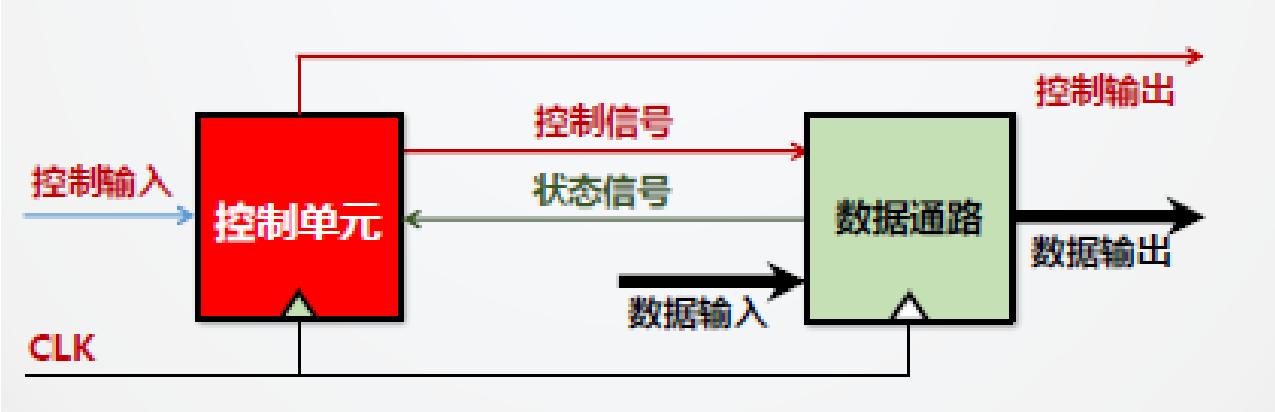
数字逻辑基础实验

——实验整体框架

实验目的

- 掌握数字系统的特点
- 掌握数字系统的设计方法：模块化、层次化
- 掌握组合逻辑的设计流程与设计思想
- 掌握同步时序电路的设计流程与设计思想
- 掌握系统集成联调的方法



数字系统的设计流程

1、数字系统的需求分析

- 外部整体输入、输出分析，包括数据、控制信号与状态

2、功能部件设计（分模块设计）

- 得到内部的控制信号、状态

3、构建数据通路

- 从数据流动的角度，连接各功能部件

4、构建控制单元

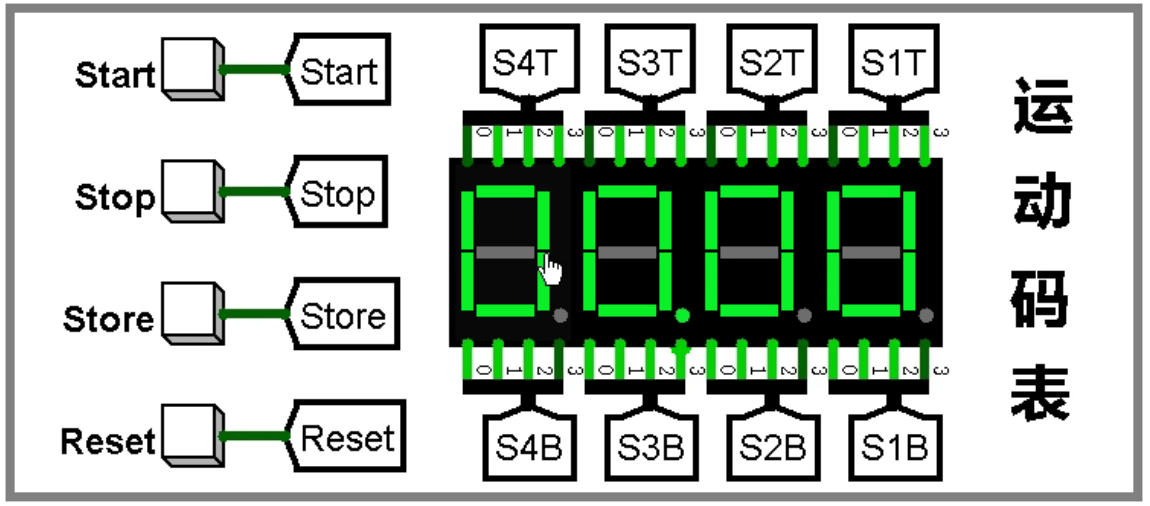
- 绘制系统状态图
- 构建状态转换电路
- 构建输出函数电路（生成所需要的内部、外部控制信号，产生状态输出）
- 构建控制单元

5、系统集成联调

- 从控制流角度，连接控制单元（4）和各执行部件构成的数据通路（3）

实验终极目标

构建小型数字系统——运动码表



◘ 运动码表初步功能需求分析

运动码表的外部特性

- 输入：4个按钮； 输出： 4个7段数码管

运动码表的功能

- **Start**：计时器归0，重新开始计时
- **Stop**：停止计时，显示当前计时数据
- **Store**：尝试更新系统记录的计时数据（满足条件：当前计时数据 < 系统记录）
- **Reset**：复位，使得计时器 = 00.00， 系统记录 = 99.99

运动码表功能部件分析

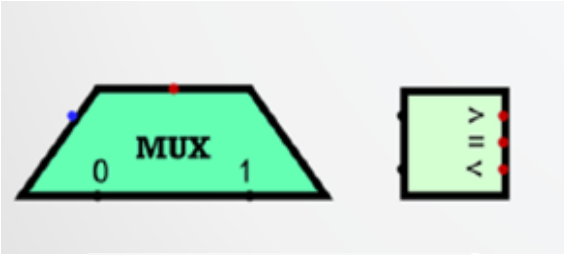
#	功能部件	控制信号	输入	输出
1	时间计数器TM	TM-En,TM-Rst	CLK	时间计数输出16位
2	16位寄存器SD	SD-En	CLK，Din(16位)	Q(16位)
3	数码管显示DP		Din(16位)	DisplayInfo(32位)
4	比较器			
5	2路选择器	Sel		

同步时序电路设计

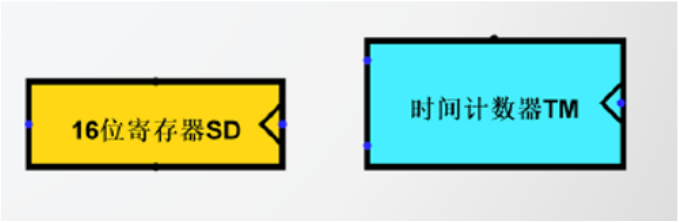
组合逻辑电路设计

实验任务分解

组合逻辑电路设计



同步时序逻辑设计



数字系统综合设计

- 运动码表数据通路建设
- 运动码表控制单元设计
- 系统集成联调

实验报告要求——运动码表设计

1、撰写实验报告。报告按照数字系统设计的流程进行组织编写；需将多个分解实验的分析和设计思路与过程清晰描述，并相应贴出测试截图。

2、实验完成后，将最终的“Logisim.circ”文件以“学号_数逻.circ”命名另存。

3、在福大课程中心平台中，本门课程的“作业”中，相应提交实验报告与电路文件。

数字逻辑基础实验（1）

——组合逻辑电路设计

一、组合逻辑设计

实验目标

- 理解组合逻辑设计的基本流程
- 掌握组合逻辑电路的设计思想—— 模块分层与迭代设计
- 熟练运用Logisim构建运动码表功能部件

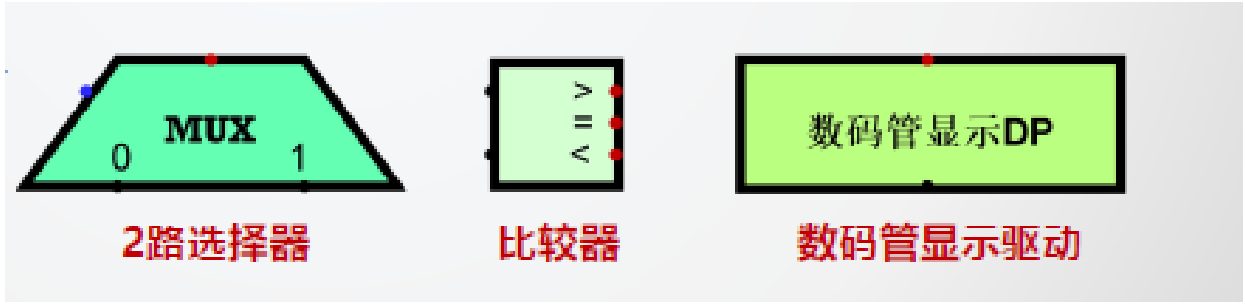
根据需求分析出逻辑表达式

实验任务

1位的2路选择、4位无符号比较器为基础任务

- 2路选择器设计
- 16位无符号比较器设计
- 码表数码管显示驱动设计

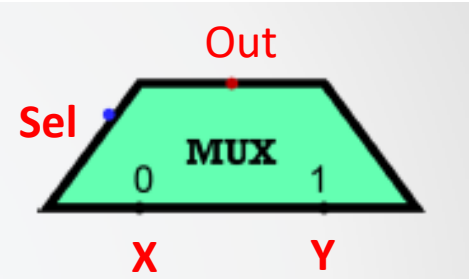
已封装外观



打开资源包中的“Logisim. circ”， 可以看到已经建立的电路（待编辑）

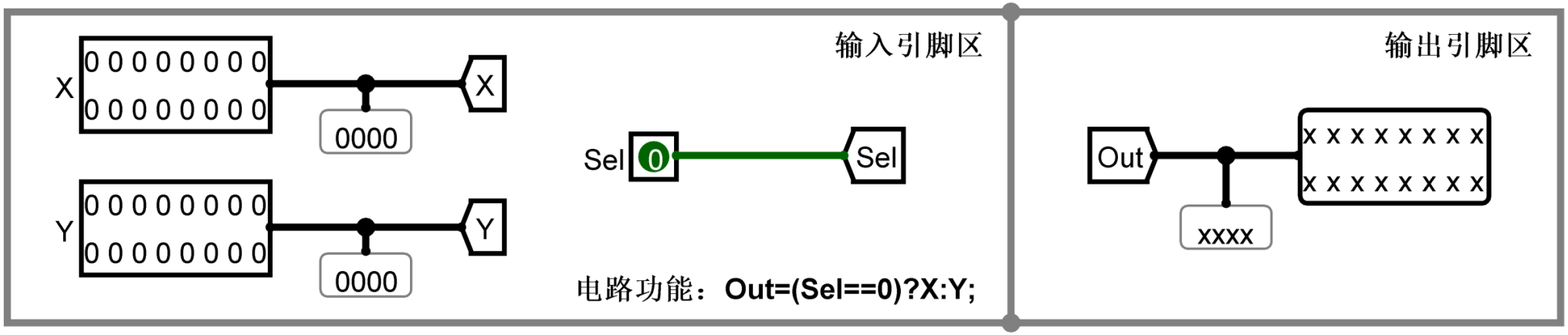
（一）：2路选择器（16位）设计

2路选择器（16位）的外部特性



- 输入：16位的输入 X， Y； 选择控制信号 Sel
- 输出：16位输出Out
- 功能：out = （Sel==0）? X:Y 当Sel为0，选择X作为输出；否则，选择Y输出

约束条件：只能运用线路库、逻辑门组件，输入输出库自行构建



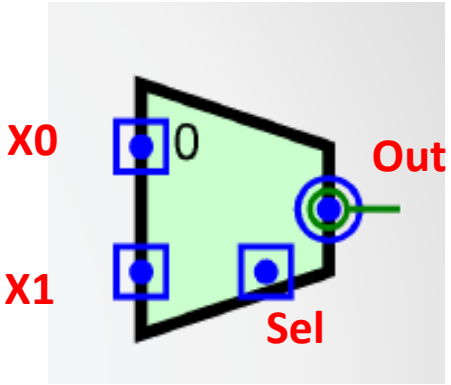
Tip：

模块化设计

设计思路：先构建1位2路选择器，再并发为16位

1. 构建2路选择器（1位）

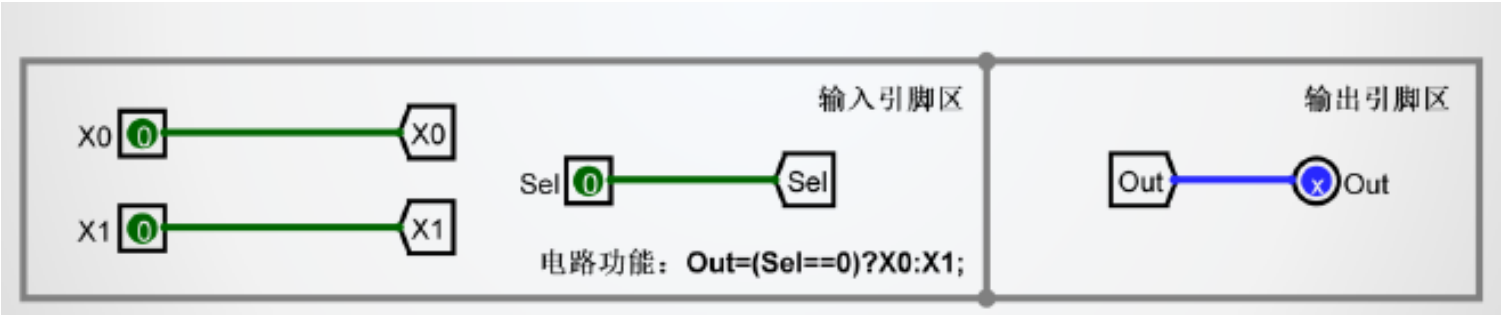
- 输入：1位输入X，Y； 选择控制信号Sel
- 输出：1位Out
- 功能：out = （Sel==0）? X:Y



