一种动态可重构物联网端侧加解密 IP 设计

饶辅天，孙瑞程，张彬彬，袁海英

（北京工业大学 信息科学技术学院 北京 100124）

摘要：随着物联网设备数量的迅速增长，端侧数据的安全与隐私保护需求日益突出。针对现有 AES 协处理器普遍存在的模式单一、密钥长度固定及缺乏统一接口等问题，本文设计了一种基于 AMBA APB3 总线的动态可重构 AES 加解密协处理器。该设计支持 ECB、CBC、CTR 三种工作模式与 128/192/256 位密钥长度，可通过寄存器配置灵活切换运行参数，满足物联网端侧多样化加解密需求。基于 FPGA 的功能与性能验证结果表明，该设计在资源占用、加解密速率和可扩展性方面表现优异，为端侧安全计算提供了标准化、低成本的硬件实现方案。

关键词：AES、动态可重构、全密钥长度支持、多工作模式支持、APB3 总线、FPGA、物联网安全

中图分类号：TN47　　**文献标识码：**A　　**文章编号：**（编辑部添加）

A Design of Dynamically Reconfigurable Encryption/Decryption IP for IoT End-Side

RAO FuTian, SUN Rui Cheng, ZHANG BinBin, YUAN HaiYin

(School of Electrical Information Engineering,BeiJing University Of Technology,BEIJING 100124,China)

Abstract: With the rapid expansion of Internet of Things (IoT) devices, data security and privacy protection at the edge have become increasingly critical. To address the limitations of existing AES coprocessors—typically restricted to a single mode, fixed key length, and lacking standardized interfaces—this paper presents a dynamically reconfigurable AES encryption/decryption coprocessor based on the AMBA APB3 bus. The proposed design supports three operation modes (ECB, CBC, CTR) and key lengths of 128, 192, and 256 bits, enabling flexible reconfiguration via register control to meet diverse IoT edge encryption demands. FPGA-based verification demonstrates excellent performance in resource utilization, throughput, and scalability, providing a standardized and reusable hardware solution for secure edge computing.

Keywords: AES, dynamic reconfiguration, full key length support, multi-mode support, APB3 bus , FPGA , IoT security

0 引 言

随着物联网（IoT）技术的迅速发展，大量智能终端设备被广泛应用于工业控制、智能家居、医疗健康和智慧城市等场景。端侧设备在采集、传输与存储数据的过程中面临潜在的安全威胁，数据加密成为保障系统安全与隐私的关键手段。作为国际标准的对称加密算法，AES（Advanced Encryption Standard）因其高安全性与高效硬件实现特性，已成为物联网数据保护的主流方案。

然而，如文献[4]和文献[6]现有 AES 协处理器多为特定场景设计，仅支持单一工作模式（如 ECB 或 CBC）或固定密钥长度，缺乏灵活性与扩展性，难以适应多样化的物联网端侧需求。在不同场景下，设备对加密强度、速率和功耗的要求存在显著差异。例如，环境传感器仅需低速低功耗的 ECB 加密，而视频流、远程控制或金融支付等高敏场景则需高吞吐率和更强的密钥安全性。表 1 列出了典型的物联网端侧应用及其加解密需求，可以看出，不同场景对工作模式、密钥长度和速率的要求差异明显。

此外，如文献[1]当前多数可重构 AES 设计仍处于实验室验证阶段，普遍采用简单寄存器或自定义信号接口实现数据交互，缺乏统一的标准总线接口。这种方式导致系统集成困难、可移植性低，不利于 SoC 级集成与大规模流片。相比之下，AMBA APB3 总线以结构简单、接口规范、集成度高等优势，已成为低功耗外设接口的事实标准，适用于加密协处理器在 SoC 系统中的模块化集成。

针对上述问题，本文提出了一种基于 APB3 总线接口的动态可重构 AES 协处理器。该设计支持多种工作模式与密钥长度，并通过寄存器配置实现在

线切换；同时在保持较低面积的前提下提升系统复用性与通用性。本文通过仿真与实物验证，验证了设计的正确性、可重构性与性能表现，该设计不仅提升了硬件资源的复用率与工程化水平，也为后续大规模流片和产业应用提供了技术基础。

