

tb_cocotb.v

AUTHORS

JAY CONVERTINO

DATES

2025/06/24

INFORMATION

Brief

Test bench wrapper for cocotb

License MIT

Copyright 2024 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.

tb_cocotb

```
module tb_cocotb #(
    parameter
    CLOCK_SPEED
    =
    20000000,
    parameter
    RX_BAUD_DELAY
    =
    0,
    parameter
    TX_BAUD_DELAY
    =
    0
) ( input wire aclk, input wire arstn, output wire parity_err, output wire t
```

Parameters

CLOCK_SPEED parameter	This is the aclk frequency in Hz
RX_BAUD_DELAY parameter	Delay in rx baud enable. This will delay when we sample a bit (default is midpoint when rx delay is 0).
TX_BAUD_DELAY parameter	Delay in tx baud enable. This will delay the time the bit output starts.

Ports

aclk	Clock for AXIS
arstn	Negative reset for AXIS
parity_err	Indicates error with parity check (active high)
frame_err	Indicates the diff line went to no diff before data catpure finished.
s_axis_tdata	Input data for UART TX.
s_axis_tuser	Information about the AXIS data {S,D,TYY} (4:0) Bits explained below:

```
- S  = SYNC ONLY (4)
  - 1 = Send only a sync pulse specified by TYY
  - 0 = Send normal sync + data.
- D  = DELAY ENABLED (3)
  - 1 = Make sure there is a delay of 4us
  - 0 = Send out immediatly
- TYY = TYPE OF DATA (2:0)
  - 000 = NA
  - 001 = REG (NOT IMPLIMENTED)
  - 010 = DATA
  - 100 = CMD/STATUS
```

s_axis_tvalid - When set active high the input data is valid s_axis_tready - When active high the device is ready for input data. m_axis_tdata - Output data from UART RX m_axis_tuser - Information about the AXIS data {S,D,TYY} (4:0)

Bits explained below:

```
- S  = SYNC ONLY (4)
  - 1 = Only received a sync pulse specified by TYY
  - 0 = Normal sync + data received.
- D  = DELAY BEFORE DATA (3)
  - 1 = Delay of 4us or more before data
  - 0 = No delay between data
- TYY = TYPE OF DATA (2:0)
  - 000 NA
  - 001 REG (NOT IMPLIMENTED)
  - 010 DATA
  - 100 CMD/STATUS
```

m_axis_tvalid - When active high the output data is valid m_axis_tready - When set active high the output device is ready for data. tx_active - Active high indicates transmit is in progress. tx_diff - transmit for 1553 (output to RX) rx_diff - receive for 1553 (input from TX)

INSTANTIATED MODULES

dut

```
axis_1553 #(
    CLOCK_SPEED(CLOCK_SPEED),
    RX_BAUD_DELAY(RX_BAUD_DELAY),
    TX_BAUD_DELAY(TX_BAUD_DELAY)
) dut ( .aclk(aclk), .arstn(arstn), .parity_err(parity_err), .frame_err(frame_err)
```

Device under test, axis_1553