HILEVEL Technology, Inc.

Q'n Apps #S1

Q: How can Symphony help me generate a list of pins and associated parameters?

Using the PinList feature

Besides being able to generate such a list, Symphony can also read from such a list. Let us demonstrate how both features can be applied.

Making a PinList

Very simple - provided you have already defined the pins. Just click the **PinList** icon on the Main sidebar. A dialog box will appear with two sections; **Input & Process List** and **Create List**. If you select "Pins & Groups" a PinList is created in a 'Notepad' file format containing not only the defined pins, but also the group definitions, **Run Setup** parameters, timing generator assignments and DC parametric test definitions. Make sure to rename this file appropriately, otherwise you may overwrite it the next time you create a PinList. If you plan to read the file in later, be sure to sort by system channel (the default) when creating the list.

Reading a PinList

Before you make such a list, it may behoove you to create your own example. To do so, simply load any SET file and create a PinList following the steps of **Making a PinList**. The result is shown on the next page. If you would like to have the SET file used to create our list, it is available online from this Q'nApp.

The specifics for Pin# and Name are actually optional; however, to satisfy the PinList syntax, you need to use '?' wherever Name and Pin# are undefined. If you later add these names and/or numbers (or for that matter make any other changes), you can again invoke **PinList**. In the dialog box, you can **Open** the file, **View** it, and **Execute** its content - which means transforming PinList information into SET file information. If you create a SET file this way, be sure to "Save As..." under the **File** drawer so that you can restore the SET file anytime you wish.

Your PinList must start with the word 'PinList' and must end with '$End'. You need to specify System Channel (i.e. tester pin), Group Number (i.e. Grp) and Type of pin (Input, Output, Bidirectional, and Split cycle i/o) - please use only a single character {I O B S}. The list should be sorted using incrementing system channel numbers for proper importing. For your own documentation purposes you can create a PinList file sorted by pin name or pin number, but be sure to save a PinList sorted by system channel in case you need to import it again sometime.

Now see the **PinList** example on the next page.

```
PinList for C:\Symphony\74F161.txt
;SysCh  Pin#  Name   Grp   Type(IOBS)
 10      7     CET    5     I
 11      6     P3     2     I
 12      5     P2     2     I
 13      4     P1     2     I
 14      3     P0     2     I
 15      2     CLK    3     I
 16      1     MR     6     I
 25      9     PE     4     I
 26      10    CET    5     I
 27      11    Q3     1     O
 28      12    Q2     1     O
 29      13    Q1     1     O
 30      14    Q0     1     O
 31      15    TC     1     O

$End
```

Example of a Pinlist file

Also see:
QnApp S16: Symphony Files
QnApp S26: Functional Test Preparation
QnApp S30: Swap Pins