文献引用格式： 韩炼冰，房利国，王松，等．基于 FPGA 的格密码关键运算模块的设计与实现 [J]. 通信技术，2022,55(12):1613-1617.
doi:10.3969/j.issn.1002-0802.2022.12.014

基于 FPGA 的格密码关键运算模块的设计与实现 *

韩炼冰，房利国，王　松，刘鸿博，杨敏旭
（中国电子科技集团公司第三十研究所，四川 成都 610041）

摘　要： 格密码是后量子密码中的一项重要技术，为提高格密码运算效率，提出了一种格密码中多项式乘法的硬件实现方法。该方法利用现场可编程门阵列（Field Program Gate Array，FPGA）内部存储器存放多项式系数，采用乒乓结构提高存储器并行读写速度，并通过预计算和预缩放简化计算过程，降低计算复杂度。同时，采用多级流水线技术，减少存取时间和蝶形运算等待时间，提升整体编译频率，提高运算性能。评估结果表明，该方法最大工作频率达到了 320 MHz，完成一次 1 024 项多项式乘法运算的时间为 41 μs。

关键词： 后量子密码；现场可编程门阵列；数论变换；多项式乘法；蝶形运算
中图分类号： TP302　　**文献标识码：** A　　**文章编号：** 1002-0802(2022)-12-1613-05

Design and Implementation for Crucial Modules of Lattice-based Cryptography Based on FPGA

HAN Lianbing, FANG Liguo, WANG Song, LIU Hongbo, YANG Mingxu
(No.30 Institute of CETC, Chengdu Sichuan 610041, China)

Abstract: Lattice-based cryptography is an important technique in post-quantum cryptography. In order to improve the computational efficiency of lattice-based cryptography, a hardware implementation of polynomial multiplication in lattice-based cryptography is proposed in this paper. The method uses FPGA (Field Program Gate Array) internal memory to store polynomial coefficients, adopts a ping-pong structure to improve memory parallel read and write speed, and simplifies the computation process by pre-computing and pre-scaling to reduce computational complexity. At the same time, it uses multi-level pipeline to reduce access time and butterfly operation time, enhance the overall compilation frequency, and improve computing performance. The evaluation results indicate that the maximum frequency of the method reaches 320 MHz, and the calculation time of 1 024 polynomial coefficients multiplication is 41 μs.

Keywords: post-quantum cryptography; FPGA; number theoretic transformation; polynomial multiplication; butterfly operation

0　引　言

随着量子计算技术的发展，量子计算机将能在人们可以接受的时间内破解许多目前计算机无法破解的密码，其中就包括目前大部分公钥密码系统所依赖的大整数质数拆分问题和离散对数问题这两大数学难题。

为应对量子计算机为传统密码系统带来的挑战，后量子密码[1]已成为国内外众多学者的重点研究对象。2016年，美国国家标准与技术研究院（Nation

* 收稿日期：2022-08-03；修回日期：2022-11-04　Received date:2022-08-03; Revised date:2022-11-04

Institute of Standards and Technology，NIST）开始了一项针对抗量子密码系统的征集计划，旨在寻找、设计、开发和标准化抗量子密码系统，以便于在未来取代现有的密码系统标准。经过 3 轮的征集提交和筛选，2022 年 7 月 NIST 发布了首批入围标准的 4 个抗量子算法：Crystals-kyber、CRYSTALS-DILITHIUM、FALCON 和 SPHINCS+。这 4 个算法中有 3 个基于格的数学难题，另一个使用了散列函数。由此可见，基于格的密码方案是抗量子计算密码学中的研究热点。基于格的密码算法中的运算大多为线性运算，因此较其他密码系统，基于格的密码系统具有计算速度快[2]、密钥和密文较小等优势[3]。本文对格密码中的关键模块——多项式乘法进行研究，给出了一种多项式乘法的运算方法和硬件实现架构，并在现场可编程门阵列（Field Program Gate Array，FPGA）中进行了实现和评估，为格密码硬件实现提供参考。

