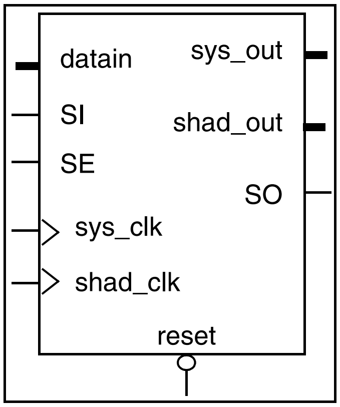
DW04_shad_reg

Shadow and Multibit Register

Version, STAR and Download Information: IP Directory

Features and Benefits



- Captures the state of system registers dynamically during system operation
- Serial access on shadow register to scan out the state of captured data
- Constructed with multi-bit flip-flop cells where possible; can be used as a simple, non-shadowed multi-bit register
- Parameterized width and number of registers (one or two)

Applications

- Diagnostics support for in-system testing
- Registers embedded for real-time emulation
- Computer, data, and telecommunications circuits that may benefit from a more efficient register implementation

Description

DW04_shad_reg is a parameterized register pair. The first register is a parallel load, parallel output system register clocked on the `sys_clk` pin with an asynchronous reset controlled by the `reset` pin. The register is implemented in multi-bit (ganged) flip-flop cells if available in the target technology. The `datain` signal drives the DW04_shad_reg inputs and the output is `sys_out`.

The second register is a shadow register, which, like the system register, is also *width* bits wide and implemented in the same target technology cells. However, the shadow register is a parallel load shift register, which captures the output of the system register when sampled by `shad_clk`. This register outputs its contents to `shad_out` (parallel) and `SO` (serial).

Table 1-1 Pin Description

Pin Name	Width	Direction	Function
datain	*width* bit(s)	Input	Data input driving the input to the system register
sys_clk	1 bit	Input	Clock that samples the system register, positive edge triggered
shad_clk	1 bit	Input	Signal that clocks the output of the system register into the shadow register, positive edge triggered
reset	1 bit	Input	Asynchronous reset signal that clears the system and shadow registers
SI	1 bit	Input	Serial scan input, clocked by shad_clk when SE is high

Table 1-1 Pin Description (Continued)

Pin Name	Width	Direction	Function
SE	1 bit	Input	Serial scan enable, active high. Enables scan only on the shadow register.
sys_out	*width* bit(s)	Output	Output of the system register
shad_out	*width* bit(s)	Output	Parallel output of shadow register that lags system register by one cycle
SO	1 bit	Output	Serial scan output from shadow register. When SE is low, represents the state of the MSB of the shadow register. When SE is high, each successive bit is shifted up one and SI is clocked into the LSB.

Table 1-2 Parameter Description

Parameter	Values	Description
width[1]	1 to 512[a]	Defines width of system and shadow registers, and the input and output buses
bld_shad_reg	0 or 1	Defines whether to build both the system and shadow registers (*bld_shad_reg* = 1) or just the system register (*bld_shad_reg* = 0)

[a] The upper bound of the legal range is a guideline to ensure reasonable compile times.

Table 1-3 Synthesis Implementations

Implementation Name	Function	License Feature Required
str	Synthesis model	DesignWare

