
Verilog Ethernet Components Readme



For more information and updates: <http://alexforencich.com/wiki/en/verilog/ethernet/start>

GitHub repository: <https://github.com/alexforencich/verilog-ethernet>

Introduction

Collection of Ethernet-related components for gigabit, 10G, and 25G packet processing (8 bit and 64 bit datapaths). Includes modules for handling Ethernet frames as well as IP, UDP, and ARP and the components for constructing a complete UDP/IP stack. Includes MAC modules for gigabit and 10G/25G, a 10G/25G PCS/PMA PHY module, and a 10G/25G combination MAC/PCS/PMA module. Includes various PTP related components for implementing systems that require precise time synchronization. Also includes full cocotb testbenches that utilize cocotbext-eth.

For IP and ARP support only, use `ip_complete` (1G) or `ip_complete_64` (10G/25G).

For UDP, IP, and ARP support, use `udp_complete` (1G) or `udp_complete_64` (10G/25G).

Top level gigabit and 10G/25G MAC modules are `eth_mac_*`, with various interfaces and with/without FIFOs. Top level 10G/25G PCS/PMA PHY module is `eth_phy_10g`. Top level 10G/25G MAC/PCS/PMA combination module is `eth_mac_phy_10g`.

PTP components include a configurable PTP clock (`ptp_clock`), a PTP clock CDC module (`ptp_clock_cdc`) for transferring PTP time across clock domains, and a configurable PTP period output module for precisely generating arbitrary frequencies from PTP time.

Example designs implementing a simple UDP echo server are included for the following boards:

- Alpha Data ADM-PCIE-9V3 (Xilinx Virtex UltraScale+ XCVU3P)
- BittWare 520N-MX (Intel Stratix 10 MX 1SM21CHU2F53E2VG)
- Digilent Arty A7 (Xilinx Artix 7 XC7A35T)
- Digilent Atlys (Xilinx Spartan 6 XC6SLX45)
- Intel Cyclone 10 LP (Intel Cyclone 10 10CL025YU256I7G)
- Terasic DE2-115 (Intel Cyclone IV E EP4CE115F29C7)
- Terasic DE5-Net (Intel Stratix V 5SGXEA7N2F45C2)
- Exablaze ExaNIC X10 (Xilinx Kintex UltraScale XCKU035)
- Exablaze ExaNIC X25 (Xilinx Kintex UltraScale+ XCKU3P)
- HiTech Global HTG-9200 (Xilinx Virtex UltraScale+ XCVU9P)
- HiTech Global HTG-640 (HTG-V6HXT-100GIG-565) (Xilinx Virtex 6 XC6VHX565T)

-
- Silicom fb2CG@KU15P (Xilinx Kintex UltraScale+ XCKU15P)
 - Xilinx KC705 (Xilinx Kintex 7 XC7K325T)
 - Xilinx ML605 (Xilinx Virtex 6 XC6VLX240T)
 - NetFPGA SUME (Xilinx Virtex 7 XC7V690T)
 - Digilent Nexys Video (Xilinx Artix 7 XC7A200T)
 - Intel Stratix 10 DX dev kit (Intel Stratix 10 DX 1SD280PT2F55E1VG)
 - Intel Stratix 10 MX dev kit (Intel Stratix 10 MX 1SM21CHU1F53E1VG)
 - Xilinx Alveo U50 (Xilinx Virtex UltraScale+ XCU50)
 - Xilinx Alveo U55C (Xilinx Virtex UltraScale+ XCU55C)
 - Xilinx Alveo U55N/Varium C1100 (Xilinx Virtex UltraScale+ XCU55N)
 - Xilinx Alveo U200 (Xilinx Virtex UltraScale+ XCU200)
 - Xilinx Alveo U250 (Xilinx Virtex UltraScale+ XCU250)
 - Xilinx Alveo U280 (Xilinx Virtex UltraScale+ XCU280)
 - Xilinx VCU108 (Xilinx Virtex UltraScale XCVU095)
 - Xilinx VCU118 (Xilinx Virtex UltraScale+ XCVU9P)
 - Xilinx VCU1525 (Xilinx Virtex UltraScale+ XCVU9P)
 - Xilinx ZCU102 (Xilinx Zynq UltraScale+ XCZU9EG)
 - Xilinx ZCU106 (Xilinx Zynq UltraScale+ XCZU7EV)
 - Arista 7132LB-48Y4C (Xilinx Virtex UltraScale+ XCVU9P)

