

可兼容 AES-128、AES-192、AES-256 串行 AES 加密解密电路设计

韩少男, 李晓江

(中国科学院 微电子研究所, 浙江 杭州 310053)

摘 要: 通过分析 AES 算法的基本原理, 对算法中的 AES-128、AES-192、AES-256 三种不同的加密解密模式进行了综合设计, 有效地利用了公共模块, 与单个分别实施各个加密解密模式相比, 大大减少了硬件电路面积. 针对目前 AES 实现方法中的 key 产生模块进行了理论分析, 提出了一种新的实现电路结构. 设计出的串行 AES 硬件加密解密电路经综合后得到的芯片面积为 31 286 门, 最高工作频率为 66MHz, 可以满足目前的大部分无线传感网络的数据交换速率的需求.

关键词: AES 算法; AES-128; AES-192; AES-256; 加密解密; ASIC

中图分类号: TN402

文献标识码: A

文章编号: 1000-7180(2010)11-0040-06

Compatible AES-128、AES-192、AES-256 Serial AES Encryption and Decryption Circuit Design

HAN Shao-nan, LI Xiao-jiang

(Institution of Microelectronics, Chinese Academy of Science, Hangzhou 310053, China)

Abstract: This paper, by analyzing the basic principles of AES algorithm, integrating AES-128, AES-192, AES-256 into a single design, which effectively uses the public modules and greatly reduces the chip's area compared with a single implementation of these modules respectively. In this paper, through the analysis of the current implementation of KEY in AES algorithm, the author proposes a new circuit structure. And the result shows that the area of serial AES encryption and decryption circuit is only 31 286, a maximum operating frequency is 66MHz, which meets most data exchange rate demands of the current wireless sensor networks.

Key words: AES algorithm; AES-128; AES-192; AES-256; encryption and decryption; ASIC

1 引言

随着互连网络的广泛应用,越来越多的数据通过网络进行,在传输过程中数据安全问题变得尤其重要.采用数据加密技术可以有效地保护数据传输的安全,因此需要一种高效率、高安全性的数据加密技术.传统的加密方法—DES(Data Encryption Standard)已经不再安全,由 J. Daemen 和 V. Rijmen 推举的 Rijndael 算法为新一代的标准加密算法,也称为 AES(Advanced Encryption Standard)加密算法

标准.随后 AES 标准被 IEEE 的 802.11i 标准,802.15.4 标准和 Zigbee 协议所采用作为其加密标准, AES 便在无线局域网(WLAN)、无线个人网络(WPAN)、无线传感网络(WSN)和智能卡系统广泛的应用^[1],设计出小面积、低功耗的 AES 硬件加密解密电路变得尤其重要.本设计采用串行 AES 加密解密电路的设计,比流水线的实现方法在面积和功耗上有着明显的优势.

在 AES 算法中被加密或解密的数据块是 128bits 的,而根据数据加密的等级不同所采用的加

收稿日期: 2009-09-12; 修回日期: 2009-11-12

基金项目: 浙江省高科技基金支持项目(2006C11107)

密密钥(keys)长度也不同.按照 AES 算法标准,一般秘密级加密的 keys 长度为 128bits(AES-128),机密级加密的 keys 长度为 192bits(AES-192),绝密级加密的 keys 长度为 256bits(AES-256)^[2].在银行系统中,通常会采用变长 keys 加密方式来对数据流进行加密,因此本设计将 AES-128、AES-192、AES-256 这三种不同的 key 长度加密解密方式相兼容,使用者可以通过设定加密解密模式来选择相对应 key 的长度进行数据的加密解密操作.

2 AES 算法

由于 AES 算法设计简单,与公钥密码算法不同,没有复杂的乘法运算,易于实现,灵活性强,轮函数良好的并行特性有利于硬件设计与实现,2006 年 AES 算法已经成为对称密钥加密中最流行的算法之一.

AES 算法的明文加密过程是由 SubBytes, ShiftRows, MixColumns 三个模块构成,Key 的加密和解密过程是由 RotWord, SubWord, XorRcon 三个模块组成.相应的密文解密的过程由 InvSubBytes, InvShiftRows, InvMixColumns 三个模块组成^[2].整个 AES 加密解密算法流程如图 1 所示.

