AXI Bus

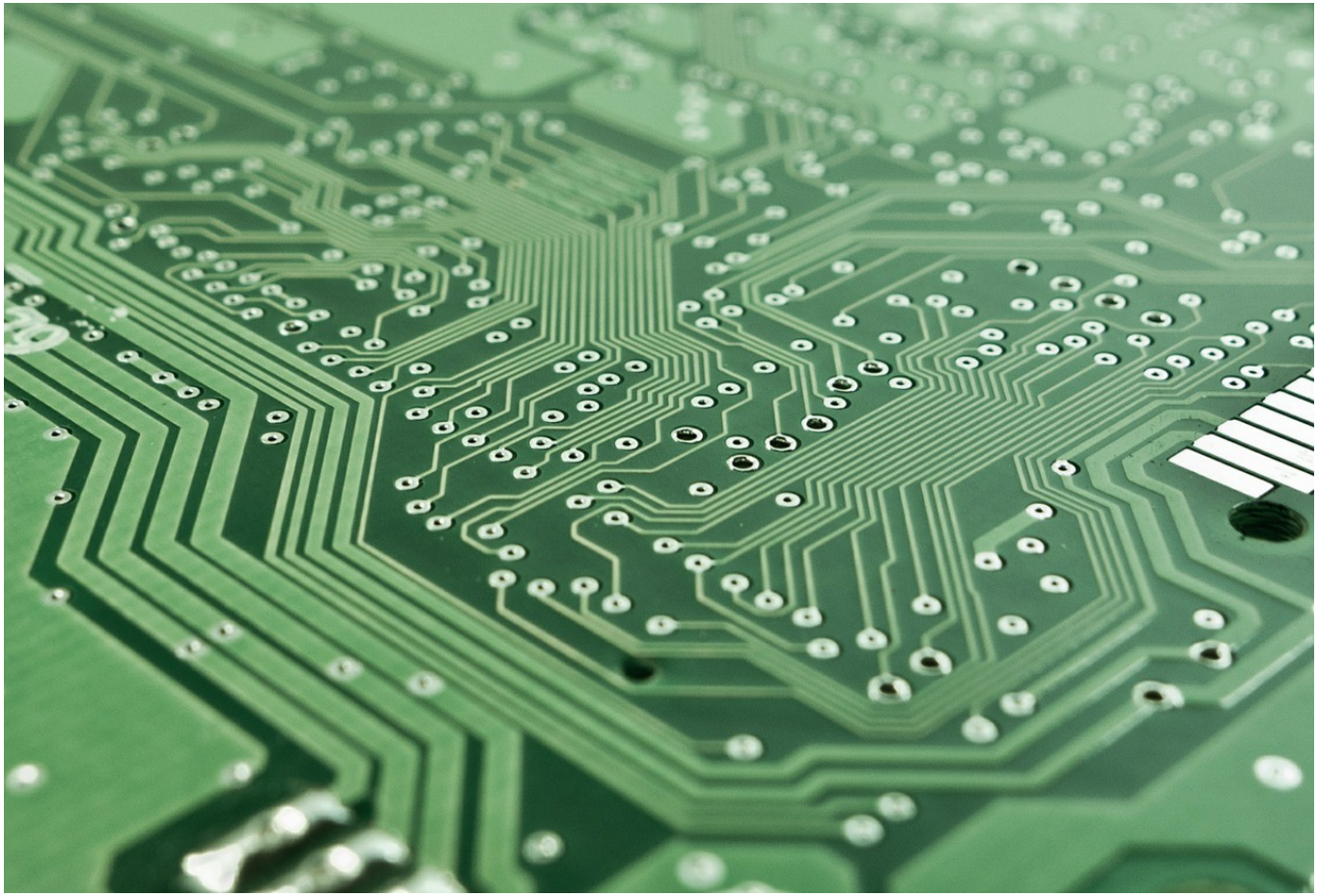
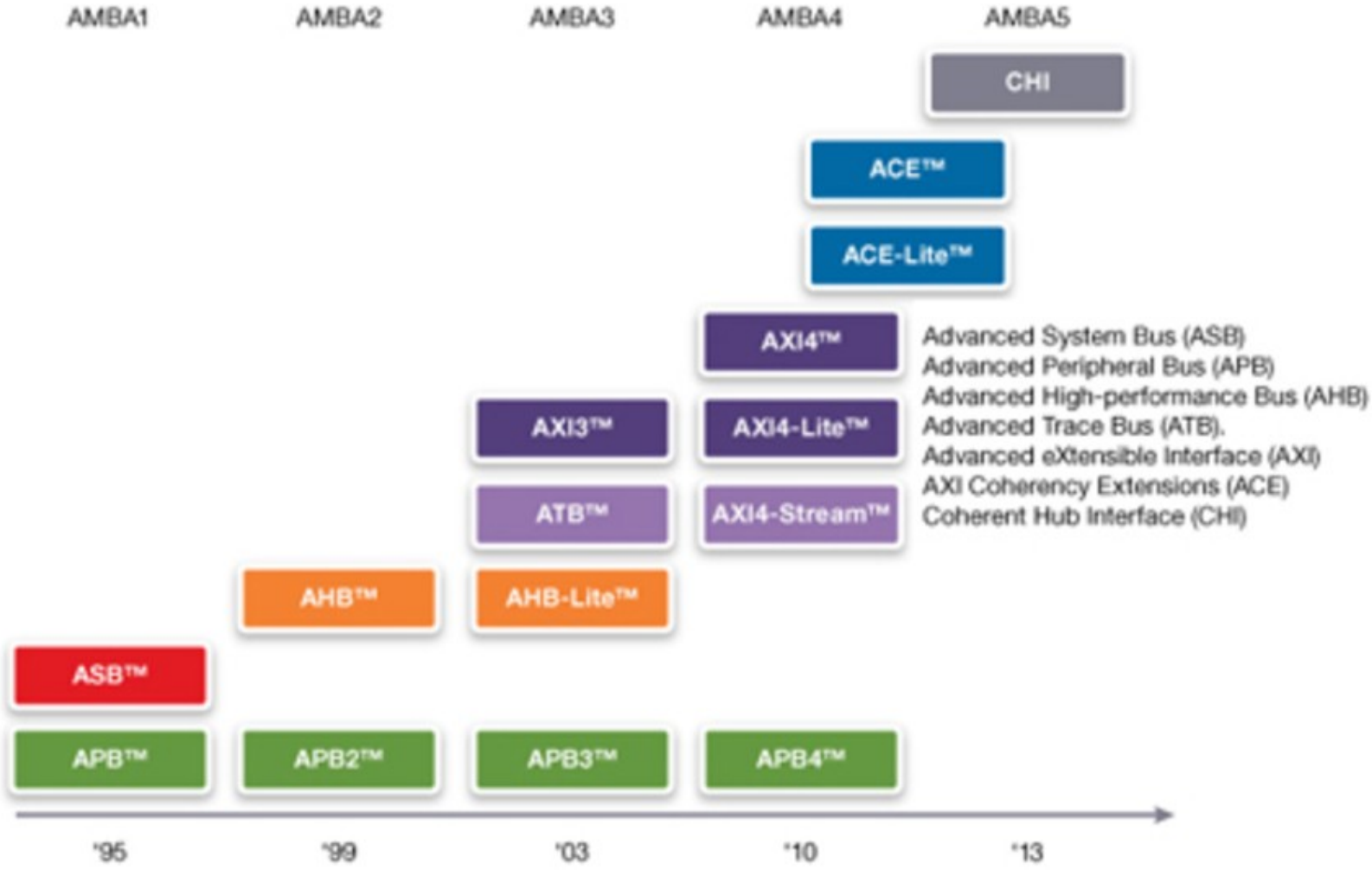


