom Tools**, and move the graph with the **Panning Tool**. Refer to **Figure 6.7** for the location of these options as well as all the zooming options.

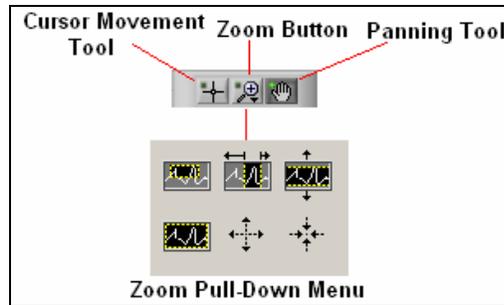


Figure 6.7: Graph Palette Options

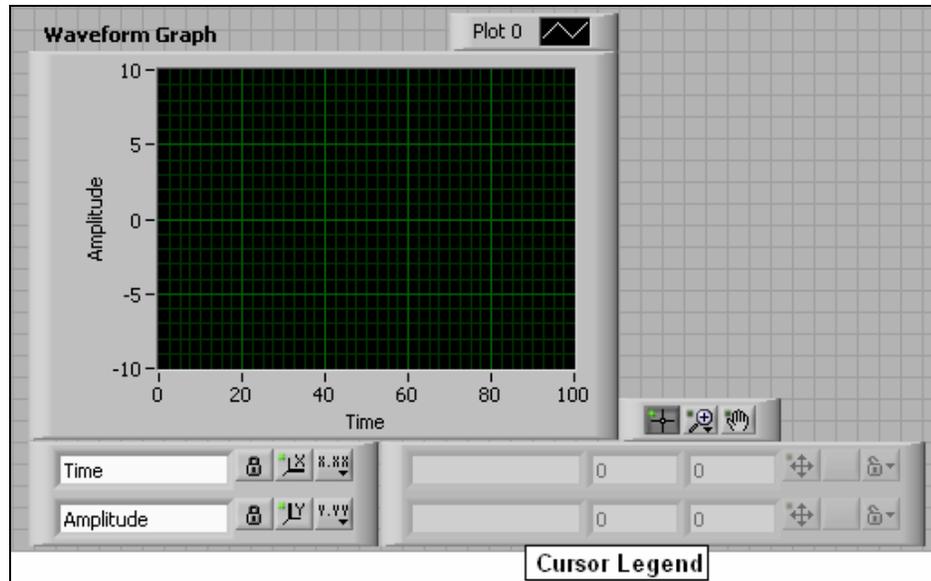


Figure 6.8: Waveform Graph Plotting Controls

- b. All of the controls labeled in **Figure 6.8** can be displayed for a Waveform Graph, if not already, by **Right-Clicking on the Graph** → **Visible** → **Select Component**
  - i. All of the controls are the same as for the **Waveform Chart**, except the **Cursor Legend**.
  - ii. The Cursor Legend displays (from left to right) the name of the cursor, the position (x and y coordinates), a toggle switch to turn the cursors on and off, cursor display options (color, point style, etc), and a cursor lock.
- c. Other options/properties can be accessed by **Right-Clicking** on the **Graph** → **Properties**, which brings up set of menus including: **Appearance**, **Format and Precision**, **Plots**, **Scales**, **Cursors (only for Waveform Graph)**, and **Documentation** as shown in **Figure 6.9**.

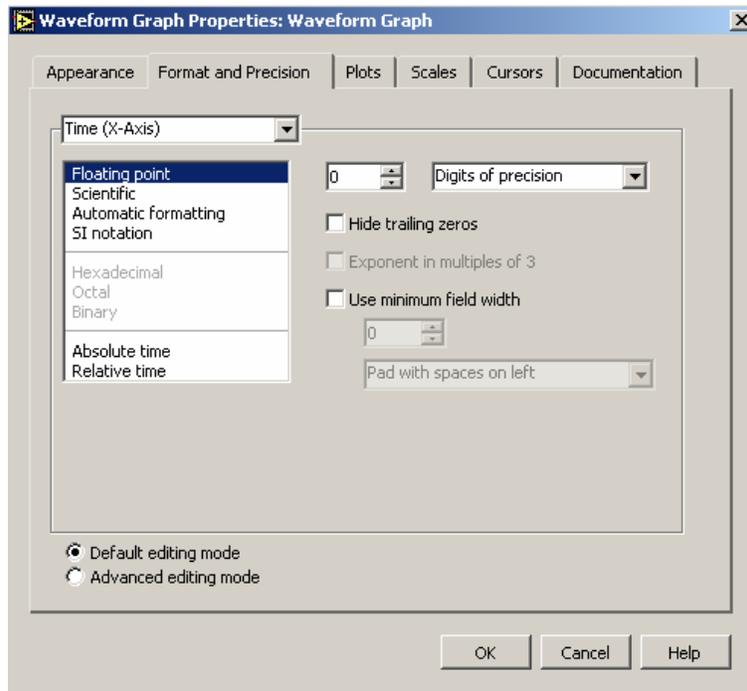


Figure 6.9: Waveform Graph Properties Window

- d. Lastly, for Waveform Charts, the plotting or updating mode can be either **Strip Chart**, **Scope Chart**, or **Sweep Chart**, which can be accessed by **Right-Clicking on the Graph** → **Advanced** → **Update Mode** → **Type of Updating Desired**

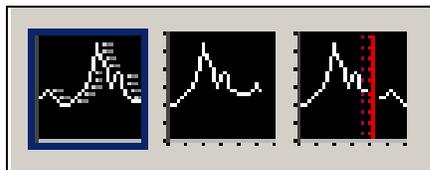


Figure 6.10: Update Modes

(From Left to Right: Strip Chart, Scope Chart, and Sweep Chart)

- i. The **Strip Chart** shows running data continuously by scrolling the data from left to right across the graph.
  - ii. The **Scope Chart** shows one item of data that scrolls partway across the graph from left to right.
  - iii. The **Sweep Chart** has a vertical line that sweeps across the graph refreshing the data, so the older data is on the right and the newer data is on the left.
  - iv. The **Scope** and **Sweep Charts** display data faster than the **Strip Chart**.
- e. For practice, open up the VI you just created “**TempFileWriting.VI**” or one of the others you created that has a plot and just experiment with the different controls to see how each one works.