



INTERFERENCE FROM GPU SYSTEM SERVICE REQUESTS

ARKAPRAVA BASU, JOSEPH L. GREATHOUSE, GURU VENKATARAMANI, JÁN VESELÝ
AMD RESEARCH, ADVANCED MICRO DEVICES, INC.

MODERN SYSTEMS ARE POWERED BY HETEROGENEITY

6th Gen. AMD A-Series Processor “Carrizo”



MODERN SYSTEMS ARE POWERED BY HETEROGENEITY

6th Gen. AMD A-Series Processor “Carrizo”



CPUs

MODERN SYSTEMS ARE POWERED BY HETEROGENEITY

6th Gen. AMD A-Series Processor “Carrizo”

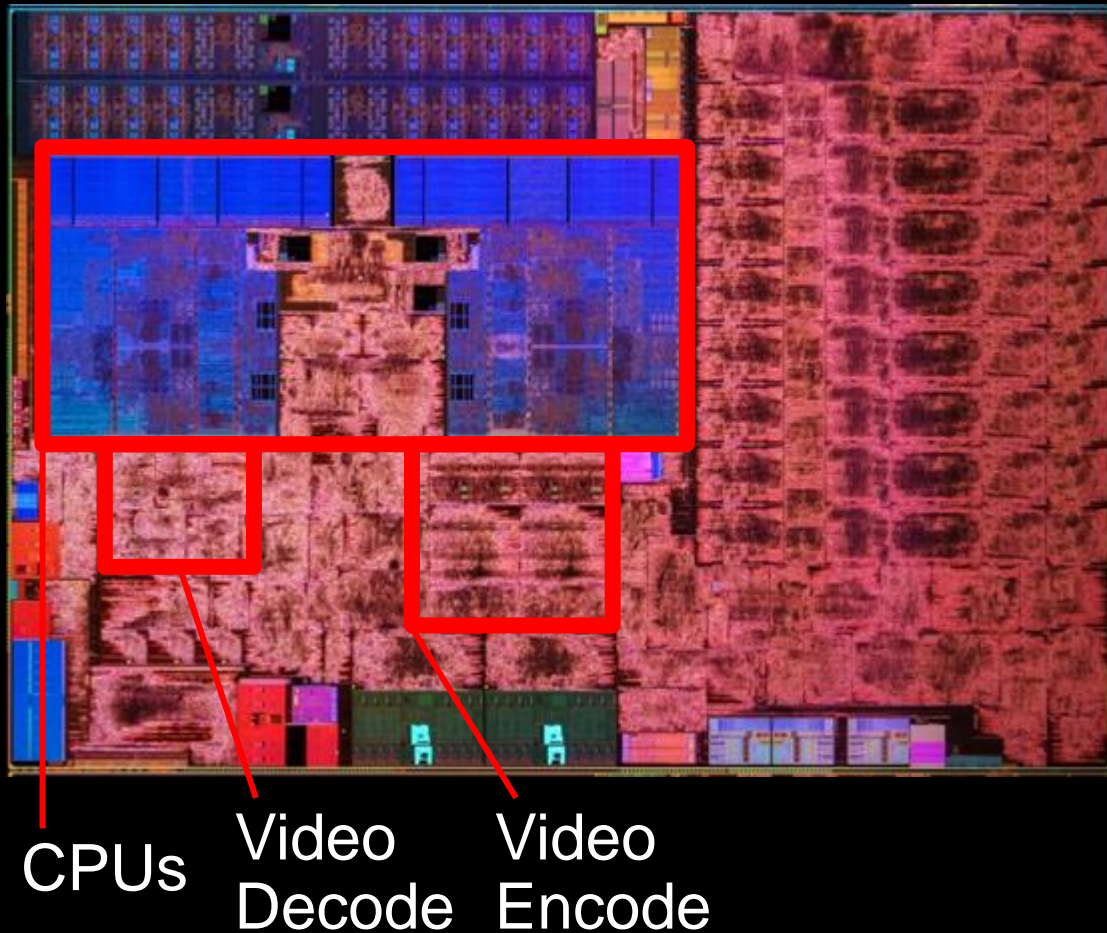


CPU

Video
Decode

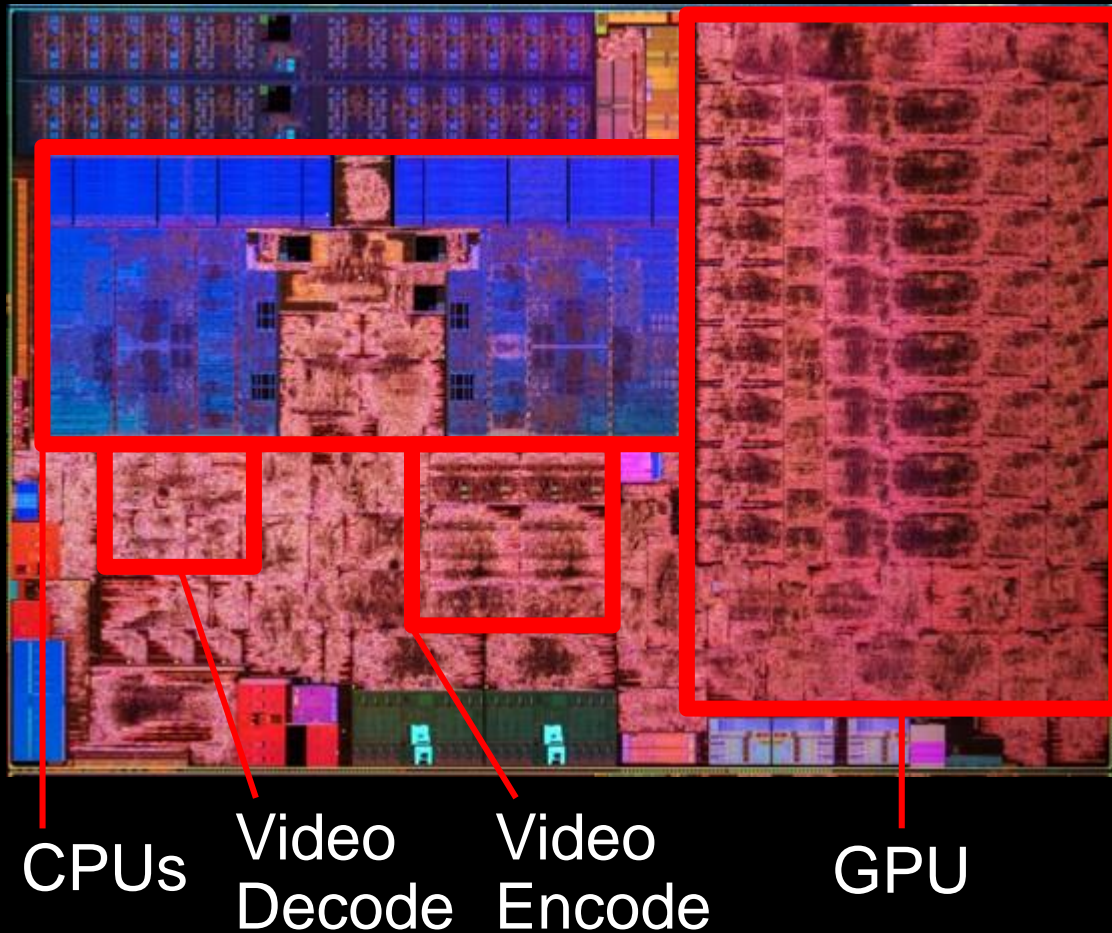
MODERN SYSTEMS ARE POWERED BY HETEROGENEITY

6th Gen. AMD A-Series Processor “Carrizo”



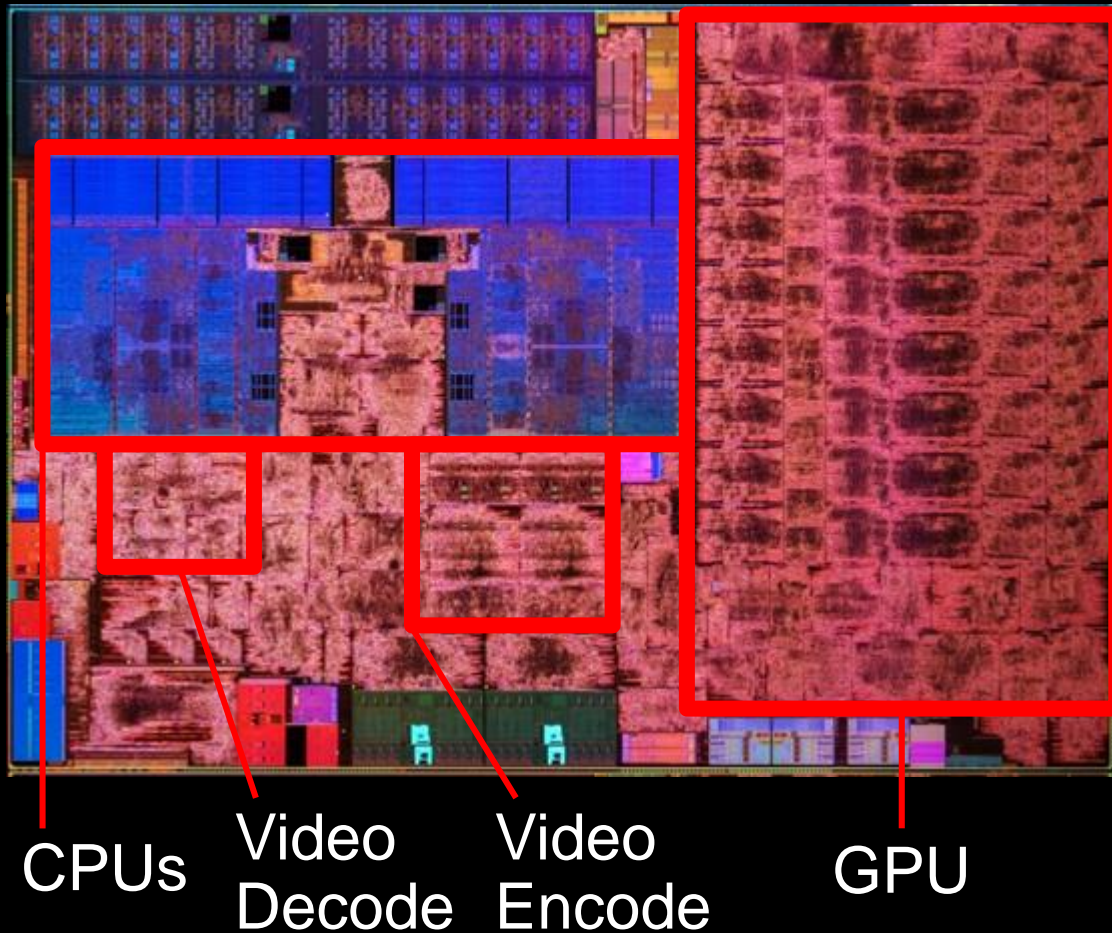
MODERN SYSTEMS ARE POWERED BY HETEROGENEITY

6th Gen. AMD A-Series Processor “Carrizo”



MODERN SYSTEMS ARE POWERED BY HETEROGENEITY

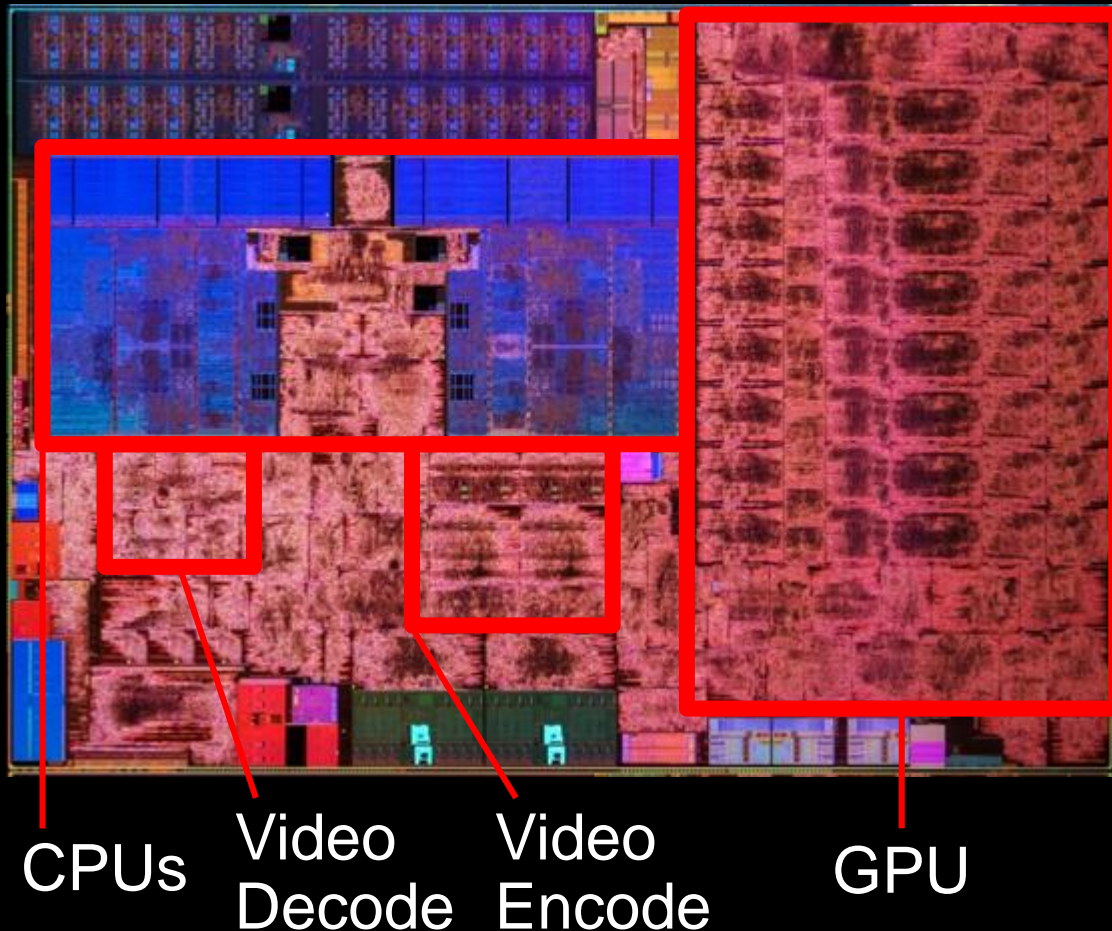
6th Gen. AMD A-Series Processor “Carrizo”



Accelerators from Industry and Academia

MODERN SYSTEMS ARE POWERED BY HETEROGENEITY

6th Gen. AMD A-Series Processor “Carrizo”



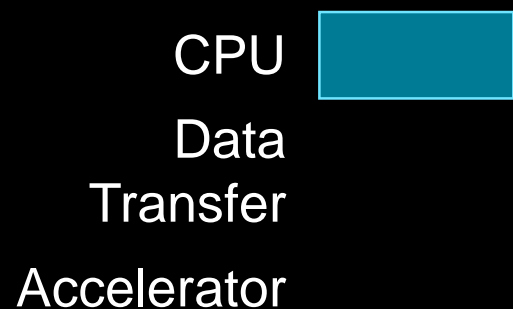
Accelerators from Industry and Academia

- Machine Learning
- Databases
- Computer Vision
- Regular Expressions
- Physics
- Graph Analytics
- Finite State Machines
- Genome Sequencing
- Reconfigurable (e.g., FPGA)

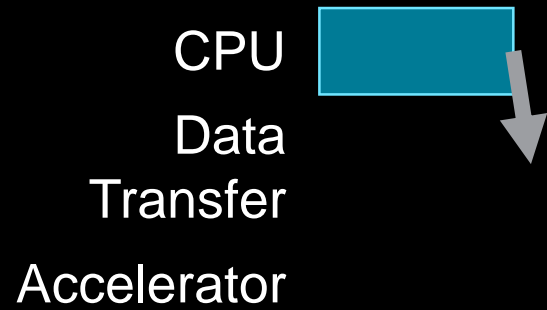
TRADITIONAL HETEROGENEOUS COMPUTING DRIVEN BY CPUS

CPU
Data
Transfer
Accelerator

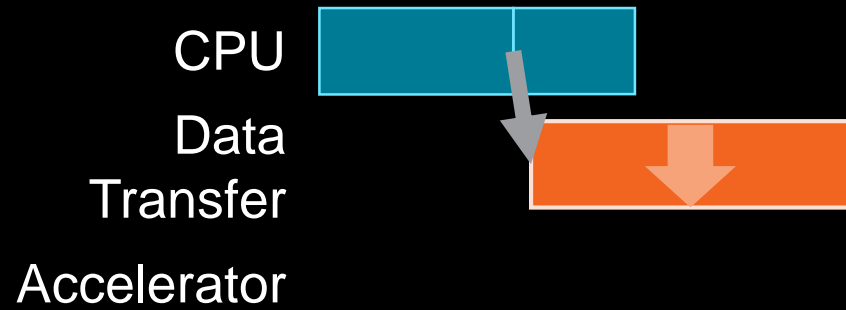
TRADITIONAL HETEROGENEOUS COMPUTING DRIVEN BY CPUS



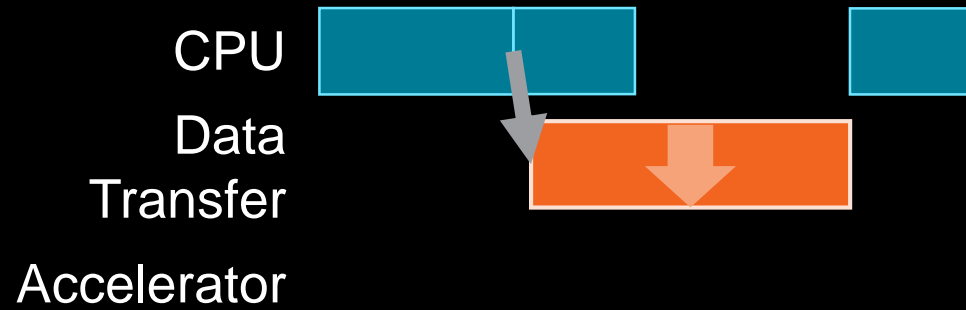
TRADITIONAL HETEROGENEOUS COMPUTING DRIVEN BY CPUS



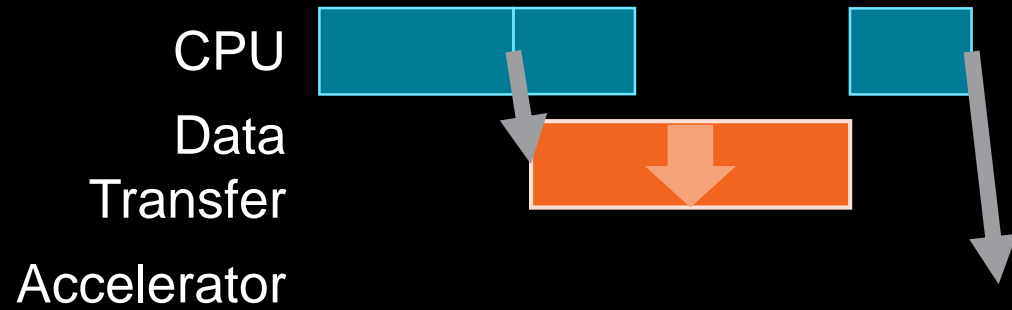
TRADITIONAL HETEROGENEOUS COMPUTING DRIVEN BY CPUS



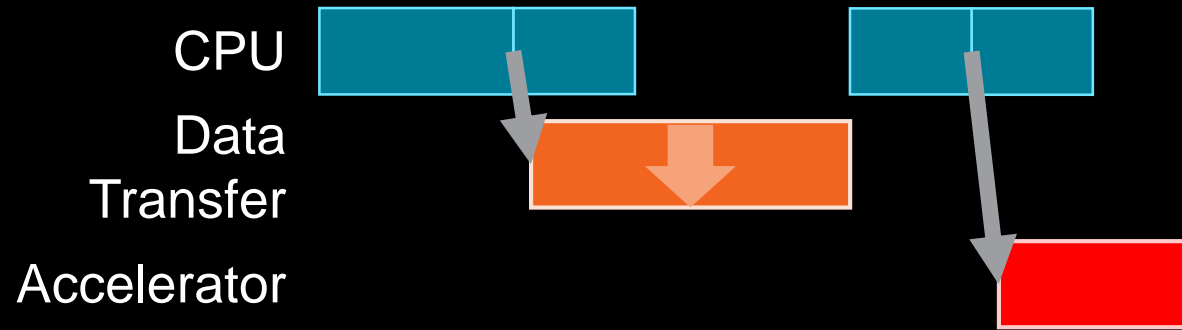
TRADITIONAL HETEROGENEOUS COMPUTING DRIVEN BY CPUS



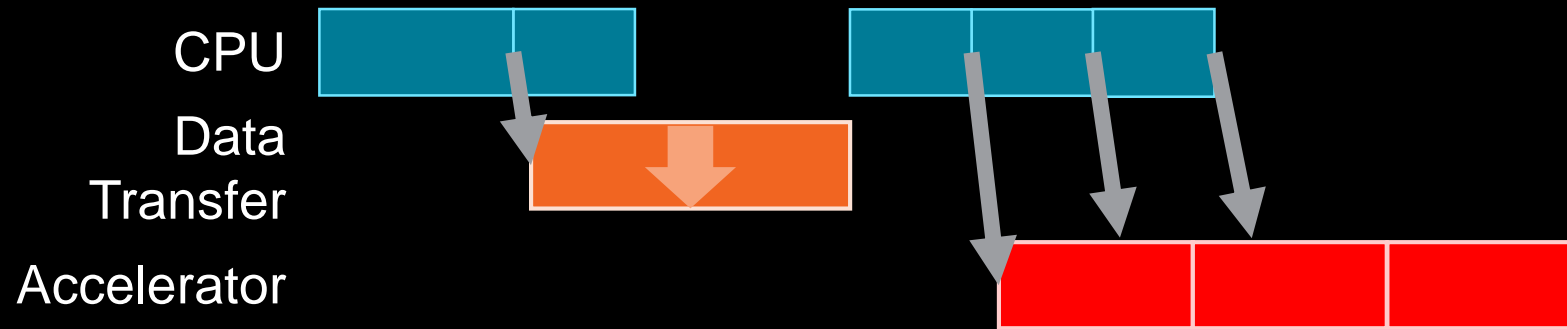
TRADITIONAL HETEROGENEOUS COMPUTING DRIVEN BY CPUS



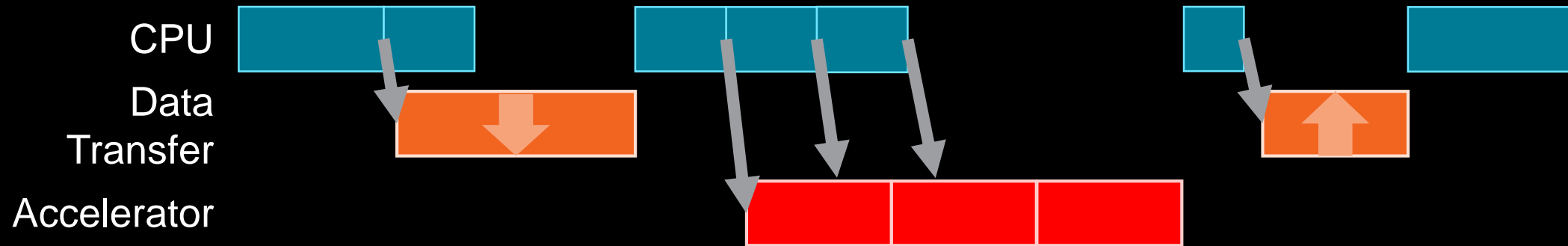
TRADITIONAL HETEROGENEOUS COMPUTING DRIVEN BY CPUS



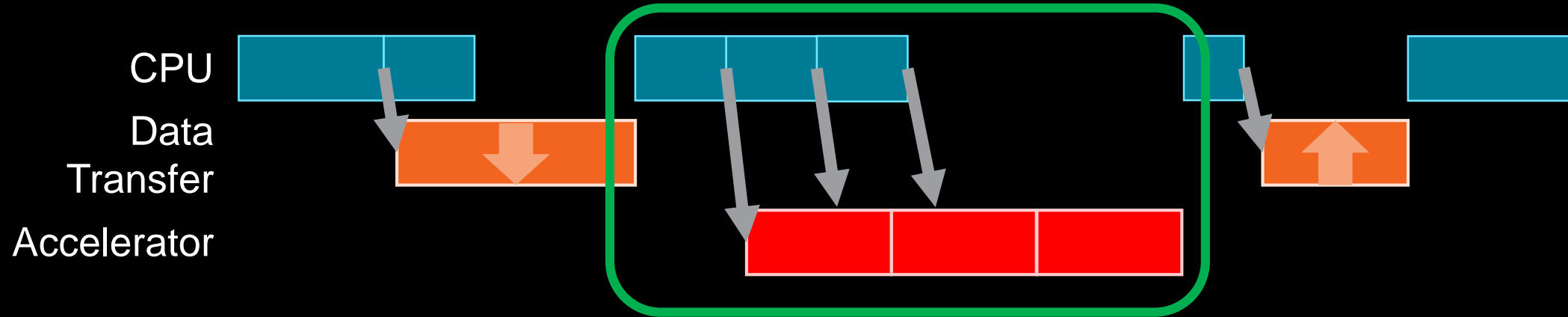
TRADITIONAL HETEROGENEOUS COMPUTING DRIVEN BY CPUS



TRADITIONAL HETEROGENEOUS COMPUTING DRIVEN BY CPUS

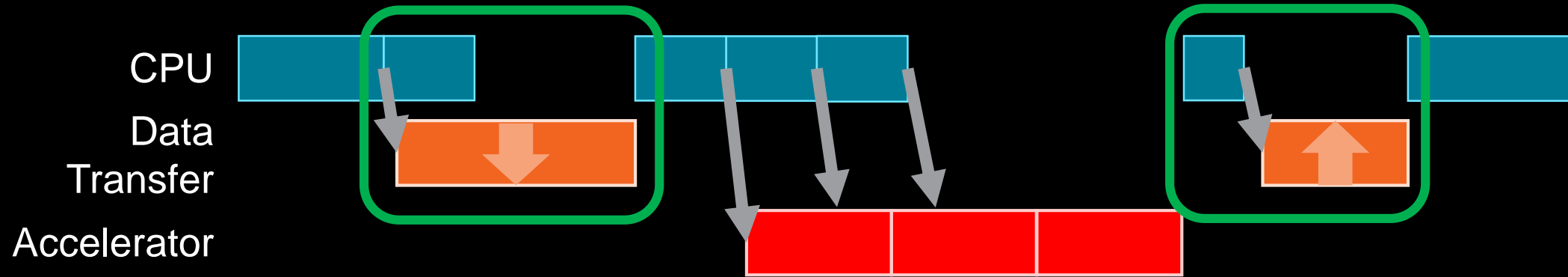


TRADITIONAL HETEROGENEOUS COMPUTING DRIVEN BY CPUS



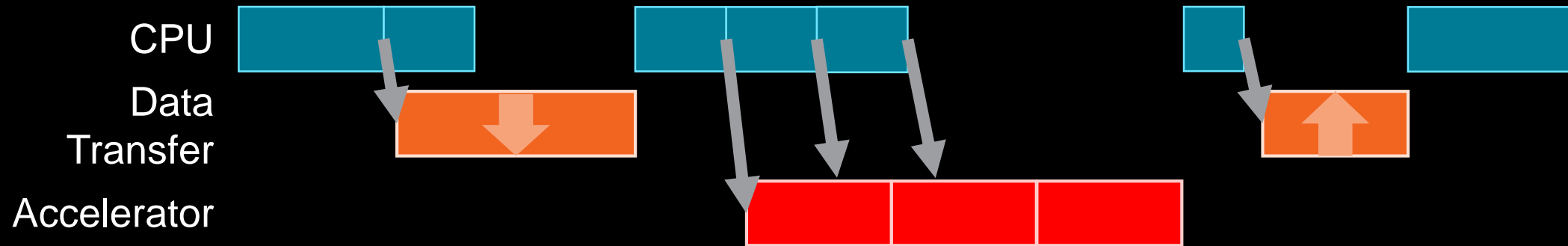
- Work launched by the CPU to accelerators

TRADITIONAL HETEROGENEOUS COMPUTING DRIVEN BY CPUS



- Work launched by the CPU to accelerators
- Data marshalled by the CPU

TRADITIONAL HETEROGENEOUS COMPUTING DRIVEN BY CPUS

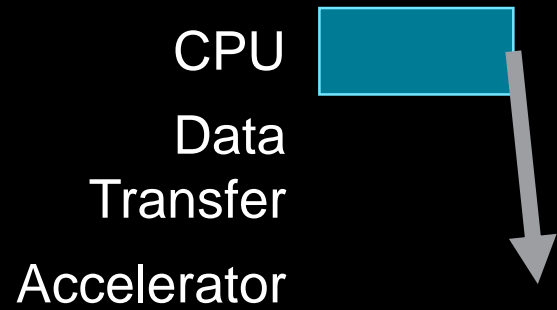


- Work launched by the CPU to accelerators
- Data marshalled by the CPU
- Communication and computation are coarse-grained

