基于 PUF 的 Logistic 混沌序列发生器

黄春光，程海，丁群
（黑龙江大学电子工程学院，黑龙江 哈尔滨 150080）

摘 要：由于 Logistic 非线性混沌系统在一定的参数下，具有初值敏感性和拓扑复杂性等特点，因此 Logistic 混沌系统可以作为随机序列信号发生器。同时由于集成电路在生产、制作的过程中，即使采用完全相同的设计方法和制造工艺，也会在器件上产生不可控的微小差异，这些微小差异便成为集成电路不可克隆的基础。基于此特点，提出了一种基于可编程逻辑阵列（FPGA）的双输出查找表（LUT）结构的物理不可克隆函数（PUF）的 Logistic 随机混沌序列信号发生器，该混沌序列发生器具有物理的唯一性，能够有效地抵抗对于系统的复制和攻击。将该系统在 Xilinx 公司的 FPGA 开发板上进行测试和验证，结果表明，同样的电路结构和配置文件在不同的 FPGA 开发板上能够产生不同的随机序列，提高了混沌序列的随机性。

关键词：Logistic 混沌系统；物理不可克隆函数；序列发生器；流密码
中图分类号：TP301
文献标识码：A
doi: 10.11959/j.issn.1000−436x.2019064

Logistic chaotic sequence generator based on physical unclonable function

HUANG Chunguang, CHENG Hai, DING Qun
Electronic Engineering College of Heilongjiang University, Harbin 150080, China

Abstract: Logistic nonlinear chaotic system has many good characters such as initial value sensitivity and topological mixing in the some parameter condition, which is used to create the random sequence signal generator. Because of the attributions of randomness and uniqueness even under the exact, the same circuit layouts and manufacturing procedures, there is still an instinct unclonable difference in each integrated circuit. Therefore, a new sequence stream generator was proposed based on Logistic chaotic system and physical unclonable function designed by double output look-up-table (LUT). The output of the Logistic sequence generator was associated with a specific physical circuit. This kind of sequence generator could resist an attack such as the replication of the keys of the system. The system was designed and tested on the Xilinx FPGA board. The results show that the same architecture of the circuit and the same config file operated on the different FPGA developing board can generate the total different random chaotic sequence stream and improve the randomness of the stream.

Key words: Logistic chaotic system, physical unclonable function, random number generator, stream ciphe

1 引言

随着通信技术的飞速发展及互联网络和移动网络的广泛使用，人们越来越关注信息的安全。混沌系统由于具有初值敏感性和拓扑复杂性等特点，得到了广大研究人员的关注[1]。

混沌系统是指在一个确定性的系统中，存在着随机的不规则运动，这种运动具有不确定性、不可

收稿日期：2018−05−14；**修回日期**：2018−08−16
通信作者：丁群，qunding@aliyun.com
基金项目：国家自然科学基金资助项目（No. 61471158, No.61571181）
Foundation Item: The National Natural Science Foundation of China (No.61471158, No.61571181)

重复性以及不可预测性，是非线性动力系统的固有特性。Logistic 混沌映射是一种一维混沌系统，当该系统的参数在一定的范围内时，系统会进入混沌状态。由于混沌特性的存在，Logistic 混沌映射系统可以应用在随机信号发生器上。为了便于数字电路和 FPGA 应用 Logistic 混沌系统，通常采用离散的 Logistic 混沌映射系统进行设计和开发[2-3]。由于离散混沌系统具有状态随机性和初值敏感性等特点，因此可以将其应用到数字图像加密[4]、随机数发生器[5]以及数字认证等相关领域。

物理不可克隆函数（PUF, physical unclonable function）是一种对集成电路芯片在制造过程中产生的细微差异进行放大并使其产生独一无二的物理不可克隆特征的函数。大多数物理不可克隆函数的实现是通过计算和分析电路的时延信息的差异进行设计的。想要对物理不可克隆函数模块进行复制，需要复制相同的逻辑单元、相同的布局布线，同时在物理电路上进行精确的设计，才能得到相同的结果。但是对于具体的物理电路而言，电路结构和布局布线的差异是由电路自身的物理参数决定的，无法进行克隆复制，因此物理不可克隆函数提供了一种可以在相同的电路结构上产生不同的差异的模块。由于 PUF 具有基于器件特征的固有随机性，很多研究人员将 PUF 应用到设备的认证环节[6]和通信环节[7]。

可编程逻辑阵列（FPGA, field-programmable gate array）是一种可编程的芯片，能够根据用户的需求通过软件手段进行更改、配置内部连接结构和逻辑单元、完成指定设计功能的数字电路。由于可编程逻辑阵列具有开发简单方便、实现效率高等优点，已广泛应用到电路设计、嵌入式系统等各个领域。在进行可编程逻辑阵列应用开发的过程中，可以使用芯片提供商提供的开发软件进行逻辑电路的设计和仿真，然后下载到可编程逻辑阵列中。在使用过程中，可以将生成的比特流数据下载到相应的逻辑电路中，完成对于系统的配置。由于电路结构相同，可以将相同的比特流下载到不同的 FPGA 逻辑电路中，得到相同的功能。由于物理不可克隆函数的存在，即使相同的配置文件，也会产生不同的结果，因此可以对设备进行识别和身份确认。

