Delay Locked Loop with Double Edge Synchronization

Goran S. Jovanović[1] and Mile K. Stojčev[2]

Abstract – **In CMOS multistage clock buffer design, the duty-cycle of clock is liable to be changed when the clock passes through several buffer stages. The pulse-width may be changed due to unbalance of the *p* and *n* MOS transistors in the long buffer. This paper describes a delay locked loop with double edge synchronization for use in a clock alignment process. Results of its SPICE simulation, that relate to 1.2μm CMOS technology, shown that the duty-cycle of the multistage output pulses can be precisely adjusted to (50±1)% within the operating frequency range, from 55MHz up to 166MHz.**

Keywords – **DLL, Duty Cycle Corrector, Delay Line.**

I. INTRODUCTION

Almost all contemporary digital VLSI systems and other digital systems rely on clock pulses to control the movement of data. To reach the highest circuit speed in CMOS applications, the clock distribution system must be carefully designed. A great deal of attention has been paid to clock recovery, clock regeneration, timing, and distribution [1].

Automatic control techniques, such as Phase-Locked Loop (PLL) and Delay-Locked Loop (DLL) have been widely used in high-speed clock alignment applications such as double-data rate (DDR) SDRAMs, pipelined microprocessors, network processors, etc. [2].

In a PLL implementation the chip has its own reference clock oscillator (VCO) that is phase-locked to an external reference clock. In general, a PLL clock aligner is superior in applications where noise on the reference clock dominates, and self-induced jitter within the VCO is negligible. On the other hand, a DLL provides superior performance when a clean reference clock is available. A DLL is commonly used to lock the phase of the buffered clock to that of the input data. Typically, we meet this in applications where no clock synthesis is required, such as often the situation for multi-chip digital systems with well-designed system clock distribution network [2].

In high-speed design a multistage clock buffer implemented with a long inverter chain is often needed to drive a heavy capacitive load. For these designs, as well as for applications in which the timing of both edges of the clock is critical [10], it is difficult to keep the clock duty cycle at its ideal value 50 %, primarily due to various asymmetries in signal paths and unbalances of the *p* and *n* transistors in the long buffer. As a consequence the clock duty cycle will deteriorate from 50 %, and in the worst case, the clock pulse may disappear inside the clock buffer, as the pulse width becomes too narrow or too wide [6].

Duty-cycle distortion is usually addressed in PLLs by simply running the PLL's VCO at twice the system frequency and using a post divider triggered on one edge of the VCO output to produce the output clock of the PLL. This ensures good 50 % duty cycle. In a DLL, however no frequency multiplication is possible. Therefore, the duty-cycle of the output signal must be corrected to 50 %. A conventional solution is to attach duty-cycle correction circuit to the clock output driver with the price of added area [4].

In this paper, we describe a new structure of a DLL circuit with clock alignment capability of both leading and trailing output pulse edges. This circuit can be used to obtain correct the duty-cycle factor (50 %) in a multistage clock buffer.

II. CLOCK ALIGNERS

The goal of clock distribution network is to organize clocks so that the delays from the source point of each clock or clock phase to its destination points are identical. In reality, however, no matter how each clock path is constructed, due to variations in wire delays and driver delays, no uniform and possibly time-varying clock load, and negative effects of supply and substrate noise, any two paths within the VLSI IC will always have a delay difference. A clock aligner's task is to phase-align a chip internal clock with a reference clock, effectively removing the variable buffer delay and reducing uncertainty in clock phase between communicating VLSI IC constituents. Clock aligners (see Fig. 1) can be built using either PLLs or DLLs.

