|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **学 期** | 2022-2023学年第1学期 | | | | **实验日期** | | 2023/03/01 | | |
| **学 院** | 信息学部 | | | | **专 业** | | 计算机科学与技术（实验班） | | |
| **班 级** | 210710 | | **学 号** | | 21071003 | | **姓 名** | | 高立扬 |
| **组 号** | 43 | | **学 号** | | 21071004 | | **姓 名** | | 石昊阳 |
| **评 阅 内 容** | | | | | | | | | |
| **总体设计** | | **详细设计** | | **下 载** | | **总 结** | | **成 绩** | |
|  | |  | |  | |  | |  | |
|  | | | | | | | | | |
| **题 目：** | 实验7：实用电路设计（交通灯控制器） | | | | | | | | |

1. **功能描述**

交通灯控制器实验电路的预期功能：

1.十字路口，东西方向、南北方向红黄绿交通灯各三盏，绿灯通行，红灯停止，黄灯过渡准备。红绿灯变化规律为：东西绿灯，南北红灯→东西黄灯，南北红灯→东西红灯，南北绿灯→东西红灯，南北黄灯→东西绿灯，南北红灯。

2.假设东西方向、南北方向两交通要道的通行时间控制基本相等。两组数码管作为东西、南北方向的倒计时显示，时间可以预置，如时间为红灯59秒、绿灯56s，黄灯3秒。

3.具有复位功能，计数器恢复初始状态。

4.加入人互干预控制，使红绿灯在人工干预下可以停止计数并保持原来状态，东西，南北均为红灯状态，待特殊情况结束后能继续计数。

5.加入左转弯信号灯指示。

1. 自定义其它功能。

我们组的扩展功能：红绿灯倒计时快结束的时候，蜂鸣器开始响起间断性的蜂鸣，红灯、黄灯蜂鸣频率不同。红灯10s开始以1hz频率蜂鸣，黄灯4s开始以2hz频率蜂鸣。

1. **总体设计**

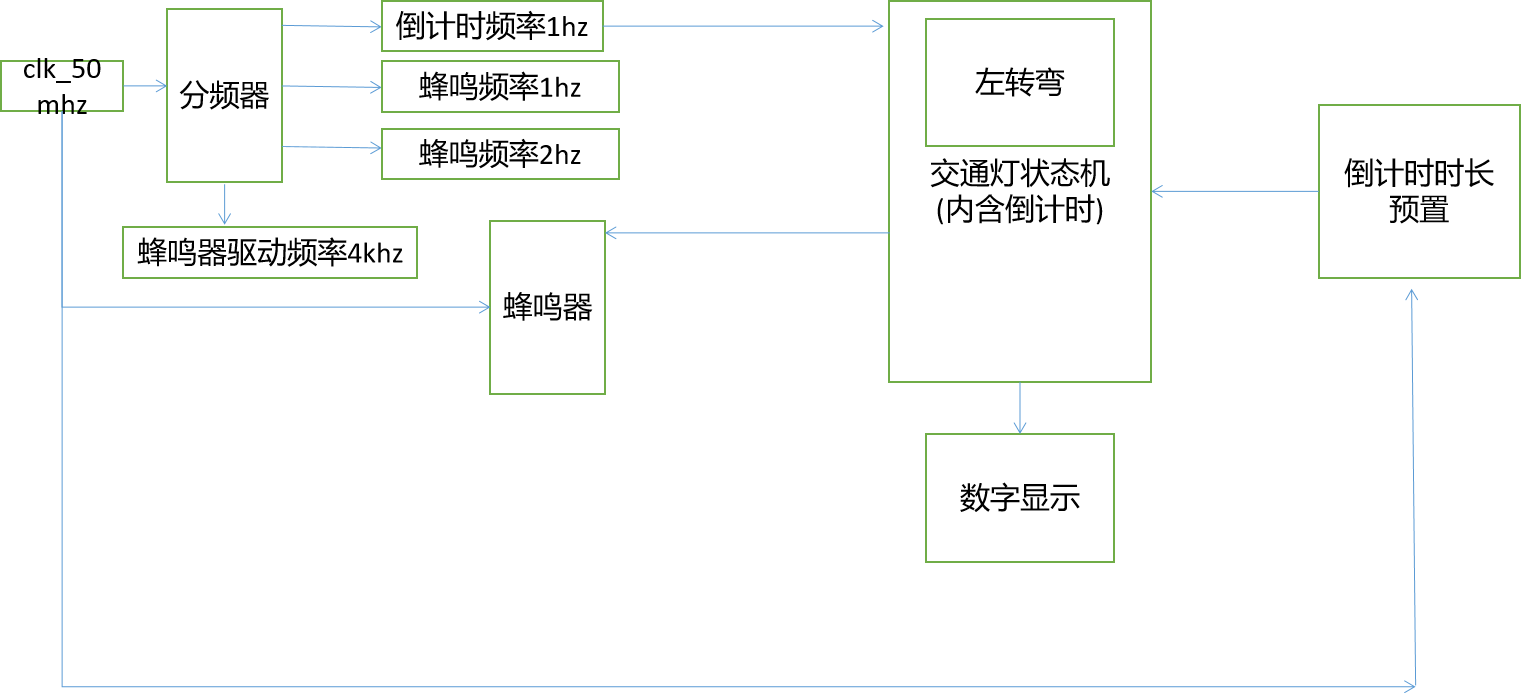


图1.总体设计

1. **详细设计**
2. 交通灯状态机模块设计

交通灯内置了时间倒计时和左转弯状态，具体设计如下：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **交通灯状态** | | | | | | | |
| **状态名** | 东西绿灯  南北红灯 | 东西黄灯  南北红灯 | 东西红灯  南北绿灯 | 东西红灯  南北黄灯 | 都是红灯 | 左转（1） | 左转（2） |
| **状态机编号** | **001** | **010** | **100** | **101** | **111** | **011** | **110** |

