OCCASION

This publication has been made available to the public on the occasion of the 50th anniversary of the United Nations Industrial Development Organisation.

DISCLAIMER

This document has been produced without formal United Nations editing. The designations employed and the presentation of the material in this document do not imply the expression of any opinion whatsoever on the part of the Secretariat of the United Nations Industrial Development Organization (UNIDO) concerning the legal status of any country, territory, city or area or of its authorities, or concerning the delimitation of its frontiers or boundaries, or its economic system or degree of development. Designations such as "developed", "industrialized" and "developing" are intended for statistical convenience and do not necessarily express a judgment about the stage reached by a particular country or area in the development process. Mention of firm names or commercial products does not constitute an endorsement by UNIDO.

FAIR USE POLICY

Any part of this publication may be quoted and referenced for educational and research purposes without additional permission from UNIDO. However, those who make use of quoting and referencing this publication are requested to follow the Fair Use Policy of giving due credit to UNIDO.

CONTACT

Please contact publications@unido.org for further information concerning UNIDO publications.

For more information about UNIDO, please visit us at www.unido.org

UNITED NATIONS INDUSTRIAL DEVELOPMENT ORGANIZATION
Vienna International Centre, P.O. Box 300, 1400 Vienna, Austria
Tel: (+43-1) 26026-0 · www.unido.org · unido@unido.org

Software For Embedded Applications: Employment Opportunities For Developing Countries

U.P. Phadke, A.B.Patki

Abstract:

Embedded systems are being extensively used in diverse application areas. This paper focuses on the engineering aspects of developing embedded systems in the context of hardware and software components. The impact of fuzzy logic on embedded systems and need for rapid prototyping is brought out. Authors discuss the software embedded applications as the new vistas for employment opportunities.

Key Words: Embedded Systems, Fuzzy Logic, CAD Tools, Employment Opportunities

1.Introduction:

Software Embedded Systems encompass a wide variety of software components running on hardware, either microprocessors or microcontrollers. These systems are contained within a larger, usually non-electronic applications environment. These embedded systems perform specific & dedicated functions in applications ranging from large systems like satellites, atomic power stations, process control through medium range applications like hydraulic manipulators, commercial fire alarm systems, machine control, to consumer applications such as automobiles, washing machines, portable and handheld gadgets. Essentially, the embedded hardware & software performs control functions including add-on intelligence with increasingly digital technology usage. Embedded applications of software emerged with the availability of microprocessors in the late seventies. The word 'embedded' is used to bring out the distinction from standard mainframes, minicomputers, workstations & personal computers. Embedded software with associated hardware performs specific functions as opposed to general functions of computers. Recent advances in microelectronics have resulted in an emerging discipline in the form of industrial practices towards hardware and software codesign including the host system environment aspects. The present level of transistor density and integration achieved in Very Large Scale

Consultant: Uday P PHADKE
BC: D. Rhind, ITPD/INF

Integrated (VLSI) circuits enable building of complete systems on a chip permitting computational functionalities beyond the capabilities of 'retrofit' approach that was predominant in the initial stages of the growth of Applications Specific Integrated Circuits (ASICs).

The market for embedded electronic systems has grown with the advent of microprocessor based microcontrollers. Apart from the industrial automation systems, the emerging trends in utilizing embedded hardware/software systems to speed up existing software applications have resulted in new vistas for embedded systems in the field of network and communication applications. A typical embedded system consists of a processor core with memory (RAM/ROM), interface circuitry for analog to digital and digital to analog conversion, glue logic and one or more ASICs.

It is expected that in future, embedded systems will have a significant role in large systems development similar to the role played by ASICs in electronic hardware on printed circuit board in the past.

