

Raspberry Pi 的 DRAM PUF 密钥生成系统研究

谷敏章,毛臻,葛成华

(中国电子科技集团公司 第五十八研究所,无锡 214072)

摘要: 介绍了基于树莓派 B+ 硬件平台实现 DRAM PUF 密钥发生系统的研究与设计。系统从嵌入式设备广泛使用的动态随机存储器中提取带有自身硬件指纹的物理不可克隆函数,通过模糊提取算法和 Hash 算法消除由于环境噪声及老化效应带来的错误信息,稳定生成 128 位 DRAM PUF 安全密钥。该系统生成的 DRAM PUF 安全密钥可靠性高达 99%,唯一性为 48.44%,均匀性为 50%,并且密钥的信息熵值为 1.0。

关键词: 树莓派 B+;密钥生成器;模糊提取;动态随机存储器;信息安全

中图分类号: TN918.2

文献标识码: A

Research on DRAM PUF Key Generation System Based on Raspberry Pi

Gu Minzhang, Mao Zhen, Ge Chenghua

(China Electronics Technology Group Corporation No. 58 Research Institute, Wuxi 214072, China)

Abstract: The research and design of a DRAM PUF key generation system based on the Raspberry Pi B+ hardware platform are introduced. The system extracts the physical unclonable function with its own hardware fingerprint from the dynamic random memory widely used in embedded devices, and achieves stable generation of 128-bit DRAM PUF security keys through fuzzy extraction algorithm and Hash algorithm to eliminate the error information due to environmental noise and aging effect. The system generates DRAM PUF security keys with 99% reliability, 48.44% uniqueness, 50% uniformity, and the information entropy value of the key is 1.0.

Keywords: Raspberry Pi B+; key generator; fuzzy extractor; DRAM; information security

0 引言

近年来,随着芯片制程工艺的不断突破和计算能力的持续加强,各类智能物联网设备已广泛应用于日常生活的很多方面^[1]。万物互联时代已然到来,涉及个人隐私、商业机密甚至国家安全的关键数据信息广泛地被计算、传输和存储于无处不在的嵌入式设备中。正因如此,嵌入式设备的数据安全问题引起人们的关注,逐渐成为信息安全领域的研究热点。现有的嵌入式信息安全防御机制通常依赖于安全密钥的加密、解密系统,然而存储在 Flash、EPROM 等非易失性存储器中的密钥等关键安全数据容易受到恶意攻击^[2]。一旦密钥等机密信息发生泄漏,设备中的其余数据信息都将面临严重的安全隐患,甚至导致整个安全防御机制的崩溃。

借鉴基于人体自身物理特征(如指纹和虹膜)进行个人身份认证的思想,研究人员将物理实体中细微的内在差异作为独一无二的标志进行单个物体识别,提出物理不可克隆函数(PUF)的概念^[3-4]。当前,多数 PUF 的研究与实现需要添加额外的硬件电路,使得生成 PUF 的成本增加,

并且不方便在嵌入式设备中直接提取。凭借存储密度高、制造成本低等特性,动态随机存储器(DRAM)作为内存单元在商用嵌入式设备中广泛应用^[5],为了保持在内存单元中存储信息不丢失,嵌入式设备的 DRAM 内存单元需要周期性刷新,基于不同 DRAM 单元在停止自动刷新期间电荷的衰减特性存在差异的原理,研究人员提出了一种基于电荷衰减原理的 DRAM PUF^[6-7]。

1 DRAM PUF 概念及原理分析

1.1 PUF 概念及定义

物理不可克隆函数可被视为芯片的硬件指纹,反映了其内部独特的特性差异,具有不可复制的激励-响应的映射关系^[8]。其源于在芯片制造过程中的纳米级工艺偏差,这些差异可能由纳米级线宽差异、氧化层厚度、随机掺杂波动等多种因素共同造成。

对 PUF 电路输入某个特定信号,作为激励会产生一个特定且不可复制的输出响应,特定的输入激励与对应生成的输出响应之间构成激励-响应对,激励-响应对反映了芯片电路内在的独特差异。如图 1 所示,即使在同一批次

