Print:
First Name:. Solutions Last Name:. .....

Student Number:. .....

University of Toronto
Faculty of Applied Science and Engineering

Test II – April 8, 2010
BA 2155

ECE532S – Digital Systems Design

Examiner – Paul Chow

1. There are 5 questions and **9** pages. Do **all** questions. The total number of marks is 45. The duration of the test is 50 minutes.
2. **ALL WORK IS TO BE DONE ON THESE SHEETS!** Use the back of the pages if you need more space. Be sure to indicate clearly if your work continues elsewhere.
3. **Faculty approved calculators are permitted.**
4. Closed book. No aids permitted.

1 [10]	
2 [12]	
3 [10]	
4 [8]	
5 [5]	
Total [45]	

1. Starting with some short answer questions:

[2 marks] (a) What is a SERDES and what is it used for?

Serializer-Deserializer - Used to transmit Serial data at high speeds.

[2 marks] (b) What is an asynchronous event?

The event occurs with an unpredictable phase relationship relative to the sampling clock.

[2 marks] (c) Many tools such as *EDK* generate large numbers of warning messages. These are just warnings, so you can ignore them. Comment on this statement and explain your reasoning.

Bad idea. Some warnings, like the creation of a latch, may actually indicate something important.

[2 marks] (d) What is the difference between *blocking* and *non-blocking* assignments in Verilog?

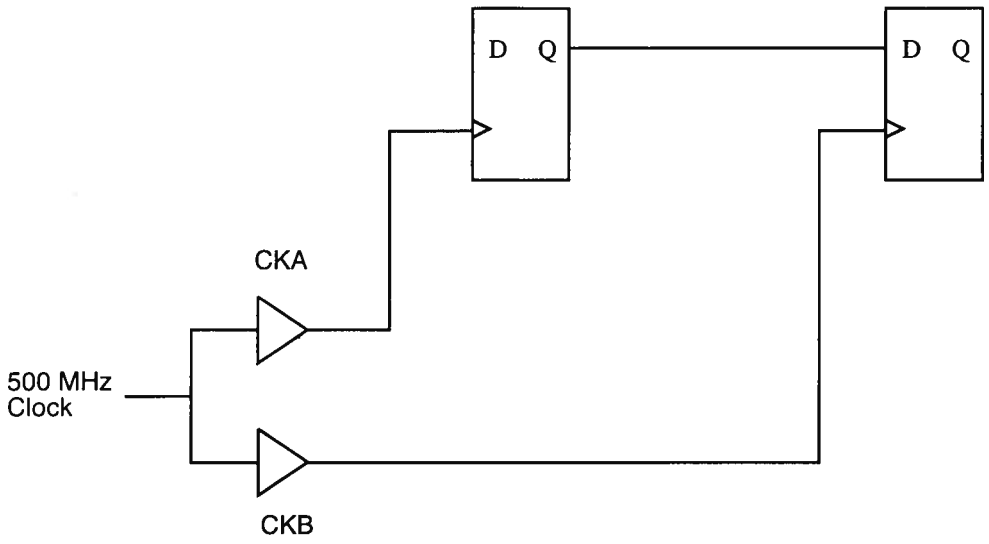
Blocking - simulation progress stops until RHS is evaluated and assigned to LHS

Non-blocking - assignment to LHS occurs at the end of the always block

[2 marks] (e) What is *8B10B* coding used for?

To ensure rich number of transitions in the serial data stream so that the receiving PLL can stay locked

[10 marks] 2. Consider the circuit shown below:



From the data sheets you have also gathered the following information about the flip flops:

$$\begin{aligned} t_{\text{setup}} &= 0.20\ ns \\ t_{\text{hold}} &= 0.10\ ns \\ t_{\text{ClockToQ}} &= 0.50\ ns \end{aligned}$$

The propagation delay on the wire between the flip flops is 0.15 *ns*.

The clock tree has a maximum skew of 0.30 *ns*, i.e., the phase of any one branch could be ahead or behind the other by 0.30 *ns*.

The clock source is a 500 MHz clock (2 *ns* period) with a jitter of 0.10 *ns*.

Being a very conservative designer, you also like to include 0.20 *ns* of extra timing margin to all of your calculations.

Question 2 continued...

Determine whether the circuit will operate correctly including the extra timing margin. Show clearly how you come to your conclusions.

[4 marks] (a) Check whether the *setup* time is satisfied ignoring skew and jitter.

Without skew + jitter, clock arrives at FFs at the same time.

Setup time = Clock period − ($t_{clktoQ} + t_{wire} + t_{margin}$)

= 2.0 − (0.5 + 0.15 + 0.2) ← add margin for worst case

= 1.15 ns > t_{setup} (OK)

[4 marks] (b) Check whether the *hold* time is satisfied ignoring skew and jitter.

