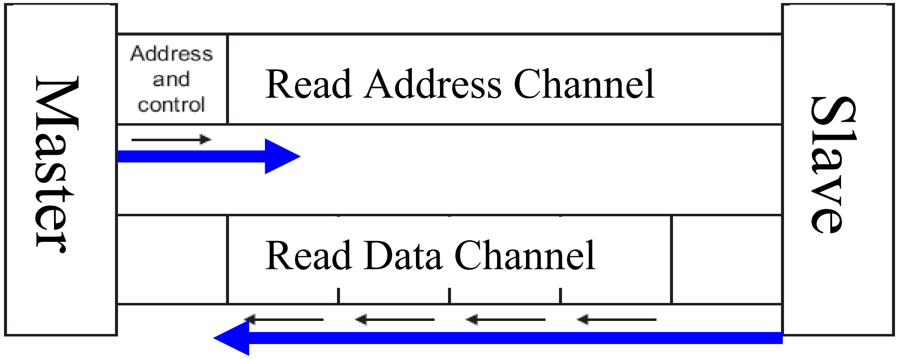
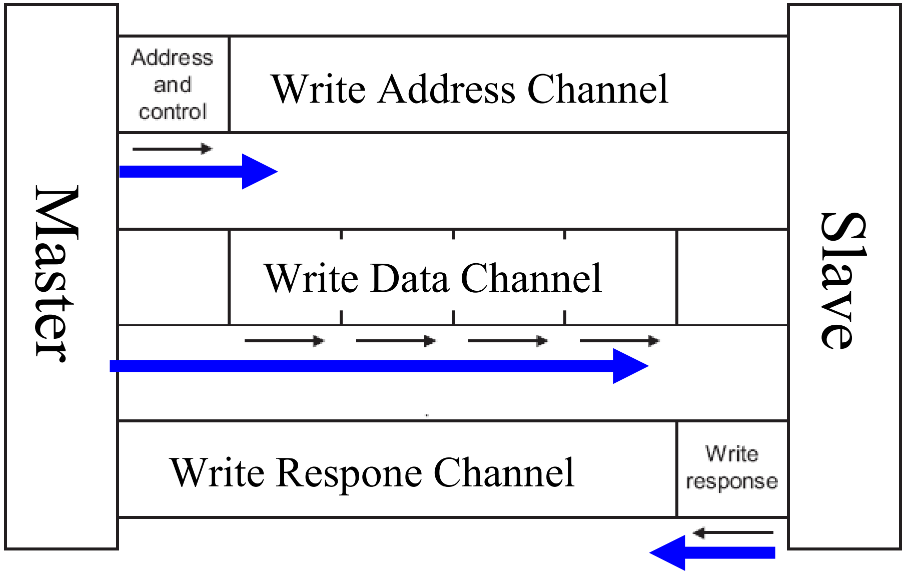
AMBA Specification
Advanced eXtensible Interface Bus
(AXI)

Architecture

- **AXI protocol is Burst-based** transactions with only start address issued**.**
  - **Address/Control is issued ahead of actual data transfer.**
  - **5 channels.**



Read Transaction



Write Transaction

Key Features of AXI Protocol

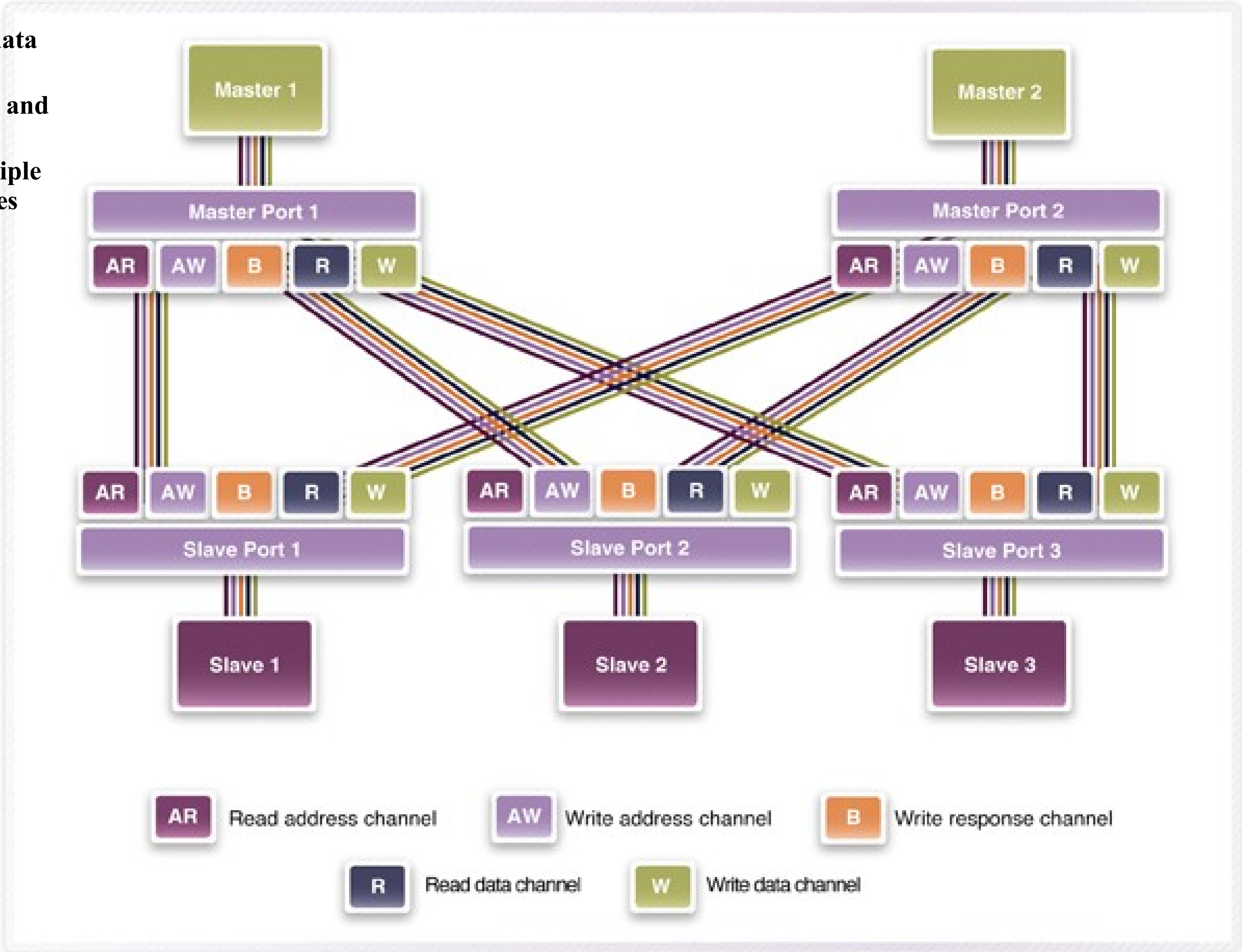
- Separate **address/control** and **data** phases
- Separate **Read** and **Write** data channels
- Support for **unaligned** data transfers using **byte strobes**
  - **Ex:Access a 32-bit data that starts at address 0x80004002**
- **Burst-based** transactions with only start address issued
- Ability to issue multiple outstanding addresses
  - **ID signals**
- Out of order transaction completion
  - **ID signals**
- Easy addition of register stages to provide timing closure
  - **5 channels independent, each channel transfers info. in only one direction.**

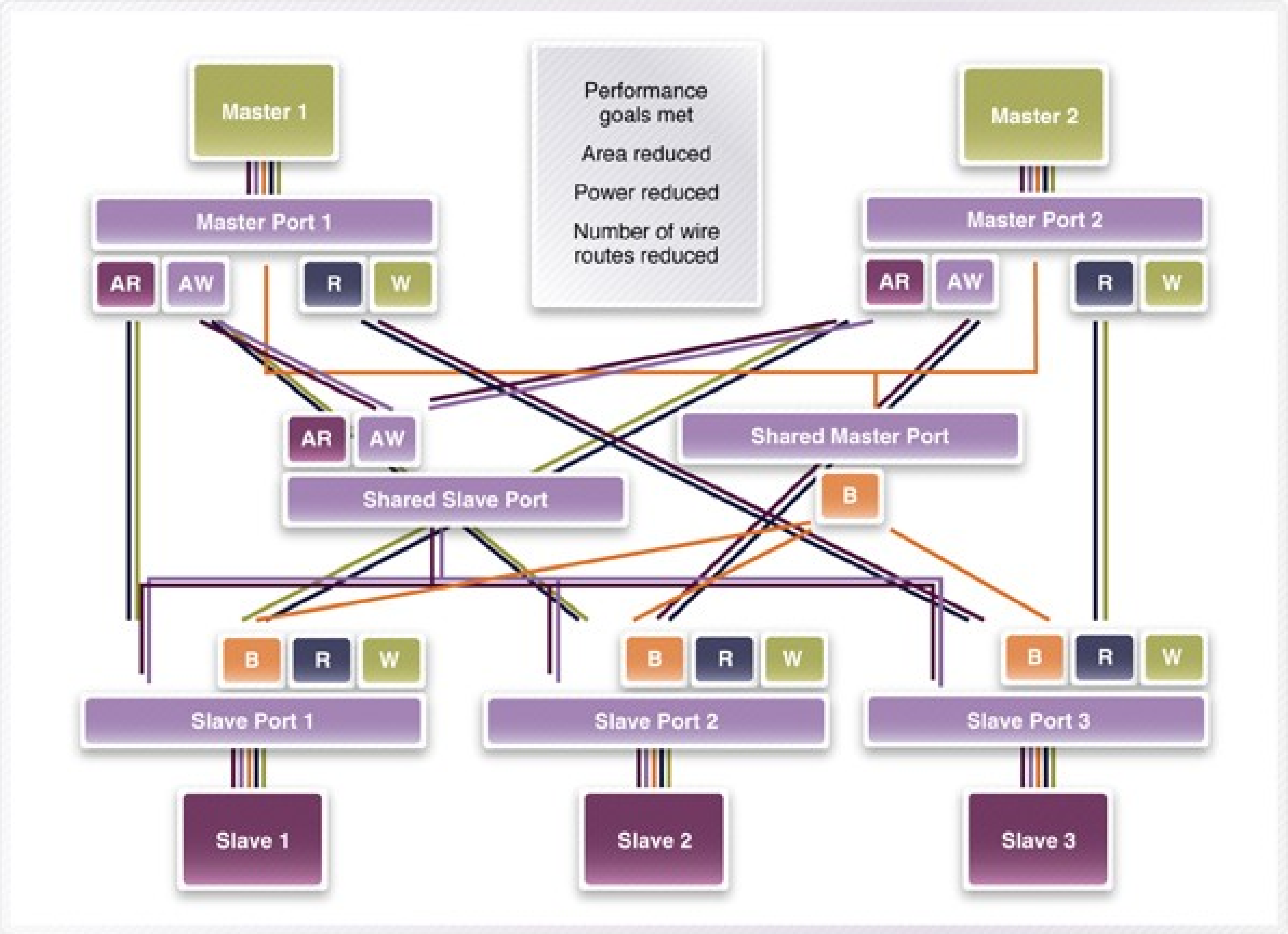
Channel Definition

- Five independent channels which consists of two-way handshake signals
  - VALID, READY
- VALID
  - Asserts when valid data or control information are available on the channel
- READY
  - Asserts when receiver can accept the data
- LAST
  - Asserts while the final data completes

