

# Open-Source Architecture Code Analyzer

Jan Laukemann

Erlangen National High Performance Computing Center (NHR@FAU)

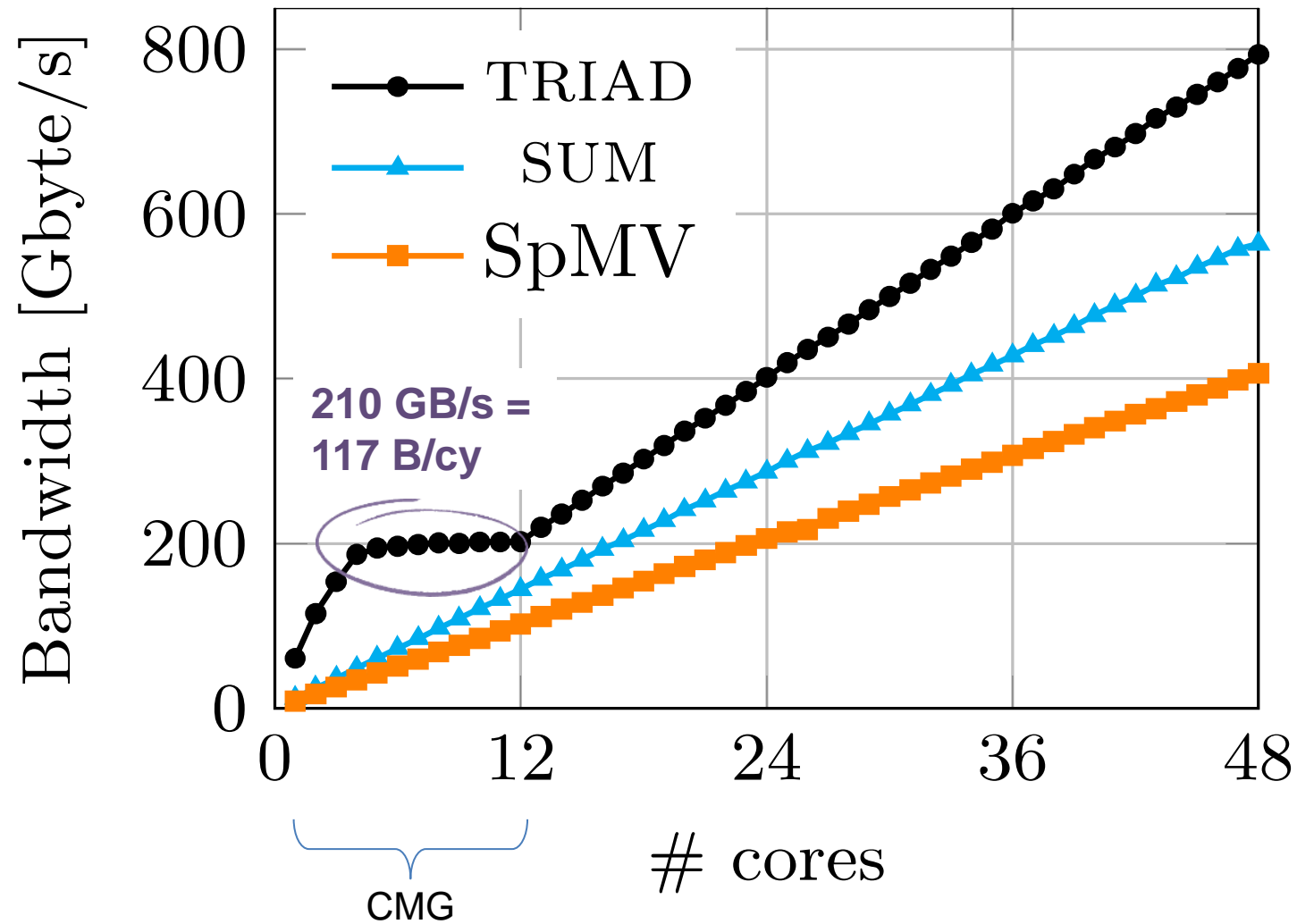
Friedrich-Alexander-University Erlangen-Nürnberg



Scalable Tools Workshop 2023, Lake Tahoe



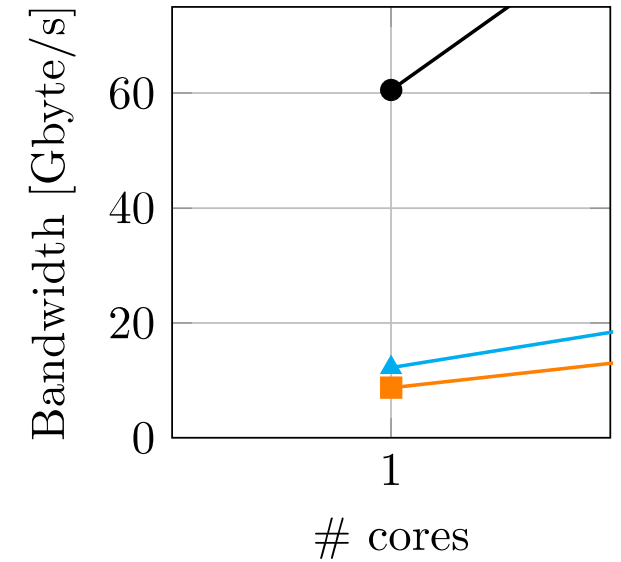
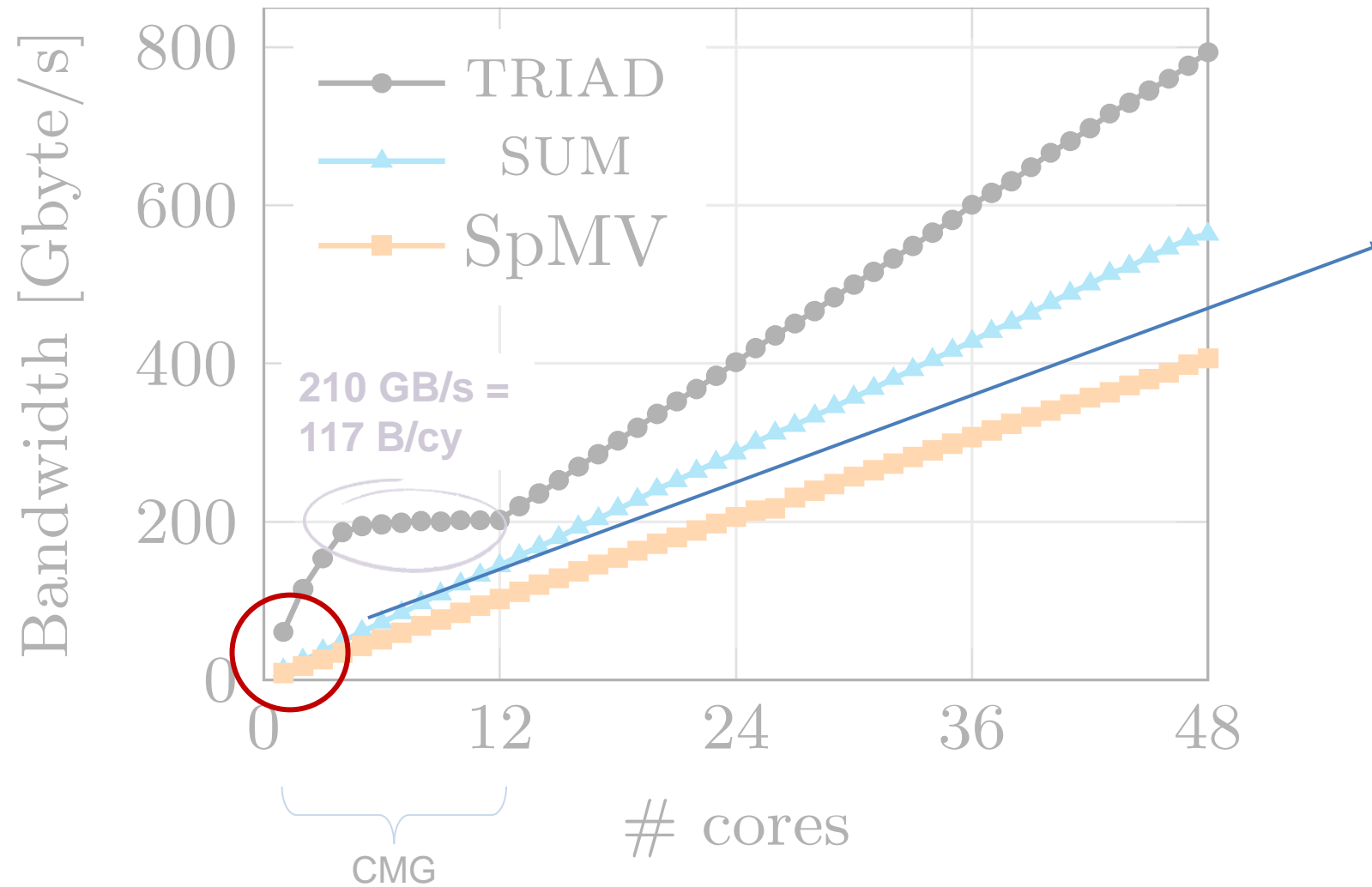
# Motivation



Thread pinning : Compact

C. Alappat, N. Meyer, J. Laukemann, T. Gruber, G. Hager, G. Wellein, and T. Wettig: *ECM modeling and performance tuning of SpMV and Lattice QCD on A64FX*. Concurrency and Computation: Practice and Experience, e6512 (2021). DOI: [10.1002/cpe.6512](https://doi.org/10.1002/cpe.6512)

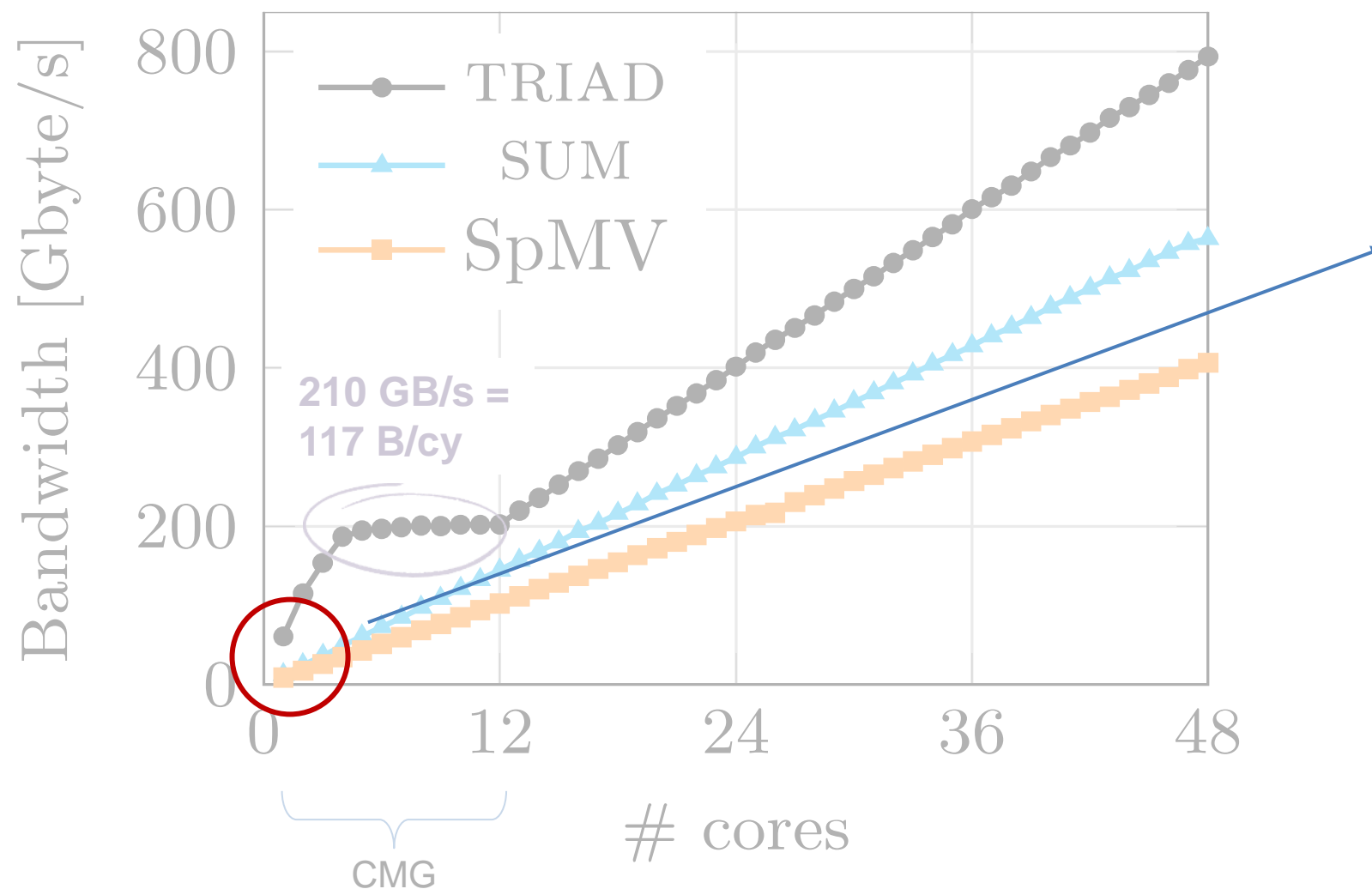
# Motivation



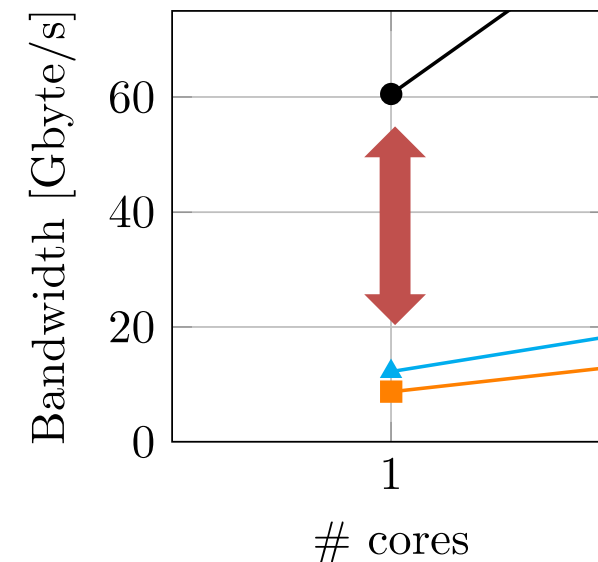
Thread pinning : Compact

C. Alappat, N. Meyer, J. Laukemann, T. Gruber, G. Hager, G. Wellein, and T. Wettig: *ECM modeling and performance tuning of SpMV and Lattice QCD on A64FX*. Concurrency and Computation: Practice and Experience, e6512 (2021). DOI: [10.1002/cpe.6512](https://doi.org/10.1002/cpe.6512)

# Motivation



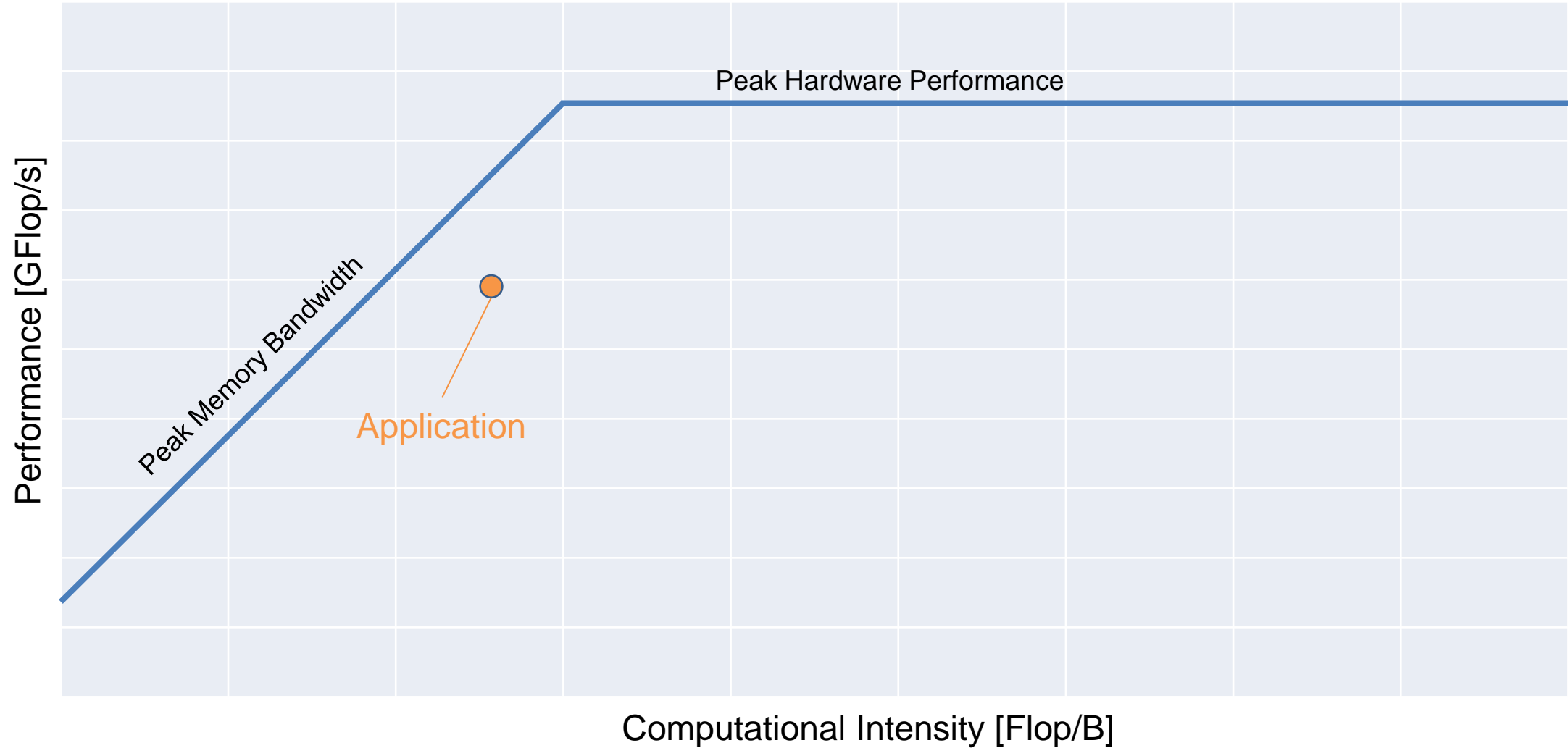
Thread pinning : Compact



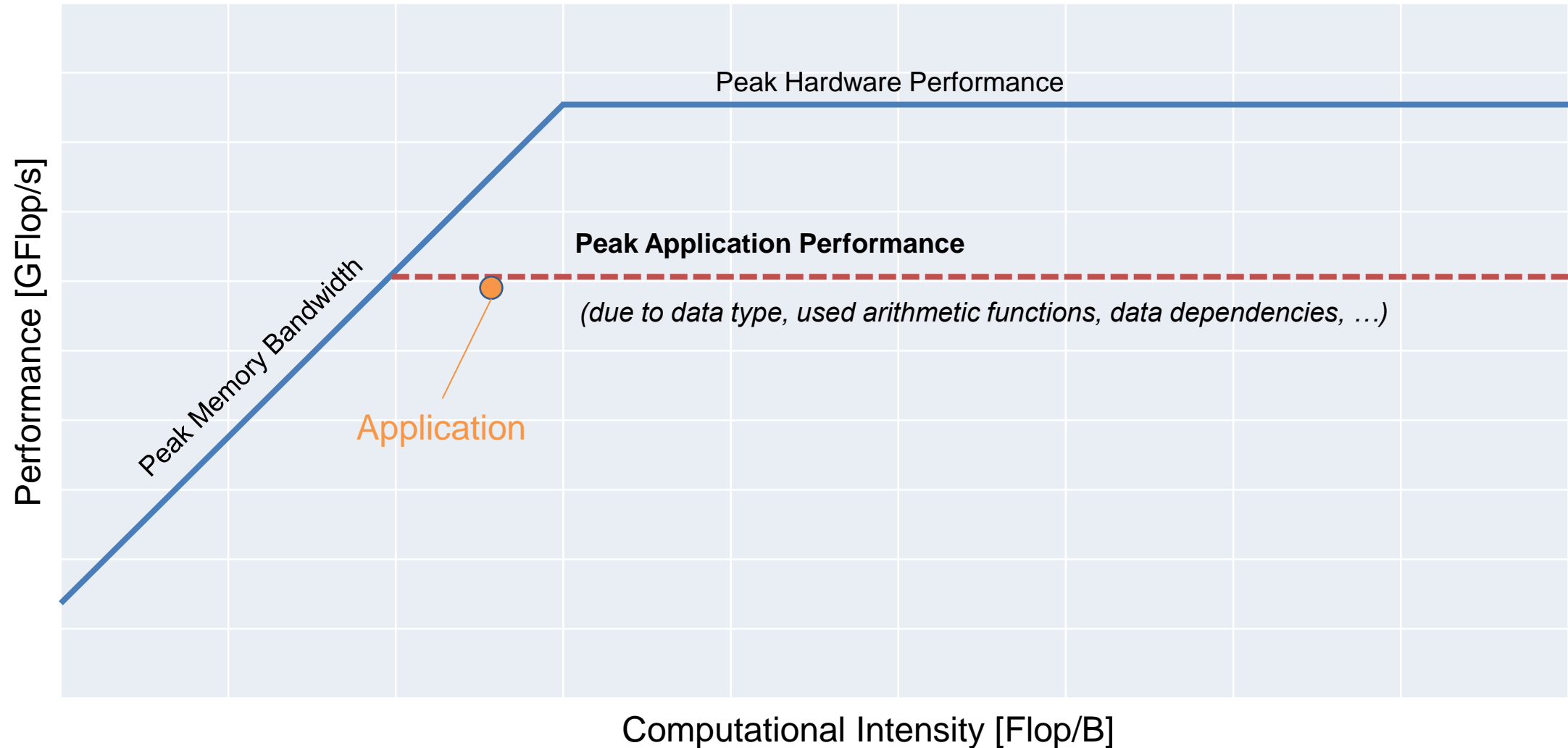
Understanding single-core performance is the key!

C. Alappat, N. Meyer, J. Laukemann, T. Gruber, G. Hager, G. Wellein, and T. Wettig: *ECM modeling and performance tuning of SpMV and Lattice QCD on A64FX*. Concurrency and Computation: Practice and Experience, e6512 (2021). DOI: [10.1002/cpe.6512](https://doi.org/10.1002/cpe.6512)

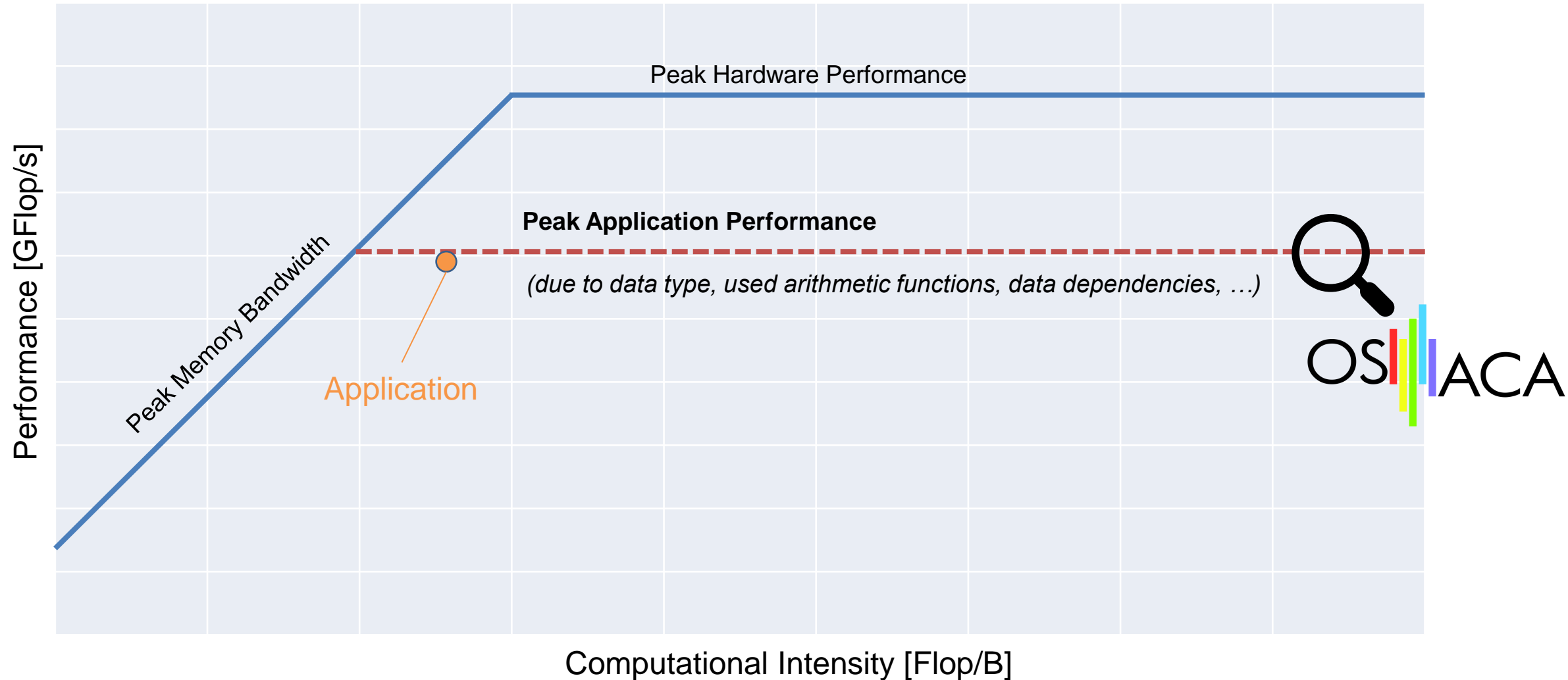
# Motivation



# Motivation

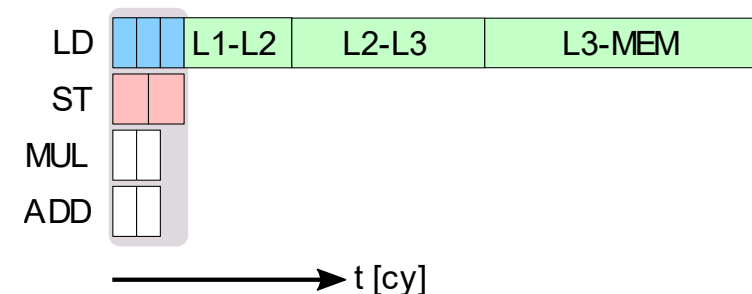
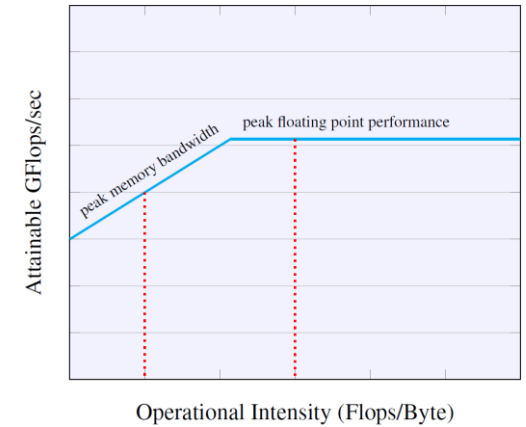


