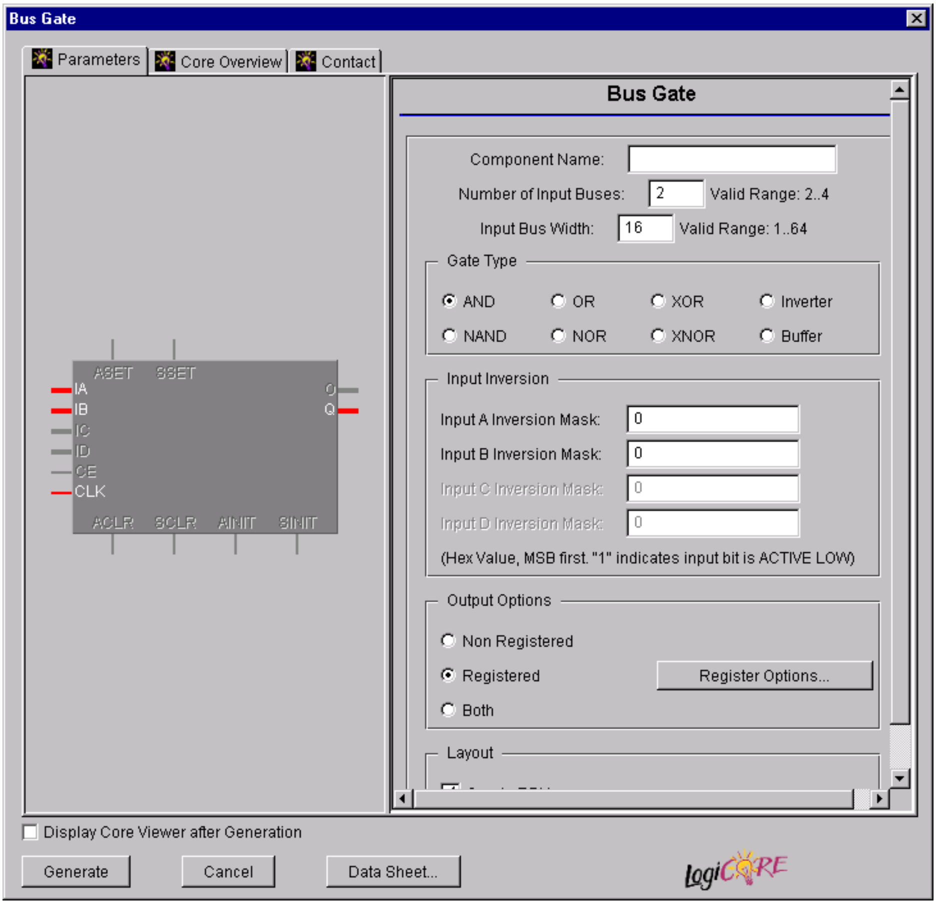
Bus Gate V3.0

November 3, 2000 Product Specification

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

Features

- Drop-in module for Virtex, Virtex™-E, Virtex™-II and Spartan™–II FPGAs
- Generates AND, NAND, OR, NOR, XOR, XNOR, Buffer and Inverter gates
- Supports 2 to 4 input buses
- Supports buses of up to 64 bits wide
- Optional registered output with optional clock enable and asynchronous and synchronous controls
- Incorporates Xilinx Smart-IP technology for maximum performance



Bus Gate

Component Name:

Number of Input Buses: 2 Valid Range: 2..4

Input Bus Width: 16 Valid Range: 1..64

Gate Type

● AND ○ OR ○ XOR ○ Inverter
○ NAND ○ NOR ○ XNOR ○ Buffer

Input Inversion

Input A Inversion Mask: 0
Input B Inversion Mask: 0
Input C Inversion Mask: 0
Input D Inversion Mask: 0
(Hex Value, MSB first. "1" indicates input bit is ACTIVE LOW)

