Arria 10 GX Dev kit으로
PCI express와 DDR4 완전 정복

송영규 차장

Staff FAE, Altera Korea

ALTERA
now part of Intel

Module Game Plan

PCI EXPRESS OVERVIEW
Performance, Productivity, & Features

High-Performance DDR4 Interface
Introduce DDR4

DDR4 Demo with example design

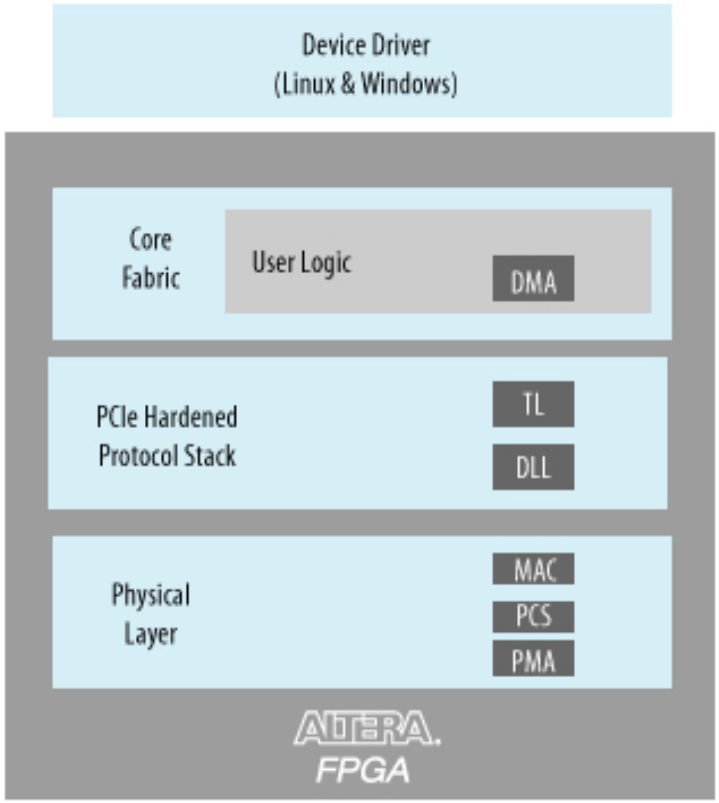
Rev Up Your Design Performance with PCI Express & DDR4 IP

Agenda

- Complete PCI Express IP Solution Overview
- Performance & Productivity You Can Expect
- PCI Express Feature Details
  - Interface Comparisons
  - Programming Options
    - Configuration via Protocol (PCIe) Initialization
    - Partial Reconfiguration over Protocol (PCIe)
- PCIe-to-DDR4 Design Introduction
  - AVMM with DMA Solution
  - Qsys Integration Snapshot

Best-in-Class PCI Express IP – Hardened Protocol Stack

- 4th generation hardened protocol stack delivers best performance & robustness
  - Highest throughput performance
  - 4 device generations
    - (65 nm / 40 nm / 28 nm / 20 nm)
  - 7 product families
- Flexible & easy to configure
  - Broad interface and configuration coverage
    - Gen1, Gen2, Gen3 support
    - x1, x2, x4, x8 lane widths
    - Root port and endpoint configurations
- Multiple user interface options
  - Avalon Streaming
  - Avalon Memory-Mapped
  - Avalon Memory-Mapped with DMA



Arria 10 High Performance PCI Express IP Solution

Best-in-Class PCI Express IP – Complete Solution

- DMA engine & drivers built for best performance & efficiency
  - 25% IOPS improvement (vs. Stratix V)
  - 3-5% throughput improvement (vs. Stratix V)
  - Scatter gather based DMA engine
  - Linux and Windows device drivers
- Configuration via Protocol (PCIe) Initialization (CvP Init)
  - For power-up programming
- Partial Reconfiguration over Protocol (PCIe) (PRoP)
  - For multiple image programming while powered
- SR-IOV feature
  - 2 Physical Functions (PFs) / 128 Virtual Functions (VFs)
  - MSI / MSI-X interrupt support
  - Expansion to 4 PFs / 4K VFs (1H'2016)

Performance & Productivity You Can Expect

- Fast solution performance
  - Highest throughput & IOPS performance vs. competition
  - Up to 6.5 GB/s DMA read & write throughput for Gen3 x8
  - Over 500K IOPS achieved for read & write directions
    - 4KB packet sizes
- Productivity
  - Linux & Windows drivers available to expedite evaluation & design-in
  - Device driver features
    - Multiple interface options to support various application models
      - IOCTL, PIO, DMA
    - Character & block device driver support
    - 3rd party, off-the-shelf, I/O benchmark tools can be used
      - Iometer & fio
    - Open source code
    - License model is Dual BSD/GPL

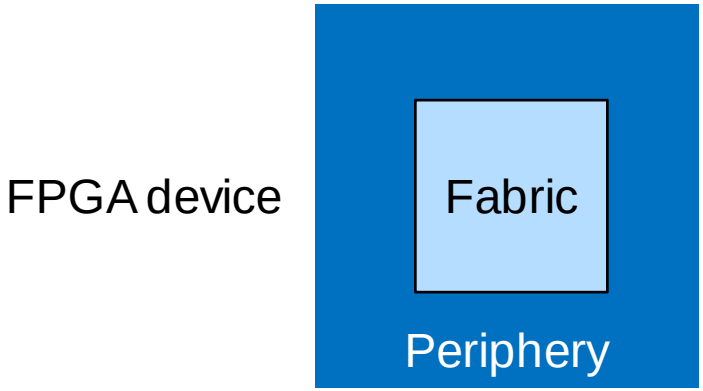
Interface Comparisons

Attribute	AVST	AVMM	AVMM with DMA
Throughput	Highest – all interfaces built for top-line performance		
Data Format	Streamed data – user to format own packets	Interface generates packet data	Interface generates packet data
Out-of-the-Box Solution	Lowest Integration ⟷ Highest Integration		
Latency (Gen3 x8)	~100 ns (HIP[1] included)	~130 ns (HIP included)	~350 ns (HIP included)
Logic Resourcing	Included in the HIP (0 ALMs / 0 Regs)	1.9k ALMs / 2.9k Regs	12k ALMs / 22k Regs (includes desc controller)
DMA Function	DMA required if application requires it	DMA required if application requires it	DMA built-in
Qsys	Qsys not required	Qsys required	Qsys required
Generation / Lane Width Support	Generation and lane width support have device family and interface dependencies. See the following webpage for details: PCI Express Protocol		

1. *HIP stands for Hardened IP (PCIe).*

Configuration via Protocol (PCIe) Initialization Introduction

- Shortened name – CvP Init
- Flash / EEPROM is used to program the FPGA periphery which includes the PCIe Hardened IP block upon power-up
- CvP Init enables a system to communicate with the hardened PCIe IP block without having the FPGA fabric programmed
  - Ensured via entering L0 (Link Active) within 100 ms window
- CvP Initialization functionality is identical across multiple device families
  - Cyclone V, Arria V, Stratix V, and Arria 10 devices

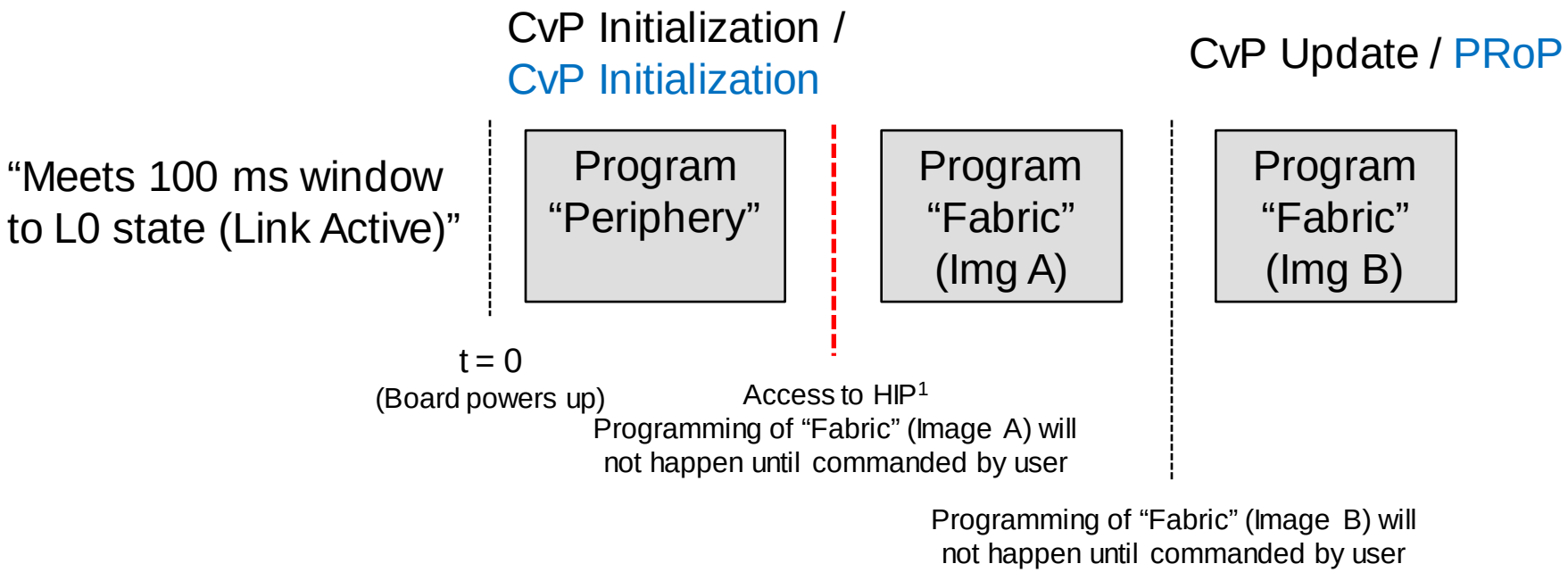


Partial Reconfiguration over Protocol (PCIe) Introduction

- Shortened name – PRoP
- PRoP enables a user to redefine & update the FPGA 'fabric' to support a different functionality (an FPGA image)
- PRoP replaces CvP Update in next generation family (Arria 10)
- Simplified version of partial reconfiguration to emulate CvP Update
- PRoP provides more flexibility and robustness
  - Peripheral components aren't impacted
    - E.g. DDR maintains state / operation
  - XCVR and DDR re-calibrations are not necessary
  - Precise and multiple portions of the design (generated "personas") can be updated, if necessary, versus updating the whole fabric

