On-Chip Bus for SoC

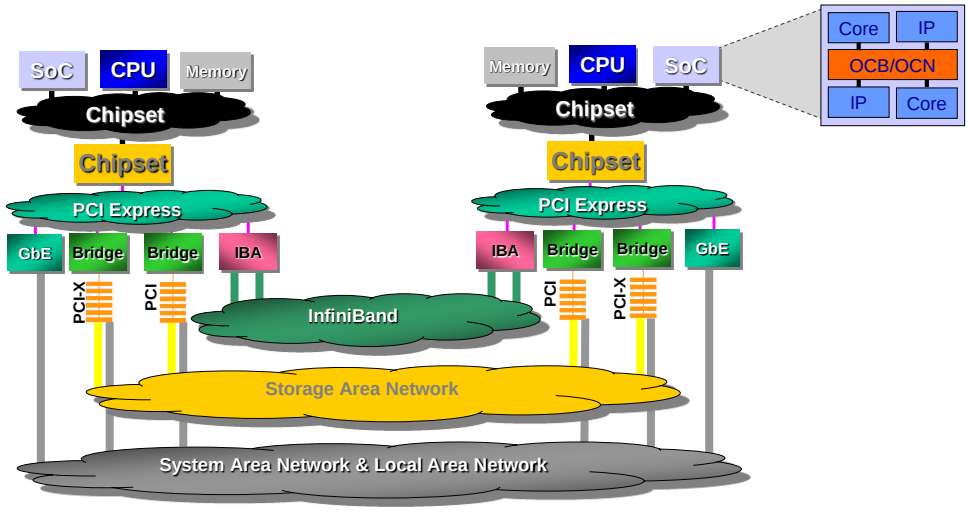
2013 - 2017

Ando Ki, Ph.D.
(adki@future-ds.com)

Agenda

- Where it is about
- What is on-chip bus
- Standard and/or de factor standard bus n ot for SoC
- Interconnect landscape not for SoC
- De factor standard buses for SoC
- ARM AMBA
- IBM CoreConnect
- Altera Avalon
- OpenCores Wishbone

Where it is about

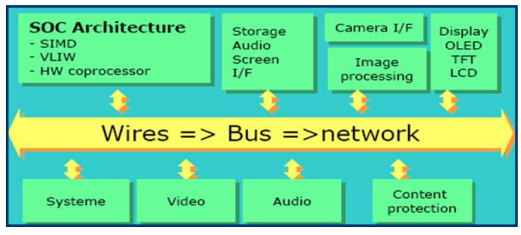


OCB: On-Chip-Bus; OCN: On-Chip-Network

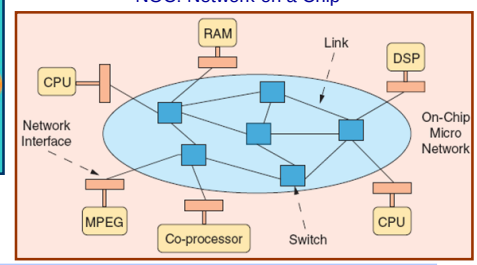
What is on-chip bus

- **Bus** means a set of common lines that electrically (or optically) connects various units (circuits) in order to transfer the data among them.
  - Protocol (communication protocol) is a set of rules to accomplish data transfer among units along the bus.
- OCB (On-Chip Bus) is an interconnection mechanism residing within a SoC and used to interconnect design blocks (i.e., IP) in the SoC.

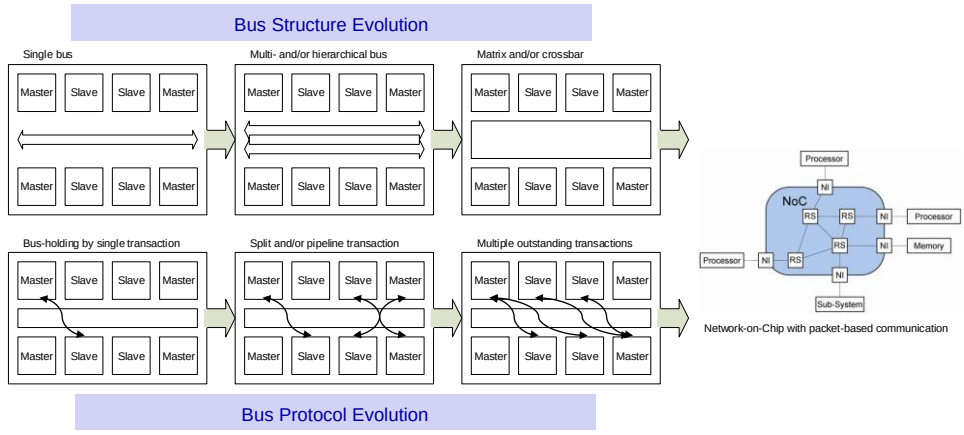
On-Chip Network evolved from wire to network through bus



NOC: Network on a Chip



On-Chip Bus Evolution



De factor standard buses for SoC

	AMBA	CoreConnect	Wishbone	Avalon
owner	ARM	IBM	OpenCores	Altera
license	Open and royalty-free, but license scheme is not clear	free, but need agreement	open/free, no license	proprietary
where	www.arm.com	www.ibm.com	www.opencores.org	www.altera.com
buses	AXI, AHB, APB	PLB, OPB, DCR		
arbitration	central	central	central	central
clock	synchronous	synchronous	synchronous	synchronous

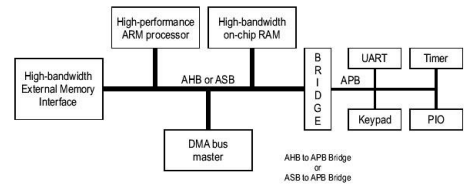
AMBA: Advanced Microcontroller Bus Architecture

