Title	Exponential extended flash time-to-digital converter
Author(s)	Chen, Peng; Staszewski, Robert Bogdan
Publication date	2016-06-15
Publication information	Second International Conference on Event-based Control, Communication, and Signal Processing (EBCCSP)
Conference details	IEEE International Nordic-Mediterranean Workshop on Time-to-Digital Converters and Applications (NoMe - TDC 2016), Krakow, Poland, June, 2016
Publisher	IEEE
Item record/more information	http://hdl.handle.net/10197/9038
Publisher's statement	© 2016 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works
Publisher's version (DOI)	http://dx.doi.org/10.1109/EBCCSP.2016.7605281

Exponential Extended Flash Time-to-Digital Converter

Peng Chen and Robert Bogdan Staszewski
University College Dublin, Dublin, Ireland
Email: peng.chen.1@ucdconnect.ie

Abstract—The digital-to-time converter (DTC)-based all-digital phase locked loop (ADPLL) attracts more and more attention due to its ultra-lower power consumption characteristic [1]. With DTC, the time-to-digital converter's (TDC) requirements are relaxed, not only for its range but also for its nonlinearity. However, the shortened TDC range, which is less than one digital controlled oscillator (DCO) output period in the new architecture makes the settling time longer and the TDC gain calibration difficult. This work introduces a technique to extend the TDC range by 16 times to accelerate the settling process, while the extended part can be disabled when ADPLL is in lock. Furthermore, the TDC gain calibration is easier.

Keywords—TDC; two stages; DTC-based ADPLL; exponential; DLL;

I. INTRODUCTION

Plenty of applications rely on the low-power features of the emerging internet-of-things (IoT). In the digital-to-time converter (DTC)-based all-digital PLL (ADPLL) architecture [1, 2], a combined arrangement of a DTC with a shorter time-to-digital converter (TDC) replaces the traditional full-length TDC in the phase detection part [3]. This low-power ADPLL architecture benefits from a phase prediction [1, 2]. The introduced DTC covers the whole range of one DCO output (denoted as 'CKV') period. It aligns the rising edges of the reference clock and CKV. Consequently, the TDC has only to measure the thermal noise and nonlinearity from other analog/RF blocks and requires much fewer stages, thus consuming less power and area. The stringent linearity requirement of TDC is thus shifted to DTC where it is much easier to handle.

The shortened-range TDC gives the overall advantage of lower power and less area. As a trade-off, the settling process takes longer. The shorter TDC range cannot produce an accurate phase error when that range is exceeded. Secondly, it also becomes difficult to estimate the unit delay of TDC as in the traditional ADPLL architecture [3]. In [1], the TDC gain is related to the DTC gain, which has to be taken care of in the layout in which the accuracy is hard to guarantee. The motivation of this work is to improve the performance of the above two points while not significantly affecting other aspects.

Based on RTL level simulations of the DTC-based ADPLL [1], the following example and analysis are given. At the start of locking, with the TDC's range being only one quarter of one CKV period, the TDC output will get stuck at the boundary for a relatively long time, as the top plot in Figure 1 shows. From 7 us to 13 us, the TDC output keeps the positive maximum value. In this example, one CKV period is 800 ps,

corresponding to the 1.25 GHz CKV frequency; TDC has 16 stages with a unit delay of 15 ps. TDC's output is coded from -8 to 7. Therefore, when the time difference of the two inputs is larger than 105 ps, the TDC positively overflows, generating the maximum value of 7. When the time difference is between 105 ps to 800 ps, the TDC can only generate 7. The phase error is then input to the loop filter, which contains an integrator. This under-response phenomenon in the tracking bank corresponds to a smaller bandwidth and makes the loop settling much slower.
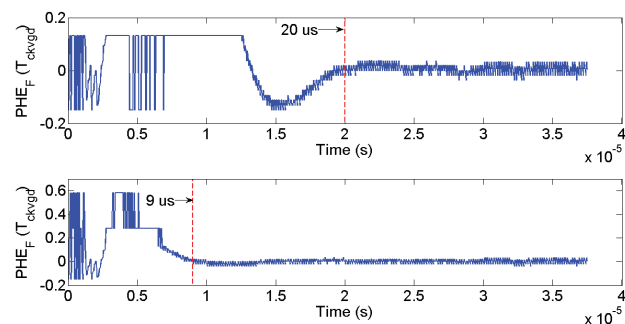

Figure 1: Simulated settling behaviors before (top) and after (bottom) using the exponentially extended TDC.

