Twos Complementer V2.0

June 30, 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/support/techsup/tappinfo
www.xilinx.com/ipcenter

Features

- Drop-in module for Virtex™-II, Virtex, Virtex™-E, and Spartan™-II FPGAs
- Supports inputs up to 64 bits wide
- Optional registered output with optional clock enable and asynchronous and synchronous controls
- Optional Bypass (Load) capability
- Incorporates Xilinx Smart-IP technology for maximum performance
- To be used with version 3.1i or later of the Xilinx CORE Generator

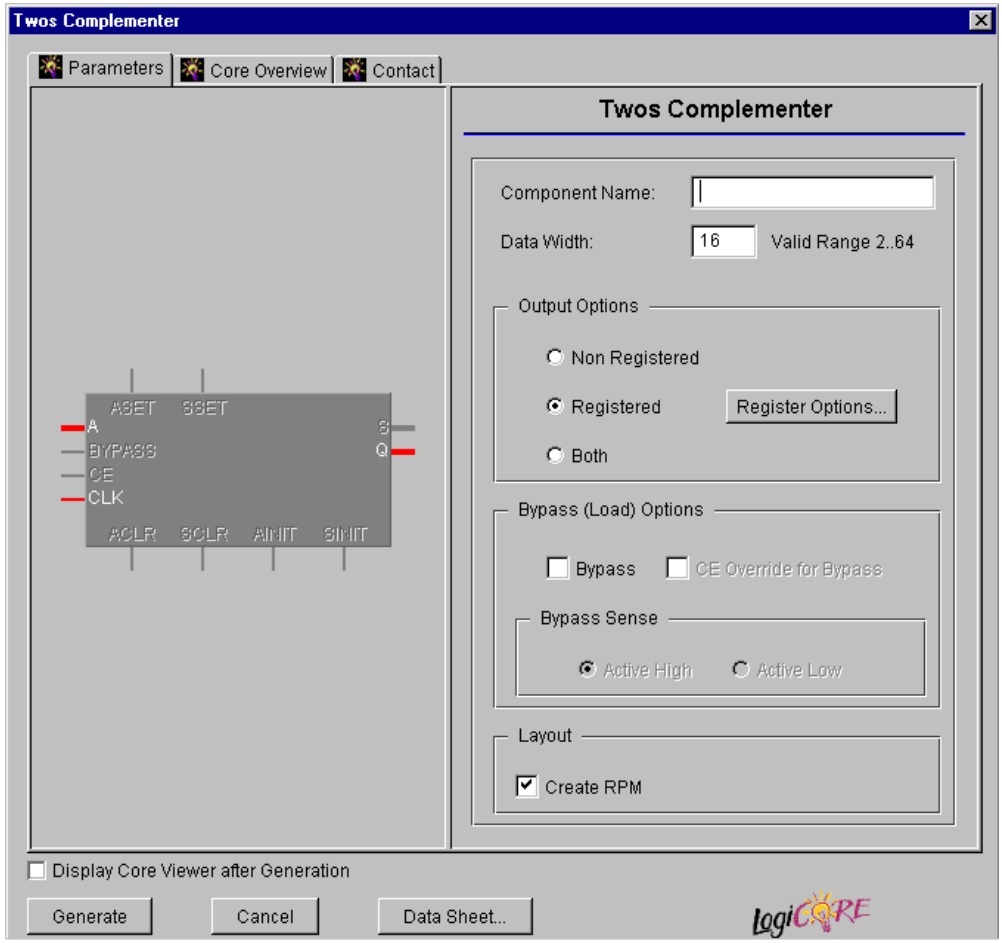

Figure 1: Main Twos Complementer Parameterization Screen

Functional Description

The Twos Complementer module converts the signed value presented on its input bus into the signed complement on its output bus. Outputs can be registered or non-registered. When a registered output is selected options are also provided for **Clock Enable**, **Asynchronous Set**, **Clear**, and **Init**, and **Synchronous Set**, **Clear** and **Init**. An optional **Bypass** capability is also provided which can load the value on the input port directly into the output register. 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:

- **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 "_".

