A primitive array is a single primitive that represents multiple identical primitives. You can use primitive arrays to create a more compact GDF or OrCAD Schematic File by entering a single primitive that the Compiler Netlist Extractor translates into multiple primitives.

You can create primitive arrays in two ways:

If all the pinstubs of the primitive are connected to buses with n members, the primitive is translated into an array of n individual primitives. Each individual node of the bus is connected to the corresponding pinstub of each individual primitive in the array. Example 1

An INPUT, INPUTC, OUTPUT, OUTPUTC, BIDIR, or BIDIRC pin primitive that is given a single-range bus name or dual-range bus name is translated into an array of primitives. You cannot use a sequential bus name to name a pin primitive.

If some of the primitive's pinstubs are connected to buses with n members and some are connected to single nodes, the primitive is translated into an array of n primitives. In this case, each individual node of the bus is connected to the corresponding pinstub of each primitive in the array, and each single node that is not part of a bus is connected to the same pinstub of each primitive. Example 2

You must name all nodes and buses that are used to create a primitive array except in the following cases, where node and bus names are optional:

A single node that is connected to a primitive array.

A bus wire that is connected to a primitive array, if at least one segment of the network that contains this bus wire is explicitly named before any connection dots or if the network is connected to a pin with a single- or dual-range bus name. Example 3

You cannot enter probe and resource assignments on primitive symbols used to create a primitive array. Refer to Guidelines for Working with Assignments for information on how to enter assignments for a primitive array.

See also:

Bus Names

Bus-to-Megafunction or Macrofunction Connections

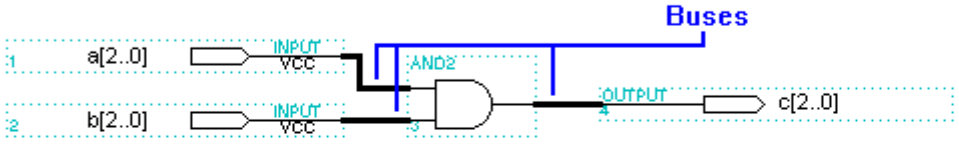
Nodes

Pin & Node Names

Example 1:

Primitive array with a primitive connected to three buses

In this example, a primitive array is created by connecting three buses, A[0..2], B[0..2], and C[0..2], to two INPUT pins, an OUTPUT pin, and an AND2 gate.



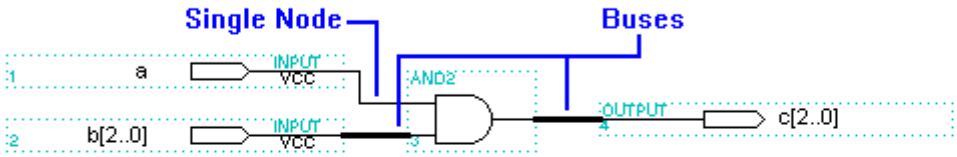
During processing, the Compiler translates this primitive array into six INPUT pins, three OUTPUT pins, and three AND2 gates, with the following connections:

- One AND2 gate is connected to nodes A0, B0, and C0.
- One AND2 gate is connected to nodes A1, B1, and C1.
- One AND2 gate is connected to nodes A2, B2, and C2.
- Input pins A0, A1, and A2 are connected to nodes A0, A1, and A2, respectively.
- Input pins B0, B1, and B2 are connected to nodes B0, B1, and B2, respectively.
- Output pins C0, C1, and C2 are connected to nodes C0, C1, and C2, respectively.

Example 2:

Primitive array with a primitive connected to two buses and a single node

In this example, a primitive array is created by connecting two buses, B[0..2] and C[0..2], and a single node (A) to two INPUT pins, an OUTPUT pin, and an AND2 gate.



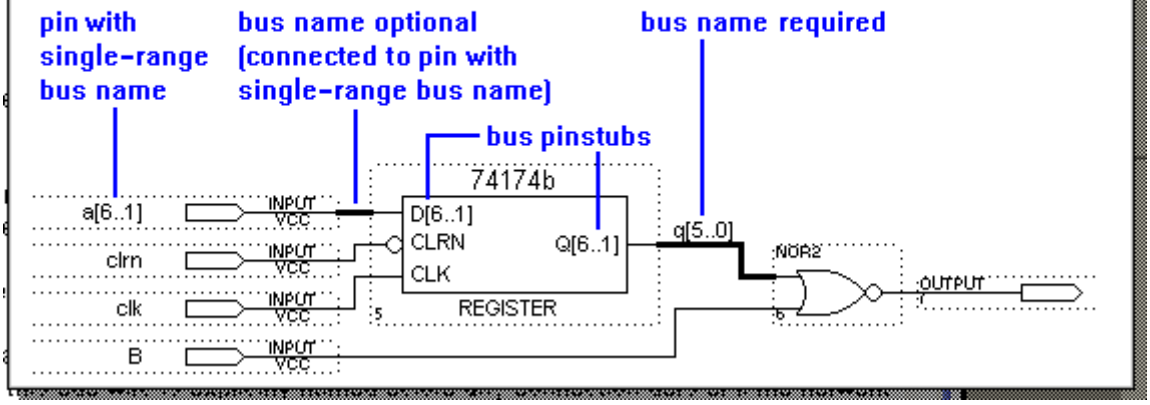
This schematic creates an array that includes six INPUT pins and six AND2 primitives.

