Improved Delay Measurement Method in FPGA based on Transition Probability

Justin S. J. Wong and Peter Y. K. Cheung *
Department of Electrical and Electronic Engineering, Imperial College, London, UK
{justin.s.wong02, p.cheung}@imperial.ac.uk

ABSTRACT

The ability to measure delay of arbitrary circuits on FPGA offers many opportunities for on-chip characterisation and optimisation. This paper describes an improved delay measurement method by monitoring the transition probability at the output nodes as the operating frequency is swept.

The new method uses optimised test vector generation to improve the accuracy of the test method. It is effectively demonstrated on a 4th order IIR filter circuit implemented on an Altera Cyclone III FPGA.

Categories and Subject Descriptors

B.8.1 [**Performance and Reliability**]: Reliability, Testing, and Fault-Tolerance

General Terms

Measurement, Performance

Keywords

FPGA, Transition Probability, Timing, Self Test

1. INTRODUCTION

Reconfigurability of FPGA, whether at power-up or during run-time, can be exploited effectively for self-testing and self-characterisation. Since the test hardware can subsequently be reconfigured to perform operational functions, the costs of including such test circuits are limited to a small overhead in memory storage for the test configuration and the extra configuration time, either during power-up or during operation. Recently a number of techniques have been proposed to provide not just "go" or "no-go" test results, but to measure the speed of either combinatorial circuit paths [12, 17, 10, 4, 11, 8] or even complete circuit modules with sequential circuits [16]. The ability to measure delay in specific circuits opens up many new possibilities. For example, these techniques can be employed to measure delay variability [17] and timing degradation [14] in the latest generation of FPGAs. These test methods concerning the unique delay mapping of circuits in FPGAs are also essential to hardware security schemes such as *Physical Unclonable Function* (PUF) [7], as well as delay-aware placement and routing methods [5, 3, 9, 13] that provide promising solutions against process variability in FPGAs to improve reliability.

Among all the proposed technique, the one based on transition probability (TP) [16] is the most promising because: 1) it is capable of measuring delays in both combinatorial and sequential circuits; 2) it is essentially a black-box approach, not requiring detail knowledge of the internal circuitry; 3) it can be implemented on existing FPGAs as built-in self-test (BIST); 4) it is time and resources efficient. While earlier results demonstrate the potential of this technique, the previously published work by the authors left a number of important fundamental questions unanswered relating to the accuracy of the measurements, the sensitivity of the technique to different types of timing errors, and the optimality of the test stimuli beyond using a test vector set that is uniformly distributed. Within this context, the new contributions of this work are: 1) a detailed analysis of the behaviour of transition probability in complex multi-path circuits; 2) an in-depth study of the timing error sensitivity of the TP technique in digital circuits; 3) a novel method to optimise the sensitivity by controlling the probability distribution of the test stimuli to provide high measurement accuracy; 4) the improved technique is applied and demonstrated on both complex combinatorial and sequential circuits on an Altera Cyclone III EP3C25 FPGA.

2. BACKGROUND

2.1 The Transition Probability Test Method

An indirect timing measurement method based on transition probability (TP) was proposed in [16]. It has three key features: (a) It is able to measure the delay of components such as interconnects, LUTs and registers involved in typical user circuits. (b) The test circuit itself is robust against timing failure, measurement accuracy is largely independent from process variation and degradation of the test circuitries. (c) No structural change or internal signals probing of the circuit-under-test (CUT) is required, delay can be measured by pure observation of the output.

The test method estimates the propagation delay of a specific circuit path indirectly by measuring the transition probabilities of the output node. The transition probability at a signal node is defined as the probability that the node will

* The authors would like to acknowledge the support of Altera Corporation, Terasic and the EPSRC under the grants: EP/H013784/1 and EP/I012036/1.

FPGA'11, February 27–March 1, 2011, Monterey, California, USA.

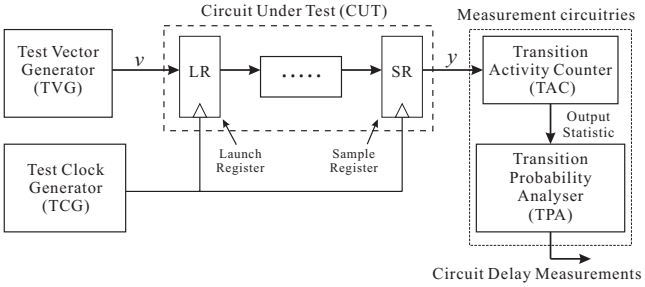

Figure 1: Basic principle of the delay measurement method.

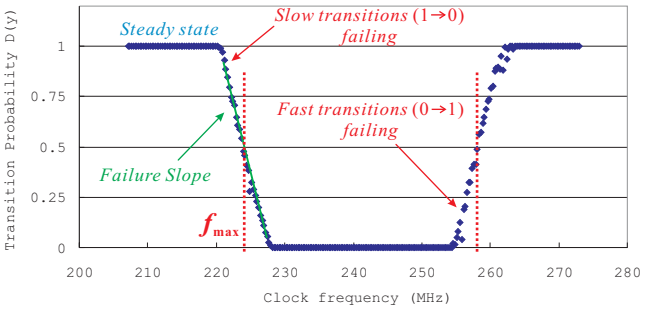

Figure 2: Example of a typical output transition probability $D(y)$ against frequency from [14].

change state when the next input stimuli is applied to the circuit. For a sequence of output samples $y(k)$, $k = 1, \ldots$, the transition probability $D(y)$ is defined in [16] as:

$$P\{y(k+1) = \overline{y(k)}\} = P\{y(k) = 0\}P\{y(k+1) = 1\} + P\{y(k) = 1\}P\{y(k+1) = 0\} \quad (1)$$

The test measures the transition probabilities at the output signal node while ramping the clock frequency up. By detecting changes in the transition probability, it is possible to indirectly derive the frequency at which the circuit starts to fail and hence its propagation delay.

Consider a functional combinatorial circuit with one input and one output z. It can be seen that any transition at the output must be the result of a transition at the input. Therefore if the input is driven by a source with stationary transition probability, the output will also exhibits a stationary transition probability (unchanging).

For our test method, we capture the output of the Circuit-Under-Test (CUT) with a register at a certain clock frequency f_{clk}. The register captures a sample $y(k)$ of the output z at time T after applying the input $v(k)$. If the clock frequency is low enough, then the CUT operates without fault: $y(k) = z(k)$ and so $D(y) = D(z)$. However, because of propagation delays in the CUT, the output z will only change some time after the input is applied. If the test clock frequency is increased, at some point the CUT will begin to fail, and y will begin to sample the value of z from the previous cycle, such that $y(k) = z(k-1)$ some of the time. This changes the output transition probability $D(y)$. Therefore, finding the frequency where the $D(y)$ begins to deviate from its stationary value will yield an accurate measure of its maximum operating frequency f_{max}. This statement holds true as long as the CUT has only a single input to output path and the input is driven by a stationary process, such as a signal that toggles every clock cycle.