The fractional part of the phase error (PHE_F) is adopted as a locking indicator. When the DCO is locked, its output is a fixed frequency signal. Its input, namely OTW, fluctuates around a fixed value. Then the phase error detection part has to generate a value fluctuating around zero, which is due to the type II ADPLL characteristic. From the above analysis, the limited range of TDC contributes to a long settling time. What if the TDC range is exponentially extended just to cover the full CKV period? By only updating the TDC part of an existing ADPLL, the RTL code is re-run to produce the lower sub-figure in Figure 1. It is shown that after replacing the narrow-range TDC with the extended TDC, the settling time is reduced from 20 us to 9 us. The phase error roughly ranges from -0.15 to 0.6, larger than the one without the extended TDC, which is limited from -0.15 to 0.15. A longer TDC range can feed back the phase error more accurately, allowing the loop to settle earlier. When the ADPLL is locked, only a few stages of the original flash TDC are used. Thus, it is reasonable to propose an extended flash TDC which consists of two stages. The first stage is the traditional flash TDC covering a narrow range. The second stage is exponentially extended, accelerating the settling process. Moreover, the second stage can be automatically disabled in the locking bank to save power.

This work is organized as follows. Section II describes the extended flash TDC architecture and its schematic. In Section

III, the DLL based calibration scheme is introduced, followed by the simulation results. At last, the conclusion is drawn.

II. ARCHITECTURE

The proposed TDC contains two parts. The first part is the normal flash TDC [4], but it is much shorter here. In the second part there are five delay units in parallel, loaded with exponentially increasing capacitors. The delay is determined by an RC model. As Figure 2 shows, the delayed rising edge is determined by the controlling capacitance. The variable-slope charging method is chosen to save power and complexity, even though the linearity is not as good as in [5]. As explained above, the extended part is mainly used for fast locking where the linearity is not that critical. What's more, five delay lines in parallel are chosen to speed up the phase detection. Even though the sigma-delta TDC [6] can also cover a long range and achieve a fine resolution, it takes many cycles to get a stable averaged value.
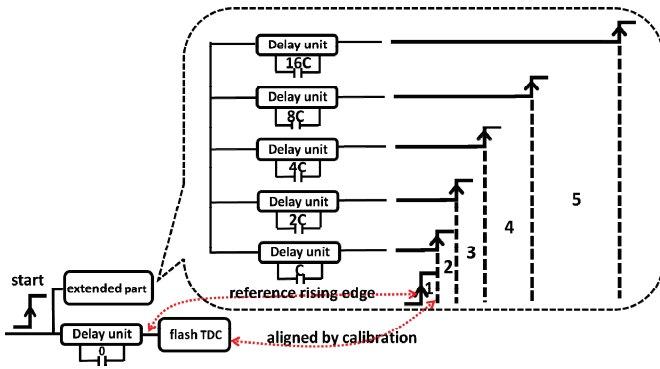

Figure 2: The proposed TDC architecture.

In Figure 2, the start signal is first delayed by one delay unit without the loading capacitor, generating a reference rising edge. By loading one unit capacitor "C", the rising edge is just after the reference rising edge. The time difference between these two rising edges is marked as the "1" region. The left four regions are obtained in the similar way. Time difference in Region "1" is denoted as T_u. In Region "2", the time diffrence is the same because "2C" is loaded. Thus, the time difference for region 2 is also T_u. Time differences for Regions 3, 4, 5 are $2T\_u$, $4T\_u$, $8T\_u$ respectively.

The reference rising edge is also used as the input to the flash TDC. Moreover, the flash TDC output is aligned with the rising edge of the output from the delay unit loaded with "C". That alignment is done by calibration, which will be covered in Section IV. The rising edge of the output from the delay unit loaded with "16C" will also be calibrated, to gurantee the time difference between it and the reference rising edge equals one CKV period, or one pre-known period. In this way, the more or less accurate value of the flash TDC resolution can be calculated, making its calibration in DTC-based ADPLL practical. As a comparison, in [1] it is hard to know the unit delay of TDC directly.
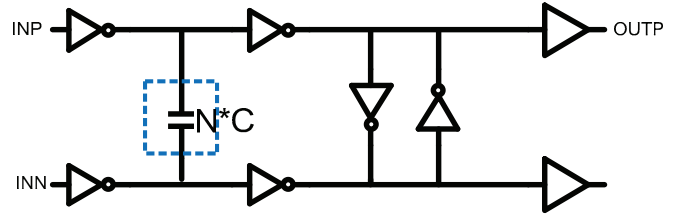

Figure 3: Delay unit schematic.

