适用于嵌入式系统的 AES 加密 IP 核设计*

周小果,唐立军,谢新辉,宋海吒
(长沙理工大学,湖南 长沙 410114)

摘 要:介绍了 AES 加密标准的 Rijndael 实现方法,设计了一种适合应用于嵌入式系统 32 位数据界面时序紧凑的 AES 加密 IP 核。该 IP 核能以较低的资源消耗实现在低端 FPGA 上速度为 256 Mb/s 的 AES 加密,且可将数据位宽扩展为 64 位或 128 位等,满足多种数据位宽应用的要求。该设计是一种低成本高性能的 AES 加密实现方法。

关键词:AES 加密;嵌入式系统;IP 核设计

中图分类号:TP3 **文献标识码**:A **文章编号**:1674-7720(2010)15-0083-03

IP core design in AES encryption to the embedded system

ZHOU Xiao Guo, TANG Li Jun, XIE Xin Hui, SONG Hai Zha
(Changsha University of Science and Technology, Changsha 410114, China)

Abstract: This paper describes the flow of encryption of AES algorithm (Rijndael). An efficient design of AES algorithm IP core using compact timing optimized methods is implemented. This implementation results in greater throughputs and less resource requirements with the 32-bit interface, and the 64-bit or 128-bit interface are optional. This makes it a viable data/communication security solution for a variety of embedded and consumer electronics.

Key words: AES algorithm; embedded system; IP core design

Rijndael 加密算法于 2000 年被确定为美国高级加密标准 AES(Advanced Encryption Standard),现在已是工业界数据加密的通用标准之一。Rijndael 算法无论从理论分析,还是在实践应用都表现出很好的抵抗各种攻击的性能,其安全性是不容置疑的。

随着计算机技术、微电子技术的不断融合,嵌入式系统应用得到了迅猛发展。近年来嵌入式技术广泛用于解决保密信息的传输、存储和管理方面的问题。而这些都需要嵌入式系统集成有可靠的加密模块。现有一些应用中的加密模块还不尽如人意。本文针对此应用需求设计一种适用于嵌入式系统的加密 IP 核。

1 AES 算法

1.1 AES 算法描述

密钥密码体制分为流密码和分组密码两种。分组密码是信息与网络安全中实现数据加密、数字签名、认证及密钥管理的核心体制,具有速度快、易于标准化和便于软硬件实现等特点。AES 采用分组密码的加密方式,其分组长度分为 128 bit、192 bit、256 bit 三种,AES 密码在相同的轮函数作用下,迭代运算次数的不同可达到不同级别的安全强度。128 bit 分组长度的情况下,循环轮数指定为 11 次,目前还没有可行的算法可以对该模型进行有效攻击[1]。每一轮处理均为作用在中间结果上的一批运算,该中间结果称为状态,用 4×4 字节矩阵表示,其中,数据矩阵称为 State、密钥矩阵称为 Key。AES 加密涉及 5 种运算,分别是字节代换(SubBytes)、行移变换(ShiftRows)、列混合变换(MixColumns)、密钥加法(AddRoundKey)和密钥扩展(ExpandedKey)。

字节代换是对 State 每个字节进行独立非线性变换,由字节在 GF(2^8)域中求其乘法逆并外加一个仿射变换完成[2]。具体实现中广泛使用查表方式完成该步变换(实现该功能单元被称为 Sbox),以避免复杂的乘法运算。

行移变换是对 State 进行按行移位操作,第 0 行不移位,第 1 行循环左移一位,第 2 行循环左移两位,第 3 行循环左移三位。

*基金项目:长沙理工大学重点学科建设项目资助;长沙市科技局科技计划项目资助(项目编号 K0803081-11)

列混合变换是在有限域下将状态的每列$[a0\ \ a1\ \ a2\ \ a3]^T$乘以一个固定多项式$C(x)$模x^4+1，多项式$C(x)=\text{'03'}x^3+\text{'01'}x^2+\text{'01'}x+\text{'02'}$。该变换以矩阵形式表示为：

$$\begin{bmatrix} b0 \\ b1 \\ b2 \\ b3 \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 02 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \times \begin{bmatrix} a0 \\ a1 \\ a2 \\ a3 \end{bmatrix}$$

密钥加法是将轮密钥Key和状态State中对应字节按位“异或”。

密钥扩展提供轮变换对应密钥加法用到的各轮密钥。各轮运算中使用到的轮密钥都不相同，密钥扩展运算通过控制迭代运算次数计算出对应轮所需轮密钥[3]。

加密过程由11轮运算组成，其中首轮只进行密钥加法，接着进行9次轮变换，轮变换由字节代换、行移变换、列混合变换及密钥加法4个步骤构成，再进行末轮变换，在末轮中跳过列混合变换。末轮完成后输出密文数据。AES加密过程如图1所示。

