OpenHW designs for configurable computing

Jamil Khatib

OpenCores.org,
khatib@opencores.org
www home pages:
http://www.opencores.org
http://home.jamilkhatib.net

Abstract. Programmable logic devices and Hardware Description Languages have made great impact on Hardware design methodology. The programmable logic designs are getting closer to software specially when Run time reconfiguration designs are considered. ***OpenHW*** designers found that these devices provide them with risk-free implementation and testing for their designs. They even have the possibility to implement full computer using ***OpenHW*** designs downloaded from the Internet range from simple IO controllers and up to powerful CPUs all on programmable components. Such efforts will start new era of system design where all hardware design will be as simple as software in every system.

1 Introduction

The complexity of Hardware designs has increased in the last ten years to provide more integrated functions and performance. Designers have started investigating new design techniques and technologies for that reason. The Hardware description languages "HDL" are becoming very popular for digital design and modeling. They are used in a software programming style to develop large chips and IP cores. The programmable logic devices provide more software-like flexibility to designers since they can change the hardware as simple as software recompilations. The ease of hardware design using HDLs and its implementation and modification using programmable logic devices have attracted many engineers around the world to develop their own home made designs. Even more they adopted the Linux operating system[13] model by publishing their designs for free over the Internet.

2 ***OpenHW*** definition

OpenHW is a new design methodology that is based on publishing all information about the hardware implementation, how it can be interfaced to other systems and how it can be used. Usually all these information are made available for free without any restriction. There are several groups that try to define the Open Hardware in terms of the available information such as OpenCollector [11] and OpenIPCore[8] organizations. One of their definition terms is the availability of the hardware design files which includes but not limited to HDL, schematic, layout files and all information needed to reproduce the design. [2] The Open Hardware Certification Program[10] also requires that "the manufacturer makes a set of promises about the availability of documentation for programming the device-driver interface of a specific hardware device".

3 *OpenHW* designs

The idea of ***OpenHW*** designs was started by some enthusiastic engineers who have extra time to show their experience. They made lot of designs through cooperation over the Internet. These designs are mainly IP cores based on HDL languages. Some projects are targeted to board design for CPUs and prototyping for cores. The article *Free chips for all*[1] mentioned many examples of ***OpenHW*** projects.

The ***OpenHW*** is not just work for amusement, but it has lot of advantages[2] which will lead to powerful designs resulted from the contribution of many people around the world with different fields of knowledge. This urged some companies to follow this trend like Sun microsystems[14] that published the source files of its PicoJava and Sparc processors to get as many comments and reviews from engineers around the world.

One of the main problems that faces ***OpenHW*** designers is the implementation of their designs because hardware requires a large initial investment in time, and money to set up a production run for something[6]. Some of the designers have satisfied with simulating the designs. The luckiest ones got some support from companies to build their designs, such as OpenCores organization[9, 4]. This problem does not prevent designers from making more designs. They start looking for cheap way to implement and test their designs. The configurable computing devices such as FPGAs (Field Programmable Gate Arrays) and CPLDs (Complex Programmable Logic Devices) are becoming more stable and can hold large number of physical resources. Such devices are suitable for ***OpenHW*** designers because they can fit wide range of designs and can be easily modified if bugs are found which make risk free implementation. Moreover, they can be programmed and tested at home using not so expensive prototyping boards.

4 Software and hardware comparison

The idea of open source has succeeded for software because it can be easily reproduced and distributed over the Internet[6]. The concept of configurable devices and Hardware Description Languages (HDLs), makes the hardware more and more closer to software. Designers can develop hardware using traditional software programming by HDLs. The result of these programs are compiled to

produce the hardware interpretation of the initial description. This interpretation can be programmed directly to the PLDs to produce the real hardware.
This advancement in the hardware design strategy attracts the hardware designers to produce open source free hardware designs where hardware can be free in this way just like free software when all HDL files are made free and targeted to FPGAs. This is instead of wiring components and soldering resistors for free. The FPGA prototyping boards will act as the PCs do for software, and HDL codes which describe IP cores can be made available through the Internet, like what OpenCores organization[9] is doing. These codes can be downloaded to FPGA plug-in boards attached to PCs to add extra functionality or accelerate some. In this case the users will be able to get latest updated hardware for their system without the need to change every thing.

