

[Beacon Engineering Consortium](#)

Educators and students can get a competitive edge for the future by learning to code using MS Excel. The data generated will be used to graph Physics motions by using pre-written macro in Visual Basic for Applications (VBA) to solve the Physics problems. A VBA macro is computer code that is available to do repetitive math operations. VBA is available for all Microsoft Office Products such as Excel, Word, Access etc. You can modify this macro to solve other Physics problems.

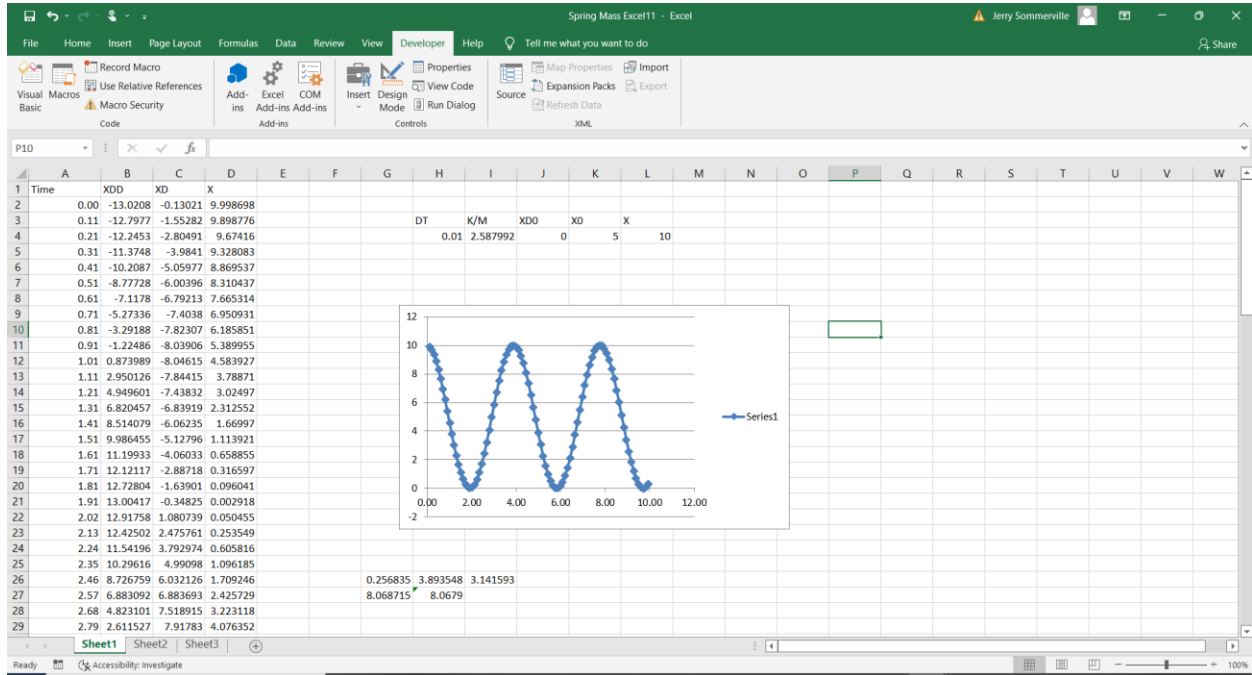


Figure 1

We will use some pre-written VBA computer code of a spring mass system to demonstrate just how useful this knowledge can be. You will be able to run circles around those that practice Physics the old-fashioned way with pencil and paper.

An explanation of the code is provided below to help you get started and to help you apply the approach to other problems.

This topic is an extension of [Coding CANVAS where computer coding is encouraged in Algebra classes](#). In this instance, we demonstrate how computer coding should be extended to Physics classes to prepare students for Virtual Reality Programming used in [real time robot control](#).

Excel files with macros have the extension .xlsm. If you have a computer anti-virus program it will likely quarantine an .xlsm file when you download it. Some macros can do your computer harm, if you are not careful to guard against malicious code.

The Excel macro approach is a particularly useful alternative to students that do not have access to tools like Matlab/Simulink and MathCad.

Math Strategy for Solving Physics Problems:

This strategy is the same for all Newtonian mechanics problems.

