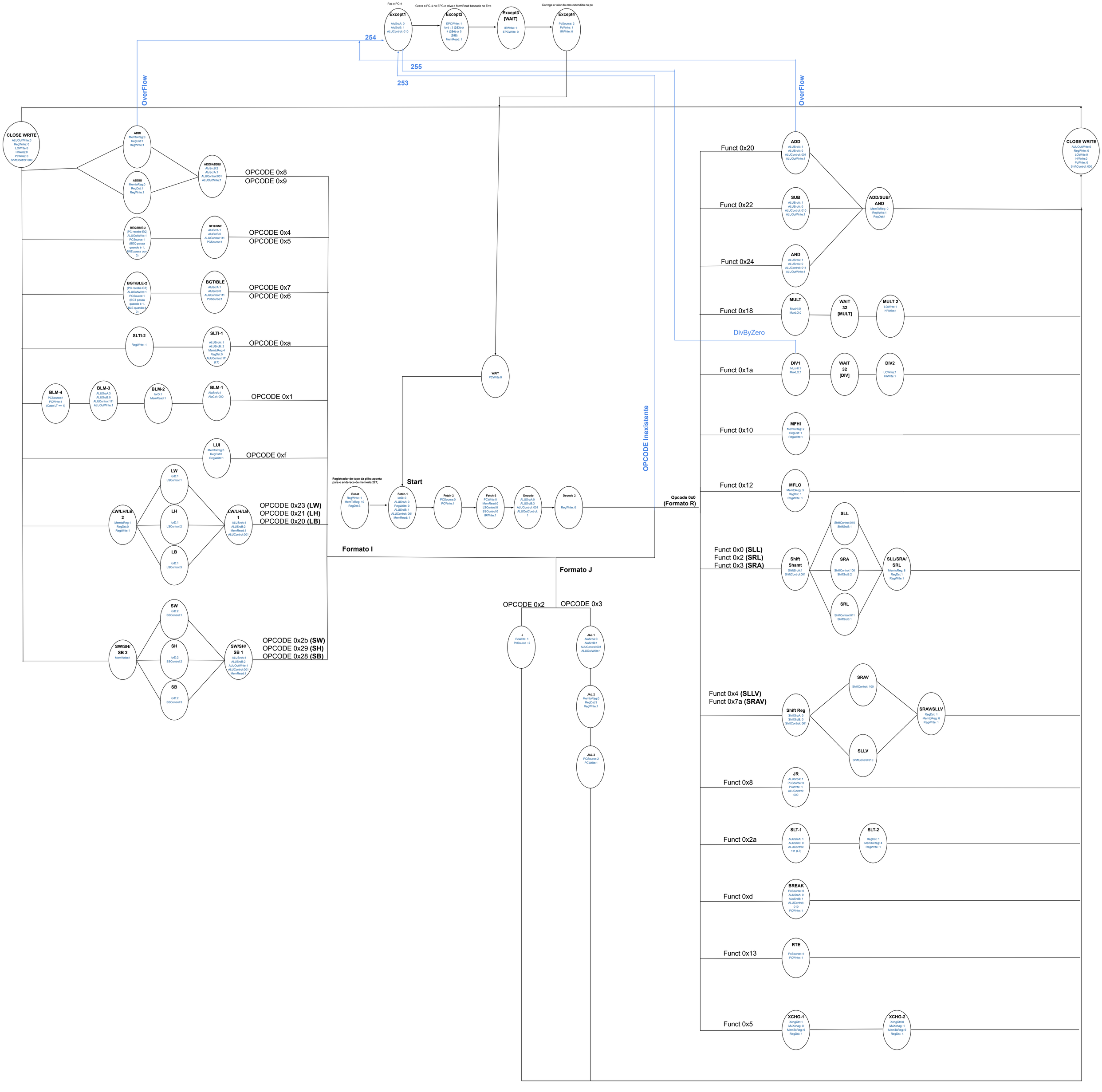
Faz o PC-4
Grava o PC-4 no EPC e ativa o MemRead baseado no Erro
Carrega o valor do erro extendido no pc
Except1
AluSrcA: 0
AluSrcB: 1
ALUControl: 010
Except2
EPCWrite: 1
Iord : 3 (253) or 4 (254) or 5 (255)
MemRead: 1
Except3 [WAIT]
IRWrite: 1
EPCWrite: 0
Except4
PcSource: 2
PcWrite: 1
IRWrite: 0
254
255
253
OverFlow
OverFlow
CLOSE WRITE
ALUOutWrite:0
RegWrite: 0
LOWrite:0
HIWrite:0
PcWrite: 0
ShiftControl: 000
ADDI
MemtoReg:0
RegDst:1
RegWrite:1
ADDIU
MemtoReg:0
RegDst:1
RegWrite:1
ADDI/ADDIU
AluSrcB:2
AluSrcA:1
ALUControl:001
ALUOutWrite:1
OPCODE 0x8
OPCODE 0x9
BEQ/BNE-2
(PC recebe EQ)
ALUOutWrite:1
PCSource:1
(BEQ passa quando é 1, BNE passa com 0)
BEQ/BNE
AluSrcA: 1
AluSrcB:0
ALUControl:111
PCSource:1
OPCODE 0x4
OPCODE 0x5
BGT/BLE-2
(PC recebe GT)
ALUOutWrite:1
PCSource:1
(BGT passa quando é 1, BLE quando é 0)
BGT/BLE
AluSrcA:1
AluSrcB:0
ALUControl:111
PCSource:1
OPCODE 0x7
OPCODE 0x6
SLTI-2
RegWrite: 1
SLTI-1
ALUSrcA: 1
ALUSrcB: 2
MemtoReg:4
RegDst:0
ALUControl:111 (LT)
OPCODE 0xa
BLM-4
PCSource:1
PCWrite:1
(Caso LT == 1)
BLM-3
ALUSrcA:3
ALUSrcB:0
ALUControl:111
ALUOutWrite:1
BLM-2
IorD:1
MemRead:1
BLM-1
AluSrcA:1
AluCtrl: 000
OPCODE 0x1
LUI
MemtoReg:6
RegDst:0
RegWrite:1
OPCODE 0xf
LW/LH/LB 2
MemtoReg:1
RegDst:0
RegWrite:1
LW
IorD:1
LSControl:1
LH
IorD:1
LSControl:2
LB
IorD:1
LSControl:3
LW/LH/LB 1
ALUSrcA:1
ALUSrcB:2
MemRead:1
ALUControl:001
OPCODE 0x23 (LW)
OPCODE 0x21 (LH)
OPCODE 0x20 (LB)
SW/SH/ SB 2
MemWrite:1
SW
IorD:2
SSControl:1
SH
IorD:2
SSControl:2
SB
IorD:2
SSControl:3
SW/SH/ SB 1
ALUSrcA:1
ALUSrcB:2
ALUOutWrite:1
ALUControl:001
MemRead:1
OPCODE 0x2b (SW)
OPCODE 0x29 (SH)
OPCODE 0x28 (SB)
WAIT
PCWrite:0
Registrador do topo da pilha aponta para o endereco de memoria 227;
Start
Reset
RegWrite: 1
MemToReg: 10
RegDst:3
Fetch-1
IorD: 0
ALUSrcA: 0