封装外观

（1）封装编辑

（2）子电路编辑：不要增删改引脚，运用的输入输出引脚的隧道标签信号构建电路。



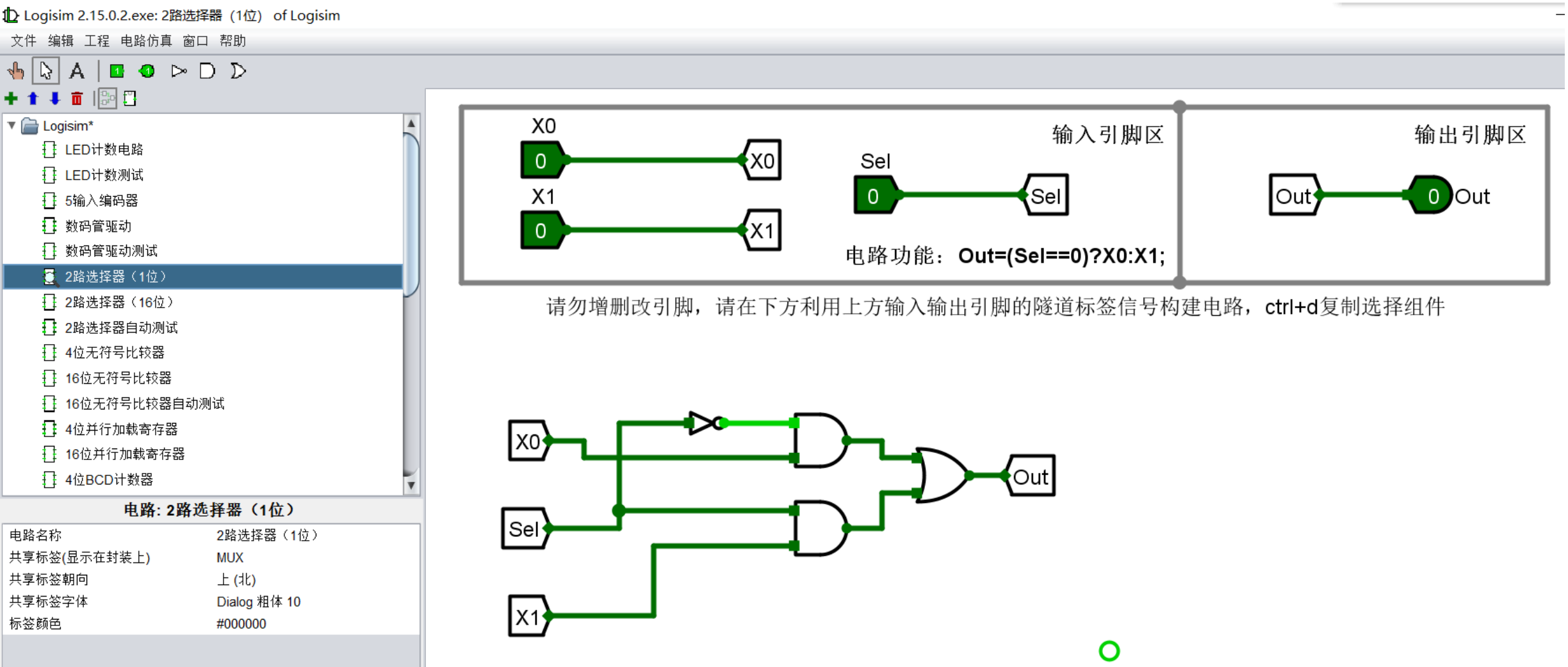
子电路编辑可选方法：

（1）手动绘制；

（2）利用真值表或逻辑表达式自动生成电路

提示：ctrl+d复制选择组件；隧道标签在线路组件库中

2路选择器（1位）的子电路编辑图例　方案不唯一

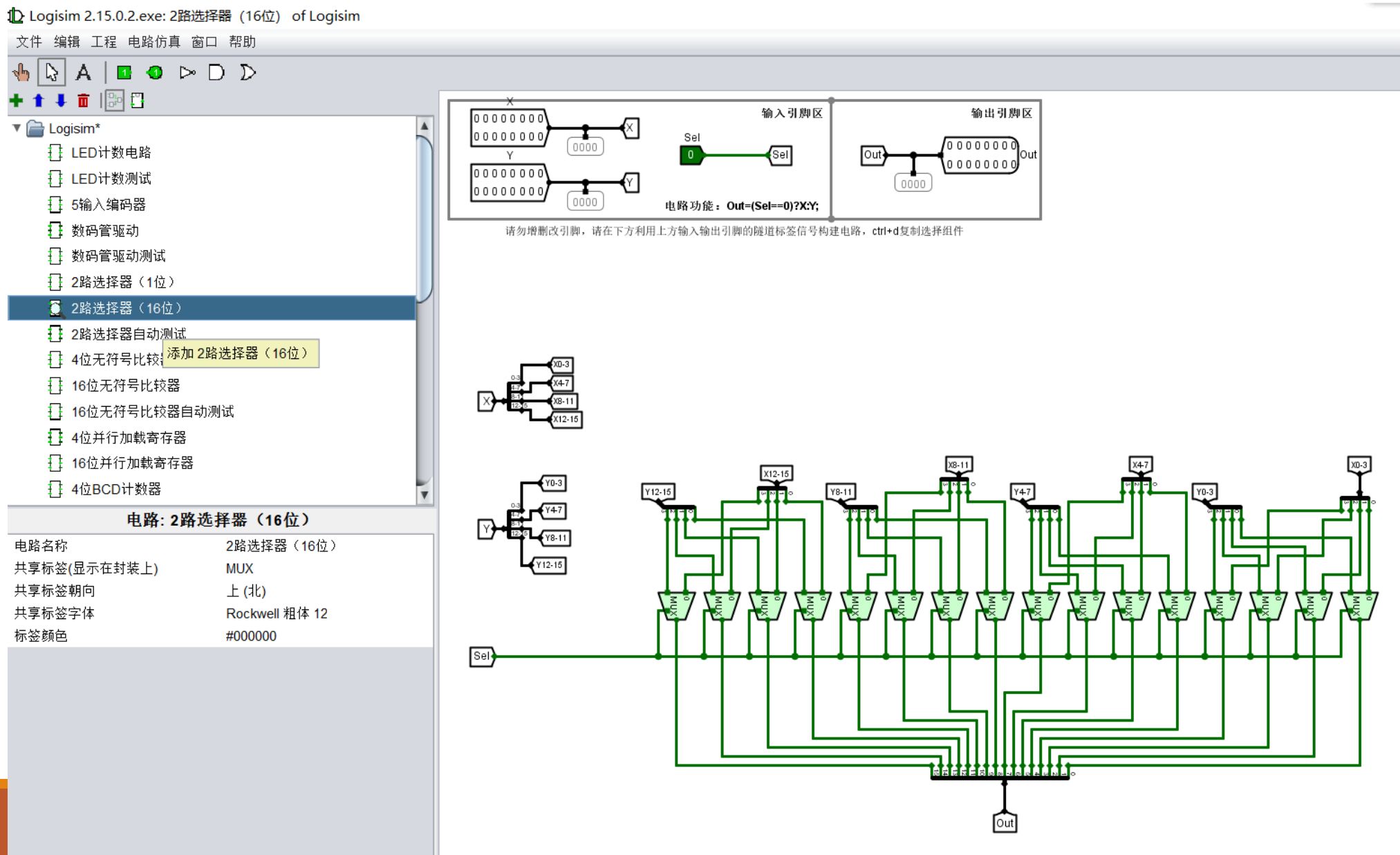


2、复制产生16个2路选择器（1位），连接形成1个2路选择器（16位）。

子电路编辑图例

方案不唯一

手动绘制时，练习标签隧道的使用，改善电路图的美观性与可读性

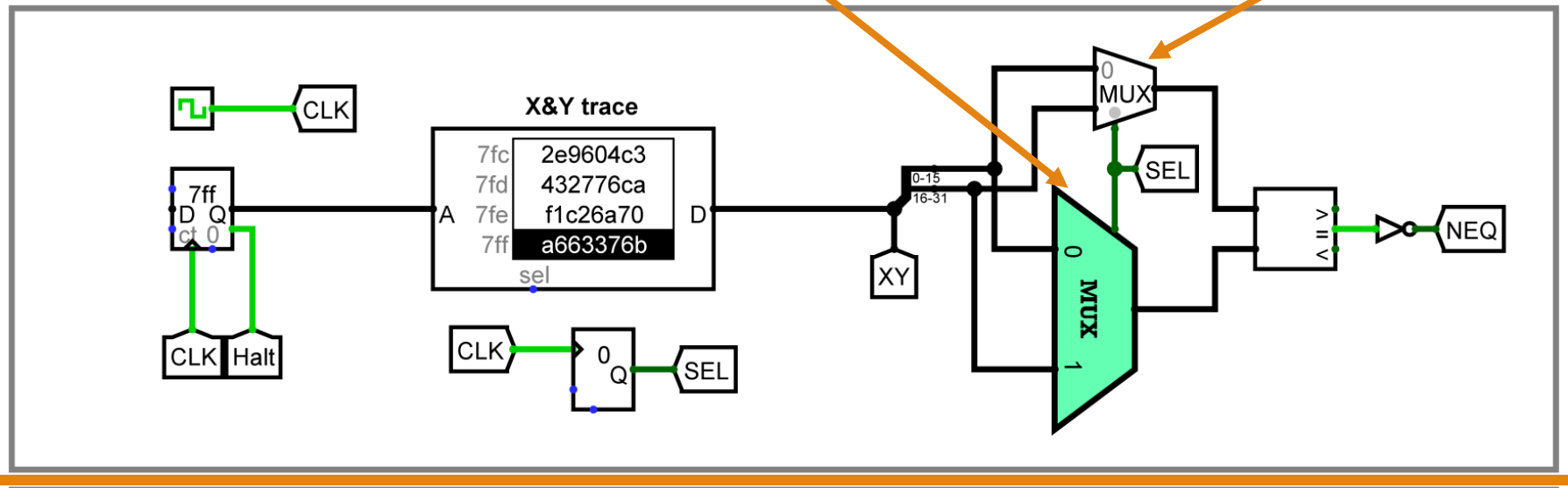


3、2路选择器（16位）的自动测试

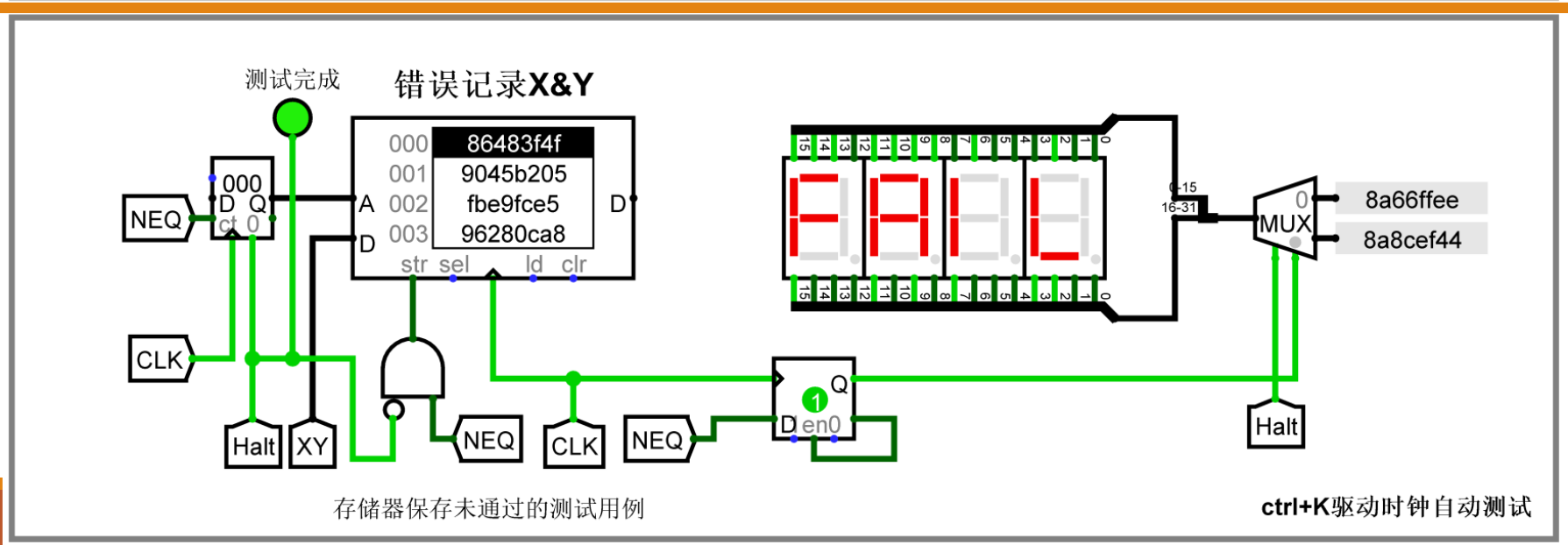
待测试的自行设计的
16位的2路选择器

用于对比的
标准2路选择器

测试逻辑图



测试电路

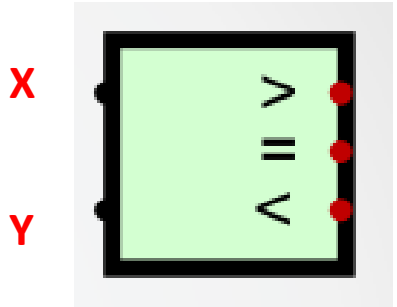


在实验文件中找到自动测试子电路

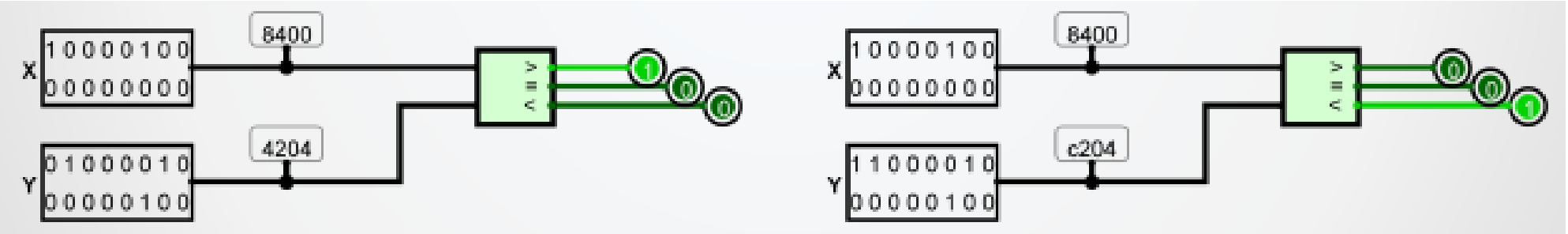
（二）：16位无符号比较器

无符号比较器的外部特性

- 输入：16位的输入 X， Y；
- 输出：大于（1位），等于（1位），小于（1位）
- 功能：无符号比较两个输入，输出结果。



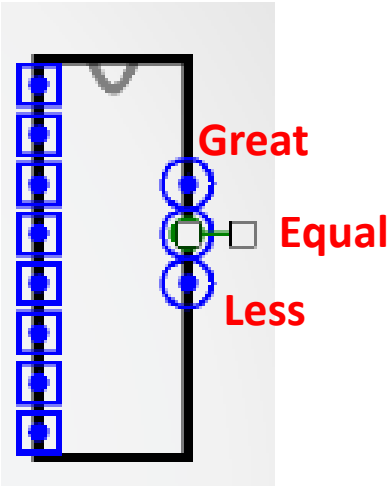
约束条件：只能运用线路库、逻辑门组件，输入输出库自行构建



Tip:

模块化设计

设计思路：先构建4位的无符号比较器，进而再扩展为16位



1. 构造4位无符号比较器

Great　　Equal　　Less

- 输入：4 位的输入 X， Y；
- 输出：大于（1位），等于（1位），小于（1位）
- 功能：无符号比较（当X>Y 时，Great = 1）

（1）封装编辑

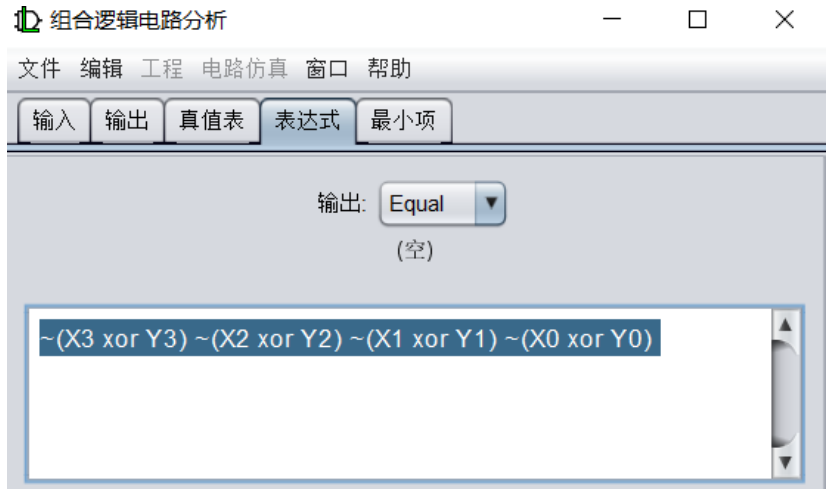
（2）子电路编辑：

分析真值表，形成各个输出的逻辑表达式，进而自动生成电路

4位比较器的简易真值表

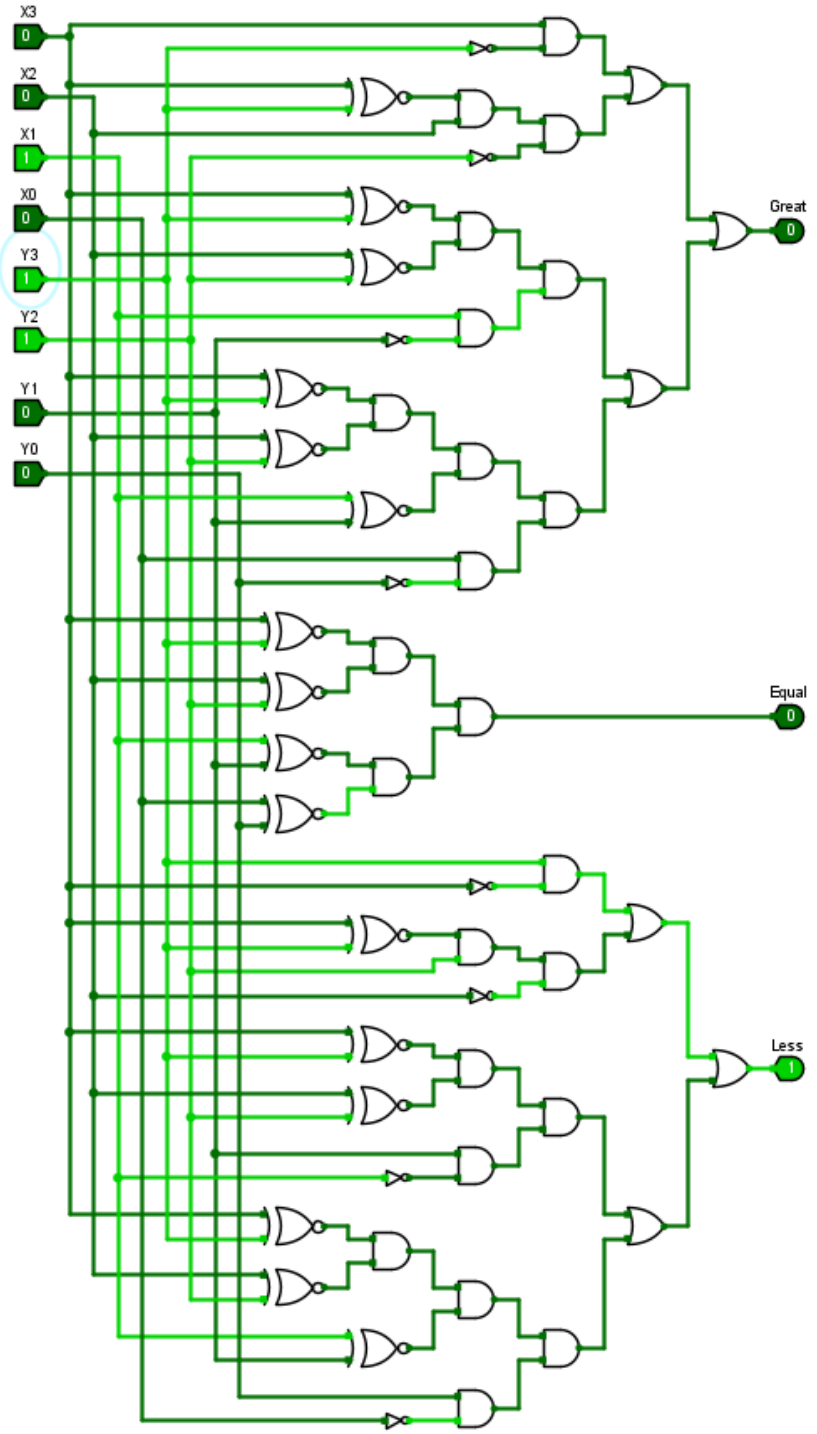
输入								输出		
X3	Y3	X2	Y2	X1	Y1	X0	Y0	Great	Equal	Less
>		X		X		X		1	0	0
=		>		X		X		1	0	0
=		=		>		X		1	0	0
=		=		=		>		1	0	0
=		=		=		=		0	1	0
<		X		X		X		0	0	1
=		<		X		X		0	0	1
=		=		<				0	0	1
=		=		=		<		0	0	1

➢ **Equal = (X3⊙Y3)&(X2⊙Y2)&(X1⊙ Y1)& X0⊙Y0**



提示：在Logisim中填入逻辑表达式时，异或为xor，同或可以看做异或的取反

➢ **Great、 Less?**

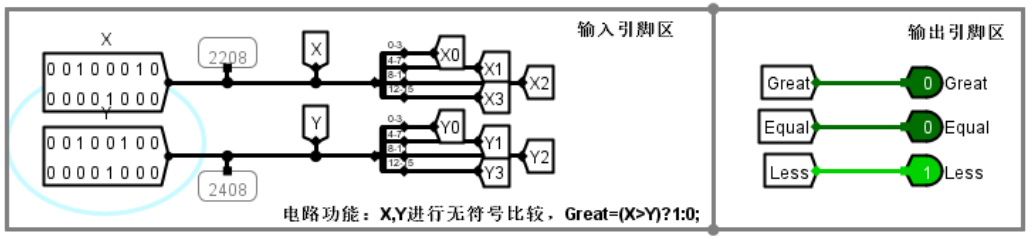


4位无符号比较器的子电路编辑图例

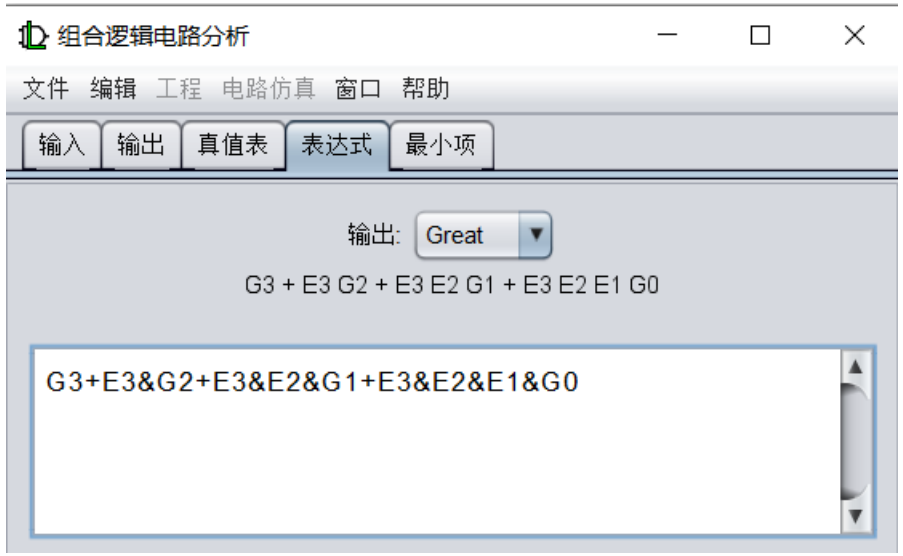
2、复制产生4个4位的比较器，再级连接形成1个16位的比较器

（1）X、Y中每4位对应1个4位比较器

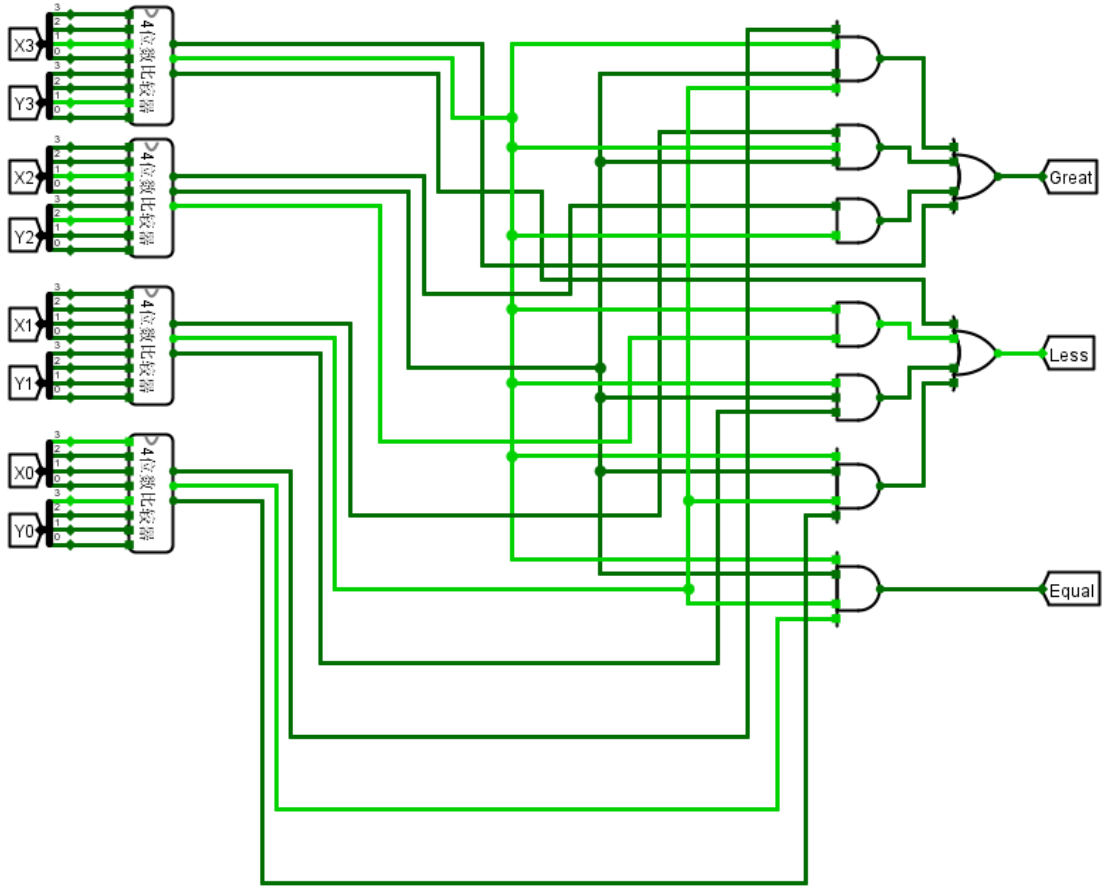
- 用已有隧道标签进行连接



（2）分析4个4位比较器的输出与最终输出的逻辑关系，生成级联电路



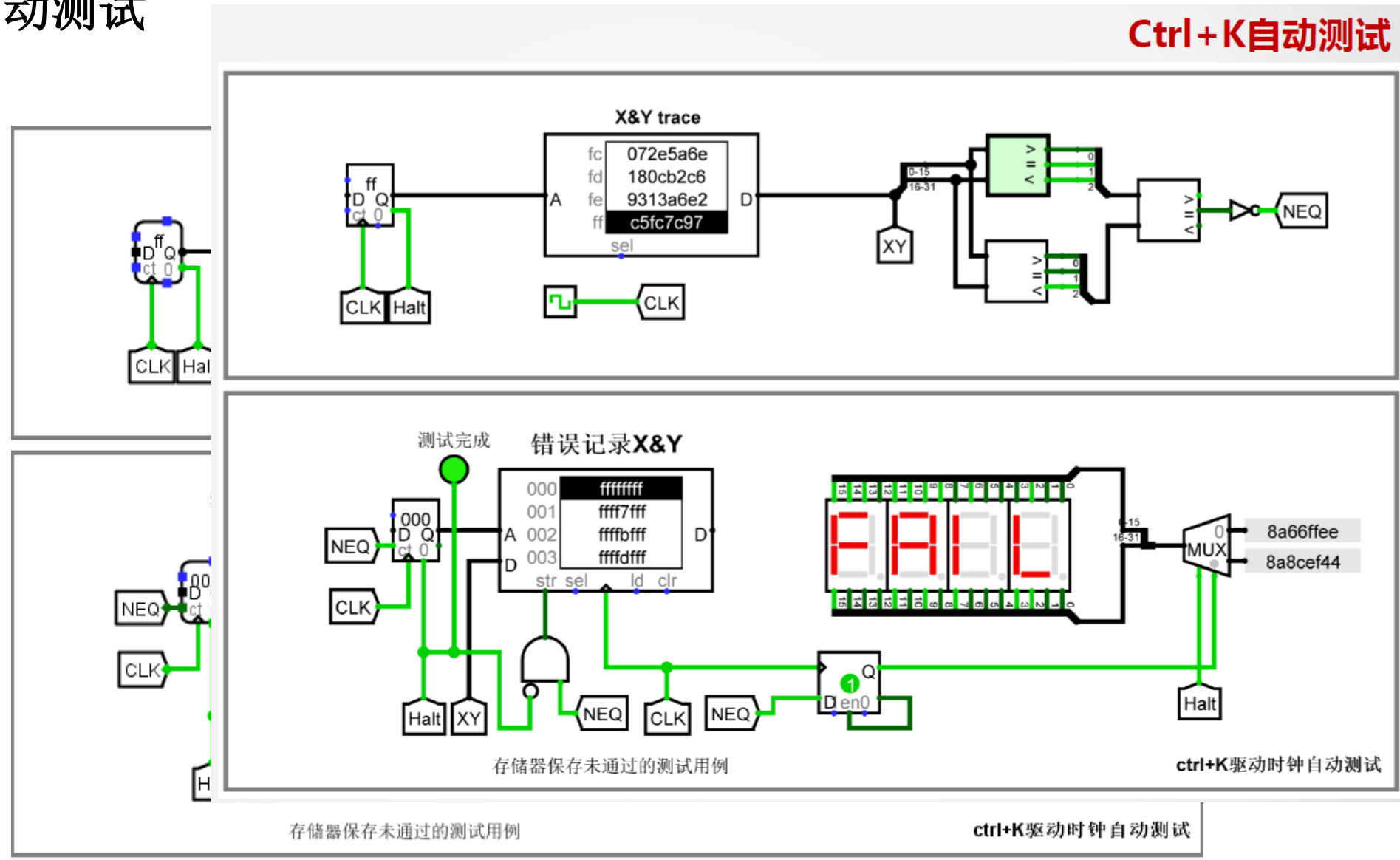
（3）整理电路，戳工具初步测试电路



16位无符号比较器的子电路编辑图例（方案不唯一）

3、16位比较器的自动测试

地址为ff时，用例全部测试完毕；可根据错误记录进一步调试



16位数无符号比较器自动测试

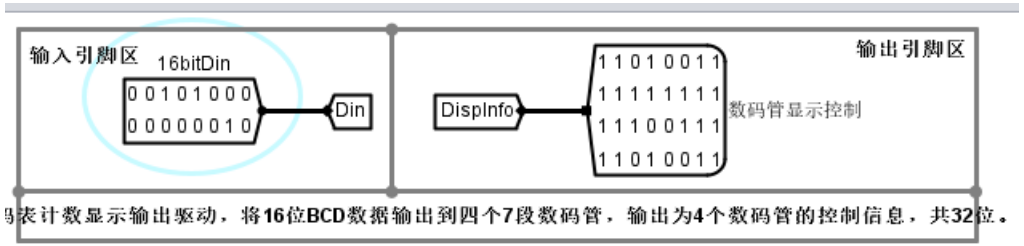
在实验文件中找到名为“16位无符号比较器自动测试”的测试电路

（三）：码表数码管的驱动

数码管显示DP

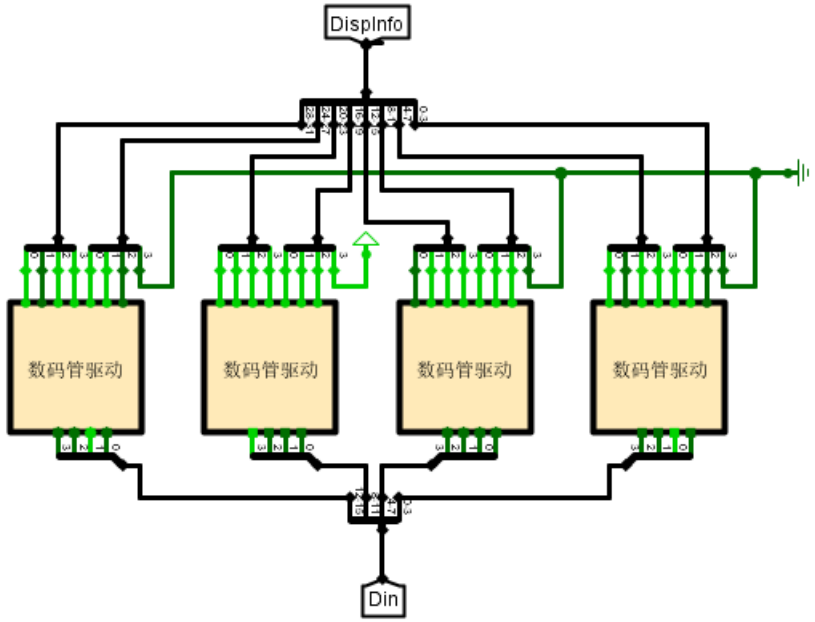
码表显示驱动的外部特性

- 输入：16位的BCD码（4位的10进制数）；
- 输出：4个7段数码管的控制信号（共32位）
  - S4T，S4B，S3T，S3B
  - S2T，S2B，S1T，S1B