5 Dynamic Reprogrammable computing

Some of the programmable devices have the option to be dynamic and partial programming. This means that parts of the internal logic can be modified while the system is working, Xilinx's Vertix and XC6200 families support such options. These features provide many advantages to the system such as increasing the density and performance of the hardware on FPGAs[3] through dynamic loading and unloading of modules.

6 Future computers

The concept of programmability of chips is not limited to digital circuits, but there are even analog components such as FPAA[20], TRAC[21] and ispPac[22]. All kinds of programmable devices with the aid of ***OpenHW*** designs will push towards new technologies and design methodologies.

Computers for example will change from the traditional architecture that is based on Software that executes on hardware resources, into algorithms that run on hardware. So the future PCs will be built of boards that contain FPGAs and connectors to external world. The hardware designers will be able to code modules (cores) for specific FPGAs computing machines (Future PCs).

The Dynamic Instruction Set Computer (DISC)[5, 3] CPU for example is based on variable instructions that can be customized for each application by the algorithm implementers and even can be dynamically loaded to the hardware when they are needed. The ***OpenHW*** designers can take the advantage of this architecture and build all of their hardware (algorithms) to act as software libraries and even Operating systems. Linux[13] the free open source operating system for example can run natively (a pure hardware) on hardware if targeted to dynamic reprogram-able machines.

7 Future Internet

Internet, the basic platform for ***OpenHW*** designers, will be affected by the concept of dynamic reconfiguration and ***OpenHW*** concepts. FPGA boards can be connected to the Internet to act as free computing and debugging machines. I think in the future new service can be added to the Internet that is called Hardware Computing Resource Protocol or *"hcrp://"*. Such service will enable people to design their algorithms (based on hardware cores) and upload them to these machines to run them. These services can enable companies to show real execution and customization of their hardware on line.

Even more when hardware is connected to the Internet it will be easy to update it with new algorithms and add new features or fix existing bugs. This will enable companies to ship their hardware faster than before and ***OpenHW*** designs will find their way to home users. This is because the designs will be tested in real life applications and bug fixes will be as easy as file download from the net. Some companies such as Xilinx[15] thought about Internet related issues. It has program called *Internet Reconfigurable Logic*, IRL that provides online update of hardware. Such programs will push towards Home Networking and easily control and update all home devices such as VCR, TV and even wash machines.

All these changes will make open source software and open source hardware developers much closer together and increase their productivity for single platform, the programmable hardware.

8 Tools and testing

OpenHW designers need tools and compilers to develop their designs. They depend on free tools like gEDA[16] and Alliance[17], others depend on tools donated to them by companies such as in OpenCores organization[9, 4]. Online web based tools can help these designers because the tools will be placed on the same location where they do their discussions and reviews. Moreover, the new Internet Hardware computing Resource Protocol can help the designers to test and debug their designs.

Recently computer based test equipments have evolved and become able to cover wide range of test parameters. These equipments can take the advantage of Internet reconfigurable concept by plugging them to the Internet and connecting them to system under test. Both the test and system under test can be configured and controlled over the Internet. In this way hardware designers will have access to high performance computing and test equipments.

Reconfigurable computing machines need also new tools to support the dynamic reconfiguration. Many researches led to new tools such as JHDL and JBits. JHDL[3, 19] is a free java based tool for FPGA design. This tool is a result of BYU researches. JBits[15, 18] is the only commercial available tool that supports dynamic and partial reconfiguration of Xilinx Vertix FPGAs using standard Java functions. Xilinx use this language in its Internet Reconfigurable Logic via standard java applications. These tools are based on Java language which is the most popular software language with Internet support. The choice of such language came to follow the Internet trends and to be easy on software programmers to develop hardware modules. Again, this will remove the difficulties between hardware and software and attracts more ***OpenHW*** and opensw projects for programmable devices.

9 *OpenHW* in business

The programmable computing devices producers care much about selling their devices, so they provide all necessary services to support devices. For example they provide some EDA tools design flow for free to reduce the overhead of using the devices in systems. For example Xilinx provides the Webpack tool and Altera provides some OEM synthesis tools.

