AN EMBEDDED CORE FOR SUB-PICOSECOND TIMING MEASUREMENTS

Sassan Tabatabaei
Vector 12 Corp.
stabatabaei@vector12.com

André Ivanov
The University of British Columbia
ivanov@ece.ubc.ca

Abstract

The continued market demand for GHz processors and high-capacity communication systems results in an increasing number of low-cost high volume ICs with multi-GHz clocks and/or multi-Gb/s serial communication interfaces. For such devices, timing specifications, e.g., jitter and skew, in the range of few picoseconds (RMS and/or p-p) are common. We describe an embedded core that allows such measurements. The core is small, functionally non-intrusive, and easily scalable for testing multiple circuits and signals on the chip. To reach the required sub-picosecond accuracy, we present a novel measurement and data processing technique, based on noise scaling. *The core has a standard low-speed serial interface.*

1. Introduction

This paper focuses on testing timing specifications of multi-GHz clock rate ICs and multi-Gb/s communication ICs. The continued market demand for GHz processors and high-capacity communication systems results in an increasing number of low-cost high volume ICs clocked at GHz rates and beyond and/or equipped with multi-Gb/s serial interfaces. For example, the production of SONET OC-192 (10 Gb/s) CMOS ICs has recently been announced [1][2]. Circuits to generate and distribute the timing signals for multi-GHz regimes also currently capture much attention [3][4][5]. The circuits achieving such clock and/or data rates are characterized by very stringent timing specifications, often dictated by governing standards (e.g., SDH, SONET, ATM) in the case of communication ICs.

Jitter and skew are types of important timing specifications that are specified in many serial communication standards. For GHz and gigabit communication ICs, jitter specifications in the range of few picoseconds (ps) RMS and/or peak-to-peak are common. Such timing specifications are not only extremely challenging to meet in design but are equally difficult to verify in characterization stages and to guarantee in volume production stages of product development [9]. Typically, a sophisticated and expensive external instrument-based test setup is used to verify such timing specifications. However, such setup is often difficult to integrate within production test environment and require large test time. Alternatively, some chips exploit certain on-chip functional testing capability, e.g., loop-back techniques [6]. Unfortunately such techniques tend to be inadequate for direct measurement and test of certain key device performance parameters.

In [7], we reviewed major techniques for timing measurement, introduced the concept of *embedded timing analysis (ETA)* for SoCs, and described a circuit core able to perform key timing specification measurements. We also described the general architecture of an *embedded timing processor (ETP*) that allows for the measurement of critical SoC timing signals. A key circuit of this core is an *embedded time to digital converter* (ETDC). In this paper, we introduce a new technique developed to reach sub-ps RMS jitter measurement accuracy in less time as compared to the technique in [7]. We illustrate the core's use for jitter specifications testing and present simulation results.

1.1. Overview of Techniques for Testing Timing Specifications

Techniques proposed for measuring timing specifications in digital applications, e.g., frequency, jitter, skew, delay, and timing variations, can be grouped into two main categories:

Signal sampling and frequency domain techniques

These techniques are based on sampling the signal with high resolution and performing time-domain or frequency domain analysis to extract phase information. Many sampling oscilloscopes perform timing measurements by estimating the signal threshold crossings from the collected samples. The well-known eye-diagram [11] method falls in this

0-7803-7542-4/02 $17.00

category. Frequency domain analysis also provides information about phase or frequency modulation components of a signal, which relate to the signal timing specifications [12]. Indirect sampling methods have also been used to estimate the jitter using ADCs [15]. Such techniques are difficult to use as the basis for embedded timing measurements because of the need for on-chip high-resolution high-speed ADCs.

Time domain techniques

These techniques use only a select set of signal threshold crossings to estimate timing specifications. This feature makes them more suitable for on-chip implementation, and particularly attractive for low-voltage process technologies, i.e., those typically used for high-speed devices. Examples of time domain techniques include frequency meters and counters, phase detectors, time interval analyzers (TIAs), time-to-digital converters (TDCs), jitter samplers, delay estimators, and logic samplers (for statistical jitter measurement) [8][13][14][16][17][18][23][24].

One class of such techniques is referred to as Vernier Oscillator. These techniques use the difference between the periods (or frequency) of two oscillators to quantize a time interval. A number of different implementations based on this technique have been disclosed that use two, three, or more oscillators [20—24]. A recent US patent [24] discloses a jitter testing technique based on measuring periods of a PLL output and using a histogram to estimate the peak-to-peak and RMS jitter. This technique is relatively easy to integrate on-chip if ring oscillators are used. However, the accumulation of period jitter in the free-running oscillators will limit the achievable accuracy. Placing the oscillators in close proximity can attenuate the effects of power supply and substrate noise-induced jitter. However, the remaining relative jitter between the oscillators will typically prevent achieving the accuracy required for measuring timing specifications associated with GHz ICs. In [7], we introduced a novel TIA architecture that achieves high-accuracy timing and jitter measurement through noise estimation and cancellation. The noise cancellation relies on a new technique that we refer to as *noise scaling*.

