# 基本数字逻辑单元HDL描述

主 讲：何宾

Email：hebin@mail.buct.edu.cn

2014.06

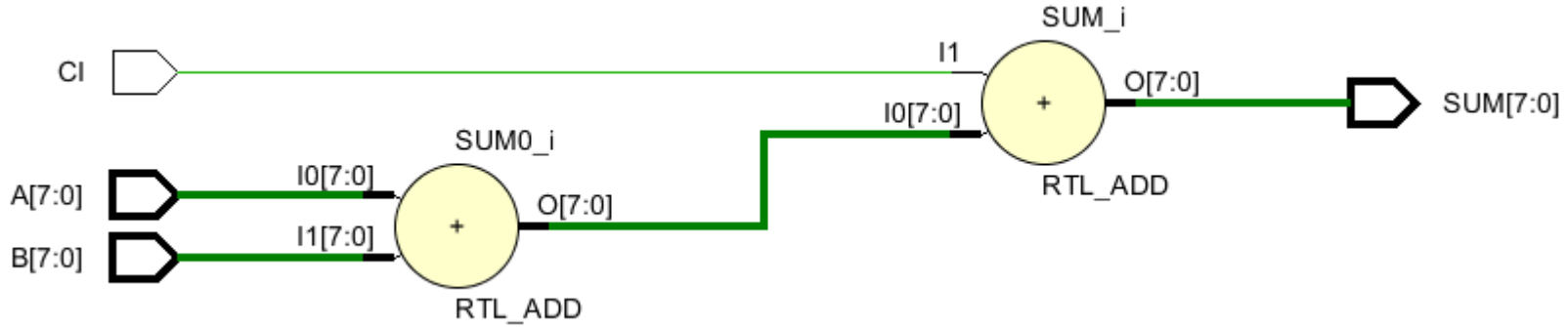
# 组合逻辑电路的HDL描述
## --数据运算操作HDL描述

数据运算操作主要包含加法操作、减法操作、乘法操作和除法操作，由这四种运算单元和逻辑运算单元一起，可以完成复杂数学运算。

- HDL语言中提供了丰富的数据算术操作的运算符。

# 加法操作HDL描述
## --带进位输入和输出无符号8位加法的例子

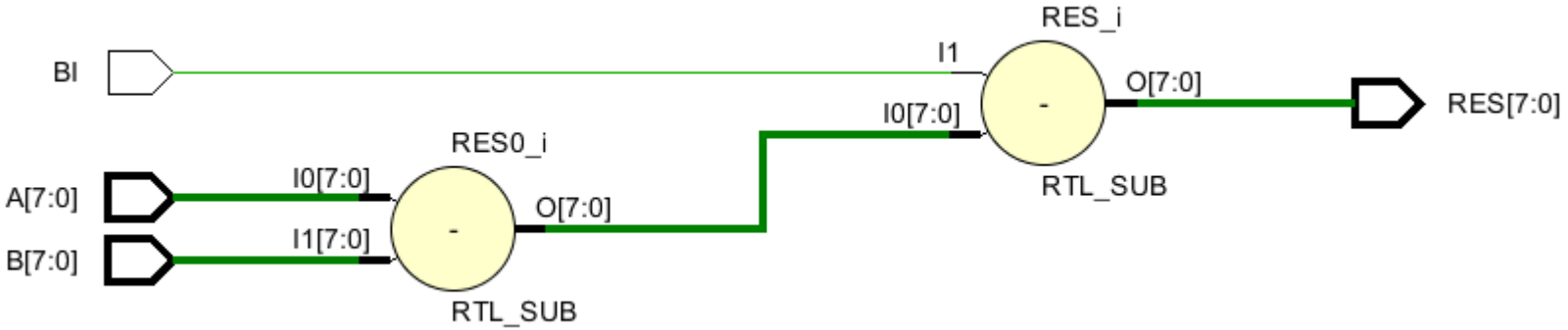
```
module v_adders_2(A, B, CI, SUM);
input [7:0] A;
input [7:0] B;
input CI;
output [7:0] SUM;
      assign SUM = A + B + CI;
endmodule
```




