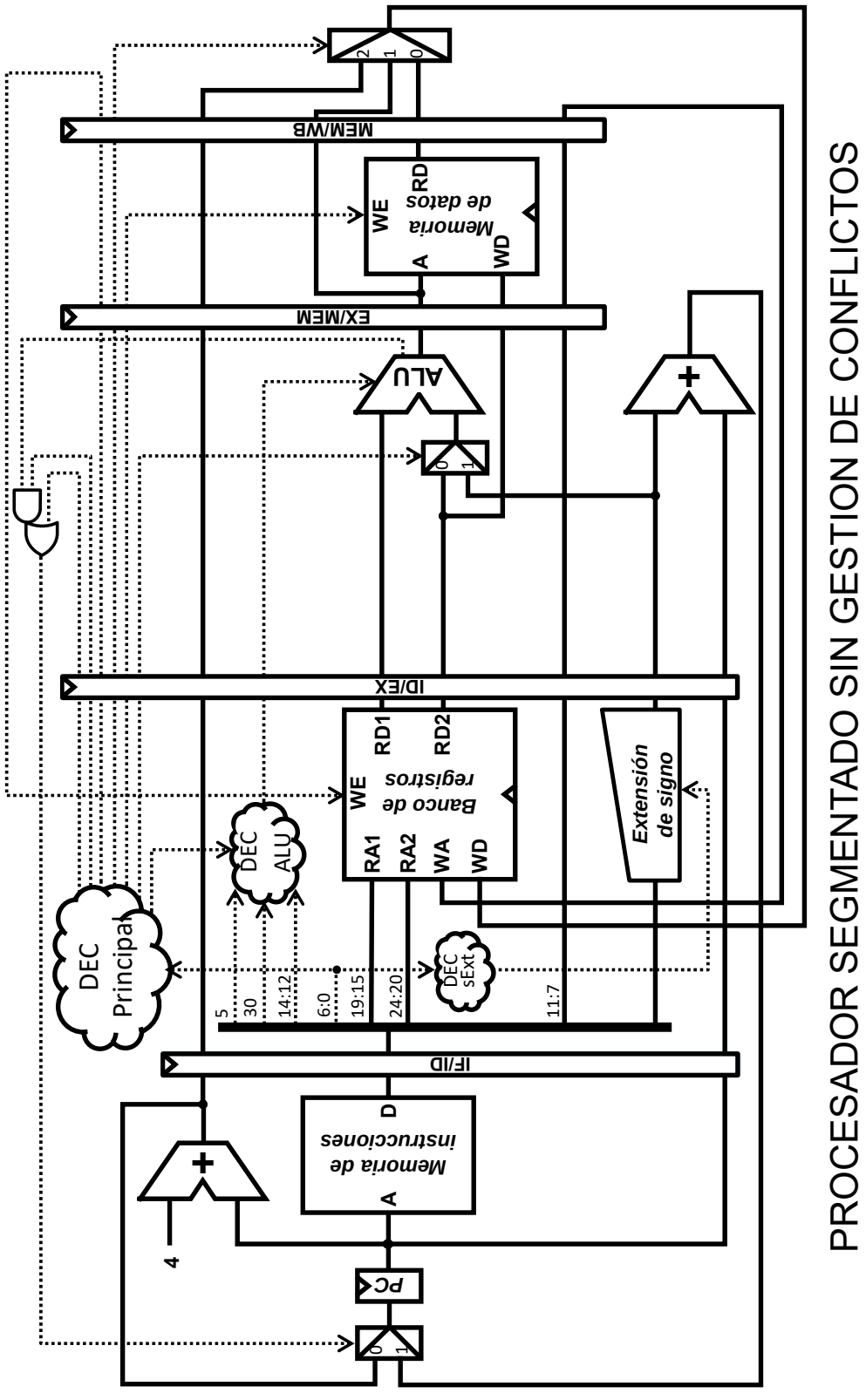
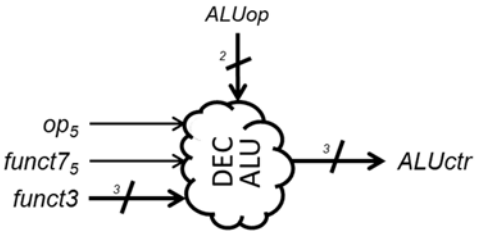
Facultad de Informática
Universidad Complutense de Madrid