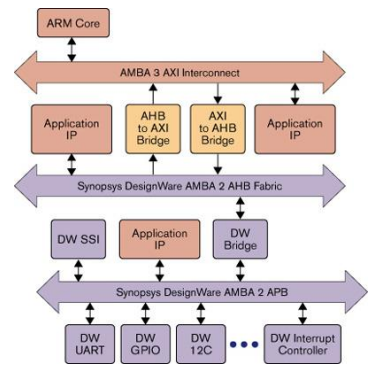
Agenda

- Where it is about
- What is on-chip bus
- Standard and/or de factor standard bus n ot for SoC
- Interconnect landscape not for SoC
- De factor standard buses for SoC
- Arbitration
- Burst transfers
- Pipelined and split transfers
- Data ordering
- Justificatin or not for data lane
- ARM AMBA
- IBM CoreConnect
- Altera Avalon
- OpenCores Wishbone

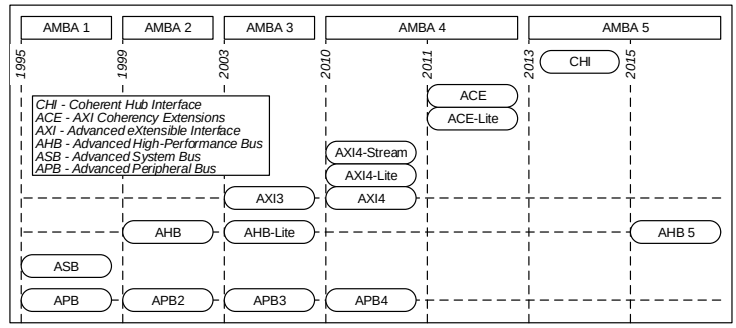
AMBA bus

- AMBA (Advanced Microcontroller Bus Architecture) AHB
  - Open standard on-chip bus
  - Single clock-edge operation
  - Non-tristate bus
  - Non-justified data lane
- AMBA buses
  - AXI4: AXI (in 2009)
  - AXI: Advanced eXtensible Interface (in 2003)
  - AHB: Advanced High-performance Bus (in 1999)
  - ASB: Advanced System Bus (in 1995)
  - APB: Advanced Peripheral Bus (in 1995)
- Other specs
  - ACE (AXI Coherency Extensions)
  - ATB (Advanced Trace Bus)





Evolution of AMBA Standards

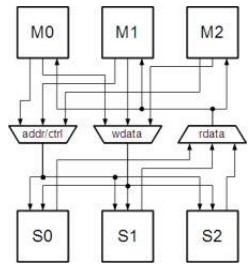


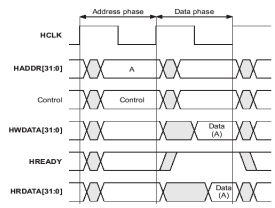
- CHI - Coherent Hub Interface - The highest performance, used in networks and servers
- ACE - AXI Coherency Extensions - Used in big.LITTLE™ systems for smartphones, tablets, etc.
- AXI - Advanced eXtensible Interface - The most widespread AMBA interface. Connectivity up to 100's of Masters and Slaves in complex SoC's
- AHB - Advanced High-Performance Bus - The main system bus in microcontroller usage
- APB - Advanced Peripheral Bus - Minimal gate count for peripherals
- ATB - Advanced Trace Bus - For moving trace data around the chip

refer to “Ashley Stevens, Introduction to AMBA® 4 ACE™ and big.LITTLE™ Processing Technology, July 2013.”

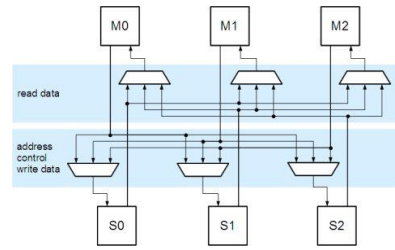
AHB and AXI

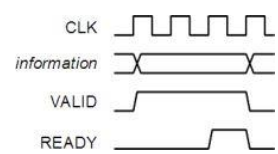
AHB





AXI





AMBA buses

AMBA 4 AXI	AMBA 3 AXI	AMBA 2 AHB
an extension of AMBA 3 AXI Burst length up to 256 Quality-of-service Removal of lock transaction Removal of write interleaving	channel architecture registers slices one address for burst up to 16 multiple outstanding bursts out of order completion data interleaving low-power interface	burst transfers pipelined operation split transactions single-cycle bus master handover single-clock edge operation multiple bus masters (up to 16) two uni-directional 32-bit data bus for read and write
AMBA 4 APB	**AMBA 3 APB**	**AMBA 2 APB**
an extension of AMBA 3 APB transaction protection (normal-privileged, secure-nonsecure, data-instruction) Sparse data transfer (partial access)	an extension of AMBA 2 APB wait state supported error response supported	low power latched address and control simple interface suitable for many peripherals

AHB

- High-performance synthesizable designs.
  - High-bandwidth operation
  - High clock frequency systems
- Features
  - multiple bus masters
  - Burst transfers
  - Split transactions
  - Single-cycle bus master handover
  - Single-clock edge operation
  - Non-tristate implementation
  - Wider data bus configurations (64/128 bits).

AHB

busname	AMBA AHB (new generation bus)
data bus width	32- 64- 128- 256-bit
address bus width	32 bit
architecture	(Multi) MASTER / (Multi) SLAVE
	arbitration logic interface well defined Single cycle bus master handover possible
data bus protocol	Single READ/WRITE transfer
	burst transfer (4 – 8 – 16 beats)
	Pipelined
	split transactions supported
	Byte/half-word/word transfer support
data ordering	No dynamic endianess
timing	Synchronous, well defined timing specs
interconnection	multiplexed implementation
supported interconnections	Non-tristate
	Separate data read & write bus required
technology	Technology independent

Reference: Overview Embedded Buses, Patrick Pelgrims

APB

- Low-power extension to the system bus (AHB/ASB)
  - Minimal power consumption
  - Reduced interface complexity.
- Local secondary bus that is encapsulated as a single AHB/ASB slave device
  - A slave module which handles local peripheral bus.
  - An APB bridge converts AHB or ASB transfers into a suitable format for the slave devices on the APB.
  - The bridge provides latching of all address, data and control signals
- Features
  - Low bandwidth
  - Unpipelined bus interface.
    - address and control valid throughout the access (unpipelined)
  - All signal transitions are only related to the rising edge of the clock
  - Zero-power interface during non-peripheral bus activity (peripheral bus is static when not in use)
  - Timing can be provided by decode with strobe timing (unclocked interface)
  - Write data valid for the whole access
    - allowing glitch-free transparent latch implementations).

