# 0.IC系统设计优化主要有两方面：

* 资源优化
* 速度优化
* 无论是在ASIC还是FPGA中，硬件设计资源即面积(Area)是一个重要的技术指标。
* 资源优化意义：

1.可使用规模更小的可编程器件，降低系统成本，提高性价比。

2.当耗用资源过多时会严重影响时序性能。

3.为后续技术升级留下更多的可编程资源。

4.资源耗用太多会使器件功耗显著上升。

# 1.资源优化

## 1.1 资源共享

通过一个例子进行说明。现需要实现功能：通过选择信号s来选择执行A0 \* B还是A1 \* B的。

描述方式1：对两个乘法电路进行选择

RTL结构：

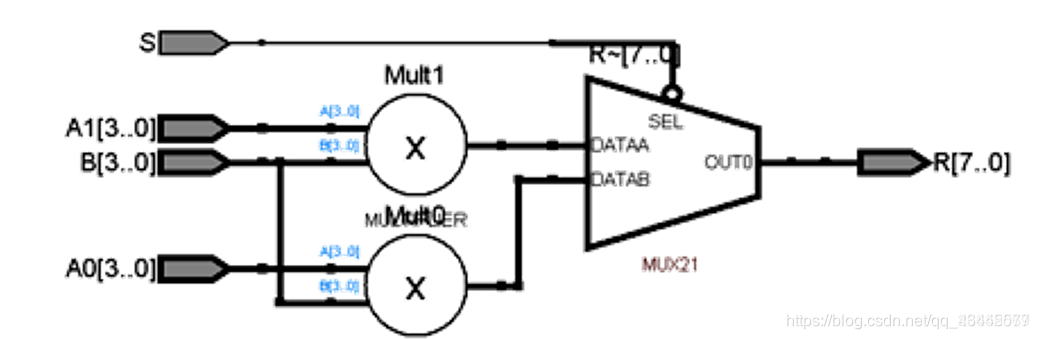


图1：先乘后选择

描述方式2: 对乘项进行选择

RTL结构：

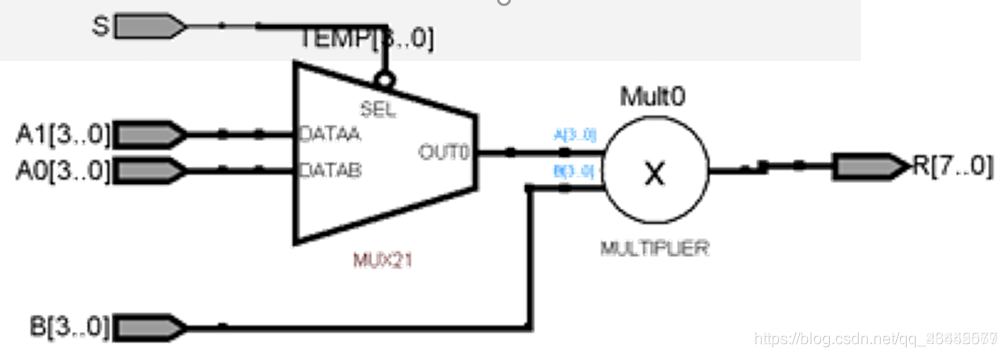


图2：先选择后乘

两种方式一比较，后者在逻辑结果上没有任何改变，但却节省了一个代价高昂的乘法器，使得整个设计占用的面积几乎减少了一半。

**总结：**

资源共享主要针对数据通路中耗费逻辑资源较多的模块，通过选择、复用的方式共享使用该模块，达到减少资源使用、优化面积的目的。

并不是在任何情况下都能以此法实现资源优化，输入与门之类的模块使用资源共享是无意义的，有时甚至会增加资源的使用（多路选择器的面积显然要大于与门）。高级的HDL综合器，如QuartusII和Synplify Pro等，通过设置能自动识别设计中需要资源共享的逻辑结构，自动地进行资源共享。

