波动动态差分逻辑RISC-V CPU芯核的功耗抑制技术研究

崔小乐[①②] 李修远[①] 李 浩[①②] 张 兴[*①②]

[①](北京大学深圳研究生院集成微系统重点实验室 深圳 518055)

[②](鹏城实验室 深圳 518055)

摘 要：差分功耗分析(DPA)攻击不仅威胁加密硬件，对加密软件的安全性也构成严重挑战。将波动动态差分逻辑(WDDL)技术应用在RISC-V指令集的处理器芯核上可减少功耗信息的泄露。但是，WDDL技术会给电路引入巨大的功耗开销。该文针对基于WDDL的RISC-V处理器芯核提出两种功耗抑制方法。虽然随机预充电使能技术与指令无关，而预充电使能指令技术需要扩充指令集，但这两种方法都是属于轻量级的设计改进。仿真结果表明，采用了随机预充电使能技术和预充电使能指令技术的Rocket 芯核的电路功耗分别是原始的WDDL Rocekt 芯核功耗的42%和36.4%。

关键词：差分功耗分析；RISC-V 芯核；波动动态差分逻辑；功耗信息泄露；功耗抑制

中图分类号：TN918; TP332.2 **文献标识码**：A **文章编号**：1009-5896(2023)09-3244-09

DOI: 10.11999/JEIT230211

The Power Suppression Techniques for the DPA-resistant RISC-V CPU Core Based on WDDL

CUI Xiaole[①②] LI Xiuyuan[①] LI Hao[①②] ZHANG Xing[①②]

[①](*Key Laboratory of Integrated Micro-systems, Peking University Shenzhen Graduate School, Shenzhen* 518055, *China*)

[②](*Peng Cheng Laboratory, Shenzhen* 518055, *China*)

Abstract: Differential Power Analysis (DPA) is a serious threat to cryptographic hardware and software. The RISC-V processor core based on Wave Dynamic Differential Logic (WDDL) is implemented to mitigate the power leakage. However, the WDDL technique results in a dramatic increase in circuit power. For WDDL-based RISC-V CPU cores, two power suppression techniques are proposed in the paper. Both of them are lightweight solutions. The simulation results show that the circuit power of the DPA-resistant Rocket core with the random precharge enabling technique and the precharge enabling instruction technique can be reduced to 42% and 36.4% of that of the original WDDL based counterpart, respectively.

Key words: Differential Power Analysis (DPA); RISC-V core; Wava Dynamic Differential Logic (WDDL); Power leakage; Power suppression

1 引言

侧信道攻击能够从设备的侧信道信号中提取加密操作使用的密钥，给加密系统带来了严重的威胁。差分功耗分析 (Differential Power Analysis, DPA)是侧信道攻击方法之一。它通过大量收集电路的功耗迹线，利用加密算法的中间值和设备运行过程中泄露的功耗信息之间的关系来获取密钥[1]。DPA是一种非侵入式的攻击方法，因此相对而言容易实施。

密码算法功能可基于硬件形式和软件形式实现。硬件形式即在专用硬件电路上实现密码算法功能，而软件形式则是将密码算法作为软件程序运行在通用处理器上。目前研究表明，许多密码硬件已经被DPA成功攻击。例如，文献[2,3]成功攻击了AES (Advanced Encryption Standard)专用加密电路，Den Boer等人[4]成功攻击了RSA (Rivset-Shamir-Adleman)硬件电路，Fan等人[5]成功破解了ECC (Elliptic Curve Cryptography)电路。另一方面，DPA攻击对运行在通用处理器上的加密程序

收稿日期：2023-04-23；改回日期：2023-08-23；网络出版：2023-08-24

*通信作者： 张兴 zhx@pku.edu.cn

基金项目：深圳学科布局项目(JCYJ20220818100814033)，深圳孔雀团队项目(KQTD20200820113105004)，广东省重点科技研发计划项目(2019B010155002)

Foundation Items: The Subject Layout Program of Shenzhen (JCYJ20220818100814033), The Peacock Plan of Shenzhen (KQTD20200820113105004), The Key-Area Research and Development Program of Guangdong Province (2019B010155002)

同样具有威胁。Mpalane等人[6]对运行AES程序的PIC MCU成功进行了比特级的DPA攻击，Petrvalsky等人[7,8]分别攻击了运行在8 051 MCU和ARM核心上的AES软件。对于基于开源指令集RISC-V的各种处理器，DPA攻击也具有严重的威胁[9]。

因此，针对DPA攻击的防御措施引起了学术界和工业界的极大关注。目前研究者提出的DPA防御措施主要分为两类：掩蔽和隐藏。掩蔽措施对密码算法的中间值进行混淆。例如，Akkar等人[10]将掩码应用于AES和DES专用加密电路上。他使用一个固定的掩码对明文进行XOR操作，从而实现了对这两种加密电路的保护。Lu等人[11]在SM4电路加密的每一轮中注入一个随机状态来混淆中间值，从而达到抵抗DPA的目的。隐藏措施的基本思想与掩蔽措施不同，该类技术是通过改变电路运行期间的功耗或时序等特性，从而减少泄露信息。隐藏措施中较为常见的是构造安全的逻辑单元门，例如，Tiri等人[12]提出了WDDL (Wave Dynamic Differential Logic)，Bucci等人[13]提出了TDPL (Three-phase Dual-rail Precharge Logic)，Bellizia等人[14]进一步提出了SC-DDPL (Standard-Cell Delay based Dual-rail Precharge Logic)来抵抗DPA攻击。