Documentation

arp module

ARP handling logic with parametrizable retry timeout parameters and parametrizable datapath.

arp_cache module

Basic hash-based cache for ARP entries. Parametrizable depth.

arp_eth_rx module

ARP frame receiver with parametrizable datapath.

arp_eth_tx module

ARP frame transmitter with parametrizable datapath.

axis_eth_fcs module

Ethernet frame check sequence calculator.

axis_eth_fcs_64 module

Ethernet frame check sequence calculator with 64 bit datapath for 10G/25G Ethernet.

axis_eth_fcs_check module

Ethernet frame check sequence checker.

axis_eth_fcs_insert module

Ethernet frame check sequence inserter.

axis_gmii_rx module

AXI stream GMII/MII frame receiver with clock enable and MII select.

axis_gmii_tx module

AXI stream GMII/MII frame transmitter with clock enable and MII select.

axis_xgmii_rx_32 module

AXI stream XGMII frame receiver with 32 bit datapath.

axis_xgmii_rx_64 module

AXI stream XGMII frame receiver with 64 bit datapath.

axis_xgmii_tx_32 module

AXI stream XGMII frame transmitter with 32 bit datapath.

axis_xgmii_tx_64 module

AXI stream XGMII frame transmitter with 64 bit datapath.

eth_arb_mux module

Ethernet frame arbitrated multiplexer with parametrizable data width and port count. Supports priority and round-robin arbitration.

eth_axis_rx module

Ethernet frame receiver with parametrizable datapath.

eth_axis_tx module

Ethernet frame transmitter with parametrizable datapath.

eth_demux module

Ethernet frame demultiplexer with parametrizable data width and port count. Supports priority and round-robin arbitration.

eth_mac_1g module

Gigabit Ethernet MAC with GMII interface.

eth_mac_1g_fifo module

Gigabit Ethernet MAC with GMII interface and FIFOs.

eth_mac_1g_gmii module

Tri-mode Ethernet MAC with GMII/MII interface and automatic PHY rate adaptation logic.

eth_mac_1g_gmii_fifo module

Tri-mode Ethernet MAC with GMII/MII interface, FIFOs, and automatic PHY rate adaptation logic.

eth_mac_1g_rgmii module

Tri-mode Ethernet MAC with RGMII interface and automatic PHY rate adaptation logic.

eth_mac_1g_rgmii_fifo module

Tri-mode Ethernet MAC with RGMII interface, FIFOs, and automatic PHY rate adaptation logic.

eth_mac_10g module

10G/25G Ethernet MAC with XGMII interface. Datapath selectable between 32 and 64 bits.

eth_mac_10g_fifo module

10G/25G Ethernet MAC with XGMII interface and FIFOs. Datapath selectable between 32 and 64 bits.

eth_mac_mii module

Ethernet MAC with MII interface.

eth_mac_mii_fifo module

Ethernet MAC with MII interface and FIFOs.

eth_mac_phy_10g module

10G/25G Ethernet MAC/PHY combination module with SERDES interface.

eth_mac_phy_10g_fifo module

10G/25G Ethernet MAC/PHY combination module with SERDES interface and FIFOs.

eth_mac_phy_10g_rx module

10G/25G Ethernet MAC/PHY combination module with SERDES interface, RX path.

eth_mac_phy_10g_tx module

10G/25G Ethernet MAC/PHY combination module with SERDES interface, TX path.

eth_mux module

Ethernet frame multiplexer with parametrizable data width and port count. Supports priority and round-robin arbitration.

eth_phy_10g module

10G/25G Ethernet PCS/PMA PHY.

eth_phy_10g_rx module

10G/25G Ethernet PCS/PMA PHY receive-side logic.

eth_phy_10g_rx_ber_mon module

10G/25G Ethernet PCS/PMA PHY BER monitor.

eth_phy_10g_rx_frame_sync module

10G/25G Ethernet PCS/PMA PHY frame synchronizer.

eth_phy_10g_tx module

10G/25G Ethernet PCS/PMA PHY transmit-side logic.

gmii_phy_if module

GMII/MII PHY interface and clocking logic.

ip module

IPv4 block with 8 bit data width for gigabit Ethernet. Manages IPv4 packet transmission and reception. Interfaces with ARP module for MAC address lookup.

ip_64 module

IPv4 block with 64 bit data width for 10G/25G Ethernet. Manages IPv4 packet transmission and reception. Interfaces with ARP module for MAC address lookup.

ip_arb_mux module

IP frame arbitrated multiplexer with parametrizable data width and port count. Supports priority and round-robin arbitration.

ip_complete module

IPv4 module with ARP integration.

