

Zynq Architecture



Objectives

- > **After completing this module, you will be able to:**
 - >> Identify the basic building blocks of the Zynq™ architecture processing system (PS)
 - >> Describe the ARM Cortex-A9 processor memory space
 - >> Connect the PS to the programmable logic (PL) through the AXI ports
 - >> Generate clocking sources for the PL peripherals
 - >> List the various AXI-based system architectural models
 - >> Name the five AXI channels
 - >> Describe the operation of the AXI streaming protocol

Outline

- > **Zynq SoC (AP SoC)**
- > Zynq SoC Processing System (PS)
- > Processor Peripherals
- > Clock, Reset, and Debug Features
- > AXI Interfaces
- > Summary



The PS and the PL

> The Zynq-7000 SoC architecture consists of two major sections

>> PS: Processing system

- Dual ARM Cortex-A9 processor based
 - Single core versions also now available
- Multiple peripherals
- Hard silicon core

>> PL: Programmable logic

- Shares the same 7 series programmable logic as
 - Artix™-based devices: Z-7010, Z-7015, and Z-7020 (high-range I/O banks only)
 - Kintex™-based devices: Z-7030, Z-7035, Z-7045, and Z-7100 (mix of high-range and high-performance I/O banks)

> This section focuses on the PS

Zynq-7000 Family Highlights

> **Complete ARM®-based processing system**

- >> Application Processor Unit (APU)
 - Dual ARM Cortex™-A9 processor
 - Caches and support blocks
- >> Fully integrated memory controllers
- >> I/O peripherals

> **Tightly integrated programmable logic**

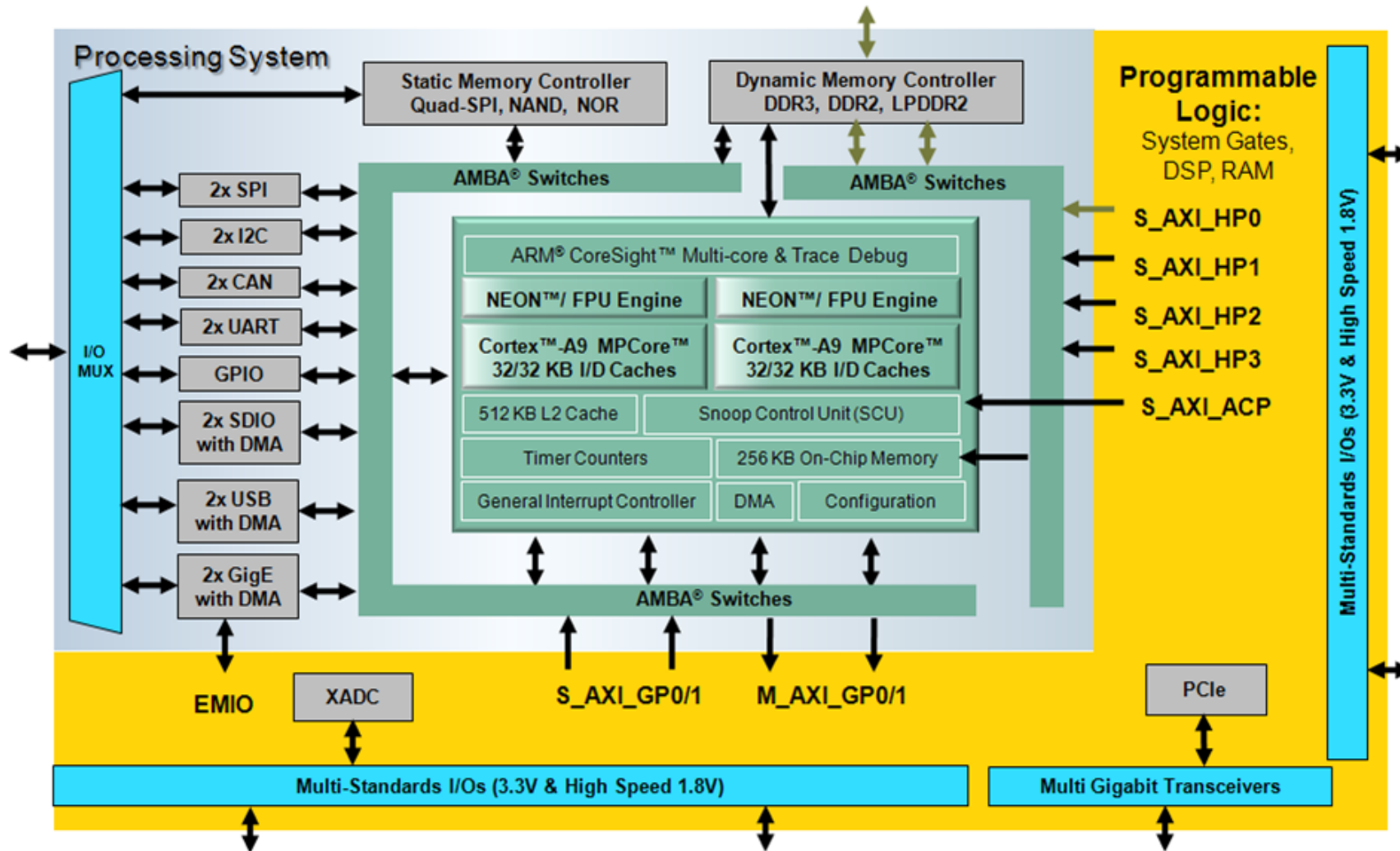
- >> Used to extend the processing system
- >> Scalable density and performance

> **Flexible array of I/O**

- >> Wide range of external multi-standard I/O
- >> High-performance integrated serial transceivers
- >> Analog-to-digital converter inputs



Zynq-7000 SoC Block Diagram



PS Components

- > **Application processing unit (APU)**
- > **I/O peripherals (IOP)**
 - >> Multiplexed I/O (MIO), extended multiplexed I/O (EMIO)
- > **Memory interfaces**
- > **PS interconnect**
- > **DMA**
- > **Timers**
 - >> Public and private
- > **General interrupt controller (GIC)**
- > **On-chip memory (OCM): RAM**
- > **Debug controller: CoreSight**

Outline

- > Zynq All Programmable SoC (AP SoC)
- > **Zynq SoC Processing System (PS)**
- > Processor Peripherals
- > Clock, Reset, and Debug Features
- > AXI Interfaces
- > Summary



ARM Processor Architecture

- > **ARM Cortex-A9 processor implements the ARMv7-A architecture**
 - >> ARMv7 is the ARM Instruction Set Architecture (ISA)
 - >> ARMv7-A: Application set that includes support for a Memory Management Unit (MMU)
 - >> ARMv7-R: Real-time set that includes support for a Memory Protection Unit (MPU)
 - >> ARMv7-M: Microcontroller set that is the smallest set
- > **The ARMv7 ISA includes the following types of instructions (for backwards compatibility)**
 - >> Thumb instructions: 16 bits; Thumb-2 instructions: 32 bits
 - >> NEON: ARM's Single Instruction Multiple Data (SIMD) instructions
- > **ARM Advanced Microcontroller Bus Architecture (AMBA®) protocol**
 - >> AXI3: Third-generation ARM interface
 - >> AXI4: Adding to the existing AXI definition (extended bursts, subsets)
- > **Cortex is the new family of processors**
 - >> ARM family is older generation; Cortex is current; MMUs in Cortex processors and MPUs in ARM

