Chapter 2
THE VERIFICATION UNIVERSE

Abstract. In this chapter we take the reader through a typical microprocessor’s life-cycle, from its first high-level specification to a finished product deployed in an end-user’s system, and overview the verification techniques that are applied at each step of this flow. We first discuss pre-silicon verification, the process of validating a model of the processor at various levels of abstraction, from an architectural specification to a gate-level netlist. Throughout the pre-silicon phase, two main families of techniques are commonly used: formal methods and simulation-based solutions. While the former provide mathematical guarantees of design correctness, the latter are significantly more scalable and, consequently, are more commonly used in the industry today. After the first few prototypes of a processor are manufactured, validation enters the post-silicon domain, where tests can run on the actual silicon hardware. The raw performance of in-hardware execution is one of the major advantages of post-silicon validation, while lack of internal observability and limited debuggability are its main drawbacks. To alleviate this, designers often augment their creations with special features for silicon state acquisition, which we review here. After an arduous process of pre- and post-silicon validation, the device is released to the market and finds its way into a final system. Yet, it may still contain subtle bugs, which could not be exposed earlier by designers due to very compressed production timelines. To combat these escaped errors, vendors and researchers in industry and academia have began investigating alternative dynamic verification techniques: with minimal impact on the processor’s performance, these solutions monitor its health and invoke specialized correction mechanisms when errors manifest at runtime. As we show in this chapter, all three phases of verification, pre-silicon, post-silicon and runtime, have their unique advantages and limitations, which must be taken into account by design houses to attain sufficient verification coverage within their time and cost budgets and to avoid major catastrophes caused by releasing faulty processor products to the commercial market.
2.1 Pre-silicon Verification

Pre-silicon verification is a multi-step process, which is aimed at establishing if a design adheres to its specification and fulfills the designer’s intentions. Pre-silicon verification, or design-time verification, is conducted before any silicon prototype is available (hence its name), and operates over a range of different descriptions of a digital design: architectural, RTL, gate-level, etc. The higher levels of abstraction of the design allow engineers to check, often through mathematical proofs, fundamental properties of a circuit’s operation, such as absence of erroneous behaviors and adherence to formally specified invariants in its functionality. The design is then progressively refined to include more detail, requiring the designers to check correctness after each transformation. However, with the growing level of detail, the time and computational effort required to validate the design’s functionality increases as well, thus, in practice, only the most critical blocks of a modern processor are fully verified at all levels of abstraction before tape-out.

The history of pre-silicon verification of digital circuits goes hand in hand with the evolution of these devices over the last several decades: what began as a fairly simple in-house activity, quickly became a large industry of its own. Today, tens of electronic design automation (EDA) companies and a handful of the largest digital design houses offer designers a wide variety of software and hardware tools to address various phases of pre-silicon verification; while researchers in industry and academia alike publish thousands of papers yearly on the subject. Some of these techniques can be applied to any logic design, while others are domain-specific heuristics that improve the performance of verification for certain types of designs. Since it would be impossible for us to cover the entire spectrum of solutions available, in this section we simply focus on presenting a concise overview of the most vital verification techniques, and provide a range of references at the end of the chapter for the readers interested in deepening their knowledge on the subject.

2.1.1 From specification to microarchitectural description

In today’s microprocessor industry, verification begins early in the design cycle, when architects of the new design develop a detailed set of specifications, outlining the major component blocks and describing their functionality. This specification document is then used as the yardstick to verify various pre-production implementations of the design at varying levels of detail. Note, however, that the document is typically written by several people in a natural human language, such as English, and, therefore, may contain hidden ambiguities or contradictions. Consequently, when bugs are found during the verification of a behavioral model, these might be due to a poor implementation or may be caused by errors in the specification itself, which then must be clarified and updated. As soon as a design specification becomes available, verification engineers begin crafting a test plan that outlines how individ-