生产的同一块晶圆上,基于相同的掩模板所生产制造的芯片,对其输入相同的激励也会产生完全不同的响应。

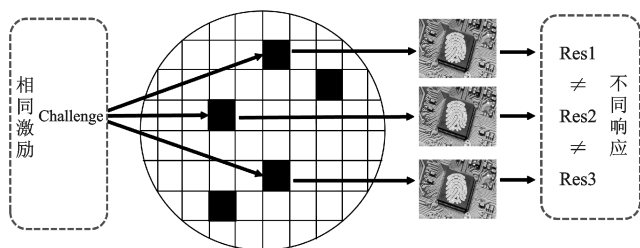


图1 PUF的实现原理

1.2 DRAM PUF 的原理分析

在 DRAM 存储单元中,晶体管负责读取存储于电容中的电荷数量,并与规定阈值比较,判断存储在 DRAM 单元中的数据表现为逻辑“0”还是逻辑“1”。由于晶体管漏电流现象的存在,需要对 DRAM 单元进行周期性刷新,以维持电容中存储的电荷^[9]。否则,由于图 2 所示的多条漏电路径的存在,电容中的电荷将随时间逐渐发生泄漏,从而导致数据发生位翻转,最终导致数据丢失,此物理现象称为 DRAM 单元的电荷衰减^[7]。在商用 DRAM 芯片中,由 DRAM 内存控制器负责周期性的刷新,以此来保证存储在 DRAM 中的数据不会丢失,进而保证设备系统的正常运行(DRAM 的刷新周期由制造厂商定义,通常为 32 ms 或 64 ms)。

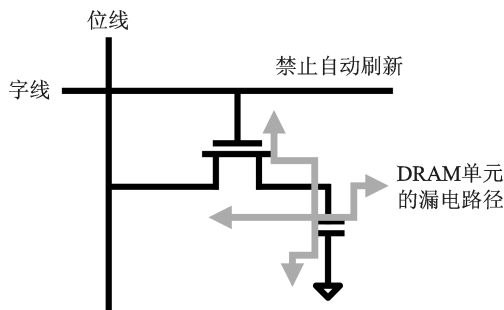


图2 DRAM 漏电路径示意图

由于 DRAM 芯片制造过程中的随机性差异,每个 DRAM 单元中电荷漏电的速度并不相同。在相同的衰减时间下,不同 DRAM 单元是否产生数据丢失的情况也有所不同。电荷衰减速度这一物理特性为单个 DRAM 单元特有,既无法复制,也无法预测,并且单个 DRAM 单元在整个 DRAM 芯片中拥有独一无二的位置信息,因此能够作为芯片唯一的身份信息用于安全认证。

2 方案设计

2.1 硬件实现

本文选用 3 块 Raspberry Pi B+ 开发板(以下简称为

Pi1、Pi2、Pi3)进行 DRAM PUF 响应的按行读取,并完成基于模糊提取算法的 DRAM PUF 密钥发生器的方案实现。如图 3 所示,Raspberry Pi B+ 开发板上搭载了一颗多核 Broadcom BCM2835 SoC 芯片,其中 CPU 为 700 MHz ARM11 76JZF-S 处理器,GPU 为 Broadcom Video Core IV 图像处理器。同时,以 PoP 封装形式在 SoC 芯片上侧搭载了一块 512 MB 的 DDR2 DRAM 内存。

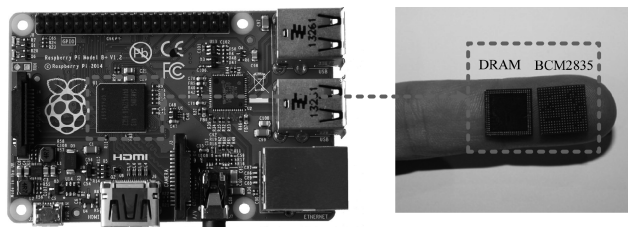


图3 Raspberry Pi B+ 开发板及其 PoP 封装^[10]

