SYNOPSYS®
Silicon to Software™

DWC LPDDR5/4/4X Controller Overview

Rev. 1.0

Horizon

Memory Controller AE Team

2021-12-28

CONFIDENTIAL INFORMATION

The information contained in this presentation is the confidential and proprietary information of Synopsys. You are not permitted to disseminate or use any of the information provided to you in this presentation outside of Synopsys without prior written authorization.

IMPORTANT NOTICE

In the event information in this presentation reflects Synopsys' future plans, such plans are as of the date of this presentation and are subject to change. Synopsys is not obligated to update this presentation or develop the products with the features and functionality discussed in this presentation. Additionally, Synopsys' services and products may only be offered and purchased pursuant to an authorized quote and purchase order or a mutually agreed upon written contract with Synopsys.

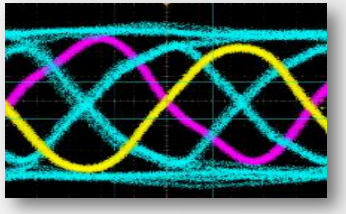
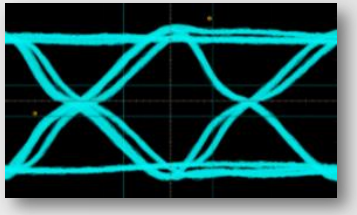
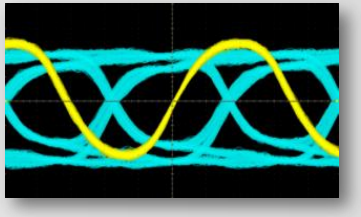
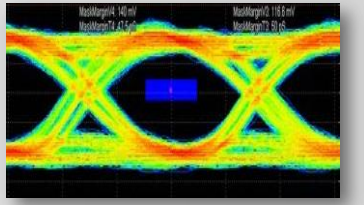
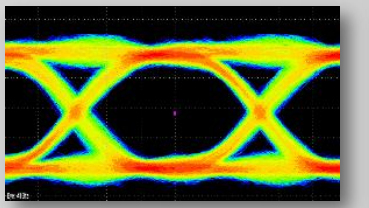
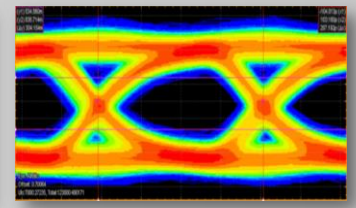
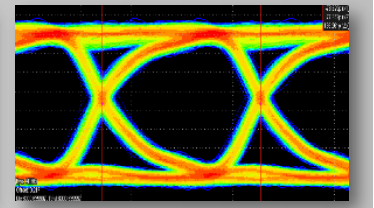
DesignWare Memory Interface IP

Fast Facts

- Support for DDR4, DDR3, DDR3L, DDR2, LPDDR4/4X, LPDDR3, LPDDR2 & LPDDR & HBM/HBM2
- DDR5, LPDDR5 & HBM2E in development
- Over 800 DDR PHY design wins
- Over 590 digital DDR controller design wins
- 70+ DDR PHY test chips
- #1 Supplier of DDR Interface IP since 2010
  - Sources: Gartner, IPnest

Synopsys DDR Silicon Firsts

- ✓ LPDDR5-6400 (Oct 2018)
- ✓ DDR5-4800 (Oct 2018)
- ✓ LPDDR4-3733 (December 2016)
- ✓ HBM in production (June 2015)
- ✓ DDR4-3200 (January 2015)
- ✓ LPDDR3-1600 (October 2012)
- ✓ LPDDR2-1066 (October 2010)
- ✓ DDR3-2133 (March 2010)

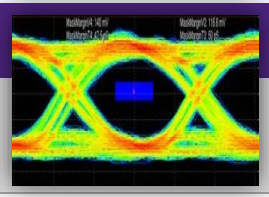
2010	2012	2013	2015	2016	2018	2018
						
DDR3-2133	LPDDR3-1600	DDR4-2400	DDR4-3200	LPDDR4-3733	DDR5-4800	LPDDR5-6400

Complete DesignWare DDR IP Solution

Value points: Lowest risk, highest performance, smallest area

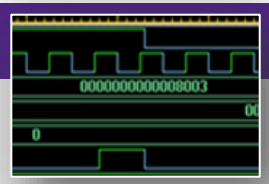
Controllers

- Highest Bandwidth & Reliability



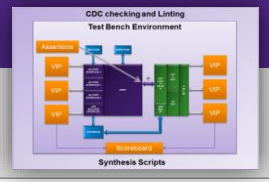
PHYs

- Best PPA, Online PHY Compiler



Verification IP

- VIP, Models, Monitors, Architecture



Interface Subsystems

- Controller, PHY, Hardening, SI, Driver

Prototyping

- FPGA, IP Prototyping Kits and VDKs

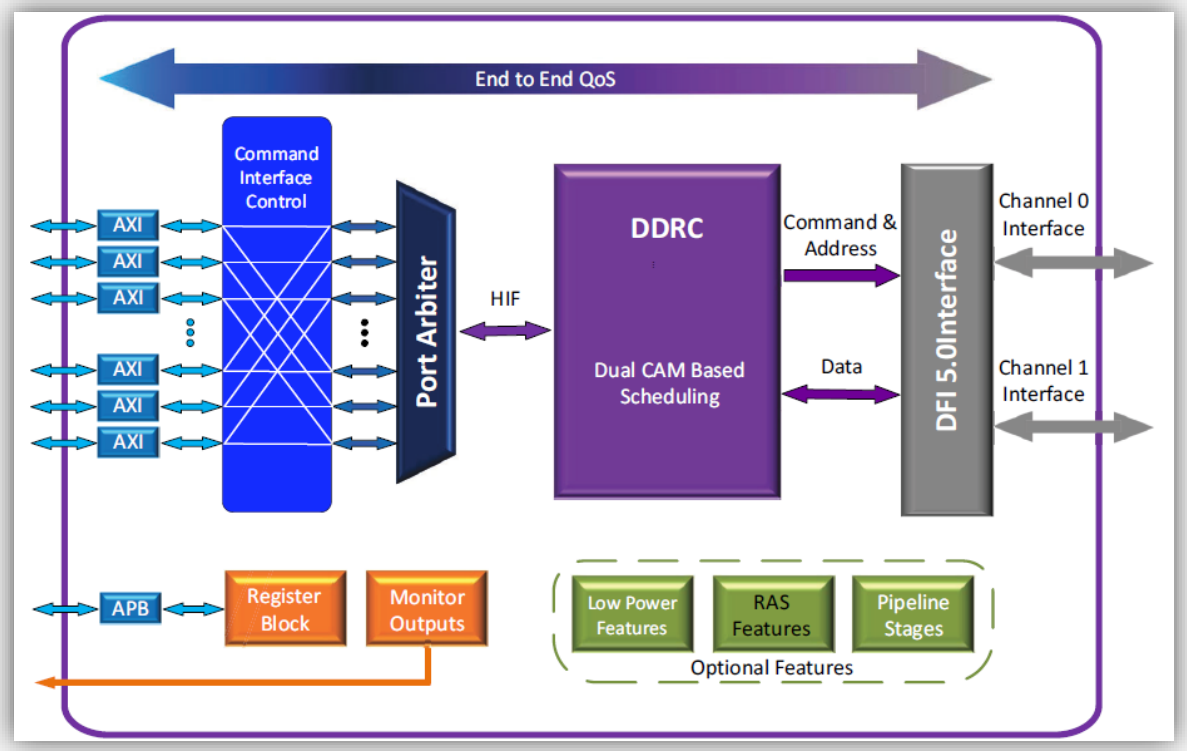
- Low risk: 1600+ DDR design wins, 10+ year history in DDR, 75+ test chips, industry's #1 DDR supplier
- Highest performance and lowest area DDR PHY in advanced technologies
- Highest performance DDR controller proven in multiple customer evaluations
- Production experience with both HBM and HBM2
- Supports process nodes to 5nm, all leading foundries
- Flexible: Rapidly configures for any market
- Interface IP subsystems: Accurate SoC integration

LPDDR5/4/4X Controller Overview

DesignWare LPDDR5/4/4X Controller

Next generation memory controller optimized for mobile and automotive solutions

- Highly Configurable Architecture
  - Supports JEDEC standard LPDDR5 and LPDDR4/4X SDRAMs
  - Single channel solution or dual channel (roadmap item) solution
  - Multiport ARM® AMBA® interface (4 AXI™ / 3 AXI™) with managed QoS or single port host interface to the DDR Controller
  - DFI 5.0 compliant interface to DesignWare DDR PHYs and other PHYs supporting the DFI interface
- Highest System Performance
  - Read Reorder Buffer (RRB) enables maximum scheduling flexibility
  - High-bandwidth design with up to 64 CAM entries and latency as low as 8 clock cycles
- Advanced Reliability and Functional Safety Features
  - Multiple reliability, accessibility and serviceability (RAS)
  - Efficient inline error correcting codes ECC
  - ASIL-B off the shelf solution, ASIL-C/D via SoW



Low Power Support

Two methods to control power

Fully automated low power control

- Programmable timers monitor traffic
- After a user defined idle time X the controller automatically moves the memory to a power down state
- After a further user defined idle time Y the controller moves the memory to self refresh
- Controller also uses the DFI LPI to enable PHY to be placed in a lower power state
- User has the option to enter Self Refresh Power Down

Hardware low power interface

- Uses AMBA® 3 AXI™ signaling protocol
- Used to allow system to force the memory into and out of self refresh via hardware interface
- Allows for AXI clocks to be stopped to the Controller
- DDRC clocks may also be stopped based on the state of the DDRC

Refresh Flexibility

Keeps traffic moving

- Refresh is getting longer
  - 380ns tRFCab for 24Gb LPDDR4 devices
  - Plus, takes time to recover
    - Open all banks (limited by tRRD/tFAW)
    - Process backlog of commands from tRFC time
- LPDDR5/4/4X Controller improves performance
  - Per-rank refresh
    - Scheduling refresh to one rank while accessing the other
  - Per-bank refresh
    - Scheduling refresh to one bank while access the others
  - Issuing speculative refreshes
    - Speculatively issuing refreshes to banks with no pending transactions
  - Bursts refreshes
    - Lowers latency by reducing pre-charge and activate commands per bank

Quality of Service

Traffic classes

- Normal Priority Read and Write (best effort, no guarantees)
  - No resource allocation and shares the resources with other classes
  - Low priority until starves in port arbiter or in DDRC
- High Priority Read (minimum latency)
  - For latency sensitive traffic
  - Eligible commands will use the bypass path for minimum latency
  - Higher priority than NPR/NPW, lower than expired VPR / expired VPW
- Variable Priority Read and Write (maximum latency bound)
  - For latency tolerant but real time traffic
  - Latency of each VPR/VPW transaction is tracked throughout the controller
  - Treated as NPR/NPW until the command spends predetermined time in the controller. Once the latency expires, VPR/VPW command become the highest priority at both decision points (port arbitration & CAM scheduling)

Features for Automotive

Internal safety mechanisms

- Targeted at customers requiring ASIL certification using all the latest safety features
  - In-line ECC support
  - On-chip parity support
  - ECC scrubbing and error injection
  - Configuration register parity protection
  - On chip command and address protection
  - Address parity protection with in-line ECC

Certification for ISO 26262 Part 5 HW Development

Configuration Options

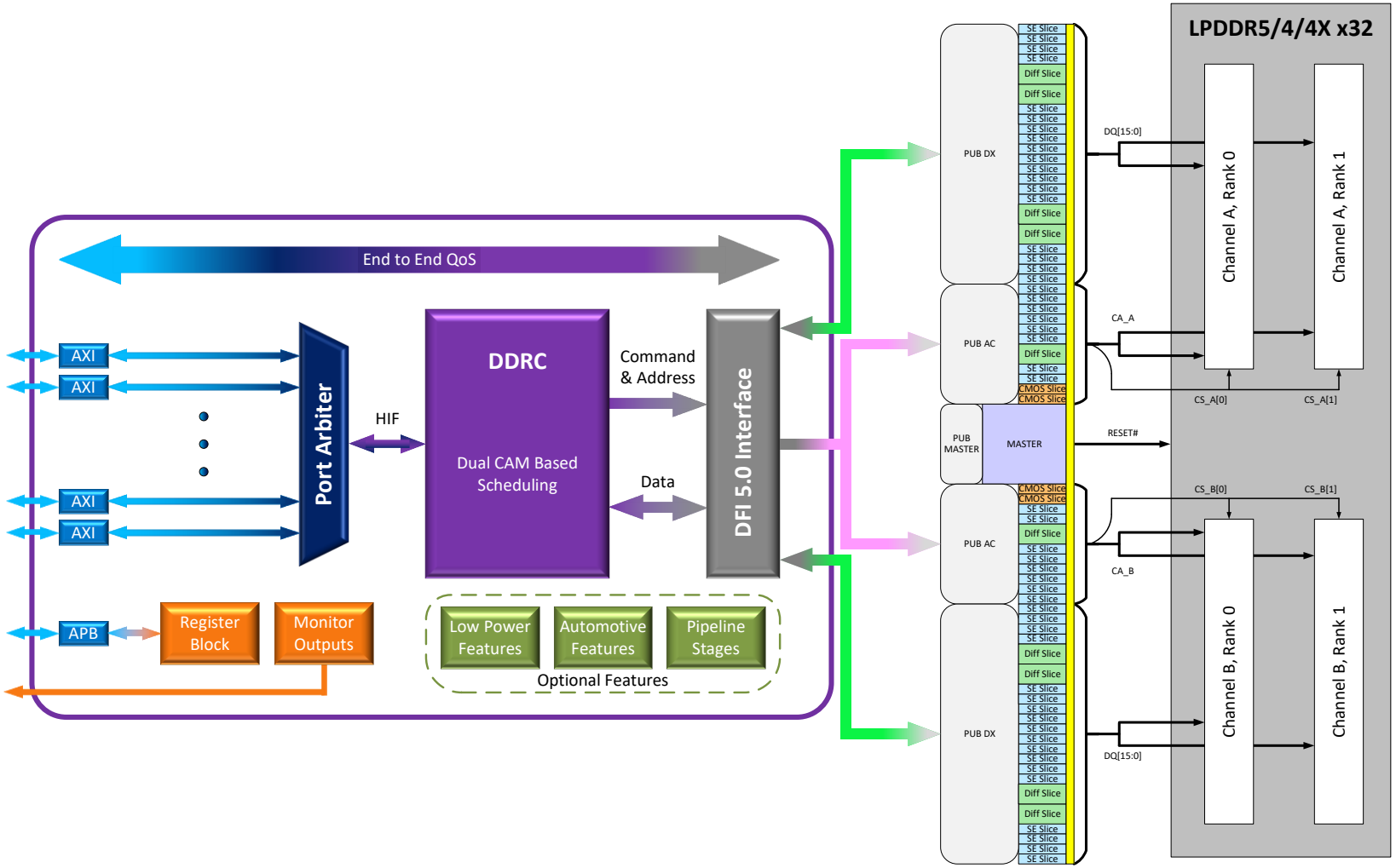
HW configuration of LPDDR5/4/4X Controller

