Are Coherence Protocol States vulnerable to Information Leakage?

Fan Yao, Milos Doroslovacki, Guru Venkataramani
The George Washington University

HPCA 2018, Vienna, Austria

The Trend of Security Threats

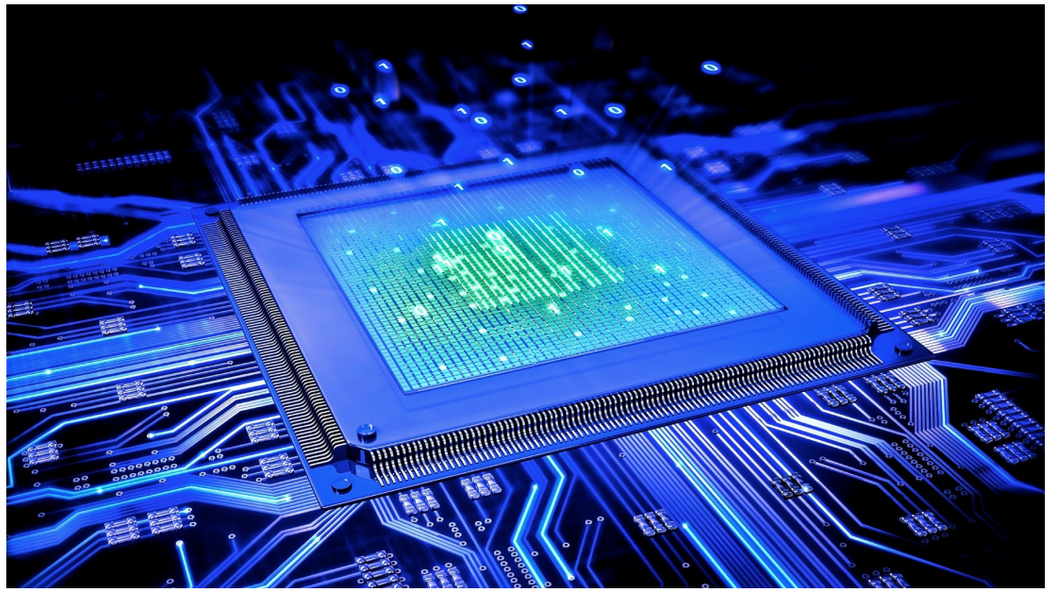
The Trend of Security Threats

Traditional Attacks

The Trend of Security Threats

Traditional Attacks

Emerging Attacks



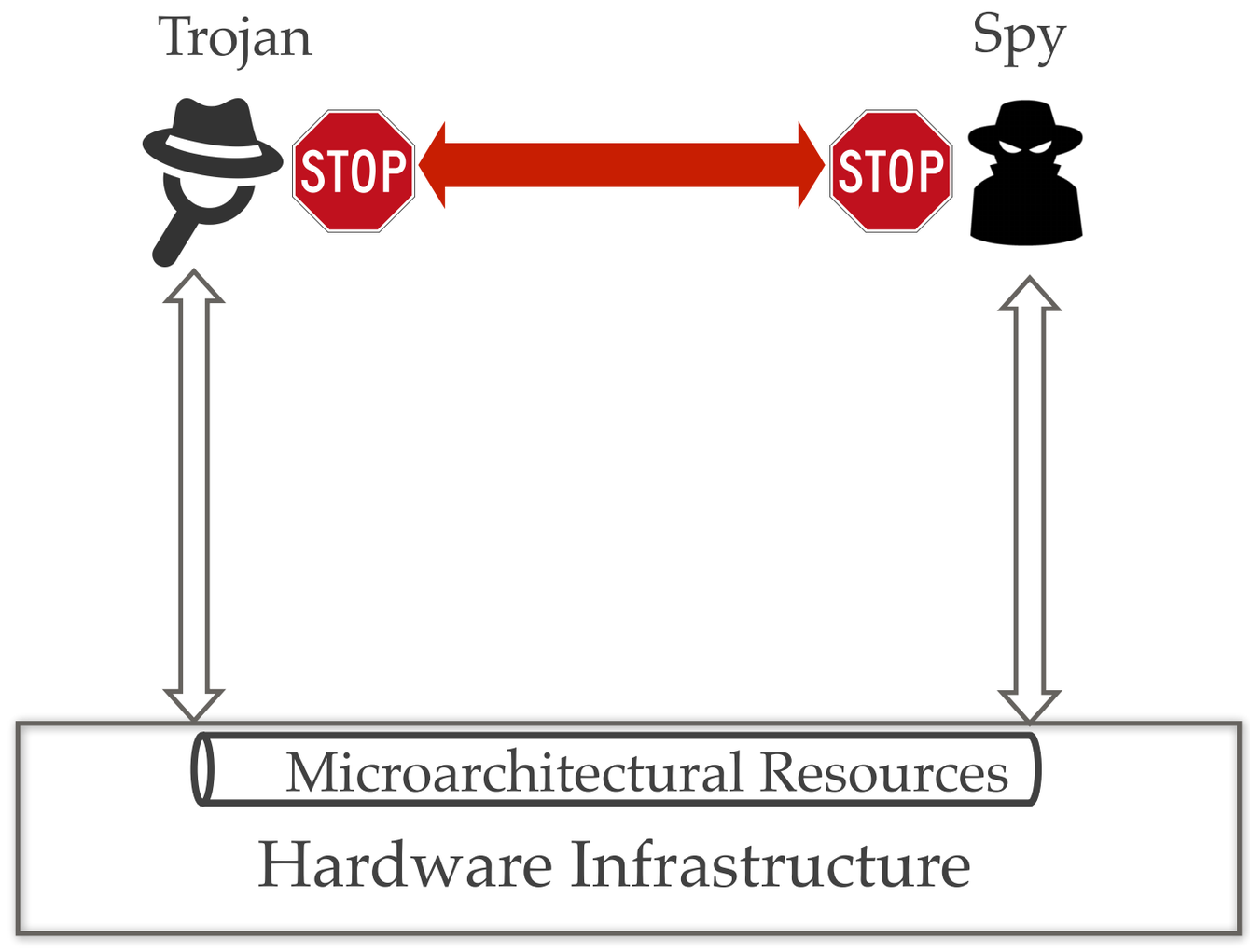
Information Leakage on Processors

Covert Timing Channels

Hardware Infrastructure

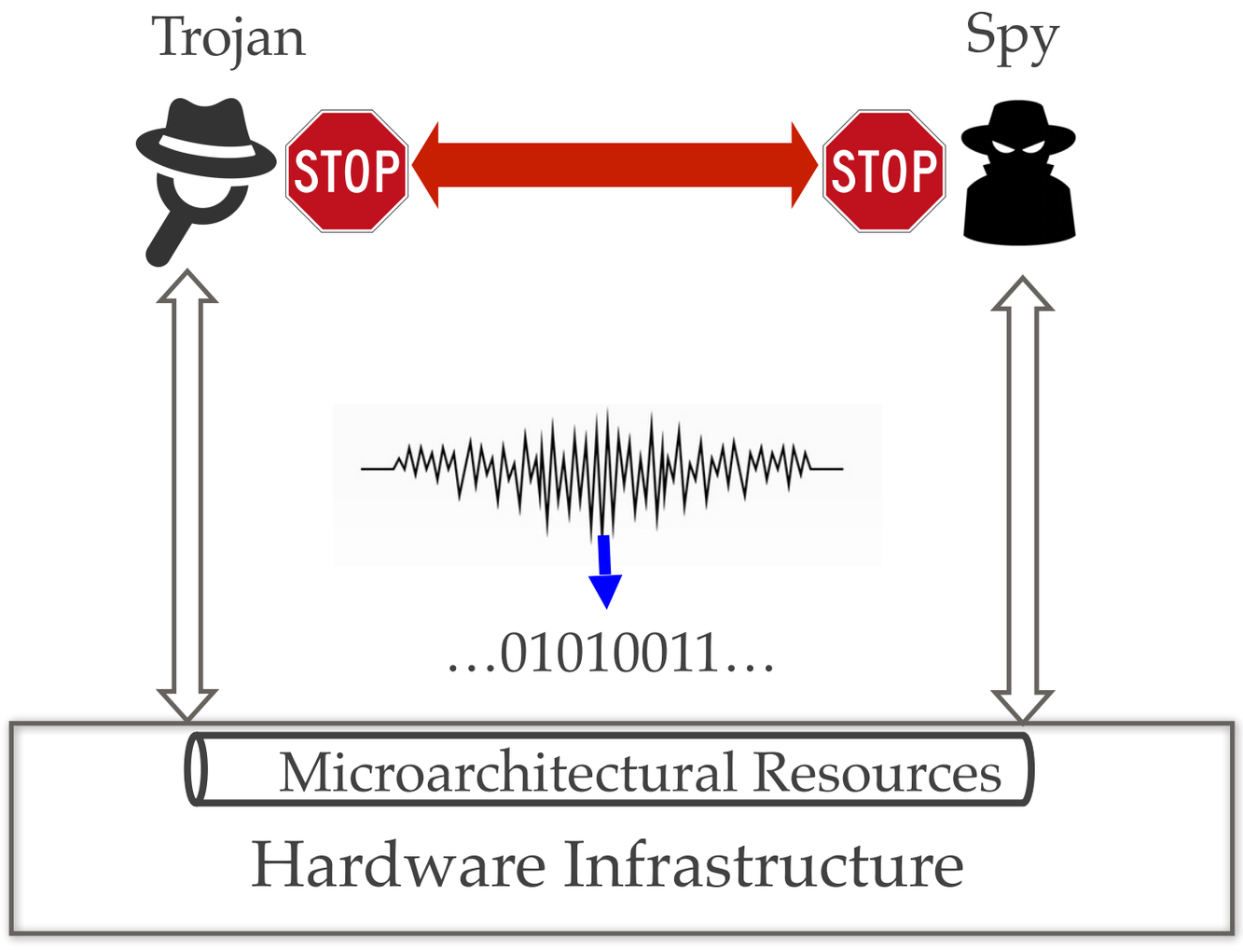
Information Leakage on Processors

Covert Timing Channels



Information Leakage on Processors

Covert Timing Channels



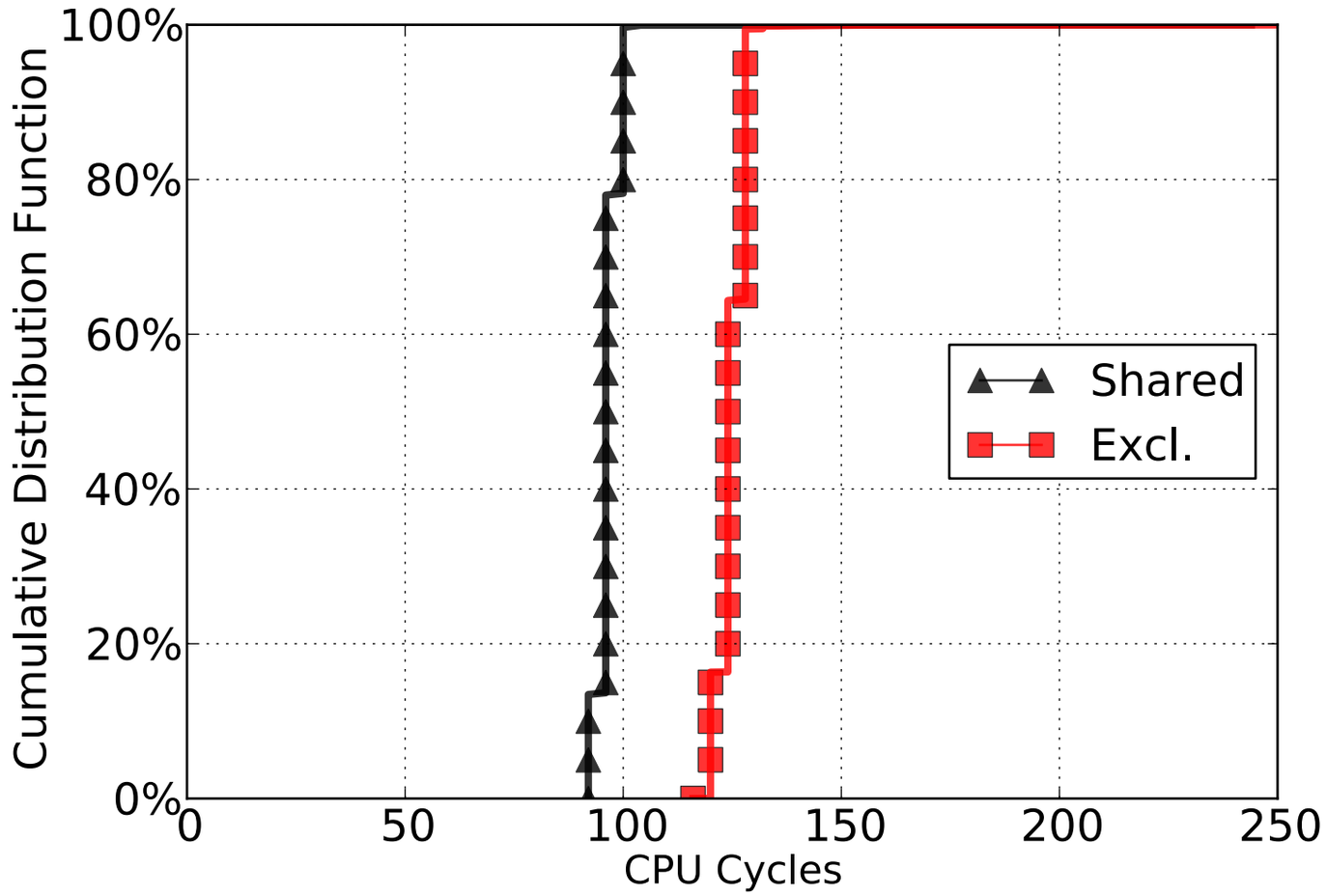
Related Works on Covert Timing Channels

- Cache timing channels are dominant
  - Caches expose relatively large attack surface
- Existing Channels
  - Flush + Reload attack
    - Spy flushes blocks and reloads them later
    - Yarom et al. USENIX Security'14, He et al. MICRO'17
  - Prime + Probe attack
    - Spy primes the cache with its own blocks and probes them later
    - Chen et al. MICRO'14, Hunger et al. HPCA'15, Yan et al. MICRO'16

Is Cache Coherence Fabric Leaky ?

Is Cache Coherence Fabric Leaky ?

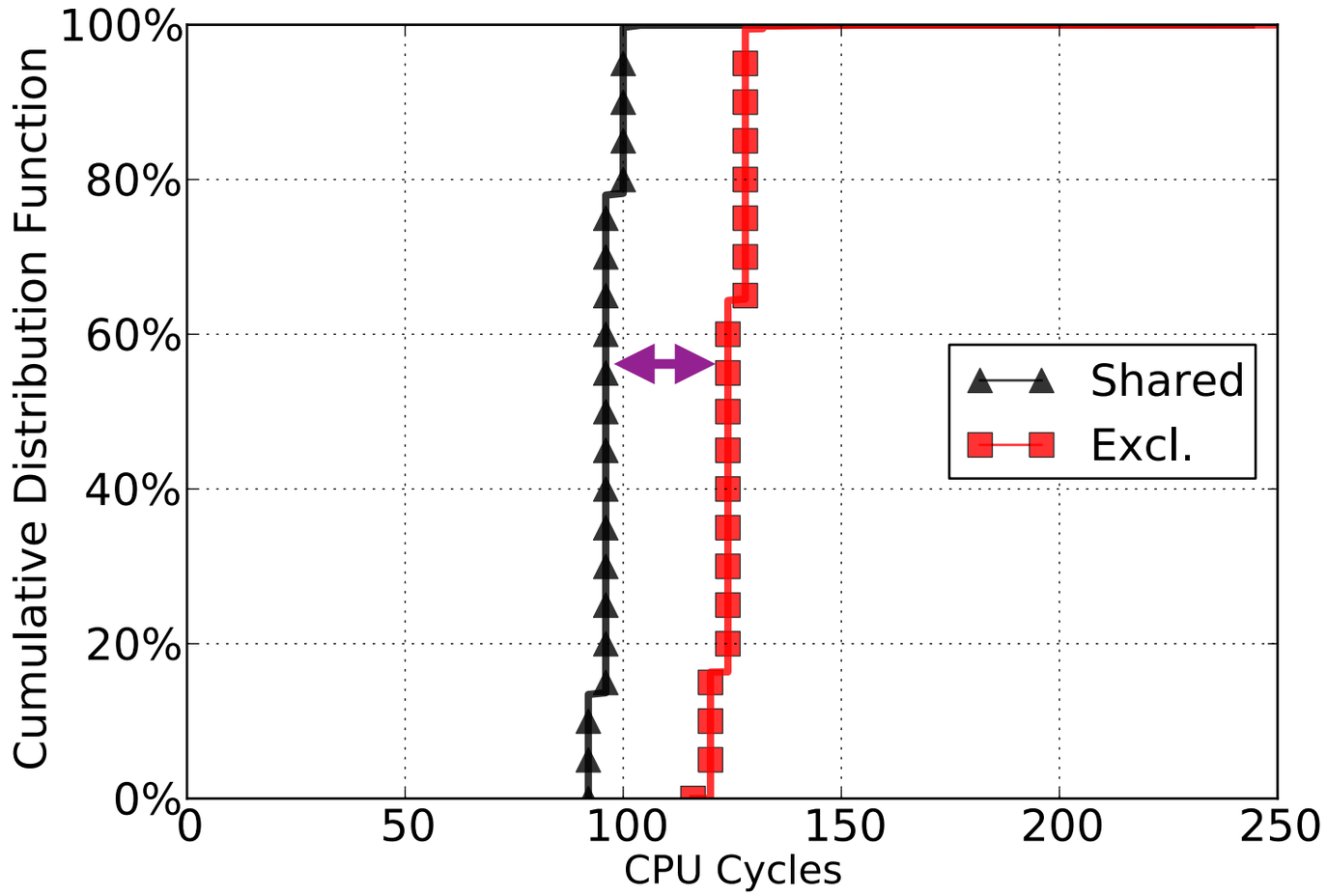
Load operation latency in Exclusive and Shared states*



* Latency measurements from Intel Xeon 5650 processor

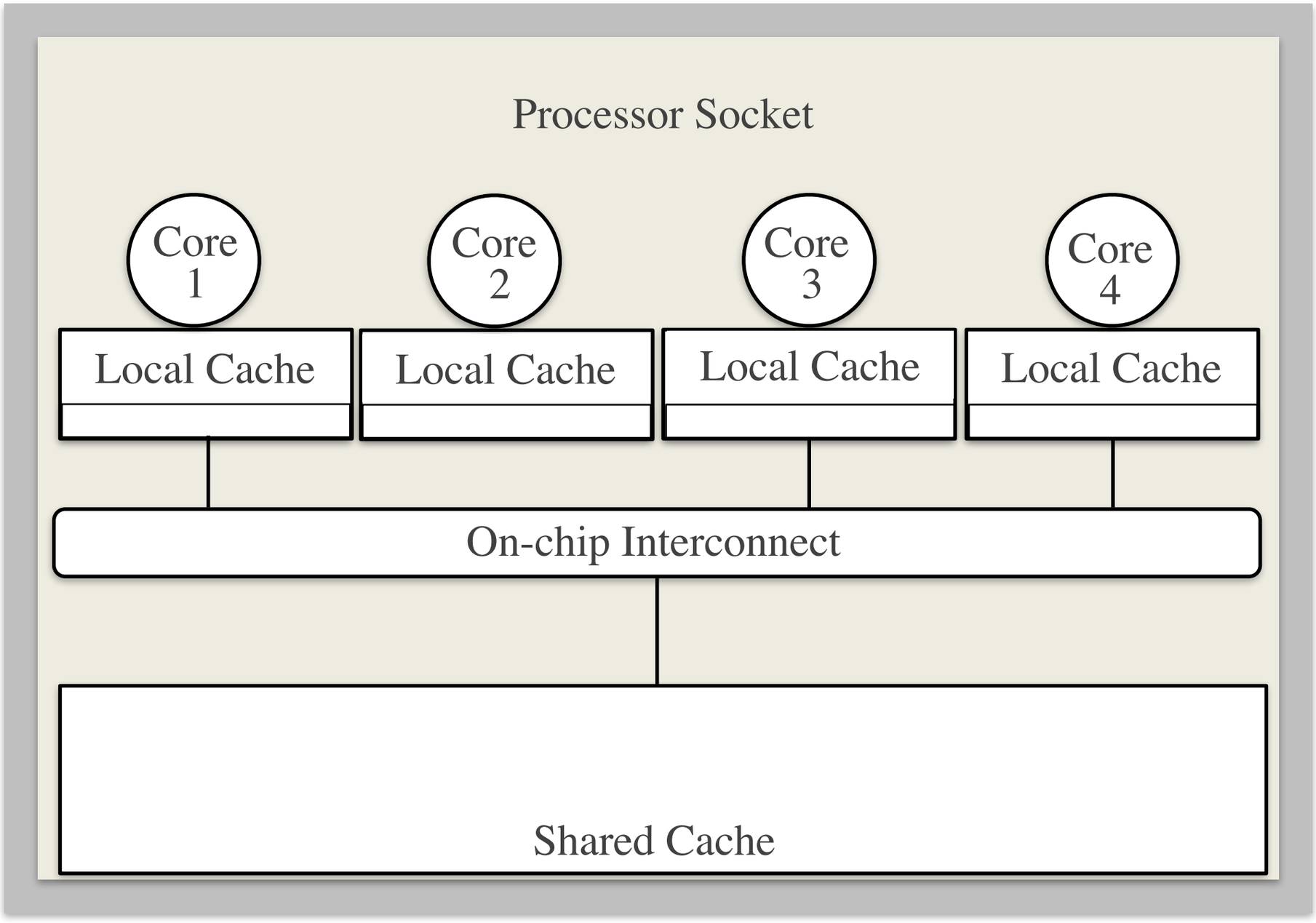
Is Cache Coherence Fabric Leaky ?

