

**数字电路与逻辑设计实验报告**

***2023***

|  |  |
| --- | --- |
| 专 业： | 计算机科学与技术 |
| 班 级： | CS2201 |
| 学 号： | U202215357 |
| 姓 名： | 王文涛 |
| 电 话： |  |
| 邮 件： |  |
| 完成日期： | 2023.12.14 |

**实验报告及电路设计评分细则**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 评 分 项 目 | 满分 | 得分 | 备注 | |  |
| 文档格式（段落、行间距、缩进、图表、  编号等） | 15 |  |  | | 实验报告总分 |
| 实验总体设计 | 10 |  |  | |
| 实验过程 | 50 |  |  | |
| 遇到的问题及处理 | 10 |  |  | |
| 设计方案存在的不足 | 5 |  |  | |
| 心得（含思政） | 5 |  |  | |
| 意见和建议 | 5 |  |  | |
| 电路（头歌） | 100 |  |  | |  |
| 教师签名 |  | | 日 期 |  | |

备注：实验过程将从电路的复杂度、是否考虑竞争和险象、电路的美观等方面进行评分。

实验课程总分=电路（头歌）\*0.4+实验报告\*0.6

目 录

1. [实验概述 1](#_bookmark0)
   1. [实验名称 1](#_bookmark1)
   2. [实验目的 1](#_bookmark2)
   3. [实验环境 1](#_bookmark3)
   4. [实验内容 1](#_bookmark4)
   5. [实验要求 3](#_bookmark5)
2. [实验总体设计 4](#_bookmark6)
   1. [实验总体设计思路 4](#_bookmark7)
   2. [实验总体设计框架 4](#_bookmark8)
3. [实验过程 7](#_bookmark9)
   1. [7 段数码管驱动电路设计 7](#_bookmark10)
   2. [无符号比较器（2 位、4 位、8 位） 10](#_bookmark11)
   3. [2 选1 选择器设计（2 位、8 位） 13](#_bookmark12)
   4. [十进制可逆计数器（包含状态机、输出函数及整体电路） 15](#_bookmark13)
   5. [两位十进制可逆计数器 19](#_bookmark14)
   6. [交通灯状态机 21](#_bookmark15)
   7. [交通灯输出函数设计 25](#_bookmark16)
   8. [交通灯控制系统 27](#_bookmark17)
4. [设计总结与心得 31](#_bookmark18)
   1. [实验总结 31](#_bookmark19)
      1. [遇到的问题及处理 31](#_bookmark20)
      2. [设计方案存在的不足 31](#_bookmark21)
   2. [实验心得 31](#_bookmark22)
   3. [意见与建议 32](#_bookmark23)

# 实验概述

## 实验名称

交通灯系统设计。

## 实验目的

本实训将提供一个完整的数字逻辑实验包，从真值表方式构建 7 段数码管驱动电路，到逻辑表达式方式构建比较器，多路选择器，利用同步时序逻辑构建 BCD 计数器，最终集成实现为交通灯控制系统。

实验由简到难，层次递进，从器件到部件，从部件到系统，通过本实验的设计、仿真、验证 3 个训练过程使同学们掌握小型数字电路系统的设计、仿真、调试方法以及电路模块封装的方法。

## 实验环境

软件：logisim-hust-20200118.exe 软件一套。

平台：https:/[/www.educoder.net/shixuns/g8vqp5xw/challenges](http://www.educoder.net/shixuns/g8vqp5xw/challenges)

## 实验内容

某个主干道与次干道公路十字交叉路口，为确保人员、车辆安全、迅速地通过，在交叉路口的每个入口处设置了红、绿、黄三色信号灯。红灯禁止通行；绿灯允许通行；黄灯亮提醒行驶中的车辆减速通行。交通灯控制系统示意图如图 1-1 所示。

设计一个交通灯控制系统，具体内容及要求如下：

1. 输入信号

输入信号包括高峰期信号 H，主干道通行请求 PCM，次干道通行请求 PCC 和总控制台控制信号 Online。

1. 输出信号

输出信号包括 1 个 7 段数码管显示数字，用于显示红灯、绿灯和黄灯的剩余时间；

6 个 Led 灯，用于显示主干道和次干道的红灯、绿灯和黄灯。

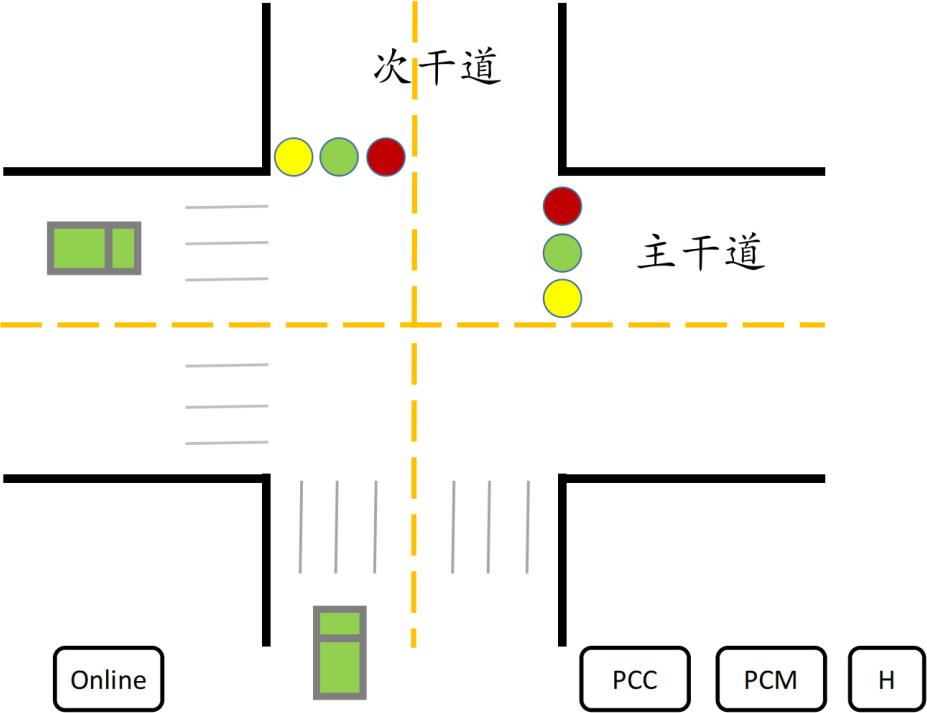


图 1-1 交通灯控制系统示意图

1. 具体功能
   1. 路口指示灯规则为：“红--绿--黄”循环；
   2. 控制参数假设：红灯 15 秒，绿灯 12 秒，黄灯 3 秒；
   3. 通行请求定义：主干道通行请求（PCM）包括：主干道方向有车辆信号和次干道有行人通过信号；次干道通行请求（PCC）包括：次干道方向有车辆信号和主干道有行人通过信号。
   4. 通行规则 1：主干道和次干道均无通行请求，主、次干道两边黄灯“闪亮”。提示：“通过时要注意观察”；
   5. 通行规则 2：主、次干道一边有通行请求，一边无通行请求，有通行请求一边绿灯亮，它的倒计时时间为 16s，归 0 后重新开始倒计时。
   6. 通行规则 3：只有主干道有通行请求 PCM，此时接收到次干道通行请求 PCC，则在绿灯倒计时为 0 时，考虑次干道方向的车辆或行人通行；只有次干道有通行请求的情况类似。
   7. 通行规则 4：非高峰时期，主、次干道均有通行请求时，主、次干道交替通行。
   8. 通行规则 5：高峰时期，主、次干道均有通行请求时，主、次干道交替通行，主干道放行时间（绿灯时间+黄灯时间）加倍。
   9. 通行规则 6：由交通控制中心发出的总控制台控制信号（Online），当 Online=1，本地交通灯控制器控制权“失效”，且主干道放行，次干道禁止通行、当 Online=0 本地交通灯控制器恢复控制权（接着原来的状态进行运行）。

## 实验要求

1. 根据给定的实验包，将交通灯控制系统切分为一个个实验单元；
2. 对每一个实验单元，按要求设计电路并使用 Logisim 软件进行虚拟仿真；
3. 设计好的电路在 educoder 平台上提交并进行评测，直到通过全部关卡。

# 实验总体设计

## 实验总体设计思路

先将系统拆分成模块，再将模块拆分成元件，然后从元件开始组装，连接。

1. 模块设计

使用交通灯状态转移与输出模块，倒计时选择模块，计时器模块，主干道倒计时模块，次干道倒计时模块，单侧通行倒计时模块以及紧急情况选择模块。

1. 元件设计
   1. 7 段数码管驱动电路：用于控制 7 段数码管。
   2. 8 位无符号比较器：首先设计 2 位无符号比较器，用 2 位无符号比较器组

成 4 位无符号比较器，进而组成 8 位无符号比较器。

* 1. 8 位 2 路选择器：首先设计 1 位 2 路选择器，进一步组成 8 位 2 路选择器。
  2. 2 位模十可逆计数器：用于计时。首先设计 1 位模十可逆计数器，然后扩

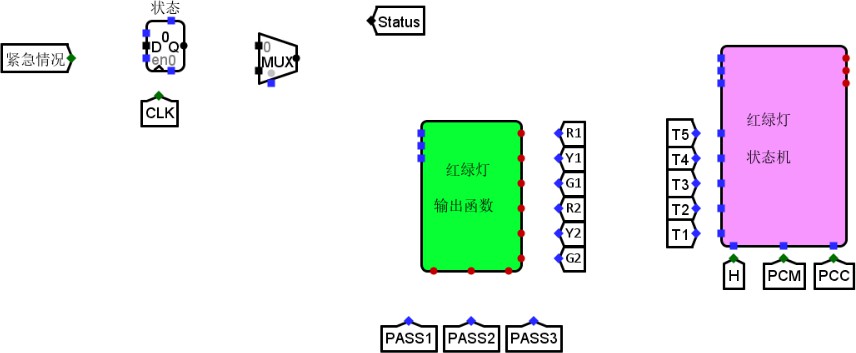
展得到 2 位模十可逆计数器。

## 实验总体设计框架

交通灯系统总体框架由交通灯系统状态转移和输出控制模块、倒计时选择模块、主干道倒计时模块、次干道倒计时模块、单侧同行倒计时模块、紧急情况控制模块以及输入和显示接口等组成。