设计思路：

- 利用新手实验已经构建的1个7段数码管驱动，并发形成4个7段数码管的驱动



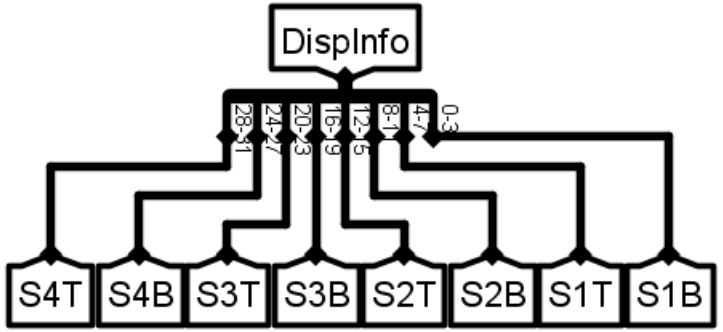
码表显示驱动电路编辑图例（方案不唯一）

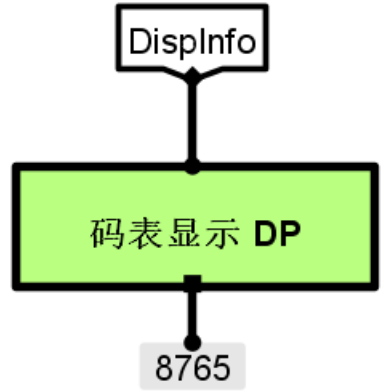
码表数码管驱动的测试

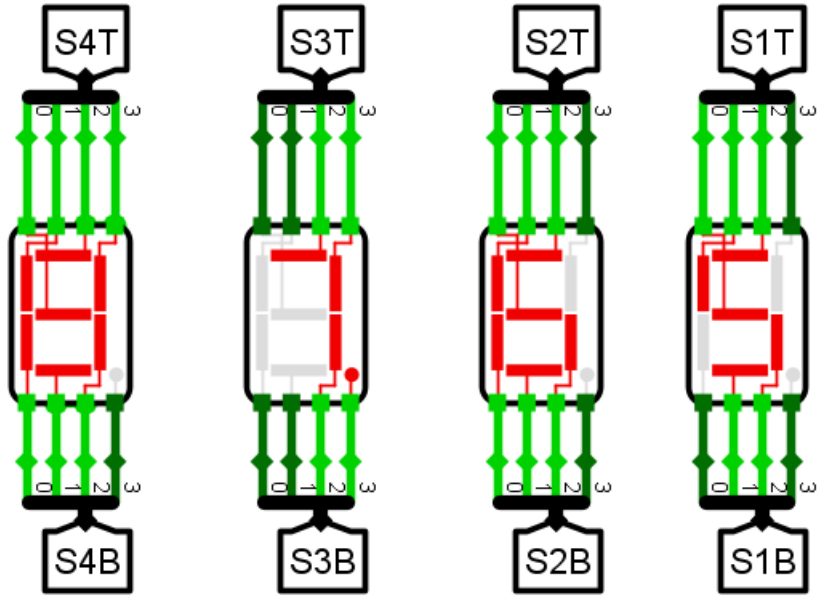
1、自行新建一个“码表显示驱动测试”文件

2、在文件中建立起如右图所示的测试电路。

- 添加码表显示驱动、常量、7段数码管、分线器等 组件
- 定义输入、输出的标签隧道
- 完成连接







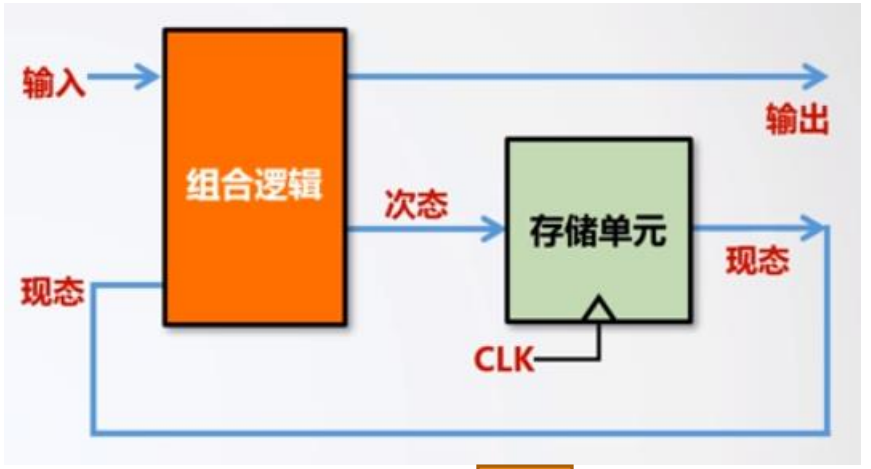
数字逻辑基础实验（2）

——同步时序逻辑设计

同步时序逻辑电路概述

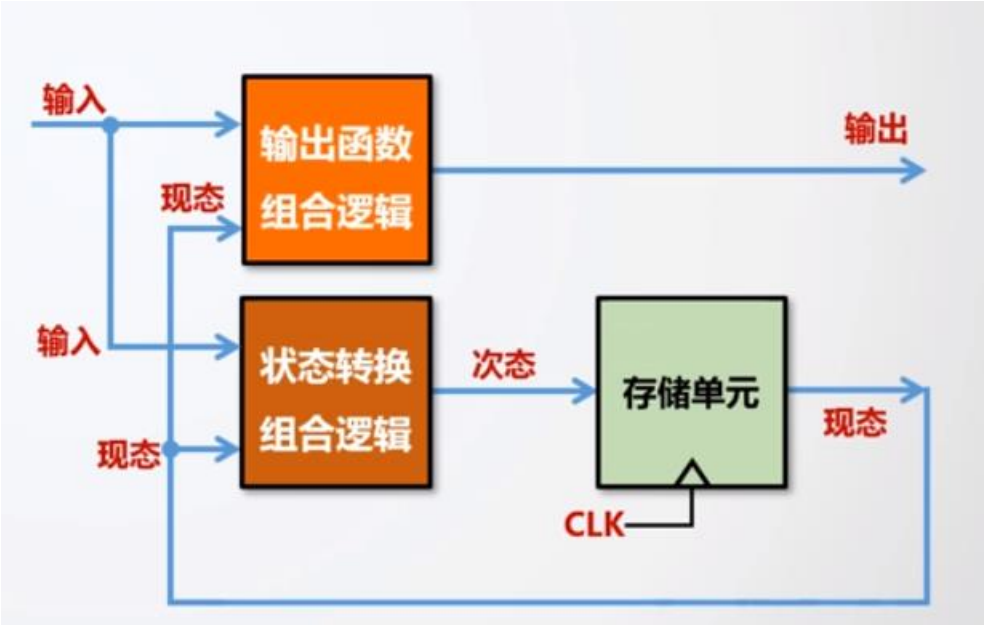
电路特征

- 由组合逻辑和存储单元构成
- 电路存在反馈
- 公共时钟进行同步



一般设计流程

- 构建状态图
- 构建状态转换逻辑
- 构建输出函数的逻辑
- 实现电路



状态转换逻辑设计

- 填写EXCEL真值表
- 根据功能需求填写状态转换表
- 自动生成状态转换电路逻辑表达式 （触发器输入函数）

S3	S2	S1	S0	In1	In2	In3	In4	In5	In6	In7	In8	最小项表达式	N3	N2	N1	N0
~S3&	~S2&	~S1&	~S0&	In1&	In2&	In3&	~In4&	In5&	~In6&							
~S3&	~S2&	~S1&	S0&	In1&		In3&	~In4&					~S3&~S2&~S1&S0&In1&In3&~In4				
												逻辑表达式->>>				

状态转换表 | 触发器输入函数自动生成 | 输出函数真值表 | 输出函数自动生成

输出函数组合逻辑设计

- 填写EXCEL真值表
- 根据功能需求填写输出函数真值表
- 自动生成输出函数逻辑表达式

S3	S2	S1	S0	In1	In2	In3	In4	In5	In6	In7	In8	最小项表达式	Out1	Out2	Out3	Out4	Out5
~S3&	~S2&	~S1&	~S0&	In1&		~In3&						~S3&~S2&~S1&~S0&In1&~In3	~S3&~S2&~S1&~S0&In1&~In3+				
~S3&	~S2&	~S1&	S0&		In2&							~S3&~S2&~S1&S0&In2					~S3&~S2&~S1&S0&In2+
~S3&	~S2&	~S1&	S0&			In3&						~S3&~S2&~S1&S0&In3				~S3&~S2&~S1&S0&In3+	
~S3&	~S2&	~S1&	S0&							In7&		~S3&~S2&~S1&S0&In7					
~S3&	~S2&	S1&	~S0&		In2&						In8&	~S3&~S2&S1&~S0&In2&In8		~S3&~S2&S1&~S0&In2&In8+	~S3&~S2&S1&~S0&In2&In8+		
~S3&	~S2&	S1&	~S0&									~S3&~S2&S1&~S0					
												逻辑表达式->>>	~S3&~S2&~S1&~S0&In1&~In3	~S3&~S2&S1&~S0&In2&In8	~S3&~S2&S1&~S0&In2&In8	~S3&~S2&~S1&S0&In3	~S3&~S2&~S1&S0&In2