FPGA 具有配置灵活、设计方便等特点，非常适合电路设计。Dabal 等[8]利用 Xilinx 公司的 FPGA 设计了基于 Logistic 混沌系统的随机序列，作为流密码发生器。Wang 等[9]利用 Simulink 工具设计了基于 FPGA 的混沌序列信号发生器，并对相关性进行分析。但是，由于数字 Logistic 混沌系统产生的序列和器件结构无关，因此容易被复制、攻击和探测。基于此，本文设计了一种基于物理不可克隆函数的 Logistic 混沌序列发生器，利用 FPGA 硬件结构的细微差异产生基于器件特征的混沌随机序列，并利用 FPGA 中的基本单元——查找表（LUT, look-up-table）在不同器件上传输时延的细微差异进行设计，所产生的混沌序列与物理器件相关，相同的电路在不同的 FPGA 器件上可以产生完全不同的结果。该系统可以作为安全设备的终端，利用在服务器上存储的硬件特征完成设备的验证或数据的加密传输。

2　相关工作

2.1　国内外发展

目前，在通信领域、视频传输、网络通信和个人消费电子产品等相关领域都需要基于 FPGA 加密技术的数字认证环节[10]。Wang 等[11]提出了一种基于 FPGA 公钥与私钥对的知识产品保护方案。Guajardo 等[12]设计了一种基于 PUF 的加密认证系统。Tuncer 等[13]利用 Logistic 混沌系统作为 PUF 的初始随机向量，然后对 Logistic 混沌系统的输出进行判断，产生“0”和“1”的随机序列。该系统采用 Logistic 信号发生器作为 PUF 的随机输入。由于 PUF 的状态有限，没有充分利用混沌系统的状态随机性和初值敏感性，而且该系统的 PUF 模块在四输入单输出的 LUT 模块上实现，利用率低，因此只对该系统在单个开发板上进行了测试，没有将其应用到其他开发板，以测试 PUF 的特性。Liu 等[14]给出了一种基于碳纳米管的物理不可克隆函数，采用的混沌方程为 Lorenz 混沌方程，利用碳纳米管工艺的不确定性，生成 PUF。但是由于碳纳米管生成的序列位数有限，而且采用模拟电路进行设计，不利于大规模开发和使用。

2.2　物理不可克隆函数

自 2002 年以来，人们对基于硅电路的 PUF 电路进行了深入的研究。到目前，研究人员已经提出了各种物理不可克隆的方法，例如，基于随机存储器的 PUF、基于触发器的 PUF、基于蝴蝶类型的 PUF、基于仲裁器的 PUF、基于环形振荡器的

PUF[15]、基于电路毛刺的 PUF[16]等。产生 PUF 的方法，大体上可以分为 2 种，介绍如下。

一种为采用随机存储的方式设计的 PUF 电路。由于存储电路是由一种状态可以改变的双稳态电路组成的，因此存储电路在通电之前会处于一种随机状态。存储电路在通电之后，寄存器上的状态是一种随机状态，此时存储电路上的高低电平可以作为 PUF 使用。但是对于芯片设计厂商来说，为了避免这种状态的出现，在电路通电之后，会进行强制复位。只有在设计电路时进行单独设计，才能够使用。

另一种为采用时延的方式设计的 PUF，利用物理器件上门电路的时延和布线的时延差异进行设计。在实现方式上，主要分为 2 种，一种是仲裁型 PUF，另一种是环形振荡器 PUF。

1) 仲裁型 PUF

采用 2 路相同的传输路径，由于集成电路制造工艺的细微差异会产生不同的时延，判决器根据 2 路传输路径到达时间的差异决定输出是“0”还是“1”，从而产生基于器件自身特点的随机信号。仲裁型 PUF 的控制端控制信号传输的路径，系统输入端输入同一个脉冲信号，利用控制信号进行路径选择，使信号经过不同的传输路径，根据信号到达仲裁器的先后顺序进行判决。

在 PUF 设计过程中，需要尽量保证传输路径的长度一致，但是由于物理元件的差异，最终会导致同一个脉冲到达仲裁器的时间不尽相同。如图 1 所示，仲裁类型 PUF 利用控制信号（C_0,C_1,…）选择信号传输的路径，输入的是同一个脉冲信号，经过不同的路径到达仲裁器，仲裁器根据脉冲信号到达的先后顺序进行判决，决定输出是“0”还是“1”。这类 PUF 需要的多路选择器级别多、路径长，否则无法分辨到达的先后顺序，即使能够判决，也容易被外界环境干扰。