Top level for gigabit IP stack.

ip_complete_64 module

IPv4 module with ARP integration and 64 bit data width for 10G/25G Ethernet.

Top level for 10G/25G IP stack.

ip_demux module

IP frame demultiplexer with parametrizable data width and port count. Supports priority and round-robin arbitration.

ip_eth_rx module

IP frame receiver.

ip_eth_rx_64 module

IP frame receiver with 64 bit datapath for 10G/25G Ethernet.

ip_eth_tx module

IP frame transmitter.

ip_eth_tx_64 module

IP frame transmitter with 64 bit datapath for 10G/25G Ethernet.

ip_mux module

IP frame multiplexer with parametrizable data width and port count. Supports priority and round-robin arbitration.

lfsr module

Fully parametrizable combinatorial parallel LFSR/CRC module.

mii_phy_if module

MII PHY interface and clocking logic.

ptp_clock module

PTP clock module with PPS output. Generates both 64 bit and 96 bit timestamp formats. Fine frequency adjustment supported with configurable fractional nanoseconds field.

ptp_clock_cdc module

PTP clock CDC module with PPS output. Use this module to transfer and deskew a free-running PTP clock across clock domains. Supports both 64 and 96 bit timestamp formats.

ptp_td_leaf module

PTP time distribution leaf clock module. Accepts PTP time distribution messages from the [ptp_td_phc](#) module, and outputs both the 96-bit time-of-day timestamp and 64-bit relative timestamp in the destination clock domain, as well as both single-cycle and stretched PPS outputs. Also supports pipelining the serial data input, automatically compensating for the pipeline delay.

ptp_td_phc module

PTP time distribution master clock module. Generates PTP time distribution messages over a serial interface that can provide PTP time to one or more leaf clocks ([ptp_td_leaf](#)), as well as both single-cycle and stretched PPS outputs. The fractional nanoseconds portion is shared between the time-of-day and relative timestamps to support reconstruction of the 96-bit time-of-day timestamp from a truncated relative timestamp. The module supports coarse setting of both the ToD and relative timestamps as well as atomically applying offsets to the ToD and relative timestamps and the shared fractional nanoseconds.

ptp_ts_extract module

PTP timestamp extract module. Use this module to extract a PTP timestamp embedded in the [tuser](#) sideband signal of an AXI stream interface.

ptp_perout module

PTP period output module. Generates a pulse output, configurable in absolute start time, period, and width, based on PTP time from a PTP clock.

rgmii_phy_if module

RGMIi PHY interface and clocking logic.

udp module

UDP block with 8 bit data width for gigabit Ethernet. Manages UDP packet transmission and reception.

udp_64 module

UDP block with 64 bit data width for 10G/25G Ethernet. Manages UDP packet transmission and reception.

udp_arb_mux module

UDP frame arbitrated multiplexer with parametrizable data width and port count. Supports priority and round-robin arbitration.

udp_checksum_gen module

UDP checksum generator module. Calculates UDP length, IP length, and UDP checksum fields.

udp_checksum_gen_64 module

UDP checksum generator module with 64 bit datapath. Calculates UDP length, IP length, and UDP checksum fields.

udp_complete module

UDP module with IPv4 and ARP integration.

Top level for gigabit UDP stack.

udp_complete_64 module

UDP module with IPv4 and ARP integration and 64 bit data width for 10G Ethernet.

