Hybrid-JIT: Hardware accelerated JIT Compilation for Embedded VLIW Processors

Rokicki Simon – Erven Rohou – Steven Derrien
Irisa Inria Irisa

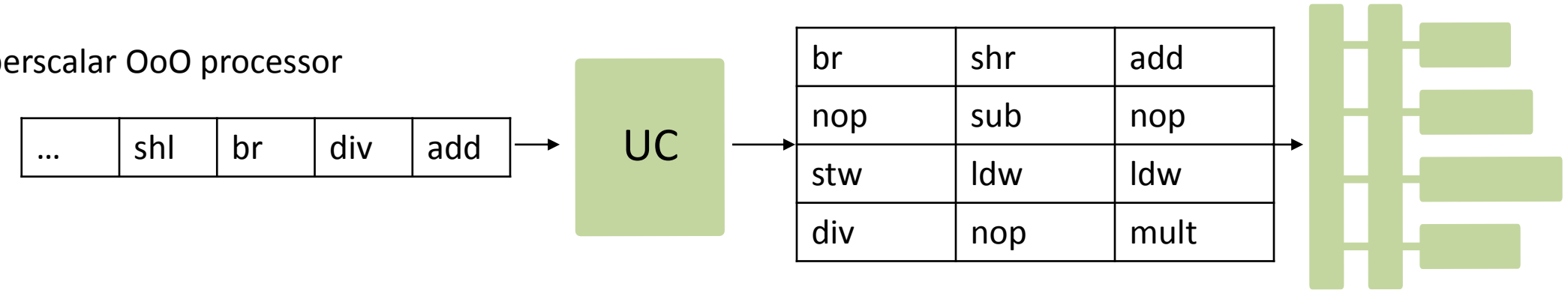
Embedded Systems

High constraints in

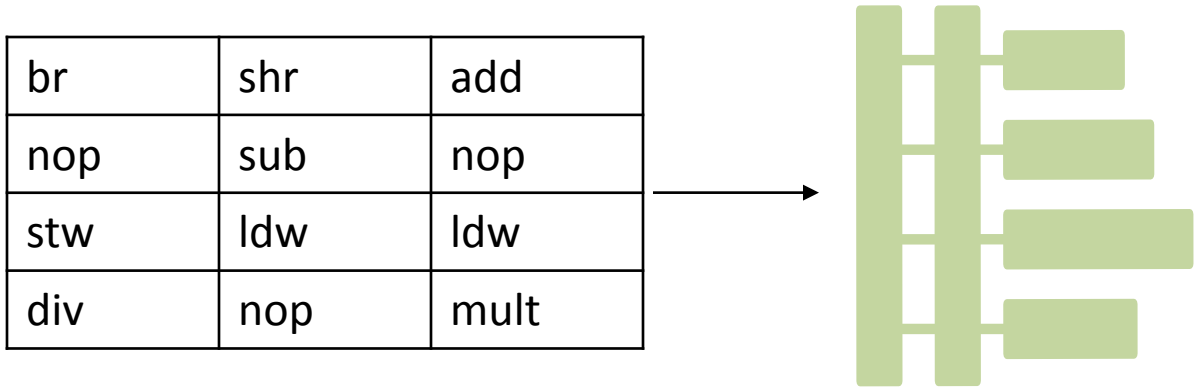
- Power consumption
- Production cost
- Performance

