

实验报告

课 程		姓 名		学 号	
指导教师		专业班级		成 绩	

实验三 存储器实验

一、实验目的

1. 掌握存储器的组成结构和工作原理。
2. 掌握静态随机存储器 RAM 工作特性及数据的读写方法。
3. 掌握存储器扩展的基本原理和方法。

二、实验设备

PC 机一台，Logisim 实验系统一套。

三、实验原理及内容

计算机在处理汉字输出的时候，需用到汉字的字形码。字形码是用点阵表示的汉字字型代码，是汉字的输出形式。汉字字库用于输出汉字字形码，字形码点阵信息量大，占用存储空间大，16*16 点阵需 256 位才能显示一个汉字，如图 3-1 所示。

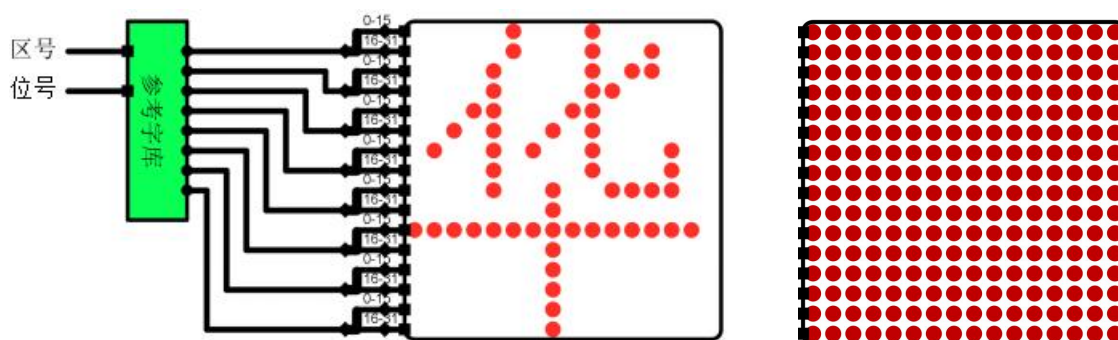


图 3-1 16*16 字形码点阵

国标 GB2312 将汉字以行列矩阵形式排列，分为 94 行，94 列，分别用 7 位区号表示行号，7 位位号表示列号，就构成了汉字的区位码，图 3-2 为部分汉字区位码。

区位码	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
16区	啊	阿	埃	挨	哎	唉	哀	皑	癌	蔼	矮	艾	碍	爱	隘	鞍	氨	安	俺	按
17区	薄	雹	保	堡	饱	宝	抱	报	暴	豹	鲍	爆	杯	碑	悲	卑	北	辈	背	贝
18区	病	并	玻	菠	播	拨	钵	波	博	勃	搏	铂	箔	伯	帛	舶	脖	膊	渤	泊
19区	场	尝	常	长	偿	肠	厂	敞	畅	唱	倡	超	抄	钞	朝	嘲	潮	巢	吵	炒
20区	础	储	矗	搐	触	处	揣	川	穿	椽	传	船	喘	串	疮	窗	幢	床	闯	创

图 3-2 汉字区位码

汉字字库通过区号、位号进行检索，输出汉字的字形码。地址输入为：区号（7 位）、位号（7 位），逻辑地址为：（区号-1）*94+位号-1，区号、位号从 1 开始。数据输出为：256 位点阵信息。

现有 8 片 16K*32 位 ROM，用于在 Logisim 平台构建 GB2312 汉字编码的 16K*256bit 点阵汉字字库，电路输入为汉字区号和位号，电路输出为 8*32 位（16*16=256 位点阵信息），如图 3-3 所示。

