

计算机设计与实践 汇编语言程序设计

2025 · 夏

哈工大



HITSZ 实验与创新实践教育中心
Education Center of Experiments and Innovations, HITSZ

实验目的

- ◆ 熟练掌握RISC-V或LoongArch汇编语言，熟悉并理解相应的指令系统
- ◆ 了解程序在单周期RISC-V或LoongArch SoC中的运行过程
- ◆ 进一步熟悉RARS、Logisim等汇编和模拟仿真工具的使用

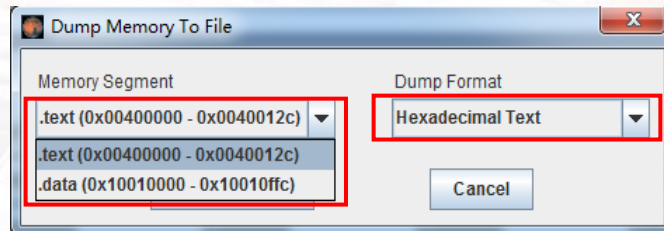
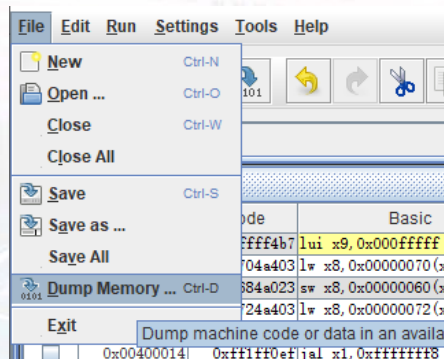