VLIW vs Superscalar OoO

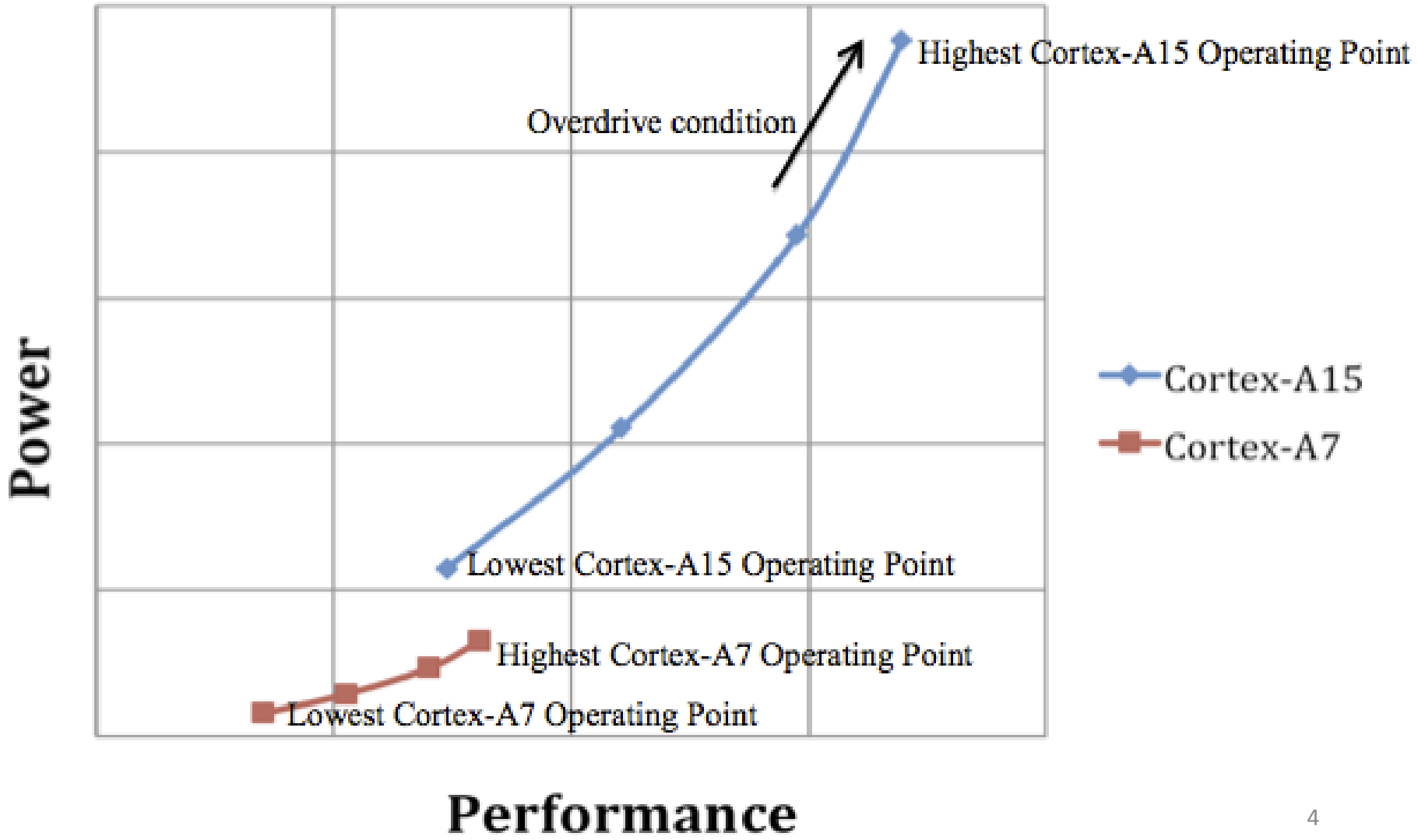
Superscalar OoO processor



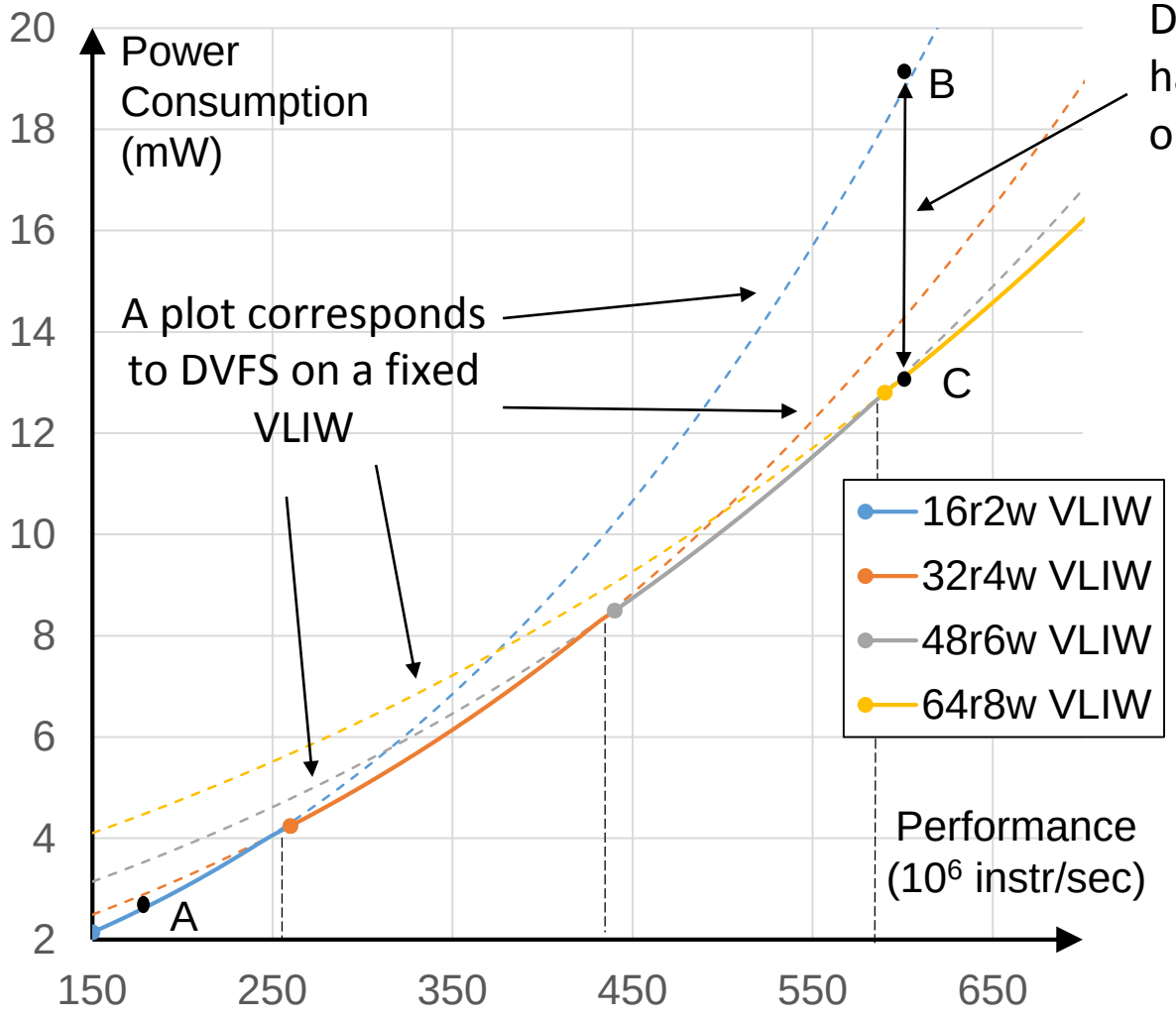
VLIW processor



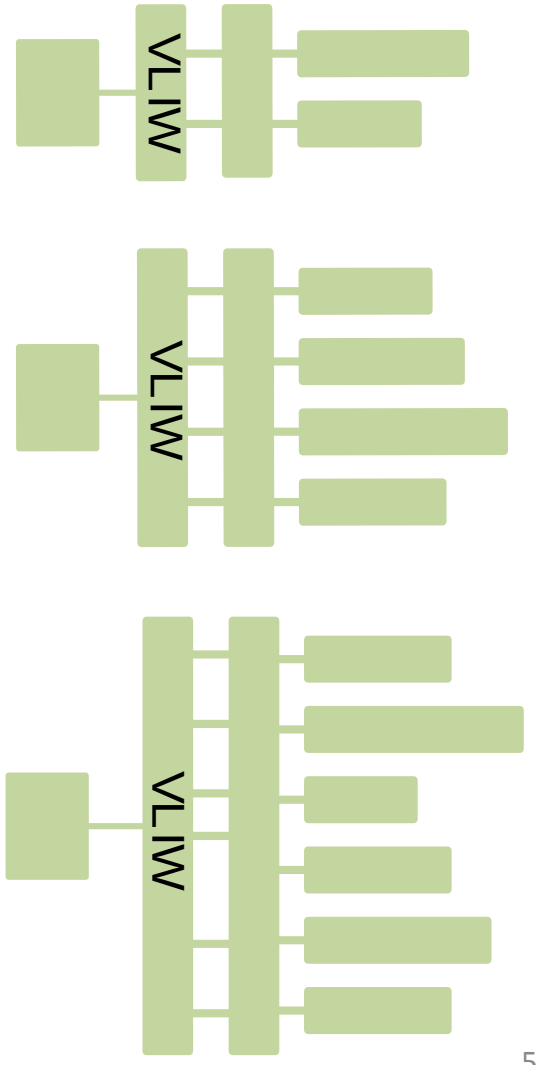
