ACM/IEEE CODES+ISSS 2018, Turin, Italy

PAGURUS: Low-Overhead Dynamic Information Flow Tracking on Loosely Coupled Accelerators

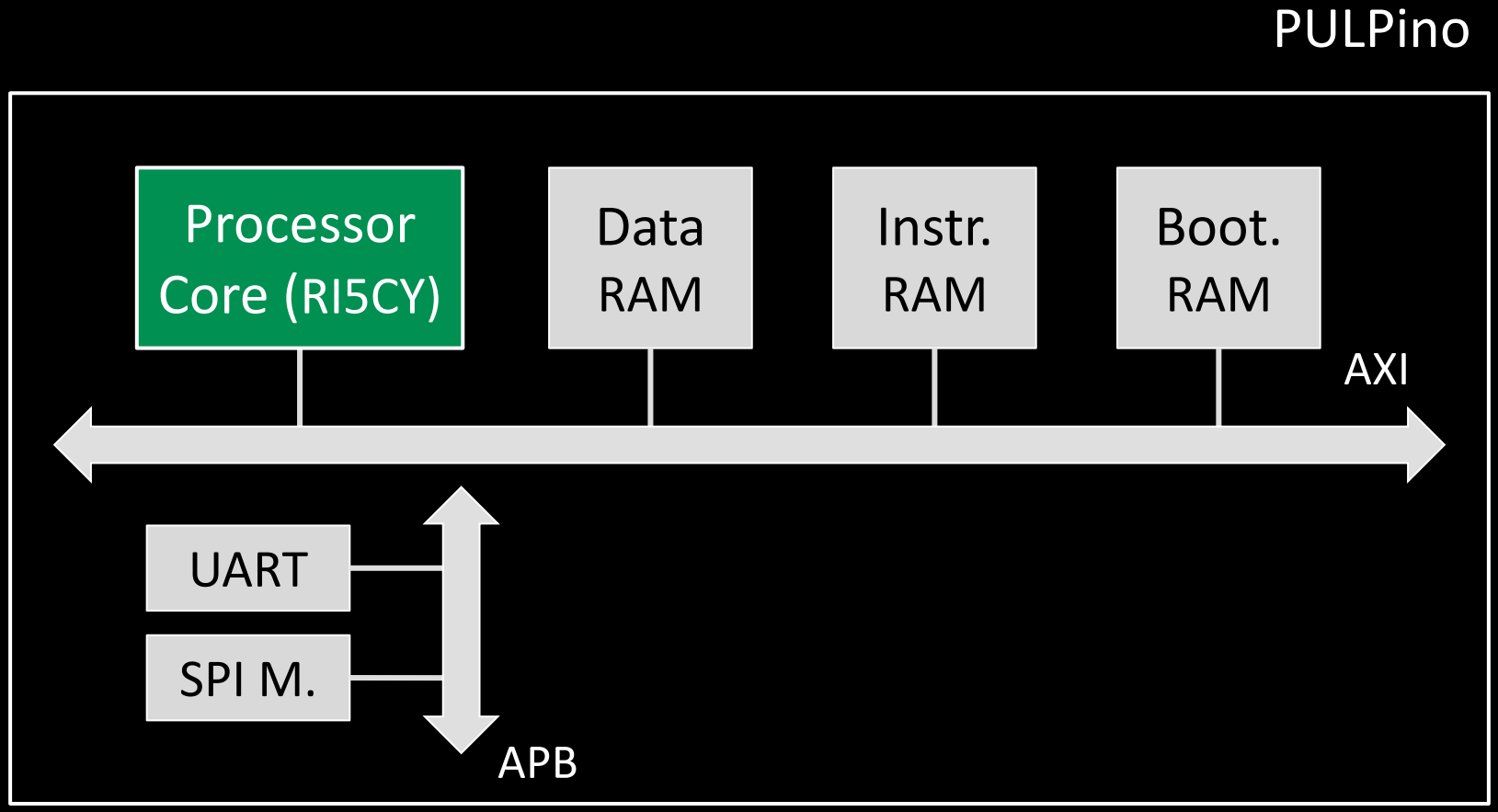
Luca Piccolboni, Giuseppe Di Guglielmo and Luca P. Carloni

Columbia University, NY, USA

COLUMBIA UNIVERSITY

Systems-on-Chip (SoCs)

Are Vulnerable to Software Attacks



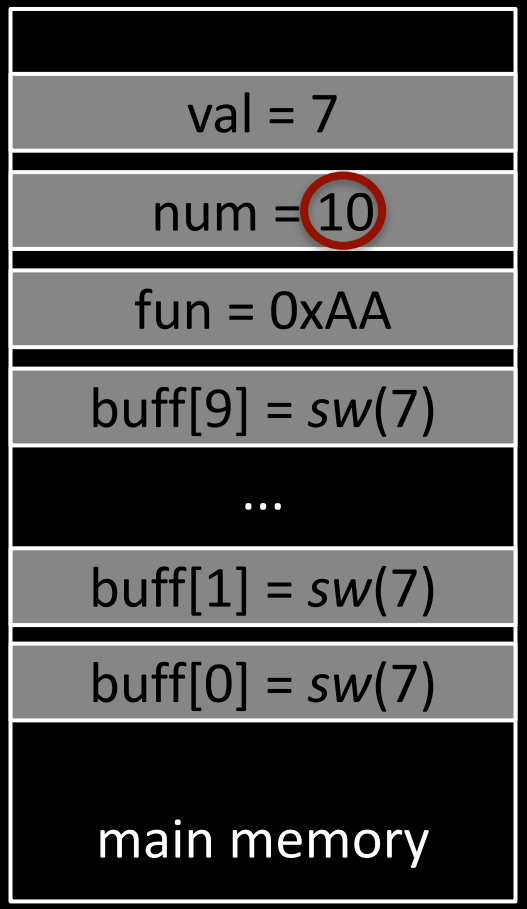
[M. Gautschi et al., IEEE VLSI ’17]

Attacking PULPino

Buffer-Overflow Attack

memory location: 0xAA

```
int buff[10], k;
int (*fun)(int) = foo;
int num = atoi(argv[1]);
int val = atoi(argv[2]);
/* this is a bad idea */
for (k = 0; k < num; ++k)
  buff[k] = sw(val);
fun(1); // call foo?
```

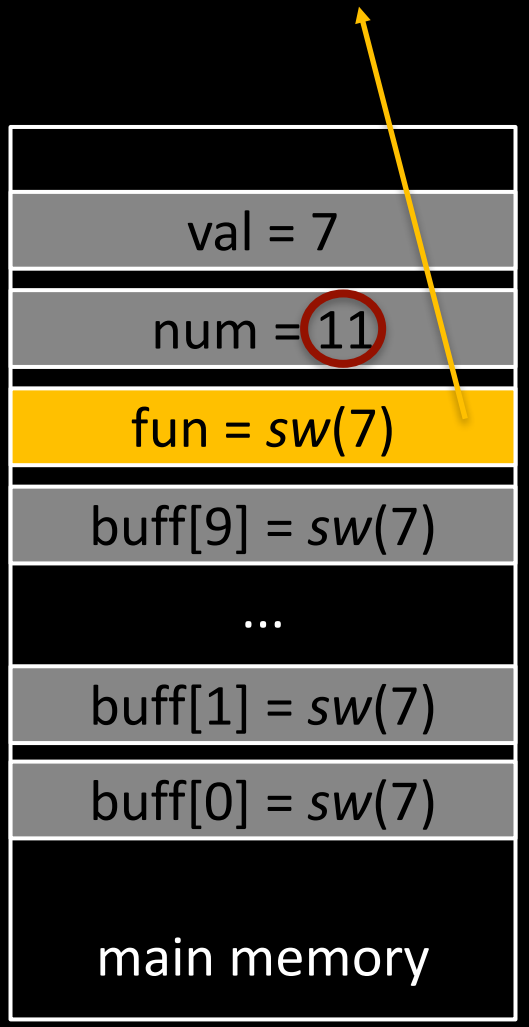


Attacking PULPino

Buffer-Overflow Attack

memory location: 0xAA

```
int buff[10], k;
int (*fun)(int) = foo;
int num = atoi(argv[1]);
int val = atoi(argv[2]);
/* this is a bad idea */
for (k = 0; k < num; ++k)
  buff[k] = sw(val);
fun(1); // call foo?
```



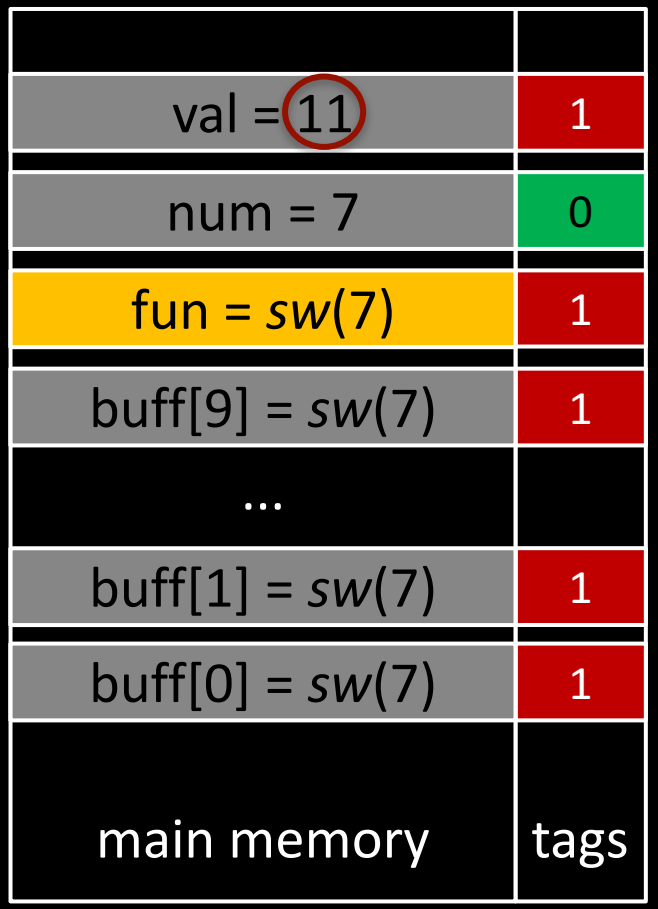
Attacking PULPino

Dynamic Information Flow Tracking (DIFT)

memory location: 0xAA

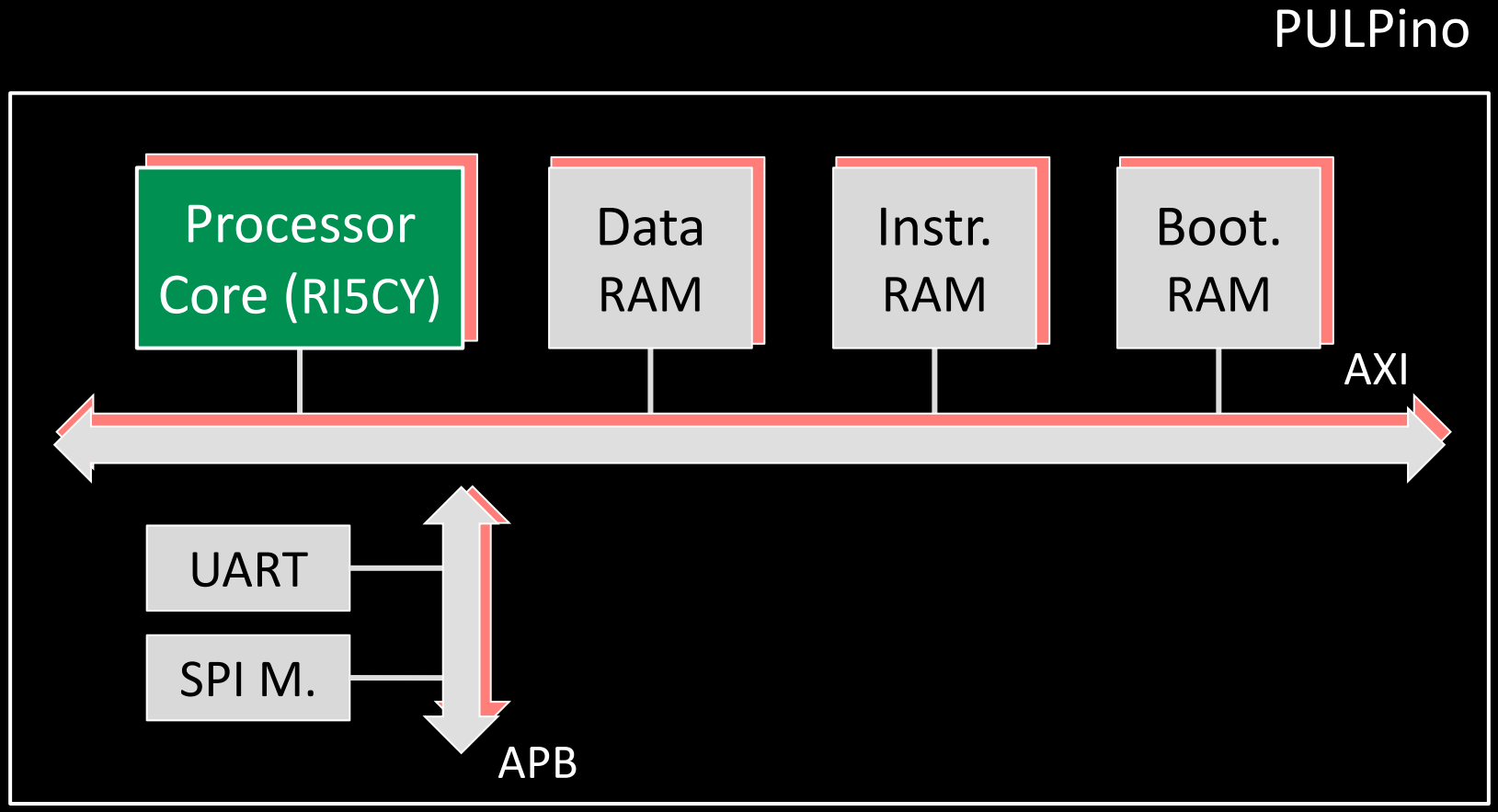
```
int buff[10], k;
int (*fun)(int) = foo;
int num = atoi(argv[1]);
int val = atoi(argv[2]);
/* this is a bad idea */
for (k = 0; k < num; ++k)
  buff[k] = sw(val);
fun(1); // call fun
```

[G. E. Suh et al., ACM ASPLOS '04]