Flexible Interconnect (Basic fabric, a lot of channels)

- **Shared** address and data buses
- **Shared** address buses and multiple data buses
- **Multilayer**, with multiple address and data buses

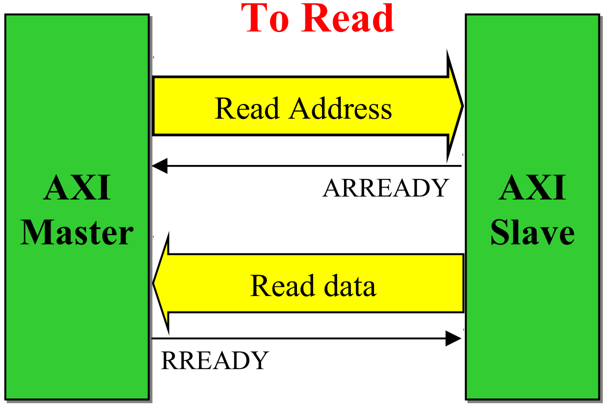


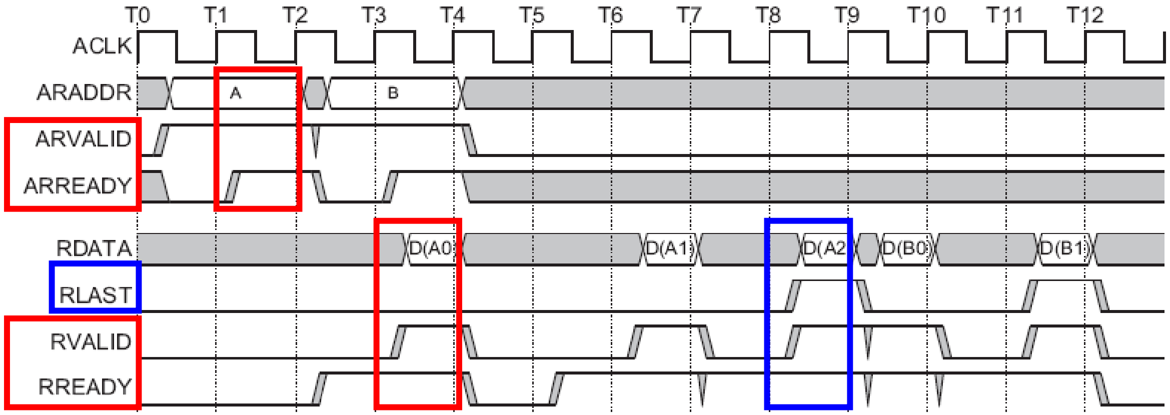


Master 1
Performance goals met
Area reduced
Power reduced
Number of wire routes reduced
Master 2
Master Port 1
Master Port 2
AR
AW
R
W
AR
AW
R
W
AR
AW
Shared Master Port
Shared Slave Port
B
B
R
W
B
R
W
B
R
W
Slave Port 1
Slave Port 2
Slave Port 3
Slave 1
Slave 2
Slave 3

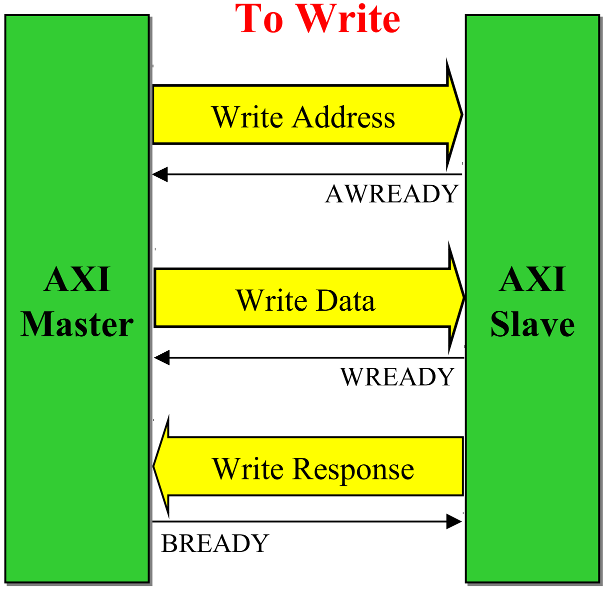
Basic Transactions

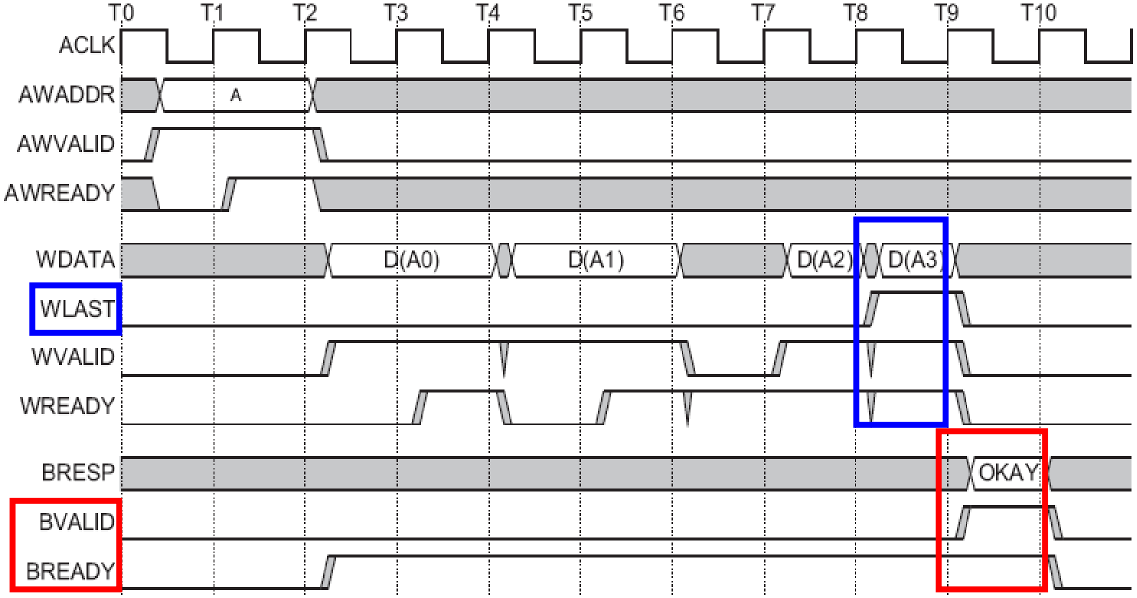
To Read





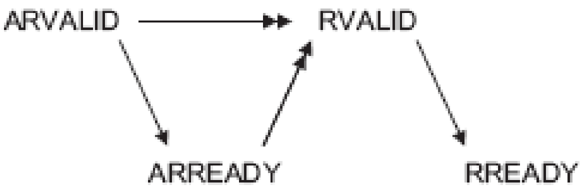
To Write





Transaction Handshake Dependencies

Read



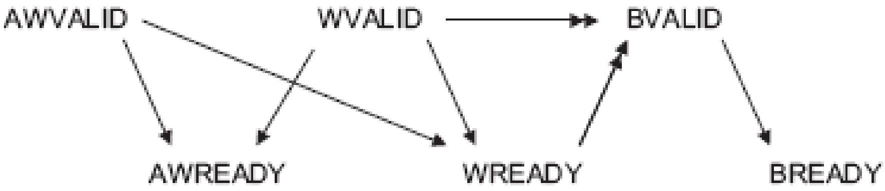
Write data can appear before Write address