2.2 软件设计

本文利用模糊提取算法提出一种基于 DRAM PUF 的高效密钥发生器方案,并在主嵌入式平台 Raspberry Pi B+ 设备上实现方案。该方案能够快速提取均匀分布的 DRAM PUF 序列,并输出稳定的 DRAM PUF 密钥,实现流程如图 4 所示。

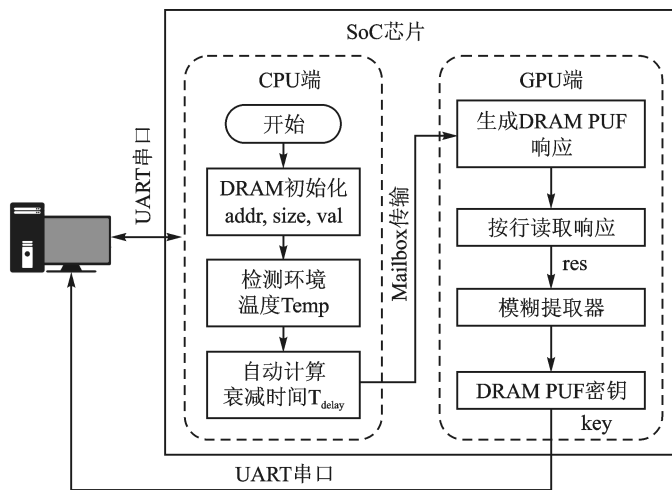


图4 基于 DRAM PUF 的密钥发生器流程

基于模糊提取算法的 DRAM PUF 密钥发生器主要由 PC 端和 SoC 端两部分组成。其中,PC 端的上位机中运行控制代码,通过 UART 接口将 minicom 串口通信工具中设置的 DRAM 初始化参数传送至 SoC 端。SoC 端包含 CPU 端和 GPU 端两部分,CPU 端负责通过 SoC 内自带的温度传感器感知周围的环境温度 Temp,进行 DRAM PUF 的电荷衰减时间 T_{delay} 的自动计算^[14],通过 Mailbox 通信机制将参数传递至 GPU 端,最后由 GPU 端完成 DRAM PUF 响应的按行读取,并通过模糊提取算法实现

稳定的 DRAM PUF 密钥输出。

2.2.1 CPU 端工作实现

CPU 端的主要工作是完成 UART 初始化以及获取 DRAM PUF 提取的测试参数,所有参数通过 Mailbox 机制从 CPU 端传送至 GPU 端,具体流程如下:

① CPU 中的控制代码首先调用 `uart_init()` 函数完成 UART 接口的初始化工作,使得接口能够正常地完成数据的接收和发送功能。

② 调用 `get_address()` 函数和 `init_val()` 函数,通过 UART 接收从 PC 端发送的 DRAM 初始化参数 `addr`、`size`、`val`,后续 GPU 端接收相关参数后完成 DRAM 测试区域的初始化操作。

③ 对 SoC 内部搭载的温度传感器相关的控制寄存器 `TS_TSENSCTL` 和状态寄存器 `TS_TSENSSTAT` 进行相关设置,通过温度感知函数 `get_temp()` 利用 SoC 芯片内部的温度传感器测量当前的测试温度 `Temp`。

④ 根据 DRAM PUF 温度-衰减时间的量化关系公式^[15],基于预先测得并存储与系统内核中的标准数据 (T_1, t_1),衰减时间计算模块将自动计算出在当前温度 `Temp` 下 DRAM PUF 序列达到 50% 翻转率时需要的衰减时间 T_{delay} 。其中, t_1 表示在 T_1 环境温度下达到 50% 翻转率所需的电荷衰减时间。

⑤ 由 Mailbox 通信机制将 CPU 收集到的 DRAM 初始化参数 `addr`、`size`、`val` 和 DRAM PUF 序列达到 50% 翻转率的延迟时间 T_{delay} 传送至 GPU 端。

