

华中科技大学

数字逻辑实验报告（1）

数字逻辑实验1		
一、系列二进制加法器 设计 50%	二、小型实验室门禁系 统设计 50%	总成绩

评语：（包含：预习报告内容、实验过程、实验结果及分析）

教师签名

姓 名： 吕鹏泽

学 号： U20161432

班 级： CS1601

指 导 教 师： 赵贻竹

计算机科学与技术学院

2018 年 5 月 31 日

华中科技大学

数字逻辑实验报告

系列二进制加法器设计预习报告

《数字电路与逻辑设计》实验报告

一、系列二进制加法器设计

1、实验名称

系列二进制加法器设计。

2、实验目的

要求同学采用传统电路的设计方法，对 5 种二进制加法器进行设计，并利用工具软件，例如，“logisim”软件的虚拟仿真功能来检查电路设计是否达到要求。

通过以上实验的设计、仿真、验证 3 个训练过程使同学们掌握传统逻辑电路的设计、仿真、调试的方法。

3、实验所用设备

Logisim2.7.1 软件一套。

4、实验内容

对已设计的 5 种二进制加法器，使用 logisim 软件对它们进行虚拟实验仿真，除逻辑门、触发器外，不能直接使用 logisim 软件提供的逻辑库元件，具体内容如下。

(1) 一位二进制半加器

设计一个一位二进制半加器，电路有两个输入 A、B，两个输出 S 和 C。输入 A、B 分别为被加数、加数，输出 S、C 为本位和、向高位进位。

(2) 一位二进制全加器

设计一个一位二进制全加器，电路有三个输入 A、B 和 C_i ，两个输出 S 和 C_o 。输入 A、B 和 C_i 分别为被加数、加数和来自低位的进位，输出 S 和 C_o 为本位和和向高位的进位。

(3) 串行进位的四位二进制并行加法器

用四个一位二进制全加器串联设计一个串行进位的四位二进制并行加法器，电路有九个输入 A_3 、 A_2 、 A_1 、 A_0 、 B_3 、 B_2 、 B_1 、 B_0 和 C_0 ，五个输出 S_3 、 S_2 、 S_1 、 S_0 和 C_4 。输入 $A=A_3A_2A_1A_0$ 、 $B=B_3B_2B_1B_0$ 和 C_0 分别为被加数、加数和来自低位的进位，输出 $S=S_3S_2S_1S_0$ 和 C_o 为本位和和向高位的进位。

(4) 先行进位的四位二进制并行加法器

《数字电路与逻辑设计》实验报告

利用超前进位的思想设计一个先行进位的四位二进制并行加法器，电路有九个输入 A_3 、 A_2 、 A_1 、 A_0 、 B_3 、 B_2 、 B_1 、 B_0 和 C_0 ，五个输出 S_3 、 S_2 、 S_1 、 S_0 和 C_4 。输入 $A = A_3A_2A_1A_0$ 、 $B = B_3B_2B_1B_0$ 和 C_0 分别为被加数、加数和来自低位的进位，输出 $S = S_3S_2S_1S_0$ 和 C_4 为本位和和向高位的进位。

(5) 将先行进位的四位二进制并行加法器封装成一个组件并验证它的正确性

将设计好的先行进位的四位二进制并行加法器进行封装，生成一个“私有”库元件并验证它的正确性，以便后续实验使用，封装后的逻辑符号参见图 1-1 所示。

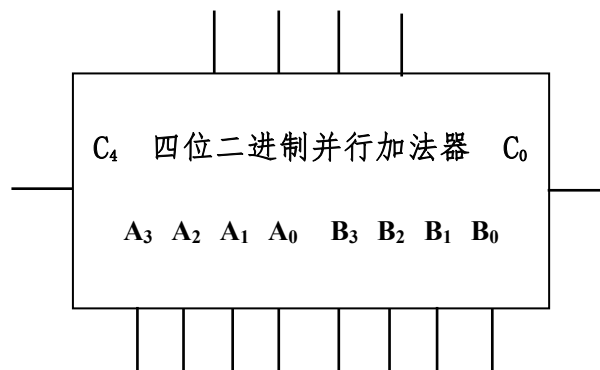


图 1-1 “私有”的先行进位的四位二进制并行加法器

5、实验方案设计

(1) 一位二进制半加器的设计方案

设 A 、 B 为半加器的输入， C 、 S 为半加器的输出，其中 S 为本位和， C 为进位，通过分析可知，当 $A=B=0$ 时， $C=S=0$ ；当 A 、 B 中有一个为 1 时， $C=0$ ， $S=1$ ；当 $A=B=1$ 时， $C=1$ ， $S=0$ 。据此写出逻辑表达式：

$$S = A \oplus B$$

$$C = AB$$

使用 logism 做出一位二进制半加器的电路图，结果如图 1-2 所示。

《数字电路与逻辑设计》实验报告

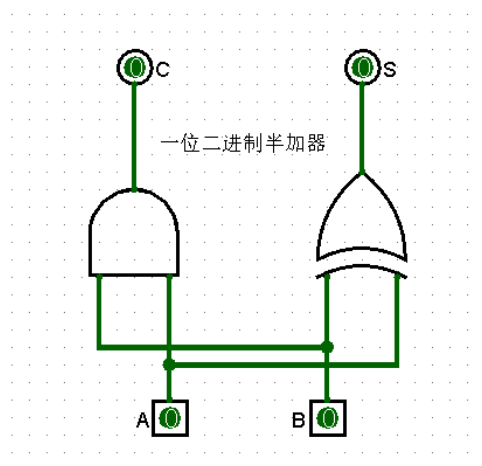


图 1-2 一位二进制半加器

(2) 一位二进制全加器的设计方案

设 A、B、 C_i 为全加器的输入，S、 C_o 为输出，其中，A、B 和 C_i 分别为被加数、加数和来自低位的进位，输出 S 和 C_o 为本位和和向高位的进位，据此可以列出全加器的真值表如表 1-1 所示。

