# Metastability in Latches, Arbiters and Data-convertors

Anthony C. Davies

*Dept. of Electronic Engineering, King's College, University of London, Strand, London, WC2R 2LS, England,*
*ph: +44-171-848-2441, fax: +44-171-836-4781, e-mail: tonydavies @ kcl.ac.uk*

*Abstract* - **Real-time digital systems and asynchronous computer systems have an inherent risk of occasional failures due to metastability in various components used to control data-transfers. This possibility has been known for a long time, although it is sometimes forgotten or overlooked. There is renewed interest in asynchronous computer systems because of the prospects they offer for lower power operation and improved electromagnetic compatibility compared to conventional fully synchronous systems. As a result, design approaches to minimise metastability effects need to be adopted. Recent increased understanding of non-linear dynamics and the availability of software for the accurate simulation and visualisation of dynamic behaviour enables metastability to be investigated and demonstrated much more readily. This paper provides a mainly-tutorial review of how metastability arises in various commonly-used components, illustrated with the results of simulations.**

## I. INTRODUCTION

Capturing external data by a real-time digital system typically involves clocking the data into some form of latch, such as a D-type flip-flop. To guarantee the proper operation of the flip-flop, data set up and hold times have to be complied with, but since the data and clock sources are independent, inevitably these timing requirements are occasionally violated: such violations can lead to metastability [1,2]. The same problem occurs in data transfers between subsystems within asynchronous computers.

Metastability arises from unstable equilibrium states in digital system components, which can result in the assumption that signals are two-level becoming invalid for brief but possibly significant periods of time. A simple example is the conventional set-reset latch (flip-flop) - it has two state-variables and two stable equilibria (from which its ability to store a 'one' or a 'zero' arises). Each stable state is surrounded by its basin of attraction but between these there is always an unstable equilibrium point. If the state-variables closely approach this point, the transient leading to either stable states may be substantially slowed down compared to normal, and this departure from expected timing may result in system failure.

Fig. 1 shows superimposed simulated trajectories from a simple latch when data-pulse timing is such as to take the latch very close to its unstable equilibrium state. Fig. 2 illustrates the flows in the state-space, from which the three equilibria can be seen, two stable, one unstable (arrow length and direction represent the gradient at each state).

To handle two or more concurrent requests to a shared resource in a computer system, it is usual to use some form of arbiter. Arbiters are also liable to metastability, and occasional operational problems can arise when the requests are near-to-simultaneous. The first explicit report of these effects seems to have been by Catt [3].

Analogue-to-digital convertors are relied upon to carry out accurate conversions of analogue signal values to digital form, but are subject to occasional data-conversion failures for similar reasons.

Observing metastability experimentally has always been difficult, which has encouraged a tendency to ignore it.

Some occasional 'failures' in real-time computing systems may be attributed to metastability, but because of the unrepeatability of the circumstances, verification is seldom possible, so other kinds of timing errors or software errors may be blamed.

Nowadays, accurate computer models of realistic digital circuits are available, together with powerful simulators which enable dynamic behaviour to be easily investigated. Most design engineers have access to powerful desktop computing resources. This makes possible the modelling and accurate dynamic simulation of complex configurations of gates and their timing behaviour using a variety of advanced software packages. Consequently, metastability may now be easily investigated for both realistic gate models and simplified idealised gate models.

Numerical solution of the non-linear differential equations of simple 'idealised gates' was used for the results illustrated in this paper.
A data-base of references to metastability accessible by WWW has been assembled [5].

