spi_master

Description

This core implements a standard Serial Peripheral Interface Bus master controller. Data is latched on the rising edge of the SPI clock.

Verilog Template

```
spi_master #(.WIDTH(32))
spi_master_inst (
    .busy(busy),                                // output
    .clk(clk),                                  // input
    .din(din),                                  // input [WIDTH-1:0]
    .dout(dout),                                // output [WIDTH-1:0]
    .miso(miso),                                // input
    .mosi(mosi),                                // output
    .spi_clk_in(spi_clk_in),                    // input
    .spi_clk_in_negedge(spi_clk_in_negedge),    // input
    .spi_clk_in_posedge(spi_clk_in_posedge),    // input
    .spi_clk_out(spi_clk_out),                  // output
    .strobe(strobe)                             // input
);
```

Specifications

Parameters

WIDTH is the number of bits in a word. Common values are 8, 16, and 32. The default is 32.

Inputs

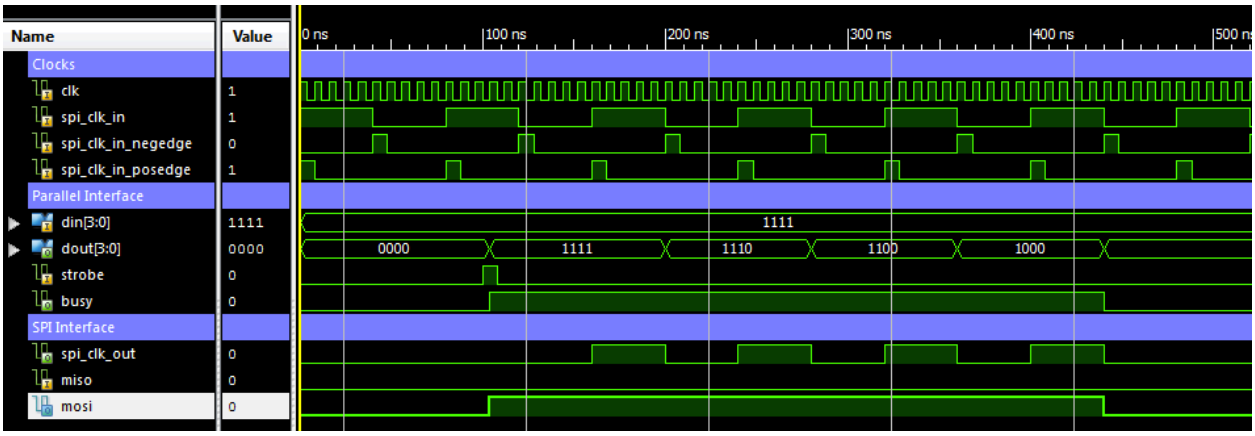
- clk: clock
- din [WIDTH]: data to be transmitted to slave when strobe is asserted
- miso: Master In Slave Out
- spi_clk_in: clock synchronous to clk and used to generate spi_clk_out
- spi_clk_in_negedge: single clk tick when spi_clk_in has a negative edge
- spi_clk_in_posedge: single clk tick when spi_clk_in has a positive edge
- strobe: used to start the transmit and receive state machine

Outputs

- busy: asserted when the core is transmitting and receiving
- dout [WIDTH]: contains data received from slave after strobe is asserted and after the deassertion of busy
- mosi: Master Out Slave In

- spi_clk_out: clock synchronous to miso and mosi and sent to the slave

Waveform



Requirements

Language for Synthesis: Verilog 2001
Synthesis Tool: Xilinx XST 13.2
Language for Verification: Verilog 2001
Verification Tool: Xilinx ISIM 13.2

References

[1] https://en.wikipedia.org/wiki/Serial_Peripheral_Interface_Bus

Contact Info

Nathan Farrington
http://nathanfarrington.com