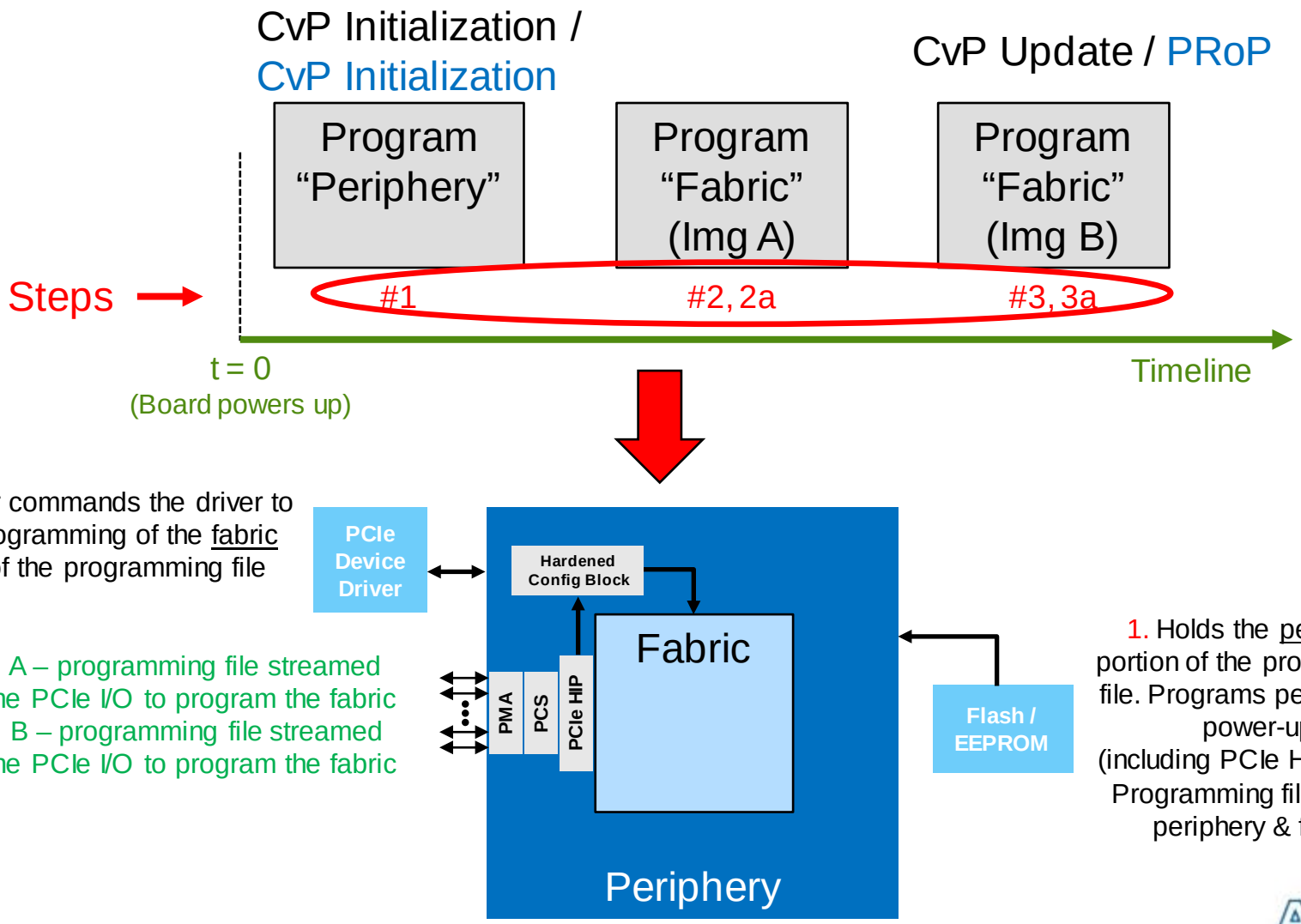
Illustration of Programming Options

- Stratix V programming modes / Arria 10 programming modes



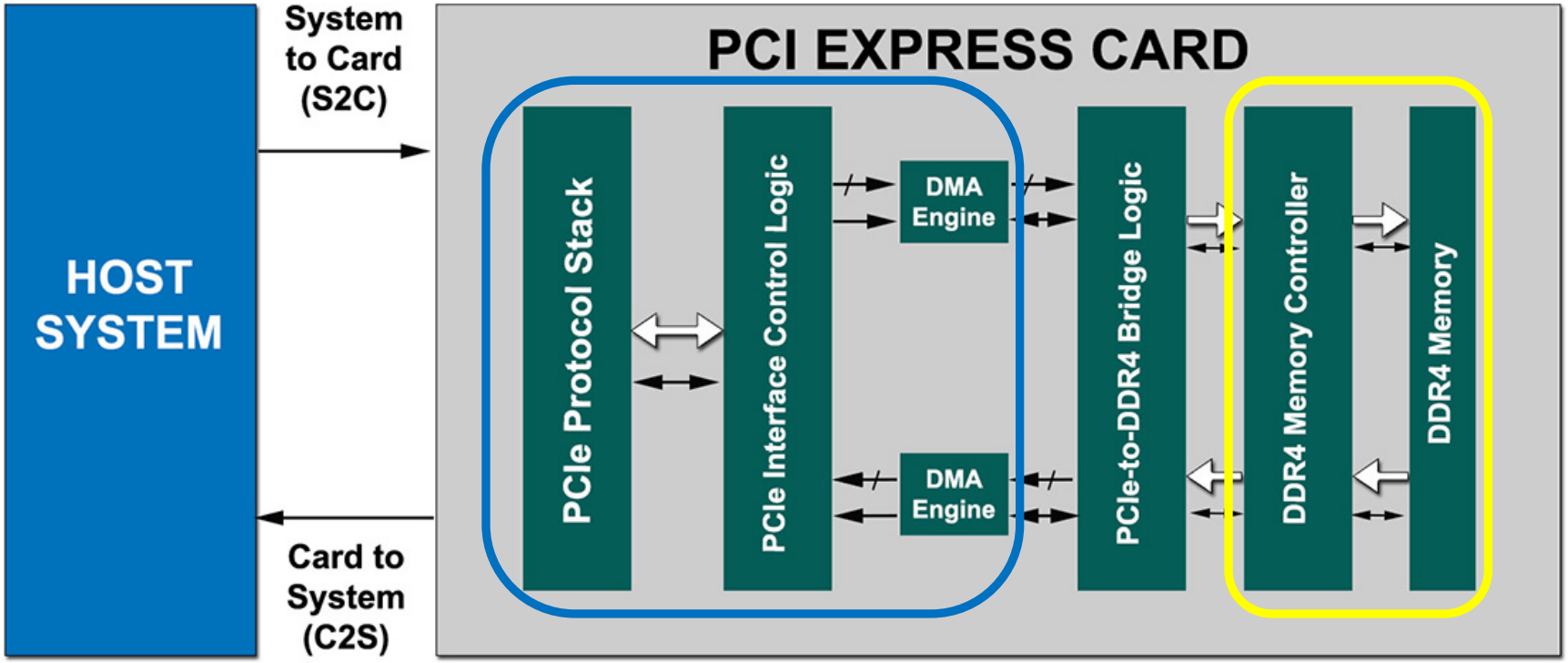
1. HIP is in L0, but transmission of packets should wait until fabric is loaded.

How Programming Flow Looks in Hardware



PCIe-to-DDR4 Block Diagram

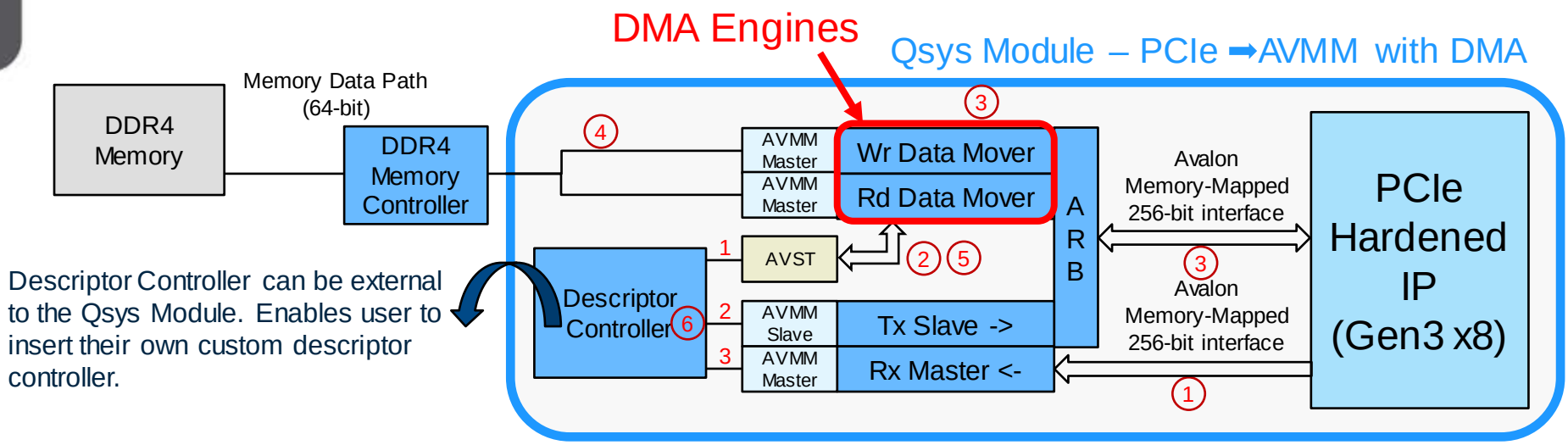
- Designed for optimal system performance



= PCI Express Design Portion of Bridge Solution

= DDR4 Design Portion of Bridge Solution

System Block Overview (AVMM with DMA)



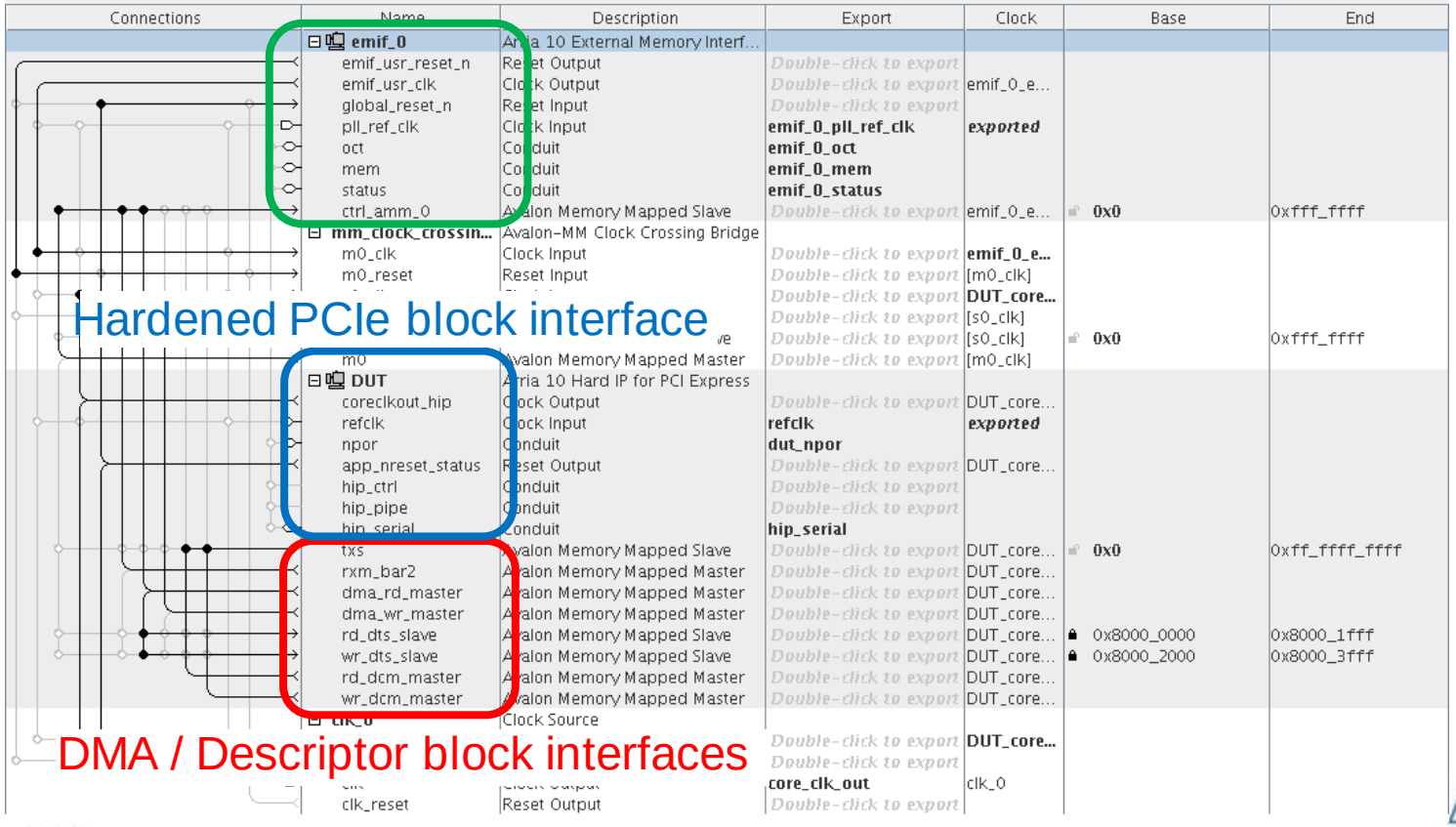
- Data movers (aka DMA engines – the “workhorses”)
  - Efficiently moves bursts of data from / to destination (4) and notes completion (5)
  - Decodes the descriptors (2)
  - Encodes and decodes PCIe TLP packets (3)
  - Supports on-chip and off-chip memory configurations
- Descriptor controller (the “brains”)
  - Device driver delivers descriptor(s) to descriptor controller (1)
  - Directs data movers in regards to providing destination / source / direction & amount of data (2)
  - Alerts host with interrupt, acknowledgement of descriptor being completed (6)
  - Modifications to existing or custom descriptor controller can be designed for application need