ARM big.LITTLE



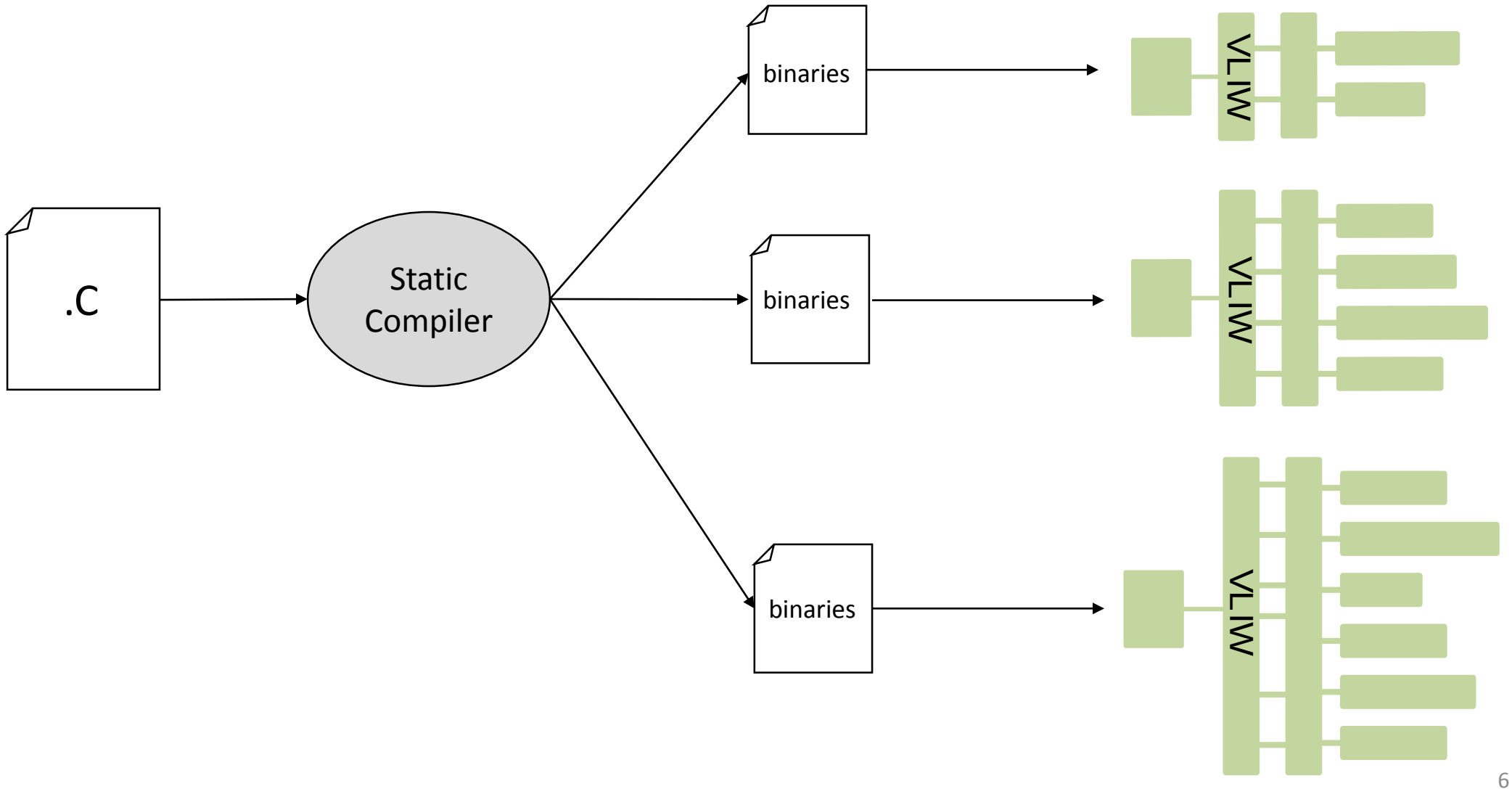
Heterogeneous Multicore VLIW



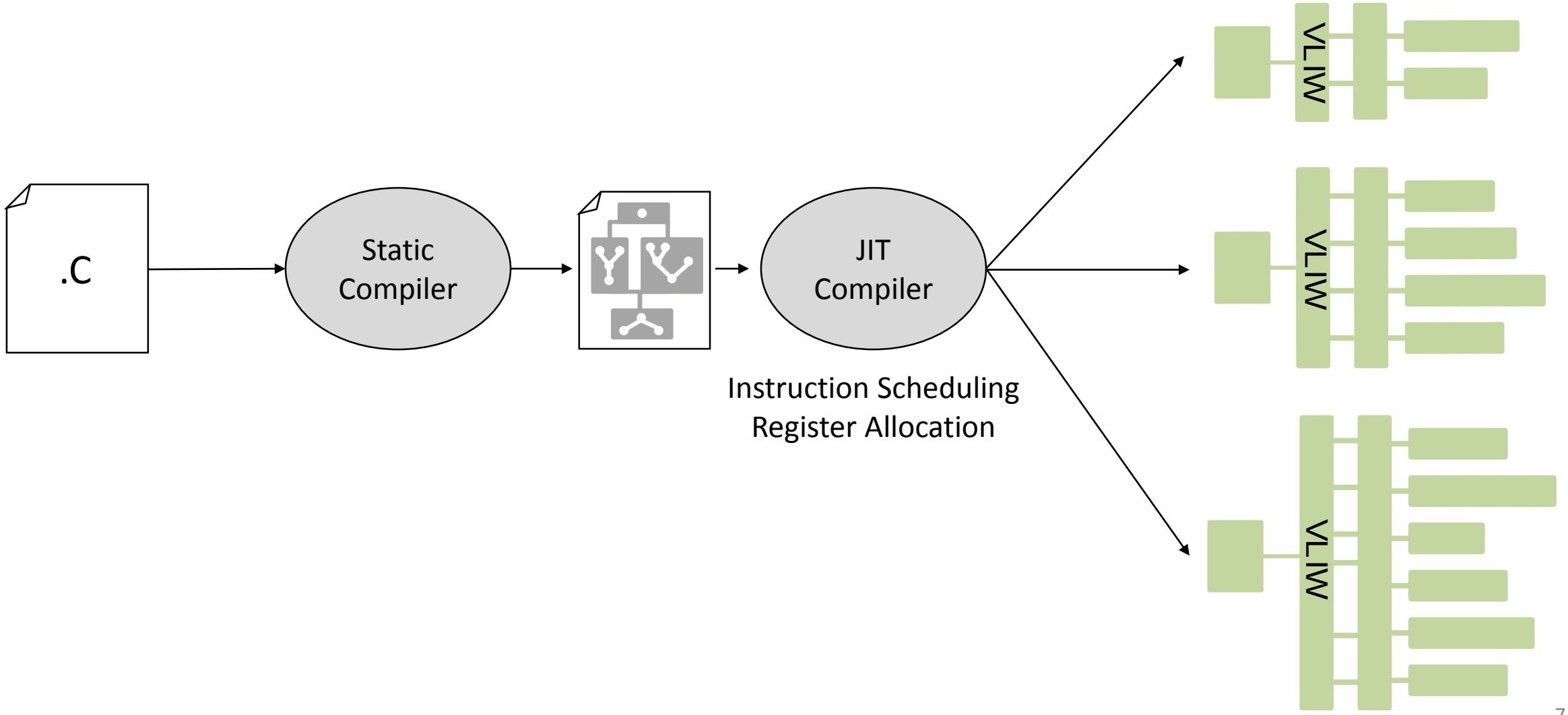
Difference while using hardware reconfiguration on top of DVFS