- Currently, LPDDR5/4/4X Controller is LCA(Limited Control Availability) phase
- IP is still in development and/or verification is not complete
- Our verification activity is based on the fixed HW configuration which is in SoW
- You might see simulation error if you change HW configuration
- Please inform your HW configuration update via SolvNet Case

DW LPDDR5/4/4X Single Channel Option

Preferred option for automotive applications

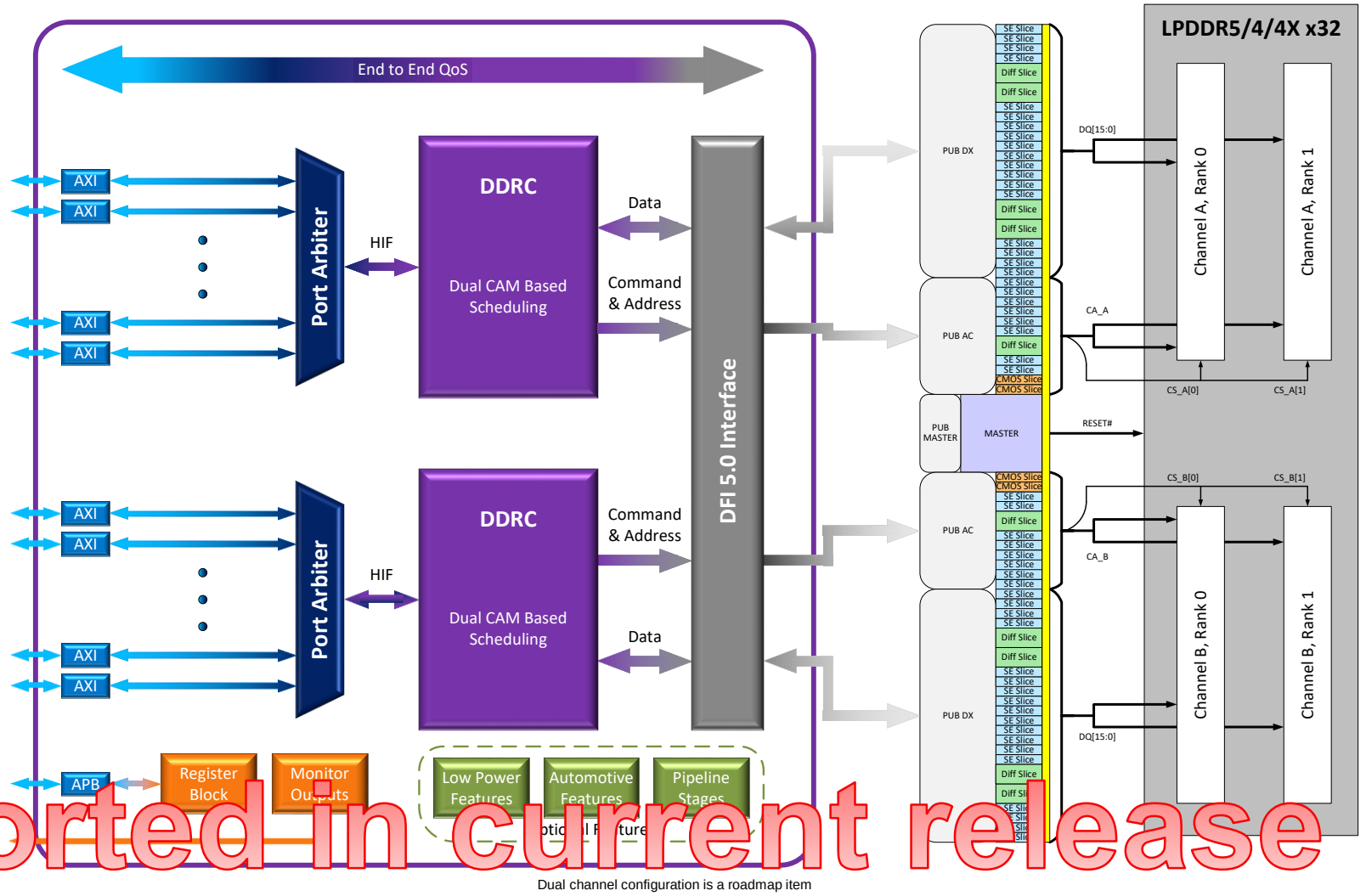
- Memory is still accessed using two channels with independent CA busses for maximum stability
- Dual PHY DFI interfaces are accessed with a single controller DFI interface
- Controller treats both memory channels as a single channel
- Solution appears as a single 32-bit channel to the system



DW LPDDR5/4/4X Dual Channel Option

- Operates two channels of DDR using independent CA busses
- Higher performance for video macroblocks and other short transactions
- Twice the number of banks available in the system compared to single channel solution
- Channels may be in different power states to save power
- Access interleaving handled at the system level

System level interleaving option



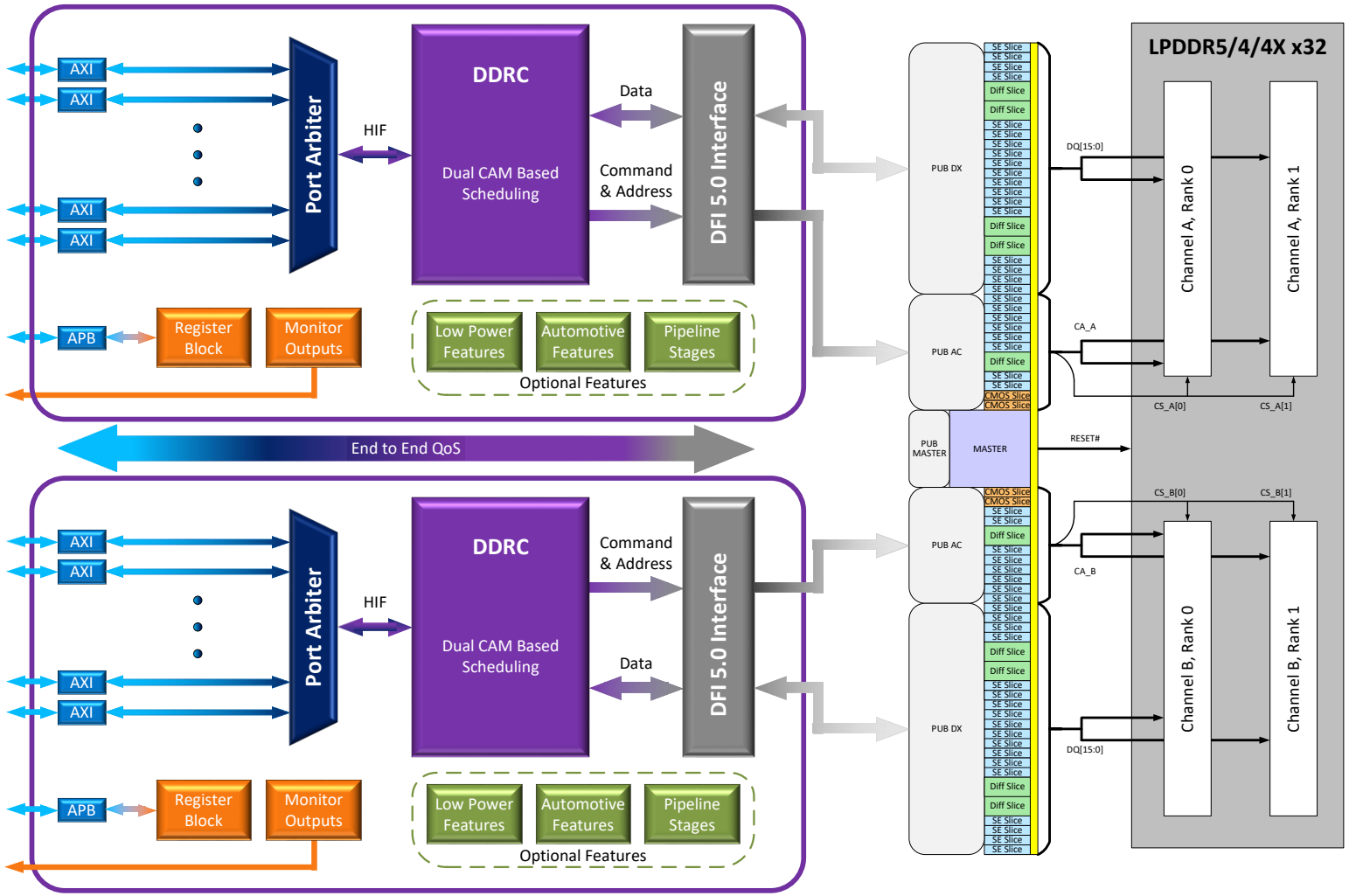
Dual channel configuration is a roadmap item

Not supported in current release

DW LPDDR5/4/4X Hybrid Channel Option

Dual channel access with single channel controllers

- Still operating two channels of DDR using independent CA busses
- Channels are fully independent from memory back to the system
- Channels may still be in different power states to save power
- Access interleaving must be handled at the system level
- Minimal area difference, if any, from dual channel option on previous slide



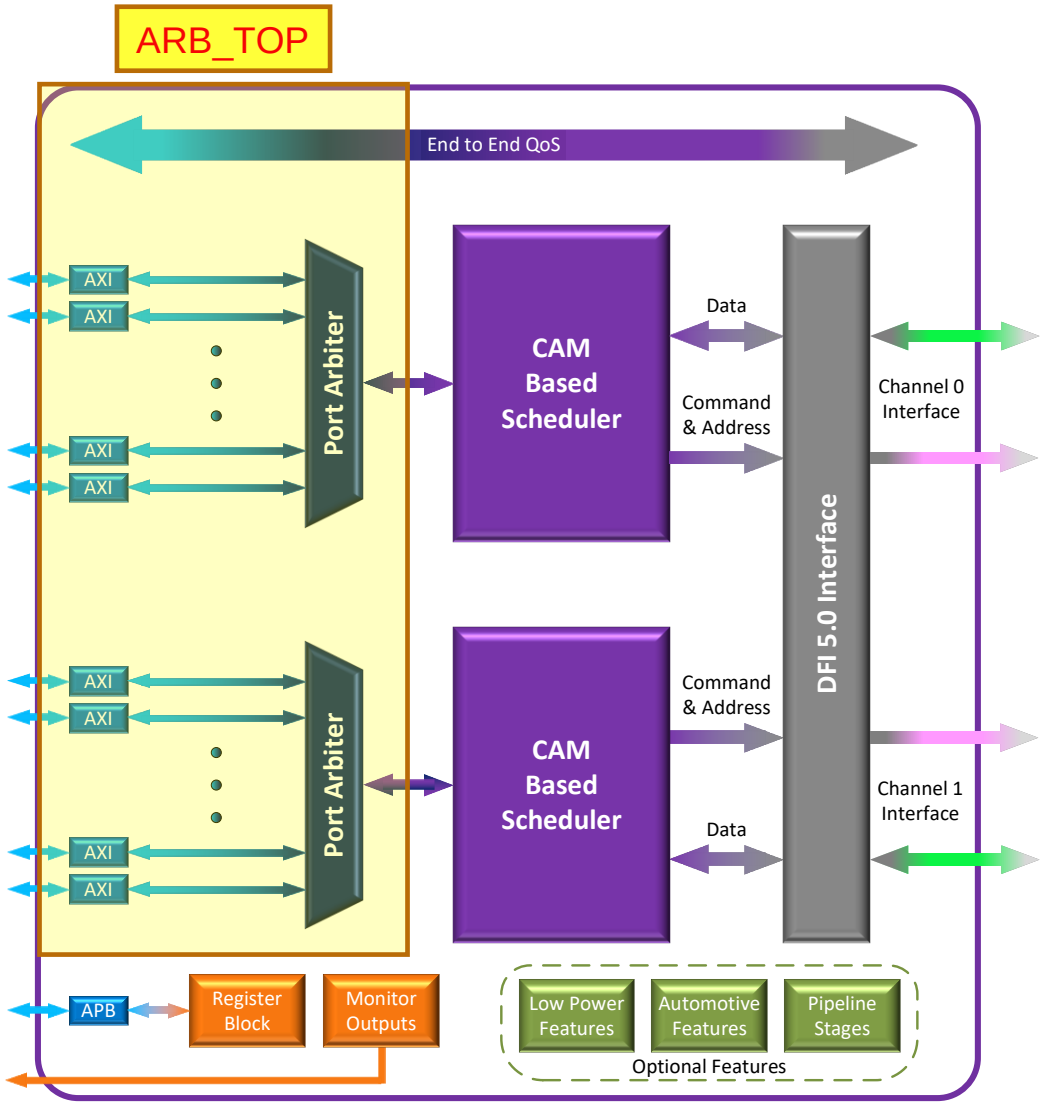
IP testcase can support only one DDRCTL instance

