



硬件综合实践实验指导书

广东海洋大学数学与计算机学院

沈金伟 编著

2024 年 2 月

目录

一、实验目的	2
二、实验仪器	3
三、实验内容	3
3.1 内容说明	3
3.2 实验要求	3
四、实验过程	4
4.1 Quartus II 平台使用	4
①案例一：一位半加器	4
②案例二：一位全加器	4
③案例三：四位全加器	5
④案例四：四位全加器	6
4.2 电子计时器	8
①简易电子计时器	8
②带功能设置的计时器	11
4.3 简易 CPU 处理器	14
五、说明	16

一、实验目的

数字逻辑是一门实践性、工程性很强的技术基础课。因此，不仅要重视理论教学，更要注重实践技能的培养和训练。

实验是本课程的重要组成部分。通过 40 学时实验，使学生学会查阅产品手册、拟定实验方案、选择与配置实验设备、查明与排除故障和分析实验现象等。培养学生的综合知识运用的能力，强化实践动手能力，加强熟练使用各种设计工具和编程、仿真环境的能力，以及培养学生撰写报告和文档的基本能力。同时，在整个项目的实践过程中，培养学生自主思考、主动研究、团队协作能力、自我学习等能力。结合软件工程人才培养方案，本课程目标如下：

- 1、学习使用 Quartus II9.0。
- 2、使用逻辑方法设计数字电路。
- 3、验证组合逻辑电路和时序逻辑电路设计方法。
- 4、掌握自顶向下设计、自底向上开发思想。

二、实验仪器

DE2 套件、Quartus 软件

三、实验内容

3.1 内容说明

1、组建团队，每组不超过 1 人，上报名单，组长负责小组成员的工作分工以及 **ED2** 开发板的管理。

2、熟读“硬件综合实践实验”相关文档。

3、根据实验要求完成 3 个实验内容，并撰写实验报告(分析、编译、仿真、连线、烧录过程)。

4、编程语言：原理图

5、最后验收：以小组为单位讲解和演示项目的运行情况，按工作量和回答问题结果打分。每位同学必须提交实验报告(小组成员不能雷同，按各自工作实际撰写，项目整体介绍和测试可以共用。)

6、成绩构成：考勤 10%+答辩*50%+报告*40%

3.2 实验要求

(1)使用 Quartus II9.0 平台完成数字电路编译和仿真步骤。

- (2)实现计时器。
- (3)实现简易处理器。

四、实验过程

4.1 Quartus II 平台使用

(1) 实验目标：

在此阶段，主要是让学生熟悉 Quartus-II 开发平台和 ED-2 开发板，主要案例有四个，分别是一位半加器，一位全加器以及两种不同的方法设计的四位全加器。

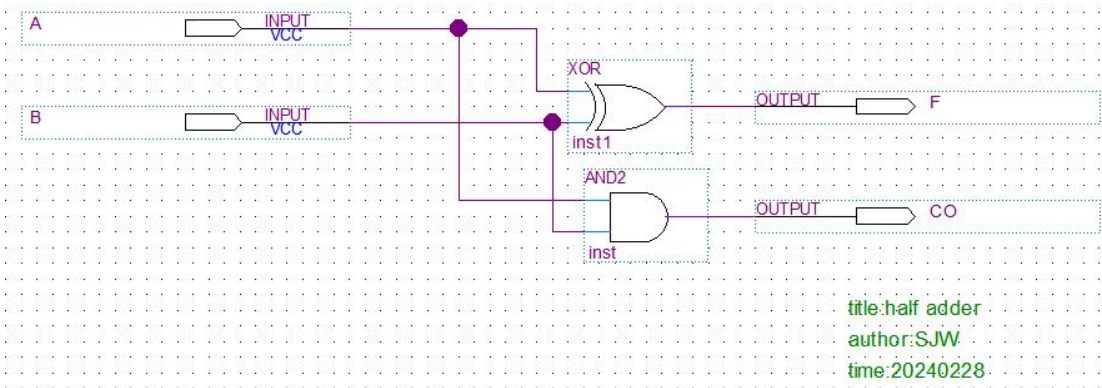
(2) 实验步骤：

①案例一：一位半加器

内容：本案例中用最基本的逻辑门器件构成一位半加器。

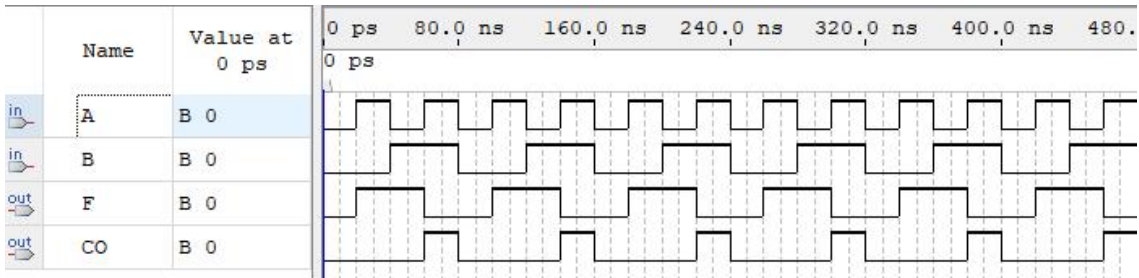
目的：让学生熟悉 Quartus-II 平台中工程的建立流程、图形输入的基本设计方法以及编译中各步骤作用。