Load operation latency in Exclusive and Shared states*

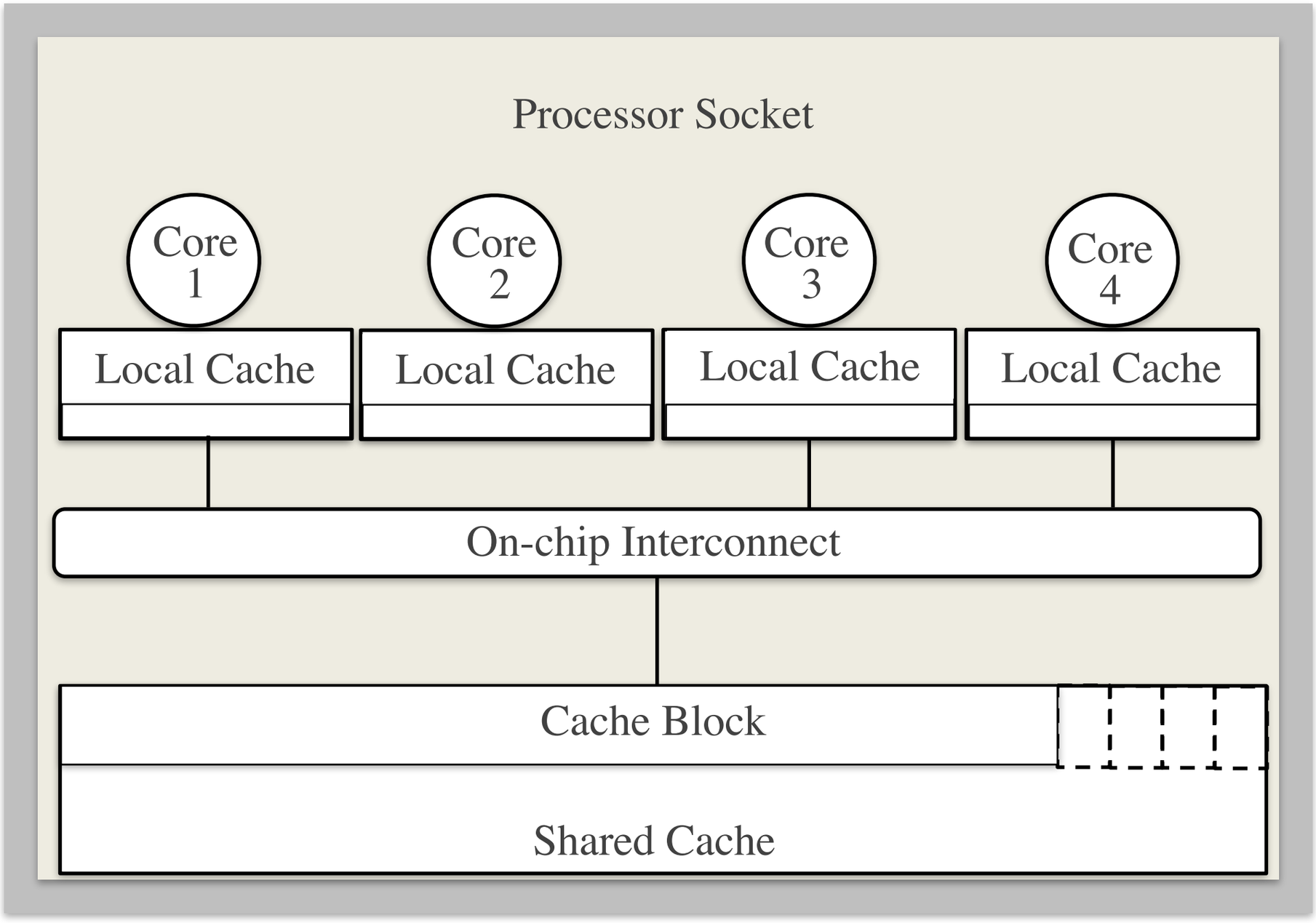


* Latency measurements from Intel Xeon 5650 processor

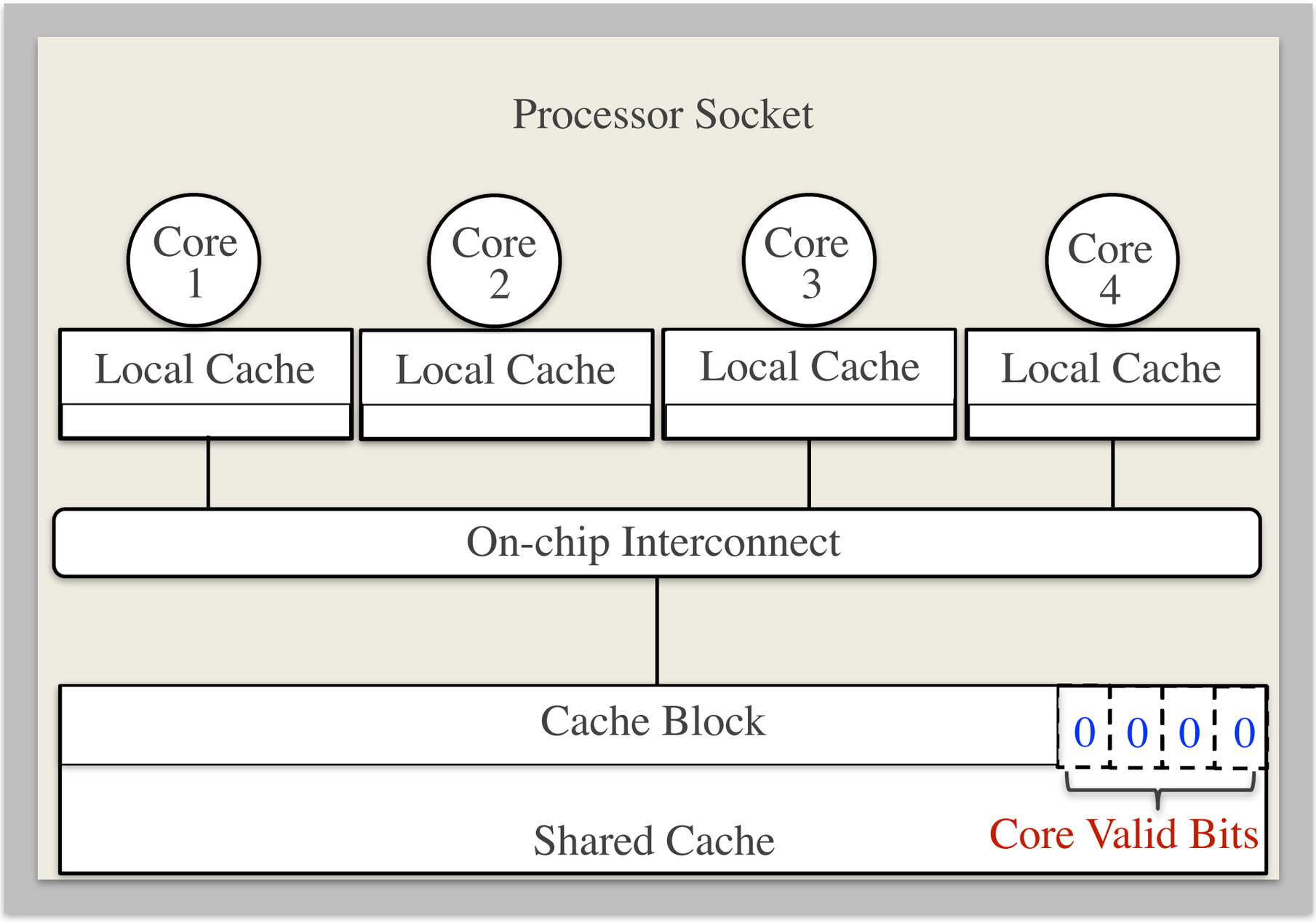
Cache Coherence in Multi-core Processors



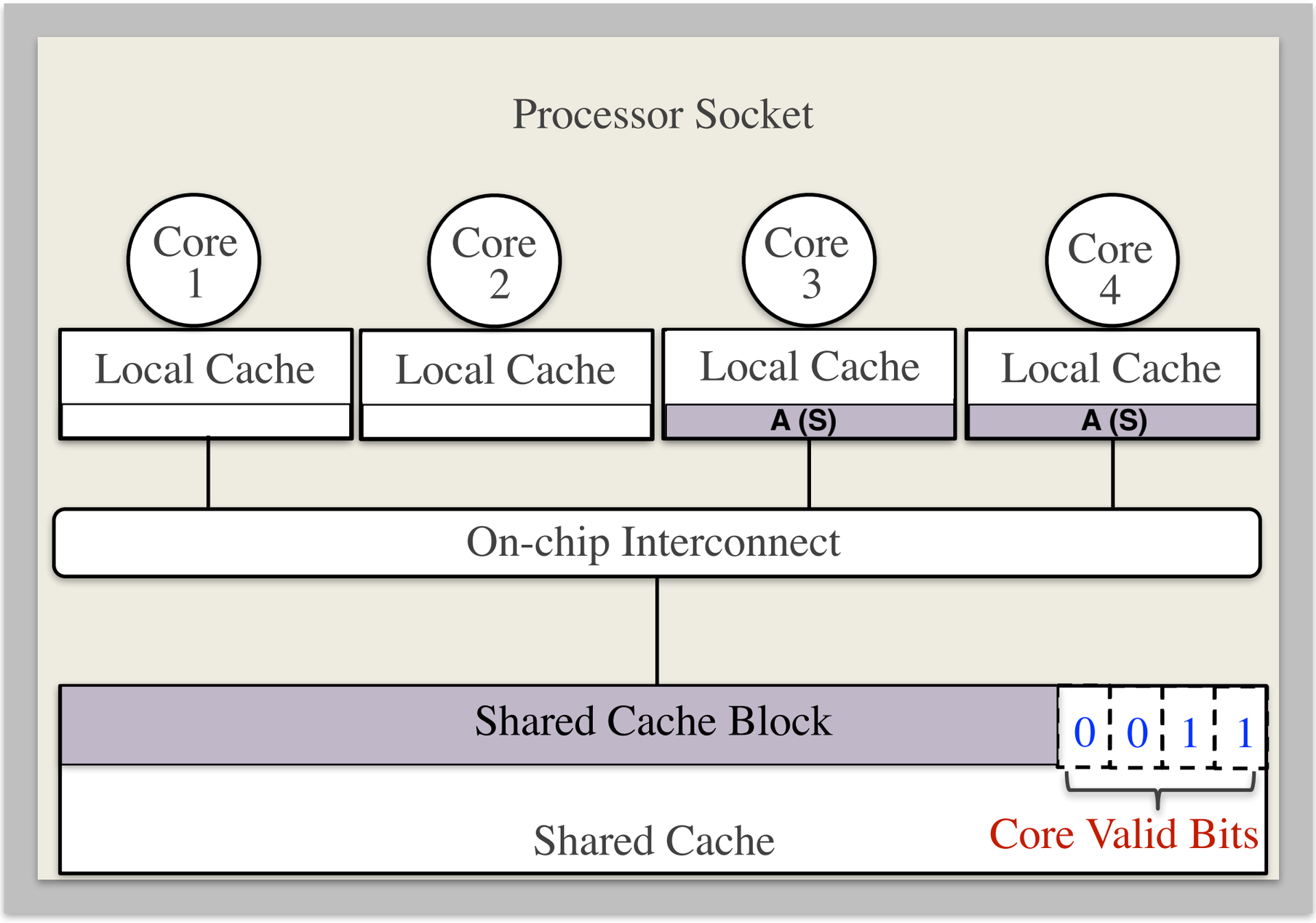
Cache Coherence in Multi-core Processors



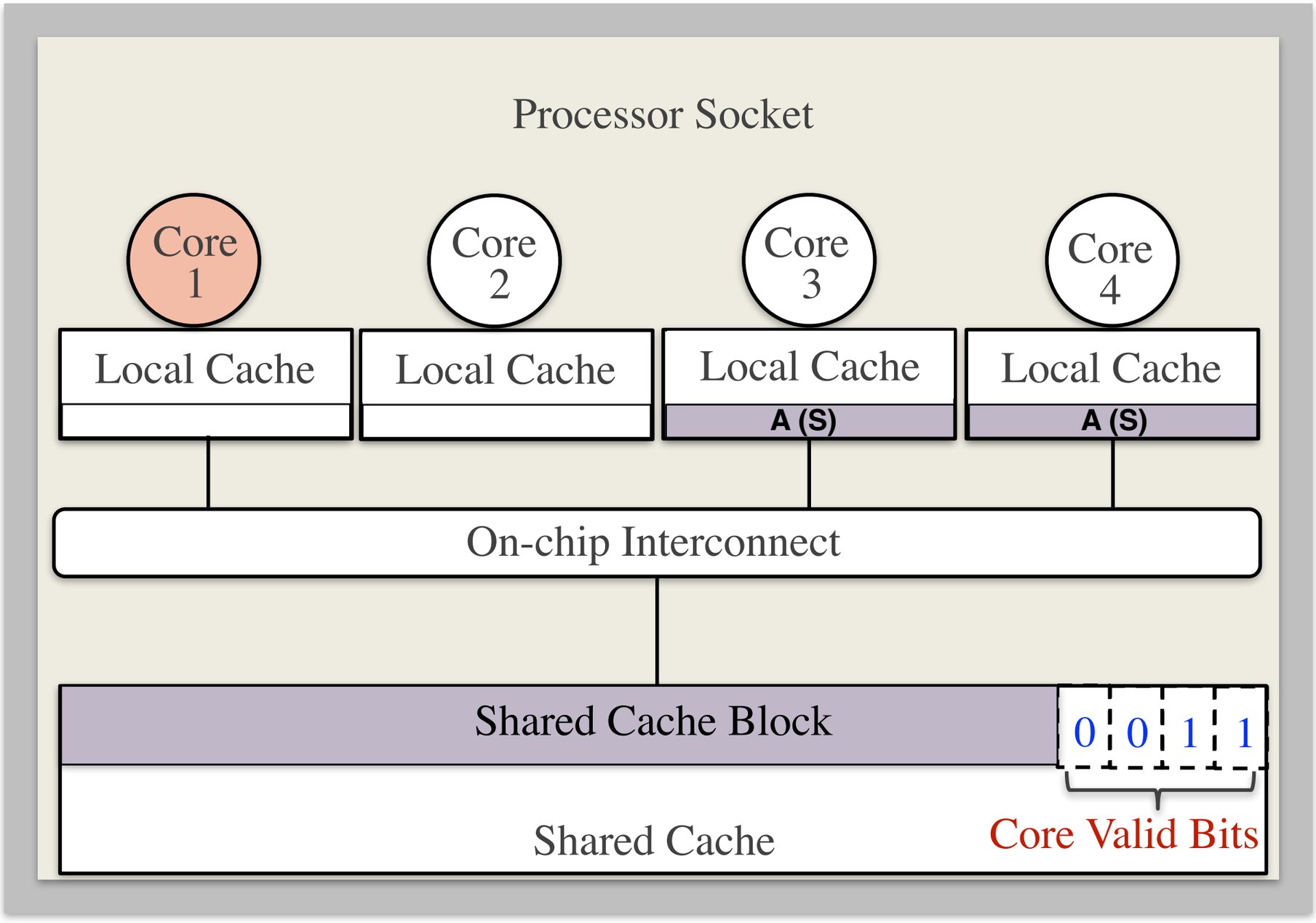
Cache Coherence in Multi-core Processors