图 1 中的输入明文为 128bits 数据,根据加密解密所采用的 key 长度不同,加密解密的所需总的循环加密周数(Nr)也不同,对应 AES-128 的 Nr 为 10, AES-192 的 Nr 为 12, AES-256 的 Nr 为 14.

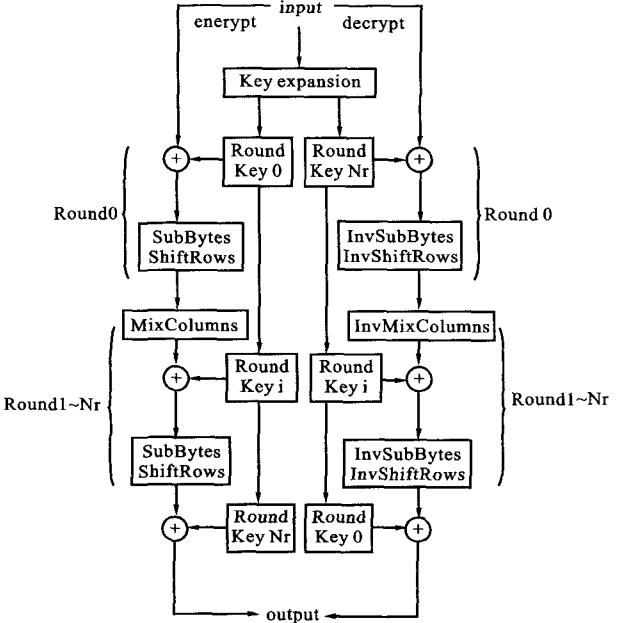


图 1 AES 加密解密流程

3 各个子模块的优化设计

3.1 SubBytes 和 InvSubBytes

SubBytes 是通过 128bit 的替代表(简称为 S-BOX)来对输入的 128bit 数据进行替换,输出为 S-BOX 中对应的 128bit 数据.每一个 S-BOX 的值是通过基于 Galois 域 $GF(2^8)$ 的乘法逆运算,和线性变换(式(1))得到.其中, $GF(2^8)$ 乘法逆运算所需的不可约多项式为: $m(x) = x^8 + x^4 + x^3 + x + 1$. 线性变换的表达式为

$$\begin{bmatrix} b'_0 \\ b'_1 \\ b'_2 \\ b'_3 \\ b'_4 \\ b'_5 \\ b'_6 \\ b'_7 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \end{bmatrix} + \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ c_6 \\ c_7 \end{bmatrix} \quad (1)$$

对应的 InvSubBytes 有以下两个步骤组成,一是对输入数据进行以下的线性变换:

$$\begin{bmatrix} b'_0 \\ b'_1 \\ b'_2 \\ b'_3 \\ b'_4 \\ b'_5 \\ b'_6 \\ b'_7 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \end{bmatrix} + \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ c_6 \\ c_7 \end{bmatrix} \quad (2)$$

二是对线性变换的结果进行有限域 $GF(2^8)$ 上的乘法逆运算.

在 AES 标准中给出了 SubBytes 和 InvSubBytes 的实现表^[2],如果直接使用查表法来实现,会在硬件电路中生成一个大的 ROM 模块,浪费硬件电路资源而且影响加密解密速度.

通过观察 SubBytes 和 InvSubBytes 的实现方法可以知道,两者仅在线性变换的实现上不同,在基于 $GF(2^8)$ 上的乘法逆运算是相同的,如果直接采用文献^[2]中提出的用扩展型欧几里得算法来实现 $GF(2^8)$ 上的乘法逆运算来进行硬件电路实现,其设计复杂,硬件电路实现代价过高^[3].可以采用文献^[4]

中提出的基于组合域 $GF((2^4)^2)$ 的乘法逆运算硬件电路实现方法,如图 2 所示.