The delay unit for the extended TDC is shown in Figure 3. The "START" signal is first fed into a single-input-differential-output block, generating INP and INN. The loading capacitor is placed at the first inverter's output. After the second inverter, a weak latch is used to regenerate the complementary signal, reducing the 'differential' mismatch. One more buffer is placed after the latch to isolate its loading capacitance. Inside the delay unit, except for the parasitic capacitances related to the first four inverters, the loading capacitor, which is cycled by the blue line, dominates the capacitance. The resistance is determined by first two inverters' sizes, which are the same for the five delay lines. As a result, the RC-modeled delay almost linearly increases as the loading capacitance increases.

A. Flash TDC

The traditional architecture is adopted for the flash TDC. The unit delay is determined by the inverter's delay. It has 8 stages here, coded from -4 to 3. When the inverter from a standard cell is loaded with another inverter and a flip-flop, the schematic-level simulations show the unit delay is around 6 ps. For easier explanation, in the following, the flash TDC resolution specification is defined as 6 ps. With extra two dummies placed at the input and output, the whole flash chain has a 60 ps delay. The unit delay can be adjusted by controlling the inverters' supply voltage. That adjustment is accompanied by the flash TDC calibration scheme in Section III.
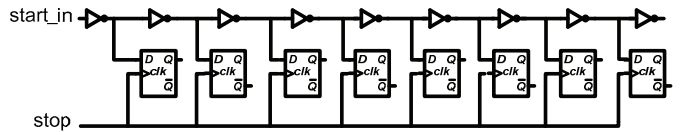

Figure 4: Flash TDC core schematic.

B. Extended TDC part

The extended part consists of five delay lines in parallel. The delay is determined by the R*C time constant as explained above. The resistances (R) are the same, while the capacitances are C=(C0+N*C_unit). N=1, 2, 4, 8, 16 for the five lines. C0 is very small compared with the loading capacitance N*C and thus can be ignored. However, due to this parasitic capacitance, the time difference of region "5" in Figure 2 is smaller than $8T\_u$. To compensate it in the layout, we can adjust the loading capacitors according to the post-layout simulations, in order to avoid employing the look up table to calibrate the INL [7]. That rough compensation can be done based on this special application. C_unit is chosen to be 5 fF in the following schematic simulations.

Referring to the example in Introduction, the extended TDC interpolates now the values between 105 ps to 800ps. The smaller the time difference, the relatively higher value it produces, which is due to the exponential characteristic. It is expected that the transfer function does not have equal spacing between every two points. The extended TDC can be disabled when the flash TDC output is not all "1" at the next operating cycle, relaxing the tradeoff between the settling time and power consumption.

III. DLL BASDED CALIBRATION

The nonlinearity of the extended TDC is less important, while the range should be rather precise. In CMOS technology, the capacitors ratio can be accurately matched. Thus, after calibrating the output of the delay unit loaded with the largest capacitance, the delay units loaded with other capacitances are adjusted as well. After the calibration, delay unit loaded with "16C" has a delay equal to one CKV period, which is known a priori. That is $16T\_u=T\_ckv$. Then T_u mentioned in the above section is obtained. It should be noticed that T_u also equals the delay from the whole flash TDC. Then the flash TDC's resolution can be calculated. In other words, there will be two calibration schemes. One is to make the whole TDC range equal to one CKV period. Another is the flash TDC's range calibration, by making the flash TDC's output aligned with the rising edge of the output from the delay unit loaded with C_unit. Then, the flash TDC's whole delay equals T_u.

First, the whole TDC's range calibration is discussed. To calibrate the whole range, one way is to add a controllable resistor inside the unit delay, as figure 5 shows.
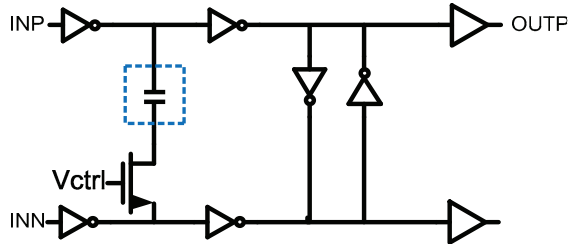

Figure 5: Controllable delay unit schematic.

