

Asynchronous Logic: Born to be wild

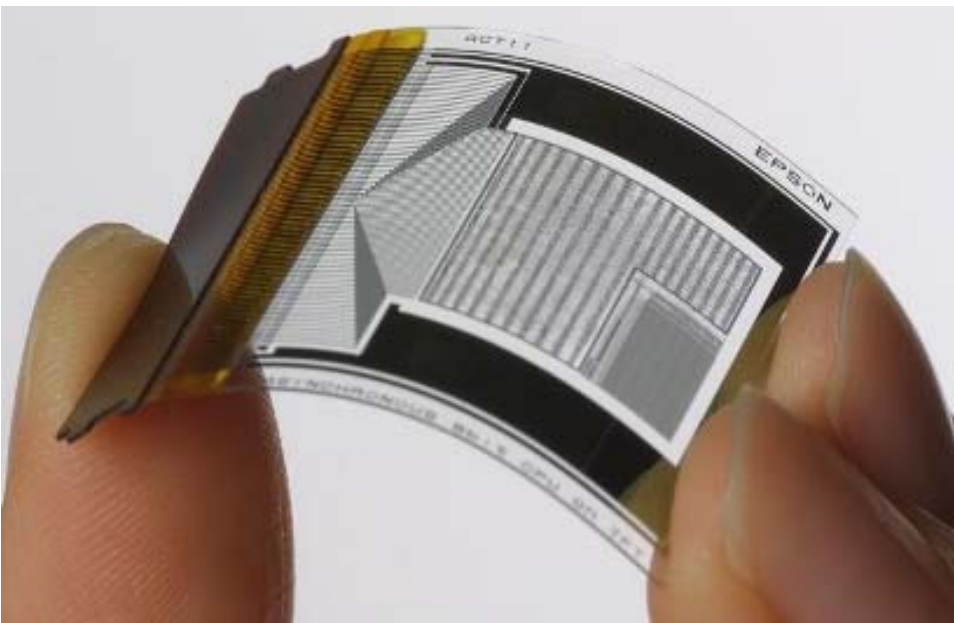
Harry Yeates
September 13, 2005

Conventional chips are inefficient. Governed by a clock signal that wakes up every component with each tick, regardless of whether or not it is actually needed for the task at hand, they run at the speed of the slowest component, wasting power and spewing excess noise as they do so.

Clockless chips, built using any of a number of different approaches, do not suffer from these limitations. They are, like Peter Fonda in *Easy Rider*, “hip about time”. However, asynchronous designers are still seen, like Fonda, as iconoclasts, pursuing their art in the main as members of academic research groups, or perhaps within a start-up company. Surely, in these days of complex system-on-chip, with their challenging integration issues, the benefits of asynchronous logic are relevant for the mainstream?

Earlier this year a UK academic announced a tool that could help to push asynchronous design into the lexicon of the average synchronous chip designer. Software developed by Damon Thompson, a research associate in the School of Engineering and Electronics at the University of Edinburgh, can take a synchronous design and translate it automatically into a clockless equivalent.

“We accept RTL as the input to the tool, and the output of our process is a generic logic netlist, which can then be technology mapped for silicon standard cell libraries, or FPGA backend architectures,” says Thompson. “The designer can optimise that depending on the characteristics they are interested in, whether it’s high performance, low power, noise generation or any other [factors].”



Epson's flexible 8-bit microcontroller using asynchronous logic



Compared to clocked chips, asynchronous designs typically employ extra lines and gates to support their signalling protocols, which impacts on silicon area. However, Thompson claims the extra real estate introduced on sample designs that have been put through his flow – including encryption engines, DSP components, filters and codecs – is 30-40 per cent. And targeting a new process is simplified too.

“Asynchronous design answers the whole process technology migration issue quite nicely, in that units designed asynchronously have homogeneous timing interfaces,” says Thompson. “You can migrate a design from one technology family to the next far more easily than you can with a clocked circuit, which would need to be redesigned and re-verified at every step through the flow.”

The research up to now has been funded by a grant from Scottish Enterprise’s Proof of Concept fund, but that ended in August and the plan is to incorporate Asynchronous, the company that will commercialise the technology, in Q3 of this year.