1 - Data Mover Control
2 - DMA Status to Host
3 - Host Control of DMA

Qsys Integration of Snapshot

- Snapshot of Qsys integration
  - Hardened PCIe block, DMA, Descriptor Controller, and DDR4 external memory connectivity

Ext. memory block interface



Hardened PCIe block interface

DMA / Descriptor block interfaces

Implementing a High-Performance DDR4 Interface in Arria10

Agenda: Implementing DDR4

- Overview
- Implementation challenges
- Addressing the challenges
- Arria 10 memory interface architecture
- Implementing Arria 10 DDR4 interfaces
- Summary

DRAM Technology Comparison(1)

	DDR3	DDR4
Voltage	1.5 V / 1.35 V	1.2 V
DQ Bus	SSTL15 CTT	POD12
Strobe	Bi-directional differential	Bi-directional differential
Strobe Configuration	Per byte	Per byte
READ Data Capture	Strobe based	Strobe based
Data Termination	VDDQ/2	VDDQ
Address/Command Termination	VDDQ/2	VDDQ/2
Burst Length	BC4, 8	BC4, 8
Number of Banks	8	16
Bank Grouping	No	4
On-Chip Error Detection	No	Command / address parity
		CRC for data bus
Configuration	x4, x8, x16	x4, x8, x16
Package	78-ball / 96-ball FBGA	78-ball / 96-ball FBGA
Data Rate (Mbps/Pin)	800 – 2,133	1,600 – 3,200+
Component Density	1 GB – 8 GB	2 GB – 16 GB
Stacking Options	DDP, QDP	Up to 8H (128-GB stack); single load

Notes: 1. Data from www.micron.com and www.jedec.org

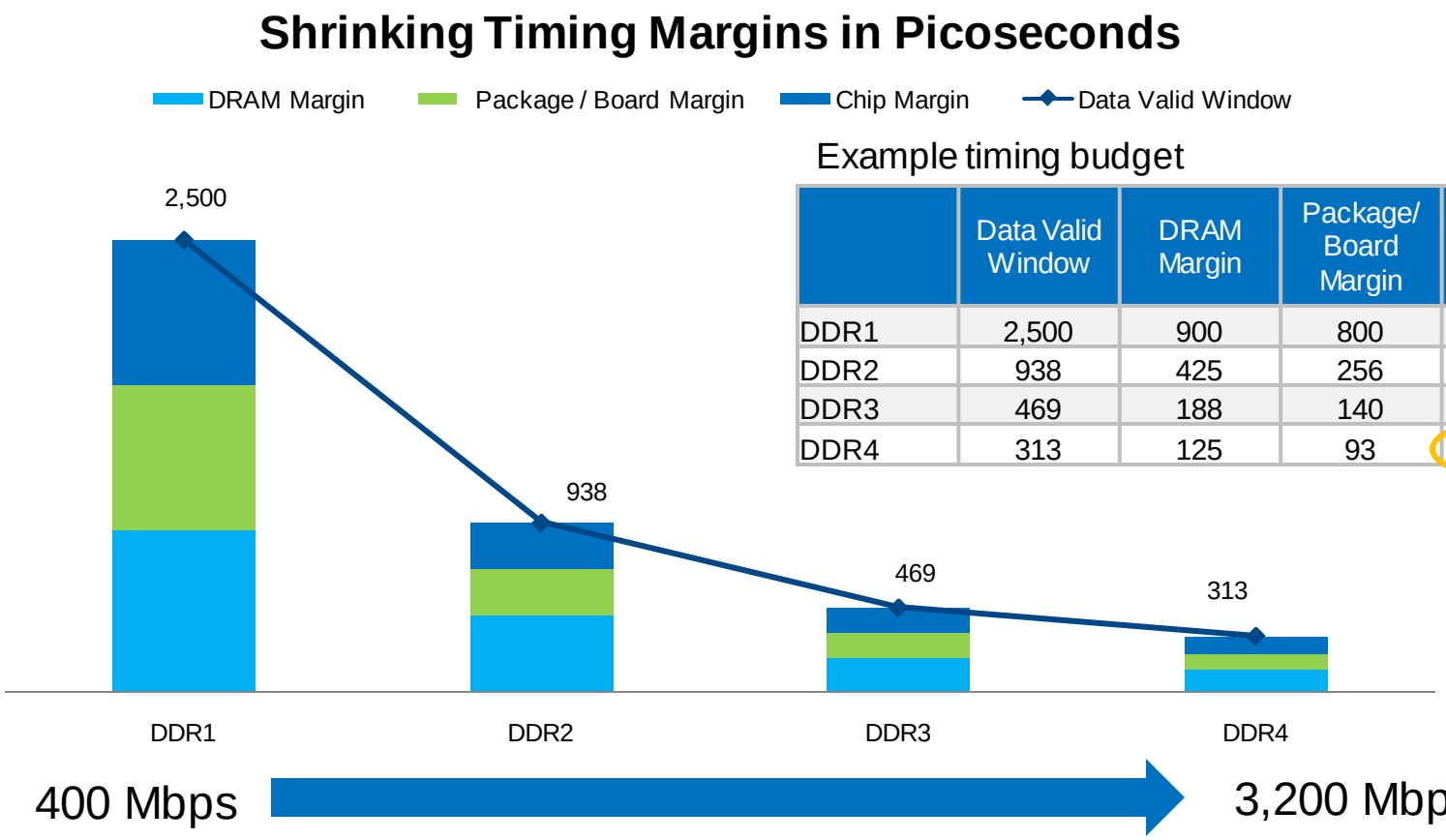
DDR4 Power Savings Features

- DDR4 voltage is 1.2 V (up to 40% savings)
  - Lower voltage than DDR3 (1.5 V)
  - On-die VREF
  - Pseudo-open drain I/Os
- Manages refreshes (up to 20% savings)
  - Based on temperature
    - New DDR4 low-power auto self-refresh (LPASR) capability
      - Changes refresh rate based on temperature
  - Only refreshes parts of array that is in use
    - Controller must allow fine-granularity refresh based on memory utilization
- Supports data bus inversion
  - Limits number of signals transitioning, reducing simultaneous switching output (SSO) and saving power

Creating a Data Valid Window

It is all about calibration

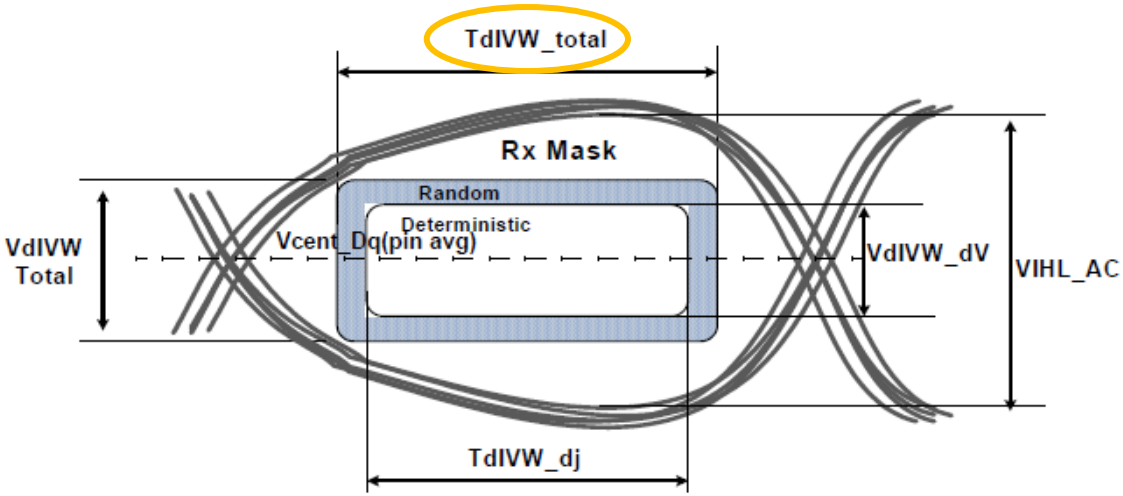
Timing Margins Are Shrinking



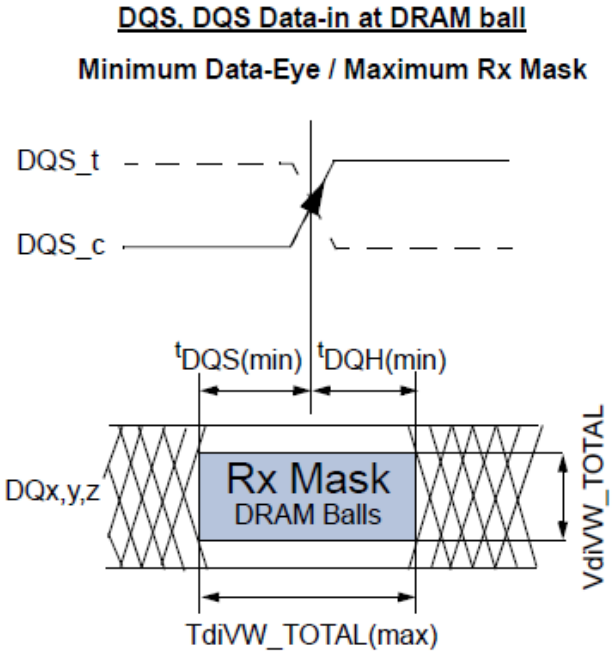
Example timing budget

	Data Valid Window	DRAM Margin	Package/ Board Margin	Chip Margin
DDR1	2,500	900	800	800
DDR2	938	425	256	256
DDR3	469	188	140	140
DDR4	313	125	93	93

