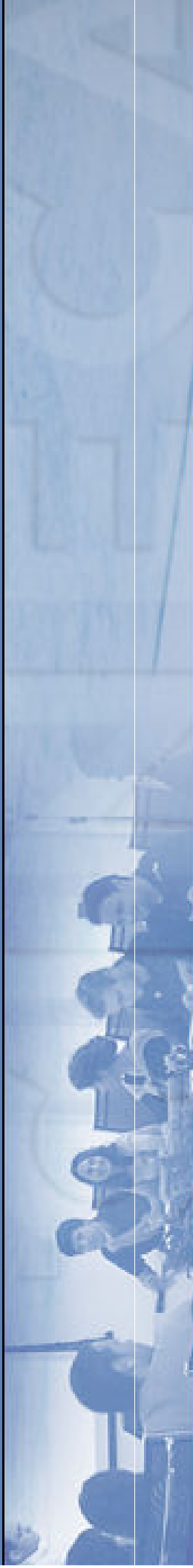


A) RISC processors

B) MIPS architecture

C) IBM POWER6 microarchitecture

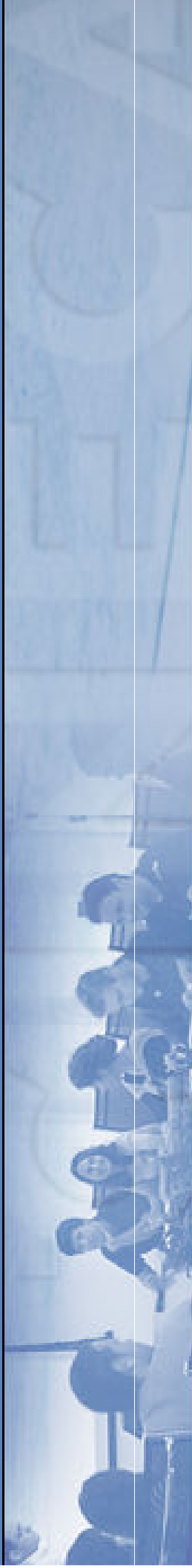


Ambra Giovannini (a.giovannini@cenea.it)

A) RISC processors

B) MIPS architecture

C) IBM POWER6 microarchitecture



Ambra Giovannini (a.giovannini@cineca.it)

Features

- ❏ Few and easy Assembly instructions, executed in the same amount of time
- ❏ Fixed-length instructions
- ❏ Many registers
- ❏ Efficient pipeline
- ❏ **lw** – **sw** → work on memory; other instructions on registers

- ✔ Pro: speed, power efficiency
- ✘ Cons: memory occupancy

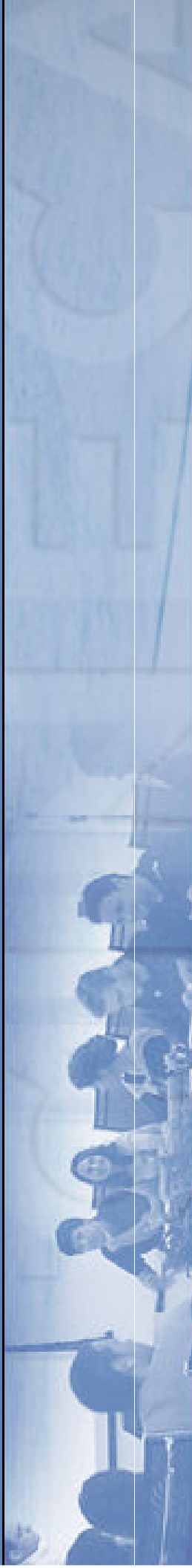
Products

- ❑ **MIPS:** PlayStation, PlayStation 2, PlayStation P., Nintendo 64
- ❑ **SPARC and UltraSPARC:** Sun Microsystem server
- ❑ **PA-RISC:** Hewlett-Packard server
- ❑ **ARM:** PDA such as Apple Newton, Game Boy Advance, Nintendo DS, iPod, Sony Ericsson and Nokia mobile
- ❑ **POWER:** IBM supercomputer, GameCube, Wii, Xbox 360, PlayStation 3

A) RISC processors

B) MIPS architecture

C) IBM POWER6 microarchitecture



Ambra Giovannini (a.giovannini@cineca.it)

Microprocessor without Interlocked Pipeline Stages

- ❑ First design released in 1985
- ❑ 32 / 64 bit
- ❑ Influenced later RISC architectures
- ❑ Primarily used in embedded systems such as Windows CE devices, routers, residential gateways, and video game consoles such as the Sony PlayStation 2 and PlayStation Portable

-31 -		format (bits)					-0-
R	opcode (6)	rs (5)	rt (5)	rd (5)	shamt (5)	funct (6)	
I	opcode (6)	rs (5)	rt (5)	immediate (16)			
J	opcode (6)	address (26)					

rs ⇒ First register source operand (R) .

rt ⇒ Second register source operand (R)

 Register destination operand (I).

rd ⇒ Register destination operand.

shamt ⇒ Shift amount.

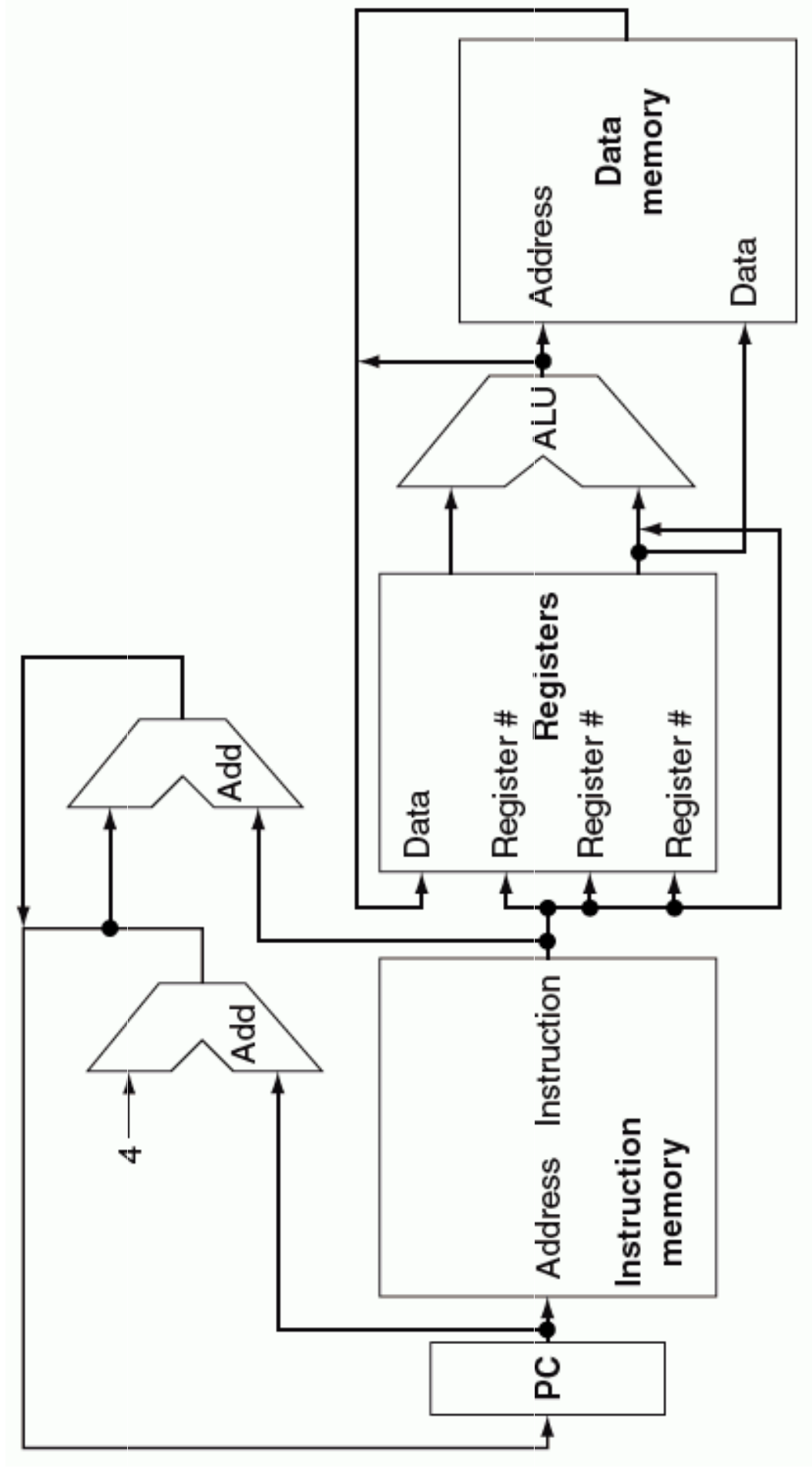
funct ⇒ Function.

Subset of the core MIPS instruction set

- Memory-reference instructions load word (lw) and store word (sw)
- Arithmetic-logical instructions (add , sub , and , or , $sllt$)
- Instructions branch equal (beq) and jump (j)

