

装

订

线

**学生实验实习报告册**

|  |  |
| --- | --- |
| 学年学期： | 202 -202 学年🞏春■秋学期 |
| 课程名称： | 数字电路与逻辑设计实验A |
| 实验项目： | 基于FPGA的数字电子钟的设计与实现 |
| 学院和专业： | 计算机科学与技术学院  计算机科学与技术专业 |
| 班 级： |  |
| 姓 名： |  |
| 学 号： |  |
| 序 号： |  |
| 指导教师： |  |

**重庆邮电大学教务处制**

1. 系统顶层模块设计

系统功能

该系统功能可分为基础功能和拓展功能。

基础功能为秒、分、小时的计时，即从00:00:00计时到23:59:59，接着再从23:59:59到00:00:00，依次循环。

本系统有9个拓展功能，分别是：

1. 整点报时：当整点时(即分钟和秒钟同时为零)，会触发蜂鸣器1秒钟；
2. 星期显示：FPGA板上LED0-LED6分别表示周一到周日（系统第一次启动默认为周一），假设当前是周一，那么LED0会亮，LED1-LED6处于灭的状态；
3. 上下午显示：FPGA板上LED7表示当前时刻是否为上午。LED7亮表示当前时刻为上午，否则为下午；
4. 6、9补断：数字6、9的显示会更加美观；
5. 时钟暂停：可以通过key4按键使时钟停止不再计时；
6. 校时：可以通过按键key1、key2、key3分别对时钟的秒、分、时进行校时；
7. 置零：可以通过按键key5对时钟进行置零操作；
8. 加杠：时钟的显示格式为时-分-秒，例如：15-12-20；
9. 闹钟：用户可以通过拨动S4开关进入退出闹钟模式，通过S2、S3对闹钟的分和小时进行设置。当时钟到达设置的闹钟时间时，蜂鸣器会被触发一分钟；

设计思路

本系统可以分为13个功能模块。分别为：

1. 分频模块：将50Mhz的频率分成1hz、500hz、1khz，供后续模块使用；
2. 计时模块：实现该系统的计时功能，输出该系统的时，分，秒；
3. 上下午模块：对计时模块的输出进行分析，判断当前时刻是否是上午；
4. 星期模块：根据计时模块输出的时位的进位信号，实现星期的显示功能；
5. 计数模块：实现模M计数（M为使用到的数码管数）；
6. 位选模块：根据计数模块的输出指定当前时刻FPGA板上哪个数码管发亮；
7. 段选模块：根据计数模块的输出指定该数码管要显示的数字；
8. 闹钟模块：根据用户的输入设计闹钟时间；
9. 选择模块：根据按键的输入选择要输出的段选信号；
10. 比较模块：两个时钟进行比较，输出比较结果；
11. 整点报时模块：根据计时模块的输入判断当前时刻是否是整点，输出判断结果；
12. 蜂鸣模块：输入若干条件，若输入条件满足，则触发蜂鸣器；
13. 译码模块：将8421BCD码转换成成数码管要显示相应数字的段码；

设计框图

白板上的字

描述已自动生成

图1 系统顶层设计模块

图示, 示意图

描述已自动生成

图2 系统顶层设计电路图

1. 分频模块电路设计及仿真
2. 模块功能

将50Mhz分频成1khz和1hz，以供后续其它模块实验。

1. 设计思路

模100分频模块设计思路，74390芯片可以实现模2、5、10、20、50、100分频。通过74390可以获得输入电路的模100分频。

分频模块设计思路如图，其中m100使用上面的模100分频模块实现，模2和模5使用74390芯片实现。

图示, 示意图

描述已自动生成

图3 分频模块设计思路图

1. 设计结果（电路）

图示, 示意图

描述已自动生成

图4 模100设计电路图

图示, 示意图

描述已自动生成

图5 分频模块设计电路图

1. 测试结果

我们将该模块的输入设置为50mhz，然后将输出分别接到LED灯上。我们可以发现，当把1khz接到LED灯上，LED灯会长亮；当把1hz接到LED灯上，LED灯会以1秒1次的频率发亮。说明该模块设计正确。

1. 计时模块设计及仿真
   1. 分、秒计时模块（模60计数）
2. 模块功能（计数、进位）

该模块的有效计数状态是0到59，输出为0000 0000到0101 1001。然后从59再到0，依次循环。同时，该模块还需要输出一个进位信号。即满60进1，当该模块作为秒计时模块时，向分钟模块进位；当该模块作为分钟模块时，向小时模块进位。

1. 设计思路

74390芯片的特性是异步清零，清零端高电平有效。当芯片输出为0110 0000(即60)时，实现清零功能。即将q[5]、q[6]相与经过延时器延时后输出到清零端。当该模块清零时，需向上一位进位，所以将q[5]、q[6]相与的结果作为进位端输出。

1. 设计结果（电路）

图示, 示意图

描述已自动生成

图6模60计数电路图

1. 仿真测试

日历

描述已自动生成

图7模60计数模块仿真图（一）

日历

描述已自动生成

图8模60计数模块仿真图（二）

* 1. 小时计时模块（模24计数）

1. 模块功能

该模块的有效计数状态是0到23，输出为0000 0000到0010 0011。然后从23再到0，依次循环。同时，该模块还需要一个进位信号。即满24进1，作用是为实现星期显示模块做准备。

1. 设计思路

74390芯片的特性是异步清零，清零端高电平有效。当芯片输出为0010 0100(即24)时，实现清零功能。即将q[2]、q[5]相与经过延时器延时后输出到清零端。当该模块清零时，需向上一位进位，所以将q[2]、q[5]相与的结果作为进位端输出。