APB

busname	AMBA APB
data bus width	8-16-32-bit
address bus width	32 bit
tagging	No tagging
architecture	(Single) MASTER (bridge) / (Multi) SLAVE
	No arbitration logic needed
data bus protocol	2 cycle single READ/WRITE transfer
	No burst transfer
	Non-Pipelined
timing	Synchronous, well defined timing specs
interconnection	Not defined
supported interconnections	Non-tristate-bus recommended
	Separate data read & write bus recommended
technology	Technology independent
power	Zero power when not in use

Reference: Overview Embedded Buses, Patrick Pelgrims

Agenda

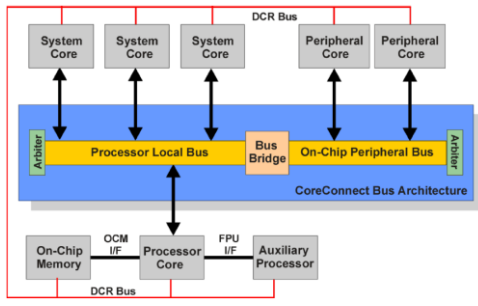
- Where it is about
- What is on-chip bus
- Standard and/or de factor standard bus n ot for SoC
- Interconnect landscape not for SoC
- De factor standard buses for SoC
- Arbitration
- Burst transfers
- Pipelined and split transfers
- Data ordering
- Justificatin or not for data lane
- ARM AMBA
- IBM CoreConnect
- Altera Avalon
- OpenCores Wishbone

CoreConnect

- Introduction
  - Owner: IBM
  - Open, but requires a restrictive license agreement
  - Version: 2.9 (32-bit), 3.5 (64-bit)
  - Operating frequency: 66MHz, 133MHz, 183MHz
  - Bandwidth: 264MB/s, 800MB/s, 2.9GB/s
  - http://www.ibm.com/chips/products/coreconnect
- PLB (Processor Local Bus)
  - High performance 32/64-bit on-chip bus used in highly integrated Core+ASIC systems
- OPB (On-chip Peripheral Bus)
  - For easy connection of on-chip peripheral devices
- DCR (Device Control Register)
  - To transfer data between the CPU's general purpose registers (GPRs) and the DCR slave logic's device control registers (DCRs)

Reference: Overview Embedded Buses, Patrick Pelgrims

CoreConnect



- PLB
  - High bandwidth capabilities
  - Pipelining
  - Mutiple masters
  - 32-, 64-, and 128-bit architecture
  - Split transactions
  - Cache Line transfers
  - Overlapped arbitration
- OPB
  - Connect to lower speed peripherals
  - Low power consumption.
  - Supports single-cycle data transfers
  - Multiple masters
- DCR Bus
  - Movement of GPR data between CPU and slave logic
  - Reduces loading and improves bandwidth of the PLB.

CoreConnect

- **CoreConnect is a complete and versatile solution, as it is well thought trough and has a good architecture. It clearly targets high performance systems, thus raising the complexity and offering many features that might be overkill in simple embedded applications.**

Reference: Overview Embedded Buses, Patrick Pelgrims

PLB

- To address the high performance and design flexibility needs of highly integrated Core+ASIC systems.
- High performance features
  - Overlapping of read and write transfers allows two data transfers per clock cycle
  - Decoupled address and data buses support split-bus transaction capability
  - Address pipelining
  - Late master request abort capability reduces latency associated with aborted requests.
  - Hidden (overlapped) bus request/grant protocol reduces arbitration latency.
  - Fully synchronous bus.
- System Design Flexibility
  - Sixteen masters and any number of slave devices.
  - Four levels of request priority for various arbitration schemes.
  - Bus arbitration-locking mechanism allows for master-driven atomic operations.
  - Byte-enable capability
  - Support for 16-, 32-, and 64-byte line data transfers.
  - Read word address capability (that is, target word-first or sequential).
  - Byte, halfword, and word burst data transfers in either direction.
  - Guarded and unguarded memory transfers allow the prefetching of instructions or data.
  - DMA buffered, flyby, peripheral to memory, memory to peripheral, and DMA memory to memory operations are supported.

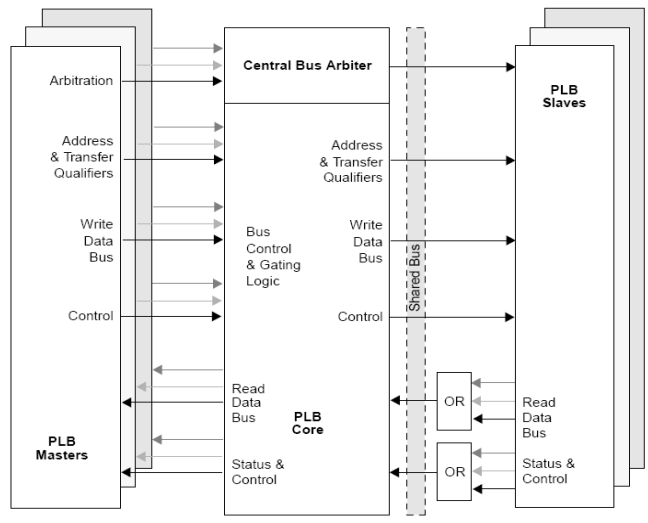
Reference: 32-bit Processor Local Bus Architecture Specifications Version 2.9

