top\_cla\_32模块

# 作用：

利用先行进位加法器完成两个32位数据的加操作。

# 参数说明

module top\_cla\_32(

input [31:0] a, b, //input A and B

input c\_in, //the carry num

output c\_out, // the carry num of result

output [31:0] s //result

);

# 设计流程



## **一位全加器设计**

### **1.真值表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| a | **b** | **c\_in** | **c\_out** | **s** |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 |
| 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 1 | 1 |

### 2.卡诺图

#### c\_out卡诺图

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| c\_in ab | 00 | 01 | 11 | 10 |
| 0 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 1 | 1 |

**c\_out = bc\_in + ab + ac\_in = ab + (a + b)c\_in**

**=> c\_out = g + pc\_in（其中g = ab，p = a + b）**

#### s卡诺图

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| c\_in ab | 00 | 01 | 11 | 10 |
| 0 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 1 | 0 |

**s = a ^ b ^ c**

## 进位器设计

根据一位加法器得：

**c[0] = g[0] + p[0]c\_in;**

**c[1] = g[1] + p[1]c[0];**

则两个一个加法器的进位产生和传递函数为：

**c\_out = g[1] + p[1](g[0] + p[0]c\_in) = g[1] + p[1]g[0] + p[1]p[0]c\_in**

**g[1,0] = g[1] + p[1]g[0]**

**p[1,0] = p[1]p[0]**

# 设计要点

先行进位加法器[1]：利用递归的思想快速得出各个进位数值，避免进位在多位数全加器的传递过程中的延时。对2,4,8,16,32位加法电路要分别进行仿真验证，以确保模块的正确性

**References:**

[1] 李亚民, 计算机原理与设计——verilog HDL 版, Vol. 先行进位加法器, 清华大学出版社.