Name: ______________________________

CS 4/55111 **Final Exam** VLSI Design

Monday 11 May 1998

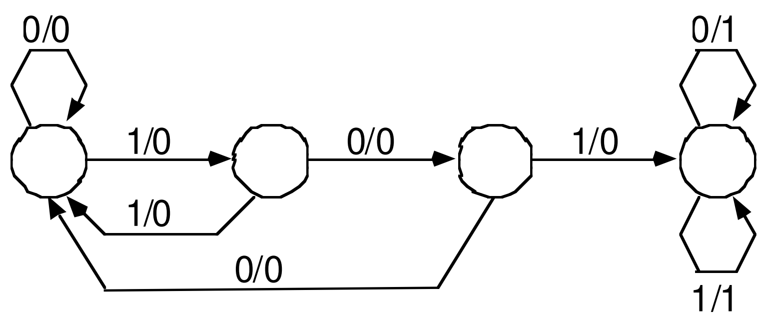
1. (10 points) Which is cheaper, a standard-cell-based or an FPLD-based implementation of a design? Explain your answer.

It depends on the quantity of chips needed. In general, standard-cell-based design has a very high fixed cost and a very low variable cost (cost per unit), whereas FPLD-based design has a low fixed cost and a higher variable cost. Smith gives numbers of $146,000 and $8 for standard-cell-based design, and $21,800 and $39 for FPLD based design. Thus FPLD design is cheaper for up to some breakpoint (using Smith's numbers, 4000 chip), and then standard-cell-based design is cheaper. Of course, these numbers vary, and the fixed cost may be lower if your department already has the necessary tools for that type of design, but in general, these conclusions still hold.

2. (15 points) Briefly explain the difference between a VHDL entity and a VHDL architecture, and also explain why there might be a need for multiple architectures that correspond to a single entity.

A VHDL entity defines a component's inputs and outputs and their types; an architecture defines an entity's implementation. Different architectures may be defined for the same entity to describe the design in different ways (e.g., behavioral description, dataflow description, structural description), at different levels of abstraction (e.g., process level, register-transfer level, gate level), or for different purposes (e.g., synthesis input, simulation behavior).

3. (25 points) The state diagram below defines a state machine to recognize the bit pattern "101". If the input is named "bit" and the output is named "correct", a label of the form x/y indicates that the state transition is made when "bit" has the value "x", and that "y" is output on "correct".



In the space below, write AHDL code for this state machine. Every detail of the syntax need not be correct, but the code should be as complete as possible.

Name: ________________________________

```
TITLE "Bit Pattern Recognizer";

SUBDESIGN finalexam
(
        clk, bit        : INPUT;
        correct         : OUTPUT;
)

VARIABLE
        sm: MACHINE
                WITH STATES (s0,s1,s2,s3);

BEGIN
        sm.clk = clk;

        CASE sm IS
                WHEN s0 =>
                        correct = GND;
                        IF bit==1 THEN
                                sm = s1;
                        ELSE
                                sm = s0;
                        END IF;

                WHEN s1 =>
                        correct = GND;
                        IF bit==0 THEN
                                sm = s2;
                        ELSE
                                sm = s0;
                        END IF;

                WHEN s2 =>
                        correct = GND;
                        IF bit==1 THEN
                                sm = s3;
                        ELSE
                                sm = s0;
                        END IF;

                WHEN s3 =>
                        correct = VCC;
                        sm = s3;
        END CASE;
END;
```

```
TITLE "Bit Pattern Recognizer";

SUBDESIGN finalexam2
(
        clk, bit        : INPUT;
        correct         : OUTPUT;
)

VARIABLE
        sm: MACHINE
                WITH STATES (s0,s1,s2,s3);

BEGIN
        sm.clk = clk;

        TABLE
                sm, bit  =>  sm, correct;

                s0, 1    =>  s1, GND;
                s0, 0    =>  s0, GND;
                s1, 0    =>  s2, GND;
                s1, 1    =>  s0, GND;
                s2, 1    =>  s3, GND;
                s2, 0    =>  s0, GND;
                s3, X    =>  s3, VCC;
        END TABLE;
END;
```

4. (50 points) A product brief accompanying this exam describes the ORCA OR3 family of field-programmable gate arrays, available from Lucent Technologies (formerly Bell Labs).

a. (15 points) How is this family of devices *similar* to the Altera FLEX 10K family, in terms of programming, base cell architecture, special features, etc.?

Both are based on SRAM programming, so they have to be configured each time the device is powered up, either by a host computer or by an EEPROM.

Both have architectures based on lookup tables (LUTs), with flip-flops / registers on the LUT outputs. Both have fast carry logic to speed up long arithmetic operations, and I/O cells around the perimeter of the chip.

Both contain RAM blocks.

Both come in very high-density versions.

Both have associated tools that provide simulation and compilation capabilities.

b. (15 points) How is this family of devices *different* from the Altera FLEX 10K family, in terms of programming, base cell architecture, special features, etc.?

The FLEX 10K uses the FastTrack interconnect structure for routing, while the ORCA OR3 uses more locally-visible SLICs attached to each logic cell.

In the FLEX 10K, each LAB contains independent 8 LEs, each of which contains a 4-input LUT. The ORCA OR3 is less hierarchical and coarser-grained, as each programmable logic cell (PLC) contains 8 4-input LUTs, but they can only be used as 2 independent sets of 4 LUTs.

In the FLEX 10K, RAM and ROM is implemented using dedicated Embedded Array Blocks, whereas in the ORCA OR3, each PLC can be configured as a RAM or ROM (closer to the Xilinx XC4000).

This data sheet shows that the ORCA OR3 family supports devices of up to 340,000 gates, whereas the FLEX 10K family only supports devices of up to 250,000 gates.

The ORCA OR3 also has a microprocessor interface and a programmable clock manager, neither of which are on the FLEX 10K.

The ORCA OR3 Product Brief mentions “interfaces to front-end design tools” and says “a back-annotated netlist can be created to allow simulation”, but never explicitly says it provides either design entry or simulation tools (although it does say it includes libraries for third-party tools). However, it does say it provides map, place, and route tools. In contrast, the Altera FLEX family comes with a complete set of design tools, including several methods of design entry and a simulator.

c. (20 points) Suppose you work for a company, and you have been using the Altera MAX+PLUS II tools and the Altera FLEX 10K family for the past two years in your designs. What factors might influence you to change to Lucent’s ORCA OR3 family for your next design?

You might change if you need the larger size, or if you can take advantage of the different LUT or routing structures, or if you can use the microprocessor interface or clock manager. However, this change might also require buying and learning a new set of tools, and converting existing designs from AHDL to a language supported by the Lucent tools.