# Motivation



# Analytical Performance Modeling

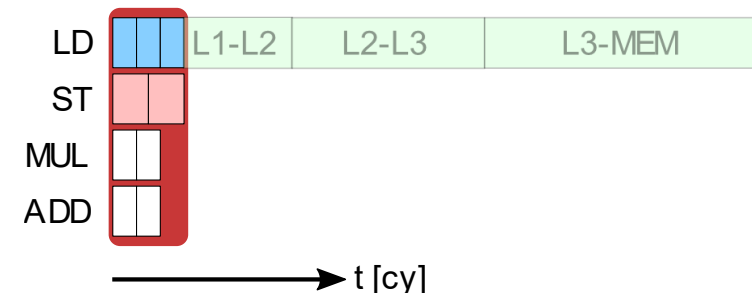
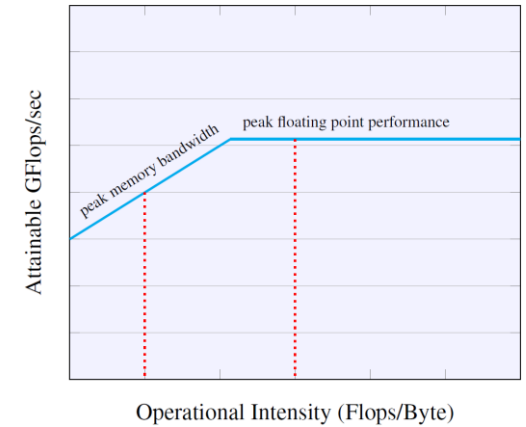
- What is the best performance my code can achieve?
- What are the relevant hardware bottlenecks?
- Apply simplified model of underlying hardware, consisting of
  - In-core execution
  - Data transfer





# Analytical Performance Modeling

- What is the best performance my code can achieve?
- What are the relevant hardware bottlenecks?
- Apply simplified model of underlying hardware, consisting of
  - **In-core execution**
  - Data transfer



# STREAM TRIAD on Intel Ice Lake

## STREAM TRIAD

$$a[i] = b[i] + s * c[i]$$

```
..B2.42:
vmovups    (%r14,%rdx,8), %zmm1
vfmadd213pd (%r15,%rdx,8), %zmm2, %zmm1
vmovupd    %zmm1, (%r12,%rdx,8)
addq       $8, %rdx
cmpq       %rsi, %rdx
jb         ..B2.42
```

# STREAM TRIAD on Intel Ice Lake

## STREAM TRIAD

$$a[i] = b[i] + s * c[i]$$

LOAD 5 cy



```
..B2.42:
vmovups    (%r14,%rdx,8), %zmm1
vfmadd213pd (%r15,%rdx,8), %zmm2, %zmm1
vmovupd    %zmm1, (%r12,%rdx,8)
addq       $8, %rdx
cmpq       %rsi, %rdx
jb         ..B2.42
```

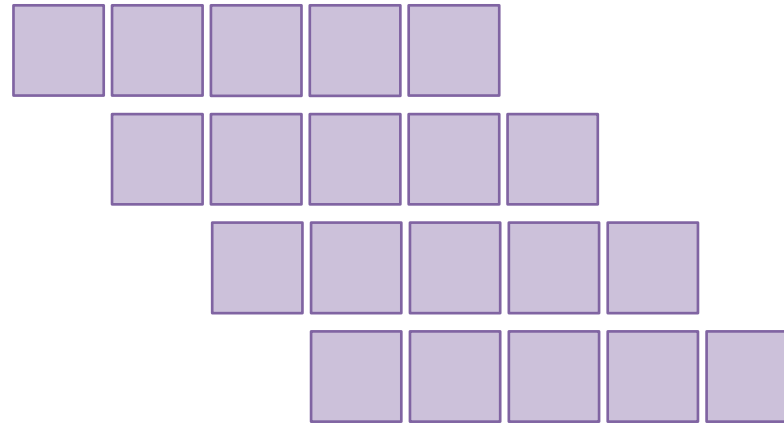
# STREAM TRIAD on Intel Ice Lake

## STREAM TRIAD

$$a[i] = b[i] + s * c[i]$$

```
..B2.42:  
vmovups    (%r14,%rdx,8), %zmm1  
vfmadd213pd (%r15,%rdx,8), %zmm2, %zmm1  
vmovupd    %zmm1, (%r12,%rdx,8)  
addq       $8, %rdx  
cmpq       %rsi, %rdx  
jb         ..B2.42
```

LOAD 5 cy



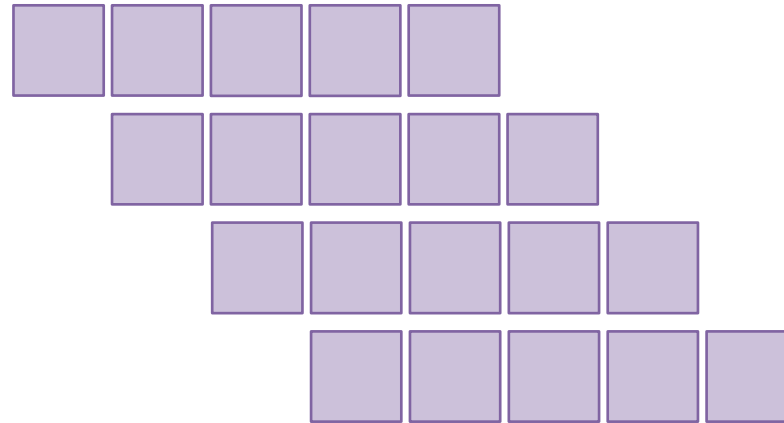
# STREAM TRIAD on Intel Ice Lake

## STREAM TRIAD

$$a[i] = b[i] + s * c[i]$$

```
..B2.42:  
vmovups    (%r14,%rdx,8), %zmm1  
vfmadd213pd (%r15,%rdx,8), %zmm2, %zmm1  
vmovupd    %zmm1, (%r12,%rdx,8)  
addq       $8, %rdx  
cmpq       %rsi, %rdx  
jb         ..B2.42
```

LOAD 5 cy

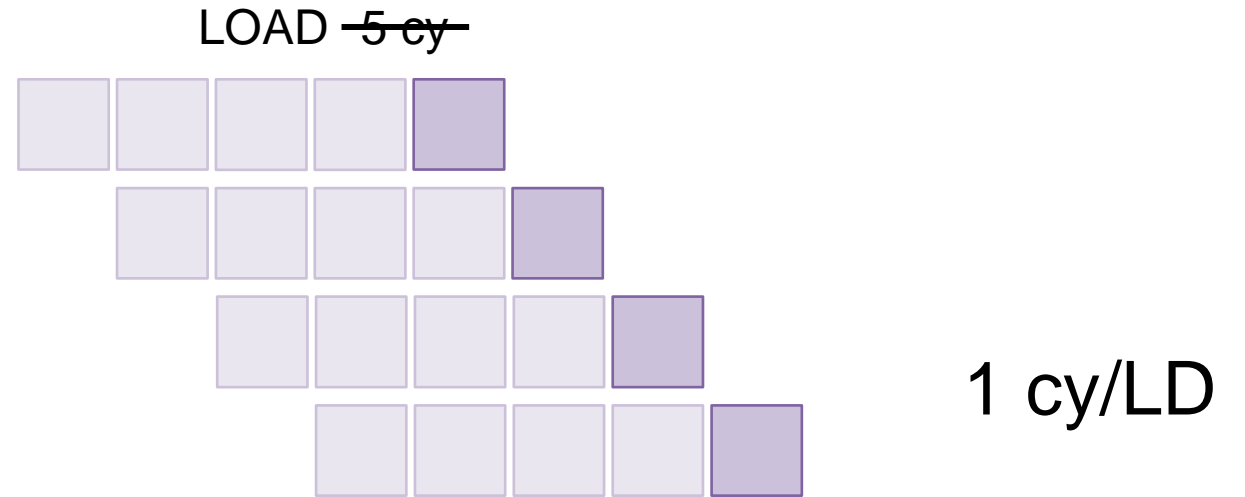


# STREAM TRIAD on Intel Ice Lake

## STREAM TRIAD

$$a[i] = b[i] + s * c[i]$$

```
..B2.42:
vmovups    (%r14,%rdx,8), %zmm1
vfmadd213pd (%r15,%rdx,8), %zmm2, %zmm1
vmovupd    %zmm1, (%r12,%rdx,8)
addq       $8, %rdx
cmpq       %rsi, %rdx
jb         ..B2.42
```

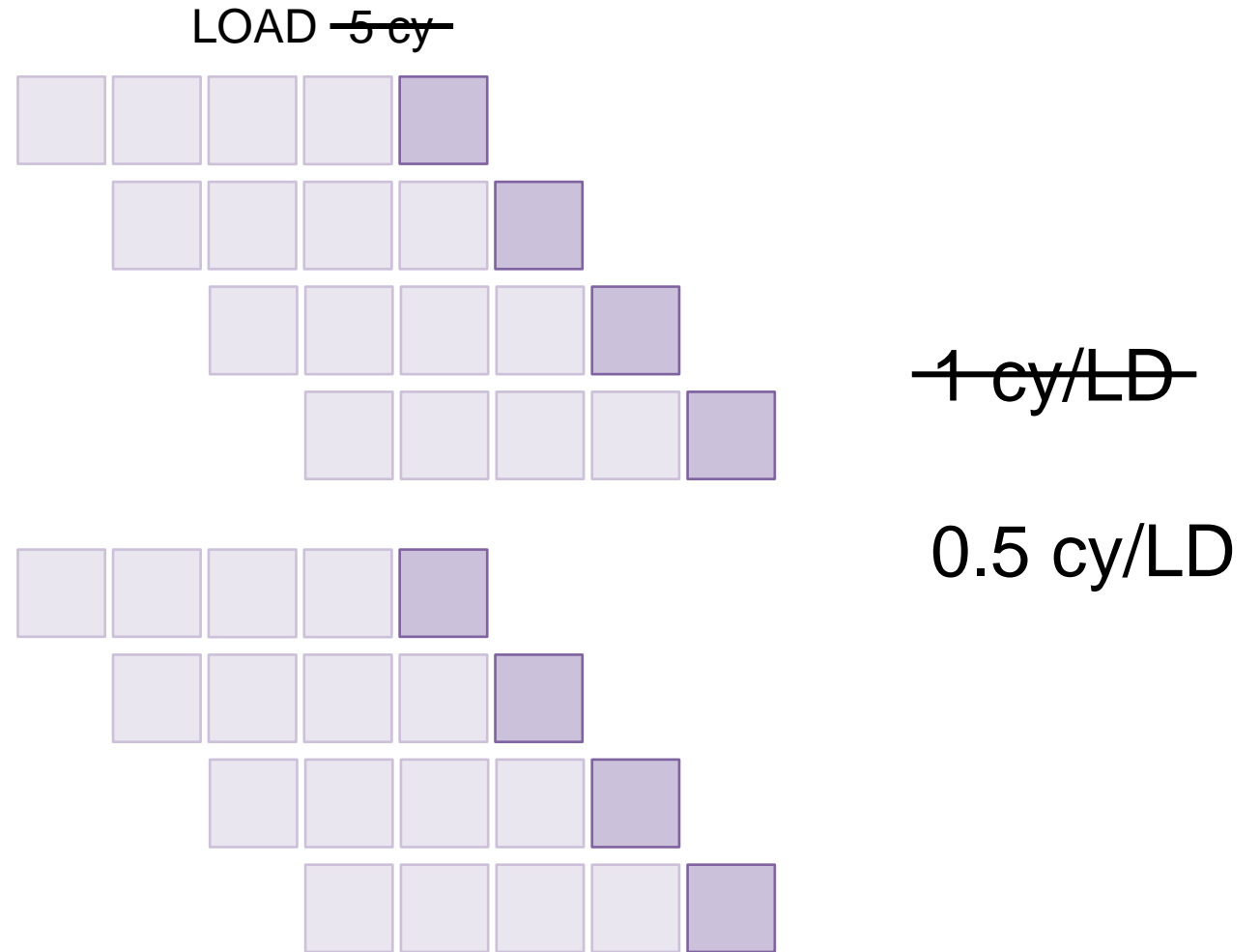


# STREAM TRIAD on Intel Ice Lake

## STREAM TRIAD

$a[i] = b[i] + s * c[i]$

```
..B2.42:  
vmovups    (%r14,%rdx,8), %zmm1  
vfmadd213pd (%r15,%rdx,8), %zmm2, %zmm1  
vmovupd    %zmm1, (%r12,%rdx,8)  
addq       $8, %rdx  
cmpq       %rsi, %rdx  
jb         ..B2.42
```



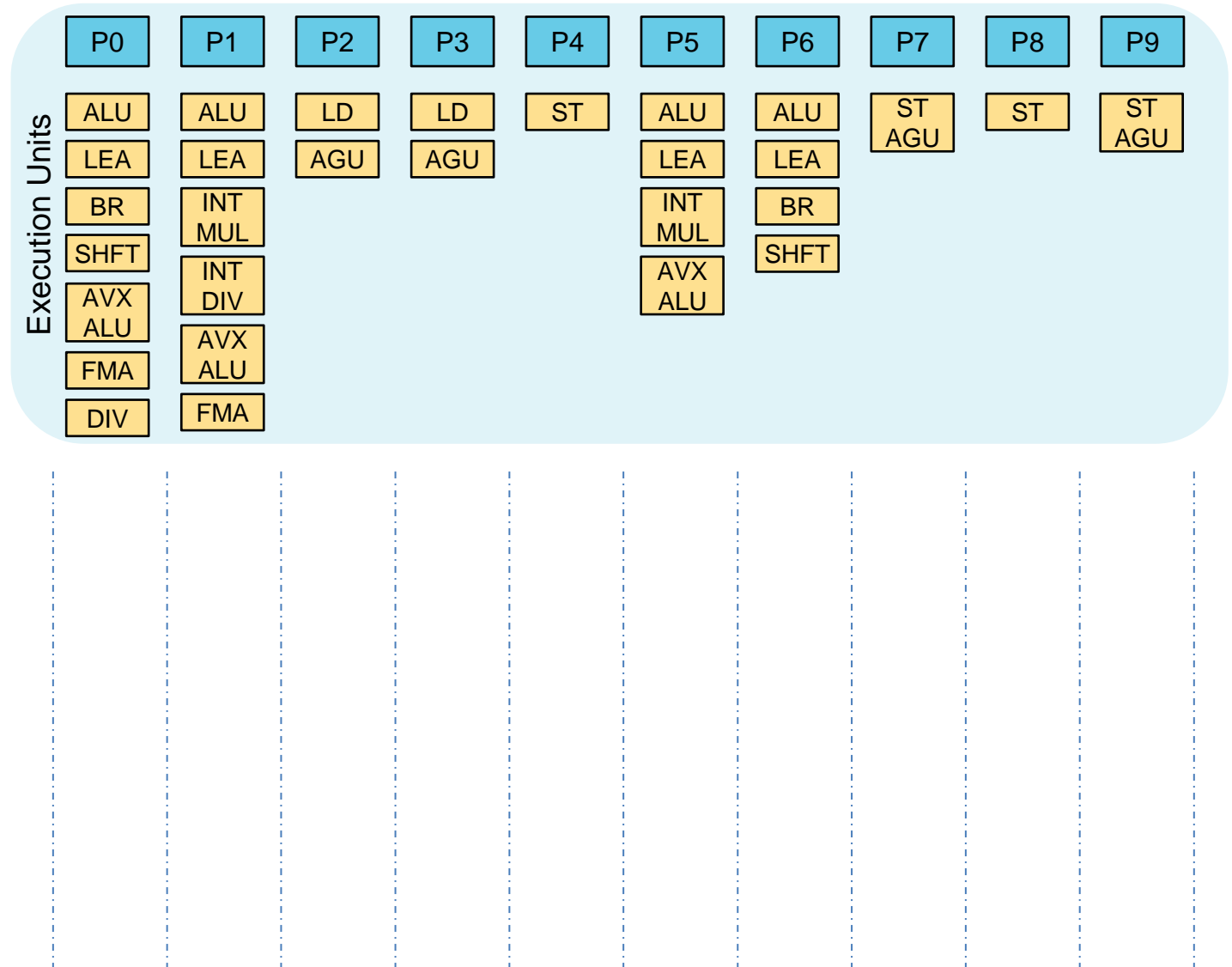
# STREAM TRIAD on Intel Ice Lake

## STREAM TRIAD

$a[i] = b[i] + s * c[i]$

..

```
vmovups    (%r14,%rdx,8), %zmm1
vfmadd213pd (%r15,%rdx,8), %zmm2, %zmm1
vmovupd    %zmm1, (%r12,%rdx,8)
addq       $8, %rdx
cmpq       %rsi, %rdx
jb         ..B2.42
```





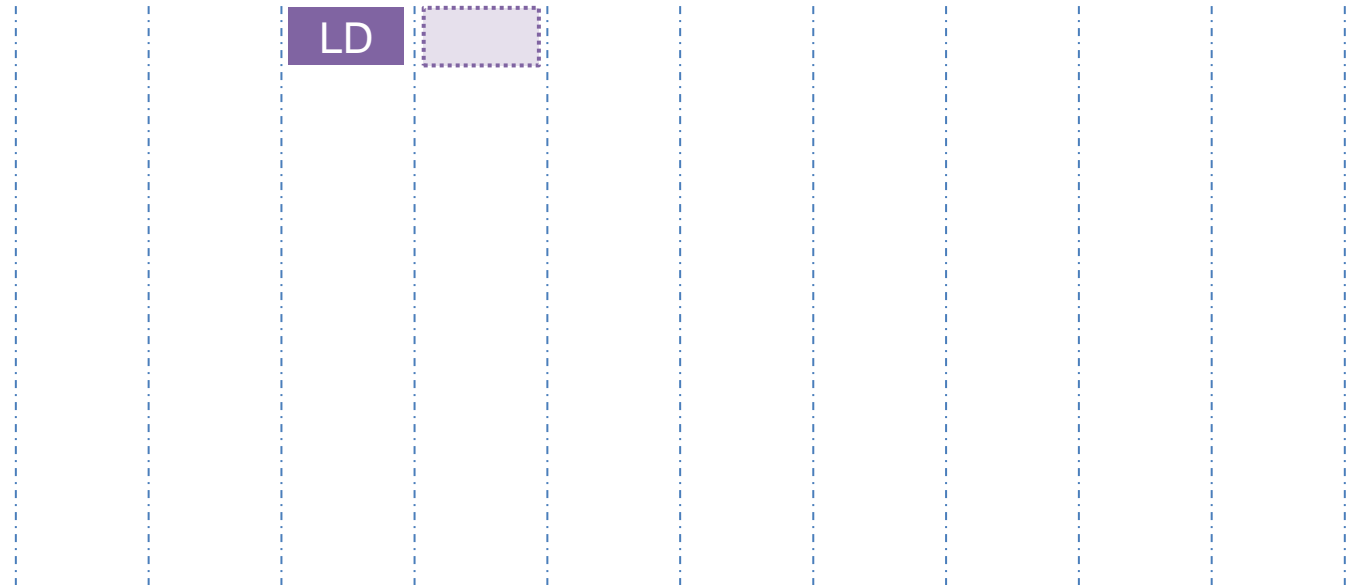
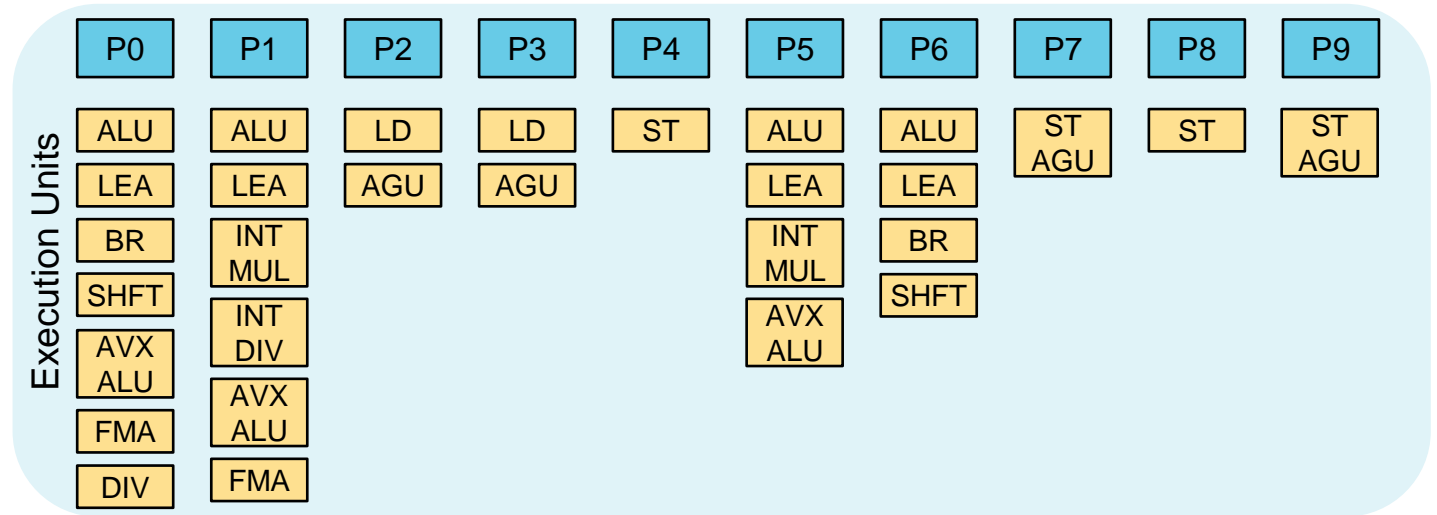
# STREAM TRIAD on Intel Ice Lake

## STREAM TRIAD

$a[i] = b[i] + s * c[i]$

..

```
vmovups    (%r14,%rdx,8), %zmm1
vfmadd213pd (%r15,%rdx,8), %zmm2, %zmm1
vmovupd    %zmm1, (%r12,%rdx,8)
addq       $8, %rdx
cmpq       %rsi, %rdx
jb         ..B2.42
```



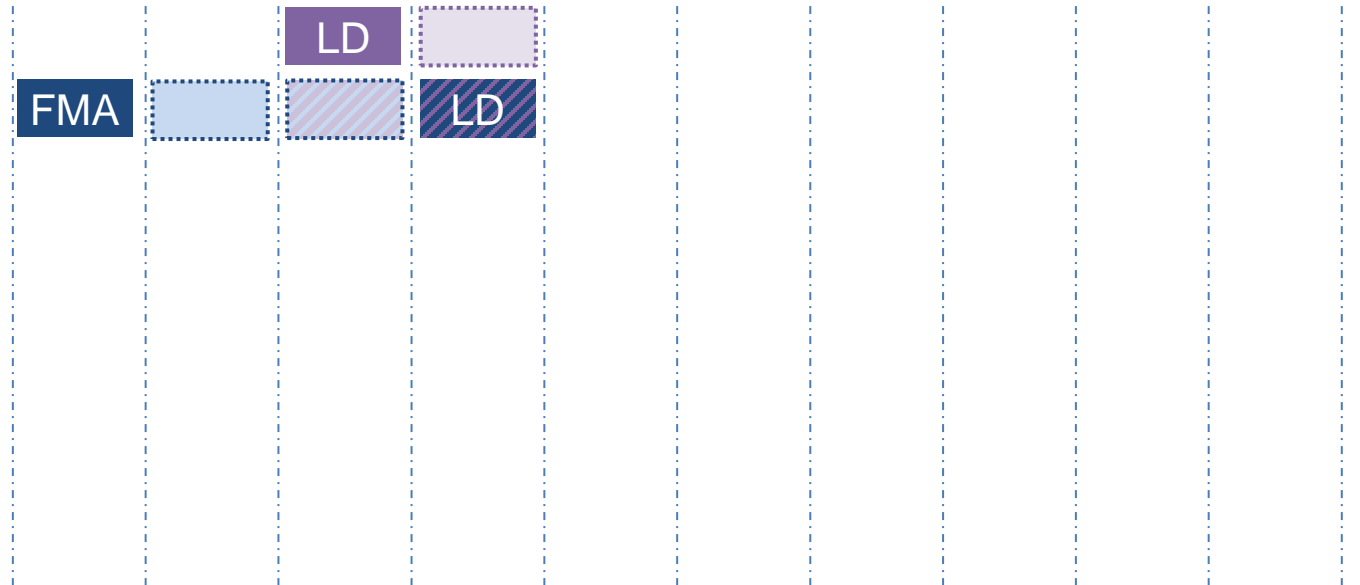
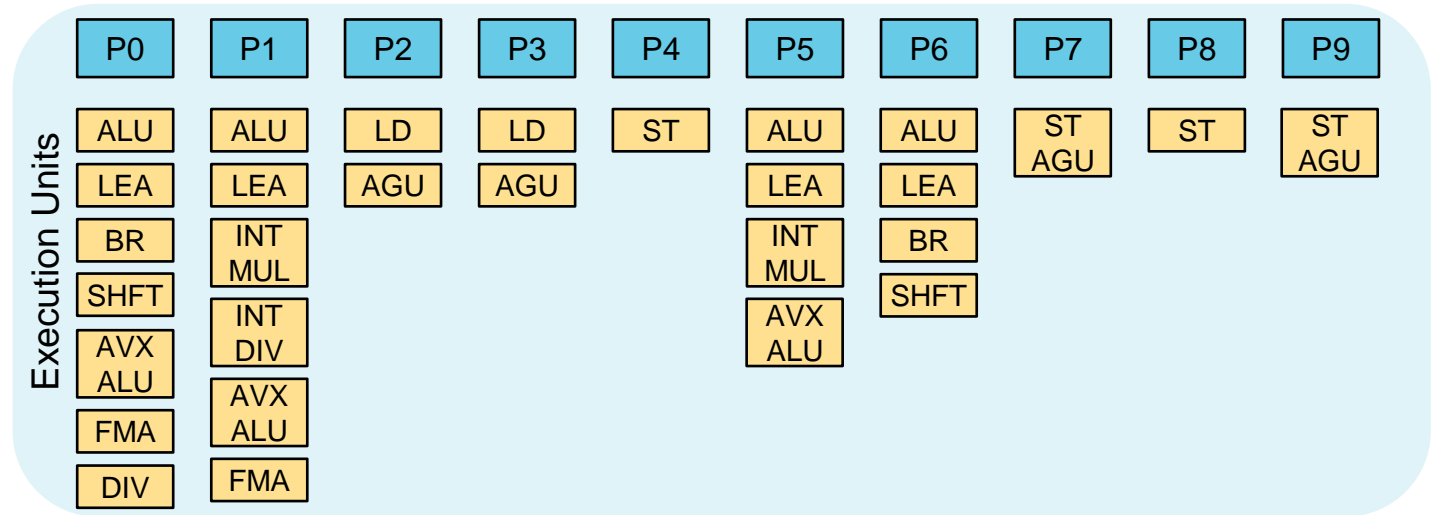
# STREAM TRIAD on Intel Ice Lake

## STREAM TRIAD

$a[i] = b[i] + s * c[i]$

..

```
vmovups    (%r14,%rdx,8), %zmm1
vfmadd213pd (%r15,%rdx,8), %zmm2, %zmm1
vmovupd    %zmm1, (%r12,%rdx,8)
addq       $8, %rdx
cmpq       %rsi, %rdx
jb         ..B2.42
```