Top level for 10G/25G UDP stack.

udp_demux module

UDP frame demultiplexer with parametrizable data width and port count. Supports priority and round-robin arbitration.

udp_ip_rx module

UDP frame receiver.

udp_ip_rx_64 module

UDP frame receiver with 64 bit datapath for 10G/25G Ethernet.

udp_ip_tx module

UDP frame transmitter.

udp_ip_tx_64 module

UDP frame transmitter with 64 bit datapath for 10G/25G Ethernet.

udp_mux module

UDP frame multiplexer with parametrizable data width and port count. Supports priority and round-robin arbitration.

xgmii_baser_dec_64 module

XGMII 10GBASE-R decoder for 10G PCS/PMA PHY.

xgmii_baser_enc_64 module

XGMII 10GBASE-R encoder for 10G PCS/PMA PHY.

xgmii_deinterleave module

XGMII de-interleaver for interfacing with PHY cores that interleave the control and data lines.

xgmii_interleave module

XGMII interleaver for interfacing with PHY cores that interleave the control and data lines.

Common signals

```
1 tdata    : Data (width generally DATA_WIDTH)
2 tkeep    : Data word valid (width generally KEEP_WIDTH, present on _64
   modules)
3 tvalid   : Data valid
4 tready   : Sink ready
5 tlast    : End-of-frame
6 tuser    : Bad frame (valid with tlast & tvalid)
```

