

## EPLD

**My design requires a 3.3 V output on the I/O pins of an XC7300 device. If I tie  $V_{CCIO}$  to a 3.3 V supply, can I still drive inputs on the I/O pins at 5 V?**

Yes, when the  $V_{CCIO}$  pin is supplied by 3.3 V, only the logic output is affected. The thresholds on all input buffers are compatible with both 5 V and 3.3 V inputs, and the ESD protection diodes are connected to 5 V. There is no need for external resistors.

**I am using an ABEL source file for a design on the XC7372. I have explicitly assigned the critical signals to the fast inputs on the device. The timing report however shows that they are treating them as standard I/O. How do I enable the fast inputs?**

The default configuration for ABEL is to route the input pins through the UIM as a normal input. In order to utilize the fast inputs, you must explicitly declare them using the property statement in ABEL. For example, if

signals A, B, and C need to be fast inputs, use the following statement in your ABEL file:

```
PLUSASM PROPERTY `INPUTPIN (FI)
  A B C';
```

**I am using an XC7300 in a design with a large number of input and output signals switching. I am concerned that I may have problems with ground bounce. What can I do to minimize this possibility?**

There are several options to minimize potential ground bounce. If the system is designed for TTL levels, you may be able to connect  $V_{CCIO}$  to a 3.3 V supply voltage. This will reduce the amount of current needed to swing outputs. Also, you can slow the output slew rate from the default 1.5 V/ns to 1.0 V/ns by using the PLUSASM command,

```
FAST OFF <signal names>
```

or in ABEL,

```
PLUSASM PROPERTY `FAST OFF
  <signal names>'
```

## ABEL

**Looking at several ABEL files, I've noticed that in some cases, the OR operator appears as the # symbol and at other times as the + symbol. Is there a difference between the two?**

ABEL supports two operator sets. Using the command

```
@alternate
```

invokes the alternative operator set. The following table lists both sets.

ABEL Operator	Alternative	Description
!	/	NOT
&	*	AND
#	+	OR
\$	:+	XOR
!\$	.*	XNOR

Note that while using the alternative operators, the mathematical operators for addition, multiplication, and division are not available.

The standard ABEL operators, however, are still in effect while using the alternative set. If you wish to mix the two, you can turn the alternative operators off by using the command

```
@standard
```

**What happens if I make multiple assignments to the same identifier in ABEL?**

When ABEL finds multiple assignments to a single identifier, they are logically ORed together. If a complement operator is found on the left side of an equation, then the complement is performed after the expressions have been ORed. For example,

```
A = B;
A = !C;
!A = D;
!A = E;
```

is equivalent to

```
A = B # !C # !(D # E);
```

# Synopsys

## How do I take a Synopsys design into a Viewlogic schematic?

If you want to move a Synopsys design into a Viewlogic schematic, then the Synopsys design must be converted into a .xnf file for a Viewlogic schematic symbol. The procedure to follow is:

- 1) Compile the design in Synopsys and write out a .sxnf or .sedif file.
- 2) run `syn2xnf -s` on the .sxnf/.sedif file.  
For example, let's say you've created a design called `count8`. To bring `count8` into a Viewlogic schematic, type:  
`syn2xnf -s count8`

This will produce an .xnf file that can be used in a Viewlogic schematic.

## How do I read an .xnf file into Synopsys?

- 1) Compile the design to an .lca file.
- 2) Run `lca2xnf` on the .lca file.
- 3) Run `xnf2vss` (`xnf2vss` is only in XSI 3.2 and greater) on the .xnf file from step (2). This

will produce a file with the extension .vxnf

- 4) Rename the .vxnf file to a .xnf file.
- 5) Read this renamed .xnf file into Synopsys.  
For example, if you have a .lca file called `count.lca`, `lca2xnf` would produce a file called `count8.xnf`. Next, `xnf2vss` would be run on `count8.xnf`. A file called `count8.vxnf` would be produced. Finally, after renaming `count8.vxnf` to `count8.xnf`, now you will have a .xnf file that can be read into Synopsys.

## Why does xnf2vss create a .vxnf file and a .nrf file when it is run?

The .vxnf file is for reading a .xnf file into Synopsys (*see previous question*). The .nrf file shows how the .xnf names are changed for the Synopsys VSS simulation flow. Synopsys and XACT using different naming conventions. So, when the .xnf is brought into VSS, sometimes the symbol names and netnames in the .xnf file are modified. The .nrf file will show you the modifications.

# OrCAD

## Can OrCAD Capture for Windows be used to enter designs in XACTstep?

The OrCAD interface in XACTstep is designed for use with SDT 386+. However, a methodology has been developed for using Capture with XACTstep. The appropriate FPGA libraries must be translated into OrCAD Capture and selected during schematic entry. When adding properties or attributes to a component, they must be named using the default names from OrCAD SDT. The schematic must be saved as an .SDF file. A document detailing this methodology can be obtained using the XDOCS E-mail document retrieval system; please request document number 26121.

We currently recommend completing existing designs in SDT 386+ before migrating to OrCAD Capture. The translation process can lose attributes that are not specified as PART FIELD properties.

# Mentor Graphics

## How can I obtain the latest copy of the Xilinx Autologic libraries?

You can obtain the latest version of the libraries (v1.2 as of October 1995) from Mentor's FTP site at [supportnet.mentorg.com](http://supportnet.mentorg.com) (137.202.128.4).

These libraries are located in:

```
/pub/mentortech/tdd/libraries/  
fpga/xc2k_1.2.tar.Z  
/pub/mentortech/tdd/libraries/  
fpga/xc3k_1.2.tar.Z  
/pub/mentortech/tdd/libraries/  
fpga/xc4k_1.2.tar.Z  
/pub/mentortech/tdd/libraries/  
fpga/xc7k_1.2.tar.Z
```

These libraries are compatible with both Autologic I and Autologic II.