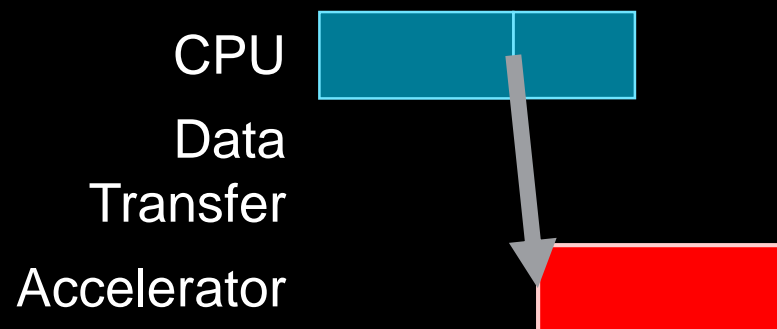
MODERN ACCELERATORS ARE EQUAL PARTNERS

CPU
Data
Transfer
Accelerator

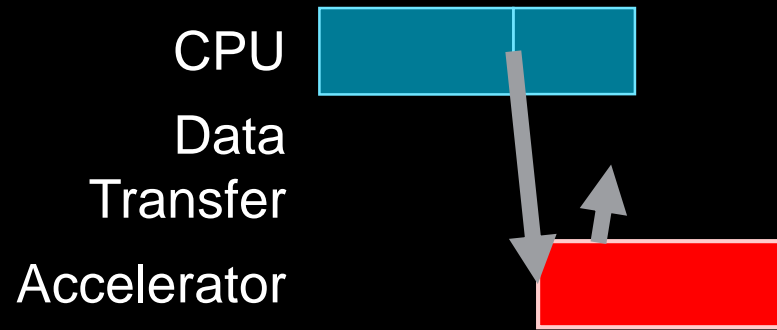
MODERN ACCELERATORS ARE EQUAL PARTNERS



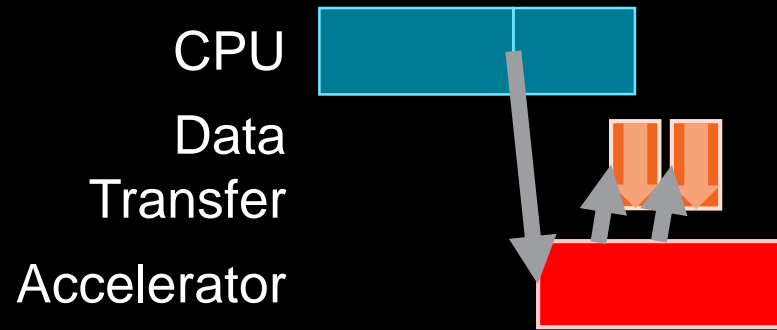
MODERN ACCELERATORS ARE EQUAL PARTNERS



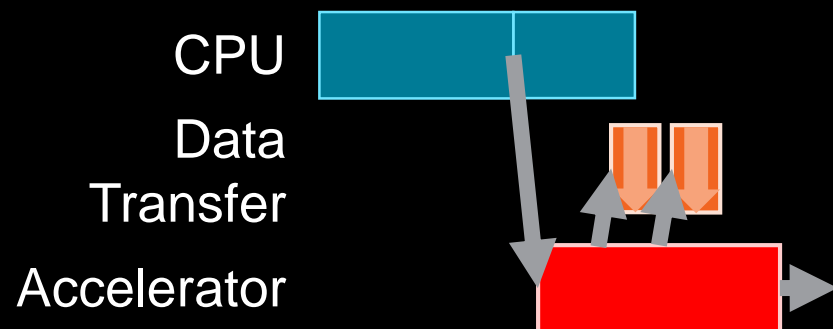
MODERN ACCELERATORS ARE EQUAL PARTNERS



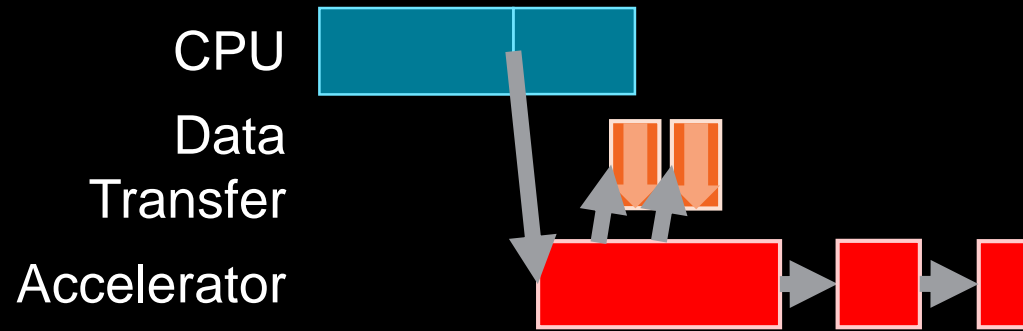
MODERN ACCELERATORS ARE EQUAL PARTNERS



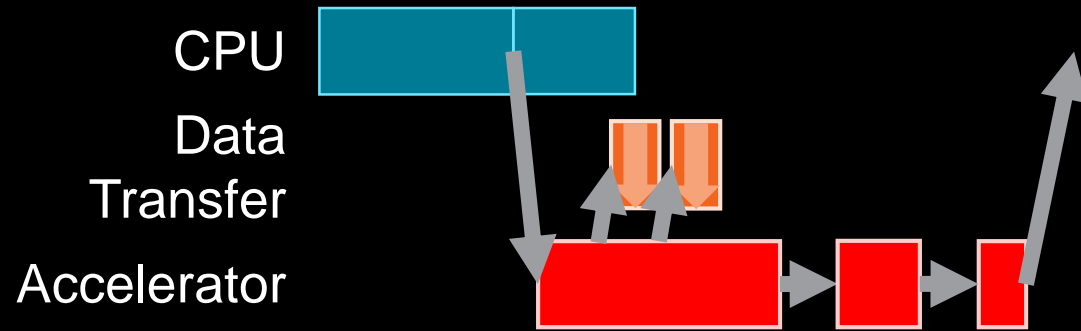
MODERN ACCELERATORS ARE EQUAL PARTNERS



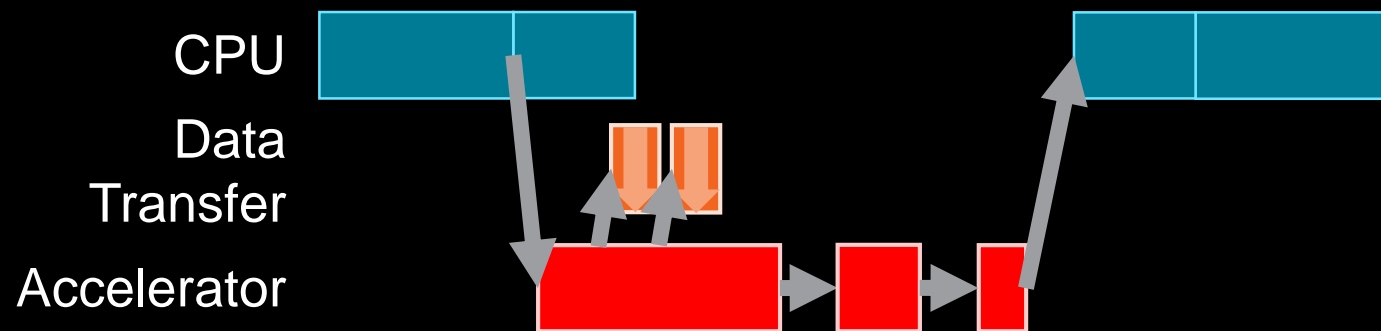
MODERN ACCELERATORS ARE EQUAL PARTNERS



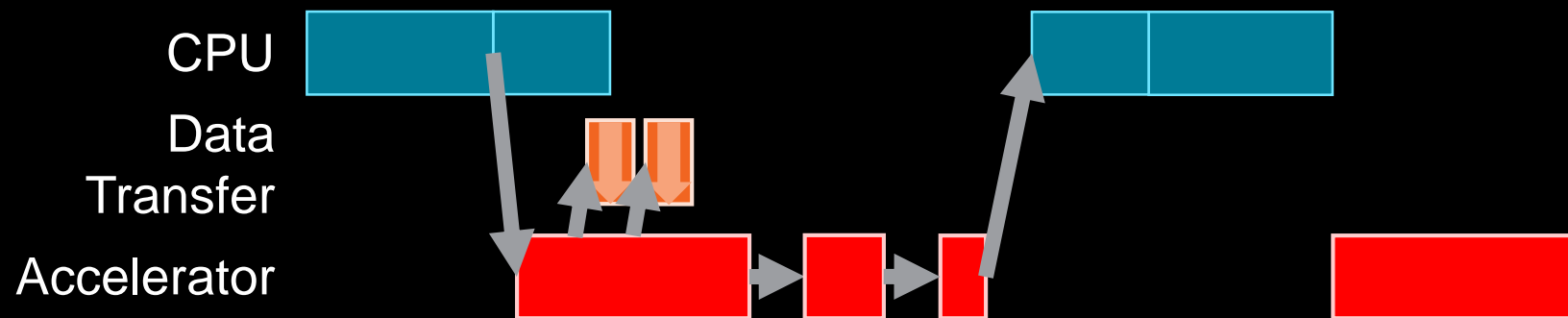
MODERN ACCELERATORS ARE EQUAL PARTNERS



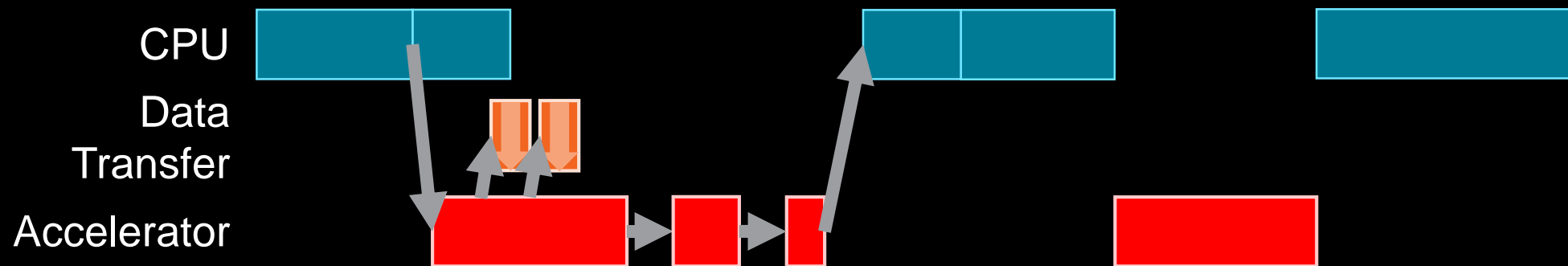
MODERN ACCELERATORS ARE EQUAL PARTNERS



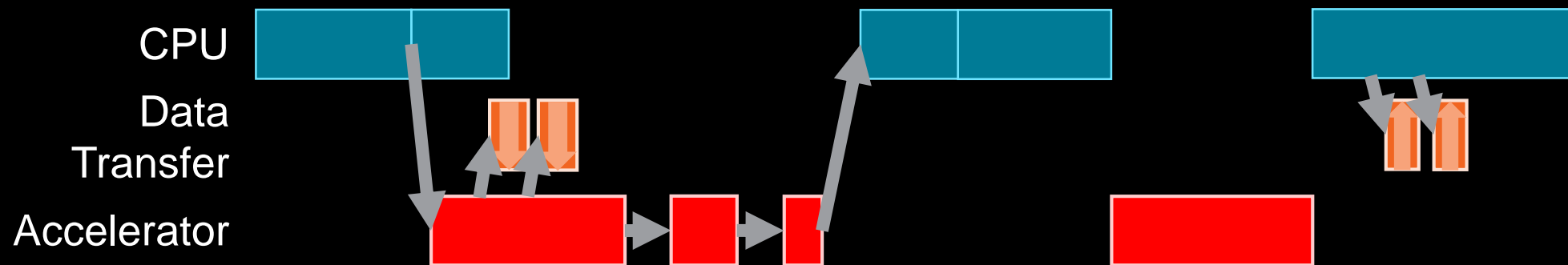
MODERN ACCELERATORS ARE EQUAL PARTNERS



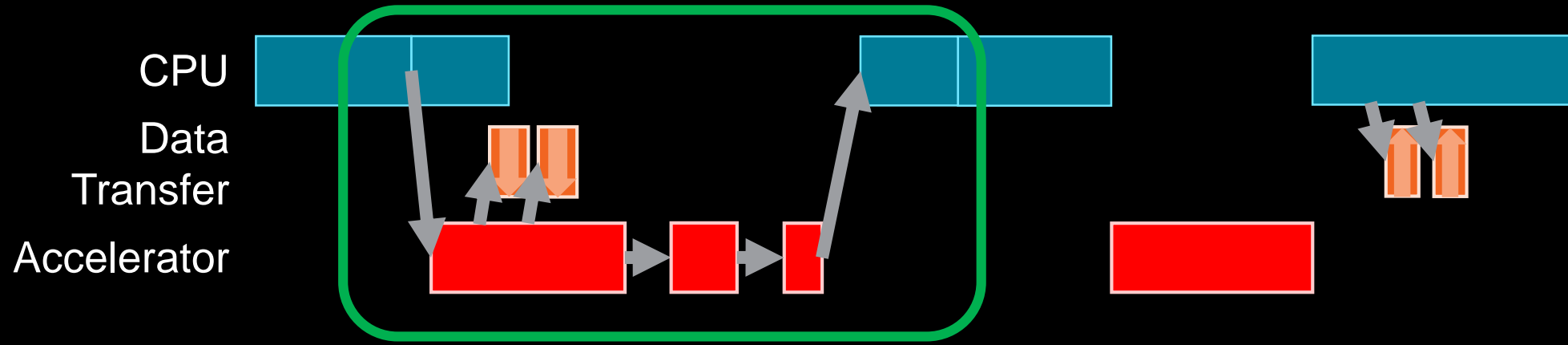
MODERN ACCELERATORS ARE EQUAL PARTNERS



MODERN ACCELERATORS ARE EQUAL PARTNERS

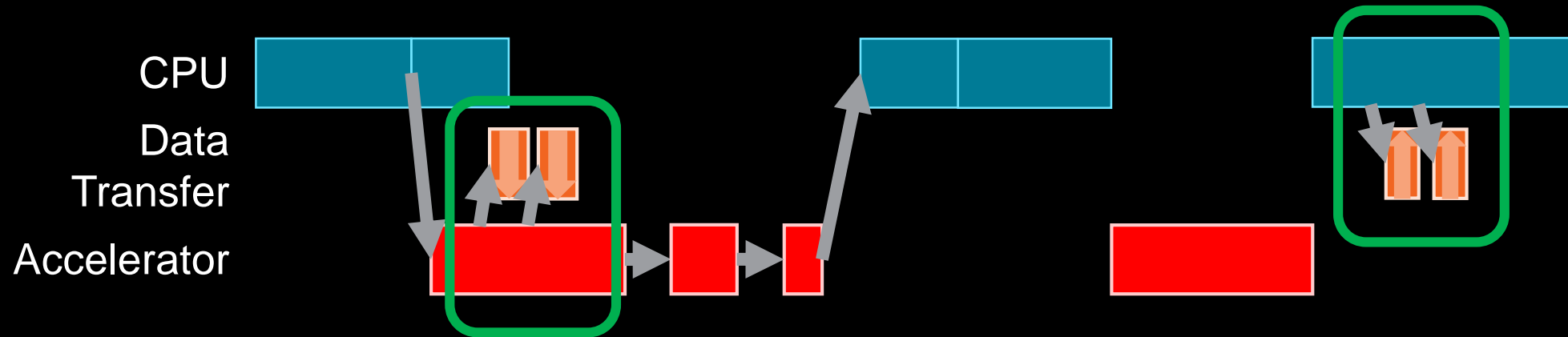


MODERN ACCELERATORS ARE EQUAL PARTNERS



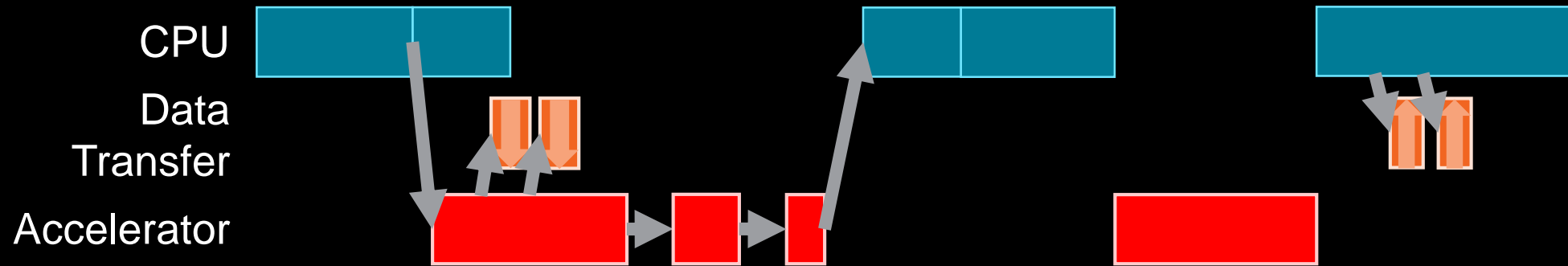
- Accelerators and CPUs can launch work to one another (and themselves)

MODERN ACCELERATORS ARE EQUAL PARTNERS



- Accelerators and CPUs can launch work to one another (and themselves)
- Data transfer implicit based on usage

MODERN ACCELERATORS ARE EQUAL PARTNERS



- Accelerators and CPUs can launch work to one another (and themselves)
- Data transfer implicit based on usage
- Communication and computation are fine-grained

ACCELERATORS CAN NOW REQUEST OS SERVICES

ACCELERATORS CAN NOW REQUEST OS SERVICES

- GPU-to-CPU Callbacks (Owens et al., UCHPC 2010)

ACCELERATORS CAN NOW REQUEST OS SERVICES

- GPU-to-CPU Callbacks (Owens et al., UCHPC 2010)
- Page Faults / Demand Paging (Veselý et al., ISPASS 2016)

ACCELERATORS CAN NOW REQUEST OS SERVICES

- GPU-to-CPU Callbacks (Owens et al., UCHPC 2010)
- Page Faults / Demand Paging (Veselý et al., ISPASS 2016)
- GPU-Initiated Network Requests (GPUnet, USENIX 2014)

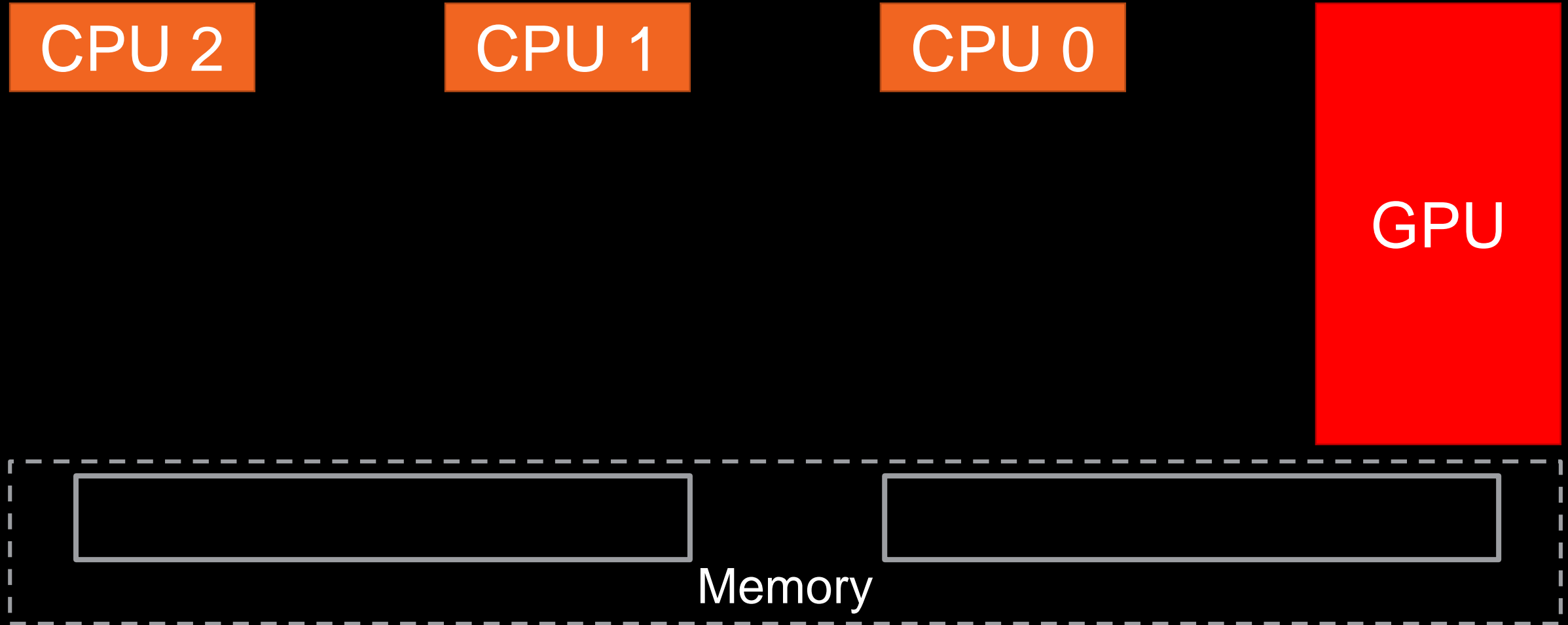
ACCELERATORS CAN NOW REQUEST OS SERVICES

- GPU-to-CPU Callbacks (Owens et al., UCHPC 2010)
- Page Faults / Demand Paging (Veselý et al., ISPASS 2016)
- GPU-Initiated Network Requests (GPUnet, USENIX 2014)
- GPU-Initiated File System Requests (GPUfs, ASPLOS 2013)

ACCELERATORS CAN NOW REQUEST OS SERVICES

- GPU-to-CPU Callbacks (Owens et al., UCHPC 2010)
- Page Faults / Demand Paging (Veselý et al., ISPASS 2016)
- GPU-Initiated Network Requests (GPUnet, USENIX 2014)
- GPU-Initiated File System Requests (GPUfs, ASPLOS 2013)
- “Generic” System Calls (Genesys, ISCA 2018)
 - ioctl() for other devices
 - Memory management (e.g., sbrk(), mmap())
 - Signals

HOW A GPU SYSTEM SERVICE REQUEST IS HANDLED



HOW A GPU SYSTEM SERVICE REQUEST IS HANDLED

CPU 2

CPU 1

CPU 0

GPU

Step 1: Set up request arguments in memory



HOW A GPU SYSTEM SERVICE REQUEST IS HANDLED

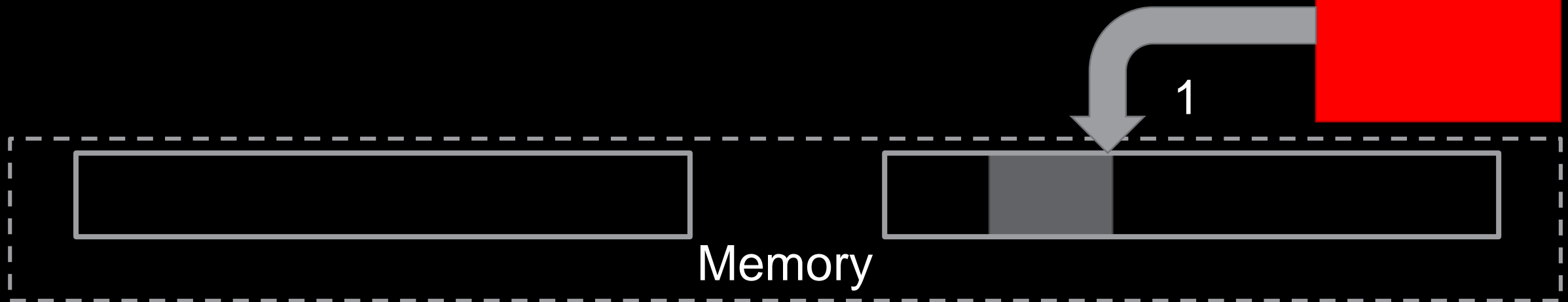
CPU 2

CPU 1

CPU 0

GPU

Step 1: Set up request arguments in memory



HOW A GPU SYSTEM SERVICE REQUEST IS HANDLED

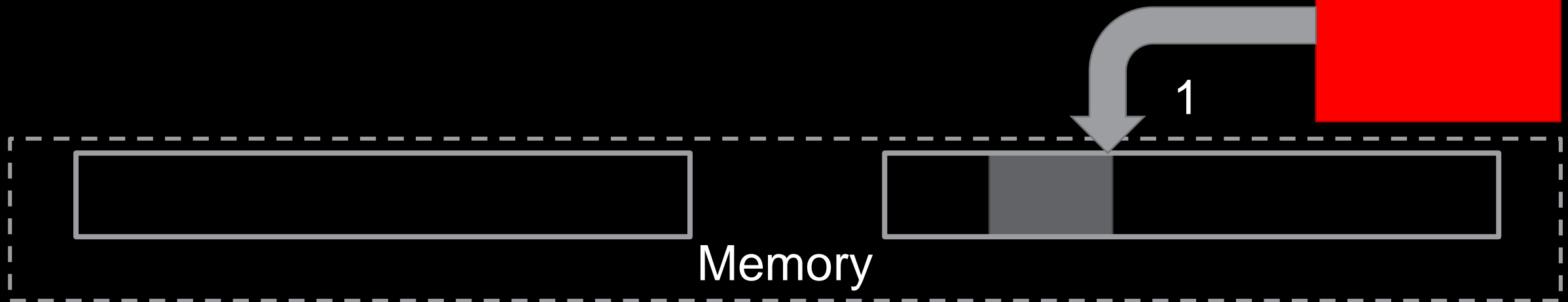
CPU 2

CPU 1

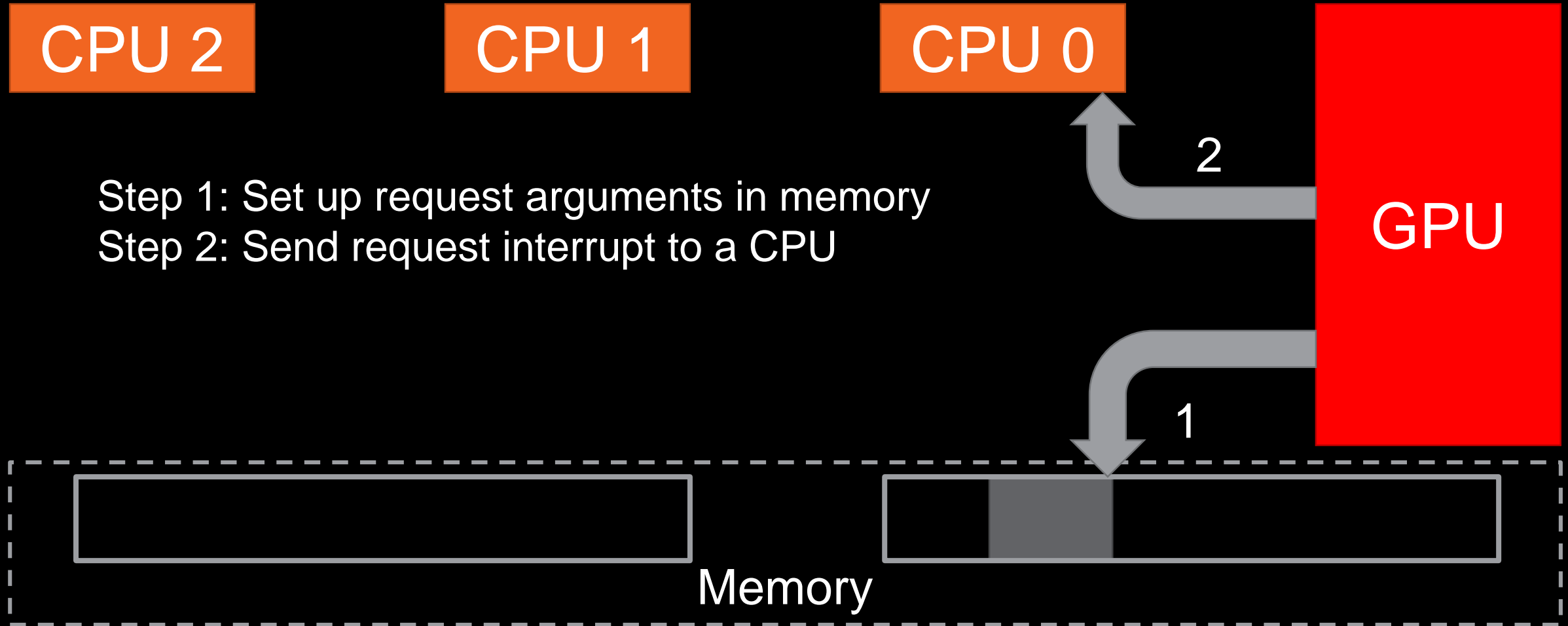
CPU 0

GPU

Step 1: Set up request arguments in memory
Step 2: Send request interrupt to a CPU



HOW A GPU SYSTEM SERVICE REQUEST IS HANDLED



HOW A GPU SYSTEM SERVICE REQUEST IS HANDLED

CPU 2

CPU 1

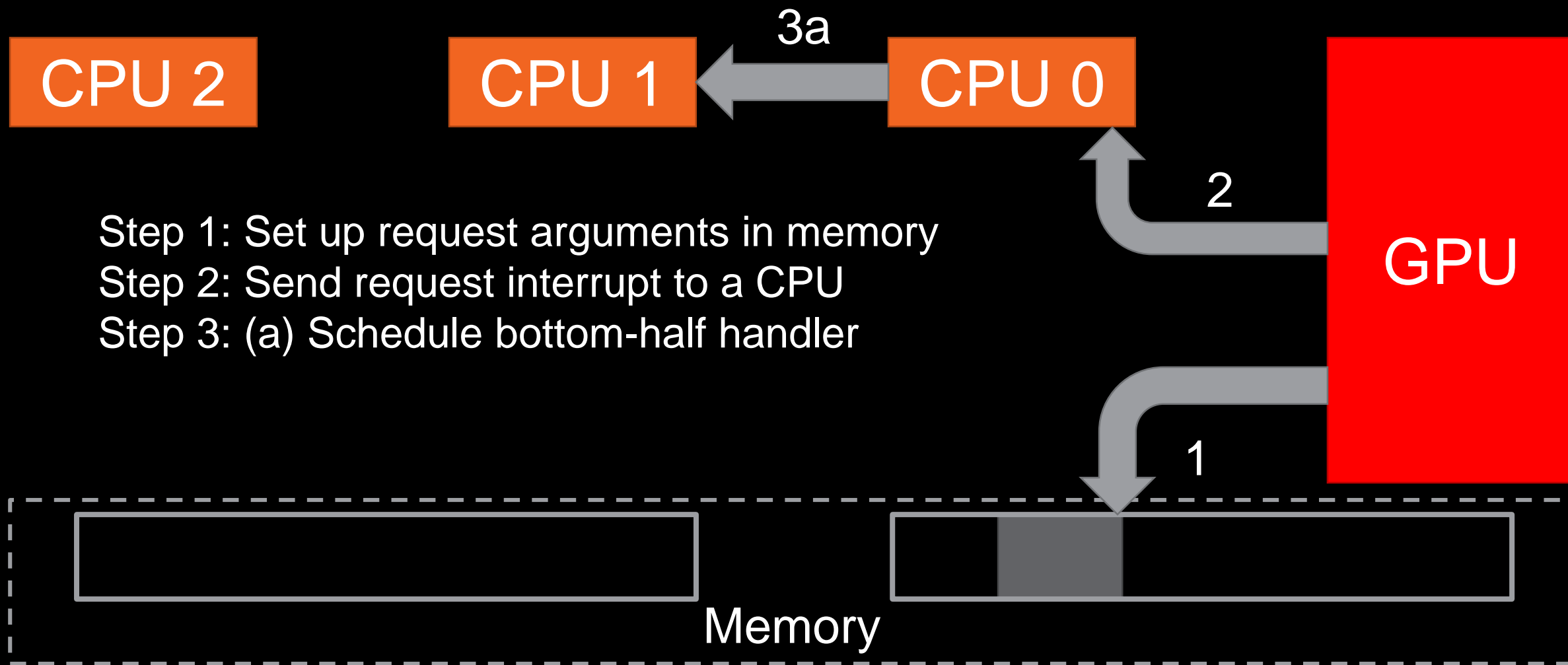
CPU 0

GPU

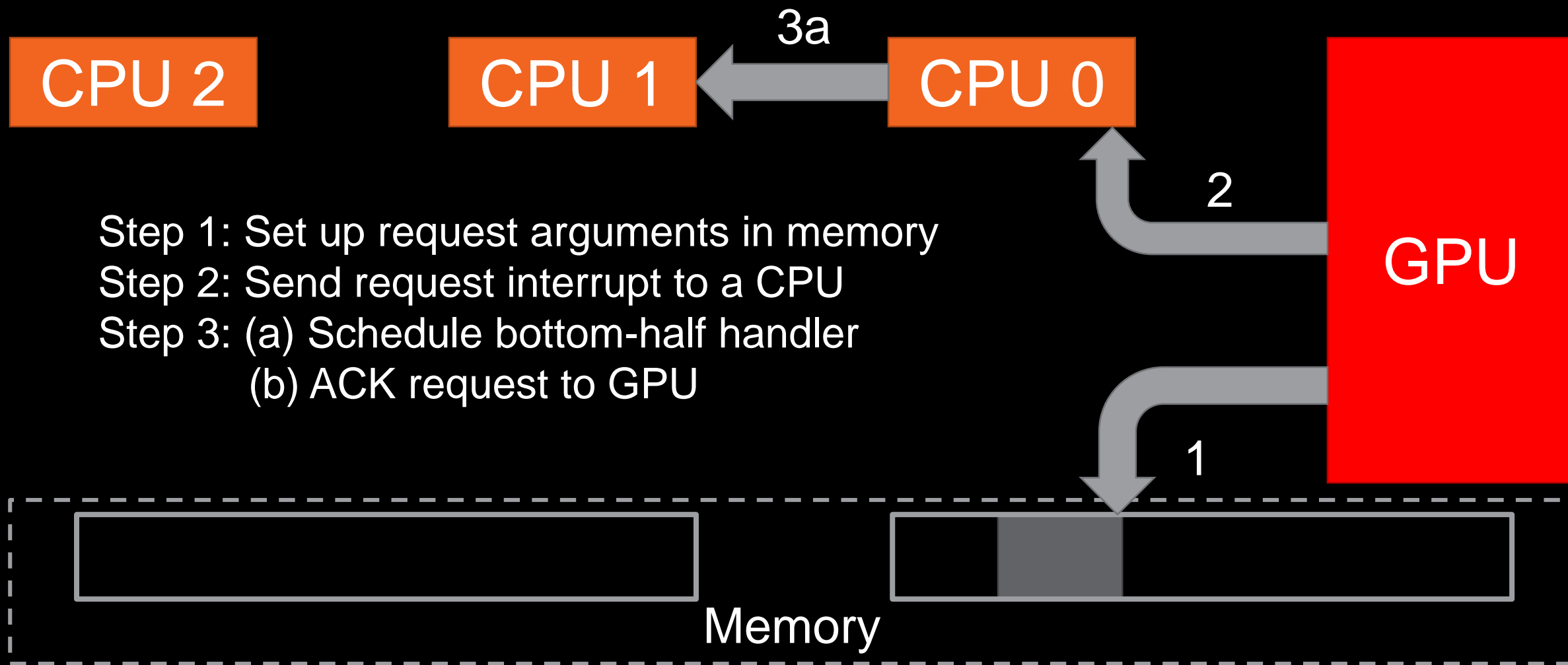
- Step 1: Set up request arguments in memory
- Step 2: Send request interrupt to a CPU
- Step 3: (a) Schedule bottom-half handler