场景	典型数据类型	工作模式	密钥长度	速度要求
环境传感器采样上报	温湿度、位置等数据	ECB	128	1kbps
设备敏感文件上传	图片、异常访问日志等数据	CBC	128	1Mbps
终端本地文件安全存储	安全策略、日志等敏感数据	CBC	128	10Mbps
工业摄像头边缘计算	视频流（全高清1920*1080）	CTR	128	10Mbps
远程控制终端	控制指令	CTR	256	25.6Mbps
工业控制器固件更新	固件文件	CBC	128	10Mbps
远程语音通信	音频流	CTR	128	512kbps
高敏个人信息与隐私数据	生物特征数据（指纹、虹膜等）、金融支付数据	CBC	256	10Mbps

表 1 常见物联网端侧场景及其典型加解密需求

1 系统 soc 架构

本文设计的 AES 协处理器总线接口采用标准 APB3 协议，方便在 SoC 系统中模块化集成。下图为本文协处理器与常见 MCU（如 CrotexM3）集成方式。

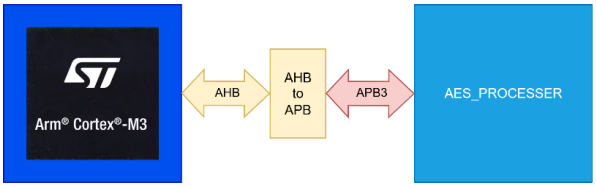


图 1 **协处理器与系统集成方式**

Fig. 1 FIGNAME

2 协处理器架构

本文设计的 AES 密码协处理器主要由模式寄存器、状态寄存器、数据寄存器堆、AES 算核及相关辅助电路构成。在架构上，本文采纳了与文献[1]类似的可重构设计思想，通过可调节的密钥扩展模块和模式选择模块，支持多种工作模式。具体而言，协处理器通过配置模式寄存器，能够控制计算通路以完成 ECB、CBC 和 CTR 三种工作模式的加解密运算。协处理器的整体架构如图 2 所示。

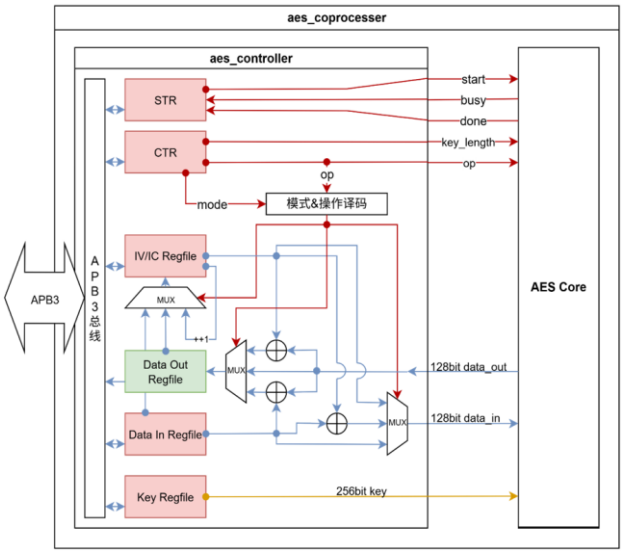
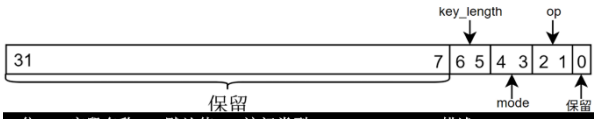


图 2 **协处理器架构**

工作模式与密钥长度通过配置模式寄存器进行设置。模式寄存器的具体位域定义见图 3。

由于 APB3 总线位宽为 32 位，而 AES 算法处理的数据块为 128 位，为解决位宽不匹配的问题，本设计为输入数据、输出结果和初始化向量（IV）/计数器（IC）值均设置了 4 个 32 位的寄存器进行缓存。为支持 128、192 及 256 三种密钥长度，密钥缓存区则设置了 8 个 32 位寄存器。其工作流程如下：主处理器通过 APB 总线依次将 128 位数据（或密钥/IV）分 4 次写入对应的 32 位寄存器。当所需数据全部就绪后，通过向状态寄存器写入启动命令 32'h01 来触发 AES 算核开始运算。此后，主处理器可通过轮询状态寄存器来等待运算完成。当状态寄存器指示运算结束时，主处理器即可从结果寄存器中分次读取 128 位的加密或解密结果。



