International Journal of Research
Available at https://edupediapublications.org/journals
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 03 Issue 11
July2016

Virtual Prototyping a Better Approach to reduce product Time to Market

Rashmi
PG Scholar
Dept. of ECE, SJCE, Mysuru
Email: rashmi.saligrama@gmail.com

Thyagaraja Murthy A*
Associate Professor
Dept. of ECE, SJCE, Mysuru
Email: trm.sjce@gmail.com

Abstarct

With the advancements in technology, the number of functionalities are being integrated into SoCs are increasing rapidly. A typical SoC consists of, microprocessor, microcontroller or Digital signal processor (DSP) core, and the multiprocessor SoCs (MPSoC) [4] are having more than one processor core. Along with processor, the SoC will also have memory blocks including RAM, ROM, EEROM and flash memory, timing sources including oscillators, other peripherals including real-time timers and counters, power-on reset generators. SoC will also contains external interfaces including USB, Ethernet, FireWire, also analog interfaces including ADCs and DACs, Voltage regulators and power management circuits, and a bus to connect all these blocks all on one chip. Along with the hardware SoC also contains the software to control these hardware. The below mentioned are some of the key activities in any product development flow.

- System engineering
- Map customer requirements to design features
- Optimizing the design to meet the requirement in the best possible way
- Design the hardware
- Design, develop and integrate the different hardware modules
- Develop reference modules to validate the HW modules developed
- Design, develop and integrate different software modules
- Develop reference modules to validate SW modules developed
- Integrate hardware and software
- Validate the system
- Build the system
- Port software onto the system
- Validate the system
- Customer Delivery

Traditionally, these activities are sequential in nature, software modules used to be developed only after developing the hardware modules, this would reduce the time to market, because the SW development is completely dependent on the HW. The concept of Virtual Prototyping is to create the software model of the entire hardware, [1] the software will be developed

based on these model, and also the HW modules will be developed based on these virtual modules, since both the HW and SW will be developed in parallel it will reduce the time to market of the product.

Key Words: TTM, Virtual Prototyping, SoC, FPGA Prototyping, RTL Simulation

IIntroduction

The integration of increasingly complex hardware and software is a significant challenge for semiconductor and OEM companies developing next-generation wireless, consumer and automotive devices. Traditional methods of serialized hardware and software development–where the vast majority of software is developed and verified after the silicon design is complete–often fail to meet aggressive product development schedules [7].

Virtual prototyping results in faster time-to-market through earlier and faster software development and improved communication throughout the supply chain[2]. They enable software engineers to start development months before the hardware design is complete, enabling full system bring-up to occur within days of silicon availability. Virtual prototypes are fast, fully functional software models of complete systems that execute unmodified production code and provide unparalleled debug efficiency.

With software development becoming the fastest growing component of NRE [8] costs for both SoC and final product development, the challenges of developing, integrating, validating, and optimizing software in the context of complex hardware architectures are dominating the embedded design process. Thus it has become a necessity to make a fast, accurate, and low-cost simulation model of the hardware available early to the embedded software team.

II Virtual Prototyping

Virtual prototyping, often known as VP, is a software-based engineering disciplinewhich involves modelling a system, simulating and visualizing its behavior under real-world operating conditions, and refining its design through an iterative process.The integration of increasingly complex hardware and software is a significantchallenge for semiconductor and OEM companies developing next-generation wireless,consumer and automotive devices. Traditional methods of serialized hardware andsoftware development–where the vast majority of software is developed and verified afterthe silicon design is complete–often fail to meet aggressive product [11] development schedules.

Virtual prototyping results in faster time-to-market through earlier and fastersoftware development and improved communication throughout the supply chain[3]. Theyenable software engineers to start development months before the hardware design iscomplete, enabling full system bring-up to occur within days of silicon availability. Virtualprototypes are fast, fully functional software models of complete systems that executeunmodified production code and provide unparalleled debug efficiency.

VP is increasingly used as a substitute for rapid prototyping. VP does not producea physical object for testing and evaluation but, as its name suggests, carries out these taskswithin a computer.

IIIVirtual Prototyping to reduce Product TTM

Before multiprocessor system-on-chip (MPSoC) architectures became so complex,the hardware and software components of an embedded system were designed sequentially.In other words, the software engineers did not begin development of the operating system,device drivers, and inter-processor communication protocol stacks until they had a verysolid hardware prototype on which to base their work as shown in Figure 1

Cost of Being Late

The traditional approach to system development is to design the hardware, make a physical prototype, write the code, and then integrate the hardware and software. Because many projects target fast-moving markets, this approach is now too slow and too risky. Athree-month delay on a product with a total lifetime of 30 months can reduce profits by50% [9].What many software teams want is to start developing the software as soon aspossible in the project lifecycle. That means finding an alternative to the traditional‘hardware-then-software’ design flow and getting started before the hardware is ready