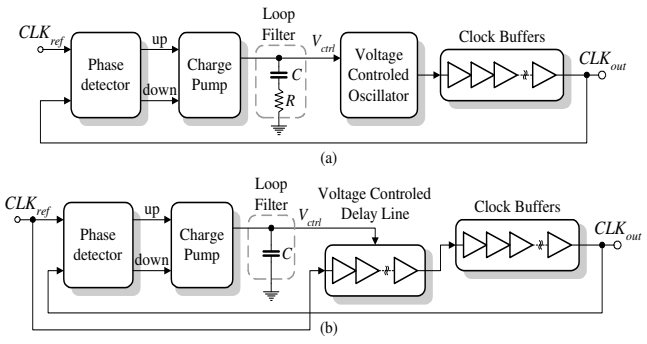

Fig. 1. Clock aligner implementations: (a) PLL clock aligner, (b) DLL clock aligner

In a PLL implementation (Fig. 1(a)) the circuit has its own oscillator (VCO) that is phase-locked to a reference clock. The phase shift introduced by the buffer delay, T_B, is assumed to be changing as a consequence of wiring delay, temperature

[1]Goran S. Jovanović is with the Faculty of Electronic Engineering, University of Niš, Aleksandra Medvedova 14, 18000 Niš, Serbia and Montenegro, Email: joga@elfak.ni.ac.yu

[2]Mile K. Stojčev is with the Faculty of Electronic Engineering, University of Niš, Aleksandra Medvedova 14, 18000 Niš, Serbia and Montenegro, Email: stojcev@elfak.ni.ac.yu

and voltage variations, etc. The buffer's delay is eliminated by inclusion in the control loop.

In DLL implementation (Fig. 1(b)) a variable delay line (VCDL) is inserted between the reference clock, CLK_{ref}, and the output clock, CLK_{out}. The delay is regulated so that $T_D+T_B=N*T_{ref}$ (usually N=1). The buffer's delay is placed within a control loop and is eliminated.

PLL and DLL implementations have complementary advantages and disadvantages. Table I summarizes the main features of PLL and DLL clock aligners.

TABLE I COMPARISONS OF PLL AND DLL

PLL	DLL
– jitter accumulation	– no jitter accumulation
– higher-order system	– 1st-order system
– can be unstable	– always stable
– hard to design	– easier to design
– costly to integrate LF	– easier to integrate LF
– less referent signal dependent	– referent signal dependent
– easy frequency multiplication	– difficult frequency multiplication
	– limited locking range

Many factors control the speed of CMOS ICs. There are device dimensions, clocking strategy, architecture, clock distribution, etc. Here we focus our attention on clocking strategy and clock distribution problems.

To meet the demand for high-speed operation today, many systems adopt a double data rate (DDR) technologies, such as DDR SDRAM, double sampling ADC, clock and data recovery circuits, microprocessor circuits, etc. In these systems, both rising and falling edges of the clock are used to sample the input data, requiring that the duty-cycle of the clock be precisely maintained at 50%. Therefore, how to generate a clock with precise 50% duty-cycle for high-speed operation is an important issue. Namely, in high-frequency operations, clock outputs with a short cycle time can be severely distorted as clock passes through many delay cells. Even if the duty cycle of CLK_{ref} is 50% at the entrance that of CLK_{out} may deviate significantly from 50%. As a consequence, it can cause the output to have phase error, which could be fatal, especially in high-speed communication applications.

A conventional solution is to attach duty-cycle correction circuits to all output drivers with the price of added area, increased jitter, and further phase mismatch due to enlarged path [4]. There are several different methods for implementing 50% duty-cycle correctors, both in PLL and DLL control loops, intended to adjust the output duty-cycle of the multistage driver [6-9]. Each of these methods has its advantages and drawbacks. In all these circuits, the variable delay element is one of the key building blocks. Its precision directly affects the overall performance of the circuit.

III. DLL WITH DOUBLE EDGE SYNCHRONIZATION

The structure of the proposed Delay Locked Loop with Double Edge Synchronization (DLL–DES) clock alignment circuit is pictured in Fig. 2. The click aligner is composed of a voltage controlled delay line, VCDL, two phase detectors, PD1 and PD2, two charge-pumps, CP1 and CP2, two first order low-pass filters, LP1 and LP2, and a multistage clock buffer, CB. The negative feedback in the loop adjusts the delay through the VCDL by integrating the phase shift errors that result between the periodic reference input, CLK_{in}, and the multistage output, CLK_{out}.