## II. ASYNCHRONOUS DIGITAL SYSTEMS

It has long been recognised that asynchronous computer systems should have some advantages over the conventional synchronous digital computers. In a synchronous system, the maximum clock frequency is limited by the need for every gate's worst-case time delay to be complied with. In an asynchronous system, there is no need to 'wait for the clock'. This offers a potential speed-increase, because behaviour is dependant on the average and not the worst-case processing delay of each subsystem, but also increases the risk of metastability.
Recently, the needs for low-power systems (especially for hand-held battery operated mobile communications systems components capable of high performance digital data handling and processing, within the popular concept of 'multi-media'), has led to renewed interest in adopting asynchronous designs [6,7]. The European Commission funds work in this area through ESPRIT (the ACiD Working Group [8]), and the UK Asynchronous Forum now meets bi-annually.
Clock distribution to all parts of a synchronous system requires a high power signal liable to radiate troublesome interference at the clock frequency and its multiples, and assuring adequate electromagnetic compatibility is not easy. Also, most parts of such a system consume power continuously, whether or not they are occupied in useful processing operations.
An asynchronous system often incorporates locally-synchronous subsystems, which need consume significant power only when they are doing needed work - for the rest of the time they can be put into a power-down ('sleep') mode. Because each such subsystem is self timed, sometimes with its own local clock, the radiation from a shared clock is avoided, and since the clocks of the various subsystems are not synchronised, any radiated interference tends to be at a much lower level and broadband, so avoiding the spectral peaks of the noise emission from synchronous systems.
A commercial example is the asynchronous re-design by Philips Semiconductor of the 80C51 microcontroller. This uses one quarter of the power and has dramatically less clock-radiation compared with the synchronous version fabricated by the same technology [9].
Shrinking of integrated-circuit component-dimensions, a major factor in the steady increase in microprocessor performance, reduces gate delays but relatively increases interconnect delays. This makes it increasingly difficult to distribute high-speed clocks across the whole of a complex chip, leading to synchronisation problems, clock-skew, etc. in synchronous systems which may be circumvented by asynchronous designs.

## III. ARBITER CIRCUITS

An arbiter is used to select between two or more concurrent requests for service or for access to a shared resource. This is an inherent requirement in many kinds of data-transfers and interrupt-handing systems. It has been known for a long time that all arbiters are subject to the risk of metastability [10, 11]
The simplest example of an arbiter is a set-reset flop-flop (normally followed by a 'filter' which is a digital circuit to reduce the probability of a metastable transient propagating to the output). The metastable level (between the high and low logic levels) can persist at the output of a flip-flop for a significant (and theoretically unbounded) time, and the effect of the 'filter' is to hide this metastable level from the following circuits - however, it can achieve this only by delaying the time at which the final output can be 'trusted' - so that at best an 'uncertain amplitude' is exchanged for an 'uncertain time-delay'.
A simple 'filtering' idea often advocated is simply to follow the metastable output by a gate with a very low (or high) threshold. The intention is to keep the output constant until the input has departed sufficiently from the intermediate metastable level for a 'clean' digital level to be reached by a fast transition at the final output. However, this can reduce but does not eliminate the risk.
It is possible to make '1 out of n' arbiters from combinations of basic gates, but unless care is taken, these can exhibit various additional forms of metastability, because of their increased number of unstable equilibrium states.
The '1 out of 3' arbiter of Fig. 3 [12, 13] suffers from several problems (such as modes where fairness in responding to input requests is not guaranteed). Fig. 4a shows output transients from a large number of initial states for the condition that all inputs are held high (for which there are three stable states). As well as normal transients, some delayed (metastable) responses can be seen. Fig. 4b shows trajectories initiated from a number of closely adjacent states in the vicinity of the state-space origin, leading near to a metastable point and terminating on one of the three stable states [14].

## IV. DIGITAL-TO-ANALOGUE CONVERTORS

The standard successive-approximation analogue to digital convertor is liable to occasional conversion errors, which may not be small. It is well-known that such errors can arise from not keeping the input analogue signal constant over the duration of the conversion process. Less often realised is the possibility that errors can occur even if the input is held absolutely constant [15]. The convertor uses a comparator to compare the

input signal level with an internally-generated level. The comparator is, in effect, a very high gain amplifier which is supposed to be always driven into saturation in one direction or the other (so generating a digital output). However, if the two input signal levels are very close the output may be at some intermediate value within the 'linear' range of the comparator. As well as possibly affecting any serial output from the convertor, this signal will be 'clocked' into the register-system which drives the internal digital-to-analogue convertor and can result in uncertainty about the state latched into the register. Of course, with a properly-designed convertor, using a high-gain comparator, the error probability is extremely small, but is not zero.

## V. SYNCHRONISERS

A synchroniser is required to accept input transitions at any time, and to generate corresponding output transitions synchonised with the timing of a local clock. Compared to a standard 'latch' which may be regarded as 'level-sensitive' (e.g. capturing an input level), the synchroniser is 'transition-sensitive'. Failure modes include a slow output transition or intermediate output-signal levels. Metastability, oscillations and suggestions of chaos have been reported [3,4,10]

## VI. ATTEMPTS TO AVOID METASTABILITY

Various unsuccessful schemes have been proposed to eliminate metastablity.

Claims that making flip-flops from Schmitt-trigger gates (e.g. gates with hysteresis) can overcome metastabilty have been shown to be false [4].

An 'inertial delay' is a filter which is supposed to 'clean' metastable transients (also called 'glitches' or 'runt pulses'). However problems of designing a perfect inertial delay are essentially the same as problems of making a perfect synchroniser [16], and so are doomed to failure.