一位半加器的顶层模块图如下：



设计说明：此半加器由基本的逻辑门组成，主要实现半加器功能。

时序仿真图如下：

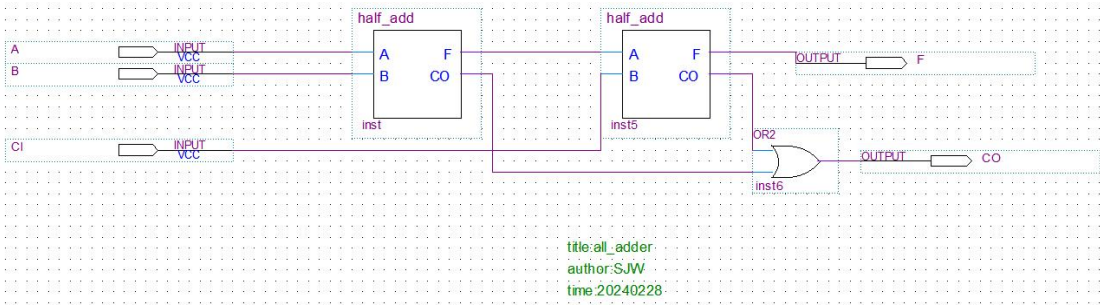


②案例二：一位全加器

内容：本案例中使用已经设计好的一位半加器构成一位全加器，然后再进行功能仿真和时序仿真，以验证设计的正确性。

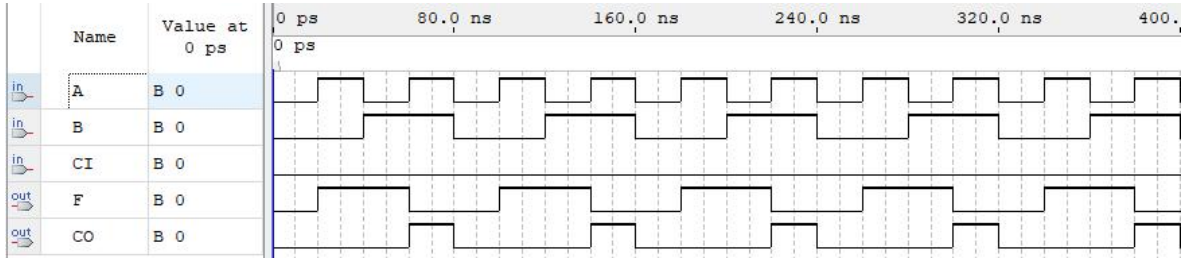
目的：让学生理解层次设计的概念，掌握模块的生成、调用和 Quartus-II 中的仿真方法，包括了解功能仿真和时序仿真的作用和区别。

一位全加器的顶层模块图如下：



设计说明：此一位全加器由两个一位半加器构成，实现一位全加器功能。一位半加器生成模块符号后，可作为非顶层模块供其他的设计使用，这样可以避免重复输入，使设计更简洁。

时序仿真图如下：

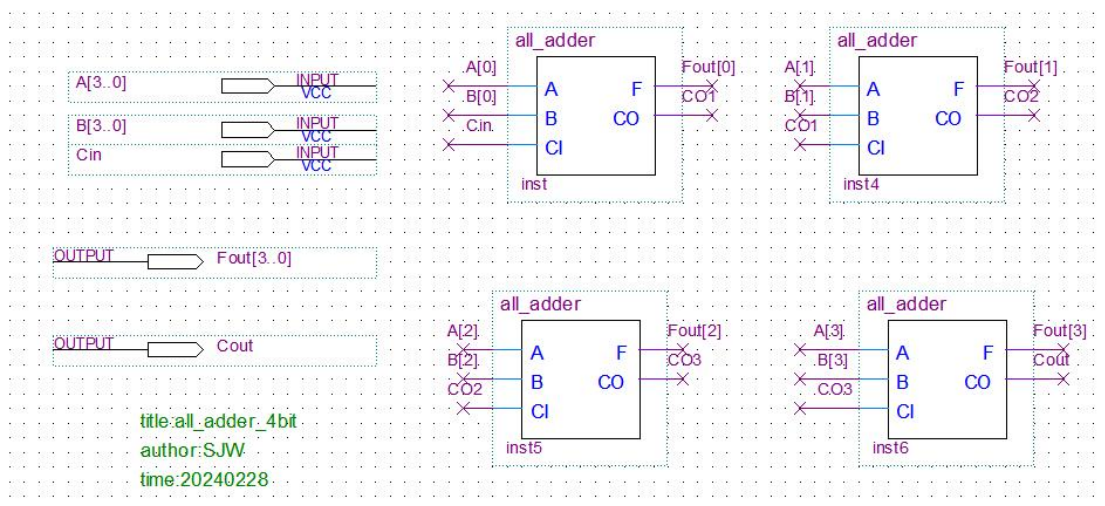


③案例三：四位全加器

内容：让学生利用已经设计好的一位全加器自主设计一个四位全加器，并进行仿真，验证设计的正确性，仿真无误后，将设计下载到 DE-2 板上运行。拨码开关作为输入，LED 灯作为输出，观察结果是否正确。

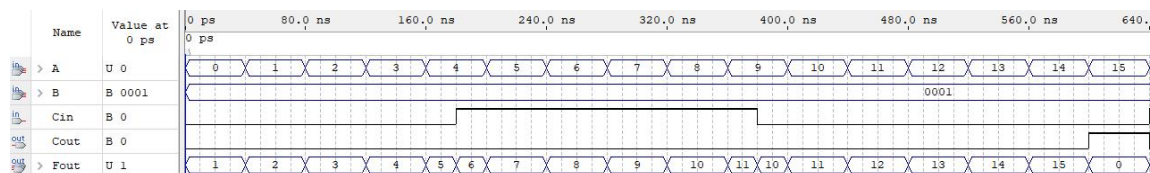
目的：让学生进一步巩固前面所学的知识，同时对 DE-2 板有初步的了解，掌握 Quartus-II 平台中设计的下载方法。