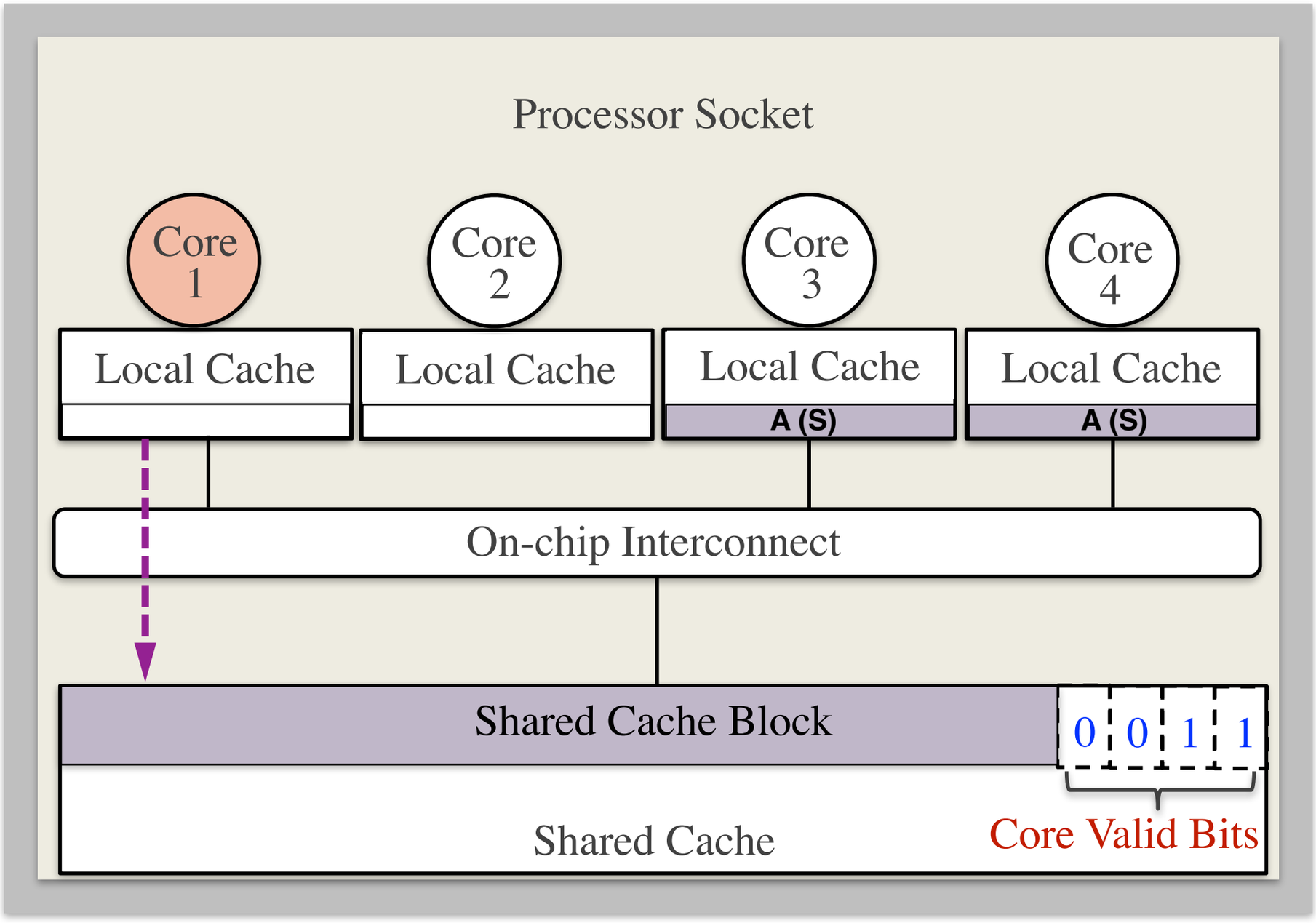
Cache Coherence - Shared Block



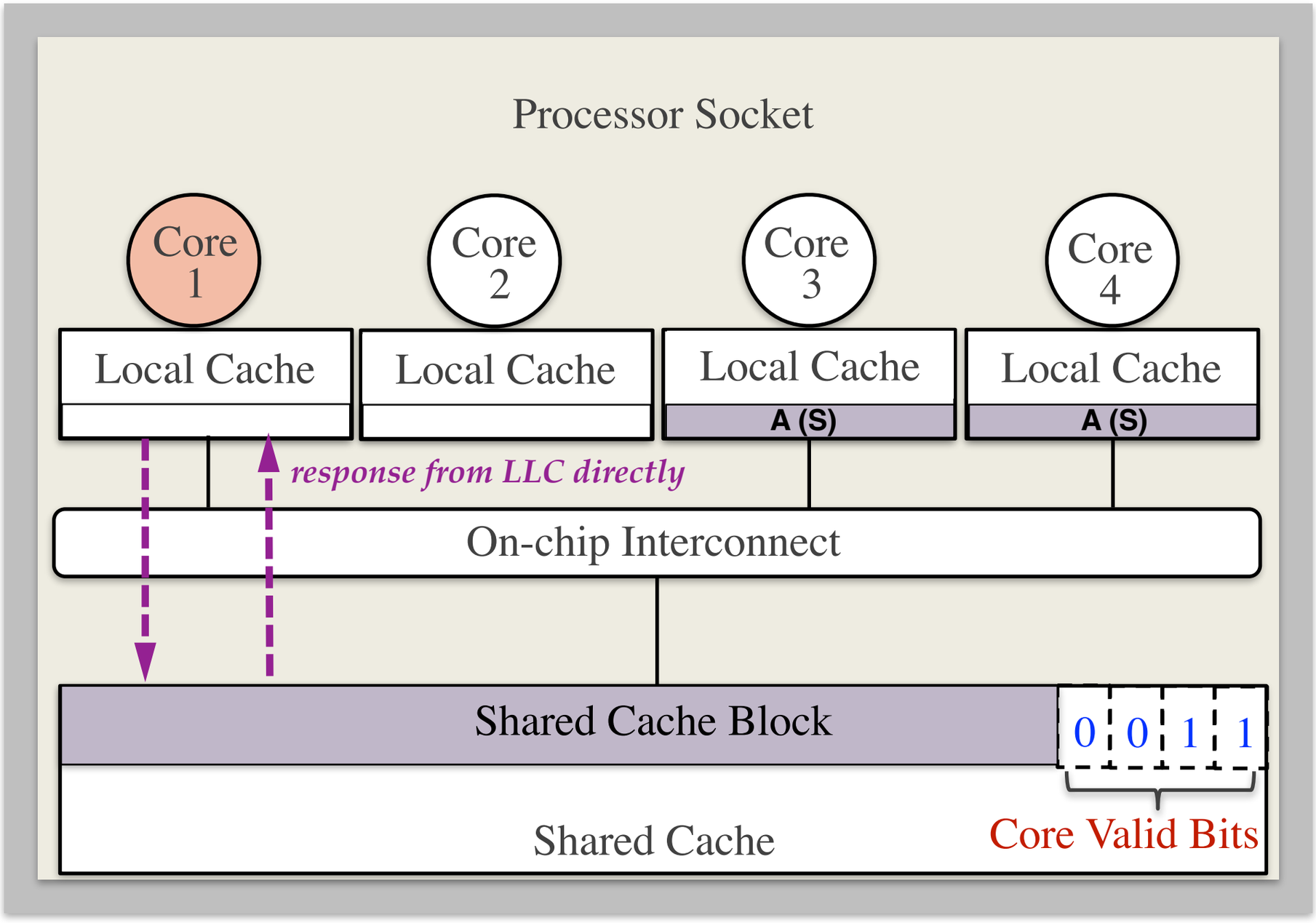
Cache Coherence - Shared Block



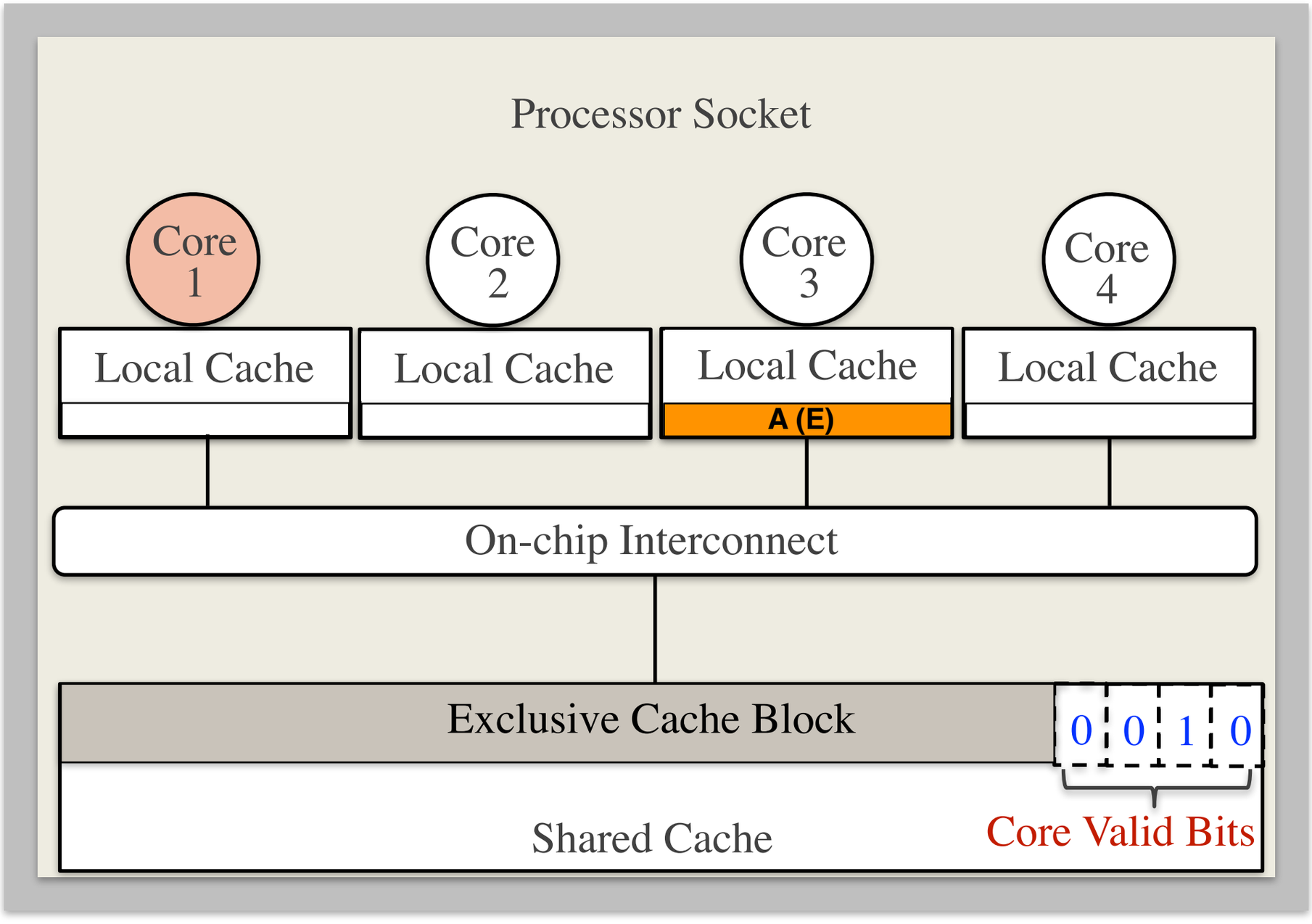
Cache Coherence - Shared Block



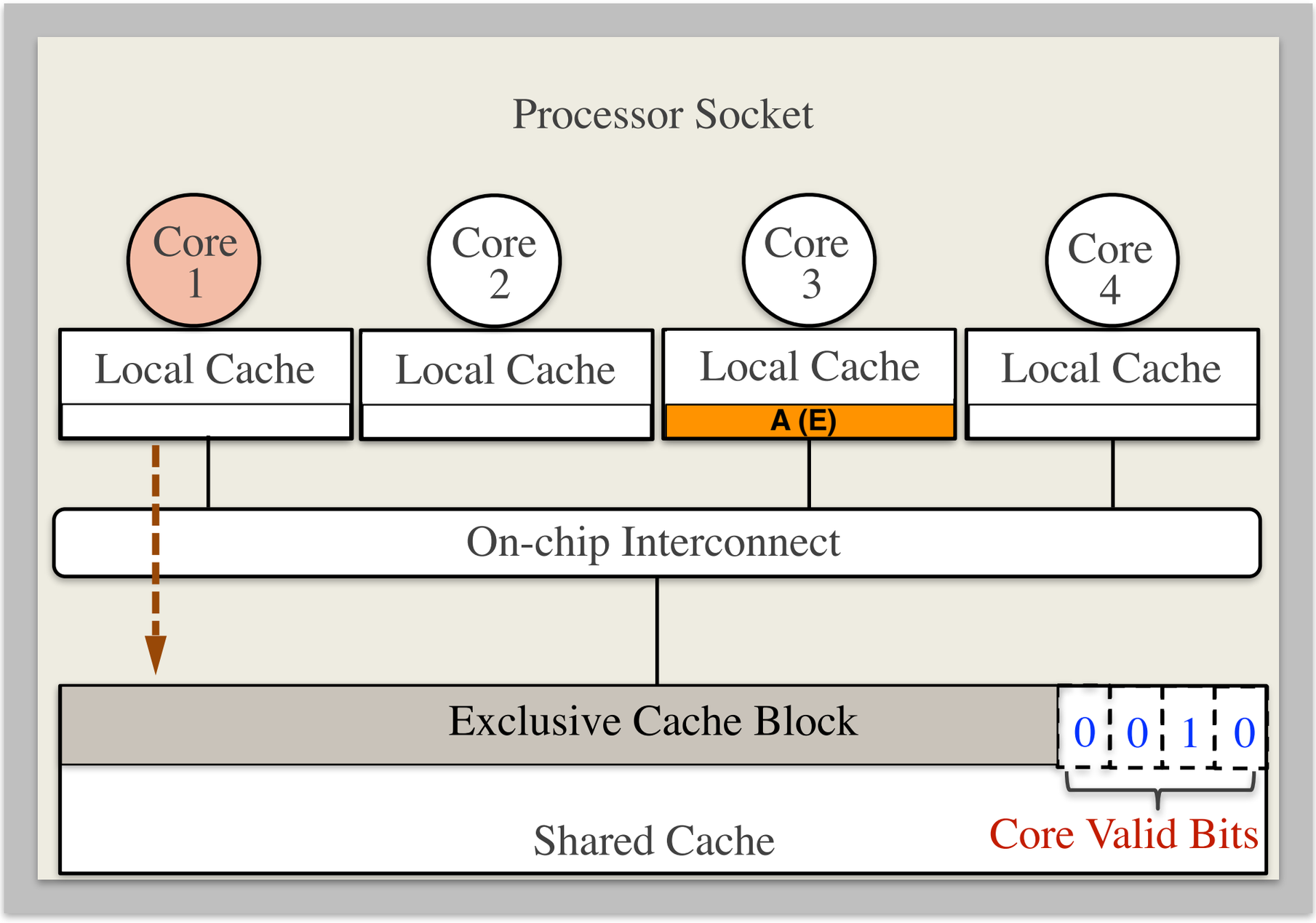
Cache Coherence - Shared Block



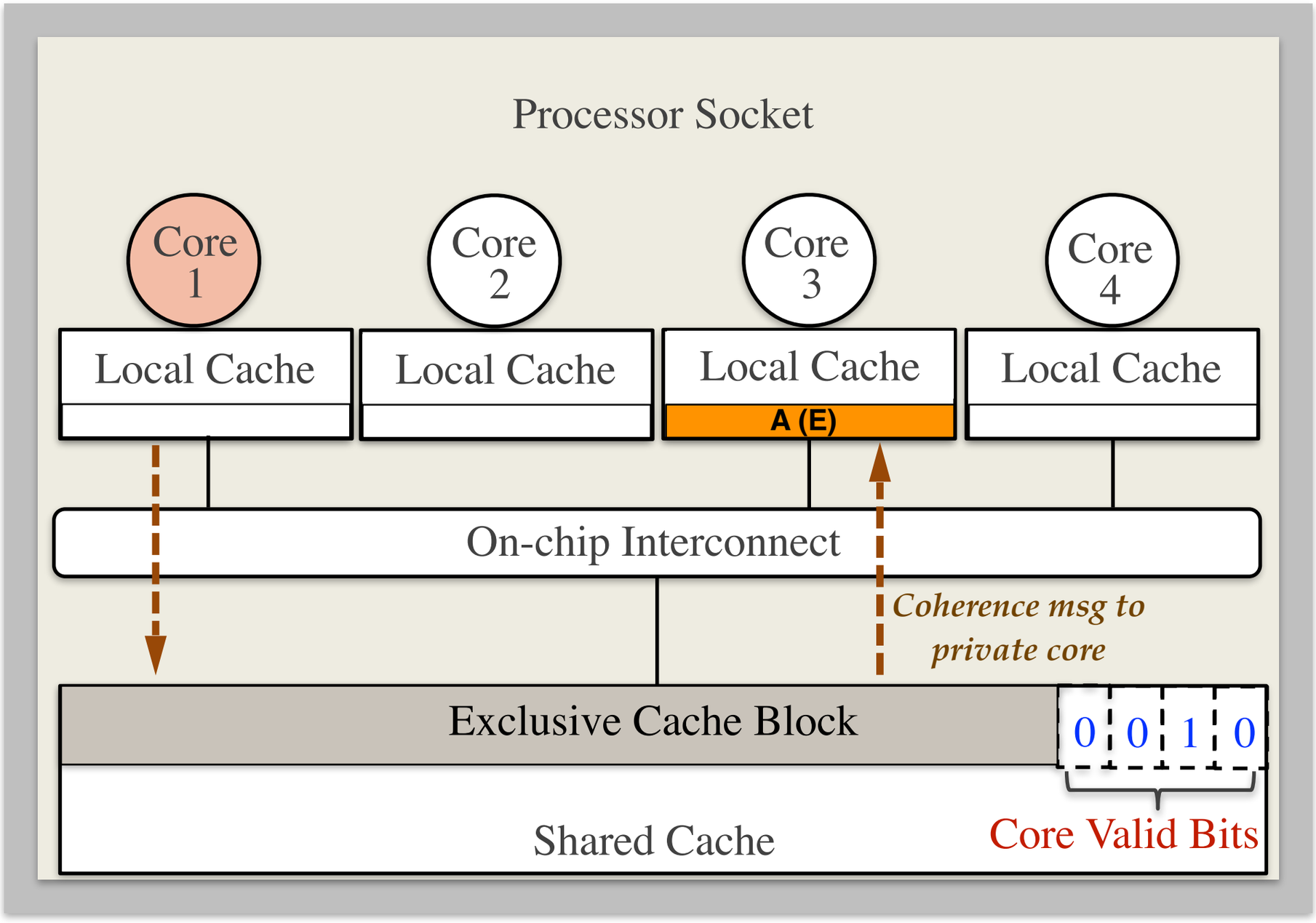
Cache Coherence – Exclusive Block



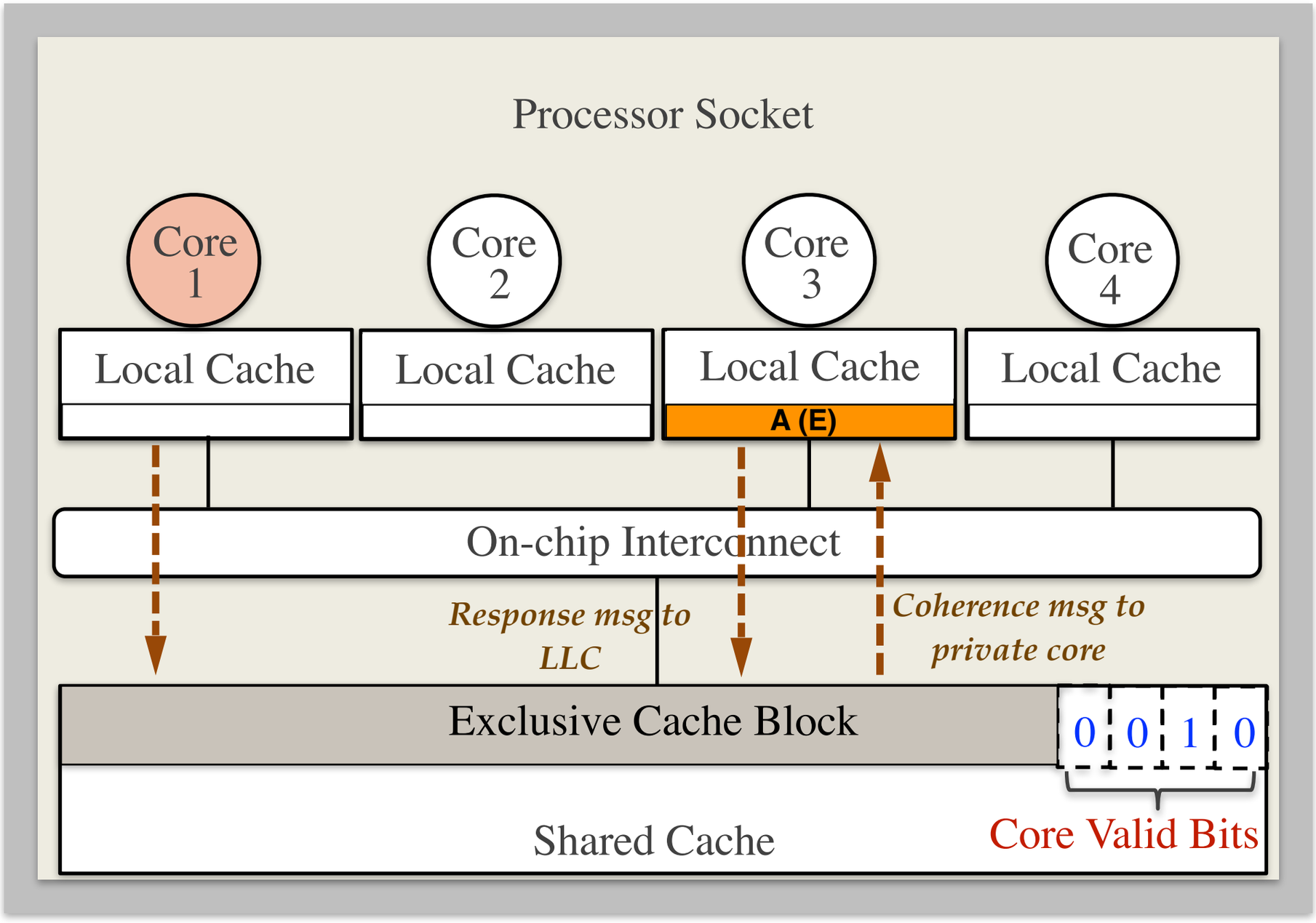
Cache Coherence – Exclusive Block



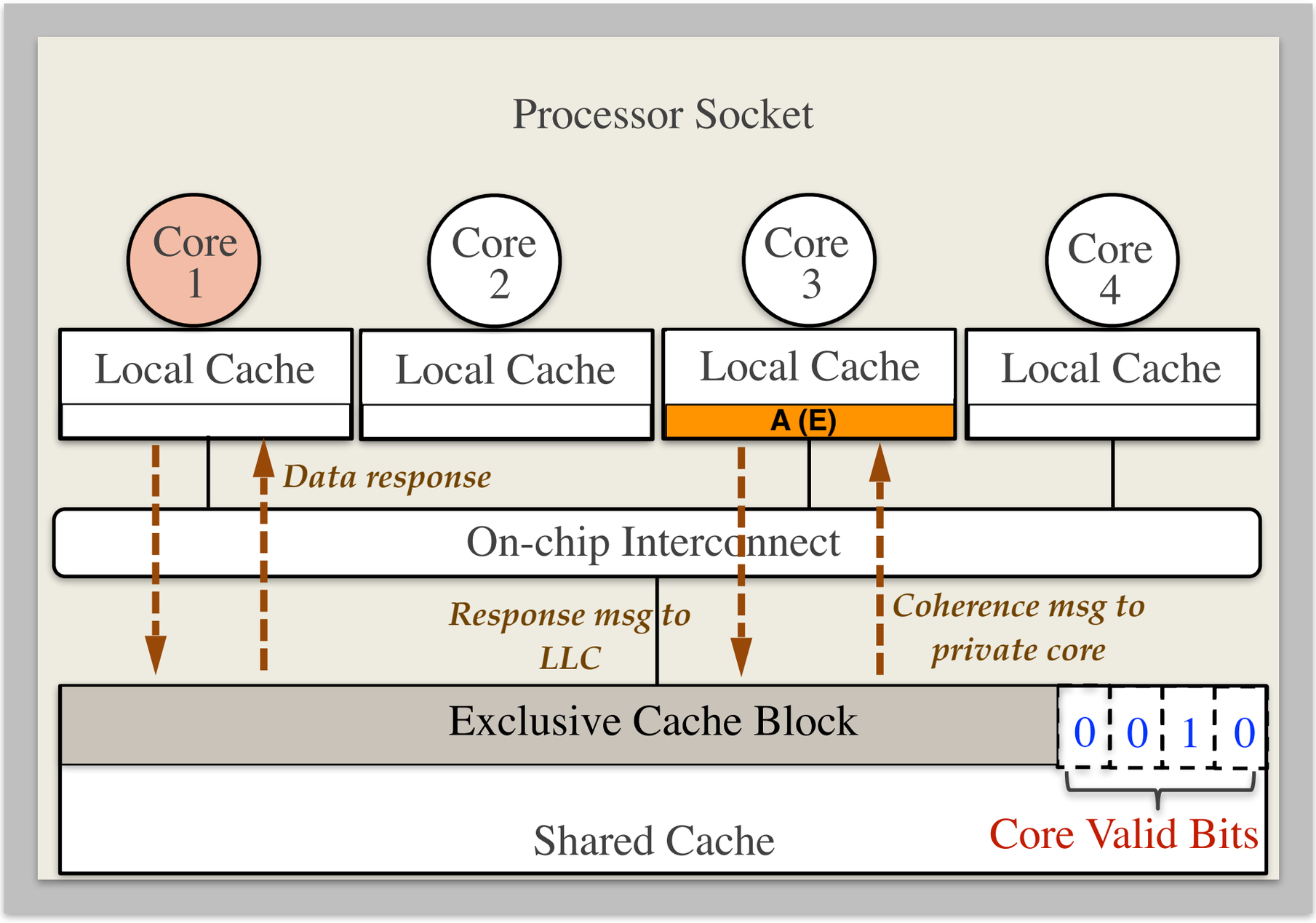
Cache Coherence – Exclusive Block



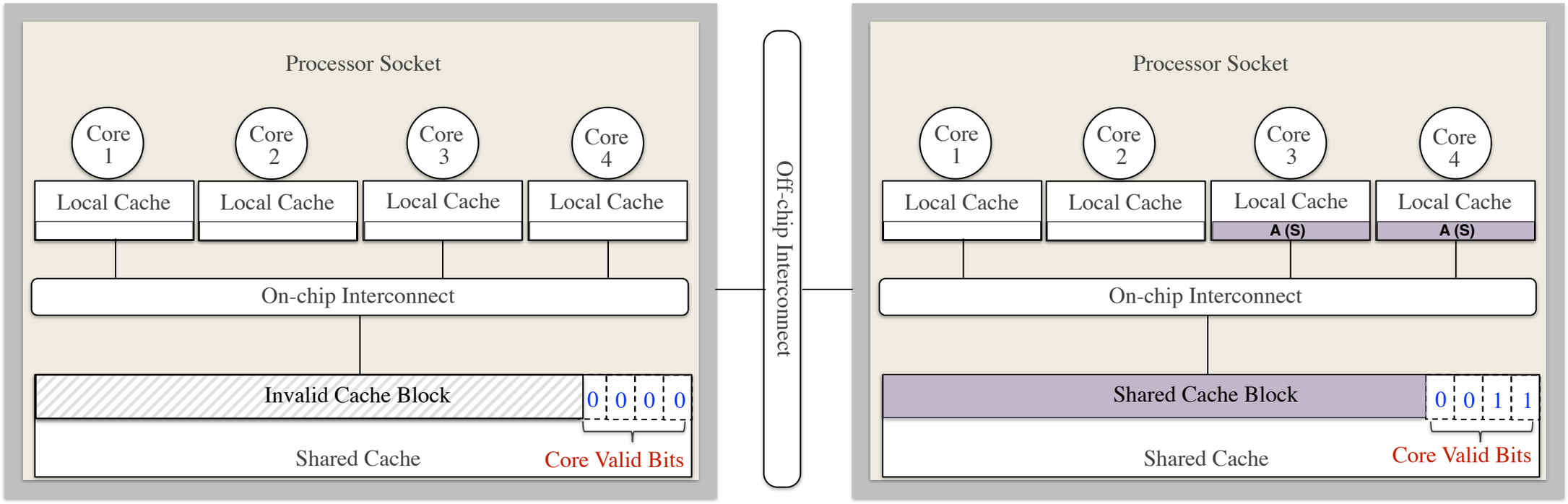
Cache Coherence – Exclusive Block



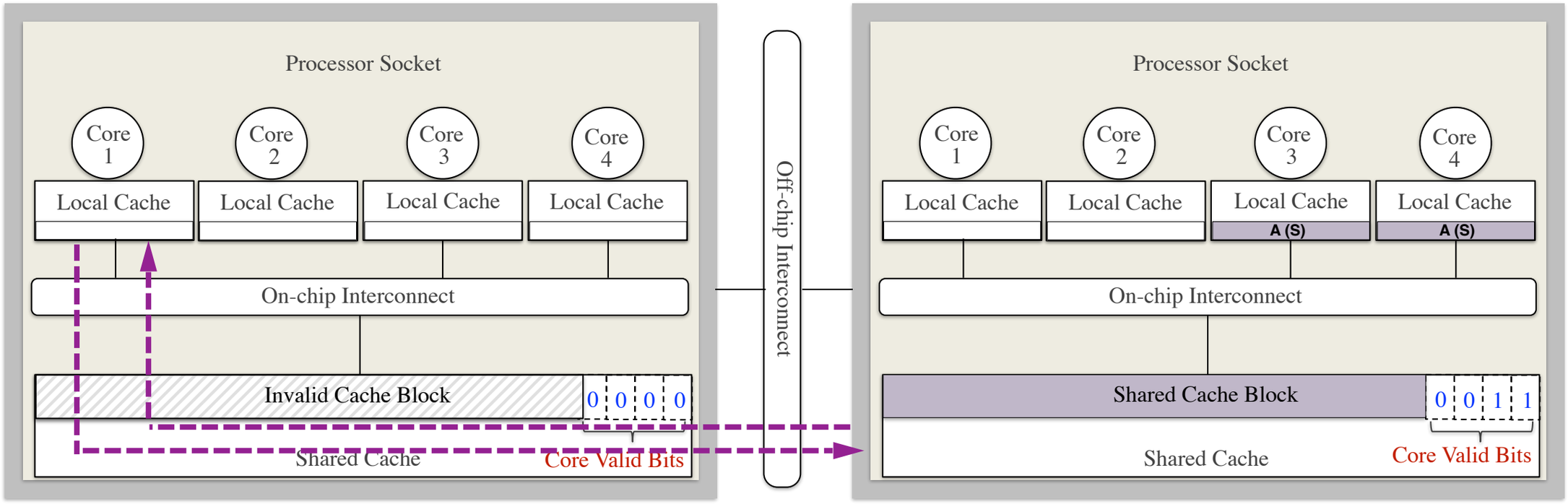
Cache Coherence – Exclusive Block