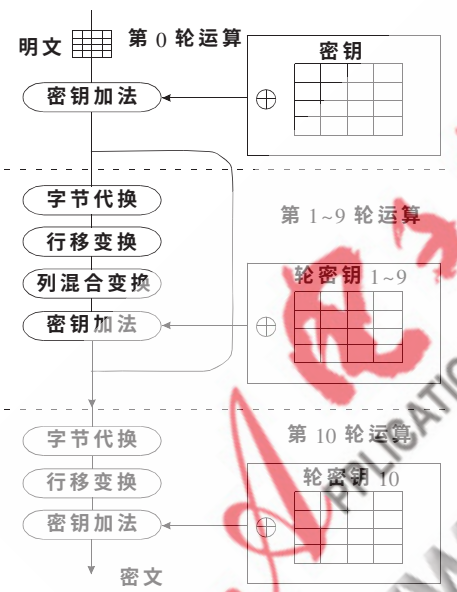

图1 AES加密流程图

1.2 常用AES优化实现

AES算法的轮操作特点看似更适合于在通用CPU平台下编程实现，而实际上，此种实现方式在性能方面存在加密速度慢等先天局限性。AES加密处理单元一般处于数据主干道上，其处理数据能力直接影响整个应用系统的外在性能表现，因此，研究数据处理能力强的硬件加密实现方式具有重要的意义。如何实现高性价比的硬件AES加密一直是加密算法应用领域研究的热点问题。

常见的硬件优化实现有如下几种方式：(1)串行方式。将轮函数展平，每轮对应一级组合逻辑，11轮迭代过程直接相连，前一级输出作为次一级的输入，每一个时钟周期均可完成一个分组处理；(2)迭代方式。各轮迭代只用一个对应轮函数功能的组合逻辑实体实现，每11个时钟周期完成一个分组处理；(3)流水线方式。用于提高系统工作时钟周期的流水线技术，一般仅在局部使用，或是与串行方式并用，可提高工作时钟频率，使其满足极大带宽的应用要求；(4)轮内实现流水线。在轮函数对应实体中插入寄存器，将一轮运算分至多个逻辑段完成，每个时钟周期仍能完成一个数据分组处理。

以上AES算法实现方式各有优缺点，但总体来说缺乏灵活性。当前应用于嵌入式系统的AES加密模块在灵活性、资源占用上还不是很理想。在对常用优化方法进行研究后，本文针对嵌入式系统设计一种AES加密IP核、实现低资源占用、高性能要求、32位数据位宽、且能方便进行并行连接，实现数据位宽扩展。

2 IP核设计

2.1 系统架构设计

IP系统分为时序控制、密钥处理、数据处理三个主要单元，其系统结构如图2所示。系统的工作模式分为闲置模式、密钥输入模式、单轮加密模式及连续加密模式。复位后系统处于闲置模式，单轮加密模式可以直接切换为连续加密模式，而连续加密模式需进入到闲置模式至少一个时钟周期后才可切换到单轮加密工作模式。

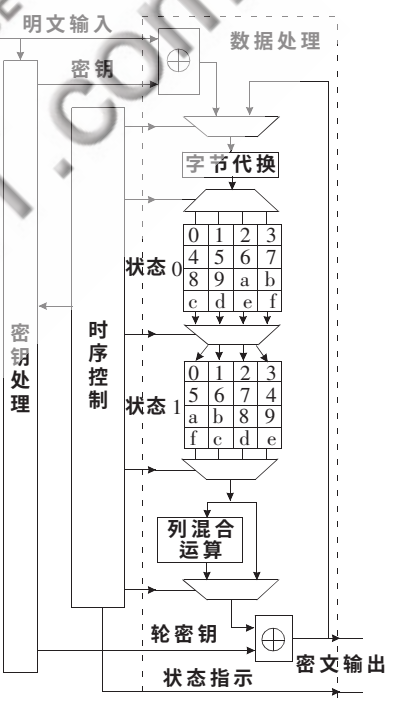

图2 IP核结构框图

密钥处理单元在系统进入密钥输入模式后的连续4个时钟周期从数据输入端口读入总共128位密钥数据，在第5个时钟周期到来时完成第一轮密钥的计算，然后系统返回闲置模式。在加密模式中密钥处理单元按算法需求实时计算各轮密钥，并按32位为一组输出，与数据通道中32位数据进行“异或”运算。在系统模式由加密模式转为闲置模式时，完成密钥处理单元的归位动作，使得单元状态与密钥输入后的状态相同，为下一次加密做准备。