状态转换表 | 触发器输入函数自动生成 | 输出函数真值表 | 输出函数自动生成

二、同步时序逻辑电路设计

实验目标

- 理解同步时序逻辑设计的基本流程
- 掌握状态转换逻辑、输出函数组合逻辑的设计方法
- 熟练运用Logisim构建运动码表中的时序逻辑部件

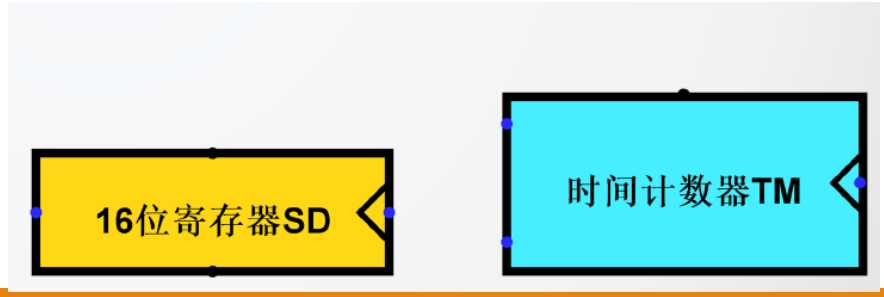
生成状态转换电路、输出函数的逻辑表达式

实验任务

- 16位寄存器（16位并行加载寄存器）
- 4位正向时间计数器（4位BCD计数器）
- 码表计数器

基础任务

已封装外观

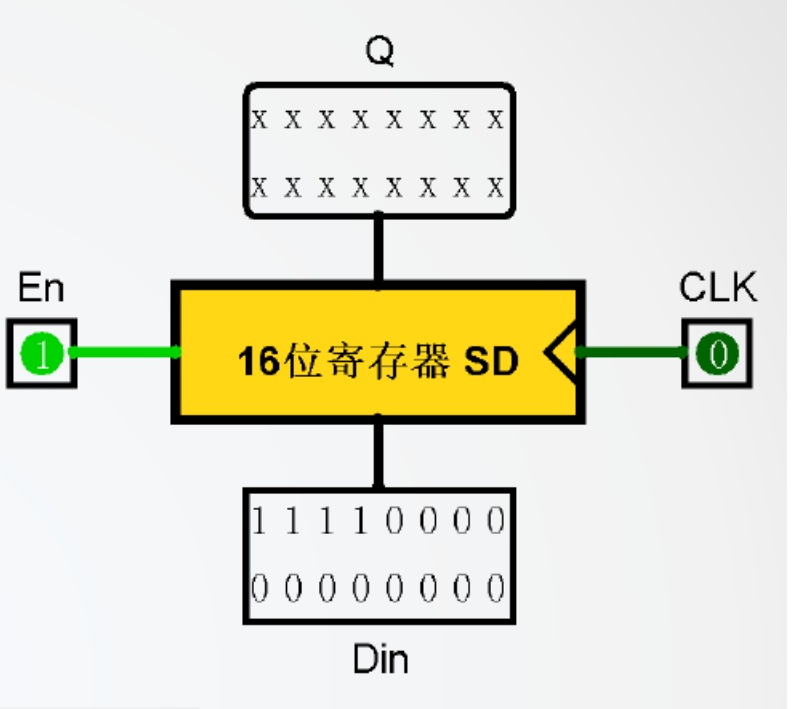


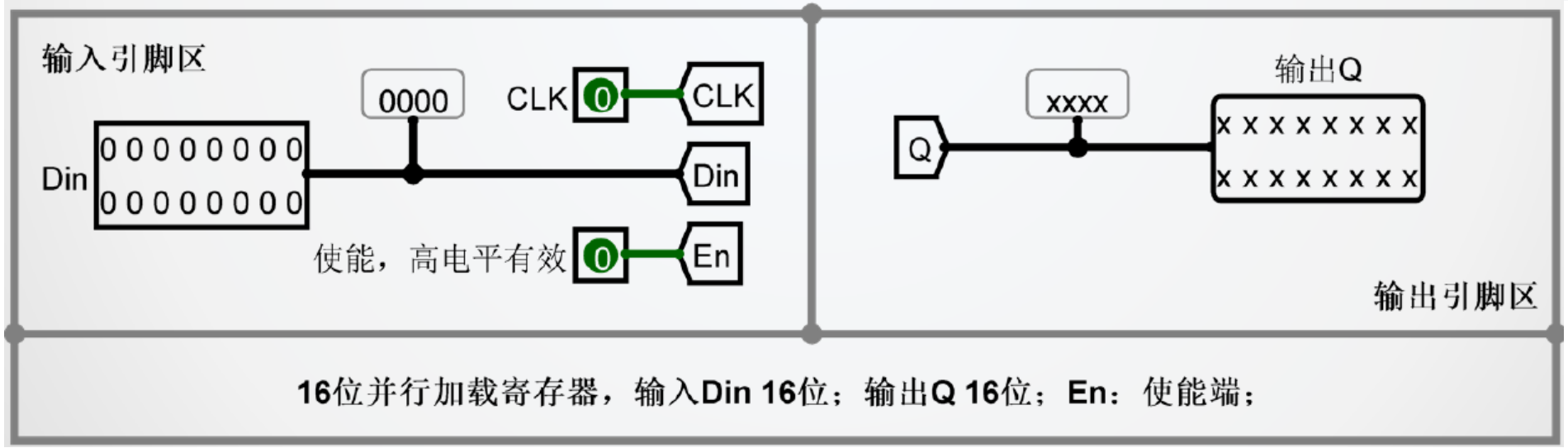
打开资源包中的“Logisim. circ”，可以看到已经建立的电路（待编辑）

（一）：16位并行加载寄存器

16位并行加载寄存器的外部特性

- 输入：16位的输入 Din， 使能信号En；
- 输出：16位输出Q
- 功能：Din → Q



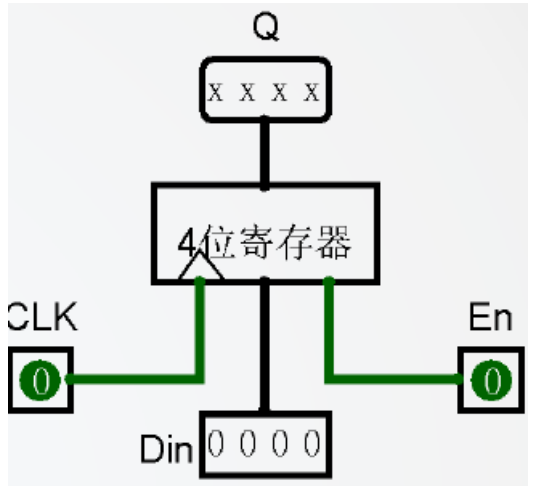


思考：16个1位触发器并发 or 4个4位触发器并发？

设计思路：先构建4位寄存器，再并发为16位寄存器

1. 构建 4位并行加载寄存器

   - 输入：Din（4位）； 使能信号En
   - 输出：Q（4位）
   - 功能：**Din → Q**

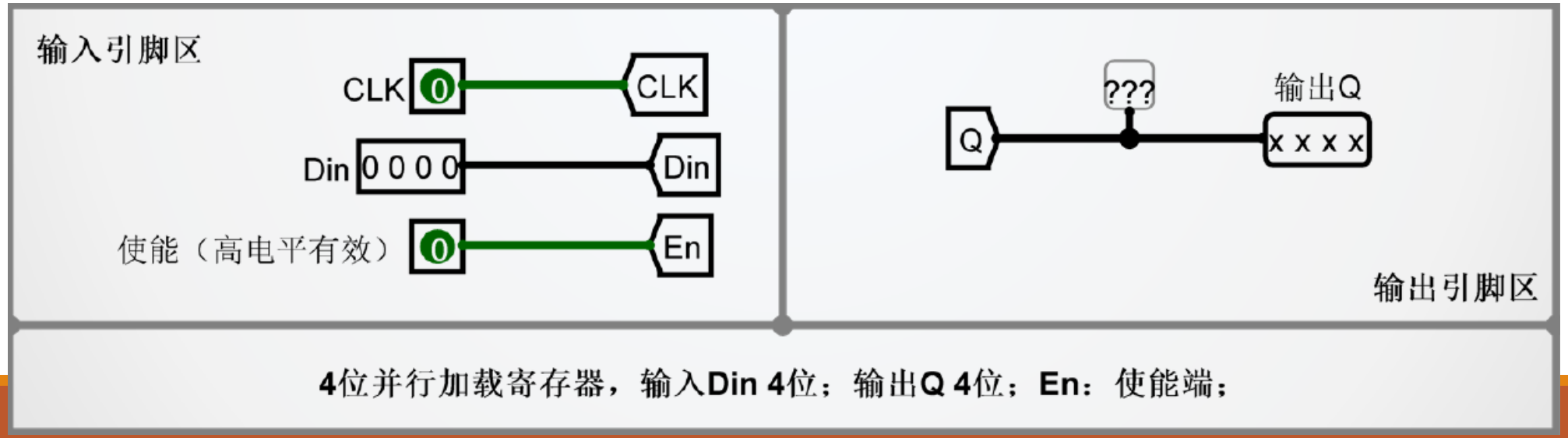


封装外观

（1）封装编辑

（2）子电路编辑：不要增删改引脚，运用的输入输出引脚的隧道标签信号构建电路。

使用D触发器为元件；属性选择包含使能端

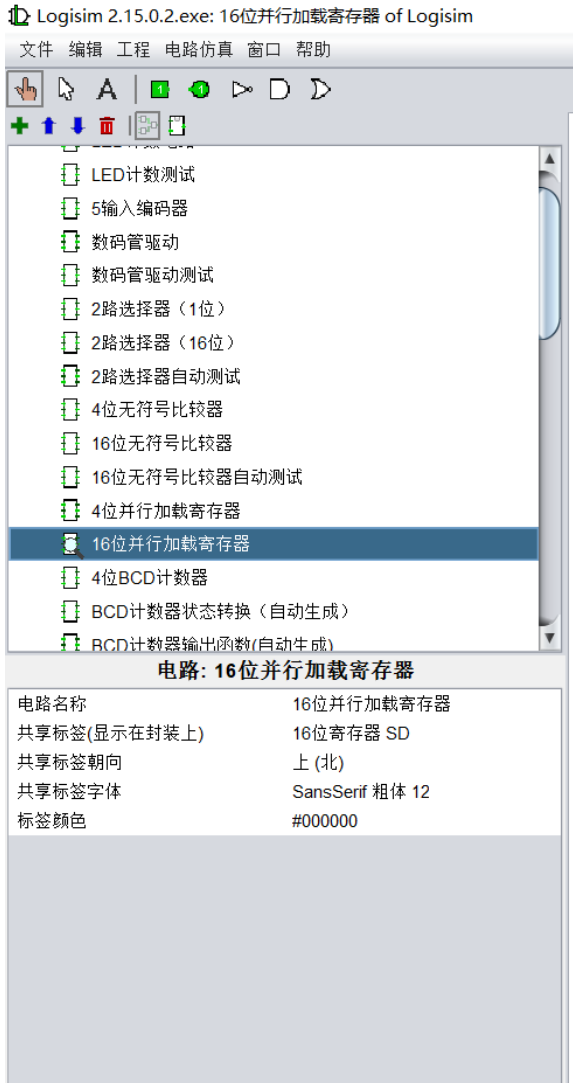


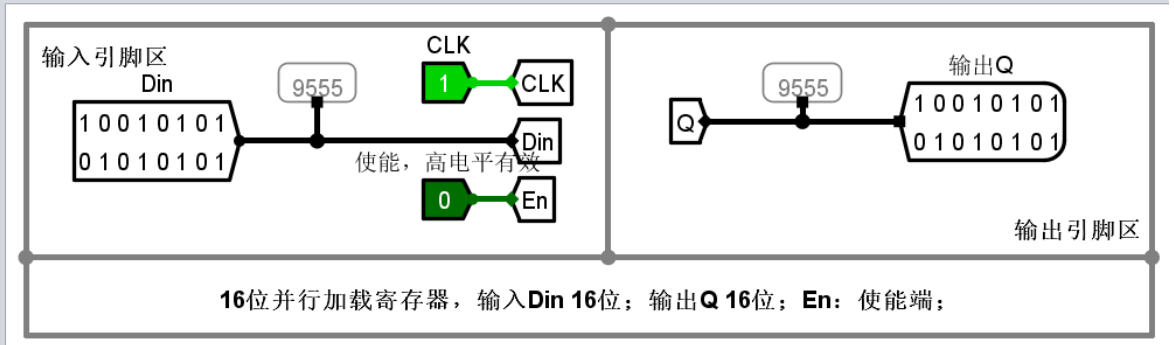
2、复制产生4个4位寄存器，再并联接形成1个16位并行加载寄存器

子电路编辑图例

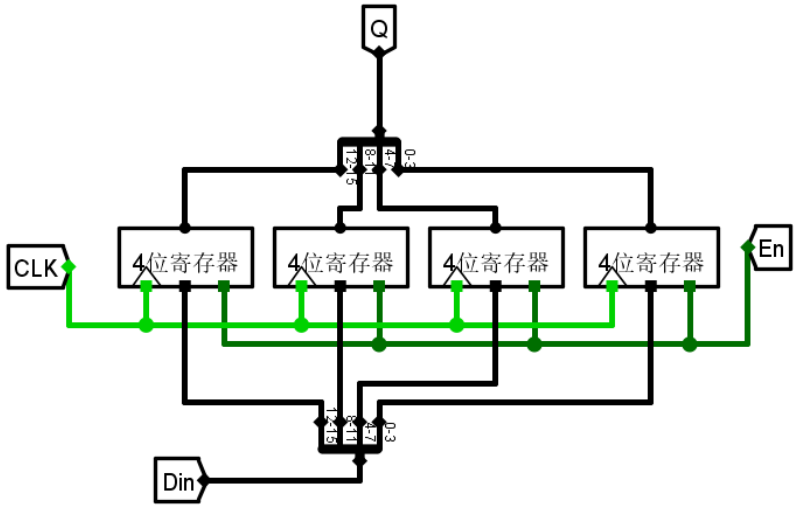
方案不唯一

手动绘制时，练习标签隧道的使用，改善电路图的美观性与可读性





请勿增删改引脚，请在下方利用上方输入输出引脚的隧道标签信号构建电路，ctrl+d复制选择组件



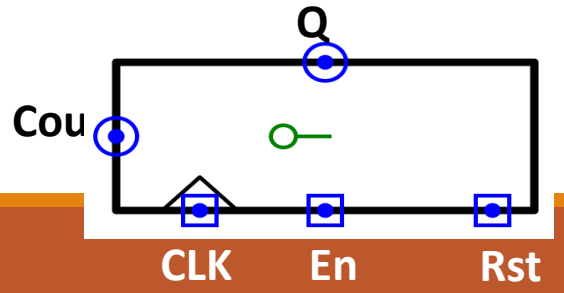
3、整理电路，运用戳工具简单测试

（二）：4位BCD计数器

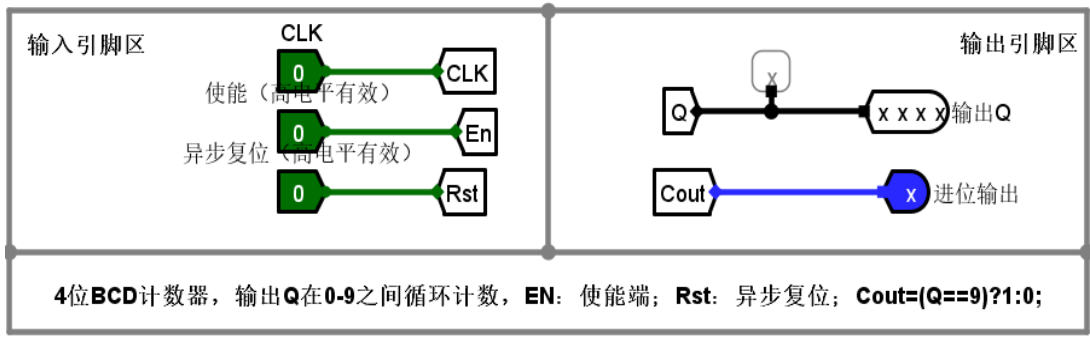
4位BCD计数器的外部特性

- 输入：时钟信号CLK， 使能信号En，异步复位Rst；
- 输出：4位输出Q，进位输出信号Cout
- 功能：

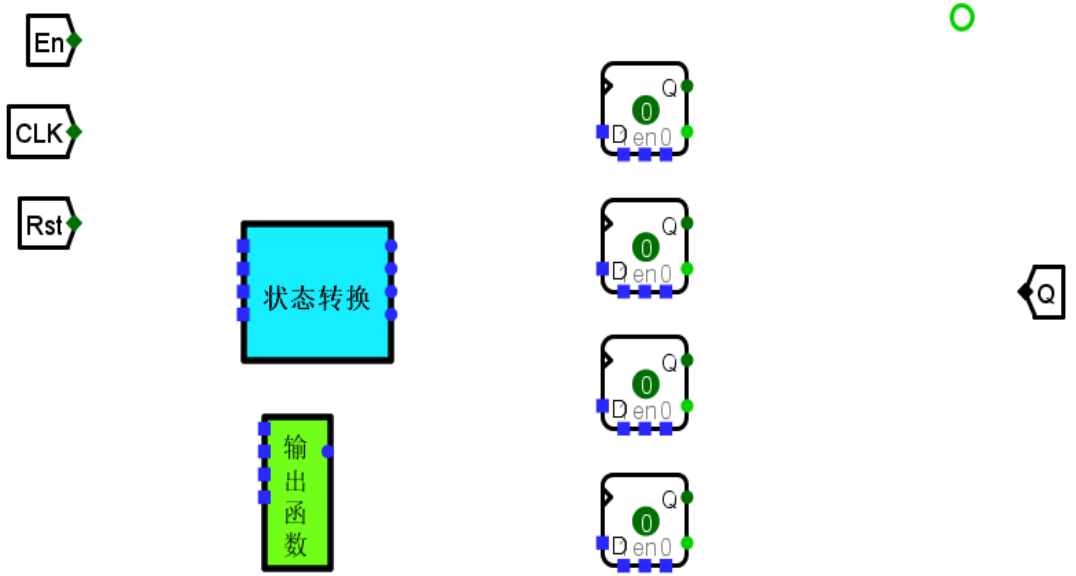
Q从9->0时，产生进位信号Cout



封装外观



请勿增删改引脚，请在下方利用上方输入输出引脚的隧道标签信号构建电路，ctrl+d复制选择组件



子电路编辑原始状态

“状态转换” 子电路设计

- 每个D触发器的输出就是1个状态位；
- 4个D触发器对应4个的状态位
- 设计重点——状态转换逻辑

1、填写“状态转换表”

2、检查“触发器输入函数自动生成”的逻辑表达式

3、打开logisim中 ”BCD计数器状态转换（自动生成）”文件，利用电路分析自动生成

状态转换表

当前状态(现态)					输入信号								下一状态（次态）				
S3	S2	S1	S0	现态10进制	In1	In2	In3	In4	In5	In6	In7	In8	次态10进制	N3	N2	N1	N0
0	0	0	0	0									1	0	0	0	1
0	0	0	1	1									2	0	0	1	0
0	0	1	0	2									3	0	0	1	1
0	0	1	1	3									4	0	1	0	0

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S3	S2	S1	S0	In1	In2	In3	In4	In5	In6	In7	In8	最小项表达式	N3	N2	N1	N0
~S3&	~S2&	~S1&	~S0&									~S3&~S2&~S1&~S0				~S3&~S2&~S1&~S0
~S3&	~S2&	~S1&	S0&									~S3&~S2&~S1&S0			~S3&~S2&~S1&S0+	
	~S2&	S1&	~S0&									~S2&S1&~S0			~S2&S1&~S0+	~S2&S1&~S0
	~S2&	S1&	S0&									~S2&S1&S0		~S2&S1&S0+		
	S2&	~S1&	~S0&									S2&~S1&~S0		S2&~S1&~S0+		S2&~S1&~S0
	S2&	~S1&	S0&									S2&~S1&S0		S2&~S1&S0+	S2&~S1&S0+	
	S2&	S1&	~S0&									S2&S1&~S0		S2&S1&~S0+	S2&S1&~S0+	S2&S1&~S0
	S2&	S1&	S0&									S2&S1&S0	S2&S1&S0+			
S3&	~S2&	~S1&	~S0&									S3&~S2&~S1&~S0	S3&~S2&~S1&~S0+			S3&~S2&~S1&~S0
S3&	~S2&	~S1&	S0&									S3&~S2&~S1&S0				