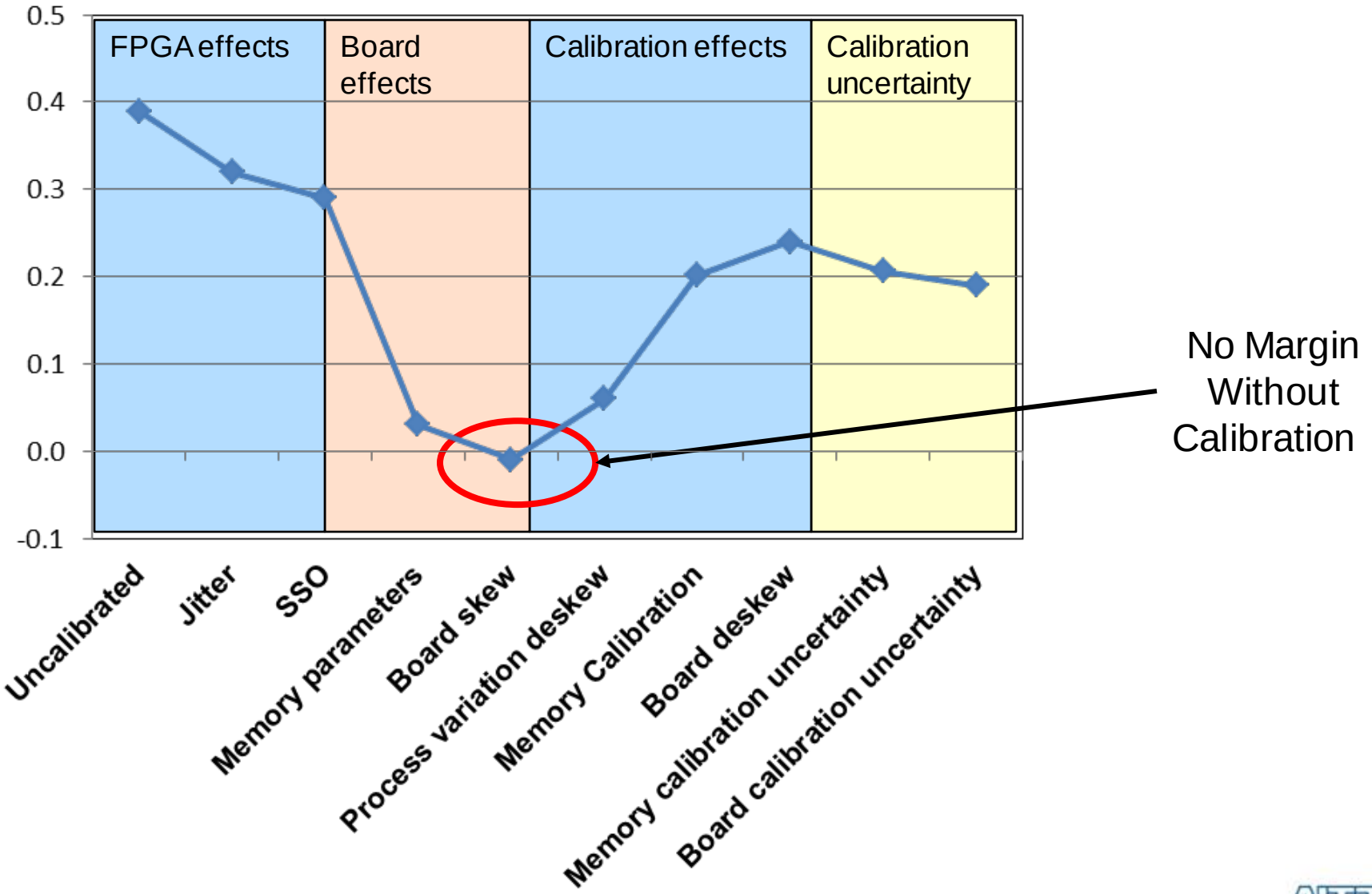
DDR4 JEDEC Definition of the Data Valid Window



DQ Receiver(RX) compliance mask

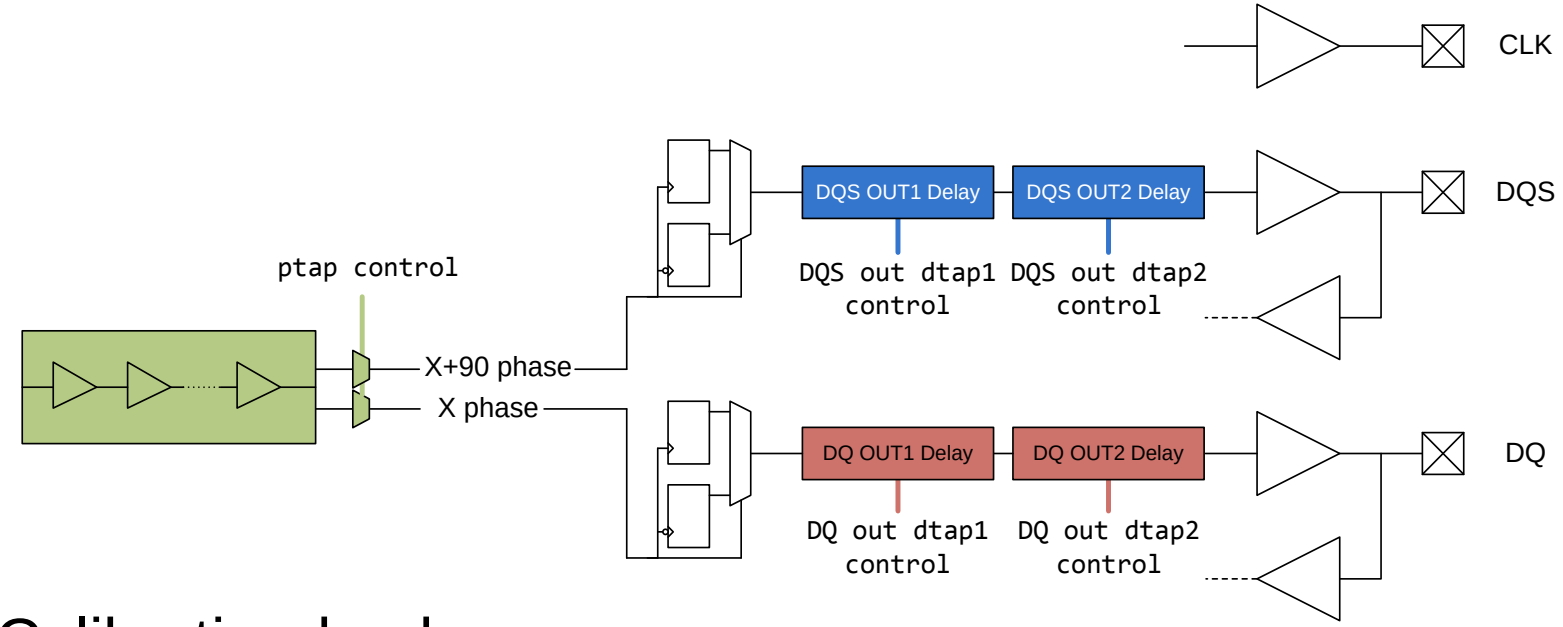


Calibration Is Critical to Offset Shrinking Margins



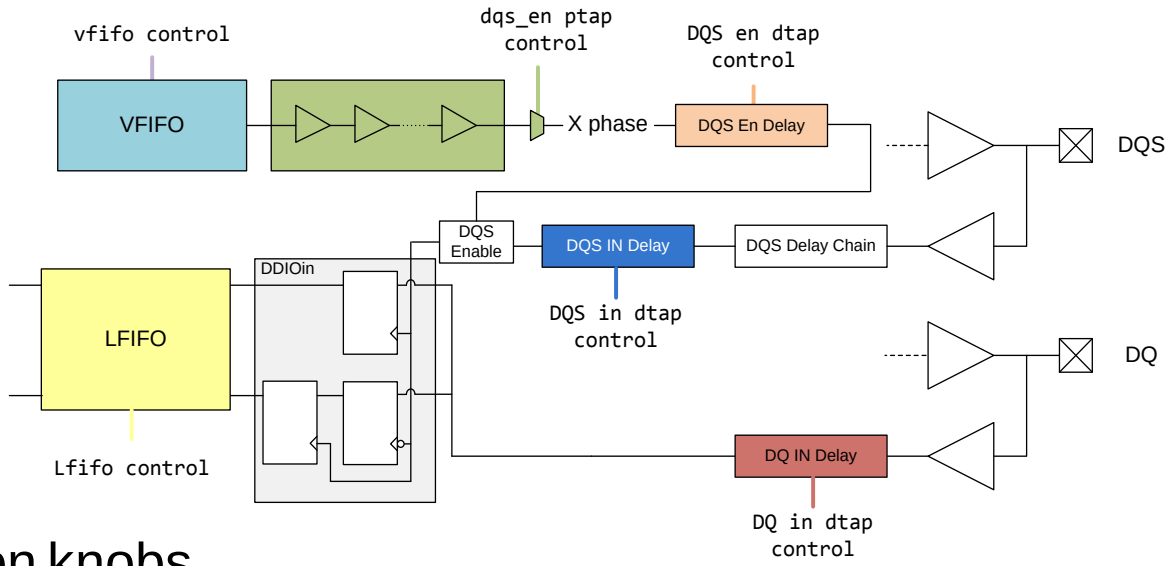
Arria 10 Addresses the DDR4 Challenges

High-Level Output Topology



- Calibration knobs
  - **DQ-out1** and **DQ-out2 delay** : Control the delay applied to outgoing DQ pins
  - **DQS-out1** and **DQS-out2 delay** : Control the delay applied to outgoing DQS pins
  - **Write leveling output** : Changes the delay on both DQ and DQS relative to the memory clock-in phase taps

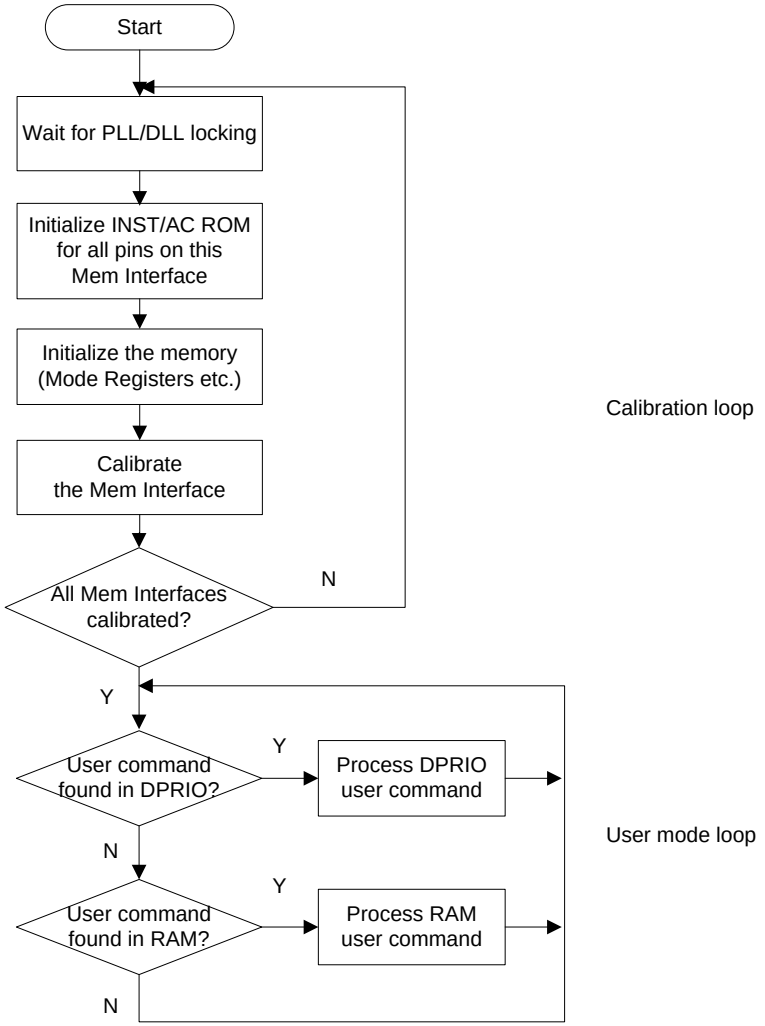
High-Level Input Topology



- Calibration knobs
  - **DQ-in delay**: Control the delay applied to incoming DQ pins
  - **DQS-in delay**: Control the delay applied to incoming DQS pins
  - **LFIFO control**: Controls number of cycles after read command that data is read out of the LFIFO
  - **DQS-En phase**: Control the delay on DQS En in phase taps
  - **DQS-En delay**: Control the delay on DQS En in dtaps
  - **VFIFO**: Adjusts the delay in cycles applied to controller-provided DQS burst signal to generate DQS enable

