Engin 103 October 30, 2008 back to e-syllabus	Topics: CW8 Plotting a Function with LabVIEW I Logbook questions

Engineering 103 –UMass Boston

CW 8

(In-Class-Work 8)

Plotting a function using 1) For Loop; 2) Evaluation of Single Variable Array; 3) Waveform Graph (see class note for additional instructions)

a) Make a VI that can plot any function between 0 and 4 (using N=101 points, for example). Use LabVIEW For Loop, Eval Single-Variable Array (version 7.1: under Analyze/Mathematics/Formula/Advanced Formula Parsing; version 8.2: under Mathematics/Scripts&Formulas/1D&2D Eval), and Bundle (under Cluster) to tell the Waveform Graph the initial value, increment for t (the function's argument), and values of the function. Use cos(t) as the input function to be plotted. Click on the Light Bulb in the Diagram, and then click Run to see how the For Loop and the program work. Save the VI into a LLB file named cw8_XX_Y, as "part a" after selecting Files/Make Current Values as Default.

b) Make another VI by modifying the VI in a) such that it uses only 5 points. Note the difference. Then use trial-error method to see what is the smallest number of points that would make the graph look like the actual function. Save the VI into the same LLB file (cw8_XX_Y) as "part b" after doing Files/Make Current Values as Default with this smallest number of points

Please insert names and dates within the Front Panels. In each team, students working together at a computer numbered between 1 and 10 will submit LabVIEW LLB file cw8_XX_a.llb, students working at a computer numbered between 11 and 20 will submit LabVIEW LLB file cw8_XX_b.llb, to the *files* folder in the server. Replace XX by 01 if team 1, etc. Each LLB file should contain two VI's, named 'cw8a' and 'cw8b' corresponding to this CW. These files need to be uploaded to the server today to receive credit.

Include your names within the files.

*Remember that this is an individual work (turn it in, as instructed, with your name and date). Home-works and class-works count 20% toward the course grade. Class-works are done in class.

back

Plotting a Function with LabVIEW I: 1)For Loop 2) Eval Single Var. Array 3) Waveform Graph:

-The computer is very good in automatic generation of long series of numbers: a) start with

an initial value b) use an increment to produce the next value c) how many numbers in the series

To plot a function f(t) between ta (initial value) and tb (final value) using N points we will generate a time series (horizontal axis)

$$ti = ta + i*\Delta$$

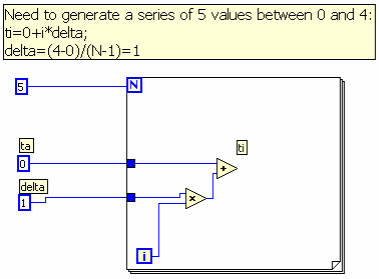
where i =0, 2, …, N-1 (with Δ the increment; Δ=(tb-ta)/(N-1);
t1=ta and tN=tb); ti: the next value: or value "i" is equal to the initial value ta plus "i" times the increment.

For example, we would like to generate N=101 values, with initial value of 0; final value of 4, what is the formula?

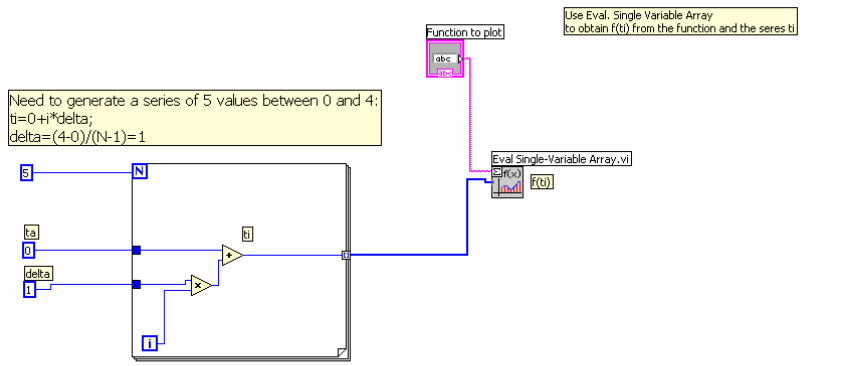
ti= 0 +i*Δ= i*0.04 (i= 0,1,2,3, ..., 100)
Δ=(tb-ta)/(N-1) = (4-0)/(101-1)=0.04

What if we would like a different series, starting from 4, going to 10; using 37 values?
ti=4+i*(10-4)/36=4+i*1.6667;