One voltage-controlled NMOS is added below the capacitor. It can be modeled as a voltage-controlled resistor. Thus the RC delay is controlled by adjusting the R. For the five lines, the delay units are identical, so that R keeps the same. The time difference between each two lines is still determined by the loading capacitors ratio. When we sweep the control voltage (Vctrl) in Figure 5, a relationship between the delay and its control voltage is revealed in Figure 6.
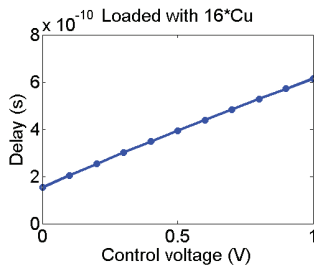

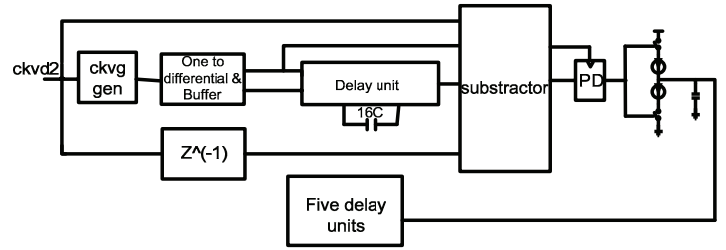Figure 6: Delay versus voltage for the delay unit with "16C" capacitance

Figure 7: Whole TDC range calibration.

In Figure 7, the whole TDC range calibration scheme is drawn. The input signal is CKVD2 (it is from CKV divided by 2 [1]). The period is around 800 ps. In [1], CKV2 is first clock-gated. The output operates at reference frequency, before fed into singe-input-differential-output and buffer block. Then it passes the delay unit loaded with "16C" and goes into the time registered subtractor [8]. CKVD2 and its next rising edge are also fed into the subtractor. The subtractor outputs two signals, whose time difference is T_CKVD2-16T_u. With a phase detection (PD) block and charge pump, the information is translated from time domain to voltage domain. The voltage of the charge pump's loading capacitor controls the resistor of the delay unit in the extended TDC. The extended TDC's delay can be adjusted to make the output of the subtractor having zero time difference. When the calibration loop is stable, the delay unit loaded with "16" has a delay of one T_CKVD2.

The simulation is done with schematic plus Verilog-A models. The subtractor, phase detection (PD) part and charge pump are modeled as Verilog-A. Other parts are built at the schematic level. The simulation results are shown in Figure 8. To observe the calibration behavior, the delay unit's control voltage is first set to 0.5 V. When the loop is stable, it stays at around 0.8 V. It takes around 1 us to finish the calibration. After the calibration is done, the calibration loop can be disabled and the results can be stored in RAM or registers.
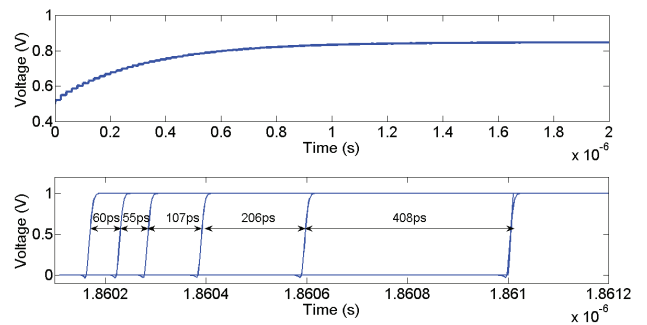

Figure 8: Simulation results of the whole TDC range calibration.

In Figure 8, the delays between each line of the extended TDC are shown in the second sub-figure. The small nonlinearity is caused by the parasitic capacitances. The added voltage-controlled MOS contributes part of the parasitic capacitance. Hence, its size should not be too large.

Secondly, the flash TDC's range calibration is introduced in the following. After the first calibration, the flash TDC's range is fixed. It is 60 ps, as shown in Figure 8. Then, the supply voltage of the 10 flash inverters is controlled to make them generate a delay of 60 ps. This is achieved by the DLL-based (delay lock loop) calibration scheme in Figure 9.
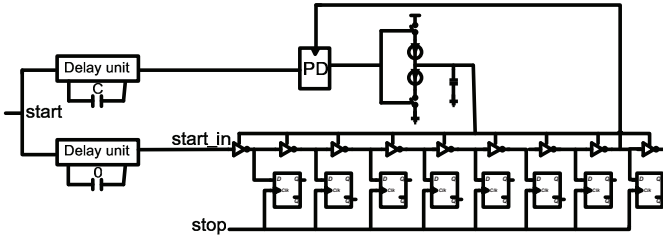

Figure 9: Flash TDC range calibration.

The "start" signal is fed into two delay units. One unit is loaded with "C"; another is loaded without a capacitor. The "start_in" is the input of the flash TDC. Flash TDC's output is fed into PD, along with the output of the delay unit loaded with "C". The loop pushes those two rising edges as close as possible, ideally zero.

