Combining Emulation And Virtual Prototyping - The Whole Is Greater Than The Sum Of Its Parts

Tomer Naftali

Intel Corporation
Haifa, Israel

www.intel.com

ABSTRACT

As big chip companies put more focus on shortening products' time to market the usage of big box emulation for pre-silicon activities increases. HW Validation teams and SW development teams are exploring the usage of emulation for RTL bug finding as well as post-silicon readiness and new use cases are targeted for execution on emulation. Emulation has many benefits but also limitations. For complex and dense designs emulation can reach speed of several hundreds of KHz. As this is sufficient speed for many use cases, it is not realistic to boot an Operating System (OS) with production BIOS on the emulated HW at these speeds. Combining emulation with Virtual Prototyping (VP) creates a hybrid platform that enables booting Windows OS with production BIOS 350X faster than SLE. This opens the door for a wide range of new use cases to execute on emulation, where OS is a prerequisite.

Table of Contents

1. Introduction 3
2. Motivation 3
3. Technical Challenges 3
4. Results 5
5. Pre-Si Platforms Continuum Strategy 6
6. Conclusions 6
7. References 7

Table of Figures

Figure 1 - Hybrid SLE platform partitioning diagram 4

Table of Tables

Table 1 - Pre-Silicon platforms summary 4

1. Introduction

This paper describes how we combine emulation with a virtual platform (VP) in order to enable use cases not possible with emulation only. The first chapter will describe the motivation of building such a platform. The second chapter will describe the technical details and challenges of creating this platform. The final chapter will describe our results and how this platform initiative relates to Intel pre-silicon platform continuum strategy.

2. Motivation

As big chip companies put more focus on shortening products' time to market the usage of big box emulation for pre-silicon activities increases. Our focus is not only to validate a full system RTL, including CPU and Chipset, but to validate a full platform including the FW/SW integration with the HW. For this type of HW/SW co-validation we target validation of the production SW stack, making sure that we maximize pre-silicon FW/SW check-out. On top of the production SW stack, including BIOS and FW, we want to boot OS and enable OS based validation and driver validation use cases.

Achieving these validation goals on System Level Emulation (SLE) is not possible due to the limited runtime performance. Although SLE reaches several houndreds of KHz on ZSE1 emulators, running the production SW stack can take up to two weeks to reach OS boot. Such long run times are not usable for use cases which require the OS boot as a prerequisite for other validation tasks.

In order to enable booting the production SW stack on our HW system at pre-silicon timeframe we developed the Hybrid SLE. The Hybrid SLE (HSLE) combines emulation with VP for building the full system. By integrating the VP Cores with the rest of the SoC on SLE we can take advantage of the VP code execution speed and reach much higher effective performance of the system. With this setup we successfully booted Windows OS with production SW with the SoC RTL at reasonable times, reaching effective performance boost of X350!

As OS boot time gets reduced by 350X achieving short enough overhead that opens the door for use cases where OS is a prereiusite, among them is OS based validation. Combining SLE with VP creates a system which is greater than the some of its parts and brings significant benefits for FW/SW validation and development teams.

A secondary motivation to enable HSLE is the reduction of emulation model size for bringing up a full system. By removing the Cores, which are now implemented by VP, we were able to reduce emulation model size by 33% and still build a full system.

3. Technical Challenges

In order to be able to combine SLE and VP both platforms should be enabled and mature enough for performing the validation tasks required of the hybrid platform. At Intel we use both SLE and VP for different pre-silicon tasks, each platform has a large number of users and use cases utilized on them. The following table summerizes the benefits, drawbacks and use cases utilized on the different platforms for shifting left validation and development tasks.

	Virtual Platform (VP)	Emulation (SLE)	Hybrid Emulation + VP (HSLE)

Properties	High speed functional simulator Good debug-ability Early availability	RTL accuracy Good debug-ability Longer integration time	RTL accuracy + functional accuracy Great debug-ability Less expensive by ~33% (#boards)
Drawbacks	Not the real HW	Low runtime speed vs. VP/FPGA Expensive ($)	Single outstanding transaction from Core
Use Cases	Software and Firmware development	RTL development and validation	HW/SW Co-validation System use case validation

Table 1 - Pre-Silicon platforms summary

In order to boost code execution, utilizing the VP benefits, we keep the Core and memories on the VP side. This platform partitioning minimizes accesses to SLE reducing it to accesses required to configure and activate IPs on the SoC SLE side and not on code or data fetches required for execution of the code. Execution slows down to SLE speed only when SoC IPs are accessed, the rest of the time we run at VP speed which gets up to 10 MIPS. In addition to memories we also map the user interfaces to VP for the same speedup reasons, so keyboard, mouse and display are also running at VP speed. When the content is CPU centric, like OS and BIOS boot, we get significant boost in effective runtime performance.

