



中山大學 软件工程学院
SUN YAT-SEN UNIVERSITY SCHOOL OF SOFTWARE ENGINEERING

计算机组成原理实验

授课老师：吴炜滨

大纲



- 课程信息
- 课程导学
- 实验环境设置
- Logisim介绍

大纲



➤ 课程信息

■ 授课教师：吴炜滨

- 主要研究方向：可信人工智能、深度学习、计算机视觉、自然语言处理、智能软件工程等，重点关注深度学习的**可靠性、安全性、可解释性与隐私性**
- 邮箱：wuwb36@mail.sysu.edu.cn
- 主页： <https://sse.sysu.edu.cn/teacher/249>
- **欢迎同学们加入我的本科科研小组！**



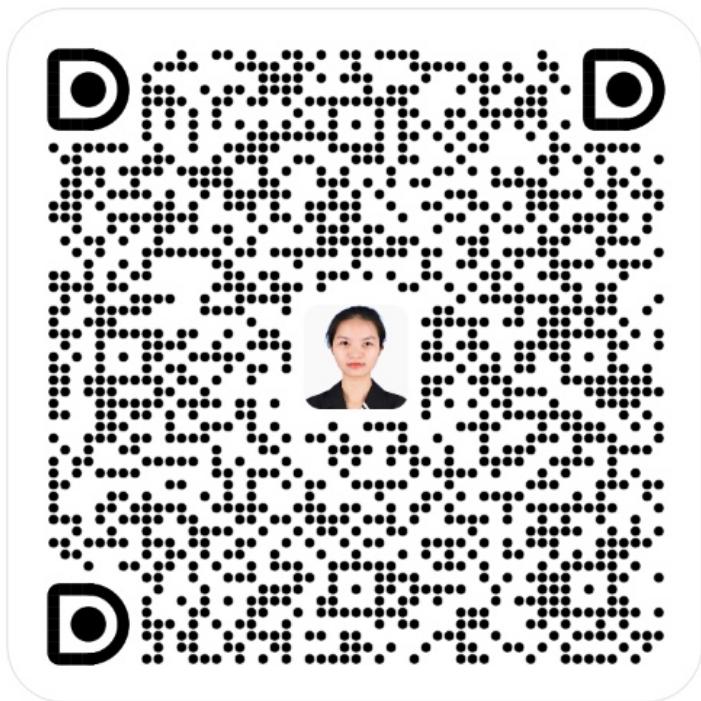
主页

课程信息



■ 助教：冼幸曦

- 办公室：瀚林一号B301
- 邮箱：xianxx3@mail2.sysu.edu.cn



钉钉



微信

课程信息



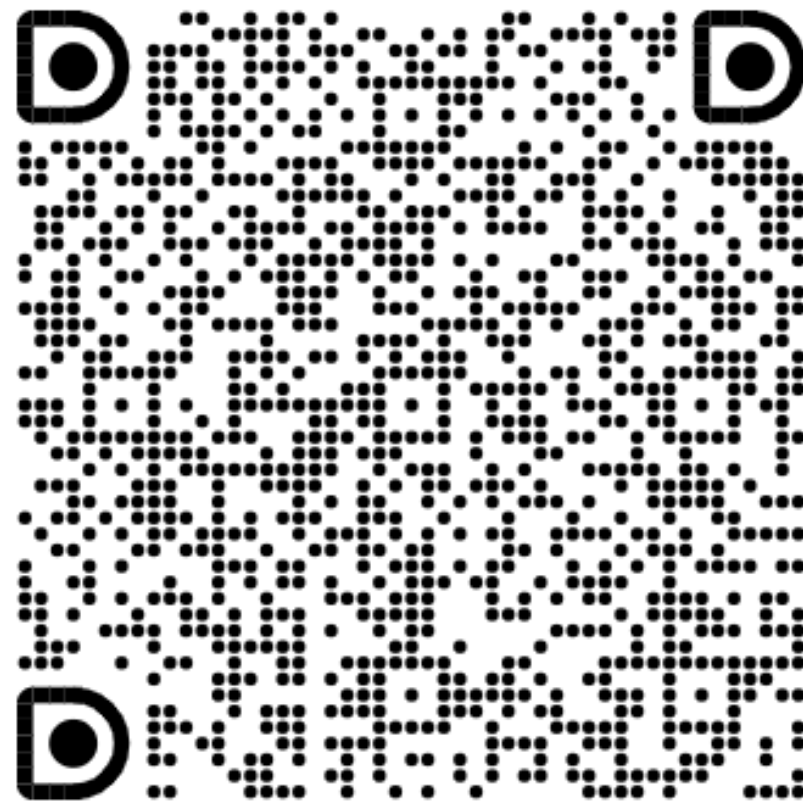
■ 课程通知群

- 微信群
- **钉钉群**

Group: 2025 计算机组成原理



2025计算机组成原理



■ 上课时间地点

- 教学班A：每周四（5-6节），教学大楼A309
- 教学班B：每周四（7-8节），教学大楼A310

■ 参考书

- 课件

■ 总成绩评定方法

- 平时成绩占总成绩的70%，期末考核占总成绩的30%

■ 平时成绩构成

- 出勤占平时成绩的10%：不少于2次考勤，每次考勤分数占比相同
- 实验项目占平时成绩的90%：按3个主要实验项目的完成情况分别给出分值，每个主要实验项目的分数占比相同
 - 按照模板撰写实验报告 + 电路文件

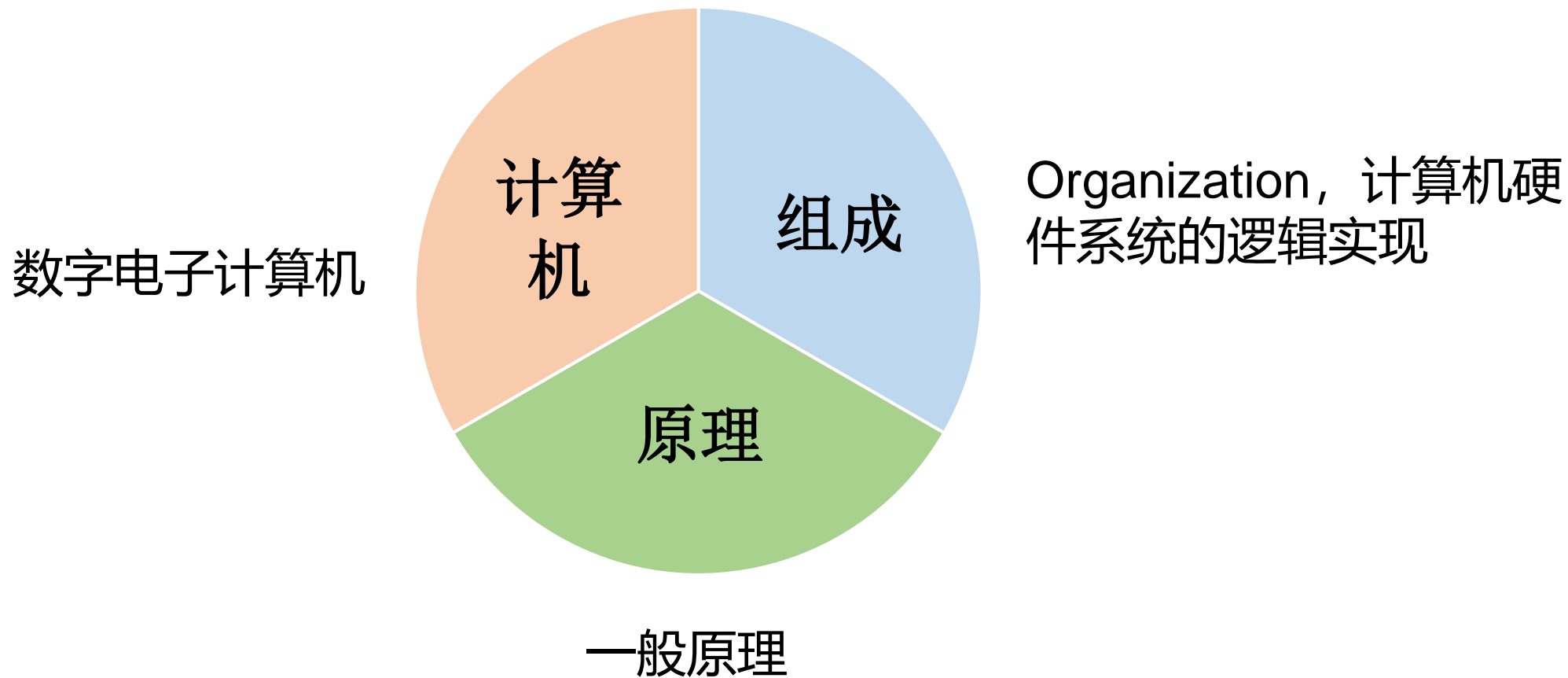
■ 期末考核

- 一个综合实验项目，按该综合实验项目的完成情况给出分值
 - 按照模板撰写实验报告 + 电路文件

大纲



➤ 课程导学



■ 实验内容

- 基本运算的操作原理
- 基本部件的结构和组织方式
- 基本部件和单元的设计思想
- 互连构成整机的方法

■ 特色

- ~~计算机组成的一般原理，不以具体机型为依托~~ 以MIPS处理器的实现为例
- ~~采用自顶向下的方式、层层细化~~ 采用自底向上的方式、层层递进



■ 课程简介与实验环境

- 课程简介
- Logisim使用介绍
- Logisim入门实验

■ 数字逻辑基础实验

- 组合逻辑电路设计
- 同步时序电路设计



■ 数据表示实验

- 汉字编码实验
- 海明校验码设计实验

■ 运算器组成实验

- 可控加减法电路
- 快速加法器
- 乘法器



■ 存储系统设计

- 存储扩展实验
- 寄存器堆设计
- RAM设计

■ CPU设计

- CPU数据通路
- 控制器设计
- 测试联调



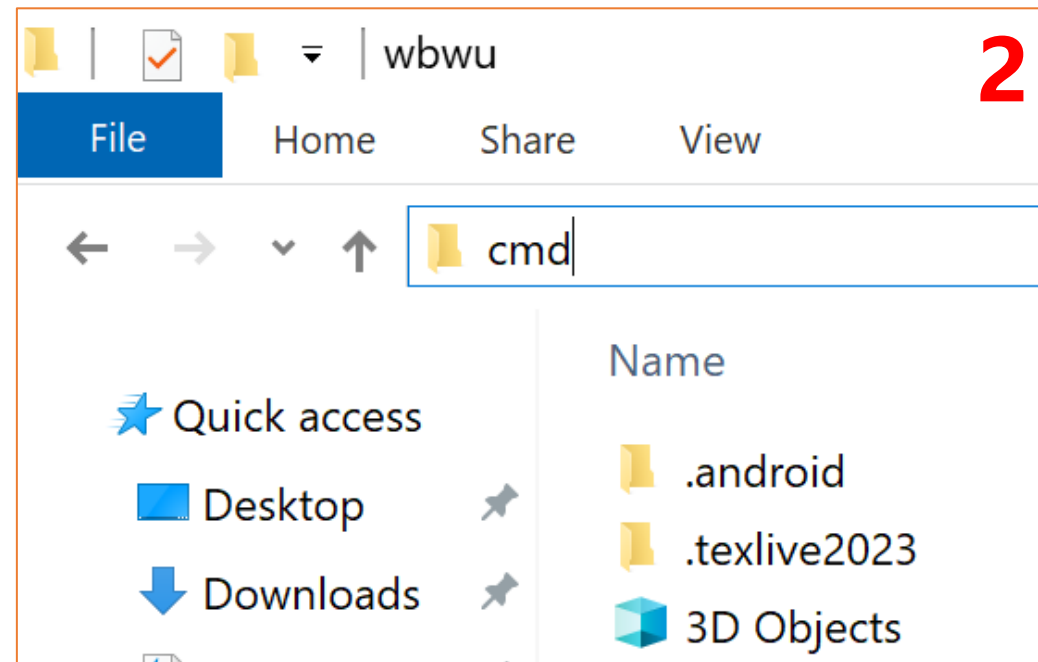
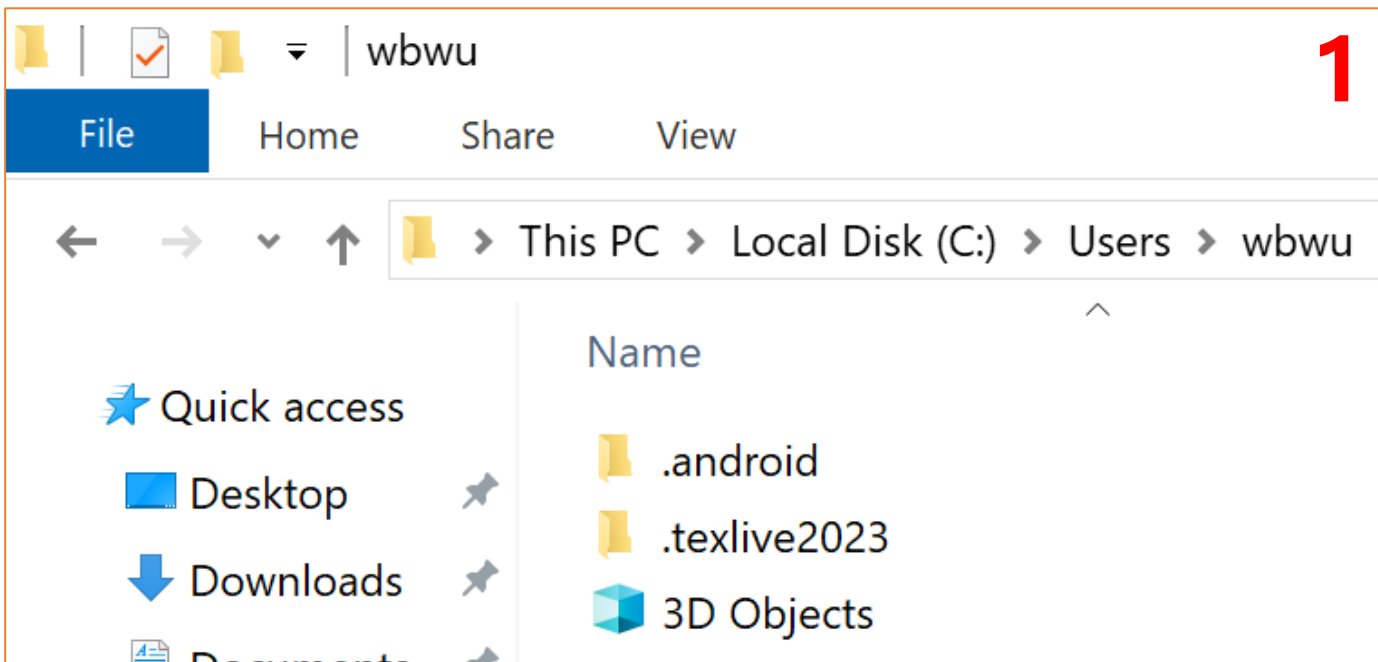
➤ 实验环境设置

JDK的安装



■ 测试JDK是否已经安装

- 进入命令行窗口：打开资源管理器，在地址栏输入 cmd 并回车



JDK的安装



■ 测试JDK是否已经安装

- 在窗口中输入命令 “java -version ” ， 回车
- 出现如下结果， 则说明JDK已经安装

```
Command Prompt
Microsoft Windows [Version 10.0.19045.3324]
(c) Microsoft Corporation. All rights reserved.

C:\Users\wbwu>java -version
java version "17.0.8" 2023-07-18 LTS
Java(TM) SE Runtime Environment (build 17.0.8+9-LTS-211)
Java HotSpot(TM) 64-Bit Server VM (build 17.0.8+9-LTS-211, mixed mode, sharing)

C:\Users\wbwu>
```

■ 测试JDK是否已经安装

- 如果出现如下错误，则需要安装JDK

 命令提示符

```
Microsoft Windows [版本 10.0.19043.1889]
(c) Microsoft Corporation。保留所有权利。

C:\Users\Samzi>java -version
'java' 不是内部或外部命令，也不是可运行的程序
或批处理文件。

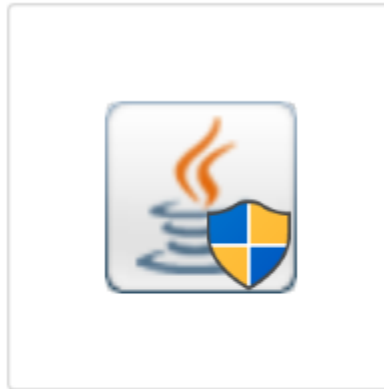
C:\Users\Samzi>
```

