文章编号:1671-4598(2015)01-0207-04　　中图分类号:TP309.7　　文献标识码:A

基于多线程的 AES-ECB 改进设计与性能分析

周文刚[1]，赵　宇[1]，朱　海[2]
(1. 周口师范学院 计算机科学与技术学院，河南 周口　466001；
2. 西安交通大学 计算机科学与技术系，西安　710049)

摘要：为了满足高速即时通信对密码运算的性能要求，在研究 AES-ECB 典型设计的基础上，提出了一种基于多线程技术的 AES-ECB 改进设计方案；通过对算法中间迭代过程进行分割，并将其例化为多个子线程，改进了算法运行流程，实现了子线程间的“类流水”操作；实例性能分析表明，相比于单线程，多线程性能提高了约 0.46～4.27 倍；当线程数为 12 时，算法处理性能最低为 520 Mbps，适用于对加解密速度要求高的应用场合。

关键词：密码算法；高级加密标准；电码本；多线程

Improved Design and Performance analysis of AES-ECB Based on Multithread Technology

Zhou Wengang[1]，Zhao Yu[1]，Zhu Hai[2]
(1. School of Computer Science and Technology，Zhoukou Normal University，Zhoukou　466001，China；
2. Department of Computer Science and Technology，Xi'an Jiaotong University，Xi'an　710049，China)

Abstract：To satisfy needs of the cryptographic performance in high speed real-time communications，after researching the typical design of AES-ECB，an improved design of AES-ECB based on multithread technology is proposed. By dividing the intermediate iterative process into some sub threads，the running flow of AES-ECB is improved，and sub threads can be organized like pipelining. The performance analysis results show that the multithread design can increase about 0.46 to 4.27 times than the single-thread design in performance，and when the sub thread number is 12，the speed can achieve 520 Mbps at least，which is suitable for applications with high requirements on the en/decryption speed.

Keywords：cryptography algorithm；advanced encryption standard；electronic codebook；multithread

0　引言

高级加密标准（advanced encryption standard，AES）又称 Rijndael 算法[1]，是美国政府采用的一种信息加密标准。该算法作为 DES 算法[2]的替换物，经过不同强度的安全性分析[3-4]，至今未发现明显漏洞，在不同领域得到了广泛应用[5-6]。自 2001 年美国国家标准与技术研究院（National Institute of Standards and Technology，NIST）发布 AES 算法至今，该算法已成为分组密码算法中最著名的标准之一。

电码本（Electronic Codebook，ECB）模式是分组密码算法的一种最基本的工作模式[7]，与密码块链接（Cipher Block Chaining，CBC）、密码反馈（Cipher Feedback，CFB）、输出反馈（Output Feedback，OFB）等模式相比，ECB 模式具有操作过程简洁，易于软硬件的并行处理实现等优点，且由于其明密文分组的独立性，ECB 模式还具有较好地防止误差传播的特性，特别适合高速即时通信对密码操作的要求。

为了提高 AES-ECB 加解密处理性能，满足高速即时通信对密码算法速度高的应用要求，在研究 AES-ECB 典型设计的基础上，本文基于多线程技术给出了一种 AES-ECB 改进设计方案。算法性能分析表明，采用多线程的 AES-ECB 加密算法效率有显著提高。

1　基础知识

1.1　AES 算法

AES 算法采用 Square 结构，支持的密钥长度有 128、192、256 位三种，分组长度为 128 位。该算法每一轮都使用代替和混合并行地处理整个数据分组，其结构由 4 个不同的环节组成：1）字节代替，将输入或中间态的每一个字节通过 S 盒查表映射为另一个字节；2）行移位，实现基于行的循环移位操作；3）列混合，利用域 $GF(2^8)$ 上算法特性进行逐列混合；4）圈密钥加，将当前分组数据和扩展密钥的一部分进行异或运算。

AES 算法加密过程为：首先，待加密的明文按 128 位分组后，构成一个 4×4 字节矩阵，并与初始圈子密钥异或；然后，进行 Nr 圈迭代运算，除最后一圈（第 Nr 圈）省略列混合变换外，每圈均包含字节代替、行移位、列混合和圈密钥加；最后，第 Nr 圈迭代后的结果仍是一个 4×4 字节矩阵，将其恢复成 128 位数据即为对应的密文分组。