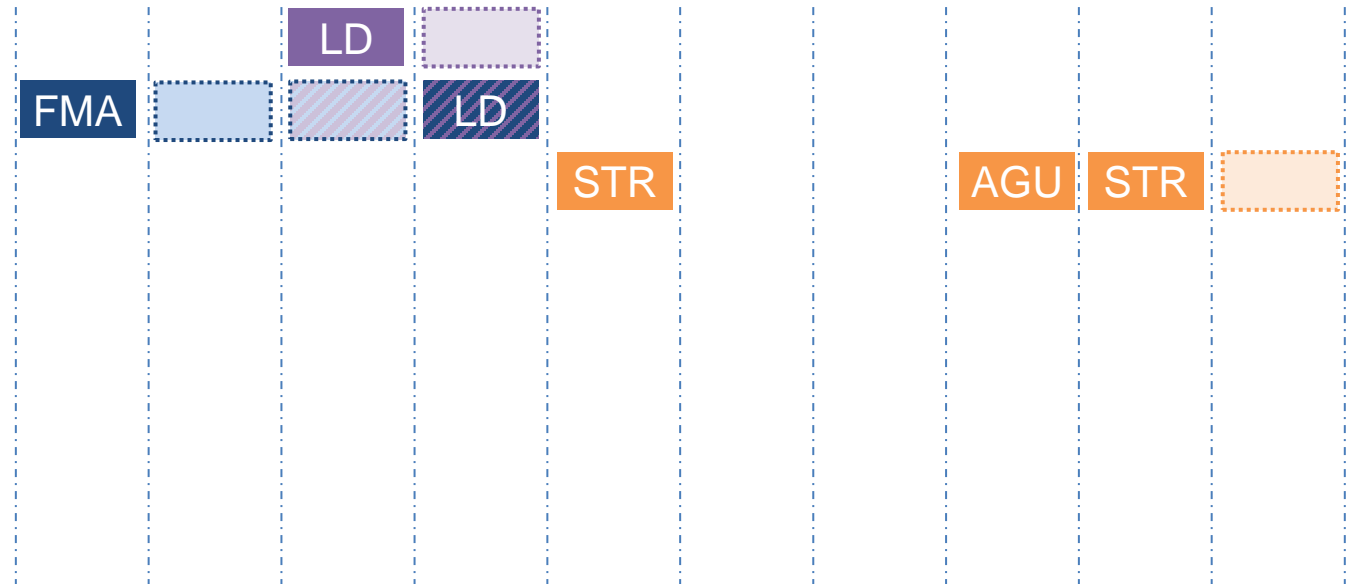
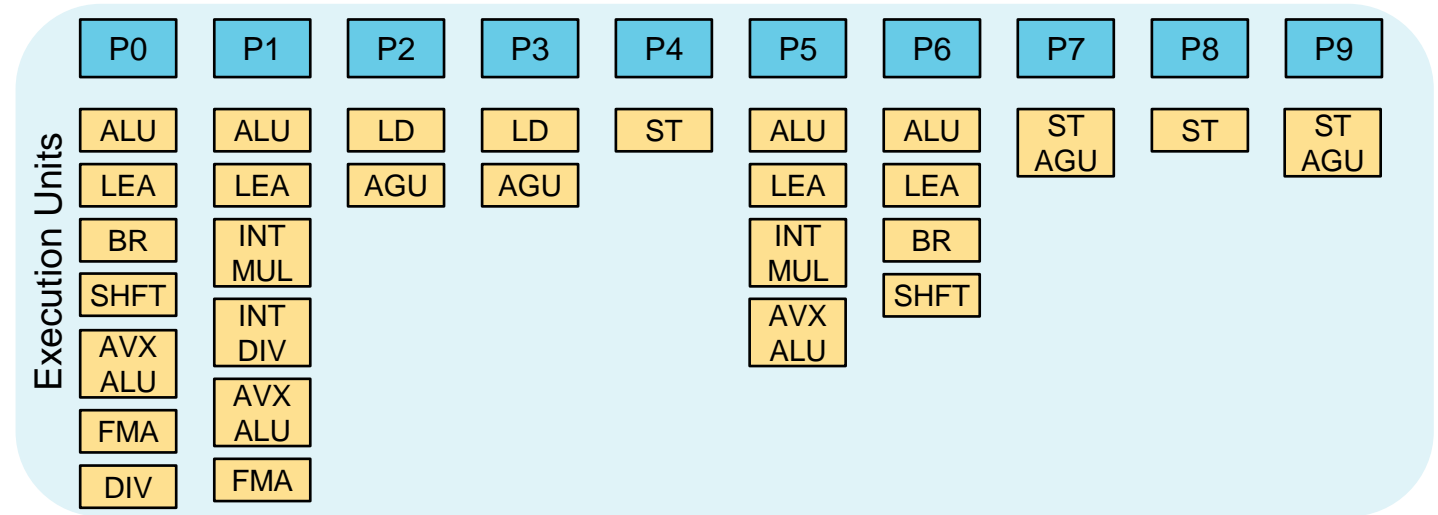
# STREAM TRIAD on Intel Ice Lake

## STREAM TRIAD

$a[i] = b[i] + s * c[i]$

..

```
vmovups    (%r14,%rdx,8), %zmm1
vfmadd213pd (%r15,%rdx,8), %zmm2, %zmm1
vmovupd    %zmm1, (%r12,%rdx,8)
addq       $8, %rdx
cmpq       %rsi, %rdx
jb         ..B2.42
```



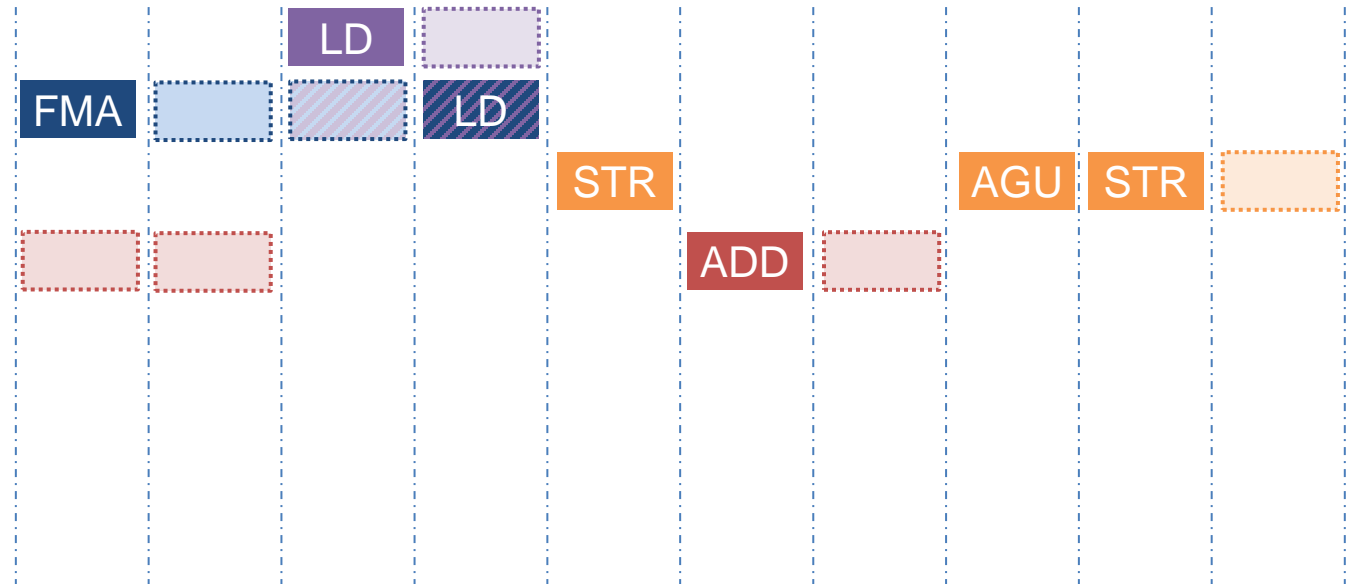
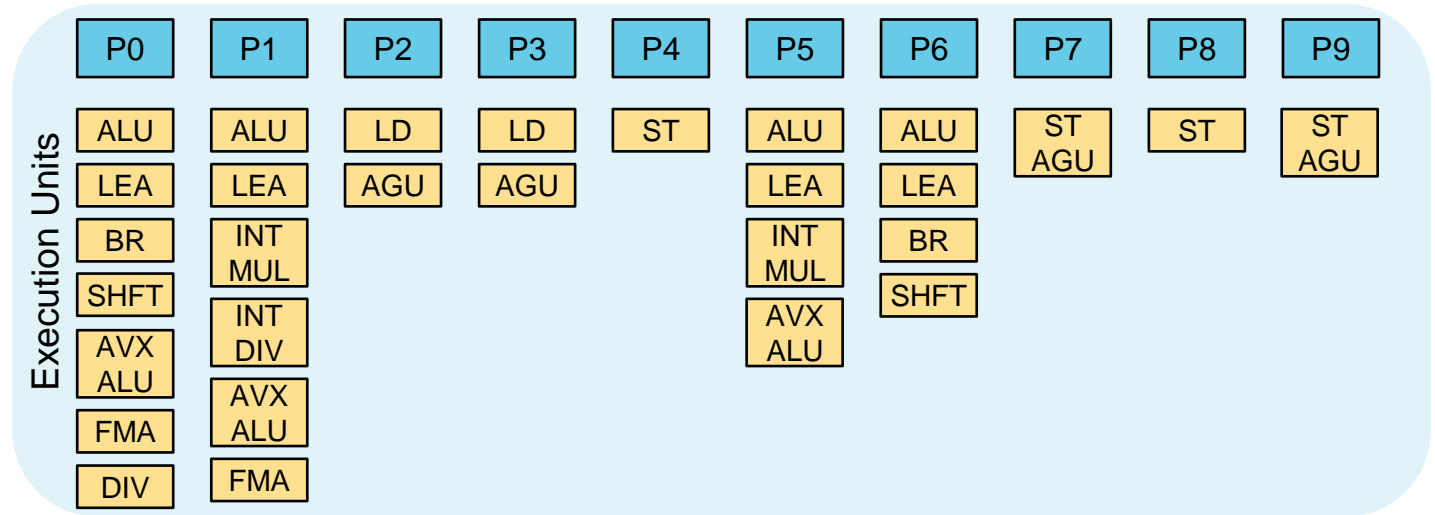
# STREAM TRIAD on Intel Ice Lake

## STREAM TRIAD

$a[i] = b[i] + s * c[i]$

..

```
vmovups    (%r14,%rdx,8), %zmm1
vfmadd213pd (%r15,%rdx,8), %zmm2, %zmm1
vmovupd    %zmm1, (%r12,%rdx,8)
addq       $8, %rdx
cmpq       %rsi, %rdx
jb         ..B2.42
```



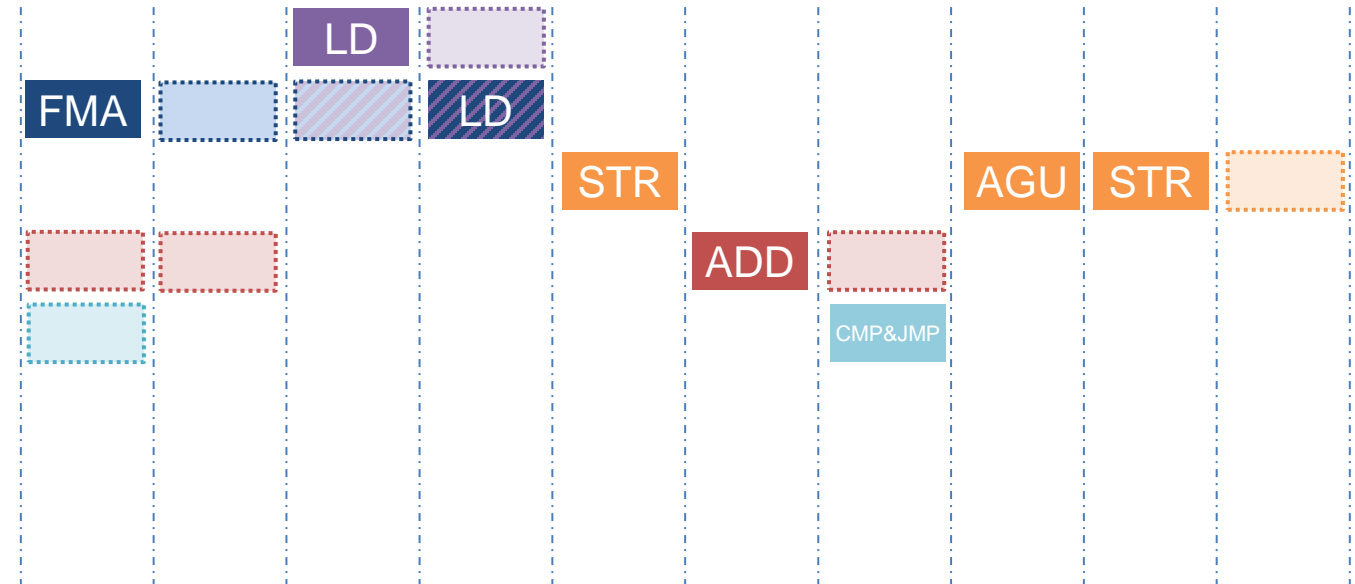
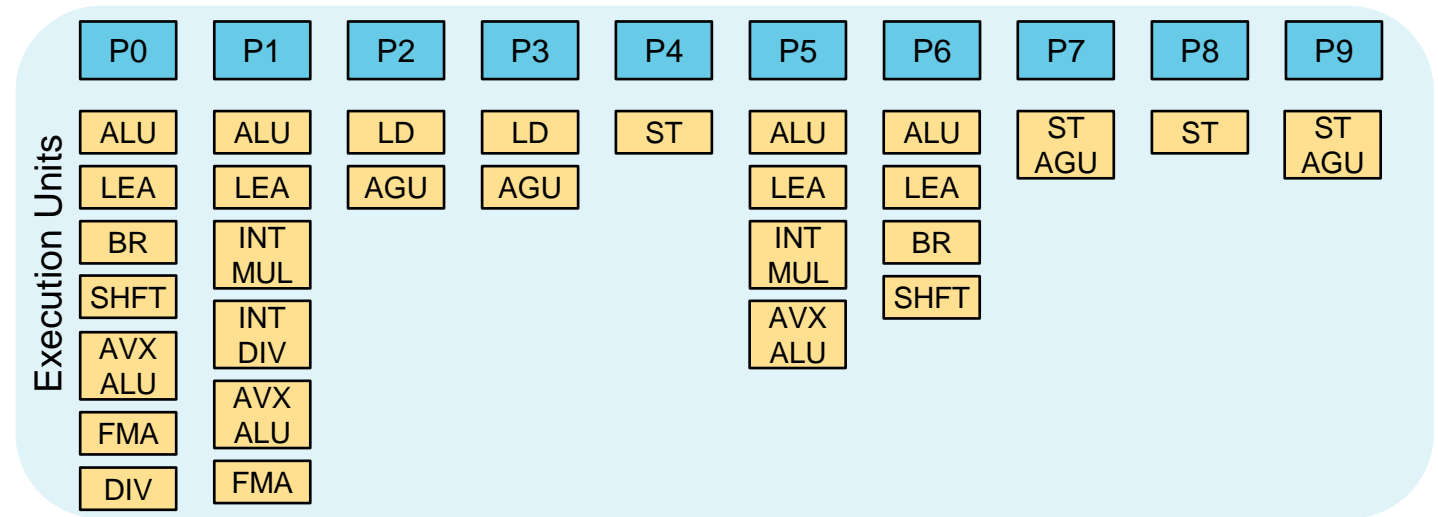
# STREAM TRIAD on Intel Ice Lake

## STREAM TRIAD

$a[i] = b[i] + s * c[i]$

..B2.42:

```
vmovups    (%r14,%rdx,8), %zmm1
vfmadd213pd (%r15,%rdx,8), %zmm2, %zmm1
vmovupd    %zmm1, (%r12,%rdx,8)
addq       $8, %rdx
cmpq       %rsi, %rdx
jb         ..B2.42
```



# STREAM TRIAD on Intel Ice Lake

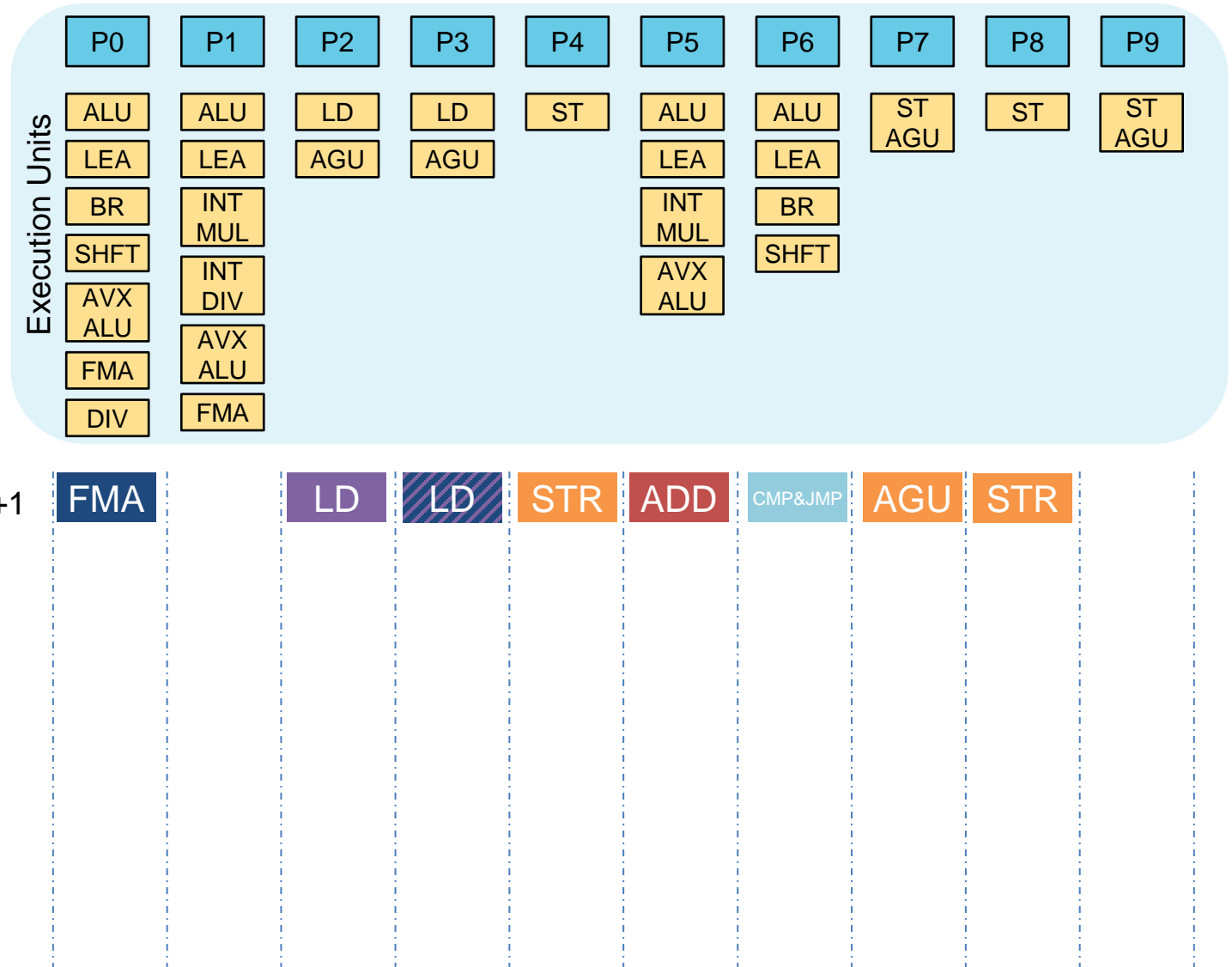
## STREAM TRIAD

$a[i] = b[i] + s * c[i]$

..B2.42:

```
vmovups    (%r14,%rdx,8), %zmm1
vfmadd213pd (%r15,%rdx,8), %zmm2, %zmm1
vmovupd    %zmm1, (%r12,%rdx,8)
addq       $8, %rdx
cmpq       %rsi, %rdx
jb         ..B2.42
```

It x+1



# STREAM TRIAD on Intel Ice Lake

## STREAM TRIAD

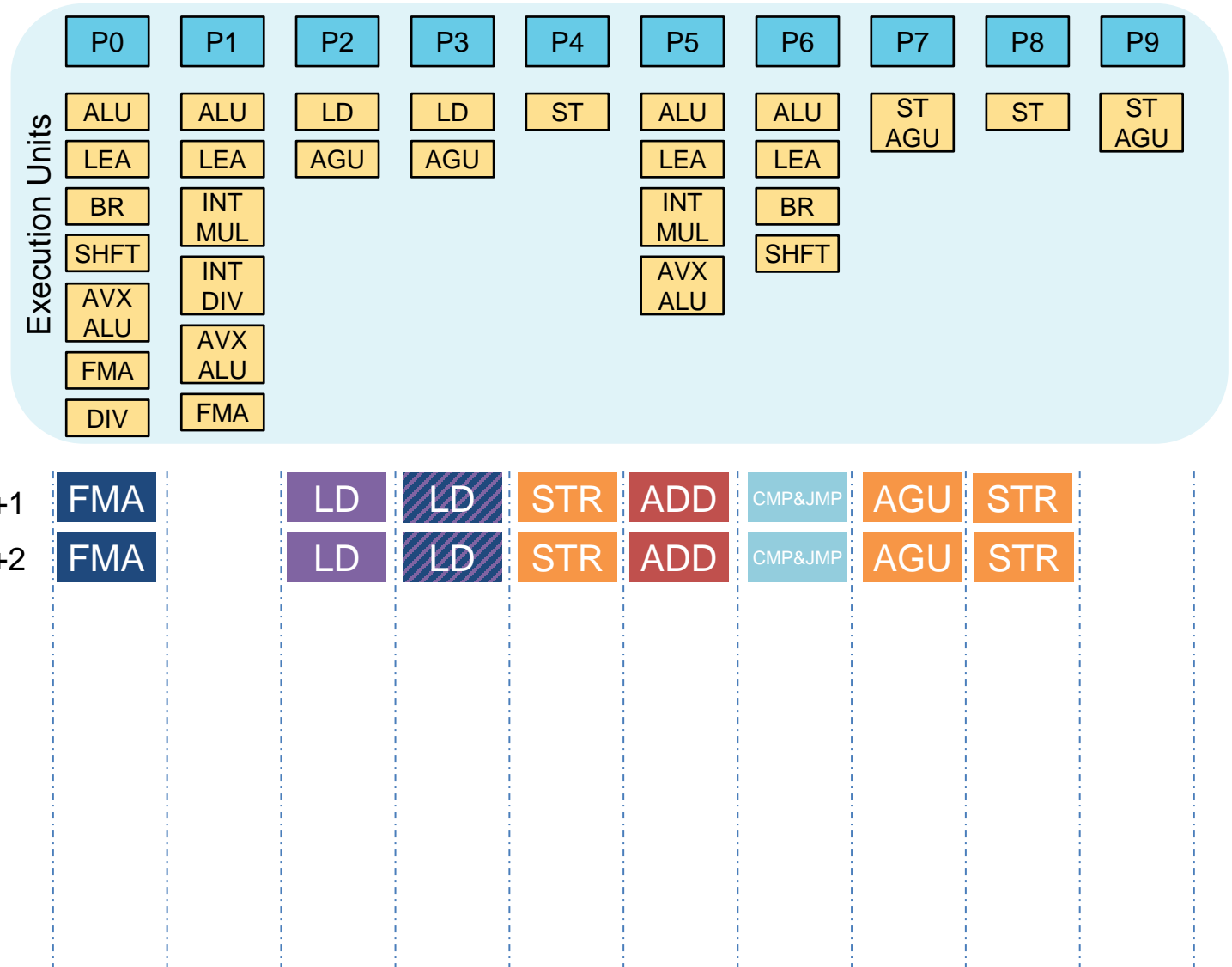
$$a[i] = b[i] + s * c[i]$$

..B2.42:

```
vmovups    (%r14,%rdx,8), %zmm1
vfmadd213pd (%r15,%rdx,8), %zmm2, %zmm1
vmovupd    %zmm1, (%r12,%rdx,8)
addq       $8, %rdx
cmpq       %rsi, %rdx
jb         ..B2.42
```

It x+1

It x+2



# STREAM TRIAD on Intel Ice Lake

## STREAM TRIAD

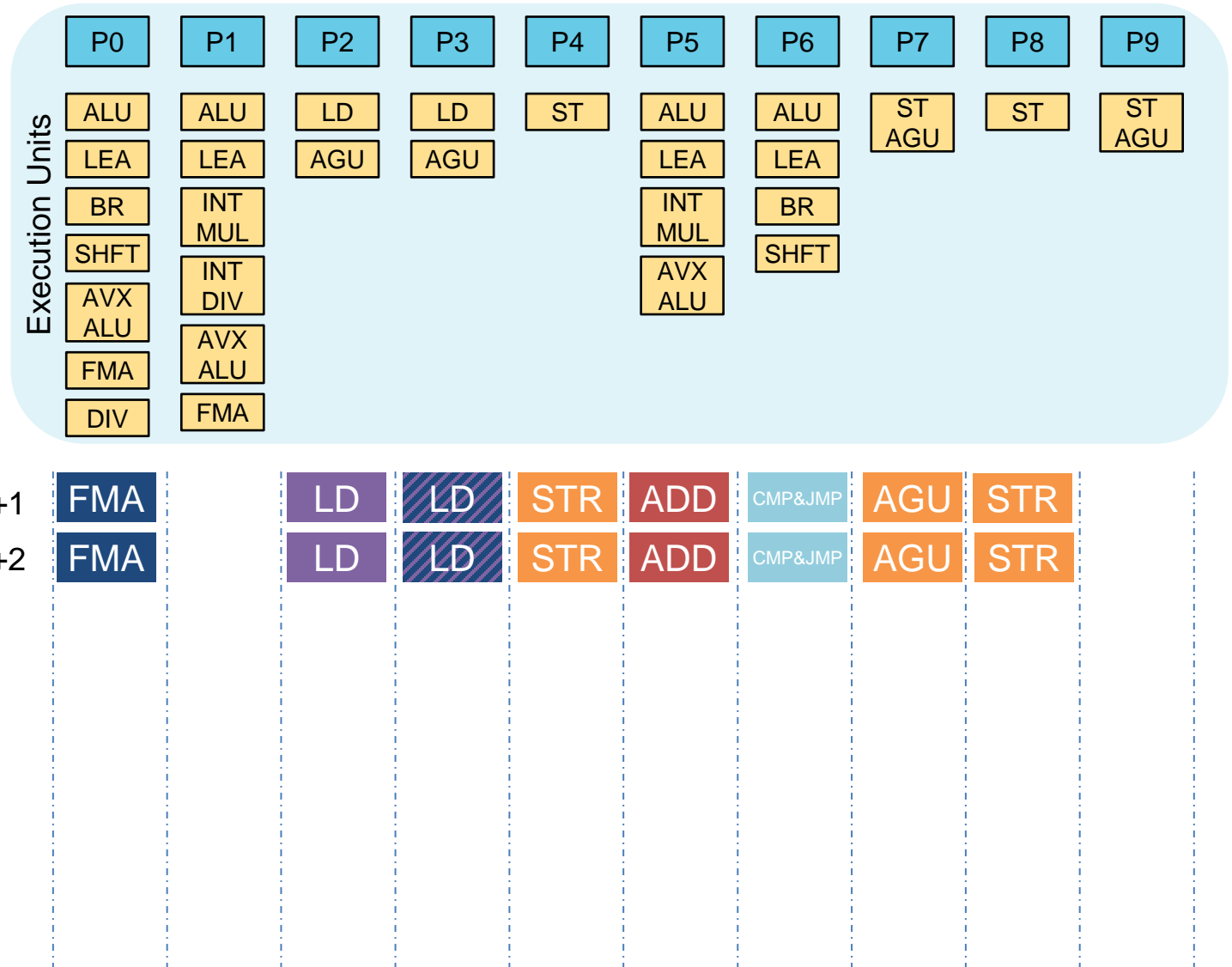
$$a[i] = b[i] + s * c[i]$$

..B2.42:

```
vmovups    (%r14,%rdx,8), %zmm1
vfmadd213pd (%r15,%rdx,8), %zmm2, %zmm1
vmovupd    %zmm1, (%r12,%rdx,8)
addq       $8, %rdx
cmpq       %rsi, %rdx
jb         ..B2.42
```

