2009 International Symposium on Computing, Communication, and Control (ISCCC 2009)
Proc .of CSIT vol.1 (2011) © (2011) IACSIT Press, Singapore

An Open Architecture through Nanocomputing

Joby Joseph1and A. Clementking 2+
[1]Head, Dept. Comp. Science, St. Anthony's College, Shillong, Meghalaya 793001, India
[2]Head, Dept. of MCA, Loyola College, Chennai, Tamil Nadu 620010, India

Abstract: In this paper we are developing the idea of an open computing architecture through nanotechnology. The features of open computing seem to be the ideal take off ground for designing a reconfigurable architecture built on nano elements. The alternative we propose to CMOS based computing is through Chemically Assembled Electronic Nanotechnology (CAEN). These can be used to realize a reconfigurable hardware. We then propose a computing architecture that is based on the theory of reconfigurability that takes advantage of innovations from the field of open source and open computing.

Keywords: Nano Computing, Open Architecture, Reconfigurability, Defect Tolerant Architecture.

1. Introduction

Open source software has gained lot of popularity and use in the current computational arena. This phenomenon has gained lot of interest and relevance in the fields of study and engineering as it allows modification and derivatives to the extent that the source remains still open. In such an open system the distinction between the 'designers' and the 'users' is replaced by a participatory role which ensures modification and improvement. Open source approach has come as a response to a need of the software community. In the presence of a large group of experts the open source allows innovation and modification that are driven my real expertise and on-the job necessity [1]. Just like open source software, any open system of computing will allow a great degree of interoperability, portability and open standards. Basing on the success and promise of the open source saga, we here proposing a software-hardware analogy, that allows an open source architectural phenomenon.

2. Open System of Computing

An open system of computing should include the elements of *Open Standards, Open Architecture* and *Open Services* that effectively accelerate and innovate technology. Let us see these in detail [2].

2.1. Open Standards

The Open Standards in such a system of computing is to ensure collaborative innovation, flexibility, interoperability, economy and freedom of growth. This is also the technical norm that describes the interface, protocol specifications and language specifications. So being 'Open' the standards must be:

- Made available to all with no restrictions,
- Regulated by an open industry with a well defined inclusive process for evaluating the standard
- Implemented and modified periodically by the standards available in the open market [3].

2.2. Open Architecture

If the open standard defines the technical norms for implementing the features of an Open System, then the *Open Architecture* defines the structure of such a system. The architecture within an open system will have to be both evolving and adaptive. The present day systems of computing will not stand up to this test.

+ Corresponding author. Tel.: + (919436337168); fax: +(913642229558).
E-mail address: (jobymj@hotmail.com).

So we need to look at some other models that are dynamically reconfigurable and adaptive to changes. So what matches to this most, is the nano computer architecture. We will see the proposed nano architecture in the pages to come.

2.3. Open Services

An open system of computing will have to tie together the various components that deliver services like putting together the building blocks of any structure. This must allow the various applications to communicate and interact with each other and bring out the service [4].

3. Moving towards an Open Nano Architecture

The transistor is the building block to a modern processor. The current silicon designed transistors are going to hit their physical limit- not merely the actualization of Moore's law- but also the problems with heat dissipation, wire connections and the materials we use to create them. Hence nanotechnology helps us to look at new ways information processing at a better speed and measure [5]. A promising alternative to the imminent challenges from the CMOS based computing, is to focus on other alternatives of nano scale precision. Chemically Assembled Electronic Nanotechnology (CAEN) is a promising technology, which uses self-alignment to construct electronic circuits from nano scale devices that take advantage of quantum mechanical effects. It uses chemical synthesis techniques to construct molecular-sized circuitry elements such as, resistors, transistors, diodes, resonant tunneling diodes (RTDs), and reconfigurable switches. Such fabricated devices are only a few nanometers in size and only tiny amounts of current are needed to operate such devices [6].

As CAEN devices are very small, connecting such components will be highly expensive process. Instead it requires that circuits be created and connected through a process of self assembly and self alignment. Several research groups have already demonstrated this. The process of self-assembly is an imprecise, thermodynamically controlled process and so we cannot for sure be certain of full reliability. Hence the devices must be testable, adjustable and possibly reconfigurable and it will also include some defects that are tolerable. Reconfigurability is therefore a key component of such nanoscale architecture. It uses large arrays of uniform, programmable logic elements and interconnects, which can be configured using electrical signals to perform required computations. Reconfigurability is commonly used to implement new circuits *post* manufacture as in Field Programmable Gate Arrays (FPGA). It has also been used to incorporate defective circuit elements into a working, high performance system as in the case of Teramac developed in 1990 by HP. [7].

4. Architectural Implications

The fundamental unit of the nano fabrication is called a nano block (Fig. 1). It has three sections: a molecular logic array where the functionality of the block is located; the latches, used for signal restoration and signal latching for sequential circuit implementation; and the I/O area used to connect the nano block to its neighbors.

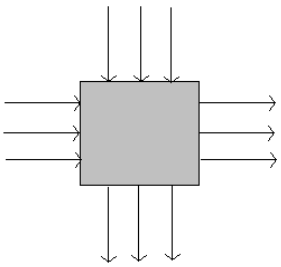

Figure1. Single Nanoblock with I/O lines.

