

实验 3：同步时序电路设计

一、实验目的

1. 掌握时序逻辑电路设计的基本方法。
2. 掌握计数器和移位寄存器的构建方法。
3. 掌握数字时钟和乘法器的设计方法。
4. 掌握寄存器堆的设计方法。

二、实验环境

Logisim 2.16

三、实验内容

1、计数器实验

根据表 3.1 给出的功能表和图 3.1 所示电路原理图构建 4 位同步二进制计数器 CNTR4U 子电路。

实验步骤如下：

- 1) 构建 4 位同步二进制计数器子电路。打开 Logisim 软件，在工作区中合理布局逻辑门电路、D 触发器，输入和输出引脚，并设置输入引脚的极性，连接各线路，设置相应属性。

表 3.1 4 位同步二进制计数器功能表

Inputs				Current State				Next State			
CLR	LD	ENT	ENP	Q3	Q2	Q1	Q0	Q3*	Q2*	Q1*	Q0*
1	x	x	x	x	x	x	x	0	0	0	0
0	1	x	x	x	x	x	x	D3	D2	D1	D0
0	0	0	x	x	x	x	x	Q3	Q2	Q1	Q0
0	0	x	0	x	x	x	x	Q3	Q2	Q1	Q0
0	0	1	1	0	0	0	0	0	0	0	1
0	0	1	1	0	0	0	1	0	0	1	0
				...							
0	0	1	1	1	1	0	1	1	1	1	0
0	0	1	1	1	1	1	0	1	1	1	1
0	0	1	1	1	1	1	1	0	0	0	0

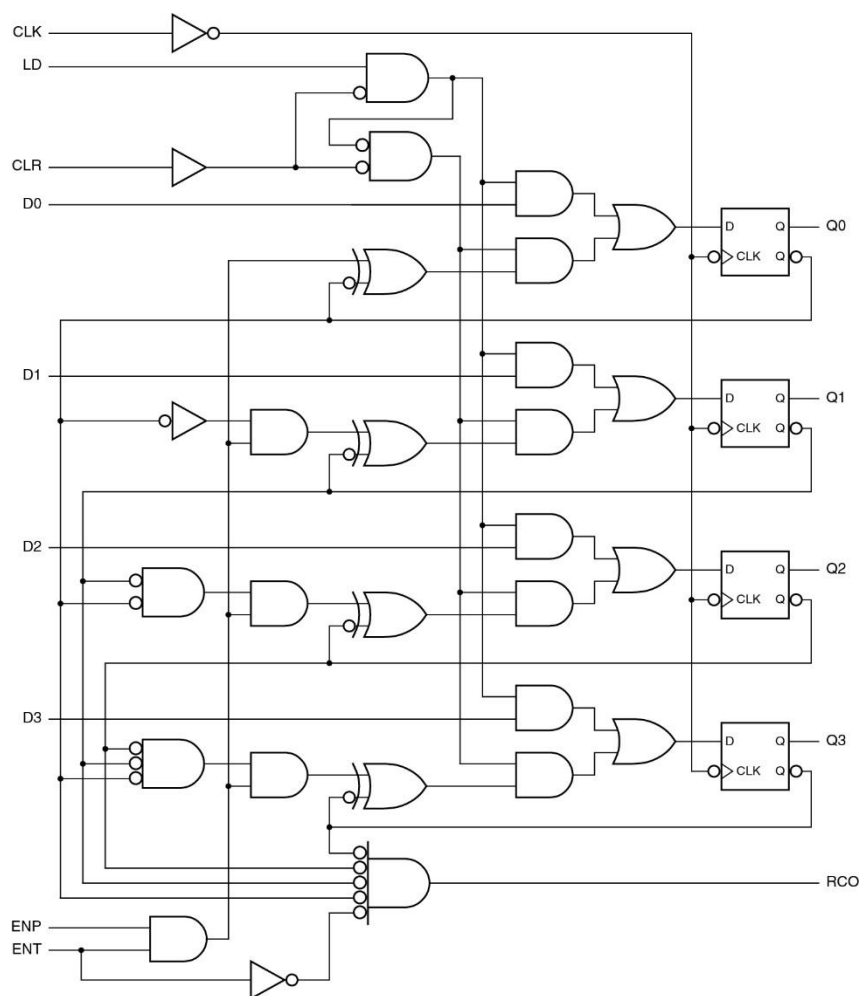
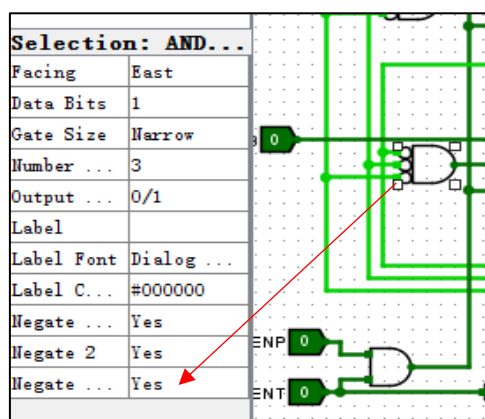
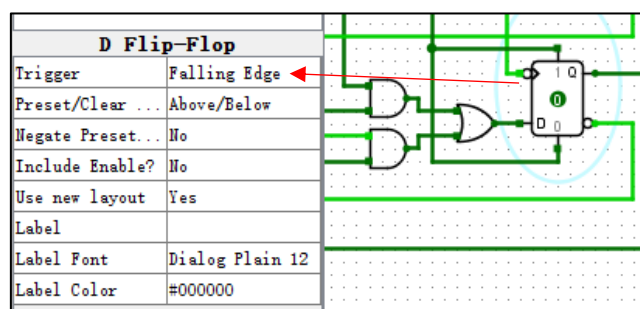