1. 设计结果（电路）

图示

描述已自动生成

图9 模24计数电路图

1. 仿真测试

图片包含 图形用户界面

描述已自动生成

图10 模24模块仿真图（一）

图片包含 图表

描述已自动生成图11 模24模块仿真图（二）

1. 数码管动态显示模块
   1. 动态显示模块的设计

动态显示模块其中含义扫描模块、位选模块、数据选择模块、译码模块四个子模块。

扫描模块和位选模块配合实现使某个数码管被选中的功能。扫描模块为模8的计数器，输出0到7的数字。扫描模块配合位选模块。位选模块为3线-8线74138译码器，该芯片输出只有一个低电平。同时我们使用的是共阴极数码管，也就是8个数码管在同一时刻只有一个数码管被点亮。将扫描模块的输出作为位选模块的输入，实现8个数码管轮流被点亮。同时我们可以将扫描频率增加至1khz，这样8个数码管看起来就像是同时被点亮了。

扫描模块和数据选择模块以及译码模块实现使被选中的数码管显示特定的数字的功能。假设扫描模块的输出是5，那么意味着此时此刻第5个数码管被点亮了。扫描模块和数据选择模块以及译码模块就决定了第5个数码管显示什么样的数字。数据选择模块由4个74151芯片组成 。74151芯片的输入端包括3个地址选择端和8个数据输入端。其功能是在地址选择端的控制下将8个数据中的某一个输出。这里4个74151芯片的地址选择端全部接扫描模块的输出，数据端由计时模块的输出控制。译码模块的输入为4个74151芯片的输出，输出为8位段码。

图示

描述已自动生成

图12 数码管动态显示模块功能框图

* 1. 扫描模块cnt8

1. 模块功能

扫描模块为一个模8的计数器，有效计数状态为0到7 。

1. 设计思路

74390芯片的特性是异步清零，清零端高电平有效。当芯片输出为0000 1000(即8)时，实现清零功能。即将counter8\_04[3]相与经过延时器延时后输出到清零端。

1. 设计结果（电路）

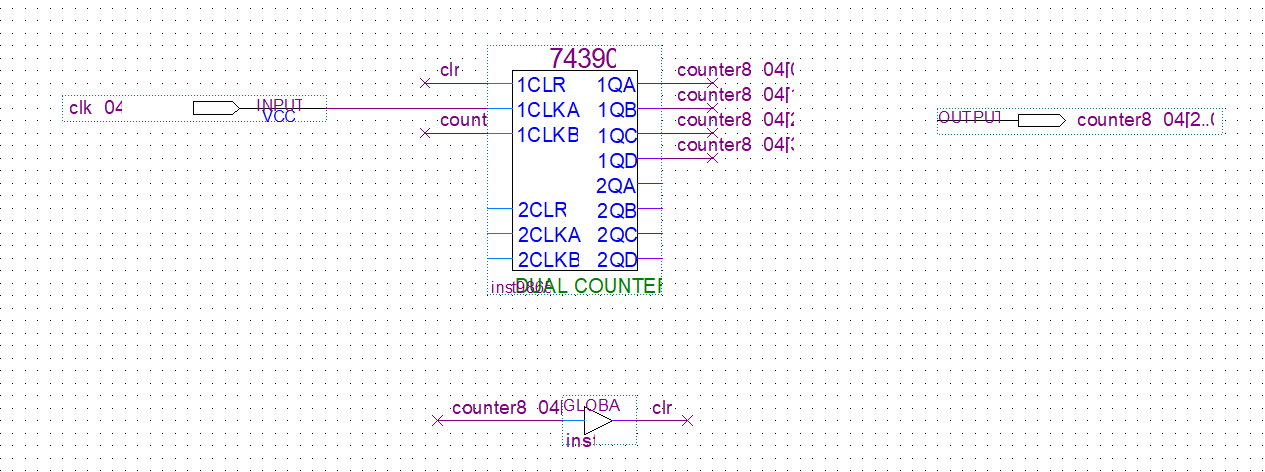


图13 扫描模块cnt8电路图

1. 仿真测试

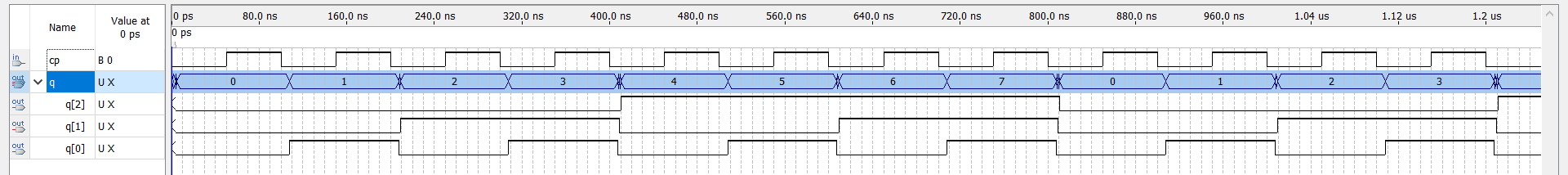


图14 扫描模块cnt8仿真图

* 1. 位选模块dig\_select

1. 模块功能

将输入的数据译码后输出。即配合扫描模块，依次选中8个数码管。

1. 设计思路

位选模块为3线-8线74138译码器，该芯片输出只有一个低电平。同时我们使用的是共阴极数码管，也就是8个数码管在同一时刻只有一个数码管被点亮。位选模块配合扫描模块。将扫描模块的输出作为位选模块的输入，实现8个数码管轮流被点亮。同时我们可以将扫描频率增加至1khz，这样8个数码管看起来就像是同时被点亮了。

1. 设计结果（电路）

图示, 示意图

描述已自动生成

图15 位选模块dig\_select电路图

1. 仿真测试

日历

描述已自动生成

图16 位选模块dig\_select仿真图

* 1. 数据选择模块code\_select

1. 模块功能

在地址端的作用下，将特定的数据输出。以达到当选中特定数码管时，使该数码管显示特定的数据的功能。

1. 设计思路

数据选择模块由4个74151芯片组成 。74151芯片的输入端包括3个地址选择端和8个数据输入端。其功能是在地址选择端的控制下将8个数据中的特定的1个数据输出。例如当地址段输入为6，那么输出为D6。4个74151芯片的地址选择端全部接扫描模块的输出。

