基本数字逻辑单元HDL描述

主讲：何宾

Email：hebin@mail.buct.edu.cn

2014.06

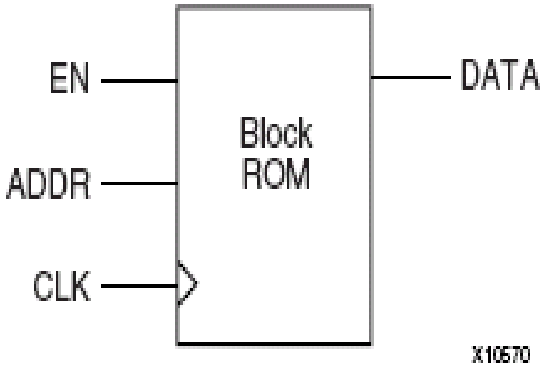
基本数字逻辑单元HDL描述
--存储器HDL描述

存储器按其类型主要分为只读存储器和随机存储器两种。

- **虽然存储器从其工艺和原理上各不相同，但有一点是相同的，即存储器是单个存储单元的集合体，并且按照顺序排列。其中的每一个存储单元由N位二进制位构成，表示存放的数据的值。**

思考：区分分布式存储器和块存储器的概念。

存储器HDL描述
--ROM HDL描述



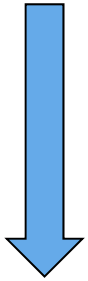
- 只读存储器的数据被事先保存到了每个存储单元中，在FPGA中保存数据的方法有很多。
- 当对ROM进行读操作时，只要在控制信号的控制下，对操作的单元给出读取的数值即可。

图中：

EN为ROM的使能信号，ADDR为ROM的地址信号，CLK为ROM的时钟信号，DATA为数据信号。

存储器HDL描述
--ROM HDL描述方法1

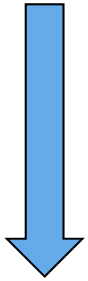
```
module rams_21a(
    input en,
    input [5:0] addr,
    input clk,
    output reg[19:0] data
    );
always @(posedge clk)
  begin
    if(en)
        case(addr)
```



连续下一页

存储器HDL描述
--ROM HDL描述方法1

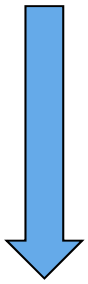
```
6'b000000: data <= 20'h0200A; 6'b100000: data <= 20'h02222;
6'b000001: data <= 20'h00300; 6'b100001: data <= 20'h04001;
6'b000010: data <= 20'h08101; 6'b100010: data <= 20'h00342;
6'b000011: data <= 20'h04000; 6'b100011: data <= 20'h0232B;
6'b000100: data <= 20'h08601; 6'b100100: data <= 20'h00900;
6'b000101: data <= 20'h0233A; 6'b100101: data <= 20'h00302;
6'b000110: data <= 20'h00300; 6'b100110: data <= 20'h00102;
6'b000111: data <= 20'h08602; 6'b100111: data <= 20'h04002;
6'b001000: data <= 20'h02310; 6'b101000: data <= 20'h00900;
6'b001001: data <= 20'h0203B; 6'b101001: data <= 20'h08201;
6'b001010: data <= 20'h08300; 6'b101010: data <= 20'h02023;
6'b001011: data <= 20'h04002; 6'b101011: data <= 20'h00303;
```



