International Journal of Engineering Research and Science & Technology

ISSN : 2319-5991
Vol. 1, No. 1
March 2015

National Conference on "Recent Prends in Communication & Information Technologies"NCRTCIT 2015

Organized by

Dept. of ECE, Indra Ganesan College of Engg., Trichy, Tamil Nadu, India.

www.ijerst.com

Email: editorijerst@gmail.com or editor@ijerst.com

International Journal of Engineering Research and Science & Technology

ISSN 2319-5991 www.ijerst.com
Vol. 4, No. 2, May 2015
© 2015 IJERST. All Rights Reserved

Research Paper

DESIGN OF FAST LOCKING ALL DIGITAL DESKEW BUFFER WITH DUTY CYCLE CORRECTION

K Kanimozhi[1]* and A S Biji[1]

*Corresponding Author: K Kanimozhi ✉ kkaniece@gmail.com

Modern high frequency, high performance system-on-chip design is heading to include more and more analog or mixed signal circuits as well as digital blocks. As the complexity of a system grows, it becomes more important to implement the system simulation and top-down design methodology . In this paper, we have designed a delay locked loop using Verilog and Xilinx. Considering the rapid growth in computer automation and computer networking sector, FPGA implementation technique of DLL has been adopted in this paper. DLL model basically used for synchronization of closed loop RF control signals. In ASIC system sometimes it is very important to synchronize the input signal with the output signal. Synchronous DLL can be used for this purpose. This paper has considered the minimum number of devices like Flip Flop, comparator, adder or subtractor due to enhance the efficiency of the system and the time delay of DLL is also considered to be much less and also design vertex FPGA series to provide zero propagation delay, low clock skew between output clock signals distributed throughout the device, and advanced clock domain control.

Keywords: Delay locked loop, voltage controlled delay, Variable delay line

INTRODUCTION

As FPGAs grow in size, quality on-chip clock distribution becomes increasingly important. Clock skew and clock delay impact device performance and the task of managing clock skew and clock delay with conventional clock trees becomes more difficult in large devices. The Virtex series of devices resolve this potential problem by providing fully digital dedicated on-chip DLL circuits that provide zero propagation delay and low clock skew between output clock signals distributed throughout the device.

Each DLL can drive up to two global clock routing networks within the device. The global clock distribution network minimizes clock skews due to loading differences. By monitoring a sample of the DLL output clock, the DLL can compensate for the delay on the routing network, effectively eliminating the delay from the external input port to the individual clock loads with in the device.

[1] MIET Engineering College, Trichy - 620007.

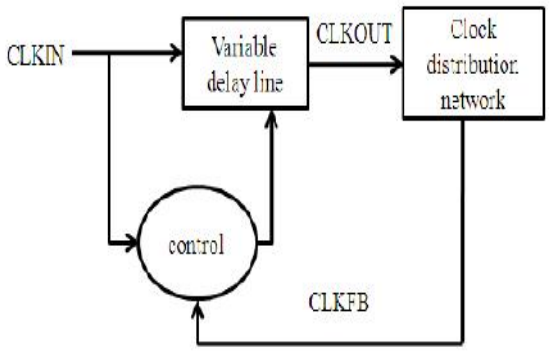
In addition to providing zero delay with respect to a user source clock, the DLL can provide multiple phases of the source clock. The DLL can also act as a clock doubler or can divide the user source clock by up to 16.Clock multiplication gives the designer a number of design alternatives. For instance, a 50 MHz source clock doubled by the DLL can drive an FPGA design operating at 100 MHz. This technique can simplify board design because the clock path on the board no longer distributes such a high-speed signal. A multiplied clock also provides designers the option of time-domain multiplexing, using one circuit twice per clock cycle, consuming less area than two copies of the same circuit. Two DLLs can be connected in series to increase the effective clock multiplication factor to four. The DLL can also act as a clock mirror. By driving the DLL output off-chip and then back in again, the DLL can be used to deskew a boardlevel clock between multiple devices. The DLL can delay the completion of configuration until after DLL locks toguarantee the system clock is established prior to initiating the device. By taking advantage of the DLL to remove on-chip clock delay, the designer can greatly simplify and improve system-level design involving highfanout, high-performance clocks.