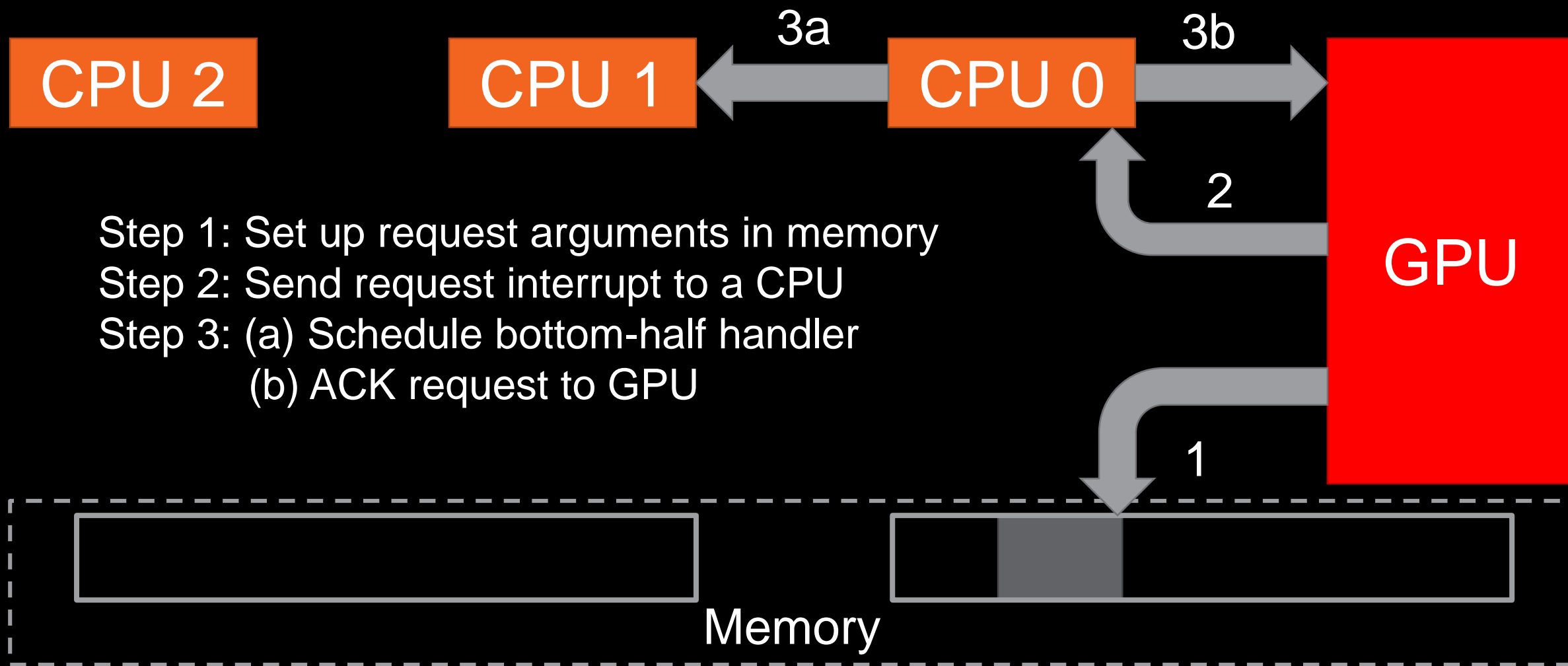
HOW A GPU SYSTEM SERVICE REQUEST IS HANDLED



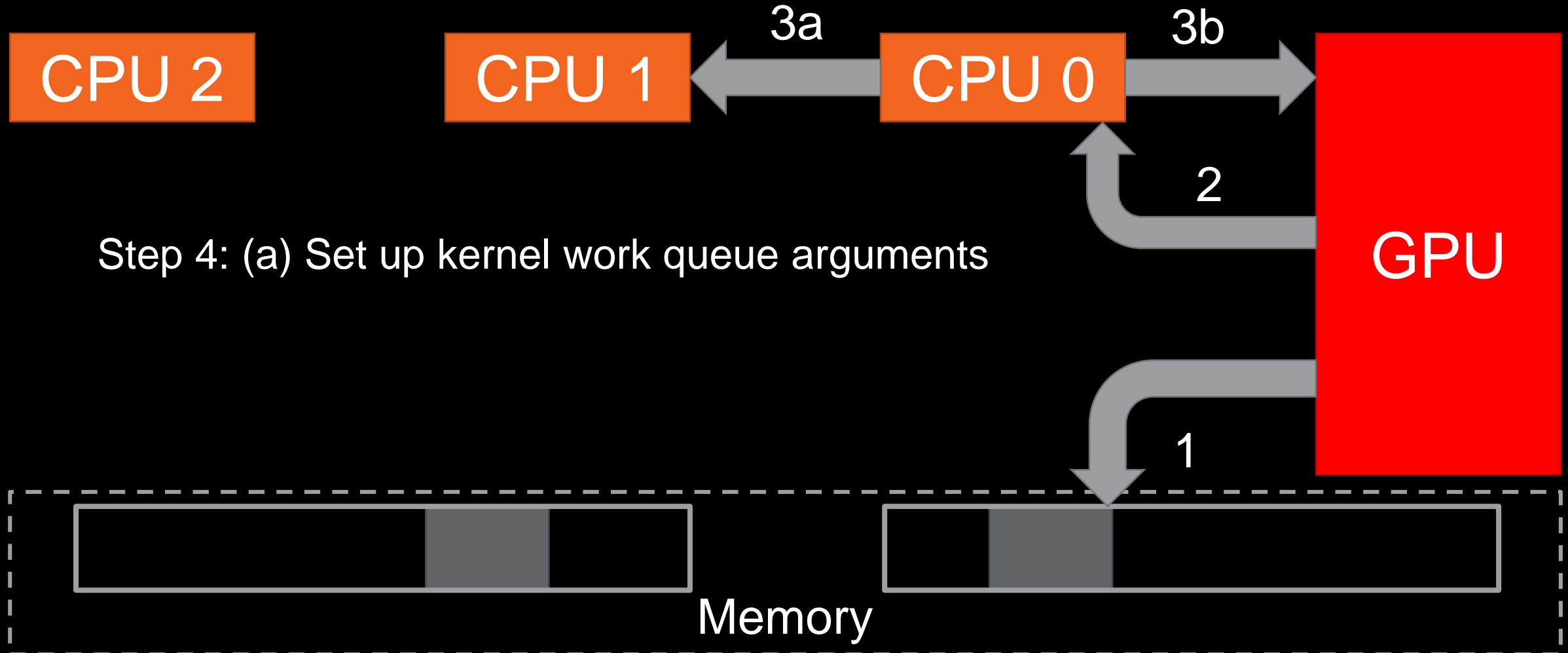
HOW A GPU SYSTEM SERVICE REQUEST IS HANDLED



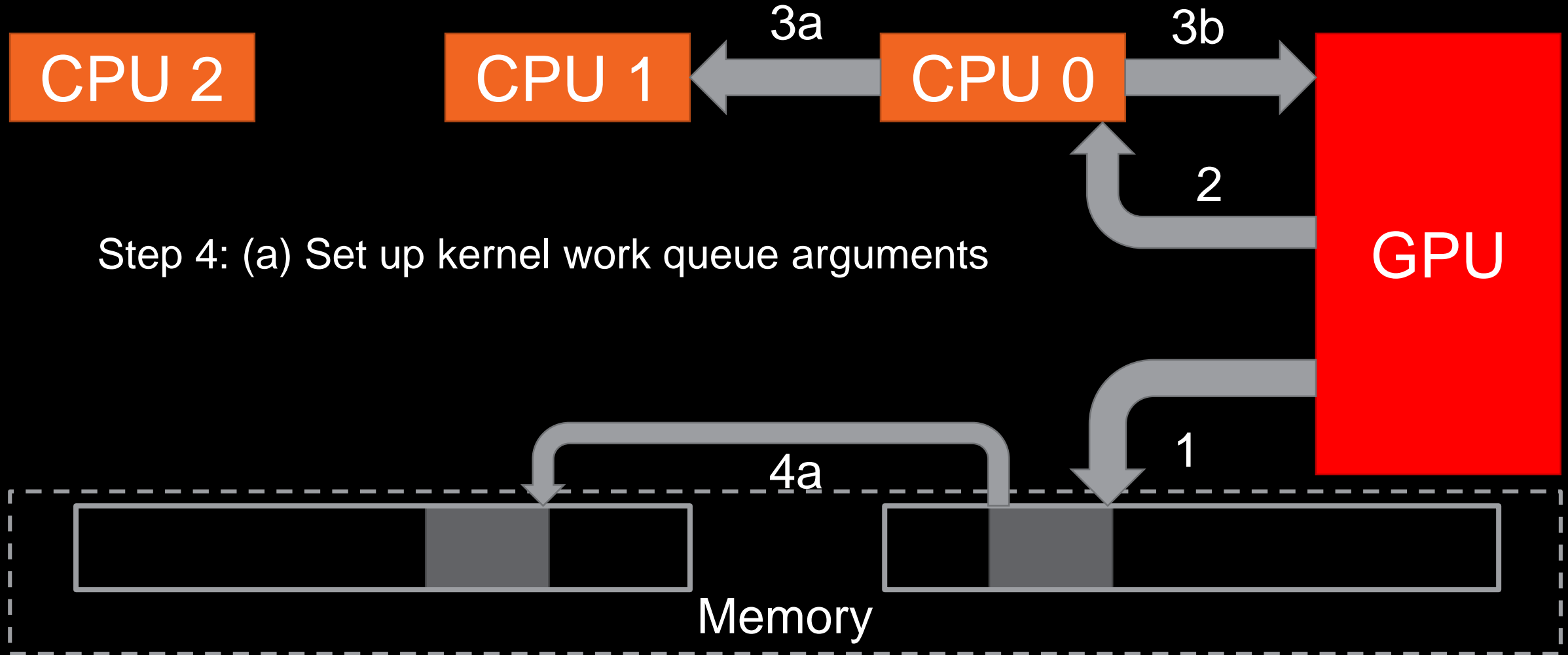
HOW A GPU SYSTEM SERVICE REQUEST IS HANDLED



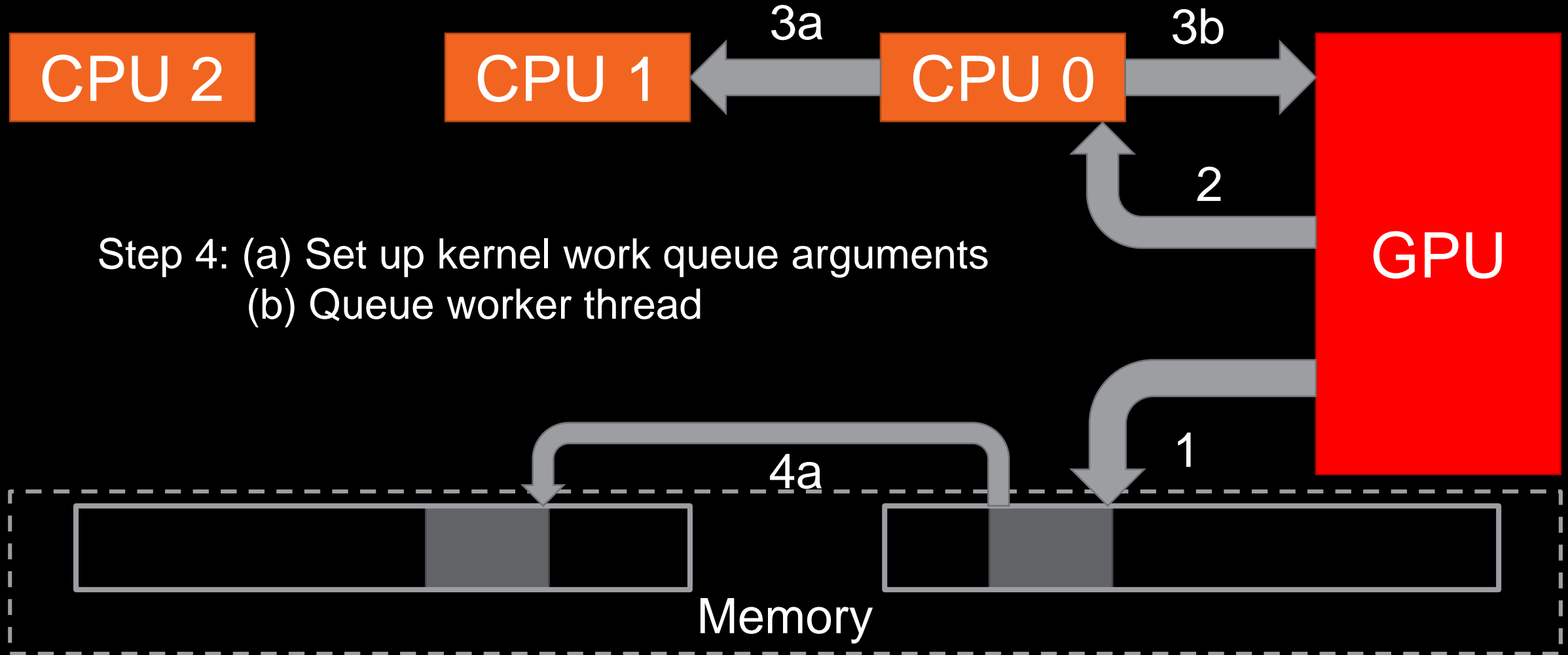
HOW A GPU SYSTEM SERVICE REQUEST IS HANDLED



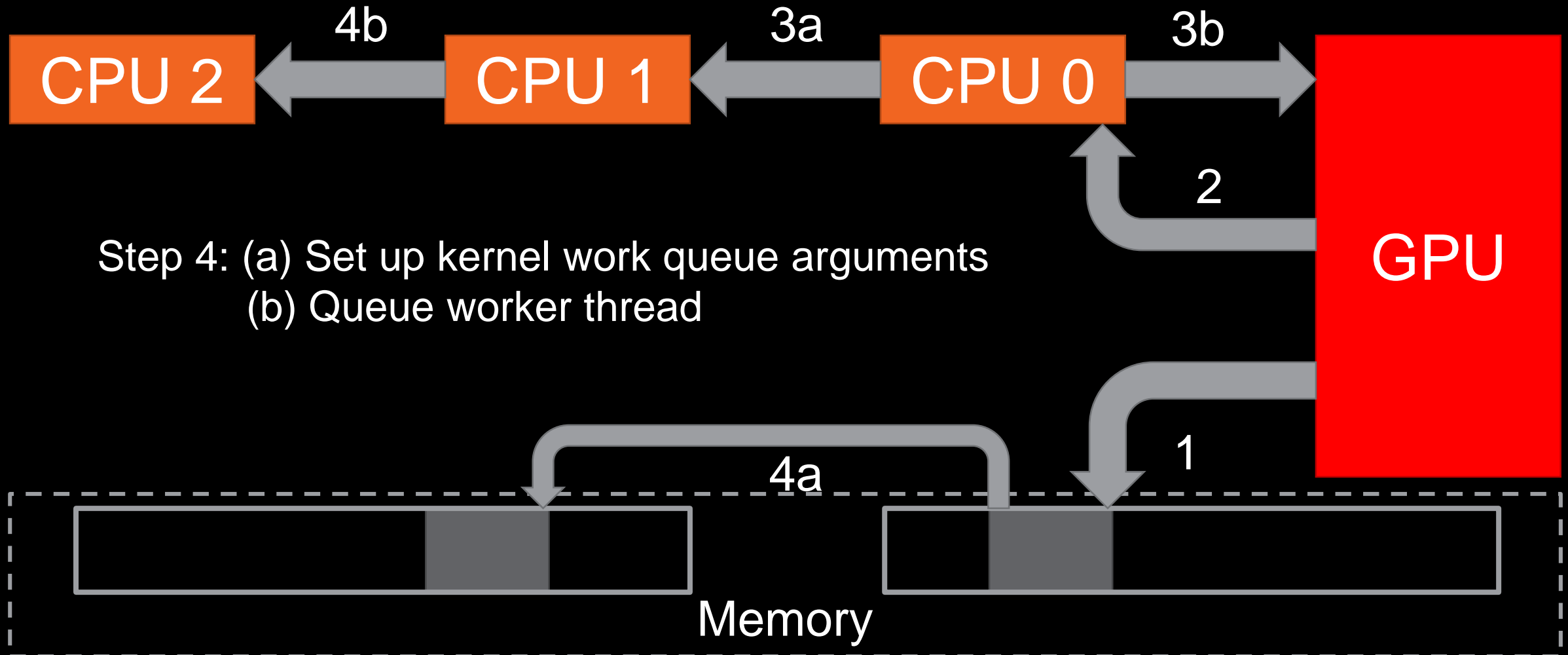
HOW A GPU SYSTEM SERVICE REQUEST IS HANDLED



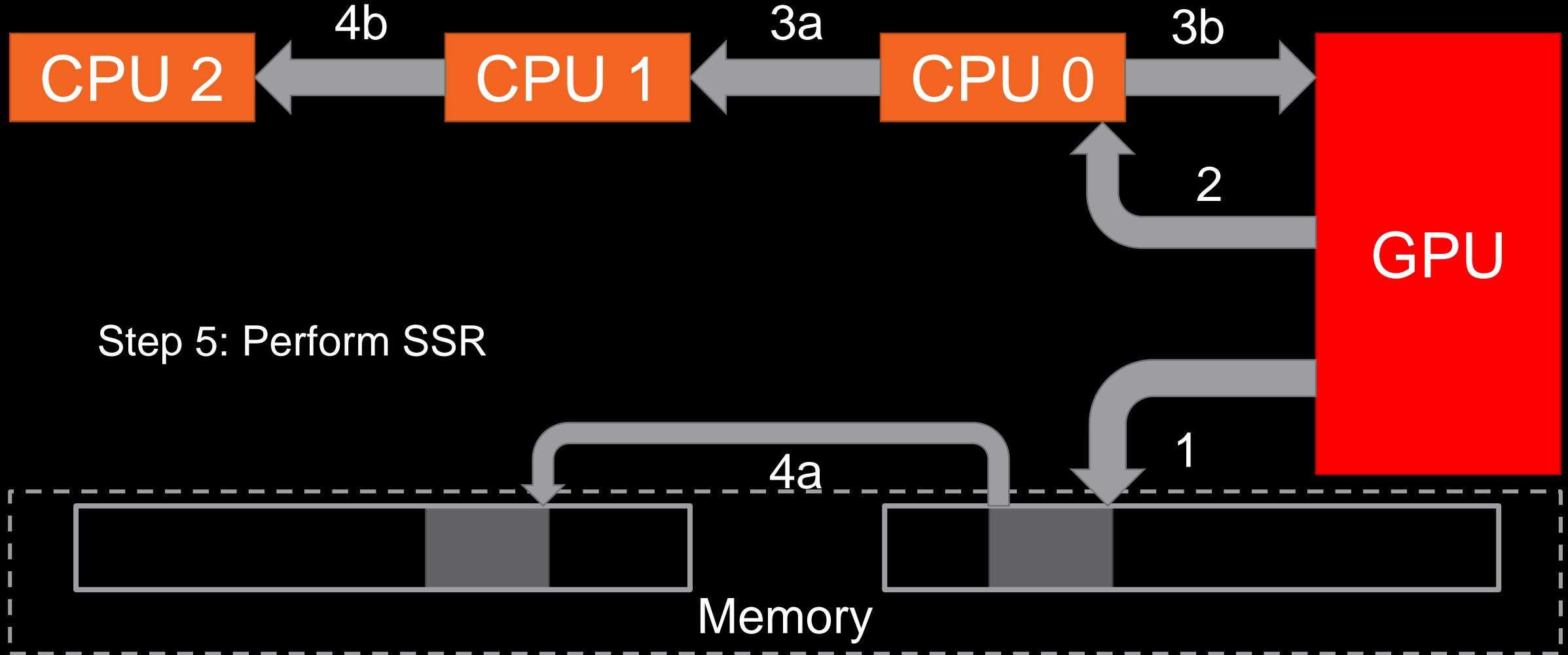
HOW A GPU SYSTEM SERVICE REQUEST IS HANDLED



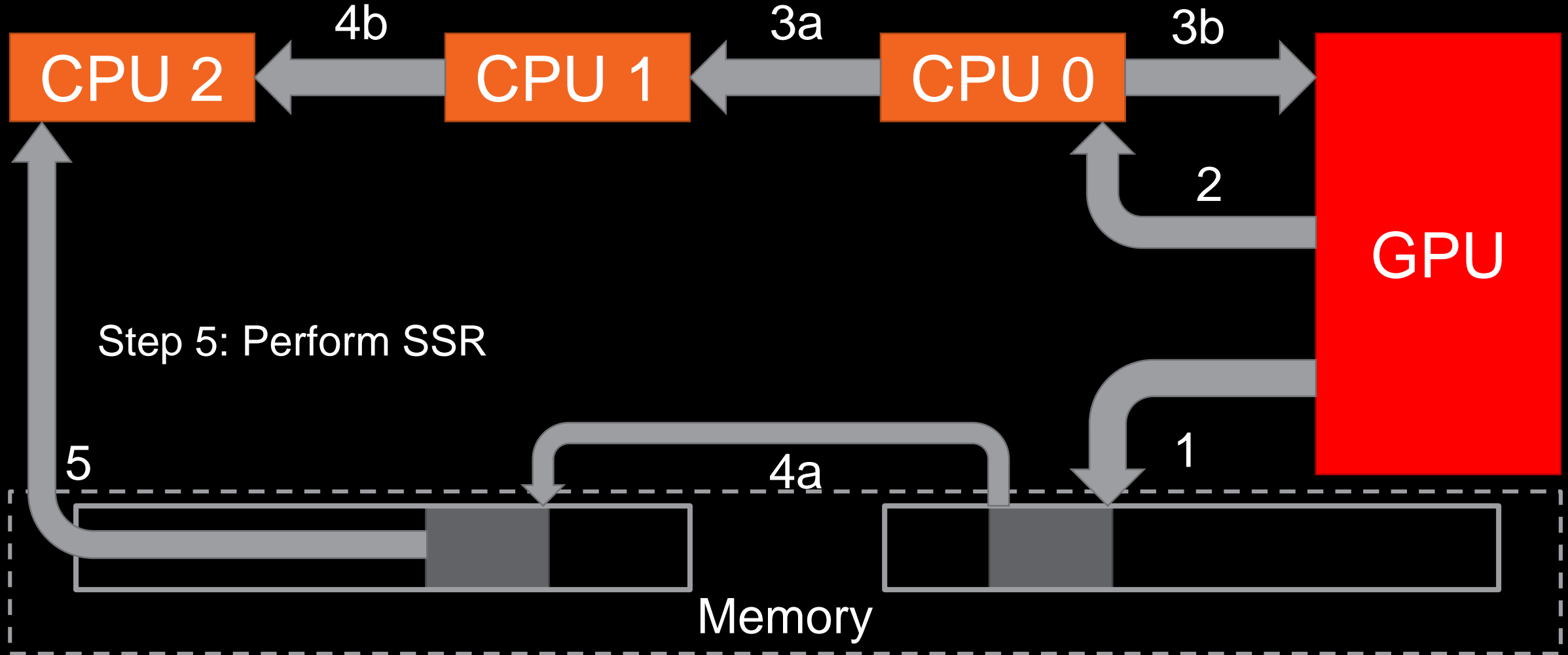
HOW A GPU SYSTEM SERVICE REQUEST IS HANDLED



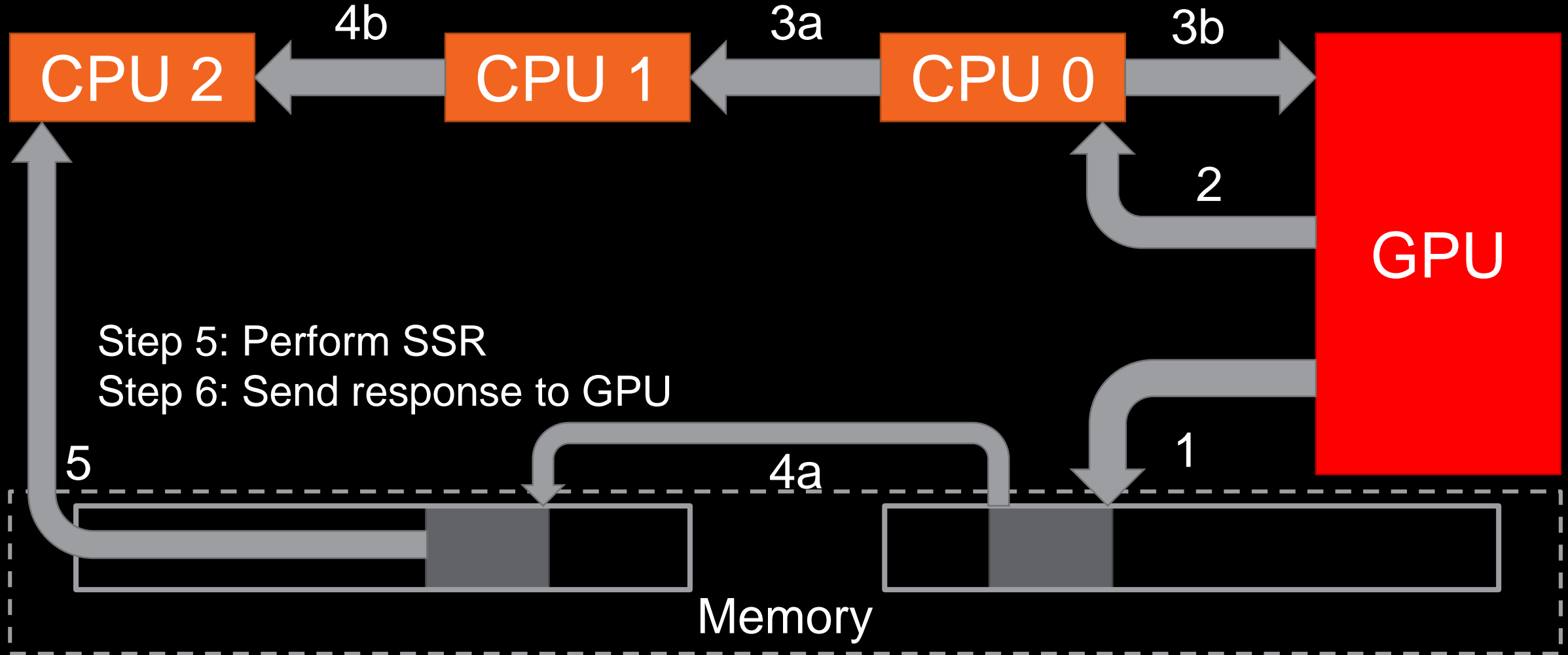
HOW A GPU SYSTEM SERVICE REQUEST IS HANDLED



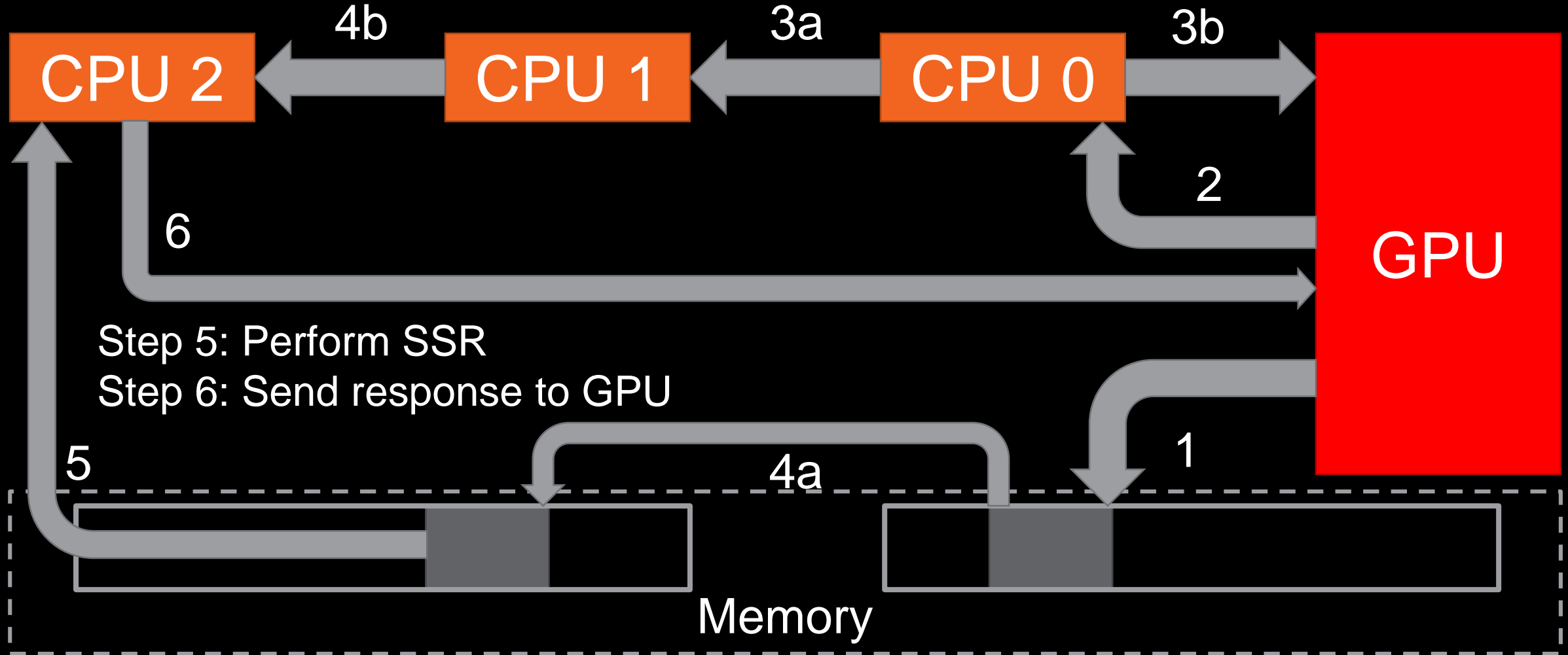
HOW A GPU SYSTEM SERVICE REQUEST IS HANDLED



HOW A GPU SYSTEM SERVICE REQUEST IS HANDLED



HOW A GPU SYSTEM SERVICE REQUEST IS HANDLED



SYSTEM SERVICE REQUESTS CAN LEAD TO DIFFICULTIES

SYSTEM SERVICE REQUESTS CAN LEAD TO DIFFICULTIES

- GPUs and accelerators can request OS (system) services

SYSTEM SERVICE REQUESTS CAN LEAD TO DIFFICULTIES

- GPUs and accelerators can request OS (system) services
- These SSRs can interfere with unrelated CPU-based work

SYSTEM SERVICE REQUESTS CAN LEAD TO DIFFICULTIES

- GPUs and accelerators can request OS (system) services
- These SSRs can interfere with unrelated CPU-based work
- Unrelated CPU-Based work can slow down GPU SSR handling

SYSTEM SERVICE REQUESTS CAN LEAD TO DIFFICULTIES

- GPUs and accelerators can request OS (system) services
- These SSRs can interfere with unrelated CPU-based work
- Unrelated CPU-Based work can slow down GPU SSR handling
- CPUs lose opportunity to sleep because of GPU SSRs