Architecture(AXI Interface: ARB_TOP)

ARB_TOP Hierarchy

DDRCTL's "ARB_TOP" hierarchy responsible for AXI2HIF Protocol conversion and QoS-Aware based port arbitration

- The AXI box is called 'XPI' in the design
  - This is the aXi Protocol Interface for DDRC
  - Converts AXI transaction to a HIF friendly format
- Dual channel interleave feature allows XPI ports to interleave transactions across the available DDRC channels.
- The ARB_TOP hierarchy can support Single/Dual DDRC channels
- Every DDRC channel has a dedicated QoS aware port-arbiter



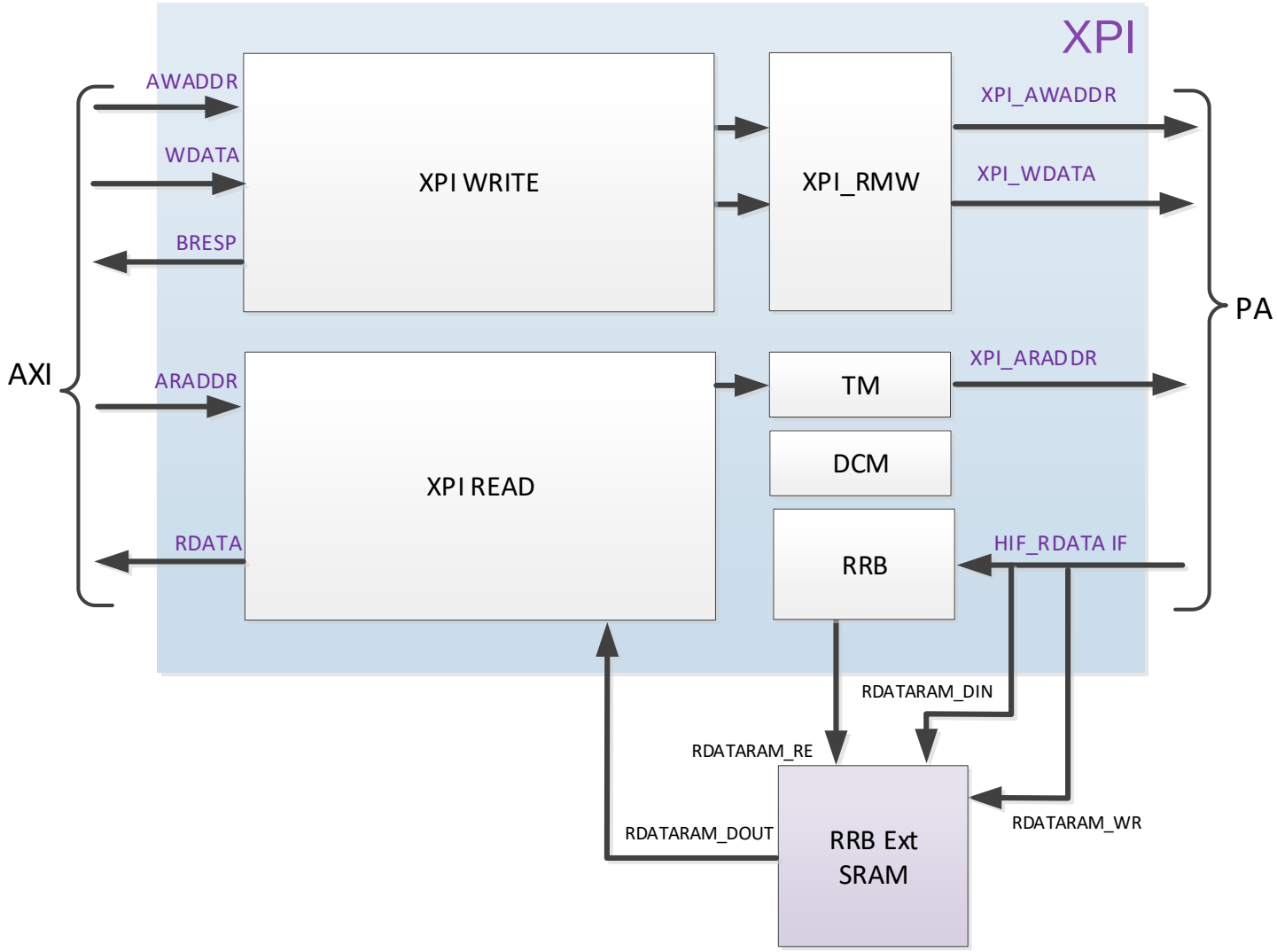
AXI Feature set support

- AXI4 protocol compliance
- Supports exclusive access monitors
- Configurable – per port – AXI data widths
- Per Port Configurable axi clock frequency
- Limitations:
  - axregion and axcache signals
  - Locked accesses
  - FIXED transaction type (that is, only INCR and WRAP are supported)
  - AxPROT signaling
  - ACE (as well as ACE-lite)

XPI Hierarchy

ARB_TOP's XPI hierarchy responsible for following functions

- Independent WR and RD paths to PA
  - RDWR_ORDERD feature allows SW to enable ordering between the RD & WR paths
- System address remap (SAR)
  - Non contiguous system address map is converted to contiguous HIF addresses
- AXI to HIF Address generation
  - AXI is a byte address
  - HIF uses DRAM col address
- AXI 2 HIF burst splitting
  - Splits long AXI transaction to multiple HIF bursts
- Read-Modify-Write detection
- AXI compliant re-ordering of HIF Rdata using a Read-Reorder-Buffer (RRB)
- upsizing/downsizing of AXI transaction to match the programmed HIF datawidth

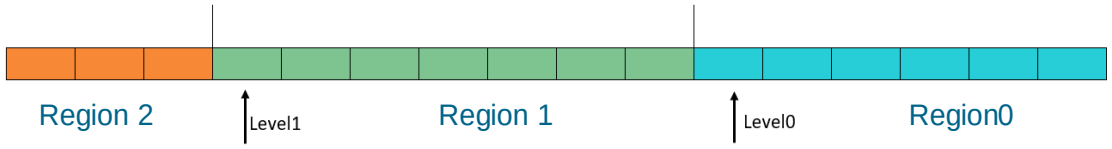


QoS in ARB_TOP

Using AXI's AxQOS for Port arbitration

- QoS Mapping
  - AXI allows 16 levels of QoS using AxQOS
  - ARB_TOP maps these QoS levels to internal priorities for requests
  - RD CMD priorities : HPR, LPR, VPR
  - WR CMD priorities : NPW, VPW

AxQOS vector :

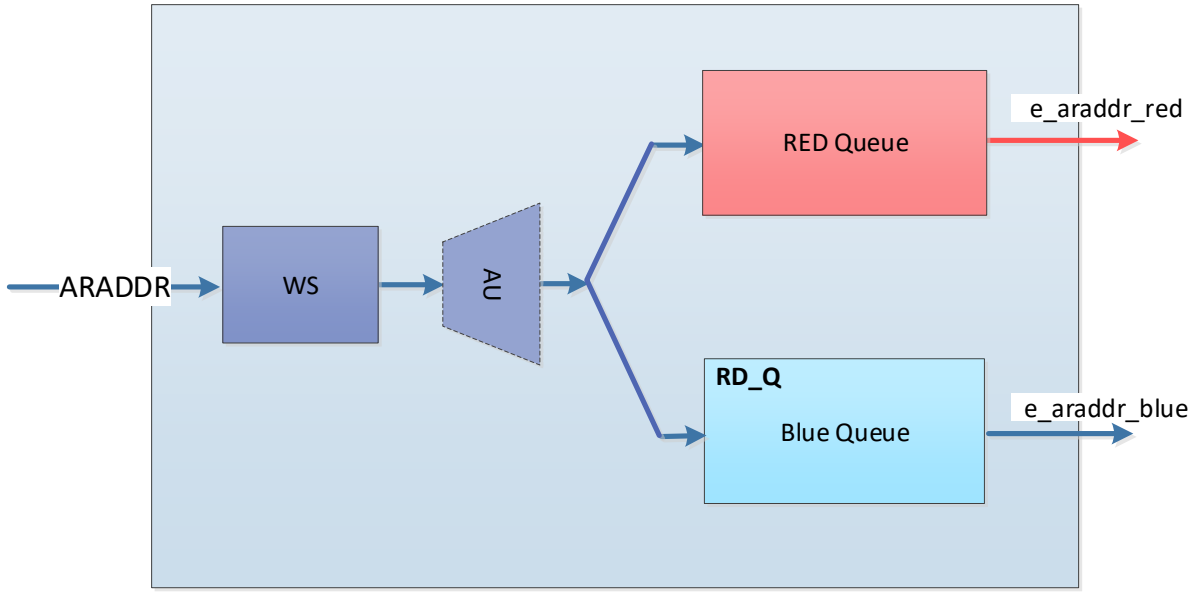


- QoS Striping
  - PCFG[R/W]QOS0 MMR enables users to create regions in AxQOS vector
  - The same register can then be use to assign priorities to regions
  - HPR,LPR, VPR, NPW, VPW
  - Region2 is present only for ARQOS when USE2RAQ feature is enabled

Dual read request Queues

USE2RAQ

- This is XPI_READ feature
  - When enabled the read request path is split into two queues
    - Red and the Blue queue
  - PA sees these as two distinct physical read ports that need arbitration
  - Each queue has its own ageing counter and VPR timeouts
- The main motivation here is to ease head of line blocking
  - HPR requests block by LPR



- AXI RD requests with QoS values mapped to region2 are pushed into RED Queue
  - Region0/1 are always in Blue queue
- XPI maintains coherency of reads between these queues
  - Same ID AXI request with one of them mapping to Red and other to Blue
  - Since PA allocates higher priority to RED queues, This ensures that PA does not break AXI protocol
- Read data interleaving feature should be enabled when using Dual RD queues.

Multi-Port arbitration

For Reads

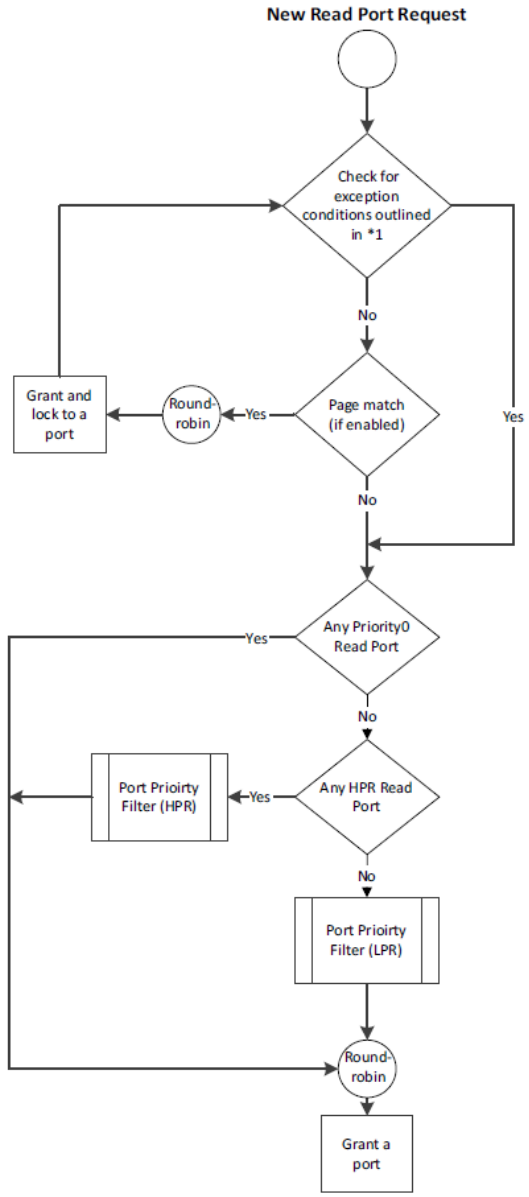
- Arbitration is sensitive to following programmable/configurable attributes of a request
  - Page match
  - Port ageing
  - Expired VPRs
  - AxUrgent signaling
  - Port priority
    - QoS based or static programming based

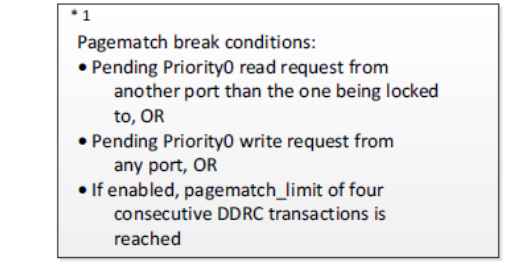
- 2-priority level arbitration based on port aging and expired-VPR (timeout - priority0)

