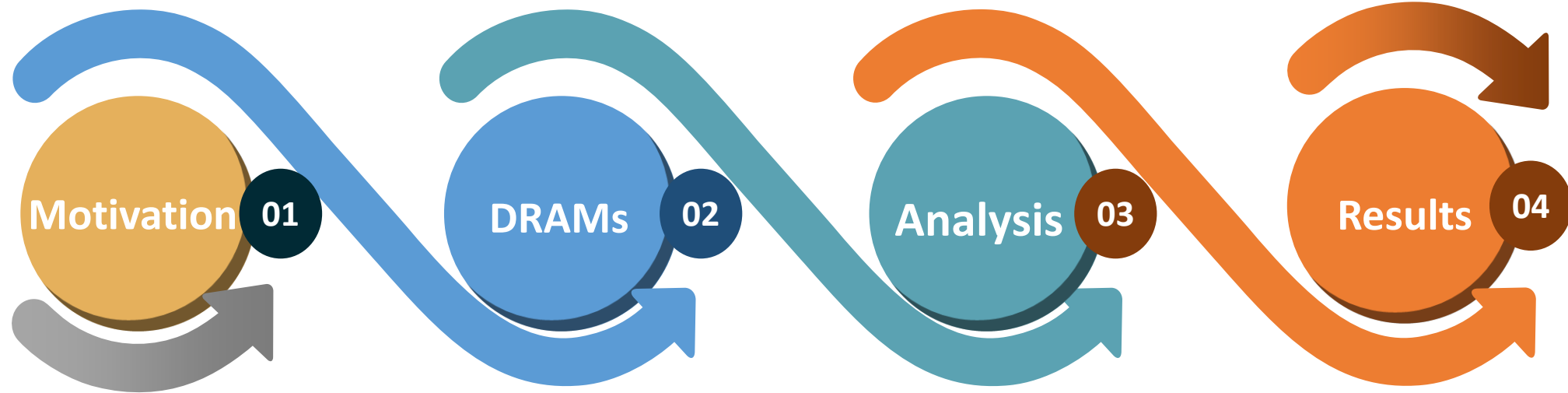


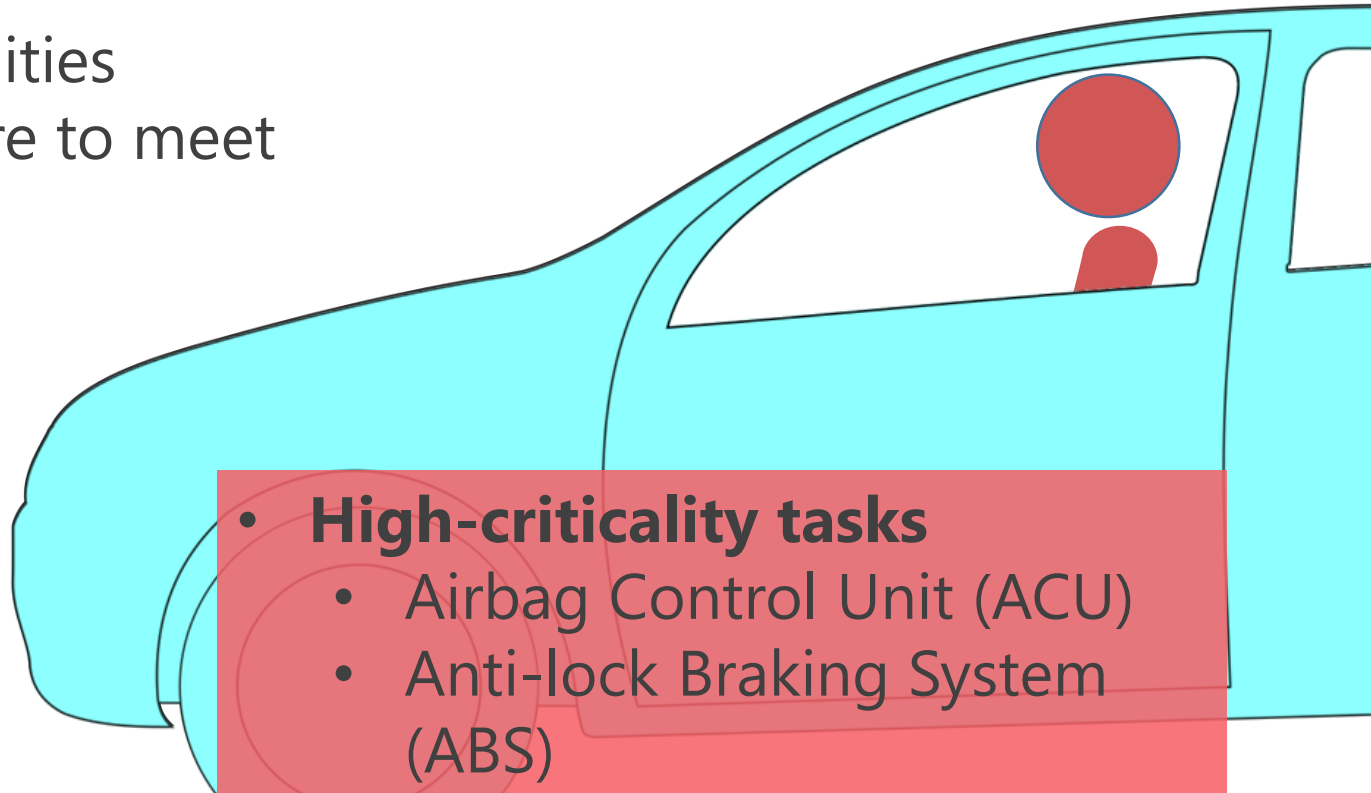
# Managing DRAM Interference in Mixed Criticality Embedded Systems

**Mohamed Hassan**



# Outline

- **Emerging Systems No longer solely hosting isolated safety-critical tasks**
  - Execute tasks with different criticalities
  - Criticality  $\alpha$  consequences of failure to meet requirements

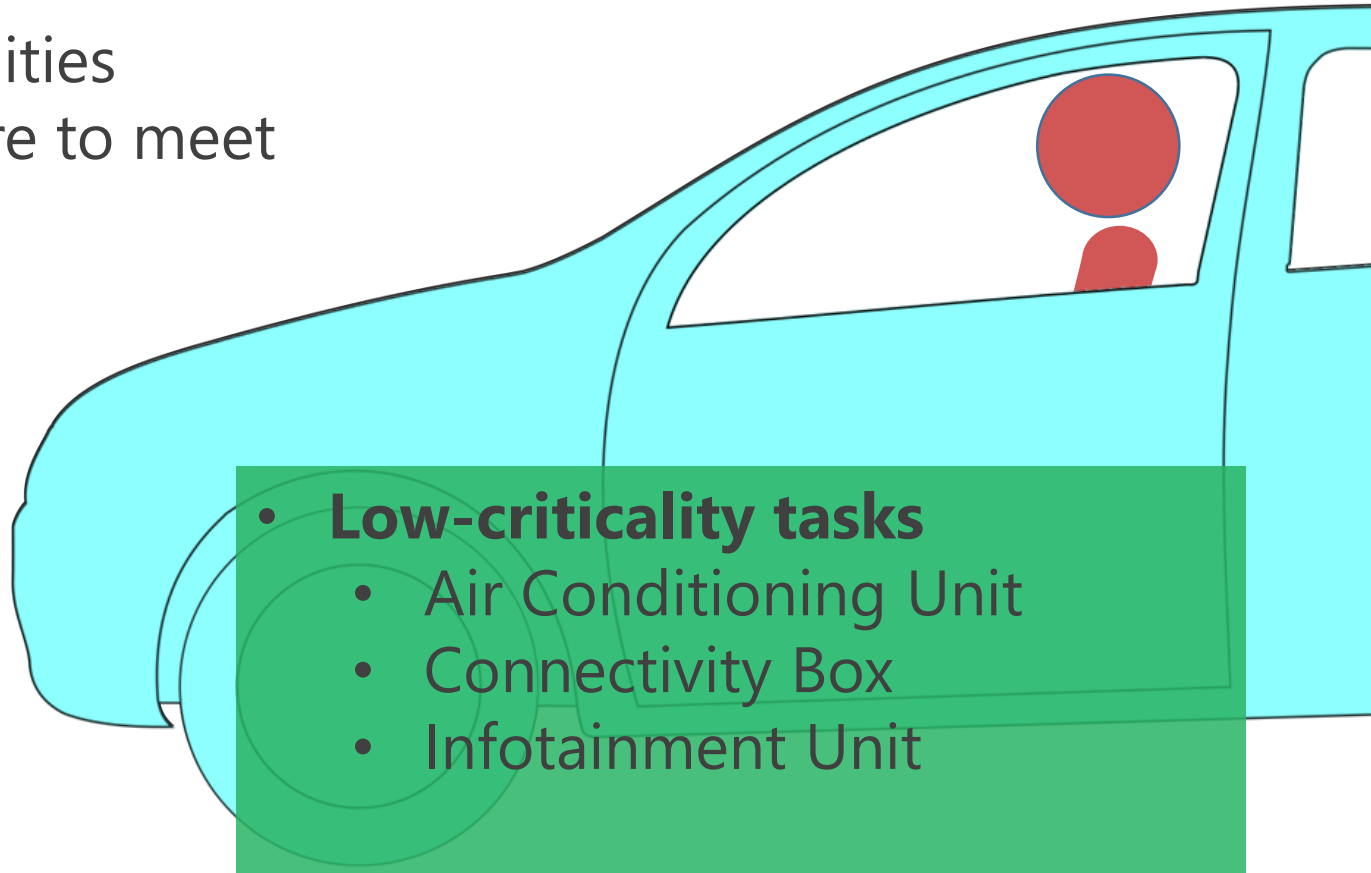


- **High-criticality tasks**
  - Airbag Control Unit (ACU)
  - Anti-lock Braking System (ABS)
  - Engine Control Unit (ECU)