Image credit: http://blog.optimumdesign.com

Overview

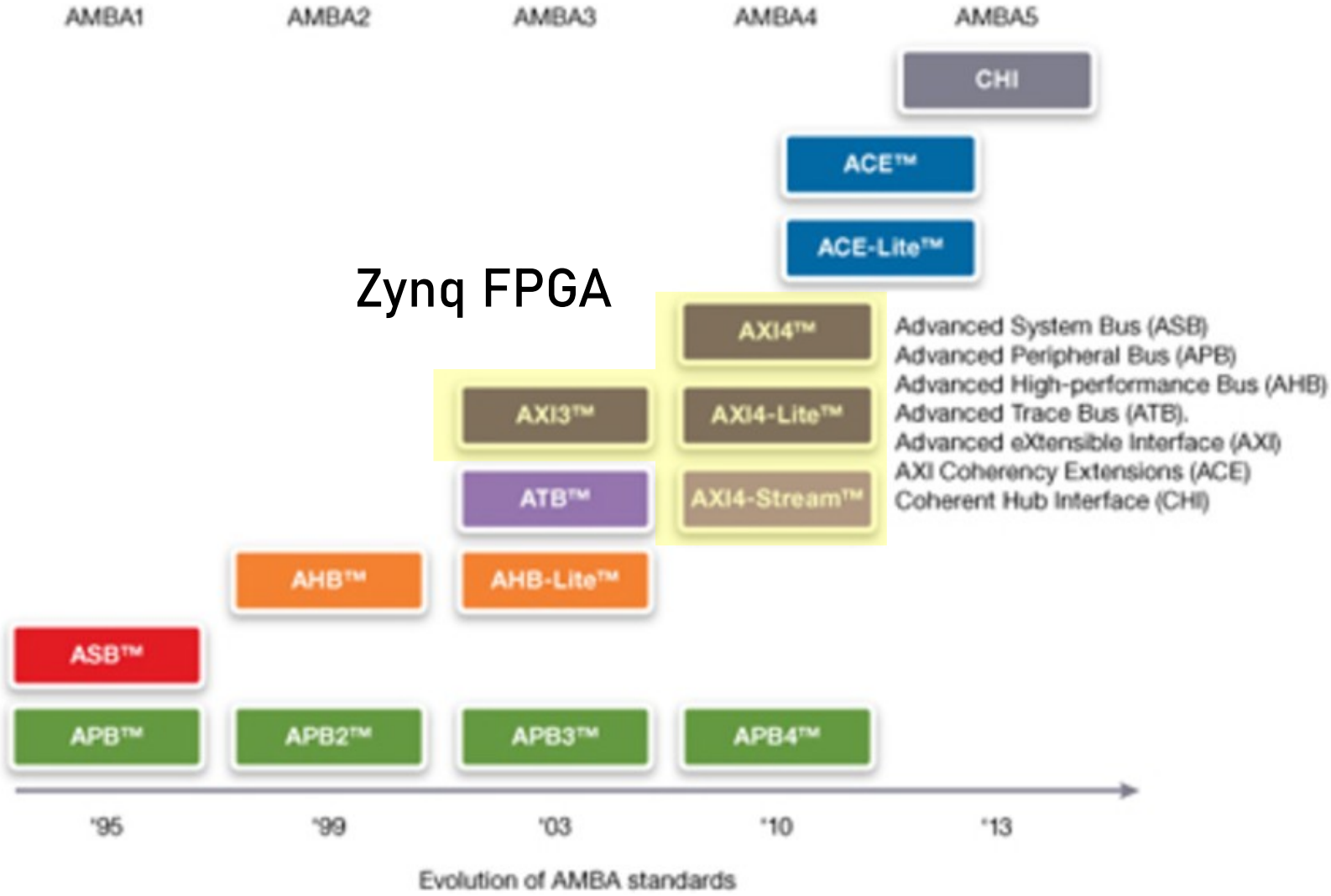
- The evolution of ARM's bus standards
- Features of the AXI bus
- Control signal operation
- Bus transactions (AXI-Stream, AXI-Lite, AXI4)
- References

Evolution of AMBA Standards



Evolution of AMBA standards

Evolution of AMBA Standards



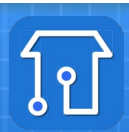
Zynq FPGA

Evolution of AMBA standards

Acronym Madness

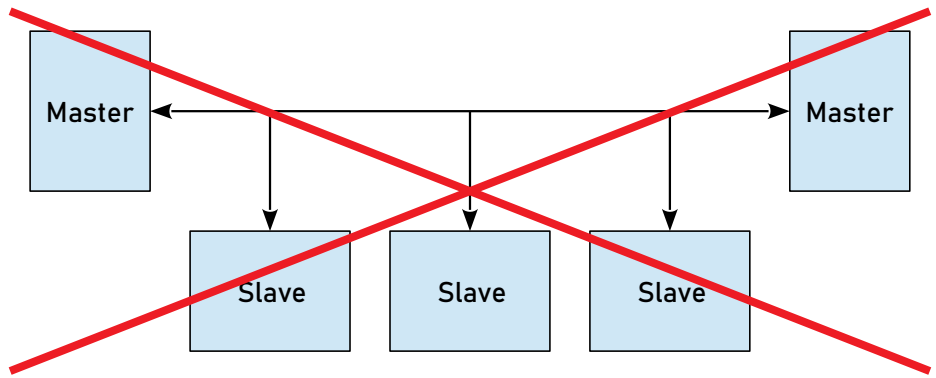
- AMBA = Advanced Microcontroller Bus Architecture

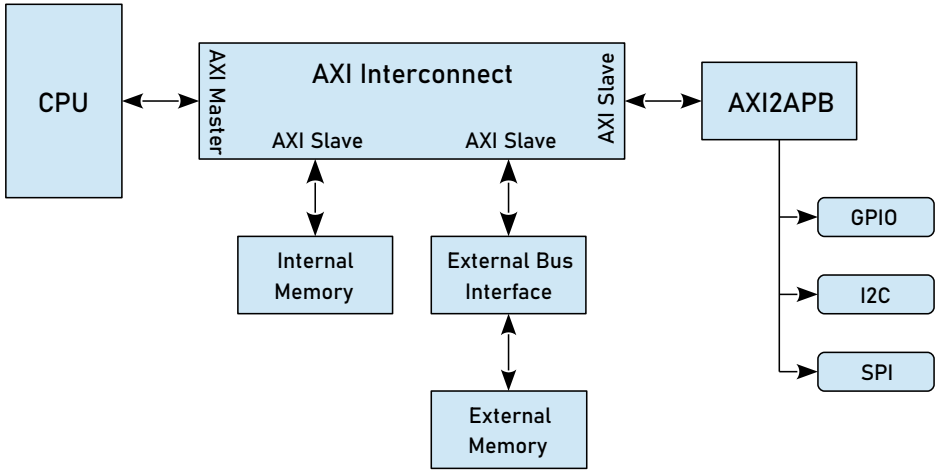
Acronym	Name	Interpretation
ASB	Advanced System Bus	Obsolete
APB	Advanced Peripheral Bus	Simple and easy for peripherals
AHB	Advanced High-Performance Bus	Used in a lot of Cortex-M designs
AXI	Advanced eXtensible Interface	Most common
ATB	Advanced Trace Bus	Part of CoreSight debug
ACE	AXI Coherency Extensions	Used in big.LITTLE systems for smartphones, tablets, etc.
CHI	Coherent Hub Interface	Highest performance for servers