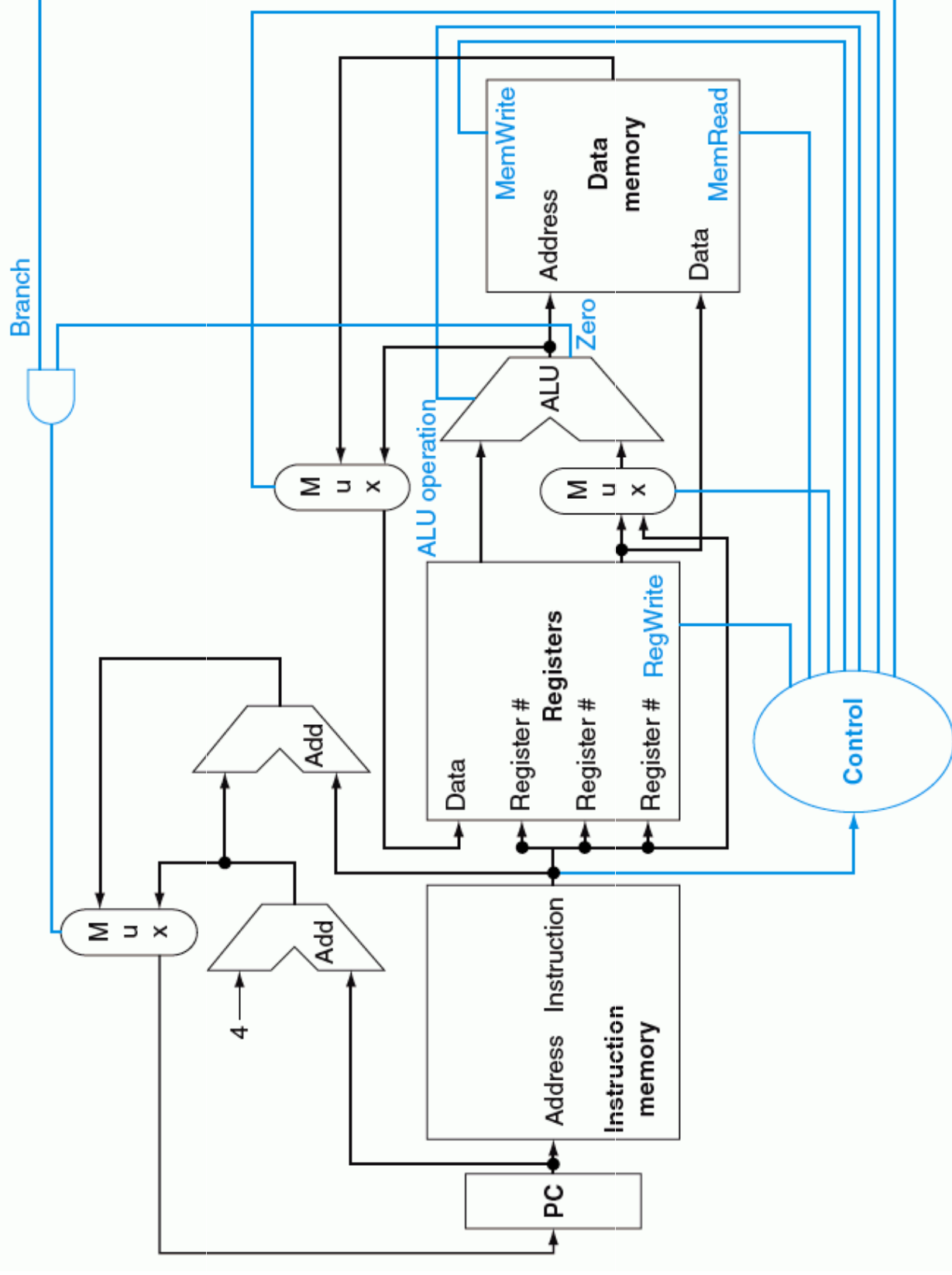
MIPS

Abstract view (1/2)

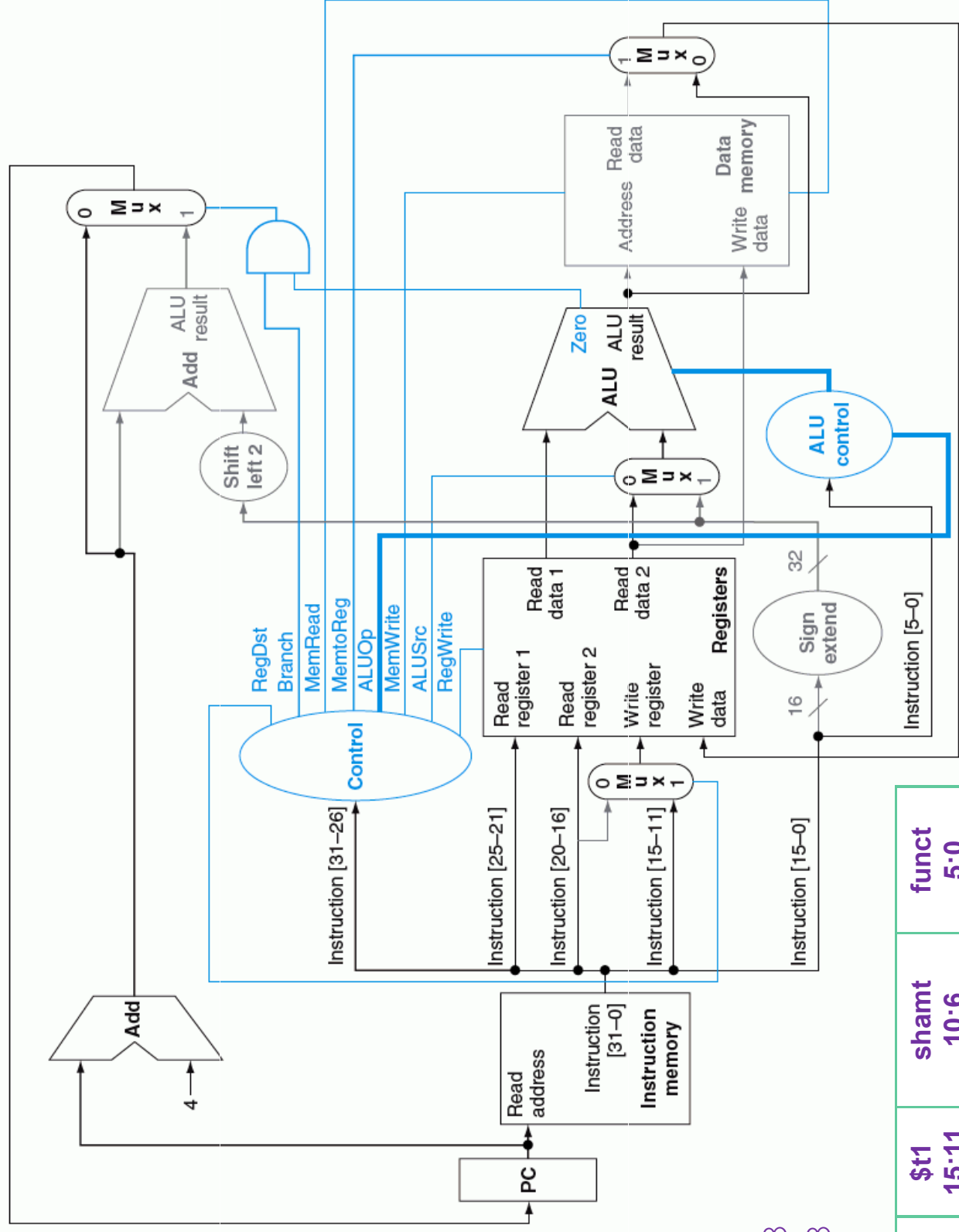


MIPS

Abstract view (2/2)