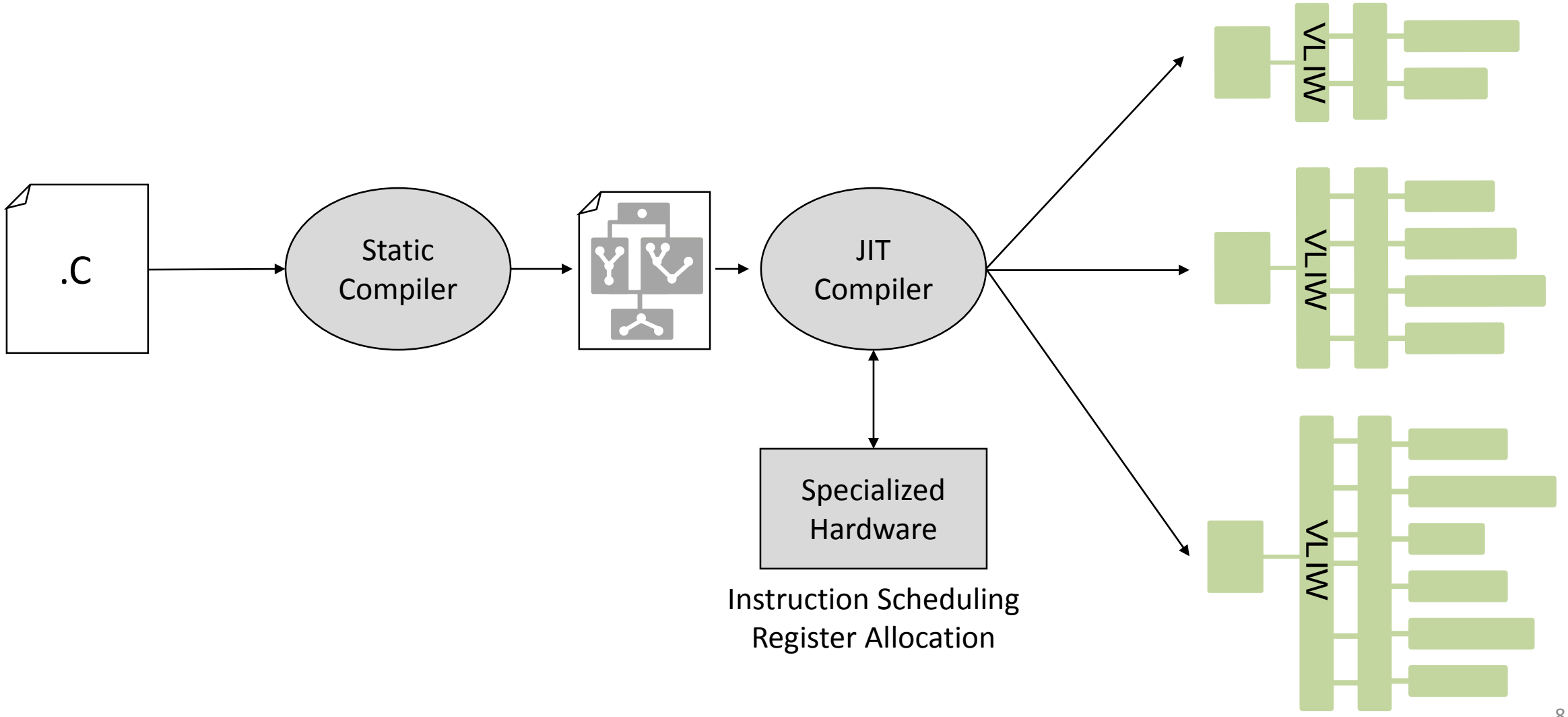
Heterogeneous Multicore VLIW



Heterogeneous Multicore VLIW



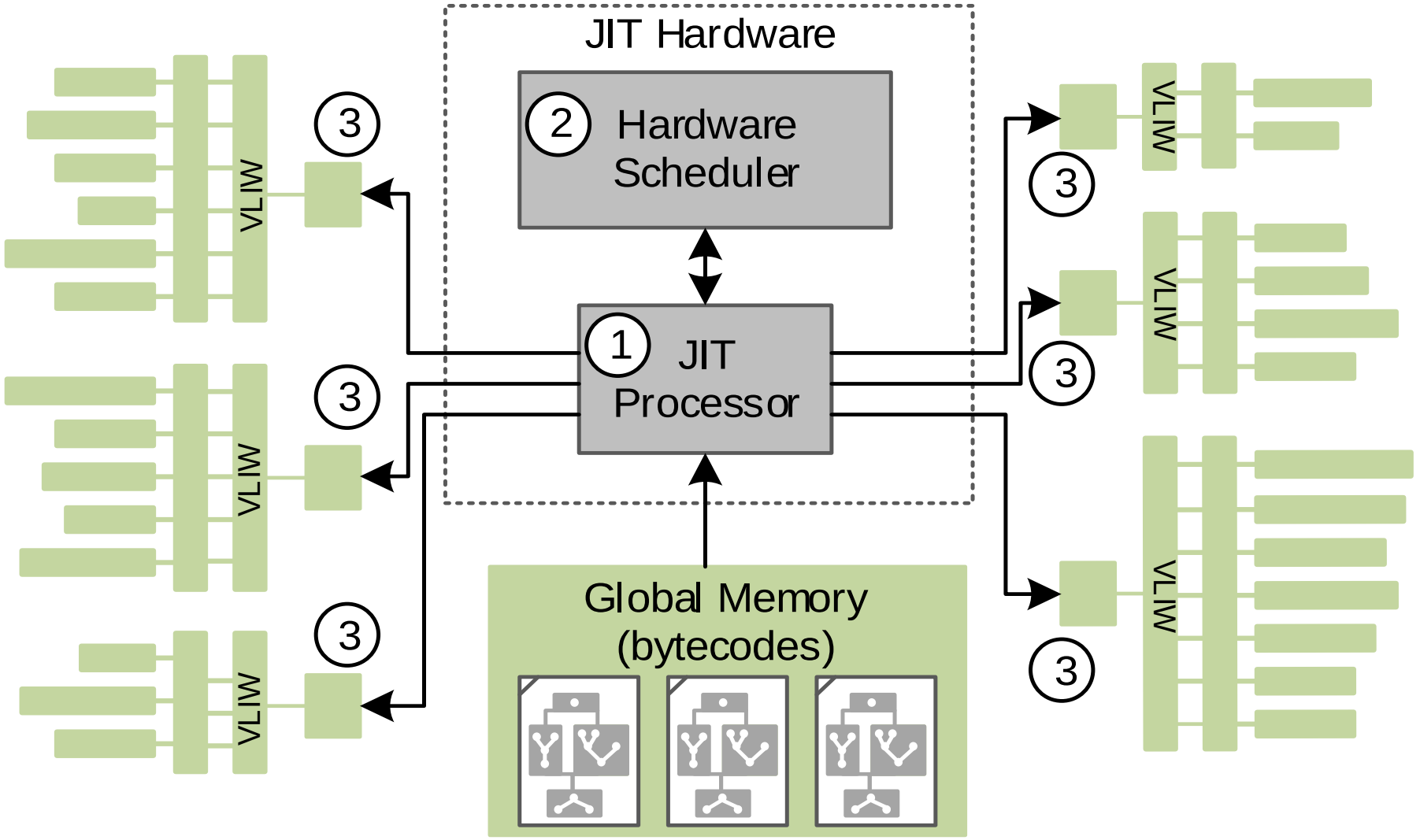
Heterogeneous Multicore VLIW



Outline

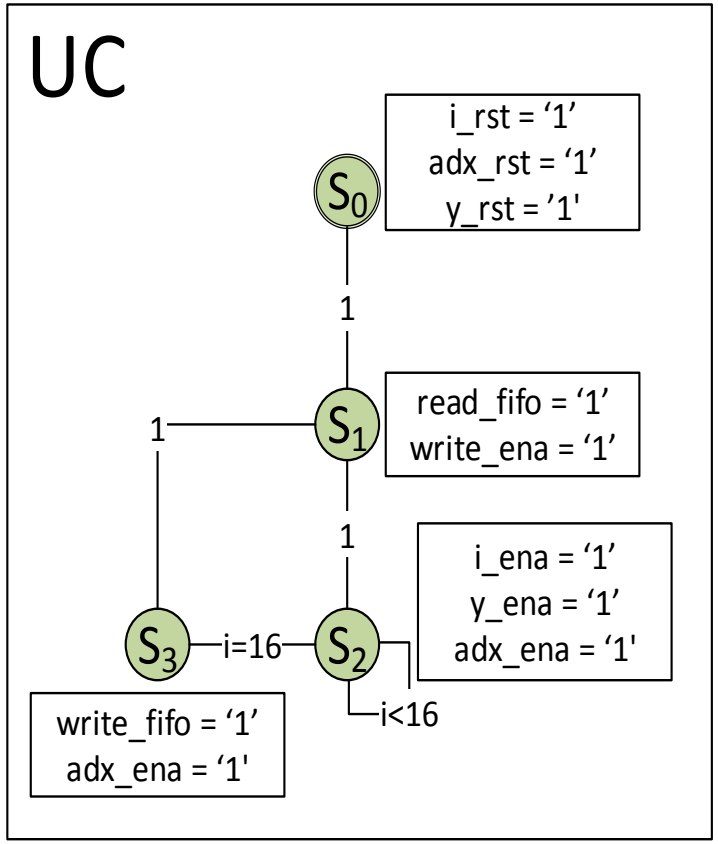
- **Hybrid-JIT Compiler**
  - **Hybrid-JIT platform**
  - **Hardware scheduler**
  - **Custom bytecode**
- Experimental Study
- Conclusion

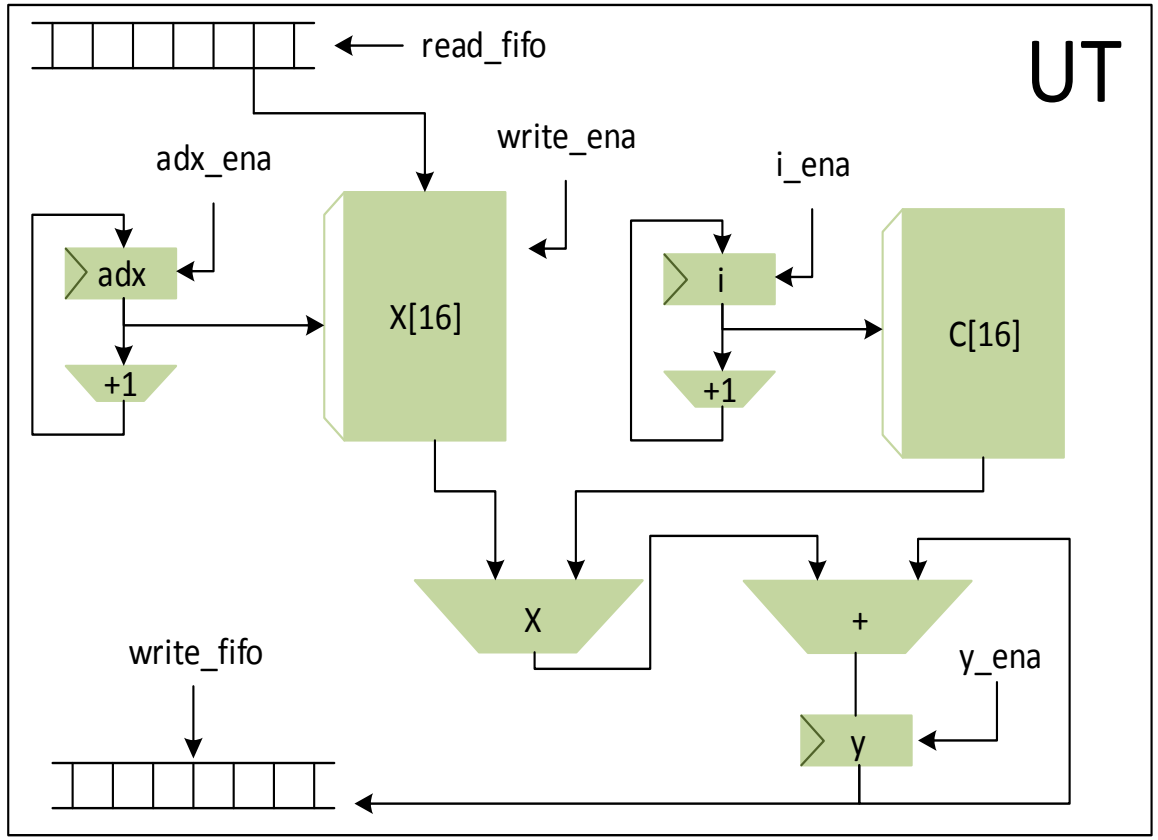
Hybrid-JIT platform



Hardware Development

```
adx = 0
while (1){
  X[adx] = read_fifo();
  y = 0; i = 0;
  adx = adx + 1 % 8;
  while (i<16){
    y = y + X[adx]*C[i];
    adx = adx + 1 % 8;
    i++;
  }
  write_fifo(y);
}
```

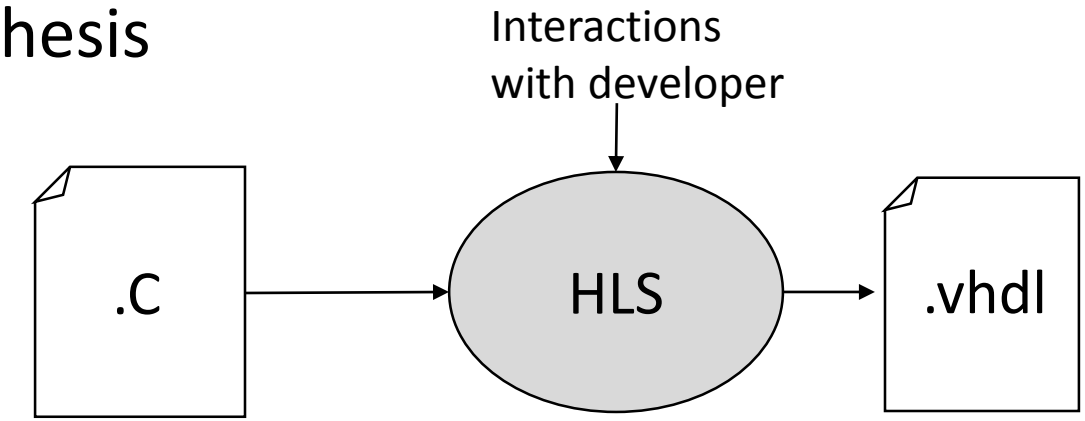


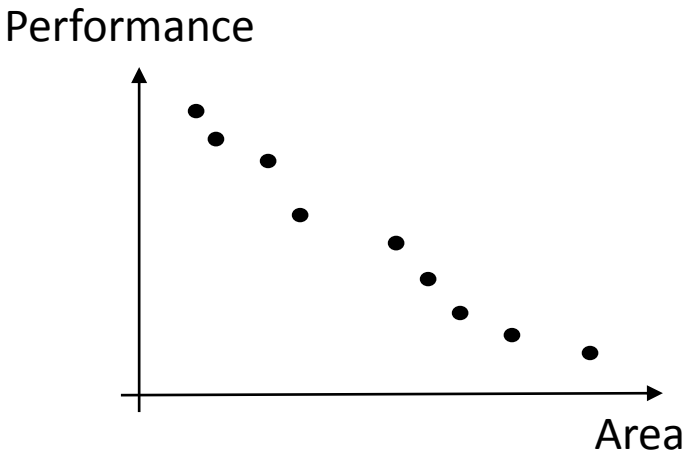


Over 700 lines of VHDL !