Output Options

○ Non Registered
● Registered
○ Both

Layout

☐ Display Core Viewer after Generation

Figure 1: Main Bus Gate Parametrization Screen

- To be used with version 3.1i and later of the Xilinx CORE Generator System

Functional Description

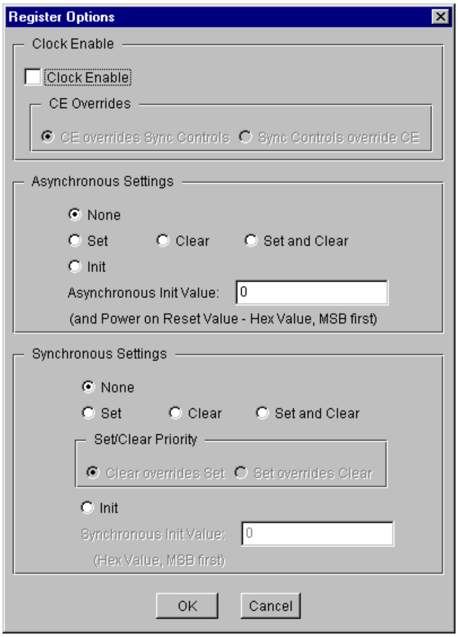
In the case of the logical operators AND, NAND, OR, NOR, XOR or XNOR, the bus gate output is generated by performing the selected logical operation bitwise on the input buses (i.e. bit 0 of all input buses are gated to generate bit 0 of the output, etc.). In the case of the Inverter and Buffer operations only one input bus (IA[N:0]) is used and operated upon appropriately. Options are provided for selecting the gate type, registered outputs and non-registered outputs. When a registered output is selected options are also provided for **Clock Enable**, **Asynchronous Set**, **Clear**, and **Init**, and **Synchronous Set**, **Clear** and **Init**. 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.

Pinout

Signal names for the schematic symbol are shown in Figure 3 and described in Table 1.

CORE Generator Parameters

The main CORE Generator parameterization screen for this module is shown in Figure 1. The parameters are as follows:



Register Options

Clock Enable

□ Clock Enable

CE Overrides

● CE overrides Sync Controls ○ Sync Controls override CE

Asynchronous Settings

● None

○ Set ○ Clear ○ Set and Clear

○ Init

Asynchronous Init Value: 0

(and Power on Reset Value - Hex Value, MSB first)

Synchronous Settings

● None

○ Set ○ Clear ○ Set and Clear

Set/Clear Priority

● Clear overrides Set ○ Set overrides Clear

○ Init

Synchronous Init Value: 0

(Hex Value, MSB first)

Figure 2: Bus Gate Register Options Parametrization Screen

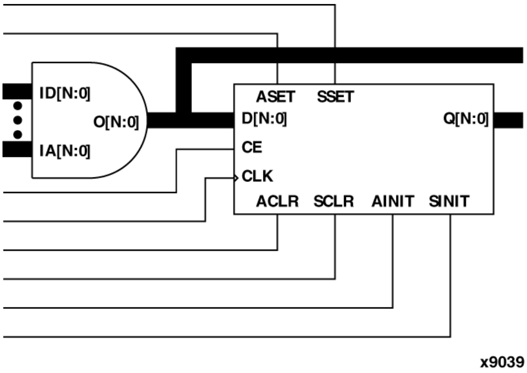


Figure 3: Core Schematic Symbol

- **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 Input Buses**: Enter the number of input buses. The valid range is 2 to 4. The default value is 2.
- **Input Bus Width**: Enter the width of the input buses (and hence the width of the output bus). The valid range is 1 to 64. The default value is 16.
- **Gate Type**: Select the appropriate radio button for the types of gate function required. The default setting is **AND**.
- **Input Inversion Masks**: For each input bus the corresponding input inversion mask defines which input bits are inverted prior to be 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 **Input Bus Width**. If a value is entered that is fewer bits than the input bus 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 is generated with relative location attributes attached. The resulting placement of the module is in a column with two bits per slice. The default operation is to create an RPM.
  Note that when a module is created as an RPM it is possible that one or more of the module dimensions may exceed those of the device being targeted. If this is the case mapping errors will occur and the compilation process will fail. In this case the module should be re-

generated with the **Create RPM** checkbox unchecked.

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 or not the SSET, SCLR, and SINIT inputs are qualified by CE. 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 also Active. Note that this is not the way that the dedicated inputs on the flip-flop primitives work, and so setting the **CE Overrides** parameter to **CE Overrides Sync Controls** will force any synchronous control functionality to be implemented using logic in the Look Up Tables (LUTs) preceding the output register. This results in increased resource utilization.

  When **Sync Controls Override CE** is selected an active level on any of the synchronous control inputs is acted upon irrespective of the state of the CE pin. This setting allows the dedicated inputs on the flip-flop primitives to be used for the synchronous control functions provided that asynchronous controls are not requested. If both asynchronous and synchronous controls are requested, the synchronous control functionality must be implemented using logic in the LUTs preceding the output register. In this case, the CE input has to be gated with the synchronous control inputs so that each synchronous control input and the CE input can generate a CE signal to the flip-flops. This results in a performance degradation for the module due to the additional gating in the CE path.