本设计保存在本书配套资源**eda_verilog\example6_10**目录下

# 减法操作HDL描述
## --无符号带借位8位减法器的例子

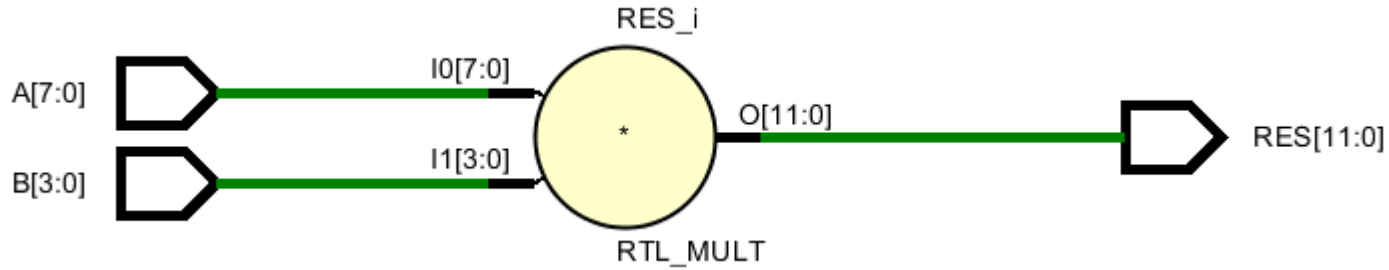
```
module v_adders_8(A, B, BI, RES);
input [7:0] A;
input [7:0] B;
input BI;
output [7:0] RES;
      assign RES = A - B - BI;
endmodule
```




本设计保存在本书配套资源**eda_verilog\example6_11**目录下

# 乘法操作HDL描述
## --8位与4位无符号数相乘的例子

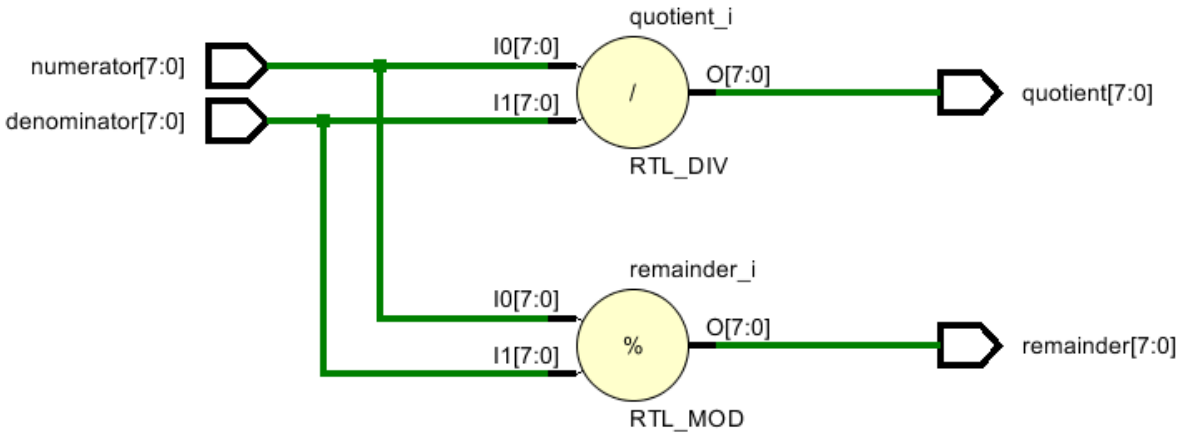
```
module v_multipliers_1(A, B, RES);

input [7:0] A;

input [3:0] B;

output [11:0] RES;

      assign RES = A * B;

endmodule
```




本设计保存在本书配套资源**eda_verilog\example6_12**目录下

# 除法操作HDL描述
## --无符号8位除法运算的例子

```
module div(
    input [7:0] numerator,
    input [7:0] denominator,
    output [7:0] quotient,
    output [7:0] remainder
    );
        assign quotient=numerator/denominator;
        assign remainder=numerator % denominator;
endmodule
```



