

NSCSCC2024龙芯杯个人赛设计报告

学校：中国石油大学（华东）

姓名：穆涛洋

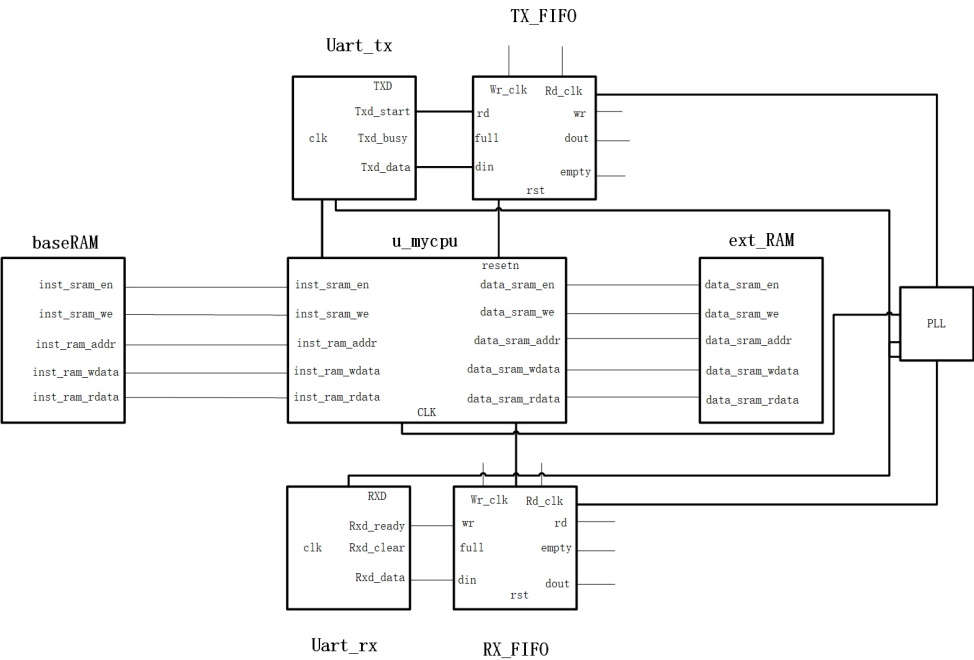
一、设计简介

本CPU是针对“龙芯杯”计算机系统能力培养大赛设计的一款单发射五级流水的LoongArch32无中断无异常的处理器核，实现了8条算术运算，6条逻辑运算指令，4条移位指令，6条分支跳转指令，4条访存指令，共计28条指令，实现了用前递技术解决数据冲突问题，使用阻塞技术解决了base-ram和ext_ram以及串口的数据冲突问题，此外该处理器还采用异步FIFO来解决cpu等待串口的问题。

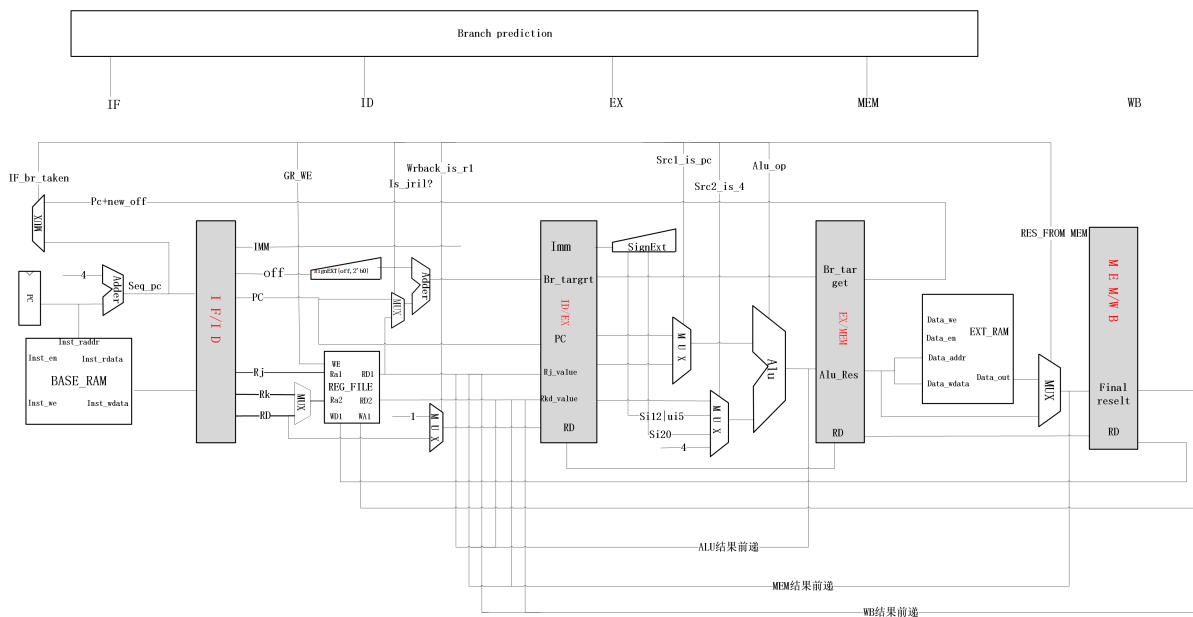
二、设计方案

(一)总体设计思路

本项目顶层文件为龙芯官方提供的thinpad_top，内部实际用到的元器件包含CPU，baseRAM，extRAM，FIFO以及串口，内存和串口控制逻辑未单独成模块而是直接写在thibpad_top里，模块的接口示意图如下：



