

On-Chip structures for Timing Measurement and Test

D.J.Kinniment, O.V.Maevsky, A.Bystrov, G.Russell, and A.V.Yakovlev

Newcastle University, UK

David.Kinniment@ncl.ac.uk, Oleg.Maevsky@ncl.ac.uk, A.Bystrov@ncl.ac.uk,
Gordon.Russell@ncl.ac.uk, Alex.Yakovlev@ncl.ac.uk

Abstract

This paper describes the use of digitally set delay lines in conjunction with MUTEX time comparison circuits, to measure on-chip signal path timing differences to accuracies of better than 10ps. Three methods of time measurement are described. The first, which uses parallel MUTEXs with a tapped delay line, is analogous to a flash A/D converter. The second one is similar to a successive approximation method. Both are fast, and efficient, but the second requires less hardware for a large number of bits. The third technique uses a MUTEX to amplify small time differences to a measurable size.

Applications for these techniques include adaptive synchronization and input tests, such as data set-up time conditions that currently require the use of very expensive test hardware. We describe an on-chip method of testing these conditions, using uncorrelated signals whose statistics are known, and accurately selecting the conditions to be tested on-chip.

1 Introduction

With the reduction in dimensions and consequent increasing speed of digital circuits, sub-micron devices show increasing variability of delays. Parametric changes, as well as spot defects, may cause the dynamic behavior of a circuit to change, resulting in timing problems. Fully asynchronous, and delay insensitive methods can be used to ensure functional correctness, but do not always produce high performance, and fully synchronous or globally asynchronous, locally synchronous systems (GALS) which rely on fixed clocks and use pre-existing libraries of synchronous cells are often preferred. While these systems can be fully tested functionally by adding test structures on chip, timing issues remain a problem, particularly the timing margins available when working at full speed, and the need to ensure set-up and hold times for the input output interfaces. The first problem requires a tester that is able to vary the clock frequencies by small increments and supply data patterns at full speed, and the second problem also requires the tester to adjust data and clock to very accurate absolute timing margins, perhaps as low as 20ps.

1.1 Timing test

All systems on chip must be interfaced to off-chip inputs and outputs, and consequently these interfaces must be tested to absolute timing margins for parameters such as set-up and hold times. Testing I/O parameters at high-speed and/or accuracies of better than 100ps involves very high-performance automatic test equipment, and the cost of such testers is becoming prohibitive (\$600k for a development tester; \$4M for a production tester). Current trends in on chip complexity and internal bandwidth, compared to the bandwidth available for external test suggest that the ratio of internal to external bandwidth will increase by over an order of magnitude from 5.4:1 in 2002 to 94:1 in 2011 [1], making the use of built in test methods essential. Current trends are to narrow the difference between the period of the device under test, and the timing accuracy of the signals from any external tester. This problem could lead to an unacceptable reduction in device yield, since uncertainty in test accuracy leads to device rejection. The ratio between device period and tester accuracy, currently around 5:1 in 2002, is likely to move towards 2.5:1 in 2011, because it is difficult to time external inputs to much better than 100ps, a physical distance of less than 30mm [1].

What is required is a cost-effective methodology that provides on chip structures capable of verifying timing conditions and measuring timing margins.

1.2 Adaptive timing

Techniques have been developed in the context of asynchronous systems design, where the system timing is provided by the circuits themselves, rather than by an external clock. In such systems a completion signal may indicate the end of a computation, and this signal is used to latch the resulting data into a register.

If, in a synchronous environment, a timing margin could be established between the completion of the computation, and the clocking of the following register, the optimum clock frequency or data delay could be determined. Dynamic clock adjustment is currently being investigated to provide pausable or adaptive clocks in GALS [2], but this may produce adjustment loops in which the clock timing in one region affects the next, and this is eventually fed back to the first region.

An alternative is to adjust the timing of the data by detecting when data and clock signals conflict, and to increase data delays where this occurs, possibly also decreasing the delay when conflict does not occur in order to improve latency, [3]. In a GALs system this is known as adaptive synchronization, and may lead to a lower probability of synchronization failure.