近年来，面向CPU的可抵抗DPA攻击的防御技术也是研究热点。Bayrak等人[15]通过随机化处理器中独立指令的执行顺序和随机化指令的调度来抵抗DPA攻击。Bruguier等人[16]的工作中，将中间值进行掩码操作的同时加入随机延时，通过产生随机噪声来保护通用处理器。针对RISC-V指令集的处理器芯核，研究人员也提出了一些抵抗DPA攻击的措施。例如，De Mulder等人[9]应用了基于掩码阈值实现的掩蔽技术来保护RISC-V CPU芯核。Dao等人[17]提出了一种随机动态频率缩放(Random Dynamic Frequency Scaling, RDFS)技术，在每次加密/解密后随机改变AES协处理器的时钟频率，从而在提高了RISC-V SOC的DPA防御能力的同时保持了较低的功耗。Antognazza等人[18]提出了代码变形方法。该方法依赖于指令语义将密码算法的源码进行动态重编译，从而产生一组相同语义的指令片段。在程序运行期间随机选择指令片段代替原来的指令进行运行。Leplus等人[19]提出了一种随机插入伪指令来增加随机延迟的方法。该伪指令的构造依据于处理器中运行的真实指令，因此攻击者不易区分该指令与真实指令。Stangherlin等人[20]设计了一个位串行的RISC-V微处理器。它在逻辑级别上使用布尔掩码将所有的数据进行保护，并且在晶体管级上使用动态多米诺逻辑有效地防御DPA攻击。

掩蔽措施虽然在一定程度上可以提高硬件的抗DPA攻击能力，但是由于大部分掩码方法需要考虑密码算法的具体实现方式，因此许多掩码方案不具有普适性。当多个算法集成在通用处理器中时，每个掩蔽措施的代价将会累加在一起，造成处理器性能的大幅下降。而隐藏措施中的抗DPA逻辑单元改变了电路运行时的功耗特性，因此从根本上阻断了功耗信息的泄露窗口，并且与算法的实现形式无关，所以相比于掩蔽措施来说，隐藏措施更加通用。文献[21]中比较了不同的抗DPA逻辑单元。有些逻辑单元虽然抗DPA的能力更好，但是需要在晶体管级进行全定制的设计，不适用于处理器之类的大规模集成电路设计。同时逻辑单元在保证一定安全性的前提下，面积和功耗的开销要尽可能的小。基于以上的考虑，本文选择波动动态差分逻辑(Wave Dynamic Differential Logic, WDDL)应用在处理器的芯核上。

WDDL由Tiri在2004年提出，是经典的双轨预充电(Dual-Rail Precharge, DRP)逻辑[12]。它能够在不同的电路输入下使加密电路的功耗迹线平坦化，从而防止信息的泄露。WDDL是支持标准单元库的DRP逻辑，可以使用EDA工具进行自动化设计，更适合应用于大规模集成电路设计中。但是WDDL的预充电行为会带来更多的功耗开销。为了能使基于WDDL的防御措施更具有实用性，本文针对基于WDDL的抗DPA攻击的RISC-V处理器芯核提出两种功耗抑制技术。

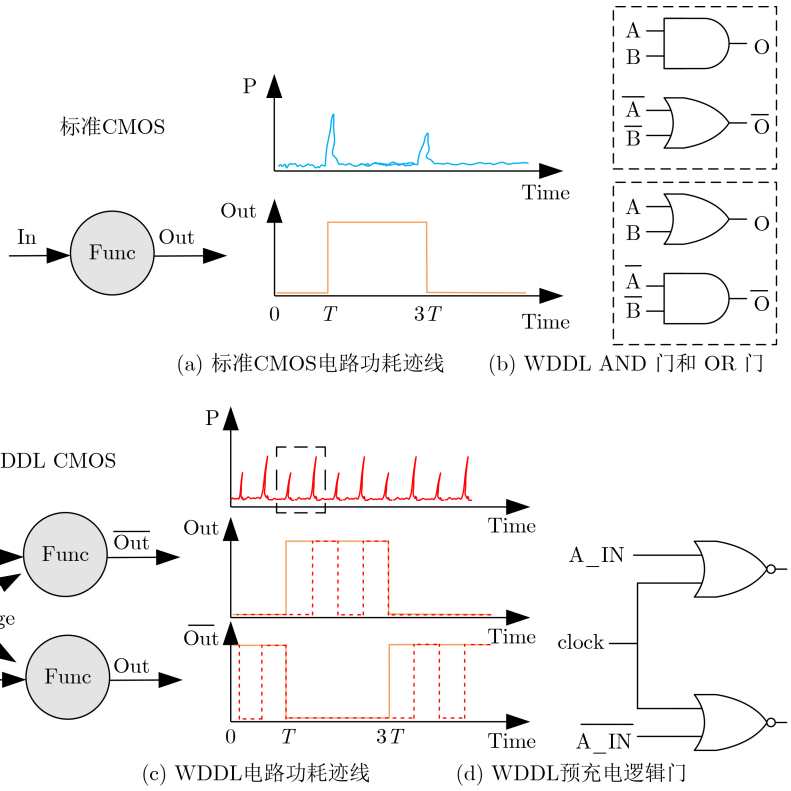
本文的剩余部分安排如下：第2节回顾了WDDL逻辑电路的基本特征，并介绍了基于WDDL单元的RISC-V处理器芯核。第3节提出了两种轻量级的电路功耗抑制的方法：随机预充电使能技术和预充电使能指令技术。第4节讨论仿真结果，第5节得出论文结论。

