Midterm Exam

ECE 448
Spring 2010

Monday Section

(15 points)

Instructions
Zip all your deliverables into an archive <last_name>.zip and submit it through Blackboard no later than Monday, March 15, 10:15 PM EST.

Introduction

Design a circuit that accepts

- **two k-bit unsigned numbers, A and B, and**
- **the carry in signal, CIN,**

and generates

- **the sum output, SUM, and**
- **the carry out signal, COUT.**

The circuit operates by serially adding consecutive "digits" (d-bit portions) of the two numbers, starting from the rightmost (least significant) "digit".

The operation of the circuit is described using the following pseudocode, interface, and the block diagram.

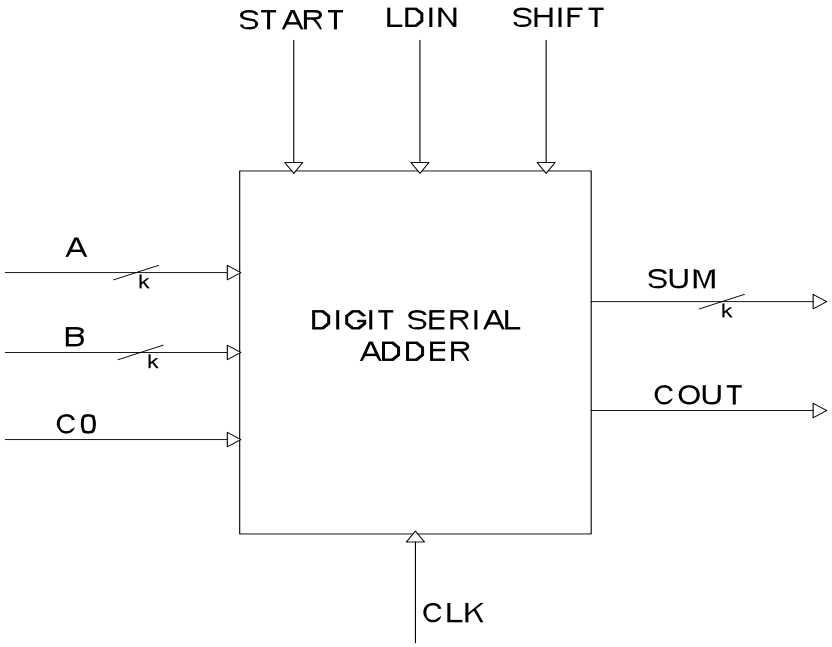
Pseudocode:

```
when LDIN = 1 and rising_edge(CLK)
{
       Areg = A;
       Breg = B;
}

for j=1 to k/d
{
     X = Areg[d-1..0];
     Y = Breg[d-1..0];
     if (j=1)
          Cdin = C0;
     else
          Cdin = Ci;
     (Cdout || S) = X + Y + Cdin;
     Sreg = (S || Sreg) >> d;
     Areg = Areg >> d;
     Breg = Breg >> d;
     Ci = Cdout;
}

SUM = Sreg;
COUT = Ci;
```

Notation:

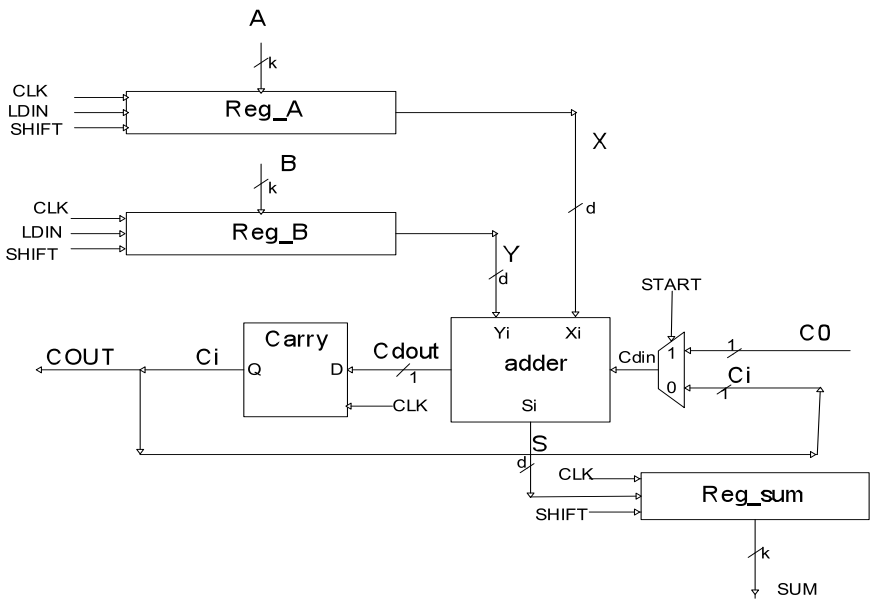
`A || B` : A concatenated with B
`A[k-1..0]` : k least significant bits of A
`>> d` : logical shift by d positions to the right
`+` : arithmetic addition.