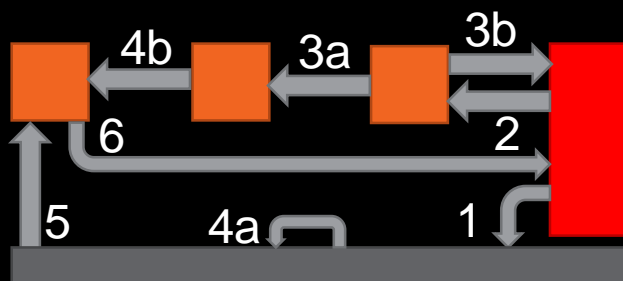
MODES OF INTERFERENCE FROM GPU SSRS

GPU

CPU0

CPU1

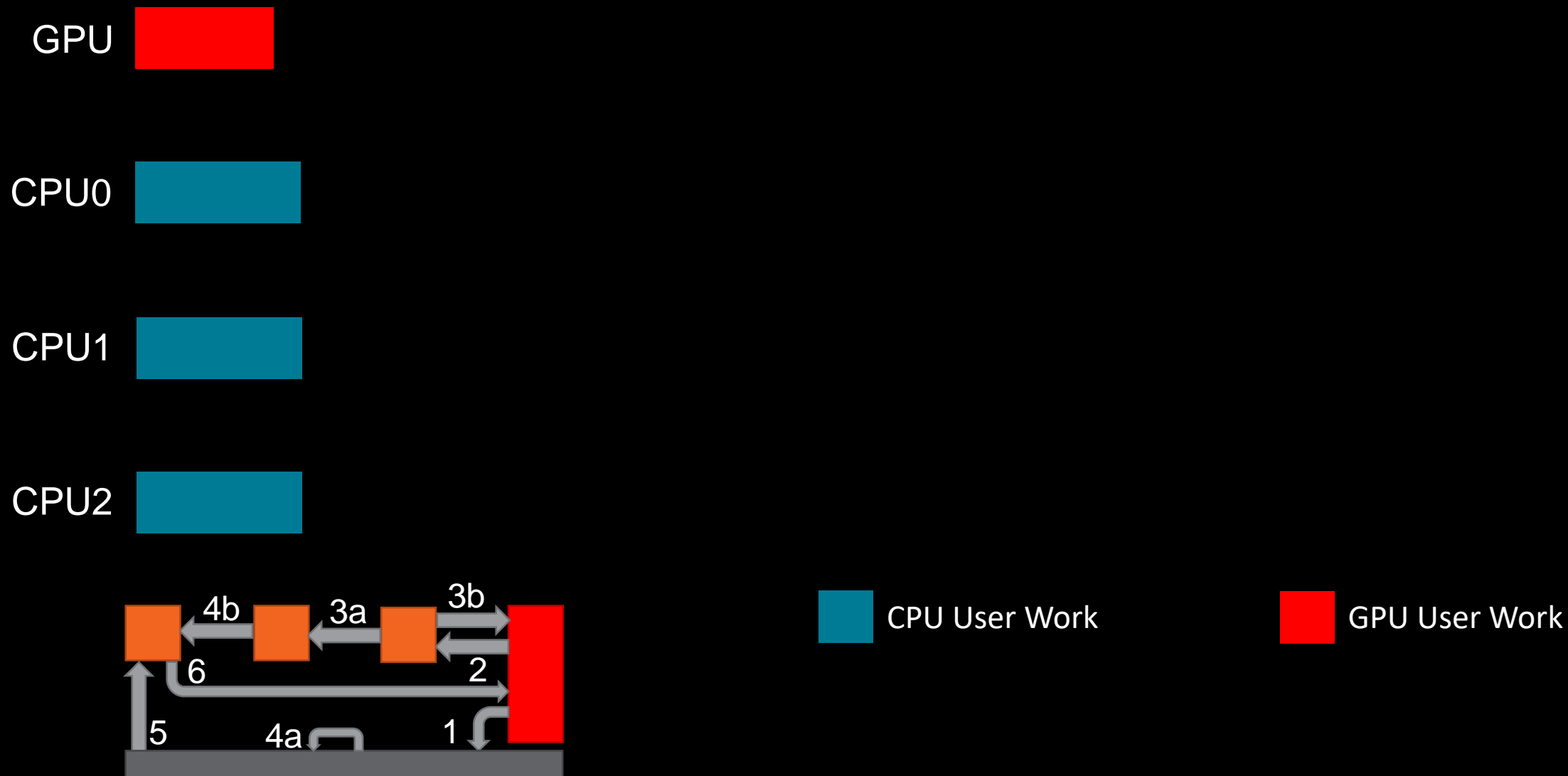
CPU2



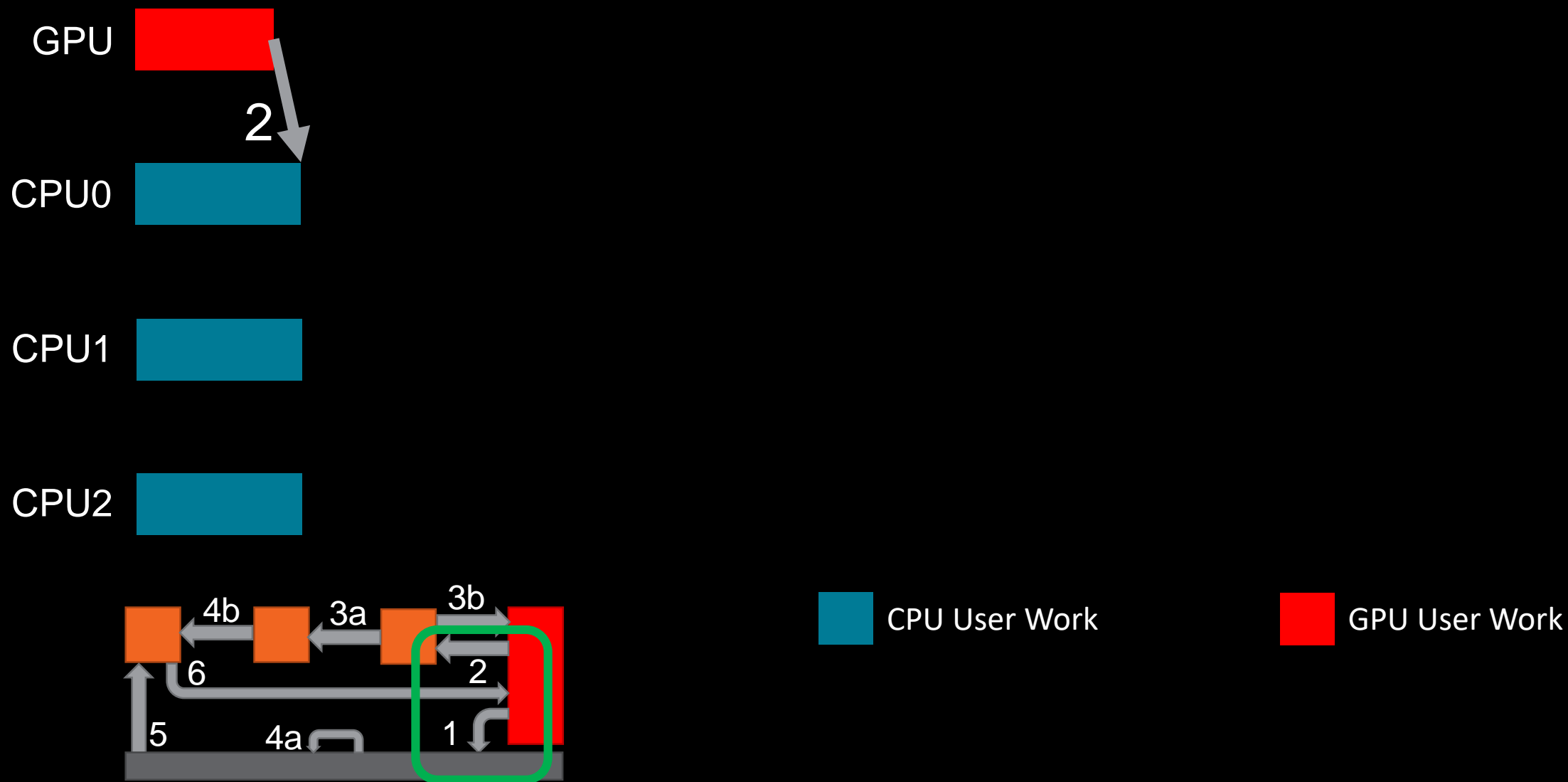
 CPU User Work

 GPU User Work

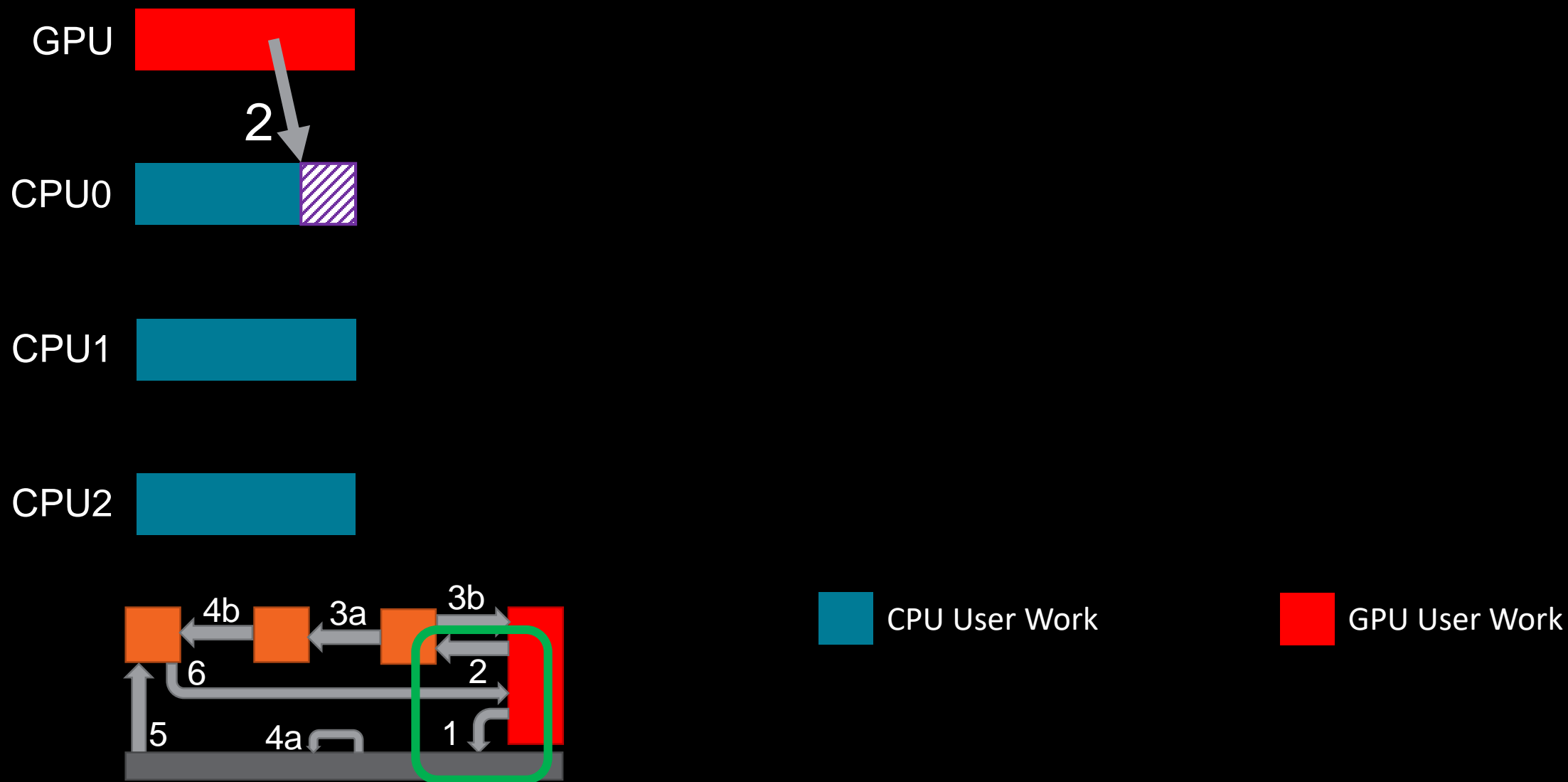
MODES OF INTERFERENCE FROM GPU SSRS



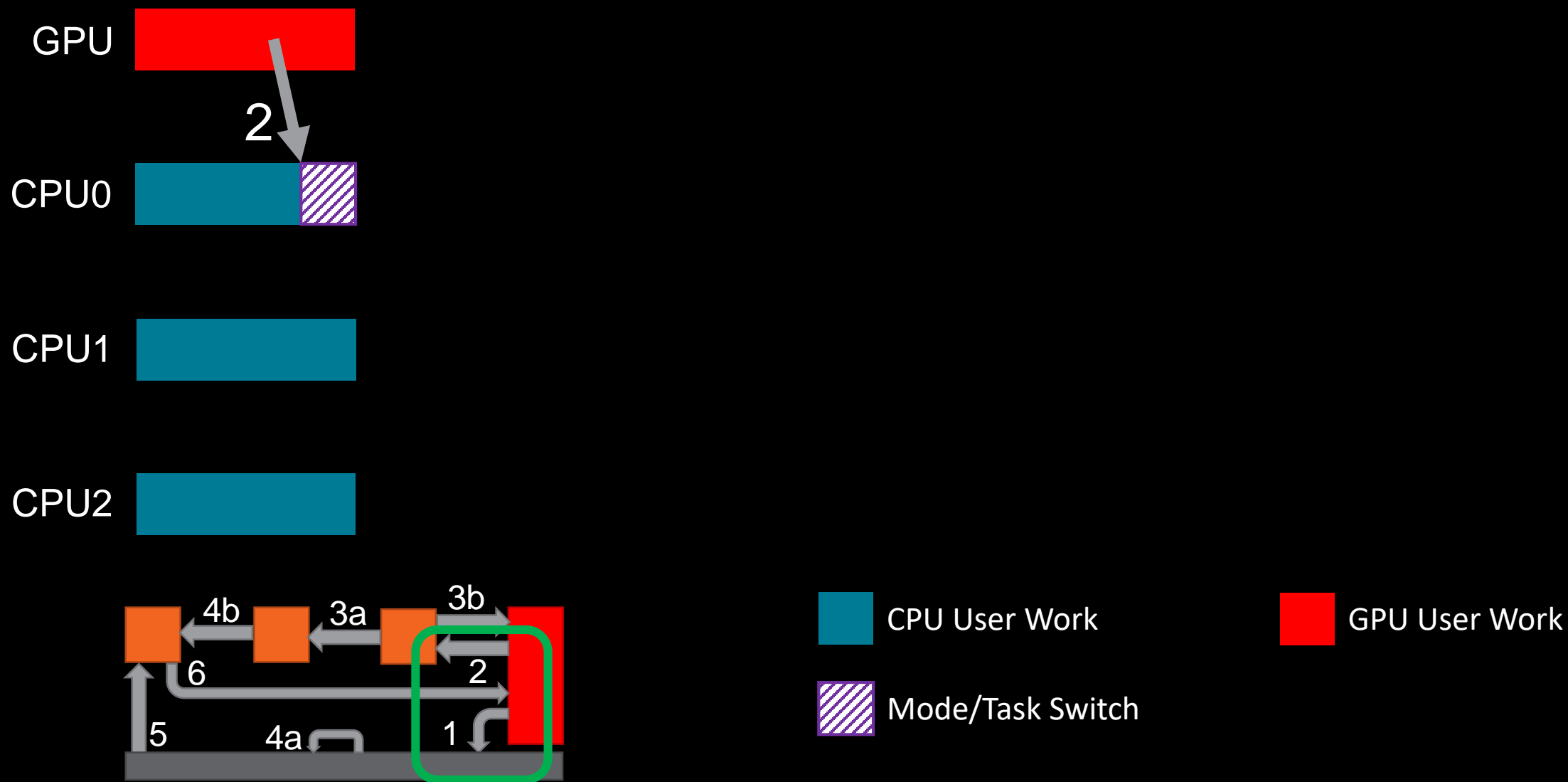
MODES OF INTERFERENCE FROM GPU SSRS



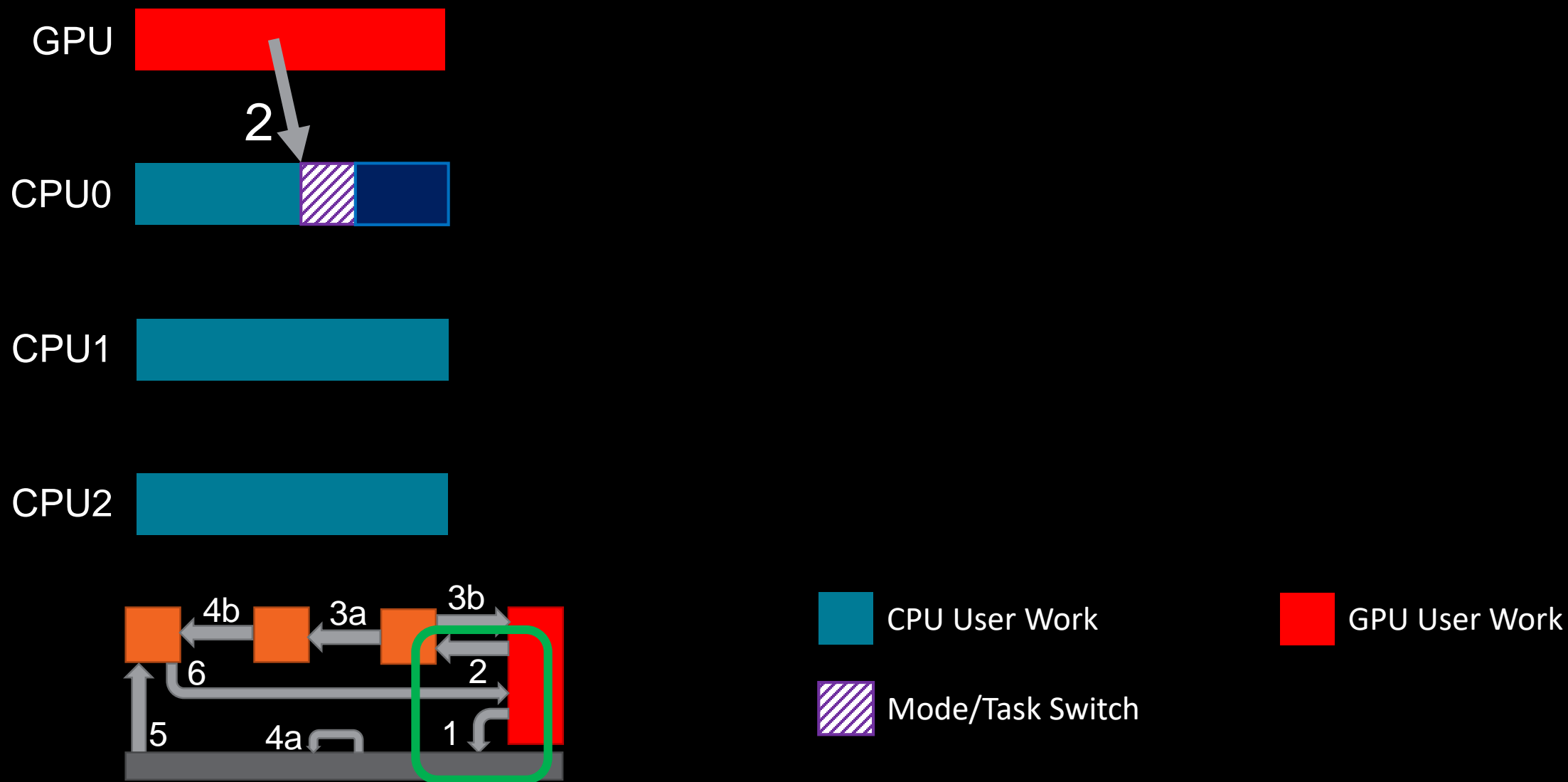
MODES OF INTERFERENCE FROM GPU SSRS



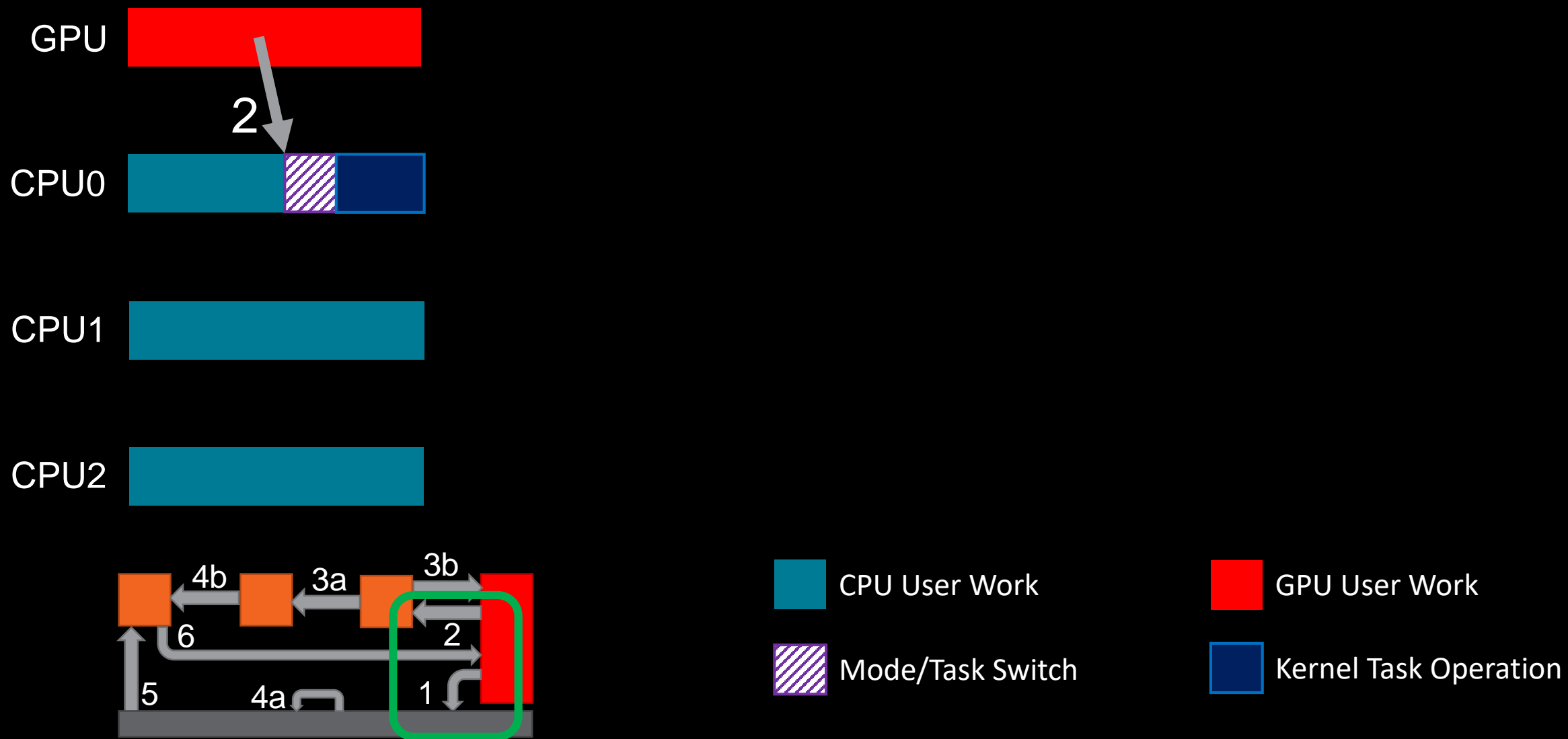
MODES OF INTERFERENCE FROM GPU SSRS



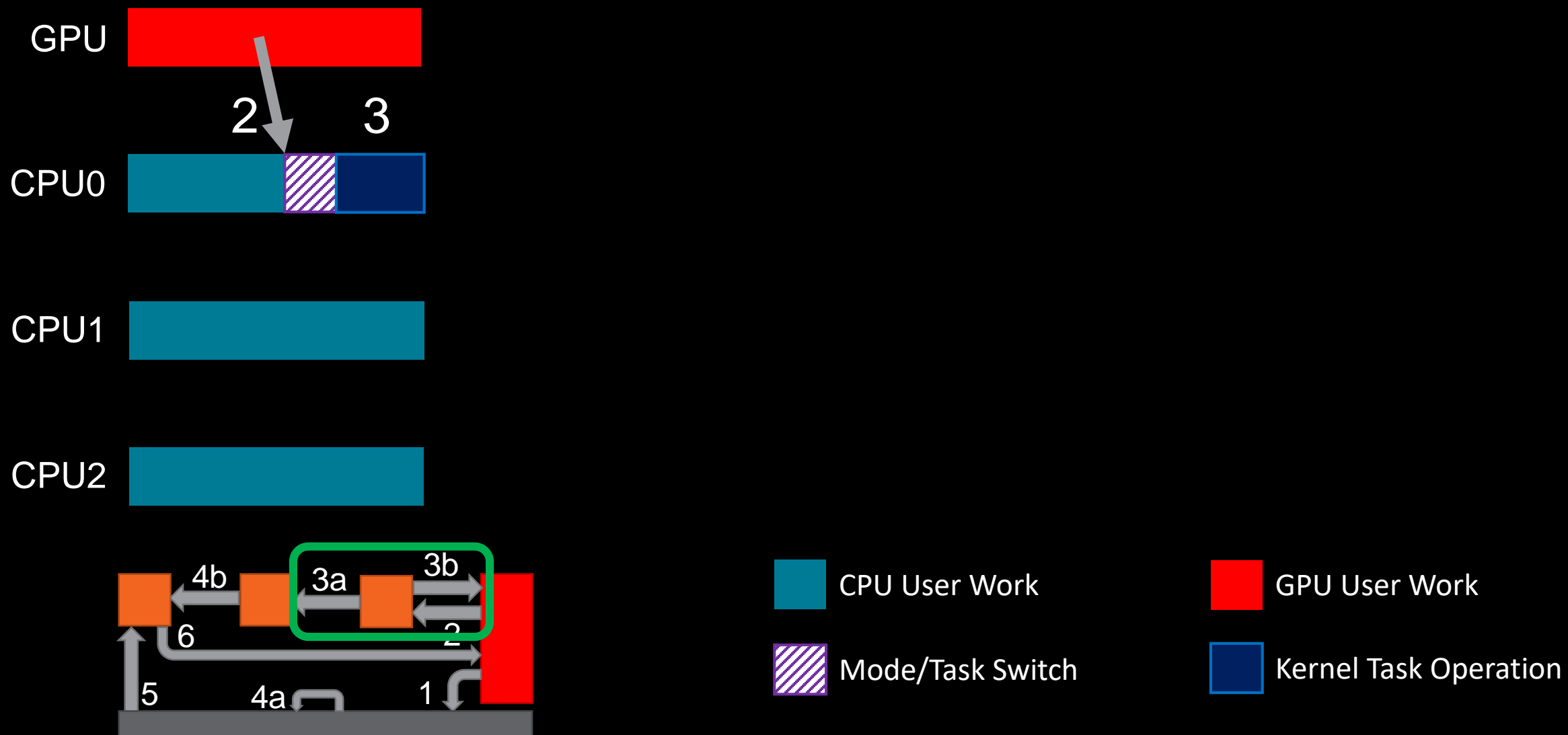
MODES OF INTERFERENCE FROM GPU SSRS



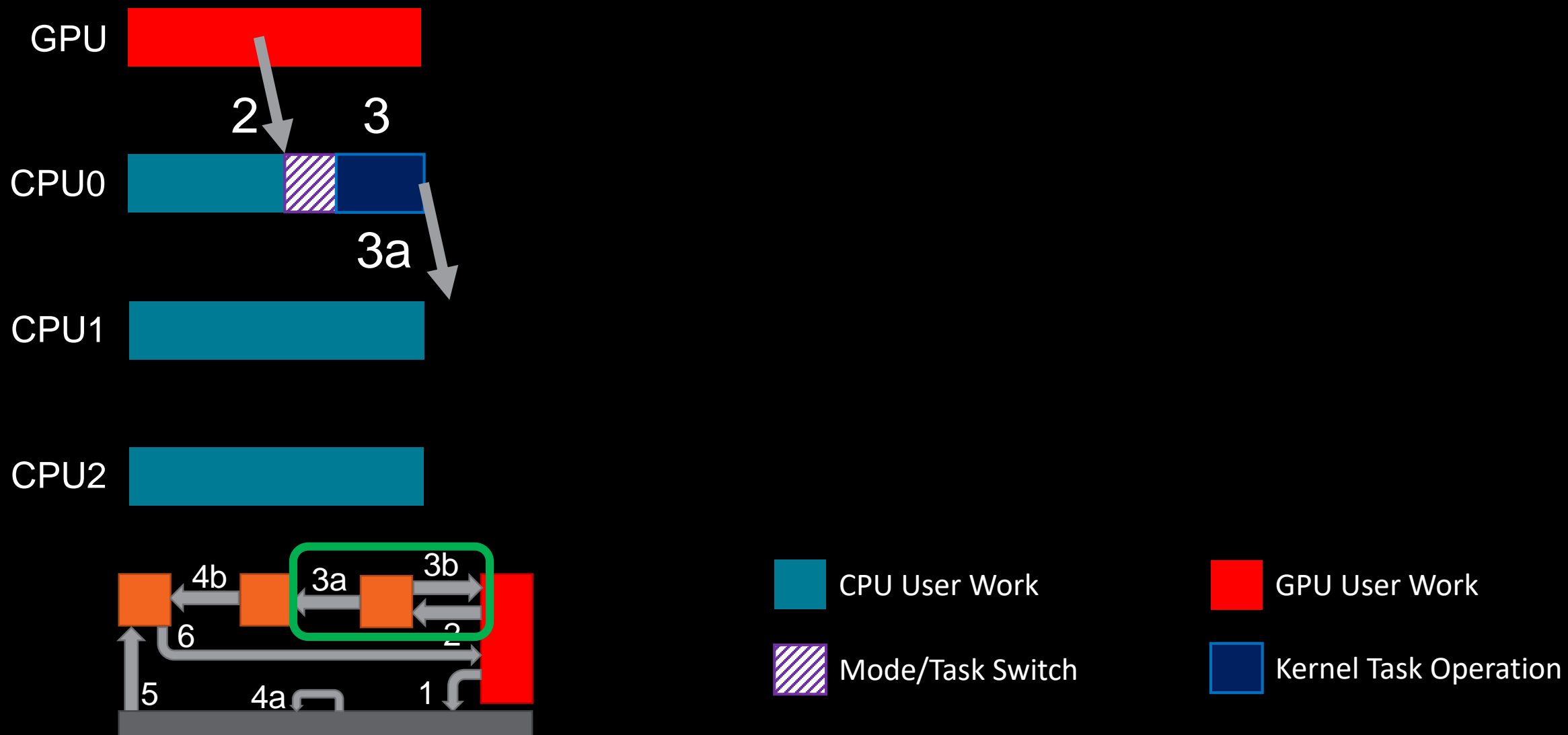
MODES OF INTERFERENCE FROM GPU SSRS



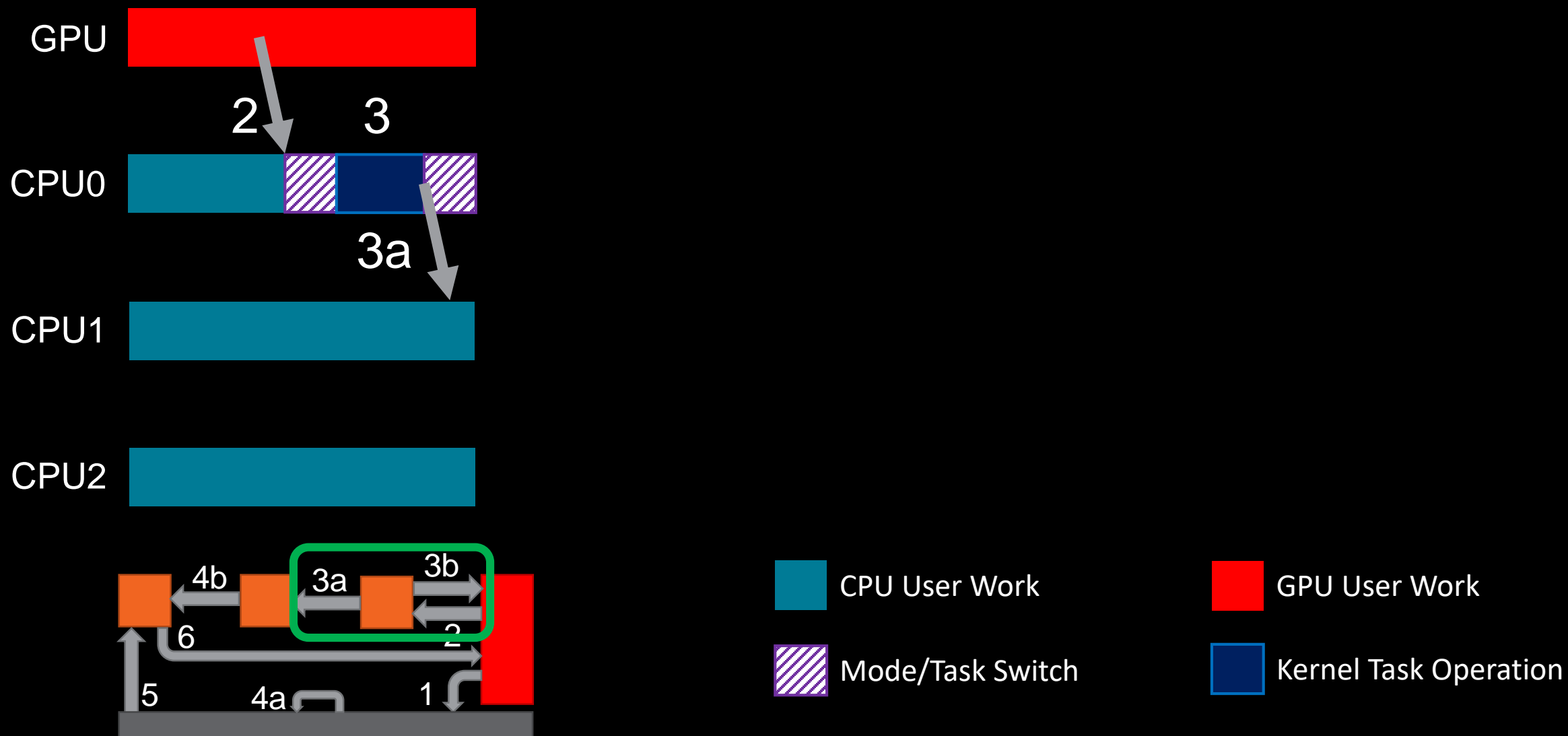
MODES OF INTERFERENCE FROM GPU SSRS



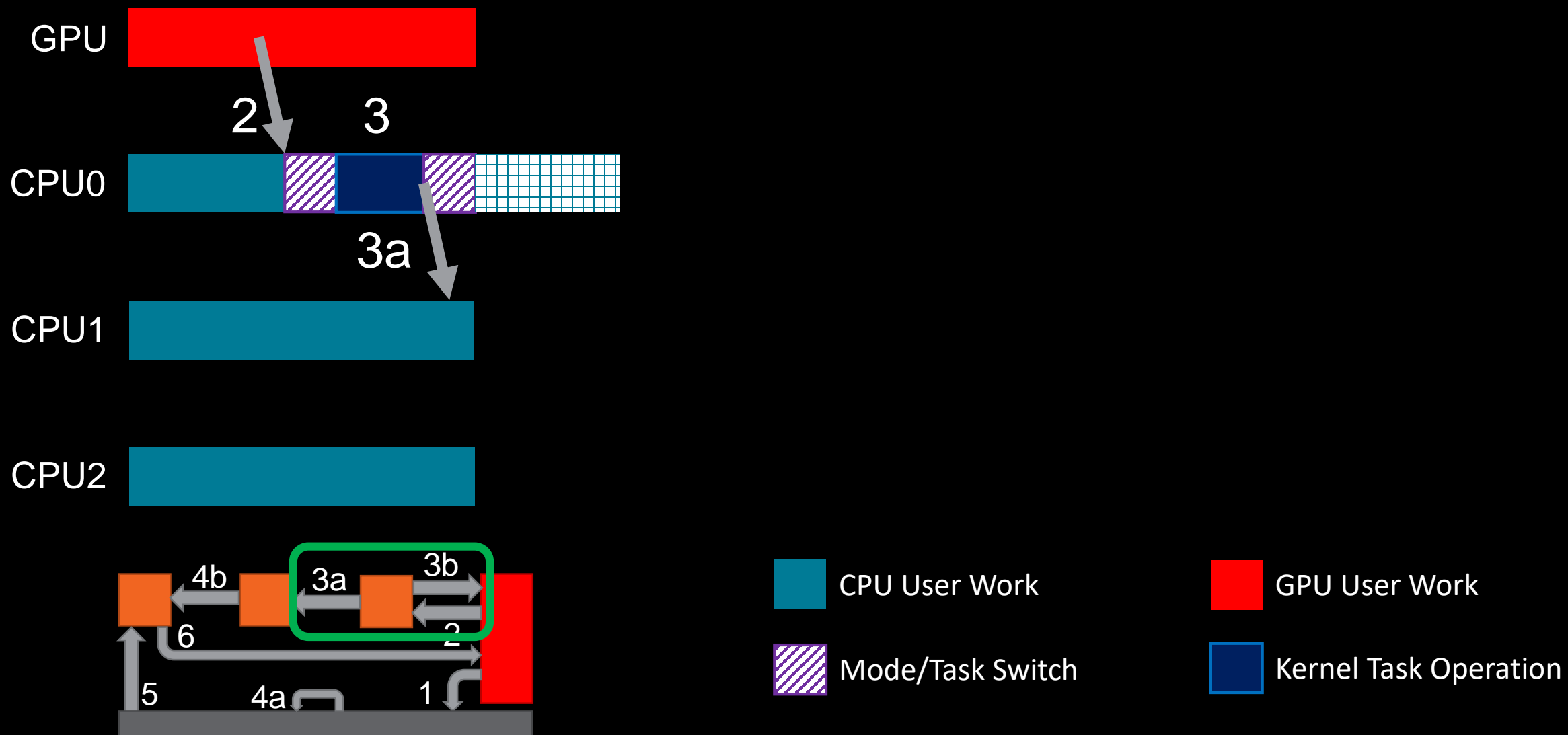
MODES OF INTERFERENCE FROM GPU SSRS



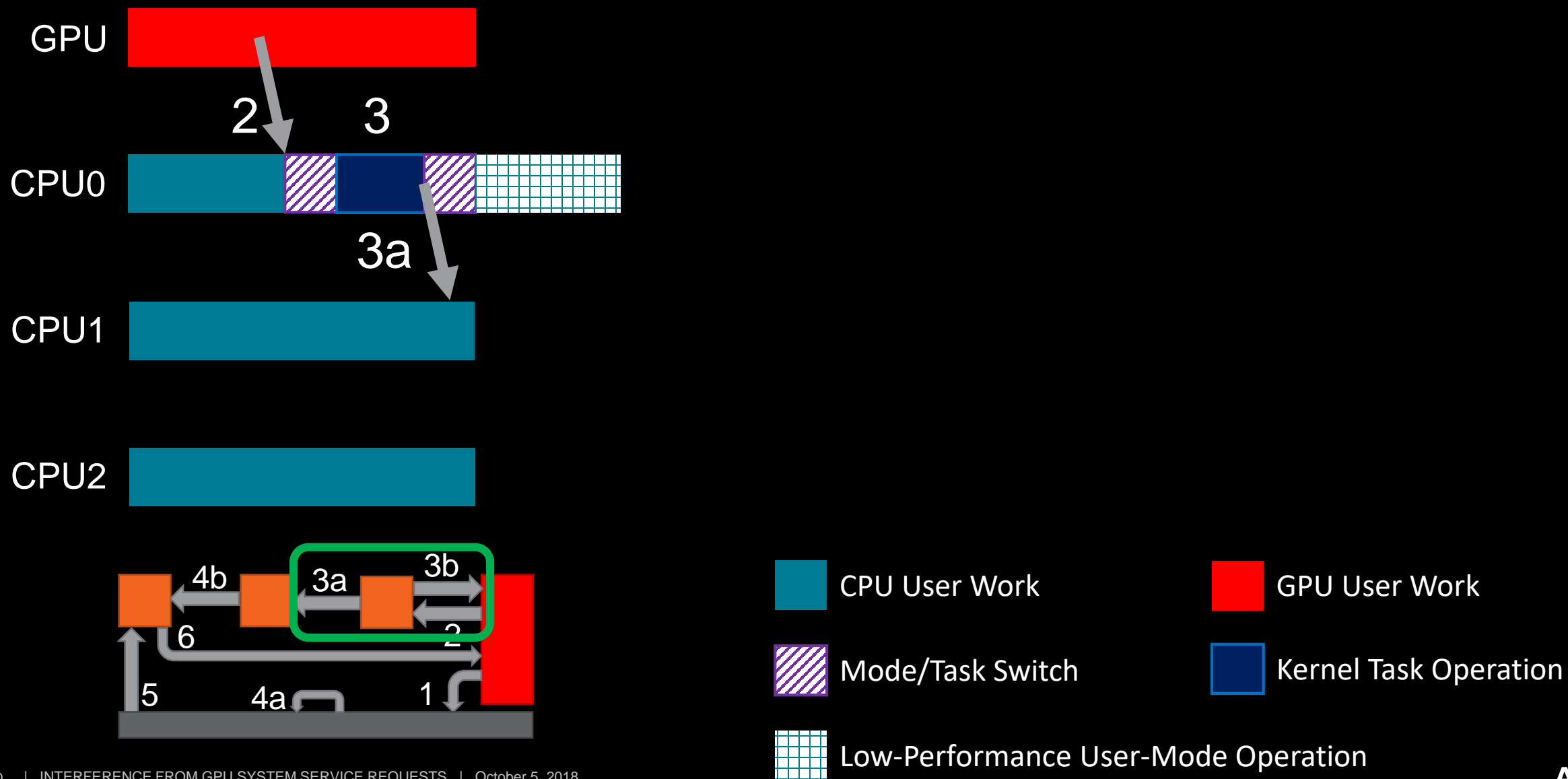
MODES OF INTERFERENCE FROM GPU SSRS



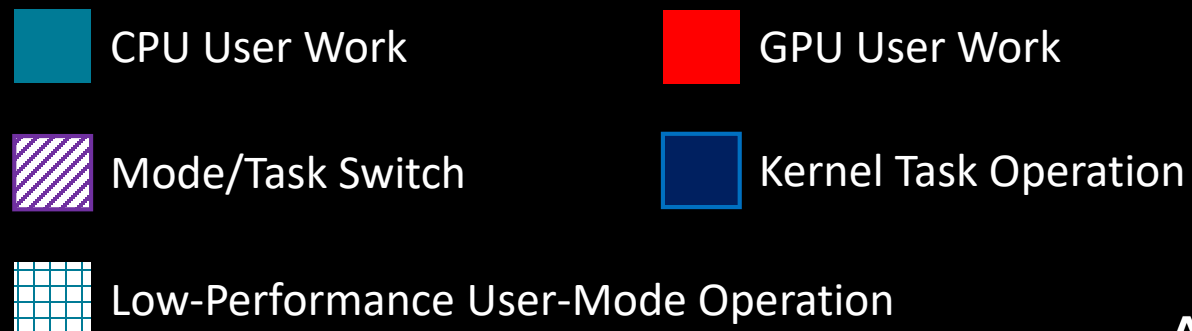
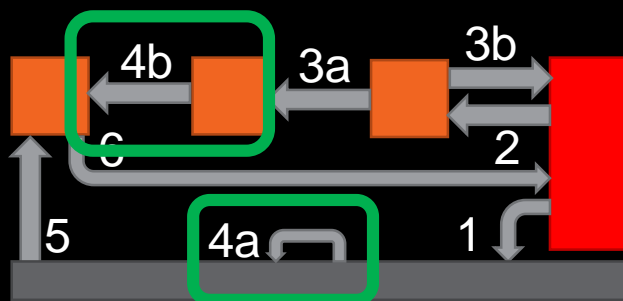
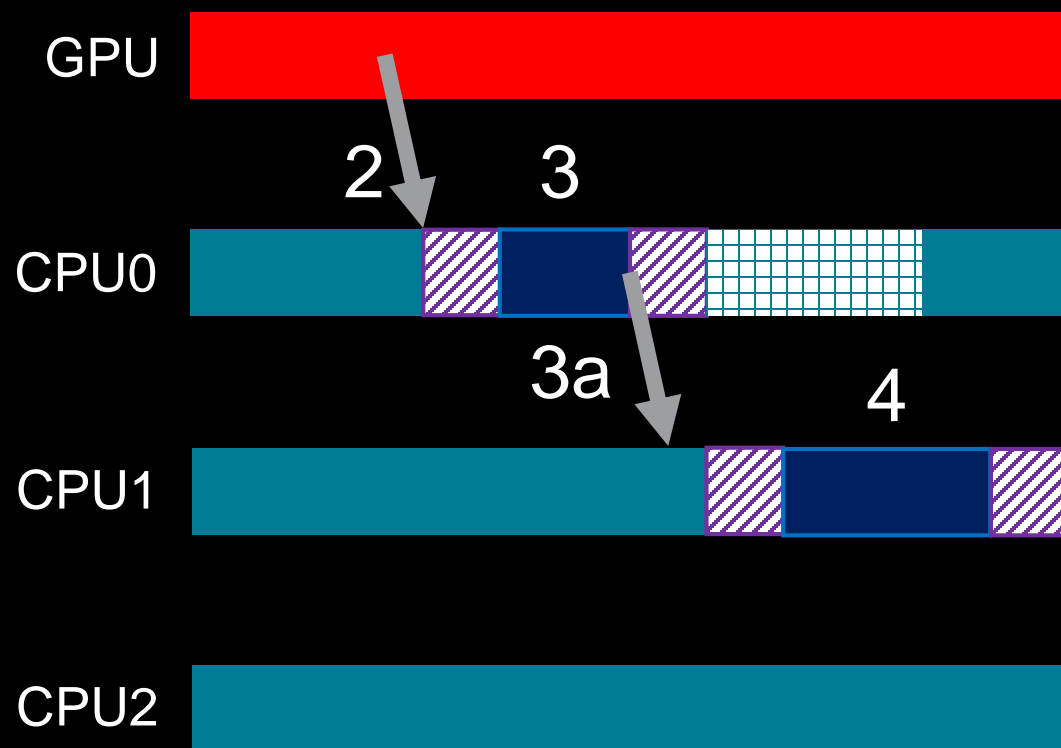
MODES OF INTERFERENCE FROM GPU SSRS



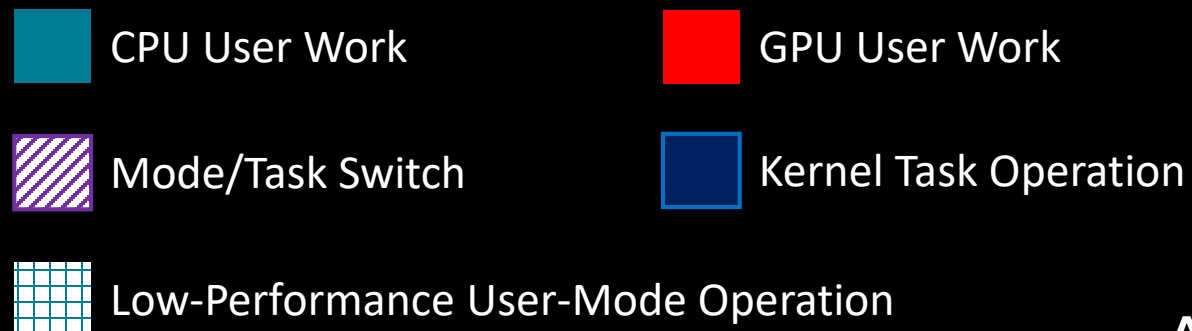
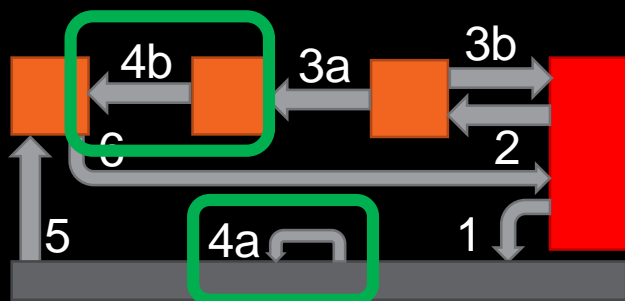
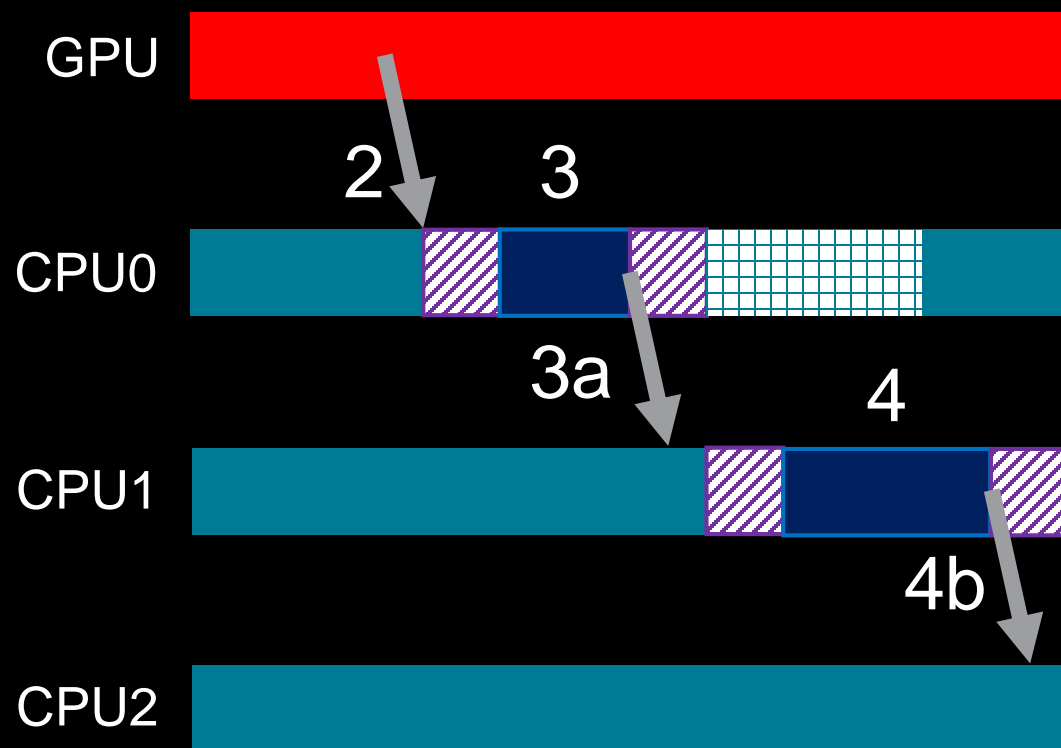
MODES OF INTERFERENCE FROM GPU SSRS