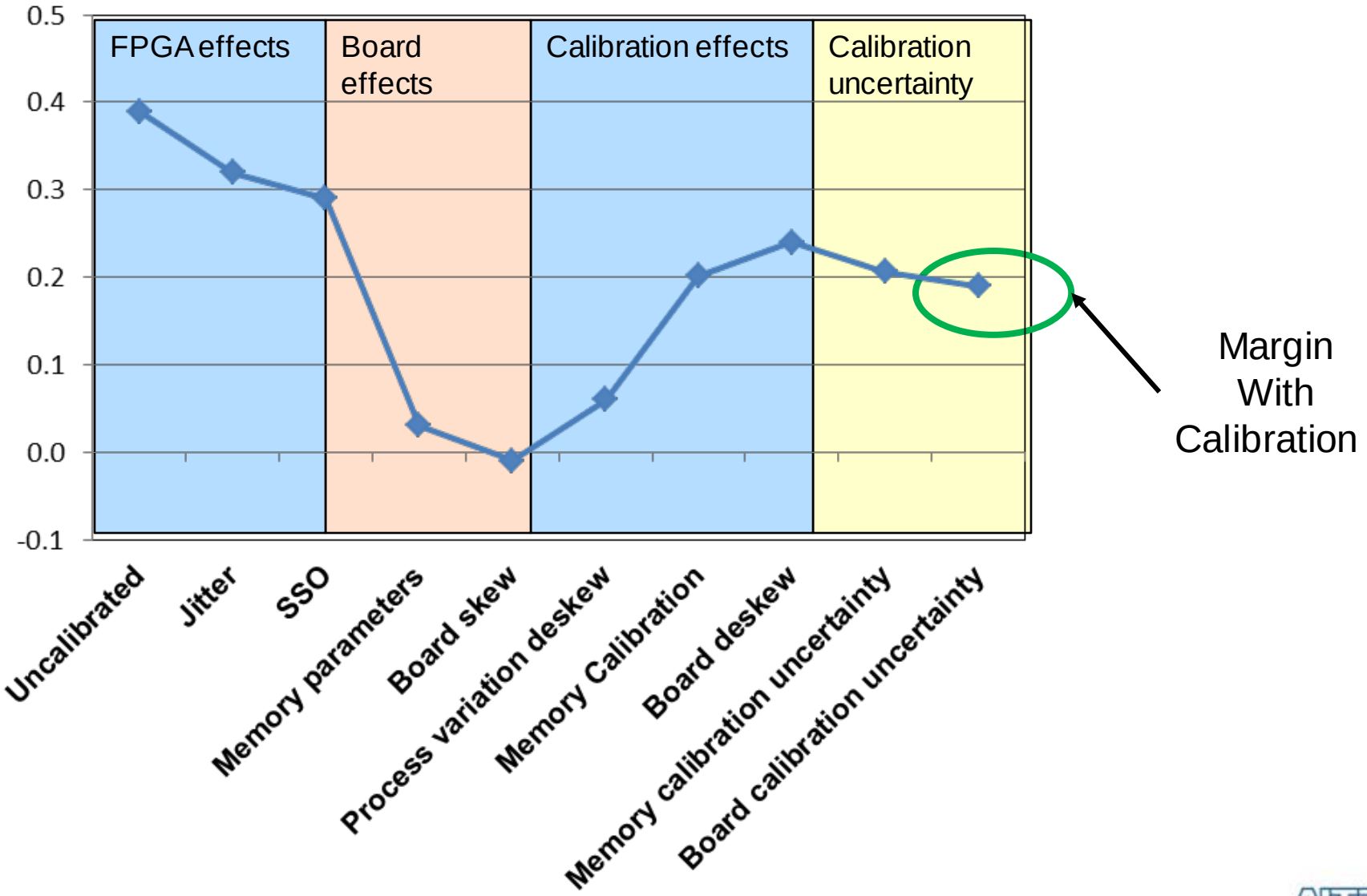
Calibration Stages

- DQS-enable calibration
  - Calibrate DQS enable (delayed read data valid) relative to DQS
- Post-amble tracking
  - Track DQS-enable across temperature variation
- Read data deskew
  - Calibrate DQS relative to read command (read leveling)
  - Calibrate DQ versus DQS (per-bit deskew) for reads
- LFIFO training
  - Calibrate LFIFO delay cycles (read latency)
- Write leveling
  - Calibrate DQS and DM to write command (write leveling)
- Write data deskew
  - Calibrate DQ versus DQS (per-bit deskew) for writes
- Address/command training (leveling and deskew)
  - Calibrate CS, CAS, RAS, and ODT versus memory clock
- VREF training (FPGA and memory)
  - Calibrates receiver voltage threshold (for DDR4 with pseudo open drain DQs)



Calibration done by hard IP

Calibration Is Critical to Shrinking Margins



Margin obtained via hard IP calibration

Arria 10 Memory Interface Overview

Arria 10 Innovative Memory Architecture

- **Hard memory PHY and controller**
  - Advancing beyond Arria V and Cyclone V
  - Ping Pong-PHY for maximizing pin usage
    - Shared address / command
- **Increased flexibility**
  - Innovative architecture to mimic soft controller
  - **More** configuration than a single controller
    - One controller in every IO Bank
  - **More** widths (144-bit)
  - **More** depths (multi-ranks)

Higher bandwidth, higher efficiency, and lower latency

Hardened Memory Interface

Easy to use

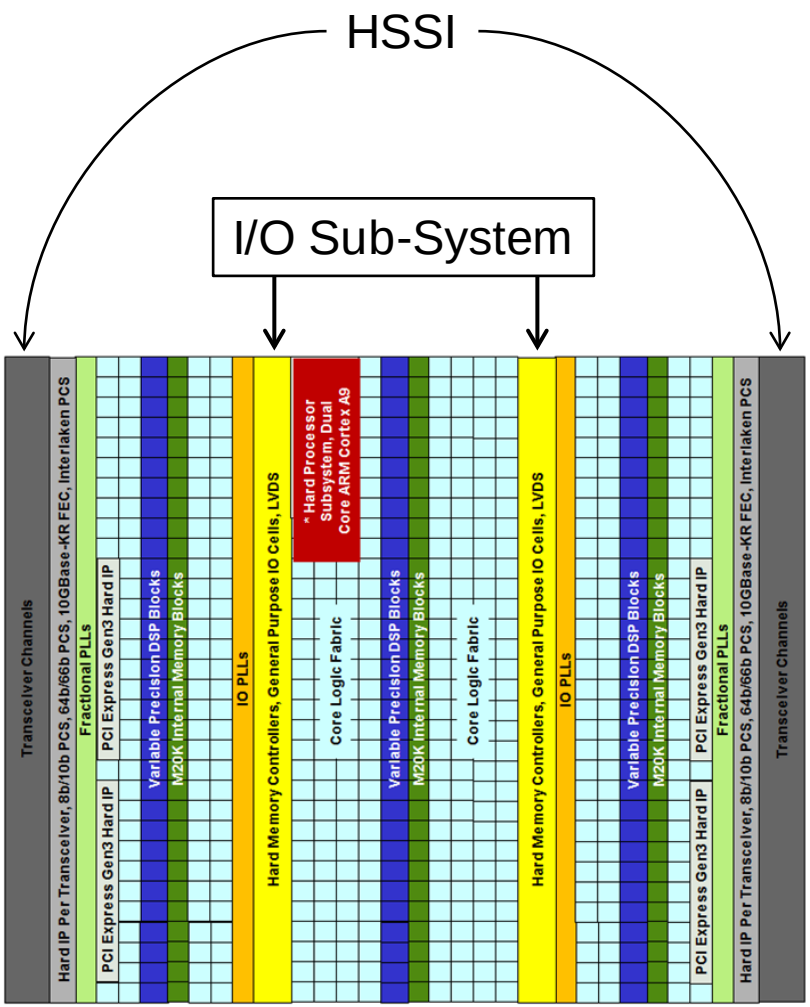
- **Guaranteed timing closure**
  - No issue with placement or routing because it's a hardened ASIC block inside the device
- Saves logic and memory resources
  - No need to use any core resources to implement DDR3/4
- Consistent performance
  - No seed variation or fitter variation. Same circuit every time in the ASIC block

Fast rollout

- DDR4 2,666 Mbps demo 4 weeks after silicon
- Maximum speed on all memory interfaces 12 weeks after silicon
- Guarantees schedule for software and characterization

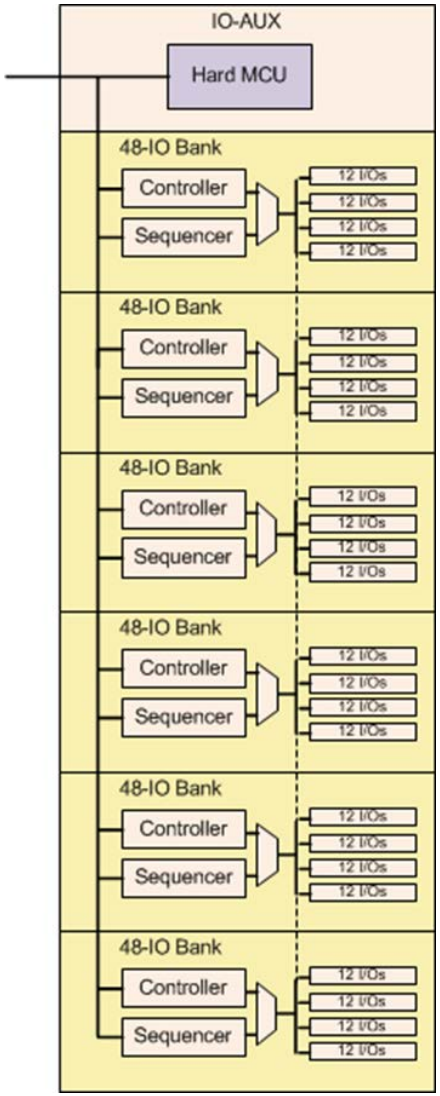
Performance + Innovative Architecture + Ease-of-use = Time to market

Arria 10 I/O Sub-System



- Organized in columns of
  - I/O banks (groups of 48 I/Os)
  - I/O Aux (one per column)