During processing, the Compiler translates this primitive array into four INPUT pins, three OUTPUT pins, and three AND2 gates, with the following connections:

- One AND2 gate is connected to nodes A, B0, and C0.
- One AND2 gate is connected to nodes A, B1, and C1.
- One AND2 gate is connected to nodes A, B2, and C2.
- Input pin A is connected to node A.
- Input pins B0, B1, and B2 are connected to nodes B0, B1, and B2, respectively.
- Output pins C0, C1, and C2 are connected to nodes C0, C1, and C2, respectively.

Example 3

Bus name exceptions for primitive arrays



Names that identify a pin or node.

A pin name is enclosed within a pin primitive symbol; a node name is a text block that is associated with a node line (wire). Selecting a node line also selects a name that is associated with the node. You can change a comment into a node name by dragging it onto a node line and vice versa.

When you enter a pin or node name in a Graphic Editor File, you must follow these rules: It can contain up to 32 name characters. Example 4

It may not contain blank spaces. Leading or trailing spaces are ignored.

It must be unique, i.e., no two pins may have the same name in the same design file at the same hierarchy level.

Any node that is connected to a bus line must be named. Example 5

Node names that are bits of a dual-range bus must be expressed in the format <name>[<width>][<size>] or <name><width>_<size>. If you name a single node in this format, it will be interpreted as part of a dual-range bus if another single-range or dual-range bus in the file uses the same <name>.

1. If you do not name a node, it has a default hierarchical node name.
2. Assigning a single-range bus name to a pin creates an array of pins. You cannot assign dual-range or sequential bus names to pins.

When you connect named pins and nodes to each other, you must follow these rules unless you use WIRE primitives to rename nodes:

A node connected to an input pin must have the same name as the input pin.

A node that feeds more than one output pin cannot be named.

A node that has one input pin as its source and one output pin as its destination cannot be named.

The WIRE primitive can be used to rename any node. Example 6

Nodes can be logically connected by name only; they need not be physically connected. Nodes are connected when their names are identical. Example 7

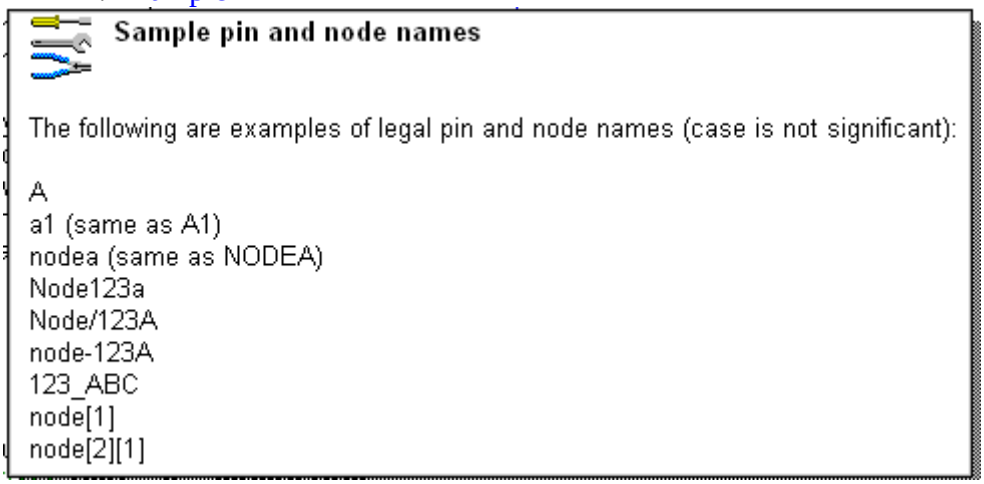
You can quickly locate a node or pin name in the current Graphic Editor file with Find Text or Find Node in Design File (Utilities menu). You can find a node or pin name in the floorplan for the project with Find Node in Floorplan (Utilities menu).

Compiler-generated names that contain the tilde (~) character may appear in the Fit File for a project. The tilde character is reserved for Compiler-generated names only; you cannot use it in your own pin, node, and group (bus) names.