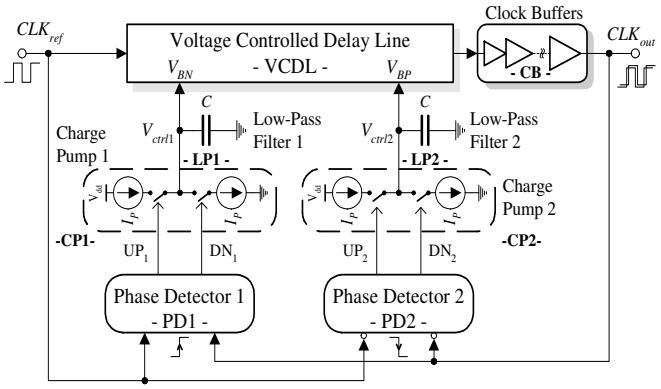

Fig. 2. DLL's architecture with double edges synchronization

The underlying idea for this approach is to provide delay regulation for both a rising and trailing edge of the output clock pulse CLK_{out}. For implementation of variable delay regulation the building block VCDL is used. The control voltage V_{BN} (V_{BP}) defines delay regulation of a rising (trailing) clock pulse edge. The phase detector PD1 (PD2) compares phase shifts of rising (trailing) edges between the input, CLK_{in}, and output, CLK_{out}, clock pulses. UP1 (UP2) pulses cause I_p to add charge to loop filter capacitor C, whereas DN1 (DN2) pulses remove charge. The LP1's (LP2's) output, $V_{ctrl}1$ (V_{ctrl2}), is connected to the VCDL control input at node V_{BN} (V_{BP}). When the system, from Fig. 2, enters in stable state both edges of CLK_{out} are synchronized and phase shifted in respect to the referent clock CLK_{in}. An important feature of this architecture is that the duty-cycle of CLK_{out} is maintained at value of 50 %.

IV. CIRCUITS IMPLEMENTATION

In the sequel we will describe, in more details, the structure and principle of operation of each constituent of a DLL-DES based clock aligner.

A. Voltage controlled delay line

The actual implementation of a VCDL is comprised of a number of cascaded variable delay buffers. Each delay buffer (adjustable timing element) is of identical structure. Current starved delay element (CSDE) was chosen as a convenient candidate for realization of the delay buffer. The main design decision for such a choice was the following: CSDE provides independent delay regulation of both rising and falling clock pulse edges. Independent delay regulation can be achieved by varying the current of *p* and *n* MOS transistors.

In conventional CSDE (see Fig. 3(a)) a single control voltage V_{ctrl}, generated by a bias circuit, modulates the on resistance of pull-down M_3, and through a current mirror,

pull-up M_4 [5]. The variable resistances control the current available to charge or discharge the parasitic load capacitance.

