

# A DLL-based Poly-phase Sampler for a Sigma-Delta ADC

S.Ouzounov, E.Roza\*, H. Hegt, G.van der Weide\* and A. van Roermund  
Eindhoven University of Technology,                      \*Philips Research Laboratories  
Den Dolech 2, EH 5.15                                              Prof. Holstlaan 4  
5600MB, Eindhoven,                                              5656AA, Eindhoven  
+31 40247 3242, +31 40245 5674,  
E-mail: [S.Ouzounov@tue.nl](mailto:S.Ouzounov@tue.nl)

*Abstract*—Due to their simple and cost-effective implementation Sigma-Delta ADCs are preferred converters in many low frequency applications. However, despite the high demand, there are still only a few designs that achieve high performance for bandwidths in the MHz range and above. A main difficulty in utilizing Sigma-Delta converters for higher frequency bandwidths comes from the basics of their operation, namely from the need for a high oversampling ratio. This high oversampling ratio is usually obtained by a high frequency sampling operation and high quality decision switches. In Sigma-Delta converters, those switches operate on weak high-speed signals and can introduce performance degrading effects like asymmetry, metastability or excessive loop delays.

In this paper an alternative implementation of the sampling circuit as suggested in [1] is described. Instead of a single phase clock, the sampler generates N equally spaced clock phases that are synchronized to an external reference clock with the help of a Delay Lock Loop (DLL). This poly-phase sampling allows for a high overall oversampling ratio with a much lower effective sampling speed. In this way the sampling operation is parallelized and the requirements towards the speed and accuracy of the individual decision switches are significantly decreased.

In this paper the implementation of an eight-phase, 1GHz poly-phase sampler is shown. The circuit level design requirements for a single delay element, chain of delay elements and a complete DLL are discussed and the basic tradeoffs are described

*Keywords*— DLL, Sigma-Delta ADC, poly-phase sampling, phase noise, spurious tones

## I. INTRODUCTION

Timing jitter and spurious tones in the clock signals that drive sampling circuits in ADCs can limit converters' achievable dynamic range [2], [3]. The impact of those clock imperfections is becoming more and more critical with the increasing sampling speed and

the increasing resolution of the converters. Those factors impose careful consideration of the properties of the clocking circuits that are used in the conversion process. The problem is additionally aggravated by the inherent difficulty to propagate high speed clock signals to and within the ICs. An option that uses high frequency (1GHz in this case) external reference clock is least attractive because of severe requirements towards the PCB design and the accuracy of the implementation. It is subject to deterioration due to external interferences, noise and propagation delays. Moreover, in most applications, high speed, high accuracy clock source is generally unavailable. The solution is to generate the clock signals inside the IC via a Phase Locked Loop (PLL) or a DLL.

This paper summarizes the basic design rules for a clock synthesizer based on a DLL and applies them in the design of an eight-phase DLL sampling circuit with 125MHz external reference clock. The technology used is that of a 0.18 $\mu$ m standard digital CMOS process.

The paper is structured in the following way: In Section II, the system level design issues are discussed. The basic principle of operation of the DLL is explained and the sources of timing errors are pointed out. Their minimization at system level is described. The next section elaborates on the design of the individual building blocks. A certain choice of the transistor implementation of a basic delay element, phase detector, charge pump and loop filter is explained and the circuit parameterization is shown.

## II. SYSTEM LEVEL DESIGN

### A. DLL vs. PLL

For clock generation or clock-data recovery two mechanisms are largely used: PLL and DLL. Dependant on the application one approach may have advantages with respect to the other. In the context of a multi-phase

clock generation the following principle differences between the two systems should be pointed out:

1) The DLL is not generating its own clock. It is producing delayed and synchronized versions of the external reference clock. In contrast, in the PLL loop, the clock is internally synthesized via the Voltage Controlled Oscillator (VCO).