Let us see briefly the plausible fabrication process of moving from basic components through self-assembled arrays of components to a complete system. In the first stage, the components (wires or devices)

are synthesized and connected together through chemical process of self assembly. In the next step, two planes of aligned wires are combined to form a two dimensional grid with configurable molecular switches at the cross points. These grids will be of the order of few microns. A separate process will then create a silicon based die providing power, clock lines, I/O interface and other support logics necessary. The grids are placed and aligned into the groves inside the die [8]. The molecular logic array portion of a nano block is composed of two orthogonal sets of wires. At each intersection of two connectors is a configurable molecular switch. The main element at the intersection of the grid will be a reconfigurable molecular switch, which can be configured 'off' or 'on'. The following diagram(Fig.2) shows how a 3x3 grid is configured to implement an AND gate.

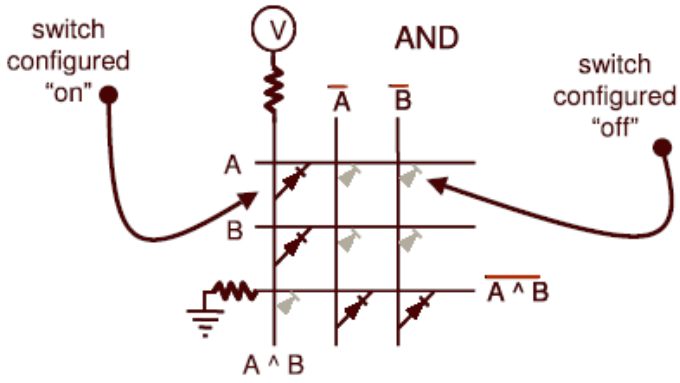

Figure 2. AND gate using reconfigurable CAEN grid

The nondeterministic nature of self-assembly will give rise to high degree of defects. Instead of eliminating these defects totally, we can perform some post-fabrication steps as done as in Teramac, to perform a self-diagnosis. The result of the self-diagnosis brings up a defect-map and the map can be used to configure the desired functions around the defects [9]. Such a system is dynamic, adaptable and evolving. And the paradigm that matches most is the open and reconfigurable mode of computing.

4.1. Defect Tolerant Hardware

Every nano fabric built into the system will have to be defect tolerant because they are regular, highly configurable, fine grained and has rich inter connect. The regularity of the defect allows us to choose where a particular function is to be implemented. The configurability allows us to pick which nano wires, blocks or parts of a nano block will implement a particular circuit. The fine grained nature of the device along with the local nature of the interconnect, reduces the impact of a defect to minimal. And finally the rich interconnect allows us to choose among many paths in implementing a circuit. Thus with defect map in hand we can realize and adapt function circuits. The defect discovery relies on the fact that we can configure the nano fabric to implement any circuit, by testing its own resources [10].

The key difficulty in testing the nano fabric is that it is impossible to test the individual elements in isolation. Hence, these will have to be tested in conjunction with an outside host(preferably a CMOS tester) as achieved in the case of Teramac. As the host gains knowledge of the fault-free regions of the fabric, it replaces more and more of the tester region with that of the nano fabric. Once a sufficient number of resources have been discovered by the host, these tested areas act as the host for the remainder of the fabric. Once the defect map has been generated we can use the nano blocks to implement necessary circuits. Though the molecules are expected to be robust over time, new defects that are inevitable might occur over time. This will not be as complex as the initial fault mapping [11]. Though the various demonstration system of this area, had their limitations, it is possible to build a computer that contains novel methods of detecting defects, like built-in ones and coupled with self-configuration mechanism the nano architecture can scale to new heights of computation.

4.2. Dynamic Reconfiguration

Reconfigurability is a key component to a nano architecture. The nano fabric uses runtime reconfiguration for defect testing and to perform its desired functions. The time that the fabric takes to configure depends on two factors: viz., first, the time it takes to download a configuration to the nano fabric and secondly, the time taken to distribute the configuration bits to the different regions of the fabric. [12].

The dynamic configurability of the nano fabric is chance to accommodate the defect prone manufacturing process. In addition, reconfigurable fabrics offer high performance and efficiency as they can implement hardware matching to each application. And if the configurations are created at compile time, one can get rid of complex circuitry. This opens up lot of possibilities for our topic of open computing:

- It can exploit parallelism at all levels of the application: instruction level, pipeline and bit-level.
- It can create customized functional units.
- It can use partial evolving and reconfiguration to reduce complexity of operations.
- It can eliminate a significant amount of control circuitry.
- It reduces sizeable amount of memory bandwidth requirements.

4.3. Putting it together

We have proposed the possibilities of reconfigurable CAEN elements as the basic building blocks of the new computing systems. This proposal brings up lot of challenges as well as possibilities. The future of nano computer architectures will stand out in its computing powers if fully realized as now visualized. The nano support at hardware level should enable high parallelism at multiple levels of instruction, data and threading. The system will be able to achieve a high level of dynamic reconfigurability both in space and time and also work on a path of high defects by routing around them.

