ECE428 Lab3 Instruction

Modern FPGAs provide various digital clock management (DCM) circuits to help designers cope with clock related design challenges (skew, jitter, etc.) in high-speed circuit design. This lab is to expose the use of such components. The clock management circuit to be used in the lab is Xilinx DCM_CLKGEN primitive [1]. In the lab, you will use it to divide a 100MHz clock to 5MHz or 4MHz clock upon the input request. The block diagram of the circuit to be designed is shown below:

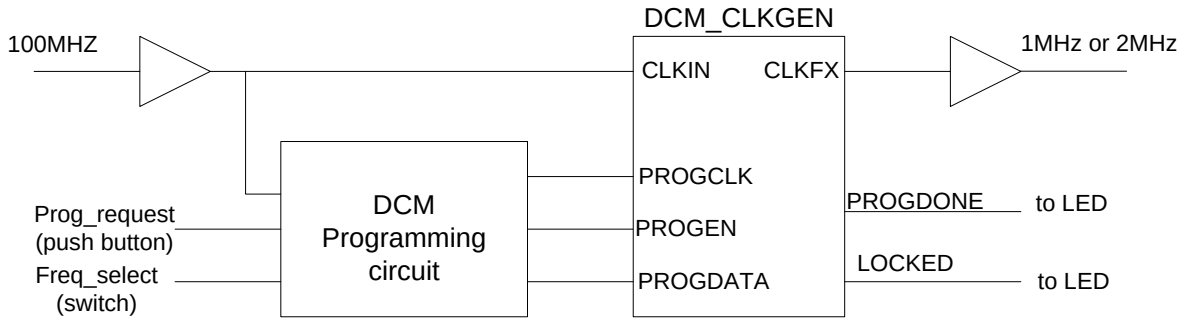

Figure 1: Circuit block diagram

The relation between the CLKFX and CLKIN is:

$$f_{CLKFX} = f_{CLKIN} \frac{CLKFX\_MULTIPLY}{CLKFX\_DIVIDE}$$

Both CLKFX_MULTIPLY and CLKFX_DIVIDE (denoted as M and D in the following discussion) need be sent the DCM_CLKGEN circuit by the DCM Programming Circuit according to the following protocols explained in Figures 2 and 3 on the following page.

You need design the DCM programming circuit to achieve following functions:

- If Freq_select=1, push the Prog_request button will program the DCM to generate a 4MHz clock at CLKFX output
- If Freq_select=0, push the Prog_request button will program the DCM to generate a 5MHz clock at CLKFX output
- CLKFX_MULTIPLY value should be 2.

Connect the CLKFX to a pin of the Pmod connector on the FPGA board such that you can use oscilloscope to verify the clock frequency. Also, connect the PROGDONE and LOCKED signals to LEDs on the FPGA board.

An ordered M/D programming sequence can be described as:

- A 2-bit LoadD command `10`, followed by 8-bit (D-1), with the LSB first. This consumes exactly 10 cycles when the PROGEN pin must remain a logic High.
- A 2-bit LoadM command `11`, followed by 8-bit (M-1), with the LSB first. This consumes exactly 10 cycles when the PROGEN pin must remain a logic High.
- A 1-bit GO command `0`. The PROGEN pin must remain a logic High for exactly one cycle.
- The SPI master monitors the PROGDONE pin and waits for it to be asserted to a logic High.
- When the DCM asserts the LOCKED signal High, a new and valid clock frequency should be present on the CLKFX pin.

Figure 2-11 illustrates an example where the M is programmed as 14 and the D as 3. There is a minimal gap requirement between any two commands. The gap between the LoadD and LoadM is at least two cycles. The gap between the LoadM and GO is at least one cycle.

Figure 2: Programming sequences

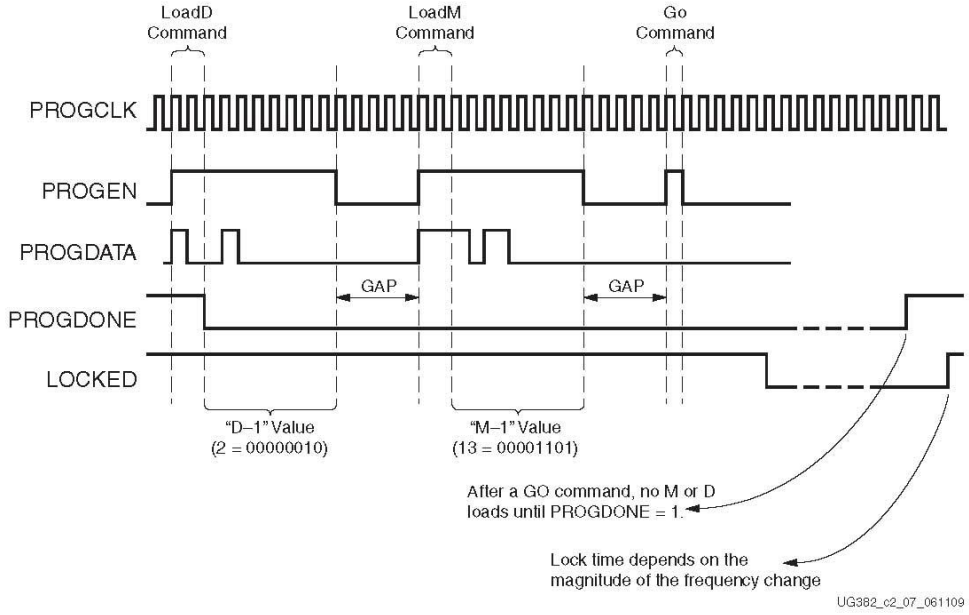

Figure 2-11: **DCM_CLKGEN M and D Configuration Timing Waveform**

Figure 3: Programming waveforms

In order to obtain a 4 MHz clock, the main clock from the Fpga needs to be divided by 25. As the value for the multiplication factor is 2 so we need to tell the DCM to divide the CLKIN by 50. To divide the CLKIN by 50 the value to be shifted into the DCM would be 2 bit load D command “10” followed by { 8 bit (D -1) command , starting with LSB first } . So the data to be shifted into the DCM will be “1010001100“ because 00110001 is 49 in decimal .

Similarly to get a 5 MHz clock, the main clock from the Fpga needs to be divided by 20. As the value for the multiplication factor is 2 so we need to tell the DCM to divide the CLKIN by 40. To divide the CLKIN by 40 the value to be shifted into the DCM would be 2 bit load D command “10” followed by { 8 bit (D -1) command , starting with LSB first } . So the data to be shifted into the DCM will be “1011100100 “ because 00100111 is 39 in decimal .

Lab report:

1) One report is needed for a group. The lab report should be printed. The report should discuss your design and lab activities. Meanwhile, it should be concise. The following materials are needed in your report:
    a. Schematic or HDL codes of your design
    b. FSM state diagram of your design if FSM is used in your design
    c. A snapshot of the simulation waveform
    d. A signed paper from TA to show that your demonstration is checked by TA.

Reference

1. Spartan 6 FPGA Clock Resources User Guide, Xilinx 2011 (http://www.xilinx.com/support/documentation/user_guides/ug382.pdf)