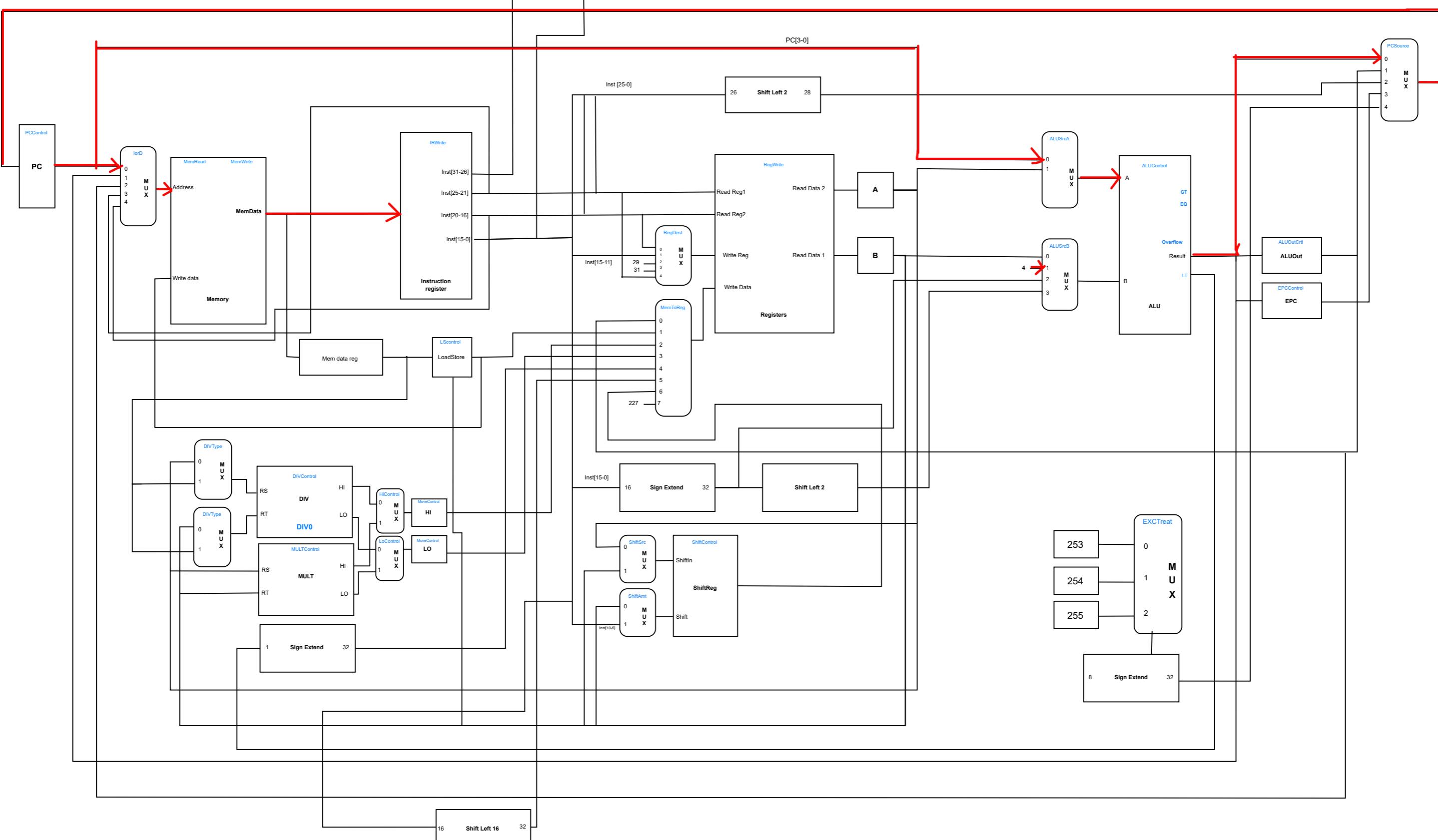
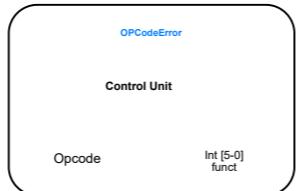
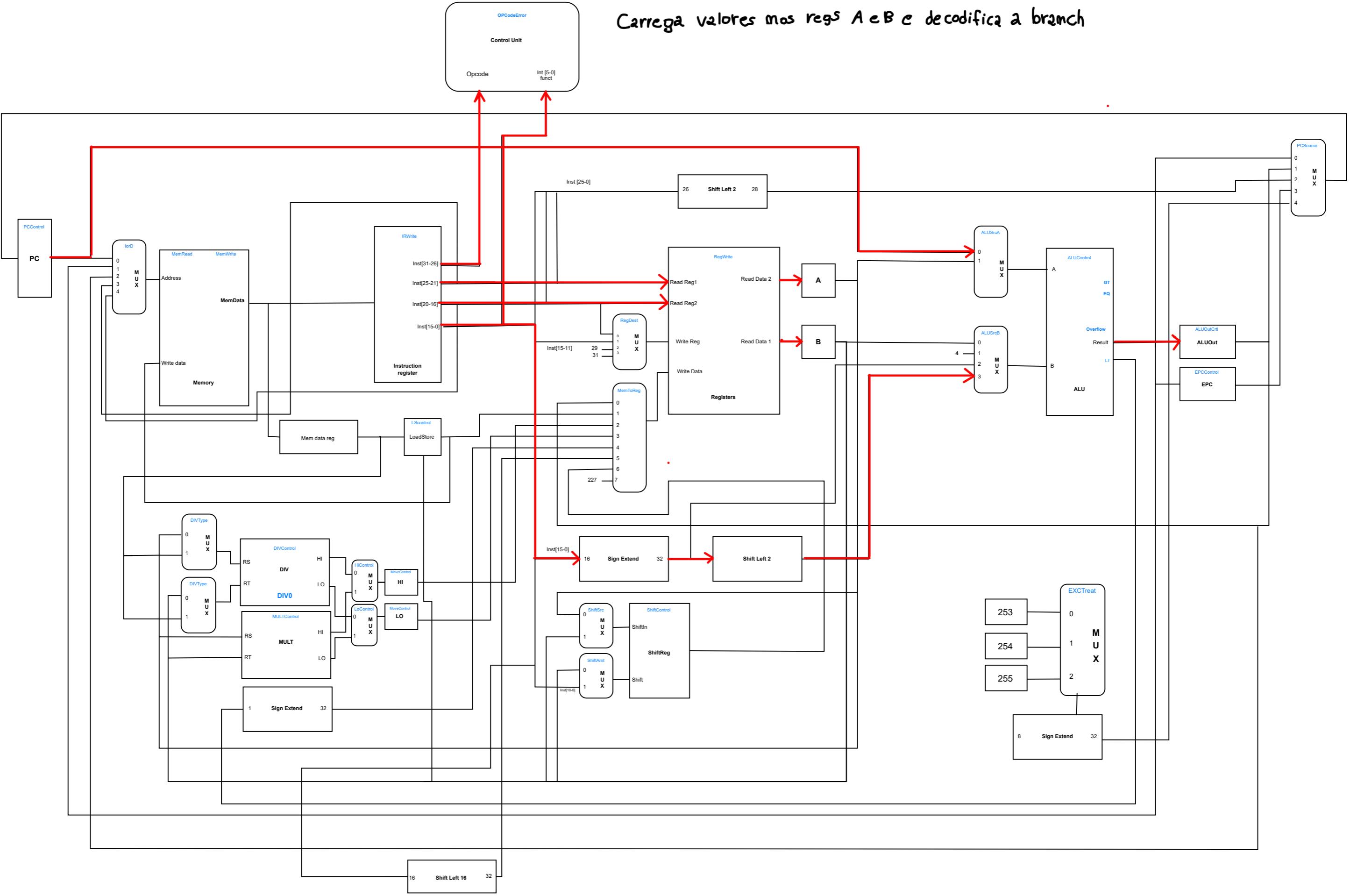


Passo comum: lê memória e incrementa PC



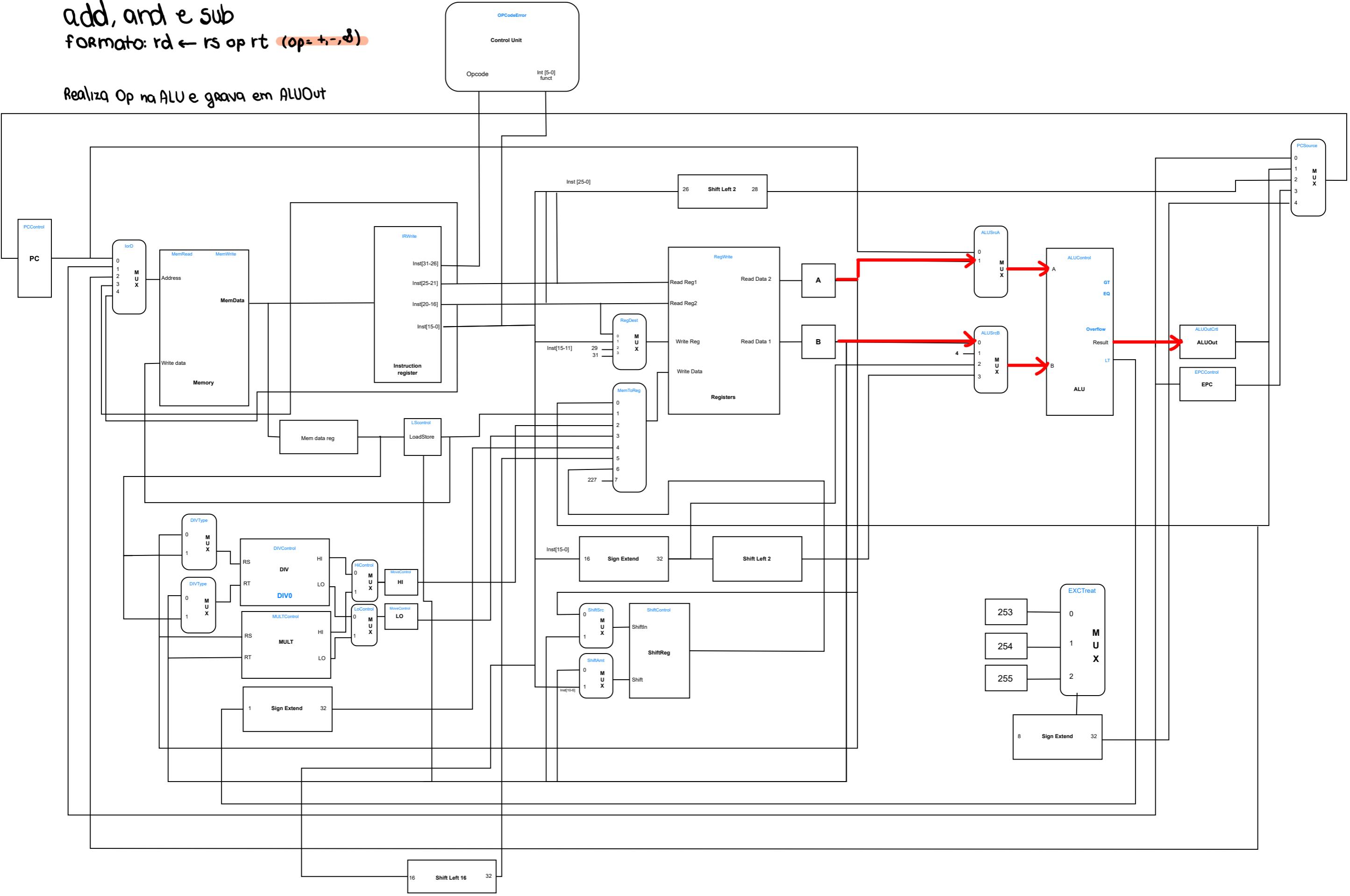
Carrega valores mas reg\$ A e B e decodifica a branch



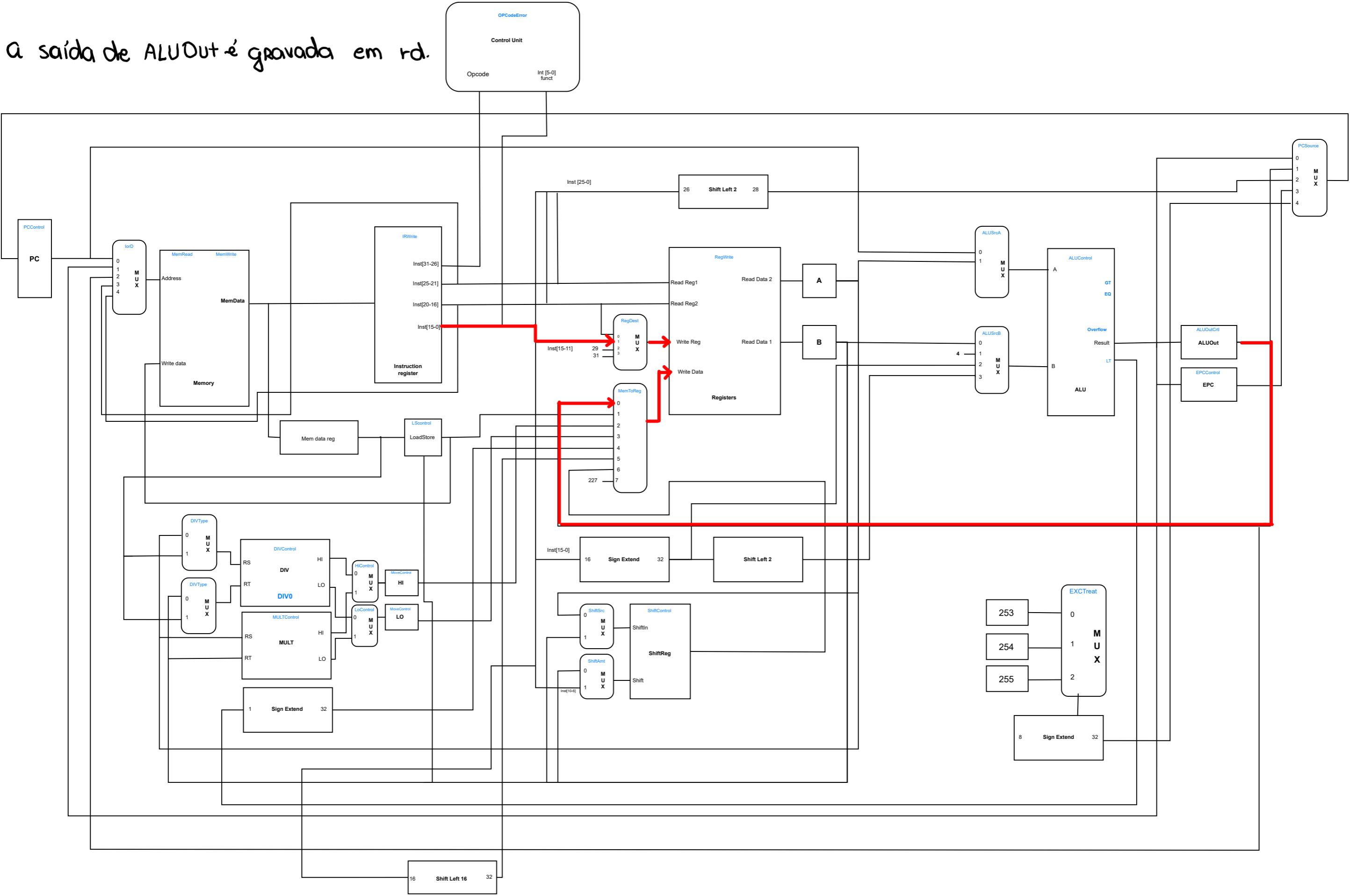
add, and e sub

formato: $rd \leftarrow rs \text{ op } rt$ ($\text{op} = +, -, \delta$)

Realiza Op na ALU e grava em ALUOut

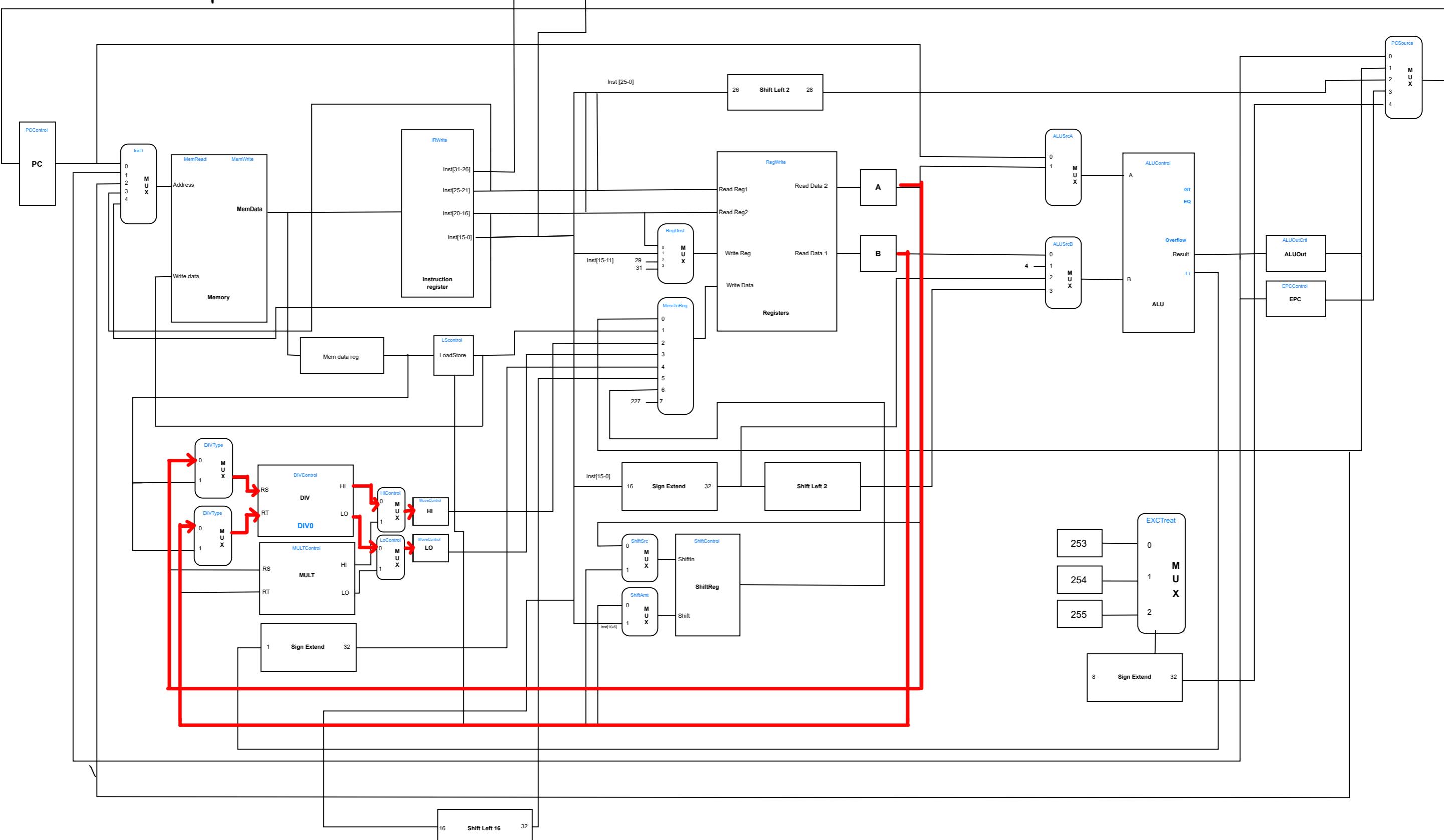
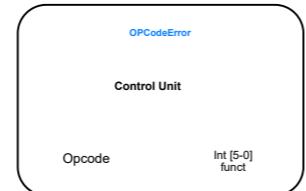


a saída de ALUOut é gravada em rd.



