Asynchronous Wait-Free Communications in Distributed Real-Time Computing Systems

What it is**, Why** it is useful and **How** to do it.

Prof. Anthony C. Davies
Department of Electronic Engineering,
King's College, University of London, Strand, London, WC2R 2LS, England
tonydavies@kcl.ac.uk

As speed and complexity of integrated circuits increases, chip designs are likely to comprise many locally synchronous processors communicating asynchronously - sometimes known as 'GALS (globally asynchronous locally synchronous) - because the difficulties of distribution of a high-frequency clock across a complex integrated circuit may make fully synchronous designs impracticable.
There may also be a trend to using fully asynchronous designs. Compared to synchronous designs, these offer the possibility of lower power consumption (since switching is needed only when computation needs to be performed) and better electromagnetic compatibility (because of the elimination of a high-power, high-frequency clock). Asynchronous designs could also be faster than synchronous ones.
In such a context, reliable interprocess communications between independent and unsynchronised processors is required. It is often assumed that all interprocess communications necessarily involve some form of synchronisation (for example, the ADA rendezvous) or a system of handshakes, which implies waiting, and a risk of deadlock in the case of either faulty design or component failure. The possibility of designing mechanisms to implement wait-free communication between processes (whether on the same or different processors) is often assumed impossible, and the need for arbiters in the access to the associated shared memory is assumed inevitable.
However, for a class of interprocess communications for which the write operation is destructive and the read operation is non-destructive, it is possible to devise communications mechanisms in which the writer is always free to write and the reader is always free to read. The data objects to be communicated are assumed to be either records with a number of fields or arrays with a number of elements, with the requirement that the reader must always obtain a coherent object which should be the most recently available one. Obviously more than one shared memory location for the data objects is necessary, since the reader and writer must be able to operate concurrently, and simultaneous reading from and writing to the same location is not possible.
The mechanism is required to satisfy properties of data coherence, data sequencing and data freshness, and additionally the effects of metastability must be allowed for when operating in a fully asynchronous environment.
A number of wait-free mechanisms have been proposed, in most cases independently of one another, based upon widely differing concepts and analysed by radically different methods. These designs are typically rather complicated and difficult to understand, so it is by no means easy to be confident about their correctness, and some designers have overlooked important properties that such mechanisms ought to satisfy.
The talk will review the basic principles, describe the properties that a wait-free asynchronous communications mechanisms has to satisfy, and illustrate this with a description of some 'four-slot' mechanisms which meet the correctness requirements in a fully asynchronous environment, and which are capable of efficient implementation in digital hardware.

A paper on this topic entitled 'A Comparison of some Wait-Free Communications Mechanisms' by Ian G. Clark and Anthony C. Davies was presented at a workshop on 'Asynchronous Interfaces: Tools, Techniques and Implementations' (AINT'2000), 19th-20th July 2000, Delft, Netherlands. Further information can be found at `http://www.eee.kcl.ac.uk/~comfort`

Anthony C Davies is professor emeritus at King's College London, having taken early retirement from administration and teaching duties at King's College in October 1999. He moved to King's College from City University London in 1990, where he was for a number of years Director of the Centre for Information Engineering. He has in the past taught at Purdue University and University of British Columbia.
He was awarded a B.Sc.(Eng.) with First Class Honours by Southampton University, an M.Phil by University of London, and a PhD by City University. He is a Fellow of IEEE and IEE
See also `http://www.eee.kcl.ac.uk/member/staff/a_davies.html`
His main recent teaching activities have been in digital signal processing and in software design methods, and in the last few years, his research funding has covered the areas of non-linear dynamics and chaos, the estimation of crowd behaviour, and more recently, aspects of asynchronous systems design.
He is currently the IEEE Circuits and Systems Society Vice President for Region 8.