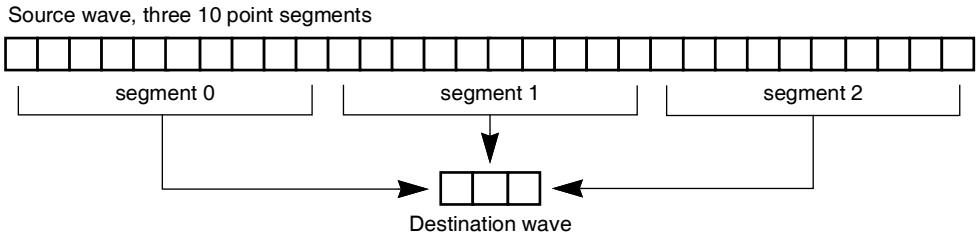
This diagram illustrates a source wave with three ten-point segments and a destination wave that will contain the mean of each of the source segments. The FindSegmentMeans function makes the destination wave.



To test FindSegmentMeans, try the following commands.

```
Make/N=100 wave0=p+1; Edit wave0
FindSegmentMeans(wave0,10)
Append wave0_m
```

The loop index is the variable “segment”. It is the segment number that we are currently working on, and also the number of the point in the destination wave to set.

Using the segment variable, we can compute the range of points in the source wave to work on for the current iteration: segment*n up to (segment+1)*n - 1. Since the mean function takes arguments in terms of a wave’s X values, we use the pnt2x function to convert from a point number to an X value.

If it is guaranteed that the number of points in the source wave is an integral multiple of the number of points in a segment, then the function can be speeded up and simplified by using a waveform assignment statement in place of the loop. Here is the statement.

```
destw = mean(source, pnt2x(source,p*n), pnt2x(source,(p+1)*n-1))
```

The variable p, which Igor automatically increments as it evaluates successive points in the destination wave, takes on the role of the segment variable used in the loop. Also, the startX, endX and lastX variables are no longer needed.

Using the example shown in the diagram, p would take on the values 0, 1 and 2 as Igor worked on the destination wave. n would have the value 10.

Working with Mismatched Data

Occasionally, you may find yourself with several sets of data each sampled at a slightly different rate or covering a different range of the independent variable (usually time). If all you want to do is create a graph showing the relationship between the data sets then there is no problem.

However, if you want to subtract one from another or do other arithmetic operations then you will need to either:

- Create representations of the data that have matching X values. Although each case is unique, usually you will want to use the **Interpolate2** operation (see **Using the Interpolate2 Operation** on page III-111) or the interp function (see **Using the Interp Function** on page III-110) to create data sets with common X values. You can also use the **Resample** to create a wave to match another.
- Properly set each wave’s X scaling, and perform the waveform arithmetic using X scaling values and Igor’s automatic linear interpolation. See **Mismatched Waves** on page II-83.

The WaveMetrics procedure file Wave Arithmetic Panel uses these techniques to perform a variety of operations on data in waves. You can access the panel by choosing Packages→Wave Arithmetic from the Analysis menu. This will open the procedure file and display the control panel. Click the help button in the panel to learn how to use it.