Homogenous SoCs

Now Secured with DIFT



DIFT Extensions

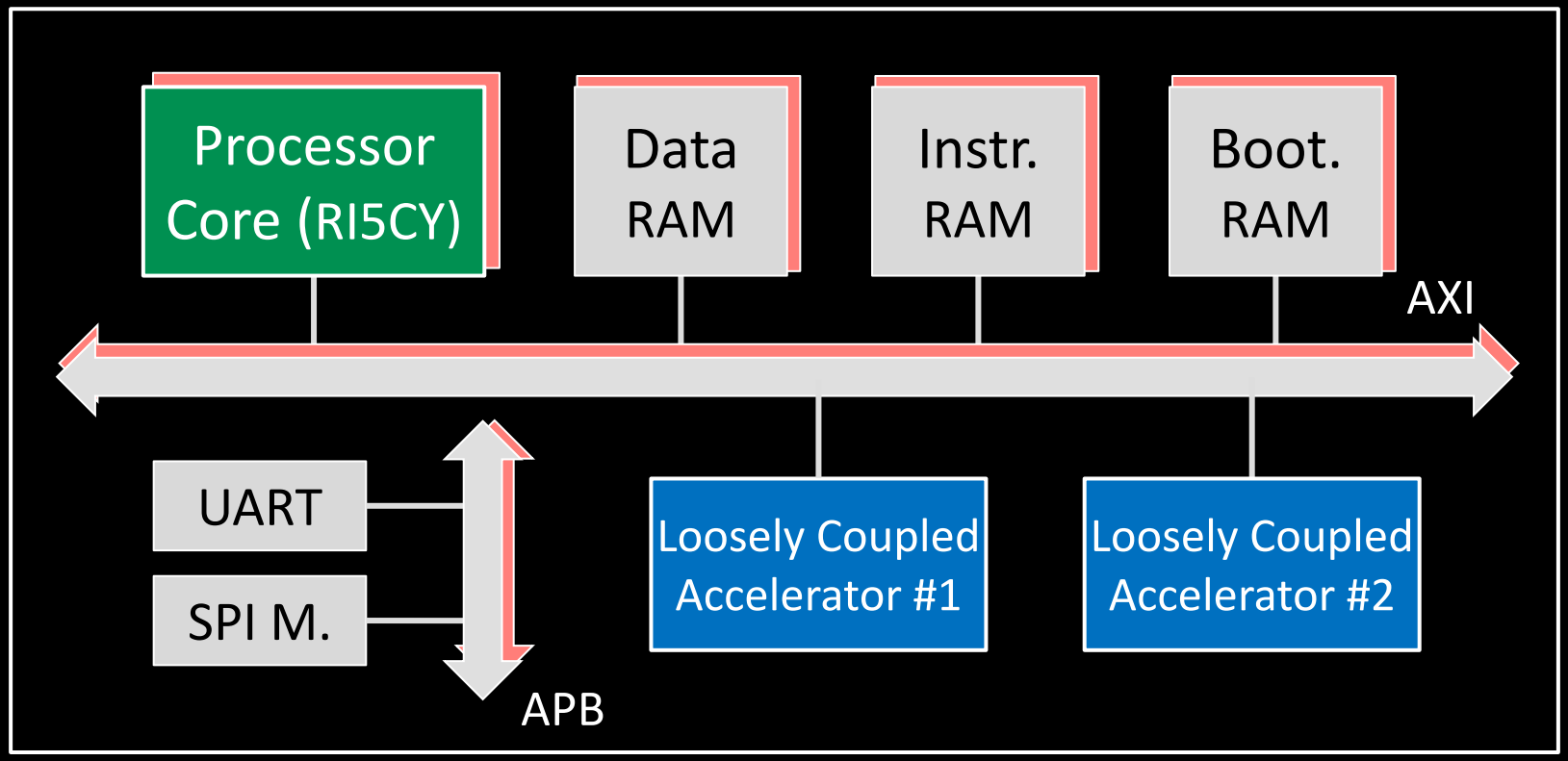
[M. Gautschi et al., IEEE VLSI ’17]

[C. Palmiero et al., IEEE HPEC ’18]

Heterogeneous SoCs

No-More-Secured with DIFT

PULPino



DIFT Extensions

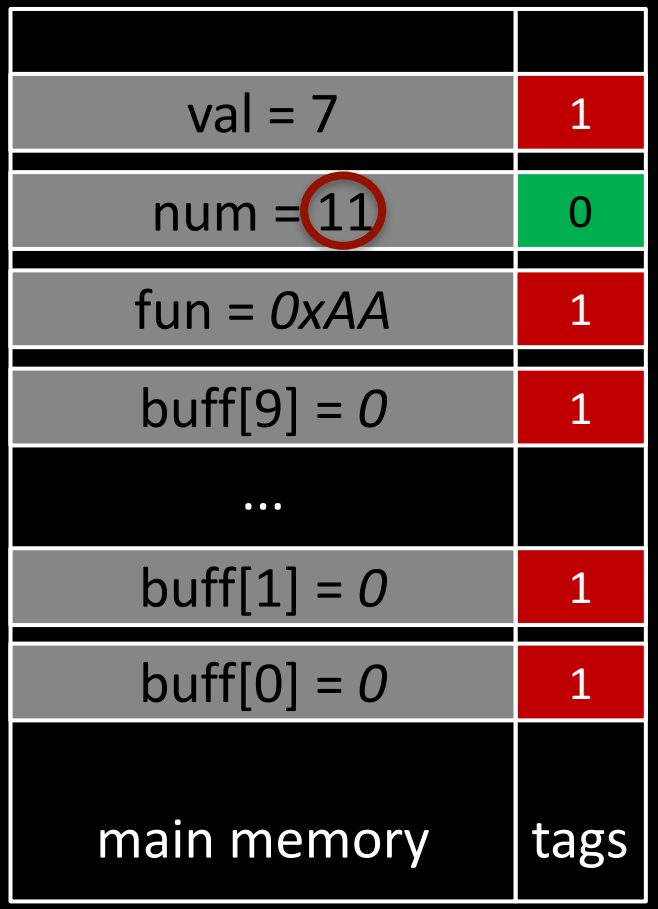
[M. Gautschi et al., IEEE VLSI ’17]

[C. Palmiero et al., IEEE HPEC ’18]

Attacking PULPino (Again)

Buffer-Overflow Attack

```
int buff[10] = {0};
int (*f)(int) = foo;
int num = atoi(argv[1]);
int val = atoi(argv[2]);
/* this is a bad idea */
hw(num, val, buff);
```

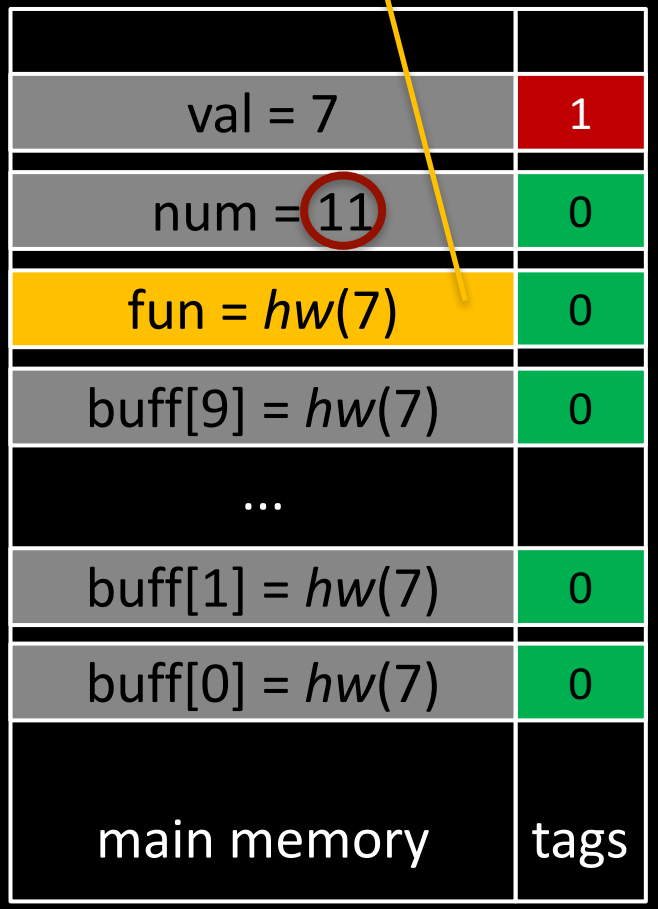


Attacking PULPino (Again)

Buffer-Overflow Attack

```
int buff[10] = {0};
int (*f)(int) = foo;
int num = atoi(argv[1]);
int val = atoi(argv[2]);
/* this is a bad idea */
hw(num, val, buff);
```

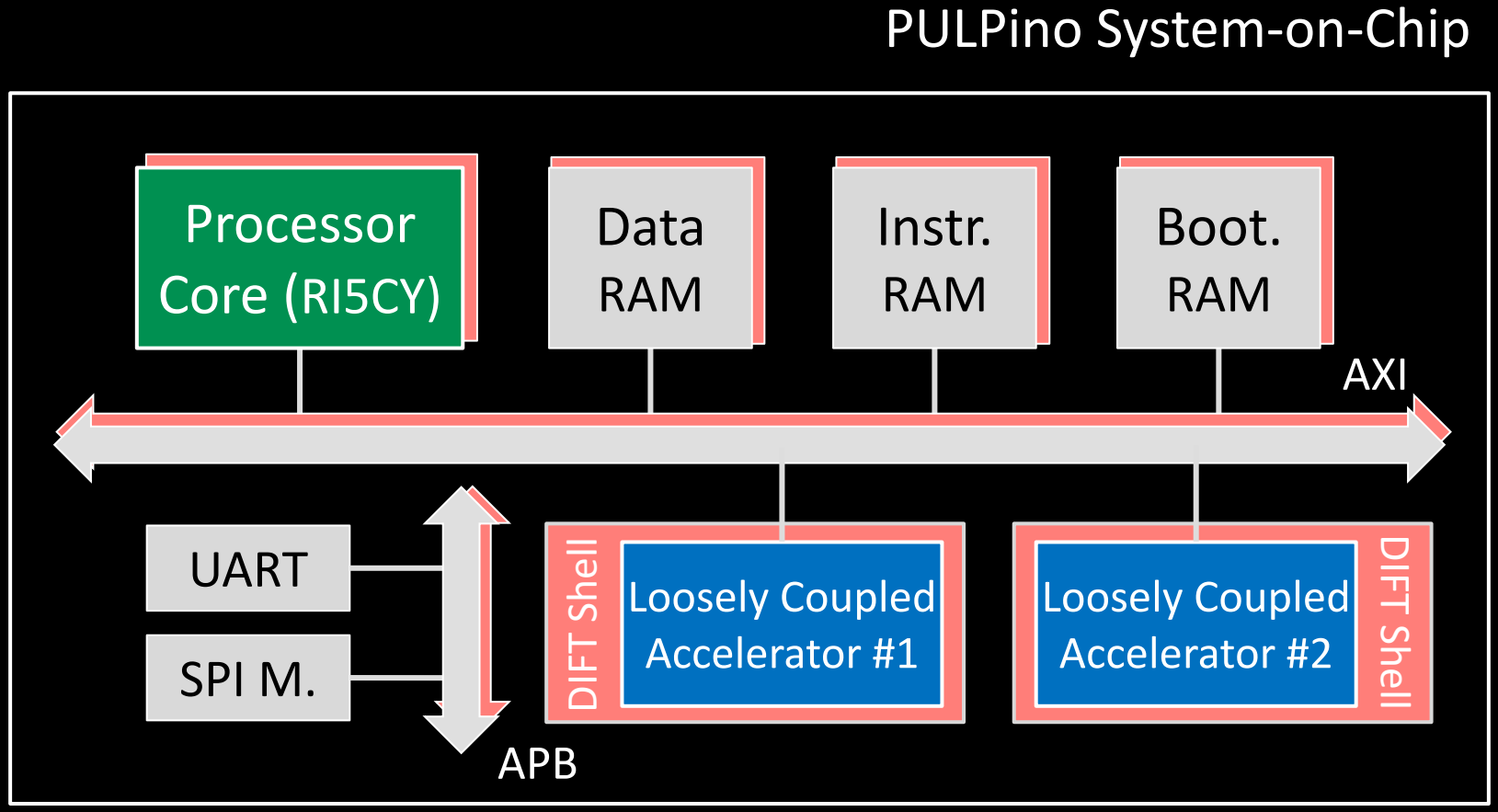
the accelerator is not able to propagate the tags



Contributions

1. We propose PAGURUS, a methodology to design a circuit shell that adds DIFT support to accelerators

Contributions



Contributions

1. We propose PAGURUS, a methodology to design a circuit shell that adds DIFT support to accelerators
    a) The shell design is *independent* from the design of the accelerators and vice versa
    b) The shell has *low overheads* on both the performance and cost of accelerators
2. We propose a metric to quantitatively measure the security guarantees provided by the shell

Preliminaries

Assumptions and Attack Model

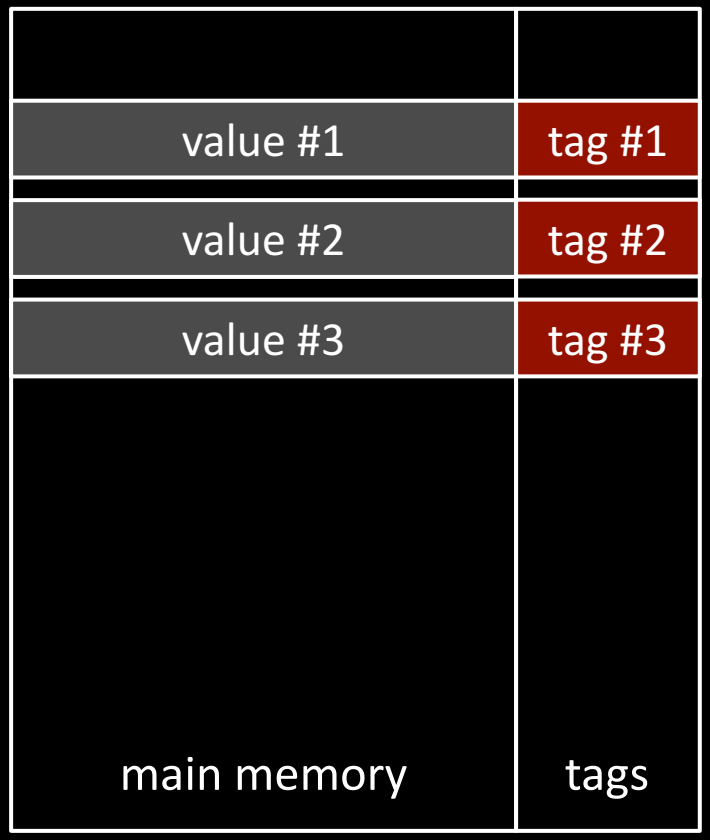
1. The hardware is safe: no hardware Trojans

2. The software is not safe: it contains bugs and vulnerabilities useful for the attackers

   → The attackers exploit these vulnerabilities through common I/O interfaces with the goal of affecting the integrity and/or the confidentiality of the hardware-accelerated software applications

Preliminaries
Tagging Scheme

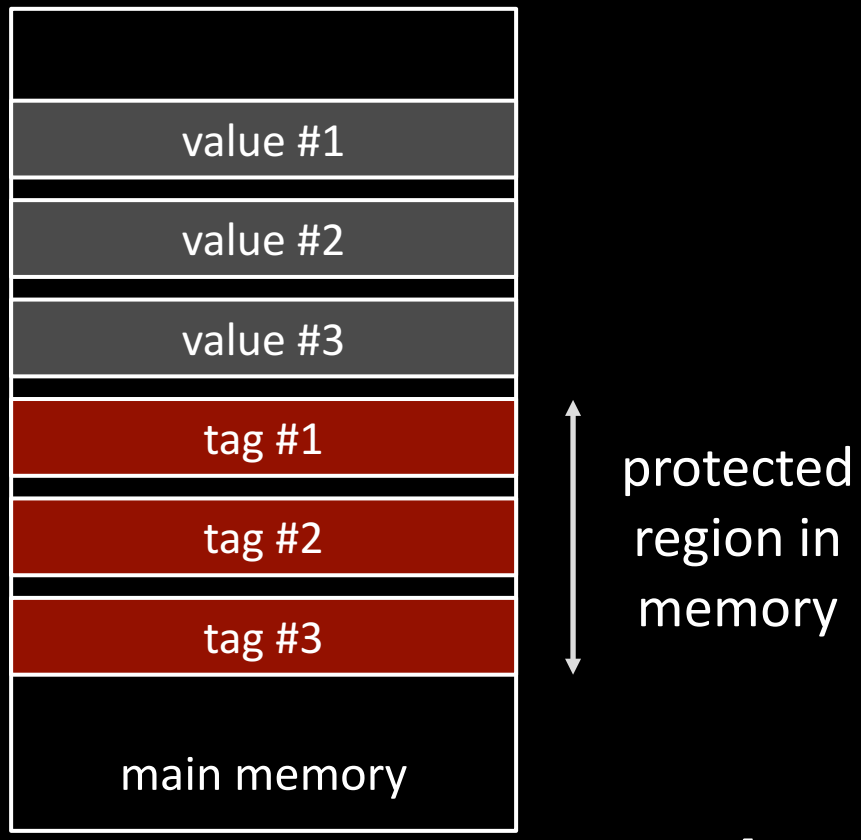
1. Coupled Scheme



[J. Porquet et al., ACM/IEEE CODES'13]