2.2.2 Mailbox 连接部分工作实现

Mailbox 为一种 CPU 与 GPU 之间实现核间通信的机制,CPU 端和 GPU 端各有一个 Mailbox 用于实现相互之间的通信。首先,PC 端通过 UART 将 DRAM 初始化参数发送至 CPU 端后,CPU 端中的内核代码将接管 CPU 端的 Mailbox,并将先前获得的 DRAM 初始化及内核代码自动计算得出的衰减时间等参数发送至 GPU 端的 Mailbox 中。与此同时,GPU 受控于固件代码,在完成内核引导之后进入中断监控状态,当位于 GPU 端的 Mailbox 收到上述参数信息后,会触发中断服务程序,开启 DRAM PUF 响应的生成与读取以及后续的 DRAM PUF 密钥的生成工作。

2.2.3 GPU 端工作实现

GPU 端的主要工作为接收 DRAM 测试参数、生成 DRAM PUF 响应并进行按行读取以及实现模糊提取算法,最终生成稳定的 DRAM PUF 密钥,并通过 UART 接口将密钥传送至 PC 端的上位机中。GPU 端的工作包括以下几个步骤:

① 通过 Mailbox 接收参数信息,触发中断服务程序,

即 DRAM PUF 提取程序。根据 2.2 小节中介绍的基于电荷衰减原理的 DRAM PUF 实现方法生成 DRAM PUF 原始响应。

② 读取每一行中的位翻转数,如果单行中有一个 DRAM 单元或者多个 DRAM 单元发生位翻转,则将该行记为逻辑“1”;否则,记为逻辑“0”,最终将按行读取的 DRAM PUF 响应数据 `res` 临时存储在一个数组中。

③ 将读取的 DRAM PUF 响应作为密钥发生器的数据源,调用模糊提取算法的 `gen()` 函数,该模糊提取算法使用的是一阶 Reed-Muller 码 $RM(7,1)$ 进行纠错,对原始 DRAM PUF 序列中由于外界因素而产生的错误信息进行纠错,若噪声的影响在纠错码能力范围之内,则能够重新构造出与注册阶段生产的完全相同的 128 位均匀分布的安全密钥 `key`。

④ 通过 UART 接口将 GPU 端生成的安全密钥 `key` 传输至 PC 端中的上位机中,便于其他程序加密、解密等应用操作的调用。

3 性能测试与分析

3.1 可靠性分析

DRAM PUF 密钥的可靠性是密钥发生器的一项重要性能指标,表示能否对不同环境温度和电路老化效应等因素带来的错误噪声能否进行成功纠错、稳定生成密钥的能力。本实验选取每块 Raspberry Pi B+ 开发板第一次生成的 128 位 DRAM PUF 密钥作为标准值,同时基于 Python 3.6 语言编写了自动测试脚本,判断此后在不同环境温度下生成的 128 位 DRAM PUF 密钥是否与初始密钥相同,即测试密钥是否保持良好的可靠性。

如图 5 所示,3 块 Raspberry Pi B+ 开发板共检测 30 000 次,最终通过密钥比较的 Secret key counting 次数为 30 000,检测到与标准密钥不同的 Error key counting 数量为 0。实验结果表明,密钥发生器生成的 128 位 DRAM PUF 密钥的可靠性超过 99%,生成超过 30 000 个密钥中错误密钥的个数小于 1,具有良好的可靠性。