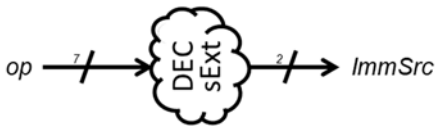
FUNDAMENTOS DE COMPUTADORES II

MICROARQUITECTURA SEGMENTADA

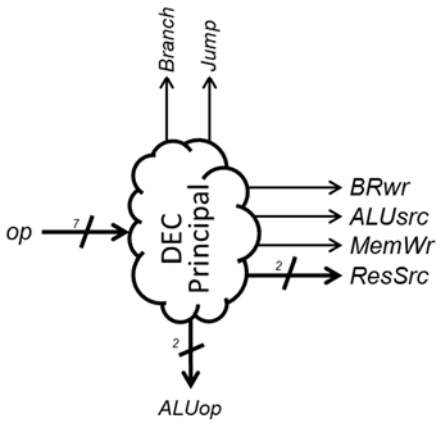




ALUop	op_5	$funct7_5$	funct3	ALUctr
$00^{(sumar)}$	X	X	XXX	$000^{(A + B)}$
$01^{(restar)}$	X	X	XXX	$001^{(A - B)}$
$10^{(operar)}$	0	X	$000^{(addi)}$	$000^{(A + B)}$
$10^{(operar)}$	1	0	$000^{(add)}$	$000^{(A + B)}$
$10^{(operar)}$	1	1	$000^{(sub)}$	$001^{(A - B)}$
$10^{(operar)}$	X	X	$010^{(slt/slti)}$	$101^{(A < B)}$
$10^{(operar)}$	X	X	$110^{(or/ori)}$	$011^{(A \mid B)}$
$10^{(operar)}$	X	X	$111^{(and/andi)}$	$010^{(A \& B)}$

