

Glitch-Free NAND-Based Digitally Controlled Delay-Lines

Davide De Caro, *Senior Member, IEEE*

Abstract—The recently proposed NAND-based digitally controlled delay-lines (DCDL) present a glitching problem which may limit their employ in many applications. This paper presents a glitch-free NAND-based DCDL which overcame this limitation by opening the employ of NAND-based DCDLs in a wide range of applications. The proposed NAND-based DCDL maintains the same resolution and minimum delay of previously proposed NAND-based DCDL. The theoretical demonstration of the glitch-free operation of proposed DCDL is also derived in the paper. Following this analysis, three driving circuits for the delay control-bits are also proposed. Proposed DCDLs have been designed in a 90-nm CMOS technology and compared, in this technology, to the state-of-the-art. Simulation results show that novel circuits result in the lowest resolution, with a little worsening of the minimum delay with respect to the previously proposed DCDL with the lowest delay. Simulations also confirm the correctness of developed glitching model and sizing strategy. As example application, proposed DCDL is used to realize an All-digital spread-spectrum clock generator (SSCG). The employ of proposed DCDL in this circuit allows to reduce the peak-to-peak absolute output jitter of more than the 40% with respect to a SSCG using three-state inverter based DCDLs.

Index Terms—All-digital delay-locked loop (ADDLL), all-digital phase-locked loop (ADPLL), delay-line, digitally controlled oscillator (DCO), flip-flops, sense amplifier, spread-spectrum clock generator (SSCG).

I. INTRODUCTION

IN RECENT deep-submicrometer CMOS processes, time-domain resolution of a digital signal is becoming higher than voltage resolution of analog signals [1]. This claim is nowadays pushing toward a new circuit design paradigm in which the traditional analog signal processing is expected to be progressively substituted by the processing of times in the digital domain. Within this novel paradigm, digitally controlled delay lines (DCDL) should play the role of digital-to-analog converters in traditional, analog-intensive, circuits. From a more practical point of view, nowadays, DCDLs are a key block in a number of applications, like all-digital PLL (ADPLL) [2]–[8], all-digital DLL (ADDLL) [9]–[16], all-digital spread-spectrum clock generators (SSCGs) [17], [18], and ultra-wide band (UWB) receivers with ranging feature [19], [20].

Manuscript received August 04, 2011; accepted December 07, 2011. Date of publication January 23, 2012; date of current version December 19, 2012.

The author is with the Department of Electronic Engineering, University of Napoli “Federico II”, 80125 Napoli, Italy (e-mail: dadecaro@unina.it).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TVLSI.2011.2181547

The “classical” approach [3]–[6], [9], [17] to design a DCDL is using a delay-cells chain and a MUX to select the desired cell output. In these mux-based DCDLs, the MUX delay increases with the increase of the number of cells. This results in a tradeoff between the delay range and minimum delay (t_{\min}) of the DCDL. It is worth to note that t_{\min} is a critical design parameter in many application. As an example in ADPLL/ADDLL, t_{\min} determines the maximum output frequency of the circuit. This property remains true also for the All-digital SSCG of [18], where a correct DCDL synchronization is obtained only by imposing that t_{\min} is lower than one half input clock period.

The large t_{\min} of MUX-based DCDLs can be reduced by using a tree-based multiplexer topology [3]. This however results in an irregular structure which complicates layout design and, consequently, also increases the nonlinearity of the DCDL.

The DCDL topology employed in [10] and [11] uses again a delay cells chain. Differently from the above approaches, in this technique each cell is constructed by using NAND gates. This apparently solves the tradeoff related to the MUX of previous structures. A deepen analysis of the structure, however, reveals that the input capacitance of the DCDL increases linearly with the number of cells. This, clearly, reintroduces a tradeoff between the number of cells and the minimum delay, similarly to MUX-based DCDLs. A similar reasoning applies also to the MUX-based DCDL proposed in [16].