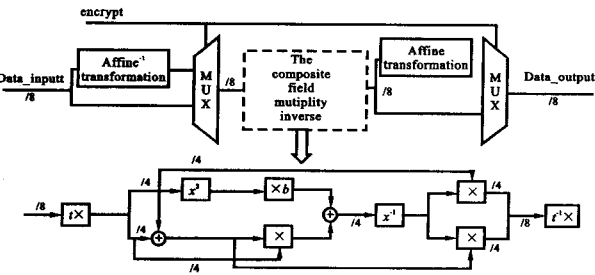


图 2 基于组合域 $GF((2^4)^2)$ 的乘法逆运算实现 SubBytes 和 InvSubBytes

文献[4]中的基于组合域 $GF((2^4)^2)$ 的乘法逆运算硬件电路实现方法是将基于 $GF(2^8)$ 上的复杂乘法逆运算转化到组合域 $GF((2^4)^2)$ 中,然后在将 $GF((2^4)^2)$ 中的乘法逆运算转化到 $GF(((2^2)^2)^2)$ 中如图 3 所示.

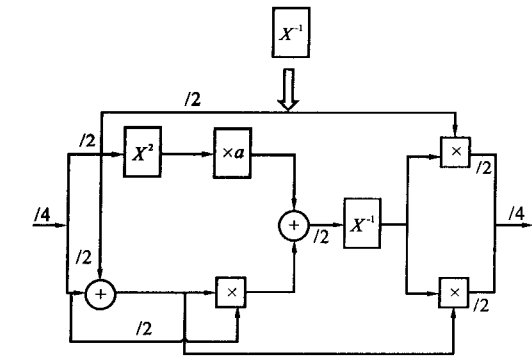


图 3 将 $GF((2^4)^2)$ 中的乘法逆运算转化到 $GF(((2^2)^2)^2)$ 中

但是经过分析可以发现,从 $GF(2^8)$ 上向 $GF((2^4)^2)$ 上进行复合域变换来计算乘法逆运算可以大大减少硬件复杂度,但是从 $GF((2^4)^2)$ 上向 $GF(((2^2)^2)^2)$ 上进行复合域变换计算乘法逆运算其硬件复杂度反而有所增加.对此本文提出了一种改进型的基于 $GF((2^4)^2)$ 实现乘法逆运算的方法.若输入信号为 4 位的 X_0, X_1, X_2, X_3 ,在经过基于 $GF(2^4)$ 上乘法逆运算后可以得到输出 Y_0, Y_1, Y_2, Y_3 的真值表,如表 1 ~ 表 4 所示.

由真值表分析可以得到 Y_0, Y_1, Y_2, Y_3 的表达式分别为

$$Y_0 = X_3X_2X_1 + X_2X_1X_0 + X_2X_0 + X_2X_1 + X_3 + X_2 + X_1 + X_0;$$
$$Y_1 = X_3X_1X_0 + X_1X_0 + X_2X_0 + X_2X_1 + X_3X_1 + X_3;$$

$$Y_2 = X_3X_2X_0 + X_3X_0 + X_2X_0 + X_1X_0 + X_2 + X_3;$$

$$Y_3 = X_3X_2X_1 + X_3X_0 + X_3X_1 + X_3X_2 + X_2 + X_1 + X_3;$$

这样就可以直接用组合逻辑实现有限域 $GF(2^4)$ 上的乘法逆运算.

表 1 Y_0 的真值表

$X_0 \backslash X_1$	X_2	0	0	1	1
$X_1 \backslash X_3$	0	0	1	0	1
00	0	0	1	1	0
01	1	1	1	1	1
10	1	0	0	0	1
11	0	0	0	1	0

表 2 Y_1 的真值表

$X_0 \backslash X_1$	X_2	0	0	1	1
$X_1 \backslash X_3$	0	0	1	0	1
00	0	0	0	0	1
01	0	1	1	1	1
10	1	0	1	1	1
11	1	0	1	0	0

表 3 Y_2 的真值表

$X_0 \backslash X_1$	X_2	0	0	1	1
$X_1 \backslash X_3$	0	0	1	0	1
00	0	0	0	0	1
01	1	0	1	1	1
10	1	0	1	1	1
11	0	1	0	0	0

