

axi_lite_write_channel_decoder.v

AUTHORS

JAY CONVERTINO

DATES

2025/12/16

INFORMATION

Brief

AXI lite write channel decoder. Block address and data paths till a valid address is presented.

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.

axi_lite_write_channel_decoder

```
module axi_lite_write_channel_decoder #(
    parameter
    integer
    ADDRESS_WIDTH
    =
    32,
    parameter
    integer
    BUS_WIDTH
    =
    4,
    parameter
    integer
    DATA_BUFFER
```

```

    =
    1,
    parameter
    integer
    TIMEOUT_BEATS
    =
    32,
    parameter
    [ADDRESS_WIDTH-1:0]
    SLAVE_ADDRESS
    =
    32'h44A20000,
    parameter
    [ADDRESS_WIDTH-1:0]
    SLAVE_REGION
    =
    32'h0000FFFF
)

output
wire
connected,
input
wire
aclk,
input
wire
arstn,
input
wire
[ADDRESS_WIDTH-1:0]
s_axi_awaddr,
input
wire
[2:0]
s_axi_awprot,
input
wire
s_axi_awvalid,
output
wire
s_axi_awready,
input
wire
[BUS_WIDTH*8-1:0]
s_axi_wdata,
input
wire
[BUS_WIDTH-1:0]
s_axi_wstrb,
input
wire
s_axi_wvalid,
output
wire
s_axi_wready,
output
wire
[1:0]
s_axi_bresp,
output
wire
s_axi_bvalid,
input
wire
s_axi_bready,

```

(

```

output
wire
[ADDRESS_WIDTH-1:0]
m_axi_awaddr,
output
wire
[2:0]
m_axi_awprot,
output
wire
m_axi_awvalid,
input
wire
m_axi_awready,
output
wire
[BUS_WIDTH*8-1:0]
m_axi_wdata,
output
wire
[BUS_WIDTH-1:0]
m_axi_wstrb,
output
wire
m_axi_wvalid,
input
wire
m_axi_wready,
input
wire
[1:0]
m_axi_bresp,
input
wire
m_axi_bvalid,
output
wire
m_axi_bready
)

```

AXI lite write channel decoder. Block address and data paths till a valid address is presented.

Parameters

ADDRESS_WIDTH parameter integer	Width of the AXI LITE address port in bits.
BUS_WIDTH parameter integer	Width of the AXI LITE bus data port in bytes.
DATA_BUFFER parameter integer	Buffer data channel, 0 to disable.
TIMEOUT_BEATS parameter integer	Number of clock cycles (beats) to count till timeout. 0 disables timeout.
SLAVE_ADDRESS parameter [ADDRESS_WIDTH- 1:0]	Array of Addresses for each slave (0 = slave 0 and so on).
SLAVE_REGION parameter [ADDRESS_WIDTH- 1:0]	Region for the address that is valid for the SLAVE ADDRESS.

Ports

connected output wire	Core has established channel connection
ack input wire	Input clock

arstn input wire	Input negative reset
s_axi_awaddr input wire [ADDRESS_WIDTH- 1:0]	Slave write input channel address
s_axi_awprot input wire [2:0]	Slave write input channel protection mode
s_axi_awvalid input wire	Slave write input channel address is valid.
s_axi_awready output wire	Slave write input channel is ready.
s_axi_wdata input wire [BUS_WIDTH* 8- 1:0]	Slave write input channel data
s_axi_wstrb input wire [BUS_WIDTH- 1:0]	Slave write input channel valid bytes
s_axi_wvalid input wire	Slave write input channel data valid
s_axi_wready output wire	Slave write input channel is ready.
s_axi_bresp output wire [1:0]	Slave write input channel response to write(s).
s_axi_bvalid output wire	Slave write input channel response valid.
s_axi_bready input wire	Slave write input channel response ready.
m_axi_awaddr	Master write output channel address.
m_axi_awprot output wire [2:0]	Master write output channel protection mode.
m_axi_awvalid output wire	Master write output channel address is valid.
m_axi_awready input wire	Master write output channel is ready.
m_axi_wdata output wire [BUS_WIDTH* 8- 1:0]	Master write output channel data.
m_axi_wstrb output wire [BUS_WIDTH- 1:0]	Master write output channel data bytes valid.
m_axi_wvalid output wire	Master write output channel data is valid.
m_axi_wvalid output wire	Master write output channel data ready.
m_axi_bresp input wire [1:0]	Master write output channel response.
m_axi_bvalid input wire	Master write output channel response valid.
m_axi_bready output wire	Master write output channel response ready.

INSTANTIATED MODULES

inst_addr_buffer

Buffer for the address

inst_addr_verify

Decoder for address bus.

inst_data_resp_buffer

If data buffer enabled, this holdbuffer will be generated.

inst_data_buffer

If data buffer enabled, this holdbuffer will be generated.