# Mixed Criticality Systems

MOTIVATION

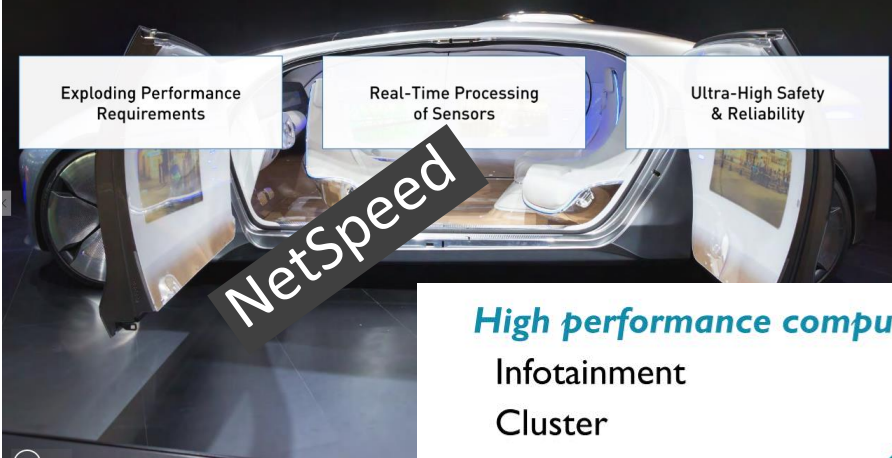
- **Emerging Systems No longer solely hosting isolated safety-critical tasks**
  - Execute tasks with different criticalities
  - Criticality  $\alpha$  consequences of failure to meet requirements



- **Low-criticality tasks**
  - Air Conditioning Unit
  - Connectivity Box
  - Infotainment Unit

# Mixed Criticality Systems

## Challenges Facing Autonomous Vehicles



### High performance compute

- Infotainment
- Cluster
- Driver assist
- Vehicle interface
- User experience



Cost

Quality

Ecosystem

Temperature

18 ©ARM 2016

## Key Requirements of Automotive-Grade IP

Reduce Risk and Accelerate Qualification for Automotive SoCs

3

- Functional Safety**: Accelerate ISO 26262 functional safety assessments to help ensure designers reach target ASIL levels
- Reliability**: Reduce risk & develop AEC-Q100 qualified IP for automotive applications

Synopsys



### Real-time control

- Safe
- Secure
- Responsive
- Reliable
- Fast boot

synopsys



ARM

ARM

# Mixed Criticality Systems

MOTIVATION

# Complementary SoC processor requirements

## High performance compute

- Infotainment
- Cluster
- Driver assist
- Vehicle interface
- User experience

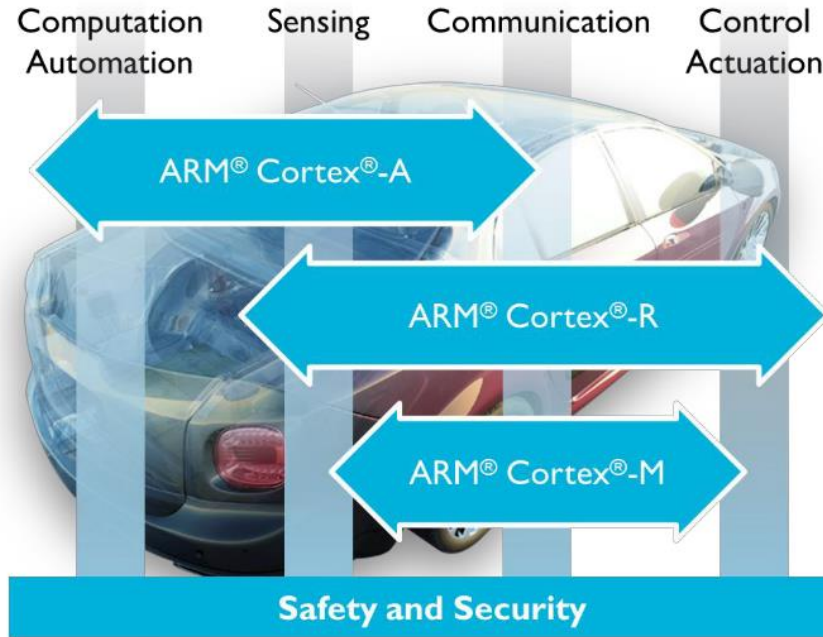
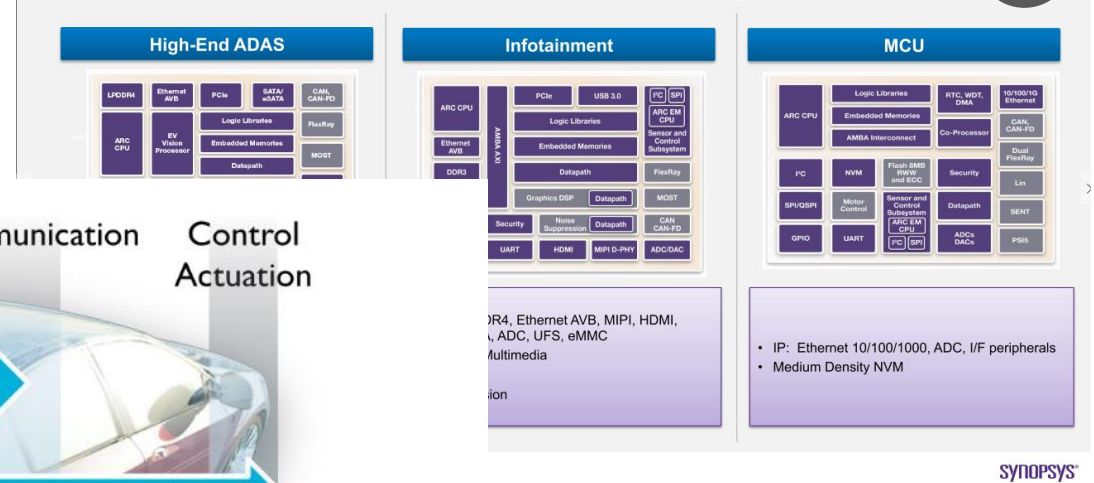


Real-time control

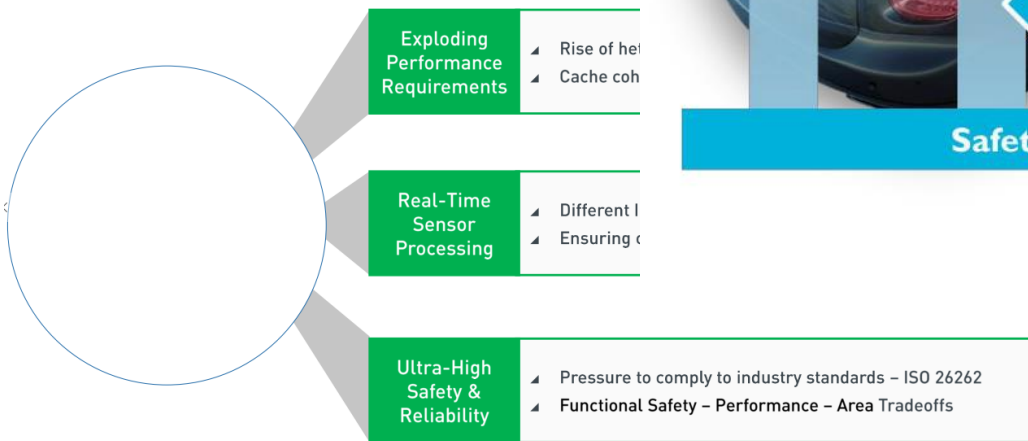


Cost Quality Ecosystem

# Automotive Applications Require Different SoC Architectures

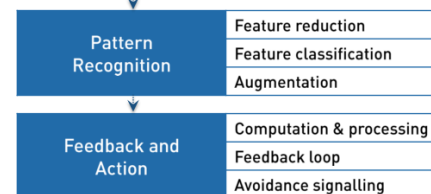
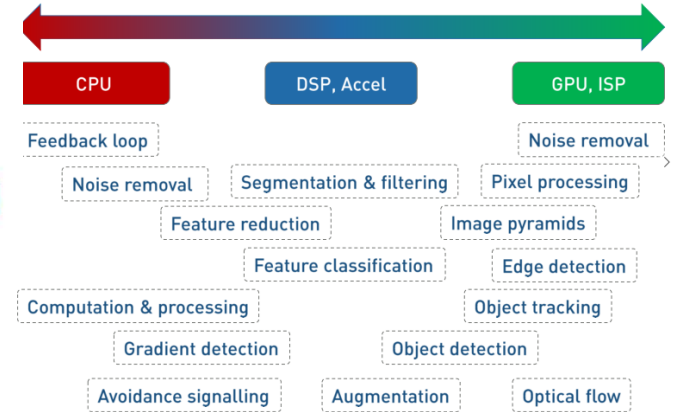


## Translating System-Level Requirements



## Computing

- Smaller amounts of data
- Highly structured data
- Complex computation/item
- Lots of data
- Simple computation/item
- Massive parallelism