位	字段名称	默认值	访问类型	描述
0	保留	0b	/	保留
[2:1]	op	00b	R/W	操作选择 2'b00 NONE 2'b01 密钥扩展 2'b10 加密 2'b11 解密
[4:3]	mode	00b	R/W	工作模式选择 2'b00 ECB 2'b01 CBC 2'b10 CTR 2'b11 NONE
[6:5]	key_length	00b	R/W	密钥长度选择 2'b00 128bit 2'b01 192bit 2'b10 256bit 2'b11 NONE
[31:7]	保留	00h	/	返回 0

图 3 **模式寄存器定义**

2.1 AES 算核系统架构

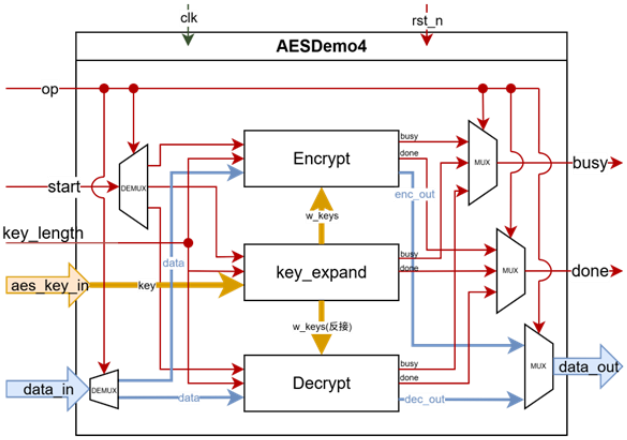


图 4 AES **算核架构图**

如图图 4 所示，本设计采用了可重构的密钥扩展结构和可重构的加密解密模块，高效地实现了多密钥长度支持。工作时，外部通过 op 信号（操作码）控制算核执行密钥扩展、加密或解密中的特定操作，同时通过 key_length 信号选择所需的密钥长度。这种设计实现了硬件单元的高效复用。

2.2 可重构加密解密模块设计

如图加密模块由 convert&add_w_key 模块、round 模块、final_round&reverse_convert 模块、状态机以及相关辅助电路组成。

convert & add_w_key 模块完成输入状态矩阵的转置和工作密钥加运算；

round 模块完成状态矩阵以 S 盒替换、行移位、列混合和工作密钥加为一轮的迭代运算。

final_round & reverse_convert 模块完成状态矩阵以 S 盒替换、行移位和工作密钥加为一轮的迭代运算以及逆转置。

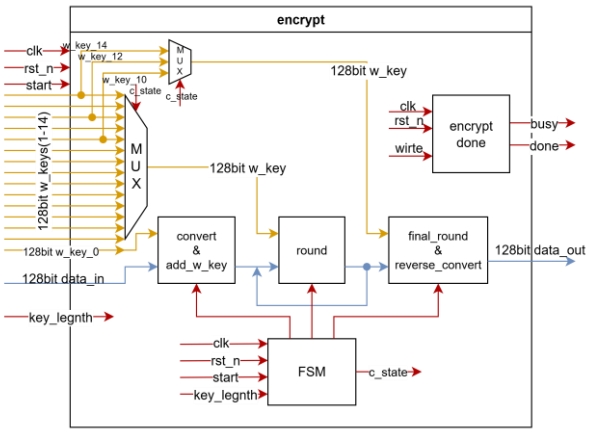


图 5 **可重构加密模块架构图**

如表 1 **AES 对不同密钥长度推荐的迭代轮数**为适配 AES 三种密钥长度所需的不同迭代轮数，实本文设计了一种伸缩状态机：根据 2bit 密钥长度码 key_length，控制状态机在不同迭代轮处跳转，高效地实现了可重构迭代轮数。如图 6。具体而言，消息矩阵经过初始转置和轮密钥加（convert & add_w_key）后，将进行 N－1 次轮函数迭代（N 的取值依密钥长度 key_length 而定），最后经由最终轮处理与逆转置（final_round & reverse_convert）完成加密。同时利用伸缩状态机的灵活性，通过状态绑定机制实现了各轮函数与对应轮密钥的自动匹配。