四位全加器的顶层模块图如下：



设计说明：此设计实现四位全加器功能，由四个一位全加器构成，充分体现了 Quartus-II 中的层次化和模块化设计的理念。

时序仿真结果如下：



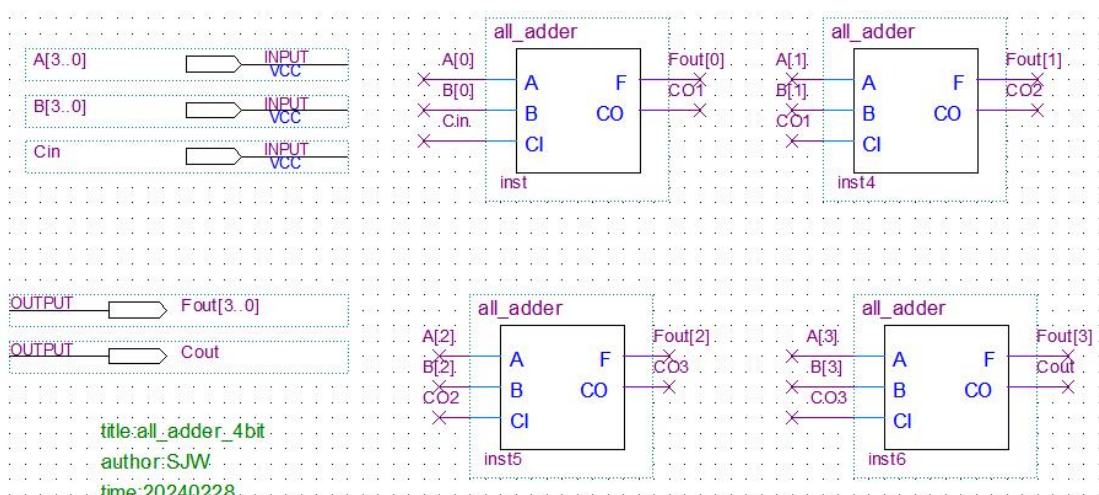
硬件验证：用 9 个拨码开关作为输入，比如 SW0-SW3 作为 A 输入，SW4-SW7 作为 B 的输入，SW9 作为进位输入 C 的输入，5 个流水灯作为输出，如 LEDR0-LEDR3 作为 SUM 的输出，LEDR4 作为输出进位 Cout 的输出，将设计下载到 DE-2 板上后观察结果正确。

④案例四：四位全加器

内容：调用 Quartus-II 平台中的宏功能模块来实现四位全加器，将此设计与前一个案例中的四位全加器从设计的方法、复杂度和仿真结果进行比较，发现两种设计方式的优缺点。最后下载到 DE-2 板上进行验证，拨码开关作为输入，使用七段数码管来显示运算结果 SUM。

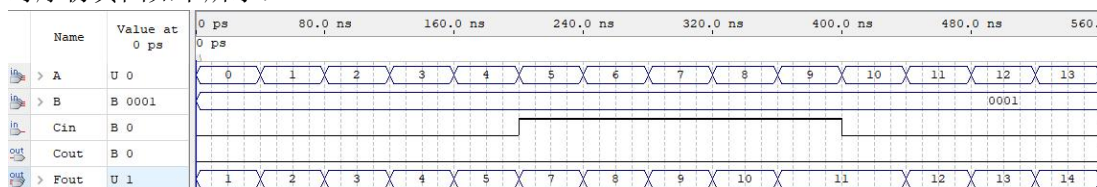
目的：让学生掌握宏功能模块的使用方法，并对 ED-2 板有进一步的了解。

利用宏功能模块实现的四位全加器的顶层模块图如下：



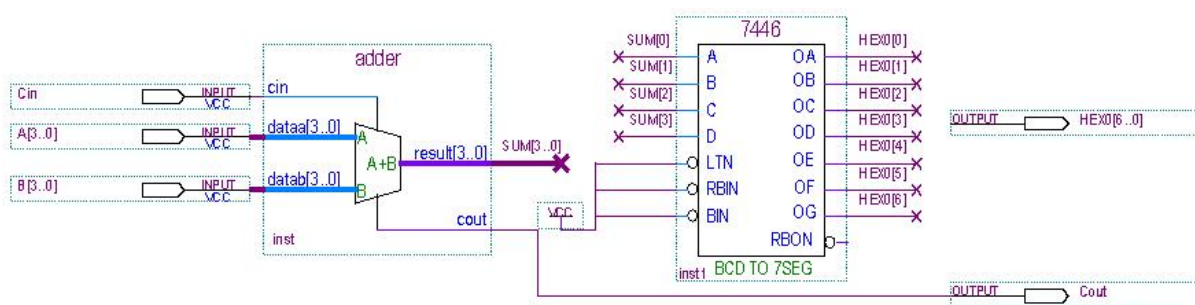
设计说明：此设计调用宏功能模块实现了四位全加器的功能，但是要使用其段数码管来显示运算结果 SUM，则需要在顶层模块中添加数码管驱动模块 7446。

时序仿真图如下所示：



硬件验证：用 9 个拨码开关作为输入，比如 SW0-SW3 作为 A 输入，SW4-SW7 作为 B 的输入，SW9 作为进位输入 Cin 的输入，5 个流水灯作为输出，如 LEDR0-LEDR3 作为 SUM 的输出，LEDR4 作为输出进位 Cout 的输出，将设计下载到 DE-2 板上后观察结果。

带数码管驱动的四位加法器顶层模块图如下：



TITLE 宏功能模块实现的四位全加器			
COMPANY 软件技术系			
DESIGNER 张健 刘桃丽			
NUMBER	1.00	REV	A
DATE	2010年11月25日	SHEET	1 OF 1

设计说明：7446 为共阳极七段数码管驱动模块，只能对 4 为 BCD 码进行译码，因此此次加法器在进行运算的时候应该使两数相加之和小于 10，这样才能正确的显示运算结果。

硬件验证：用 9 个拨码开关作为输入，比如 SW0-SW3 作为 A 输入，SW4-SW7 作为 B 的输入，SW9 作为进位输入 C 的输入，1 个七段数码管作为输出， LEDR1 作为输出进位 Cout 的输出，将设计下载到 DE-2 板上后观察结果。

4.2 电子计时器

（1）实验目标：

本阶段主要以模仿为主，旨在训练学生综合设计能力。本阶段中首先给定学生一个功能相对简单的案例进行分析和学习，然后在此基础上对该案例的功能进行进一步的完善。主要使用的案例有两个，一个是给定学生的案例--简单的计时器，第二个则是在第一个案例的基础上设计的具有外部功能设定的计时器。