2　基于WDDL-ALU的Rocket处理器芯核

图1(a)显示了一个标准CMOS电路的输出信号和其功耗迹线之间的关系。当输出端信号的翻转情况不同时，该电路的功耗信息也不同，因此可以根据功耗的变化来分辨输出信号的翻转情况。而WDDL逻辑的基本思想是通过预充电-求值的动态操作，使电路在每个周期内的功耗均匀化。图1(b)为WDDL逻辑的与/或门结构示意图。一个WDDL单元由原始逻辑门和其互补逻辑门组成。WDDL中使用的预充电逻辑门如图1(d)所示。使用时钟信号clock作为预充电控制信号，可以保证在每次求值操作前都进行预充电。在图1(c)中，每个周期内电路的功耗分为预充电功耗和求值功耗。WDDL单元保证了

在不同的信号翻转情况下电路每周期的总功耗保持相同。WDDL单元被证明能够有效地防御DPA攻击。

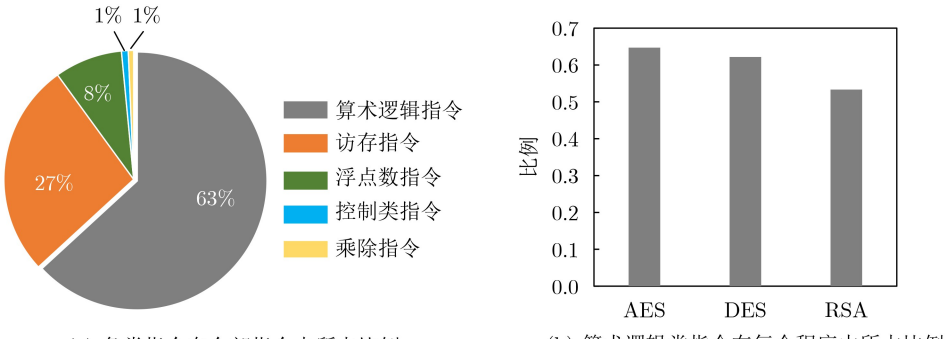
Rocket Core是加州大学伯克利分校开发的一款开源RISC-V处理器芯核，本文配置了一款支持RV64G指令集的64位Rocket芯核作为实验对象。该芯核由5个模块组成：ALU, BreakpointUnit, CSRFile, Ibuf和MulDiv。ALU模块实现算术运算和逻辑运算功能；BreakpointUnit模块实现硬件断点功能；CSRFile模块实现控制和状态寄存器组；Ibuf模块实现读取/缓存执行指令；MulDIV模块实现乘法除法功能。对于AES, DES, RSA等加密程序，图2统计了程序主函数中各类指令的数目。从图2(a)中得出，在3个程序的全部指令中，逻辑算术类指令所占比例最大，超过了60%；而从图2(b)中可知，对于每个程序，逻辑算术类指令所占比例均超过50%。图2表明这些加密程序所使用的主要指令属于逻辑算术类，这些指令都是由Rocket芯核的ALU模块处理的。WDDL电路相比于标准CMOS电路具有更大的功耗和面积开销，把整个Rocket芯核都替换为WDDL单元的成本非常高，因此本文只将ALU模块的CMOS单元替换为WDDL单元。本文将使用了WDDL-ALU的Rocket芯核称为“sw-Rocket”，把所有模块都用标准CMOS单元综合的Rocket芯核称为“sc-Rocket”。图3显示，WDDL-ALU模块的面积比标准CMOS-ALU模块增加了2.7倍；而与sc-Rocket芯核相比，sw-Rocket芯核的整体面积增加了15%。



(a) 标准CMOS电路功耗迹线 (b) WDDL AND 门和 OR 门

(c) WDDL电路功耗迹线 (d) WDDL预充电逻辑门

图 1 标准CMOS电路与WDDL电路单元对比



(a) 各类指令在全部指令中所占比例 (b) 算术逻辑类指令在每个程序中所占比例

图 2 密码程序的汇编指令比例分析

本文使用这两种芯核进行了Helloworld, AES, DES和RSA等程序的仿真，并进行了功耗分析，结果如表1所示。在运行AES加密程序时， WDDL-ALU模块的功耗是CMOS-ALU模块的37.4倍，并且sw-Rocket芯核的整体功耗是sc-Rocket芯核的3.1倍。这是因为预充电-求值机制使得所有WDDL单元在每个时钟周期内都产生两次信号翻转，然而CMOS单元在每个周期内最多产生1次翻转。此外，sc-Rocket内部的CMOS-ALU模块只有当被程序调用时才会产生动态功耗，然而sw-Rocket内部的WDDL-ALU模块则在从上电到断电的完整运行时间内均在产生动态功耗。因此，需要功耗抑制技术来使sw-Rocket芯核更具有实用性。

3　功耗抑制技术

WDDL的预充电行为带来了大量的功耗开销，本节提出了两种技术对其进行控制从而降低功耗。

3.1　随机预充电使能技术

基于预充电行为的随机化思想，可通过一个随机预充电使能发生模块来控制预充电的行为。随机预充电使能发生器模块的结构如图4所示。它由分频器Divider、伪随机数发生器(Pseudo Random Number Generator, PRNG)和预充电信号产生器Counter等3个子模块组成。分频器的两个输入信号分别为Rocket芯核的全局时钟信号和全局复位信号。该分频器可根据配置参数N_C的取值，产生1个周期是clock的N_C倍的新的时钟信号clock_div，以驱动下一级子模块PRNG。如图5所示，PRNG子模块主要由LFSR(Linear Feedback Shift Register)和值偏置电路组成。该LFSR每周期可以生成1个16 bit的伪随机数R_d。值偏置电路通过取模和加法操作，可对原始随机数R_d进行计算处理，从而得到最终输出R_p。该随机数会发送到PRNG子模块的输出端口rand_val。R_d和R_p之间满足关系式(1)

$$R_p = (R_d \bmod (P_C + 1)) + (N_C - (P_C + 1)) \quad (1)$$

其中N_C和P_C为配置参数，N_C和P_C共同决定了随机数的范围为$[N_C - P_C - 1, N_C - 1]$。

预充电信号发生器是一个由全局时钟驱动的循环计数器，计数范围为$[0, N_C - 1]$。在每个周期内，计数值都会与输入信号rand_val进行比较。如果计数值大于或等于rand_val，则将输出信号pre_en置为1，否则置为0。