表 1-1 全加器真值表

A	B	C_i	S	C_o
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

利用卡诺图化简并进行异或变换得到最简输出函数表达式为

$$S = A \oplus B \oplus C_i$$

$$C_o = AB + AC_i + BC_i = (A_i \oplus B_i)C_{i-1} + A_iB_i$$

使用 logism 做出一位二进制全加器的电路图，结果如图 1-3 所示。

《数字电路与逻辑设计》实验报告

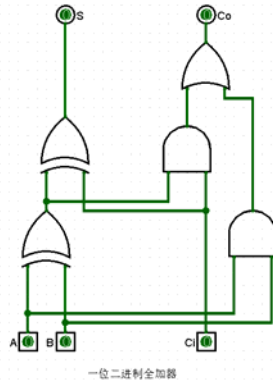


图 1-3 一位二进制全加器

(3) 串行进位的四位二进制并行加法器的设计方案

串行进位的四位二进制并行加法器可以由四个一位二进制全加器级联构成，其中高位的 C_i 即为其相邻低位的 C_o ，因此电路从最低位开始运算，得到本进位 C_i 以及本位后进行次低位的运算，以此类推，直到运算到最高位。设输入 $A = A_3A_2A_1A_0$ 、 $B = B_3B_2B_1B_0$ 和 C_o 分别为被加数、加数和来自低位的进位，输出 $S = S_3S_2S_1S_0$ 和 C_o 为本位和和向高位的进位。

使用 logism 做出串行进位的四位二进制并行加法器的电路图，结果如图 1-4 所示。

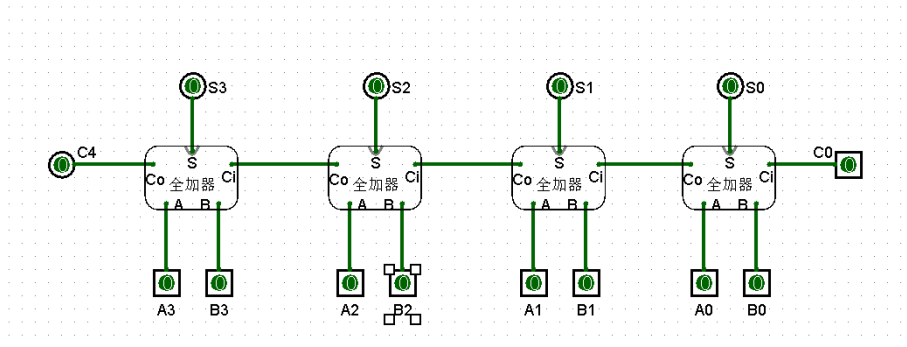


图 1-4 串行进位的四位二进制并行加法器

(4) 先行进位的四位二进制并行加法器的设计方案

串行进位的并行加法器高位的运算需要低位的运算结果参与，因此运算速度较慢，需对其进行优化。由全加器的逻辑表达式可知第 i 位的进位输出函数 C_o 及本位和函数 S_i 的表达式为

$$C_i = A_i B_i + A_i C_{i-1} + B_i C_{i-1} = (A_i \oplus B_i) C_{i-1} + A_i B_i$$

$$S_i = A_i \oplus B_i \oplus C_{i-1}$$

当第 i 位被加数 A_i 和 B_i 均为 1 时，有 $C_i=1$ ，定义 $G_i = A_i B_i$ 为进位产生函数。当 $A_i \oplus B_i = 1$ 时， $C_i = C_{i-1}$ ，定义 $P_i = A_i \oplus B_i$ 为进位传递函数，将 P_i 和 C_i 代入 C_i 和 S_i 中

《数字电路与逻辑设计》实验报告

得到

$$S_i = P_i \oplus C_{i-1}$$

$$C_i = P_i C_{i-1} + G_i$$

因此 4 位二进制并行加法器各位的进位输出函数和输出函数表达式分别为：

$$C_1 = P_1 C_0 + G_1$$

$$C_2 = P_2 C_1 + G_2 = P_2 P_1 C_0 + P_2 G_1 + G_2$$

$$C_3 = P_3 C_2 + G_3 = P_3 P_2 P_1 C_0 + P_3 P_2 G_1 + P_3 G_2 + G_3$$

$$C_4 = P_4 C_3 + G_4 = P_4 P_3 P_2 P_1 C_0 + P_4 P_3 P_2 G_1 + P_4 P_3 G_2 + P_4 G_3 + G_4$$

$$S_0 = P_1 \oplus C_0$$

$$S_1 = P_2 \oplus C_1$$

$$S_2 = P_3 \oplus C_2$$

$$S_3 = P_4 \oplus C_3$$

使用 logism 作出先行进位的四位二进制并行加法器的电路图,结果如图 1-5 所示。

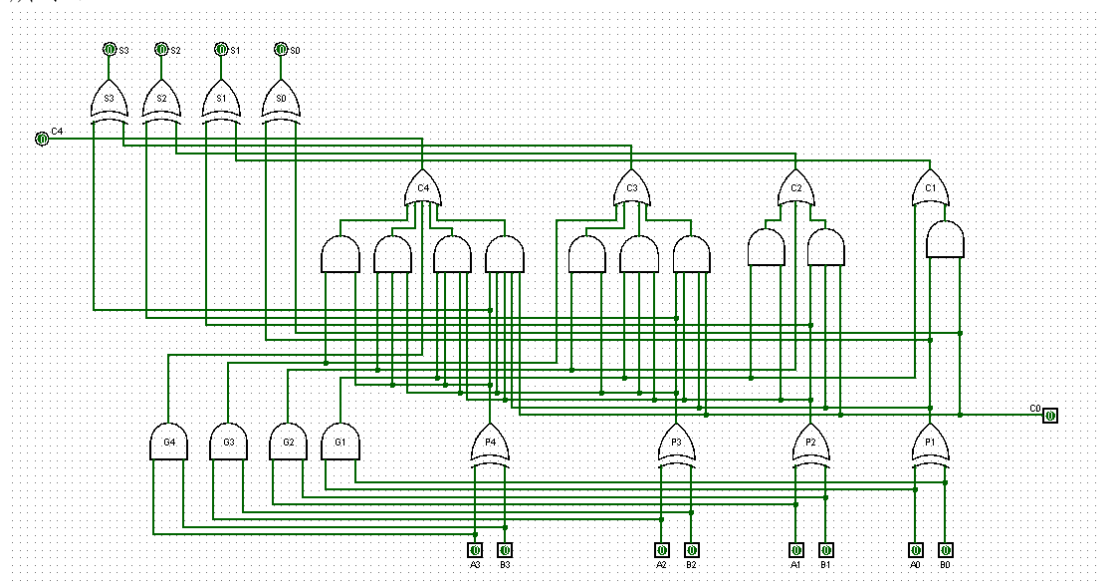


图 1-5 先行进位的四位二进制并行加法器

(5) 封装先行进位的四位二进制并行加法器电路

对先行进位的四位二进制并行加法器进行封装, 其中输入 $A = A_3A_2A_1A_0$ 、 $B = B_3B_2B_1B_0$ 和 C_0 分别为被加数、加数和来自低位的进位, 输出 $S = S_3S_2S_1S_0$ 和 C_4 为本位和和向高位的进位。