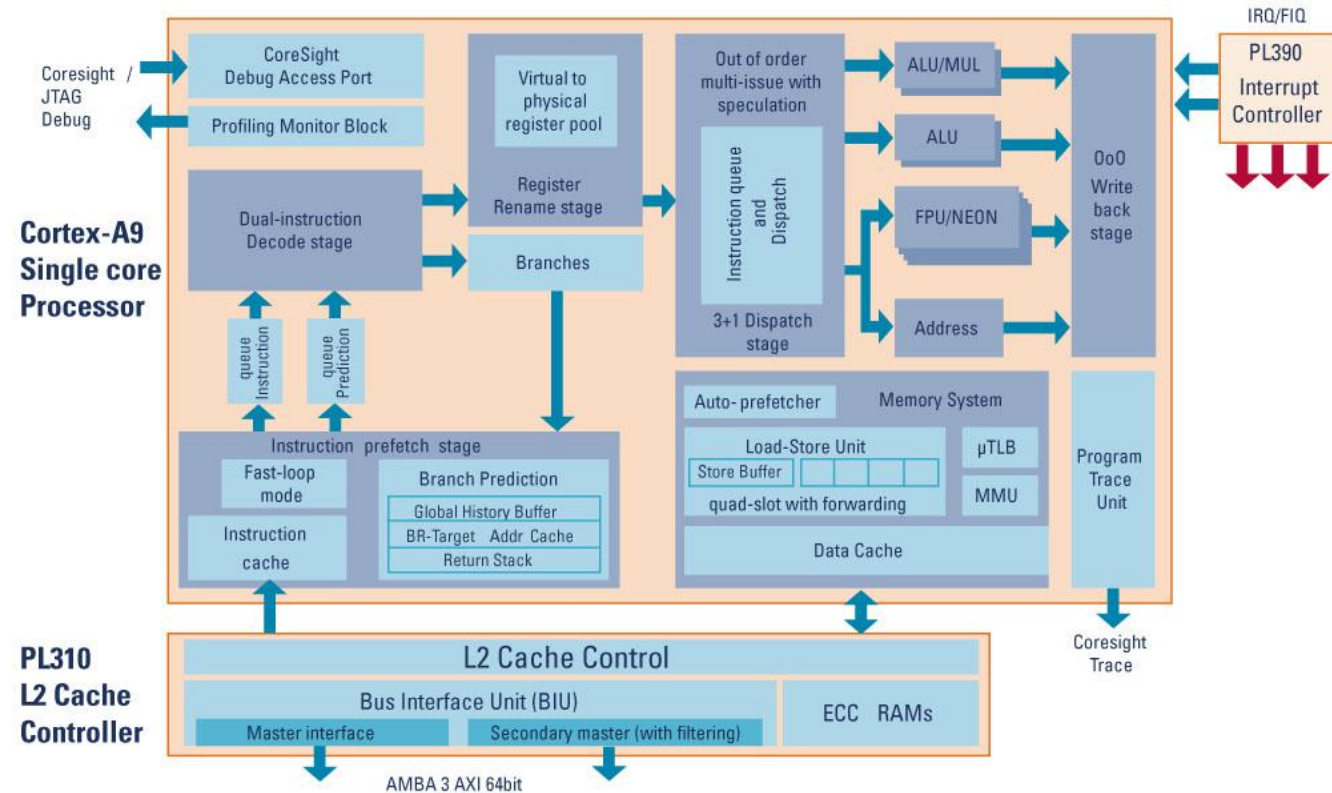
ARM Cortex-A9 Processor Power

- > **Dual-core processor cluster**
- > **2.5 DMIP/MHz per processor**
- > **Harvard architecture**
- > **Self-contained 32KB L1 caches for instructions and data**
- > **External memory based 512KB L2 cache**
- > **Automatic cache coherency between processor cores**
- > **1GHz operation (fastest speed grade)**



ARM Cortex-A9 Processor Micro-Architecture

- > Instruction pipeline supports out-of-order instruction issue and completion
- > Register renaming to enable execution speculation
- > Non-blocking memory system with load-store forwarding
- > Fast loop mode in instruction prefetch to lower power consumption

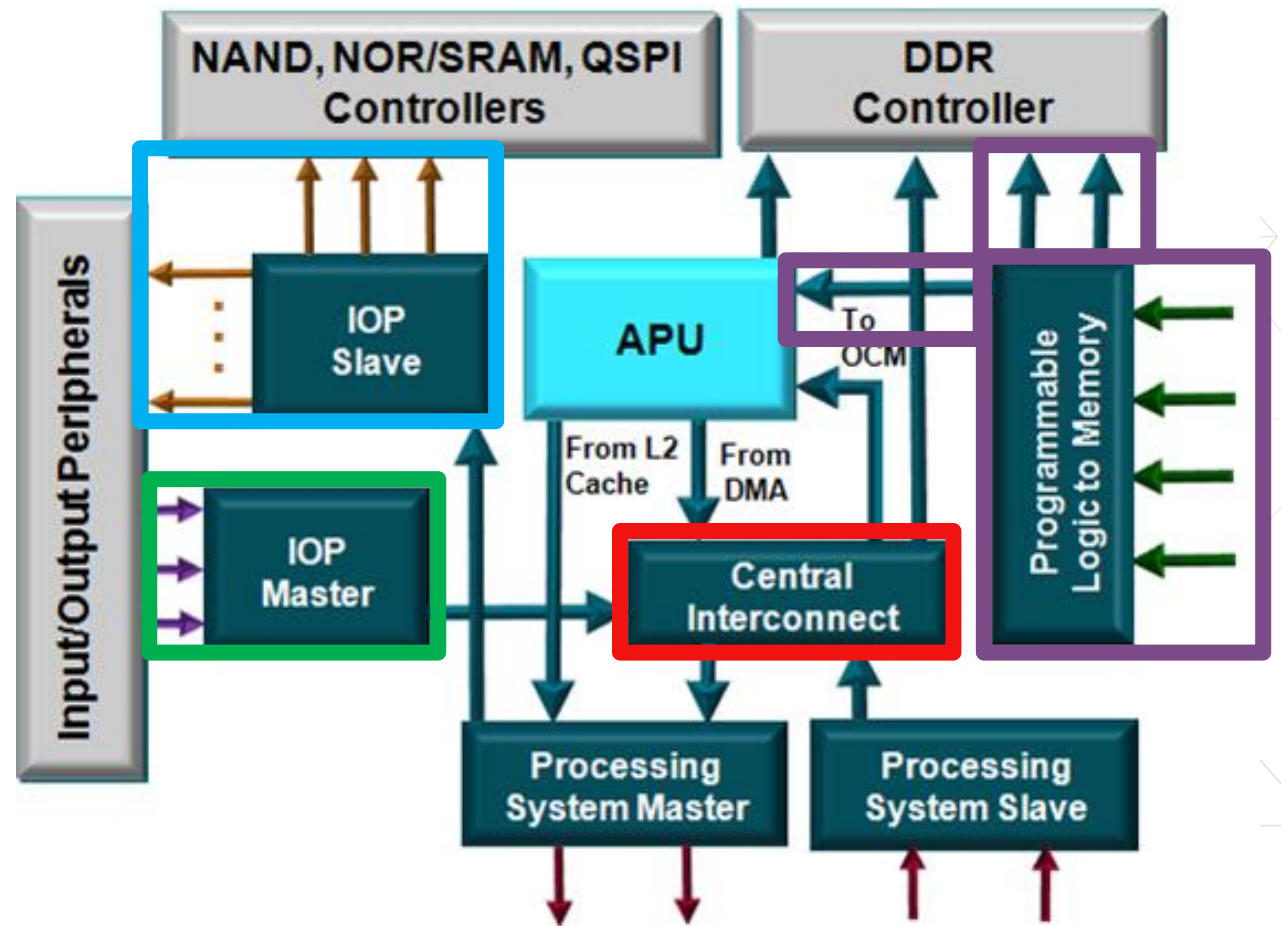


ARM Cortex-A9 Processor Micro-Architecture

- > **Variable length, out-of-order, eight-stage, super-scalar instruction pipeline**
 - >> Advanced pre-fetch with parallel branch pipeline enabling early branch prediction and resolution
 - >> Multi-issued into
 - Primary data processing pipeline
 - Secondary full data processing pipeline
 - Load-store pipeline
 - Compute engine (FPU/NEON) pipeline
- > **Speculative execution**
 - >> Supports virtual renaming of ARM physical registers to remove pipeline stalls due to data dependencies
 - >> Increased processor utilization and hiding of memory latencies
 - >> Increased performance by hardware unrolling of code loops
 - >> Reduced interrupt latency via speculative entry to Interrupt Service Routine (ISR)

Processing System Interconnect (1)

- > **Programmable logic to memory**
 - >> Two ports to DDR
 - >> One port to OCM SRAM
- > **Central interconnect**
 - >> Enables other interconnects to communicate
- > **Peripheral master**
 - >> USB, GigE, SDIO connects to DDR and PL via the central interconnect
- > **Peripheral slave**
 - >> CPU, DMA, and PL access to IOP peripherals