收稿日期:2014-06-09；　**修回日期**:2014-07-08；
基金项目:国家自然科学基金(61103143)；河南省科技厅基础与前沿技术研究计划项目(132300410276)；河南省教育厅自然科学研究计划项目(2010B520036)。

作者简介:周文刚(1972-)，男，河南人周口人，硕士，副教授，主要从事智能算法分析与设计等方向的研究。
赵　宇(1973-)，男，河南周口人，硕士，副教授，主要从事无线传感器网络方向的研究。
朱　海(1978-)，男，河南西峡人，博士后，副教授，主要从事云计算方向的研究。

AES 算法解密流程与其加密流程类似，基本运算中除圈密钥加不变外，其余字节代替、行移位、列混合都要分别用其逆变换代替。

1.2 ECB 模式

一个分组密码以固定长度的 n 位分组来加密明文（对于 AES 算法，$n=128$）。若消息长度超出了 n 位，最直接的方法是将消息分成 n 位的分组，然后分别对每一分组独立进行加密或解密操作，这种处理方法即 ECB 模式，其描述如算法 1 所示。其中，E_K 指分组密码 E 中以密钥 K 为参量的加密算法，E_K^{-1} 表示对应解密算法。

算法 1：ECB 模式[7]

输入：n 位的明文分组 $x_0,\ldots,x_t$；k 位密钥 K

输出：n 位的密文分组 $c_0,\ldots,c_t$；解密恢复明文

1）加密：对于 $0\leqslant j\leqslant t$，$c_j\leftarrow E_K(x_j)$

2）解密：对于 $0\leqslant j\leqslant t$，$x_j\leftarrow E_K^{-1}(c_j)$

1.3 多线程技术

在计算机系统中，一个程序开始运行就对应一个进程，且每个进程可包含一个或多个线程。线程是程序中的一个执行流，是程序执行的最小单位，每个线程都有自己的专用寄存器（栈指针、程序计数器等），但代码区可以共享。多线程是指一个进程中可以同时包含多个执行流，且每个执行流分别用来完成各自的分配任务。随着技术的进步，多线程作为一种并发型、多任务的工作机制，具有提高程序响应、改善程序结构、使多 CPU 并行更有效等优点[8]。

2 算法设计

2.1 AES-ECB 典型设计

考虑到 AES 算法加解密流程具有“相似性”[1]，本文涉及 AES 算法时均以其加密流程为例。AES-ECB 典型设计原理如图 1 所示，首先根据 AES 算法加密流程描述，将圈密钥加、字节代替、行移位、列混合设计为独立的函数模块，分别记为 AddRoundKey、SubBytes、ShiftRows、MixColumns；然后通过 Nr 圈迭代调用相应模块，形成完整的 AES 加密函数逻辑；最后，在主线程完成明文分组、密钥扩展的基础上，再循环调用 AES 加密函数逻辑实现对全部明文分组的加密运算。

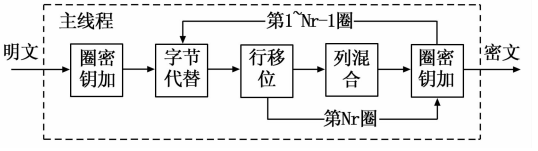

图 1 AES-ECB 典型设计

与图 1 对应的 AES-ECB 加密典型流程如算法 2 所示，其优点是：算法设计逻辑结构清晰，只涉及对环节模块的循环迭代调用操作，算法实现相对简单；其缺点是：由于所有操作都由主线程串行处理，每一个明文分组都要等待 Nr 圈迭代处理完成后再进行下一个明文分组，对于 ECB 加密模式其处理效率较低，不适合对加解密速度要求高的应用场景。