图 3.1 4 位同步二进制计数器原理图

2) 可设置组件的输入端口属性为负逻辑，则在图形符号中输入端就带有反相圈，如图 3.2(a)所示；可设计 D 触发器的输入时钟触发属性设置为下降沿触发，则在图形符号的时钟输入端口带有反相圈，如图 3.2(b)所示。最终将实现电路，测试功能，并验证实验结果。保存电路，文件名为 lab3，修改导航区中的电路名称“main”为“CNTR4U”。



(a) 设置输入端口为反相



(b) 设置 D 触发器的时钟信号为下降沿触发

图 3.2 设置部件的相应属性

将设计电路封装成子电路，并进行输入输出信号标注。这里需要注意封装的时序电路为子电路时，没有时钟信号的输入端口，在仿真时，采用共同的时钟信号。如果需要在封装时显式地显示时钟输入端口，可在子电路设计中将时钟信号 CLK 组件改为输入引脚组件而不用时钟组件，如图 3.3 所示。

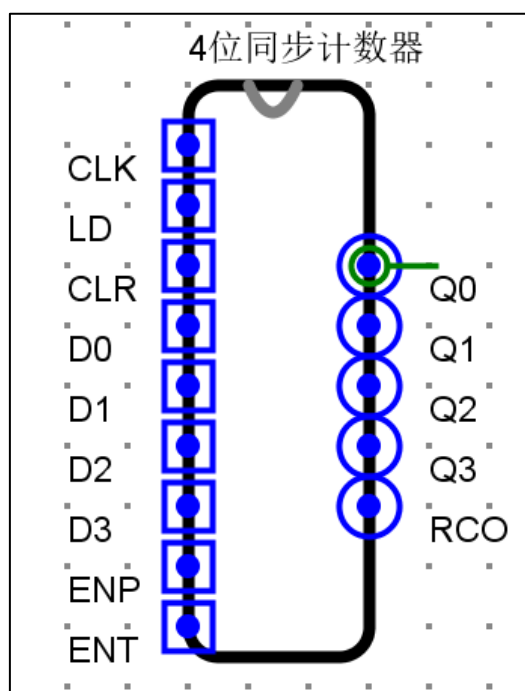


图 3.3 4 位同步二进制计数器封装图

2、数字时钟实验（验收）

要求：显示时分秒 6 个 BCD 码数字，到 23:59:59 后清零，同时蜂鸣 1 秒，到整点时点亮 LED 灯 1 秒，可设置 BCD 码格式的初始时间。

在 Logisim 导航区中。添加一个名为“数字时钟”的子电路，其组件布局如图 3.4 所示。根据要求实现数字时钟，并测试功能。

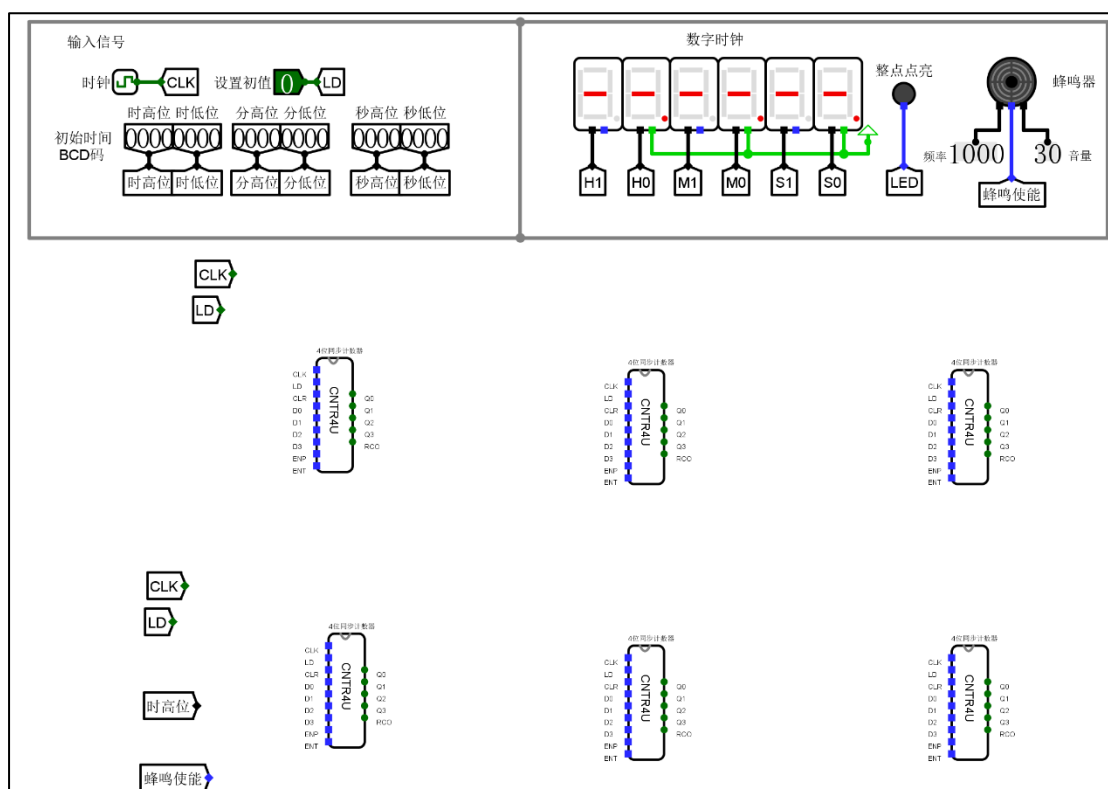


图 3.4 数字时钟布局图

3、移位寄存器实验

根据表 3.2 给出的功能描述和图 3.5 给出的电路原理图，构建 4 位通用移位寄存器 SHRG4U 子电路，该移位寄存器带有异步复位（清 0）信号 CLR，它是低电平有效信号，当它为低电平时，所有 D 触发器的状态输出为 0。

