axi_lite_uart_lite.v

AUTHORS

JAY CONVERTINO

DATES

2024/02/29

INFORMATION

Brief

AXI Lite UART is a core for interfacing with UART devices.

License MIT

axi_lite_uart_lite

AXI Lite based uart device.

Parameters

ADDRESS_WIDTH	Width of the axi address bus
BUS_WIDTH	Number of bytes for the data bus
CLOCK_SPEED	This is the aclk frequency in Hz
BAUD_RATE	Serial Baud, this can be any value including non-standard.
PARITY_TYPE	Set the parity type, 0 = none, 1 = even, 2 = odd, 3 = mark, 4 = space.
STOP_BITS	Number of stop bits, 0 to crazy non-standard amounts.
DATA_BITS	Number of data bits, 1 to 8.
RX_BAUD_DELAY	Delay in rx baud enable. This will delay when we sample a bit (default is midpoint

	when rx delay is 0).
TX_BAUD_DELAY	Delay in tx baud enable. This will delay the time the bit output starts.

Ports

aclk	Clock for all devices in the core
arstn	Negative reset
s_axi_awvalid	Axi Lite aw valid
s_axi_awaddr	Axi Lite aw addr
s_axi_awprot	Axi Lite aw prot
s_axi_awready	Axi Lite aw ready
s_axi_wvalid	Axi Lite w valid
s_axi_wdata	Axi Lite w data
s_axi_wstrb	Axi Lite w strb
s_axi_wready	Axi Lite w ready
s_axi_bvalid	Axi Lite b valid
s_axi_bresp	Axi Lite b resp
s_axi_bready	Axi Lite b ready
s_axi_arvalid	Axi Lite ar valid
s_axi_araddr	Axi Lite ar addr
s_axi_arprot	Axi Lite ar prot
s_axi_arready	Axi Lite ar ready
s_axi_rvalid	Axi Lite r valid
s_axi_rdata	Axi Lite r data
s_axi_rresp	Axi Lite r resp
s_axi_rready	Axi Lite r ready
irq	Interrupt when data is received
tx	transmit for UART (output to RX)
rx	receive for UART (input from TX)

up_rreq

```
wire up_rreq
```

uP read bus request

up_rack

```
wire up_rack
```

uP read bus acknowledge

up_raddr

```
wire [ADDRESS_WIDTH-(
BUS_WIDTH
                                                                  /
```

```
2
)-1:0] up_raddr
```

uP read bus address

up_rdata

```
wire [31:0] up_rdata
```

uP read bus request

up_wreq

```
wire up_wreq
```

uP write bus request

up_wack

```
wire up_wack
```

uP write bus acknowledge

up_waddr

```
wire [ADDRESS_WIDTH-(
BUS_WIDTH
                                                                    /
2
)-1:0] up_waddr
```

uP write bus address

up_wdata

```
wire [31:0] up_wdata
```

uP write bus data

INSTANTIANTED MODULES

inst_up_axi

```
up_axi #(
                                                                    .
AXI_ADDRESS_WIDTH(ADDRESS_WIDTH)
) inst_up_axi ( .up_rstn (arstn), .up_clk (aclk), .up_axi_awvalid(s_axi_aw
```

Module instance of up_axi for the AXI Lite bus to the uP bus.

inst_up_uart_lite

```
up_uart_lite #(
                                                                        .
ADDRESS_WIDTH(ADDRESS_WIDTH),
                                                                        .
BUS_WIDTH(BUS_WIDTH),
                                                                        .
CLOCK_SPEED(CLOCK_SPEED),
                                                                        .
BAUD_RATE(BAUD_RATE),
                                                                        .
PARITY_TYPE(PARITY_TYPE),
                                                                        .
STOP_BITS(STOP_BITS),
                                                                        .
DATA_BITS(DATA_BITS),
                                                                        .
RX_BAUD_DELAY(RX_BAUD_DELAY),
                                                                        .
TX_BAUD_DELAY(TX_BAUD_DELAY)
) inst_up_uart_lite ( .clk(aclk), .rstn(arstn), .up_rreq(up_rreq), .up_rack(
```

Module instance of up_uart creating a Logic wrapper for uart axis bus cores to interface with uP bus.