2. Engineering Considerations in Building Embedded Systems:

Usually the life span of an embedded system is the same as the life of the host system. Thus, embedded systems have to satisfy several stringent requirements including safety aspects, real time response, fault tolerance considerations etc. Embedded computer systems and their production technologies are important engineering considerations. Since the electronic embedded systems use digital technology, it is natural that they are built around commercially available processor cores running embedded software. When the standard microprocessor/microcontroller or digital signal processing chip based approach is adopted for developing embedded systems, the software plays a significant and critical role in the performance of such systems. This methodology is preferred for small volumes as against the approach of VLSI level hardware dominance for high volume requirements. Use of quality techniques in the early phases of embedded systems development are of utmost significance. Organizations which have already adopted ISO-9000 standard practices have a better potential to gear up quickly to respond to software embedded applications development since the ISO-9000 practices are required to be adhered to at all stages of system development covering the hardware, software, host environment etc.

2.1 Hardware and Software Design Issues:

Typically, an embedded system is implemented partly in hardware and partly in software. Thus, from design angle the three broad areas viz: hardware design, design of embedded software and hardware/software codesign need to be integrated. While many commercial CAD tools are available for synthesis, optimization, verification for hardware at logic level, these tools lack higher level design support such as application specific processor architecture alternatives, CAD tools required in planning phase etc. The hardware design has been further influenced with the advent of hardware description languages. Thus, VHDL/Verilog based VLSI CAD tool suites with adequate support from silicon foundries are being increasingly preferred by the VLSI design community . VHDL is very similar to a programming language but the end result is net-list description of a piece of hardware and not an algorithm to be executed on a processor in workstation, personal computer or a mainframe [1]. Developing and simulating VHDL models is a major concern in designing large embedded systems. In order to reduce the cost and time needed to design, upgrade and replace digital embedded hardware, VHDL with object oriented extensions needs to be used throughout the design process from system to gate level. This facilitates abstract modeling required to study architecture level trade-offs and performance of embedded applications [2].

The VLSI CAD library containing a broad range of standard cells as also several foundry specific mega-cells offers the possibility of integrating complex components such as microprocessors and memory devices into embedded system design. These components are highly optimized from area/time considerations. The hardware design automation perspective needs to be extended to include the support for architecture level abstraction, application specific Instruction Set Processor, Arithmetic Logic Unit optimization etc. Design of Instruction Set Processor involves software profiling and analysis to optimise the instruction set.

In VLSI hardware development, any design modification at a later stage means at least one additional fabrication run. Thus design process for embedded applications calls for the starting point shift in the conventional sense of VLSI design i .e. to commence design at 'high level', with the integrated tools. Thus, planning phase CAD tools fully integrated with VLSI design activity are needed and should include in-depth validation and verification software. The embedded system hardware developers view the VLSI hardware design automation tools equivalent to library elements used in the design for processor, memory and glue logic. Gate and operation level parallelism is supported in classical hardware design tools; but the CAD support for instruction level parallelism for 'fetch-decode-execution' is the requirement for complex embedded systems, calling for more design trade-offs at higher levels of abstraction. This leads to wide acceptance of design techniques from the computer architecture domain in embedded systems area. In the absence of integrated planning and VLSI design CAD tools, the practice of reliance on human skills is adopted. This results in longer project development cycle with possibility of adverse effect on production phase as well.

The characteristics of embedded software typically include relatively modest size, tight hardware software coupling, low error rates, real time constraint requirements, safety-critical issues etc. Typically, a C program is considered as a hardware specification but is regarded as implementation in software. The code optimization for embedded system goes beyond the traditional compiler optimization. Also, a mix of assembly language with C is the practice followed in developing the embedded systems software having the typical code size in the range of 10,000 lines.

2.2 Impact of Fuzzy Logic Technology on Embedded System Applications:

Fuzzy Logic is based on Fuzzy Set theory developed by Prof. L.A. Zadeh. It uses concepts, principles and methods developed in fuzzy set theory for creating various algorithms for approximate reasoning. The boundaries of the fuzzy sets are not required to be precise. Three decades of the Fuzzy Systems development have moved from the academic phase, through transformation phase including successful practical applications development into the fuzzy boom phase which is characterized by a rapid increase in successful industrial applications [3].

