A Proof of the Adversary-Path Conjecture

Alain J. Martin
Department of Computer Science
California Institute of Technology
Pasadena CA 91125, USA
CaltechCSTR:2015.001

April 10, 2015

This proof is extracted from a more general document in preparation on the role of time in the implementation of digital computations. It is not entirely self-contained.

1 Background

The 1989 paper on the limitations to delay-insensitivity in asynchronous circuits[1] introduced the notion of isochronic fork and QDI logic. It showed that an arbitrary delay on an isochronic branch of such a fork could lead to malfunction, which implied that the length of the transition delays involving such forks had to be bounded in some way.

The timing assumption proposed in the paper was a direct reflection of the way in which the fork was created: At the logical level of production-rule (PR) representation, a single variable can be shared by several operators. At the physical gate level, the shared variable is distributed by a fork to a number of variables, one for each gate input. But in the PR model, all those variables switch together since they are represented by a unique variable; hence it seemed appropriate to assume that the transition delays on all variables corresponding to the tines of the fork were equal. While this assumption is sufficient to enforce a correct behavior, it is not necessary. Indeed, in many cases it is too strong for a reliable physical implementation: Any physical implementation of a QDI circuit relying on that assumption would be very brittle in the presence of any parameter variation.

But all circuits we designed and implemented turned out to be very robust, which suggests that a weaker assumption must exist. In [2], I proposed a weaker timing assumption based on the notion of adversary path and conjectured that it is the weakest timing assumption necessary and sufficient for the correct implementation of isochronic forks. In [3], Sean Keller and Michael Katelman give a proof of this conjecture. The present paper offers a simpler proof.

Adversary Path Conjecture: *Let $(x, x1, x2)$ be an isochronic fork, with concurrent transitions $t(x1)$ and $t(x2)$, where $t(x1)$ is acknowledged but $t(x2)$ is not. Let $x2$ be an input to gate $g2$, and let v be another input to $g2$. Then the timing requirement for the proper behavior of the circuit is that $t(x2)$ must complete before the completion of a chain of transitions (the adversary path) starting at $t(x1)$ and ending at a transition $t(v)$ on v.*

This requirement is a one-sided inequality on delays, which can always be satisfied by making the adversary path longer, e.g. by adding inverter gates on the path. In the remainder of the paper, we prove that, for any QDI circuit, the timing requirement for the proper behavior of any isochronic fork is always realized by the adversary path requirement.

2 Acknowledgment and Transition Causality

As stated in [4], *"a computation implements a partial order of transitions. In absence of timing assumptions, the partial order is based on a causality relation.* This causality ordering, which

we call *transition causality* in[3], is the same as the successor relation in [1]. It is constructed from the acknowledgment relation: if transition $x\uparrow$ makes guard By of transition $y\downarrow$ true, then $y\downarrow$ *acknowledges* $x\uparrow$, or $y\downarrow$ is the immediate successor of $x\uparrow$. By adding transitivity and anti-reflexivity, a partial order is constructed.

In the following, we will argue that if transition $t2$ follows transition $t1$, then either $t2$ is the immediate successor of $t1$, or there exists a non-empty chain of transitions between $t1$ and $t2$ that are totally ordered by the transition causality order.

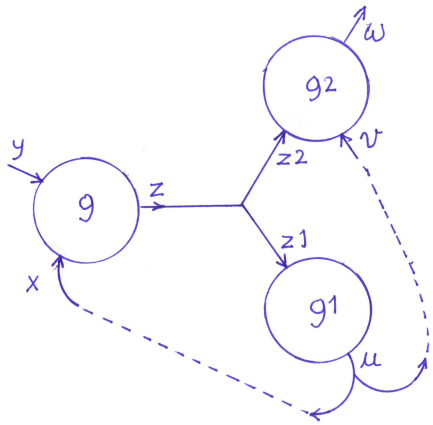

Figure 1: Three gates connected by fork (z,z1,z2) where some transitions on z2 are isochronic, i.e. do not cause a transition on output w.

3 Proof

Consider a close, non-redundant, QDI circuit S satisfying the stability and non-interference requirements. The circuit consists of a finite set of gates (operators) connected by wires and forks. Through an analysis similar to that of the Q-element example in [1], we have identified at least one isochronic fork in the circuit. Without loss of generality, we assume the fork to have exactly two tines. We focus on the three-gate subcircuit C containing this fork. Gate g has output z that connects by fork $(z, z1, z2)$ to gate $g1$ through $z1$ and gate $g2$ through $z2$.

Consider a non-terminating computation of S that involves C. Each transition $t(x)$ on a variable x is followed, after possibly a number of transitions on other variables, by the complementary transition $t'(x)$. Hence, every gate is part of at least one ring of gates, containing at least the minimum number of restoring gates so that the ring behaves as a ring oscillator.

Figure 1 shows gates g and $g1$ as part of a ring: the dotted arrow from output u of $g2$ to input x of g represents a feedback path closing the ring. A similar feedback path exists between output w of $g2$ and an input of g, possibly x as well.

The proof is in three parts.

3.1 First we prove that gate g2 has another input v whose value must change for g2 to fire effectively

Observe that $g2$ necessarily has another input v. Otherwise, $g2$ would be an inverter and any transition on $z2$ would cause a transition on the output, which contradicts the hypothesis that branch $(z, z2)$ is isochronic.

As part of the isochronic branch behavior of $(z, z2)$, gate g may go through a number of transitions on its output z without causing an effective transition on the output of $g2$. But since no part of the circuit is superfluous, at some point in the computation, a transition on input $z2$ of $g2$ will cause an effective transition on its output.