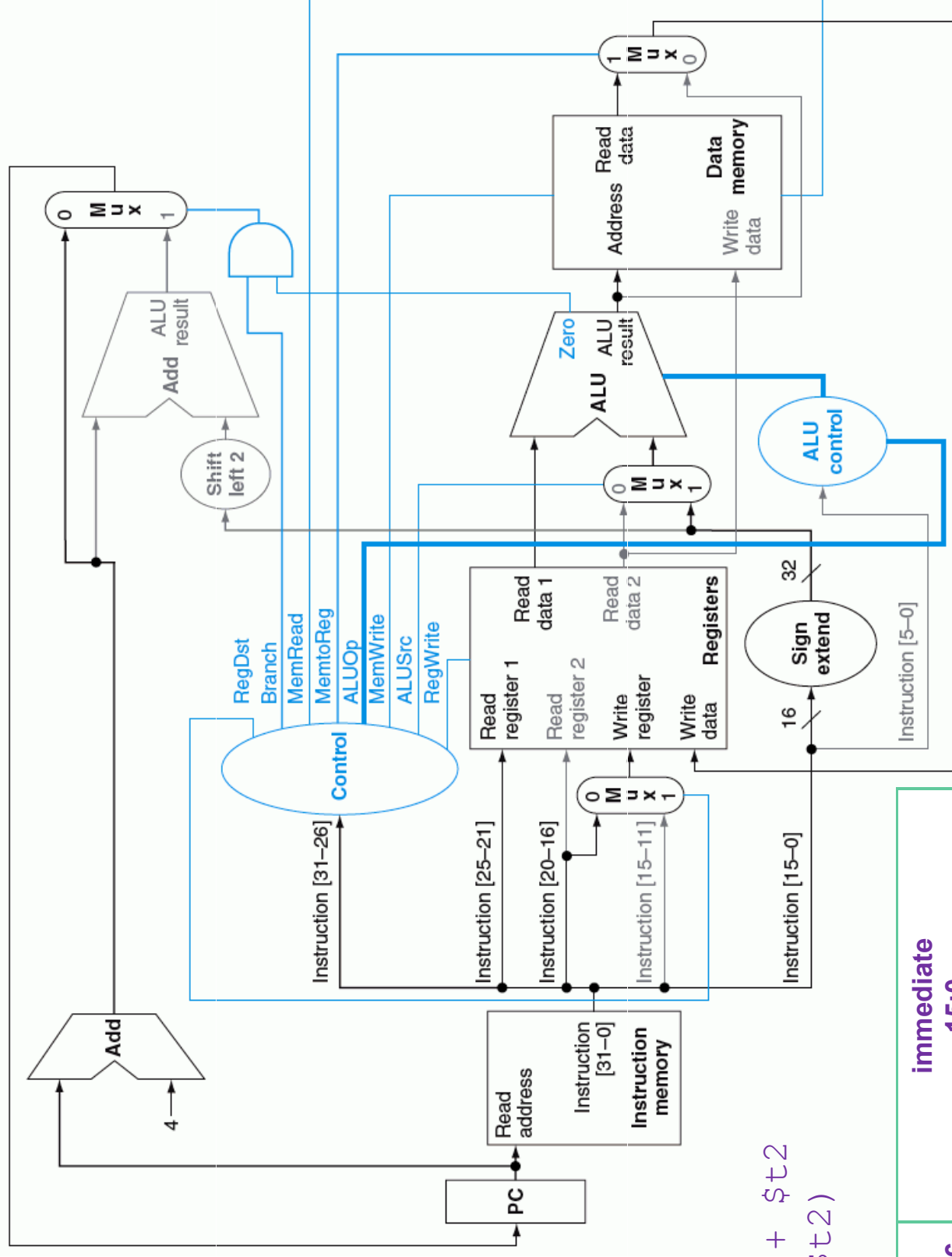
MIPS Datapath for an R-type instruction



\$t1 = \$t2 + \$t3
add \$t1, \$t2, \$t3

opcode	\$t2	\$t3	\$t1	shamt	funct
31:26	25:21	20:16	15:11	10:6	5:0

MIPS Datapath for an I-type instruction



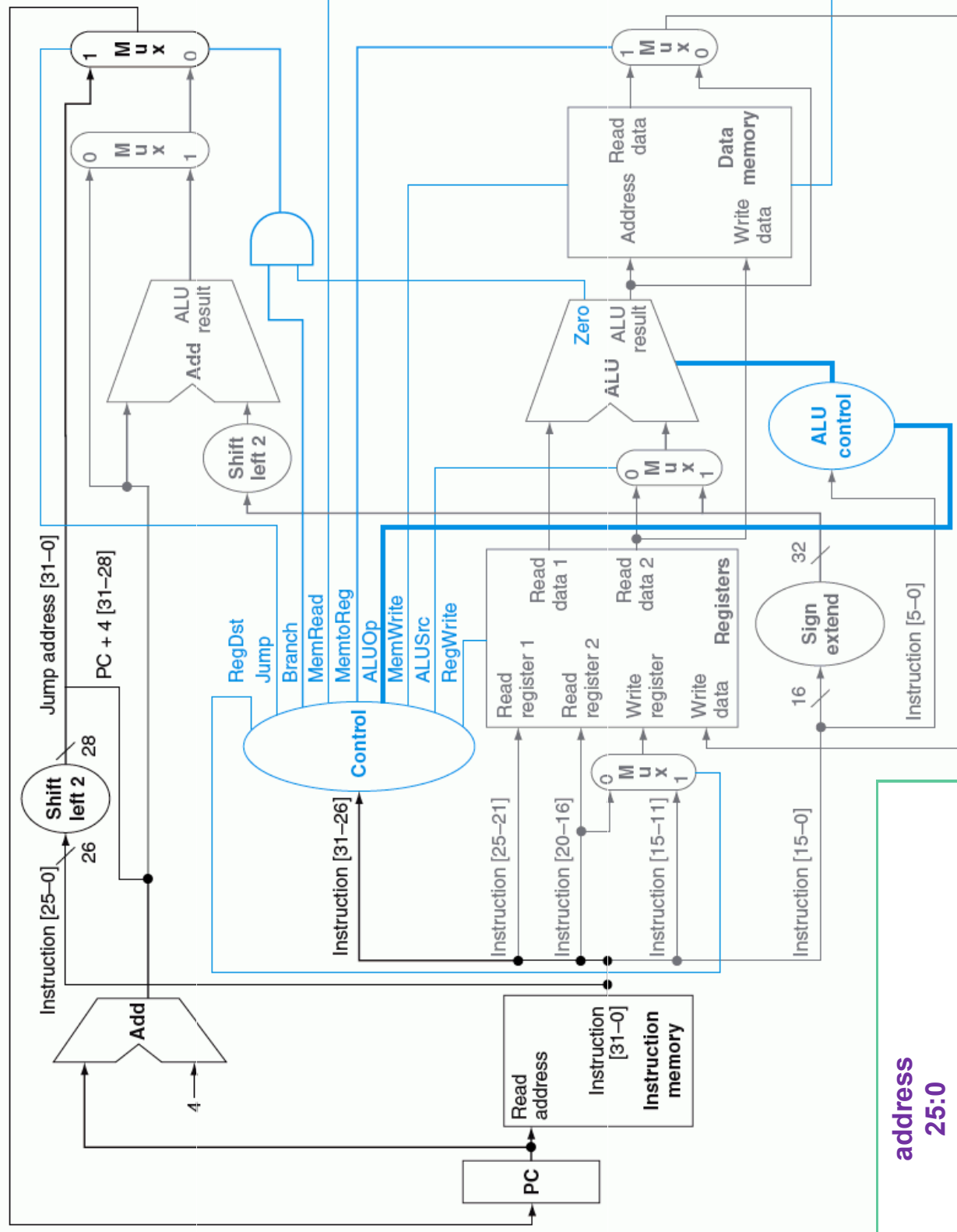
\$t1 <- offset + \$t2
lw \$t1, offset(\$t2)

opcode 31:26	\$t2 25:21	\$t1 20:16	immediate 15:0
-----------------	---------------	---------------	-------------------

MIPS

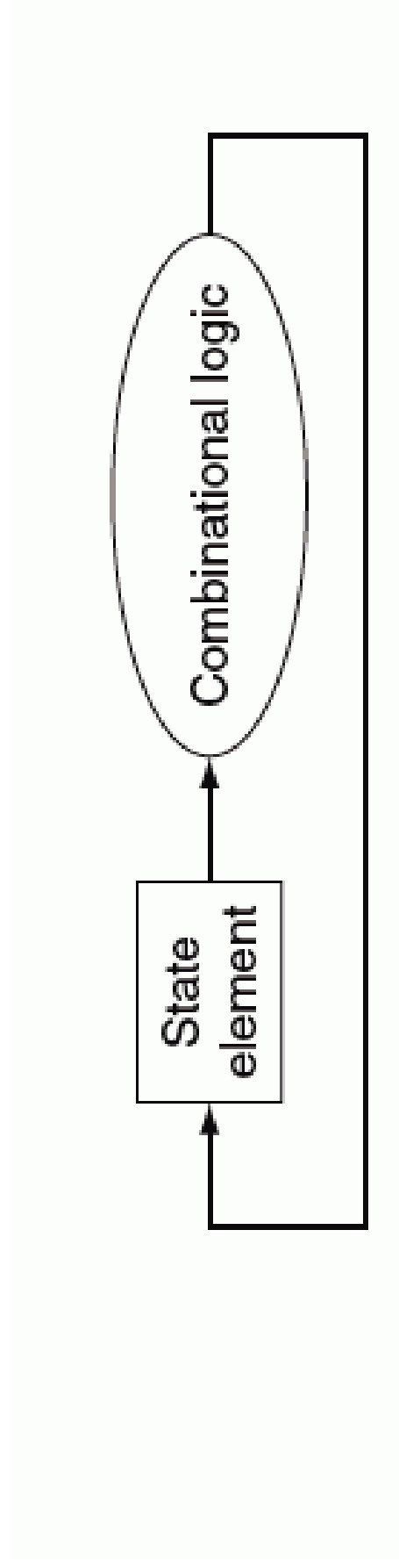
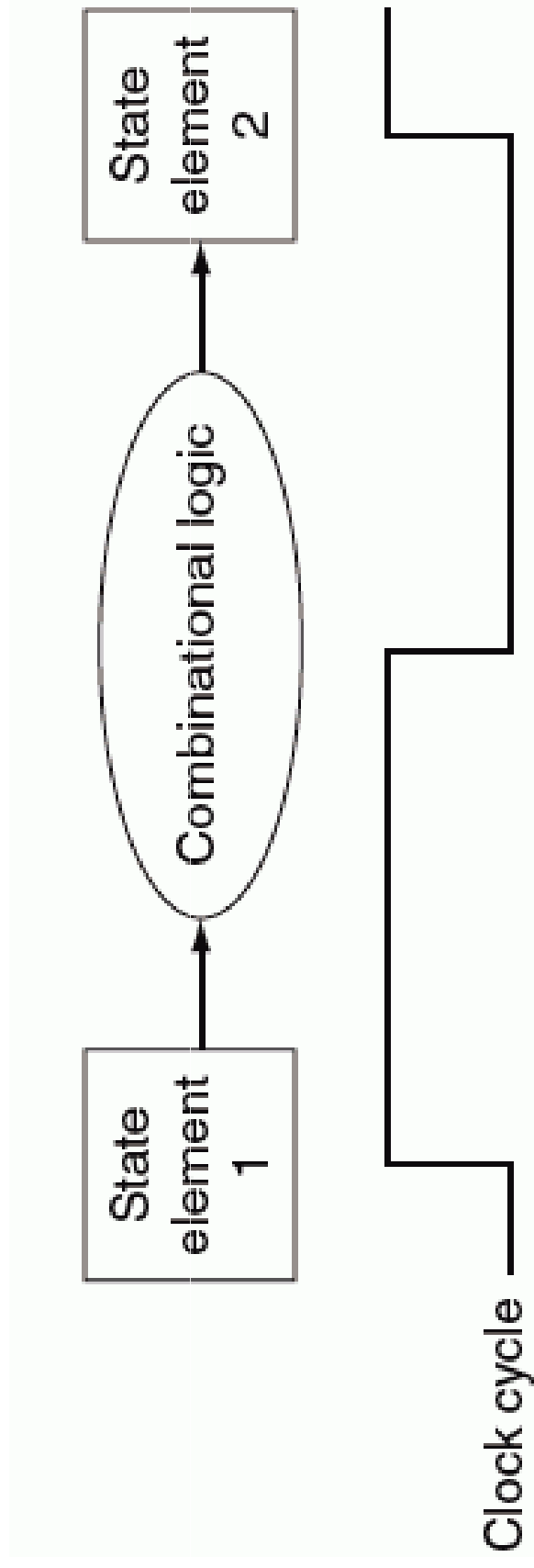
Datapath for a J-type instruction

