Single Output Gate V1.0.2

October 15, 1999 Product Specification

Xilinx Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
E-mail: coregen@xilinx.com
URL: www.xilinx.com/ipcenter

Features

- Drop-in module for Virtex, Virtex-E and Spartan2 families
- Generates AND, NAND, OR, NOR, XOR and XNOR gates
- Supports 2 to 64 input gates
- Optional registered output with optional clock enable and asynchronous and synchronous controls
- Incorporates Xilinx Smart-IP technology for maximum performance
- To be used with Xilinx Core Generator V2.1i Tool and later versions

Functional Description

The single bit output gate is a member of the BaseBLOX series of building blocks for the Virtex architecture. The output is generated by performing the selected logical operation on all of its inputs. Options are provided for selecting the gate type (AND, NAND, OR, NOR, XOR or XNOR), registered outputs and non-registered outputs. When a registered output is selected options are also provided for **Clock Enable**, **Asynchronous Set and Clear**, and, **Synchronous Set and Clea**r. The module can optionally be generated as a Relationally Placed Macro (RPM) or as unplaced logic. When an RPM is generated the logic is placed in a column.

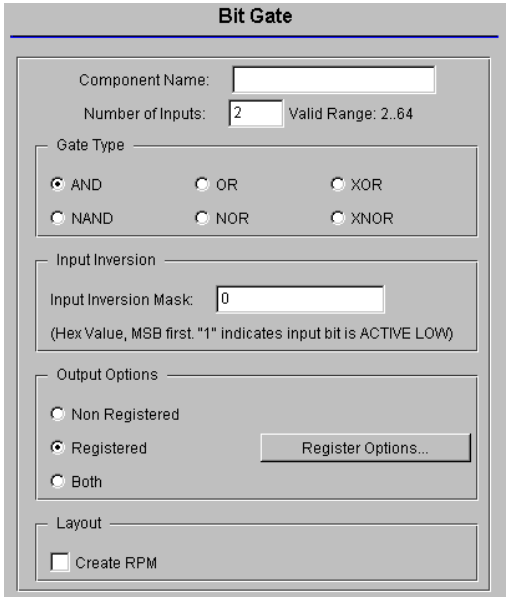

Figure 1: Main Gate Bit Parameterization Screen

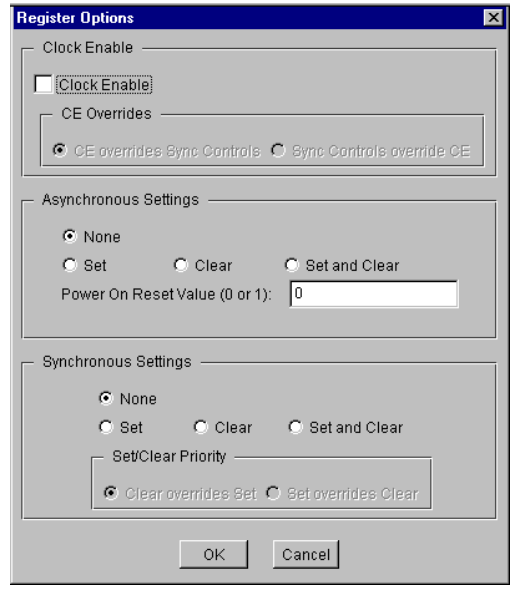

Figure 2: Gate Bit Register Options Parameterization Screen

Pinout

Signal names are shown in Figure 3 and described in Table 1.

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
I[N:0]	Input	Gate inputs
O	Output	Output for non-registered module
D	Internal	Internal Data Input connection to optional Output Register
CE	Input	Clock Enable
CLK	Input	Clock-rising edge
ASET	Input	Asynchronous Set: forces the registered output to a High state when driven
ACLR	Input	Asynchronous Clear: forces outputs to a Low state when driven
SSET	Input	Synchronous Set: forces the registered output to a High state on next concurrent clock edge
SCLR	Input	Synchronous Clear: forces the registered output to a Low state on next concurrent clock edge
Q	Output	Output for registered module

Note:
All control inputs are Active High. Should an Active Low input be required for a particular control pin an inverter must be in the path to the pin. The inverter will be absorbed appropriately by the core logic during mapping.