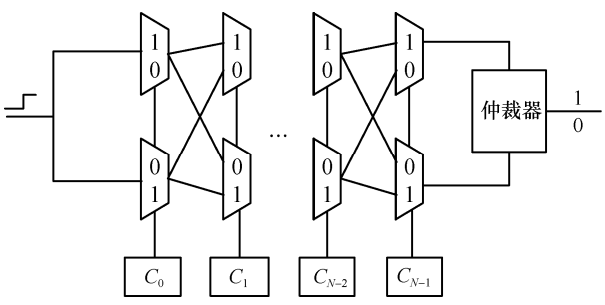


图 1 仲裁型 PUF

2)环形振荡器 PUF

利用反相器级联形成振荡电路，通过计数器进行计数。由于物理电路结构的微小差异，经过多次计数，2 个计数器的结果不尽相同，然后通过比较器，得到最终的结果。

图 2 中与门作为控制端，控制环形振荡器 PUF 的起振和停止。与门和后面的多个反相器相连接，构成一个连续反向的环。该环形振荡器产生的脉冲信号作为计数器的时钟，进行计数。计数器设定计数的截止上限，当其中任意一个计数器计数达到上限，则环形振荡器停止工作。在进行布局布线过程中，要求保证路径长度一致。比较器通过比较 2 个计数器数值的大小，输出“0”或“1”。

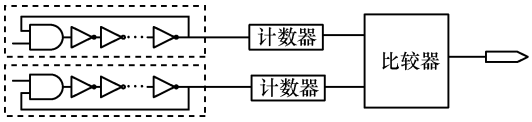


图 2 环形振荡器 PUF

在环形振荡器 PUF 中，如果反相器的个数为奇数，需要使用与门作为控制模块，控制环形振荡器的工作状态；如果环形振荡器中反相器的个数为偶数，则采用与非门作为控制模块。

环形振荡器 PUF 所需的与非门的个数少，相对仲裁型 PUF 来说占用资源少，外界环境影响小。

2.3 Logistic 混沌方程

Logistic 方程为混沌系统的经典实例，系统结构简单，便于 FPGA 设计、实现和验证，同时便于推广到其他混沌系统。

Logistic 函数的数学表达式为

$$x_{n+1}=\mu x_n\left(1-x_n\right),n=0,1,2,3\cdots \tag{1}$$

其中，μ 是 Logistic 方程的系统参数，系统的初值设为 x_0（$0<x_0<1$）。当 $3.569\,999<\mu\leqslant 4$ 时，系统进入混沌状态。

如图 3 所示，随着 μ 的增加，Logistic 混沌系统分岔越来越多，系统逐渐进入混沌状态。如果在 FPGA 上实现 Logistic 混沌电路，则需要对 Logistic 方程进行离散化，Logistic 方程的每次迭代结果采用 N 位二进制数表示，即

$$X'_{n+1}=x'_{N-1}2^{N-1}+x'_{N-2}2^{N-2}+\cdots+x'_0 2^0 \tag{2}$$

式(2)是 Logistic 方程的二进制表现形式。其中，X'_{n+1} 为二进制的表现形式，x' 是二进制每一位的值。N 的位数越大，Logistic 方程的精度越好，同时需要的运算量就越大。

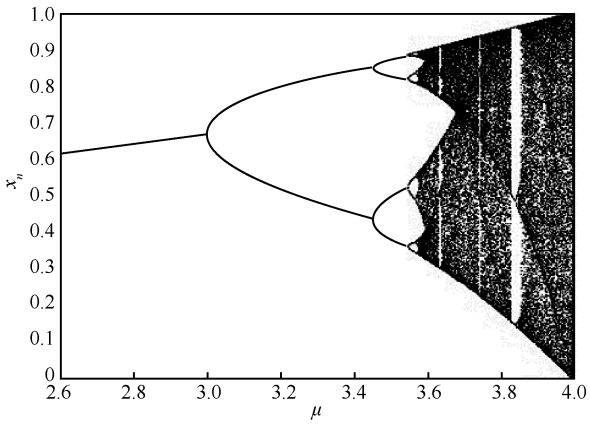


图 3　Logistic 分岔图

3　系统设计

基于环形振荡器的 PUF，利用有限个数的反相器，构建一个反馈链。基于仲裁型的 PUF，可以利用控制位选择相应的路径，根据判决器的结果产生“0”或“1”。本文提出基于混合型 PUF 的 Logistic 序列发生器，同时利用环形振荡器 PUF 与仲裁型 PUF 的特点，产生基于物理器件特征的随机数。利用 Logistic 混沌方程对系统初值的敏感性（如图 3 所示），初值的细微差异会产生完全不同的变化轨迹。利用 Logistic 方程的特点，将 PUF 与 Logistic 迭代方程相结合，既利用 PUF 的硬件唯一性，又利用 Logistic 函数的敏感性，产生与硬件相关的混沌序列，同时在 Xilinx 公司的 7-series FPGA 上进行设计和验证。