Fuzzy Logic has influenced both the Information Technology products and the consumer products. The characteristics of portable products incorporating fuzzy logic include limited space, lightness in weight, ease of operation, low battery power drain etc. Fuzzy logic based embedded applications are providing intelligent control making systems more adaptable. These use specialized ASICs and/or a mix of digital circuits and dedicated embedded software implementing a fuzzy algorithm. Thus, consumer products like washing machines, vacuum cleaners, lift control systems, video systems etc., execute fuzzy logic based algorithms. There is a strong need for a Rapid Prototyping CAD environment for developing fuzzy logic based embedded systems. A preliminary study carried out by the authors for developing rapid prototyping CAD environment indicates a need for incorporating user interaction right at the design stage. A case study of developing an embedded application around fuzzy logic based Rater System [4] brings out the need for prototyping CAD tools. Some of the important considerations are:

a) characterizing features to standardize input-output representations and extracting product specifications.
b) reducing dependence on manual intervention in software simulation phase for incorporating user interactions with the product at the design stage.
c) provision for supporting iterative processing for implementations in product layout plan and associated functions, behavioral software.
d) providing linkage and interface with product layout plan to the ASIC and standard ICs in interconnections as an entity vis-a-vis downloading in Field Programmable Gate Array.

The pilot study in developing Rapid Engineering CAD Integrated Prototyping Environment (RECIPE) highlights the need of developing an appropriate platform with provisions of features to support commercially available design automation software tools, along with integrated functioning of programming language like C/C++, VHDL, software utilities for VLSI layout, printed circuit board design etc., as object oriented modules. The development of fuzzy logic based hardware itself is a growing area and will have its impact on hardware design methodology for embedded applications. Some experiences in developing fuzzy logic hardware are reported which have impact on existing software language translators like VHDL, C++ [5]. It is felt that as the discipline of embedded system application grows in the near future, fuzzy logic is likely to play a significant role in embedded systems hardware and software development.

2.3 Testing of Embedded System Applications:

Testing is an important engineering aspect in embedded system development. Unlike general purpose software testing, software embedded systems have to be tested in the host environment. Usually, the electronics with software is tested for test data vectors conforming to the internal functions. But the host system environment is in simulator/emulator form. Thus, there is a need to have extra test points and test patterns for host system interface. The major challenges are in demonstrating safety-critical requirements & fault tolerance during testing . While the real time response requirements are usually satisfied, the safety-critical aspects are difficult to test. The electronic '0' and '1' test vector approach for safety-critical requirement amounts to functional testing and the actual safety-critical testing can be carried out only when the embedded system is in operation in the actual host environment. Safety-critical testing includes tools for creating and analyzing fault trees, failure mode effects & criticality, automatic generation of guard code etc [6]. For smaller systems, it is easy to demonstrate but for complex systems like process control embedded system, testing of complete system for safety-critical domain can only be undertaken as field trials. Usually, the functional testing of embedded software is carried out in a workstation based host environment, simulating the hardware drivers and interrupt handlers of the target system. The other building blocks in the host environment affect the performance of embedded system module severely when it comes to assessment of the safety critical aspects. In electromechanical systems, the inertial delays play a significant role in propagating the faults resulting in failures. CAD support does not exist to simulate such inertial delays in the host system building blocks.

3. Software Embedded Applications - Developmental Aspects:

Development of the software process takes a central role in the embedded systems engineering community. The problem is further compounded as the electronics and automation industries are not only re-engineering their products but modernizing the environments used to maintain the control software used in embedded applications. This underscores the need to introduce effective tools for managing software changes in software embedded systems. The reason behind software changes is not only the need to correct bugs/faults, but also adaptation to a new development environment. Reverse engineering tools for producing structure charts based on the existing code are useful. Thus, software process and software development issues take a more important role in embedded systems development as compared to applications software development and include process modelling, reusability and maintenance as part of application management. Some of the important elements in Software development for embedded systems are as follows:

- Synthesis of software programs for embedded control applications.
- Object oriented constructs for VHDL for abstract modeling support to study architecture level trade-offs.
- Software Reuse Technology for embedded applications.
- Incremental prototyping technology for embedded software using graphical animation.
- Code generator capable of producing full real-time C-code from high-level structured designs and specifications.
- Requirements engineering for embedded computer systems (structured and object oriented techniques).
- Tools for Object-oriented development of embedded systems including documentation guidelines and testing practices.
- Simulation based testing of embedded software including Fault Tree, Failure Mode Effects & Critically Analysis.

