everything on cpu

Terry Yin

September 28, 2014

Hi Craig,

I read you prediction “we will see the full integration of non-volatile memory onto the same silicon die as the CPU,” and the question you asked. My first thought was that this is not desirable and not likely to happen. But when I think more about this idea, I start to feel it might be a good idea.

Let’s don’t confuse this with the already existing idea of “microcontroller”, Wikipedia (2014), which is simply put the CPU and memory on one physical chip and has limited areas of using. I guess you are proposing a deeply integrated chip with the emerging technology and will be general-purposed, like it will be used to build both small phones and large data center.

Nowadays CPU slowed down to increase in frequency but start to have more cores. Because of the need for parallel processing, functional programming languages will perhaps dominate the future, Hughes (1989) (yes, the reference is a bit too old). Unlike the structured or object-oriented programming paradigms, the functional programming does not share mutable states among the processes, and information is pass in the form of closures. So let’s be very open minded, having each core of the CPU having its own non-volatile memory as a highly cohesive unit might be a superb idea. Maybe even the system software on the same chip as well. The concept of closure could be extended to “you can unplug the physical unit at any time and ship it oversea and continues to execute from there.” We could build highly distributed, functional style system using these chips.

Things that belong together should be kept together (Cohesive Software Design, by Tim Ottinger, Jeff Langr). Nobody keeps the two pieces of a pair of scissors separately. This is called cohesion. What things should be kept together? I think they should be **the things that is used together and changed together**. In the traditional computer architectures, CPU and memory are used together, but often changed separately. I think that is why they are usually decoupled. Will the emerging functional programming style of design change this situation and make them a cohesive unit?

br, Terry

References

Hughes, J. (1989), ‘Why functional programming matters’, *Computer Journal* **32**(2), 98–107. cited By (since 1996)139.
URL: *http://www.scopus.com/inward/record.url?eid=2-s2.0-0024648265&partnerID=40&md5=326e561e4b4e4897a29292ccfd9dbde5*

Wikipedia (2014), ‘Microcontroller — wikipedia, the free encyclopedia’. [Online; accessed 28-September-2014].
URL: *http://en.wikipedia.org/w/index.php?title=Microcontroller&oldid=627227088*