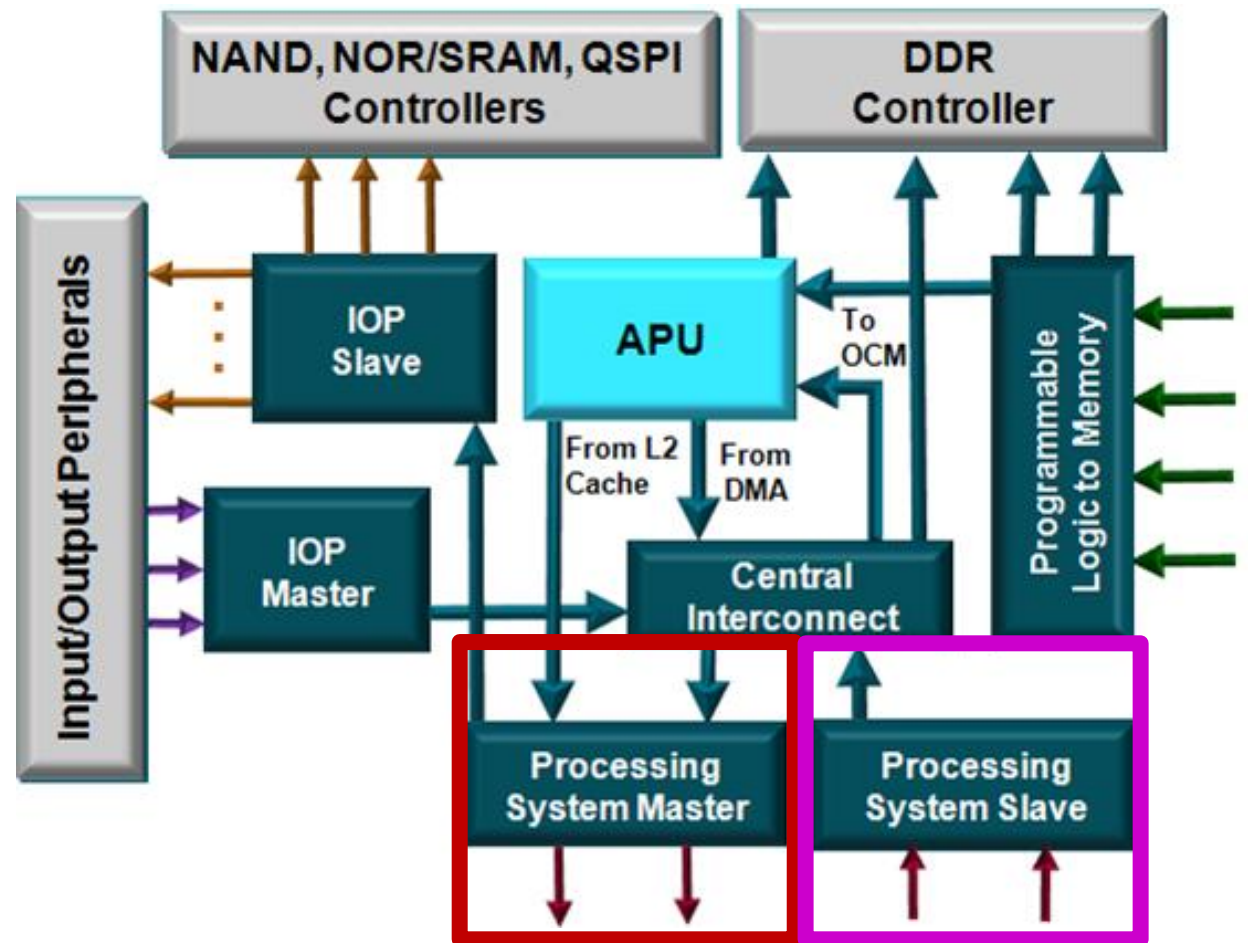
Processing System Interconnect (2)

> Processing system master

- >> Two ports from the processing system to programmable logic
- >> Connects the CPU block to common peripherals through the central interconnect


> Processing system slave

- >> Two ports from programmable logic to the processing system




Memory Map

- > The Cortex-A9 processor uses 32-bit addressing
- > PS peripherals and PL AXI slave interfaces are memory mapped to the Cortex-A9 processor cores
- > All AXI slave PL peripherals will be located between
4000_0000 and 7FFF_FFFF (connected to GP0)
and
8000_0000 and BFFF_FFFF (connected to GP1)



FFFC_0000 to FFFF_FFFF	OCM
FD00_0000 to FFFB_FFFF	Reserved
FC00_0000 to FCFF_FFFF	Quad SPI linear address
F8F0_3000 to FBFF_FFFF	Reserved
F890_0000 to F8F0_2FFF	CPU Private registers
F801_0000 to F88F_FFFF	Reserved
F800_1000 to F880_FFFF	PS System registers,
F800_0C00 to F800_0FFF	Reserved
F800_0000 to F800_0BFF	SLCR Registers
E600_0000 to F7FF_FFFF	Reserved
E100_0000 to E5FF_FFFF	SMC Memory
E030_0000 to E0FF_FFFF	Reserved
E000_0000 to E02F_FFFF	IO Peripherals
C000_0000 to DFFF_FFFF	Reserved
8000_0000 to BFFF_FFFF	PL (MAXI_GP1)
4000_0000 to 7FFF_FFFF	PL (MAXI_GP0)
0010_0000 to 3FFF_FFFF	DDR(address not filtered by SCU)
0004_0000 to 000F_FFFF	DDR(address filtered by SCU)
0000_0000 to 0003_FFFF	OCM



Zynq SoC Memory Resources

- > **On-chip memory (OCM)**
 - >> RAM
 - >> Boot ROM
- > **DDRx dynamic memory controller**
 - >> Supports LPDDR2, DDR2, DDR3
- > **Flash/static, memory controller**
 - >> Supports SRAM, QSPI, NAND/NOR FLASH



PS Boots First

- > **CPU0 boots from OCM ROM; CPU1 goes into a sleep state**
- > **On-chip boot loader in OCM ROM (Stage 0 boot)**
- > **Processor loads First Stage Boot Loader (FSBL) from external flash memory**
 - >> NOR
 - >> NAND
 - >> Quad-SPI
 - >> SD Card
 - >> Load from JTAG; not a memory device—used for development/debug only
- > **Boot source selected via package bootstrapping pins**
- > **Optional secure boot mode allows the loading of encrypted software from the flash boot memory**

Configuring the PL

- > **The programmable logic is configured after the PS boots**
- > **Performed by application software accessing the hardware device configuration unit**
 - >> Bitstream image transferred
 - >> 100-MHz, 32-bit PCAP stream interface
 - >> Decryption/authentication hardware option for encrypted bitstreams
 - In secure boot mode, this option can be used for software memory load
 - >> Built-in DMA allows simultaneous PL configuration and OS memory loading