It x+1

It x+2





# STREAM TRIAD on Intel Ice Lake

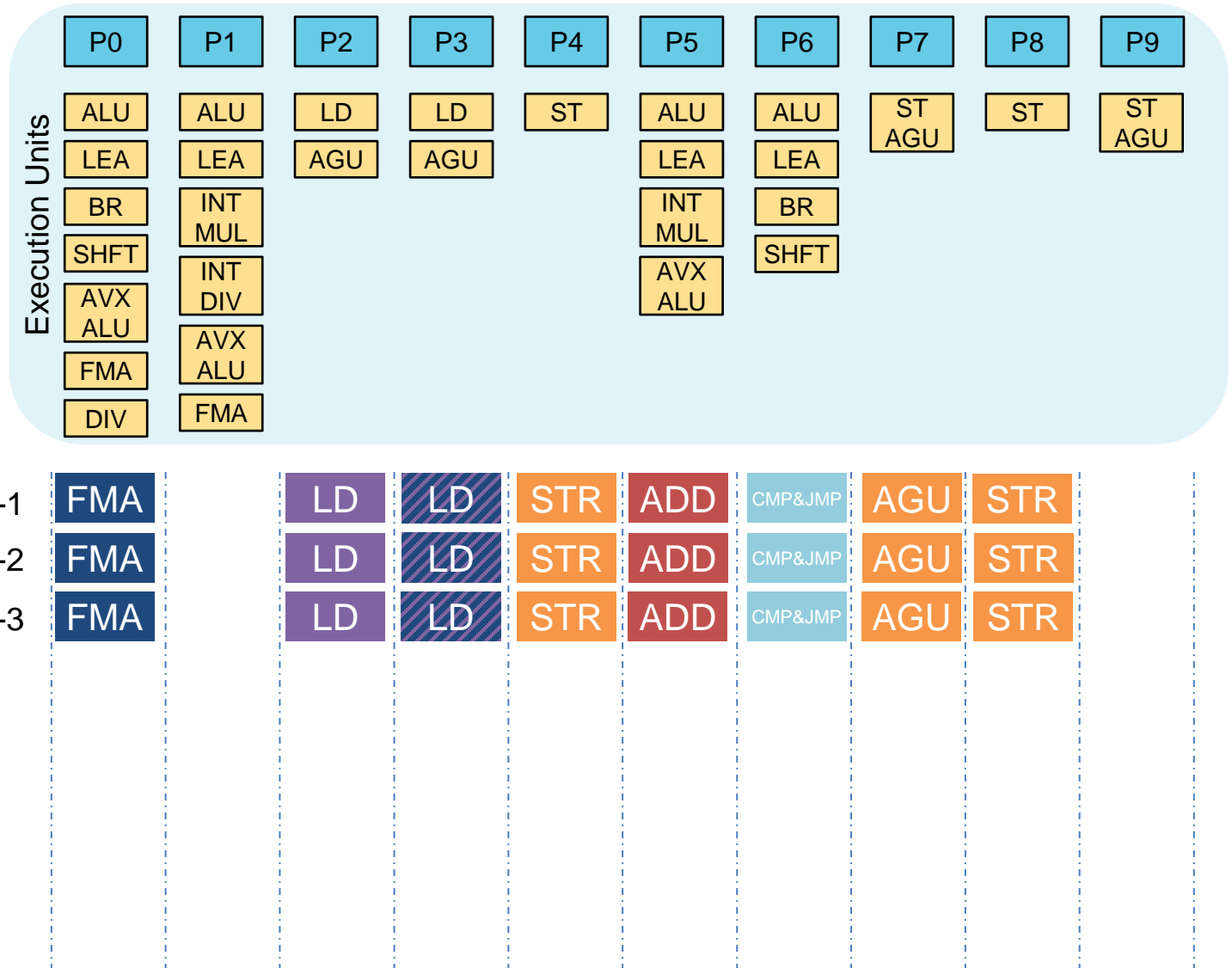
## STREAM TRIAD

$$a[i] = b[i] + s * c[i]$$

..B2.42:

```
vmovups    (%r14,%rdx,8), %zmm1
vfmadd213pd (%r15,%rdx,8), %zmm2, %zmm1
vmovupd    %zmm1, (%r12,%rdx,8)
addq       $8, %rdx
cmpq       %rsi, %rdx
jb         ..B2.42
```

It x+1  
It x+2  
It x+3



# STREAM TRIAD on Intel Ice Lake

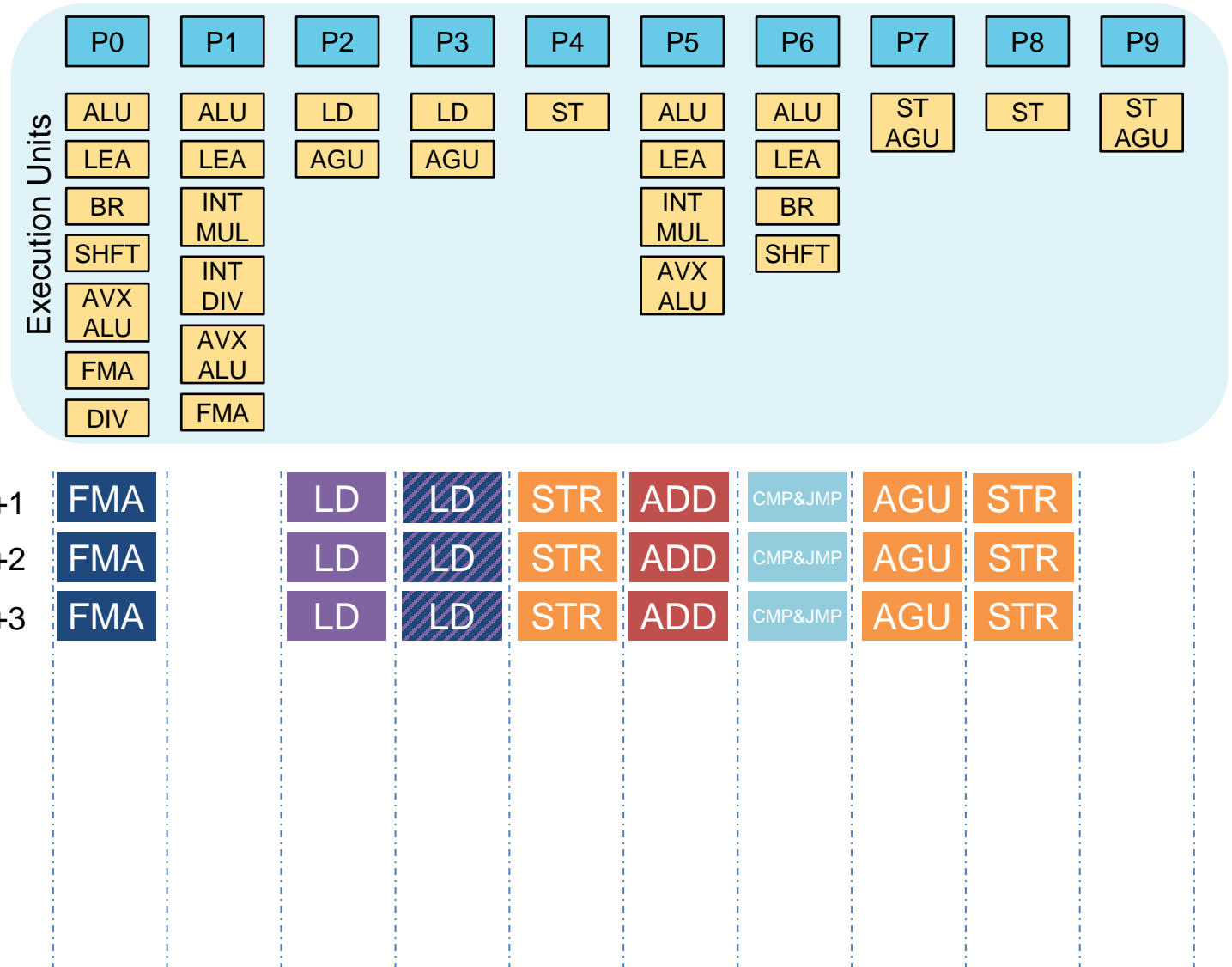
## STREAM TRIAD

$$a[i] = b[i] + s * c[i]$$

..B2.42:

```
vmovups    (%r14,%rdx,8), %zmm1
vfmadd213pd (%r15,%rdx,8), %zmm2, %zmm1
vmovupd    %zmm1, (%r12,%rdx,8)
addq       $8, %rdx
cmpq       %rsi, %rdx
jb         ..B2.42
```

It x+1  
It x+2  
It x+3



# STREAM TRIAD on Intel Ice Lake

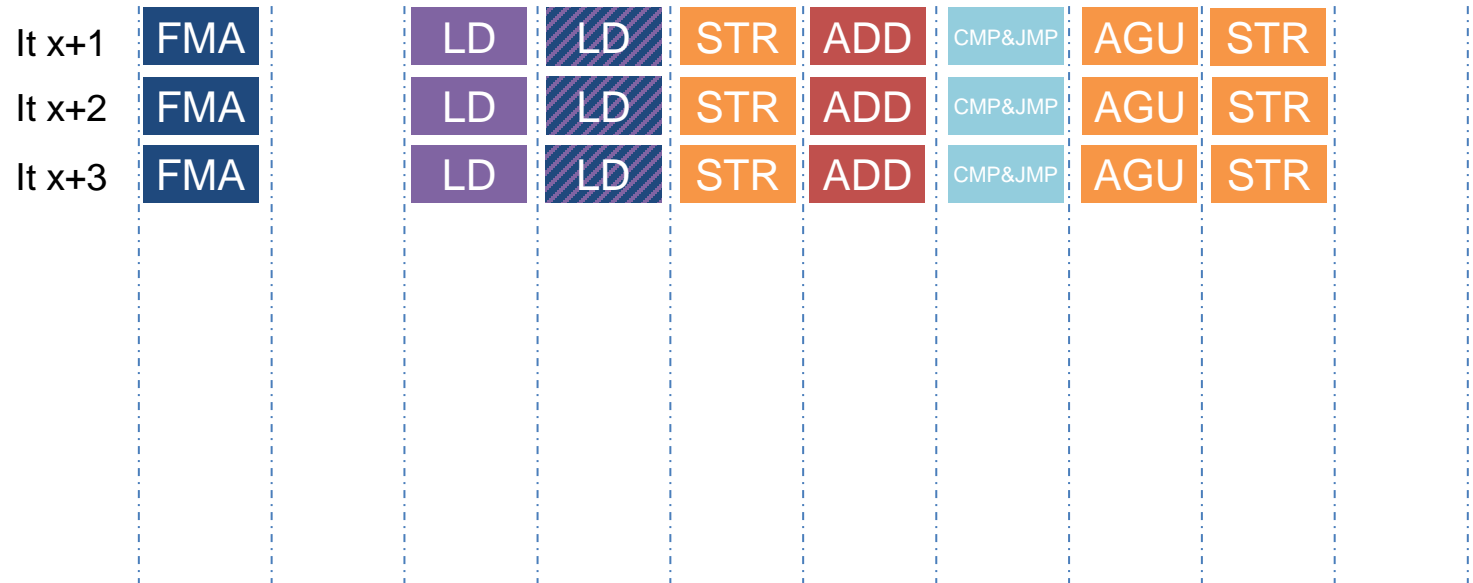
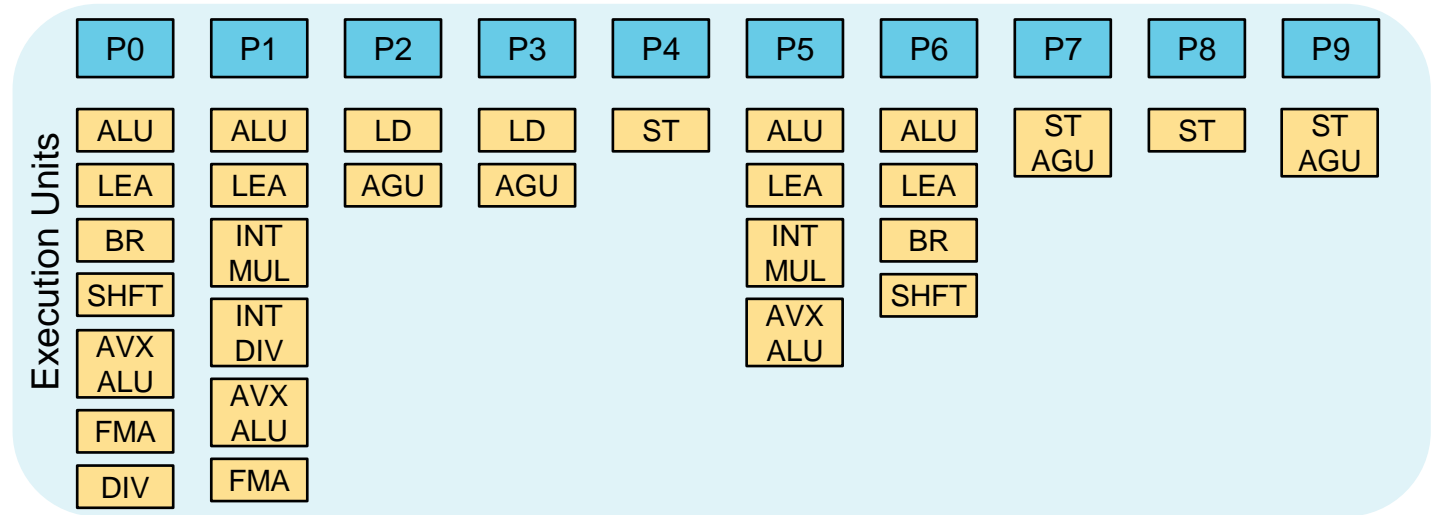
## STREAM TRIAD

$a[i] = b[i] + s * c[i]$

..B2.42:

```
vmovups    (%r14,%rdx,8), %zmm1
vfmadd213pd (%r15,%rdx,8), %zmm2, %zmm1
vmovupd    %zmm1, (%r12,%rdx,8)
addq       $8, %rdx
cmpq       %rsi, %rdx
jb         ..B2.42
```

1 cy / 8 it



# OSACA

- Open Source Architecture Code Analyzer
- Static in-core code analysis for x86 and ARM  $\mu$ -archs

## Assumptions

- Steady-state execution (no warm-up/cool-down)
- All data in L1
- Perfect out-of-order scheduling
- (currently) no front-end, i.e., no limit in instruction fetching, decoding, etc...
- Architecture specific model for each  $\mu$ -arch

- Python module

```
$ pip install osaca
```

# Workflow

## Marked assembly

x86

```
# OSACA-BEGIN
..B2.42:
    vmovups (%r14,%rdx,8),%zmm1
    vfmadd213pd (%r15,%rdx,8),%zmm2,%zmm1
    vmovupd %zmm1, (%r12,%rdx,8)
    addq $8,%rdx
    cmpq %rsi,%rdx
    jne ..B2.42
# OSACA-END
```

arm

```
// OSACA-BEGIN
.L18:
    ldr q2, [x20, x0]
    ldr q1, [x21, x0]
    fmla v1.2d, v2.2d, v0.2d
    str q1, [x19, x0]
    add x0, x0, #16
    cmp x22, x0
    bne .L18
// OSACA-END
```

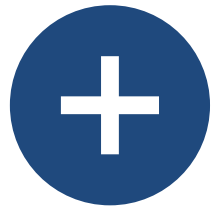
## Machine file / database

```
load_latency: {gpr: 5, xmm: 5, ymm: 5, zmm: 5}
load_throughput_default: [[1,"23"], [1,["2D","3D"]]]
store_throughput_default: [[1,"79"], [1,"48"]]

- # ...
- name: vfmadd213pd
  operands:
    - class: register
      name: zmm
      source: True
      destination: False
    - class: register
      name: zmm
      source: True
      destination: False
    - class: register
      name: zmm
      source: True
      destination: True
  throughput: 0.5
  latency: 4 # 1*p05
  port_pressure: [[1,"05"]]
- # ...
```

# Workflow – Creation of machine file

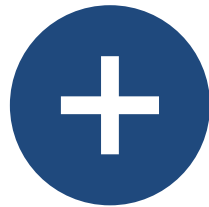
---



# Workflow – Creation of machine file

ibench

```
...  
loop:  
  inc      i  
  INSTR    zmm3, zmm0, zmm1  
  INSTR    zmm4, zmm1, zmm0  
  INSTR    zmm5, zmm0, zmm2  
  cmp      i, N  
  INSTR    zmm6, zmm2, zmm0  
  INSTR    zmm7, zmm1, zmm2  
  INSTR    zmm8, zmm2, zmm1  
  jl       loop  
...
```



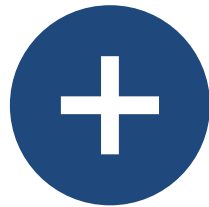
# Workflow – Creation of machine file

ibench

```
...  
loop:  
  inc      i  
  INSTR    zmm3, zmm0, zmm1  
  INSTR    zmm4, zmm1, zmm0  
  INSTR    zmm5, zmm0, zmm2  
  cmp      i, N  
  INSTR    zmm6, zmm2, zmm0  
  INSTR    zmm7, zmm1, zmm2  
  INSTR    zmm8, zmm2, zmm1  
  jl       loop  
...
```

asmbench

```
asmbench -f 2.4 -t 8 -p 6 "vfmadd213pd  
{src:<4 x double>:x}, {src:<4 x double>:x},  
{srcdst:<4 x double>:x}"
```





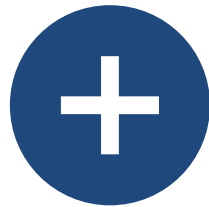
# Workflow – Creation of machine file

ibench

```
...  
loop:  
  inc      i  
  INSTR    zmm3, zmm0, zmm1  
  INSTR    zmm4, zmm1, zmm0  
  INSTR    zmm5, zmm0, zmm2  
  cmp      i, N  
  INSTR    zmm6, zmm2, zmm0  
  INSTR    zmm7, zmm1, zmm2  
  INSTR    zmm8, zmm2, zmm1  
  jl       loop  
...
```

asmbench

```
asmbench -f 2.4 -t 8 -p 6 "vfmadd213pd  
{src:<4 x double>:x}, {src:<4 x double>:x},  
{srcdst:<4 x double>:x}"
```



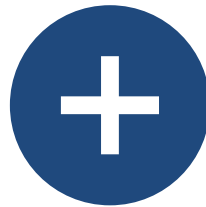
# Workflow – Creation of machine file

ibench

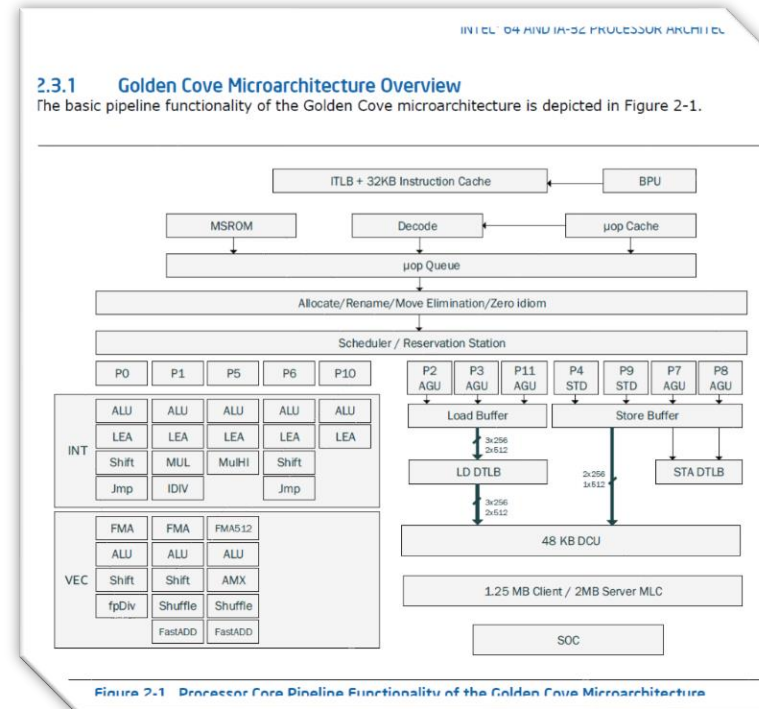
```
...  
loop:  
  inc      i  
  INSTR    zmm3, zmm0, zmm1  
  INSTR    zmm4, zmm1, zmm0  
  INSTR    zmm5, zmm0, zmm2  
  cmp      i, N  
  INSTR    zmm6, zmm2, zmm0  
  INSTR    zmm7, zmm1, zmm2  
  INSTR    zmm8, zmm2, zmm1  
  jl       loop  
...
```

asmbench

```
asmbench -f 2.4 -t 8 -p 6 "vfmadd213pd  
{src:<4 x double>:x}, {src:<4 x double>:x},  
{srcdst:<4 x double>:x}"
```



vendor documentation



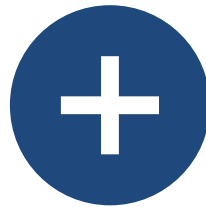
# Workflow – Creation of machine file

ibench

```
...
loop:
  inc      i
  INSTR    zmm3, zmm0, zmm1
  INSTR    zmm4, zmm1, zmm0
  INSTR    zmm5, zmm0, zmm2
  cmp      i, N
  INSTR    zmm6, zmm2, zmm0
  INSTR    zmm7, zmm1, zmm2
  INSTR    zmm8, zmm2, zmm1
  jl       loop
...
```

asmbench

```
asmbench -f 2.4 -t 8 -p 6 "vfmadd213pd
{src:<4 x double>:x}, {src:<4 x double>:x},
{srcdst:<4 x double>:x}"
```



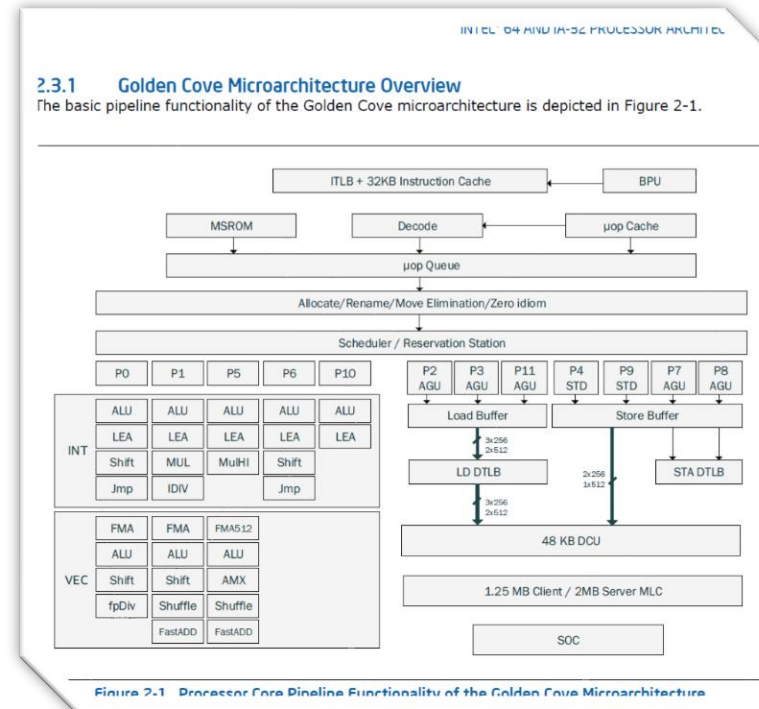
vendor documentation



uops.info

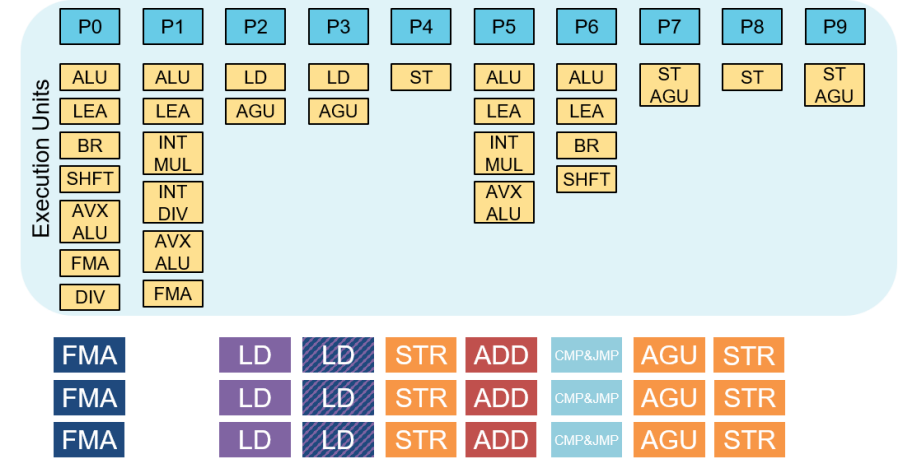
uops.info interface showing search results for the instruction `vfmadd` on the `Ice Lake` architecture.