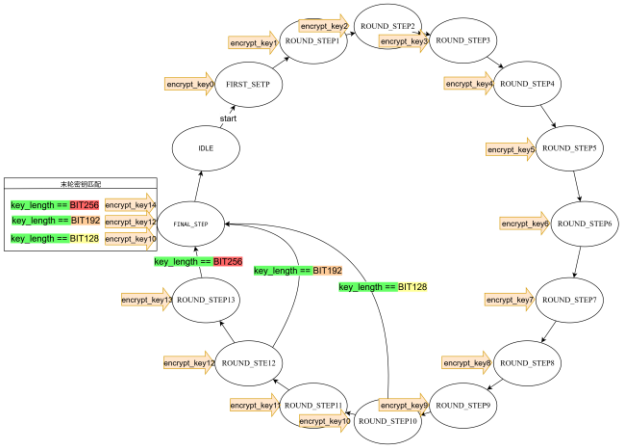


图 6 **可重构加密模块状态机**

表 1 **AES 对不同密钥长度推荐的迭代轮数以及 K、M 值**

密钥长度(bit)	128	192	256
key_length 码	00B	01B	10B
迭代轮数(轮)	10	12	14
M	4	6	8
K	40	48	52

解密模块设计架构与加密模块相似，包括 convert&add_w_key 模块、round_de 模块、final_round_de&reverse_convert 模块、状态机以及相关辅助电路。