Hardware Development

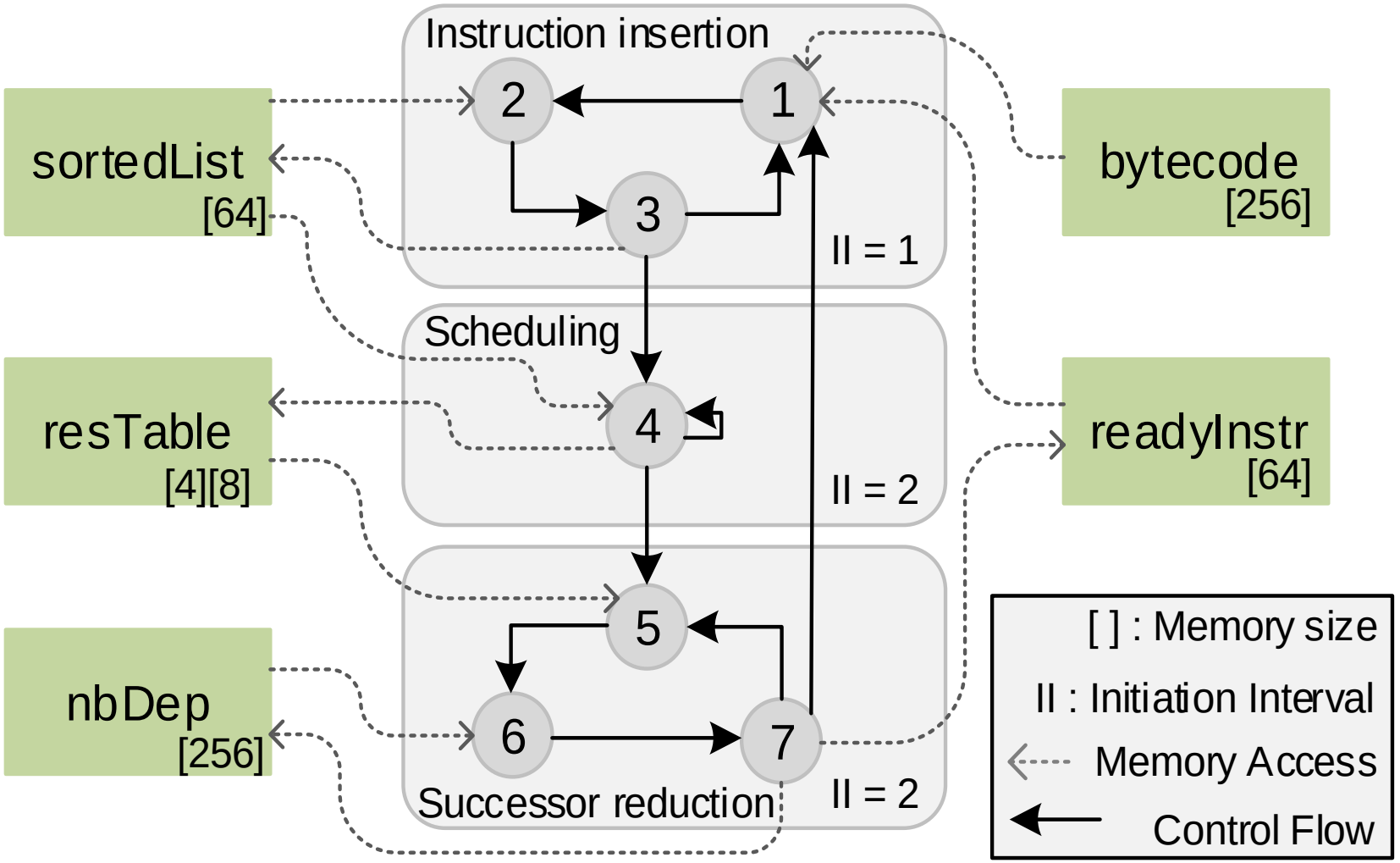
- High-Level Synthesis



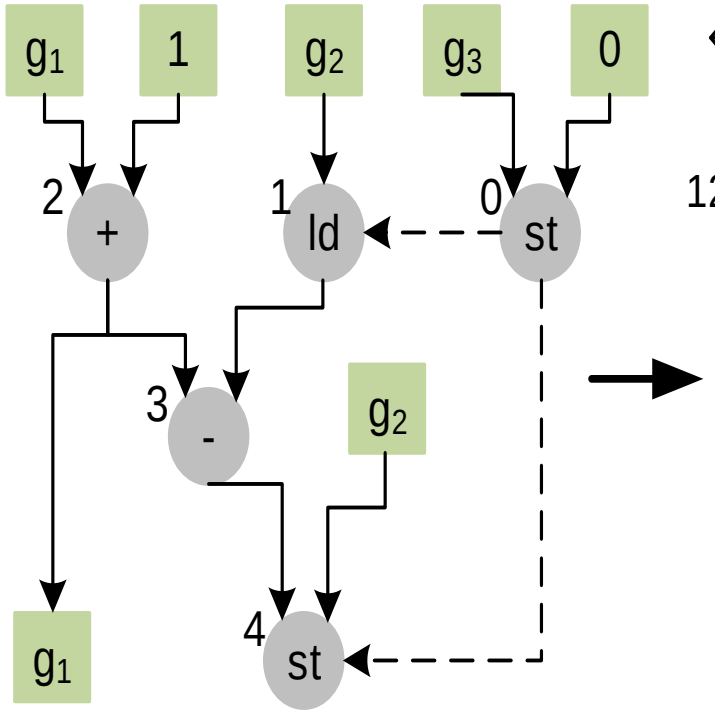


5x productivity vs. VHDL

Hardware scheduler



Custom Bytecode



128 bits instructions

128	96	nbDep	nbDSucc	nbSucc	64	32	0

op	vregs[4]				prior	succNames[7]						
0 - st	@g3 = 0	0	0	2	0	1	4	-	-	-	-	-
1 - ld	r1 = @g2	1	1	1	2	3	-	-	-	-	-	-
2 - addi	g1 = g1 1	0	1	1	2	3	-	-	-	-	-	-
3 - sub	r3 = r1 g1	2	1	1	2	4	-	-	-	-	-	-
4 - st	@g2 = r3	2	0	0	2	-	-	-	-	-	-	-

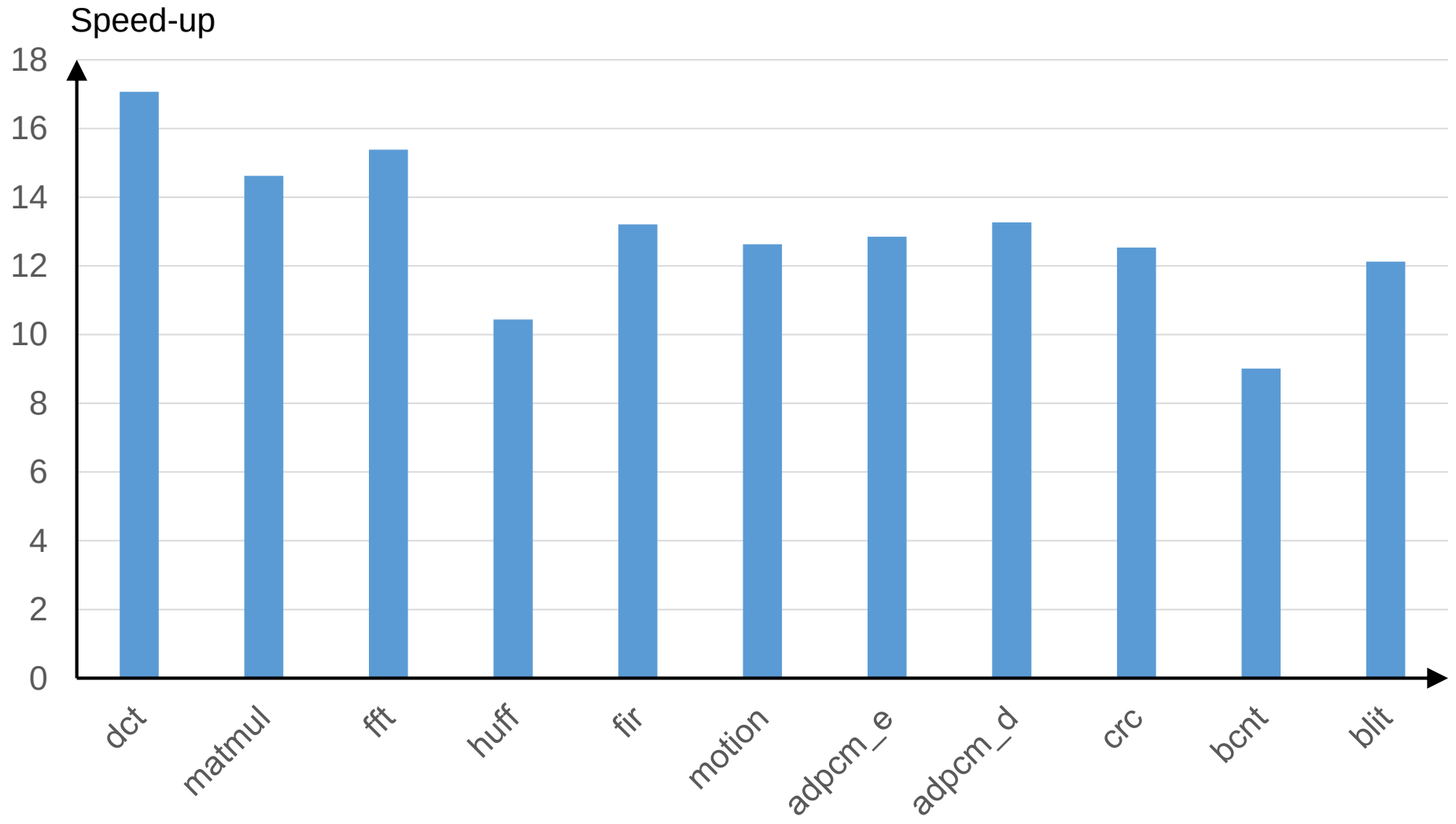
Outline

- Hybrid-JIT Compiler
- **Experimental Study**
  - **Speed-up**
  - **Energy Improvement**
  - **Hardware reconfiguration**
- Conclusion

