IIT JODHPUR

Minor-2 Examination (OPEN-BOOK): EEL7770 Formal Verification

Guidelines (Total time: 60 minutes, Maximum Marks: 15):

- Please read the question paper very carefully (both sides of this paper). **NO clarification is required in any question.** In case of any doubt, assume whatever you wish to and state that in your answer.
- This is an OPEN-BOOK examination. However, usage of internet is NOT allowed.
- Please adopt mathematical reasoning as far as possible in your answers.

---

1. FIFO (First-In-First-Out) are widely used in the modern designs for data transfer. They are also utilized in implementing memories when arranged in BRAM (Block random access memory) fashion. BRAMs are a very essential component while mapping the designs on to Field-programmable gate arrays (FPGA) boards. Consider the following diagram of BRAM where the input and output ports are clearly shown (The size of the FIFO is parametrized by the generic parameter ITEMS. Items in the FIFO are divided into blocks and the design allows detection of the situation when only the last free block of the FIFO is available. The number of items in one block is set by the BLOCK_SIZE constant.):

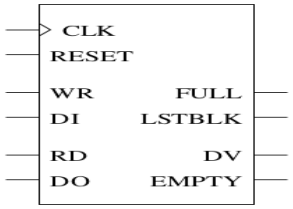

Figure 1: BRAM Diagram

   - There are the following four regular input ports in the component interface:
     - CLK is a port of the Clock signal
     - RESET is a port of the Reset signal
     - WR is the Write Request port
     - RD is the Read Request port
   - The data ports of the component are the following:
     - DO is 16bit Data Output port
     - DI is 16bit Data Input port
   - The output ports are defined as follows:
     - EMPTY transmits the signal which is active if and only if the FIFO is empty.
     - FULL transmits the signal which is active if and only if the FIFO is full.
     - LSTBLK transmits the last block detection signal which is active if and only if the FIFO contains just BLOCK_SIZE or less than BLOCK_SIZE number of free items.
     - DV transmits the data valid signal which is active as soon as the valid output data are available on the DO port

   Write sufficient properties (in terms of above signals) with LTL and CTL operators to completely verify the behavior of above BRAM (for example-no overflow/underflow, correct data transfer etc.). If you are comfortable with System Verilog format, you may write the assertions in System Verilog as well. [6]

2. Modern designs utilize multiple clock domains (i.e., frequencies) to achieve higher performance and low power dissipation. However, timing problems may arise around the logic regions when there is crossing of clock from one frequency to another. This is referred to as CDC (clock domain crossing) in the

industrial flows. We need to verify these crossings to ensure that we have data integrity (i.e., no data is lost). Following figure is a high level representation (so, no signal names are mentioned) of CDC showing transition from fast clock to slow clock as we move from left to right (i.e., input to output). Write down sufficient assertion(s) to carry out this verification. Assume d_in (input data), d_out (output data) and other important signals as per your understanding/choice. (Here synchronization is achieved with the help of edge-triggered flip-flops and cloud-like structures represent the combinational logic). [4]

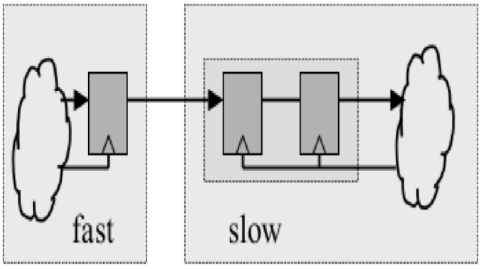

Figure 2: Clock Domain Crossing Illustration

Hint: In the diagram above, please take signals as per the instructions in the question and your choice, then argue about the overall functionality that can be translated into assertions/properties.

3. Mr. KLMN is a senior engineering manager in one of the top semiconductor companies. He has the responsibility of analyzing products that are returned back from different customers (because of various reasons) and preparing lessons so that such mistakes are not repeated in any of the future products. During the analysis of a product (chip) returned from a customer, he observed that one small module in the design was sometimes behaving wrongly and needed further analysis. He found the suspect verilog code as twelve lines that are shown below:

```
input i1,i2, i3, clk;
wire i11, i12;
reg memo1,memo2;
output o1, o2;
i11 = i1 and memo1;
i12 = i2 nand i3;
always @(posedge clk) begin
memo1 <= i11;
memo2 <= i12;
end
o1 = i2 or memo1;
o2 = i1 nor memo2;
```

Figure 3: Small Verilog Code Snippet

Use any FV technique such as model checking to find out if it is possible to obtain the following output values: 00, 01, 11, 10? Give formal arguments as to how/why these values can/can not be reached? [4]

4. What have you learnt in the course so far? Please do not mention the course contents, rather explain (in one or two statements) what skills/new knowledge you have gained so far from this course? [1]