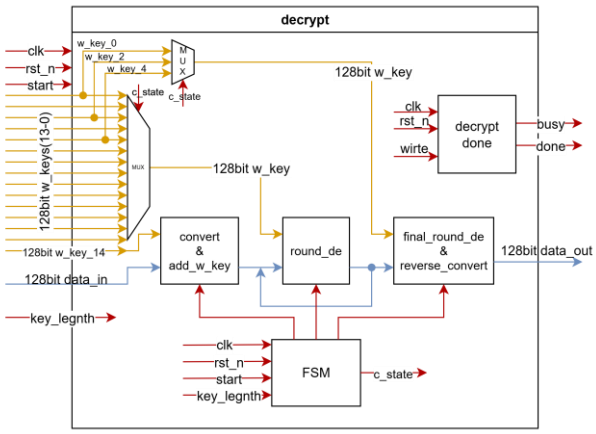


图 7 **可重构解密模块架构图**

解密模块的架构设计与加密设计相同，唯一区别在于逆置的工作密钥匹配方式，如图状态机所示

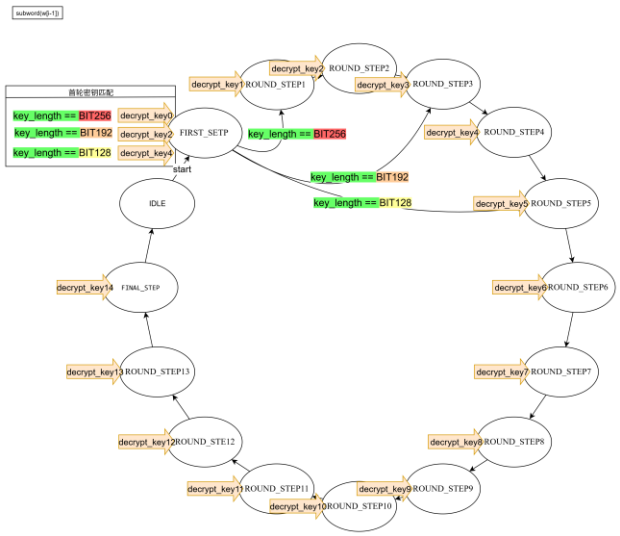


图 8 **可重构解密模块状态机**

2.3 可重构密钥扩展模块设计

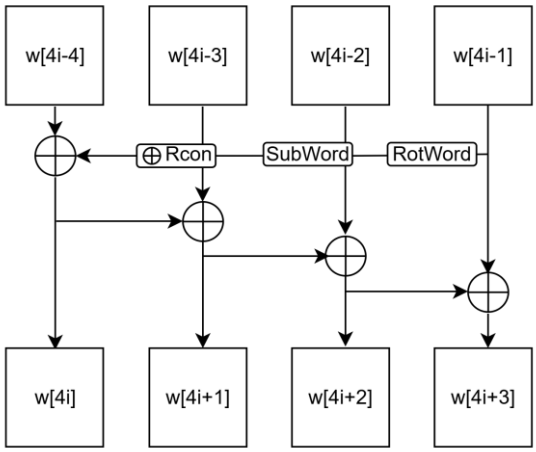


图 9 **AES-128 密钥扩展过程**

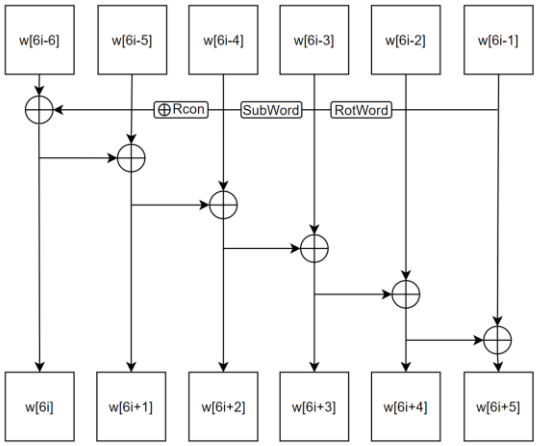


图 10 AES-192 **密钥扩展过程**

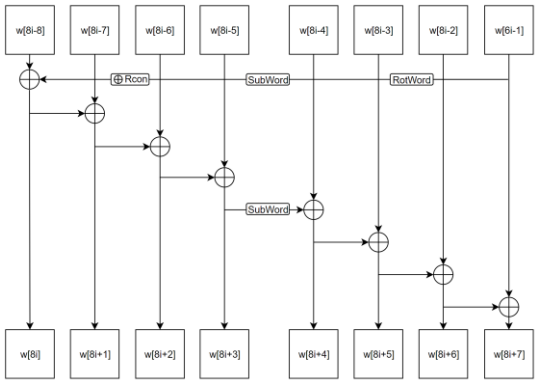


图 11 AES-256 **密钥扩展过程**

根据 AES 生成工作密钥的逻辑，如上三图所示：

1) 首先，AES 将初始密钥输入到一个 4×M(M 由密钥长度决定，键4×4的状态矩阵中，这个 4×M 矩阵的每一列的 4 个字节组成一个字，矩阵 M 列的 4 个字依次命名为

w[0]、w[1]、w[M-1]，它们构成一个以字为单位的数组 w。

2) 接着，对 w 数组扩充 K 个新列，构成总共 k+M 列的扩展密钥数组。对新列 i（M<i<K+M）以如下的递归方式产生：

- 如果 i 不是 M 的倍数，那么第 i 列由如下等式确定：
  - w[i]=w[i-M]⊕w[i-1]
- 如果 i 是 M 的倍数，那么第 i 列由如下等式确定：
  - w[i]=w[i-M]⊕T(w[i-1])

T 函数如下

$$T(w[i-1])=SubWord(RotWord(w[i-1]))\oplus Rcon[i/M]$$

- 如果当前密钥长度为 AES-258，且 i%8 == 4，那么第 i 列由如下等式确定：

w[i]=w[i-8]⊕SubWord(w[i-1])

T 函数中，SubWord、RotWord、⊕Rcon 是字节代换、字循环和轮常量异或，这 3 部分的作用分别如下。

1) RotWord 字循环：将 1 个字中的 4 个字节循环左移 1 个字节。即将输入字[b0, b1, b2, b3]变换成[b1,b2,b3,b0]。
2) SubWord 字节代换：对字循环的结果使用 S 盒进行字节代换。
3) ⊕Rcon 轮常量异或：将前两步的结果同轮常量 Rcon[j]（j=i/M，$1 \le j \le 10$）进行异或，其中 j 表示轮数。

轮常量 Rcon[j]是一个字，其值见下表。

j	1	2	3	4	5
Rcon[j]	01 00 00 00	02 00 00 00	04 00 00 00	08 00 00 00	10 00 00 00
j	6	7	8	9	10
Rcon[j]	20 00 00 00	40 00 00 00	80 00 00 00	1B 00 00 00	36 00 00 00