Without skew + jitter, the hold time is just the delay from the transmitting clock edge to the receiver input

Hold time = $t_{clktoQ} + t_{wire} - t_{margin}$ (subtract margin for worst case)

= 0.5 + 0.15 − 0.20

= 0.45 ns > t_{hold} (OK)

[2 marks] (c) Check whether the *setup* time is satisfied including skew and jitter.

The worst case is that the uncertainties combine to shorten the clock period.

$t_{uncertainty} = t_{skew} + t_{jitter}$ = 0.3 + 0.1 = 0.4 ns

Setup time = (Clock period − $t_{uncertainty}$) − ($t_{clktoQ} + t_{wire} + t_{margin}$)

= (2.0 − 0.4) − (0.5 + 0.15 + 0.2)

= 0.75 ns > t_{setup} (OK)

[2 marks] (d) Check whether the *hold* time is satisfied including skew and jitter.

The worst case is the uncertainties combine to make the receiving clock edge later by $t_{uncertainty}$ (from above)

Fastest Data Change = $t_{clktoQ} + t_{wire} - t_{margin}$ = 0.45 ns

Slowest Receiver Clock Path = $t_{uncertainty}$ = t_{skew} = 0.3 ns

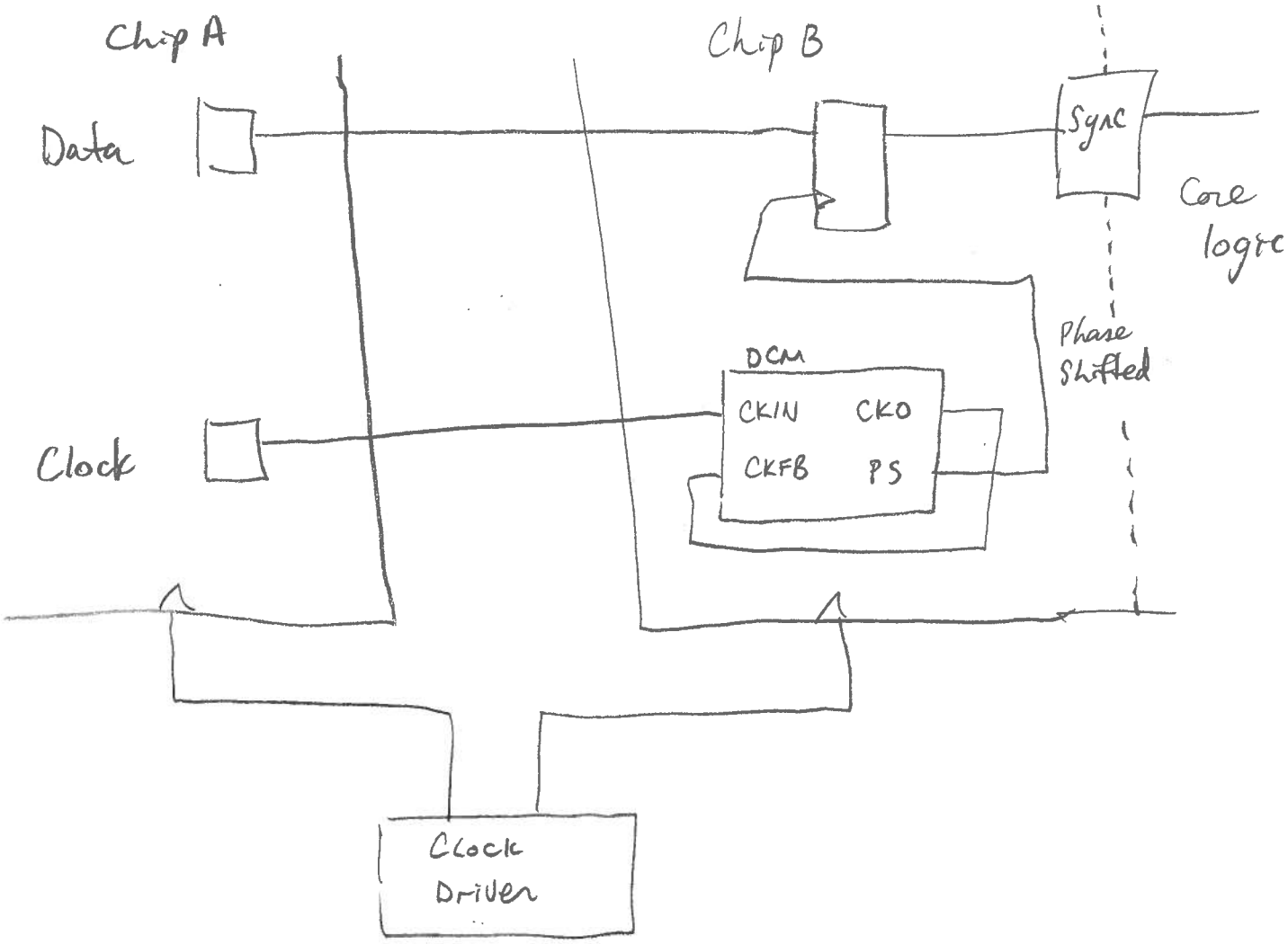
No Jitter effects since same clock edge

Hold time = Data Delay − Clock Delay

= 0.45 − 0.30 = 0.15 ns > t_{hold} (OK)

3. High speed synchronous interfaces, such as for synchronous memories, use a technique called *source synchronous* timing or *clock forwarding*.