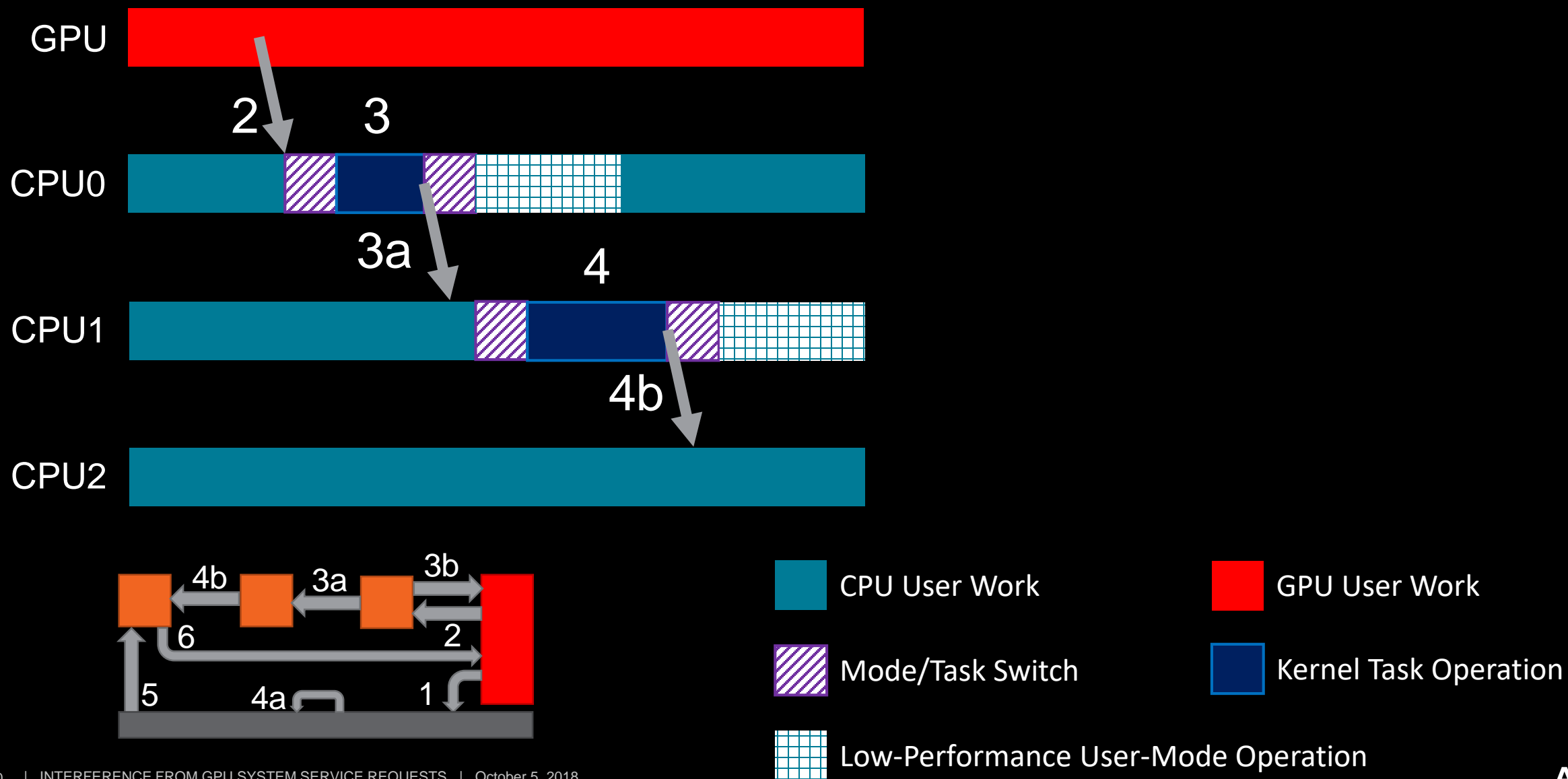
MODES OF INTERFERENCE FROM GPU SSRS



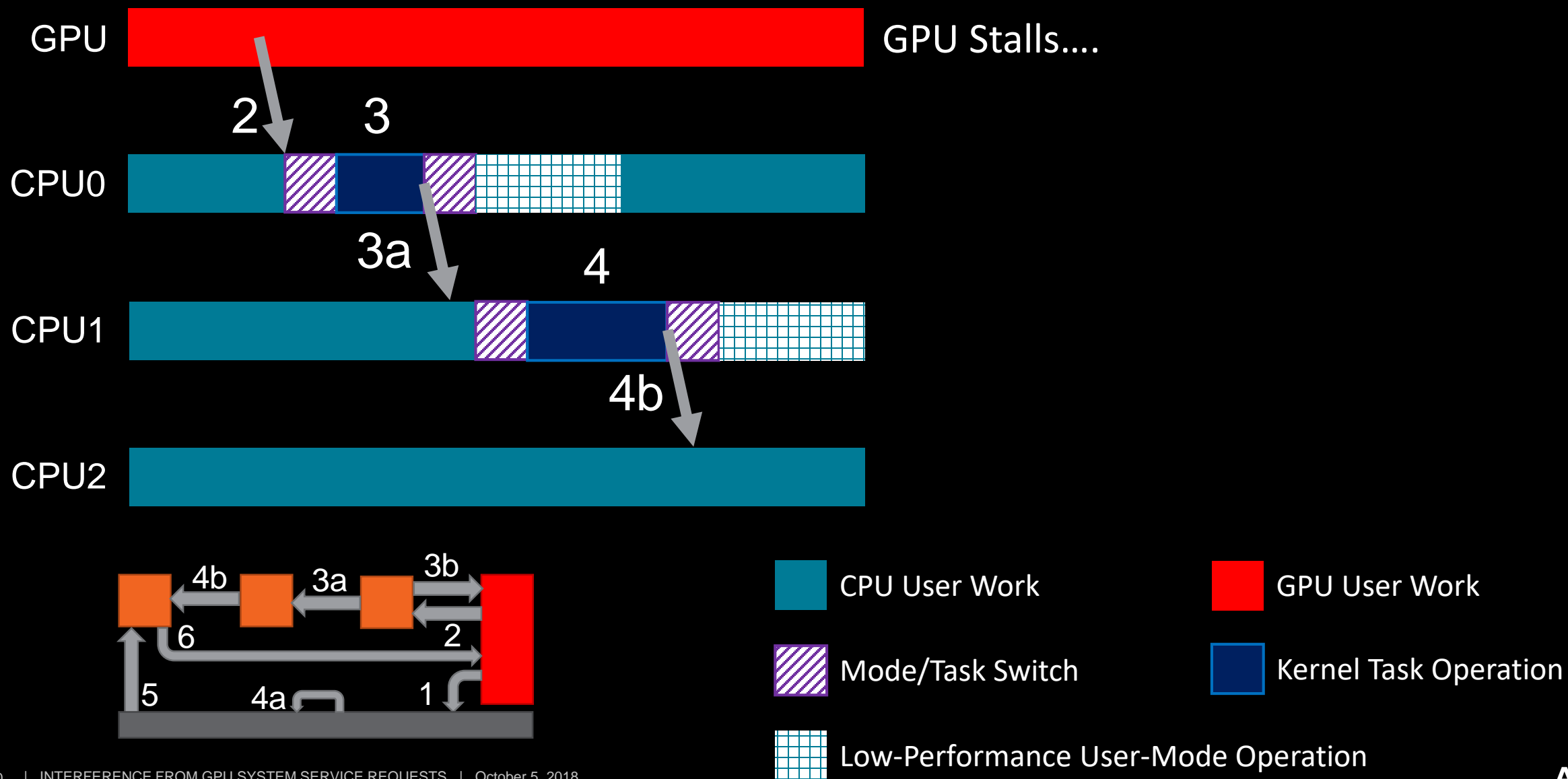
MODES OF INTERFERENCE FROM GPU SSRS



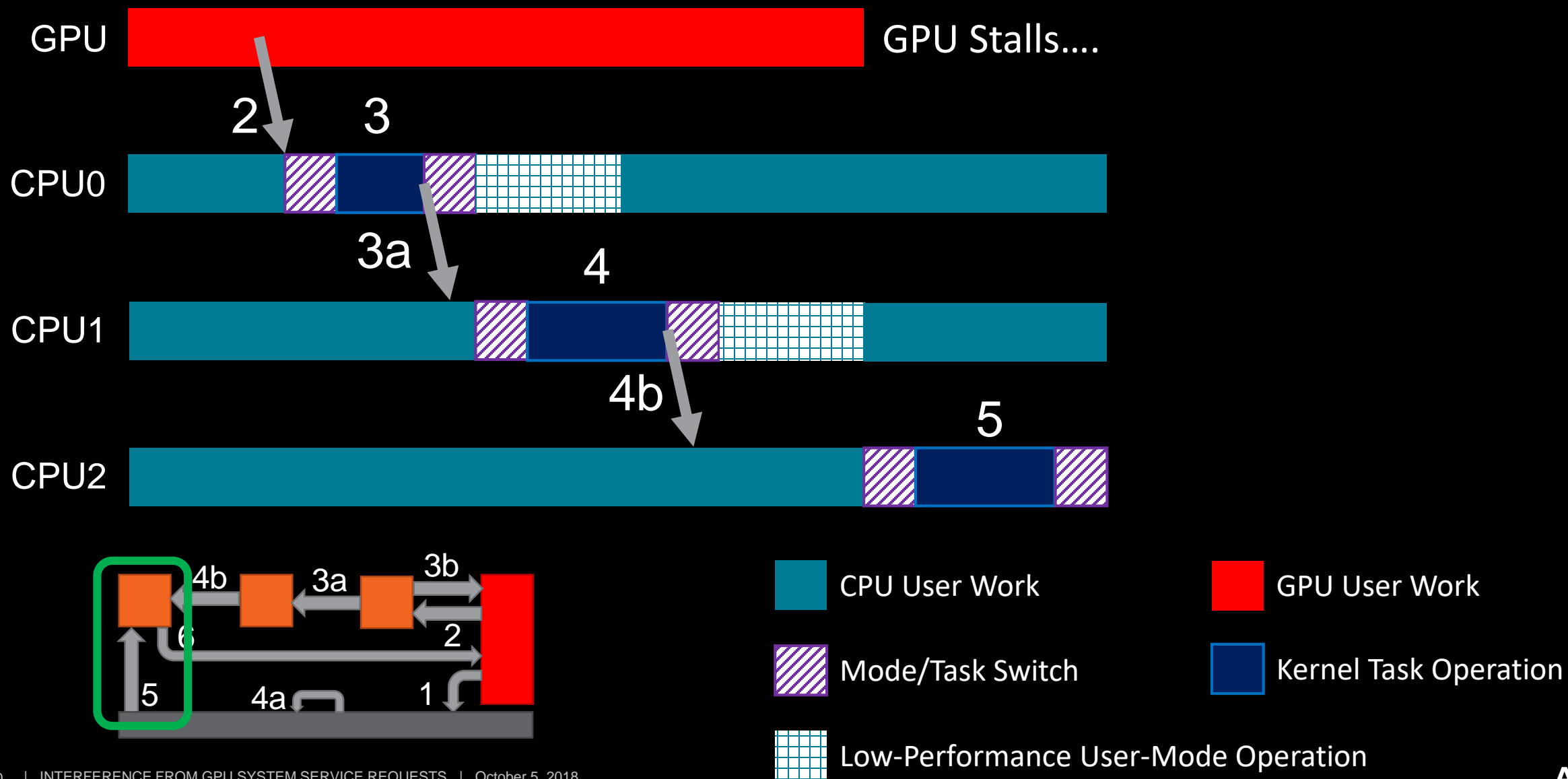
MODES OF INTERFERENCE FROM GPU SSRS



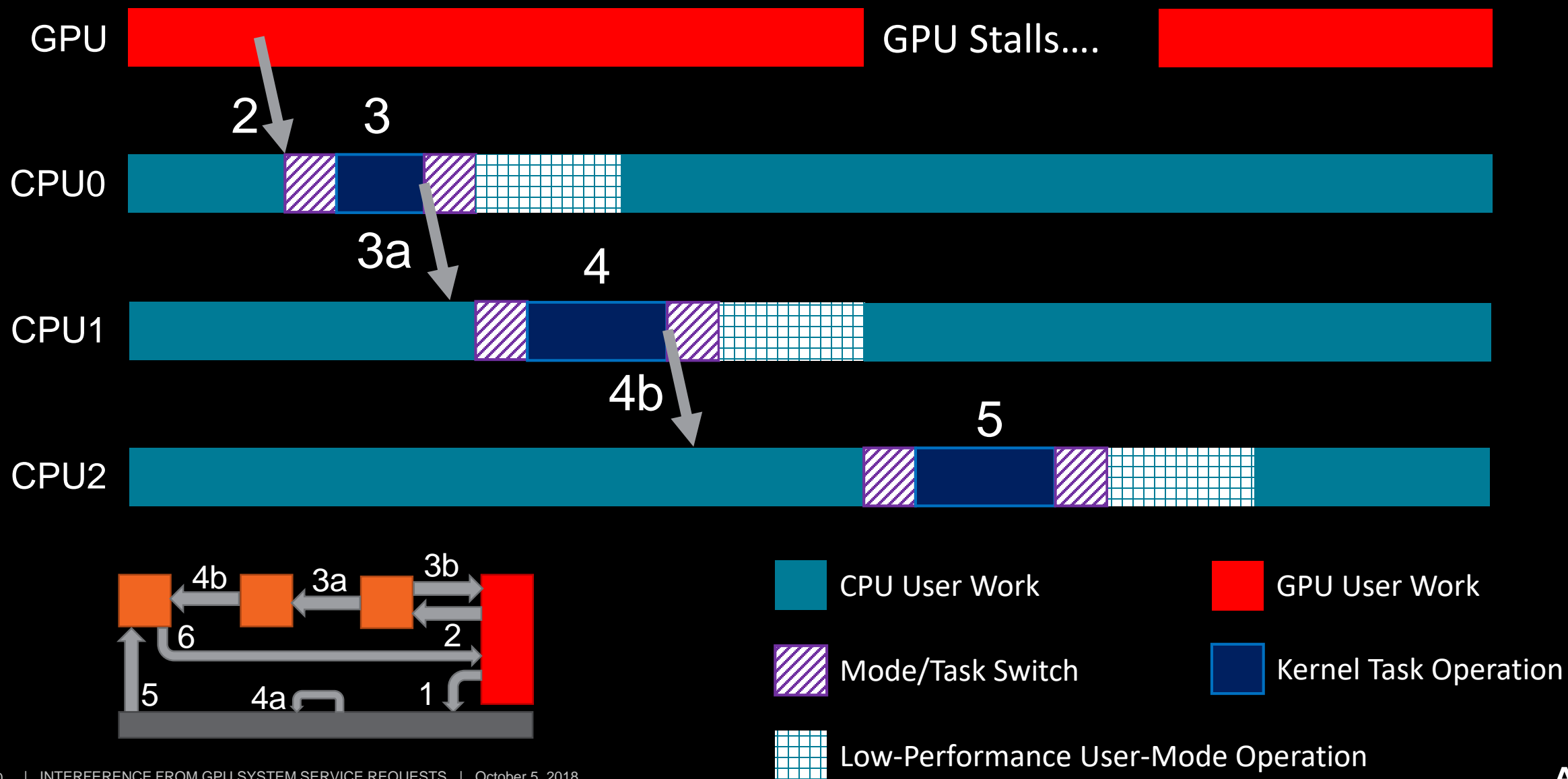
MODES OF INTERFERENCE FROM GPU SSRS



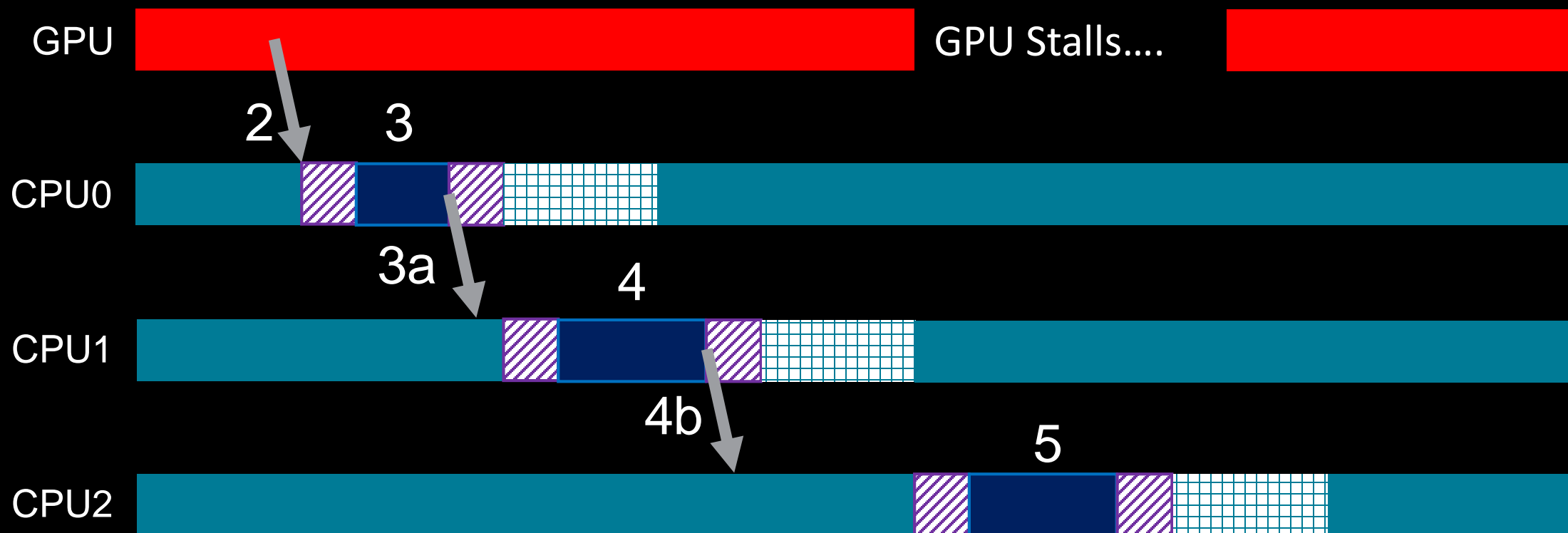
MODES OF INTERFERENCE FROM GPU SSRS



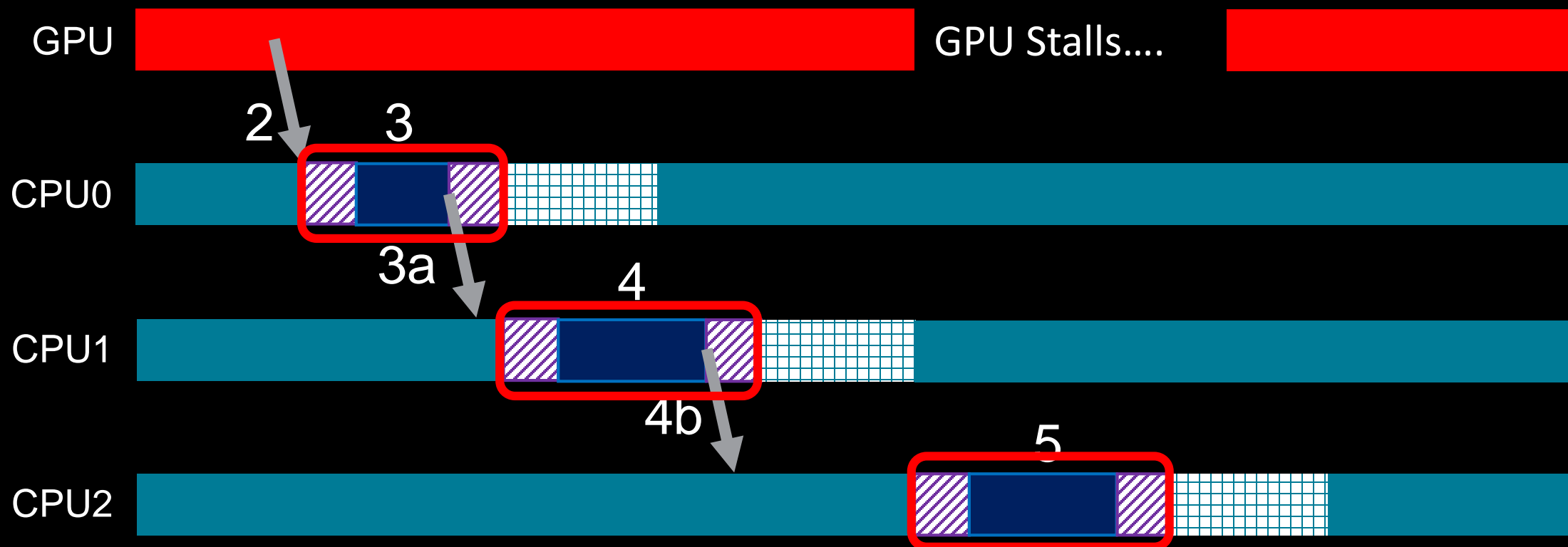
MODES OF INTERFERENCE FROM GPU SSRS



MODES OF INTERFERENCE FROM GPU SSRS

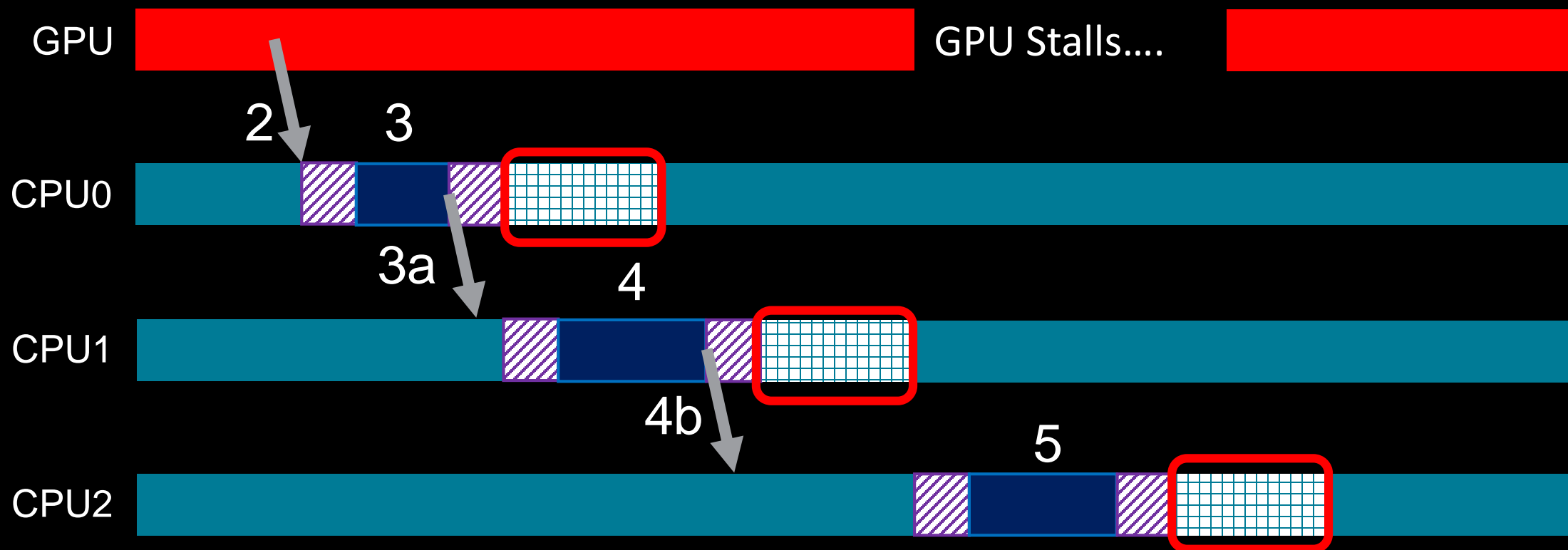


MODES OF INTERFERENCE FROM GPU SSRS



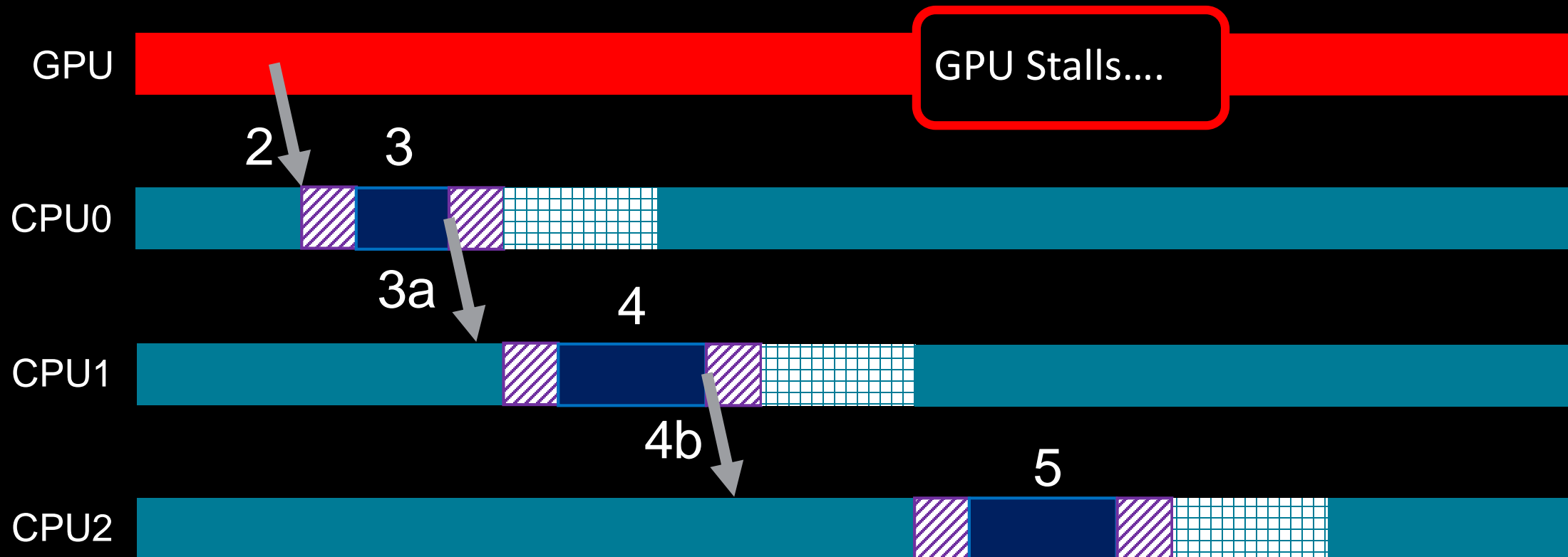
- Direct CPU Overheads

MODES OF INTERFERENCE FROM GPU SSRS



- Direct CPU Overheads
- Indirect CPU Overheads

MODES OF INTERFERENCE FROM GPU SSRS



- Direct CPU Overheads
- Indirect CPU Overheads
- GPU Overheads

TEST PLATFORM USED TO DEMONSTRATE THIS

TEST PLATFORM USED TO DEMONSTRATE THIS

- AMD A10-7850K APU
 - 4x 3.7 GHz CPU cores. AMD Family 15h Model 30h
 - 720 MHz AMD GCN 1.1 (“Sea Islands”, gfx700) GPU, 8 CUs
 - 32 GB Dual-Channel DDR3-1866

TEST PLATFORM USED TO DEMONSTRATE THIS

- AMD A10-7850K APU
 - 4x 3.7 GHz CPU cores. AMD Family 15h Model 30h
 - 720 MHz AMD GCN 1.1 (“Sea Islands”, gfx700) GPU, 8 CUs
 - 32 GB Dual-Channel DDR3-1866
- Ubuntu 14.04.3 LTS (AMD64)
 - Linux[®] kernel 4.0.0 with HSA Drivers (amdkfd 1.6.1)

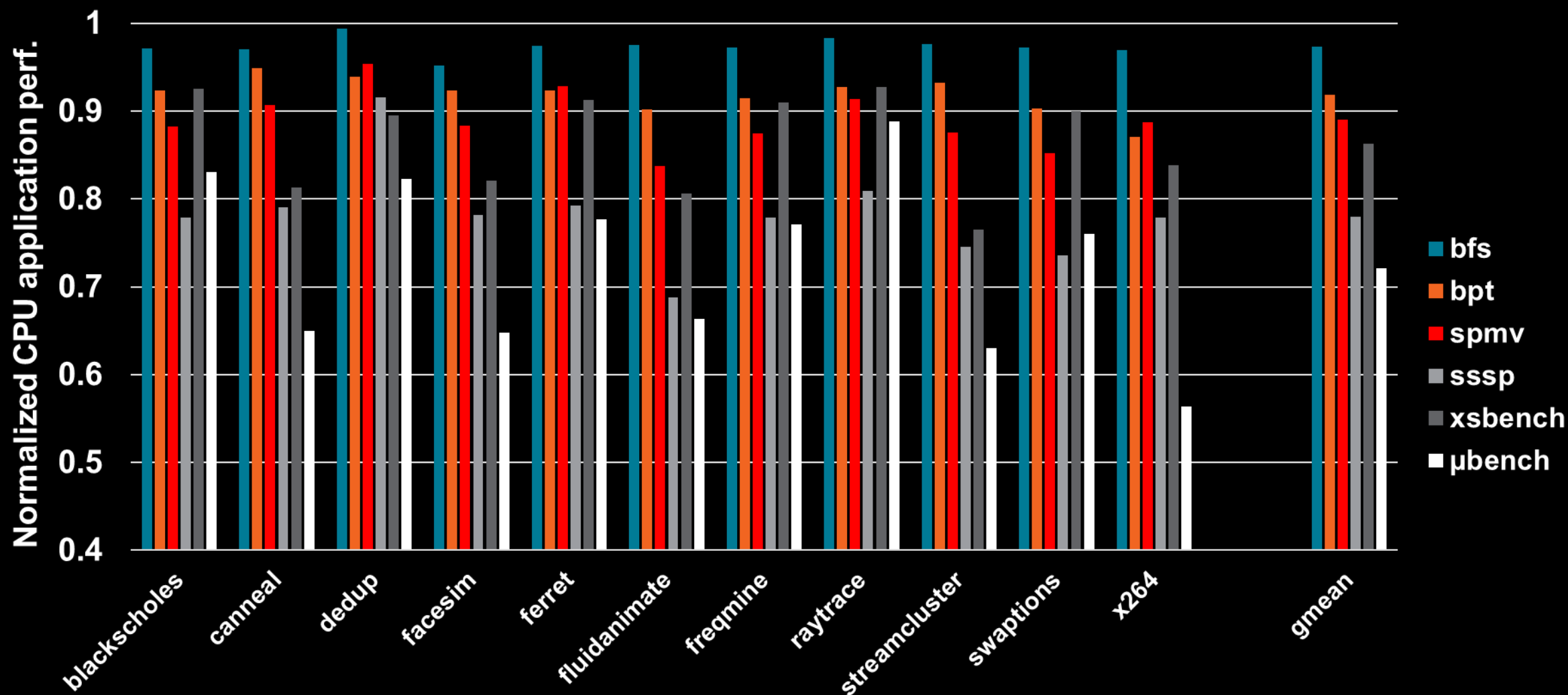
TEST PLATFORM USED TO DEMONSTRATE THIS

- AMD A10-7850K APU
 - 4x 3.7 GHz CPU cores. AMD Family 15h Model 30h
 - 720 MHz AMD GCN 1.1 (“Sea Islands”, gfx700) GPU, 8 CUs
 - 32 GB Dual-Channel DDR3-1866
- Ubuntu 14.04.3 LTS (AMD64)
 - Linux® kernel 4.0.0 with HSA Drivers (amdkfd 1.6.1)
- PARSEC benchmarks for “CPU work”

TEST PLATFORM USED TO DEMONSTRATE THIS

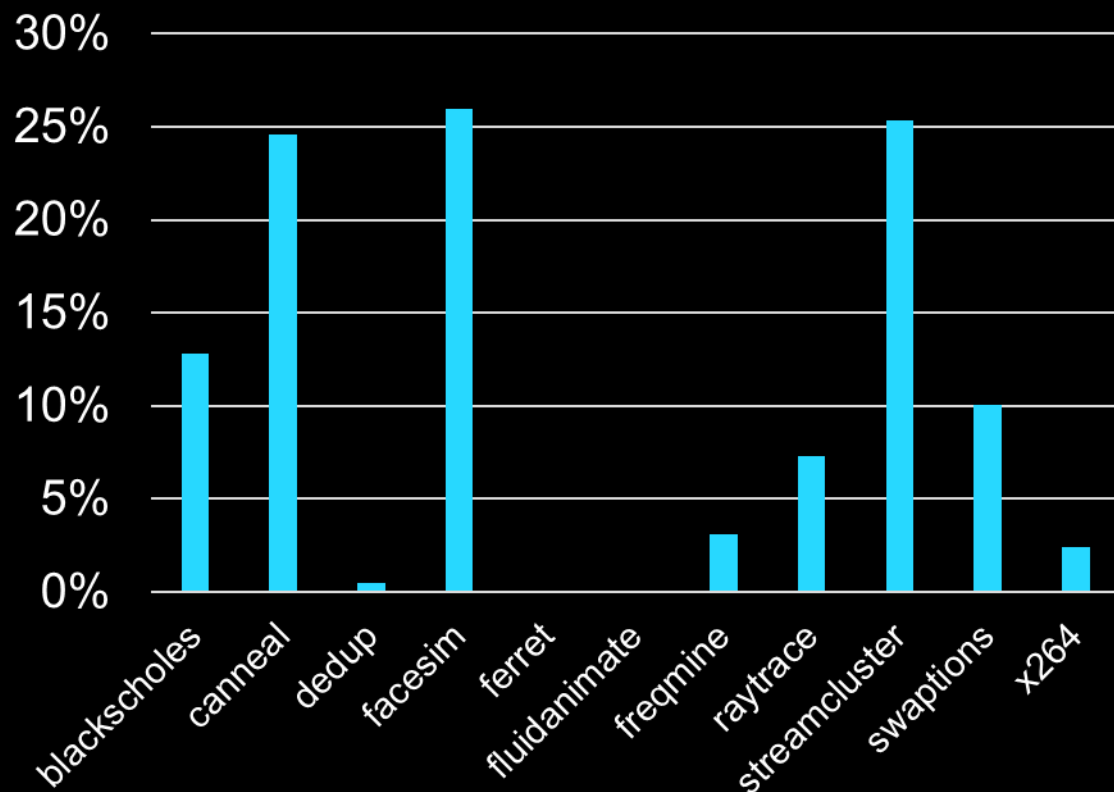
- AMD A10-7850K APU
 - 4x 3.7 GHz CPU cores. AMD Family 15h Model 30h
 - 720 MHz AMD GCN 1.1 (“Sea Islands”, gfx700) GPU, 8 CUs
 - 32 GB Dual-Channel DDR3-1866
- Ubuntu 14.04.3 LTS (AMD64)
 - Linux[®] kernel 4.0.0 with HSA Drivers (amdkfd 1.6.1)
- PARSEC benchmarks for “CPU work”
- OpenCL[™] benchmark applications modified to create SSRs (page faults)
 - BPT [Vesely et al., ISPASS 2016]
 - XSBench [Vesely et al., ISPASS 2016]
 - SHOC BFS
 - SHOC SpMV
 - Pannotia SSSP
 - μ Benchmark

CPU PERFORMANCE W/ SSR-USING GPU APPLICATIONS

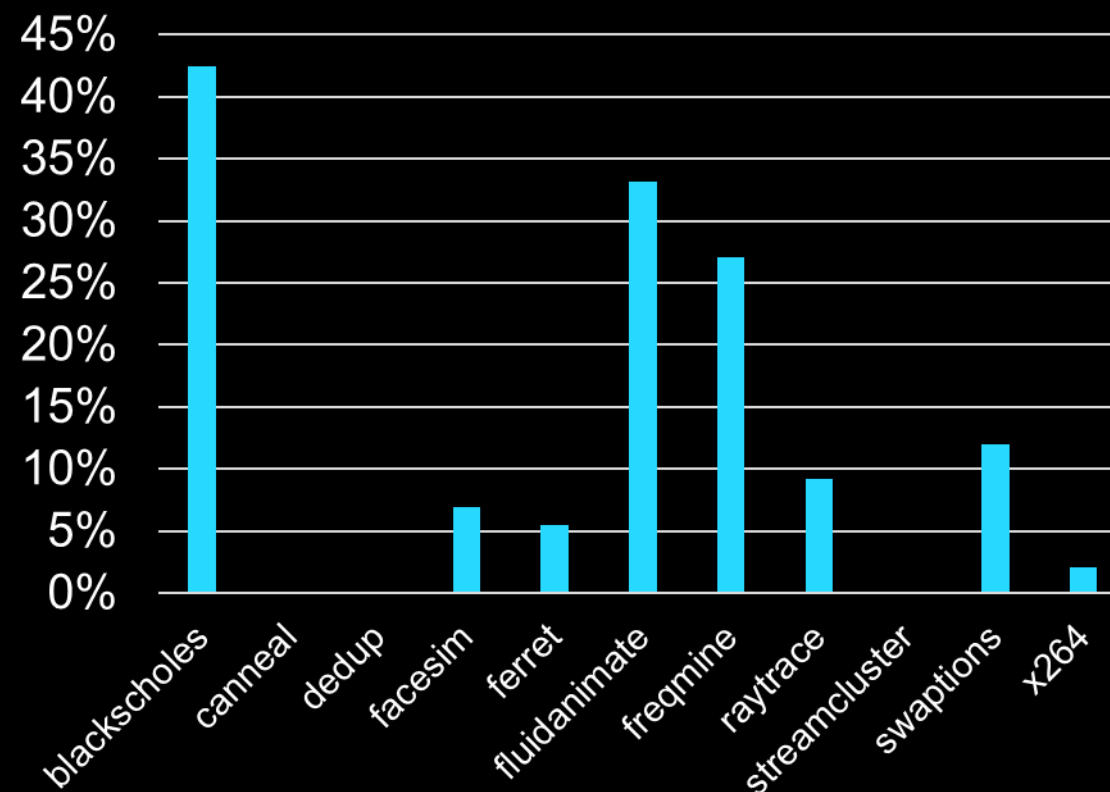


INDIRECT CPU OVERHEADS FROM GPU SSRS

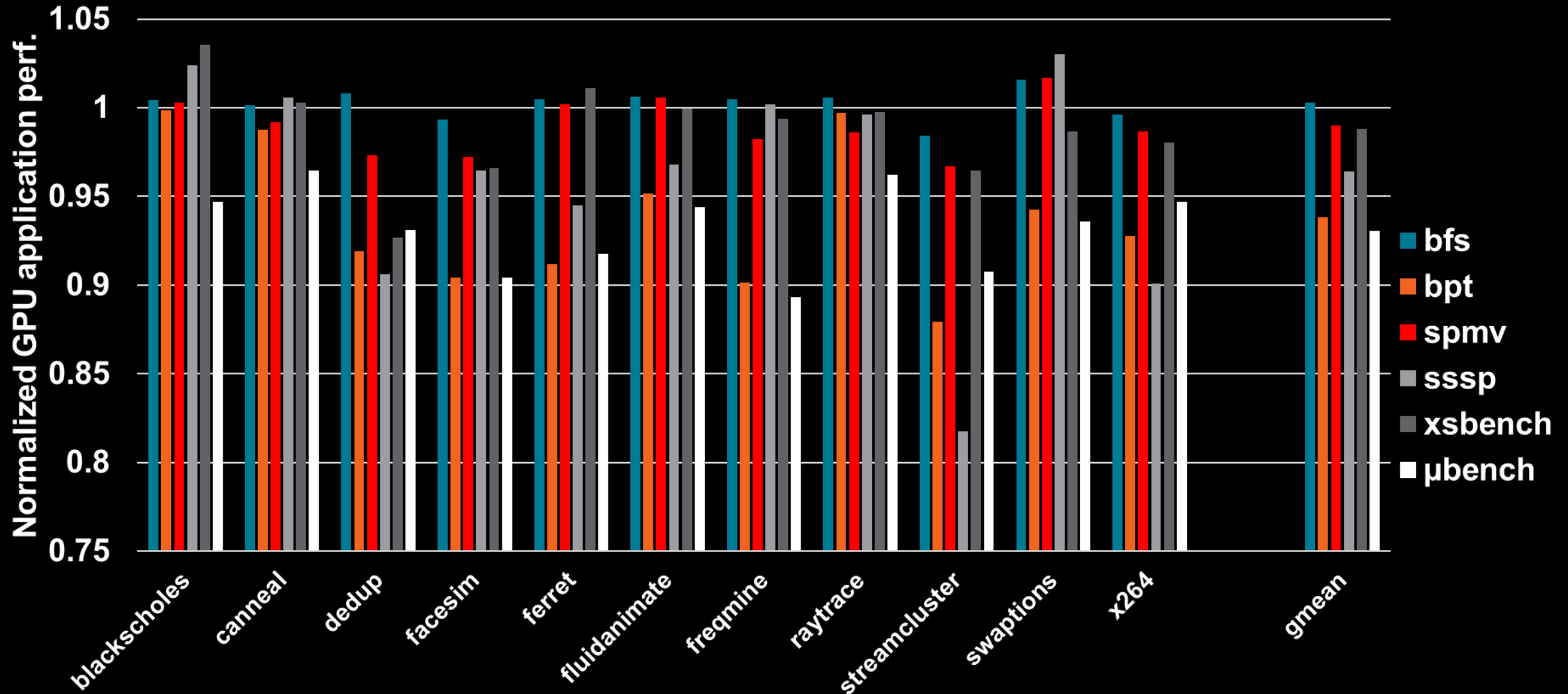
Increase of User-Level
Branch Mispredict Rate



Increase of User-Level
L1 Cache Miss Rate

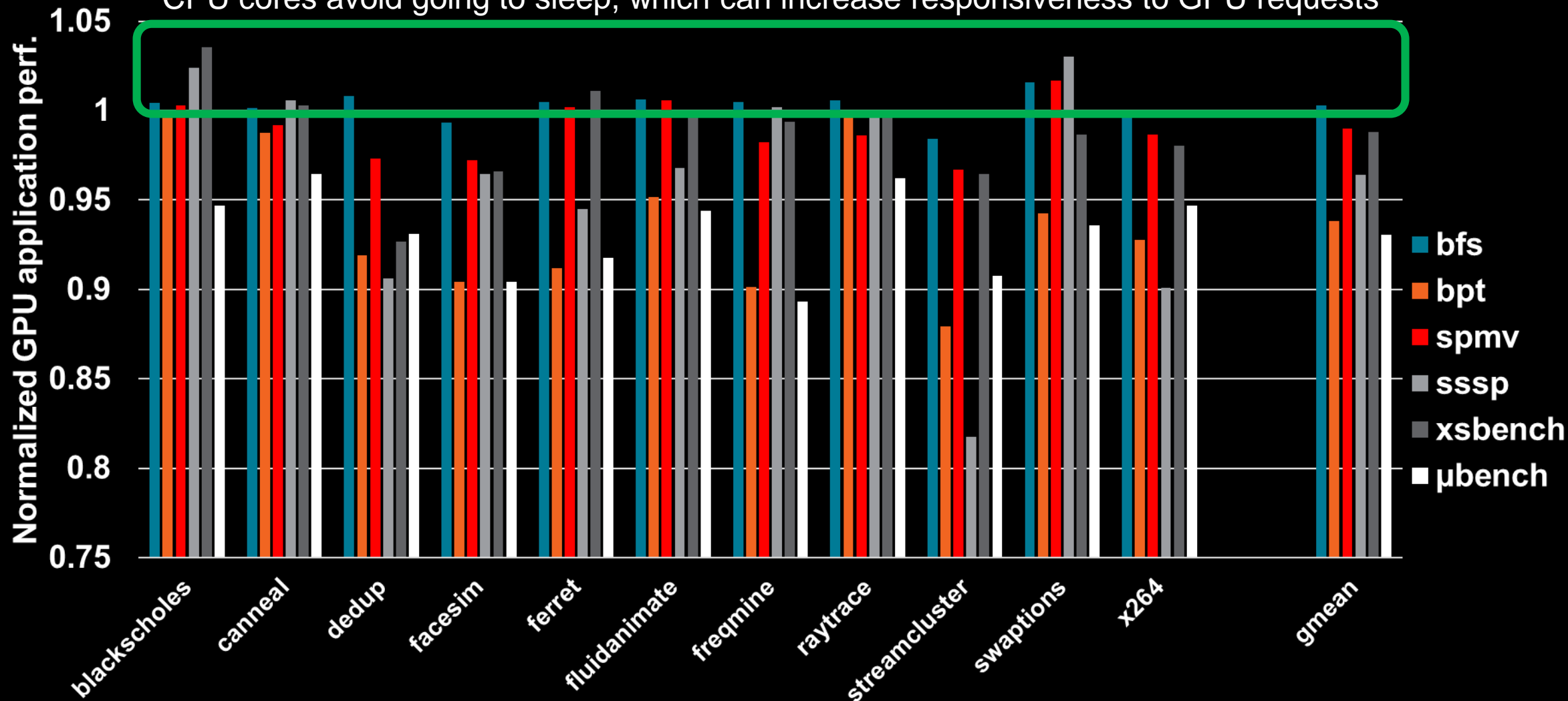


GPU PERFORMANCE W/ CONCURRENT CPU APPS



GPU PERFORMANCE W/ CONCURRENT CPU APPS

CPU cores avoid going to sleep, which can increase responsiveness to GPU requests



Mitigation Strategies

TAKE INSPIRATION FROM OTHER DOMAINS
(E.G. HIGH PERFORMANCE NETWORKING)