Arria 10 I/O Sub-System



- Supports
  - General Purpose I/Os (GPIO)
    - I/O Registers & I/O Buffers
  - PLLs
    - IOPLL for EMIF and user logic
  - External Memory Interfaces (EMIF)
    - Hard Memory Controller
    - Hard PHY
    - Hard MCU / Calibration logic
    - DLL
  - On-chip Termination Control (OCT)
  - LVDS or 2.5V/3V

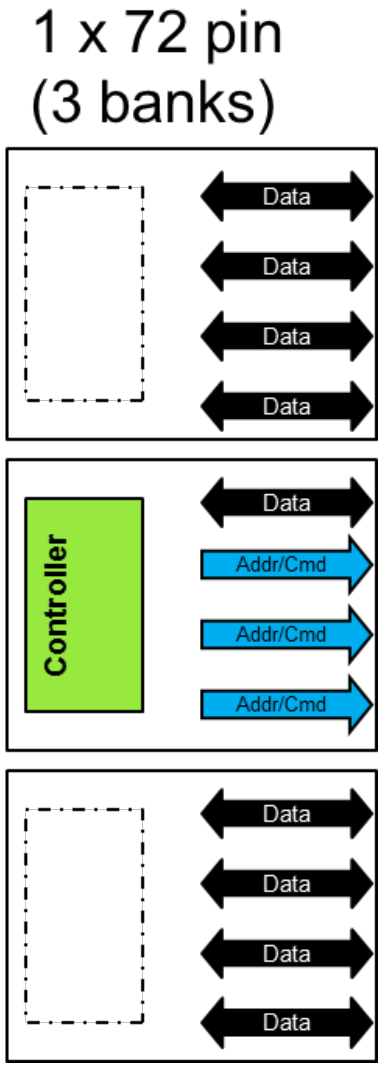
Arria 10 I/O Banks

- LVDS / DDR I/O
  - Supports LVDS and single ended up to 1.8V
  - LVDS pairs configurable as input or output
  - Delivers highest performance
- 3V / DDR I/O
  - Supports single ended up to 3V
  - 3.3V input tolerant
  - Limited to 533 MHz performance
- JTAG interface
  - Limited to 1.8V

I/O Standard	LVDS / DDR I/O	3V / DDR I/O
3V LVTTL/ CMOS	N	Y
2.5V CMOS	N	Y
1.8V CMOS	Y	Y
1.5V CMOS	Y	Y
1.2V CMOS	Y	Y
LVDS	Y	N

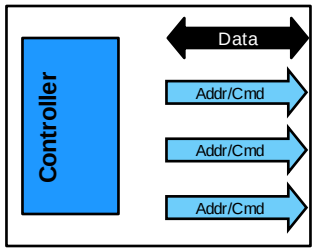
Arria 10 EMIF Example – 72bit Interface (3 banks)

- Controller & sequencer only drive Address and Command (A/C) lines to I/O lanes in same bank
  - Can drive data groups to banks above and below
  - A/C pins will have fixed locations in bank
    - Similar to hard interface in Arria V & Cyclone V

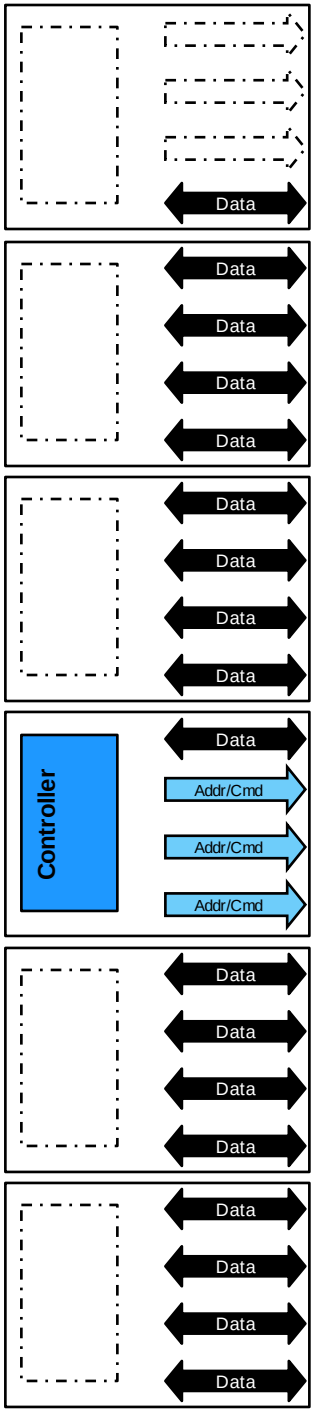


Additional Examples

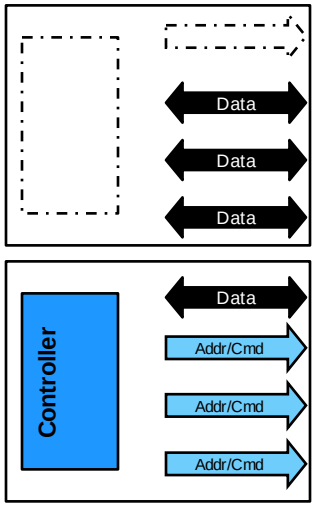
1 x 8 pin
(1 banks)



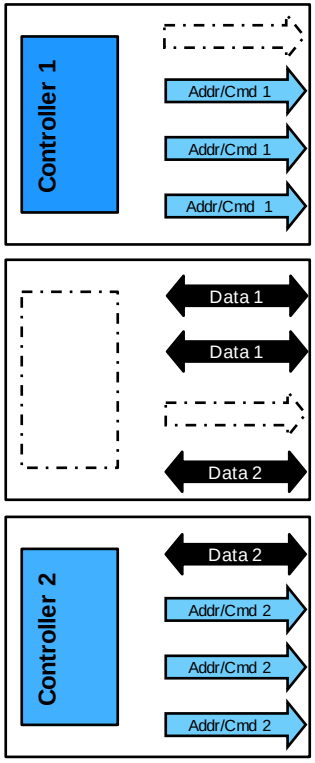
1 x 144 pin
(6 banks)



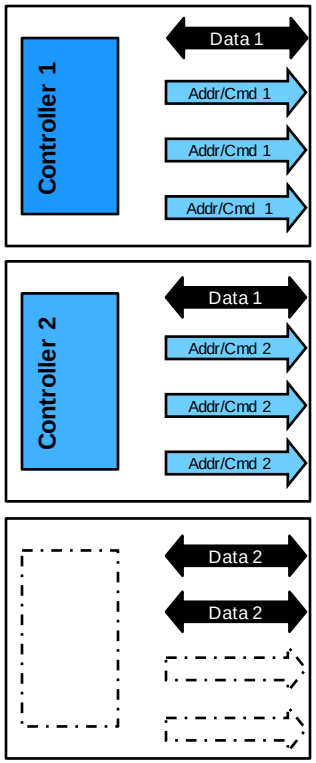
1 x 32 pin
(2 banks)



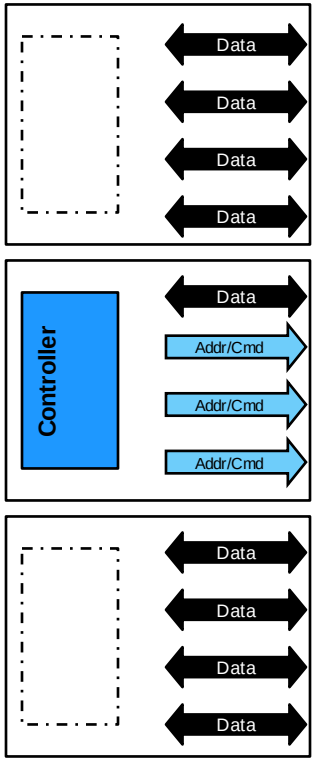
2 x 16 pin
(3 banks)



2 x 16 pin
(3 banks)



1 x 72 pin
(3 banks)



NOTE locations of Address / Command versus controller

Key Controller Features: General

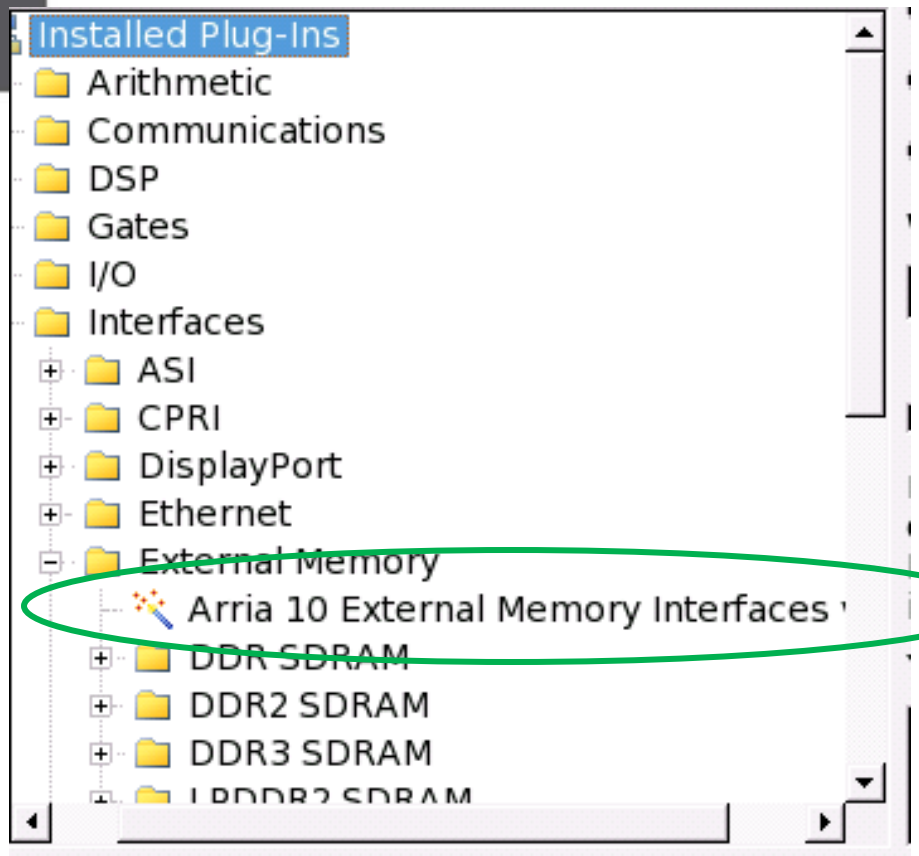
- Ping-Pong PHY (DDR3/4 only)
- Avalon-MM, Avalon-ST, or AXI interface
- Half-rate or Quarter-rate operation
- 8 bits to 144 bits (in 8-bit step size)
- Up to 4 ranks (logical ranks)
- DQS tracking
- Efficiency optimization:
  - Burst adaptor
  - Open page policy (default: closed page)
    - Controller will intelligently keep row open based on incoming traffic
  - Preemptive bank management
  - Data reordering
  - Additive Latency (AL)
  - Quasi-1T for half-rate, Quasi-2T for quarter-rate address/command (allows two A/C per controller clock)
- User options
  - User-requested priority
  - User-controller refresh
  - Low power modes (power down, self-refresh, auto-power down)
  - Partial array self-refresh (PASR), panic refresh
- "PHY-only" by-pass mode for custom controller