3.1　基本单元模块

本文采用 FPGA 进行设计和开发。由于 FPGA 具有大量的可编程资源，为系统的实现提供了相应的功能模块，既解决了定制电路的不足，又克服了原有可编程器件门电路有限的缺点。Xilinx 公司的 FPGA 具有 6 输入 2 输出的 LUT 模块，有利于多输入多输出的应用模块设计。

Xilinx 公司的 7-series FPGA 的基本单元为 CLB，每个 CLB 由 2 个 SLICE 组成，每个 SLICE 由 4 个 5 输入的 LUT、3 个 MUX、一个 CARRY 和 8 个 FF 组成。

如图 4 所示，Xilinx 公司 7-series FPGA 的每个 LUT 有 6 个输入端口，当 A6 设置为高电平时，可以作为 2 个 5 输入的 LUT 使用，可以有 2 个输出。在设计过程中，需要对 LUT6_2 模块进行配置，将其配置为 2 输入 2 输出的与门电路和 2 输入 2 输出的非门电路。

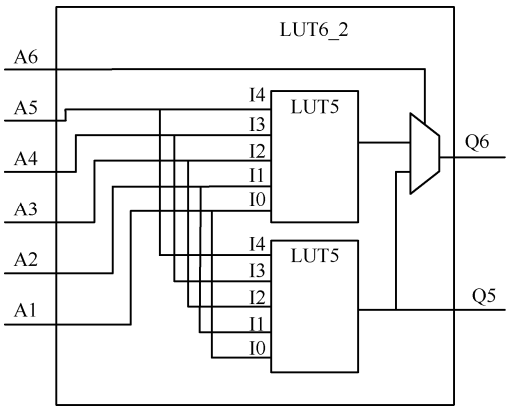


图 4　FPGA 的基本单元 LUT6_2

对于图 5 中的 2 输入 2 输出非门电路，即图中虚线内部的模块，可以用一个 LUT6_2 进行设计。可以充分利用 LUT6_2 的 2 个输出的特性，将 A6 设置为高电平，而 A5 和 A4 输入没有使用，可以配置高电平或低电平，不能悬空使用。在本设计中，A4 和 A5 设置为低电平，这样就可以使用 Q5 和 Q6 这 2 个输出，则电路等效如图 5 所示。

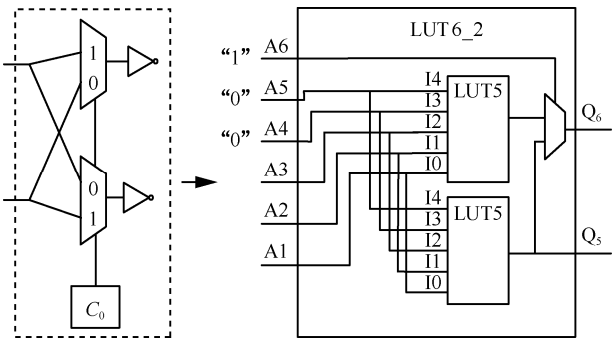


图 5　基于 LUT6_2 的双路非门电路

每个模块有 3 个输入，其中，A3 作为选择信号，A2 和 A1 作为数据的 2 个输入端口，此时，该电路模块为 3 输入 2 输出的功能模块。根据真值表进行计算，可以得到 LUT6_2 的配置码为 0x0000c0a0_0000a0c0，此时 LUT 可实现双路非门和路径选择的功能。

同理，将 LUT6_2 设计为 2 输入 2 输出的与门时，每个模块有 4 个输入，其中，A6 为高电平，A5 为低电平，A4 作为使能信号，A3 作为选择信号，A2 和 A1 作为数据的 2 个输入端口，则根据电路的功能设置相应 LUT 的数值。根据真值表进行计算，可以得到 LUT6_2 的配置码为 0x00000035_00000053，此时 LUT 可以实现双路与门及路径选择的功能，如图 6 所示。

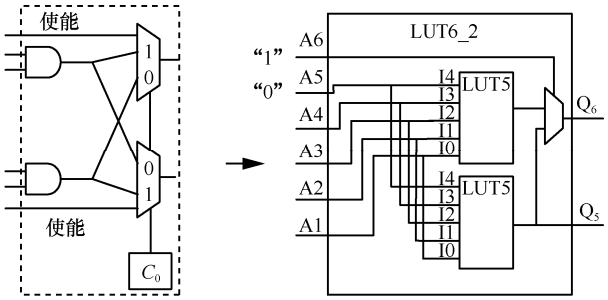


图 6　基于 LUT6_2 的双路与门电路

3.2　系统框图

本文设计一种基于 Logistic 方程的 PUF，系统框图如图 7 所示。利用 LUT6_2 进行配置生成与门和非门，利用控制端口选择信号传输的路径，形成环形振荡器 PUF，通过改变控制端口信号的状态，可以改变信号传输的路径，得到不同的随机数。如果一个系统有 N 级选择电路，则可以产生 2^N 种状态。

