New EDIF Netlist Controls

Synplicity provides you with the ability to control the formatting of EDIF netlists for use with Xilinx FPGAs.

by Margaret E. Albrecht, Technical Marketing Manager, Synplicity
maggie@synplicity.com

There are several commonly used conventions for delimiting busses in netlists such as EDIF. Mixing conventions can cause problems with your design flow, so Synplicity has added additional user control to allow easier conformance to your chosen convention.

As of Synplify 5.1 and later releases, you can specify the bus format and the case of nets in the Xilinx EDIF netlist. The two new attributes**, syn_edif_bit_format** and **syn_edif_scalar_format**, allow you to specify the format either directly in HDL code, or by applying the attribute preferences graphically in Synplify's constraints editor (SCOPE).

How to Apply the syn_edif_bit_format Attribute

Applied to the top level, and all black box modules and components in a design, **syn_edif_bit_format** can be specified for vectors and takes the following values:

- "%C<%i>" – Use <> to delimit bits of a bus.
- "%C[%i]" – Use [ ] to delimit bits of a bus.
- "%C(%i)" – Use ( ) to delimit bits of a bus.
- "%C_%i" – Use _ to delimit bits of a bus [implies **syn_noarray_ports**=1].
- "%C%i" – Append bit to the name of a bus [implies **syn_noarray_ports** = 1].

The value of C can be:

- u – To upshift the case of the base name of a bus.
- d – To downshift the case of the base name of a bus.

  n – To preserve the case of the base name of a bus.

How to Apply the **syn_edif_bit_format** Attribute in a Constraint File (.sdc) or Graphically in SCOPE:

```
# Attributes
define_global_attribute syn_edif_bit_format {%n[%i]}
```

Note that %u and %d can be substituted for %n for upshifting and downshifting the base name.

Figure 1, shows a screen shot of Synplify (upper left), HDL Analyst – RTL View (upper right), SCOPE graphical constraints editor (left center), Verilog source file (lower left), and the Synplify EDIF netlist (lower right). This screen shot shows how you can set an EDIF netlist attribute in SCOPE (left center). The side by side netlists (bottom left and right) illustrate the change in the bit format from your HDL source (bottom Left: [3:0] Y) to the EDIF netlist output (bottom right: Y<3:0>).

(Continued)

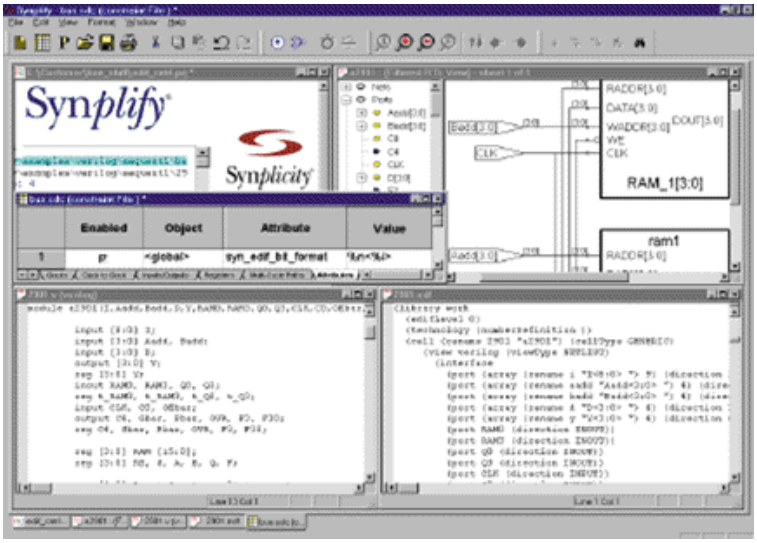

Figure 1

Applying the syn_edif_scalar Attribute

Applied to top level and black box modules and components in a design, the **syn_edif_scalar_format** can be applied on scalars and takes the following values:

- "%u" – Upshifts the case of the base name of a scalar.
- "%d" – Downshifts the case of the base name of a scalar.
- "%n" – Preserves the case of the base name of a scalar.

Verilog

Apply on a module basis:

```
module test(...) /* synthesis syn_edif_bit_format =
"%d[%i]" syn_edif_scalar_format = "%u"
*/
```

VHDL

Apply on an architecture/component basis:

```
Component
                component test
                 port( ...
        );
                end component;
                attribute syn_edif_bit_format string;
                attribute syn_edif_scalar_format :
                string;
                attribute syn_edif_bit_format of test
                component is "%u(%i)";
                attribute syn_edif_scalar_format of
                 test component is "%u";
Architecture
                architecture rtl of test is
                attribute syn_edif_bit_format string;
                attribute syn_edif_scalar_format :
                string;
                attribute syn_edif_bit_format of rtl
                architecture is "%u(%i)";
                attribute syn_edif_scalar_format of
test component is "%u";
```

Applying the **syn_edif_scalar_form**at Attribute in a Constraint File (.sdc) or Grapically in SCOPE:

```
# Attributes
define_global_attribute syn_edif_scalar_forma t {%n}
```

Note that %u and %d can be substituted for %n for upshifting and downshifting of the base name.

Attribute Examples given a bus definition, My_Bus[1:0]:

- %n<%i> should rename My_Bus to: My_Bus<1:0>.
- %u(%i) should rename My_Bus to: MY_BUS(1:0).
- %d[%i] should rename My_Bus to: my_bus[1:0].
- %n_%i should rename My_Bus to: My_Bus_1 My_Bus_0.
- %u%i should rename My_Bus to: MY_BUS1 MY_BUS0.

Summary

This article shows how to control the formatting of Synplicity-Xilinx EDIF netlist in your designs. By providing an interface to customize the Synplify netlist, you can integrate high-performance IP or legacy code with a variety of net nomenclatures into your existing Synplify design flow.

For more information about Synplify, SCOPE, and HDL Analyst, please see the Synplicity website at http://www.synplicity.com