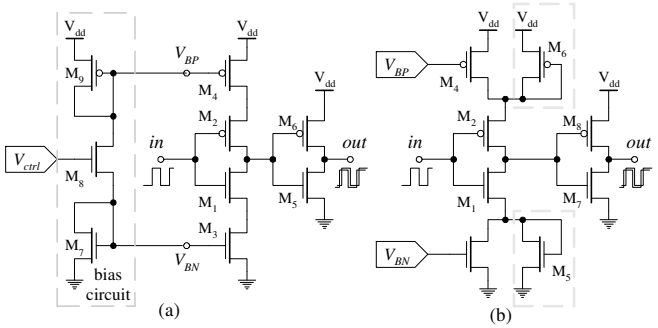

Fig. 3. Modified current starved delay element

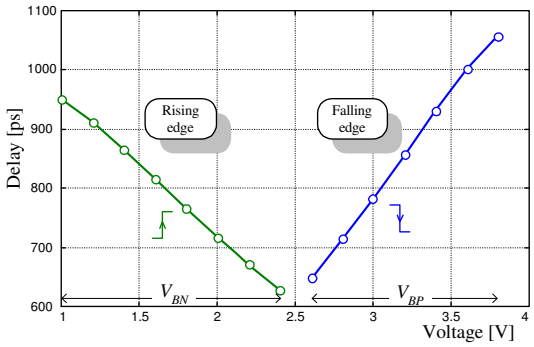

Fig. 4. Rising and Falling edge delay in term of control voltages

In order to achieve independent, instead of single, variable resistances control, we propose here a modified version of CSDE, as one given in Fig. 3(b). In our approach, both control voltages, V_{BN} and V_{BP}, directly drive gates of M_3 and M_4 MOS transistors, respectively. Transistors M_5 and M_6 act as symmetric loads and are used for two purposes: a) to linearize a voltage-to-delay transfer function of the CSDE; and b) provides correct initial condition for DLL operation even in a case when both control voltages V_{BN} and V_{BP} are out-of-regulation limits (for example, M_4 and M_3 are switched off). The modified CSDE was designed for 1.2µm CMOS technology, for 5V power supply voltage. A SPICE simulation results that correspond to delay functions of both rising and falling pulse edges are given in Fig. 4. The obtained results in Fig. 4 show that linear regulation of voltage-versus-delay can be achieved. In general, CSDE offers good delay line stability in respect to temperature and supply voltage variations. Its main disadvantage is relatively limited range of delay regulation, i.e. low-sensitivity.

B. Phase detector

The phase detector measures the phase difference between the time reference and the delay chain. High precision dynamic phase detection circuit based on true single phase logic [3] is adopted in our design. The main advantages of this circuit are simple hardware structure, high-speed of operation, and small dead zone [5]. The UP*x* and DN*x* (*x* refers to 1 or 2) are used to control the charge-pump circuit CP*x*. The PD1 (PD2) is sensitive to rising (falling) clock pulse edge. A modification, in respect to standard solution [5], is performed by substituting MOS transistors P_{12}, N_{12}, P_{22}, and N_{22} (see Fig. 5 (a)) with complementary ones N_{11}, P_{13}, N_{21}, and P_{23} (see Fig. 5(b), respectively.

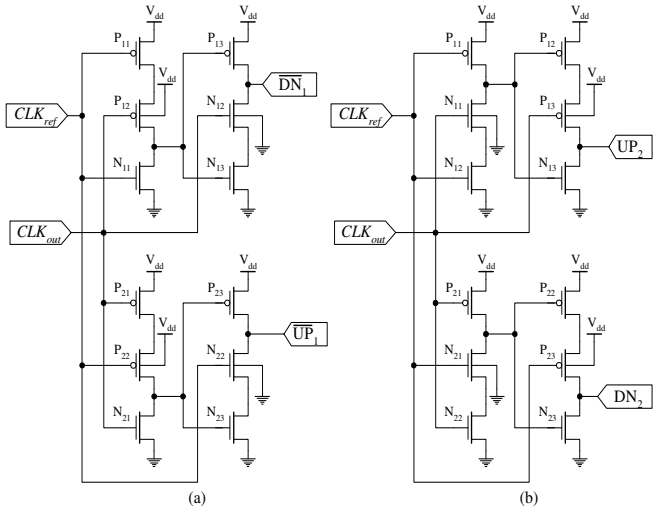

Fig. 5. Implementation of phase detectors for (a) raising and (b) falling edges

Operational principles of PD1 and PD2 are shown in Fig. 6. The widths of UP and DN signals are proportional to the phase difference of the input signals. Fig. 6a (6b) shows the operation of PD1 (PD2). Waveforms on the left side of Fig. 6a (6b) correspond to a case when the signal CLK_{out} (see Fig. 2) leads in respect to the signal CLK_{ref}. Otherwise, timing diagrams on the right side are valid (CLK_{ref} leads the CLK_{out} signal).