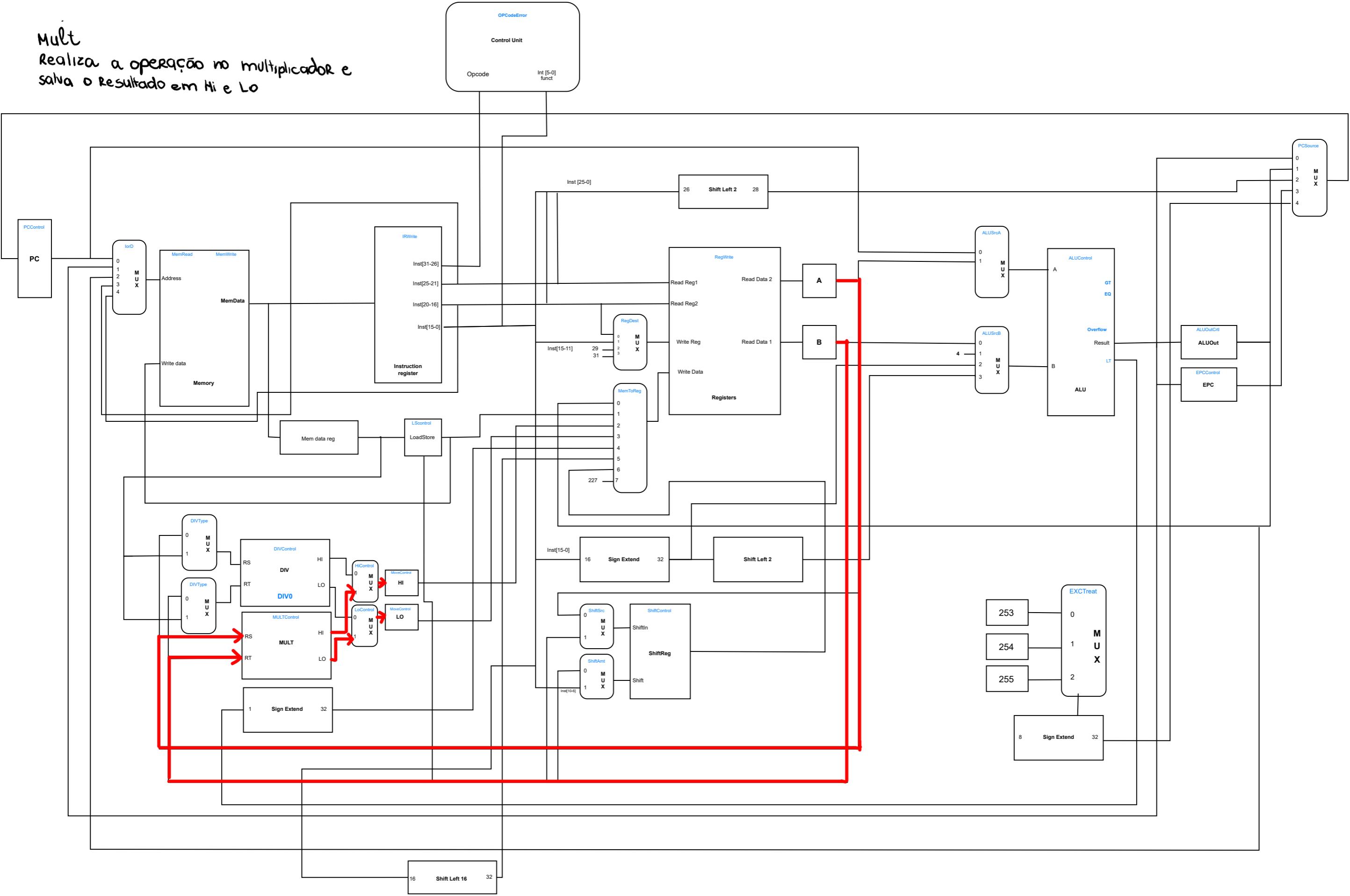
DIV

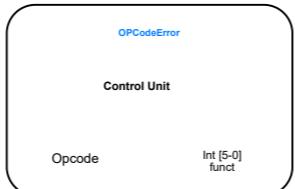
Realiza a operação no divisor e salva
o resto em HI e o quociente em LO