如图3.每当红绿灯倒计时结束的时候，状态机就向下一状态推进，否则一直保持当前状态；需要1个按钮来负责复位功能，一个开关负责开启状态机，一个开关负责人工干预功能。

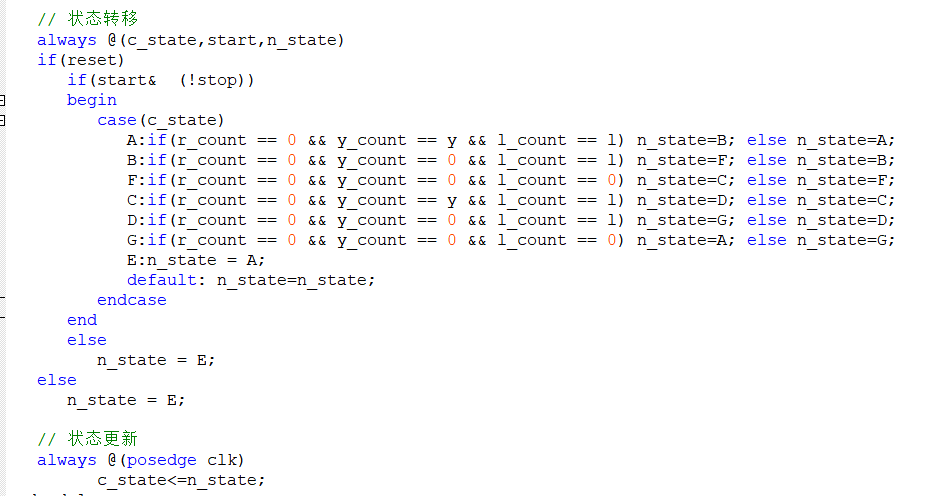
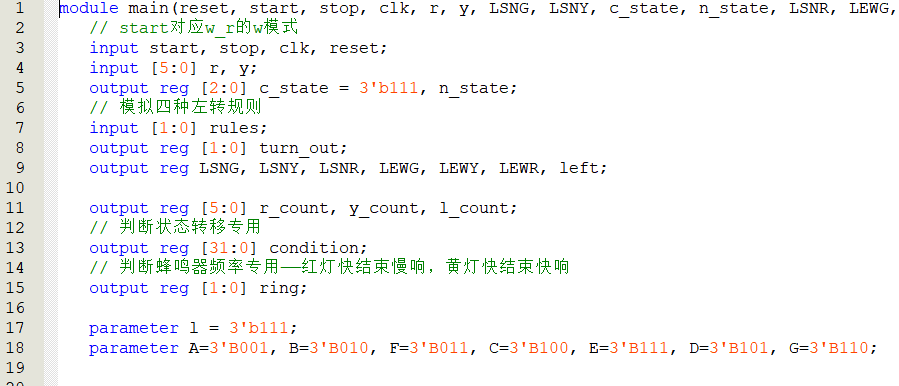


图2.本模块所有的变量 图3.状态机

如图4.倒计时部分负责红绿灯的倒计时，一共有三个倒计时寄存器，分别负责红灯倒计时，黄灯倒计时和左转弯倒计时，当stop开关开启或者reset按钮按下，倒计时会锁定或重置。同时，通过倒计时的时间判断，该部分代码还会输出一个叫ring的一位二进制数字，来控制蜂鸣器的鸣叫。

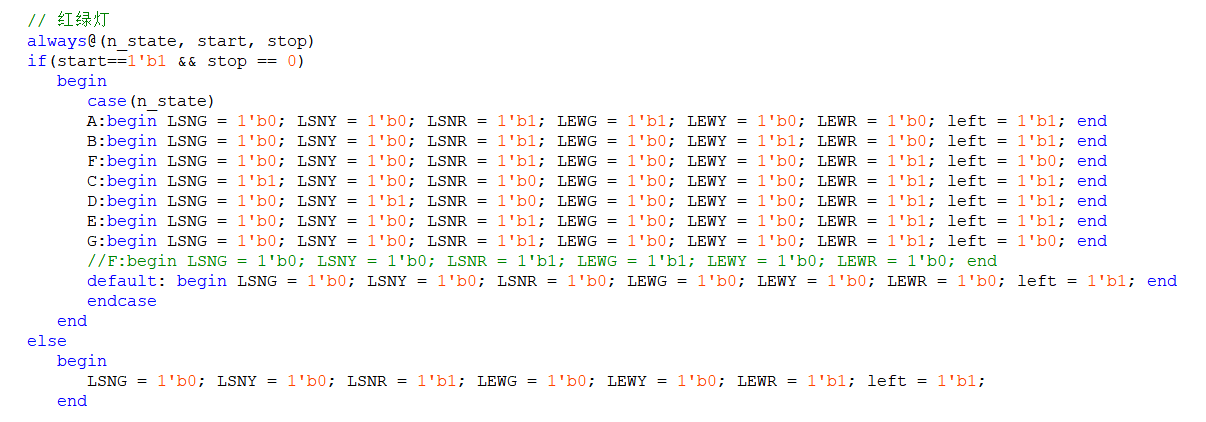
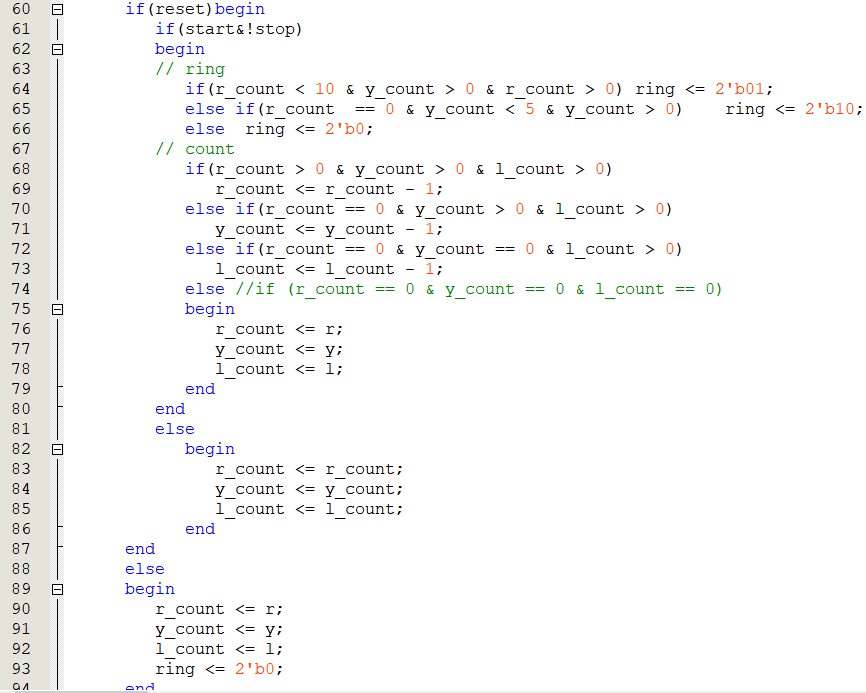


