

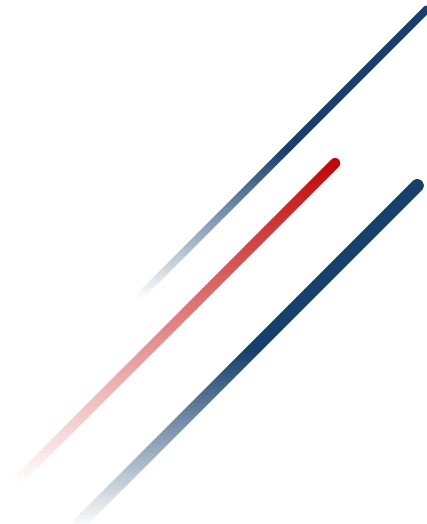
# 计算机组成原理

## 实验3 Booth乘法器

### 马世禹



HITSZ 实验与创新实践教育中心  
Education Center of Experiments and Innovations, HITSZ



# 实验目的

---

- 理解Booth乘法器的原理。
- 掌握Booth乘法器的设计方法。



# 实验题目

---

设计一个接受16位有符号乘数与16位有符号被乘数、输出32位积的Booth乘法器。

输入、输出均采用补码表示。

## 注意：

本实验实现时可采用Verilog的+运算符，但不可直接使用\*运算符，需要使用Booth算法实现乘法器。建议不要使用-实现减法，而采用补码加的方式。



# 实验题目

---

## 附加题(1分)

使用改进的Booth算法完成前述实验。

测试用例会对乘法计算的周期进行考察，要求在10周期以内出结果。



# 实验原理

被乘数：  $[X]_{\text{补}} = X_0.X_1X_2 \dots X_n$

乘数：  $[y]_{\text{补}} = y_0.y_1y_2 \dots y_n$

$[X \cdot y]_{\text{补}} = [z_n]_{\text{补}} + (y_1 - y_0)[X]_{\text{补}}$

$y_i y_{i+1}$ 对部分积的影响为：

$y_i y_{i+1}$	$y_{i+1} - y_i$	操作
00	0	部分积右移一位
01	1	部分积加 $[X]_{\text{补}}$ ，再右移一位
10	-1	部分积加 $[-X]_{\text{补}}$ ，再右移一位
11	0	部分积右移一位



# 实验步骤

---

- 项目框架
  - testbench (testbench.v) —测试模块，模拟输入输出并进行正确性检查
  - booth (booth.v) —Booth 乘法器模块（需要完成）
- 具体步骤
  1. 完成 booth.v
  2. 添加testbench，进行仿真测试。

# 实验步骤

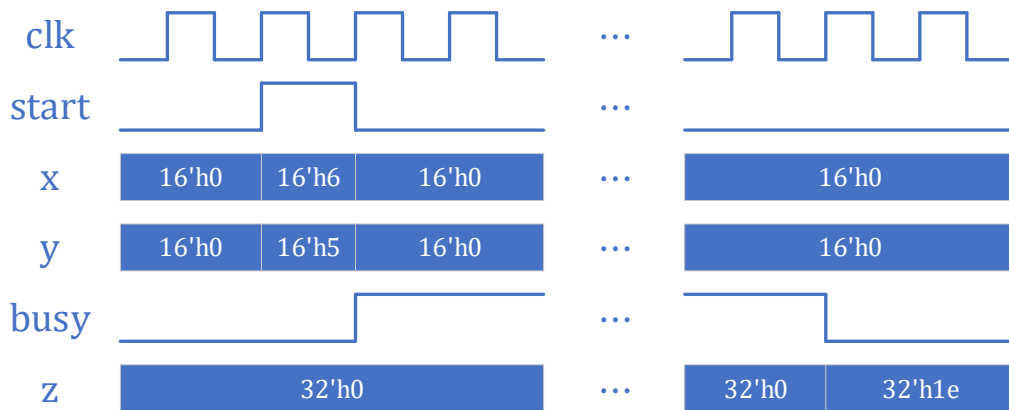
---

- Booth乘法器的I/O定义

属性	名称	含义	位宽
输入	clk	时钟	1
输入	x	被乘数	16
输入	y	乘数	16
输入	start	输入就绪信号	1
输出	z	积	32
输出	busy	忙标志信号	1

# 实验步骤

- Booth乘法器时序



如上图所示，当start信号拉高时，x、y信号有效且只会存在一个周期，乘法器接受到start下一个周期拉高busy信号，直到乘法器工作完成时拉低busy，同时输出运算结果z并一直保持。

请注意，本实验要求，busy的周期不得超过20，即乘法的运算周期不得过长，否则将被判定为无效实现。



# 实验提交

---

提交格式：学号\_姓名.zip

注意：如有出现雷同，雷同者均不得分！

## 提交内容

booth.v : 3分

booth2.v : 1分

实验报告：4分



谢谢！



HITSZ 实验与创新实践教育中心  
Education Center of Experiments and Innovations, HITSZ