

高性能自研处理器物理设计频率提升方法

何小威 乐大珩 郭 维 隋兵才 邓 全

(国防科技大学计算机学院 长沙 410073)

(先进微处理器芯片与系统重点实验室(国防科技大学) 长沙 410073)

(xw_he1980@163.com)

Promoting Frequency Method for Our Own High Performance Processor Physical Design

He Xiaowei, Yue Daheng, Guo Wei, Sui Bingcai, and Deng Quan

(College of Computer Science and Technology, National University of Defense Technology, Changsha 410073)

(Key Laboratory of Advanced Microprocessor Chips and Systems (National University of Defense Technology), Changsha 410073)

Abstract Promoting core's frequency is the key method for increasing performance of processor. It is hard to achieve high frequency for processor core by traditional physical design flow. Based on main place and route tools, with the same process, comparable implementation area and power consumption, our own processor core frequency can be promoted by about 30% compared with original design at signoff stage, by employing manually written block netlist, logic and physical design co-optimization, custom routing rule optimization and physical design methodology adjustment.

Key words place and route; co-optimization; physical design; signoff; frequency

摘 要 提升处理器核的频率是提升处理器性能的重要手段。传统的物理设计流程难以实现高主频的处理器核。基于业界主流的布局布线工具,通过嵌入手工定制部件的网表、逻辑和物理设计协同优化、优化定制布线规则、优化物理设计方法学等组合策略。在相同工艺、面积、功耗对等条件下,达到流片签核要求时,自研处理器核物理设计频率比原始设计可提升约 30%。

关键词 布局布线;协同优化;物理设计;签核;频率

中图法分类号 TN47

在集成电路设计发展过程中,高性能微处理器的性能提升一直是业界追逐的热点目标。提升微处理器的频率将给提升性能带来直接好处。当前先进制造工艺存在成本过高、研发进度放缓等问题,如何采用较低的工艺制程实现较高主频的微处理器是众多厂家面临的主要挑战。处理器核逻辑复杂、时序收敛难度高,往往是处理器中频率最高的模块。传统的半定制物理设计流程和方法对提升处理器核的频率

收效甚微。为了研制实现更高主频的微处理器,已有不少研究对特定处理器核的物理设计进行了改进优化:文献[1-2]针对 Cortex 系列核进行了物理设计优化,在不同工艺条件下实现了要求的频率指标;文献[3]则针对 RISC CPU 核进行了物理设计;文献[4-5]则重点针对时钟树进行设计优化;文献[6]改进优化了深亚微米下物理设计方法学;文献[7-8]研究层次化的物理设计技术以解决芯片的时序收敛问题;文献[9]

收稿日期: 2023-11-27; 修回日期: 2024-03-06

基金项目: 国防科技大学科研计划项目(ZK22-05); 全军共用信息系统装备预研专用技术项目(31513010105)

This work was supported by the Scientific Research Project of National University of Defense Technology (ZK22-05) and the Specific Technology of Advanced Research for Military Information System Equipments (31513010105).

通信作者: 郭维(wineer_guowei@nudt.edu.cn)

通过机器学习来优化时序,进而提升 CPU 的性能;文献[10]针对高性能 CPU 核进行了频率提升和功耗优化技术研究,改进了传统的物理设计流程;文献[11-12]分别在不同工艺下实现了不同频率的处理器核.这些研究针对特定的处理器核在一定程度上提升了实现频率.为了实现极致的处理器频率目标,本文针对自研的处理器核,基于业内主流的商用工具采取多种措施进行了针对性的实验和优化,包括规整部件网表的手工定制、逻辑和物理设计协同优化、布线规则定制优化、针对时序的物理设计流程与方法学优化等,大幅度提升了处理器核的签核频率.

1 自研核基本信息

自研处理器核全部逻辑单元数目接近 400 万,包含运算部件和 1MB 二级缓存(cache),其中存储体(memory)种类多且占用面积大,基于国内工艺在典型情况下预定的流片频率为 2 GHz,核的实现面积为 2500 um \times 2500 um,实验布局如图 1 所示.