RELATED WORK

Delay-Locked Loop

As shown in Figure 1, a DLL in its simplest form consists of a variable delay line and control logic. The delay line produces a delayed version of the input clock CLKIN. The clock distribution network routes the clock to all internal registers and to the clock feedback CLKFB pin. The control logic must sample the input clock as well as the feedback clock in order to adjust the delay line.

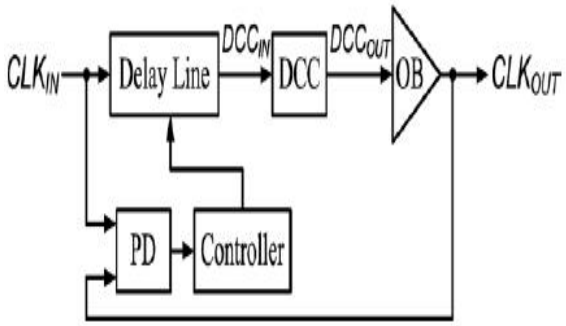
Figure 1: Delay Locked Loop



Delay lines can be built using a voltage controlled delayor as a series of discrete delay elements. For optimumperformance, the Virtex DLL uses a discrete digital delayline. A DLL works by inserting delay between the input clock and the feedback clock until the two rising edgesalign, putting the two clocks 360° out of phase (meaningthey are in phase). After the edges from the input clockline up with the edges from the feedback clock, the DLL"locks."As long as the circuit is not evaluated until afterthe DLL locks, the two clocks have no discernible difference. Thus, the DLL output clock compensates for the delay in the clock distribution network, effectively removing the delay between the source clock and its loads.

The circuit diagram of a conventional deskew buffer with duty cycle correction is illustrated in Figure 2. Its main architecture is based on DLL with a DCC circuitry in front of the output buffer (OB). Hence the output clock can satisfy the requirement of 50% duty cycle. The controller accepts the lead-lag information derived from the phase detector (PD) so as to adjust the output delay of the delay line. The DCC can be of either analog or digital type.

Figure 2: Conventional deskew buffer with DCC



B. Implementation

Implementation of the DLL or PLL can be accomplished using either analog or digital circuitry ;each holds its own advantages. An analog implementation with careful design can produce a DLL or PLL with a finer timing resolution. Analog implementations can additionally take less silicon area. Conversely, digital implementations offer advantages in noise sensitivity, lower power consumption, and jitter performance. Digital implementations also provide the ability to stop the clock, facilitating power management. Analog implementations can require additional power supplies, require close control of the power supply, and pose problems in migration to new process technologies.

C. DLL vs. PLL

When it comes to choosing between a PLL or a DLL for a particular application, the differences in the architectures must be understood. The oscillator used in the PLL inherently introduces in stability and an accumulation of phase error. This in turn degrades the performance of the PLL when attempting to compensate for the delay of the clock distribution network.

Conversely, the unconditionally stable DLL architecture does not accumulate phase error. For this reason, for delay compensation and clock conditioning, DLL architecture should be used. On the other hand, the PLL typically has an advantage when it comes to frequency synthesis.

EXPERIMENTAL EVALUAT ION

Figure 3 show the DLL primitives. These symbols provide access to the complete set of DLL features when implementing more complex applications.