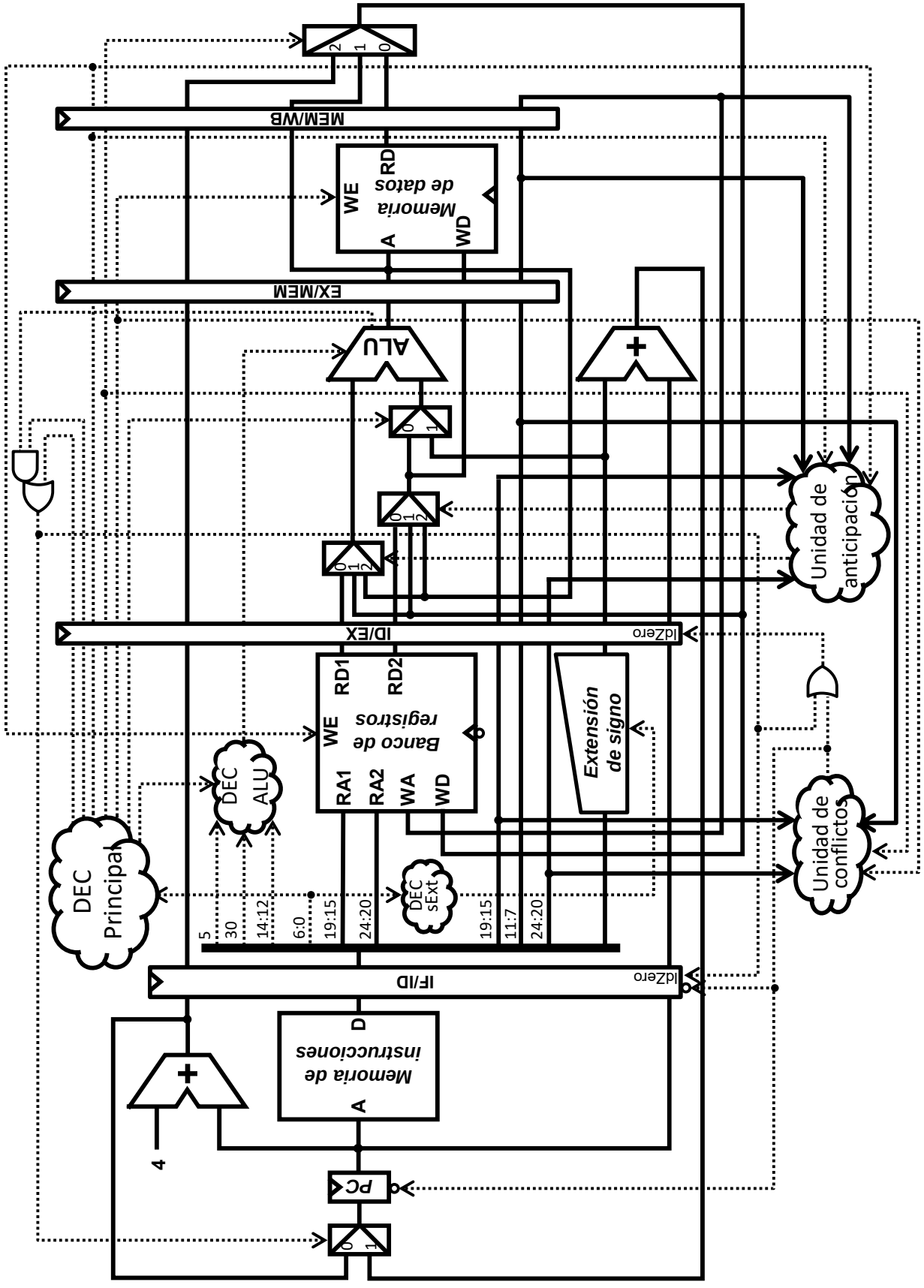


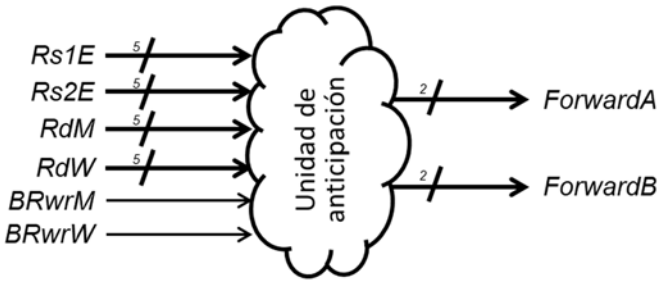
Op	ImmSrc
$0000011^{(lw)}$	$00^{(tipo-I)}$
$0100011^{(sw)}$	$01^{(tipo-S)}$
$0010011^{(tipo-I)}$	$00^{(tipo-I)}$
$0110011^{(tipo-R)}$	–
$1100011^{(beq)}$	$10^{(tipo-B)}$
$1101111^{(jal)}$	$11^{(tipo-J)}$



op	Branch	Jump	BRwr	ALUsrc	ALUop	MemWr	ResSrc
$0000011^{(lw)}$	0	0	1	1	$00^{(sumar)}$	0	00
$0100011^{(sw)}$	0	0	0	1	$00^{(sumar)}$	1	–
$0010011^{(tipo-I)}$	0	0	1	1	$10^{(operar)}$	0	01
$0110011^{(tipo-R)}$	0	0	1	0	$10^{(operar)}$	0	01
$1100011^{(beq)}$	1	0	0	0	$01^{(restar)}$	0	–
$1101111^{(jal)}$	0	1	1	–	–	0	10

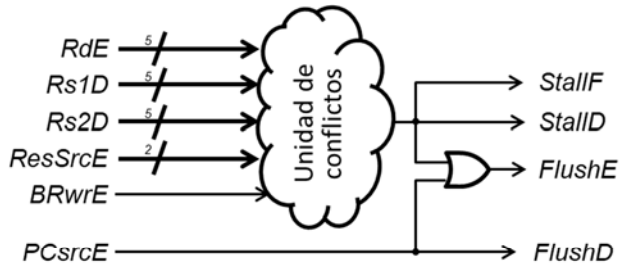
PROCESADOR SEGMENTADO CON GESTION DE CONFLICTOS





if ((Rs1E ≠ 0) & BRwrM & (Rs1E = RdM)) *then* (ForwardA ← $10^{\text{(anticipar MEM)}}$)
  elsif((Rs1E ≠ 0) & BRwrW & (Rs1E = RdW)) *then* (ForwardA ← $01^{\text{(anticipar WB)}}$)
    else (ForwardA ← $00^{\text{(no anticipar)}}$)

if ((Rs2E ≠ 0) & BRwrM & (Rs2E = RdM)) *then* (ForwardB ← $10^{\text{(anticipar MEM)}}$)
  elsif((Rs2E ≠ 0) & BRwrW & (Rs2E = RdW)) *then* (ForwardB ← $01^{\text{(anticipar WB)}}$)
    else (ForwardB ← $00^{\text{(no anticipar)}}$)



if ((ResSrcE = 0) & BRwrE & ((Rs1D = RdE) | (Rs2D = RdE))) *then* (lwStall ← 1) (parar pipeline)
else (lwStall ← 0) (no parar pipeline)
StallF = StallD = lwStall
FlushD = PCsrcE
FlushE = lwStall | PCsrcE