go to addr
j addr



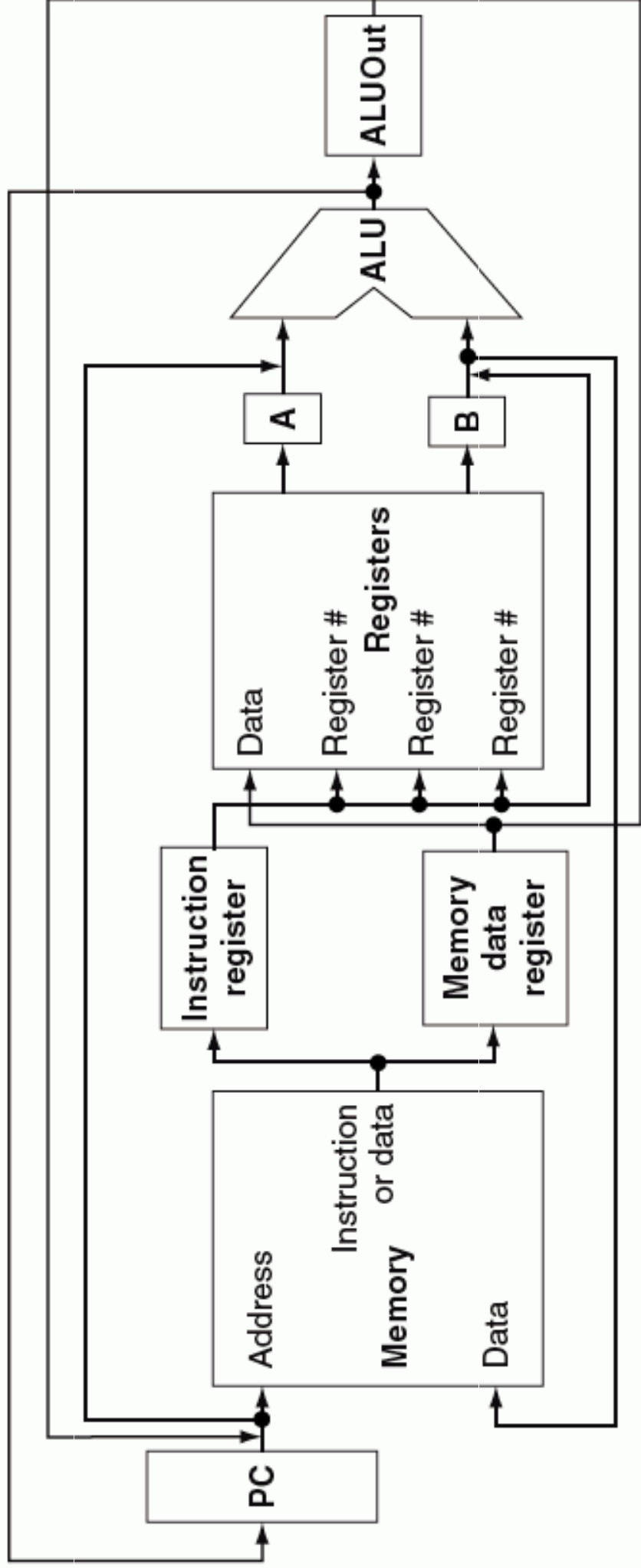
opcode
31:26

address
25:0

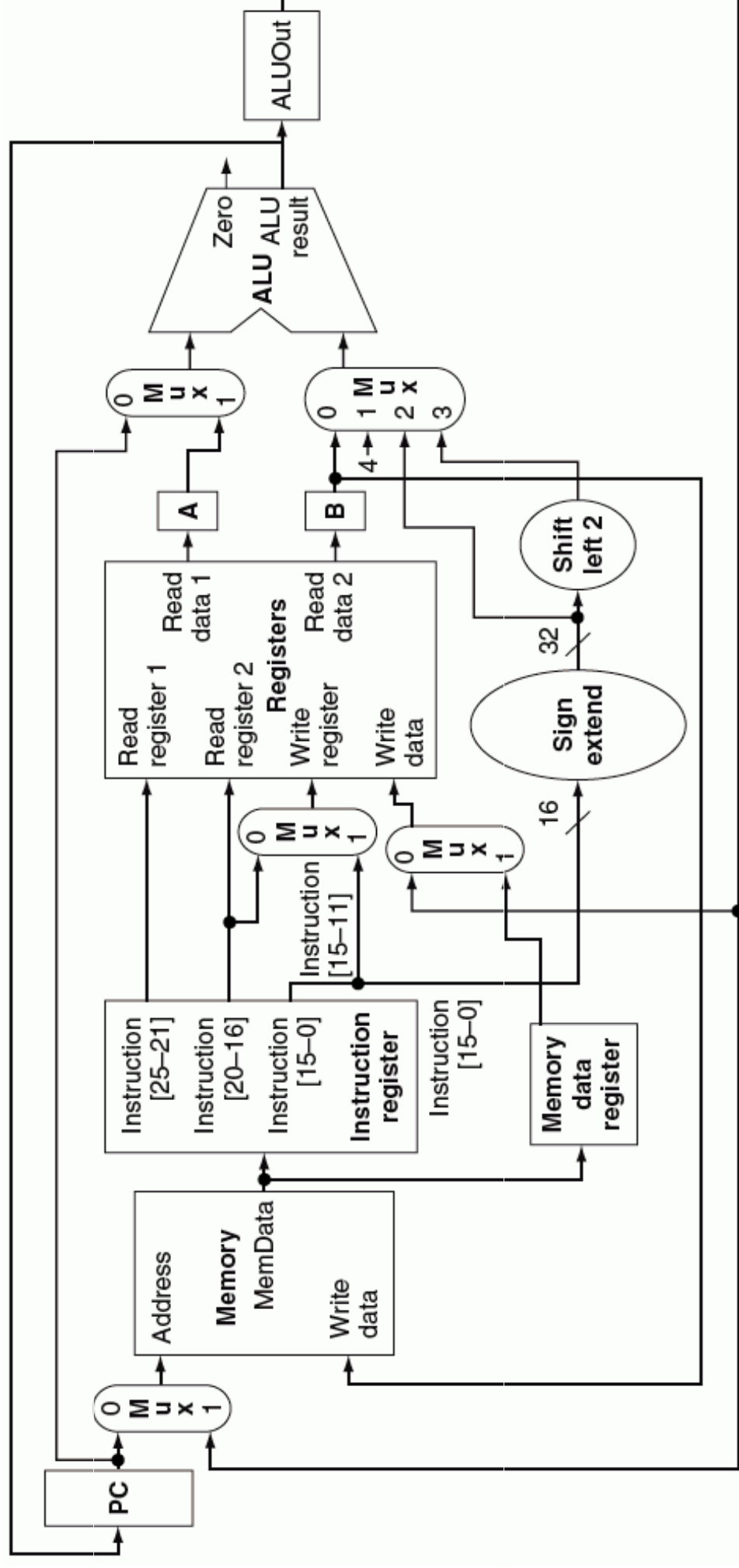


MIPS

High-level view of the multicycle



MIPS Multicycle datapath



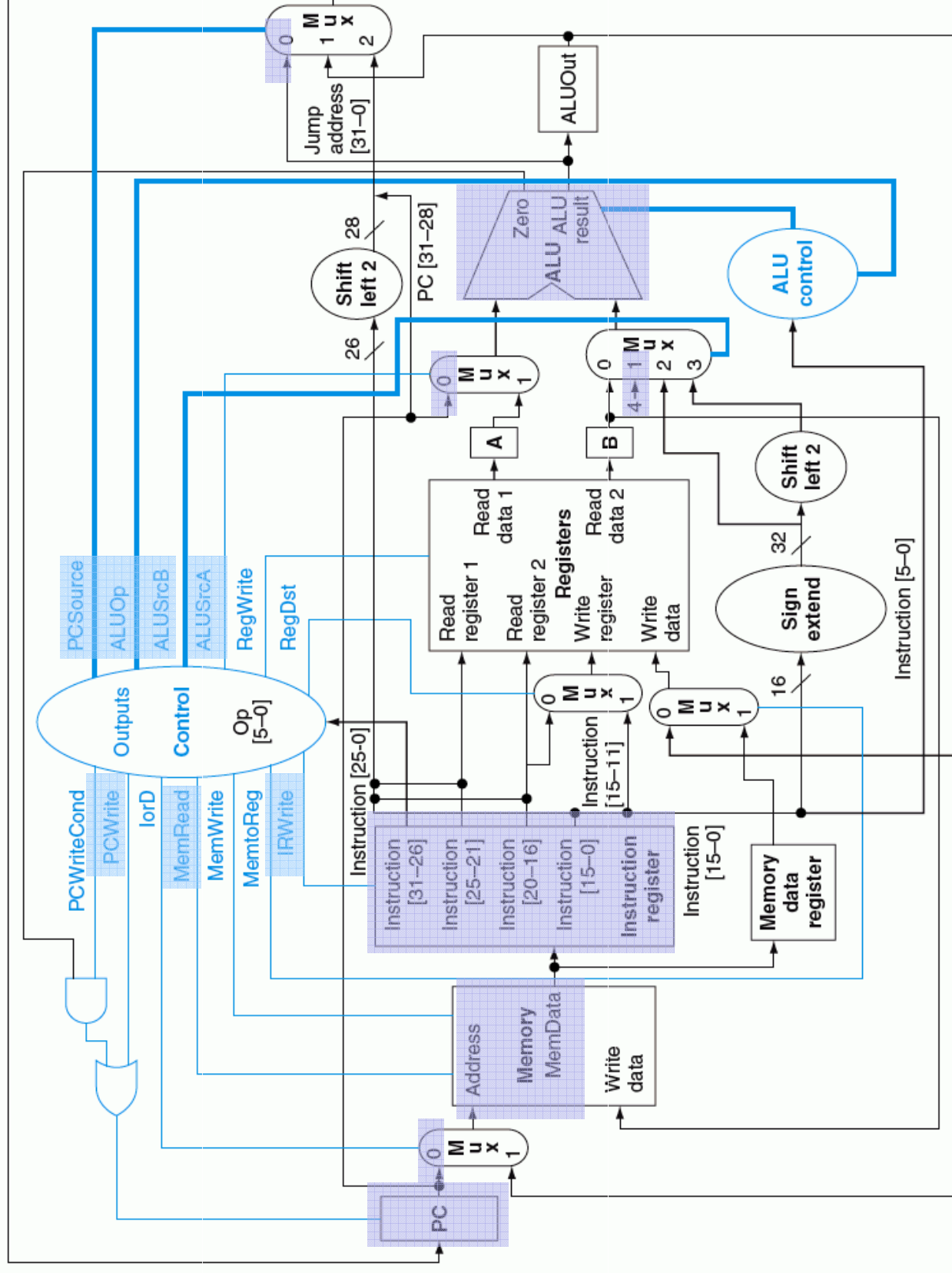


MIPS

Multicycle execution

Step name	Action for R-type instructions	Action for memory-reference instructions	Action for branches	Action for jumps
Instruction fetch		IR <= Memory[PC] PC <= PC + 4		
Instruction decode/register fetch		A <= Reg [IR[25:21]] B <= Reg [IR[20:16]] ALUOut <= PC + (sign-extend (IR[15:0]) << 2)		
Execution, address computation, branch/jump completion	ALUOut <= A op B	ALUOut <= A + sign-extend (IR[15:0])	if (A == B) PC <= ALUOut	PC <= {PC [31:28], (IR[25:0]), 2'b000}
Memory access or R-type completion	Reg [IR[15:11]] <= ALUOut	Load: MDR <= Memory[ALUOut] or Store: Memory [ALUOut] <= B		
Memory read completion		Load: Reg[IR[20:16]] <= MDR		

