

tb_cocotb_up.v

AUTHORS

JAY CONVERTINO

DATES

2025/04/01

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
  GPIO_WIDTH
  =
  32,
  parameter
```

```
    IRQ_ENABLE
    =
    0
) ( input clk, input rstn, input up_rreq, output up_rack, input [ADDRESS_WIDTH-1:0] up_raddr, output [BUS_WIDTH-1:0] up_rdata, input [GPIO_WIDTH-1:0] gpio_io_i, output [GPIO_WIDTH-1:0] gpio_io_o, output [GPIO_WIDTH-1:0] gpio_io_t )
```

uP GPIO testbench

Parameters

ADDRESS_WIDTH	Width of the uP address port, max 32 bit.
BUS_WIDTH	Width of the uP bus data port.
GPIO_WIDTH	Width of the GPIO for inputs and outputs
IRQ_ENABLE	Enable interrupt

Ports

clk	Clock for all devices in the core
rstn	Negative reset
up_rreq	uP bus read request
up_rack	uP bus read ack
up_raddr	uP bus read address
up_rdata	uP bus read data
up_wreq	uP bus write request
up_wack	uP bus write ack
up_waddr	uP bus write address
up_wdata	uP bus write data
irq	Interrupt when data is received
gpio_io_i	Input for GPIO
gpio_io_o	Output for GPIO
gpio_io_t	Tristate for GPIO

INSTANTIATED MODULES

dut

```
up_gpio #(
    ADDRESS_WIDTH(ADDRESS_WIDTH),
    BUS_WIDTH(BUS_WIDTH),
    GPIO_WIDTH(GPIO_WIDTH),
    IRQ_ENABLE(IRQ_ENABLE)
) dut ( .clk(clk), .rstn(rstn), .up_rreq(up_rreq), .up_rack(up_rack), .up_raddr(up_raddr), .up_rdata(up_rdata), .up_wreq(up_wreq), .up_wack(up_wack), .up_waddr(up_waddr), .up_wdata(up_wdata), .gpio_io_i(gpio_io_i), .gpio_io_o(gpio_io_o), .gpio_io_t(gpio_io_t) )
```

Device under test, up_gpio