3.2 唯一性分析

唯一性作为 DRAM PUF 密钥的另一项重要安全参数,表示面对相同的输入激励时,不同开发板生成的 DRAM PUF 密钥之间的差异性。本小节分别从使用同一型号 DRAM 芯片的 3 块 Raspberry Pi B+ 开发板中生成 3 组 128 位的 DRAM PUF 密钥,并比较它们之间的差异性。基于 Python 3.6 语言编写 DRAM PUF 密钥唯一性数据分析脚本,分别测试每两块 Raspberry Pi B+ 开发板生成的 DRAM PUF 密钥之间的唯一性,具体数据如图 6 所示。

```

MINGW64/c/Users/56880/Desktop/DRAM PUF key Test/robustness
Key comparison times is 9996
Raspberry Pi1 key comparison is successful
Raspberry Pi2 key comparison is successful
Raspberry Pi3 key comparison is successful
Secret key counting is 29988
Error key counting is 0

Key comparison times is 9997
Raspberry Pi1 key comparison is successful
Raspberry Pi2 key comparison is successful
Raspberry Pi3 key comparison is successful
Secret key counting is 29991
Error key counting is 0

Key comparison times is 9998
Raspberry Pi1 key comparison is successful
Raspberry Pi2 key comparison is successful
Raspberry Pi3 key comparison is successful
Secret key counting is 29994
Error key counting is 0

Key comparison times is 9999
Raspberry Pi1 key comparison is successful
Raspberry Pi2 key comparison is successful
Raspberry Pi3 key comparison is successful
Secret key counting is 29997
Error key counting is 0

Key comparison times is 10000
Raspberry Pi1 key comparison is successful
Raspberry Pi2 key comparison is successful
Raspberry Pi3 key comparison is successful
Secret key counting is 30000
Error key counting is 0

All sequence comparison is completed.
The total number of comparisons is 30000.
Secret key counting is 30000.
Error key counting is 0.

```

图5 DRAM PUF 密钥可靠性脚本测试

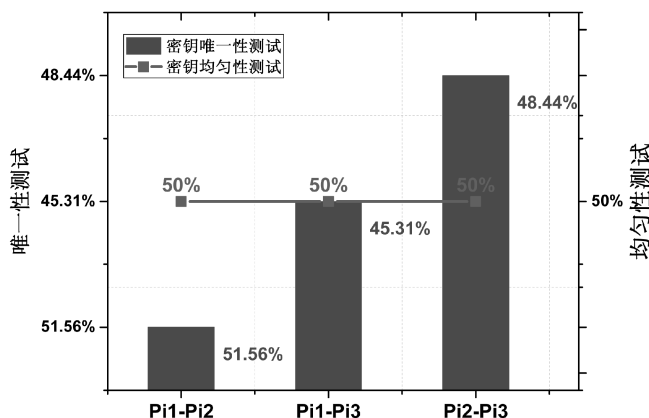


图6 DRAM PUF 密钥唯一性与均匀性测试

图6中的柱状图显示,3组DRAM PUF密钥之间的唯一性分别为51.56%、45.31%和48.44%,DRAM PUF密钥的唯一性均值约为48.44%,接近理想数值50%,说明基于DRAM PUF的密钥发生器生成的128位DRAM PUF密钥具有较好的唯一性。

同时,此实验结果表明,即使是相同型号的DRAM芯片,由于DRAM自身内在的制造差异,即使在同一激励的条件下,提取的原始DRAM PUF响应序列之间仍然存在较大的差异,这种源于DRAM芯片本身的差异性超过了模糊提取器中Reed-Muller纠错码的纠错能力,最终使得不同嵌入式设备生成的128位DRAM PUF密钥之间保持较大的差异性,具有良好的唯一性。

3.3 均匀性分析

为了保证DRAM PUF密钥的安全性,在理想情况

下,128位DRAM PUF安全密钥中的逻辑“0”和逻辑“1”的占比应相同,即128位的序列中“0”和“1”各为50%,以防止被攻击者采取暴力攻击等恶意手段获取准确的密钥,进而保证整个系统的安全性。本节基于Python 3.6脚本语言统计了3块Raspberry Pi B+开发板生成的3组128位DRAM PUF密钥中逻辑“0”和逻辑“1”的占比情况,如图6中折线图所示。

