Autodesk Inventor Tutorials

by Sean Dotson, PE
www.sdotson.com
sean@sdotson.com

Linked & Embedded Parameters – Part Two

Latest Revision: 9/25/02

In the first part of this tutorial we discussed how to embed Excel sheets into parts to control the size and shape. In this lesson we will discuss linking Excel sheets into multiple parts as well as the top level assembly to control entire assemblies. While a lot of what we will discuss could be achieved through adaptivity, linking spreadsheets can give you a predefined set of assembly configurations that a user of any skill level can use.

In this example we will be building a simple machine frame assembly. We will be able to change the size and shape of the frame based off one mouse click in the Excel document.

Let's begin our project by creating our Excel sheet. In this example let's start the parameters on row 10 to leave some space for formatting above (see Figure 1). Since we are going to have three different tubes making our frame we have parameters for the length and width of each of the tubes (*tubelen* and *tubewidth* respectively). (In reality the tubing thickness might change as well but for the purposes of this example we are going to keep it simple.) Enter the parameter names, equations, units and optional comments into the Excel sheet. Above the parameters I have formatted some use friendly cells. Typically I like to highlight the ones where uses can change values. Also notice that we warned the users to not change the data below a certain line.

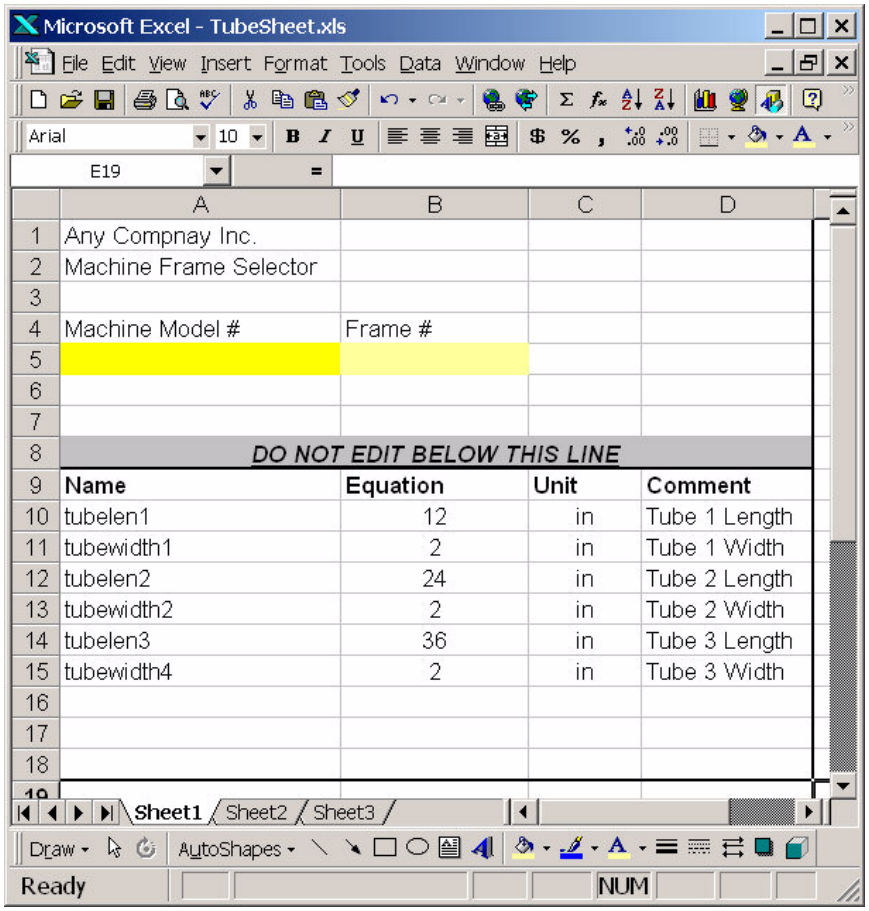

Figure 1 - Excel Sheet Format

Now start a new part file and then click on the parameters button. As before choose the link button. When prompted select the Excel file and choose Link (not embed). We do this as we want the data to be shared among several files and not embedded in just this one. Also be sure to set the start cell to A10. (See Figure 2)