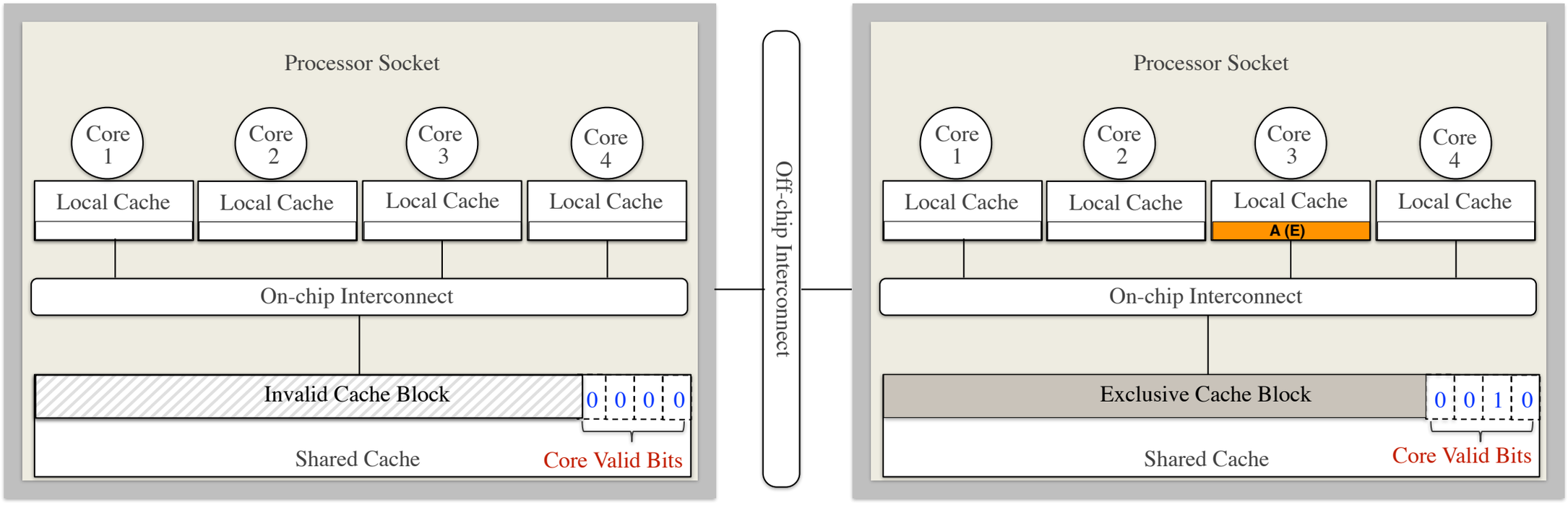
Cache Coherence – Inter-Socket



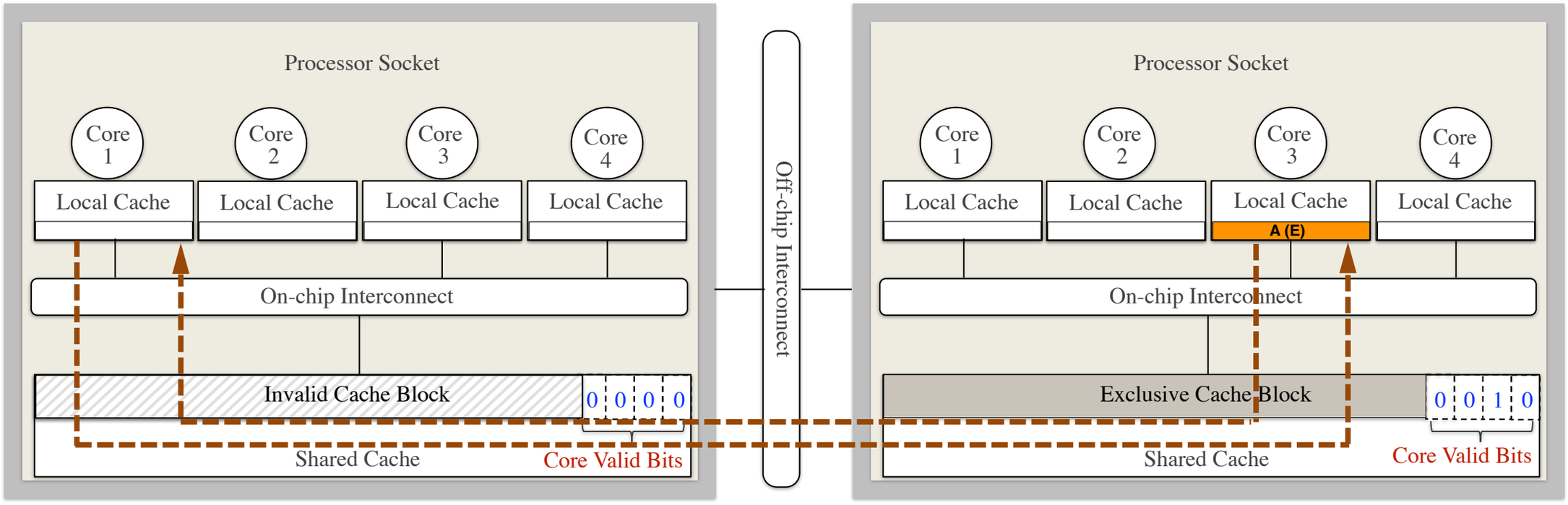
Cache Coherence – Inter-Socket



Cache Coherence – Inter-Socket

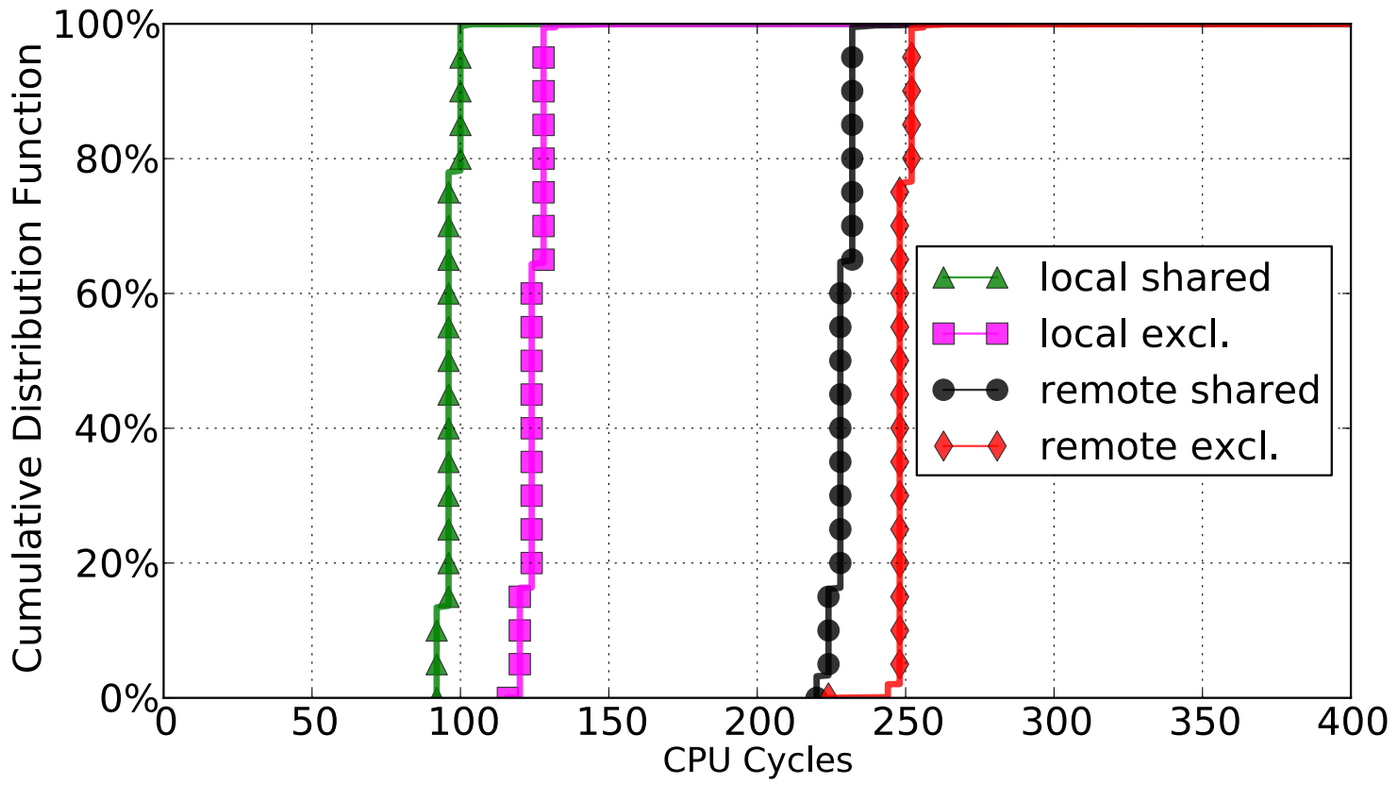


Cache Coherence – Inter-Socket



Full Latency Profiles

- Intel Xeon Dual Socket Server



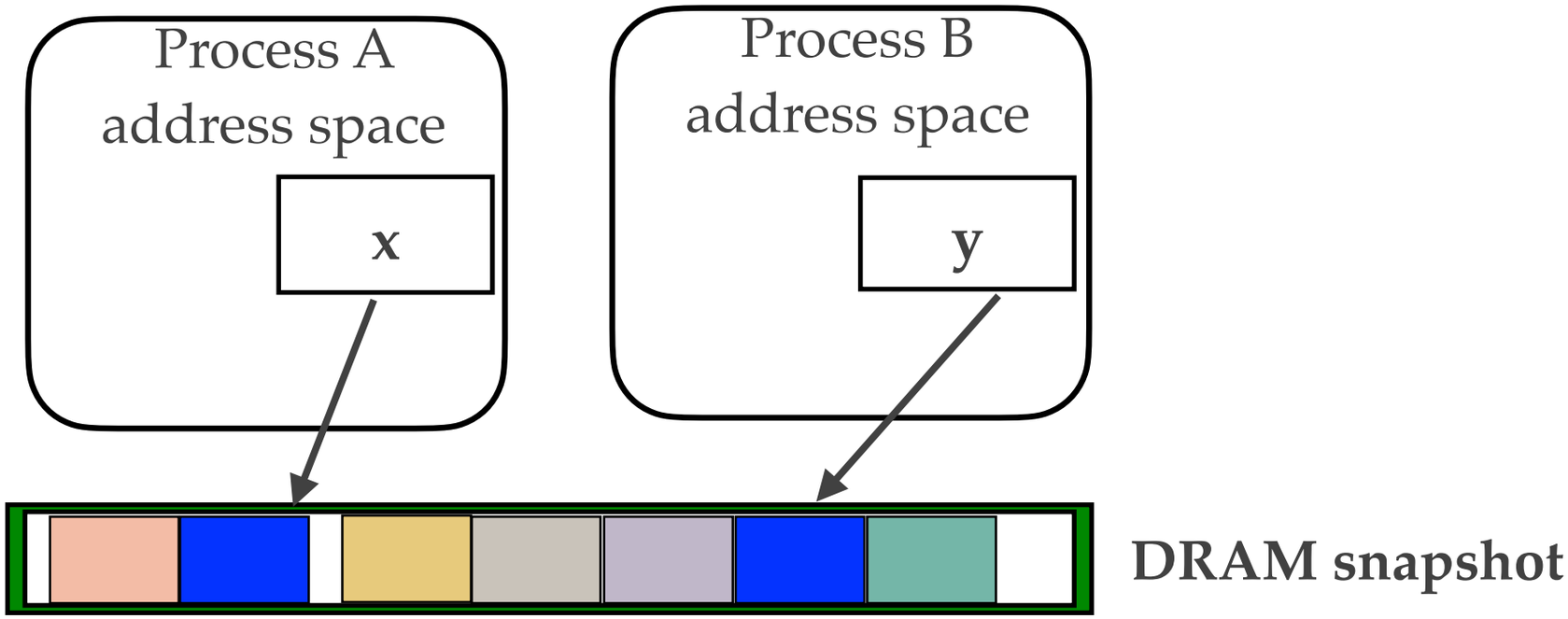
Load operation latency in (location, coherence state) combinations

How to Create Shared Memory?

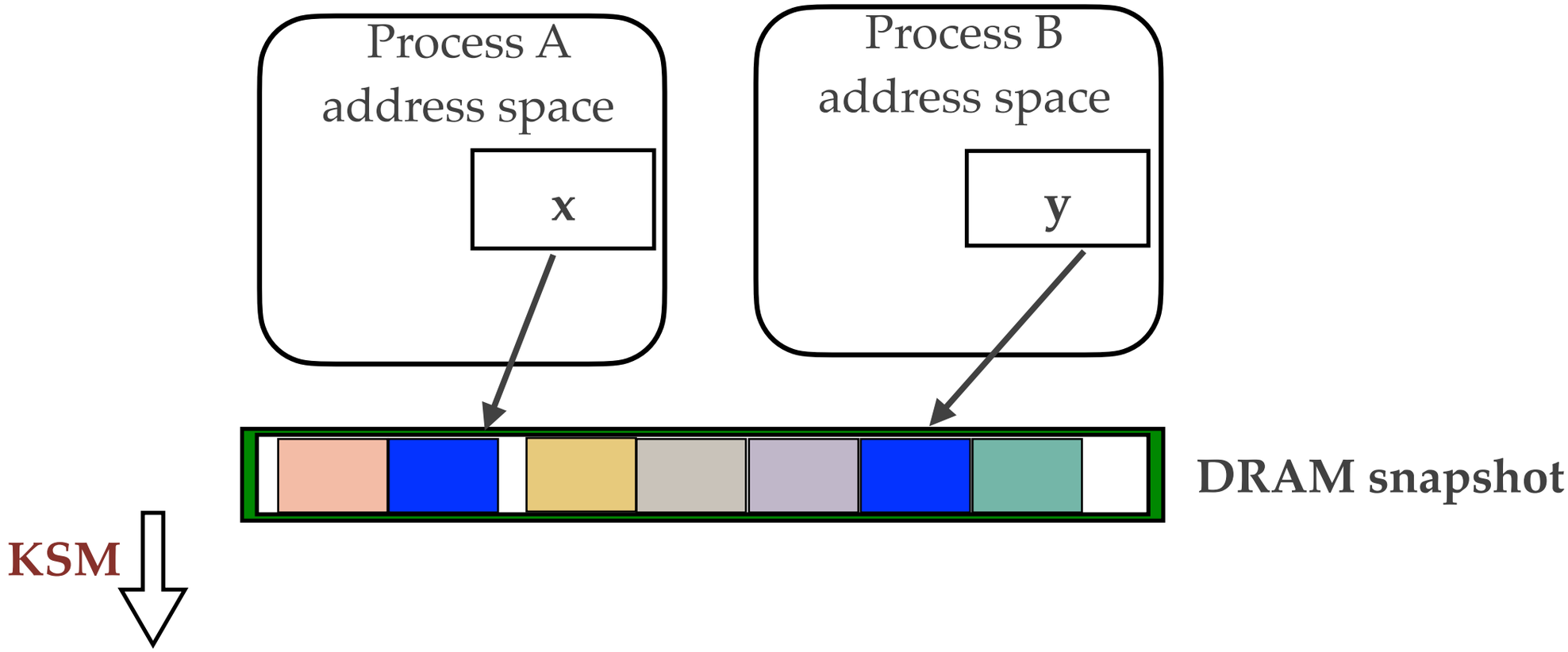
- Coherence transactions happen on shared memory
- One possible way: through shared libraries
- We use Kernel Same-page Merging (KSM)
  - OS routinely merges DRAM pages with same content
  - More stealthy