Write data appear in the same cycle as the address

Read data always come after Read address

Write response always come after write data

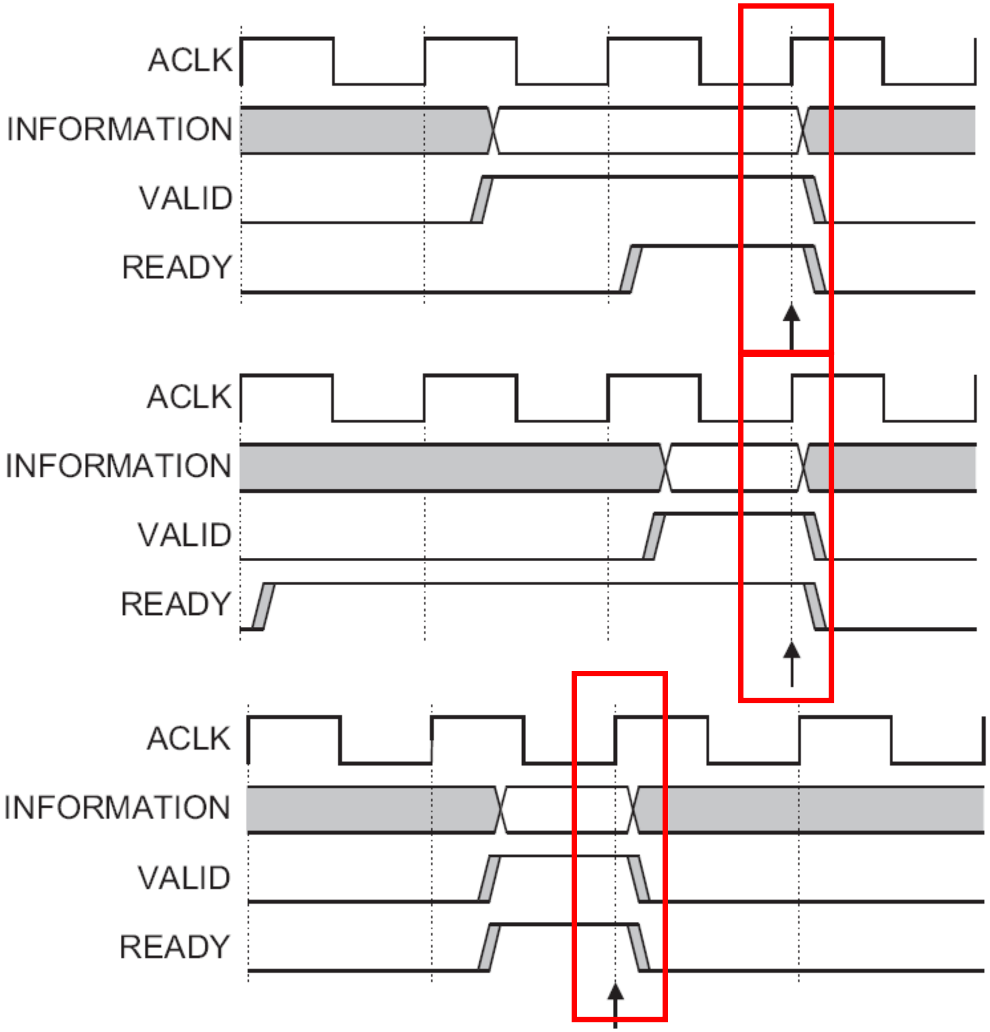
Write



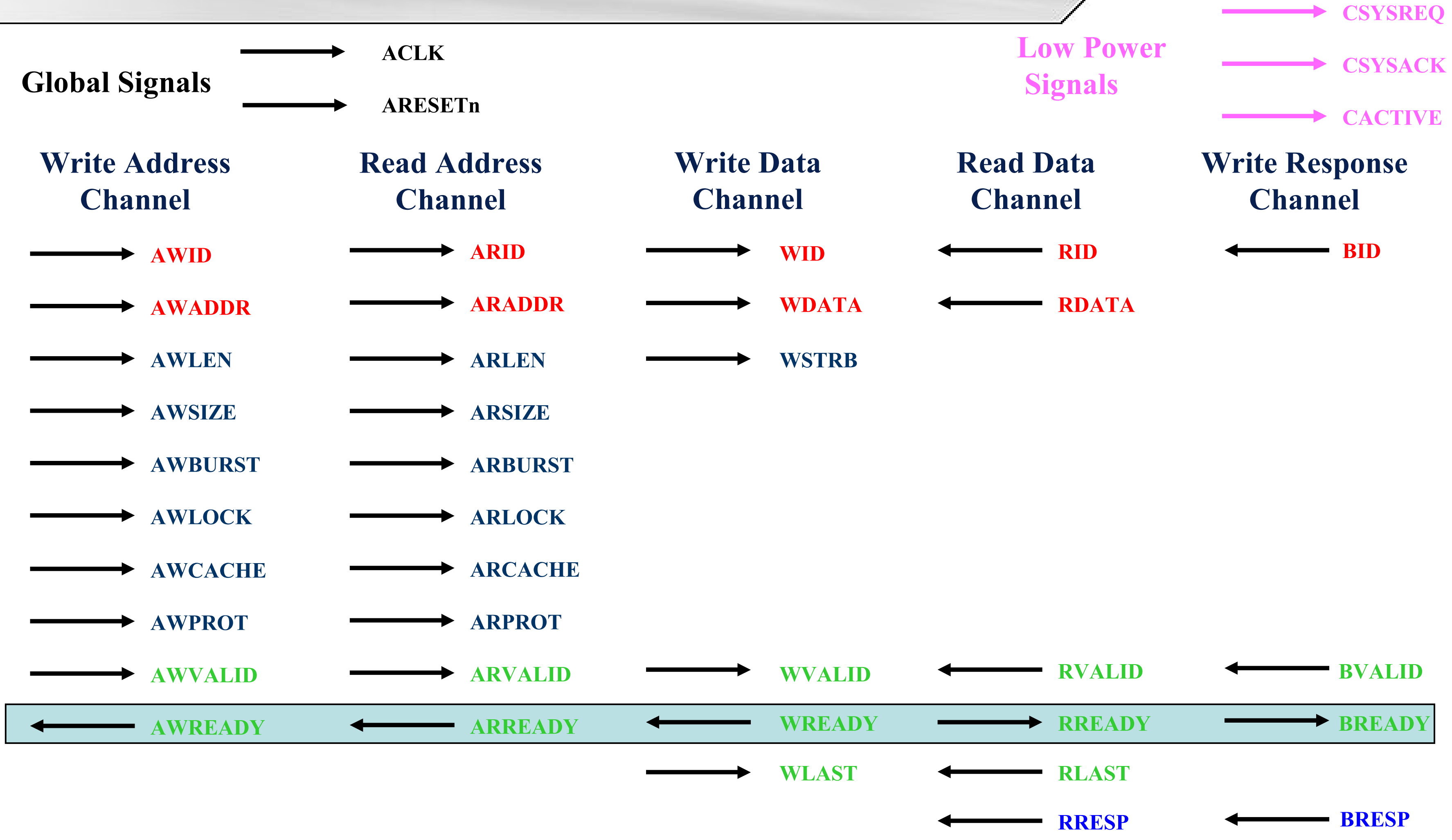
Channel Handshaking

The source presents the data /control_info and drives the VALID signal HIGH.

The data/control_info from the source remains stable until the destination drives the READY signal HIGH, indicating that it accepts the data/control_info.



AXI Signals



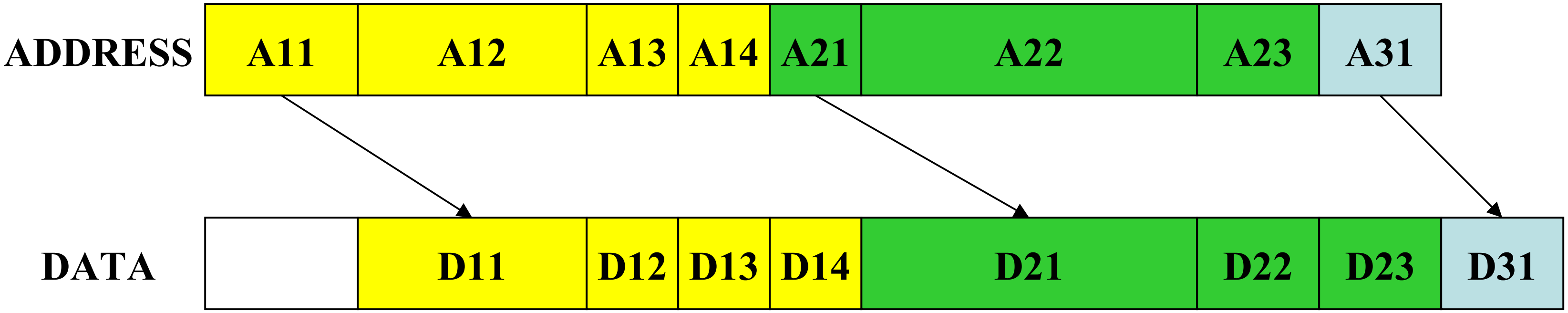
Global Signals
- → ACLK
- → ARESETn

Low Power Signals
- → CSYSREQ
- → CSYSACK
- → CACTIVE

Write Address Channel	Read Address Channel	Write Data Channel	Read Data Channel	Write Response Channel
→ AWID	→ ARID	→ WID	← RID	← BID
→ AWADDR	→ ARADDR	→ WDATA	← RDATA	
→ AWLEN	→ ARLEN	→ WSTRB		
→ AWSIZE	→ ARSIZE			
→ AWBURST	→ ARBURST			
→ AWLOCK	→ ARLOCK			
→ AWCACHE	→ ARCACHE			
→ AWPROT	→ ARPROT			
→ AWVALID	→ ARVALID	→ WVALID	← RVALID	← BVALID
← AWREADY	← ARREADY	← WREADY	→ RREADY	→ BREADY
		→ WLAST	← RLAST	
			← RRESP	← BRESP