With the above motivation we can see the platform partitioning between VP and SLE in the following diagram:

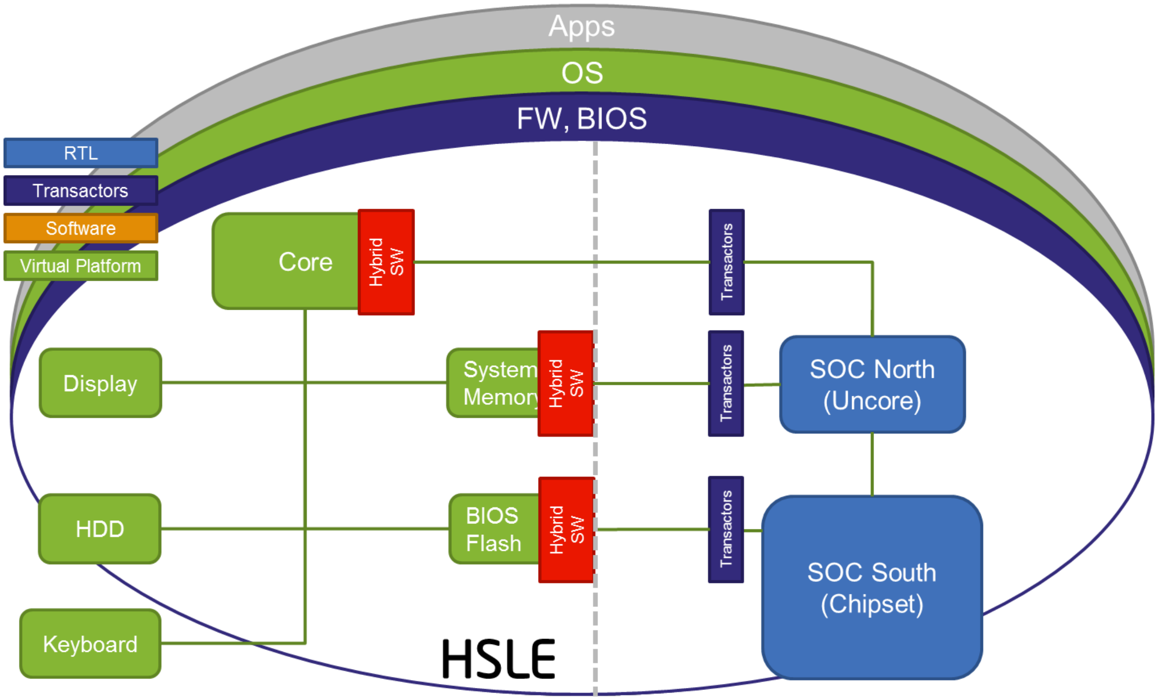

Figure 1 - Hybrid SLE platform partitioning diagram

The hybrid SW is the bridge between SLE and VP. Its main purpose is to bridge the gap between the different abstraction levels. VP is a high level architectural simulator and SLE is implementing RTL at full low level accuracy and details. As the diagram above depicts transactors are playing a significant role in the hybrid connection. Their main purpose is to bridge between HW signal level protocols to SW consumable transaction level modeling (TLM). Using a dedicated transactor per HW interface on the hybrid partitioning boundary we can connect RTL with SW.

In addition to the basic gap between signal level and TLM the hybrid SW takes care of handling micro-architectural handshakes and flows that the VP is not aware of. A basic example is the getting-out-of-reset sequence where HW goes through multiple steps to wake up the SoC IPs but VP wakes up immediately and assumes it can start executing code.

Towards the VP SW the hybrid SW layer makes sure to implement all the SW interfaces utilized in the pure VP environment in order to seamlessly replace native VP components with the gateway to SLE. This is a translation layer bridging the VP side SW interfaces and the transactors SW interfaces, each protocol with its own language.

In order to keep memory coherency between what the Core executes from and what the SoC IP's may access in DMA cycles we also connect SLE and VP on the DDR system memory interface, utilizing the DDR transactor for this connection. We basically redirect all memory accesses from both VP and SLE to a single instance of system memory accessible from both platforms.