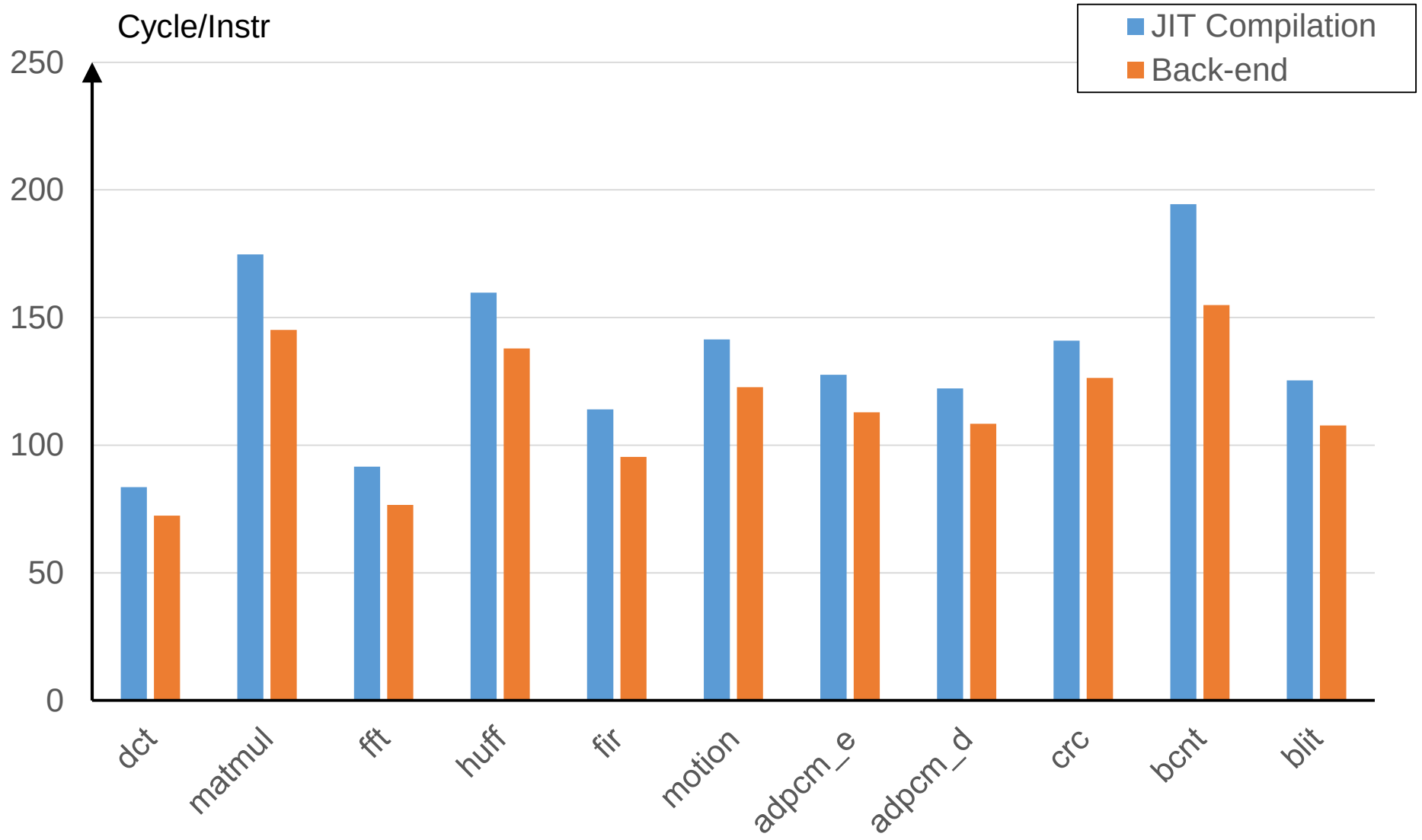
Experimental Study

- Hybrid-JIT platform on FPGA (speed-up):
  - JIT processor : Nios II
  - Altera DE2 115
  - 50 MHz
- Hybrid-JIT platform on ASIC (Energy Consumption + Area):
  - Compiled with design compiler
  - Simulated with ModelSim
  - 250 MHz

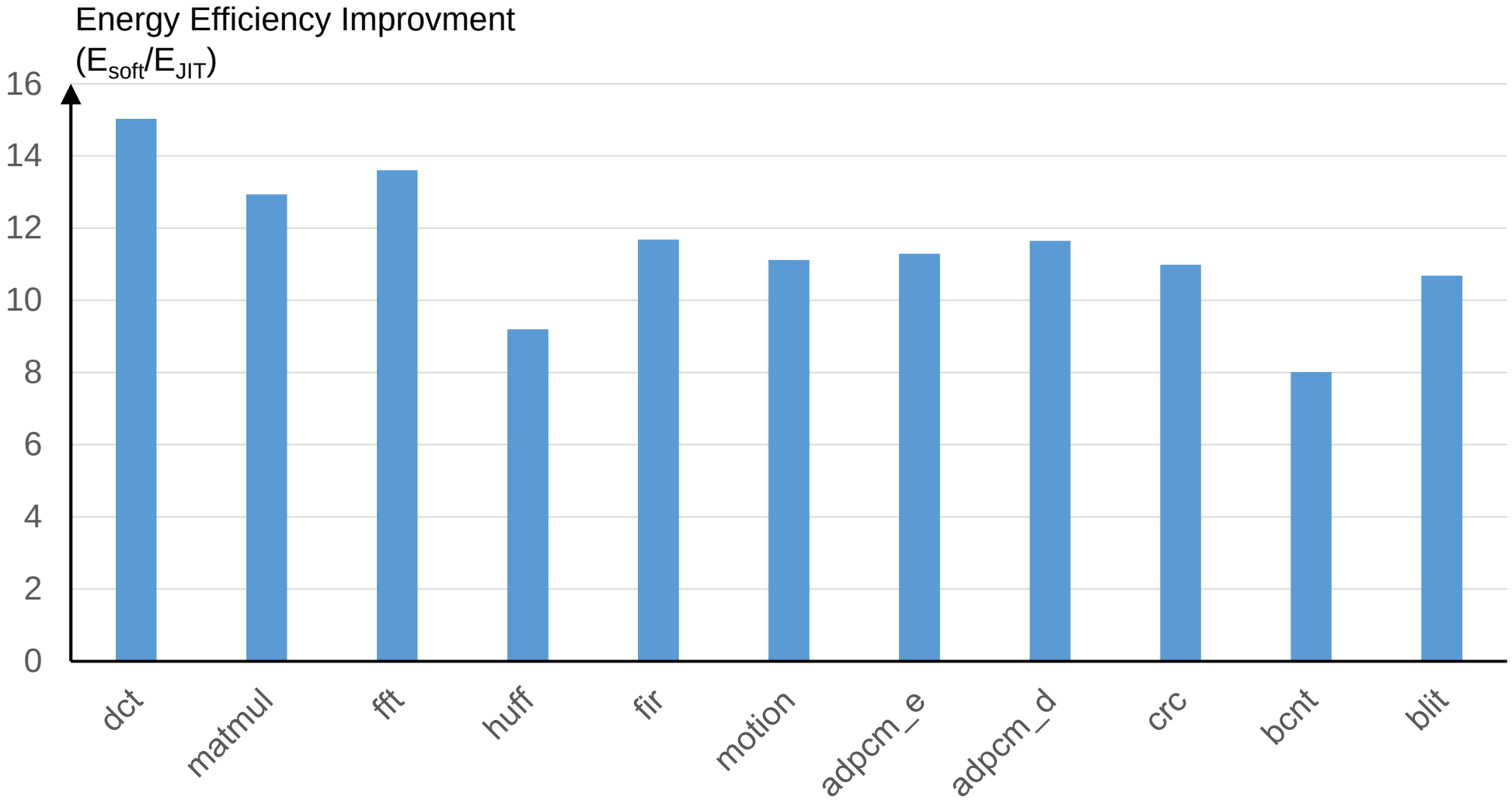
Speed-up


Speed-up
18
16
14
12
10
8
6
4
2
0
dct
matmul
fft
huff
fir
motion
adpcm_e
adpcm_d
crc
bcnt
blit