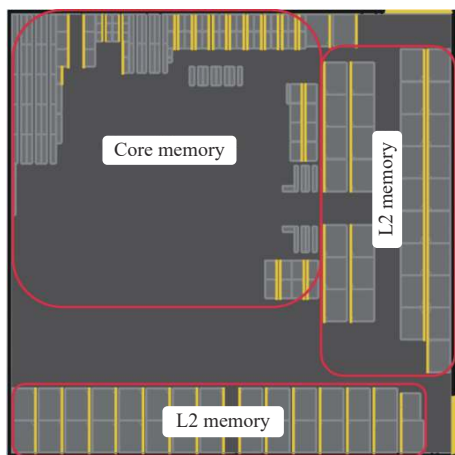


Fig. 1 Floorplan of our own processor core
图 1 自研处理器核布局

2 物理设计频率提升方法

对自研处理器核采用传统的物理设计流程进行了时序评估实验后,频率离预期目标还有较大差距,关键路径时序难以收敛.为此,从物理设计流程和方法学等方面进行了多种改进和优化.

2.1 寄存器文件部件网表手工定制

门级网表描述了搭建电路结构的各种逻辑门和触发器的连接关系,是物理设计必须的输入文件.根据传统的设计流程,逻辑代码经过工具综合得出物

理设计需要的门级网表.但综合工具在保证功能正确的情况下,往往不能智能地识别规整的模块电路,综合得出的门级网表规模在同等时序条件下并不一定是最精简的,那么在布局阶段可能需要部署更复杂的电路结构、占用更多的实现面积,增加了频率提升难度.

寄存器文件(register file, RF)是处理器设计中的关键存储逻辑,对处理器的性能具有至关重要的作用.为了满足性能需求,RF 通常具有超过 2 组的读写端口,主要逻辑包括:读写端口的地址译码、存储阵列以及相关的旁路控制功能.在追求高性能的通用处理器中,通常 RF 的规模比较大,其实现质量对于处理器的物理设计具有重要影响.RF 的读写地址译码和存储阵列属于比较规则的功能逻辑,随着设计规模的增大,译码部分的逻辑单元数目也呈指数级增长,相比于人工定制,商用综合工具难以发掘 RF 逻辑的规整性.因此,为了简化电路结构并节省更多的面积,缓解复杂路径对频率提升的限制,可以在保证功能正确的前提下跳过综合工具,从逻辑功能上直接手工定制核内关键部件 RF 的门级网表,用尽可能少的逻辑单元搭建出功能正确的电路.手工定制 RF 网表的好处是比全定制设计速度快、比综合工具产生的网表规模更小,而且结构规整、接口时序更好.无论是嵌入到核逻辑中进行综合,还是直接对 RF 进行物理设计后作为核的子模块实现,都将对核的频率提升带来巨大好处.图 2 显示了 RF 模块使用工具综合得出的门级网表完成物理设计后的布局图,而图 3 则为 RF 模块手工定制门级网表完成物理设计后的布局图.

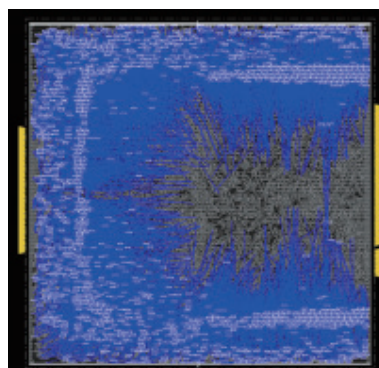


Fig. 2 Floorplan of physical design for RF synthesis netlist
图 2 RF 综合网表物理设计布局图

由表 1 可以明显看出,在相同的面积和端口位置不变的情况下,使用相同的时序约束完成物理设计后,手工网表的实现密度(density)和单元数明显降低,

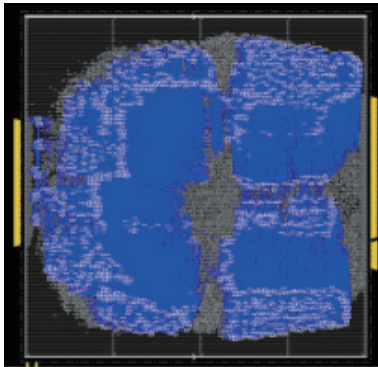


Fig. 3 Floorplan of physical design for RF manual netlist
图 3 RF 手工网表物理设计布局图

Table 1 Physical Implementation Results of RF Synthesis Netlist and RF Manual Netlist

表 1 RF 综合网表和 RF 手工网表的物理实现结果

核	密度/%	时序			单元数目
		in2reg/ns	reg2out/ns	reg2reg/ns	
RF 综合网表	75	-0.171	-0.074	0	44 218
RF 手工网表	60	-0.125	-0.049	0	33 188

接口时序 in2reg 和 reg2out 也普遍较优,内部时序 reg2reg 均没有违反.表 1 汇总了各项实验参数.