引入随机化前后的预充电逻辑如图6所示。图6(a)是加入随机化前的原始预充电逻辑，图6(b)则是加入了随机化的预充电逻辑，随机预充电使能发生模块的输出和时钟信号、复位信号共同控制预充电的行为，从而驱动sw-Rocket芯核内部的WDDL单元工作。随机化的预充电使能信号可减少WDDL-ALU模块的预充电行为数量，从而降低了它的功耗。

本文提出的随机预充电使能发生器的两个参数(N_C,P_C)均是可以配置的。本文对不同配置情况下的随机预充电使能发生器的面积开销和功耗开销进

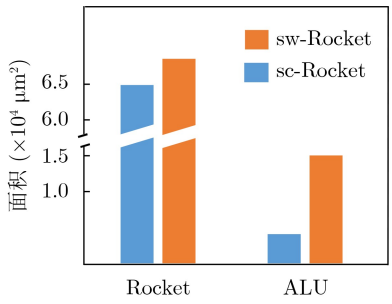


图 3 sw-Rocket芯核的面积开销

表 1　不同程序的功耗比较(mW)

	sc-Rocket		sw-Rocket	
	Rocket	ALU	Rocket	ALU
Helloworld	1.55	0.04	5.60(×3.6)	4.09(×102.3)
DES	2.16	0.13	6.14(×2.8)	4.12(×31.6)
AES	1.94	0.11	5.93(×3.1)	4.11(×37.4)
RSA	1.72	0.05	5.77(×3.4)	4.09(×81.8)

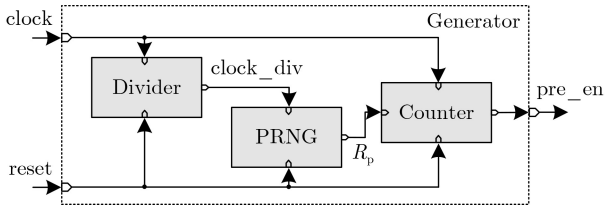


图 4 随机预充电使能发生器电路结构示意图

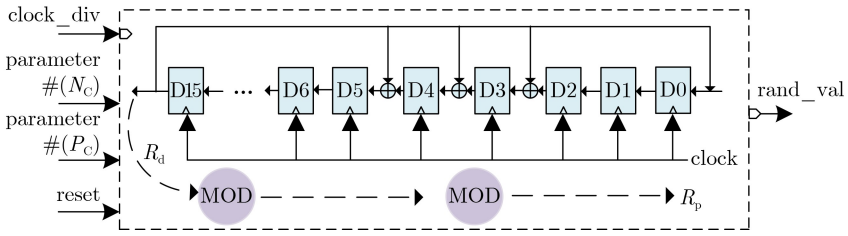
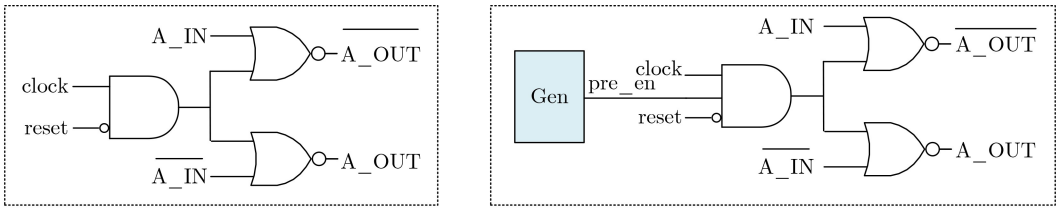


图 5 PRNG电路示意图



(a) 标准WDDL预充电逻辑单元 (b) 随机预充电WDDL预充电逻辑单元

图 6 预充电逻辑单元电路结构

行了分析，结果如表2所示。表2的第3～5列分别列出了分频器Divider，随机数发生器PRNG和随机预充电使能发生器的顶层模块Generator的面积开销。从表中可以发现，3个模块的面积都随着N_C的增加而增加，而只有PRNG和counter的面积随着P_C的增加而增加。在表2中最大面积开销由最大配置参数对(N_C,P_C)=(64, 64)带来。在此配置下，顶层模块随机预充电使能发生器的面积开销为709.6 μm^2，该面积仅为WDDL-ALU面积的4.6%。表2的第6列是随机预充电使能发生器在不同配置下的功耗开销。从表中可以发现，电路的功耗随着N_C,P_C的增加轻微地上升。随机预充电使能发生器所带来的功耗仅在μW量级，而WDDL-ALU的功耗为mW量级。这意味着本文所提出的预充电使能技术是一个轻量级的解决方法，并且这种技术在不同架构的CPU中均有应用潜力。

表 2 随机预充电发生器的面积和功耗开销

N_C	P_C	面积(μm^2)			功耗(μW)
		Divider	PRNG	Generator	Generator
8	2	33.1	270.4	346.7	26.6
	6	33.1	354.6	436.7	27.7
16	5	46.4	271.0	372.2	31.3
	8	46.4	400.0	517.3	31.4
	16	46.4	451.8	579.2	31.7
32	5	58.7	272.0	398.2	35.8
	10	58.7	453.2	597.6	35.6
	16	58.7	453.6	604.8	35.7
	30	58.7	465.5	615.9	36.3
64	6	71.6	356.8	519.8	40.6
	8	71.6	402.5	571.0	40.6
	16	71.6	457.2	632.9	40.8
	30	71.6	466.9	642.6	40.9
	64	71.6	520.6	709.6	41.0