Key Controller Features: DDR4

- Mode resister set/read (MRS)
  - Also multi-purpose register (MPR) read
- Bank group support
  - Support different timing parameters between bank groups
- Data bus CRC, BER testing
- Address/command bus parity check & alert
- Fine-granularity refresh
- Low power auto self-refresh (LPASR)
- Gear down mode
  - Fixed configuration only, can't change on-the-fly

Arria 10 EMIF IP:
GUI & Generation

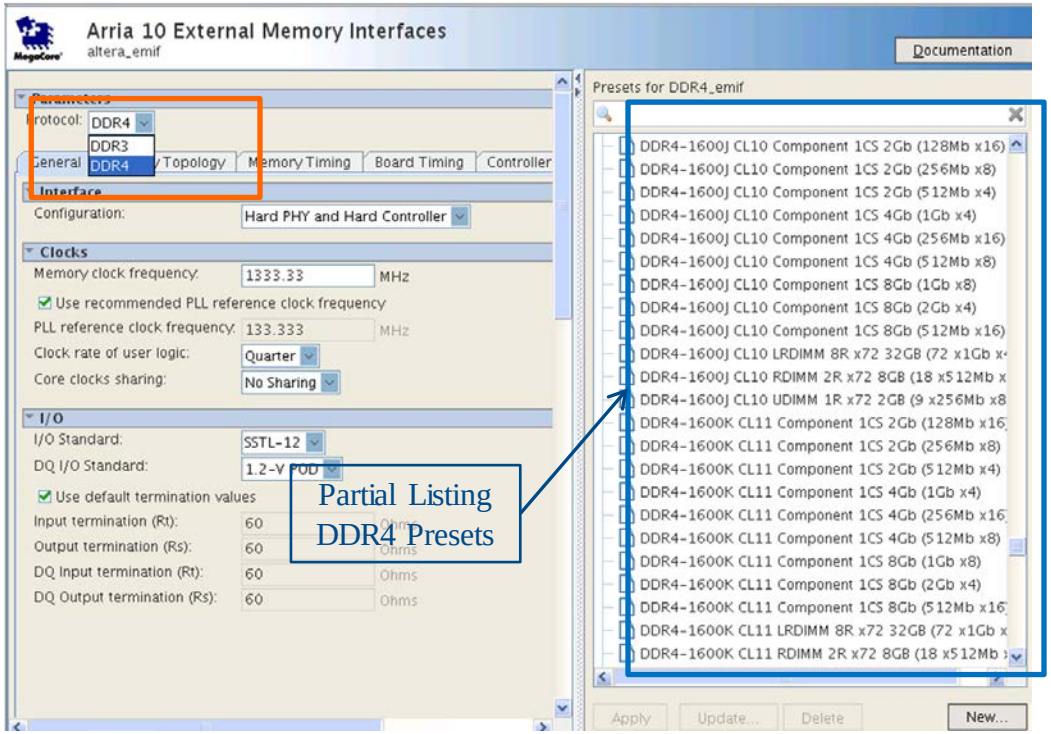
Selecting the Arria 10 Memory Interface IP



Inside of the MegaWizard in Interfaces -> External Memory select: Arria 10 External Memory Interfaces

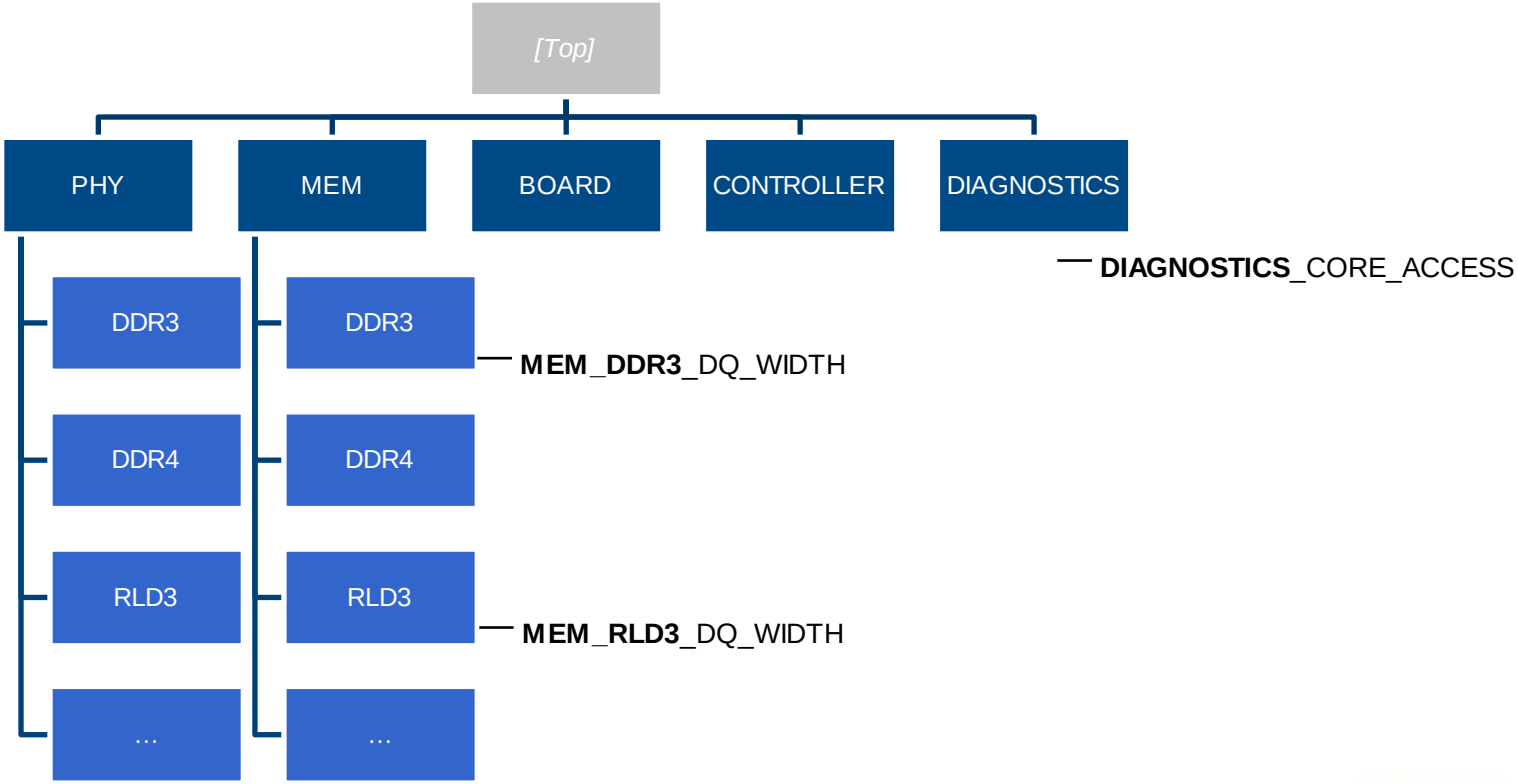
New EMIF IP for Arria 10 Devices

- Redesigned and rewritten
  - Similar look & feel to 28nm
- Single entry point for all memory protocols
  - Protocol is one of the parameters
    - 28nm: one IP per protocol
- Design output:
  - Generates IP in < 20s
  - Generates well-documented IP



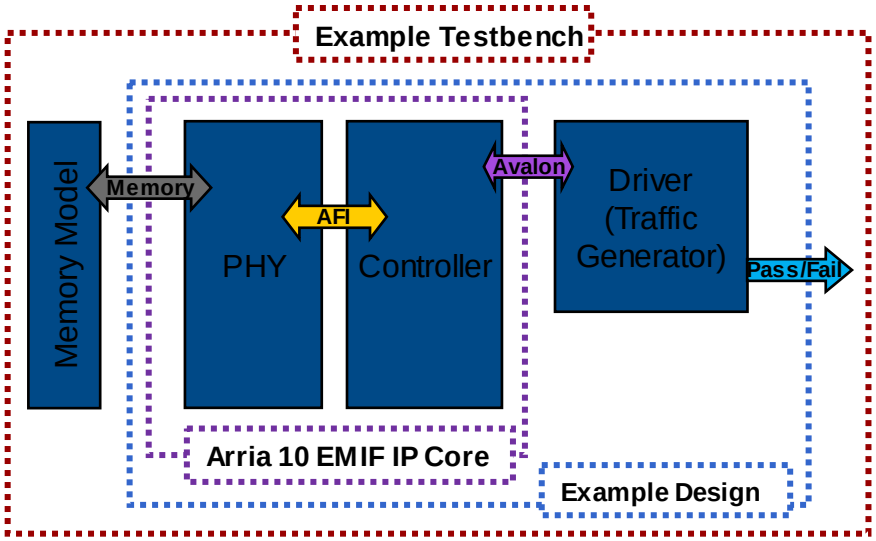
Each Protocol Has Own Set of Parameters

- Parameters are divided into namespaces
  - Namespaces are partitioned based on functionality and protocol
  - Each protocol maintains its own set of parameters
    - Will be retained even if you switch to other protocol



Arria 10 EMIF Generation Output

- Arria 10 EMIF IP core generates:
  - Synthesis fileset for compilation (Altera EMIF IP core only)
  - Simulation fileset for simulation (Altera EMIF IP core only)
  - Example design for synthesis
    - Includes traffic generator
  - Example design for simulation
    - Includes traffic generator and Altera memory model
- Top-level IP is clear-text Verilog
  - Optional VHDL wrapper
  - Lower-level hardware blocks have encrypted simulation models
- Includes scripts (QIP, SDC)
- Readme "datasheet"
  - Ports & parameters w/ descriptions
- Example design is dynamic
  - Matches IP configuration



Arria 10 EMIF IP: Simulation

EMIF Simulation Use Cases

- Two main *use cases* for simulation:

1. System-level Simulation
   - Purpose: Focus on user logic
   - EMIF IP Core: Just needs to provide interface to store/retrieve data
     - Could be an abstracted model
     - Latency and efficiency of interface need to be accurate
     - User is not interested in details of core, nor calibration
   - Simulation time expectation: Fast

2. EMIF-specific Simulation
   - Purpose: Explore the details of EMIF IP core, calibration
   - EMIF IP Core: Must be accurately represented versus hardware
     - User may want to see individual transactions
     - Visibility into calibration algorithm and stages may be required
     - This level of detail can be used to support debug
   - Simulation time expectation: Slow

EMIF Simulation Use Cases: Arria 10 Solution

- Arria 10 simplifies simulation modes
  - And satisfies the two key use cases

1. System-level Simulation → Use "simulation" fileset
   - EMIF IP is a treated as a black box, and may be abstracted if needed for simulation time
   - Calibration will be skipped; jump right to user mode
   - Multiple interfaces are independent (i.e. no merging into I/O column)
   - Latency and efficiency of interface will remain accurate

2. EMIF-specific Simulation → Use "synthesis" fileset
   - Simulate the IP used for synthesis
   - Will enable full calibration (note: this will be very slow)
   - Multiple interfaces calibrate independently (i.e. no merging into single IOAux yet)
   - Optionally, can simulate post-fit netlist
     - Multiple interface will calibrate sequentially

