User defined primitives

- The logic gates used in Verilog descriptions with keywords **and, or,** etc., are defined by the system and are referred to as *system primitives.*
- The user can create additional primitives by defining them in tabular form. These types of circuits are referred to as *user-defined primitives* (UDPs).
- they are declared with the keyword pair **primitive** . . . **endprimitive.**

UDPs

- It is declared with the keyword **primitive**, followed by a name and port list.
- There can be only one output, and it must be listed first in the port list and declared with keyword **output**.
- There can be any number of inputs. The order in which they are listed in the **input** declaration must conform to the order in which they are given values in the table that follows.
- The truth table is enclosed within the keywords **table** and **endtable.**
- The values of the inputs are listed in order, ending with a colon (:). The output is always the last entry in a row and is followed by a semicolon (;).
- The declaration of a UDP ends with the keyword **endprimitive**.

UDPs

```
// Verilog model: User-defined Primitive
primitive  UDP_02467 (D, A, B, C);
  output  D;
  input    A, B, C;
//Truth table for D 5 f (A, B, C) 5 Σ(0, 2, 4, 6, 7);
  table
//      A      B      C      :      D      // Column header comment
        0      0      0      :      1;
        0      0      1      :      0;
        0      1      0      :      1;
        0      1      1      :      0;
        1      0      0      :      1;
        1      0      1      :      0;
        1      1      0      :      1;
        1      1      1      :      1;
  endtable
endprimitive
```

Instantiation of UDPs

```
// Verilog model: Circuit instantiation of Circuit_UDP_02467

module Circuit_with_UDP_02467 (e, f, a, b, c, d);
 output        e, f;
 input         a, b, c, d

 UDP_02467     M1   (e, a, b, c);
 and           G1   (f, e, d);          // Option gate instance name omitted
endmodule
```