

计算机设计与实践 汇编语言程序设计

2024 · 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汇编语言，熟悉并理解RISC-V指令系统
- ◆ 了解程序在单周期RISC-V CPU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

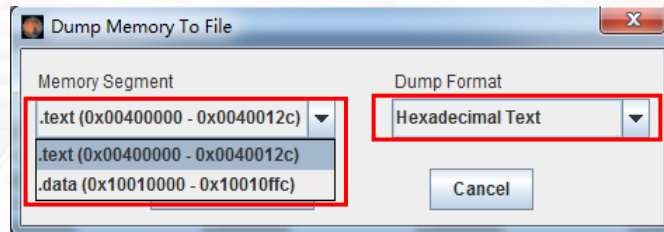
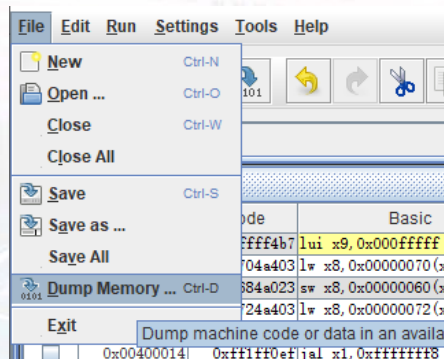
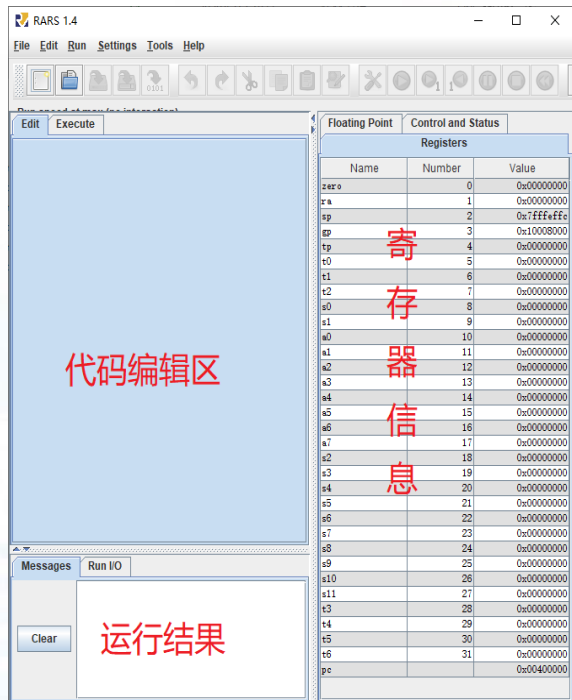


