

Scripting How-To: How Can I Get Information About the Layout Cell Associated with a Particular Element Model

Sometimes it is useful to find out information about a layout cell that is associated with a particular element model. This is particularly true for PCell developers. Some questions that can be answered are the following:

1. What is the name of the layout cell associated with a given element model?
2. What version of the cell are we currently using in layouts associated with this schematic element?
3. What is the name of the module that contains this layout cell?
4. What is the exact path on my system to the module from which this layout cell was loaded?
5. Can I get a short description of this layout cell?

[Scripting How-To: How Can I Determine Which Version of a Model is Being Used and Where it is Being Loaded From?](#) we'll use the MLIN model as the starting point for answering these questions. First let's write some code to determine the name of the layout cell that is associated with the MLIN model. To do this we'll make use of the CellMappings collection on the application object. Here's an example:

```
' Get the layout cell name associated with MLIN model
Sub Main
    Debug.Clear
    Debug.Print MWOOffice.CellMappings("MLIN").CellName
End Sub
```

Running this code we get the following output:

MLIN*

So the name is the same but with an asterisk. Now we know that how to go from the Model name MLIN to the layout cell name. Next let's use this with the LayoutCells collection on the Application object to get a reference to this cell and print information about it.

```
' Get information about the MLIN model's layout cell.
Sub Main
    Dim layCell As LayoutCell
    Dim cellName As String

    cellName = MWOOffice.CellMappings("MLIN").CellName
    Set layCell = MWOOffice.LayoutCells(cellName)

    Debug.Clear
    Debug.Print "Name" = " & layCell.Name
    Debug.Print "Description" = " & layCell.Description
    Debug.Print "Version" = " & layCell.Version
    Debug.Print "Module Name" = " & layCell.ModuleName
    Debug.Print "Module Filename" = " & layCell.ModuleFilename
End Sub
```

Here we do the same thing as before and get the cell name into a string. Then we use this string as an index into the LayoutCells collection to get a LayoutCell reference in

layCell

. Next we use our layout cell reference to print information about this cell. Running the program we get the following output:

```
Name = MLIN*
Description = Default microstrip line cell
Version = 0
Module Name = Cel_defs.dll
Module Filename = H:\MWO_Dev\Debug\cells\Cel_defs.dll
```

So this is the "MLIN*" cell which is the "Default microstrip line cell", we are currently using version number 0 which is loaded from the "Cel_defs.dll" that is located at "H:\MWO_Dev\Debug\cells\Cel_defs.dll". That's a pretty good summary.

Using PowerShell we can do the same thing in the following four lines:

```
PS > $MWOffice = new-object -comobject MWOApp.MWOffice
PS > $cellName = $MWOffice.CellMappings.Item("MLIN").CellName
PS > $Cell = $MWOffice.LayoutCells.Item($cellName)
PS > $Cell | select Name,Description,Version,ModuleName,ModuleFilename
```

```
Name           : MLIN*
Description      : Default microstrip line cell
Version          : 0
ModuleName       : Cel_defs.dll
ModuleFilename   : H:\MWO_Dev\Debug\cells\Cel_defs.dll
```

PS >

Either way we can get a detailed report on the layout cell associated with a given element model in the AWR Design Environment.