表 4 Y_3 的真值表

$X_0 \backslash X_1$	X_2	0	0	1	1
$X_1 \backslash X_3$	0	0	1	0	1
00	0	0	0	1	1
01	1	1	1	0	0
10	1	0	1	0	0
11	1	0	0	0	1

3.2 ShiftRows 和 InvShiftRows

ShiftRows 是把一个 4×4 - bytes 数据块的每一行进行对应 0~3byte 的循环移位. InvShiftRows 是 ShiftRows 的逆操作,为了减少数据读取 SubBytes 和 InvSubBytes 的操作,提高数据加密解密效率,通常把 ShiftRows 和 SubBytes 模块相结合,InvShiftRows 和 InvSubBytes 模块相结合^[5].

3.3 MixColumns 和 InvMixColumns

MixColumns 是将每一列看作为 $GF(2^8)$ 上的多

项式, 并且和多项式 $a(x)$ 进行模 $x^4 + 1$ 的乘法运算, 其中 $a(x) = \{03\}x^3 + \{01\}x^2 + \{01\}x + \{02\}$, 用矩阵表达式形式^[2]为 $s'(x) = a(x) \times s(x)$:

$$\begin{bmatrix} s'_{0,c} \\ s'_{1,c} \\ s'_{2,c} \\ s'_{3,c} \end{bmatrix} = \begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{bmatrix} \begin{bmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{bmatrix} \tag{3}$$

InvMixColumns 是 MixColumns 的逆运算, 即是把每一列看作为 GF(2⁸) 上的多项式并和 $a^{-1}(x)$ 进行模为 $x^4 + 1$ 的乘法逆运算, 其中 $a^{-1}(x) = \{0b\}x^3 + \{0d\}x^2 + \{09\}x + \{0e\}$, 用矩阵表达式形式为 $t'(x) = a^{-1}(x) \times s(x)$:

$$\begin{bmatrix} t'_{0,c} \\ t'_{1,c} \\ t'_{2,c} \\ t'_{3,c} \end{bmatrix} = \begin{bmatrix} e & b & d & 9 \\ 9 & e & b & d \\ d & 9 & e & b \\ b & d & 9 & e \end{bmatrix} \begin{bmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{bmatrix}$$
$$= \begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} + \begin{bmatrix} 4 & 0 & 4 & 0 \\ 0 & 4 & 0 & 4 \\ 4 & 0 & 4 & 0 \\ 0 & 4 & 0 & 4 \end{bmatrix} \begin{bmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{bmatrix} \tag{4}$$

通过式(4)可以看到 InvMixColumns 的运算过程中包含了 MixColumns 的运算, 因此可以把这两个运算相结合, 节省硬件电路资源, 减少芯片面积, 具体的实现方法参考文献[5]. 在本设计中把 1 个 word 的混合电路和 1 个 128bits 的寄存器组成一个 128bits 的 MixColumns 和 InvMixColumns 的混合运算电路, 如图 4 所示.

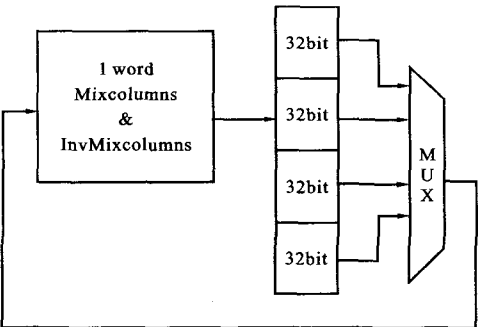


图 4 128bits 的 MixColumns 和 InvMixColumns 的混合电路结构图

3.4 key 加密解密的产生

根据文献[2], key 的加密过程是由 RotWord, SubWord, XorRcon 三个模块组成, 如图 5 所示. RotWord 是在 key 的加密过程中将 1 个 word 中的 4bytes 数据进行循环左移一位. 如: $[b_0, b_1, b_2, b_3]$ 经过 RotWord 变换之后成为 $[b_1, b_2, b_3, b_0]$. 相对应的, 在 key 解密过程中为将 1 个 word 的 4bytes 数据进行循环右移一位. SubWord 是在 key 的加密过程中将 1 个 word 中的 4bytes 数据分别进行对应的 S-BOX 变换, 对应的在 key 的解密过程中对 1 个 word 中的 4bytes 数据分别进行对应的 INV-S-BOX 变换. XorRcon 是在 key 的加密解密过程中将 1word 数据与每个 round 中的 1 个固定的值进行异或运算.