表 2 轮常量值表

如图 12 密钥扩展模块包括 6 个部分：T 函数、异或
输入滑动窗口寄存器、工作密钥 w 寄存、控制状态机。

本文整合原先三种密钥扩展的数据通路和状态机，三种长度密钥的生成共用一个 T 函数和异或。同时引入移位寄存滑动缓冲机制，避免动态索引大型数组，有效减少了组合逻辑深度，缩短了关键路径延时，提升了时序性能也降低了面积。具体状态机见

与文献 [2] 中“密钥扩展与加密并行”的设计相比，本设计选择了预计算式的密钥扩展策略。文献 [2] 的并行结构虽然节省了部分寄存器资源，但每个轮密钥需 4 个时钟周期生成，使得每个消息块加密过程至少增加 4 个周期延时，对于 10 轮迭代的 AES-128 延迟将提升至 52 拍（原先 12 拍的 4 倍），解密过程需等待全部工作密钥计算完毕，这在晶振不高但需要较高加解密速率的场景中不可接受，如工业摄像机等视频流场景。

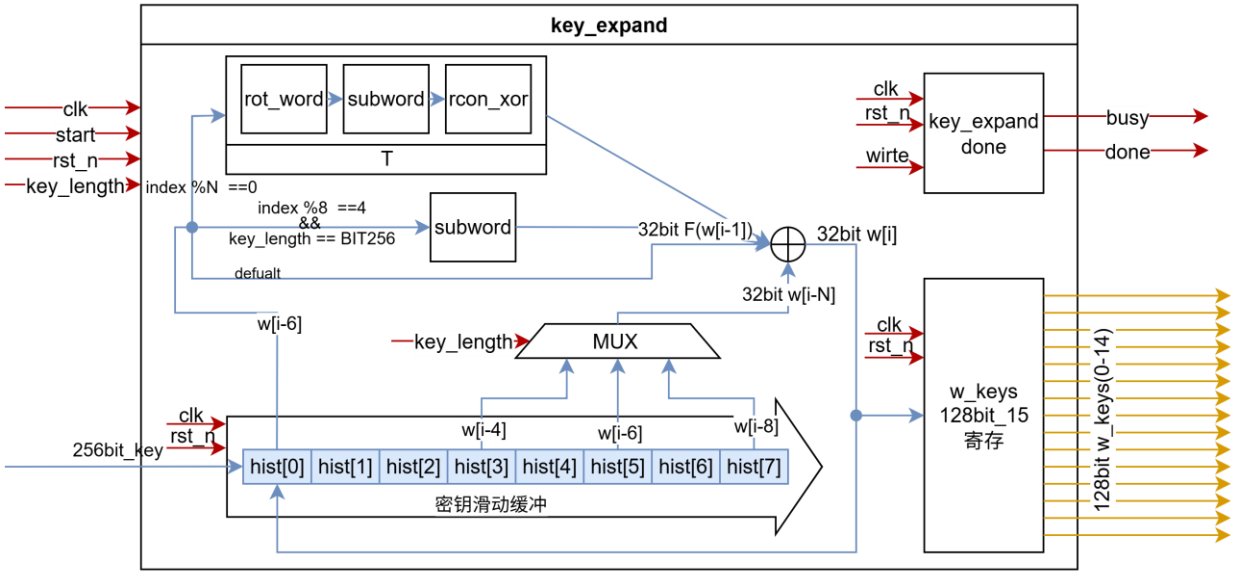


图 12 可重构密钥扩展模块架构图

针对这一问题，本文采用预先生成全部轮密钥并寄存存储的设计，使加解密过程可在每轮仅用一个时钟周期直接访问对应轮密钥。该方法在面积可控的前提下，使密钥扩展仅需执行一次运算，从而在低功耗下和较低的面积代价下显著提升了加解密速率。

图 13 **可重构密钥扩展模块状态机**

3 对比与总结

本文基于 Robei EDA 5.0 工具进行设计，使用搭载 Artix-7 50T 芯片的 RX50T 开发板作为硬件平台。综合布线报告显示，设计资源消耗为 2280 个 SLICE、5558 个 LUT 和 3815 个 FF，最高工作频率为 62 MHz。在开发板 50 MHz 时钟下通过功能测试。