图4.倒计时模块 图5.红绿灯根据状态而亮起

如图5.红绿灯如何亮起，取决于状态机的状态如何，为了消除c\_state的滞后导致红绿灯转换滞后，这里选择通过n\_state来判断红绿灯的转换。

1. 红绿灯时长预置模块设计

为了方便实验的演示，预置时长最多为20秒，左转弯倒计时固定位7秒，红黄绿灯的预置简化为红黄灯的预置（因为预置东西红灯相当于预置南北绿灯），用到1个按钮负责时长+1，1个开关负责切换红黄两个灯的倒计时选择预置哪个，预置模式需要1个开关来激活。

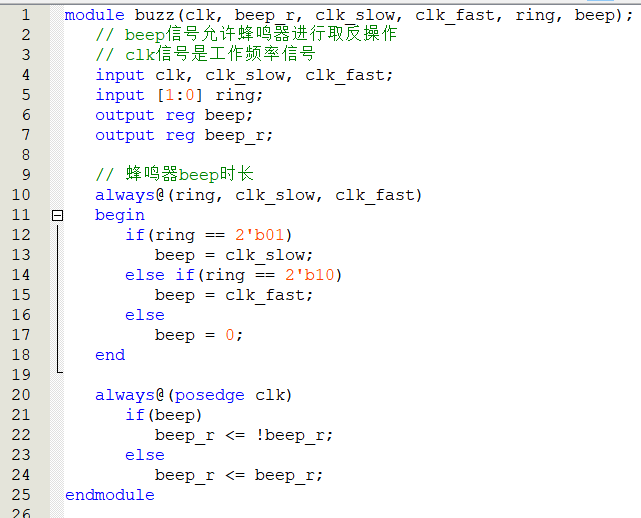
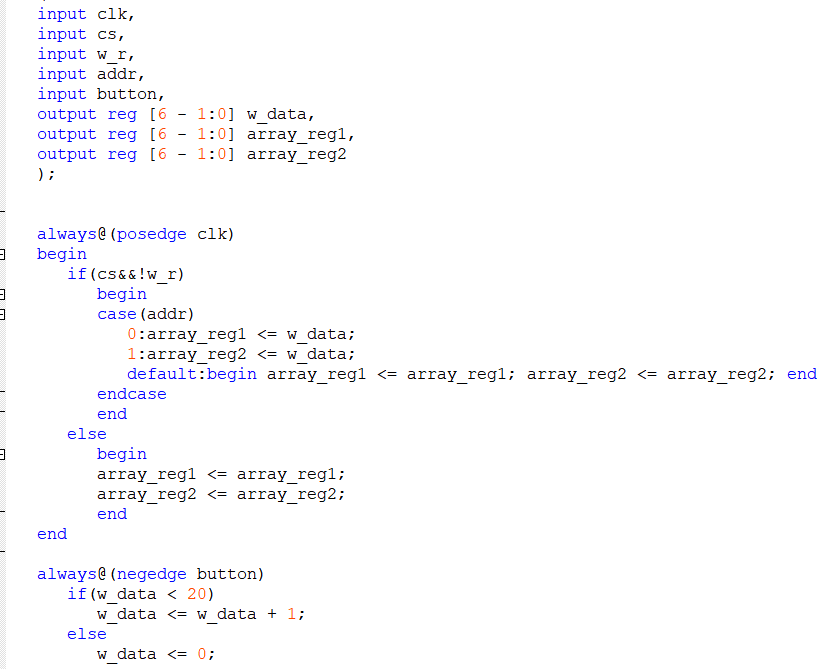


图6.时长预置模块 图7.蜂鸣器模块

1. 分频器模块设计

同实验三的分频器

1. 蜂鸣器模块设计

根据倒计时来决定响的频率，蜂鸣器驱动需要2k-4k hz之间的频率，发出声音的原理是：在2k-4k的频率下，蜂鸣器不断接收到此频率的01010...信号。落实在编程，可以通过一个1’b ring信号控制1’b beep信号接收1hz频率或2hz频率；随着50mhz的时钟，beep\_r信号根据beep不断取反并传输给蜂鸣器实现持久蜂鸣；其他情况beep\_r就会停止取反，蜂鸣器接收不到持续取反的beep，因此静音。beep\_r按照一定频率变化，可以实现持续蜂鸣变为间断的蜂鸣