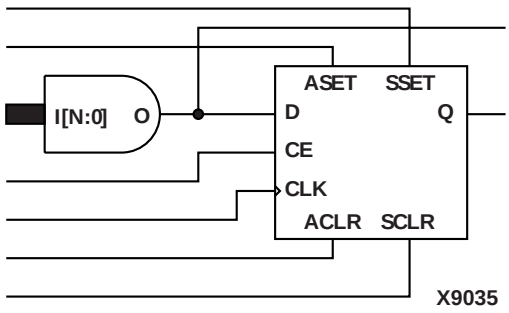

Figure 3: Core Schematic Symbol

CORE Generator Parameters

- The main CORE Generator parameterization screen for this module is shown in Figure 2. The parameters are as follows:
- **Component Name**: **The** component name is used as the base name of the output files generated for this module. Names must begin with a letter and must be composed from the following characters: a to z, 0 to 9 and "_".
- **Number of Inputs**: Enter the number of gate inputs. The valid range is 2 to 64. The default value is 2.
- **Gate Type**: Select the appropriate radio button for the types of gate function required. The default setting is **AND**.
- **Input Inversion Mask**: This parameter defines which input bits are inverted prior to being operated upon by the selected logical function. A "1" in the mask pattern indicates that the corresponding bit in the input is to be inverted. The text box accepts a hex value which must be equal to or less than the **Number of Inputs**. If a value is entered that is fewer bits than the input width of the gate it is padded with zeros. An invalid value is highlighted in red in the text box. The default value is 0.
- **Output Options**: Select the appropriate radio button for the types of outputs required. The default setting is **Registered**.
- **Register Options**: This button is only enabled when a registered output has been requested via the **Output Options**. Clicking on this button brings up the Register Options parameterization screen (see Figure 2).
- **Create RPM**: When this box is checked the module will be generated with relative location attributes attached. The resulting placement of the module will be in a column with two bits per slice. The default setting is to create an RPM.

The Register Options parameterization screen for this module is shown in Figure 2. The parameters are as follows:

- **Clock Enable**: When this box is checked the module is generated with a clock enable input. The default setting is unchecked.
- **CE Overrides**: This parameter controls whether the SSET and SCLR inputs are qualified by CE or not. This parameter is only enabled when a **Clock Enable** input has been requested.
  When **CE Overrides Sync Controls** is selected an Active level on any of the synchronous control inputs will only be acted upon when the CE pin is Active. Note that this is not the way that the dedicated inputs on the flip-flop primitive work, and so setting the **CE Overrides** parameter to **CE Overrides Sync Control**s will force the synchronous control functionality to be implemented using logic in the Look Up Table (LUT) preceding the flip-flop. This results in increased resource utilization even when asynchronous controls are not present.

When **Sync Controls Override CE** is selected an Active level on any of the synchronous control inputs will be acted upon irrespective of the state of the CE pin. This setting is more efficient when asynchronous inputs are not present because it allows the dedicated inputs on the flip-flop primitive to be used. It is less efficient when the presence of asynchronous inputs force the synchronous control functionality to be implemented using logic in the LUT preceding the flip-flop. This is because the CE signal has to be gated with the synchronous control inputs so that they can all generate a CE signal to the flip-flop, slowing down the CE path and resulting in slower overall operation of the module.
The default setting is **Sync Controls Override CE** so that the more efficient implementation can be generated.

- **Asynchronous Settings**: All asynchronous controls are implemented using the dedicated inputs on the flip-flop primitive. The module can be generated with the following asynchronous control inputs by clicking on the appropriate button:
  - **None**: No asynchronous control inputs. This is the default setting.
  - **Set**: An ASET input pin is generated.
  - **Clear**: An ACLR input pin is generated.
  - **Set and Clear**: Both ASET and ACLR input pins are generated. ACLR has priority over ASET when both are asserted at the same time.
- **Power On Reset Value**: This text box accepts a value of 0 or 1 and defines the power on value for the output flip-flop. The default value is 0.
- **Synchronous Settings**: When no asynchronous controls are implemented (i.e. the **Asynchronous Setting** is **None**) the synchronous controls can be implemented using the dedicated inputs on the flip-flop primitive. There are exceptions to this, see the description of the **Set/Clear Priority** and **CE Overrides** parameters.
  When asynchronous controls are present any synchronous control functionality must be implemented using logic in the Look Up Tables (LUTs) preceding the flip-flop. In the case where a non-registered output is not present, this logic can (in some cases) be absorbed into the same LUTs used to implement the gate function. In cases where this is not possible the synchronous control logic will consume an additional LUT.
  The module can be generated with the following synchronous control inputs by clicking on the appropriate button:
  - **None**: No synchronous control inputs. This is the default setting.
  - **Set**: An SSET input pin is generated.
  - **Clear**: An SCLR input pin is generated.
  - **Set and Clear**: Both SSET and SCLR input pins are generated. SCLR/SSET priority is defined by the setting of the Set/Clear Priority parameter.
- **Set/Clear Priority**: By selecting the appropriate radio button the priority of synchronous clear to synchronous set can be controlled. This parameter is only enabled when both synchronous set and synchronous clear have been requested.
  It is not possible for **Set** to override **Clear** when the synchronous control functionality is implemented using the dedicated inputs on the flip-flop primitive. This can only be implemented using logic in the LUT(s) preceding the flip-flop.
  The default setting is **Clear Overrides Set** so that a more efficient implementation can be generated.

Parameter Values in the XCO File

Names of XCO file parameters and parameter values are identical to names and values shown in the GUI, except that underscore characters (_) are used instead of spaces. The text in an XCO file is case insensitive.