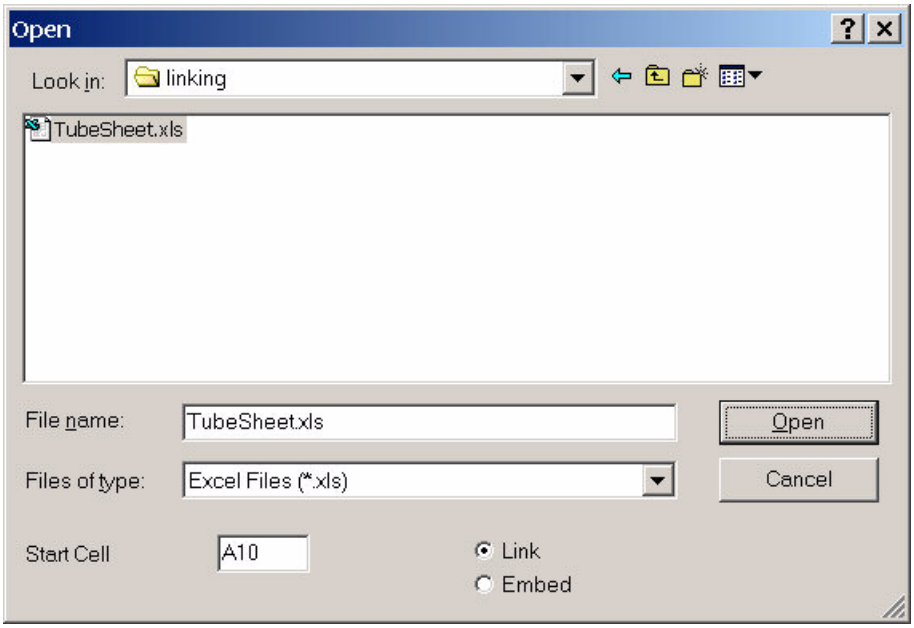

Figure 2 - Linking the Excel Sheet

A quick look at the parameter dialogue shows that the parameters were successfully imported. Now draw the shape bellow and give it a height and width of "tubewidth1". Extrude the profile a length of "tubelen1". You should end up with a part like the one in Figure 3. Save this part as Tube1.ipt

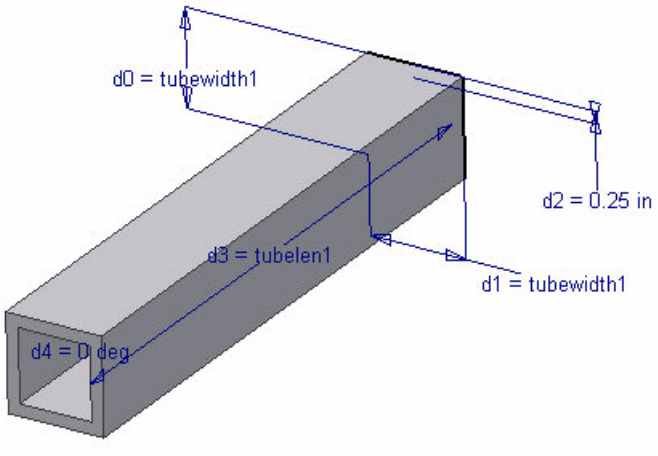

Figure 3 - Tube1 Dimensions

The quick and easy way to create the other two tube parts is to issue the "Save Copy As.." command and save copies of Tube1.ipt as Tube2.ipt and Tube3.ipt Now open these parts and edit the sketch and extrusion feature dimensions to be equal to "tubelen2","tubewidth2 etc.. for each part. While in each part, also change the color so you can tell which is which in the assembly. If the parts in your assembly are not similar you can simply link the Excel spreadsheet to each of them individually. Remember that you can have parameters linked via an Excel spreadsheet that are not used in one part model but are in another.