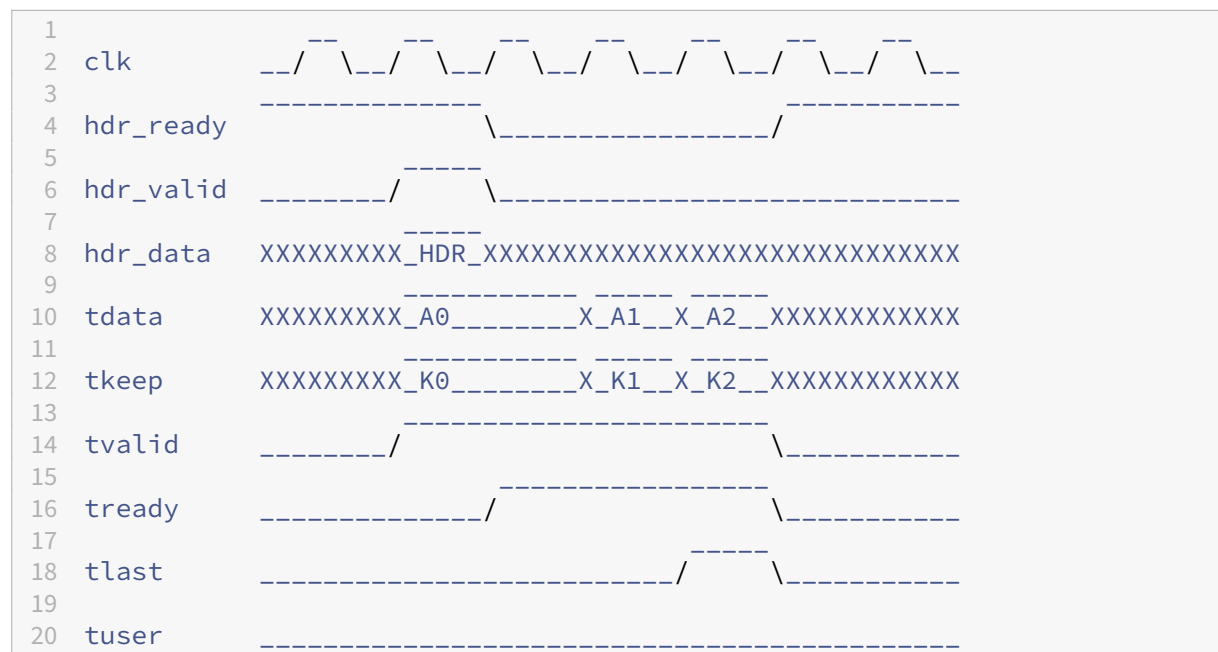
Source Files

1	rtl/arp.v	: ARP handling logic
2	rtl/arp_cache.v	: ARP LRU cache
3	rtl/arp_eth_rx.v	: ARP frame receiver
4	rtl/arp_eth_tx.v	: ARP frame transmitter
5	rtl/eth_arb_mux.py generator	: Ethernet frame arbitrated multiplexer
6	rtl/axis_eth_fcs.v	: Ethernet FCS calculator
7	rtl/axis_eth_fcs_64.v	: Ethernet FCS calculator (64 bit)
8	rtl/axis_eth_fcs_insert.v	: Ethernet FCS inserter
9	rtl/axis_eth_fcs_check.v	: Ethernet FCS checker
10	rtl/axis_gmii_rx.v	: AXI stream GMII/MII receiver
11	rtl/axis_gmii_tx.v	: AXI stream GMII/MII transmitter
12	rtl/axis_xgmii_rx_32.v	: AXI stream XGMII receiver (32 bit)
13	rtl/axis_xgmii_rx_64.v	: AXI stream XGMII receiver (64 bit)
14	rtl/axis_xgmii_tx_32.v	: AXI stream XGMII transmitter (32 bit)
15	rtl/axis_xgmii_tx_64.v	: AXI stream XGMII transmitter (64 bit)
16	rtl/eth_arb_mux.v	: Ethernet frame arbitrated multiplexer
17	rtl/eth_axis_rx.v	: Ethernet frame receiver
18	rtl/eth_axis_tx.v	: Ethernet frame transmitter
19	rtl/eth_demux.v	: Ethernet frame demultiplexer
20	rtl/eth_mac_1g.v	: Gigabit Ethernet GMII MAC
21	rtl/eth_mac_1g_fifo.v	: Gigabit Ethernet GMII MAC with FIFO
22	rtl/eth_mac_1g_gmii.v	: Tri-mode Ethernet GMII/MII MAC
23	rtl/eth_mac_1g_gmii_fifo.v FIFO	: Tri-mode Ethernet GMII/MII MAC with
24	rtl/eth_mac_1g_rgmii.v	: Tri-mode Ethernet RGMII MAC
25	rtl/eth_mac_1g_rgmii_fifo.v	: Tri-mode Ethernet RGMII MAC with FIFO
26	rtl/eth_mac_10g.v	: 10G/25G Ethernet XGMII MAC
27	rtl/eth_mac_10g_fifo.v	: 10G/25G Ethernet XGMII MAC with FIFO
28	rtl/eth_mac_mii.v	: Ethernet MII MAC
29	rtl/eth_mac_mii_fifo.v	: Ethernet MII MAC with FIFO
30	rtl/eth_mac_phy_10g.v	: 10G/25G Ethernet XGMII MAC/PHY
31	rtl/eth_mac_phy_10g_fifo.v FIFO	: 10G/25G Ethernet XGMII MAC/PHY with
32	rtl/eth_mac_phy_10g_rx.v with FIFO	: 10G/25G Ethernet XGMII MAC/PHY RX
33	rtl/eth_mac_phy_10g_tx.v with FIFO	: 10G/25G Ethernet XGMII MAC/PHY TX
34	rtl/eth_mux.v	: Ethernet frame multiplexer
35	rtl/gmii_phy_if.v	: GMII PHY interface
36	rtl/iddr.v	: Generic DDR input register
37	rtl/ip.v	: IPv4 block
38	rtl/ip_64.v	: IPv4 block (64 bit)
39	rtl/ip_arb_mux.v	: IP frame arbitrated multiplexer
40	rtl/ip_complete.v	: IPv4 stack (IP-ARP integration)
41	rtl/ip_complete_64.v bit)	: IPv4 stack (IP-ARP integration) (64
42	rtl/ip_demux.v	: IP frame demultiplexer
43	rtl/ip_eth_rx.v	: IPv4 frame receiver