The remainder of this paper describes a new embedded time to digital converter (ETDC), which enables embedded time analysis with unprecedented accuracy in standard CMOS technology. This ETDC core is scalable to support multiple test points and its performance and functionality can be traded-off between off-chip and on-chip processing. Like the circuit in [7], this circuit employs two ring oscillators allowing for robust and programmable on-chip implementation. However, our main contribution lies primarily in having introduced special features to this basic technique to overcome the accuracy limitations arising from various noise sources while reducing the measurement time as compared to the technique in [7]. That is, we developed novel circuit features and measurement data processing to estimate and subsequently cancel the oscillator noise jitter based on a new noise scaling technique. We refer to this capability of exceeding noise barrier limitations as noise floor tunneling. The new EDTC achieves a high accuracy measurement capability in the presence of on-chip noise, while still using practical and easily realizable circuit techniques.

The remainder of this paper is organized as follows. Section 2 describes the core circuits and their operation. Section 3 focuses on the analysis of the noise that affects the accuracy of the ETDC. Then we illustrate how the noise floor limitations can be overcome (*noise floor tunneling*) using RMS jitter measurements as an example application. Section 4 presents simulation results when applying our ETDC to jitter testing. Finally, Section 5 concludes.

2. Embedded Time to digital Converter (ETDC) Core

Figure 1 illustrates the block diagram of our time to digital converter (ETDC) circuit core. The ETDC measures a time interval T_d:

$$T_d = t_{STOP} - t_{START} \quad (1)$$

where t_{START} and t_{STOP} are the time instances at which the rising edges of STOP and START signals occur, respectively. The **Double-Resolution Time Quantizer** (DTQ) block quantizes time with a resolution of $T_{\Delta f}$, which is set by the **Resolution Adjustment** (RA) block to a value less than a programmable threshold. The three counters connected to the DTQ hold three numbers upon completion of an interval measurement: fine, coarse and noise-floor numbers. These numbers contain information to estimate the input interval and also characterize the noise-floor on the ETDC. Since the DTQ maximum measurement range is limited, the **Range Extender** (RE) block is used to extend the capability of the DTQ for measuring longer time intervals. The **Calibration Controller** calibrates the DTQ using a reference clock to provide a precise estimate of $T_{\Delta f}$. The **TDC Controller** controls the communication and sequence of operation of the different blocks.

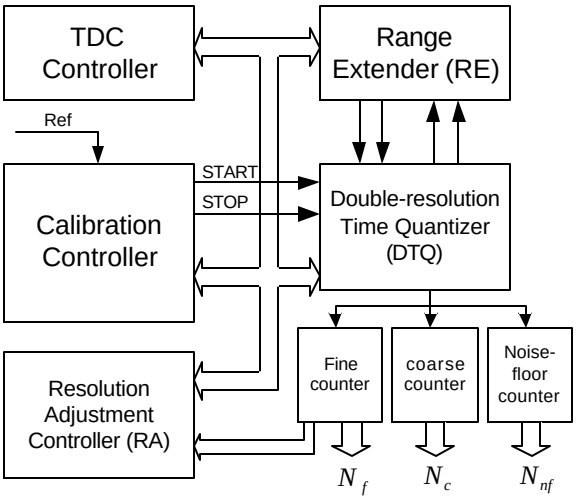

Figure 1: Block diagram of the ETDC circuit

2.1. Notation and Definitions

The notation and definitions used in the remainder of this paper follows. Note that any variable denoted by t refers to an instant in time, T refers to a time interval, and t refers to a time delay associated with a physical structure in the circuit, e.g., gates, routing, etc.

- t_{START} : time instance when the START signal is set HIGH
- t_{STOP} : time instance when the STOP signal is set HIGH
- $T_d = t_{STOP} - t_{START}$: time interval to be measured
- clkA: output signal of oscillator A (OscA)
- clkB: output signal of oscillator B (OscB)
- T_A (T_B): clkA (clkB) period
- $t_{X(i)}$: time instance when the i-th rising edge of clkX ($X = A$ or B) occurs.
- $T_{\Delta f}$: DTQ fine resolution.
- $T_{\Delta c}$: DTQ coarse resolution
- N_f : fine counter number
- N_c : coarse counter number
- N_{nf} : noise floor counter number
- M_A (M_B): output state of the k-bit counter CntrA(CntrB) in RE circuit

2.2. Double-Resolution Time Quantizer (DTQ)

As in [7], we use a DTQ instead of a single-resolution time quantizer to accelerate the measurements and also reduce noise effects. The block diagram of our new DTQ and associated waveforms appear in Figures 2(a) and (b).