  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 primitives. 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 control pin is generated.
  - **Clear**: An ACLR control pin is generated.
  - **Set and Clear**: Both ASET and ACLR control pins are generated. ACLR has priority over ASET when both are asserted at the same time.
  - **Init**: An AINIT control pin is generated which, when asserted, will asynchronously set the output register to the value defined in the **Asynchronous Init Value** text box.
- **Asynchronous Init Value**: This text box accepts a hex value whose equivalent bit width must be less than or equal to the **Input Bus Width**. If a value is entered that has fewer bits than the **Input Bus Width** it is padded with zeros. An invalid value is highlighted in red in the text box.
  The default value is 0.
- **Synchronous Settings**: When no asynchronous controls are requested (i.e. the **Asynchronous Setting** is **None**) the synchronous controls can be implemented using the dedicated inputs on the flip-flop primitives. There are exceptions to this which are described in the sections for 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 output register. With modules where a non-registered output is not required there are combinations of parameters that allow this logic to be absorbed into the same LUTs used to implement the function. In cases where this absorption is not possible the synchronous control logic will require an additional LUT per output bit.
  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 control pin is generated.
  - **Clear**: An SCLR control pin is generated.
  - **Set and Clear**: Both SSET and SCLR control pins are generated. SCLR/SSET priority is defined by the setting of the **Set/Clear Priority** parameter.
  - **Init**: An SINIT control pin is generated which, when asserted, will synchronously set the output register to the value defined in the **Synchronous Init Value** text box.
- **Set/Clear Priority**: By selecting the appropriate radio button the relative priority of SCLR and SSET can be controlled. This parameter is only enabled when **Set and Clear** is selected for **Synchronous Settings**.

  A setting of **Clear Overrides Set** corresponds to the native operation of the flip-flop primitive. This setting will result in a more efficient implementation when asynchronous controls are not requested. A setting of **Set Overrides Clear** can only be implemented using logic in the LUTs preceding the output register.

  The default setting is **Clear Overrides Set** so that the dedicated inputs on the flip-flops can be used if available.
- **Synchronous Init Value**: This text box accepts a hex value whose equivalent bit width must be less than or equal to the **Input Bus Width**. If a value is entered that has fewer bits than the **Input Bus Width** it is padded with zeros. An invalid value is highlighted in red in the

text box. This parameter is only enabled when the **Synchronous Settings** parameter is set to **Init**. The default value is 0.

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
IA[N:0] .. ID[N:0]	Input	Input buses
O[N:0]	Output	Output for non-registered module
D[N:0]	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
AINIT	Input	Asynchronous Initialize: forces the registered outputs to a user defined state when driven
SINIT	Input	Synchronous Initialize: forces the registered outputs to a user defined state on next concurrent clock edge
Q[N:0]	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 placed in the path to the pin. The inverter will be absorbed appropriately during mapping.

Power On Conditions

See the FD-based Register datasheet for information on the power up values for registered modules.

Parameter Values in the XCO File

Names of XCO file parameters and their parameter values are identical to the 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 component_name = abc123
CSET number_of_input_buses = 2
CSET input_bus_width = 16
CSET gate_type = and
CSET input_a_inversion_mask = 0000
CSET input_b_inversion_mask = 0000
CSET input_c_inversion_mask = 0000
CSET input_d_inversion_mask = 0000
CSET output_options = registered
CSET create_rpm = TRUE
CSET clock_enable = FALSE
CSET ce_overrides = sync_controls_override_ce
CSET asynchronous_settings = none
CSET async_init_value = 0000
CSET synchronous_settings = none
CSET sync_init_value = 0000
CSET set_clear_priority = clear_overrides_set
```

Core Resource Utilization

For an accurate measure of the usage of primitives, slices, and CLBs for a particular point solution, check the **Display Core Viewer after Generation** checkbox, in CoreGen.

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 3.1i and later. The Core Generator System 3.1i tool is bundled with the Alliance 3.1i and Foundation 3.1i 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_input_buses	"1" for gate Type settings of inverter of buffer. For all other Gate Type settings the value is an Integer in the range of 2 to 4	2
input_bus_width	Integer in the range 1 to 64	16
gate_type	One of the following keywords: and, nand, or, nor, xor, xnor, inverter or buffer	and
input_a_inversion_mask	Hex value whose value does not exceed $2^{\text{input\_bus\_width}} - 1$	0
input_b_inversion_mask	Hex value whose value does not exceed $2^{\text{input\_bus\_width}} - 1$	0
input_c_inversion_mask	Hex value whose value does not exceed $2^{\text{input\_bus\_width}} - 1$	0
input_d_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	true
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, init	none
async_init_value	Hex value whose value does not exceed $2^{\text{input\_bus\_width}} - 1$	0
synchronous_settings	One of the following keywords: none, set, clear, set_and_clear, init	none
sync_init_value	Hex value whose value does not exceed $2^{\text{input\_bus\_width}} - 1$	0
set_clear_priority	One of the following keywords: clear_overrides_set or set_overrides_clear	clear_overrides_set