PLB

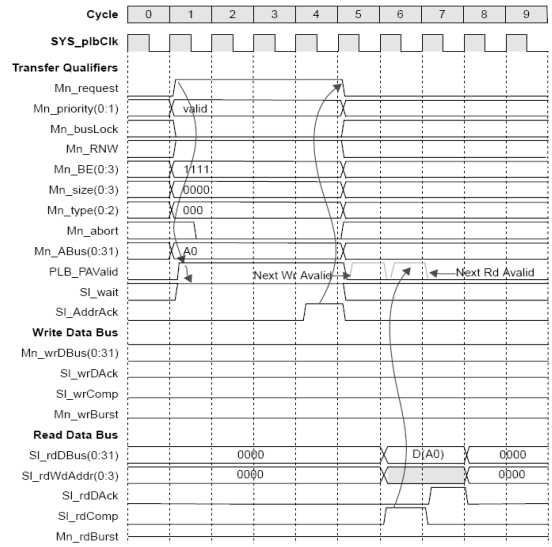
busname	CORECONNECT PLB
data bus width	32-, 64-, 128-,256-bit
address bus width	32-bit (with address pipelining, reducing latency)
architecture	(Multi) MASTER [MAX 8]/ (Multi) SLAVE
	Arbiters with different priority schemes available as soft-core
data bus protocol	Single READ/WRITE transfer
	Overlapped READ & WRITE (2transfers/cycle)
	Burst transfer (16-64 byte bursts)
	pipelining
	Split transfer support
	Special DMA modes (flyby,…)
timing	Fully synchronous
interconnection	multiplexed implementation (=crossbar switch)
supported interconnections	Non tri-state
	Separate data read & write bus
technology	Technology independent