Current methods proposed for adaptive synchronization are based on simple inverter chain delays controlled by a counter and produce relatively large changes in the data delay path if there is a conflict, but if the delay steps are much smaller the system may not be able to track high rates of clock drift.

Here the ability to make either small or large changes within one clock cycle would be an advantage, since the data latency could then be kept to a minimum at the same time as reducing the probability of failure. Figure 1 shows an adaptive synchronization scheme in which the relative timing between the clock and the data is measured to an accuracy of a few picoseconds, much less than any potential drift, and the result expressed in a binary form. If the previous value of time difference is held in a register, a more sophisticated control algorithm might then be able to track the clock drift more efficiently.

Prerequisites to producing a more reliable adaptive synchronizer are:

1. More accurate delay calibration and adjustment,
2. Delay measurement to a few picoseconds and within one clock cycle.

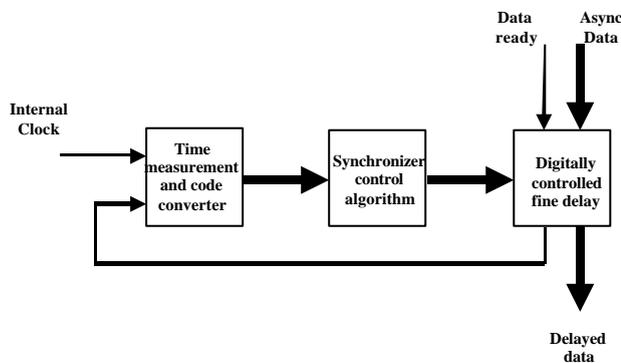


Figure 1 Adaptive synchronizer

We aim to show how these objectives may be achieved on chip.

2 Timing measurement

Techniques which are useful in this context, are timing adjustment methods, such as adjustable delay lines made of inverter chains [2][3], and specialised circuits for indicating which of two signals appears first. The design of digitally adjustable delay lines is relatively straightforward, since delays can be made from a network of inverters that are switched to produce

delay paths alterable in increments two inverters as shown in Figure 2.

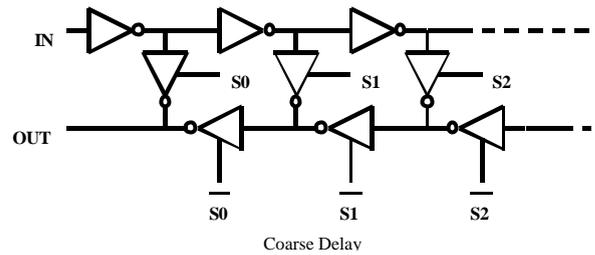


Figure 2 Inverter chain delay

Delay steps given by two inverters will not usually be sufficiently accurate for time measurements, but finer steps can be produced by adding capacitance to the node between two inverters as shown in Figure 3. Here the size of the capacitance is increased in roughly equal steps so that fifteen or more intermediate values of delay can be produced between $2t_d$ and $4t_d$, where t_d is the delay of a single inverter. We used SPICE modelling of an 0.6μ CMOS technology to show that delays between 200ps and 320ps in steps of approximately 8ps could be produced reasonably accurately. The delay increment of the inverter chain of Figure 2 was 120ps, so that a capacitor loaded inverter pair followed by a switched inverter chain will therefore produce delays from $2t_d$ upwards in steps of $0.125t_d$ to as the value of the clock period, and beyond if necessary.