AHB/AXI Burst Transfer

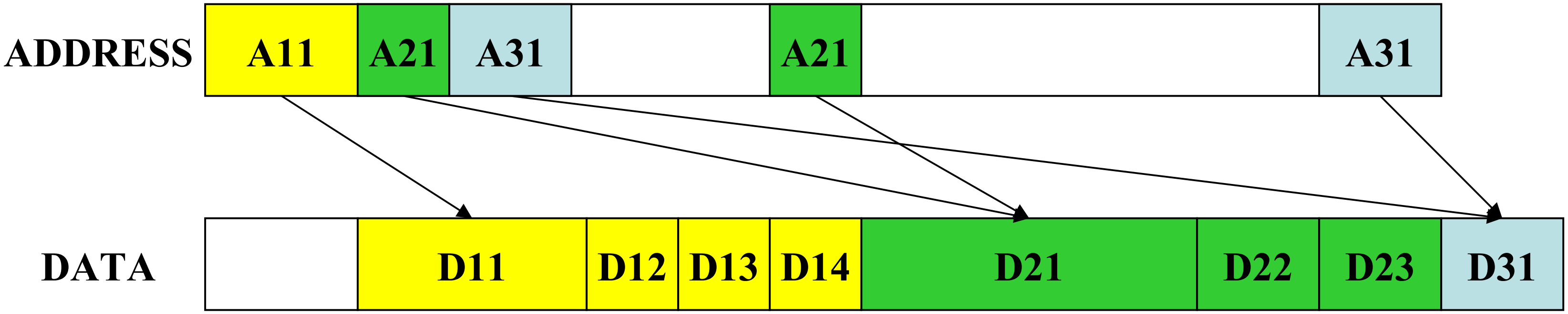
AHB



ADDRESS A11 A12 A13 A14 A21 A22 A23 A31

DATA D11 D12 D13 D14 D21 D22 D23 D31

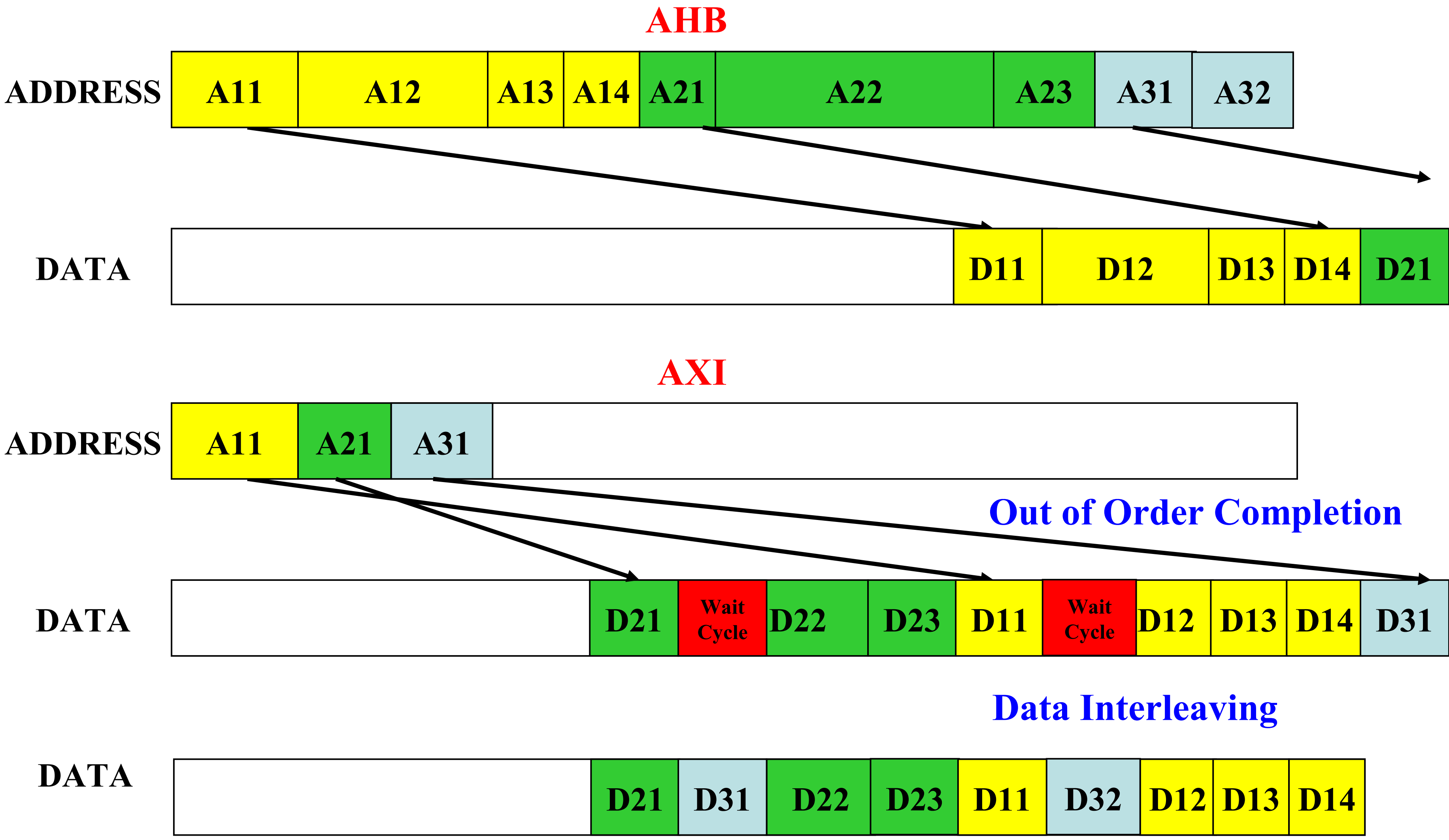
AXI



ADDRESS A11 A21 A31 A21 A31

DATA D11 D12 D13 D14 D21 D22 D23 D31

AHB/AXI Burst Transfer



AHB

ADDRESS A11 A12 A13 A14 A21 A22 A23 A31 A32

DATA D11 D12 D13 D14 D21

AXI

ADDRESS A11 A21 A31

Out of Order Completion

DATA D21 Wait Cycle D22 D23 D11 Wait Cycle D12 D13 D14 D31

Data Interleaving

DATA D21 D31 D22 D23 D11 D32 D12 D13 D14

AXI Ordering Model

Write Address Channel → AWID

Read Address Channel → ARID

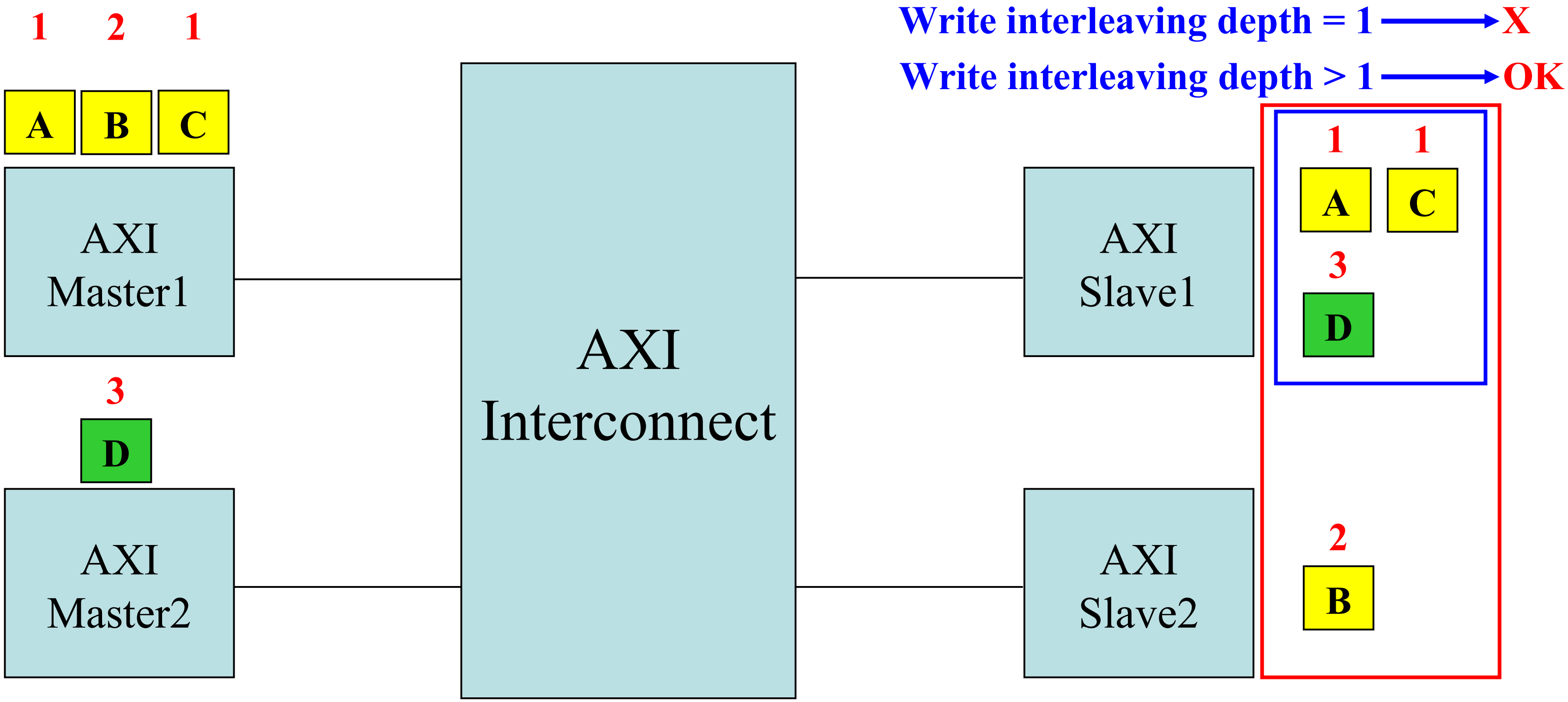
Write Data Channel → WID

Read Data Channel ← RID

Write Response Channel ← BID

1. Out of order completion

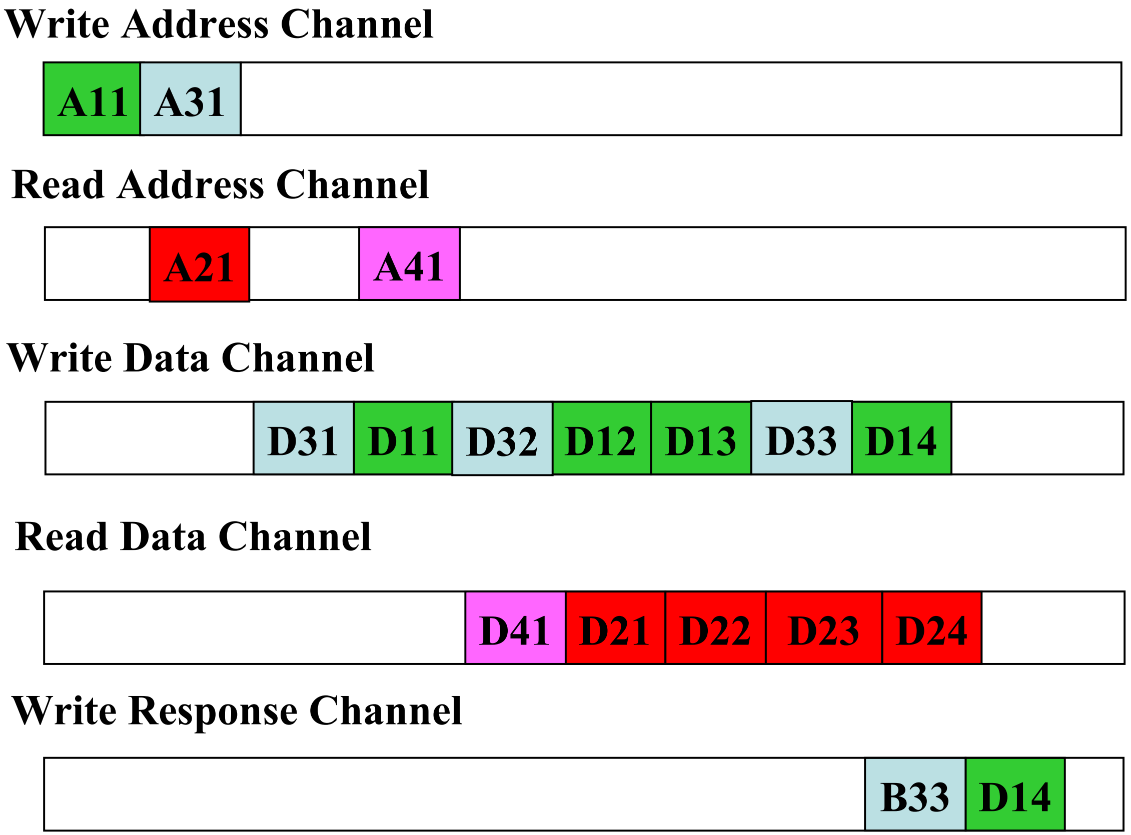
2. Write interleaving



Write interleaving depth = 1 → X

Write interleaving depth > 1 → OK

AXI Transfer Behavior



Write Address Channel

A11 A31

Read Address Channel

A21 A41

Write Data Channel

D31 D11 D32 D12 D13 D33 D14

Read Data Channel

D41 D21 D22 D23 D24

Write Response Channel

B33 D14

Out of Order Completion

- Write data channel
- Read data channel

Data Interleaving

- Write data channel

Register Slices

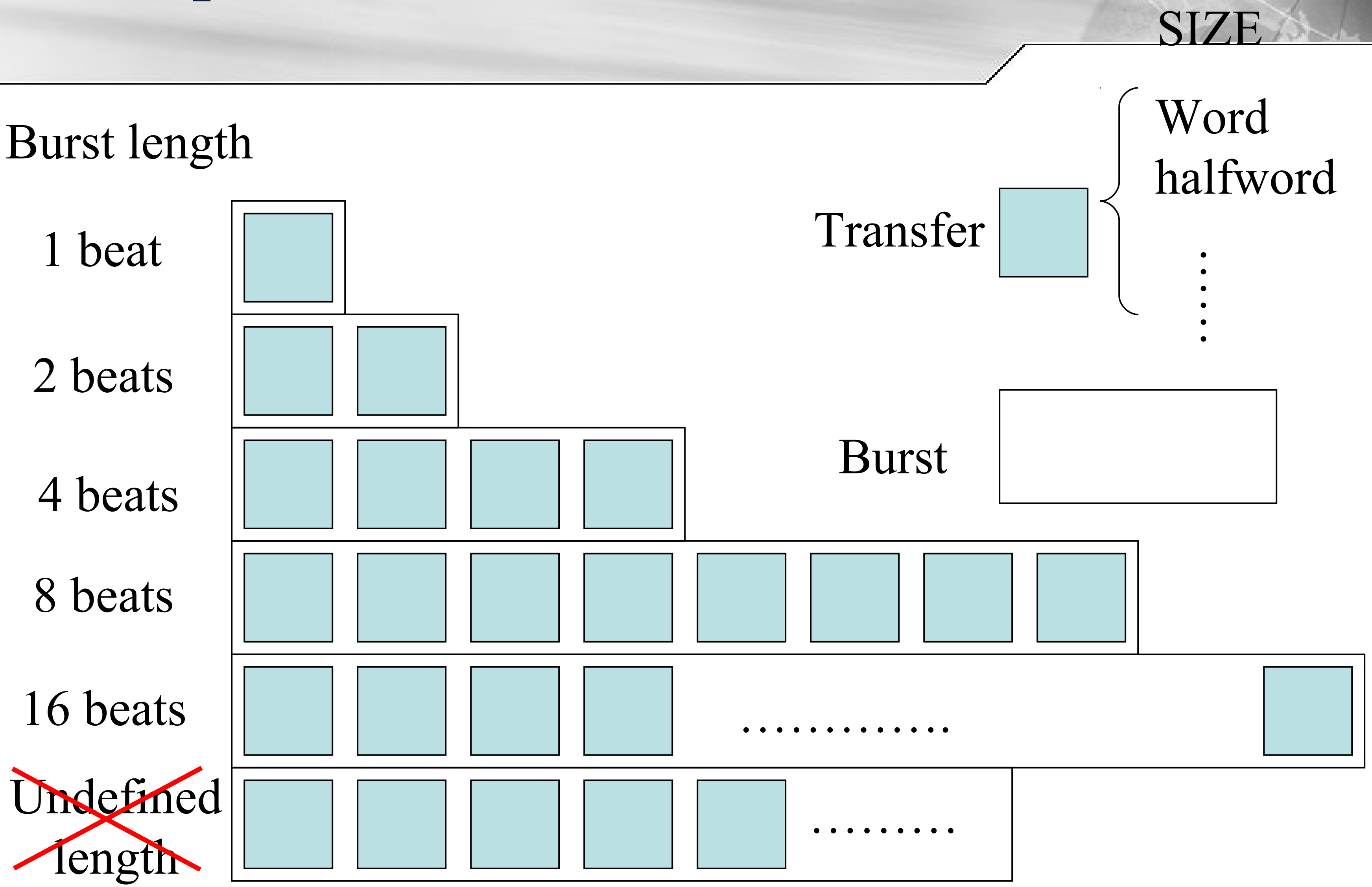
*Matching channel latency for max Freq

*trade-off between cycles of latency and maximum frequency of operation

*Apply to across any channel

- **All channels are unidirectional and independent the order in which AXI channel**
- **Register slices can be inserted across any channel**
  - No change in functionality
  - Increase in latency
- **slices can be inserted at any point in the interconnect**
  - isolate paths to non-performance-critical devices but retain low-latency connections to performance-critical devices

