



Bookmarks

- ▶ [Hello! Computer Organization!](#)
- ▶ [基础知识](#)
- ▶ [Logisim](#)
- ▶ [Verilog-HDL与 ISE](#)
- ▶ [MIPS指令集及汇编语言](#)
- ▶ [P0-Logisim简单部件与状态机](#)
- ▼ [P1-Verilog简单部件与状态机](#)

课下测试

P1-Verilog简单部件与状态机 > 课下测试 > Verilog部件设计（二）

Verilog部件设计（二）

🔖 Bookmark this page

2、Verilog 实现 ALU

使用 Verilog 搭建一个 32 位六运算 ALU 并提交。具体模块端口定义如下：

信号名	方向	描述
A [31:0]	I	参与 ALU 计算的第一个值。
B[31:0]	I	参与 ALU 计算的第二个值。
ALUOp[2:0]	I	ALU 功能的选择信号。 000: ALU 进行加法运算。 001: ALU 进行减法运算。 010: ALU 进行与运算。 011: ALU 进行或运算。 100: ALU 进行逻辑右移。 101: ALU 进行算数右移。
C[31:0]	O	ALU 的计算结果。

模块功能定义如下：

序号	功能名称	功能描述
1	无符号加运算	$C=A+B$ (A,B 无符号)。
2	无符号减运算	$C=A-B$ (A,B 无符号)。
3	与运算	$C=A\&B$ 。
4	或运算	$C=A B$ 。
5	逻辑右移	$C=A>>B$ (A, B 无符号)。
6	算术右移	$C=A>>>B$ (A 有符号 B 无符号)。

知识补充：

在这里我们首先需要知道逻辑右移和算数右移的区别：

逻辑右移不考虑符号位，右移一位，左边补零即可。

算术右移需要考虑符号位，右移一位，若符号位为 1，在左边补 1；否则，补 0。

例如，8 位二进制数 10111101 分别右移 2 位。

- 逻辑右移结果为 **00**101111
- 算术右移结果为 **11**101111

要求:

- 必须严格按照模块的端口定义
- 文件内模块名: alu

Verilog实现ALU

1 point possible (ungraded)

提交入口

Choose Files No file chosen

Submit

Discussion

Topic: P1-Verilog简单部件与状态机: Lab0-课下测试 / Verilog部件设计 (二)

Show Discussion