Save all of these files and begin an assembly. Insert the tubes into the assembly and construct the frame as shown in Figure 4. Be sure to constrain the tubes in such a manner that the length or widths of the tubes is not used. In other words use mate and flush commands with a 0 value to avoid any problem when we change the tube lengths or widths.

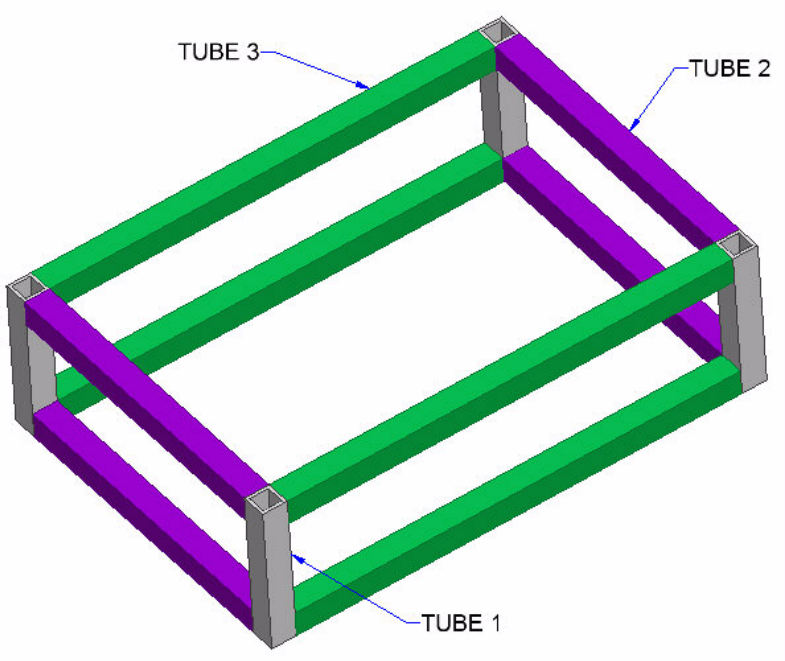

Figure 4 - Basic Machine Frame

We now want to link the Excel sheet to the Top Level assembly. If you were not going to base any further geometry off the values of the lengths and/or width of the tubes then it would not be necessary to link the Excel file to this assembly. However in our case we will be adding one more tube member. Also it makes changing the assembly much easier as it is available in the assembly file from the browser. Using the procedure described above, link the Excel sheet into the top-level assembly.

Now insert one more instance of Tube2.ipt. Constrain it to be flush with the bottom of the frame and mated at one end to one of the bottom Tube1's. We now want to center it in the middle of the frame. Place a mate between one face of the latest Tube2 and one of

the existing bottom Tube2s and give it a value of *tubelen3/2 – tubewidth2/2*. (See Figure 5)