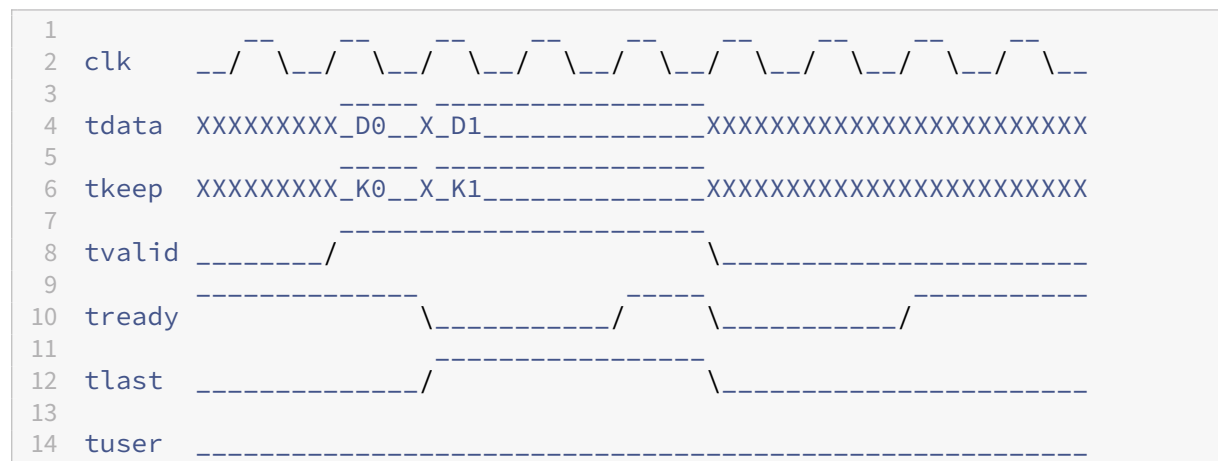
44	rtl/ip_eth_rx_64.v	: IPv4 frame receiver (64 bit)
45	rtl/ip_eth_tx.v	: IPv4 frame transmitter
46	rtl/ip_eth_tx_64.v	: IPv4 frame transmitter (64 bit)
47	rtl/ip_mux.v	: IP frame multiplexer
48	rtl/lfsr.v	: Generic LFSR/CRC module
49	rtl/mii_phy_if.v	: MII PHY interface
50	rtl/oddr.v	: Generic DDR output register
51	rtl/ptp_clock.v	: PTP clock
52	rtl/ptp_clock_cdc.v	: PTP clock CDC
53	rtl/ptp_td_leaf.v	: PTP time distribution leaf clock
54	rtl/ptp_td_phc.v	: PTP time distribution master clock
55	rtl/ptp_ts_extract.v	: PTP timestamp extract
56	rtl/ptp_perout.v	: PTP period out
57	rtl/rgmii_phy_if.v	: RGMII PHY interface
58	rtl/ssio_ddr_in.v	: Generic source synchronous IO DDR
	input module	
59	rtl/ssio_ddr_in_diff.v	: Generic source synchronous IO DDR
	differential input module	
60	rtl/ssio_ddr_out.v	: Generic source synchronous IO DDR
	output module	
61	rtl/ssio_ddr_out_diff.v	: Generic source synchronous IO DDR
	differential output module	
62	rtl/ssio_sdr_in.v	: Generic source synchronous IO SDR
	input module	
63	rtl/ssio_sdr_in_diff.v	: Generic source synchronous IO SDR
	differential input module	
64	rtl/ssio_sdr_out.v	: Generic source synchronous IO SDR
	output module	
65	rtl/ssio_sdr_out_diff.v	: Generic source synchronous IO SDR
	differential output module	
66	rtl/udp.v	: UDP block
67	rtl/udp_64.v	: UDP block (64 bit)
68	rtl/udp_arb_mux.v	: UDP frame arbitrated multiplexer
69	rtl/udp_checksum_gen.v	: UDP checksum generator
70	rtl/udp_checksum_gen_64.v	: UDP checksum generator (64 bit)
71	rtl/udp_complete.v	: UDP stack (IP-ARP-UDP)
72	rtl/udp_complete_64.v	: UDP stack (IP-ARP-UDP) (64 bit)
73	rtl/udp_demux.v	: UDP frame demultiplexer
74	rtl/udp_ip_rx.v	: UDP frame receiver
75	rtl/udp_ip_rx_64.v	: UDP frame receiver (64 bit)
76	rtl/udp_ip_tx.v	: UDP frame transmitter
77	rtl/udp_ip_tx_64.v	: UDP frame transmitter (64 bit)
78	rtl/udp_mux.v	: UDP frame multiplexer
79	rtl/xgmii_baser_dec_64.v	: XGMII 10GBASE-R decoder
80	rtl/xgmii_baser_enc_64.v	: XGMII 10GBASE-R encoder
81	rtl/xgmii_deinterleave.v	: XGMII data/control de-interleaver
82	rtl/xgmii_interleave.v	: XGMII data/control interleaver

