# 课程设计任务一

## 任务要求

参考实验板手册，采用Maxplus 2 实现1位全加器设计

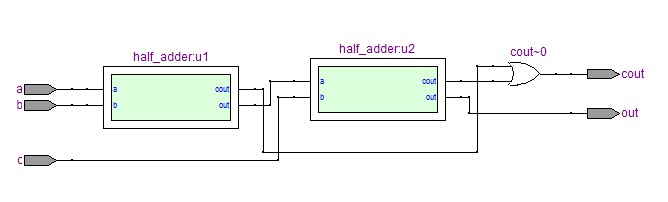
* 具有三个输入端口，，以及来自上一级全加器模块的进位C
* 开发与仿真平台：quartus II 9.0
* 具有
* **功能逻辑真值表如下：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **input** | | | **output** | |
| A | B | C | Out | Cout |
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

## 系统顶层设计

根据真值表以及三个数字的加法竖式运算可得，一位全加器可由两个一位半加器及一个或门构成。半加器应当具有a与b的加法功能（不考虑上级进位）。

RTL级电路如下图所示：



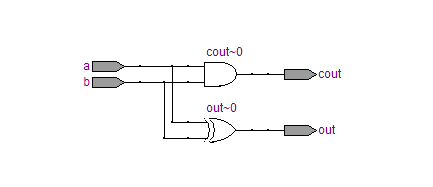
## 底层模块设计

* **半加器模块**

根据半加器所应实现的功能，列半加器真值表如下：

|  |  |  |  |
| --- | --- | --- | --- |
| Input |  |  | Output |
| A | B | Out | Cout |
| 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 1 | 1 | 1 |

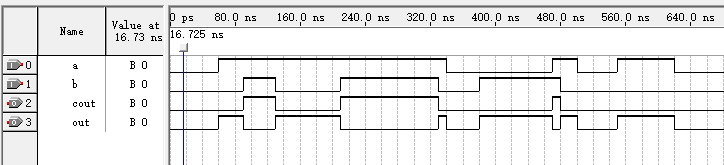
根据真值表设计得到的半加器RTL级电路如下：



## 仿真

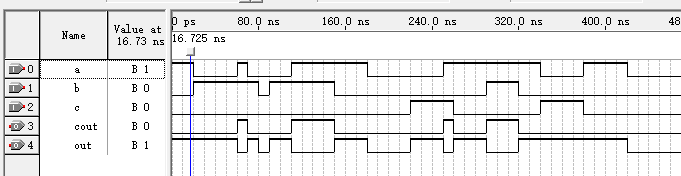
### 功能仿真

1. **半加器功能仿真：**



半加器的功能仿真可以看出满足设计要求

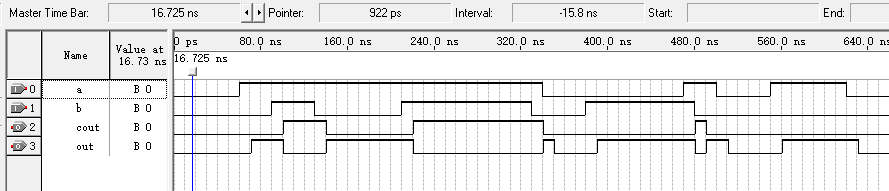
1. **全加器功能仿真：**



**从全加器的功能仿真来看满足设计要求**

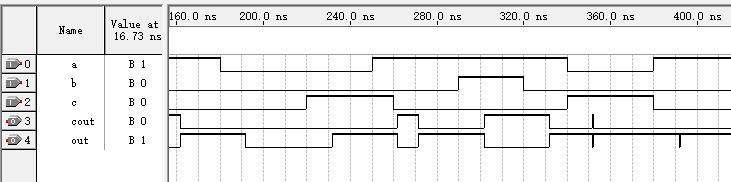
### 时序仿真

1. **半加器时序仿真：**



**可以看出由于所用的器件的速度不高，输入和输出之间存在10ns左右的延迟**

1. **全加器时序仿真：**



**同样由于器件速度的不高，产生了延迟甚至产生了毛刺现象，但由于脉冲时间非常短不产生效果**

## 硬件测试

### 管脚约束

### 硬件配置

# 课程设计任务二

## 任务要求

参考Quartus II 手册，采用 Quartus II (9.0) 实现4位全加器设计

## 系统顶层设计

## 底层模块设计

## 仿真

### 功能仿真

### 时序仿真

## 硬件测试

### 管脚约束

### 硬件配置

# 课程设计任务三

## 任务要求

* 采用Quartus II 实现一个包含2分频，4分频，8分频的分频器模块，编译并得出仿真结果
* 采用Quartus II 实现一个包含分频系数为12的分频器模块，编译后，分别得到功能仿真结果与时序仿真结果