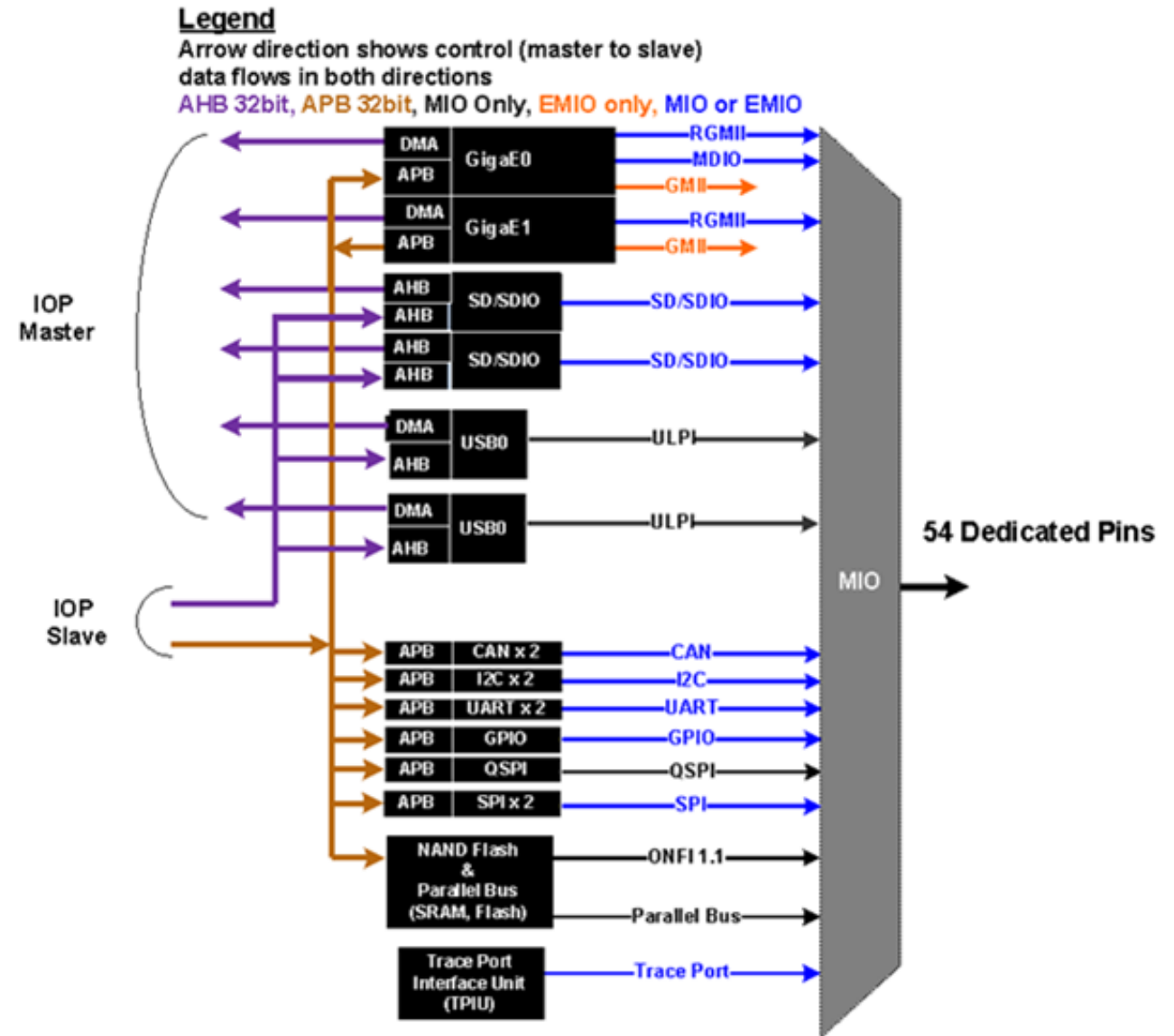
Outline

- > Zynq All Programmable SoC (AP SoC)
- > Zynq SoC Processing System (PS)
- > ***Processor Peripherals***
- > Clock, Reset, and Debug Features
- > AXI Interfaces
- > Summary



Input/Output Peripherals

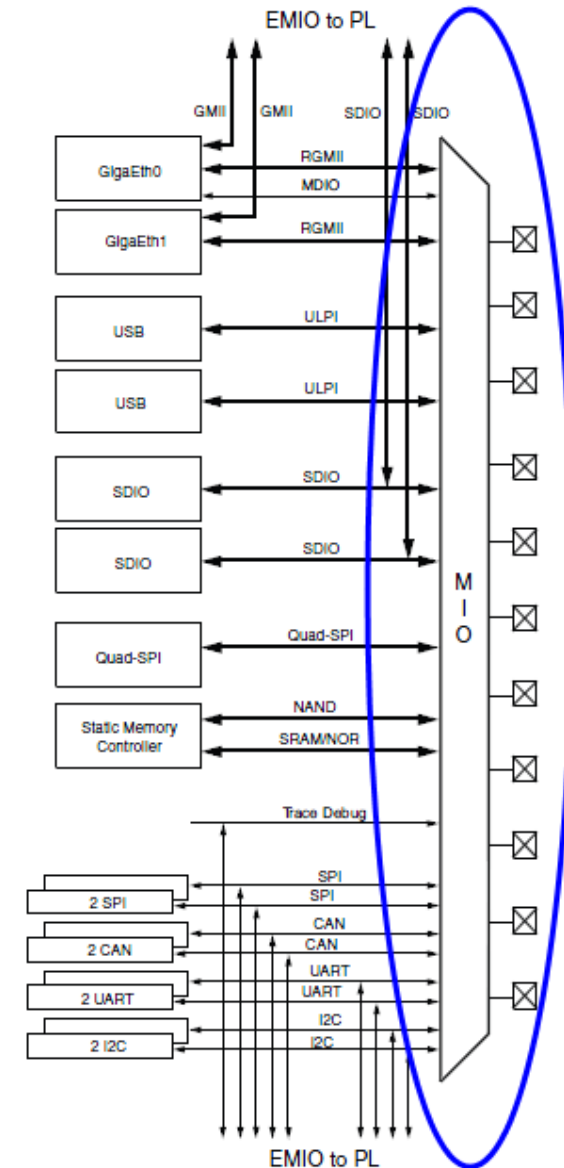
- > Two GigE
- > Two USB
- > Two SPI
- > Two SD/SDIO
- > Two CAN
- > Two I2C
- > Two UART
- > Four 32-bit GPIOs
- > Static memories
 - >> NAND, NOR/SRAM, Quad SPI
- > Trace ports



Multiplexed I/O (MIO)

> External interface to PS I/O peripheral ports

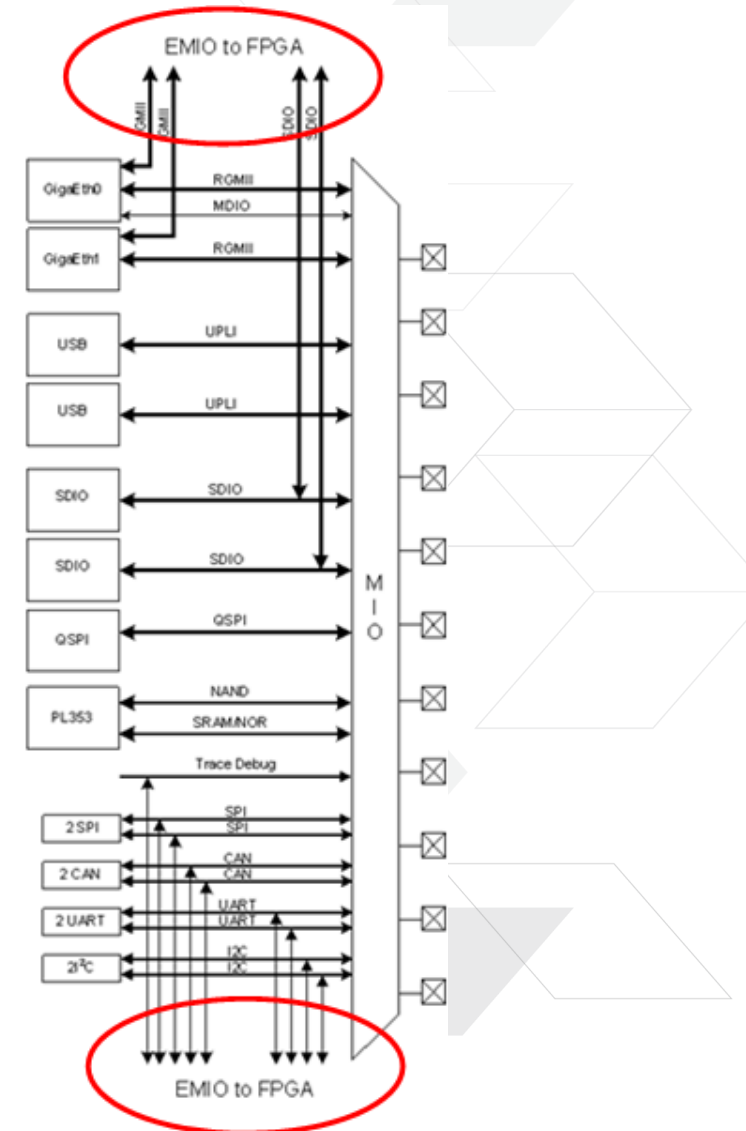
- >> 54 dedicated package pins available
- >> Software configurable
 - Automatically added to bootloader by tools
- >> Not available for all peripheral ports
 - Some ports can only use EMIO



Extended Multiplexed I/O (EMIO)