Fetch
Decode
Execute
Memory
Write back

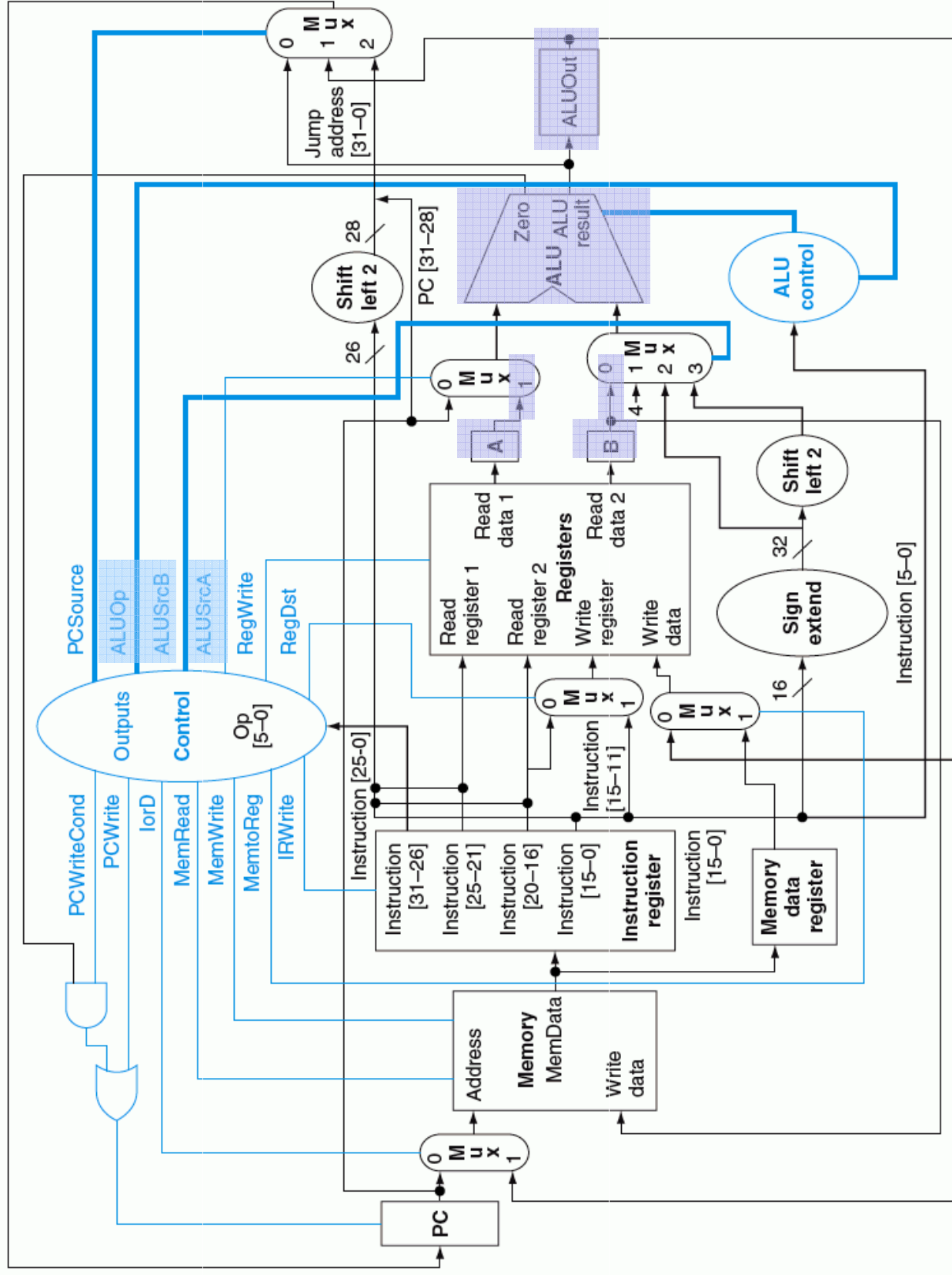


www.cineca.it



MIPS

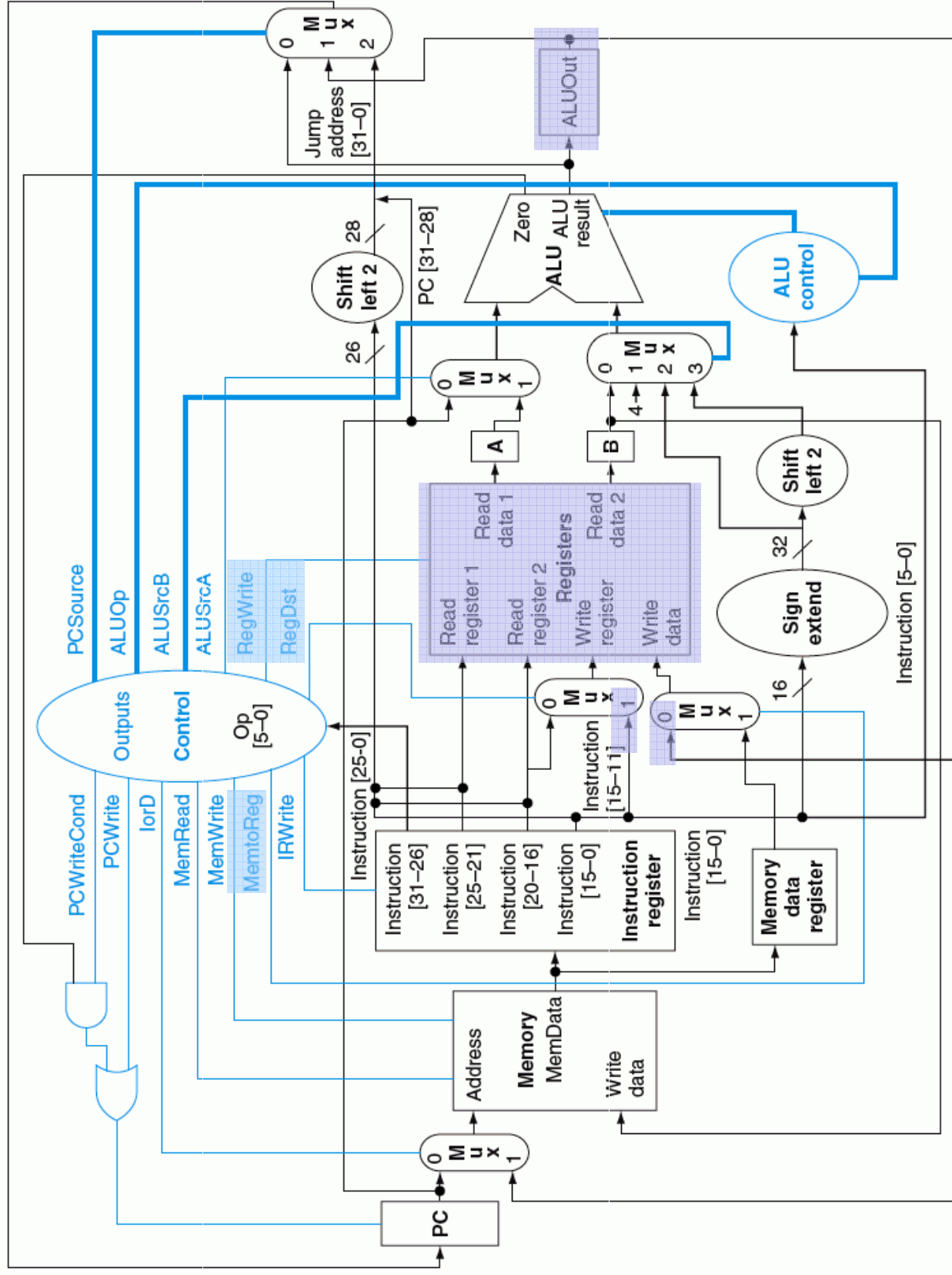
Multicycle datapath R-type instr. (3/4)



Fetch
Decode
Execute
Memory
Write back

MIPS

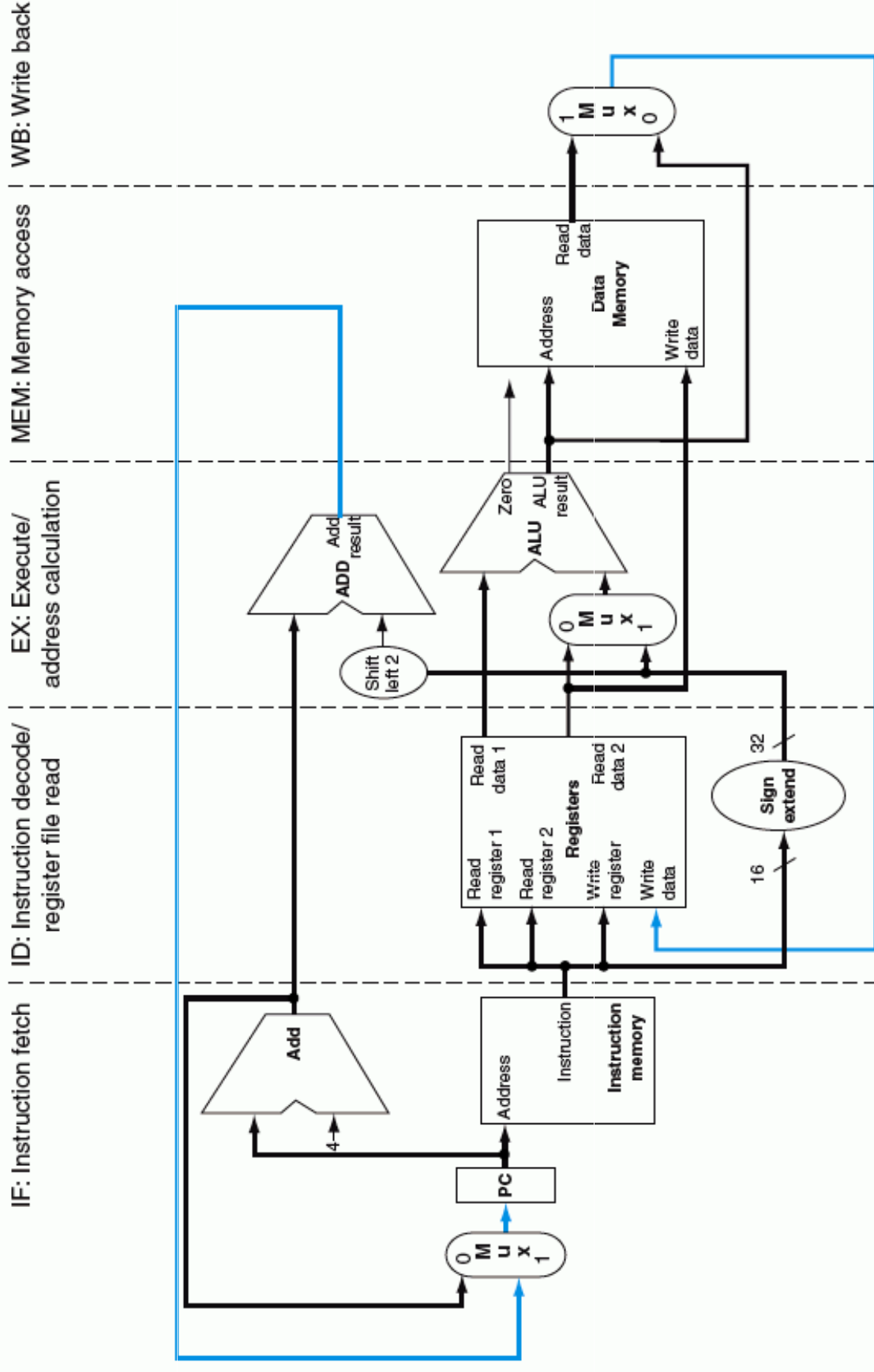
Multicycle datapath R-type instr. (4/4)



Fetch
Decode
Execute
Memory
Write back

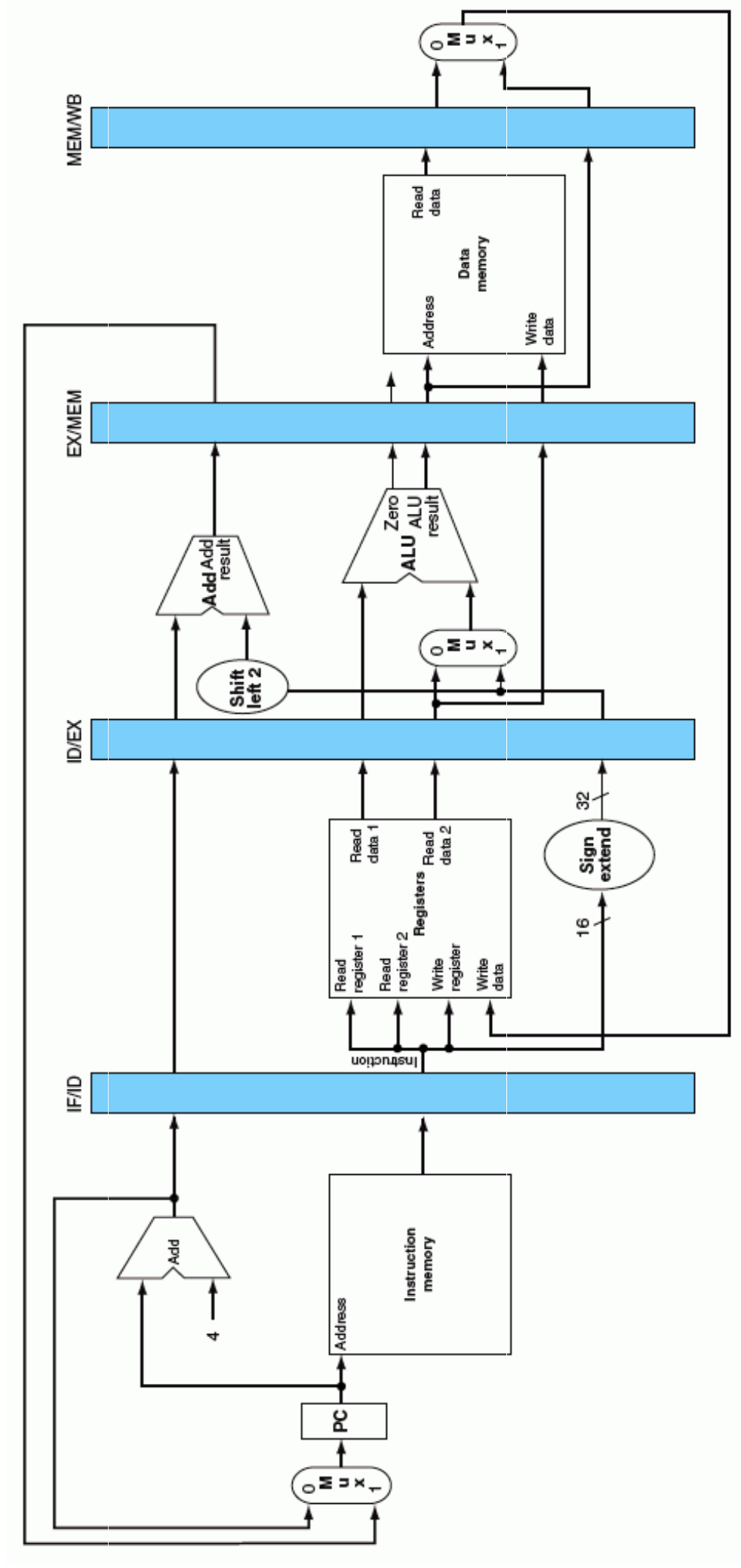
MIPS

Singlecycle datapath – Pipeline (1/2)



