Universität des Saarlandes
FR 6.2 - Informatik
Prof. Dr. W.J. Paul
Dipl.-Ing. Christoph Baumann

Exercise Sheet 9
Computer Architecture II
(Due: Jan 7th, 2014)

Exercise 1: (Non-Interference Properties) **(3+3+3+3 Points)**

In the lecture we used a number of properties that allowed to commute disk steps accross processor steps which do not access the hard disk. Prove the following statements for "ISA with hard disk"-configurations d, e and external event vector eev!

a) Disk steps do not change the processor part of the configuration.

$$proc(\eta(d)) = proc(d)$$

b) The effect of processor steps not accessing the disk depends only on the processor part of the configuration.

$$/hdacc(d) \wedge proc(d) = proc(e) \implies proc(\delta(d, eev)) = proc(\delta(e, eev))$$

c) Processor steps which do not access the disk, do not change the disk part of configuration.

$$/hdacc(d) \implies hd(\delta(d, eev)) = hd(d)$$

d) Disk steps only depend on the disk state.

$$hd(d) = hd(e) \implies hd(\eta(d)) = hd(\eta(e))$$

Exercise 2: (Hard Disk Sending Interrupts) **(6+8+6+8 Points)**

So far the only communication channel between the processor and the hard disk are the $hd{-}ports$ in memory. The processor must poll the command-and-status register until the disk finishes handling the last read or write request and sets the $cmsr$ to $idle$ state. Now we want to consider an alternative model where the disk sends an external interrupt to the processor to signal the end of the requested operation. The $cmsr$ is here not changed by the hard disk. Instead $eev[1]$ is raised and kept up until the processor acknowledges the interrupt by writing 00 into the $cmsr$. After that the disk will eventually lower $eev[1]$ again. Note, that the processor may only set the disk back to $idle$ in this way, if the disk signaled that it is idle via $eev[1]$ beforehand. Note also that the processor may ignore the disk for a while by masking external interrupts in the status register.

a) Adapt the hardware and ISA models from the lecture to the new situation where the disk may send interrupts! Also update the operating conditions for both simulation layers!

b) State and prove an adapted simulation theorem between the new ISA and hardware models!

c) Since the processor can mask responses from the disk, we only need to consider disk steps when external interrupts are enabled. In fact we can reorder the disk steps to occur directly after the processor steps which enable interrupts. Formulate a new reordering theorem according to this idea!

d) Prove your new reordering theorem! To this end first try to replay the old reordering proof and adapt it to the new situation where it is necessary. You may have to modify the properties of Exercise 1 or find new ones.