1. 红绿灯倒计时数字显示模块设计

同七段数码管译码器。改进如下：

二位数显示：配合频扫，使传入数码管译码器的数据不断被赋值为十位和个位

在红绿灯扩展版上的数字显示：由于倒计时的逻辑是，当红黄灯和左转弯灯的倒计时都为0的时候，状态机才会转回A状态，三个倒计时才会重置。因此扩展板的倒计时数字显示可以通过判断倒计时是否为0来决定显示哪个倒计时，比如红灯倒计时大于零的时候，说明此时一定是红灯状态，如果红灯倒计时为0，但是黄灯倒计时不为0，则此时一定是黄灯状态。

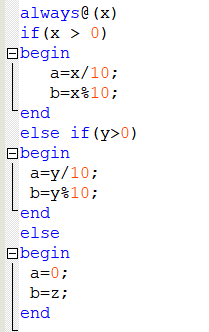
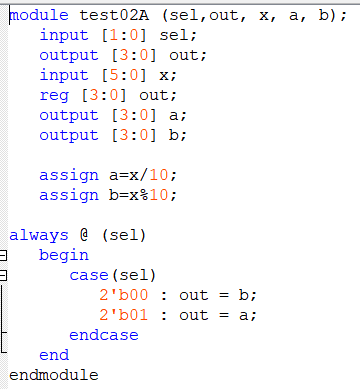


图8.二位数显示 图9.扩展版数字显示

1. 顶层设计

我们采用文字编程来编写顶层文件，下面展示除变量声明之外的代码。通过所有模块的先后和逻辑顺序，来编写顶层文件。

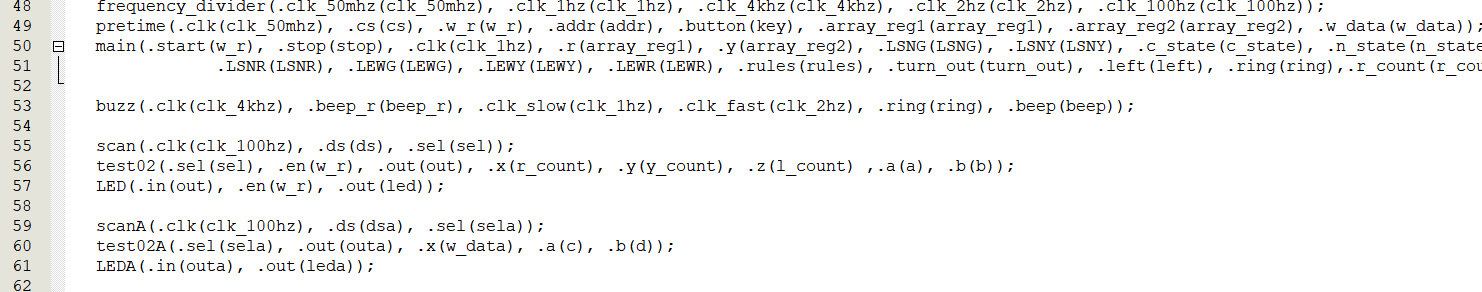


图10.顶层文件代码

1. **下载调试**
2. 引脚分配

表1.引脚分配

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 端口名称 | 输入端 | | | | | | | | |
| Clk\_50mHZ | cs | W\_r | addr | stop | Rules[0] | Rules[1] | key | reset |
| 引脚编号 | T1 | M20 | N18 | AA15 | V13 | F8 | E7 | AA22 | R18 |
| 平台端口 | T1 | SW2 | SW1 | SW3 | SW4 | SW8 | SW7 | F6 | F10 |

表1.续表

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 端口名称 | 输出端 | | | | | | | | |
| LEWG | LEWR | LEWY | LSNG | LSNR | LSNY | Beep\_r | Ds[1] | Ds[0] |
| 引脚编号 | E13 | A7 | F13 | F10 | B16 | D13 | A8 | B3 | B4 |
| 平台端口 | LEWG | LEWR | LEWY | LSNG | LSNR | LSNY | BUZZ | SEG\_H1 | SEG\_H2 |

表1.续表

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 端口名称 | 输出端 | | | | | | | | |
| Dsa[1] | Dsa[0] | Led[6] | Led[5] | Led[4] | Led[3] | Led[2] | Led[1] | Led[0] |
| 引脚编号 | V16 | AA17 | N19 | P20 | W14 | AB16 | Y13 | AB14 | R19 |
| 平台端口 | DS8 | DS7 | SEC\_A | SEC\_B | SEC\_C | SEC\_D | SEC\_E | SEC\_F | SEC\_G |

表1.续表

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 端口名称 | 输出端 | | | | | | | |
| left | Leda[6] | Leda[5] | Leda[4] | Leda[3] | Leda[2] | Leda[1] | Leda[0] |
| 引脚编号 | U12 | AA20 | W20 | R21 | P21 | N21 | N20 | M21 |
| 平台端口 | LED1 | LA | LB | LC | LD | LE | LF | LG |