AXI Stream Interface Example

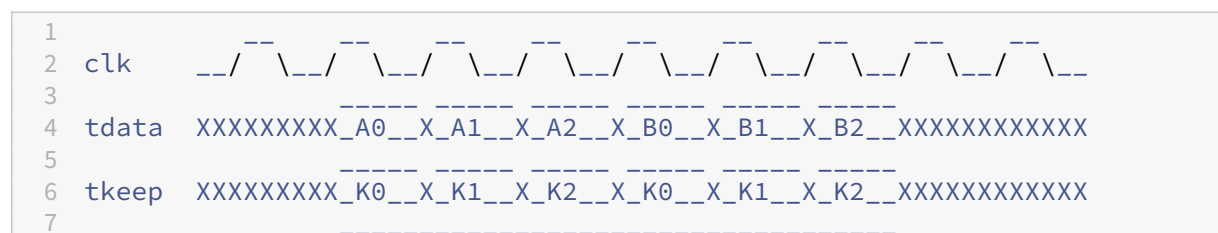
transfer with header data

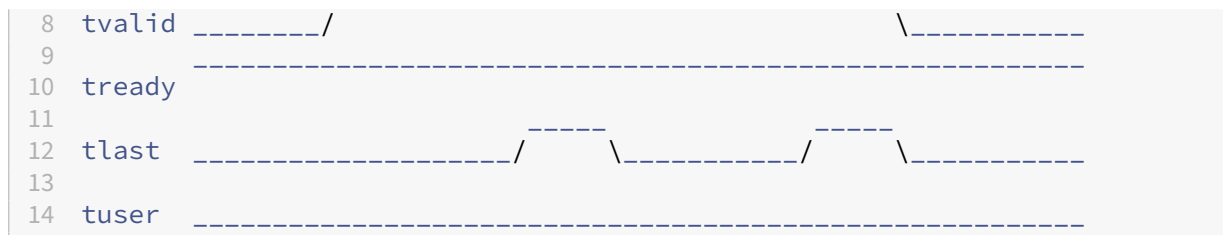


two byte transfer with sink pause after each byte

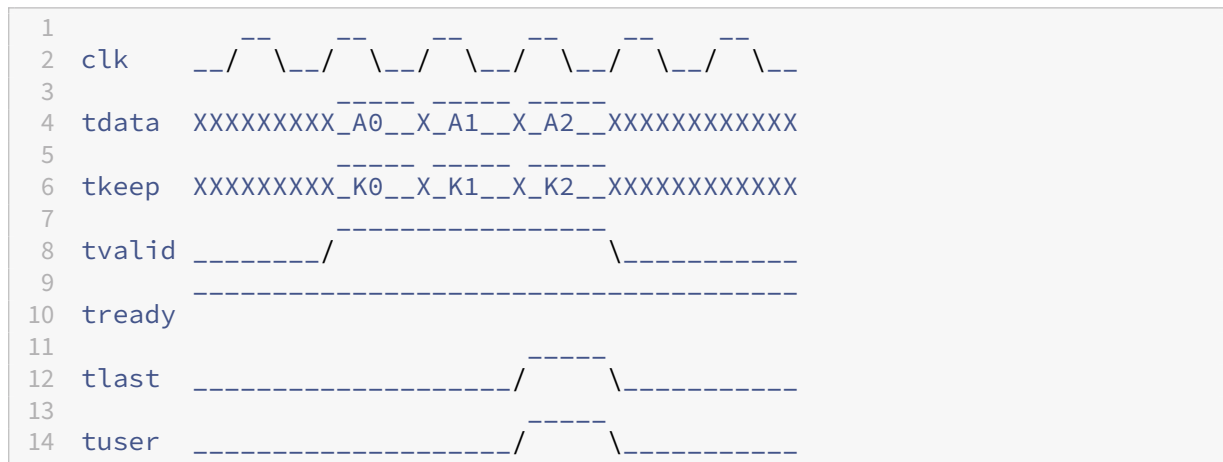


two back-to-back packets, no pauses





bad frame



Testing

Running the included testbenches requires cocotb, cocotbext-axi, cocotbext-eth, and Icarus Verilog. The testbenches can be run with pytest directly (requires cocotb-test), pytest via tox, or via cocotb makefiles.