将手工定制的 RF 门级网表例化 3 次后直接内嵌到自研核的逻辑中,使用商用工具对核综合后进行物理设计,实验结果与不采用手工定制 RF 网表的物理实现结果对比如表 2 所示.可以看出,在相同实现面积下,定制 RF 网表后核的 reg2reg 时序比采用综合 RF 网表整体更优,违反值从-26 ps 优化到-9 ps,违反条数也有大幅度下降,同时实现密度降低 1.3 个百分点.当然,定制 RF 门级网表核物理实现后单元数目并没有大幅度降低,是因为物理实现时还要考虑 RF 网表和核的接口之间的时序优化,但是总体来讲,RF 定制网表带来了更好的时序和更低的实现密度.

Table 2 Physical Implementation Results of Our Own Cores of RF Custom Netlist and RF Synthesis Netlist

表 2 RF 定制网表和 RF 综合网表自研核的物理实现结果

核	面积/ mm ²	密度/ %	reg2reg/ ns	reg2reg 违反条数	单元 数目
定制 RF 网表	6.25	54.6	-0.026	635	3 863 485
综合 RF 网表	6.25	53.3	-0.009	360	3 834 328

2.2 逻辑和物理设计协同优化

通常在物理设计流程中,逻辑设计仅负责提供设计输入文件(网表和时序约束),并不参与物理设计.物理设计则根据输入文件进行布局布线,不会深

究电路的逻辑功能.处理器核通路功能复杂、关键路径的组合逻辑级数一般都比较大的,有的甚至超过 30 级,而且可能存在前后级路径逻辑级数都很大的情况,这给高主频物理实现带来极大挑战.如何遵从数据流走向,完成相对合理的布局,是高主频物理设计的关键所在.这需要进行逻辑和物理设计的协同优化,即物理设计中的关键时序问题需要从逻辑设计角度进行优化,例如在非关键路径上加站、调整优化站与站之间的逻辑、变换逻辑功能的实现方式、得到更加合理的流水线结构等,目的是为关键路径进一步挤出时序余量,再经过布局布线实现更高的主频.

图 4 是核的运算模块的布局图,包括多种不同类型的静态随机存取存储器(static random access memory, SRAM),每类 SRAM 的逻辑作用不同,时序收敛难度也不尽相同.基于 SRAM 的当前时序,SRAM 相关路径的扇入、扇出情况,以及 SRAM 路径的前级和后级路径的时序违反情况进行综合考虑后,对部分 SRAM 的位置按照数据流进行了几轮调整优化,使得 SRAM 位置更加靠近对应的逻辑和寄存器.优化后运算模块布局的总体时序违反量有了大幅度下降,reg2reg 总体违反值从-69.553 ns 降低到-53.388 ns,违反条数从 7 375 降低到 5 732,实现密度也缩小了 1 个百分点.由此可见,在逻辑设计协同辅助下,物理设计不仅能更准确地顺应模块间的数据流,还能使模块内部的宏单元摆放也尽可能遵循数据流导向,减少单元数量并避免绕线资源的浪费.

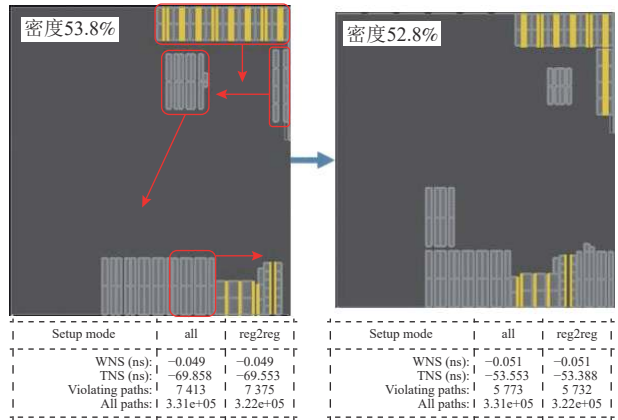


Fig. 4 SRAM location adjustment by logic and physical co-optimization
图 4 逻辑物理协同优化下的 SRAM 位置调整

此外,逻辑设计还可以根据布局后模块的实际位置和面积对逻辑进行适当地删减或优化,调整出与物理设计布局更契合的设计,更有利于频率的提升.