表 3.2 4 位移位寄存器功能表

功能	输入			下一个状态			
	CLR	S1	S0	Q3*	Q2*	Q1*	Q0*
清零	0	x	x	0	0	0	0
保持	1	0	0	Q3	Q2	Q1	Q0
右移	1	1	0	RIN	Q3	Q2	Q1
左移	1	0	1	Q2	Q1	Q0	LIN
装载	1	1	1	D3	D2	D1	D0

实验步骤如下：

1) 构建 4 位移位寄存器子电路。打开 Logisim 软件，在导航区添加名为“SHRG4U”的子电路，在工作区中按照图 3.5 给出的原理图，合理布局逻辑门电路、D 触发器，输入和输出引脚，并设置输入引脚的极性，连接各线路，设置相应属性。

2) 实现电路，测试功能，并验证实验结果。保存电路。

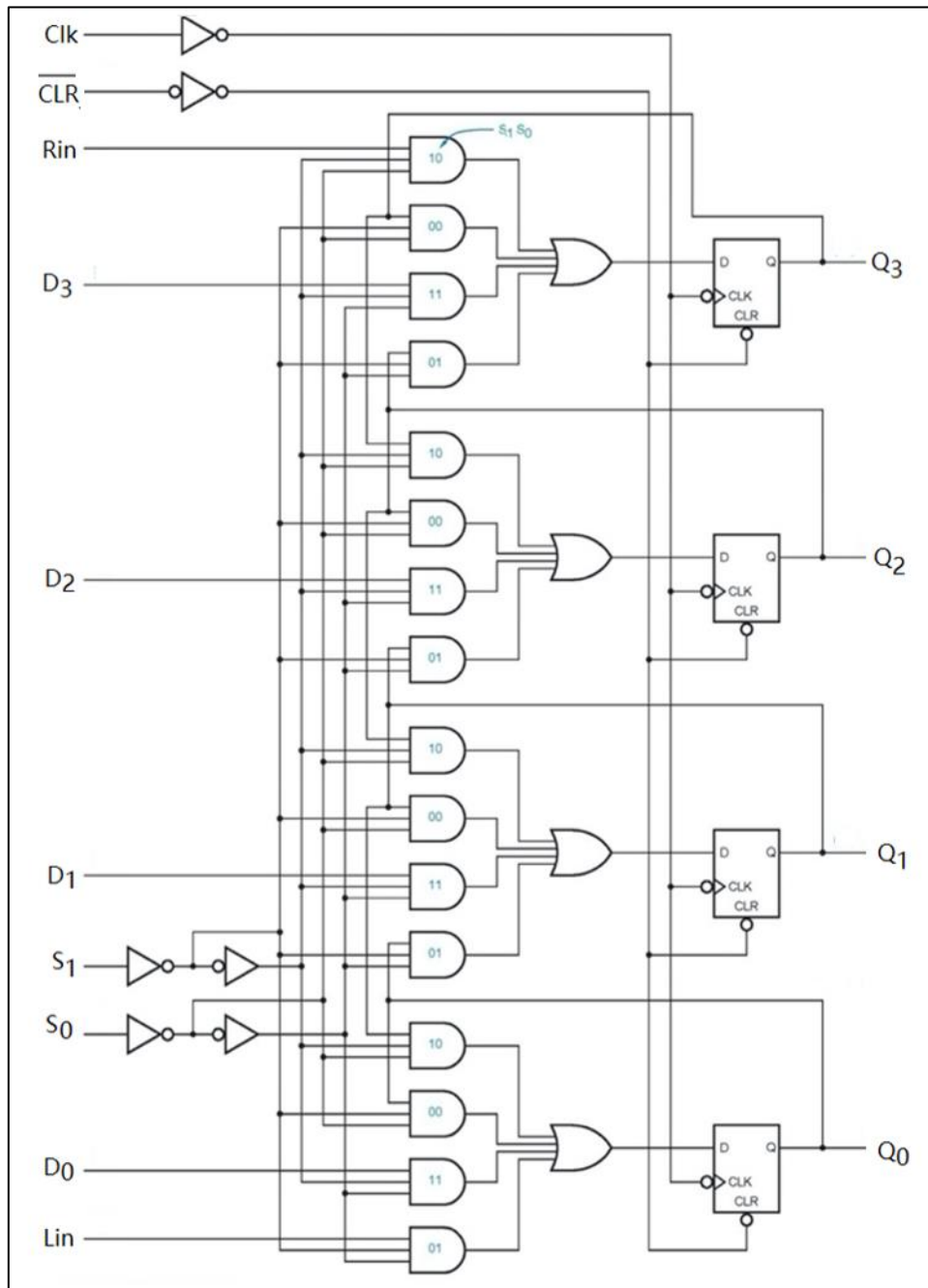


图 3.5 4 位移位寄存器原理图

3) 将设计电路封装成子电路, 并进行输入输出信号标注, 如图 3.6 所示。

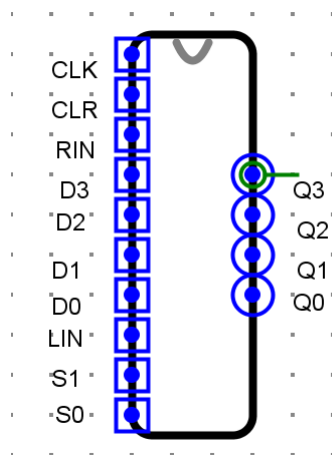


图 3.6 4 位同步二进制计数器封装图