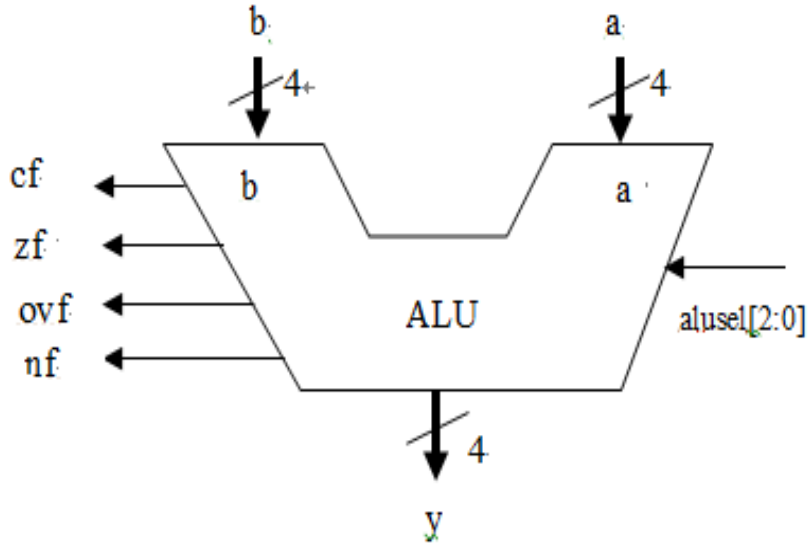

本设计保存在本书配套资源**eda_verilog\example6_13**目录下

# 数据运算操作HDL描述
## --算术逻辑单元HDL描述

前面介绍了加法器和减法器电路的设计。通过增加一些逻辑操作，设计一个叫做算术/逻辑单元ALU的模块。

由于ALU包含了所希望实现的功能集的电路，因此很容易替换/扩展来包含不同的操作。

# 数据运算操作HDL描述
## --算术逻辑单元HDL描述




ALU操作

| alusel[2:0] | 功能 | 输出 |
|---|---|---|
| 000 | 传递a | a |
| 001 | 加 | a+b |
| 010 | 减1 | a-b |
| 011 | 减2 | b-a |
| 100 | 取反 | not a |
| 101 | 与 | a and b |
| 110 | 或 | a or b |
| 111 | 异或 | a xor b |

# 数据运算操作HDL描述
## --算术逻辑单元HDL描述

- 由于ALU完成8种功能，所以选择线为3位.
- 此外，ALU提供4位的y输出和四个标志位。其中：
  - cf为进位标志；
  - ovf为溢出标志；
  - zf为0标志（当输出为0时，该标志有效）；
  - nf为负标志（当输出的第4位为1时，该标志有效）。

# 数据运算操作HDL描述
## --算术逻辑单元HDL描述

- 为了讨论进位标志和溢出标志的不同，考虑一个8位的加法（最高位为符号位）。当无符号的数的和超过255时，进位标志被设置。当有符号数的和超过了-128-+127的范围时，溢出标志被设置。

# 数据运算操作HDL描述
# --算术逻辑单元HDL描述

- 考虑下面的几个例子(最高位为符号位)：
  - $53_{10}+25_{10}=35_{16}+19_{16}=78_{10}=4E_{16}$,cf=0,ovf=0
  - $53_{10}+91_{10}=35_{16}+5B_{16}=144_{10}=90_{16}$,cf=0,ovf=1
  - $53_{10}-45_{10}=35_{16}+D3_{16}=8_{10}=108_{16}$,cf=1,ovf=0
  - $-98_{10}-45_{10}=9E_{16}+D3_{16}=-143_{10}=171_{16}$,cf=1,ovf=1
- 当满足条件：（第六位向第七位进位）xor （第七位向cf进位）时，ovf=1。