2.3 布线规则定制优化

布线规则是指绕线时需要遵循的规则. 为了提高主频设计下时序能够更好地收敛, 一般要对某些线制定特殊的布线规则(non default rule, NDR), 对指定的线使用特殊的绕线规则进行绕线. 传统的物理设计流程中, 一般只对时钟线制定 NDR, 使时钟线的延时和受到的干扰尽可能小, 以保证时钟质量. 然而时序优化时也可根据时序违反情况, 对部分信号线精准设定 NDR, 解决关键路径的时序问题, 达到提升频率的最终目的.

NDR 一般是通过增加金属线的绕线宽度和间距, 提高绕线金属层数, 以牺牲绕线资源为代价改进时序和性能. 增加绕线宽度就降低了绕线电阻, 所以会带来路径延时的降低; 增加绕线间距, 可以减小密集绕线之间的串扰, 同样可以减小路径延时; 提高绕线金属层数, 因为高层金属电阻低, 所以用高层金属绕线也会降低路径延时. 无论是增加线宽还是加大绕线间距都会占用更多的绕线资源, 在实际使用时需要对关键路径所处区域的绕线资源进行充分的分析评判, 对所处位置以及周边的布线资源提前干预和规划. 盲目地对大量的绕线设置 NDR 肯定是不可取的, 极有可能带来绕线不通的问题甚至引起时序恶化. 通过人工精确进行局部布局密度控制, 或者绕线密度控制可以在某个区域保留足够的绕线资源.

需要注意的是, 在实际绕线时工具并不一定能完全遵守所设定的 NDR. 例如, 设定某些线按照 2 倍间距来绕线, 但是因为绕线资源的限制, 工具并不能

保证完全按照 2 倍间距来绕完这些线, 部分线可能还是按照单倍间距来进行绕线. 为此, 我们提出了一种新的方法, 即对某些金属层的布线轨道(track)按照 2 倍间距重新排布, 例如图 5 所示, 原来偶数布线轨道(图中白色竖线标识)就不存在了, 只有奇数布线轨道. 因为绕线只能在布线轨道上进行, 这样做就可以使得指定的线能够在这个布线层上全部按照 2 倍间距完成绕线, 天然达到 2 倍绕线间距的 NDR 效果.

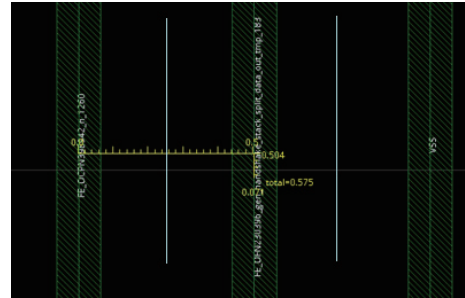


Fig. 5 Routing after re-assigned tracks

图 5 重新排布轨道后的绕线

图 6 是没使用 NDR 和使用了 NDR 这 2 种策略时某一条关键时序路径优化前后的数据延时比较. 可以看出使用了 NDR 之后串扰引起的附加(delta)延时大幅度降低, 经过优化后整条数据路径的整体延时减小了 94 ps (1.007 ns - 0.913 ns = 0.094 ns)!

在进行自研核的时序优化时, 根据布线资源、时序违反瓶颈、信号串扰等因素综合设置 NDR 后, 部分信号线在使用 2 倍间距绕线后串扰影响大幅度减小, 关键数据路径延时明显降低, 对时序收敛起到了