Interface:



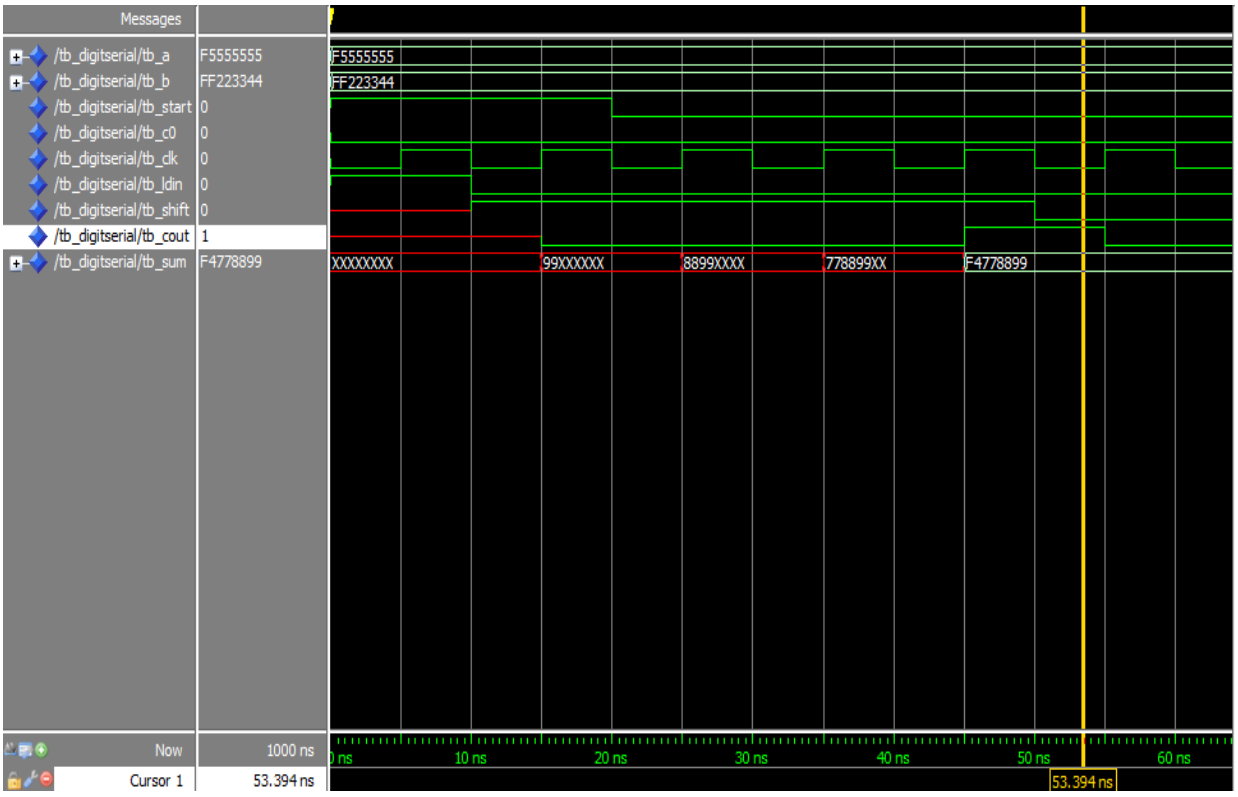
Port	Mode	Width	Function
CLK	Input	1	System clock.
C0	Input	1	Carry in.
A	Input	k	Input A.
B	Input	k	Input B.
START	Input	1	Control signal indicating the first iteration of the computations.
LDIN	Input	1	Control signal indicating loading inputs A and B to the internal registers of the circuit.
SHIFT	Input	1	Control signal indicating shifting data in all internal registers by d bits to the right.
SUM	Output	k	Sum of A and B.
COUT	Output	1	Carry out.

Block diagram:



The timing diagram below show the operation of the circuit for the following values of the parameters k and d, and inputs A, B, C0:

```
k=32; d=8;
A=F5555555;
B=FF223344;
C0 = 0;
```



Design Requirements

The combinational portion of the circuit should be described using the dataflow VHDL code, and the sequential portion of the circuit should be described using the synthesizable behavioral code. Your code should infer a circuit that requires a minimum amount of FPGA resources. The target clock frequency should be 100 MHz.

Tasks

Perform the following tasks:

1. Write a synthesizable VHDL code representing the described above circuit for the generic values of k and d.
2. Write a testbench verifying the operation of your circuit for the case of k=32, d=8, and several possible values of A, B, and C0.
3. Perform functional simulation of your circuit and use it to debug your VHDL code. Save the obtained timing waveforms.
4. Synthesize your circuit for the case of k=32 and d=8. Save the obtained RTL schematic.
5. Perform post-synthesis simulations of your circuit. Save the obtained timing waveforms.
6. Implement your circuit using
   FPGA family: Spartan3E,
   Device: 3s100cp132
   Speed Grade: -4.
7. Run the static timing analysis of your circuit.
8. Based on the circuit block diagram and the implementation reports, determine the most critical path in your circuit and the circuit maximum clock frequency.
9. Based on the implementation reports and the report from the static timing analysis, determine the number of CLB slices, Logic Cells, LUTs, D flip-flops, and pins used by the circuit.

Deliverables

1. VHDL code of your entire circuit fulfilling the requirements specified in the Design Requirements section above.
2. VHDL code of your testbench.
3. RTL schematic of your circuit.
4. Timing waveforms from the functional and post-synthesis simulations demonstrating the correct operation of your circuit.
5. Description of the critical path of your circuit.
6. FPGA resource utilization (as defined in Task 9 above)
7. Minimum clock period and maximum clock frequency of your circuit.