EDUCATION FOR THE DEEP SUBMICRON AGE: BUSINESS AS USUAL?

H. De Man
Katholieke Universiteit Leuven/IMEC, Belgium
deman@imec.be

Abstract:

Exploitation of deep-submicron technology will depend critically on the availability of global system engineers able to bridge the gap between software-centric system thinking and hardware-software implementation of it in novel silicon architectures. This requires a rethinking of present engineering schools which are not well equipped to tackle global system engineering aspects. The concept of design institute is introduced where, based on visionary system design demonstrators, new methodologies, tools, libraries and courses are created and distributed over the global network. Design institutes provide a learning school for new design paradigms and form the ideal environment for the education of global system designers.

1. Deep-Submicron system design needs more brains than tools

As a community of CAD developers we often want to solve new emerging design problems by the creation of new tools. However, as stated by J. Borel at last ISSCC, deep-submicron technology evolves at an even faster rate than predicted by the SIA roadmap. The 0.18µm milestone will be reached one year ahead of schedule. This creates new challenges for the design of novel computing and communication systems in silicon but...tools to cope with these challenges can only be created when the problems are well understood and formalised. Since the time between concept and exploitation of tools for new paradigms is 7..10 years, we must now be working on methods to support system design in the year 2005 when silicon will have reached the 0.05µm level.

Since nobody really knows what these silicon architectures will look like, the first priority is not tools but the education of a new breed of engineers in design centric engineering schools. They should be able to bridge the gap between a new product definition and its implementation on the “silicon board”. The research topics of these schools must be on future information system architectures and their implementation in deep-submicron technology. Formalisation of such methodologies can give rise to new tools after the global architectural issues have been well understood but in the mean time the people to design such systems will at least be there if we start the process in time. The need for investing in education and training of silicon system engineers is grossly underestimated and the nature of that education may be fairly different from today’s engineering education. This is the subject of this paper.


A lot of the actual CAE is serving mathematically well formulated generic design problems such as logic-FSM-RT level synthesis, place and route and simulation. It serves part of an ASIC oriented community and is based on our ability to make valid abstractions of the underlying physical level. This was perfectly possible down to the 0.5µm level. Below that, big changes are occurring both at the physical level but, most importantly, also at the global architectural level of future silicon systems. Precisely this is at the origin of the need for a new breed of engineers: the global system designers. We now discuss some of these changes.

2.1: Changes at the physical level

Deep Submicron effects such as dominance of interconnect delay and “analog” effects due to high performance and low power, low voltage operation are conflicting with the safe abstraction wall between physics and logic. Standard cell layout style may no longer be consistent with interconnection strategies or new low power circuit techniques. Actual logic and RT level synthesis does cope with power management issues and low power clocking schemes and architectures. DRAM-logic technology, crucial for multimedia systems, conflicts with IP standardisation issues etc. Important as this may be, there are changes of a more fundamental nature at the architectural and at the global system level.

2.2: Changes at the architectural level

Analysing the recent major advances in electronic systems, reported at ISSCC and CICC, we see that the main paradigm shifts in system design are generally the result of brain based system engineering and/or a daring system vision often resulting from a close interaction between industry and academia. A.o. some of the main advances are:

- Implementation of DSP and digital communication systems into silicon architectures (DSP cores, silicon compilation 2,3)
- Advanced silicon architectures for compression of speech and images 4
- Very low power circuit, architecture and algorithm design techniques 5
- The ability to integrate complete RF front-ends in CMOS silicon 6,7
Fig. 1: Evolution of the chip from a stand-alone component to a multi-paradigm system embedded in a network providing services to the consumer. It will take a long time before design technology catches up with the creativity of design itself. In the mean time “people ware” must be produced! Notice enormous impact of software.

• Advent of ASIP’s with embedded software and reconfigurable compilers.
• Advances in low power high throughput AD interfacing and filtering.
• Advances in high throughput interfacing techniques, embedded DRAM and reconfigurable logic.

Fig. 1 shows how these system paradigms, first developed as specialised stand-alone components, are now all merging into the same silicon carrier which becomes a “silicon board” containing a heterogeneous architecture.

Notice that existing CAD tools do not yet serve all these individual component domains let alone the interaction and trade-offs between them. However, precisely these interdependencies are crucial to create a working system. Therefore, it is of utmost importance to create the engineers able to optimise the global architecture of such systems. But there is more...

2.3: Changes in the global information system architecture

Until recently, the electronics industry was nicely segmented into computing, communications and consumer electronics. However computing has evolved first from professional computing to personal local computing and recently to communication and consumer oriented network computing. Communication and data access becomes globally wired and locally wireless.

Under pressure of internet, the wired infrastructure rapidly becomes broadband and multi-user wireless local networks will follow that trend bringing full video mobile internet access to the consumer.