Burst Operation



SIZE

Burst length

Transfer

Word

halfword

⋮

1 beat

2 beats

4 beats

Burst

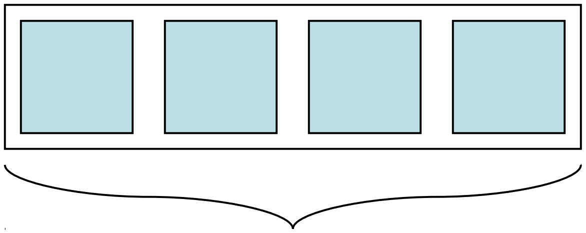
8 beats

16 beats

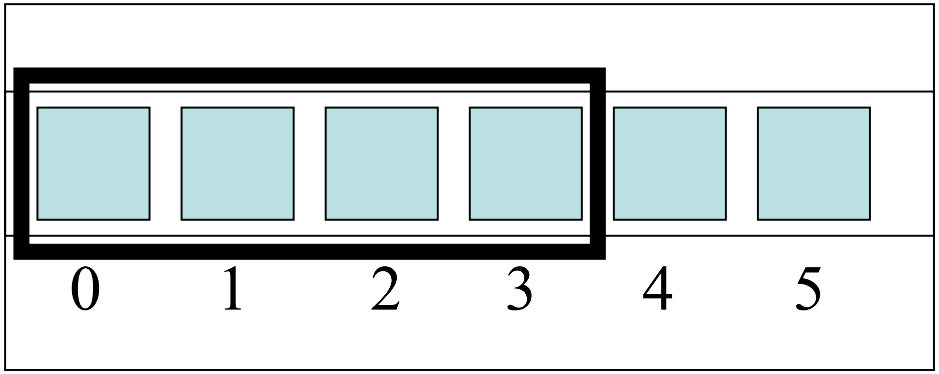
~~Undefined length~~

Burst Transfer Type

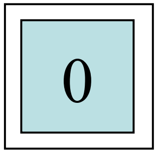
Burst
- beat 4
- Size 1 word



4 x 1word=4 word

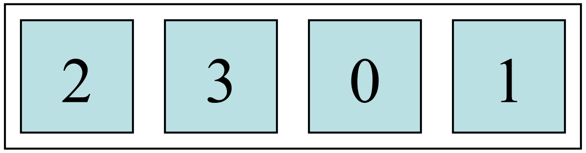


Fixed

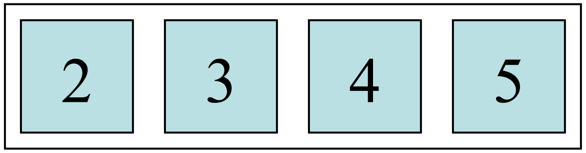


FIFO

Wrap

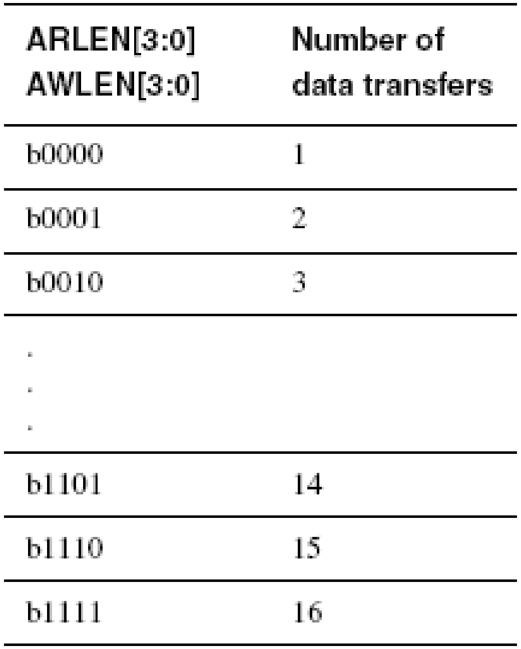


Increment



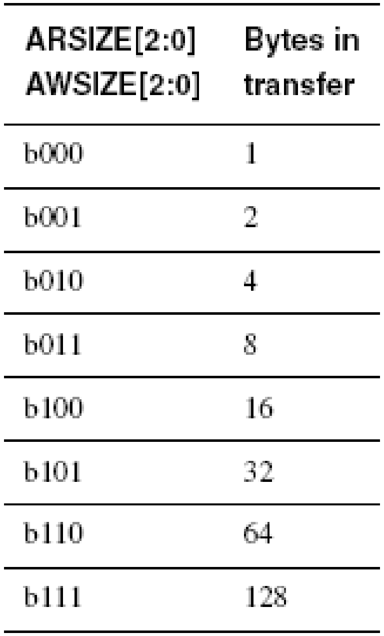
Burst Control signals

Burst length



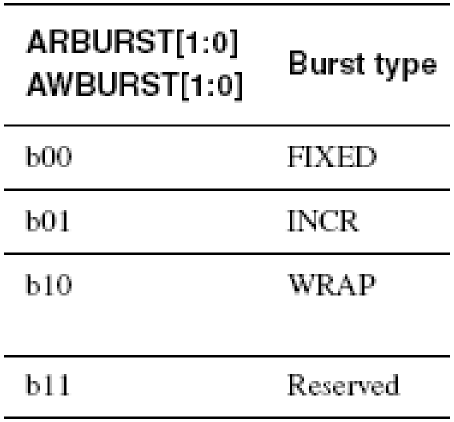
ARLEN[3:0] AWLEN[3:0]	Number of data transfers
b0000	1
b0001	2
b0010	3
. . .	
b1101	14
b1110	15
b1111	16

Burst Size



ARSIZE[2:0] AWSIZE[2:0]	Bytes in transfer
b000	1
b001	2
b010	4
b011	8
b100	16
b101	32
b110	64
b111	128

Burst Type



ARBURST[1:0] AWBURST[1:0]	Burst type
b00	FIXED
b01	INCR
b10	WRAP
b11	Reserved

Every transaction must have the number of transfers

No component can terminate a burst early to reduce the number of data transfers.

Write Strobes

WSTRB : Write strobes. This signal indicates which byte lanes to update in memory. There is one strobe for each eight bits of the write data bus.

WDATA : 8 – 1024 bits wide

WSTRB : 1 – 128 bits wide

64-bit write data bus

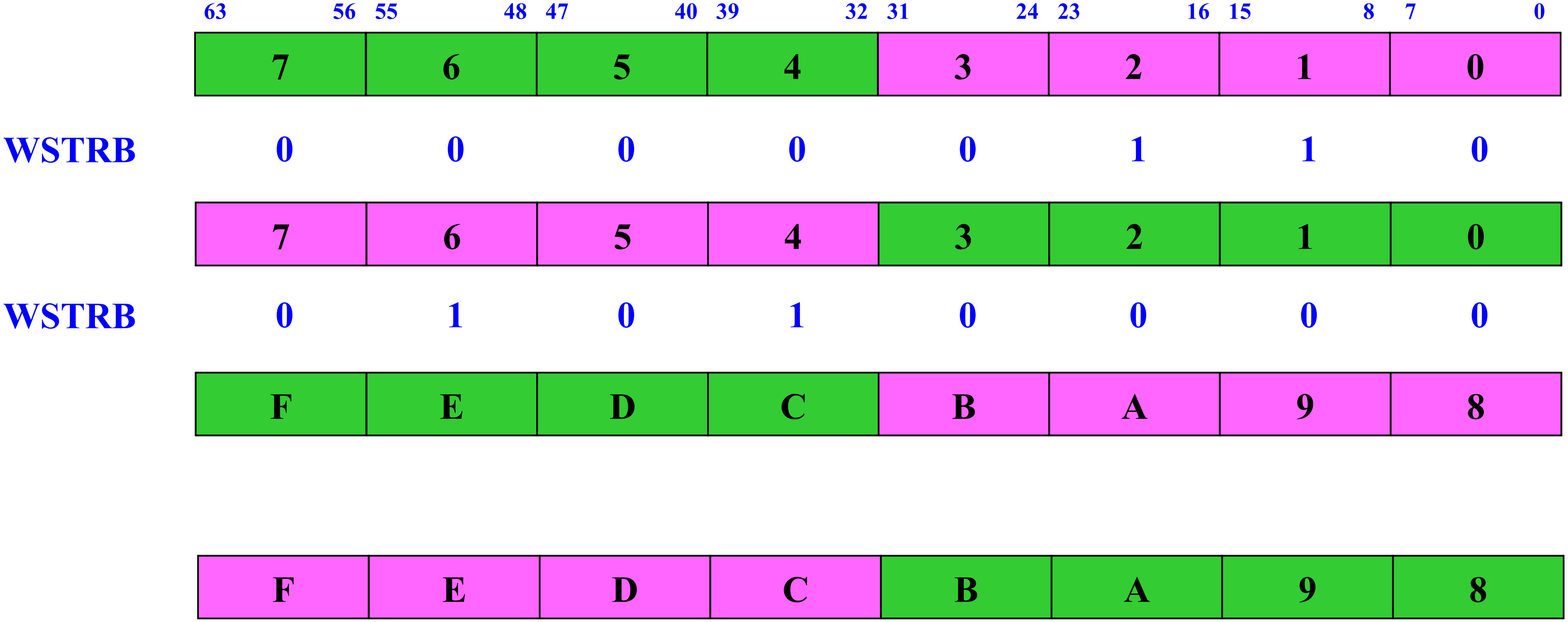
	63 56	55 48	47 40	39 32	31 24	23 16	15 8	7 0
	7	6	5	4	3	2	1	0
WSTRB	1	0	1	0	0	1	0	0

Aligned Transfer

Address: 0x00
Transfer size: 32 bits
Burst type: incrementing
Burst length: 4 transfers