What Is AXI

- Not an electrical spec
- Not a multi-master or multi-slave bus
- It is a point-to-point connection only
- Multi-slave/master requires a crossbar or interconnect



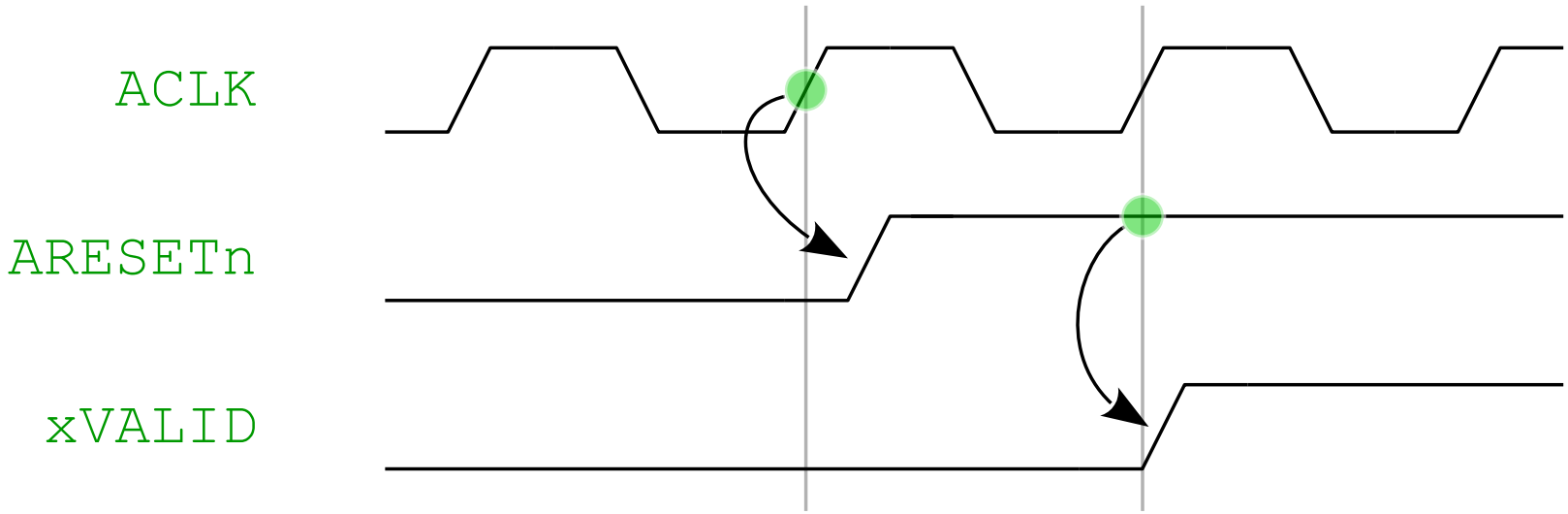


Types of AXI Bus for FPGA

- **AXI4-Stream:** For high-speed streaming data
- **AXI4-Lite:** Optimised for simple, low-throughput memory-mapped communication (i.e. to and from control and status registers)
- **AXI3:** Older version of AXI4
- **AXI4:** For high-performance memory-mapped requirements

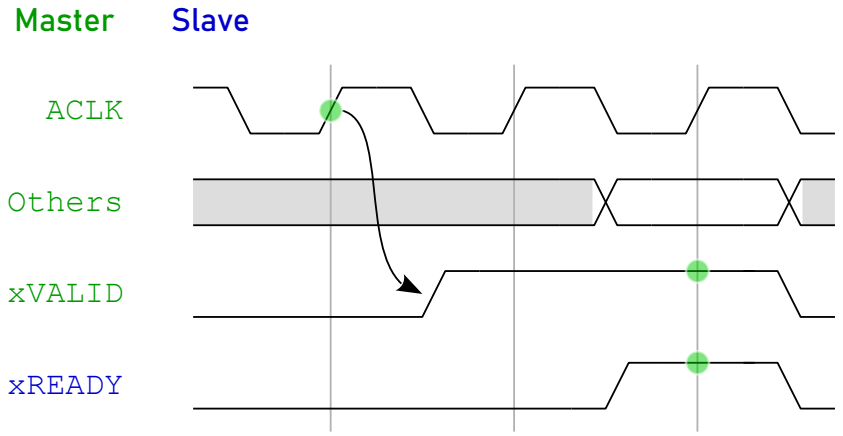
Control: Coming Out Of Reset

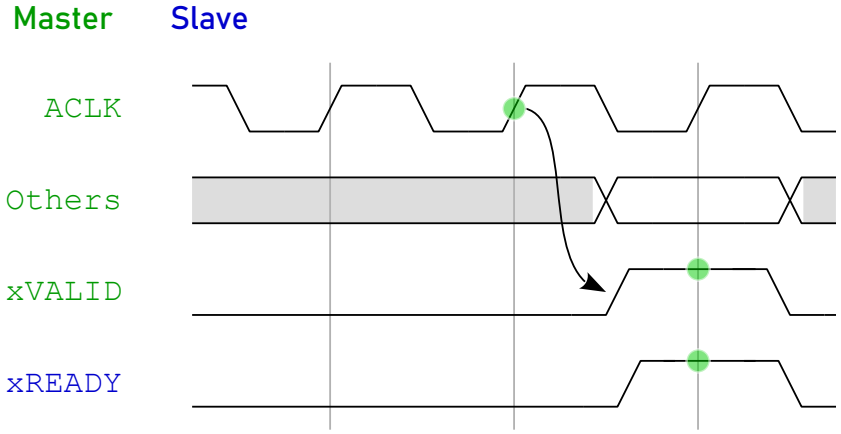
- **`ARESETn`** can asynchronously enter reset, but must synchronously exit reset
- **`xVALID`** is ready at least 1 cycle after