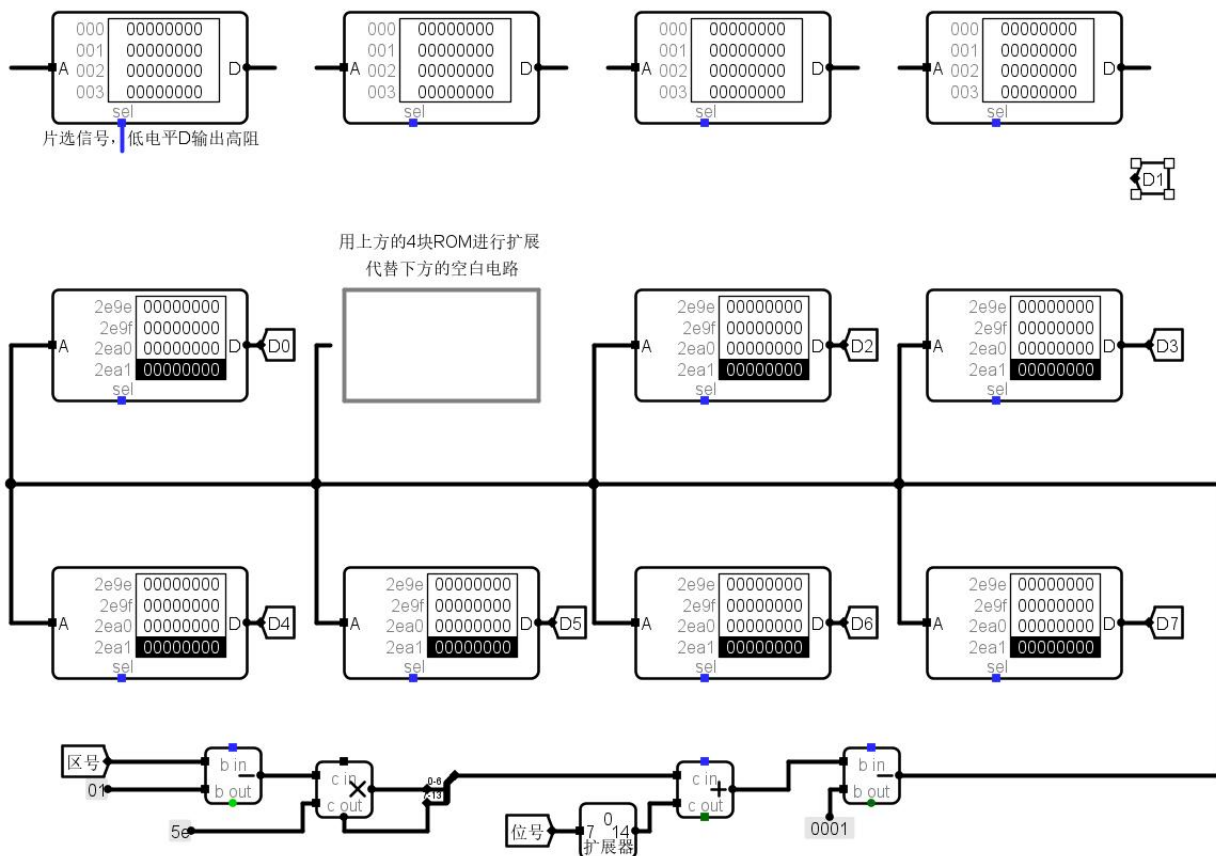


图 3-3 汉字字库

图中有一块空白的 16K*32 位电路，需要用 4 片 4K*32 位 ROM 替换。请设计该空白电路的电路连接图。

1. 设计思想

用 4 片 4K×32 位 ROM 按照字扩展的扩展方式实现 16K×32 位 ROM。通过计算得出，实现上图电路，共需要 16K×32 位/4K×32 位=4 片 ROM 芯片。

地址线连接，4K×32 位 ROM 中地址线是 12 位，16K×32 位 ROM 中地址线的位数是 14 位。要用分线器分出来两个接口。将 4 片 4K×32 位 ROM 的地址(A)端并联，将并联后的地址段连接在分线器的 0-11 位对应的分线端口，分线器 12-13 两位对应的分线端口作为选能端连接译码器和多路选择器的选择端。

Selection: Splitter	
Facing	North
Fan Out	2
Bit Width In	14
Appearance	Left-handed
Bit 0	1 (Left)
Bit 1	1 (Left)
Bit 2	1 (Left)
Bit 3	1 (Left)
Bit 4	1 (Left)
Bit 5	1 (Left)
Bit 6	1 (Left)
Bit 7	1 (Left)
Bit 8	1 (Left)
Bit 9	1 (Left)
Bit 10	1 (Left)
Bit 11	1 (Left)
Bit 12	0 (Right)
Bit 13	0 (Right)

图 3-4 分线器数据图

数据线连接，选择多路选择器，数据为 32 位，但输出 LED 矩阵位为 16×16，该数据线应该有 16 位。多路选择器的输入端分别与四片 ROM 的数据（D）端连接，选择端连接分线器 12-13 两位对应的分线端口，输出端为 D1。