For wrapping burst type, all transfers are aligned transfers

64-bit write data bus



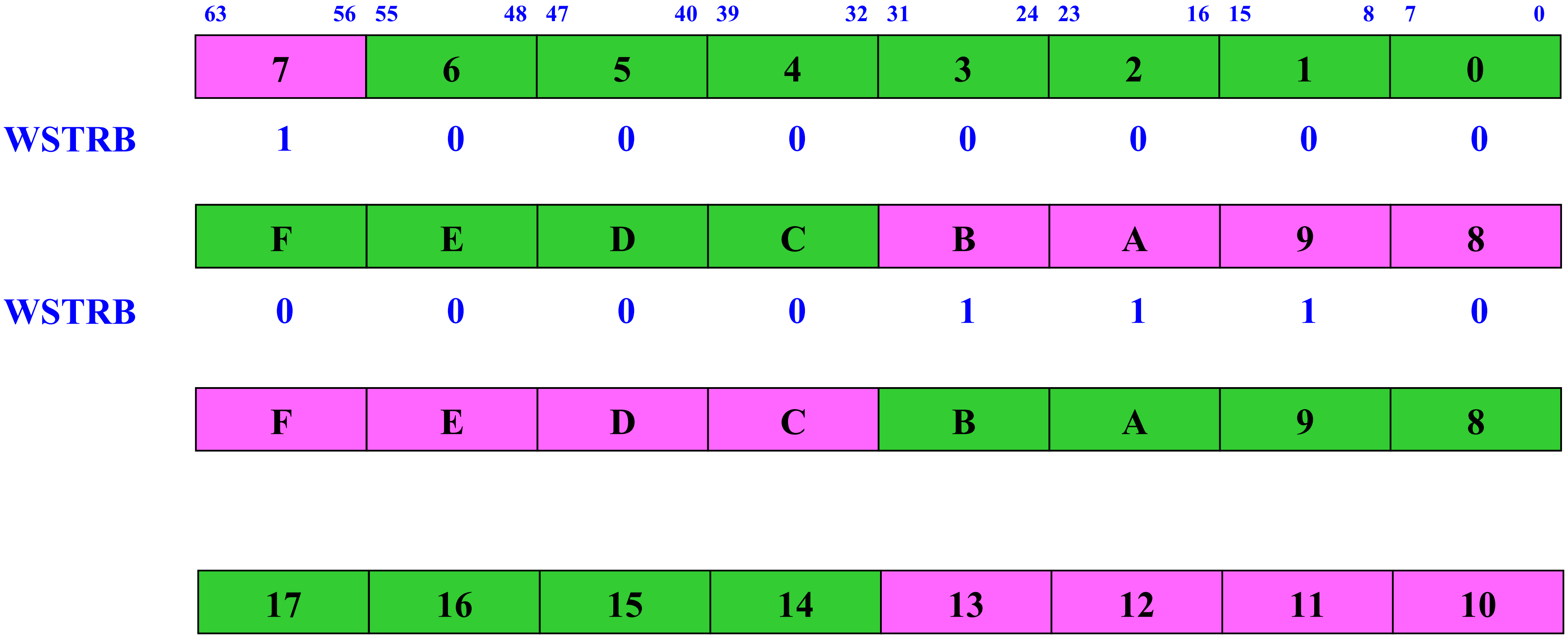
	63 56	55 48	47 40	39 32	31 24	23 16	15 8	7 0
	7	6	5	4	3	2	1	0
WSTRB	0	0	0	0	0	1	1	0
	7	6	5	4	3	2	1	0
WSTRB	0	1	0	1	0	0	0	0
	F	E	D	C	B	A	9	8
	F	E	D	C	B	A	9	8

Unaligned Transfer

Address: 0x07
Transfer size: 32 bits
Burst type: incrementing
Burst length: 4 transfers

For incrementing burst type, fist transfer can be unaligned transfers, but the rest transfers are aligned transfers

64-bit write data bus



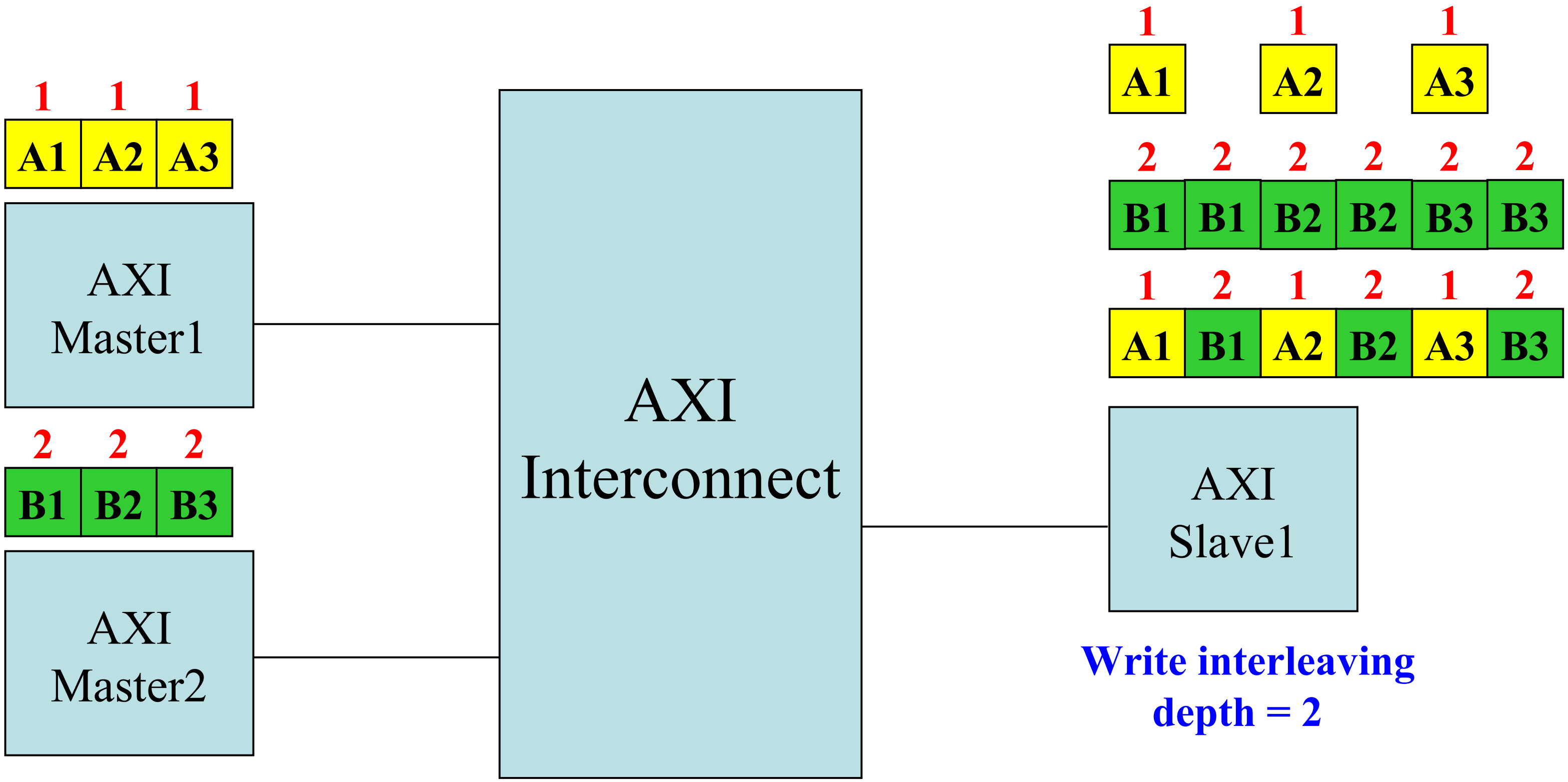
	63 56	55 48	47 40	39 32	31 24	23 16	15 8	7 0
	7	6	5	4	3	2	1	0
WSTRB	1	0	0	0	0	0	0	0
	F	E	D	C	B	A	9	8
WSTRB	0	0	0	0	1	1	1	0
	F	E	D	C	B	A	9	8
	17	16	15	14	13	12	11	10

Cache Support

- Support for system level caches and other performance enhancing components
  - Bufferable (B) bit, ARCACHE[0] and AWCACHE[0]
  - Cacheable (C) bit, ARCACHE[1] and AWCACHE[1]
  - Read Allocate (RA) bit, ARCACHE[2] and AWCACHE[2]
  - Write Allocate (WA) bit, ARCACHE[3] and AWCACHE[3]

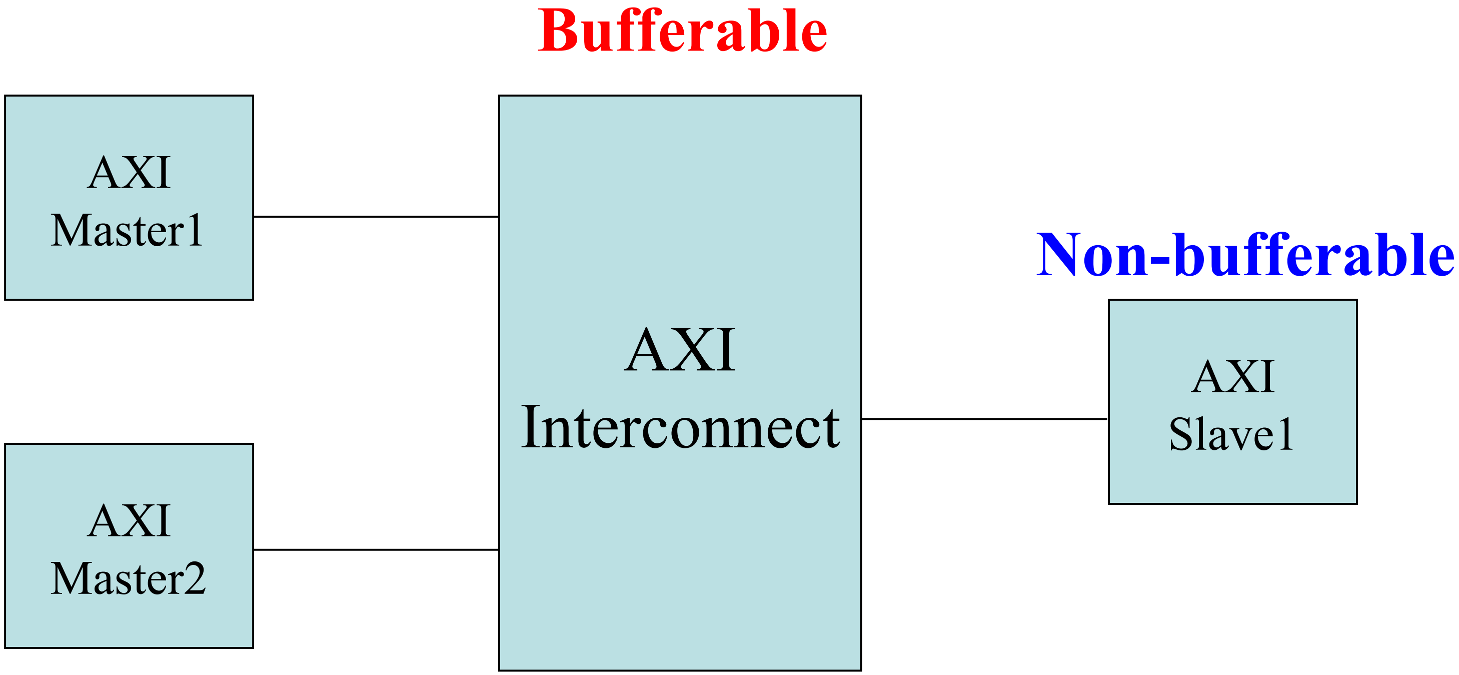
Bufferable Bit

- The interconnect or any component can delay the transaction for an arbitrary number of cycles, usually only relevant to writes



Bufferable Bit (Conti.)