**ARM**<sup>®</sup> Application Processors  
**Cortex**<sup>™</sup> A53  
64-bit Quad-Core with Virtualization

 **Power Management**  
Multiple Power Domains  
Power Gated Islands

**ARM**<sup>®</sup> Real-Time Processors  
**Cortex**<sup>™</sup> R5  
32-bit Dual-Core Application Offload


 **Safety & Reliability**  
IEC61508, ISO26262  
System Isolation & Error Mitigation, Lockstep

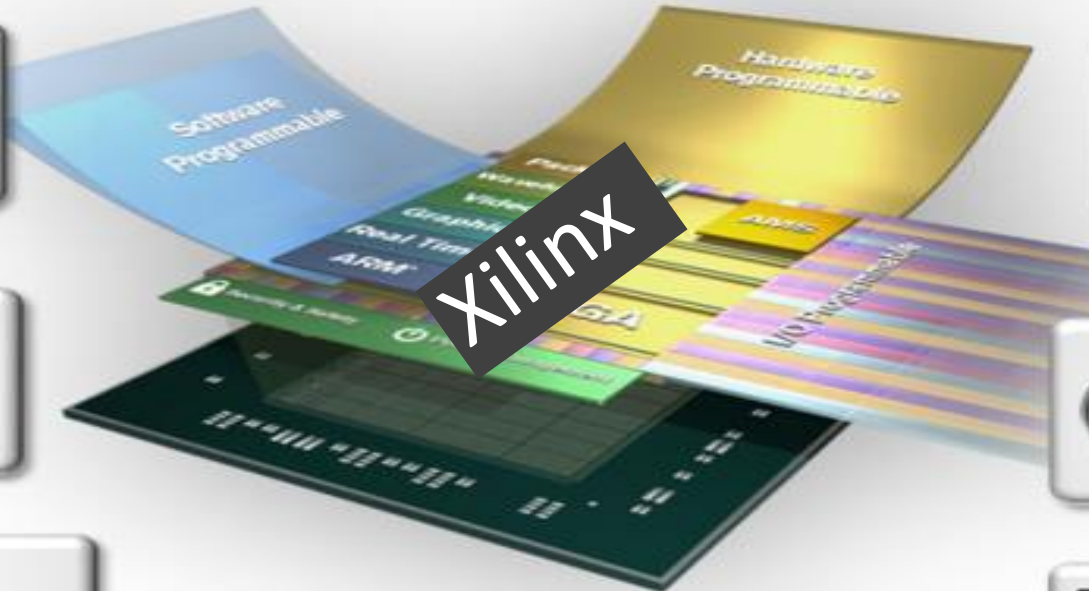
**mali**  
**H.265 HEVC**  
Graphics/Video  
ARM Mali-400MP  
H.265/264 CODECs

 **Security**  
Information Assurance, Trust, Anti-Tamper, TrustZone  
Key and Vault Management

 **UltraScale FPGA Logic**  
UltraRAM, PCIe Gen4, 100G Ethernet, AMS

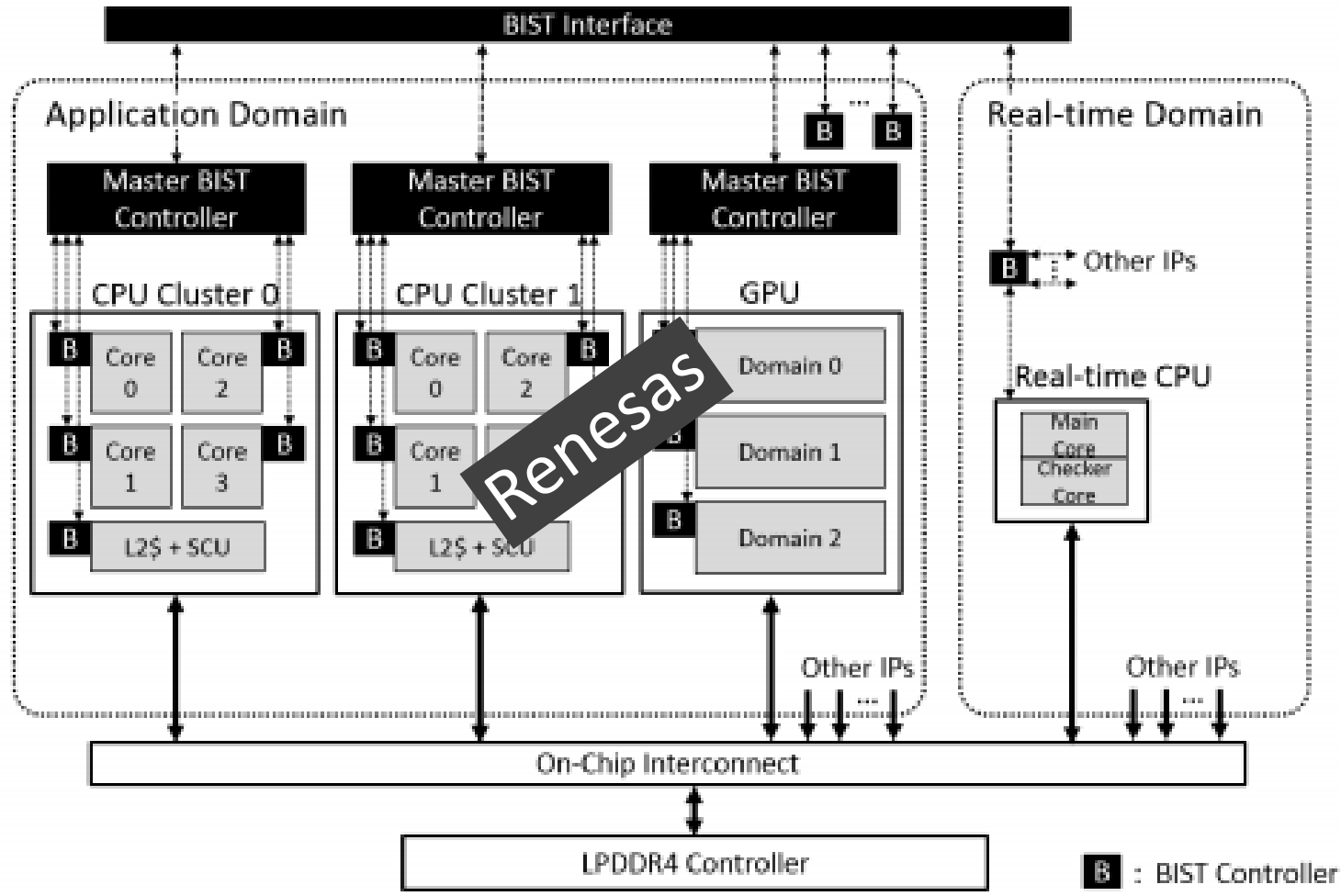
 **Runtime SW & Tools**  
OS, RTOS, AMP, Hypervisor Development, Heterogeneous Debug, Hardware/Software Profiling & Performance Analysis

