

# On-Chip Source Synchronous Interface Timing Test Scheme with Calibration

Hyunjin Kim and Jacob A. Abraham

Computer Engineering Research Center

The University of Texas at Austin, University Station C8800, TX 78712

{hkim, jaa}@cerc.utexas.edu TEL : 512-471-8011 FAX: 512-471-8967

**Abstract**—This paper presents an on-chip test circuit with a high resolution for testing source synchronous interface timing. Instead of a traditional strobe-scanning method, an on-chip delay measurement technique which detects the timing mismatches between data and clock paths is developed. Using a programmable pulse generator, the timing mismatches are detected and converted to pulse widths. To obtain digital test results compatible with low-cost ATE, an Analog-to-Digital Converter (ADC) is used. We propose a novel calibration method for the input range for the ADC using a binary search algorithm. This enables test results to be measured with high resolution using only a 4-bit flash ADC (which keeps the area overhead low). The method achieves a resolution of 21.88 ps in 0.18 $\mu$  technology. We also present simulation results of the interface timing characterization, including timing margins and timing pass/fail decisions.

*Keywords* - Source-Synchronous, Memory Interfaces, ATE, Delay Measurement, Flash ADC, Calibration

## I. INTRODUCTION

With increasing data rate requirements, the consequent difference in the bandwidth of the interface between the processor and memory has become a major bottleneck for overall system performance. A memory system with source synchronous Double Data Rate (DDR) interface has been widely used to improve memory speed performance. The maximum speed for memory is projected to rise to 3200 MHz for DDR4 from 200 MHz for DDR1 [1]. DDR transfers data words both at rising and falling clock edges. Increasing speed and double-data operations make at-speed testing of memory interface timing very difficult.

Automated test Equipment (ATE) is commonly used to test memory I/O parameters. However, ATE has no source synchronous function, and data coming to the ATE from the device under test (DUT) is not deterministic. Therefore, ATE needs to implement complicated functions like phase alignment and data-strobe scanning logic [2], [3]. Higher frequency requirement needs higher Edge Placement Accuracy (EPA) and higher resolution, which increases the hardware complexity of ATE. This makes I/O timing test costly and limited by the clock frequency and the accuracy of the tester. Furthermore, off-chip test methods for testing high speed signals cannot avoid signal integrity issues because of impedance mismatch and additional parasitic capacitance

loads. Even small amount of parasitics can critically affect the interface timing for high speed operation. These issues exacerbate accurate test for high speed interface timing at the wafer probe, which increases time-to-market. Therefore, on-chip solutions are promising because of lower cost, good resolution and shorter time-to-market.

In [4] and [5], a Built-In Self Test (BIST) method is presented for I/O timing test. The basic idea is to generate the time difference between data and clock using a Voltage Controlled Oscillator (VCO) [4] and a Delay Locked Loop (DLL) [5], and sweep the time difference to capture valid data. However, the strobe-scanning method takes a long time to obtain full test coverage for all data patterns. In [6], a differential Time-to-Digital Converter (TDC) based on vernier delay lines was implemented to generate the timing interval between data and clock. The method achieves a high resolution of 10 ps, but it is costly because of the requirement of a high-speed external clock.

This paper presents a technique for I/O timing test which overcomes the limitations of the previously presented methods. The goal of I/O timing test is to measure the timing mismatch between data and clock and thus find the timing parameters (setup and hold times). Our method uses a delay measurement method instead of strobe-scanning method for testing I/O timing to decrease test time. We use a programmable pulse generator to generate the timing delay and convert it into a pulse. The output of the pulse generator is converted to voltage using a pulse-to-voltage converter. Finally an ADC is used to obtain a digital output compatible with low-cost ATE. We also present a novel calibration method for the input range of the ADC to keep area overhead low.