封装结果如图 1-6 所示。

《数字电路与逻辑设计》实验报告

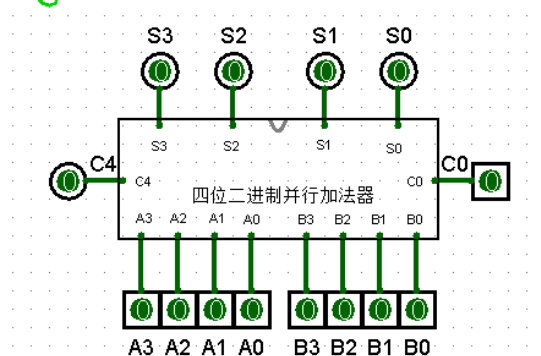


图 1-6 先行进位的四位二进制并行加法器封装图

接下来对该加法器进行测试，由于输入值得组合较多，这里选取部分输入进行测试

1. 输入 $A=0B, B=1011B, C_0=0B$ ，理论输出 $C_4=0B, S=1011B$ ，实际输出如图 1-7 所示，与理论结果一致

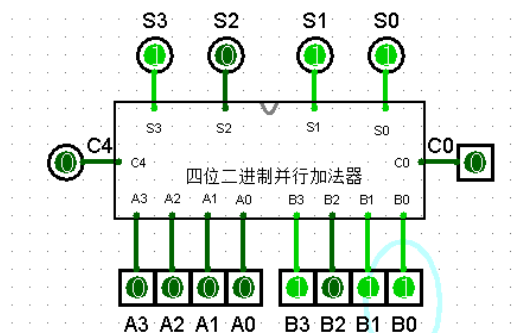


图 1-7 测试样例 1

2. 输入 $A=1111B, B=1111B, C_0=1B$ ，理论输出 $C_4=1B, S=1111B$ ，实际输出如图 1-8 所示，与理论结果一致

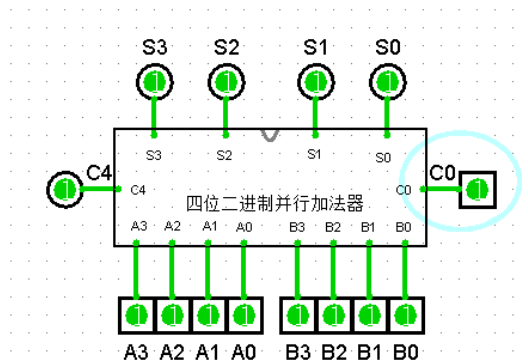


图 1-8 测试样例 2

3. 输入 $A=1100B, B=0011B, C_0=1B$ ，理论输出 $C_4=1B, S=0000B$ ，实际输出如图 1-9 所示，与理论结果一致

《数字电路与逻辑设计》实验报告

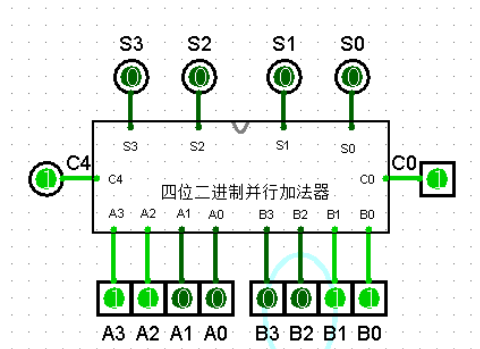


图 1-9 测试样例 3

4. 输入 $A=0100B$, $B=1001B$, $C_0=1B$, 理论输出 $C_4=0B$, $S=1110B$, 实际输出如图 1-10 所示, 与理论结果一致

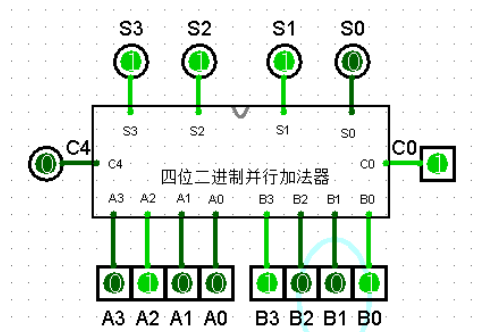


图 1-10 测试样例 4

华中科技大学

数字逻辑实验报告

小型实验室门禁系统设计实验报告

《数字电路与逻辑设计》实验报告

二、小型实验室门禁系统设计

1、实验名称

小型实验室门禁系统设计。

2、实验目的

要求同学采用传统电路的设计方法，对一个“设计场景”进行逻辑电路的设计，并利用工具软件，例如，“logisim”软件的虚拟仿真来检查这个小型实验室门禁系统的设计是否达到要求。

通过以上实验的设计、仿真、验证 3 个训练过程使同学们掌握小型电路系统的设计、仿真、调试方法以及电路模块封装的方法。

3、实验所用设备

Logisim2.7.1 软件一套。

4、实验内容

设计场景：某小型保密实验室需要安装一个门禁系统，用于监测、控制和显示该实验室内上班人数，该实验室只有一个门，最多只能容纳 15 人。假设员工进出实验室都要刷校园卡，并且保证一次刷卡后有且只有一人能进出。实验室空置时人数显示为 0，刷卡进入时实验室人数加 1，刷卡离开时实验室人数减 1。当实验室满员时，还有员工在门外刷卡进入时，门禁系统“不”动作，系统报警提示满员。