In the equation $F=M*a$, the acceleration term, a , is the second derivative of position:

1. In Macro, solve for the acceleration term in your [$F=M*a$, $F=M*XDD$ equation], “a” or “XDD” term where D represents the derivative
2. Integrate the acceleration term, XDD, to find the velocity, XD
 - a. Backwards rectangular integration is used for this example, where DT represents the time step
 - b. You can play around with this variable a bit to see if the answer changes appreciably when going up or down in size.
3. Integrate the velocity term, XD, to find the position, X
4. Record data to Excel spreadsheet columns using Cells(Row, Col).Value function
 - a. column 1 represents Time in seconds
 - i. adjust “Time_” variable in macro to represent how long the motion will occur
 - b. column 2 represents acceleration, a, XDD
 - c. column 3 represents velocity, XD
 - d. column 4 represents position, X
5. Don’t record every point from your calculations (too much data)
 - a. Instead pick an appropriate sample rate, Tsamp, that shows trends of motion
 - b. Adjust variable to sample motion every “Tsamp” seconds
6. Use Excel to plot position vs. time, for example, or any other parameter vs time.
7. Check if simulation frequency matches theoretical equation for frequency
 - a. $2*\pi*freq = \omega = \sqrt{K/m} = \sqrt{1000/32/12} = 1.61 \text{ rad/sec}$
 - b. $freq = 1.61/(2*\pi) = 0.257 \text{ Hz}$
 - c. From graph, Figure 1, above and screen shot of data below, Figure 2, one cycle ends in 3.89 seconds
 - d. $Frequency = 1 \text{ cycle}/(3.89 \text{ seconds}) = 0.257 \text{ Hz}$,
 - e. Excellent agreement!

36	3.56	-11.234	4.023732	9.354087	
37	3.67	-12.2201	2.725457	9.719771	
38	3.78	-12.8221	1.341525	9.93712	
39	3.89	-13.0212	-0.08457	9.999303	
40	4.00	-12.8111	-1.508	9.904365	
41	4.11	-12.1983	-2.88405	9.655291	
42	4.22	-11.2021	-4.16945	9.259908	

Figure 2

To get to the VBA macro in Excel, you have to go to the “Developer Tab”. (See Screen shot above, Figure 1)

I suggest that you Google Excel Macros to get some more background on how to access the “Developer Tab” ... Some Excel Settings may need to be changed.

Click Visual Basic Icon in Developer Tab to see the VBA source Code. Comments about on each line are provided in green as explanations that correlate to the **Math Strategy** above.

```

Microsoft Visual Basic for Applications - Spring_Mass_Excel_2.xlsm
File Edit View Insert Format Debug Run Tools Add-Ins Window Help
Ln 12, Col 12
Project - VBAProject
VBAProject (Spring_Mass_Excel_2.xlsm)
  Microsoft Excel OT
  Sheet1 (Sheet1)
  Sheet2 (Sheet2)
  Sheet3 (Sheet3)
  ThisWorkbook
  Forms
  Modules
  Module1
Spring_Mass_Excel_2.xlsm - Module1 (Code)
(Springmass)
Sub Springmass() ' Beginning of sub
    DT = 0.01 'integration step
    KDM = 1000 / 32 / 12 ' Spring constant ,K, divided by mass
    XD = 0 'initial velocity
    X = 10 'initial position
    Tsamp = 0.1 'Sample period seconds
    tlst = -1 ' Last Sample Time
    i = 2 'current row of Excel spreadsheet being recorded
    TStart = 0 ' Start time, seconds
    TEnd = 10.2 ' End time, Seconds

    For Time_ = TStart To TEnd Step DT ' Start of integration loop

        XDD = KDM * (5 - X) ' acceleration XDD= F/m= K*(x-x0)/mass
        XD = XD + XDD * DT ' integrate acceleration to get velocity
        X = X + XD * DT ' integrate velocity to get position
        If (Time_ - tlst) > Tsamp Then 'Don't sample every point

            Cells(i, 1).Value = Time_
            Cells(i, 2).Value = XD
            Cells(i, 3).Value = XD
            Cells(i, 4).Value = X

            tlst = Time_
            i = i + 1
        End If

    Next Time_ 'End of integration loop

End Sub 'End of Sub

```

Figure 3

Press Run icon to run the macro

Press Stop icon to stop the macro

Enjoy.

May you run circles (loops) around your classmates with your newly found skills to solve Physics problems in Excel.

You can download the Spring Mass Demonstration Excel file [here](#).