基于 CPLD/FPGA 的 AES 算法混合流水实现

彭艮鹏　　刘常澍　　李志华
（天津大学电子信息工程学院　天津　300072）

摘　要： 在加解密算法的硬件实现中，使用流水线结构可以显著地提高加密解密速度，但是由于这类结构并不适合于大多数的反馈模式，因而此类结构在当前密码学中的应用较少。为此，该文采用一种补偿手段，基于交叉 CBC（Interleaved Cipher Block Chaining）模式，以混合流水结构成功地实现了 AES (Advanced Encryption Standard)的算法。该方案允许并行处理 4 个数据块（称为一次加密或解密），同时两次加密或解密之间还可实现部分并行。该方案在 EP20k300EBC652-1（Ateral 公司产品）上已得到成功验证。

关键词： AES，流水线，混合流水，硬件实现，加密，解密
中图分类号： TN918.4　　**文献标识码：** A　　**文章编号：** 1009-5896(2005)01-0155-03

The Hybrid Pipelining Implementation of AES in the Feedback Mode Based on CPLD/FPGA

Peng Gen-peng　　Liu Chang-shu　　Li Zhi-hua
(*School of Electronic and Information Engineering, Tianjin University, Tianjin* 300072, *China*)

Abstract　Although using pipelining structure in the hardware implementation can generally provide higher throughput, the application of this structure in current cryptography is limited, because they are not suitable for most common feedback modes. This paper puts forward a design of the hybrid pipelining architecture of AES. By including in the AES standard interleaved modes of operation, the design successfully implements the algorithm, which operates in the CBC mode. In this design, four data blocks can be dealt with in parallel (called one-encryption or one-decryption), and at the same time two encryptions or decryptions can be partially overlapped. The design has been implemented on EP20k300EBC652-1 device (Ateral).

Key words　AES, Pipelining, Hybrid pipelining, Hardware implementation, Encryption, Decryption

1　引言

AES 算法是一种分组长度为 128b,而密钥长度为 128b/192b/256b 的分组密码算法。根据密钥长度的不同，数据加密和解密的循环轮数也不同（见表 1）。

表 1　密钥长度与加密轮数的关系

AES 类型	密钥长度 N_k	分组大小 N_b	加密轮数 N_r
AES-128	4 字	4 字	10
AES-192	6 字	4 字	12
AES-256	8 字	4 字	14

加解密所需的循环密钥可以从密钥产生方案所生成的密码密钥中获得。密钥产生方案由两部分构成：密钥扩展和循环密钥选择[1]。该密钥扩展方案支持 N_k=4, 6, 8　3 种密钥扩展方案，并且 1 个时钟即可完成 1 个密钥字的计算。当前密钥的计算与前一数据加密并行进行，因此并不占用额外时钟。密钥计算完成后存储在 RAM 中。这里仅给出密钥扩展方案结构图（见图 1）。

AES 将输入分组作为一个 4×4 的字节矩阵（称为状态矩阵），然后对其施以不同的变换。

AES 的加密算法由 3 个部分组成[1]：(1) 与初始循环子密钥的“异或”；(2) N_{r-1} 轮循环加解密；(3) 最后一轮加解密。其解密算法为其逆过程。AES 算法的流程图见图 2。

