



北京邮电大学

Beijing University of Posts and Telecommunications

计算机系统能力培养大赛 决赛答辩

于海鑫 尹思维

指导教师：周锋

2019-08-20

Contents

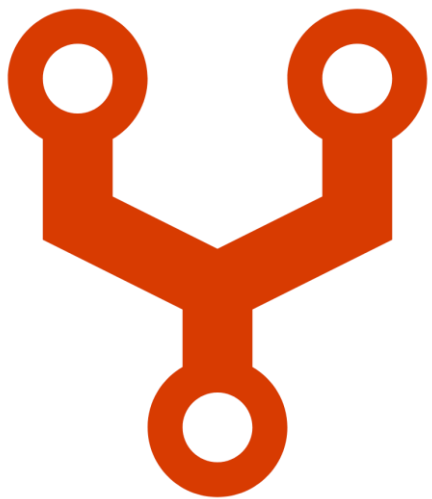
处理器设计

系统软件

总结

处理器设计

处理器设计：特点



顺序双发射结构

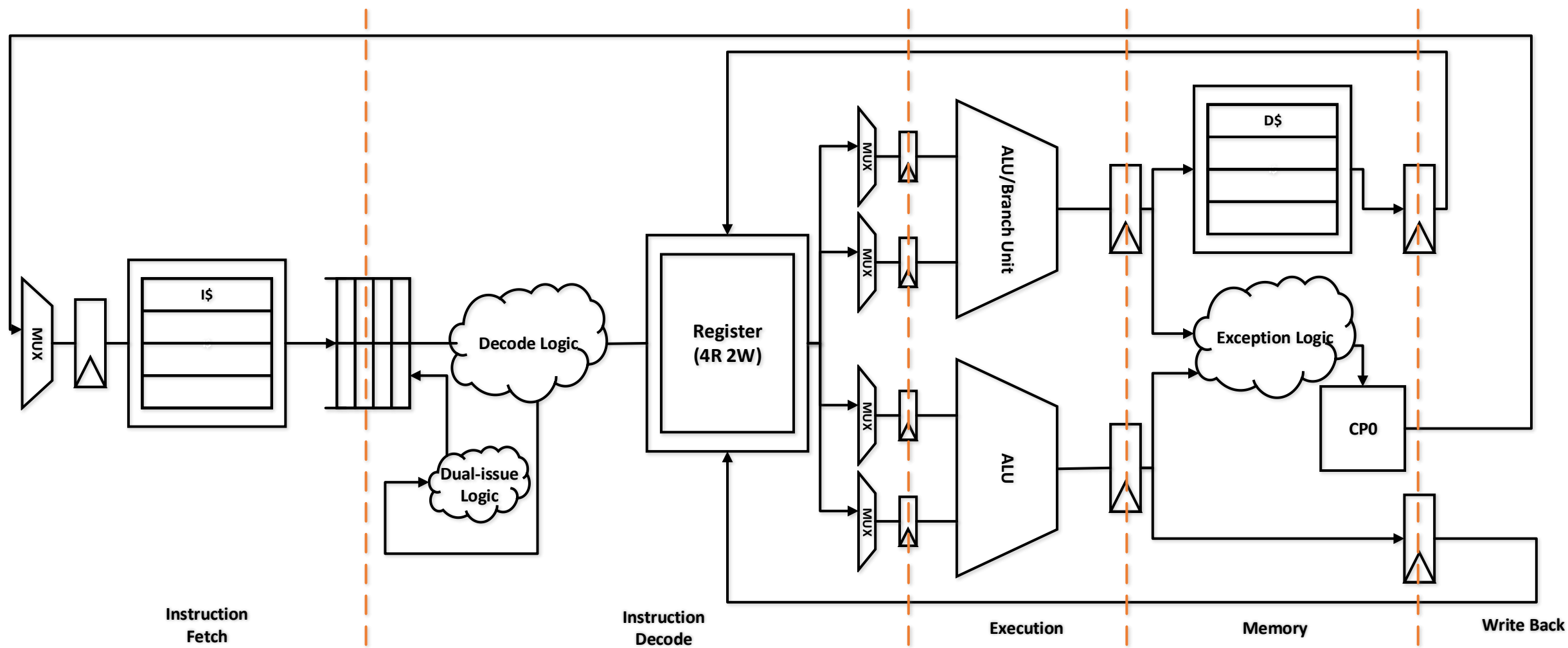


取指-执行分离



写缓冲

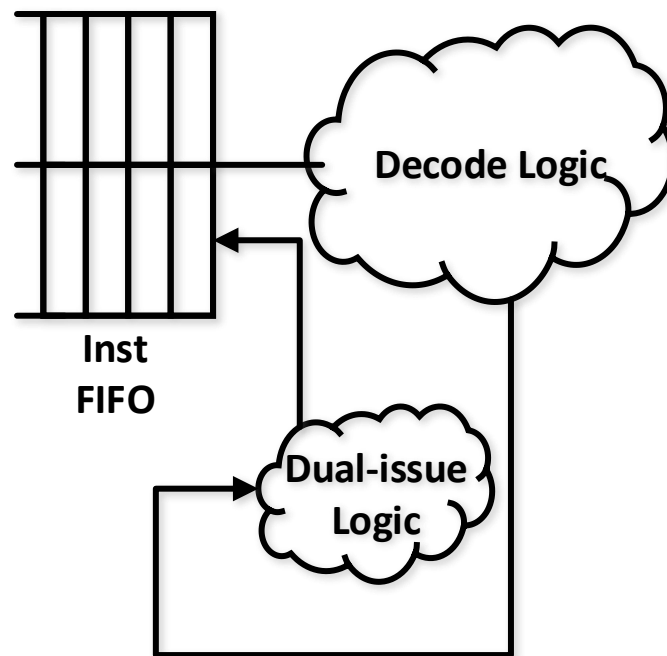
处理器设计：流水线结构



处理器设计：指令调度

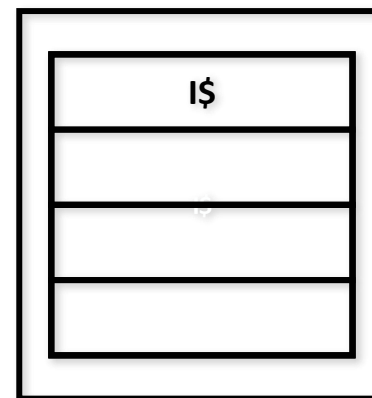
指令缓冲队列与解码器配合双发射引擎完成调度

- 实时检测指令缓冲队列内的前两条指令
- 两条指令间存在真相关时，单发射
- 任一指令为特权指令时，单发射
- 任一指令为乘除法相关指令时，单发射