JDK的安装



■ 安装包

- 实验环境\Windows

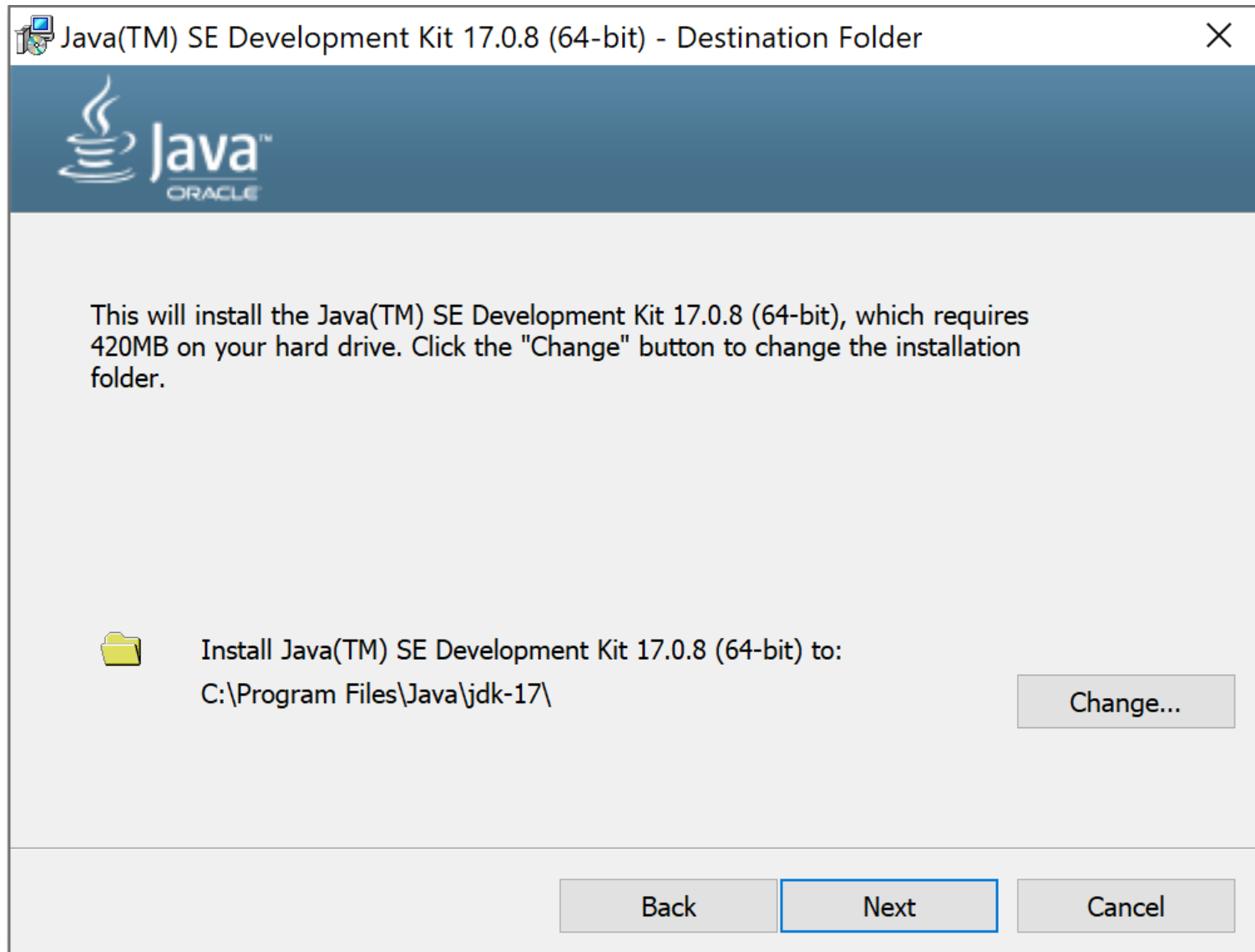


jdk-17_windows-
x64_bin.exe

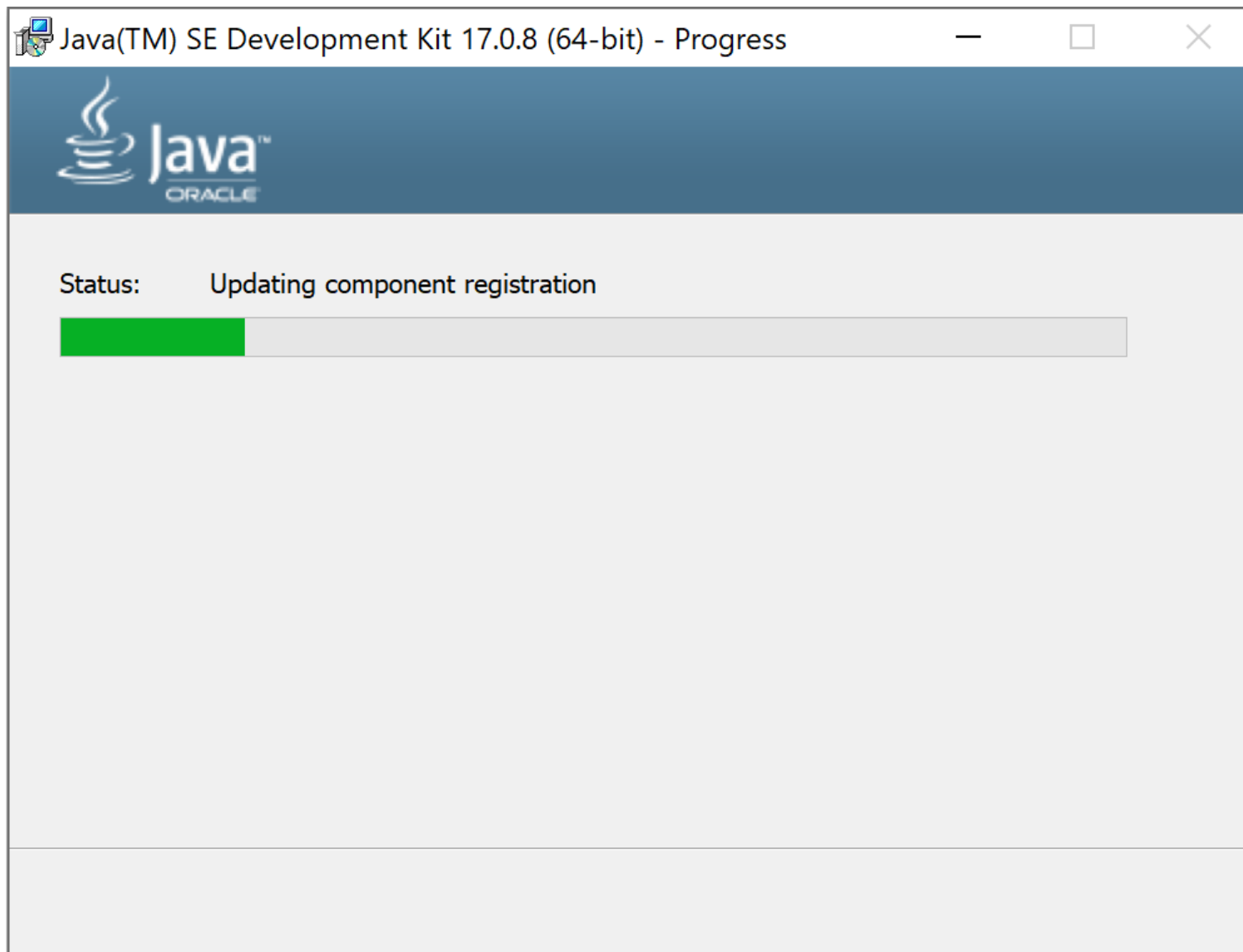
JDK的安装



JDK的安装



JDK的安装



JDK的安装

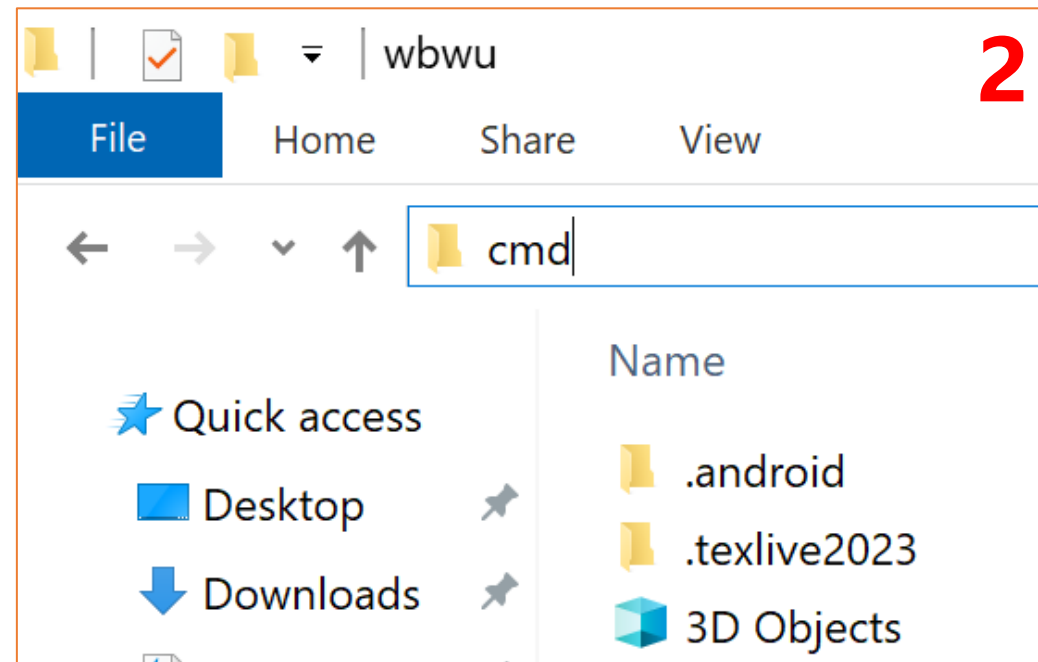
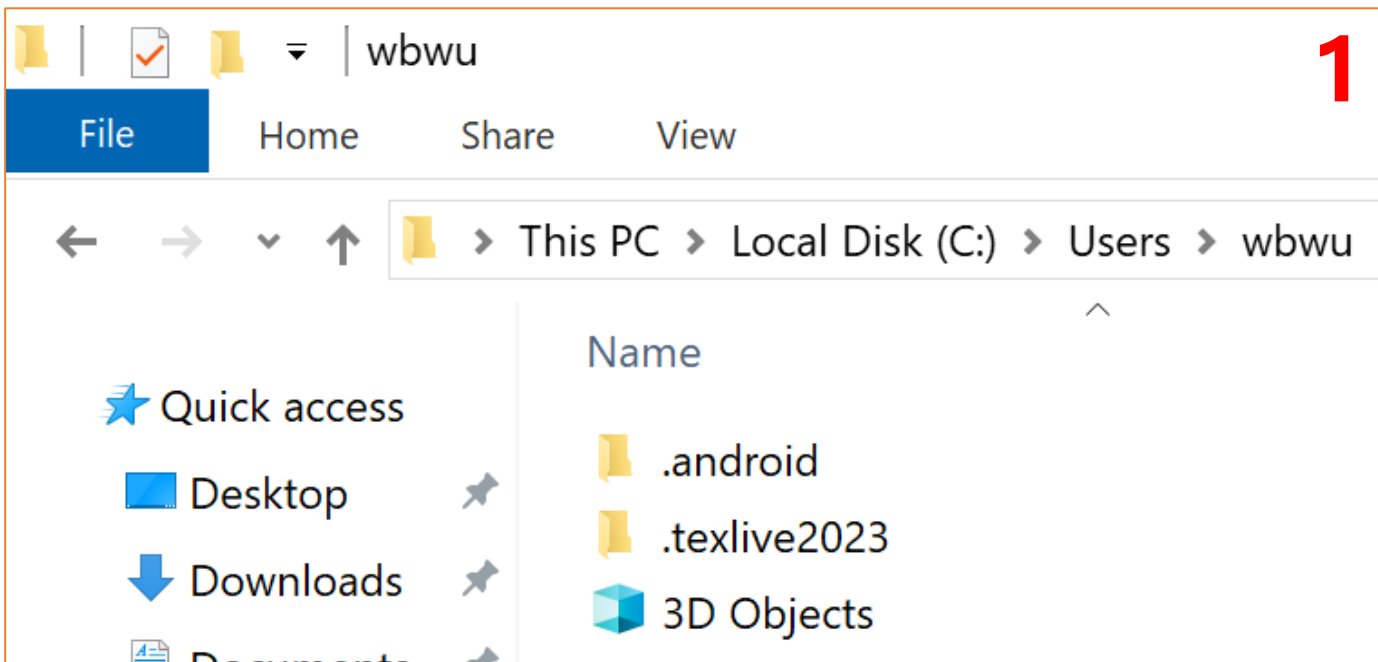


JDK的安装



■ 测试JDK是否安装成功

- 进入命令行窗口：打开资源管理器，在地址栏输入 cmd 并回车



JDK的安装



■ 测试JDK是否安装成功

- 在窗口中输入命令 “java -version ” ， 回车
- 出现如下结果， 则说明JDK安装成功

```
Command Prompt
Microsoft Windows [Version 10.0.19045.3324]
(c) Microsoft Corporation. All rights reserved.

C:\Users\wbwu>java -version
java version "17.0.8" 2023-07-18 LTS
Java(TM) SE Runtime Environment (build 17.0.8+9-LTS-211)
Java HotSpot(TM) 64-Bit Server VM (build 17.0.8+9-LTS-211, mixed mode, sharing)

C:\Users\wbwu>
```

■ 测试JDK是否安装成功

- 如果出现如下错误，则需要重新安装JDK或者配置环境变量Path

 命令提示符

```
Microsoft Windows [版本 10.0.19043.1889]
(c) Microsoft Corporation。保留所有权利。

C:\Users\Samzi>java -version
'java' 不是内部或外部命令，也不是可运行的程序
或批处理文件。

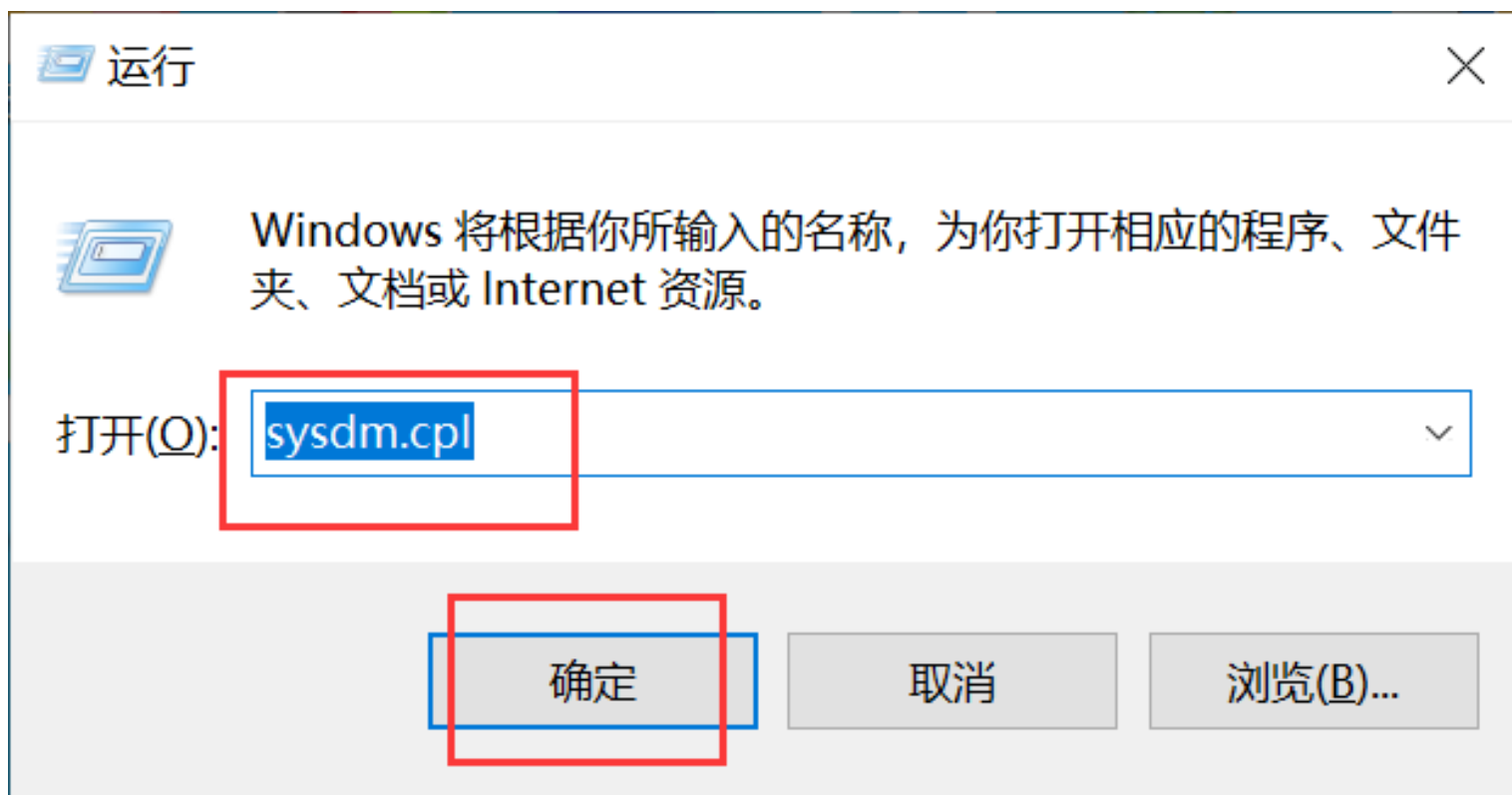
C:\Users\Samzi>
```

JDK的安装



■ 环境变量Path的配置步骤（非必须）

- 按下“win+R”快捷键，打开运行对话框，输入指令：sysdm.cpl



JDK的安装



■ 环境变量Path的配置步骤（非必须）

- 单击环境变量按钮

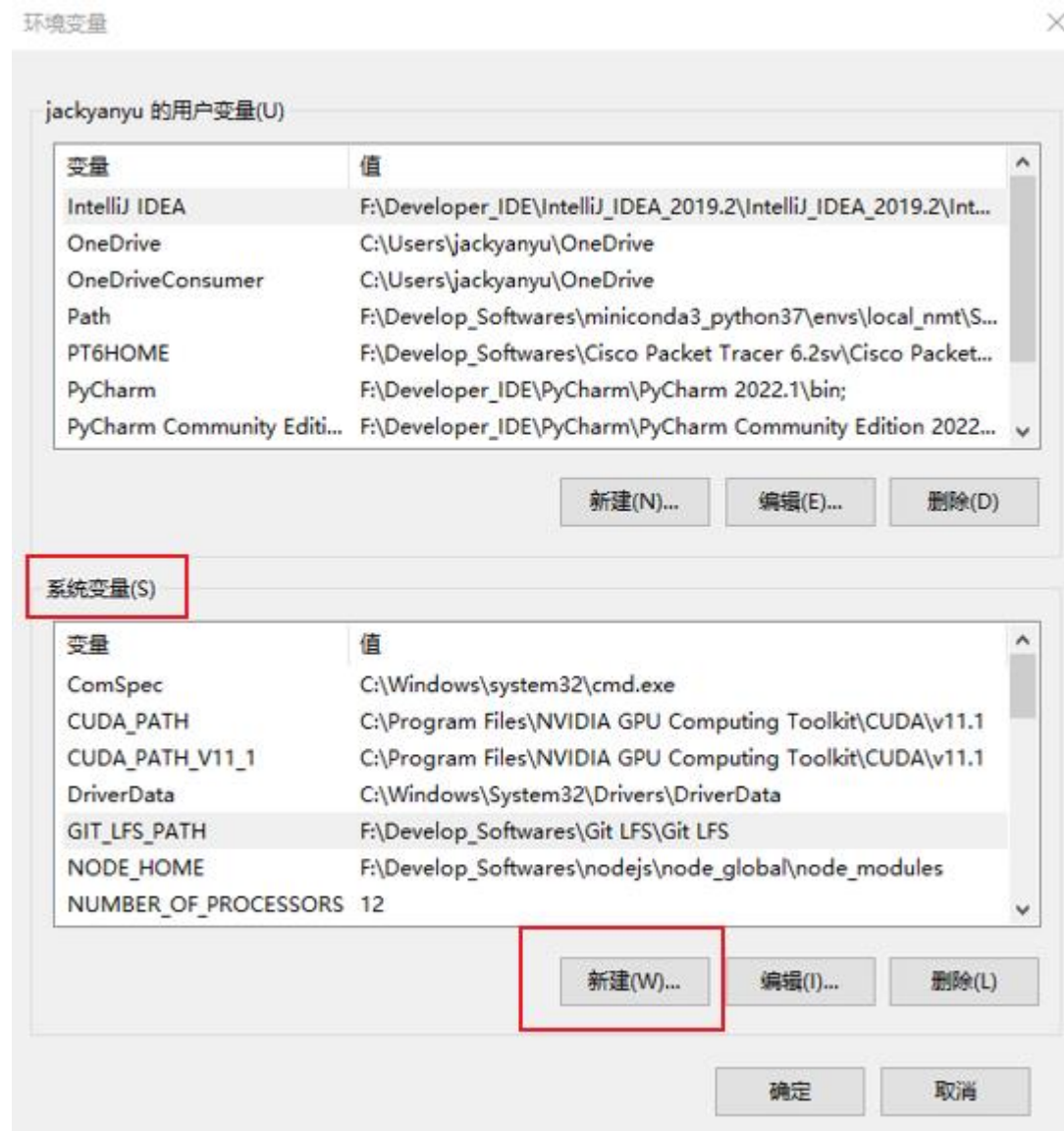


JDK的安装



■ 环境变量Path的配置步骤（非必须）

- 进入图示操作主界面
- 单击系统变量-->新建按钮

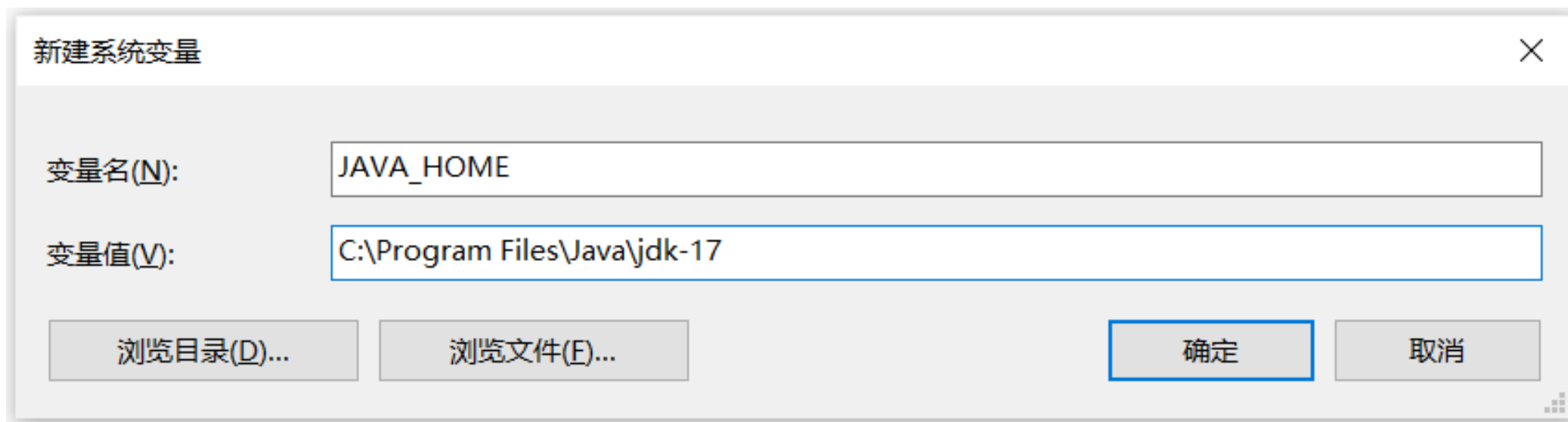


JDK的安装



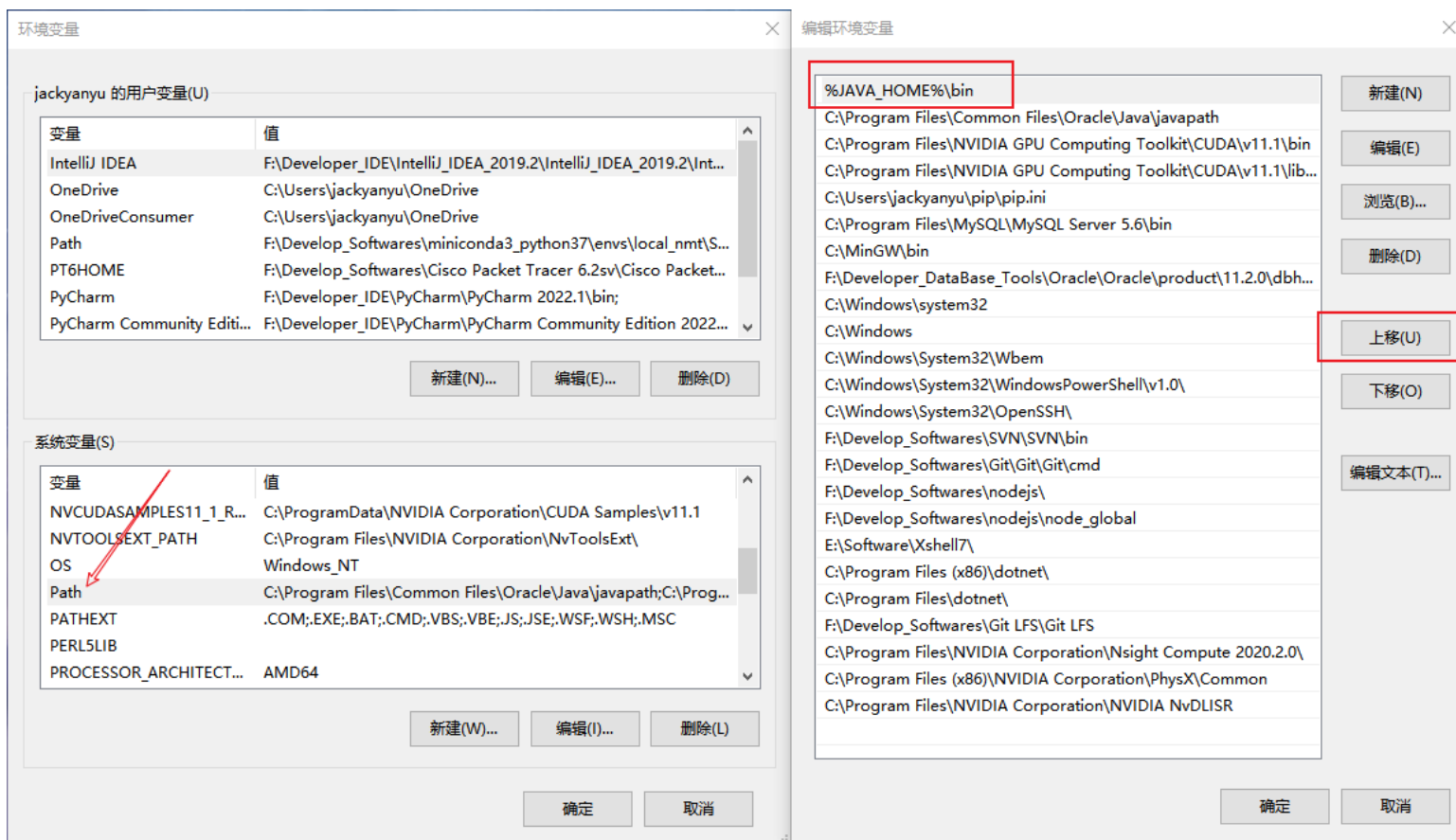
■ 环境变量Path的配置步骤（非必须）

- 新建JAVA_HOME变量（用于说明JDK的安装目录），点击浏览目录，找到刚刚安装JDK的目录，点击确定



■ 环境变量Path的配置步骤（非必须）

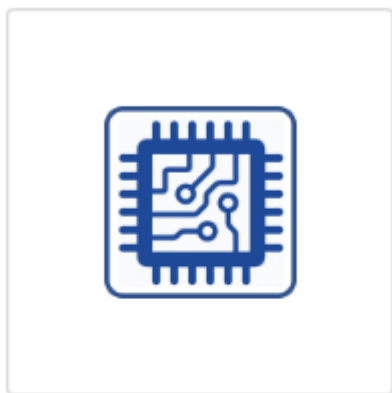
- 修改系统环境变量Path，新增%JAVA_HOME%\bin，并将其上移到第一个，然后点击确定