实验工具 — RARS

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

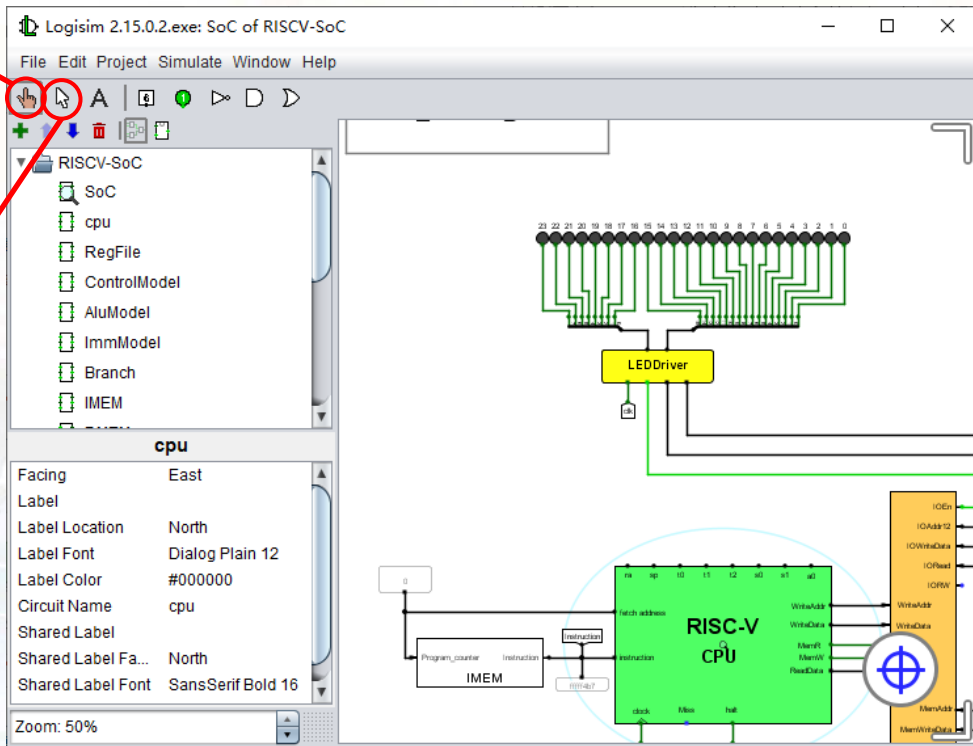
编辑工具

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

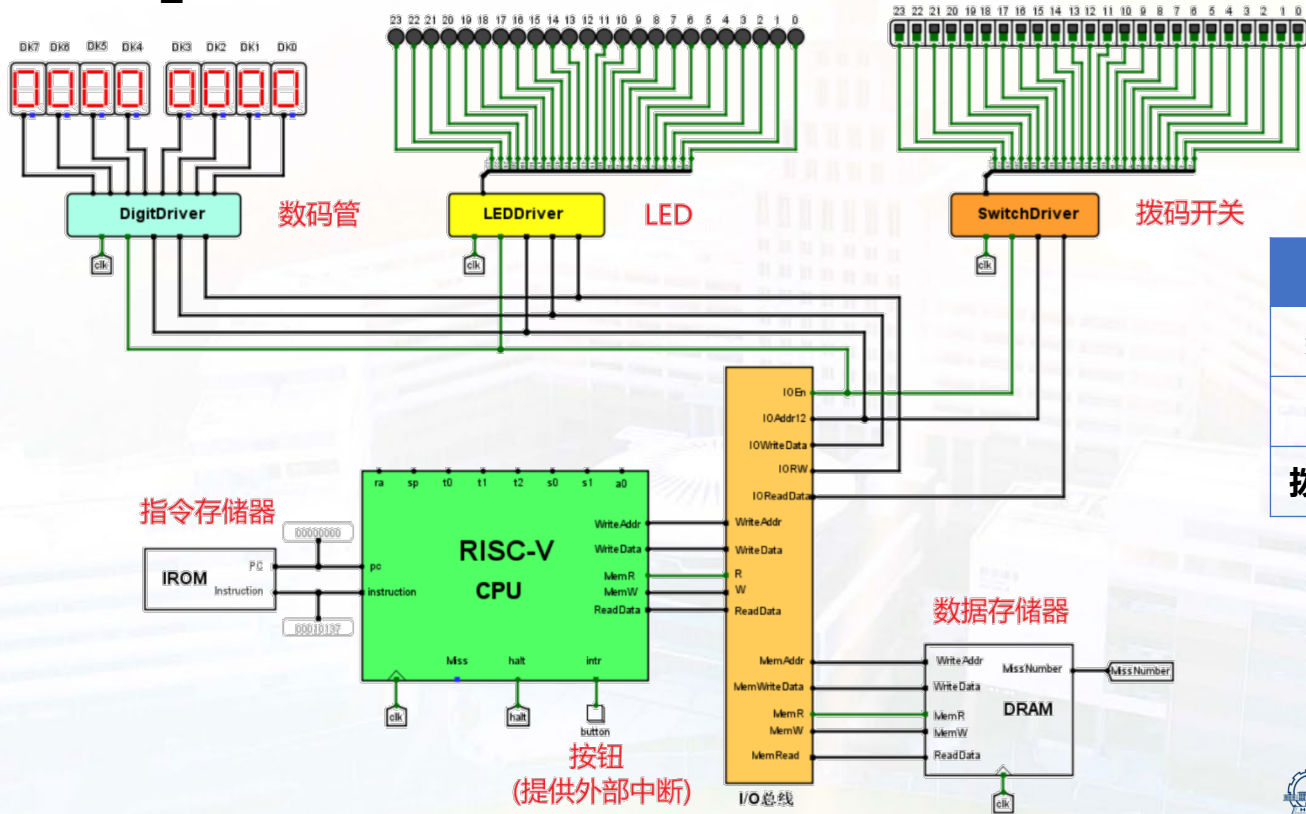
Ctrl + T: 时钟单步

Ctrl + K: 时钟连续



RISC-V_SoC电路

RISC-V 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	0xFFFFF000
LED	0xFFFFF060
拨码开关	0xFFFFF070

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：简易计算器设计

- ◆ 运用miniRV的指令，编写汇编程序实现8位简易计算器
 - 后续实现CPU后，本程序需要下板演示，故不要使用不实现的指令
 - 输入：操作符、操作数A、操作数B（从拨码开关输入）
 - 输出：运算结果（显示在数码管DK7~DK0）

拨码开关	SW[23:21]	SW[20:16]	SW[15:8]	SW[7:0]
输入功能	操作符	保留	操作数A	操作数B

- A、B均为无符号小数：高4位整数、低4位表示0~9的小数
 - 例：输入A为10110101 (11.5)，输入B为00100111 (2.7)，则A+B=14.2