（2）实验步骤：

① 简易电子计时器

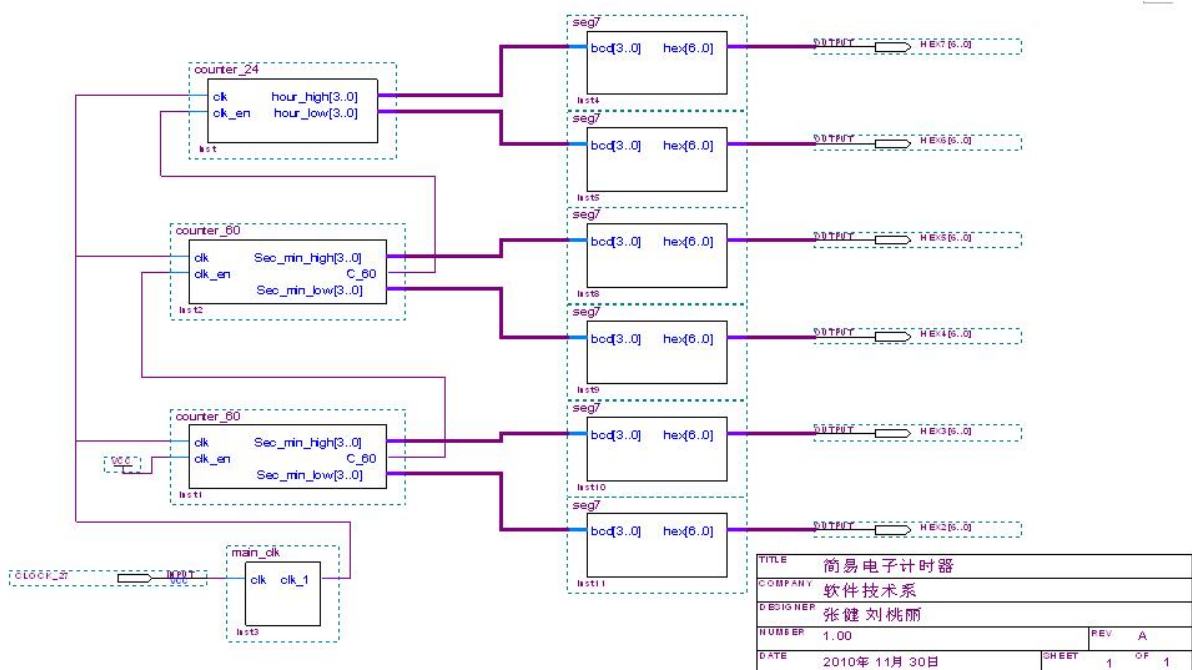
内容：计时器实现简单的按秒计数功能，从 0 时 0 分 0 秒开始作加法计时，小时、分钟和秒的计数各使用 2 个数码管来显示。这个设计采用分模块分层次的方法来完成。

目的：此案例主要让学生了解稍复杂的工程设计方法，培养模块设计和层次设计的习惯。

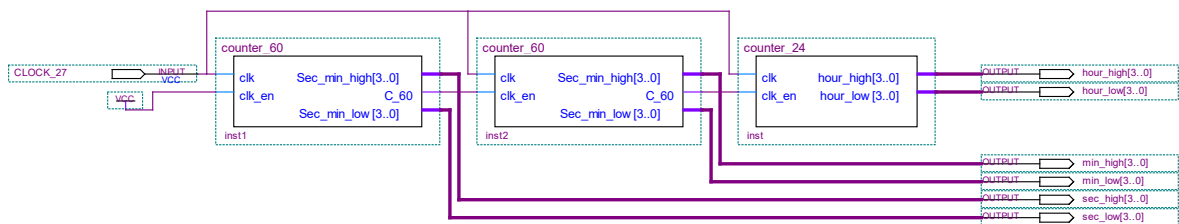
设计说明：整个设计分 10 个模块，包括：

- 1) 一个分频模块 `main_clk`，主要产生一个 1Hz 的时钟周期，作为整个计时器的标准计时周期；
- 2) 两个相同的 60 进制计数器 `counter_60`，一个作为秒计数器，一个作为分钟计数器；一个 24 进制的计数器 `counter_24`，用来作为小时计数器。
- 3) 六个七段数码管译码驱动模块 `seg7`，分别用来显示小时、分钟、秒的个位和十位。

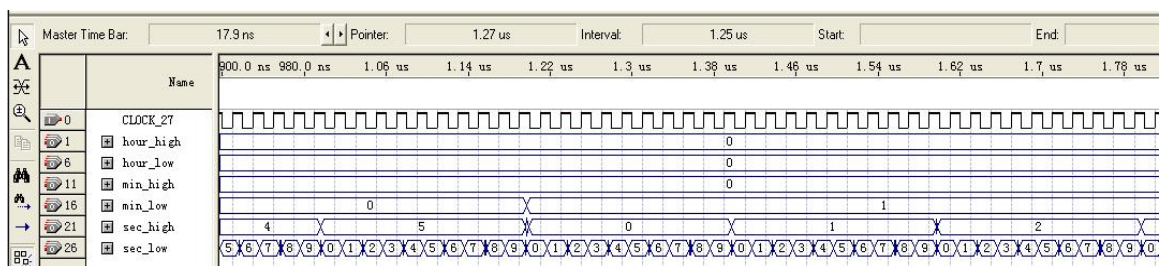
简单计时器的顶层模块图如下页图所示：



时序仿真说明：由于分频器的存在，给仿真带来一定不便，因此仿真是将分频器模块 **main_clk** 去掉，同时此设计着重在于三个计数器的设置，因此将 6 个七段数码管译码驱动模块，设计图修改后如下所示：