- IF a transaction is bufferable
  - It is acceptable for a bridge or system level cache to provide write response
- If non-bufferable
  - Final destination to provide response



Cache Support

- Cacheable Bit
  - Write : a number of different writes can be merged together
  - Read : a location can be pre-fetched or can be fetched just once for multiple read transactions
- Read Allocate Bit
  - If the transfer is a read and it misses in the cache then it should be allocated
- Write Allocate Bit
  - If the transfer is a write and it misses in the cache then it should be allocated

Cacheable Write – Sparse Strobes

Cacheable Bit

Write : a number of different writes can be merged together

ARM11-MPCore processor and L220 Level 2 cache controller use sparse strobes. This is due to the fact that they use a merging write buffer. If the application writes several bytes at address 0x0, 0x3, 0x4, the write buffer will be drained using a 64-bit transfer, and strobes will be 0b00011001 and the AXI slave must only update the bytes that are enabled.

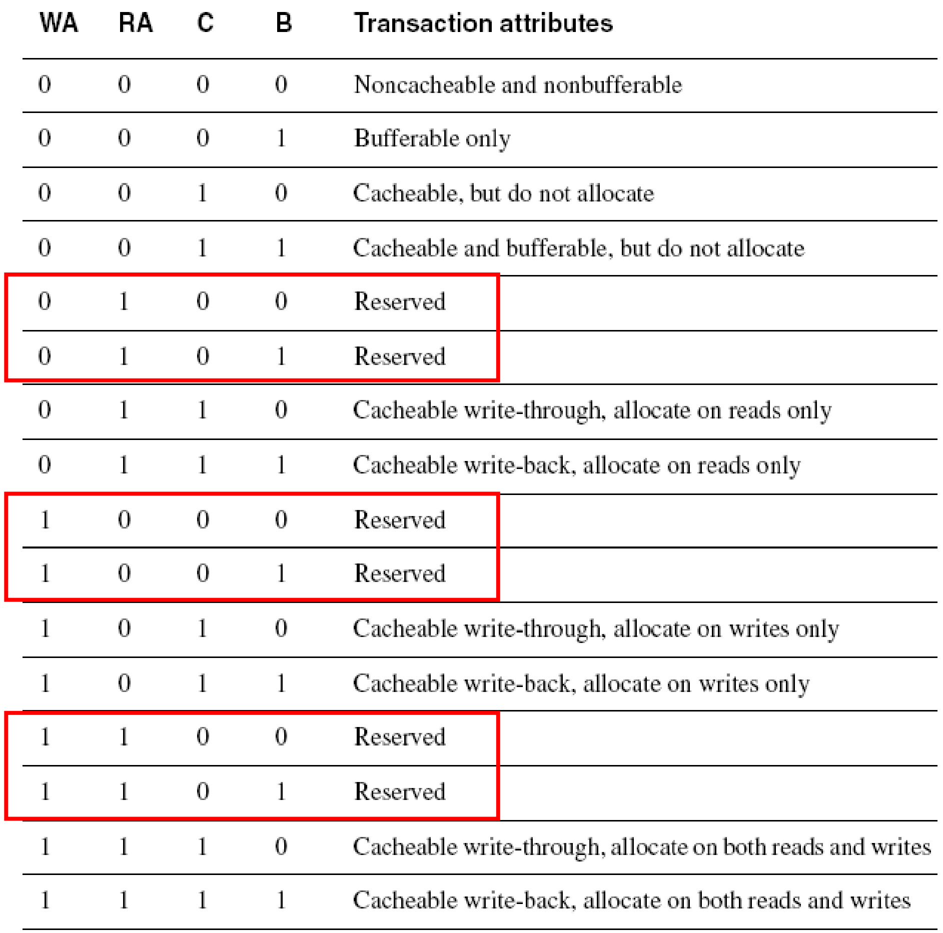
64-bit write data bus

	63 56	55 48	47 40	39 32	31 24	23 16	15 8	7 0
	7	6	5	4	3	2	1	0
WSTRB	0	0	0	1	1	0	0	1

Cache Encoding

ARCACHE[3:0]

AWCACHE[3:0]



WA	RA	C	B	Transaction attributes
0	0	0	0	Noncacheable and nonbufferable
0	0	0	1	Bufferable only
0	0	1	0	Cacheable, but do not allocate
0	0	1	1	Cacheable and bufferable, but do not allocate
0	1	0	0	Reserved
0	1	0	1	Reserved
0	1	1	0	Cacheable write-through, allocate on reads only
0	1	1	1	Cacheable write-back, allocate on reads only
1	0	0	0	Reserved
1	0	0	1	Reserved
1	0	1	0	Cacheable write-through, allocate on writes only
1	0	1	1	Cacheable write-back, allocate on writes only
1	1	0	0	Reserved
1	1	0	1	Reserved
1	1	1	0	Cacheable write-through, allocate on both reads and writes
1	1	1	1	Cacheable write-back, allocate on both reads and writes

C : low

RA : low

WA : low

Protection Unit Support

- To support complex system design, for the interconnect and other devices in the system to provide protection against illegal transactions

  - Normal or privileged, ARPROT[0] and AWPROT[0]

    Low High

  - Secure or non-secure, ARPROT[1] and AWPROT[1]

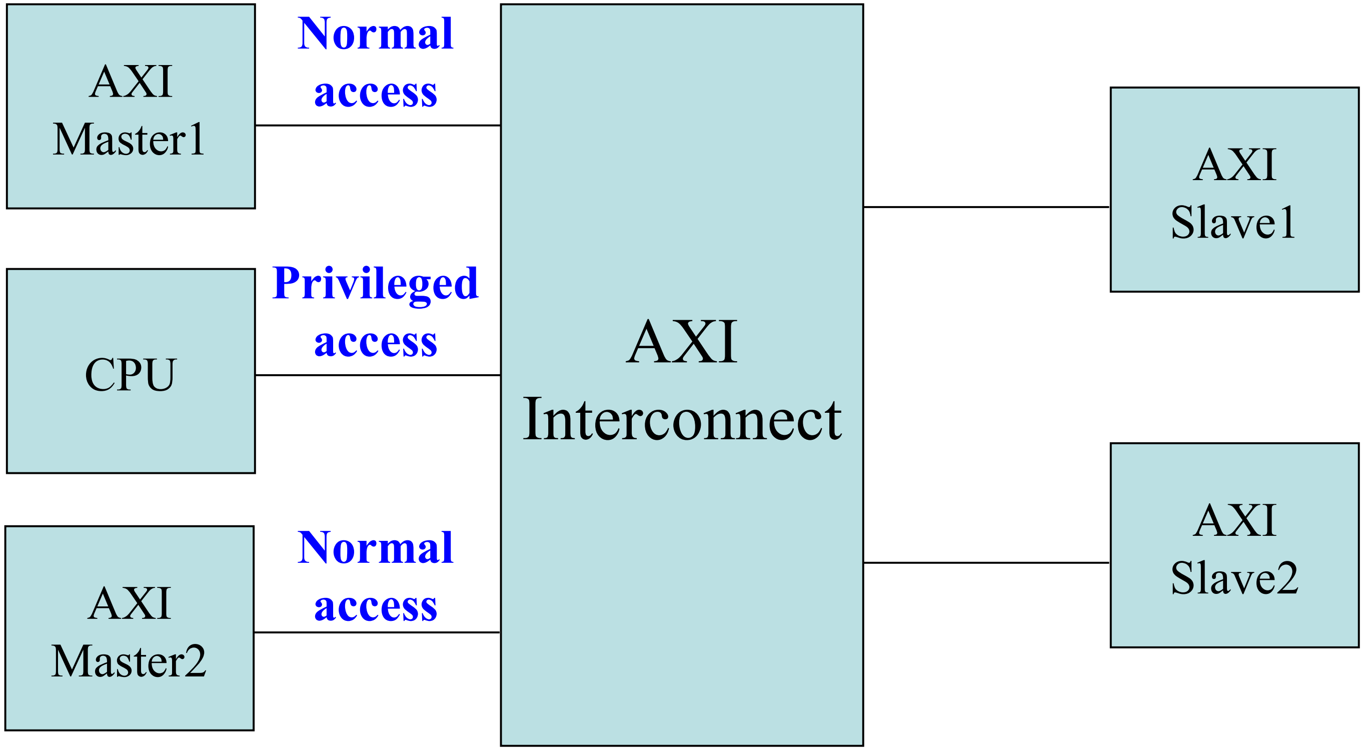
    Low High

  - Data or instruction, ARPROT[2] and AWPROT[2]

    Low High

Normal / Privileged

- This is used by some **masters** to indicate their processing mode. A privileged processing mode typically has a **greater lever of access** within a system

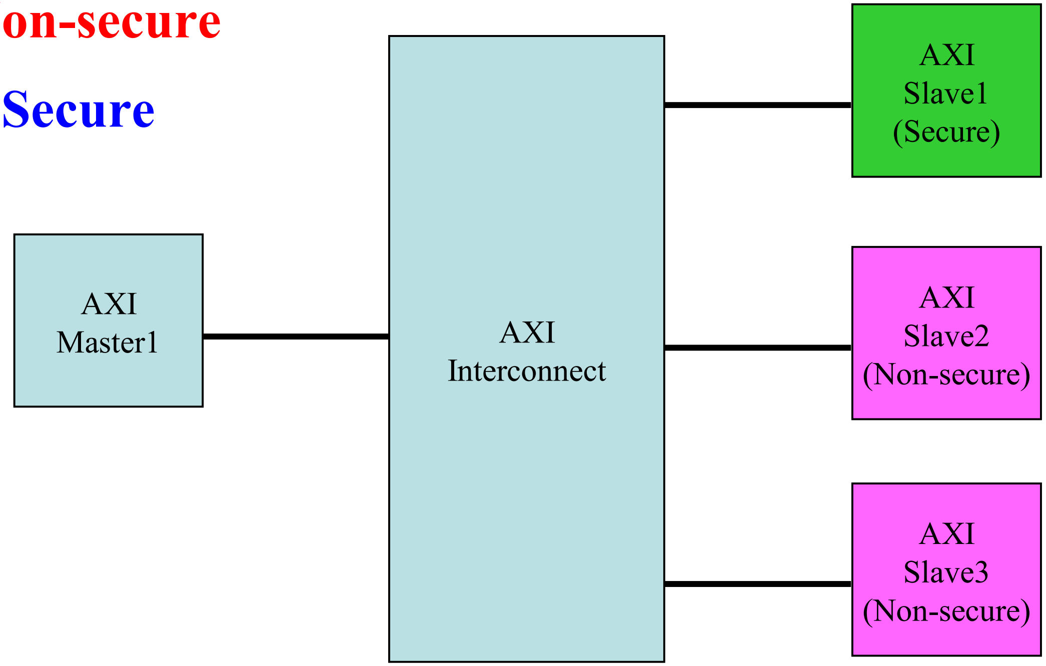


Secure / Non-secure

- This is used in systems where a **greater degree of differentiation** between processing modes is required

