Zero-counting Circuit

Signature and Grading Sheet

Group #:_____ Name(s):_____ .

Signature

Section 4.2(c): _____.

Grading

- Section 4.1(a)(b): pseudo algorithm and registers (10 points): .
  Attach the 1-page algorithm and explanation use of registers
- Section 4.1(c): ASMD chart (20 points):_____.
  Attach 1-page detailed ASMD chart
- Section 4.1(d): VHDL code (25 points):_____.
  Attach code printout
- Section 4.1(e): simulation (20 points):_____.
  Attach two screen captures
- Section 4.2(b): VHDL code (5 points):_____.
  Attach code printout
- Section 4.2 (c) : demo signature (20 points):_____.

Total points: .

Experiment
Zero-counting Circuit

1 Purpose

To use FSMD methodology to design and implement an intermediate-sized digital circuit.

2 Reading

Chapter 6 of *FPGA Prototyping by VHDL Examples 2nd edition.*

3 Project specification

The purpose of this project is to construct a circuit that counts number of 0's of an 8-bit input word. For example, the output returns "0011" if the input is "11001110". In addition to the clock and reset signals, the input signals of this circuit are

- start: one-bit command to initialize the counting
- a: 8-bit input data word

The circuit outputs include the following:

- ready: one-bit status indicating that the circuit is ready
- count: 4-bit result showing the number of 0's in a.

The design must be synchronous or 50% will be deducted.

4 Design Procedures

4.1 FSMD design and simulation

(a) Derive a pseudo algorithm.

(b) Determine the registers needed.

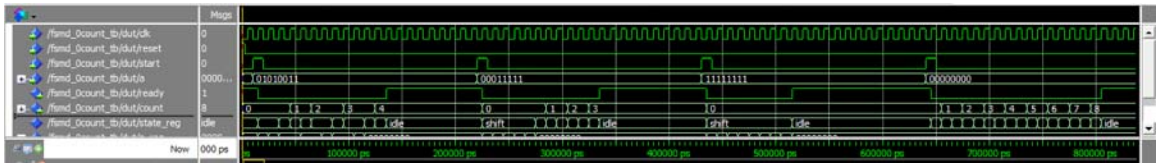
(c) Derive the detailed ASMD chart according to the algorithm.

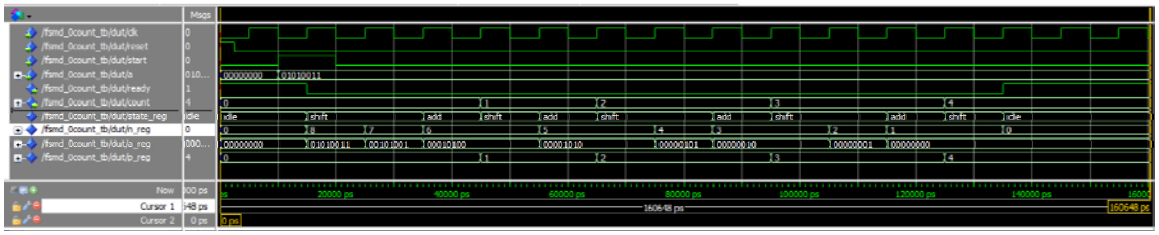
(d) Derive VHDL code according to the ASMD chart. The entity declaration of this design is

```
entity fsmd_0count is
   port(
      clk, reset: in std_logic;
      start: in std_logic;
      a: in std_logic_vector(7 downto 0);
      ready: out std_logic;
      count: out std_logic_vector(3 downto 0)
   );
end fsmd_0count;
```

Derive the architecture body.

(e) Use the testbench (fsmd_0count_tb.vhd) to simulate your VHDL code. Perform two screen captures. The first one should show all 4 test patterns and the results. The second one should expand the operation of one test pattern to make state and internal register values visible. Use proper number formats (unsigned or binary) for these signals.





4.2 Implementation and testing

(a) Use 10 switches for the reset, start, and a signals and 5 LEDs for the read and count signals.

(b) Derive the top-level wrapping VHDL code.

(c) Synthesize the code, program the FPGA device, and verify the operation of the physical circuit. Demonstrate the circuit to the instructor and get a signature.