使用 logisim 软件对小型电路进行虚拟实验仿真，除逻辑门、触发器、7 段数码显示管外，不能直接使用 logisim 提供的逻辑元件库，具体要求如下。

（1）设计一个四位二进制可逆计数器电路并进行封装和验证它的正确性

用 D 触发器设计一个四位二进制可逆计数器，并进行封装。该计数器有一个清零端 CLR、一个累加计数脉冲端 CPU（输入刷卡进入请求）、一个累减计数脉冲端 CPD（输入刷卡离开请求），四个计数输出端 QDQCQBQA 记录当前实验室人数。

将设计好的 4 位二进制可逆计数器进行封装，生成一个“私有”库元件，以便后续实验使用，4 位二进制可逆计数器逻辑符号参见图 2-1 所示。

《数字电路与逻辑设计》实验报告

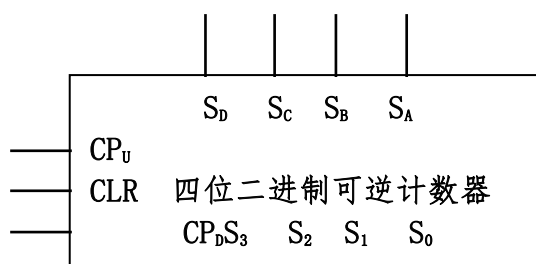


图 2-1 “私有”的一个 4 位二进制可逆计数器

(2) 用实验 1 中已封装的“先行进位的四位二进制并行加法器”设计一个将实验室内人数转换成 8421BCD 码的电路

用实验一中已封装的“先行进位的四位二进制并行加法器”和适当的逻辑门将二进制数表示的实验室内人数转换成两位十进制数的 8421BCD 码。

(3) 设计 7 段译码器，并采用“7 段数码显示管”显示人数的电路

设计一个 7 段译码器（参考书的 7448 芯片），将两位十进制数的 8421BCD 码表示的实验室内人数用“7 段数码显示管”显示出来。

该 7 段译码器有四个输入 A3A2A1A0 和七个输出 abcdefg，A3A2A1A0 为 8421BCD 码，abcdefg 为 7 段数码显示管对应的段。

(4) 设计当实验室满员时，门禁“不”动作，系统报警提示满员的电路

当实验室满员时，在累加计数脉冲端 CPU 输入刷卡进入请求，计数输出端数据保持不变，门禁“不”动作，系统报警提示满员。当实验室空时，逻辑上不会有实验室内累减计数脉冲端 CPD 输入刷卡离开请求。为防止信号干扰，在计数输出为 0 时，若 CPD 端有脉冲，也应使计数输出端数据保持不变，门禁“不”动作，但不用报警。

(5) 设计小型实验室门禁系统电路并进行封装和验证它的正确性

设计满足要求的小型实验室门禁系统电路并进行封装，生成一个小型实验室门禁系统芯片，封装后的小型实验室门禁系统逻辑符号参见图 2-2 所示。

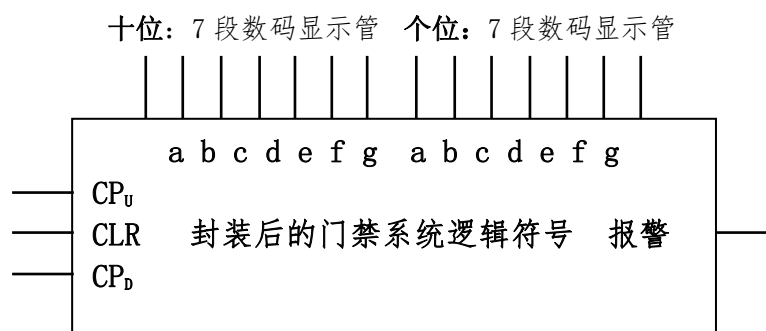


图 2-2 封装后的小型实验室门禁系统

《数字电路与逻辑设计》实验报告

5、实验方案设计

(1) 设计一个四位二进制可逆计数器电路

经分析采用异步时序逻辑电路，根据题意列出四位二进制可逆计数器的二进制状态表如表 2-1 所示。

表 2-1 四位二进制可逆计数器的二进制状态表

现态 $Q_3Q_2Q_1Q_0$	次态 $Q_3^{n+1}Q_2^{n+1}Q_1^{n+1}Q_0^{n+1}$	
	CP_U	CP_D
0000	0001	1111
0001	0010	0000
0010	0011	0001
0011	0100	0010
0100	0101	0011
0101	0110	0100
0110	0111	0101
0111	1000	0110
1000	1001	0111
1001	1010	1000
1010	1011	1001
1011	1100	1010
1100	1101	1011
1101	1110	1100
1110	1111	1101
1111	0000	1110

根据表 2-1 所示状态表，采用 D 触发器，可确定在输入脉冲作用下的状态转移关系和激励函数真值表，如表 2-2 所示。

表 2-2 状态转移关系及激励函数真值表

输入 CP_DCP_U	现态 $Q_3Q_2Q_1Q_0$	次态 $Q_3^{n+1}Q_2^{n+1}Q_1^{n+1}Q_0^{n+1}$	状态跳变				激励函数							
	$Q_3Q_2Q_1Q_0$	$Q_3^{n+1}Q_2^{n+1}Q_1^{n+1}Q_0^{n+1}$	Q_3	Q_2	Q_1	Q_0	C_3	D_3	C_2	D_2	C_1	D_1	C_0	D_0
01	0000	0001				↑		d		d		d	↓	1
	0001	0010			↑	↓		d		d	↓	1	↓	0
	0010	0011				↑		d		d		d	↓	1
	0011	0100		↑	↓	↓		d	↓	1	↓	0	↓	0
	0100	0101				↑		d		d		d	↓	1
	0101	0110			↑	↓		d		d	↓	1	↓	0
	0110	0111				↑		d		d		d	↓	1
	0111	1000	↑	↓	↓	↓	↓	1	↓	0	↓	0	↓	0
	1000	1001				↑		d		d		d	↓	1
	1001	1010			↑	↓		d		d	↓	1	↓	0
	1010	1011				↑		d		d		d	↓	1