2) The two systems respond in a different way to the sources of phase errors in the loop. In the DLL the jitter accumulated by the end of the delay chain does not contribute to the next cycle, because at each cycle the loop is ‘reset’ by the reference. In the PLL, the correction for the jitter generated in the VCO is dependent on the loop operation [4]. For example, in a narrow band PLL (narrow band loop filter), a significant part of the VCO phase noise may enter the loop. Its correction usually requires several loop cycles [5]. In broadband PLLs this problem is alleviated with a penalty of a longer locking time and worse stability.

3) The generation of N synchronized clock phases is intrinsic for the DLL operation. In the PLL case, generation of multiple phases can be implemented by an N tap VCO design. However, this is in contradiction with the optimal high frequency VCO design (would require minimum number of stages) [13] and may lead to performance degradation of the total PLL loop.

4) The DLL is a first order system, while the PLL is at least a second order system and is much more prone to instability.

The DLL properties described above were the main arguments for choosing it for the implementation of a multi-phase sampling circuit. The rest of this work concentrates on system and circuit design issues viewed from the DLL perspective.

### B. DLL block diagram

The structure of a DLL based sampler is shown in fig.1. It consists of a voltage controlled delay line (VCDL) that produces N equally spaced phases ( $P_{1...N}$ ) of the input reference clock. The VCDL is built from N equal Delay Cells (DC). The phase difference between the input clock and its delayed version produced by the VCDL, (the last phase  $P_N$ ) is compared in a Phase Detector (PD). The Charge Pump (CP) converts this phase difference into an electrical signal by depositing or withdrawing charges from its output node. The purpose of the Loop Filter (LF) is to remove the high frequency component and average the CP output.

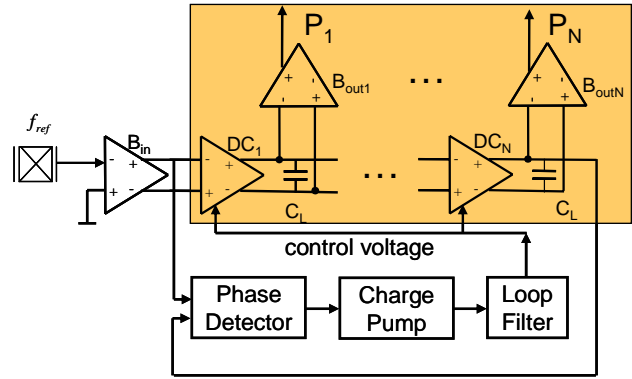


Figure 1, DLL Block Diagram

The filtered output is then used as a control voltage that speeds up or delays the VCDL. In order to use the DLL as a multi-phase sampling circuit the following additional blocks are added: Firstly, the quality of the generated signals depends on the quality of the reference clock. The best option is to use as a reference a monolithic crystal oscillator which has a very good jitter performance. The weak input signal is buffered ( $B_{in}$  Fig.1) and the supplied to the delay line and the phase detector. At each output of the VCDL a buffer/driver is added ( $B_{outN}$ ) that assures the correct driving levels for the sampling switches.

### C. Types and Sources of Errors in the DLL

Two types of errors associated with the DLL operation can be distinguished: 1) random errors that lead to appearance of phase noise in the output and 2) systematic errors that cause spurious tones in the output signal.

Phase noise can be defined as random fluctuations of the clock period (Fig. 2). In the time domain, it is usually referred to as timing jitter. In the frequency domain, phase noise is displayed as a spread of noise energy around the center frequency of the oscillator.

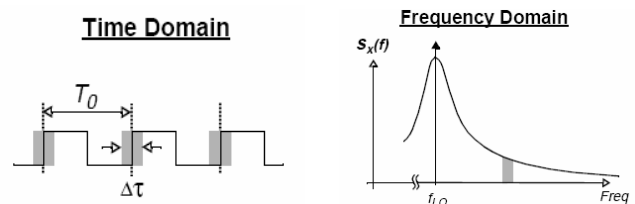


Figure 2, Timing jitter and phase noise frequency spectrum

The phase noise of an oscillator at a given frequency offset is found from the ratio of the power in a 1Hz bandwidth at the offset frequency, to the total power of the carrier (usually specified in dBc/Hz at a given offset).

