It has turned out that decision diagrams are a well-suited data structure for Boolean functions, since in many cases, the representation is very compact, while also efficient manipulation techniques are known. One application of DDs in circuit synthesis, i.e. to efficiently represent Boolean functions, has been described in the previous chapter.

In this chapter, two main methods to derive circuits from DDs are described. They have in common that they do not only benefit from the functional description of DDs, but also take structural properties into account, using the internal representation to generate the circuit.

The first one is based on a direct mapping to a (multiplexer-based) circuit [ADK91a, Bec92, ADK93, DB93, BDM93, BD95b, SHWM95, YCS00]. The main advantage of this approach is that there is a close correspondence between the representation as a DD and the resulting circuit, i.e. the size of the circuit almost equals the size of the DD. Therefore, the size of the DD is a very good estimate of the resulting area during the minimization phases. The delay of the circuit is linear to the number of inputs in the worst case. Minimization techniques for different types of DDs are outlined, since they can be directly applied to minimize the area of the circuit. Some target technologies are described in more detail, like Pass Transistor Logic (PTL). Furthermore, it is shown that testability properties of the resulting circuits can be determined directly from the DD representation.

The major drawback of the first approach is that the delay might become too large, which is not acceptable in many cases. Thus, we discuss a second approach based on Boolean Matrix Multiplication (BMM) [Ish92, HDE+96].
There, the resulting circuits only have logarithmic depth, which is usually much smaller. Therefore, the approach can also be applied to high-speed circuits, where the area usage is not the main concern. Also some testability results are given.

The chapter closes with a discussion of the two approaches.

6.1 DIRECT MAPPING OF DECISION DIAGRAMS

The easiest way to map a decision diagram to a circuit is to substitute each node of the DD by a sub-circuit realizing the corresponding decomposition function. This approach is described in more detail in the following.

A circuit for a DD representation can easily be obtained by traversing the corresponding graph in topological order and replacing each node by a sub-circuit over library STD. The sub-circuit realizes the decomposition function associated to the corresponding variable. The depth of the resulting circuit is linear in the number of variables in the worst case. On the other hand the size directly corresponds to the representation size of the DD.

Example 6.1 On the left-hand side in Figure 6.1 a KFDD with DTL $d = (S, pD, nD, S)$ for the function

$$f(x_1, x_2, x_3, x_4) = \overline{x_1}x_2x_3 \oplus x_1\overline{x_2}x_3 \oplus \overline{x_2}x_3x_4 \oplus x_1x_2$$

is shown. A dot on an edge symbolizes a complemented edge.

The KFDD-circuit corresponding to the KFDD is shown on the right-hand side of Figure 6.1. The circuit is drawn “upside-down” to show the close correspondence to the KFDD. The sub-circuits realizing the decomposition functions for each node are shown by dashed boxes.

The sub-circuits can be simplified if the corresponding nodes in the DD are connected to terminal nodes, which represent constant functions 1 and 0. For instance, in the example above the multiplexer connected with $x_4$ could be realized by a single inverter.