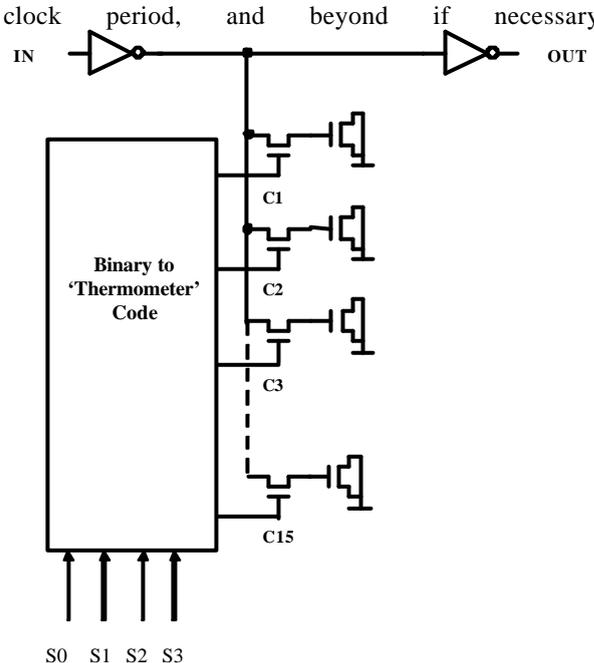


Figure 3 Fine increment delays

Because of variations in individual inverter pair delays may be up to 10ps, as shown by SPICE simulation, very accurate on-chip adjustment is more difficult, however, if very precise matching to a particular absolute time is needed smaller capacitors can enable adjustment to within 1ps with off chip calibration of the delay lines.

On-chip circuits have the advantage that they track typical signal path delays because they are affected in similar ways by process, temperature and power supply variations.

Absolute time measurement relies on off-chip calibration of the delay increments by frequency measurement over a relatively long period. Connecting the delay line as an oscillator as shown in Figure 4 allows the frequencies generated by two different settings of the delay to be measured accurately off-chip. The time difference in delay, d_t , is given by $d_t = (1/f_2 - 1/f_1)/2$, and absolute delays can also be inferred after an estimate for an individual inverter is made.

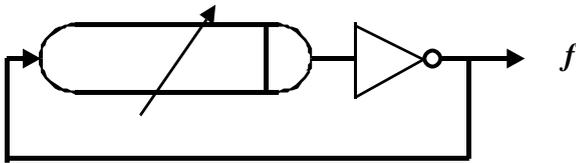


Figure 4 Delay calibration

Comparison of the timing between two signals can be done by means of a MUTEX circuit which indicates which of two requests arrive first [3][4]. A MUTEX circuit is shown in Figure 5 in which the assertion of an input signal is compared with a reference, after resolution of any metastability, the circuit indicates which was the first input to occur. Provided the MUTEX is symmetrical with respect to signal and reference, the accuracy of the result with respect to the time difference $\Delta_t = t_s - t_r$ between signal and reference is limited only by noise. Consequently, resolution to an accuracy of 0.1ps in a circuit with a MUTEX time constant, τ , of 100ps should be possible [5]. In other words, if the value of Δ_t is greater than this accuracy limit, the output “Signal first” (or “Reference first”) will respond after a finite metastability resolution time T_m . If, however, Δ_t is too small, then it can only be measured with some error E_Δ (affected by noise). In practice imperfections in fabrication may affect the transistor sizes, and power supply variations will also reduce the overall circuit accuracy. Results from simulations of MUTEX circuits in AMS 0.6 μ CMOS technology with up to 10% variation in transistor nominal size are shown in Table 1. If only one of the eight transistors has a size of 10% above nominal, the time offset is no more that 6.7ps, but the worst case distribution of sizes within the range 0-10% gives a maximum offset of 12.5ps. The probability of this worst-case situation occurring in a circuit with eight closely spaced transistors is fairly low.

The effect of transistor size variation on time error E_Δ can be reduced by introducing negative feedback in the flip-flop of the MUTEX, as shown in Figure 6. In this case the response time of the MUTEX increases

approximately by 3.5 times, however E_Δ is reduced approximately 25 times.