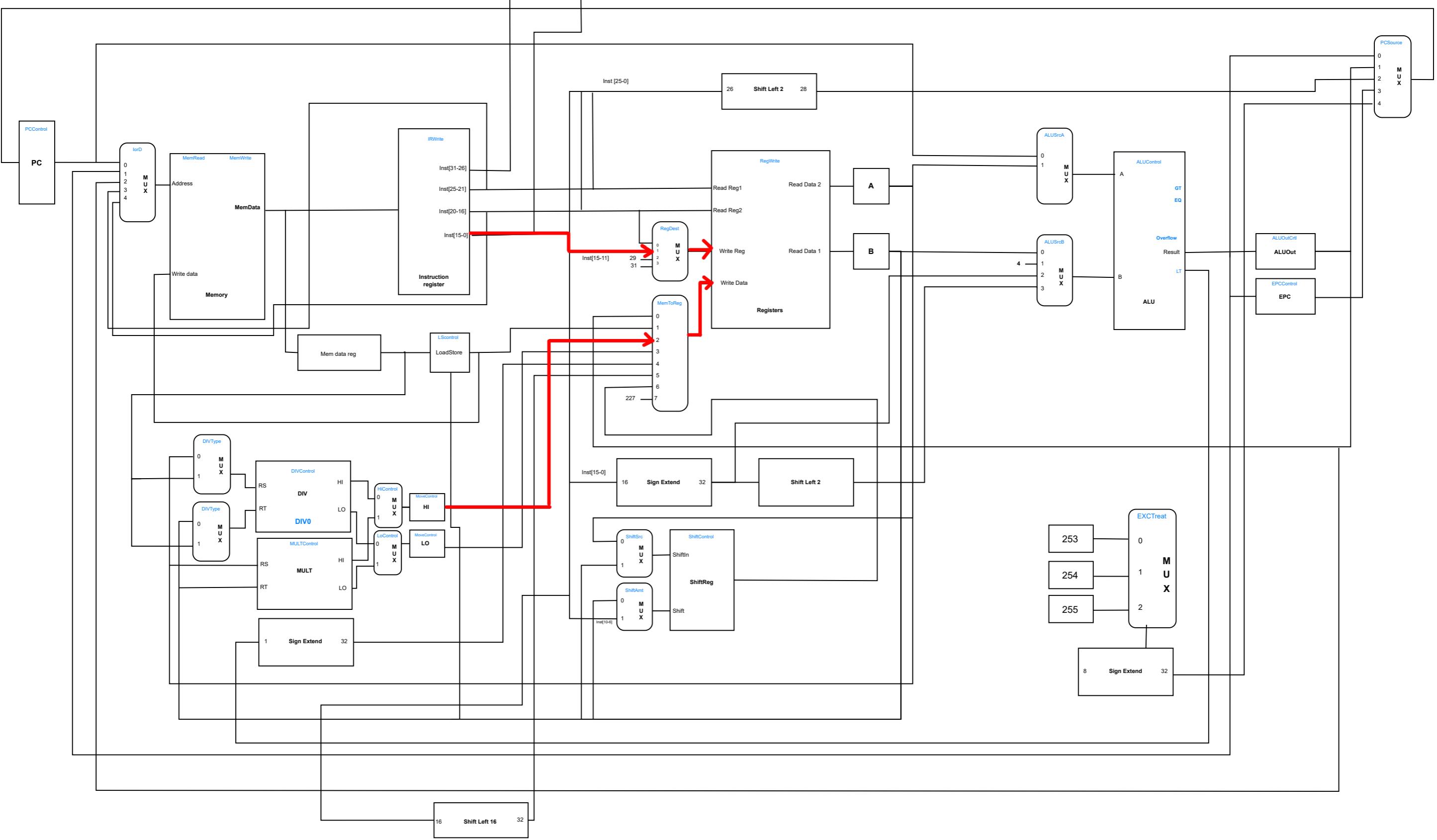
Mult

Realiza a operação no multiplicador e
salva o resultado em Hi e Lo

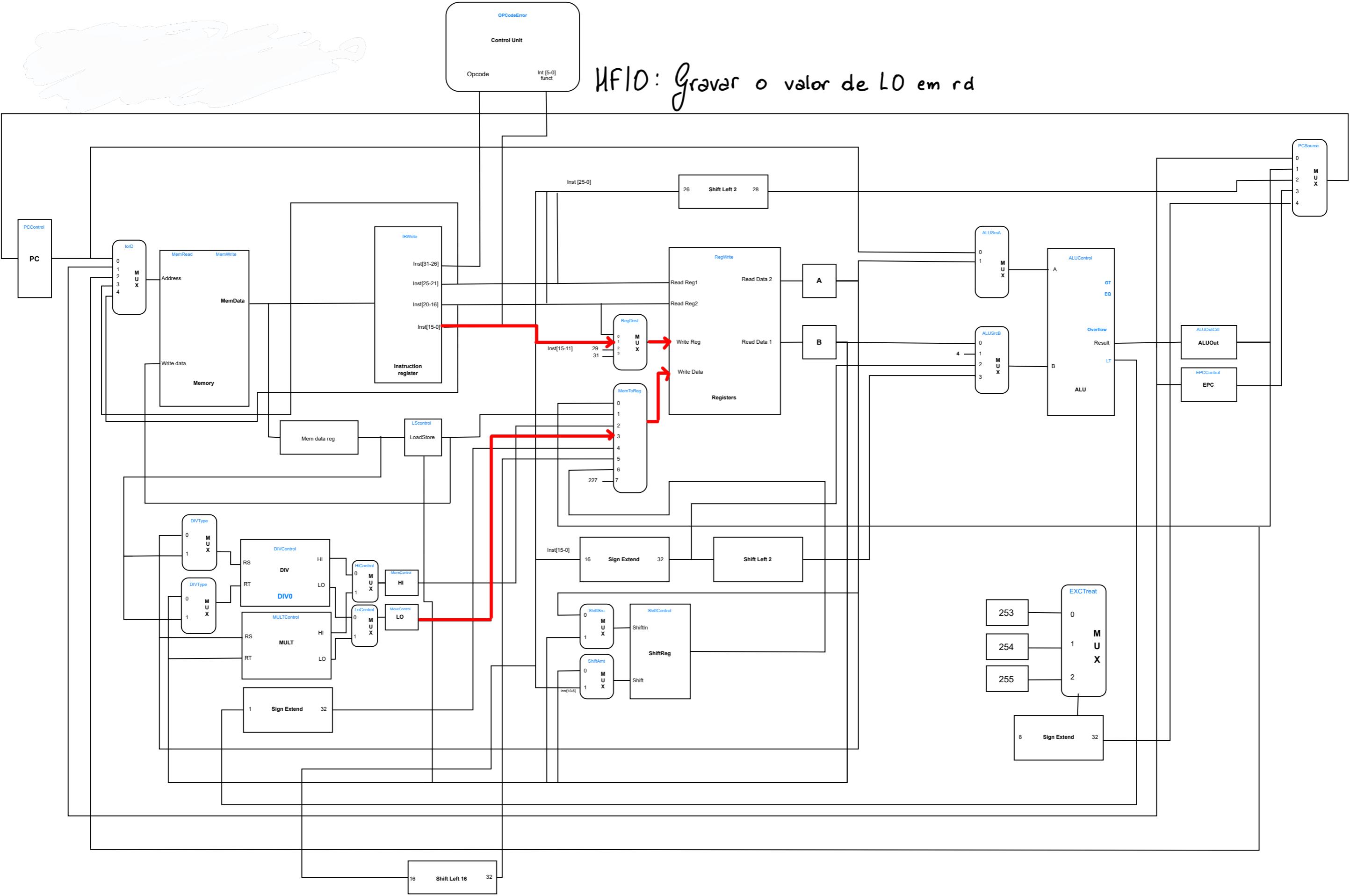


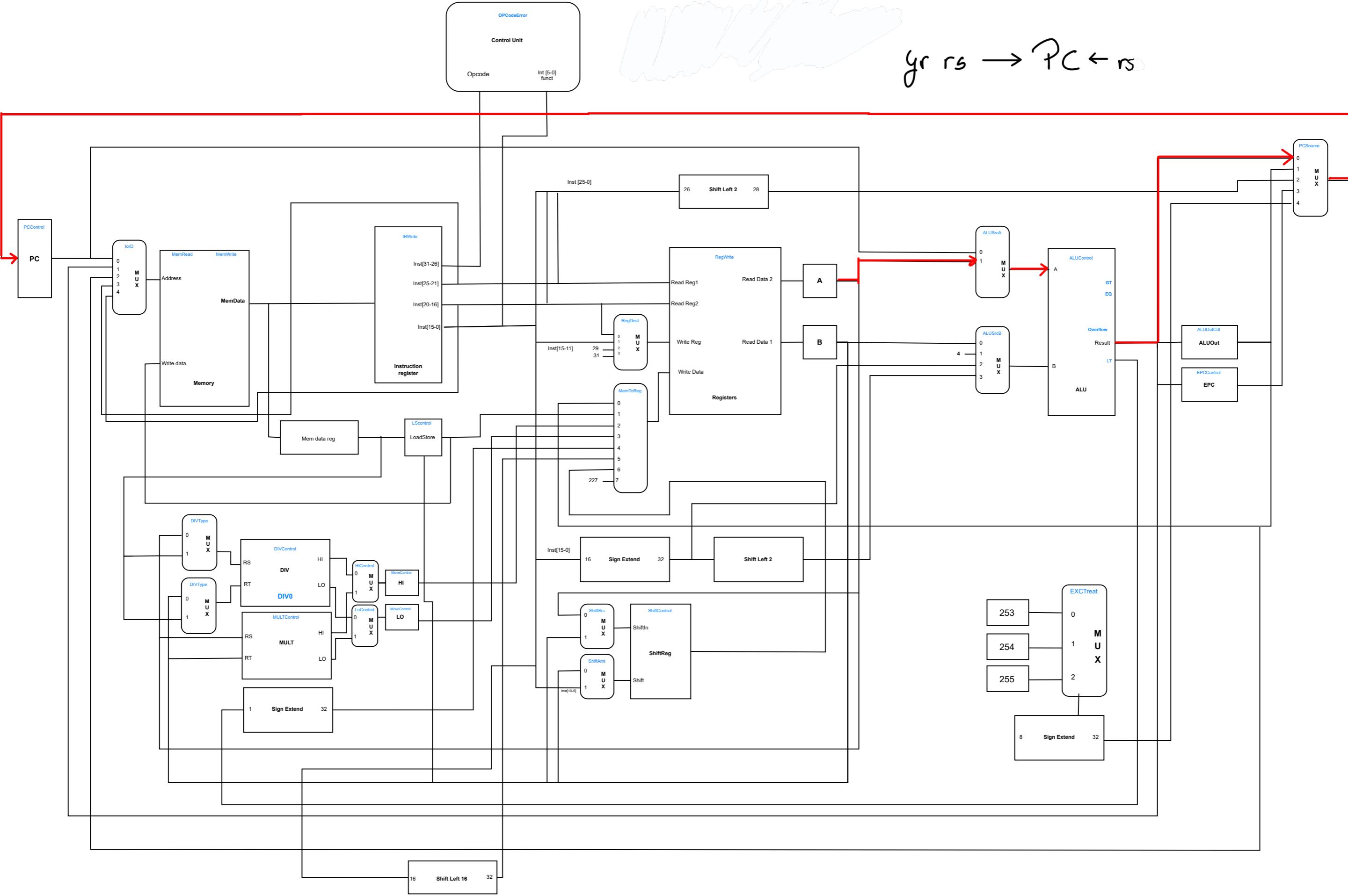


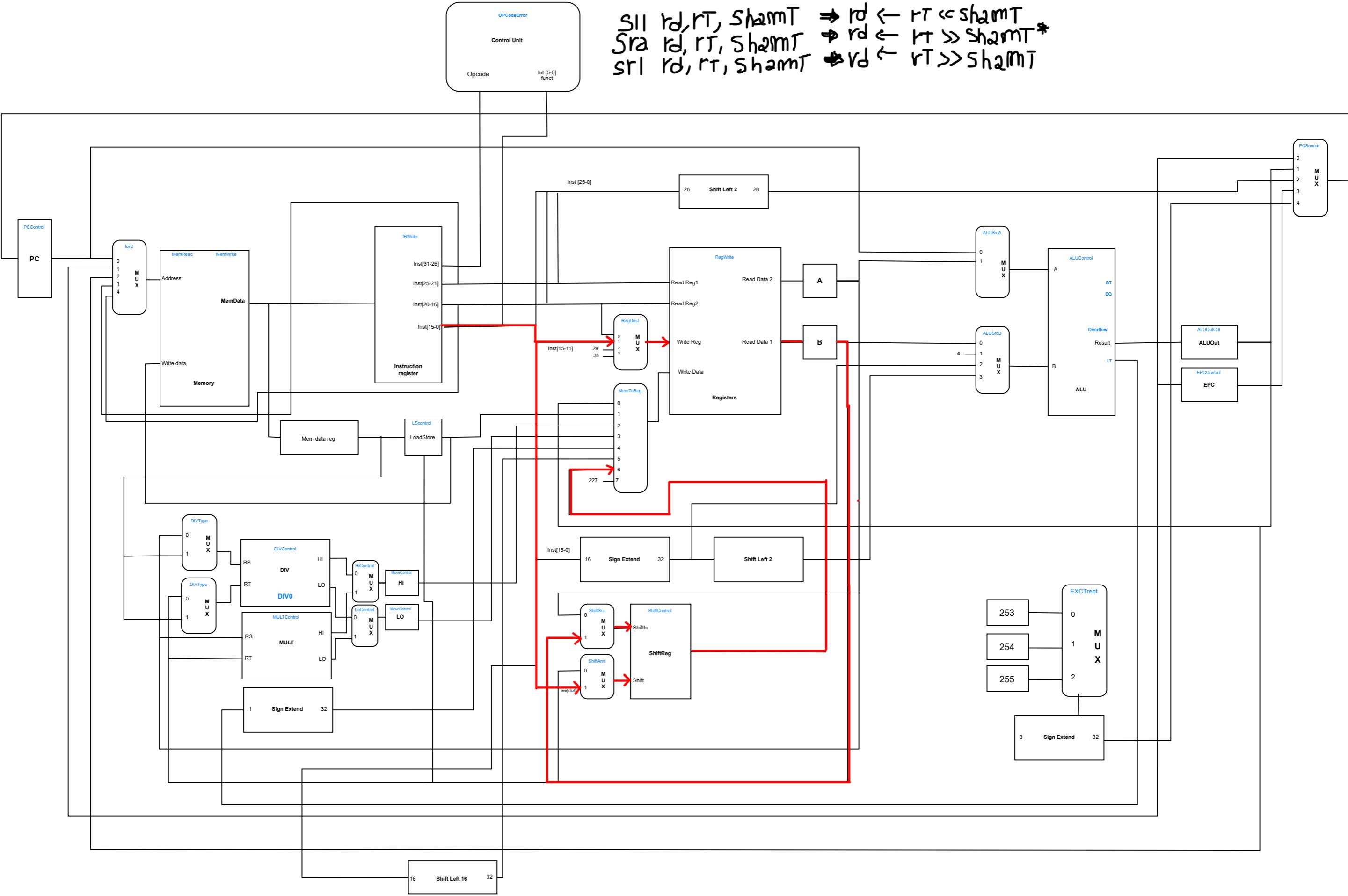
MFHI: Gravar o valor de HI em rd



HF10: Gravar o valor de LO em rd



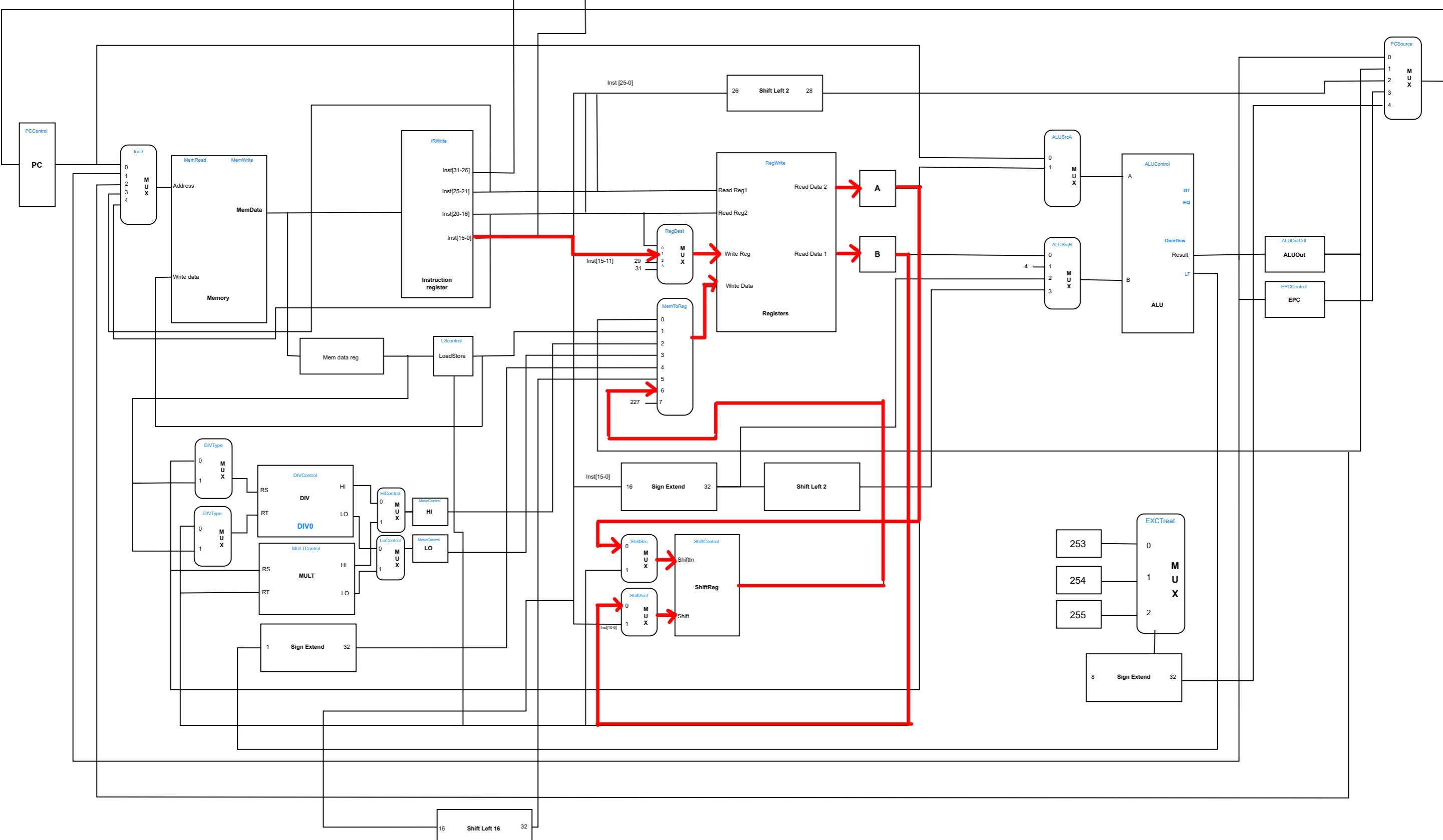
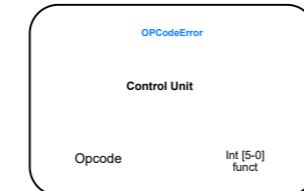




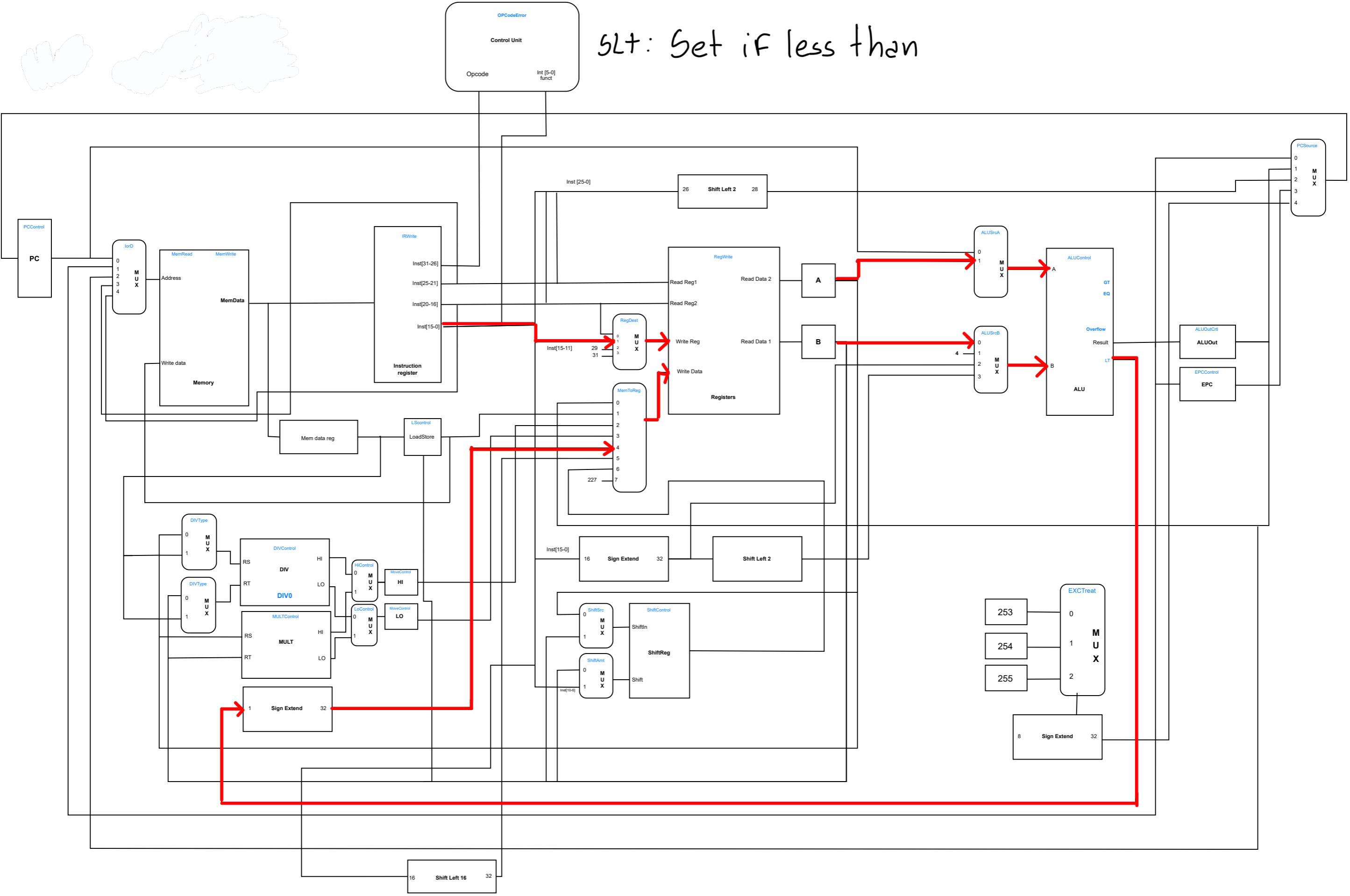
SLV, SRAV

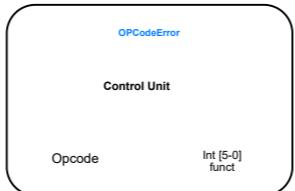
$rd \leftarrow rs \text{ shift } rt$
(shift: \ll ou \gg)