A secondary benefit we want to exploit from this platform is the reduction of emulation model size required for bulding a full system. For this purpose we're taking the CPU full-chip and removing the RTL Cores from it. Instead of the RTL Cores we integrate transactors on the different HW interfaces between the Core and the rest of the SoC. Removing the Cores from the emulation model reduces the emulation model size by approximately 33%! With a set of automation scripts we simplify diriving the HSLE model from the full SLE model, reducing the effort required for enabling this model to a minimum of a couple of days on top of the SLE build.

Being able to enable this new use case does not come without a price. There are several validation coverage compromises taken when working with HSLE. This is the reason HSLE is a complimentary solution to SLE and does not replace it altogether. It is important that HSLE platform customers are aware of these limitations and choose the appropriate validation test cases to exercise on the platform. The validation coverage compromises include the following: 1) a SW simulated Core is used instead of the real HW meaning we're not validating the Cores under the new use cases 2) by using an architectural level simulator the interaction between Core and SoC is missing the micro-architectural features that the real Cores implement for example cache-ability attributes are excluded 3) as VP is a SW simulator each access it performs towards the HW must complete before the next operation is performed resulting in a single transaction at a time missing the parallel behaviour of the real HW.

4. Results

We successfully enabled the HSLE platform with Intel CPU and Chipset products in their pre-silicon timeframe during 2014. This HSLE platform boots Windows 8.1 OS running production BIOS with minimal modifications – achieving X350 effective performance boost!

By removing the Cores we reduced the emulation model size by approximately 33%! This enables better utilization of the available emulation capacity, meaning more platforms with the same budget investment.

Reducing emulation model size can potentially increase emulation runtime performance, but we did not gain significant boost in this domain and will explore this further in future products HSLE platforms.

The HSLE platform was utilized by OS based validation teams to execute Windows based validation applications testing several system level features, as well as exercising reset flows. We successfully enabled new use cases with emulation of the SoC RTL not possible before with SLE only. This platform opens the door for wide range of new validation use cases to be executed pre-silicon.

5. Pre-Si Platforms Continuum Strategy

Building a successful HSLE platform can only be achieved if the VP is accurate and mature enough to replace the RTL IP blocks and supporting OS boot. If we are able to integrate different IPs from VP with emulated RTL we can utilize them at different stages of pre-silicon validation.

Taking this step back allows us to view the pre-silicon validation process as a continuum where each incremental point is near the previous. The continuum means that we want to provide continuous capabilities across RTL development and validation, performed with RTL simulation or emulation, Software and Firmware development, performed with pure VP, and HW/SW co-validation, performed with HSLE. Although each platform is utilized for different use cases and usage models and have very different environemtns we want to be able to reuse as much as possible between them and reduce the additional effort per platform. Reuse across platforms includes reusing test bench components and even test content. This is an ongoing process that we're looking to integrate into our workflows at Intel.

Internally we're applying techniques to reuse our investments in building a functional accurate VP for building HSLE and SLE plaforms more efficiently. But in a wider perspective we need the collaboration from our partners and vendors to supply the tools and methods to support the continuum. For example: 1) using the same tool for building RTL for simulation or emulation to reduce model build efforts 2) vendors providing VP models with purchased collaterals or SoC IPs to reduce VP development efforts 3) providing infrastructure to reuse UVM environments from simulation to emulation for reducing content development efforts, and others.

6. Conclusions

Combining emulation and VP enables new use cases not possible with SLE alone enhancing HW/SW co-validation. With the described platform partitioning between emulation and VP we achieved X300 boost in effective performance and reduce 33% in emulation model size along the way. We have seen that when combining emulation and virtual platform the whole is greater than the sum of its parts. HSLE is an excellent solution to compliment SLE features and validation teams must select the test cases to be exercised on each platform, taking its advantages and drawbacks into account.

Reusing VP components together with SLE either as validation collaterals or for replacing SoC IPs for building HSLE platforms allow us to remove duplications and reduce development efforts when looking at the pre-silicon platforms as part of a continuum we apply for validating our product to reduce time to market.

Following the successes we had with this platform we continue to develop HSLE platforms for current CPU products in the pre-silicon time frame. Taking it further focuses us on bring up efficiency, performance boost and enabling additional use cases on the HSLE platform.

7. References

[1] Need for a Presilicon Continuum. Bill Hodges, Intel Corporation. SNUG 2014, Austin Texas USA

[2] What Intel Needs from Pre-Silicon Prototyping. Chris Lawless, Intel Corporation. EDPS 2014, California USA