Figure 3: standard DLL pin diagram CLKDLL

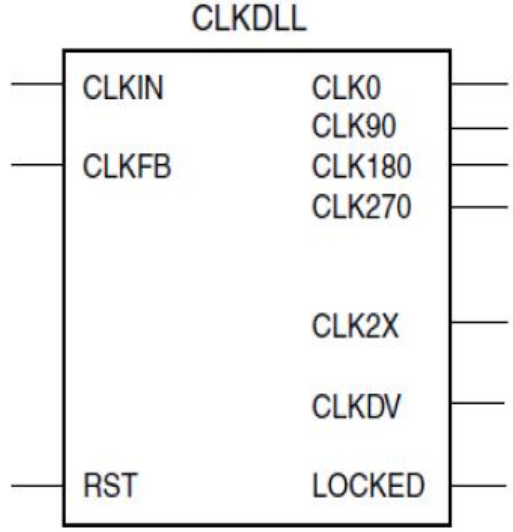


Table 1: Timing summary

Parameter	Seconds
Minimum period:	12.770ns
Minimum input arrival time before	12.056ns
Maximum output required time after	11.081ns
Maximum delay:	10.367ns

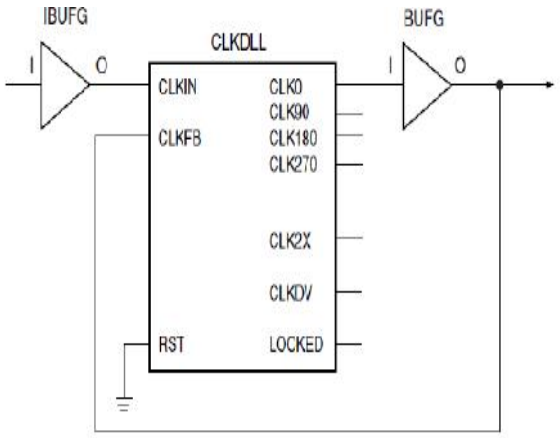
Table 2: Device utilization summary

Name	Used Blocks	Percentages (%)
Number of Slices	18 out of 192	9
Number of Slice Flip Flops	5 out of 384 (FDE:1 FDRR:4	1
Number of 4 input LUTs	32 out of 384	8
Number of bonded IOBs	7 out of 90 IBUF:6 OBUF:1	7
Number of GCLKs	1 out of 4	25

BUFGDLL Pin Descriptions

The BUFGDLL macro is used as the simplest way to provide zero propagation delay for a high fanout on-chip clock from an external input. This macro uses the IBUFG, CLKDLL, and BUFG primitives to implement the most basic DLL application, as shown in Figure 5.

Figure 4: BUFGDLL schematic



This symbol does not provide access to the advanced clock domain controls or to the clock multiplication or clock division features of the DLL. This symbol also does not provide access to the RST, or LOCKED pins of the DLL. For access to these features, a designer must use the library DLL primitives described in the following sections.

Source Clock Input — I

The I pin provides the user source clock, the clock signal on which the DLL operates, to the BUFGDLL. For the BUFGDLL macro, the source clock frequency must fall in the low frequency range as specified in the data sheet.

The BUFGDLL requires an external signal source clock. Therefore, only an external input port can source the signal that drives the BUFGDLL I pin.

Clock Output — O

The clock output pin O represents a delay-compensated version of the source clock (I) signal. This signal, sourced by a global clock buffer BUFG symbol, takes advantage of the dedicated global clock routing resources of the device. The library CLKDLL primitives provide access to the complete set of DLL features needed when implementing more complex applications with the DLL.

Source Clock Input — CLKIN

The CLKIN pin provides the user source clock (the clock signal on which the DLL operates) to the DLL. The

CLKIN frequency must fall in the ranges specified in the data sheet. The clock input signal can be provided by one of the following:

- BUFG — Internal global clock buffer
- IBUFG — Global clock input buffer on the same edge of the device (top or bottom)
- IO_LVDS_DLL — the pin adjacent to a global clock pin.