计时模块中小时、分钟、秒都分为个位和十位，个位和时位都时4位的8421BCD码。数据选择模块的功能是到当选中特定数码管时，使该数码管显示特定的数据。例如，当扫描模块的输出为1，即位选模块里秒钟的十位被选中了。此时段选模块的数据选择端也为1 。那么4片74151的D1应该接分钟的4位8421码。依次类推。

1. 设计结果

图片包含 图示

描述已自动生成

图17 数据选择模块code\_select电路图

1. 仿真测试

图形用户界面, 应用程序

描述已自动生成

图18 数据选择模块code\_select仿真图

* 1. 译码模块decoder

1. 模块功能

将4位8421BCD码译码为7为段码，使得数字可以正确的显示在数码管上。

1. 设计思路

使用7448共阴极数码管译码器，可以将4位8421BCD译码为7位段码。

1. 设计结果

图形用户界面, 图示, 示意图

描述已自动生成

图19 译码模块decoder电路图

1. 仿真测试

日历

描述已自动生成

图20 译码模块decoder仿真图

* 1. 动态显示模块电路图

日历

描述已自动生成

图21 动态显示模块电路图

1. 其他扩展功能
   1. 整点报时
2. 模块功能

当整点时(即分钟和秒钟同时为零)，会触发蜂鸣器1秒钟；

1. 设计思路

将计时模块的分钟的8位8421BCD码和秒钟的8位8421BCD码或非后与1khz频率相与输出给蜂鸣器。这样设计的原因是只有这16位输入均为0时，蜂鸣器才会被触发。16位输入均为0也就是整点。

1. 设计结果

电脑萤幕画面

低可信度描述已自动生成

图22 整点报时模块电路图

1. 仿真测试

表格, Excel

描述已自动生成

图23 整点报时模块仿真图（一）表格

描述已自动生成

图24 整点报时模块仿真图（二）

* 1. 星期的显示

1. 模块功能

FPGA板上LED0-LED6分别表示周一到周日（系统第一次启动默认为周一），假设当前是周一，那么LED0会亮，LED1-LED6处于灭的状态。

1. 设计思路

将计数模块的时位的进位输出作为该模块的时钟信号，这意味着当时位进位时，该模块的计数器加一。再将计数器的输出结果作为74138的输入。又由于LED灯是高电平发亮，所以74138译码器的输出需要取反后输出。

1. 设计结果

图示, 示意图

描述已自动生成

图25 星期的显示电路图

1. 仿真测试

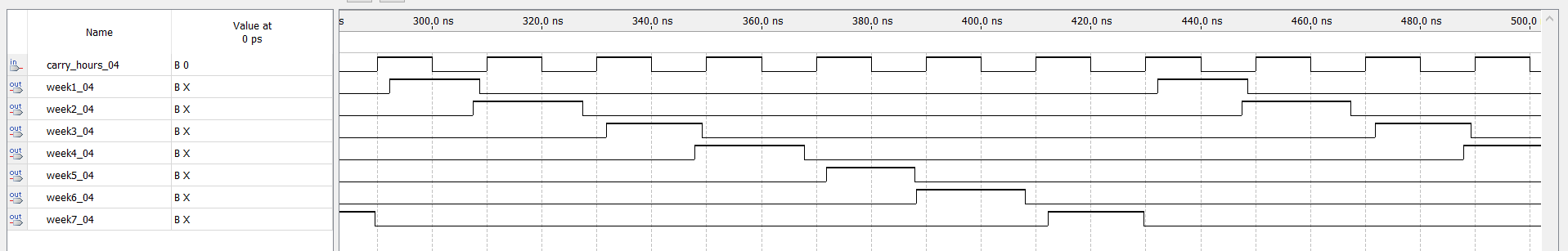


图26 星期的显示仿真图

* 1. 上下午

1. 模块功能

FPGA板上LED7表示当前时刻是否为上午。LED7亮表示当前时刻为上午，否则为下午。

1. 设计思路

满足下午的条件：

1. 小时的十位8421码中第2位为1。这时的时间为20:00到23:00；
2. 小时的十位8421码中第1位为1，个位的第3位或者第4位只要有一位为1。这时的时间为14:00到19:00；
3. 小时的十位8421码中第1位为1，个位的第1位和第2位同时为1。这时的时间为13:00；

将这三个条件或起来输出给LED7。

1. 设计结果

图片包含 散点图

描述已自动生成

图27 上下午显示的电路图

1. 仿真测试

表格

中度可信度描述已自动生成

图28 上下午显示的电路图（一）

表格, Excel

中度可信度描述已自动生成

图29 上下午显示的电路图（二）

* 1. 6、9补断

1. 模块功能

使数字6、9的显示会更加美观。将6的最上面的横杠变亮，将9最下面的横杠变亮。

1. 设计思路

当译码的输入为6（0110）时使seg[0]为1；

当译码的输入为9（1001）时使seg[3]为1；

将code\_select\_04[0]取反和code\_select\_04[1]、code\_select\_04[2]相与再与7448的第一个输出端相或接到seg\_04[0]。

将code\_select\_04[0]、code\_select\_04[3]相与的结果和code\_select\_04[1]、code\_select\_04[2]相与再取反的结果相或所得到的结果和7448第4个输出相或输出到seg\_04[3]。

