RAM-based Shift Register V1.0.3

December 17, 1999 | 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/appinfo
www.xilinx.com/ipcenter

Features

- Drop-in module for Virtex, Virtex™-E, and Spartan™-II FPGAs
- Generates fast, compact FIFO-style shift registers, delay lines or time-skew buffers using the SRL16 mode of the Select RAM
- User options to create Fixed-length or Variable-length shift registers
- Optional output register capability with clock enable and asynchronous and synchronous controls
- Incorporates Xilinx Smart-IP technology for maximum performance
- To be used with version 2.1i or later of the Xilinx CORE Generator System

Functional Description

The RAM-based Shift Register module provides a very efficient multi-bit wide shift register for use in FIFO-like applications or as a delay line or time skew buffer. Fixed length shift registers and variable length shift registers can be created. An option is also provided to register the outputs of the module. If output registering is selected there are additional options for **Clock Enable**, **Asynchronous Set**, **Clear**, and **Init**, and **Synchronous Set**, **Clear** and **Init** of the output register. The module can optionally be generated as a Relationally Placed Macro (RPM) or as unplaced logic.

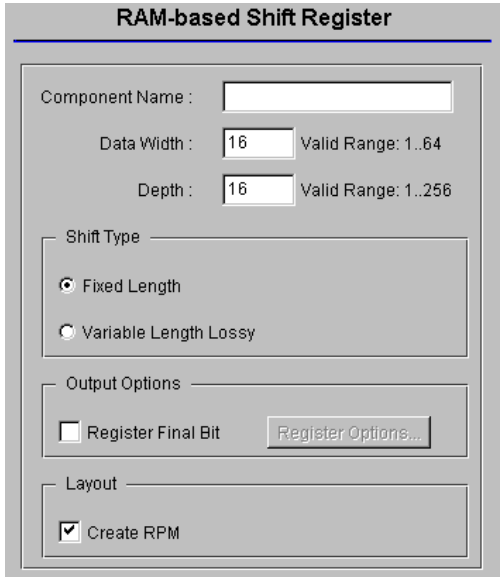

Figure 1: Main RAM-based Shift Register

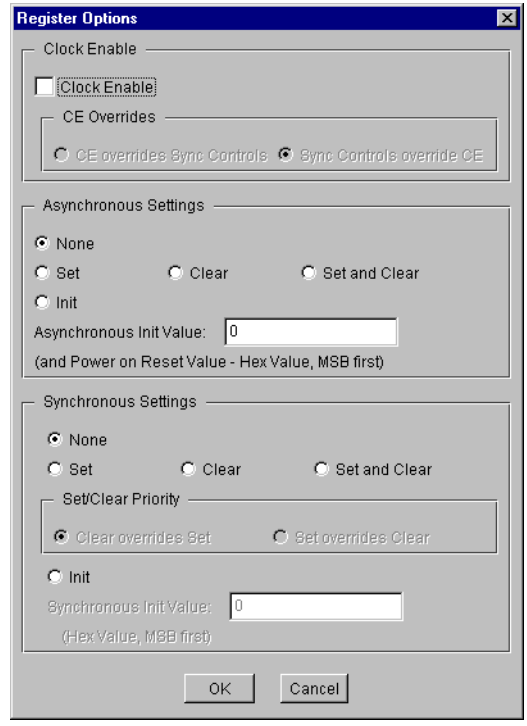

Figure 2: RAM-based Shift Register Options Parameterization Screen

Table 1: Core Signal Pinout

Signal	Signal Direction	Description
D[N:0]	Input	Parallel Data Input
A[M:0]	Input	Address Input (Only present on Variable Length modules)
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
Q[N:0]	Output	Parallel Data 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.

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 "_".