desloca A em uma quantidade
de shamt

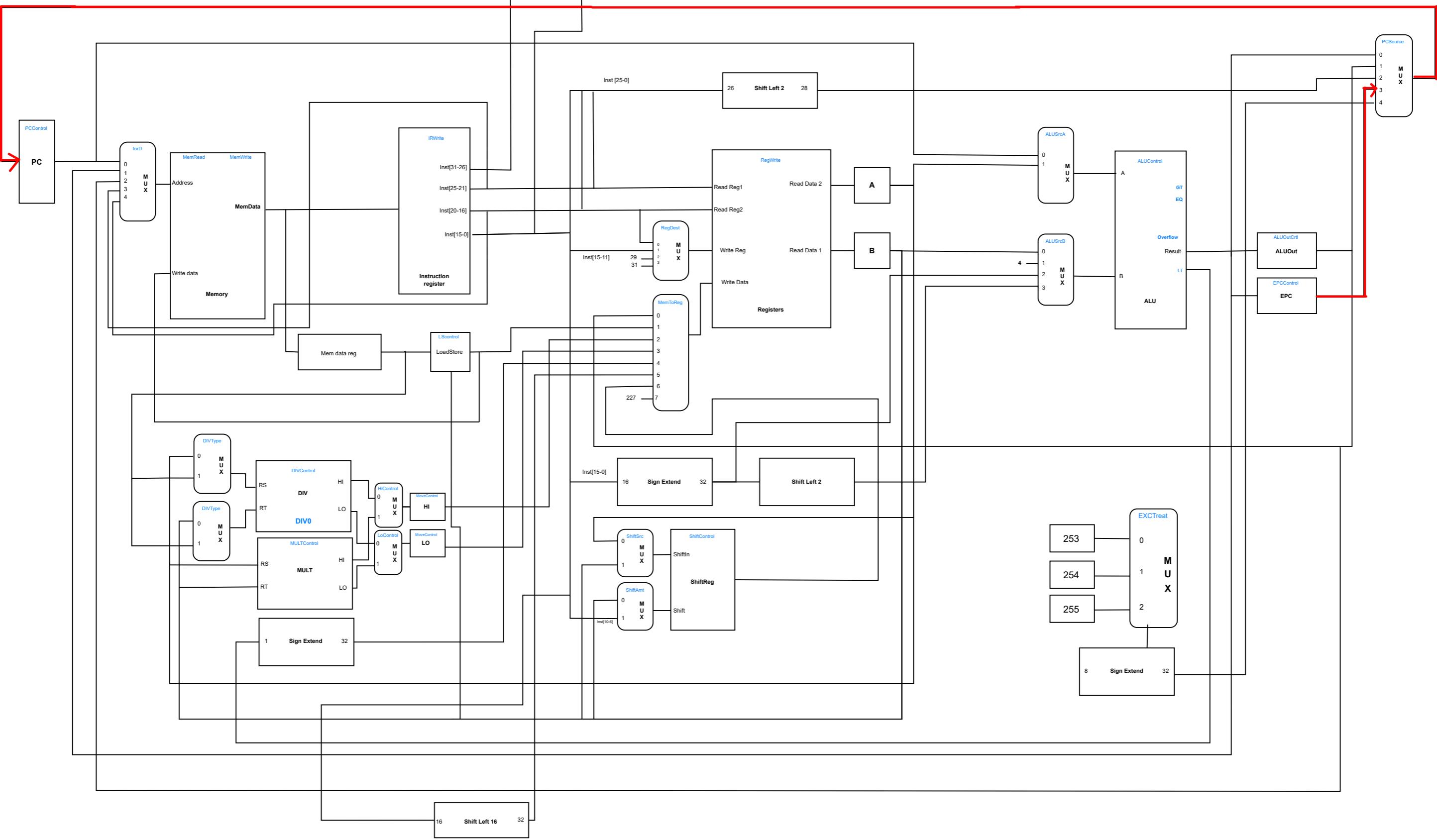


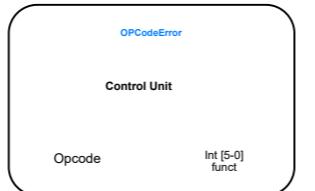
SLT: Set if less than



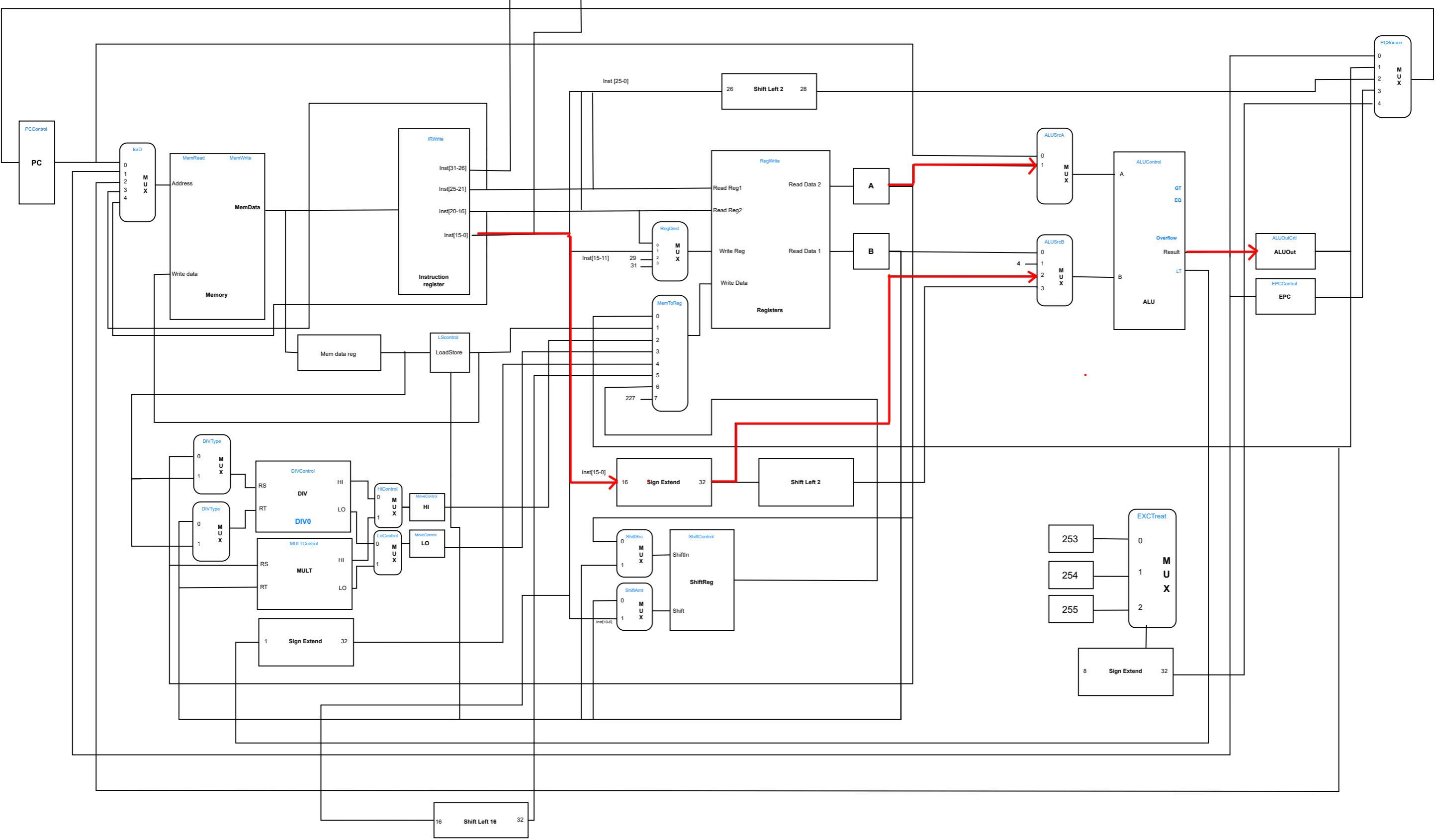


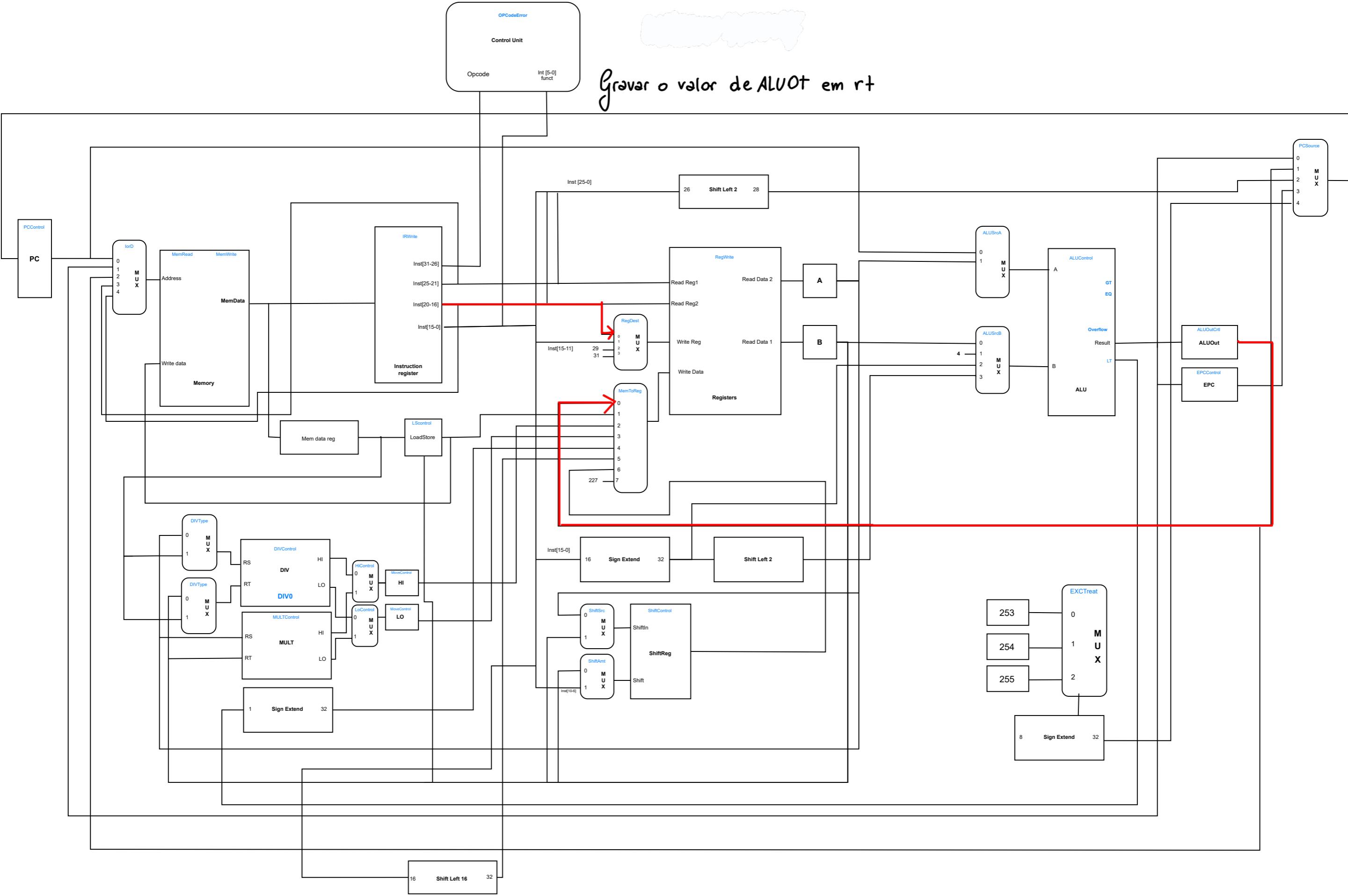
RTC \Rightarrow PC \leftarrow EPC

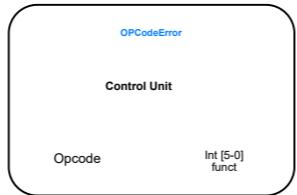




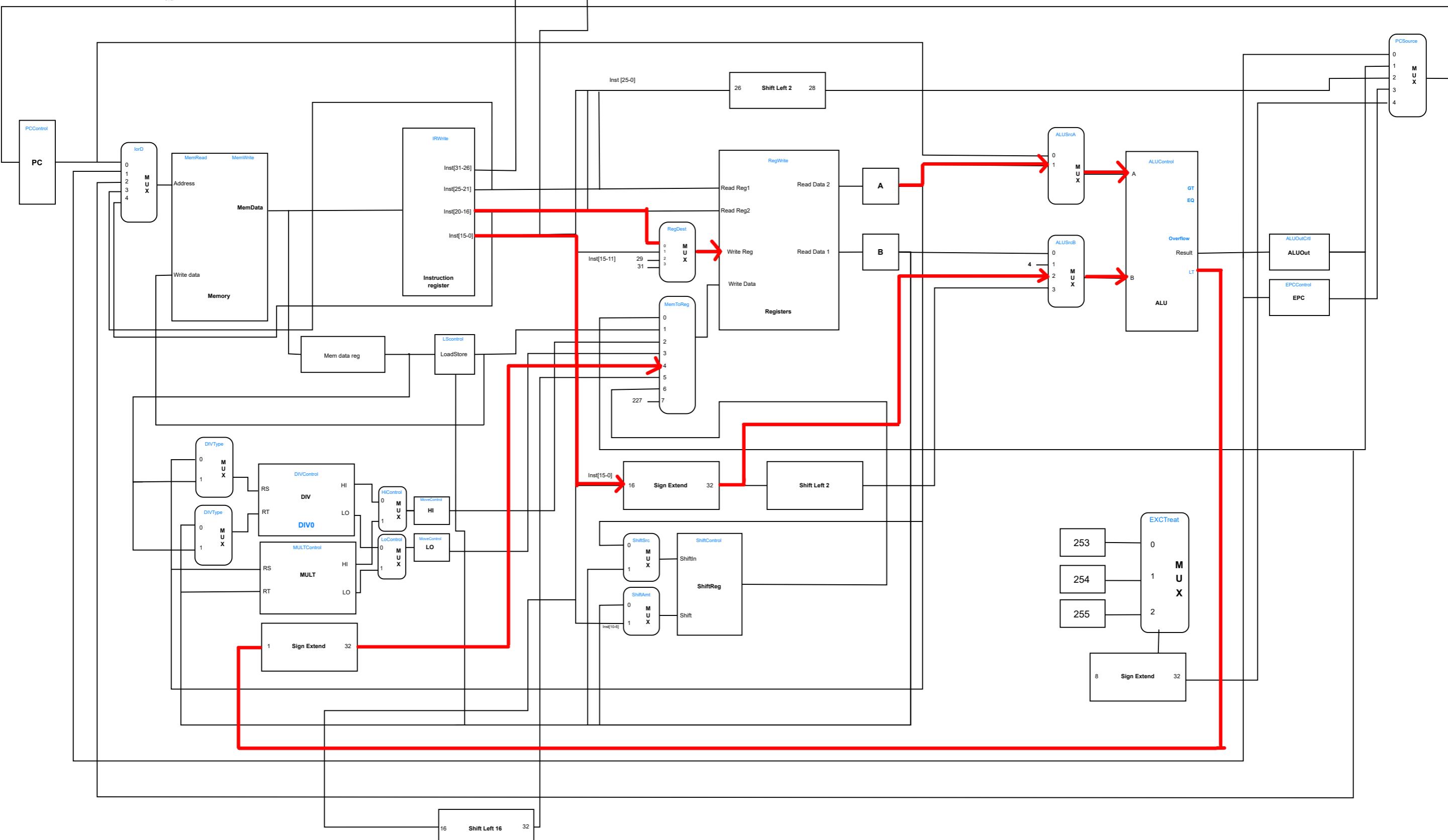
Addi RT, rs, immidato \Rightarrow RT \leftarrow rs + immidato
 Addiu RT, rs, immidato \Rightarrow RT \leftarrow rs + immidato
 (sem overflow)





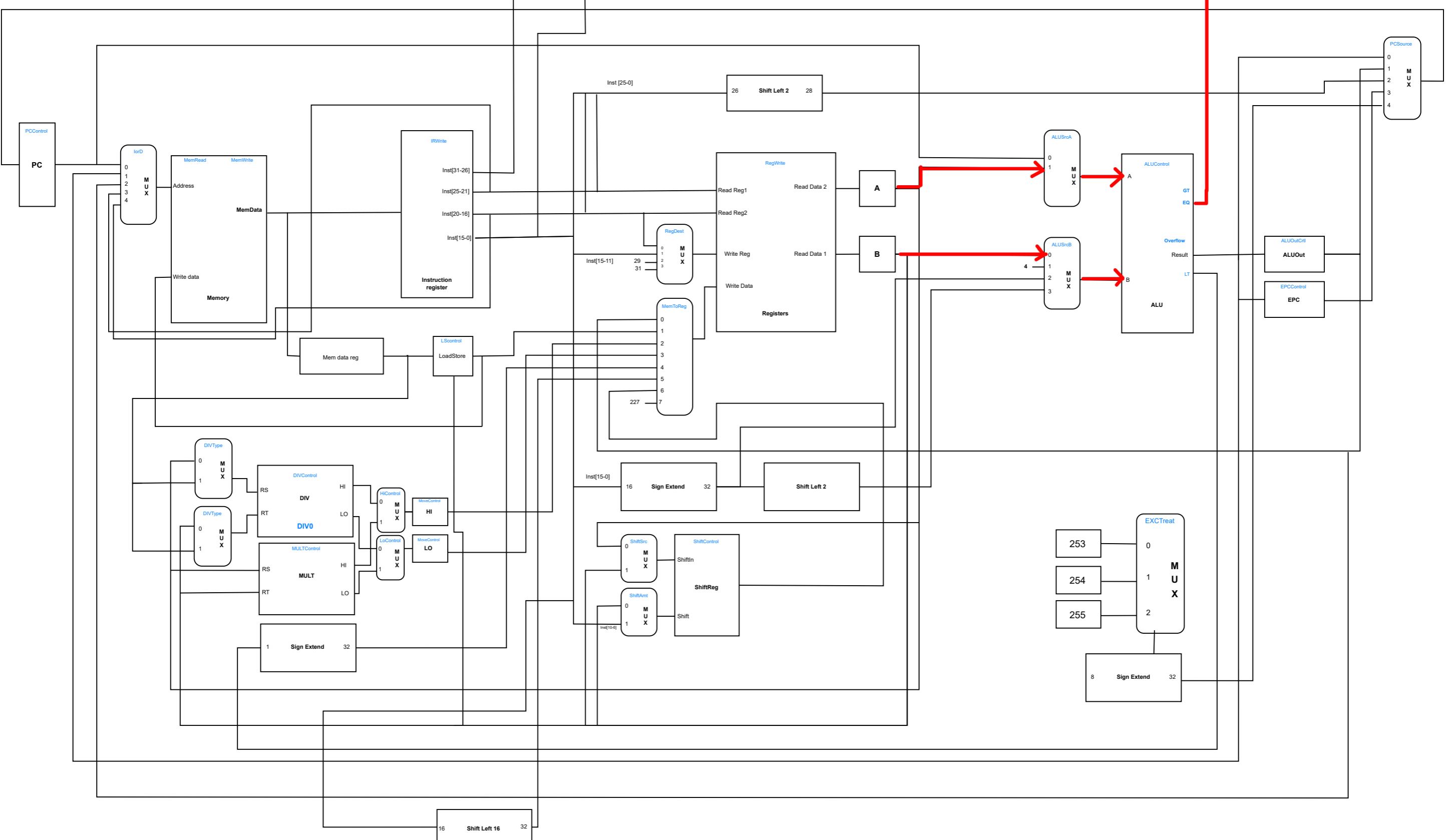


SLT1: Set if less than (constante)



BEQ rs, rt, offset (se $rs == rt$, desvia)

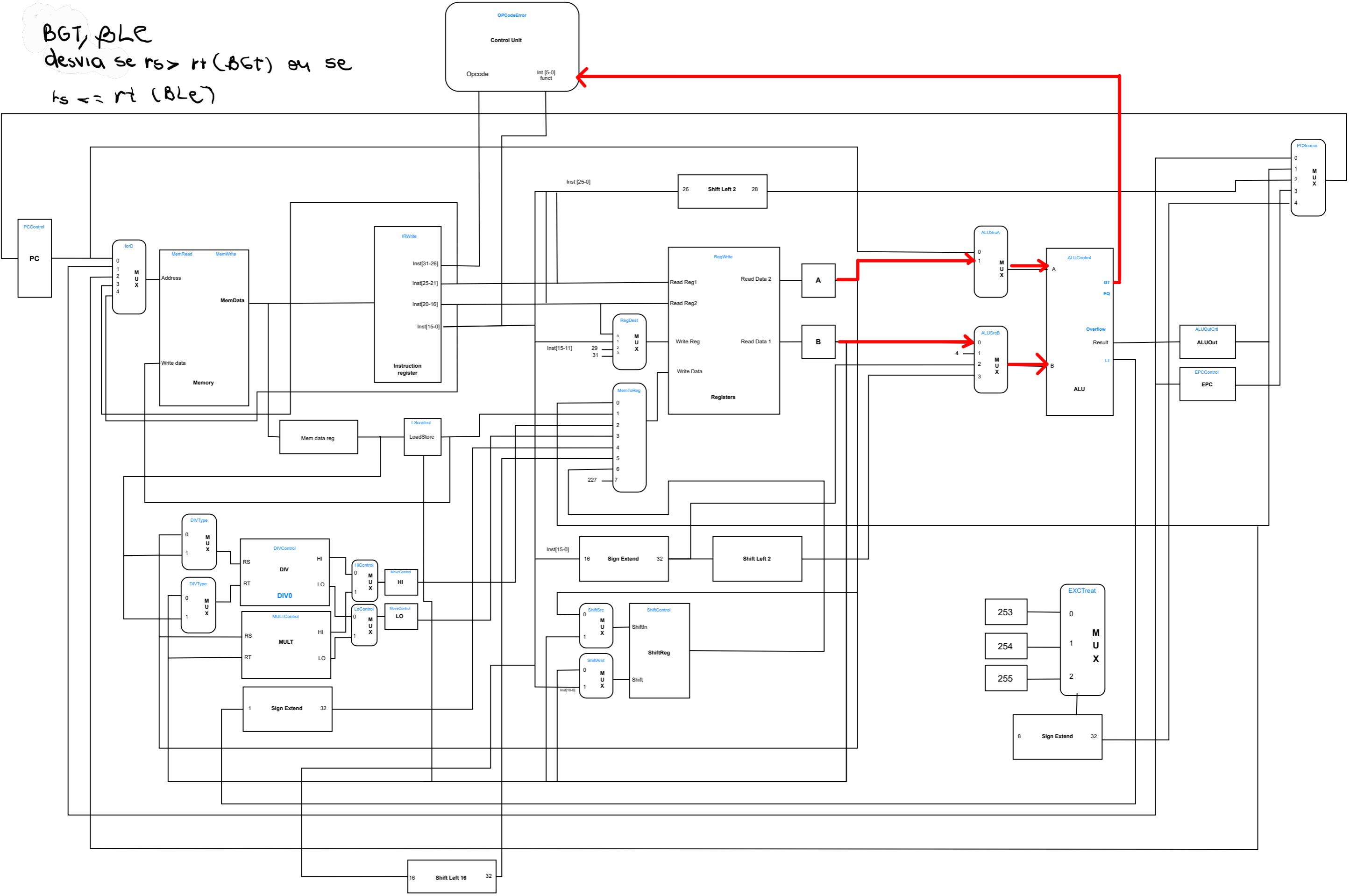
BNE rs, rt, offset (se $rs != rt$, desvia)