Arria 10 EMIF IP:
Compilation & Timing Closure

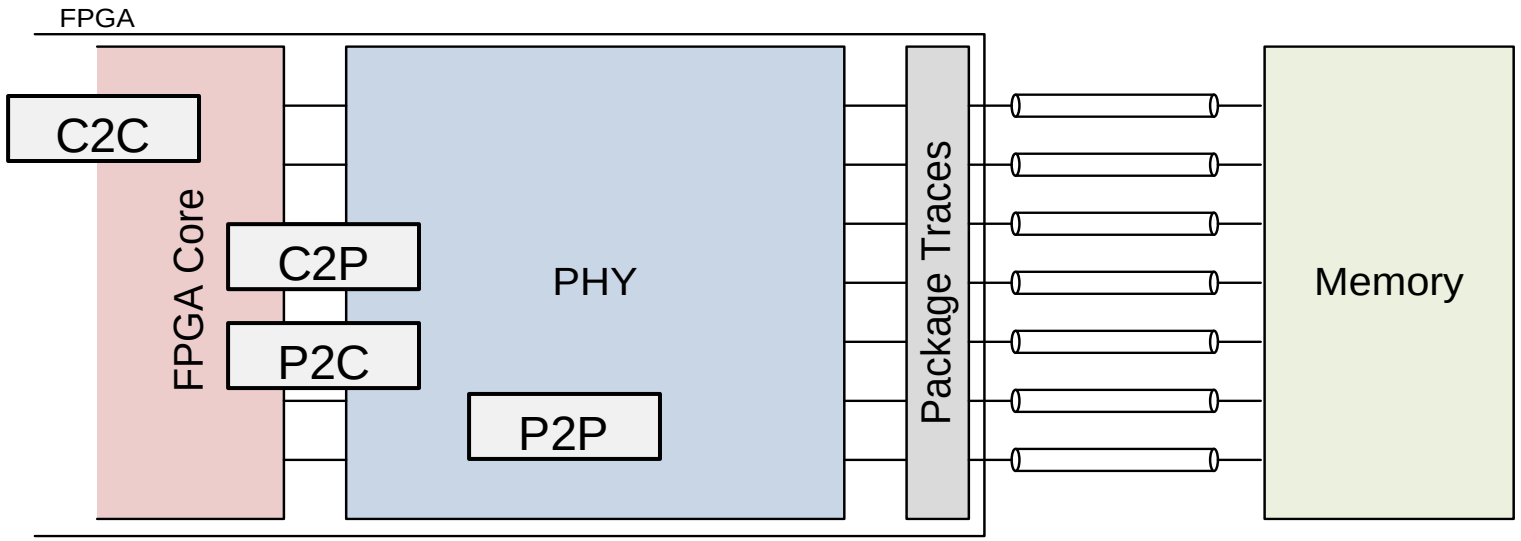
Key Arria 10 EMIF Usability Enhancements

- No requirement for <core>_pin_assignments.tcl script
  - Will be handled internally by Quartus II software
- No RTL connections from OCT cal block to I/Os
  - Will be handled by new QSF assignments (generated by IP core)
- No Quartus II timing analysis of hard paths
  - ASIC block analyzed and timing closed by ICD before tapeout
- Eliminate PLL/DLL/OCT sharing
  - No need for GUI option
  - No need to connect master to slave via RTL
  - PLL/DLL/OCT exist in every IO block
  - Fitter responsible for merging interface to share an IO block (*certain conditions apply)
- Simplified PLL generation
  - Arria 10 – instantiate IOPLL directly
    - Plus, IOPLL may automatically-create clock constraints (simpler SDC)

Understanding the Arria 10 EMIF Fitter

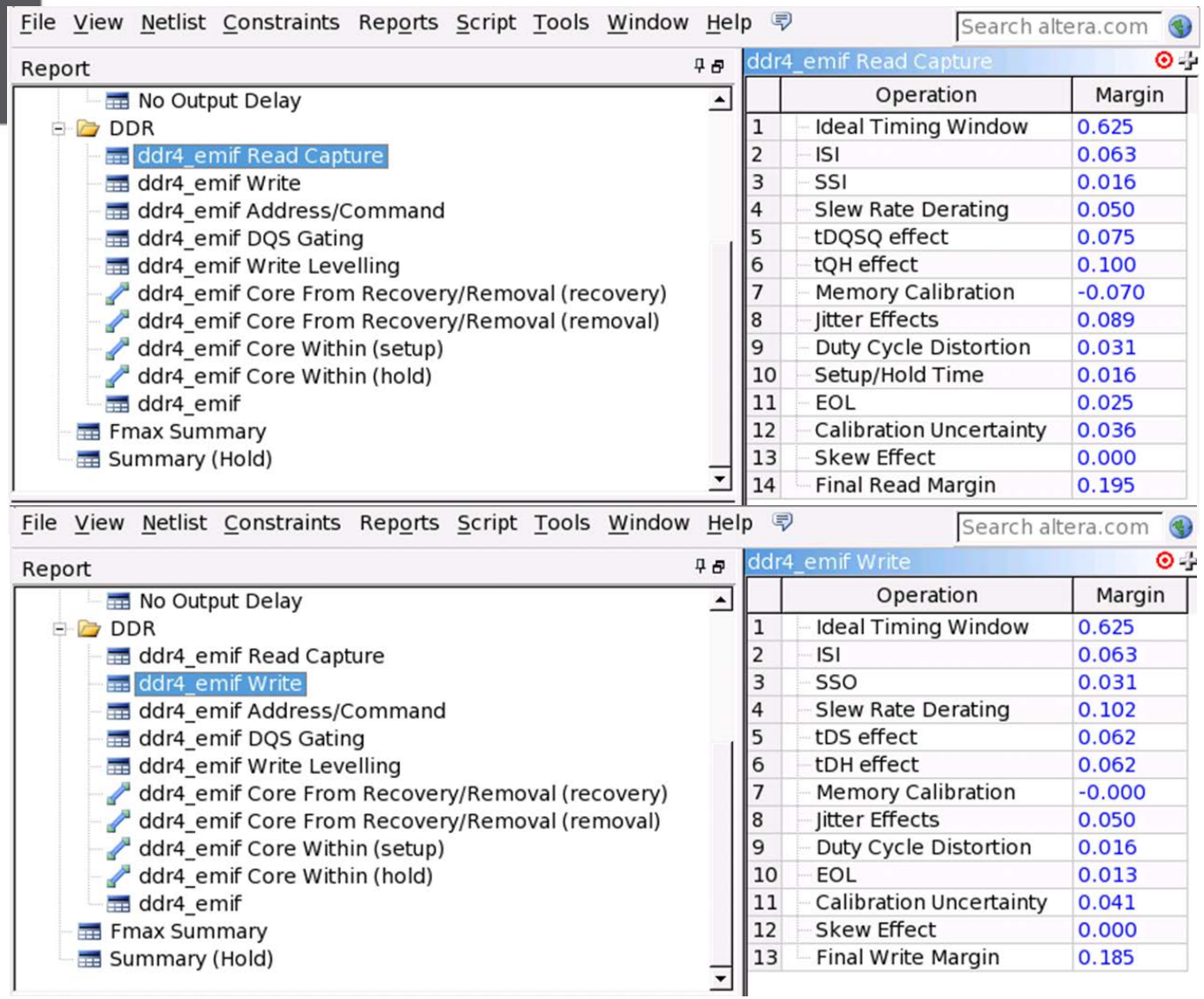
- Responsible for finding placement of all EMIF blocks
- EMIF interfaces in same column will merge IO-AUX
- Can move a lane between banks
  - But ***cannot*** merge two lanes
- Can swap pin locations within a lane
  - But ***cannot*** move EMIF pins into or out of a lane
- Place GPIO I/Os in unallocated locations in a lane
- *Cannot* merge LVDS and EMIF in same bank

Arria 10 EMIF Timing Analysis



- C2C: Core-to-Core paths
  - Analyzed by Quartus II TimeQuest, same as 28nm
- C2P/P2C: Core-to-Periphery
  - Analyzed by Quartus II TimeQuest, but dedicated hardware to match phases
- P2P: Periphery-to-Periphery
  - Closed by ICD; Quartus II TimeQuest just sees *min pulse* requirement
- I/O: External transfer between FPGA and Memory
  - Formula-based, similar to 28nm; report_ddr script shows individual transfers
  - Still factors in GUI parameters for memory and board settings

TimeQuest DDR4 Timing Example

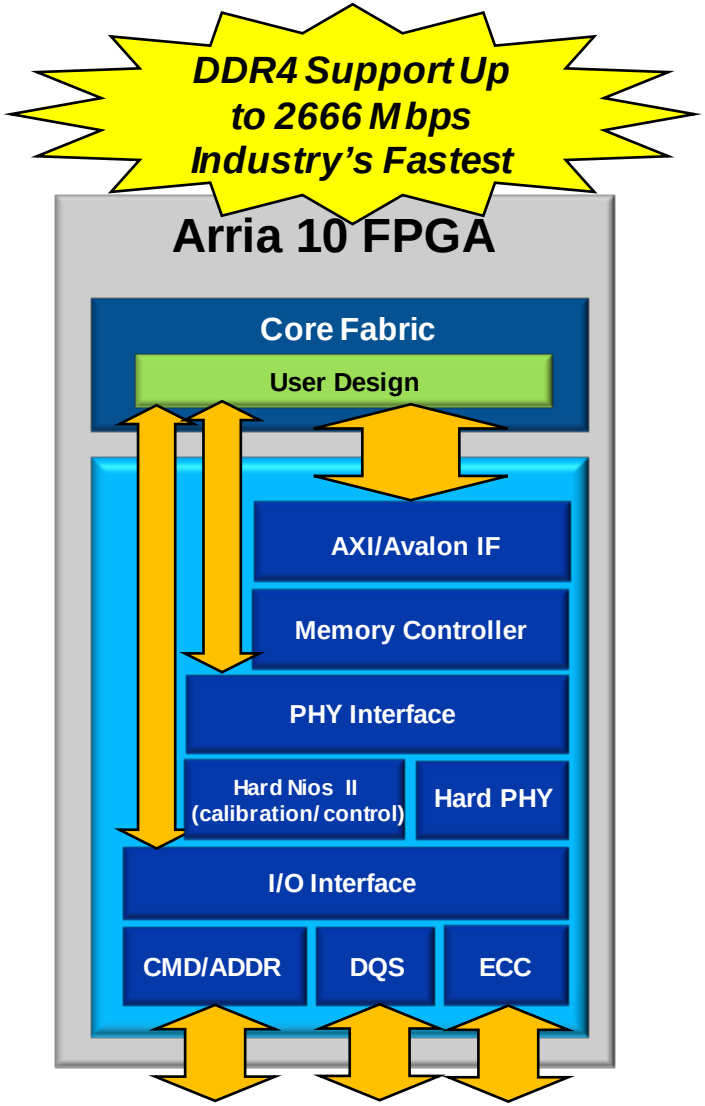


Summary

- DDR4 implementation challenging due to shrinking data valid window
- Arria 10 addresses those challenges with hardened memory interface supporting PVT calibration
- PCB design will need to be simulated to ensure success

Hardened Memory Controller & PHY

- High performance hardened memory controller
  - Built-in timing closure shortens engineering cycles
  - Saves logic and memory resources
    - 5k LEs and 29 M20K blocks per x72 DDR3 IF
  - Up to x144 support
  - Up to 4 x72 DDR3 interfaces on single device
- Hardened memory controller supports
  - DDR4, DDR3, LPDDR3
  - Controller and PHY bypass-able for flexibility to support emerging & legacy standards
- Additional support (soft controller)
  - RLDRAM 3, QDR IV, QDR II+ Xtreme, QDR II+, QDR II
- Intelligent calibration and dynamic skew control via hardened Nios II processor



Hardened Memory Controller & PHY

Thank You