

PRAM-On-Chip: A Quest for Not-So-Obvious Non-Obviousness

Uzi Vishkin¹

University of Maryland Institute for Advanced Computer Studies and Electrical and
Computer Engineering Department, College Park MD 20742-3251, USA,
vishkin@umd.edu,

XMT home page: <http://www.umiacs.umd.edu/users/vishkin/XMT/>

Abstract. Consider situations where once you were told about a new technical idea you reacted by saying: “but this is so obvious, I wonder how I missed it”. I found out recently that the US patent law has a nice formal way of characterizing such a situation. The US patent law protects inventions that meet three requirements: utility, novelty and non-obviousness. Non-obviousness¹ is considered the most challenging of the three to establish.

The talk will try to argue that a possible virtue for a technical contribution is when, in retrospect, its non-obviousness is not too obvious; and since hindsight is always 20/20, one may often need to resort to various types of circumstantial evidence in order to establish non-obviousness.

There are two reasons for bringing this issue up in my talk:

(i) seeking such a virtue has been an objective of my work over the years, and

(ii) issues of taste in research are more legitimate for invited talks; there might be merit in reminding younger researchers that not every “result” is necessarily also a “contribution”; perhaps the criterion of not-so-obvious non-obviousness could be helpful in some cases to help recognize a contribution.

The focus of the second focal point for my talk, the PRAM-On-Chip approach, meets at least one of the standard legal ways to support non-obviousness: “Expressions of disbelief by experts constitute strong evidence of non-obviousness”. It is well documented that the whole PRAM algorithmic theory was considered “unrealistic” by numerous experts in the field, prior to the PRAM-On-Chip project. In fact, I needed recently to use this documentation in a reply to the U.S. patent office.

An introduction of the PRAM-On-Chip approach follows. Many parallel computer systems architectures have been proposed and built over the last several decades. The outreach of the few that survived has been severely limited due to their programmability problems. The question

¹ The legal definition of the non-obviousness requirement is as follows: A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains.

of how to think algorithmically in parallel has been the fundamental problem for which these architectures did not have an adequate answer. A computational model, the Parallel Random Access Model (PRAM), has been developed by numerous (theoretical computer science) algorithm researchers to address this question during the 1980s and 1990s and is considered by many as the easiest known approach to parallel programming. Despite the broad interest the PRAM generated, it had not been possible to build parallel machines that adequately support it using multi-chip multiprocessors, the only multiprocessors that were buildable in the 1990s since low-overhead coordination was not possible. Our main insight is that this is becoming possible with the increasing amounts of hardware that can be placed on a single chip.

From the PRAM, as a starting point, a highly parallel explicit multi-threaded (XMT) on-chip processor architecture that relies on new low-overhead coordination mechanisms and whose performance objective is reducing single task completion time has been conceived and developed. Simulated program executions have shown dramatic performance gains over conventional processor architectures. Namely, in addition to the unique parallel programmability features, which set XMT apart from any other current approach, XMT also provides very competitive performance.

If XMT will meet expectations, its introduction would greatly enhance the normal rate of improvement of conventional processor architectures leading to new applications.