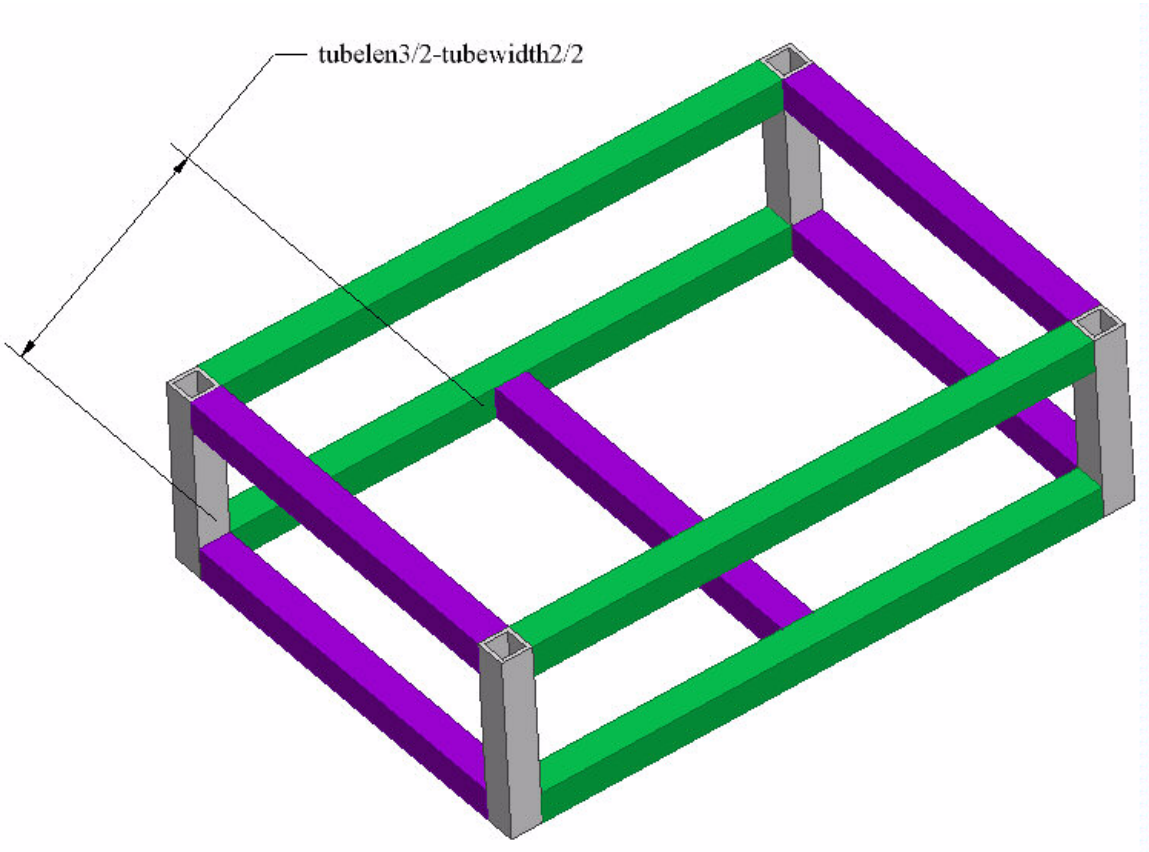

Figure 5 - Adding the Middle Tube Member

Without linking in the top level sheet we would not have been able to refer to the parameters that describe the lengths and widths of the tubes. Now the middle tube will remain centered regardless of the size of the frame.

As it stands now we can enter the browser and under 3rd Party, right click on TubeSheet.xls and choose edit. We can manually edit the parameters, click save[1] and then return to Inventor to hit update and see the changes in the model. This is a perfectly acceptable use of linking Excel sheets. However in this tutorial we are going to take it a step further and automate the creation of the frames based on machine model numbers.

Go back into your Excel sheet and add the information at the bottom of the sheet as shown in Figure 6.

[1] There is a bug in the current release of Inventor (5.3) that does not update the model with the correct parameters each time you save the Excel file. A few workarounds are to click the save button several times in a row or to save the Excel file, save the assembly, close IV and then reopen Iv and the assembly file. You may have to play around with it to get this to work correctly each time.