算法 2：AES-ECB 加密典型流程

输入：n 位的明文分组 $x_0,\ldots,x_t$；扩展密钥指针 W

输出：n 位的密文分组 $c_0,\ldots,c_t$

For $j=0$ to t

byte state [4, 4] $=x_j$；

AddRoundKey (state, W)；

For r = 1 to Nr

SubBytes (state)；

ShiftRows (state)；

if (r<Nr) MixColumns (state)；

AddRoundKey (state, W+4×r)；

c_j =state；

2.2 AES-ECB 改进设计

通过分析算法 2 可以看出，在迭代调用 4 个环节处理一个明文分组时，所有环节模块在时空上仅服务于该明文分组。根据“空间换时间”的思想，结合多线程技术的优点，可以将相同的中间迭代过程进行分割，并分别例化为多个子线程，主线程只负责明文的初始圈密钥加操作，以及对子线程进行参数管理和运行控制，其原理如图 2 所示。通过主线程的有机调度和控制，实现子线程间的“类流水”操作，不同的子线程在时空上可同时服务于多个明文分组，从而提高算法处理效率。

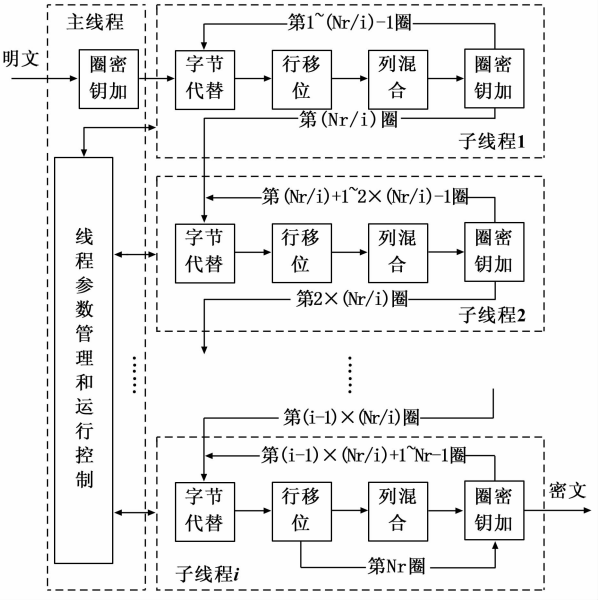

图 2 AES-ECB 改进设计

为了使多个子线程的运行时间尽可能相近，在对中间迭代过程进行分割时，为每个子线程分配的子迭代圈数相同。令 i 为总圈数 Nr 的约数，则对于子线程 $s(1\leqslant s\leqslant i)$，其子迭代圈 r 为：

$$(s-1)\times\frac{Nr}{i}+1\leqslant r\leqslant s\times\frac{Nr}{i}$$

当 $i=1$ 时，AES-ECB 改进设计等价于 AES-ECB 典型设计；当 $i=Nr$ 时，Nr 个子线程分别负责 Nr 个中间迭代过程，AES-ECB 改进设计线程空间开销最大。与图 2 对应的 AES-ECB 加密改进流程如算法 3 所示。

算法 3：AES-ECB 加密改进流程

输入：n 位的明文分组 $x_0,\ldots,x_t$；扩展密钥指针 W

输出：n 位的密文分组 $c_0,\ldots,c_t$

For $j=0$ to t

byte $\text{state}_{\text{jmodi}}$ [4, 4] $=x_j$；

AddRoundKey ($\text{state}_{\text{jmodi}}$, W)；

对于子线程 $s(1\leqslant s\leqslant i)$，执行下列操作：

For $r=(s-1)\times\frac{Nr}{i}+1$ to $s\times\frac{Nr}{i}$

SubBytes（$state_{jmodi}$）；

ShiftRows（$state_{jmodi}$）；

if（r＜Nr）MixColumns（$state_{jmodi}$）；

AddRoundKey（$state_{jmodi}$，W＋4×r）；

c_j ＝$state_{jmodi}$；

3　算法实现与性能分析

3.1　算法程序实现

在本文算法实现中，我们以 C 语言作为编程语言，分别实现了 AddRoundKey、SubBytes、ShiftRows、MixColumns 4 个函数。子线程实现原理是：为每个子线程 $s(1\leqslant s\leqslant i)$ 分配一个 4×4 字节型状态矩阵 $state_S$ 、线程使能控制信号 enb_S ，以及整数型子迭代圈起止值；当 enb_S 有效时，调用上述函数对 $state_S$ 进行子迭代处理，完成后置 enb_S 失效；在线程参数管理和运行控制模块作用下，启动多个子线程并实现多线程间的“类流水”操作，全部明文分组处理完成后结束所有子线程。