实验内容 — 题目2：简易计算器设计

- 计算器功能：

SW[23:21]	运算类型	数码管显示内容
000	无	32'h0000_0000
001	$A + B$	高4位显示整数部分，低4位显示小数部分
010	$ A - B $	高4位显示整数部分，低4位显示小数部分
011	$A \times 2^B$	高4位显示整数部分，低4位显示小数部分
100	$A \div 2^B$	高4位显示整数部分，低4位显示小数部分
101	生成随机数	以 {A, B, A, B} 为种子显示32位随机数，每隔约1s更新一次

可通过移位
实现

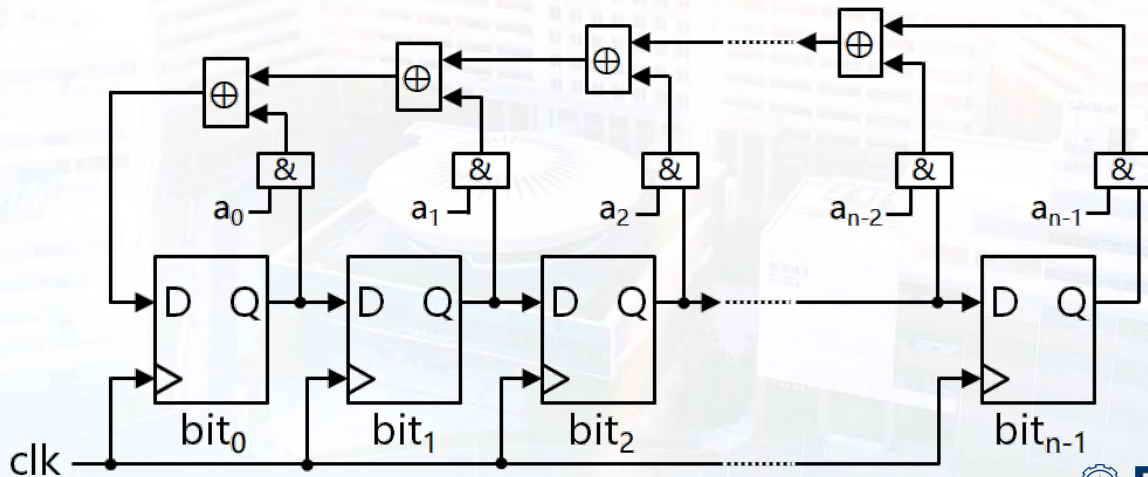
LFSR



实验原理 — LFSR

- 线性反馈移位寄存器

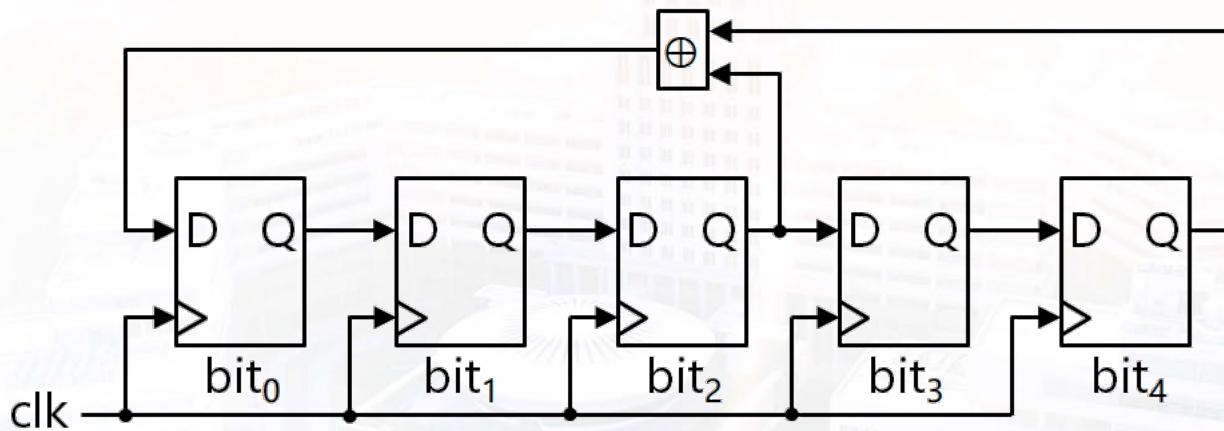
- 移位寄存器的基础上添加反馈输入，常用于生成随机数、白噪声
- 初值称为种子
- a_i 表示 bit_i 对应的反馈回路是否存在



实验原理 — LFSR

- 线性反馈移位寄存器

- 5位的LFSR**: $\{a_4, a_3, a_2, a_1, a_0\} = 5'b10100$



- 对于32位的LFSR, a_{31} 、 a_{21} 、 a_1 、 a_0 为1, 其余为0

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：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 12th 23:00**





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