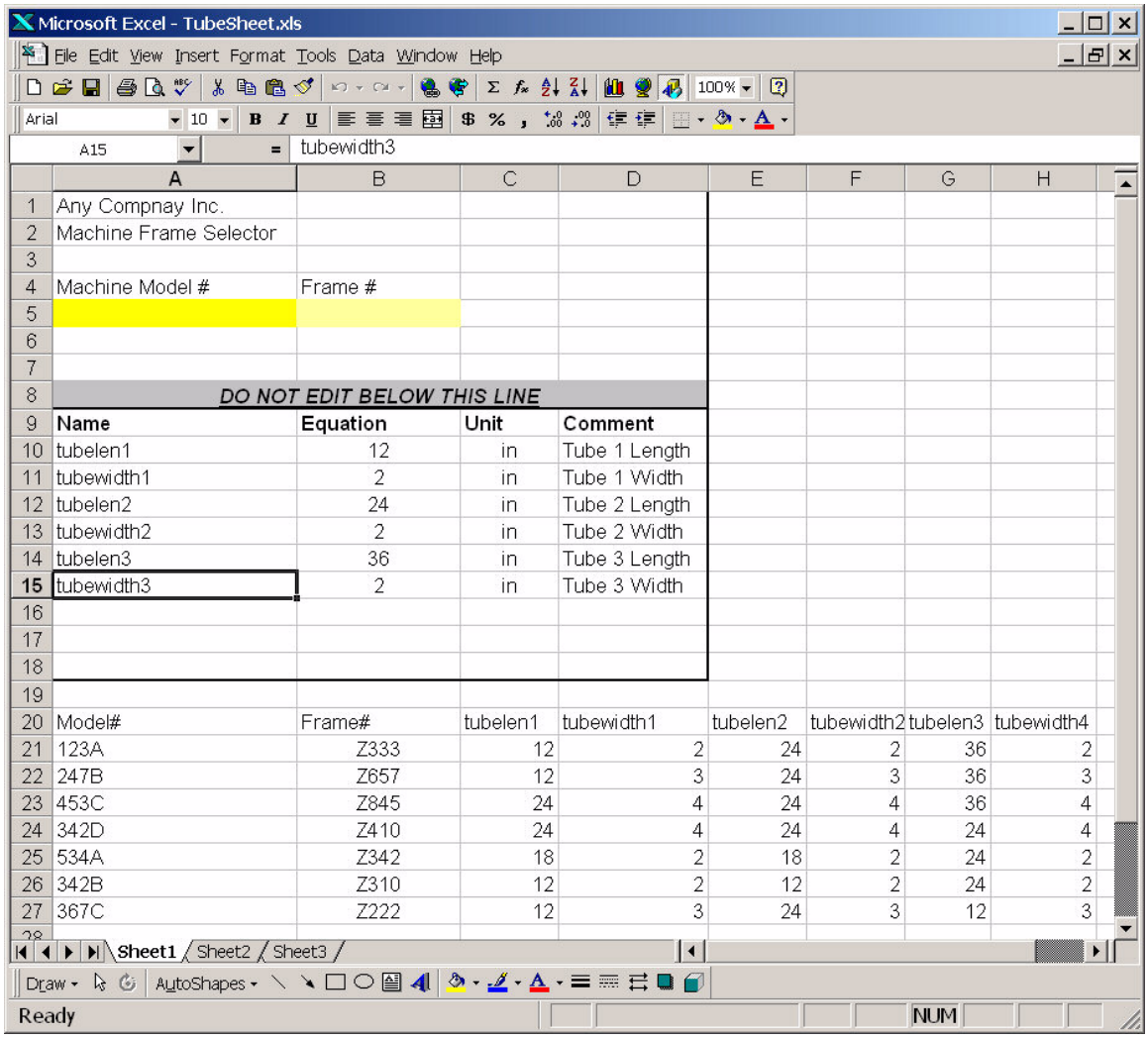

Figure 6 - Adding Machine Size Data

In this section we have added a chart that describes the value of the parameters and frame # based on the machine # (by the way this is all fictitious data). Note that the parameter names at the top of the columns are not necessary but help to keep the data straight.

In the yellow Machine Model# cell use the Data>Validation command to create a pulldown list of the machine models. This process was described in Part One of this tutorial.

Next we are going to explore a very powerful Excel command named VLOOKUP. This command looks for a specified value in a chart and returns another value based on the first one. However the first row of data must be in ascending order, so select our chart cells A:20 – H:27 and select Data>Sort from the top menu. Choose to sort by Model# and choose ascending. This sorts the chart into ascending order by Model#.