线程参数管理和运行控制流程如图 3 所示。当程序启动时，主线程启动 i 个子线程，为每个子线程分配子迭代圈起止值，并置每个 enb_S 失效；对当前明文分组进行初始圈密钥加后，将其值赋给 $state_1$ ，同时置 enb_1 有效，此时子线程 1 运行，定时检查 enb_1 是否失效；若有效，则忽略此次检查，并开始下一次定时检查，直到检查到 enb_1 失效，则判断是否还有明文分组；若还有明文分组，则对下一个明文分组进行初始圈密钥加，并将其值赋给 $state_1$ ，同时置 enb_1 有效，否则定时检查直到每个子线程 enb_S 失效时结束程序；在检查到 enb_1 失效的同时，将 $state_1$ 赋给 $state_2$ ，同时置 enb_2 有效，此时子线程 2 运行，定时检查 enb_2 是否失效；若有效，则忽略此次检查，并开始下一次定时检查，直到检查到 enb_2 失效；依此理，分别对线程 s 进行参数管理和运行控制，直到定时检查到 enb_i 失效，保存当前密文分组。

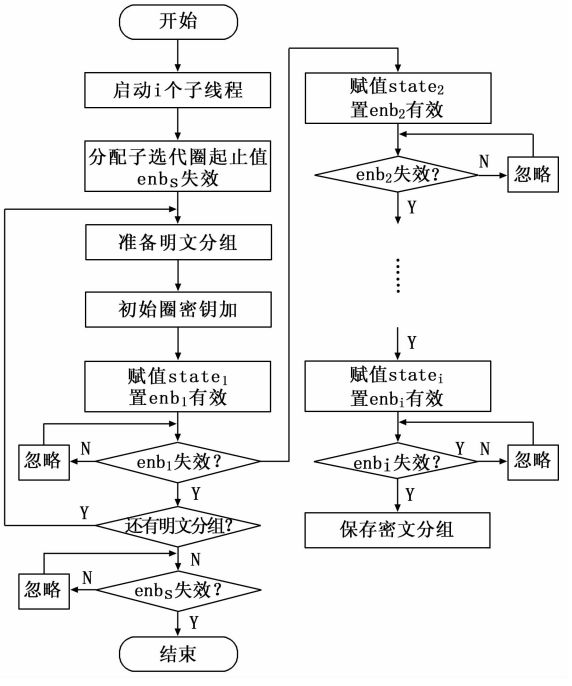

图 3　线程参数管理和运行控制流程

3.2　实现性能分析

依据上述思路实现的 AES _ ECB 算法接口为：

```
int AES_ECB(
char *   key,   // 密钥数据指针
int   keylen,   // 密钥数据字节长度
char *   inBlk,   // 输入数据指针
int   inBlklen,   // 明文数据字节长度
char   edflag,   // 加解密标志,0：加密，1：解密
int   threadnum,   // 子线程数
char *   outBlk,   // 输出数据指针
int *   outBlklen   // 输出数据字节长度指针
);
```

本文以密钥长度为 192 位（keylen＝24）的 AES-ECB 加密流程为对象，通过配置 threadnum 为 1、2、3、4、6、12，分析了算法的加密效率。性能分析包括两部分：1）以固定量的明文数据作为算法输入，通过配置 threadnum 为不同值，比较程序多次运行的加密效率；2）分别固定 threadnum，以不同量的明文数据作为算法输入，分别比较程序运行的加密效率。试验结果如图 4 所示。运行平台为：中标麒麟 32 位操作系统（V3.2.2）、Intel 酷睿 I5 处理器四核、主频 3.2 GHz、内存 2G。