	Delta	Incr	Path
clock coreclk (rise edge)		0.000	0.000
clock network delay (propagated)		0.659	0.659
U_XJG1/g_core_0_g_big_u_xjg_core/u_vcpu/u_cpu/u10l2c/u_b1/u_data/ram_reg_0_20/CK		0.000	0.659
U_XJG1/g_core_0_g_big_u_xjg_core/u_vcpu/u_cpu/u10l2c/u_b1/u_data/ram_reg_0_20/Q		0.029%	0.688
placeFE OFC1798274 ram_data_in_m4_q_0_20_1/I	0.000	0.001%	0.689
placeFE OFC1798274 ram_data_in_m4_q_0_20_1/Z		0.015%	0.704
placeFE OFC1575729 ram_data_in_m4_q_0_20_1/I	0.006	0.035%	0.739
placeFE OFC1575729 ram_data_in_m4_q_0_20_1/Z		0.017%	0.756
placeFE OFC1779222 ram_data_in_m4_q_0_20_1/I	0.019	0.056%	0.812
placeFE OFC1779222 ram_data_in_m4_q_0_20_1/Z		0.012%	0.824
postrouteFE PDS2173 FE_OFN829187 ram_data_in_m4_q_0_20_1/I	0.027	0.048%	0.872
postrouteFE PDS2173 FE_OFN829187 ram_data_in_m4_q_0_20_1/Z		0.018%	0.890
placeFE OFC1779223 ram_data_in_m4_q_0_20_1/I	0.007	0.010%	0.900
placeFE OFC1779223 ram_data_in_m4_q_0_20_1/Z		0.023%	0.923
ctsFE OFC192953 FE_OFN1593540 n/I	0.004	0.016%	0.939
ctsFE OFC192953 FE_OFN1593540 n/Z		0.027%	0.966
SRAM_IN_BUF_17320/I	0.012	0.025%	0.991
SRAM_IN_BUF_17320/Z		0.016%	1.007
U_XJG1/g_core_0_g_big_u_xjg_core/u_vcpu/u_cpu/u10l2c/u_b1/u_data/u_sram3/D[20]	0.000	0.000%	1.007
	Delta	Incr	Path
clock coreclk (rise edge)		0.000	0.000
clock network delay (propagated)		0.670	0.670
U_XJG1/g_core_0_g_big_u_xjg_core/u_vcpu/u_cpu/u10l2c/u_b1/u_data/ram_reg_0_20/CK		0.000	0.670
U_XJG1/g_core_0_g_big_u_xjg_core/u_vcpu/u_cpu/u10l2c/u_b1/u_data/ram_reg_0_20/Q		0.030%	0.700
placeFE OFC1798274 ram_data_in_m4_q_0_20_1/I	0.000	0.001%	0.701
placeFE OFC1798274 ram_data_in_m4_q_0_20_1/Z		0.019%	0.720
placeFE OFC1575729 ram_data_in_m4_q_0_20_1/I	0.004	0.023%	0.743
placeFE OFC1575729 ram_data_in_m4_q_0_20_1/Z		0.023%	0.766
placeFE OFC1779222 ram_data_in_m4_q_0_20_1/I	0.004	0.021%	0.787
placeFE OFC1779222 ram_data_in_m4_q_0_20_1/Z		0.023%	0.810
placeFE OFC1779223 ram_data_in_m4_q_0_20_1/I	0.010	0.031%	0.841
placeFE OFC1779223 ram_data_in_m4_q_0_20_1/Z		0.018%	0.859
SRAM_IN_BUF_17320/I	0.009	0.037%	0.896
SRAM_IN_BUF_17320/Z		0.017%	0.913
U_XJG1/g_core_0_g_big_u_xjg_core/u_vcpu/u_cpu/u10l2c/u_b1/u_data/u_sram3/D[20]	0.000	0.000%	0.913

Fig. 6 Comparison of critical data path delay with NDR and without NDR

图 6 带 NDR 和不带 NDR 时关键数据路径延时比较

关键作用. 表 3 显示了精确使用 NDR 和不使用 NDR 时自研核绕线阶段的时序优化情况, reg2reg 时序优化效果非常明显, 设计规则检查 (design rule check, DRC) 数量并无明显增多, 在可控的范围内.

Table 3 Timing Optimization Comparison of Our Own Core Used with NDR and Without NDR

表 3 使用 NDR 和不使用 NDR 自研核的时序优化比较

核	面积/ mm ²	密度/ %	reg2reg/ ns	reg2reg 违反条数	DRC 数量
不使用 NDR	6.25	53.4	- 0.118	654	56
使用 NDR	6.25	53.3	- 0.021	125	62

2.4 针对 setup/hold 时序的物理设计方法学优化

针对高主频的设计目标, 在配置物理设计流程时应以提高频率为最高优先级. 因此在传统的流程基础上, 优先开启对建立时间 (setup) 时序更友好的设置.

首先, 在物理设计中有用偏斜 (useful skew) 是常被用来处理 setup 违反的手段之一. 而为了最大限度地修复 setup 违反, 为提升频率挤出更多的空间, 可选择将工具的 useful skew 设置为极限, 可以最大化地借用 useful skew 来优化 setup 时序, 这对于提升核的频率尤为重要.