实验工具 — RARS

1. TheThirdOne / rars

汇编IDE：编辑器+汇编器+模拟器

.text存储在指令存储器，.data存储在数据存储器
如果汇编代码没有定义.data，则不会生成.data段



实验工具 — Logisim

2. Logisim



电路图设计工具

戳工具

可直接查看组件的值
可显示连线当前的值

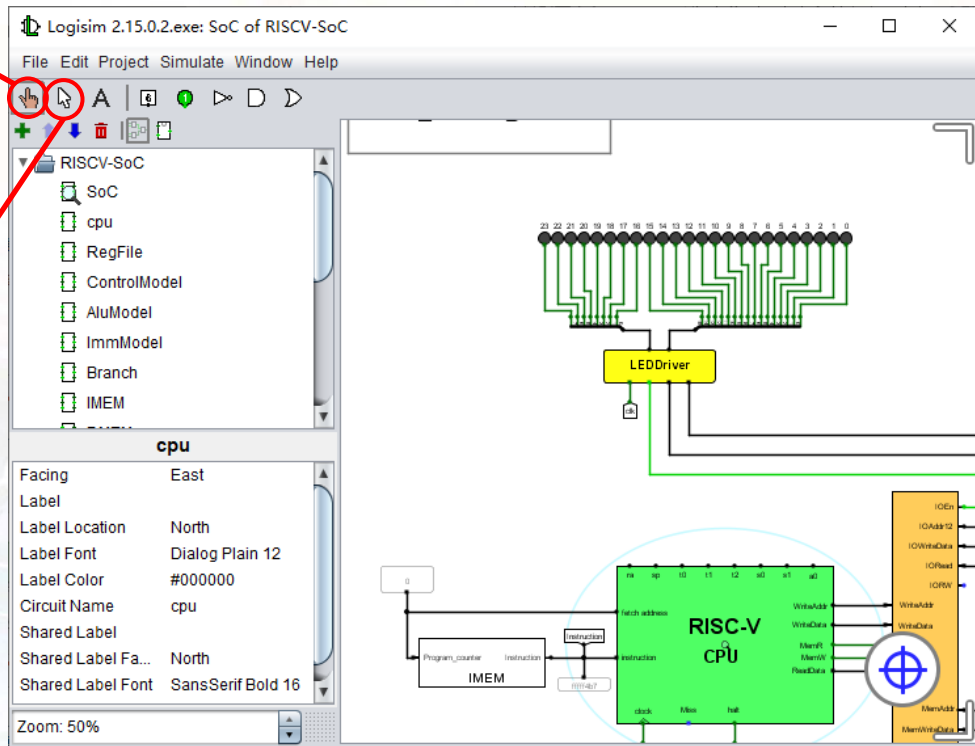
编辑工具

允许用户重新安排现有组件
修改组件属性并添加连线

Ctrl + R: 电路复位

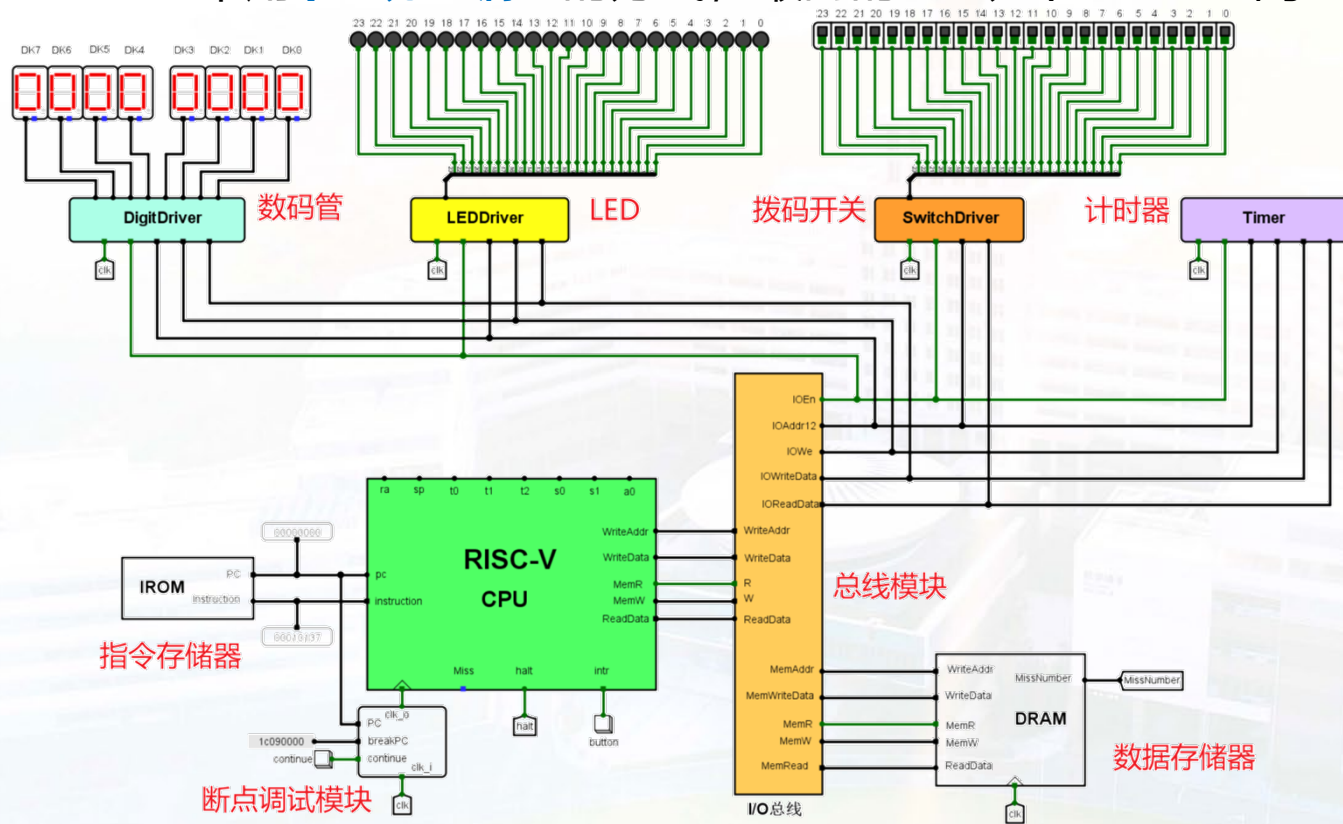
Ctrl + T: 时钟单步

Ctrl + K: 时钟连续



SoC电路

SoC采用**I/O统一编址**的方式，最高的4KB为I/O地址空间



外设	基址
数码管	0xFFFF_F000
计时器	0xFFFF_F020
	0xFFFF_F024
LED	0xFFFF_F060
拨码开关	0xFFFF_F070

