

## 第6章 用户定义的原语

在前一章中，我们介绍了 Verilog HDL 提供的内置基本门。本章讲述 Verilog HDL 指定用户定义原语 UDP 的能力。

UDP 的实例语句与基本门的实例语句完全相同，即 UDP 实例语句的语法与基本门的实例语句语法一致。

### 6.1 UDP 的定义

使用具有如下语法的 UDP 说明定义 UDP。

```
primitive UDP_name (OutputName, List_of_inputs
    Output_declaration
    List_of_input_declarations
    [Reg_declaration]
    [Initial_statement]
    table
        List_of_tabel_entries
    endtable
endprimitive
```

UDP 的定义不依赖于模块定义，因此出现在模块定义以外。也可以在单独的文本文件中定义 UDP。

UDP 只能有一个输出和一个或多个输入。第一个端口必须是输出端口。此外，输出可以取值 0、1 或 x (不允许取 z 值)。输入中出现值 z 以 x 处理。UDP 的行为以表的形式描述。

在 UDP 中可以描述下面两类行为：

- 1) 组合电路
- 2) 时序电路 (边沿触发和电平触发)

### 6.2 组合电路 UDP

在组合电路 UDP 中，表规定了不同的输入组合和相对应的输出值。没有指定的任意组合输出为 x。下面以 2-1 多路选择器为例加以说明。

```
primitive MUX2x1 (Z, Hab, Bay, Sel;
    output Z;
    input Hab, Bay, Sel;

    table
        // Hab Bay Sel : Z 注：本行仅作为注释。
        0    ?    1 : 0 ;
        1    ?    1 : 1 ;
        ?    0    0 : 0 ;
        ?    1    0 : 1 ;
        0    0    x : 0 ;
```

```

1      1      x : 1 ;
endtable
endprimitive

```

字符?代表不必关心相应变量的具体值,即它可以是0、1或x。输入端口的次序必须与表中各项的次序匹配,即表中的第一列对应于原语端口队列的第一个输入(例子中为*Hab*),第二列是*Bay*,第三列是*Sel*。在多路选择器的表中没有输入组合01x项(还有其它一些项);在这种情况下,输出的缺省值为x(对其它未定义的项也是如此)。

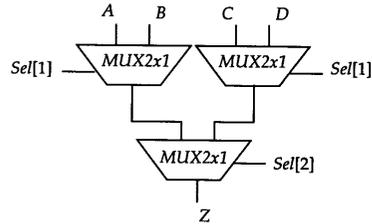


图6-1 使用UDP构造的4-1多路选择器

图6-1为使用2-1多路选择器原语组成的4-1多路选择器的示例。

```

module MUX4x1 (Z, A, B, C, D, Sel);
    input A, B, C, D;
    input [2:1] Sel;
    output Z;
    parameter tRISE = 2, tFALL = 3;

    MUX2x1 #(tRISE, tFALL)
        (TL, A, B, Sel[1]),
        (TP, C, D, Sel[1]),
        (Z, TL, TP, Sel[2]);
endmodule

```

如上例所示,在UDP实例中,总共可以指定2个时延,这是由于UDP的输出可以取值0、1或x(无截止时延)。

## 6.3 时序电路UDP

在时序电路UDP中,使用1位寄存器描述内部状态。该寄存器的值是时序电路UDP的输出值。共有两种不同类型的时序电路UDP:一种模拟电平触发行为;另一种模拟边沿触发行为。时序电路UDP使用寄存器当前值和输入值决定寄存器的下一状态(和后继的输出)。

### 6.3.1 初始化状态寄存器

时序电路UDP的状态初始化可以使用带有一条过程赋值语句的初始化语句实现。形式如下:

```
initial reg_name = 0,1,or x;
```

初始化语句在UDP定义中出现。

### 6.3.2 电平触发的时序电路UDP

下面是D锁存器建模的电平触发的时序电路UDP示例。只要时钟为低电平0,数据就从输入传递到输出;否则输出值被锁存。

```

primitive Latch (Q, Clk, D);
    output Q;
    reg Q;
    input Clk, D;

    table

```

```
// Clk  D  Q(State)  Q(next)
0  1  :?  :  1 ;
0  0  :?  :  0 ;
1  ?  :?  :  - ;

endtable
endprimitive
```