KSM Page Sharing Mechanism

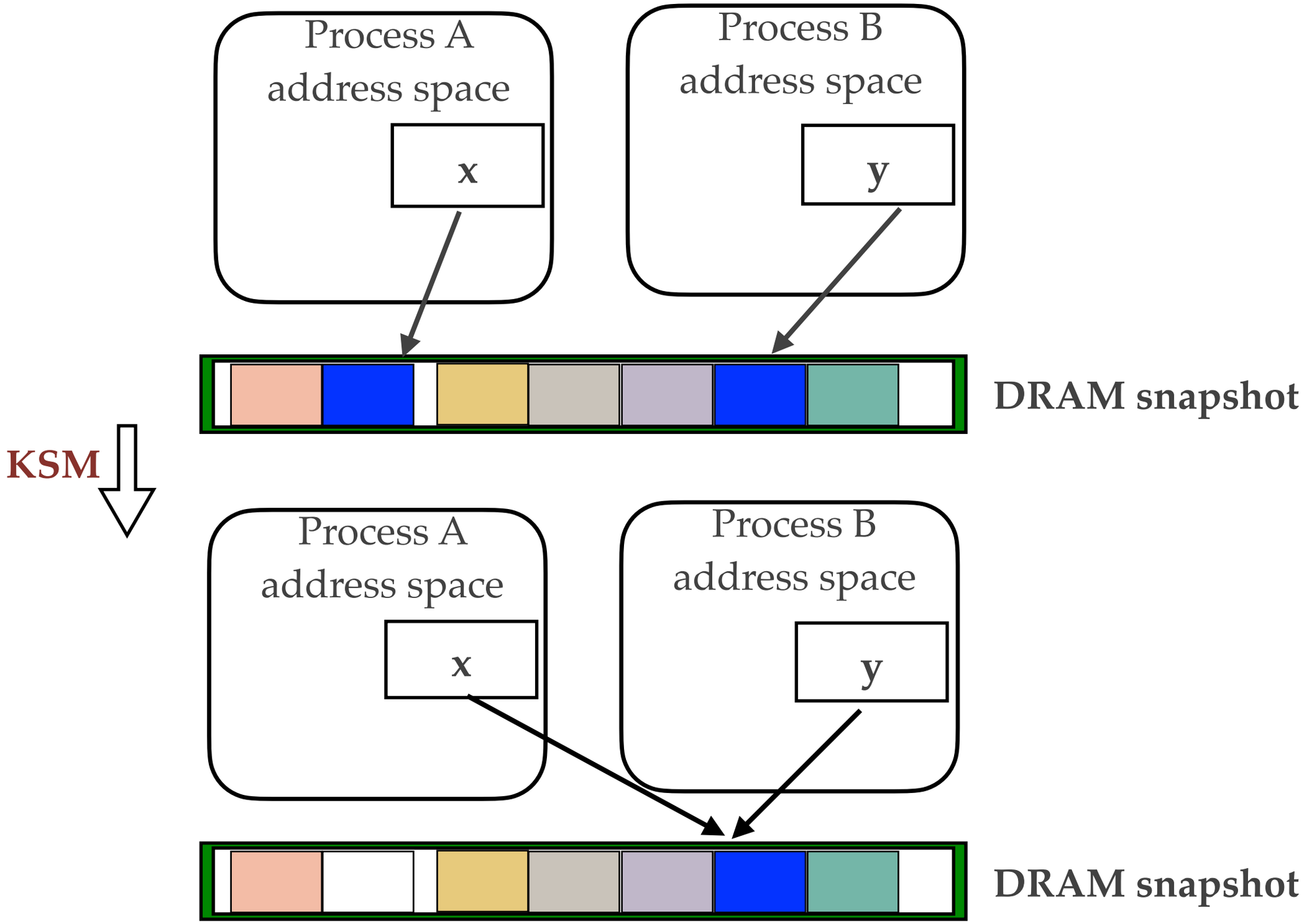
KSM Page Sharing Mechanism



KSM Page Sharing Mechanism



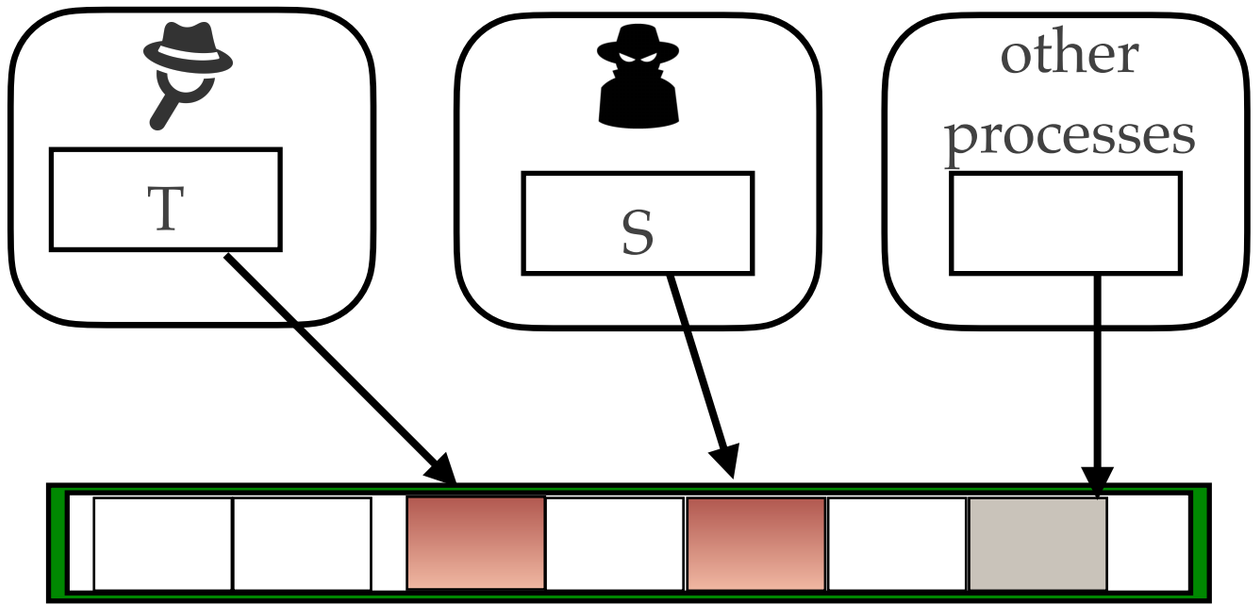
KSM Page Sharing Mechanism



Trojan and Spy Pre-transmission

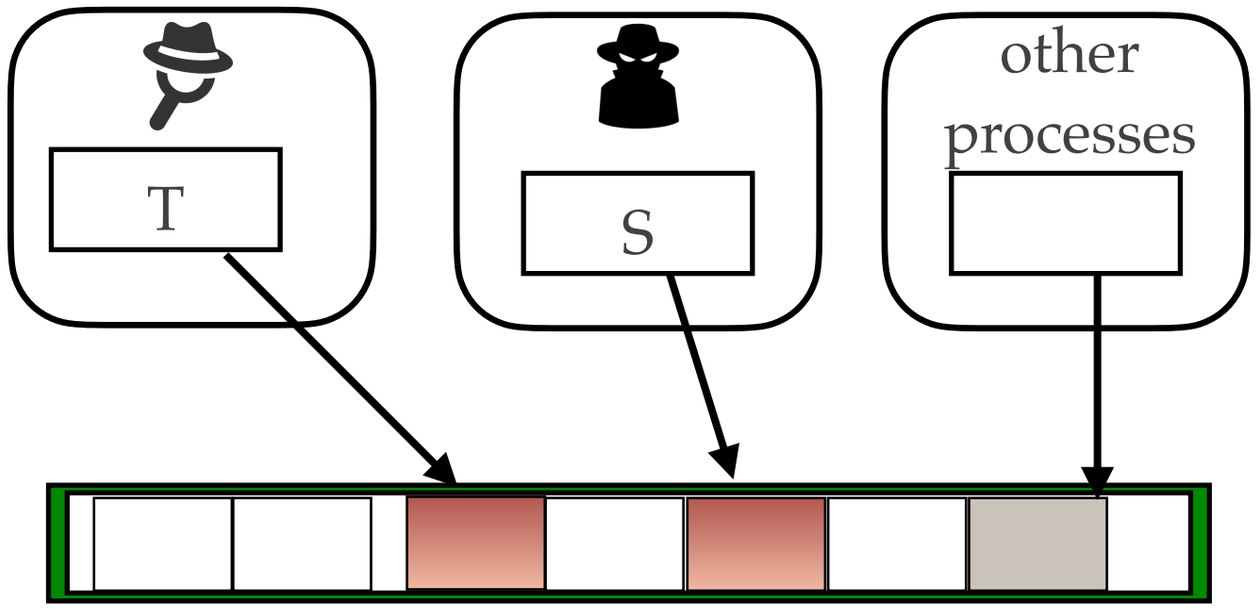
Trojan and Spy Pre-transmission

1. Trojan and Spy create pages with same pattern (content)



Trojan and Spy Pre-transmission

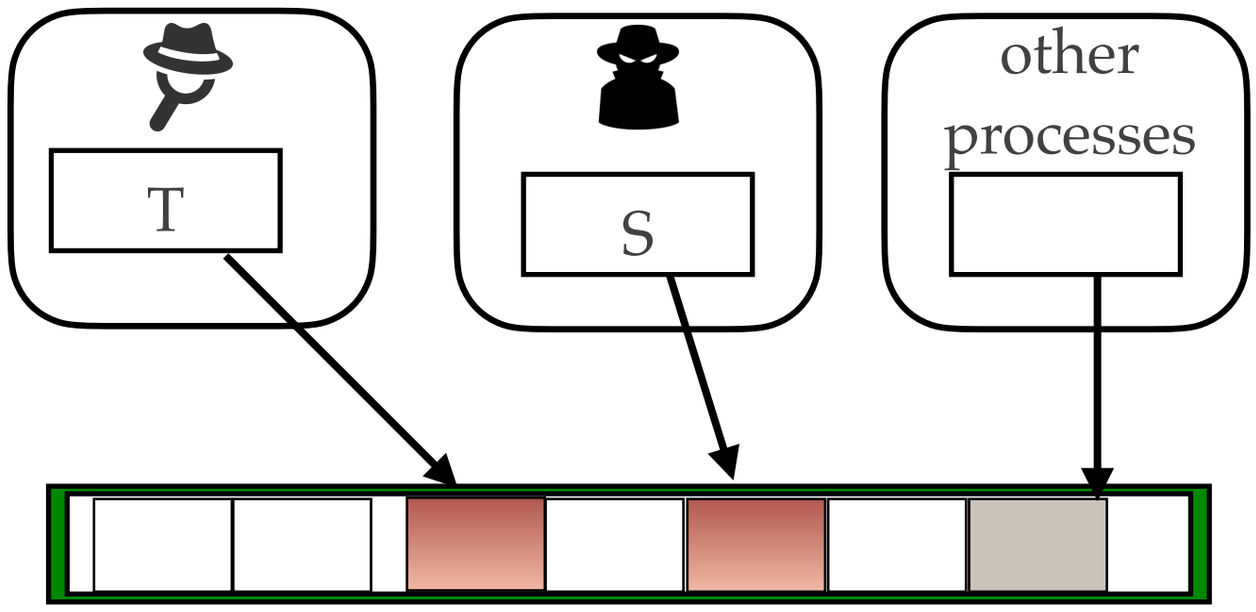
1. Trojan and Spy create pages with same pattern (content)



2. Trojan and Spy wait for KSM to merge pages

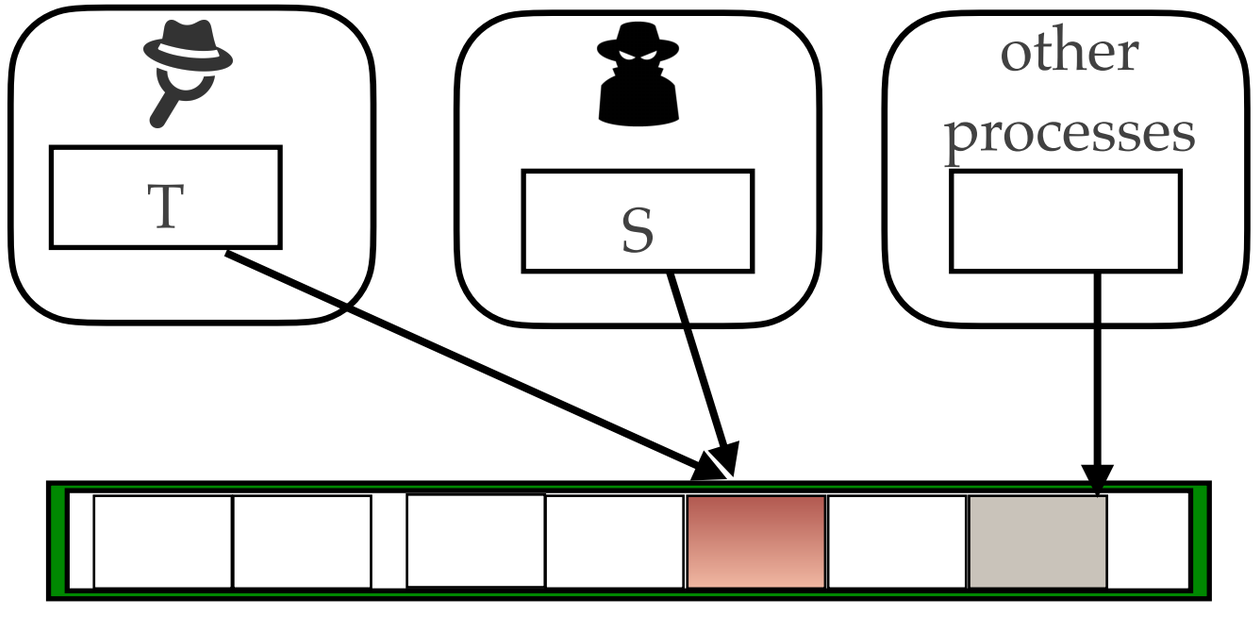
Trojan and Spy Pre-transmission

1. Trojan and Spy create pages with same pattern (content)



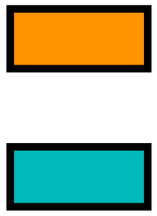
2. Trojan and Spy wait for KSM to merge pages

3. Trojan and Spy test page merging

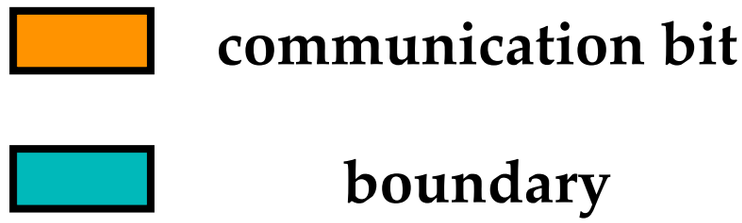


Encoding in an Information Leakage Channel

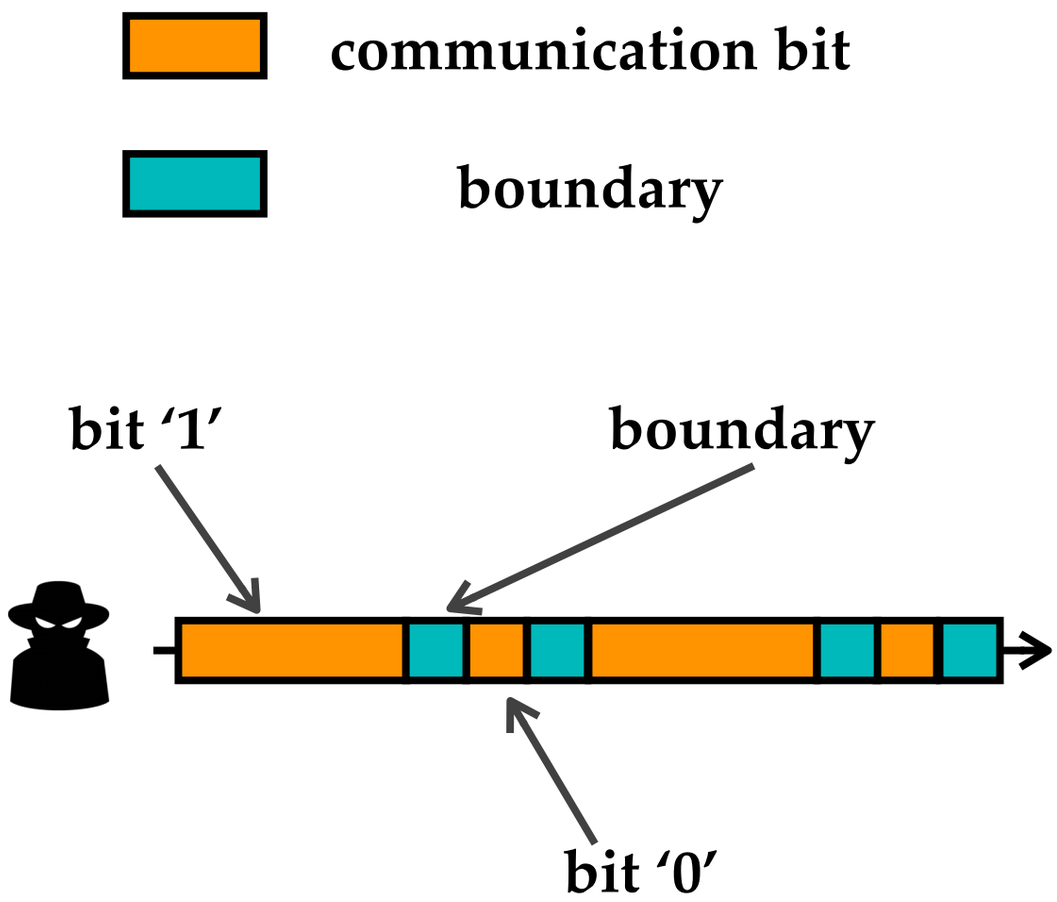
Encoding in an Information Leakage Channel



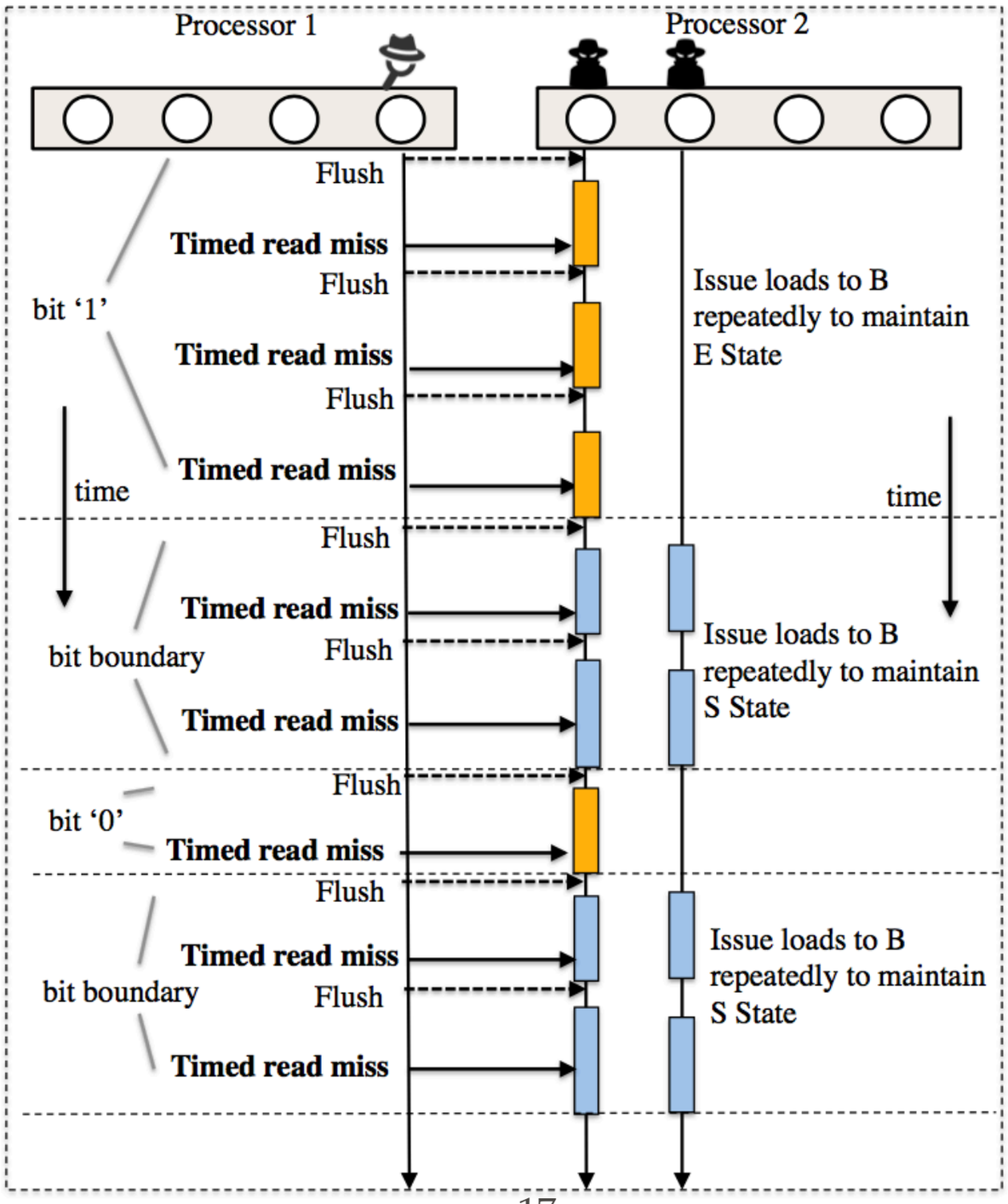
Encoding in an Information Leakage Channel



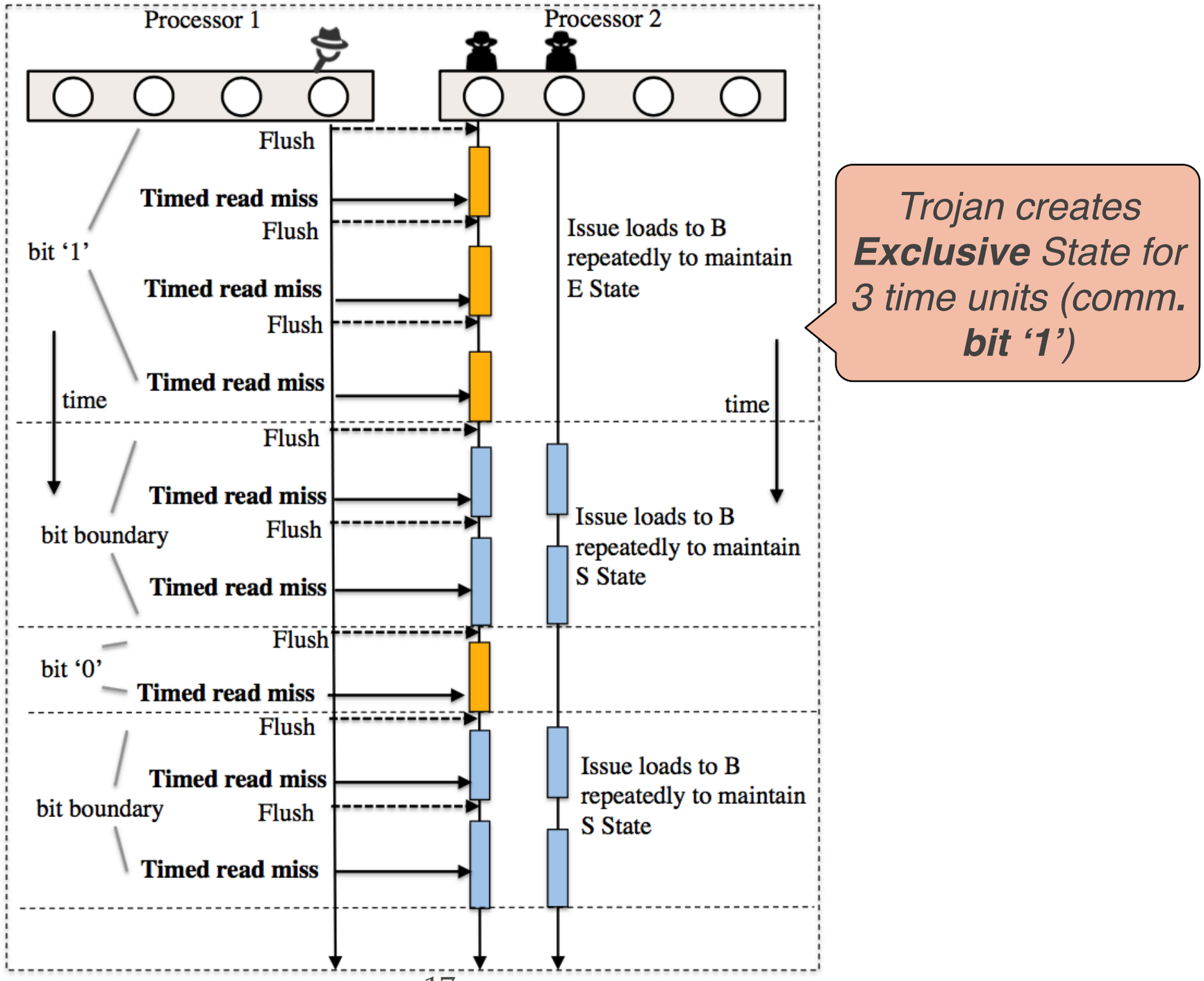
Encoding in an Information Leakage Channel