Table 1 MUTEX time errors

Signal Gate transistor widths, μ		Reference gate transistor widths μ				Time error		
Input A		Input B		Input A		Input B		E_Δ
p	n	p	n	p	n	p	n	ps
6.7	7.5	6.7	7.5	6.7	7.5	6.7	7.5	0
7.37	7.5	6.7	7.5	6.7	7.5	6.7	7.5	-2.9
6.7	8.25	6.7	7.5	6.7	7.5	6.7	7.5	+1.2
6.7	7.5	7.37	7.5	6.7	7.5	6.7	7.5	-1.6
6.7	7.5	6.7	8.25	6.7	7.5	6.7	7.5	+6.7
7.37	7.5	7.37	7.5	6.7	8.25	6.7	8.25	-12.5

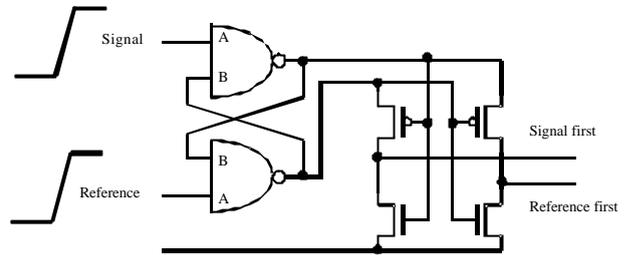


Figure 5 MUTEX time measurement

Furthermore, SPICE simulation showed that, in all experiments, the power supply variation up to 10% increased E_Δ approximately twice.

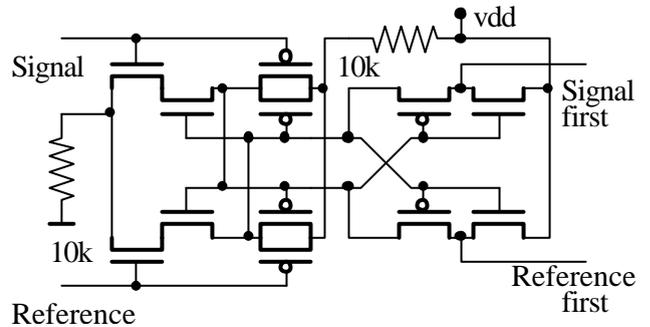


Figure 6 MUTEX with negative feedback

Using calibrated delays and timing comparisons it is possible to measure the delay between the assertions of two signals. These two signals could be the set up of data on an input bus, and the advent of the clock, or alternatively the final carry output from an addition operation, and the clocking of the result register. Times can be measured by delaying one signal by an accurately calibrated delay, and comparing its timing with the other using a MUTEX. Since MUTEXs are very simple circuits, it may be economic to compare the same signal with many different delays in parallel, thus measuring the time to a high degree of accuracy in one

pass as shown in Figure 12. Here the tapped delay is a simple chain of inverter pairs, and the outputs of the MUXES give a ‘thermometer code’ representation of the value of the delay, with ‘signal one first’ being asserted for all taps where the difference in timing between signal one and signal two, $t_{s_2} - t_{s_1} < 2nt_d$. This is easily converted to a binary representation if required. Time measurements more accurate than the delay of one inverter, t_d , can be made by using alternate MUXES circuits with NAND and NOR gates, and down to 8ps by successive measurements using the different settings of the fine increment delay.