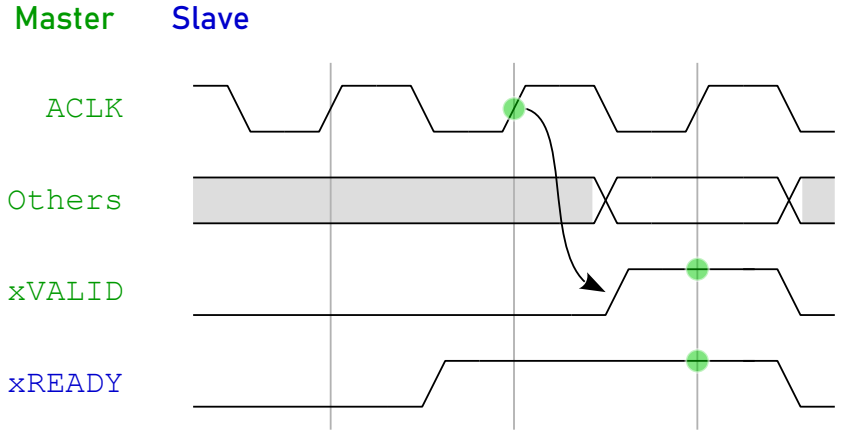


Control: Ready & Valid Signalling

- Bus values are only used when READY and VALID are high at a rising clock edge

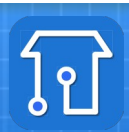






Control: Ready & Valid Signalling

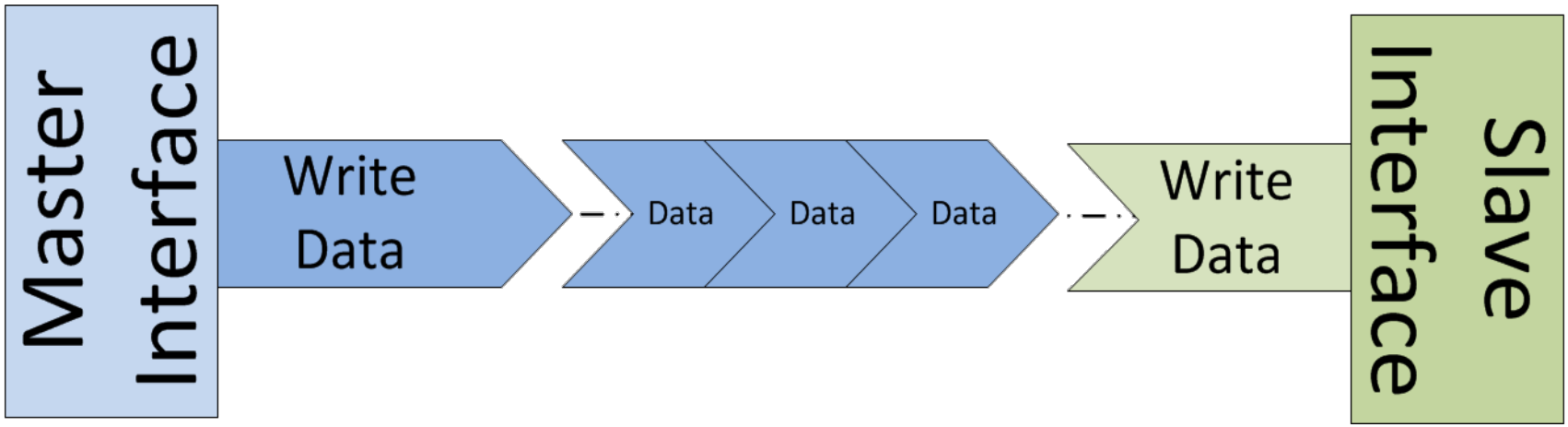
- In any transaction:
  - The **VALID** signal of one AXI component is not dependent on the **READY** signal of the other component in the transaction
  - The **READY** signal can wait for assertion of the **VALID** signal



AXI Stream

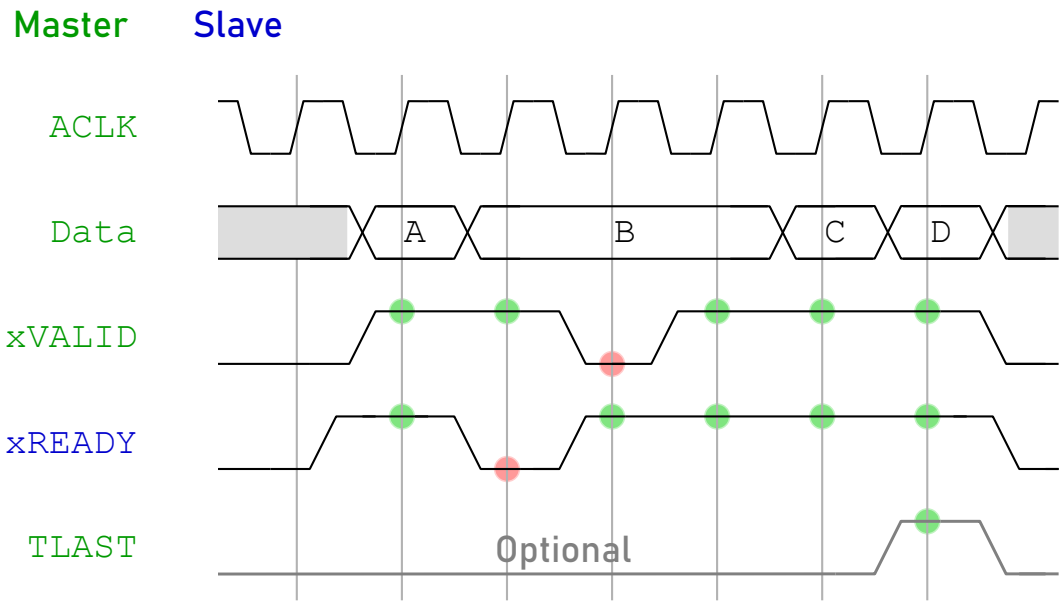
- Highest bandwidth
- No addressing
- Data only goes from Master to Slave

- Can't connect directly to the PS



Minimal AXI-Stream

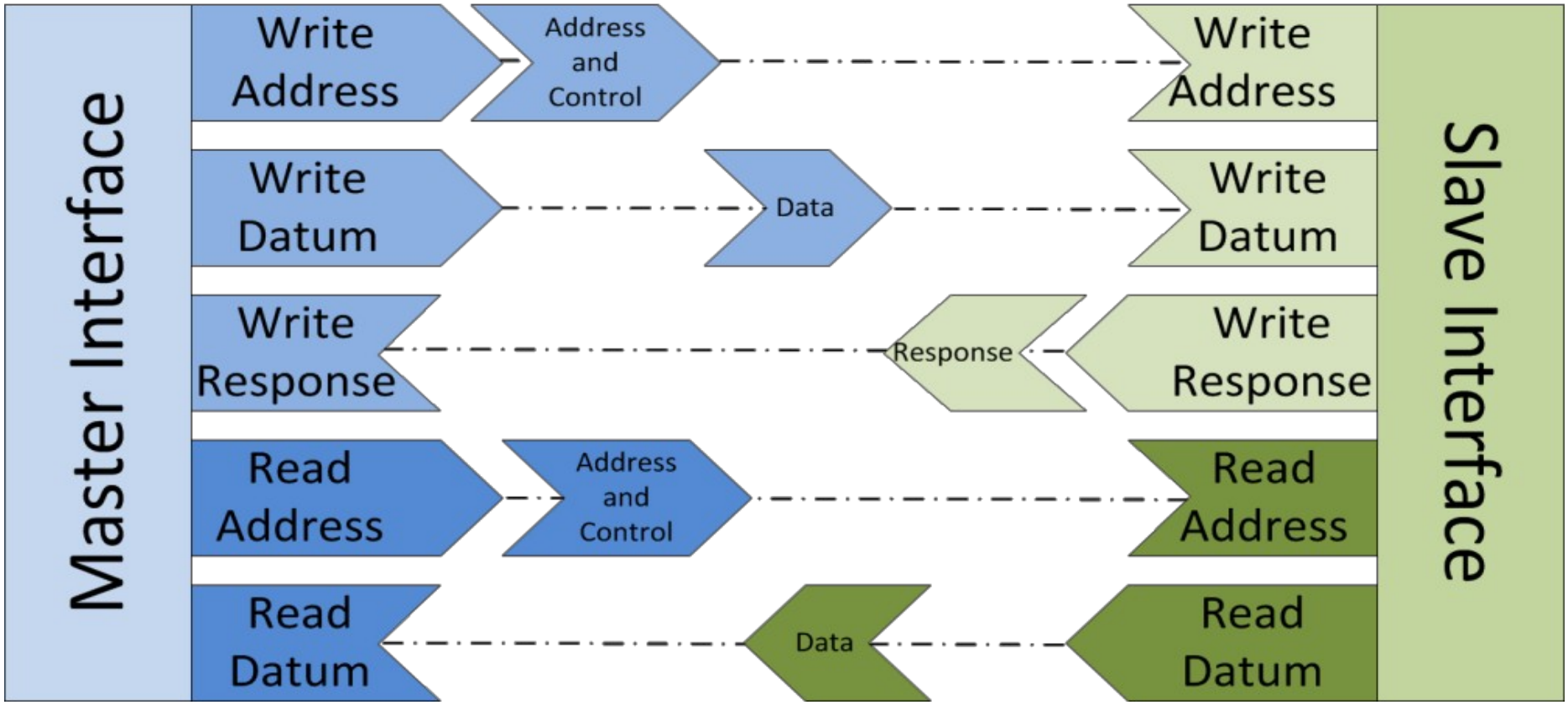
- **READY** is asserted when the slave device can accept new data
- The slave device can de-assert **READY**.
- Master stops the data transfer at most one sample later.



