Experiment 5

Part 1 (2 points)

Implement a clock management circuit based on the following diagram

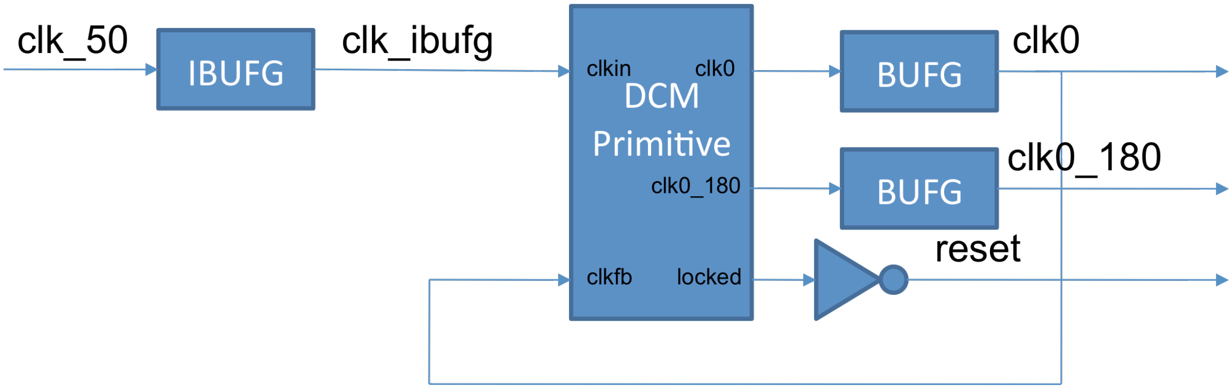

Figure 1 Clock management Circuit

The input clock will be 50 MHz. Knowing this, you should produce the following clocks:

a) 50 MHz, dejittered clock
b) 50 MHz, dejittered clock 180 degrees out of phase
c) 1 kHz built by a counting clock divider (must still run through an IBUFG)

This unit should be completely enclosed in a separate entity.

Part 2 (2 points)

Design an HDLC transmitter and receiver based on the specifications shown in the lab5 slides.

The receiver shall accept two signals: the external clock (assumed to be 50 MHz, but no guarantee) and the incoming data. The input data and input clock will be coming in on JA(0) and JA(1), respectively. Data should be sampled on the rising edge of the incoming clock. The receiver should also have two outputs: output data and a one cycle enable, signaling that new data has arrived. This signal should not go high after receiving flags, but only for data.

The transmitter shall have two inputs: eight bits of data to send (coming from the switches) and a one cycle strobe signaling the transmitter to accept the data. This strobe will be more than one cycle long and should be edge detected to cross the clock boundary. The transmitter should have two outputs: one bit data and a clock. These will be connected to JD(0) and JD(1) respectively. The data transitions should be aligned with the falling edge of the output clock (this is why the clock (b) above was generated).

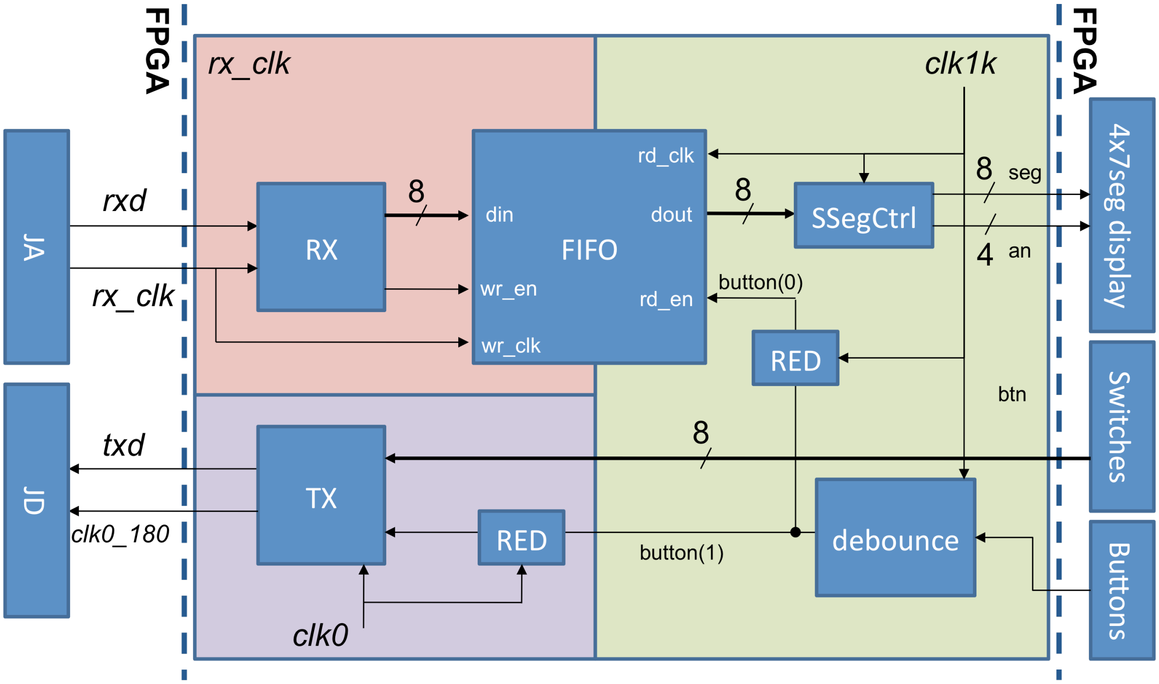

Figure 2 HDLC and surrounding logic

Part 3 (2 points)

Synthesize, implement, and generate a bit file of the entire circuit. This bit file will then be loaded on the board for testing in a real environment.

For the overall design, determine the following properties:

1. Number of CLB slices
2. Minimum clock period after synthesis [ns]
3. Maximum clock frequency after synthesis [MHz]
4. Minimum clock period after implementation [ns]
5. Maximum clock frequency after implementation [MHz]
6. Minimum latency after implementation [ns]
7. Maximum throughput after implementation [bits/s]

As a part of the design process:

1. Draw an optimized block diagram of the transmitter using medium scale logic components, such as multiplexers, one-bit shifters, etc .
2. Translate your block diagram (and the rest of the circuits) into behavioral VHDL.
3. Write a separate testbench capable of verifying the functionality of both of your filters. The testbench should input an impulse into the filter.
4. Synthesize, Implement, and load your design on the Basys board to verify functionality.

In the lab report include:

1. Optimized block diagram of the HDLC transmitter (hand-drawn hardcopy submitted in class and used during an exit quiz, and an electronic copy submitted using Blackboard in the pdf or MS Word format (preferably, the scanned version of the hand-drawn hardcopy).
2. VHDL source codes for:
   a. Clock management unit
   b. HDLC Receiver
   c. HDLC Transmitter
   d. Top Level Unit
   e. User Constraint File (UCF)
3. Testbench and waveforms from the functional simulation (electronic versions submitted using Blackboard) for both filters.

Important Dates

	Monday section	Tuesday section	Wednesday section	Thursday section
Hands-on Session and Introduction to the Experiment	03/01/2010	03/02/2010	03/03/2010	03/04/2010
Demonstration and Deliverables Due	**03/29/2010**	**03/30/2010**	**03/31/2010**	**04/01/2010**