MIPS

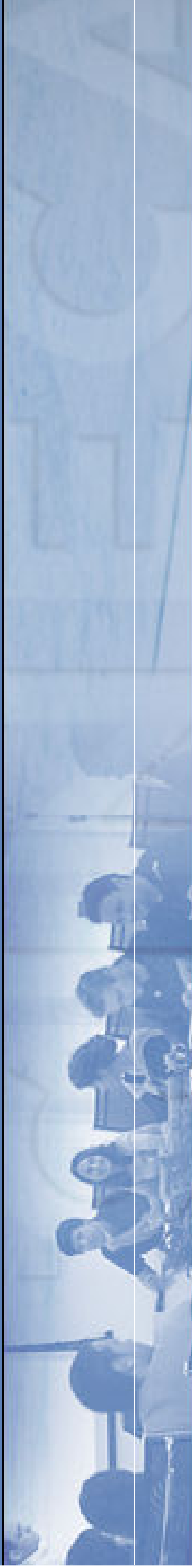
Singlecycle datapath – Pipeline (2/2)



A) RISC processors

B) MIPS architecture

C) IBM POWER6 microarchitecture



Ambra Giovannini (a.giovannini@cineca.it)

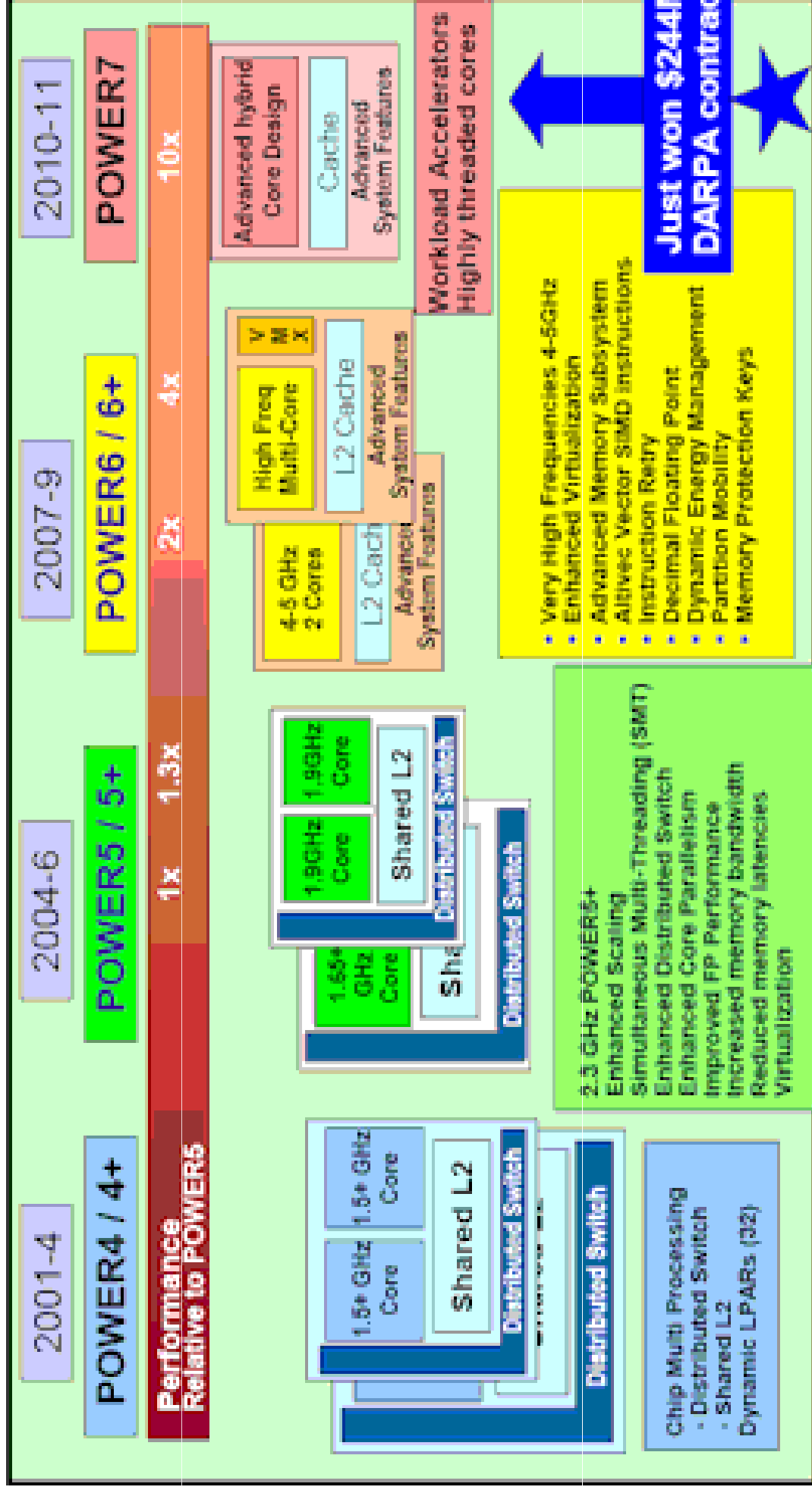
POWER*

Processors history

1990	POWER1	32 bit 1FPU multiply/add – 1FXU 8K D-cache 32K I-cache 25MHz
1993	POWER2	2FPU multiply/add - 2FXU 256K D-cache
1998	POWER3	Istr. 64 bit 2FPU – 2FXU – 2LS L1 I. 64Kb (s.a.), D. 32Kb (s.a.), L2 8Mb (d.m.) 64bit 375MHz
2001	POWER4	Dual CORE 2FPU – 2FXU – 2LS – 1BR L1 I. 64Kb (d.m.), D. 32Kb (d.m), L2 1.44Mb shared, L3 128Mbyte 1.3GHz
2005	POWER5	Dual CORE - Simultaneous multithreading (SMT) -> 4 logical procs. L1 32+32Kb/core, L2 1.875 MB/chip, L3 36Mb/chip 2.3GHz