 **High Speed Peripherals**  
USB 3.0, PCIe Gen2, GbE  
SATA3.0, DisplayPort



# Heterogenous MPSoCs with Real-time Processors

MOTIVATION

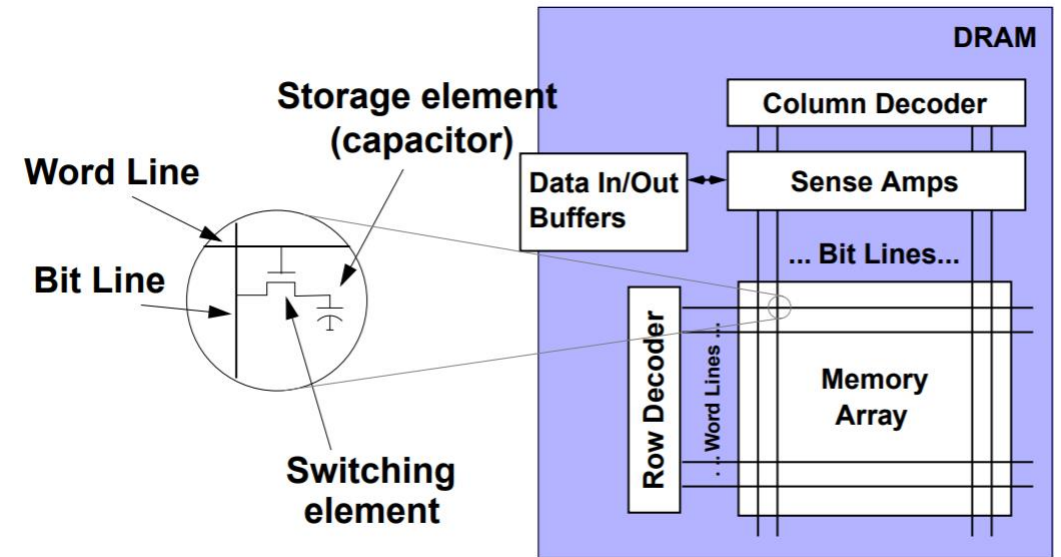


# Heterogenous MPSoCs with Real-time Processors

MOTIVATION



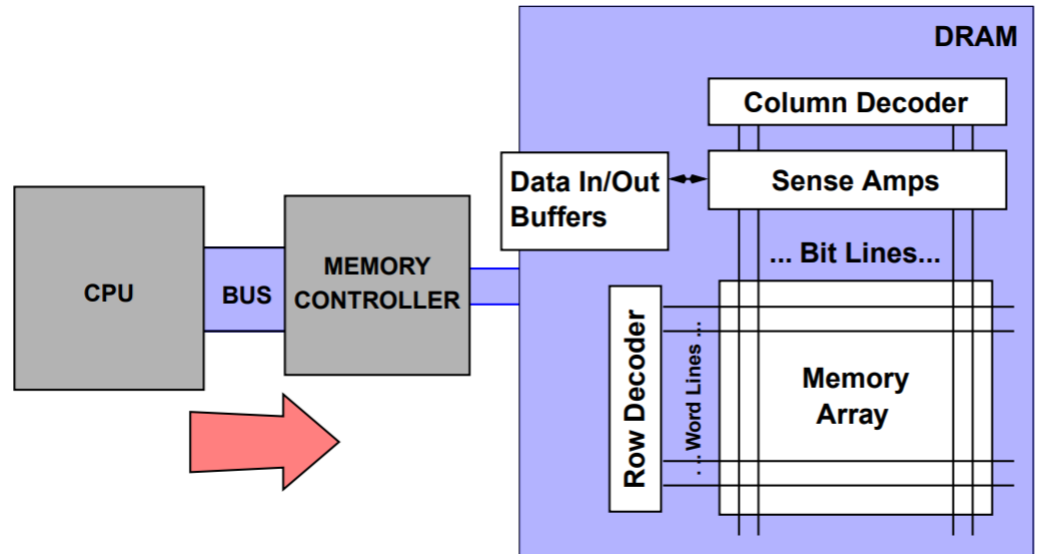
- DRAM Consists of multiple banks



# Background

# DRAM

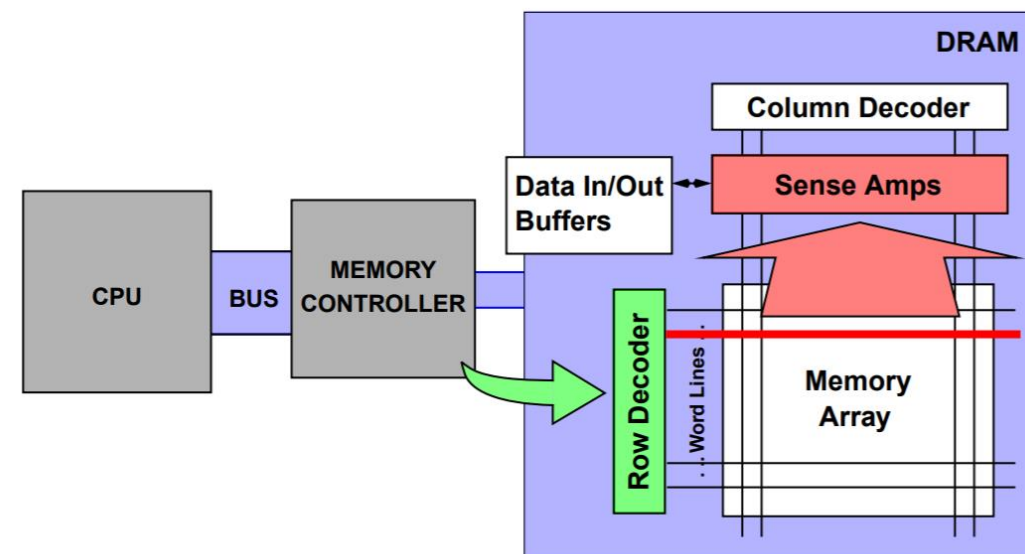
- DRAM Consists of multiple banks
- The memory controller (MC) manages accesses to DRAM



# Background

# DRAM

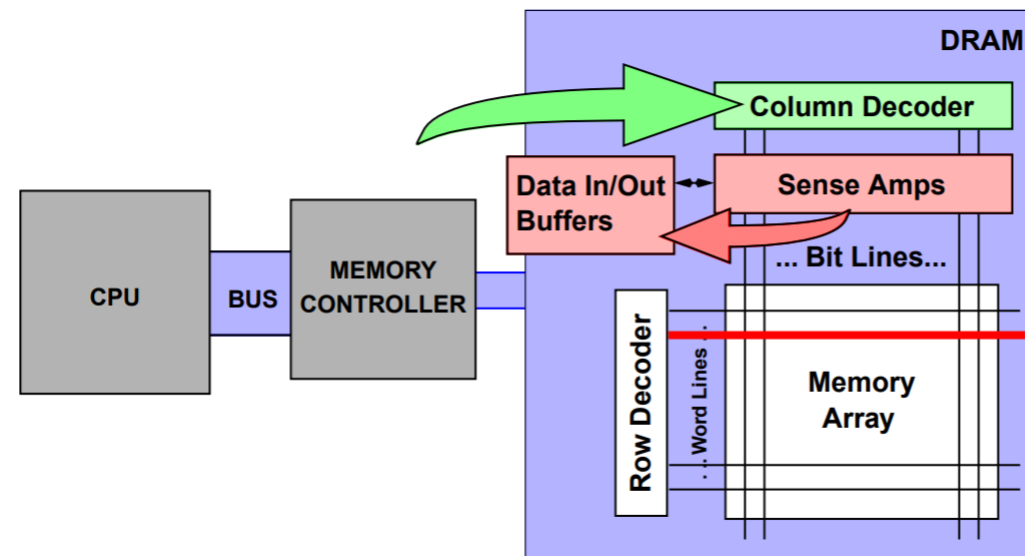
- DRAM Consists of multiple banks
- The memory controller (MC) manages accesses to DRAM
- A request in general consists of:
  - ACTIVATE command:
    - Bring data row from cells into sense amplifiers



# Background

# DRAM

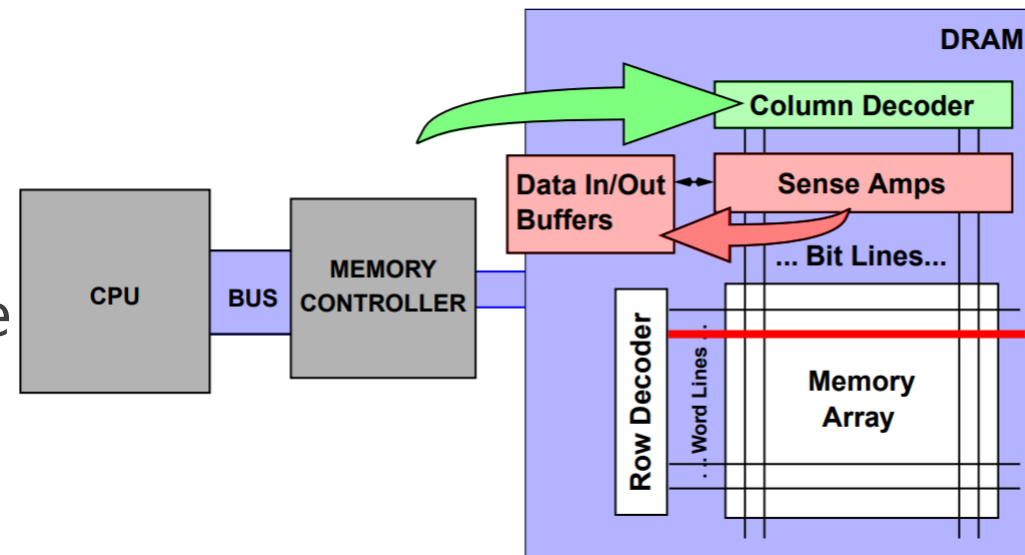
- DRAM Consists of multiple banks
- The memory controller (MC) manages accesses to DRAM
- A request in general consists of:
  - ACTIVATE command:
    - Bring data row from cells into sense amplifiers
  - RD/WR commands:
    - To read/write from specific columns in the sense amplifiers



# Background

# DRAM

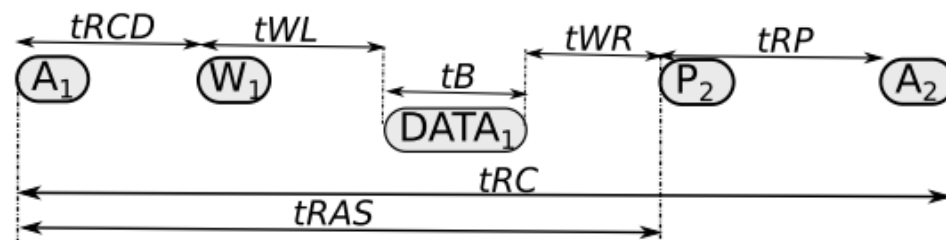
- DRAM Consists of multiple banks
- The memory controller (MC) manages accesses to DRAM
- A request in general consists of:
  - ACTIVATE command:
    - Bring data row from cells into sense amplifiers
  - RD/WR commands:
    - To read/write from specific columns in the sense amplifiers
  - PRECHARGE command:
    - to write back a previous row in the sense amplifiers before bringing the new one