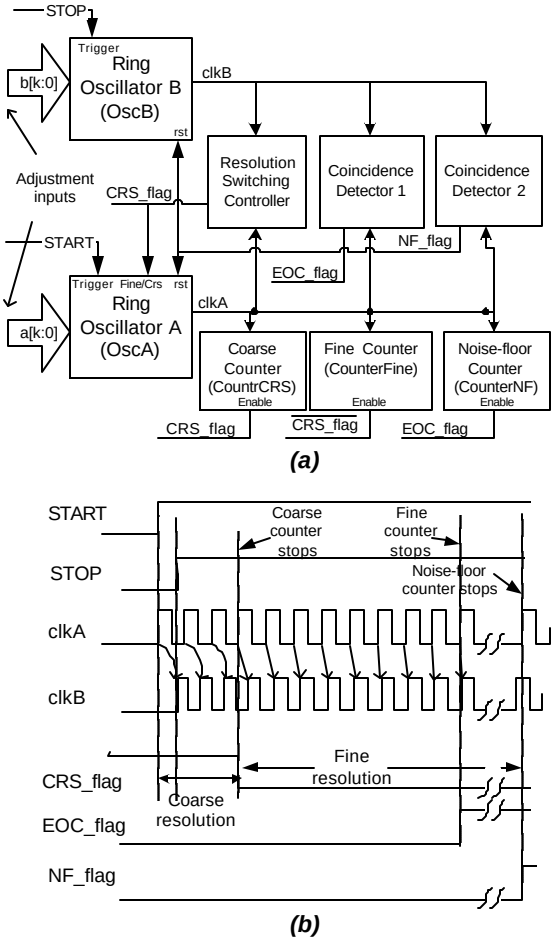

Figure 2: DTQ: (a) block diagram, (b) waveforms

The DTQ is comprised of two oscillators: OscA, triggered by a START edge; and OscB, triggered by a STOP edge. The oscillator periods are initially different by approximately 2% to 10% (typically, 40 - 200 ps). This corresponds to the coarse resolution. After both oscillators begin oscillating, their respective i-th edges start getting closer to each other, until they finally coincide for some value of i. Some time before the coincidence occurs, the period of one of the oscillator switches to fine resolution such that the difference between the oscillator periods is of the order of 0.1% to 0.5% of the OscA period (typically, 2 – 10 ps). OscA output is connected to three counters: fine counter that counts the clkA cycles from the start of the oscillation to the time when the oscillation period is switched, the coarse counter that counts the oscillation cycles from this time to the first coincidence time, and noise-floor

counter that counts clkA cycles from first coincidence time to the second coincidence time. The fine and coarse counters are sufficient for measuring the input interval, but the noise-floor counter is used to provide additional information required for estimating the ETDC noise floor, as will be shown in Section 3.1.

Figure 3 shows a particular DTQ implementation. Ring oscillator architecture is used for OscA and OscB, and Triple D flip-flops are used as phase detectors [7].

The delay element D1 in Figure 3 contributes a delay of t_{fine} between the i-th edges of clkA and clkB. Such delay ensures that CRS_flag is set LOW before the EOC detector is activated, switching the resolution from coarse to fine. The counters *CounterCRS*, *CounterFine*, and *CounterNF* hold the coarse (N_c), fine (N_f), and noise floor (N_{nf}) measurement numbers, respectively.

The delay D2 in clkB path is used to generate a second coincidence event, resulting in the accumulation of the number N_{nf} in the noise-floor counter.

At the end of each measurement sample, the TDC generates three numbers: coarse, fine, and noise-floor numbers, which are related to T_d as follow:

$$T_d = N_c T_{\Delta c} + N_f T_{\Delta f} - T_C - T_{Q1} + T_{R1} \qquad (2a)$$

$$T_d = N_c T_{\Delta c} + (N_f + N_{nf}) T_{\Delta f} - t_{D2} - T_C - T_{Q2} + T_{R2} \qquad (2b)$$

where T_C is a constant offset time, T_{Q1} and T_{Q2} are the quantization errors, and T_{R1} and T_{R2} are random error terms due to intrinsic jitter of the gates, flip-flops, and the oscillators. In a typical design $T_{\Delta c} \cong 10 T_{\Delta f}$ is selected. Since the coincidence detector flag is generated when $T_A - T_B = T_{\Delta f}$, the effective resolution in measuring T_d is $T_{\Delta f}$.

2.3. Measurement Time

To estimate the DTQ measurement time (T_{meas}), assume that the error terms in Eqn. 2b are negligible. Then $T_d + T_C + t_{D2} = N_c T_{\Delta c} + (N_f + N_{nf}) T_{\Delta f}$. Since it takes $N = N_c + N_f + N_{nf}$ cycles of clkA to perform a time interval measurement, the measurement time is:

$$T_{meas} = (N_c + N_f + N_{nf}) T_A = \left(\frac{T_d + T_C - t_{fine}}{T_{\Delta c}} + \frac{t_{fine} + t_{D2}}{T_{\Delta f}}\right) T_A \qquad (3)$$