片选信号连接，选择译码器，连接原理为：将整个存储系统地址线的低位用于译码器的输入端，而译码器的输出端用于连接四个芯片各自合适的片选信号线。在页面中增加一个译码器，译码器的选择端连接分线器 12-13 两位对应的分线端口，输出端分别与四个 ROM 的 sel 端连接。

2.电路连接图

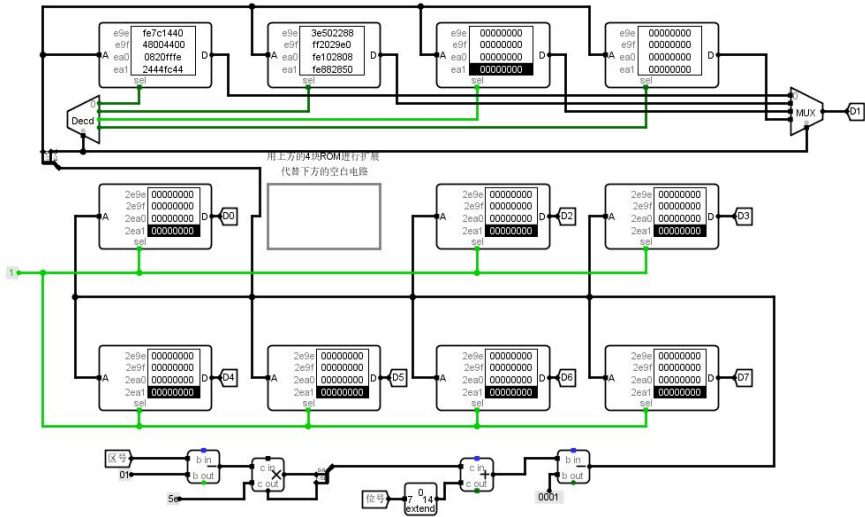


图 3-5 存储器扩展电路图

3.拆分字库文件到 4 片 4K*32 位 ROM

完成空白电路设计后，在 8 片 16K*32 位 ROM 中载入字库文件（空白电路的字库文件 HZK16_1 需拆分成 4 个字库文件，载入 4 片 4K*32 位 ROM 中）。

每片 16K×32 位 的 ROM 的内部各自存储 16K 个地址，每个地址里存放 4 个字节数据。地址范围都一样：0x0000~0x3FFF。每片 4K×32 位的 ROM，地址范围都一样：0x000~0xFFFF，每个共有 4K 个地址，现在将 16K×32 位的 ROM 中的数据按照顺序，每 4K 个为一组分为四组，分别放到 4 片 4K×32 位 的 ROM 中去。将 HZK16_1.txt 中的 1~4096 个数据放到 0 号 4K 的 ROM 中，4097~ 8192 个数据放到 1 号 4K 的 ROM 中，8193~12288 个数据放到 2 号 4K 的 ROM 中，12289~16384 个数据放到 3 号 4K 的 ROM 中。