4) 利用该 SHRG4U 子电路和少量门电路设计一个二进制序列生成器，要求设置初始值为 $D_3D_2D_1D_0=1000$ 。若采用右移模式，且反馈方程为 $RIN=Q_0 \oplus Q_1$ 。将 Q_0 、 Q_1 、CLK 等信号连接到数字示波器组件输入引脚，设置数字示波器的属性，如设置输入引脚数为 4，状态数为 20。使用时钟单步 (Ctrl+T) 方式观察输出波形，记录输出序列。

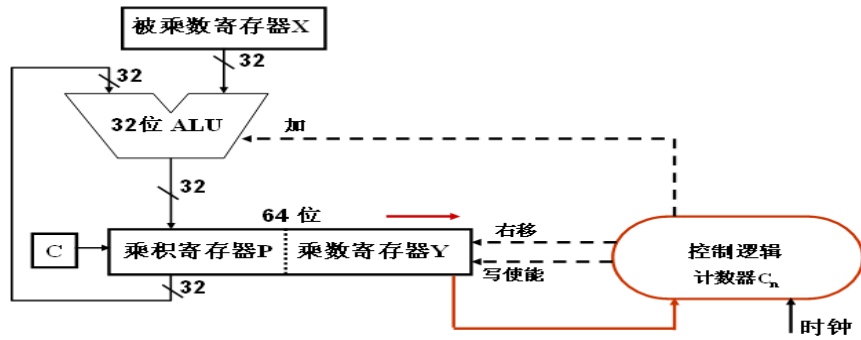
4、4 位无符号数乘法器（验收）

实验将实现两个四位二进制无符号数相乘的功能，并通过数码管将其转换成十六进制显示出来。

计算机中两个无符号数相乘，类似手算乘法。主要步骤如下。

- ① 每次将乘数 Y 的一位乘以被乘数得 $X \times Y_i$ 后，就将该结果与前面所得的结果累加，得到 P_i ，称之为部分积。
- ② 在每次求得 $X \times Y_i$ 后，不是将它左移与前次部分积 P_i 相加，而是将部分积 P_i 右移一位与 $X \times Y_i$ 相加。
- ③ 对乘数中为 1 的位执行加法和右移运算，对为 0 的位只执行右移运算，而不需执行加法运算。

其逻辑结构图如图 3.7 所示。



实验步骤如下:

- 1) 构建 4 位无符号数乘法器子电路。打开 Logisim 软件，在导航区添加名为“4 位无符号数乘法器”的子电路，在工作区中按照图 3.8 给出的原理图，合理布局 4 位寄存器、计数器、8 位桶形移位器、4 位加法器、2 路选择器等组件，并设置相应属性。
- 2) 为了便于测试，利用计数器和比较器来实现选择装载乘数信号 LOAD，和结束移位信号 STOP，以及复位信号 RST。
- 3) 实现电路，测试功能，验证并记录实验结果。保存电路。

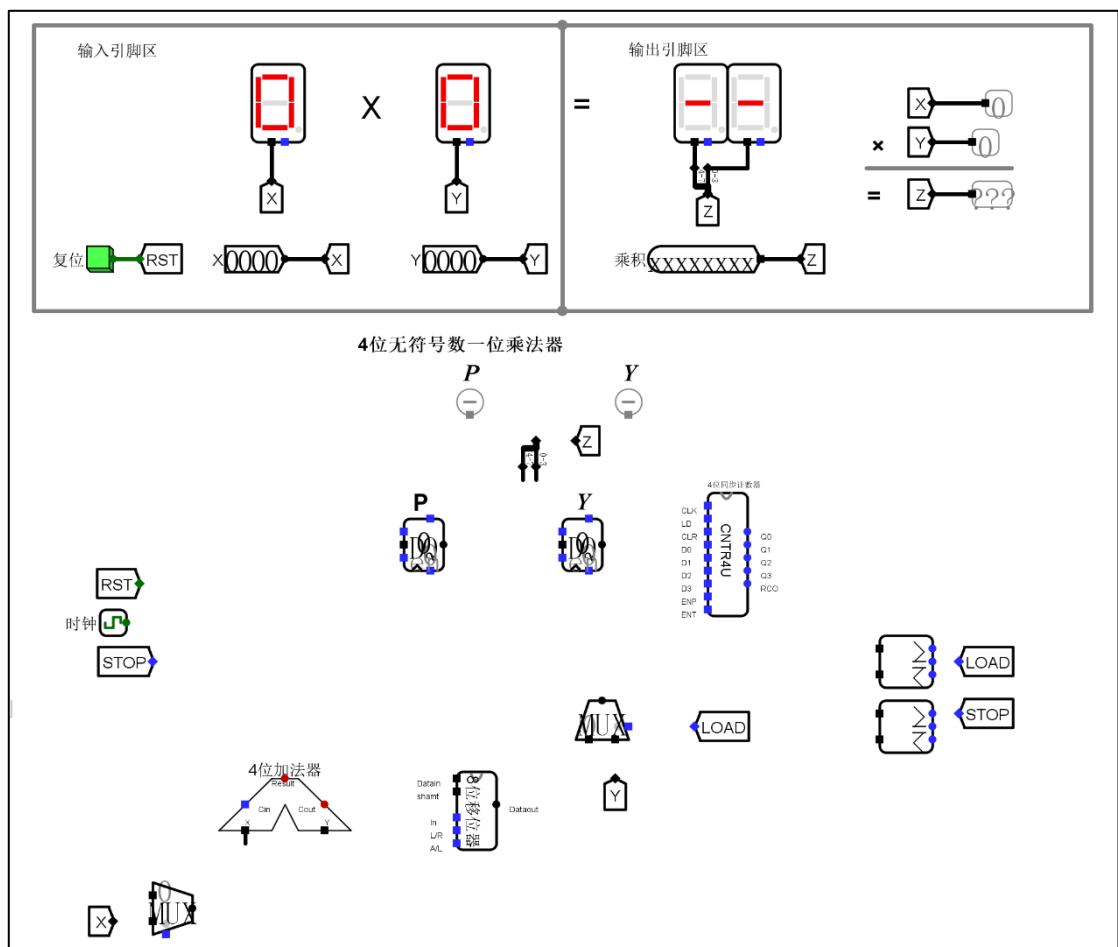


图 3.8 4 位无符号数乘法运算的组件布局图

4) 提示:需要修改实验2中的8位桶形移位器,增加移入位输入引脚。可以使用 Logisim 提供的加法器、计数器以及集线器来实现。

5、寄存器堆实验（验收）