CPU采用经典五级流水，结构及部分控制信号设置如下图所示。



如图所示，该CPU分为IF, ID, EX, MEM, WB五级，IF阶段进行下条地址的计算，ID阶段进行译码取数以及控制信号(部分跳转指令的跳转信号，各个指令的取数信号，寄存器，ext读写信号，前递判断逻辑，阻塞逻辑)的生成，EX阶段进行alu运算，部分跳转指令(bne, beq, blt)跳转信号的生成以及extram读写信号的发送，MEM阶段进行extram的访问并获取结果，WB阶段将MEM结果写回寄存器。

(二) IF模块设计

IF模块：主要功能为下条指令地址的计算（包含顺序执行指令的下址计算和发生跳转时的下址计算），对baseRAM的读取以及IF/ID阶段的流水段数据发送。对应IF_stage.v文件。

(三) ID模块设计

ID模块：主要功能为接受IF/ID流水段送来的数据，指令的译码，内容包括指令的运算类型，操作数，是否写Extram，是否写寄存器，写的话要写入的地址。对b,bl, jrnl这类跳转目标地址不需要用到从寄存器中读到的数的指令来说跳转信号也在此阶段发出。同时对load_use和其他数据冒险的判断，处理以及ID/EX流水段数据发送也都在此阶段进行。对应ID_stage.v文件。

译码器模块：主要功能为将指令的各个部分进行译码。对应tools.v文件。

Regfile模块：实现了32个32位通用寄存器，有两个读端口和一个写端口，对应regfile.v文件。

(四) EX模块设计

EX模块：主要功能为接受ID/EX送来的流水段数据，根据译码结果对指定的操作数进行指定的运算并给出运算结果，写入的寄存器地址或者访存地址并将bne,beq,blt指令的跳转信号送到IF，将alu的运算结果前递至ID阶段，以及EX/MEM的流水段数据发送。此外由于采用的是同步ram，ram的读写控制信号以及读数据的地址也需在此阶段送给ram，对应MEM.v文件。

计算模块：主要功能为接受计算类型和计算数据，最后送出计算的结果。对应alu.v文件。

(五) MEM模块设计

MEM模块：主要功能为执行访存指令以及接受sram芯片传来的数据，并将访存阶段的最终结果前递回ID阶段，对于ldb指令，则是通过将32位数据移位后再读取低八位。对应MEM_stage.v文件。

(六) WB模块设计

WB模块：将回写阶段将MEM/WB中传来的写回数据按照写回地址写入reg_file里，并将写回数据前递到ID阶段。对应WB_stage.v文件。

(七)内存及串口控制模块设计

内存控制模块：集成在thinpad_top.v文件中，根据访问地址的不同对base_ram，ext_ram的读写使用信号，地址以及数据接口进行赋值，并采取截取的方式将32位cpu地址与20位ram地址建立映射。

串口控制模块：

- 集成在thinpad_top.v文件中，根据访问地址的不同对串口的start信号，cpu和串口的状态信号等进行赋值，并选择是否将数据发送或者是否接受从串口传来的数据等。
- FIFO：对于cpu要发送的数据，先将其按照cpu的时序送入队列，当start信号开始时再按照入队顺序将其按照串口的时序送入发送串口的数据端。

对于CPU要接受的数据，先将其按照串口时序送入队列，当cpu需要用到接受数据时再按照入队顺序将数据按照CPU的时序发出。

三、设计结果

(一) 设计交付物说明

本项目源代码的目录层次如下

```
thinpad_top.srcs
├─ constrs_1
│   └─ new
│       └─ thinpad_top.xdc          //约束文件
├─ sim_1
│   └─ imports
│       │   └─ CFImemory64Mb_bottom.mem
│       │   └─ CFImemory64Mb_top.mem
│       └─ new
│           │   └─ 28F640P30.v
│           │   └─ clock.v
│           │   └─ cp1d_model.v
│           │   └─ flag_sync_cp1d.v
│           │   └─ include
│           │   └─ sram_model.v      //baseram和extram实例化模块（可不关注）
│           └─ tb.sv                //功能仿真
└─ sources_1
    └─ ip                          //所使用的ip核,在vivado的IP Catalog中添加修改（仅需调参，无需阅读代码）
        │   └─ FIFO                //先进先出队列，用于优化串口输入输出
        │   └─ pll_example          //时钟分频模块
        └─ new
            │   └─ async.v           //串口实例化模块（可不关注）
            │   └─ SEG7_LUT.v        //七段数码管模块（可不关注）
            │   └─ thinpad_top.v     //顶层文件
            │   └─ vga.v
            │   └─ tools.v           //包含ID阶段所需的多种译码器
            │   └─ alu.v             //执行各种计算的模块
            └─ Branchpre.v           //分支预测模块
```

```
└─ IF_stage.v          //IF阶段
└─ ID_stage.v          //ID阶段
└─ EX_stage.v          //EX阶段
└─ MEM_stage.v         //MEM阶段
└─ WB_stage.v          //WB阶段
└─ mycpu_top.v         //cpu的顶层模块
└─ regfile.v           //寄存器模块
```

环境配置说明：

1. 请使用Vivado2019.2版本打开本项目，使用其他版本打开可能有兼容问题。
2. 使用前请仔细查看上述项目目录，本人已将需要了解的文件加以注释，本地仿真前请修改tb文件中初始化baseram（指令存储器）的文件地址

（二）设计演示结果

得分	
一级评测	100
二级测评	Fib Finish
三级测评	xx
性能测试	xx

四、参考设计说明

[1]YSJ_36在CSDN标题为[CPU设计实战：版 lab7的接口定义以及流水线框架的设计](#)。

[2]龙芯杯NSCSCC适用于个人赛的工具包（golden_trace工具）中关于内存和串口的控制逻辑。

五、参考文献

[1]汪文祥,邢金璋.CPU设计实战. 北京: 机械工业出版社,2021.