Fig. 1 shows the general structure of the test circuit. The circuit-under-test (CUT) input is driven by a Test Vector Generator (TVG). The CUT contains two registers (LR and SR) for launching the input and sampling the output of the combinatorial circuit between them. The registers are controlled by a common clock from the Test Clock Generator (TCG). The TCG contains runtime reconfigurable PLLs, allowing the clock frequency to be changed during a test. The timing resolution of the test is given by $\Delta t \approx \frac{\Delta f}{f^2}$ in [16], which depends on the clock frequency (f) and the size of frequency steps (Δf) during the frequency sweep. Using $\Delta f = 0.25$MHz at 500MHz would yield a considerably good timing resolution of 1ps. The output from the sample register is processed by the Transition Activity Counter (TAC) which is essentially a simple asynchronous counter that counts the number of transition in y over a certain period of time. Transition probability can be derived from the transition count using:

$$D(y) = \frac{\text{signal transition count}}{K} \quad (2)$$

where K is the number of output samples or the number of clock cycles elapsed in the counting period. The calculation of $D(y)$ is carried out by the Transition Probability Analyser (TPA) using (2) and then organised into a detailed Transition Probability profile (TP profile) of the CUT over the range of test frequencies. An example of a TP profile is shown in Fig. 2 taken from a CUT containing 9 LUTs on a Cyclone III EP3C25 [14]. As can be seen, the profile begins with a stationary plot at low frequency but declined steeply when timing failure began at approximately 220MHz. This change corresponds to the failure of slower signal transitions. In this case, the $1 \rightarrow 0$ transitions. The gradient and shape of the failure slope is related to the clock jitter and characteristics of the registers respectively. The second failure slope shows the failure of the faster $0 \rightarrow 1$ signal transitions and the transition probability returns to its initial stationary level after both types of transitions have completely failed.

The beauty of the TP method is that it gives more than just the worst-case delay for each CUT — it is able to measure the two types of transition separately. This can potentially be useful for design level timing optimisation where signals are deliberately inverted between combinatorial nodes to even out and reduce the impact of the slow transitions on the overall worst-case propagation delay. The versatility of test results, non-invasive nature and the high measurement precision makes it an ideal candidate for delay measurement of a wide range of arbitrary circuits on FPGAs.

3. MEASURING COMPLEX MULTI-PATH CIRCUITS

The general TP measurement circuitry shown earlier in Fig. 1 can be adapted to test complex multi-path circuits by using a pseudo random test vector generator. Since the vector generation process is stationary, the statistics of the resultant random test vectors are also stationary. Apart from transition probability $D(y)$, the random vectors' statistics can also be quantified by the probability of a logical *high* occurring, which we termed the *High Probability* (HP) or

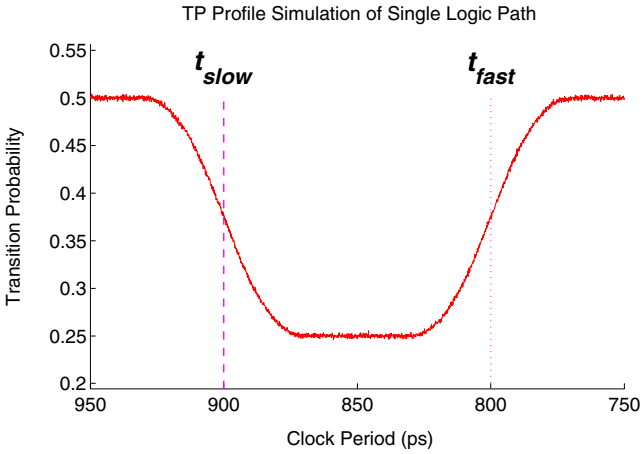

Figure 3: An example of basic TP profile of a single logic path failing at t_{slow} and t_{fast} for falling and rising transitions respectively.

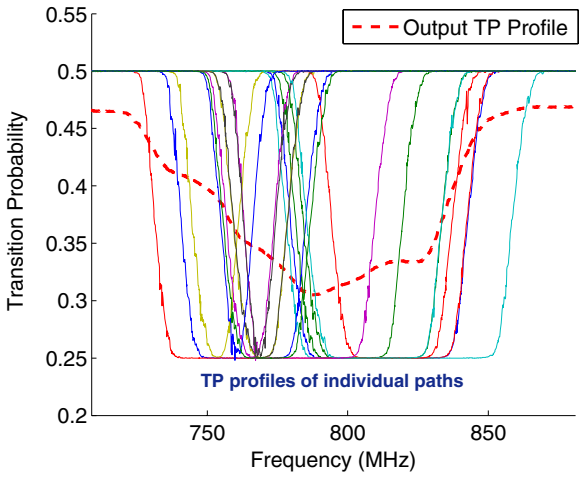

Figure 4: A TP profile measurement of the 2nd LSB output of a 9x9 embedded multiplier on the Cyclone III EP3C25. The unusual shape of the TP profile is the result of individual paths failing at different times. The corresponding paths are isolated and tested separately to obtain their basic TP profile components for reference.

$H(y)$ of signal y. It has a range of 0 to 1, where 0 and 1 implies a stuck at *low* or *high* respectively. In the case of random bit sequences, the values of TP and HP are linked by a simple quadratic relationship:

$$D(y) \approx 2 \times H(y) \times (1 - H(y)) \tag{3}$$

When defining or quantifying the statistic of random input sequences, the use of HP is preferred, because it represents a unique random bit pattern. TP values, on the other hand, could result in two different HP solutions according to (3) with opposite bit patterns, causing unnecessary confusions. The only exception where TP points to a unique random bit pattern is when it is at its maximum — $D(y) = 0.5$.

3.1 Characteristics of Transition Probability

3.1.1 Basic TP Model

Fig. 3 depicts a simulated TP profile of a single logic path with uniformly distributed random input sequence. The falling and rising transitions are assigned different propagation delay values t_{slow} and t_{fast}. The gradual failure slopes are caused mainly by the stochastic behaviour of clock jitter [16], where it can be describe by a random variable τ in terms of the relative time from the expected clock edge, with a specific probability density function $PDF_{Jitter}(\tau)$. By assuming each clock edge has independent random jitter and consistent $PDF_{Jitter}(\tau)$ throughout the test frequency range, the behaviour of the TP profile as a function of clock period (T) can be approximated from the cumulative distribution of the PDFs centered at t_{slow} and t_{fast}:

$$TP_{indep}(T) \approx \frac{1}{2}\left[\frac{3}{4} + \left(\frac{1}{2} - \int_{-\infty}^{t_{fast}-T} PDF_{Jitter}(\tau)\, d\tau\right) \times \left(\frac{1}{2} - \int_{-\infty}^{t_{slow}-T} PDF_{Jitter}(\tau)\, d\tau\right)\right] \tag{4}$$

where $t_{fast} \leq t_{slow}$.

The behaviour of the resultant TP profile also depends on the degree of jitter correlation between consecutive clock edges, which affects the timing failure interaction between the rising and falling transitions through the CUT. According to [6], most PLL generated clock signals are likely to exhibit multi-cycle jitter, which introduces edge-to-edge jitter correlation. Therefore, it is important for the model to also cover such correlated case. The TP behaviour with complete correlation is given by:

$$TP_{corr}(T) \approx \frac{1}{2}\left[1 - \frac{1}{2}\left(\int_{-\infty}^{t_{slow}-T} PDF_{Jitter}(\tau)\, d\tau - \int_{-\infty}^{t_{fast}-T} PDF_{Jitter}(\tau)\, d\tau\right)\right] \tag{5}$$

where $t_{fast} \leq t_{slow}$.