字节变换是一种非线性面向字节（中间状态矩阵的每一个字节）的变换。可以通过构造可逆 S 盒实现该变换。而其逆变换则通过构造逆 S 盒加以实现。

行变换是将状态矩阵的各行进行循环移位，0 行不移，第 1 行移 1 个字节，第 2 行移 2 个字节，第 3 行移 3 个字节。逆行变换为上述过程的逆过程。

列变换是将中间状态矩阵各列作如下变换（式(1)为列变换，式(2)为逆列变换）：

2003-06-17 收到，2003-12-11 改回

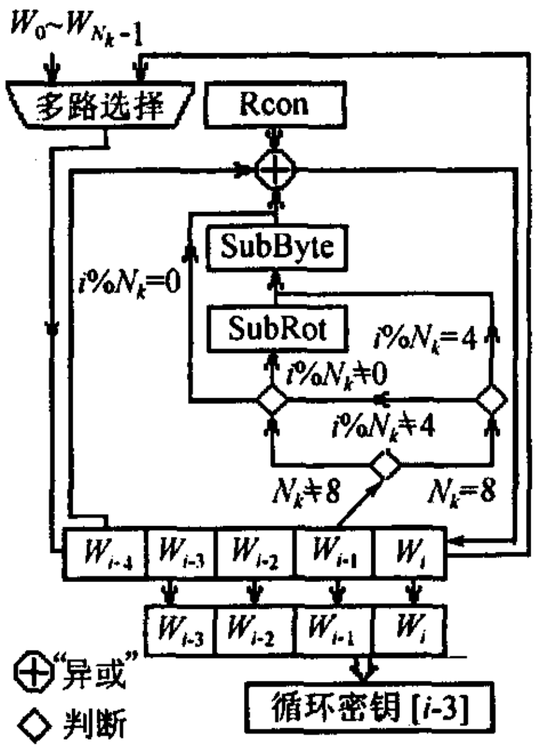

图 1 密钥扩展方案实现的结构图

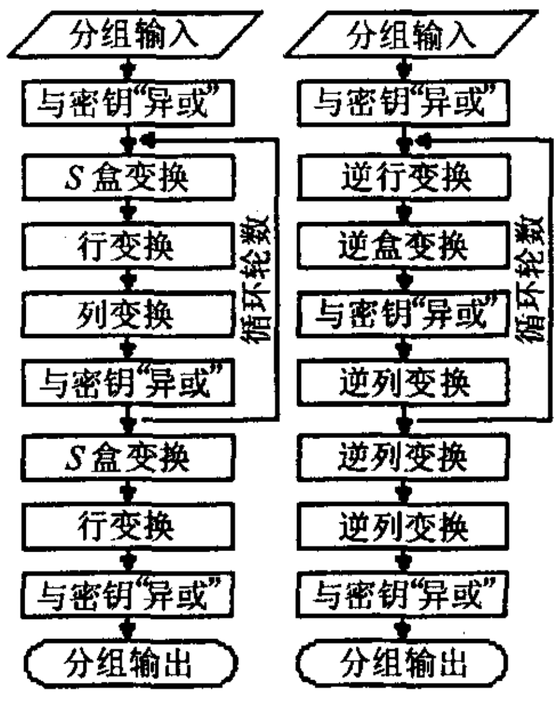

图 2 AES 加解密流程

$$\begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 01 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} \tag{1}$$

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 0E & 0B & 0D & 09 \\ 09 & 0E & 0B & 0D \\ 0D & 09 & 0E & 0B \\ 0B & 0D & 09 & 0E \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} \tag{2}$$

上述矩阵乘法运算为有限域上的运算，其中的乘法运算由于有一个因子为常数，因而可以转化为“异或”运算[2]；而矩阵中的加法运算其实亦为“异或”运算，因而矩阵乘法运算可以通过一些简单的“异或”门加以实现。图 3 所示为逆列变换中单列变换的实现电路图(列变换实现与之类似)，而一轮变换中为提高速度共需要 4 个这样的相同结构来并行地实现状态矩阵总共 4 列的列变换。

