FPGA-BASED DIGITAL CLOCK MANAGER

Here’s a digital clock manager that is easy to implement. The design is very flexible with many programmable inputs

VINEET GOEL

Design a digital clock manager (DCM) that gives flexible, complete control over the clock frequency.

The complete DCM module is divided into five sub-modules: digital frequency synthesiser, duty-cycle correction, programmable phase shifter, programmable duty-cycle synthesiser and coarse phase shifter.

The digital frequency synthesiser (DFS) performs the function of clock multiplication/division. The multiplication of a clock frequency by a programmable number is often necessary in high-performance signal synthesisers as well as in FPGAs. Most of the systems utilise PLL or DLL for clock multiplication. These systems result in high phase accuracy with higher programmability but at the expense of high complexity and high power consumption.

The output clock frequency of the DFS is given by:

\[ f_{\text{DFS}} = f_{\text{clk}} \frac{\text{multiplier}}{\text{divider}} \]

The output clock from the DFS is phase symmetric with 50% duty cycle.

The duty-cycle correction module corrects the duty cycle of the input clock to 50%. If the input clock has 30% duty cycle, the output-clock duty cycle is corrected to exactly 50% with no change in frequency.

The programmable phase shifter (PPS) provides programmable phase-shifted versions of the input clock. It has integer values ranging from –255 to 255 to which phase-shift attribute is assigned.

The programmable duty-cycle synthesiser varies the duty cycle of the input clock as per the programmable input. This helps to improve the metastability factor due to set-up/hold time violations. This module assigns 0 to 100% duty cycle to the output clock.

The coarse phase shifter provides output clocks that are 90°, 180° and 270° phase-shifted versions of the input clock.

An introduction

The DCM is simulated in VHDL language using XILINX Project Navigator software. It does not rely on phase-locked loop (PLL), delay-locked loop (DLL) or any feedback loop, but exhibits most of the functionalities of a DCM like digital frequency synthesis (multiplier/divider), duty-cycle correction, programmable phase shifter, programmable duty-cycle synthesiser and coarse phase shifter.

The output clock from the DCM is phase symmetric with 50% duty cycle. The duty-cycle correction module corrects the duty cycle of the input clock to 50%. If the input clock has 30% duty cycle, the output-clock duty cycle is corrected to exactly 50% with no change in frequency.

The programmable phase shifter (PPS) provides programmable phase-shifted versions of the input clock. It has integer values ranging from –255 to 255 to which phase-shift attribute is assigned.

The programmable duty-cycle synthesiser varies the duty cycle of the input clock as per the programmable input. This helps to improve the metastability factor due to set-up/hold time violations. This module assigns 0 to 100% duty cycle to the output clock.

The coarse phase shifter provides output clocks that are 90°, 180° and 270° phase-shifted versions of the input clock.
Since DLL or PLL is not the basis of this design, the maximum allowed input reference clock is of 1 MHz. All the sub-modules of the DCM are more or less based on the same basic concept.

**Hardware description**

Consider that the DFS reference clock is to be multiplied by factor 'M' and divided by factor 'D'.

The DFS basically consists of a master clock of frequency Fc, a divide-by-'M' counter, an accumulator counter counting to 'K' and a divide-by-'K' counter. The value of 'K' is given by:

\[ K = \frac{F_c}{M} \times T \]

The output of the divide-by-'D' counter will be a clock with time period given by:

\[ F_{out} = \frac{F_c}{K} = \frac{M}{T} \]

This output is given to the input of the divide-by-'D' counter whose output will be a clock with frequency:

\[ F_{out} = \frac{M}{TD} \]

The values of 'M' and 'D' are programmable inputs to the DFS.

The duty-cycle correction module is also based on the same concept as DFS.

The accumulator counter counts to 'K' over time period 'T' driven by very high-frequency master clock. The logic part is such that the output clock changes state when the value of the accumulator counter is either K/2 or K. In this way, you can have exact 50% duty cycle output clock irrespective of the duty-cycle of the input clock.

The programmable phase shifter has a programmable 8-bit input, the MSB of which defines the positive or negative phase shift with reference to the input clock and the bits '0' to '7' decide the amount of phase shift. The hardware description is the same as for duty-cycle correction module, except for the logic part. The portion of clock to be shifted per unit of the programmable input is estimated on the basis of value 'K' accumulated in the counter and that much lag or lead is done by the master clock.

The hardware representation for the programmable duty-cycle synthesiser is also similar to that of the PPS. The logic part is implemented such that on the basis of the percentage of the duty cycle required, the output clock is kept high for some part of 'K' (0 to K1) and low for the other part of 'K' (K1 to K). The value of K1 is derived from the input value of the duty cycle required.

The coarse phase shifter provides the outputs as CLK0, CLK90, CLK180 and CLK270. The output also depends on the duty_cycle_correction input. If it is true, the output clock has 50% duty cycle. If it is false, the output clock has the same duty cycle as the input clock.
implementation and the values of delay at the gate and flip-flop level. As per the delay values provided in the datasheet of XILINX Spartan-3 FPGA XC3S1500, seven inverters are placed as delay element. The reset signal is provided to the ring oscillator as well as to all the counters and is an active-high signal.

**Error analysis**

There are mainly two types of errors to be analysed in this implementation of the DCM. One of the errors occurs because the period of the reference clock may not be an integral multiple of the master clock period. This error results in jitter. The second error occurs because the accumulated value \( K \) in the counter may not be exactly divisible by any of the divide-by-\( N \) counters. This error results in phase asymmetry.

Both these errors can be minimised by having master clock at very high frequency so that the counter counts up to very large values.

But since we cannot employ ring oscillators with very high frequencies (due to the need for the circuit to process the OSC signal—typically, a few nanoseconds), this limits the maximum frequency that can be controlled by the DCM. In this case, based on the current technology of FPGA used, the output frequency is below 20 MHz.

**Simulation results**

All the results shown in Fig. 5 are simulated on XILINX Spartan-3 software.

The block diagram of the DCM with inputs and outputs is shown in Fig. 6.

The author is a research engineer at Center for Development of Telematics (C-Dot), Delhi