Note that both TP_{indep} and TP_{corr} gives identical results in normal cases when the failure caused by t_{fast} and t_{slow} do not overlap (Fig. 3). Yet, when the two types of failure do overlap, the jitter correlation causes their respective change of TP to cancel each other out, reducing the magnitude of change in the TP profile.

Clock signals in real systems are likely to exhibits both independent and correlated jitter. Therefore, a combination of TP_{indep} and TP_{corr} can be used:

$$TP(T) = (1 - k) \times TP_{indep} + k \times TP_{corr} \tag{6}$$

where k defines the correlation factor ranging from 0 to 1. In reality, it is highly unlikely to have perfect edge-to-edge correlation ($k = 1$). Therefore, the TP profile should always show a measurable amount of change, even if t_{fast} and t_{slow} are exactly identical.

3.1.2 Analysis of Multi-Path TP Profile

The previously described single path models are useful for predicting the TP profile of a failing path. Yet, the problem with them is that they are not scalable to more complex multi paths circuits. Fig. 4 depicts the TP profile of the 2nd LSB output of a 9x9 embedded multiplier on the Cyclone III EP3C25. As can be seen, the observed output TP profile is related to all the basic TP profiles of each individual path. While the TP profile may appear to be a direct combination

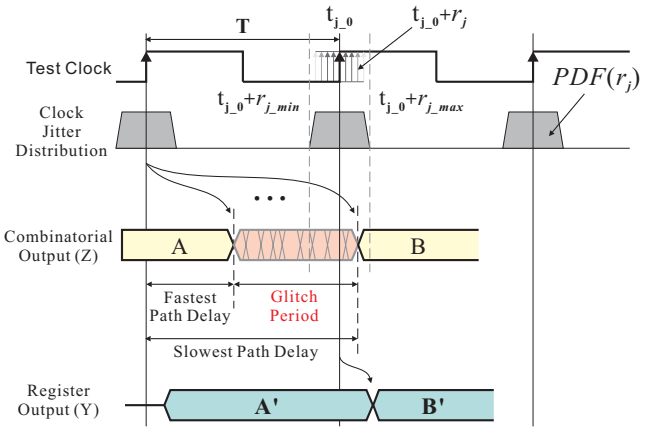

Figure 5: Timing diagram showing the activity of the output bit (Z) and registered output (Y) in a multi input paths to single output circuit. A certain *glitch period* occurs after each clock edge due to variations between propagation delay of different paths. The position of clock edge is governed by the jitter distribution $PDF(r_j)$, and the probability of the register capturing the correct value in B′ depends on both the glitch pattern and the overlapping jitter region.

of the basic TP profile components of the paths, it is actually not possible to recreate the exact TP profile using the basic single path components alone. The main reason for this is that the failure process of the paths are interrelated with each other in a difficult to predict manner.

Consider the timing illustration in Fig. 5, where a circuit with multiple internal paths is stimulated by random vectors. The probability that an input transition through a particular path is observable at the output depends on the input pattern and the state of the other paths, which means each path could contribute differently to the observed TP profile. Such behaviour is only predictable if the exact circuit implementation, structure and layout are known.

Although each active path may produces a signal transition some time after the clock edge, their different arrival time result in a "glitch period" containing a series of unwanted transition activities. These glitch activities are unpredictable especially when random input vectors are used. When the glitch period coincides with the next clock edge, where the clock edge position itself is unpredictable due to clock jitter, the actual value captured by the register (B′) is not deterministic, and hence the resultant transition probability cannot be determined with certainty. Also, the rapid transitions in the glitch period could cause undesirable metastability problem in the output register [2], further increasing the unpredictability of the output value.

For these reasons, the direct approach of modelling the TP profile based on specific path quickly becomes impractical with complexity. For FPGA designs, a mere change of placement and routing could produce a layout with completely different TP profile. The only way that a precise model of the TP profile can be obtained is if a perfect physical model of the circuit is available with precise information on signal propagations, interactions, and clock jitter behaviour, so that the exact glitch pattern is known and the registered output value is predictable. Though, if such perfect physical model exists, a delay measurement method would not be necessary in the first place. A better strategy would be to consider the timing error sensitivity of TP rather than its exact profile, and deduce an effective way to control its sensitivity to timing errors in complex circuits, such that good measurement accuracy is achieved.

3.1.3 Controlling Sensitivity of TP to Timing failure

Timing error Sensitivity of TP for a circuit is defined as the difference between the normal operating level of output TP and the level of TP after the slowest type of signal transitions through the worst-case path has failed. The higher the difference, the more likely errors are detected and hence provide better sensitivity to timing failure. The ability to control the sensitivity of TP against timing failure allows the test method to produce more reliable results, avoiding inaccuracy caused by sensitivity loss. There are three typical cases where sensitivity could be affected:

(i) Sensitivity dilution – a logic block with large number of inputs converging to one output suffers from reduced observable TP failure response. This problem can be easily observed in an N-input AND gate where errors can only propagate through when all inputs are *high* and the TP sensitivity decreases as N increases.

(ii) Sensitivity blocking – in a circuit with multiple combinatorial stages separated by pipeline registers, the changes in TP profile due to timing failure of one stage could be blocked by its following stage(s) under certain conditions, causing it to be invisible at the output.

(iii) Failure blind spot – when a logic block with N inputs is supplied with inputs S_N with certain $H(S_N)$, the failure of specific internal paths may not cause any observable change at the output TP profile.

The problem of diluted sensitivity (i) is unavoidable in most cases, especially with random test vectors. Yet, the sensitivity is only reduced and never completely lost, meaning that it can be improved by taking a higher number of transition count samples to form a TP profile with less residue noise from the random inputs and hence higher relative sensitivity (see (2)). This approach, however, increases the total test time and it does not solve the problems in cases (ii) and (iii) where complete loss of sensitivity is possible.

To provide a general solution for the three cases while maintaining short test time, we propose a method that can improve TP sensitivity by controlling the statistic of the random input vector in terms of high probability (HP).

In Fig. 6, the sensitivity of rising or falling transition failure in a single path can be improved by adjusting the HP of input vector V. The usual choice of uniformly distributed random test vectors, where $H(V) = 0.5$, do not actually provide the best sensitivity to errors. Instead, a maximum sensitivity can be achieved when $H(V)$ is 0.33 or 0.67 depending on whether the rising or falling transitions fail first.

This unusual asymmetrical phenomenon can be explained and modelled probabilistically through the following cases. Consider 3 cycles of input vector sequences $V(k), k = 1, 2, 3$. If the falling transitions fail to propagate within 1 cycle, a transition is only detected at the output register on the 4th cycle when V has a sequence of $0 \rightarrow 0 \rightarrow 1$ or $1 \rightarrow 0 \rightarrow 0$.