## II. DESIGN METHODOLOGY

In source-synchronous interfaces, the data and clock paths are designed to match. However, delay mismatches are not avoidable due to process variations, crosstalk, Inter-Symbol Interference (ISI), and power noise. The amount of mismatch between two paths directly affects the interface timing parameters. Our goal is to measure the delay difference between data and clock paths instead of sweeping the data-strobe with respect to the data. For this technique, complicated test vectors

or timings are not required. Also, our scheme provides a digital test result to be compatible with a low-cost tester.

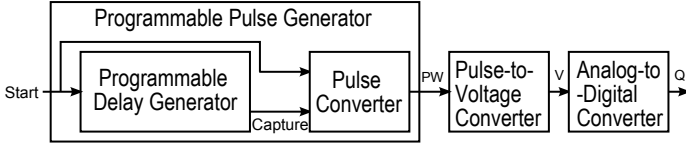


Fig. 1. A Delay Measurement Method

To measure the small delay mismatches, we generate a pulse and measure the pulse width variation while the pulse is passing through data and clock paths. Figure 1 shows our developed scheme that consists of a Programmable Pulse Generator (PPG), a Pulse-to-Voltage Converter (PVC), and an Analog-to-Digital Converter (ADC). The PPG consists of a Programmable Delay Generator (PDG) and a pulse converter to generate a pulse. Using the PVC and ADC, we digitally measure the pulse width. Moreover, we present a novel calibration method to reduce the area overhead for ADC while maintaining a high resolution.

### III. CIRCUIT IMPLEMENTATION

#### A. Circuit Architecture

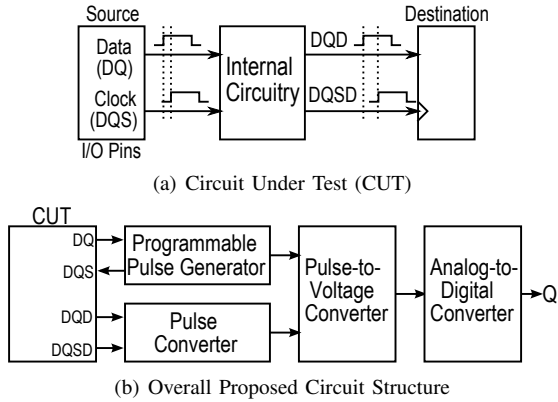


Fig. 2. Circuit Structure for the Interface Timing Tests

We model a Circuit Under Test (CUT) to apply our delay measurement scheme for testing the interface timing parameters is shown in Figure 2(a). The CUT has models of  $DQ$  and  $DQS$  paths to test the data setup and hold times in the DDR memory devices. The  $DQ$  and  $DQS$  are the signals at the source, and the  $DQD$  and  $DQSD$  are the propagated signals to the destination. The amount of mismatch from the source to the destination needs to be measured to check if the timing specifications between  $DQ$  and  $DQS$  are satisfied. Therefore, we measure the time differences between  $DQ$  and  $DQS$ , and  $DQD$  and  $DQSD$ . Then, the delay mismatch between the source and the destination is calculated. The overall circuit architecture is shown in Figure 2(b). At the source, using the PCG,  $DQS$  is generated with a programmable delay with respect to  $DQ$ . The time differences between  $DQ$  and  $DQS$  paths are converted to pulses using the PPG. The difference

between the pulse widths indicates the delay mismatch. To digitally read out the amount of the delay mismatch, the pulse widths are converted to voltage levels using ADC. This makes our design compatible with low cost ATE.