3.1 功能验证

测试数据采用 NIST 标准测试向量，包括明文/密文对、密钥及初始偏移量。本文构建了完整的仿真与实物验证平台，对 ECB、CBC、CTR 三种工作模式分别进行了测试，涵盖 128、192、256 三种密钥长度，测试消息块包括单 128 位块与四 128 位块。具体测试集见下：

表 3 协处理器测试

序号	测试
1	AES-128（4 块）ecb 加密操作测试
2	AES-128（4 块）ecb 解密操作测试
3	AES-128（4 块）cbc 加密操作测试
4	AES-128（4 块）cbc 解密操作测试
5	AES-128（4 块）ctr 加密操作测试
6	AES-128（4 块）ctr 解密操作测试
7	AES-192（4 块）ecb 加密操作测试
8	AES-192（4 块）ecb 解密操作测试
9	AES-192（4 块）cbc 加密操作测试
10	AES-192（4 块）cbc 解密操作测试
11	AES-192（4 块）ctr 加密操作测试
12	AES-192（4 块）ctr 解密操作测试
13	AES-256（4 块）ecb 加密操作测试
14	AES-256（4 块）ecb 解密操作测试
15	AES-256（4 块）cbc 加密操作测试
16	AES-256（4 块）cbc 解密操作测试
17	AES-256（4 块）ctr 加密操作测试
18	AES-256（4 块）ctr 解密操作测试

以 CBC-256 模式下 64 字节消息的测试波形为例，结果如下图所示，与 NIST 样例对比无误。

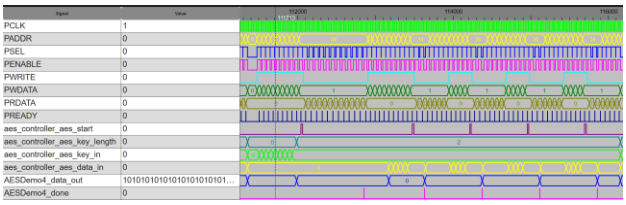
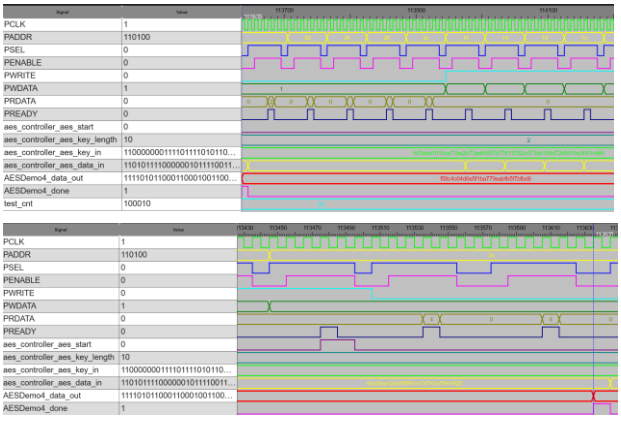


图 14 CBC-256 模式 64 BYTE 测试

图 15 CBC-256 模式 64 BYTE 测试第一 16BYTE 消息块输入、结果波形



3.2 性能测试

板上资源评估在 50 MHz 时钟与 RX50T 约

束条件下，通过 Vivado 综合布线获得，主要评估指标包括 LUT、FF 与 MUX，结果如下：

表 4 资源消耗

LUT	FF	MUX
5558	3815	701

在 50 MHz 时钟下，AES 算核与协处理器的加解密吞吐率如下（单位：Mbps）：

表 5 协处理器和算核吞吐率

密钥长度（bit）	128	192	256
算核（Mbps）	533.3	457．1	400
协处理器（Mbps）	88.9	88.9	88.9

尽管引入了额外的总线控制逻辑，但通过将总线逻辑与工作模式计算逻辑合并优化，使得在低端芯片上的面积开销与文献[1]在高端芯片上的实现面积基本相当。

表 6 与现有成果对比

设计	芯片	密钥长度	工作模式	总线支持	面积（slice）
文献[1]	XC7VX690T	128/192/256	ECB/CBC/CFB/OFB/CTR	无	1947
文献[3]	XC7A100T	128 或 192 或 256	ECB	RoCC/MMIO	-
文献[6]	-	256	CTR	-	-
文	XC7A100T	128/192/256	ECB/CBC/CTR	APB3	2142