连续下一页

存储器HDL描述
--ROM HDL描述方法1

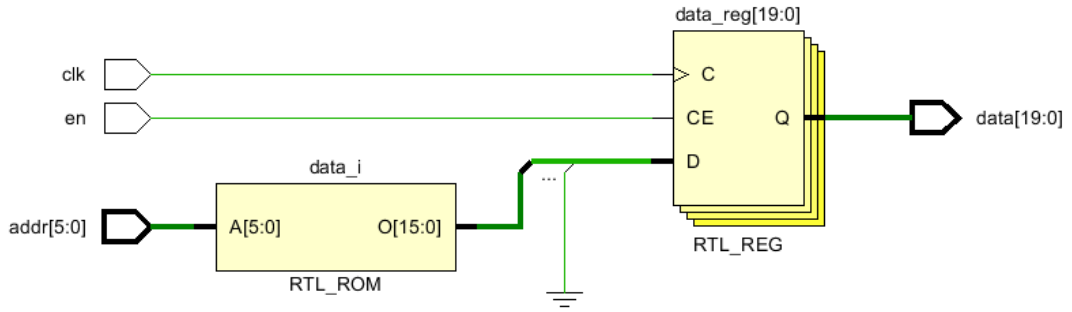
```
6'b001100: data <= 20'h08201; 6'b101100: data <= 20'h02433;
6'b001101: data <= 20'h00500; 6'b101101: data <= 20'h00301;
6'b001110: data <= 20'h04001; 6'b101110: data <= 20'h04004;
6'b001111: data <= 20'h02500; 6'b101111: data <= 20'h00301;
6'b010000: data <= 20'h00340; 6'b110000: data <= 20'h00102;
6'b010001: data <= 20'h00241; 6'b110001: data <= 20'h02137;
6'b010010: data <= 20'h04002; 6'b110010: data <= 20'h02036;
6'b010011: data <= 20'h08300; 6'b110011: data <= 20'h00301;
6'b010100: data <= 20'h08201; 6'b110100: data <= 20'h00102;
6'b010101: data <= 20'h00500; 6'b110101: data <= 20'h02237;
6'b010110: data <= 20'h08101; 6'b110110: data <= 20'h04004;
6'b010111: data <= 20'h00602; 6'b110111: data <= 20'h00304;
```



连续下一页

存储器HDL描述
--ROM HDL描述方法1

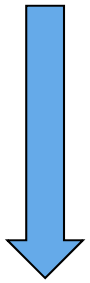
```
6'b011000: data <= 20'h04003; 6'b111000: data <= 20'h04040;
6'b011001: data <= 20'h0241E; 6'b111001: data <= 20'h02500;
6'b011010: data <= 20'h00301; 6'b111010: data <= 20'h02500;
6'b011011: data <= 20'h00102; 6'b111011: data <= 20'h02500;
6'b011100: data <= 20'h02122; 6'b111100: data <= 20'h0030D;
6'b011101: data <= 20'h02021; 6'b111101: data <= 20'h02341;
6'b011110: data <= 20'h00301; 6'b111110: data <= 20'h08201;
6'b011111: data <= 20'h00102; 6'b111111: data <= 20'h0400D;
    endcase
  end
endmodule
```



本设计保存在本书配套资源\eda_verilog\example6_30目录下

存储器HDL描述
--ROM HDL描述方法2

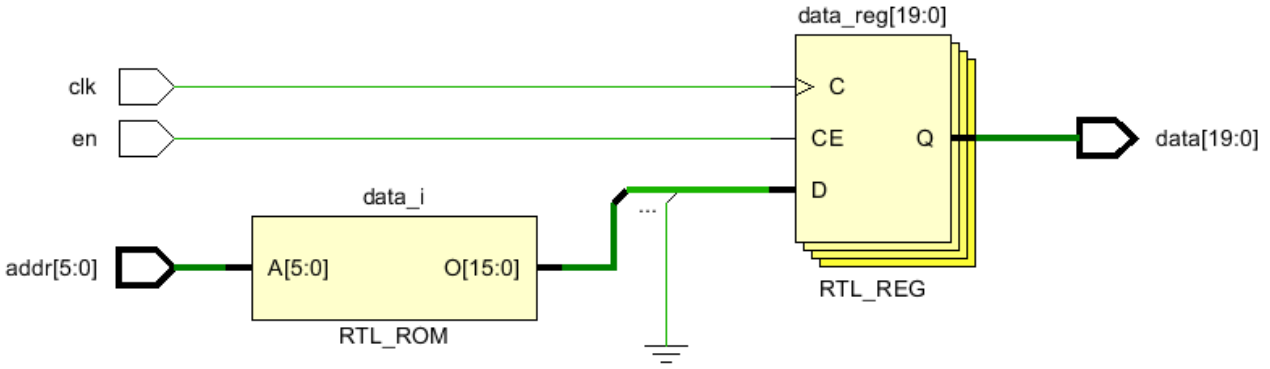
```
module rams_21a(
    input en,
    input [5:0] addr,
    input clk,
    output reg[19:0] data
    );
parameter ADDRESS_WIDTH=6;
localparam rom_depth=2**ADDRESS_WIDTH-1;
 (*rom_style="block"*)  reg [19:0] memory[0:rom_depth];
initial
begin
     $readmemh("text.txt", memory);
end
```