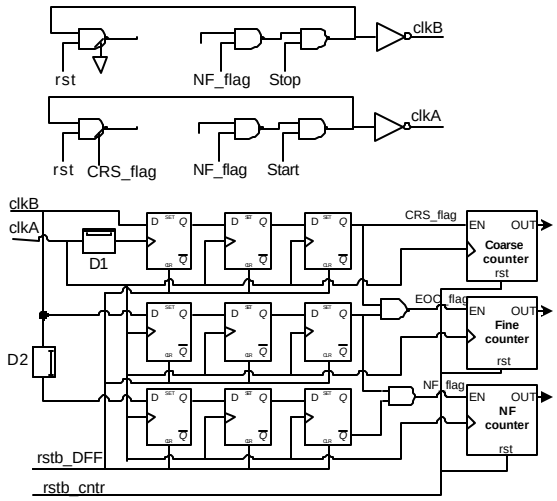

Figure 3: DTQ implementation

2.4. Measurement Range Extension

From the waveforms in Figure 2(b), if $T_d > T_A - D * T_B$, where D is the duty cycle of clkB, EOC_DFF will sample a HIGH at the second rising edge of clkA, erroneously signaling an end-of-conversion. Also, if $T_d < T_C$, the first rising edge of clkA will sample a HIGH regardless of the value of T_d. Therefore, the valid measurement range of T_d for this circuit is $-T_C < T_d < T_A - D * T_B$.

A Range Extender (RE) circuit, shown in Figure 4, extends this range. The RE generates a flag signal (*RE_Flag*) when $-T_C < t_{B(i)} - t_{A(i)} < T_A - D * T_B$. This flag masks the erroneous coincidence detector output flag. In Figure 4(a) the signal clkA is delayed by t_{re} to generate *clkA_re*. The two k-bit counters, CntrA and CntrB count the number of rising edges of *clkA_re* and clkB, respectively. t_{re} is selected such that the number in CntrA (i.e., M_A) remains larger than the number in CntrB (i.e., M_B) as long as $t_{B(i)} - t_{A(i)} > T_A - D * T_B$. The "Comp & Latch" block contains a comparator and a latch. When $t_{B(i)} - t_{A(i)} < T_A - D * T_B$, for a short duration $M_A = M_B$, causing the comparator to generate a pulse that becomes wider at the subsequent clkB rising edges. This pulse, when sufficiently wide, activates a latch to set the *RE_Flag*, allowing the valid EOC_flag to pass through. We used time-diversity sampling within the "Comp and Latch" block

to avoid the risk of erroneous RE flag due to glitches generated by asynchronous counter output switchings. More details regarding the RE's operation is found in [25].

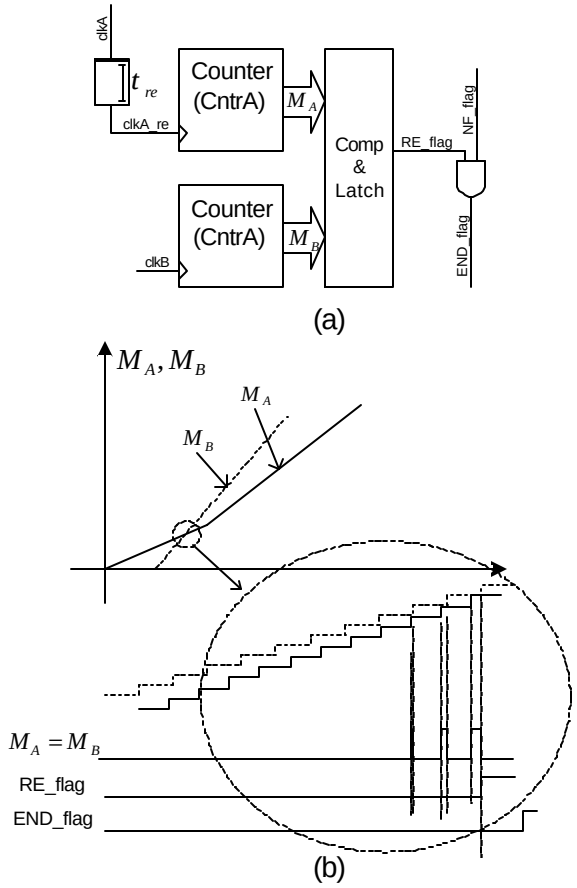

Figure 4: Range Extender (RE), (a) block diagram, (b) waveforms

2.5. DTQ Calibration

In practice, $T_{\Delta c}$ and $T_{\Delta f}$ will vary from one cycle of clkA and clkB to the next due to jitter in OscA and OscB. In Eqn. 2a, the oscillator noise effects are lumped into the term T_{R1}. Hence, $T_{\Delta c}$ and $T_{\Delta f}$ are in fact average values of fine and coarse resolution. For accurate calibration, we replace the delay D1 in the DTQ with a delay element whose delay can be switched between two different values, t_{fine1} and t_{fine2}. We use a reference clock (RefClk) with period T_{ref} to perform M_{cal} sets of calibration measurements. The i-th set includes the following three calibration measurements:

$T_{d(i1)} = T_{ref}$ and $t_{fine} = t_{fine2}$.

$T_{d(i2)} = 2T_{ref}$ and $t_{fine} = t_{fine1}$.

$T_{d(i3)} = 4T_{ref}$ and $t_{fine} = t_{fine1}$.