We can now focus on the VLOOKUP command. Select the yellow cell under Frame# (at the top) and then hit the = button next to the cell editing bar. Next use the pulldown to the left to select More Functions in the left window and then find VLOOKUP in the right (see Figure 7)

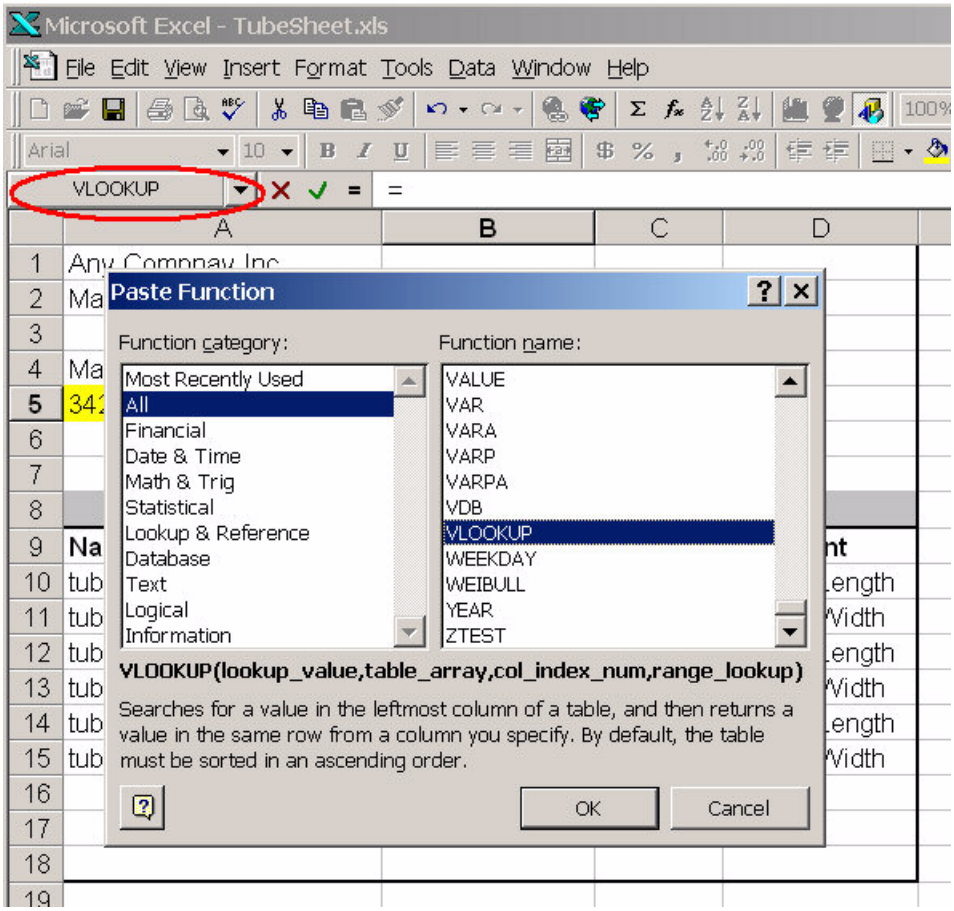

Figure 7 - Using the VLOOKUP Command

After issuing the command a helper window will appear. It should look like Figure 8. In the first row click the icon to the right and select the Lookup_value. This is the value we will key off of. Select cell A5. In the next row we want the value of the data table in which we are going to look up this value. In this row click the icon and select the rows A:21-H:27. In the third row we need to select which column value we want to return. In this case we want to know the Frame# so we enter 2 (the second column). You can leave the fourth row blank. Click OK and the command will exit.

You can now select a machine # via the pulldown menu and the frame # should change based on this machine # value. Now we want to extend this logic to the parameters. Select cell B10 and issue the VLOOKUP command once more. Fill out the dialogue as shown in Figure 9, paying special notice to the addition of the $ symbol.