Most often, the phase noise is a result of thermal and

1/f noise in the active and passive devices in the circuit. Understanding the mechanisms that transforms those noise sources into a phase error is important for optimal circuit design. The design techniques for the minimization of the phase noise are discussed in the following section in the context of the design of the building blocks.

On the other hand, spurious tones are systematic timing fluctuations in the oscillator waveform (Fig. 3). They appear as discrete frequency components in the output spectrum. The sources of spurious tones may vary in different architectures and transistor implementations. In the DLL, most important sources of systematic errors are: the mismatch in the delay cells (leads to different delay times per stage), and the asymmetry in the CP up and down currents. Due to the DLL structure all spurious that would appear are harmonics of the input crystal reference frequency. The mechanism of their generation is again discussed in more detail in the next section.

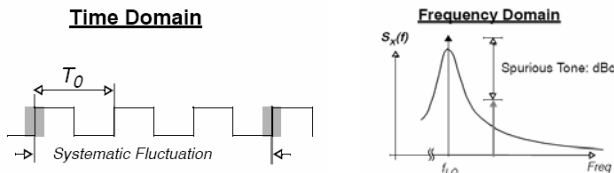


Figure 3, Spurious tones in time and frequency domain

### III. TRANSISTOR DESIGN OF THE BUILDING BLOCKS

#### A. Basic delay cell design

Two different approaches can be chosen in the design of the basic delay cell: a single ended inverter or a differential inverter cell. The advantages of the single-ended approach are the higher possible switching speed and simpler structure. However, a differential implementation would have higher immunity to common mode noise like supply variations and substrate noise coupling. Such an implementation provides also for a better control of the gain per stage and the voltage swing at each point of the delay chain. It also has a better potential for lower switching noise generation and a better matching. The chosen transistor implementation of the DC is shown in fig. 4. It is a fully balanced (differential) configuration and consists of a source coupled differential pair. The tail current source is split to allow for coarse and fine tuning of the current. The load is implemented as PMOS transistors operating in triode region.

The timing delay between the crossover point of the output transition of one stage and the crossover point of the output transition of the next stage can be

approximated as [6]:

$$t_{delay,i} = \frac{V_{SW,i} C_{L,i}}{I_{S,i}} \quad (1)$$

where  $I_S$ ,  $V_{SW}$ ,  $C_L$  are respectively, the tail current, the voltage swing at the output and the output capacitance of each individual cell  $i$ .

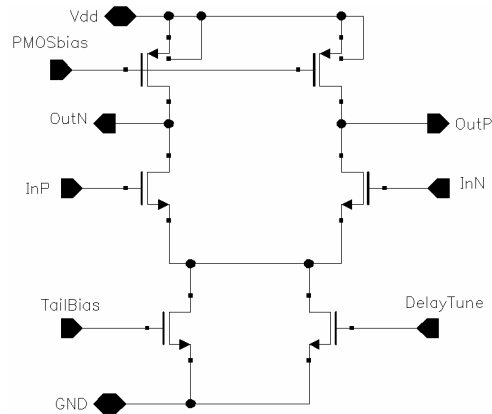


Figure 4, Differential delay cell: circuit schematic

Expression (1) can be used to determine the strategy for implementation and control of the delay of each cell. The following considerations apply:

1) The output capacitance is a combination of the parasitic capacitances seen at the output node of the delay cell. Those are: the output capacitance of the delay cell itself, the input capacitance of the next delay cell, the input capacitance of the buffer for the sampling switches and the interconnection capacitances. In order to minimize the impact of those mainly parasitic capacitances, a nominal  $C_{L,nom}$  should be added with a value well above the parasitics. That is limiting the designers possibilities to choose the  $C_{L,nom}$  and makes it impractical to use for a control of the delay time.