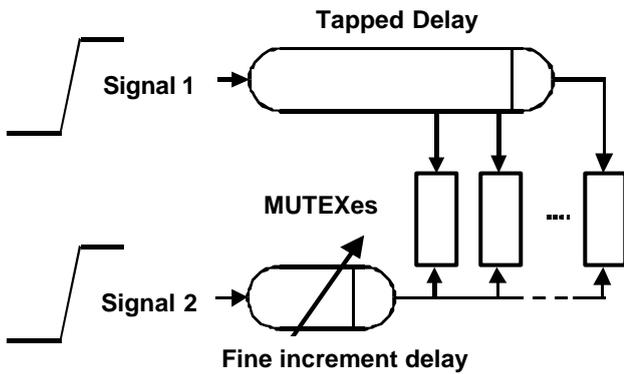


Figure 7 Time difference measurement

A more sophisticated method for measuring time difference uses a successive approximation conversion technique patented in [6]. Figure 8 shows a simplified cell for a single stage of conversion to a binary representation of time. Two input signals are applied to the inputs of a MUXES, and then delayed by the time T_m required for the MUXES to resolve metastability with sufficient high probability. The first of the inputs to go high is delayed by a further $2^k T$ and the other is not delayed. The value of the T_m delay is at least two inverters, so that it does not matter that fine increments of delay below two inverter delays cannot be obtained, the additional delay, $2^k T$, being the difference between two delays, can still be controlled to within about 8ps. The output of the MUXES represents the most significant (sign) bit of the delay difference. Subsequent cells delay one input by an additional $2^{k-1} T$, $2^{k-2} T$, etc., as shown in Figure 9. In each cell the additional delay is always added to the earliest signal, so that the outputs from the last cell will be closer in time than the minimum delay increment T . The result appears, at worst, $kT_m + (2^{k+1} - 1)T + (k - 1) \cdot t_{mpx}$ after the first signal, where t_{mpx} is the multiplexer time delay. The MUXES outputs will then provide a binary representation of the

time difference between the signals. This method requires significantly fewer MUXES circuits than that of Figure 7, since for k bit output only k MUXES are needed, while the previous method uses 2^k . The total time required for conversion from time difference to binary output is similar for the two methods, being approximately proportional to the maximum time difference expected.

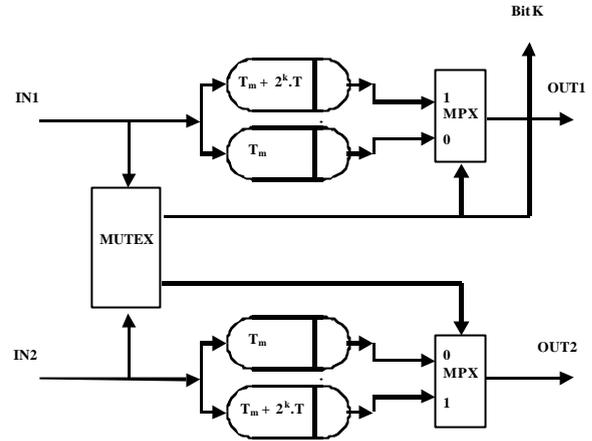


Figure 8 Successive approximation cell

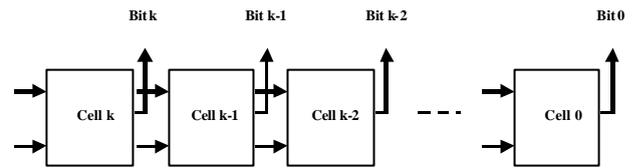


Figure 9 Successive approximation time measurement

3 Time amplifier

The measurement circuits described in the previous section are limited by the accuracy of the delay lines, and do not make full use of the potential of the MUXES circuit in comparing two signals. Potentially more accurate time measurements can be made by using the response time of the MUXES to effectively amplify the input time difference. The difference of the output voltages of a bistable in metastability is approximately given by: $\Delta_V = q \cdot \Delta_t \cdot e^{-t/\tau}$, where τ is the device time constant, q is the conversion factor from time to initial voltage at the metastable nodes, and Δ_t is the input time overlap between the two inputs [5]. (This formula is correct for $|\Delta_t| \in (0, |\Delta_V|/q]$, and provides a practically acceptable approximation for $|\Delta_t| \ll |\Delta_V|/q$.) Figure 12 shows the characteristics of a typical MUXES in 0.6 μ m technology, where it can be seen that the output time has a log relation to the input time difference. If we choose $q = 3$ mV/ps, and $\tau = 125$ ps, the theoretical curve given by $t = -\tau \ln(q \cdot \Delta_t / \Delta_V)$ is close to those obtained

by simulation of a MUTEX circuit for time differences below 30ps. The corresponding output times are greater than 300ps. This enables the output time to be measured by the variable delays, described in section 3 with increasing accuracy as the input time goes below 10ps (because of the log scale).