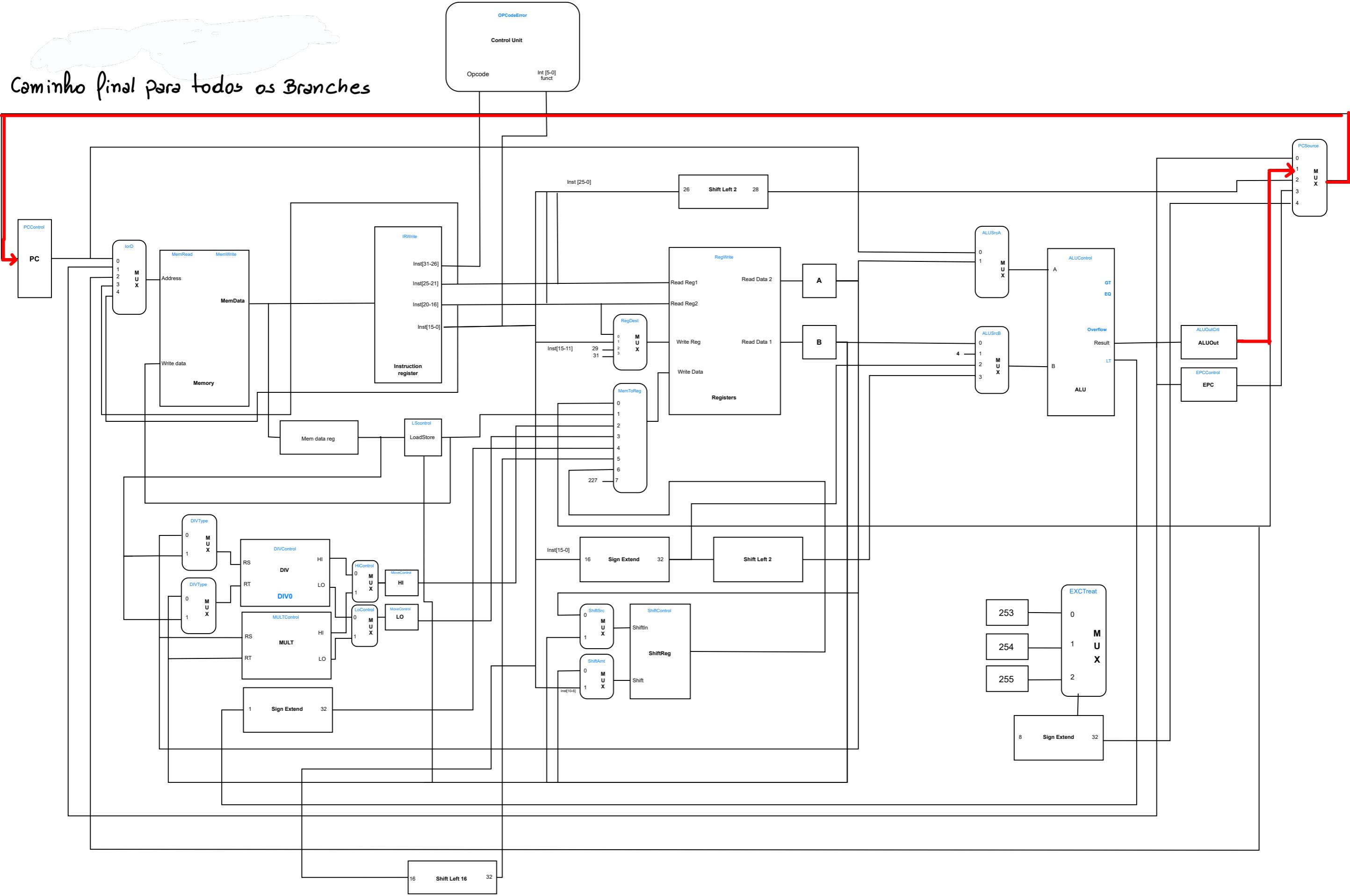


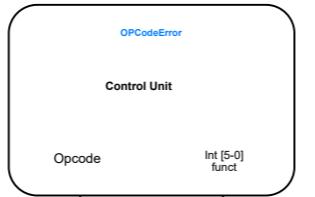
BGT, BLE

desvia se rs > rt (BGT) ou se

rs <= rt (BLE)

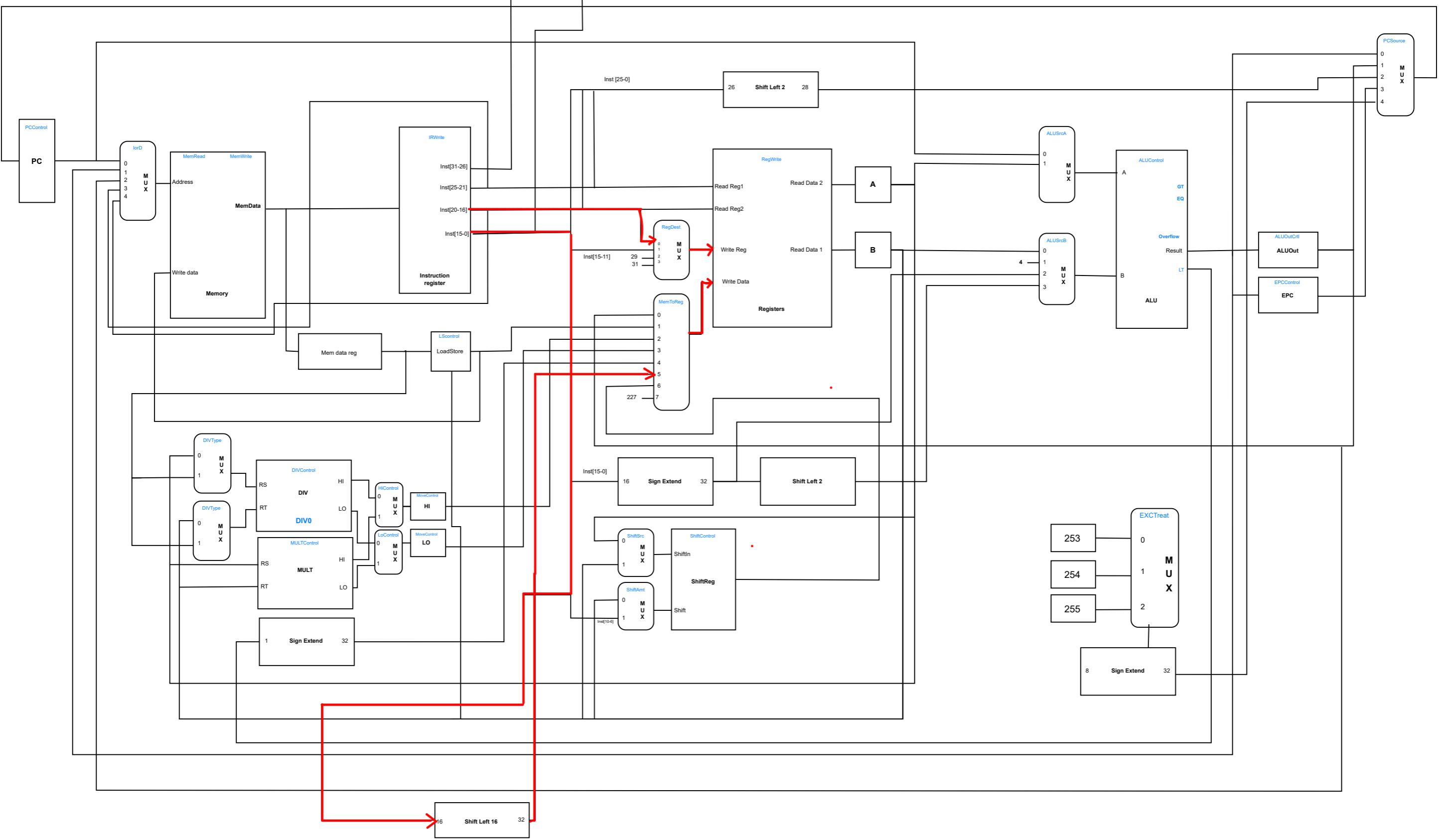


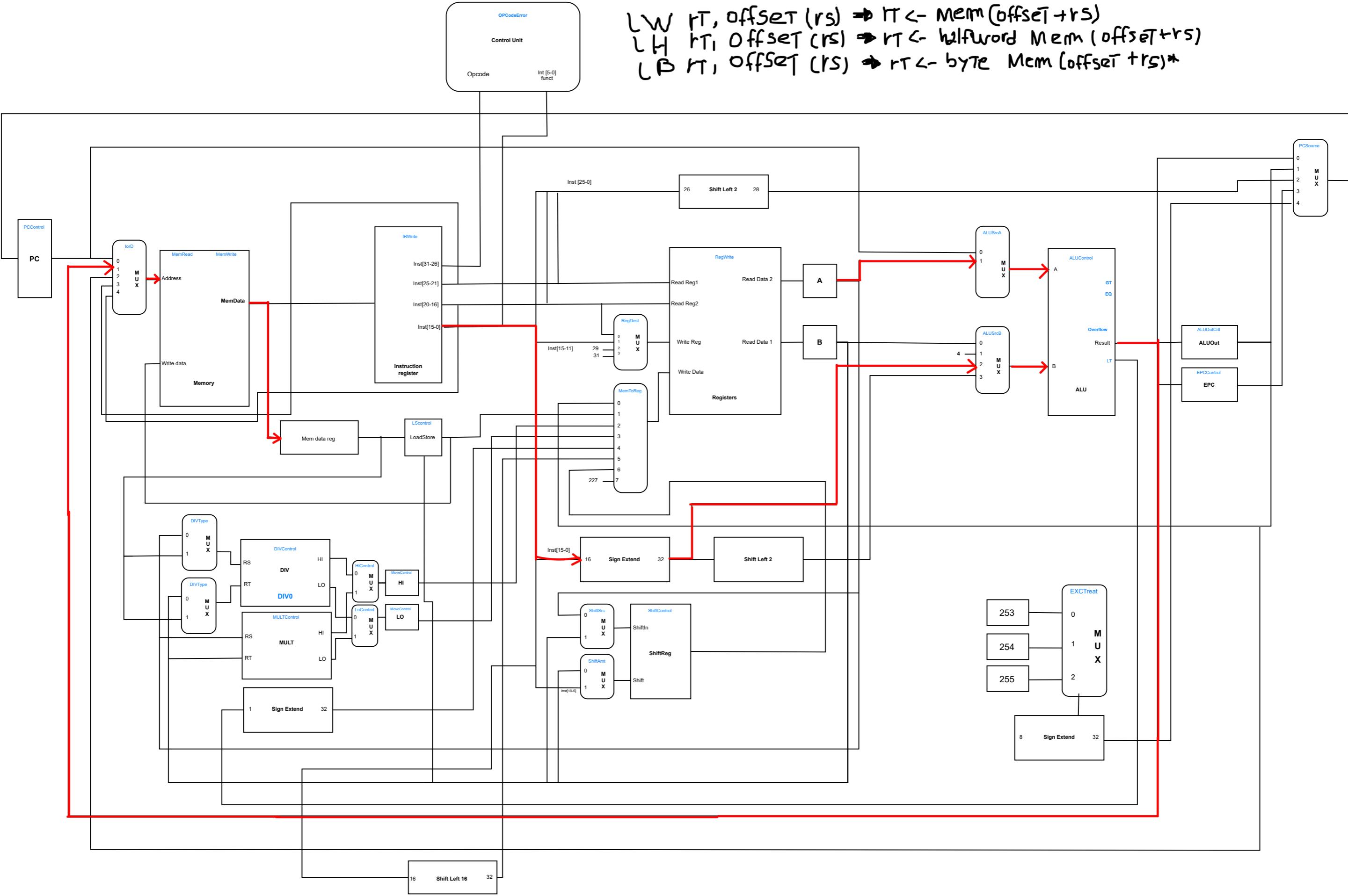


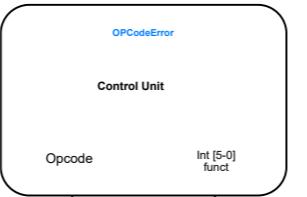


$lui \ rt, \text{immediato} \Rightarrow rt \leftarrow \text{Immed} | 270 \ll 16$

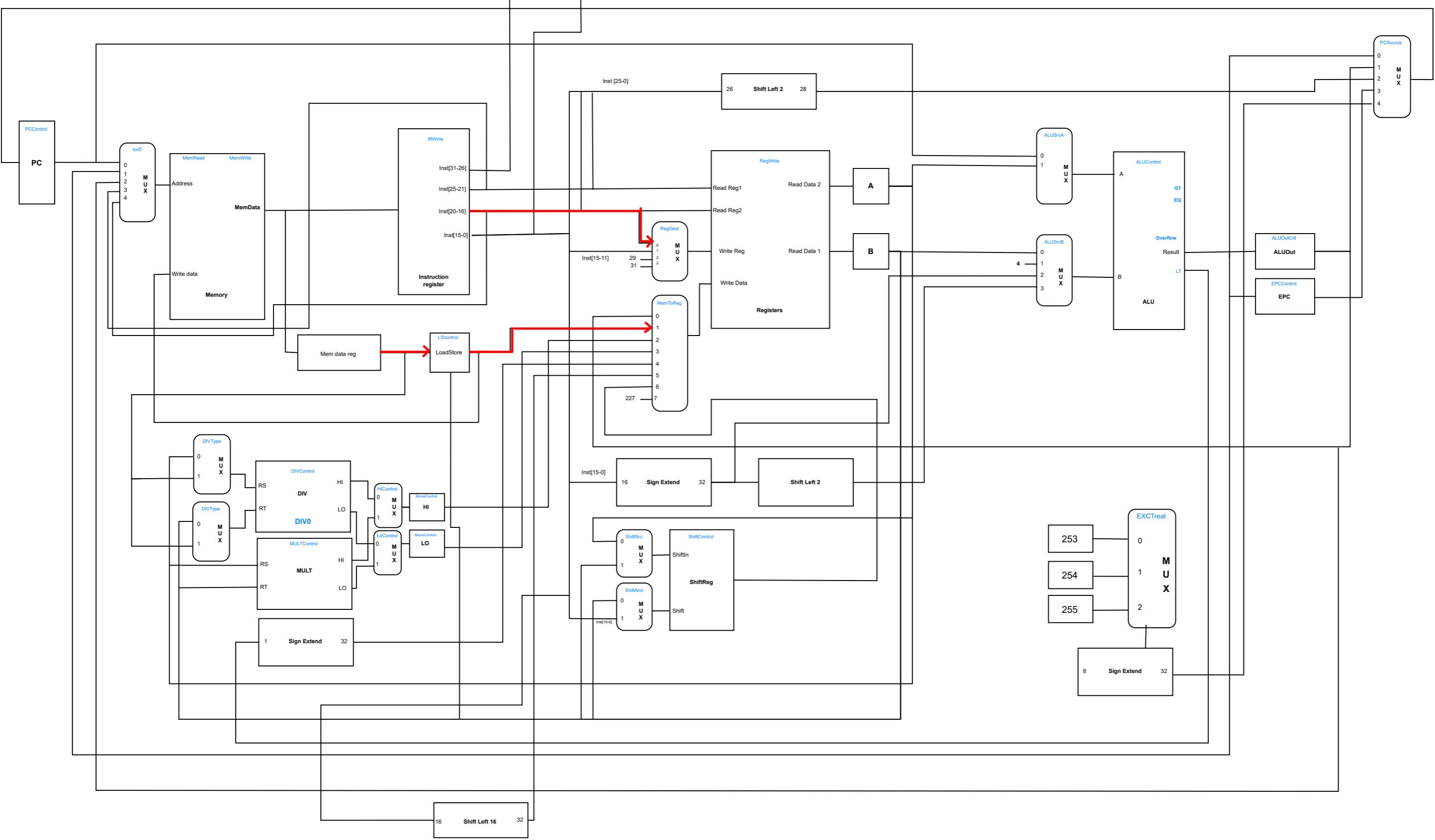
LUI rt, immediato ($rt \leftarrow \text{immediate} \ll 16$)

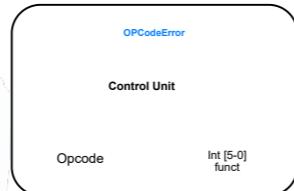






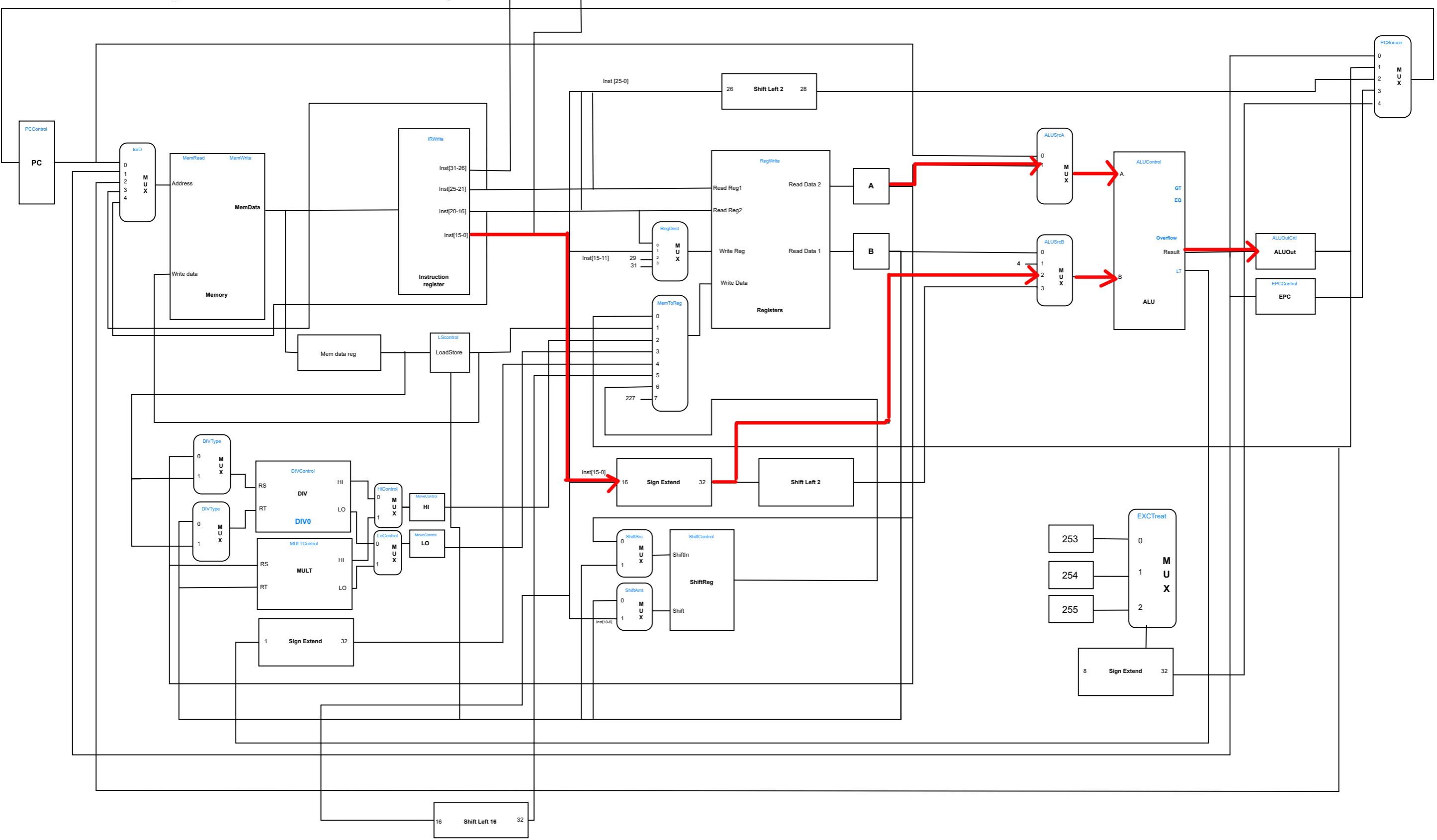
MDR passa por LS para ser registrado em rt