#### B. Programmable Pulse Generator

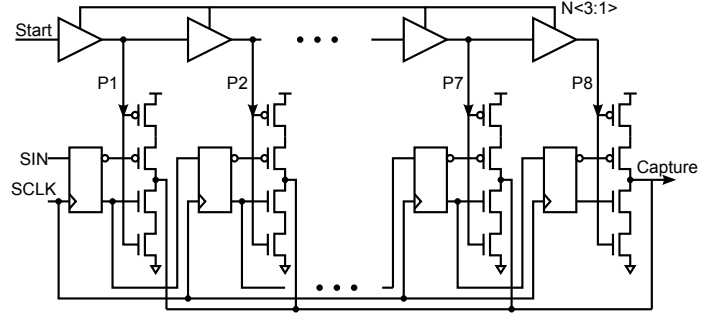


Fig. 3. Programmable Delay Generator

The PPG consists of the PDG and the pulse converter. The schematic of the PDG is shown in Figure 3. For the PDG, we modified the programmable capture generator presented in [7]. The programmable delay from *Start* to *Capture* is set by the control signals,  $N<3:1>$ , of the buffers and the scannable flip-flops. The number of the buffers selected is determined by a one-hot sequence,  $SIN$ . The buffer was implemented using a segmented-type driver to precisely control the buffer delay. The pulse converter consists of a rising edge detector and a SR latch. The rising edge detector produces a signal with the pulse width corresponding to three inverter delays. The SR latch simply forms the pulse width using the *Start* and *Capture* signals. On the other hand, the propagation delays of the *Set* and *Reset* operations of the SR latch are not the same, and thus the generated pulse width is less than the actual delay between *Start* and *Capture*. The non-symmetrical characteristics of the SR latch is not a problem for the test circuit. This is because we measure the relative delay difference at the source and the destination by using the same pulse converter at both ends.

#### C. Pulse-to-Voltage Converter

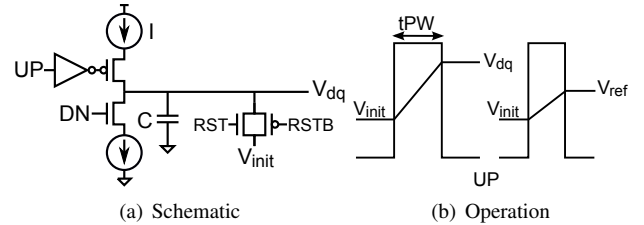


Fig. 4. Pulse-to-Voltage Converter

The pulse widths generated at the source and the destination are converted to voltage levels to measure the pulse difference. For this operation, the PVC is implemented as shown in Figure 4(a). It consists of the current steering logic and the capacitor. The generated pulse,  $UP$ , increases the voltage across the capacitor after the pulse switches on. Thus, the output voltage level,  $V_{dq}$ , is changed according to the pulse width as

shown in Figure 4(b). As the pulse width becomes narrower, the output voltage level becomes lower. We also define  $V_{init}$  as the initial voltage level of  $V_{dq}$  to have the same initial level at the beginning of each test. Moreover, the voltage-level range can be calculated since we can estimate the range of  $UP$  pulse. Thus,  $C$ , the capacitor value, is also determined by the equation (1) during the design. The capacitor needs to be fully charged to  $V_{dq}$  during the pulse period. Accordingly, the capacitor value is determined by the pulse width of  $UP$  and the maximum variation between  $V_{init}$  and  $V_{dq}$ . Therefore, we need to consider the maximum allowable pulse width ( $\Delta tPW_{max}$ ) during the design to select the optimum capacitor value.

$$C = \frac{I(\Delta tPW_{max})}{\Delta V_{max}}, \quad \Delta V_{max} = I(\Delta tPW_{max})/C \quad (1)$$

$$V_{dq, max} = V_{init} + \Delta V_{max}$$

#### D. Analog-to-Digital Converter with Calibration

The ADC is used to read out the test results digitally which makes it compatible with low cost ATE. The ADC structure is shown in Figure 5(b). We use a flash ADC because of its simple structure. The flash ADC consists of a bias generator, a resistor ladder, differential amplifiers, comparators, latches, and an encoder. However, it consumes a large area and power to achieve high resolution because it needs  $2^N - 1$  comparators and amplifiers, respectively, where  $N$  is the ADC resolution. We use a novel calibration method to reduce the number of

