University of the Saarland
Department 6.2 - Informatik
Prof. Dr. W.J. Paul

Computer Architecture I - WS 02/03

(due: Monday 10.02.2003)

Excercise 1: (Saturating Counter) **(4 points)**

In the lecture we introduced saturating counters. A n bit saturating counter is a circuit with inputs $reset$, inc, dec and output $o[n-1:0]$. The output o is defined by:

$$o = \begin{cases} 0^n : reset = 1 \\ o+1 : inc = 1 \wedge o \neq 1^n \\ o-1 : dec = 1 \wedge inc = 0 \wedge o \neq 0^n \\ o : \text{otherwise} \end{cases}$$

Construct a n bit saturating counter. For this you can use a circuit which implements a standard n bit counter with $reset$, inc and dec inputs.

Excercise 2: (Return Stack Address Computation) **(4 + 2 points)**

In the lecture we used a so called return stack to predict the destination addresses of jumps used to return from functions. To access this return stack we had the two signal *push* (adds a new entry to the top of the stack) and *pop* (removes top entry from the stack). If we push more addresses on the stack than it has entries then of course the oldest ones get overwritten.

- In the lecture we implemented the return stack as a standard two ported RAM. You can use a register called $rs.addr$ as a pointer to the current *top* element of the return stack. How do you have to compute the write and read addresses depending of this register?
- Construct a circuit which contains a register for the return stack address $rs.addr$ and computes depending on the signals *pop* and *push* the new content for the $rs.addr$ register. The circuit should also output the read and write addresses $rs.r.a$ and $rs.w.a$ of the return stack (the positions in the stack where read and write accesses take place).

Excercise 2: (Optimized Forwarding Circuit) **(3 + 5 points)**

In figure 1 you see the original forwarding circuit we used in the lecture. The Or(5) is used to detect accesses to register R0. Instead of using the result of the Or(5) to compute the hit signals we can use it as input for a 32 bit AND circuit (see figure 2).

- Compute cost and delay of the new forwarding circuit and compare it with the original one.
- With the new forwarding circuit it is possible to change the realization of the GPR environment. Especially one can save some hardware used only for register R0. What changes in the GPR environment are possible? Compare cost and delay of the modified GPR environment with the old one.

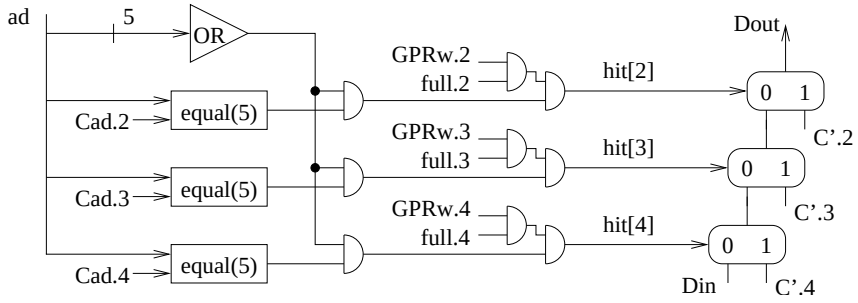

Figure 1: Original Forwarding Circuit

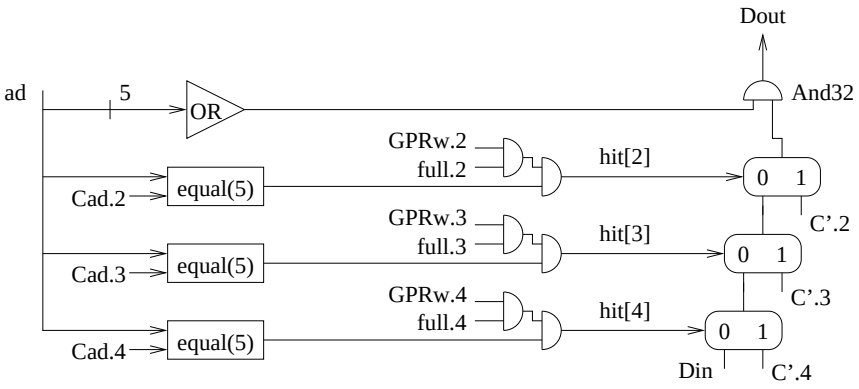

Figure 2: Modified Forwarding Circuit