RegWrite: 0
ALUSrcB: 1
ALUControl: 001
MemRead: 1
Fetch-2
PCSource:0
PCWrite:1
Fetch-3
PCWrite:0
MemRead:0
LSControl:0
SSControl:0
IRWrite:1
Decode
ALUSrcA:0
ALUSrcB:3
ALUControl: 001
ALUOutControl: 1
Decode 2
RegWrite: 0
Formato I
OPCODE Inexistente
Opcode 0x0 (Formato R)
Formato J
OPCODE 0x2
OPCODE 0x3
J
PcWrite: 1
PcSource : 2
JAL 1
AluSrcA:0
AluSrcB:1
ALUControl:001
ALUOutWrite:1
JAL 2
MemtoReg:0
RegDst:3
RegWrite:1
JAL 3
PCSource:2
PCWrite:1
Funct 0x20
ADD
ALUSrcA: 1
ALUSrcA: 0
ALUControl: 001
ALUOutWrite:1
Funct 0x22
SUB
ALUSrcA: 1
ALUSrcA: 0
ALUControl: 010
ALUOutWrite:1
Funct 0x24
AND
ALUSrcA: 1
ALUSrcA: 0
ALUControl: 011
ALUOutWrite:1
ADD/SUB/ AND
MemToReg: 0
RegWrite:1
RegDst:1
Funct 0x18
MULT
MuxHI:0
MuxLO:0
WAIT 32 [MULT]
MULT 2
LOWrite:1
HIWrite:1
DivByZero
Funct 0x1a
DIV1
MuxHI:1
MuxLO:1
WAIT 32 [DIV]
DIV2
LOWrite:1
HIWrite:1
Funct 0x10
MFHI
MemtoReg: 2
RegDst: 1
RegWrite:1
Funct 0x12
MFLO
MemtoReg: 3
RegDst: 1
RegWrite: 1
Funct 0x0 (SLL)
Funct 0x2 (SRL)
Funct 0x3 (SRA)
Shift Shamt
ShiftSrcA:1
ShiftControl:001
SLL
ShiftControl:010
ShiftSrcB:1
SRA
ShiftControl:100
ShiftSrcB:2
SRL
ShiftControl:011
ShiftSrcB:1
SLL/SRA/ SRL
MemtoReg: 8
RegDst:1
RegWrite:1
Funct 0x4 (SLLV)
Funct 0x7a (SRAV)
Shift Reg
ShiftSrcA: 0
ShiftSrcB: 0
ShiftControl: 001
SRAV
ShiftControl: 100
SLLV
ShiftControl:010
SRAV/SLLV
RegDst: 1
MemtoReg: 8
RegWrite: 1
Funct 0x8
JR
ALUSrcA: 1
PCSource: 0
PCWrite: 1
ALUControl: 000
Funct 0x2a
SLT-1
ALUSrcA: 1
ALUSrcB: 0
ALUControl: 111 (LT)
SLT-2
RegDst: 1
MemToReg: 4
RegWrite: 1
Funct 0xd
BREAK
PcSource: 0
ALUSrcA: 0
ALuSrcB: 1
ALUControl: 010
PCWrite: 1
Funct 0x13
RTE
PcSource: 4
PCWrite: 1
Funct 0x5
XCHG-1
XchgCtrl:1
MuXchag: 0
MemToReg: 9
RegDst: 1
XCHG-2
XchgCtrl:0
MuXchag: 1
MemToReg: 9
RegDst: 4
CLOSE WRITE
ALUOutWrite:0
RegWrite: 0
LOWrite:0
HIWrite:0
PcWrite: 0
ShiftControl: 000