1 相关数学基础

1.1 格密码数学基础

线性独立空间上有集合 $\boldsymbol{v}_1,\boldsymbol{v}_2,\cdots,\boldsymbol{v}_n \in \mathbf{R}^n$，格就是这些向量的线性组合，这一过程的表达式为：

$$L(\boldsymbol{v}_1,\boldsymbol{v}_2,\cdots,\boldsymbol{v}_n)=\{a_1\boldsymbol{v}_1+a_2\boldsymbol{v}_2+\cdots+a_n\boldsymbol{v}_n \mid a_1,a_2,\cdots,a_n\in\mathbf{Z}\} \quad (1)$$

式中：系数 a 均在整数域 $\mathbf{Z}$ 中。任意一组可以生成格的线性无关向量都称为格的基，格的维度等于格的基中的向量个数。

目前常用的两个基于格的困难问题是短整数问题（Shortest Integer Problem，SIS）和错误学习问题（Learning With Error，LWE），但基于上述两个问题的加密方案需要的密钥量大、效率低、资源消耗高，无法在实际中运用。因此，Lyubashevsky 等人[4]在 LWE 的基础上提出了环上错误学习（Ring Learning With Errors，RLWE）问题。基于 RLWE 的加密方案在性能上有着显著的优势[5]，这是现在许多格密码算法的理论基础。RLWE 在环 $Z_q[x]=f$ 上进行操作，其中 f 是 n 项的不可约多项式，通常 $f=x^n+1$，其中 n 是 2 的幂，q 为素数。

1.2 环多项式乘法

对于 RLWE 密码算法，其中最为耗时的是环多项式乘法。环多项式乘法有两种实现方式[6-7]，分别为经典乘法和快速数论变换（Number Theoretic Transform，NTT）乘法。

1.2.1 经典乘法

经典乘法先把多项式 a 中的每一项与多项式 b 中的每一项相乘，再把每个多项式相加得到最终结果。如果多项式 a 和 b 的最高次项都为 $n-1$，那么计算复杂度为 $O(n^2)$，需要 n^2 个乘法和 $(n-1)^2$ 个加减法。

1.2.2 NTT 乘法

NTT 是基于快速傅里叶变换（Fast Fourier Transform，FFT）实现的，其将 FFT 中的旋转因子由复数变成了整数。设正整数序列 $x(n)$，其所有元素均小于模数 M，有如下 NTT 变换[8]：

$$X(m)\equiv\sum_{n=0}^{N-1}x(n)a^{mn}(\mathrm{mod}\,M),0\leqslant n\leqslant N-1 \quad (2)$$

$$x(n)\equiv\frac{1}{N}\sum_{n=0}^{N-1}X(m)a^{-mn}(\mathrm{mod}\,M),0\leqslant n\leqslant N-1 \quad (3)$$

式（2）为 NTT 正变换，式（3）为逆 NTT 变换。其中，a 为模 M 的 N 阶本原单位根，满足：

$$a^N\equiv 1(\mathrm{mod}\,M)$$

$$a^{\frac{N}{2}}\equiv -1(\mathrm{mod}\,M)$$

a^{mn} 是 a^{-mn} 在模 M 下的逆元，满足以下性质：

$$a^{mn}\times a^{-mn}\equiv 1(\mathrm{mod}\,M)$$

NTT 运算是一个递推的过程，图 1 给出了 N=8 点的 NTT 运算结构。如图 1 中的椭圆标识所示，NTT 变换后的结果顺序与原输入顺序呈二进制的倒序关系，因此为避免在计算完成后进行顺序变换，通常采用逆序的方式进行运算，运算结构如图 2 所示。图 3 为一次蝶形运算。N 通常可以表示为 2 的幂的形式，$N=2^L$，则 N 点 NTT 运算需要执行 $L\times\frac{N}{2}$ 次蝶形运算，所以 8 点 NTT 需要执行 12 次蝶形运算。