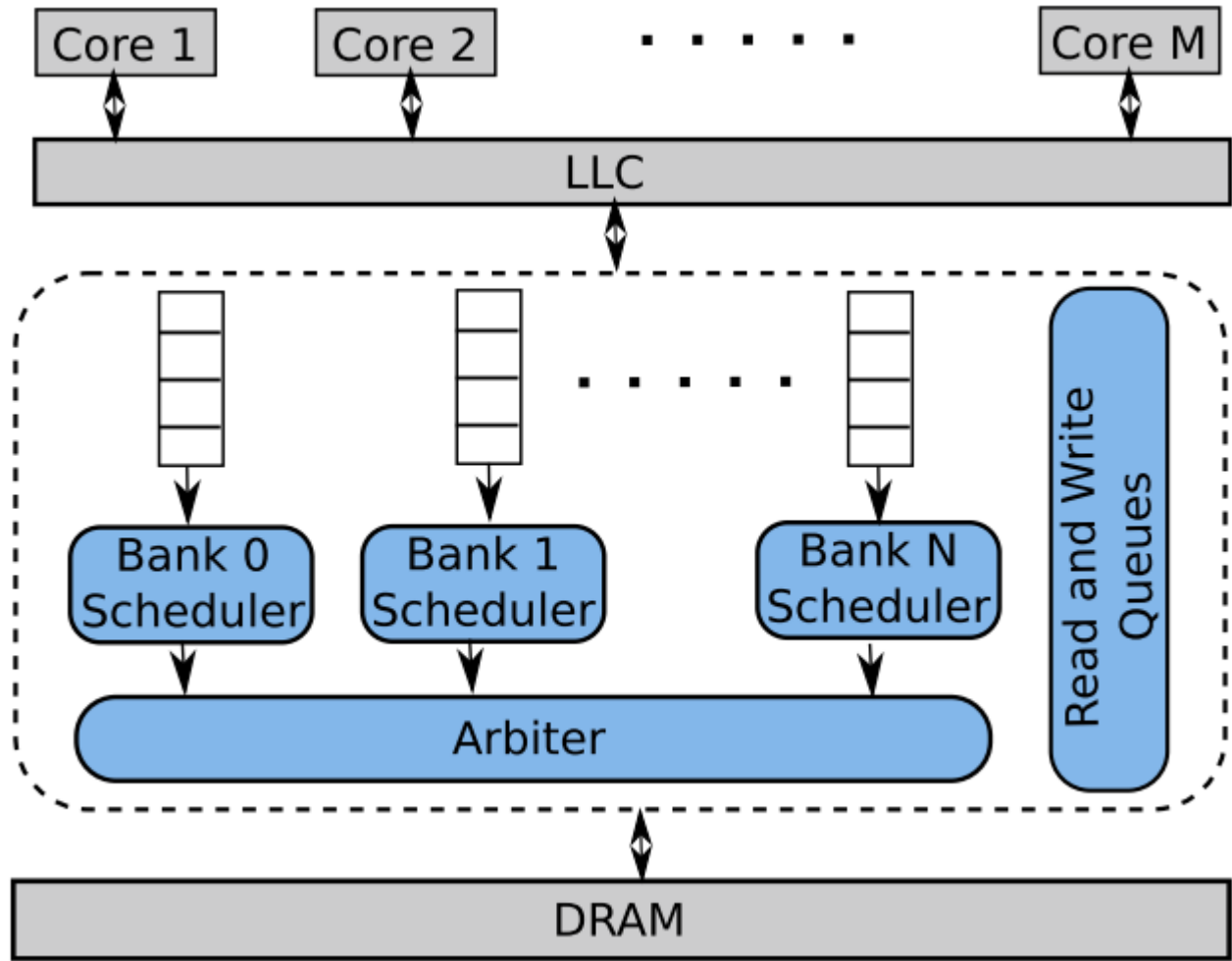


# Background

# DRAM

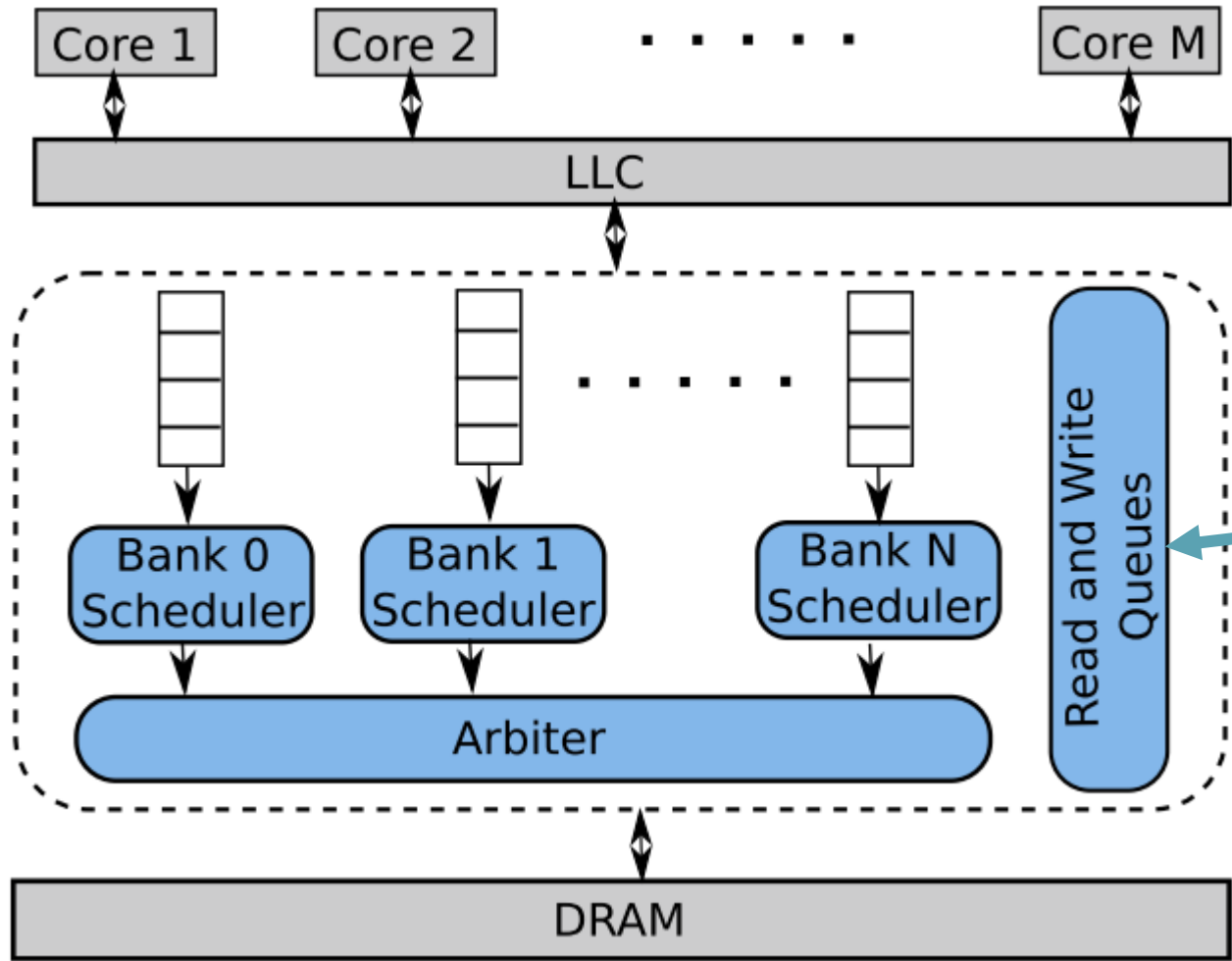
- DRAM Consists of multiple banks
- The memory controller (MC) manages accesses to DRAM
- A request in general consists of:
  - ACTIVATE command:
    - Bring data row from cells into sense amplifiers
  - RD/WR commands:
    - To read/write from specific columns in the sense amplifiers
  - PRECHARGE command:
    - to write back a previous row in the sense amplifiers before bringing the new one
- All commands have associated timing constraints that have to be satisfied by the controller