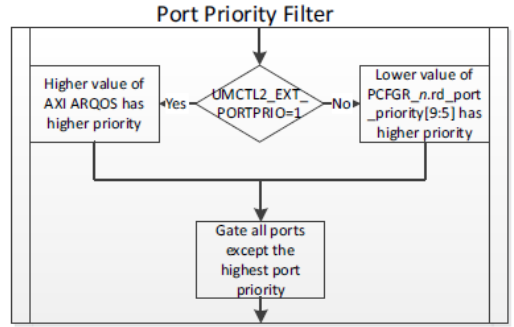
- 2-priority level arbitration for read requests based on DDRC read priorities (HPR/LPR-VPR)

- 32-priority level arbitration based on internal port aging or 16-priority level arbitration based external AXI QoS inputs (selectable by hardware parameter)

- Round-robin arbitration to resolve ports having the same priority after passing all stages of arbitration







Multi-Port arbitration

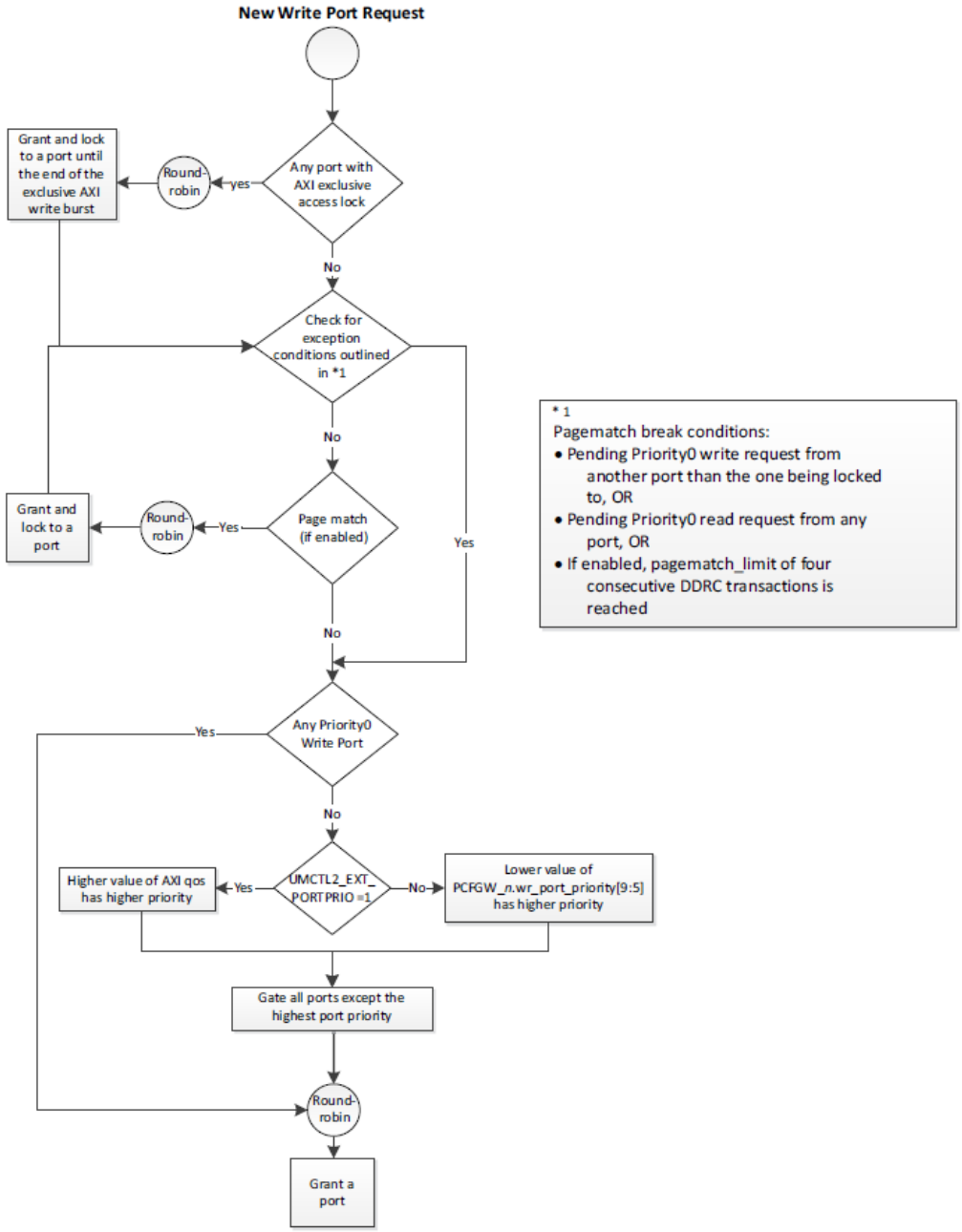
For Writes

- Arbitration is sensitive to following programmable/configurable attributes of a request
  - Page match
  - Port ageing
  - Expired VPRs
  - AxUrgent signaling
  - Port priority
    - QoS based or static programming based

- 2-priority level arbitration based on port aging and expired-VPW commands (timeout - priority0)

- 32-priority level arbitration based on internal port aging or 16-priority level arbitration based external AXI QoS inputs (selectable by hardware parameter)

- Round-robin arbitration to resolve ports having the same priority after passing all stages of arbitration

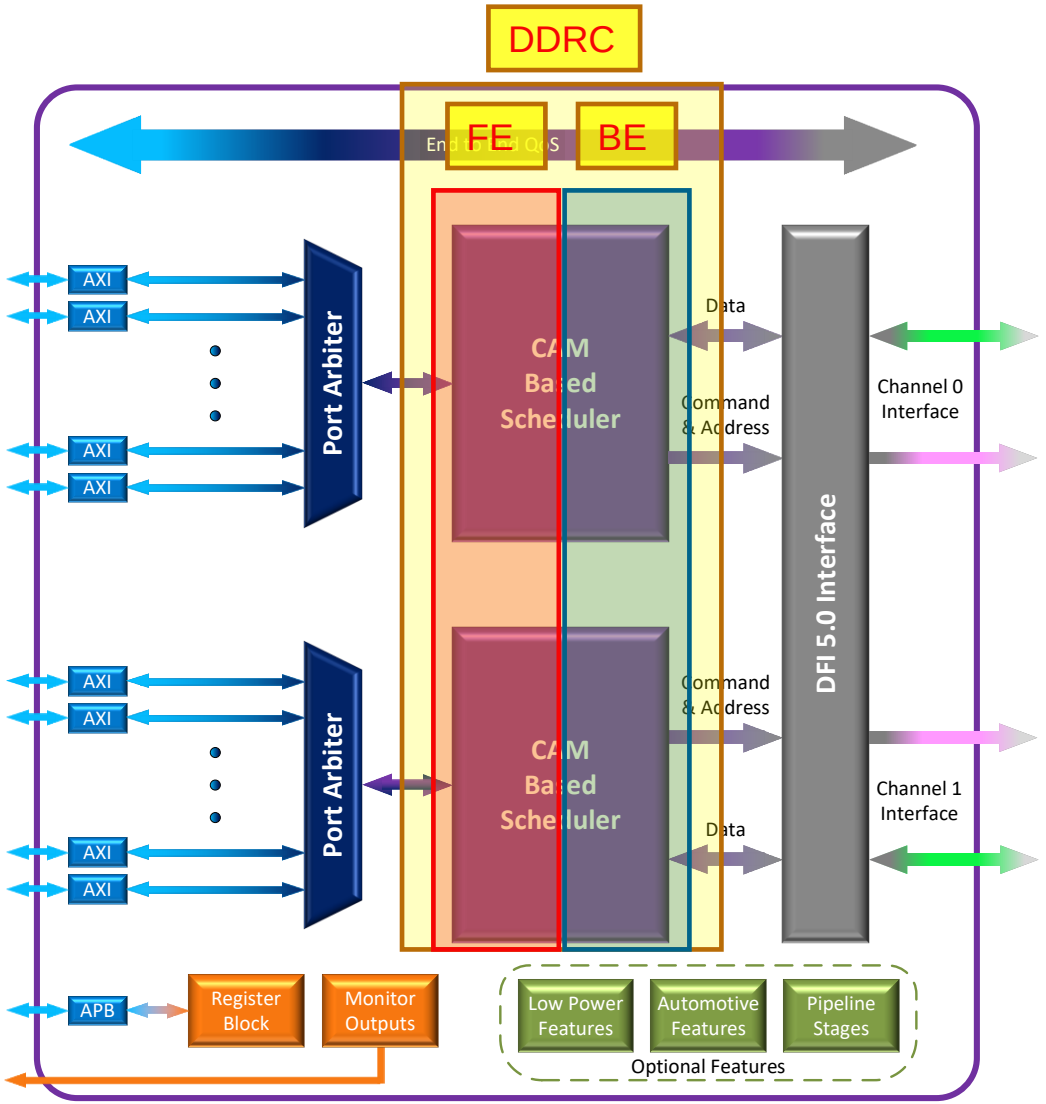


Architecture(DDRC)

DDRC Hierarchy

DDR Controller (DDRC) block responsible scheduling commands to be sent to the PHY, based on priority, bank/rank status and DDR timing constraints

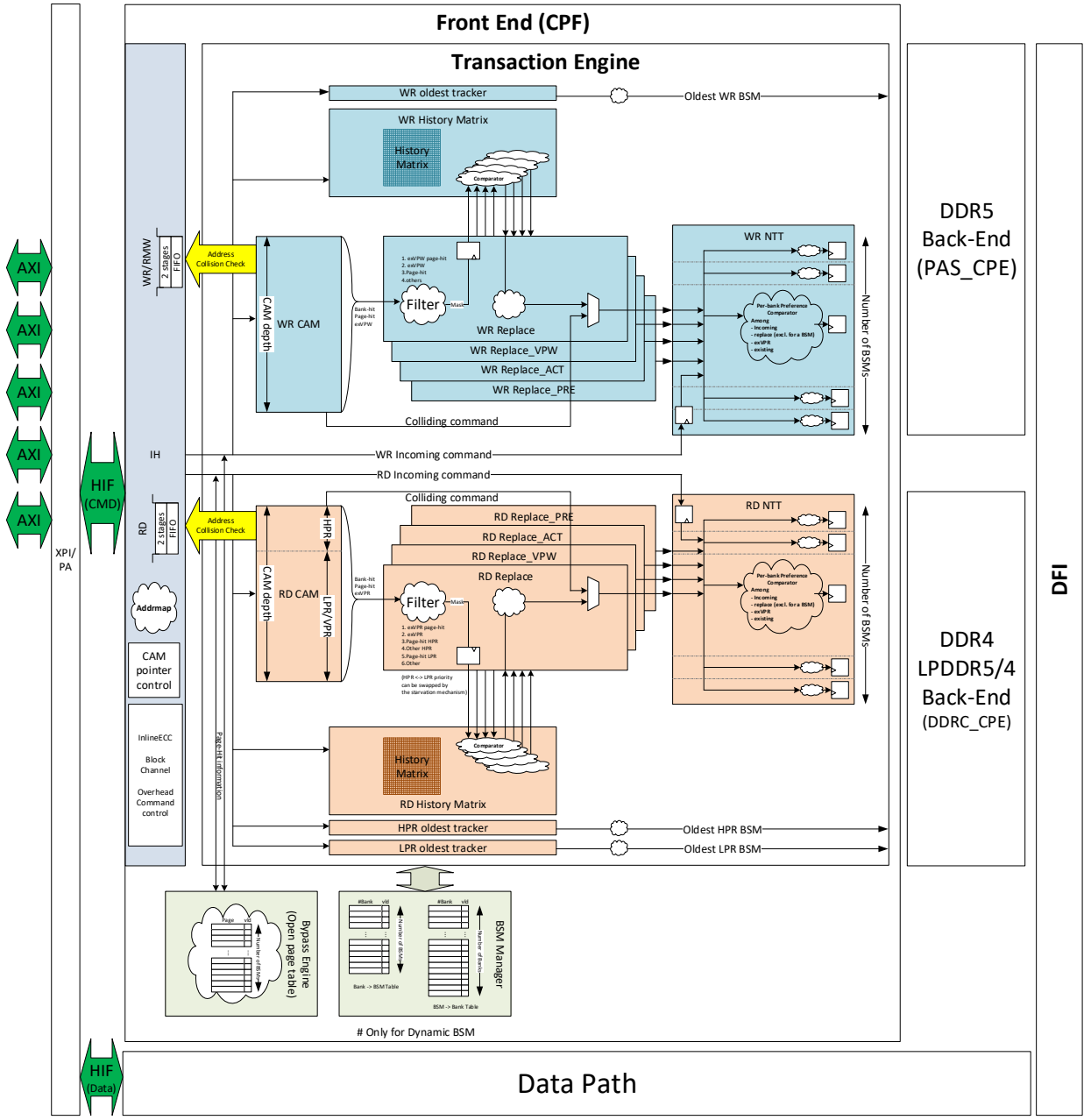
- Write data is stored in a SRAM (internal and external to the DDRCTL) until its associated command is issued to the PHY. The SRAM can be instantiated as embedded memory external to the DDRCTL or internal
- Read data is handled by the response engine in the DDRC and is returned in the order of scheduled read commands on the HIF
- ECC handling is an optional function, which is handled by logic modules within the DDRC in the write data path and in the response engine



Architecture
(Scheduling: Command Path Front End)

