

Course Examinations 2001-2002

Course Code & Title : ..... **CEG 5010 Rapid Prototyping of Digital Systems** .....

Time allowed : ..... **2** ..... hours ..... **0** ..... minutes

Student I.D. No. : ..... Seat No. : .....

---

**Open book examination. Please answer all questions.**

1. Datapath design. The algorithm for computing the greatest common divisor (GCD) of two positive integer inputs is given in C below:

```
gcd(int s, int l)
{
    int    a, tmp;

    while (s != l)
    {
        a = l - s;
        if (a > 0)
            l = a;
        else {
            tmp = l;
            l = s;
            s = tmp;
        }
    }
    return l;
}
```

- (a) (10 marks) Draw a block diagram of a datapath to implement this design using bit-parallel arithmetic. The design should have two 8-bit inputs, in1 and in2, and one 8-bit output which is the GCD of the two inputs. A one-hot control input ctrl, is used to select which operation is performed by the datapath.

The VHDL entity description (where xx is the number of bits required by your control logic) is given below.

```
entity GCDdpath is
port (
    clk: in std_logic;
    reset: in std_logic;
```

```

in1: in std_logic_vector(7 downto 0);
in2: in std_logic_vector(7 downto 0);
ctrl: in std_logic_vector(xx downto 0);
out: out std_logic_vector(7 downto 0)
);
end GCDdpath;

```

Your answer should only use standard logic gates (AND, OR, XOR etc), multiplexers, adders and subtractors. In your answer, try to minimize the number of registers and computational elements (i.e. comparators, adders, subtraction units etc) used. Make a table of all control inputs and their associated function.

- (b) (10 marks) Express your datapath in VHDL.
- (c) (5 marks) Estimate the number of Xilinx Virtex slices required to implement your datapath by labelling each block in your datapath with your estimate.
2. Control design. For the datapath that you designed in the previous question:
- (a) (10 marks) Draw a state transition diagram for a finite state machine required to implement the GCD processor. Higher marks will be given to answers which use fewer cycles.
- (b) (10 marks) Express the above finite state machine in VHDL.
- (c) (5 marks) Estimate the number of Xilinx Virtex slices required to implement your finite state machine. Justify how you arrived at your answer.
3. Architectural improvements. For the design given in the previous questions:
- (a) (5 marks) Give two ways one could reduce the area of the design at the expense of requiring more cycles.
- (b) (10 marks) Is it possible to pipeline an iterative implementation of the GCD processor? Justify your answer. If pipelining is possible, explain using figures how it can be done and estimate the performance improvement.
- (c) (10 marks) Design a systolic implementation of the GCD processor using a linear systolic array. Each cell should compute a single iteration of the design. Draw a block diagram of each cell including pipelining latches and also show how they can be connected together. The output at the last element of the array should be the GCD. In your answer, explain any changes you need to make to the design to map it into a systolic structure. Also, explain how many processing elements are required for an 8-bit GCD processor (remember that in1 and in2 are numbers greater than zero).
4. General FPGA questions:
- (a) (5 marks) How many registers can be implemented in a Xilinx Virtex logic slice? How many bits of storage are available in the same slice? Show how you arrived at your answer.
- (b) (5 marks) Give two advantages and two disadvantages of using FPGAs to perform computation over using a microprocessor.
- (c) (5 marks) We wish to implement the boolean logic function
- $$f = (ab + c)d$$
- using 4 input lookup tables. Write out the contents of an FPGA  $16 \times 1$  ROM which can be used to implement this equation.
- (d) (5 marks) We wish to implement a 7 bit parity function, the output of which is equal to '1' iff the number of set bits in the input are odd. Explain how this can be implemented using 2 lookup tables and draw a circuit of their implementation which includes the values of the ROMs for both tables.

(e) (5 marks) Explain using a diagram how a delay locked loop can be used to double a clock's frequency.

**-End-**