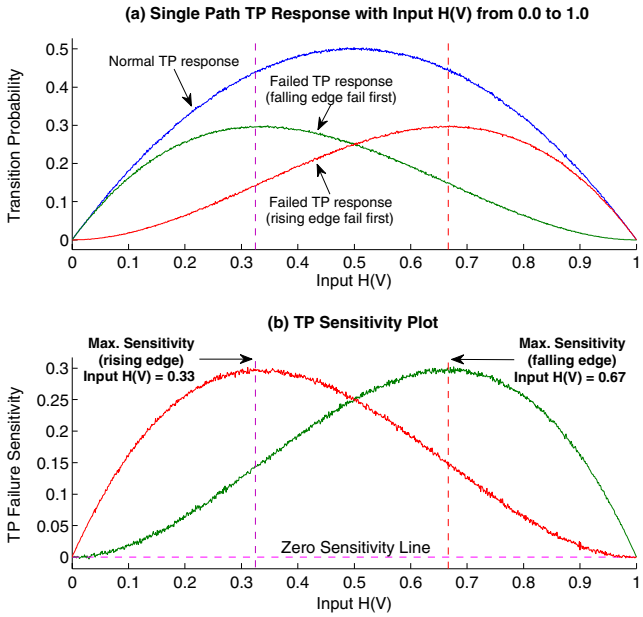

Figure 6: Plots evaluating the sensitivity of TP to timing failure in a circuit path. Maximum sensitivity is achieved when the input vector V has high probability $H(V) = 0.67$ when falling transitions fail first, or $H(V) = 0.33$ when rising transitions fail first.

Therefore, the output TP of the failed path in terms of V is given by the probability of the two sequences occurring:

$$\begin{aligned} TP_{\text{fall\_failed}} &= 2 \times P(V=1) \times P(V=0) \times P(V=0) \\ &= 2 \times P(V=1) \times (1 - P(V=1))^2 \\ &= 2H(V)(1 - H(V))^2 \end{aligned} \quad (7)$$

In the same way, when the rising transitions fail, a transition is only detected when V is $0 \rightarrow 1 \rightarrow 1$ or $1 \rightarrow 1 \rightarrow 0$. This produce a similar probability expression:

$$\begin{aligned} TP_{\text{rise\_failed}} &= 2 \times P(V=1) \times P(V=1) \times P(V=0) \\ &= 2 \times P(V=1)^2 \times (1 - P(V=1)) \\ &= 2H(V)^2(1 - H(V)) \end{aligned} \quad (8)$$

By subtracting these failed TP responses from the normal TP response (TP_{normal}) which is given earlier by (3), the TP sensitivity of both falling and rising transitions can be derived:

$$\begin{aligned} \text{Sensitivity}_{\text{fall}} &= TP_{\text{normal}} - TP_{\text{fall\_failed}} \\ &= 2H(V)^2(1 - H(V)) \end{aligned} \quad (9)$$

$$\text{Sensitivity}_{\text{rise}} = 2H(V)(1 - H(V))^2 \quad (10)$$

These expressions describe exactly the sensitivity behaviour observed in Fig. 6 and the HPs corresponding to their maximums (peaks) computed through solving their derivatives, giving exactly the observed optimal HP values: 0.33 (1/3) and 0.67 (2/3) for rising and falling transitions respectively.

This asymmetrical sensitivity to different transition types means that uniformly distributed random vectors is not necessary the optimal choice, given the CUT is known to have one type of transitions failing at a significantly lower clock frequency than the other. Such behaviour is common in CMOS circuits where the pull-up and pull-down transistors are sized differently or when extra pull-up or down transistors are added to improve signal strength. The only advantage of uniformly distributed random vectors is when the CUT has exactly matched rising and falling transition delay or their failure order is not known in advance.

3.1.4 *TP Response and Sensitivity Mapping of Logic Circuits*

To further understand how varying the input HP can improve the cases with potential sensitivity loss – sensitivity blocking and failure blind spot, we carried out a series of sensitivity simulation on a 2-input logic block. The layout of the block is depicted in Fig. 7, where it has two internal paths, each with its corresponding rising and falling transitions delays ($t_{\text{A-fall}}$, $t_{\text{A-rise}}$ and $t_{\text{B-fall}}$, $t_{\text{B-rise}}$). The idea is to stimulate both inputs of the circuit with random vectors A and B of varying $H(A)$ and $H(B)$ to create extensive two-dimensional mappings of TP response and sensitivity, and identify possible sensitivity issues.

The first issue we encountered is sensitivity blocking, which occurs in circuits with multiple pipeline stages. Fig. 8 demonstrates how certain failure response from the preceding logic stage could be blocked by simple logic functions. For a circuit with multiple pipeline stages, it is important to have the TP response caused by failure of early stages to propagate all the way through to the output, so that it can be detected. This process can, however, be blocked by logic stages, if the input statistics $H(A)$ and $H(B)$ change in a specific way that follow the contour lines in the TP response maps. Each of the lines represents a constant level of output TP. Thus, $H(A)$ and $H(B)$ changing along these lines would yield no output TP change, effectively blocking any timing failure response from reaching the output. In this case an XOR gate posses the most problem, because it has a large flat region at the centre where variation of $H(A)$ and $H(B)$ would not produce any change at the output. The obvious solution against this problem is to adjust the input HPs such that the observed TP response blocking does not happen.

Another serious issue with TP sensitivity is when a timing failure in a circuit leads to no change of TP with specific input HPs – the failure blind spot. Such cases could be demonstrated in 2-input functions and they are depicted in Fig. 9. In the three cases of 2-input functions: AND, OR and XOR, the falling transition delay from input A is set to have the worst-case delay and hence it fails first in the simulation. The deviation of TP caused by A failing is recorded for all possible input HPs of A and B to form a sensitivity map for each case. The level where sensitivity is zero is marked by contour lines. Therefore, any $H(A)$ and $H(B)$ values that fall on or near these lines will result in undetectable TP response. Clearly, for AND and OR function, the blind spots with zero sensitivity are rare and can be avoided relatively easily. On the other hand, XOR has a wide spread region across the middle where $H(A) = 0.5$. Such region should be avoided by using different values for $H(A)$ and $H(B)$. For linked input HP values where $H(A) = H(B)$, $H(A) = H(B) = 0.87$ gives approximately the best sensitivity. It can also be seen from the sensitivity maps that when $H(A)$ is 1, 0, and 1 or 0 for the AND, OR and XOR cases respectively, the optimal sensitivity is achieved at $H(B)$ predicted by (9) where falling transitions are assumed to be slower.

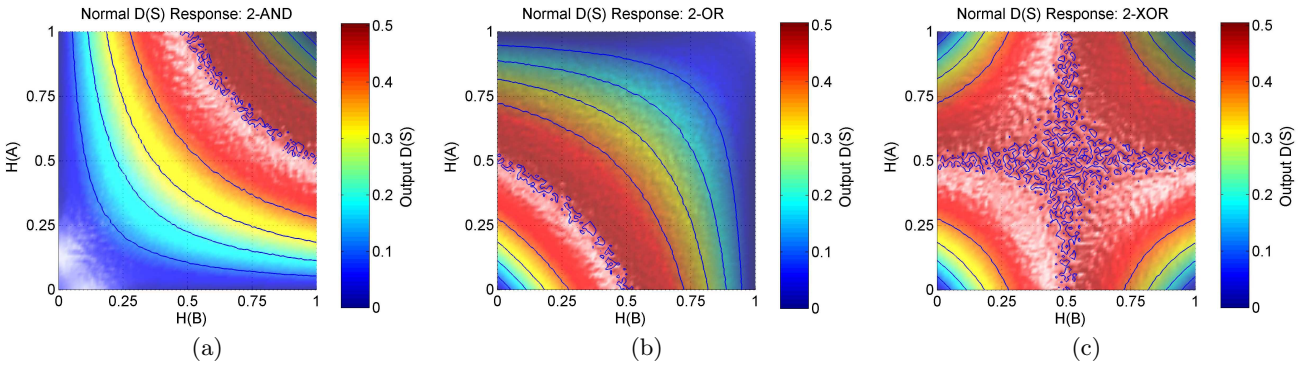

Figure 8: The output TP response mapping of all possible input HPs for (a) AND gate, (b) OR gate and (c) XOR gate. The contour lines on the maps represent levels of the same output TP, hence any change of input $H(A)$ and $H(B)$ along the contour lines leads to an unchanging output $D(S)$, possibly blocking failure responses from the previous logic stages.