AXI-Lite

- AXI-lite is subset of AXI4
  - No data bursts
  - All data accesses use the full width of the data bus which can be either 32-bit or 64-bit.
  - Bytes can be masked with WSTRB
  - All accesses are Non-modifiable, Non-bufferable (no out-of-order, no cache)
  - Exclusive accesses are not supported.
- Good for small transfers such as register map access

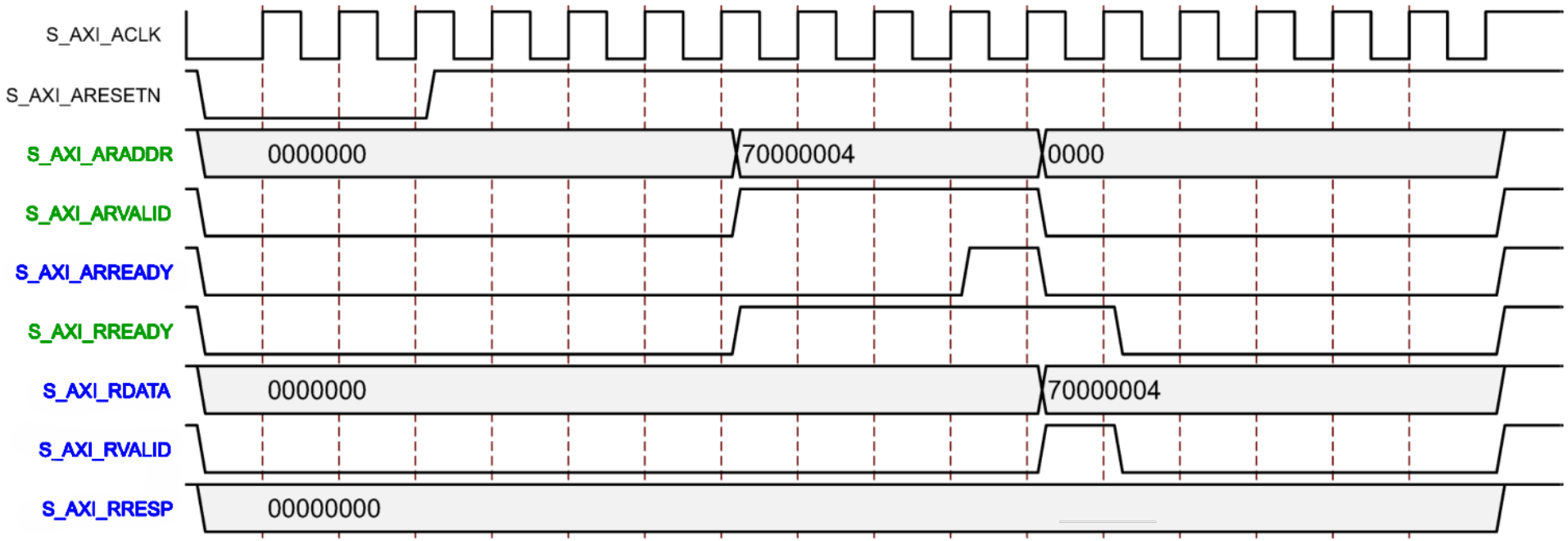
AXI4-Lite: 5 Pin Groups/Channels



- No data bursts

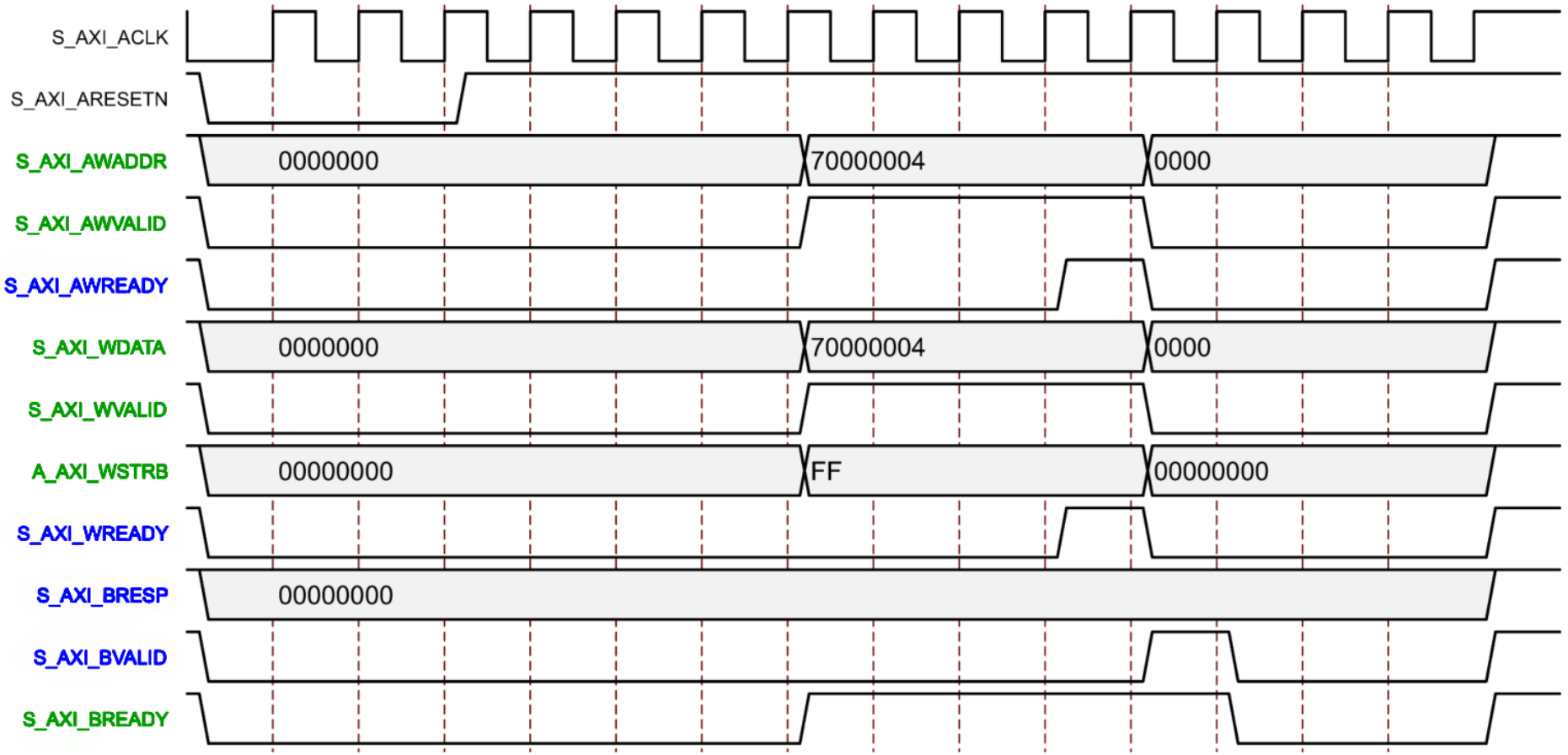
Image credit: Xilinx

AXI4-Lite Read

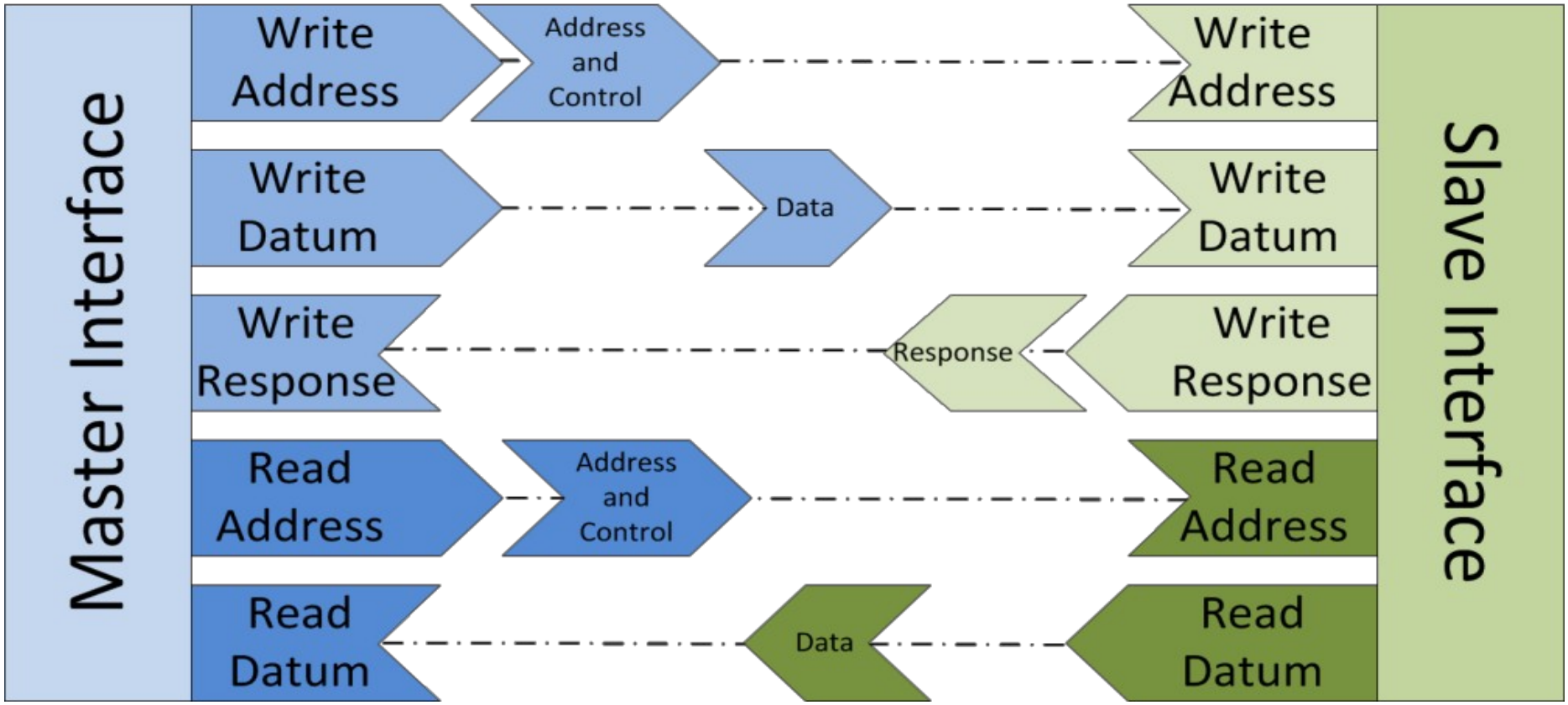