Embedded System applications pose new design challenges. Hardware and Software design should be carried out as a combined approach with host system included as a run-time environment all through [7,8,9]. Thus, a co-synthesis approach to achieve computer aided design of embedded system is the key methodology. Co-synthesis is divided into performance modeling, partitioning of hardware and software, integrated approach for synthesis & simulation of hardware and software to incorporate user feedback at design stage itself. The impact of other emerging technologies on software embedded applications also affect the software design of embedded systems.

Thus, software embedded applications are likely to create many new avenues for research, applications development, development of new sets of skills etc. The main opportunities will be in the following areas:

i) New CAD Tools for design & testing
ii) Training Kits using Multimedia
iii) Collaboration between Hardware and Software Industrial Units.

3.1 Need for New CAD Tools Development for design & testing:

The present practice in developing embedded systems is predominantly a manual design process at abstraction level leading to long design times and high costs of design. The need to meet the growing demands for embedded applications has given rise to CAD tools for design including rapid prototyping of integrated hardware-software codesign, simulation and product adaptation to incorporate user feedback. The hardware software cosynthesis spans many disciplines from CAD theoretic aspects such as algorithms to system implementation issues of partitioning. This creates opportunities for development of special CAD tools for embedded systems development addressing hardware architectural selection, software profiling and analysis, system performance estimation, simulation capable of incorporating user feedback for improving product adaptability etc [10,11,12].

Automatic partitioning into hardware and software is an important step in embedded system development. Critical parts in software are identified by profiling the code and a simulated annealing based algorithm is used to select the parts to be moved to hardware. This is also necessitated by the trends in the area of VLSI design.

All the above CAD tools will have to be supported on PCs as well as workstations requiring DOS/Windows and UNIX operating systems, respectively. By and large design automation software works on workstations in the UNIX operating system. However, in view of the potential of Field Programmable Gate Array (FPGA) based approach of building digital hardware systems, some electronic design automation tools like Viewlogic systems are available on PC-DOS platform. This trend will further open up opportunities for porting many VLSI CAD related and associated software packages from workstation arena to PC platform. This trend suggests development of new CAD tools for hardware-software codesign on PC platform to facilitate third party software support.

3.2 Training Kits Using Multimedia:

Development of embedded systems requires a multidisciplinary approach. In the present education systems, the emphasis is either on teaching Computer Science or Computer Engineering. On several occasions, this proves to be a handicap when engineering systems are to be developed, calling for continuing education and/or on the job specialised courses. Since host environment is of paramount importance for embedded systems, the developers need to have basic background in engineering with hardware & software as tools of a "computer workshop". Thus, it is expected that multimedia based training kits will be required to be developed & will replace the existing methodology of training through software tutorials and help utilities. This opens up several opportunities in the area of "training for trainers" giving hands-on exposure in embedded systems. Such multimedia systems will be required to have integration capabilities with instrumentation and devices like logic analyzers required for testing the software embedded applications.

3.3 Collaboration between Hardware and Software Industrial Units:

The spectrum of industrial units depicts a scenario of localized software houses, electronic hardware manufacturers, system integration units etc. Multidisciplinary nature of embedded systems & growing market opportunities likely to come up in this area underscore the need for strong collaborative efforts amongst hardware and software industrial units. Presently, such integrated approaches are on a very limited scale. In the coming decade, however, this will be on the rise. In this regard, one can expect the growth to be on similar lines to that seen in the past between the electronic design automation tool developers and silicon foundry units with each playing a complementary role to the other.

4. Employment Opportunities for Developing Countries:

In order to consider the employment opportunities in this field, we discuss the present practices adopted in software embedded applications industry. The hardware is built around the Central Processing Unit (CPU) which is commercially available. The popular CPUs used are 8051, 683XX, Z80/180, 386/486, 680X0 etc. The programming language adopted is usually a mix of assembly and C with the code size in the range of 10,000 lines. Preferred debugging tools include Emulators, ROM monitors, Logic analyzers, Scope etc. Also the developers per project are in the range of 4 to 5 with an average developer having an experience of about 10 years. Most of the developers refer to technical magazines like Embedded Systems Programming, EDN, Electronic Design, Computer Design etc.