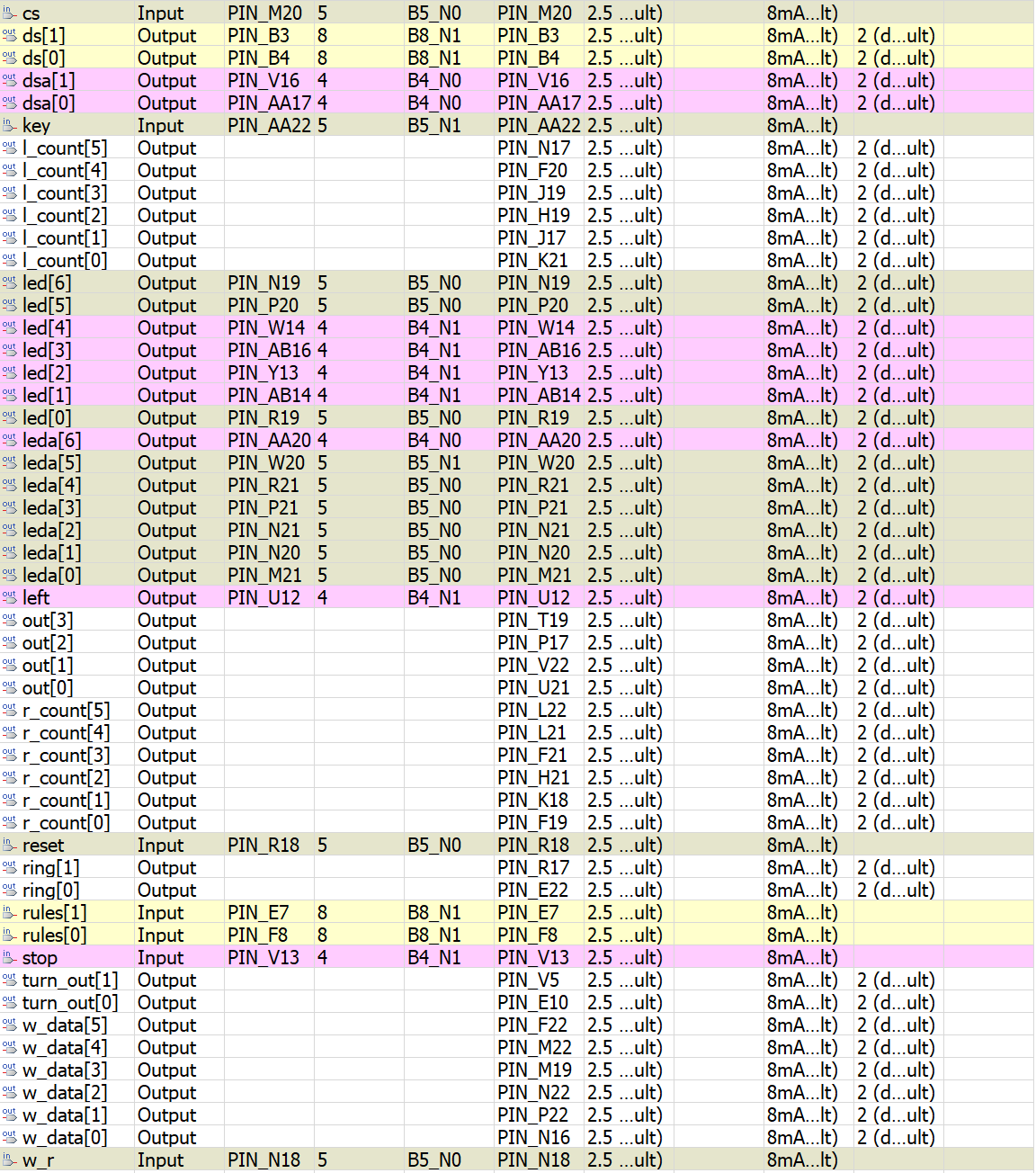
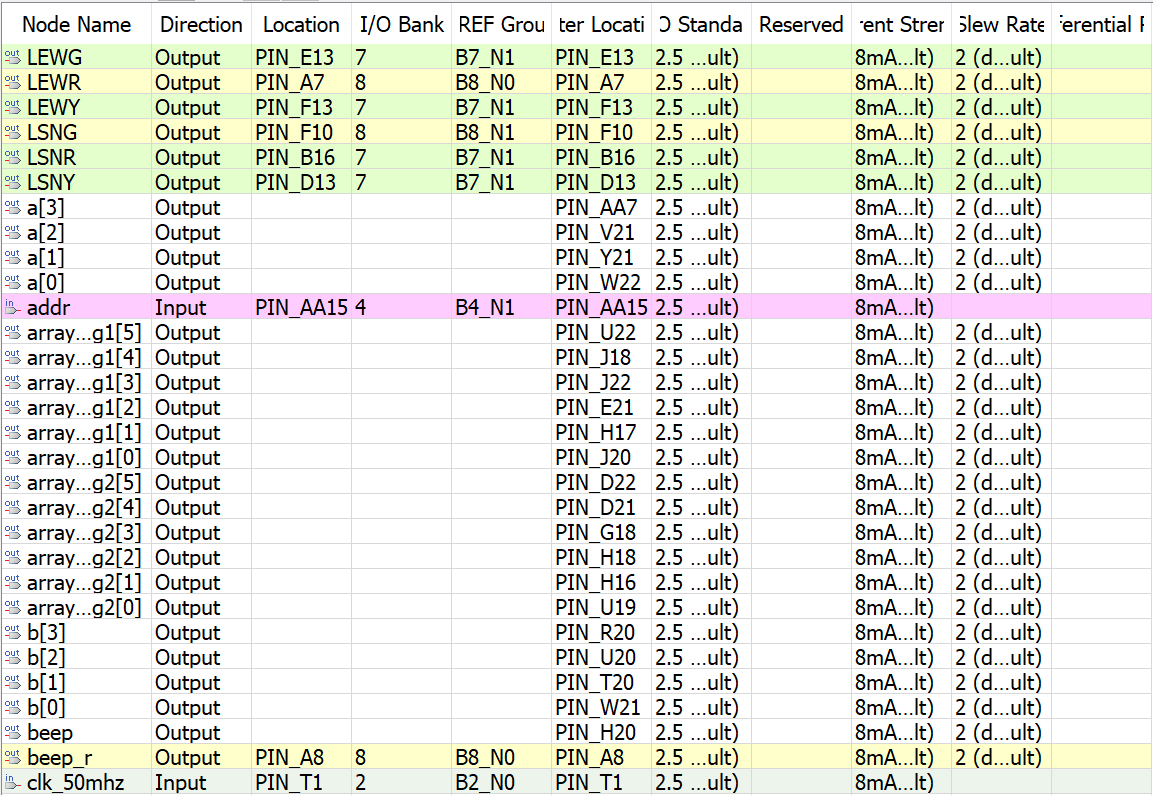