- Note: **ARADDR** is sampled/processed before **RDATA** is valid (**RVALID**)

AXI4-Lite Write



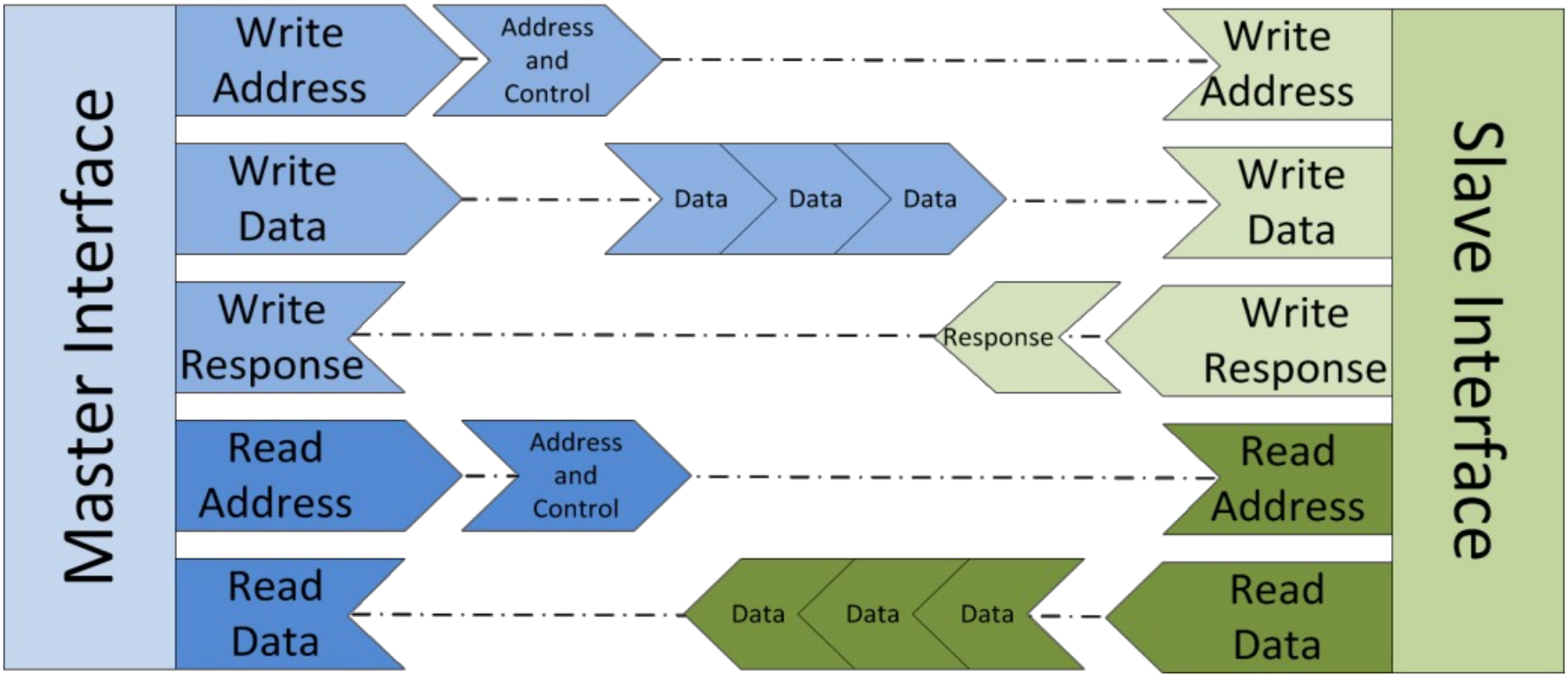
AXI4-Lite: 5 Pin Groups/Channels



- No data bursts

Image credit: Xilinx

AXI4: 5 Pin Groups/Channels



- Data bursts are possible

AXI Signalling Summary

AXI Stream	AXI-Lite	AXI4	
ACLK			
ARESETN			
		ARID	Read Address
	ARADDR		
		ARLEN	
		ARSIZE	
		ARBURST	
		ARLOCK	
	ARCACHE		
	ARPROT		
		ARQOS	
		ARREGION	
		ARLOCK	
		ARUSER	
ARVALID			
ARREADY			
		RID	Read Data
		RDATA	
		RRESP	
		RLAST	
		RUSER	
RVALID			
RREADY			

