

tb_cocotb_up.v

AUTHORS

JAY CONVERTINO

DATES

2025/04/29

INFORMATION

Brief

Test bench wrapper for cocotb

License MIT

Copyright 2025 Jay Convertino

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.BUS_WIDTH

tb_cocotb

```
module tb_cocotb #(
  parameter
  ADDRESS_WIDTH
  =
  32,
  parameter
  BUS_WIDTH
  =
  4,
  parameter
  WORD_WIDTH
  =
  4,
  parameter
```

```

CLOCK_SPEED
=
100000000,
parameter
SELECT_WIDTH
=
16,
parameter
DEFAULT_RATE_DIV
=
0,
parameter
DEFAULT_CPOL
=
0,
parameter
DEFAULT_CPHA
=
0,
parameter
FIFO_ENABLE
=
0
)

```

```

input
clk,
input
rstn,
input
up_rreq,
output
up_rack,
input
[ADDRESS_WIDTH-(BUS_WIDTH/2)-1:0]
up_raddr,
output
[(BUS_WIDTH*8)-1:0]
up_rdata,
input
up_wreq,
output
up_wack,
input
[ADDRESS_WIDTH-(BUS_WIDTH/2)-1:0]
up_waddr,
input
[(BUS_WIDTH*8)-1:0]
up_wdata,
output
irq,
output
sclk,
output
mosi,
input
miso,
output
[SELECT_WIDTH-1:0]
ss_n
)

```

SPI Master core with axis input/output data. Read/Write is size of BUS_WIDTH bytes. Write activates core for read.

Parameters

ADDRESS_WIDTH parameter	Width of the uP address port, max 32 bit.
BUS_WIDTH parameter	Width of the uP bus data port(can not be less than 2 bytes, max tested is 4).
WORD_WIDTH parameter	Width of each SPI Master word. This will also set the bits used in the TX/RX data registers. Must be less than or equal to BUS_WIDTH 1 to 4.
CLOCK_SPEED parameter	This is the aclk frequency in Hz, this is the the frequency used for the bus and is divided by the rate.
SELECT_WIDTH parameter	Bit width of the slave select, defaults to 16 to match altera spi ip.
DEFAULT_RATE_DIV parameter	Default divider value of the main clock to use for the spi data output clock rate. 0 is 2 ($2^{(X+1)}$ X is the DEFAULT_RATE_DIV)
DEFAULT_CPOL parameter	Default clock polarity for the core (0 or 1).
DEFAULT_CPHA parameter	Default clock phase for the core (0 or 1).

Ports

clk input	Clock for all devices in the core
rstn input	Negative reset
up_rreq input	uP bus read request
up_rack output	uP bus read ack
up_raddr input [ADDRESS_WIDTH-(BUS_WIDTH/ 2)- 1:0]	uP bus read address
up_rdata output [(BUS_WIDTH* 8)- 1:0]	uP bus read data
up_wreq input [(BUS_WIDTH* 8)- 1:0]	uP bus write request
up_wack output [(BUS_WIDTH* 8)- 1:0]	uP bus write ack
up_waddr input [ADDRESS_WIDTH-(BUS_WIDTH/ 2)- 1:0]	uP bus write address
up_wdata input [(BUS_WIDTH* 8)- 1:0]	uP bus write data
irq output [(BUS_WIDTH* 8)- 1:0]	Interrupt when data is received
sclk output [(BUS_WIDTH* 8)- 1:0]	spi clock, should only drive output pins to devices.
mosi output [(BUS_WIDTH* 8)- 1:0]	transmit for master output
miso input [(BUS_WIDTH* 8)- 1:0]	receive for master input
ss_n output [SELECT_WIDTH- 1:0]	slave select output

INSTANTIATED MODULES

dut

Device under test, up_spi_master