Feedback Clock Input — CLKFB

The DLL requires a reference or feedback signal to provide the delay-compensated output. Only the CLK0 or CLK2X DLL outputs should be connected to the feedback clock input(CLKFB) pin to provide the necessary feedback to the DLL. The feedback clock input signal can be driven by an internal global clock buffer (BUFG), one of the global clock input buffers(IBUFG) on the same edge of the device (top or bottom)If an IBUFG sources the CLKFB pin, the following special rules apply:

1. An external input port must source the signal that drives the IBUFG input pin.

2. The CLK2X output must feedback to the device if both the CLK0 and CLK2X outputs aredriving off-chip devices.
3. That signal must directly drive only OBUFs and nothing else.
4. These rules enable the software determine which DLL clock output sources the CLKFB pin.

Reset Input — RST

When the reset pin RST activates, the LOCKED signal deactivates within four source clock cycles. The RST pin, active High, must either connect to a dynamic signal or tied to ground. As the DLL delay taps reset to zero, glitches can occur on the DLL clock output pins.

Activation of the RST pin can also severely affect the duty cycle of the clock output pins. Furthermore, the DLL output clocks no longer deskew with respect to one another. The DLL must be reset:

1. When the input clock frequency changes.
2. If the device is reconfigured in Boundary-Scan mode.
3. If the device undergoes a hot swap.
4. After the device is configured if the input clock is not stable during the startup sequence.

In each case, the DLL RST pin can be driven by internal logic or routed to a user I/O and driven externally.

2x Clock Output — CLK2X

The output pin CLK2X provides a frequency-doubled clock with an automatic 50/50 duty-cycle correction.

Until the CLKDLL has achieved lock, the CLK2X output appears as a 1x version of the input clock with a 25/75 duty cycle. This behavior allows the DLL to lock on the correct edge with respect to source clock. This pin is not available on the CLKDLLHF primitive.

Clock Divide Output — CLKDV

The clock divide output pin CLKDV provides a lower frequency version of the source clock.

TheCLKDV_DIVIDE property controls CLKDV such that the source clock is divided by N where Nis either 1.5, 2, 2.5, 3, 4, 5, 8, or 16.

The CLKDV output pin is provided with automatic duty cycle correction. The CLKDV output pin has a 50/50 duty cycle only for all values of the division factor N except for non-integer division in High Frequency (HF) mode.

For division factor 1.5 the duty cycle in the HF mode is 33.3%High and 66.7% Low. For division factor 2.5, the duty cycle in the HF mode is 40.0% High and60.0% Low.

1x Clock Outputs — CLK[0|90|180|270]

Table 3 Relationship of phase shifted output clock to period shift.

Phase (degrees)	Period shift
0	0%
90	25%
180	50%
270	75%

The DLL provides duty cycle correction on all 1x clock outputs such that all 1x clock outputs by default have a50/50 duty cycle. The DUTY_CYCLE_CORRECTIONproperty (TRUE by default), controls this feature. In orderto deactivate the DLL duty cycle correction, attachtheDUTY_CYCLE_CORRECTION=FALSE

property tothe DLL symbol. When duty cycle correction deactivates,the output clock has the same duty cycle as the sourceclock.

The DLL clock outputs can drive an OBUF, aBUFG, or they can route directly to destination clockpins. The DLL clock outputs can only drive the BUFGsthat reside on the same edge (top or bottom).

Locked Output — LOCKED

In order to achieve lock, the DLL might need to sampleseveral thousand clock cycles. After the DLL achieveslock, the LOCKED signal activates. The DLL timingparameter section of the data sheet provides estimates forlocking times. To guarantee that the system clock isestablished prior to the device "waking up," the DLL candelay the completion of the device configuration processuntil after the DLL locks. The STARTUP_WAITproperty activates this feature.

Until the LOCKED signal activates, the DLL outputclocks are not valid and can exhibit glitches ,spikes, orother spurious movement. In particular, the CLK2Xoutput appears as a 1x clock with a 25/75 duty cycle.