In [12]–[15], the DCDL is constructed by using a regular cascade of equal delay elements (DE). In this circuit, the multiplexer of previous DCDL is conceptually spread among all cells. In this way the minimum delay t_{\min} is very low and becomes independent of the number of cells. In addition the highly regular topology allows a simple layout organization [18] which provides very low nonlinearity layout effects. Each DE in [12]–[15] is constructed by using only NAND gates, obtaining a very good linearity and resolution. An analysis of the circuit reveals that the DCDL resolution (t_R) is given by $2 \cdot t_{\text{NAND}}$ (t_{NAND} being the delay of a NAND gate).

The DCDL proposed in [8] uses again a structure of cascaded delay elements. Differently from [12]–[15], each element is constructed by using three-state inverters (TINV), obtaining a resolution $t_R = 2 \cdot t_{\text{TINV}}$. Since the pull-up network of a TINV requires two series devices whereas a NAND gate uses a single device in the pull-up, we can expect that the resolution of this solution is higher than the resolution of NAND-based DCDLs [12]–[15].

The DCDL proposed in [18] is also based on a cascade of equal delay elements, which allows a simple layout organization. In this case each delay element is constructed by using an inverter and an inverting multiplexer. This INVERTER +

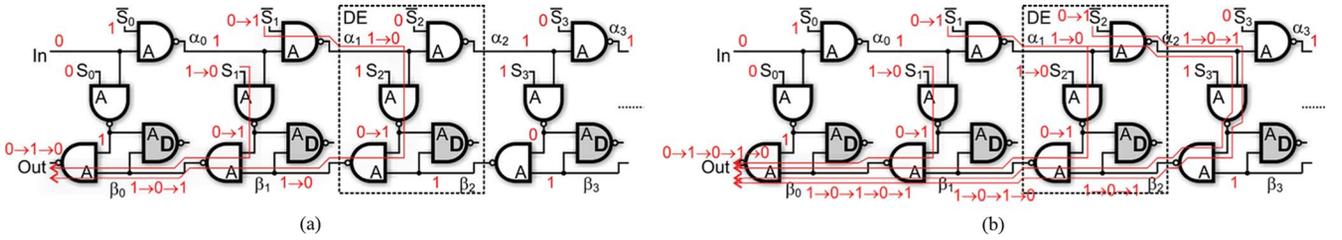


Fig. 1. Glitching problem of NAND-based DCDL [12]–[15] (a) glitching when the delay control-code increases by one; (b) glitching when the delay control-code increases by two.

MUX topology, however, presents two drawbacks. A first weakness is due to the different delays of the inverter and the multiplexer which results in a t_{\min} mismatch between odd and even control-codes. This mismatch results in an increased INL. A second drawback is due to the large multiplexer delay, which provides a resolution higher than the resolution of both NAND-based DCDLs and TINV-based DCDLs.

Glitching is a common design problem in systems employing DCDLs. In the most common applications, DCDLs are employed to process clock signals, therefore a glitch-free operation is required. A necessary condition to avoid glitching is designing a DCDL which have no-glitch in presence of a delay control-code switching. This is an issue at the DCDL-design level. Many approaches are known to avoid glitching in mux-based DCDLs [21]–[23]. It is interesting to observe that the DCDL topologies of [8] and [18], from a logical point of view, correspond to distributed MUX-based structure. Glitching in these topologies can be avoided by using a thermometer code for the control-bits, or using the approach of [23]. On the other hand the NAND-based DCDL topology of [12]–[15] presents a glitching problem that, to the best of our knowledge, is still not known in Literature. It is worth to note that in the ADDLL topologies of [12]–[15] the DCDL glitching is filtered by the phase detector and harmonic locking circuitry during locking phase. In other applications, however, the presence of this glitching phenomenon can substantially limit the employ of NAND-based DCDLs. This represents a substantial drawback of this topology in comparison to the solutions of [8] and [18]. The errors that in some applications can originate from DCDL glitching will be also discussed within this paper.

This paper gives two contributions to the design of NAND-based DCDLs. First it is shown and analyzed the glitching problem of the NAND-based DCDL of [12]–[15]. Afterwards a novel glitch-free NAND-based DCDL is presented. The proposed NAND-based DCDL allows to achieve a resolution $t_R = 2 \cdot t_{\text{NAND}}$, similarly to the NAND-based DCDL of [12]–[15].