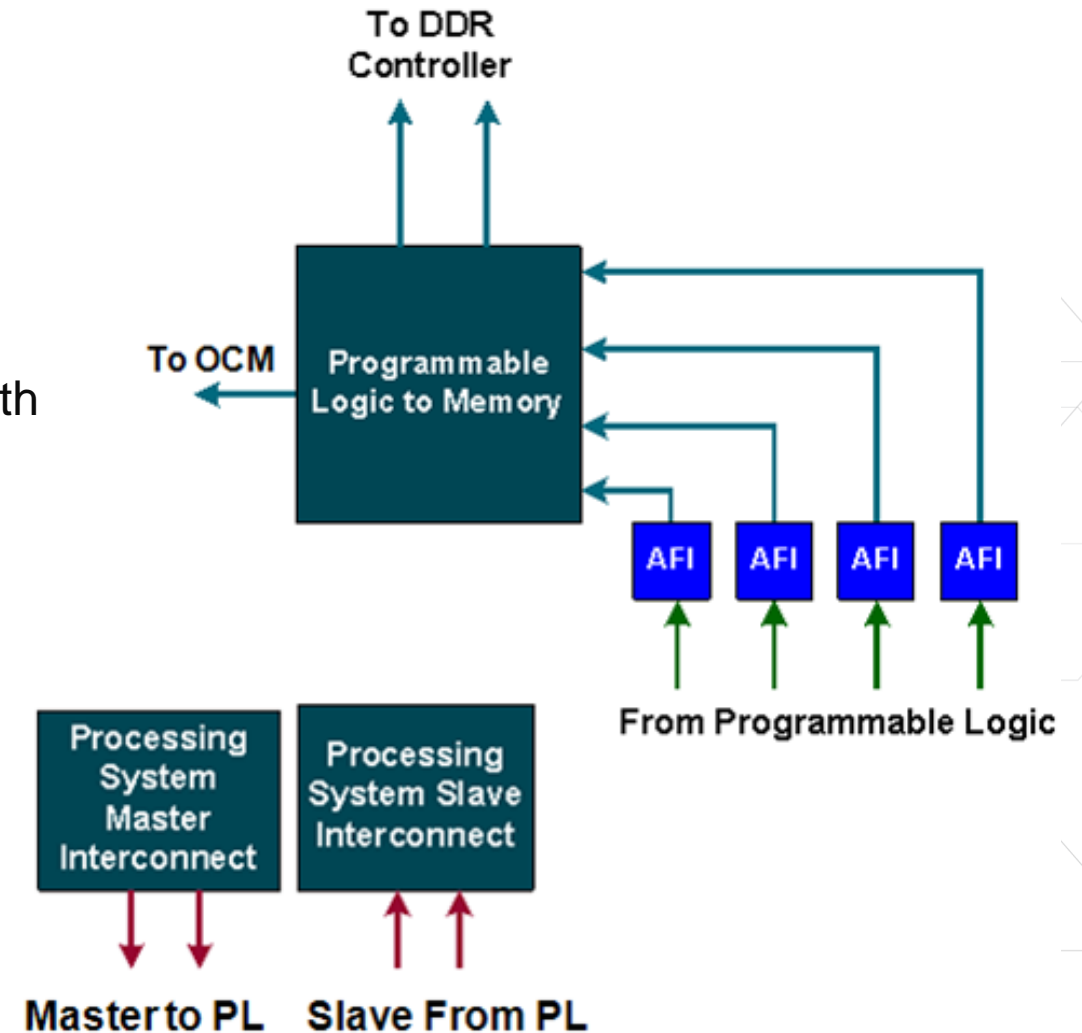
> Extended interface to PS I/O peripheral ports

- >> EMIO: Peripheral port to programmable logic
- >> Alternative to using MIO
- >> Mandatory for some peripheral ports
- >> Facilitates
 - Connection to peripheral in programmable logic
 - Use of general I/O pins to supplement MIO pin usage
 - Alleviates competition for MIO pin usage



PS-PL Interfaces

- > **AXI high-performance slave ports (HP0-HP3)**
 - >> Configurable 32-bit or 64-bit data width
 - >> Access to OCM and DDR only
 - >> Conversion to processing system clock domain
 - >> AXI FIFO Interface (AFI) are FIFOs (1KB) to smooth large data transfers
- > **AXI general-purpose ports (GP0-GP1)**
 - >> Two masters from PS to PL
 - >> Two slaves from PL to PS
 - >> 32-bit data width
 - >> Conversation and sync to processing system clock domain



PS-PL Interfaces

- > **One 64-bit accelerator coherence port (ACP) AXI slave interface to CPU memory**
- > **DMA, interrupts, events signals**
 - >> Processor event bus for signaling event information to the CPU
 - >> PL peripheral IP interrupts to the PS general interrupt controller (GIC)
 - >> Four DMA channel RDY/ACK signals
- > **Extended multiplexed I/O (EMIO) allows PS peripheral ports access to PL logic and device I/O pins**
- > **Clock and resets**
 - >> Four PS clock outputs to the PL with enable control
 - >> Four PS reset outputs to the PL
- > **Configuration and miscellaneous**

Outline

- > Zynq All Programmable SoC (AP SoC)
- > Zynq SoC Processing System (PS)
- > Processor Peripherals
- > ***Clock, Reset, and Debug Features***
- > AXI Interfaces
- > Summary



PL Clocking Sources

> PS clocks

- >> PS clock source from external package pin
- >> PS has three PLLs for clock generation
- >> PS has four clock ports to PL

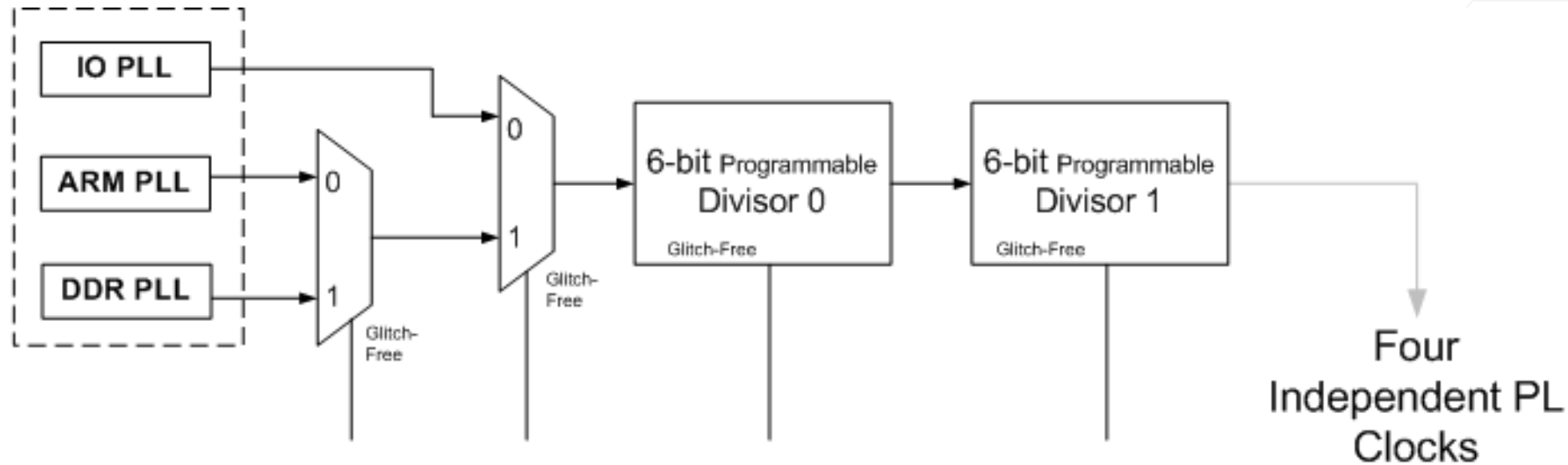
> The PL has 7 series clocking resources

- >> PL has a different clock source domain compared to the PS
- >> The clock to PL can be sourced from external clock capable pins
- >> Can use one of the four PS clocks as source

> Synchronizing the clock between PL and PS is taken care of by the architecture of the PS

> PL cannot supply clock source to PS

Clocking the PL



PL Fabric Clock	Control Register	Mux Ctrl Field	Mux Ctrl Field	Divider 0 Ctrl Field	Divider 1 Ctrl Field
PL Fabric 0	FPGA0_CLK_CTRL	SRCSEL, 4	SRCSEL, 5	DIVISOR 0, 13:8	DIVISOR 1, 25:20
PL Fabric 1	FPGA1_CLK_CTRL	SRCSEL, 4	SRCSEL, 5	DIVISOR 0, 13:8	DIVISOR 1, 25:20
PL Fabric 2	FPGA2_CLK_CTRL	SRCSEL, 4	SRCSEL, 5	DIVISOR 0, 13:8	DIVISOR 1, 25:20
PL Fabric 3	FPGA3_CLK_CTRL	SRCSEL, 4	SRCSEL, 5	DIVISOR 0, 13:8	DIVISOR 1, 25:20

→ FCLKCLK0
 → FCLKCLK1
 → FCLKCLK2
 → FCLKCLK3

Clock Generation (Using Zynq Tab)

- > **The Clock Generator allows configuration of PLL components for both the PS and PL**
 - >> One input reference clock
- > **Access GUI by clicking the Clock Generation Block, or select from Navigator**
- > **Configure the PS Peripheral Clock in the Zynq tab**
 - >> PS uses a dedicated PLL clock
 - >> PS I/O peripherals use the I/O PLL clock and ARM PLL

Clock Configuration [Summary Report](#)