* Performance Optimization With Enhanced RISC

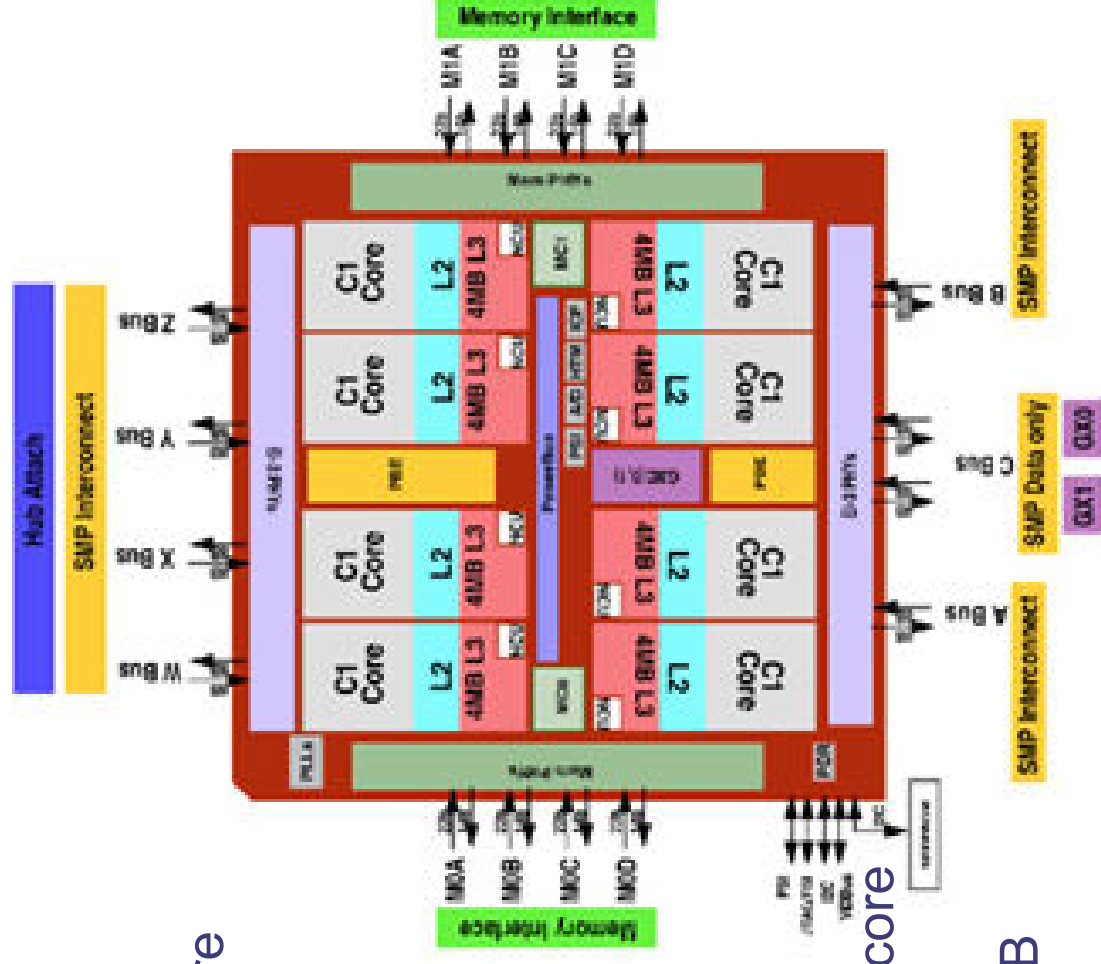
POWER Processor roadmap



BINARY COMPATIBILITY

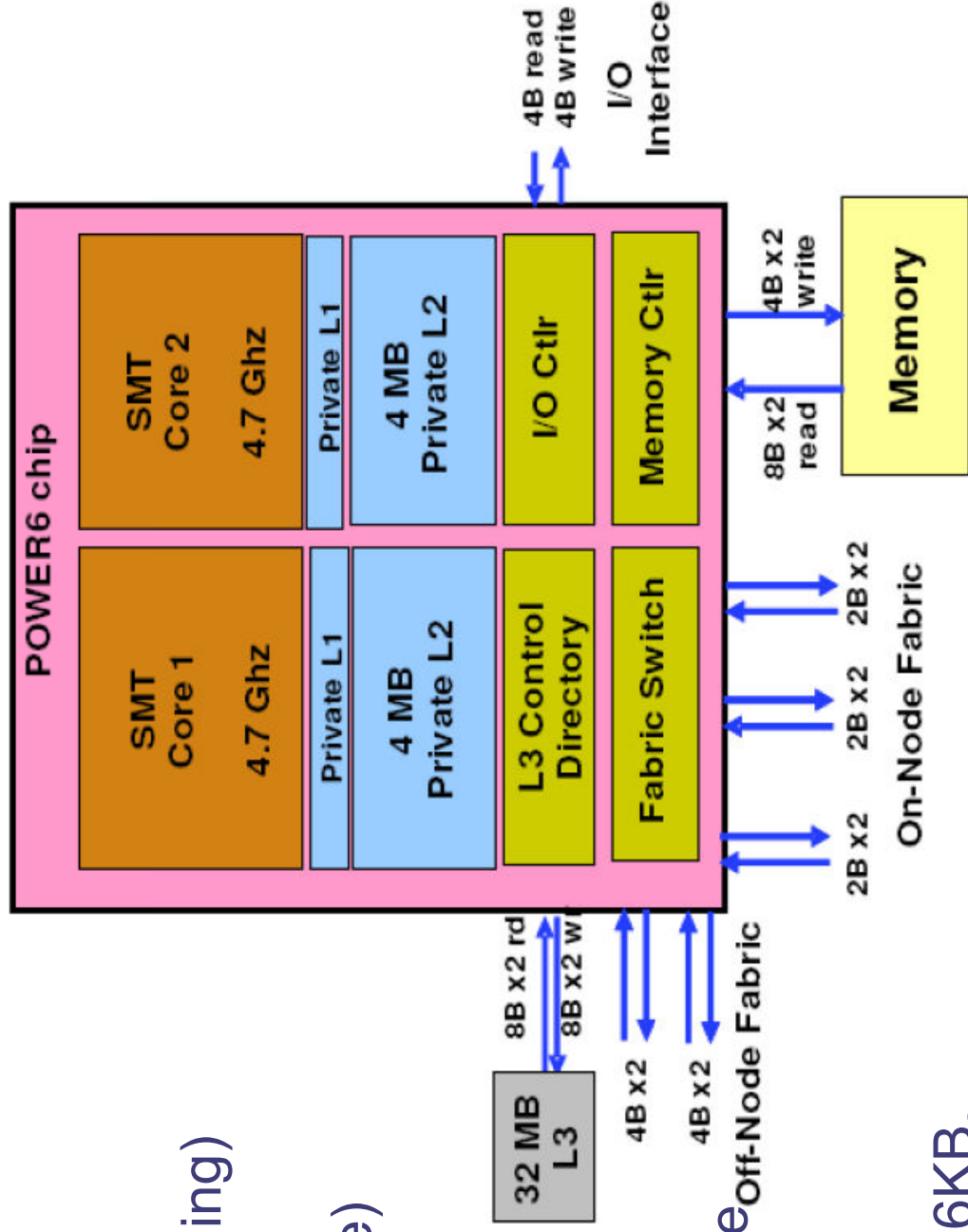
POWER7 Schematic architecture


- ❏ 2010
- ❏ 8 core
- ❏ 4 SMT (simultaneous multithreading) per core
- ❏ 12 execution units per core
 - ❏ 2 fixed-point units
 - ❏ 2 load/store units
 - ❏ 4 double-precision fp units
 - ❏ 1 vector unit supporting
 - ❏ 1 decimal floating-point unit
 - ❏ 1 branch unit
 - ❏ 1 condition register unit
- ❏ 32+32 kB L1 instruction and data cache per core
- ❏ 256 kB L2 Cache (per core)
- ❏ 4 MB L3 cache per core with max up to 32MB



POWER6 Schematic architecture

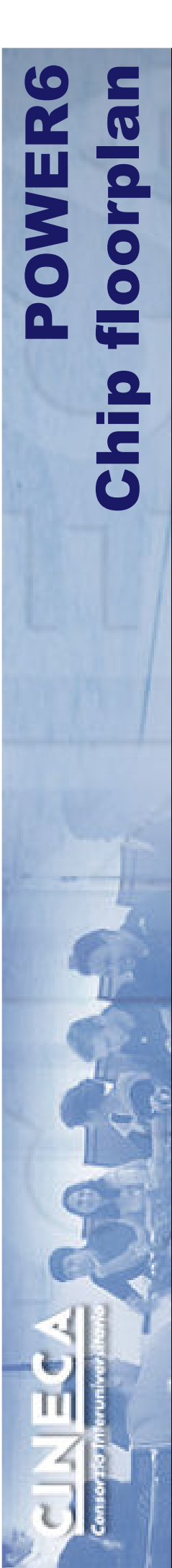
- 2007
- 64 bit
- Dual core SMT
(Simultaneous multithreading)
- 4.7 GHz (vs 1.9GHz P5)
- SMT speed-up (cache size)
- Up to 7 instructions simult.
for both threads
- 1 cycle b2b FX(I+X)
- 6 cycles FPU
- 4 Floating Point Ops / cycle
- Memory page size: 4KB, 16KB,
16MB, 16GB

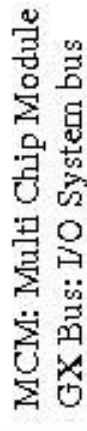


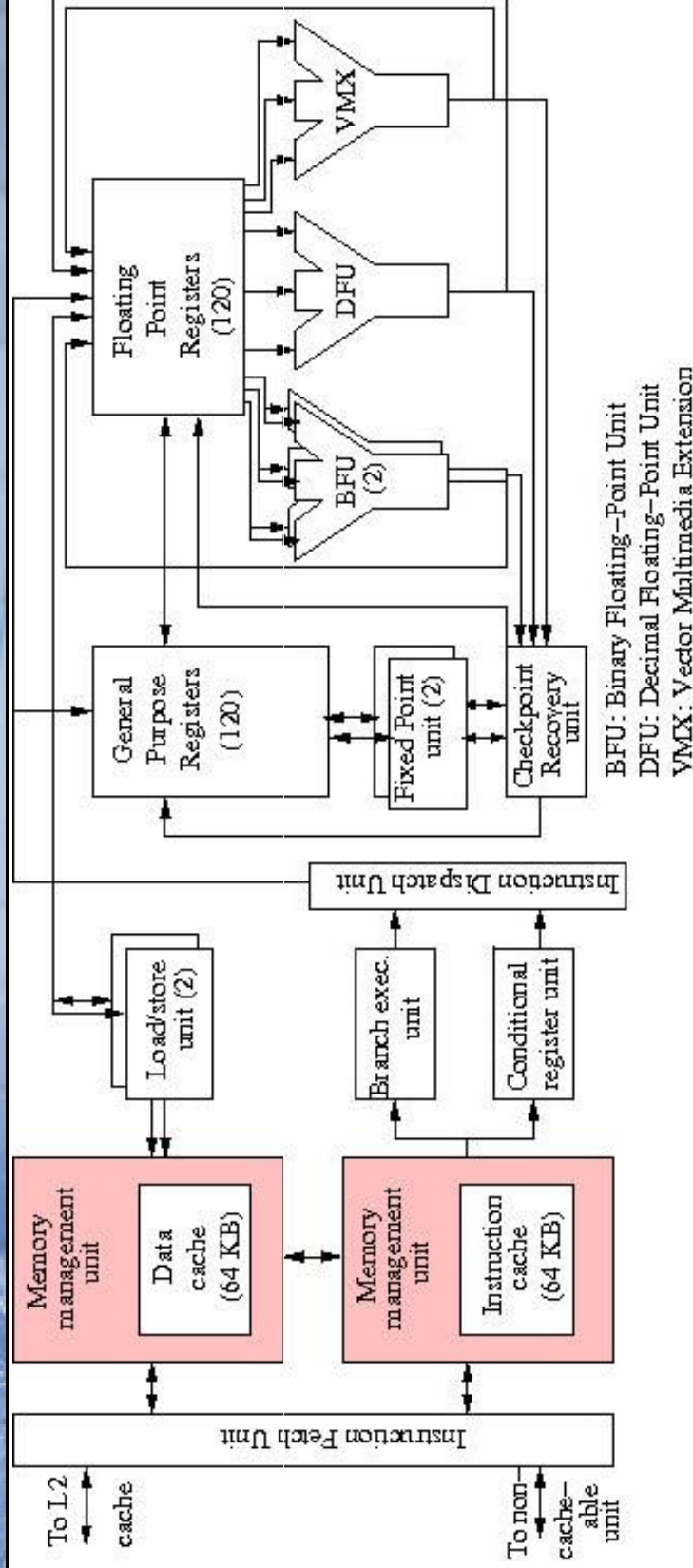


CINECA
Consorzio Interuniversitario

POWER6 Chip floorplan







1 Decimal FPU

1 Branch Resolution Unit (BRU)

1 VMX Unit

- 4 FMAs /
SIMD (32-bit)

2 Floating Point Units (FPU)

- most flops with 6-7 cycles latency
- fused multiply-adds (FMA)
- hardware divisions and sqrts not pipelined, but do not stall core
- pipelined 14 bit accurate single-cycle reciprocals and inverse sqrts
- out-of-order execution