The effect of the XOR's blind spot is demonstrated in Fig. 10 and Fig. 11 in terms TP profile, where cases with different path delay order is shown. In Fig. 10, the error sensitivity is completely lost, due to the uniform input HPs. Whereas in Fig. 11, the TP profile showed a certain change when $t_{\text{B-fall}}$ is violated, but still missed the failure of the worst-case path ($t_{\text{A-fall}}$). In both case, the sensitivity is restored and improved dramatically when $H(A) = H(B) = 0.87$ is used.

3.2 Self-Optimising Complex Circuit Test Platform

The complete complex circuit test platform is depicted by Fig. 12. The test circuit automatically optimise its random input vectors with specific probability weights to improve the TP's sensitivity against timing errors in the CUT.

3.2.1 *Adaptive Input Probability Weighting*

The *circuit response tester* (CRT) stimulates the CUT by toggling one input bit at a time while cycling the remaining bits with a counter every two clock cycles. Each count would form a pair of input patterns differ only by the single toggle bit. This forms a set of exhaustive *single input change* (SIC) test vectors. This approach effectively exercise every path in the combinatorial logic blocks in the CUT with full input access. The Output pattern is analysed by the *circuit response checker* (CRC). Input pattern pairs from the CRT that leads to actual activities at specific output bit are recorded and marked as "effective". Since a significant number of input patterns are likely to produce no output transitions, the refined "effective" input patterns would form a vector series with distinctive average HP values for each input bit when applied in sequence. Such HP values are then applied to the *probability weighting circuit* as HP weights to generate weighted random sequences with specific HP. The HP optimised random vectors are likely to exercise the internal paths of the CUT more thoroughly than the uniformly distributed random vectors, because it is probabilistically similar to the "effective" input patterns that exercised every paths in the exposed combinatorial parts of the CUT.

For the earlier 2-input XOR example, the effective input vectors are: $*0$, $0*$, $*1$ and $1*$, where $*$ represents the input bit being toggled by the CRT. Assuming a toggling bit is

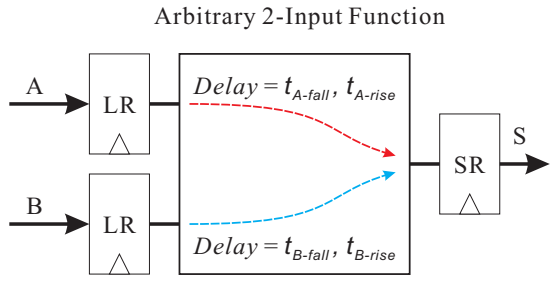

Figure 7: A simple two input arbitrary functional block for testing the sensitivity of transition probability to timing failure with multiple signal paths.

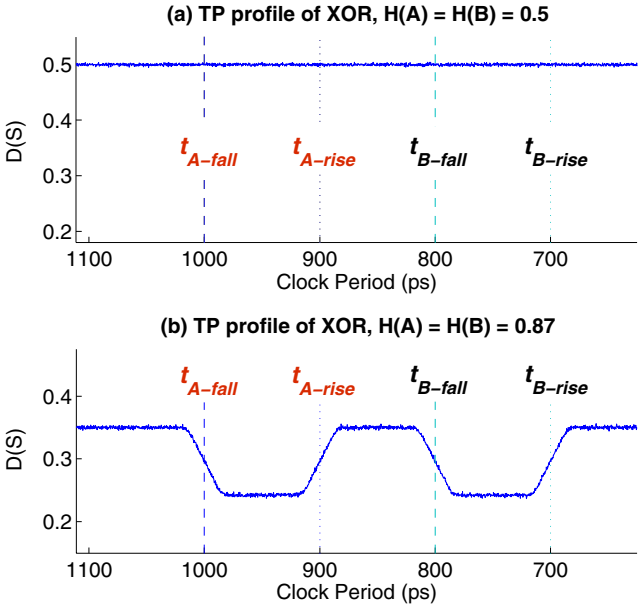

Figure 10: Simulated TP profile of an XOR gate showing (a) sensitivity loss to timing failure in all paths when using uniformly distributed random inputs, and (b) sensitivity restored using $H(A) = H(B) = 0.87$.

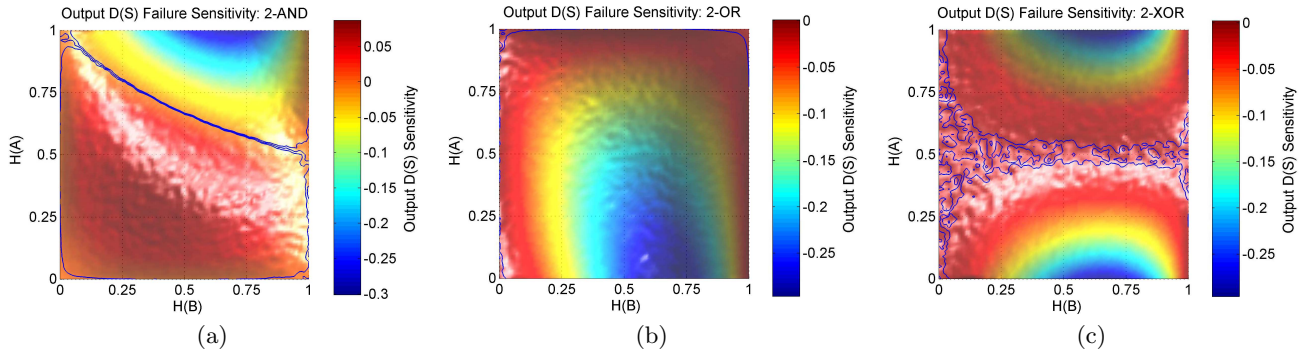

Figure 9: The TP failure sensitivity mapping of all possible input HP for (a) AND gate, (b) OR gate and (c) XOR gate. The contour lines represents the level at which sensitivity is zero. Both positive and negative sensitivity values represent a measurable change of TP, but in different directions.