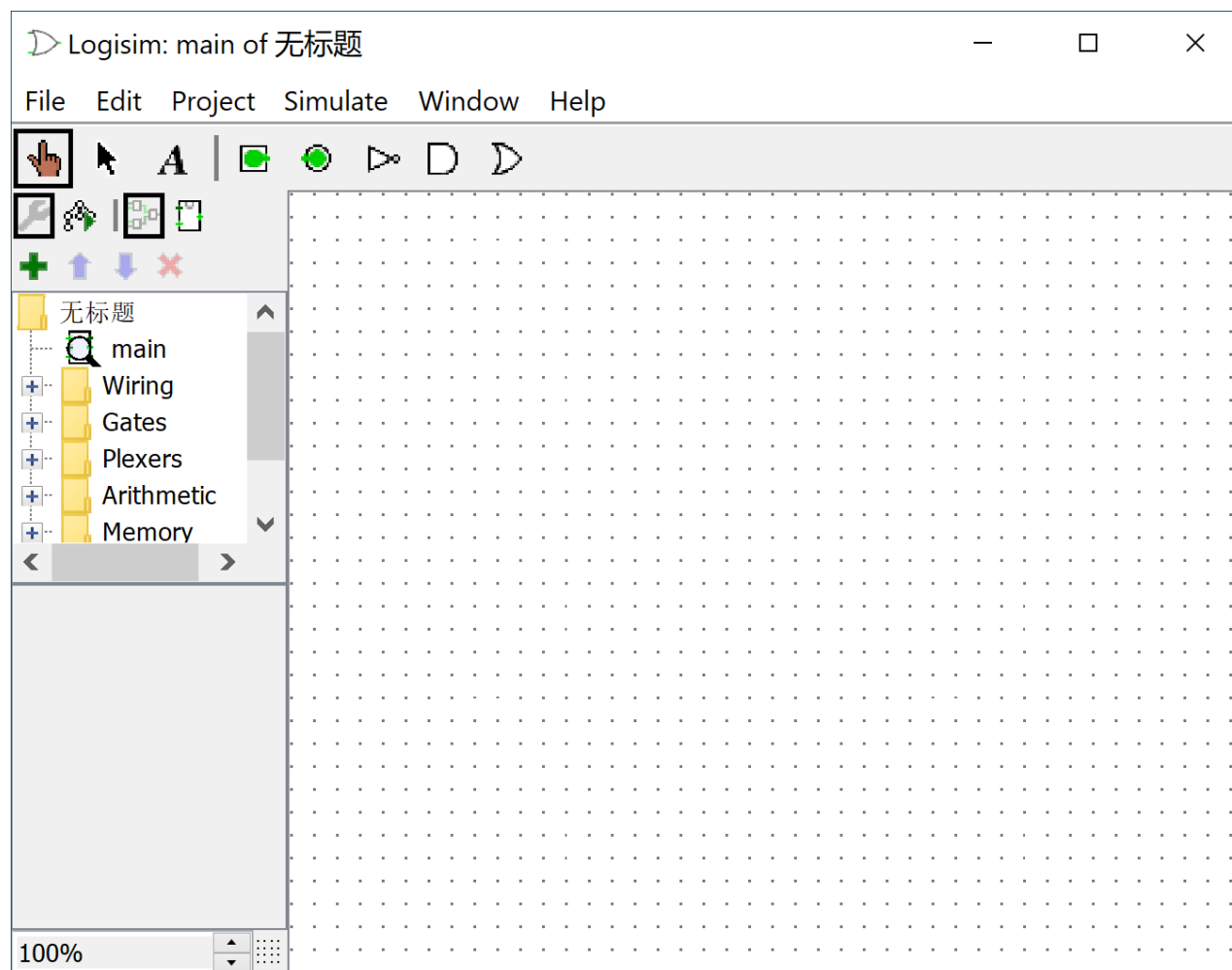
实验环境设置



- 直接双击logisim-win-2.7.1-cn.exe

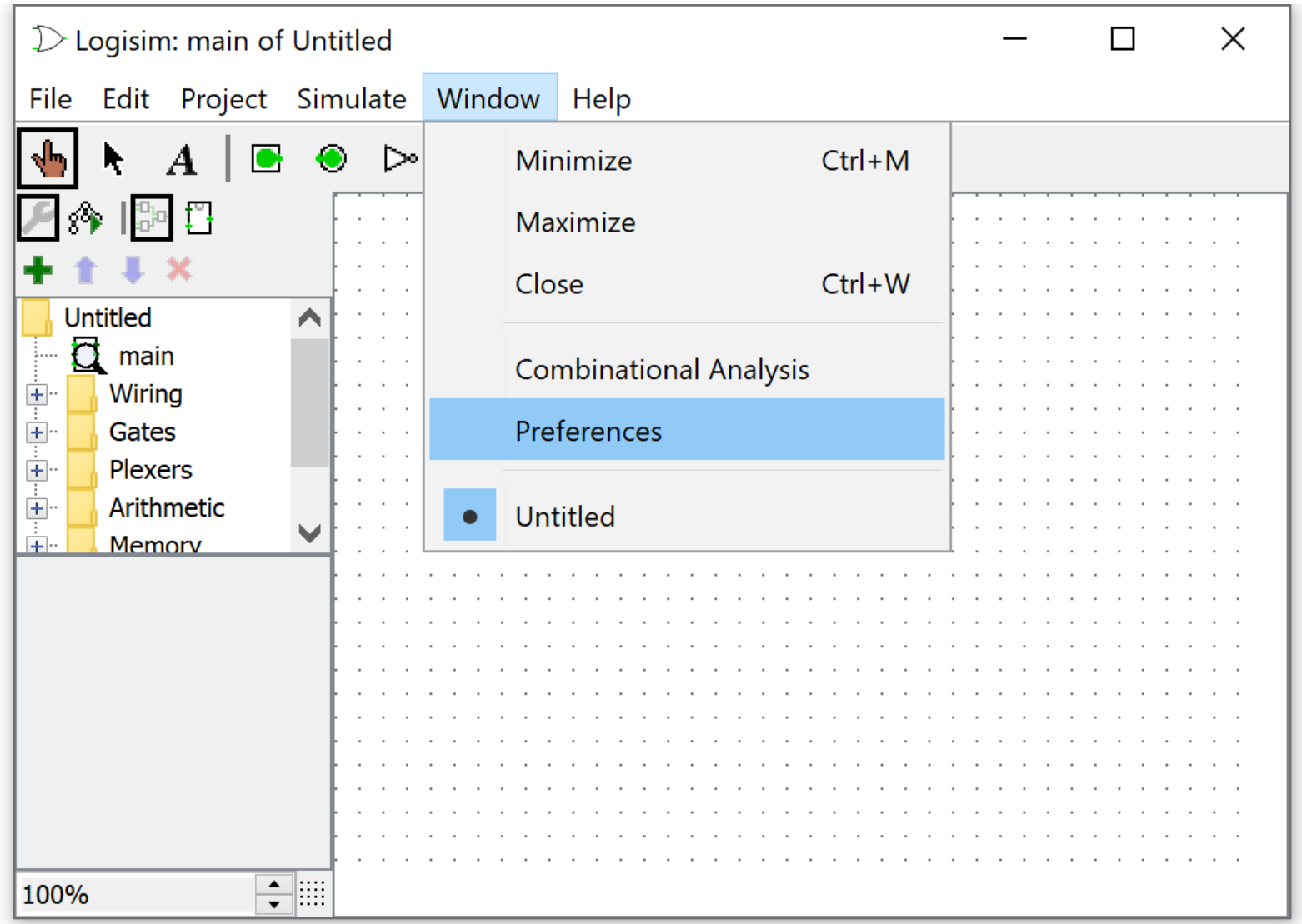


logisim-win-2.7.1
-cn.exe





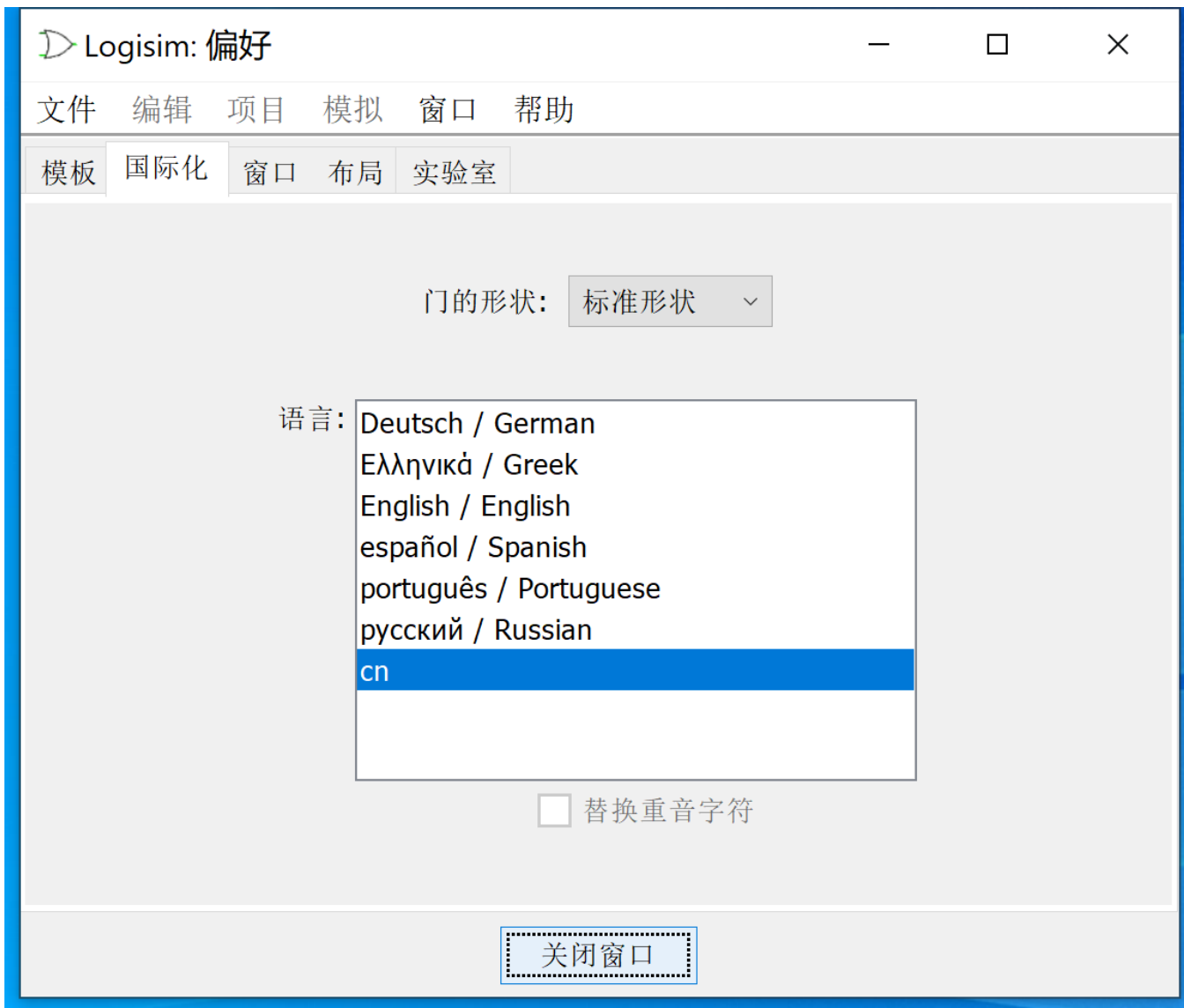
■ 更改语言



实验环境设置



■ 更改语言





➤ Logisim介绍

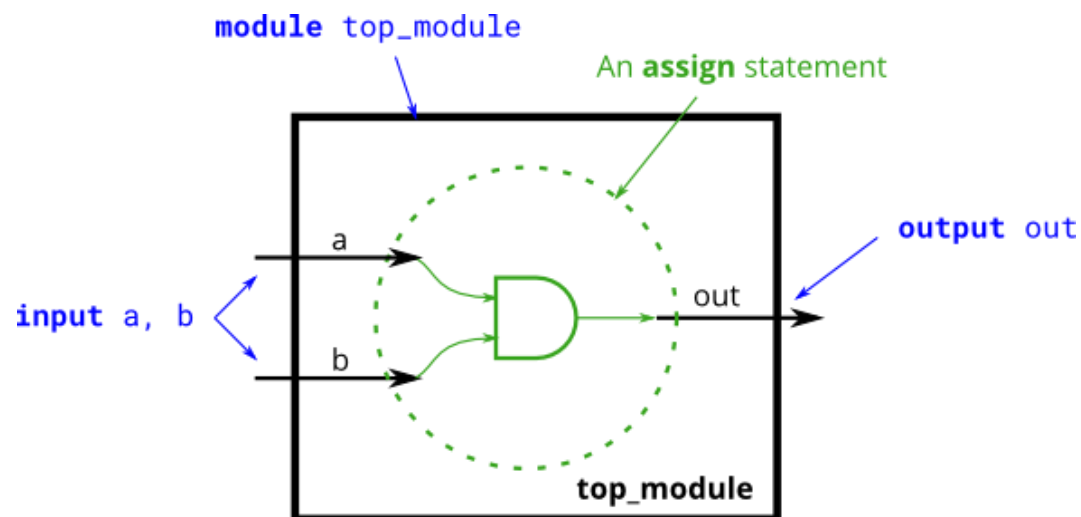
Logisim介绍



■ Logisim: 用于数字电路开发、CPU设计的仿真软件

■ 特点

- 无需硬件即可实验
- 实验平台简单易学，无需先导课程
 - 回避了硬件描述语言过于抽象、硬件设计程序化的问题
 - 利用绘制原理图的方式构建硬件电路，仿真直观，调试简单



Verilog代码

```
module top_module(  
    input a,  
    input b,  
    output out );  
    assign out = a & b;  
endmodule
```

Logisim介绍



■ Logisim: 用于数字电路开发、CPU设计的仿真软件

■ 特点

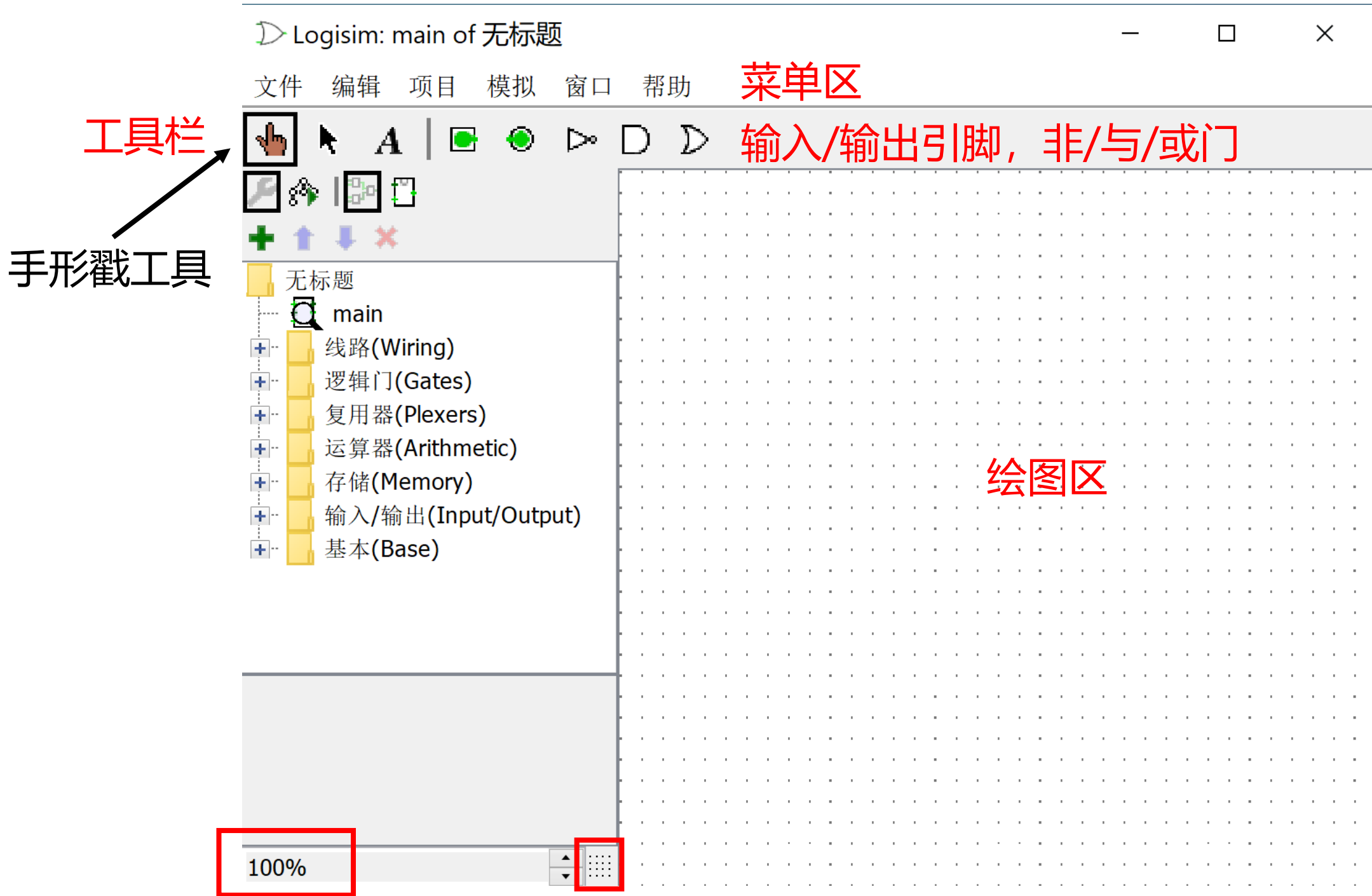
- 无需硬件即可实验
- 实验平台简单易学，无需先导课程
 - 回避了硬件描述语言过于抽象、硬件设计程序化的问题
 - 利用绘制原理图的方式构建硬件电路，仿真直观，调试简单
- 支持子电路封装，易于构建复杂数字电路系统（CPU）