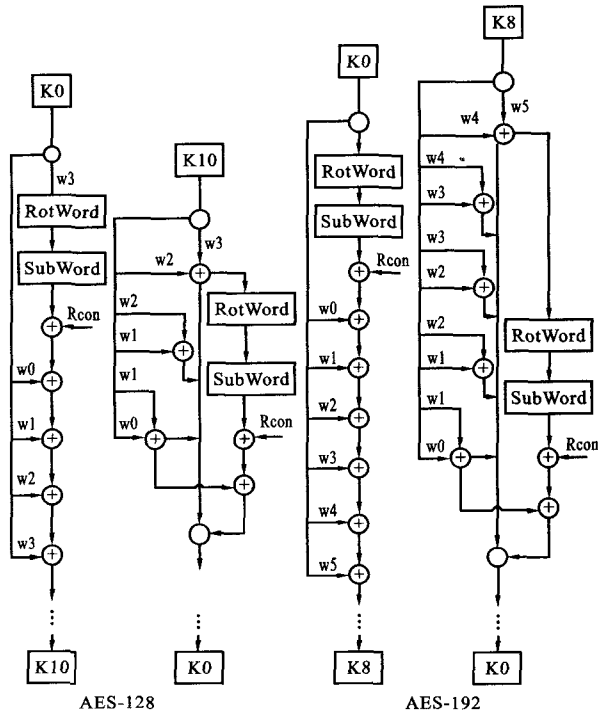


图 5 AES-128 和 AES-192 的 key 加密和解密算法步骤

对于使用 key 的长度不同, 所需要的加密解密步骤及加密解密周期也不同. AES-128 和 AES-192 的 key 有着相似的加密解密过程如图 5 所示, 但稍有区别的是 AES-128 的 key 加密解密周期为 10, 而 AES-192 的周期为 8.

而对于 AES-256 的 key 加密解密过程则有了很大的不同. AES-256 的 key 高 128bits 数据采用了和 AES-128, AES-192 相同的加密解密方法, 低 128bits 的数据则采用了另一种处理方法, 见图 6.

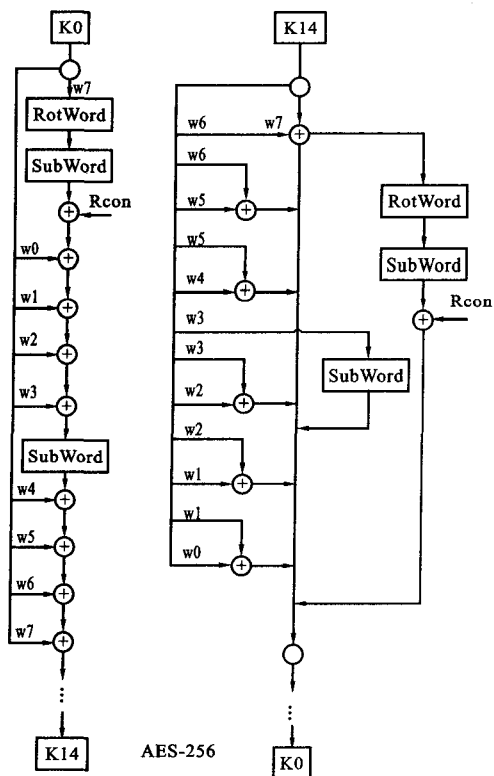


图6 AES-256的key加密和解密算法步骤

在对明文进行加密的过程中, key 的加密过程可以跟明文的加密过程同步, 但是在对密文的解密过程中, 首先用到的是 AES-128 的 round10 的 key、AES-192 的 round12 的 key、AES-256 的 round14 的