根据图3.9中寄存器堆原理图，构建含有32个32位寄存器的寄存器堆Regfile的读写电路，包含两个读数据端口和一个写数据端口，并封装成子电路。寄存器堆的读操作属于组合逻辑操作，无须时钟控制，即当寄存器地址信号RA或RB到达后，经过一个“读取时间”的延迟，读出的数据输出到端口busA或busB上。寄存器堆的写操作则属于时序逻辑操作，需要时钟信号的控制，即在写使能信号（WE）有效的情况下，有效时钟触发边沿到来时开始将端口busW上的信息写入RW所指定的寄存器中。

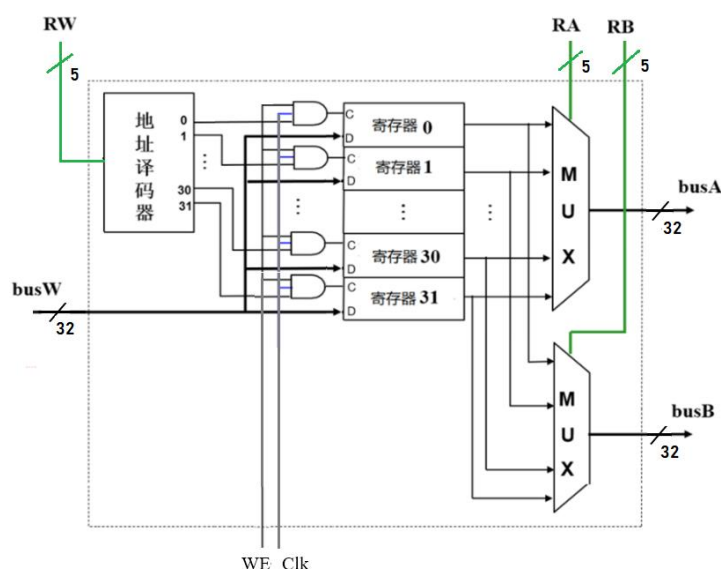


图 3.9 寄存器堆设计原理图

实验步骤如下。

1) 创建寄存器堆子电路。在工程中添加一个名为“regfiles”的子电路，并双击该子电路名称，在右侧工作区中构建相应电路。为了能在后续实验中直接引用该寄存器堆模块，在工作区中按照图 3.10 给出的引脚图进行设计，在实验时不要改变引脚和隧道的名称。