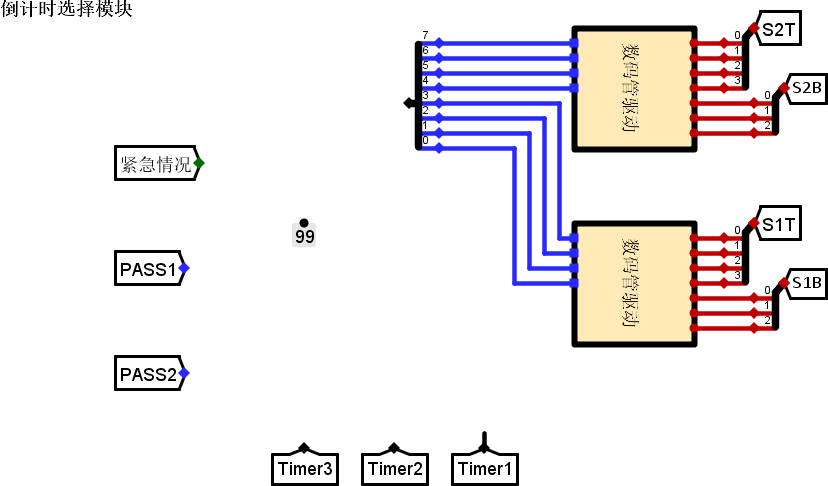
1. 红绿灯状态转移和输出模块

由交通灯状态机和交通灯输出函数组成。

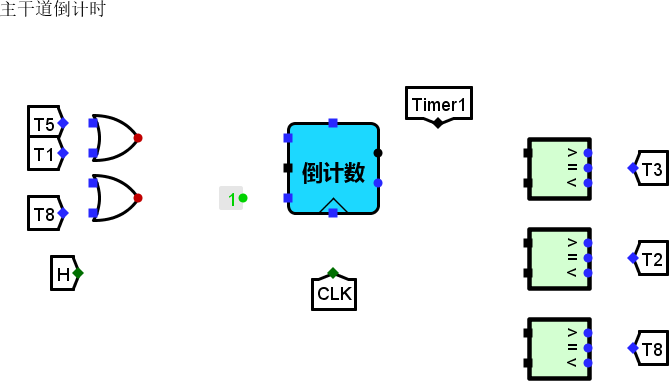


1. 倒计时选择模块

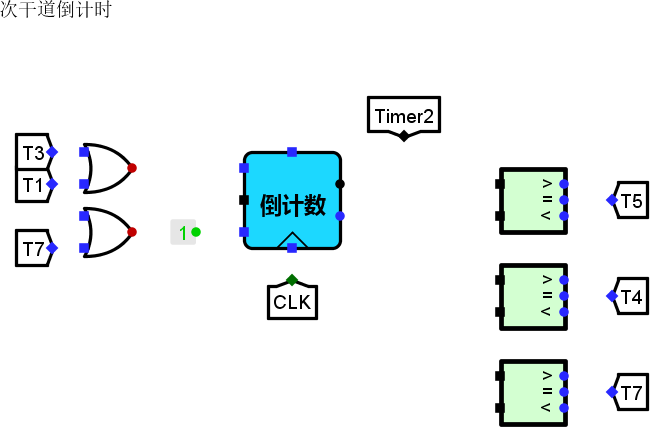
由三个 8 位 2 路选择器和两个 7 段数码管驱动构成



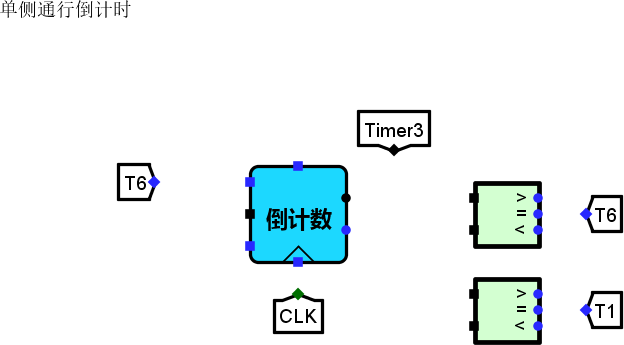
1. 主干道倒计时



1. 次干道倒计时



1. 单侧通行倒计时

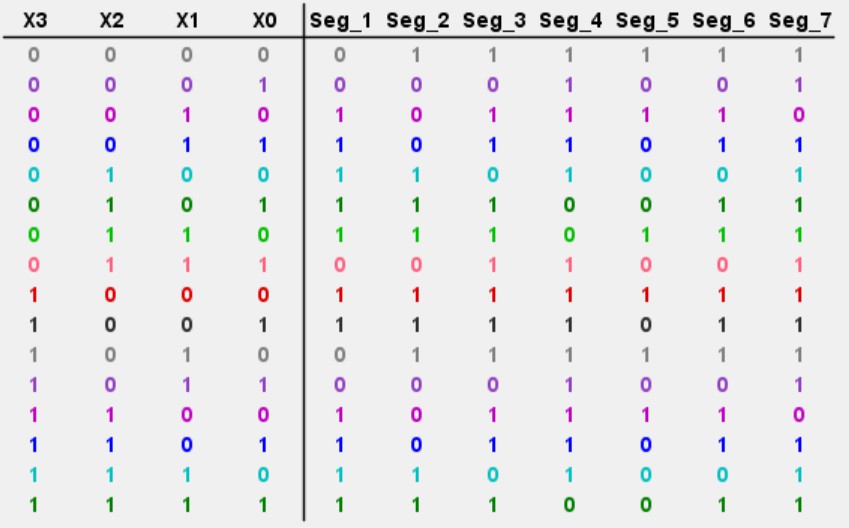


# 实验过程

## 7 段数码管驱动电路设计

1. 设计思路及设计过程

根据数码管引脚顺序设计真值表，使用真值表自动生成电路。



1. 电路图

图 3.1.1 7 段数码管驱动真值表

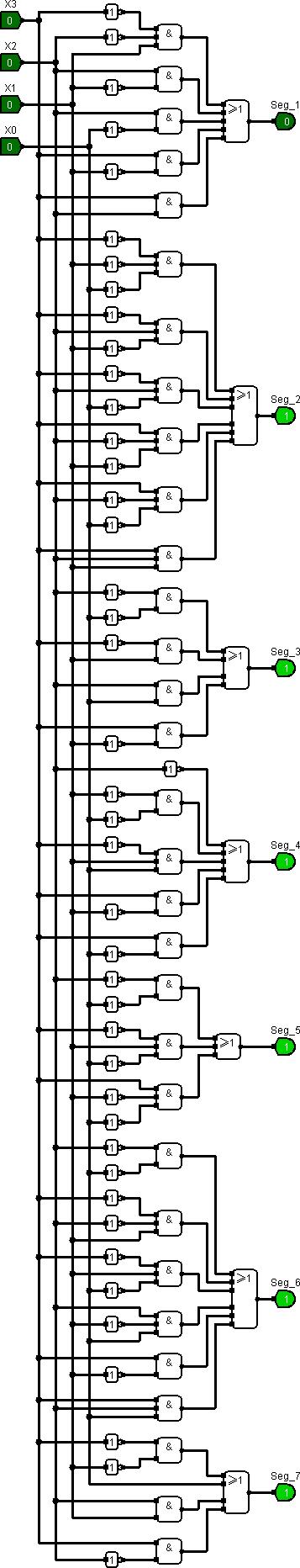
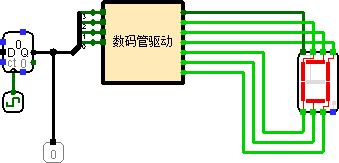


图 3.1.2 7 段译码器电路图

1. 测试图



1. 测试分析

图 3.1.3 7 段译码器测试电路图

* 1. 测试用例 1: 使得输入 BCD 码为 0010，如图 3.1.4 输出为 2，符合预期。

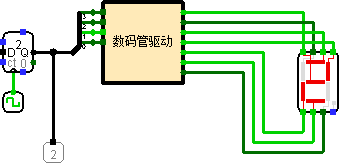


图 3.1.4 7 段译码器测试用例 1

* 1. 测试用例 2：使得输入 BCD 码为 0100，如图 3.1.5 输出为 4，符合预期。

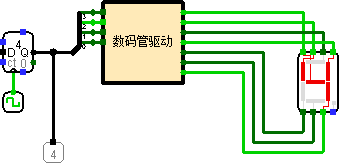


图 3.1.5 7 段译码器测试用例 2

1. 测试用例 3：使得输入 BCD 码为 1001，如图 3.1.6 输出为 9，符合预期。

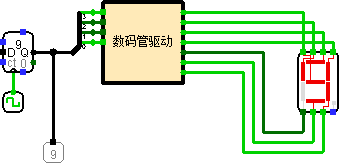


图 3.1.6 7 段译码器测试用例 3

## 无符号比较器（2 位、4 位、8 位）

1. 设计思路及设计过程
   1. 2 位无符号比较器，根据先比高位，再比低位的原则，由条件表达式写出逻辑表达式。

𝐺𝑟𝑒𝑎𝑡 = (𝑋1 > 𝑌1)||(𝑋1 == 𝑌1)&&(𝑋0 > 𝑌0)

= X1Y1 + (X1 ⊙ Y1)X0Y0

𝐸𝑞𝑢𝑎𝑙 = (𝑋1 == 𝑌1)(𝑋0 == 𝑌0)

= (𝑋1 ⊙ 𝑌1)(𝑋0 ⊙ 𝑌0)

𝐿𝑒𝑠𝑠 = (𝑋1 < 𝑌1)||(𝑋1 == 𝑌1)&&(𝑋1 < 𝑌1)