系统工作流程如图 8 所示。序列发生器首先利用 PUF 产生的随机数作为 Logistic 每次迭代运算的初值 x_n 所对应的二值序列的低位，然后代入混沌方程进行迭代，产生 x_{n+1}，再将 x_{n+1} 的二值序列的高位作为 PUF 控制信号，最后利用 PUF 生成的随机序列构建 Logistic 方程的初值 x'_{n+1}，并进行迭代运算。

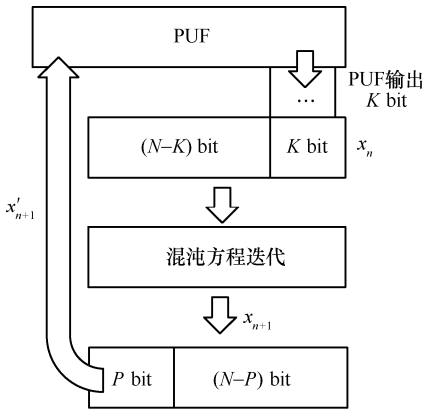


图 8　系统工作流程

系统工作状态机如图 9 所示，分为 5 个步骤。

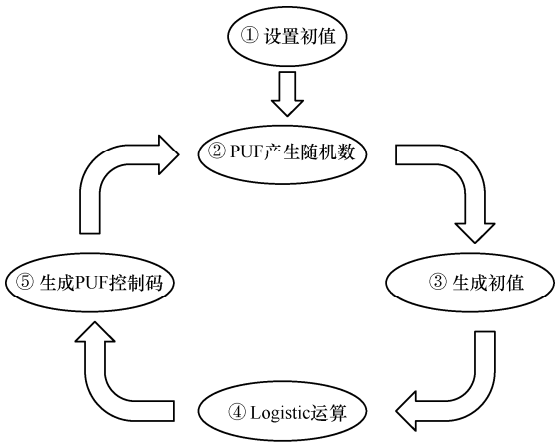


图 9　系统工作状态机

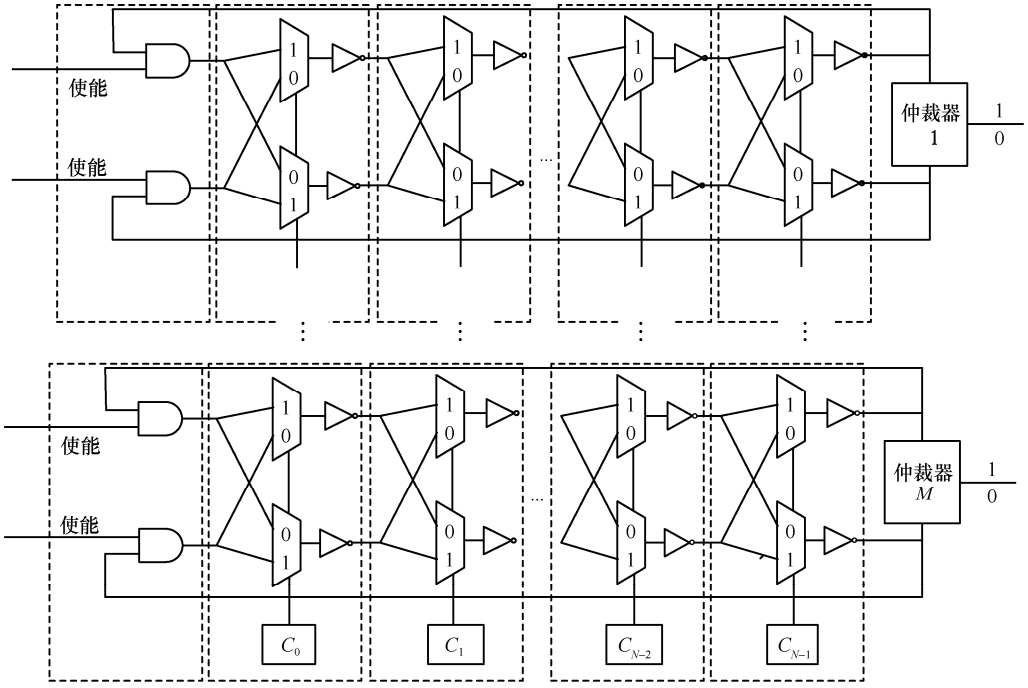


图 7　系统框图

步骤 1　设置系统的初值，PUF 的结构有 P 个控制端，用来选择 PUF 的不同路径。由于 PUF 的特点，不同的 PUF 所产生的结果不完全一样。

步骤 2　利用 PUF 模块生成 K bit，将 K bit 传递给 Logistic 混沌系统。

步骤 3　利用 PUF 传递过来的 K bit 和原有的 N bit 的后 K bit 进行异或处理，得到 x_n。

步骤 4　将生成的 N bit 的数据输入 Logistic 混沌系统进行迭代运算，生成 x_{n+1}。

步骤 5　将 x_{n+1} 的前 P bit 作为 PUF 的控制信号，生成下一轮迭代运算的初值 x'_{n+1}。

4　系统分析与测试

在本文系统中，由于 PUF 的存在，使原来混沌系统无法迭代到的点重新成为系统的初值，破坏了离散混沌系统的周期性，为了测试基于 PUF 的 Logistic 混沌序列发生器的特性，采用 Xilinx 公司的 7-series FPGA 开发板进行测试，并对序列发生器的相关特性进行了分析和讨论。

4.1　实验环境

本文在 Xilinx 公司的 7-series FPGA 上设计了基于物理不可克隆函数的 Logistic 混沌序列发生器，设计软件采用 Xilinx 公司的 vivado 2017.4 版本进行设计开发，并利用 vivado 自带逻辑分析仪进行数据测试和分析。在实验过程中，对 4 块同样的 FPGA 开发板进行测试，通过 XDC 文件对模块的具体位置和走线利用约束条件进行限定，同时将 PUF 布置到 FPGA 的不同位置进行测试。

本系统利用 LUT6_2 设计了 32 路 PUF，通过对 32 路 PUF 的计数器输出的数值进行比较，得到 32 bit 的输出。

4.2　实验数据

在实验过程中，同样的配置代码在不同的 FPGA 开发板上的结果并不相同。首先，相同的 Logistic 初始向量作用到 PUF 上，不同的开发板得到的结果如表 1 所示，结果呈现出随机性。

表 1　不同开发板在相同配置下的输出结果