《数字电路与逻辑设计》实验报告

10	1011	1100		↑	↓	↓		d	↓	1	↓	0	↓	0
	1100	1101				↑		d		d		d	↓	1
	1101	1110			↑	↓		d		d	↓	1	↓	0
	1110	1111				↑		d		d		d	↓	1
	1111	0000	↓	↓	↓	↓	↓	0	↓	0	↓	0	↓	0
	0000	1111	↑	↑	↑	↑	↓	1	↓	1	↓	1	↓	1
	0001	0000				↓		d		d		d	↓	0
	0010	0001			↓	↑		d		d	↓	0	↓	1
	0011	0010				↓		d		d		d	↓	0
	0100	0011		↓	↑	↑		d	↓	0	↓	1	↓	1
	0101	0100				↓		d		d		d	↓	0
	0110	0101			↓	↑		d		d	↓	0	↓	1
	0111	0110				↓		d		d		d	↓	0
	1000	0111	↓	↑	↑	↑	↓	0	↓	1	↓	1	↓	1
	1001	1000				↓		d		d		d	↓	0
	1010	1001			↓	↑		d		d	↓	0	↓	1
	1011	1010				↓		d		d		d	↓	0
	1100	1011		↓	↑	↑		d	↓	0	↓	1	↓	1
	1101	1100				↓		d		d		d	↓	0
	1110	1101			↓	↑		d		d	↓	0	↓	1
	1111	1110				↓		d		d		d	↓	0

利用卡诺图化简得到

$$D_3 = \bar{Q}_3$$

$$D_2 = \bar{Q}_2$$

$$D_1 = \bar{Q}_1$$

$$D_0 = \bar{Q}_0$$

在加计数时,CPU 有脉冲,通过观察表 6-2 可以看出 C3 所要求的触发信号'↓'可由 Q2Q1Q0 提供,C2 所要求的触发信号'↓'可由 Q1Q0 提供,C1 所要求的触发信号'↓'可由 Q0 提供,C0 所要求的触发信号'↓'可由 CPU 提供.

在减计数时,CPD 有脉冲,通过观察表 6-2 可以看出 C3 所要求的触发信号'↓'可由 $\bar{Q}_2\bar{Q}_1\bar{Q}_0$ 提供,C2 所要求的触发信号'↓'可由 $\bar{Q}_1\bar{Q}_0$ 提供,C1 所要求的触发信号'↓'可由 \bar{Q}_0 提供,C0 所要求的触发信号'↓'可由 CPU 提供.

综合加计数和减计数,可知

$$C_0 = CP_U + CP_D$$

$$C_1 = Q_0CP_U + \bar{Q}_0CP_D$$

$$C_2 = Q_1Q_0CP_U + \bar{Q}_1\bar{Q}_0CP_D$$

$$C_3 = Q_2Q_1Q_0CP_U + \bar{Q}_2\bar{Q}_1\bar{Q}_0CP_D$$

使用 Logism 做出电路图如图 2-1 所示:

《数字电路与逻辑设计》实验报告

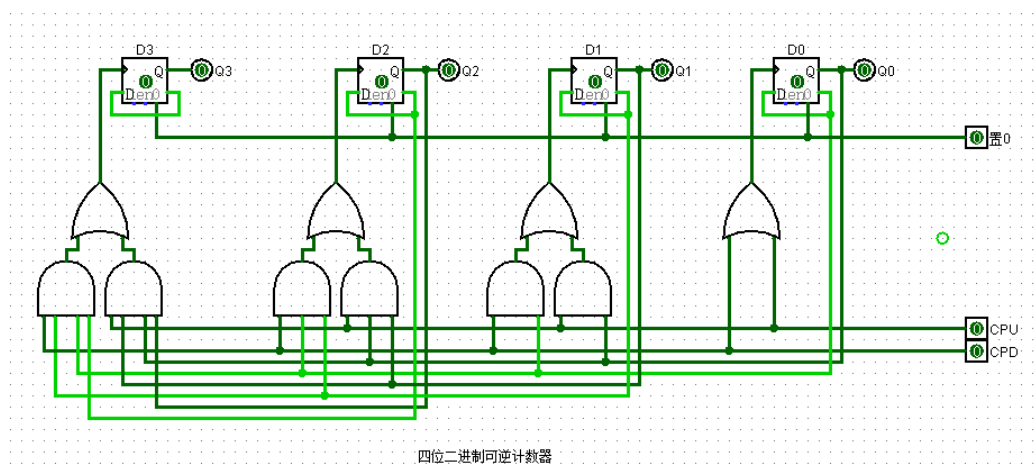


图 2-1 一个四位二进制可逆计数器

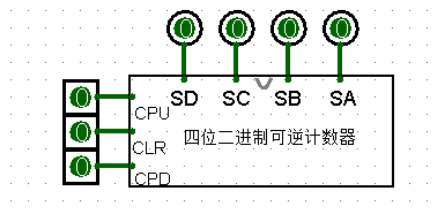


图 2-2 四位二进制可逆计数器封装图

(2) 用实验一中已封装的“先行进位的四位二进制并行加法器”设计将实验室内人数转换成 8421BCD 码的电路

由题意可列出四位二进制数和 8421 码的对应关系，如表 2-3 所示

表 2-3 四位二进制数与 8421 码对应表

十进制数	输入(4 位二进制数)				输出(8421 码)					修正控制
N	A ₃	A ₂	A ₁	A ₀	C ₄	S ₈	S ₄	S ₂	S ₁	Z
0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	0	1	0
2	0	0	1	0	0	0	0	1	0	0
3	0	0	1	1	0	0	0	1	1	0
4	0	1	0	0	0	0	1	0	0	0
5	0	1	0	1	0	0	1	0	1	0
6	0	1	1	0	0	0	1	1	0	0
7	0	1	1	1	0	0	1	1	1	0
8	1	0	0	0	0	1	0	0	0	0
9	1	0	0	1	0	1	0	0	1	0
10	1	0	1	0	1	0	0	0	0	1
11	1	0	1	1	1	0	0	0	1	1

《数字电路与逻辑设计》实验报告

12	1	1	0	0	1	0	0	1	0	1
13	1	1	0	1	1	0	0	1	1	1
14	1	1	1	0	1	0	1	0	0	1
15	1	1	1	1	1	0	1	0	1	1