= 𝑋1Y1 + (𝑋1 ⊙ 𝑌1)𝑋0Y0

根据表达式自动生成电路。

* 1. 4 位无符号比较器，使用 2 位无符号比较器先比较高 2 位，再比较低 2

位。逻辑与 2 位无符号比较器相同，手动连接电路。

* 1. 8 位无符号比较器，使用 4 位无符号比较器先比较高 4 位，再比较低 4

位。逻辑与 2 位无符号比较器相同，手动连接电路。

1. 电路图

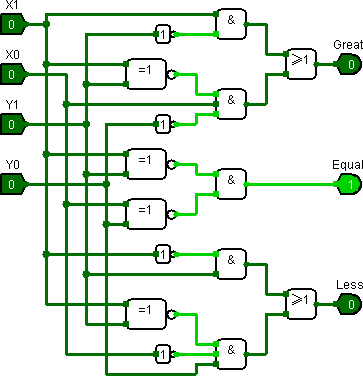
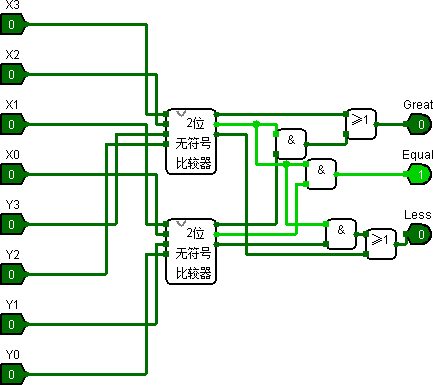
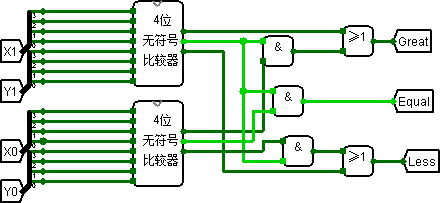
 

图 3.2.1 2 位无符号比较器 图 3.2.2 4 位无符号比较器



1. 测试图

图 3.2.3 8 位无符号比较器

* 1. 2 位无符号比较器

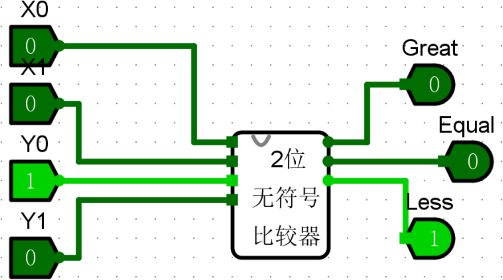
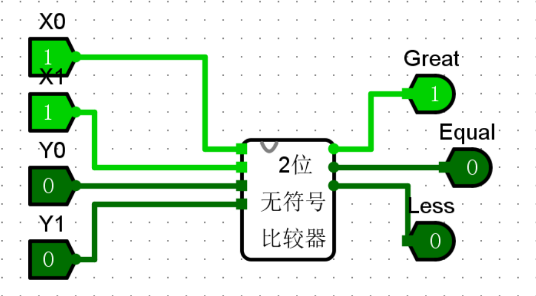
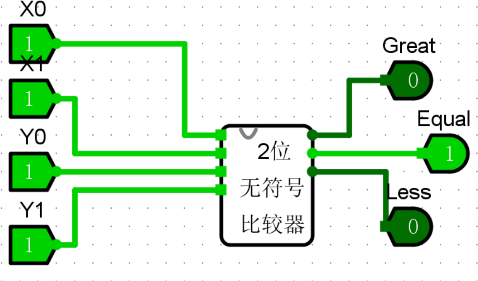


图 3.2.4 2 位无符号比较器测试电路

* 1. 4 位无符号比较器

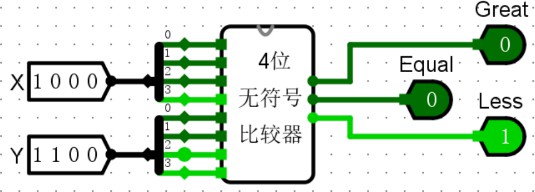
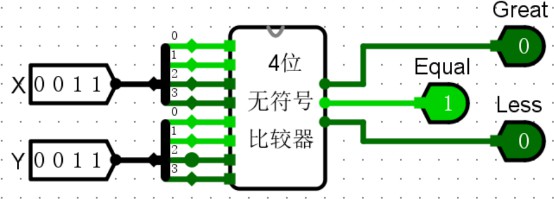
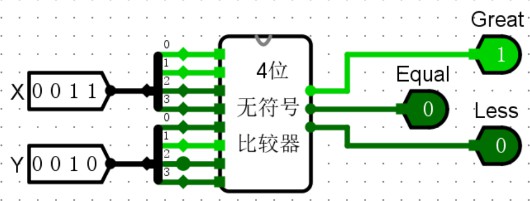
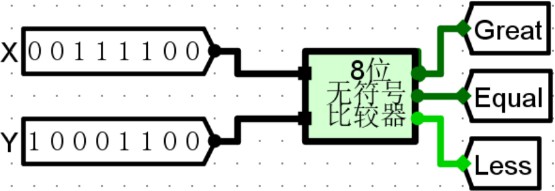
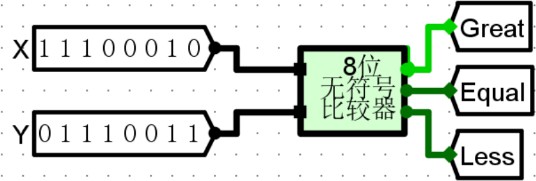


图 3.2.5 4 位无符号比较器测试电路

* 1. 8 位无符号比较器

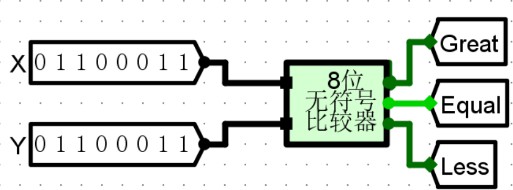
1. 测试分析

图 3.2.6 8 位无符号比较器测试电路

* 1. 2 位无符号比较器

表 3.2.1 2 位无符号比较器测试样例及结果

|  |  |  |
| --- | --- | --- |
| 输入(X1X0 Y1Y0) | 预期输🎧 | 实际输🎧 |
| 11 11 | Equal | Equal |
| 11 00 | Great | Great |
| 00 10 | Less | Less |

实验输出符合预期，功能正常。

* 1. 4 位无符号比较器

表 3.2 4 位无符号比较器测试样例及结果

|  |  |  |
| --- | --- | --- |
| 输入(X, Y) | 预期输🎧 | 实际输🎧 |
| 0011 0010 | Great | Great |
| 0011 0011 | Equal | Equal |
| 1000 1100 | Less | Less |

实验输出符合预期，功能正常。

* 1. 8 位无符号比较器

表 3.2.3 8 位无符号比较器测试样例及结果

|  |  |  |
| --- | --- | --- |
| 输入(X, Y) | 预期输🎧 | 实际输🎧 |
| 01100011 01100011 | Equal | Equal |
| 00111100 10001100 | Less | Less |
| 11100010 01110011 | Great | Great |

实验输出符合预期，功能正常。

* 1. 2 选 1 选择器设计（1 位、8 位）

1. 设计思路及设计过程

首先设计 1 位二路选择器，由表达式

Out = Sel ⋅ X1 + Sel ⋅ X0

使用一个非门，两个与门，一个或门连接电路。

实现 1 位二路选择器后，上述电路基本不变，将 Sel 和逻辑门位宽扩展为

8，即可实现 8 位 2 路选择器。

1. 电路图

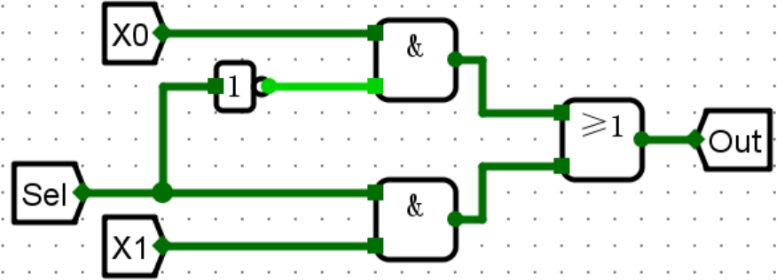


图 3.3.1 1 位二路选择器

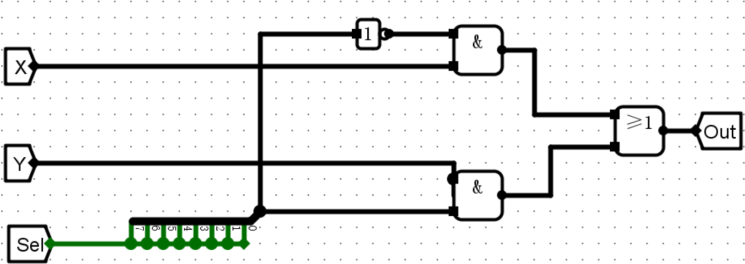


图 3.3.2 8 位二路选择器

1. 测试图

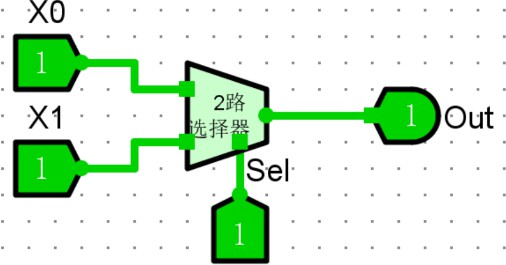
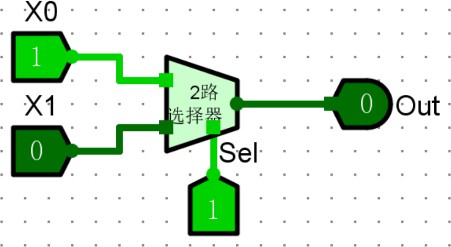
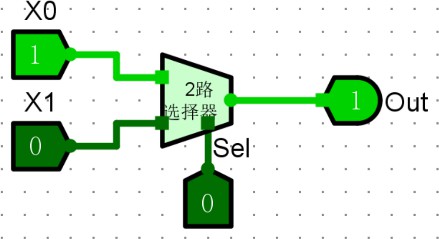
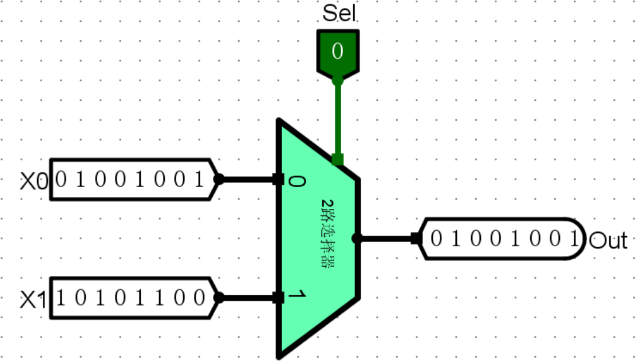