Preliminaries
Tagging Scheme

1. Coupled Scheme
2. Decoupled Scheme



[J. Porquet et al., ACM/IEEE CODES'13]

Preliminaries
Tagging Scheme

1. Coupled Scheme
2. Decoupled Scheme
   2.1. Interleaved Scheme

tag offset = # words in memory between two consecutive values

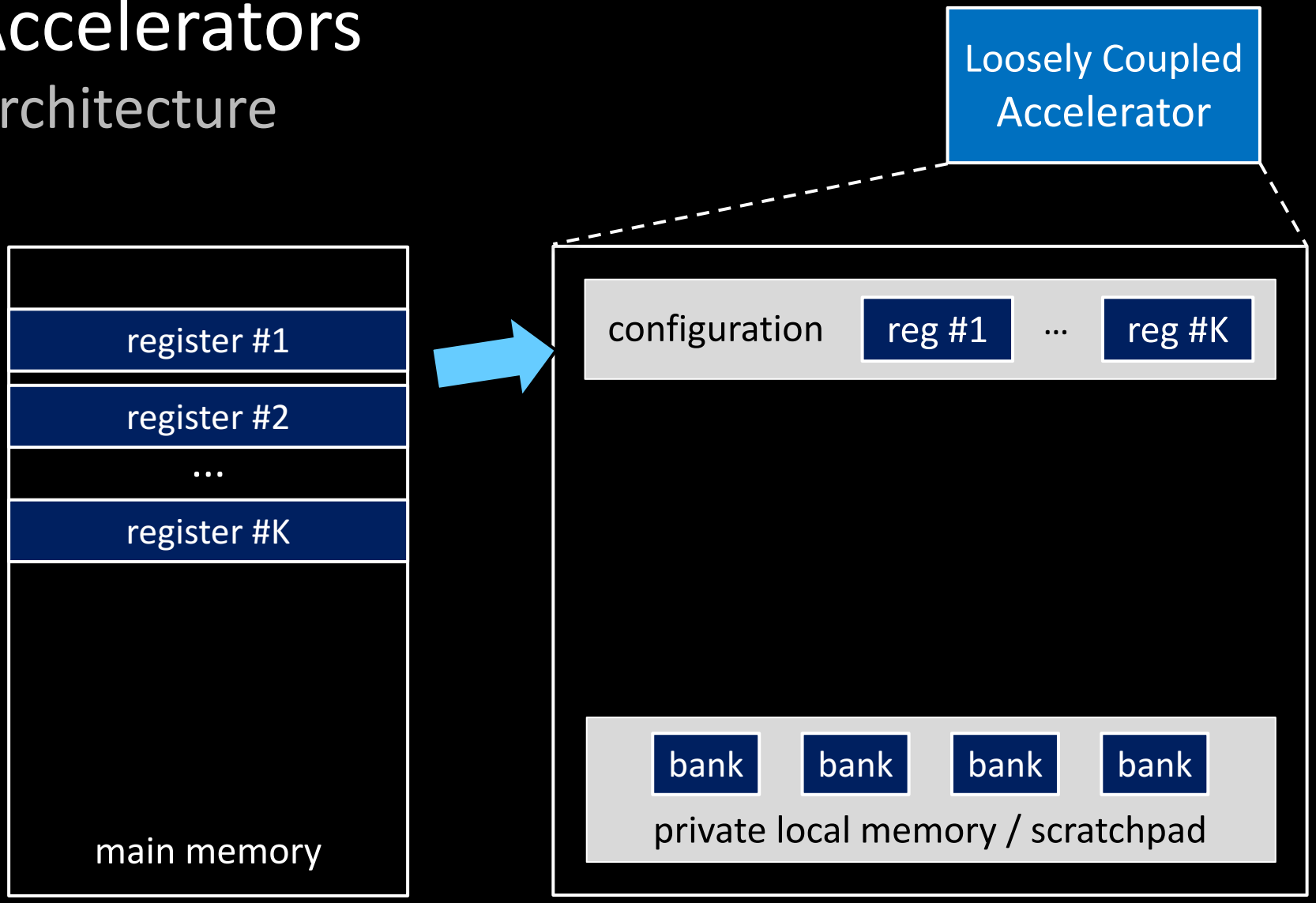
(tag offset = 1)

[J. Porquet et al., ACM/IEEE CODES'13]

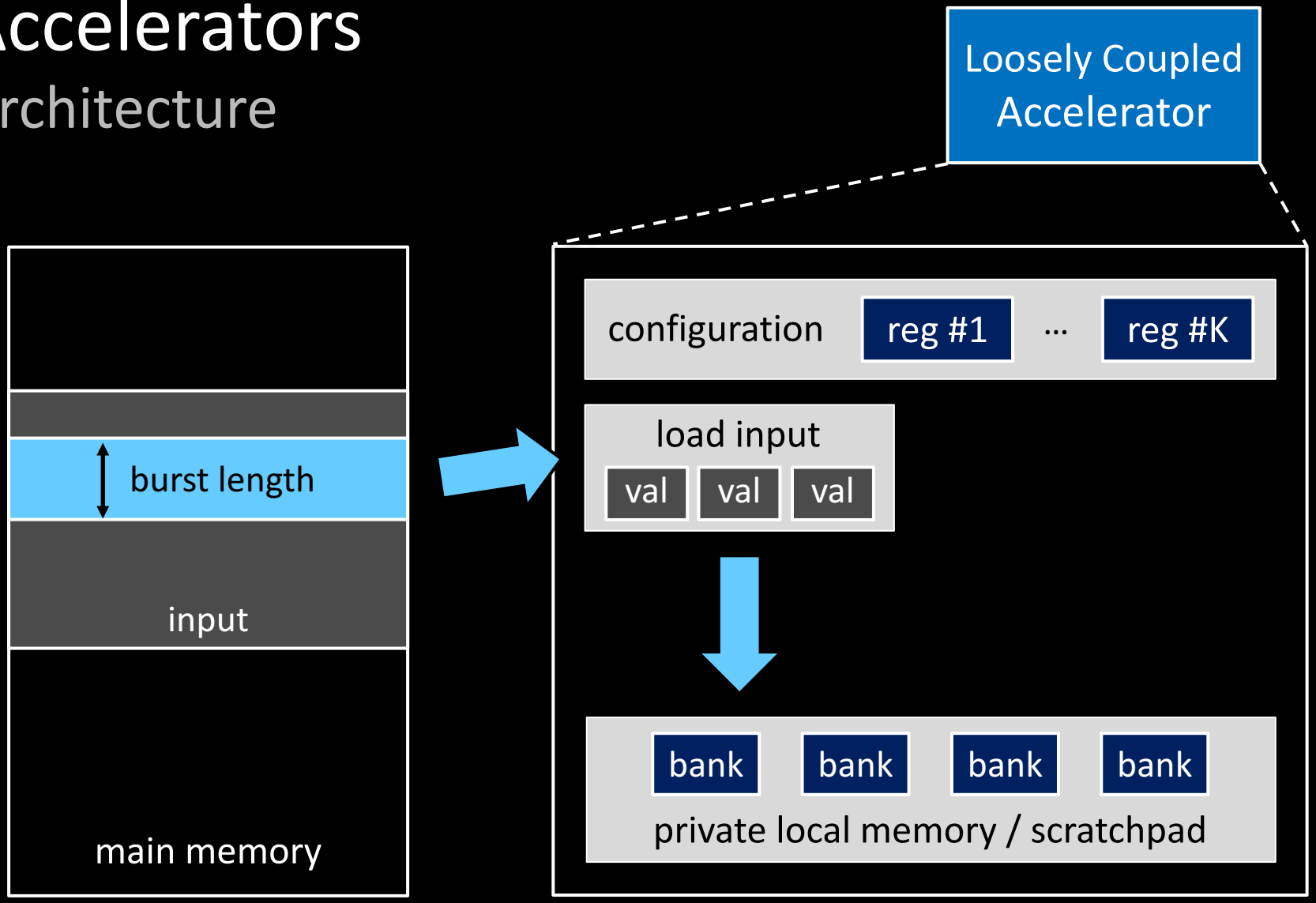
Contributions

1. We propose PAGURUS, a methodology to design a circuit shell that adds DIFT support to accelerators
    a) The shell design is *independent* from the design of the accelerators and vice versa
    b) The shell has *low overheads* on both the performance and cost of accelerators

