Chapter 4. Methodology

One way to compare the relative performance of different technologies is to design and implement a circuit in each technology, and then analyze it. In general the steps involved in this process are designing a circuit, testing of the design, mapping the design to a technology, determining a mapped design's area, simulating a design, and recording timing information from the simulation. The execution of these steps differ for each technology and, in fact, the process is often different for the same technology at different foundries. The methods used to derive this study's implementations for full custom, standard cell, sea-of-gates and FPGA technologies is presented in this section and summarized in Figure 11.

Full custom designs were created using the CAD tool Magic. This tool allows a designer to graphically lay out elements of a full custom circuit. Unlike other technologies, a full custom design is its implementation—in other words, an implementation itself is designed, not just a functional representation. Therefore the designs do not need further mapping to a technology. Magic also keeps track of the design size, so the area is easily determined when the design is completed. To simulate the design, Spice-3f4[16] was used. The input to Spice is created by running ext2spice on the full custom implementation and adding a model for the transistors.

The standard cell process started by designing each circuit in Verilog. Synopsys was used to compile and optimize each Verilog file using a technology library compatible with Epoch. The result of this compilation was saved as a structural Verilog file. Epoch was used to compile the new Verilog file into a standard cell implementation. This implementation and its size can then

be viewed. Epoch's simulator TACTIC was used to obtain performance data about each implementation.

The sea-of-gates process used is similar to that of a standard cell. The initial Verilog file from the standard cell process is used again. It is loaded into Synopsys and compiled, this time using libraries compatible with the sea-of-gates software, Ocean [8]. Ocean requires the output from Synopsys to be saved in EDIF format, which is then read into Ocean's database. Then Seadali is used to place and route the circuit onto a sea-of-gates structure. The size of the resulting circuit can be determined by inspection. This implementation is simulated with

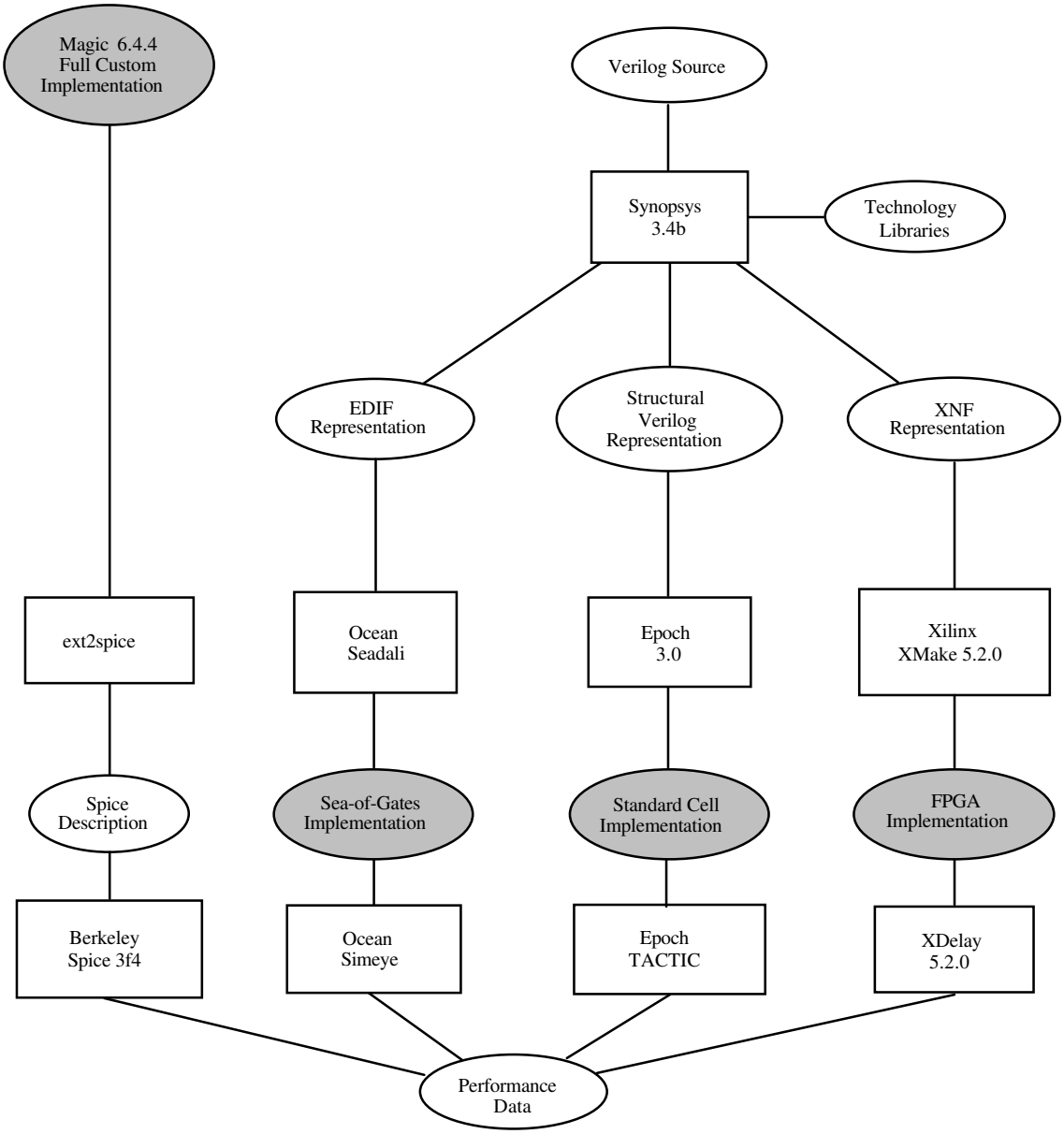

Figure 11. Summary of the process used to create and analyze each circuit.

Simeye to generate its performance data.

To create an FPGA implementation, the initial Verilog file is again loaded into Synopsys and compiled with the Xilinx technology libraries. The result is saved in the XNF file format. Xilinx's tool xmake is run on the XNF file, which results in a configuration file for an FPGA. Also output from xmake are several report files that, among other things, report the design size. The performance of the design is determined by running the design through xdelay, which determines all delays in the circuit and is used to locate the critical delay.