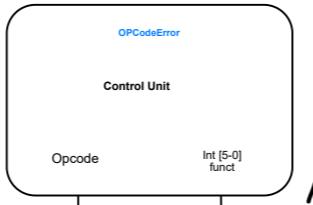




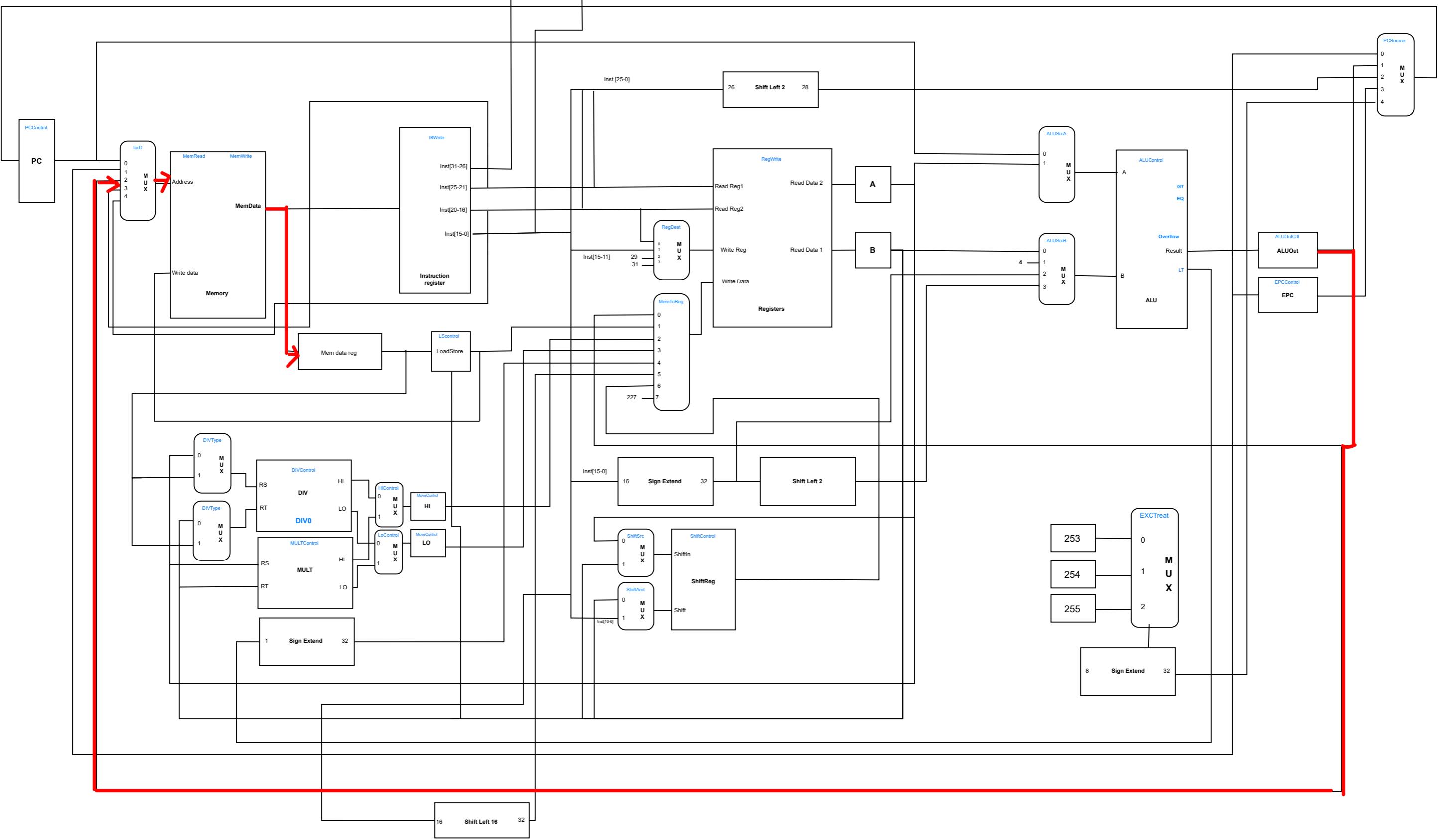
SW rt, offset(rs)
SH rt, offset(rs)
SB rt, offset(rs)

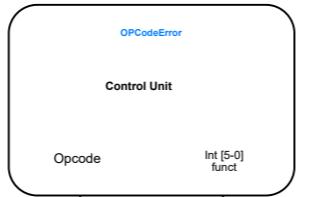
Carrega em ALUout



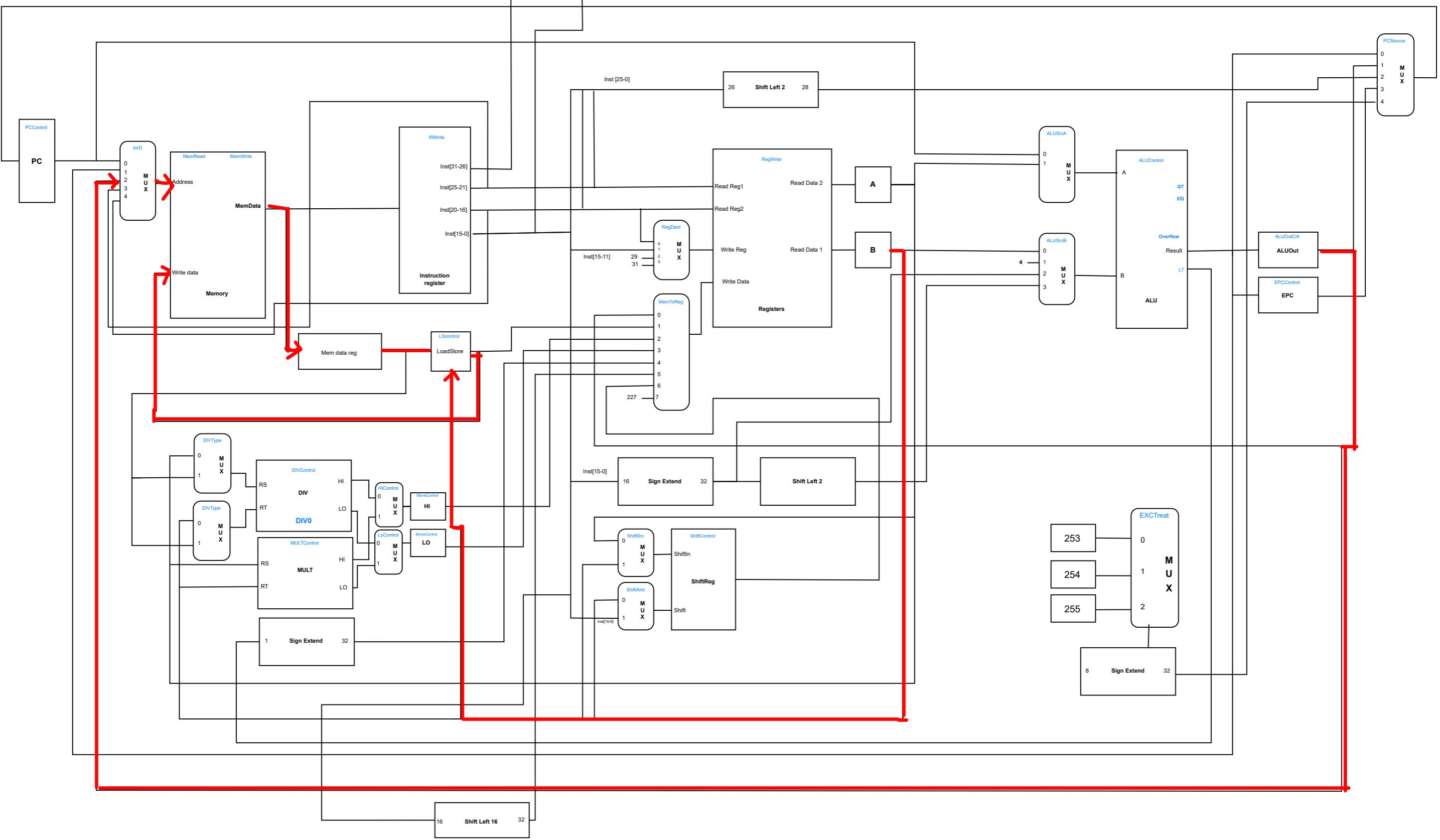


MDR recebe as informações em ALUout

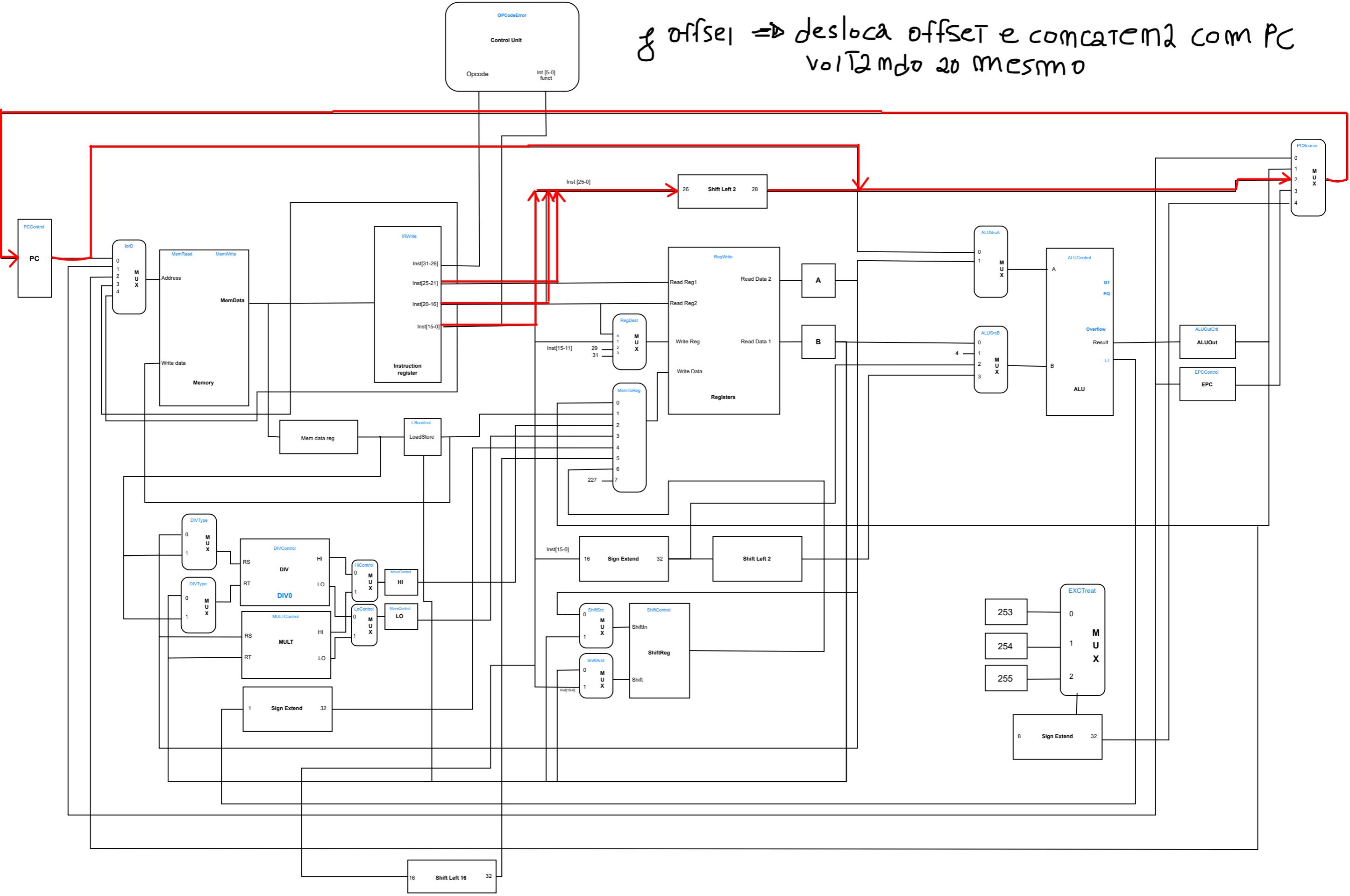




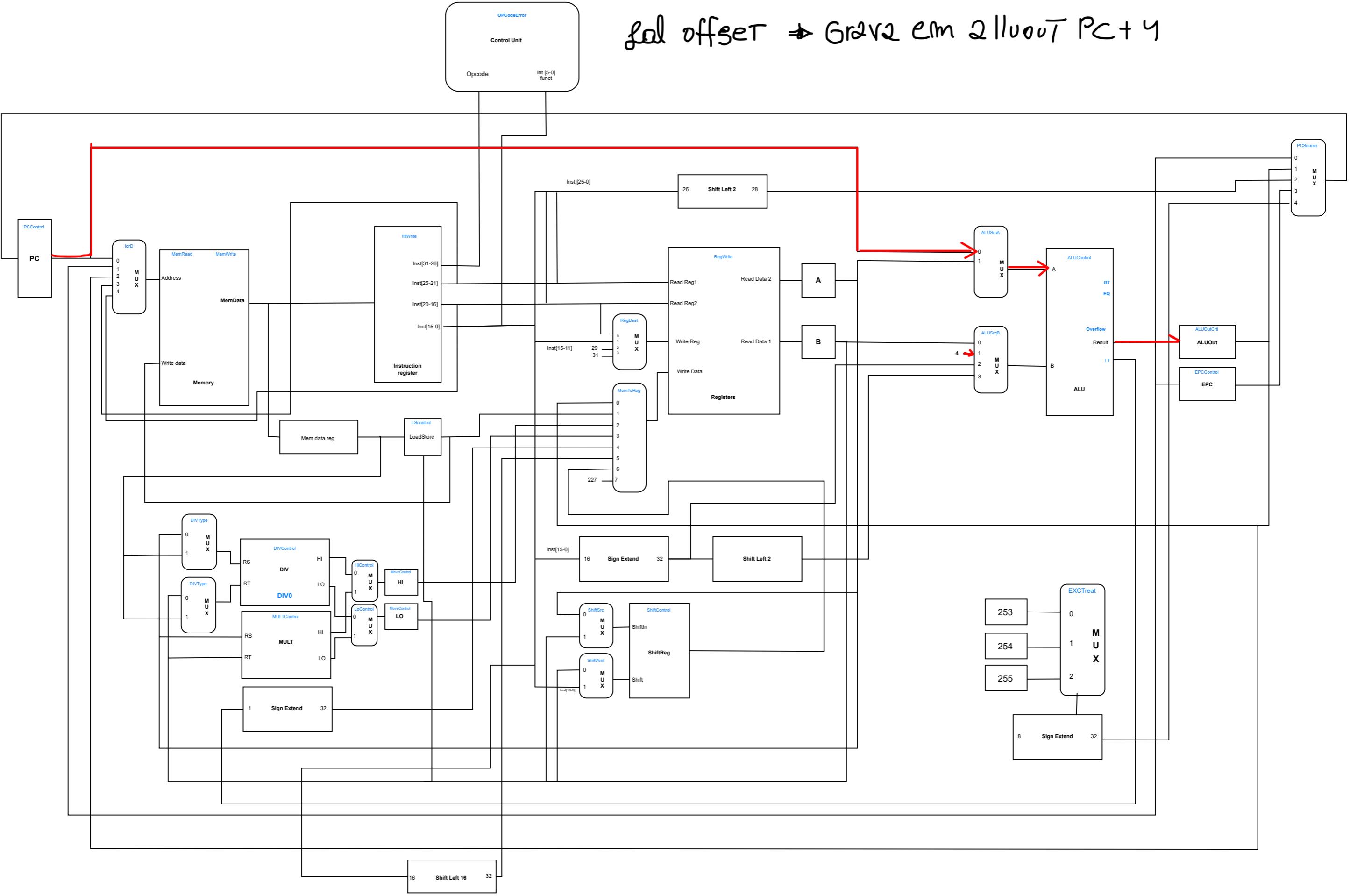
Valor de B passa por LS e, então, é carregado na posição definida por ALUout



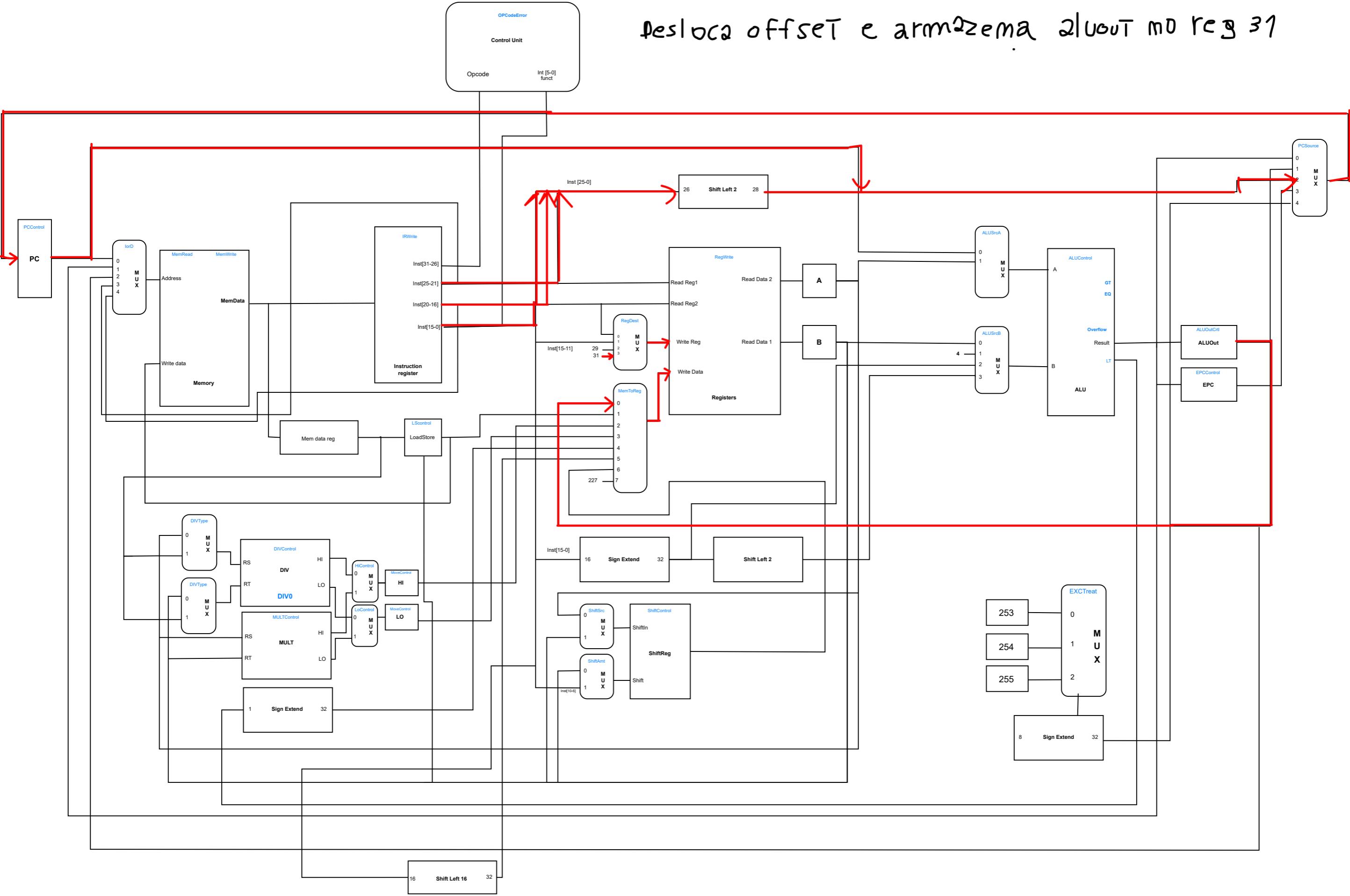
*offset ⇒ desloca offset e concatena com PC
voltando ao mesmo*