Front End - CPF overview

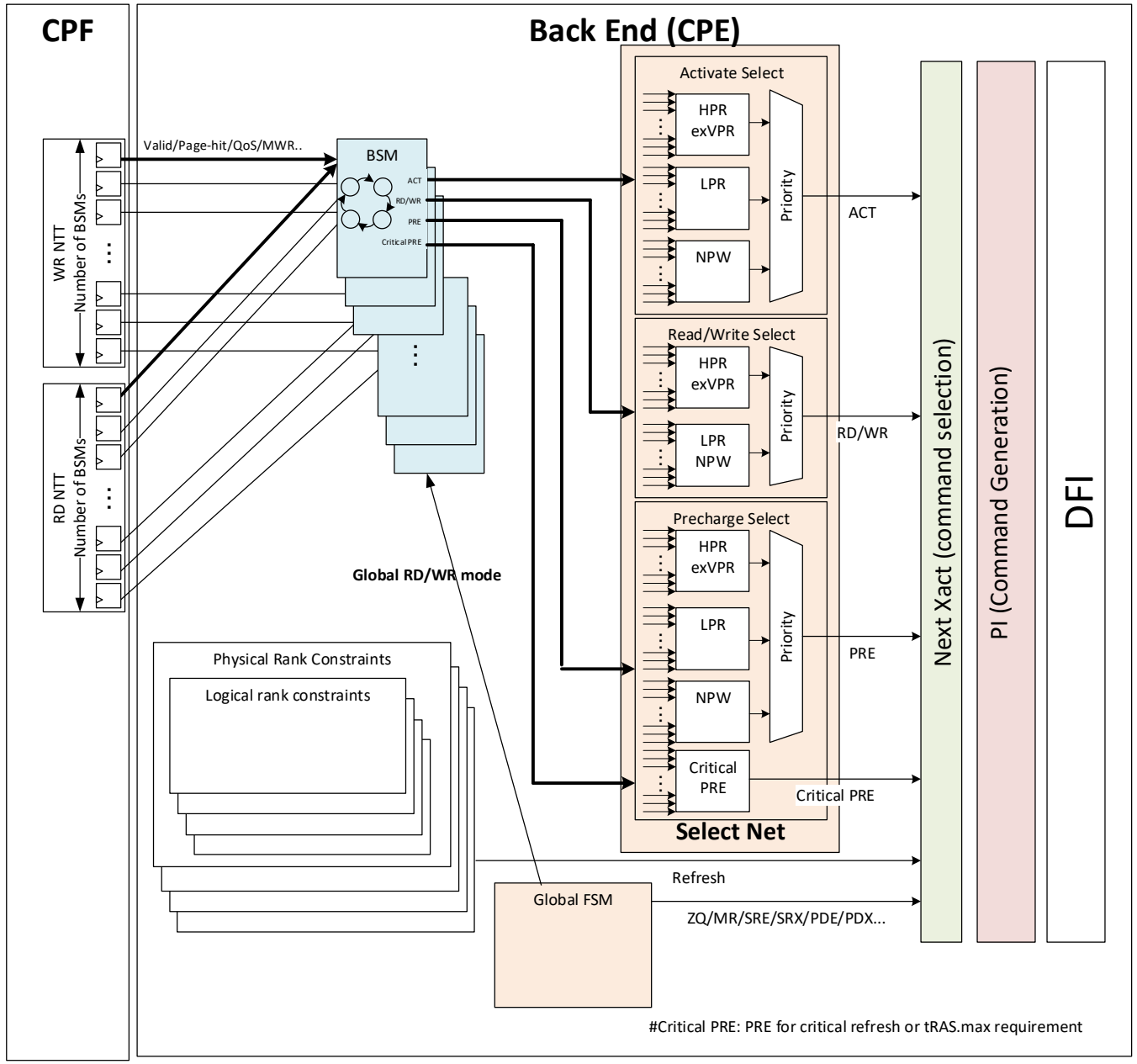
- Input : Command from HIF
- Output : Next Transaction Table (per-BSM)
- Components
  - IH (Input Handler)
    - Address Map
    - CAM pointer management
  - TE (Transaction Engine)
    - CAM
    - Replace logic
    - Age tracking
      - History Matrix
      - Oldest tracker
    - NTT (Next Transaction Table)
  - BE (Bypass Engine)
    - Open Page Table (always there)
    - RD Bypass control (MEMC_BYPASS==1 only)
  - BM (BSM Manager)
    - Dynamic BSM management
    - UMCTL2_DYN_BSM==1 only



Architecture
(Scheduling: Command Path Back End DDRC_CPE)

Back End - CPE overview

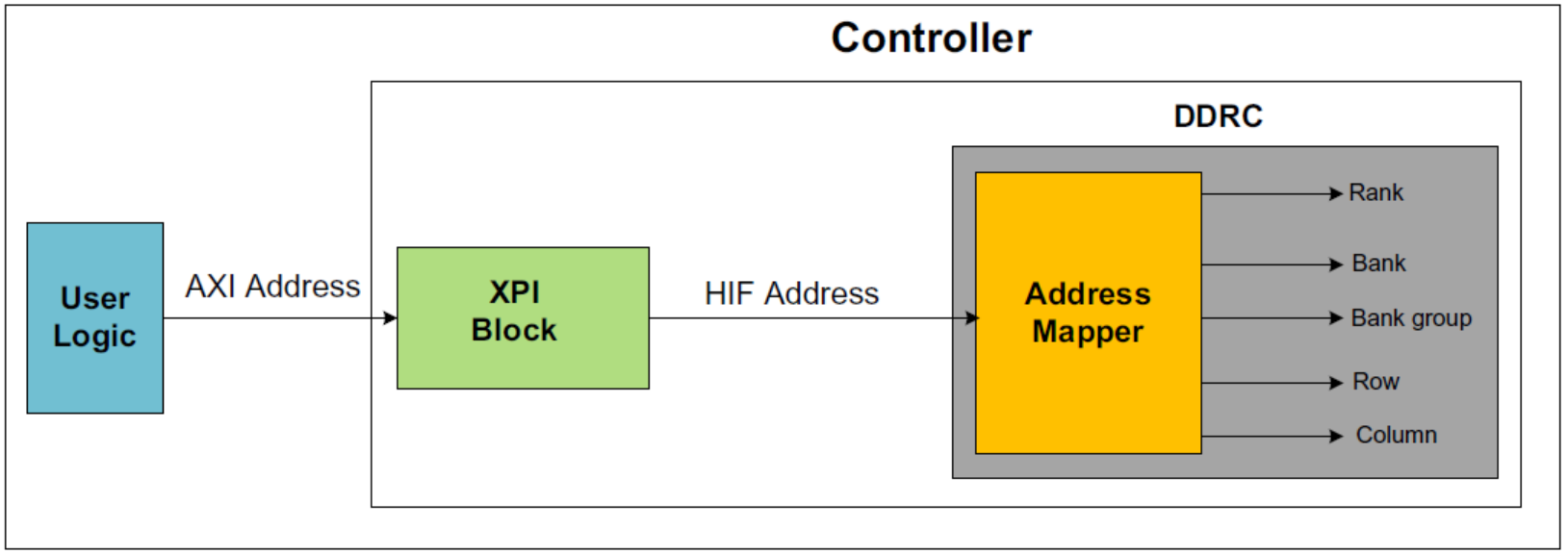
- Input : NTT (Per BSM request)
- Output : DFI (Command to PHY)
- Components
  - BSM (Bank State machine)
    - DRAM timing management (bank based)
    - Generate ACT/RD/WR/PRE request
    - Per BSM Read/Write switching (For PRE/ACT)
  - Select net (Priority encoder)
    - Pre ACT/RDWR/PRE and priority/directions
    - Choose one of request from BSM
  - Global Scheduler
    - Next Xact
      - Choose commands for a DFI clock cycle
      - Including maintenance command e.g.) REF/ZQ/MR
    - Global FSM
      - Manage global RD/WR switching
      - Power management, etc.
    - Physical rank constraints
      - Physical rank specific timing management (RD/WR gap etc.)
    - Logical rank constraints (3DS device only)
      - Logical rank specific timing management (tRRD)
      - Refresh control
  - PI (PHY interface)
    - Command generation



Address Mapping

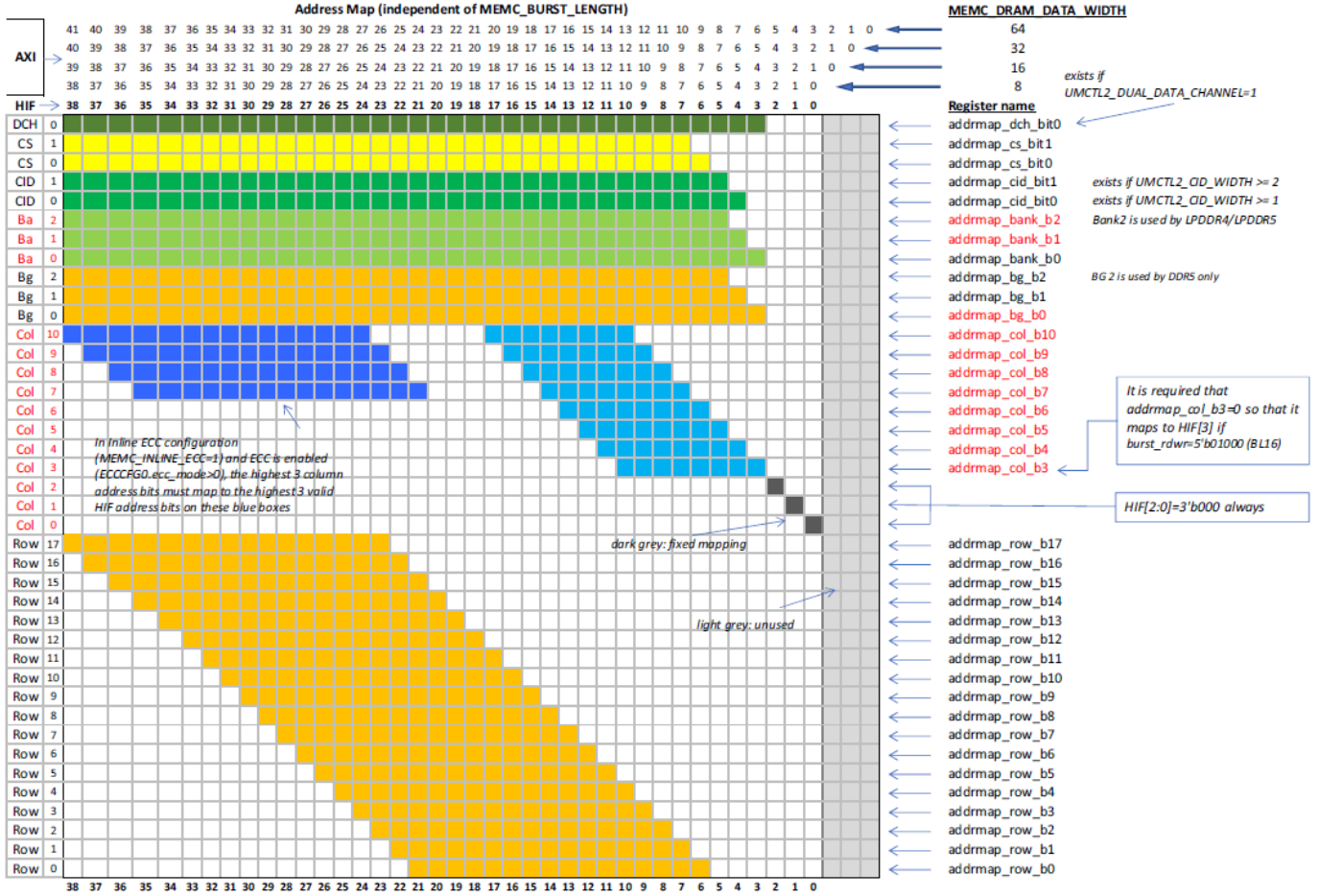
Address Mapper

- Read and write requests are provided to the LPDDR5/4/4X Controller with a system address
- System address is the command address of a transaction as presented on AXI data port
- LPDDR5/4/4X Controller converts this system address to a physical address
- It maps the system address to the SDRAM rank, bank group, bank, row, and column addresses depending on ADDRMAP* register setting