Reference: Overview Embedded Buses, Patrick Pelgrims

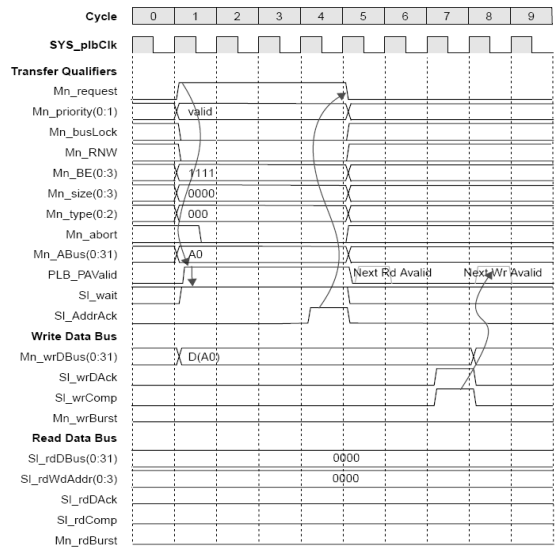
PLB Interconnect



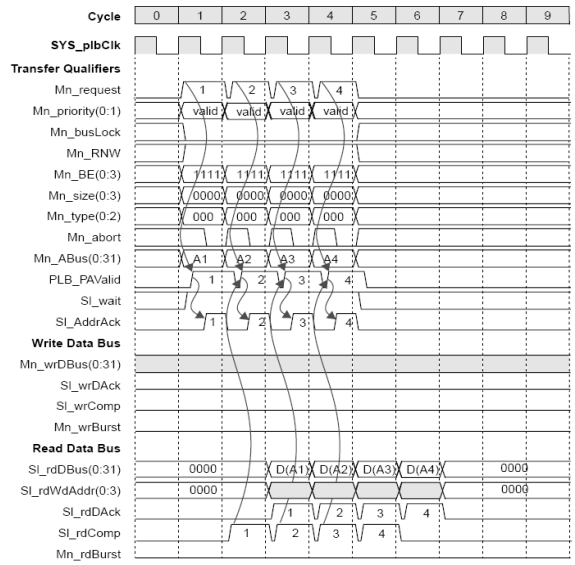
PLB Read Transfer



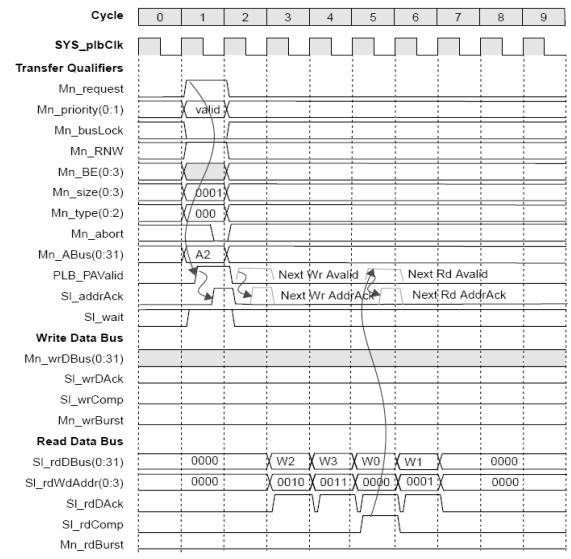
PLB Write Transfer



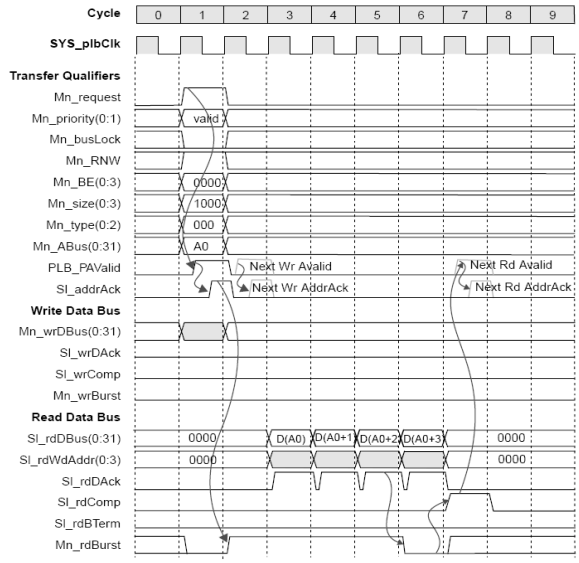
Back-to-Back Read Transfers



Four-word Line Read Transfers



Sequential Burst Read Transfers



OPB

- For easy connection of on-chip peripheral devices
- It is not intended to connect directly to the processor core.
  - The processor core can access the slave peripherals on this bus through the PLB to OPB bridge unit
- Features
  - Up to a 64-bit address bus / 32-bit or 64-bit data bus implementations
  - Fully synchronous
  - Provides support for 8-bit, 16-bit, 32-bit, and 64-bit slaves / 32-bit and 64-bit masters
  - Dynamic bus sizing; byte, halfword, fullword, and doubleword transfers
  - Optional Byte Enable support
  - Uses a distributed multiplexer method
  - Byte and halfword duplication for byte and halfword transfers
  - Single cycle transfer of data between OPB bus master and OPB slaves
  - Sequential address protocol support
  - A 16-cycle fixed bus timeout provided by the OPB arbiter
    - OPB slave is capable of disabling the fixed timeout counter to suspend bus timeout error
  - Support for multiple OPB bus masters
  - Bus parking for reduced latency
  - OPB masters may lock the OPB bus arbitration
  - OPB slaves capable of requesting retry to break possible arbitration deadlock
  - Bus arbitration overlapped with last cycle of bus transfers

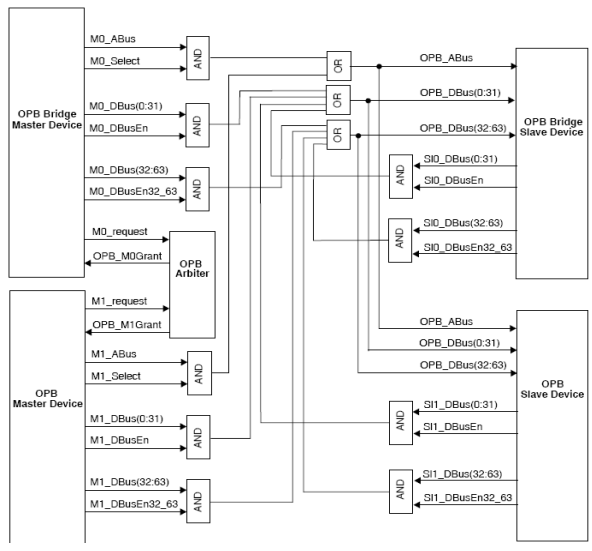
Reference: On-Chip Peripheral Bus Architecture Specifications Version 2.1

OPB

busname	CORECONNECT OPB
data bus width	8-, 16-,32-bit
address bus width	32-bit
architecture	(Multi) MASTER / (Multi) SLAVE
	Arbiters with different priority schemes available as soft-core
	Dynamic bus sizing possible
data bus protocol	Single READ/WRITE transfer
	Burst support
	Retry support
	Single byte, half word or word transfers
	DMA support
timing	Fully synchronous
interconnection	multiplexed implementation
supported interconnections	Non tri-state
	Separate data read & write bus
technology	Technology independent
power	Bus parking support

Reference: Overview Embedded Buses, Patrick Pelgrims

OPB Implementation



DCR Bus

- To transfer data between the CPU's general purpose registers (GPRs) and the DCR slave logic's device control registers (DCRs)
- To remove configuration registers from the memory address map.
  - To reduces loading
  - To improves bandwidth of the processor local bus.
- Fully synchronous
  - The slower clock's rising edge always corresponds to the faster clock's rising edge.
- The DCR bus is typically implemented as a distributed mux
  - Each sub-unit not only has a path to place its own DCRs on the CPU's DCR read path
  - but also has a path which bypasses its DCRs and places another unit's DCRs on the CPU's DCR read path.
- The DCRs are on-chip registers that exist architecturally outside the processor core.
  - Move to device control register (mtdcr) instruction
  - Move from device control register (mfdcr) instructions.
- Features
  - 10-bit address bus and 32-bit data bus
  - 2-cycle minimum read or write transfers extendable by slave or master
  - Handshake supports clocked asynchronous transfers
  - Slaves may be clocked either faster or slower than master
  - Single device control register bus master
  - Distributed multiplexer architecture
  - A simple but flexible interface

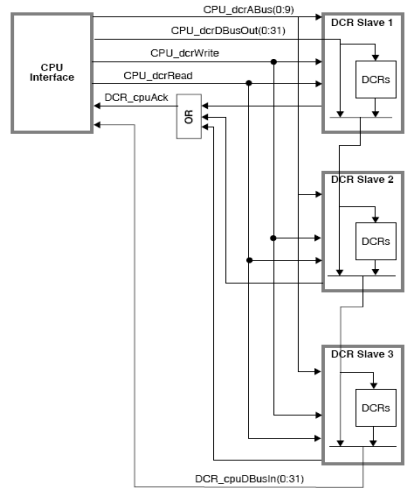
Reference: 32-Bit Device Control Register Bus Architecture Specifications Version 2.9

DCR Bus

busname	CORECONNECT DCR
data bus width	32-bit
address bus width	10-bit
Interconnection	multiplexed implementation
purpose	Transfer data between the CPU's general purpose registers (GPR) and other (peripheral) registers, not meant for real data transfers Designed to reduce load on PLB and OPB

Reference: Overview Embedded Buses, Patrick Pelgrims

DCR Connection



Agenda

- Where it is about
- What is on-chip bus
- Standard and/or de factor standard bus n ot for SoC
- Interconnect landscape not for SoC
- De factor standard buses for SoC
- Arbitration
- Burst transfers
- Pipelined and split transfers
- Data ordering
- Justificatin or not for data lane
- ARM AMBA
- IBM CoreConnect
- Altera Avalon
- OpenCores Wishbone

Avalon

- Introduction
  - Owner: Altera
  - Proprietary
  - Version: 1.2
  - Operating frequency: user defined
  - http://www.altera.com