4.实验结果分析

打开 logisim 中的字库测试电路，设置时钟频率为 8Hz，按 Ctrl+k 进行电路仿真，将仿真结果与参考字库输出结果进行对比。

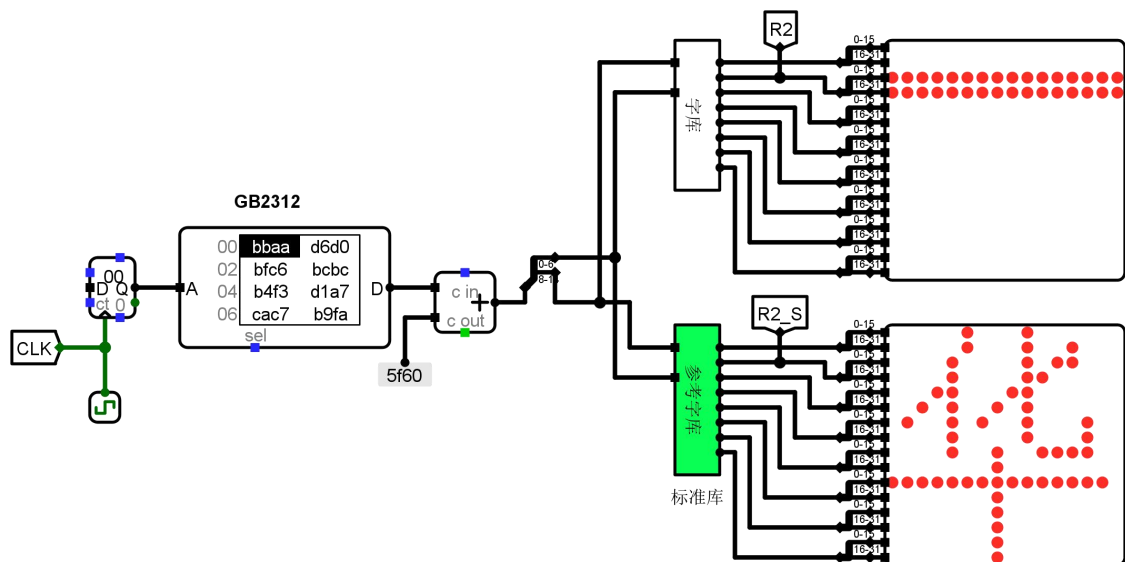


图 3-6 字库测试电路

刚开始电路仿真时，结果错误，LED 灯亮红，检查后，发现拆分字库有问题。四个 4K×32 位的 ROM 当中，数据并不是平分的，而是以 4K 为一组分别放在四组不同的芯片当中，需注意地址区间的划分。数据导入到 ROM 中，顺序错误，连接未一一对应，导致存储扩展的输出引脚显示不了部分字形阵，需按固定顺序从左到右地址增加。修改后结果正确。