Address Mapping – From System to DRAM

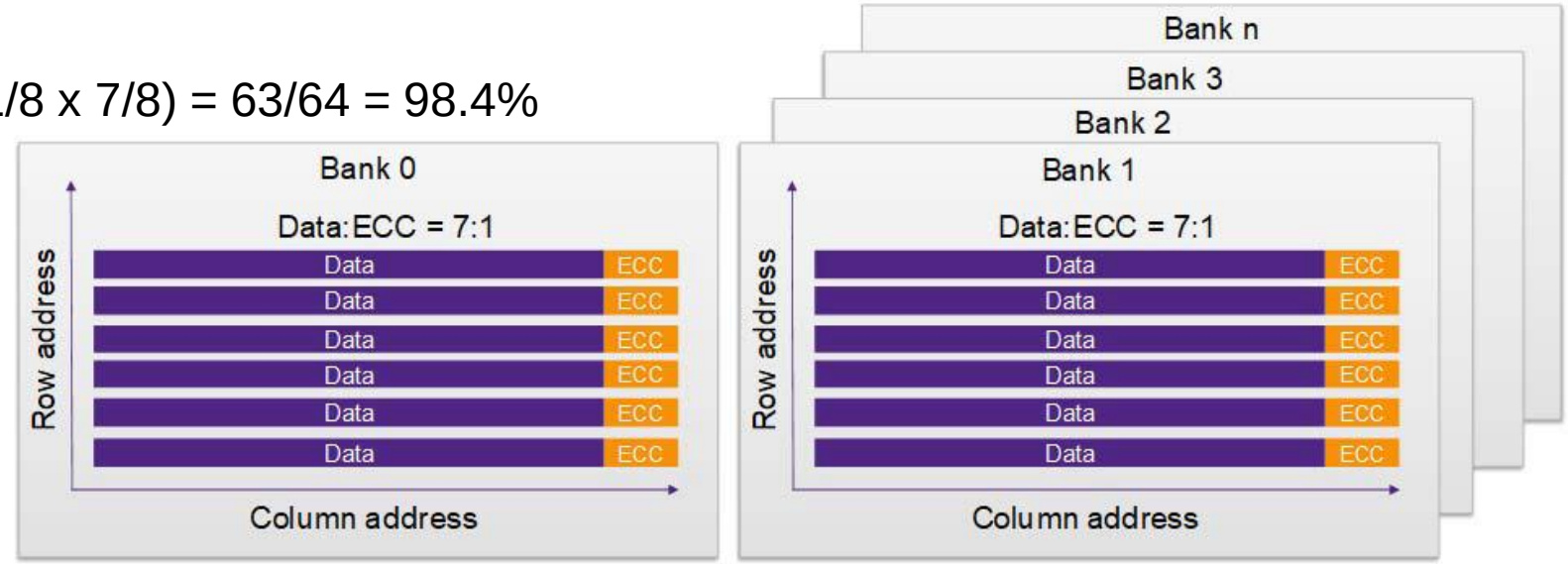
- Address mapping to be used depends on the use-case
- LPDDR5/4/4X Controller provides a set of registers that allows flexible re-programming of logical to physical address mapping
- Registers ADDRMAPx are used to program the address mapper
- Please refer ADDRMAPx registers information in “Register Descriptions” section in the Programming Guide



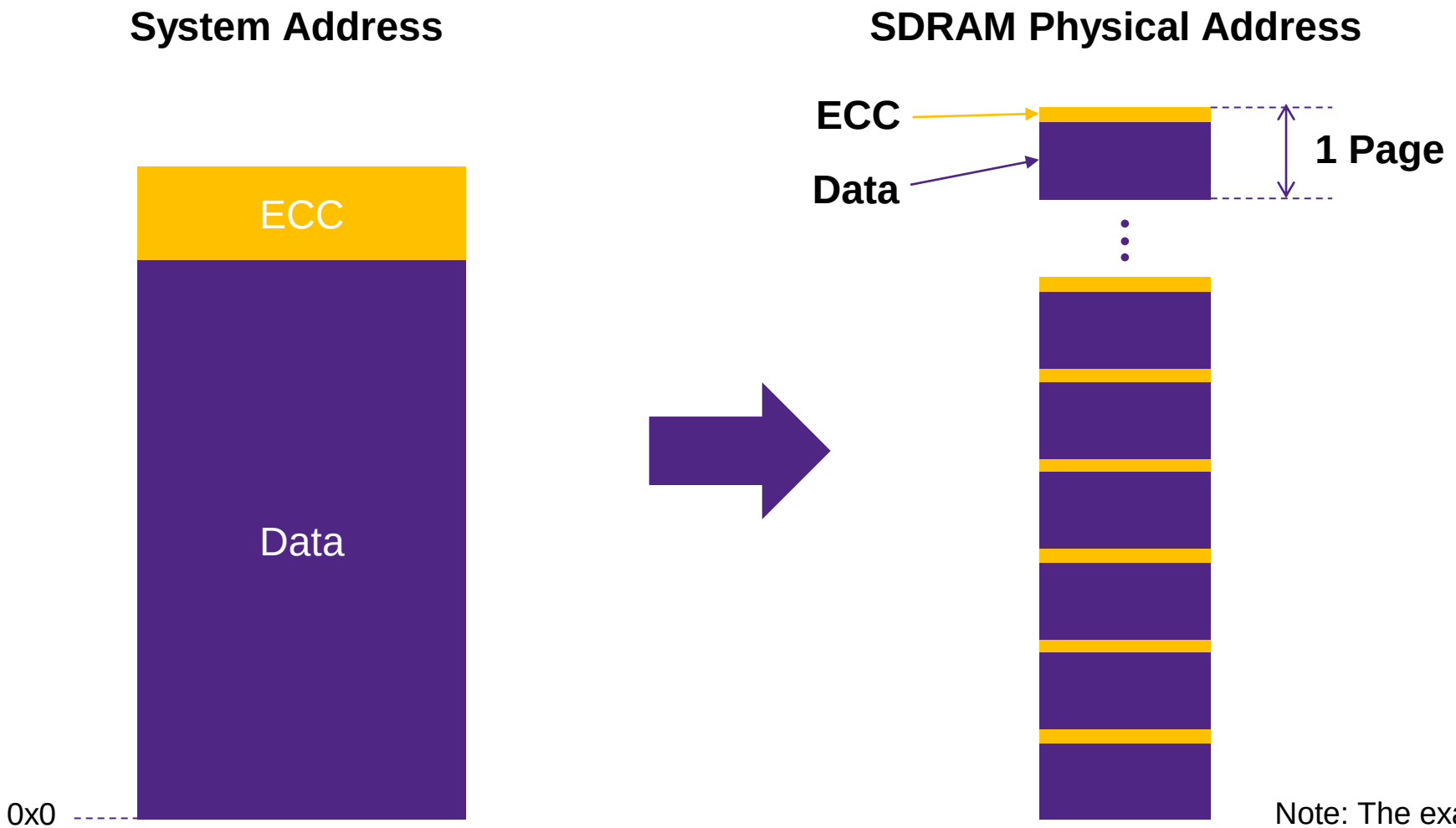
Inline ECC

Overview of Inline ECC

- Inline ECC is an alternative to sideband ECC
  - Unlike Sideband ECC, no dedicated devices for ECC parity are required
  - It's useful especially for LPDDR devices
- 64 bit data + 8 bit ECC parity (SECDED, hamming code)
  - Data : ECC ratio = 8 : 1
- Reserved upper 1/8 of each row for ECC parity for ease of implementation
  - Usable area is 7/8 = 87.5%
  - Utilization percentage is (7/8 + (1/8 x 7/8) = 63/64 = 98.4%



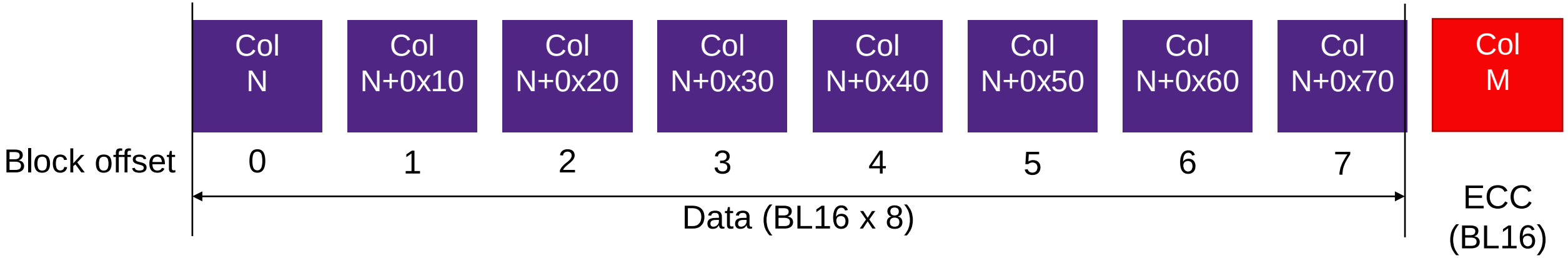
Address Mapping – System to SDRAM



Note: The example is for LPDDR4

ECC Block

- An ECC block contains 8 HIF commands (8 x BL16) + 1 ECC (BL16)
  - Data and ECC are on Same bank/Same Page



Example: BL16, number of column bits=10, N and M are the following

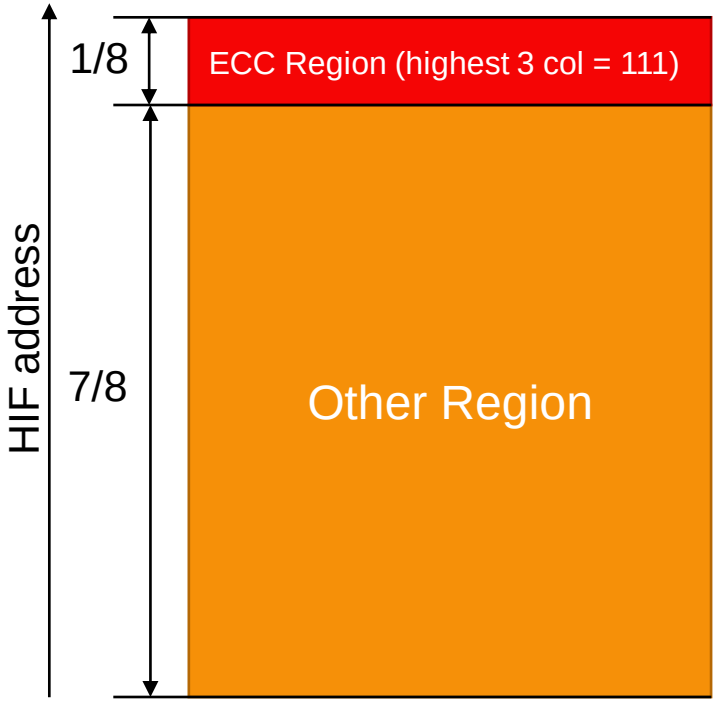
N	0x0	0x80	0x100	0x180	0x200	0x280	0x300
M	0x380	0x390	0x3A0	0x3B0	0x3C0	0x3D0	0x3E0

Note: Col=0x3F0 is unused

Address mapping constraints

- Highest 3 column bits must be mapped to the highest valid HIF address bits
- Lowest 4 column bits must be mapped to the lowest HIF address bits
- Column[6:4] address must be mapped between HIF[9] and HIF[4], which will be an offset address of an ECC block.

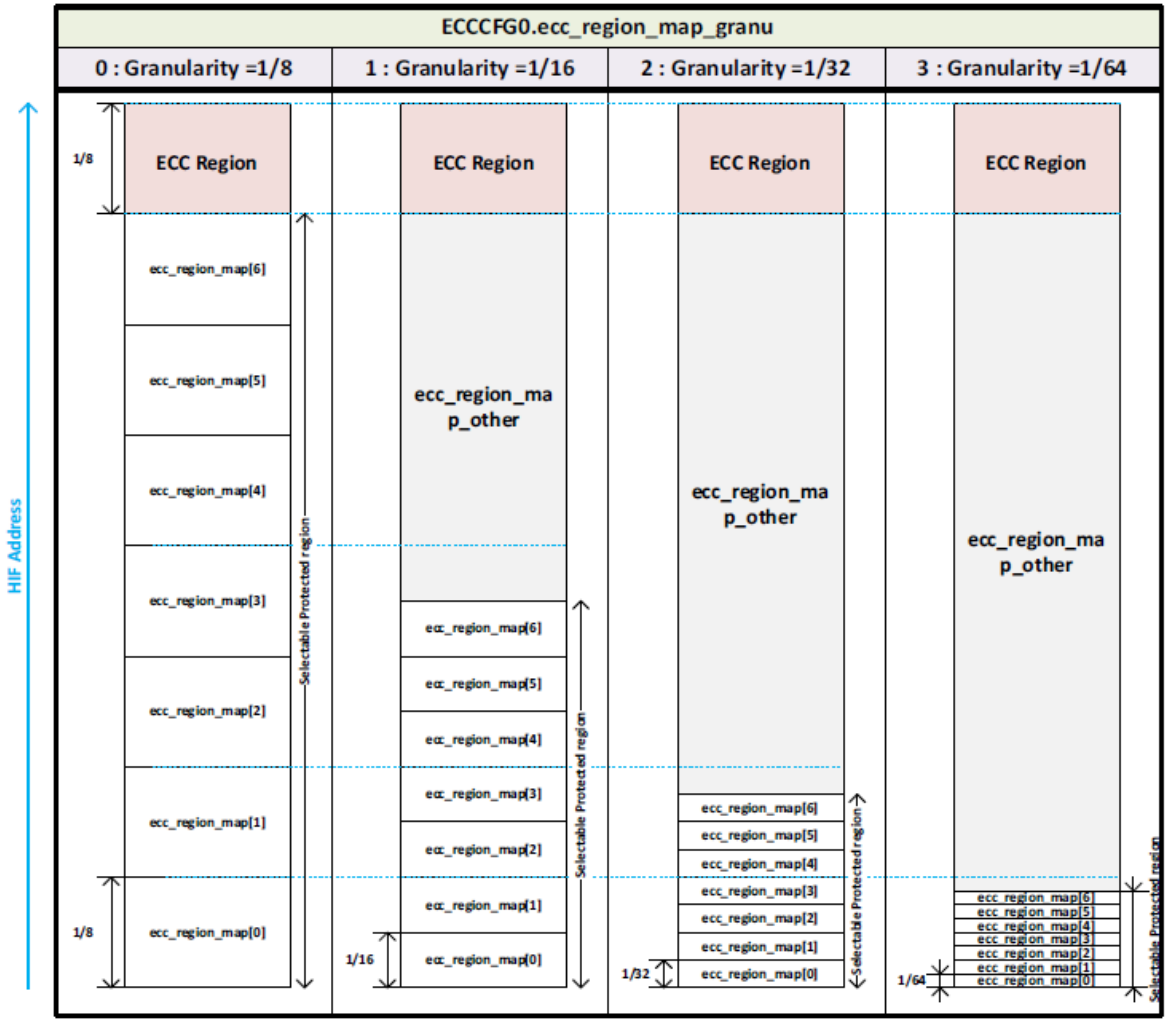
LPDDR5	NO BG rotation																		
				H0	H1	H2	13	12	11	10	9	8	7	6	5	4	3	2	1	0
				C9	C8	C7	R	R	R	BA1	BA0	BG1	BG0	C6	C5	C4	C3	C2	C1	C0
	2 BG rotation																		
				H0	H1	H2	13	12	11	10	9	8	7	6	5	4	3	2	1	0
				C9	C8	C7	R	R	R	BA1	BA0	BG1	C6	C5	C4	BG0	C3	C2	C1	C0
	4 BG rotation																		
				H0	H1	H2	13	12	11	10	9	8	7	6	5	4	3	2	1	0
				C9	C8	C7	R	R	R	BA1	BA0	C6	C5	C4	BG1	BG0	C3	C2	C1	C0
	8 Bank rotation (don't need, not realistic)																		
				H0	H1	H2	13	12	11	10	9	8	7	6	5	4	3	2	1	0
				C9	C8	C7	R	R	R	BA1	C6	C5	C4	BA0	BG1	BG0	C3	C2	C1	C0