表 4 显示了自研核在不开启 useful skew 时布局布线后的 setup 时序结果, 表 5 为自研核开启 useful skew 为极限时布局布线后的 setup 时序结果. 实验时, 在典型端角 (typical corner) 和慢速端角 (worst corner) 下分别合理设置了不同的频率. 可以看出, 在同样的流程下这 2 个工艺角开启 useful skew 为极限时, reg2reg 时序的最大违反 (worst negative slack, WNS) 和总违反

Table 4 setup Timing of Place and Route Without useful skew

表 4 useful skew 不开启时布局布线的 setup 时序 ns

setup	reg2reg	in2reg	reg2out
典型端角	WNS:- 0.119	WNS:- 0.051	WNS:- 0.042
	TNS:- 29.345	TNS:- 15.312	TNS:- 10.215
慢速端角	WNS:- 0.123	WNS:- 0.038	WNS:- 0.045
	TNS:- 28.198	TNS:- 13.115	TNS:- 10.221

Table 5 setup Timing of Place and Route with Extreme useful skew

表 5 useful skew 开启为极限时布局布线的 setup 时序 ns

setup	reg2reg	in2reg	reg2out
典型端角	WNS:- 0.021	WNS:- 0.055	WNS:- 0.049
	TNS:- 0.372	TNS:- 15.977	TNS:- 10.235
慢速端角	WNS:- 0.018	WNS:- 0.040	WNS:- 0.050
	TNS:- 0.286	TNS:- 13.228	TNS:- 10.238

(total negative slack, TNS) 都小很多, setup 时序收敛速度非常快, 同时 in2reg 时序和 reg2out 时序基本相当.

当然, useful skew 也会带来保持 (hold) 时间违例的增大、增多, 而修复这些 hold 违例需要插入更多的单元, 进而增加功耗. 对于极限提升频率来讲, 使用 useful skew 的代价是可以忍受的, 甚至可以忽略, 因为并不是所有的路径都使用了 useful skew. 如果不使用 useful skew, 几乎没有可能将频率提升到极致. 表 6 中列出了使用 useful skew 和不使用 useful skew 情况下, 完成布局布线后典型端角下的 reg2reg setup 和 hold 时序违反结果, 可以看出, 不使用 useful skew 情况下 setup 时序就很难收敛了. 虽然使用 useful skew 后 hold 的最大违反值和违反条数都有增加, 但是只要可以通过其他手段进行修复, 就不存在设计风险. 而且, 修复 hold 插入的延迟单元功耗很小, 对处理器的整体功耗影响不大.

Table 6 setup and hold Violation Comparisons with Extreme useful skew and Without useful skew

表 6 开启极限 useful skew 和不开启 useful skew 时 setup 和 hold 的违反比较 ns

useful skew	reg2reg setup	reg2reg hold
关闭	WNS:- 0.119	WNS:- 0.215
	TNS:- 29.345	TNS:- 546.89
开启	WNS:- 0.021	WNS:- 0.363
	TNS:- 0.372	TNS:- 1 138.215

其次, 由于布局布线工具报出来的 hold 时序违反并不十分精确, 最终的 hold 时序违反应以静态时序分析工具得出的结果为标准, 于是在物理设计流程的布局布线阶段应该不进行 hold 违例的修复. 这样既能节省布局布线时间, 又能防止布局布线工具过度修复 hold 导致 setup 时序恶化, 从而限制频率提升.

表 7 给出了在布局布线阶段使用 useful skew, 开启修复 hold 和关闭修复 hold 时典型端角下的 setup/hold 违反对比结果. 由此可见, 物理设计工具在布局布线阶段不修复 hold 可使得 setup 时序更容易收敛, 而 hold 违反尚在可控制范围内, 这 2 种方案的 hold

Table 7 setup and hold Timing Comparison Under Typical Corner After Place and Route with /Without Fixing hold

表 7 开启和关闭修复 hold 布局布线后典型端角下 setup 和 hold 的时序对比 ns

布局布线策略	reg2reg setup	reg2reg hold
开启修复 hold	WNS:- 0.032	WNS:- 0.079
	TNS:- 5.868	TNS:- 59.352
关闭修复 hold	WNS:- 0.021	WNS:- 0.363
	TNS:- 0.372	TNS:- 1 138.215

违反都可通过其它工具单独修复,快速达到收敛水平.