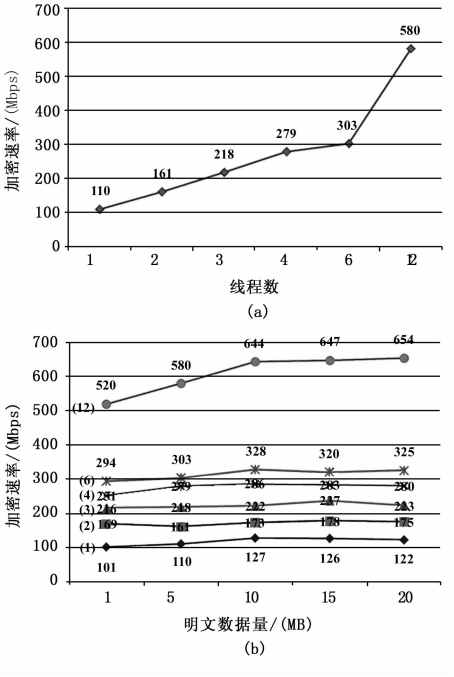

图 4　AES-ECB 算法性能分析

由图 4（a）可知，随着子线程数的增加，算法处理性能呈上升趋势。相比于 1 线程，2、3、4、6、12 线程效率分别提高了约 0.46 倍、0.98 倍、1.53 倍、1.75 倍、4.27 倍。由图 4（b）可知，从纵向上来看，可得出与图 4（a）类似的结论；从横向上来看，对于每一线程数，随着明文数据量的增加，算法处理效率经历上升阶段后趋于平稳，基本保持在某一区间内。其中，当线程数为 12 时，算法处理性能最低为 520 Mbps，最高为 654 Mbps，可满足对加解密速度要求高的应用场合。

4 结论

AES作为目前最流行的分组密码算法之一，在不同领域得到了广泛应用。为了满足即时通信中实时、高效软加密的应用需求，通过引入多线程并发机制，本文提出了一种AES-ECB改进设计，相比于AES-ECB典型设计，其性能提高效果明显。作为下一步，我们将参考已有成果[9-10]，结合课题实际需要，研究嵌入式环境下AES改进算法设计，进而满足手持式终端对密码算法的复杂应用要求。

参考文献：

[1] Daemen J，Rijmen V. The design of rijndael：AES-the advanced encryption [M]. Berlin：Springer-Verlag Press，2002.

[2] National Institute of Standards and Technology (NIST). Federal Information Processing Standards Publication 81 (FIPS PUB 81)：DES Modes of Operation [EB/OL]. (1980-12-02). http：//www.it1.nist.gov/fipspubs/fip8.htm

[3] 金晨辉，孙 莹. AES密码算法S盒线性冗余研究 [J]. 电子学报，2004 (4)：639-641.

[4] 胡志华，覃中平. 一种新的8轮AES_128不可能差分分析 [J]. 小型微型计算机系统，2013，34 (9)：2111-2115.

[5] 王海科，范伊红，廉飞宇，等. AES加密算法在不停车收费系统中的应用 [J]. 计算机测量与控制，2006，14 (1)：95-97.

[6] 向 涛，余晨韵，屈晋宇，等. 基于改进AES加密算法的DICOM医学图像安全性研究 [J]. 电子学报，2012，40 (2)：406-411.

[7] Menezes A J，Oorschot P C，Vanstone S A. 应用密码学手册 [M]. 胡 磊，王 鹏，等译. 北京：电子工业出版社，2005.

[8] 翟一鸣，任满杰，孔繁茹，等. 计算机操作系统 [M]. 北京：清华大学出版社，2012.

[9] 封 斌，齐德昱. AES快速算法的扩展指令集实现 [J]. 华南理工大学学报 (自然科学版)，2012，40 (6)：97-102.

[10] 陈 亮. 基于嵌入式CPU的数据加解密子系统的设计研究 [D]. 杭州：浙江大学，2013.