- This map is to reserve top 1/8 of page to be reserved (ECC region)

Selectable Protected Regions

- The controller provides Selectable Protected Regions
  - Protected region
    - Protected by SECDED ECC
    - Overhead commands are injected
  - Non-Protected region
    - No overhead commands are injected
- Depends on system/application, use protected/unprotected region selectively
  - e.g.
    - Only 1/64 area to be protected
    - Only 1/64 area to be un-unprotected
- This is configured by the following registers
  - ECCCFG0.ecc_region_map_granu
  - ECCCFG0.ecc_region_map



LPDDR5 Update

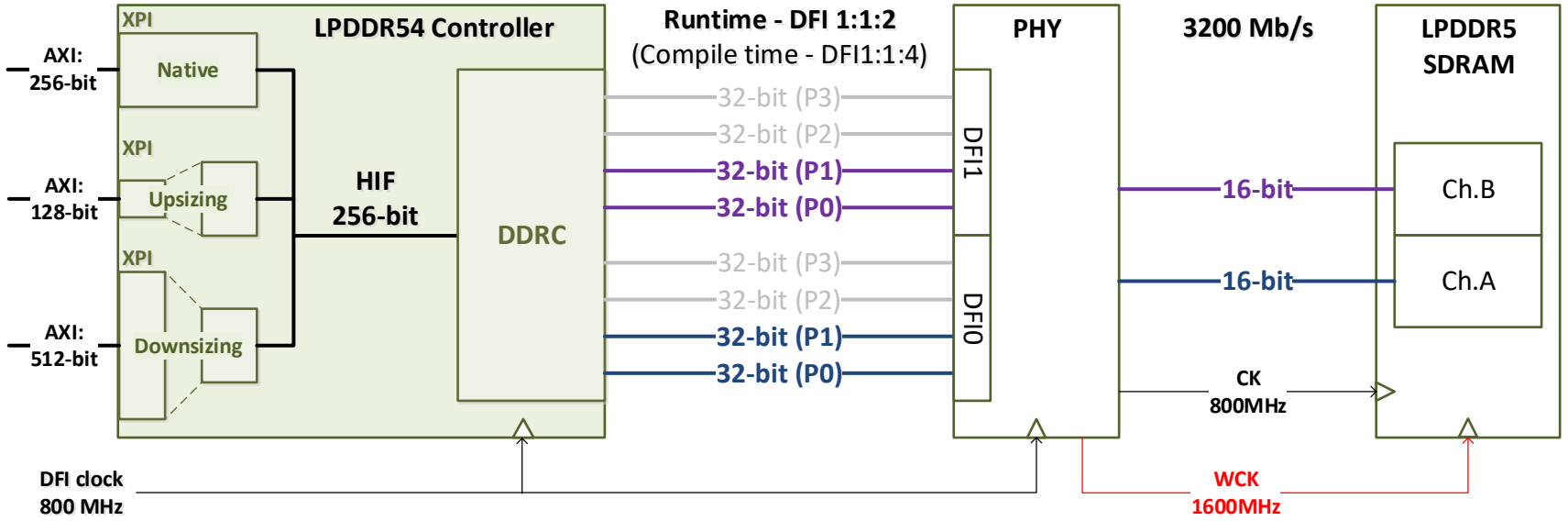
Architecture – Supported DFI frequency ratio

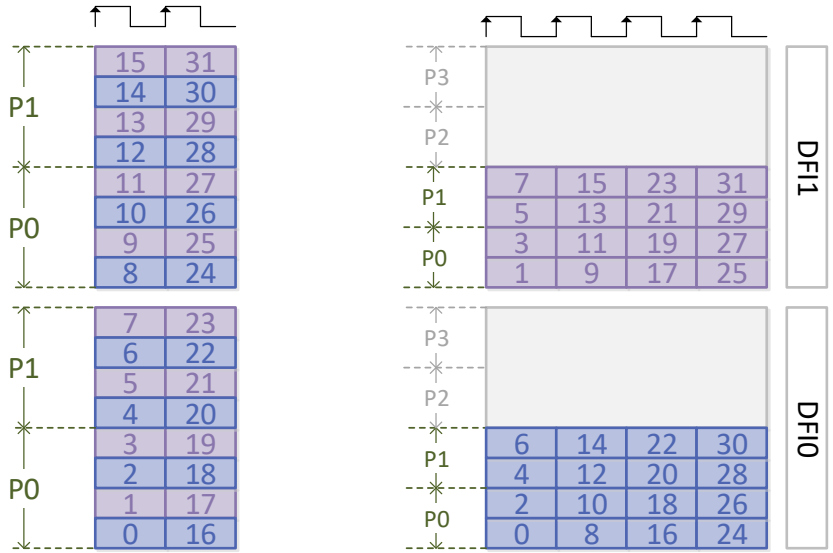
		DFI frequency ratio			
	HW config (Compile time)	**1:1**	**1:2**	**1:4**	
	SW setting (Runtime)	**n/a**	**1:2 mode**	**1:2 mode**	**1:4 mode**
LPDDR54 Controller	LPDDR4	No longer supported	Not supported	Supported	Supported
	LPDDR5	No longer supported	Not supported	Supported	Supported

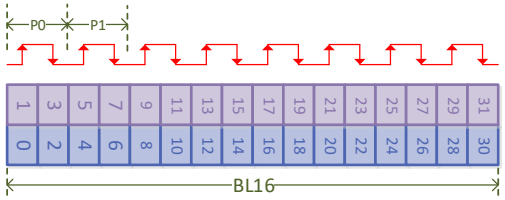
DFI 1:4 configuration has been added in LPDDR5/4/4X Controller

Note - For LPDDR5, 1:2 means “1:1:2”, and 1:4 means “1:1:4”

Architecture – LPDDR5 DFI 1:1:2 Freq Ratio (FBW)



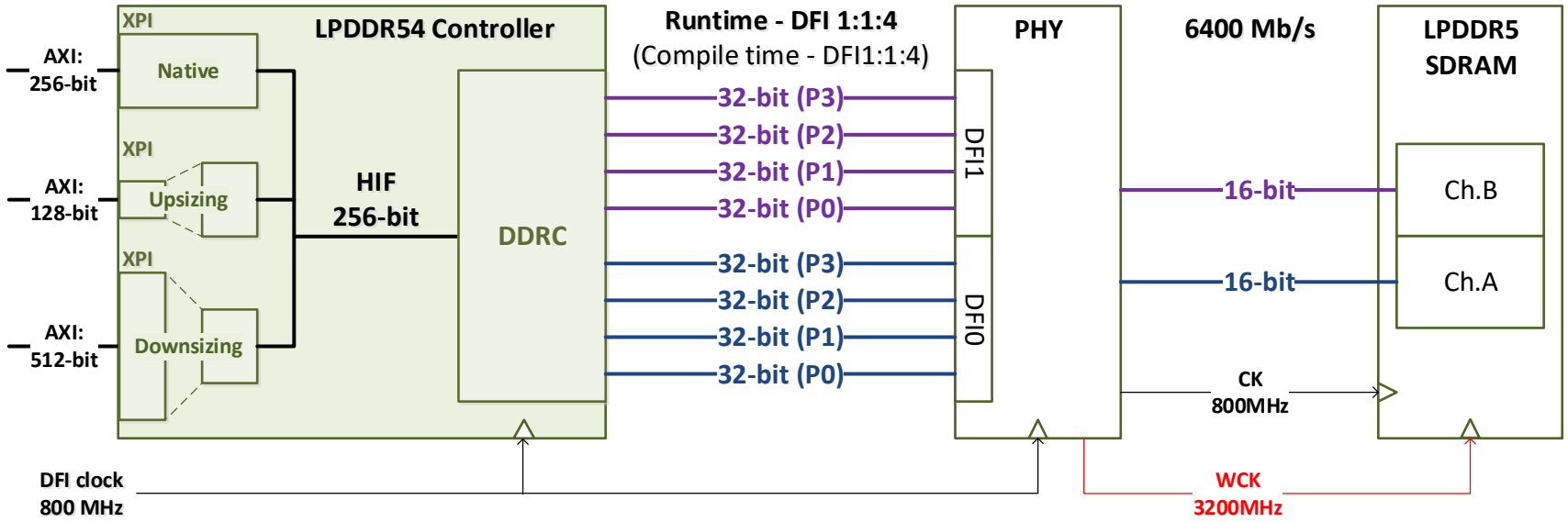


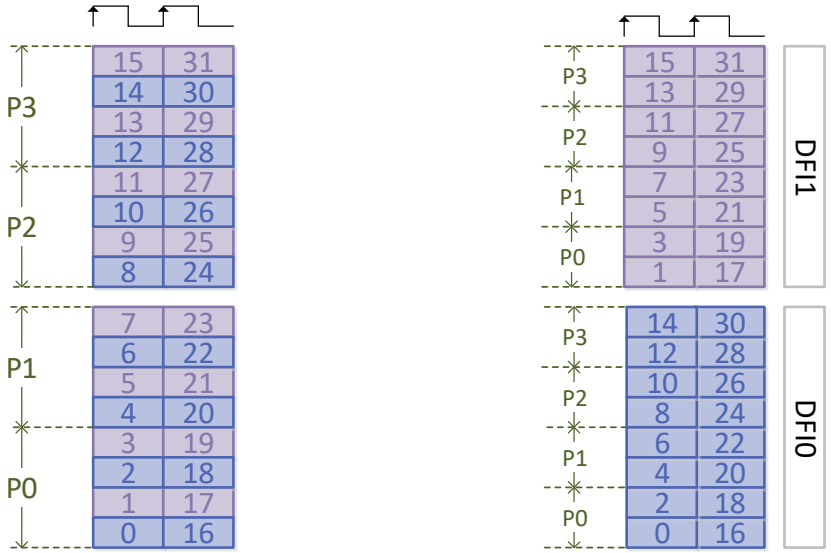


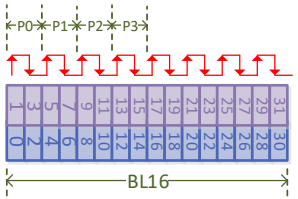
Note that hardware configuration is DFI 1:1:4, but 1:1:2 mode is used (DFI1:1:2 configuration is not supported)

Note that LPDDR4 is also quite similar (but it is called as DFI 1:2 mode, there is no WCK)

Architecture – LPDDR5 DFI 1:1:4 Freq Ratio (FBW)







LPDDR5 features

- Three bank architectures
  - ✓ BG Mode = 4 banks, 4 bank groups
  - ✓ 8B Mode = 8 banks, no bank groups (This is not supported in LPDDR5/4 controller)
  - ✓ 16B Mode = 16 banks, no bank groups

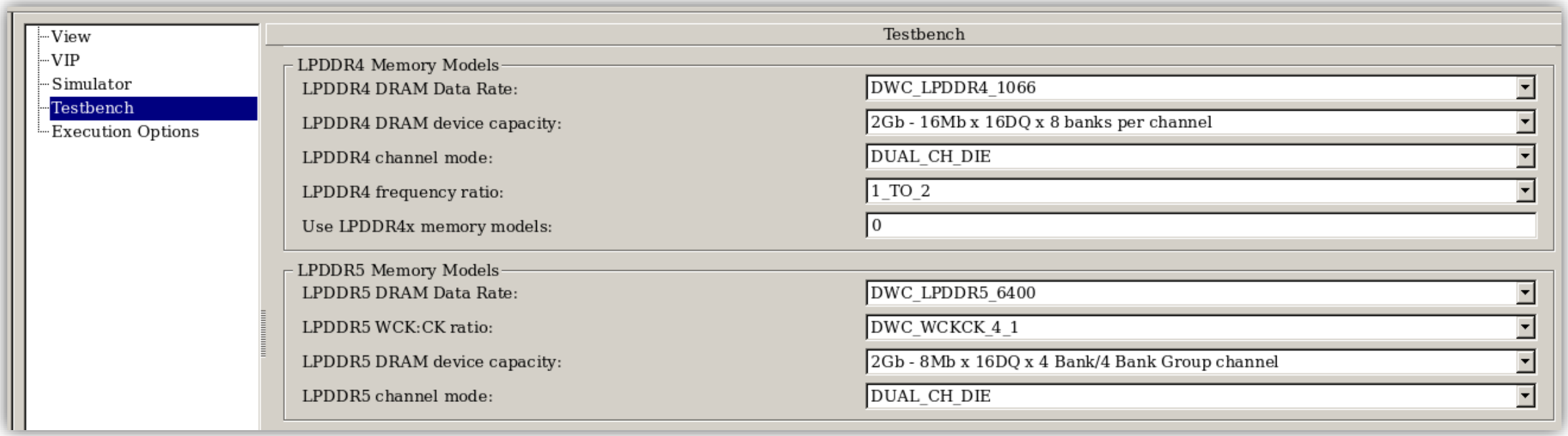
DFI:CK:WCK ratio	Bank organization	Data rate	WCK frequency	CK frequency
1:1:2	16B mode	<=3200 Mbps	<=1600 MHz	<=800 MHz
1:1:4	16B mode	<=3200 Mbps	<=1600 MHz	<=400 MHz
	BG mode	> 3200 Mbps	>1600 MHz	>400MHz