Eight inverter stages are chosen here. Along with two extra dummies, the flash TDC has 10 stages in total. It should have around 6 ps resolution after the calibration is carried out. For the inverter delay, it is inversely proportionally to its supply voltage. In the schematic simulation, when the supply voltage is 1 V, the unit delay of the flash TDC is just around 6 ps. That means, after the calibration, the supply voltage for the flash TDC is around 1V, which is shown in the top plot of Figure 10.
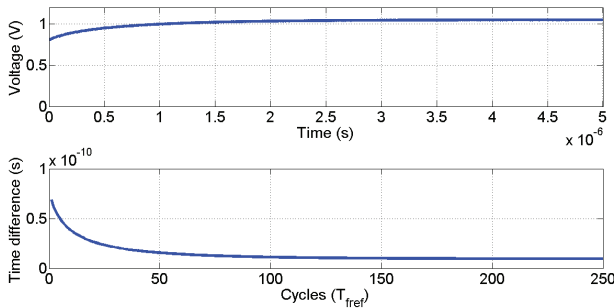

Figure 10: Simulation results of the flash TDC resolution calibration.

The time difference shown in the bottom plot approaches 10 ps. That is because the inverters' supply voltage is interfered by other signals. Better isolation or other method to tune the inverter delay could be chosen to overcome this issue. That calibration loop can also be disabled once the calibration is done. And there's no need to enable it again at all to keep the flash TDC resolution fixed.

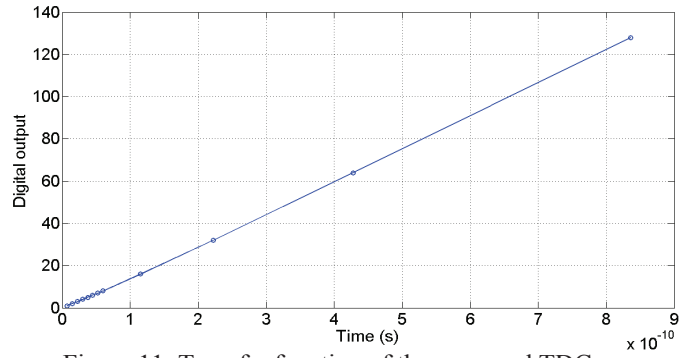After the two calibrations above, the transfer function of the proposed TDC is drawn in Figure 11.

Figure 11: Transfer function of the proposed TDC.

IV. CONCLUSION

An exponentially extended flash TDC is proposed in this work, which is to be used in a DTC-based ADPLL. Analysis, architecture, schematic and simulation results are given. Further, two DLL-scheme based calibration loops are introduced. The extended part and the calibration loops are disabled during the tracking bank. This proposed TDC can speed up the settling time and calibrate the TDC gain without extra power when the PLL is in lock.

REFERENCES.

[1] V. K. Chillara, et al. "An 860μW 2.1-to-2.7 GHz all-digital PLL-based frequency modulator with a DTC-assisted snapshot TDC for WPAN (Bluetooth Smart and ZigBee) applications," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 2014.

[2] JC. Zhuang, and R. B. Staszewski. "A low-power all-digital PLL architecture based on phase prediction," in *IEEE Int. Electronics, Circuits and Systems Conf. (ICECS)*, 2012.

[3] R. B. Staszewski, and P. T. Balsara. "Phase-domain all-digital phase-locked loop," *IEEE Trans. Circuits Syst. II: Express Briefs,* vol. 52, no. 3, pp 159-163, Mar. 2005.

[4] R. B. Staszewski, et al. "1.3 V 20 ps time-to-digital converter for frequency synthesis in 90-nm CMOS." *IEEE Trans. Circuits Syst.*, vol 53, no. 3, pp 220-224, Mar. 2006.

[5] ZY Ru, et al. "A high-linearity digital-to-time converter technique: constant-slope charging," *IEEE J. Solid-State Circuits*, vol. 50, no. 6, pp 1412-1423, June, 2015.

[6] P. Chen, et al. "Design and built-in characterization of digital-to-time converters for ultra-low power ADPLLs," in *Proc. Eur. Solid-State Circuits Conf. (ESSCIRC)*, 2015.

[7] K. Raczkowski , et al. "A 9.2–12.7 GHz wideband fractional-N subsampling PLL in 28 nm CMOS with 280 fs RMS jitter," *IEEE J. Solid-State Circuits*, vol. 50, no. 5, pp 1203-1213, May, 2015.

[8] Y. Wu, et al. "A 103fs rms 1.32 mW 50MS/s 1.25 MHz bandwidth two-step flash-ΔΣ time-to-digital converter for ADPLL," in *IEEE RFIC Symp. Dig.*, 2015.