Developing countries can play a role of consumer as well as producer for software embedded applications. As a consumer, the role would include the ability to spot potential areas for software embedded applications, drawing up detailed technical specifications for outright purchase or development of products/systems, installation & commissioning, operation, maintenance, upgrades & replacements etc. Embedded systems call for some special skills which can be acquired through short term courses specially oriented for the experienced work force. In addition, the new entrants can be equipped with specialised laboratory course modules during academic curricula at Diploma level. This results in an immediate requirement in the form of 'Training for trainers'. Multimedia based training kits can serve a useful purpose for this requirement.

The new vistas opened up by software embedded applications bring out needs for development of new CAD software tools. In the past, developing countries have played a significant role in software development, both on-shore and off-shore. However, unlike the software development for Data Base systems and commercial & business applications like banking & insurance, the embedded system software is highly engineering oriented. Thus, the new employment opportunities will be primarily in the CAD tool development for engineering applications. There is tremendous potential but undertaking such tasks is difficult as an off-shore activity carried out solely in developing country. However, with the interest shown by multinational companies in opening operations in developing countries likelihood of a solution path exists. This calls for some actions on part of developing countries as well.

From the above, it is clear that software for embedded applications is more towards the higher end of the software domain and is currently a niche market. One of the competitive advantages which developing countries have vis-a-vis the developed countries is that of relatively inexpensive manpower, spanning the range from unskilled manual labour to highly skilled and trained professionals with engineering and science degrees. It must, however, be reiterated that even the skilled professionals would require considerable upgradation of their skills and also acquisition of new inter-disciplinary skills before they can contribute effectively to this area. Given the complex nature of this area, it is felt that employment opportunities would be much less to cater to the domestic market in developing countries and the potential is more likely to be for end-use in the developed countries. This could be in the form of offshore development for the developed countries or as on-site development contracts in the developed countries. An interesting parallel can be drawn from the experience in India which is acknowledged internationally as a developing country with significant potential in software. While the domestic software in India grew from Rs. 4900 million (1 US $ = Rs. 35) in 1992-93 to Rs. 16900 million in 1995-96, the software and services exports grew from Rs. 6750 million to Rs. 25500 million. A recent internal study has projected a turnover of Rs. 125000 million by 2001-2002 for the domestic software industry as compared to Rs. 218000 million of software and services exports. More interestingly, in the area of ASIC design and related services, there is insignificant domestic market in India whereas the export turn over in 1995-96 is about US $ 10 million which could be boosted to US $ 200 million with an aggressive pro-active role by the Government.

In the area of trained manpower for software embedded applications, the situation today is not very encouraging and needs introduction of theoretical & practical laboratory courses in academic curricula at university level education. Authors have had first-hand experience of the substantial efforts needed in training students from engineering colleges to bring them upto a level before they can be introduced to embedded system applications design. In most developing countries the bottleneck in the IT area is not a lack of resources to develop infrastructure but the nonavailability of professionals who can identify meaningful applications and, thereafter, design the requisite software and then implement these systems. University programs and curricula along with user training could be two major focus areas of immediate attention. Developing countries like India, Malaysia, Africa, Zambia and others have already taken steps in training manpower in Information Technology area. These efforts need to be further focused in the area of embedded system software development.

User training is often a neglected area resulting into low absorption rate of new technologies and software embedded applications are no exception to this. Training methodologies for end-user application development should focus on extensive end-user participation in application development activity calling for development of customized training material and its delivery at the pace organisations can absorb. The practice of borrowing curricula from elsewhere needs to be modified to suit the local requirements and infrastructure of the country and also to reflect the application environment.