数据处理单元在加密模式下对明文数据进行迭代运算。该单元检测到当前是最末轮数据处理时自动跳过列混合运算。在系统进入加密模式后，数据处理单元从输入端口分4次读入128位明文数据，经过接下来的40

个时钟周期数据运算过程后，得到密文中的第一个32位数据段。

时序控制单元负责整个系统关键控制信号的生成，控制信号集中由一个单元负责产生，不仅利于软件综合出较高的时钟频率，而且输出的时钟相位也有较优的一致性。

系统正常工作状态为先进行一次密钥载入操作，然后触发进入加密模式进行多次的数据加密。在需要时可以在闲置状态下再次进行密钥模式对密钥进行更新。

在系统闲置状态下，密钥加载信号被检测为有效时，系统进入密钥输入模式，对密钥数据进行读入、保存及生成第一轮密钥待用，而忽略密钥处理单元中是否先前已存在密钥数据。数据加载引脚指示系统由闲置模式输入单轮加密模式，因为128位数据要在4个时钟周期完成读入，加密后数据也需要4个时钟周期时间才能完成输出。因此，从明文数据输入到密文数据输出共需要等待40个时钟周期。如果此时检测到数据加载引脚信号有效则在输出密文的同时进行下一轮明文的读入，系统进入到连续加密模式，否则在接下来的4个时钟周期将密文输出后系统由单轮加密模式切换到闲置模式。连续加密模式适合用于进行批量数据加密处理，系统每40个时钟周期会从输入端口读入128位明文数据，同时在这40个时钟周期中将提供密文数据。在进行大量数据处理时，载入密钥及载入第一组加密数据的几个时钟周期均可忽略，系统性能为每40个时钟周期处理128位数据。在读入明文时，若检测到载入数据信号无效，则退出连续加密模式，系统回复到闲置状态。

2.2 设计要点

嵌入式系统中资源相对较少，一般数据位宽为32位或更少，如果设计的AES数据通道位宽达到128位或更多，虽然轮处理时间较短，但数据通道在I/O接口段必然利用率不高，而且占用资源难以降低，故本设计采用主通道数据位宽为32位的结构。由于每轮中列混合变换需要的32位数据与前4个时钟周期的行移变换输出结果相关，因此在行移与列混合单元间使用128位数据位宽，每4个时钟周期进行一次State0到State1转换。

从AES加密方法流程图中可见首轮与末轮均有特殊处理，未经过完整的4个轮处理过程，在一些设计中将首轮与末轮使用单独硬件实现，这样可使硬件代价减少2轮的运算时间。首轮结构简单，与标准轮处理过程差异较大，单独实现只需要在输入端加上32个“异或”门，能以较小的代价换取一轮的运算时间。而末轮与标准处理过程仅差列混合运算，单独实现需要将近多一倍的轮处理硬件，所以在本设计中由时序控制单元控制末轮处理时跳过列混合运算。

Sbox作为非线性运算部分，必需具有良好的差分特性和比较复杂的代数结构，如果使用独立逻辑电路实现，面积优化空间不大，多采用查表法实现。AES实现中的密钥扩展与数据处理都需要多个Sbox，通过分析综合软件资源消耗结果报告可知单个Sbox占用资源为208个LCs或是2KB RAM。减少Sbox的使用无疑成为降低资源占用的主要手段。经过调研，一般嵌入式系统对AES加密性能要求在160 Mb/s到480 Mb/s之间，考虑到本设计可灵活扩展的特性，设计中在数据处理路径使用4个Sbox进行时分复用，另采用4个Sbox进行密钥实时扩展。

3 硬件实现

本硬件实现在QuartusII8.0下使用Verilog HDL语言进行描述，在ModelSim6.2环境下进行调试与仿真，使用Synplify9协助完成综合与关键路径分析工作。主要分析该IP核综合到目标器件EP1C4F324C6中在80 MHz频率的性能表现及资源占用情况。同时在更高性能的目标器件EP2S15F484C3中也进行了综合及后仿真，以作纵向对比。

在QuartusII环境下选定目标器件为低成本Cyclone系列EP1C4F324C6设置速度与面积均衡优化模式，目标工作频率为90 MHz，使用逻辑单元实现Sbox查找表功能。综合报告显示实际综合频率为87.82 MHz(period=11.387 ns)，本IP核占用资源2 647(Logic Cells)，其中密钥扩展单元占用1 388(LCs)，时序控制单元占用45(LCs)。文中均以此IP核运行于80 MHz时钟频率进行性能分析。

选定综合到StratixII系列中EP2S15F484C3器件，综合频率FMAX达到169.12 MHz时占用资源Logic utilization 9%，其中Combinational ALUTS 834/12 480(7%)，Dedicated logic registers 598/12 480(5%)。将此综合结果在ModelSim中用133 MHz时钟驱动进行后仿真。