图 3.3.3 1 位二路选择器测试电路



1. 测试分析

图 3.3.4 8 位二路选择器测试电路

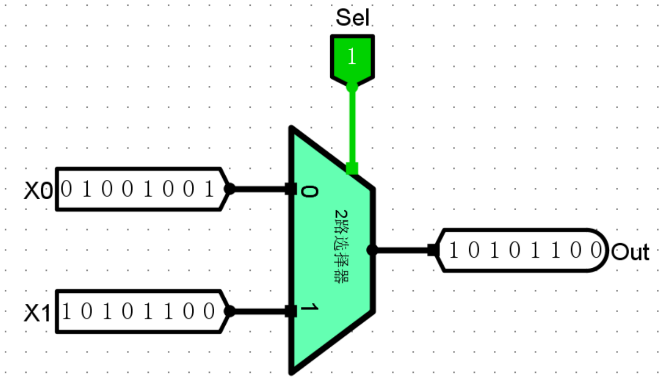
* 1. 1 位二路选择器

表 3.3.1 1 位 2 路选择器测试样例

|  |  |  |
| --- | --- | --- |
| 输入(X0 X1 Sel) | 预期输🎧(Out) | 实际输🎧 |
| 1 0 0 | 1 | 1 |
| 1 0 1 | 0 | 0 |
| 1 1 1 | 1 | 1 |

实验输出符合预期，功能正常。

* 1. 8 位二路选择器

表 3.3.2 8 位 2 路选择器测试样例

|  |  |  |
| --- | --- | --- |
| 输入(X0 X1 Sel) | 预期输🎧(Out) | 实际输🎧 |
| 01001001 10101100 1 | 10101100 | 10101100 |
| 01001101 00011100 0 | 01001101 | 01001101 |

实验输出符合预期，功能正常。

## 十进制可逆计数器（包含状态机、输出函数及整体电路）

1. 设计思路及设计过程

先分别设计状态机电路与输出函数电路，再组成整体电路。

* 1. 先完成状态机电路，利用 excel 表格自动生成表达式。如图 3.4.1 所示，状态机接收 5 个输入，分别为代表现态的 S3S2S1S0，代表计数模式的 Mode，输出为次态 N3N2N1N0。Mode 为 0 时，加法计数，现态为 9 时变为 0，Mode 为 1 时减法计数，现态为 0 时变为 9。

图 3.4.1 模十可逆计数器状态转换表

* 1. 再完成输出函数电路。电路具有五个输入，分别为代表现态的 S3、S2、 S1、S0，表示计数模式的为 Mode，输出 Cout 为当前的进位或借位信号。当 Mode 为 1，现态状态数为 0 时，发出借位信号，Cout 为 1；当 Mode 为 0，现态状态数为 9 时，发出进位信号，Cout 为 1。根据表达式：

𝐶𝑜𝑢𝑡 = 𝑆3 𝑆2 𝑆1 𝑆0 𝑀𝑜𝑑𝑒 + 𝑆3 𝑆2 𝑆1 𝑆0 𝑀𝑜𝑑𝑒

自动生成电路。

* 1. 最后完成整体电路，利用设计完成的十进制可逆计数器的状态机和输出函数，采用 D 触发器设计可逆十进制计数器。该计数器支持异步预置功能，当预置控制位 PreSet 信号为 1，直接通过 D 触发器的异步置位端

口，将 Din 数据传入到触发器中，另外通过输出函数电路对 Mode 和现态进行处理得到 Cout 输出，然后使用分线器根据现态输出 Q 的状态，并且将现态和 Mode 共同接入到状态转换电路、次态接入到 D 触发器的激励端口来实现状态转换，最后把时钟端 CLK 和使能端 En 接入到 D 触发器中，完成模十可逆计数器的设计

1. 电路图
   1. 模十可逆计数器转换电路

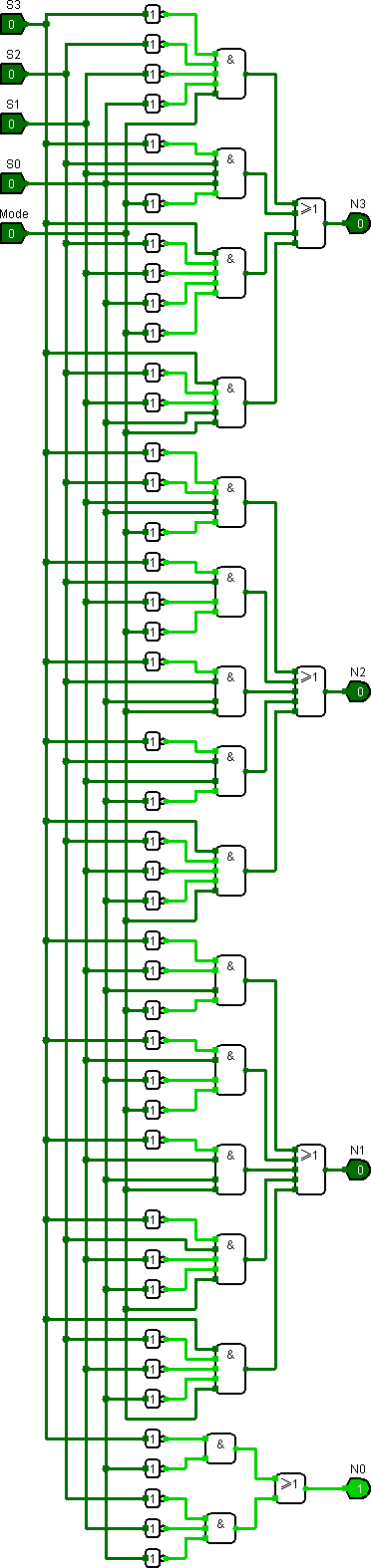


图 3.4.2 模十可逆计数器转换电路

* 1. 模十可逆计数器输出函数电路

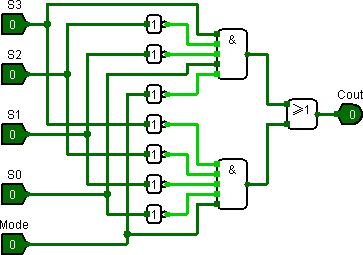
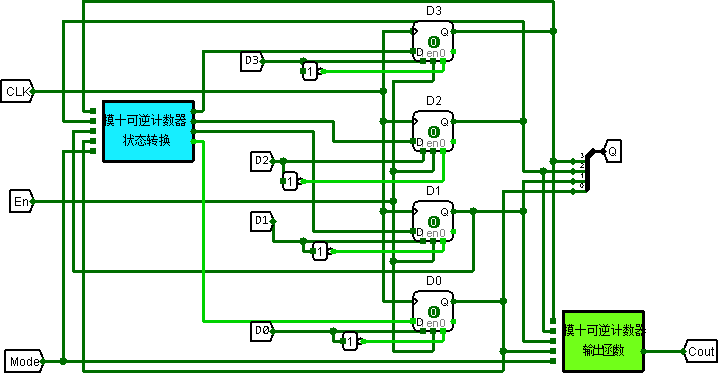


图 3.4.3 模十可逆计数器输出函数电路

* 1. 模十可逆计数器电路



1. 测试图

图 3.4.4 模十可逆计数器电路

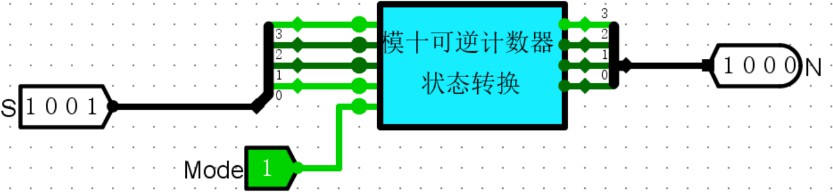
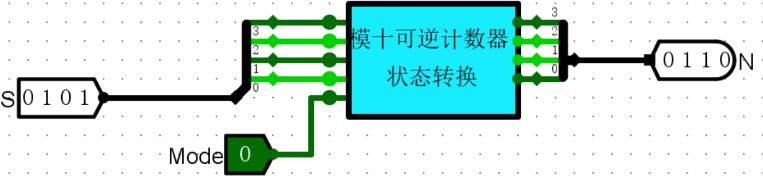
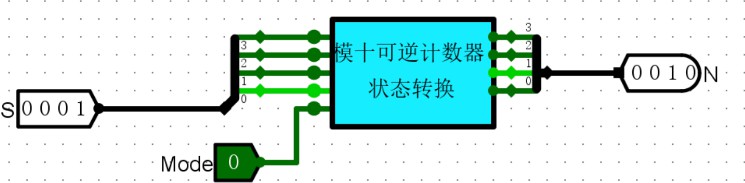
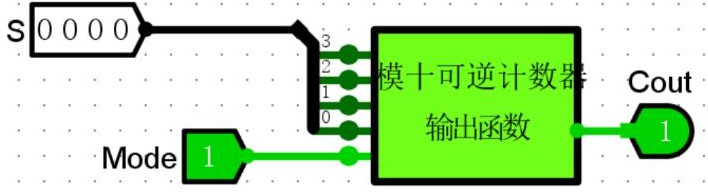
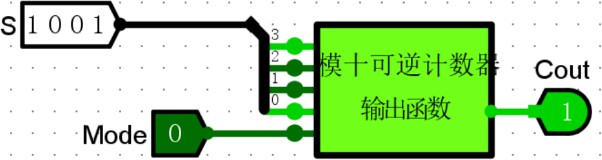
* 1. 模十可逆计数器状态机