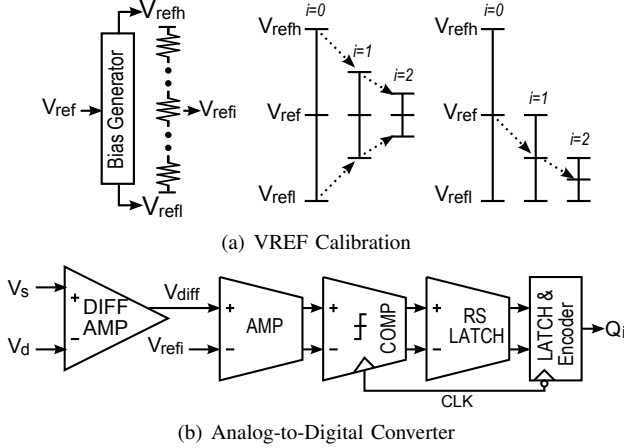


Fig. 5. Analog-to-Digital Converter with Calibration

ADC bits without compromising high resolution. The goal of this calibration method is to adaptively change the input range (difference between  $V_{refl}$  and  $V_{refh}$ ) of ADC.  $V_{diff}$  indicates the pulse width difference between source and destination. Since input pulse width ( $V_{diff}$ ) is known, we use it to calibrate the input range of ADC. Figure 5(a) shows two ways of calibrating the input range of ADC. One is to change  $V_{refl}$  and  $V_{refh}$  and the other is to change  $V_{ref}$ .  $V_{ref}$  is the generated signal at source based on the given timing specification. The ADC reference voltages of  $V_{refl}$ ,  $V_{refh}$  and  $V_{ref}$  are defined by the following

equation (2), where  $V_{ref}$  and the  $\Delta V_{max}$  are known.

$$V_{refh} = V_{ref} + V_m, \quad V_{refl} = V_{ref} - V_m, \quad V_m = \frac{1}{2}\Delta V_{max} \quad (2)$$

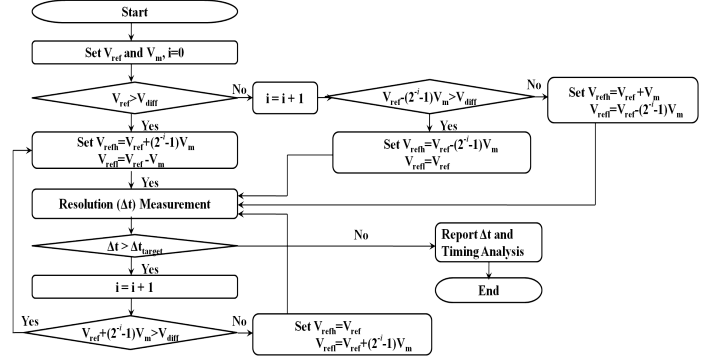
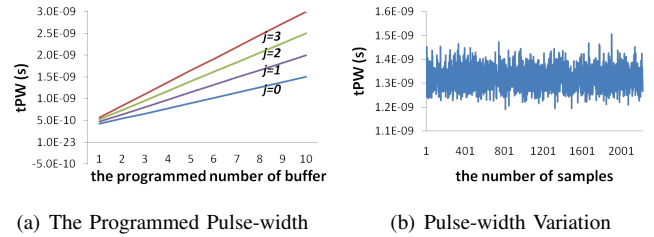


Fig. 6. Calibration Flow Chart

The calibration flow chart for  $V_{refl}$  and  $V_{refh}$  is described in Figure 6.  $V_{diff}$  lower than the  $V_{ref}$  means that the timing specifications are satisfied.  $V_{refl}$  and  $V_{refh}$  are initially set to ' $V_{ref} \pm V_m$ '. The calibration uses the binary search algorithm which compares  $V_{diff}$  with the middle of the reference levels. This process minimizes the difference between the reference levels and  $V_{diff}$ . At the initial step,  $i=0$ , the test circuit determines the pass or fail for the timing specifications by comparing the  $V_{diff}$  with the  $V_{ref}$ . After the decision, the test resolution is calculated and then compared with the target resolution,  $\Delta t_{target}$ . If the test resolution is not within the tolerance range compared to the target value, the reference levels are narrowed down. These steps are repeated until the test resolution is within the tolerance range. Because of the binary search algorithm, we obtain the target resolution with only a few calibration steps. After the calibration is completed, the test circuit reports the test resolution and the ADC outputs, and then analyzes the timing parameters. Thus, we obtain high test resolution using a small number of ADC bits.