Avalon

- To accommodate peripheral development for the system-on-a-programmable-chip (SOPC) environment
- Altera's parameterized interface bus used by the Nios embedded processor
- Features
  - Separate Address, Data and Control Lines
    - Provides the simplest interface to on-chip logic.
    - Using dedicated address and data paths
  - Up to 128-bit Data Width
  - Synchronous Operation
  - Dynamic Bus Sizing
    - Handles the details of transferring data between peripherals with different data widths
    - Avalon peripherals with differing data widths can interface easily with no special design considerations.
  - Simplicity
  - Low resource utilization
  - High performance
    - Up to one-transfer-per-clock

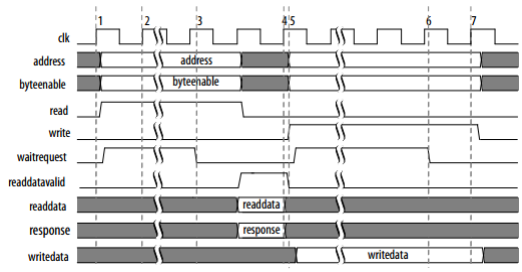
Reference: Avalon Interface Specification

Avalon

Busname	AVALON
data bus width	8, 16 or 32 bits
address bus width	32-bit
architecture	multi-master / multi slave
	multi-master arbitration logic
specific features	interrupt-priority assignment
	wait-state generation
	read & write transfers with latency
data bus protocol	one or more bus cycles
	streaming transfers (burst)
	single byte, half word or word transfers
	fixed- or peripheral-controlled wait states
	with or without setup time
timing	all signals synchronous with Avalon clock
	simple timing behavior
size	minimal FPGA resources
supported interconnections	separate address, data and control lines
	tri-state signals (external) only with bridge
technology	Altera Avalon can only be implemented on Altera devices using SOPC Builder

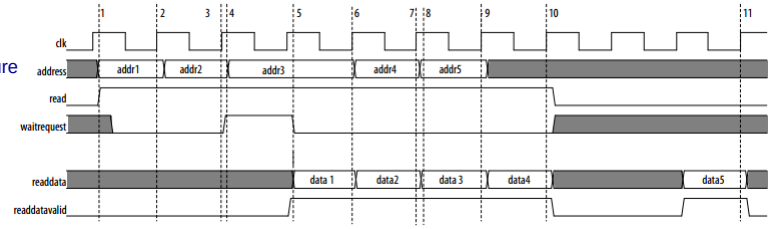
Reference: Overview Embedded Buses, Patrick Pelgrims

Avalone read/write

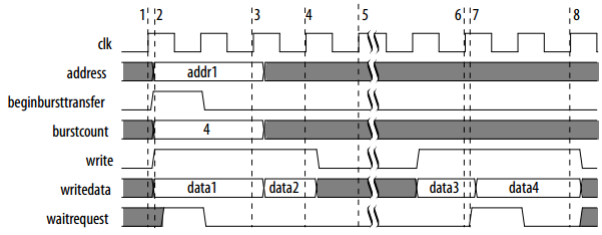


Normal case

Pipeline feature

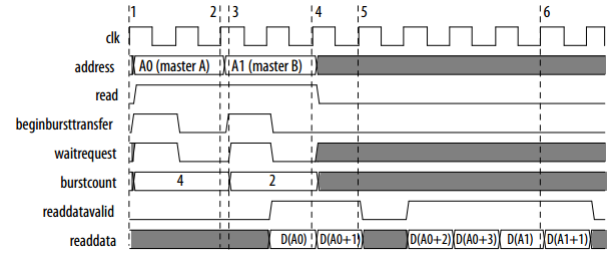


Avalone read/write



Write burst

Read burst

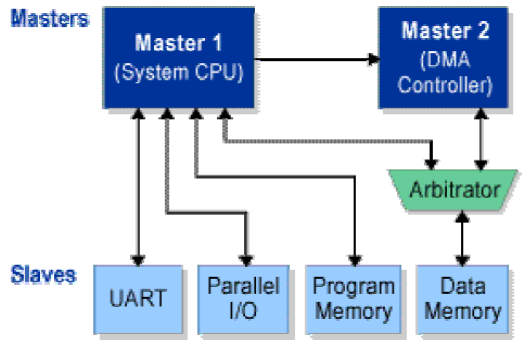


Avalon

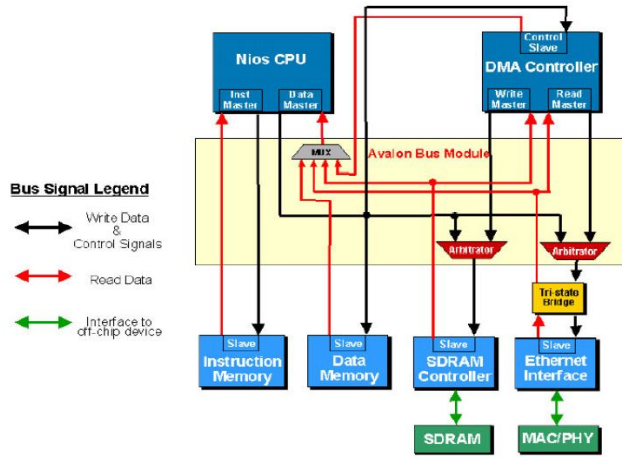
- Avalon switch fabric has a set of pre-defined signal types with which a user can connect one or more intellectual property (IP) blocks.
- The wizard-based Altera's SOPC Builder system development tool automatically generates the Avalon switch fabric logic.
- The generated switch fabric logic includes
  - chipselect signals for data-path multiplexing
  - address decoding
  - wait-state generation
  - interrupt-priority assignment
  - dynamic bus sizing
  - multi-master arbitration logic
  - advanced switch fabric transfers
- Avalon masters and slaves interact with each other based on slave-side arbitration.