2 多项式乘法 FPGA 实现

2.1 多项式乘法算法

NTT 中的每一次蝶形运算都需要做一次乘法和乘法结果取模运算，因此快速完成乘法和取模运算是提高 NTT 运算效率的关键。本文采用了 Longa 等人[9]提出的适用于模数 $M=k2^p+1$ 的高效取模算法，即 LN 取模算法，再结合 FPGA 内部的高效乘法器来实现 NTT 快速运算。

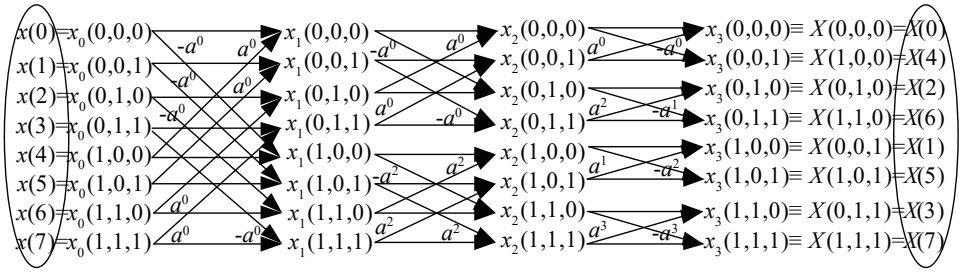

图 1　8 点 NTT 运算结构

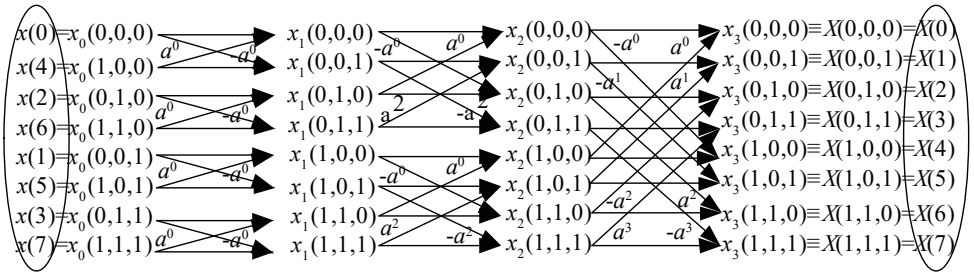

图 2　8 点 NTT 逆序运算结构

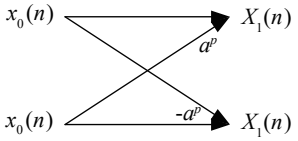

图 3　蝶形运算

LN 取模算法有 K–RED 和 K–RED2x 两种形式[10]，如算法 1 和算法 2 所示。算法 1 适用于对加法结果取模，算法 2 适用于对乘法结果取模。

算法 1：K–RED 算法

输入：x 和模数 M，$x \in (0,(M-1)^2)$，$M=k2^p+1,k<2^p$

输出：$s=xk(\mathrm{mod}\ M)$

1.$x=x_1 2^p+x_0$
2.$x_0=x(\mathrm{mod}\ 2^p)$
3.$x_1=x/2^p$
4.$s'=(kx_0-x_1)$
5.$s=s'(\mathrm{mod}\ M)$

算法 2：K–RED2x 算法

输入：x 和模数 M，$x \in (0,(M-1)^2)$，$M=k2^p+1,k<2^p$

输出：$s=xk^2(\mathrm{mod}\ M)$

1.$x=x_2 2^{2p}+x_1 2^p+x_0$
2.$x_0=x(\mathrm{mod}\ 2^p)$
3.$x_1=x/2^p(\mathrm{mod}\ 2^p)$
4.$x_2=x/2^{2p}$
5.$s'=k^2x_0-kx_1+x_2$
6.$s=s'(\mathrm{mod}\ M)$

NTT 变换和逆 NTT 变换算法如下：

算法 3：NNT 和逆 NTT 变换

输入：x 为多项式输入，预计算 a^i，a^{-i}，i=1,2,3,⋯,p，分别将 d^i 和 a^{-i} 的结果存放在表 t_1 和 t_2 中，$a=g^{\frac{M-1}{2^r}}$，r=1,2,3,⋯,p