利用卡诺图化简得到修正控制

$$Z = A_3A_2 + A_3A_1 = A_3(A_2 + A_1)$$

修正方式为 S3S2S1S0+0ZZ0

据此可以画出四位二进制数转 8421 码的电路图如图 2-3 和图 2-4 所示：

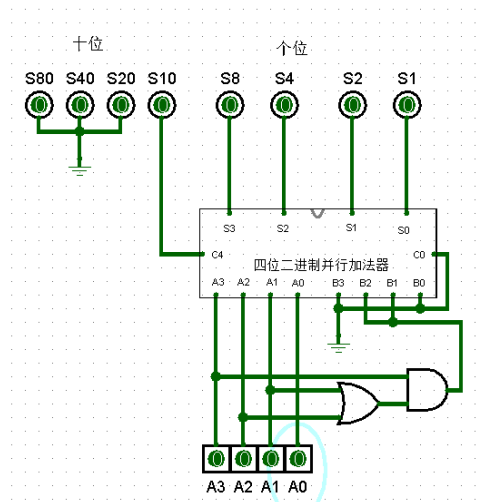


图 2-3 四位二进制数转 8421 码电路图

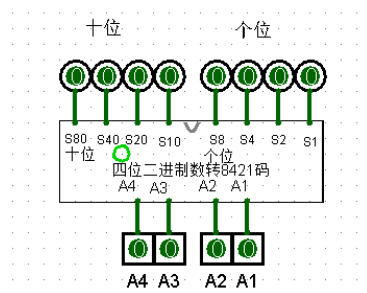


图 2-4 四位二进制数转 8421 码封装图

(3) 设计 7 段译码器，并采用“7 段数码显示管”显示人数的电路

(A) 设计一个 7 段译码器

7 段译码器的真值表如表 2-4 所示：

表 2-4 7 段译码器真值表

输入				输出						
A ₃	A ₂	A ₁	A ₀	a	b	c	d	e	f	g

《数字电路与逻辑设计》实验报告

0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	1	1	0	0	0	0
0	0	1	0	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	0	0	1
0	1	0	0	0	1	1	0	0	1	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	0	0	0	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	0	0	1	1
1	0	1	0	0	0	0	1	1	0	1
1	0	1	1	0	0	1	1	0	0	1
1	1	0	0	0	1	0	0	0	1	1
1	1	0	1	1	0	0	1	0	1	1
1	1	1	0	0	0	0	1	1	1	1
1	1	1	1	0	0	0	0	0	0	0

利用卡诺图化简得：

$$a = \overline{A_2}\overline{A_1}\overline{A_0} + \overline{A_3}\overline{A_2}A_1 + \overline{A_3}A_2A_0 + A_3\overline{A_1}A_0$$

$$b = \overline{A_3}\overline{A_2} + \overline{A_1}\overline{A_0} + \overline{A_2}\overline{A_1} + \overline{A_3}A_1A_0$$

$$c = \overline{A_3}A_2 + \overline{A_2}\overline{A_1} + \overline{A_2}A_0$$

$$d = A_2\overline{A_1}A_0 + \overline{A_2}\overline{A_0} + \overline{A_2}A_1A_0 + A_2A_1\overline{A_0}$$

$$e = A_1\overline{A_0} + \overline{A_2}\overline{A_0}$$

$$f = \overline{A_1}\overline{A_0} + A_2\overline{A_1} + A_3\overline{A_2}\overline{A_1} + A_2\overline{A_0}$$

$$g = A_2\overline{A_1} + A_3\overline{A_2} + \overline{A_2}A_1 + A_1\overline{A_0}$$

使用 logisim 做出电路图，如图 2-5 所示

《数字电路与逻辑设计》实验报告

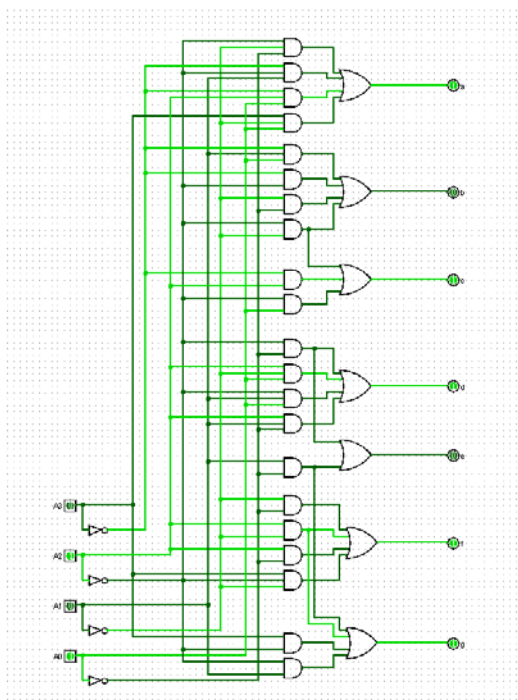


图 2-5 7 段译码器

(B) 设计用“7 段数码显示管”显示人数的逻辑电路

将四位二进制数转 8421 码的十位输出和个位输出分别连接到两个 7 段译码器上，然后将 7 段译码器连接到数码显示管得到显示实验室人数的电路，电路的输入是二进制数表示的实验室人数

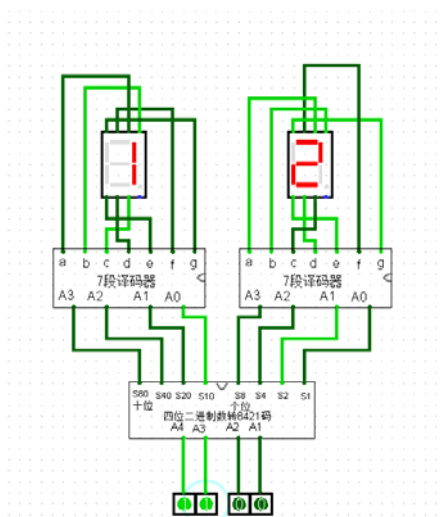


图 2-6 用 7 段数码显示管显示实验室人数的电路

(4) 设计当实验室满员时，门禁“不”动作，系统报警提示满员的电路

《数字电路与逻辑设计》实验报告

当实验室满员时计数器输出均是高电平，将 4 个输出端与后得到满员状态的信号。当实验室无人时计数器的输出均是低电平，将 4 个输出端或后得到无人状态信号。利用 $A \cdot 0 = 0$, $A + 0 = A$, 设满员信号为 M, 无人信号为 N, 有:

$$M = S_D S_C S_B S_A$$

$$N = S_D + S_C + S_B + S_A$$

将 U 和 \bar{M} 与后接入芯片的 CPU, 则当未满员时, U 的脉冲可传递到芯片的 CPU, 当满员时, U 来脉冲时和 0 与, 结果为 0, 脉冲未传递到芯片的 CPU。