3.2 预充电使能指令技术

RISC-V指令集允许使用者进行自定义指令的扩展，它启发我们设计新的指令来控制WDDL-ALU的预充电行为。RISC-V指令集支持的指令扩展方式主要有两种：第1种是使用操作码opcode在自定义空间(custom空间)进行指令拓展。例如Rocket芯核会将custom指令通过Rocket自定义协处理器接口发送到协处理器中进行处理，协处理器处理完成后会将结果返回Rocket核心内部。然而，这个方法不适用于预充电控制，因为WDDL-ALU模块位于Rocket芯核内部，而不是作为协处理器挂载在其外部。第2种指令的拓展方法是使用操作码opcode的保留空间(reserved空间)来定义全新的指令。此拓展方法需要修改硬件的每一级流水线来实现该新指令所定义的功能，同时还需要修改编译器来支持该新指令，实现十分不便。

本文提出一种自定义指令扩展方法，该方法利用了RISC-V基本指令集中的CSR类指令。这一类指令提供了一个软件和控制状态寄存器(Control and Status Register, CSR)之间交互的接口。通过CSR指令，软件可以控制和观察CSR寄存器的状态。RISC-V指令集架构规定地址空间0x000-0xFFF为控制状态寄存器的地址空间，同时支持设计者在允许的地址空间内扩展新的CSR寄存器。本文首先在地址0x800处扩展了一个预充电使能CSR寄存器Reg_enPre，该地址被允许进行自定义读写操作。扩展CSR寄存器后，硬件的预充电行为可由Reg_enPre寄存器直接控制。此时预充电行为如图7所示。使能信号enpre，即CSR寄存器Reg_enPre的第0 bit，控制WDDL-ALU预充电的开启和关断：当调用指令“csrwi 0x800, 0x1”将1写入Reg_enPre时，会使ALU开启预充电；当调用指令“csrwi 0x800, 0x0”将0写入Reg_enPre时，会使ALU关闭预充电。该寄存器Reg_enPre的位宽与Rokcet处理器芯核的通用寄存器位宽相同，都是64 bit，因此若有需求，可以使寄存器的每一位来控制不同模块预充电的开启和关闭。

通过扩展CSR寄存器的实例和调用csrwi指令，对扩展后CSR寄存器状态的控制相当于扩展了自定义的预充电使能指令，从而可达到由指令直接控制硬件预充电行为的目的。Rocket处理器芯核的流水线不需要修改。由于CSR指令已经是RISC-V指令

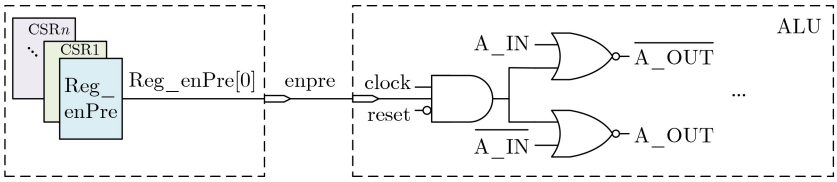


图 7 预充电使能指令通过CSR Reg_enPre控制预充电

集的标准指令，编译器也不需要修改。以“csrwi addr, value”形式实现的预充电使能指令可以在汇编程序的任意位置调用，因此该指令理论上可以控制密码算法程序的任意代码段。在UMC 55 nm的工艺库下，新扩展的CSR寄存器的面积仅有631.8 μm^2，面积开销几乎可以忽略。只有当CSR中的数据发生翻转时，该新扩展的CSR寄存器才会产生动态功耗，因此功耗开销也可以忽略不计。

4　仿真和讨论

4.1　仿真工具与流程

本文使用Design Compiler工具在UMC 55 nm工艺库下对Rocket芯核的源码进行综合。得到网表后，使用VCS将Rocket门级网表和相关接口文件编译，得到基于VCS的Rocket硬件仿真器。然后使用该硬件仿真器运行AES加密程序的可执行文件，得到波形文件。最后使用PrimeTimePX在time_based模式下对波形文件进行功耗分析，得到功耗迹线。具体流程如图8所示。

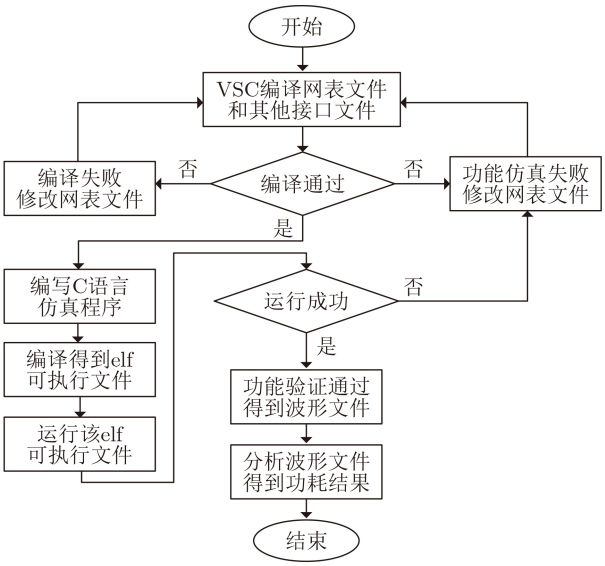


图 8 功能仿真及功耗分析流程

4.2　实验结果

本文将第3节提出的两种功耗抑制技术应用在不同的Rocket处理器芯核上进行仿真并分析结果。应用了随机预充电使能技术的Rocket芯核称为rw-Rocket (random based WDDL-ALU Rocket)，而应用了预充电使能指令技术的Rocket芯核称为iw-Rocket (instruction based WDDL-ALU Rocket)。对于rw-Rocket芯核，根据表2中各参数配置下随机预充电使能发生器的面积和功耗开销，选择了参数对(N_C,P_C)=(32,5)进行配置以获得平均水平。表3比较了不同Rocket芯核的面积开销。与sw-Rocket芯核相比，rw-Rocket和iw-Rocket的面积分别只增加了0.5%和0.7%左右。