4 数据分析

仿真结果见表1，No.1采用常用测试数据，密钥为：2b7e1516_28aed2a6_abf71588_09cf4f3c，输入明文为3243f6a8_885a308d_313198a2_e0370734时，得到输出密文3925841d_02dc09fb_dc118597_196a0b32，结果正确无误。

表1 加密数据仿真结果

No.	输入明文	输出密文
1	3243f6a8_885a308d_313198a2_e0370734	3925841d_02dc09fb_dc118597_196a0b32
2	00010203_04050607_08091011_12131415	3b52086a_fa7b6d16_de6a4ae3_aaa3c410

该IP核工作在80 MHz时钟频率下时，数据吞吐量为128 bit×80 MHz/40 clk=256 Mb/s。速度/资源比(Mb/s)/Slice=256/(2647/2)=0.193。当并行连接IP核进行位宽扩展时，密钥扩展单元与时序控制单元可共用，进一步提高资源利用率。当扩展为128位数据位宽时，数据吞吐

量成倍增加，而速度/资源比也有所提高，几乎能达到(Mb/s)/Slice=1024/((2647×4-(1388+45)×3)/2)=0.326。

表2中数据显示本设计在32位数据位宽的同类设计中有一定的优势，从适用于嵌入式系统应用的角度考虑，本设计更具优越性。128位数据位宽的设计[6]中原文计算速度/资源比值时未考虑所占用的RAMs资源，而且文中设计为25 MHz时钟频率，进行数据分析时却将工作频率直接换算为54 MHz，而未对其设计是否可正常工作于此频率进行论证。其设计主要考虑建立流水作业以提高性能。参考文献[6]中采用6级流水线技术及复合域方法优化Sbox，达到了较优的设计指标，但其固定的128位数据位宽在嵌入式系统中应用有一定的局限性。

表2 不同实现的比较

参考文献	器件	Slices	RAMs(Block)	位宽	频率/MHz	吞吐率/(Mb/s)	(Mb/s)/Slices
[4]	XCV821BGE	2744		128	20.2	258	0.094
[5]	XC2V3000	4817	20	128	54	7040	1.461
[6]	XCV1000BG560	3374		128	83.2	1065	0.316
[7]	XC2530-6	522		32	60	69	0.132
本文	EP1C4F324C6	3145		128	80	1024	0.326
本文	EP1C4F324C6	1324		32	80	256	0.193

注：换算关系为2 LCs折合一个Slice，32 bit折合一个Slice。

在Synplify9下选定目标器件EP2S15F484C3，优先考虑提高速度，综合结果报告最高时钟频率超过240 MHz，说明本IP核设计合理，较好地利用了目标器件资源。如果将本IP核应用在更高性能目标器件上或是设计为ASIC将会有更大的性能提升。

本文设计的IP核在低端FPGA能以较低的资源消耗提供I/O性能，AES实现达到256 Mb/s，并提供适合应用于嵌入式系统中32位数据界面。在输入、输出端加FIFO数据缓存器可减少主器件被中断数据传输的次数，提供标准通信界面、简化主器件的操作时序。该IP核具有一定的灵活性，可将数据位宽扩展为64位或128位等，满足多种数据位宽应用的要求，是一种低成本高性能的AES加密实现方法。

参考文献

[1] 张文涛.分组密码的分析与设计[D].北京:中国科学院研究生院，2003.

[2] 马虹博，刘连浩.AES的S盒和逆S盒的代数表达式[J].计算机工程，2006，32(18):149-151.

[3] JOAN D，VINCENT R. AES proposal Rijndael (2nd version) [EB/OL].http://csrc.nist.gov/encryption/aes/rijndael/Rijndael.pdf，1999.

[4] SAQIB N A，HENRIQUES F B，PEREZ A D. AES algorithm implementation-an efficient approach for sequential and pipeline architectures[C]. Pro. Fourth Mexican International Conf. on Computer Science(ENC'03)，2003.

[5] TANG M，ZHANG H，LIU S，et al. High performance hardware design and study of AES [J]. Computer Engineering. 2006 (4):257-259.

[6] 钱松，周钦，俞军.AES算法的一种高效FPGA实现方法[J].微电子学与计算机，2005，22(7):89-91.

[7] CHODOWIEC P，GAJ K. Very compact FPGA implementation of the AES algorithm[M]. LNCS'03，2003.

(收稿日期：2010-01-13)

作者简介：

周小果，男，1978年生，研究生，主要研究方向：系统设计、集成电路设计与制造。

唐立军，男，1963年生，博士，教授，硕士生导师，主要研究方向：电子信息。