实验内容 — 题目1：基本输入输出

- ◆ 在Logisim运行示例程序**Exercise1.asm** ([Exercise1A.asm for miniLA](#))
 - 阅读该程序源码，分析程序功能
 - 学习汇编程序如何访问I/O接口及外设
 - 根据实验指导书，运行程序，熟悉实验过程及各工具的使用



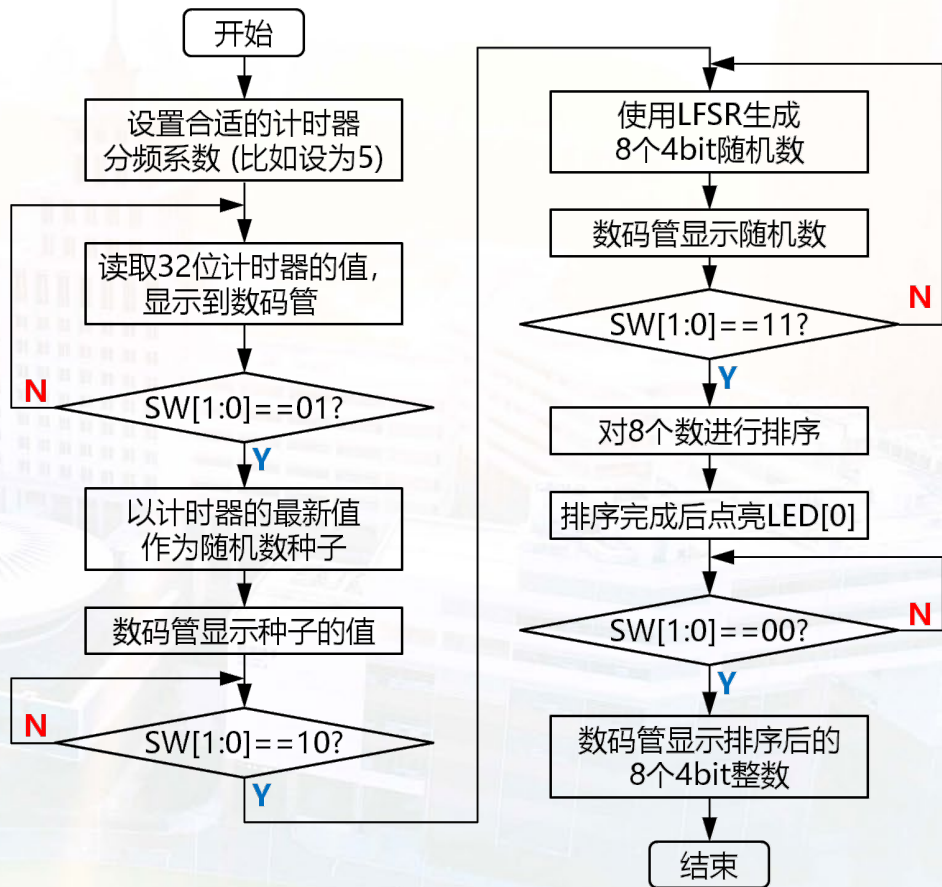
实验内容 — 题目2：随机数排序

◆ 运用miniRV/miniLA指令，
编写汇编程序：

- 输入：拨码开关sw[1:0]
- 输出：数码管、LED[0]
- 后续实现CPU后，本程序

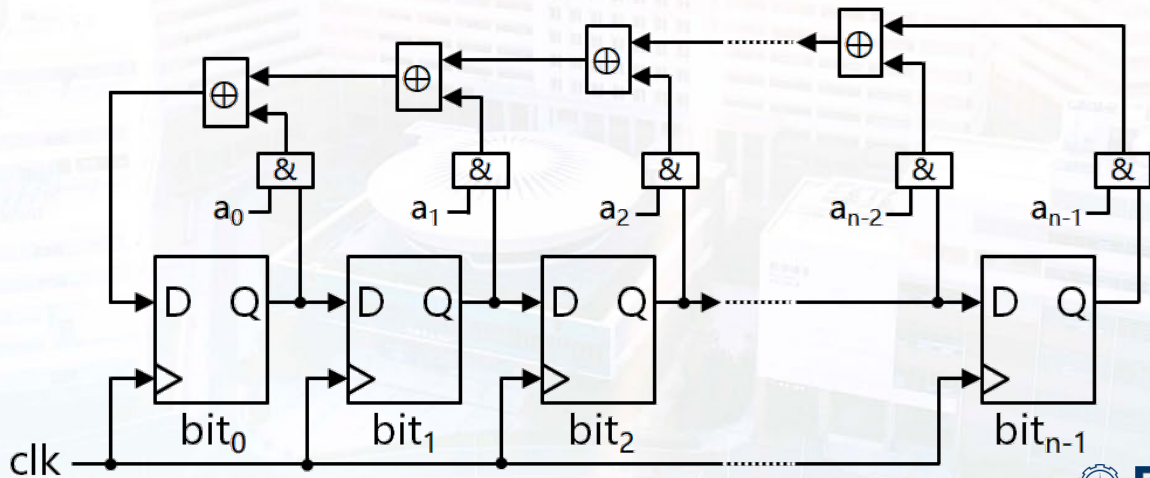
需下板演示，故不要使用

不准备实现的指令



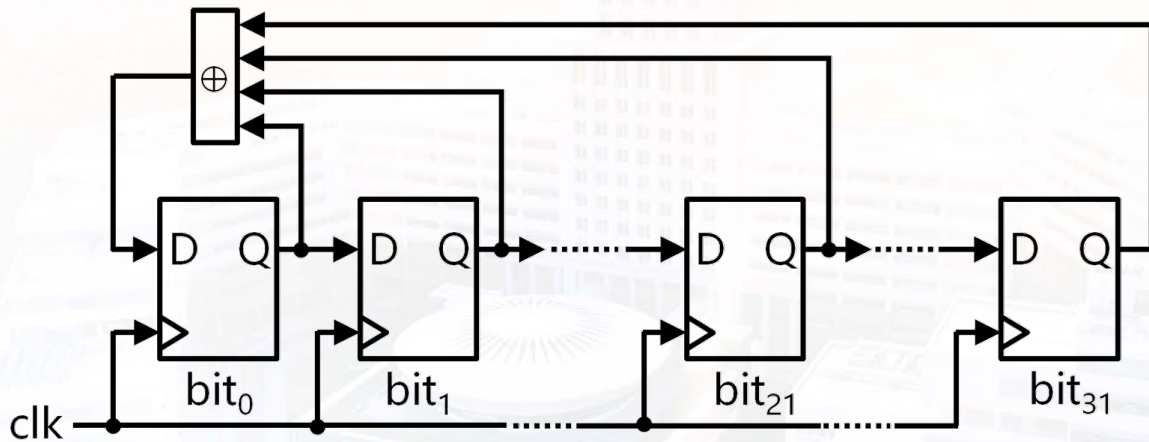
实验原理 — LFSR

- 线性反馈移位寄存器
 - 移位寄存器**的基础上添加**反馈输入**，常用于生成随机数、白噪声
 - LFSR的初始值即为随机数种子
 - a_i 表示 bit_i 对应的反馈回路是否存在



实验原理 — LFSR

- 线性反馈移位寄存器
 - 32位的LFSR**: $\{a_0, a_1, a_{21}, a_{31}\} = 4'b1111$



验收&提交

- **课堂验收**

- 课上检查题目2功能是否正确

- **提交内容**

- 题目1：自行完成，不需提交
 - 题目2：.asm源文件 (**!!!写注释!!!**)、汇编生成的.hex机器码

- 将上述文件打包成.zip，以 **“学号_姓名.zip”** 命名提交到作业系统

- ◆ 注意：**如有雷同，双方均0分！**

- 数据通路表、控制信号表DDL: **July 4th 23:00**



HITSZ 实验与创新实践教育中心
Education Center of Experiments and Innovations, HITSZ