# Background: Memory Controller

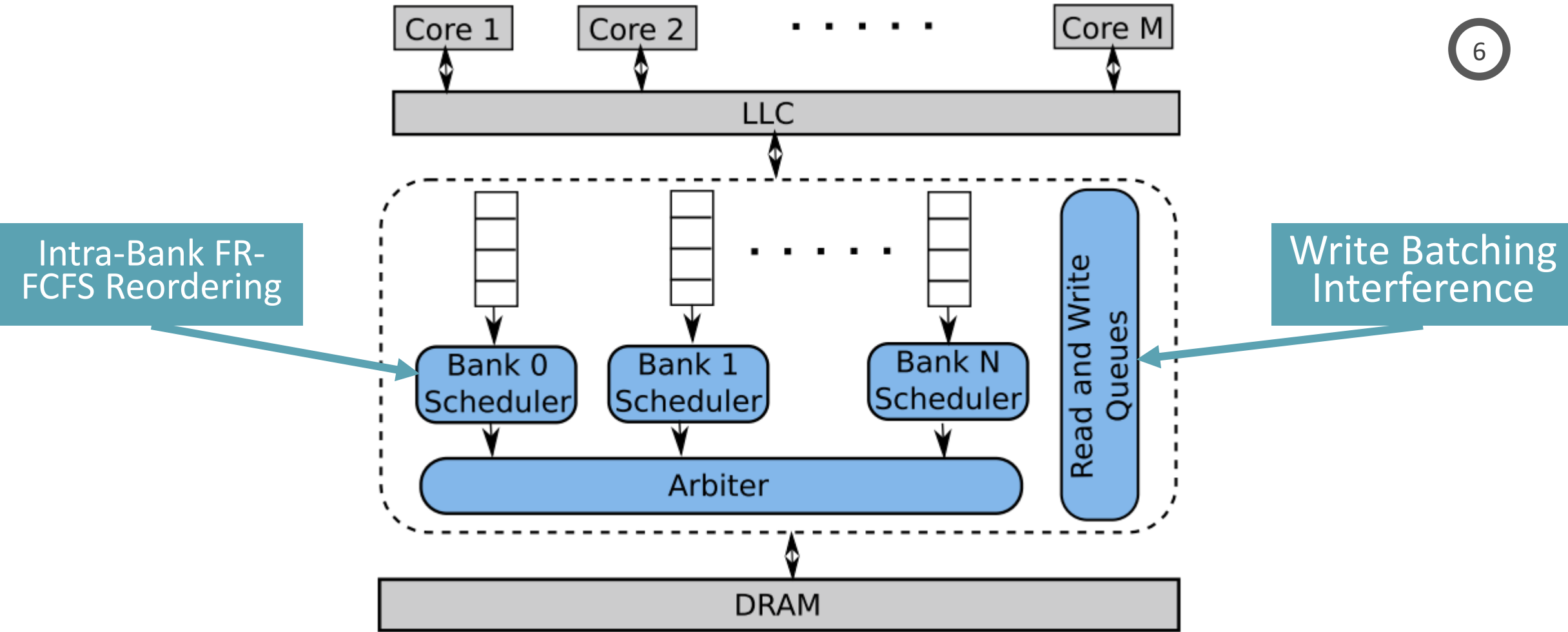
DRAM



# DRAM Interference

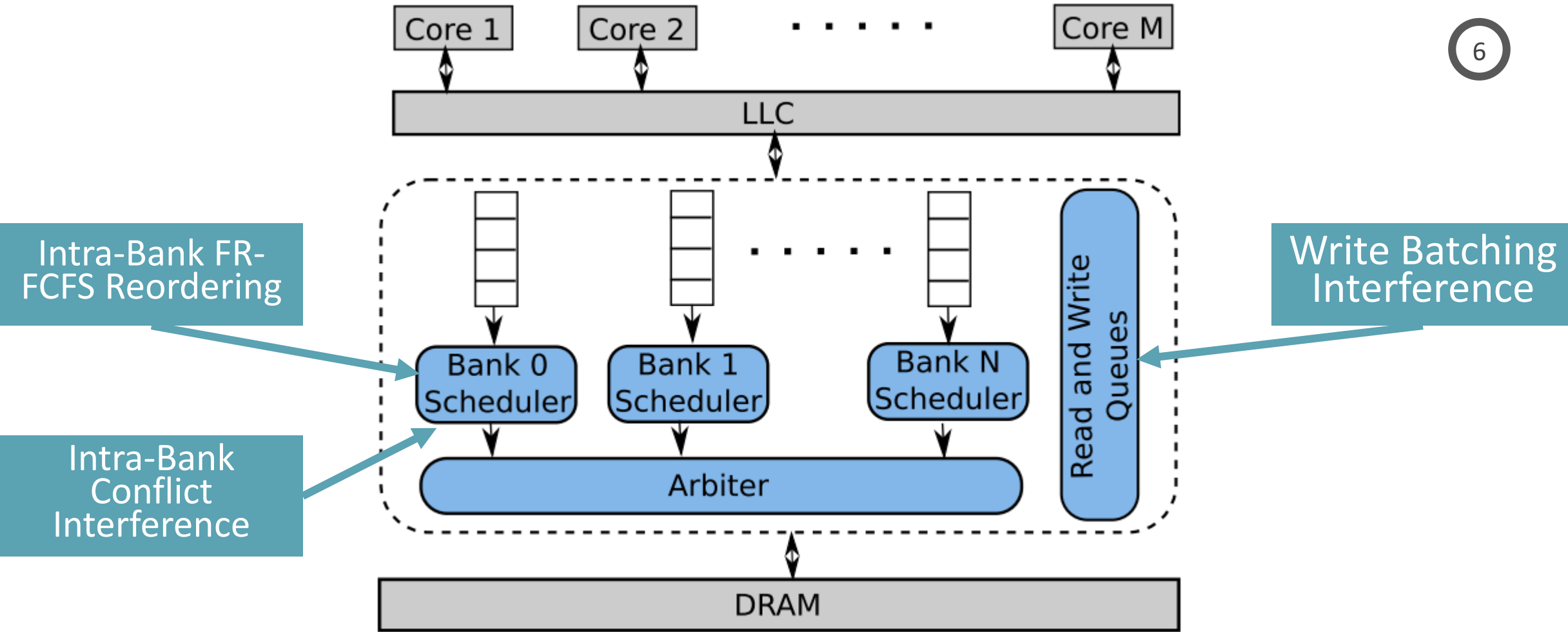
DRAM





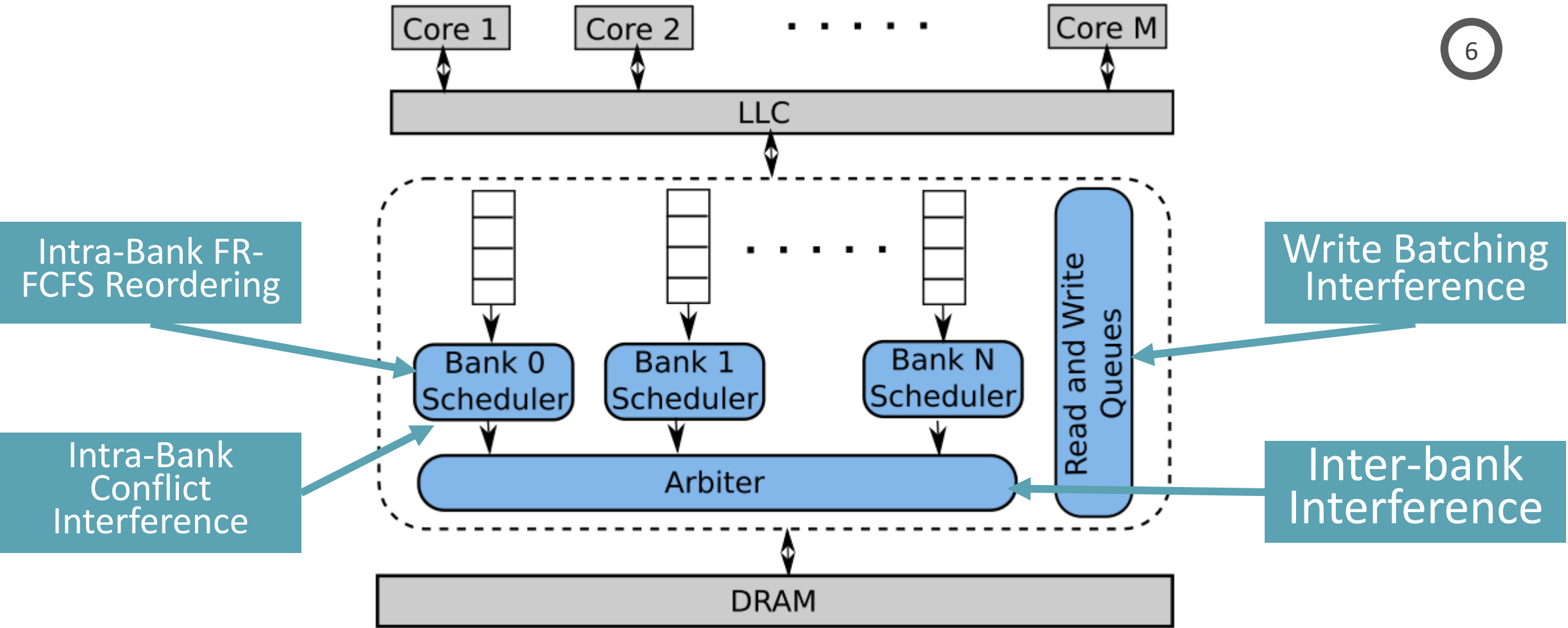
# DRAM Interference

DRAM



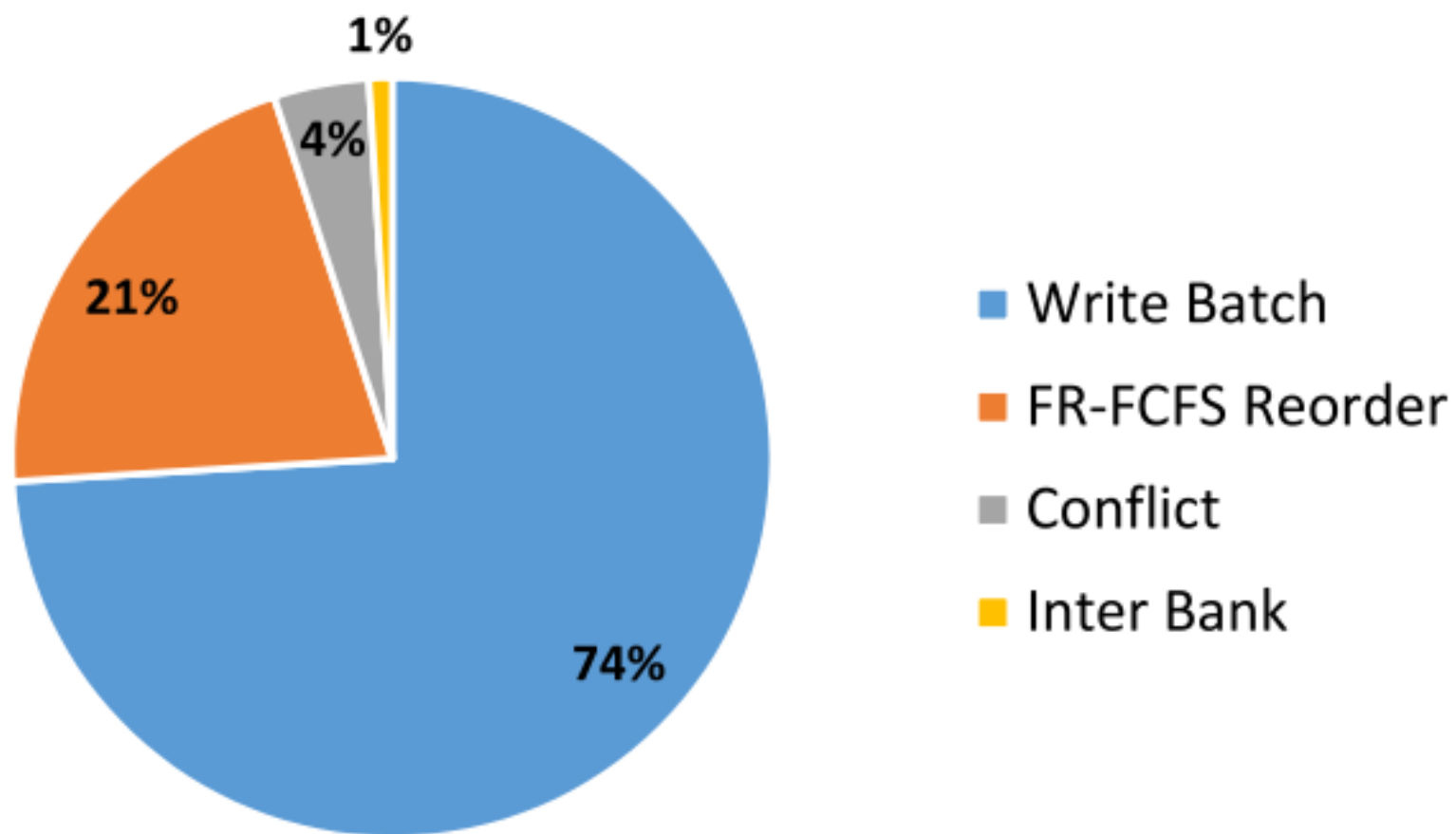
# DRAM Interference

DRAM



# DRAM Interference

DRAM



# DRAM Delay Components

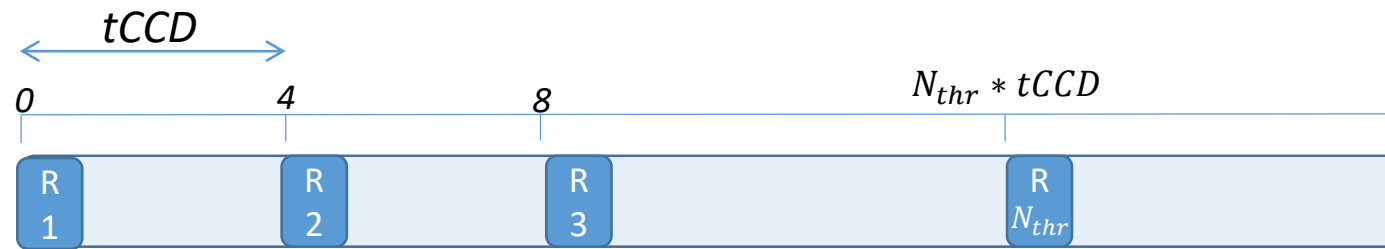
## DRAM

This paper:

- We study the effect of the largest two components in a mixed criticality system
- We derive both WCD bounds for critical cores and theoretical BW for non-critical cores

