Midterm Exam
ECE 448
Spring 2012
Tuesday Section
(15 points)

Instructions:
Zip all your deliverables into an archive <last_name>.zip and submit it through Blackboard no later than Tuesday, March 6, 10:15 PM EST.

Lab Midterm Exam

Introduction:

The described below circuit performs encryption of N-bit message, using a simple algorithm based on addition, subtraction, multiplication, rotation, shifts and XOR operations. Message is entered to the circuit in blocks of 4 bits, and the corresponding ciphertexts leave the circuit in blocks of 4 bits. The result of encryption depends on two input parameters, key (which needs to be kept secret) and IV (initialization vector, which can be chosen at random on the sender's side and transmitted in clear to the receiver's side).

Interface:

Assume the following interface to your circuit.

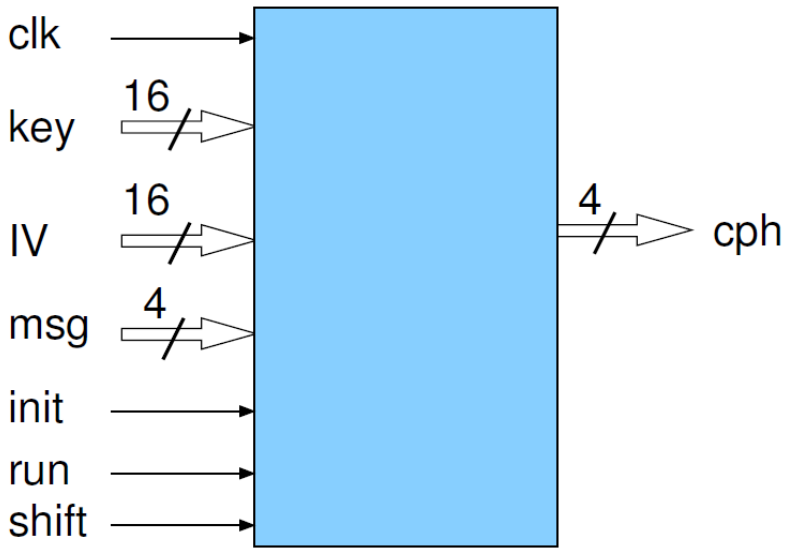
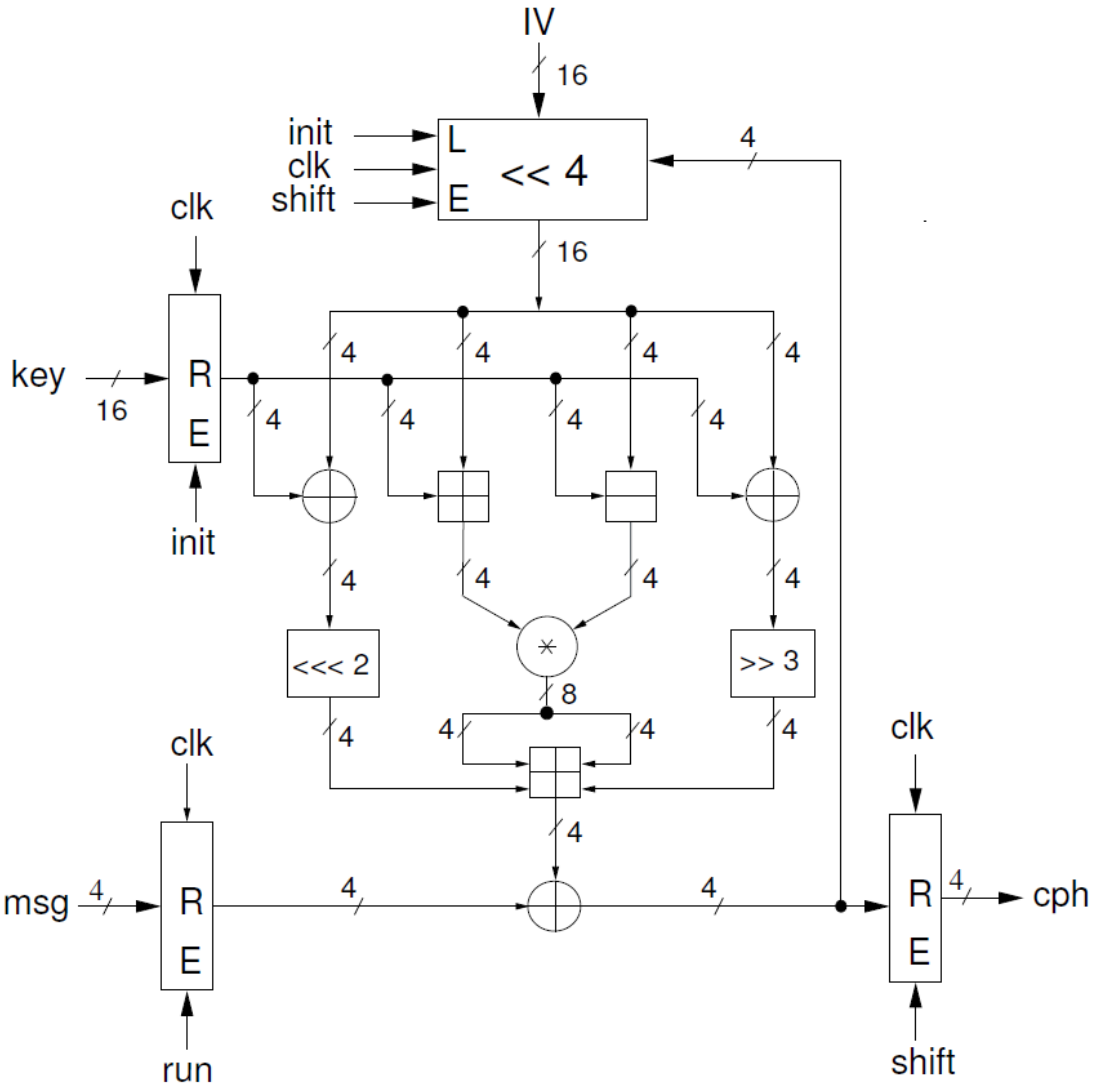