#### IV. SIMULATION RESULTS

The on-chip test circuit shown in Figure 2(b) was implemented using the 0.18  $\mu m$  TSMC process. Transistor-level simulations using Hspice as well as behavior-level simulations using Matlab were performed to validate the test circuit and the calibration technique, respectively. Figure 7(a) shows the



(a) The Programmed Pulse-width

(b) Pulse-width Variation

Fig. 7. Programmable Pulse Generator

programmed pulse-width, the simulation result of the PPG

circuit. According to the number of the selected buffer, the pulse width is represented by the following equation (3).

$$t_{PW} = t_{uk} + t_{offset}, \quad t_{uk} = k \times (t_c + t_{fj}); \quad t_{fj} = j \times t_f \quad (3)$$

$$k = [1, N_c], \quad j = [1, N_f]$$

where  $t_{uk}$  is the programmed delay and  $t_{offset}$  is the constant offset delay due to the capacitance loadings.  $t_c$  is the one buffer delay and  $t_f$  is one segment unit delay where  $N_c$  is the number of the buffer and  $N_f$  is the number of segmented buffer unit. For the PCG, 10 buffers are used and one buffer is segmented into 3 unit delays. According to the programmed numbers,  $k$  and  $j$ , the programmed pulse width is shown in Figure 7(a) for a range from 500 ps to 3 ns. Figure 7(b) shows the result of 2000-trial Monte Carlo simulations to estimate the pulse-width variations due to intra-chip parameter variations. 200 ps variation of the input range of ADC is used for its calibration. Since we know the pulse width and its variation as shown in Figure 7, our calibration technique works well. The pulse width is converted to a voltage level to measure the test results. To validate the conversion relationship, PVC simulations are performed. Figure 8(a) shows the relationship

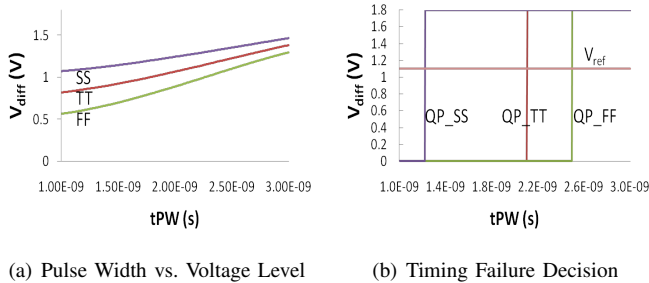


Fig. 8. Pulse-to-Voltage Converter

between  $tPW$  and  $V_{diff}$  for different process corners. As can be seen,  $V_{diff}$  is changed from 0.8 V to 1.36 V while  $tPW$  is controlled from 1 ns to 3 ns under the typical conditions. In other words, the voltage level variation is 0.56 V while the pulse width variation is 2 ns. Accordingly, the conversion ratio,  $R$ , between two parameters is approximately ‘2.8 mV/10 ps’. Since the pulse width is linearly converted to the voltage level, the parameterizations of the test results are available. After the conversion,  $V_{ref}$  is chosen to start the calibration. At the first calibration step, the timing pass or fail is determined by comparing  $V_{diff}$  with  $V_{ref}$ . The timings fail when  $V_{diff}$  is larger than  $V_{ref}$ . For the simulation, we set  $V_{ref}$  to 1.08 V, which is the middle value of  $V_{diff}$  range under the typical conditions. However,  $tPW$  and  $V_{ref}$  can be controlled depending on the required timing specifications. Figure 8(b) shows the comparison results of  $V_{diff}$  and  $V_{ref}$ ,  $QP$ , to show the points where the timing failure occurs. The difference between  $V_{diff}$  and  $V_{ref}$  indicates the timing margin. We precisely find the difference between  $V_{diff}$  and  $V_{ref}$  through the calibration procedure. We extract the test resolution and then compare with the target resolution at each calibration step. The test resolutions are calculated by equation (4) where  $i$  indicates