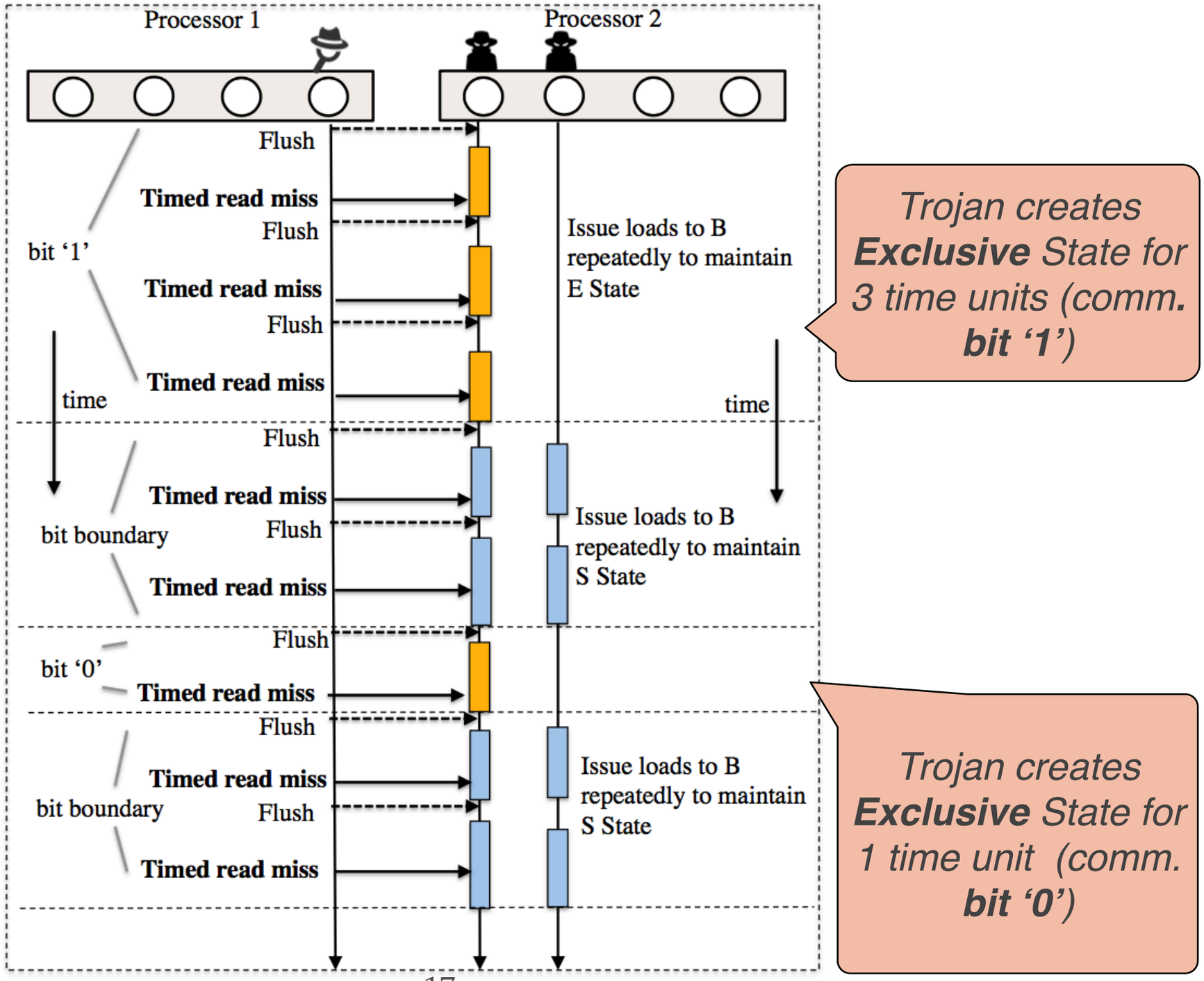
Example Communication Protocol



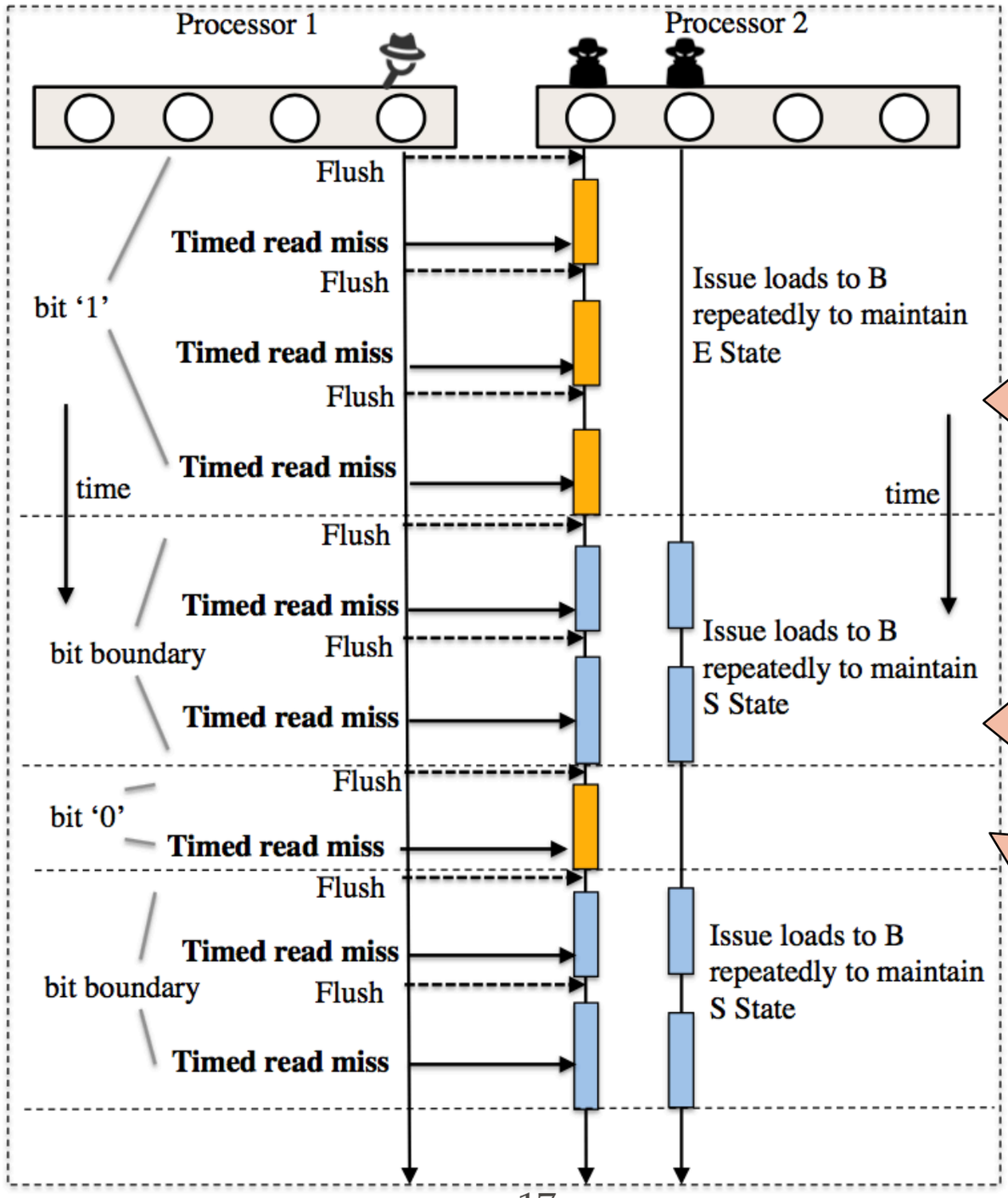
Example Communication Protocol



Example Communication Protocol



Example Communication Protocol

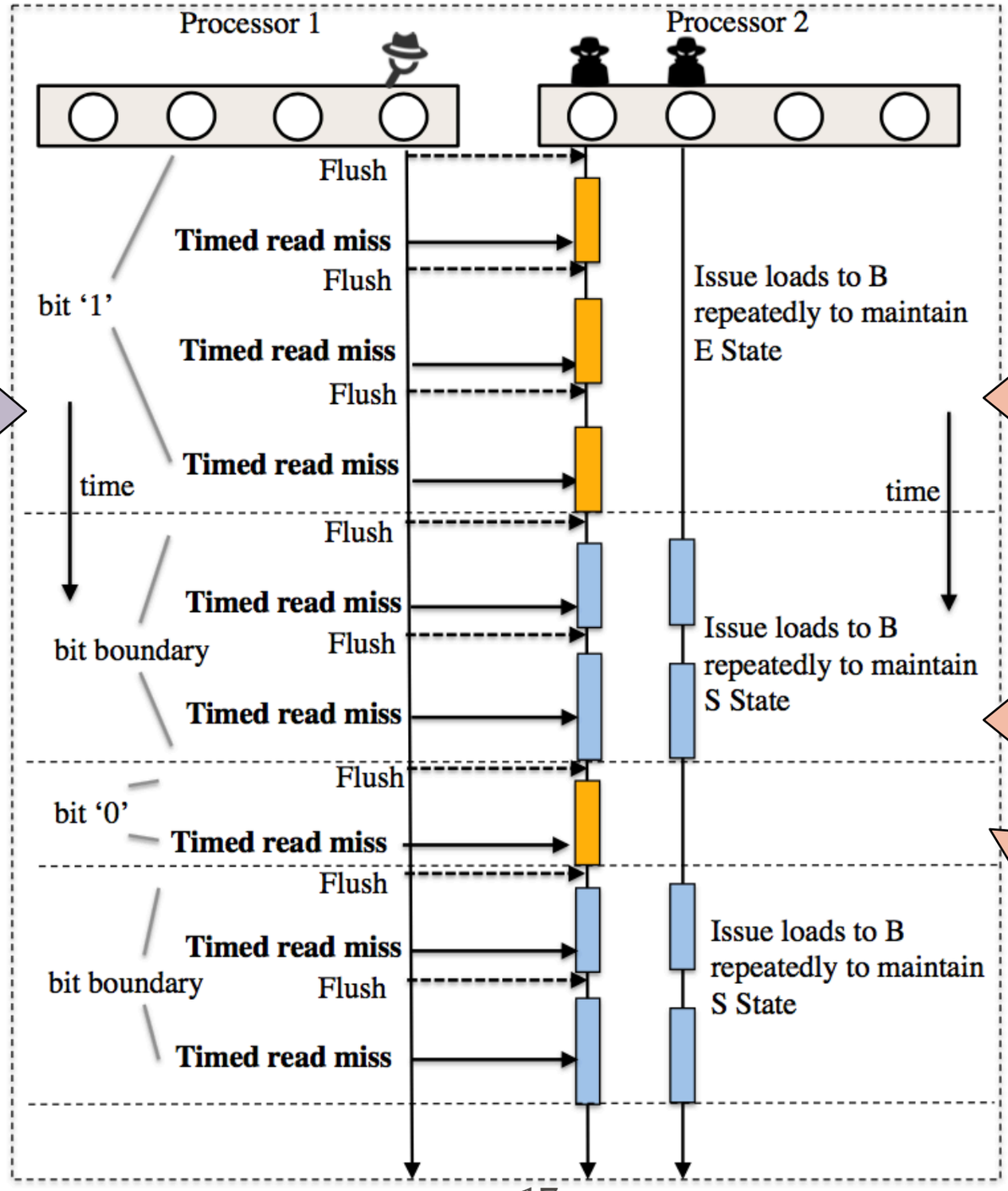


*Trojan creates **Exclusive** State for 3 time units (comm. **bit '1'**)*

*Trojan creates **Shared** State for 2 time units (boundary)*

*Trojan creates **Exclusive** State for 1 time unit (comm. **bit '0'**)*

Example Communication Protocol



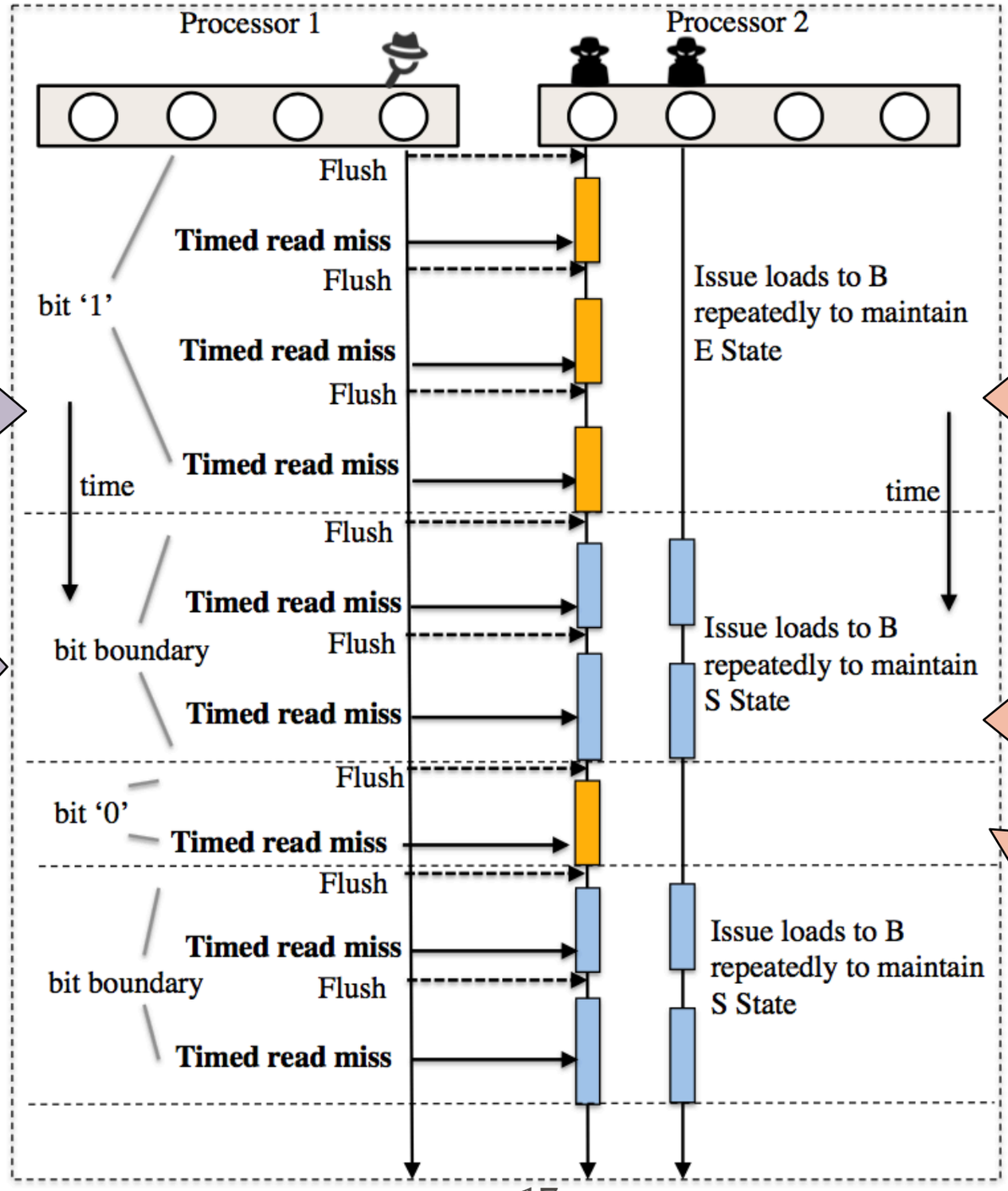
Spy sensed ***Exclusive*** *state 3 times ,* ***LONG*** *=>* ***bit '1'***

Trojan creates ***Exclusive*** *State for 3 time units (comm.* ***bit '1'****)*

Trojan creates ***Shared*** *State for 2 time units (boundary)*

Trojan creates ***Exclusive*** *State for 1 time unit (comm.* ***bit '0'****)*

Example Communication Protocol



Spy sensed ***Exclusive*** *state 3 times ,* ***LONG*** *=>* ***bit '1'***

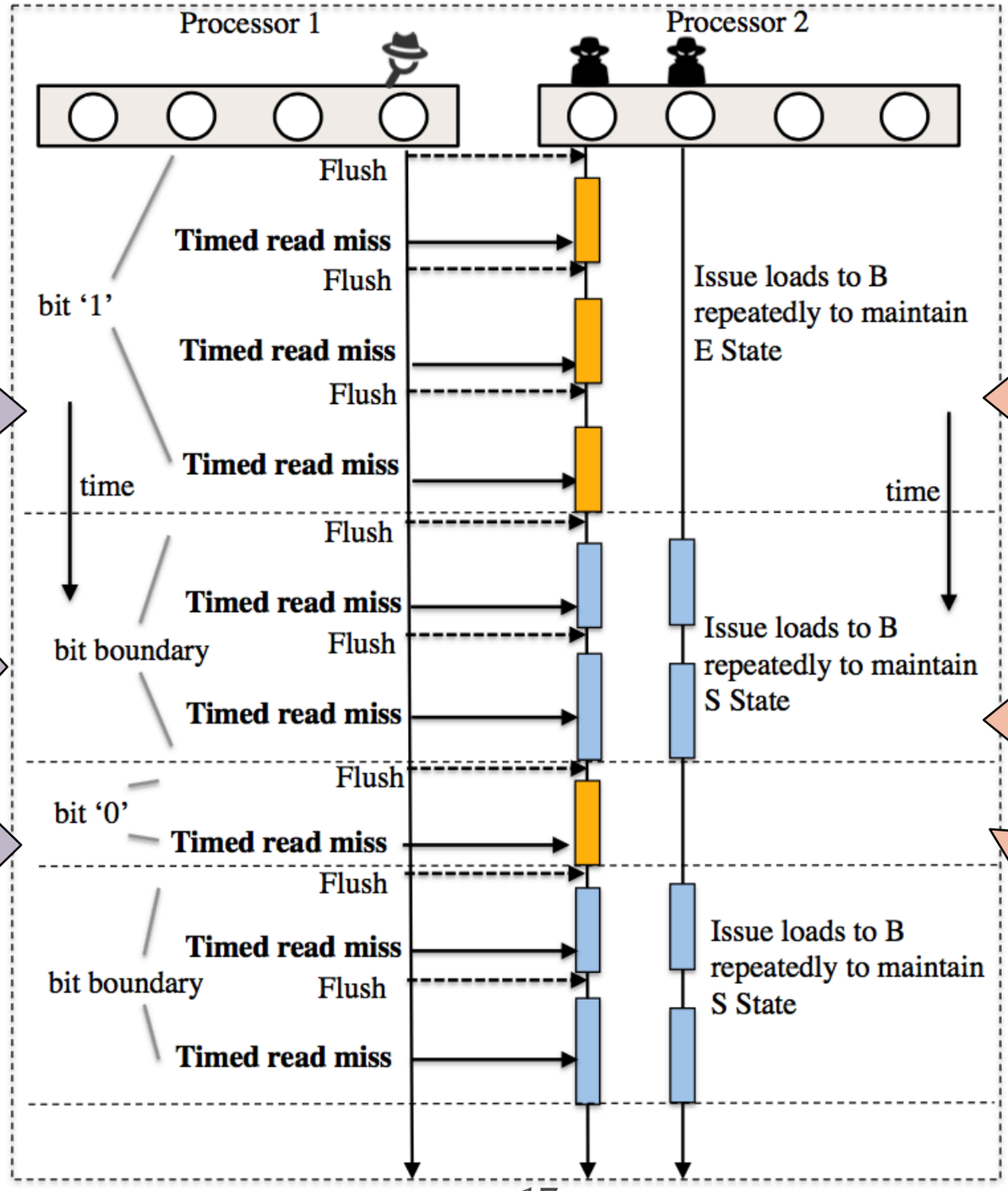
Spy sensed ***Shared*** *=>* ***boundary***

Trojan creates ***Exclusive*** *State for 3 time units (comm.* ***bit '1'****)*

Trojan creates ***Shared*** *State for 2 time units (boundary)*

Trojan creates ***Exclusive*** *State for 1 time unit (comm.* ***bit '0'****)*

Example Communication Protocol



Spy sensed ***Exclusive*** *state 3 times ,* ***LONG*** *=>* ***bit '1'***

Spy sensed ***Shared*** *=>* ***boundary***

Spy sensed ***Exclusive*** *state 1 time,* ***SHORT*** *=>* ***bit '0'***

Trojan creates ***Exclusive*** *State for 3 time units (comm.* ***bit '1'****)*

Trojan creates ***Shared*** *State for 2 time units (boundary)*

Trojan creates ***Exclusive*** *State for 1 time unit (comm.* ***bit '0'****)*

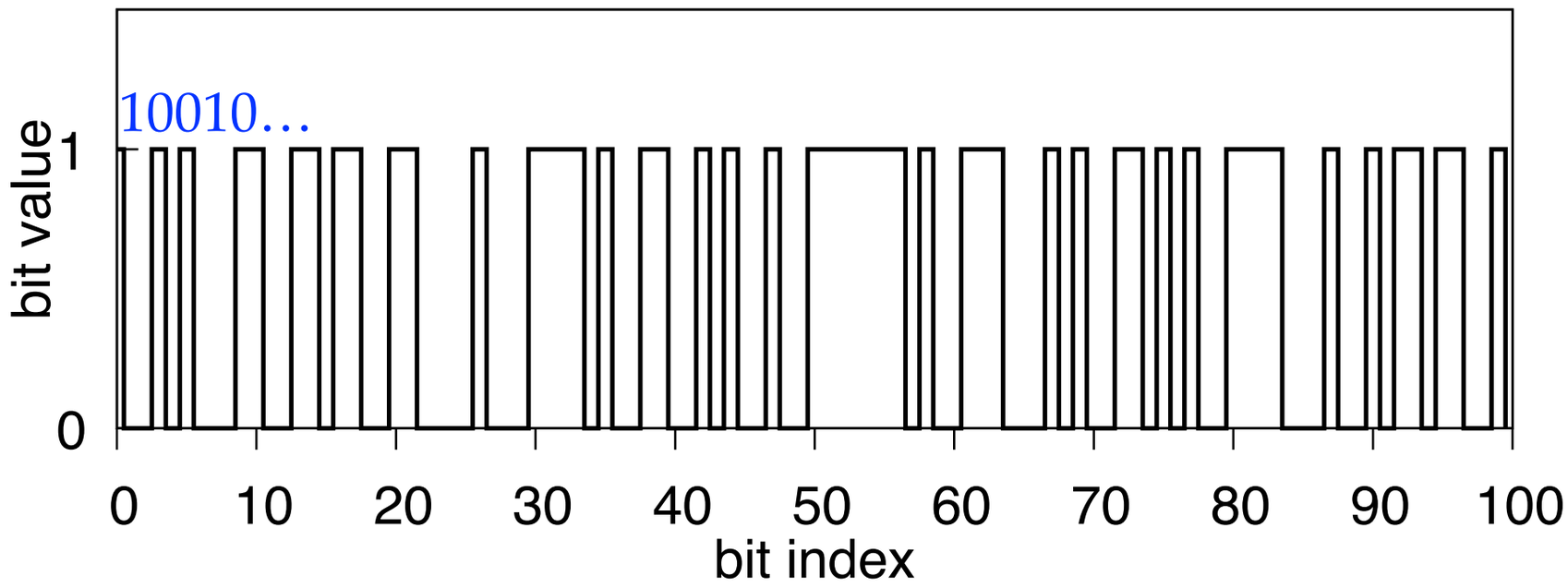
The Attack Scenarios

- Differing latency profiles
  - Local Exclusive
  - Local Shared
  - Remote Exclusive
  - Remote Shared
- Binary channel encoding - (bit comm., boundary)

Location and Coherence State	Notation	Number of Trojan threads
(Local Excl., Local Shared)	$LExcl_{comm}$-$LShared_{bound}$	2 (local)
(Remote Excl., Remote Shared)	$RExcl_{comm}$-$RShared_{bound}$	2 (remote)
(Remote Excl., Local Excl.)	$RExcl_{comm}$-$LExcl_{bound}$	2 (1 local, 1 remote)
(Remote Excl., Local Shared)	$RExcl_{comm}$-$LShared_{bound}$	3 (2 local, 1 remote)
(Remote Shared, Local Excl.)	$RShared_{comm}$-$LExcl_{bound}$	3 (1 local, 2 remote)
(Remote Shared, Local Shared)	$RShared_{comm}$-$LShared_{bound}$	4 (2 local, 2 remote)

Covert Channel through Coherence States

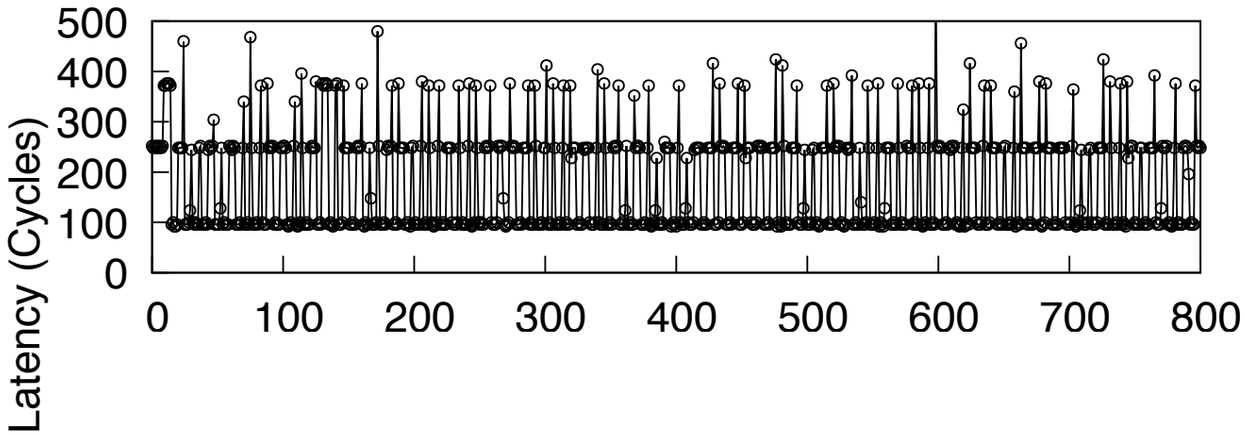
- Trojan's transmitted bits



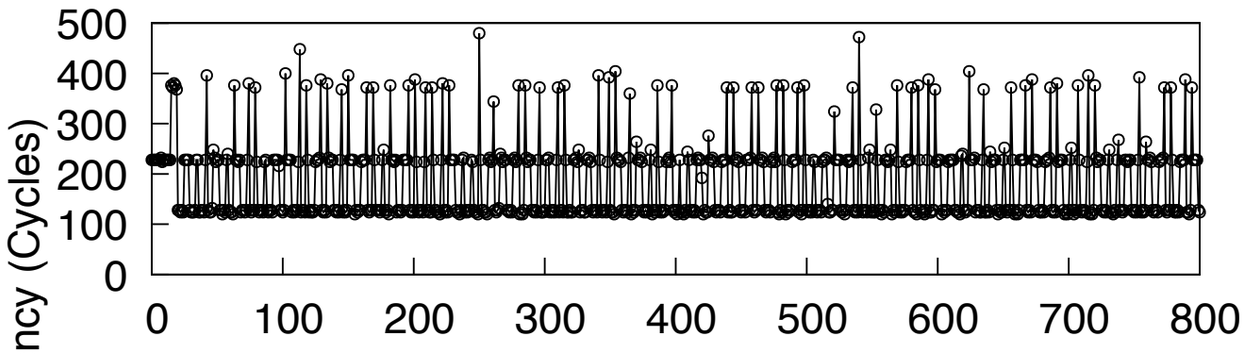
Bit pattern (100 bits) covertly transmitted by the trojan