Reference: Overview Embedded Buses, Patrick Pelgrims

Slave-Side Arbitration



Avalon Interconnection



Agenda

- Where it is about
- What is on-chip bus
- Standard and/or de factor standard bus n ot for SoC
- Interconnect landscape not for SoC
- De factor standard buses for SoC
- Arbitration
- Burst transfers
- Pipelined and split transfers
- Data ordering
- Justificatin or not for data lane
- ARM AMBA
- IBM CoreConnect
- Altera Avalon
- OpenCores Wishbone

WISHBONE

- Introduction
  - Owner: Silicore Corporation
  - Open standard, no license requred
  - Version: Rev. B.3
  - Operating frequency: user defined
  - http://www.opencores.org/wishbone

WISHBONE

- Simple, compact, logical IP core hardware interfaces
- Variable core interconnection methods
  - Point-to-point
  - Shared bus
  - Crossbar switch
  - Data flow interconnection
  - switched fabric interconnections
  - Off chip
- User-defined tags
  - Applying information to an address bus, a data bus or a bus cycle.
  - Helpful when modifying a bus cycle to identify information such as:
    - Data transfers
    - Parity or error correction bits
    - Interrupt vectors
    - Cache control operations
- Multi-MASTER
- Arbitration methodology is defined by the end user

Reference: WISHBONE System-on-Chip (SoC) Interconnection Architecture for Portable IP Cores Revision: B.3

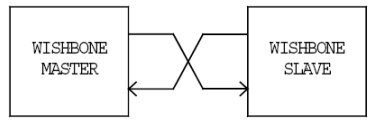
WISHBONE

busname	Wishbone
data bus width	8 to 64 bits
address bus width	8 to 64 bits
tagging	address, data-in and out, cycle tags are user defined
architecture	(Multi) MASTER / (Multi) SLAVE
	arbitration logic is user defined (priority, round-robin,… arbiter)
data bus protocol	single READ / WRITE cycle
	BLOCK transfer cycle
	RMW (read-modify-write) cycle
	EVENT cycle
	Up to one data transfer per clock cycle.
data ordering	LITTLE and BIG ENDIAN support
timing	synchronous (simple design, ease of test)
	simple timing specs
size	very few logic gates (dependant on architecture)
Interconnection	point to point (a)
	Data flow (b)
	shared bus (c)
	Crossbar switch (d)
supported interconnections	unidirectional bus
	bi-directional bus
	Multiplexer based interconnections
	Tristate based interconnections
	off-chip I/O
	…
technology	Technology independent