key, 也就是 key 加密最后一轮产生的 key. 在很多目前的设计方法中, 通常是在对密文解密之前, 先把 key 加密的过程完成, 并用寄存器存储 round1 到 round10 中的 key (以 AES-128 为例)^[5], 这种设计的优势在于只需设计出 key 的加密电路, 而且解密过程可以直接从寄存器中读取 key 的值, 不用额外的 key 解密生成电路. 但是这种设计的缺陷在于用于保存各个 round 中的 key 值所需的寄存器数量过多, 而且由于在对密文进行解密的过程中, 对 key 的要求并不是即时的, 即允许一定时间来产生相应 round 的 key, 这样可以设计出一个 key 加密解密通用电路, 可以在较少的占用硬件电路资源的情况下, 满足系统对加密解密速度需求, 如图 7 所示.

由 key 的加密解密步骤可以发现, 在每个 round 的 key 加密和解密过程中需要 5 个时钟周期, Rotword、SubWord、Rcon 三个占用硬件电路面积最大, 功耗最高的模块仅仅在 1 个时钟周期运行, 也就是说在进行 key 的内部 byte 之间的异或运算时完全可以关掉这三个高功耗的模块. 这里引入一个门控时钟电路对 Rotword、SubWord、Rcon 三个模块的时钟进行控制, 在这三个模块没有运行的时候关闭三个模块的时钟, 从而可以降低系统功耗. 门控时钟电路要注意避免毛刺产生, 本设计中采用的门控时钟电路是由文献[6]提出的电路结构.

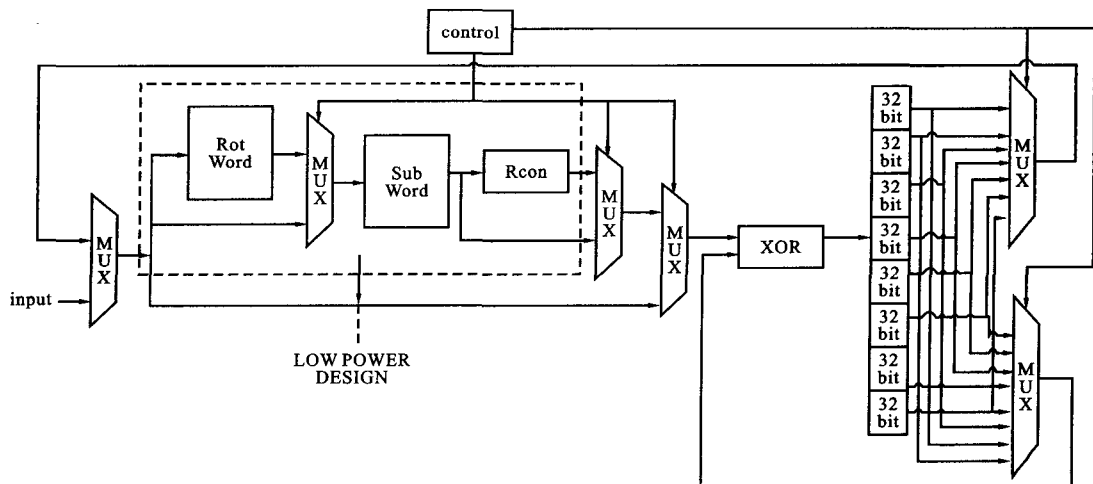


图7 AES-128、AES-192、AES-256 通用 key 加密解密电路

4 串行 AES 硬件加密解密电路结构

图 8 显示了整个串行 AES 硬件加密解密顶层电路结构, 包括了两个部分: 一是 key 的产生模块, 二是 AES 的加密解密功能模块. 这两个模块都受控

制中心控制, 并采用了串行的电路结构设计. 具体的内部电路组成如图 9 所示. 在 AES-128 的情况下每 44 个时钟周期完成一次 128bit 数据加密, 每 54 个时钟周期完成一次 128bit 数据解密. 在 AES-192 的情况下每 50 个时钟周期完成一次 128bit 数据加密,

每 66 个时钟周期完成一次 128bit 数据解密. 在 AES-256 的情况下每 64 个时钟周期完成一次 128bit 数据加密, 每 86 个时钟周期完成一次 128bit 数据解密.