- WCK clocking
  - ✓ WCK always ON mode can be enabled or disabled – both are supported
- MPC ZQCal Latch command is periodically sent to support LPDDR5 background ZQ calibration
  - ✓ Command base ZQCal Start/Latch is NOT supported for LPDDR5
  - ✓ Command base ZQCal Start/Latch is supported only for LPDDR4 (same as uMCTL2)
- Refresh command interval is adjusted according to MR4 refresh rate in SDRAM, by issuing MRR4 periodically
- Software initiated Deep Sleep Mode (DSM) is supported

PVE Customer Verification Environment

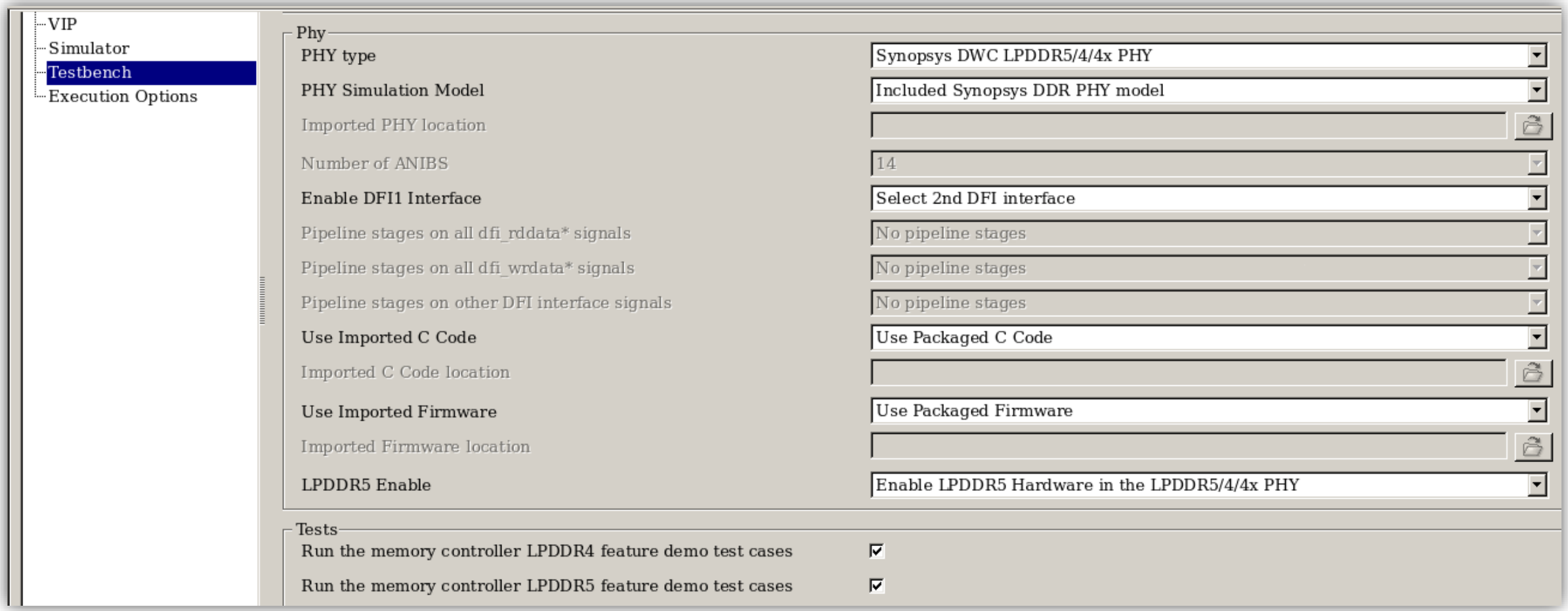
coreConsultant GUI

Setup and Run Simulations
- Memory Models



coreConsultant GUI

- PHY and Tests

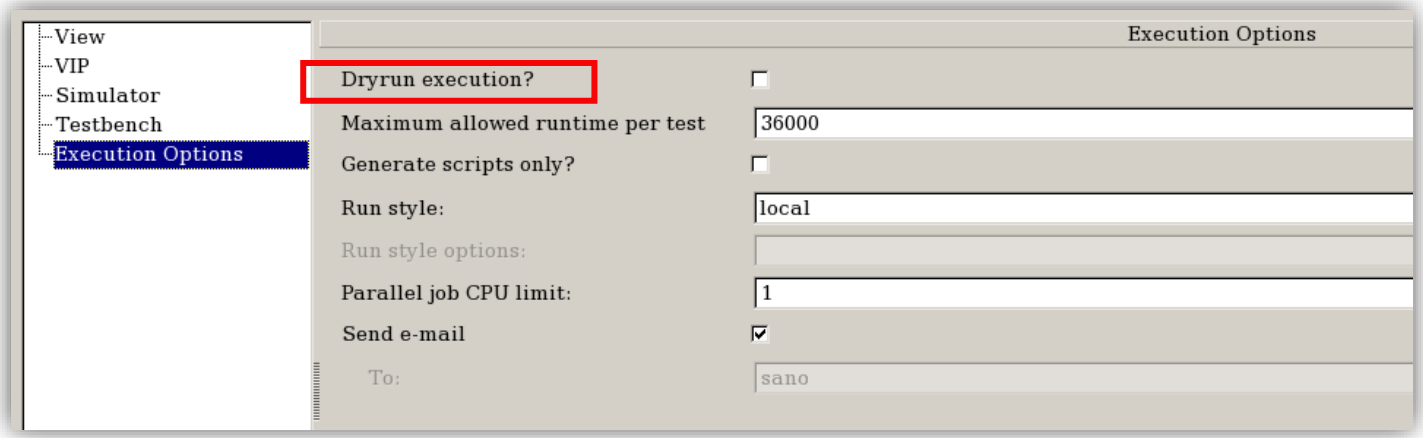


Running IP testcase

- Once you select all simulation options from GUI, click Apply to start simulation jobs
  - However all testcase run
- If you want to run specific testcase, please check “Dryrun execution?”
  - This populates the workspace but does not execute any VCS simulations
- When the workspace directory is populated with a configured design, each test can be run from the command line by using the following simple steps

% cd <path_to_cC_workspace>/sim/<test_name>

% ./test.startsim -recompile -fsdb



coreConsultant GUI

Simulation Options

- Customer can select
  - Memory Models
    - LPDDR4 and LPDDR5 VIP memory models
  - Testbench Options
    - Disable/Enable assertions, CDC assertions (MISSAMPLES)
    - Code coverage
    - Waveform dumping etc
  - Execution Options
    - GRD, run style etc..
    - DryRun
  - PHY Options
    - PHY type, imported, encrypted, unencrypted etc
    - ANIBS
  - Test Selection
    - Tests are grouped by protocol

Not all options/combinations are supported in EA/LCA releases

Workspace

```
/workspace
    /src
    /export
    /scratch
    /doc
    /…
    /sim
        Makefile
        run.scr
        compile_vip.sh
        /testbench
        /sw_utilities
        /test_<>_shared_simv
        /test_dwc_ddrctl_*_lpddr4
        /test_dwc_ddrctl_*_lpddr5
            test.startsim
            test.sim_command
            test.plusargs
```

RTL Source

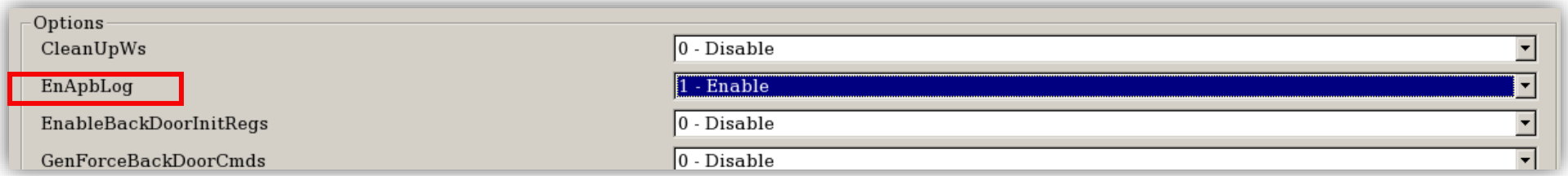
- Databook
- Programming Guide
- User Guide
- Installation Guide
- Release Notes

All testbench source code

Testcase/Compile directory for each protocol

Programming Registers for SoC

- **dwc_ddrctl_apb.log**: Log file of all APB accesses to the LPDDR5/4/4X Controller and all configuration register accesses to the PUB which have occurred during the simulation
- Please enable EnApbLog from Testbench Options



- Example of dwc_ddrctl_apb.log file

APB MON TRACE FILE

Time	PADDR	R/W	DATA	BLOCK	REGISTER	FIELDS	FIELD_VALUE
Attention: presetn is asserted at 279547 ps!!!!!							
333561 ps	0x000010ff8	READ	0x03130312a	DWC_ddrctl_map_REGB_DDRC_CH0	DDRCTL_VER_NUMBER	ver_number	0x03130312a
348565 ps	0x000010ffc	READ	0x08c633031	DWC_ddrctl_map_REGB_DDRC_CH0	DDRCTL_VER_TYPE	ver_type	0x08c633031
378572 ps	0x000010b84	WRITE	0x000000001	DWC_ddrctl_map_REGB_DDRC_CH0	OPCTRL1	dis_dq	0x000000001
378572 ps	0x000010b84	WRITE	0x000000001	DWC_ddrctl_map_REGB_DDRC_CH0	OPCTRL1	dis_hif	0x000000000
420583 ps	0x000010000	WRITE	0x000080002	DWC_ddrctl_map_REGB_DDRC_CH0	MSTR0	lpddr4	0x000000001
420583 ps	0x000010000	WRITE	0x000080002	DWC_ddrctl_map_REGB_DDRC_CH0	MSTR0	lpddr5	0x000000000
420583 ps	0x000010000	WRITE	0x000080002	DWC_ddrctl_map_REGB_DDRC_CH0	MSTR0	en_2t_timing_mode	0x000000000
420583 ps	0x000010000	WRITE	0x000080002	DWC_ddrctl_map_REGB_DDRC_CH0	MSTR0	data_bus_width	0x000000000
420583 ps	0x000010000	WRITE	0x000080002	DWC_ddrctl_map_REGB_DDRC_CH0	MSTR0	burst_rdwr	0x000000008
450590 ps	0x000010100	WRITE	0x000000000	DWC_ddrctl_map_REGB_DDRC_CH0	DERATECTL0	derate_enable	0x000000000
450590 ps	0x000010100	WRITE	0x000000000	DWC_ddrctl_map_REGB_DDRC_CH0	DERATECTL0	lpddr4_refresh_mode	0x000000000
450590 ps	0x000010100	WRITE	0x000000000	DWC_ddrctl_map_REGB_DDRC_CH0	DERATECTL0	derate_mr4_pause_fc	0x000000000
450590 ps	0x000010100	WRITE	0x000000000	DWC_ddrctl_map_REGB_DDRC_CH0	DERATECTL0	dis_trefi_x6x8	0x000000000
450590 ps	0x000010100	WRITE	0x000000000	DWC_ddrctl_map_REGB_DDRC_CH0	DERATECTL0	dis_trefi_x0125	0x000000000
480598 ps	0x000010104	WRITE	0x00000000f	DWC_ddrctl_map_REGB_DDRC_CH0	DERATECTL1	active_derate_byte_rank0	0x00000000f
492601 ps	0x000010118	WRITE	0x000000001	DWC_ddrctl_map_REGB_DDRC_CH0	DERATECTL6	derate_mr4_tuf_dis	0x000000001
522608 ps	0x000010180	WRITE	0x000020010	DWC_ddrctl_map_REGB_DDRC_CH0	PWRCTL	selfref_en	0x000000000
522608 ps	0x000010180	WRITE	0x000020010	DWC_ddrctl_map_REGB_DDRC_CH0	PWRCTL	powerdown_en	0x000000001
522608 ps	0x000010180	WRITE	0x000020010	DWC_ddrctl_map_REGB_DDRC_CH0	PWRCTL	en_dfi_dram_clk_disable	0x000000000

  - *Please note APB address of PHY in the log file has offset due to Testbench limitation*

DFI connection between Memory Controller and DDR PHY

- Testbench file <ddr_chip.v> is a wrapper file in IP testcase
- Please refer ddr_chip.v as a reference to build DDR subsystem for your SoC environment

- Path : <Workspace>/sim/testbench/modules/ddr_chip.v
  - This gets generated after you run a simulation in coreConsultant environment

SYNOPSYS®
Silicon to Software™

Thank You