图 3.4.5 模十可逆计数器状态机测试电路

* 1. 模十可逆计数器输出函数电路



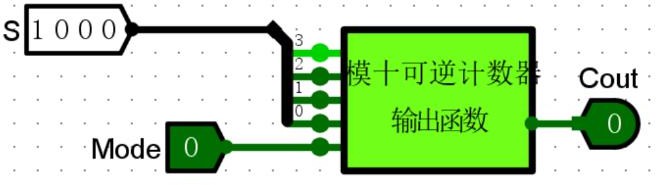


图 3.4.6 模十可逆计数器输出函数测试电路

* 1. 模十可逆计数器电路

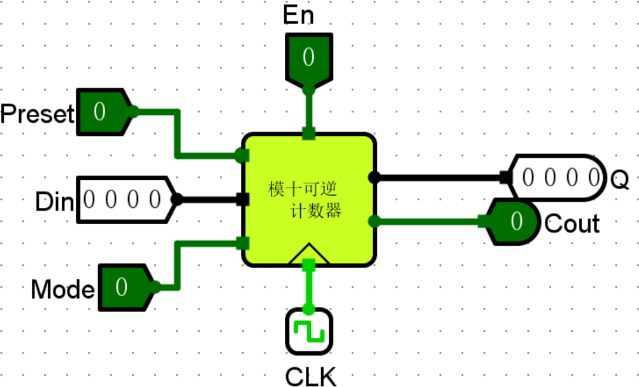


图 3.4.7 模十可逆计数器测试电路

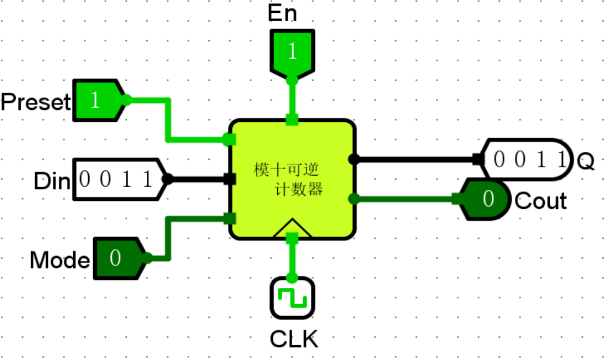
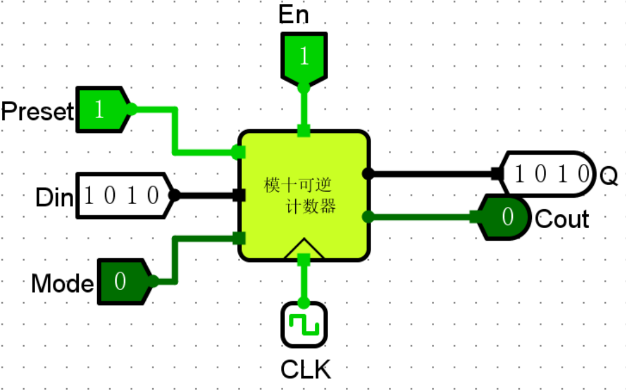


图 3.4.8 模十可逆计数器测试电路

1. 测试分析
   1. 模十可逆计数器状态机

图 3.4.9 模十可逆计数器测试电路

表 3.4.1 模十可逆计数器状态机测试样例

|  |  |  |
| --- | --- | --- |
| 输入(S Mode) | 预期输🎧(N) | 实际输🎧 |
| 0001 0 | 0010 | 0010 |
| 0101 0 | 0110 | 0110 |
| 1001 1 | 1000 | 1000 |

实验输出符合预期，功能正常。

* 1. 模十可逆计数器输出函数电路

表 3.4.1 模十可逆计数器输出函数测试样例

|  |  |  |
| --- | --- | --- |
| 输入(S Mode) | 预期输🎧(Cout) | 实际输🎧 |
| 1001 0 | 1 | 1 |
| 0000 1 | 1 | 1 |
| 1000 0 | 0 | 0 |

实验输出符合预期，功能正常。

* 1. 模十可逆计数器电路

先测试使能端 En，如图 3.4.7，当 En 为 0 时，模十可逆计数器输出不会发生改变，En 为 1 时，模十可逆计数器正常工作。

再测试异步预置功能，如图 3.4.8 与 3.4.9 所示，Preset 为 1 时，Din 为 1010 时，状态 Q 为 1010，Din 为 0011 时，状态 Q 为 0011，功能正常。最后测试模十可逆计数器整体电路计数功能，如表 3.4.2 所示。

表 3.4.2 模十可逆计数器整体电路计数功能测试

|  |  |  |
| --- | --- | --- |
| 现态/输入(Mode) | 预期 次态/输🎧(Cout) | 次态/输🎧(Cout) |
| 0001/1 | 0000/0 | 0000/0 |
| 0000/1 | 1001/1 | 1001/1 |
| 1001/0 | 0000/1 | 0000/1 |

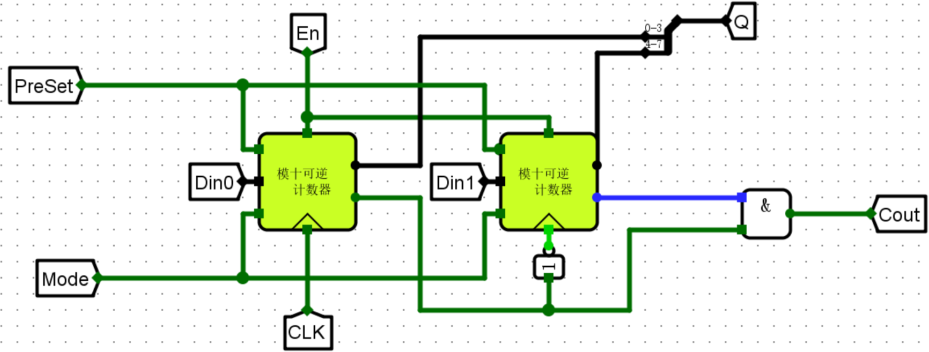
实验输出符合预期，功能正常。

## 两位十进制可逆计数器

1. 设计思路及设计过程

将 CLK、Mode 和异步预置控制统一，然后把低位计数器的进位/借位端口 (Cout)接入到高位计数器的使能端，仅在低位借位或者进位时，高位计数发生变化。当高、低位计数器都发出借位/进位信号时，二位十进制计数器发出进位/借位信号。异步预置数据和输出数据通过分线器的拆分和组合，将信号分配给两个一位十进制计数器。

1. 电路图



1. 测试图

图 3.5.1 二位十进制可逆计数器电路

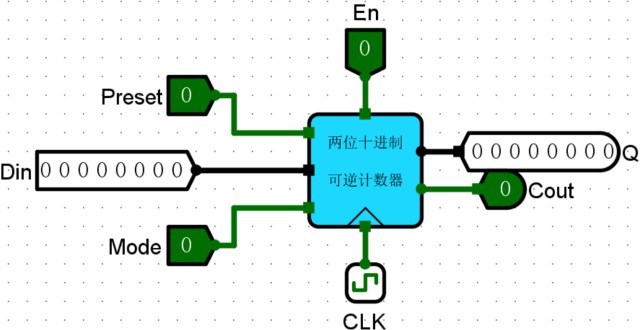
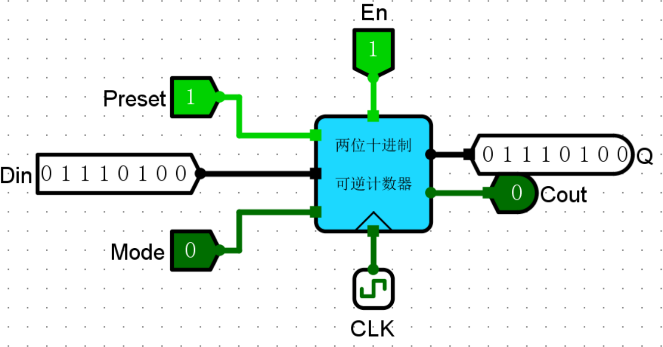


图 3.5.2 二位十进制可逆计数器测试电路



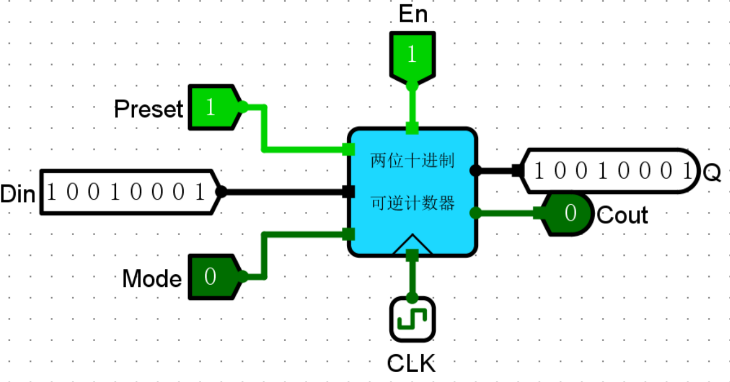
1. 测试分析
   1. 测试使能端 En，当 En 为 0 时，二位十进制可逆计数器输出不会发生改变，En 为 1 时，二位十进制可逆计数器正常工作。
   2. 测试预制功能，如表 3.5.1 所示。

表 3.5.1 二位十进制可逆计数器测试样例

|  |  |
| --- | --- |
| 预置状态/异步置位控制 | 实际状态 |
| 01110100/1 | 01110100 |
| 10010001/1 | 10010001 |

实验输出符合预期，功能正常。

* 1. 测试计数功能，如表 3.5.2 所示。

表 3.5.2 二位十进制可逆计数器测试样例

|  |  |  |
| --- | --- | --- |
| 现态/输入(Mode) | 预期 次态/输🎧(Cout) | 次态/输🎧(Cout) |
| 00000000/1 | 10011001/1 | 10011001/1 |
| 00000001/1 | 10010000/0 | 10010000/0 |
| 10011001/0 | 00000000/1 | 00000000/1 |

实验输出符合预期，功能正常。

## 交通灯状态机

1. 设计思路及设计过程

首先根据设计需求，设计出交通灯的 8 个状态：

S0 为两道路均无通行需求，主、次干道均为黄灯闪烁；

S1 为主干道有通行请求，次干道无通行请求，非高峰期主干道单侧通行，绿灯；

S2 为主干道无通行请求，次干道有通行请求，非高峰期次干道单侧通行，绿灯；

S3 为两侧都有通行需求，非高峰期主干道通行，绿灯； S4 为两侧都有通行需求，非高峰期主干道通行，黄灯； S5 为两侧均有通行需求，非高峰期次干道通行，绿灯； S6 为两侧均有通行需求，非高峰期次干道通行，黄灯； S7 为高峰期状态，主干道为绿灯。