输出：$X=NTT(x)$

1. 对多项式倒位序；
2.for(r=1,i=1;i<N;i<<1,r++)do{
3.　len=i<<1;
4.　for(j=0;j<N;j+=len)do{
5.　　gg=1；
6.　　for(k=0;k<i;++k)do{
7.　　u=x[j+k]mod M
8.　　v=x[j+k+i]×gg mod M
9.　　x[j+k]=(u+v)mod M
10.　　x[j+k+i]=(u−v)mod M
11.　　gg=t[r][k]}}}
12. 逆变换时每项乘以 1/N。

算法 3 中 M 为模数，g 为单位根，N 为多项式的项数，如果 N 不满足 2 的整数次幂需要补 0。步骤 6–9 为一次蝶形运算。步骤 11 正变换查 t_1，逆变换查 t_2。

算法 4：基于 LN 的 NTT 运算

输入：x 为多项式输入，预计算 d^i，a^{-i}，i=1,2,3,⋯,p，并分别将 d^i 和 a^{-i} 的结果存放在表 t_1 和 t_2 中，$a=g^{\frac{M-1}{2^r}}$，r=1,2,3,⋯,p

输出：$X=NTT(x)$

1. 对多项式倒位序；
2.for(r=1,i=1;i<N;i<<1,r++)do{
3.　len=i<<1;
4.　for(j=0;j<N;j+=len)do{
5.　　gg=1
6.　　for(k=0;k<i;++k)do{
7.　　u=K–RED(x[j+k])
8.　　w=(x[j+k+i]×gg)
9.　　v=K–$RED2x$(w)
10.　　x[j+k]=(u+v)mod M
11.　　x[j+k+i]=(u−v)mod M
12.　　gg=t[r][k]}}}
13. 逆变换时每项乘以 1/N

算法 4 中的步骤 7 每运算一次相当于在该项上乘以了 k，步骤 8 每运算一次相当于在该项上乘以 k^2，如果对步骤 8 中每个 gg 都乘以 k^{-1}，经过步骤 8 运算后也相当于该项上乘以 k。NTT 每一项的运算次数为算法 4 中步骤 2 总的循环次数，即 $\log_2 N$ 次（N 为多项式的项数）。所以每项都增加了 $k^{\log_2 N}$ 倍，增加部分可以通过预计算的方式消除。

算法 5：多项式乘法运算

输入：x, y 为多项式输入，预计算 $k^{-1}a^i, k^{-1}a^{-i}, i=1,2,3,\cdots,p$，并分别将结果存放在表 t_1 和 t_2 中，预计算 $k^{-(2+\log_2 N)}$，$k^{-(4+\log_2 N)}N^{-1}$

输出：$z=x\times y$

1.for(i=1;i<N;i++)do{
　x_i'=K–$RED2x$($x_i\times k^{-(2+\log_2 N)}$)
　y_i'=K–$RED2x$($y_i\times k^{-(2+\log_2 N)}$)}
2.X=NTT(x')
3.Y=NTT(y')
4.for(i=0;i<N;i++)do{
　Z_i=K–$RED2x$($X_i\times Y_i$)}
5.z'=$INTT$(Z)
6.for(i=0;i<N;i++)do{
　z_i=K–$RED2x$($z_i'\times k^{-(4+\log_2 N)}N^{-1}$)}
7.return z

算法 5 中的步骤 1 是为了消除 NNT 运算时每项增加的 $k^{\log_2 N}$ 倍而做的预处理。步骤 6 中的 $k^{-(4+\log_2 N)}N^{-1}$ 则是为了消除 INNT 运算后扩大的 $k^{\log_2 N}$ 倍，并完成 INNT 运算后的除法运算。