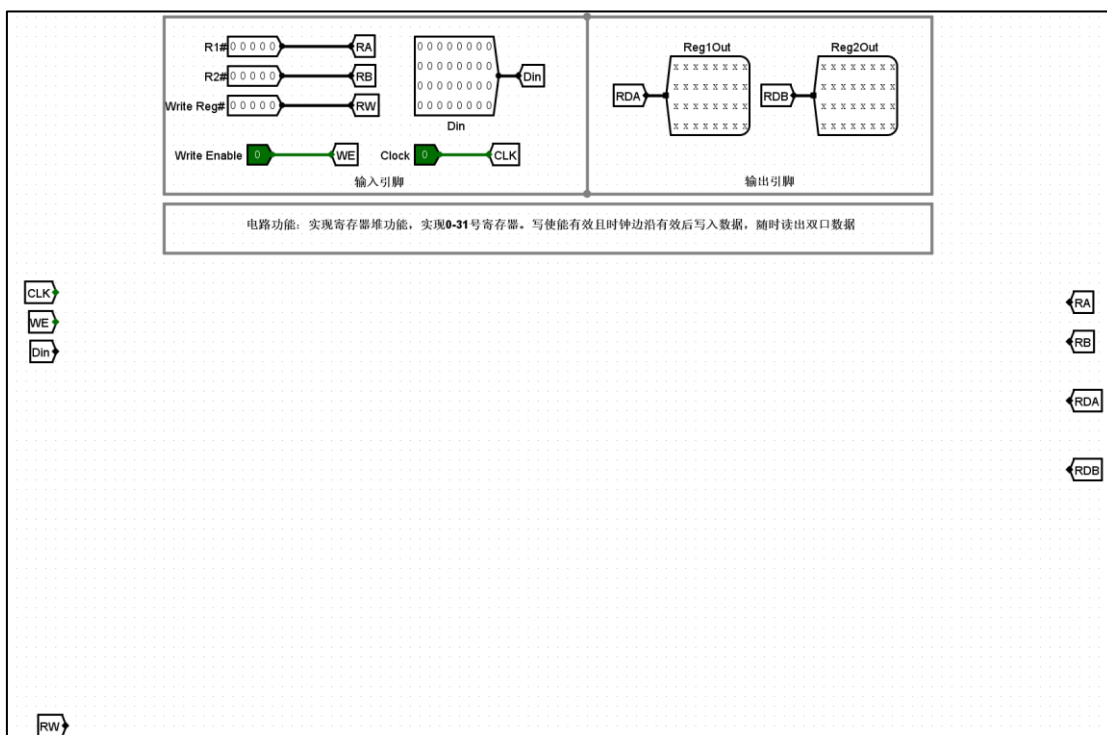


图 3.10 寄存器堆设计引脚图

在工作区放置 32 个 32 位寄存器、译码器 32 路多路选择器以及读写地址、数据端口、使能信号等输入输出端口隧道，连接相关部件，设置组件属性。

2) 在工程中添加一个名为“寄存器堆读写”的子电路，并双击该子电路名称，在右侧工作区中构建相应电路。如图 3.11 所示，在工作区中放置 regfiles 子电路（图中 RegFile）以及地址端口、数据端口、时钟和使能部件等。

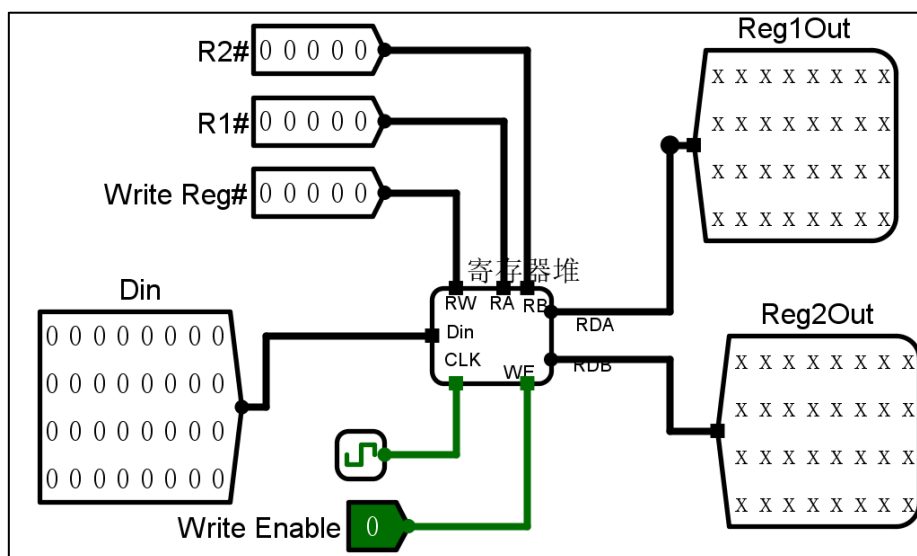


图 3.11 寄存器堆读写电路测试

3) 仿真测试。选择时钟单步（Ctrl+T）方式进行仿真，将寄存器触发边沿设置为上

升沿或下降沿来测试电路的读写功能。随机写入不同编号的寄存器，再读出验证。保存电路文件，记录实验过程。

四、思考题

1. 在数字时钟实验中，如果需要增加闹钟功能，电路中需要如何修改？
2. 如何修改 4 位无符号数乘法器电路，使其实现 4 位 Booth 乘法器的功能？
3. 如何实现 4 位无符号数除法器？