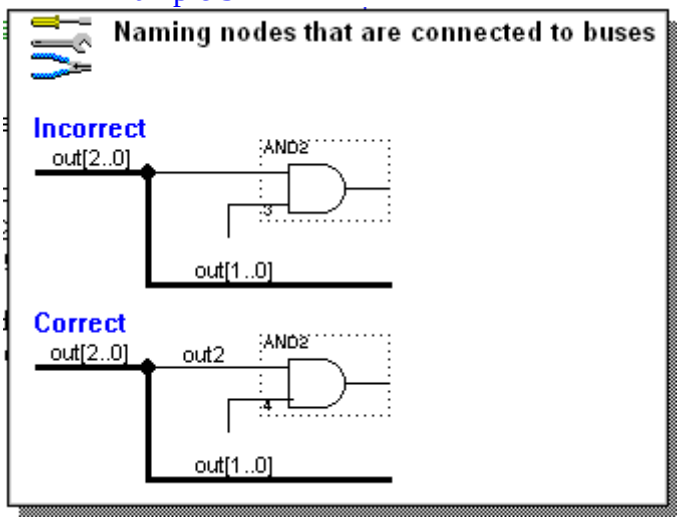
See also:

Bus Names
Changing the Pin Name or Default Value
Hierarchical Node & Symbol Names
Naming a Node or Bus

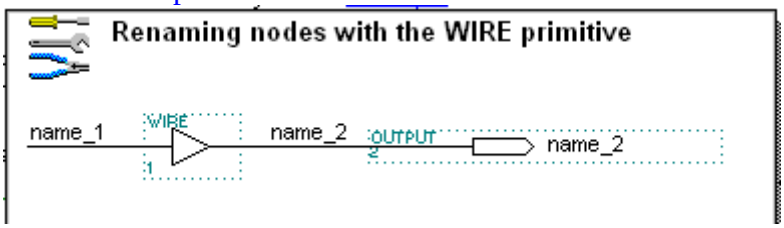
. Example 4



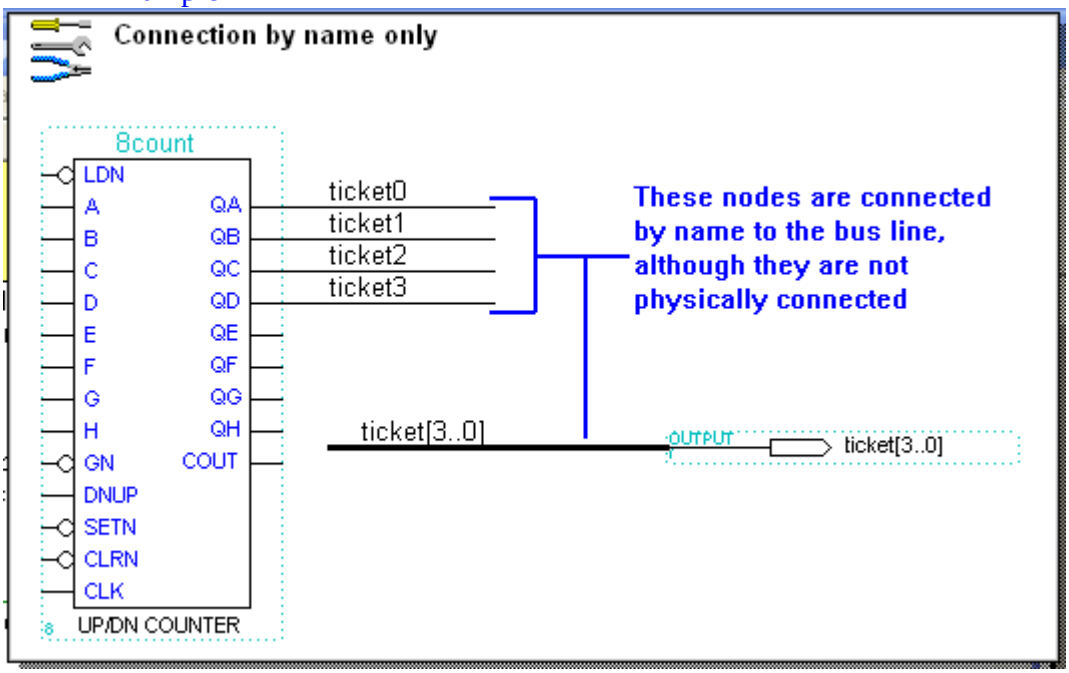
Example 5



Example 6



Example 7



4count (counter)
4-Bit Binary Up/Down Counter with Synchronous Load (LDN), Asynchronous Clear and Asynchronous Load (SETN)

Default Signal Levels: GND--A, B, C, D, CLK
VCC--LDN, CIN, DNUP, SETN, CLRN

AHDL Function Prototype (port name and order also apply to Verilog HDL):

```
FUNCTION 4count (clk, clrn, setn, ldn, cin, dnup, d, c, b, a)
RETURNS (qd, qc, qb, qa, cout);
```

Inputs										Outputs				
CLK	CLRN	SETN	LDN	CIN	DNUP	D	C	B	A	QD	QC	QB	QA	COUT (Note)
x	L	x	x	x	x					L	L	L	L	x
x	H	L	x	x	x	d	c	b	a	d	c	b	a	x
⌡⌠	H	H	L	x	x	d	c	b	a	d	c	b	a	x
⌡⌠	H	H	H	L	X					Hold				x
⌡⌠	H	H	H	H	H					Count Down				L
⌡⌠	H	H	H	H	L					Count Up				L

See also:
Counter Macrofunctions
Megafunctions/LPM
Old-Style Macrofunctions (by function)
Unused Inputs to Primitives, Megafunctions & Macrofunctions