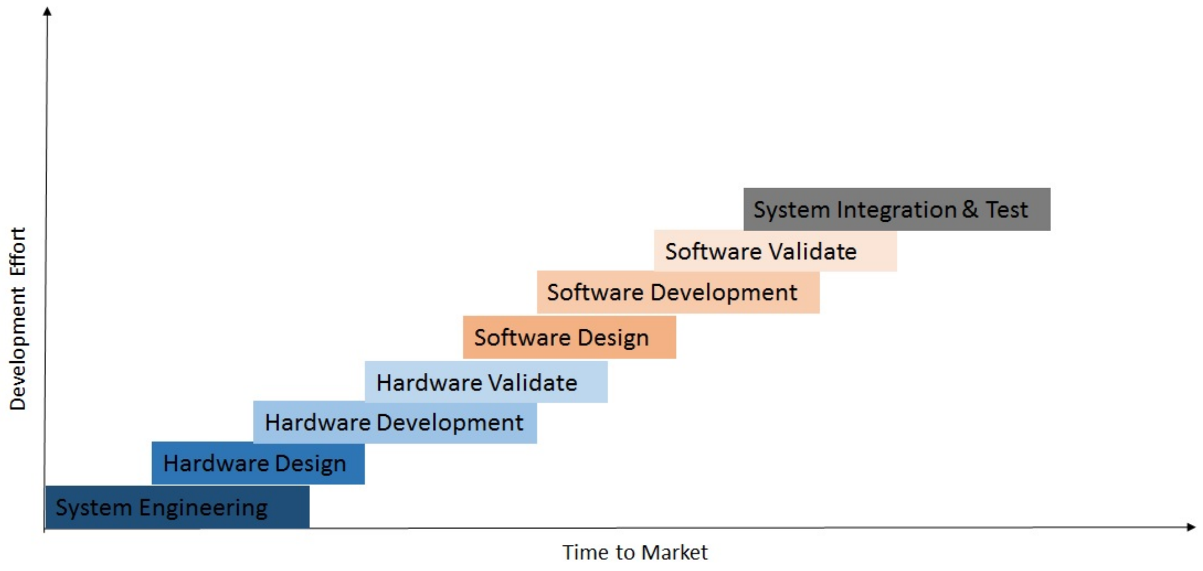

Figure 1: Traditional hardware-then-software embedded project designflow

Enabling an Early Start on Software

p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 03 Issue 11
July2016

Virtual prototyping allows a software team to start its development efforts evenbefore the hardware team has begun writing RTL [5]code. This approach can lead to a nine to12-month market advantage, which for a growing number of development teams isimpossible to ignore. And many find that the secondary benefits of virtual prototyping aremaking their teams so much more productive and efficient – for example, duringdebugging. Virtual prototypes offer excellent control and visibility for the purpose ofdebugging and analyzing the behavior of the software. In fact, the debug capabilities arefar better than those available when using the hardware itself. Virtual prototypes offer otheradvantages in addressing multicore designs and in enabling globally distributed teams towork together [6].

Virtual prototypes can be made available just a few weeks into the project schedule,which allows the software team to begin porting operating systems and developing devicedrivers without having to wait for the hardware team to write a single line of RTL [11] code.Figure 2 shows how a "parallelized" flow leads to a significant schedule improvement.