Logisim介绍



■ Java程序，开源

- 目前已停更，GitHub上衍生出若干版本
- 高分辨率支持
- 有bug（大范围元件移动），但完全可用，重启即可



Logisim: main of 无标题

文件 编辑 项目 模拟 窗口 帮助



组件库

当前电路名字

子电路 (电路模块) 名字

管理窗

■ 实际电路如何搭建

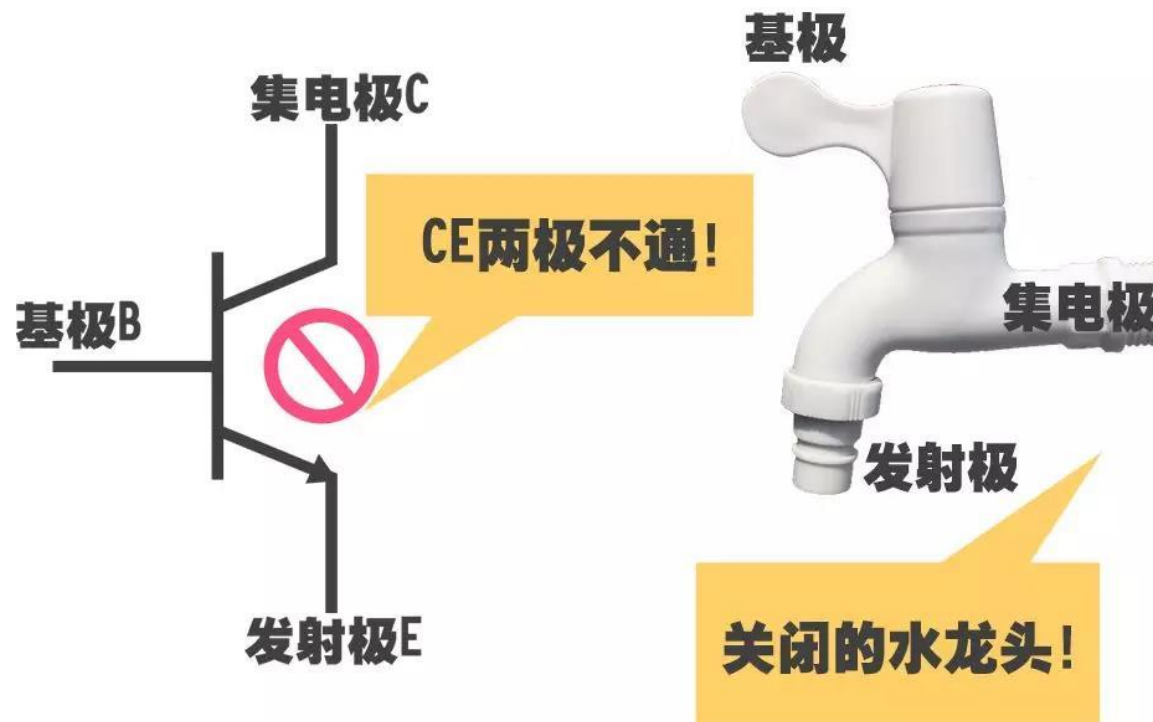
- 非门电路：可由晶体三极管组成

■ 晶体三极管的三种工作状态

- 截止状态
- 放大状态
- 饱和导通状态

■ 晶体三极管的三种工作状态

- 截止状态



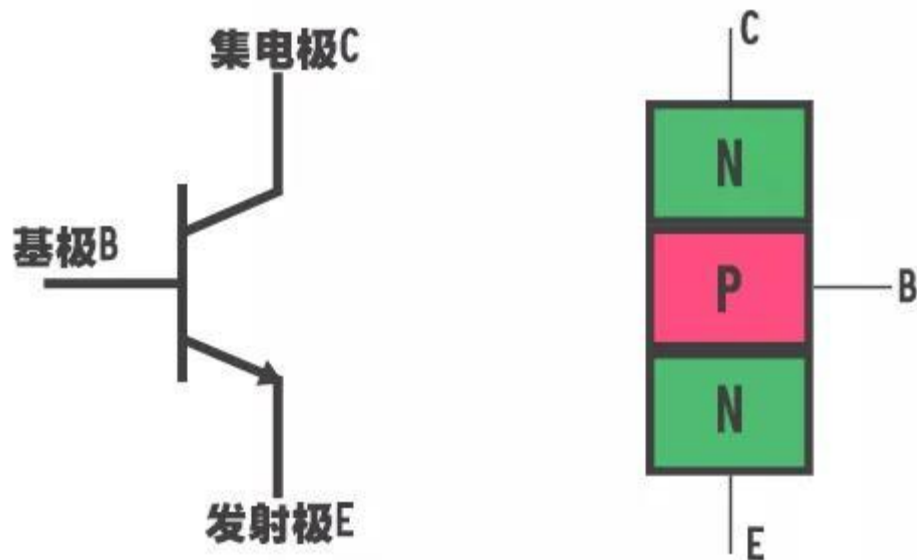
Logisim介绍



■ 晶体三极管的三种工作状态

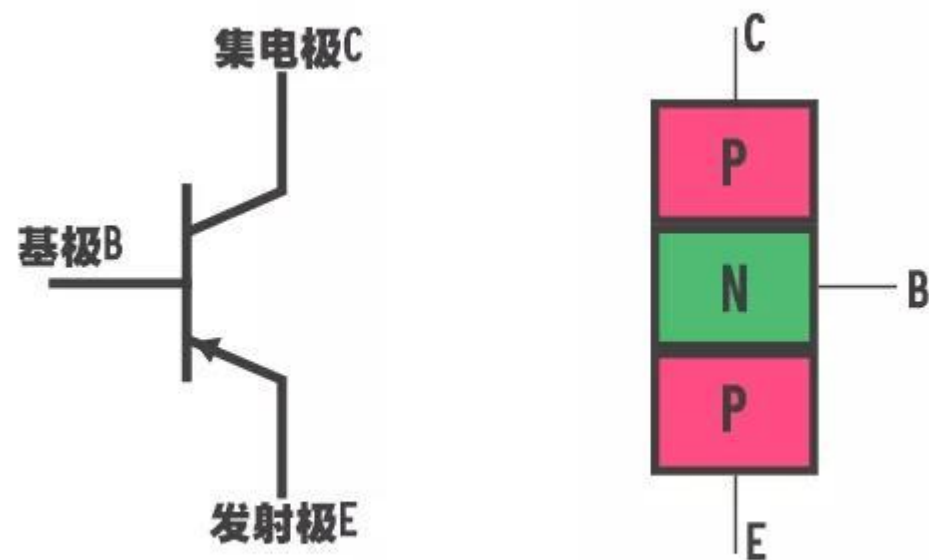
- 截止状态

对于NPN型三极管，



发射结、集电结反偏，即 $V_C > V_B, V_E > V_B$

对于PNP型三极管，



发射结、集电结反偏，即 $V_C < V_B, V_E < V_B$

■ 晶体三极管的三种工作状态

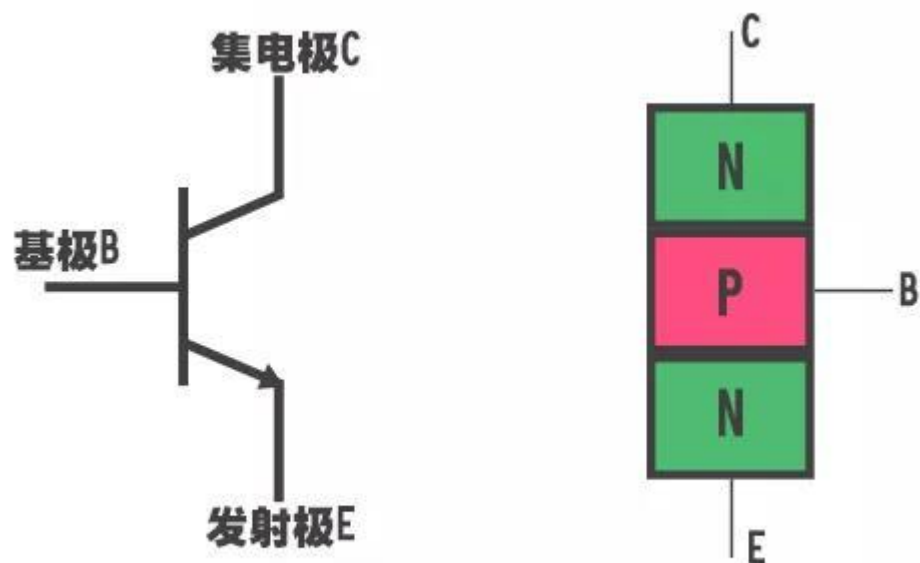
- 饱和导通状态



■ 晶体三极管的三种工作状态

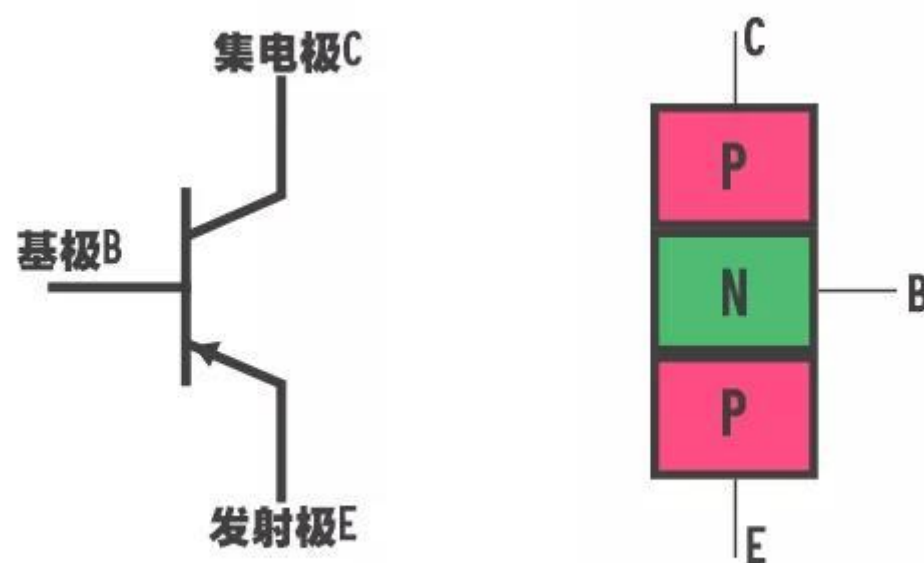
- 饱和导通状态

对于NPN型三极管，



发射结、集电结正偏，即 $V_B > V_C, V_B > V_E$

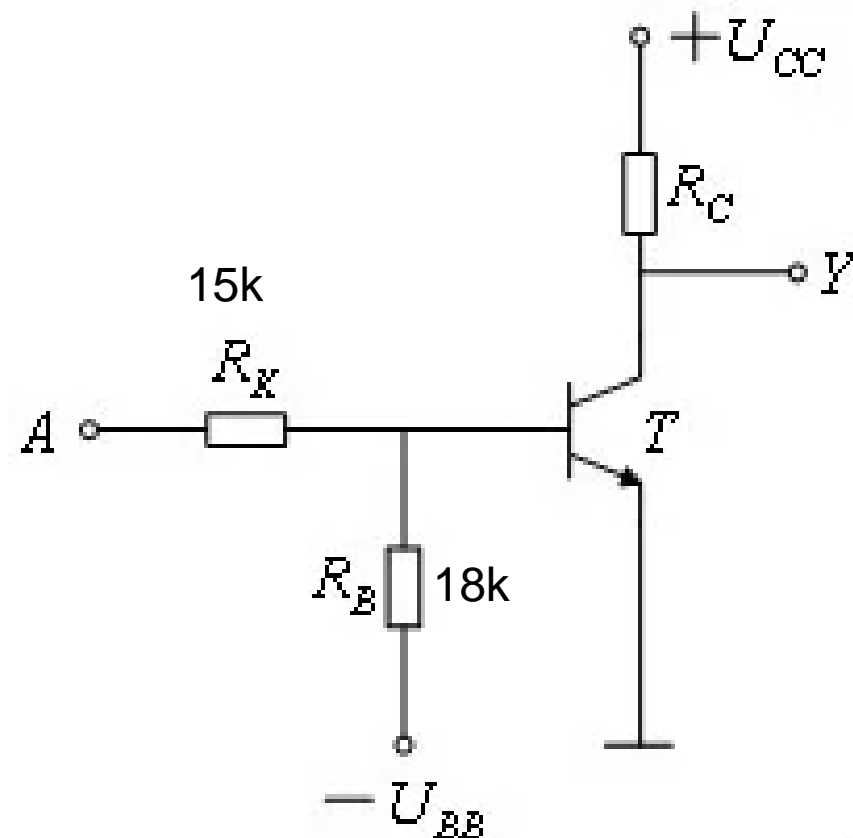
对于PNP型三极管，



发射结、集电结正偏，即 $V_C > V_B, V_E > V_B$

■ 由晶体三极管组成的非门电路

- 三极管的工作状态或从截止转为饱和，或从饱和转为截止
- “非” 门电路只有一个输入端A，当A为 “1” （设其电位为3V） 时，晶体管饱和，其集电极，即输出端Y为 “0” （其电位在零伏附近）
- 当A为 “0” 时，晶体管截止，输出端Y为 “1” （其电位近似等于 U_{CC} ）
- “非” 门电路也称为反相器

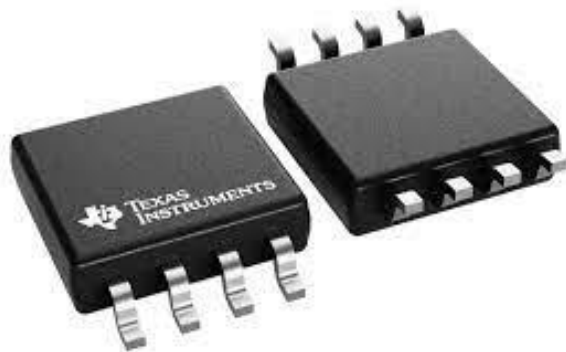


Logisim介绍

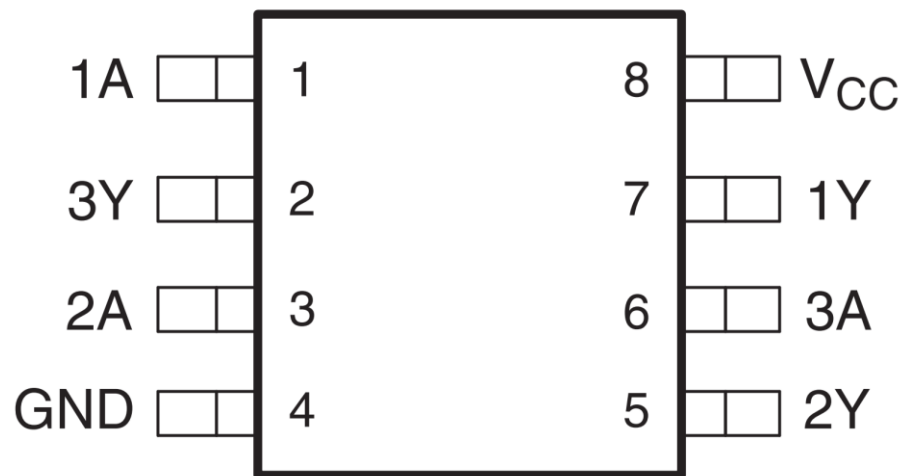


■ 芯片

- SN74LVC3G04



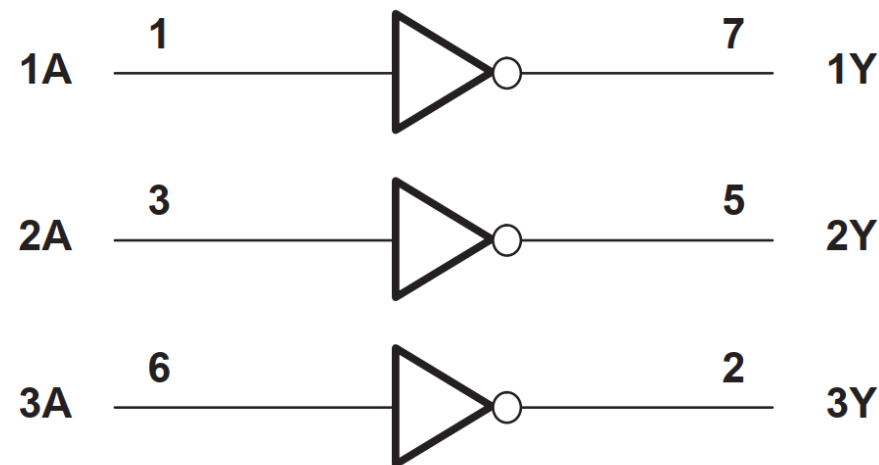
**DCT PACKAGE
(TOP VIEW)**



**Function Table
(Each Inverter)**

INPUT A	OUTPUT Y
H	L
L	H

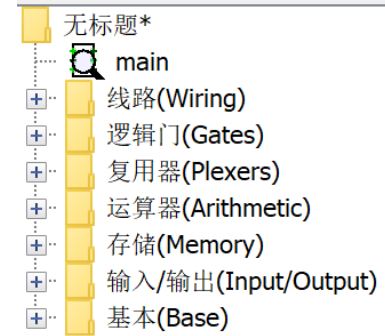
Logic Diagram (Positive Logic)





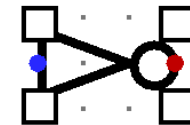
Logisim: main of 无标题

文件 编辑 项目 模拟 窗口 帮助



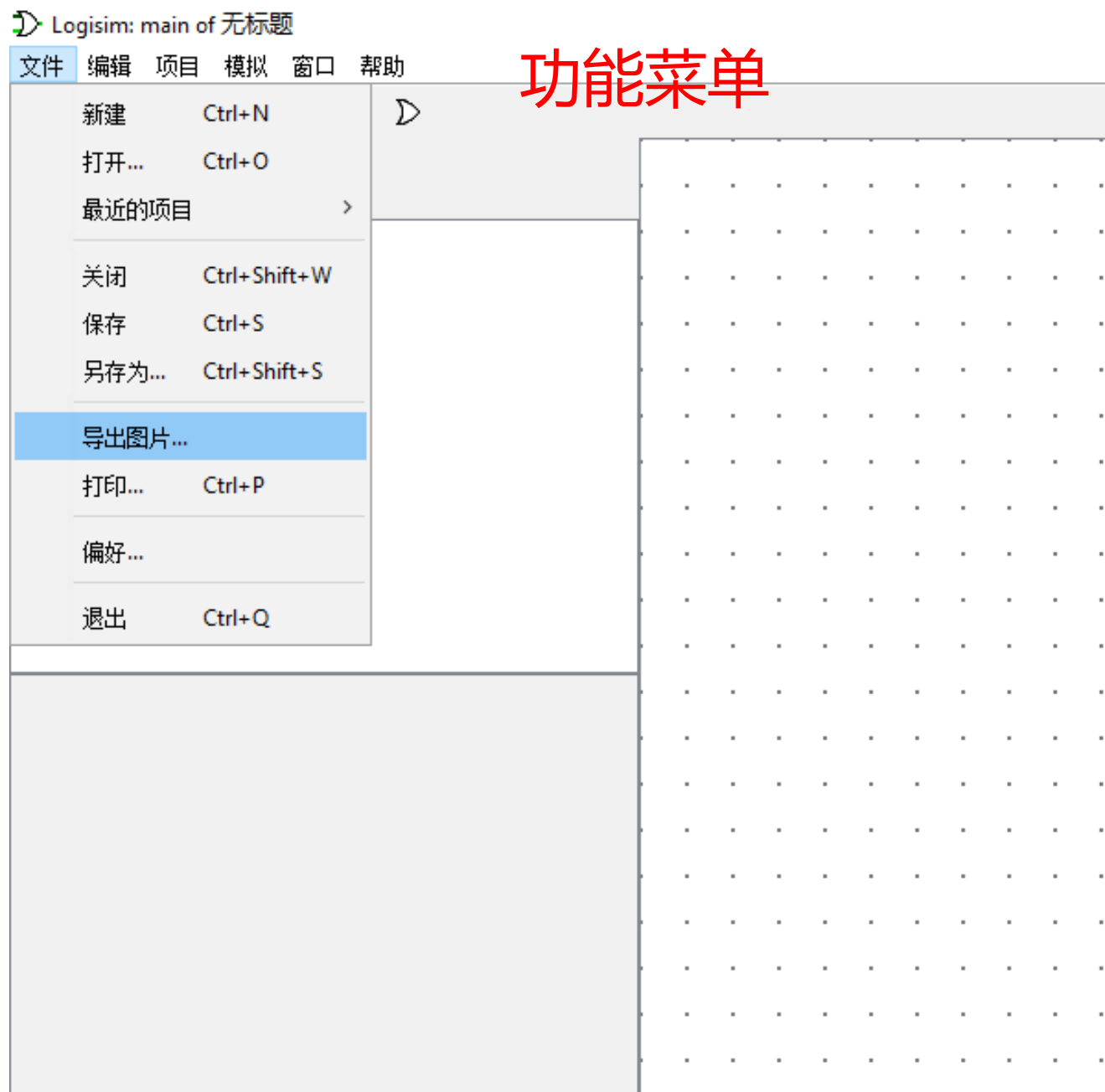
选区: 非门(NOT Gate)

朝向	东
数据位宽	1
门尺寸	宽
输出值	0/1
标签	
标签字体	SansSerif 标准 12



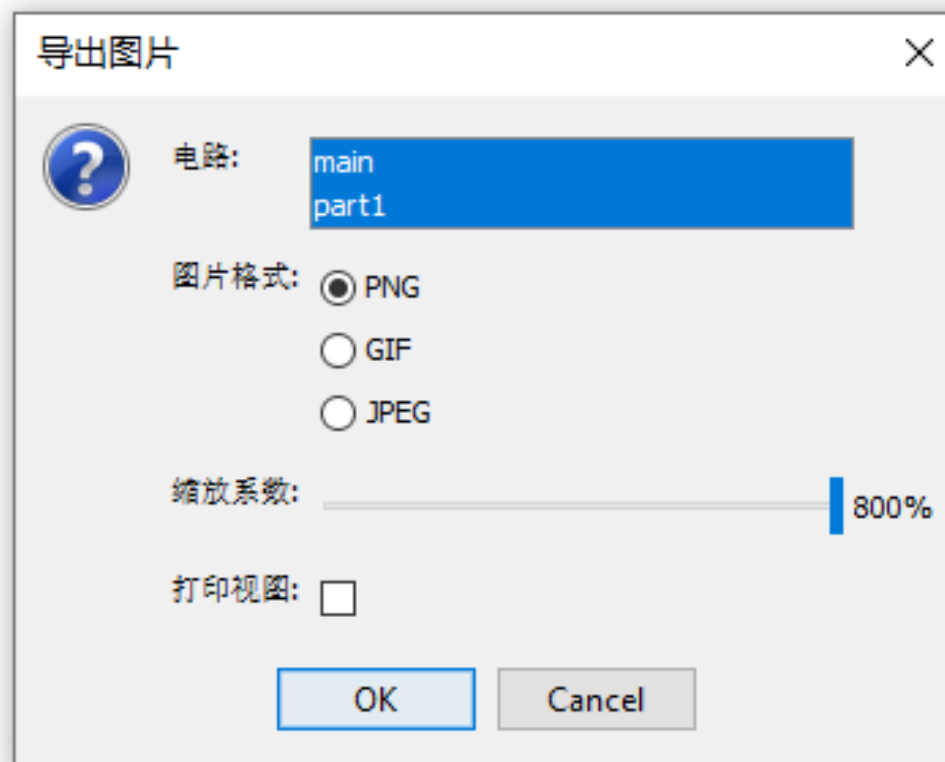
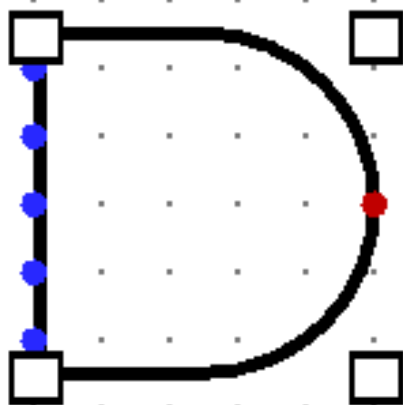
属性表:
设置组件属性 (朝向、位宽等)
, 很方便形成所需组件