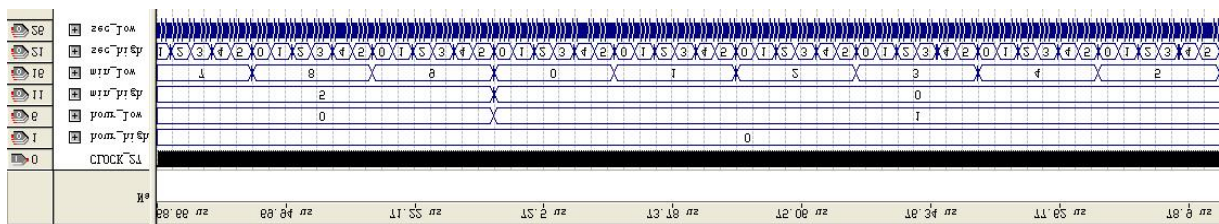


由于仿真的时间较长，为了便于观察，仿真图分 3 个展示仿真结果，如下所示：



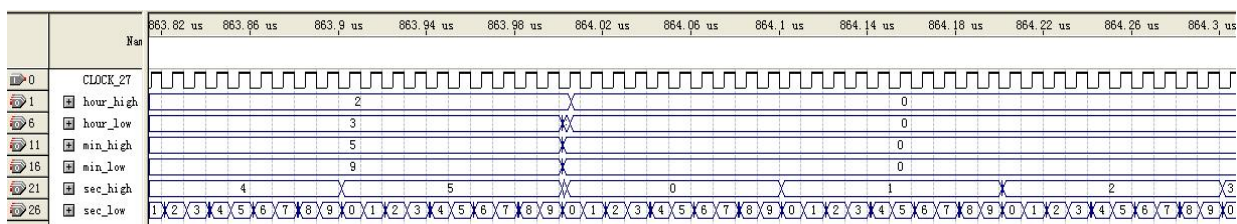
简易计时器时序仿真图一

图一主要显示的是秒的计时和到分钟的进位，当秒计数到 59 时，下一个时钟周期分钟加 1，符合设计要求。



简易计时器时序仿真图二

图二主要显示的是分钟的计时和到小时的进位，当分钟计数到 59 时，下一个时钟周期小时数加 1，符合设计要求。



简易计时器时序仿真图三

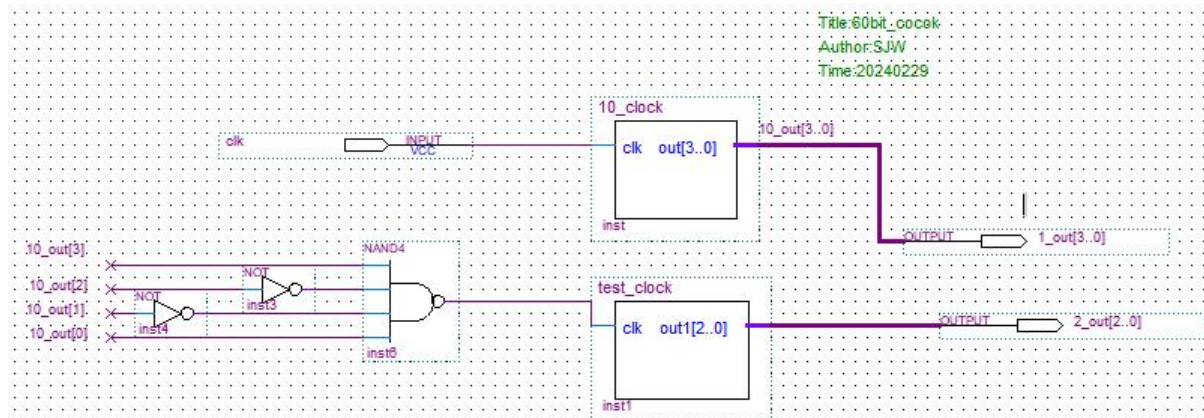
图三显示计时器经一轮计时后，即计时达到 23 时 59 分 59 秒后，再从 0 时 0 分 0 秒重新开始计数，符合设计要求。

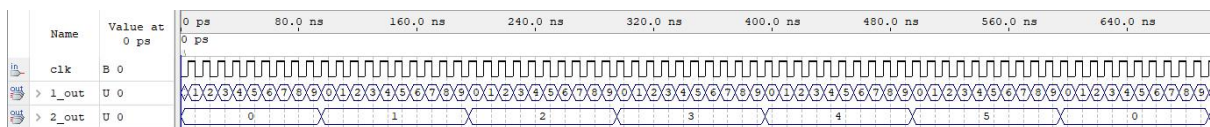
硬件验证：指定 DE-2 板上的 27MHz 的晶振作为输入时钟，6 个 7 段数码管分别作为小时、分钟和秒的计时输出，将设计下载到 DE-2 板上后观察结果。

参考思路：

(1) 分频模块 main_clk：由于 DE2 开发板自带 2 个时钟脉冲：CLOCK_27、CLOCK_50，必须将 27MHz/50MHz 的晶振脉冲分频产生一个 1Hz 的时钟周期，即设计一个 27M/50M 的计数器。计数器可以自主设计，也可以考虑使用宏模块设计。

(2) 60 进制/24 进制计数器：①考虑触发器组成时序逻辑电路。②考虑将 60 进制分成 2 位数，6 进制计数器+10 进制计数器+组合逻辑电路。





(3) 数码显示器：N 进制转 7 进制计数器。考虑集成电路器件 7446/7447，功能表如下：

Decimal or Function	Inputs						BI/RBO (Note 1)	Outputs							Note
	LT	RBI	D	C	B	A		a	b	c	d	e	f	g	
0	H	H	L	L	L	L	H	L	L	L	L	L	L	H	(2)
1	H	X	L	L	L	H	H	H	L	L	H	H	H	H	
2	H	X	L	L	H	L	H	L	L	H	L	L	H	L	
3	H	X	L	L	H	H	H	L	L	L	L	H	H	L	
4	H	X	L	H	L	L	H	H	L	L	H	H	L	L	
5	H	X	L	H	L	H	H	L	H	L	L	H	L	L	
6	H	X	L	H	H	L	H	H	H	L	L	L	L	L	
7	H	X	L	H	H	H	H	L	L	L	H	H	H	H	
8	H	X	H	L	L	L	H	L	L	L	L	L	L	L	
9	H	X	H	L	L	H	H	L	L	L	H	H	L	L	
10	H	X	H	L	H	L	H	H	H	H	L	L	H	L	
11	H	X	H	L	H	H	H	H	H	L	L	H	H	L	
12	H	X	H	H	L	L	H	H	L	H	H	H	L	L	
13	H	X	H	H	L	H	H	L	H	H	L	H	L	L	
14	H	X	H	H	H	L	H	H	H	H	L	L	L	L	
15	H	X	H	H	H	H	H	H	H	H	H	H	H	H	
BI	X	X	X	X	X	X	L	H	H	H	H	H	H	H	(3)
RBI	H	L	L	L	L	L	L	H	H	H	H	H	H	H	(4)
LT	L	X	X	X	X	X	H	L	L	L	L	L	L	L	

