

**数字逻辑实验报告（2）**

无符号数的乘法器设计

一、无符号数的乘法器设计

1、实验名称

无符号数的乘法器的设计。

2、实验目的

要求使用合适的逻辑电路的设计方法，通过工具软件logisim进行无符号数的乘法器的设计和验证，记录实验结果，验证设计是否达到要求。

通过无符号数的乘法器的设计、仿真、验证3个训练过程，使同学们掌握数字逻辑电路的设计、仿真、调试的方法。

3、实验所用设备

Logisim2.7.1软件一套。

4、实验内容

**（1）四位乘法器设计**

四位乘法器Mul4🞨4实现两个无符号的4位二进制数的乘法运算，其结构框图如图1-1所示。设被乘数为b(3:0)，乘数为a(3:0)，乘积需要8位二进制数表示，乘积为p(7:0)。

Mul4🞨4

a(3:0)

b(3:0)

p(7:0)

图1-1 四位乘法器结构框图

四位乘法器运算可以用4个相同的模块串接而成，其内部结构如图1-2所示。每个模块均包含一个加法器、一个2选1多路选择器和一个移位器shl。

图1-2中数据通路上的数据位宽都为8，确保两个4位二进制数的乘积不会发生溢出。shl是左移一位的操作，在这里可以不用逻辑器件来实现，而仅通过数据连线的改变（两个分线器错位相连接）就可实现。



图1-2 四位乘法器内部结构

**（2）32🞨4乘法器设计**

32🞨4乘法器Mul32🞨4实现一个无符号的32位二进制数和一个无符号的4位二进制数的乘法运算，其结构框图如图1-3所示。设被乘数为b(31:0)，乘数为a(3:0)，乘积也用32位二进制数表示，乘积为p(31:0)。这里，要求乘积p能用32位二进制数表示，且不会发生溢出。

Mul32🞨4

a(3:0)

b(31:0)

p(31:0)

图1-3 32🞨4乘法器结构框图

在四位乘法器Mul4🞨4上进行改进，将数据通路上的数据位宽都改为32位，即可实现Mul32🞨4。

**（3）32🞨32乘法器设计**

32🞨32乘法器Mul32🞨32实现两个无符号的32位二进制数的乘法运算，其结构框图如图1-4所示。设被乘数为b(31:0)，乘数为a(31:0)，乘积也用32位二进制数表示，乘积为p(31:0)。这里，要求乘积p能用32位二进制数表示，且不会发生溢出。

Mul32🞨32

a(31:0)

b(31:0)

p(31:0)

图1-4 32🞨32乘法器结构框图

用32🞨4乘法器Mul32🞨4作为基本部件，实现32🞨32乘法器Mul32🞨32。

设被乘数为b(31:0)=(b31b30b29b28**···**b15b14b13b12**···**b4b3b2b1b0)2

乘数为a(31:0)=(a31a30a29a28**···**a15a14a13a12**···**a3a2a1a0)2

=(a31a30a29a28)2🞨228+**···**+**(** a15a14a13a12)2🞨212+**···**+ (a3a2a1a0)2🞨20

所以，

p(31:0)= b(31:0) 🞨 a(31:0)

=b(31:0)🞨((a31a30a29a28)2🞨228+**···**+**(**a15a14a13a12)2🞨212+**···**+ (a3a2a1a0)2🞨20)

= b(31:0)🞨(a31a30a29a28)2🞨228 +**···**+ b(31:0) 🞨 **(** a15a14a13a12)2🞨212 +**···**+ b(31:0) 🞨 (a3a2a1a0)2🞨20

从上述推导可知，Mul32🞨32可以用8个Mul32🞨4分组相乘，然后通过4的倍数位的左移（相当于乘2i），再将左移结果两两相加得到。



**数字逻辑实验报告（2）**

无符号数的除法器设计

二、无符号数的除法器设计

1、实验名称

无符号数的除法器的设计。

2、实验目的

要求使用合适的逻辑电路的设计方法，通过工具软件logisim进行无符号数的除法器的设计和验证，记录实验结果，验证设计是否达到要求。

通过无符号数的除法器的设计、仿真、验证3个训练过程，使同学们掌握数字逻辑电路的设计、仿真、调试的方法。

3、实验所用设备

Logisim2.7.1软件一套。

4、实验内容

**（1）四位除法器设计**

四位除法器实现两个无符号的4位二进制数的除法运算，其结构框图如图2-1所示。设被除数为n2(3:0)，除数为d(3:0)，商为quot(3:0)，余数为rem(3:0)。

Div4

n2(3:0)

d(3:0)

quot(3:0)

rem(3:0)