开发板	00000000	10101010	11111111	11110000
开发板 1	7f9dbb56	bfe7df9	22106ab8	f2461a6f
开发板 2	3a72c143	fd57fed	1a221223	69262cd5
开发板 3	81611c49	24c7c640	5534eb82	a48d9547
开发板 4	f6bc6a1e	a67905cb	31eb8529	658b6610

系统的硬件开销如表 2 所示。由于系统采用 LUT6_2 的单元电路，每个单元模块包括 6 个输入和 2 个输出，因此每个 LUT 模块可以实现 2 路 PUF 电路，由于 LUT 模块具有 6 个输入端口，可以选择其中一个端口作为控制端口，使得硬件的利用率得到提升。Tuncer 等[13]使用 LUT4_1 模块，每个模块可以实现一路 PUF 电路，利用率低。

表 2　硬件开销说明

方案	功能模块	每个功能模块包含单元电路	数量
本文方案	PUF	振荡器	64
		反相器	7
		与门	1
		计数器	32
		比较器	32
	Logistic	32 bit 乘法器	1
		比特重组	1
Tuncer 等[13] 方案	PUF	振荡器	128
		反相器	3
		与门	1
		计数器	2
		64 选 1 多路选择器	2
		比较器	1
	Logistic	32 bit 乘法器	2

在设计过程中，Logistic 混沌方程采用 64 bit 精度进行计算。PUF 单元的布局布线如图 10 所示，K=32，N=64。利用 PUF 的输出序列构建混沌系统的 x_n，利用 vivado 2017.4 中 Multiplier 模块进行乘法器设计，然后产生 64 bit 的 x_{n+1}。设计模块采用 32 位 PUF，选取 32 bit 的数值作为 PUF 的控制码进行下一轮迭代运算。每个模块的布局布线要求对称，以保证传输路径长度一致。

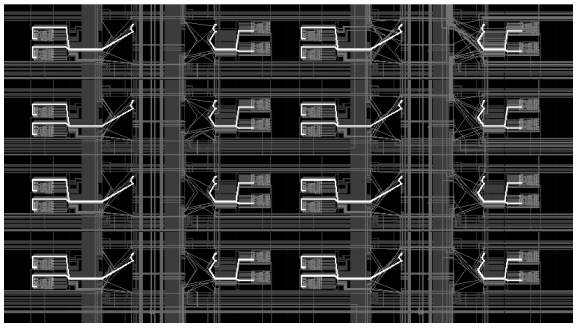


图 10　PUF 单元的布局布线

4.3 数据分析

4.3.1 相关性分析

为了验证本文改进算法对于 Logistic 混沌输出序列的影响，对 FPGA 输出的序列进行了测试。由于在 FPGA 上运行 Logistic 混沌方程，必然要对混沌系统进行离散化处理，混沌系统由于系统的离散化而发生性能的改变[17]。自相关性是衡量一个信号在不同时间的相互依赖关系，即系统的随机性是否相互独立。设 x_n 是“0”或“1”的二值序列信号，$R_x(m)$ 表示此信号的自相关函数，可以表示为

$$R_x(m)=\sum_{n=-\infty}^{+\infty}x(n)x(n+m) \tag{3}$$

通过对混沌序列发生器的输出进行分析和设计，对于原始的 Logistic 混沌系统，采用 32 bit 的精度进行计算，自相关函数如图 11 所示，对于不同的 Logistic 初值，系统会出现一些周期现象。