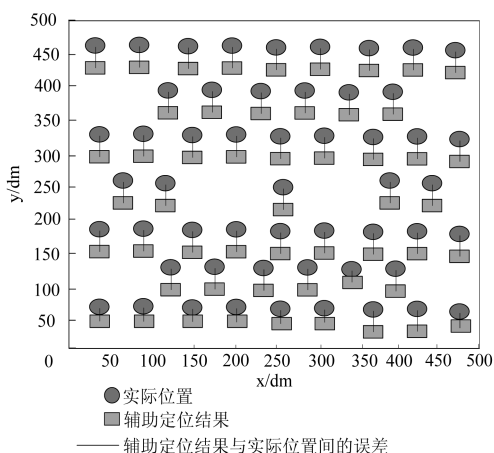
DRAM PUF密钥的均匀性测试结果显示3组128位的DRAM PUF密钥中逻辑“0”和逻辑“1”的数量均各为64位,密钥的均匀性全部为50%。同时,3块Raspberry Pi B+开发板生成的3组DRAM PUF密钥的信息熵值均为1,生成的密钥均为满熵密钥。实验结果表明,基于DRAM PUF的高效密钥发生器生成的128位密钥为满熵密钥,表现出良好的均匀性。

4 结语

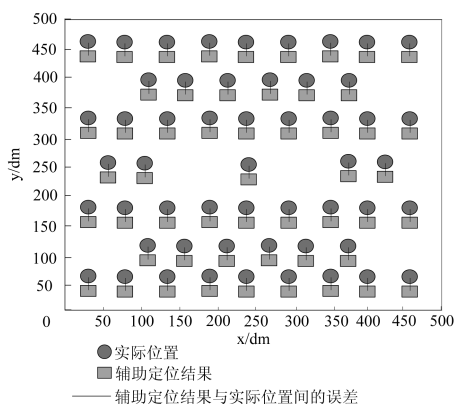
本文提出了一种基于DRAM PUF的高效密钥发生器的设计方案,在Raspberry Pi B+嵌入式设备平台实现了DRAM PUF序列的快速提取,并且兼具良好的安全特性,并以此作为密钥发生器的数据源,生成满足安全性需求的128位满熵DRAM PUF密钥,具有一定的实际工程应用价值。

参考文献

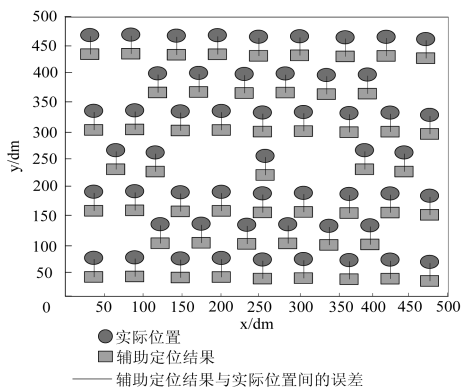
- [1] Mutlu O. Memory scaling: A systems architecture perspective [C]//Proc. Of 5th IEEE International Memory Workshop. IEEE, 2013; 21-25.
- [2] Medaglia C M, Serbanati A. An overview of privacy and security issues in the internet of things [C]//Proc. of the Internet of Things, Springer, New York, 2010; 389-395.
- [3] Pappu R, Recht B, Taylor J, et al. Physical one-way functions [J]. Science, 2002, 297(5589): 2026-2030.
- [4] 美高森美携手 IntrinsicID 为 Polar Fire FPGA 提供 SRAM-PUF [J]. 单片机与嵌入式系统应用, 2017, 17(8): 88.
- [5] Harry Luan. 一种成本更低的全新静态 DRAM 存储单元 [J]. 单片机与嵌入式系统应用, 2017, 17(1): 80-81.
- [6] 张振华. DRAM PUF 的设计与实现 [D]. 武汉: 华中科技大学, 2015.
- [7] Schaller A, Xiong W, Anagnostopoulos N A, et al. Decay-based DRAM PUFs in commodity devices [J]. IEEE Transactions on Dependable and Secure Computing, 2018, 16(3): 462-475.
- [8] 张紫楠, 郭渊博. 物理不可克隆函数综述 [J]. 计算机应用, 2012, 32(11): 3115-3120.
- [9] 任智源, 杨伦, 柴凯中. DRAM 的现状与发展方向 [J]. 电子元件与信息技术, 2022(4): 6.
- [10] Alan M. Computer Science, Maths, Raspberry Pi [EB/OL]. [2023-05]. <https://www.cl.cam.ac.uk/~am21/teaching/Cutlers13.pdf>.