# 数据运算操作HDL描述
## --算术逻辑单元HDL描述的例子

```
module ALU(
input wire [2:0] alusel,
input wire [3:0] a,
input wire [3:0] b,
output reg nf,
output reg zf,
output reg cf,
output reg ovf,
output reg [3:0] y
);
reg [4:0] temp;
always @(*)
```

# 数据运算操作HDL描述
## --算术逻辑单元HDL描述

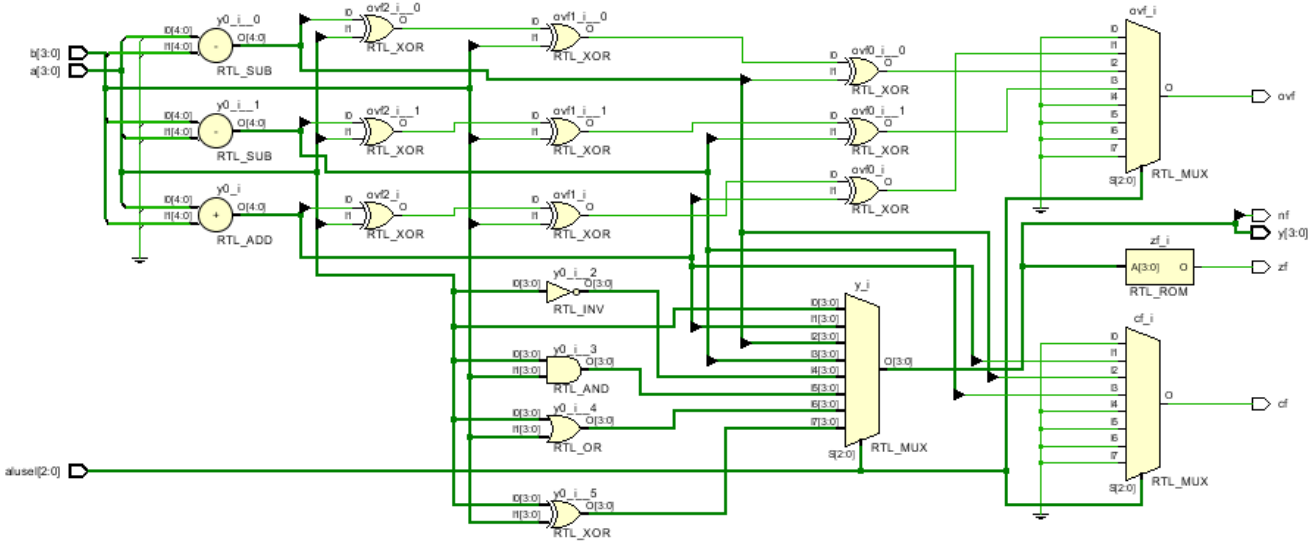
```
begin
   cf = 0;
   ovf =0;
   temp = 5'b00000;
    case (alusel)
      3'b000 : y = a;
      3'b001 :
        begin
          temp = {1'b0,a} + {1'b0,b};
          y = temp[3:0];
          cf = temp[4];
          ovf = y[3] ^ a[3] ^ b[3] ^ cf;
        end
      3'b010 :
```

# 数据运算操作HDL描述
## --算术逻辑单元HDL描述

```
begin
   temp = {1'b0,a} - {1'b0,b};
   y  = temp[3:0];
   cf = temp[4];
   ovf = y[3] ^ a[3] ^ b[3] ^cf;;
end
3'b011 :
 begin
   temp = {1'b0,b} - {1'b0,a};
   y = temp[3:0];
   cf = temp[4];
   ovf = y[3] ^ a[3] ^ b[3] ^ cf;
 end
```

# 数据运算操作HDL描述
## --算术逻辑单元HDL描述

```
        3'b100 : y = ~a;
        3'b101 : y = a & b;
        3'b110 : y = a | b;
        3'b111 : y = a ^ b;
        default : y = a;
    endcase
    nf = y[3];
    if(y == 4'b0000)
        zf = 1;
    else
        zf = 0;
    end
endmodule
```




本设计保存在本书配套资源**eda_verilog\example6_14**目录下