TABLE I  
RESOLUTION CALCULATION

$i$	$\Delta V_{max}$	$\Delta t$
1	0.8 V	87.5 ps
2	0.4 V	43.75 ps
3	0.1 V	21.88 ps

the calibration step.

$$\Delta t = \frac{C}{I} \frac{\Delta V_{max}}{2^i} \frac{I}{2^N} \quad (4)$$

The resolutions are calculated as shown in Table I. The  $\Delta V_{max}$  is changed by the variable reference levels during the calibration procedure. By using the binary search algorithm, we obtain sufficient resolution only at the third step of calibration. Therefore, only a 4-bit flash ADC is required to measure the test results with high resolution. After the calibration, the ADC digital outputs,  $Q_i$ , are reported and the timing margin,  $tM$ , is parameterized using the  $tPW$ ,  $\Delta t$ , and the conversion ratio ( $R$ ) as shown in equation (5).

$$tM = |tPW(V_{ref}) - tPW(V_{diff})| \times R \times \Delta t \times Q_i \quad (5)$$

## V. CONCLUSION

This paper has addressed the issues of increasing cost and time for testing interfaces by using an on-chip test circuit with a novel calibration technique. The test methodology is based on the delay measurement of data and clock paths instead of the strobe-scanning method. For testing source synchronous interface timing, we detect the timing mismatches between data and clock paths by measuring the path delay difference. The presented method compares the signals generated at different timings and then parameterizes the timing margin. A low area overhead ADC with calibration is designed for the measurements. Because of the predictable ADC input range, the calibration technique is applied to save an area that on-chip testing circuits take up. We have achieved around 20 ps resolution in 0.18 $\mu$  technology by using a 4-bit flash ADC with a calibration involving only 3 steps.

## REFERENCES

- [1] JEDEC Solid State Technology Association, <http://www.jedec.org/>.
- [2] A. T. Sivaram, M. Shimanouchi, H. Maassen, and R. Jackson, “Tester Architecture For The Source Synchronous Bus,” in *Proc. Int. Test Conf.*, 2004, pp. 738–747.
- [3] B. Laquai, M. Braun, S. Walther, and G. Schulze, “Flexible and scalable methodology for testing high-speed source synchronous interfaces on automated test equipment (ATE) with multiple fixed phase capture and compare,” in *IET Computers & Digital Techniques*, vol. 1, no. 3, 2007, pp. 154–158.
- [4] H. Kim and J. Abraham, “A Low Cost Built-In Self-Test Circuit For High-Speed Source Synchronous Memory Interfaces,” in *Proc. Asian Test Sympo.*, pp. 123–128.
- [5] C. Jia and L. Milor, “A BIST Solution for the Test of I/O Speed,” in *Proc. Int. Test Conf.*, 2003, pp. 1023–1030.
- [6] K. Yamamoto, M. Suda, and T. Okayasu, “2GS/s, 10ps Resolution CMOS Differential Time-to-Digital Converter for Real-Time Testing of Source-Synchronous Memory Device,” in *Proc. IEEE Custom Integrated Circuits Conf.*, 2007, pp. 145–148.
- [7] H. Kim and J. Abraham, “On-Chip Programmable Dual-Capture for Double Data Rate Interface Timing Test,” in *Proc. Asian Test Sympo.*, 2011.