■ 导出图片



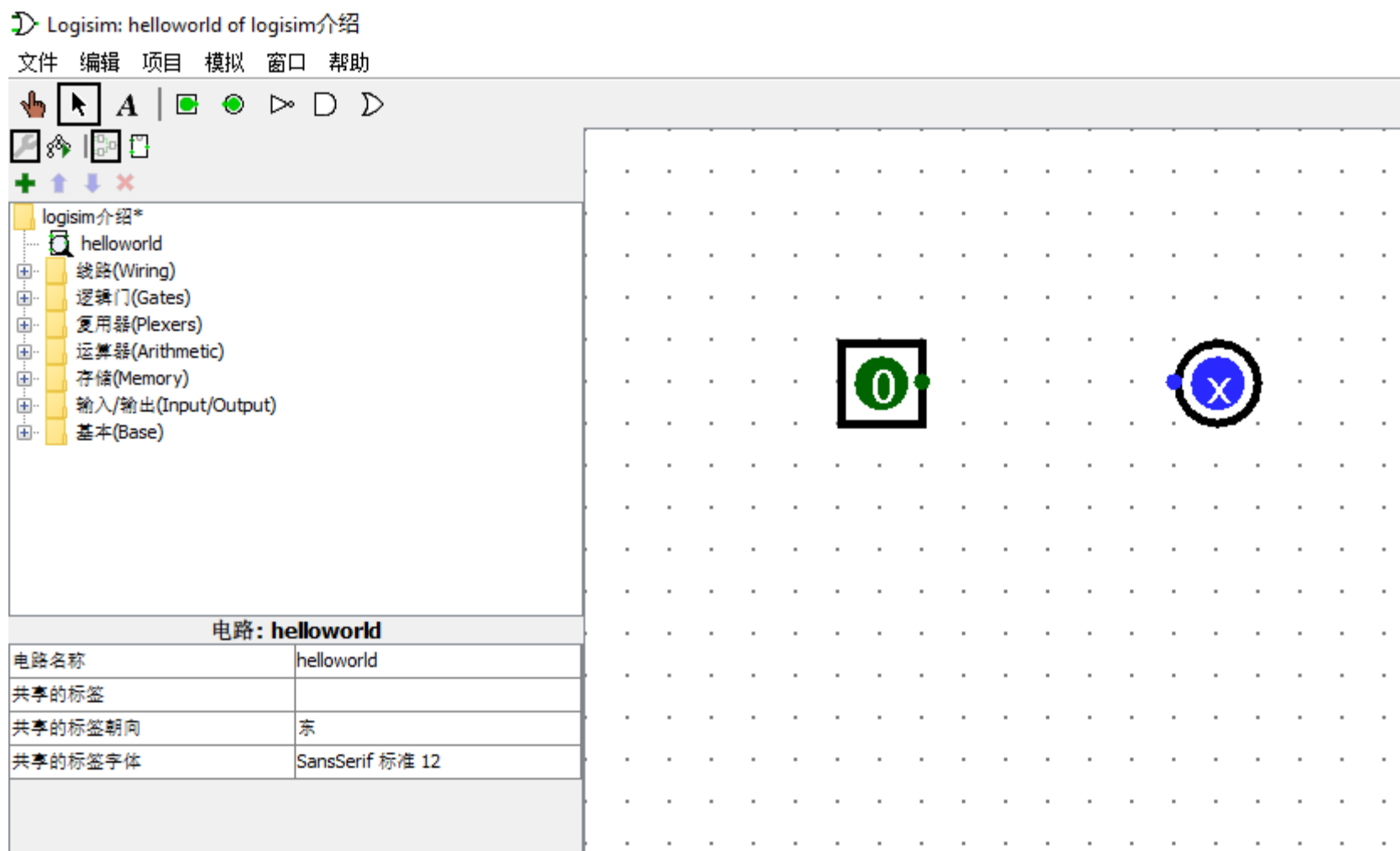
■ 导出图片

- 注意：PNG、缩放系数：800%、取消勾选打印视图



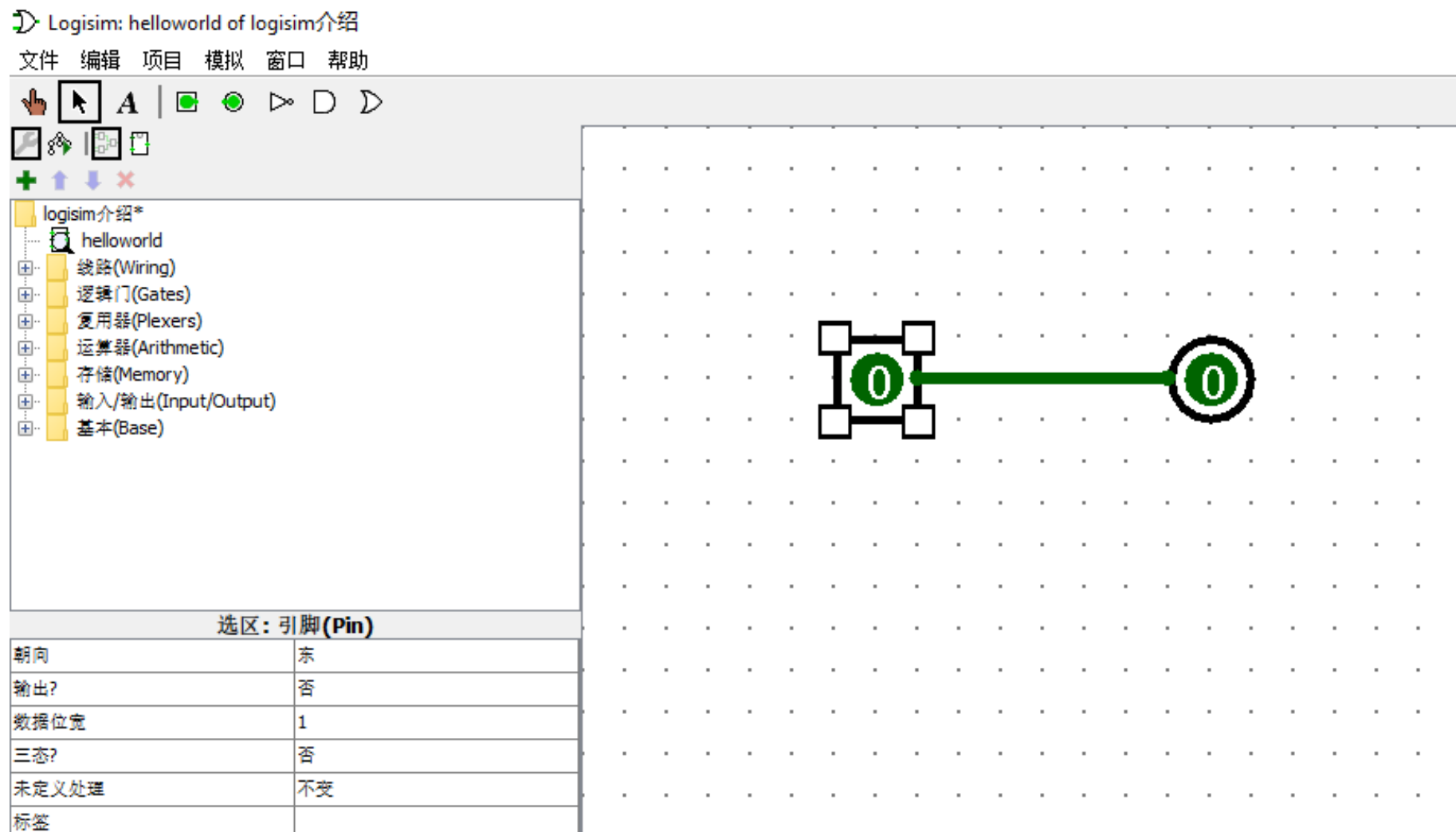
■ 输入输出引脚

- 电路模块输入输出信号的接口



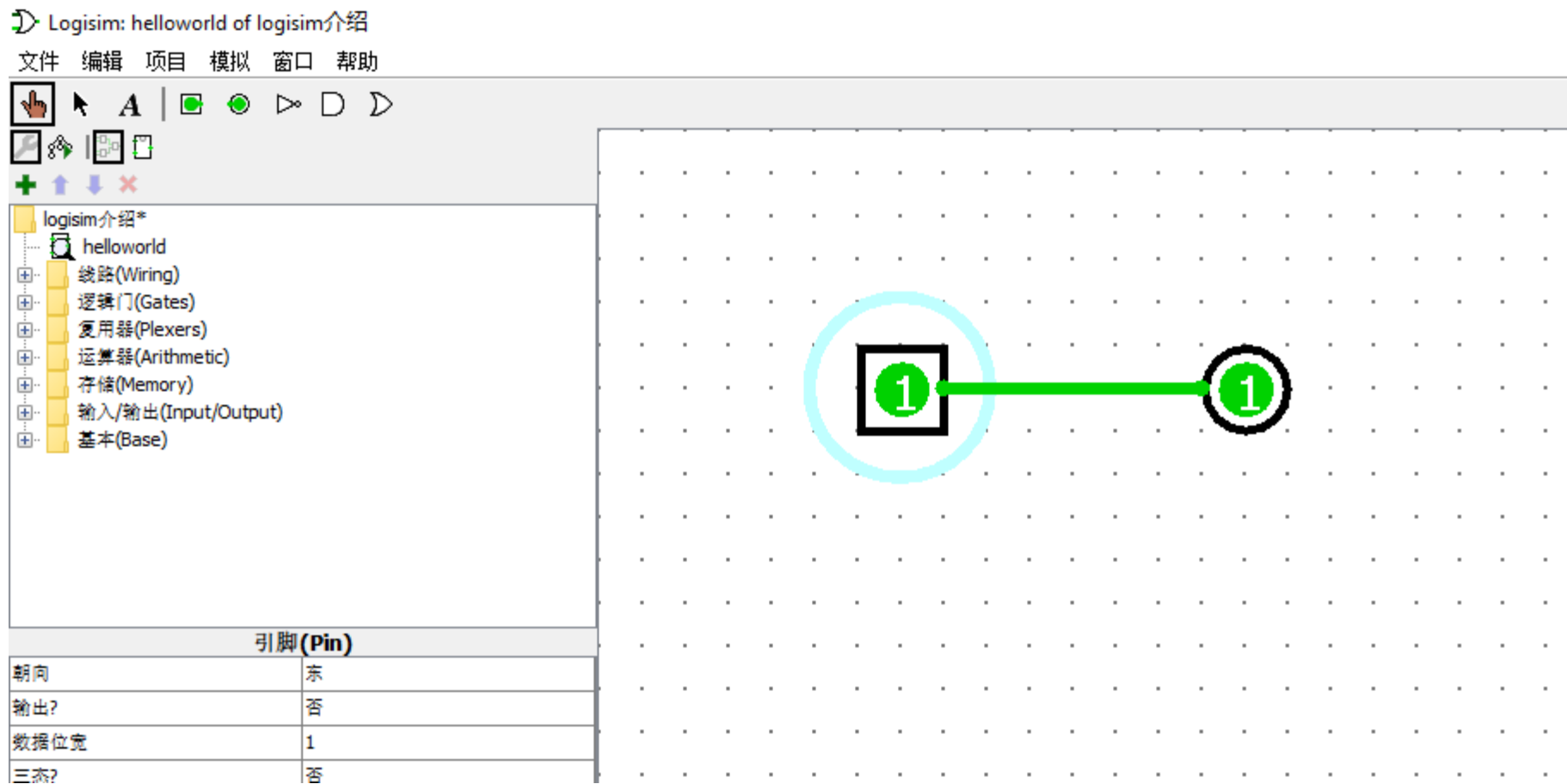
■ 添加连线

- 实时仿真

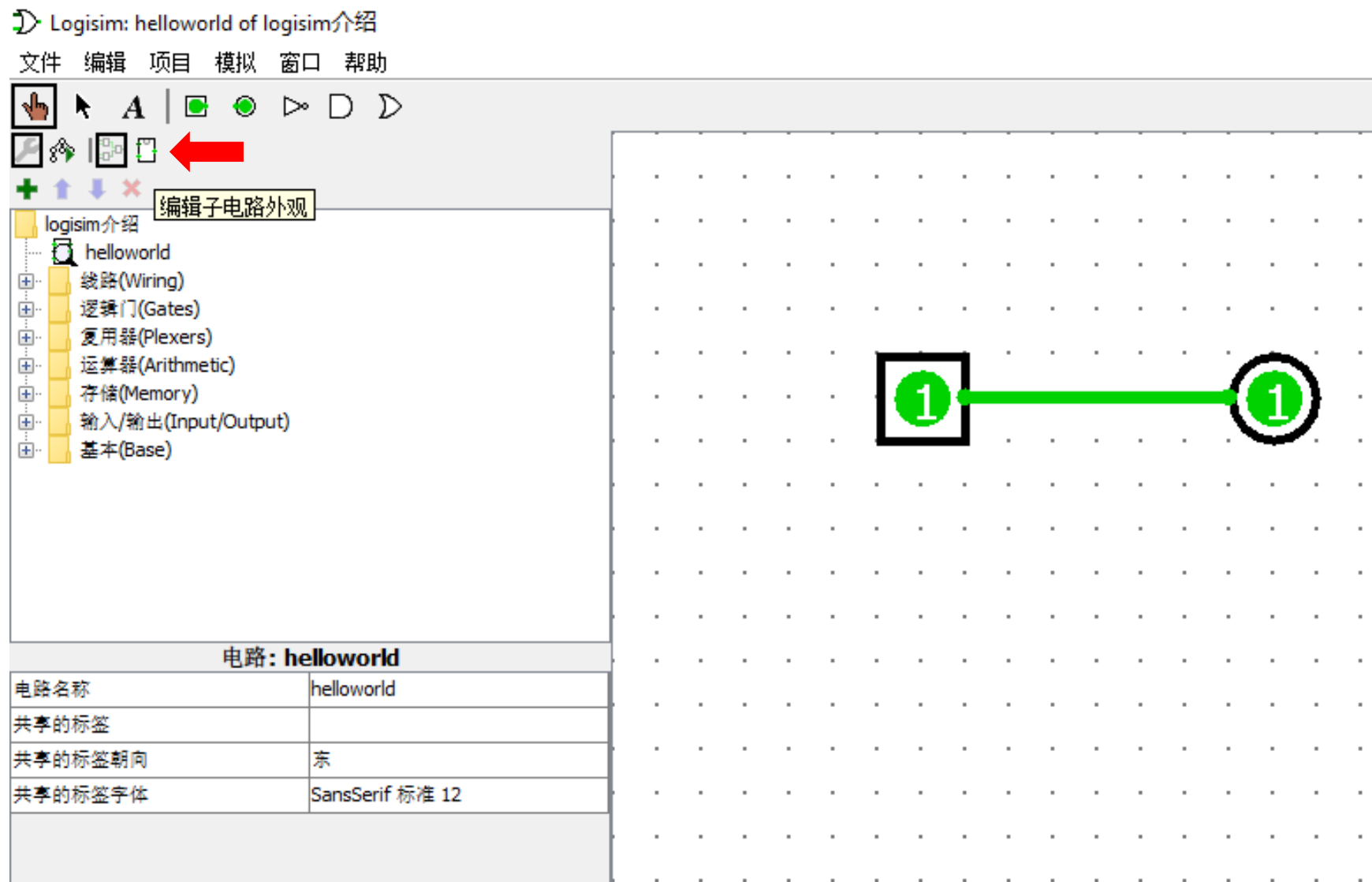


■ 改变输入信号值

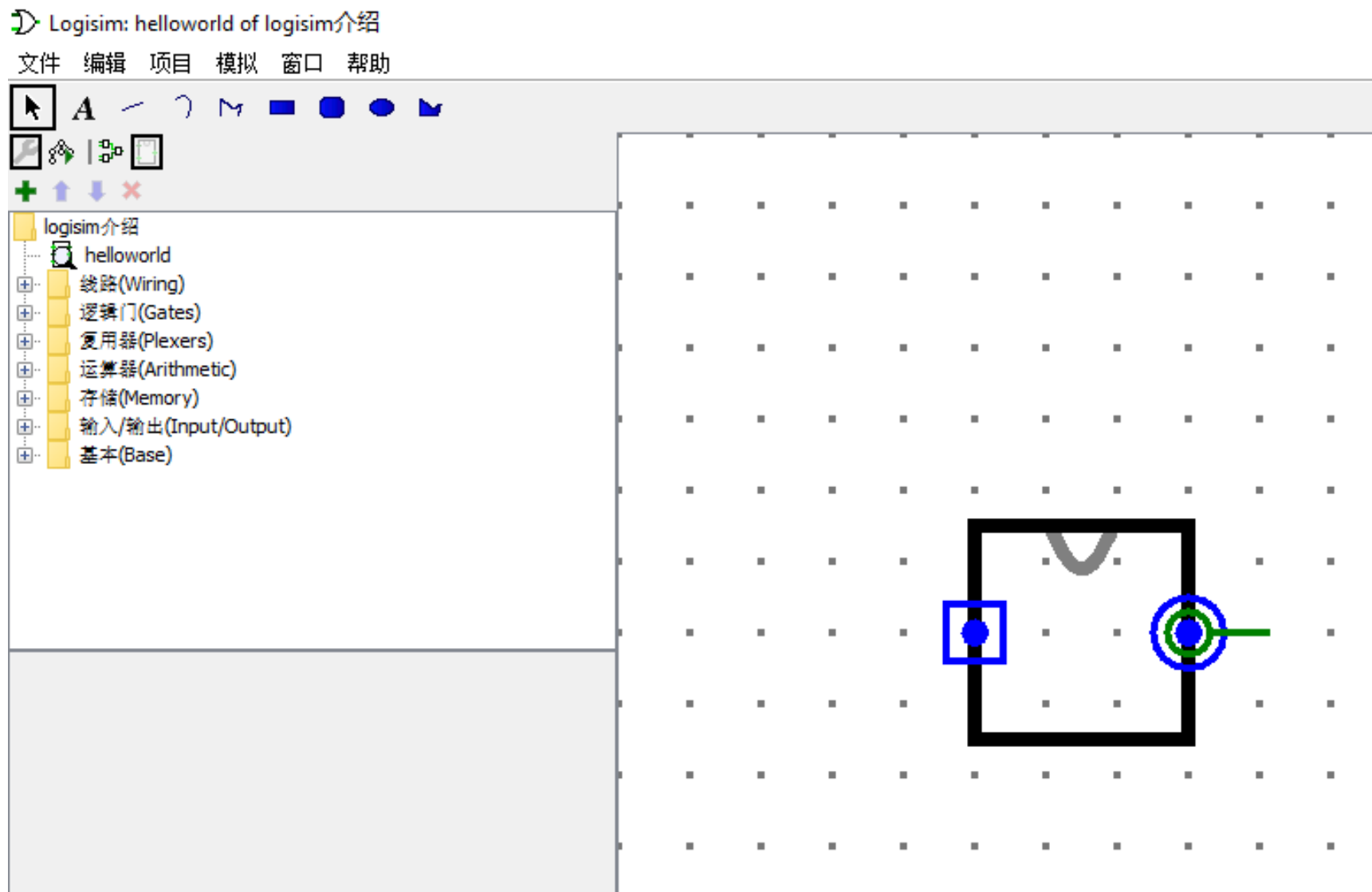
- 实时仿真



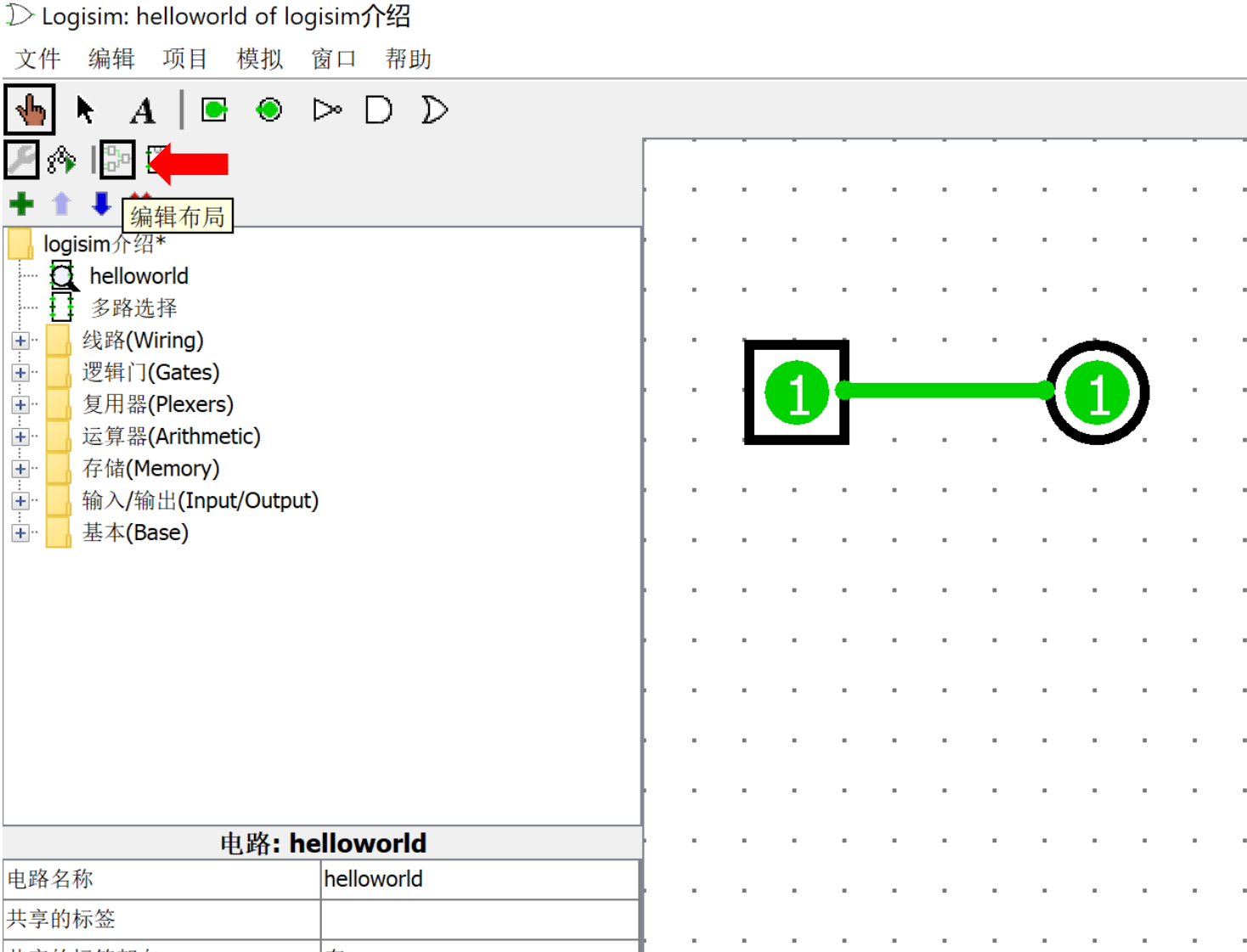
■ 封装子电路模块



■ 封装子电路模块



■ 返回电路编辑



■ 快捷键

- 朝向：光标键
- 位宽：alt+数字（字母区上方的数字键，不是小键盘区的数字键）

Logisim: helloworld of logisim介绍

文件 编辑 项目 模拟 窗口 帮助

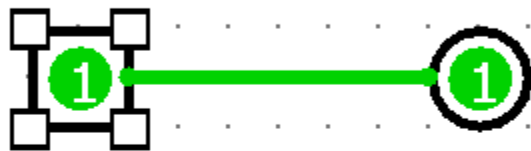


logisim介绍*

- helloworld
- 线路(Wiring)
- 逻辑门(Gates)
- 复用器(Plexers)
- 运算器(Arithmetic)
- 存储(Memory)
- 输入/输出(Input/Output)
- 基本(Base)