Table 2 shows the XCO file parameters and values, and summarizes the GUI defaults. The following is an example of the CSET parameters in an XCO file:

```
CSET synchronous_settings = none
CSET component_name = c_gate_bit
CSET number_of_inputs = 2
CSET output_options = registered
CSET ce_overrides = sync_controls_override_ce
CSET gate_type = and
CSET power_on_reset_value = 0
CSET set_clear_priority = clear_overrides_set
CSET clock_enable = FALSE
CSET input_inversion_mask = 0
CSET asynchronous_settings = none
CSET create_rpm = FALSE
```

Core Resource Utilization

The LUT resource utilization figures for this core are shown in Table 3. When registered outputs are required a single flip-flop is used.

When the synchronous control functionality cannot be implemented using the dedicated control inputs of the flip-flop (i.e. when asynchronous controls are also requested) and the **CE Overrides** are set to **Sync Controls Override CE** an additional LUT per module is required.

Ordering Information

This core is downloadable free of charge from the Xilinx IP Center (www.xilinx.com/ipcenter), for use with the Xilinx Core Generator System version 2.1i and later. The Core Generator System 2.1i tool is bundled with the Alliance 2.1i and Foundation 2.1i implementation tools.

To order Xilinx software contact your local Xilinx sales representative at www.xilinx.com/company/sales.htm.

Table 2: Default Values and XCO File Values

Parameter	XCO File values	Default GUI Setting
component_name	ASCII text starting with a letter and based upon the following character set: a .. z, 0..9 and _	blank
number_of_inputs	Integer in the range 2 to 64	2
gate_type	One of the following keywords: and, nand, or, nor, xor, xnor	and
input_inversion_mask	Hex value whose value does not exceed $2^{\text{INPUT BUS WIDTH}} - 1$	0
output_options	One of the following keywords: non_registered, registered, both	registered
create_rpm	One of the following keywords: true, false	false
clock_enable	One of the following keywords: true, false	false
ce_overrides	One of the following keywords: sync_controls_override_ce, ce_overrides_sync_controls	sync_controls_override_ce
asynchronous_settings	One of the following keywords: none, set, clear, set_and_clear	none
power_on_reset_value	0 or 1	0
synchronous_settings	One of the following keywords: none, set, clear, set_and_clear	none
set_clear_priority	One of the following keywords: clear_overrides_set or set_overrides_clear	clear_overrides_set

Table 3: LUT Count by Number of Inputs for Different Output Options

Number of Inputs	Output Options					
	Non-registered	Registered			Registered and Non-Registered	
		No sync. controls implemented in logic	Sync. Control implemented in logic		No sync. controls implemented in logic	Sync. Control implemented in logic
			1 control input	2 control inputs		
2	1	1	1	1	1	2
3	1	1	1	2	1	2
4	1	1	2	2	1	2
5	2	2	2	2	2	3
6	2	2	2	3	2	3
7	2	2	3	3	2	3
8	3	3	3	3	3	4
9	3	3	3	4	3	4
10	3	3	4	4	3	4
11	4	4	4	4	4	5
12	4	4	4	5	4	5
13	4	4	5	5	4	5
14	5	5	5	5	5	6
15	5	5	5	6	5	6
16	5	5	6	6	5	6
17	6	6	6	6	6	7
18	6	6	6	7	6	7
19	6	6	7	7	6	7
20	7	7	7	7	7	8
21	7	7	7	8	7	8
22	7	7	8	8	7	8
23	8	8	8	8	8	9
24	8	8	8	9	8	9
25	8	8	9	9	8	9
26	9	9	9	9	9	10
27	9	9	9	10	9	10
28	9	9	10	10	9	10
29	10	10	10	10	10	11
30	10	10	10	11	10	11
31	10	10	11	11	10	11
32	11	11	11	11	11	12
33	11	11	11	12	11	12
34	11	11	12	12	11	12
35	12	12	12	12	12	13
36	12	12	12	13	12	13
37	12	12	13	13	12	13
38	13	13	13	13	13	14
39	13	13	13	14	13	14
40	13	13	14	14	13	14
41	14	14	14	14	14	15

Table 3: (cont.) LUT Count by Number of Inputs for different Output Options

Number of Inputs	Output Options					
	Non-registered	Registered			Registered and Non-Registered	
		No sync. controls implemented in logic	Sync. Control implemented in logic		No sync. controls implemented in logic	Sync. Control implemented in logic
			1 control input	2 control inputs		
42	14	14	14	15	14	15
43	14	14	15	15	14	15
44	15	15	15	15	15	16
45	15	15	15	16	15	16
46	15	15	16	16	15	16
47	16	16	16	16	16	17
48	16	16	16	17	16	17
49	16	16	17	17	16	17
50	17	17	17	17	17	18
51	17	17	17	18	17	18
52	17	17	18	18	17	18
53	18	18	18	18	18	19
54	18	18	18	19	18	19
55	18	18	19	19	18	19
56	19	19	19	19	19	20
57	19	19	19	20	19	20
58	19	19	20	20	19	20
59	20	20	20	20	20	21
60	20	20	20	21	20	21
61	20	20	21	21	20	21
62	21	21	21	21	21	22
63	21	21	21	22	21	22
64	21	21	22	22	21	22