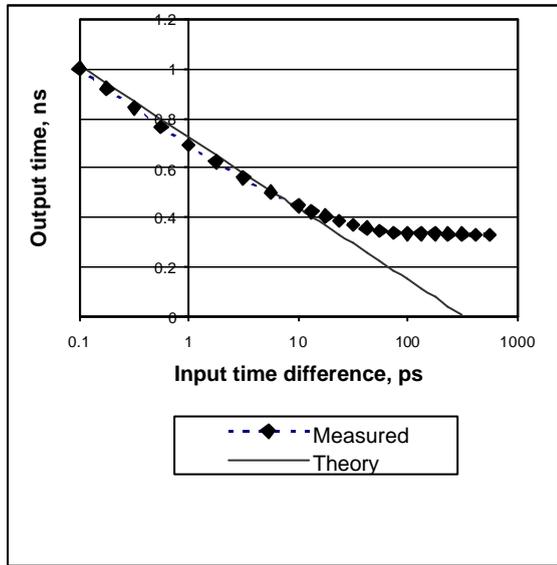


Figure 10 MUTEX characteristics

Figure 11 shows how this relationship can be used in a circuit which compares two rising inputs, and produces two outputs, the time difference between these two being proportional to $-\ln|\Delta_t|$. The accuracy of this circuit also depends on variation in the geometry of the MUTEX, but with careful layout it should be possible to achieve better than 5ps. Calibration can be done feeding known identical signals into both inputs, and measuring the metastability time and polarity of the result.

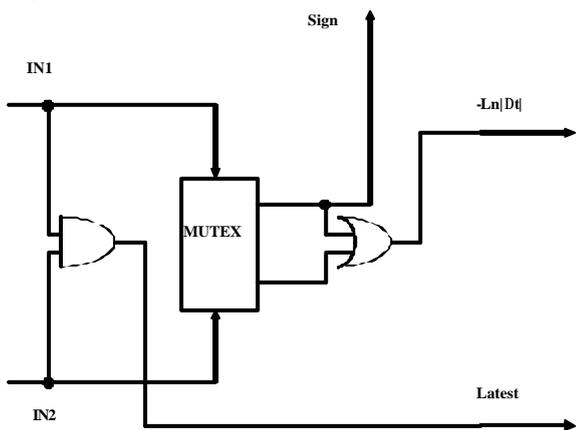


Figure 11 Time amplifier circuit

In Figure 11 a standard MUTEX circuit is used as a time amplifier, which produces response times in the range of between 1000ps (or longer, due to metastability) and 600ps for the input difference in the range between -2 ps and $+2$ ps. A MUTEX with a higher value of τ would allow a wider range of inputs up

to the delay of an inverter pair to be accurately measured. One of possible such circuits is shown in Figure 12. In this circuit the capacitor C can be used to increase the time constant of the MUTEX. The resistors R_p and R_n are used for reducing the error E_{Δ} to an acceptable value. For example, if $C=0$ and $R_p=R_n=50k\Omega$ then response times vary practically between 13.54ns and 12.79ns for input difference varying between -6 ps and $+6$ ps. In this case, variation of any transistor in the circuit increases E_{Δ} to 0.11ps. This results in the 50% error of measurement around the edge points -6 ps and $+6$ ps. Increasing the value of capacitor C helps increase the range of measurement by several times, but this method also increases the error E_{Δ} .

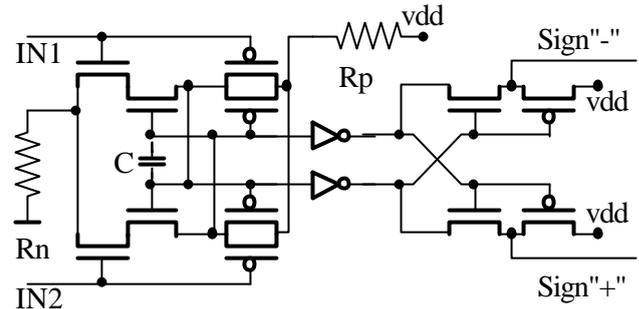


Figure 12 MUTEX with enlarged input range

4 Input timing tests

Checking of input pin set-up and hold times usually requires the provision of accurately timed signals from an external source. An alternative is to provide a wide range of input conditions from a relatively cheap source, such as two uncorrelated inputs from free running oscillators. Thus all possible timing relationships would be generated, and by selecting the timing relationship required from amongst many on chip, much greater accuracies are possible since the path distances can be kept short. Techniques like this are currently in use to measure the characteristics of synchronizers to the picosecond level [7], and the basic method is shown in Figure 13.

