tb_coctb.v

AUTHORS

JAY CONVERTINO

DATES

2024/12/10

INFORMATION

Brief

Test bench wrapper for cocotb

License MIT

tb_cocotb

```
module tb_cocotb #(
parameter
PARITY_ENA
 =
0,
parameter
PARITY_TYPE
 =
1,
parameter
STOP_BITS
 =
1,
parameter
```

```
DATA_BITS
 =
8,
parameter
DELAY
 =
0,
parameter
BUS_WIDTH
 =
1
) ( input aclk, input arstn, input [BUS_WIDTH*8-1:0] s_axis_tdata, input s_a
```

Test bench for AXIS UART TX, simple UART TX from AXI Streaming interface.

Parameters

PARITY_ENA parameter	Enable Parity for the data in and out.
PARITY_TYPE parameter	Set the parity type, 0 = even, 1 = odd, 2 = mark, 3 = space.
STOP_BITS parameter	Number of stop bits, 0 to crazy non-standard amounts.
DATA_BITS parameter	Number of data bits, 1 to crazy non-standard amounts.
DELAY parameter	Delay in tx data output. Delays the time to output of the data.
BUS_WIDTH parameter	BUS_WIDTH for axis bus in bytes.

Ports

aclk	Clock for AXIS
arstn	Negative reset for AXIS
s_axis_tdata	Input data for UART TX.
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.
uart_clk	Clock used for BAUD rate generation
uart_rstn	Negative reset for UART, for anything clocked on uart_clk
uart_ena	When active high enable UART transmit state.
uart_hold	Output to hold back clock in reset state till uart is in transmit state.
txd	transmit for UART (output to RX)

INSTANTIATED MODULES

dut

```
axis_uart_tx #(
                                                                    .
PARITY_ENA(PARITY_ENA),
                                                                    .
PARITY_TYPE(PARITY_TYPE),
                                                                    .
STOP_BITS(STOP_BITS),
```

```
                                                                                .
 DATA_BITS(DATA_BITS),
                                                                                .
 DELAY(DELAY),
                                                                                .
 BUS_WIDTH(BUS_WIDTH)
) dut ( .aclk(aclk), .arstn(arstn), .s_axis_tdata(s_axis_tdata), .s_axis_tva
```

Device under test, axis_uart_tx