The Muller 'C-element' [17] is a popular replacement for latches as a memory element in asynchronous circuits. It detects and follows when two inputs are both high or both low, and stores the corresponding level while the inputs differ. Denoting the inputs by $x$, $y$ and the output by $z$, it may be represented by the assignment:

$$z := (x \textbf{ and } y) \textbf{ or } (z \textbf{ and } (x \textbf{ or } y))$$

Metastability is possible if the inputs change level in opposite directions or do not remain at one level long enough for the element to respond properly.

Fig. 5 shows metastable transients at the output $z$ from a simulation of a C-element made from 'and' and 'or' gates. These occur because the duration of the simultaneous high level on $x$ and $y$ is too short for the C-element to respond adequately.

The 'asymmetric-C element' [7] is a variation in which one input is dominant: the output $z$ goes high if $x$ or $y$ are high, while if $x$ is low, the output $z$ goes low. This can also exhibit metastablity.

## VI. DISCUSSION AND CONCLUSIONS

Asynchronous designs offer scope for overcoming limitations of synchronous designs which arise in the context of faster and smaller systems and in low power and hand held systems, but a reappraisal is required of the risks of metastablity. The recent availablity of excellent tools for non-linear dynamic systems analysis and for visualisation offer the prospect of improved insights and a more detailed evaluation.

The key problem lies in the mapping between discrete and continuous domains. In the time domain, the sequencing of events (which represents the behaviour at the software level) has to be mapped to the continuous time domain and in this process occasional unpredictable delays from metastability may occur. In the amplitude domain, latching a level by a clock and conversion of a continuous-amplitude to one of a discrete number of levels (by an analogue-to-digital convertor) leads to the possiblity of occasional conversion errors. Uncertainty in level may be exchanged with uncertainty in time but complete elimination of these problems cannot be achieved. Good design can reduce the probability and consequences of these infrequent events to an acceptable level but this requires system designers to be aware of them.

*Acknowledgements:* Colleagues in the Department of Computing Science, University of Newcastle-upon-Tyne, and at Matra BAe Dynamics are thanked for discussions about the material in this paper. The U.K. EPSRC is thanked for financial support (Grant No. GR/L92471).

## REFERENCES

[1] L.R. Marino "General Theory of Metastable operation", *IEEE Trans.*, C-30, pp. 107-115, 1981

[2] A. C. Davies "Analysis of Metastable Dynamics of Bistable Flip-flops", *Proc. 6th Int. Symp. on Networks, Systems and Signal Processing*, Zagreb, Yugoslavia, pp. 379-382, June 1989

[3] I. Catt "Time loss through gating of asynchronous logic signal pulses" *IEEE Trans.*, EC-15, pp. 108-111, 1966

[4] T.J. Chaney "Comments on "A note on synchroniser or interlock maloperation" *IEEE Trans.*, C-28, pp. 802-804, 1979

[5] www.eee.kcl.ac.uk/~comfort

[6] S. Nowick, M.B. Josephs, C.H. van Berkel (editors) *Proc. IEEE,* Special Issue on 'Asynchronous Circuits and Systems', Feb. 1999

[7] J. Kessels and P. Marston "Designing Asynchronous Standby Circuits for a Low-Power Pager" *Proc. IEEE*, pp. 257-267, 1999

[8] www.scism.sbu.ac.uk/ccsv/ACiD-WG
[9] H. van Gagaldonk *et al* "An asynchronous low-power 80c51 microcontroller" *Proc. Async'98*, San Diego, pp96-107, 30 March - 2 April 1998
[10] T.J. Chaney and C.E. Molnar "Anomalous Behavior of Synchronizer and Arbiter Circuits", *IEEE Trans.*, C-22, pp. 421-422, 1973
[11] W. Fleischhammer and O. Dortok "The anomalous behavior of flip-flops in synchronizer circuits", *IEEE Trans.*, C-28, pp. 273-276 1979
[12] C.H. van Berkel and C.E. Molnar "Beware the three-way arbiter' *accepted for IEEE Trans. Solid-state Circuits*, June 1999
[13] A. Bystrov and A. Yakovlev "Revisiting the problem of Fair Arbitration", *5th U.K. Asynch. Forum*, Cambridge, England, 17-18th December 1988
[14] A.C. Davies "Multi-flops - a view of the dynamic behaviour" *accepted for NDES'99*, Denmark, July 1999
[15] D.J Kinniment *et al* "Towards Asynchronous A-D Conversion", *Proc. Async'98*, pp206-215, 30 March - 2 April 1998
[16] J.C. Barros and B.W. Johnson "Equivalence of arbiter, synchronisers, latches, and inertial delays" *IEEE Trans.,* C-32, pp603-614, 1983
[17] C.L. Seitz "System Timing", Chapter 7 of C. Mead and L. Conway *'Introduction to VLSI Systems'* Addison Wesley, pp. 218-262, 1980