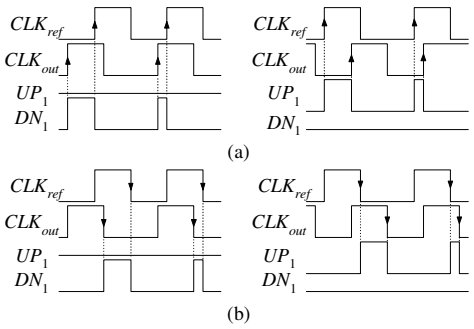

Fig. 6. Waveforms of input and output signals for (a) phase detector 1 and (b) phase detector 2

C. Charge pump and loop filter

The charge-pump and loop filter structure is presented in Fig. 7. Transistors P_1 and N_1 act as switching elements driven by pulses UP and DN, while transistors P_2 and N_2 are employed as current sink and source, respectively. The charge-pump charges or discharges the filter capacitor, C. The voltage on this capacitor, V_{ctrl} (V_{BP} or V_{BN} in Fig. 2), sets the VCDL stage propagation delay. The charge-pump the realization of an integrator transfer function with no additional active amplifier, resulting in a zero-phase error in steady state. A small capacitor, *C*, is used for the low-pass loop filter. The current level of the charge-pump and the charge delive-

red/accepted at every rising/falling clock edge transition are set to a small value [5]. This allows the implementation of the loop capacitor on chip.

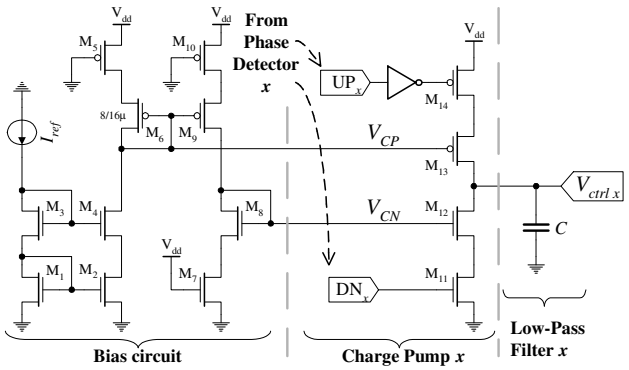

Fig. 7. Current pump and loop filter

The bias circuit provides correct operation of the charge pump. Its structure is given on the left side of Fig. 7. This circuit generates two control voltages, V_{CP} and V_{CN}. These voltages define the charge and discharge currents of loop capacitor, C, that pass through transistors M_{12} and M_{13}.

V. Simulation Results

The DES-DLL sketched in Fig. 2 is implemented in 1.2 μm CMOS technology. It is supplied with V_{dd}=5V. SPICE simulation results that relate to referent clock excitation f_{ref}=80MHz are given in Fig. 8. The DES-DLL is operatives within the frequency range from 55 MHz up to 166 MHz.

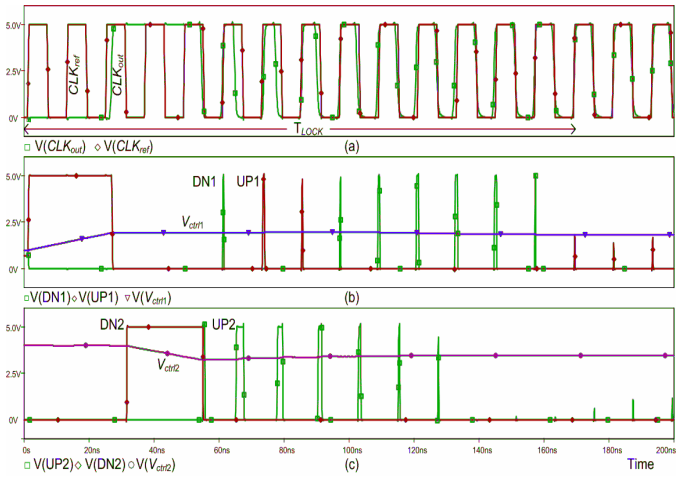

Fig. 8. Simulation of DLL with Double Edges Synchronization

Timing diagrams that correspond to referent input clock pulses, CLK_{ref}, and buffer output, CLK_{out}, are given in Fig. 8(a). This Fig. shows that the locking time, T_{LOCK}, between the referent CLK_{ref} and output CLK_{out} pulses is less than 200ns. We define T_{LOCK} as a time interval starting from initial condition up to the instant when total coincidence of rising and falling edges between both pulses, CLK_{ref} and CLK_{out}, exists, (see Fig. 8(a)). If we assume that CLK_{ref} is symmetrical then the coincidence corresponds to 50% duty-cycle of CLK_{out}. According to numerous simulations, within the operating range, duty cycle deviations are less then 1%. Compared to the results presented in [6-9] we obtain identical or better accuracy in duty cycle regulation.