根据上述需求，画出交通灯状态转换图，如图 3.6.1 所示。

最后作出交通灯的状态转换真值表如图 3.6.2 所示，并通过 excel 获得状态转换的逻辑函数表达式，使用 logisim 生成交通灯状态机的电路。

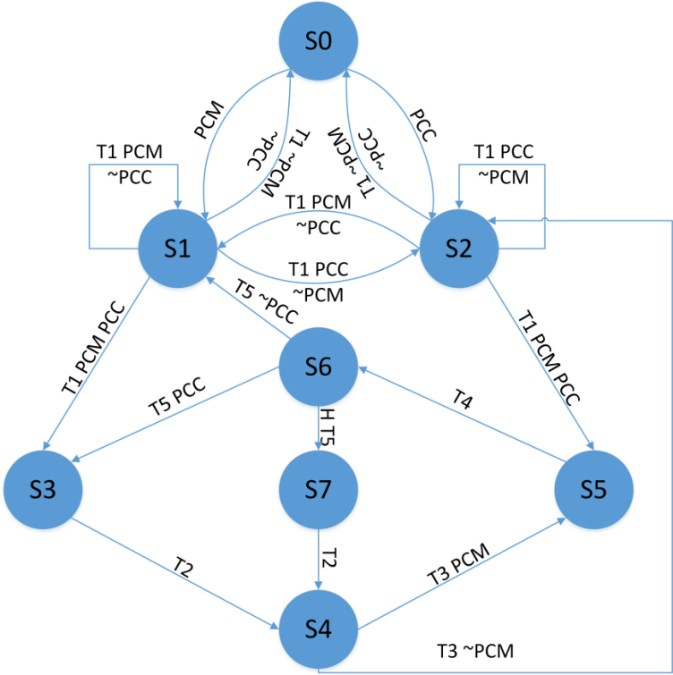
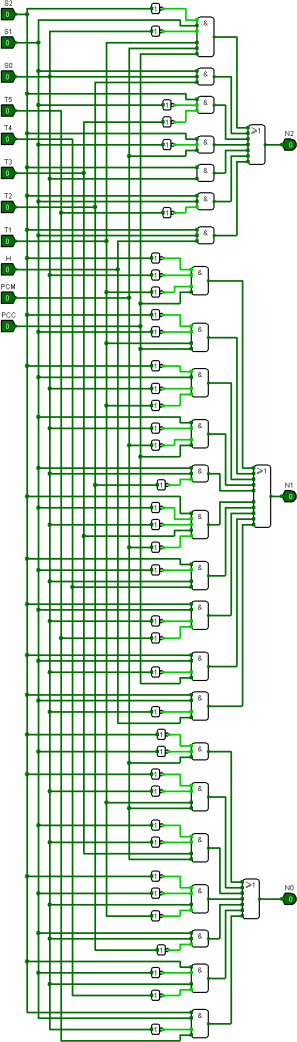


图 3.6.2 交通灯状态转换图

图 3.6.3 交通灯状态机状态转换表

1. 电路图



1. 测试图

图 3.6.4 交通灯状态机电路

如图 3.6.5 所示，输入引脚 S 从高到低位对应现态 S2,S1,S0，输入引脚 T 从高到低位表示输入信号 T5,T4,T3,T2,T1，输出引脚 N 高到低位对应次态 N2,N1,N0。

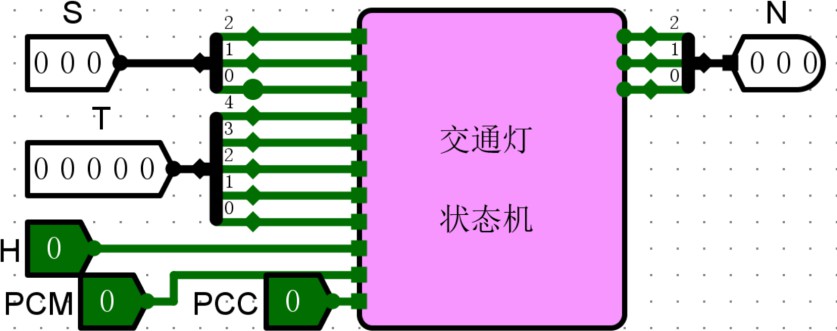


图 3.6.5 交通灯状态机测试电路

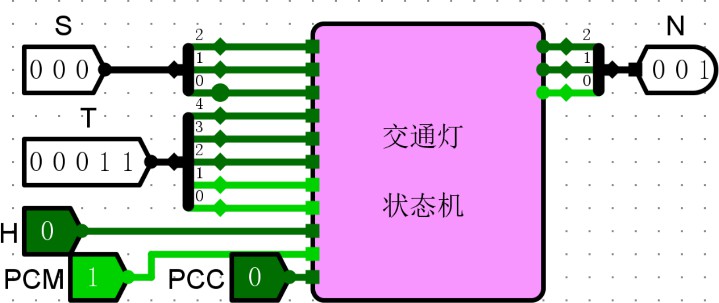
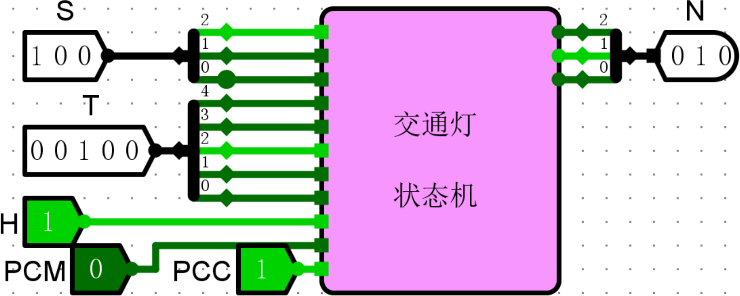
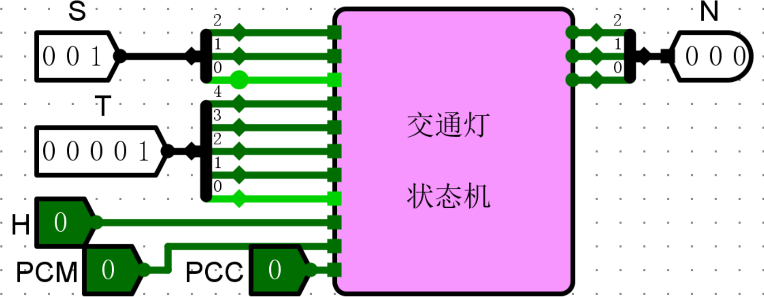
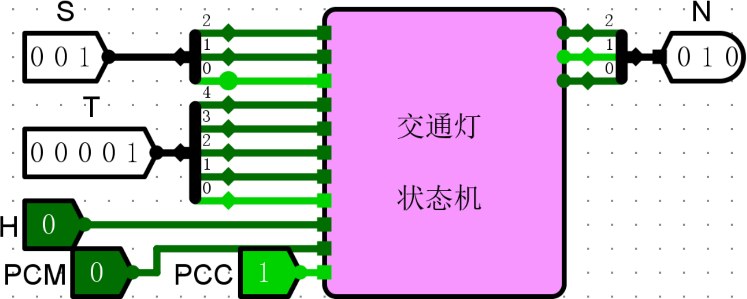


图 3.6.6 交通灯状态机测试样例

1. 测试分析

表 3.6.1 交通灯状态机测试样例

|  |  |  |
| --- | --- | --- |
| 现态(S)/输入(T/H/PCM/PCC) | 预期输🎧(N) | 实际输🎧(N) |
| 001/00001/0/0/1 | 010 | 010 |
| 001/00001/0/0/0 | 000 | 000 |
| 100/00100/1/0/1 | 010 | 010 |
| 000/00011/0/1/0 | 001 | 001 |

实验输出符合预期，功能正常。

## 交通灯输出函数设计

1. 设计思路及设计过程根据通行规则需求，得到交通灯状态如图 3.7.1 所示，交通灯输入输出信号如图 3.7.2 所示。



图 3.7.1 交通灯输出输入解析

图 3.7.2 交通灯状态解析

根据上述信息，输入状态与输出控制信号之间的关系，作出交通灯系统输出函数真值表如图 3.7.3 所示。

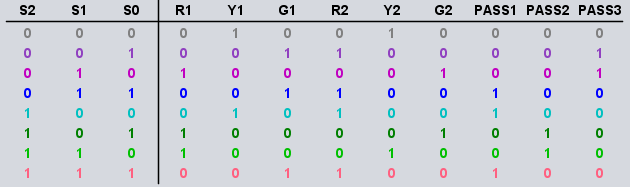
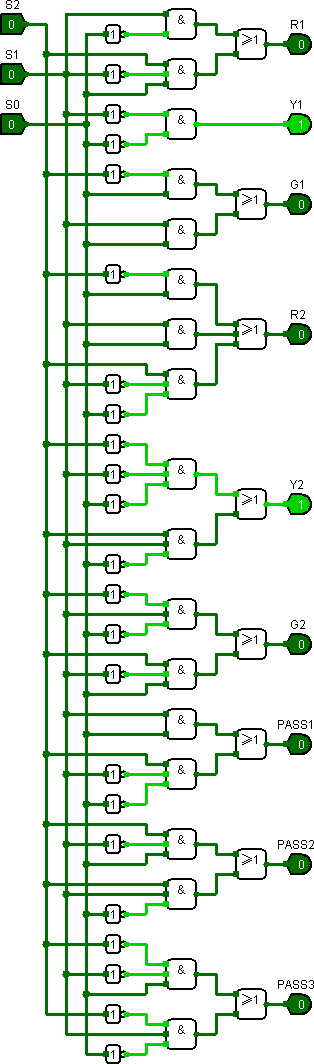


图 3.7.3 交通灯系统输出函数真值表

利用真值表在 logisim 中自动生成对应的逻辑电路。

1. 电路图



1. 测试图

图 3.7.4 交通灯输出函数电路

如图 3.7.5 所示，输入引脚 S 从高到低位对应交通灯状态 S2 S1 S0，输出 RYG 从高到低依次对应 R1、Y1、G1、R2、Y2、G2,PASS 从高到低依次对应 PASS1、PASS2、PASS3。