## 系统顶层设计

## 底层模块设计

## 仿真

### 功能仿真

### 时序仿真

## 硬件测试

### 管脚约束

### 硬件配置

# 课程设计任务四

## 任务要求

根据要求设计一个数字秒表

### 功能要求：

1. 计时精度应大于1/100S，计时器能显示1/100S的时间，提供给定时器内部的定时的时钟应大于100Hz，这里选用1Khz。
2. 计时器的最大计时时间为1小时，为此需要6位的显示器，嫌弃的最长时间位59分59.99秒
3. 设置有复位和起/停开关，复位开关用来使计数器清零，做好计时准备。

起停开关的使用方法和传统的机械式计数器相同，即按一下，启动计时器开始计时，再按一下计时终止。

### 初始条件：

按键，时钟信号等。

### 设计实现方法：

自顶向下设计，自底向上实现；

### 编译及调试环境：

Quartus II 9.0

## 功能实现

本设计实现了任务要求的全部任务，能够通过按键（使用单脉冲按键）的来控制计数器的启停，除此之外，还实现了多道计时功能，即能存储多个赛道的时间，并回放。

具体输入对应功能如下：

|  |  |  |
| --- | --- | --- |
| 输入（按键） | 对应功能 | 备注 |
| key\_1 | 启动/停止/复位 | 当初次按下该按键时，秒表启动计时。再次按下，秒表计时终止。再次按下，秒表复位，计数值复位为零。再次按下后，再次启动计数，循环往复。 |
| Key\_2 | 记录时间/回放时间 | 当秒表处于计数状态时，每当按下该键时，存储会当前的时间。并在计数器处于停止状态后，每次按下该键，会从后向前一次再现之前的时间记录。 |

## 系统顶层设计

### 系统设计框图：

系统主要模块以及关系如下图所示，整个系统的时钟源来来自实验板供给的50Mhz时钟，经过预分频模块的分频可以得到准确1KHZ时钟后，时钟计数模块对时钟计数并且将计数值转化为相应的时分秒等单位，将数据流送往数码管输出模块进行输出。另外，按键触发通过按键处理模块之后得到相应的控制事件送往时钟计数模块进而控制数字时钟的模式。

系统主要分为核心控制与按键处理、时钟分频、时钟计数、RAM和显示锁存输出模块。

整个系统的时钟源来自于实验板供给的50MHZ时钟，经过预分频模块的分频后可以得到准确的1KHZ时钟，并送往时钟计数模块。时钟计数模块提供了本数字秒表的基础计时功能，其中包含七个具有异步复位/使能/同步装载的一位十进制/十进制计数器，分别代表当前计时的十分位、分位、十秒位、秒位、百毫秒位、十毫秒位、毫秒位，输出的当前计数值送往核心调度器等待处理。综合出的RAM模块是本系统实现存储多道数据的关键，利用了FPGA中的block ram综合出能够进行同步寻址、读写使能的宽度24bit/深度16的存储阵列。核心控制及按键处理模块是系统设计的中心，接受了来自按键的控制信号，同时利用有限状态机来实现整个系统的逻辑控制，调度需要送往数码管显示的显示信号。显示锁存模块，由于实验电路使用了显示译码器，并且每个数码管之间采用独立式信号引脚，所以功能较为简单，只需要将核心调度模块送来的显示信号进行锁存输出就可以了。