These measurements are repeated for $i = 1, ..., M_{cal}$ and each set is averaged over M_{cal} samples. This yields the three following equations:

$$\overline{N}_{c(1)}T_{\Delta(c)} + \overline{N}_{f(1)}T_{\Delta(f)} = \overline{T}_{d(1)} + T_C + \overline{T}_{Q(1)} + \overline{T}_{R(1)}$$

$$\overline{N}_{c(2)}T_{\Delta(c)} + \overline{N}_{f(2)}T_{\Delta(f)} = \overline{T}_{d(2)} + T_C + \overline{T}_{Q(2)} + \overline{T}_{R(2)}$$

$$\overline{N}_{c(3)}T_{\Delta(c)} + \overline{N}_{f(3)}T_{\Delta(f)} = \overline{T}_{d(3)} + T_C + \overline{T}_{Q(3)} + \overline{T}_{R(3)}$$

where $\overline{X}$ represents the average value of X. For sufficiently large M_{cal}, the noises due to T_R and T_Q terms, and also the jitter in the reference clock are reduced to negligible values due to noise averaging [25]. The selection of two different values of t_{fine1} and t_{fine2} for calibration ensures a non-singular characteristic determinant for the above system of equations, which is essential for reliable calibration. The resulting equations can be solved to compute $T_{\Delta c}$, $T_{\Delta f}$ and T_C. Assuming 50 ps RMS jitter in RefClk, and 30 ps jitter in the oscillators A and B, this scheme estimates $T_{\Delta c}$ and $T_{\Delta f}$ to within 1% accuracy with $M_{cal} = 5000$ samples [25].

2.6. Automatic Resolution Adjustment (RA)

The circuit in Figure 3(a) provides a high resolution, i.e., a small $T_{\Delta f}$, by generating a small difference between the loop-around delays of OscA and OscB. However, any mismatch between the gate delays and interconnect wiring in the OscA and OscB can cause a significant increase $T_{\Delta f}$, resulting in resolution degradation and increased quantization noise. The mismatch can also result in $T_A < T_B$, causing a measurement error.

To overcome the effects of mismatch, we use an automatic resolution adjustment technique. Using this technique, T_A and T_B are controlled digitally to ensure $0 < T_{\Delta f} < T_{th}$, where T_{th} is a user defined threshold for maximum resolution. This is achieved by replacing a number of the delay elements in OscA and OscB with controllable delay elements (CDE), which increase T_A or T_B when activated.

Assume that some CDEs are activated. The resolution adjustment circuit measures two known time intervals, e.g., T_{ref} and $2T_{ref}$, with a single resolution of $T_{\Delta f}$ to yield two numbers. The difference N_Δ between the two resulting counts is then determined. Assuming the measurement error is negligible, N_Δ and $T_{\Delta f}$ are related by $T_{ref} = N_\Delta T_{\Delta f}$. Since T_{ref} is constant, a larger N_Δ indicates a smaller $T_{\Delta f}$. A controller circuit monitors N_Δ and adjusts the CDEs to achieve a desired $T_{\Delta f}$.

Different adjustment algorithms are possible [25]. We refer to one such algorithm as *incremental step delay adjustment*. In this scheme, the CDEs are designed such that $t_{CDE_i^A} = xt_{CDE_{i-1}^A}$, where $t_{CDE_i^A}$ is the delay added to the total ring oscillator loop delay when CDE_i^A is activated, and $1 < x < 2$ is a constant. Such design results in overlapping resolution adjustment steps, which ensures the desired resolution is achieved under process and temperature and power supply variations [25].

3. Noise Performance Analysis and RMS Jitter Measurement

From Eqn. 2a, one of the major sources of measurement error is T_{R1}, which arises from different noise sources in the DTQ. Assuming random noise sources within OscA and OscB as the major error sources [7], the variance of T_{R1} is given by:

$$s_{R1}^2 = 4NMs_g^2 \quad (4)$$

where $N = N_c + N_f$ is the total number of clkA cycles for measuring an interval, M is the number of gates in the ring oscillator A or B, and s_g is the standard deviation of noise for each gate in OscA and OscB. Similarly, the following yields the standard deviation of T_{R2}:

$$s_{R2}^2 = 4N_2Ms_g^2 \quad (5)$$

where $N_2 = N_c + N_f + N_{nf}$.

Eqn. 4 and 5illustrate the noise accumulation problem in ring oscillators as it shows how the RMS noise is directly proportional to the number of transitions in the ring oscillator. In Sec. 3.1, we describe a novel noise scaling technique where we exploit this accumulation behavior to estimate and subsequently achieve increased accuracy. The independence assumption may not necessarily hold for power supply and substrate noise in each of OscA and OscB. However, for common mode noise sources due to power supply and substrate noise, the differential architecture of the DTQ will attenuate such noise significantly. Typically, the differential structure of the DTQ reduces the effect of power supply and substrate noise by a factor of 8 to 10 [25].

3.1. Example ETDC Application: RMS Jitter Measurement

An important specification of many timing circuits, such as PLLs and DLLs is RMS jitter. In the following, we illustrate the use of our ETDC to measure RMS jitter accurately.

We define RMS jitter in the time interval T_d time intervals as follows:

$$J_{RMS} = \sqrt{\frac{1}{M}\sum_{i=1}^{M}(T_{d(i)} - \overline{T_d})^2} \quad (6)$$

where M is the number of samples taken, $T_{d(i)}$ for $i = 1,...,M$, are the measured time intervals, and $\overline{T_d}$ is the average of the $T_{d(i)}$'s.