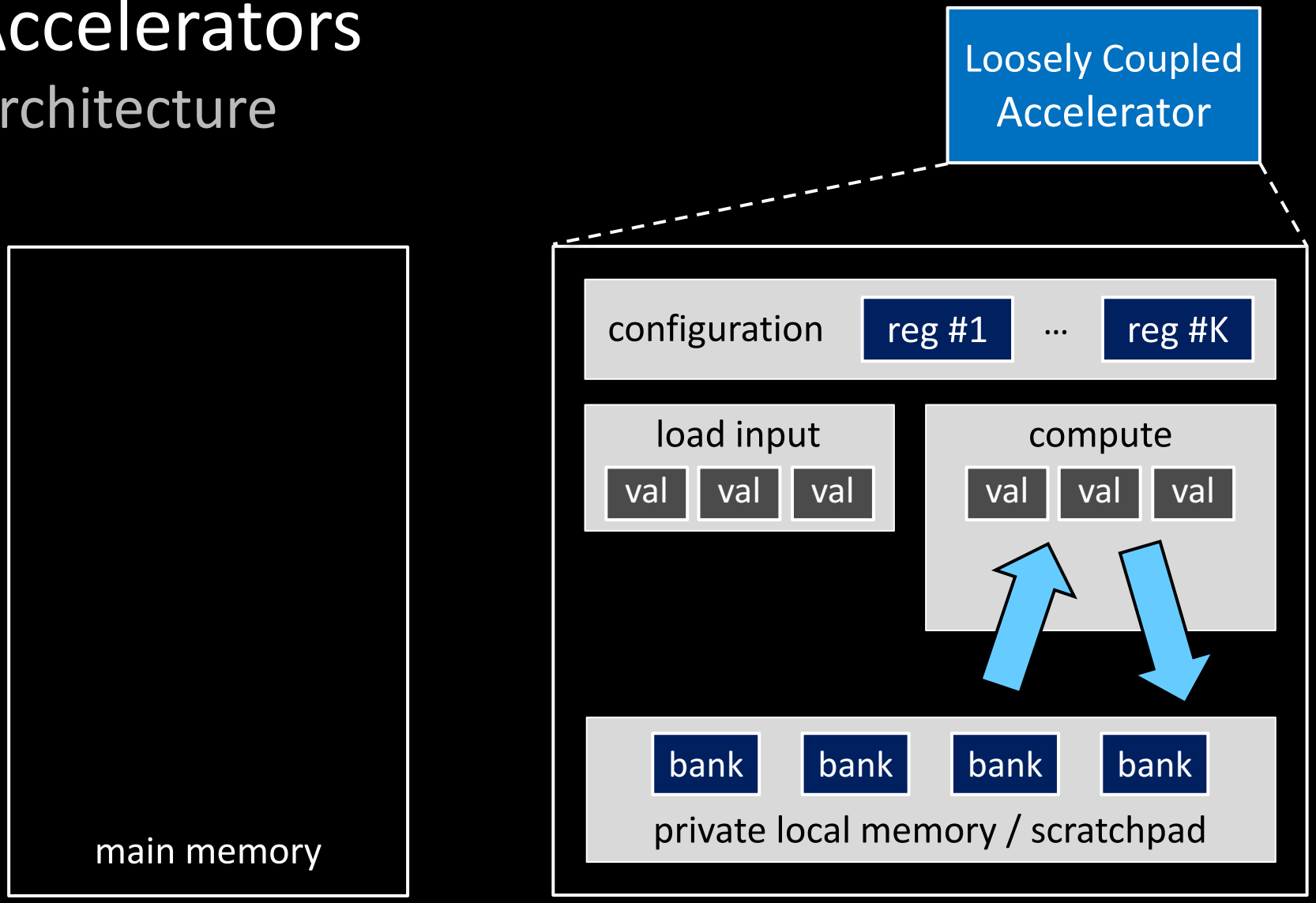
Accelerators
Architecture



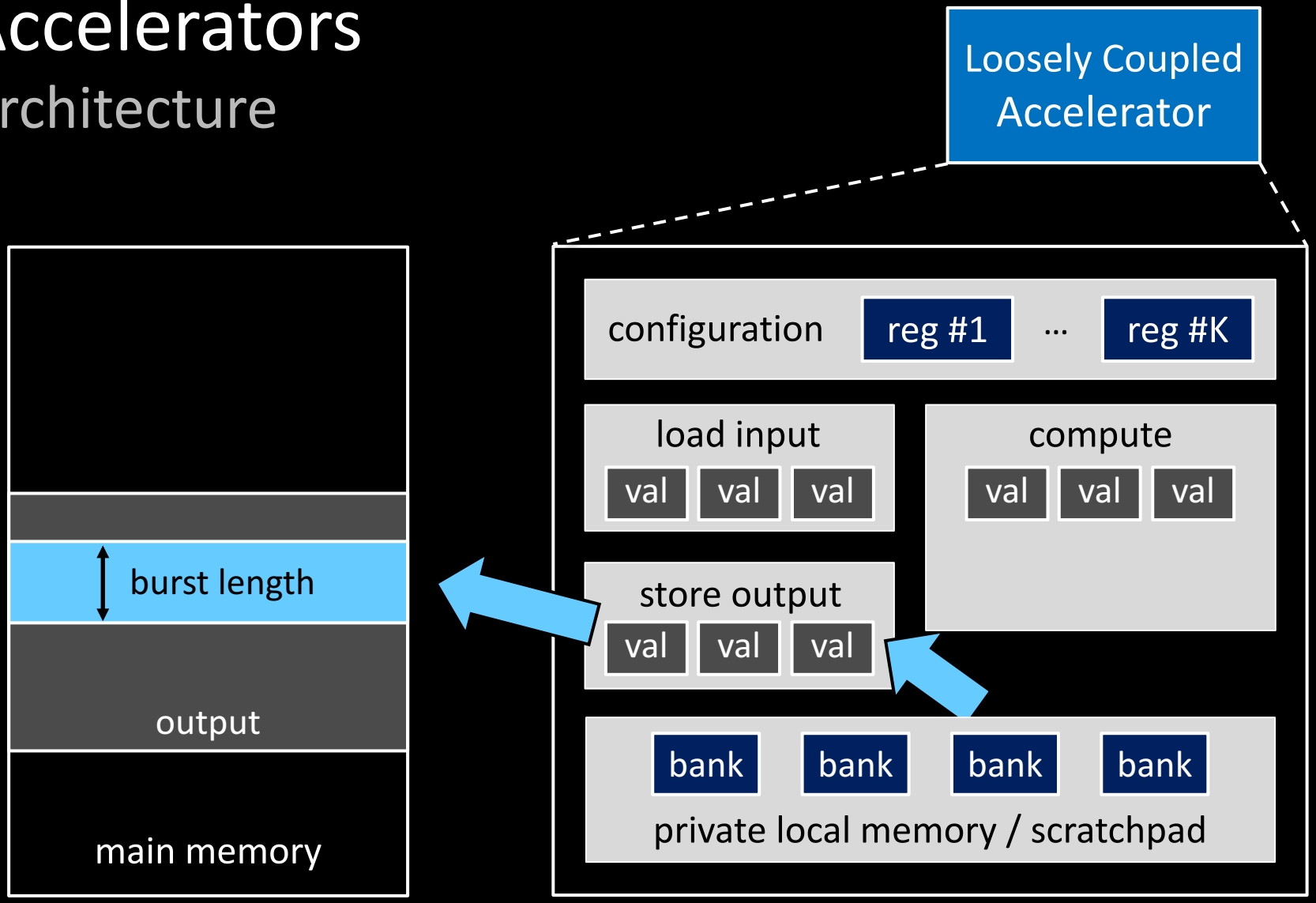
Accelerators
Architecture



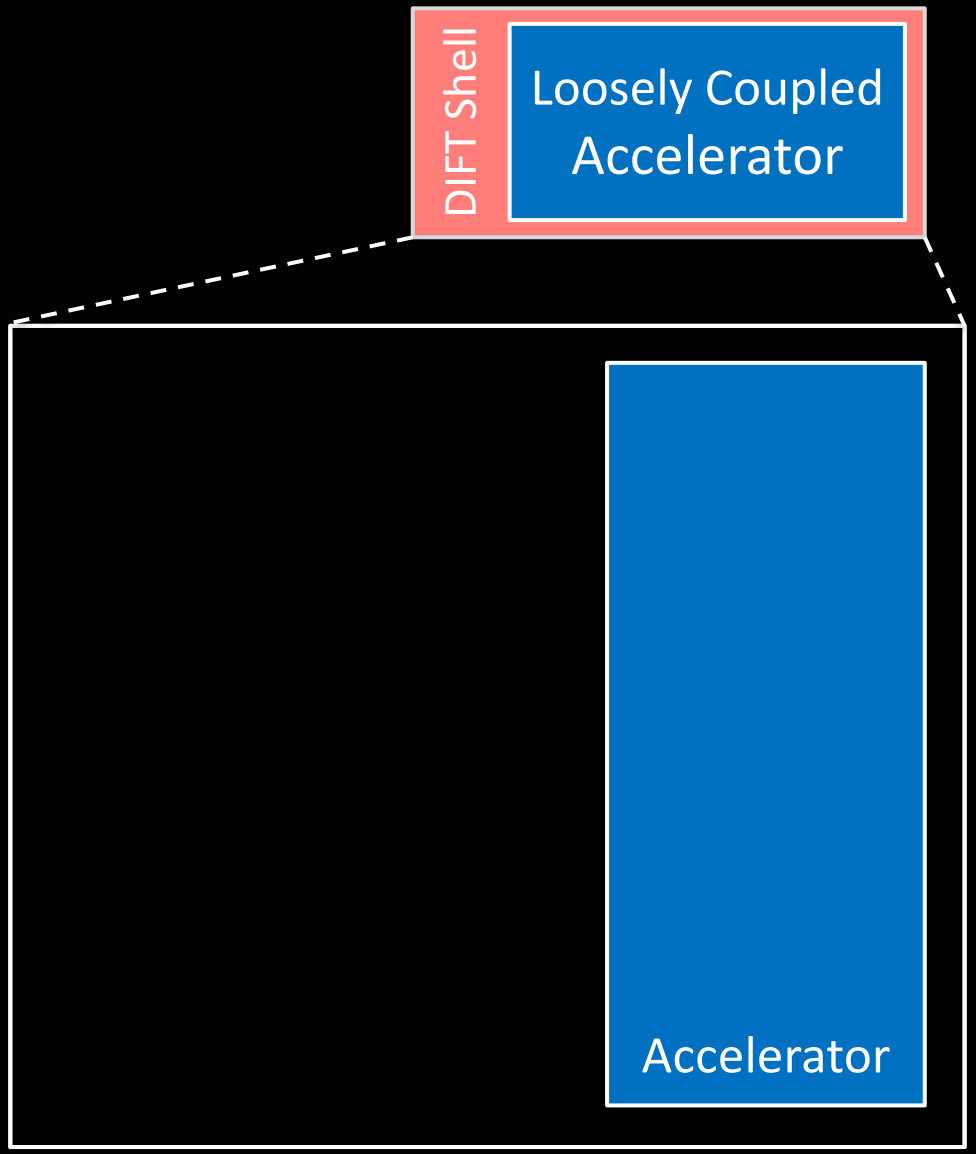
Accelerators
Architecture



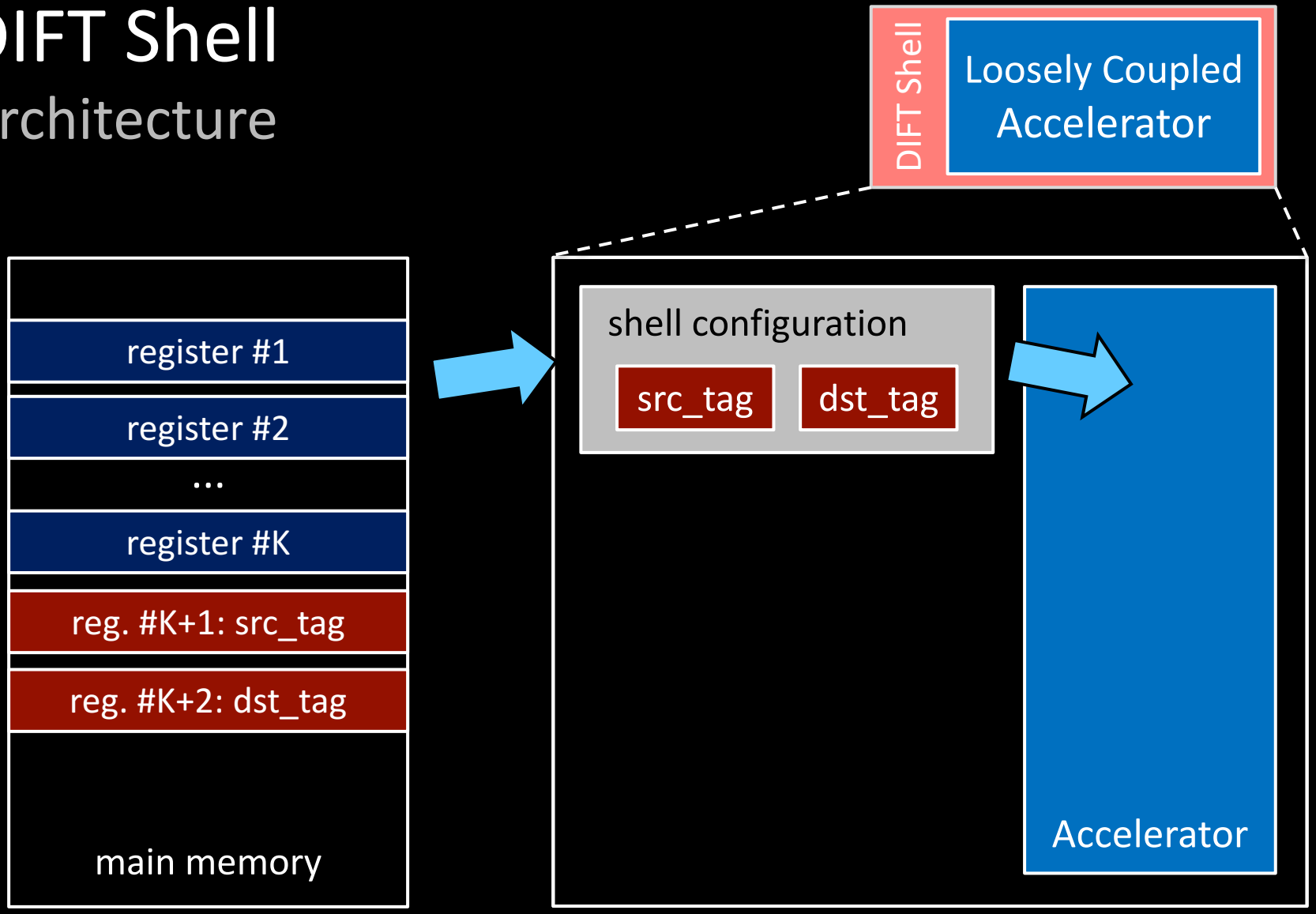
Accelerators
Architecture



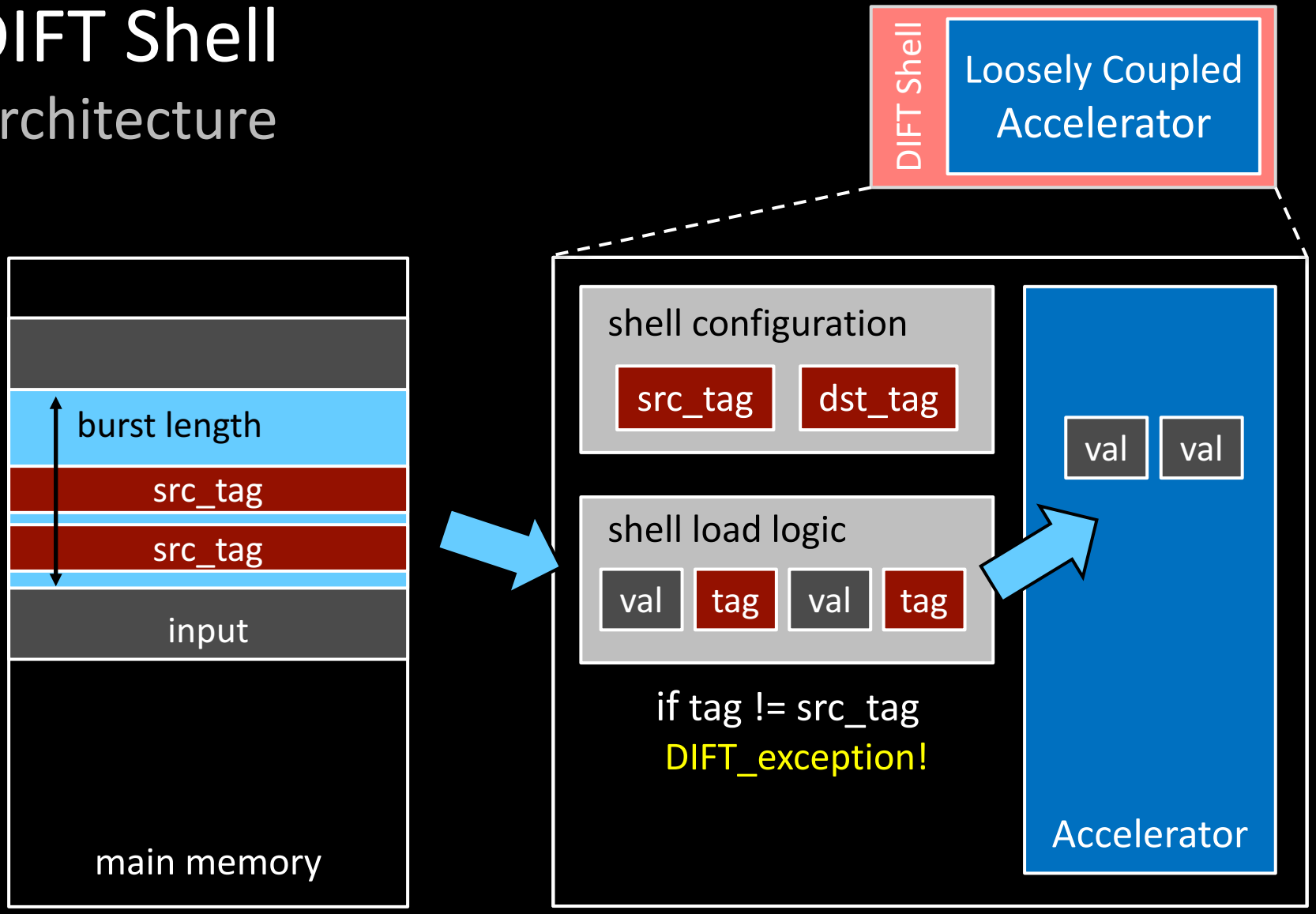
DIFT Shell
Architecture



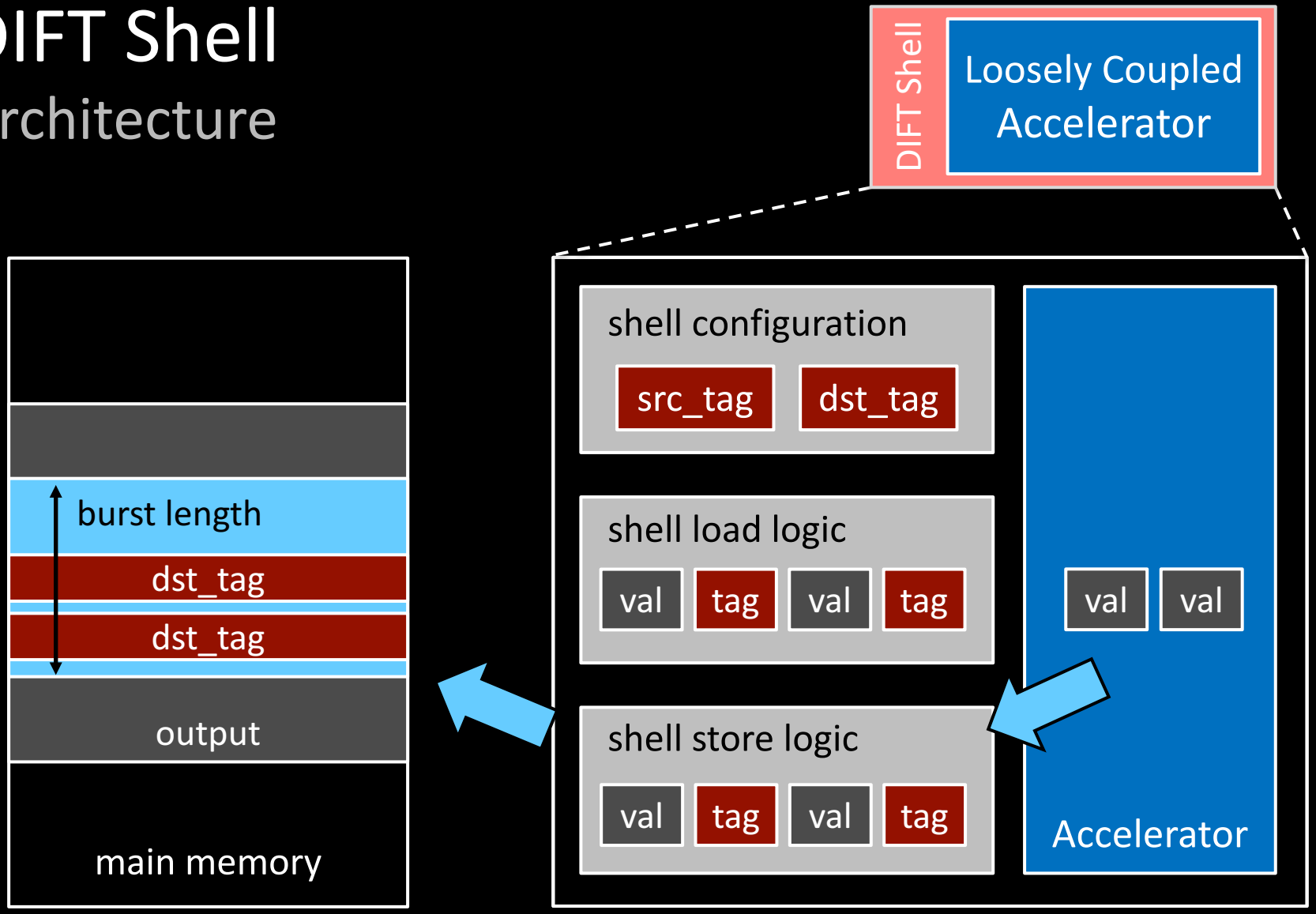
DIFT Shell
Architecture



DIFT Shell
Architecture



DIFT Shell
Architecture

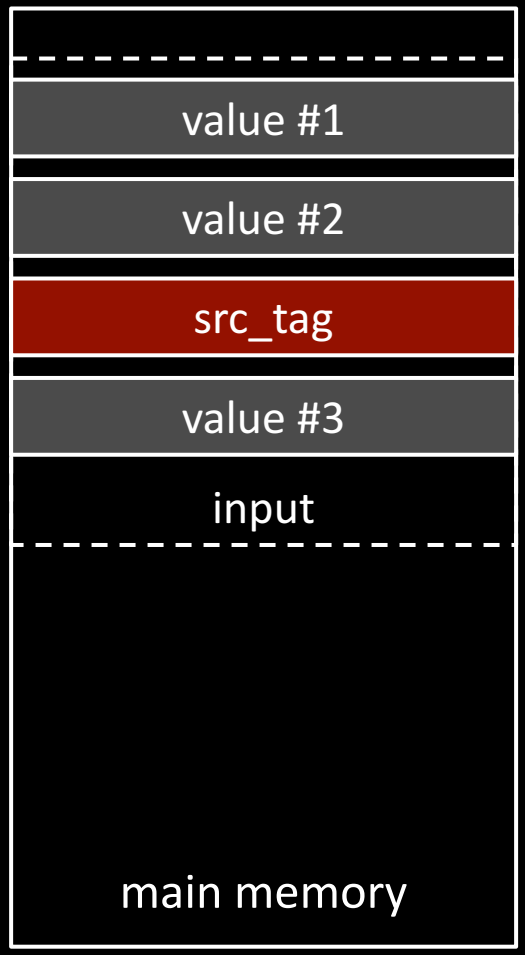


Contributions

1. We propose PAGURUS, a methodology to design a circuit shell that adds DIFT support to accelerators
    a) The shell design is *independent* from the design of the accelerators and vice versa
    b) The shell has *low overheads* on both the performance and cost of accelerators
2. We propose a metric to quantitatively measure the security guarantees provided by the shell

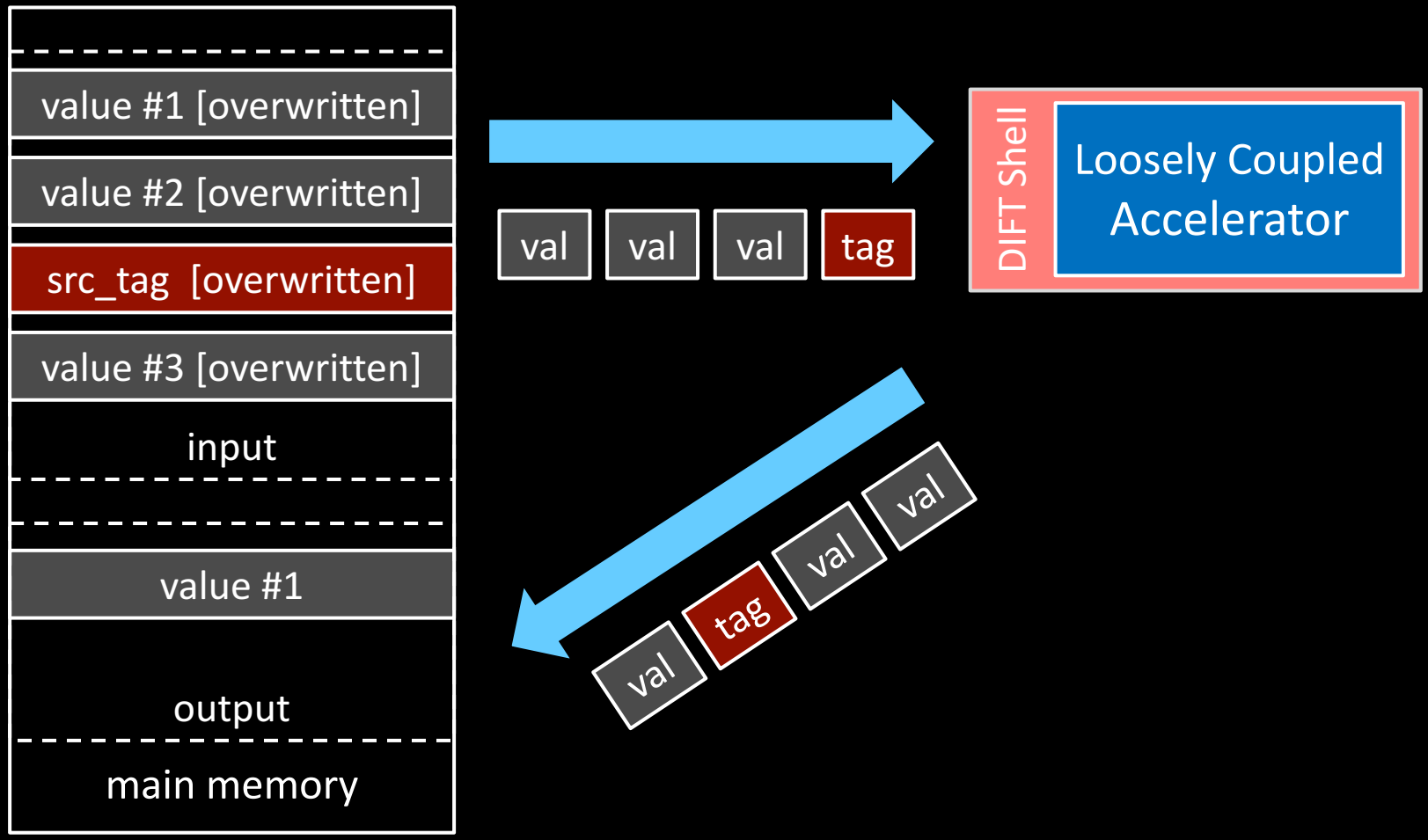
A Security Metric

Definition



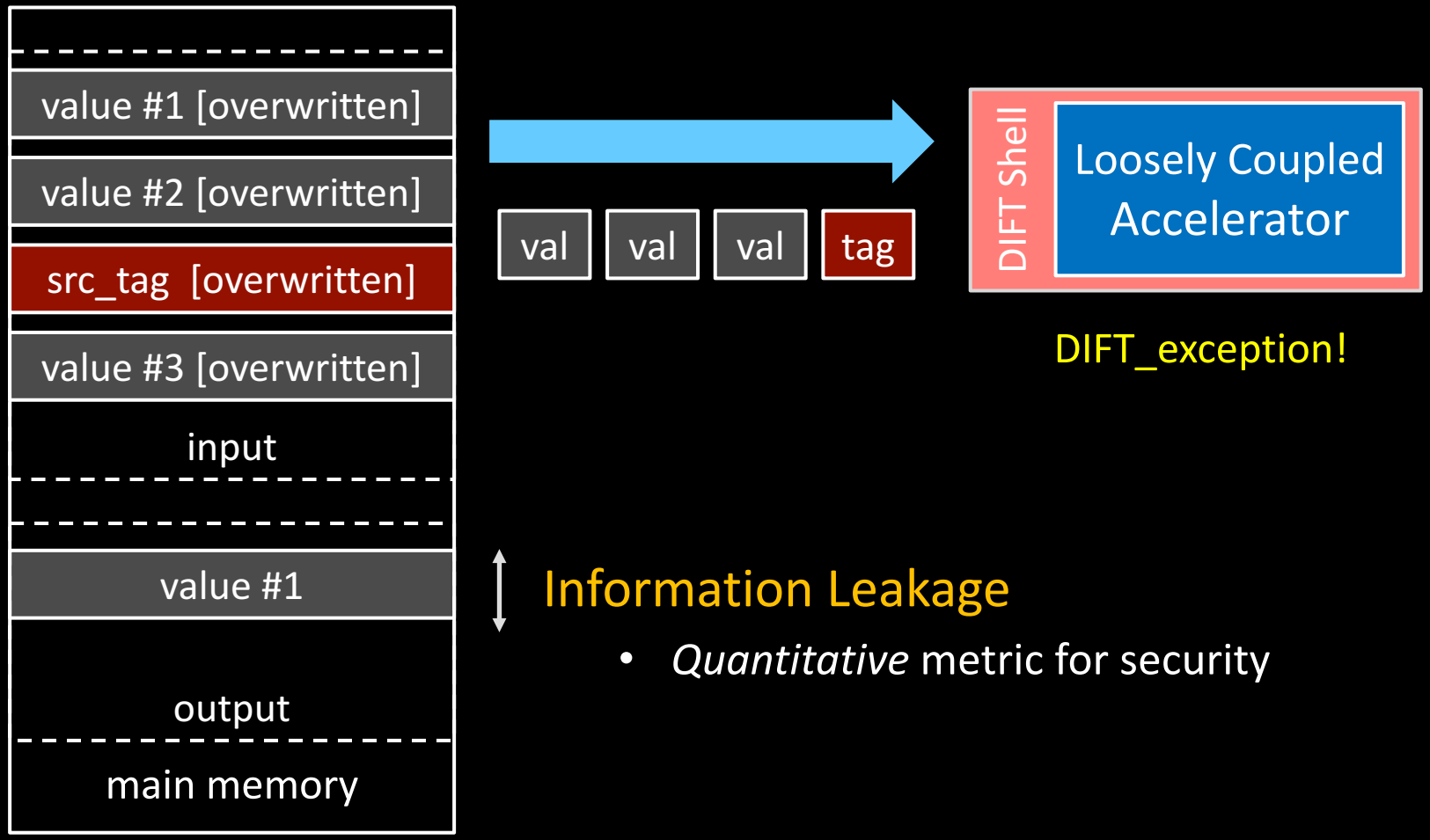
A Security Metric

Definition



A Security Metric

Definition



- *Quantitative* metric for security

A Security Metric

Analysis