Basic Clocking | Advanced Clocking

Input Frequency (MHz): 50.000000 CPU Clock Ratio: 6:2:1

Search: Q-

Component	Clock Source	Requested Frequ...	Actual Frequency(...)	Range(MHz)
Processor/Memory Clocks				
CPU	ARM PLL	650	650.000000	50.0 : 667.0
DDR	DDR PLL	525	525.000000	200.000000 : 534.000...
IO Peripheral Clocks				
SMC	IO PLL	100	100.000000	10.000000 : 100.000000
QSPI	IO PLL	200	10.000000	10.000000 : 200.000000
ENET0	IO PLL	1000 Mbps	10.000000	
ENET1	IO PLL	1000 Mbps	10.000000	
SDIO	IO PLL	50	50.000000	10.000000 : 125.000000
SPI	IO PLL	166.666666	166.666672	0.000000 : 200.000000
CAN				
PL Fabric Clocks				
<input checked="" type="checkbox"/> FCLK_CLK0	IO PLL	100	100.000000	0.100000 : 250.000000
<input checked="" type="checkbox"/> FCLK_CLK1	IO PLL	250	250.000000	0.100000 : 250.000000
<input checked="" type="checkbox"/> FCLK_CLK2	IO PLL	125	125.000000	0.100000 : 250.000000
<input type="checkbox"/> FCLK_CLK3	IO PLL	50	10.000000	0.100000 : 250.000000

IP Integrator – Advanced Clocking in Zynq

- > **Basic Clocking allows selection of desired frequency**
 - >> Tools will auto calculate nearest achievable frequency
- > **Advanced clocking allows access to clock multiply and divide values for various PLLs in the Zynq PS**
- > **Provides more control for user**

Clock Configuration [Summary Report](#)

Basic Clocking | **Advanced Clocking**

Input Frequency (MHz) CPU Clock Ratio

☐ Override Clocks

PLL Divisors	ARMPLL	DDRPLL	IOPLL
Multiplier	26	21	20
PLL Freq(MHz)	1300.000	1050.000	1000.000

Search:

Component	Clock Source	First Divisor	Second Divisor	Actual Fr...	Range(MHz)
▼ Processor/Memory Clocks					
CPU	ARM PLL	2	NA	650.000...	50.0 : 667.0
DDR	DDR PLL	2	NA	525.000...	200.000000 : 534.000...
▼ IO Peripheral Clocks					
SMC	IO PLL	1	NA	10.000000	10.000000 : 100.000000
QSPI	IO PLL	5	NA	200.000...	10.000000 : 200.000000
ENET0	IO PLL	8	1	125.000...	0.1 : 125
ENET1	IO PLL	1	1	10.000000	0.1 : 125
SDIO	IO PLL	20	NA	50.000000	10.000000 : 125.000000
SPI	IO PLL	1	NA	10.000000	0.000000 : 200.000000
UART	IO PLL	10	NA	100.000...	10.000000 : 100.000000
DCI	DDR PLL	52	2	10.096154	0.100000 : 177.000000

Zynq Resets

> Internal resets

- >> Power-on reset (POR)
- >> Watchdog resets from the three watchdog timers
- >> Secure violation reset

> PS resets

- >> External reset: PS_SRST_B
- >> Warm reset: SRSTB

> PL resets

- >> Four reset outputs from PS to PL
- >> FCLK_RESET[3:0]

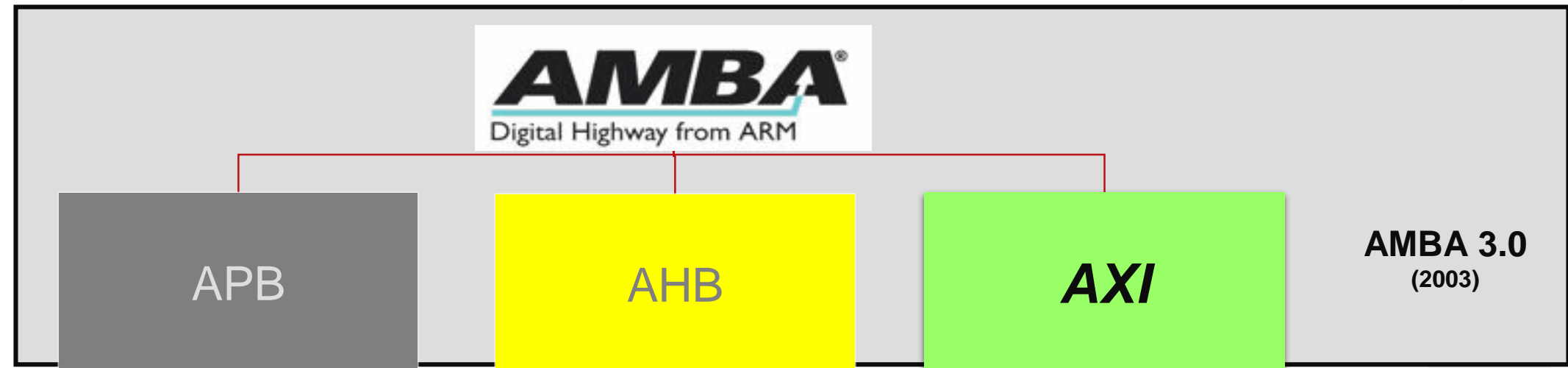


Outline

- > Zynq All Programmable SoC (AP SoC)
- > Zynq SoC Processing System (PS)
- > Processor Peripherals
- > Clock, Reset, and Debug Features
- > ***AXI Interfaces***
- > Summary