状态转换表 | 触发器输入函数自动生成 | 输出函数真值表 | 输出函数自动生成

子电路编辑原始状态

“输出函数” 子电路设计

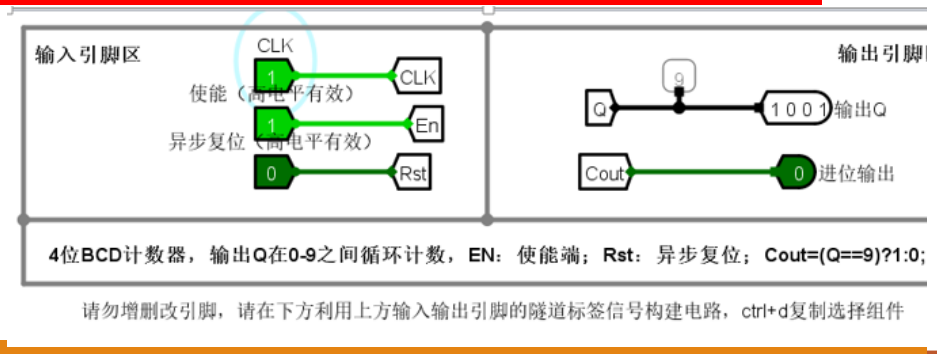
- 输入：4位现态S3-S0
- 输出：Cout（进位输出）
- 何种情况下Cout=1？

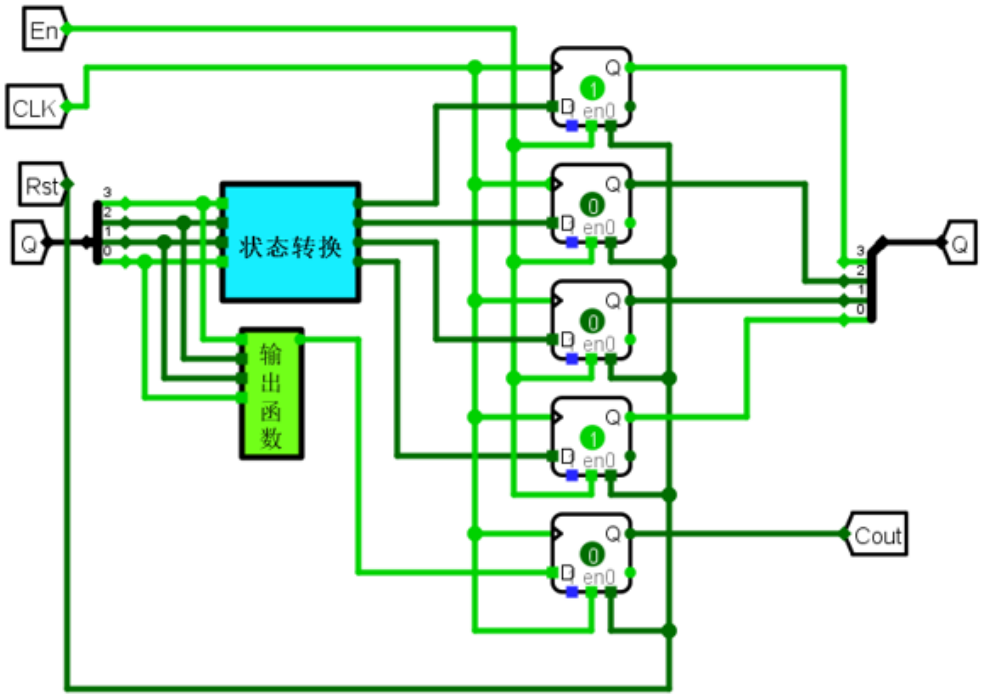
1、写出真值表 or Cout的逻辑表达式

2、打开logisim中 ”BCD计数器输出函数(自动生成)” 文件，利用电路分析功能自动生成电路

整体连接，并利用戳工具初步测试

- Rst有效时，应产生什么效果？
- Cout如何更新？



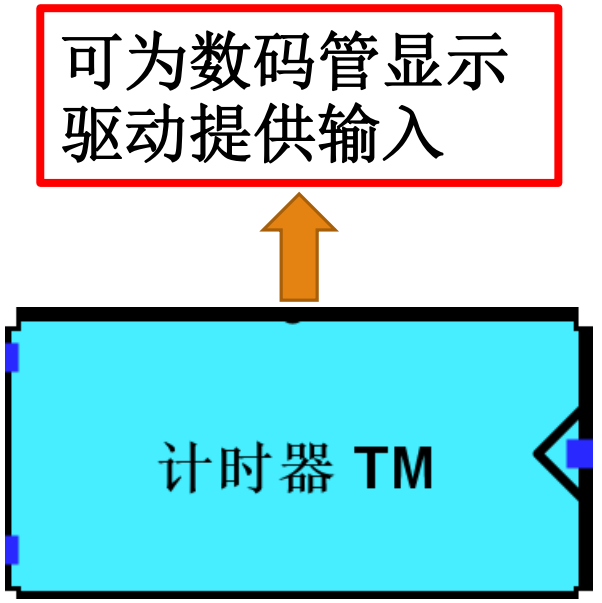


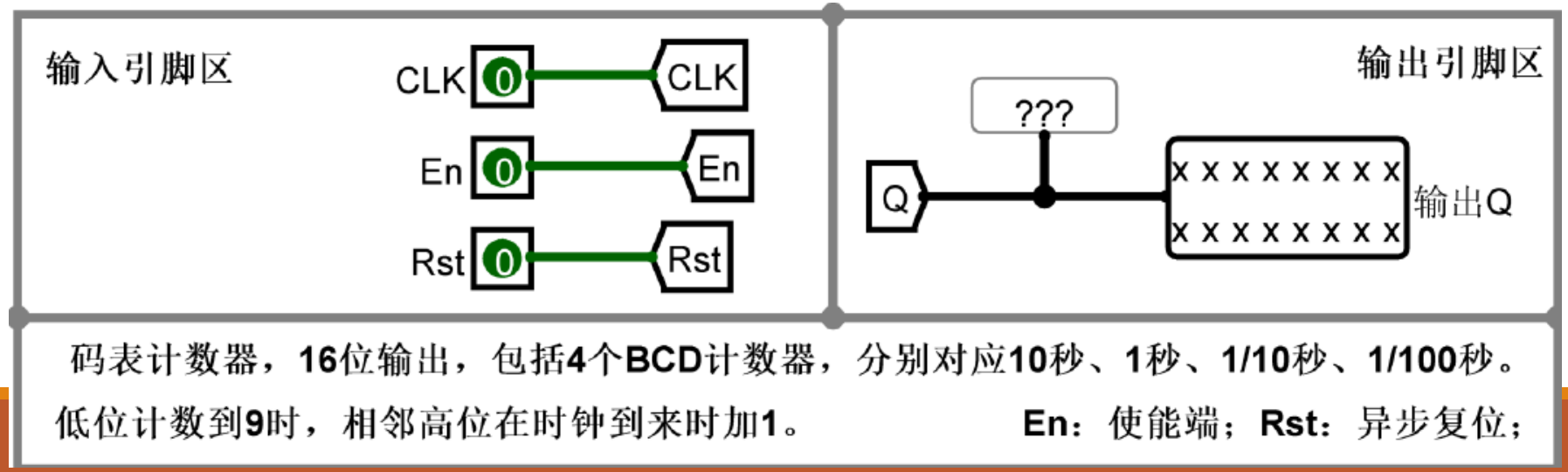
子电路编辑图例　方案不唯一

（三）：码表计数器（16位输出）

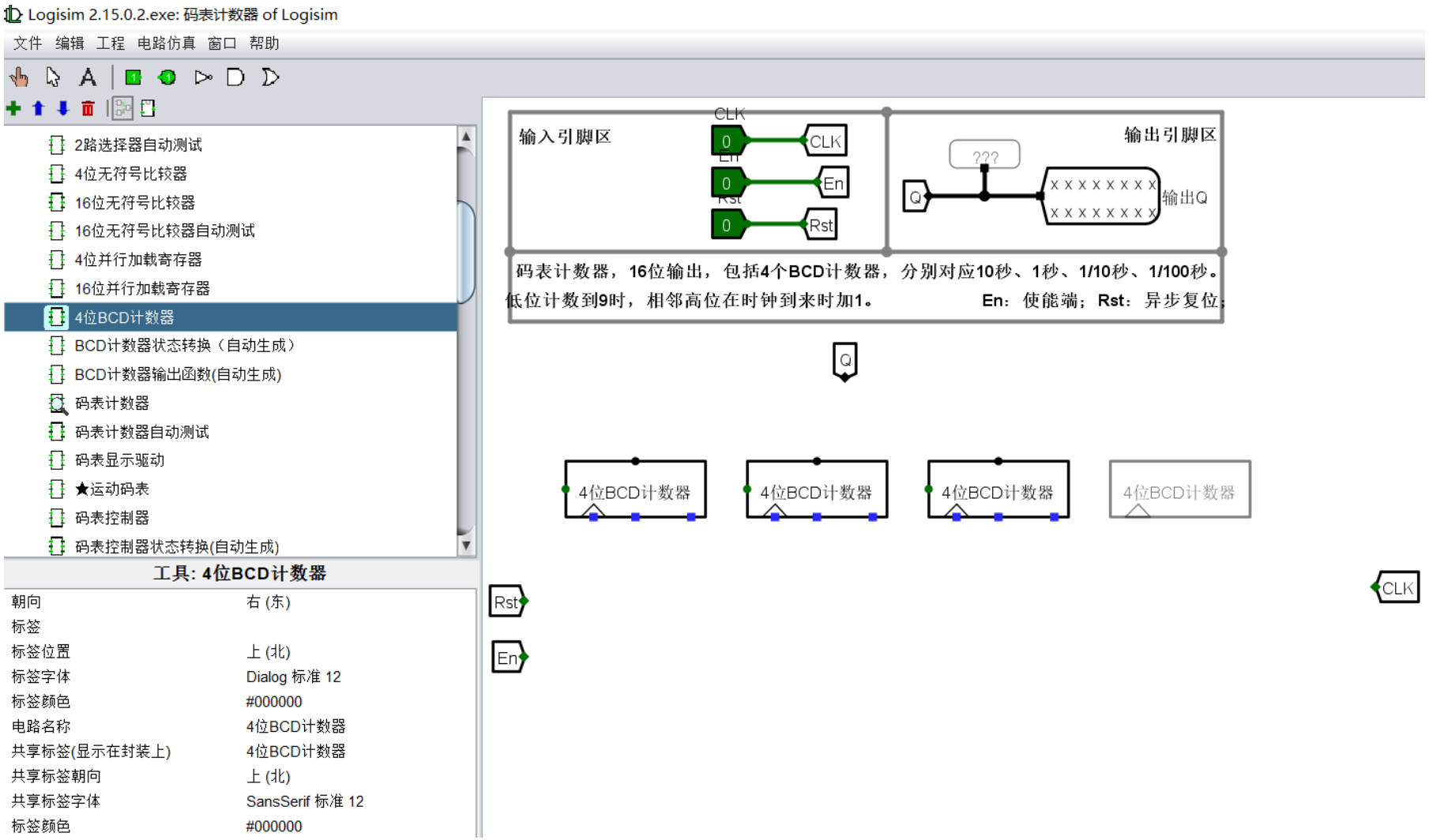
码表计数器的外部特性

- 输入：时钟信号CLK， 使能信号En，异步复位Rst；
- 输出：16位输出Q
- 结构：包含4个BCD码计数器
- 功能：低位计数器从9 到 0 时，相邻高位计数器 加1