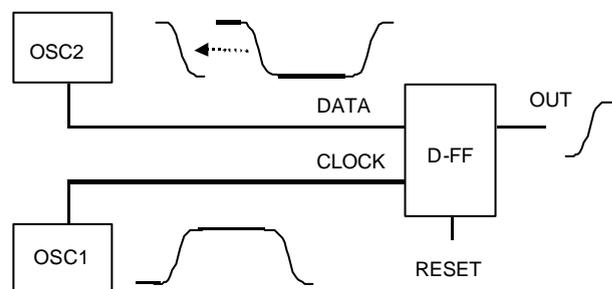


Figure 13 Two oscillator method

Here two independent oscillators provide inputs to clock and data inputs of a D flip flop at about the clock period, T_c , required for the system. When the clock

risers, the data input may be high, or low, and because the phase relationship between the two oscillators is slowly changing, the set-up time, where the data input is stable before the clock goes high will vary from 0 to $T_c / 2$, with the number of events being within Δ_t of any particular time given by $n = N \cdot \Delta_t \cdot 2 / T_c$, where N is the number of clock cycles over which the measurement is made. At a clock frequency of 1GHz, only 500 cycles (500ns) are required to ensure that at least one input condition within 1ps of that required is produced. Since the number of occurrences of a particular event is proportional to Δ_t at the inputs, it is possible to plot a histogram of the device propagation delay as a function of Δ_t in this test as shown in Figure 14.

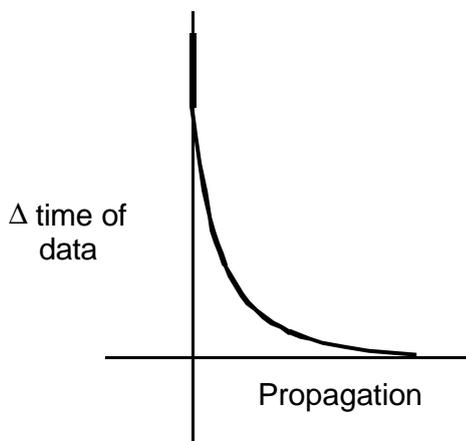


Figure 14 Event histogram

To ensure the accuracy required in a test of set up conditions, we must only look at flip flop responses that are produced relatively rarely, and this is done by selecting only those events which cause metastability to last longer than a given value. As demonstrated above, it may only require about 500ns getting accuracies of better than 1ps. In order to select a particular value for the set-up time test we select all of those events for which the data is asserted more than the set-up time before the clock. Figure 15 shows how this is achieved.

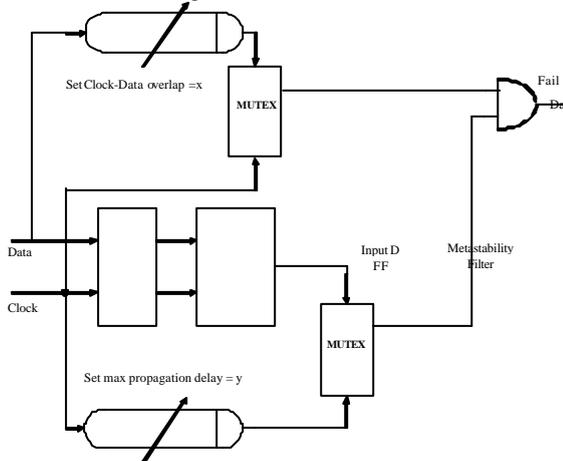


Figure 15 Input timing test

Here a variable delay line is set to measure the required Clock-Data overlap, and only those conditions in which the data is first by at least x ps are selected. The state of the input D Flip-Flop is monitored by a simple metastability filter which gives a high output when metastability is resolved, thus if metastability lasts for longer than y ns after the clock, we indicate a fail condition.