选区: 引脚(Pin)

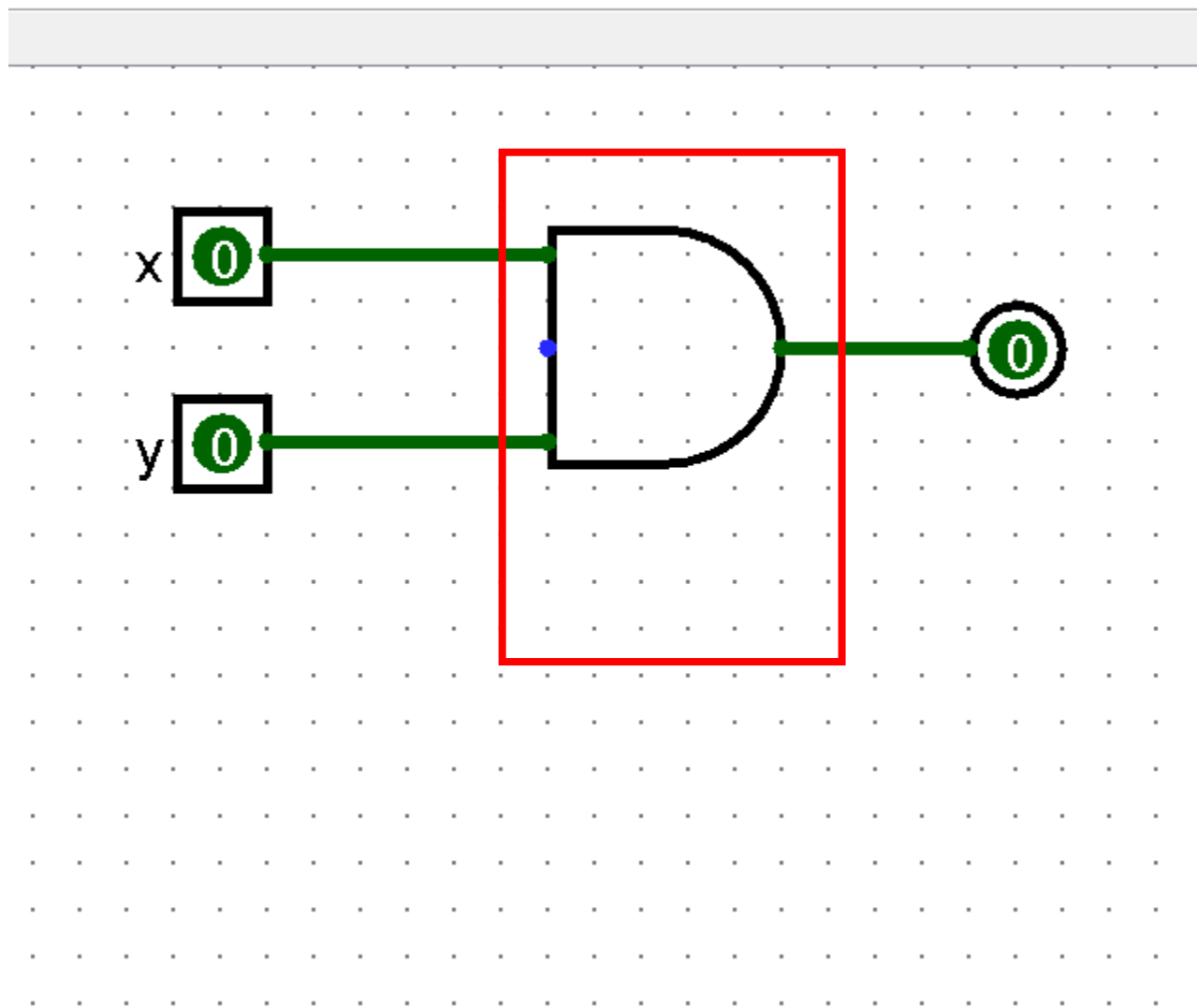
朝向	东
输出?	否
数据位宽	1
三态?	否
未定义处理	不变
标签	
标签位置	西
标签字体	SansSerif 标准 12



更改属性

■ 快捷键

- Ctrl+D: 复制组件
- 引脚数: 数字键



■ 更改输入信号值

Logisim: helloworld of logisim介绍

文件 编辑 项目 模拟 窗口 帮助

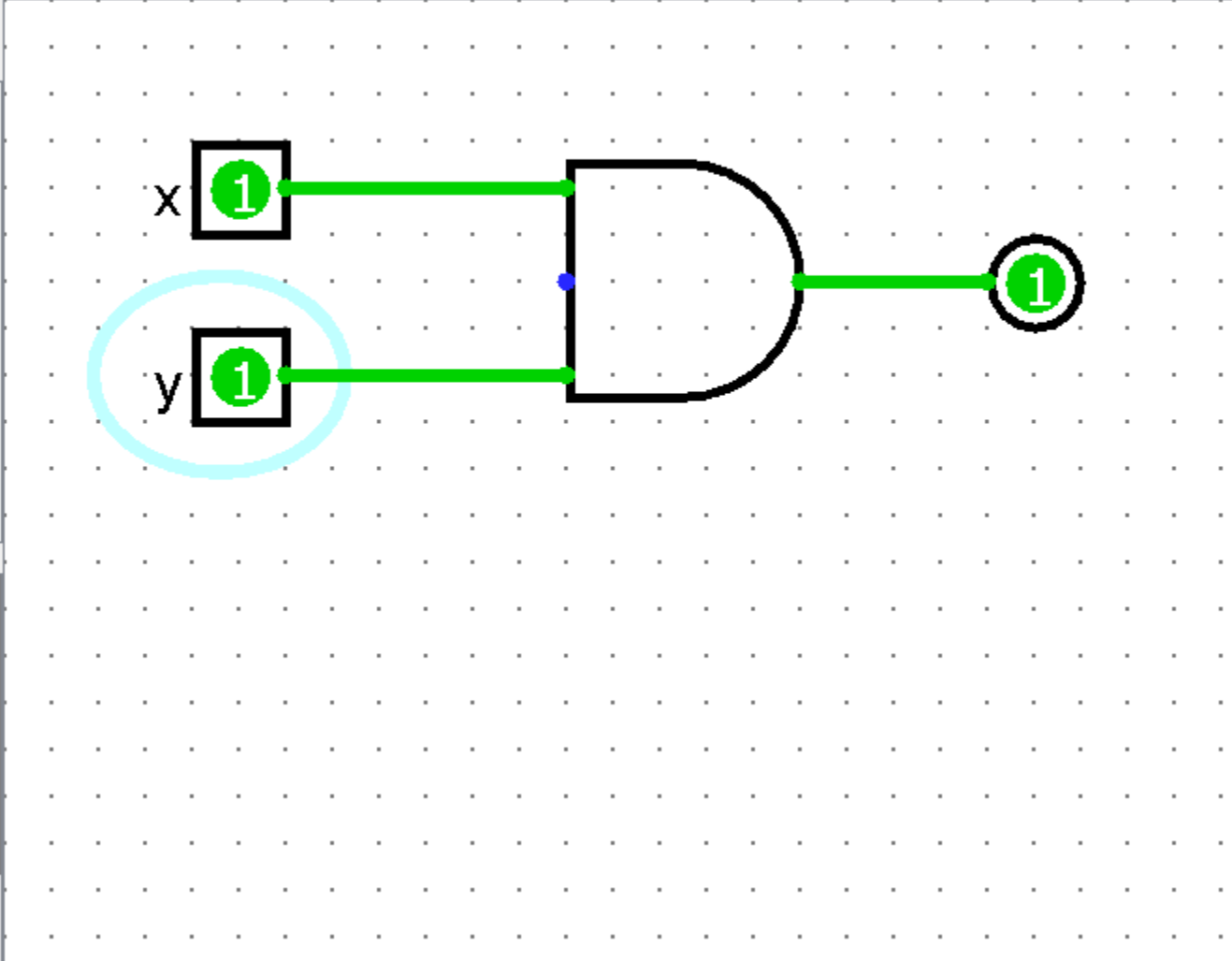
✎ 🔍 ⏏ ⏏ ⏏ ⏏ ⏏ ⏏

✚ ⬆ ⬇ ⬇ ✖

- logisim介绍
 - helloworld
 - 线路(Wiring)
 - 逻辑门(Gates)
 - 复用器(Multiplexers)
 - 运算器(Arithmetic)
 - 存储(Memory)
 - 输入/输出(Input/Output)
 - 基本(Base)

引脚(Pin)

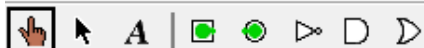
朝向	东
输出?	否
数据位宽	1
三态?	否
未定义处理	不变
标签	y
标签位置	西
标签字体	SansSerif 标准 12



构造复杂电路

Logisim: helloworld of logisim介绍

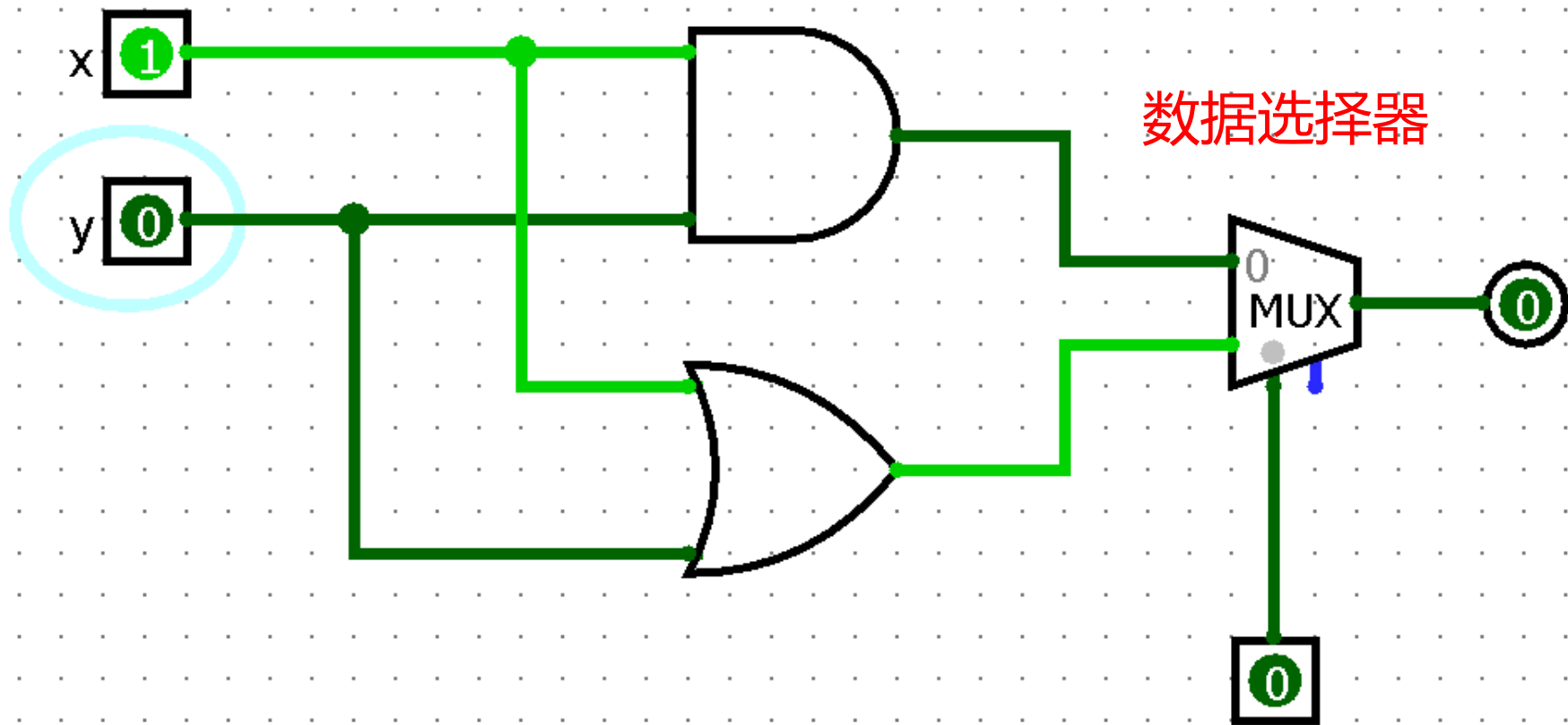
文件 编辑 项目 模拟 窗口 帮助



- logisim介绍*
- helloworld
- 线路(Wiring)
- 逻辑门(Gates)
- 复用器(Plexers)
- 数据选择器(Multiplexer)
- 解复用器(Demultiplexer)
- 解码器(Decoder)
- 优先编码器(Priority Encoder)
- 位选择器(Bit Selector)
- 运算器(Arithmetic)
- 存储(Memory)
- 输入/输出(Input/Output)
- 基本(Base)

引脚(Pin)

朝向	东
输出?	否
数据位宽	1
三态?	否
未定义处理	不变
标签	y
标签位置	西
标签字体	SansSerif 标准 12



数据选择器

选择信号

构造复杂电路

Logisim: helloworld of logisim介绍

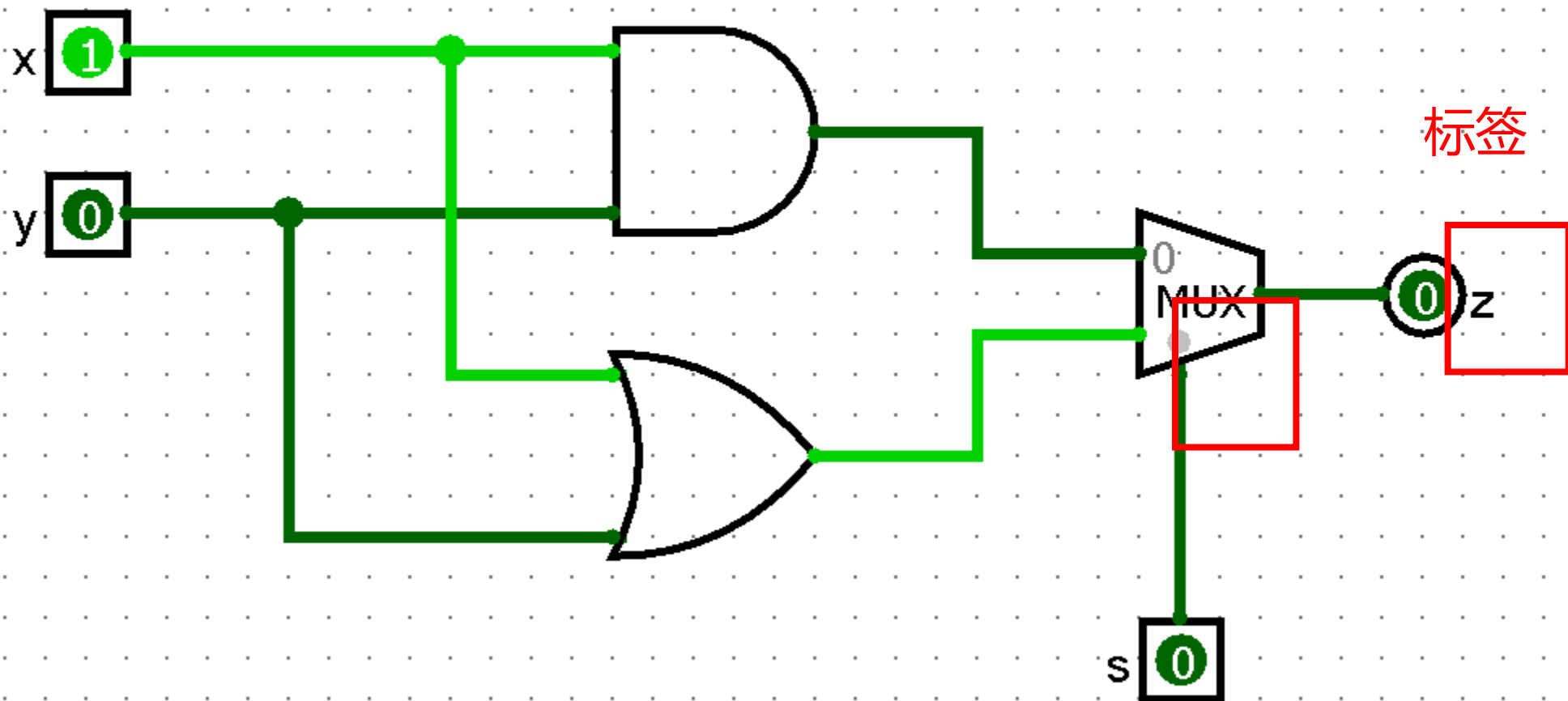
文件 编辑 项目 模拟 窗口 帮助



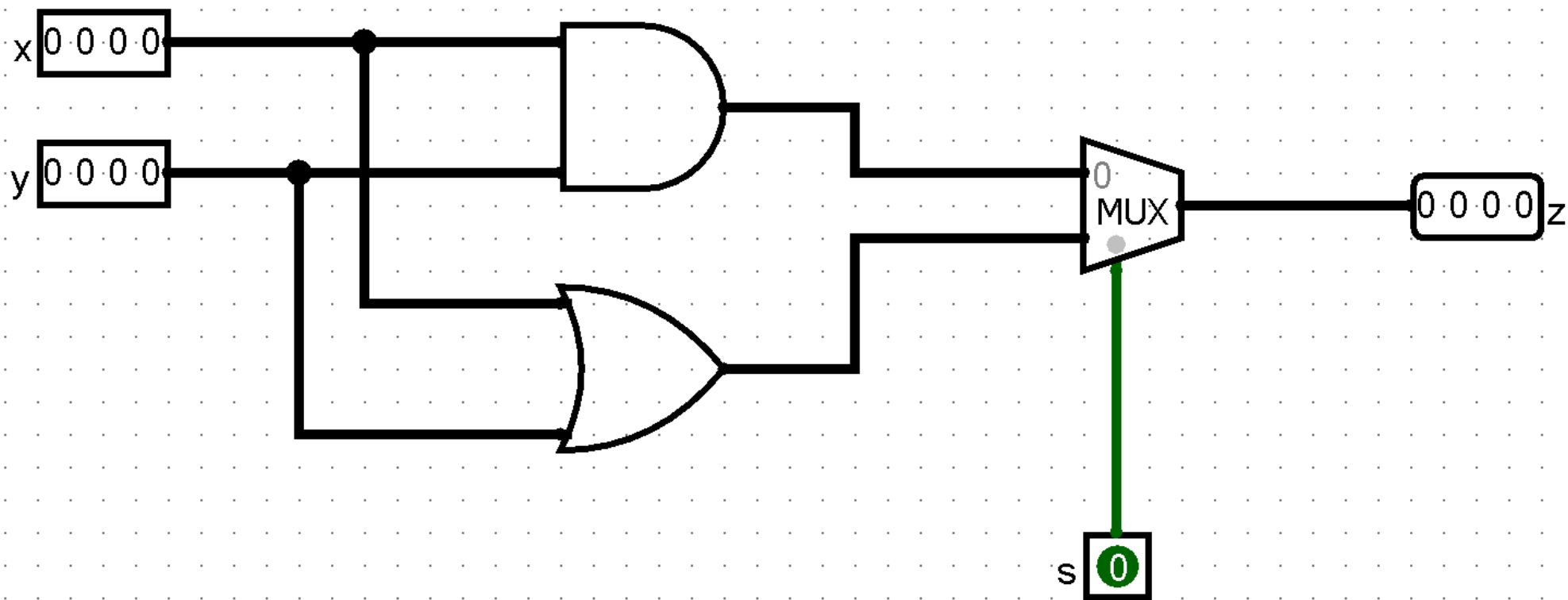
- logisim介绍*
- helloworld
- 线路(Wiring)
- 逻辑门(Gates)
- 复用器(Plexers)
- 数据选择器(Multiplexer)
- 解复用器(Demultiplexer)
- 解码器(Decoder)
- 优先编码器(Priority Encoder)
- 位选择器(Bit Selector)
- 运算器(Arithmetic)
- 存储(Memory)
- 输入/输出(Input/Output)
- 基本(Base)