Table of input/ output ports:

Port	Mode	width	Function
clk	IN	1	System clock
init	IN	1	Loads IV and key into the circuit
run	IN	1	Enable signal to enter a block of 4-bits in each clock cycle
shift	IN	1	Enable signal to shift 4-bit block of ciphertext output and IV input
key	IN	16	Cipher key
IV	IN	16	Initialization vector
msg	IN	4	Message block before encryption
cph	OUT	4	Ciphertext block after encryption

Block diagram:



Notation:

⊕ : Bitwise XOR
⊞ : Unsigned addition modulo 2^4, i.e., addition with carry out discarded
⊟ : Signed subtraction without borrow
⊛ : Unsigned multiplication modulo 2^4
R : Register with an enable input E
<<< 2 : Left rotation with rotation amount = 2
>> 3 : Arithmetic right shift with shift amount = 3
<< 4 : Shift register with a parallel load signal L, enable signal E, and a digit-serial input din

Inputs, Outputs and Waveform:

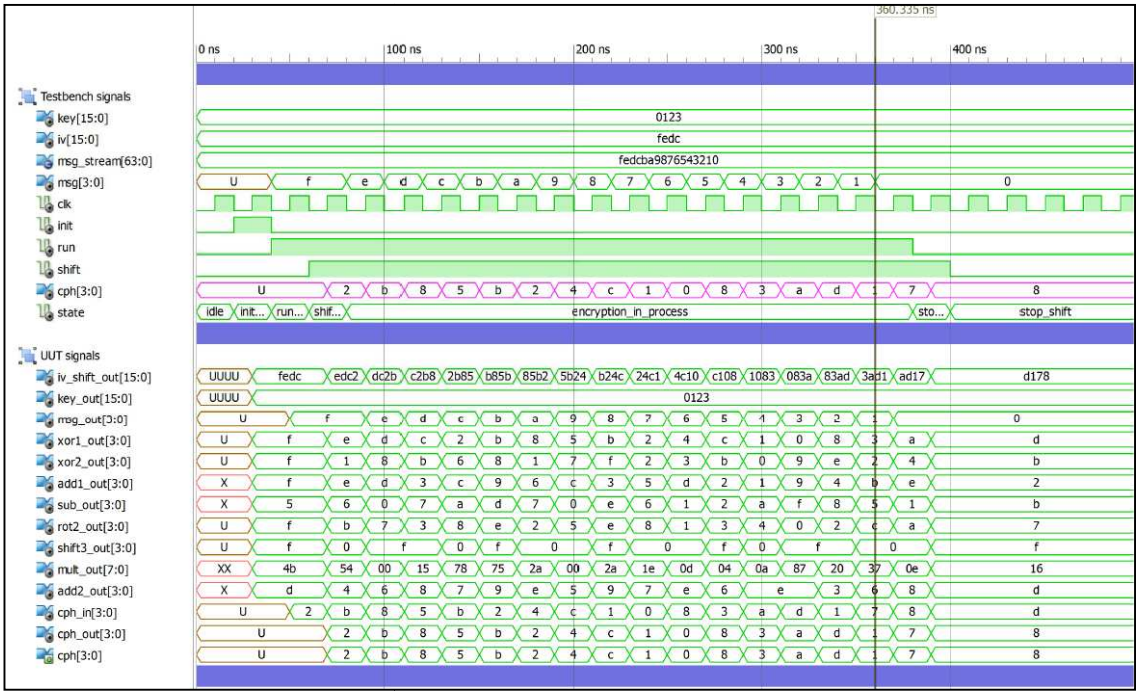
Inputs:

Msg_Stream = x"FEDCBA9876543210"
Key = x"0123";
IV = x"FEDC";

Output (Ciphertext):

cph =x"2B85B24C1083AD17"

Functional waveform:



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 **50 MHz**.

Tasks:

Perform the following tasks:

1. Write a synthesizable VHDL code representing encryption circuit (shown in the block diagram above).
2. Write a testbench verifying the operation of your encryption circuit.
3. Perform functional simulation of your circuit and use it to debug your VHDL code. Take a print out of the waveform showing the entire operation using default PDF conversion tool installed in the lab (Use multiple page option in order to display necessary information on multiple pages, if required).
4. Synthesize your circuit.
5. Implement your circuit using
    a. FPGA family: Spartan 3E
    b. Device: 3s100cp132
    c. Speed Grade: -4
6. Run the static timing analysis of your circuit.
7. Based on the circuit block diagram and the report from the static timing analysis, determine the most critical path in your circuit and the circuit maximum clock frequency.
8. Based on the implementation reports, determine the number of CLB slices, Logic Cells, LUTs, D flip-flops and pins used by the circuit.
9. Perform the timing simulation of your circuit at the **maximum clock frequency** returned by the static timing analysis. Take a screen shot and include that in the report.

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. Timing waveforms from the functional and timing simulations demonstrating the correct operation of your circuit.
4. Description of the critical path in your circuit
5. FPGA resource utilization (as defined in Task 8 above).
6. Minimum clock period and maximum clock frequency of your circuit.