From Eqns. 2a and 6, for sufficiently large values of M, J_{RMS}^2 is estimated from the following equation [7][25]:

$$J_{RMS}^2 = T_{RMS}^2 - T_{\Delta f}^2/12 - s_{R1}^2 \quad (7)$$

where $T_{RMS}^2 = \frac{1}{M}T_{\Delta f}^2\sum_{i-1}^{M}(N_i - \overline{N})$.

In Eqn. 7, $T_{\Delta f}$ is known from calibration. The RMS error of this estimation is:

$$s_{J_{RMS}^2}^2 = 2/M\,(T_{\Delta f}^2/12 + s_{R1}^2) \quad (8)$$

This error can be very small by making M large (e.g., range from 1000 to 100,000). From Eqn. 7, the accuracy of RMS jitter measurement does not depend on the resolution $T_{\Delta f}$ because this value can be estimated accurately through calibration. Since, reducing s_{R1}^2 implies a difficult design challenge, we use a special technique to estimate s_{R1}^2 accurately and perform specific computations to effectively overcome its effects, and achieve what we refer to as noise floor tunneling. This technique is less limited by design quality, and therefore scales much more readily to different chips and processes.

As evident from Eqn. 7, it is possible to estimate the input jitter accurately if the internal jitter of the ETDC is known. The general form of Eqn. 7 is:

$$s_{tot}^2 = s_{ip}^2 + s_{\text{int}}^2 \quad (9)$$

where s_{tot}^2 is the variance of the jitter estimated from the raw measurements, and s_{ip}^2 and s_{int}^2 are the variances due to the ETDC input (signal under test) jitter and due to the EDTC's internal component (e.g., oscillators, gates) jitters, respectively. We next describe how our new EDTC uses noise scaling [7] to accurately estimate the input jitter.

Eqns. 4 and 5 indicate that the variance of the internal jitter increases with the number of OscA cycles necessary to complete the measurement. After calibration, From Eqns. 2a and 2b, we form Eqns. 10a and 10b, respectively:

$$s_{tot(1)}^2 = s_{ip}^2 + s_{R1}^2 \quad (10a)$$

$$s_{tot(2)}^2 = s_{ip}^2 + s_{R2}^2 \quad (10b)$$

where $s_{tot(1)}^2$ and $s_{tot(2)}^2$ are the total jitter measured with and without considering N_{nf}. From Eqns. 4 and 5, s_{R1}^2 and s_{R2}^2 are related with a scaling factor $a = (\overline{N_c + N_f + N_{nf}})/(\overline{N_c + N_f})$, hence the term *noise scaling*.

Under a random noise assumption, it is possible to achieve arbitrarily high measurement accuracy using this technique by making M sufficiently large, regardless of the input or internal noise distribution. For example, measuring RMS jitter in a 10 GHz signal with 0.1ps accuracy can be achieved using approximately $M = 80{,}000$ samples in a noisy environment. This implementation of noise scaling scheme provides the same accuracy with approximately half the number of samples than the scheme in [7]. This is because, effectively, the intervals used for normal and noise floor samples are the same, as opposed to coming from different sample sets of a statistical superset. Also, this method reduces test time due to reduced number of samples.

An application of the ETDC is to measure RMS period jitter, as shown in Figure 5 [7]. The current version of our design allows for test controller to pass two consecutive edges of IN1 to the ETDC. This procedure can be repeated until a predetermined number of samples of the input signal periods are measured allowing for variance and peak-to-peak jitter to be computed, easily either on-chip or off-chip. A JTAG TAP is used to transfer data for off-chip processing.

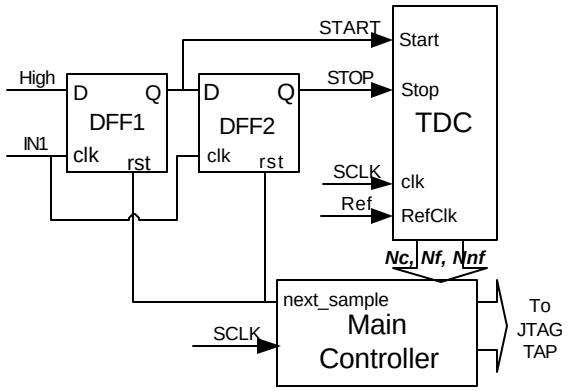

Figure 5: Edge sampler (ES) for period jitter measurement

4. ETDC Simulation Results

We simulated our circuits to evaluate the effectiveness of noise tunneling technique. To accelerate the simulations, a C model was developed for the circuits to simulate the jitter measurement process. The plots in Figure 6 illustrate how our noise estimation and cancellation technique is essential for accurate measurement.