与密钥“异或”变换指的是将数据块与循环子密钥按位“异或”；其逆变换亦为“异或”。

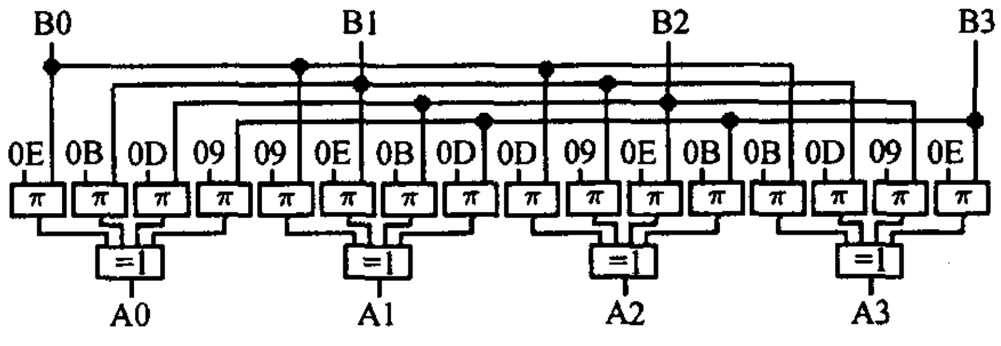

图 3 逆混合列变换的实现

2 分组密码的工作模式

为了提高算法的抗攻击能力，分组密码一般工作在某一特定的工作模式下。从硬件实现的角度看，这些工作模式有两类，即反馈模式和非反馈模式。在这两种模式中，明文由需要处理的连续数据块组成。在反馈模式中，各独立数据块只能进行串行处理。而在非反馈模式中，数据块可以做成流水或并行处理，这样便可获得更高的吞吐率。但对反馈模式而言，对各个明文数据块进行并行加密是不可能的，流水不仅无法获得性能的改善，而且可能会导致实现算法所需资源的大量增加。

上述限制可通过交叉处理模式加以克服[3]。在这种模式下，首先将明文分成若干个明文块，每个明文块又由 N 个数据块组成，使用传统的反馈模式和 N 个不同的初始向量 **IV** 对这些明文块进行加密。当一个明文（N 个数据块）加密完成后，即可对下一个明文块（接着的 N 个数据块）的加密。例如，交叉处理 CBC 模式也可被定义为

$$C_i=\mathrm{AES}(M_i\oplus \mathrm{IV}_i)\mid i=1,2,\cdots,N;\quad C_i=\mathrm{AES}(M_i\oplus C_{i-N})\mid i>N$$

采用交叉 CBC 模式可保证其速度与采用非反馈模式时的速度相当，而其安全性则不亚于采用反馈模式时的安全性。

3 AES 算法实现结构[4]

AES 算法实现结构有如下几种：基本结构、内部流水、循环轮展开结构、外部流水、混合流水。本文通过将循环轮逻辑地划分为 m 个部分（尽可能地使各部分地延时大致相同），再用寄存器将所划分的 m 个部分分割开来，从而在循环轮内各个划分就形成了内部流水段；然后将 k 个循环轮作为单一的组合逻辑加以实现，再在每个 k 循环轮展开之间加入寄存器而形成外部流水线结构。这样就可形成一个 $k\times m$ 个流水段的流水线。

4 AES 混合流水结构实现方案

该方案采用混合流水结构，将 $k=2$ 个循环轮展开，其中每轮被分成 $m=2$ 个流水段，这样就形成一个 2×2=4 个流水段的流水线。采用交叉处理 CBC 模式(N=4)，就可并行处理 4 个数据块。取 $k=2$，循环轮数(Rounds)正好为整数。见表 2。

表 2 循环论数的确定

N_k	10	12	14
k	2	2	2
循环论数	5	6	7

方案的解密结构图见图 4，其中 R1～R3 以及 my_sbox1 和 my_sbox2 用于建立流水线（由于引入 S 盒实现，故加入 R3 对流水进行补偿）；引入 R0 可以实现两次加密之间的部分并行工作，并保证流水线不断流；引入 R4 起输出同步作用；引入 R5～R12 为存储初始向量 **IV**（$\mathbf{IV}_1$~$\mathbf{IV}_4$）和正在解密的数据块以便下一次解密（这里实际上采用了共享技术）。需要指出的是，在密钥的输入处引入了两个 128 位的寄存器 R13 及 R14，其中 R13 用于存储最后一个密钥 Key[N_k]，R14