Instruction	Lat	TP	Uops	Ports
<b>AVX512EVEX</b>				
<code>VFMADD132PD (XMM_K_XMM_M128)</code>	[4:≤11]	0.50 / 0.50	1 / 2	1'p01+1'p23
<code>VFMADD132PD (XMM_K_XMM_M64_1to2)</code>	[4:≤11]	0.50 / 0.50	1 / 2	1'p01+1'p23
<code>VFMADD132PD (XMM_K_XMM_XMM)</code>	4	0.50 / 0.50	1 / 1	1'p01
<code>VFMADD132PD (YMM_K_YMM_M256)</code>	[4:≤12]	0.50 / 0.50	1 / 2	1'p01+1'p23
<code>VFMADD132PD (YMM_K_YMM_M64_1to4)</code>	[4:≤12]	0.50 / 0.50	1 / 2	1'p01+1'p23
<code>VDD132PD (YMM_K_YMM_XMM)</code>	4	0.50 / 0.50	1 / 1	1'p01



# Triad on ICX with OSACA

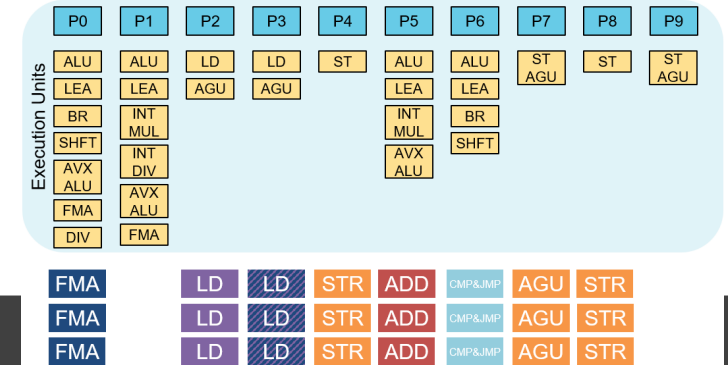
- Recap: Manual analysis resulted in 1 cy/8 it



- Recap: Manual analysis resulted in 1 cy/8 it

# Triad on ICX with OSACA

- Recap: Manual analysis resulted in 1 cy/8 it



```
$ osaca --arch ICX triad.s
```

```
Open Source Architecture Code Analyzer (OSACA) - 0.4.9
```

```
Architecture: ICX
```

## Throughput prediction

Port pressure in cycles													CP	LCD
0 - 0DV	1 - 1DV	2 - 2D	3 - 3D	4	5	6	7	8	9					
1														
2			0.50 0.50	0.50 0.50									5	
3	0.50		0.50 0.50	0.50 0.50		0.50							4	
4					1.00			0.50	1.00	0.50			0	
5	0.25	0.25				0.25	0.25							1
6	0.00	0.50				0.00	0.50							
7														
0.75 0.75 1.00 1.00 1.00 1.00 1.00 0.75 0.75 0.50 1.00 0.50													9	1

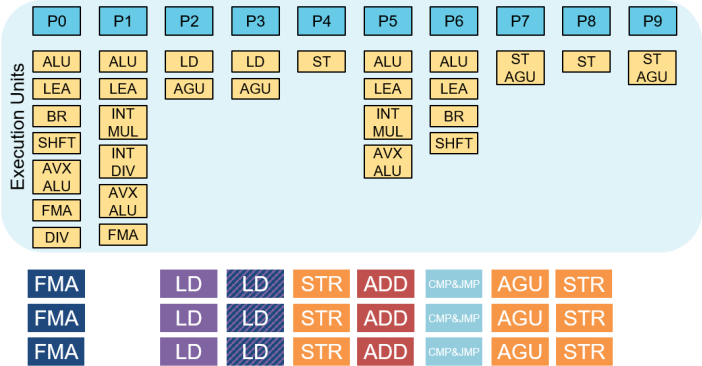
## Loop-Carried Dependencies Analysis Report

```
5 | 1.0 | addq $8, %rdx | [5]
```

```
..B2.42:
vmovups    (%r14,%rdx,8),%zmm1
vfmadd213pd (%r15,%rdx,8),%zmm2,%zmm1
vmovupd    %zmm1, (%r12,%rdx,8)
addq       $8, %rdx
cmpq       %rsi, %rdx
* jb       ..B2.42
```

# Triad on ICX with OSACA

- Recap: Manual analysis resulted in 1 cy/8 it



```
$ osaca --arch ICX triad.s
Open Source Architecture Code Analyzer (OSACA) - 0.4.9
Architecture: ICX
```

## Throughput prediction

	Port pressure in cycles											
	0 - 0DV	1 - 1DV	2 - 2D	3 - 3D	4	5	6	7	8	9		
1												
2			0.50 0.50	0.50 0.50								
3	0.50		0.50 0.50	0.50 0.50		0.50						
4					1.00			0.50	1.00	0.50		
5	0.25	0.25				0.25	0.25					
6	0.00	0.50				0.00	0.50					
7												
	0.75	0.75	1.00 1.00	1.00 1.00	1.00	0.75	0.75	0.50	1.00	0.50		

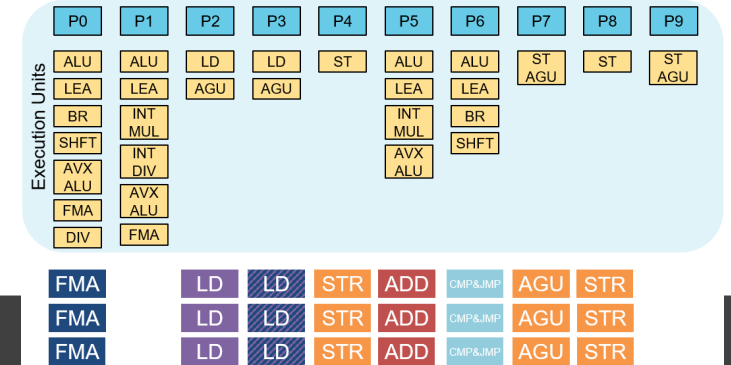
## Critical Path analysis

	CP	LCD	
1			..B2.42:
2	5		vmovups (%r14,%rdx,8),%zmm1
3	4		vfmadd213pd (%r15,%rdx,8),%zmm2,%zmm1
4	0		vmovupd %zmm1, (%r12,%rdx,8)
5		1	addq \$8, %rdx
6			cmpq %rsi, %rdx
7			* jb ..B2.42
	9	1	

```
Loop-Carried Dependencies Analysis Report
-----
5 | 1.0 | addq $8, %rdx | [5]
```

# Triad on ICX with OSACA

- Recap: Manual analysis resulted in 1 cy/8 it



```
$ osaca --arch ICX triad.s
```

```
Open Source Architecture Code Analyzer (OSACA) - 0.4.9
```

```
Architecture: ICX
```

## Throughput prediction

Port pressure in cycles												
	0	1	2	3	4	5	6	7	8	9		
1												
2			0.50	0.50	0.50	0.50						
3	0.50		0.50	0.50	0.50	0.50		0.50				
4					1.00			0.50	1.00	0.50		
5	0.25	0.25				0.25	0.25					
6	0.00	0.50				0.00	0.50					
7												
	0.75	0.75	1.00	1.00	1.00	1.00	0.75	0.75	0.50	1.00	0.50	

## Critical Path analysis

CP	LCD
5	
4	
0	1
9	1

```
..B2.42:
vmovups    (%r14,%rdx,8),%zmm1
vfmadd213pd (%r15,%rdx,8),%zmm2,%zmm1
vmovupd    %zmm1, (%r12,%rdx,8)
addq       $8, %rdx
cmpq       %rsi, %rdx
* jnb      ..B2.42
```

## Loop-Carried Dependencies Analysis Report

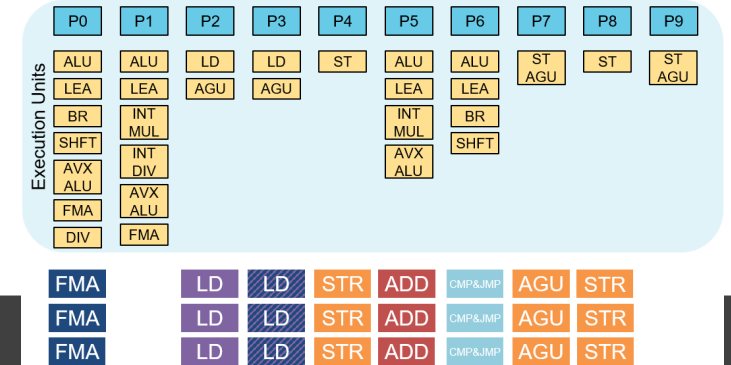
```
5 | 1.0 | addq    $8, %rdx | [5]
```

## Loop-carried dependency analysis



# Triad on ICX with OSACA

- Recap: Manual analysis resulted in 1 cy/8 it



```
$ osaca --arch ICX triad.s
```

```
Open Source Architecture Code Analyzer (OSACA) - 0.4.9
```

```
Architecture: ICX
```

Port pressure in cycles													
	0	1	2	3	4	5	6	7	8	9	CP	LCD	
1													..B2.42:
2			0.50	0.50	0.50	0.50					5		vmovups (%r14,%rdx,8),%zmm1
3	0.50		0.50	0.50	0.50	0.50		0.50			4		vfmadd213pd (%r15,%rdx,8),%zmm2,%zmm1
4					1.00			0.50	1.00	0.50	0		vmovupd %zmm1, (%r12,%rdx,8)
5	0.25	0.25				0.25	0.25					1	addq \$8, %rdx
6	0.00	0.50				0.00	0.50						cmpq %rsi, %rdx
7													* jnb ..B2.42
	0.75	0.75	1.00	1.00	1.00	1.00	1.00	0.75	0.75	0.50	1.00	0.50	9 1

```
Loop-Carried Dependencies Analysis Report
```

```
5 | 1.0 | addq $8, %rdx | [5]
```

# Useful gimmicks

- Marker insertion
- Dependency graph creation
- Compiler Explorer integration



# Marking the region of interest

```
$ osaca --arch icx --insert-marker
```

Blocks found in assembly file:

```
..B1.18
...
..B1.21
...
..B1.46
...
..B1.49
...
```

-----  
Possible blocks to be marked:

```
..B1.18
..B1.21
..B1.46
..B1.49
```

Choose block to be marked [**..B1.49**]: ☐

will be marked with **byte** markers, i.e.:

```
movl $111,%ebx; .byte 100,103,144; (x86)
...
movl $222,%ebx; .byte 100,103,144;
```

```
..B1.18: # Preds ..B1.18 ..B1.17
# Execution count [8.00e-01]: Infreq
movq %rcx, (%r14,%rbx,8) #34.5
movq %rax, (%r12,%rbx,8) #35.5
incq %rbx #33.3
cmpq %r9, %rbx #33.3
jb ..B1.18 # Prob 81% #33.3
```

```
..B1.21: # Preds ..B1.21 ..B1.20
# Execution count [8.00e-01]: Infreq
vmovupd %ymm1, (%r14,%r9,8) #34.5
vmovupd %ymm0, (%r12,%r9,8) #35.5
vmovupd %ymm1, 32(%r14,%r9,8) #34.5
vmovupd %ymm0, 32(%r12,%r9,8) #35.5
vmovupd %ymm1, 64(%r14,%r9,8) #34.5
vmovupd %ymm0, 64(%r12,%r9,8) #35.5
vmovupd %ymm1, 96(%r14,%r9,8) #34.5
vmovupd %ymm0, 96(%r12,%r9,8) #35.5
addq $16, %r9 #33.3
cmpq %rax, %r9 #33.3
jb ..B1.21 # Prob 81% #33.3
```

```
..B1.46: # Preds ..B1.44 ..B1.46
# Execution count [2.22e+01]: Infreq
vmovsd (%rbx,%rsi,8), %xmm3 #53.24
vfmadd231sd (%r14,%rsi,8), %xmm3,%xmm0 #53.9
incq %rsi #52.7
cmpq %rdi, %rsi #52.7
jb ..B1.46 # Prob 82% #52.7
```

```
..B1.49: # Preds ..B1.49 ..B1.48
# Execution count [2.22e+01]: Infreq
vmovupd (%rbx,%rdi,8), %ymm6 #53.24
vmovupd 32(%rbx,%rdi,8), %ymm7 #53.24
vmovupd 64(%rbx,%rdi,8), %ymm8 #53.24
vmovupd 96(%rbx,%rdi,8), %ymm9 #53.24
vfmadd231pd (%r14,%rdi,8), %ymm6, %ymm2 #53.24
vfmadd231pd 32(%r14,%rdi,8), %ymm7, %ymm5 #53.24
vfmadd231pd 64(%r14,%rdi,8), %ymm8, %ymm4 #53.24
vfmadd231pd 96(%r14,%rdi,8), %ymm9, %ymm3 #53.24
addq $16, %rdi #52.7
cmpq %rsi, %rdi #52.7
jb ..B1.49 # Prob 82% #52.7
```

Selects all inner-most loops and  
suggest the one with the most SIMD instructions

# Dependency graph creation

Port pressure in cycles																	
	0	- 0DV	1	- 1DV	2	- 2D	3	- 3D	4	5	6	7	8	9	CP	LCD	
1																..B2.42:	
2					0.50	0.50	0.50	0.50							5	vmovups (%r14,%rdx,8),%zmm1	
3	0.50				0.50	0.50	0.50	0.50		0.50					4	vfmadd213pd (%r15,%rdx,8),%zmm2,%zmm1	
4									1.00			0.50	1.00	0.50	0	vmovupd %zmm1, (%r12,%rdx,8)	
5	0.25		0.25							0.25	0.25					1	addq \$8, %rdx
6	0.00		0.50							0.00	0.50						cmpq %rsi, %rdx
7																*	jb ..B2.42
	0.75		0.75		1.00	1.00	1.00	1.00	1.00	0.75	0.75	0.50	1.00	0.50	9	1	

# Loop-Carried Dependencies Analysis Report

```
5 | 1.0 | addq    $8, %rdx | [5]
```

```
$ osaca --arch icx --export-graph dependencies.dot triad.s
$ dot -Tpdf dependencies.dot -o dep_graph.pdf
```

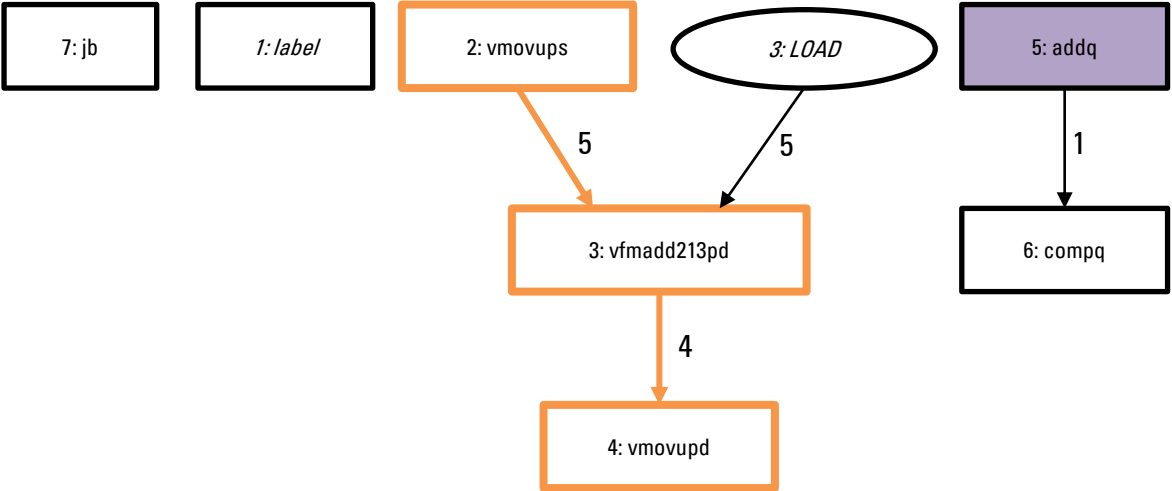
# Dependency graph creation

Port pressure in cycles													
	0 - 0DV	1 - 1DV	2 - 2D	3 - 3D	4	5	6	7	8	9	CP	LCD	
1													
2			0.50 0.50	0.50 0.50									5   vmovups (%r14,%rdx,8),%zmm1
3	0.50		0.50 0.50	0.50 0.50		0.50							4   vfmadd213pd (%r15,%rdx,8),%zmm2,%zmm1
4					1.00			0.50	1.00	0.50			0   vmovupd %zmm1, (%r12,%rdx,8)
5	0.25	0.25				0.25 0.25							1   addq \$8, %rdx
6	0.00	0.50				0.00 0.50							cmpq %rsi, %rdx
7													* jb ..B2.42
	0.75	0.75	1.00 1.00	1.00 1.00	1.00	0.75 0.75	0.50	1.00	0.50	9	1		

Loop-Carried Dependencies Analysis Report

5 | 1.0 | addq \$8, %rdx | [5]

```
$ osaca --arch icx --export-graph dependencies.dot triad.s
$ dot -Tpdf dependencies.dot -o dep_graph.pdf
```



# Compiler Explorer integration

The screenshot displays the Compiler Explorer web application interface. The top bar shows the 'COMPILER EXPLORER' logo and navigation links. The main area is divided into several panels:

- Source Code:** A C++ program for benchmarking a loop. It calculates the sum of squares of integers from 0 to N-1, where N is derived from a command-line argument. The program includes timing and printing logic.
- Compiler:** Set to 'x86-64 gcc 2021.7.1'.
- Options:** A list of compiler flags including `-Ofast -qopenmp-simd -qopt-zmm-usage=low -xICELAKE-SERVER -fargument-noalias -funroll-loops -fno-builtin -diag-disable=10441`.
- Assembly:** A list of assembly instructions generated by the compiler, such as `vcvtq2pd %xmm21, %ymm17` and `vfmadd231pd %ymm2, %ymm11, %ymm11`.
- Output:** A table showing the execution results of the program. The table has columns for instructions and their corresponding values. The output shows that the program executed successfully, with a total walltime of 0.000770 seconds.
- Execution Results:** A summary of the program's execution, including the number of elements (1000000), the calculated value of PI (3.141592653589793), and the total walltime (0.000770 seconds).

■ Try it yourself:

<https://godbolt.org/z/GveaKe3v1>



# Compiler Explorer integration

[illegible]

- Try it yourself: <https://godbolt.org/z/GveaKe3v1>

# Compiler Explorer integration

The screenshot displays the Compiler Explorer web application. The top-left pane shows the C++ source code for a benchmark program. The top-right pane displays the generated assembly code for the x86-64 target. The bottom-left pane shows the execution results, including the program's output and timing information. The bottom-right pane shows the Loop-Carried Dependencies Analysis Report.

```
int main(int argc, char** argv) {
    double wct_start, wct_end;
    int N;
    double size;
    double s = 0.0;
    double *a, *b;

    if (argc != 2) {
        printf("Usage: %s <elements>\n", argv[0]);
        exit(1);
    }
    N = atoi(argv[1]);
    double delta_x = 1./N;
    double x;

    // time measurement
    wct_start = getTimestamp();
    // =====
    // benchmark loop
    for(int i=0; i<N; ++i) {
        x = (i + 0.5) * delta_x;
        s += (4.0 / (1.0 + x * x));
    }
    // end of benchmark loop
    // =====
    wct_end = getTimestamp();
    if(s<0.) printf("%lf", s);

    double pi = s * delta_x;
    printf("PI = %.15f\n", pi);
    printf("%d elements\n", N);
    printf("Cycles per high-level iteration: Cannot be determined\n");
}
```

Assembly output (x86-64 icc 2021.7.1):

```
62 vcvtdq2pd %xmm21, %ymm17 #41.12
63 vcvtdq2pd %xmm26, %ymm22 #41.12
64 vaddpd %ymm9, %ymm4, %ymm10 #41.16
65 vaddpd %ymm6, %ymm4, %ymm13 #41.16
66 vpadd %xmm7, %xmm26, %xmm6 #41.5
67 vaddpd %ymm17, %ymm4, %ymm18 #41.16
68 vaddpd %ymm22, %ymm4, %ymm23 #41.16
69 vmulpd %ymm10, %ymm5, %ymm11 #41.23
70 vmulpd %ymm13, %ymm5, %ymm14 #41.23
71 vmulpd %ymm18, %ymm5, %ymm19 #41.23
72 vmulpd %ymm23, %ymm5, %ymm24 #41.23
73 vfmadd213pd %ymm2, %ymm11, %ymm11 #42.28
74 vfmadd213pd %ymm2, %ymm14, %ymm14 #42.28
75 vfmadd213pd %ymm2, %ymm19, %ymm19 #42.28
76 vfmadd213pd %ymm2, %ymm24, %ymm24 #42.28
77 vdivpd %ymm11, %ymm2, %ymm12 #42.28
78 vdivpd %ymm14, %ymm2, %ymm15 #42.28
```

Execution results (x86-64 icc 2021.7.1):

```
OSACA (0.5.0) x86-64 icc 2021.7.1 (Editor #1, Compiler #1)
--arch icx --lines 55-86

78 | 1.00 | 8.00 | | | | | | | | | | 13.0 | | | vdivpd %ymm14, %ymm2, %ymm15
79 | 1.00 | 8.00 | | | | | | | | | | | | | vdivpd %ymm19, %ymm2, %ymm20
80 | 1.00 | 8.00 | | | | | | | | | | | | | vdivpd %ymm24, %ymm2, %ymm25
81 | 0.00 | | 1.00 | | | | | | | | | | | vfmadd213pd %ymm3, %ymm12, %ymm0
82 | 0.00 | | 1.00 | | | | | | | | | | 4.0 | 4.0 | vfmadd213pd %ymm3, %ymm15, %ymm8
83 | 0.00 | | 1.00 | | | | | | | | | | 4.0 | 4.0 | vfmadd213pd %ymm3, %ymm20, %ymm0
84 | 0.00 | | 1.00 | | | | | | | | | | 4.0 | 4.0 | vfmadd213pd %ymm3, %ymm25, %ymm8
85 | 0.00 | | 0.00 | | | | | 0.00 | 1.00 | | | | | cmpl %edx, %eax
86 | | | | | | | | | | | | | | * jnb ..B1.7 # Prob 82%

12.0 32.0 12.0 8.00 2.00 41 8.0
```

Loop-Carried Dependencies Analysis Report

```
56 | 4.0 | vpadd %xmm7, %xmm6, %xmm16 #41.5| [56, 59, 61, 66]
57 | 1.0 | addl $16, %eax #40.3| [57]
81 | 8.0 | vfmadd213pd %ymm3, %ymm12, %ymm0 #42.5| [81, 83]
82 | 8.0 | vfmadd213pd %ymm3, %ymm15, %ymm8 #42.5| [82, 84]
```

Execution summary:

```
Program returned: 0
Program stdout
PI = 3.141592653589793
1000000 elements
Cycles per high-level iteration: Cannot be determined via godbolt.org
Total walltime: 0.000770
```

■ Try it yourself:

<https://godbolt.org/z/GveaKe3v1>





# Compiler Explorer integration

The screenshot displays the Compiler Explorer interface with the following components:

- Source Code (C++):**

```

19 int main(int argc, char** argv) {
20
21     double wct_start, wct_end;
22     int N;
23     double size;
24     double s = 0.0;
25     double *a, *b;
26
27     if (argc != 2) {
28         printf("Usage: %s <elements>\n", argv[0]);
29         exit(1);
30     }
31     N = atoi(argv[1]);
32     double delta_x = 1./N;
33     double x;
34
35     // time measurement
36     wct_start = getTimestamp();
37     // =====
38     // benchmark loop
39     for(int i=0; i<N; ++i) {
40         x = (i + 0.5) * delta_x;
41         s += (4.0 / (1.0 + x * x));
42     }
43     // end of benchmark loop
44     // =====
45     wct_end = getTimestamp();
46     if(s<0.) printf("%lf", s);
47
48     double pi = s * delta_x;
49     printf("PI = %.15f\n", pi);
50     printf("%d elements\n", N);
51     printf("Cycles per high-level iteration: Cannot be determined\n");
52 }

```
- Compiler:** x86-64 gcc 10.2.1 (Editor #1)
- Options:** -Ofast -qopenmp-simd -qopt-zmm-usage=low -xICELAKE-SERVER -fargument-noalias -funroll-loops -fno-builtin -diag-disable=10441
- Assembly Output (x86-64 gcc 10.2.1):**

```

62 vcvtdq2pd %xmm21, %ymm17          #41.12
63 vcvtdq2pd %xmm26, %ymm22          #41.12
64 vaddpd   %ymm9, %ymm4, %ymm10     #41.16
65 vaddpd   %ymm6, %ymm4, %ymm13     #41.16
66 vpadd    %xmm7, %xmm26, %xmm6     #41.5
67 vaddpd   %ymm17, %ymm4, %ymm18    #41.16
68 vaddpd   %ymm22, %ymm4, %ymm23    #41.16
69 vmulpd   %ymm10, %ymm5, %ymm11    #41.23
70 vmulpd   %ymm13, %ymm5, %ymm14    #41.23
71 vmulpd   %ymm18, %ymm5, %ymm19    #41.23
72 vmulpd   %ymm23, %ymm5, %ymm24    #41.23
73 vfmadd213pd %ymm2, %ymm11, %ymm11 #42.28
74 vfmadd213pd %ymm2, %ymm14, %ymm14 #42.28
75 vfmadd213pd %ymm2, %ymm19, %ymm19 #42.28
76 vfmadd213pd %ymm2, %ymm24, %ymm24 #42.28
77 vdivpd   %ymm11, %ymm2, %ymm12    #42.28
78 vdivpd   %ymm14, %ymm2, %ymm15    #42.28

```
- Execution Results:**

```

x86-64 gcc 10.2.1 (C++, Editor #1)
1000000
Program returned: 0
Program stdout
PI = 3.141592653589880
1000000 elements
Cycles per high-level iteration: Cannot be determined via godbolt.org
Total walltime: 0.000770

```
- Loop-Carried Dependencies Analysis Report:**

```

56 | 4.0 | vpadd    %xmm7, %xmm6, %xmm16    #41.5| [56, 59, 61, 66]
57 | 1.0 | addl     $16, %eax                 #40.3| [57]
81 | 8.0 | vfmadd213pd %ymm3, %ymm12, %ymm0    #42.5| [81, 83]
82 | 8.0 | vfmadd213pd %ymm3, %ymm15, %ymm8    #42.5| [82, 84]

```



- Try it yourself:

<https://godbolt.org/z/GveaKe3v1>

# Compiler Explorer integration

The screenshot displays the Compiler Explorer interface with the following components:

- Source Code (Left):** C++ code for a benchmark loop. A red box highlights the loop body, and a red arrow points to the assembly output.
- Assembly (Top Right):** x86-64 assembly code generated by ICC 2021.7.1. A red box highlights a section of the assembly, and a red arrow points to the source code.
- Output (Bottom Right):** Execution results for the benchmark loop, showing a table of values and a summary of the results.
- Compiler Options:** -Ofast -qopenmp-simd -qopt-zmm-usage=low -xICELAKE-SERVER -fargument-noalias -funroll-loops -fno-builtin -diag-disable=10441
- Execution Results:** Program returned: 0, Program stdout: PI = 3.141592653589793, 1000000 elements, Cycles per high-level iteration: Cannot be determined via godbolt.org, Total walltime: 0.000770.