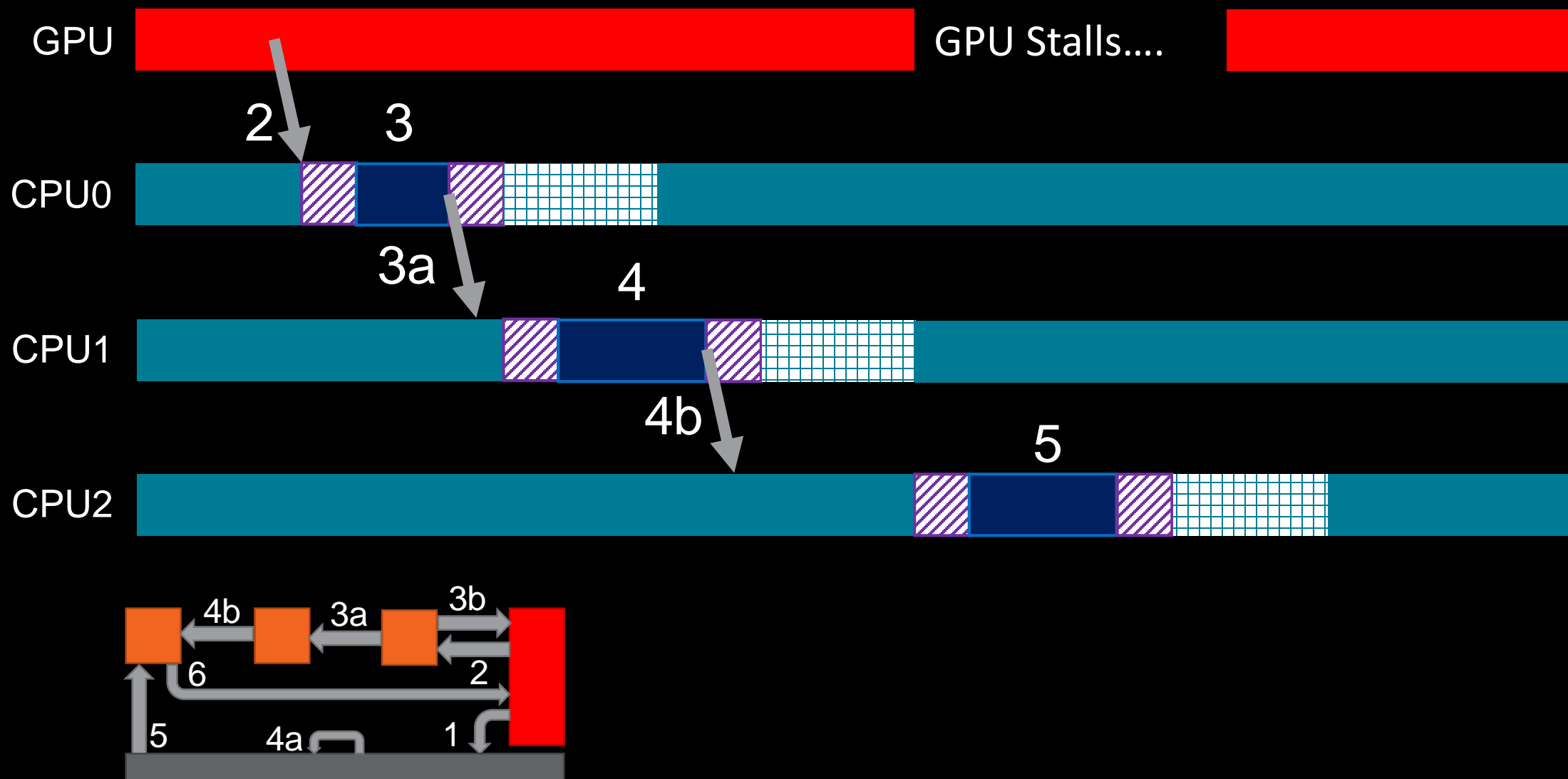
Interrupt Coalescing

Interrupt Steering

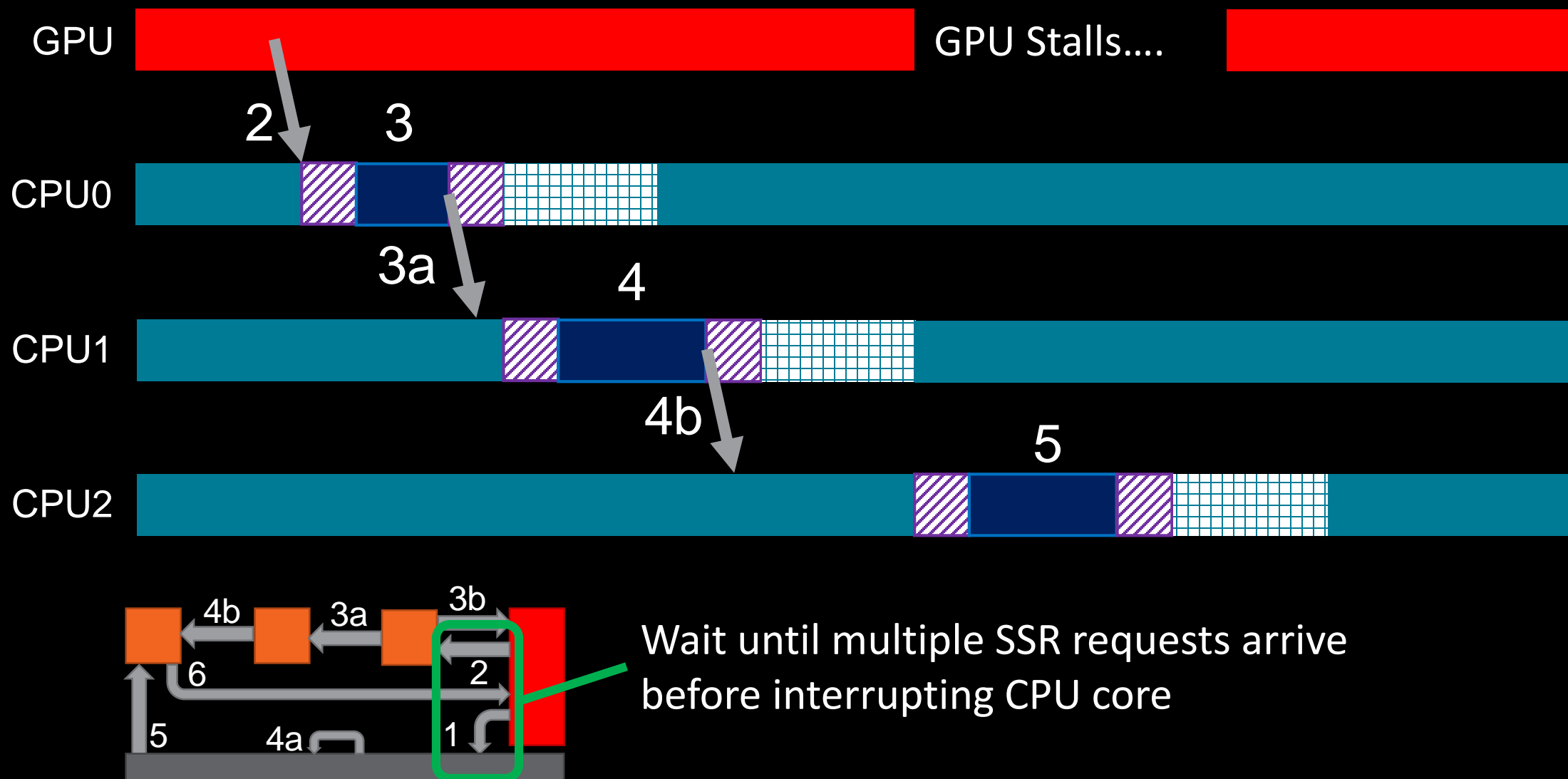
Merged SSR Handlers

Driver for Enforcing CPU QoS

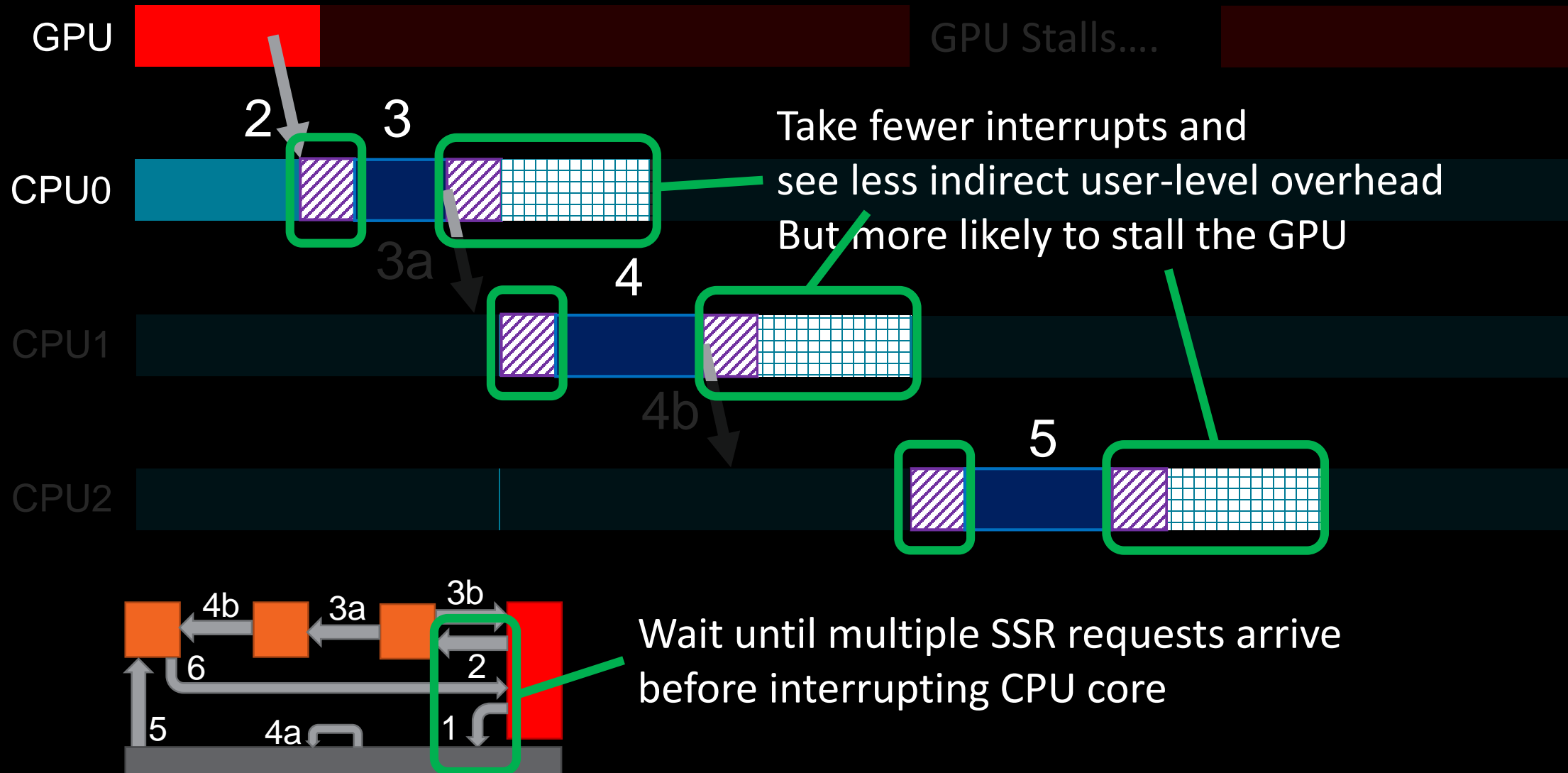
MITIGATION: INTERRUPT COALESCING



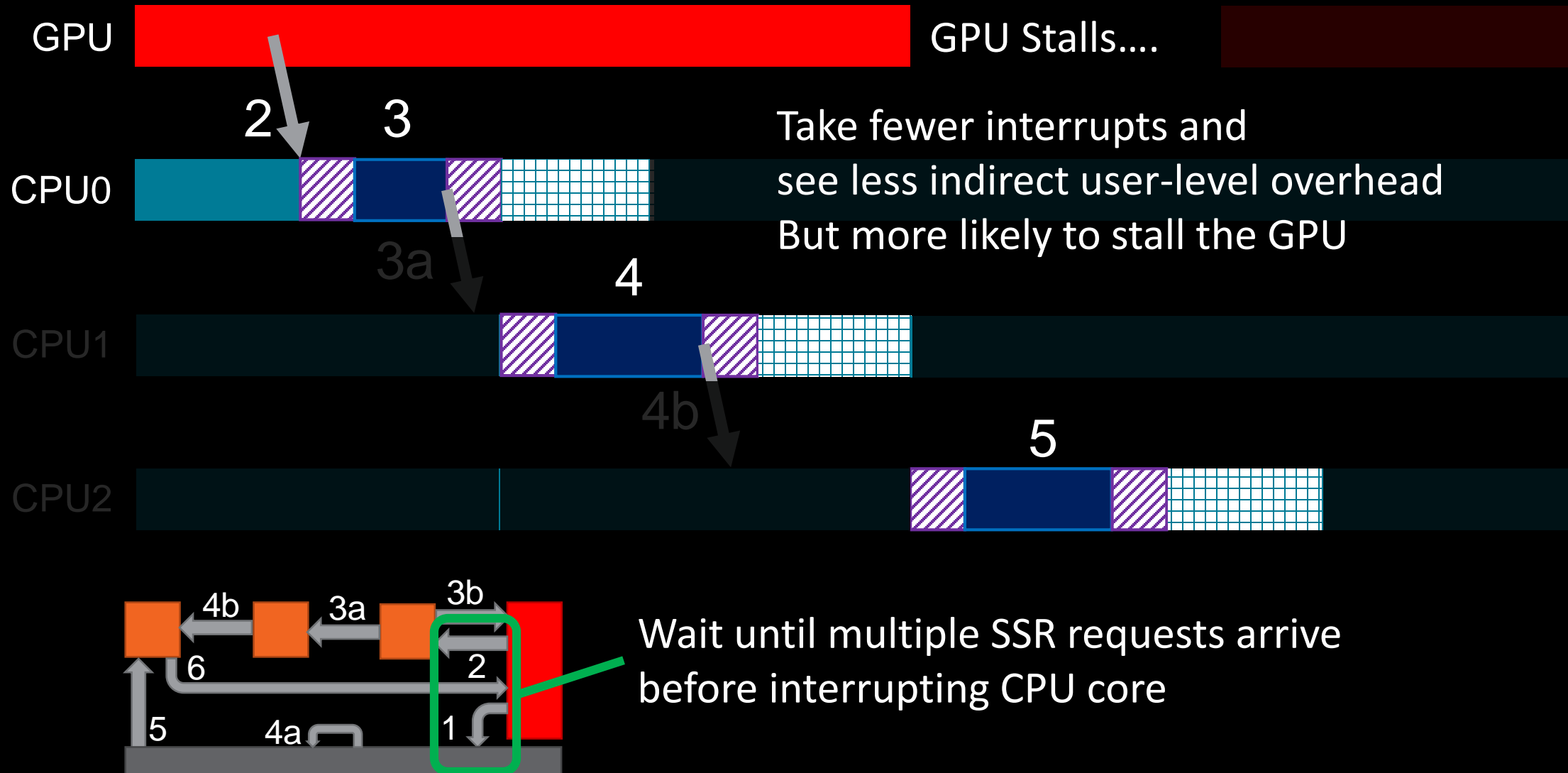
MITIGATION: INTERRUPT COALESCING



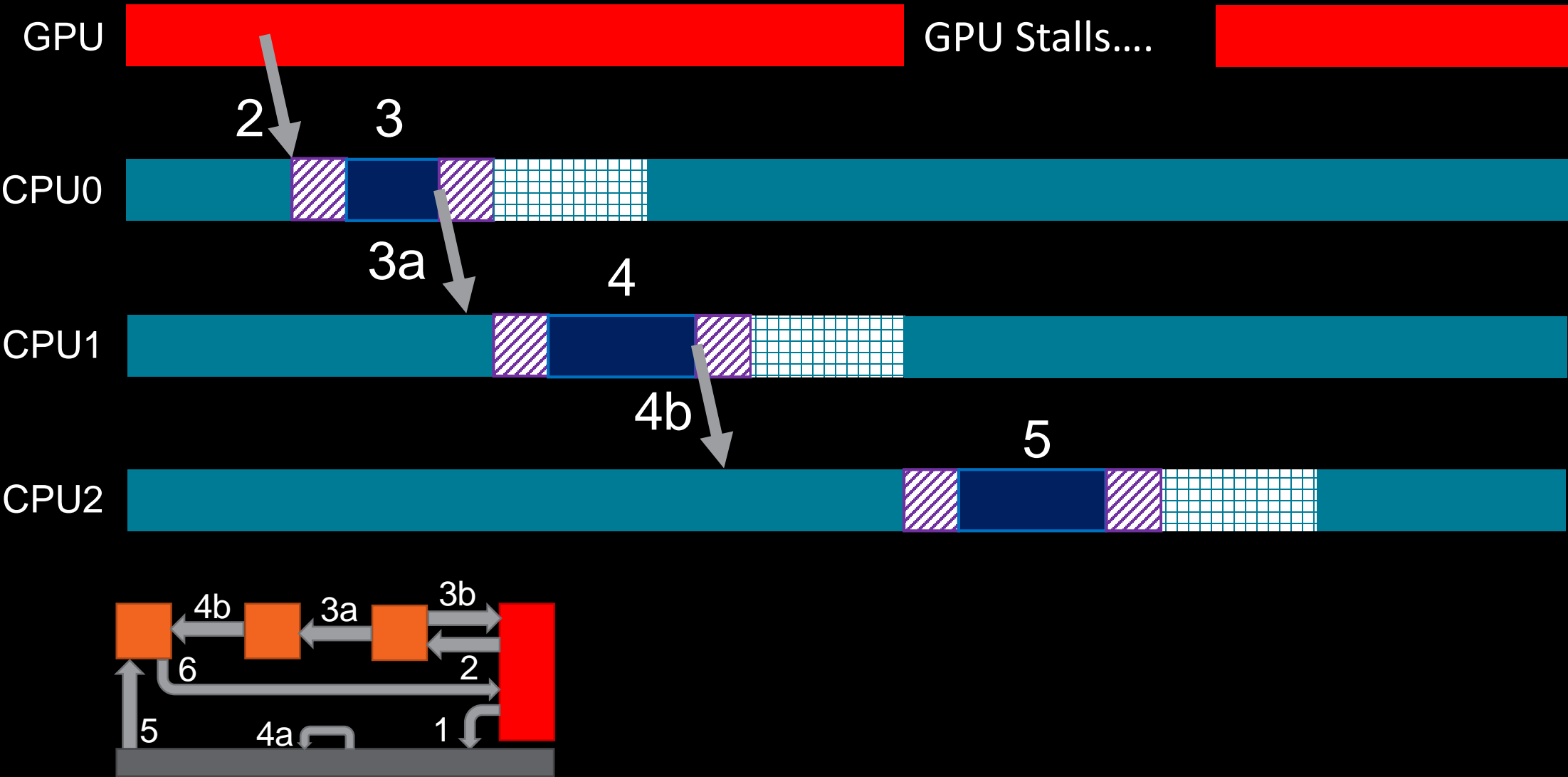
MITIGATION: INTERRUPT COALESCING



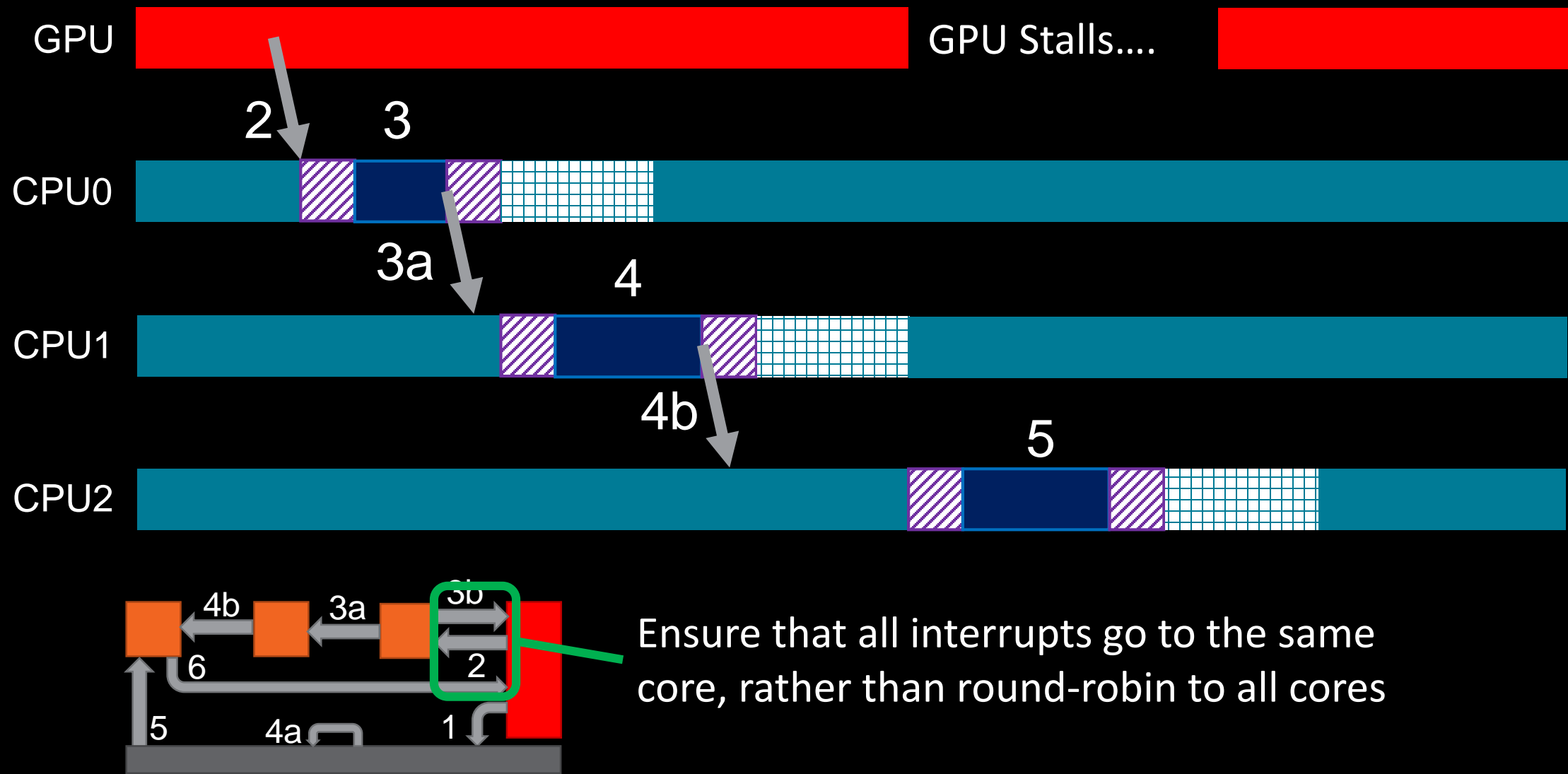
MITIGATION: INTERRUPT COALESCING



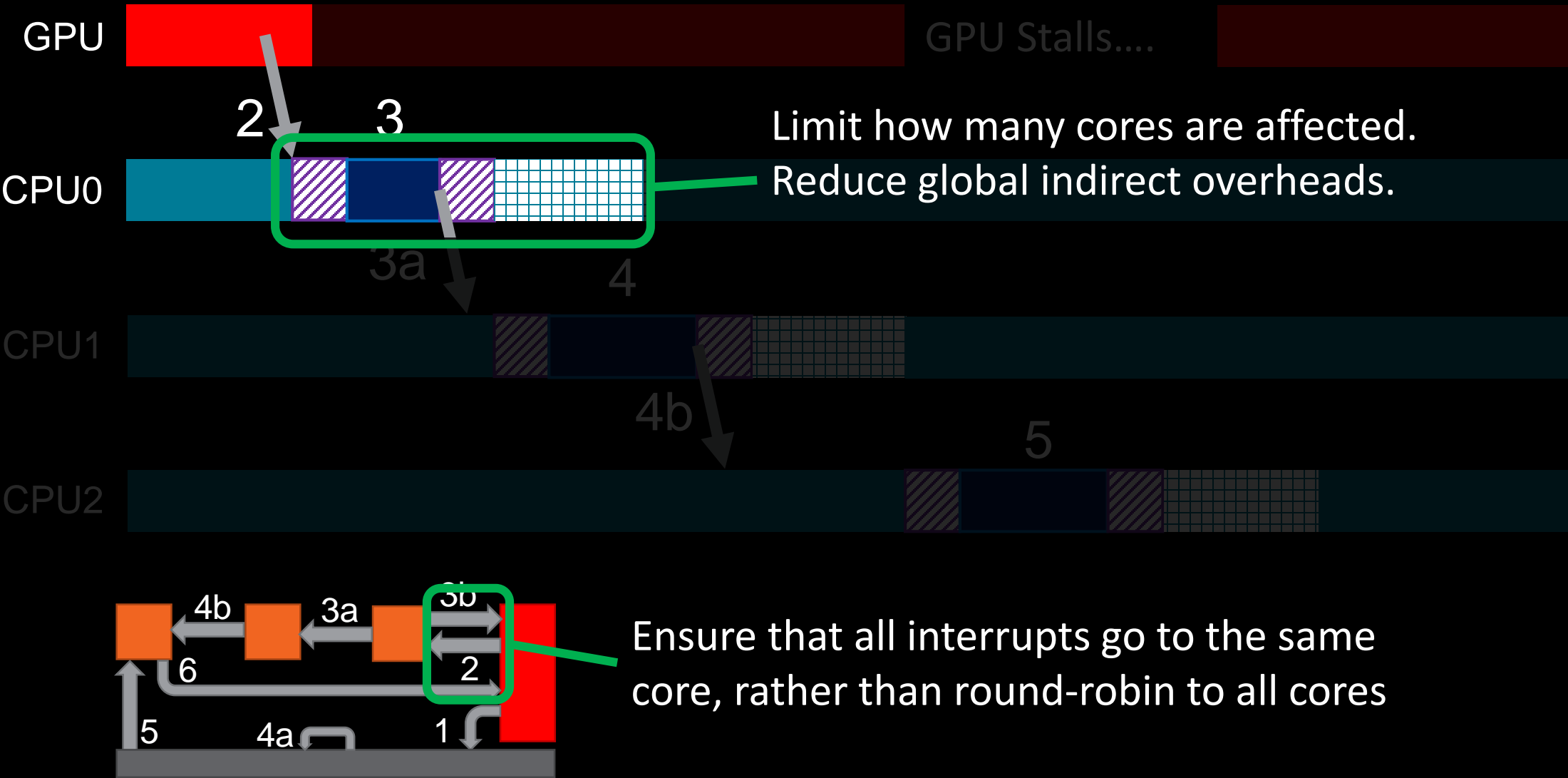
MITIGATION: INTERRUPT STEERING



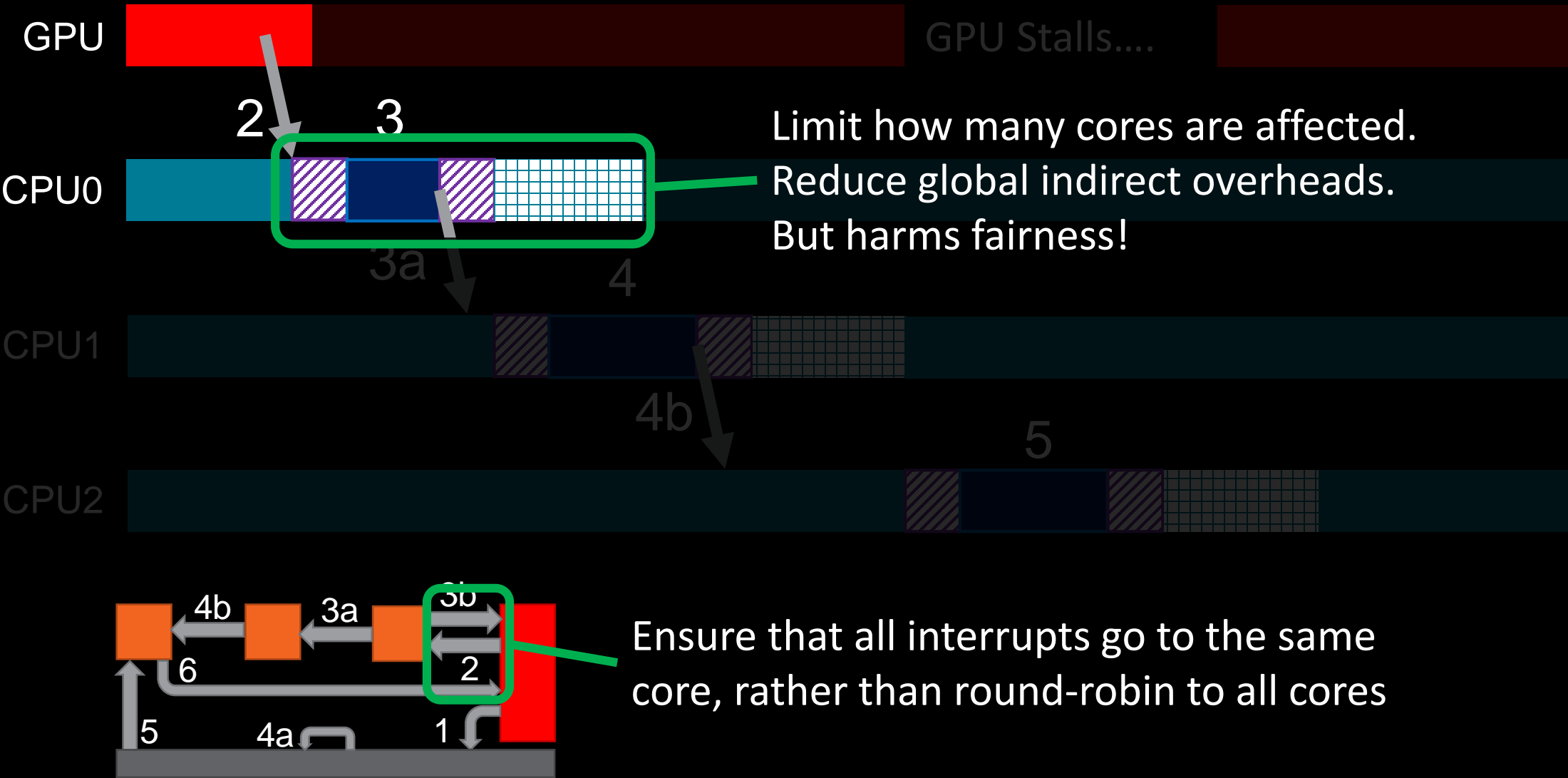
MITIGATION: INTERRUPT STEERING



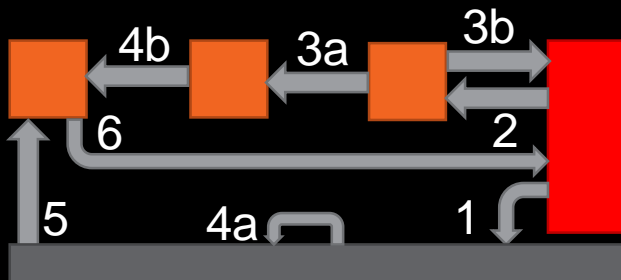
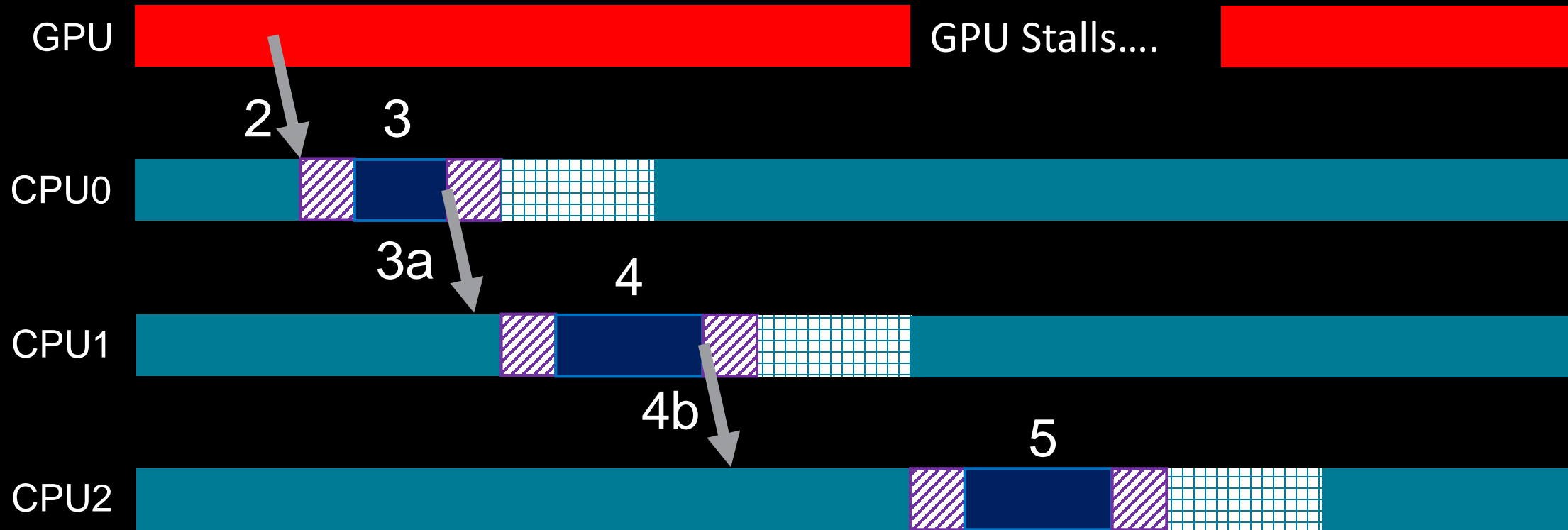
MITIGATION: INTERRUPT STEERING