The plots illustrate the measurement accuracy results from simulations at 1 GHz for several values of the input signal jitter (s_{in}), and ETDC internal jitter s_{R1}^2 (s_{R1}^2 is expressed as RMS period independent random period jitter in OscA and OscB). As evident from these plots, our noise cancellation scheme provides accurate estimates of input jitter by estimating and canceling the ETDC internal noises. Plot (a) and (b) in Figure 6 show that accuracy increases with an increase in the number of samples, while this would not occur without using the noise tunneling algorithms (plot (c)). These plots indicate an increase of accuracy by a factor of 1.5 to 2 as compared with the results in [7] for the same total number of period samples. The apparent saturation of accuracy for low s_{R1}^2 values is due to quantization noise effects.

Table 1 reports the number of samples and measurement time required for measuring RMS jitter for different signal frequencies [26]. Typically, the measurement time increase significantly at higher speeds because of the need for higher accuracy. However, even for the very high accuracy required at high speed, the measurement can be completed in 80 ms or less. The measurement time, include sample

measurements, data transfer to off-chip, and processing on a host computer.

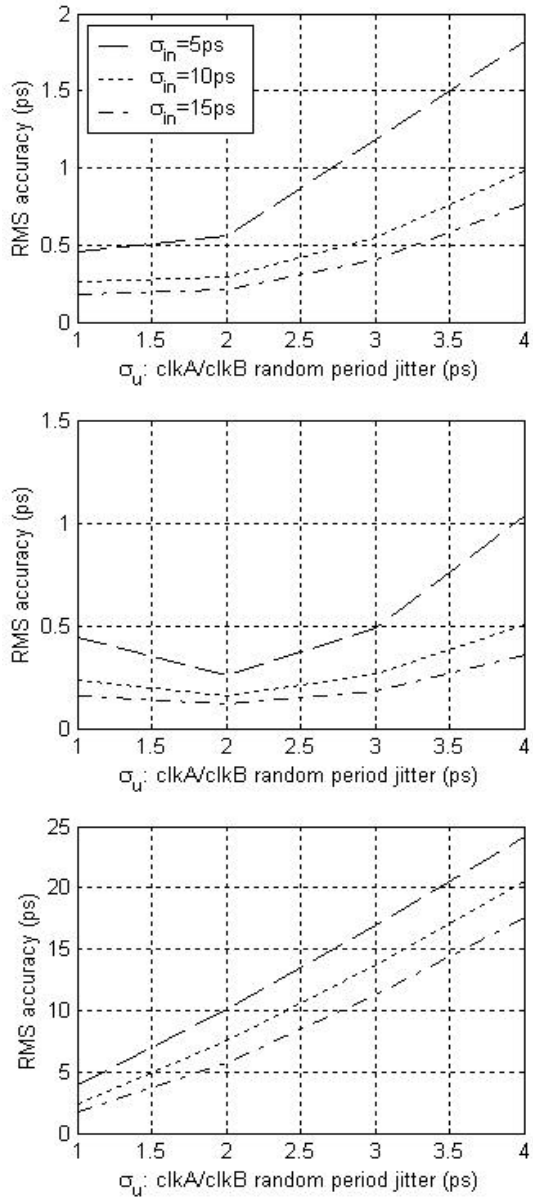

Figure 6: RMS jitter measurement accuracy (f = 1 GHz), (a) 20,000 period samples with noise tunneling, (b) 100,000 period samples with noise tunneling, (c) 20,000 and 100,000 period samples without noise tunneling

Table 1: Total measurement time for RMS jitter measurement

Frequency (MHz)	RMS jitter under test (ps)	RMS accuracy desired (ps)	# of samples required	Total measu-rement time (ms)
500	20	2	1,000	3
1,000	10	1	2,000	8
2,000	5	0.5	5,000	8
5,000	2	0.3	40,000	28
10,000	1	0.2	80,000	60

5. Conclusions

We presented a novel embedded circuit core aimed at performing high-resolution/high-accuracy timing measurements. The accuracy can be of the order of sub-picoseconds in standard CMOS. This capability is particularly useful for testing critical timing specifications of rapidly emerging GHz and Gb/s ICs and SoCs. The circuit is particularly attractive since it amounts to small area. The main contributions of this paper are the added circuit and data processing techniques and features aimed at achieving accuracy not otherwise possible due to oscillator noise limitations. The high accuracy performance is achieved by exploiting a novel noise estimation and cancellation technique based on *noise scaling* that allows for the measurement accuracy to exceed that otherwise dictated by the chip noise floor, what we refer to here as *noise floor tunneling*. The attractive feature of our noise floor tunneling is that it requires only simple computation and therefore can easily be performed off-chip by a standard processor or performed on-chip on a simple dedicated processor. Trade-offs in test time and accuracy are possible giving the circuits and associated techniques further flexibility.

This EDTC can be used within the scalable embedded time interval analysis (ETIA) architecture described in [7] to satisfy practical implementation requirements, such as closeness to test points, support of multiple test points, and better digital noise immunity.

The extensions of our work aim at the accurate measurement of many other specification such as long term jitter, cycle-to-cycle jitter, jitter harmonics, jitter transfer, pattern dependent jitter, frequency, period, skew, delay, PLL/DLL lock time, PLL frequency response, PLL step response, rise/fall time, etc. We are also developing techniques and algorithm

to maintain high accuracy in the case periodic noise sources due to different clock sources on the chip.