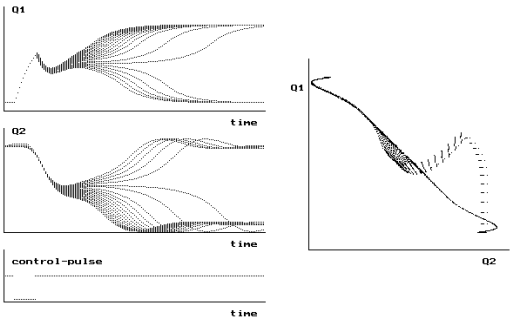


Figure 1. Metastable transients of flip-flop

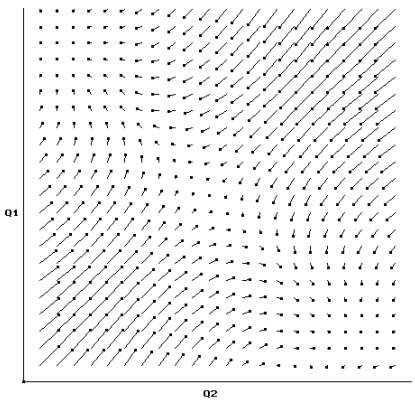


Figure 2. Flip-flop: flows in state-space

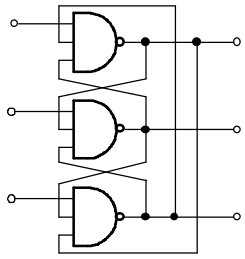

Figure 3. '1 out of 3' arbiter

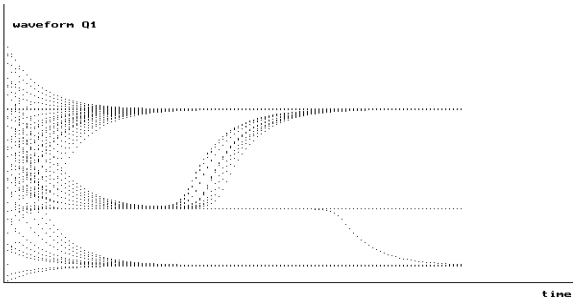


Figure 4a. Transients from '1 out of 3' arbiter

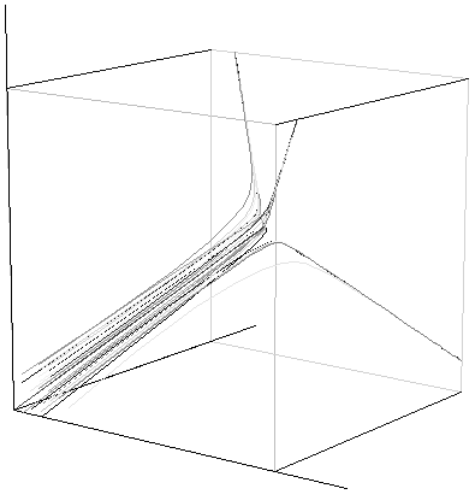

Figure 4b. Trajectories of '1 out of 3'arbiter

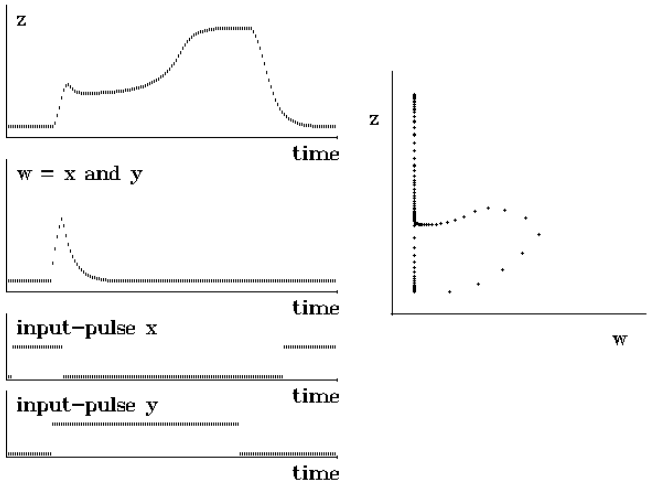


Figure 5. Metastability of 'C-element'