The ***OpenHW*** concept will aid the manufacturers of programmable devices because as described earlier these devices are the most suitable platform for ***OpenHW*** designs. More prototyping and test boards based on programmable devices will be sold to these communities and it will push towards more devices in every future PC.

Moreover ***OpenHW*** concept has more business models[7, 2] such as:

- **Design distribution:** where companies can pack set of designs and sell the distribution just like Linux os distributions. OpenTech[12] cdrom is an example of this method. This project is one of OpenCores projects that compiles open source EDA tools and open source HW designs.
- **Design support:** experts can give support for ***OpenHW*** designs.
- **Design implementation:** companies can implement the designs and sell them.

In short customers for the hardware products will be the same suppliers of the intellectual property[6].

10 *OpenHW* The road map

The ***OpenHW*** road map can be divided into three main stages. If we can follow these stages we will have new industrial and technological revolution.

1. **Simple prototyping boards:** A set of small generic prototyping boards can be designed to test small IP cores. The designers will use the available free tools which give lot of feedback to the EDA developers.
2. **Advanced boards and IP Cores:** Designers will publish ***OpenHW*** cores and ***OpenSource*** software for the boards. The boards will be much advanced in this stage and new companies will emerge that depend on selling the fabrication of the ***OpenHW*** boards. So anyone can download the design files from the Internet and implement it on one of the prototyping boards. At the end of this stage FPGA and CPLD companies will publish the programming technique or provide free tools for their devices.

3. **Run time Configurable logic systems:** will be more popular in this stage where relation between the software and hardware will change. The user will not feel if the program he is running is executing instructions of the host PC or if new hardware is being created on the board. The use of this idea will produce new free software-hardware compilers and new design methodologies.

11 Conclusion

We will see in the future people working and kids playing games on "software" that make the full use of the available hardware. This hardware will be built on programmable logic devices that will change how the Hardware can be implemented and designed that can be handled by everyone. The open hardware design concept will make the best use of this new technology and will push towards new business strategies for programmable logic. Even open source developers will join the open hardware trend because there will be no difference between software and hardware when dynamic reconfiguration of logic is considered. At last the open hardware developers will push towards an FPGA in every PC.

References

1. Jamil Khatib. Free chips for all. In IBM developers work site. "http://www-4.ibm.com/software/developer/library/openhw.html"
2. Jamil Khatib. OpenHW design trend : A survey. In the author's site "http://www.geocities.com/SiliconValley/Pines/6639/docs/OPENIP.PS"
3. Jamil Khatib. Configurable computing. In the author's site "http://www.geocities.com/SiliconValley/Pines/6639/fpga/rtr.html"
4. Peter Clerk. Momentum builds for open-source processors. Article in *EETimes*. "http://www.eet.com/story/OEG20010201S0050"
5. M. J. Wirthlin and B. L. Hutchings. A dynamic instruction set computer. In *Proceedings of IEEE Workshop on FPGAs for Custom Computing Machines*, pages 99-107,Napa, CA, April 1995.
6. Industry Analysis Paper "http://web.dodds.net/h̃ozer/opensource.html"
7. Business Models for Open Source Hardware Design. "http://pages.nyu.edu/g̃mp216/papers/bmfosh-1.0.htm"
8. OpenIPCore Project site "http://www.opencores.org/OIPC/"
9. OpenCores site "http://www.opencores.org"
10. The Open Hardware Certification Program. Sponsored by the FSF, Debian, and FreeBSD. "http://www.debian.org/OpenHardwre"
11. OpenCollector site "http://opencollector.org"
12. OpenTech cdrom project "http://www.opencores.org/OIPC/projects/OpenTech"
13. Linux Operating system "http://www.linux.org"
14. Sun Microsystems "http://www.sun.com"
15. Xilinx Inc. "http://www.xilinx.com"
16. gEDA site "http://www.geda.seul.org"
17. Alliance site "http://www-asim.lip6.fr/alliance"

18. Jamil Khatib Jbits' site "http://www.geocities.com/SiliconValley/Pines/6639/fpga/jbits.html"
19. JHDL project's site "http://www.jhdl.org"
20. FPAA components. "http://www.anadyne-micro.com"
21. TRAC components. "http://www.fas.co.uk/"
22. ispPac components. "http://www.latticesemi.com/products/devices/isppac.cfm"