Though still at its infancy, another novel development that assures lot of promises is the Programmable Artificial Cell Evolution (PACE) project. This large inter disciplinary research aims to create "nano-scale artificial protocells" that are able to self replicate and evolve under controlled conditions. These cells are intended to act as nano robots that can be used to encode information. If these become a reality, then we can programe them to build multi cellular and dynamically evolving computing systems. Such systems open a vast area of possibilities in computing. Like the study of *swam intelligence, Genetic Algorithms* and *artificial intelligence* [13].

5. The Open Reconfigurable Model Proposed

In this final section, we are proposing a novel model incorporating all the ideas proposed hereto. It is a 4-layer architecture building up to an open and dynamic system of computing. The architecture has the following properties of: configurable and defect tolerant hardware, capacity for operator and functional parallelism at multi layers and possibility of having adaptable software. The architectural diagram is shown in Fig. 3. The proposed model has four layers. The realization of the autonomous reconfiguration is based on two assumptions: Adaptation of the software coupled with the dynamic reconfiguration of the reconfigurable hardware. This is achieved by the careful coordination of the various layers. The configuration manager has the role of capturing real time data and decides on the configuration to be done and sends the information to the hardware fabric which performs the desired reconfiguration. The configuration manager will also involve itself in the process of learning and correcting reconfiguration decisions if needed [14].

In the next layer we have the OS kernels. Here we propose two levels of components: a real time OS kernel and a Configurable OS kernel. The former is to take care of real time critical operations that need fast time responses. The latter is required to control and assist with the mapping and dynamic reconfiguration of the hardware fabric. The development of such an OS kernel is area needing deep research. There are number of real time OS that come up to this requirement. And it is here that the Open source market has its catch. The open feature of the OS, like that of Linux, makes it easy to interface to software and hardware layers [15].

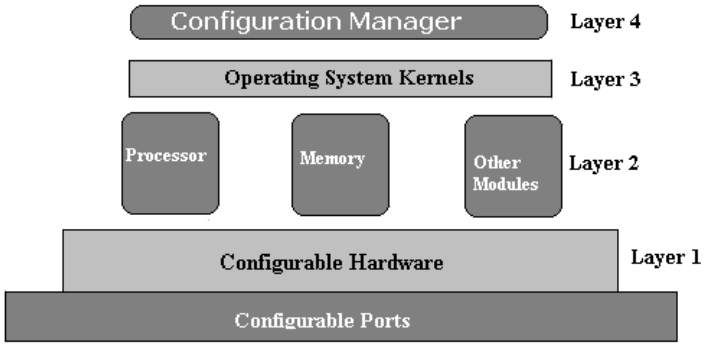

Figure 3 A Reconfigurable Architecture

The next layer is an interface between the top software part and the reconfigurable hardware structure. As so, there will be some amount of overlapping in design and function between this layer and next layer of hardware. This layer will have to bring together high end processor and memory modules that take advantage of nano fabrication to give high end processing power. In addition to that, the layer will also provide software support for the top two layers. The base layer is the evolvable hardware that can change its architecture and behaviour dynamically and autonomously by interacting with its processing environment. And the elements of Chemically Assembled Electronic Nanotechnology (CAEN) are sure candidates to satisfy this demand.

6. Conclusion

Having presented the entire theme of open computing through reconfigurable nano technology, let us make some final remarks. The world of Open Computing is widening. With the Open source market gaining popularity, it's the turn of Open hardware now to catch up. If reconfigurable nano elements can be built then the world of dynamic computing will be soon a realization. A reconfigurable architecture will need more research focus so as to answer to the problems faced from fault tolerance and dynamic reconfiguration. As explained above the field of dynamic open computing can become the paradigm for complex problems from the research fields of Genetic algorithms, swam and artificial intelligence.

7. References

[1] http://rixc.lv/ram/enpublic02.html

[2] http://www.ibm.com/developerworks/products/newto/open.html

[3] http://www.ibm.com/developerworks/products/newto/open.html

[4] http://www.ibm.com/developerworks/products/newto/open.html

[5] Richard Booker and Earl Boysen, Nanotechnology, Wiley Publishing, Inc. New Delhi, 2005.

[6] http://reports-archive.adm.cs.cmu.edu/anon/2002/CMU-CS-02-181.ps

[7] http://reports-archive.adm.cs.cmu.edu/anon/2002/CMU-CS-02-181.ps

[8] http://www.cs.cmu.edu/~phoenix/papers/isca01.pdf

[9] http://www.cs.cmu.edu/~phoenix/papers/isca01.pdf

[10] http://www.cs.cmu.edu/~phoenix/papers/isca01.pdf

[11] http://www.cs.cmu.edu/~phoenix/papers/isca01.pdf

[12] http://crpit.com/confpapers/CRPITV6Beckett.pdf

[13] http://www.cs.ucl.ac.uk/staff/B.Kaewkamnerdpong/bkpb-ipmm05.pdf

[14] http://klabs.org/mapld05/papers/254_papachristou_paper.pdf

[15] http://klabs.org/mapld05/papers/254_papachristou_paper.pdf