- Information Leakage: amount of data that can be produced as output by an accelerator before its shell realizes that the input has been corrupted

1. Tag offset: ⇑ tag offset ⇑ leakage
2. Algorithm: ⇓ I/O ratio ⇑ leakage

I/O ratio: the number of load bursts necessary to produce a store burst

A Security Metric

Analysis

- Information Leakage: amount of data that can be produced as output by an accelerator before its shell realizes that the input has been corrupted

1. Tag offset: ⇑ tag offset ⇑ leakage
2. Algorithm: ⇓ I/O ratio ⇑ leakage
3. Implementation: ⇑ burst len. ⇓ leakage
4. Workload: ⇓ work. size ⇓ leakage

Experimental Results

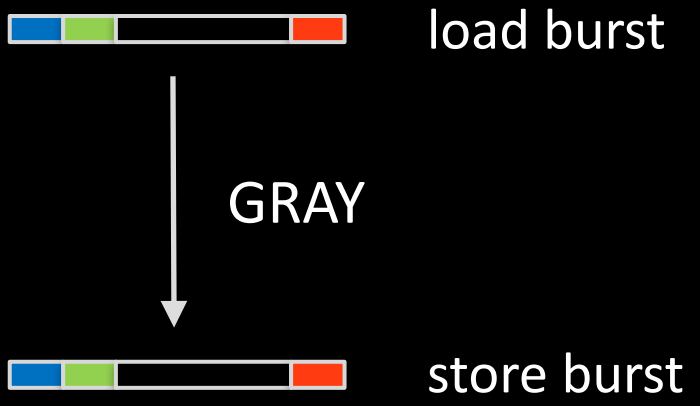
Experimental Setup (1/2)

- We designed three loosely coupled accelerators:
  - GRAY: converts a RGB image into a grayscale image
  - MEAN: calculates the mean of a 2D matrix (columns)
  - MULTS: mutiplies a 2D matrix by its transpose

Experimental Results

Experimental Setup (1/2)

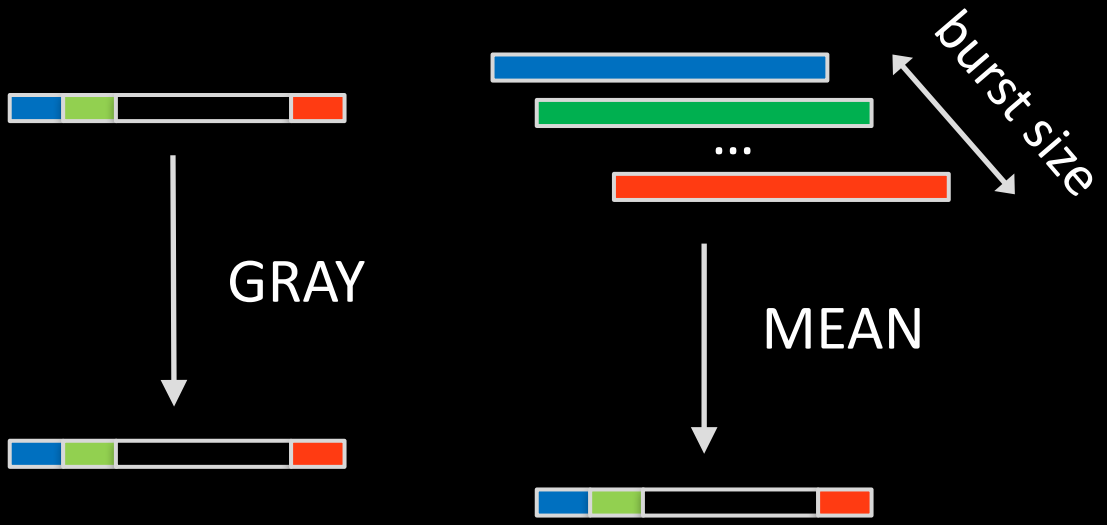
- We designed three loosely coupled accelerators:
  - GRAY: converts a RGB image into a grayscale image
  - MEAN: calculates the mean of a 2D matrix (columns)
  - MULTS: mutiplies a 2D matrix by its transpose



Experimental Results

Experimental Setup (1/2)

