

Course Examinations 2003-2004

Course Code & Title : CEG 5010 Reconfigurable Computing

Time allowed : 2 hours 0 minutes

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

Open note examination. Please answer all questions.

1. General questions.

- (a) (5 marks) Explain how a delay locked loop (DLL) can be used to minimize clock skew.
- (b) (5 marks) Give a precise description of the situations in which distributed arithmetic can be applied.
- (c) (5 marks) Implementations of embedded systems can be made using microprocessor, FPGA and ASIC technologies. For each technology, give one example in which it is the most suitable and justify your answer.
- (d) (5 marks) The sum output of a full adder is computed by the equation “ $f = A \text{ xor } B \text{ xor } C$ ” where A, B and C are the two inputs and carry. Write out the contents of an FPGA 16×1 ROM which can be used to implement this equation and explain how A, B and C connect to the ROM.
- (e) (5 marks) Explain how the cordic algorithm can be used to calculate $\sin(\theta)$ by giving the initial values and update equations required in the computation.

2. We wish to compute the sum of products

$$y[n] = \frac{x[n] + 2x[n-1] + 3x[n-2]}{8}$$

using distributed arithmetic with all inputs, outputs and coefficients being represented as 4-bit non-negative fractional numbers in the range $0 \leq x < 1$ (e.g. $1001 = 2^{-1} + 2^{-4}$).

- (a) (10 marks) Draw a datapath showing how this can be implemented. The only input is $x[n]$ in a parallel format. Make sure you show the width of all bus connections.
- (b) (5 marks) Give the entire ROM contents for your implementation in binary.
- (c) (10 marks) Illustrate all of the intermediate steps in computing $y[n]$ if the inputs are $x[n]=0.0625$, $x[n-1]=0.25$ and $x[n-2]=0.125$. Compare your answer with the expected one.

3. The Sieve of Eratosthenes. The following pseudocode is an algorithm for finding all prime numbers less than n , given n and its square root inputs:

```

init(int n)
{
    A[1] = FALSE;
    for i = 2 to n
        A[i] = TRUE;
}

/* A is a bit-array of length n (i.e. A[1 .. n])
/* after execution, all elements of A which are TRUE are primes */
SIEVE(int n, int sqrtn)
{
    for i = 2 to sqrtn
    {
        if (A[i] == TRUE) {
            j = 2 * i;
            while (j <= n) {
                A[j] = FALSE;
                j = j + i;
            }
        }
    }
    return A
}

```

- (a) (15 marks) Draw a block diagram of a datapath to implement the SIEVE algorithm using bit-parallel arithmetic. Assume that A[] has already been initialized to the appropriate values as done in the init() routine. The device available is a Xilinx XCV1000E device which has 27,648 logic cells. Give an estimation of the largest value of n that the design can handle **without using the 4 Kb dedicated BlockRAMs in the FPGA**. Higher marks will be awarded to more resource efficient designs.

Your answer should only use standard logic gates (AND, OR, XOR etc), multiplexers, adders, subtractors, comparators, registers and distributed RAMs/ROMs. You can assume that you have a module generator which can generate arbitrary sized RAMs/ROMs, but fully specify the size of any RAM/ROM used. Furthermore, RAMs/ROMs and registers can be initialized to any value (please specify). In your answer, try to minimize the number of FPGA logic cells required. Show the width of all datapaths and make a table of all control inputs and explain their associated function.

- (b) (10 marks) Draw a state transition diagram for a finite state machine which can control the sieve datapath that you designed in part (a). The state transition diagram should be drawn as a graph with the datapath control signal settings shown in the nodes and the edges indicating the transition condition between the states. Also show which lines of the pseudocode are implemented in each node of your diagram. Higher marks will be given to answers which use fewer cycles.
- (c) (15 marks) Write a complete VHDL description of the sieve processor.
- (d) (10 marks) Label each block in your datapath with your estimate of the number of Xilinx Virtex slices required to implement the block. What are the total number of slices required?

-End-