logisim 运行的实验结果如下（截取部分图）：

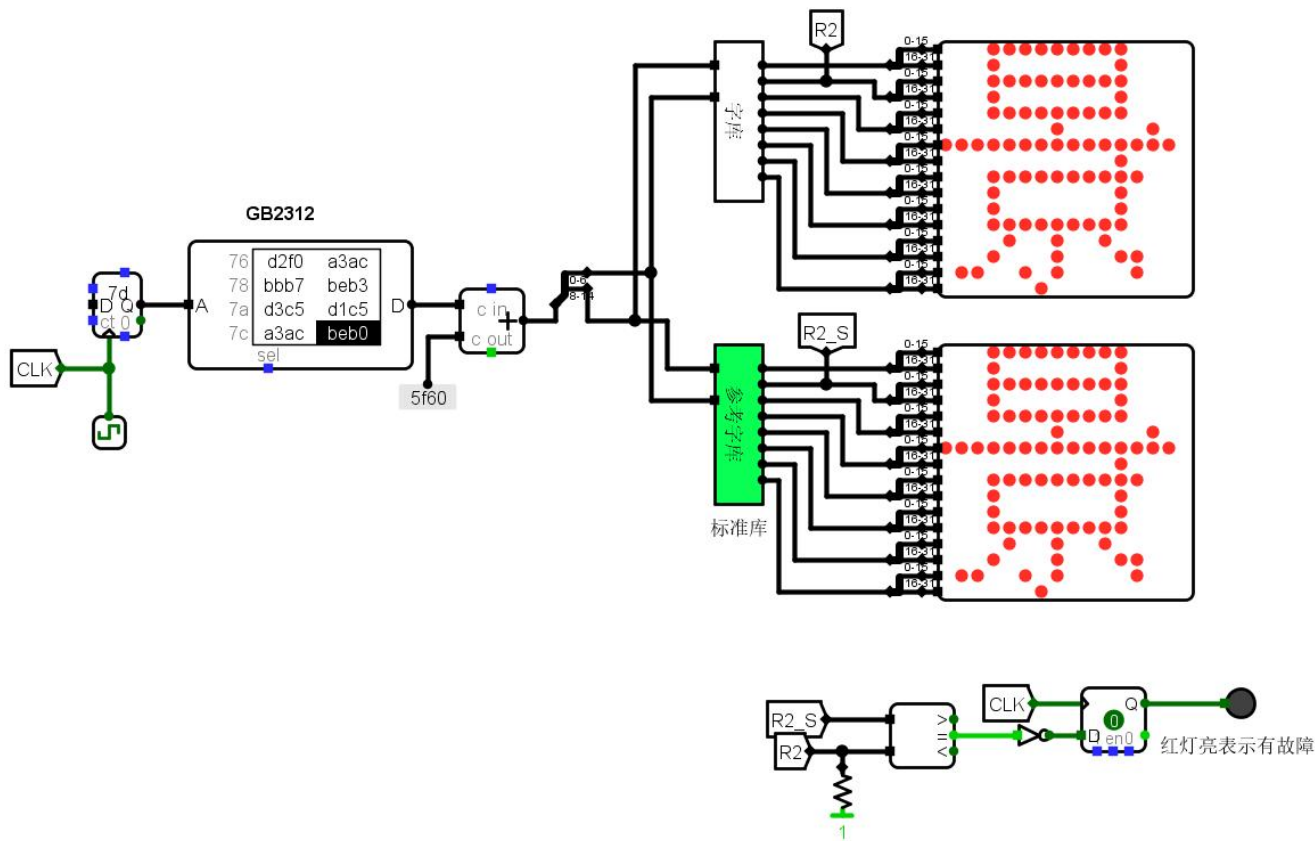


图 3-7 字库测试电路结果 1

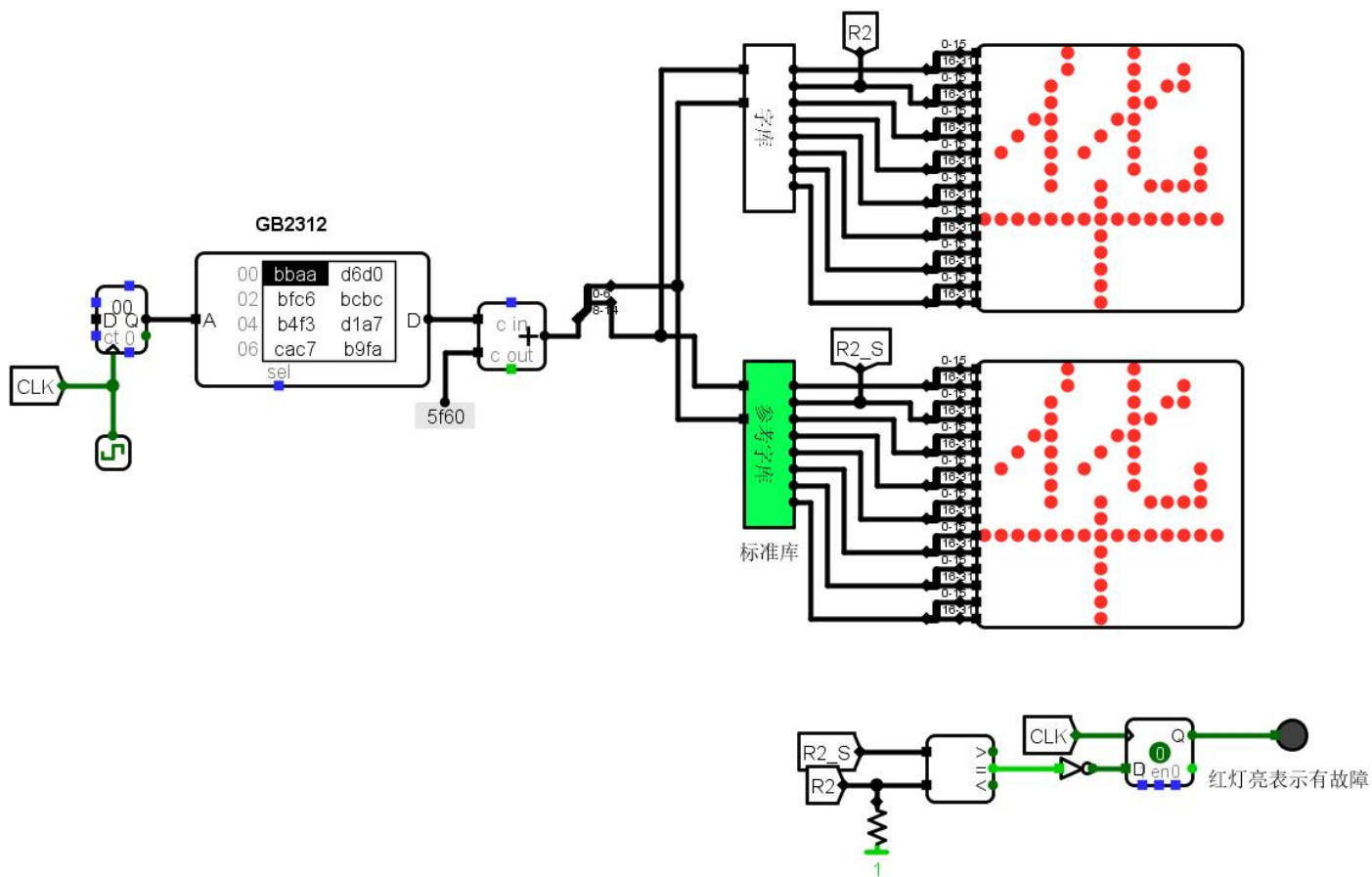


图 3-8 字库测试电路结果 2

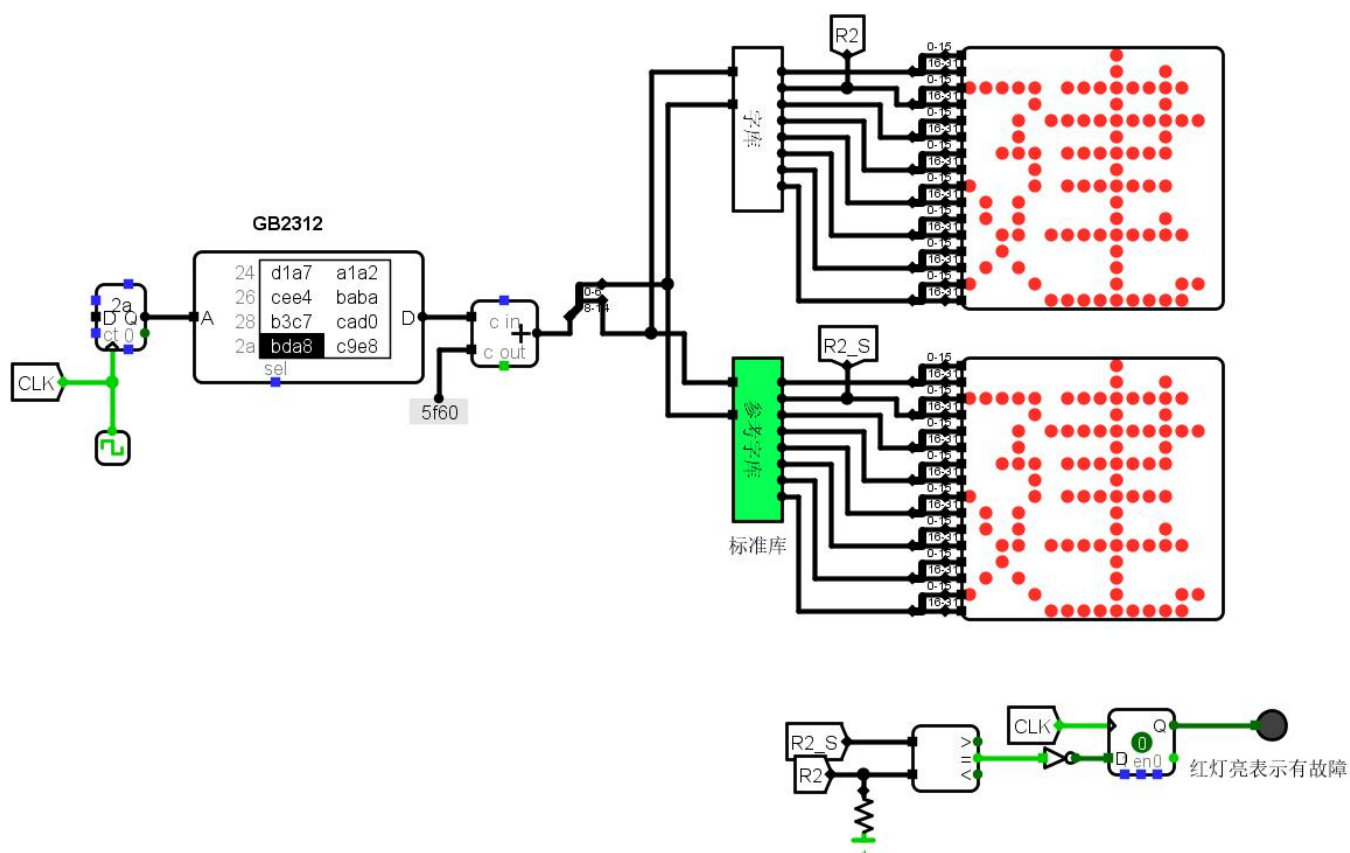


图 3-9 字库测试电路结果 3

四、思考题

1. 某计算机的主存地址空间中，0x0000 到 0x3FFF 为 ROM 存储区域，0x4000 到 0x5FFF 为保留地址区域，0x6000 到 0x7FFF 为 RAM 地址区域。ROM 的控制信号为 CS* 和 OE*，RAM 的控制信号为 CS*、OE* 和 WE*，CPU 地址线 A15~A0，数据线 D7~D0，读控制线为 RD*，写控制线为 WR*。若 ROM 采用 8K×8 的芯片，RAM 芯片采用 4K×4 的芯片，试分析每个芯片的地址范围，画出与总线的连接图。