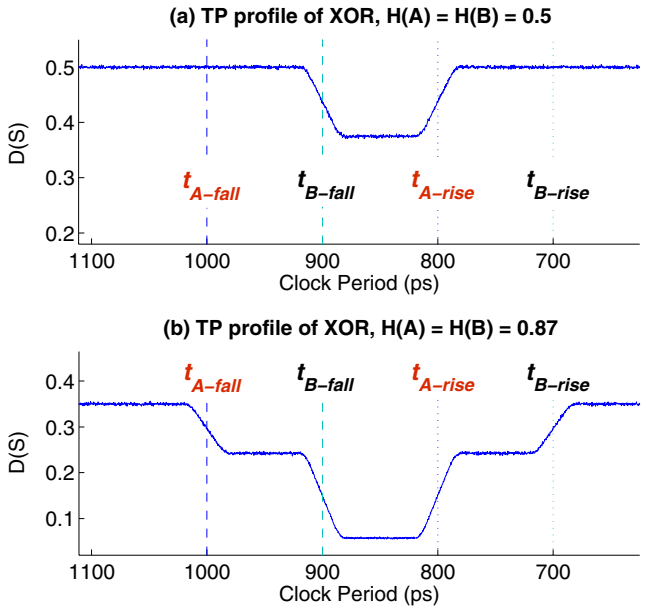

Figure 11: (a) shows the sensitivity loss to failure of the slowest type of transitions in the XOR and regain sensitivity when both type of transitions have failed. (b) shows the regained sensitivity using $H(A) = H(B) = 0.87$.

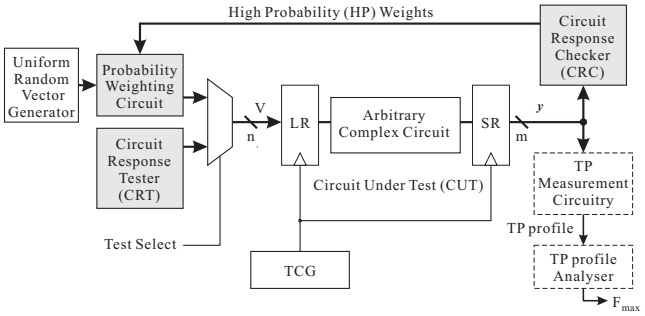

Figure 12: Block diagram of the self-optimising complex circuit test platform.

assigned an HP of 0.5, one could argue that the average HP of the four vector pairs equates to 0.5 for both input bits, which is shown to give zero sensitivity. However, due to the asymmetrical sensitivity of delay paths described earlier by (9) and (10), the overall average of the effective vectors may not form appropriate HP weights. Instead, the vectors should be divided into groups according to the pattern of the non-toggling bits and compute their average HP weights separately. In this case, they form two groups: $\{*0, 0*\}$ and $\{*1, 1*\}$. Assuming falling transitions are slower, optimal sensitivity is achieved when the $*$ bits are assigned an HP of 0.66, and the resultant HP weight of both input bits for the 1st and 2nd group are $(0.66 + 0)/2 = 0.33$ and $(0.66 + 1)/2 = 0.83$ respectively. Given that HP weights greater than 0.5 favour higher failure sensitivity of the slower falling transitions in this case, the HP weight from the 2nd group (0.83) closest to the optimal HP weights of 0.87 shown earlier should be used. For circuits with unknown internal structure, the HP weight pattern from each vectors group can be applied and tested separately for maximum accuracy. The test time would increase but not in multiples of a single test, because the test clock frequency range would be reduced considerably after the first few HP weight patterns.

While this approach may appear to neglect sequential feedbacks in circuits, where combinatorial blocks with feedback inputs may not be directly controllable from the proposed input sequences; it is the very nature of feedback in sequential circuits that allows the TP test method to maintain high timing error sensitivity, where errors are accumulated through the feedback paths and cause a significant change in the output TP response.

3.2.2 *Generating Weighted Probability Test Vectors*

Weighted random sequences can be generated easily by combining several independent uniformly distributed random bit streams together with simple boolean logic [15]. Table 1 shows an example of 9 levels HP weighting using three independent random bit streams: $R0$, $R1$ and $R2$. For FPGAs with dynamic LUT mask reconfigurability, the HP weight can be modified easily through changing the LUT's function on the fly. Otherwise, the same controllable HP can be implemented with several LUTs at the expense of slightly more area. For the Cyclone III EP3C25 without dynamic reconfigurable LUTs, a weighted random bit stream with 17 HP levels requires three 4-input LUTs to implement.

Table 1: Example of weighted random generation logics with different high probability (HP).

Weights	HP	Logic Expression
1	0	GND
2	0.125	$R2 \cdot R1 \cdot R0$
3	0.25	$R2 \cdot R1$
4	0.375	$R2 \cdot (R1 + R0)$
5	0.5	$R2$
6	0.625	$R2 + R1 \cdot R0$
7	0.75	$R2 + R1$
8	0.875	$R2 + R1 + R0$
9	1.0	VCC

4. TEST PLATFORM IMPLEMENTATION AND EVALUATION ON FPGA

The proposed complex circuit test platform is implemented on the Cyclone III EP3C25 FPGA to evaluate its accuracy and efficiency. Fig. 13 depicts the hardware layout of the test circuit on the FPGA. In this particular case, the TP measurement circuitries and CRC are placed next to the CUT for a more compact representation. However, there are no limitation on where these circuitries should be placed, because they are completely asynchronous from the CUT and do not suffer from timing issues if placed at a remote location. The random vector generator is implemented as an LFSR and is followed by a 17 levels HP weighting circuit.

The test procedure contains two phases. First the circuit's response is analysed by the CRT and CRC to generate the optimised HP weights, then they are used to conduct the TP test to obtain its maximum operating frequency or worst-case delay measurements. The response analysis phase is only required once for each design. In some cases, it can be skipped completely if the optimised HP weights can be obtained through analysis or simulation of the CUT during the design process. Results consistency are ensured through repeated tests until the FPGA's temperature stabilises.

The test platform is evaluated by two types of CUTs: A 4x4 LUT based multiplier and a Butterworth IIR Filter. The layout in Fig. 13 is taken from the Butterworth Filter case. The test candidates were chosen such that both combinatorial and sequential circuits are evaluated. Since practical FPGA applications in general contain mostly LUT based functions, the LUT based multiplier test would give us a clear guideline on how well the test method performs in general.

4.1 Multiplier Test Case

The LUT based multiplier is tested with both the proposed TP method and a full exhaustive test method proposed in [17] to give an absolute measurement reference for accuracy comparison. For the TP test, the random inputs with and without optimised HP weighting are tested to identify their effectiveness. The placement and routing of the CUT are kept exactly identical between both tests.

4.1.1 Results

The measured maximum operating frequencies of the multiplier are shown in Fig. 14. The results with optimised input HP tracks the exhaustive test results very closely and is accurate within 1% of the results. The apparent accuracy difference between the normal and optimised HP results are not very high in this case because the test using uniformly

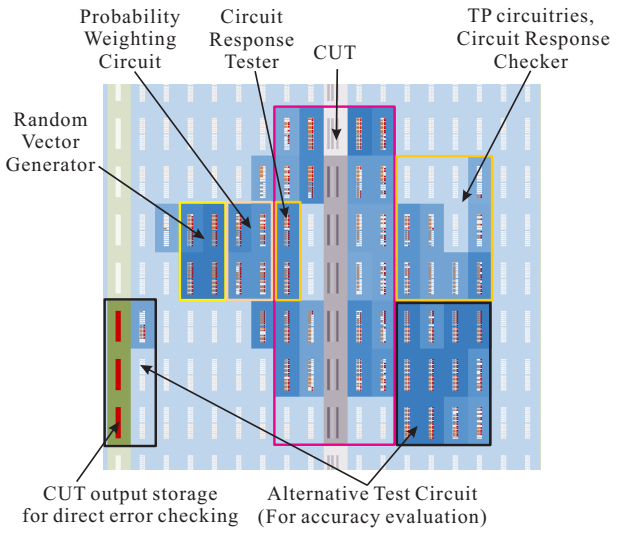

Figure 13: Layout of the hardware test platform on a Cyclone III EP3C25 for complex CUT. An alternative test circuit is included for accuracy evaluation of the TP test platform.