将 D 和 N 与后接入芯片的 CPD, 则当有人时, U 的脉冲可传递到芯片的 CPD, 当无人时, D 来脉冲时和 0 与, 结果为 0, 脉冲未传递到芯片的 CPD。

据此写出 CPU 和 CPD 的表达式

$$CPU = U \bar{M}$$

$$CPD = DN$$

当满员信号为 1 且 U 来了脉冲, 进行报警, 报警信号 A 的表达式为

$$A = UM$$

使用 logism 作图, 结果如图 2-7 所示

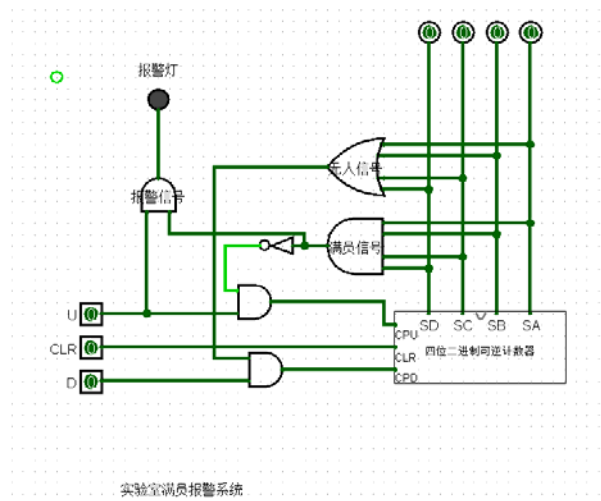


图 2-7 报警电路

(5) 设计小型实验室门禁系统电路

将报警系统四位二进制数转 8421 码的芯片, 然后通过 7 段译码器连接数码管, 得到电路如图 2-8 所示。

《数字电路与逻辑设计》实验报告

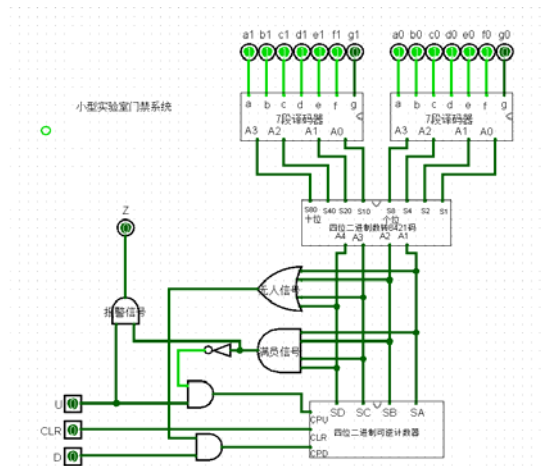


图 2-8 小型实验室门禁系统电路

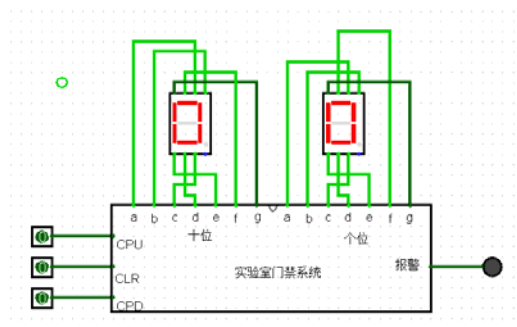


图 2-9 小型实验室门禁系统封装电路

6、实验结果记录

(1) 给出“私有”库元件（采用一个四位二进制可逆计数器进行封装）的测试电路

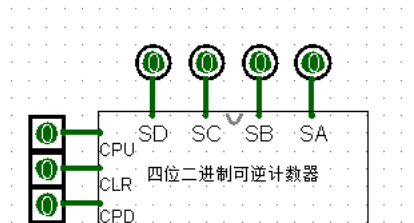


图 2-10 一个四位二进制可逆计数器

《数字电路与逻辑设计》实验报告

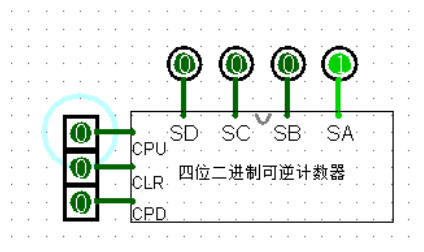


图 2-11 来一个 CPU 脉冲

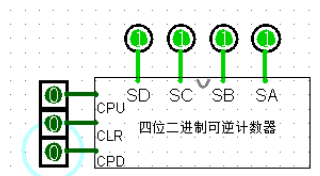


图 2-12 来一个 CPD 脉冲

(2) 给出采用实验 1 中已封装的“先行进位的四位二进制并行加法器”设计的将实验室内人数转换成 8421BCD 码的电路

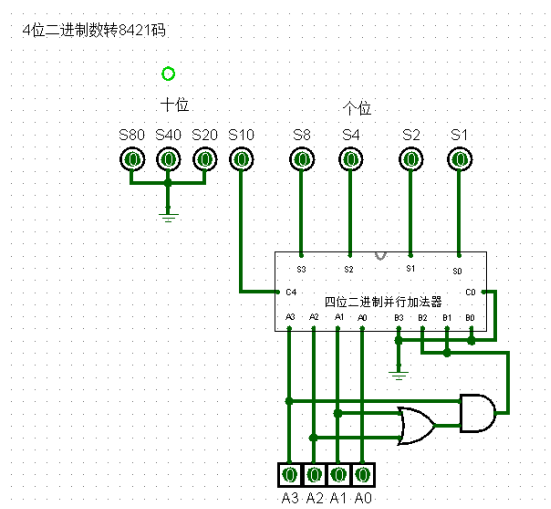


图 2-13 二进制数转换成 8421BCD 码的电路

(3) 给出采用“7 段数码显示管”显示人数的电路

《数字电路与逻辑设计》实验报告

