

基于 Nova132A 的 SoC 设计报告

学校 东北大学
姓名 倪仁涛

一、设计简介

本次比赛所提交的 SoC 设计基于本人之前编写并开源的 Nova132A 处理器开发而来。其采用七级流水线，哈弗架构，最高可在 64MHz 的频率下稳定通过测试。该处理器通过团队赛功能测试进行验证，共实现了 53 条指令，可通过前 64 个与 CP0 无关的功能测试点。可在在线测试平台中通过三级功能测试和性能测试，并达到不错的性能。

二、设计方案

（一）总体设计思路

该设计将个人赛的内容划分为处理器和总线两部分进行实现，通过团队赛使用的同步 SRAM 接口进行连接。其总体框架如图 1 所示：

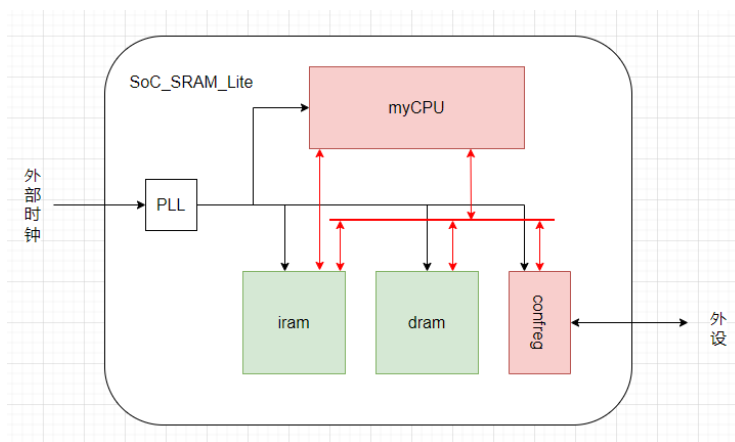


图 1 总体架构图

该架构借鉴了团队赛 SRAM 接口功能测试的 SoC_SRAM_Lite 架构。并针对个人赛需要对 iram 进行数据读写的行为，基于 1x2 bridge 模块，修改地址掩码以及增加新的 SRAM 控制接口，实现了 1x3 bridge 模块。为支持对 iram 的数据读写，还需要处理器在发起对 iram 的数据读写请求时，阻塞取指请求，使访存行为不冲突。

（二）流水线设计

该处理器的流水线结构大致如图 2 所示：

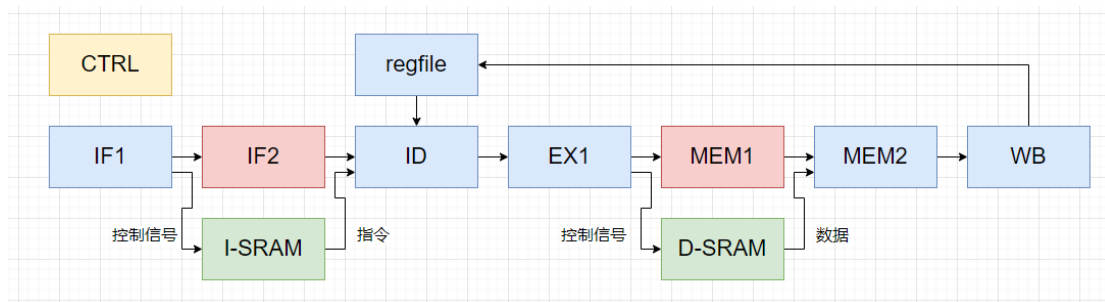


图 2 流水线结构示意图

该流水线与标准五级流水线的区别在于为了给同步 SRAM 腾出时间，增加了 IF2 和 MEM1 这两个流水段，并加上相对应的控制逻辑和前递路径。这种设计使 SRAM 本身也嵌入了流水线中。为使 SRAM 芯片能达到团队赛功能测试中同步 SRAM 的效果，在 SRAM 的前后加入了寄存器来稳定控制信号输入和数据输出，并做到隔离延迟。其结构如图 3 所示：