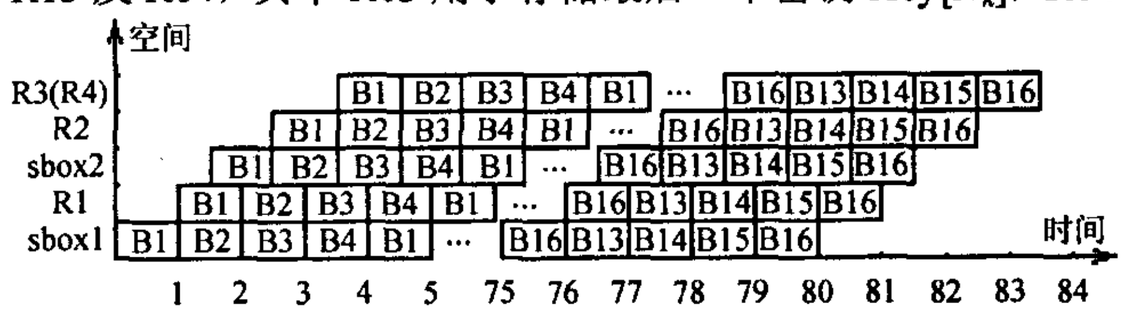

图 4 混合流水结构实现 AES 解密算法

用于密钥的延后输出，以 2clk 作为时钟，从而得到所需要的密钥的并行输入，这里实际上也采用了共享技术。该方案完成一个数据块的解密共需要 5（6，7）个时钟，而密钥的输入顺序为 K10, K9,…, K0, …, K9, …, K0, …

必须指出的是，在用 FPGA 实现中，上述寄存器用存储器实现（而图中的 R5～R12 可以做成 FIFO），这样可以直接利用器件中的 EAB/ESB（Embedded Array Block/Embedded System Block）资源。

加密结构与解密结构类似，不同点在于使用了 5 个 128 位的寄存器存储初始向量 **IV** 和当前输出的数据块（为了下一次加密），其图略。

5　混合流水结构方案分析

图 5 为 AES 解密方案流水处理的时空图。这里假设 N_k =4，明文长度为 16 个数据块（B1～B16）。注意：在每个数据块的最后一轮加密中 R3 处于空闲状态，数据块进入 R4 中；而实际上 R4 并非流水用寄存器，图中示出仅为方便起见。