(4) 尤其注意时钟跳转时候电路信号变化，为防止竞争冒险问题，建议使用脉冲触发方式。

②带功能设置的计时器

- 内容：1) 本设计要求该电子计时器具有四种工作模式：正常计时、从外部设置当前的小时数、从外部设置当前的分钟数、从外部设置当前的秒数，同时具有可逆的计时功能；
- 2) 在设置当前某位时间的时候，显示该时间的两个七段数码管按 1Hz 的频率进行闪烁，提醒当前设置的是哪个时间数，其余的数码管则处于暂停状态；
- 3) 使用 DE-2 板上的一个拨码开关要来设置当前计时器进行加法计时和减法计时状态，根据某个按键按下的次数来决定当前计时器的工作状态，未按下时为正常的加、减计数状态，按下一次为从外部设置当前的小时数，按下两次为从外部设置当前的分钟数，按下三次为从外部设置当前的秒数。在进行外部设置时间数的阶段，使用另外一个按键手动按下一次来作为时间数的加 1、减 1 的手动计数脉冲。

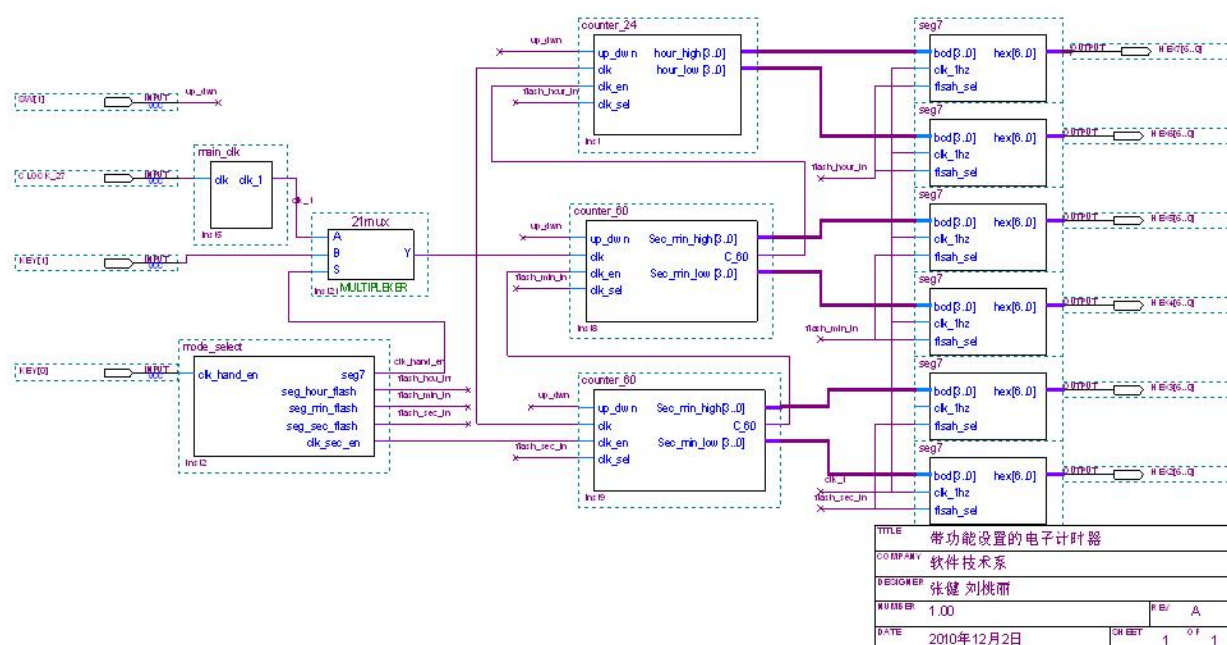
目的：本设计主要由学生在简易计时器的基础独立完成，主要使学生掌握基本的开发方法，训练学生的独立自主的开发能力。

设计说明：本设计共有 12 个模块组成，包括：

- 1) 一个分频模块 `main_clk`，主要产生一个 1Hz 的时钟周期；
- 2) 一个状态发生器 `mode_select`，主要用来产生四种工作模式的控制信号和秒计数模块的计数使能控制信号；

- 3) 一个二选一的数据选择器模块 21mux, 用来选择当前输入的是手动脉冲还是 1Hz 的标准秒脉冲;
- 4) 两个 60 进制的可逆计数模块 counter_60, 并带有暂停功能, 分别用来对秒和分钟进行计数;
- 5) 一个 24 进制的可逆计数模块 counter_24, 并带有暂停功能, 用来对小时进行计数;
- 6) 六个带有闪烁控制端的七段数码管译码驱动模块, 用来显示对应时间数的计数值, 并在外部输入时间数的时候对应的数码管进行频率为 1Hz 的闪烁。

该设计的顶层模块图如下页所示:



为了便于观察, 仿真时同样去掉了分频器 main_clk 模块, 将 CLOCK_27 直接连接到 21mux 模块的 A 输入端, 在各个计数器模块的输出端引出输出小时、分钟和秒的高位和低位输出引脚, 分别为 hour_high 和 hour_low、min_high 和 min_low、sec_high 和 sec_low。修改顶层图后仿真图分 3 个展示仿真结果, 各引脚的含义如下:

CLOCK_27 表示系统标准的时钟;

KEY[1]表示外部手动输入时钟;

KEY[0]表示模式选择;

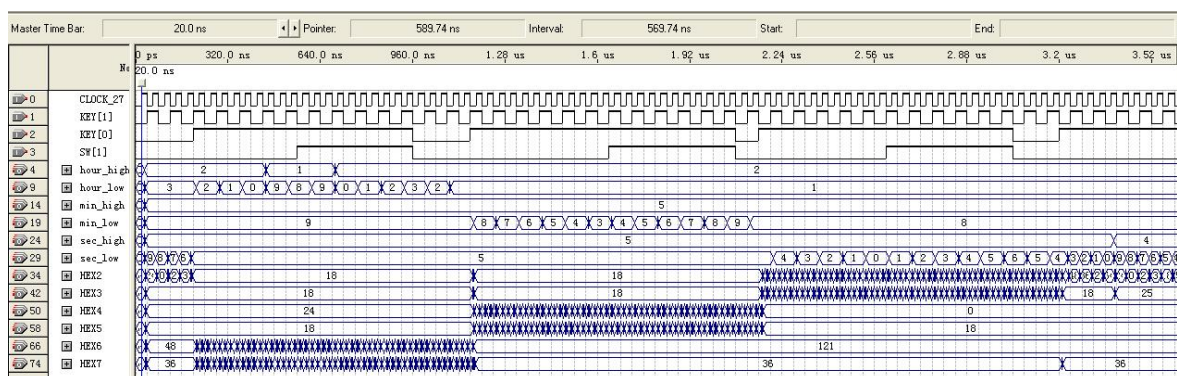
SW[1]表示加法、减法计数选择;

hour_high 和 hour_low 表示小时的高位和地位, HEX7 和 HEX6 表示小时的高位和地位经过 seg7 译码后的值;

min_high 和 min_low 表示分钟的高位和地位, HEX5 和 HEX4 表示分钟的高位和地位经过 seg7 译码后的值;

sec_high 和 sec_low 表示秒的高位和地位, HEX3 和 HEX2 表示秒的高位和地位经过 seg7 译码后的值;