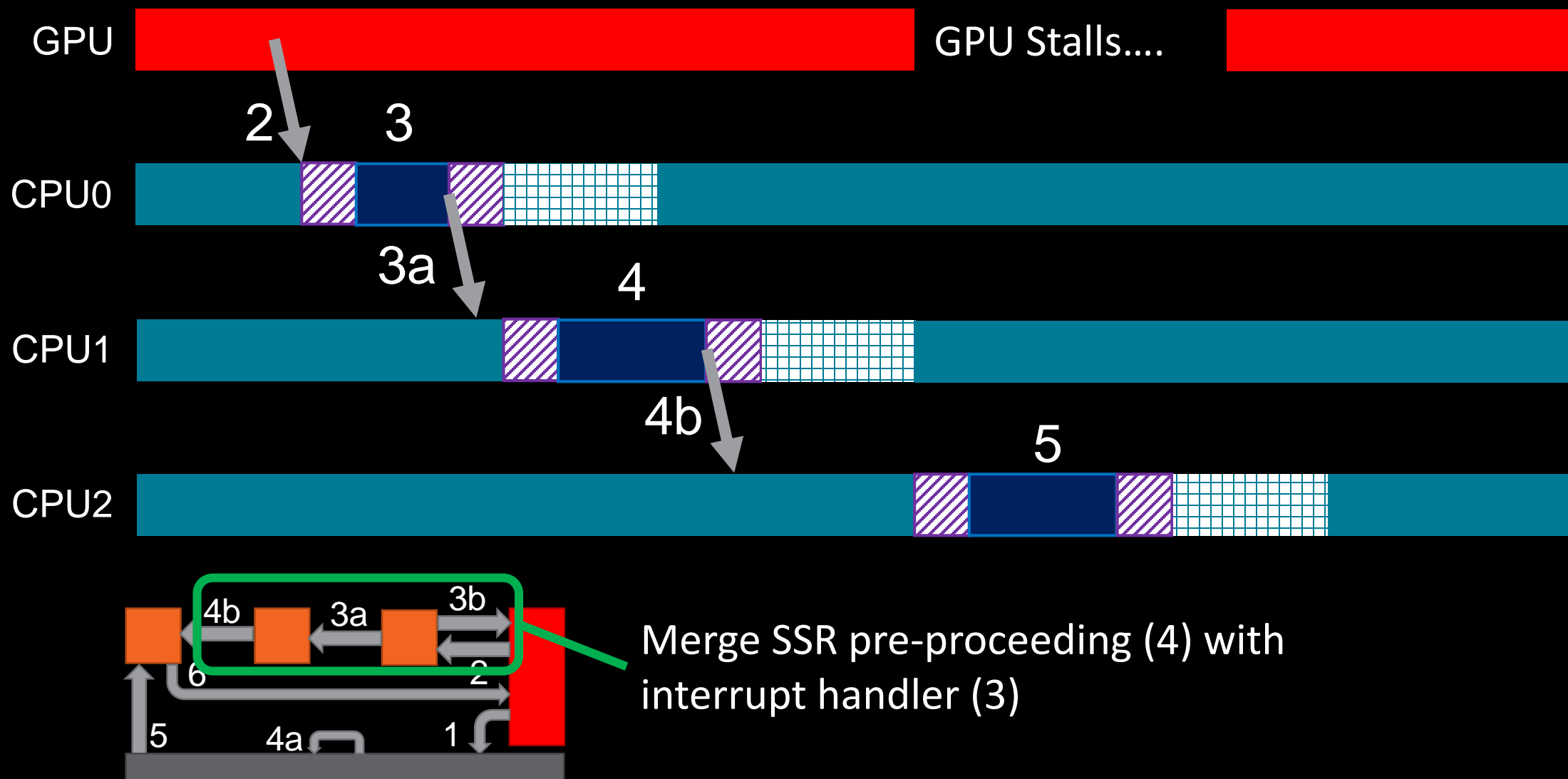
MITIGATION: INTERRUPT STEERING



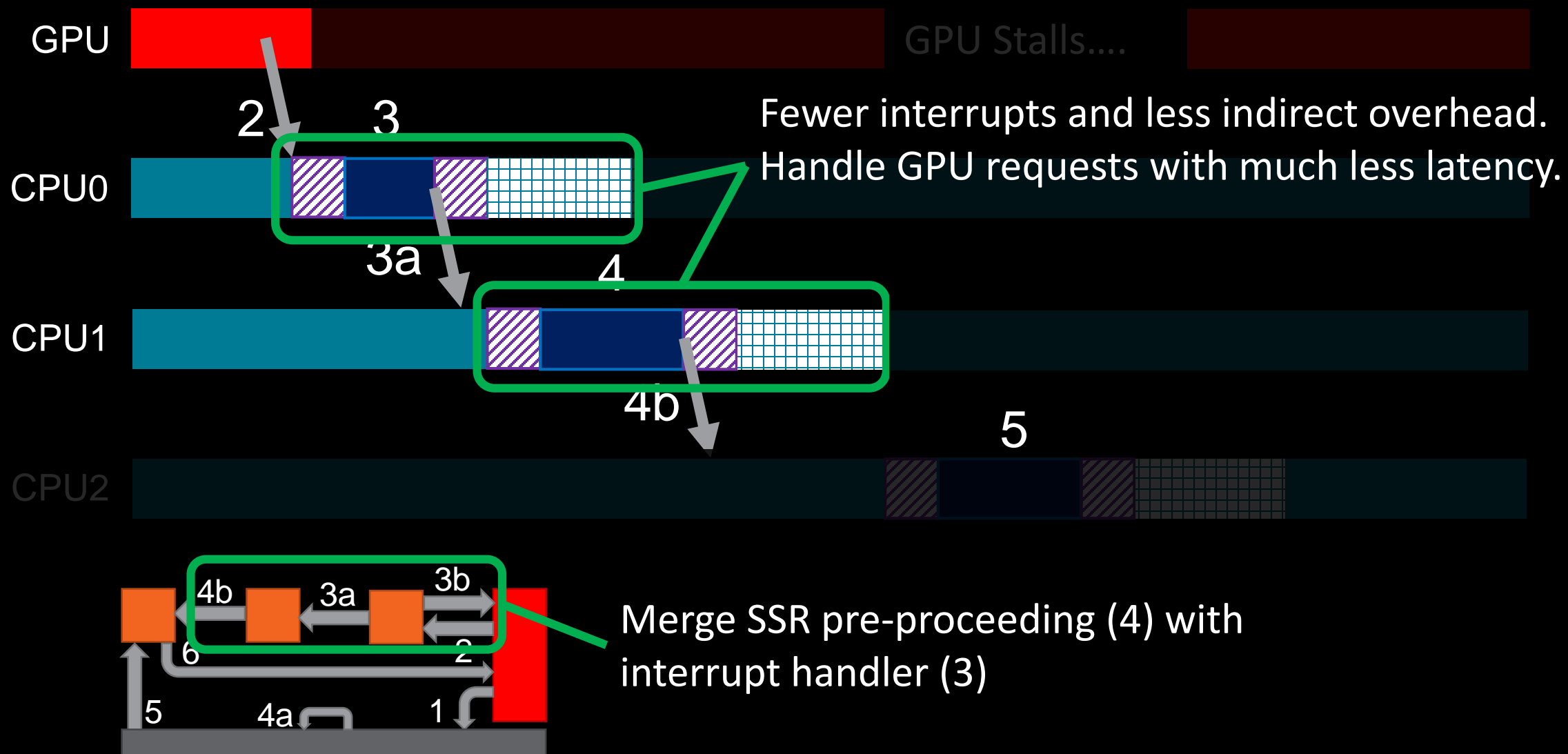
MITIGATION: MERGED SSR HANDLER



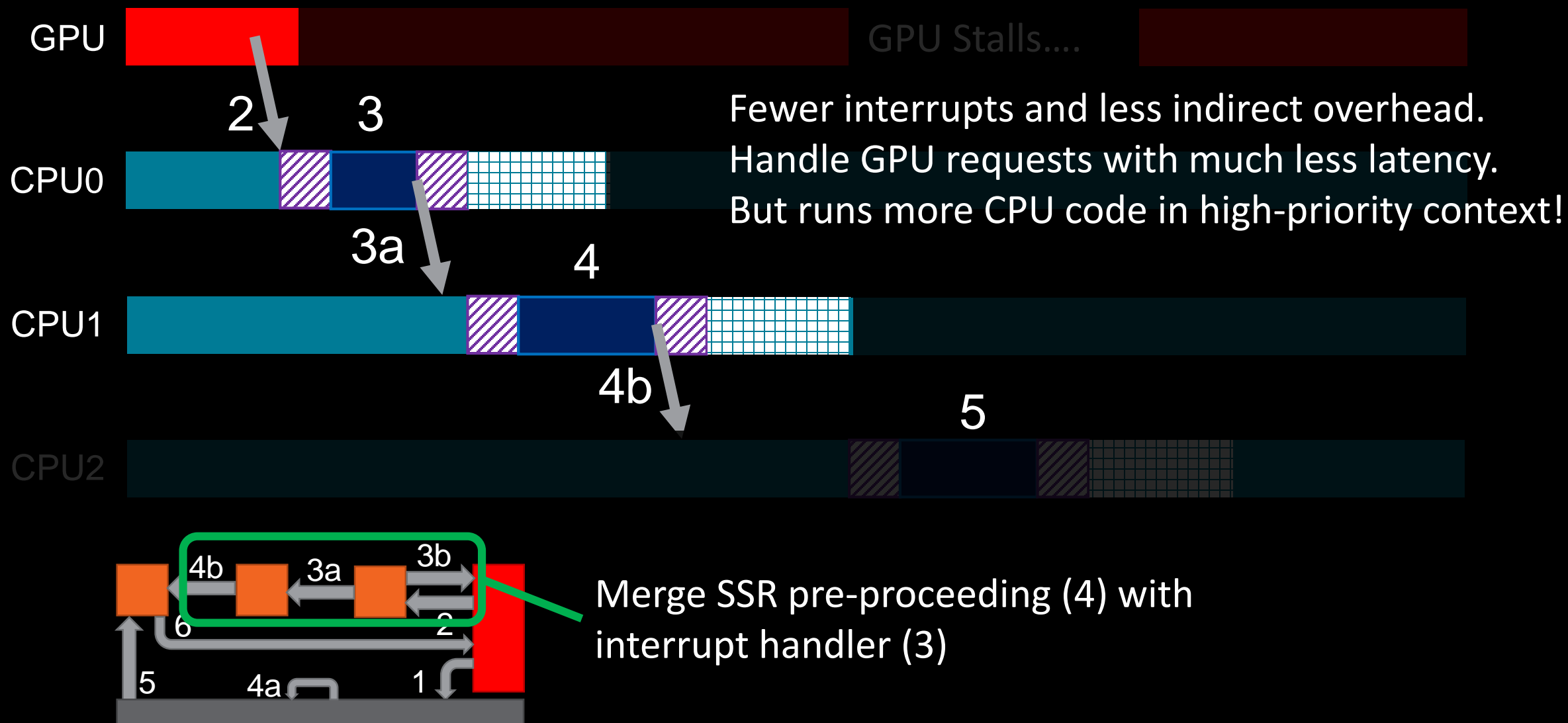
MITIGATION: MERGED SSR HANDLER



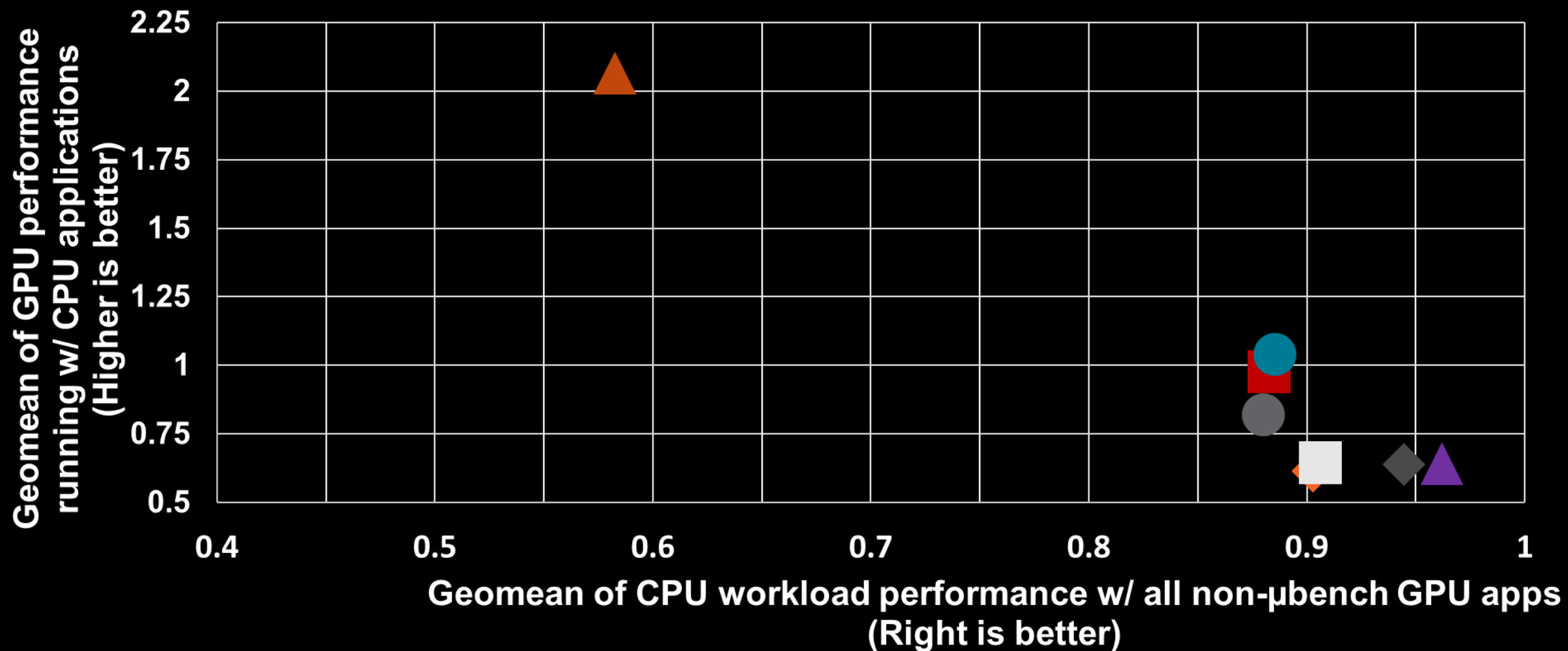
MITIGATION: MERGED SSR HANDLER



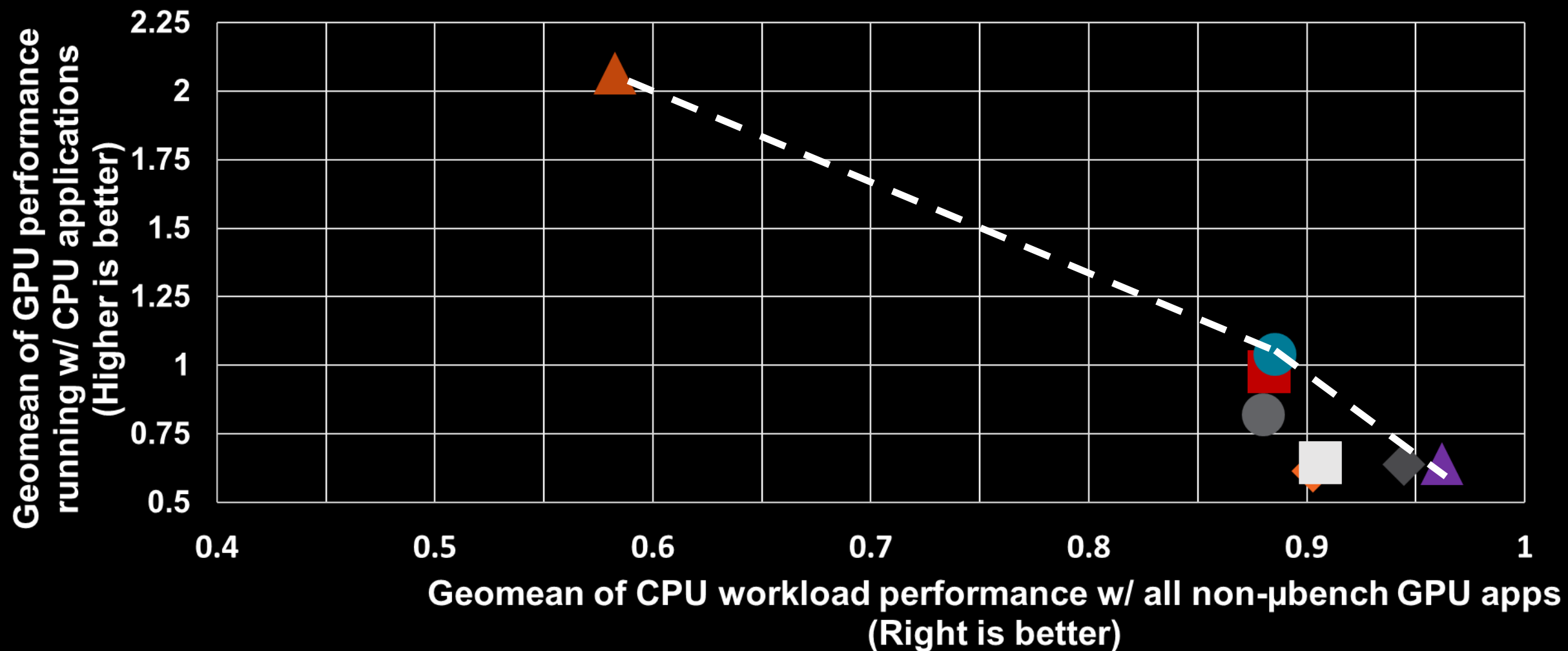
MITIGATION: MERGED SSR HANDLER



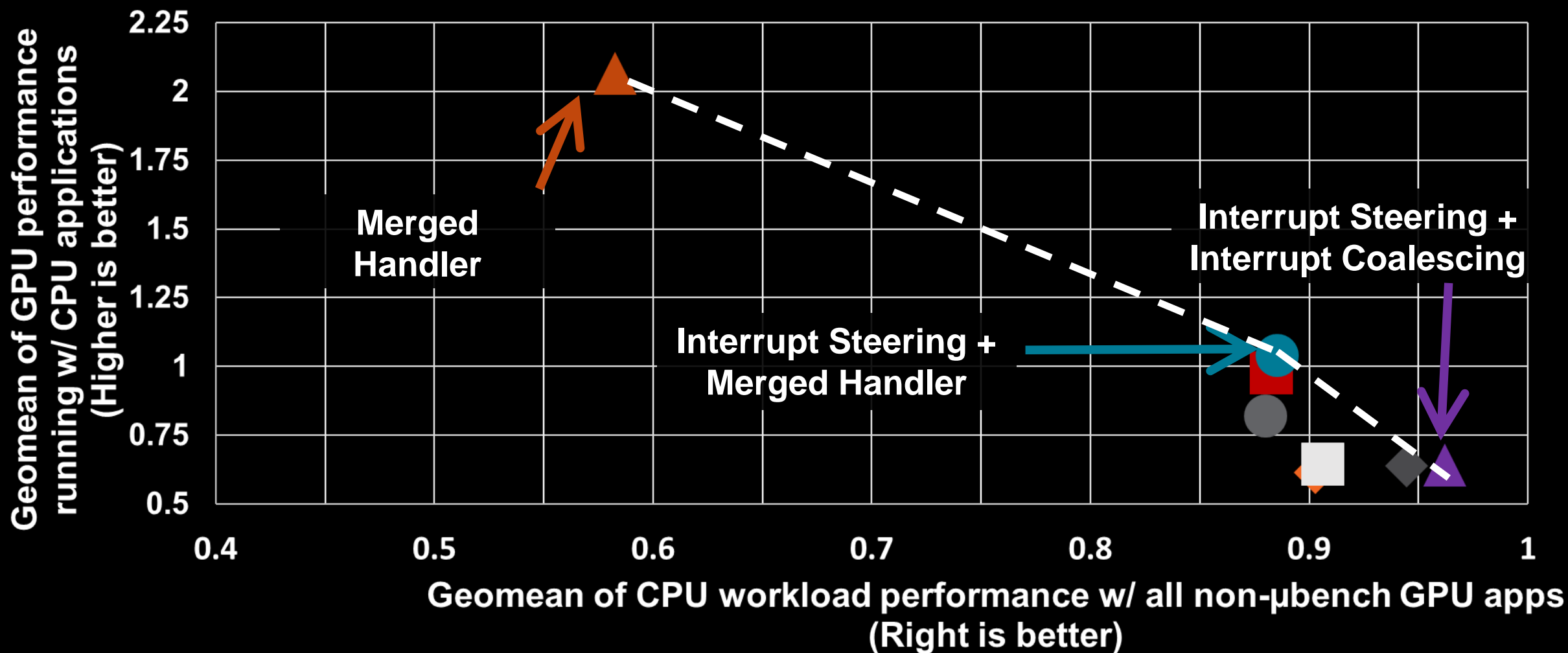
PARETO CURVE OF MITIGATION STRATEGY TRADEOFFS



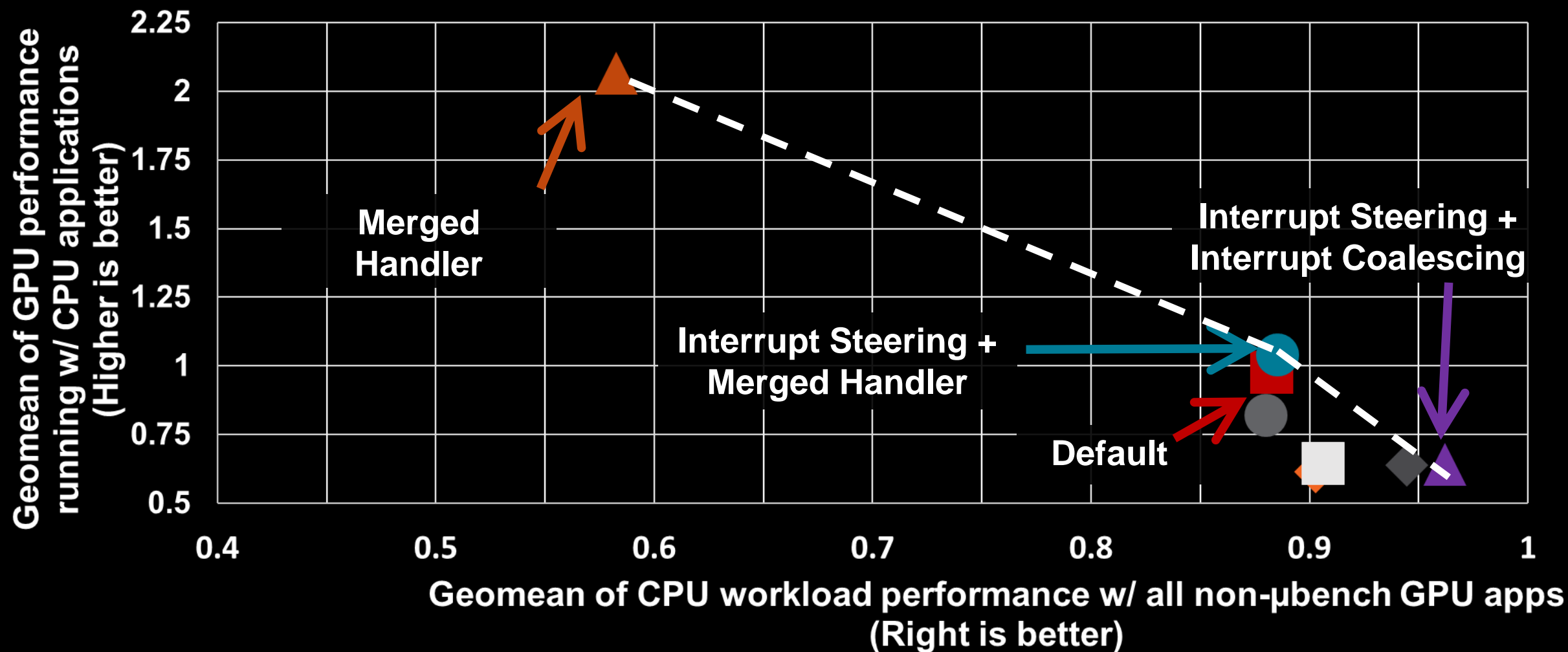
PARETO CURVE OF MITIGATION STRATEGY TRADEOFFS



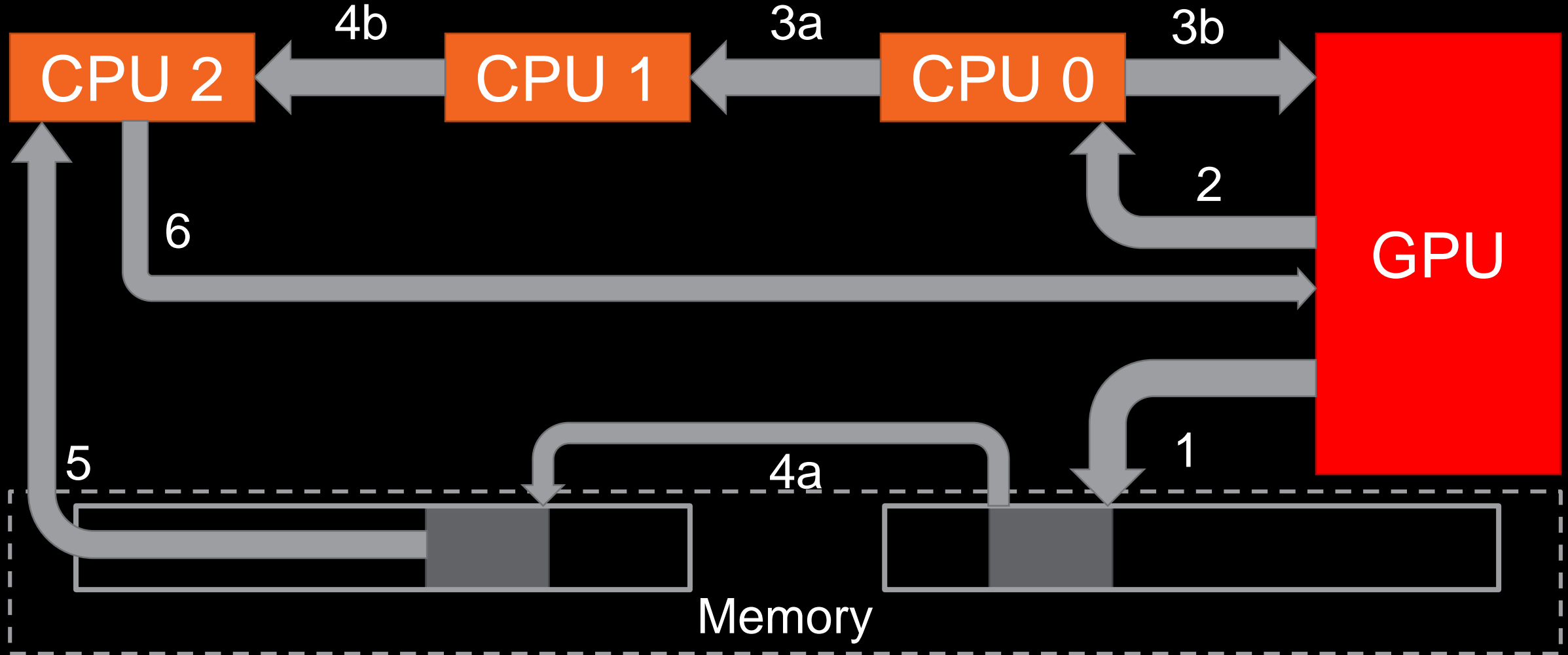
PARETO CURVE OF MITIGATION STRATEGY TRADEOFFS



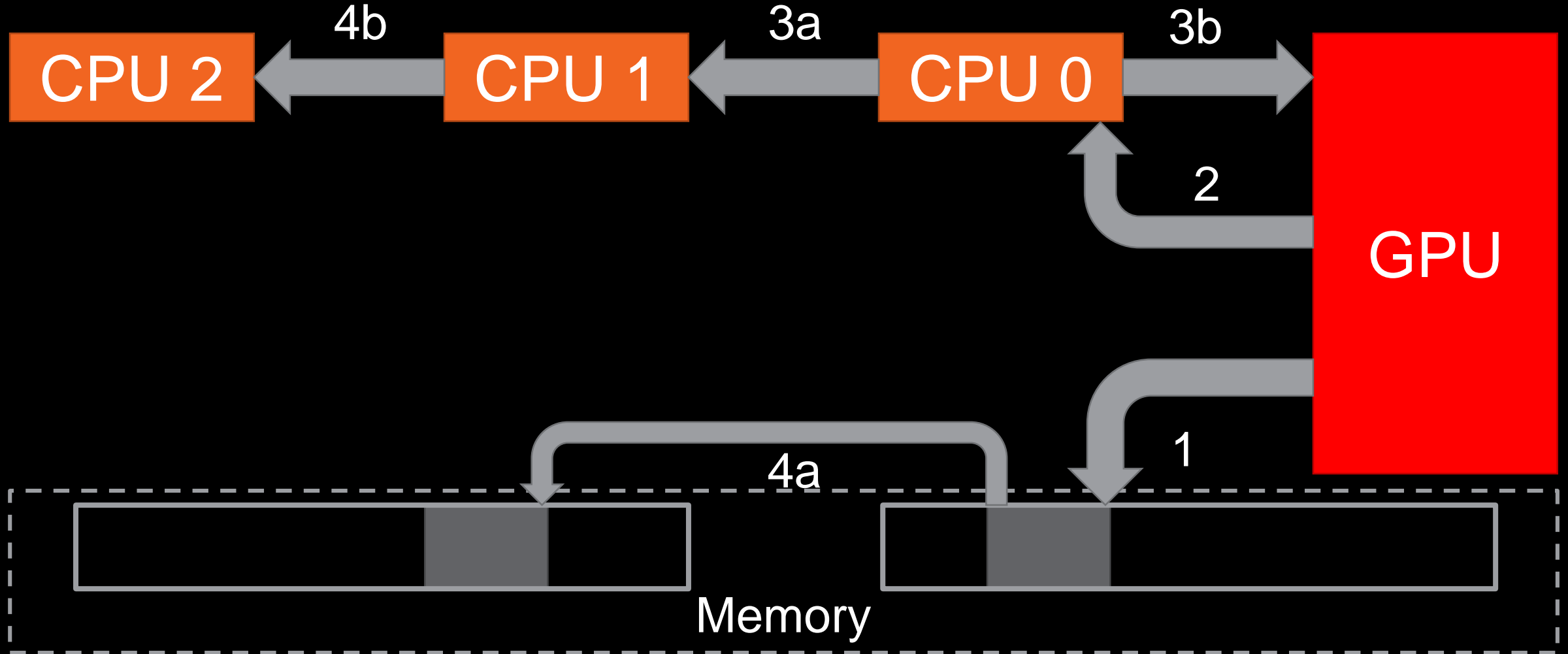
PARETO CURVE OF MITIGATION STRATEGY TRADEOFFS



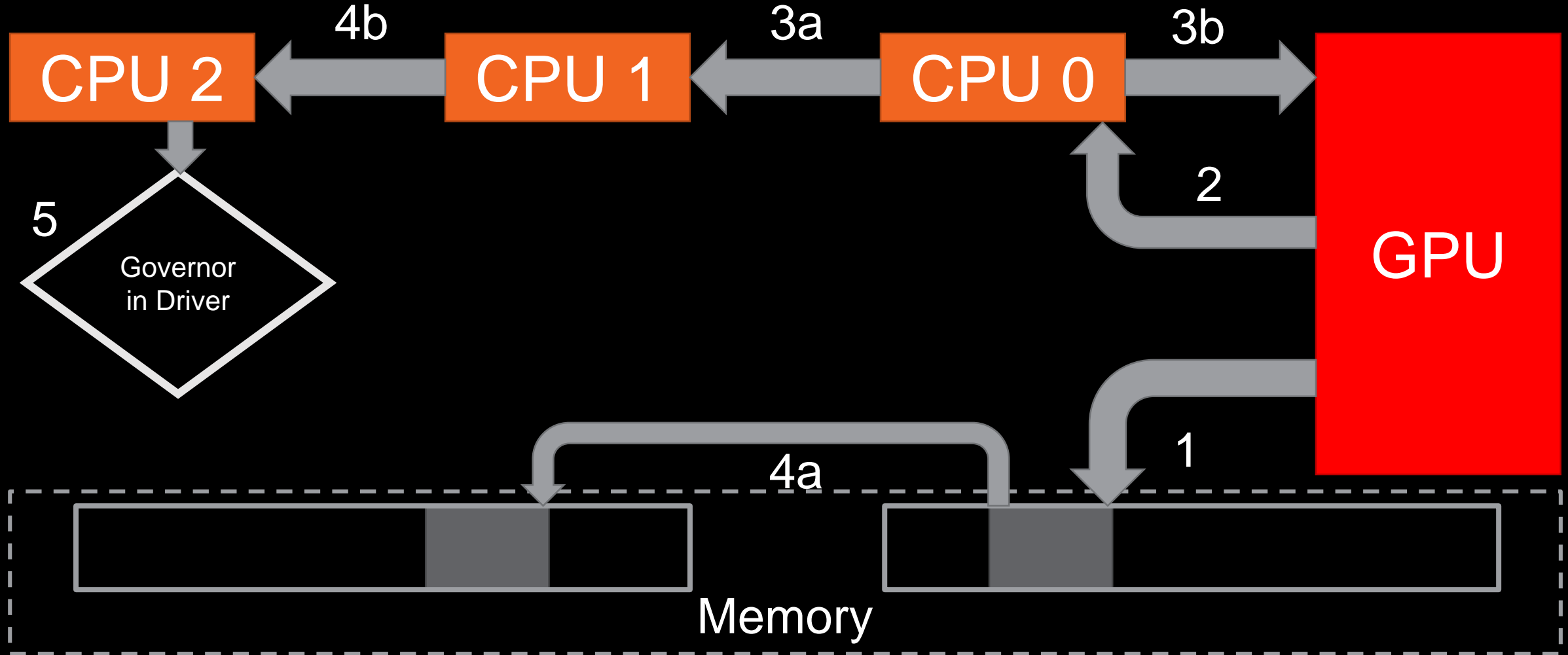
DRIVER MODIFICATIONS TO ENSURE CPU QOS



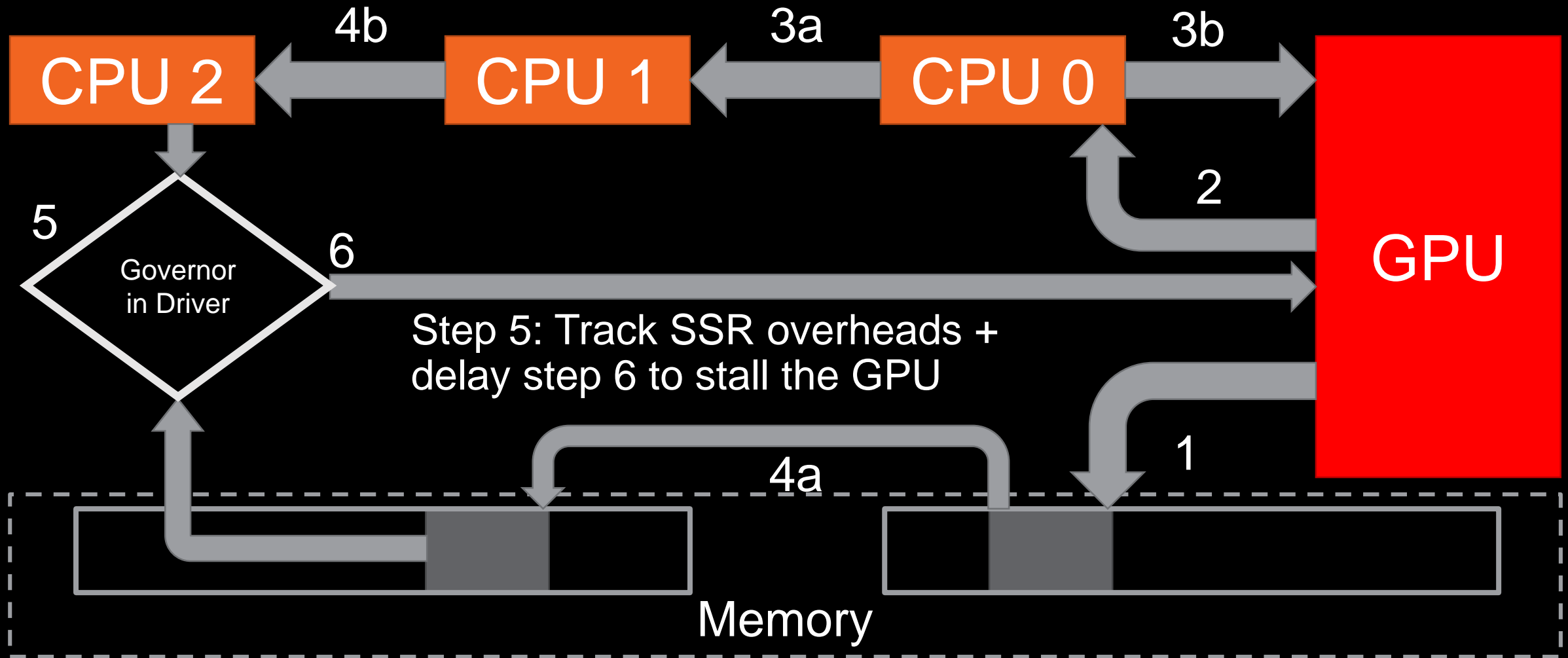
DRIVER MODIFICATIONS TO ENSURE CPU QOS



DRIVER MODIFICATIONS TO ENSURE CPU QOS



DRIVER MODIFICATIONS TO ENSURE CPU QOS



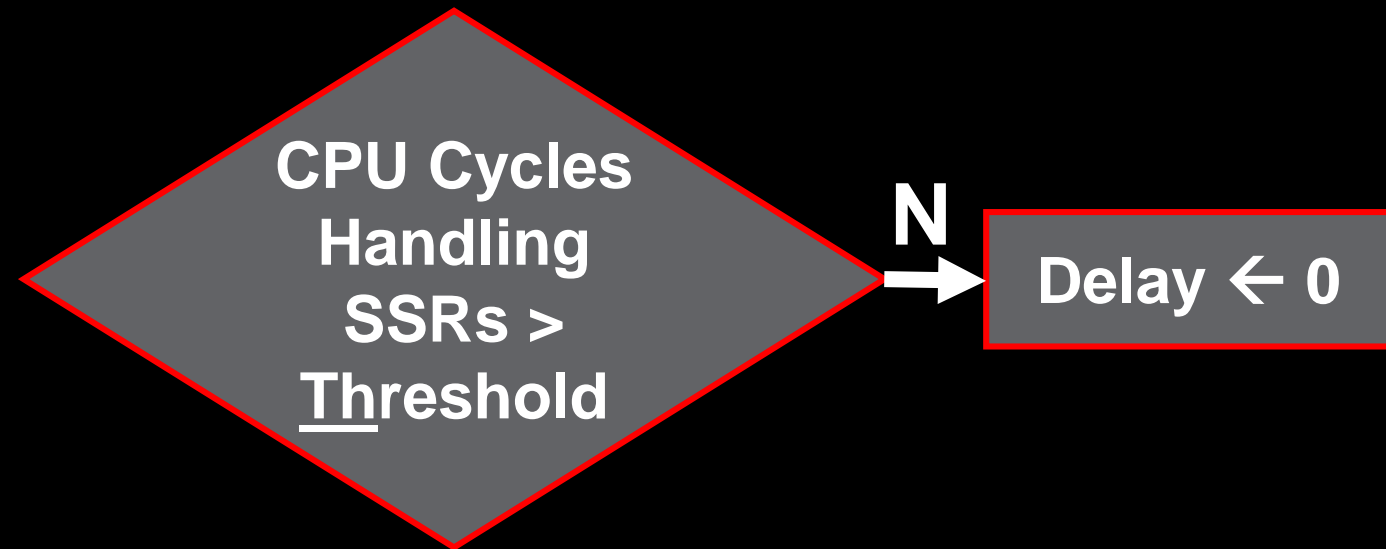
GOVERNOR TO CONTROL CPU QOS

GOVERNOR TO CONTROL CPU QOS

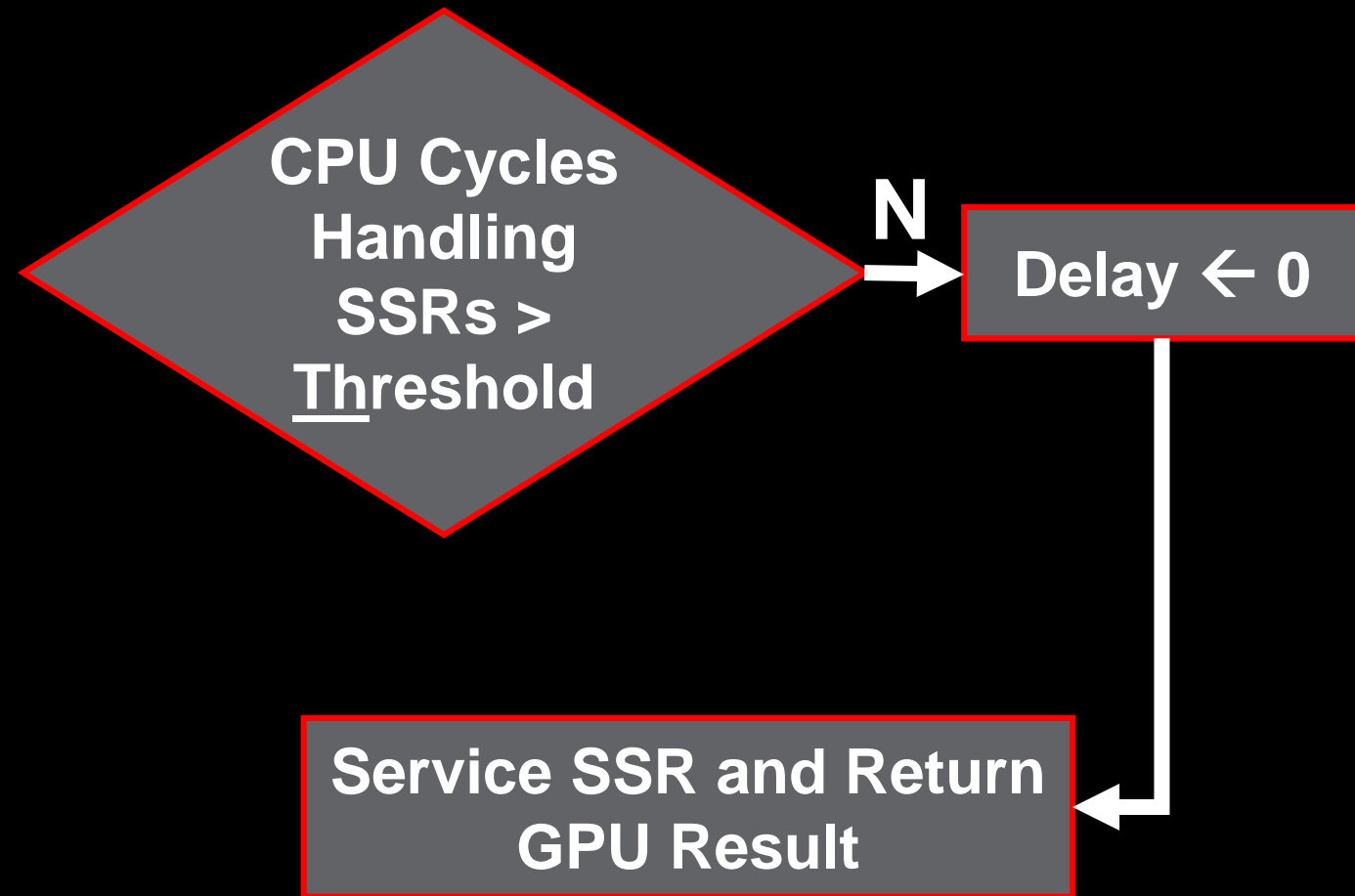


CPU Cycles
Handling
SSRs >
Threshold

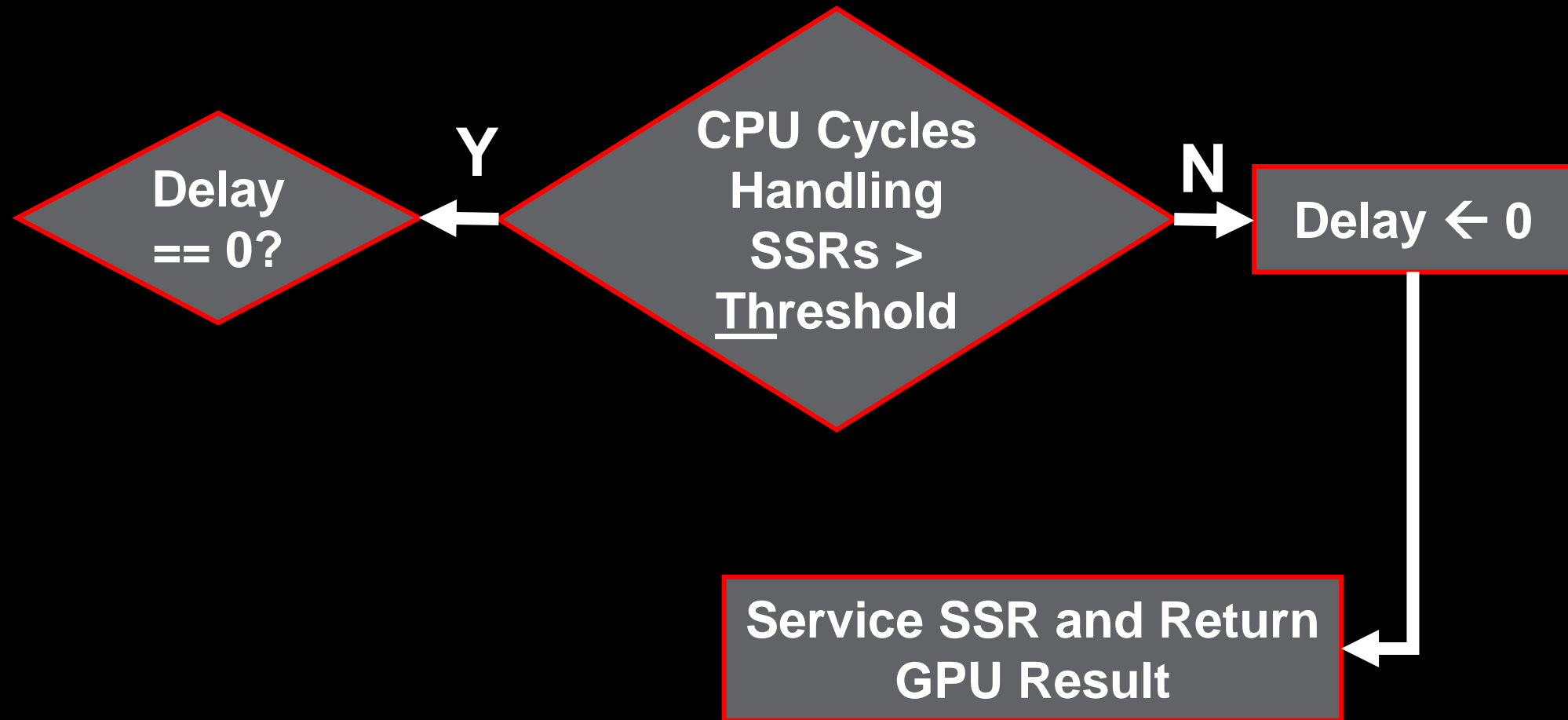
GOVERNOR TO CONTROL CPU QOS



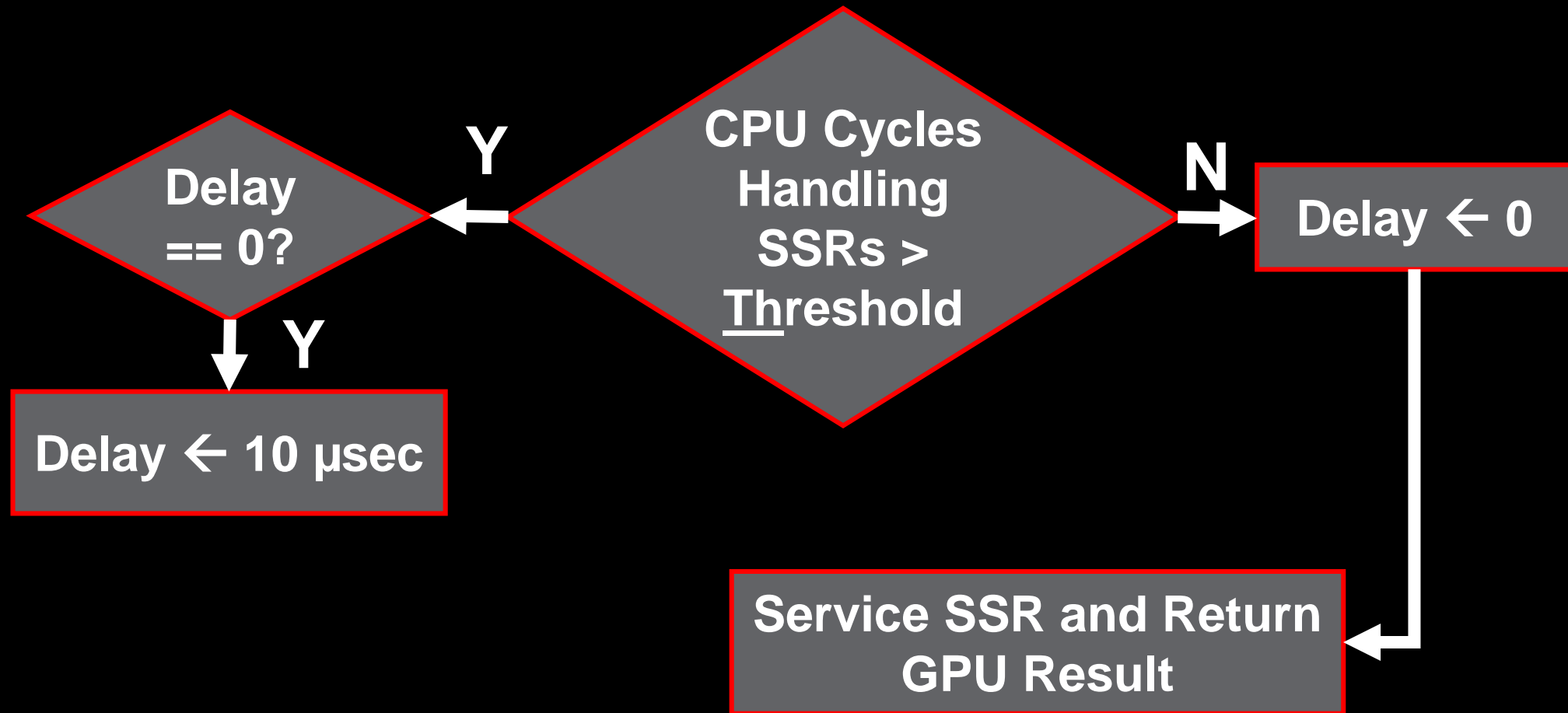
GOVERNOR TO CONTROL CPU QOS



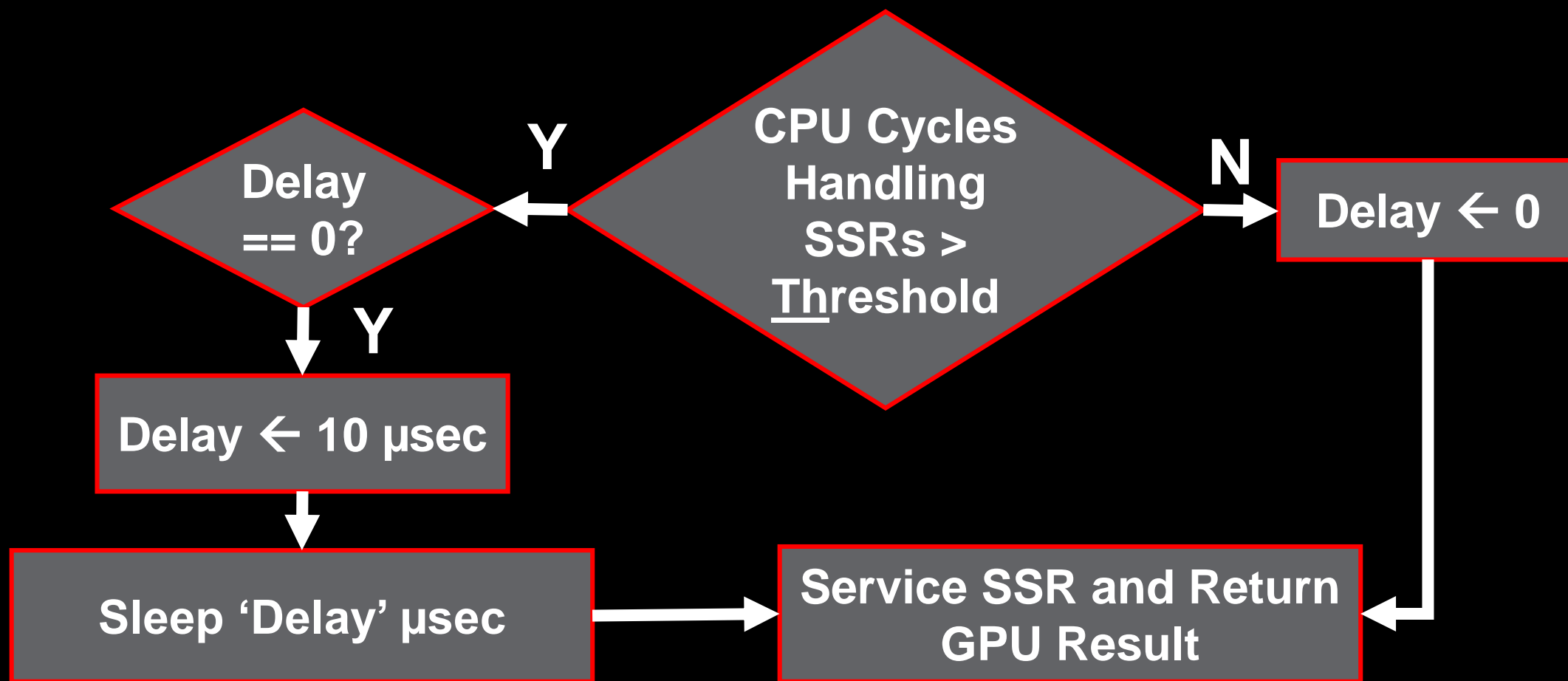
GOVERNOR TO CONTROL CPU QOS



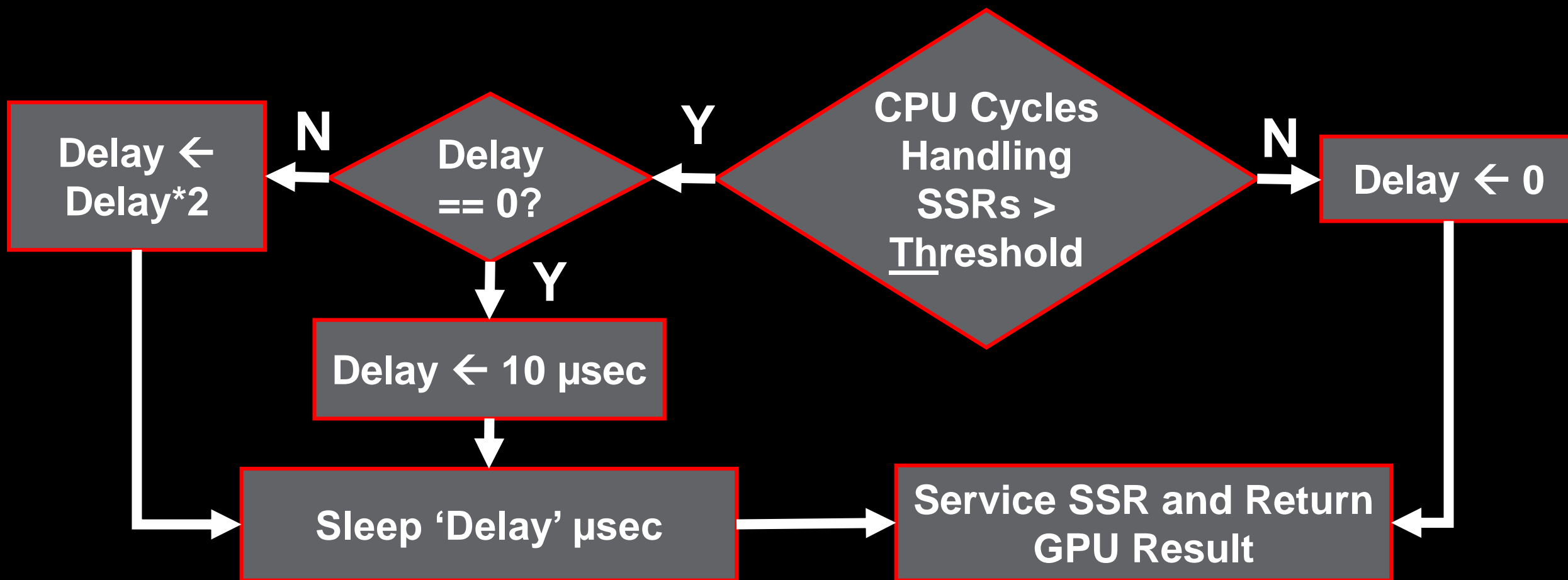
GOVERNOR TO CONTROL CPU QOS



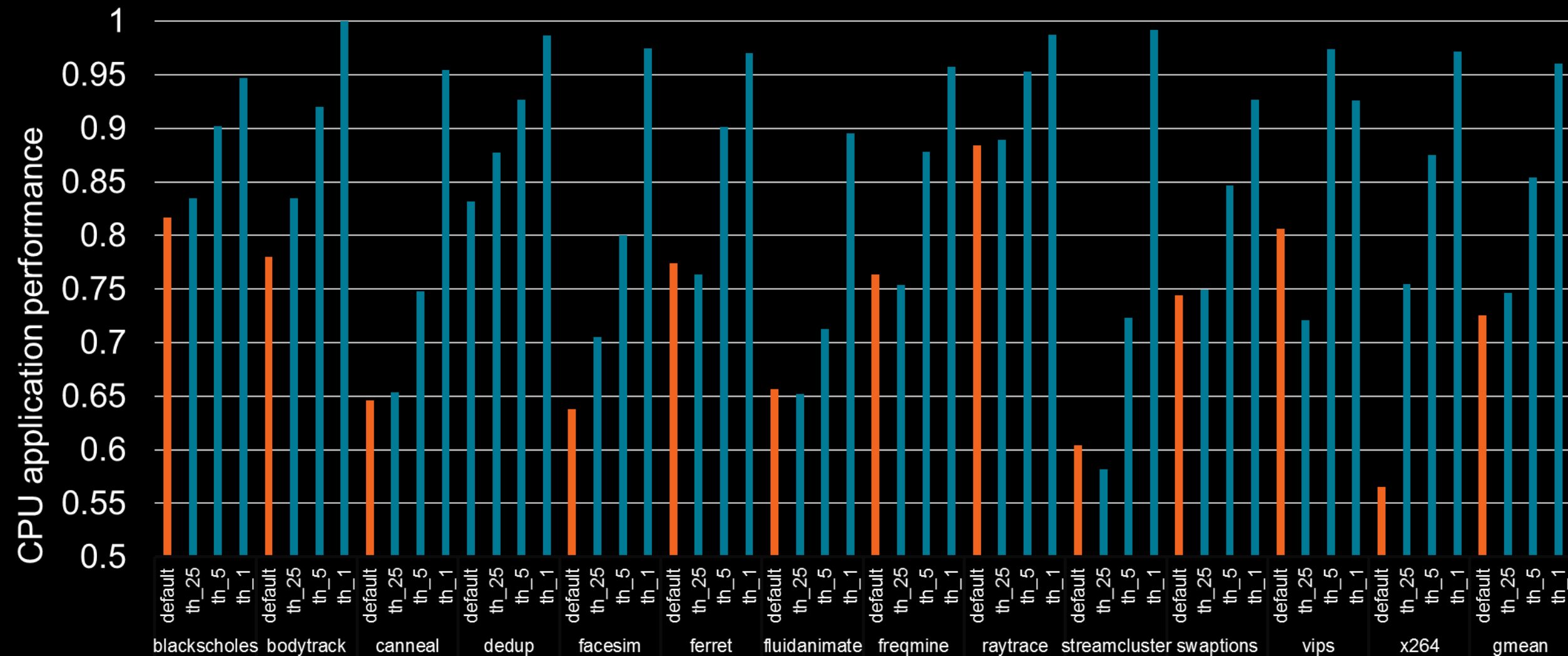
GOVERNOR TO CONTROL CPU QOS



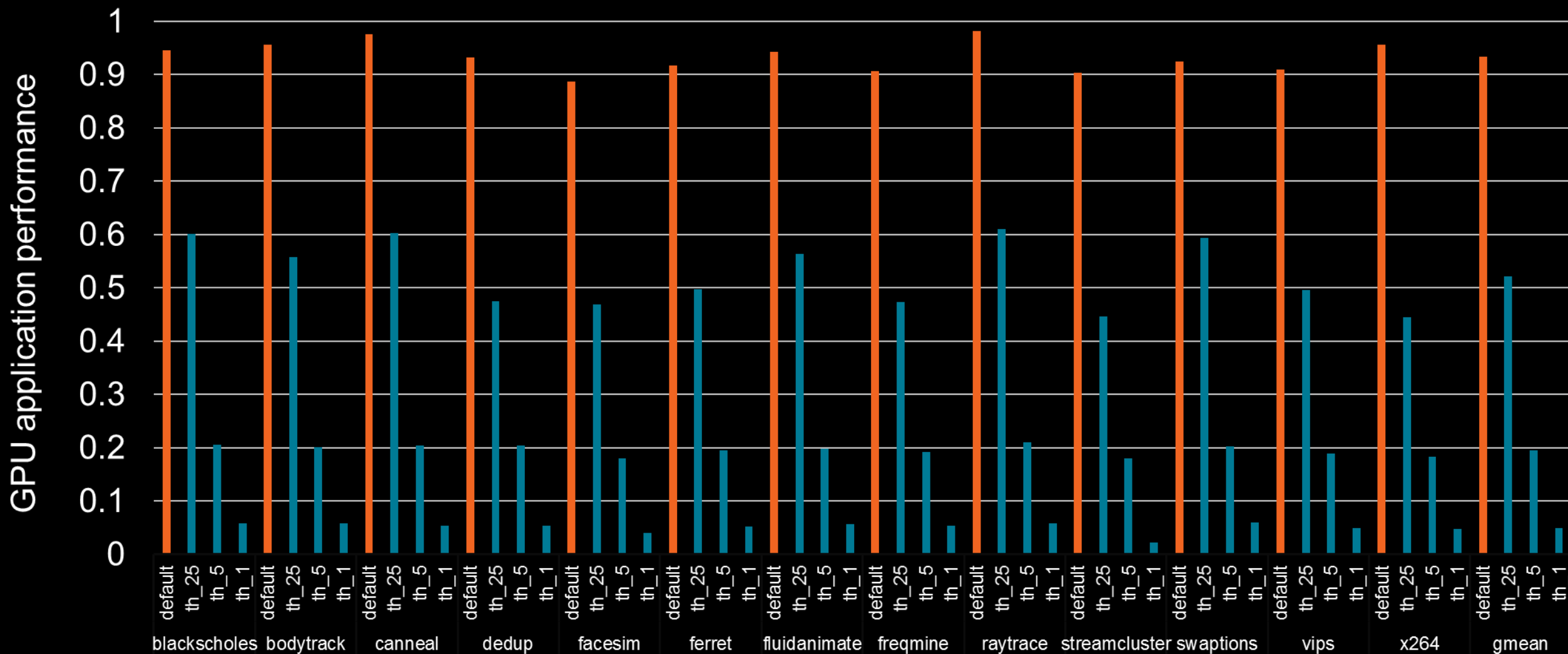
GOVERNOR TO CONTROL CPU QOS



CPU PERFORMANCE AT DIFFERENT QOS LEVELS



GPU PERFORMANCE SUFFERS FOR CPU QOS



SUMMARY

- Heterogeneous systems can include increasingly more accelerators
- GPUs and accelerators now request system services
- These can cause interference between accelerators & unrelated CPU work
- Problem may worsen in the future
- Existing mitigation strategies help, but are not complete solution



QUESTIONS?

Disclaimer

The information presented in this document is for informational purposes only and may contain technical inaccuracies, omissions and typographical errors.