2.2 多项式乘法 FPGA 实现

多项式乘法的 FPGA 实现如图 4 所示。

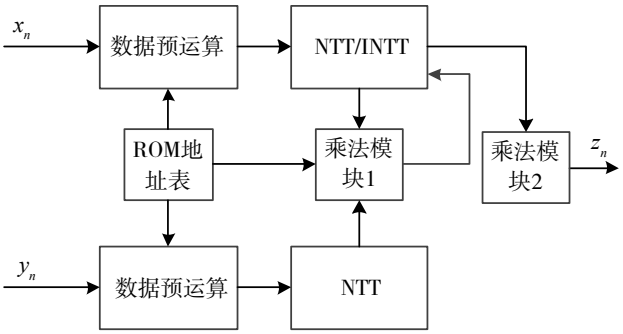

图 4 多项式乘法 FPGA 实现

2.2.1 数据预运算模块

数据预运算模块用于对多项式数据进行预处理，同时完成对多项式的倒位序。ROM 地址表内存放预计算好的位序映射表。根据 ROM 读出的地址读取原始序列，预运算后写入 NTT 模块内的存储器中。

2.2.2 NTT 模块

图 5 为 NTT 模块的实现。

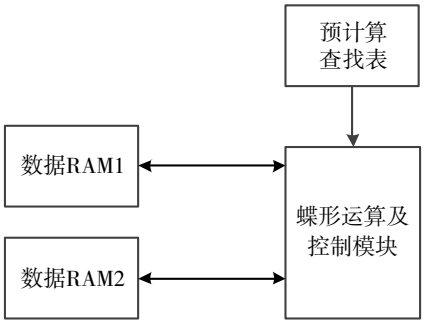

图 5 NTT 实现

数据 RAM1 和数据 RAM2 为多项式系数存储区，由于 FPGA 内部实现的 RAM 通常只有 2 路通道，不能满足蝶形运算同时对 RAM 的 2 次读和写操作。为了解决这个问题，本文设计了 RAM1 和 RAM2 两个数据存储区。当 RAM1 作为数据读取区时，蝶形运算的结果写入 RAM2 区，当 RAM2 作为数据读取区时，蝶形运算的结果写入 RAM1 区，由内部控制模块乒乓切换两个数据区的读写模式。

预计算查找表用于存放蝶形运算所需的预计算数据，该数据可以预先计算好后固化在存储区内部，不占用 NTT 的计算时间。预计算的数据包括 $k^{-1}a^i$，$k^{-1}a^{-i}$，$k^{-(2+\log_2 N)}$ 和 $k^{-(4+\log_2 N)}N^{-1}$。

蝶形运算及控制模块通过状态机控制 NTT 的 3 层循环，以及每次蝶形数据和预计算数据的读取，调用 K–RED 和 K–RED2x 完成运算。因蝶形运算下一次的输入数据不会用到上一次的结果，所以蝶形运算可实现流水操作，从而提升运算性能。

2.2.3 乘法模块

乘法模块用于完成算法 5 的步骤 4 和步骤 6。其中乘法模块 1 用于完成两个多项式转换到 NTT 域后各项的相乘，并根据 ROM 地址表内存放的地址读取多项式的值相乘，将结果存放在 NNT 的 RAM 内，用于逆 NNT 运算。乘法模块 2 用于完成逆 NTT 运算结果除 N 运算和消除 K–RED2x 运算产生的 k^2 缩放。由于 k 通常都不大，K–RED2x 内部的 k^2x_0 和 kx_1 可以转换为由移位和加法实现，不需要乘法运算。

3 实现结果评估

为了便于结果评估，本文选用模数 M=12 289，并设多项式的项数 N=1 024，测试平台采用 Xilinx 公司的 XC7K325T 型号 FPGA。

Kuo 等人 [11] 运用了适合于硬件实现的模约减方法，但使用了较多的加法器，编译频率不高。Oder 等人 [12] 使用的模约减模块包含延时较大的关键路径，且存取效率不高，编译频率也较低。本文的蝶形运算模块及 LN 模运算模块均采用流水线实现，所以实现频率较高，达到了 320 MHz。由于采用流水实现，预算模块和 NTT 运算可以并行执行，且 NTT 内部的蝶形运算模块同样为流水结构，从而大大提高了运算性能。表 1 为本文多项式乘法硬件实现与现有一些硬件实现的比较结果。其中，查找表（Look-Up-Table，LUT）、寄存器（REGister，REG）、块存储器（Block RAM，BRAM）和乘法器（Digital Signal Processing，DSP）分别为 FPGA 内硬件资源。