时序仿真结果如下 3 图所示:



带功能设置的计时器时序仿真图一

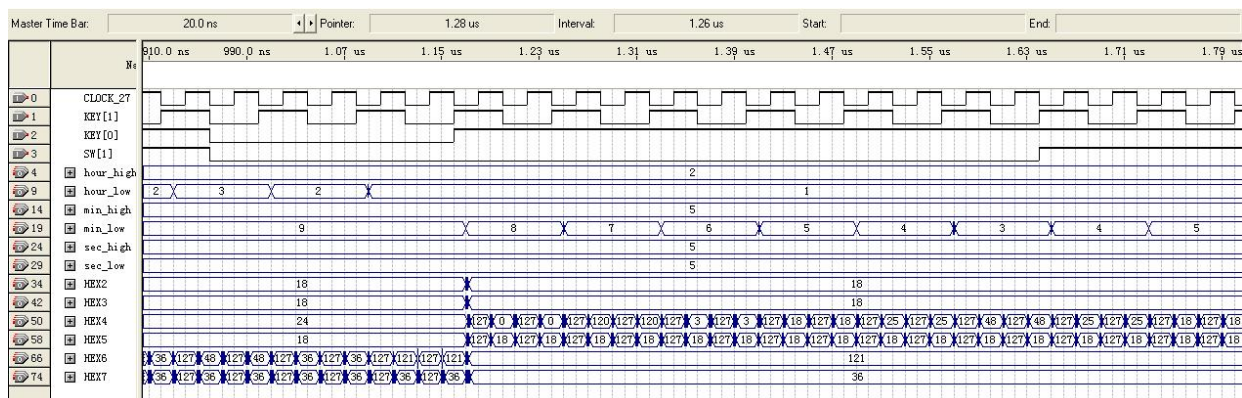
从图一可以看出：

当 KEY[0] 的第一个上升沿到来时，小时计时 hour_high 和 hour_low 按照 KEY[1] 的周期开始执行，并根据 SW[1] 的状态来进行加法或者减法计时，其他的计时器则处于暂停状态，同时小时计时数码管 HEX6 和 HEX7 开始按照 KEY[1] 的频率输出闪烁值，其他数码管处于暂停状态；

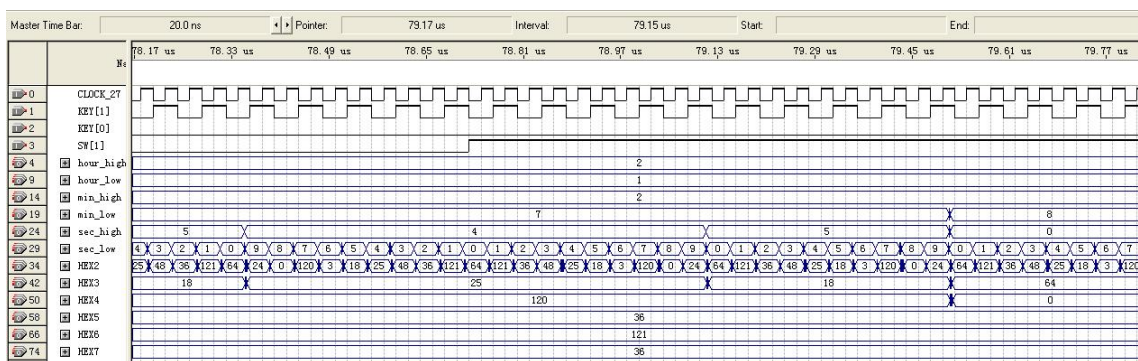
当 KEY[0] 的第二个上升沿到来时，分钟计时 min_high 和 min_low 按照 KEY[1] 的周期开始执行，并根据 SW[1] 的状态来进行加法或者减法计时，同时分钟计时数码管 HEX5 和 HEX4 开始按照 KEY[1] 的频率输出闪烁值，其他是计时器和数码管处于暂停状态；

当 KEY[0] 的第三个上升沿到来时，秒计时 sec_high 和 sec_low 按照 KEY[1] 的周期开始执行，并根据 SW[1] 的状态来进行加法或者减法计时，同时秒计时数码管 HEX6 和 HEX7 开始按照 KEY[1] 的频率输出闪烁值，其他的计时器和数码管暂停；图二为部分细节放大图，显示数码管闪烁的细节。

当 KEY[0] 的第四个上升沿到来时，系统按照根据 CLOCK_27 的频率处于正常的工作状态。



带功能设置的计时器时序仿真图二



带功能设置的计时器时序仿真图三

图三显示的是在正常工作情况下，系统根据 SW[1] 的状态进行加、减计数的情况，当 SW[1] 为 0 时，系统做减法计时，为 1 时做加法计时。

硬件验证：指定 DE-2 板上的 27MHz 的晶振作为输入时钟，6 个 7 段数码管分别作为小时、分钟和秒的计时输出，按键 KEY0 作为模式选择按钮，按键 KEY1 作为外部手动时钟输入按钮，拨码开关 SW1 作为可逆计数的控制开关，将设计下载到 DE-2 板上后观察结果。