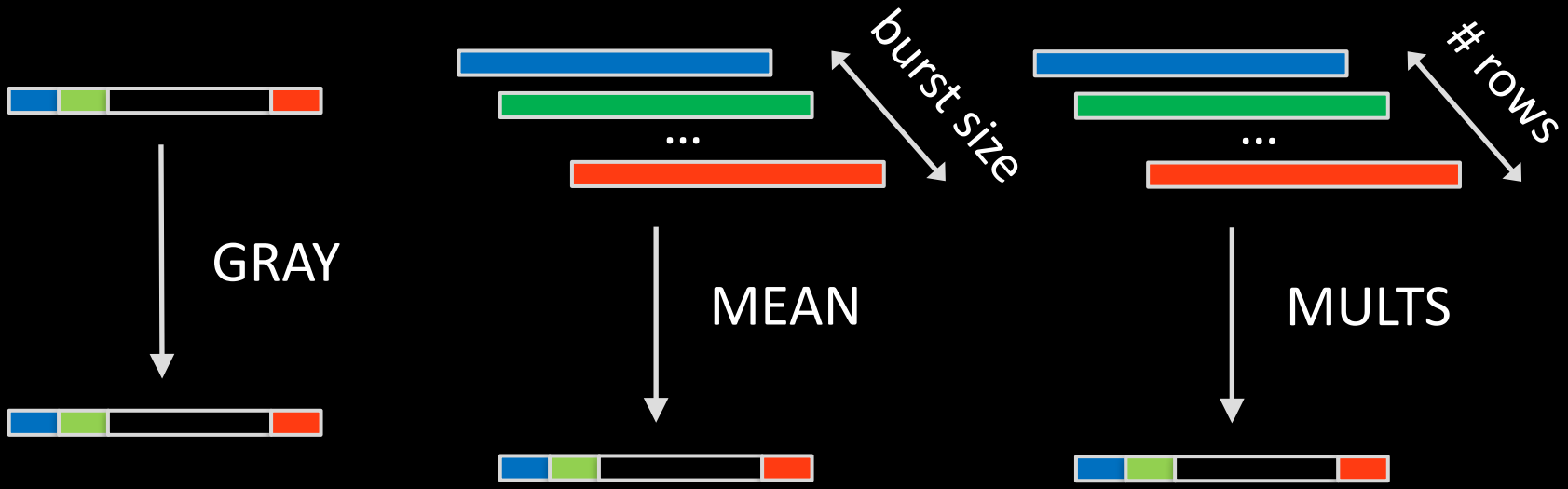
- We designed three loosely coupled accelerators:
  - GRAY: converts a RGB image into a grayscale image
  - MEAN: calculates the mean of a 2D matrix (columns)
  - MULTS: mutiplies a 2D matrix by its transpose



Experimental Results

Experimental Setup (1/2)

- We designed three loosely coupled accelerators:
  - GRAY: converts a RGB image into a grayscale image
  - MEAN: calculates the mean of a 2D matrix (columns)
  - MULTS: mutiplies a 2D matrix by its transpose

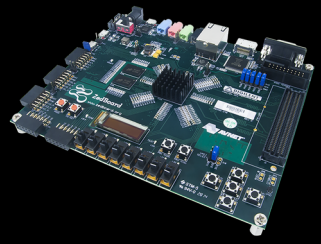


Experimental Results
Experimental Setup (1/2)

- We designed three loosely coupled accelerators:
  - GRAY: converts a RGB image into a grayscale image
  - MEAN: calculates the mean of a 2D matrix (columns)
  - MULTS: mutiplies a 2D matrix by its transpose
- We designed the accelerators and the shell in SystemC
- We used *Cadence Stratus HLS* for high-level synthesis and *Xilinx Vivado* for logic synthesis → Virtex-7 FPGA

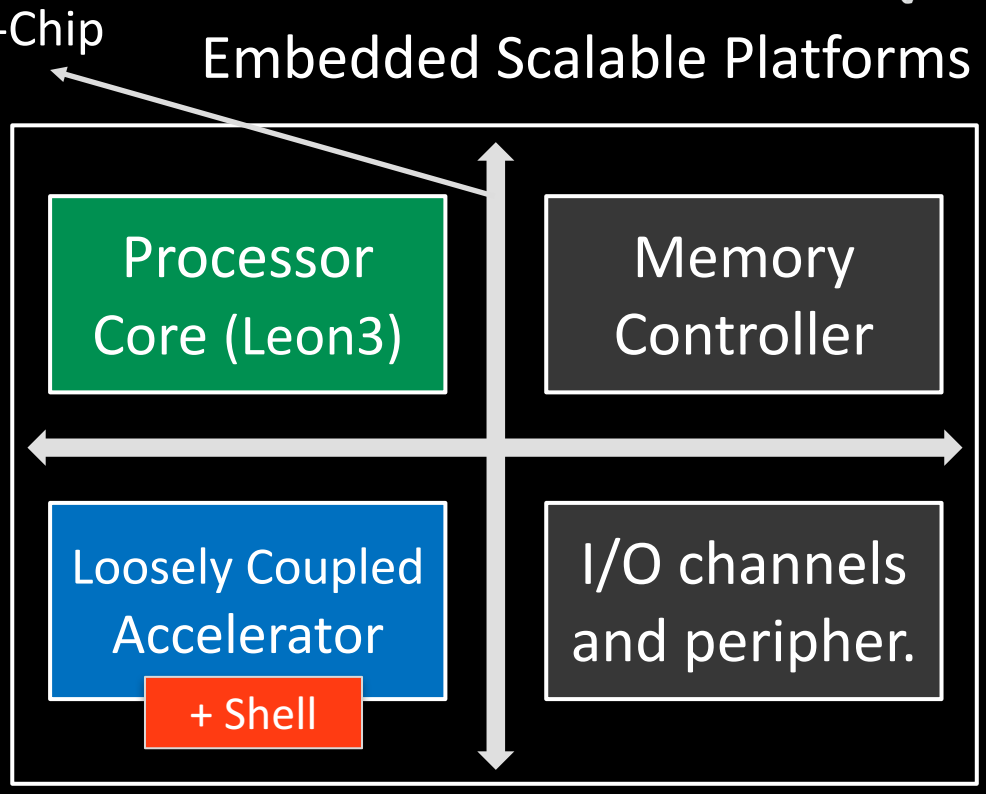
Experimental Results

Experimental Setup (2/2)



We explored different alternatives by varying:

- accelerator
- tag offset
- burst size
- workload
  - 128 x 128 - small
  - 512 x 512 - medium
  - 2048 x 2048 - large

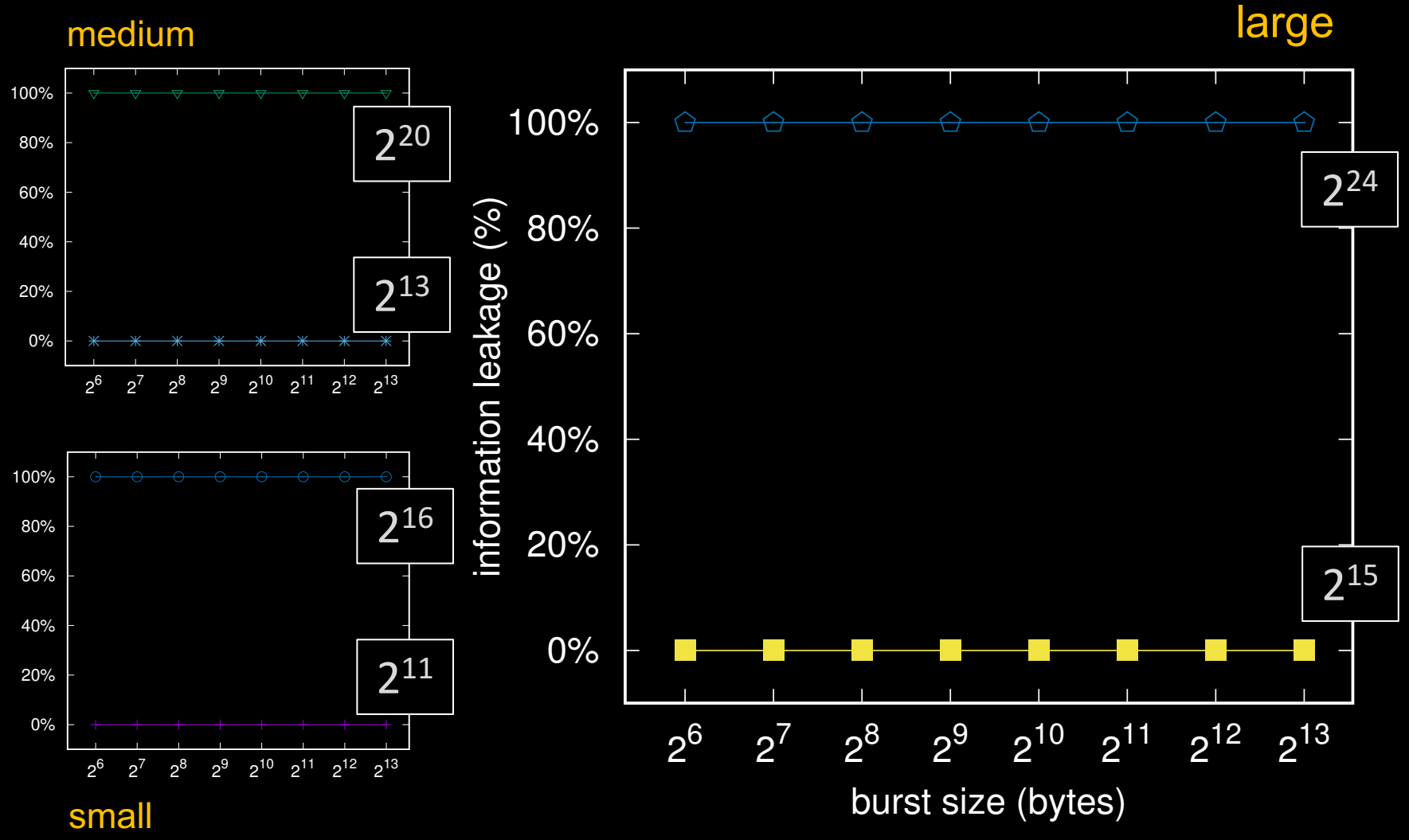


[P. Mantovani et al., ACM/IEEE DAC '16]

[L. P. Carloni, ACM/IEEE DAC '16]