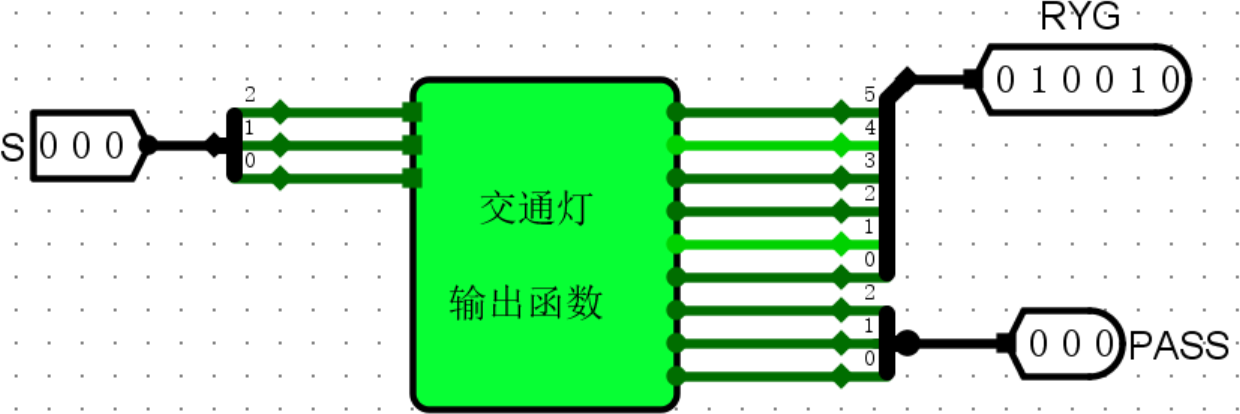


图 3.7.5 交通灯输出函数测试图

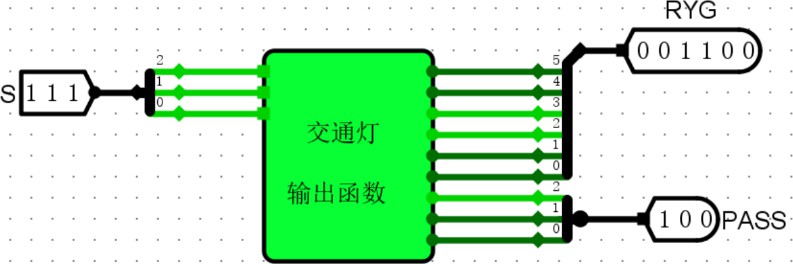
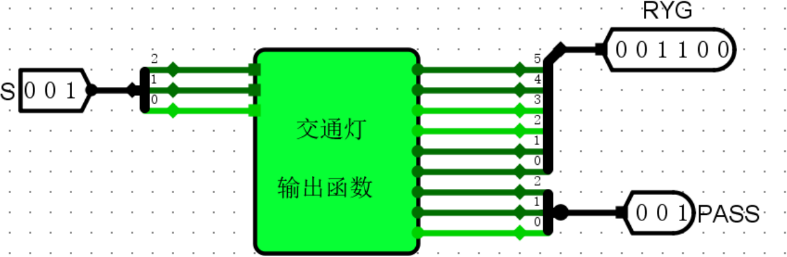
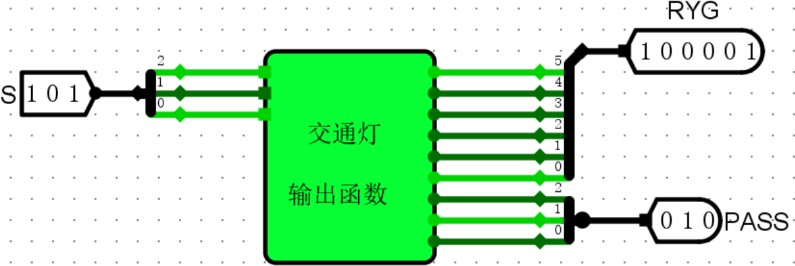


图 3.7.6 交通灯输出函数测试样例

1. 测试分析

表 3.7.1 交通灯输出函数测试样例

|  |  |  |
| --- | --- | --- |
| 输入(S) | 预期输🎧(RYG/PASS) | 实际输🎧(RYG/PASS) |
| 101 | 100001/010 | 100001/010 |
| 001 | 001100/001 | 001100/001 |
| 111 | 001100/100 | 001100/100 |

实验输出符合预期，功能正常。

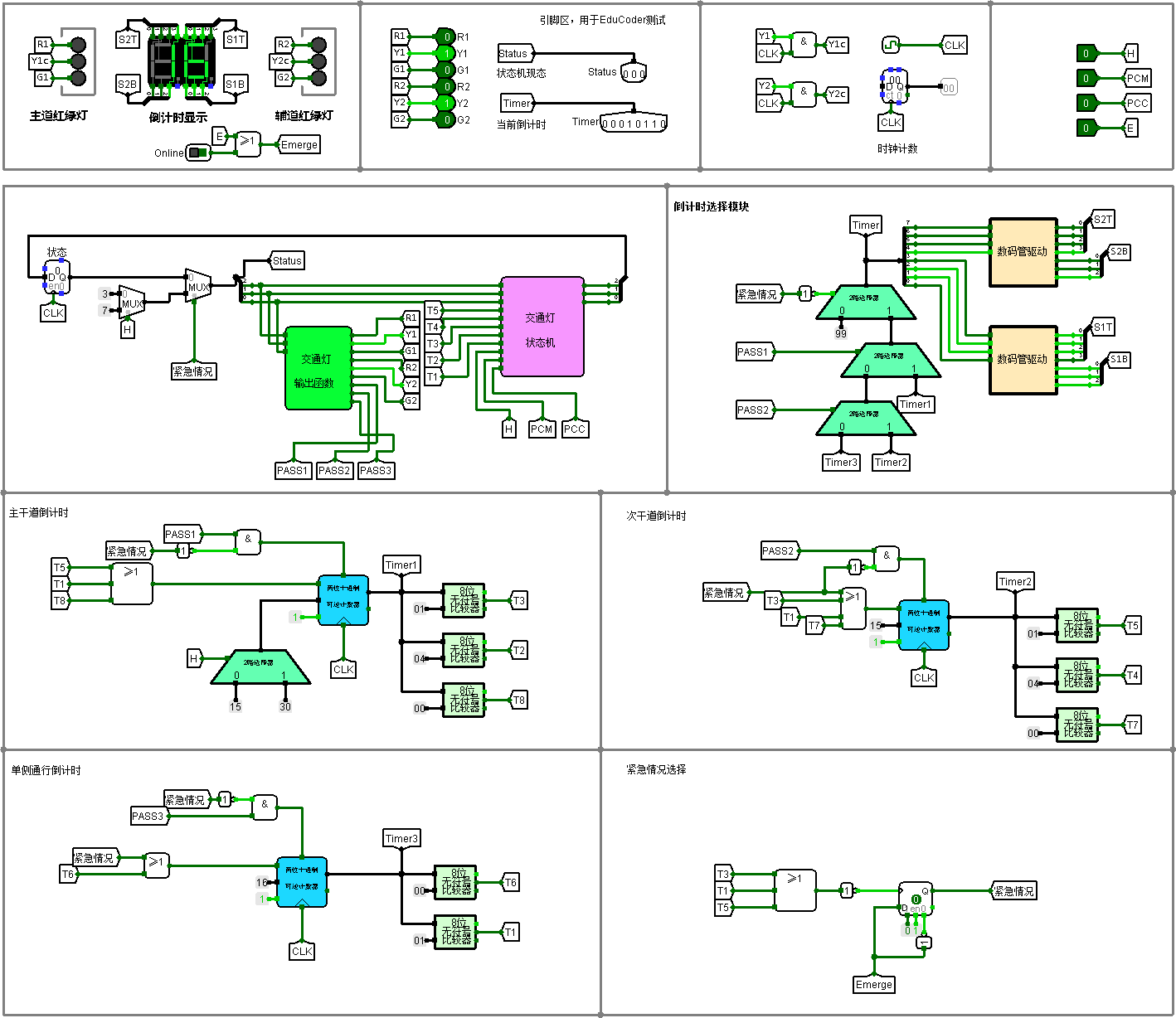
## 交通灯控制系统

1. 设计思路及设计过程

依次设计倒计时选择模块、主干道倒计时模块、次干道倒计时模块、单侧同行倒计时模块和紧急情况选择模块，并在主电路中增加紧急状况状态异步切换功能。最后将以上模块组合，得到交通灯系统的完整电路。

* 1. 倒计时选择模块：首先根据红绿灯系统的特性，使用三个 8 位 2 路选择器，判断是否为紧急情况，如果是紧急情况，则计时器始终显示 99；然后判断 PASS1 是否有效，如果有效，则使用主道计时器，反之，则检查 PASS2。PASS2 有效，则使用次道计时器；若 PASS1、PASS2 都无效，则使用单侧通行计时器。上述逻辑通过选择器实现后，连接至两个数码管驱动电路，实现数码管显示数据。
  2. 主干道倒计时模块：首先通过两位十进制可逆计数器，将计数状态传入 Timer1 通道作为主干道倒计，该计数器 Mode 置 1，始终倒计时。然后当次干道黄灯结束(T5)、单侧通行结束时(T1)，该计数器为倒计时初始状态，数值由高峰期信号(H)决定，高峰期通过 2 路选择器选择 30，普通时段选择 15。该计数器仅在 PASS1 有效且没有紧急情况下，即主道被许可通行时，才能进行计时，以此条件作为其使能端。最后，该模块通过使用两个 8 位无符号比较器进行比较、利用主干道绿灯结束和黄灯结束的信号，来实现交通灯状态转换，当计数状态为 01，说明黄灯结束，T3 有效；当计数状态为 04，说明绿灯结束，T2 有效。
  3. 次干道倒计时模块：设计和主干道类似，区别为次干道计数器异步预置值始终为 15。
  4. 单侧通行倒计时模块：该模块的使能端和前两个模块相似，异步预置控制端由 T6 控制。在计时为 01 时，T1 有效，如果交通灯状态使得 PASS3仍然有效，则计时进入 00，此时 T6 有效，倒计时被置位为 16，不断循环。
  5. 紧急情况选择模块：根据红绿灯系统的规则，当紧急开关打开后，只有在以上三个模块计时结束后，才能转换为紧急状态。因此使用异步脉冲时序电路的思想，将三个模块结束信号，即 T1、T3、T5 信号或运算的结果作为时钟信号，Emerge 信号作为 D 端口信号，Emerge 的反信号作为异步置零的信号，最后 D 触发器输出信号为紧急状态信号。