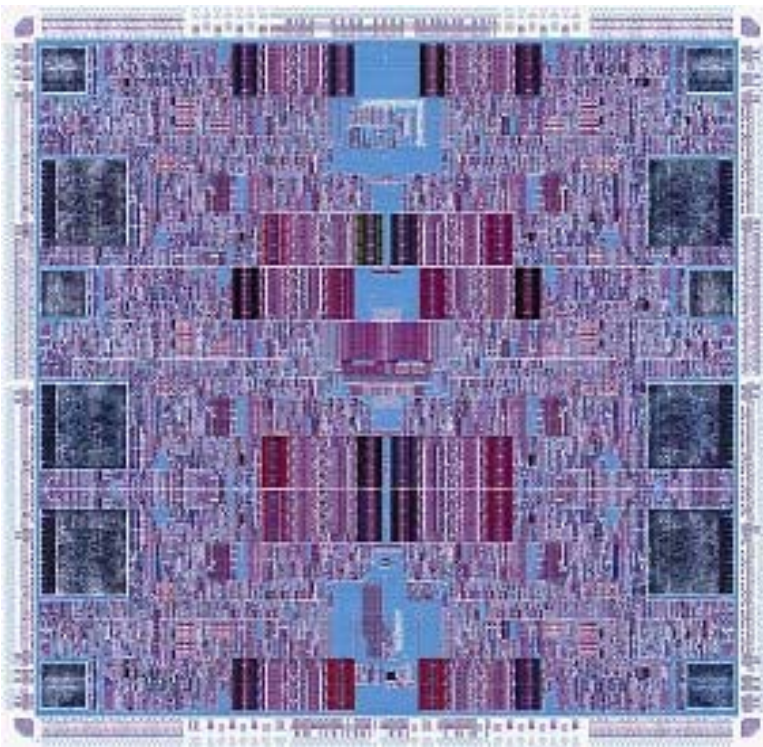
The firm will join a handful of start-ups developing clockless technology – among them Fulcrum Microsystems and Theseus Logic in the US; the Philips spin-out Handshake Solutions; and Self-Timed Solutions, Professor Steve Furber’s company at the University of Manchester.

In fact, Philips’ clockless technology has been in products since 1995, and is now in use in more than 100 million ICs for applications such as contactless smartcards, in which the low EM signature is a security bonus. At the beginning of last year the Dutch firm created Handshake Solutions, based at the Philips Technology Incubator in Eindhoven, to sell and license products, including its Haste language, which it originally made available as the Tangram ‘silicon compiler’.

“What we offer is design tools to make clockless designs, and instead of using a clock we use what we call handshakes to organise the activities on the chip,” explains Rik van de Wiel, the company’s COO.

“The tools that we have allow you to start at a very high level description, a behavioural level - it looks a bit like C - and translate it automatically to a netlist and then use standard EDA tools to go to a layout,” says van de Wiel. “So it is just a front-end to normal way of working, but it is a different design style, so people have to get used to that.”

A handshake is a two way messaging interaction between components on the chip. A request from one component to another to start up is answered with an acknowledge signal. In this way only the components required to execute the particular operation are woken up. Handshake Solutions’ priority is low power.



Fulcrum's 1GHz processor using asynchronous circuits

Van de Wiel says that testing asynchronous designs has historically been a challenge, but that Handshake has developed technology to tackle it. “In an asynchronous design style [testing is] quite difficult,” he says. “We have found a solution for that, and in our approach you can just use the standard way of testing as you do for a synchronous design, so you get the same quality level.”

Developing methods for the formal verification of asynchronous design has been the preoccupation of Dr Georgios Theodoropoulos, a lecturer in the School of Computer Science at the University of Birmingham. Theodoropoulos has been working with Furber’s Advanced Processor Technologies group to link up the Balsa language, which is based on Tangram, with formal verification and simulation.

However, while simulation can guarantee correctness in synchronous designs, in asynchronous equivalents it cannot. For example, temperature changes might affect signal speed, which alters the order in which events happen and can result in a deadlock.

“I think the main obstacle to widespread uptake of an asynchronous design methodology is the lack of tools,” says Theodoropoulos.

So maybe it can be summed up like this. It is all very well being hip about time, but if you do not carry some tools then, like Peter Fonda’s Easy Rider character, you are in danger of being blown away with a shotgun. Something like that, anyway.