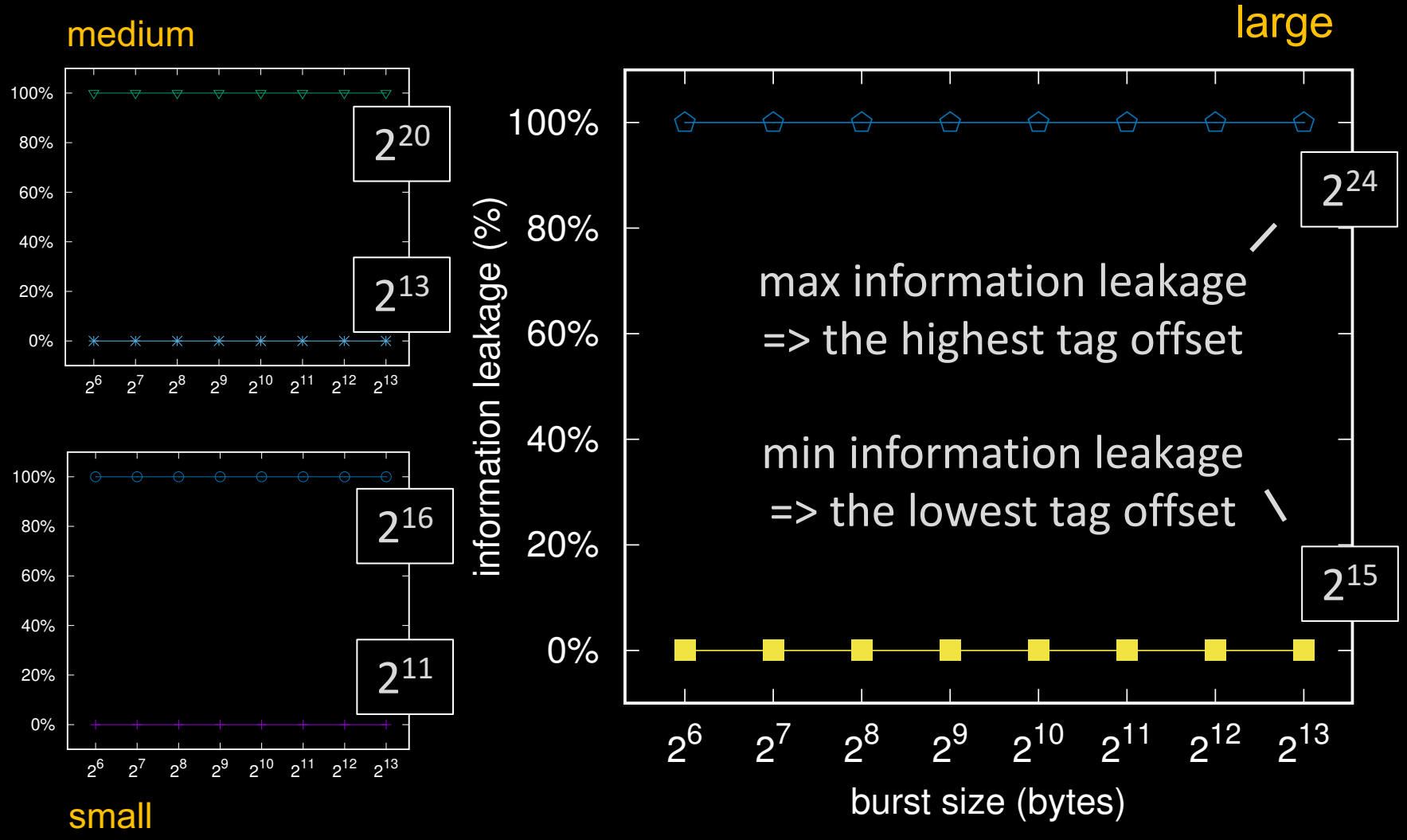
Experimental Results

Quantitative Security Analysis - MEAN



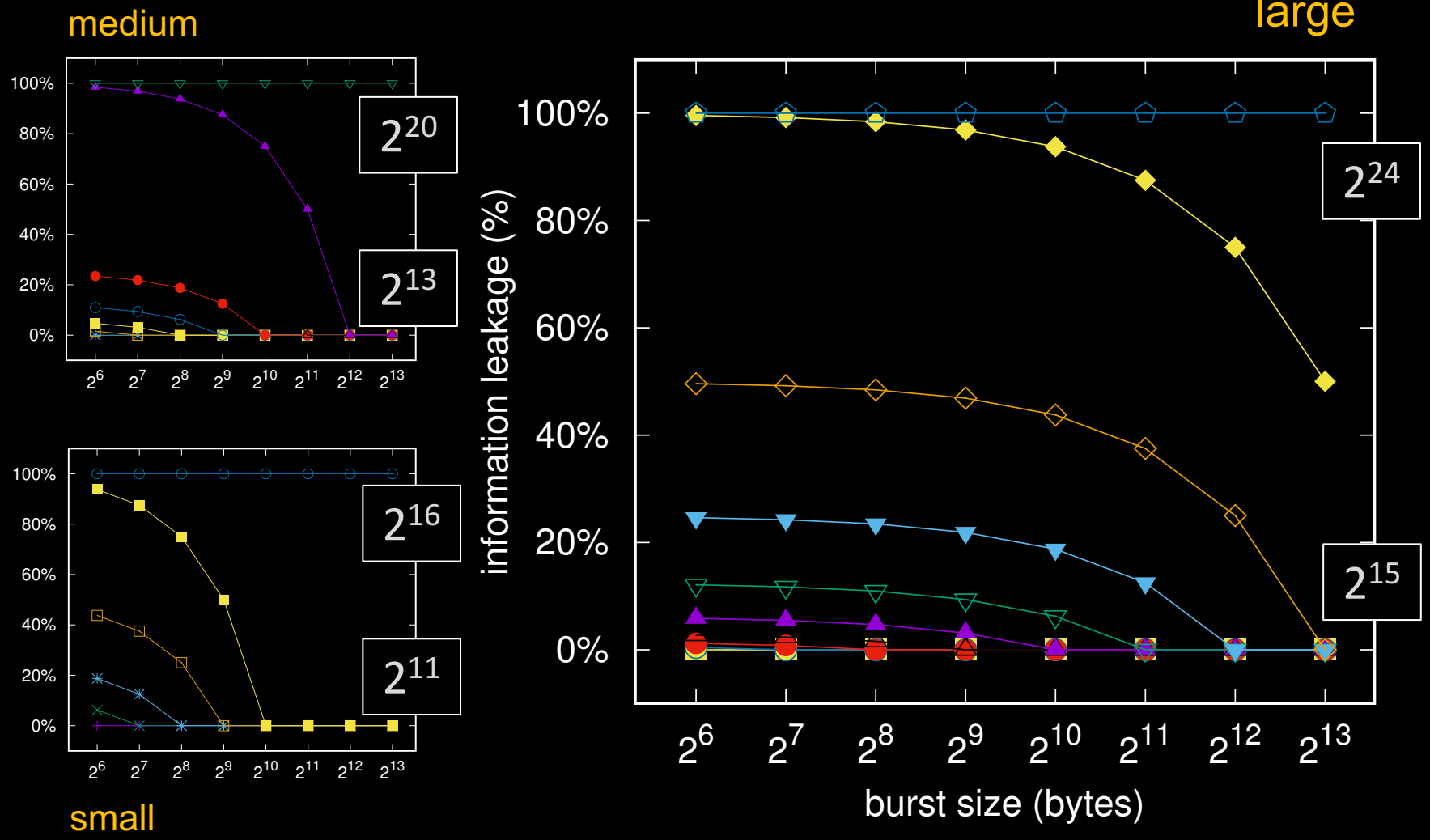
Experimental Results

Quantitative Security Analysis - MEAN



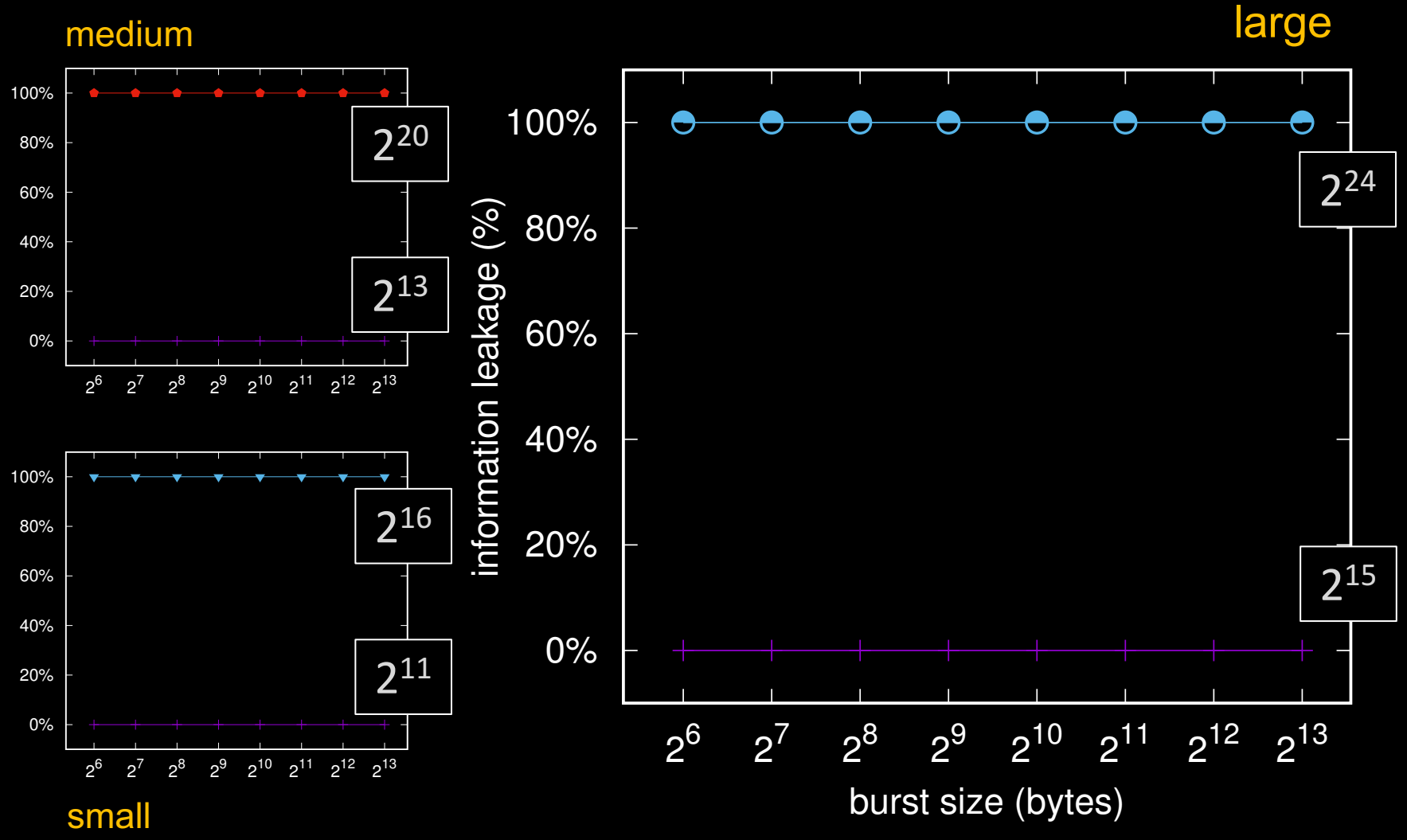
Experimental Results

Quantitative Security Analysis - MEAN



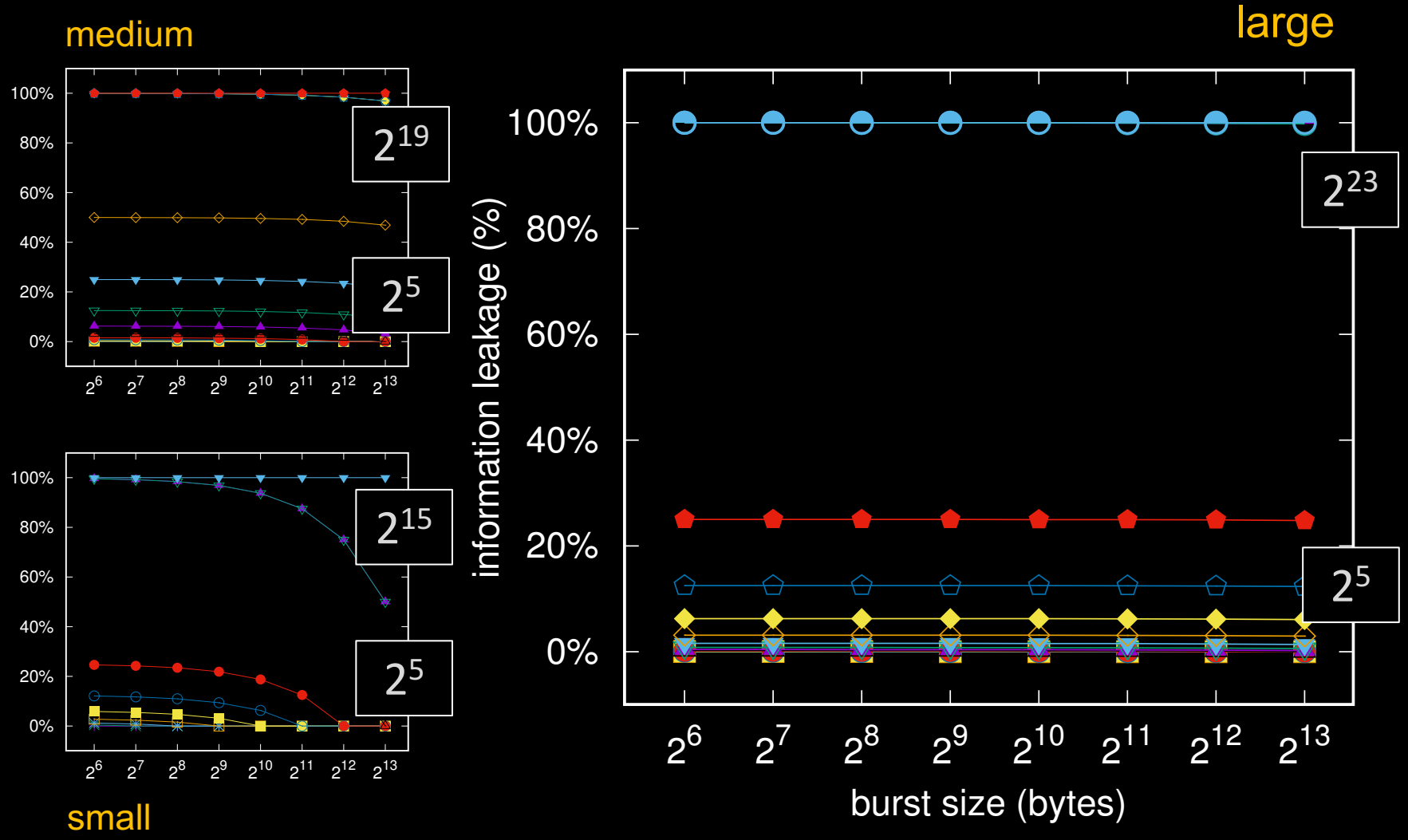
Experimental Results

Quantitative Security Analysis - GRAY



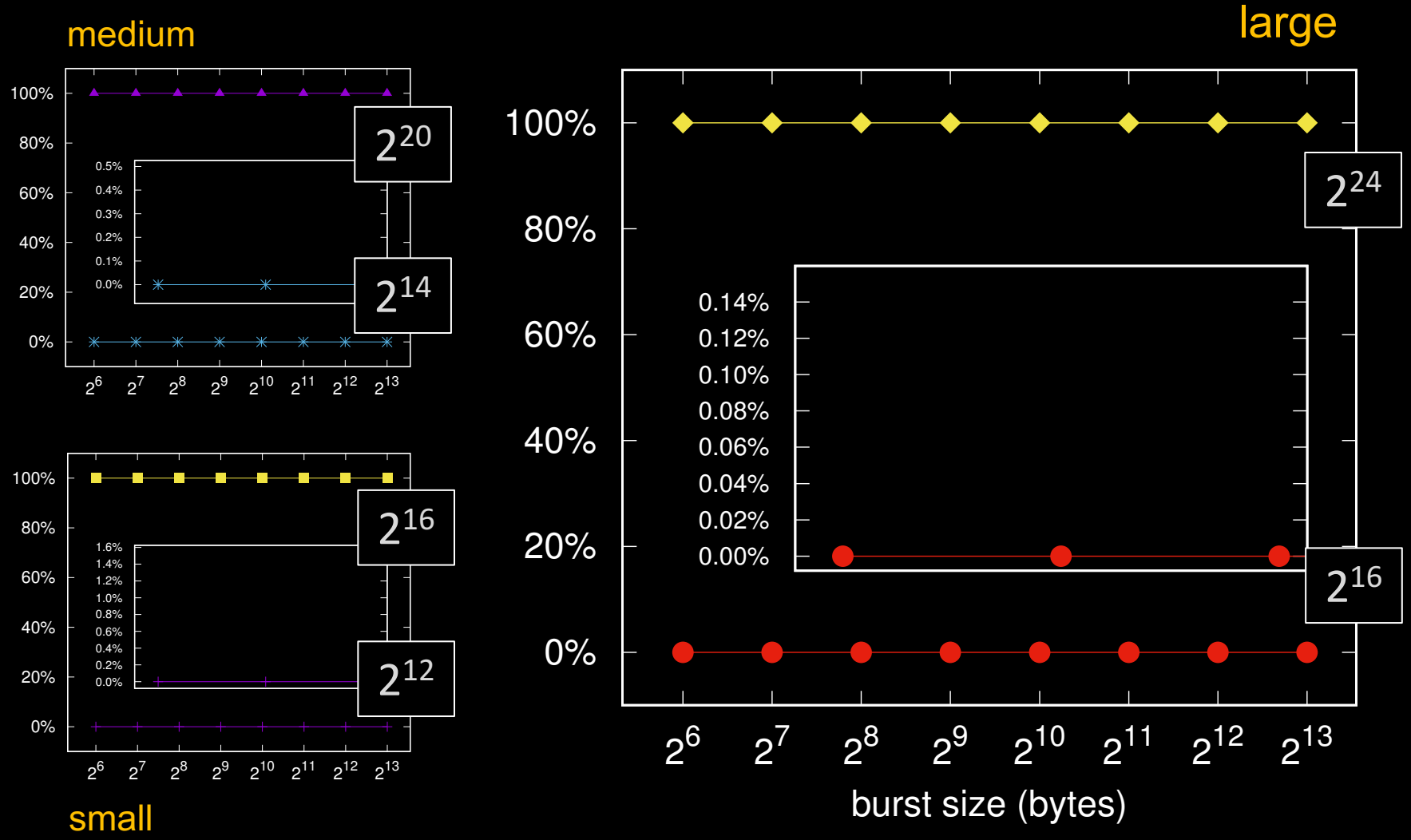
Experimental Results

Quantitative Security Analysis - GRAY



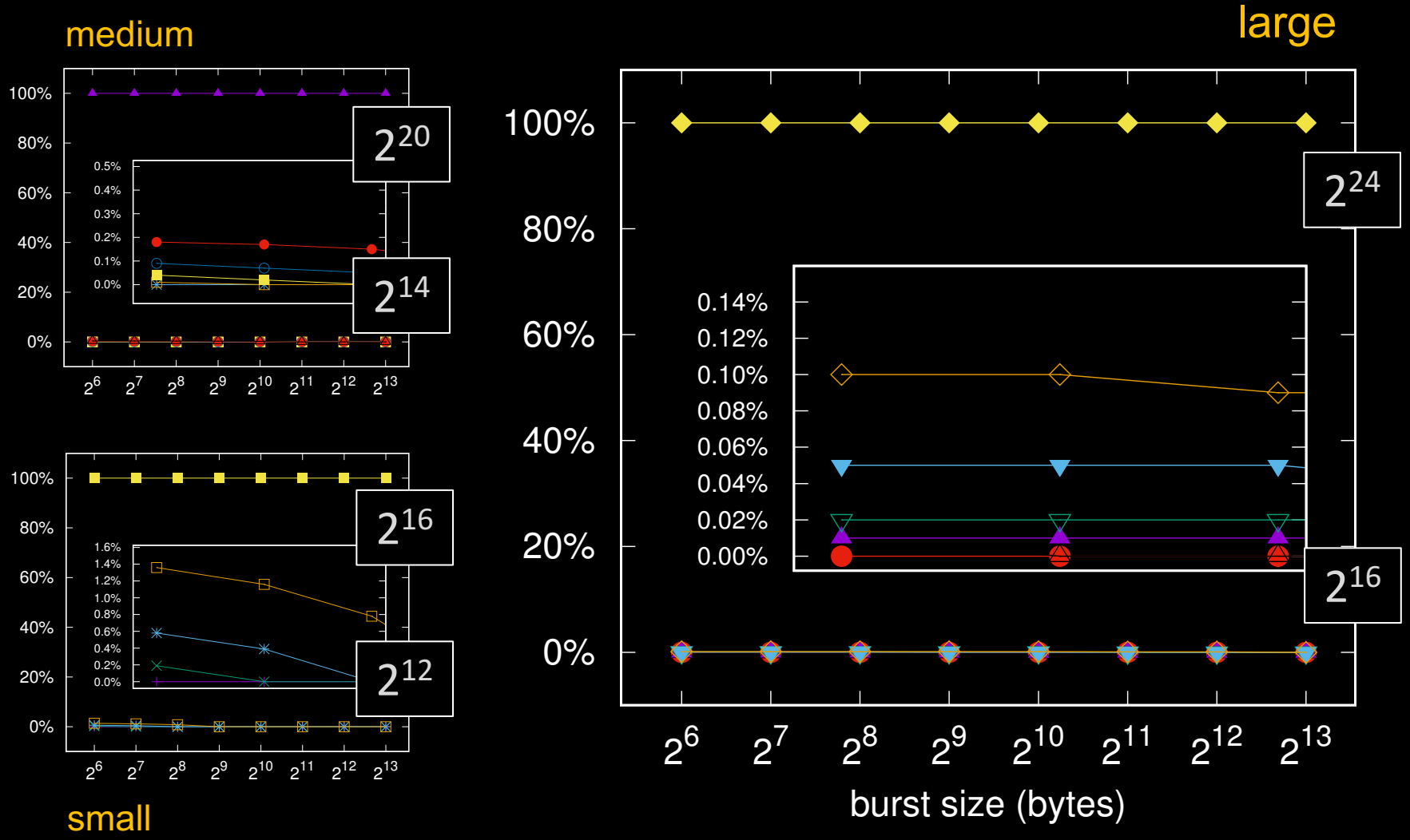
Experimental Results

Quantitative Security Analysis - MULTS



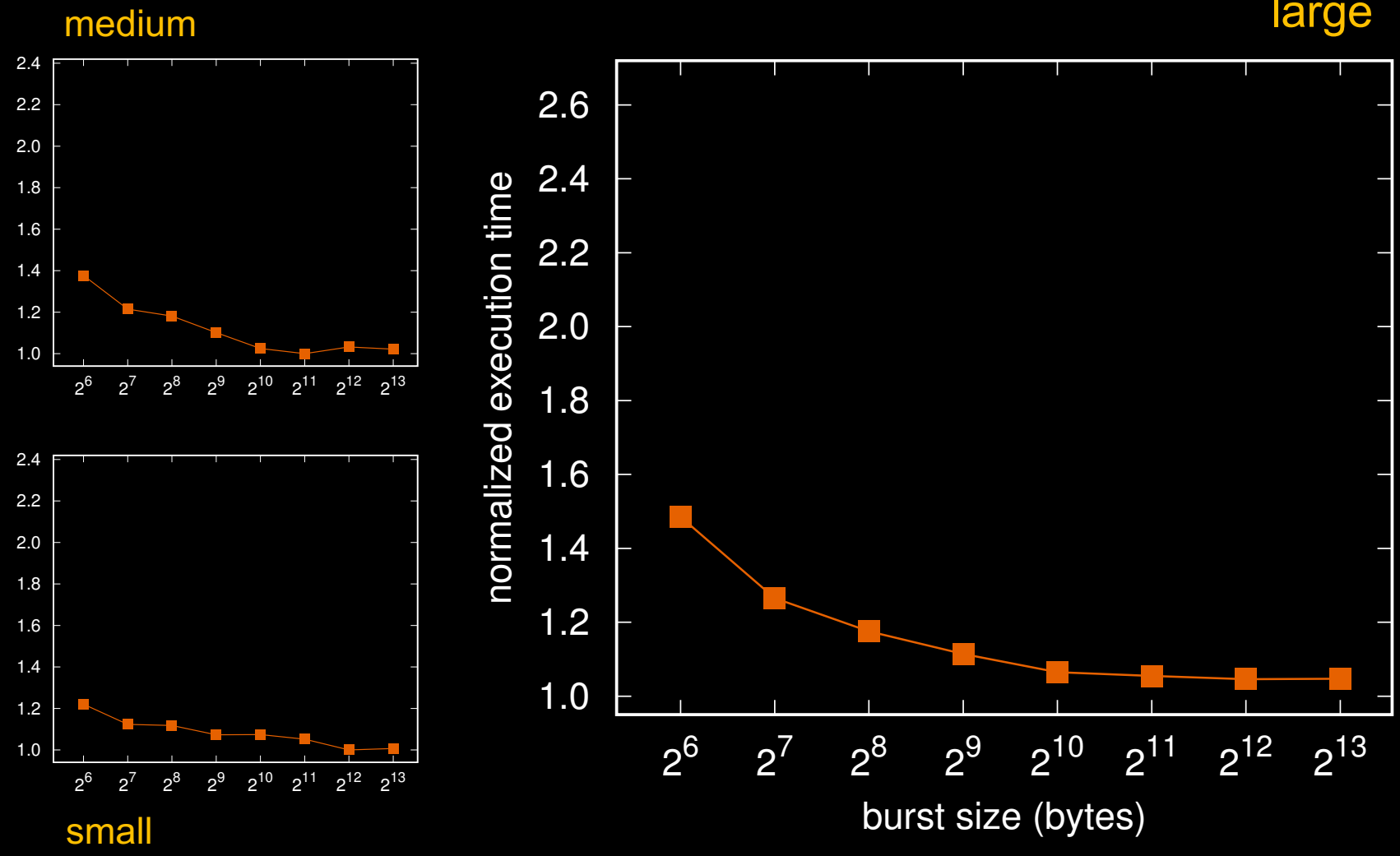
Experimental Results

Quantitative Security Analysis - MULTS



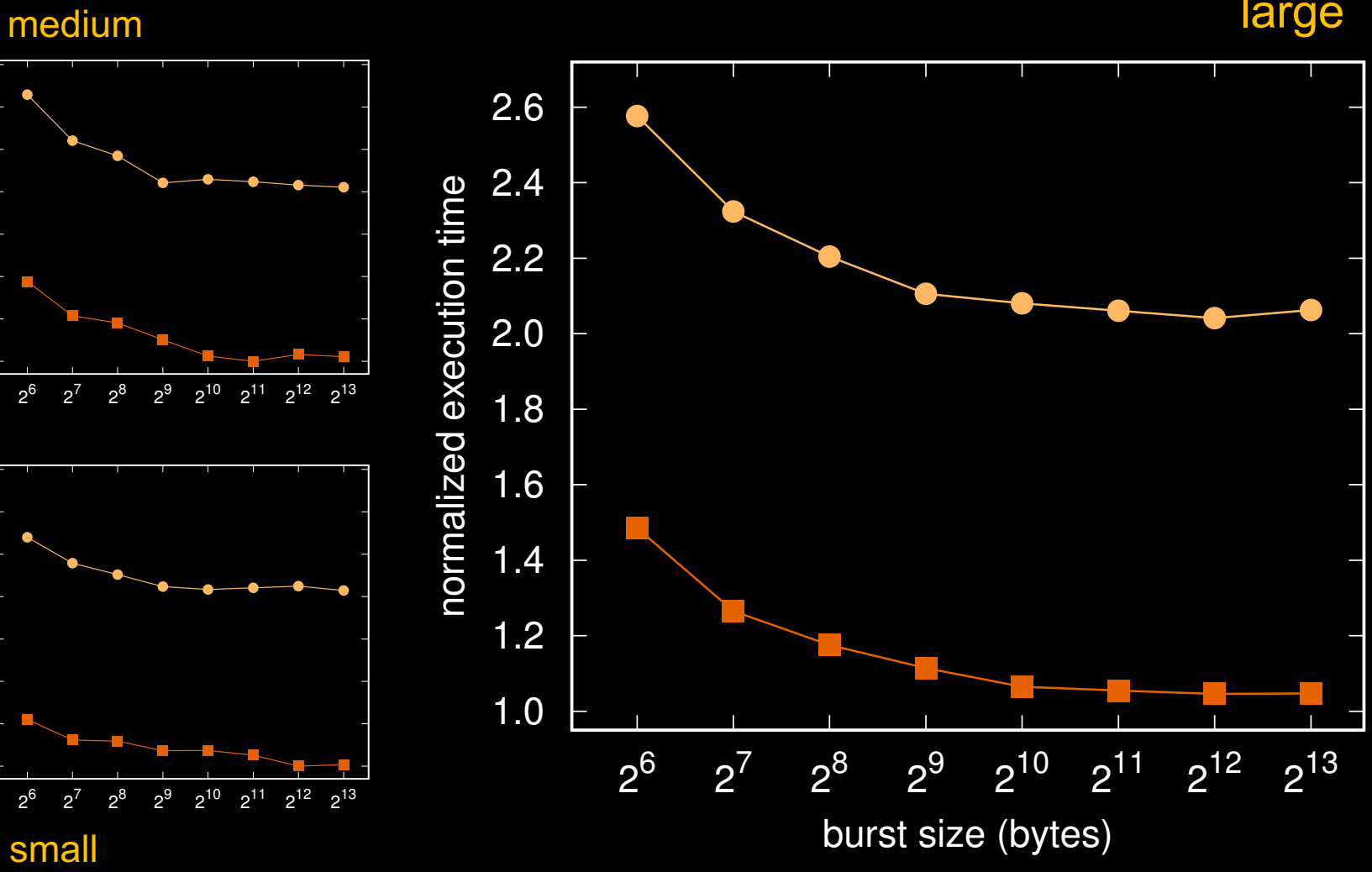
Experimental Results