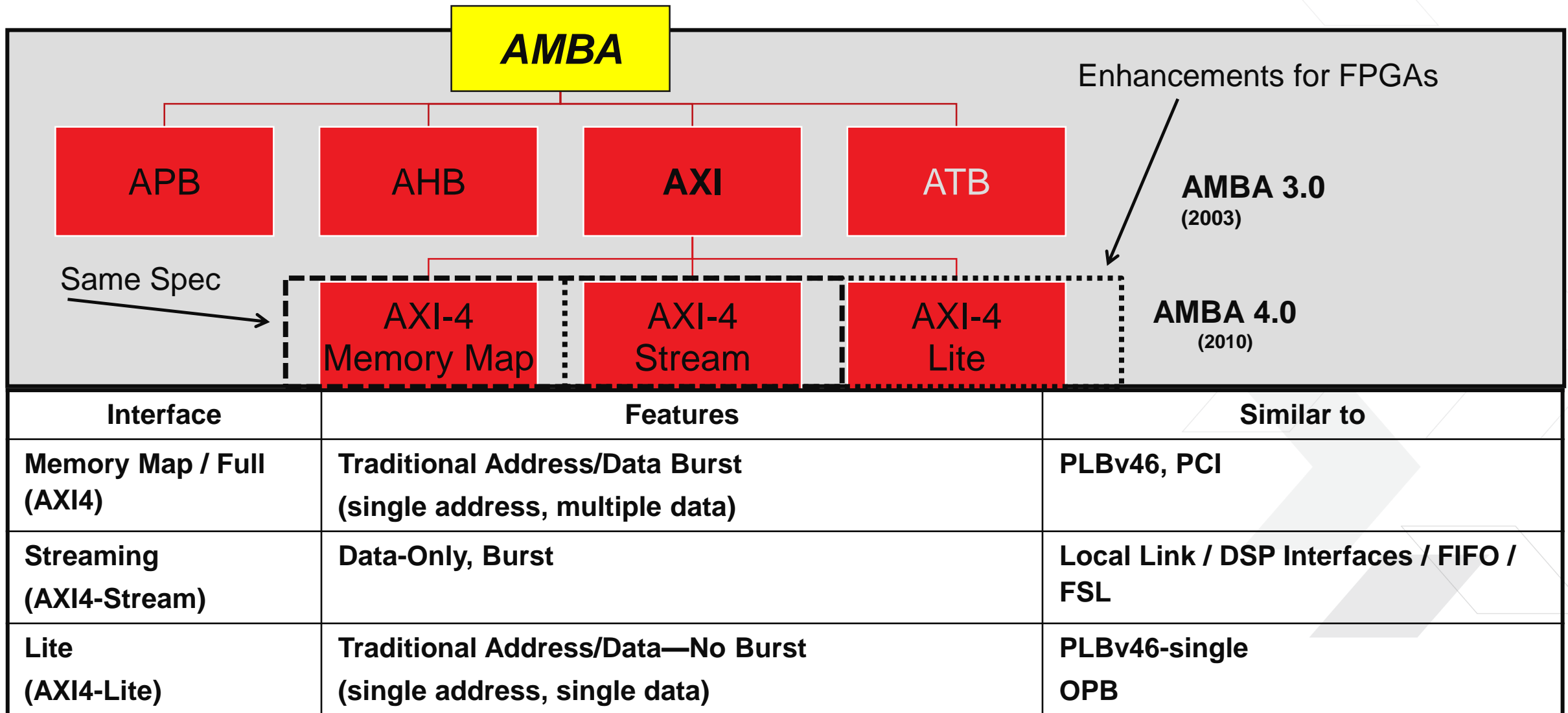
AXI is Part of ARM's AMBA



AMBA: Advanced Microcontroller Bus Architecture

AXI: Advanced Extensible Interface

AXI is Part of AMBA



Basic AXI Signaling – 5 Channels

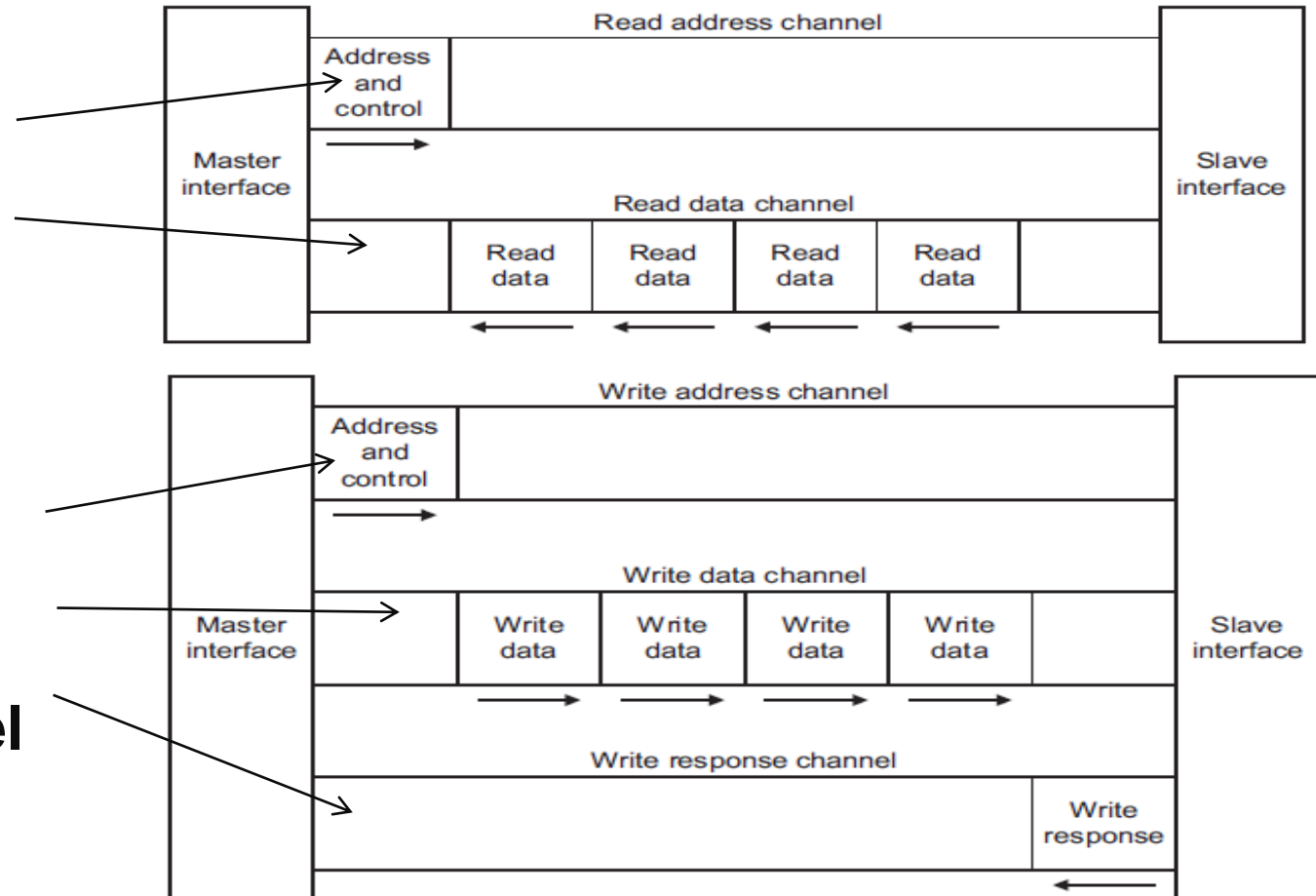
1. Read Address Channel

2. Read Data Channel

3. Write Address Channel

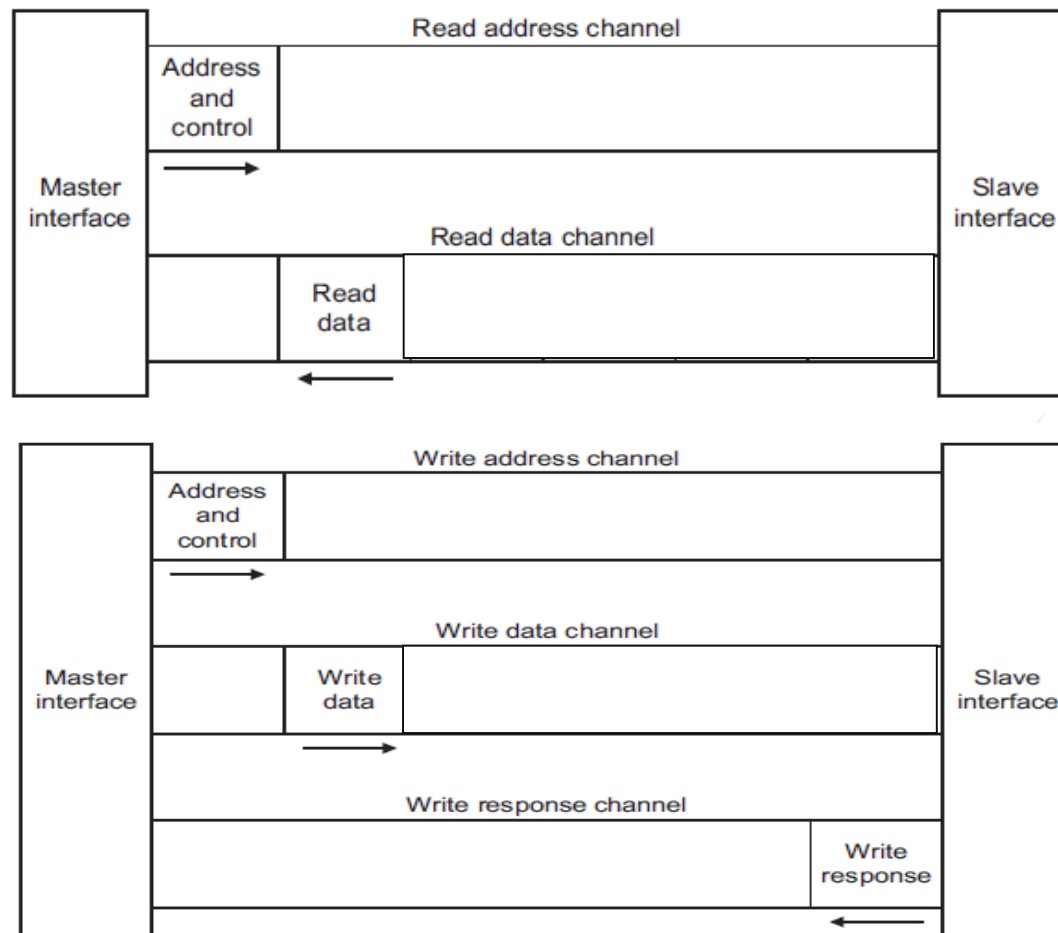
4. Write Data Channel

5. Write Response Channel



The AXI Interface - AXI4-Lite

- > No burst
- > Data width 32 or 64 only
 - >> Xilinx IP only supports 32-bits
- > Very small footprint
- > Bridging to AXI4 handled automatically by AXI_Interconnect (if needed)