表 1　多项式乘法硬件评估结果

方　案	器　件	LUT/ 个	REG/ 个	BRAM/DSP/ 个	频率 /MHz	时延 / μs
Kuo 等人方案 [11]	A7	2 382	1 381	10/8	150	52
Oder 等人方案 [12]	A7	432	278	4/2	125	1008
陈朝晖等人方案 [10]	A7	389	346	3.5/1	307	57
本文方案	K7	1 114	1 358	4.5/2	320	41

4　结　语

本文提出的多项式乘法硬件实现方法，采用不完全模约减的方式取模，大大减少了取模的时间。同时采用了乒乓切换、流水技术和双 NTT 模块架构，一方面提高了存储器读写带宽，另一方面减少了运算过程中的等待时间，从而提升了运算性能。此外，由于采用了流水设计，编译主频也较高，达到了 320 MHz。因此，本设计无论是在资源占用方面还是在处理性能方面都具有一定的优势，对基于格的后量子密码的硬件实现具有一定的参考意义。

参考文献：

[1] CHEN L,JORDAN S,LIU Y,et al.Report on post quantum cryptography[R/OL].(2016-04-28)[2022-04-15].http://dx.doi.org/10.6028/NIST.IR.8105.

[2] NEJATOLLAHI H,DUTT N,RAY S,et al.Post-quantum lattice-based cryptography implementations:A survey[J]. ACM Computing Surveys,2019,51(6):1-41.

[3] BOS J,COSTELLO C,DUCAS L,et al.Frodo:Take off the ring!practical,quantum-secure key exchange from LWE[C]// Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security,2016:1006-1018.

[4] LYUBASHEVSKY V,PEIKERT C,REGEV O.On ideal lattices and learning with errors over rings[C]// Annual International Conference on the Theory and Applications of Cryptographic Techniques,2010:1-23.

[5] 蒋亚丽 . 基于格的密码方案的研究与设计 [D]. 济南 : 山东大学 ,2011.

[6] 芮康康 , 王成华 , 范赛龙 , 等 . 一种高性能 R-LWE 格加密算法的电路结构及其 FPGA 实现 [J]. 数据采集与处理 ,2019,34(4):689-696.

[7] 施佺 , 韩赛飞 , 黄新明 , 等 . 面向全同态加密的有限域 FFT 算法 FPGA 设计 [J]. 电子与信息学报 ,2018,40(1):57-62.

[8] 谢星 , 孙玲 , 黄新明 , 等 . 基于 FPGA 的有限域 NTT 算法设计与实现 [J]. 现代电子技术 ,2020,43(9):79-82.

[9] LONGA P,NAEHRIG M.Speeding up the number theoretic transform for faster ideal lattice-based cryptography[C]//Proceedings of International Conference on Cryptology and Network Security,2016:124-139.

[10] 陈朝晖 , 马原 , 荆继武 . 格密码关键运算模块的硬件实现优化与评估 [J]. 北京大学学报 (自然科学版),2021,57(4):595-604.

[11] KUO P C,LI W D,CHEN Y W,et al.High performance post-quantum key exchange on FPGAs[EB/OL].(2017-06-12)[2020-02-21].https://eprint.iacr.org/ 2017/690.pdf.

[12] ODER T,GÜNEYSU T.Implementing the NewHope-simple key exchange on low-cost FPGAs[C]// International Conference on Cryptology and Information Security in Latin America,2019:128-142.

作者简介：

韩炼冰（1984—），男，学士，高级工程师，主要研究方向为信息安全、通信安全技术；

房利国（1977—），男，硕士，高级工程师，主要研究方向为信息安全、通信安全技术、计算机应用；

王　松（1985—），男，学士，高级工程师，主要研究方向为信息安全、通信安全技术；

刘鸿博（1981—），女，学士，高级工程师，主要研究方向为信息安全、通信安全技术；

杨敏旭（1994—），女，学士，工程师，主要研究方向为信息安全、通信安全技术。