tb_cocotb.v

AUTHORS

JAY CONVERTINO

DATES

2025/05/21

INFORMATION

Brief

Test bench wrapper for cocotb

License MIT

tb_cocotb

```
module tb_cocotb #(
parameter
RAM_DEPTH
 =
1,
parameter
BYTE_WIDTH
 =
1,
parameter
ADDR_WIDTH
 =
1,
parameter
```

```
HEX_FILE
 =
"",
parameter
RAM_TYPE
 =
"block",
parameter
RD_CLOCK_SPEED
 =
10000000,
parameter
WR_CLOCK_SPEED
 =
10000000
) ( input rd_clk, input rd_rstn, input rd_en, output [(BYTE_WIDTH*8)-1:0] rd
```

Generic Dual Port RAM wrapper

Parameters

RAM_DEPTH parameter	Number of words using the size of BYTE_WIDTH.
BYTE_WIDTH parameter	Width of the data bus in bytes.
ADDR_WIDTH parameter	Width of the address bus in bits.
HEX_FILE parameter	Read a hex value text file as the initial state of the RAM.
RAM_TYPE parameter	Used to set the ram_style atribute.
RD_CLOCK_SPEED parameter	COCOTB CLOCK SPEED FOR READ IN HZ.
WR_CLOCK_SPEED parameter	COCOTB CLOCK SPEED FOR WRITE IN HZ.

Ports

rd_clk	Read clock positive edge
rd_rstn	Read reset active low
rd_en	Read enable active high
rd_data	Read data output
rd_addr	Read data address select
wr_clk	Write clock positive edge
wr_rstn	Write reset active low
wr_en	Write enable active high
wr_ben	Write byte enable, each bit represents one byte of write data.
wr_data	Write data input
wr_addr	Write data address select

INSTANTIATED MODULES

dut

```
dc_block_ram #(
                                                                                .
RAM_DEPTH(RAM_DEPTH),
                                                                                .
BYTE_WIDTH(BYTE_WIDTH),
                                                                                .
ADDR_WIDTH(ADDR_WIDTH),
                                                                                .
HEX_FILE(HEX_FILE),
                                                                                .
RAM_TYPE(RAM_TYPE)
) dut ( .rd_clk(rd_clk), .rd_rstn(rd_rstn), .rd_en(rd_en), .rd_data(rd_data)
```

Device under test, dc_block_ram