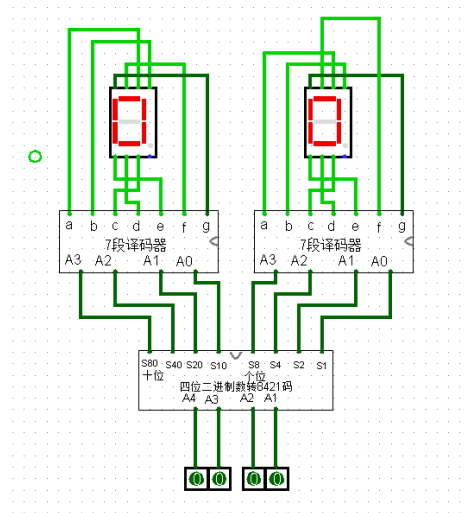


图 2-14 人数显示的电路

(4) 给出当实验室满员时，门禁不动作，系统报警提示满员的电路
当实验室满员时，U 来一个脉冲报警灯会亮。

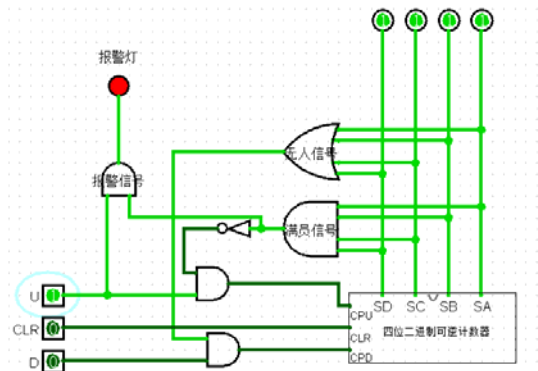


图 2-15 系统满员报警电路

当实验室无人时，D 来一个脉冲计数器无变化

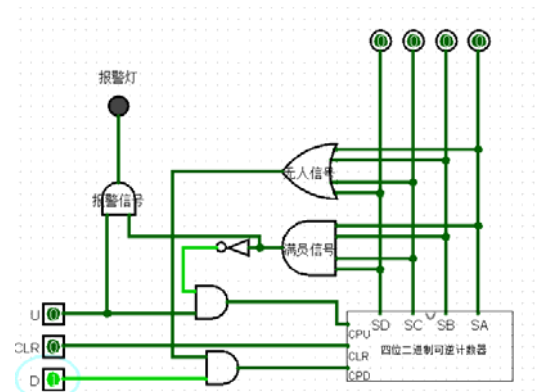


图 2-16 系统无人报警电路

(5) 给出“私有”库元件（采用小型实验室门禁系统电路进行封装）的测

《数字电路与逻辑设计》实验报告

试电路

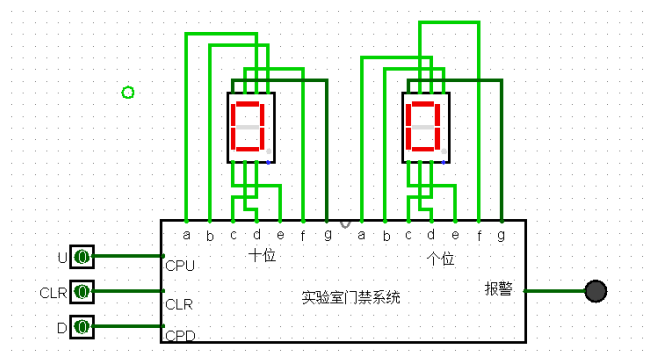


图 2-16 实验室门禁系统电路

CPU 每来一个脉冲，数码管显示的数字就会加一，CPD 每来一个脉冲，数码管显示的数字就会减一。

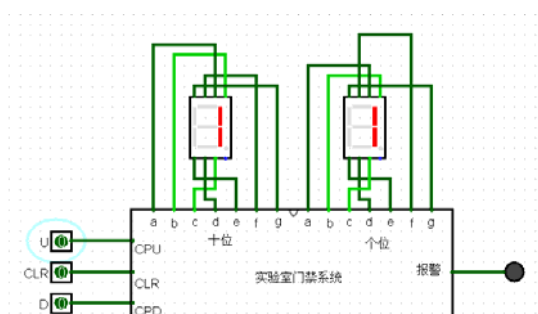


图 2-17 来了 11 个 CPU 脉冲后的电路

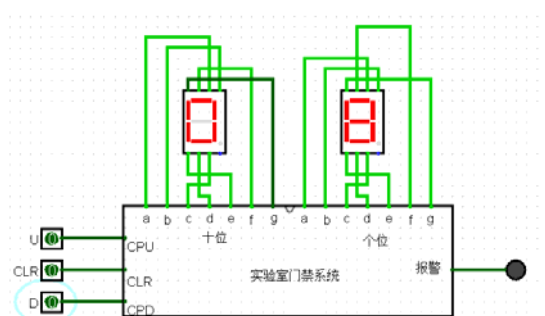


图 2-18 又来了 3 个 CPD 脉冲后的电路

7、实验后的思考

(1) 这两次实验的难点你认为在哪些方面？

第一次实验初次使用 logism，对许多操作还不是很熟练，在使用 logism 作图时花费了许多时间。第二次实验首先是计数器的设计比较困难，第二就是七段译码器的线路过于庞杂，出现了连线错误的问题。

(2) 你是如何解决的？

《数字电路与逻辑设计》实验报告

第一次实验多次练习，熟能生巧。第二次实验通过与同学讨论，参考课本获得解决问题的大致思路，然后做出真值表，然后分析变量间的关系来求出表达式，最后利用 logism 作图并调试实现计数器。7 段译码器的难点在于连线，所以我采用每连接一个输出端口就使用 logism 自带的分析电路功能比对真值表是否正确，若正确才连接下一个输出端口，一步一步来。

(3) 意见和建议

实验 2 的内容量多于实验 1，希望能合理分配实验量，将实验 2 的任务分配给实验 1 一些。