FUTHER IMPLEMENTATION

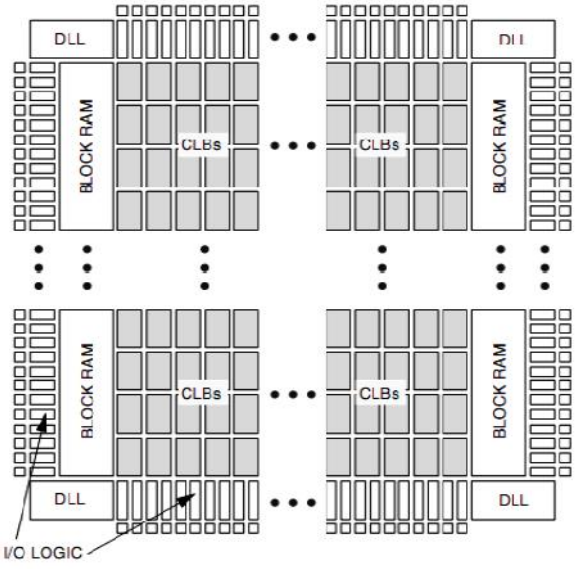
Figure 5 is composed of five major configurableelements, IOBs provide the interface between the packagepins and the internal logic , CLBs provide the functionalelements for constructing most logic, Dedicated blockRAM memories of 4096 bits each, Clock DLLs for clockdistributiondelay compensation and clock domaincontrol, Versatile multi-level interconnect structureAs can be seen in Figure 3, the CLBs form the centrallogic structure with easy access to all support and routingstructures. The IOBs are located around all the logic andmemory elements for easy and quick routing of signals onand off the chip.Values stored in static memory cells control all theconfigurable logic elements and interconnect resources.

These values load into the memory cells on power-up, and can reload if necessary to change the function of the device. Each of these elements will be discussed in detail in the following sections.

Configurable Logic Block

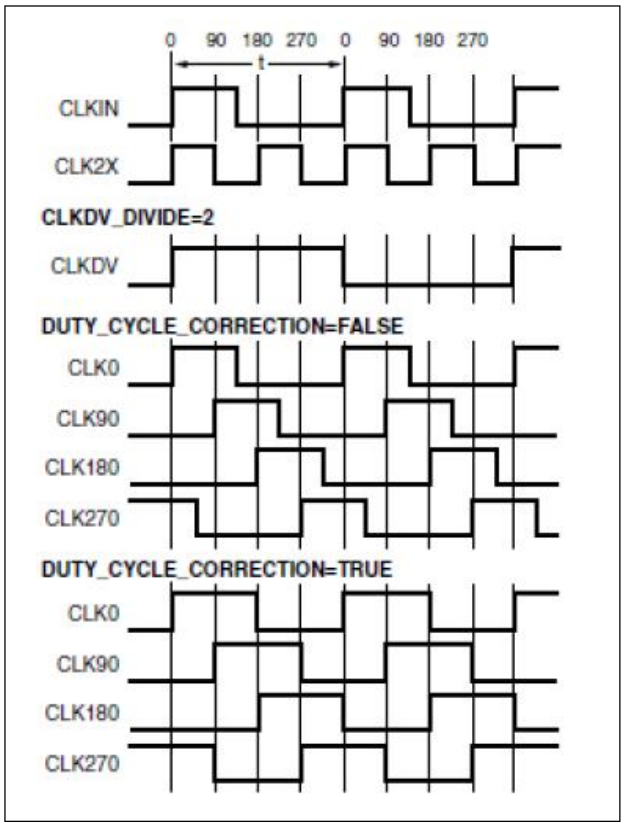
The basic building block of the Spartan-IIE FPGA CLB is the logic cell (LC). An LC includes a 4-input function generator, carry logic, and storage element. The output from the function generator in each LC drives the CLB output or the D input of the flip-flop. Each Spartan-IIE FPGA CLB contains four LCs, organized in two similar slices

Figure 5: FPGA architecture with DLL



DLL output characteristics

The DLL provides duty cycle correction on all 1x clock outputs such that all 1x clock outputs by default have a 50/50 duty cycle. The DUTY_CYCLE_CORRECTION property (TRUE



by default), controls this feature. In order to deactivate the DLL duty cycle correction, attach theDUTY_CYCLE_CORRECTION=FALSE property to the DLL symbol. When duty cycle correction deactivates, the output clock has the same duty cycle as the source clock.