The spectacular growth of the telecom sector results from the massive move towards mobile communication and a much more intensive use of the existing wired infrastructure due to the exponential increase of internet traffic (growth from 10 to 400 Tbytes/months in two years). These events create an insatiable demand for engineers able to design new products and services made possible by the deep-submicron silicon technology. But this will not happen by simple extrapolation of past system architectures.

Indeed, already today’s 0.35μm technology requires some fundamental changes in design methodology. First, the enormous NRE cost and redesign risk of these complex multi-paradigm chips inevitably leads to a much greater embedded software content. However software will run on architectures which, for power consumption and cost reasons, have to be drastically different from actual computing architectures. This is imposed by the fact that “Bill Gates” or “Infinite Resource” type software leads to the emerging memory and power crisis of classical mono-processor computing architectures as discussed at last ISSCC.

Hence, there will be a lot more domain specificity in design solutions than before and, besides the expensive μP technology driving the SIA road map, a cheap combined memory-logic CMOS technology may emerge to implement low power consumer products for the end user in the information society.

That same CMOS technology will be used to implement also the RF parts for local wireless access on the same silicon carrier. The combination of all these design technologies can lead to the Personal Internet Device that is linked wirelessly to the home base station that provides access to the broadband wired global network.

Second, after 2005, silicon technology will most probably hit some economical barriers at the 0.05μm level. However, at that point its integration potential will be so overwhelming that the problem will no longer be in the hardware design part but in the ability to provide enough software productivity to keep up with new services to be provided. This situation will occur less than a decade from now. Beginning freshmen at the university will reach their Ph.D. and the design market by that time!

But also engineers working in the information industry today will have to cope with these rapid changes...let us use information technology and a rethinking of engineering education to take care of these problems.

3. Needed: the global system engineer:

As shown in Fig. 1, silicon is rapidly becoming the unifying medium into which system design paradigms from RF to analog-digital interfacing, hardware processors and embedded real time software are merged to implement...
network services for the consumer. Experience shows that such systems must be globally optimised within the environment in which they operate (network or even service level). For example, designing a single chip wireless transceiver requires a deep understanding of interdependencies between RF-ANALOG-DIGITALSOFTWARE architectures AND the properties of the network into which it is to be embedded. Correct partitioning determines the ultimate power, feasible dynamic range, final area, packaging cost, time-to-market and hence economic product viability. Such trade-offs are not yet supported by tools but require global multi-disciplinary system thinking. This can only be learned by submerging students and faculty into real system design projects with the goal to formalise and encapsulate experience into reusable methods, libraries and tools shared by the entire educational community.

That requires in the first place that we remove the communication gap in many engineering schools between the “VLSI” education and the communication and computer science departments.

First, within “VLSI” classes usually too much attention is paid to RT level digital design of control logic and too little to DSP, communication, analog and RF silicon architectures. The link to such realities as power dissipation, memory architectures and area consumption is often hidden behind blind application of logic synthesis tools. The latter are not yet capable of synthesising power and layout efficient architectures or novel circuit and clocking schemes for low voltage architectures.

Second, communication classes focus on new algorithms for digital communications but little or no attention is paid to implementation issues. RF classes stick to classical heterodyne principles and do not consider challenges and opportunities offered by deep-submicron technology.

Third, Computer Science classes focus a lot on ”infinite Resource Software” gobbling up gigabytes of memory and making microprocessors dissipate over 100 Watt’s of power...hardly what you need for the new consumer products. What we need for silicon systems is more attention to the link between hardware and software. This means implementation strategies on new, cheap, more dedicated and hence power efficient silicon architectures for the personal multimedia devices of the future. This requires a lot of rethinking of compiler technology for yet to design flexible multi-processor real-time architectures with distributed memory.

One must prevent that students are pushed too much into the corner of one particular discipline such that they lack the ability for global system thinking. This situation indeed often leads to the optimisation of non-issues which add nicely to the citation index but are of little use for progress of applied information technology.

The gaps mentioned above are, unfortunately, also a reflection of similar gaps in the system industry itself. This causes a tremendous loss in time-to-market because of communication hick-ups whereby silicon engineers have to re-engineer network protocols and communication architectures to be able to map them into silicon. This is the gap between the C++/MATLAB world and the VERILOG/VHDL world. In the sequel we first reflect on the skills required from the global system engineer able to exploit deep-submicron opportunities for the new products and services of the information age. We will define this engineer as the global system engineer. Then we will reflect on the curriculum and modus operandi for the design institute producing this peopleware.

4. Skills of the global system engineer (GSE) :

The GSE is a broadly educated engineer who is able to bridge the gap between communication and computing products and their implementation in silicon, photonics and display components. This implies an education based on an active knowledge of “rf-analog-digital-hardware-software-network-services” concepts.