图11.图12.pin planner分配图（没分配引脚的变量是验证变量功能的时候残留下来的）

2、实验现象

实验开始后将CS开关开启，按下key按钮预制红灯，开启adrr开关摁下key预制黄灯。预制结束后将CS开关关闭，W\_r开关开启，红绿灯开始工作。红灯倒计时结束后进入黄灯状态，黄灯倒计时结束后进入左转状态，左转倒计时结束后进入红灯状态，并依次循环。在黄灯倒计时三秒内，及红灯倒计时十秒内，蜂鸣器依次以较快的频率与较慢的频率蜂鸣。开启stop开关，将会亮起所有红灯并暂停倒计时。关闭stop开关，红绿灯按原状态继续运行。按下reset摁钮，红绿灯会回到初始状态。

1. **问题总结**

本次实验是数字逻辑的最终大实验，我们在进行实验的过程中遇到了很多困难，但是都一一解决，最后在规定时间内顺利完成了任务。

①顶层文件编写的时候，我们出现了传错变量的情况，通过老师的指导，我们发现了错误赋值的变量并及时改正；

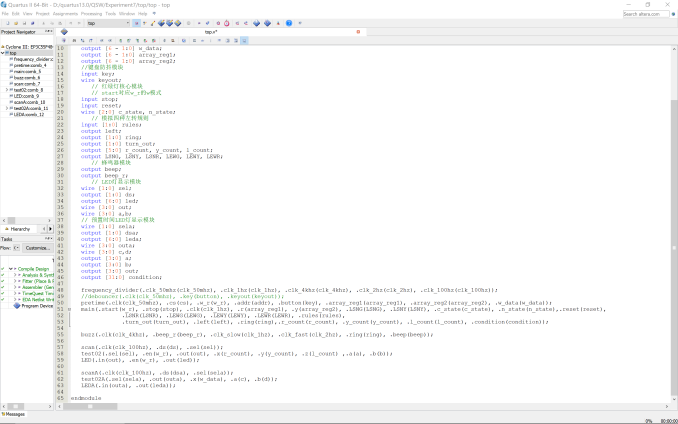
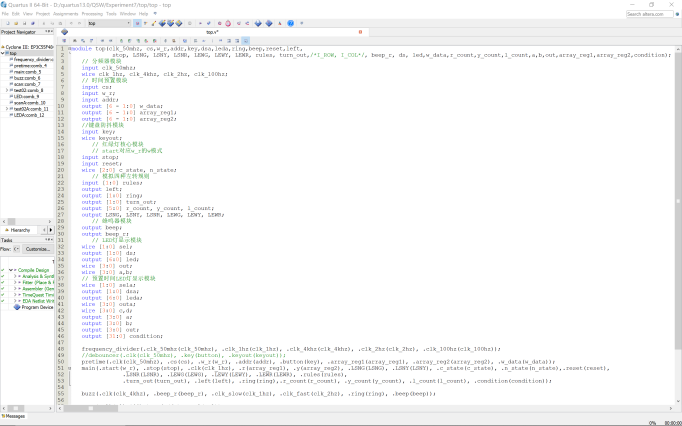
②由于实验涉及了四种不同的频率，因此跑波形图的时候会出现没波形的情况，如果通过波形图来验证实验的话，需要把所有的频率统一为50mhz。

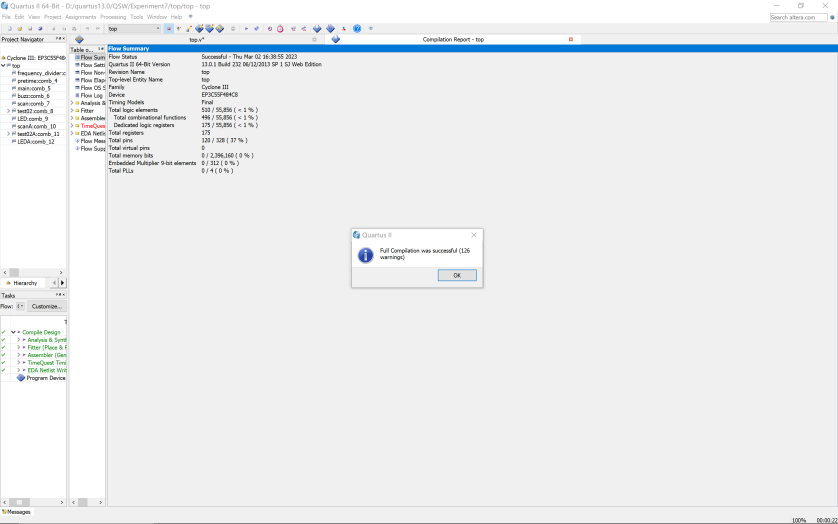
③预置时间的时候，我们的代码没有有效利用if语句，导致变量繁多复杂，无法赋值。通过不断演算，我们简化了代码，实现了预期的功能。