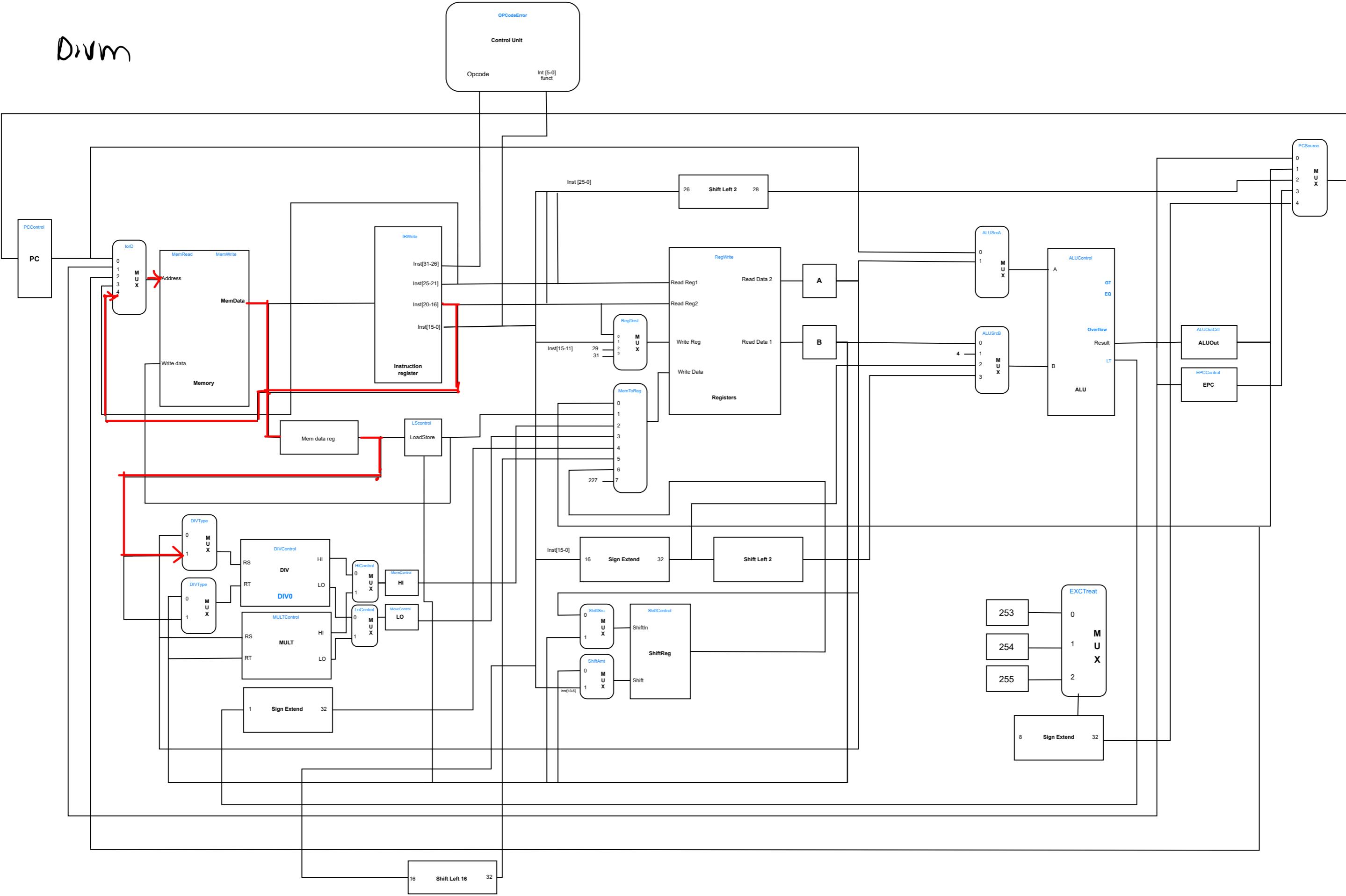
for offset \Rightarrow Gr2V2 em 21100T PC + 4

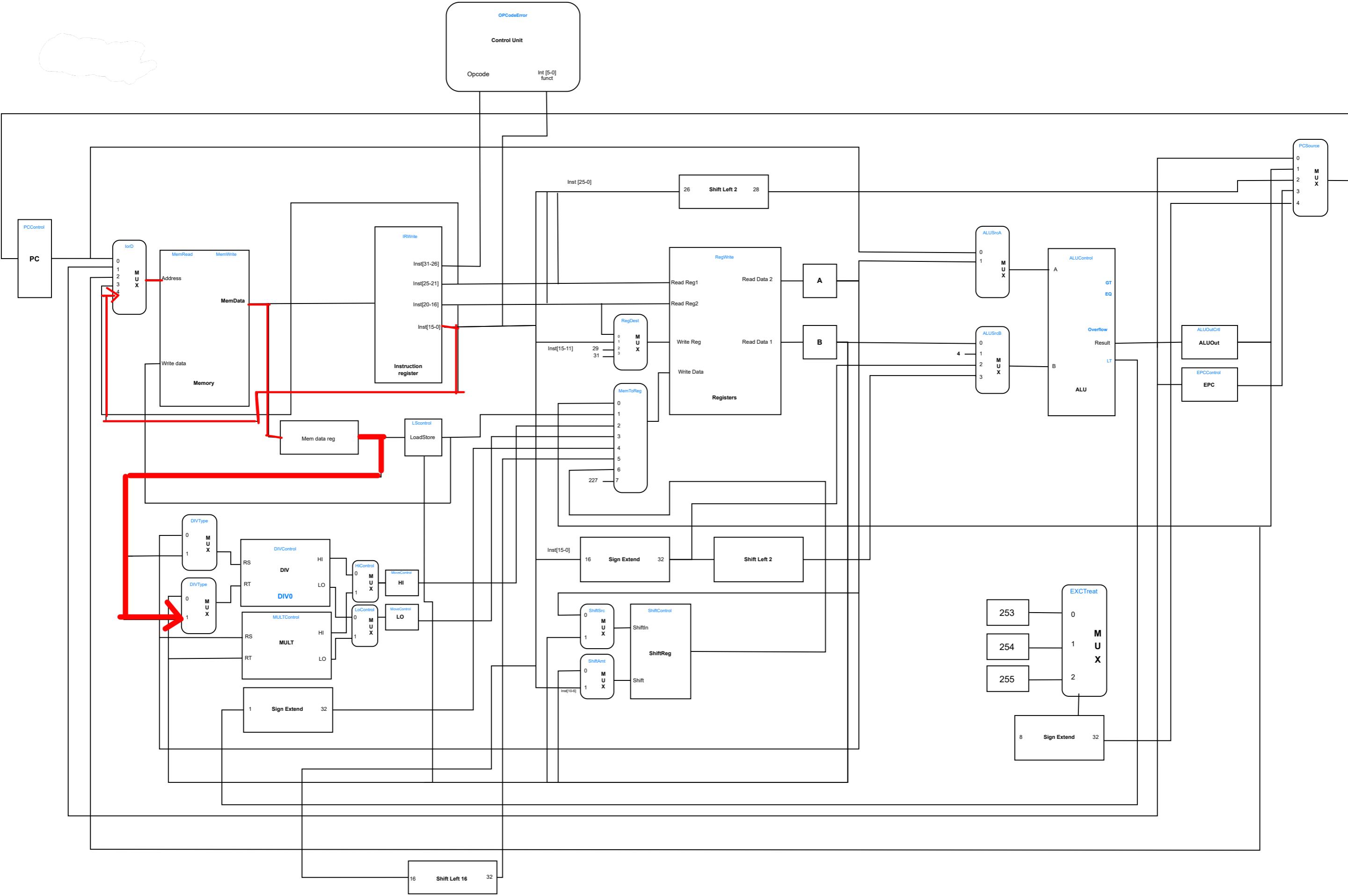


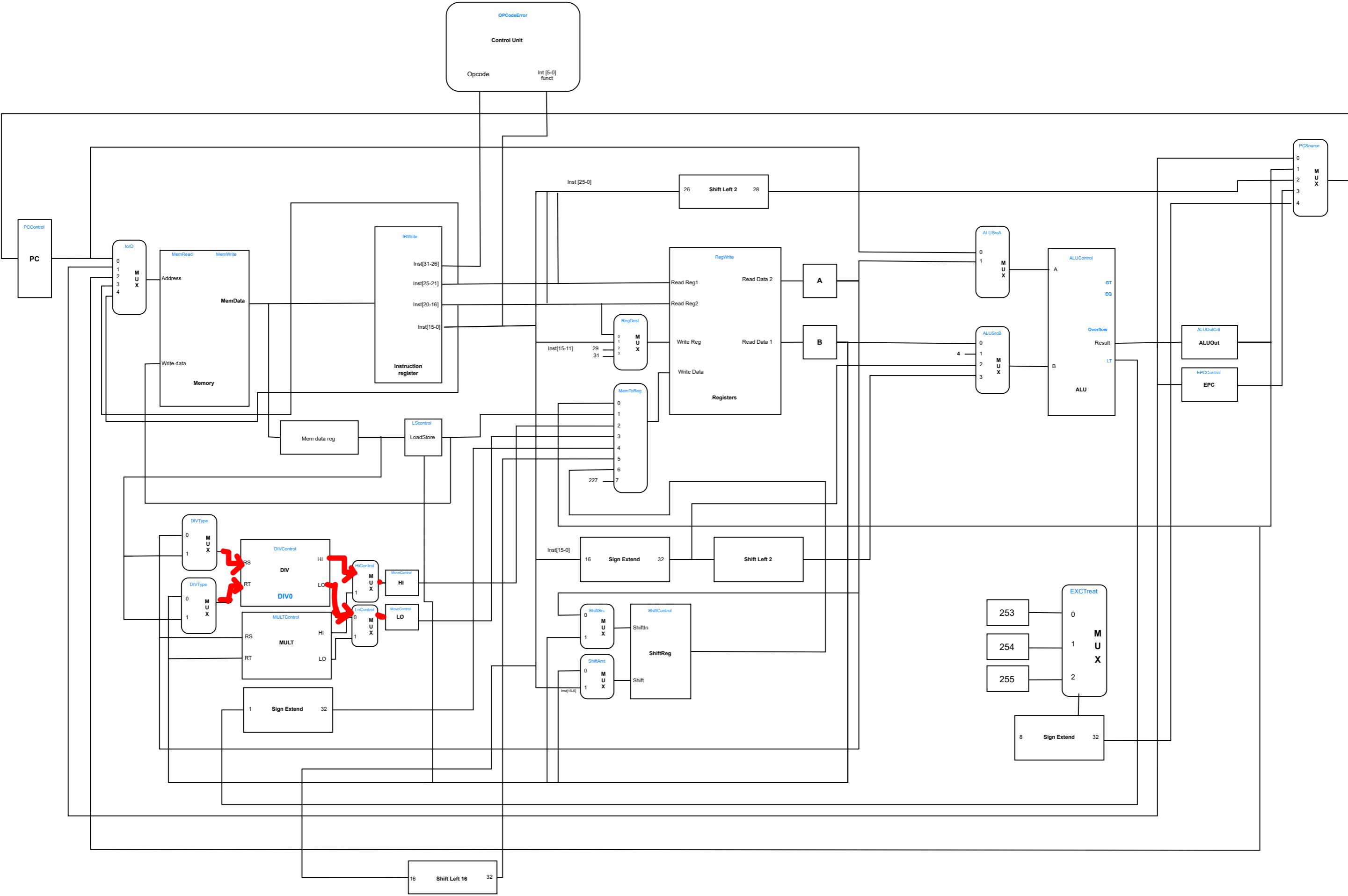
desloca offset e armazena 2100T m0 reg 31