Fig. 2 illustrates the concept of the education of the GSE. This education model contrasts sharply with a classical deep specialisation in a single discipline of engineering science (the ideal of space and military days in the 60’s and 70’s ). The skills of the global system engineer can be summarised as follows:

![Diagram](image_url)

Fig.2. The training of the global system engineer creates a bridge between product needs in information systems and implementation in heterogeneous systems within socio-economic constraints. Design and non-technical skills result from major multi-disciplinary projects managed by a professor team. This team gets its vision from interaction with advanced system industry.

1. Must be broadly educated in basic subjects of EE-CS with emphasis on problem solving and self-learning skills. This requires a well balanced, rather rigid, major-minor system in EE - CS. Education in “software engineering close to hardware” of EE’s must be drastically increased in view of the evolution shown in Fig. 1 and the note on “silicon technology saturation” in 2005.

2. Problem-solving is learned by project work in multi-disciplinary concurrent engineering teams at the system level as opposed to the "genius Ph.D." doing his individual thing. Projects must evolve from analytical nature at undergraduate level to state-of-the art open-end product design at graduate level.

3. As they cannot possibly know everything, GSE’s must be well trained for self-learning and communication. Since the key issue is product design they also must be aware of the economic aspects of engineering.
4. Training in communication skills and reporting is acquired within the projects under 2. Therefore the projects have a management structure in which students and faculty play “leader and team” member roles. Help by industrial psychology and economics department may be helpful here. Principles of entrepreneurship must be part of the education as students should be encouraged to create start-ups in innovative products resulting from research (see Fig. 3 below).

5. GSE’s must be able to embed, formalise and document design expertise, methodology, libraries and algorithms in well conceived, reusable software objects that are further made available over the network such that a design knowledge base becomes available for bootstrapping the next generation of students. This way of working reflects the trend towards increased use of IP over the web. It also points at the important role of the semiconductor and CAE industry to make IP and software cheaply available to these design centric schools. In return the added value of academic IP is available to them and it creates the engineers with the right skills.

In order to set-up an appropriate framework for the education of global system engineers, one must also introduce global system design experience within the university system. That requires also a different mode of operation of the faculty where the professor now has to team up in multi-disciplinary design projects. That leads to the concept of a “design institute” discussed below.

5. The concept of design institute

The fundamental problem of classical engineering schools is that they are based on individually acting professors and students. However the complexity of global systems exceeds the capacity of a single professor who is presently awarded for being extremely smart in an isolated subject in engineering science...Few engineering professors have experience in system design. Is the creation of schools in system design possible? The answer is positive since examples of it are rapidly emerging as the next speakers will amply illustrate. Indeed, many basic advances in system architectures mentioned in section 2. point to an academic origin. They result from multi-disciplinary research teams with a sufficient critical mass having an intense interaction with the systems industry. We will call such an operation an “design institute.”

Characteristic is that their research results are proven by working demonstrators and publicly available software rather than by “theorem proving” in abstract publications. The latter are more like a by-product which tends to be more useful as it refers to the solution of real–instead of a smart but fictitious problems.

The Infopad project at UCB16, the Human Computer Interaction Institute at CMU17, the wireless modem project at UCLA18, the broadband wired and wireless modem project at IMEC19, are but a few examples of such design institutes which provide the breeding ground for the education of the global system engineer. These projects or institutes are all characterised by the following:

1. Their research, at the basis of teaching, is the demonstration of the feasibility of a global system concept at the leading-edge of technology. Research is theme driven and 5..10 years ahead of current practice. The goal of such projects is the formalisation of design methodologies and the encapsulation of it in reusable software. Direct transfer of the technology in itself is not the direct goal. Actual transfer takes place in the brains of people trained for technical leadership, ready to create new products either in new start-up companies or in the affiliated major system houses or CAE companies (See Fig. 3). The project students are designers and on-the-fly tool developers at the same time.

2. The ambitious research goal can only be achieved by faculty team building willing to share and build knowledge to reach a common goal each within and at the fringes of disciplines, where “concepts and ideas rub and spark against each other to create new approaches and new solutions” [Dan R. Olsen, ref 17]. Notice that this may not be consistent with the actual “science” dominated publication policy but it sure helps to create the right people. It also requires a deeply respected visionary academic to take the lead of such an institute (the project champion).

3. Design institutes also have a strong Industrial Affiliate Program which makes industry an active partner. This keeps research goals realistic, generates long term system requirements by visionary dialogues and entitles the industry to prototype background and foreground research results and direct contact with most important product: the well trained global system engineering student.