The paper is organized as follows. The NAND-based DCDL of [12]–[15] is recalled in Section II. In the same section the glitching problem of this DCDL is analyzed. The structure of proposed, glitch-free, NAND-based DCDL is presented in Section III. Section IV analyzes theoretically the novel DCDL structure by deriving the conditions (timing constraints) needed to avoid glitching in proposed circuit. These results are used to propose three different driving circuits for the delay control-bits of proposed DCDL. Section V presents the obtained simulation results for a 90-nm CMOS technology. The results presented

in this section, in addition to verify the correctness of the analysis of Section IV, demonstrate the performances of proposed DCDLs in comparison to previously proposed structures. Finally, Section VI describes the employ of proposed DCDL to implement an All-digital SSCG designed with the approach described in [18].

II. PREVIOUSLY PROPOSED NAND-BASED DCDL AND GLITCHING

Fig. 1(a) shows the NAND-based DCDL of [12]–[15]. The circuit is composed by a series of equal delay-elements (DE), each composed by four NAND gates. In the figure “A” denotes the fast input of each NAND gate. Gates marked with “D” are dummy cells added for load balancing. The delay of the circuit is controlled through control-bits S_i , which encode the delay control-code c with a thermometer code: $S_i = 0$ for $i < c$ and $S_i = 1$ for $i \geq c$. By using this encoding, each DE in Fig. 1(a) can be either in pass-state ($S_i = 0$) or in turn-state ($S_i = 1$). In Fig. 1(a) all NAND gates present the same load (two NAND gates) and, therefore, in a first order approximation, present the same delay. This consideration allows to write the delay δ , from In to Out , as follows:

$$\delta = 2t_{\text{NAND}} + 2t_{\text{NAND}} \cdot c \quad (1)$$

where $t_{\text{NAND}} = (t_{\text{NAND LH}} + t_{\text{NAND HL}})/2$ while $t_{\text{NAND LH}}$ and $t_{\text{NAND HL}}$ represent the delay of each NAND gate for a low-to-high and high-to-low output commutation, respectively. It is interesting to observe that (1) holds both for low-to-high and high-to-low Out commutations. Equations (1) suggests that $t_{\min} = 2t_{\text{NAND}}$ and $t_R = 2t_{\text{NAND}}$.

In DCDL applications, to avoid DCDL output glitching, the switching of delay control-bits is synchronized with the switching of In input signal. Glitching is avoided if the control-bits arrival time is lower than the arrival time of the input signal of the first DE which switches from or to the turn-state. Unfortunately in the DCDL of Fig. 1(a) this condition is not sufficient to avoid glitching. In this circuit, in fact, it is possible to have output glitches also considering only the control-bits switching, with a stable input signal. Some examples of glitching problems of this DCDL are highlighting in Fig. 1. Let us name $\mathbf{S} = [S_0, S_1, \dots]$ the vector of the control-bits of the DCDL. In Fig. 1(a) it is assumed that $In = 0$ and that the control-code c of the DCDL is switched from 1 ($\mathbf{S} = [0, 1, 1, 1, \dots]$) to 2 ($\mathbf{S} = [0, 0, 1, 1, \dots]$). Please note that, within the structure, the switching of S_1 and \bar{S}_1 results in two different paths that generate an output glitch. It can be

متن کامل مقاله

دریافت فوری ←

ISIArticles

مرجع مقالات تخصصی ایران

- ✓ امکان دانلود نسخه تمام متن مقالات انگلیسی
- ✓ امکان دانلود نسخه ترجمه شده مقالات
- ✓ پذیرش سفارش ترجمه تخصصی
- ✓ امکان جستجو در آرشیو جامعی از صدها موضوع و هزاران مقاله
- ✓ امکان دانلود رایگان ۲ صفحه اول هر مقاله
- ✓ امکان پرداخت اینترنتی با کلیه کارت های عضو شتاب
- ✓ دانلود فوری مقاله پس از پرداخت آنلاین
- ✓ پشتیبانی کامل خرید با بهره مندی از سیستم هوشمند رهگیری سفارشات