- 任一指令为访存指令时候，单发射
- 流水线阻塞，不发射
- 其余情况为双发射



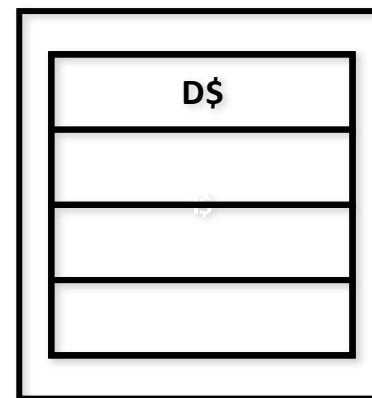
处理器设计：指令缓存

- 8 KB 直接映射指令缓存
- 行大小 64 字节（16 条指令）
- 最大限度利用 AXI3 协议的猝发传输特性
- 采用 Xilinx LogiCORE™ IP Distributed Memory Generator 实现，单周期内返回数据
- 一次命中最多可以返回两条指令
- 支持 CACHE 控制指令



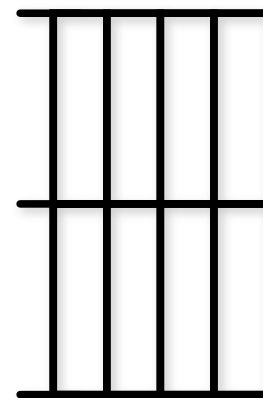
处理器设计：数据缓存

- 8 KB 直接映射数据缓存
- 行大小 64 字节
- 最大限度利用 AXI3 协议的猝发传输特性
- 采用 Xilinx LogiCORE™ IP Distributed Memory Generator 实现，单周期内返回数据
- 一次命中最多处理一条读 / 写请求
- 支持 CACHE 控制指令