Performance Analysis - GRAY



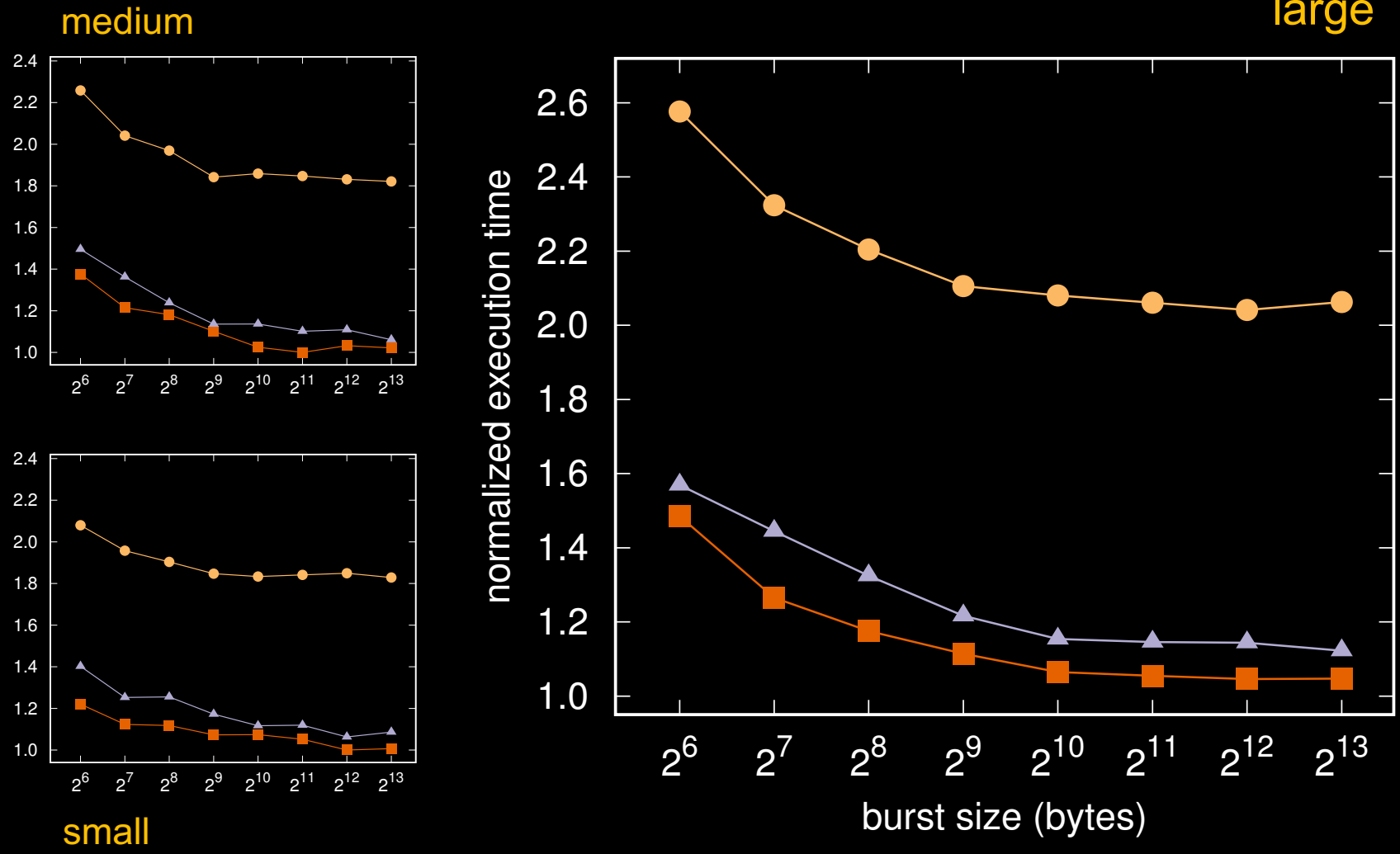
Experimental Results

Performance Analysis - GRAY



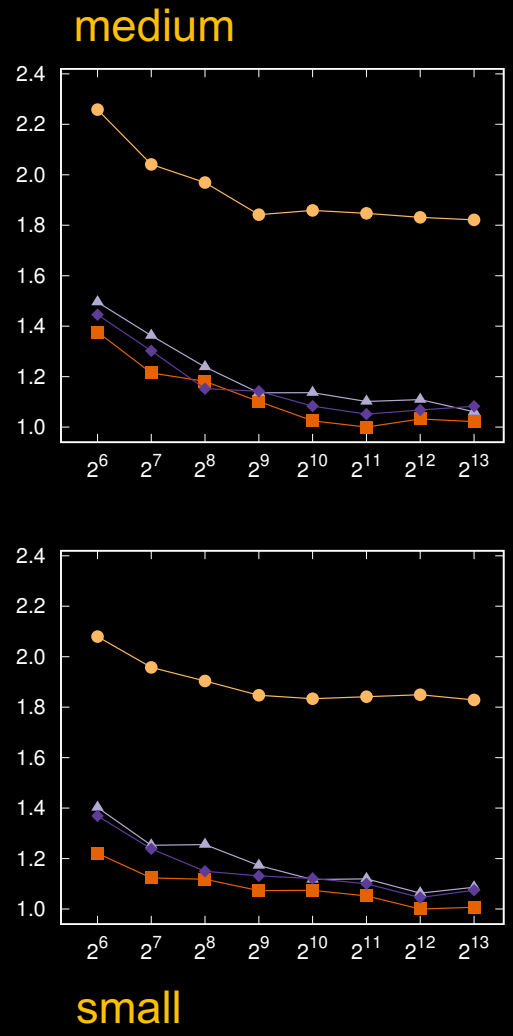
Experimental Results

Performance Analysis - GRAY



Experimental Results

Performance Analysis - GRAY



Conclusions

- We propose PAGURUS, a flexible methodology to design a shell that extends DIFT to accelerators
  1. The shell design is independent from the accelerator design and vice versa
  2. The shell has negligible cost overhead and reasonable performance overhead
- We define the metric of information leakage for accelerators to quantitatively measure security

PAGURUS: Low-Overhead Dynamic Information Flow Tracking on Loosely Couple Accelerators

Questions?

Speaker: Luca Piccolboni
Columbia University, NY