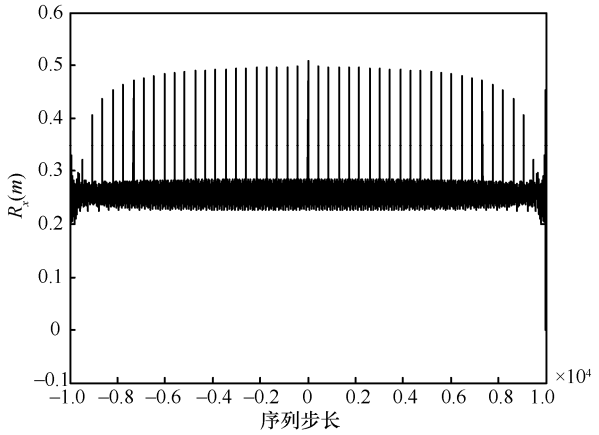


图 11 32 bit 的 Logistic 混沌方程自相关结果

如果采用基于 PUF 的 Logistic 混沌系统进行设计，可以阻止 Logistic 混沌系统周期现象的出现。同时，由于 PUF 系统的存在，使 Logistic 混沌轨道上无法迭代到的点可以成为系统的初值进行计算，改善了系统的周期性，如图 12 所示，系统不存在明显的周期性现象。

4.3.2 随机性分析

为了验证序列的随机性，本文采用美国国家标准与技术研究院（NIST, National Institute of Standard and Technology）提供的 15 种随机数检测方法 NIST SP 800-22 进行检验，评价指标 P 值的对比结果如表 3 所示，通过全部测试表明，基于 PUF 的混沌序列信号发生器产生的数据是满足随机性的要求的。在本文的方案中，$\mu=4$，而在 Tuncer 等[13]的方案中，Logistic 参数 $\mu=3.99$，相应地会增加一个乘法器来计算 Logistic 方程，带来硬件资源的消耗。

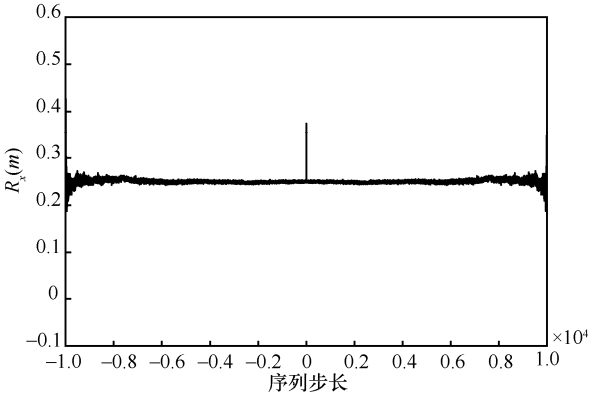


图 12 基于 PUF 的 Logistic 序列自相关函数

表 3 评价指标 *P* 值的对比结果

统计	本文方案 P 值（μ=4）	Tuncer 等[13]方案 P 值（μ=3.99）	结果
频率检验	0.322 332	0.757	通过
块内频数检验	0.155 513	0.135	通过
游程检验	0.979 957	0.801	通过
块内最长游程检验	0.951 515	0.497	通过
二元矩阵秩检验	0.767 895	0.336	通过
离散傅里叶变换检验	0.426 73	0.501	通过
非重叠模块匹配检验	0.999 103	0.698	通过
重叠模块匹配检验	0.640 417	0.630	通过
Maurer 的通用统计检验	0.999 914	0.435	通过
线性复杂度检验	0.351 439	0.644	通过
序列检验	0.185 09	0.444	通过
近似熵检验	0.319 519	0.949	通过
累加和检验	0.312 105	0.880	通过
随机游程检验	0.072 597	0.770	通过
随机游动状态频数检验	0.480 935	0.478	通过

5 结束语

本文提出了一种基于物理不可克隆函数（PUF）的 Logistic 混沌序列发生器，利用 FPGA 器件的 LUT 以及传输路径在生产过程中的工艺差异进行设计，所以即使同样配置码下载到不同的 FPGA 开发板上，产生的结果也不同。首先利用 Xilinx 的 FPGA 中的 LUT6_2 设计了 2 路 PUF 结构，相对于传统的 FPGA 而言，节省了一半的资源；然后利用 PUF 产生的随机数对 Logistic 混沌系统进行优化，设计了 Logistic 混沌序列信号发生器。该序列发生器产生

的序列与物理器件的特性相关，利用物理器件的特征参数作为序列发生器的原始密钥，保证了序列发生器的唯一性。NIST 检验表明，混沌序列信号发生器产生的序列满足随机性的要求。

参考文献：

[1] SBIAA F, BAGANNE A, ZEGHID M, et al. A new approach for encryption system based on block cipher algorithms and Logistic function[C]//International Multi-Conference on Systems, Signals & Devices. 2015: 1-5.