We simulated this circuit using SPICE, and showed that, for example, a value of 440ps set up on the x delay line selected inputs for which data and clock overlapped by 440ps or more. With an overlap of 440ps, metastability in the input flip-flop lasted for 2910ps. For a value of less than this in the y delay line a fail output would result.

In this diagram only one input data bit is shown, but the principle could easily be extended to large pin count devices where multiple inputs are associated with a single clock. In this case each input flip-flop would have its own metastability filter and completion outputs from all the bits in a register fed to an AND gate whose output indicates the last bit to complete.

The test shown is for set up time, however it is not difficult to modify the selected condition, for example to be those inputs where the data does not change for a time equal to at least the hold time following the clock rising edge.

5 Conclusions

We have described techniques that allow the design of digitally set delay lines with accuracy better than 8ps. When used in conjunction with MUTEX time comparison, and time amplification circuits it is possible to measure on-chip signal path timing differences to accuracies of better than 10ps. This accuracy is currently mainly limited by the effects of fabrication imperfections in the MUTEX circuits that have been modelled in an 0.6μ CMOS technology. Processes with smaller dimensions may allow this figure to be reduced to less than 5ps, because the MUTEX circuits and inverters will be faster, but further reductions are likely to be dependent on careful interconnect layout, which will then dominate the delay setting accuracy.

Three methods of time measurement have been described, the first, which uses parallel MUTEXs with a tapped delay line, is analogous to a flash A/D converter, and the second to a successive approximation method. Both are fast, efficient methods of extracting a binary value for the time differences between signals, but the second requires less hardware for a large number of bits. The third method involves the use of a MUTEX to effectively amplify small time differences so that they can be more accurately measured. This technique allows accuracies of as good as 5ps.

These time measurement methods can be used in delay fault testing and in input tests, such as data set-up time conditions. Timing tests for multi-pin, submicron

circuits currently require the use of very expensive test hardware because it is very difficult to maintain a timing accuracy of less than 20ps with an externally connected system. We describe an on-chip method of testing these conditions, which may take its inputs from the chip boundaries in the same way as the external tester, but instead of using expensive, accurate external signal sources, we use uncorrelated signals whose statistics are known, and accurately select the conditions to be tested on-chip.

Given that timing margins can be measured, and conditions tested, a further application of this work is in adaptive synchronization where the methods could be used to adjust the clock or data delays to correct timing problems on-chip, using the variable delay lines described, and an appropriate control algorithm.

References

- [1] Y.Zorian. "Testing the monster chip". IEEE Spectrum, Vol. 36 No. 7, July 1999, pp.54-60.
- [2] S.W.Moore, G.S.Taylor, P.A.Cunningham, R.D.Mullins and P.Robinson. "Using Stoppable Clocks to Safely Interface Asynchronous and Synchronous sub-systems". Proceedings AINT2000, Delft, 19-20 July 2000 pp.129-132.
- [3] R.Ginosar and R.Kol. "Adaptive Synchronization". Proceedings AINT2000, Delft, 19-20 July 2000 pp.93-101.
- [4] C.Molnar, I.Jones. "Simple Circuits that Work for Complicated Reasons". Proceedings Sixth International Symposium on Asynchronous Circuits and Systems, Eilat, April 4-6: 2000.
- [5] D.J.Kinniment A.Bystrov, and A.V.Yakovlev. "Synchronization Circuit Performance". to be published JSSC, 2002.
- [6] O.V.Maevsky and E.A.Edel. "Converter of time intervals to code". USSR Patent Certificate 1591183, Class H03M1/50, Bulletin No. 33, 070990.
- [7] C.Dike and E.Burton. "Miller and Noise Effects in a Synchronizing Flip-Flop". IEEE Journal of Solid State Circuits Vol. 34 No. 6, pp.849-855, June 1999.