Covert Channel through Coherence States

- Spy's reception



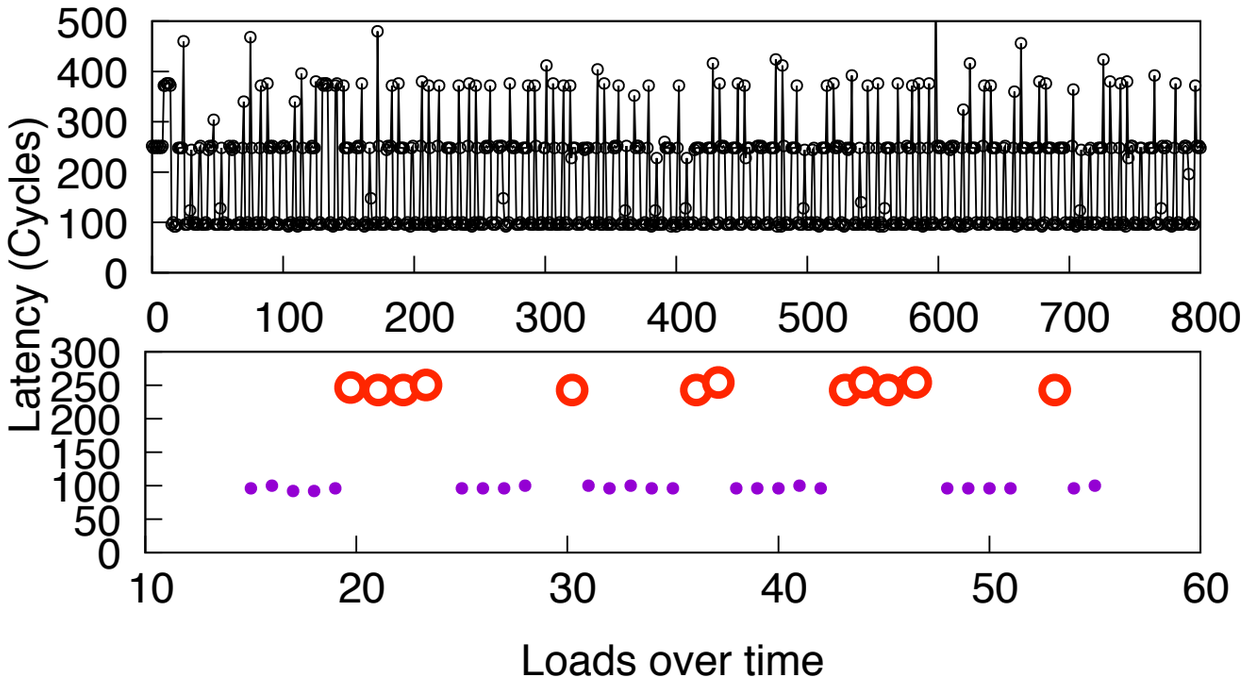
$RExcl_{comm}$-$LShared_{bound}$



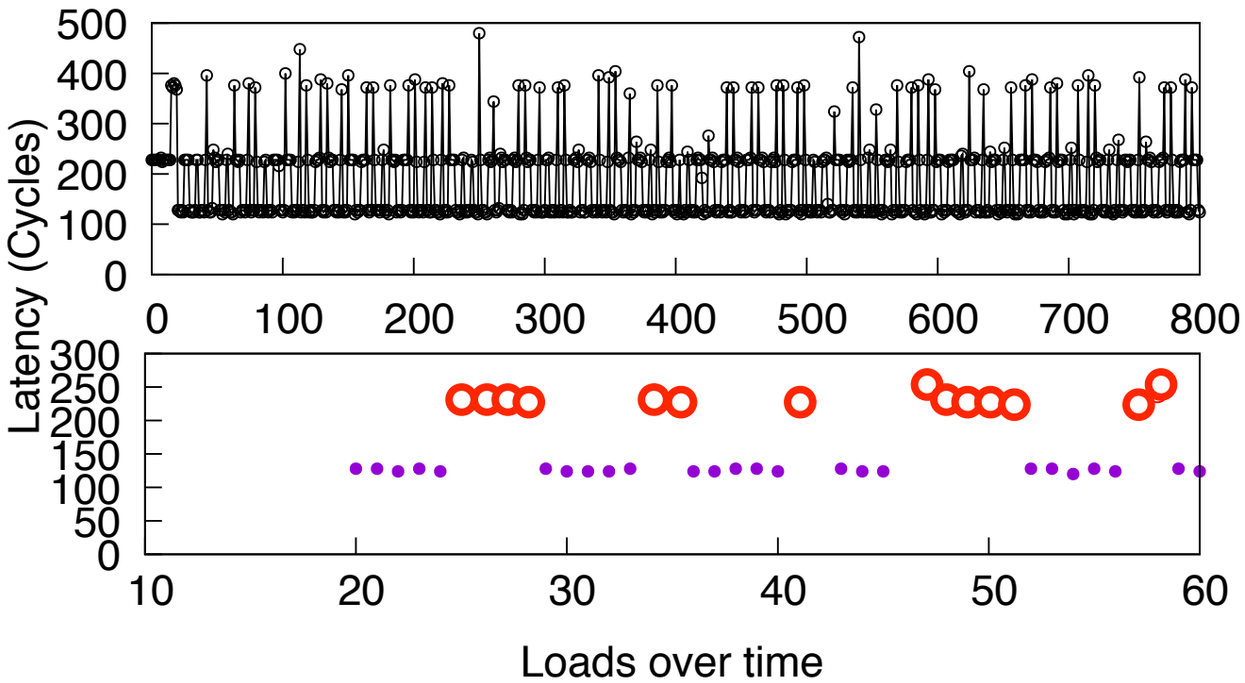
$RShared_{comm}$-$LExcl_{bound}$

Covert Channel through Coherence States

- Spy's reception



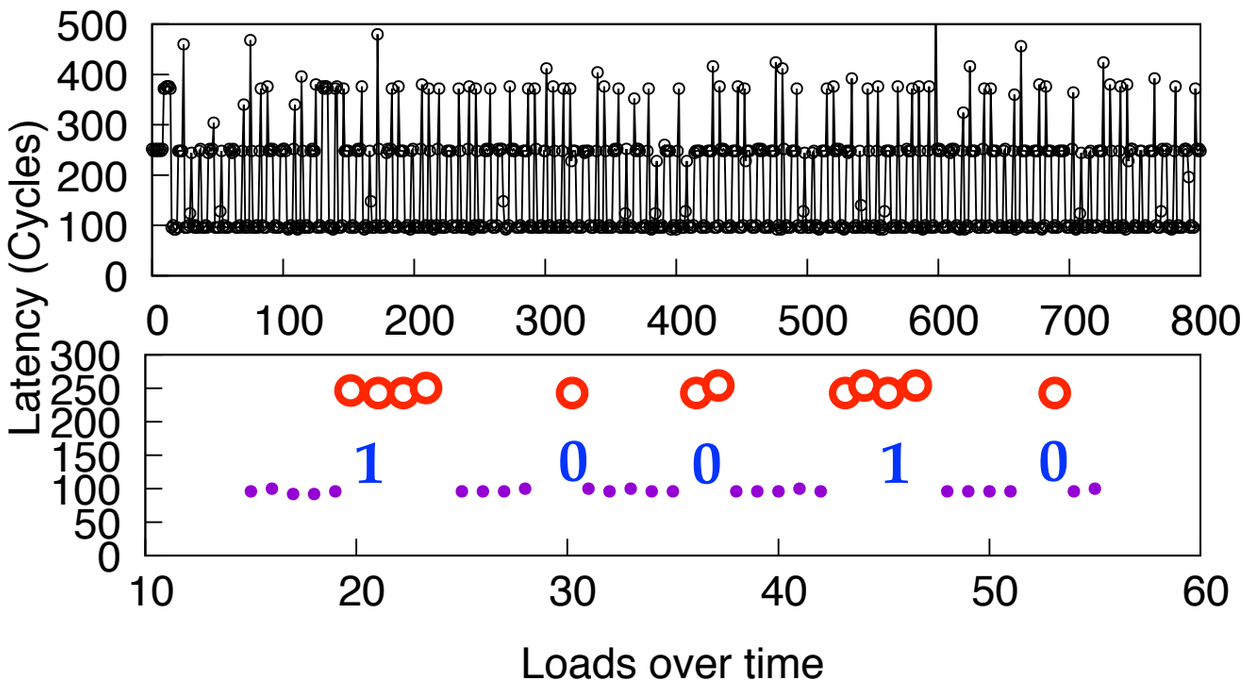
RExcl$_{comm}$-LShared$_{bound}$



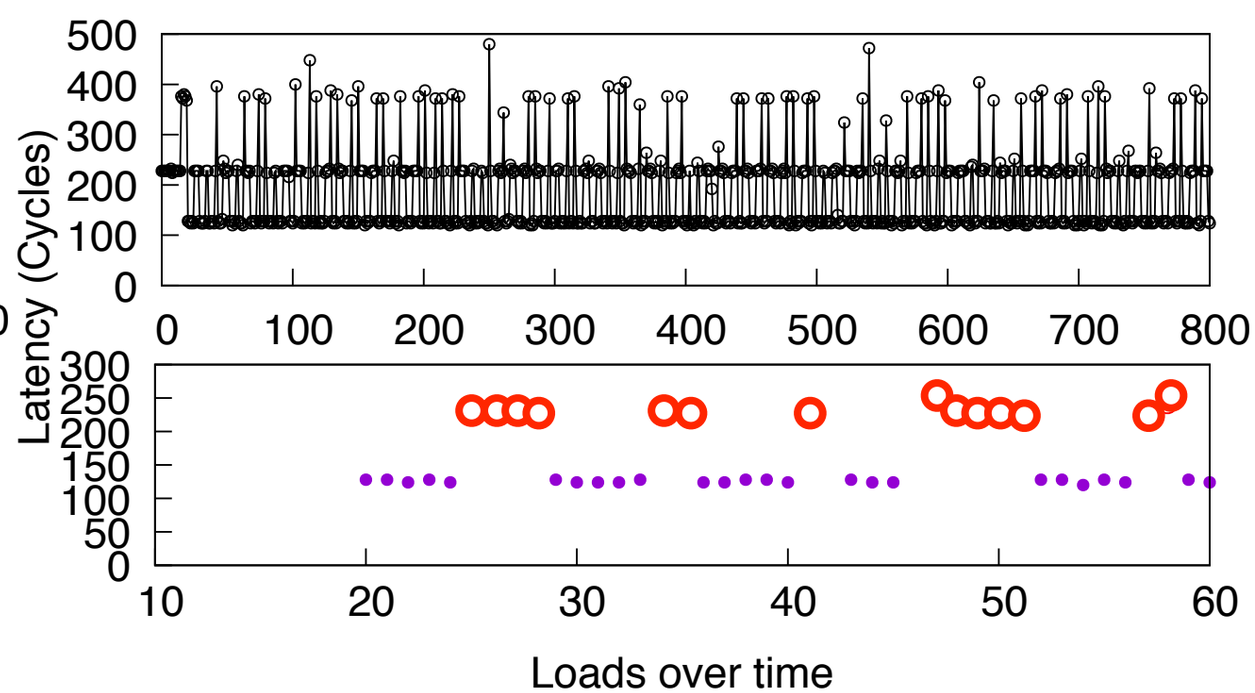
RShared$_{comm}$-LExcl$_{bound}$

Covert Channel through Coherence States

- Spy's reception



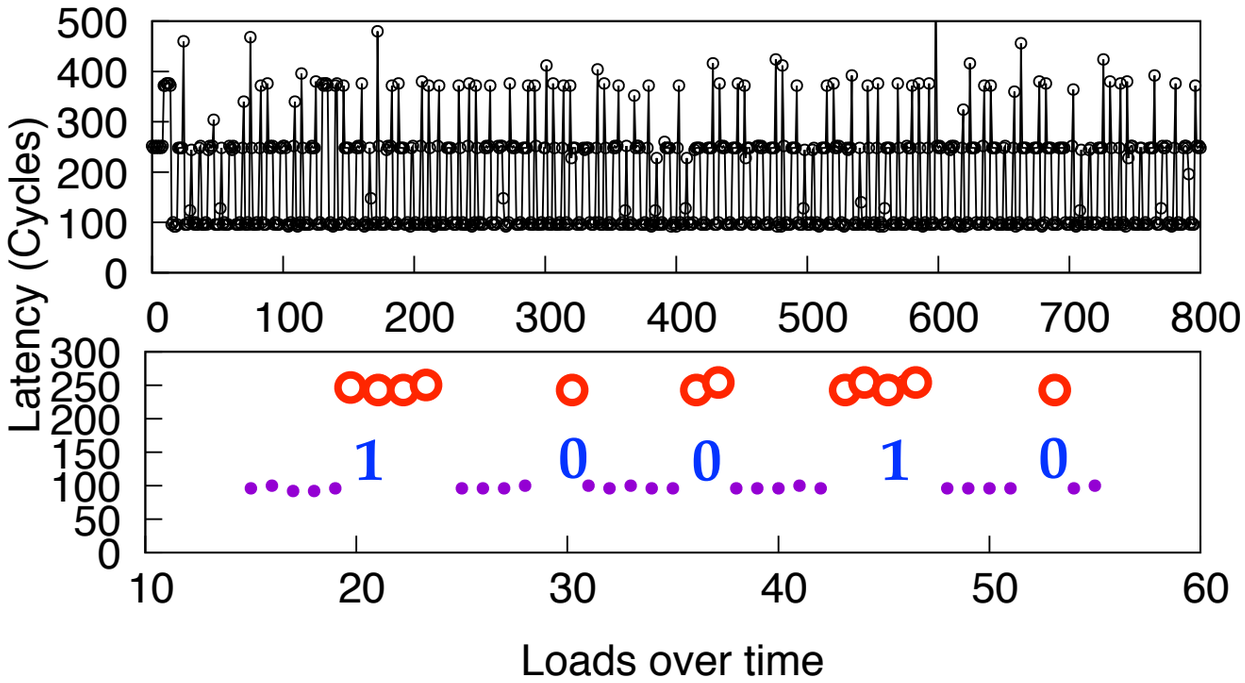
RExcl$_{comm}$-LShared$_{bound}$



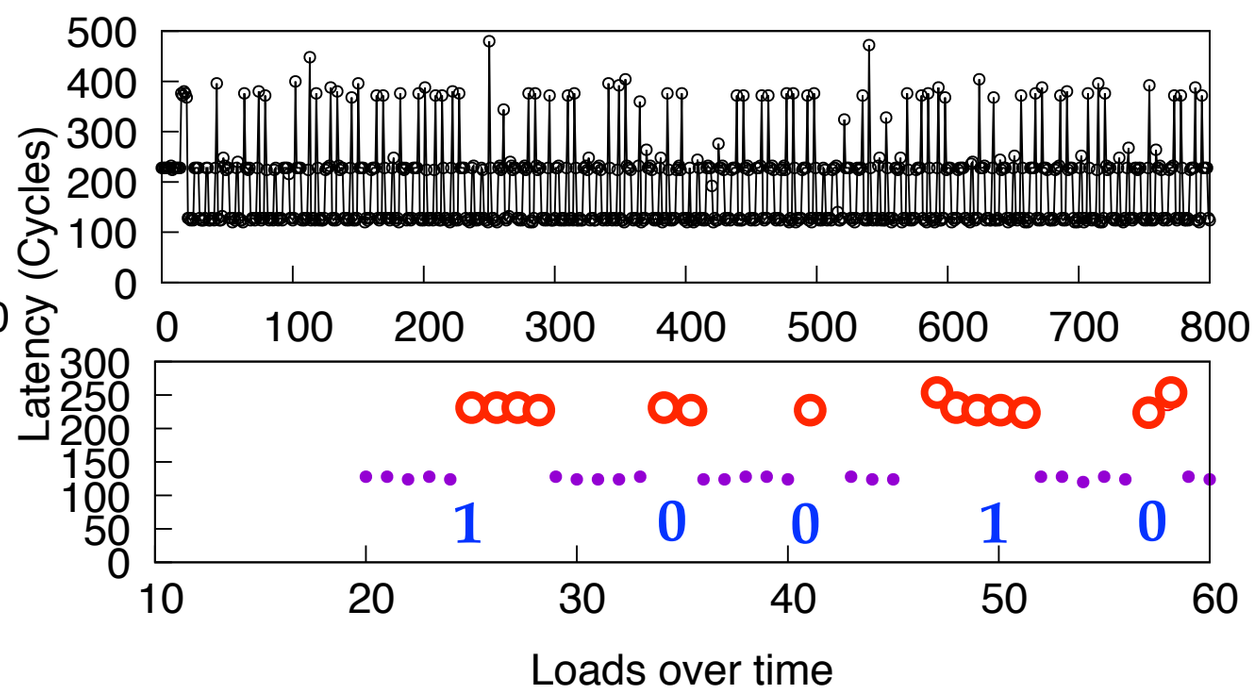
RShared$_{comm}$-LExcl$_{bound}$

Covert Channel through Coherence States

- Spy's reception



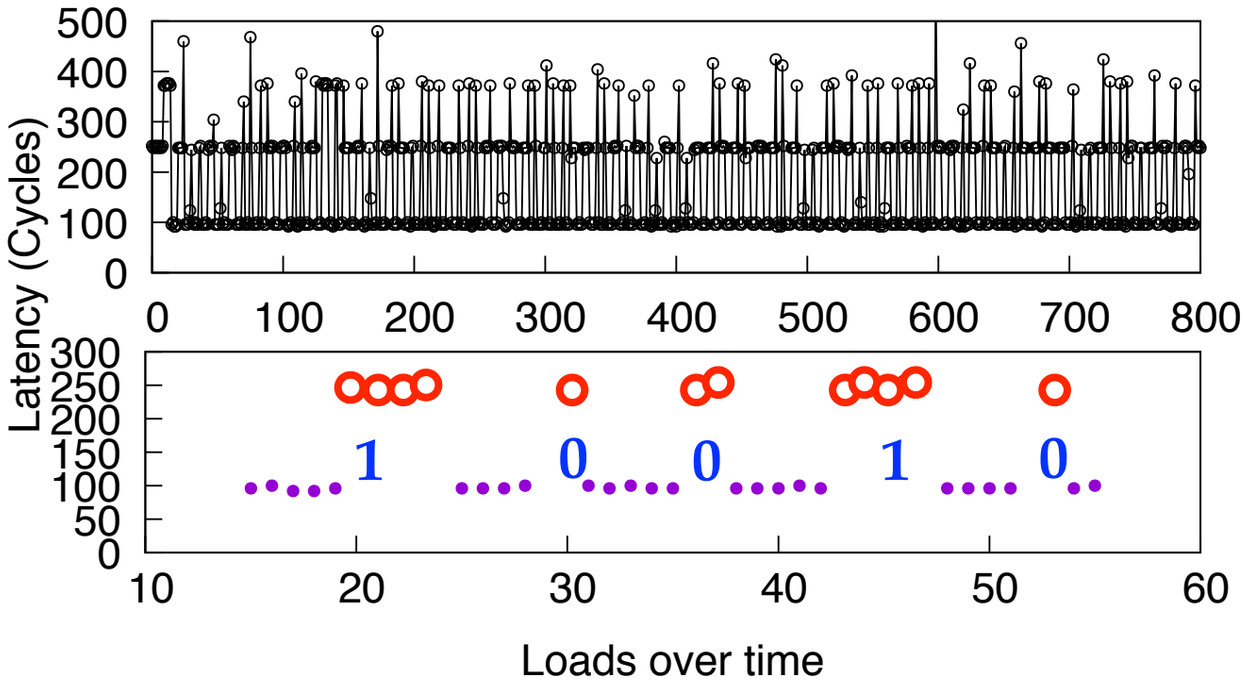
RExcl$_{comm}$-LShared$_{bound}$



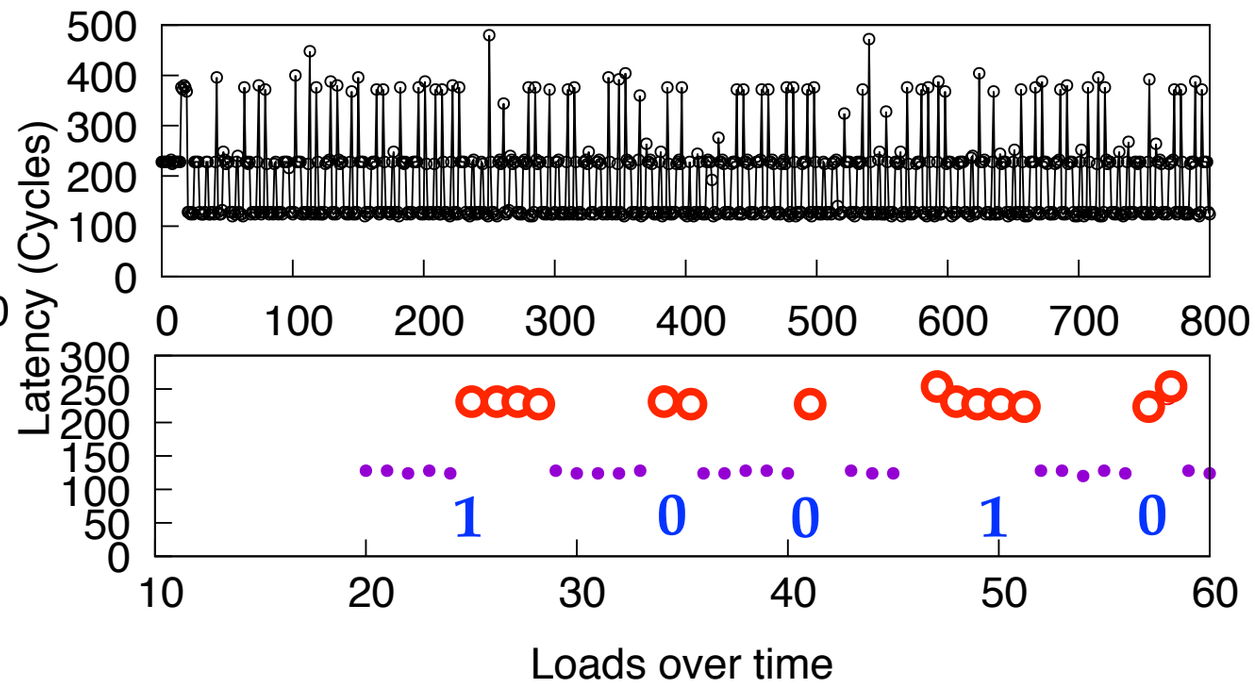
RShared$_{comm}$-LExcl$_{bound}$

Covert Channel through Coherence States

- Spy's reception



RExcl$_{comm}$-LShared$_{bound}$



RShared$_{comm}$-LExcl$_{bound}$

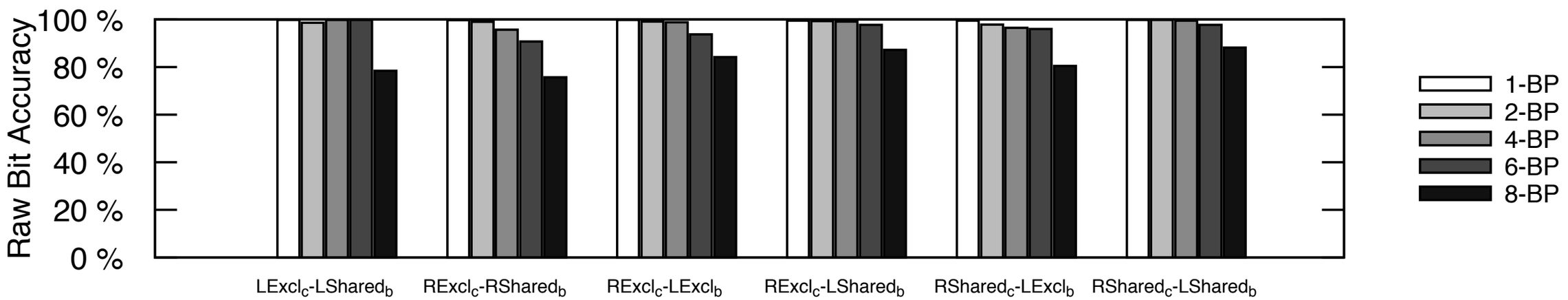
Achieved bitrate: 700Kbps

Sensitivity to External Noise

- Additional noise
  - Run memory intensive background threads (Kernel Build)
  - Vary number of threads from 1~8