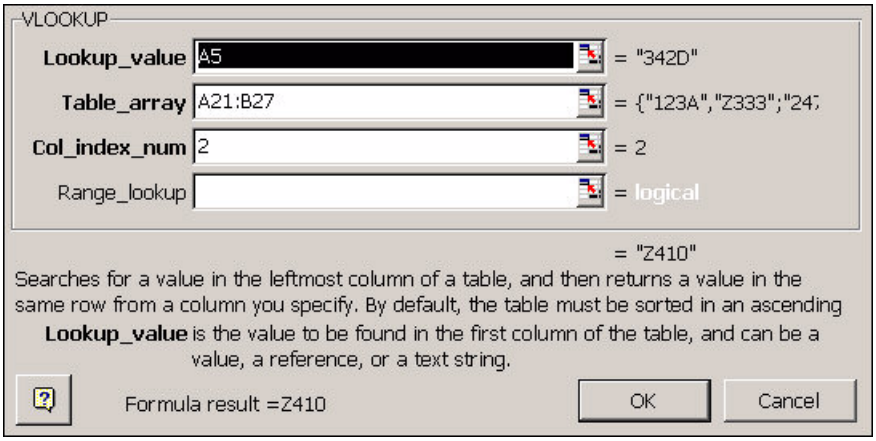

Figure 8 – Filling out the First VLOOKUP Table

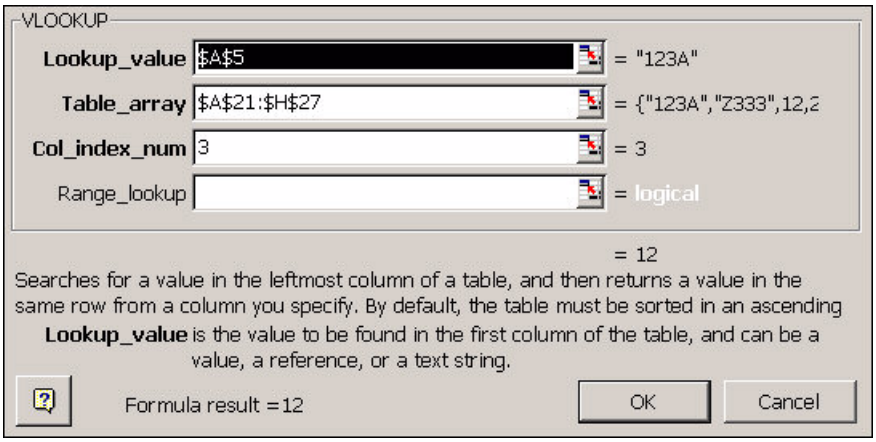

Figure 9 - Filling out the Second VLOOKUP Table

The purpose of the $ is to tell Excel that we want to lock the row and column values of these entries. This way we can copy cell B10 into B11-B15 and the values of the table array and lookup value will not increment along with them. (This is a fundamental Excel function. If you do not understand it I suggest looking at Excel's online help)

So you can now copy this information into cells B11-B15. You must however edit each of these new cells and change the *col_index_num* to the correct value for each parameter. You can do this by selecting the cell and hitting the = in the edit bar (which will bring up the helper window) or by editing the formula directly in the edit bar.

The last thing to do is to make sure the parameter cells are formatted as numbers. RMB on the cells then Format Cells/Number/Number.

If you have successfully edited the formulas you should now be able to pull down a value in the Machine Model # cell and the parameter values will change to reflect the changes. The advantage of this type of setup is that you can add additional frame sizes and the assembly will update to these values as long as you edit the formulas to include the new values in the table.

Now change the value of Machine Model # to a new value and hit save[2]. Return to IV and hit update and the frame should have changed size. Try playing around with the different sizes. You can quickly see that many different styles and shapes of assemblies can be made from one simple selection in the Excel sheet. Figure 10 illustrates some of the forms that our frame can take.