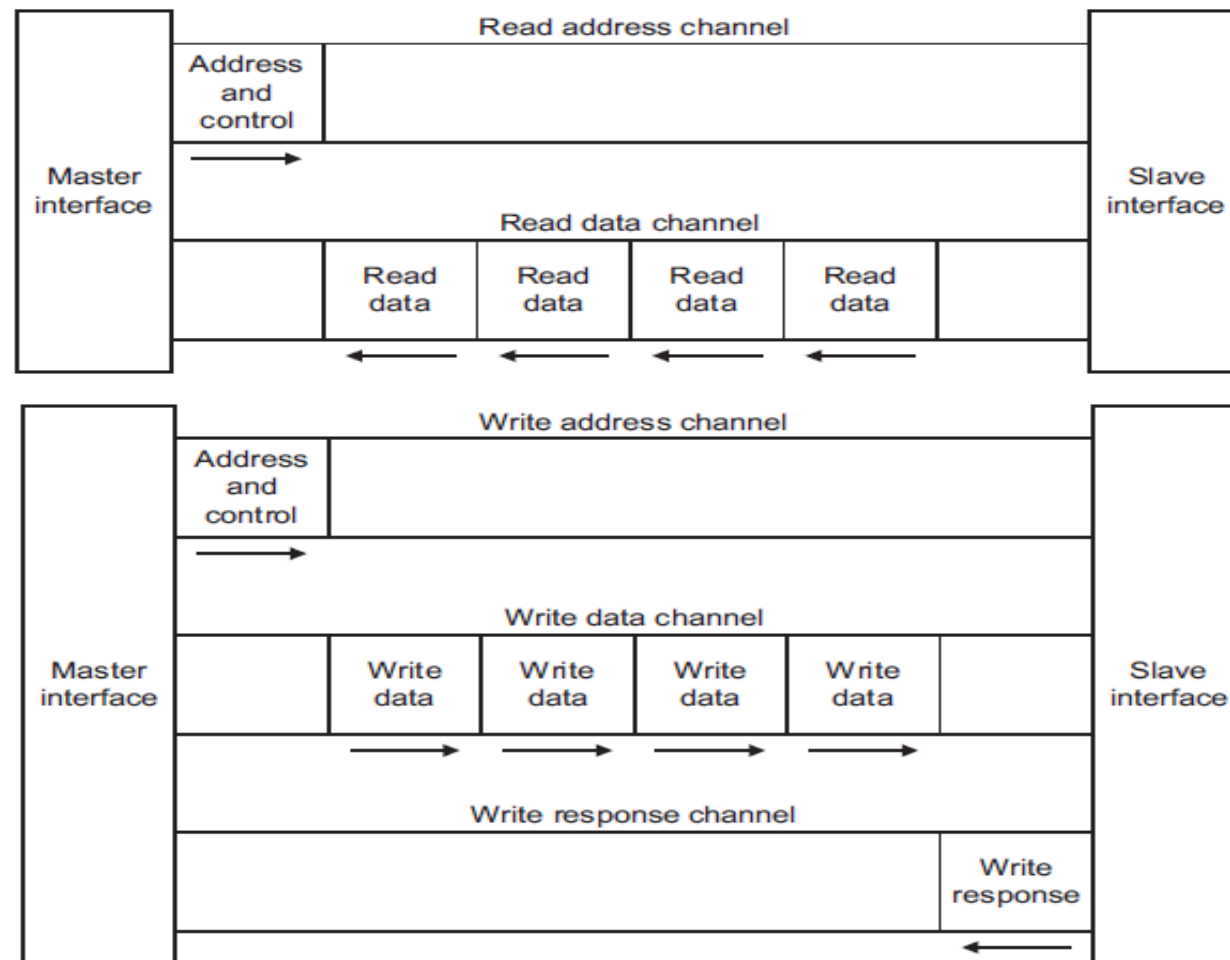


AXI4-Lite Read

AXI4-Lite Write

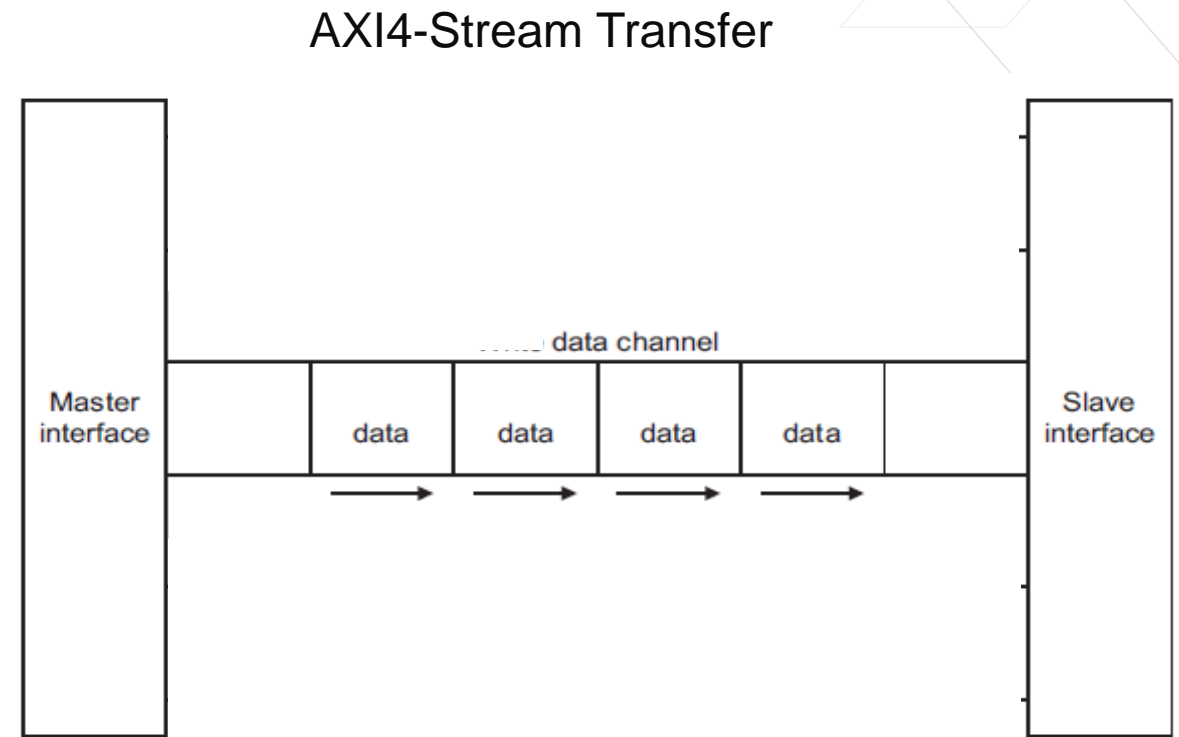
The AXI Interface - AXI4

- > Sometimes called “Full AXI” or “AXI Memory Mapped”
- > Single address multiple data
 - >> Burst up to 256 data beats
- > Data Width parameterizable
 - >> 1024 bits



The AXI Interface—AXI4-Stream

- > **No address channel, no read and write, always just master to slave**
 - >> Effectively an AXI4 “write data” channel
- > **Unlimited burst length**
 - >> AXI4 max 256
 - >> AXI4-Lite does not burst
- > **Virtually same signaling as AXI Data Channels**
 - >> Protocol allows merging, packing, width conversion
 - >> Supports sparse, continuous, aligned, unaligned streams



Streaming Applications

> May not have packets

>> E.g. Digital up converter

- No concept of address
- Free-running data (in this case)
- In this situation, AXI4-Stream would optimize to a very simple interface

> May have packets

>> E.g. PCIe

- Their packets may contain different information
- Typically bridge logic of some sort is needed



Outline

- > Zynq All Programmable SoC (AP SoC)
- > Zynq SoC Processing System (PS)
- > Processor Peripherals
- > Clock, Reset, and Debug Features
- > AXI Interfaces
- > ***Summary***



Summary

- > **The Zynq-7000 processing platform is a system on a chip (SoC) processor with embedded programmable logic**
- > **The processing system (PS) is the hard silicon dual core consisting of**
 - >> APU and list components
 - Two Cortex-A9 processors
 - NEON co-processor
 - General interrupt controller (GIC)
 - General and watchdog timers
 - >> I/O peripherals
 - >> External memory interfaces

Summary

- > **The programmable logic (PL) consists of Xilinx “7 series” technology (28nm)AXI is an interface providing high performance through point-to-point connection**
- > **AXI has separate, independent read and write interfaces implemented with channels**
- > **The AXI4 interface offers improvements over AXI3 and defines**
 - >> Full AXI memory mapped
 - >> AXI Lite
 - >> AXI Stream
- > **Tightly coupled AXI ports interface the PL and PS for maximum performance**
- > **The PS boots from a selection of external memory devices**
- > **The PL is configured by and after the PS boots**