设计思路：利用4个的4位BCD计数器级联而成

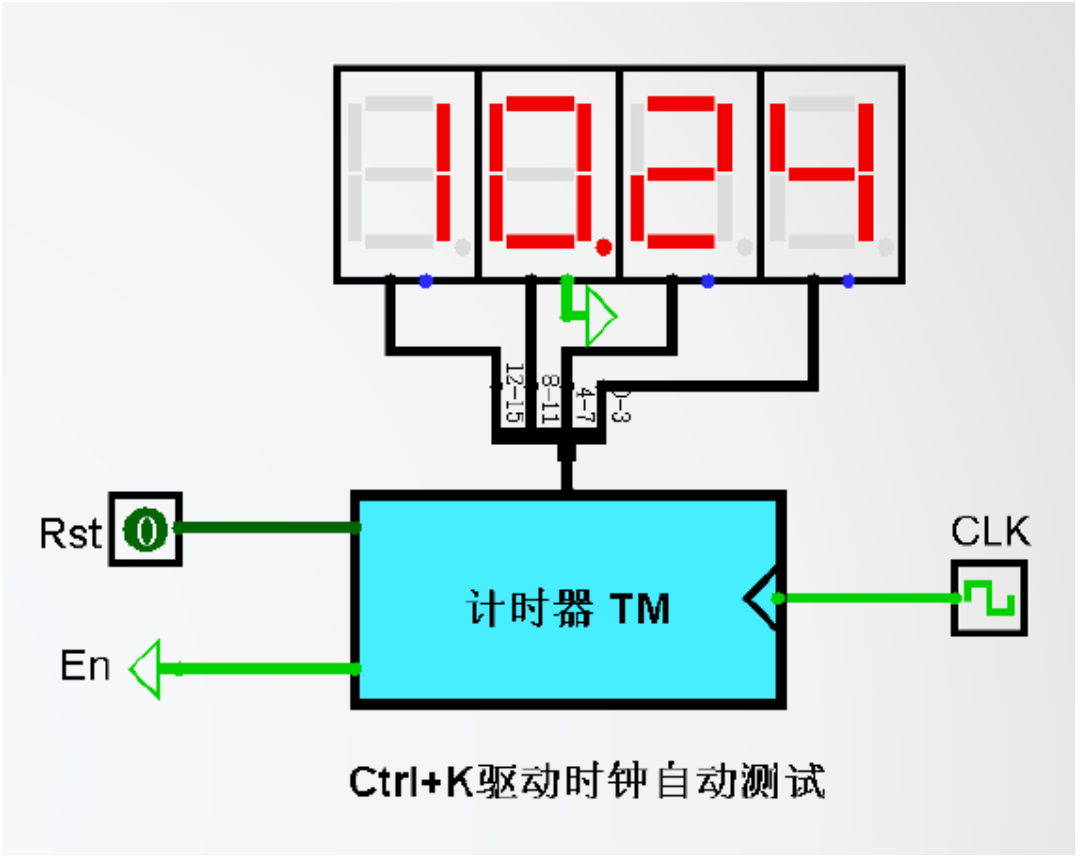


- 每个4位BCD计数器的使能端En、复位端Rst、输出端Q，与整个码表计数器的引脚如何连接？

- 低1级的数码管如何驱动高一级的数码管计数？

码表计数器的自动测试

- 打开自动测试文件，按下Crtl+K
  自动开始测试
- 也可以根据需要单步时钟调试
- 注意观察低位向高位的进位是否
  有错



Ctrl+K驱动时钟自动测试

数字逻辑基础实验（3）

——小型数字系统运动码表

三、数字系统综合设计

数字系统的设计流程

1、数字系统的需求分析

- 外部整体输入、输出分析，包括数据、控制信号与状态

2、功能部件设计（分模块设计）

- 得到内部的控制信号、状态

3、构建数据通路

- 从数据流动的角度，连接各功能部件

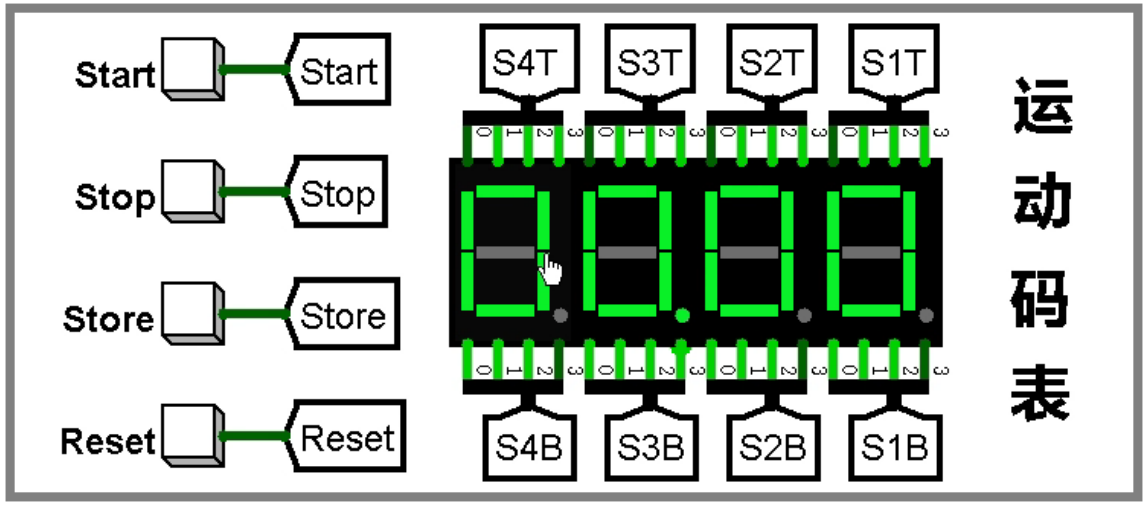
4、构建控制单元

- 绘制系统状态图
- 构建状态转换电路
- 构建输出函数电路（生成所需要的内部、外部控制信号，产生状态输出）
- 构建控制单元

5、系统集成联调

- 从控制流角度，连接控制单元（4）和各执行部件构成的数据通路（3）

构建小型数字系统——运动码表



1、运动码表初步功能需求分析

运动码表的外部特性

- 输入：4个按钮； 输出： 4个7段数码管

运动码表的功能

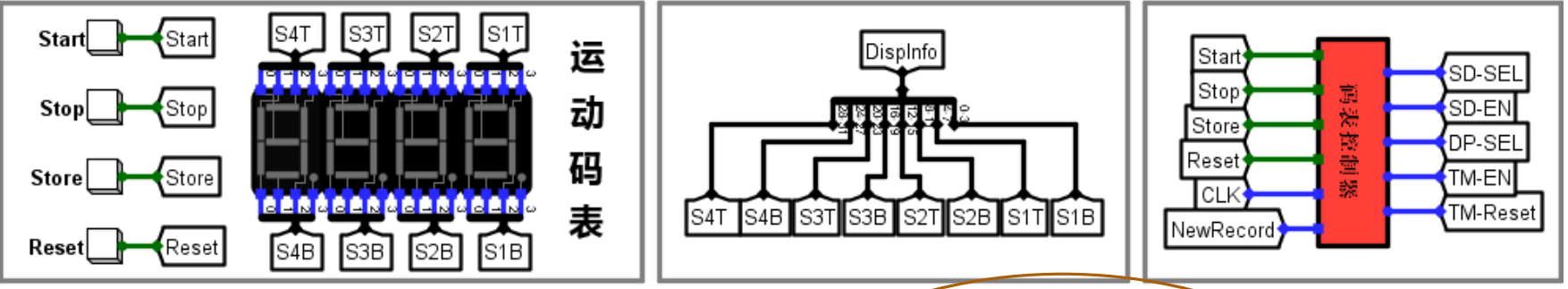
- **Start**：计时器归0，重新开始计时
- **Stop**：停止计时，显示当前计时数据
- **Store**：尝试更新系统记录的计时数据（满足条件：当前计时数据 < 系统记录）
- **Reset**：复位，使得计时器 = 00.00， 系统记录 = 99.99

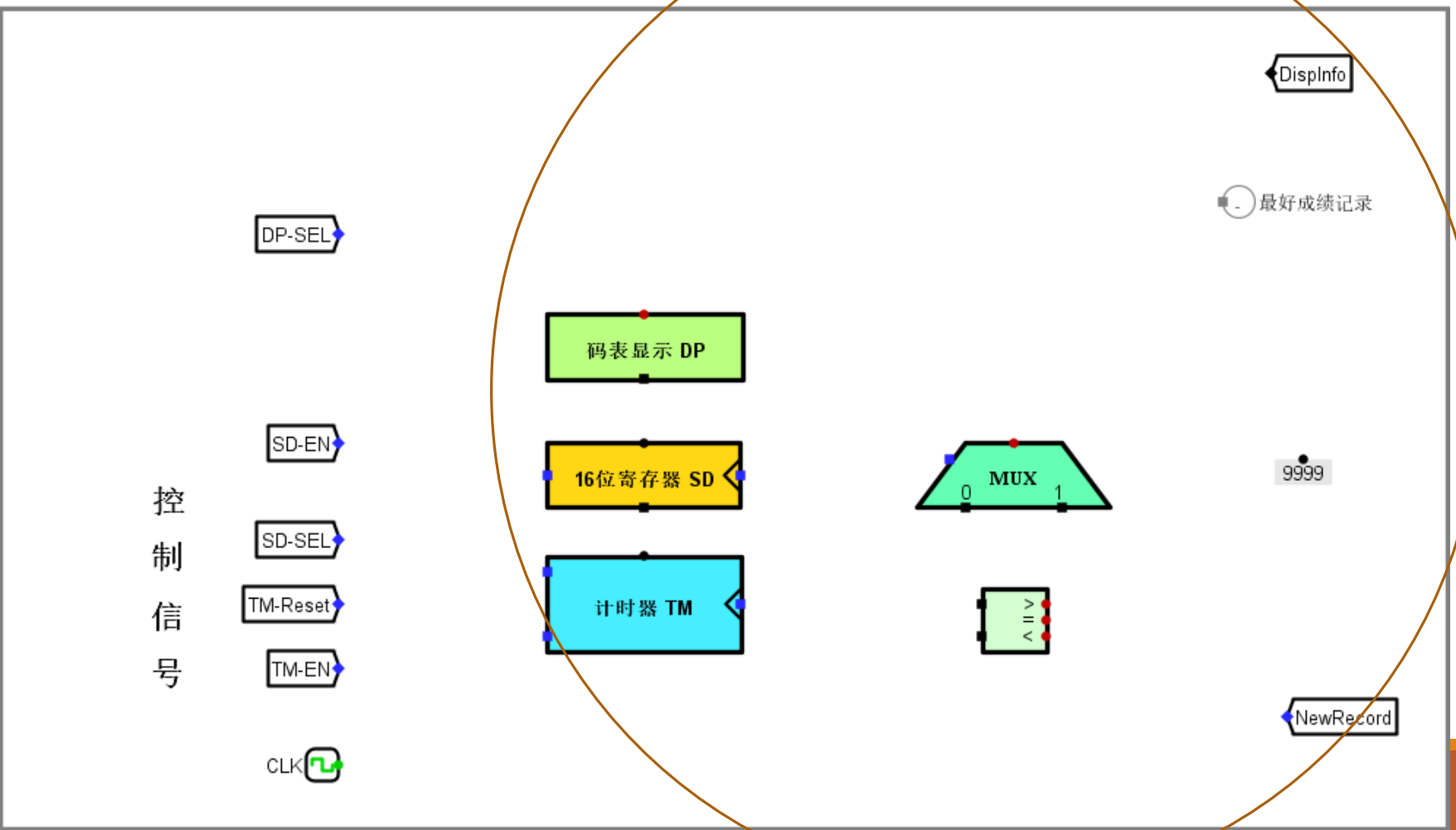
2、运动码表功能部件分析

#	功能部件	控制信号	输入	输出
1	时间计数器TM	TM-En,TM-Rst	CLK	时间计数输出16位
2	16位寄存器SD	SD-En	CLK，Din(16位)	Q(16位)
3	数码管显示DP		Din(16位)	DisplayInfo(32位)
4	比较器			
5	2路选择器	Sel		

同步时序电路设计

组合逻辑电路设计





运动码表的电路编辑界面

- **Start**：计时器归0，重新开始计时
- **Stop**：停止计时，显示当前计时数据
- **Store**：尝试更新系统记录的计时数据（满足条件：当前计时数据 < 系统记录）
- **Reset**：复位，使得计时器 = 00.00， 系统记录 = 99.99

3、运动码表数据通路构建

1、完成各功能部件输入来源表

#	功能部件	数据输入	来源	备注
1	时间计数器TM			
2	16位寄存器SD	CLK，Din(16位)	99.99 或 当前记录	增加2路选择器 SD-Sel
3	数码管显示DP	Din(16位)	TM.Q 或 SD.Q	增加2路选择器 DP-Sel
4	比较器		当前计时 & SD.Q	NewRecord