1. 设计结果

图示

描述已自动生成

图30 6、9补断电路图

1. 仿真测试

日历

描述已自动生成

图31 6、9补断仿真图

* 1. 时钟暂停

1. 模块功能

可以通过key4按键使时钟停止不再计时。

1. 设计思路

将key4与时钟信号相或后输给时钟模块。当key4为1时，相或的结果永远为1，那么时钟模块就停止计时。

1. 设计结果

日历

描述已自动生成

图32 时钟暂停模块电路图

1. 仿真测试

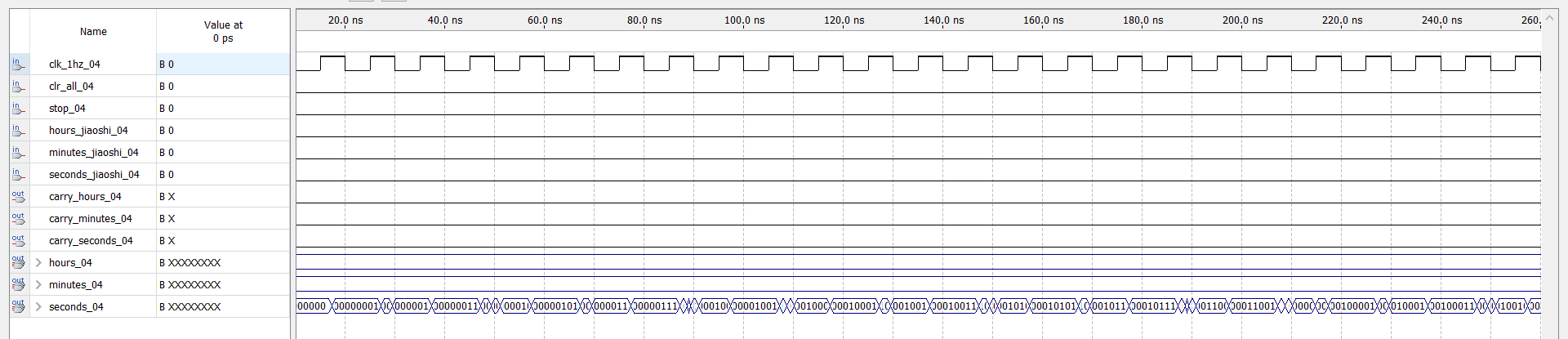


图33 时钟暂停模块仿真图（一）

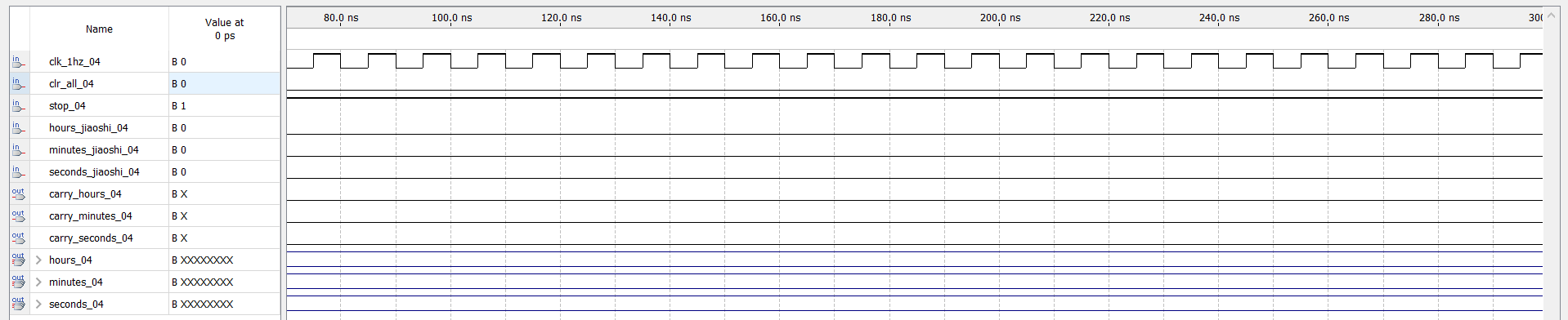


图34 时钟暂停模块仿真图（二）

* 1. 校时

1. 模块功能

可以通过按键key1、key2、key3分别对时钟的秒、分、时进行校时。

1. 设计思路

将按键和时钟信号相或的结果作为计时模块的时钟输入。当不按键时，时钟模块的时钟输入为原始的时钟信号。每按一次键，就会产生一次脉冲信号。

1. 设计结果

图片包含 图示

描述已自动生成

图35 校时模块电路图

1. 仿真测试

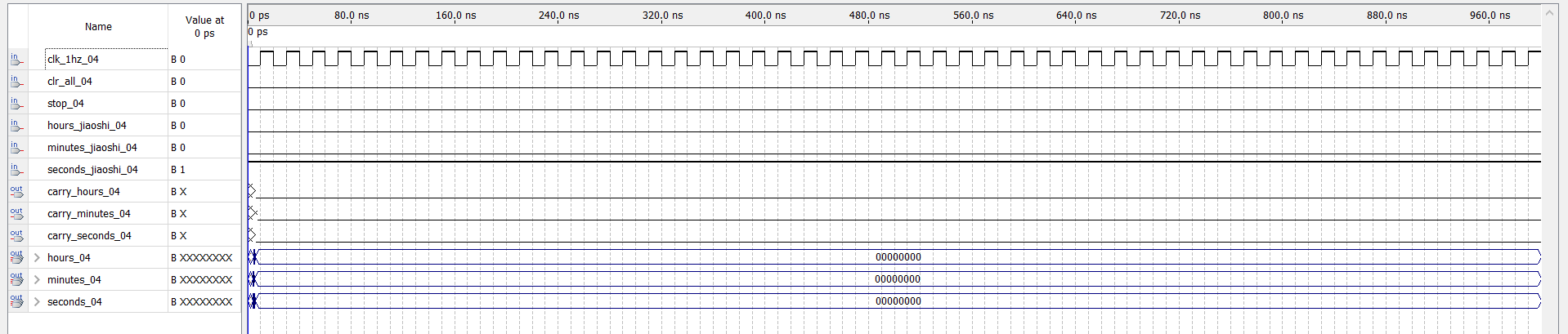


图36 秒钟校时仿真图