References

[1] M. Green et al.,"OC-192 Transmitter in Standard 0.18µm CMOS", *Digest of Technical Papers of the IEEE ISSCC,* San Francisco, CA, Feb. 2002, pp. 248-249.

[2] J. Cao et al.,"OC-192 Receiver in Standard 0.18µm CMOS", *Digest of Technical Papers of the IEEE ISSCC,* San Francisco, CA, Feb. 2002, pp. 250-251.

[3] I. Hwang et al., "A Self-Regulating VCO with Supply Sensitivity of <0.15%-Delay/1% Supply", *Digest of Technical Papers of the IEEE ISSCC,* San Francisco, CA, Feb. 2002, pp. 140—141.

[4] C. Kim et al., "Low-Power Small-Area $\pm 7.28\,ps$ Jitter 1 GHz DLL-Based Clock Generator", *Digest of Technical Papers of the IEEE ISSCC,* San Francisco, CA, Feb. 2002, pp. 142 – 143.

[5] P. Restle et al., "The Clock Distribution of the Power4 Microprocessor", *Digest of Technical Papers of the IEEE ISSCC,* San Francisco, CA, Feb. 2002, pp. 144-145.

[6] Y. Cai, B. Laquai, and K. Luehman, "Jitter Testing for Gigabit Serial Communication Transceivers", *IEEE Design and Test,* Vol. 19, No. 1, Jan./Feb. 2002, pp. 66 – 76.

[7] S. Tabatabaei and A. Ivanov, "Embedded Timing Analysis: A SOC Infrastructure", *IEEE Design and Test of Computers,* vol. 19, No. 3, pp 24-36, May-June. 2002.

[8] B. Nadeau-Dostie, "*Design For At-Speed Test, Diagnosis And Measurement*", Kluwer Academic Publishers, 2000

[9] International Technology Roadmap for Semiconductors, 2001 Edition.

[10] U. Shankar, "Test challenges for SONET/SDH physical layer OC3 devices and beyond," *Intl. Test Conf.*, pp. 502-511, 2001.

[11] M. Lauterbach, "Getting more out of eye diagrams," IEEE Spectrum, pp. 60--63, March 1997.

[12] T. Yamaguchi and M. Soma, "A method for measuring the cycle-to-cycle period jitter of high-frequency clock signals," *IEEE VLSI Test Symposium*, pp 102-110, 2001

[13] R. Kelkar, I. Novof, and S. D. Wyatt, "Integrated circuit chip having built-in self measurement for PLL jitter and phase error," *US Patent #5663991 assigned to IBM Corp*., Sept. 1997.

[14] L. D. Smith and Norman E., "On-chip PLL phase and jitter self-test circuit," *US Patent #5889435 assigned to Sun Microsystems Corp*., March 1999.

[15] S. Cherubal and A. Chatterjee, "A high-resolution jitter measurement technique using ADC sampling," *Proc. of Int. Test Conf.*, pp. 838-847, 2001.

[16] Dan Porat, "Review of sub-nanosecond time-interval measurements," *IEEE Trans. on Nuclear Sciences*, vol. Ns-20, pp. 36-51, October 1976.

[17] D. M. Santos, "A CMOS delay locked and sub-nanosecond time-to-digital converter chip," *IEEE Trans. on Nuclear Science*, vol. 43, pp.1717-1719, June 1996.

[18] J. Kalisz, R. Szplet, J. Pasierbinski, and A. Poniecki, "Field-programmable-gate-array-based time-to-digital converter with 200-ps resolution," *IEEE Trans. on Instrumentation and Measurement*, vol. 46, pp. 51-55, February 1997.

[19] C. Kimsal and J. B. Wilstrup, "Time interval measurement system incorporating a linear ramp generation circuit," *US Patent #6,194,925 assigned to Wavecrest Corporation*, February 27 2001.

[20] Z. Tarczy-Hornoch and P. Young, "Interpolating time interval counter with course count ambiguity elminating means," *US Patent #3,505,594 assigned to W. K. Rosenberry*, April 7 1970.

[21] Walter Curtice, "Time interval measurement," *US Patent #4,165,459 assigned to RCA Corporation*, August 21 1979.

[22] David Chu, "Double vernier time interval measurement using triggered phase-locked oscillators," *US Patent #4164648 assigned to Hewlett-Packard Company*, August 14 1979.

[23] A. Chan and G. Roberts, "A synthesizable, fast and high-resolution timing measurement device using a component-invariant vernier delay line," *Proc. of Int. Test Conf.*, pp. 858-867, 2001.

[24] A. Frisch and T. Rinderknecht, "Jitter measurement system and method," *US Patent #6,295,315 assigned to Fluence Technologies*, September 25 2001.

[25] S. Tabatabaei, *Embedded Test Circuits and Methodologies for Mixed-Signal ICs*. PhD thesis, The University of British Columbia, April 2000.

[26] S. Tabatabaei, "Timing Specification Built-In Self-Test Circuits and Methodologies", *Internal Report,* Vector 12 Corp., February 2001.