To do this in LabVIEW we will use the FOR LOOP



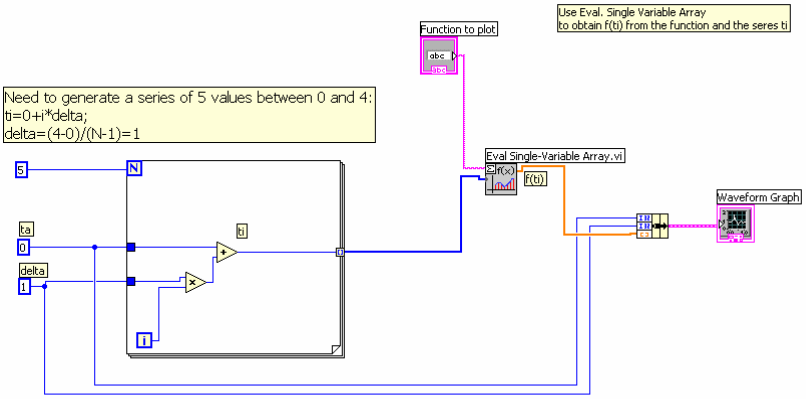
To plot f(t) (vertical axis of our 2D Waveform Graph) versus t (horizontal axis of our Graph), we now need to evaluate f(t) at ti, i.e. obtaining f(ti). Then plot the points (ti, f(ti)) using the Waveform Graph. LabVIEW automatically connect these points with lines. We will enter a function using a String Control in the Front Panel.
Eval. Single Variable Array
(version 7.1: under Analyze/Mathematics/Formula/Advanced Formula Parsing; version 8.2: under Mathematics/Scripts&Formulas/1D&2D Eval) takes as inputs the function f(t) and the input series ti , and produces the output series f(ti)

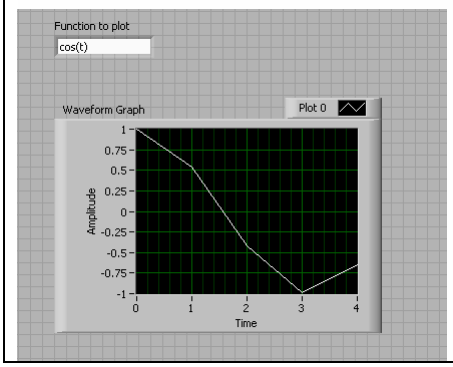


Use Waveform Graph, it has one input terminal, but requires three information:

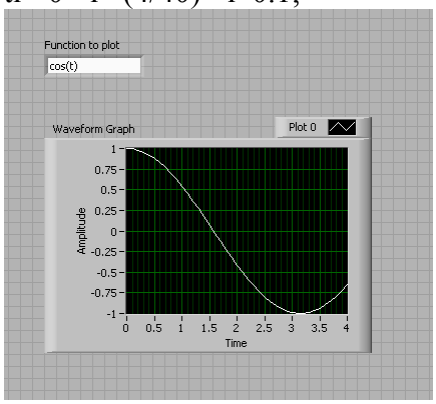
1) Initial value for the horizontal axis
2) Increment for the time series
3) f(ti)

To "bundle" these 3 information into one terminal, we will use "bundle"
Bundle (under Cluster, then right-click on its left side to "Add Input" to have three input terminals, since the Waveform Graph requires three inputs in this order: ta, Δ, and f(ti))



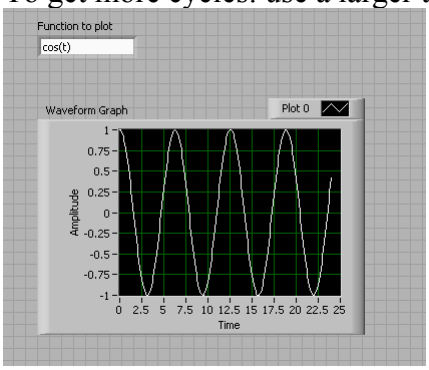


To get a better curve for the cosine function: we need to change the time series:
we need to change N, since the graph shows just 5 points (ti, f(ti)) connected with lines.
N=41;
ti= 0 +i* (4/40)= i*0.1;



If the number of points is large or
the increment Δ is small, then the lines are so short that the overall curve looks smooth which resembles the function we are trying to plot.

To get more cycles: use a larger tb by using more points with the same delta



back

back

back

LOGBOOK: example of a logbook page

-Use a quadrille notebook; number all pages; date all entries
-Write your notes for all activities, thoughts, problems and solutions, and learning conclusions related to Engin 103. You should write down progress, outcomes, and conclusions on projects and teamwork; conclusions from class work (including LabVIEW) and homework.
-In addition you should answer in the logbook all questions listed in these notes in blue, as shown below:

31) LabVIEW: in this Classwork what formula we tried to implement using the For Loop?, What are the 'N' and 'i' for? Once we have the time series (horizontal axis in the Waveform graph), how did we calculate values of the function f for each element of the time series to get f(ti) (vertical axis in the Waveform graph)?

32) Why do we need a 'Bundle' for the Waveform graph? How many inputs should the 'Bundle' have and what should be connected to those inputs?

back