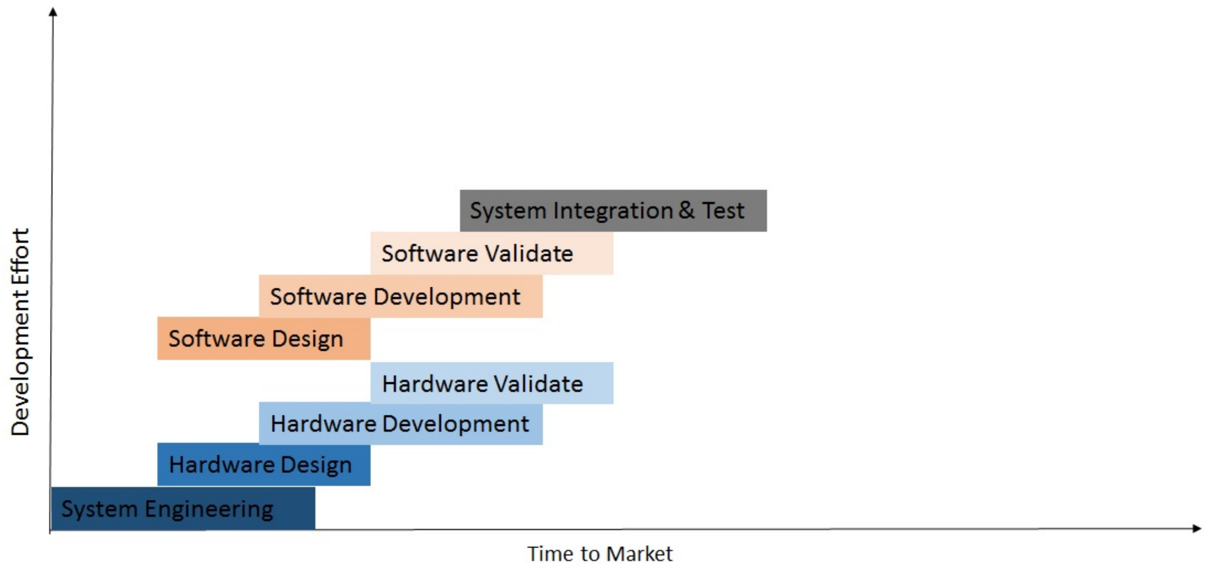

Figure 4.2: Parallelized hardware-software design flow using virtual prototypes

Increasingly, semiconductor companies are looking for ways to differentiate themselvesbeyond the feature sets of their chips. By adopting virtual prototyping they can offer bettersupport to their systems and OEM customers.

IV Advantages and Benefits of Virtual Prototyping

- A virtual prototype is a fully functional software representation of a hardware design that encompasses virtually any combination of hardware. For example, asingle- or multicore SoC, multiple SoCs, microcontrollers, peripheral devices, I/O,and optionally a model of the user interface.
- It provides an unambiguous executable specification of the system design, whichdevelopers can use to develop, integrate and test the software.
- A virtual prototype runs on a general-purpose PC or workstation and is detailedenough to execute unmodified production code, including drivers, the OS,middleware and applications at speeds close to – or even faster than – real-time.
- Enables a reduced time to market.
- Allows for early testing.
- Can conduct expensive or impossible tests.
- Reduces the need for a physical prototype.
- Improves operator safety and comfort.
- Removes geographic boundaries.
- Provides a common design standard and language.
- Protects profit margin. Increases company agility.
- Reduces development costs.
- Reduces the scope and scale of engineering changes.
- Engenders a right-first-time attitude.
- Unravels design complexity.
- Enables full participation by all interested parties in the product-developmentprocess.

Developers can refine the virtual prototype as the project progresses, for example, extending the virtual prototype based on the software design task. To support an initial OS port will require the ISS and a few key peripherals such as a UART [10] and timer. The development team can add more complex peripherals to support the needs of the schedule.

- Virtual prototypes provide more visibility and controllability for the software developer. This functionality must come with virtual prototype specific development tools covering system-level software debug and analysis as well as the combination of hardware-software analysis. They must also integrate with existing software tools such as debuggers, which should be able to synchronize with the virtual prototyping software tools.
- Virtual prototypes should also integrate with the hardware verification and system validation flows. These flows use additional tools such as FPGA prototypes, RTL simulation, physical system simulation, test bench equipment and even virtual IO that enable the virtual prototype to connect to physical hardware, such as USB or Ethernet.

Conclusion

Virtual Prototyping provides an early accurate functional model of the hardware to software engineers even before the hardware is available. It can run software on embedded processor models at speeds on-par with actual hardware. Virtual

Prototyping provide additional capabilities and benefits, such as the visibility and control to debug complex software/hardware interactions and the ability to optimize the software to meet final product performance and power goals, hence it reduces the Product Time to Market.

References

[1] Tse-Chen Yeh, Zin-Yuan Lin and Ming-Chao Chiang, "Optimizing the Simulation Speed of QEMU and SystemC-based Virtual Platform"
[2] Prasad AVSS, Sasidharan Prasant, Rajiv Jain "Virtual Prototyping Increases Productivity - A Case Study" 978-1-4244-2782-6/09/$25.00 ©2009 IEEE
[3] Màrius Montón, Antoni Portero, Marc Moreno†, Borja Martínez, Jordi Carrabina, "Mixed SW/SystemC SoC Emulation Framework" 1-4244-0755-9/07/$20.00 '2007 IEEE
[4] Wayne Wolf, Fellow, IEEE, Ahmed Amine Jerraya, and Grant Martin "Multiprocessor System-on-Chip (MPSoC) Technology", IEEE Transactions On Computer-Aided Design Of Integrated Circuits And Systems, Vol. 27, No. 10, October 2008
[5] Weiwei Mao, Ravi K. Gulati, "Improving Gate Level Fault Coverage by RTL Fault Grading*", INTERNATIONAL TEST CONFERENCE, 0-7803-3540-6/96 $5.00 Q 1996 IEEE
[6] http://www.synopsys.com/prototyping/virtualprototyping/Pages/default.aspx
[4] https://en.wikipedia.org/wiki/Virtual_prototyping
[8] https://en.wikipedia.org/wiki/Non-recurring_engineering
[9] http://www.arenasolutions.com/resources/articles/time-to-market/
[10] https://www.freebsd.org/doc/en/articles/serial-uart/
[11] https://www.mentor.com/esl/vista/virtual-prototyping/