电路: helloworld

电路名称	helloworld
共享的标签	
共享的标签朝向	东
共享的标签字体	SansSerif 标准 12



■ 更改电路位宽



■ 更改电路位宽

Logisim: helloworld of logisim介绍

文件 编辑 项目 模拟 窗口 帮助

✎ ⏏ A | 🟢 ⏏ ⏏ ⏏ ⏏

📁 📁 📁 📁

✚ ⬆ ⬇ ⬇ ✖

logisim介绍*

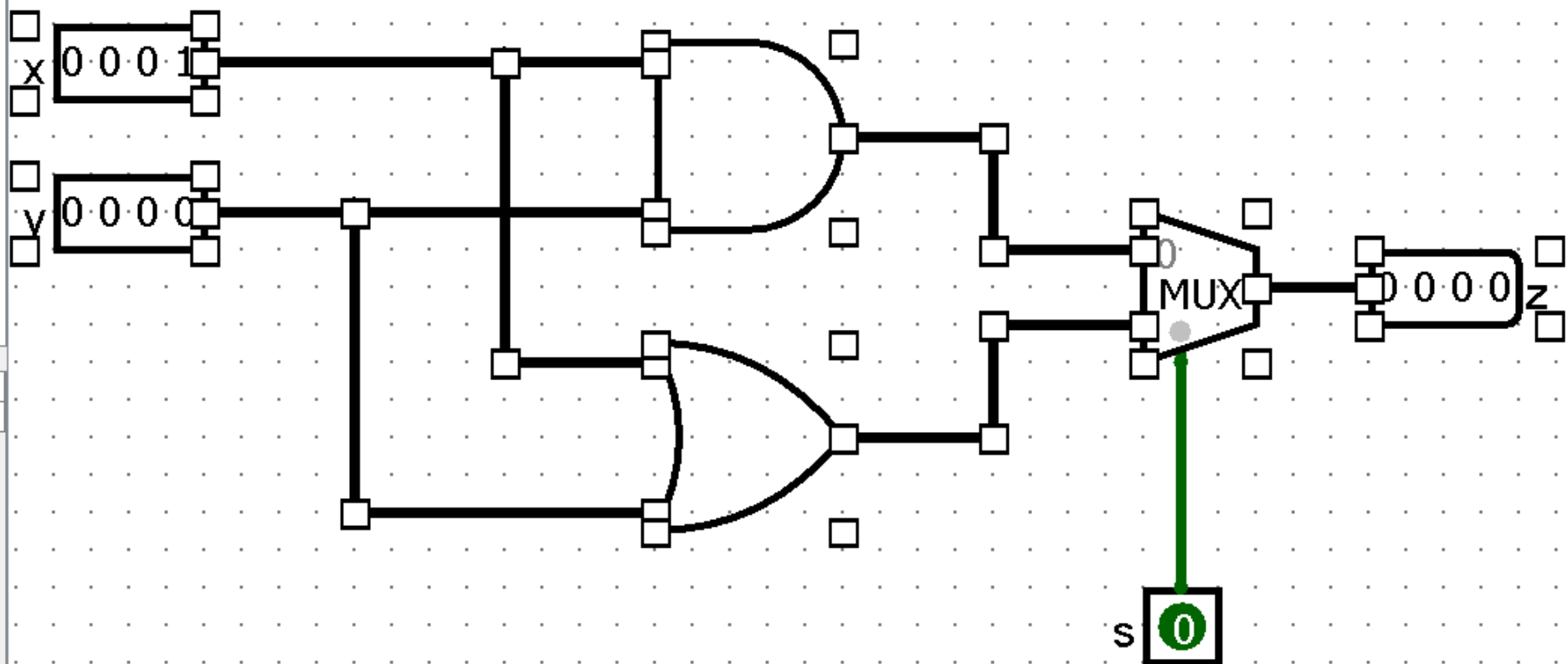
- helloworld
- 线路(Wiring)
- 逻辑门(Gates)
- 复用器(Plexers)
 - 数据选择器(Multiplexer)
 - 解复用器(Demultiplexer)
 - 解码器(Decoder)
 - 优先编码器(Priority Encoder)
 - 位选择器(Bit Selector)
- 运算器(Arithmetic)
- 存储(Memory)
- 输入/输出(Input/Output)
- 基本(Base)

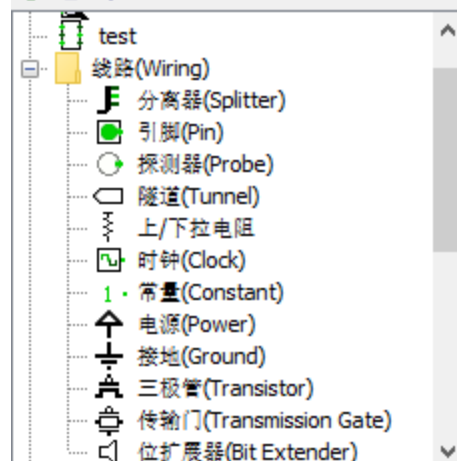
选区: 6个组件

朝向

数据位宽

4

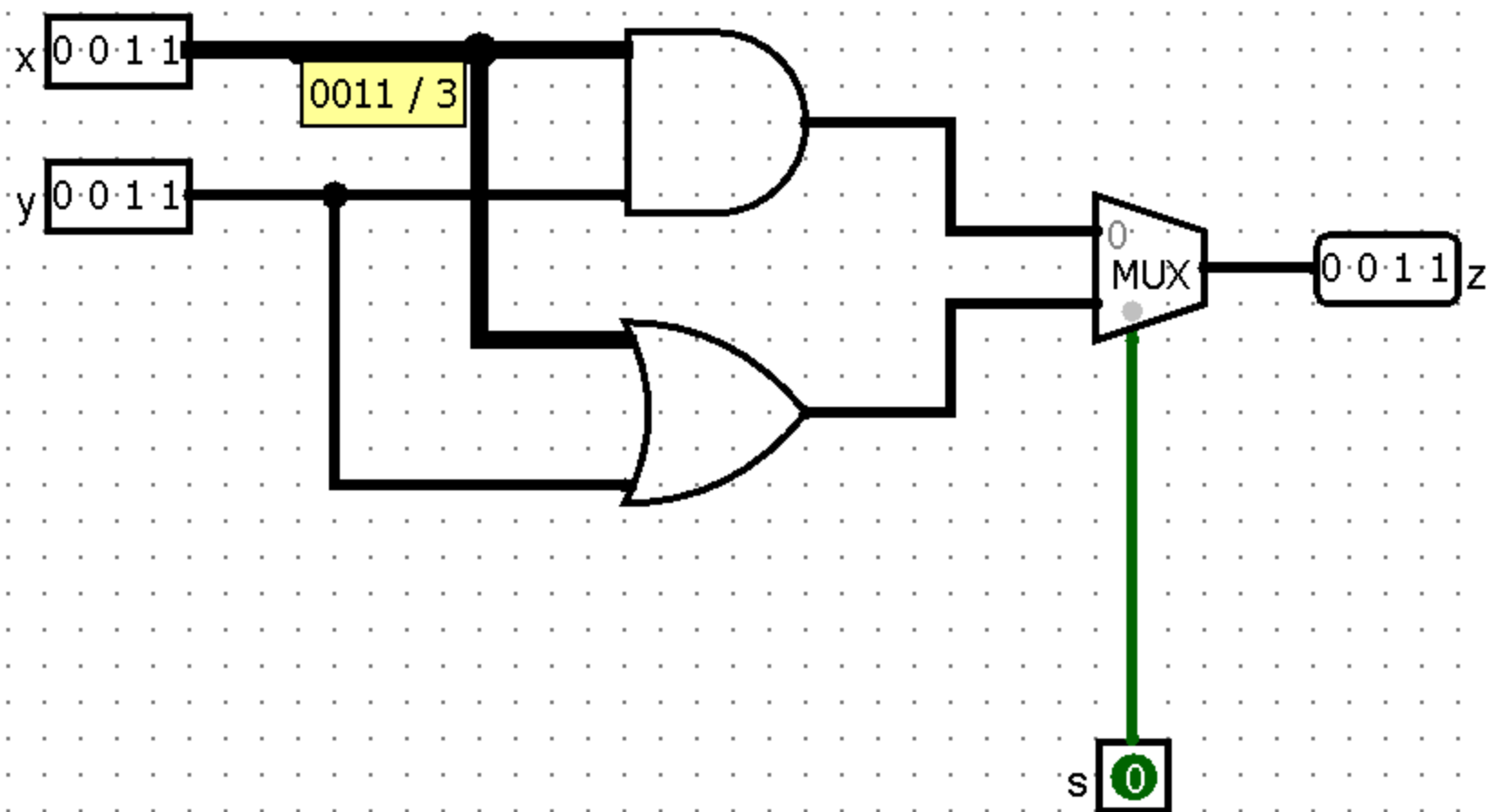


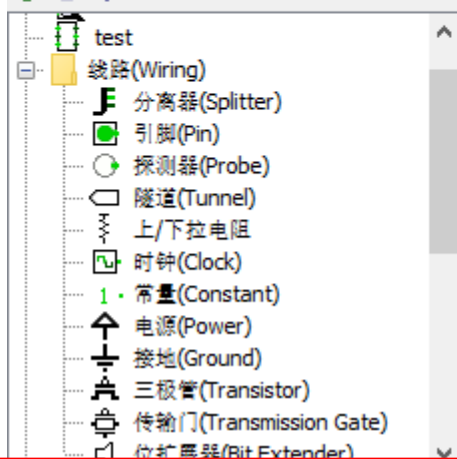
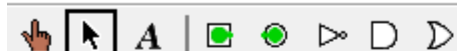


电路: helloworld

电路名称	helloworld
共享的标签	
共享的标签朝向	东
共享的标签字体	SansSerif 标准 12

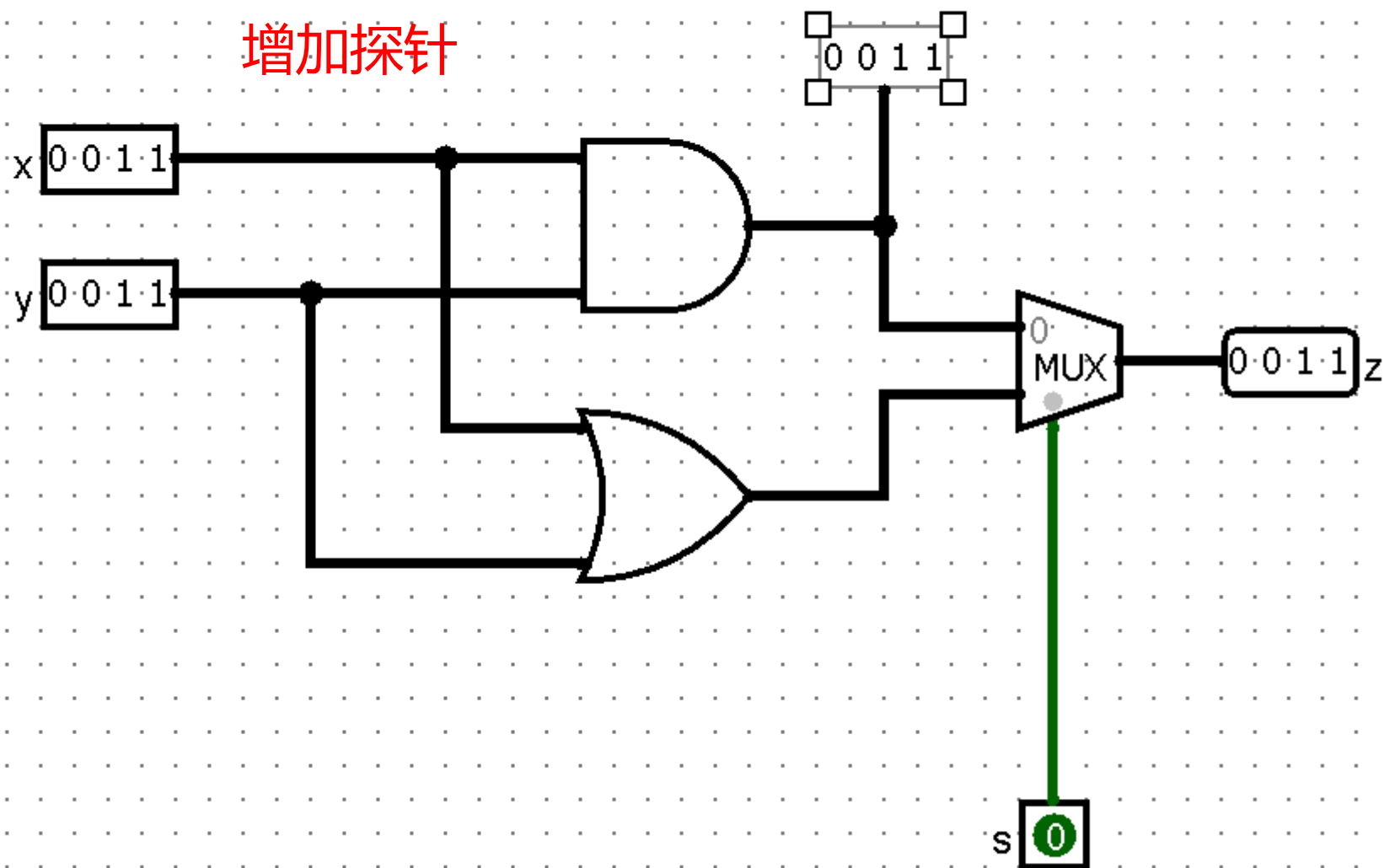
显示输入值

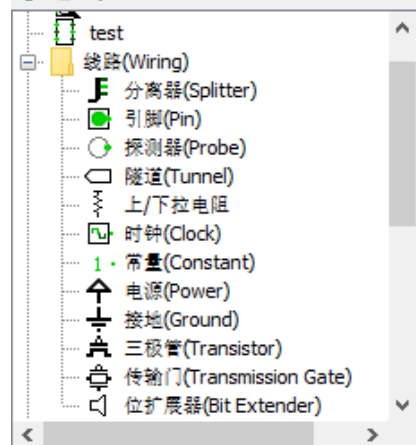
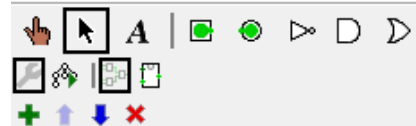




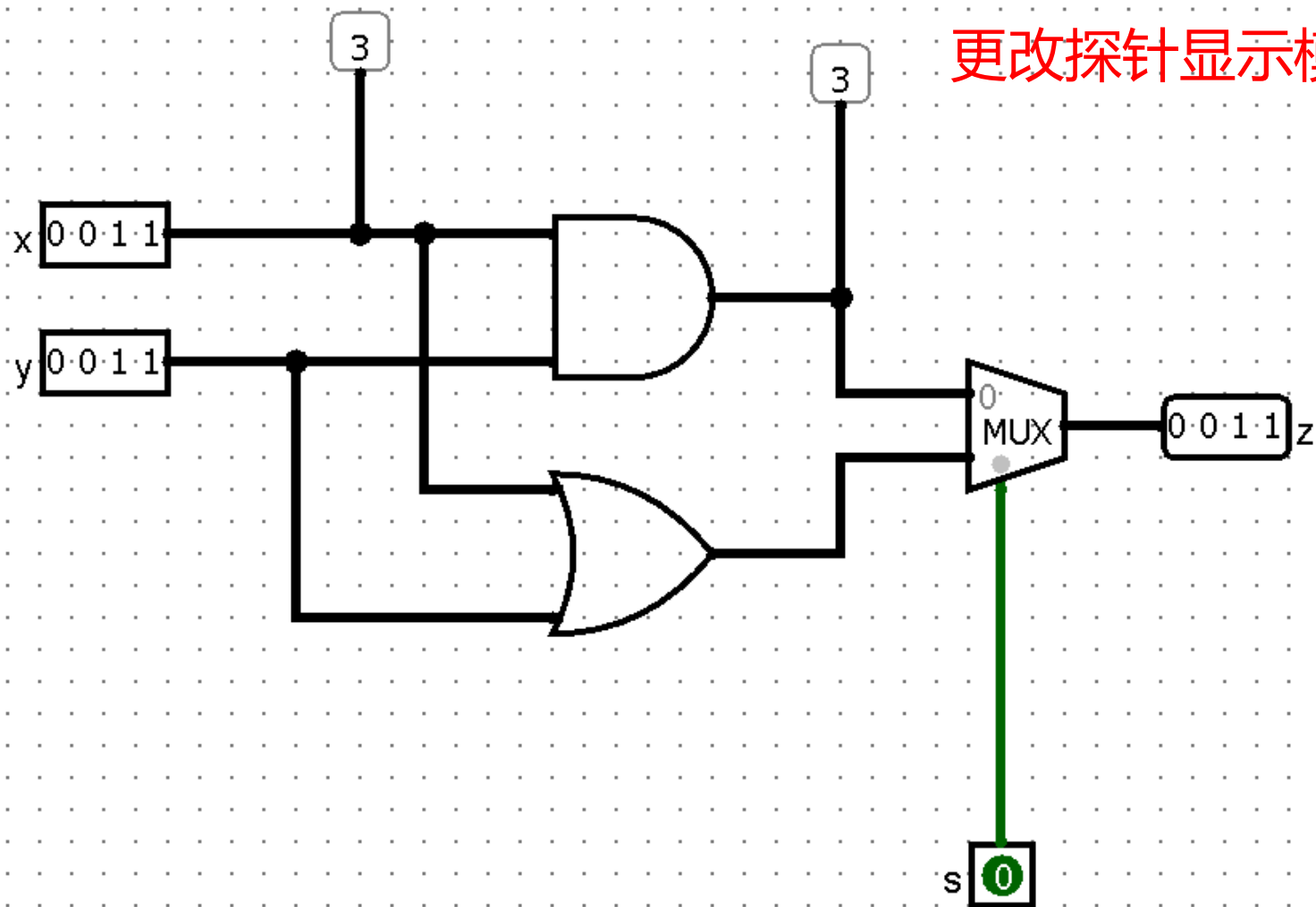
选区: 探测器(Probe)	
朝向	南
进制	二进制
标签	
标签位置	西
标签字体	SansSerif 标准 12

增加探针





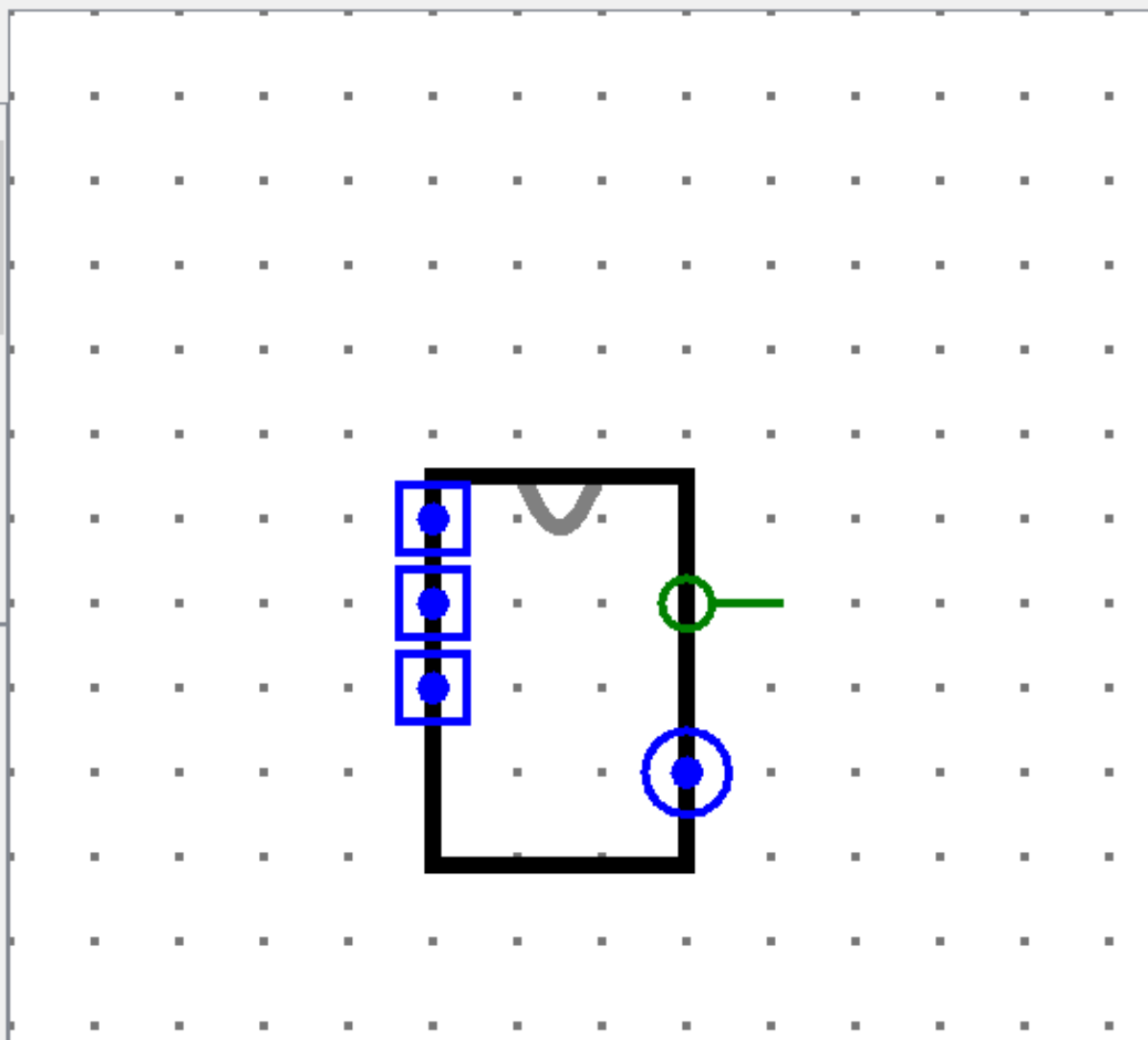
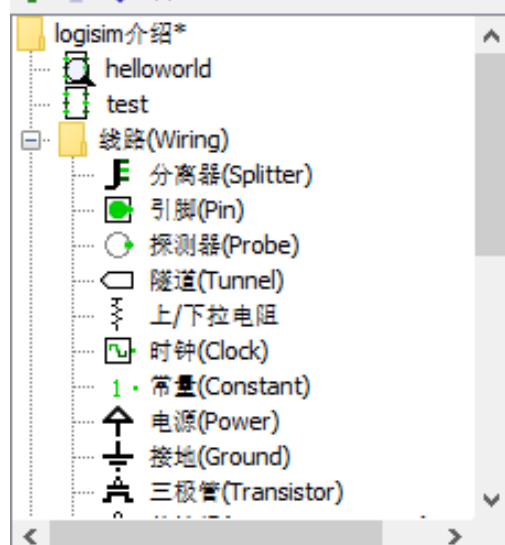
电路: helloworld	
电路名称	helloworld
共享的标签	
共享的标签朝向	东
共享的标签字体	SansSerif 标准 12