图2-1 四位除法器结构框图

四位除法器Div4算法步骤如下：

1. 设n1="0000"，将被除数以n1:n2 的形式拼接，除数为d；
2. 重复4次：

将n1:n2左移1位；

if (n1>=d) begin n1= n1-d; n2 (0)=1 end

1. 商和余数的结果为：quot= n2 ；rem= n1 。

四位除法器也可以用4个相同的模块串接而成。每个模块均包含一个减法器、两个2选1多路选择器、一个比较器和一个移位器shl。请参照四位乘法器的设计思路，实现两个无符号的4位二进制数的除法器。

**（2）32位除法器设计**

32位除法器Div32实现两个无符号的32位二进制数的除法运算，其结构框图如图2-2所示。设被除数为n(31:0)，除数为d(31:0)，商为quot(31:0)，余数为rem(31:0)。

Div32

n(31:0)

d(31:0)

quot(31:0)

rem(31:0)

图2-2 32位除法器结构框图

对四位除法器Div4中4个相同的模块之一进行改进，将数据通路上的数据位宽都扩展为32位，得到一个Div1。将32个Div1拼接起来即可实现Div32。

8、思考题

（1）乘法器/除法器中的延时主要取决于加法器/减法器的延时，其它组件延时可忽略不计。假设每个加法器/减法器的延时都为Δt，你所设计的乘法器Mul4🞨4、Mul32🞨4、Mul32🞨32、除法器Div4、Div32的延时各是多少？它们是组合逻辑电路、同步时序逻辑电路还是异步时序逻辑电路？

我所设计的乘法器Mul4🞨4的延时为4\*Δt，Mul32🞨4的延时为4\*Δt，Mul32🞨32的延时为7\*Δt。设计的除法器Div4的延时是4\*Δt，Div32的延时是32\*Δt。它们是异步时序逻辑电路。

（2）通过改变设计，乘法器Mul32🞨32的延时能不能再减少？如果能减少，它的最小值是多少？

分析我们设计的Mul32🞨32乘法器，我们使用了将8个Mul32🞨4的乘法器进行同时运算，然后通过移位，再把8个结果相加，得到最后的结果。来分析其中的消耗：8个并行的Mul32🞨4提供了4个Δt；将8个结果相加提供了3个Δt。

思考后我们发现，如果我们保持乘数b不变，而改变乘数a被分解的份数，能影响最后的Δt。比如我们将a分解成32份，即我们需要32个Mul32🞨1，这32个并行实现，消耗的是1个Δt，然后我们将32个数两两相加，求32个数的和，消耗的是5\*Δt。这样总消耗就是6个Δt。

更一般得，若我们保持b不变，将a分解成32/m份，那么我们就需要32/m个Mul32🞨m（Mul32🞨m和实验中Mul32🞨4用的是同一种设计方法），这32/m个并行实现，消耗m\*Δt。然后我们将这m个数两两相加，消耗的加法器为，这里考虑到我们的m只能取1、2、4、8、16，即消耗的加法器为。即总时间延迟为。

分别将1、2、4、8、16、32带入，算得

故我们由以上设计可以看出当m=1或者m=2时都可以将延时降到最小（并行度比较高）。即我们用Mul32🞨1或Mul32🞨2来搭建Mul32🞨32的计算。（但是为了电路画的简洁和节省版面需要，推荐用Mul32🞨32）。

（3）目前的除法器Div32性能并不好，你是否有性能更好设计？

目前除法器是由32个单个模块串联起来而成的，最大的性能缺陷就是在单次运算的时候由于电路太长（所通过的门数量较多），所以到后面的DIV1的时候会有震荡出现，我们要做的是消除震荡，即尽量减少单次运算所通过门的数量。

我们可以用时钟脉冲来解决这个问题。每来一次时钟脉冲，DIV1模块运算一次，然后运算32次得到结果。为实现这一电路，具体设计思路如下：

1.设计一个产生32个脉冲的发生器。即用一个计数器和一个比较器使得在连续外部时钟输入下，只产生32个脉冲，直到外部清零（装载）信号载入，将计数器清0。

2.主要模块还是DIV1，其中输入d接被除数，输入n1和n2前面接一个寄存器，若外部来了一个清零信号（装载信号），则将外部输入装载入寄存器，否则在32个脉冲发生器的脉冲作用下更新从输出n1o和n2o的反馈值，输出n1o和n2o接一个寄存器，在32个脉冲发生器作用下更新，然后反馈给输入n1、n2。输出的商就是n2o，余数就是n1o。

于是在这种设计的状态下，我们在每一次时钟周期的作用下，只运算了一次，并且把算得的值存入寄存器中，便不会出现一次计算通过门太多而产生的震荡情况。在这种运算的情况下，若想调整运算的速度，只需要调整时钟的频率即可。