(a) 优化前的辅助定位效果



(b) 优化后的辅助定位效果



(c) 参考文献[6]方法优化后的辅助定位效果

图4 优化前后的医疗设备辅助定位效果

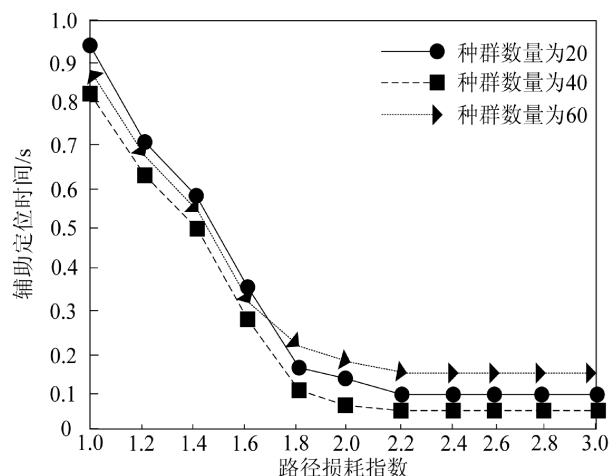


图5 医疗设备辅助定位时间分析结果

备管理人员的医疗设备盘点效率,以及医疗设备盘点准确性。

参考文献

- [1] 陈含蓓,王方良. 计及小信号稳定的医疗设备产品振动信号监测[J]. 计算机仿真, 2020, 37(10): 325 - 328.
- [2] 张章,韦玲玲,闫林,等. 用于植入式医疗设备的高 PSRR 无片外电容 LDO[J]. 微电子学, 2020, 50(1): 55 - 59.
- [3] 胡东海,邵元,陈莹,等. 基于改进粒子滤波的室内自适应定位算法[J]. 计算机工程与应用, 2019, 55(9): 65 - 71, 167.
- [4] 刘岚,刘立,屈爱平,等. 基于物联网定位技术的无人机急诊医疗物流平台建设的研究[J]. 中华急诊医学杂志, 2021, 30(7): 872 - 873.
- [5] 戴丽,袁琴,朱顶贵,等. 基于 CSS 及边缘计算的移动医疗设备定位管理系统研究[J]. 电子器件, 2021, 44(1): 192 - 196.
- [6] 严洪立,关俊文,李永红,等. 基于增强现实理论的神外颅体表定位系统[J]. 生物医学工程学杂志, 2019, 36(3): 428 - 434.
- [7] 刘伟,李卓,杨晓斐,等. 基于 SCE - PSO 的无线传感器网络节点三维定位方法[J]. 传感技术学报, 2019, 32(5): 756 - 761.

王旭(高级工程师),主要研究方向为医院信息化。通信作者:王旭, jjihe05070899@163.com。

(责任编辑:薛士然 收稿日期:2023-05-04)

- [11] 韩宇. 基于 PUF 的密钥存储安全技术[J]. 信息技术与信息化, 2020(12): 3.
- [12] 李森森,黄一才,郁滨,等. 基于物理不可克隆函数的蓝牙密钥生成器[J]. 电子测量与仪器学报, 2018(2): 9.
- [13] 黄春光,程海,丁群. 基于 PUF 的 Logistic 混沌序列发生器[J]. 通信学报, 2019, 40(3): 8.
- [14] Chen S, Li B, Cao Y. Intrinsic Physical Unclonable Function (PUF) Sensors in Commodity Devices[J]. Sensors, 2019, 19(11): 24 - 28.

- [15] Xiong W, Anagnostopoulos N A, Schaller A, et al. Spying on Temperature using DRAM[C]//Proc. of the Design, Automation & Test in Europe Conference & Exhibition (DATE), Florence, Italy, 2019: 13 - 18.

谷敏章(工程师),主要研究方向为芯片信息安全应用。通信作者:谷敏章, 568805843@qq.com。

(责任编辑:薛士然 收稿日期:2023-05-23)