ROM: 0x0000 - 0x3FFF 8k×8 芯片
 RAM: 0x6000 - 0x7FFF 4k×4 芯片

CPU 地址线 A15-A0 数据线 D7-D0

ROM: $0x3FFFH - 0x0000H + 1 = 4000H = 2^{14}$

芯片数: $\frac{2^{14} \times 8}{8k \times 8} = 2$

RAM: $0x7FFFH - 0x6000H + 1 = 2000H = 2^{13}$

芯片数: $\frac{2^{13} \times 8}{4k \times 4} = 4$

图 4-1 芯片个数

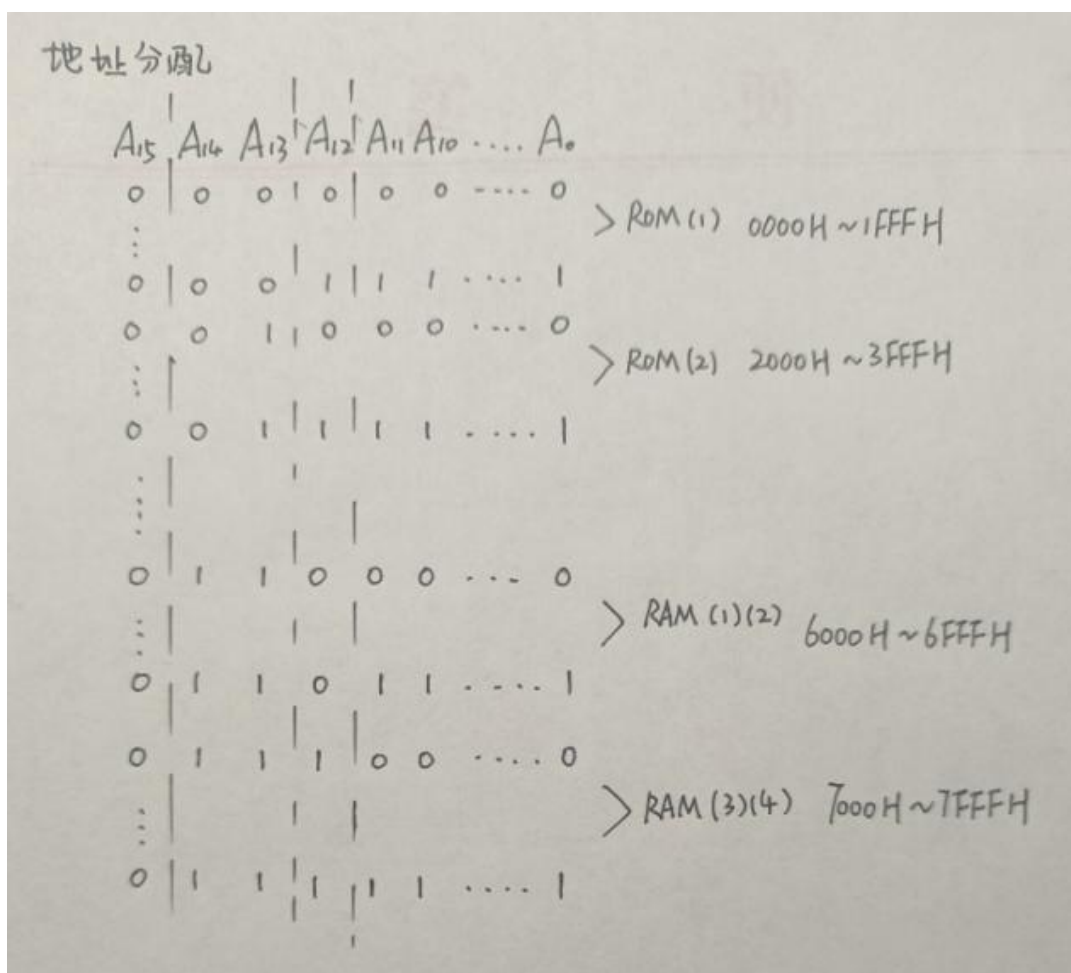


图 4-2 地址分配

片选逻辑，将 A13, A14, A15 作为 74LS138 译码器的输入端，选择 / Y0、/ Y1、/ Y3 作为片选信号。/ Y0、/ Y1 分别为 ROM (1) (2) 的片选信号 / CS0 和 / CS1。根据图 4-2 地址分配可以看出 A12 为 0 时，RAM (1) (2) ，将 / Y3 和 A12 相或后作为其片选信号 / CS2；A12 为 1 时，RAM (3) (4) ，将 / Y3 和非 A12 相或后作为其片选信号 / CS3。