(上接第206页)

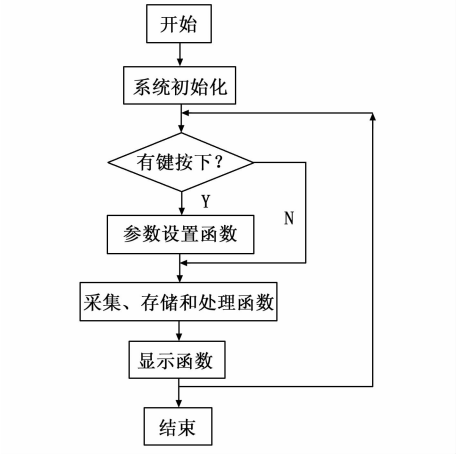

图6 NiosII软核处理器软件设计流程图

数信号发生器可以产生频率范围为1 Hz～30 MHz、幅度范围为2 mVpp ～2 Vpp的正弦波、方波和三角波信号。通过按键设置放大器的各项参数，在放大器的输入端加入测试信号，使用GDS-1102A型数字存储示波器对放大器输出的各项参数进行测试。部分参数测量结果如表1所示。

测试结果表明，该放大器实现了频率范围在1 Hz～16 MHz变化，当输入电压变化范围10 mV～1 V时，输出电压保持在0.1～5 V，增益控制范围达0.1～500，具有参数精度高、可靠性强等特点。

表1 放大器部分参数测量结果(电压为V_{PP})

被测信号	1 Hz 10 mV	1 kHz 0.1 V	1 MHz 0.5 V	10 MHz 0.8 V	16 MHz 1 V
1.0～1.5 V	√	√	√	√	√
输出信号/V	1.143	1.233	1.441	1.426	1.482
增益/dB	24.633	23.440	10.121	5.972	3.847

放大器的误差主要来源于：系统的外部误差，如被测量信号与放大器的阻抗不匹配；系统的内部误差包含A/D在进行数据采集时，可能会采集到毛刺的电压，使采集到的峰峰值存在误差、D/A的转换精度不够引起的误差和频率测量方法引起的误差。因此智能放大器系统还需要进一步改进测频方法和提高D/A的转换精度、A/D的采样速率来提高放大器的精度。

5 结束语

基于FPGA的智能放大器，可以将电路的数字逻辑部分和控制部分（NiosII软核处理器）置于一块FPGA芯片内，大大降低了电路的复杂程度，具有集成度高，可靠性高，功耗低、开发周期短、软硬件升级方便等众多优点，体现了采用FPGA技术方案的优越性。该放大器已应用于植物种苗磁电场复合诱导繁育控制系统项目中，运行测试数据证明：该放大器具有参数精度高、可靠性强等特点，在仪器仪表电路中具有广阔的应用前景。

参考文献：

[1] 鲜 果，龚晓峰. 基于FPGA的新型虚拟逻辑分析仪的设计 [J]. 电子技术应用，2011，37 (12)：87-89.

[2] 苗康乐，杨日杰，杨成伟. 高精度声呐信号预处理系统的改进与实现 [J]. 仪器仪表学报，2011，32 (12)：2720-2724.

[3] 谷 銮，徐贵力，王友仁. FPGA动态可重构理论及其研究进展 [J]. 计算机测量与控制，2007，15 (11)：1415-1418.

[4] 张雅珍. 基于FPGA和ADS830的数字示波器设计 [J]. 电子测量技术，2009，32 (10)：121-124.

[5] 邓耀华，吴黎明，张力锴，李业华. 基于FPGA的双DDS任意波发生器设计与杂散噪声抑制方法 [J]. 仪器仪表学报，2009，30 (11)：2256-2260.

[6] 刘俊斌，吴松林，周卫星. 基于FPGA实现的高速等效采集系统 [J]. 电子技术应用，2011，37 (10)：84-86.

[7] 何 琼，陈 铁，程 鑫. 基于FPGA的DMA方式高速数据采集系统设计 [J]. 电子技术应用，2011，37 (12)：40-43.

[8] 乔永征，梁志毅，朱毖徽. 基于OV7620和FPGA的图像采集系统设计 [J]. 计算机测量与控制，2009，17 (9)：1857-1859.

[9] 倪明辉，周 军，杨 庚. USB在FPGA控制的高速数据采集系统中的应用 [J]. 计算机测量与控制，2006，14 (2)：268-271.