Saarland University
Department 6.2 – Computer Science
Prof. Dr. W. J. Paul
M. Sc. Petro Lutsyk

Computer Architecture – WS14/15

Exercise Sheet 11 (due: 27.01.15, 36 points)

Exercise 1: (snoop conflict) (12 points)

In the lecture it was suggested to change from the "conservative" definition of the $snoopconflict$ predicate (used in the script) to the following one

$$snoopconflict \equiv \neg sidle \wedge pa = badin$$

Argue whether this change is reasonable. For that

(a) explain how the new definition can affect performance of the cache system.

(b) prove or disprove (using lemmas learned so far) that the new definition breaks the correctness proof of the shared memory construction.

Exercise 2: (data paths) (12 points)

Recall the *data paths* for the data, tag and state RAMs used in the construction of the cache system. Using the lecture notes (or the corresponding sections of the script) repeat the following arguments.

(a) Explain what we need the auxiliary registers for.

(b) Explain why we use the register $souta'$ on transition $wait \rightarrow m0$, and why not on $flush \rightarrow m0$.

(c) Explain why we forward state I to the inputs of circuit $C1$.

(d) Explain why we forward state M to the inputs of circuit $C2$.

Exercise 3: (bus arbitration) (12 points)

Recall the design of the *master* and *slave* control automata of the cache. Also recall the construction of the *bus arbiter*. Observe that the master automaton keeps its request active in every state of the *hot* phase. Together with the delayed (!) generation of the *grant* signal, this leads to situation when the master being in state *idle* still retains ownership of the bus.

(a) Show that the prescribed is indeed happening. Suggest how to change the control to obtain

$$idle(i)^t \rightarrow \neg grant[i]^t$$

(b) Try to prove that the change from (a) does not affect the overall correctness. In case it does, explain how to modify the design further to make the proof work.