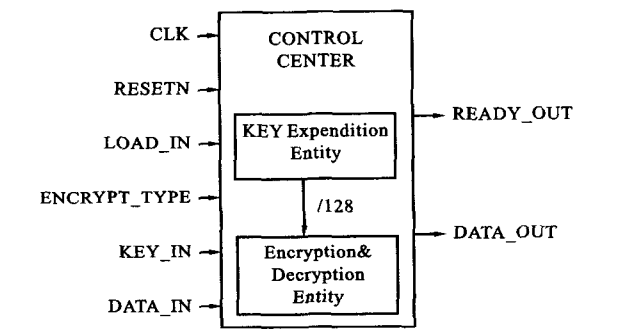


图 8 串行 AES 硬件加密解密顶层模块

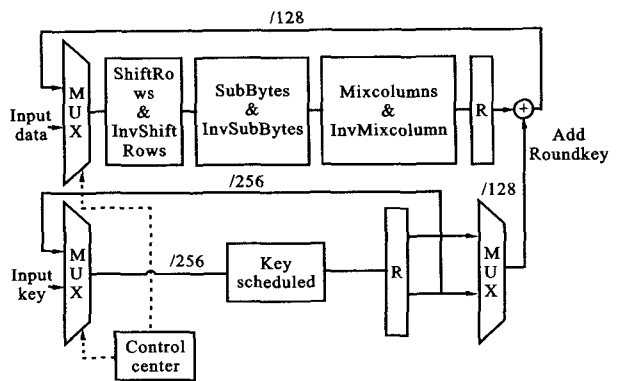


图 9 串行 AES 硬件加密解密内部电路结构图

控制中心(control center)是由一个状态机来完成对数据加密或解密过程的控制. 该状态机的状态示意图为图 10 所示.

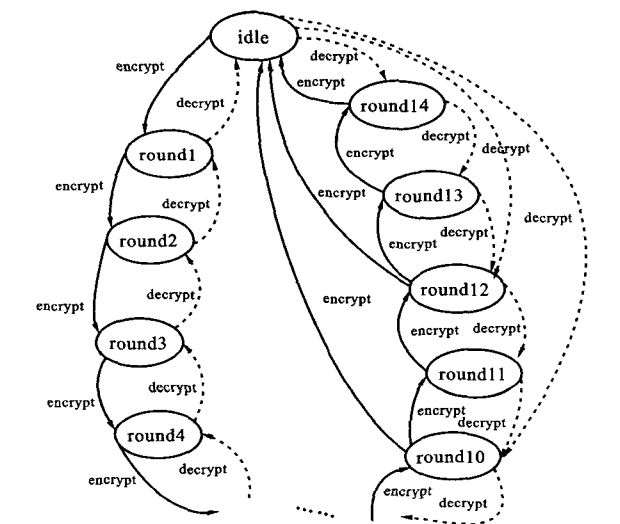


图 10 控制中心状态机状态示意图

5 电路性能比较

用 Synopsys 的 Design Ware 工具在中芯国际 (SMIC)0.18 μ m 的工艺下对以上的串行 AES 硬件加密解密电路进行综合仿真, 得到的结果如表 5 所示, 各个模式加密解密的性能如表 6 所示.

表 5 本设计电路综合仿真结果

组成部分	门电路数	百分比/%
ShiftRows/InvShiftRowsS-BOX/ INV-S-BOX	8 348	26.69
MixColumns/InvMixColumns	3 792	12.12
Key Scheduled	4 311	13.78
Control Center + registers	13 518	43.20
Others	1 317	4.21
Total	31 286	100

表 6 串行 AES 电路性能

工作频率	AES-128 数据吞吐率	AES-192 数据吞吐率	AES-256 数据吞吐率
66MHz	191(Mb/s)/ 156(Mb/s)	169(Mb/s)/ 128(Mb/s)	132(Mb/s)/ 98(Mb/s)