The DLL clock outputs can drive an OBUF, a BUFG, or they can route directly to destination clock pins. The DLL clock outputs can only drive the BUFGs that reside on the same edge (top or bottom).

CONCLUSION

DLL is widely used in wireless communication systems as well as telecommunication system. These dedicated DLLs can be used to implement several circuits, which improve and simplify system-level design. FPGA implementation ensures easy and computer control over the system. This paper has discussed the PLL basic structure and designed a PLL that can be used efficientlyfor device synchronization purpose. Xilinx is used fordesigning the schematic diagram using the RTL logic.And also designed FPGA with DLL. TheVirtex™ FPGA series offers fully digital dedicated onchipDelay-Locked Loop (DLL) circuits providing zeropropagation delay, low clock skew between output clocksignals distributed throughout the device, and advancedclock domain control.

REFERENCES

1. Martin Kumm, HaraldKlingbeil, Peter Zip (2009),“ An FPGA Based Linear All-Digital delay -Locked Loop”, *IEEE transactions on circuits and systems—i:regularpapers*, September.
2. Wang J S, Cheng C Y, Liu J S, Liu Y C and Wang Y M (2010), “Aduty-cycle-distortion-tolerant half-delay-linelow-power fast-lock-inall-digital delay-locked loop”, *IEEE J. Solid-State Circuits*, Vol. 45, No. 5, pp. 1036-1047.
3. Chang H H and Liu S I (2005), “A wide-range and fastlockingall-digitalcycle-controlled delay-locked loop”, *IEEE J. Solid-State Circuits*, Vol. 40, No. 3, pp. 661–670.
4. Kao S K and Liu S I (2007), “A 62.5 i“dá“25-MHz anti-resetall digital delay locked loop”, *IEEE Trans. Circuits Syst. II, Exp. Briefs*, Vol. 54, No. 7,pp. 566–570.
5. Hatakeyama A, Mochizuki H, Aikawa T, and Takita M (1997), “A 256-MbSDRAM using a register-controlleddigital DLL”, *IEEE J. Solid-State Circuits*, Vol. 32, No. 11, pp. 1728–1734.
6. Kwak J T, Kwon C K, Kim K W, Lee S H,

and Kih J S (2003), "Lowcost high performance register-controlled digital DLL for 1 Gbps 32DDR SDRAM", in *IEEE Int.Symp. VLSI Circuits Dig. Tech. Papers*, pp. 283–284.

7. Dehng D K and Liu S I (2000), "Clock-deskew buffer using a SAR-controlled delay-locked loop", *IEEE J. Solid-State Circuits*, Vol. 35, No. 8, pp. 1128–1136.
8. Anu Gupta, Mohammad Waqar Ahamed, Abhishek Dhir (2011), "Novel method to implement highfrequency All Digital Phase-Locked Loop on FPGA,International Conference on VLSI Communication & Instrumentation (ICVCI)", *Proceedings published byInternational Journal of Computer Applications(IJCA).*
9. Klingbeil H (2005) "A fast DSP-based phase-detector forclosedloop RF control in synchrotrons", *IEEE Trans.Instrum. Meas.,* Vol. 54, No. 3, pp.1209 -1213.
10. Namgoong W (2010), "Observer-controller based digitalPLL", *IEEE Trans. Circuits Syst. I*, Reg. Papers, Vol. 57, No. 3, pp. 631-641.
11. en.wikipedia.org/wiki/delay locked loop.
12. digitalvlsi design with verilog: a textbook fromsilicon valley technical institute-john Williams.
13. Verilog Designers library-Bob Zeidman

International Journal of Engineering Research and Science & Technology
Hyderabad, INDIA. Ph: +91-09441351700, 09059645577
E-mail: editorijlerst@gmail.com or editor@ijerst.com
Website: www.ijerst.com