Line	Op	Opnd1	Opnd2	Opnd3	Opnd4	Opnd5	Opnd6	Opnd7	Opnd8	Opnd9	Opnd10	Opnd11	Opnd12	Opnd13	Opnd14	Opnd15	Opnd16	Opnd17	Opnd18	Opnd19	Opnd20	Opnd21	Opnd22	Opnd23	Opnd24	Opnd25	Opnd26	Opnd27	Opnd28	Opnd29	Opnd30	Opnd31	Opnd32	Opnd33	Opnd34	Opnd35	Opnd36	Opnd37	Opnd38	Opnd39	Opnd40	Opnd41	Opnd42	Opnd43	Opnd44	Opnd45	Opnd46	Opnd47	Opnd48	Opnd49	Opnd50	Opnd51	Opnd52	Opnd53	Opnd54	Opnd55	Opnd56	Opnd57	Opnd58	Opnd59	Opnd60	Opnd61	Opnd62	Opnd63	Opnd64	Opnd65	Opnd66	Opnd67	Opnd68	Opnd69	Opnd70	Opnd71	Opnd72	Opnd73	Opnd74	Opnd75	Opnd76	Opnd77	Opnd78	Opnd79	Opnd80	Opnd81	Opnd82	Opnd83	Opnd84	Opnd85	Opnd86	Opnd87	Opnd88	Opnd89	Opnd90	Opnd91	Opnd92	Opnd93	Opnd94	Opnd95	Opnd96	Opnd97	Opnd98	Opnd99	Opnd100	Opnd101	Opnd102	Opnd103	Opnd104	Opnd105	Opnd106	Opnd107	Opnd108	Opnd109	Opnd110	Opnd111	Opnd112	Opnd113	Opnd114	Opnd115	Opnd116	Opnd117	Opnd118	Opnd119	Opnd120	Opnd121	Opnd122	Opnd123	Opnd124	Opnd125	Opnd126	Opnd127	Opnd128	Opnd129	Opnd130	Opnd131	Opnd132	Opnd133	Opnd134	Opnd135	Opnd136	Opnd137	Opnd138	Opnd139	Opnd140	Opnd141	Opnd142	Opnd143	Opnd144	Opnd145	Opnd146	Opnd147	Opnd148	Opnd149	Opnd150	Opnd151	Opnd152	Opnd153	Opnd154	Opnd155	Opnd156	Opnd157	Opnd158	Opnd159	Opnd160	Opnd161	Opnd162	Opnd163	Opnd164	Opnd165	Opnd166	Opnd167	Opnd168	Opnd169	Opnd170	Opnd171	Opnd172	Opnd173	Opnd174	Opnd175	Opnd176	Opnd177	Opnd178	Opnd179	Opnd180	Opnd181	Opnd182	Opnd183	Opnd184	Opnd185	Opnd186	Opnd187	Opnd188	Opnd189	Opnd190	Opnd191	Opnd192	Opnd193	Opnd194	Opnd195	Opnd196	Opnd197	Opnd198	Opnd199	Opnd200	Opnd201	Opnd202	Opnd203	Opnd204	Opnd205	Opnd206	Opnd207	Opnd208	Opnd209	Opnd210	Opnd211	Opnd212	Opnd213	Opnd214	Opnd215	Opnd216	Opnd217	Opnd218	Opnd219	Opnd220	Opnd221	Opnd222	Opnd223	Opnd224	Opnd225	Opnd226	Opnd227	Opnd228	Opnd229	Opnd230	Opnd231	Opnd232	Opnd233	Opnd234	Opnd235	Opnd236	Opnd237	Opnd238	Opnd239	Opnd240	Opnd241	Opnd242	Opnd243	Opnd244	Opnd245	Opnd246	Opnd247	Opnd248	Opnd249	Opnd250	Opnd251	Opnd252	Opnd253	Opnd254	Opnd255	Opnd256	Opnd257	Opnd258	Opnd259	Opnd260	Opnd261	Opnd262	Opnd263	Opnd264	Opnd265	Opnd266	Opnd267	Opnd268	Opnd269	Opnd270	Opnd271	Opnd272	Opnd273	Opnd274	Opnd275	Opnd276	Opnd277	Opnd278	Opnd279	Opnd280	Opnd281	Opnd282	Opnd283	Opnd284	Opnd285	Opnd286	Opnd287	Opnd288	Opnd289	Opnd290	Opnd291	Opnd292	Opnd293	Opnd294	Opnd295	Opnd296	Opnd297	Opnd298	Opnd299	Opnd300	Opnd301	Opnd302	Opnd303	Opnd304	Opnd305	Opnd306	Opnd307	Opnd308	Opnd309	Opnd310	Opnd311	Opnd312	Opnd313	Opnd314	Opnd315	Opnd316	Opnd317	Opnd318	Opnd319	Opnd320	Opnd321	Opnd322	Opnd323	Opnd324	Opnd325	Opnd326	Opnd327	Opnd328	Opnd329	Opnd330	Opnd331	Opnd332	Opnd333	Opnd334	Opnd335	Opnd336	Opnd337	Opnd338	Opnd339	Opnd340	Opnd341	Opnd342	Opnd343	Opnd344	Opnd345	Opnd346	Opnd347	Opnd348	Opnd349	Opnd350	Opnd351	Opnd352	Opnd353	Opnd354	Opnd355	Opnd356	Opnd357	Opnd358	Opnd359	Opnd360	Opnd361	Opnd362	Opnd363	Opnd364	Opnd365	Opnd366	Opnd367	Opnd368	Opnd369	Opnd370	Opnd371	Opnd372	Opnd373	Opnd374	Opnd375	Opnd376	Opnd377	Opnd378	Opnd379	Opnd380	Opnd381	Opnd382	Opnd383	Opnd384	Opnd385	Opnd386	Opnd387	Opnd388	Opnd389	Opnd390	Opnd391	Opnd392	Opnd393	Opnd394	Opnd395	Opnd396	Opnd397	Opnd398	Opnd399	Opnd400	Opnd401	Opnd402	Opnd403	Opnd404	Opnd405	Opnd406	Opnd407	Opnd408	Opnd409	Opnd410	Opnd411	Opnd412	Opnd413	Opnd414	Opnd415	Opnd416	Opnd417	Opnd418	Opnd419	Opnd420	Opnd421	Opnd422	Opnd423	Opnd424	Opnd425	Opnd426	Opnd427	Opnd428	Opnd429	Opnd430	Opnd431	Opnd432	Opnd433	Opnd434	Opnd435	Opnd436	Opnd437	Opnd438	Opnd439	Opnd440	Opnd441	Opnd442	Opnd443	Opnd444	Opnd445	Opnd446	Opnd447	Opnd448	Opnd449	Opnd450	Opnd451	Opnd452	Opnd453	Opnd454	Opnd455	Opnd456	Opnd457	Opnd458	Opnd459	Opnd460	Opnd461	Opnd462	Opnd463	Opnd464	Opnd465	Opnd466	Opnd467	Opnd468	Opnd469	Opnd470	Opnd471	Opnd472	Opnd473	Opnd474	Opnd475	Opnd476	Opnd477	Opnd478	Opnd479	Opnd480	Opnd481	Opnd482	Opnd483	Opnd484	Opnd485	Opnd486	Opnd487	Opnd488	Opnd489	Opnd490	Opnd491	Opnd492	Opnd493	Opnd494	Opnd495	Opnd496	Opnd497	Opnd498	Opnd499	Opnd500	Opnd501	Opnd502	Opnd503	Opnd504	Opnd505	Opnd506	Opnd507	Opnd508	Opnd509	Opnd510	Opnd511	Opnd512	Opnd513	Opnd514	Opnd515	Opnd516	Opnd517	Opnd518	Opnd519	Opnd520	Opnd521	Opnd522	Opnd523	Opnd524	Opnd525	Opnd526	Opnd527	Opnd528	Opnd529	Opnd530	Opnd531	Opnd532	Opnd533	Opnd534	Opnd535	Opnd536	Opnd537	Opnd538	Opnd539	Opnd540	Opnd541	Opnd542	Opnd543	Opnd544	Opnd545	Opnd546	Opnd547	Opnd548	Opnd549	Opnd550	Opnd551	Opnd552	Opnd553	Opnd554	Opnd555	Opnd556	Opnd557	Opnd558	Opnd559	Opnd560	Opnd561	Opnd562	Opnd563	Opnd564	Opnd565	Opnd566	Opnd567	Opnd568	Opnd569	Opnd570	Opnd571	Opnd572	Opnd573	Opnd574	Opnd575	Opnd576	Opnd577	Opnd578	Opnd579	Opnd580	Opnd581	Opnd582	Opnd583	Opnd584	Opnd585	Opnd586	Opnd587	Opnd588	Opnd589	Opnd590	Opnd591	Opnd592	Opnd593	Opnd594	Opnd595	Opnd596	Opnd597	Opnd598	Opnd599	Opnd600	Opnd601	Opnd602	Opnd603	Opnd604	Opnd605	Opnd606	Opnd607	Opnd608	Opnd609	Opnd610	Opnd611	Opnd612	Opnd613	Opnd614	Opnd615	Opnd616	Opnd617	Opnd618	Opnd619	Opnd620	Opnd621	Opnd622	Opnd623	Opnd624	Opnd625	Opnd626	Opnd627	Opnd628	Opnd629	Opnd630	Opnd631	Opnd632	Opnd633	Opnd634	Opnd635	Opnd636	Opnd637	Opnd638	Opnd639	Opnd640	Opnd641	Opnd642	Opnd643	Opnd644	Opnd645	Opnd646	Opnd647	Opnd648	Opnd649	Opnd650	Opnd651	Opnd652	Opnd653	Opnd654	Opnd655	Opnd656	Opnd657	Opnd658	Opnd659	Opnd660	Opnd661	Opnd662	Opnd663	Opnd664	Opnd665	Opnd666	Opnd667	Opnd668	Opnd669	Opnd670	Opnd671	Opnd672	Opnd673	Opnd674	Opnd675	Opnd676	Opnd677	Opnd678	Opnd679	Opnd680	Opnd681	Opnd682	Opnd683	Opnd684	Opnd685	Opnd686	Opnd687	Opnd688	Opnd689	Opnd690	Opnd691	Opnd692	Opnd693	Opnd694	Opnd695	Opnd696	Opnd697	Opnd698	Opnd699	Opnd700	Opnd701	Opnd702	Opnd703	Opnd704	Opnd705	Opnd706	Opnd707	Opnd708	Opnd709	Opnd710	Opnd711	Opnd712	Opnd713	Opnd714	Opnd715	Opnd716	Opnd717	Opnd718	Opnd719	Opnd720	Opnd721	Opnd722	Opnd723	Opnd724	Opnd725	Opnd726	Opnd727	Opnd728	Opnd729	Opnd730	Opnd731	Opnd732	Opnd733	Opnd734	Opnd735	Opnd736	Opnd737	Opnd738	Opnd739	Opnd740	Opnd741	Opnd742	Opnd743	Opnd744	Opnd745	Opnd746	Opnd747	Opnd748	Opnd749	Opnd750	Opnd751	Opnd752	Opnd753	Opnd754	Opnd755	Opnd756	Opnd757	Opnd758	Opnd759	Opnd760	Opnd761	Opnd762	Opnd763	Opnd764	Opnd765	Opnd766	Opnd767	Opnd768	Opnd769	Opnd770	Opnd771	Opnd772	Opnd773	Opnd774	Opnd775	Opnd776	Opnd777	Opnd778	Opnd779	Opnd780	Opnd781	Opnd782	Opnd783	Opnd784	Opnd785	Opnd786	Opnd787	Opnd788	Opnd789	Opnd790	Opnd791	Opnd792	Opnd793	Opnd794	Opnd795	Opnd796	Opnd797	Opnd798	Opnd799	Opnd800	Opnd801	Opnd802	Opnd803	Opnd804	Opnd805	Opnd806	Opnd807	Opnd808	Opnd809	Opnd810	Opnd811	Opnd812	Opnd813	Opnd814	Opnd815	Opnd816	Opnd817	Opnd818	Opnd819	Opnd820	Opnd821	Opnd822	Opnd823	Opnd824	Opnd825	Opnd826	Opnd827	Opnd828	Opnd829	Opnd830	Opnd831	Opnd832	Opnd833	Opnd834	Opnd835	Opnd836	Opnd837	Opnd838	Opnd839	Opnd840	Opnd841	Opnd842	Opnd843	Opnd844	Opnd845	Opnd846	Opnd847	Opnd848	Opnd849	Opnd850	Opnd851	Opnd852	Opnd853	Opnd854	Opnd855	Opnd856	Opnd857	Opnd858	Opnd859	Opnd860	Opnd861	Opnd862	Opnd863	Opnd864	Opnd865	Opnd866	Opnd867	Opnd868	Opnd869	Opnd870	Opnd871	Opnd872	Opnd873	Opnd874	Opnd875	Opnd876	Opnd877	Opnd878	Opnd879	Opnd880	Opnd881	Opnd882	Opnd883	Opnd884	Opnd885	Opnd886	Opnd887	Opnd888	Opnd889	Opnd890	Opnd891	Opnd892	Opnd893	Opnd894	Opnd895	Opnd896	Opnd897	Opnd898	Opnd899	Opnd900	Opnd901	Opnd902	Opnd903	Opnd904	Opnd905	Opnd906	Opnd907	Opnd908	Opnd909	Opnd910	Opnd911	Opnd912	Opnd913	Opnd914	Opnd915	Opnd916	Opnd917	Opnd918	Opnd919	Opnd920	Opnd921	Opnd922	Opnd923	Opnd924	Opnd925	Opnd926	Opnd927	Opnd928	Opnd929	Opnd930	Opnd931	Opnd932	Opnd933	Opnd934	Opnd935	Opnd936	Opnd937	Opnd938	Opnd939	Opnd940	Opnd941	Opnd942	Opnd943	Opnd944	Opnd945	Opnd946	Opnd947	Opnd948	Opnd949	Opnd950	Opnd951	Opnd952	Opnd953	Opnd954	Opnd955	Opnd956	Opnd957	Opnd958	Opnd959	Opnd960	Opnd961	Opnd962	Opnd963	Opnd964	Opnd965	Opnd966	Opnd967	Opnd968	Opnd969	Opnd970	Opnd971	Opnd972	Opnd973	Opnd974	Opnd975	Opnd976	Opnd977	Opnd978	Opnd979	Opnd980	Opnd981	Opnd982	Opnd983	Opnd984	Opnd985	Opnd986	Opnd987	Opnd988	Opnd989	Opnd990	Opnd991	Opnd992	Opnd993	Opnd994	Opnd995	Opnd996	Opnd997	Opnd998	Opnd999	Opnd1000	Opnd1001	Opnd1002	Opnd1003	Opnd1004	Opnd1005	Opnd1006	Opnd1007	Opnd1008	Opnd1009	Opnd1010	Opnd1011	Opnd1012	Opnd1013	Opnd1014	Opnd1015	Opnd1016	Opnd1017	Opnd1018	Opnd1019	Opnd1020	Opnd1021	Opnd1022	Opnd1023	Opnd1024	Opnd1025	Opnd1026	Opnd1027	Opnd1028	Opnd1029	Opnd1030	Opnd1031	Opnd1032	Opnd1033	Opnd1034	Opnd1035	Opnd1036	Opnd1037	Opnd1038	Opnd1039	Opnd1040	Opnd1041	Opnd1042	Opnd1043	Opnd1044	Opnd1045	Opnd1046	Opnd1047	Opnd1048	Opnd1049	Opnd1050	Opnd1051	Opnd1052	Opnd1053	Opnd1054	Opnd1055	Opnd1056	Opnd1057	Opnd1058	Opnd1059	Opnd1060	Opnd1061	Opnd1062	Opnd1063	Opnd1064	Opnd1065	Opnd1066	Opnd1067	Opnd1068	Opnd1069	Opnd1070	Opnd1071	Opnd1072	Opnd1073	Opnd1074	Opnd1075	Opnd1076	Opnd1077	Opnd1078	Opnd1079	Opnd1080	Opnd1081	Opnd1082	Opnd1083	Opnd1084	Opnd1085	Opnd1086	Opnd1087	Opnd1088	Opnd1089	Opnd1090	Opnd1091	Opnd1092	Opnd1093	Opnd1094	Opnd1095	Opnd1096	Opnd1097	Opnd1098	Opnd1099	Opnd1100	Opnd1101	Opnd1102	Opnd1103	Opnd1104	Opnd1105	Opnd1106	Opnd1107	Opnd1108	Opnd1109	Opnd1110	Opnd1111	Opnd1112	Opnd1113	Opnd1114	Opnd1115	Opnd1116	Opnd1117	Opnd1118	Opnd1119	Opnd1120	Opnd1121	Opnd1122	Opnd1123	Opnd1124	Opnd1125	Opnd1126	Opnd1127	Opnd1128	Opnd1129	Opnd1130	Opnd1131	Opnd1132	Opnd1133	Opnd1134	Opnd1135	Opnd1136	Opnd1137	Opnd1138	Opnd1139	Opnd1140	Opnd1141	Opnd1142	Opnd1143	Opnd1144	Opnd1145	Opnd1146	Opnd1147	Opnd1148	Opnd1149	Opnd1150	Opnd1151	Opnd1152	Opnd1153	Opnd1154	Opnd1155	Opnd1156	Opnd1157	Opnd1158	Opnd1159	Opnd1160	Opnd1161	Opnd1162	Opnd1163	Opnd1164	Opnd1165	Opnd1166	Opnd1167	Opnd1168	Opnd1169	Opnd1170	Opnd1171	Opnd1172	Opnd1173	Opnd1174	Opnd1175	Opnd1176	Opnd1177	Opnd1178	Opnd1179	Opnd1180	Opnd1181	Opnd1182	Opnd1183	Opnd1184	Opnd1185	Opnd1186	Opnd1187	Opnd1188	Opnd1189	Opnd1190	Opnd1
------	----	-------	-------	-------	-------	-------	-------	-------	-------	-------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	-------

# Compiler Explorer integration

The screenshot displays the Compiler Explorer web interface. On the left, the C++ source code is shown, with a red box highlighting a benchmark loop. The main panel shows the generated assembly code for x86-64 using GCC 10.2.1, with a red box highlighting a section of the assembly. The bottom panel shows the analysis reports, including a table of instructions and a Loop-Carried Dependencies Analysis Report.

**C++ Source Code (Highlighted Loop):**

```
28 if (argc != 2) {
29     printf("Usage: %s <elements>\n", argv[0]);
30     exit(1);
31 }
32 N = atof(argv[1]);
33 double delta_x = 1./N;
34 double x;
35 // time measurement
36 wct_start = getTimestamp();
37 // =====
38 // benchmark loop
39 for(int i=0; i<N; ++i) {
40     x = (i + 0.5) * delta_x;
41     s += (4.0 / (1.0 + x * x));
42 }
43 // end of benchmark loop
44 // =====
45 wct_end = getTimestamp();
46 if(s<0.) printf("%lf", s);
47
48 double pi = s * delta_x;
49 printf("PI = %.15f\n", pi);
50 printf("%d elements\n", N);
51 printf("Cycles per high-level iteration: Cannot be determined\n");
52
```

**Assembly Output (Highlighted Section):**

```
62 vcvtdq2pd %xmm21, %ymm17 #41.12
63 vcvtdq2pd %xmm26, %ymm22 #41.12
64 vaddpd %ymm9, %ymm4, %ymm10 #41.16
65 vaddpd %ymm6, %ymm4, %ymm13 #41.16
66 vpadd %xmm7, %xmm26, %xmm6 #41.5
67 vaddpd %ymm17, %ymm4, %ymm18 #41.16
68 vaddpd %ymm22, %ymm4, %ymm23 #41.16
69 vmulpd %ymm10, %ymm5, %ymm11 #41.23
70 vmulpd %ymm13, %ymm5, %ymm14 #41.23
71 vmulpd %ymm18, %ymm5, %ymm19 #41.23
72 vmulpd %ymm23, %ymm5, %ymm24 #41.23
73 vfmadd213pd %ymm2, %ymm11, %ymm11 #42.28
74 vfmadd213pd %ymm2, %ymm14, %ymm14 #42.28
75 vfmadd213pd %ymm2, %ymm19, %ymm19 #42.28
76 vfmadd213pd %ymm2, %ymm24, %ymm24 #42.28
77 vdivpd %ymm11, %ymm2, %ymm12 #42.28
78 vdivnd %ymm14, %ymm2, %ymm15 #42.28
```