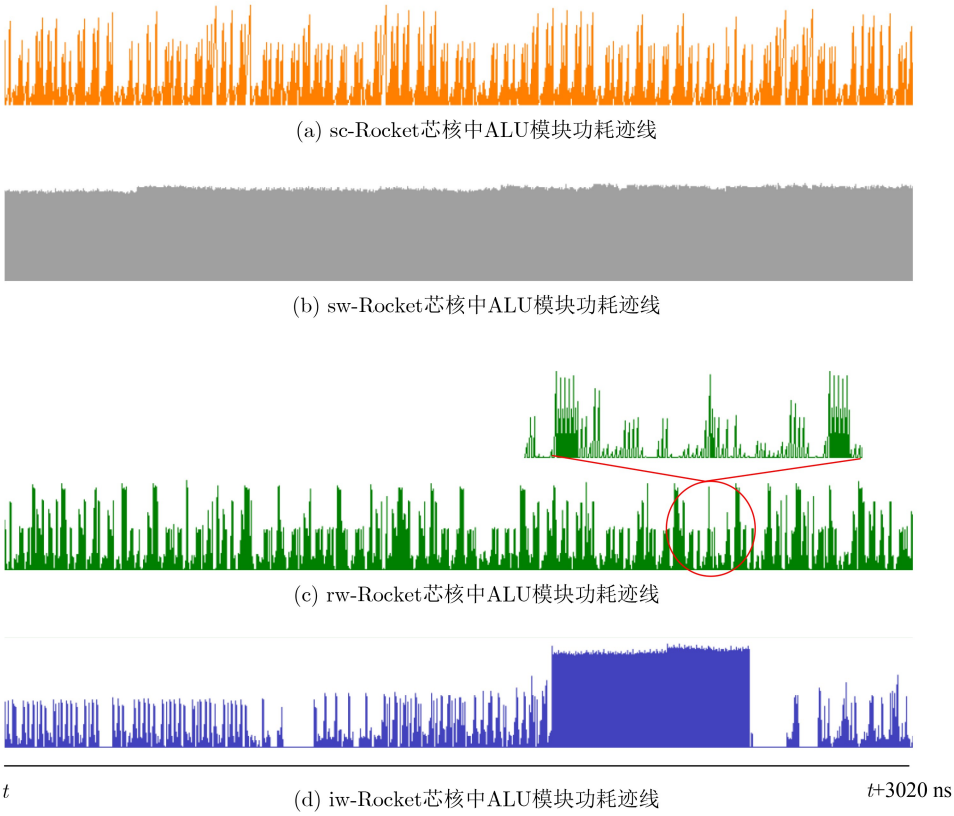
表 3　不同Rocket核心的面积开销(μm^2)

	sc-Rocket	sw-Rocket	rw-Rocket	iw-Rocket
Rocket	74644.2	85620.6	86020.9	86252.8
ALU	4063.0	15039.4	15439.7	15010.9
CSRFile	15794.3	15794.3	15794.3	16426.1

用C语言编写的AES-128加密程序分别运行在不同的Rocket芯核上。AES-128的密钥长16 Byte，它每次只加密16 Byte长度的明文。对于在iw-Rocket芯核上运行的AES-128程序，需要在程序源码中插入预充电使能指令。由于riscv-gcc内联汇编的支持，可以直接在C语言代码中插入“asm volatile (“csrwi 0x800, 0x1”)”和“asm volatile (“csrwi 0x800, 0x0”)”来控制预充电的开启和关闭。文献[8]对ARM核心的DPA攻击是在AES-128算法的字节替换过程中进行的。因此本文以最易受到攻击的字节替换代码段为示例，在该程序段开始时开启硬件预充电，当字节替换结束时关闭硬件预充电。

图9显示了不同的Rocket芯核运行AES-128程序时的ALU模块功耗迹线。图9(a)是sc-Rocket芯核中的ALU模块功耗迹线。图9(b)是sw-Rocket芯核中ALU模块的功耗迹线。它表明在运行AES-128程序时，WDDL-ALU模块的功耗在每个周期内几乎相同。

图9(c)是rw-Rocket芯核的内部ALU模块功耗迹线。随机预充电使能技术已经把功耗迹线随机化，并且破坏了AES算法的中间值和实时功耗之间的相关性。由于时间分辨率的原因，在图9(c)的功耗迹线中所表现出的预充电行为并不明显。但是放大其中的一段后可以发现，ALU模块中的预充电行为已经由使能信号进行了随机化控制。在使能信号



(a) sc-Rocket芯核中ALU模块功耗迹线

(b) sw-Rocket芯核中ALU模块功耗迹线

(c) rw-Rocket芯核中ALU模块功耗迹线

(d) iw-Rocket芯核中ALU模块功耗迹线

图 9 AES-128程序运行在不同的Rocket芯核上ALU模块部分功耗迹线

有效期间，ALU模块在进行预充电操作，硬件得到了功耗平均化方式的保护。而实际上，rw-Rocket芯核的抗DPA能力归因于预充电的随机性。根据式(1)，参数N_C越大随机性越好，因为N_C决定了产生的随机数R_p的上限；而参数P_C越接近N_C时随机性越好。因为在N_C一定时，P_C决定了随机数R_p的下限。也就是说，当N_C越大，且P_C越接近N_C时，随机数R_p的范围越宽，从而可使预充电使能信号的随机性越好。