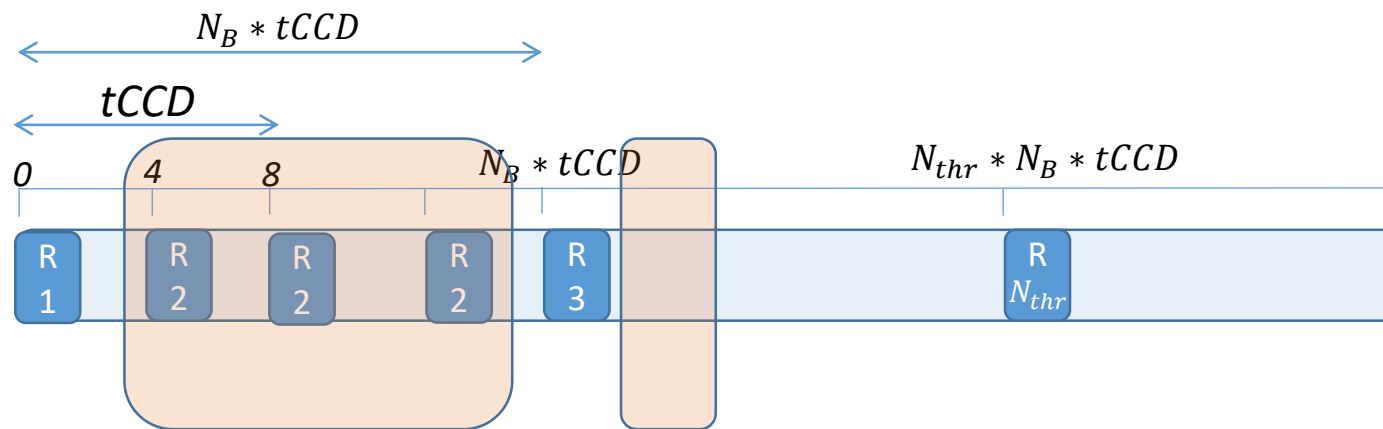
DRAM Delay Components

DRAM



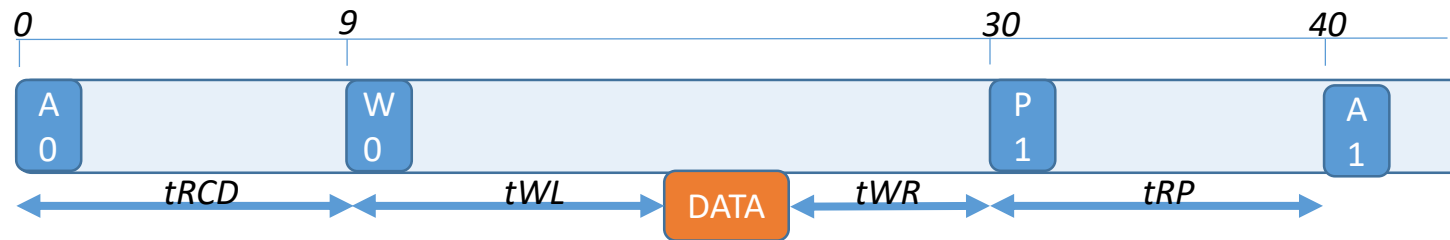
FR-FCFS WCD

DRAM



# FR-FCFS WCD

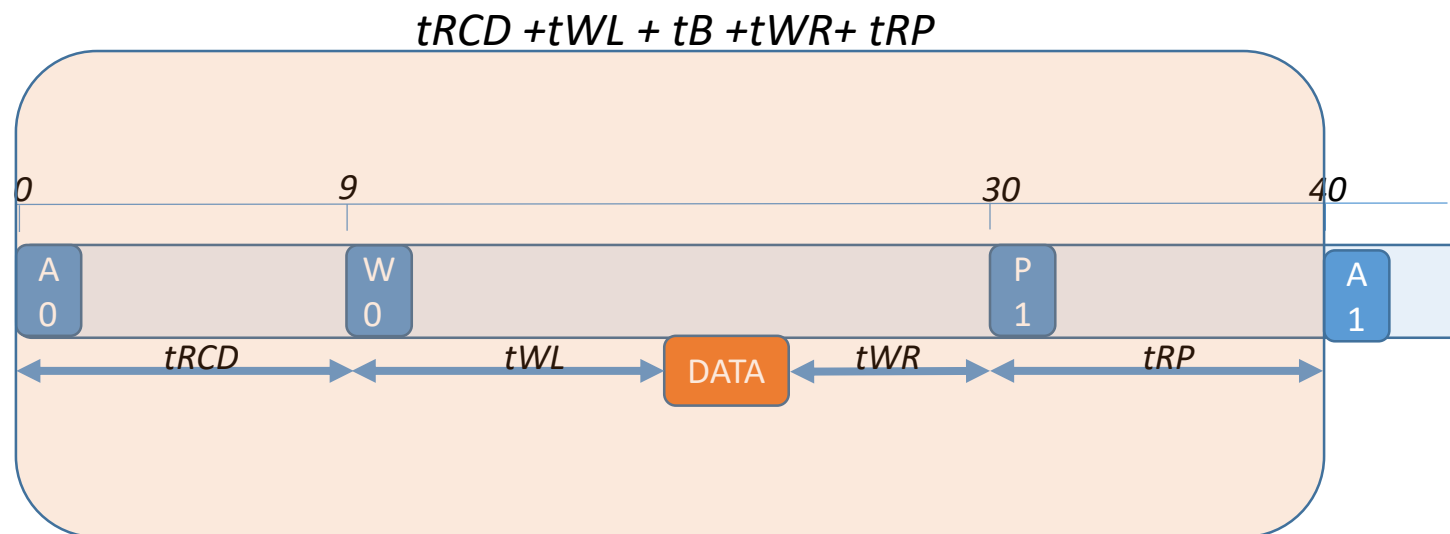
# DRAM



# Write Batching WCD

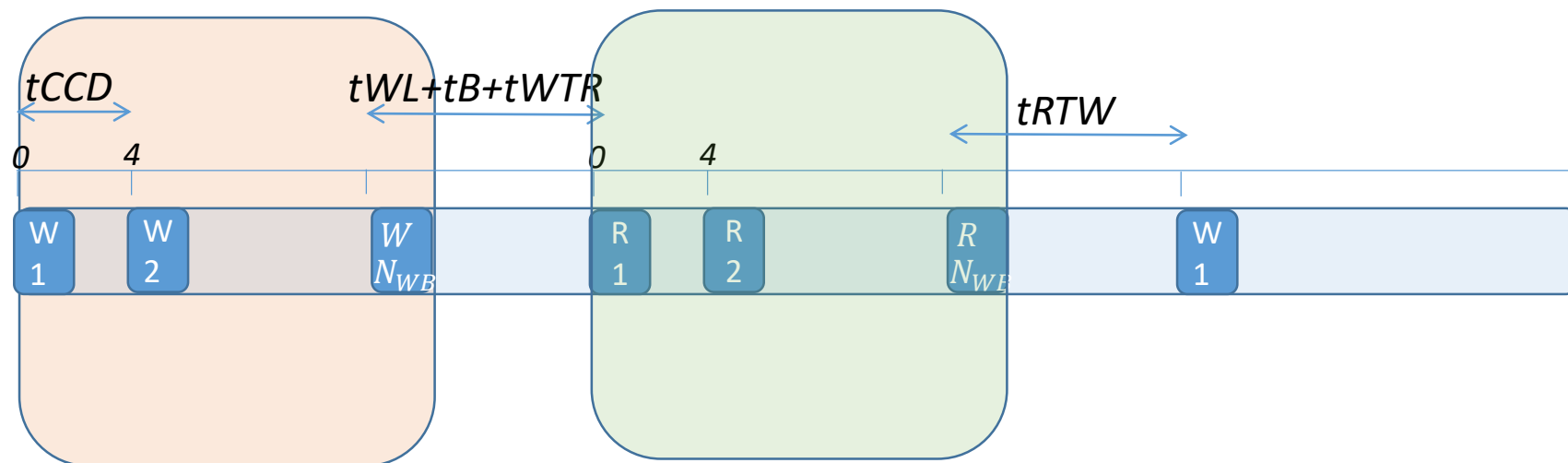
DRAM





# Write Batching WCD

DRAM



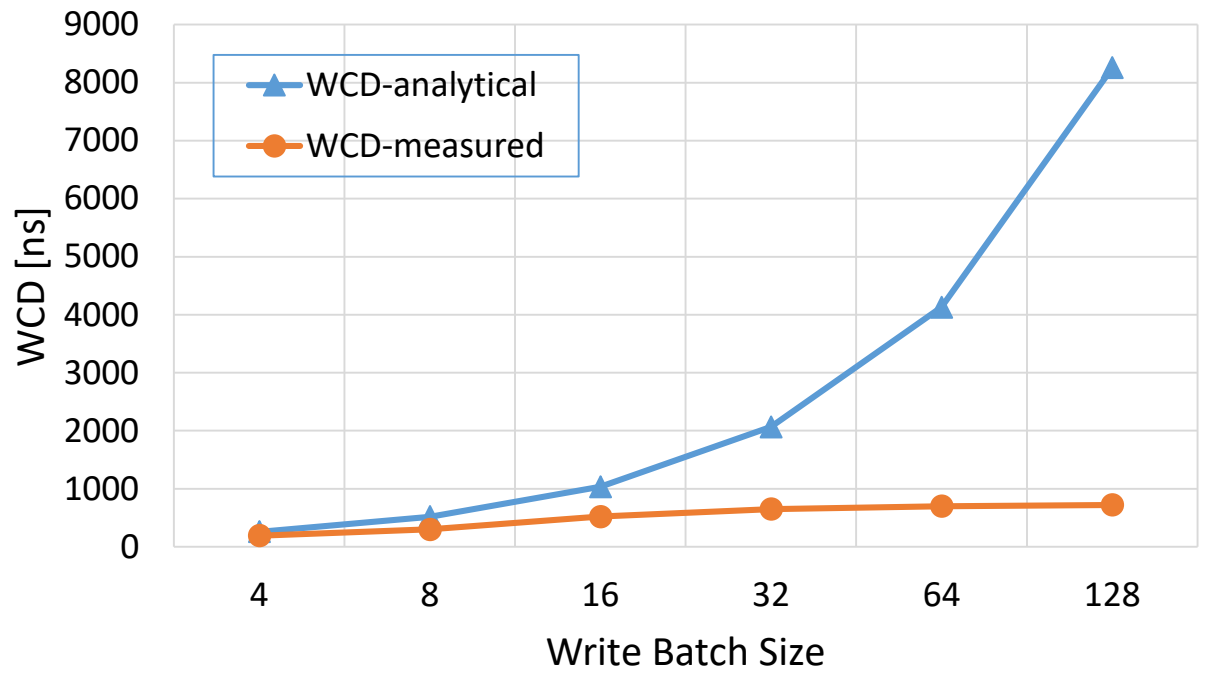
# Background

# DRAM

proc	2 Processors 1 OOO and 1 in-order pipeline a private 16KB L1 a shared 1MB L2 cache
DRAM	DDR3-1600
Benchmarks	Latency and BW benchmarks