总之,和传统的物理设计流程相比,需要优先考虑解决 setup 违反,然后再精确修复 hold 违反,在此过程中保持 setup 不恶化,再针对性地进行功耗和面积优化,只有这样才能更好地提升频率.

3 自研核整体物理实现结果

整合上述优化实现手段,参照流片厂商推荐的签核设置,经过实验优化完成了自研处理器核的布局布线、时钟树生成、setup 和 hold 违例修复.除了时序收敛之外,还完成了其他物理设计步骤,例如物理规则检查、功能等价性检查和电压降(IR-drop)检查等内容,达到了流片水准.按照悲观的翻转率设置,在典型端角下自研处理器核的功耗达到 5 W.但是静态 IR-drop 仅有 1.8%,远低于签核标准,如图 7 所示;动态 IR-drop 为 17.59%,满足签核要求,如图 8 所示.

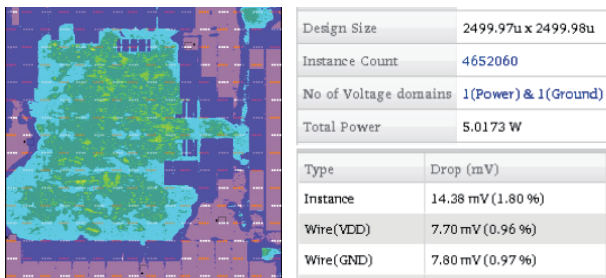


Fig. 7 Static IR-drop result
图 7 静态 IR-drop 结果

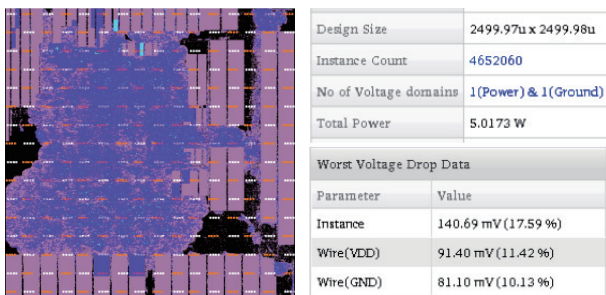


Fig. 8 Dynamic IR-drop result
图 8 动态 IR-drop 结果

自研处理器核最初预定的频率目标为 2 GHz,在没有采用手工定制 RF 网表、没有针对性地使用 NDR,数据流优化也做得不够完善,但是同样使用了 useful skew、也进行了 hold 修复的情况下,典型端角下物理实现频率仅能达到 1.8 GHz.在 SoC 流片时,该自研核最终的签核频率在典型端角下达到 2.35 GHz,和原来的实现频率相比提升了 30%.

4 结 论

针对自研的某处理器核,通过多种设计策略实现了核频率的大幅度提升,为类似处理器核的高主频物理设计提供了有益的参考.

作者贡献声明:何小威提出了多种频率提升的方法并撰写和修改论文;乐大珩完成论文部分实验并修改论文;郭维撰写论文引言,并对论文提出修改意见;隋兵才完成论文中部分实验;邓全校验论文文字.

参 考 文 献

- [1] Zhang Jing. Physical design of high-performance Cortex-A55 CPU module based on TSMC12nm process[D]. Xi'an: Xidian university, 2021 (in Chinese)
(张静. TSMC 12 nm 工艺下高性能 Cortex-A55 CPU 模块的物理设计[D]. 西安: 西安电子科技大学, 2021)
- [2] Wang Yingxiang, Sun Yan, Wang Weiqi. Implementation of high speed ARM Cortex-A17 quad-core CPU using Cadence Innovus[J]. China Integrated Circuit, 2018, 27(12): 37-43 (in Chinese)
(王翔翔, 孙艳, 王伟奇. 基于 Cadence Innovus 的高速 4 核 ARM Cortex-A17 CPU 物理实现 [J]. 中国集成电路, 2018, 27(12): 37-43)
- [3] Wu Weixian, Zhou Jianyang, Xu Weijian, et al. The physical design of a RISC CPU core based on the SVP[J]. Microelectronics & Computer, 2005, 22(3): 162-169 (in Chinese)
(吴伟贤, 周剑扬, 许伟坚, 等. 基于硅虚拟原型的 RISC CPU 核物理设计[J]. 微电子学与计算机, 2005, 22(3): 162-169)
- [4] Wang Hong. Physical design and timing optimization based on CPU chips[D]. Beijing: Beijing University of Technology, 2020 (in Chinese)
(王红. 基于 CPU 芯片的物理设计与时序优化[D]. 北京: 北京工业大学, 2020)
- [5] Wang Bing, Peng Ruihua, Fu Yuxi. Clock tree design method with front-end and back-end combined[J]. Computer Engineering, 2008, 34(12): 227-229, 232 (in Chinese)
(王兵, 彭瑞华, 傅育熙. 前后端协同的时钟树设计方法[J]. 计算机工程, 2008, 34(12): 227-229, 232)
- [6] Zeng Hong. IC physical design methodology research under DSM[J]. China Integrated Circuit, 2010, 19(2): 30-35, 49 (in Chinese)
(曾宏. 深亚微米下芯片后端物理设计方法学研究[J]. 中国集成电路, 2010, 19(2): 30-35, 49)
- [7] Wei Shaoxiong. Hierarchical physical design of the CorePac in YHFT-X DSP[D]. Changsha: National University of Defense Technology, 2014 (in Chinese)
(魏少雄. YHFT-X 芯片内核的层次化物理设计[D]. 长沙: 国防科技大学, 2014)