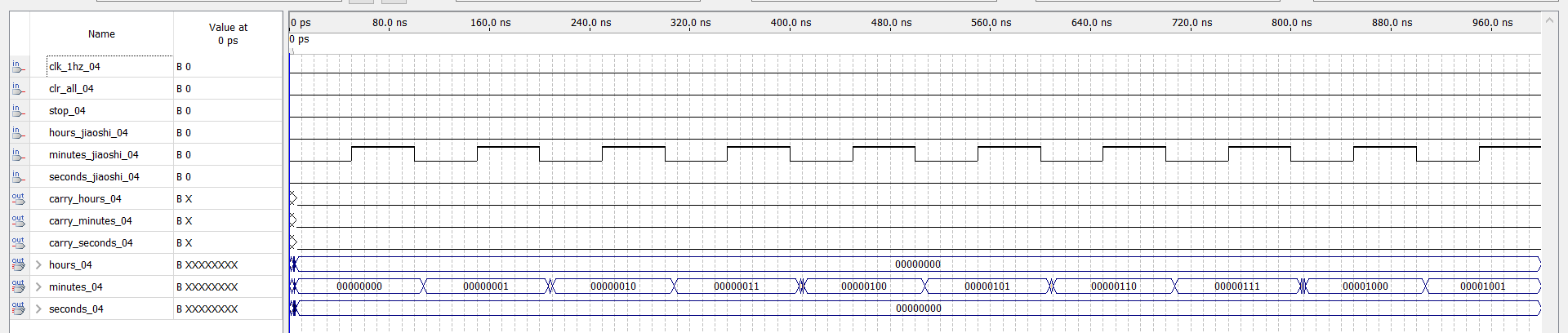


图37 分钟校时仿真图

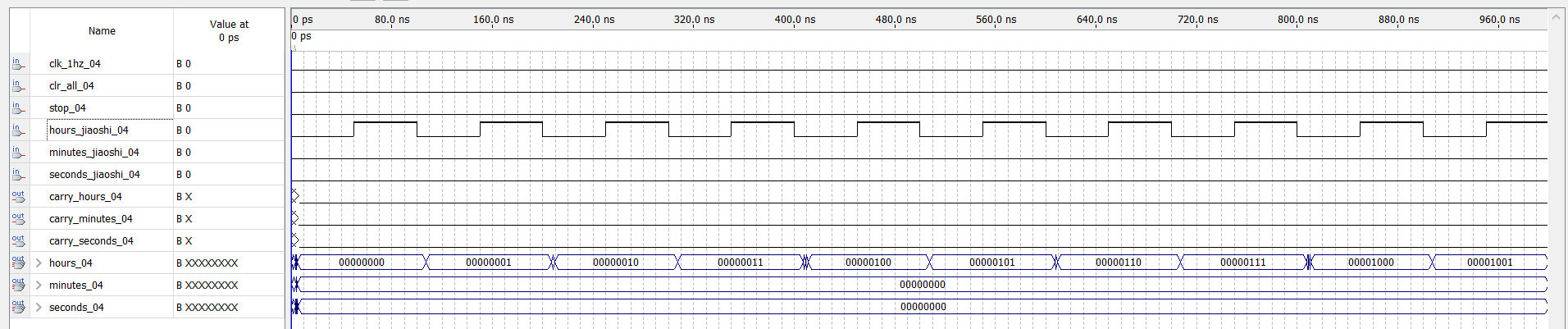


图38 小时校时仿真图

* 1. 置零

1. 模块功能

可以通过按键key5对时钟进行置零操作。

1. 设计思路

将按键key5与计时模块原始的置零输入相或的结果输入给计时模块的置零端。

当不按键时，置零端的输入为原始的置零输入。

1. 设计结果

图示

描述已自动生成

图39 置零模块电路图

1. 仿真测试

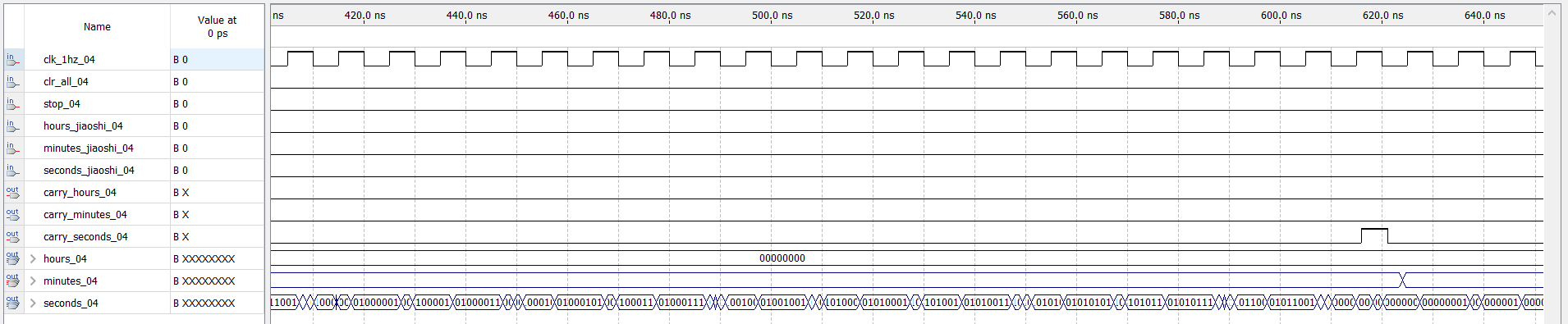


图40 置零模块仿真图（一）

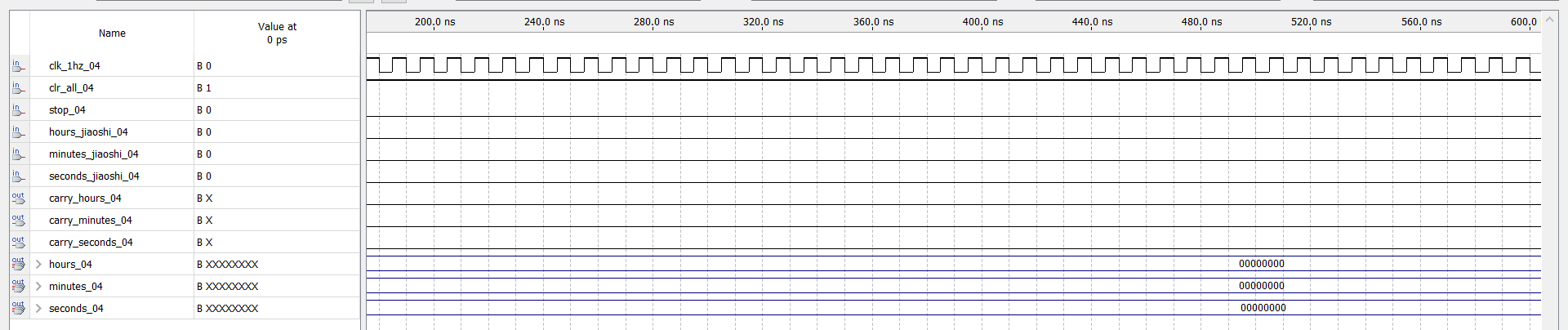


图41 置零模块仿真图（二）

* 1. 加杠

1. 模块功能

时钟的显示格式为时-分-秒，例如：15-12-20。

1. 设计思路

我们很容易可以发现，杠的位置为第3个数码管和第6个数码管。我们将74151的D3和D6都设置为1。那么当扫描到这个第3个或者第6个位置时，段选模块会输出1111。

在译码模块的输出端进行一次判断，如果段选信号是1111的话，seg\_04[6]会输出1 。因为1111原本的译码结果为数码管全灭。综上就可以实现加杠的功能了。