Notice that such a design institute is a reflection of the modus operandi of the industry in which the engineering students will work after graduation. It provides the framework for global thinking by induction from their team professors (as opposed to the old-fashioned isolated genius). As Fig. 3 indicates, a design institute presupposes some centralisation of the best minds around a common theme requiring their centre expertise and their willingness to team-up. The common theme results from a champion vision distilled from a pro-active interaction with major industrial partners. Willingness of academics to team-up is not very consistent with past and current academic practice. Hence funding encouragement is required to glue faculty together in places where the necessary critical mass exists. This requires the courage of funding agencies to award effective leadership in innovation and a willingness to focus funding instead of spreading it thinly.

![Fig. 3: The structure, interaction and products of a design institute. The numbers indicate time in years between concept and result.](image-url)
This does however not mean that smaller universities (N in Fig. 3) are excluded from such a system. They should be encouraged to join theme projects to participate in well defined research on open problems identified by the leading institutes and the affiliated industrial partners. Modern network technology provides an excellent opportunity to get rid of geographical barriers. Just as virtual companies emerge, virtual engineering schools can be conceived. Networked sharing of educational IP of the best schools will contribute to a better education, a better use of education funding and a faster progress in technology. In conclusion one can state that a design institute will:

- Create professors understanding the global system issues themselves;
- Make them able to teach it;
- Will attract more students as subjects become nearer to reality;
- Will prepare students better for the information industry.

6. Education program for the global system engineer

Fig. 4 shows a tentative master and Ph.D. program embedded in a design institute for information technology. It has the following characteristics:

1. Regular courses are based on team teaching and are restricted to basic concepts. Team teaching focuses on engineering aspects and context. As the grey arrows indicate, at least 30% of the time budget is spent on project work of which the majority (vertical arrows) is devoted to interdisciplinary projects.

2. The first two years are analytical in nature and should provide the scientific basis to engineering. "Basic science" should however be thought by engineers for engineers. Also the basics of EE, CS and Mechanics are thought. Basic science is linked to engineering by devoting 25% of the time on product analysis seminars. These are hands-on seminars whereby basic science professors relate theory to practice by analysing aspects of a popular product (VCR, CD-player, Modem...). Global design aspects of such product are explained by engineers from industry.

3. In view of the important role of software in future information systems, the master program should provide a balance between EE and CS subjects whereby CS should be EE oriented i.e. pay less attention to "infinite resource programming" but more to:

- Embedded software: real time aspects;
- Software impact on power dissipation, memory architecture etc.;
- Computational models (data-flow, control flow, concurrency);
- OO programming as a means for system modelling, knowledge reuse and on the fly tool development;
- Use of networking as an engineering tool;
- Use of commercial and university CAE to focus on architectural aspects.

This means less computer "science" but more software/hardware engineering.

4. The emphasis of the master program is on the ability of problem solving both individually and in team. Hence at least 25% of the time budget has to be spent on interdisciplinary design projects (vertically shaded arrows). In these projects students get involved in more and more complex design problems from specification to implementation involving several EE and CS disciplines. A management structure involving both faculty and students "runs" these projects. In this way concurrent engineering principles, team work and communication skills are introduced in practice. The master thesis is devoted on a more in depth individual design task (including tool development if appropriate).

5. The Ph.D. program is devoted to research within the theme project of the design institute. Ph.D. students focus on aspects of large system design problems, whereby emphasis is placed on combining engineering disciplines. The Ph.D. program should be oriented to create enhanced design productivity by demonstrating new design methodologies. Such methodologies should be documented by reusable software available over the network. This form of publication may turn out to be a lot more useful than the actual overproduction of academic papers of which the impact can...
actually not be easily tested...Methods, tools and libraries shared by researchers will be naturally selected. Their acceptance rate can be a new and, from an engineering viewpoint, more interesting "citation index". Ph.D. programs in design institutes must lead to the creation of technical leaders (as opposed to scientists). The character and mode of operation of design institutes as discussed in section 5 encourages that.

7. Conclusions.

Deep-submicron technology combined with the tremendous growth of global networking requires a rethinking of computing and communication architectures and of new multi-disciplinary design methodologies. This creates the need for global system engineers to bridge the gap between system level concepts and their implementation in heterogeneous silicon architectures. The concept of design institute is introduced where academic staff and students team-up around visionary information technology design projects. Building actual software-hardware prototype demonstrators is undertaken as a means to create new methodologies that give rise to new CAE tools, libraries, teaching material and training courses available over the network.

A design institute is the ideal environment for educating the global system engineer. In such schools emphasis is on broad education in the basics of computing, communications and components. Design practice (including non-technical skills) are learned by team-based integrated projects in product design.

8. References:

5. http://infopad.eecs.berkeley.edu
17. http://pecan.srv.cs.cmu.edu/afs/cs/user/hcii/www/