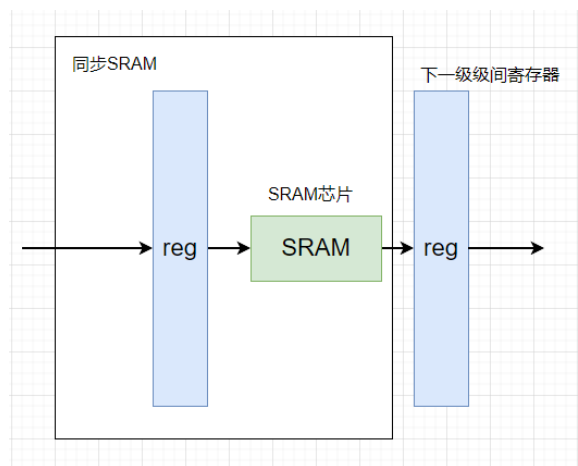


图 3 同步 SRAM 实现方式

增加两段流水级的修改会造成分支损失的增加，访存延迟增加这两个问题。但由于流水级数的增加，处理器也能够运行到更高的频率。其在性能测试中的影响表现如下：

1. 在 STREAM 测试中，每次循环会比正常五级流水线多产生两个周期的空泡。由于实际执行指令中 nop 的占比太高，高频并不能使其超过 50MHz 的五级流水线。
2. 在 MATRIX 测试中，每次跳转会额外产生一次空泡，访存指令会多产生一次空泡。由于该测试循环中的逻辑算术指令较多，跳转和访存的占比相对较少，最终实际执行的指令中 nop 的占比相对 STREAM 测试要少，所以其性能相对五级流水线有了一定提升，但并不明显。
3. 在 CRYPTONIGHT 测试中，由于跳转和访存指令占比的进一步降低，高频带来的提升效果比空泡造成的效率损失更明显，性能有了比较明显的提升。

（三）串口模块设计

串口模块的设计参考了团队赛包中的 confreg 模块的设计，处理器通过向指定地址的寄存器写入参数以实现对其功能控制。串口模块控制示意图如图 4 所示：

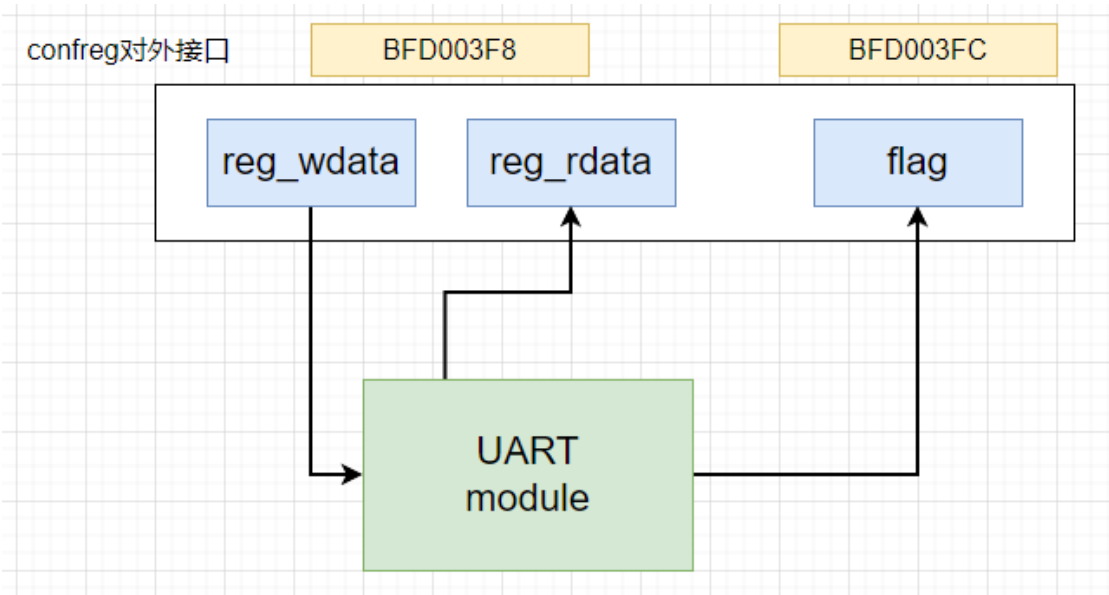
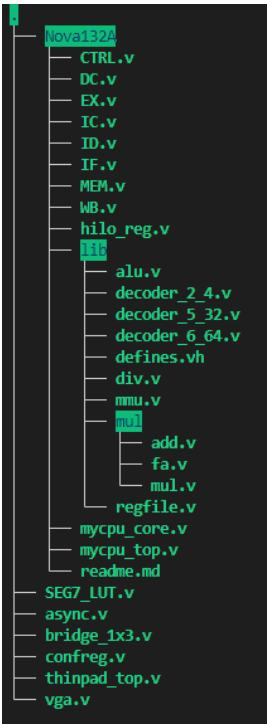