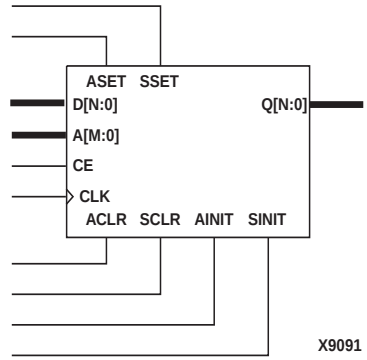

Figure 3: Core Schematic Symbol

- **Data Width**: Enter the width of the input to the shift register. The valid range is 1 to 64. The default value is 16.
- **Depth**: Enter the depth of the shift register. The valid range is 1 to 256. The default value is 16.
- **Shift Type**: Select the appropriate radio button to specify the required type of shift register. The default selection is **Fixed Length**.
  - **Fixed Length**: Parallel data is clocked into the shift register and appears at the output bus **Depth** clock cycles later.
  - **Variable Length Lossy**: The delay (in number of clocks) that it takes for data to be cycled through from input bus to output bus is defined by the value on the A[M:0] (Address) input bus. This module is referred to a "lossy" because when the Address is changed the output cannot be guaranteed to be correct for C clock cycles where C is the new value for the address.
- **Output Options**:.
  - **Register Final Bit**: Only available for **Fixed Length** modules. This checkbox defines the presence of the FD-based registers on the output of the module. This effectively adds one extra clock to the shift delay through the module. Register Options are not enabled unless this option is checked. The default setting is unchecked.

Register Options: Clicking on this button brings up the Register Options parameterization screen (see Figure 2). This button is only enabled for **Fixed Length** modules.

- **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 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 data_width = 16
CSET depth = 16
CSET shift_type = Fixed_Length
CSET register_final_bit = FALSE
CSET create_rpm = TRUE
CSET clock_enable = FALSE
CSET ce_overrides = sync_controls_override_ce
CSET asynchronous_settings = none
CSET asynchronous_init_value = 0000
CSET synchronous_settings = none
CSET synchronous_init_value = 0000
CSET set_clear_priority = clear_overrides_set
```

Core Resource Utilization

Resource utilization figures per bit of **Data Width** are shown for different **Shift Type** settings in Tables 3, 4 and 5.

Additional resources are required for **Fixed Length** modules with **Register Final Bit** set to True in some cases. In the case where synchronous control functionality has been requested which cannot be implemented using the direct control inputs of the flip-flop (i.e., when asynchronous controls are also requested), an additional LUT is used per **Data Width** bit. 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 version 2.1i 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: File Values and Default Settings

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
depth	Integer in the range 1 to 256	16
shift_type	One of the following keywords: fixed_length, variable_length_lossy	fixed_length
register_final_bit	One of the following keywords: true, false	false
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 or set_overrides_clear	clear_overrides_set

Table 3: Resource Utilization by Depth for fixed length modules with Register Final Bit = False (LUT/ FF)

Number of Inputs	Resources Used
1-16	1/0
17	2/0
18-33	2/1
34	3/2
35-50	3/2
51	4/2
52-67	4/3
68	5/3
69-84	5/4
85	6/4
86-101	6/5

Number of Inputs	Resources Used
102	7/5
103-118	7/6
119	8/6
120-135	8/7
136	9/7
137-152	9/8
153	10/8
154-169	10/9
170	11/9
171-186	11/10
187	12/10

Number of Inputs	Resources Used
188-203	12/11
204	13/11
205-220	13/12
221	14/12
222-237	14/13
238	15/13
239-254	15/14
255	16/14
256	16/15

Table 4: Resource Utilization by Depth for fixed length modules with Register Final Bit = True (LUT/ FF)

Number of Inputs	Resources Used
1	0/1
2-17	1/1
18	1/2
19-34	2/2
35	2/3
36-51	3/3
52	3/4
53-68	4/4
69	4/5
70-85	5/5
86	5/6

Number of Inputs	Resources Used
87-102	6/6
103	6/7
104-119	7/7
120	7/8
121-136	8/8
137	8/9
138-153	9/9
154	9/10
155-170	10/10
171	10/11
172-187	11/11

Number of Inputs	Resources Used
188	11/12
189-204	12/12
205	12/13
206-221	13/13
222	13/14
223-238	14/14
239	14/15
240-255	15/15
256	15/16

Table 5: Resource Utilization per bit of Data Width by Depth for variable length modules (LUT/ MUXF5/MUXF6)

Number of Inputs	Resources Used
1-16	1/0/0
16-32	3/0/0
32-48	4/0/0
48-64	5/0/0
65-80	7/1/0
81-96	8/1/0

Number of Inputs	Resources Used
97-112	9/1/0
113-128	10/1/0
129-144	12/2/1
145-160	13/2/1
161-176	14/2/1
177-192	15/2/1

Number of Inputs	Resources Used
193-208	17/2/1
209-224	18/2/1
225-240	19/2/1
241-256	20/2/1