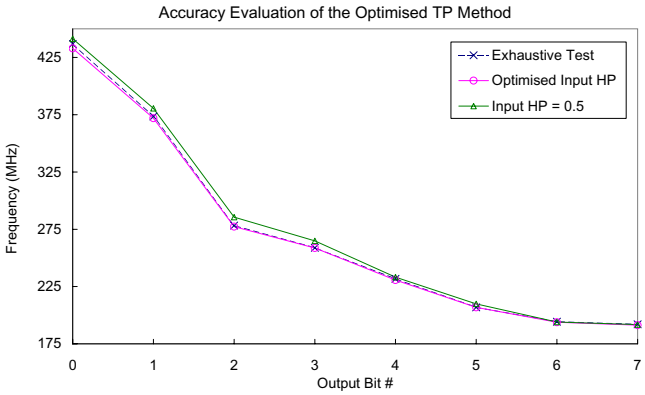

Figure 14: Accuracy evaluation of the TP method on a 4x4 LUT based multiplier with optimised input HP against an exhaustive test method proposed in [17].

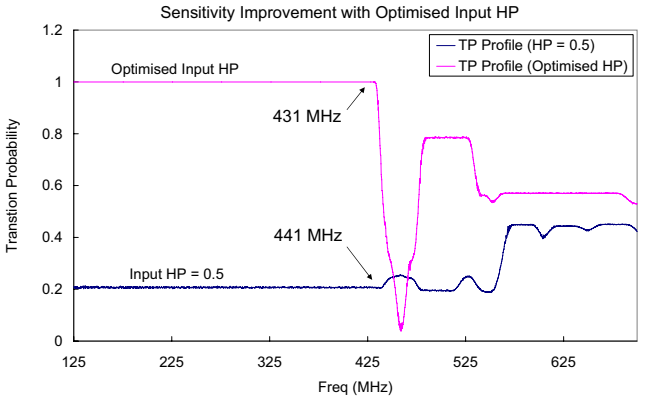

Figure 15: Plot showing the error sensitivity improvement of the TP profile of a single multiplier output bit using optimised input HP.

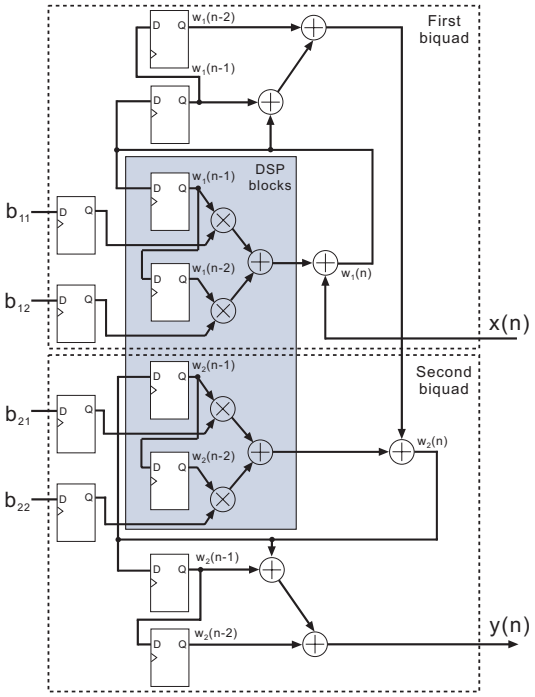

Figure 16: A 4th order Butterworth IIR filter design from Altera [1], where x(n) is the input and y(n) is the output.

distributed random inputs is already very close to the exhaustive test references. Nonetheless, a clear improvement can still be seen between the two. We suspect that the good accuracy from the uniform random inputs is due to a high degree of glitch acitivity at the combinatorial output of the multiplier. Referring back to Section 3.1.2 and Fig. 5, it can be seen that when the glitch period is violated by the clock edge and jitter region, the registered output becomes highly unpredictable. Although it is not possible to predict the exact TP response, the increased uncertainty may have caused a more distinctive TP deviation from its normal value and thus increased the TP sensitivity.

In Fig. 15, the TP profiles taken from the same output bit using uniform HP and optimised HP are compared. The TP profile with optimised HP shows a significantly higher timing error sensitivity – a larger deviation in TP response and more accurate measurement using simple TP thresholds.

The optimised TP method may produce slightly more conservative results than the exhaustive test because it is not possible to take the effect of clock jitter into account with the complex TP behaviour produced by multiple paths failing. Whereas the exhaustive test method examine each path individually and is able to produce a nominal $f_{\max}$ according to the expected clock edge position at the centre of the jitter distribution. Please refer to [17] for more information.

The test time of the multiplier is under 3 seconds, assuming the optimal input HP weights are extracted in advance or pre-computed from the functional model of the multiplier.

4.2 Butterworth IIR Filter Test Case

The Butterworth IIR Filter (Fig. 16) is implemented with multiple 18x18 embedded multipliers, adders, feedback paths and register stages on the Cyclone III. Such complexity of

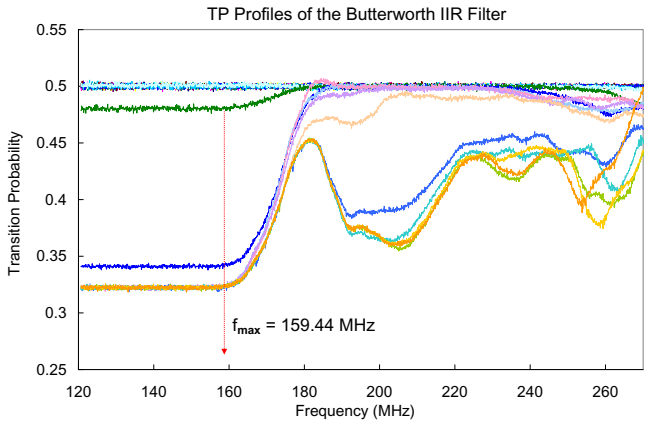

Figure 17: The TP profiles of all 21 output bits of the Butterworth IIR filter.

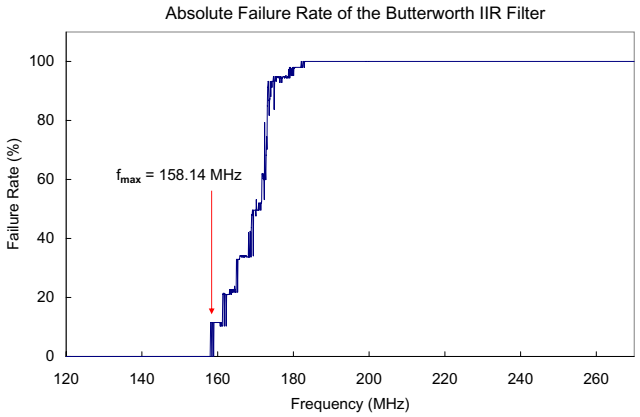

Figure 18: The absolute failure rate of the Butterworth IIR Filter between 120 **and** 270**MHz.**

the CUT resembles most practical designs in FPGAs and it would give a good representation of the TP test platform performance in terms of accuracy in realistic situation.

