Print:

First Name:. . . . . Last Name:. . . . .

Student Number:. . . . .

University of Toronto
Faculty of Applied Science and Engineering

Test I – February 23, 2009
BA 2165

ECE532S – Digital Systems Design

Examiner – Paul Chow

1. There are 4 questions and **10** pages. Do **all** questions. The total number of marks is 50. 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 [20]	
3 [8]	
4 [12]	
Total [50]	

1. Starting with some short answer questions:

[2 marks] (a) Give three features that *EDK* provides above a basic synthesis, place and route tool like *ISE*.

- System-level design - HW+SW IP
- MicroBlaze, PPC
- software integration
- various tools + features can be listed

[2 marks] (b) A common error message when a C program crashes is something like, "segmentation fault, core dumped". It is typically caused by an uninitialized pointer. What address is that uninitialized pointer accessing and how did the pointer get that value?

Address 0

Uninitialized variables are set to 0 in the C runtime, which is executed before the main( ) is started.

[2 marks] (c) What is a pcore in the EDK environment?

An IP block that can be integrated into an EDK project.

Contained in a directory with a particular set of files describing various aspects of the core

[2 marks] (d) What is the difference between an mhs file and an mss file?

mhs - describes the hardware in the EDK project
mss - describes the software "

[2 marks] (e) What does the vsim command do?

Invokes the Modelsim simulator

[20 marks] 2. You are asked to prototype a simple video player. The videos are to be stored in MPEG format files on a flash device. The flash device should have a standard file format so that the device can be plugged into any computer for updating the video files. You are given an MPEG decoder chip that can take a stream of compressed MPEG as an input and outputs a digital video stream. The output video stream is in RGB mode with 8 bits per colour, meaning you will get 3 values of 8 bits each for each output pixel. Recall that video is output in scan lines (rows of pixels) with a certain number of rows making up a frame of the video. The video from the decoder chip comes out in interlaced mode, which means that it alternates between outputting a frame of the odd scan lines and a frame of the even scan lines. Your monitor is non-interlaced, so it requires both the even and odd scan lines in one frame. You also see that the MPEG decoder chip has a manual of over 100 pages and over 50 registers that could be configured or monitored. There is a simple address/data port to access the 16-bit registers.

Design an FPGA-based prototype of the simple video player. As with all initial design specifications, there is probably not enough information. Under normal cases, you should be asking more questions. In this case, you cannot do this so clearly state any assumptions you need to make.

You will develop your system with the XUP board connected to a host PC that runs all of the tools you need. A daughter card containing the MPEG chip and flash device can be part of the design. The daughter card will be designed to plug into the connectors on the XUP board. Assume that you have enough pins on the board and the FPGA to do what you need to do.

Draw a block diagram of your proposed system with enough detail so that it could be presented at a design review of your peers where they can provide comments.