④状态机模块出错最多。一开始是状态转换逻辑出错，后来检查发现是倒计时模块有错，通过演算和仿真，我们去除了冗余的逻辑和变量，有效利用倒计时逻辑和if语句实现了预期功能。

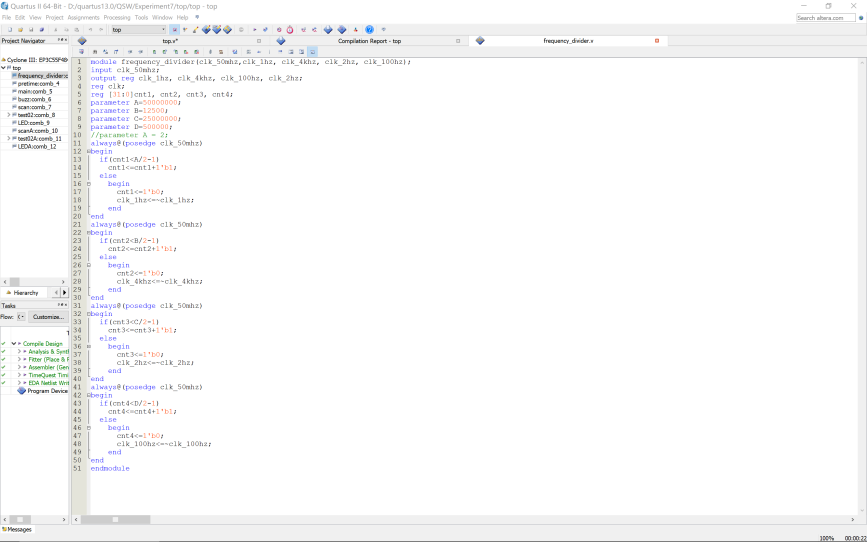
1. **任务分配**

|  |  |
| --- | --- |
| 模块 | 完成人 |
| 红绿灯状态机 | 高立扬 |
| 时间预置 | 高立扬 |
| 分频器 | 石昊阳 |
| 数字显示 | 石昊阳 |
| 蜂鸣器 | 石昊阳 |
| 顶层设计 | 高立扬 |

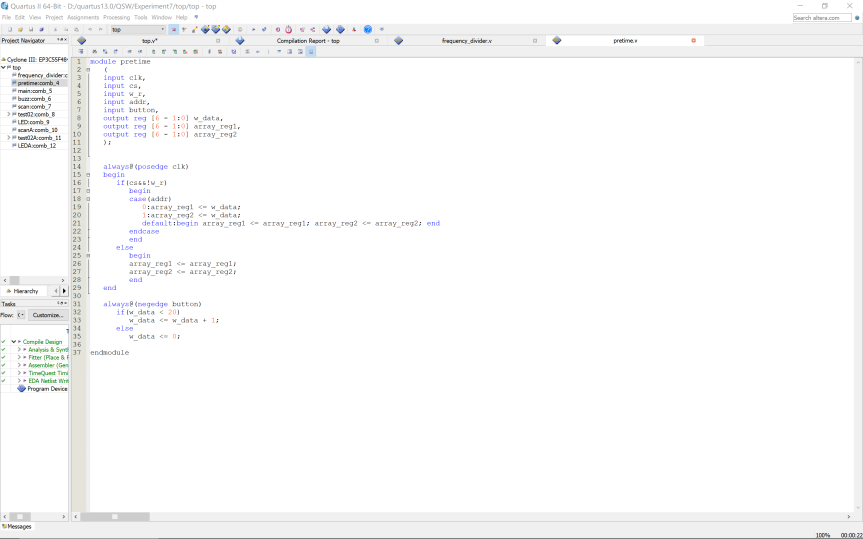




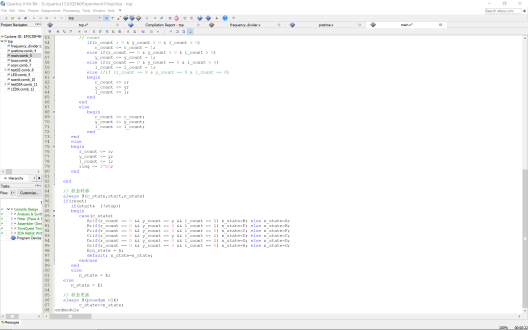
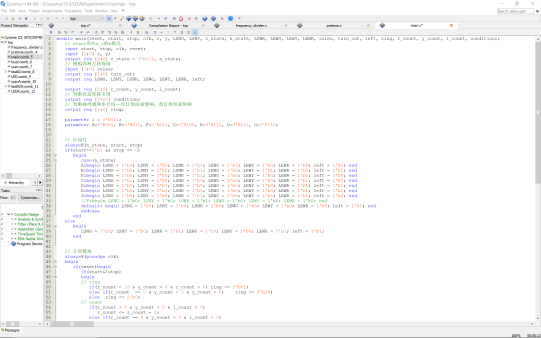
附图1：顶层电路及编译成功信息