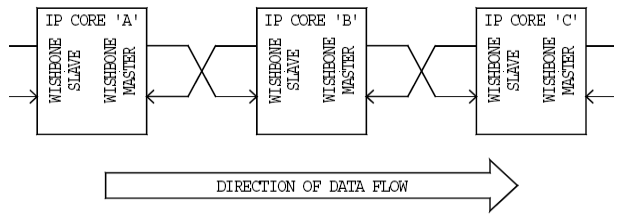
uses, Patrick Pelgrims

Interconnection

- Point-to-point interconnection

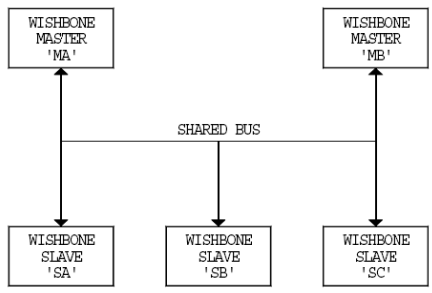


- Data Flow Interconnection

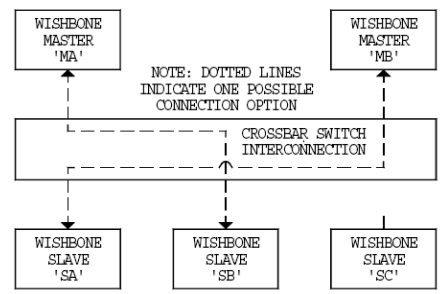


Interconnection

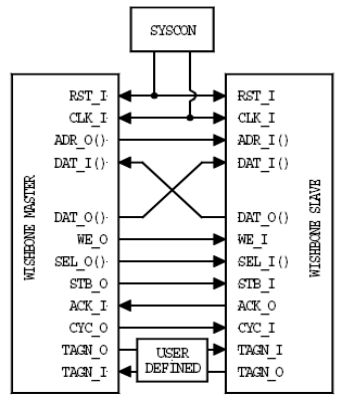
- Shared bus interconnection



- Crossbar switch interconnection

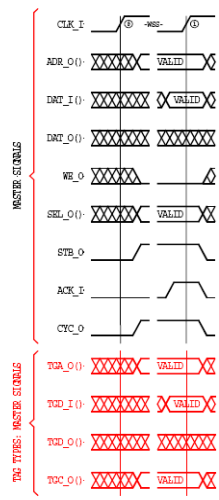


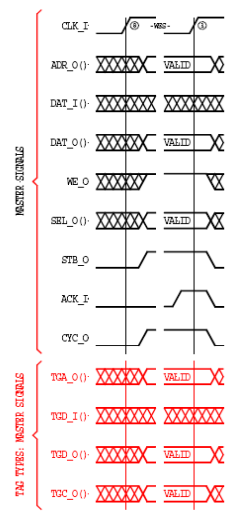
Standard Connection



Single Transfers

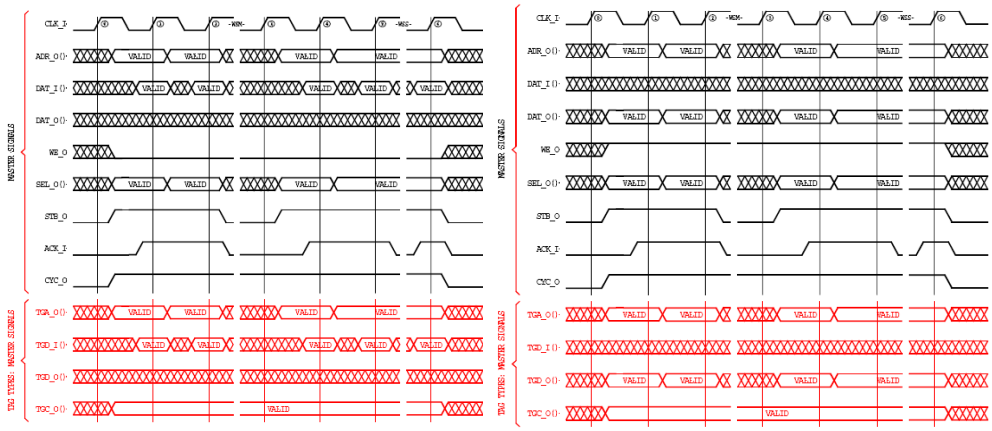
- Single read
- Single write



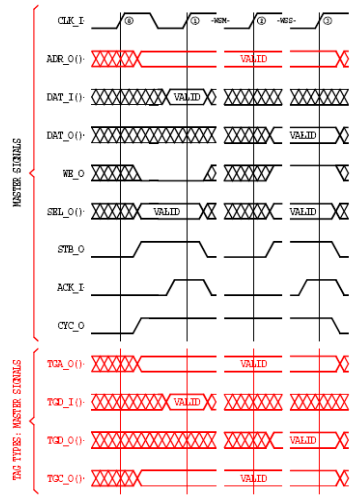


Block Transfer

- Block Read Transfer
- Block Write Transfer



RMW Cycle



Summary

bus name	bus owner	status [Open Standard/ Licensed]	version	year
AMBA	ARM	OS	Rev. 2.0	1999
Avalon	Altera Corporation	OS	1.3	2005
CoreConnect	IBM	OS	2.9 (32 bit PLB)	2001
			3.5 (64 bit PLB)	2001
			4.6 (128 bit PLB)	2004
			2.1 (OPB)	2001
			2.9 (DCR)	2000
Wishbone	OpenCores	OS	Rev. B.3	2002
SiliconBackplane	Sonics	L	III	2002
CoreFrame	Palmchip Corporation	L	Rev. 1,01	2002
Marble	The Uni. of Manchester	n/a	n/a	1999
PI bus	OMI	OS	Rev. 0.3d	1996
hline OCP	OCP-IP	OS	Rev. 2.1	2005
hline CVI	VSIA	OS	Rev. 2.0	n/a

Reference: An Overview of On-Chip Buses, Milica Mitic, Mile Stojcev

	Topology						Arbitration						Bus width		Transvers						
Name	Point-to-point	Ring	Unilevel shared bus	Hierarchical bus	Interconnection network	Synchronous/Asinhronous	Static priority	TDMA	Lottery	Round-robin	Token Passing	CDMA	Data bus width [bit]	Address bus width [bit]	Handshaking	Split transfer	Pipelined transfer	Burst transfer	Broadcast	Multicast	Operating frequancy
AMBA	-	-	-	×	-	S	7*	7*	7*	7*	7*	7*	8*	32	×	×	×	×	n/a	n/a	11*
Avalon	×	-	-	-	-	S	13*	13*	13*	13*	13*	13*	1-128	1-32	-	-	×	×	-	-	n/a
Core Connect	-	1*	-	1*	-	S	4*	-	-	-	-	-	9*	10*	×	×	×	×	n/a	n/a	12*
Wishbone	×	×	×	-	×	S	3*	3*	3*	3*	3*	3*	8,16,32,64	1-64	×	n/a	-	×	n/a	n/a	11*
Silicon Backplane	-	-	-	-	×	S	-	6*	-	6*	-	-	8,16,32,64	n/a	×	×	×	×	×	×	n/a
Core Frame	14*	-	-	-	-	S	3*	3*	3*	3*	3*	3*	n/a	n/a	2*	-	n/a	×	×	n/a	n/a
Marble	-	-	-	×	-	A	×	-	-	-	-	-	n/a	n/a	×	×	×	×	×	n/a	n/a
PI bus	-	-	×	-	-	S	3*	3*	3*	3*	3*	3*	1-32	1-32	×	-	×	-	-	-	n/a
OCP	×	-	-	-	-	S	-	-	-	-	-	-	n/a	n/a	×	-	×	×	×	-	n/a
VCI	n/a	n/a	n/a	n/a	n/a	S	3*	3*	3*	3*	3*	3*	n/a	n/a	×	×	×	n/a	-	n/a	n/a
Lotterybus	-	-	-	×	-	S	-	-	×	-	-	-	n/a	n/a	n/a	n/a	n/a	×	n/a	n/a	n/a

Exceptions for Table:1* Data lines shared, control lines point-to-point ring; 2* Palmbus uses handshaking, Mbus does not; 3* Application specific, arbiter can be designed regarding to the application requirements; 4* Programmable priority fairness; 5* Two level arbitration, first level TDMA, second level static priority; 6* Two level arbitration, first TDMA, second round-robin token passing; 7* Application specific except for APB which requires no arbitration; 8* For AHB and ASB bus width is 32, 64, 128 or 256 byte, for APB 8, 16 or 32 byte; 9* For PLB bus width is 32, 64, 128 or 256 byte, for OPB 8, 16 or 32 byte and for DCR 32 byte; 10* For PLB and OPB bus width is 32 byte, and for DCR 10 byte; 11* User defined operating frequency; 12* Operating frequency depending on PLB width; 13* Slave side arbitration; 14* System of buses, Palmbus and Mbus, both are point-to-point;

Reference: An Overview of On-Chip Buses, Milica Mitic, Mile Stojcev

References

- AMBA Specification, Rev 2.0, ARM Limited.
- AHB-Lite Overview, ARM Limited, 2001.
- Multi-layer AHB Overview, ARM Limited, 2001.
- AMBA AXI Protocol Specification, v1.0, ARM Limited, 2004.
- AMBA 3 APB Protocol Specification, v1.0, ARM Limited, 2004.

- AMBA Specification, Rev 2.0, ARM Limited.
- AHB-Lite Overview, ARM Limited, 2001.
- Multi-layer AHB Overview, ARM Limited, 2001.