2 Fixed Point Units (FXU)

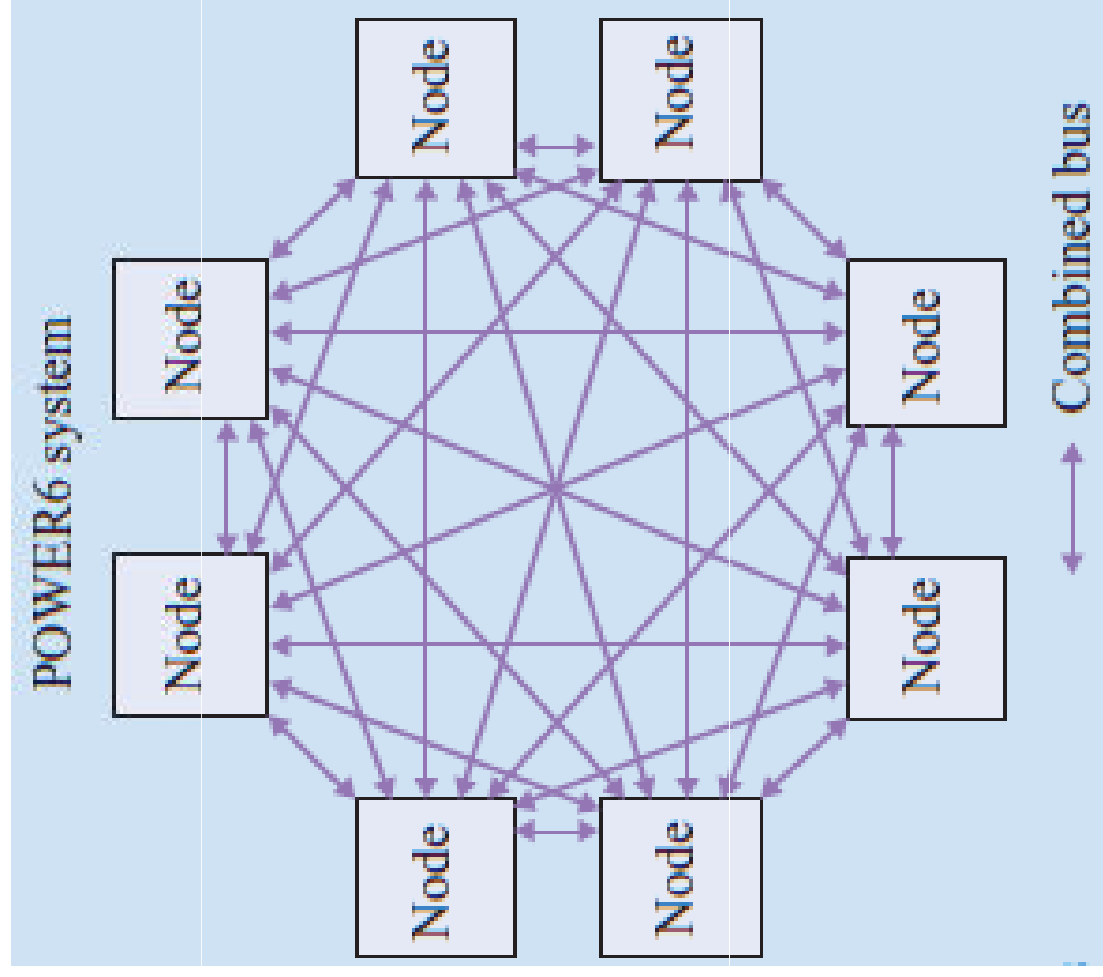
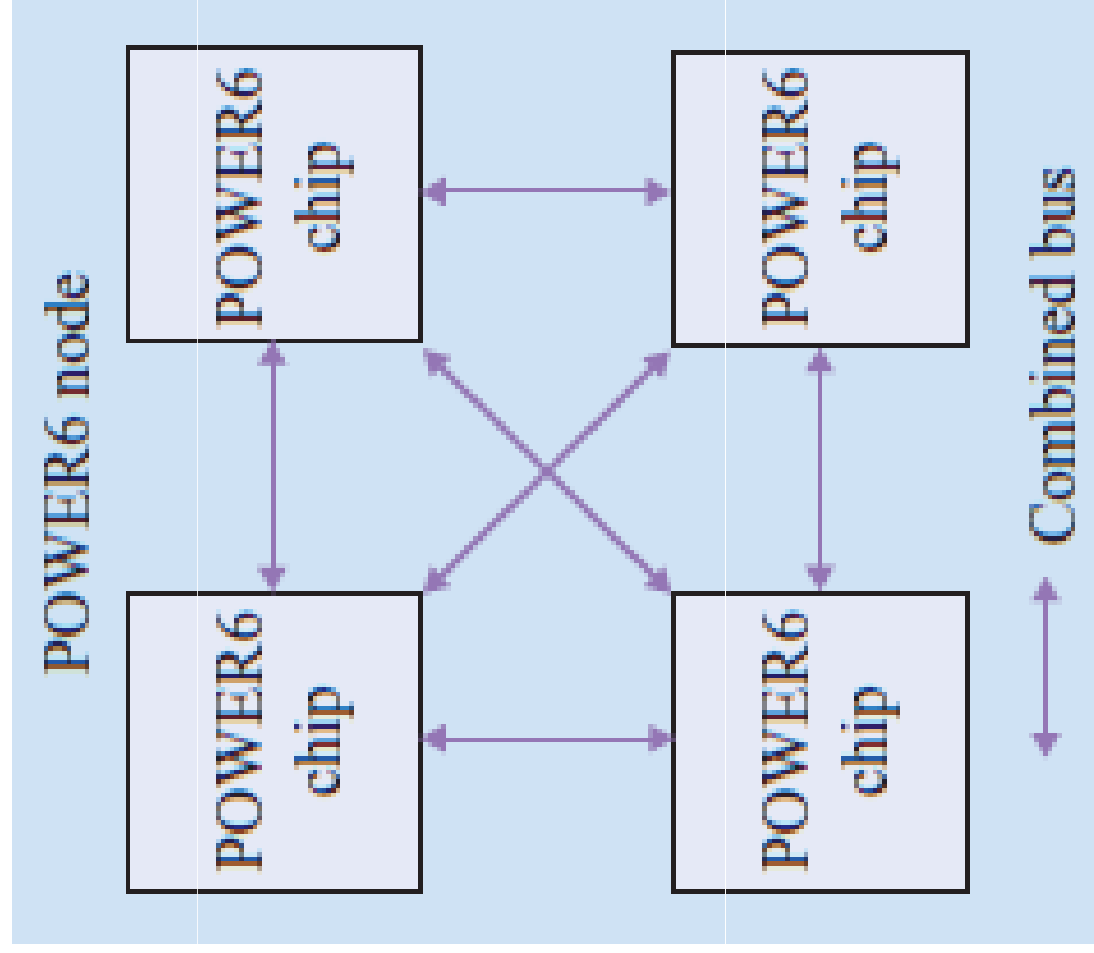
- integer ops with 2-4 cycles latency
 - integer multiplies & divides delegated to FPU (higher latency)

2 Load/Store Units (LSU)

- 2 64 bit loads / cycle
- 2 64 bit stores / 2 cycles

<i>Cache attribute</i>	<i>L1 instruction</i>	<i>L1 data</i>	<i>L2</i>	<i>L3</i>
Capacity	64 KB	64 KB	4 MB	32 MB
Shares (cores)	1	1	1	2
Location	Within core	Within core	On-chip	Off-chip
Line size (bytes)	128	128	128	128
Associativity	4 way	8 way	8 way	16 way
Update policy	Read only	Store through	Store in	Victim
Line inclusion rules	Resides in L2	Resides in L2	None	None
Snooped	No	No	Yes	Yes
Error protection	Parity	Parity	ECC	ECC

POWER6 Topology



- Model: IBM pSeries 575
- Architecture: IBM P6-575 Infiniband Cluster
- Processor Type: IBM Power6, 4.7 GHz
- Computing Cores: 5376
- Computing Nodes: 168
- RAM: 21 TB (128 GB/node)
- Internal Network: Infiniband x4 DDR
- Disk Space: 1.2 PB
- Operating System: AIX 6
- Peak Performance: 101 TFlop/s
- Available compilers: Fortran90, C, C++
- Parallel libraries: MPI, OpenMP, LAPACK



POWER6 p575 node architecture

How many?

 2 login node

1 visualization node

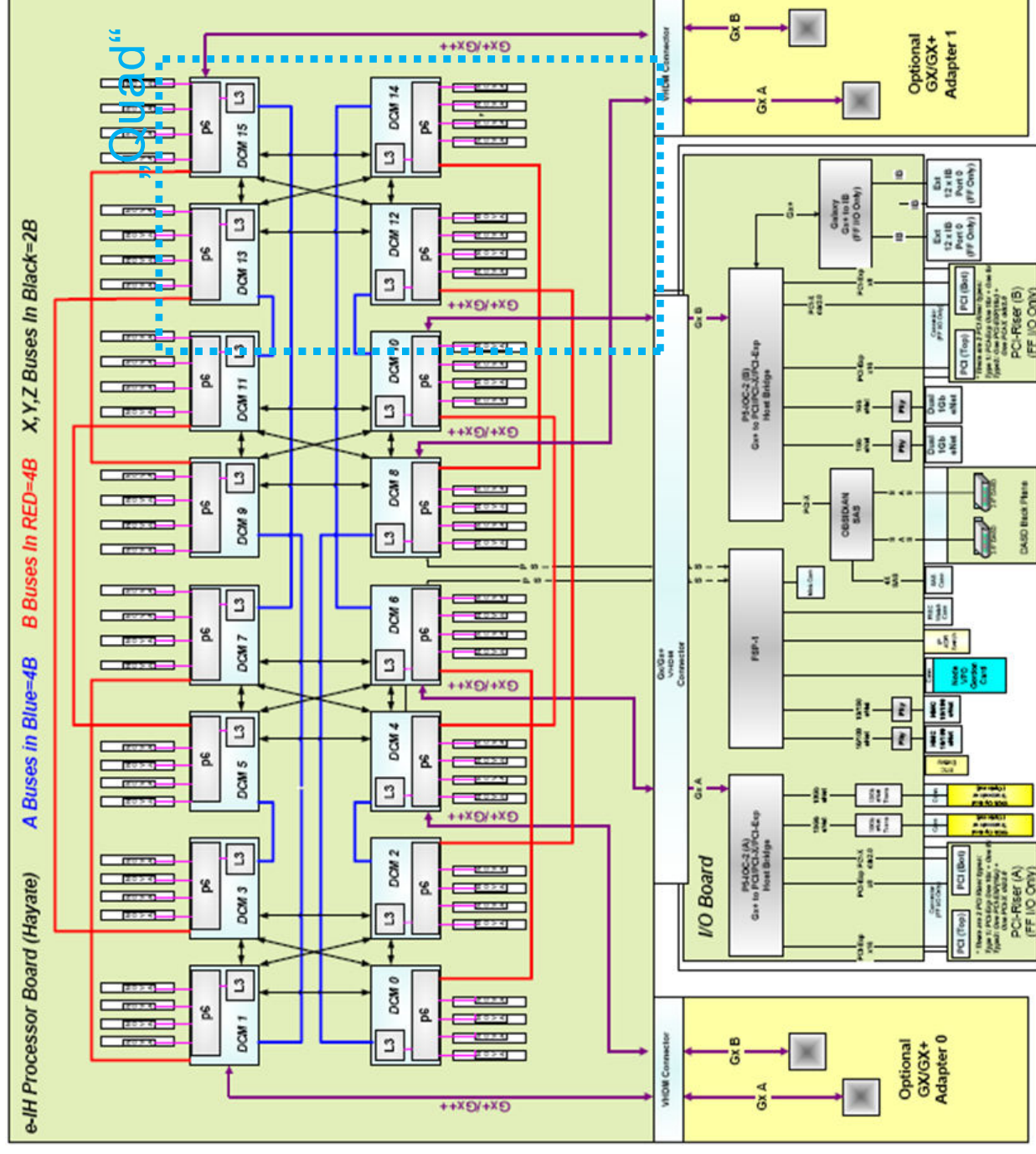
 168 compute node


Node:

 16 dual core chip

32 core

64 virtual cpu

 RAM: 128Gb

-  *H. Q. Le, W. J. Starke, J. S. Fields, F. P. O’Connell,
D. Q. Nguyen, B. J. Ronchetti, W. M. Sauer,
E. M. Schwarz, M. T. Vaden,*
IBM POWER6 microarchitecture,
IBM Journal of Research and Development,
Volume 51 Issue 6, November 2007 .