```
   // Make output wave based on the first source wave
   Wave first = waves[0]
   Duplicate/O first, $outputWaveName
   Wave wOut = $outputWaveName
   wOut = 0

   Variable numWaves = numpnts(waves)
   Variable i
   for(i=0; i<numWaves; i+=1)
      Wave source = waves[i]
      wOut += source          // Add source to output
   endfor

   wOut /= numWaves           // Divide by number of waves

   return wOut
End
```

This function shows how you might call WavesAverage from another function:

```
Function DemoWavesAverage()
   Make/FREE/N=10 w0 = p
   Make/FREE/N=10 w1 = p + 1

   Make/FREE/WAVE waves = {w0, w1}

   Wave wAverage = WavesAverage(waves, "averageOfWaves")
   Display wAverage
End
```

Finding the Mean of Segments of a Wave

An Igor user who considers each of his waves to consist of a number of segments with some number of points in each segment asked us how he could find the mean of each of these segments. We wrote the FindSegmentMeans function to do this.

```
Function/WAVE FindSegmentMeans(source, n)
   Wave source
   Variable n

   String dest                    // name of destination wave
   Variable segment, numSegments
   Variable startX, endX, lastX

   dest = NameOfWave(source)+"_m"  // derive name of dest from source
   numSegments = trunc(numpnts(source) / n)
   if (numSegments < 1)
      DoAlert 0, "Destination must have at least one point"
      return $""                   // Null wave reference
   endif

   Make/O/N=(numSegments) $dest
   WAVE destw = $dest
   for (segment = 0; segment < numSegments; segment += 1)
      startX = pnt2x(source, segment*n)      // start X for segment
      endX = pnt2x(source, (segment+1)*n - 1)// end X for segment
      destw[segment] = mean(source, startX, endX)
   endfor

   return destw
End
```