图 4 串口模块控制示意图

三、设计结果

（一）设计交付物说明



该目录位置: fpga_template_utf8_v1.00\thinpad_top.srcs\sources_1\new

(二) 设计演示结果

1. STREAM 测试结果

```
=== Test STREAM ===
Boot message: 'MONITOR for MIPS32 - initialized.'
User program written
  Program Readback:
    1080043c4080053c3000063c21308600050086100400a5240000828cfcffa2acfi
Program memory content verified
Data memory content verified
Test STREAM run for 0.111s
```

图 5 STREAM 测试结果

2. MATRIX 测试结果

```
=== Test MATRIX ===
Boot message: 'MONITOR for MIPS32 - initialized.'
User program written
  Program Readback:
    4080043c4180053c4280063c60000724251800001a00671080400300405203002
Program memory content verified
Data memory content verified
Test MATRIX run for 0.168s
```

图 6 MATRIX 测试结果

3. CRYPTONIGHT 测试结果

```
=== Test CRYPTONIGHT ===
Boot message: 'MONITOR for MIPS32 - initialized.'
User program written
  Program Readback:
    4080043cadde053cefbca534cefa063c0cb0c6341000073c25180400251000000
Program memory content verified
Data memory content verified
Test CRYPTONIGHT run for 0.426s
```

图 7 CRYPTONIGHT 测试结果

四、参考设计说明

该设计中, 存在以下引用、参考和借鉴:

1. 流水线的部分使用本人自行实现且已开源的 Nova132A 的历史版本。
2. Nova132A 中的数据流参考《CPU 设计实战》中的设计, 控制流参考《自己动手写

CPU》。

3. 1x3 bridge 修改自龙芯在团队赛 SRAM 环境中提供的 1x2 bridge 模块。
4. confreg 模块参考了龙芯在团队赛 SRAM 环境中提供的 confreg 模块。
5. 串口模块使用了 thinpad.v 中的示例模块。

五、额外收获

本次实现过程中,本人移植了一些团队赛的工具来降低参赛难度和提高处理器实现速度。希望本人的一些想法和移植的工具能为将来的参赛者提供帮助。

该设计的流水线与总线之间通过团队赛功能测试中的同步 SRAM 接口进行交互,可借助团队赛测试环境对流水线进行验证,且只要流水线接口满足团队赛测试环境的要求,即可进行快速移植。通过该方法可一定程度上降低个人赛难度。

通过在本人第一届实现的处理器上移植团队赛的 golden trace 生成环境,实现了一套适用于个人赛 golden trace 比对环境。可帮助新手进行本地仿真,在三级测试中快速找到自己的问题。

六、参考文献

- [1] 姚永斌. 超标量处理器设计[M]. 北京:清华大学出版社, 2014.
- [2] 汪文祥,邢金璋. CPU 设计实战[M]. 北京:机械工业出版社, 2021.
- [3] 雷思磊. 自己动手写 CPU[M]. 北京:电子工业出版社,2014.