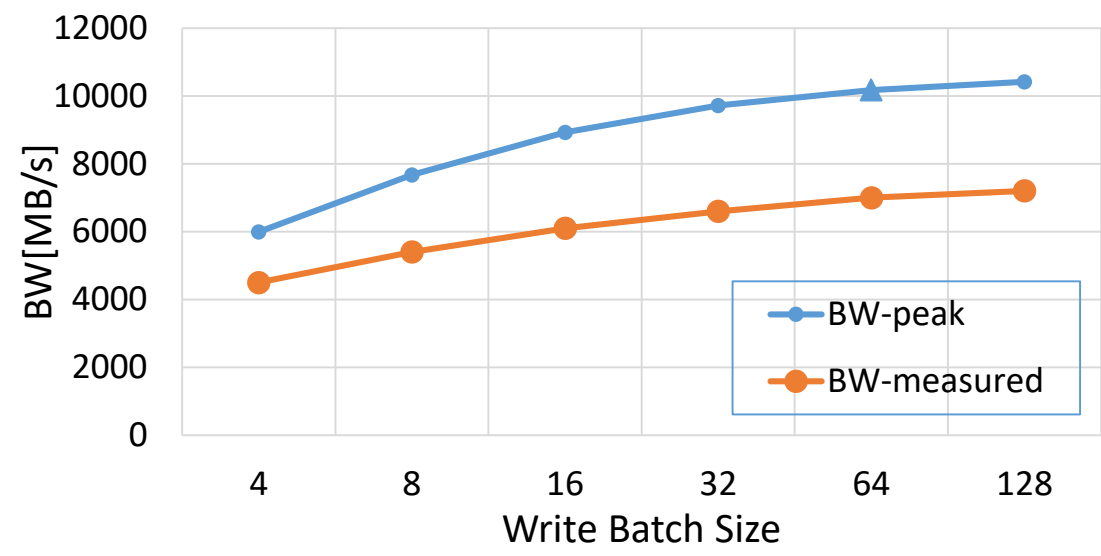
# Evaluation Setup

## RESULTS



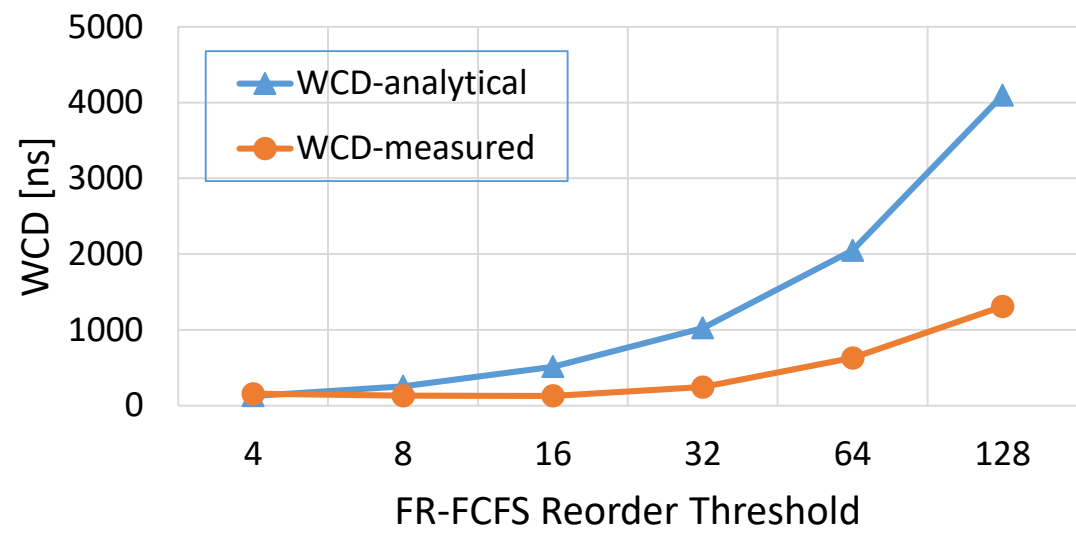
# Write Batching WCD

RESULTS



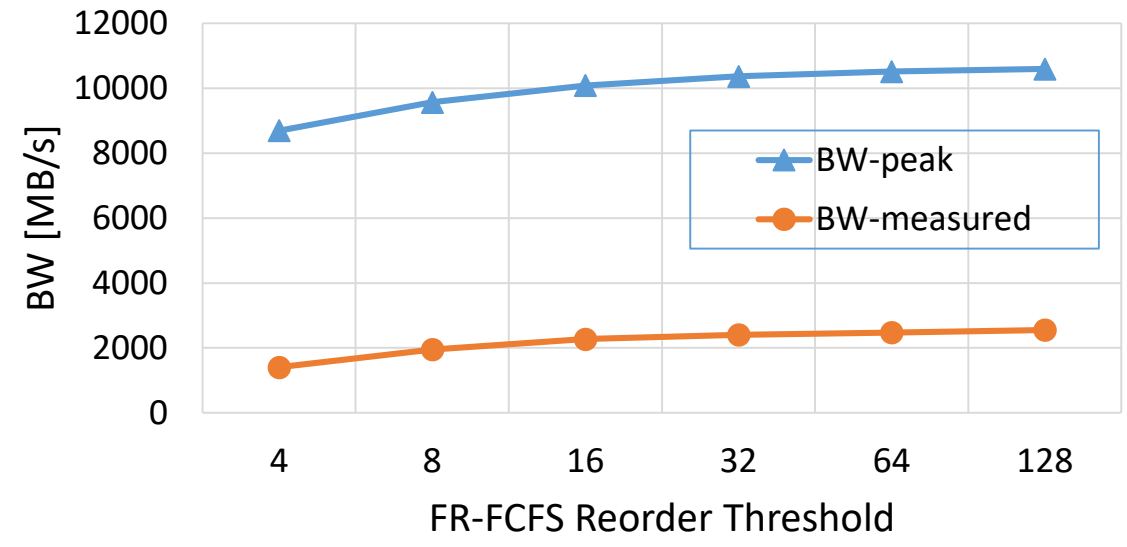
# Write Batching BW

RESULTS



# FR-FCFS WCD

## RESULTS



# FR-FCFS BW

# RESULTS

## Main Lessons:

1. We investigate the major sources of memory interference in COTS architectures upon deploying mixed criticality systems.
2. We identified two main contributors to this interference:
  - Write batching and FR-FCFS arbitration.
3. Although theoretically increasing the amount of reordering done helps increasing the system peak BW, after certain point there are diminishing returns in practice
4. such increase also significantly increases worst-case delays
5. Reaching a compromise point between BW and WCD to meet all requirements of MCS is the potential direction to explore. Such point will be application-dependent.
6. We provide analytical equations as well as experimental evaluation to help designers deciding this point

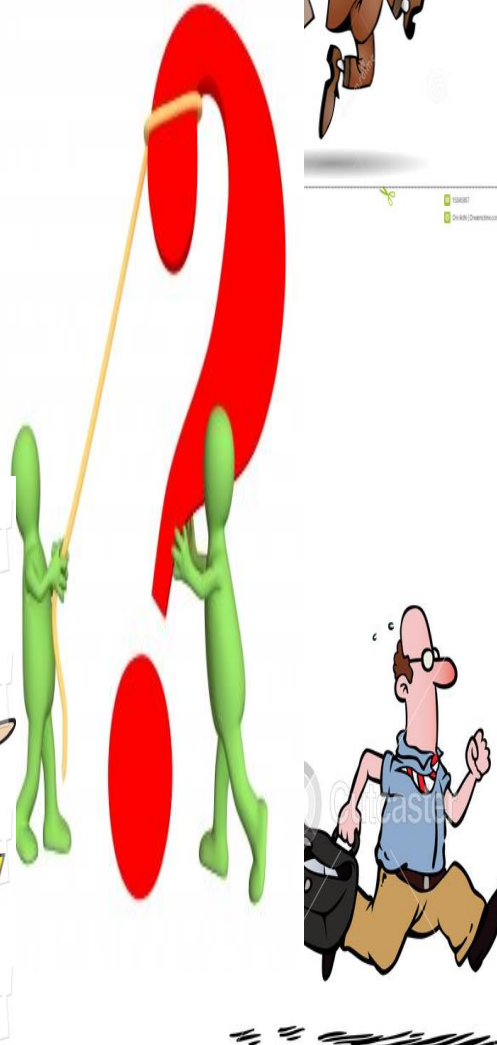




www.clipartof.com · 5817



Copyright © Ron Leiden · http://MyTuxClipart.com/260



cutcasten.com - 90356208

## Main Lessons:

1. We investigate the major sources of memory interference in COTS architectures upon deploying mixed criticality systems.
2. We identified two main contributors to this interference:
  - Write batching and FR-FCFS arbitration.
3. Although theoretically increasing the amount of reordering done helps increasing the system peak BW, after certain point there are diminishing returns in practice
4. such increase also significantly increases worst-case delays
5. Reaching a compromise point between BW and WCD to meet all requirements of MCS is the potential direction to explore. Such point will be application-dependent.
6. We provide analytical equations as well as experimental evaluation to help designers deciding this point