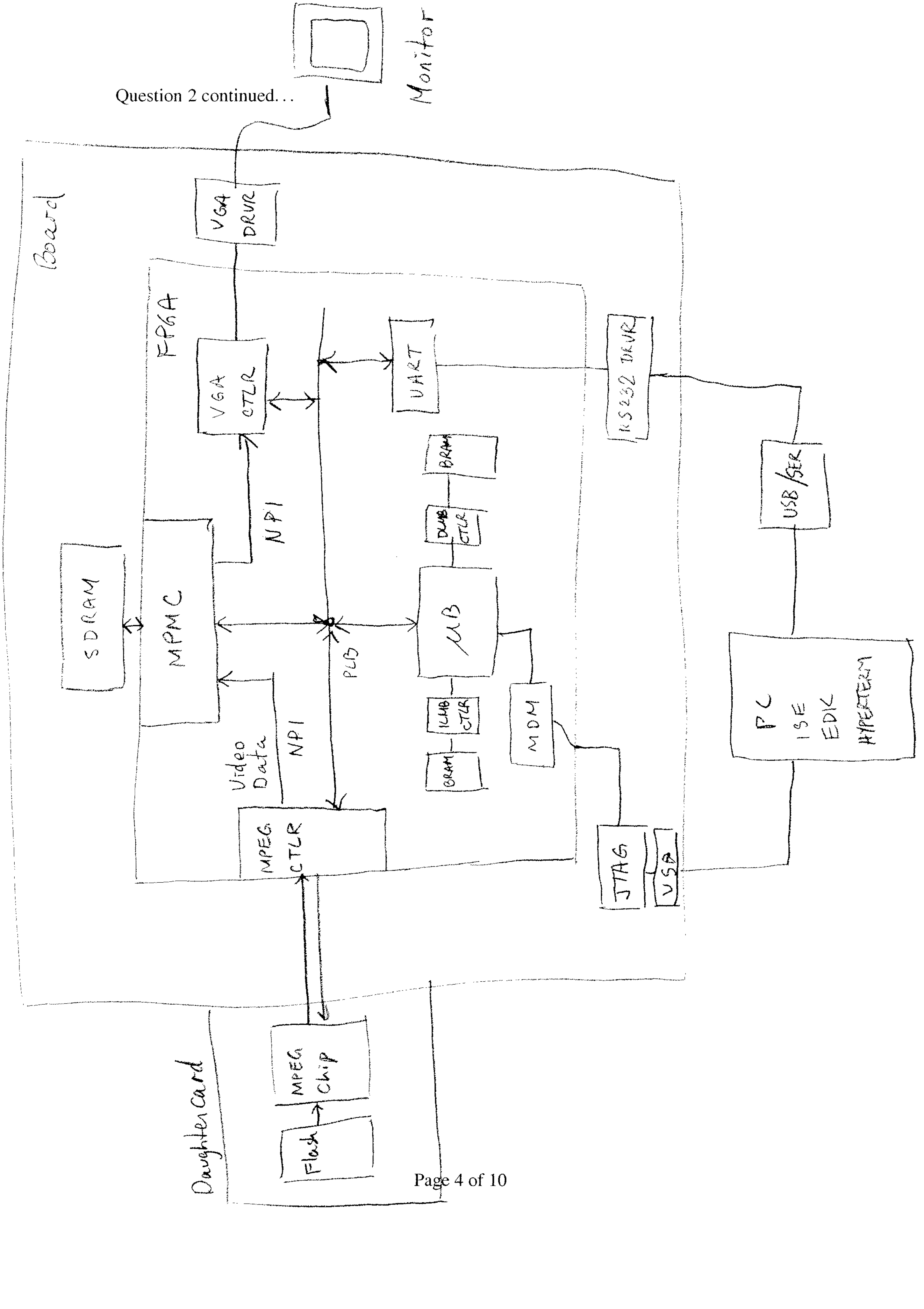
Your diagram must show all of the necessary components, including the external chips, how the components are connected and how they are implemented (indicate clearly whether a component is in FPGA logic, an external chip, etc.). Do not forget to sketch out how the daughter card should work. To help the reviewers understand the design environment, you should include how the prototype is connected to the PC and how it interacts with the tools.

For each block in your diagram, briefly explain the function of the block and whether it is custom-designed logic or a block that you know is available from the EDK library of IP. If the block is custom, indicate whether you have to write it from scratch or whether there is a tool that can help you.

The evaluation of your solution will also be based on how easy it is to understand! Clarity of presentation is very important.

There is more space on the next two pages to answer this question.