AXI Stream	AXI-Lite	AXI4	
ACLK			
ARESETN			
		AWID	Write Address
	AWADDR		
		AWLEN	
		AWSIZE	
		AWBURST	
		AWLOCK	
	AWCACHE		
		AWPROT	
		AWQOS	
		AWREGION	
		AWLOCK	
		AWUSER	
AWVALID			
AWREADY			
		WDATA	Write Data
		WSTRB	
		WLAST	
		WUSER	
WVALID			
WREADY			
		BID	Write Resp
		BRESP	
		BUSER	
	BVALID		
	BREADY		

Signalling Details: AXI Write Address

- AWID - The identification tag for the write address group of signals
- AWADDR – Write address
- AWLEN – The exact number of transfers in a burst (AXI3/4 differ)
- AWSIZE - The size of each transfer in the burst
- AWBURST – Burst type
- AWLOCK – Atomic lock characteristics (AXI3/4 differ)
- AWCACHE – Indicates how transactions must flow through the system
- AWPROT - Privilege + security level; data or instruction access
- AWQOS – (AXI4 only) Quality of Service ID
- AWREGION – (AXI4 only) Lets 1x slave have multiple logical interfaces
- AWUSER – (AXI4 only) Optional user-defined signal
- AWVALID – Write address valid
- AWREADY – Write address ready

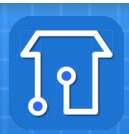
Signalling Details: AXI Read Address

- ARID - The identification tag for the read address group of signals
- ARADDR – Read address
- ARLEN – The exact number of transfers in a burst (AXI3/4 differ)
- ARSIZE - The size of each transfer in the burst
- ARBURST – Burst type
- ARLOCK – Atomic lock characteristics (AXI3/4 differ)
- ARCACHE – Indicates how transactions must flow through the system
- ARPROT - Privilege + security level; data or instruction access
- ARQOS – (AXI4 only) Quality of Service ID
- ARREGION – (AXI4 only) Lets 1x slave have multiple logical interfaces
- ARUSER – (AXI4 only) Optional user-defined signal
- ARVALID – Write address valid
- ARREADY – Write address ready

Signalling Details: AXI Write Data

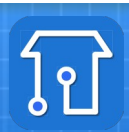
- WID - (AXI3 only) The identification tag for the write data transfer
- WDATA – Write data
- WSTRB - Indicates which byte lanes hold valid data
- WLAST - Indicates the last transfer in a write burst
- WUSER – (AXI4 only) Optional user-defined signal
- WVALID - Valid write data and strobes are available
- WREADY – Slave indicating that it's ready to accept data

- BID – The ID tag of the write response
- BRESP – Indicates the state of the write transaction
- BUSER - (AXI4 only) Optional user-defined signal
- BVALID – Indicates the channel is signalling a valid write response
- BREADY – Indicates the master can accept a write response



Signalling Details: AXI Read Data

- `RID` – The identification tag for the read data transfer
- `RDATA` – Read data
- `RRESP` – Indicates the status of the read transfer
- `RLAST` – Indicates the last transfer in a write burst
- `RUSER` – (AXI4 only) Optional user-defined signal
- `RVALID` – Indicates the channel is signaling the required read data
- `RREADY` – Master indicating that it's ready to accept read data



AXI4 Bus Mechanisms

- Full AXI4 can support:
  - Variable-length bursts, from 1 to 16 data transfers per burst (AxLEN signal)
  - Bursts with a transfer size of 8-1024 bits (AxSIZE signal) – Max 256 in Xililnx parts
  - Wrapping, incrementing, and non-incrementing bursts (AxBURST signal)

AXI4 Bus Mechanisms (cont)

- Atomic operations, using exclusive or locked accesses (AxLOCK signal)
- System-level caching and buffering control (AxCACHE signal)
- Protection information (AxPROT signal)
- Control information on read/write channels are maintained until the corresponding XReady signal is asserted

AXI4 Address Boundary Gotcha

- Data bursts must not cross a 4KB address boundary
- Any multiple of 4096
- That's bit 12: 0b0001_0000_0000_0000

- Valid Example:
  - AxADDR = 4084
    AxSIZE = 4 bytes
    AxLEN = 2
  - Last addr = 4091
  - Safe

- Invalid Example:
  - AxADDR = 4088
    AxSIZE = 4 bytes
    AxLEN = 3
  - Last addr = 4099
  - Addr crossed 4096