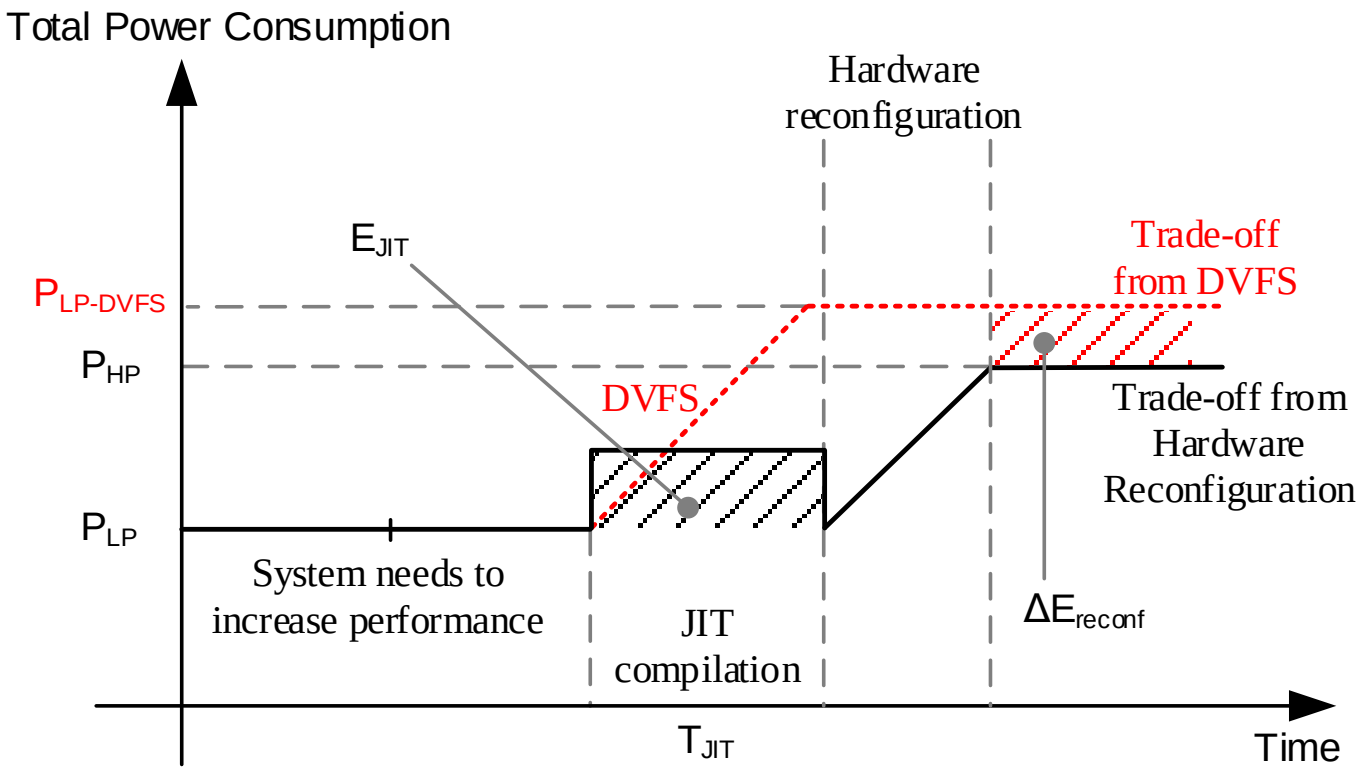
JIT Cost per Bytecode Instruction

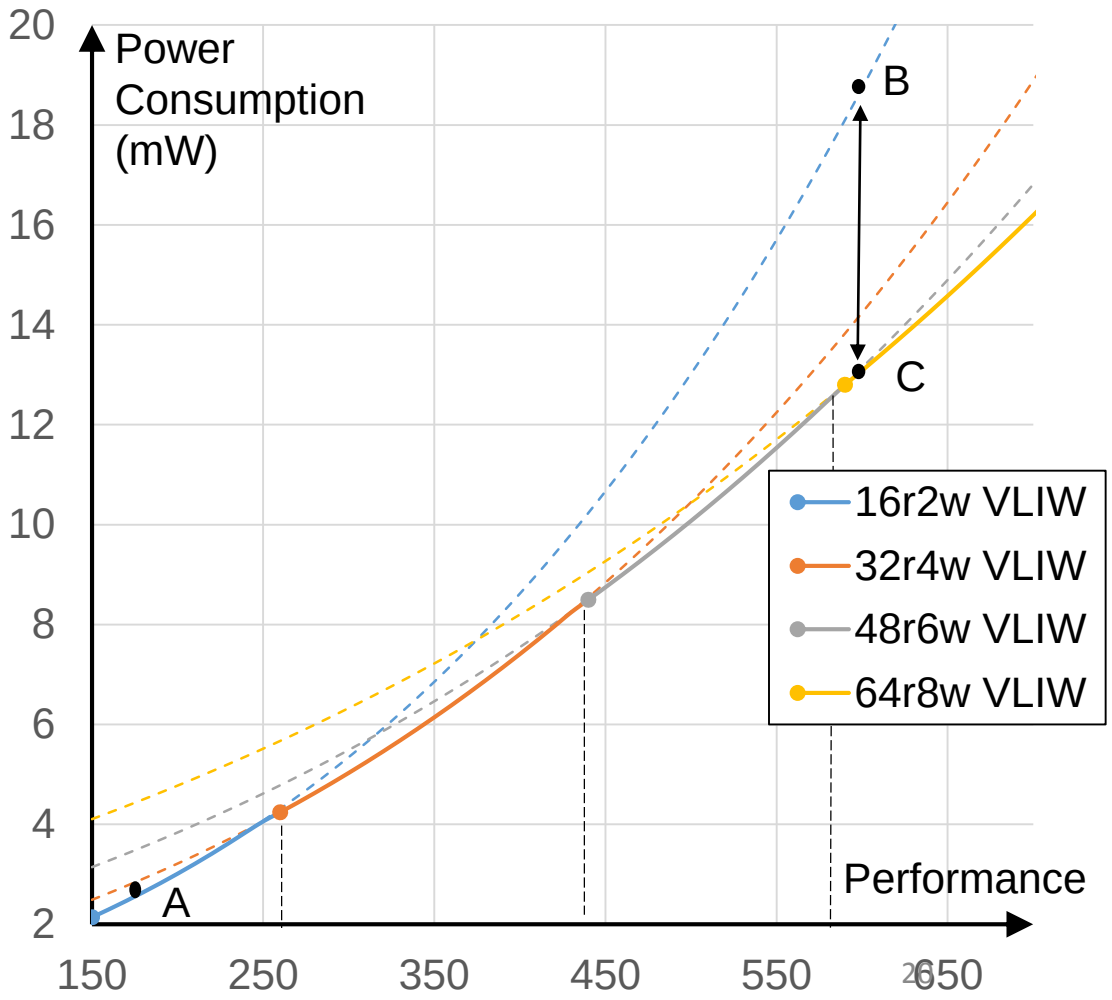


Energy Efficiency

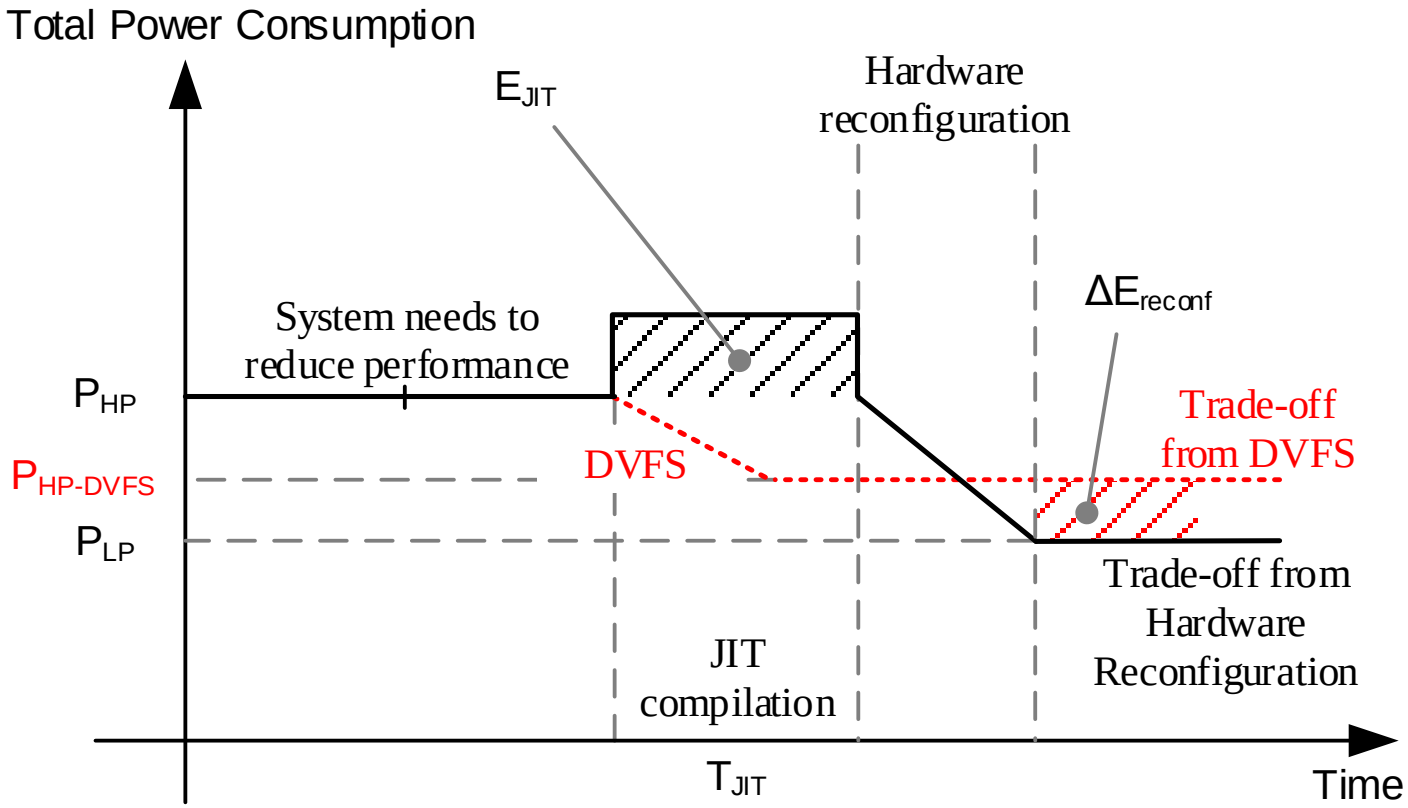


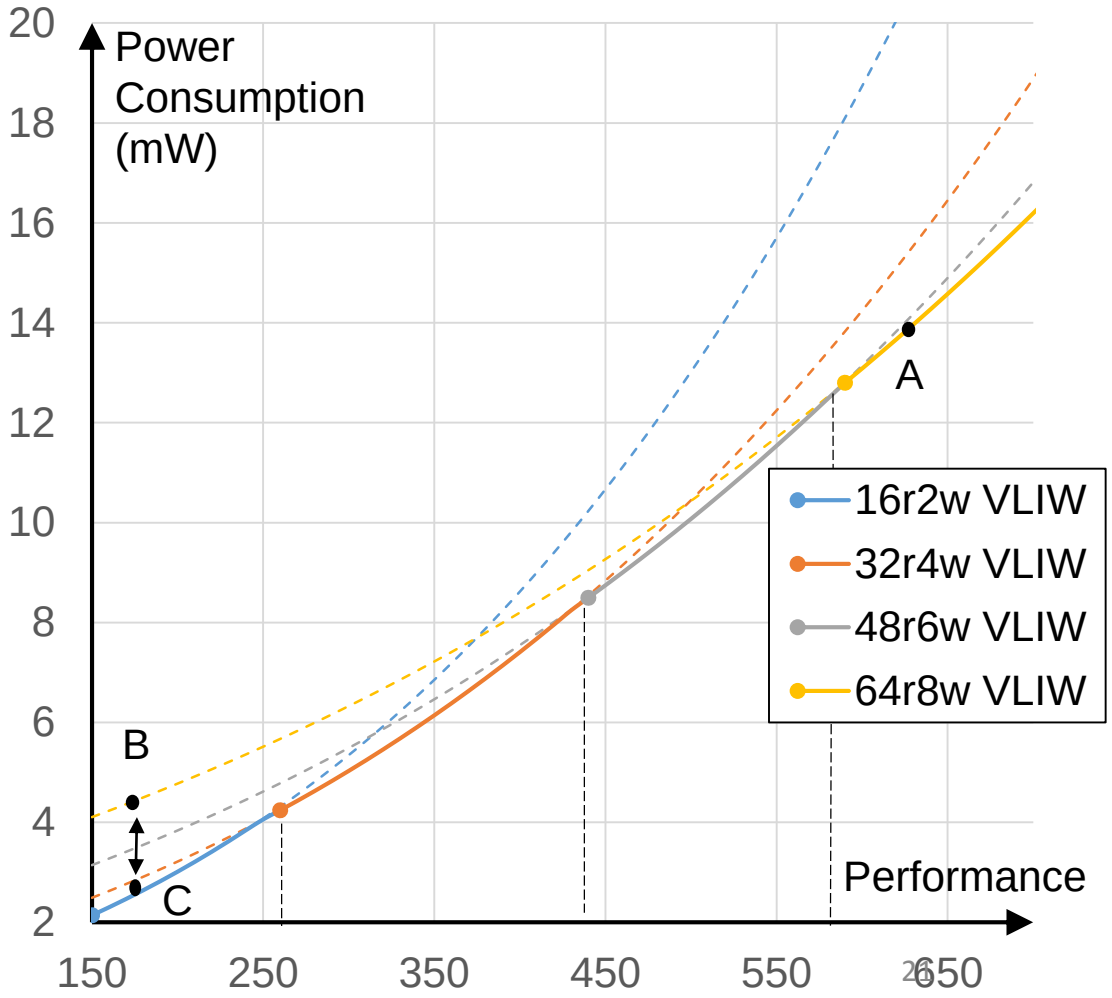
Hardware reconfiguration



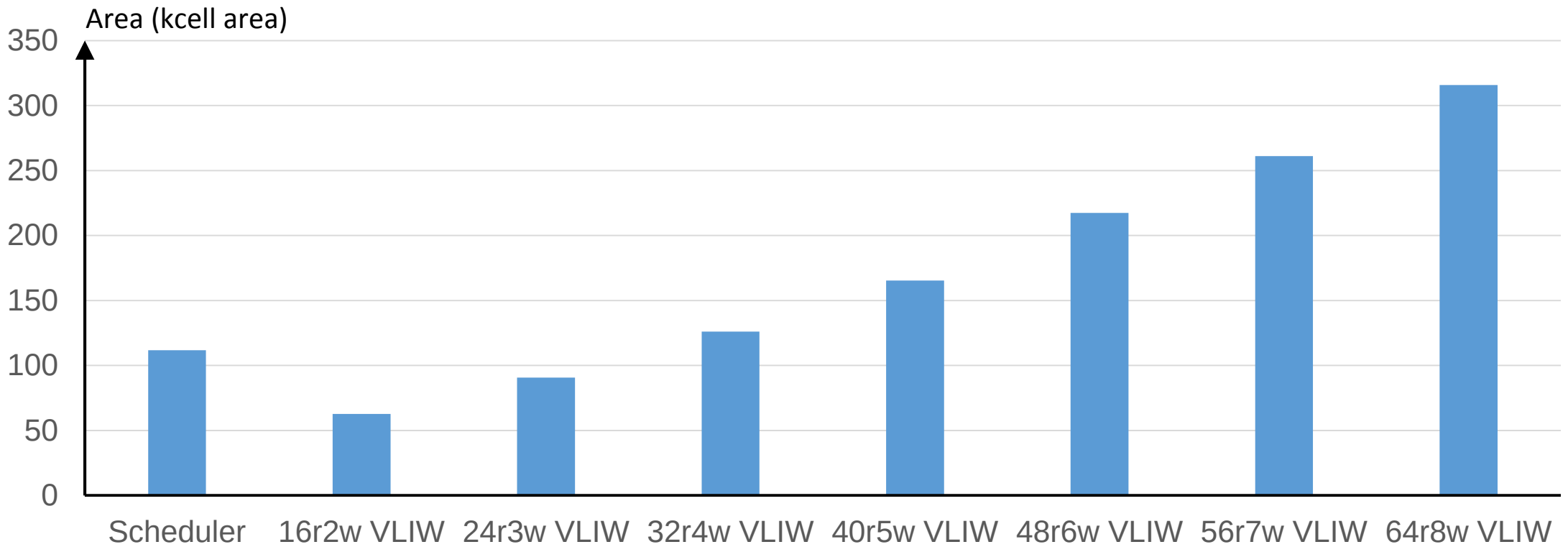


Hardware reconfiguration





Area


Area (kcell area)
350
300
250
200
150
100
50
0
Scheduler
16r2w VLIW
24r3w VLIW
32r4w VLIW
40r5w VLIW
48r6w VLIW
56r7w VLIW
64r8w VLIW

Outline

- Hybrid-JIT Compiler
- Experimental Study
- **Conclusion**

Conclusion

- Hybrid-JIT platform
  - Hardware scheduler
  - Custom bytecode
  - Important speed-up and energy improvement
  - Area overhead

- Future work
  - Register allocation
  - Modeling a manycore system based on VLIW
  - More complex architectures (CGRA)