文献[1]、[2]与[4]均未集成通用总线，限制了其在实际工业场景中的大规模应用。文献[3]虽支持 RISC-V 扩展指令，但其仅支持 risc-v 生态，架构绑定性强，在当前阶段通用性较弱。

4 结 论

本文设计并实现了一款采用 AMBA APB3 总线接口的可重构 AES 加解密协处理器。该设计通过设置外部寄存器实现工作模式（ECB、CBC、CTR）和密钥长度（128/192/256 位）的在线动态配置，有效提升了 IP 核在物联网终端加密场景中的适应性与复用性，为后续大规模流片和产业应用提供了技术基础。

在验证方面，基于 NIST 标准测试向量构建了完整的仿真与硬件测试平台，验证了多模式、多密钥配置下的功能正确性。综合结果表明，在 50 MHz 时钟下，协处理器实现最高 88.9 Mbit/s 的加解密吞吐率，资源占用为 2142 个 LUT，在 Artix-7 50T 低端 FPGA 上实现的面积与文献[1]在高端 Virtex-7 芯片上的结果相近，展现出优良的结构效率与可移植性。

综上所述，本文实现了一款具备可重构能力、标准化接口与较高资源利用效率的 AES 协处理器，为物联网终端加密模块的工程化与芯片化提供了可行方案。未来工作将聚焦于提升运算吞吐率、优化功耗，并扩展其对多种密码算法（如 SM4、ChaCha20 等）的可重构支持能力。

参考文献

[1] 李炽阳,雷倩倩,杨延飞.全通用 AES 加密算法的 FPGA 实现[J].计算机工程与应用,2020,56(10):83-87
C. Li, Q. Lei, and Y. Yang, "A fully-general FPGA implementation of the AES encryption algorithm," Computer Engineering and Applications, vol. 56, no. 10, pp. 83-87, 2020

[2] 韩少男,李晓江.可兼容 AES-128、AES-192、AES-256 串行 AES 加密解密电路设计[J].微电子学与计算机,2010,27(11):40-4550
S. Han and X. Li, "Design of a serial AES encryption/decryption circuit compatible with AES-128, AES-192, and AES-256," Microelectronics & Computer, vol. 27, no. 11, pp. 40-45, 50, 2010.

[3] GOMES T, SOUSA P, SILVA M, et al. FAC-V: An FPGA-based AES coprocessor for RISC-V[J]. Journal of Low Power Electronics and Applications, 2022, 12(4): 50.

[4] TIRI K, VERBAUWHEDE I. A logic level design methodology for a secure DPA resistant ASIC or FPGA implementation [C] // Proceedings of the Design, Automation and Test in Europe Conference and Exhibition (DATE 2004). 2004, 1: 246-251.

[5] GUPTA M, MAHTO S, PATEL A. Implementation of 128, 192 & 256 bits Advanced Encryption Standard

on Reconfigurable Logic[J]. International Journal of Engineering Trends and Technology, 2017, 50: 305-309.

[6] 刘政文,赵曙光.基于 FPGA 的 AES 加密算法优化设计 [J]. 计算机科学与应用, 2022, 12(5):9.DOI:10.12677/CSA.2022.125138.
LIU Zhengwen, ZHAO Shuguang. Optimized Design of AES Encryption Algorithm Based on FPGA [J]. Computer Science and Application, 2022, 12(5): 9.

饶辅天（本科生），主要研究方向为数字集成电路设计；

孙瑞程（研究生），主要研究方向为数字集成电路设计；

张彬彬（本科生），主要研究方向为数字集成电路设计；

袁海英（副教授），主要研究方向为高性能智能计算芯片与系统

通信作者：饶辅天，常用 3555957586@qq.com。

指导教师 袁海英
参赛队员 饶辅天 张彬彬
杯赛名称 Robei 杯
所获奖项 国赛二等奖
队伍编号 CICC0902558