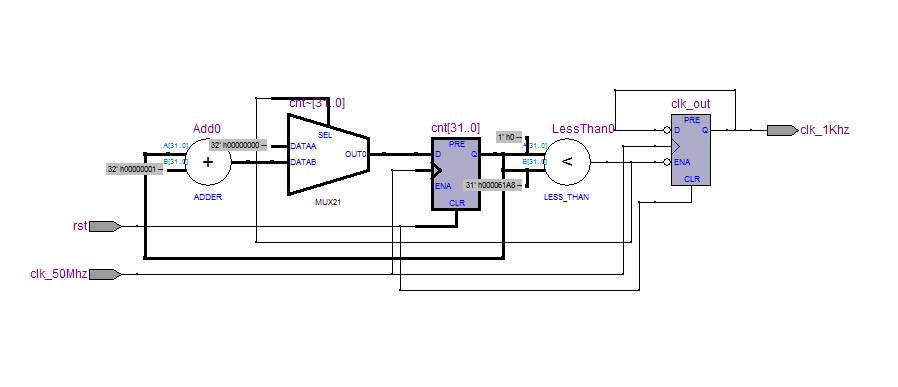


## 底层模块设计

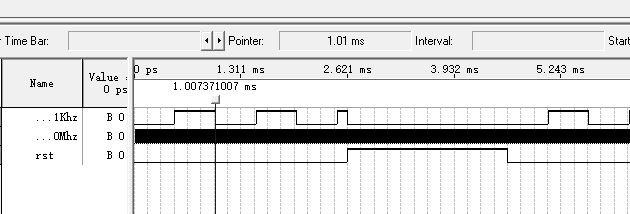
### 时钟分频模块

时钟分频模块的输入为50MHZ输入时钟clk\_50Mhz·，输出为clk\_1Khz。由于是简单的偶数次分频，只需要简单的对输出进行周期为25000次输入时钟计数的周期性反转即可得到1Khz的输出时钟。

**RTL级建模：**



**功能仿真：**



可以看到时钟分频模块能够分频出数字时钟所需要的1KHZ频率时钟且符合精度要求。

### 时钟计数模块

时钟计数模块是实现计时功能的核心，对于从分频模块输入的1KHZ(毫秒)进行计数，得到最终的 六位计时时间。具有同步计数，同步使能/装载，异步复位功能。具体实现上，六位数字分别由六位一位六/十进制计数器来实现。

以一位六进制计数器为例，具有同步计数，同步使能/装载，异步复位功能的一位六进制计数器RTL建模如下:



**功能仿真：**

### RAM存储模块

要实现多道计时的再现，就需要对其进行存储，FPGA中的RAM中有两种，一种是分别在LE中的分布式RAM，当用来配置为存储单元时会耗费板子的逻辑单元；另外一种是集中的Block RAM，显然块RAM适合配置为专用的存储单元。

Quartus II 中自带的LPM(Library of Parameterized Modules) 中有丰富的宏功能模块，可以通过LPM生成单端口RAM。



**功能测试：**

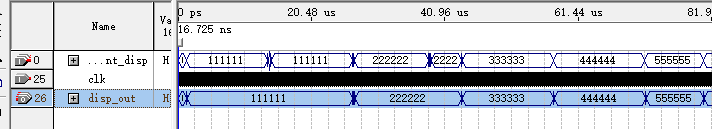
### 数码管显示输出模块

由于实验板电路自己自带有译码电路，并且每位数码管都采用独立信号线。显示管输出模块仅需要对控制单元送出的待输出信号锁存输出即可。



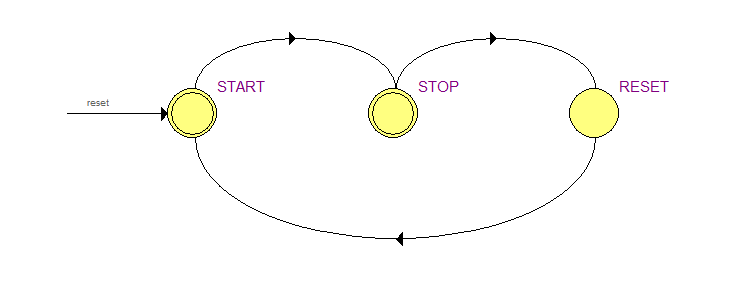
**时序仿真：**

从时序仿真来看，完全能够稳定原待输出信号中可能带来的毛刺 ，消除毛刺带来的影响。



### 核心控制及按键处理模块

Start/stop按键使用了状态机作为时钟的不同状态控制。



## 完整系统的仿真

### 功能仿真

### 时序仿真

## 硬件测试

### 管脚约束

### 硬件配置