[5 marks] (a) Draw the basic model of a *source synchronous* interface.



[3 marks] (b) Explain the basic operation of a *source synchronous* interface.

Transmit clock in parallel with the data. Use a phase-shifted version of the clock at receiver to sample at the centre of the data eye. This data must then be synchronized with the core logic

[2 marks] (c) A *source synchronous* interface can be *trained* to minimize the likelihood of error in receiving the data. Explain what might be done to train such an interface.

Transmit a known data sequence

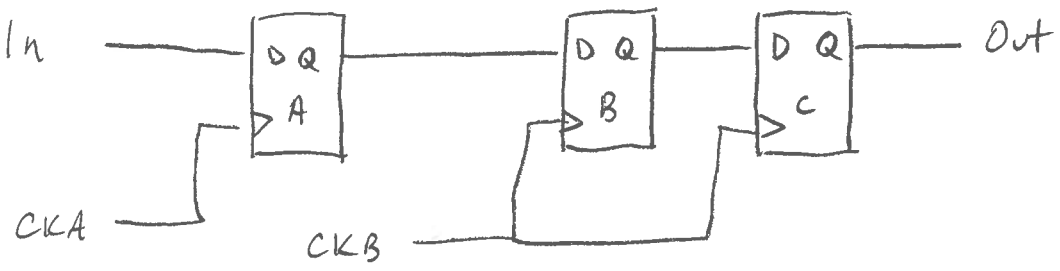
Adjust receiver clock phase to find the edges of the data eye. Set the phase shift to sample data in the middle of the eye.

4. Moving data between clock domains is a common requirement in modern digital systems because of the large number of clocks used.

[2 marks] (a) What is a clock domain?

Region of logic clocked by the same clock

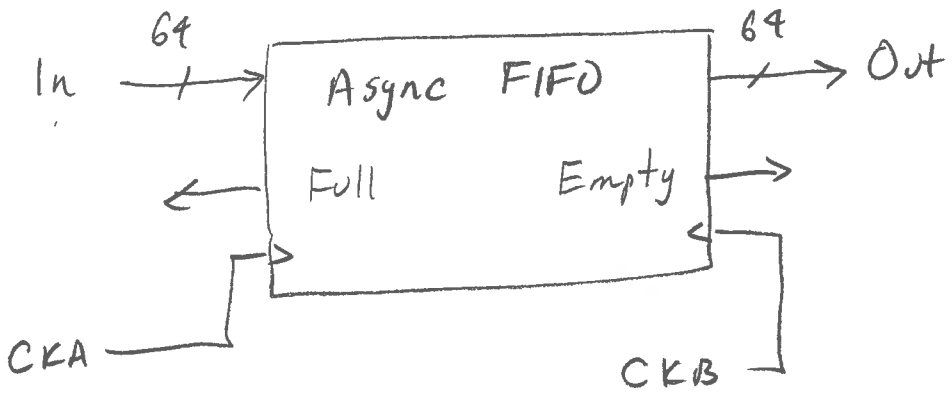
[3 marks] (b) Show a synchronizer for transporting a single bit across a clock boundary. Explain how it works



If FF B goes metastable, have about 1 clock period for it to stabilize before it is sampled by C.

Question 4 continued...

[3 marks] (c) Show a synchronizer for transporting a 64-bit bus across a clock boundary. Explain how it works.



The async FIFO is built with an async memory (no clock) that has 2 ports. Read and write can be done at different speeds

[5 marks] 5. You are starting your job at a new startup company and attending the kick-off meeting for their new chip design. The chip is very large and each designer will be given a particular component of the chip to implement. After assigning the components to each person, the design manager says that everyone should now go off and work on their pieces individually. The task of combining the components into the full system will be dealt with when everyone is done.

What do think about this? Does this bode well for a highly successful company or should you be looking for another job? Justify your thoughts. If you had a recommendation for the manager, what would it be and explain why you think it would be better.

Does not sound like a path to success. Integration can be a significant challenge unless planned from the beginning. Look for another job!

Recommendation - Plan the partitions and the interfaces. The best case would be to build some kind of simulation where the interfaces can be tested as early in the design as possible.