To evaluate the measurement accuracy, an absolute comparison based test method that basically gathers the filter outputs at a series of finely spaced clock frequencies steps and compare them against a set of pre-calculated reference results to identify any timing errors. The layout of the extra test circuit is depicted in Fig. 13 as "Alternative Test Circuit". Note that this alternative test circuit is built purely for the purpose of accuracy evaluation, its area overhead and test time are far too high for practical use.

4.2.1 Results

Results from the optimised TP test platform in the form of TP profiles (Fig. 17) gave a maximum operating frequency ($f_{\max}$) measurement of 159.44MHz, which is within 1% of the absolute $f_{\max}$ obtained from the comparison based method (Fig. 18). This reference $f_{\max}$ is derived from the point where error starts to occur in the failure rate plot.

The test time in this case is similar to the previous multiplier test case, where a test takes approximately 3 seconds to complete. This is mainly because the test time is linked to the range of frequency sweep and a relatively short frequency range was needed to obtain the results in both cases.

5. CONCLUSIONS

In this paper, we have shown that the proposed optimised TP test method could provide a highly accurate delay and frequency measurements in both complex combinatorial and sequential circuits. Providing accuracy within 1% of the absolute measurements from the much more time consuming and area expensive full exhaustive and direct comparison reference methods. Effects of environmental variations were minimised by placing the test circuits together and repeating the test process until temperature stabilises. The proposed technique to optimise random input test vectors in terms of high probability weights has enabled a large variety of complex circuits to benefit from the elegant TP test method with highly accurate and reliable timing results. Moreover, the test circuit is highly area efficient, where overhead is not directly proportional to the CUT's complexity but the number of input and output bits, and it is contributed mainly to input vectors generation. The TP circuitries can also be shared among different outputs or circuits to achieve further area reduction at the cost of longer test time. Otherwise, multiple TP counters could be used in parallel for very short test time.

The main limitation of the test method is that the actual response in a TP profile cannot be reliably predicted for complex circuits due to glitches and clock jitter uncertainties. That means there could be a certain degree of unpredictability in the measurement's accuracy. However, given the achieved accuracy in the test cases, such unpredictability could be easily guarded using a relatively small guard band and have minimal impact on the results optimality. Also, as future work, memory oriented designs as well as a wider variety of circuits should be tested to explored and improve the effectiveness and accuracy of the measurement method to further reinforce its general usability.

The generalised test modules and the flexibility on placement location of the TP measurement circuitries allow FPGA users to easily apply the test platform to their circuit designs for accurate and efficient physical delay measurements. Such test platform could potentially be integrated into conventional FPGA design flow, to give users an immediate knowledge of their circuit's timing performance under the actual FPGA hardware and physical conditions. Timing models in existing FPGA design tools are usually made to be highly conservative to account for process, temperature and voltage variations (PVT) as well as possible delay degradation over the FPGA's life. This often leads user to under rate their designs' operating speed and wastes a significant amount of potential performance. With the proposed test platform as a quick physical timing analysis tool, such problems could be mitigated and greatly increase the productivity of FPGAs.

6. REFERENCES

[1] Altera Corp. *Implementing High Performance DSP Functions in Stratix & Stratix GX Devices*, 2004.

[2] Altera Corp. *Understanding Metastability in FPGAs*, 2009.

[3] L. Cheng, J. Xiong, L. He, and M. Hutton. FPGA performance optimization via chipwise placement considering process variations. In *Proc. International Conference on Field Programmable Logic and Applications (FPL)*, pages 44 – 49, Aug. 2006.

[4] K. Katoh, T. Tanabe, H. Zahidul, K. Namba, and H. Ito. A delay measurement technique using signature registers. In *Proc. 18th Asian Test Symposium (ATS 2009)*, pages 157 – 162, Nov. 2009.

[5] K. Katsuki, M. Kotani, K. Kobayashi, and H. Onodera. A yield and speed enhancement scheme under within-die variations on 90nm LUT array. In *Proc. IEEE Custom Integrated Circuits Conference*, pages 601 – 604, Sept. 2005.

[6] K. Kundert. *Predicting the Phase Noise and Jitter of PLL-Based Frequency Synthesizers.* Designer's Guide Consulting Inc., 4g edition, Aug. 2006.

[7] A. Maiti and P. Schaumont. Improving the quality of a physical unclonable function using configurable ring oscillators. In *Proc. 19th International Conference on Field Programmable Logic and Applications (FPL)*, pages 703 – 707, Aug. 2009.

[8] T. Matsumoto. High-resolution on-chip propagation delay detector for measuring within-chip variation. In *International Conference on Integrated Circuit Design and Technology*, pages 217 – 220, May 2005.

[9] Y. Matsumoto, M. Hioki, T. Kawanami, T. Tsutsumi, T. Nakagawa, T. Sekigawa, and H. Koike. Performance and yield enhancement of FPGAs with within-die variation using multiple configurations. In *Proc. ACM/SIGDA International Symposium on Field Programmable Gate Arrays - FPGA*, pages 169 – 177, Feb. 2007.

[10] S. Pei, H. Li, and X. Li. A low overhead on-chip path delay measurement circuit. In *Proc. 18th Asian Test Symposium (ATS 2009)*, pages 145 – 150, Nov. 2009.

[11] A. Raychowdhury, S. Ghosh, and K. Roy. A novel on-chip delay measurement hardware for efficient speed-binning. In *Proceedings - 11th IEEE International On-Line Testing Symposium, IOLTS 2005*, pages 287 – 292, Jul. 2005.

[12] M. Ruffoni and A. Bogliolo. Direct measures of path delays on commercial FPGA chips. In *Proceedings - 6th IEEE Workshop on Signal Propagation on Interconnects, SPI*, pages 157 – 159, May 2002.

[13] P. Sedcole and P. Y. K. Cheung. Parametric yield modelling and simulations of FPGA circuits considering within-die delay variations. *ACM Transactions on Reconfigurable Technology and Systems*, 1(2), 2008.

[14] E. A. Stott, J. S. J. Wong, P. Sedcole, and P. Y. Cheung. Degradation in FPGAs: Measurement and modelling. In *Proc. ACM/SIGDA International Symposium on Field Programmable Gate Arrays - FPGA*, pages 229 – 238, Feb. 2010.

[15] L.-T. Wang, C.-W. Wu, C.-W. Wu, and X. Wen. *VLSI test principles and architectures: design for testability.* The Morgan Kaufmann series in systems on silicon. Academic Press, 2006.

[16] J. S. J. Wong, P. Sedcole, and P. Y. K. Cheung. A Transition Probability based delay measurement method for arbitrary circuits on FPGAs. In *Proc. IEEE International Conference on Field-Programmable Technology*, pages 105 – 112, Dec. 2008.

[17] J. S. J. Wong, P. Sedcole, and P. Y. K. Cheung. Self-measurement of combinatorial circuit delays in FPGAs. *ACM Transactions on Reconfigurable Technology and Systems (TRETS)*, 2(2):1 – 22, 2009.