[2] KANSO A, SMAOUI N. Logistic chaotic maps for binary numbers generations[J]. Chaos Solitons & Fractals, 2009, 40(5): 2557-2568.

[3] 蔡丹, 季晓勇, 史贺, 等. 改进分段 Logistic 混沌映射的方法及其性能分析[J]. 南京大学学报(自然科学), 2016, 52(5): 809-815.
CAI D, JI X Y, SHI H, et al. Method for improving piecewise Logistic chaotic map and its performance analysis[J]. Journal of Nanjing University, 2016, 52(5): 809-815.

[4] XU L, LI Z, LI J, et al. A novel bit-level image encryption algorithm based on chaotic maps[J]. Optics & Lasers in Engineering, 2016, 78(21): 17-25.

[5] 宣蕾, 闫纪宁. 基于混沌的“一组一密”分组密码[J]. 通信学报, 2009, 30(Z2): 105-110.
XUN L, YAN J N. The “one group one cipher” cryptograph of block cipher based on chaotic[J]. Journal on Communications, 2009, 30(Z2): 105-110.

[6] 王俊, 刘树波, 梁才, 等. 基于 PUF 和 IPI 的可穿戴设备双因子认证协议[J]. 通信学报, 2017, 38(6): 127-135.
WANG J, LIU S B, LIANG C, et al. Two-factor wearable device authentication protocol based on PUF and IPI[J]. Journal on Communications, 2017, 38(6): 127-135.

[7] 郭渊博, 张紫楠, 杨奎武. 基于 PUFS 的不经意传输协议[J]. 通信学报, 2013, 34(Z1): 38-43.
GUO Y B, ZHANG Z N, YANG K W. Oblivious transfer based on physical unclonable function system[J]. Journal on Communications, 2013, 34(Z1): 38-43.

[8] DABAL P, PELKA R. FPGA implementation of chaotic pseudo-random bit generators[C]//Mixed Design of Integrated Circuits and Systems. 2012: 260-264.

[9] WANG H J, SONG B, LIU Q, et al. FPGA design and applicable analysis of discrete chaotic maps[J]. International Journal of Bifurcation & Chaos, 2014, 24(4): 917-921.

[10] ADAMO O, MOHANTYZ S P, KOIUGIANOS E, et al. VLSI architecture and FPGA prototyping of a digital camera for image security and authentication[C]//Region 5 Conference. 2006: 154-158.

[11] WANG Y, RENFA L I. FPGA based unified architecture for public key and private key cryptosystems[J]. Frontiers of Computer Science, 2013, 7(3): 307-316.

[12] GUAJARDO J, KUMAR S S, SCHRIJEN G J, et al. FPGA intrinsic PUFs and their use for IP protection[C]//International Workshop on Cryptographic Hardware and Embedded Systems. 2007: 63-80.

[13] TUNCER T. The implementation of chaos-based PUF designs in field programmable gate array[J]. nonlinear dynamics, 2016, 86(2): 1-12.

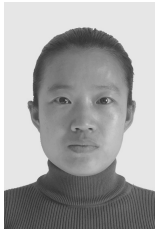
[14] LIU L, HUANG H, HU S. Lorenz chaotic system based carbon nanotube physical unclonable functions[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2017, PP(99): 1.

[15] KUMAR S. The buttefly PUF : protecting IP on every FPGA[C]//IEEE International Workshop on Hardware Oriented Security and Trust. 2008.

[16] 庞子涵, 周强, 高文超, 等. 高效能 FPGA 毛刺 PUF 设计与实现[J]. 计算机辅助设计与图形学学报, 2017, 29(6): 1135-1144.
PANG Z H, ZHOU Q, GAO W C, et al. Design and implementation of high efficiency PUF circuit on FPGA[J]. Journal of Computer-Aided Design & Computer Graphics, 2017, 29(6): 1135-1144.

[17] CHENG H, SONG Y, HUANG C, et al. Self-adaptive chaotic Logistic map: an efficient image encryption method[J]. Journal of Internet Technology, 2016, 17(4): 743-752.

[作者简介]



黄春光（1980− ），女，黑龙江哈尔滨人，黑龙江大学讲师，主要研究方向为保密通信、信息安全。

程海（1979− ），男，黑龙江哈尔滨人，黑龙江大学副教授、硕士生导师，主要研究方向为保密通信、信息安全。

丁群（1957− ），女，黑龙江哈尔滨人，黑龙江大学教授、博士生导师，主要研究方向为保密通信、信息安全。