图9(d)是iw-Rocket芯核中ALU模块的功耗迹线。预充电使能指令应用于字节替换程序段。从图9(d)可以看出预充电的行为准确地发生在了字节替换的操作过程中，并且ALU模块的功耗在AES-128程序的字节替换过程中始终保持相同。

表4列出了AES-128程序运行在不同Rocket芯核上的功耗。rw-Rocket芯核中ALU模块的功耗约为sw-Rocket芯核中ALU功耗的16%，而整个rw-Rocket芯核的功耗是sw-Rocket功耗的42%。

表 4 不同Rocket核心运行AES-128程序的功耗开销(mW)

	sw-Rocket	rw-Rocket	iw-Rocket
Rocket	5.93	2.49	2.16
ALU	4.11	0.67	0.33
CSRFile	0.24	0.24	0.26

iw-Rocket核心中ALU模块的功耗只有sw-Rocket芯核中ALU功耗的8%，整个iw-Rocket芯核的功耗是sw-Rocket功耗的36.4%。这表明提出的两种技术都能够抑制使用了WDDL-ALU的Rocket芯核的功耗。

在rw-Rocket芯核和iw-Rocket芯核上运行AES-128程序的功耗分别为2.49 mW和2.16 mW，相比于sc-Rocket的功耗分别上升了28.3%和11.3%。但是，在可接受的电路功耗开销下，RISC-V处理器芯核防御DPA攻击的能力得到了明显增强。

4.3 优势与不足

本文提出随机预充电使能技术和预充电使能指令技术具有两大明显的优势。其一是对任意加密算法都具有通用性。应用了随机预充电使能技术的rw-Rocket芯核和应用了预充电使能指令技术的iw-Rocket芯核理论上可以对所有加密算法提供抗DPA攻击的保护。当使用rw-Rocket芯核时，密码算法的源码完全不需要修改；当使用iw-Rocket芯核时，可以在源码中插入自定义的预充电使能指令来精确地保护关键的程序段。相比于文献[17]中使用的仅对AES算法提供保护的协处理器方案，本文提出的两种方法对密码算法具有通用性。本文方法的另一大优势在于较高的安全性。文献[16]中使用插入随机延时的方法。但是由于弹性对齐算法和机器学习的引入，DPA攻击也可以对非对齐的功耗

迹线产生威胁。因此该方案存在被攻破的风险。而文献[9]中曾使用了掩码的方案来防御DPA攻击，但是该方案只能防御一阶的DPA攻击。高阶的DPA攻击可以对多个采样点进行联合分析，从而破解掩码的作用。本文提出的两种方案都是基于安全的逻辑单元WDDL，它从根本上改变了电路运行时的功耗特征，从而阻断了信息泄露的途径。因此本方案能够有效地防御使用了弹性对齐算法的DPA攻击以及高阶的DPA攻击，具有较高的安全性。

同时本文也存在一些不足之处。用WDDL逻辑单元将ALU模块的CMOS单元替换后会增加关键路径的延时，从而降低处理器的性能。但是从仿真结果来看，在可接受的性能开销范围内，处理器防御DPA攻击的能力得到了明显增强。目前我们针对两种方案的安全性验证都是基于实验原理以及仿真结果的，未能在流片后的芯片上进行DPA攻击。后续我们将在条件允许的情况下进一步完善这方面的工作。

5　结束语

本文针对基于WDDL-ALU的抗DPA攻击的RISC-V处理器芯核提出了两种功耗抑制方法。随机预充电使能技术与CPU架构无关，它通过随机化预充电的行为改善了RISC-V芯核的抗DPA攻击能力。预充电信号的随机性由随机预充电使能产生器的参数对(N_C, P_C)共同控制。预充电使能指令技术使用了一种新的指令拓展方法，它不需要对处理器的硬件以及编译器进行修改。这个方法也是通过控制WDDL预充电的行为来保护RISC-V芯核。该方法的灵活性在于它是通过重用CSR指令来实现的，而且新的预充电控制指令可以在程序的任何地方调用。仿真结果表明，本文提出的两种技术都能够保护RISC-V芯核免受DPA攻击。在本文的AES-128程序仿真中，随机预充电使能技术和预充电使能指令技术分别把rw-Rocket和iw-Rocket中ALU模块的功耗降低到了sw-Rocket芯核中ALU模块功耗的16%和8%，rw-Rocket芯核和iw-Rocket芯核的整体功耗分别降低至sw-Rocket芯核功耗的42%和36.4%。这两种技术都是轻量级的解决方案，因为功耗和面积开销相比基于WDDL-ALU的Rocket芯核只略有增加。这两种技术使WDDL成为RISC-V芯核抵抗DPA攻击的实用方法，因为与原始的RISC-V芯核相比，其功耗只分别增加了28.3%和11.3%。

参 考 文 献

[1] KOCHER P, JAFFE J, and JUN B. Differential power analysis[C]. The 19th Annual International Cryptology Conference, Santa Barbara, USA, 1999: 388–397. doi: 10.1007/3-540-48405-1_25.

[2] ORS S B, GURKAYNAK F, OSWALD E, *et al*. Power-analysis attack on an ASIC AES implementation[C]. The International Conference on Information Technology: Coding and Computing, Las Vegas, USA, 2004: 546–552. doi: 10.1109/itcc.2004.1286711.

[3] CHEN Juncheng, NG J S, KYAW N A, *et al*. Normalized differential power analysis - for ghost peaks mitigation[C]. 2021 IEEE International Symposium on Circuits and Systems, Daegu, Korea, 2021: 1–5. doi: 10.1109/ISCAS51556.2021.9401487.