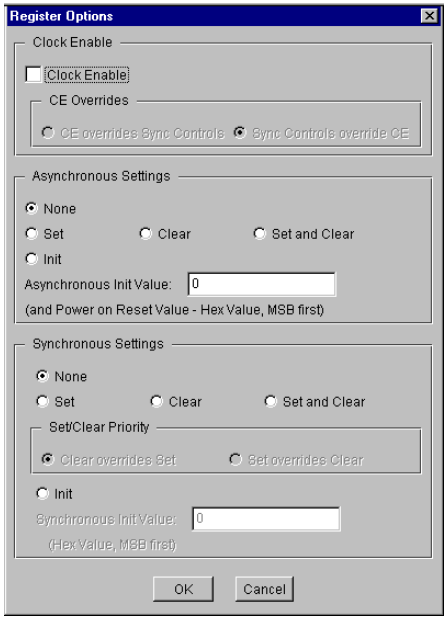

Figure 2: Twos Complementer Register Options Parameterization Screen

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
A[N:0]	Input	Input bus
BYPASS	Input	Bypass Control Signal
CE	Input	Clock Enable
CLK	Input	Clock: rising edge clock signal
ASET	Input	Asynchronous Set: forces 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 registered output to a high state on next concurrent clock edge
SCLR	Input	Synchronous Clear: forces registered output to a low state on next concurrent clock edge
AINIT	Input	Asynchronous Initialize: forces registered outputs to user defined state when driven
SINIT	Input	Synchronous Initialize: forces registered outputs to user defined state on next concurrent clock edge
S[P:0]	Output	Asynchronous output
Q[P:0]	Output	Registered output

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.
P=N+1

- **Data Width**: Enter the width of the data to be operated upon. The valid range is 2 to 64. The default value is 16.
- **Output Options**: Select the appropriate radio button for the types of outputs required. The output options settings selected here apply to all outputs. 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).
- **Bypass**: Activating the BYPASS pin allows the value on the input to pass through the logic and be loaded into the output register on the next active clock edge. This check box is only available on a registered module. The default is for no BYPASS pin to be generated.

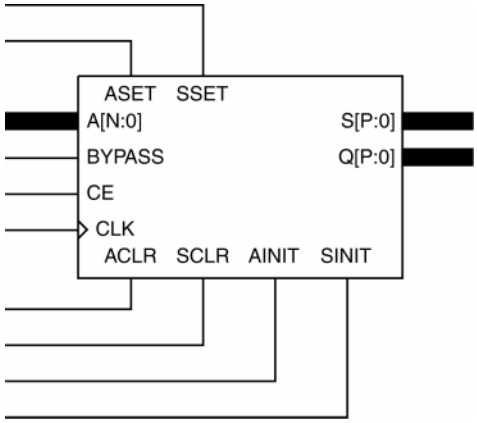

Figure 3: Core Schematic Symbol

- **CE Override for Bypass**: This parameter controls whether or not the BYPASS input is qualified by CE. When this box is checked the activation of the BYPASS signal will also enable the register. When this box is unchecked the register needs to have CE active in order to load the Port A data. By default this check box is unchecked.
- **Bypass Sense**: BYPASS is the only pin that has a parameter to control its active sense. This is because selection of an **Active Low** bypass results in a significant area saving for the module. By default this parameter is set to **Active High** so that it conforms with the active sense of all other control signals.
- **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. This will reduce the performance of the module since the placement will no longer be controlled.

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 **Data Width**. If a value is entered that has fewer bits than the **Data Width** it is padded with zeros. An invalid value is highlighted in red in the text box. The value specified in this text box also functions as the power on reset value for the output register. 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 **Data Width**. If a value is entered that has fewer bits than the **Data 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.

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 3 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 data_width = 16
CSET output_options = registered
CSET bypass = FALSE
CSET ce_override_for_bypass = TRUE
CSET bypass_sense = active_high
CSET create_rpm = TRUE
CSET clock_enable = FALSE
CSET ce_overrides = sync_controls_override_ce
CSET asynchronous_settings = none
CSET async_init_value = 0
CSET synchronous_settings = none
CSET sync_init_value = 0
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 version 3.1i or later of the Xilinx CORE Generator System. The CORE Generator System is bundled with the Alliance and Foundation implementation tools.

To order Xilinx software contact your local Xilinx sales representative. For information on the Xilinx sales office nearest you, please refer to:
http://www.xilinx.com/company/sales.htm

Table 2: XCO File Values and Default 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
data_width	Integer in the range 1 to 64	16
output_options	One of the following keywords: non_registered, registered, both	registered
bypass	One of the following keywords: true, false	false
ce_override_for_bypass	One of the following keywords: true, false	false
bypass_sense	One of the following keywords: active_high, active_low	active_high
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 $2^{data\_width} - 1$	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 $2^{data\_width} - 1$	0
set_clear_priority	One of the following keywords: clear_overrides_set, set_overrides_clear	clear_overrides_set