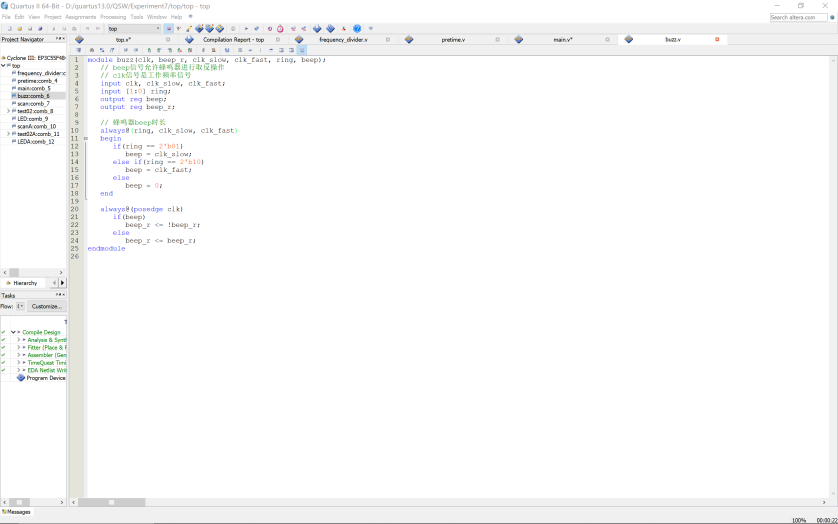
附图2：分频器模块代码



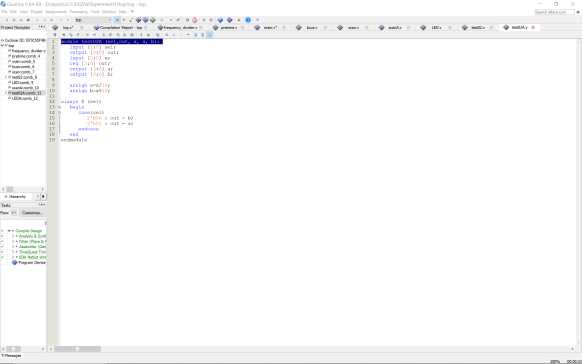
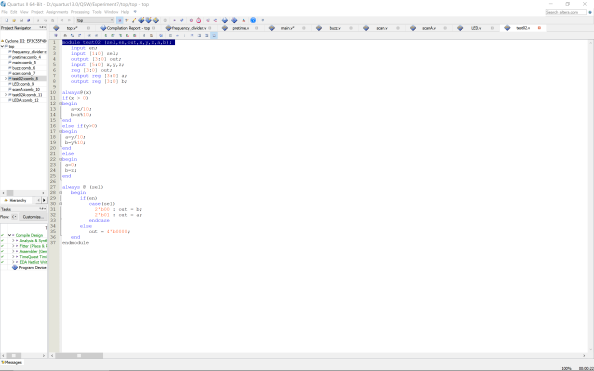
附图3：预置时间模块代码

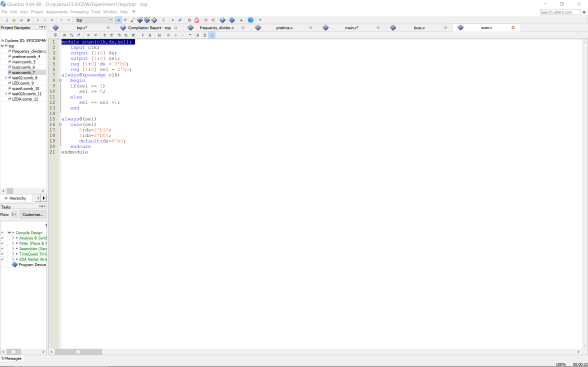
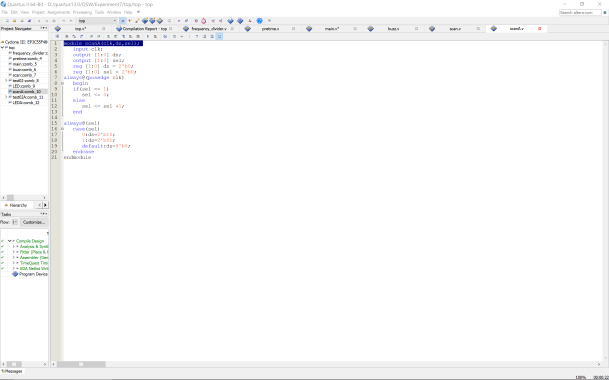


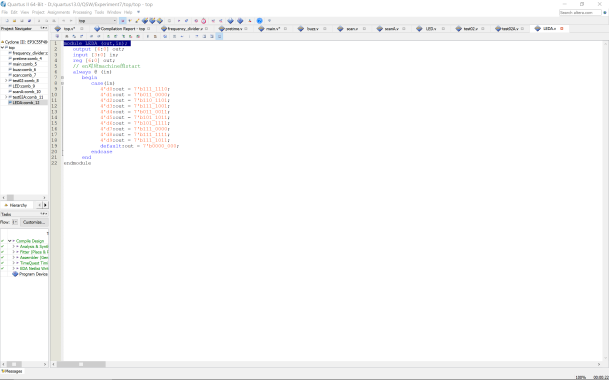
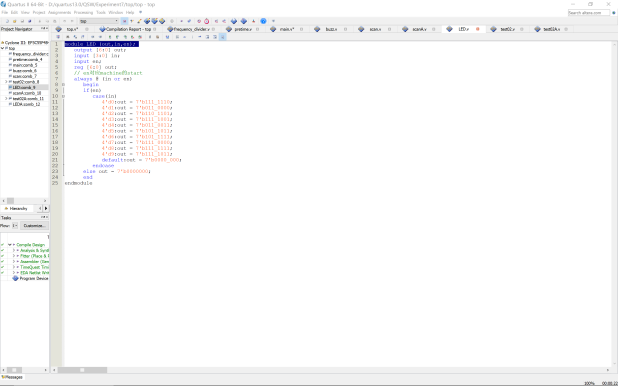
附图4：状态机模块代码



附图5：蜂鸣器模块代码







附图6：LED模块代码