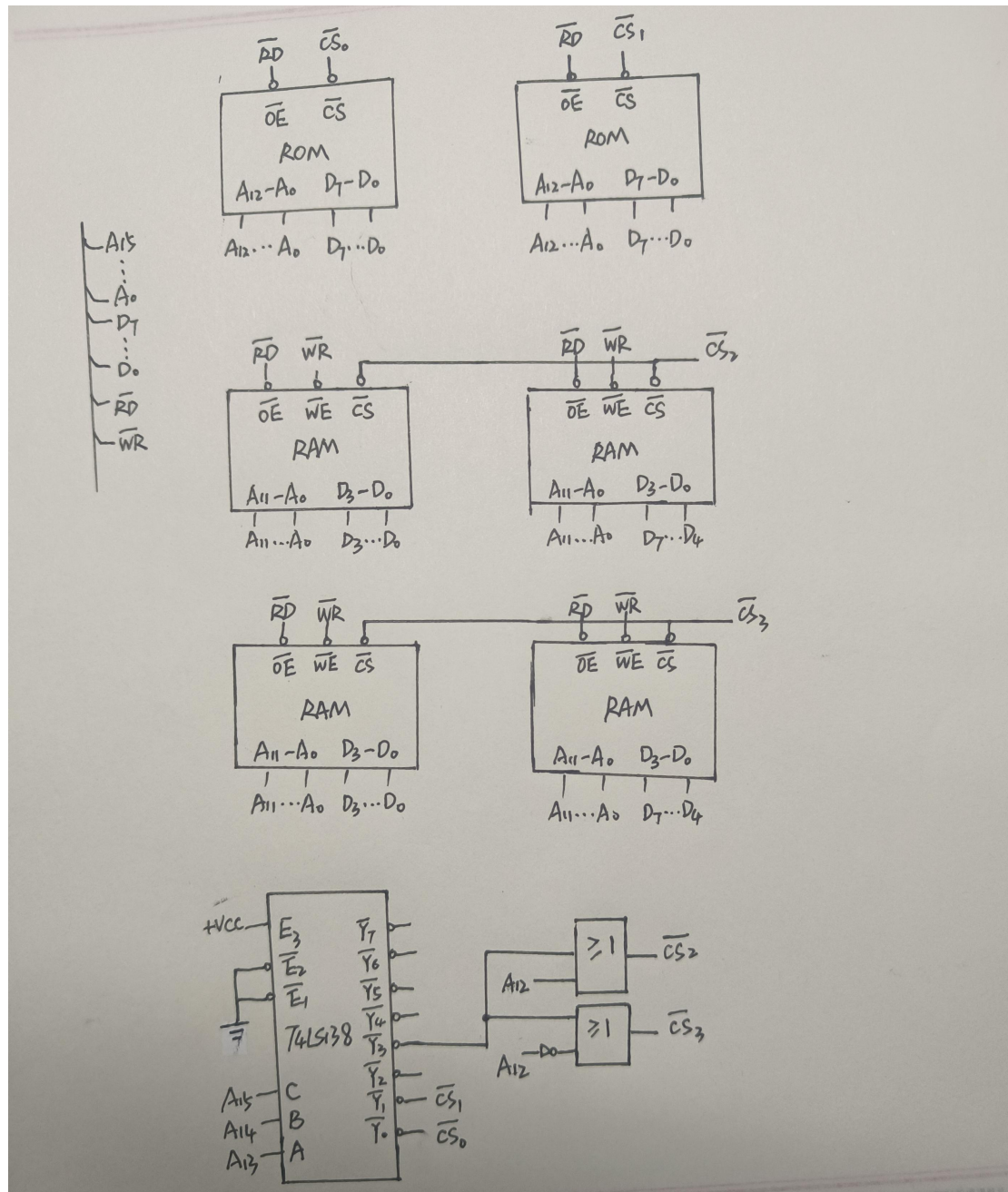


图 4-3 连接图

五、实验心得体会