**Analysis Reports (Highlighted Table):**

Line	Op	Opnd1	Opnd2	Opnd3	Opnd4	Opnd5	Opnd6	Opnd7	Opnd8	Opnd9	Opnd10	Opnd11	Opnd12	Opnd13	Opnd14	Opnd15	Opnd16	Opnd17	Opnd18	Opnd19	Opnd20	Opnd21	Opnd22	Opnd23	Opnd24	Opnd25	Opnd26	Opnd27	Opnd28	Opnd29	Opnd30	Opnd31	Opnd32	Opnd33	Opnd34	Opnd35	Opnd36	Opnd37	Opnd38	Opnd39	Opnd40	Opnd41	Opnd42	Opnd43	Opnd44	Opnd45	Opnd46	Opnd47	Opnd48	Opnd49	Opnd50	Opnd51	Opnd52	Opnd53	Opnd54	Opnd55	Opnd56	Opnd57	Opnd58	Opnd59	Opnd60	Opnd61	Opnd62	Opnd63	Opnd64	Opnd65	Opnd66	Opnd67	Opnd68	Opnd69	Opnd70	Opnd71	Opnd72	Opnd73	Opnd74	Opnd75	Opnd76	Opnd77	Opnd78	Opnd79	Opnd80	Opnd81	Opnd82	Opnd83	Opnd84	Opnd85	Opnd86	Opnd87	Opnd88	Opnd89	Opnd90	Opnd91	Opnd92	Opnd93	Opnd94	Opnd95	Opnd96	Opnd97	Opnd98	Opnd99	Opnd100	Opnd101	Opnd102	Opnd103	Opnd104	Opnd105	Opnd106	Opnd107	Opnd108	Opnd109	Opnd110	Opnd111	Opnd112	Opnd113	Opnd114	Opnd115	Opnd116	Opnd117	Opnd118	Opnd119	Opnd120	Opnd121	Opnd122	Opnd123	Opnd124	Opnd125	Opnd126	Opnd127	Opnd128	Opnd129	Opnd130	Opnd131	Opnd132	Opnd133	Opnd134	Opnd135	Opnd136	Opnd137	Opnd138	Opnd139	Opnd140	Opnd141	Opnd142	Opnd143	Opnd144	Opnd145	Opnd146	Opnd147	Opnd148	Opnd149	Opnd150	Opnd151	Opnd152	Opnd153	Opnd154	Opnd155	Opnd156	Opnd157	Opnd158	Opnd159	Opnd160	Opnd161	Opnd162	Opnd163	Opnd164	Opnd165	Opnd166	Opnd167	Opnd168	Opnd169	Opnd170	Opnd171	Opnd172	Opnd173	Opnd174	Opnd175	Opnd176	Opnd177	Opnd178	Opnd179	Opnd180	Opnd181	Opnd182	Opnd183	Opnd184	Opnd185	Opnd186	Opnd187	Opnd188	Opnd189	Opnd190	Opnd191	Opnd192	Opnd193	Opnd194	Opnd195	Opnd196	Opnd197	Opnd198	Opnd199	Opnd200	Opnd201	Opnd202	Opnd203	Opnd204	Opnd205	Opnd206	Opnd207	Opnd208	Opnd209	Opnd210	Opnd211	Opnd212	Opnd213	Opnd214	Opnd215	Opnd216	Opnd217	Opnd218	Opnd219	Opnd220	Opnd221	Opnd222	Opnd223	Opnd224	Opnd225	Opnd226	Opnd227	Opnd228	Opnd229	Opnd230	Opnd231	Opnd232	Opnd233	Opnd234	Opnd235	Opnd236	Opnd237	Opnd238	Opnd239	Opnd240	Opnd241	Opnd242	Opnd243	Opnd244	Opnd245	Opnd246	Opnd247	Opnd248	Opnd249	Opnd250	Opnd251	Opnd252	Opnd253	Opnd254	Opnd255	Opnd256	Opnd257	Opnd258	Opnd259	Opnd260	Opnd261	Opnd262	Opnd263	Opnd264	Opnd265	Opnd266	Opnd267	Opnd268	Opnd269	Opnd270	Opnd271	Opnd272	Opnd273	Opnd274	Opnd275	Opnd276	Opnd277	Opnd278	Opnd279	Opnd280	Opnd281	Opnd282	Opnd283	Opnd284	Opnd285	Opnd286	Opnd287	Opnd288	Opnd289	Opnd290	Opnd291	Opnd292	Opnd293	Opnd294	Opnd295	Opnd296	Opnd297	Opnd298	Opnd299	Opnd300	Opnd301	Opnd302	Opnd303	Opnd304	Opnd305	Opnd306	Opnd307	Opnd308	Opnd309	Opnd310	Opnd311	Opnd312	Opnd313	Opnd314	Opnd315	Opnd316	Opnd317	Opnd318	Opnd319	Opnd320	Opnd321	Opnd322	Opnd323	Opnd324	Opnd325	Opnd326	Opnd327	Opnd328	Opnd329	Opnd330	Opnd331	Opnd332	Opnd333	Opnd334	Opnd335	Opnd336	Opnd337	Opnd338	Opnd339	Opnd340	Opnd341	Opnd342	Opnd343	Opnd344	Opnd345	Opnd346	Opnd347	Opnd348	Opnd349	Opnd350	Opnd351	Opnd352	Opnd353	Opnd354	Opnd355	Opnd356	Opnd357	Opnd358	Opnd359	Opnd360	Opnd361	Opnd362	Opnd363	Opnd364	Opnd365	Opnd366	Opnd367	Opnd368	Opnd369	Opnd370	Opnd371	Opnd372	Opnd373	Opnd374	Opnd375	Opnd376	Opnd377	Opnd378	Opnd379	Opnd380	Opnd381	Opnd382	Opnd383	Opnd384	Opnd385	Opnd386	Opnd387	Opnd388	Opnd389	Opnd390	Opnd391	Opnd392	Opnd393	Opnd394	Opnd395	Opnd396	Opnd397	Opnd398	Opnd399	Opnd400	Opnd401	Opnd402	Opnd403	Opnd404	Opnd405	Opnd406	Opnd407	Opnd408	Opnd409	Opnd410	Opnd411	Opnd412	Opnd413	Opnd414	Opnd415	Opnd416	Opnd417	Opnd418	Opnd419	Opnd420	Opnd421	Opnd422	Opnd423	Opnd424	Opnd425	Opnd426	Opnd427	Opnd428	Opnd429	Opnd430	Opnd431	Opnd432	Opnd433	Opnd434	Opnd435	Opnd436	Opnd437	Opnd438	Opnd439	Opnd440	Opnd441	Opnd442	Opnd443	Opnd444	Opnd445	Opnd446	Opnd447	Opnd448	Opnd449	Opnd450	Opnd451	Opnd452	Opnd453	Opnd454	Opnd455	Opnd456	Opnd457	Opnd458	Opnd459	Opnd460	Opnd461	Opnd462	Opnd463	Opnd464	Opnd465	Opnd466	Opnd467	Opnd468	Opnd469	Opnd470	Opnd471	Opnd472	Opnd473	Opnd474	Opnd475	Opnd476	Opnd477	Opnd478	Opnd479	Opnd480	Opnd481	Opnd482	Opnd483	Opnd484	Opnd485	Opnd486	Opnd487	Opnd488	Opnd489	Opnd490	Opnd491	Opnd492	Opnd493	Opnd494	Opnd495	Opnd496	Opnd497	Opnd498	Opnd499	Opnd500	Opnd501	Opnd502	Opnd503	Opnd504	Opnd505	Opnd506	Opnd507	Opnd508	Opnd509	Opnd510	Opnd511	Opnd512	Opnd513	Opnd514	Opnd515	Opnd516	Opnd517	Opnd518	Opnd519	Opnd520	Opnd521	Opnd522	Opnd523	Opnd524	Opnd525	Opnd526	Opnd527	Opnd528	Opnd529	Opnd530	Opnd531	Opnd532	Opnd533	Opnd534	Opnd535	Opnd536	Opnd537	Opnd538	Opnd539	Opnd540	Opnd541	Opnd542	Opnd543	Opnd544	Opnd545	Opnd546	Opnd547	Opnd548	Opnd549	Opnd550	Opnd551	Opnd552	Opnd553	Opnd554	Opnd555	Opnd556	Opnd557	Opnd558	Opnd559	Opnd560	Opnd561	Opnd562	Opnd563	Opnd564	Opnd565	Opnd566	Opnd567	Opnd568	Opnd569	Opnd570	Opnd571	Opnd572	Opnd573	Opnd574	Opnd575	Opnd576	Opnd577	Opnd578	Opnd579	Opnd580	Opnd581	Opnd582	Opnd583	Opnd584	Opnd585	Opnd586	Opnd587	Opnd588	Opnd589	Opnd590	Opnd591	Opnd592	Opnd593	Opnd594	Opnd595	Opnd596	Opnd597	Opnd598	Opnd599	Opnd600	Opnd601	Opnd602	Opnd603	Opnd604	Opnd605	Opnd606	Opnd607	Opnd608	Opnd609	Opnd610	Opnd611	Opnd612	Opnd613	Opnd614	Opnd615	Opnd616	Opnd617	Opnd618	Opnd619	Opnd620	Opnd621	Opnd622	Opnd623	Opnd624	Opnd625	Opnd626	Opnd627	Opnd628	Opnd629	Opnd630	Opnd631	Opnd632	Opnd633	Opnd634	Opnd635	Opnd636	Opnd637	Opnd638	Opnd639	Opnd640	Opnd641	Opnd642	Opnd643	Opnd644	Opnd645	Opnd646	Opnd647	Opnd648	Opnd649	Opnd650	Opnd651	Opnd652	Opnd653	Opnd654	Opnd655	Opnd656	Opnd657	Opnd658	Opnd659	Opnd660	Opnd661	Opnd662	Opnd663	Opnd664	Opnd665	Opnd666	Opnd667	Opnd668	Opnd669	Opnd670	Opnd671	Opnd672	Opnd673	Opnd674	Opnd675	Opnd676	Opnd677	Opnd678	Opnd679	Opnd680	Opnd681	Opnd682	Opnd683	Opnd684	Opnd685	Opnd686	Opnd687	Opnd688	Opnd689	Opnd690	Opnd691	Opnd692	Opnd693	Opnd694	Opnd695	Opnd696	Opnd697	Opnd698	Opnd699	Opnd700	Opnd701	Opnd702	Opnd703	Opnd704	Opnd705	Opnd706	Opnd707	Opnd708	Opnd709	Opnd710	Opnd711	Opnd712	Opnd713	Opnd714	Opnd715	Opnd716	Opnd717	Opnd718	Opnd719	Opnd720	Opnd721	Opnd722	Opnd723	Opnd724	Opnd725	Opnd726	Opnd727	Opnd728	Opnd729	Opnd730	Opnd731	Opnd732	Opnd733	Opnd734	Opnd735	Opnd736	Opnd737	Opnd738	Opnd739	Opnd740	Opnd741	Opnd742	Opnd743	Opnd744	Opnd745	Opnd746	Opnd747	Opnd748	Opnd749	Opnd750	Opnd751	Opnd752	Opnd753	Opnd754	Opnd755	Opnd756	Opnd757	Opnd758	Opnd759	Opnd760	Opnd761	Opnd762	Opnd763	Opnd764	Opnd765	Opnd766	Opnd767	Opnd768	Opnd769	Opnd770	Opnd771	Opnd772	Opnd773	Opnd774	Opnd775	Opnd776	Opnd777	Opnd778	Opnd779	Opnd780	Opnd781	Opnd782	Opnd783	Opnd784	Opnd785	Opnd786	Opnd787	Opnd788	Opnd789	Opnd790	Opnd791	Opnd792	Opnd793	Opnd794	Opnd795	Opnd796	Opnd797	Opnd798	Opnd799	Opnd800	Opnd801	Opnd802	Opnd803	Opnd804	Opnd805	Opnd806	Opnd807	Opnd808	Opnd809	Opnd810	Opnd811	Opnd812	Opnd813	Opnd814	Opnd815	Opnd816	Opnd817	Opnd818	Opnd819	Opnd820	Opnd821	Opnd822	Opnd823	Opnd824	Opnd825	Opnd826	Opnd827	Opnd828	Opnd829	Opnd830	Opnd831	Opnd832	Opnd833	Opnd834	Opnd835	Opnd836	Opnd837	Opnd838	Opnd839	Opnd840	Opnd841	Opnd842	Opnd843	Opnd844	Opnd845	Opnd846	Opnd847	Opnd848	Opnd849	Opnd850	Opnd851	Opnd852	Opnd853	Opnd854	Opnd855	Opnd856	Opnd857	Opnd858	Opnd859	Opnd860	Opnd861	Opnd862	Opnd863	Opnd864	Opnd865	Opnd866	Opnd867	Opnd868	Opnd869	Opnd870	Opnd871	Opnd872	Opnd873	Opnd874	Opnd875	Opnd876	Opnd877	Opnd878	Opnd879	Opnd880	Opnd881	Opnd882	Opnd883	Opnd884	Opnd885	Opnd886	Opnd887	Opnd888	Opnd889	Opnd890	Opnd891	Opnd892	Opnd893	Opnd894	Opnd895	Opnd896	Opnd897	Opnd898	Opnd899	Opnd900	Opnd901	Opnd902	Opnd903	Opnd904	Opnd905	Opnd906	Opnd907	Opnd908	Opnd909	Opnd910	Opnd911	Opnd912	Opnd913	Opnd914	Opnd915	Opnd916	Opnd917	Opnd918	Opnd919	Opnd920	Opnd921	Opnd922	Opnd923	Opnd924	Opnd925	Opnd926	Opnd927	Opnd928	Opnd929	Opnd930	Opnd931	Opnd932	Opnd933	Opnd934	Opnd935	Opnd936	Opnd937	Opnd938	Opnd939	Opnd940	Opnd941	Opnd942	Opnd943	Opnd944	Opnd945	Opnd946	Opnd947	Opnd948	Opnd949	Opnd950	Opnd951	Opnd952	Opnd953	Opnd954	Opnd955	Opnd956	Opnd957	Opnd958	Opnd959	Opnd960	Opnd961	Opnd962	Opnd963	Opnd964	Opnd965	Opnd966	Opnd967	Opnd968	Opnd969	Opnd970	Opnd971	Opnd972	Opnd973	Opnd974	Opnd975	Opnd976	Opnd977	Opnd978	Opnd979	Opnd980	Opnd981	Opnd982	Opnd983	Opnd984	Opnd985	Opnd986	Opnd987	Opnd988	Opnd989	Opnd990	Opnd991	Opnd992	Opnd993	Opnd994	Opnd995	Opnd996	Opnd997	Opnd998	Opnd999	Opnd1000	Opnd1001	Opnd1002	Opnd1003	Opnd1004	Opnd1005	Opnd1006	Opnd1007	Opnd1008	Opnd1009	Opnd1010	Opnd1011	Opnd1012	Opnd1013	Opnd1014	Opnd1015	Opnd1016	Opnd1017	Opnd1018	Opnd1019	Opnd1020	Opnd1021	Opnd1022	Opnd1023	Opnd1024	Opnd1025	Opnd1026	Opnd1027	Opnd1028	Opnd1029	Opnd1030	Opnd1031	Opnd1032	Opnd1033	Opnd1034	Opnd1035	Opnd1036	Opnd1037	Opnd1038	Opnd1039	Opnd1040	Opnd1041	Opnd1042	Opnd1043	Opnd1044	Opnd1045	Opnd1046	Opnd1047	Opnd1048	Opnd1049	Opnd1050	Opnd1051	Opnd1052	Opnd1053	Opnd1054	Opnd1055	Opnd1056	Opnd1057	Opnd1058	Opnd1059	Opnd1060	Opnd1061	Opnd1062	Opnd1063	Opnd1064	Opnd1065	Opnd1066	Opnd1067	Opnd1068	Opnd1069	Opnd1070	Opnd1071	Opnd1072	Opnd1073	Opnd1074	Opnd1075	Opnd1076	Opnd1077	Opnd1078	Opnd1079	Opnd1080	Opnd1081	Opnd1082	Opnd1083	Opnd1084	Opnd1085	Opnd1086	Opnd1087	Opnd1088	Opnd1089	Opnd1090	Opnd1091	Opnd1092	Opnd1093	Opnd1094	Opnd1095	Opnd1096	Opnd1097	Opnd1098	Opnd1099	Opnd1100	Opnd1101	Opnd1102	Opnd1103	Opnd1104	Opnd1105	Opnd1106	Opnd1107	Opnd1108	Opnd1109	Opnd1110	Opnd1111	Opnd1112	Opnd1113	Opnd1114	Opnd1115	Opnd1116	Opnd1117	Opnd1118	Opnd1119	Opnd1120	Opnd1121	Opnd1122	Opnd1123	Opnd1124	Opnd1125	Opnd1126	Opnd1127	Opnd1128	Opnd1129	Opnd1130	Opnd1131	Opnd1132	Opnd1133	Opnd1134	Opnd1135	Opnd1136	Opnd1137	Opnd1138	Opnd1139	Opnd1140	Opnd1
------	----	-------	-------	-------	-------	-------	-------	-------	-------	-------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	-------

# Compiler Explorer integration

Live demo: <https://hpc-mover.rrze.uni-erlangen.de/compiler-explorer/z/va3zxe>



&



# Supported micro-architectures so far

---

- **Intel x86:**

Sandy Bridge, Ivy Bridge, Haswell, Broadwell,  
Skylake X, Cascade Lake X, Ice Lake Server

- **AMD x86:**

Zen 1, Zen 2, Zen 3

- **AArch64:**

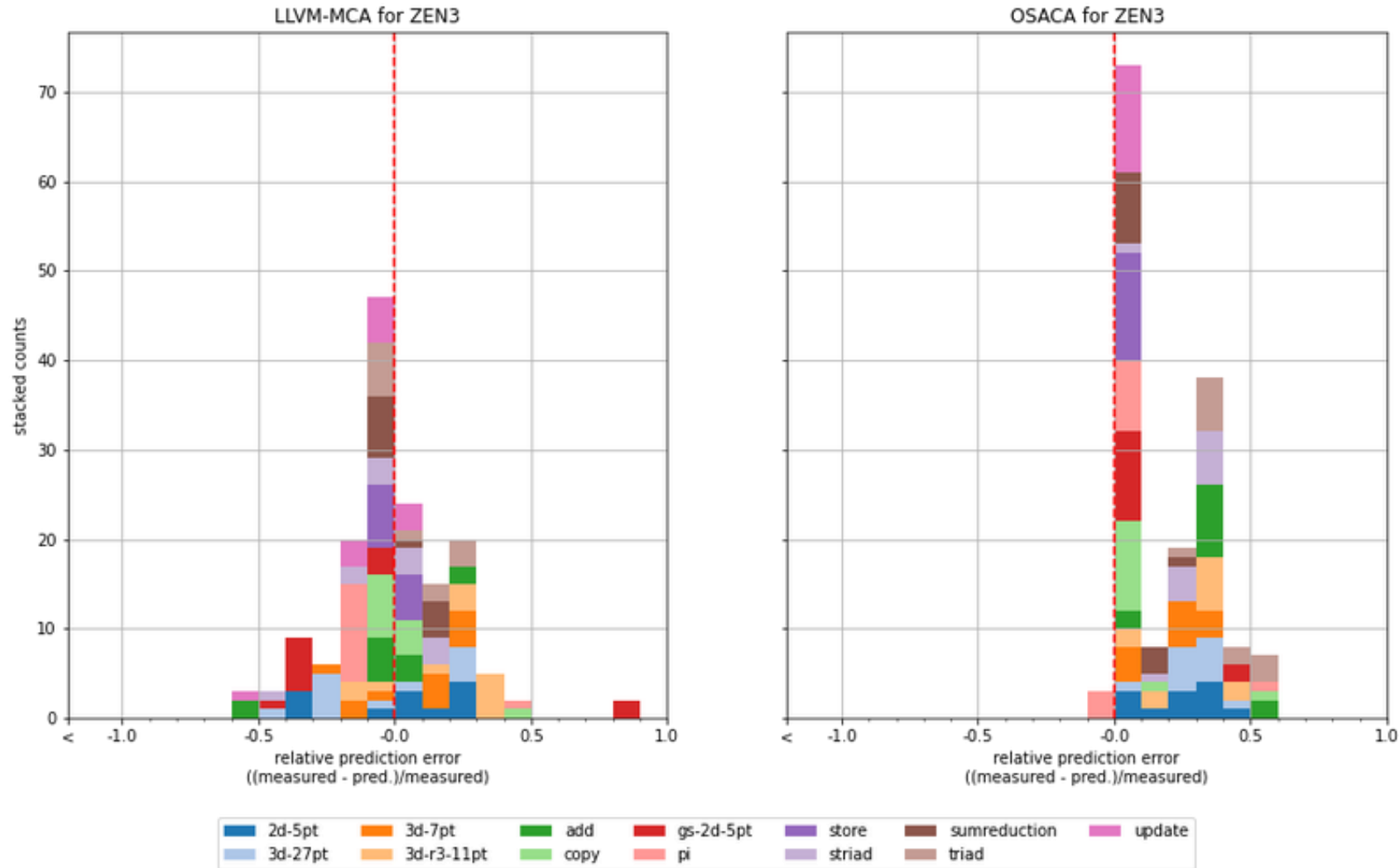
Marvell ThunderX2, ARM Neoverse N1, ARM Cortex A72,  
Fujitsu A64FX, HiSilicon TaiShan v110

# There is not just THE one code analyzer

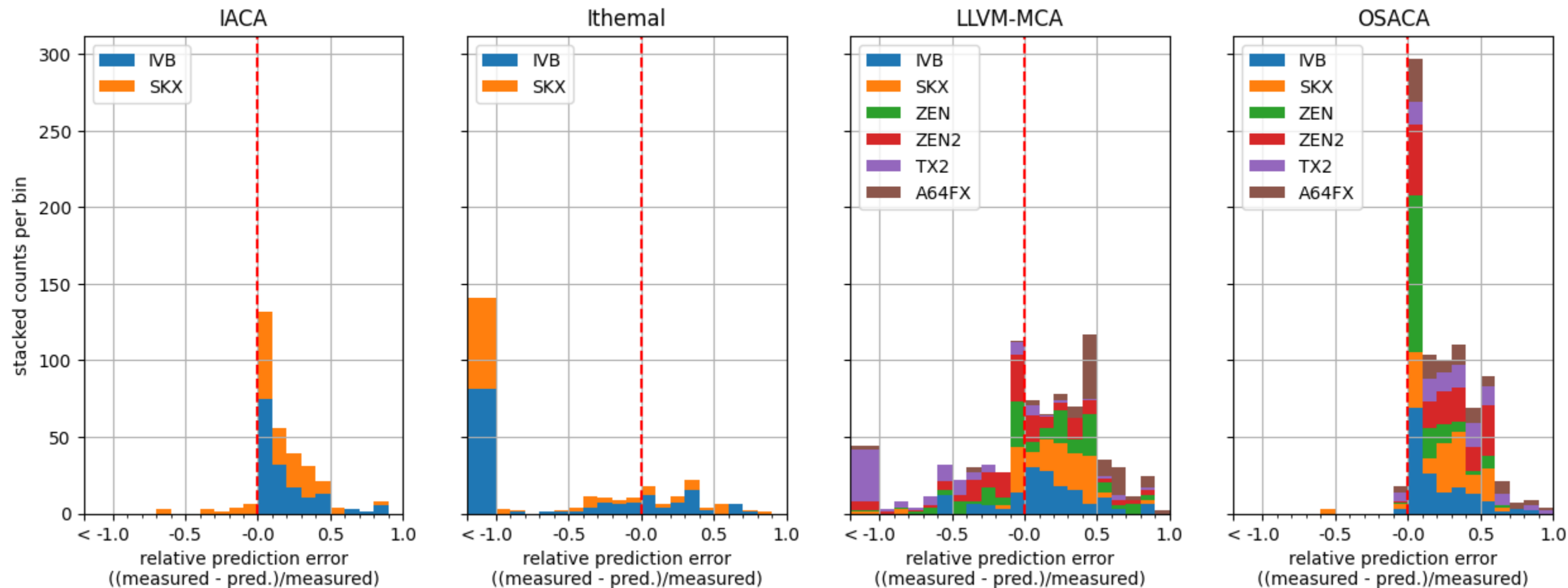
- **OSACA:** <https://github.com/RRZE-HPC/OSACA>
- **uiCA:** <https://www.uops.info/uiCA.html>
- **LLVM-MCA:** <https://llvm.org/docs/CommandGuide/llvm-mca.html>
- **IACA (EoL):**  
<https://www.intel.com/content/www/us/en/developer/articles/tool/architecture-code-analyzer.html>

# Accuracy

## ■ Comparison LLVM-MCA vs OSACA for Zen3



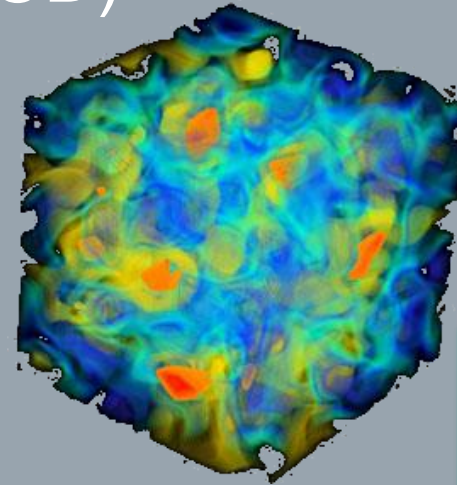
# Accuracy





# Case Study: Domain Wall (DW) Kernel

from Quantum Chromodynamics (QCD)



© Brookhaven National Lab

Based on:

C. Alappat, N. Meyer, J. Laukemann, T. Gruber, G. Hager, G. Wellein, and T. Wettig:  
*ECM modeling and performance tuning of SpMV and Lattice QCD on A64FX.*  
Concurrency and Computation: Practice and Experience, e6512 (2021).

DOI: [10.1002/cpe.6512](https://doi.org/10.1002/cpe.6512)

# Context

- Lattice QCD simulates the strong interaction
- Iterative multigrid techniques on regular (4D or 5D) lattices
- Core component: Apply Dirac operator  $D$  to quark-field vector  $\Psi$
- Domain Wall (DW) formulation: quark field lives on 4D boundary of a 5D space-time volume  $V_4 \times L_s$

$$(D\Psi)(n, s)_{\alpha a} =$$

$$\sum_{\mu=1} \sum_{\beta=1} \sum_{b=1} \left\{ U_{\mu}(n)_{ab} (1 + \gamma_{\mu})_{\alpha\beta} \psi(n + \hat{\mu}, s)_{\beta b} + U_{\mu}^{\dagger}(n - \hat{\mu})_{ab} (1 - \gamma_{\mu})_{\alpha\beta} \psi(n - \hat{\mu}, s)_{\beta b} \right\}$$

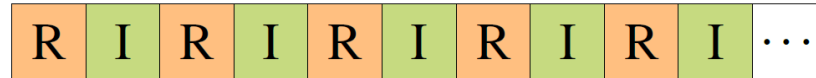
# DW stencil kernel (simplified)

```
#define x_p 1 // x-plus direction
#define x_m 2 // x-minus direction
#define y_p 3 // y-plus direction
...
#pragma omp parallel for schedule(static)
for{t,z,y,x} = 1:{Lt-2,Lz-2,Ly-2,Lx-2} //collapsed loop over 4d space-time
{
    for(int s=0; s<Ls; ++s) //loop over fifth dimension
    {
        O[t][z][y][x][s] = R(x_p) · U[x_p][t][z][y][x] · P(x_p) · I[t][z][y][x+1][s] +
                           R(x_m) · U[x_m][t][z][y][x] · P(x_m) · I[t][z][y][x-1][s] +
                           R(y_p) · U[y_p][t][z][y][x] · P(y_p) · I[t][z][y+1][x][s] +
                           R(y_m) · U[y_m][t][z][y][x] · P(y_m) · I[t][z][y-1][x][s] +
                           R(z_p) · U[z_p][t][z][y][x] · P(z_p) · I[t][z+1][y][x][s] +
                           R(z_m) · U[z_m][t][z][y][x] · P(z_m) · I[t][z-1][y][x][s] +
                           R(t_p) · U[t_p][t][z][y][x] · P(t_p) · I[t+1][z][y][x][s] +
                           R(t_m) · U[t_m][t][z][y][x] · P(t_m) · I[t-1][z][y][x][s];
    }
}
```

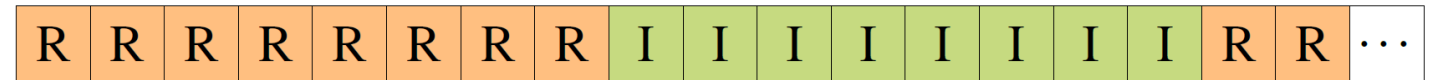
- “Grid” lattice QCD framework
- Uses SVE intrinsics
- Data type: double complex

# Complex numbers data layout choice

AoS (standard)




AoSoA



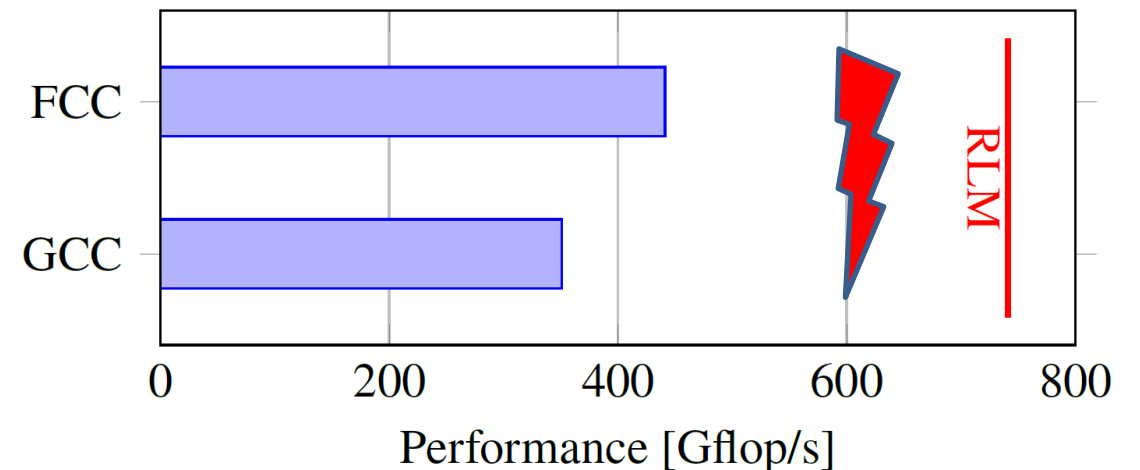
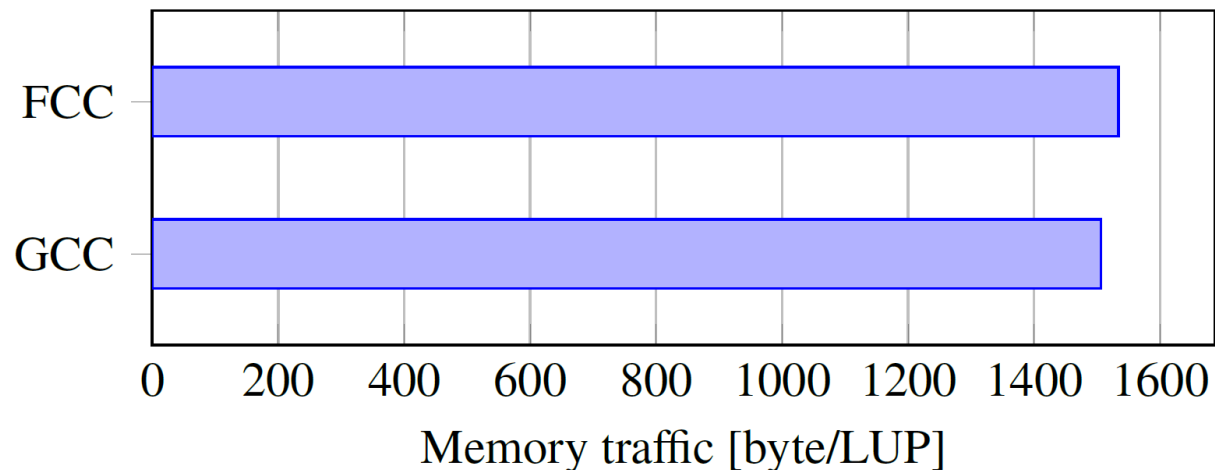
vector length

# Observed performance

- Starting point: AoS layout, ACLE intrinsics, GCC/FCC
  - 1320 flops/LUP (theoretical)
  - Measured code balance: 1500 byte/LUP
  - A64FX (FX1000):  $B_m = 0.25 \frac{\text{byte}}{\text{flop}} \rightarrow$  expect memory bound
- 
- $$B_c \approx 1.14 \frac{\text{byte}}{\text{flop}}$$

# Observed performance

- Starting point: AoS layout, ACLE intrinsics, GCC/FCC
  - 1320 flops/LUP (theoretical)
  - Measured code balance: 1500 byte/LUP
  - A64FX (FX1000):  $B_m = 0.25 \frac{\text{byte}}{\text{flop}} \rightarrow$  expect memory bound
- $B_c \approx 1.14 \frac{\text{byte}}{\text{flop}}$



# In-core analysis (complex-AoS)

```
$ osaca --arch a64fx riri-base-gcc.s
[...]
```

Combined Analysis Report  
-----

Port pressure in cycles															
	0 - 0DV	1	2	3	4	5 - 5D	6 - 6D	7	CP	LCD					
-----															
560												.L41:			
561	0.00		0.00	0.50	0.50							lsl w2, w13, 3			
562						0.50 0.50	0.50 0.50					ld1d z16.d, p0/z, [x11]			
563	0.00		0.00	0.50	0.50							add x18, sp, 160			
564						0.50 0.50	0.50 0.50					ld1d z18.d, p0/z, [x11, #-4, mul vl]			
565				0.50	0.50							sxtw x2, w2			
566						0.50 0.50	0.50 0.50					ld1d z19.d, p0/z, [x11, #-3, mul vl]			
[...]															
1367	1.00					1.00	1.00					st1d z2.d, p0, [x0, #4, mul vl]			
1368	1.00					1.00	1.00					st1d z13.d, p0, [x0, #5, mul vl]			
1369				0.00	1.00							cmp w14, w13			
1370								1.00				bne .L41			
680		500		30		30		118.5 98.5		118.5 98.5		1.0 158 1.0			

Loop-Carried Dependencies Analysis Report  
-----

```
1360 | 1.0 | add w13, w13, 1 | [1360]
```

# In-core analysis (complex-AoS)

```
$ osaca --arch a64fx riri-base-gcc.s
[...]
```

Combined Analysis Report  
-----

Port pressure in cycles															
	0 - 0DV	1	2	3	4	5 - 5D	6 - 6D	7	CP	LCD					
-----															
560												.L41:			
561	0.00		0.00	0.50	0.50							lsl w2, w13, 3			
562						0.50 0.50	0.50 0.50					ld1d z16.d, p0/z, [x11]			
563	0.00		0.00	0.50	0.50							add x18, sp, 160			
564						0.50 0.50	0.50 0.50					ld1d z18.d, p0/z, [x11, #-4, mul vl]			
565				0.50	0.50							sxtw x2, w2			
566						0.50 0.50	0.50 0.50					ld1d z19.d, p0/z, [x11, #-3, mul vl]			
[...]															
1367	1.00					1.00	1.00					st1d z2.d, p0, [x0, #4, mul vl]			
1368	1.00					1.00	1.00					st1d z13.d, p0, [x0, #5, mul vl]			
1369				0.00	1.00							cmp w14, w13			
1370								1.00				bne .L41			
680		500		30		30		118.5 98.5		118.5 98.5		1.0 158 1.0			