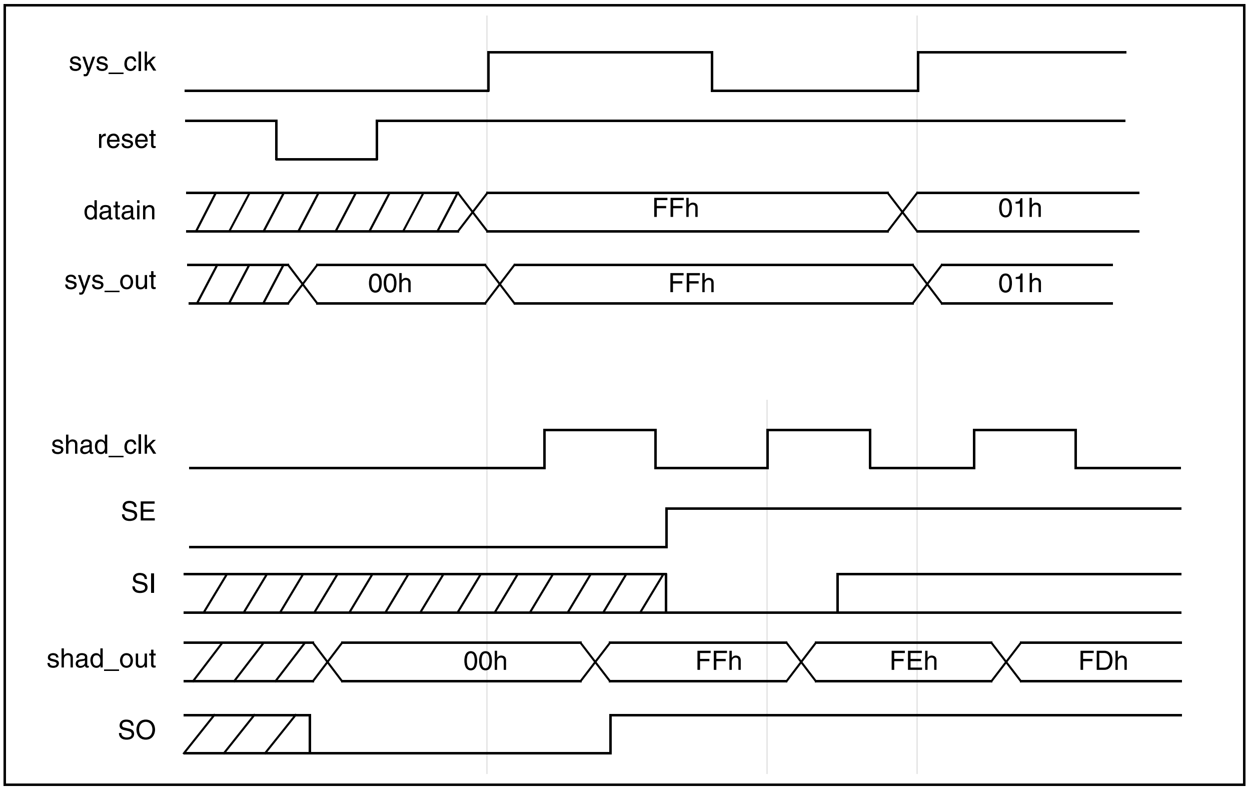
Table 1-4 Simulation Models

Model	Function
dw/sim_ver/DW04_shad_reg.v	Verilog simulation model source code

For normal operation, `SE` (scan enable) should be de-asserted (LOW). However, to scan out the contents of the shadow register, assert `SE` (HIGH) and toggle `shad_clk`. The shadow register is then in shift register mode, with the Most Significant Bit (MSB) driving `SO`, and data set-up on to the `SI` pin being clocked into the Least Significant Bit (LSB).

Timing Diagram

Figure 1-1 Timing Waveforms



Related Topics

- Memory – Registers Overview
- DesignWare Building Block IP Documentation Overview

HDL Usage Through Component Instantiation - VHDL

```
library IEEE,DWARE,DWARE;
use IEEE.std_logic_1164.all;
use DWARE.DWpackages.all;
use DWARE.DW_foundation_comp.all;

entity DW04_shad_reg_inst is
  generic ( inst_width         : POSITIVE := 8;
            inst_bld_shad_reg  : NATURAL := 1 );
  port ( inst_datain   : in std_logic_vector(inst_width-1 downto 0);
         inst_sys_clk  : in std_logic;
         inst_shad_clk : in std_logic;
         inst_reset    : in std_logic;
         inst_SI       : in std_logic;
         inst_SE       : in std_logic;
         sys_out_inst  : out std_logic_vector(inst_width-1 downto 0);
         shad_out_inst : out std_logic_vector(inst_width-1 downto 0);
         SO_inst       : out std_logic );
end DW04_shad_reg_inst;

architecture inst of DW04_shad_reg_inst is
begin

  --Instance of DW04_shad_reg
  U1 : DW04_shad_reg
    generic map ( width => inst_width,  bld_shad_reg => inst_bld_shad_reg )
    port map ( datain => inst_datain,   sys_clk => inst_sys_clk,
               shad_clk => inst_shad_clk,   reset => inst_reset,
               SI => inst_SI,   SE => inst_SE,   sys_out => sys_out_inst,
               shad_out => shad_out_inst,   SO => SO_inst );
end inst;

-- pragma translate_off
configuration DW04_shad_reg_inst_cfg_inst of DW04_shad_reg_inst is
  for inst
  end for; -- inst
end DW04_shad_reg_inst_cfg_inst;
-- pragma translate_on
```

HDL Usage Through Component Instantiation - Verilog

```
module DW04_shad_reg_inst( inst_datain, inst_sys_clk, inst_shad_clk,
                           inst_reset, inst_SI, inst_SE, sys_out_inst,
                           shad_out_inst, SO_inst );

  parameter width = 8;
  parameter bld_shad_reg = 1;

  input [width-1 : 0] inst_datain;
  input inst_sys_clk;
  input inst_shad_clk;
  input inst_reset;
  input inst_SI;
  input inst_SE;
  output [width-1 : 0] sys_out_inst;
  output [width-1 : 0] shad_out_inst;
  output SO_inst;

  // Instance of DW04_shad_reg
  DW04_shad_reg #(width, bld_shad_reg)
    U1 ( .datain(inst_datain),   .sys_clk(inst_sys_clk),
         .shad_clk(inst_shad_clk),   .reset(inst_reset),   .SI(inst_SI),
         .SE(inst_SE),   .sys_out(sys_out_inst),   .shad_out(shad_out_inst),
         .SO(SO_inst) );

endmodule
```

Copyright Notice and Proprietary Information

Destination Control Statement

All technical data contained in this publication is subject to the export control laws of the United States of America. Disclosure to nationals of other countries contrary to United States law is prohibited. It is the reader's responsibility to determine the applicable regulations and to comply with them.

Disclaimer

SYNOPSYS, INC., AND ITS LICENSORS MAKE NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Trademarks

Third-Party Links

Any links to third-party websites included in this document are for your convenience only. Synopsys does not endorse and is not responsible for such websites and their practices, including privacy practices, availability, and content.

Synopsys, Inc.
690 E. Middlefield Road
Mountain View, CA 94043

www.synopsys.com