It is felt that Industry Associations will have to come forward to face some of these challenges if they want to tap the market segment of software embedded applications. While collaborations with other organisations are stop-gap arrangements, Industry Association's participation as sponsors for boosting opportunities in this field has to come more liberally and enthusiastically. Policy formulators and decision makers in the Governments of developing countries would have to create a conducive environment for such a role by Industry Associations. The initiative could come in the form of establishment of Regional Centres for Embedded Software Applications which can play a pioneering role in the area of applications development & demonstrations, as well as training the work force. Such centres can play a pivotal role in opening up avenues for enterprenuers to form consortiam for this industrial sector. With the cooperative joint collaborations amongst academic institutions, industry, R & D Laboratories, the activity could be expanded for tie-ups with developed countries. This will help in opening new market linkages.

In order to set up infrastructure facilities for 'training the trainers', in many developing countries UNESCO efforts were found useful in the past. With the advent of new technology like multimedia, the objectives can be achieved within shorter timeframes of 3-5 years resulting into many employment opportunities in developing countries. The employment potentials for project leaders, CAD tool developers, trainers, application developers etc. are likely to be seen in the near future.

5. Conclusions:

This paper outlines the software embedded applications and its impact on employment opportunities for developing countries. The paper brings out the technical issues associated with embedded applications development and illustrates the scope & potential for software development. The impact of emerging technology like fuzzy logic on software embedded applications is brought out. It is felt that with the joint effort of Industry Associations and policy formulators and decision makers in the Governments of the developing countries, employment opportunities for developing countries can be boosted in the field of software embedded applications.

6. Acknowledgment :

Authors acknowledge the active participation of student trainees associated with the pilot study undertaken by them for developing Rapid Engineering CAD Integrated Prototyping Environment for software embedded fuzzy logic based application. The facilities and support extended by Department of Electronics, Government of India and willingness of educational institution to permit their students in exploring new areas is acknowledged.

7.References :

1. *R.Camposoano, W.Wolf (editors), "High-Level VLSI synthesis" Kluwer Academic Publishers, 1991.*

2. *M.J. Flynk, "Some Computer Organisations and their effectiveness", IEEE Transaction on Computers C-21 (9), September 1972, pp 948-960.*

3. *Patki AB, Phadke UP - Fuzzy Logic Based Hardware Systems, Technology Watch, Electronics Information & Planning, pp 214- 217, January 1994*

4. *Deshpande S, Patki AB, Raghunathan GV - Towards Rapid Prototyping For Fuzzy Logic Based Portable Products, First On-line Workshop on Soft Computing, Nagoya University, Japan, August 1996*

5. *Patki AB, - Fuzzy Logic Based Hardware: Some Experiences Proceedings of the International Discourse on Fuzzy Logic and Management Complexity (FLAMOC'96), Sydney, Australia, January 1996.*

6. *Patki A.B.- Fuzzy Set techniques for Failure Modes, Effects and Criticality analysis, Proceedings of National Seminar on Fuzzy System & Non Standard Logic, Calcutta, May 1984*

7. *W.Wolf, "Hardware-Software Co-Design of Embedded Systems", Proc. IEEE, Vol.82, No.7, July 1994, pp. 967-989.*

8. *D.E. Thomas, J.K. Adams, H. Schmit, "A Model and Methodology for Hardware-Software Codesign", IEEE Design & Test of Computers, September 1993, pp 6-15. 9.G.De Micheli, "Computer Aided Hardware Software Co-design, IEEE Micro, Vol.14, No.4, August 1994, pp 10-16.*

10. *M.Chiodo, P.Giusto, A. Jurecska, H.Haich, L-Lavagns, A. Sangiovanni, "A formal methodology for Hardware/Software Co-design of Embedded Systems", IEEE Micro. Vol.14, No.4, August 1994 pp; 26-36.*

11. *B.Dasarathy, "Timing Constraints of Real Time Systems: Constructs for Expressing Them, Method of Validating Them", IEEE Transaction Software Engineering, Vol.SE-11, No.1, January 1985, pp. 80-86.*

12. *A.Shaw, "Reasoning about Time in Higher Level Language Software", IEEE Trans. Software Engineering, Vol.15, No.7, July 1989, pp. 875-889.*