Durm

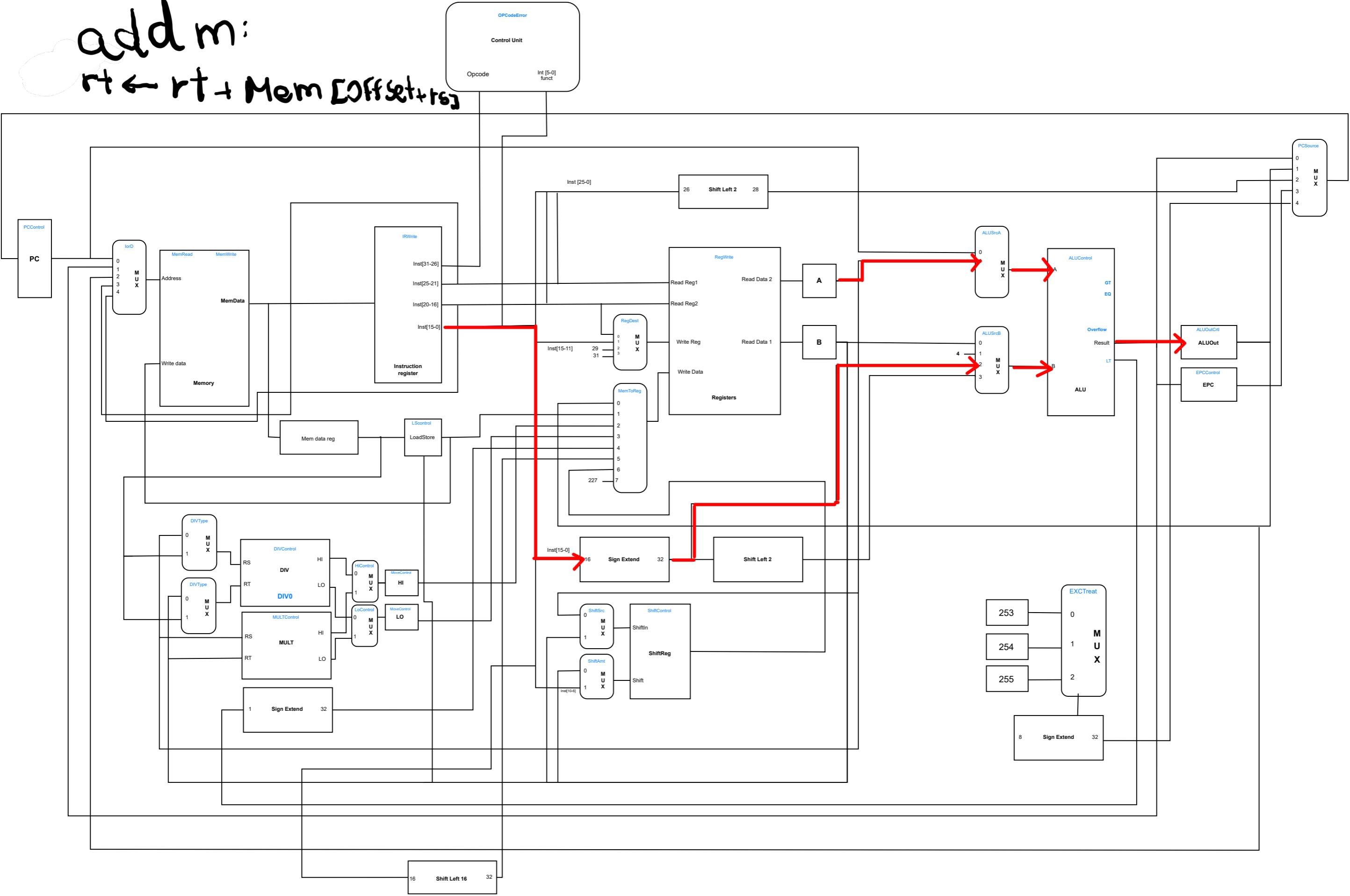


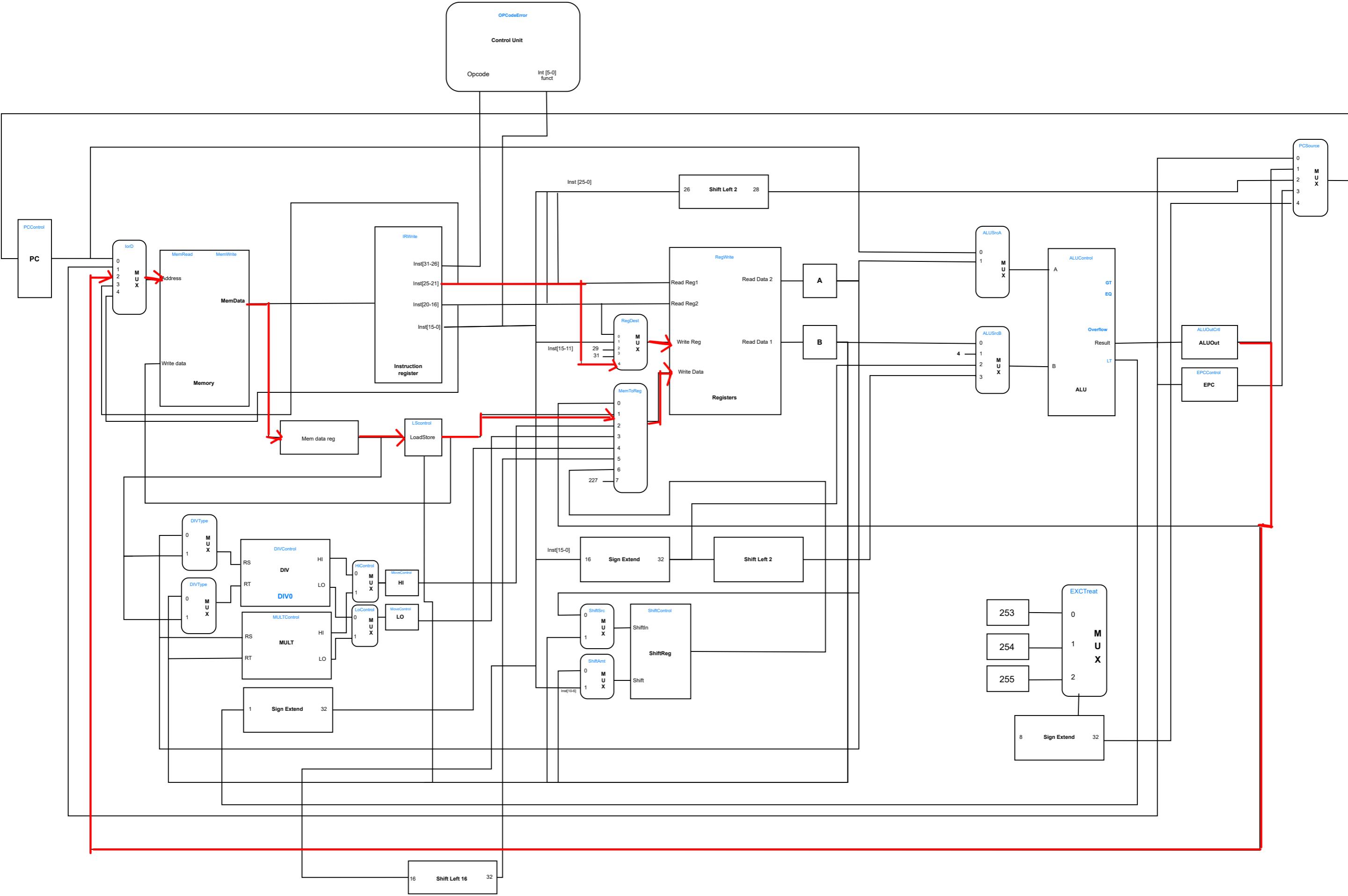


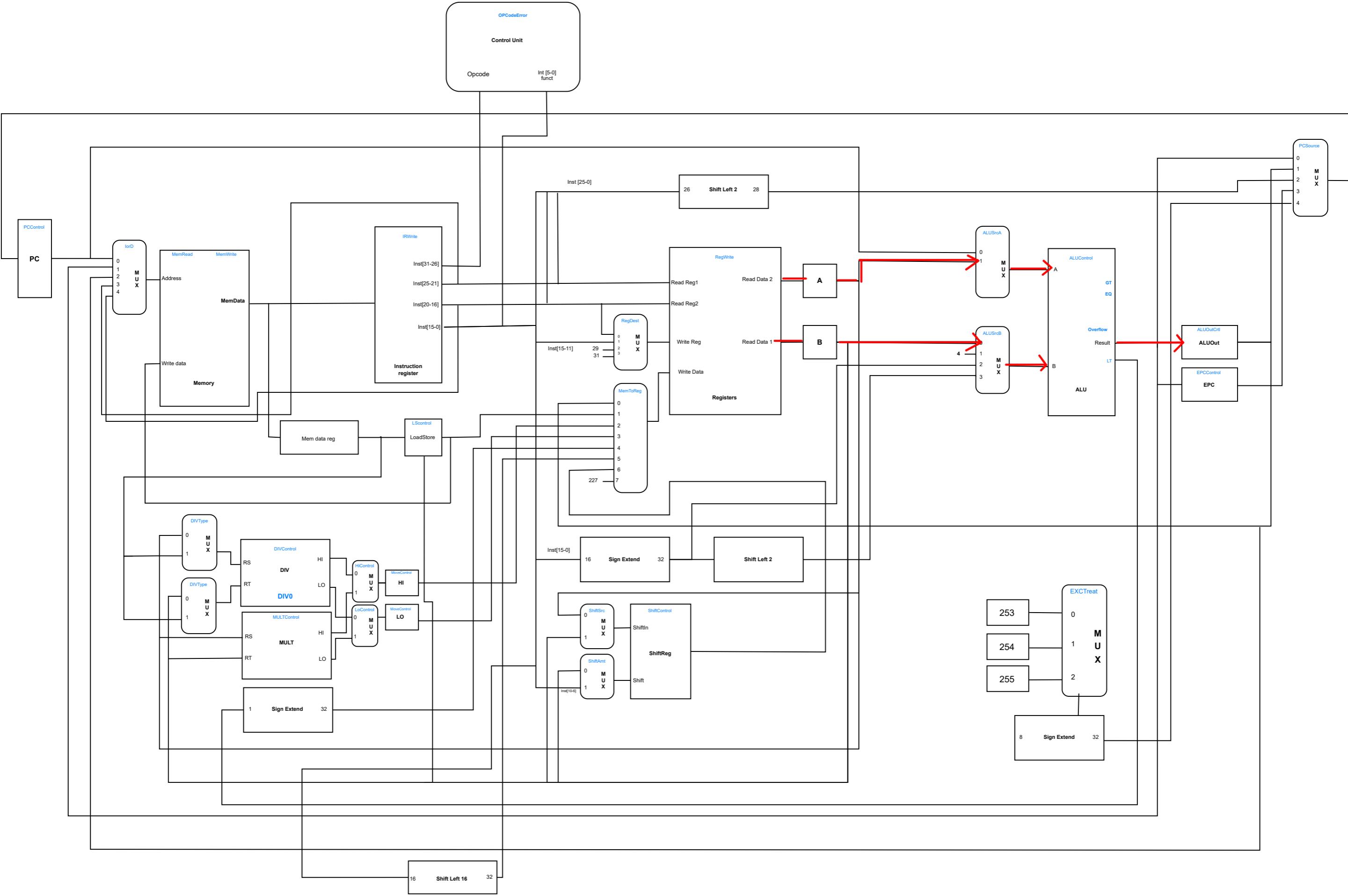


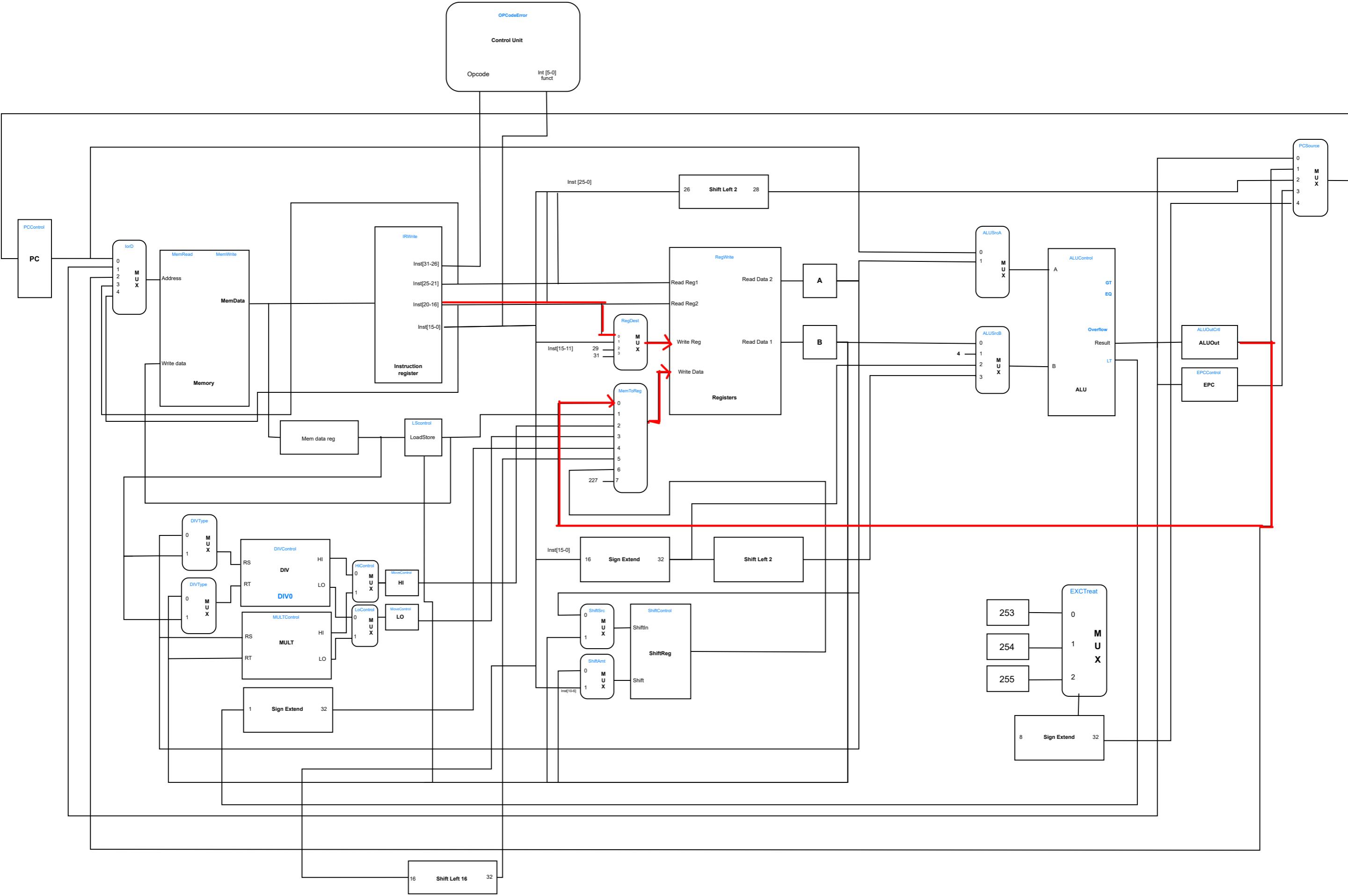
add m:

$r_t \leftarrow r_t + \text{Mem}[\text{Offset}_t + r_{\text{sp}}]$

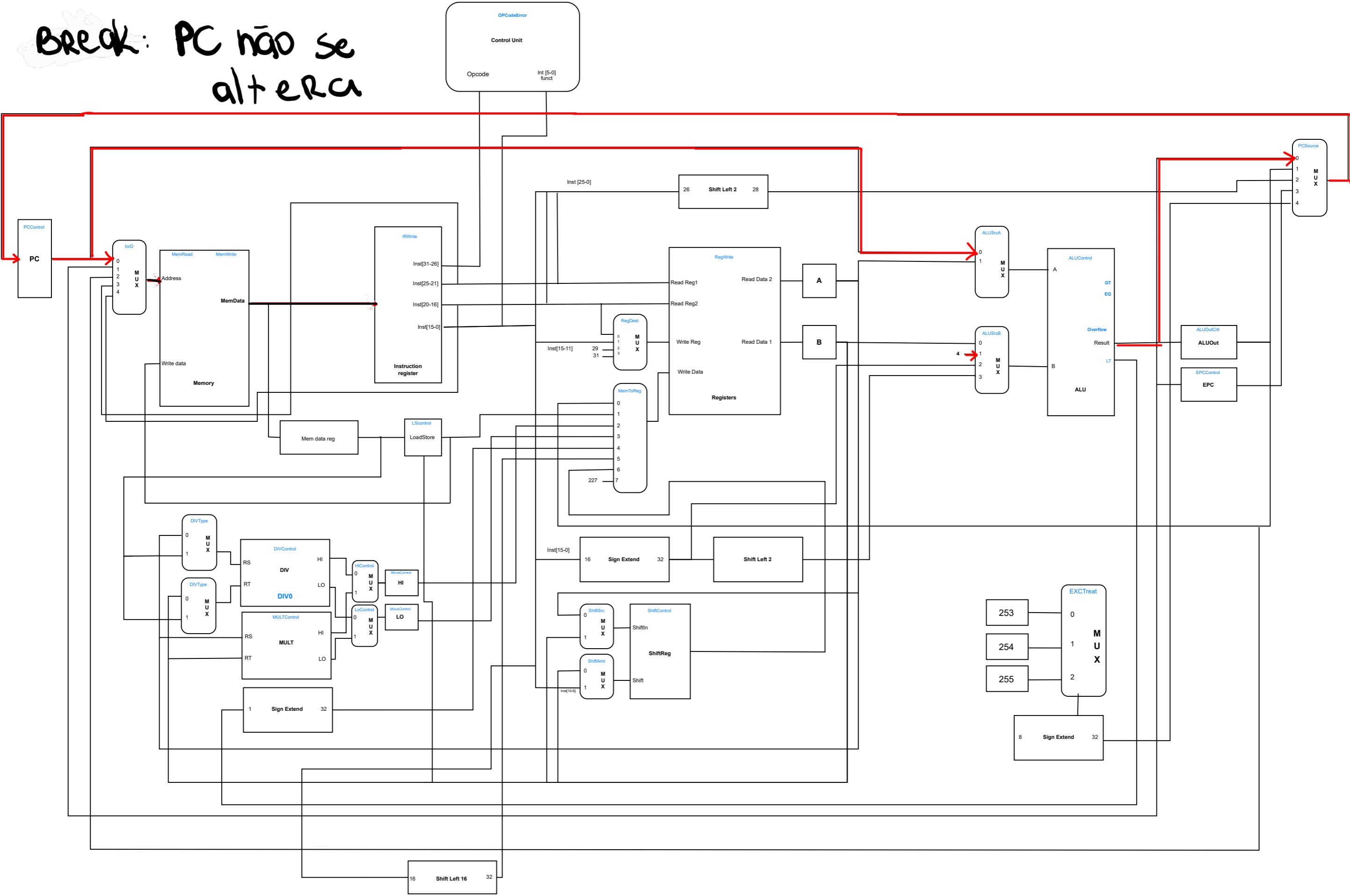




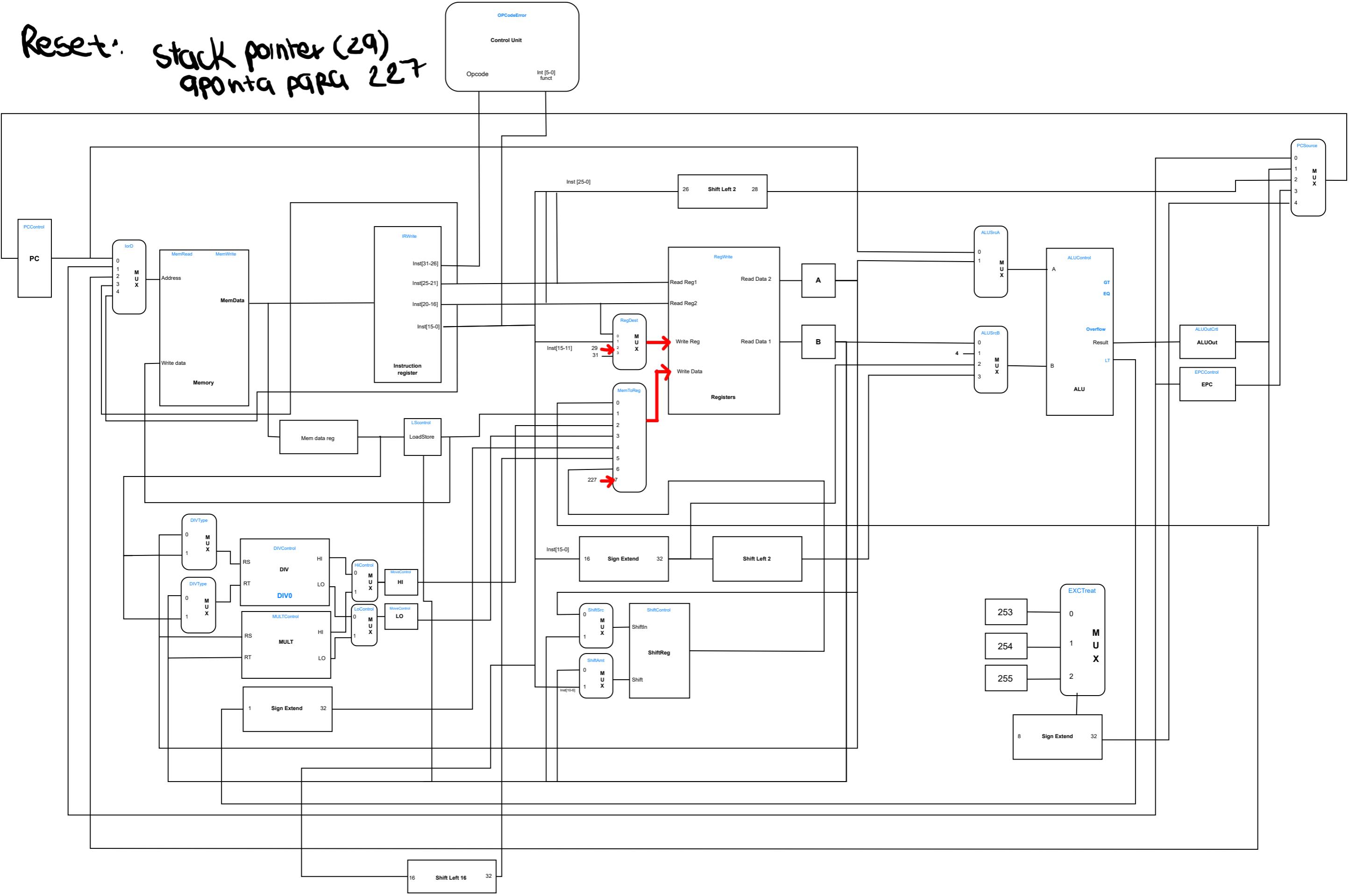




BREAK: PC nāo se alterea



Reset: stack pointer (29)
aponta para 227



EXCEÇÕES.