6 结束语

本文通过分析 AES 算法的基本原理,对算法中的 AES-128、AES-192、AES-256 三种不同的加密解密模式进行了综合设计,大大减少了硬件电路面积. 本文针对目前 AES 实现方法研究不多的 key 产生模块进行了理论分析,并提出了一种新的实现电路,该电路相比目前各种设计所采用的方法,减少了硬件电路的面积,降低了功耗,这对于 AES 广泛应用的无线传感网络方面有着重要的意义. 设计出的串行 AES 硬件加密解密电路经过 Synopsys 的 Design Ware 工具在中芯国际 (smic)0.18 μ m 的工艺下综合,得到的芯片面积为 31 286 门,最高工作频率为 66MHz,对 128bit 数据在 key 的长度为 128bits 加密和解密的速率为 191(Mb/s)/156(Mb/s),在 key 的长度为 192bits 加密和解密的速率为 169(Mb/s)/128(Mb/s),在 key 的长度为 256bits 加密和解密的速率为 132(Mb/s)/98(Mb/s),可以满足目前的大部分无线传感网络的数据交换速率的需求.

(下转第 50 页)

- [2] Behzad Razav. Design of analog CMOS integrated circuits [M]. New York: The McGraw - Hill Companies, Inc, 2001:60 - 61.
- [3] Mohamed A I Mostafa, Sherif H K, Elmala M. A 60 - dB 246 - MHz CMOS variable gain amplifier for sub - sampling GSM receivers [J]. IEEE Transactions on Very Large Scale Integration (VLSI) System, 2003, 11(5): 835 - 838.
- [4] Bryan Scott Puckett. Design and analysis of an attenuator - based, four - channel, differential, 150MHz, linear - in - dB VGA with sub - nanosecond delay dispersion for PET applications[D]. Knoxville: The University of Tennessee, 2001.
- [5] Chien M Ta, Chee Hong Yong, Wooi Gan Yeoh. A 2. 7mW, 0.064mm² linear - in - dB VGA with 60dB tuning range, 100MHz bandwidth, and two DC offset cancellation loops[C]// IEEE International Workshop on Radio - Frequency Integration Technology. Austria: Graz, 2005:74 - 77.
- [6] Ji - Hun Kim, Chang - Seok Chae, Young - Jin Woo. A CMOS variable gain amplifier with wide dynamic range and accurate dB - linear characteristic[C]// ICACT. Korea, 2006,1:831 - 835.
- [7] 薛庆华,周玉梅.一种高性能 CMOS 能隙电压参考源的设计[J]. 微电子学与计算机,2003,20(10):38 - 40.

作者简介:

黄晓辉 男,(1984-),硕士研究生.研究方向为模拟集成电路设计.

阎跃鹏 男,(1963-),博士生导师.研究方向为无线通信系统,卫星导航定位等.

曾云 男,(1956-),博士生导师.研究方向为电子器件与集成电路及其应用.

(上接第 45 页)

参考文献:

- [1] Feldhofer M, Dominikus S, Wolkerstorfer J. Strong authentication for RFID systems using the AES algorithm [J]. Cryptographic Hardware and Embedded Systems - CHES,2004(3156):357 - 370.
- [2] Advanced Encryption Standard. National institute of standards and technology[S/OL]. [2001 - 11 - 26]. <http://csrc.nist.gov/publication/drafts/dfips-AES.pdf>.
- [3] 高磊,戴冠中. AES 算法中 SubBytes 变换的高速硬件实现[J]. 微电子学与计算机,2006,23(7):49 - 51.
- [4] Satoh A, Morioka S, Takano K, et al. A compact rijndael hardware architecture with S - box optimization[C]// Cryptology - ASIACRYPT 2001, 7th International Conference on the Theory and Application of Cryptology and Information Security. Gold Coast, Australia, 2001:239 - 254.
- [5] Jia Zhao, Xiaoyang Zeng, Jun Han, et al. Very Low - cost VLSI implementation of AES algorithm[C]// IEEE Asian Solid - State Circuits Conference. China: Hangzhou, 2006:223 - 226.
- [6] 王彬,任艳颖. 数字 IC 系统设计[M]. 西安:西安电子科技大学出版社,2005:73 - 77.

作者简介:

韩少男 男,(1984-),硕士研究生.研究方向为专用集成电路设计 and 应用.

李晓江 男,(1972-),硕士生导师.研究方向为专用集成电路设计 and 应用.