

作业说明——计算最大公约数 GCD

1. 要求：

- 设计一个电路，实现最大公约数计算
- 建议使用 python 程序 gcd.py 所示的仅仅依赖相减运算的算法
- 设计测试代码(tb.vhd)，并完成 gcd.vhd 文件的仿真验证工作
- 所需提交的代码在提供的部分代码 gcd.vhd 上修改，
- 在课程网站提交你的作业结果 gcd.vhd 和相应的测试 testbench 代码 tb.vhd

2. 接口说明：

例程对应的 entity 结构为：

```
entity gcd is
    port
    (
        clock, new_input : in std_logic;
        a, b      : in  unsigned(15 downto 0);
        result : out unsigned(15 downto 0):=to_unsigned(0,16);
        busy      : out std_logic
    );
end;
```

各个信号功能说明如下：

- Clock：时钟，在它的上升沿作用下进行数据计算
- New_input：该信号='1'时，表示需要在 clock 上升沿输入待计算的两个数 a 和 b
- Result：运算结果，当 busy='0'时输出最新的运算结果
- Busy：表明当前电路处于运算中，还是运算完成，它为'1'的时候表明运算尚未完成，此时 result 的内容不可用，但它为'0'时，表明运算结束，此时 result 存放的是运算结果（最近一次 new_input='1'时输入数据 a 和 b 的最大公约数）

3. 作业的执行步骤提示

- 阅读理解提供的代码框架，查阅书本和文档，确认里面每句话的含义
- 根据提供的“不完整”的设计代码 gcd.vhd 进行修改，实现所述功能
- 完成修改后，可以在 Diamond 软件环境下完成电路的仿真和验证，或者执行脚本 do.py、do.bat 利用 GHDL+GTKWAVE 来验证你的设计。
- 如果修改正确应该看到类似于如下波形



4. 注意：

你所设计的电路中的 out_busy 和 out_gcd 的实际波形时间可以和上图不同，但运算结

果必须正确，即：但 busy 输出'0'时候（表示运算结束）的 result 出现的结果需要等于之前 new_input='1'时输入数据的最大公约数。当 busy='1'时 result 可以是任意其他数据（因为此时数据还在运算中，结果尚未获得）