Question 2 continued…



Question 2 continued…

For any reasonable resolutions, you cannot use the PLB-
∴ provide ports for data input and output directly on memory using the native port interface (NPI) of the MPMC. Also provide μB a port for access to video buffers and for μB inst./data if needed

Function of blocks:

Flash: Flash memory - probably USB key to allow easy loading of files from a computer

MPEG chip: Assume it has USB interface - many commodity chips available have flash interface

MPEG CTLR: Interface to MPEG chip - Has NPI port to MPMC and PLB bus interface so processor can access registers Custom core. Might be able to use IPIF for PLB interface

MPMC: multi-ported memory controller - from library

SDRAM: external RAM for frame buffers - use double buffering - de-interlace on writing by writing first to "even" addresses and then to "odd" addresses - read sequentially

VGA controller - reads data from output frame buffer, handles picture synchronization - accessible via PLB for control - uses NPI on MPMC

VGA driver - D/A on board to drive monitor - custom, may borrow from existing designs

UART - for external control + communication with μB - library

RS232 DRVR - analog drivers for serial interface

Questron 2 continued

USB/SER: converter from USB on PC to serial port on board

μB: MicroBlaze - from EDK library - processor, write C/C++ code, drivers available for UART

I/D LMB controllers - instruction/data local memory bus controllers to provide on-chip BRAM memory for processor - library

BRAM - block RAM - internal memory in FPGA

MDM - debug module to access and control μB.

JTAG - JTAG controller on board

USB - USB to JTAG converter - connect to PC via USB

PC - runs Xilinx ISE/EDK tools

- Hyperterminal to connect to serial port

[4 marks] 3. (a) The advent of FPGAs means that it is no longer necessary to do simulation of hardware designs. We can save on the cost of buying Modelsim licenses. Comment and justify any statements you make.

Must always do simulations. Finding errors by running the design on the FPGA is a long and tedious process because of the time it takes to do synthesis, place and route.

Simulation provides much greater control and Observability.

Simulation allows testing of corner cases that may not be encountered frequently in real use

Can simulate small components without needing full system

[4 marks] (b) A very difficult problem is managing the partitioning of a design so that it can be implemented by several designers at the same time. What are the potential problems and how would you minimize the time it takes to integrate all of the pieces?

Potential Problems mainly occur at
Interfaces - must agree on signals + timing

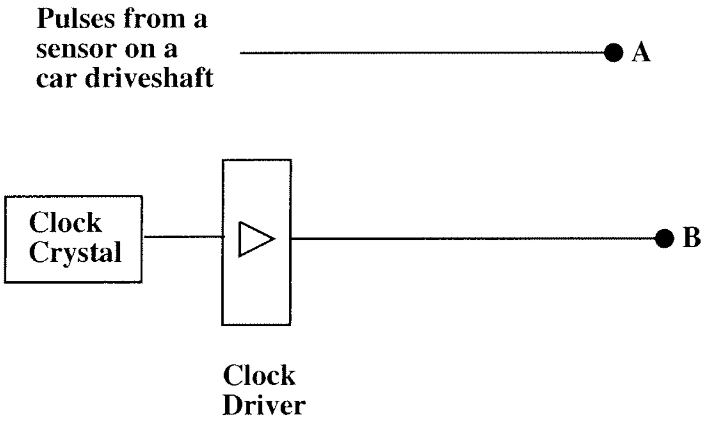
Can do mixed-mode simulation:
First build the golden model using languages like SystemC → capture all interface definitions and timing in an executable model.
Each designer simulates their piece against the golden model by replacing their partition in the model with their HDL. Simulate in this mixed mode to make sure that HDL will simulate correctly.

"Executable" spec is best as written specs can still have mistakes or be misinterpreted

4. Almost every modern chip design must handle multiple clocks. Many of the clocks arise from the different requirements of the I/Os connected to the chip. Therefore, it is important to recognize different clocking scenarios so that you know how they must be handled. When analysing clocking, there will always be the transmitting clock and the receiving clock. The clocks may or may not be in the same domain.

[8 marks] (a) For each of the clocking scenarios shown below, comment on the arrival rates of the clock edges and the relative phases of the two clocks at the points marked **A** and **B**. Assume **A** is in the transmit domain and **B** is in the receive domain. If you happen to remember the name (I don't always...) for the clocking scenario, give it for *possible* extra consideration, but only if you get the important part correct!