1. 设计结果

图示, 示意图

描述已自动生成

图42 译码模块电路图

图示

描述已自动生成

图43 加杠模块电路图

1. 仿真测试

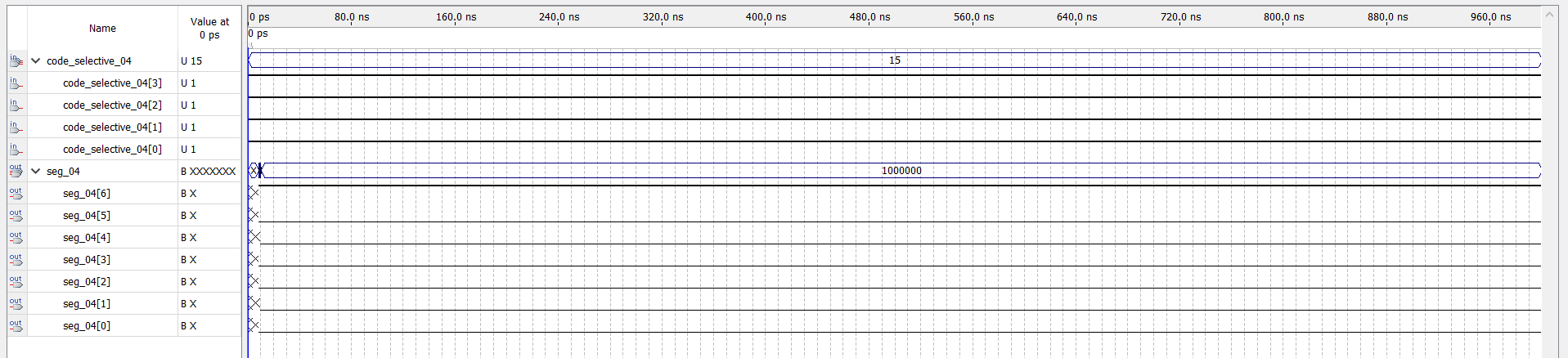


图44 加杠模块仿真图

* 1. 闹钟

1. 模块功能

用户可以通过拨动S4开关进入退出闹钟模式，通过S2、S3对闹钟的分和小时进行设置。当时钟到达设置的闹钟时间时，蜂鸣器会被触发一分钟；

1. 设计思路

转换模块

将开关S4与段选码A相与的结果与S4的反与段选码B相与的结果相或后输出。这样可以达到这样的效果：S4为高电平时，输出段选码A；S4为低电平时，输出段选码B。再将输出接入译码模块。这样就达到了切换时间显示的功能。

闹钟模块

拨码开关S2、S3作为闹钟模块的分、时的时钟信号，每拨动一次，闹钟的分和时就会加1。分和时依然为模60和模24。

比较模块

将闹钟的时位的8位8421码和分位的8位8421码与当前时间比较。7485芯片的性质是，当A0-A3与B0-B3位位相等时，AEBO才会输出1。将4片7485芯片的AEBO输出端相与后的结果与1khz相与后接到蜂鸣器上。也就是说只有当这16位数据全部相等时，蜂鸣器才会触发。当前时间的分钟或者时钟变化时，蜂鸣器触发失败。也就是说到达闹钟时间时，蜂鸣器的触发时间位1分钟。