- [8] Chen Yuxuan, Liang Liping. An improved hierarchy physical design flow for high speed circuits[J]. Journal of Hunan University (Natural Sciences), 2018, 45((10)): 115–120 (in Chinese)
(陈宇轩, 梁利平. 高速数字模块的层次化物理实现技术[J]. 湖南大学学报: 自然科学版, 2018, 45(10): 115–120)
- [9] Bian Shaoxian, Feng M, Yue D, et al. Innovus machine learning application in performance CPU design[J]. Application of Electronic Technique, 2020, 46(8): 54–59, 63 (in Chinese)
(边少鲜, Feng M, Yue D, 等. Innovus 机器学习在高性能 CPU 设计中的应用[J]. 电子技术应用, 2020, 46(8): 54–59, 63)
- [10] He Xiaowei. Physical design method for high performance CPU core frequency improvement and power optimization[J]. Computer & Digital Engineering, 2020, 48(11): 2677–2679 (in Chinese)
(何小威. 高性能 CPU 核频率提升和功耗优化物理设计方法[J]. 计算机与数字工程, 2020, 48(11): 2677–2679)
- [11] Koppanalil J, Yeung G, O'Driscoll D, et al. A 1.6 GHz dual-core ARM cortex A9 implementation on a low power high-K metal gate 32 nm process[C] //Proc of the IEEE Int Symp on VLSI Design, Automation and Test. Piscataway, NJ: IEEE, 2011: 239–242
- [12] Chen Chen, Xiang Xiaoyan, Liu Chang, et al. Xuantie-910: A commercial multi-core 12-stage pipeline out-of-order 64-bit high performance RISC-V processor with vector extension[C] //Proc of the ACM/IEEE 47th Annual Int Symp on Computer Architecture. New York: ACM, 2020: 52–64



He Xiaowei, born in 1980. PhD, associate research fellow. Member of CCF. His main research interests include physical design of digital and analog integrated circuits, and full custom design of circuits.

何小威, 1980 年生. 博士, 副研究员. CCF 会员. 主要研究方向为数字/模拟集成电路物理设计、全定制电路设计.



Yue Daheng, born in 1980. PhD, associate research fellow. His main research interests include physical design of VLSI and high performance computing CPU design.

乐大珩, 1980 年生. 博士, 副研究员. 主要研究方向为超大规模集成电路物理设计、高性能计算 CPU 设计.



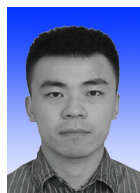
Guo Wei, born in 1986. PhD, assistant research fellow. His main research interests include integrated circuit, microprocessor architecture, and accelerator architecture.

郭 维, 1986 年生. 博士, 助理研究员. 主要研究方向为集成电路、微处理器体系结构、加速器体系结构.



Sui Bingcai, born in 1981. PhD, associate research fellow. His main research interest includes microprocessor architecture.

隋兵才, 1981 年生. 博士, 副研究员. 主要研究方向为微处理器体系结构.



Deng Quan, born in 1989. PhD, assistant research fellow. His main research interests include processing in memory and non-volatile memory.

邓 全, 1989 年生. 博士, 助理研究员. 主要研究方向为存内计算、非易失性存储器.