2) The output voltage swing  $V_{SW}$  has to be chosen with several considerations in mind:

a) The small signal gain of each delay stage can be expressed as proportional to the output swing:

$$A = g_m R_L = \frac{2(I_s/2)}{V_{GS(N)} - V_{TN}} \frac{V_{SW}}{I_S} = \frac{V_{SW}}{V_{GS(N)} - V_{TN}} \quad (2)$$

where  $R_L$  is the equivalent resistance of the triode region PMOS and  $g_m$  is the transconductance of the input stage. Equation 2 is setting an upper boundary for the voltage swing. In order to minimize the amplification of

the noise from stage to stage, the gain should be just high enough ( $1 < A < 2$ ) to assure the propagation of the signal through the delay chain. A higher gain would lead to amplification of the generated noise along the chain and consequently to accumulation of phase noise within one reference clock cycle – an effect that cannot be compensated.

b) The triode load has the role of a linear controlled resistance and is setting another requirement for  $V_{SW}$ . The triode mode of operation is assured as far as:

$$V_{DS,P} = V_{SW} < V_{GS,P} - V_{T,P} \quad (3)$$

The range for  $V_{GS,P}$  can be close to the full supply voltage, and therefore, the upper bound for  $V_{SW}$  is  $V_{DD} - V_{T,P}$ . In practice, the PMOS load devices should be kept well inside the triode region over the full voltage swing range to ensure a relatively-constant channel resistance, such that, the -3dB frequency of the delay stage would not vary greatly due to the PMOS load variations.

c) On the other hand,  $V_{SW}$  should be maximized in order to assure switching of the following stage and the sampling buffer and results in a higher relative SNR.

3) Referring back to (1), the easiest way to control the delay of each stage is via the tail current source  $I_S$ . However, for a proper locking, the control of each stage should be monotonic. Then a determined delay corresponds to a control voltage value. That can be achieved if the other terms in (1) would remain very close to their initial value for a variation of the tail current. That would require a circuit that keeps the voltage swing  $V_{SW}$  constant. This requirement can be implemented via a replica bias circuit [7], [9]. One possible implementation of this replica bias is described in the following subsection.

For the chosen design technology of 0.18 $\mu$ m CMOS process with  $V_{dd}=1.8$ , the optimal  $V_{SW}$  was calculated to be around 400mV. The required eight ( $N=8$ ) phases with combined frequency of 1GHz determine the choice of  $C_L$  and  $I_S$ . For a better matching and 1/f noise performance, relatively large devices ( $L \approx 10L_{min}$ ) were used for the input transistors and the PMOS load (Fig. 4). An evaluation showed that in such a case their parasitic capacitance would provide around 25% of the total load capacitance  $C_L$ . Next  $C_L$  was estimated as 1pF with 750fF assigned to the added capacitance  $C_{L,nom}$ . Using

(1) the required current consumption per delay stage was evaluated to be 640 $\mu$ A.

### B. Replica bias circuit

The transistor implementation of the replica bias is shown in fig. 5. The circuit adjusts the gate bias of the PMOS load devices for a fixed swing at the output. The transistors  $M_1$  and  $M_2$  from fig. 5 are exact replica (same sizes and preferably layout) of the PMOS load and the tail current source transistors from fig. 4. The voltage swing over  $M_1$  is compared to a reference voltage  $V_{REF}$ , where  $V_{REF} = V_{dd} - V_{SW}$ , via a differential to single-ended amplifier. The amplified difference is then used to adjust the voltage swing over  $M_1$ . Similar amplifier circuit is described in [10]. The source follower  $M_3$  is introduced to prevent the capacitive loading of the amplifier output from the delay line.

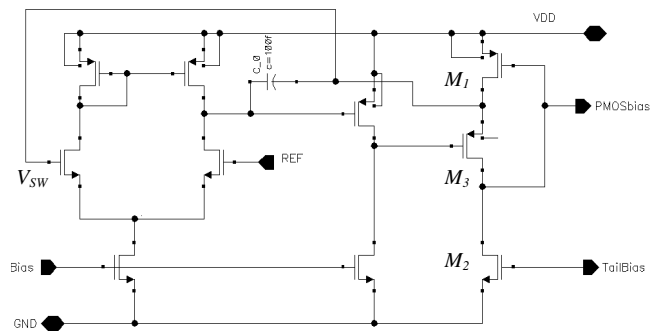


Figure 5, Replica bias circuit

Next, the design of the control path: phase detector, charge pump and loop filter is described.