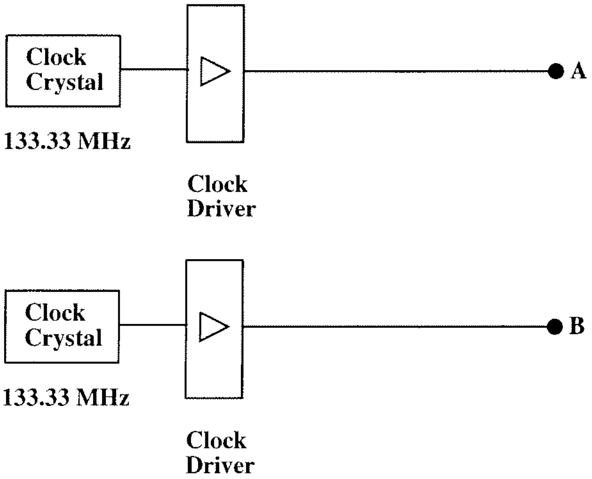
i.



Asynchronous

- A signals arrive at arbitrary times
- no phase relationship

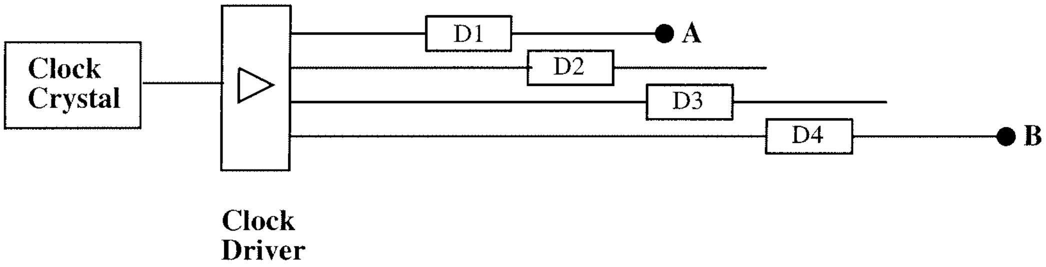
ii.



Plesiochronous

- no fixed phase relationship
  - phase difference slowly varying
- arrival rates approximately same but really these are different clocks
  - crystals can never be exact match

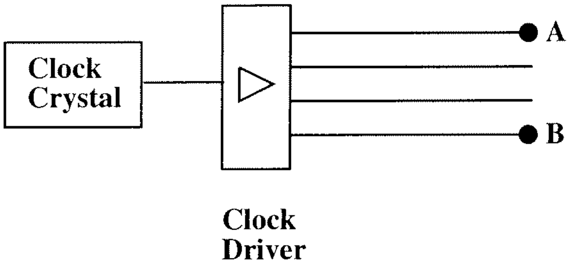
iii. The boxes labeled D1, D2, etc. represent known fixed delays.



Mesochronous

- edges arrive at same average rate
- phase relationships are fixed

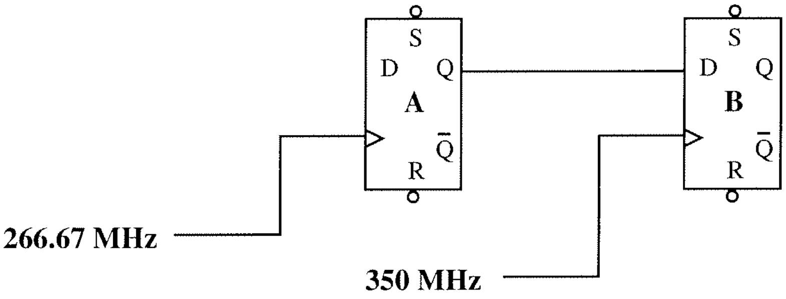
iv. All outputs are matched.



Synchronous

- edges arrive at same rate
- edges in phase

[4 marks] (b) The most common problems in a multi-domain clocking system occur when transferring data between clock domains. The following circuit is an attempt to transmit data from Flip Flop **A** to Flip Flop **B**. What can go wrong? Explain you answer.



Flip Flop B can go metastable because there is no guarantee that data changes from A will obey setup and hold times for B.