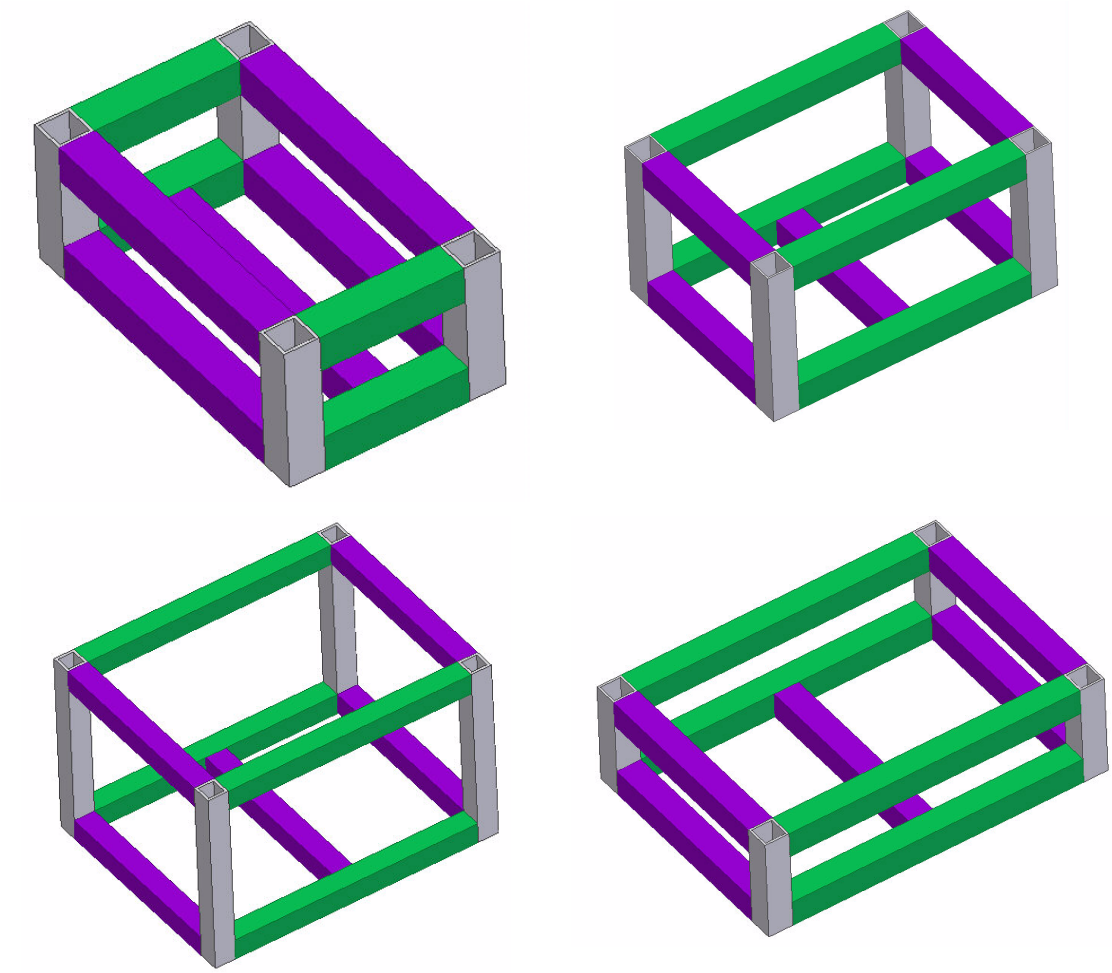

Figure 10 - Examples of Frame Shapes

I hope this tutorial has taught you the power of linking and embedding Excel sheets into parts and assemblies. There are additional features that can be accomplished by linking spreadsheets to parts but I leave some of that exploration to you. I suggest checking out some of the Excel driven parts found on Charles Bliss' site: located at:
http://www.cbliss.com/inventor/

[2] There is a bug in the current release of Inventor (5.3) that does not update the model with the correct parameters each time you save the Excel file. A few workarounds are to click the save button several times in a row or to save the Excel file, save the assembly, close IV and then reopen Iv and the assembly file. You may have to play around with it to get this to work correctly each time.