### C. Phase detector

The phase detector (Fig. 6) senses the phase difference between input and output voltages of the delay chain and generates a pair of UP/DN signals to control the charge pump output current. The tri-state phase detector described in [11] was used. This phase detector generates two pulses whose pulse-width difference is ideally equal to the phase difference of the input signals. In order to achieve a better matching between UP/DN pulses, extra delays are inserted in the reset path to increase both pulses' widths.

The PD operates properly, if the difference between the input phases is within  $T_{REF}/2$ . For higher differences, the PD may try to track an incorrect edge and force the VCDL in the opposite direction (for example, instead of speeding up, it can additionally delay the chain). This situation can be avoided if the total

phase delay of the VCDL is design to be within  $\pm\pi/2$  of the reference period.

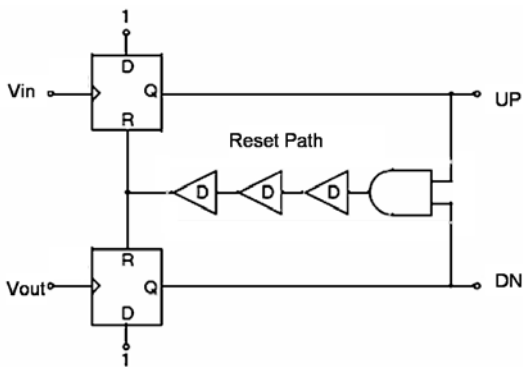


Figure 6, Phase-frequency detector implementation

D. Charge pump circuit

The charge pump creates a signal whose magnitude is proportional to the phase error. When the inputs to the phase detector are perfectly in-phase, the UP/DN signals are identical and the current at the output node of the CP is ideally zero. The main goal in the charge pump design is to minimize the dynamic mismatch between the UP and DN currents. The dynamic mismatch can be caused by the asymmetry in the rise and fall times of the UP/DN signals, and the difference in the electron mobility for NMOS/PMOS.

A circuit described in [12] is used for the CP implementation (Fig. 7). The main advantage of this circuit is that it assures small voltage variations over the current sources  $M_1$ ,  $M_2$ . An important consideration is the matching of the current mirrors.

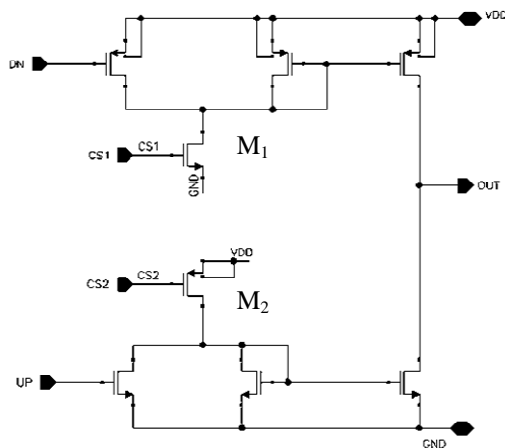


Figure 7, Charge pump circuit schematic

E. Loop filter implementation

The function of a loop filter is to produce a control voltage for the current sources in the delay stages. It also has to remove the high frequency components due to

switching in the charge pump output. The simplest implementation of the LF is just a capacitor. However, an active implementation provides a virtual ground at the CP output node and keeps it constant during transients. The circuit schematic of the LF is shown in fig. 8. The LF is built with operational amplifier with capacitive feedback. The voltage drop across  $R_L$  determines the required current for the detected phase difference. This current is mirrored to the delay stage current sources via  $M_d$ .