4.3 简易 CPU 处理器

(1) 实验目标：

试设计一个简易的处理器，该处理器处理的数据宽度是 4bit，可以实现算术加法、算术减法、逻辑与、逻辑或、逻辑非、逻辑与非、逻辑或非和逻辑异或共 8 种运算。

用 8 条指令表示 8 种运算，2 个操作数分别是 A[3..0] 和 B[3..0]，运算结果是 Y[3..0] 以及标志位 OVERFLOW。

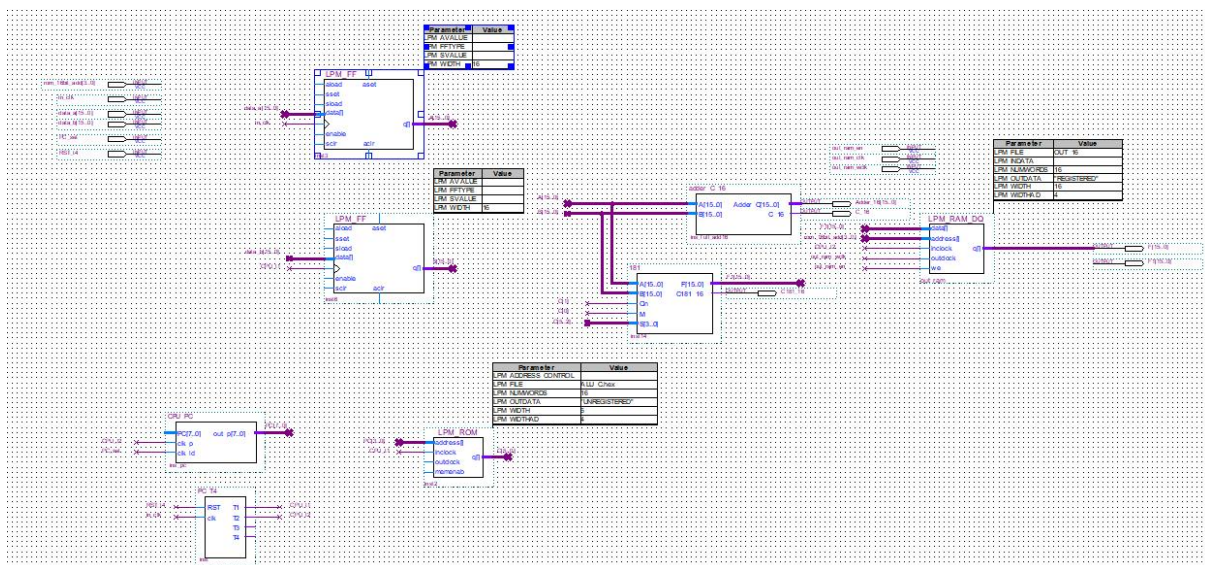
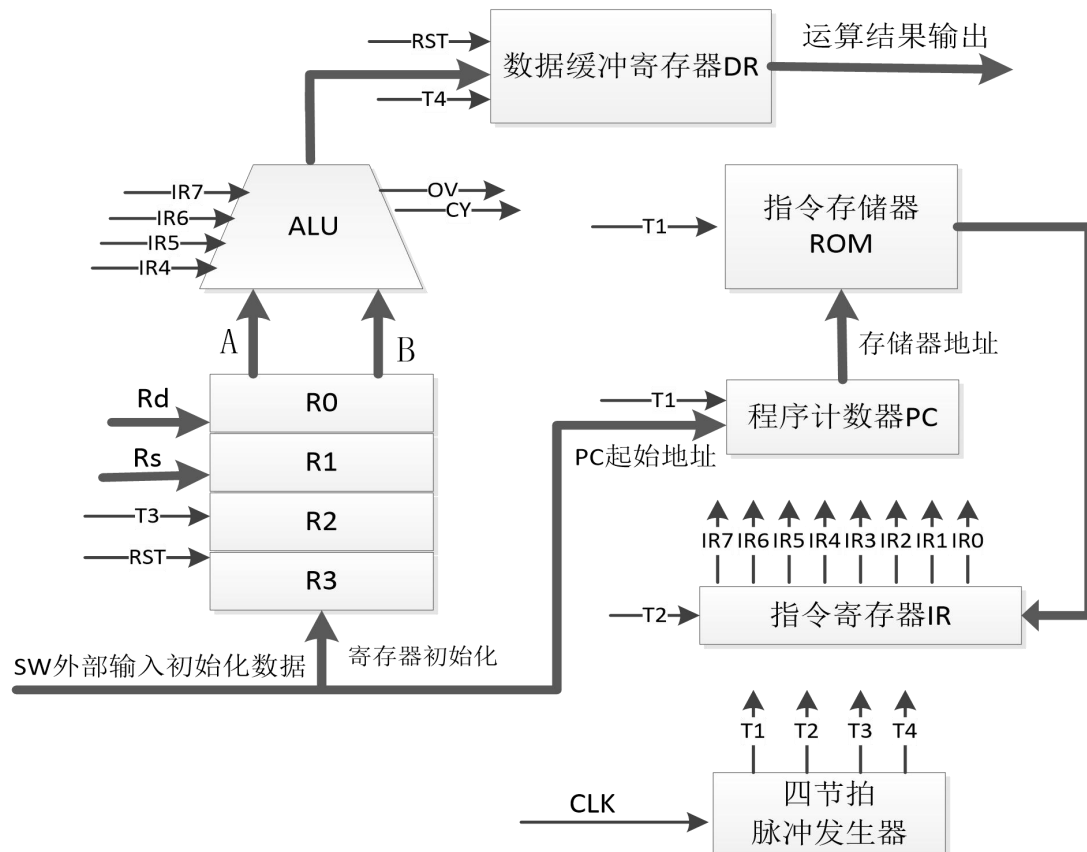
(2) 实验步骤：

1) 设计要求

- (1) 应用 QUARTUS 软件和模块化、层次化的设计方法进行设计；
- (2) 对各模块进行必要的仿真验证；
- (3) 设计结果需要在 DE2 开发板上进行硬件验证；
- (4) 提交完整的设计报告，包括设计原理图；
- (5) 依据 (3)、(4) 和现场答辩确定课程成绩，其中设计报告占 40%，硬件验证和答辩占 60%。

2) 实验参考

参考数据通路如下：



①节拍发生器：分频器（N 进制计数器）

②指令存储器：预存外部指令(宏模块 ROM 或者读取外部文件)

③程序计数器：+1 加法器

④指令寄存器：译码器

⑤寄存器：触发器

⑥ALU:逻辑电路

注意：各个时钟节拍有效信号控制各个模块。

五、说明

以上要求是本课程考核的最低要求，达到该要求的最高得分是 70 分。同学可以在此基础上任意添加处理器的功能和性能要求，如算术运算可以考虑是补码运算，输入、输出采用存储缓冲等等，直至完整的全功能的处理器。依据设计的复杂度和设计质量给出课程成绩评定。