1. 设计结果

图示

描述已自动生成

图45 选择模块电路图

图示, 示意图

描述已自动生成

图46 闹钟模块电路图

图示

描述已自动生成

图47 比较模块电路图

1. 仿真测试

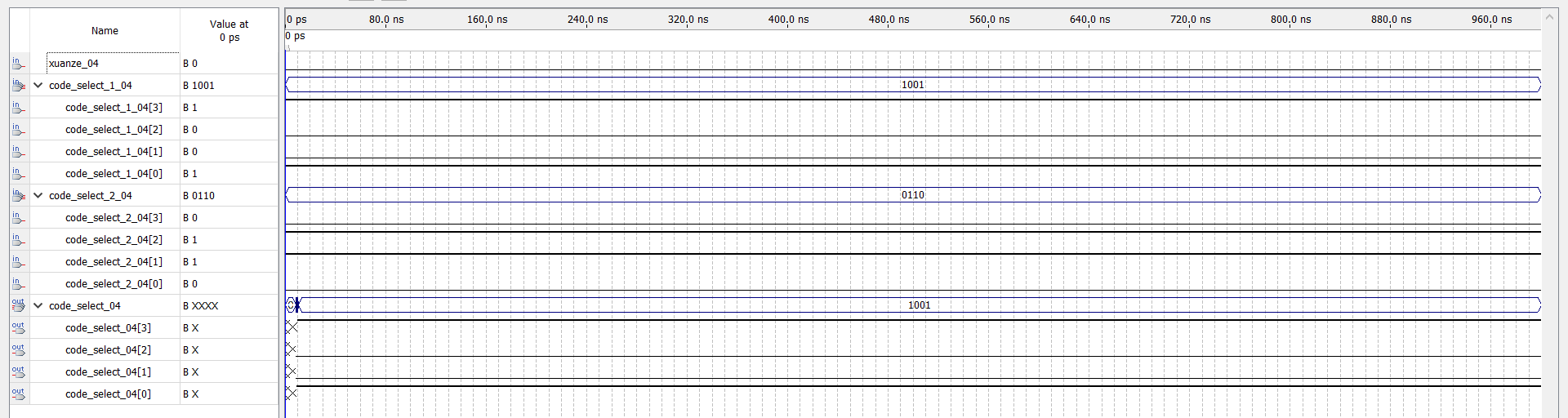


图48 选择模块仿真图（一）

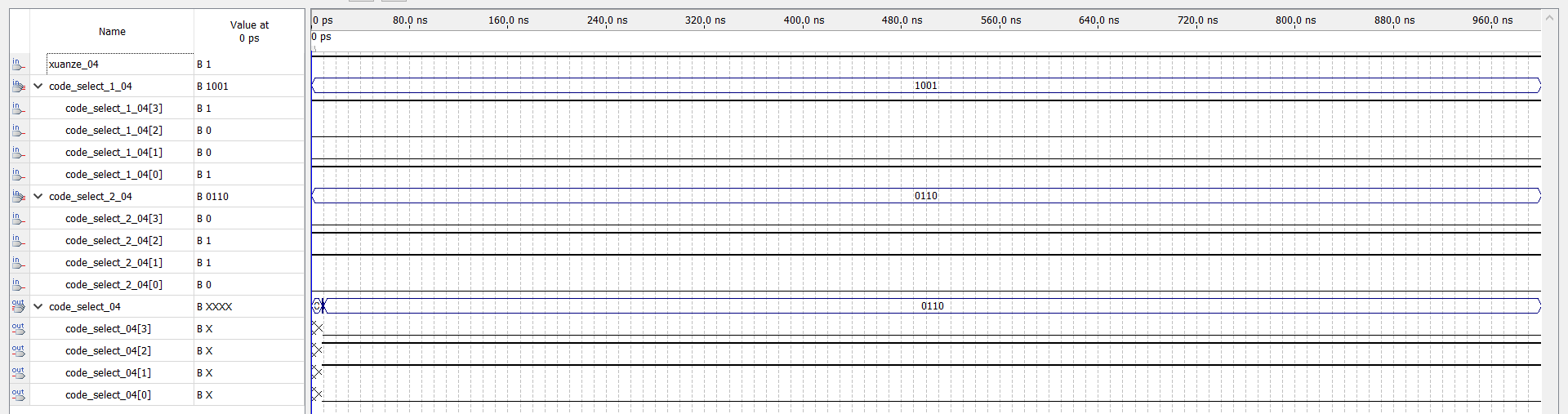


图49 选择模块仿真图（二）

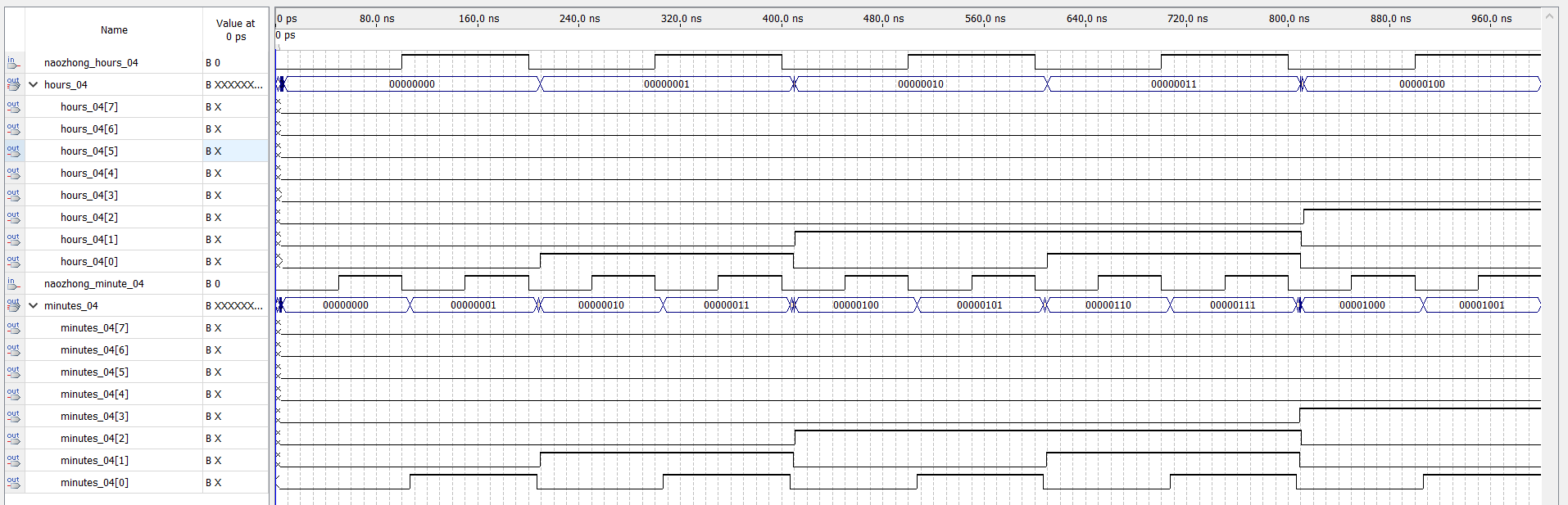


图50 闹钟模块仿真图

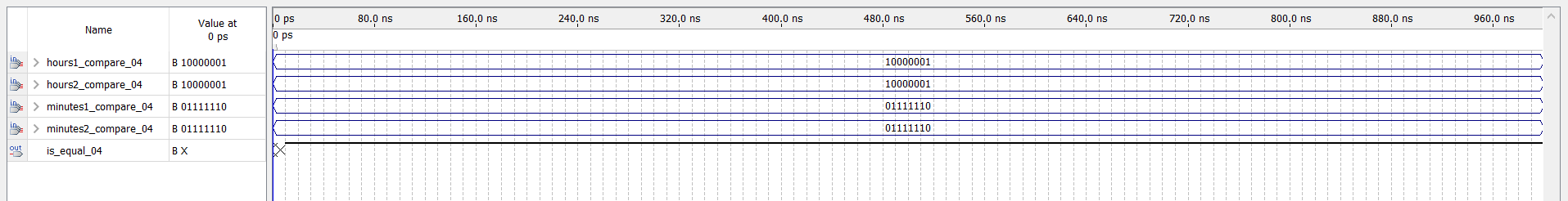


图51 比较模块仿真图（一）

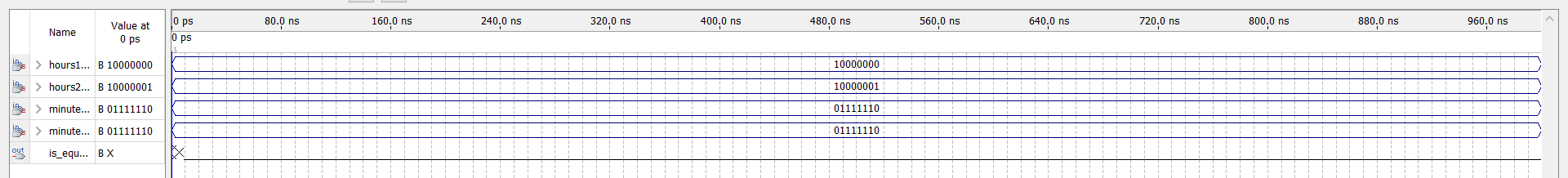


图52 比较模块仿真图（二）

1. 系统总体测试

详见“附录一：评分细则及测试原始数据记录”。

1. 系统设计实现过程中遇到的主要问题、解决思路和解决方案

问题1：如何判断当前时间是上午还是下午？

问题描述：我一开始想到的时使用2片7485芯片将当前时间的时位和12进行比较，根据7485的输出来判断到底时上午还是下午。但是同学想到了一个更好的方法。

解决方案：上面的解决方案实际上可行。但是同学想出了一种更简单的方法，判断13-23点时位输出的数据有什么特点。通过一些与、或、非门来完成这一功能(上面有这种方法的解释)。这种方法看起来很巧妙不容易想到。但是仔细思考一下，这就是我们最开始学的画真值表-写出函数表达式-卡诺图化简法啊。这个方法我以为我掌握了，但我只是理解了它的用法，但是不能灵活运用。

解决思路：

满足下午的条件：

1. 小时的十位8421码中第2位为1。这时的时间为20:00到23:00；
2. 小时的十位8421码中第1位为1，个位的第3位或者第4位只要有一位为1。这时的时间为14:00到19:00；
3. 小时的十位8421码中第1位为1，个位的第1位和第2位同时为1。这时的时间为13:00；

将这三个条件或起来输出给LED7。（具体可以见附加功能—上下午功能）