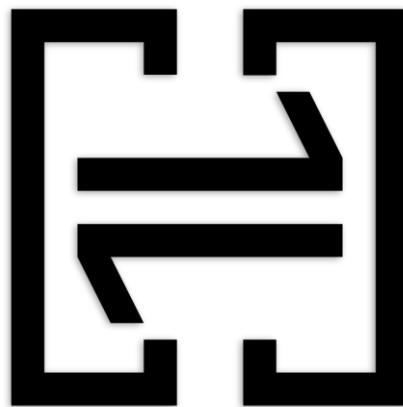
处理器设计：写缓冲

- 减少写入造成的流水线暂停
- 实现为 FIFO
- 由数据缓存控制器一并管理
- 队列未满时单周期内即可完成写入操作
- 任何读请求都会在之前的写入操作完成后开始执行



处理器设计：TLB

- 用于虚实地址转换
- 共 8 个表项，每项可以翻译奇偶两页
- 固定页大小为 4 KB
- 不影响主频
- 支持 TLBWI、TLBWR、TLBP、TLBR 等指令



处理器设计：优化

优化思路：

通过静态时序分析找到关键路径，对关键路径进行优化，在不影响 IPC 的情况下尽可能地提升主频

优化 #1 将虚实地址转换提前到执行流水级进行

优化 #2 将数据缓存命中判断逻辑提前到执行流水级进行

优化 #3 将分支指令的目标地址生成逻辑提前到译码流水级进行

优化 #4 将 ALU 的源数据选择逻辑延后到执行流水级进行

优化 #5 减小 TLB 项数到 8 项

优化 #6 设置 5 个 TLB 实例，分别用于取指(4 个实例)/访存(1 个实例)



处理器设计：性能测试结果@100MHz

序号	测试程序	myCPU			gs132	IPC _{mycpu} /IPC _{g s132}
		上板计时(16进制)		CPU count*2 : SoC cout	上板(16进制)	
		数码管显示 (CPU count)	数码管显示 (SoC count)		数码管显示 (CPU count)	
cpu_clk : sys_clk		100MHz : 100MHz		-	50MHz : 100MHz	-
1	bitcount	1FF97	3FF93	0.999614555	4E3DD2	39.15210702
2	bubble_sort	C58DE	18B223	0.99993636	1EF74EA	40.12725197
3	coremark	23C767	478F35	0.999978037	43399B0	30.06240769
4	crc32	1A3F89	347F78	0.999970353	2A86A88	25.9224079
5	dhrystone	4D581	9AB69	0.999837464	7F000A	26.27227187
6	quick_sort	E94E1	1D2A28	0.999946634	1C65821	31.15882723
7	select_sort	10410E	208282	0.999952125	1B7FFF2	27.07004684
8	sha	DD185	1BA370	0.999943687	1D2E296	33.7871169
9	stream_copy	FA18	1F497	0.999196261	214F0D	34.09541734
10	stringsearch	B4247	1684F5	0.999930209	14286C6	28.6463666

系统软件

系统软件：PMON

```
6. COM3 (USB Serial Port (COM3))
CPU SiriusG @ 99.99 MHz / Bus @ 99.99 MHz
Memory size 128 MB (128 MB Low memory, 0 MB High memory) .
Primary Instruction cache size 8kb (64 line, 1 way)
Primary Data cache size 8kb (64 line, 1 way)

BEV1
BEV2
BEV3
BEV0
BEV in SR set to zero.

NAND DETE
NAND device: Manufacturer ID: 0xec, Chip ID: 0xf1 (Samsung NAND 128MiB 3,3V 8-1
t)
NAND_ECC_NONE selected by board driver. This is not recommended !!
Scanning device for bad blocks
Bad eraseblock 109 at 0x00da0000
NANDFlash info:
erasesize      131072 B
writesize      2048 B
oobsize        64 B
unknow flash type
unknow flash type
Mfg 0, Id 60
PMON>
```