2、按数据流动过程连接数据通路

Tip：有多个输入来源时，需增加多路选择器

4、运动码表控制单元构建（重点！！）

外部的控制输入信号

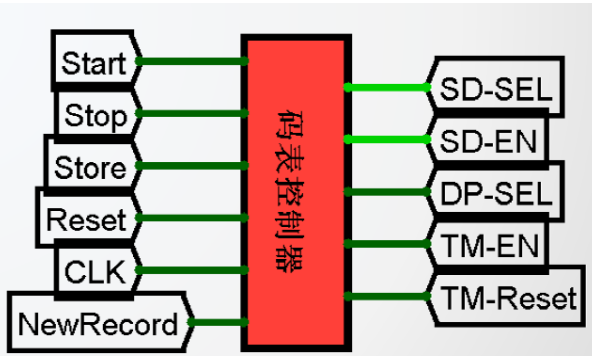
- Start 、Stop、Store、Reset

内部的状态反馈信号

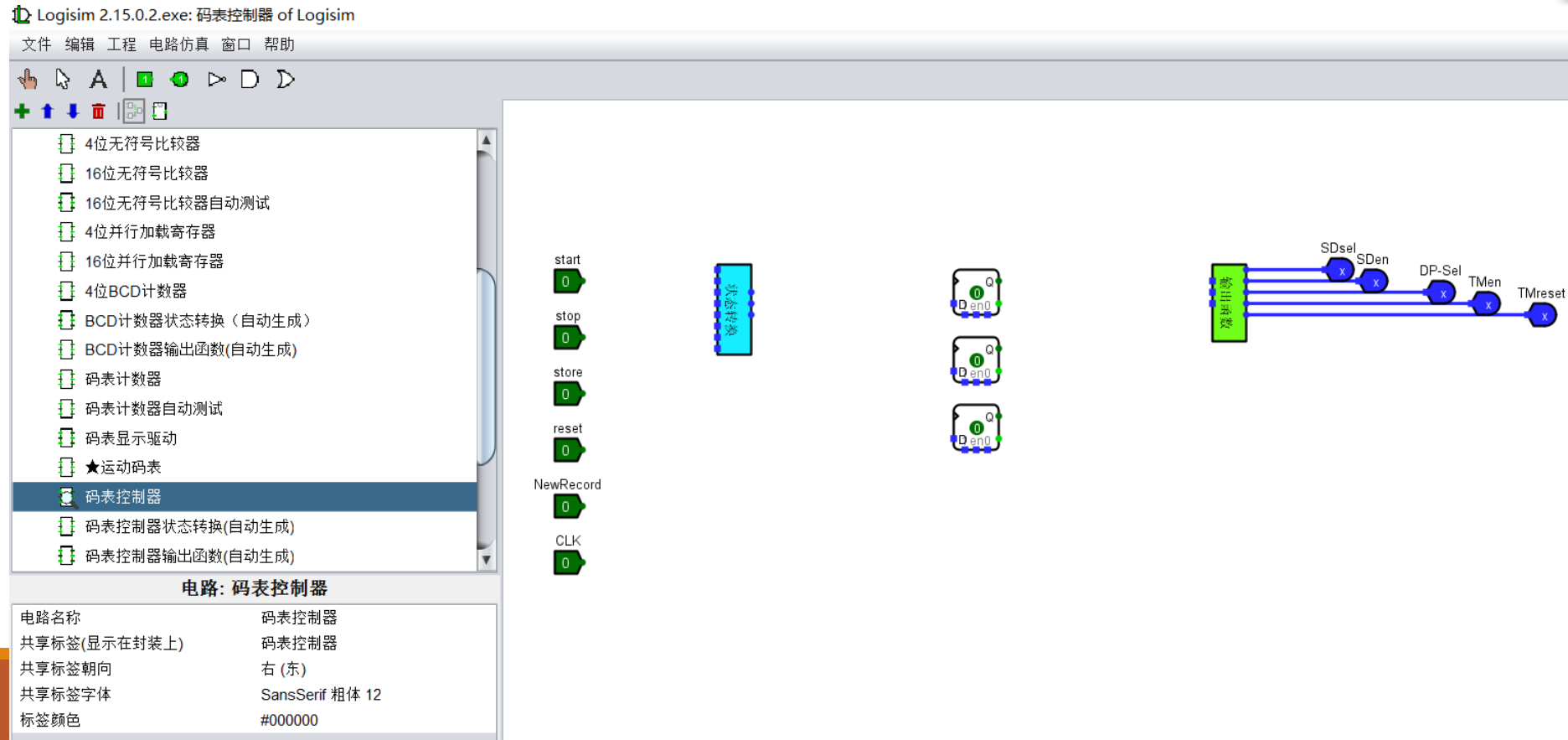
- NewRecord

定义控制单元的输出信号

- SD-SEL、SD-EN 、TM-EN、TM-Rest、 DP-SEL

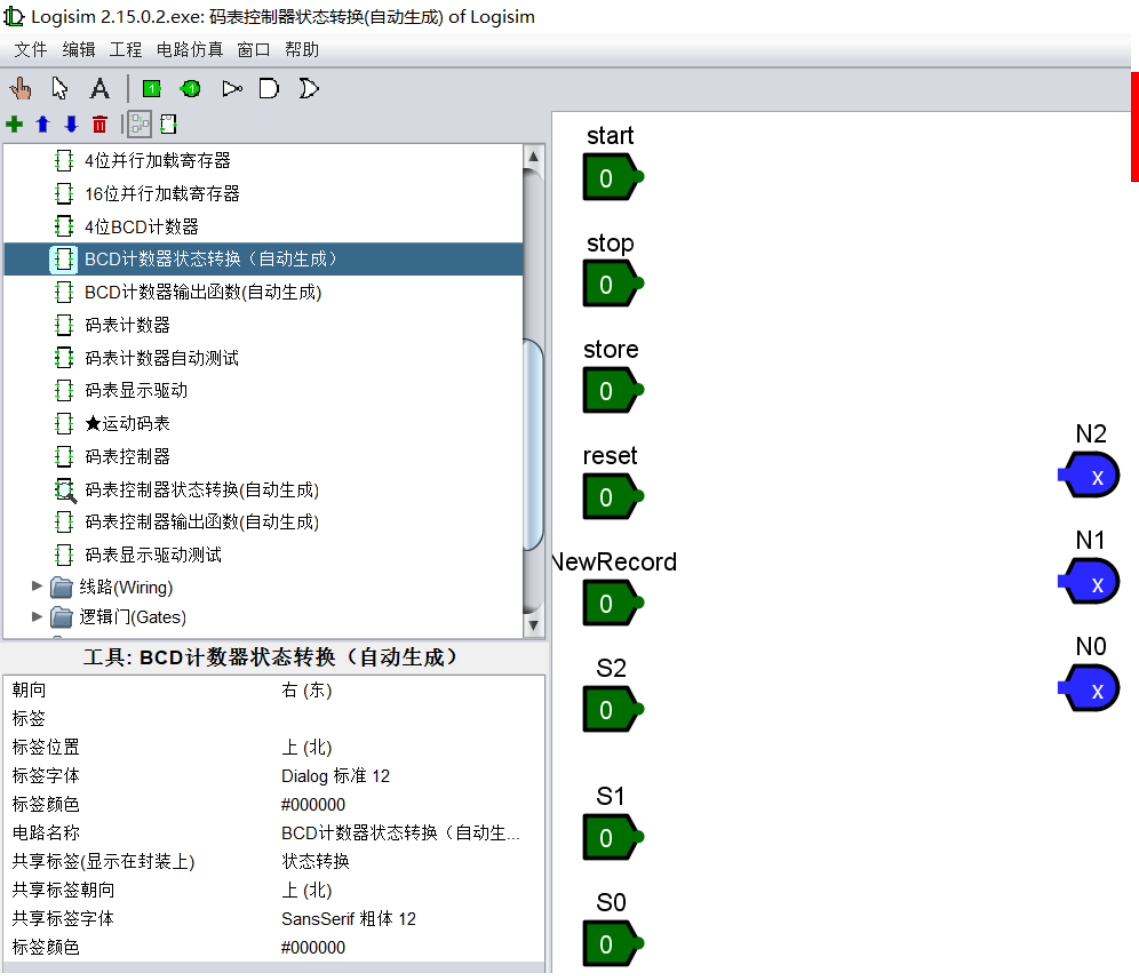


封装外观

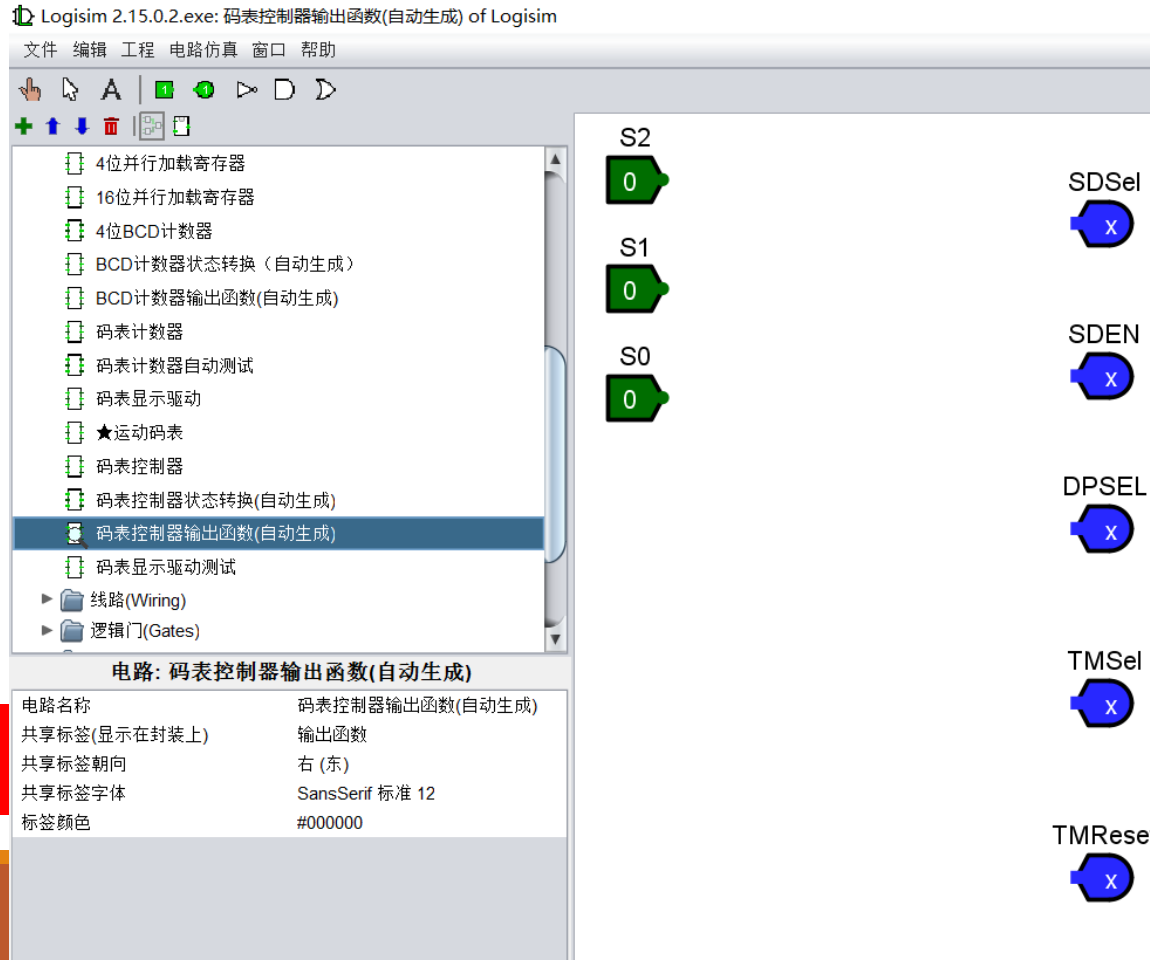


电路编辑界面

“码表控制状态转换”的设计



“码表控制输出函数”的设计



设计思路与方法

条件

（1）根据功能需求构建状态图

画出状态图（现态 -------> 次态）

- 运动码表状态定义（可自行设计、方案不唯一）

状态	状态编码 S2 S1 S0	主要功能
初始（000）		TM.Q = 00.00, SD.Q = 99.99
从0计数		TM从0开始计数
停止		显示TM.Q
存储		当产生NewRecord时， TM.Q -->SD.Q

- 状态转移的输入条件

外部输入控制信号：**Start、Stop、Store、Reset**

内部反馈的状态信号：**NewRecord**

（2）填写状态转换表

Tip：借助Excel表格，不需要的输入条件不用填写

（下图仅为部分状态转移内容）

当前状态(现态)					输入信号								下一状态（次态）				
S3	S2	S1	S0	现态10进制	NewRecord	start	stop	store	reset	In6	In7	In8	次态10进制	N3	N2	N1	N0
0	0	0	0	0		0	0	0	0				0	0	0	0	0
0	0	0	0	0		1	0	0	0				1	0	0	0	1
0	0	0	0	0		0	0	0	1				0	0	0	0	0

注意：所有可能出现的转移条件均要考虑。不要遗漏保持当前状态不变的情况

（3）生成状态转换逻辑表达式，进而利用Logisim电路分析功能，完成“码表控制状态转换”子电路

- 利用戳工具简单测试所有的状态转换情况，必要时通过真值表微调电路

（4）填写输出函数真值表

- 输入

  控制器现态： S2 S1 S0

- 输出

  送往数据通路的控制信号：

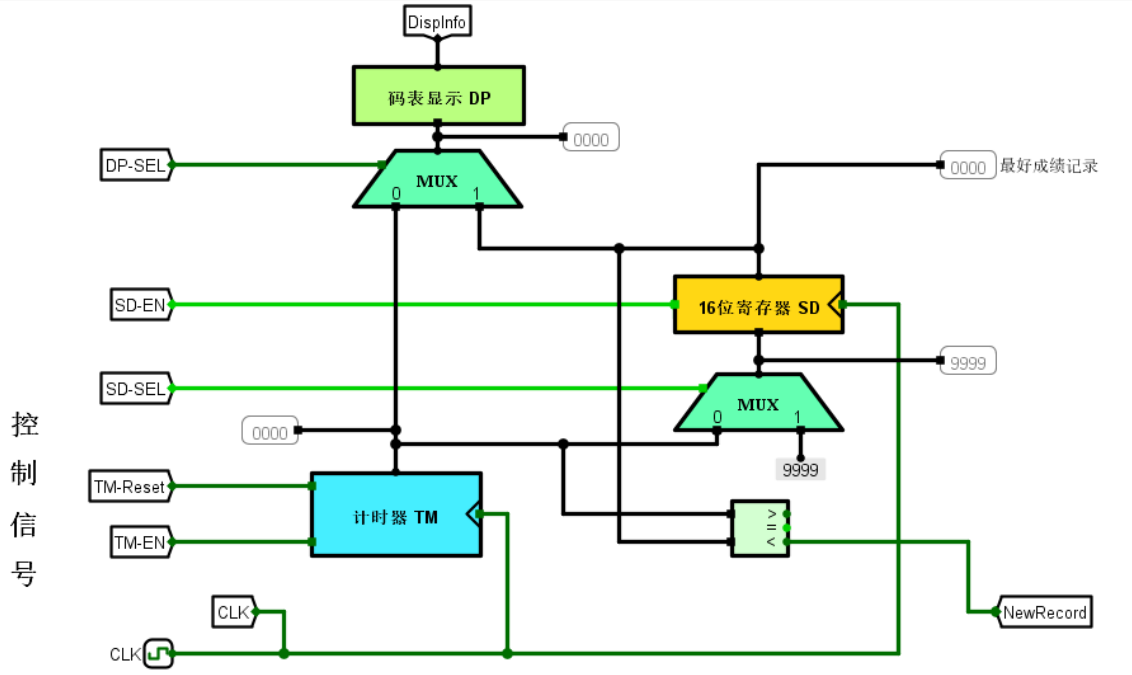
SD-SEL、SD-EN 、TM-EN、

TM-Rest、 DP-SEL

- 参照已完成的数据通路图，确定每个状态需要产生的输出控制信号

简易真值表举例

现态	数据通路变化	需要产生的内部控制信号
初始（000）	停止计数、TM.Q = 00.00, SD.Q = 99.99 显示TM.Q（默认选择）	~TM-EN、TM-Rest SD-SEL、SD-EN ~DP-SEL
计数		
停止		
存储		



数据通路图举例

➢ 自行改建出供码表设计使用的EXCEL，填写真值表

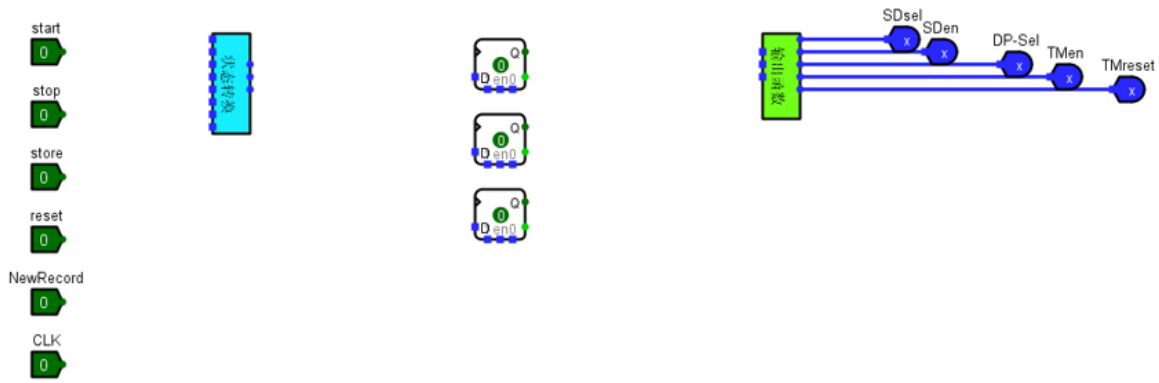
当前状态(现态)					输入信号												
S3	S2	S1	S0	现态10进制	NewRecord	start	stop	store	reset	In6	In7	In8	SDSel	SDEN	DPSEL	TMSel	TMReset
0	0	0	0	0									1	1			1

（图中仅为部分真值表内容）

（5）生成输出函数逻辑表达式，进而利用Logisim电路分析功能，“码表控制输出函数”子电路

➢ 利用戳工具简单测试所有整体情况

（6）完成“码表控制器”的内部逻辑的连接。

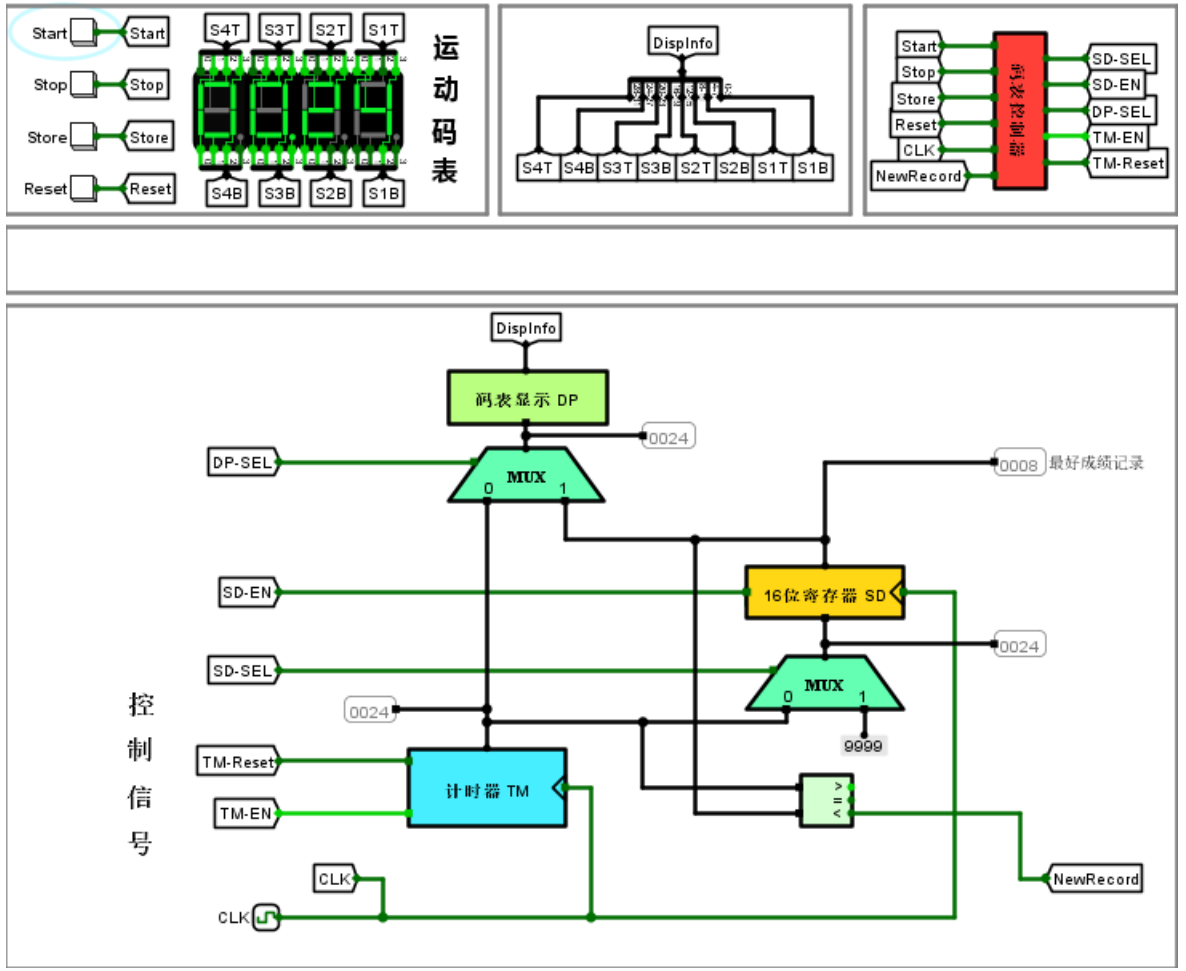


可以根据个人设计需要，适当调整原给定的部分连接线路

5、运动码表系统集成与联调

连接控制单元与数据通路

- 对照功能部件分析表、部件输入来源表，完成控制单元与数据通路各个部件的引脚连接
- 可适当添加与调整基本部件、门电路、线路等



运动码表功能测试

- 适当添加探针，便于观察
- 发现错误后，再调试、修改各个部分

实验报告要求——运动码表设计

1、撰写实验报告。报告按照数字系统设计的流程进行组织编写；需将多个分解实验的分析、设计思路与过程清晰描述（包括真值表、状态定义、状态转移、输出函数等），并相应贴出测试截图。

2、实验完成后，将最终的“Logisim.circ”文件以“学号_数字码表.circ”命名另存。

3、在福大课程中心平台中，本门课程的“作业”中，相应提交实验报告与电路文件。