问题2：如何实现闹钟功能？

问题描述：一开始做这个功能的时候感觉到处是难处，怎么将用户输入的时间存进去？用户输入的时候数码管怎么显示呢？等等等等

解决思路：我一开始是想去网上搜索有没有比较高级的库可以调用，使用这些库来完成这个功能可能会更加简单。或者使用编程的方式来实现，因为我觉得使用C语言实现这个功能就很简单。解决方案：跟同学讨论后决定：

转换功能

将开关S4与段选码A相与的结果与S4的反与段选码B相与的结果相或后输出。这样可以达到这样的效果：S4为高电平时，输出段选码A；S4为低电平时，输出段选码B。再将输出接入译码模块。这样就达到了切换时间显示的功能。

闹钟功能

拨码开关S2、S3作为闹钟模块的分、时的时钟信号，每拨动一次，闹钟的分和时就会加1。分和时依然为模60和模24。

比较功能

将闹钟的时位的8位8421码和分位的8位8421码与当前时间比较。7485芯片的性质是，当A0-A3与B0-B3位位相等时，AEBO才会输出1。将4片7485芯片的AEBO输出端相与后的结果与1khz相与后接到蜂鸣器上。也就是说只有当这16位数据全部相等时，蜂鸣器才会触发。当前时间的分钟或者时钟变化时，蜂鸣器触发失败。也就是说到达闹钟时间时，蜂鸣器的触发时间位1分钟。（具体可以见附加功能—闹钟功能）

1. 心得体会

首先，通过这个大作业，我的数字电路的动手设计能力、对各个芯片模块的理解能力、对一些常见功能的设计能力都有了很大的提高。我感觉我做完这个大作业并完成这个报告后，对于一些简单的功能可以很轻松的设计出来了。而且我对芯片的使用更加熟练了。之前如果问我74151芯片、74138芯片的功能，我可能还分不太开。但是现在我认为我对这几款常用的芯片都非常熟悉了。除此之外，我还学到了一些课本上学不到的知识，比如说软件报错了到底是出了哪里问题？把哪些组件封装起来会使得整个项目更加简洁可读性更高？

除此之外，我还学会了如何和同学讨论，一起解决问题。我发现和同学一起学习讨论真的可以学习到很多知识，成长的很快。

最后，我认识到做事情态度的很重要。有的同学非常认真对待这次大作业，图画的非常公整，非常美观。我觉得我应该向他们学习。