CHAPTER

4

BUS-BASED DESIGN EXAMPLE

To set a framework for our discussion, this chapter introduces a typical SoC bus-based design example that consists of various common design components. Each of the following chapters demonstrate the assertion-based IP creation process on many common design components found in our bus-based design example.

Why did we choose an SoC bus-based design example? Our goal is to tie the process of creating assertion-based verification IP to a real design example that you might encounter on a daily basis. With the increased pressure on time-to-market, rapidly changing requirements, and increasing design and manufacturing costs, SoC bus-based design methodologies have recently emerged as a means to address many of the shortcomings of traditional design approaches. For example, SoC bus-based design methodologies allow you to integrate new features relatively quickly by selecting third-party IP. Multiple IP design components are often interconnected using standard interfaces combined with bus-based design techniques.

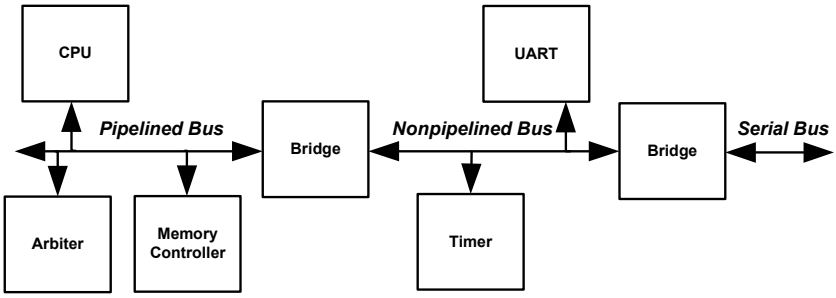
4.1 Bus-based design overview

Figure 4-1 illustrates a typical SoC bus-based design example consisting of various functional design components, which are all interconnected using a common bus. For example, an arbiter component manages the resource use of Bus A to ensure that only one CPU bus master initiates a data transfer at a time. In general, you can implement any arbitration scheme, such as high priority or fair access, depending on the SoC application requirements.

Our example also contains a bridge component, which connects the SoC's internal *Pipelined Bus* to a *Nonpipelined Bus,* and a second bridge component, which connects the SoC's internal *Nonpipelined Bus* to a *Serial Bus*. The bridge manages the transport of data between these buses. Nearly all SoC designs today have multiple busses, which generally consists of four major functional design components directly or indirectly connected to these busses. Hence, it makes sense for us to organize our chapter discussions into the following set of common design components:

- Interfaces (see Chapter 5, “Interfaces”)
- Arbiters (see Chapter 6, “Arbiters”)
- Controllers (see Chapter 7, “Controllers”)
- Datapath (see Chapter 8, “Datapath”)

Figure 4-1 A typical SoC bus-based design



The following chapters demonstrate how to create assertion-based IP for various classes of interfaces, arbiters,

controllers, and datapath components that are typically found in a SoC bus-based design. We apply the process and other methodological guidelines introduced in Chapter 3, "The Process" to create assertion-based IP. There are many process similarities for creating assertion-based IP across each of the various types of design components discussed in the following chapters. However, there are some unique process steps required for specific types of design components, which we cover in their appropriate chapters.

4.2 Summary

In this chapter we introduced a typical SoC bus-based design example that consists of various common design components. Each of the following chapters demonstrate the assertion-based IP creation process on many of the common design components found in our bus-based design example.