Loop-Carried Dependencies Analysis Report  
-----

```
1360 | 1.0 | add w13, w13, 1 | [1360]
```



# In-core analysis (complex-AoS)



# In-core analysis (complex-AoS)

```
$ osaca --  
[...]
```

Combined A

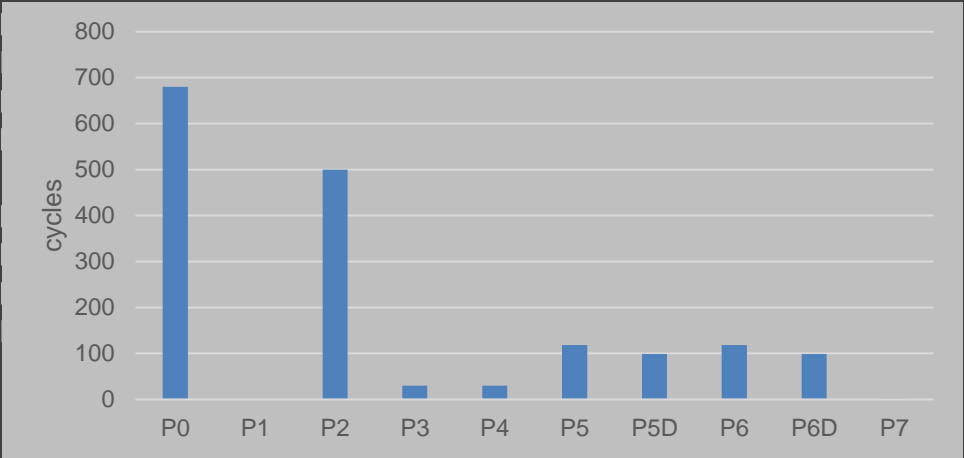
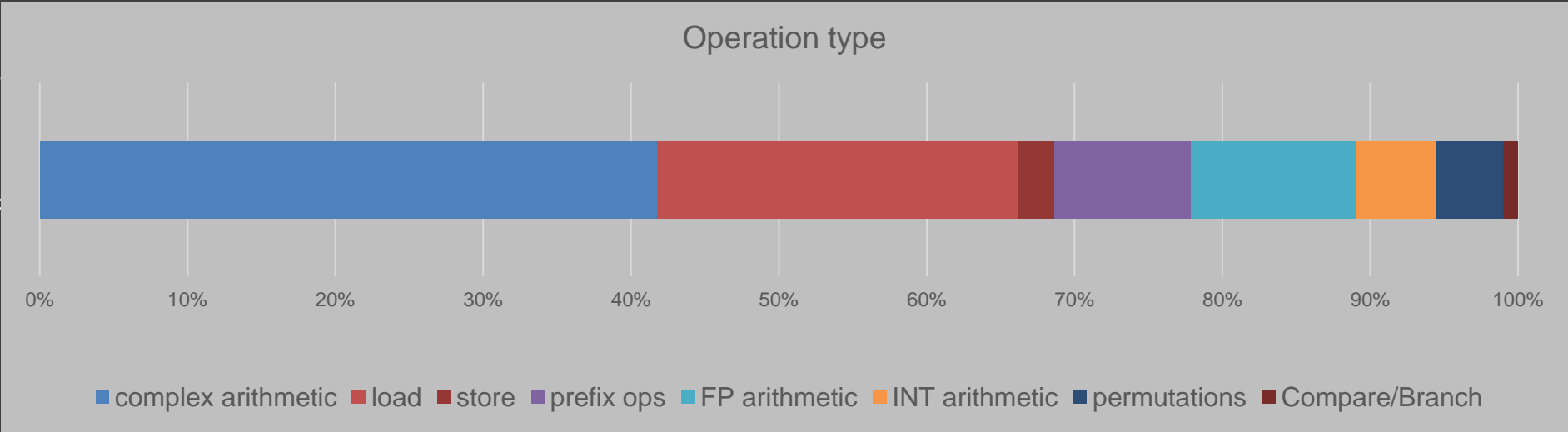
| 0 -

```
560 |  
561 | 0.00  
562 |  
563 | 0.00  
564 |  
565 |
```

5	FCMLA	Zd, Pg, Zn, Zm, c	2cy on P0, 1cy on P2
1	FCADD	Zd, Pg, Zn, Zm, c	1cy on P0, 1cy on P2
1			
1369		0.00 1.00	
1370			1.00
680	500	30 30	118.5 98.5 118.5 98.5 1.0

Loop-Carried Dependencies Analysis Report

```
1360 | 1.0 | add w13, w13, 1 | [1360]
```



# In-core analysis (complex-AoS)

```
$ osaca --  
[...]
```

Combined A

| 0 -

560 |

561 | 0.00

562 |

563 | 0.00

564 |

565 |

5

1

1

1369 |

1370 |

680

500

30

30

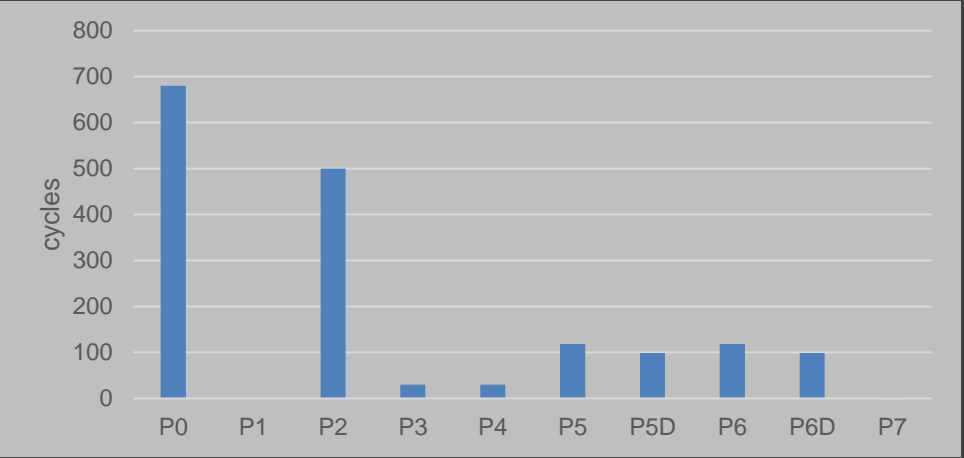
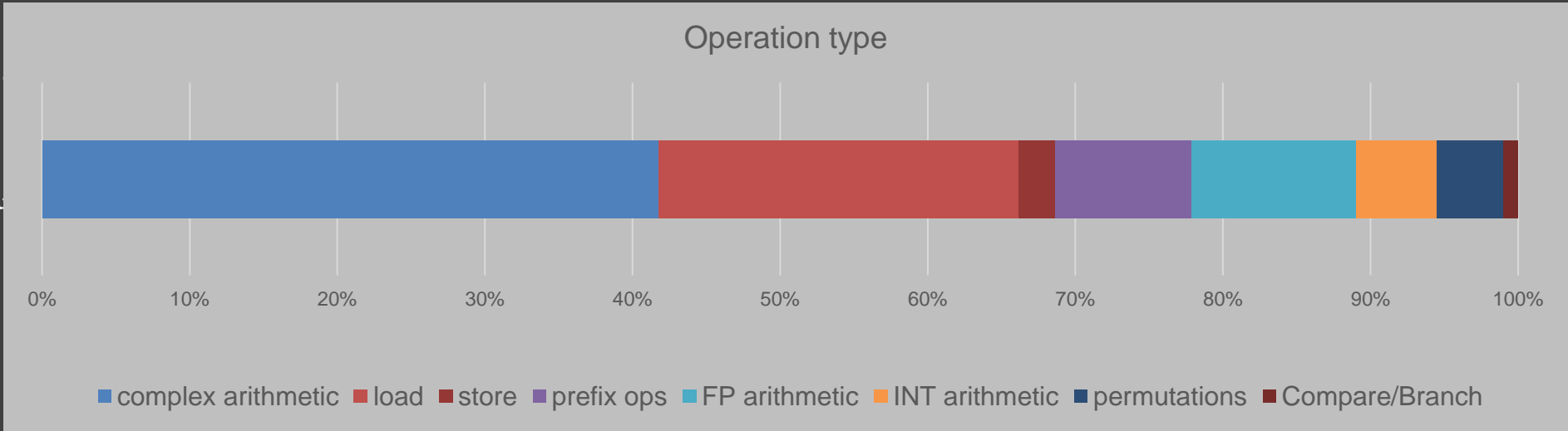
118.5

98.5

118.5

98.5

1.0



## Loop-Carried Dependencies Analysis Report

```
1360 | 1.0 | add w13, w13, 1 | [1360]
```

**FMLA** Zd, Pg, Zn, Zm 1cy on P0 OR P2  
**FADD** Zd, Pg, Zn, Zm 1cy on P0 OR P2

# In-core analysis (complex-AoSoA)

```
$ osaca --arch a64fx rrii-ol-gcc.s
[...]
```

Combined Analysis Report  
-----

Port pressure in cycles																												
		0	-	0DV		1		2		3		4		5	-	5D		6	-	6D		7			CP		LCD	
<hr/>																												
433																												.L66:
434										2.50		2.50																madd x0, x1, x0, x19
435		1.00												0.50				0.50										str x0, [sp, 1896]
436		0.00						0.00		0.50		0.50																add x1, x1, x0
437		1.00												0.50				0.50										str x1, [sp, 1936]
438										0.50		0.50																cmp x0, x1
[...]																												
2803														0.00	0.00			1.00	1.00									ldr x0, [sp, 1784]
2804														0.00				1.00										prfd pldl2strm, p0, [x0]
2805																						1.00						b .L64
2806																												.L38:
2807		0.00						0.00		0.50		0.50																add x1, x1, 1
2808		0.00						0.00		0.00		1.00																mov x19, 0
2809																						1.00						b .L66
		567				1.0		567		247		247		488.5	275.5		488.5	275.5		14				92		1.0		

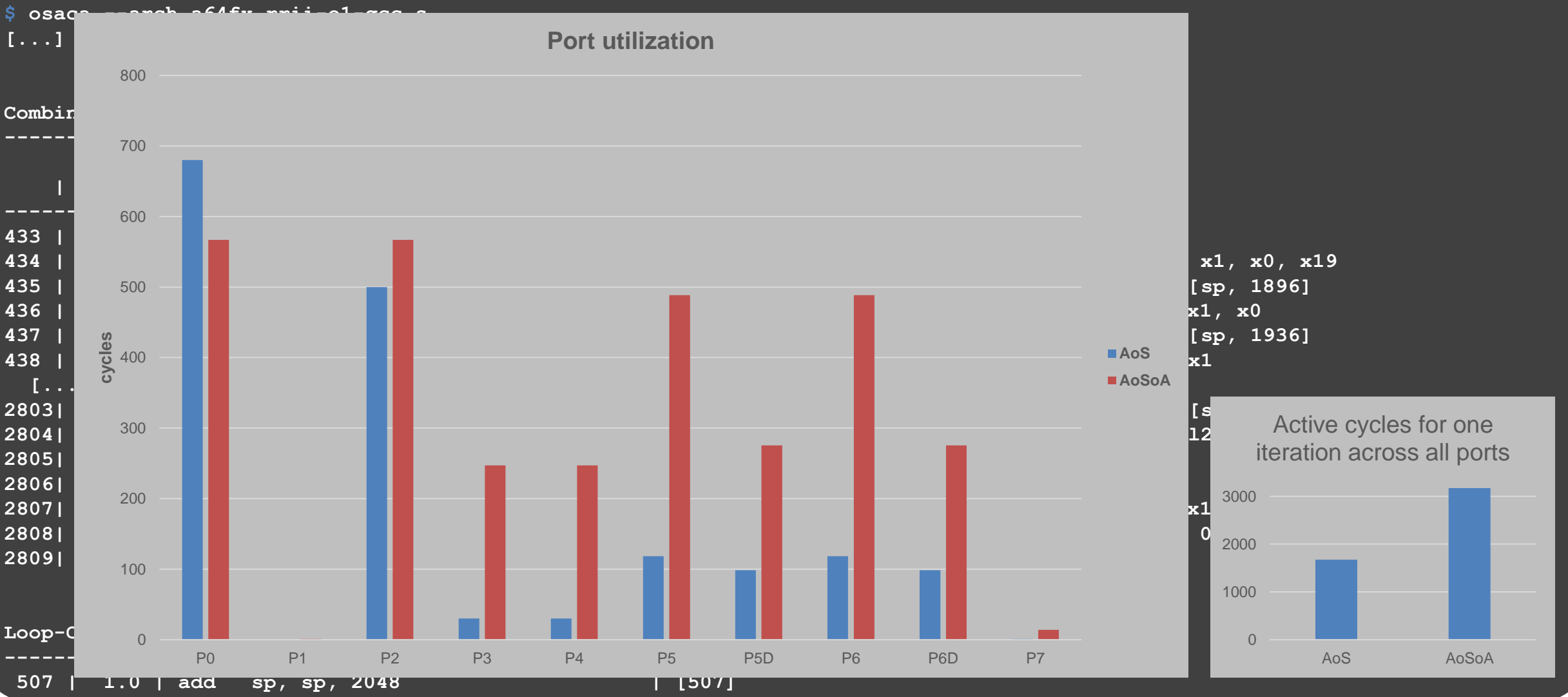
```
.L66:
madd x0, x1, x0, x19
str x0, [sp, 1896]
add x1, x1, x0
str x1, [sp, 1936]
cmp x0, x1
[...]
```

```
ldr x0, [sp, 1784]
prfd pldl2strm, p0, [x0]
b .L64
.L38:
add x1, x1, 1
mov x19, 0
b .L66
```

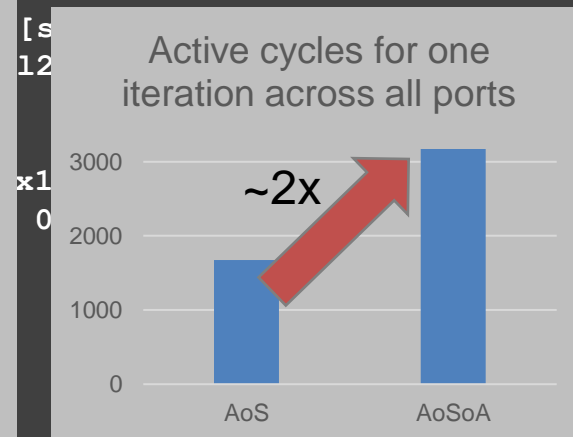
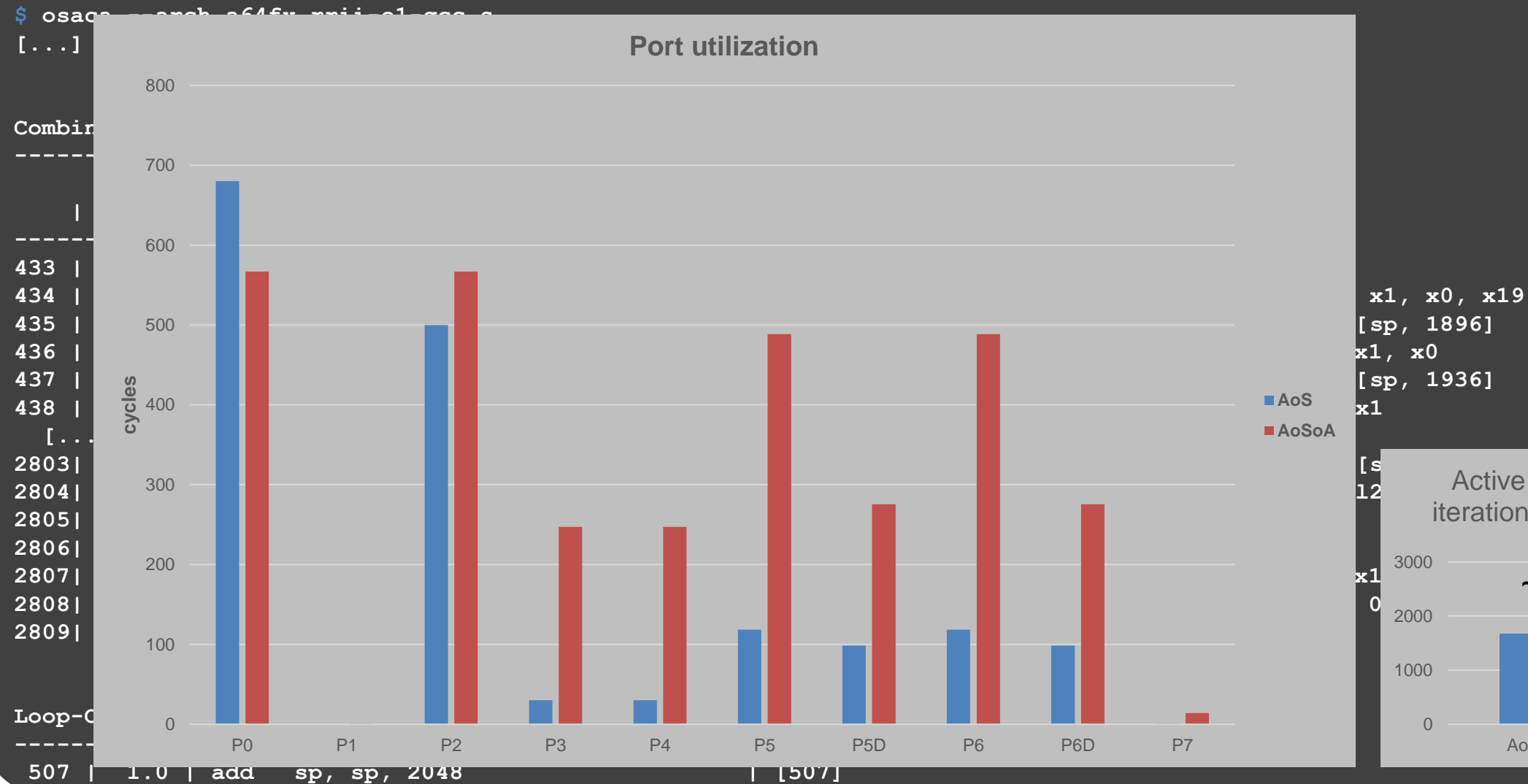
Loop-Carried Dependencies Analysis Report  
-----

```
507 | 1.0 | add sp, sp, 2048 | [507]
```

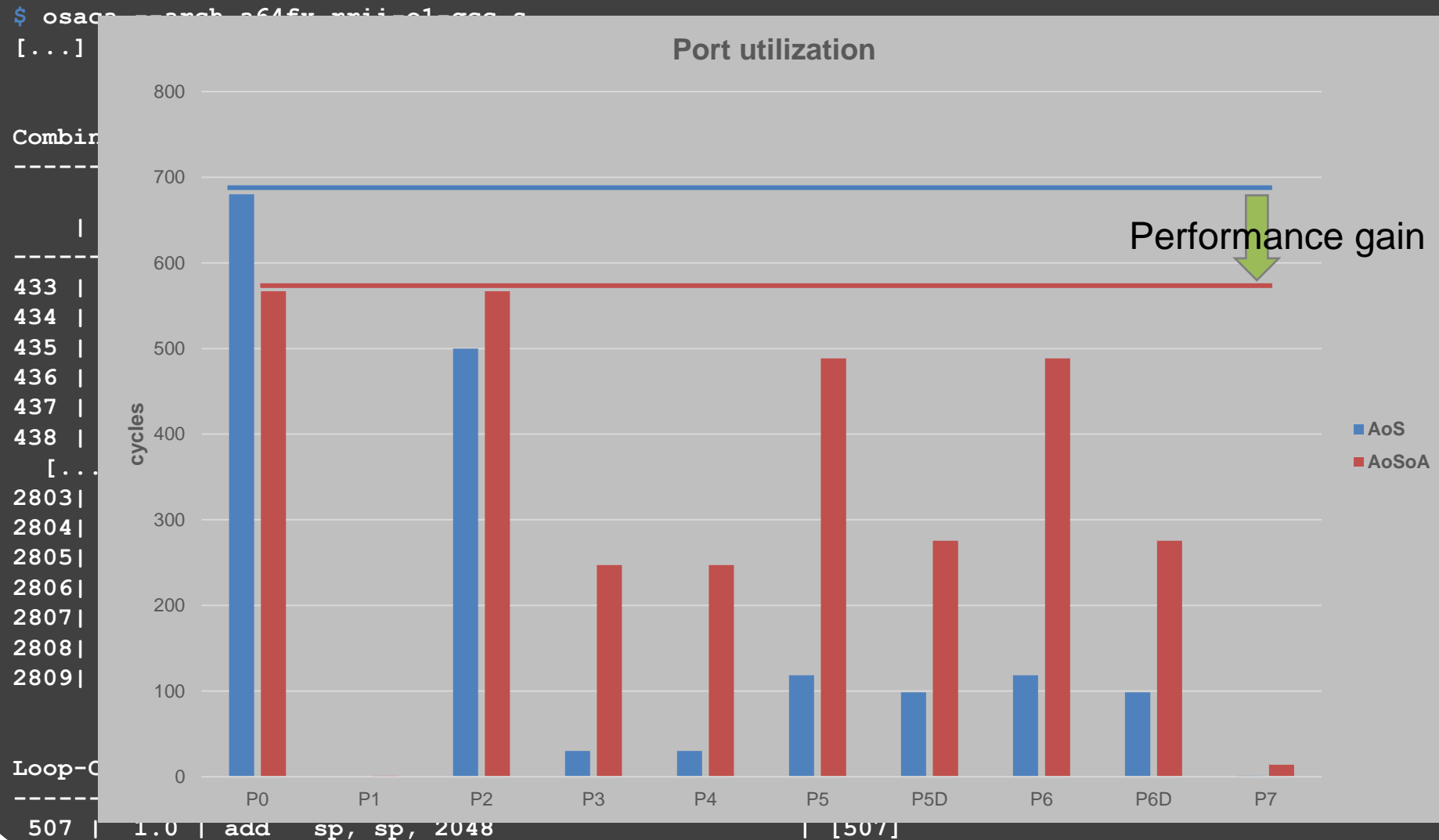
# In-core analysis (complex-AoSoA)



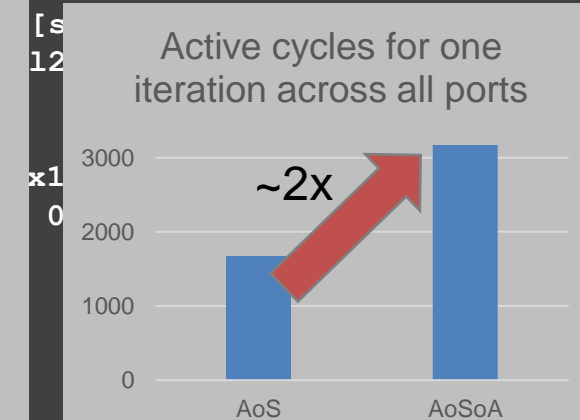
# In-core analysis (complex-AoSOA)



# In-core analysis (complex-AoSoA)

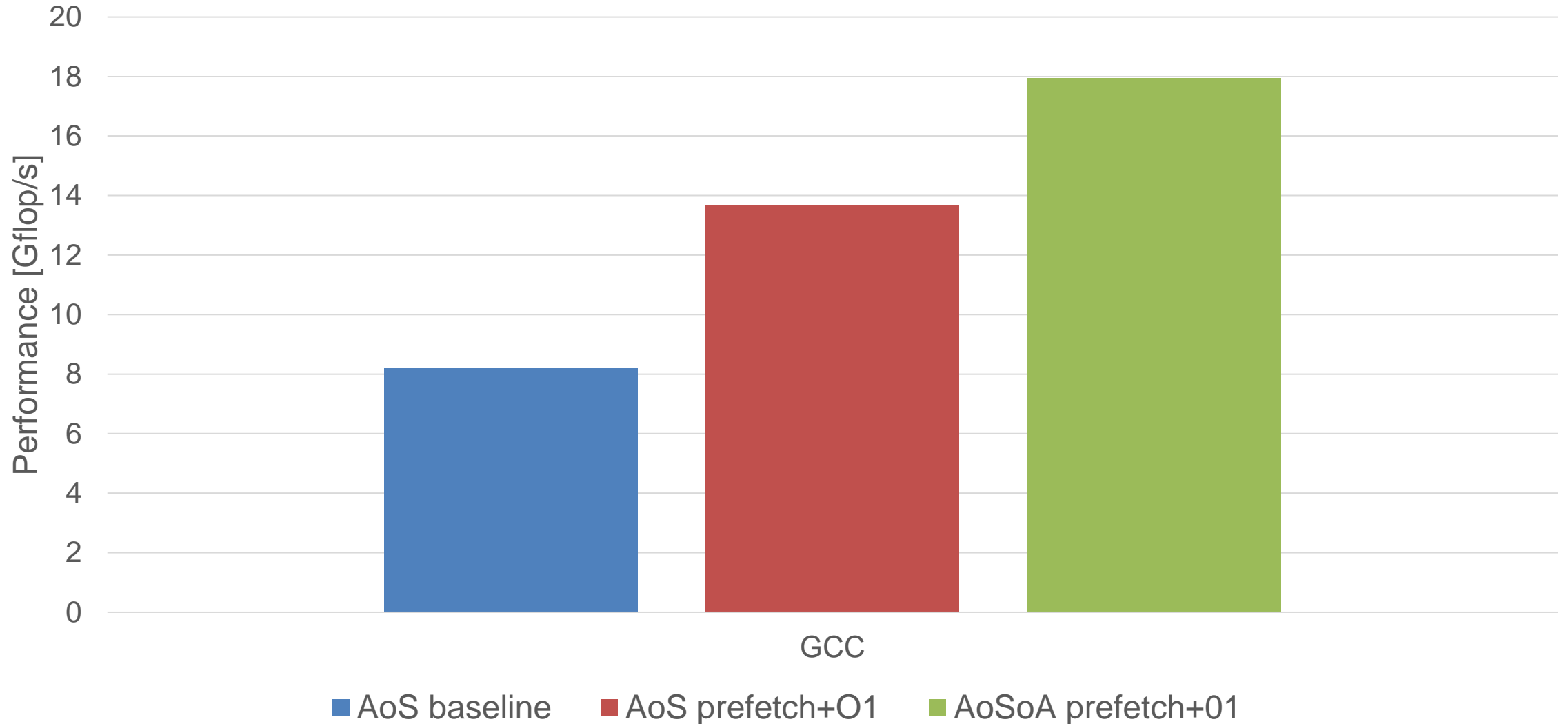


```
x1, x0, x19
[sp, 1896]
x1, x0
[sp, 1936]
x1
```



# DW kernel optimizations

DW kernel





# Summary

---

- static in-core analysis of assembly basic blocks  
→ no compilable asm necessary!
- runtime prediction based on throughput, latency, and loop-carried dependency
- supports various Intel, AMD, and ARM  $\mu$ -architectures
- Python based and available in the Compiler Explorer

# Outlook

---

- Continuously expanding list of supported  $\mu$ -archs (Zen4 and SPR in the pipeline)
- Consideration of  $\mu$ -arch frontends (decode, dispatch, retire, reg files, LSB, ...)
- Support for Intel asm syntax
- Support for IBM Power

Thank you! Questions?



OSACA:

<https://github.com/RRZE-HPC/osaca>

pip: `$ pip install -u osaca`



*Check out our other tools:*