Non-secure

Secure



SLVERR response

OKAY response

DECERR response

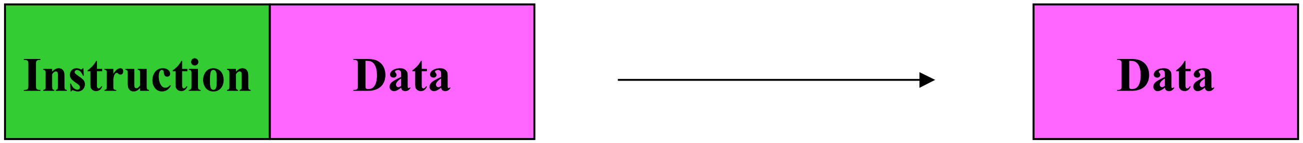
Non-secure slave disappears from the memory map during secure accesses

Data / Instruction

- This bit gives an indication if the transaction is an instruction or a data access

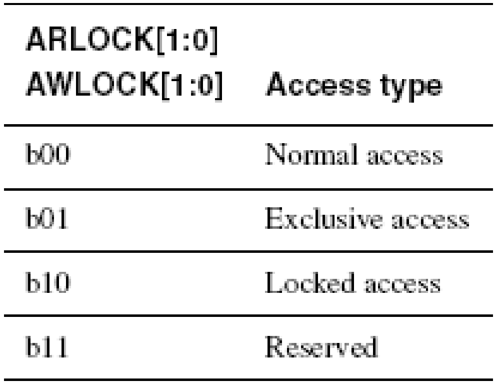
When a transaction contains a mix of instruction and data items

It’s recommended that, by default, an access is marked as a data access unless specifically known to be an instruction access



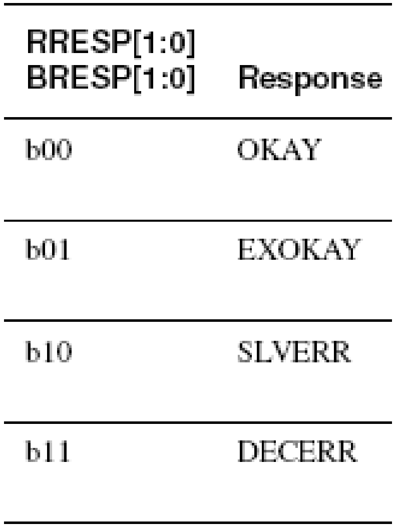
Access / Response Signals

Access Signals



ARLOCK[1:0] AWLOCK[1:0]	Access type
b00	Normal access
b01	Exclusive access
b10	Locked access
b11	Reserved

Response Signals



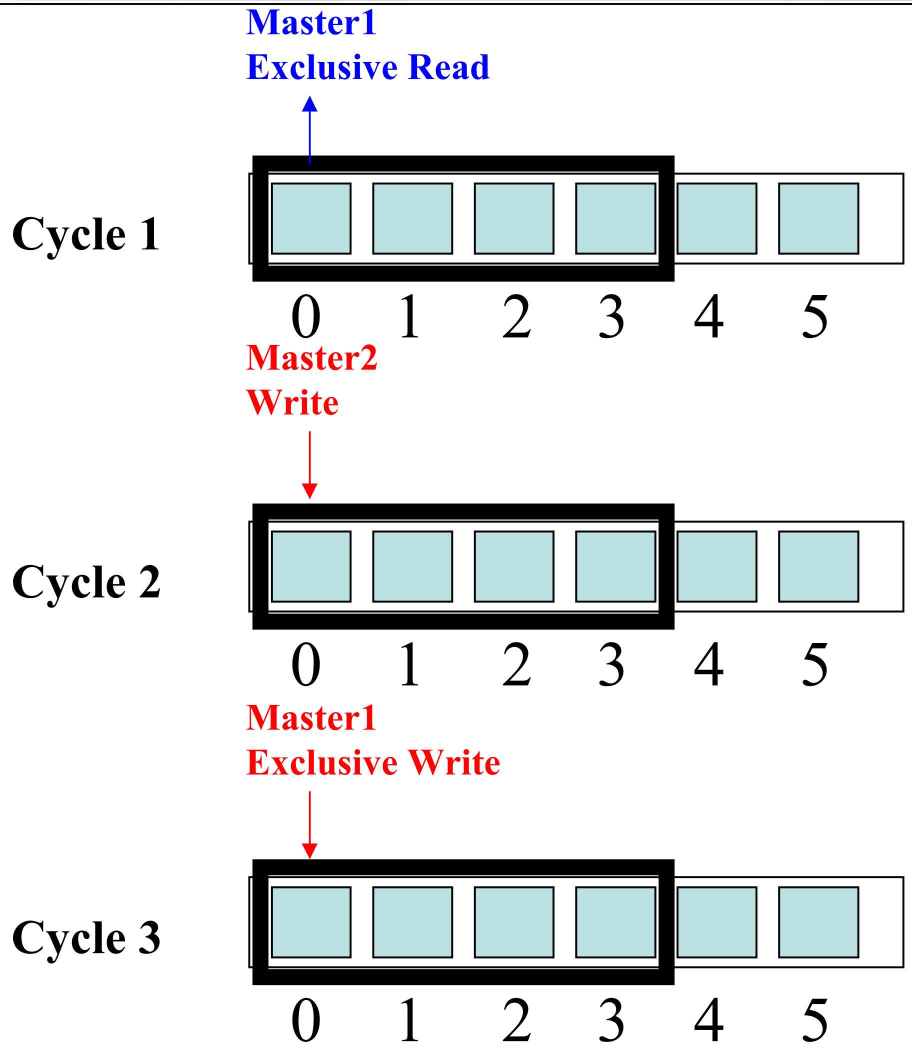
RRESP[1:0] BRESP[1:0]	Response
b00	OKAY
b01	EXOKAY
b10	SLVERR
b11	DECERR

Slave-generated errors

Address decode errors

Exclusive Access

Semaphore type without locking the bus access



Exclusive Read success
Response : EXOKAY

Exclusive Read fail
Response : OKAY

Exclusive Write success
Response : EXOKAY

Exclusive Write fail
Response : OKAY

Low-power interface signals

Signal	Source	Description
CSYSREQ	Clock controller	System low-power request. This signal is a request from the system clock controller for the peripheral to enter a low-power state.
CSYSACK	Peripheral device	Low-power request acknowledgement. This signal is the acknowledgement from a peripheral of a system low-power request.
CACTIVE	Peripheral device	Clock active. This signal indicates that the peripheral requires its clock signal: 1 = peripheral clock required 0 = peripheral clock not required.

CSYSREQ and CSYSACK handshake

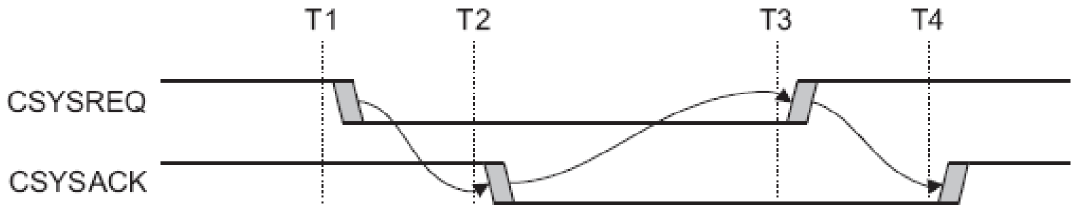
CSYSREQ

To request that the peripheral enter a low-power state, the system clock controller drives the **CSYSREQ** signal LOW. During normal

operation, **CSYSREQ** is HIGH.

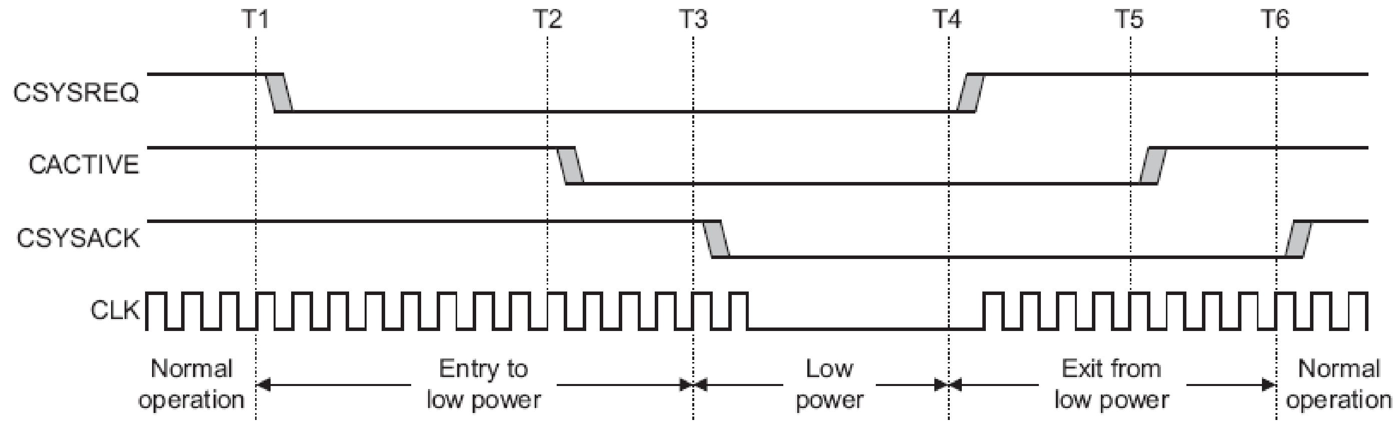
CSYSACK

The peripheral uses the **CSYSACK** signal to acknowledge both the low-power state request and the exit from the low-power state.

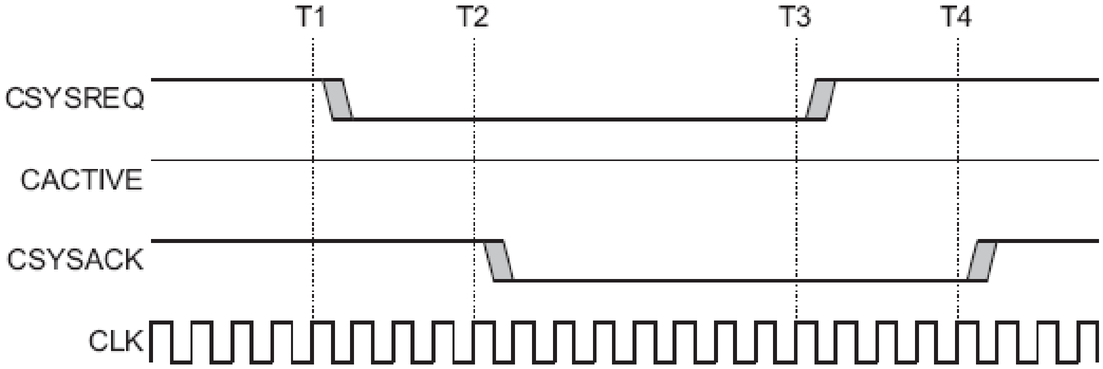


Acceptance/ Denial of low-power request

Acceptance



Denial



Advantages of AXI Protocol

- Independently acknowledged address and data channels
- Out-of-order completion of bursts
- Write data interleaving
- Exclusive access (atomic transaction)
- Access security support
- System level cache support
- Unaligned address & byte strobe
- Static burst, which allows bursts to FIFO memory
- Low power mode