系统软件：uCore

```
HERE
```

```
HERE  
check_boot_pgdir() succeeded!  
----- BEGIN -----  
----- END -----  
check_slab() succeeded!  
kmallocc_init() succeeded!  
check_vma_struct() succeeded!  
check_pgfault() succeeded!  
check_vmm() succeeded.  
sched class: RR_scheduler  
ramdisk_init(): initrd found, magic: 0x2f8dbe2a, 0x000005dc secs  
sfs: mount: 'simple file system' (153/34/187)  
vfs: mount disk0.  
kernel_execve: pid = 2, name = "sh".  
KERNEL EXECVE BEGINUser sh is running!!!
```

```
      /\_/\_/\_/\_/\_/\_/  
     /\_/\_/\_/\_/\_/\_/  
    /\_/\_/\_/\_/\_/\_/  
   /\_/\_/\_/\_/\_/\_/  
  /\_/\_/\_/\_/\_/\_/  
 /\_/\_/\_/\_/\_/\_/  
\_\_,\_\_/\_\_/\_\_/\_\_/\_\_  
  
ucore for SiriusG
```

```
$ █
```

系统软件：Linux

挑战从 Linux 主线内核直接移植

CPU

- #1 补全 Linux 运行所需的指令
- #2 补全 CP0 中的寄存器
- #3 支持 13 种异常处理

软件

- #1 添加早期初始化代码 init.c
- #2 添加设备树描述文件 sirius_fin.dts
- #3 添加默认内核配置文件
- #4 利用 feature 去除机制删减 Linux 内核对于指令以及 CP0 寄存器的依赖
- #5 修改 Makefile，避免产生 branch-likely 指令
- #6 修改 asm/bug.h，防止产生 tne 指令
- #7 使用 QEMU 进行验证

遗憾

上板后由于未知原因无法正常启动

总结

61

初赛性能分 61.377
位列第 2 名

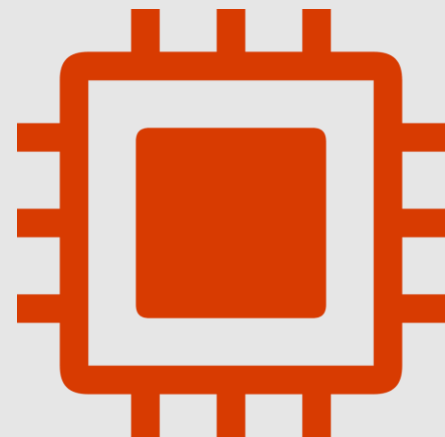


10k

代码 + 文档总行数约为
10000 行

141

141 天，从对 System Verilog
一窍不通到实现能运行操作系
统的 MIPS CPU



总结：时间线



总结：实现的指令

大赛要求的指令

算术运算指令 ADD ADDI ADDIU SUB SUBU SLT SLTI SLTU SLTIU DIV
DIVU MULT MULTU SLLV SLL SRAV SRA SRLV SRL

逻辑运算指令 AND ANDI LUI NOR OR ORI XOR XORI

分支跳转指令 BEQ BNE BGEZ BGTZ BLEZ BLTZ BGEZAL BLEZAL J JAL
JR JALR

数据移动指令 MFHI MFLO HTHI HTLO

自陷指令 BREAK SYSCALL

访存指令 LB LBU LH LHU LW SR SH SW

特权指令 ERET MFCO MTCO

额外实现的 MIPS32 指令

CLO CLZ

MADD MADDU MSUB SUBU MUL

MOVN MOVZ

PREF SYSC WAIT

LWL LWR SWL SWR

CACHE

总结：实现的CP0寄存器

寄存器号	Sel.	名称	描述
0	0	Index	Index into the TLB array
1	0	Random	Randomly generated index into the TLB array
2	0	EntryLo0	Low-order portion of the TLB entry for even-numbered virtual pages
3	0	EntryLo1	Low-order portion of the TLB entry for odd-numbered virtual pages
4	0	Context	Pointer to page table entry in memory
6	0	Wired	Controls the number of fixed ("wired") TLB entries
8	0	BadVAddr	Reports the address for the most recent address-related exception
9	0	Count	Processor cycle count
10	0	EntryHi	High-order portion of the TLB entry
11	0	Compare	Timer interrupt control
12	0	Status	Processor status and control
13	0	Cause	Cause of last general exception
14	0	EPC	Program counter at last exception
15	0	PRId	Processor identification and revision
16	0	Config	Configuration register
16	1	Config1	Configuration register 1



北京邮电大学
Beijing University of Posts and Telecommunications

谢谢，
请各位专家评委指教。