

第九届

全国大学生集成电路创新创业大赛

报告类型： 设计报告

参赛杯赛： 竞业达杯

作品名称： 基于RISCV指令集的5级流水线CPU

队伍编号： CICC0900689

团队名称： DeepSleep

快速简介

本报告展示了项目整体设计架构和对CPU的设计展示

5级流水线CPU设计报告

项目组：Deepsleep

# 一、项目概述

项目名称：基于5级流水线结构的RISC CPU设计。

设计目标：设计一个支持基本RISC指令集的5级流水线CPU，实现指令并行执行，支持forwarding避免数据冒险，通过简易branch prediction解决控制冒险。

开发语言与平台：

HDL语言：Verilog/System Verilog HDL  
仿真平台：哈工大cdp-trace仿真测试平台  
综合工具：Xilinx Vivado  
开发硬件：JYD FPGA   
指令集架构：RISC-V

# 二、总体架构设计

## 1. 流水线结构

5级流水线划分：  
- IF（Instruction Fetch）  
- ID（Instruction Decode）  
- EX（Execute）  
- MEM（Memory access）  
- WB（Write Back）

## 2. 关键模块

指令存储器（ROM）  
寄存器堆（RegFile）  
程序计数器（PC）  
算术逻辑单元（ALU）  
数据存储器（DRAM）  
控制单元（Store/Load Unit + ALU Control）  
冒险处理单元（Hazard Detection + Forwarding Unit）

## 3. 总体数据通路图

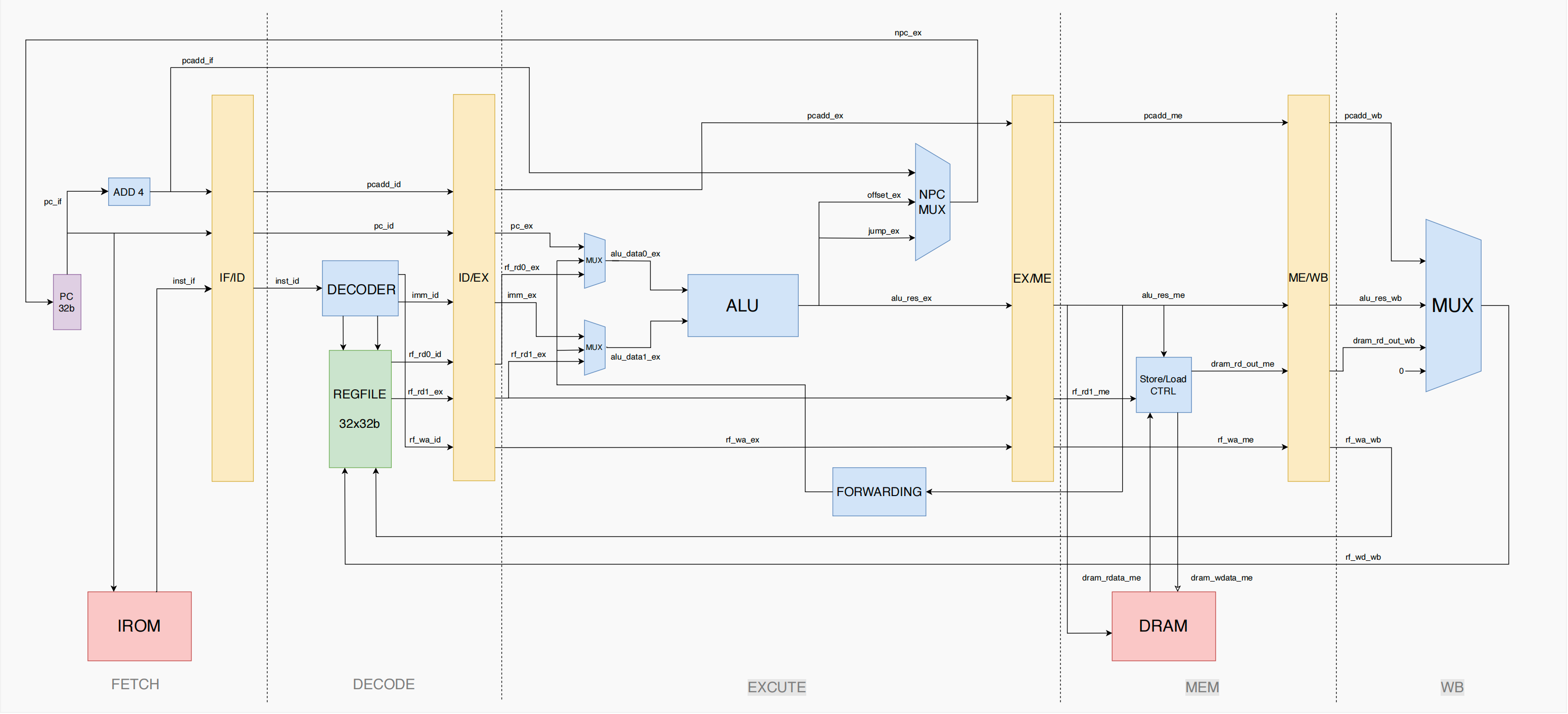


图 1：整体数据通路架构图

# 指令集架构（ISA）

**实现的RV32I指令集，不包含fence,ebreak和ecall指令。**

# 四、模块设计详述

## 1. IF阶段

功能：根据当前PC读取指令，并计算下一个PC  
实现要点：  
- 支持正常+4、自定义分支目标选择器  
- 输出信号：经过flush、stall判断后的输出PC、指令数据  
- 输入信号：Next\_pc

## 2. ID阶段

功能：解析指令，读取寄存器，生成控制信号  
模块：  
- Decoder（根据opcode生成信号）  
- 寄存器堆（支持2读1写）  
输出：控制信号、操作数、立即数、目标寄存器编号等

## 3. EX阶段

功能：执行ALU操作、生成写地址  
子模块：  
- ALU + 控制器（依据ALUOp与funct字段）  
- Forwarding单元（选择是否前递来自MEM/WB的数据）  
信号选择：  
- ALUSrc选择立即数或寄存器  
- Forwarding：MUX控制数据来源

## 4. MEM阶段

功能：访问数据存储器  
行为：  
- 如果是LW：根据ALU地址读取数据  
- 如果是SW：写入数据  
- 数据宽度32bit

## 5. WB阶段

功能：将数据写回寄存器堆  
MUX控制：来自ALU结果或MEM读取数据

# 五、冒险处理机制

## 数据冒险（Data Hazard）

使用Forwarding单元：  
- EX阶段检测来自MEM/WB阶段的依赖  
- ForwardA/B信号生成逻辑基于寄存器号比较

## 控制冒险（Control Hazard）

采用简单的branch prediction，默认向后执行指令，如果发生跳转则利用flush信号清空pipeline中的指令，再重新从跳转指令开始执行。