连续下一页

存储器HDL描述
--ROM HDL描述方法2

```
always @(posedge clk)
begin
    if(en)
        data<=memory[addr];
    end
endmodule
```

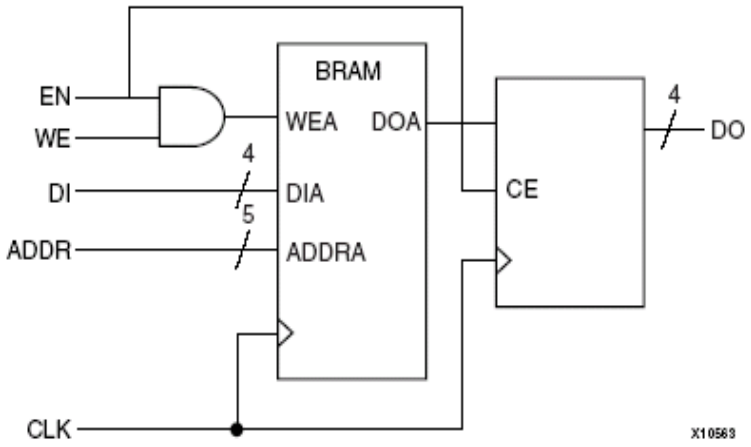


思考与练习1：请读者比较两种描述ROM方法的不同点
思考与练习2：请说明属性(*rom_style="block"*) 的意义。

本设计保存在本书配套资源\eda_verilog\example6_30_1目录下

存储器HDL描述
--RAM HDL描述

RAM和ROM的区别，在于RAM有读写两种操作，而ROM只有读操作。另外，RAM对读写的时序也有着严格的要求。

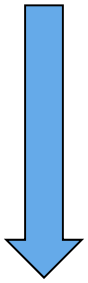


图中：

- EN为RAM使能信号；
- WE为RAM写信号；
- DI为RAM数据输入信号；
- ADDR为RAM地址信号；
- CLK为RAM时钟信号；
- DO为RAM数据输出信号。

存储器HDL描述
--RAM HDL描述的例子

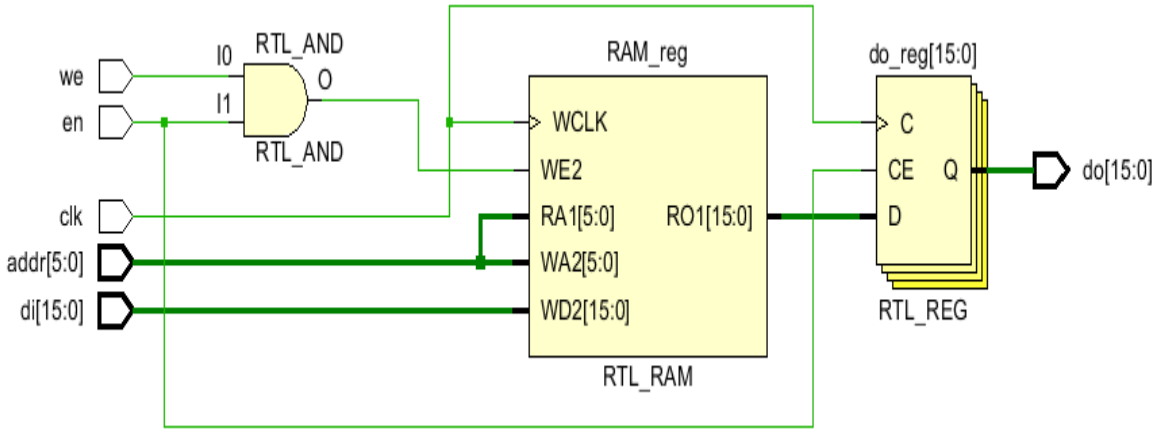
```
module rams_01(
    input clk,
    input we,
    input en,
    input [5:0] addr,
    input [15:0] di,
    output reg[15:0] do
    );
(*ram_style="block"*) reg [15:0] RAM [63:0];
```



连续下一页

存储器HDL描述
--RAM HDL描述的例子

```
always @(posedge clk)
begin
    if(en)
       begin
          if(we)
             RAM[addr] <= di;
             do  <= RAM[addr];
       end
 end
endmodule
```



本设计保存在本书配套资源\eda_verilog\example6_31目录下