AXI4 Read and Read Burst

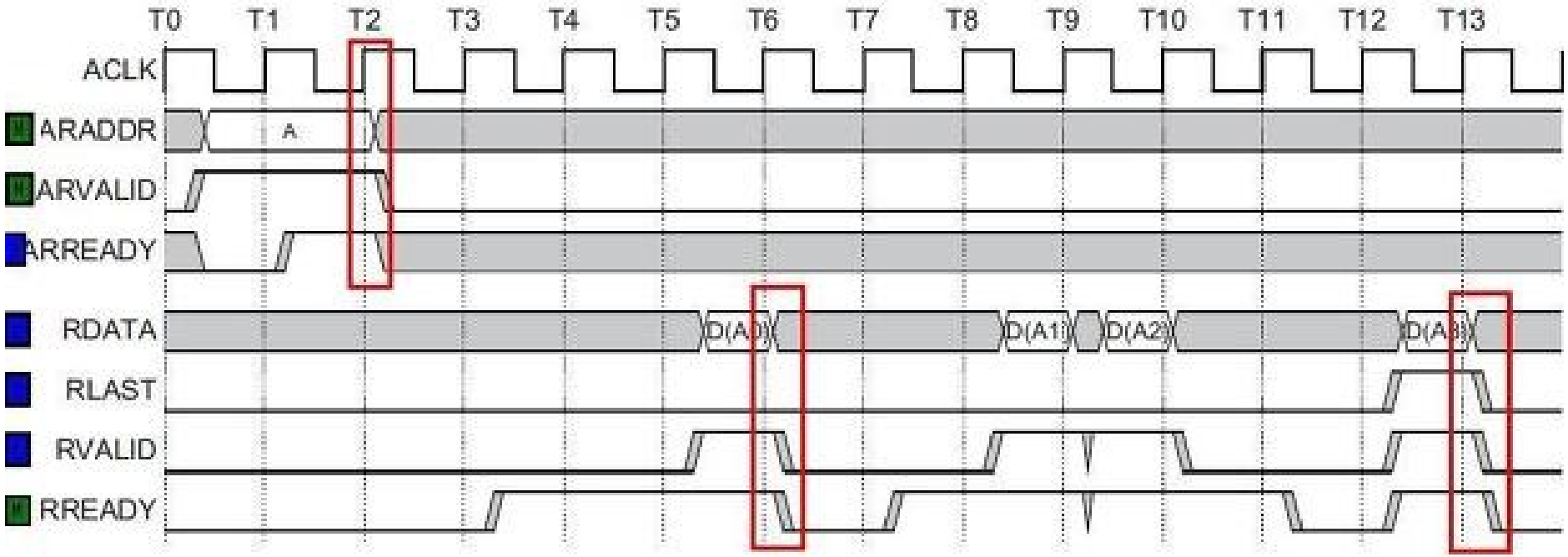


Figure 1-4 Read burst

AXI4 Write and Write Burst

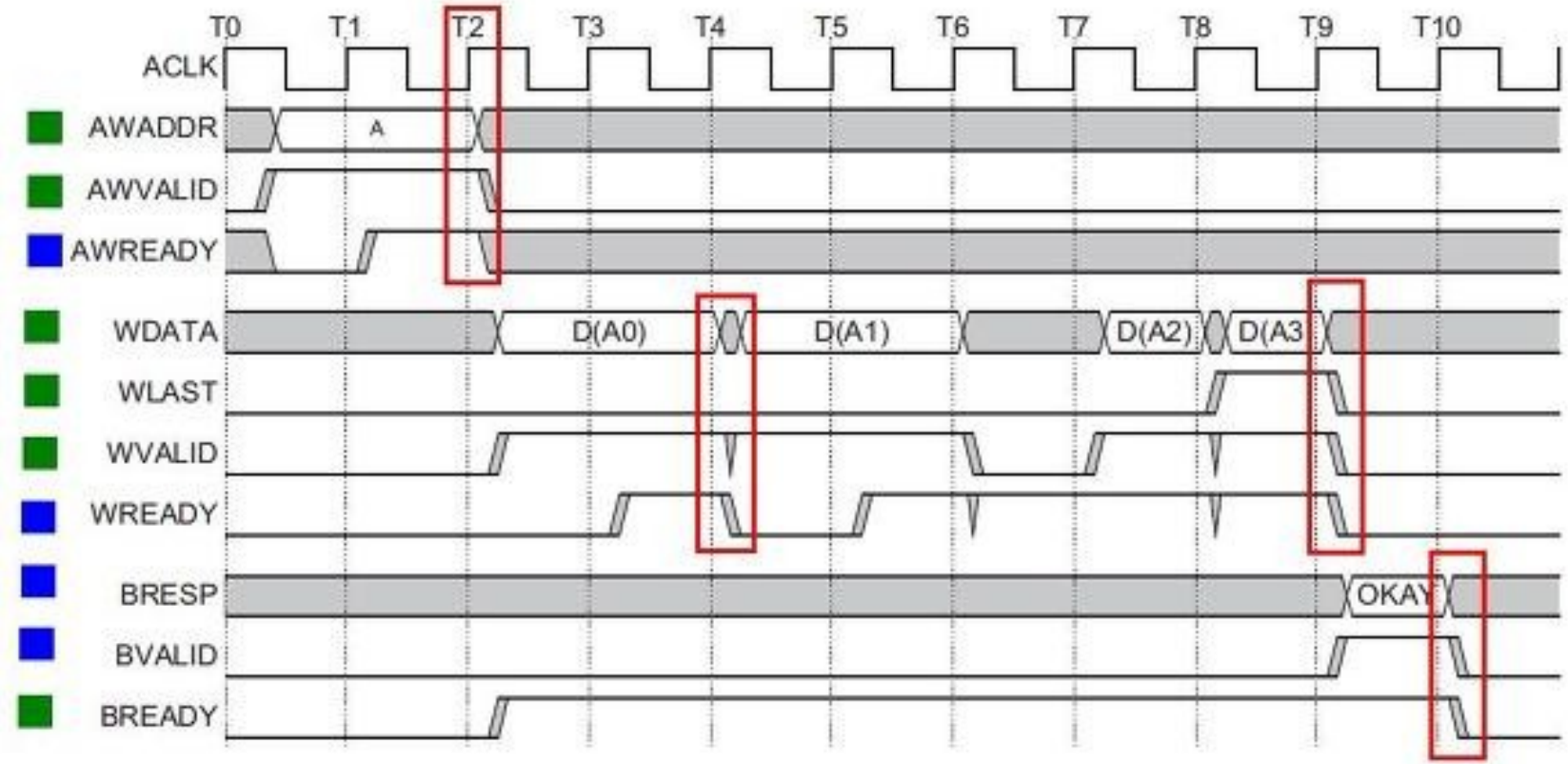


Figure 1-6 Write burst

AXI4 Overlapping Read Bursts

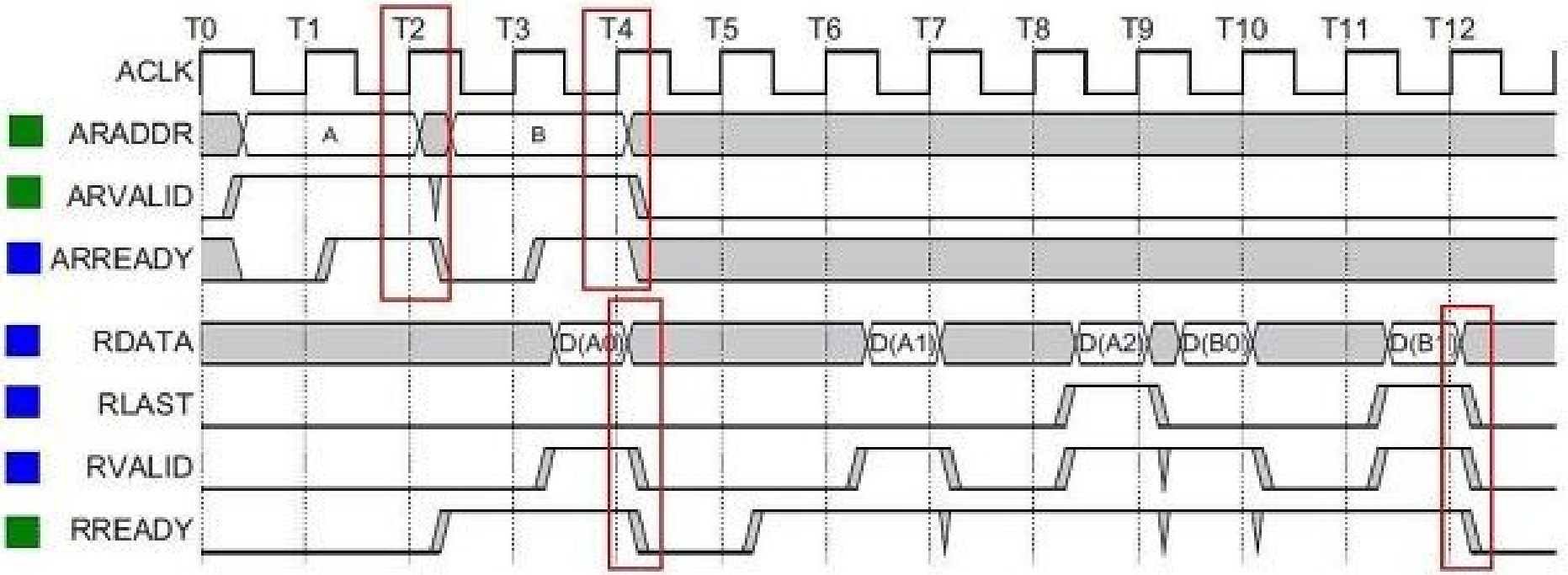


Figure 1-5 Overlapping read bursts

References

- ARM AXI4 Specifcation (ARM IHI 0022D)
- Xilinx UG761 – AXI Reference Guide
- Xilinx UG1037 – Vivado AXI Reference Guide