Fig. 8(b) (8(c)) deals with waveforms that are obtained at the outputs UP1 (UP2), DN1 (DN2), and V_{ctrl1} (V_{ctrl2}). As can be seen from Fig. 8(b) (8(c)) UPx and DNx signals define the control voltage V_{ctrlx} during the transition period ($0 < t < T_{LOCK}$). After that the system enters in stable state and UPx and DNx signals disappear and $V_{ctrl\,x}$ takes a constant value.

According to the obtained results we can conclude that the proposed DES–DLL can be seen to have a wide-operational range and good duty–cycle correction capability.

VI. Conclusion

In this paper a new DLL architecture with clock alignment capability of both leading and trailing edges is described. The proposed circuit was simulated using models for 1.2 μm CMOS technology and SPICE simulator. The clock aligner has been designed specifically to correct precisely the duty-cycle factor in a multistage clock buffer to (50±1) % within the operating frequency range from 55 MHz up to 166 MHz. Timing diagrams are measured by simulation. The proposed DLL based clock aligner keeps the same benefits of conventional DLL's such as good absolute stability, fast-response, and low-level output jitter for both (rising and falling) edges. Such circuits serve in many applications including clock distribution network within the VLSI ICs, high-speed DRAM, and core-to-core interconnects within a system-on-chip designs.

References

[1] Oklobdzija V., Stojanovic M., Markovic D., and Nedovic N., *Digital System Clocking: High-Performance and Low-Power Aspects*, John Wiley & Sons Pub., Hoboken, New Jersey, 2003.

[2] Razavi B., ed., *Phase-Locking in High-Performance Systems: From Devices to Architectures*, John Wiley & Sons Pub., Hoboken, New Jersey, 2003.

[3] Yuan J., and Svensson C., *High-Speed CMOS Circuit Technique*, IEEE JSSC, vol.24, No.1, pp.62-70, February 1989.

[4] Garleep B., et al., *A Portable Digital DLL for High-Speed CMOS Clock Buffers*, IEEE JSSC, Vol. 34, No. 5, pp. 632-643, May 1999.

[5] Moon Y., et al., *An All-Analog Multiphase DLL Using a Replica Delay Line for Wide-Range Operation and Low-Jitter Performance*, IEEE JSSC, Vol. 35, No. 3, pp. 377-384, March 2000.

[6] Fenghao M., Svensson C., *Pulse Control Loop in High-Speed CMOS Clock Buffers*, IEEE JSSC, Vol. 35, No. 2, pp. 134-141, February, 2002.

[7] Han S.-R., Liu S.-I., *A 500MHz- 1.25GHz Fast-Locking Pulsewidth Control Loop with Presettable Duty-Cycle*, IEEE JSSC, Vol. 39, No. 3, pp. 463-468, March 2004.

[8] Yang P.-H., Wang J.-S., *Low Voltage Pulsewidth Control Loops for SoC Applications*, IEEE JSSC, Vol. 37, No. 10, pp. 1348-1351, October 2003.

[9] Liu W.- M., Huang H.- Y., *A Low-Jitter Mutual-Correlated Pulsewidth Control Loop Circuit*, IEEE JSSC, Vol. 39, No. 8, pp. 1366- 1369, August 2004.

[10] Kim C., Kang S., *A Low-Swing Clock Double-Edge triggered Flip-Flop*, IEEE JSSC, Vol. 37, No. 5, pp. 648-652, May, 2002.