Let us analyze the part of the computation when transition $z2\downarrow$ is isochronic but the following transition $z2\uparrow$ is effective, i.e., causes a transition, say $w\downarrow$, on output w of $g2$. The partial sequence of transitions under investigation is

$$z\downarrow; (z1\downarrow, z2\downarrow); \ldots \quad z\uparrow; (z1\uparrow; \ldots\ ), (z2\uparrow; w\downarrow; \ldots\ ) \ .$$

(The comma indicates that two transitions, or two sequences of transitions take place concurrently, i.e., are unordered by the successor relation.) In the state preceding $z\downarrow$, $z2 = 1 \wedge w = 1$ hold, and we assume without loss of generality that $v = 0$ also holds in that state. Hence we have in that state: $z2 = 1 \wedge w = 1 \wedge v = 0$.

In the state following $w\downarrow$, $z2 = 1 \wedge w = 0$ hold. But for inputs $z2 = 1 \wedge v = 0$ of $g2$, output w is 1. Therefore, in order for $g2$ to produce output $w = 0$ with input $z2 = 1$, input v must have changed value from 0 to 1. Let's call this transition tv. When and how did tv occur?

3.2 Next, we prove that the transition on v is caused by a sequence of transitions on the adversary path.

Transition tv occurs after $z\downarrow$ and obviously before $w\downarrow$, since it causes $w\downarrow$. Hence, tv must be caused by $z\downarrow$: there must be a sequence of transitions—each acknowledging, and thus caused by, the previous one—starting at transition $z\downarrow$ and ending at tv. Transition $z\downarrow$ is followed by $z1\downarrow$ and $z2\downarrow$. But $z2\downarrow$ cannot be the next transition in this chain since it is isochronic, and thus is not acknowledged. Consequently, tv must be caused by a transition on $g1$'s output u: there exists a chain of gates starting at $g1$ and ending at input v of $g2$ such that $z\downarrow$ causes $v\uparrow$ through a sequence of transitions $(z1\downarrow; u\uparrow; \ldots v\uparrow)$ along this chain. This sequence of transitions is called *the adversary path of isochronic branch* $z2$. (Sometimes the term is used for the chain of gates.)

3.3 Third, we prove that a timing assumption is needed to avoid misfiring of g2.

Transition $z2\uparrow$ being effective, the condition on the inputs of $g2$ causing $w\downarrow$ is of the form $z2 = 1 \wedge v = 1 \wedge C$, where C is a condition on the other inputs of $g2$ if any; C is identically true if there are no other inputs.

Suppose that the last isochronic transition $z2\downarrow$ (i.e. the one immediately preceding the effective transition $z2\uparrow$) is slow. All transition pairs $z1\uparrow, z2\uparrow$ and $z1\downarrow, z2\downarrow$ on the two branches of the fork are concurrent. Therefore, isochronic transition $z2\downarrow$ is not ordered with any of the transitions of the adversary path $(z1\downarrow; u\uparrow; \ldots v\uparrow)$: $z2\downarrow$ could take place concurrently or after $v\uparrow$, which would lead to either instability of transition $w\downarrow$ or executing $w\downarrow$ out of order, since $z2 = 1$ holds before $z2\downarrow$.

Since causality is not sufficient to enforce the correct behavior of the computation, an ordering based on relative timing has to be relied upon. In order for $v\uparrow$ to take place after $z2\downarrow$, the propagation delay along the adversary path has to be longer than the time it takes for the single transition $z2\downarrow$ to complete.

Hence, the Adversary Path Theorem:

Theorem 1 *Consider a close, non-redundant, QDI circuit satisfying the stability and non-interference requirements, and containing an isochronic fork* $(z, z1, z2)$*, from output* z *of gate* g *to input* $z1$ *of gate* $g1$ *and input* $z2$ *of gate* $g2$*. Branch* $(z, z2)$ *is isochronic in some computation: for a concurrent pair of transitions* $t1$ *on* $z1$ *and* $t2$ *on* $z2$*, transition* $t1$ *is acknowledged but* $t2$ *is not. Then there exists a sequence of transitions starting at* $t1$ *and ending at a transition on input* v *of* $g2$*, called the adversary path of* $z2$*, such that, for any correct computation, the completion delay of* $t2$ *is always shorter than the propagation delay along the adversary path.*

This timing requirement, called the *adversary path condition* is necessary and sufficient, and is the weakest timing assumption to guarantee the proper behavior of a QDI circuit.

4 Transition Completion

When a transition on an input of a gate is not causing a transition on the output, when is the transition completed? An input transition either enables or inhibits a transition on the output. In restoring CMOS logic (no pass transistors), enabling means that the voltage change on a transistor's gate will contribute to tying a path between the output and a power rail; inhibiting means that the voltage change on the transistor's gate will cut a path between the output and a power rail.

The requirement that the isochronic transition $z2\downarrow$ completes before adversary path transition $v\uparrow$ means that the voltage change on $z2$ must cut a pull-up or pull-down transistor chain of gate $g2$ before the voltage change on v ties it.

References

[1] Alain J. Martin, The limitations to delay insensitivity in asynchronous circuits. In *Proceedings of the sixth MIT conference on advanced research in VLSI.* 1990.

[2] Alain J. Martin, Pyuish Prakash. Asynchronous nano-electronics: Preliminary investigations. In *ASYNC' 08.* 2008.

[3] Sean Keller, Michael Katelman, Alain J. Martin. A necessary and sufficient timing assumption for speed-independent circuits. In *ASYNC' 09.* 2009.

[4] Alain J. Martin, Mika Nyström, Asynchronous techniques for SoC design. In *Proceedings of the IEEE,* vol. 94, nr 6, 2006.