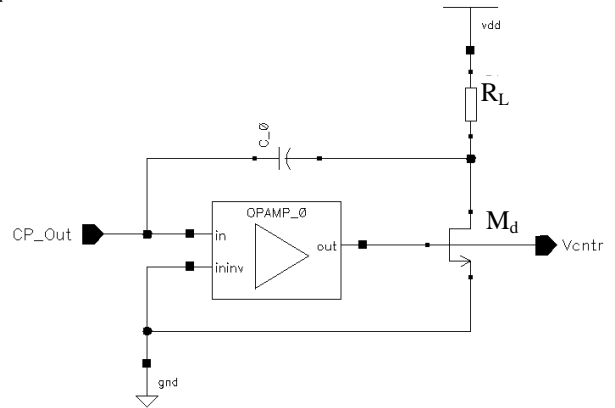


Figure 8, Active loop filter implementation

IV. CONCLUSIONS

The implementation of an 8-phase, 1GHz DLL frequency synthesizer in 0.18 $\mu\text{m}$ , 1.8V CMOS process was described. The design requirements of each DLL building block for low phase noise and low spurious tones generation were discussed. An implementation that uses a fully differential delay cell and a replica bias circuit to fix the output voltage swings in a VCDL was shown. The delay per stage is controlled via variation of the tail current sources. The control voltage is generated with a 3-state phase/frequency detector, charge pump and an active loop filter. The proper operation of each building block was confirmed via transistor level simulations.

REFERENCES

- [1] E. Roza, Poly-Phase Sigma-Delta Modulation, *IEEE Transaction on Circuit and Systems-II: Analog and Digital Signal Processing*, vol. 44, No. 11, 1997.
- [2] N. Da Dalt, M. Harteneck, C. Sandner, A. Wiesbauer, On the jitter requirements of the sampling clock for analog-to-digital converters, *IEEE Transaction on Circuits and Systems I: Fundamental Theory and Applications*, Volume: 49, Issue: 9, Sept.2002 Pages:1354 – 1360.
- [3] J.A. Cherry, W.M. Snelgrove, Clock jitter and quantizer metastability in continuous-time delta-sigma modulators, *Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, IEEE, Volume: 46, Issue: 6, June 1999 Pages:661 – 676.

- [4] M.J.E.Lee, W.J. Dally, T. Greer, N. Hiok-Tiaq; R. Farjad-Rad, J. Poulton, R. Senthinathan, Jitter transfer characteristics of delay-locked loops - theories and design techniques; *IEEE Journal of Solid-State Circuits*, Volume: 38, Issue: 4, April 2003 Pages:614 – 621.
- [5] B. Kim; T.C. Weigandt, P.R. Gray, PLL/DLL system noise analysis for low jitter clock synthesizer design, ISCAS 1994.
- [6] T. Weigandt, “Low-Phase-Noise, Low-Timing-Jitter Design Techniques for Delay Cell Based VCOs and Frequency Synthesizers“ *Ph.D. Thesis, MIT*, 1991.
- [7] J.G. Maneatis, Low-jitter process-independent DLL and PLL based on self-biased techniques; *IEEE Journal of Solid-State Circuits*, Volume: 31, Issue: 11, Nov. 1996, Pages:1723 – 1732
- [8] Ya-Lan Tsao; Ming-Chao Chung; Shyh-Jye Jou, Delay-difference DLL and its-application on skewed output buffer; *IEEE Asia-Pacific Conference on ASIC*, 2002 Pages:279 – 282.
- [9] B. Kim, D. Helman, P. R. Gray, “A 30-MHz Hybrid Analog/Digital Clock Recovery Circuit in 2-mm CMOS,” *IEEE J. Solid-State Circuits*, vol. 25, no. 6, pp. 1385-1394, Dec. 1990.
- [10] P. R. Gray, R. G. Meyer, Analysis and Design of Analog Integrated Circuits, *Wiley and Sons*, 1993.
- [11] D. H. Wolaver, Phase-Locked Loop Circuit Design, *Prentice Hall*, 1991.
- [12] A. Waizman, A delay line loop for frequency synthesis of de-skewed clock, *IEEE International Solid-State Circuits Conference*, 1994.
- [13] J.D.v.d.Tang, High-frequency oscillator design for integrated transceivers, *Ph.D. Thesis TU/e*, ISBN 90-386-1970-7, December 2002.