6.004 Design Project Implementation Notes

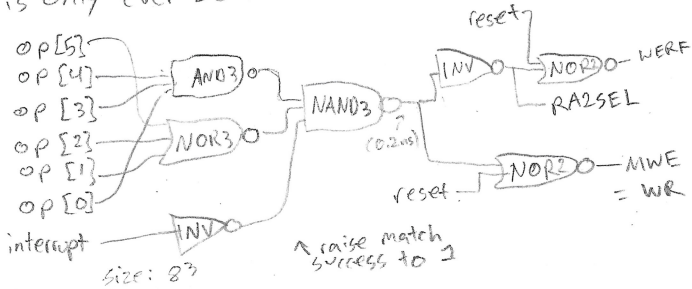
Control logic ROM replacement

WERF is only ever zero for a store instruction (oprode == 011001_5) but is forced to 1 if interrupt is occuring (IRQ and not supervisor) and to zero if reset

MWE is only ever 1 for a store instruction, but forced to 0 on an interrupt or reset

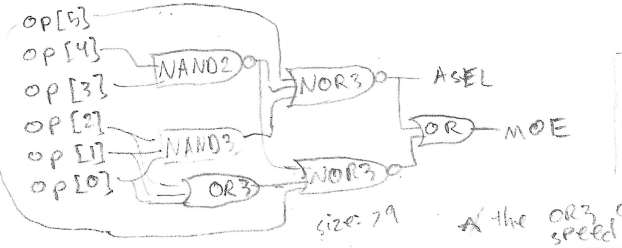
RA2SEL can validly be identical to MWE (but could be computed quicker if it is a bottleneck, though not necessarily much quicker)

so



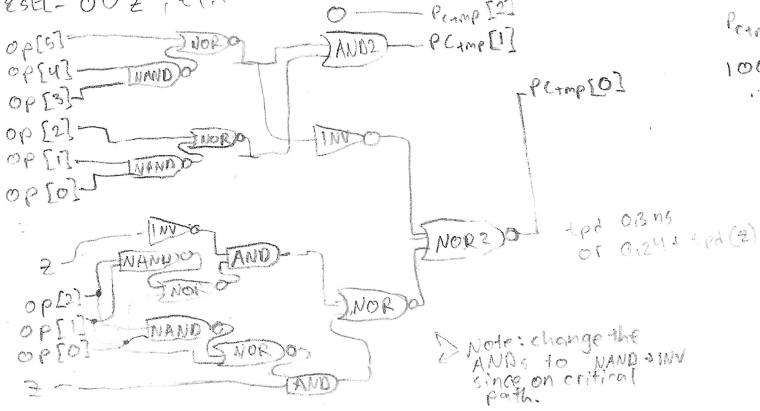
WASEL is only 1 for an interrupt or Illop (if oprode[5:3] in [000, 001, 010] and several others, but might be better to default 1 and switch to zero on a illegal instruction.

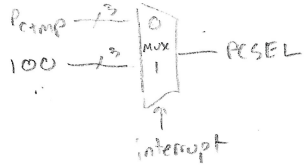
ASEL is only 1 for a LOR istruction (oprode == 011111) and MOE is only 1 for LD or LDR (op[5:3] == 011) and (op[2:0] == 111 or op[2:0] == 00



BSEL is only 0 for an OP instruction (not an OPC). So where oprode[5:3] is either 100 or 101. but not 111. or 110

PCSEL : If interrupt, PCSEL = 100, elif JMP (oprode == 011011) PCSEL = 010, elif BEQ (oprode == 011101) PCSEL = 001, elif BNE (oprode == 011110) PCSEL = 001, else PCSEL = 0





For now, I'm leaving ALUFN[5:0], WASEL, and WDSEL[1:0] to the ROM, but these should be split if pipelining.

Also, could buffer each oprode bit since CTL has lots of load on them.