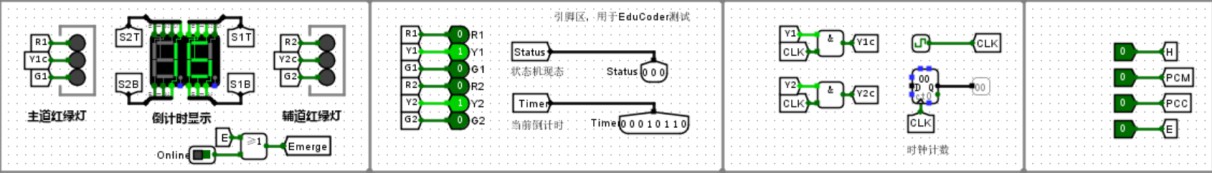
1. 电路图



1. 测试图

图 3.8.1 交通灯系统完整电路

使用如图 3.8.2 所示电路测试测试。



1. 测试分析

图 3.8.2 交通灯系统测试图

* 1. 样例 1，如图 3.8.3 所示,无通行信号，两个黄灯闪烁，倒计时恒为 16。实验符合预期。

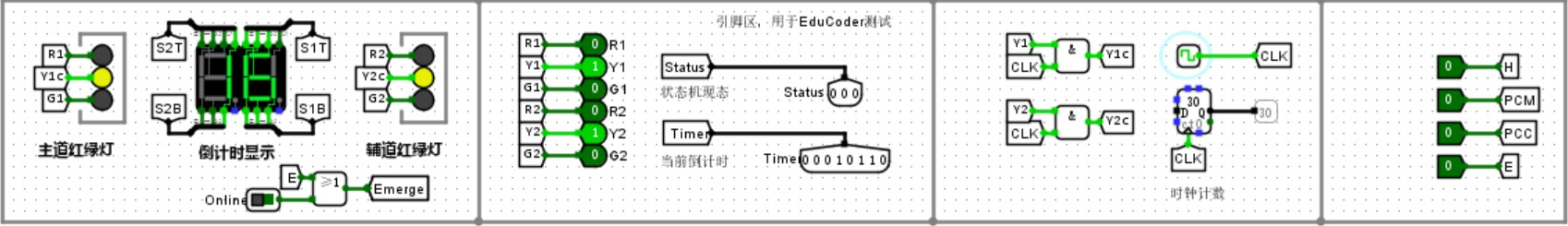


图 3.8.3 交通灯测试样例 1

* 1. 样例 2，如图 3.8.4 所示。两侧均有通行信号，为高峰期，红绿灯交替，且主干道倒计时为 30 秒。测试符合预期。

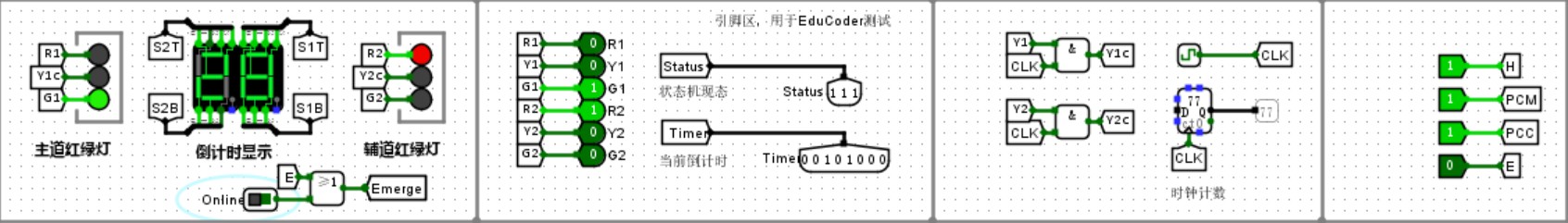


图 3.8.4 交通灯测试样例 2

* 1. 样例 3，如图 3.8.5 所示。主干道无通行信号，次干道有通行信号，单侧通行，倒计时从 16 开始。测试符合预期。

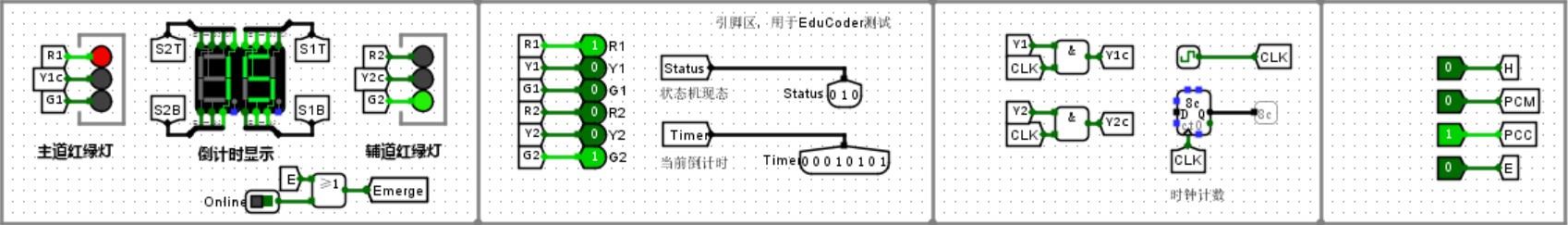


图 3.8.5 交通灯测试样例 3

* 1. 测试样例 4，如图 3.8.6 所示。进入紧急状态，倒计时始终为 99。测试符合预期。

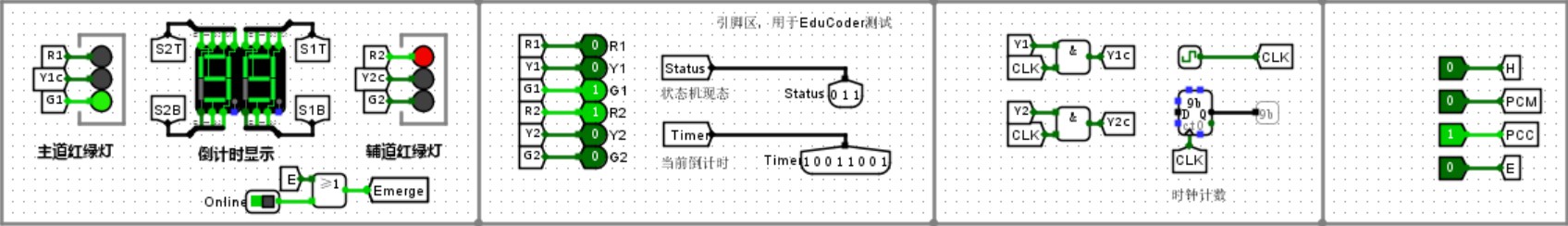


图 3.8.6 交通灯测试样例 4

以上测试结果均与预期相符，说明交通灯系统功能正常。

# 设计总结与心得

## 实验总结

这次试验中一开始由于缺少相关知识以及对 logisim 的使用不熟悉导致出现问题，最后的交通灯控制系统比较复杂，经过与同学们的讨论才明白原理。

## 遇到的问题及处理

* + - 1. 设计 7 段数码管驱动电路时，由与是第一个实验，对 logisim 的使用很不熟练，没有弄明白封装图引脚与实际电路图引脚之间的对应关心，导致输出错误。在询问同学之后，明白了 logisim 的使用方法，看清了引脚之间的对应关系之后成功将电路更正。
      2. 设计十进制可逆计数器时，由于数字逻辑课程还没有学到这里来，不清楚作用原理，导致不知道该怎么连接电路。在学习了相关知识后，明白了作用机制，成功设计出电路。
      3. 设计交通灯控制系统时，最开始时没有理解紧急状态模块的作用机制，对紧急状态模块的理解不到位。并且没有深刻理解脉冲的延迟作用，导致一直不能正确连接电路。最后，经过我与其他同学的长时间讨论与测试，逐渐理解了作用原理。最后成功完成了模块的设计。

## 设计方案存在的不足

交通灯状态机提供的状态转换图存在问题，同时最后的交通灯控制系统出现了一些之前的实验中未出现的东西，跨度较大而缺少提示信息。

## 实验心得

在这次实验中，通过 logisim 软件将数字电路的知识运用到实际中。让我体会到了数字逻辑电路在现实生活中的具体作用。这种自下而上，从易到难，先构造局部再组成整体的设计方法也让我受益匪浅。

此外，经过这次实验，我设计了各种时序逻辑电路，组合逻辑电路和异步脉冲逻辑电路。其中的选择器，比较器，状态机都是在课内学习过的。这次实践，让我对相关知

识的理解更深，提高了我的学科能力。

总之，本次实验加深了我对数字逻辑电路的理解，是一次难能可贵的经历。

## 意见与建议

实验时间与数字逻辑课程搭配得不是很好，导致做一些实验时还没有学到相应的知识，不能理解电路的内在逻辑。同时本次实验开始引导和提示十分细致，而到后面交通灯系统的设计出现了一些之前没出现过的内容，又没有相关提示，导致完成十分困难。因此，希望在后面交通系统设计上补充必要的信息，加强对该部分设计的引导内容。

|  |
| --- |
| 原创性声明 |
| 本人郑重声明本报告内容，是由作者本人独立完成的。有关观点、方法、数据和文献等的引用已在文中指出。除文中已注明引用的内容外，本报告不包含任何其他个人或集体已经公开发表的作品成果，不存在剽窃、抄袭行为。  已阅读并同意以下内容。 判定为不合格的一些情形：   1. 请人代做或冒名顶替者； 2. 替人做且不听劝告者； 3. 实验报告内容抄袭或雷同者； 4. 实验报告内容与实际实验内容不一致者； 5. 实验电路抄袭者。   **作者签名：** |