“-”字符表示值“无变化”。注意UDP的状态存储在寄存器D中。

### 6.3.3 边沿触发的时序电路UDP

下例用边沿触发时序电路UDP为D边沿触发触发器建模。初始化语句用于初始化触发器的状态。

```
primitive D_Edge_FF (Q, Clk, Data);
output Q;
reg Q;
input Data, Clk

initial Q = 0;
table
// Clk  Data  (State)  Q(next)
(01)  0  :  ?  :  0 ;
(01)  1  :  ?  :  1 ;
(0x)  1  :  1  :  1 ;
(0x)  0  :  0  :  0 ;
// 忽略时钟负边沿:
(?0)  ?  :  ?  :  - ;
// 忽略在稳定时钟上的数据变化:
?  (??):  ?  :  - ;

endtable
endprimitive
```

表项(01)表示从0转换到1，表项(0x)表示从0转换到x，表项(?0)表示从任意值(0,1或x)转换到0，表项(??)表示任意转换。对任意未定义的转换，输出缺省为x。

假定D\_Edge\_FF为UDP定义，它现在就能够象基本门一样在模块中使用，如下面的4位寄存器所示。

```
module Reg4 (Clk, Din, Dout);
input Clk;
input [0:3] Din;
output [0:3] Dout;

D_Edge_FF
DLAB0 (Dout[0],Clk, Din[0]),
DLAB1 (Dout[1],Clk, Din[1]),
DLAB2 (Dout[2],Clk, Din[2]),
DLAB3 (Dout[3],Clk, Din[3]),

endmodule
```

### 6.3.4 边沿触发和电平触发的混合行为

在同一个表中能够混合电平触发和边沿触发项。在这种情况下，边沿变化在电平触发之

前处理，即电平触发项覆盖边沿触发项。

下例是带异步清空的D触发器的UDP描述。

```
primitive D_Async_FF(Q, Clk, Clr, Data);
  output Q;
  reg Q;
  input Clr, Data, Clk;

  table
    // Clk   Clr   Data   (Qstate) Q(next)
    (01)    0   0   :   ?   :   0 ;
    (01)    0   1   :   ?   :   1 ;
    (0x)    0   1   :   1   :   1 ;
    (0x)    0   0   :   0   :   0 ;
    // 忽略时钟负边沿:
    (?0)    0   ?   :   ?   :   - ;
    (??)    1   ?   :   ?   :   0 ;
    ?       1   ?   :   ?   :   0 ;
  endtable
endprimitive
```

## 6.4 另一实例

下面是3位表决电路的UDP描述。如果输入向量中存在2个或更多的1，则输出为1。

```
primitive Majority3(Z, A, B, C);
  input A, B, C;

  output Z;
  table
    //A   B   C   :   Z
    0     0   ?   :   0 ;
    0     ?   0   :   0 ;
    ?     0   0   :   0 ;
    1     1   ?   :   1 ;
    1     ?   1   :   1 ;
    ?     1   1   :   1 ;
  endtable
endprimitive
```

## 6.5 表项汇总

出于完整性考虑，下表列出了所有能够用于UDP原语中表项的可能值。

符 号	意 义	符 号	意 义
0	逻辑0	(AB)	由A变到B
1	逻辑1	*	与(??)相同
x	未知的值	r	上跳变沿，与(01)相同
?	0、1或x中的任一个	f	下跳变沿，与(10)相同
b	0或1中任选一个	p	(01)、(0x)和(x1)的任一种
-	输出保持	n	(10)、(1x)和(x0)的任一种

## 习题

1. 组合电路UDP与时序电路UDP如何区别?
2. UDP可有一个或多个输出, 是否正确?
3. 初始语句可用于初始化组合电路UDP吗?
4. 为图5-12中显示的优先编码器电路编写UDP描述。使用测试激励验证描述的模型。
5. 为T触发器编写UDP描述。在T触发器中, 如果数据输入为0, 则输出不变化。如果数据输入是1, 那么输出在每个时钟沿翻转。假定触发时钟沿是时钟下跳沿, 使用测试激励验证所描述的模型。
6. 以UDP方式为上跳边沿触发的JK触发器建模。如果J和K两个输入均为0, 则输出不变。如果J为0, K为1, 则输出为0。如果J是1, K是0, 则输出是1。如果J和K都是1, 则输出翻转。使用测试激励验证描述的模型。