This chapter discusses the process of integrating completed macros into the whole chip environment. The topics are:

- Integration overview
- Integrating soft macros
- Integrating hard macros
- Integrating RAMs and datapath generators
- Physical design

10.1 Integration Overview

The key issues in the final integration of the SoC design include:

- Logical design
- Synthesis and physical design
- Chip-level verification

This chapter addresses the first two subjects and the following chapter addresses verification issues.
10.2 Integrating Soft Macros

There are several key issues in designing a chip that uses soft macros provided by an external source:

- Selecting and/or specifying the macro
- Installing the macro into the design environment
- Designing and verifying the interfaces between the macro and the rest of the chip
- Functional verification of the macro in the chip
- Meeting timing requirements with the macro
- Meeting power requirements with the macro

10.2.1 Soft Macro Selection

The first step in selecting a macro from an external source, or in specifying a macro that is to be developed by an internal source, is to determine the exact requirements for the macro. For a standards-based macro, such as a PCI core or a IEEE1394 core, this means developing a sufficient understanding of the standard involved.

Once the requirements for the macro are fully understood, the choices can quickly be narrowed to those that (claim to) meet the functional, timing, area, and power requirements of the design. The most critical factors affecting the choice between several competing sources for a soft macro are:

**Quality of the documentation**
Good documentation is key to determining the appropriateness of a particular macro for a particular application. The basic functionality, interface definitions, timing, and how to configure and synthesize the macro should be clearly documented.

**Robustness of the verification environment that accompanies the macro**
Much of the value, and the development cost, of a macro lies in the verification suite. A rich set of models and monitors for generating stimulus to the macro and checking its behavior can make the overall chip verification much easier.

**Robustness of the design**
A robust, well-designed macro still requires some effort to integrate into a chip design. A poorly designed macro can create major problems and schedule delays. Verifying the robustness of a macro in advance of actually using it is difficult. A careful review of the verification environment and process is a first step. But for a macro to be considered robust, it must have been proven in silicon.