Chapter 1
VERIFICATION OF A MODERN PROCESSOR

Abstract. Over the past four decades, microprocessors have come to be a vital and inseparable part of the modern world, becoming the digital brain of numerous electronic devices and gadgets that make today’s lifestyle possible. Processors are capable of performing computation at astonishingly high speeds and are extremely integrated, occupying only a few square centimeters of silicon die. However, this computational power comes at a price: the task of verifying a modern microprocessor and guaranteeing the correctness of its operation is increasingly challenging, even for most established processor vendors. To deliver always higher performance to end-users, processor manufacturers are forced to design progressively more complex circuits and employ immense verification teams to eliminate critical design bugs in a timely manner. Unfortunately, too often size doesn’t seem to matter in verification, as schedules continue to slip, and microprocessors find their way to the marketplace with design errors. In this chapter we overview the life-cycle of a microprocessors, discuss the challenges of verifying these devices, and show examples of hardware errors that have escaped into production silicon because of insufficient validation and their impact.

1.1 The Birth of the Microprocessor

Over the past four decades microprocessors have permeated our world, ushering in the digital age and enabling numerous technologies, without which today’s life style would be all but impossible. Processors are microscopic circuits printed onto silicon dies and consisting of hundreds of millions of transistors interconnected by wires. What distinguishes microprocessors from other integrated circuits is their ability to execute arbitrary software programs. In other words, processors make digital devices programmable and flexible, so a single device can efficiently perform various operations, depending on the program that is running on it. In our every day activities we encounter and use these tiny devices hundreds of times, often without even realizing it. Processors allow us to untether our phones from the wired network and...
enable mobile communications, while their counterparts, deployed by phone companies, made communications richer and much more reliable. Processors monitor the health of hospital patients, control airplanes, tally election votes and predict weather. And, of course, they power millions of personal computers of all shapes and sizes, as well as the backbone of the Internet, a vital and inseparable part of modern life. The computational power of these devices grows every year at an astonishing pace: not long ago processors were only capable of executing just a few thousands operations per second, while today they can perform billions of complex computations per second. Finally, in the past few years, hardware design houses have introduced multi-core processors, that is, systems comprising multiple processors (cores) on the single silicon die. These systems, can execute several programs concurrently, thereby multiplying the overall performance delivered to the user.

However, to be so powerful, processors implement extremely complex architectures, making the design and manufacturing of these devices a major challenge for the semiconductor industry. Companies such as Intel, IBM and AMD are forced to dedicate hundreds of engineers for years at a time to continue to advance microprocessor technology and deliver the next generation processors to end-users. Moreover, as these designs grow in complexity, it becomes increasingly harder to verify them and ensure that they operate properly. Design houses report that today verification efforts significantly outweigh design activities, and that they often staff their teams with two verification engineers per designer. Unfortunately, the complexity and the number of features in each new generation of CPUs have quickly outpaced the capabilities of even the largest industrial teams. As a consequence, today it is impossible to provide high-quality verification of microprocessors with traditional means, and products released to the public are becoming less and less reliable. Furthermore, early in the development process engineers must assess the verifiability of all the features that they want to introduce into the new product: if proposed features cannot pass high-quality validation on time and within budget, they cannot be deployed in the final product and are removed from the design plan, resulting in a reduced set of capabilities and performance of the new system.

The consequences of this trend of diminishing quality in verification can be dramatic: indeed, the impact of bugs in production microprocessors can range widely from innocuous to devastating, for several reasons. For instance, it is possible for a computer system to become compromised, in terms of safety and security, because of a hardware bug. As a result, a system with a buggy processor becomes vulnerable to security attacks. Attacks of this type could be perpetrated even on systems running completely correct software, since they rely exclusively on underlying hardware flaws [Int04, Kas08]. Moreover, bugs can have a disastrous financial impact on the manufacturing company by triggering a costly recall of faulty hardware, as was the case in a past Intel processor [Mar08]; or causing significant delays in product release, similar to what happened with AMD’s Phenom, released in 2007 [Val07]. The impact in both cases is estimated in billions of dollars, due to the large volume of defective components that a functional bug always entails. To prevent devastating errors from seeping into the released designs, a variety of techniques have been devised to detect and correct issues during system’s design and manufacturing. Con-