[4] DEN BOER B, LEMKE K, and WICKE G. A DPA attack against the modular reduction within a CRT implementation of RSA[C]. The 4th International Workshop Redwood Shores, Redwood Shores, USA, 2003: 228–243. doi: 10.1007/3-540-36400-5_18.

[5] FAN Junfeng and VERBAUWHEDE I. An updated survey on secure ECC implementations: Attacks, countermeasures and cost[M]. NACCACHE D. Cryptography and Security: From Theory to Applications. Berlin, Heidelberg: Springer, 2012: 265–282. doi: 10.1007/978-3-642-28368-0_18.

[6] MPALANE K, TSAGUE H D, GASELA N, *et al*. Bit-level differential power analysis attack on implementations of advanced encryption standard software running inside a PIC18F2420 microcontroller[C]. 2015 International Conference on Computational Science and Computational Intelligence, Las Vegas, USA, 2015: 42–46. doi: 10.1109/csci.2015.115.

[7] PETRVALSKY M, DRUTAROVSKY M, and VARCHOLA M. Differential power analysis of advanced encryption standard on accelerated 8051 processor[C]. 2013 23rd International Conference Radioelektronika, Pardubice, Czech Republic, 2013: 334–339. doi: 10.1109/radioelek.2013.6530942.

[8] PETRVALSKY M, DRUTAROVSKY M, and VARCHOLA M. Differential power analysis attack on ARM based AES implementation without explicit synchronization[C]. 2014 24th International Conference Radioelektronika, Bratislava, Slovakia, 2014: 1–4. doi: 10.1109/radioelek.2014.6828434.

[9] DE MULDER E, GUMMALLA S, and HUTTER M. Protecting RISC-V against side-channel attacks[C]. The 56th Annual Design Automation Conference, Las Vegas, USA, 2019: 45. doi: 10.1145/3316781.3323485.

[10] AKKAR M L and GIRAUD C. An implementation of DES and AES, secure against some attacks[C]. The 3rd International Workshop on Cryptographic Hardware and Embedded Systems, Paris, France, 2001: 309–318. doi: 10.

1007/3-540-44709-1_26.

[11] LU Tong, ZHOU Fang, WU Ning, *et al*. Implementation of SM4 based on random state to resist DPA[C]. 2021 IEEE 4th International Conference on Electronics Technology, Chengdu, China, 2021: 717–721. doi: 10.1109/ICET51757.2021.9450907.

[12] TIRI K and VERBAUWHEDE I. A logic level design methodology for a secure DPA resistant ASIC or FPGA implementation[C]. The Design, Automation and Test in Europe Conference and Exhibition, Paris, France, 2004: 246–251. doi: 10.1109/date.2004.1268856.

[13] BUCCI M, GIANCANE L, LUZZI R, *et al*. Three-phase dual-rail pre-charge logic[C]. The 8th International Workshop on Cryptographic Hardware and Embedded Systems, Yokohama, Japan, 2006: 232–241. doi: 10.1007/11894063_19.

[14] BELLIZIA D, BONGIOVANNI S, OLIVIERI M, *et al*. SC-DDPL: A novel standard-cell based approach for counteracting power analysis attacks in the presence of unbalanced routing[J]. *IEEE Transactions on Circuits and Systems I:Regular Papers*, 2020, 67(7): 2317–2330. doi: 10.1109/tcsi.2020.2979831.

[15] BAYRAK A G, VELICKOVIC N, IENNE P, *et al*. An architecture-independent instruction shuffler to protect against side-channel attacks[J]. *ACM Transactions on Architecture and Code Optimization*, 2012, 8(4): 20. doi: 10.1145/2086696.2086699.

[16] BRUGUIER F, BENOIT P, TORRES L, *et al*. Cost-effective design strategies for securing embedded processors[J]. *IEEE Transactions on Emerging Topics in Computing*, 2016, 4(1): 60–72. doi: 10.1109/tetc.2015.2407832.

[17] DAO B A, HOANG T T, LE A T, *et al*. Correlation power analysis attack resisted cryptographic RISC-V SoC with random dynamic frequency scaling countermeasure[J]. *IEEE Access*, 2021, 9: 151993–152014. doi: 10.1109/ACCESS.2021.3126703.

[18] ANTOGNAZZA F, BARENGHI A, and PELOSI G. Metis: An integrated morphing engine CPU to protect against side channel attacks[J]. *IEEE Access*, 2021, 9: 69210–69225. doi: 10.1109/access.2021.3077977.

[19] LEPLUS G, SAVRY O, and BOSSUET L. Insertion of random delay with context-aware dummy instructions generator in a RISC-V processor[C]. 2022 IEEE International Symposium on Hardware Oriented Security and Trust, McLean, USA, 2022: 81–84. doi: 10.1109/HOST54066.2022.9840060.

[20] STANGHERLIN K and SACHDEV M. Design and implementation of a secure RISC-V microprocessor[J]. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 2022, 30(11): 1705–1715. doi: 10.1109/TVLSI.2022.3203307.

[21] TENA-SÁNCHEZ E, POTESTAD-ORDÓÑEZ F E, JIMÉNEZ-FERNÁNDEZ C J, *et al*. Gate-level hardware countermeasure comparison against power analysis attacks[J]. *Applied Sciences*, 2022, 12(5): 2390. doi: 10.3390/app12052390.

崔小乐：男，教授，研究方向为集成电路的可测性、可靠性和安全性.

李修远：男，硕士生，研究方向为高安全RISC-V处理器设计.

李 浩：男，硕士生，研究方向为低功耗RISC-V处理器设计.

张 兴：男，教授，研究方向为集成电路的新器件、新结构、新工艺.

责任编辑：马秀强