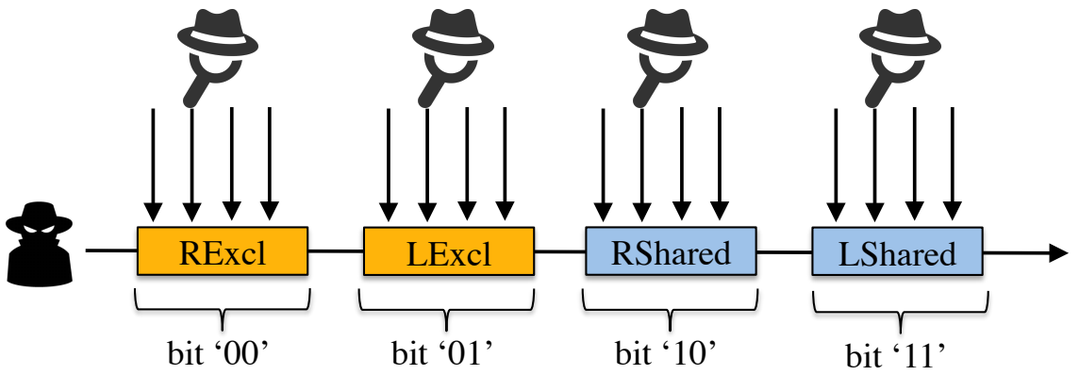
Sensitivity to External Noise

- Additional noise
  - Run memory intensive background threads (Kernel Build)
  - Vary number of threads from 1~8

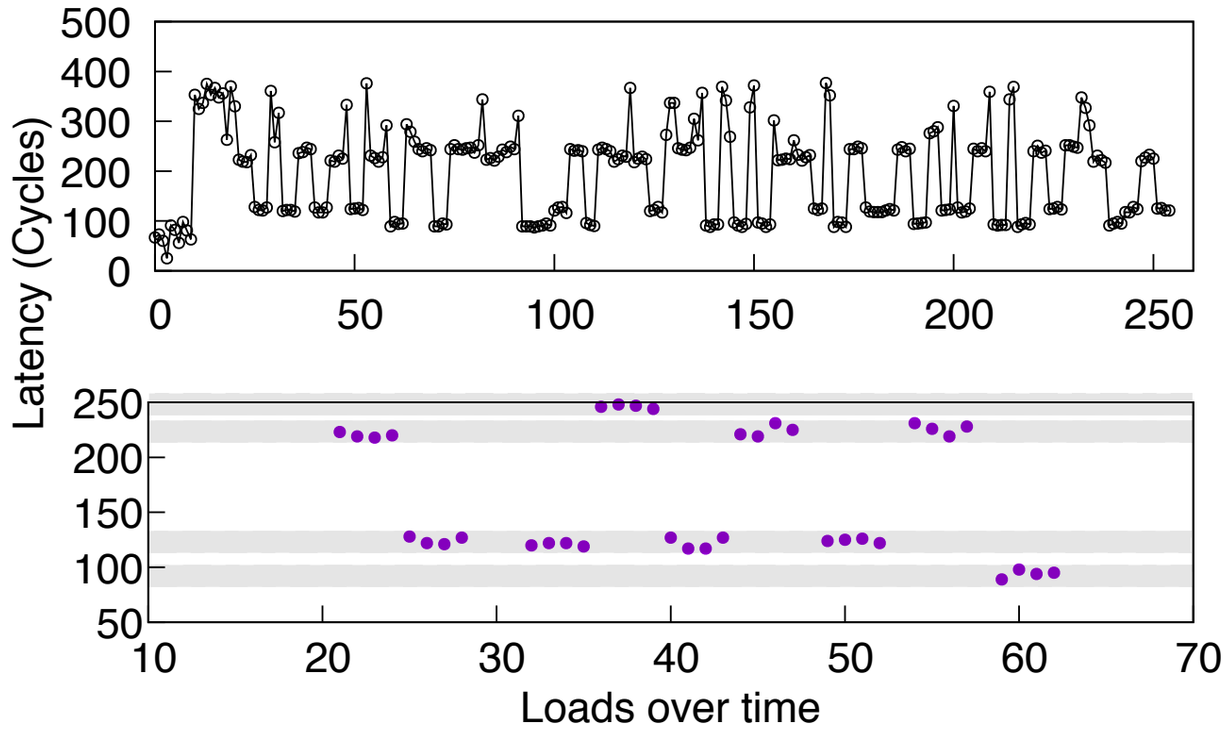


Multi-bit Symbol Transmission

- With 4 latency values, we can create 2-bit symbols
  - Each latency value represents a symbol
  - Much higher transmission bitrate



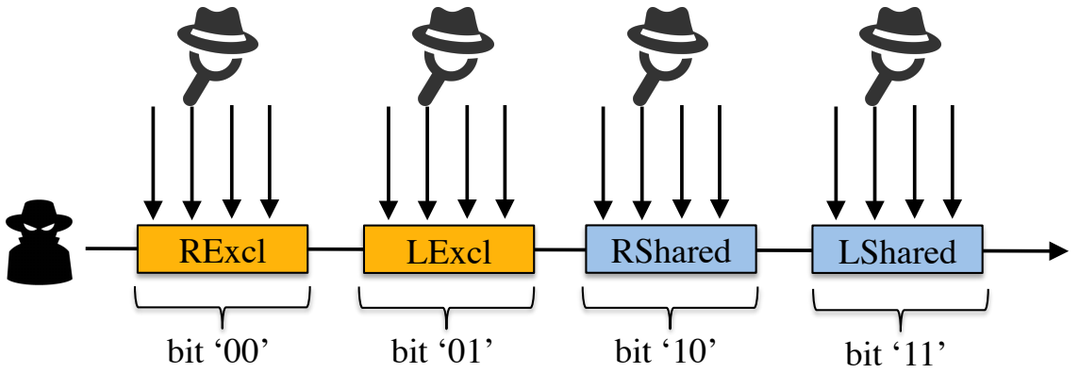
Multi-bit Symbol Transmission Scheme



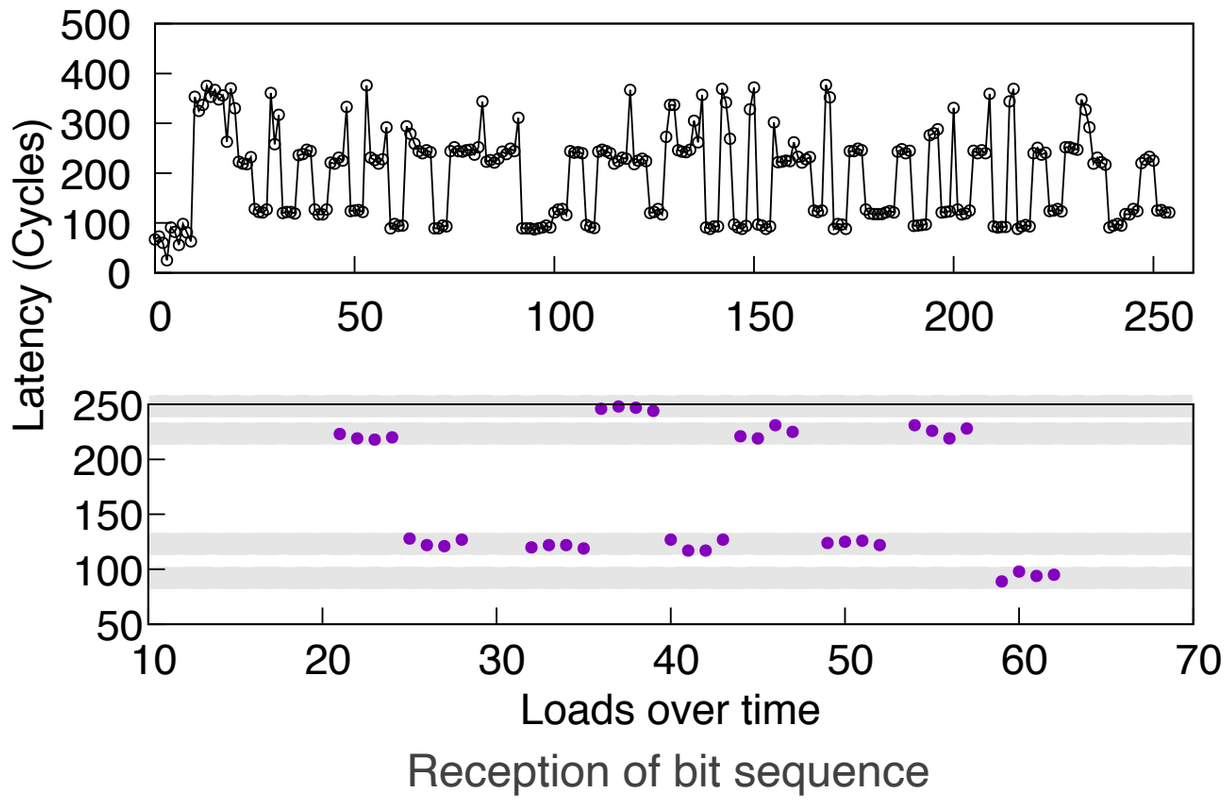
Reception of bit sequence

Multi-bit Symbol Transmission

- With 4 latency values, we can create 2-bit symbols
  - Each latency value represents a symbol
  - Much higher transmission bitrate



Multi-bit Symbol Transmission Scheme



Reception of bit sequence

Achieved bitrate: 1100 Kbps!

Mitigating Coherence State-based Leakage

Key Observation: current processor treats E and S state differently

E state cache line is owned by private cache

S state cache line is owned by shared cache

Hardware-based Mitigation

Shared cache directly responds with cache blocks in E state

Add notification to shared cache when core writes to private cache

Software-based Mitigation

Software based timing obfuscators

KSM usage pattern analysis and detection

Summary

- We *highlight* the vulnerability of cache coherence states
  - ***We demonstrate*** **that they are prone to information leakage**
- This vulnerability can compromise sensitive user data
  - **Can affect millions of processors**
  - **Can achieve considerably high transmission rates**
- We provide insights for information security aware cache coherence design

Thanks! Questions?

Contacts

Fan Yao: albertyao@gwu.edu

Guru Venkataramani: guruv@gwu.edu

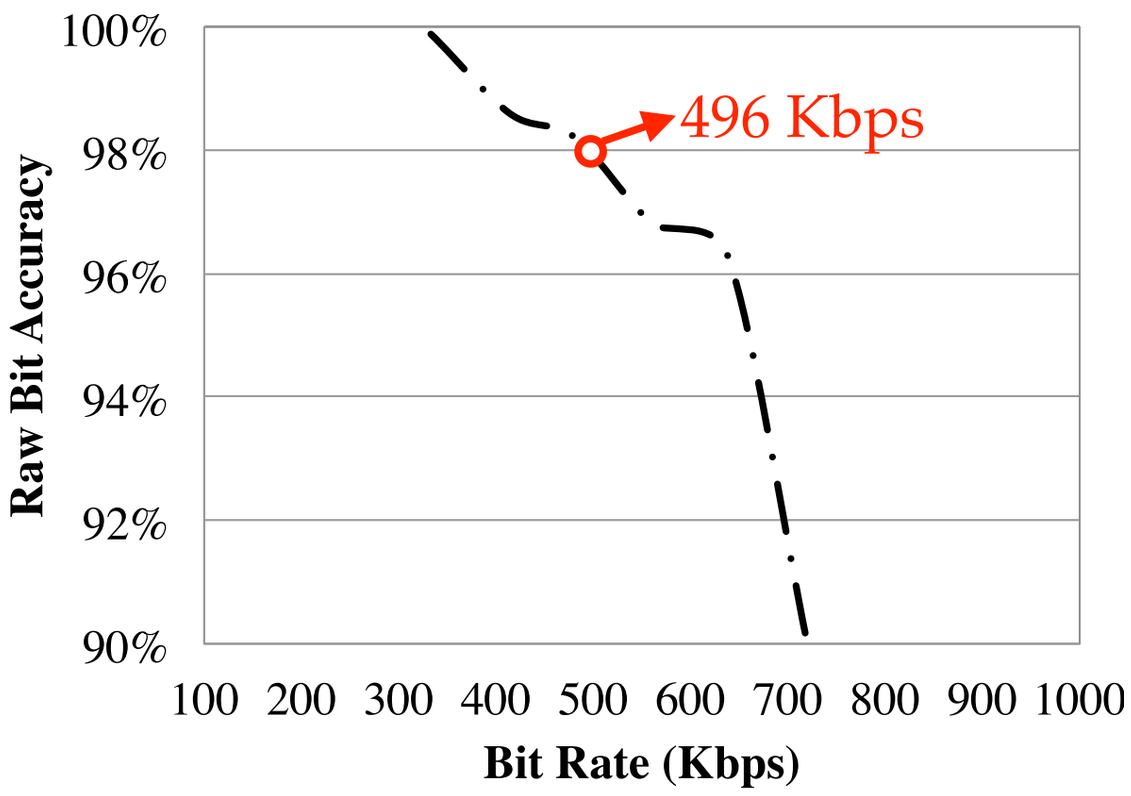
Acknowledgements

NSF

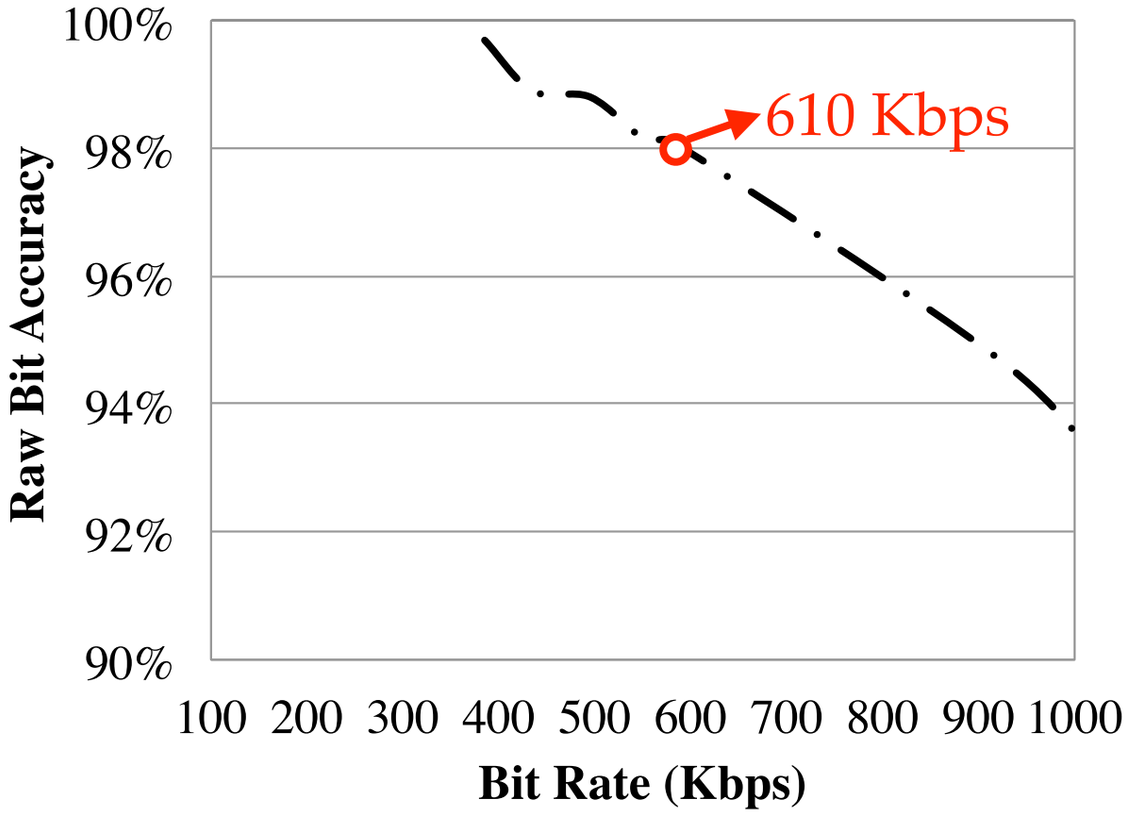
SRC® Semiconductor Research Corporation

Bit Accuracy vs. Bitrate

- Factors that influence raw bit rates
  - Spy's operation interval
  - Number of occurrences for info. bit states
  - Number of occurrences for bit boundary states



$RShared_{comm}$-$LExcl_{bound}$



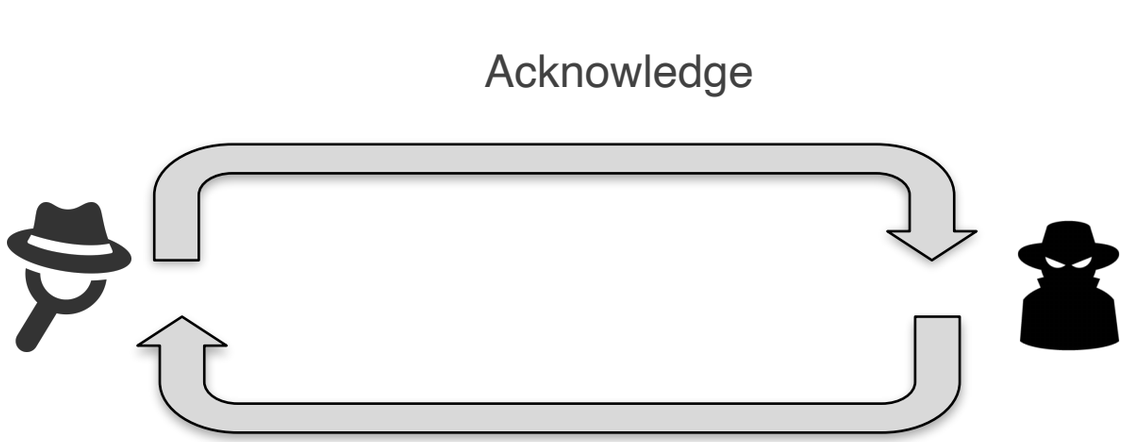
$RExcl_{comm}$-$LShared_{bound}$

Channel with Retransmission Scheme

- Simple retransmission scheme
  - Trojan and spy switch role (spy sends ACK bit)

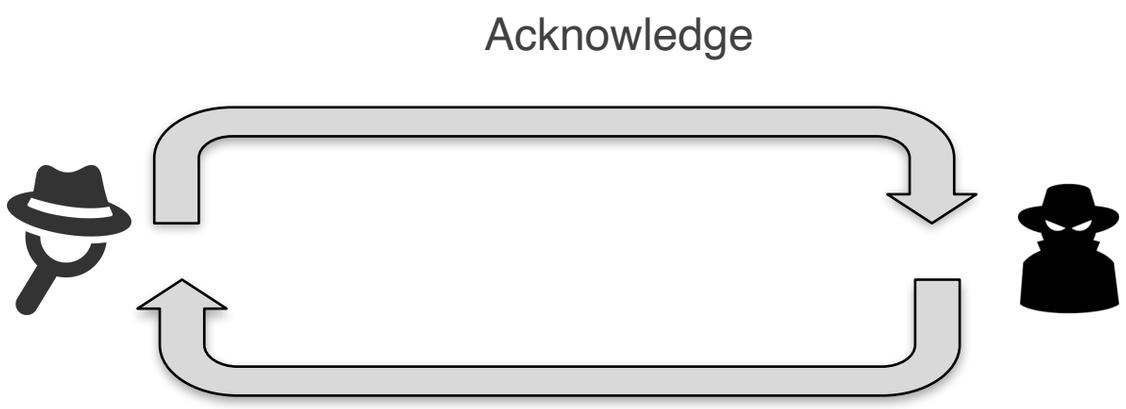
Channel with Retransmission Scheme

- Simple retransmission scheme
  - Trojan and spy switch role (spy sends ACK bit)

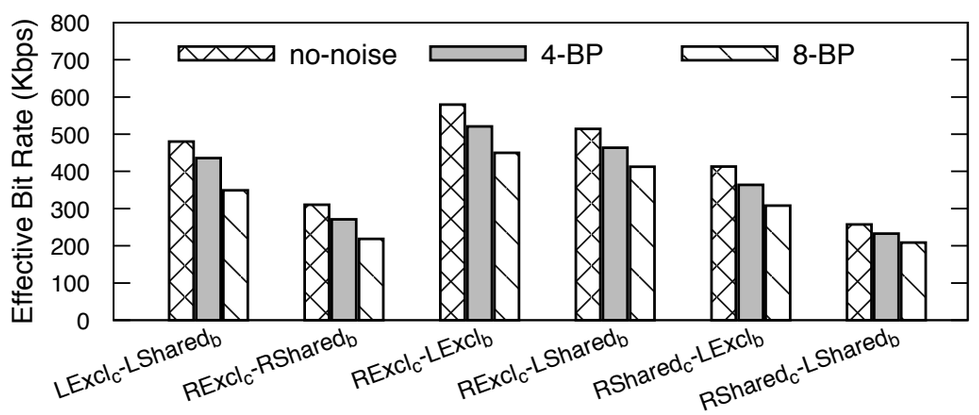


Channel with Retransmission Scheme

- Simple retransmission scheme
  - Trojan and spy switch role (spy sends ACK bit)



transmit secrets through Cache Coherence States



Effective rates under different noise levels

Secure E and S States