## 1.2 逻辑优化

使用优化后的逻辑进行设计，可以明显减少资源的占用。

状态机的设计要尽可能简洁，不要搞出不必要的，多个状态对同一个数据做处理。

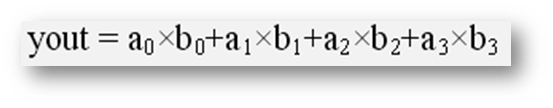
if语句中用单bit的标志位代替多bit的数据来进行条件判断。

设计尽可能简洁，不要弄出冗余赋值。

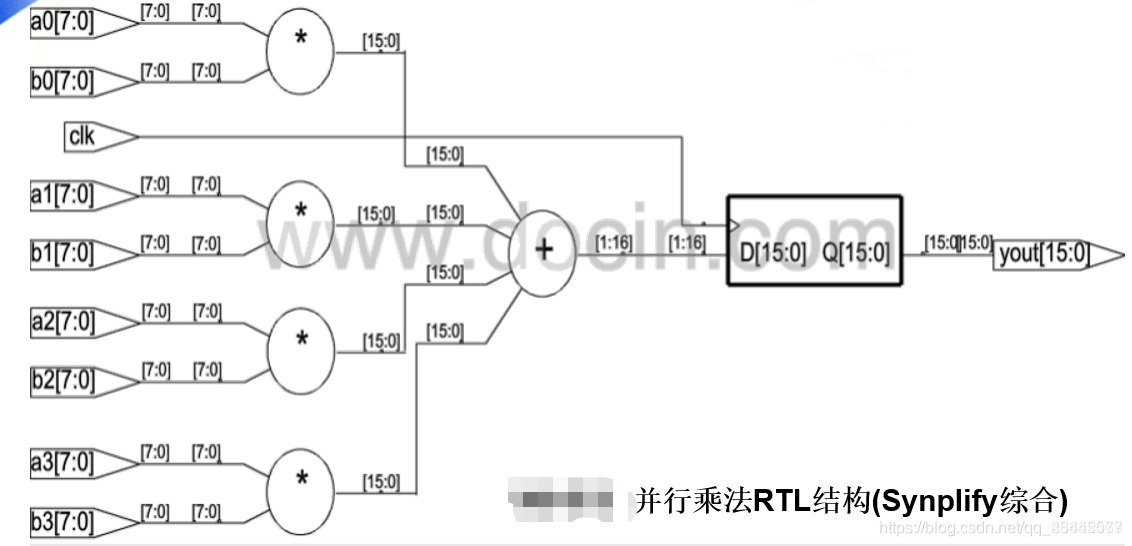
## 1.3 串行化

将原来耗用资源巨大、单时钟周期内完成的并行执行的逻辑块分割开，提取出相同的逻辑模块（一般为组合逻辑块），在时间上利用该逻辑模块，用多个时钟周期完成相同的功能，其代价是工作速度被大为降低。

例 ：实现



描述方式1： 采用并行逻辑设计：分别相乘再相加



描述方式2：采用串行化设计，逐个相乘逐个相加：只需要勇1个8位的乘法器和1个16位的加法器。但是速度明显会降低，将需要耗时5个clk才能完成一次运算，并且还需要一个附加信号start。

# 2. 速度优化

一般来说，速度优化比资源优化更重要，需要优先考虑。速度优化包括：FPGA的结构特性、HDL综合器性能、系统电路特性、PCB制版情况等，也包括Verilog的编程风格。下面主要讨论电路结构方面的速度优化方法。

## 2.1 流水线设计

流水线(Pipelining)是一种在复杂组合逻辑之间添加寄存器的方法，是最常用的速度优化技术之一。它能显著地提高设计电路的运行速度上限。

## 2.2 寄存器配平 (Register Balancing)

寄存器配平是使较长路径缩短，较短路径加长，使其达到平衡从而提高工作频率的一种技术。

若设计中，若两个组合逻辑块的延时差别过大，若T1>T2，则总体的工作频率 fmax取决于T1, 即最大的延时模块。对不合理设计进行改进，即将原本设计中的组合逻辑1的部分逻辑转移到组合逻辑2中，使t1≈t2，且T1+T2=t1+t2， T1>t1则总体的工作频率 fmax提高。

## 2.3 关键路径法

关键路径：指设计中从输入到输出经过的延时最长的逻辑路径。优化关键路径是提高设计工作速度的有效方法。

EDA工具中的综合器及设计分析器都提供关键路径的信息以便设计者改进设计。Quartus中的静态时序分析工具可以帮助找到延时最长的关键路径。

## 2.4 乒乓操作法

乒乓操作法是FPGA开发中的一种数据缓冲优化设计技术，可视作另一种形式的流水线技术。乒乓操作本质是使用2倍的硬件资源，通过将数据产生时间和数据使用时间重叠，解决一个数据生产效率低于数据使用效率的问题。是一种拿面积换性能的方法。通过“输入数据流选择单元”和“输出数据流选择单元”按节拍、相互配合的切换，将经过缓冲的数据流“无缝”地送到“数据流运算处理模块。

## 2.5 树形结构法

若要实现A+B+C+D。首先实现AB=A+B，CD=C+D，将AB/CD锁存一个时钟周期再相加。树形结构法和上面面积优化提到的串行化是相反的。