The information contained herein is subject to change and may be rendered inaccurate for many reasons, including but not limited to product and roadmap changes, component and motherboard version changes, new model and/or product releases, product differences between differing manufacturers, software changes, BIOS flashes, firmware upgrades, or the like. AMD assumes no obligation to update or otherwise correct or revise this information. However, AMD reserves the right to revise this information and to make changes from time to time to the content hereof without obligation of AMD to notify any person of such revisions or changes.

AMD MAKES NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE CONTENTS HEREOF AND ASSUMES NO RESPONSIBILITY FOR ANY INACCURACIES, ERRORS OR OMISSIONS THAT MAY APPEAR IN THIS INFORMATION.

AMD SPECIFICALLY DISCLAIMS ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE. IN NO EVENT WILL AMD BE LIABLE TO ANY PERSON FOR ANY DIRECT, INDIRECT, SPECIAL OR OTHER CONSEQUENTIAL DAMAGES ARISING FROM THE USE OF ANY INFORMATION CONTAINED HEREIN, EVEN IF AMD IS EXPRESSLY ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

AMD, the AMD Arrow logo, and combinations thereof are trademarks of Advanced Micro Devices, Inc. Linux is a registered trademark of Linus Torvalds. OpenCL is a trademark of Apple Inc. used by permission by Khronos. Other names used herein are for identification purposes only and may be trademarks of their respective companies.

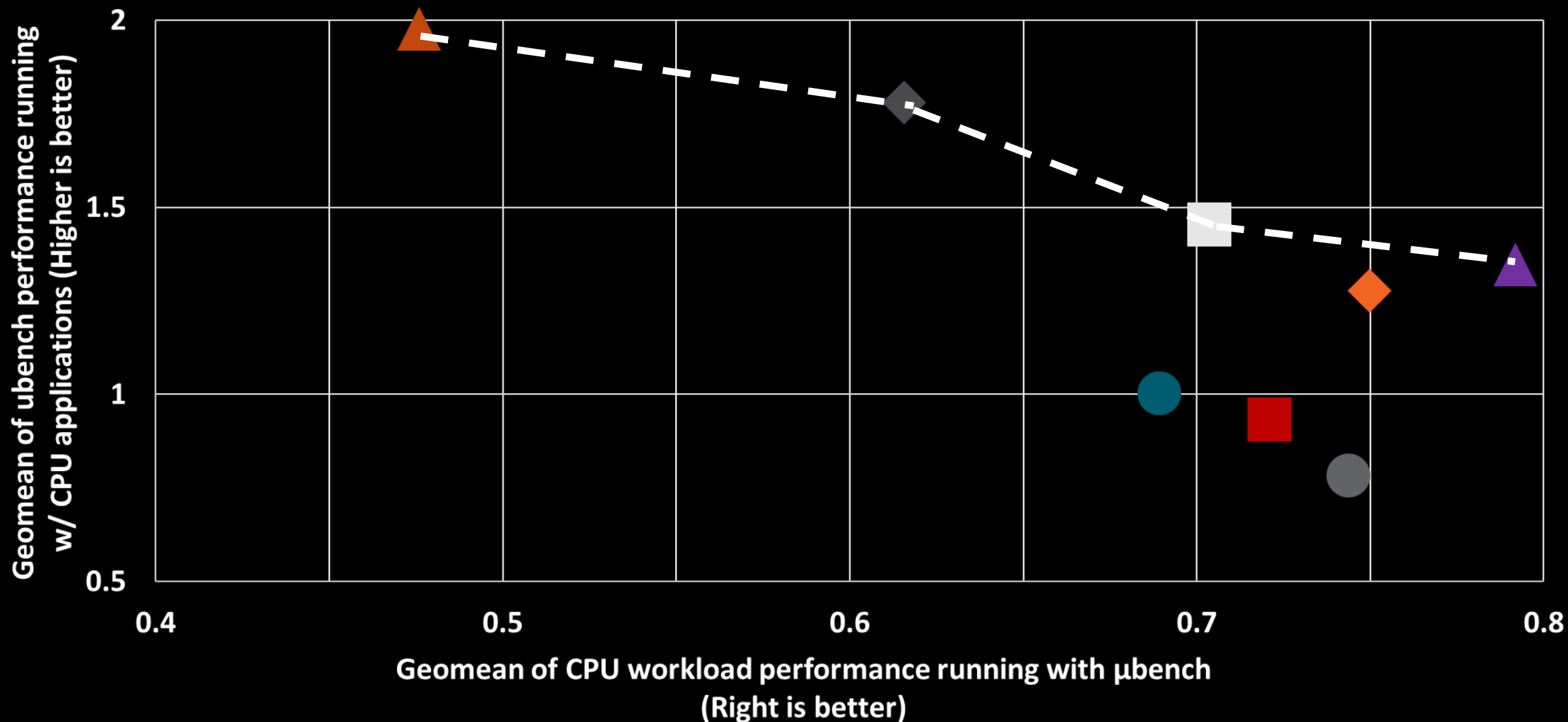


BACKUP SLIDES

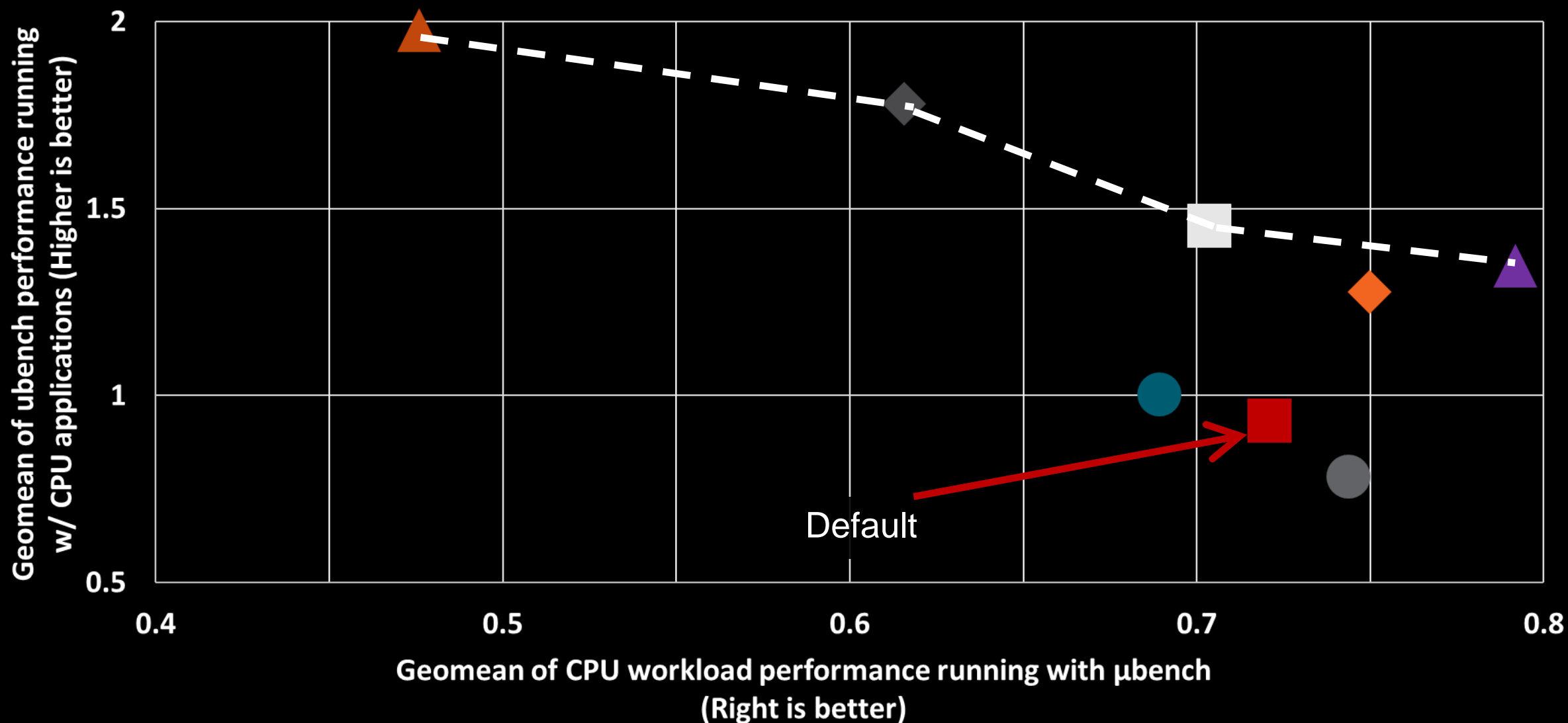
PARETO CURVE OF MITIGATION STRATEGY TRADEOFFS



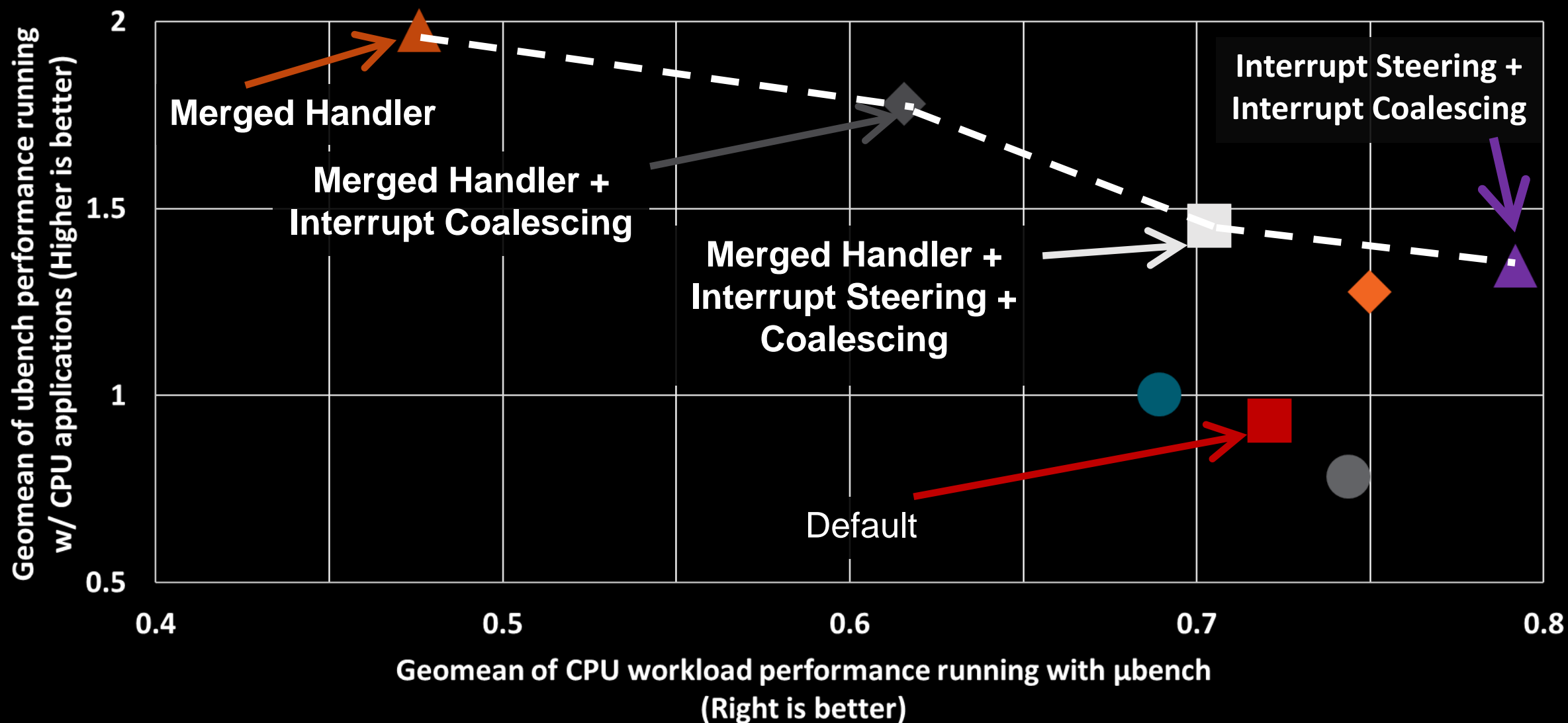
PARETO CURVE OF MITIGATION STRATEGY TRADEOFFS



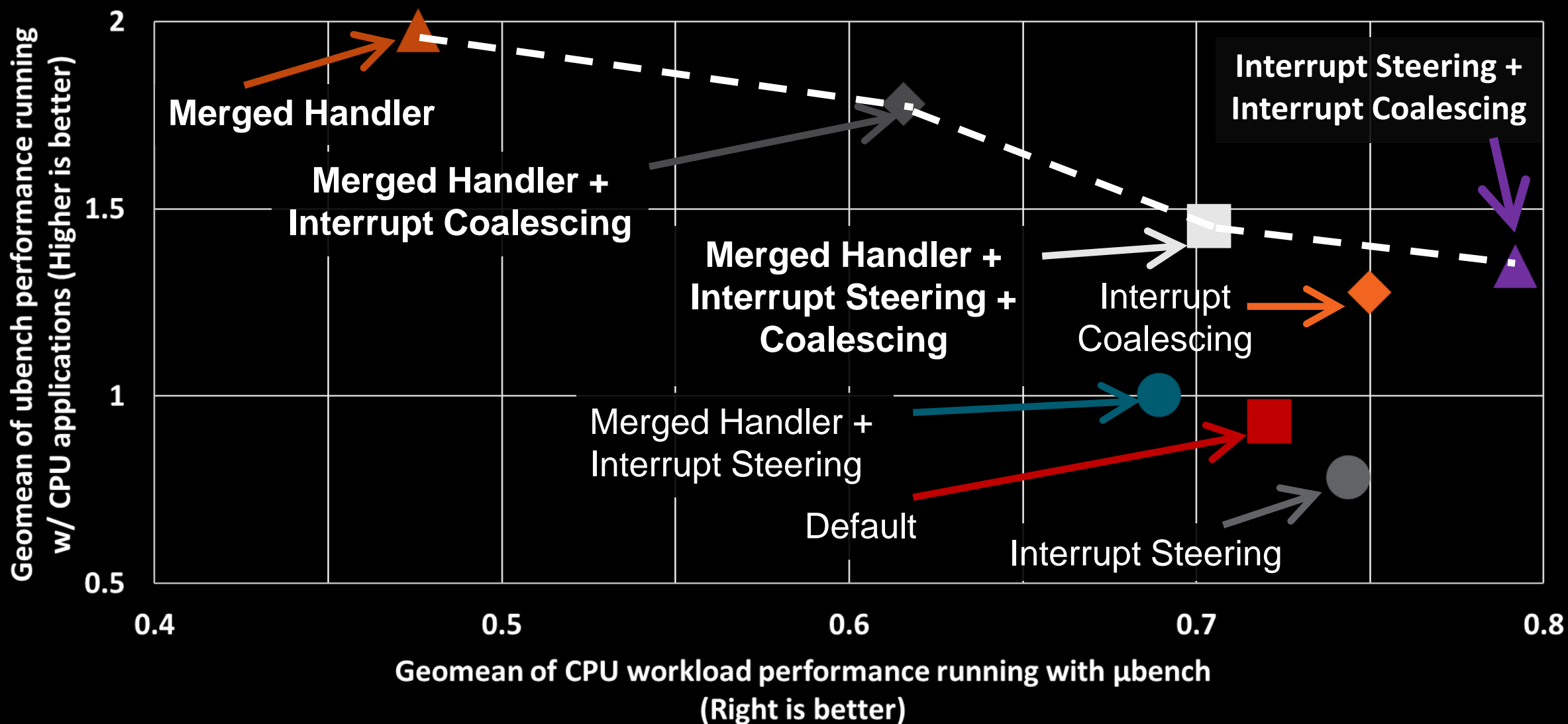
PARETO CURVE OF MITIGATION STRATEGY TRADEOFFS



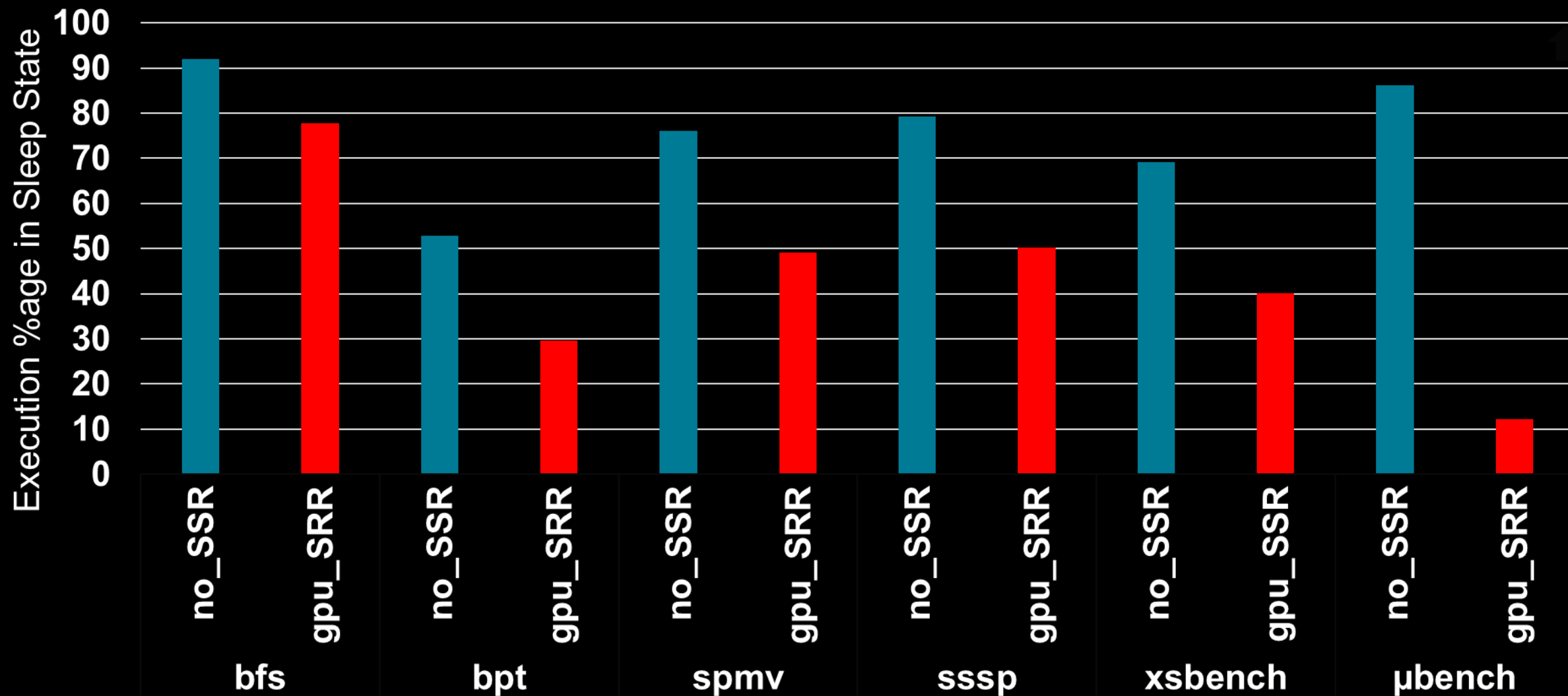
PARETO CURVE OF MITIGATION STRATEGY TRADEOFFS



PARETO CURVE OF MITIGATION STRATEGY TRADEOFFS



SSRS LIMIT LOW-POWER SLEEP STATES



MITIGATION EFFECT ON SLEEP STATES

