Vorlesungsprüfung aus

Hardware Modeling

15.04.2021

You have 1 hour to finish the exam. During the exam you are only allowed, without exception, to use a writing device, a ruler, a (non programmable) calculator and the distributed cheatsheet. Write your answers and solutions (including how you derived them) with a pen (not red, NO pencil) and strike everything out that you do not want to be graded. Aborting the exam after receiving the assignment will lead in every case to an evaluation.

Enter your name, study code and student ID right at the beginning of the exam in the provided table and mark every additional page, that you want to submit, with your name and student ID in the top right corner.

Good Luck!

family name:	first name:
study code:	student ID:

Do not label this table.

Example:	1	2	3	4	5	6	7	8	9	10	**Overall**
possible points:	3	9	6	12	6	9	12	6	9	24	96
received points:											

1. (3 points) Name the purpose of the **architecture** in VHDL.

2. (9 points) Draw the hardware implementation of the following code piece. What are the biggest differences compared to an execution in software?

```
for (i=0; i<n; i++){
  x *= a;
}

return x;
```

3. (6 points) Which design style was used in **architecture** `beh`? Transform it to the `sequential design style`!

```
architecture beh of test is
begin

  F <= D and E;

end architecture;
```

4. (12 points) Explain the `all` keyword in the `sensitivity list`! Derive an equivalent (explicit) `sensitivity list` for the shown example. Finally develop an equivalent representation by using a `wait` statement instead!

```
architecture beh of test is
  signal A, B, C, D, E : STD_LOGIC;
begin

  process (all)
     variable F : STD_LOGIC;
  begin

     F := A xor B;
     D <= A and B;
     E <= A or B or F;

  end process;

end architecture;
```

5. (6 points) Name the effect of the keyword **open** in the shown code piece! What properties for port `B` have to be fulfilled for this to work? Name a common situation when this keyword can be used!

```
uut: test
port map(
  A => A,
  B => open
);
```

6. (9 points) Describe the bottom-up approach in hardware verification! Name at least two advantages!

7. (12 points) Name at least two possible causes of delay! Explain the delay models pure, inertial and Involution delay!

8. (6 points) Explain physically aware synthesis! What is the difference to regular synthesis?

9. (9 points) Explain the optimization step `operator resource sharing` by providing an example! Which constraints must be adhered such that this optimization is possible?

10. Consider the following **entity** (all signals are high-active):

```
library ieee;
use     ieee.std_logic_1164.all;

entity statemachine is
  port
  (
    clk          : in std_logic;
    res_n        : in std_logic;
    button_down  : in std_logic;
    button_up    : in std_logic;
    down         : in std_logic;
    up           : in std_logic;
    fire         : in std_logic;
    motor_down   : out std_logic;
    motor_up     : out std_logic
  );
end entity statemachine;
```

Design a Moore state machine which implements an elevator control circuit according to the following specification:

As long as `button_down` is being pressed, the motor shall run (`motor_down`) until the end position `down` is reached. In the same fashion `button_up`, `motor_up` und `up` should work. When the button is released the elevator should stop. Are both buttons pressed only the button press which happened earlier in time shall be considered. You can assume that inputs `button_down` and `button_up` do not change their value in the same clock cycle. When there is a `fire`, the elevator shall move all the way down and shall not move up, even if `button_up` is pressed.

(a) (18 points) Create a state transition and an output table! Specify in the header the signal names concatenated by & as shown in the lecture. This allows you to write the logic values as a vector, e.g., as 10000! You may use don't care symbols ('-') for arbitrary values on a signal or "otherwise" to denote all missing cases.

(b) (6 points) Draw the state diagram including the transition conditions, the output values and the initial state (double frame).