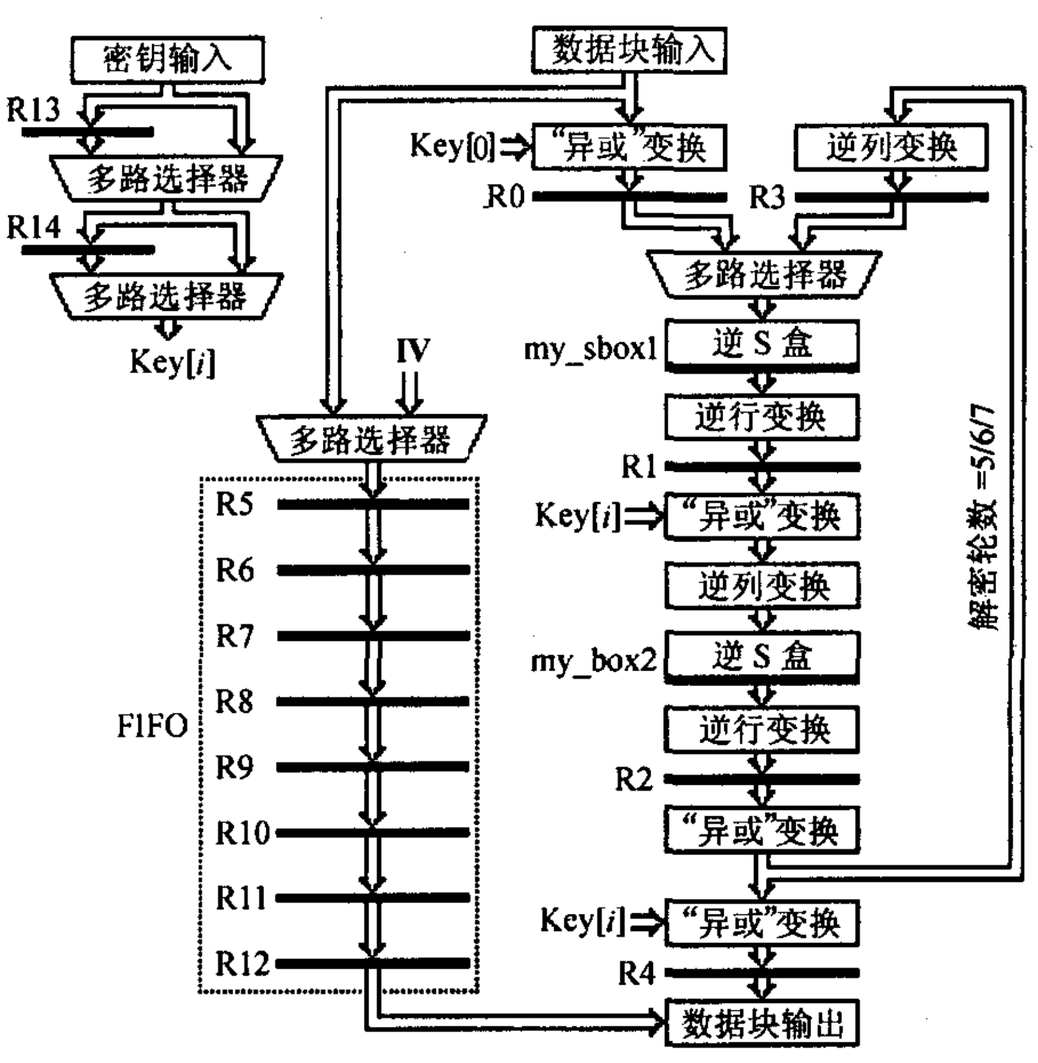

图 5　流水处理的时空图

根据时延分析知，该结构最大的时钟频率 f_{max} 为 45MHz。

(1) 本方案完成一个数据块解密大约需要 5 个时钟，其最大吞吐率计算公式为 $TP_{max}=128f/5$（f 表示时钟频率），而实际吞吐率比最大吞吐率要小。对于 n（假定 n 为 4 的倍数）个数据块以时钟频率为 f 解密，其实际吞吐率为 $TP=128nf/(4+4+5n)=128nf/(5n+8)$。可见当 n 足够大时，$5n>>8$，这样就有 $TP=TP_{max}$。本例中的最大吞吐率和实际吞吐率分别为 1.05Gbit/s 和 954Mbit/s.

（2）从时空图中可以很容易计算出流水线的效率为 η=（n 个任务实际占用的时空区）/（m 个段总的时空区）=91.43%。

（3）采用基本循环结构，一般可以做到大约 10 个系统时钟完成一个数据块的解密；而采用本方案，完成一个数据块解密平均大约需要 5 个时钟，因此加速比 $S_p \approx 2$。

6　结论

（1）采用交叉处理模式使得采用流水线结构实现工作于反馈模式的 AES 算法成为可能，提高了加解密的安全性；

（2）混合流水结构综合了内、外流水结构以及循环轮展开结构的各自优点，虽相应增加了所用资源，但使处理效率得到了显著提高；

（3）与软件实现相比，硬件实现在整体性能和系统安全两方面有着显著的优势；而设计的灵活性必将导致 CPLD/FPGA（Complex Programmable Logic Device/Field Programmable Gate Arrey）在网络数据加解密领域的应用达到新的高潮。

参 考 文 献

[1] Daemen J, Rijmen V. AES Proposal: Rijndael. Available at http://csrc.nist.gov/encryption/aes/rijndael/Rijndael.pdf

[2] Clodowiec P, Gaj K, Bellows P, Schott B. Experimental testing of the gigabit ipSec-compliant implementations of Rijndael and Triple DES using SLAAC-1V FPGA accelerator board. Information Security 4th International Conference, ISC2001 Malaga, Spain, October 2001: 78－92.

[3] Gaj K, Chodowiec P. Fast implementation and fair comparison of the final candidates for Advanced Encryption Standard using Field Programmable Gate Arrays. Available at http://ece .gmu.edu

[4] Nechvatal J, BarkerE, Bassham L, Burr W, Dworkin M, Foti J, Roback E. Report on the development of the Advanced Encryption Standard (AES). Available at http://www.nist.gov/aes/

彭艮鹏：　男，1971 年生，硕士生，学习方向：电路与系统.

刘常澍：　男，1946 年生，教授，研究方向：电路与系统、计算机测控.

李志华：　男，1963 年生，讲师，研究方向：计算机测控.