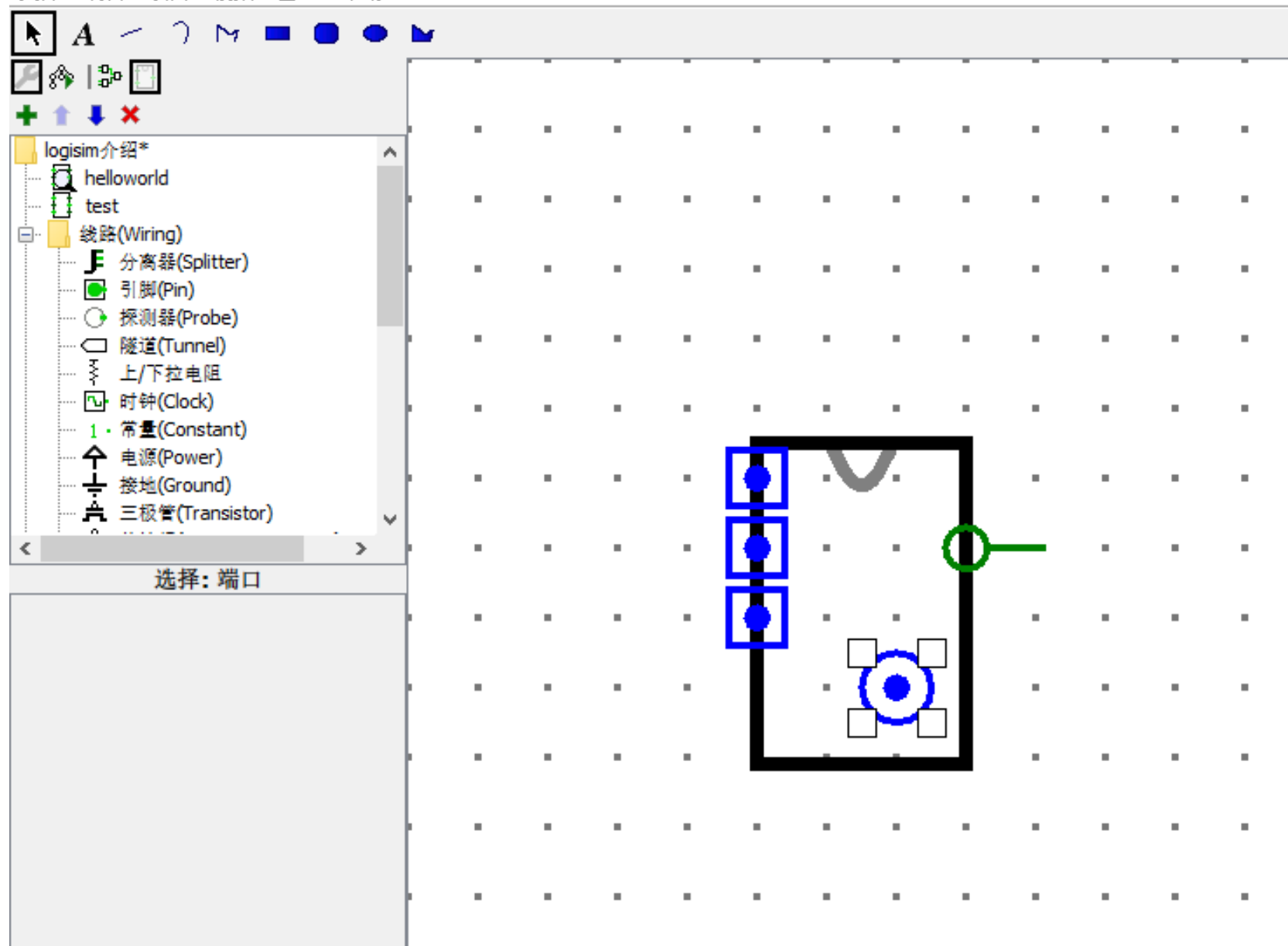
更改探针显示模式

Logisim: helloworld of logisim介绍

文件 编辑 项目 模拟 窗口 帮助



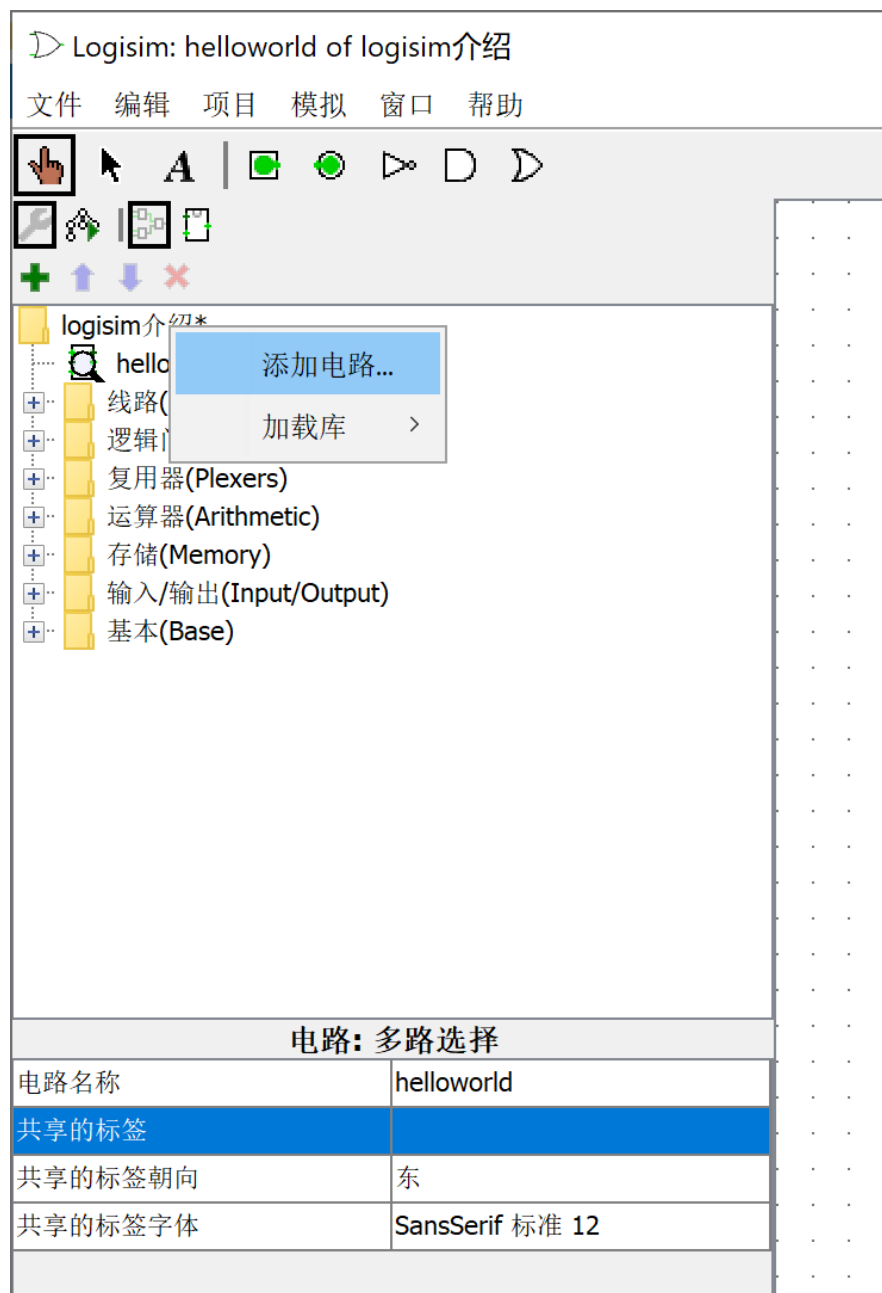
子电路封装

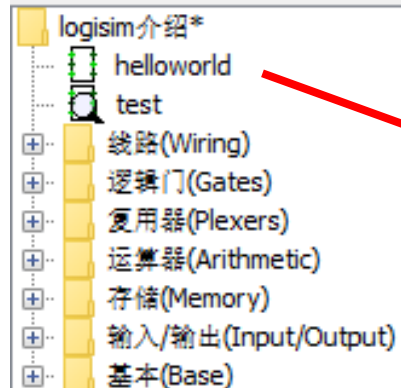


更改引脚位置

■ 创建新的子电路

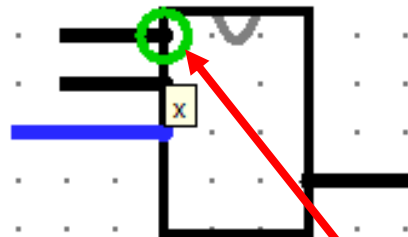
右击





电路: test

电路名称	test
共享的标签	
共享的标签朝向	东
共享的标签字体	SansSerif 标准 12



在新子电路中使用
已有模块:
点击选中, 添加

鼠标悬浮, 显示引脚标签

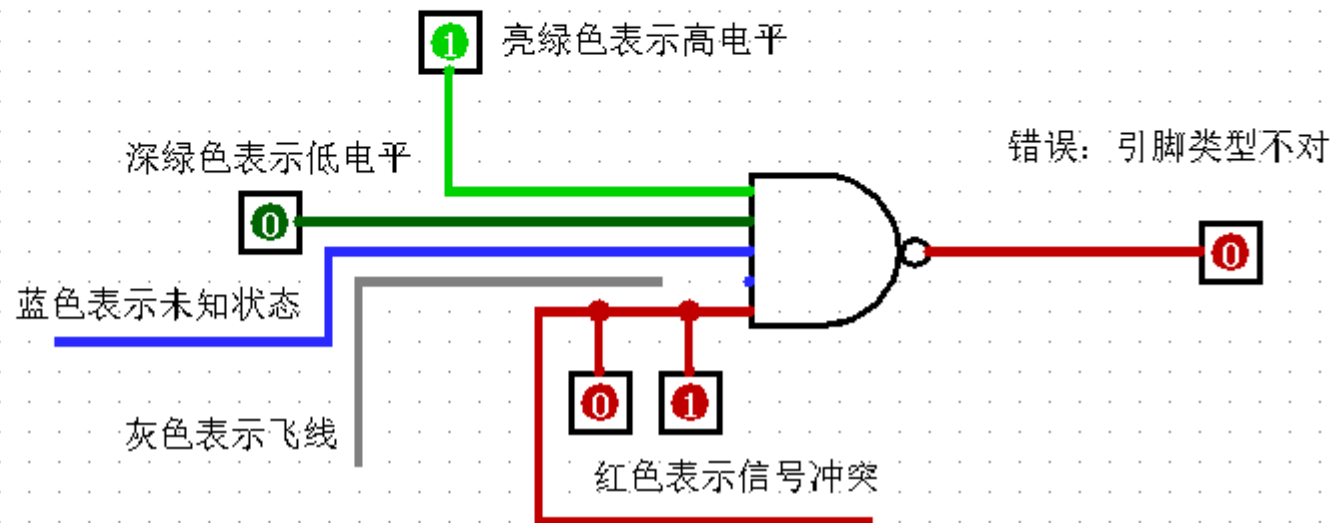


Logisim 0基础入门(1)

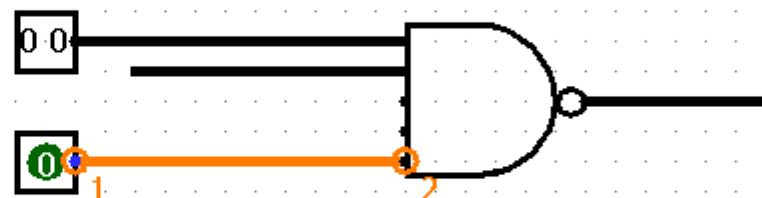
- 从零开始
- 测试电路
- 不同颜色线缆
- 从零开始-利用分析电路生成电路
- 线路(Wiring)
- 逻辑门(Gates)
- 复用器(Plexers)
- 运算器(Arithmetic)
- 存储(Memory)
- 输入/输出(Input/Output)
- 基本(Base)

电路: 不同颜色线缆

电路名称	不同颜色线缆
共享的标签	蓝色线表示未知状态
共享的标签朝向	北
共享的标签字体	SansSerif 标准 12

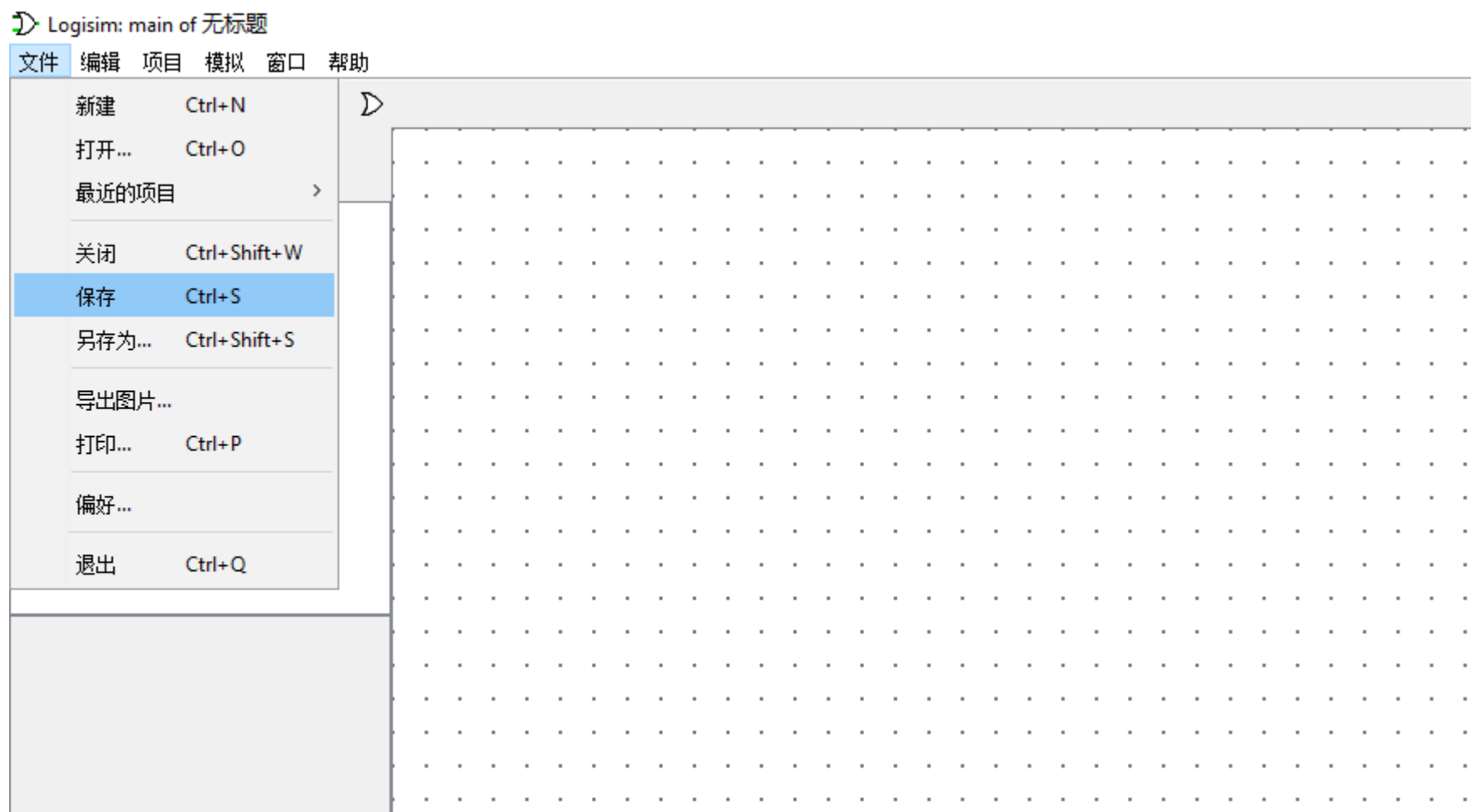


黑色表示多位总线

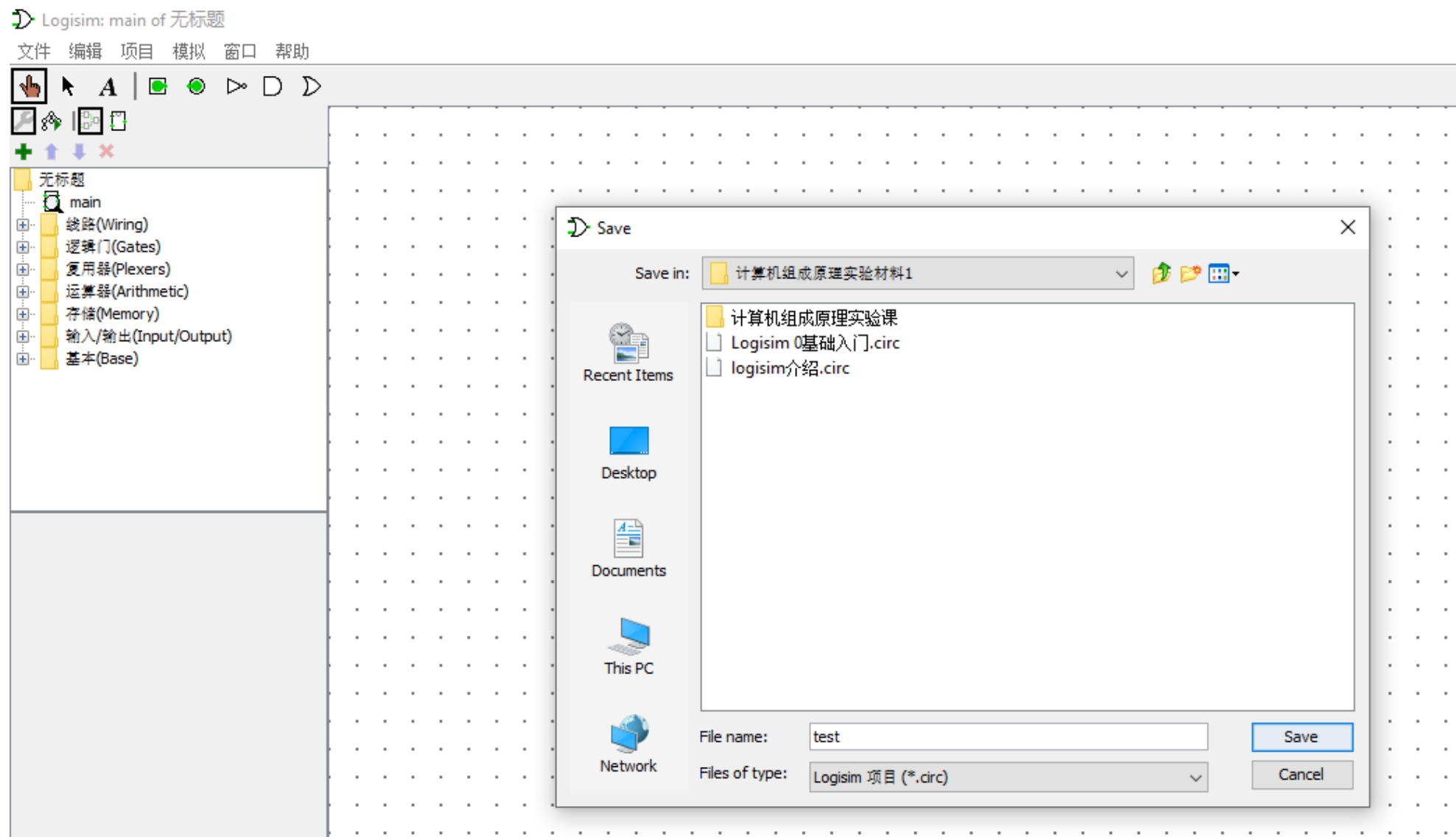


不兼容的位宽

■ 保存电路文件



■ 保存电路文件





谢谢！