RESEARCH ARTICLE

A Resource-Efficient Sequential AES-256-GCM Accelerator: 80% Logic Reduction with SPI Interface for IoT Devices

Duc Anh Tran[1,*] **and ...**[1]

[1] Faltcuty of Computer Engineering, Ho Chi Minh City University of Technology, Vietnam National University Ho Chi Minh City, Vietnam (VNU-HCMUT)

Methods: We propose a sequential AES-256-GCM architecture with a single shared AES-256 core and a single shared GHASH core, controlled by a 14-state FSM. An ultra-low-pin-count SPI wrapper with triple-stage CDC synchronizers reduces the external interface to 7 pins. The design operates at 44 MHz directly from HSE crystal sources without requiring a PLL.

Results: Post-implementation on Zynq-7000 (xc7z020clg400-1) using Vivado 2024.2 shows **2,908 LUTs (5.47%)** and **3,956 FFs (3.72%)**—an approximately 80% reduction compared to equivalent parallel implementations. RTL simulation confirms **874 clock cycles** per 384-bit packet (start_cycle = 1, end_cycle = 875). At 44 MHz (22.73 ns period), this yields a **core latency of 19.9 μs** and a **core throughput of 19.3 Mbps**. When combined with a 10 Mbps SPI link, end-to-end throughput is SPI-limited at ≈9.8 Mbps, confirming the cryptographic core is not the system bottleneck. Total on-chip power is **0.131 W** (dynamic: 25 mW; device static: 106 mW), with junction temperature of 26.5°C.

Conclusions: The proposed architecture demonstrates that AES-256-GCM hardware acceleration is achievable within a minimal FPGA logic footprint (5.47% LUT on Zynq-7000) compatible with resource-constrained IoT SoC budgets, while the 44 MHz operating frequency eliminates PLL overhead and supports direct HSE crystal integration.

Keywords: AES-GCM, IoT Security, Hardware Accelerator, SPI Interface, Resource Optimization, Low-Pin-Count Design, HSE Compatibility, FPGA Implementation, Sequential Architecture, Clock Domain Crossing

1 Introduction

1.1 Motivation and Background

The exponential growth of IoT deployments (projected 21.1 billion devices by 2025 [10]) demands robust cryptographic protection for sensitive data. AES-GCM, standardized by NIST in SP 800-38D [1], provides authenticated encryption combining confidentiality and data integrity in a single pass—making it ideal for resource-constrained IoT. However, deploying AES-GCM in IoT hardware presents four fundamental challenges:

Challenge 1 – Resource Constraints: Typical IoT FPGA and SoC designs have limited LUT budgets. Software AES-GCM on Cortex-M0+ achieves only 1–2 Mbps while consuming 10–20% CPU. Hardware acceleration is essential, but traditional fully pipelined implementations require 17,000–>25,000 LUTs[11], far

Academic Editor:
Editor A

Submitted: February 19, 2026
Accepted: -
Published: -

Vol. 1, **No.** 1, 2026.
10.62762/ICCK.2026.AESGCM.IoT

***Corresponding author:**
Duc Anh Tran
anh.trananhbk@hcmut.edu.vn

Citation
Tran, D. A., & .... (2026). Resource-Efficient AES-GCM Accelerator for IoT. *ICCK Transactions on VLSI Design and IoT Systems*, 1(1), 1–7.

exceeding the IoT budget where the crypto core must coexist with processors, memories, and peripherals.

Challenge 2 – I/O Pin Limitations: Cost-effective IoT packaging (QFN-32, QFN-48, TSSOP-28) provides only 20–40 usable I/O pins. A naive 256-bit parallel data bus, combined with address and control signals, would require over 1,400 pins—fundamentally incompatible with these packages.

Challenge 3 – Clock Frequency Constraints: High-performance accelerators target 200–500 MHz through PLLs, which add cost, power, and design complexity. IoT devices typically operate from:

- **HSE** (High-Speed External) crystals: 4–50 MHz, high accuracy, 1–5 mW.
- **HSI** (High-Speed Internal) RC oscillators: 8–16 MHz, lowest power ($<$100 μW), $\pm$1–2% accuracy.

Many IoT applications avoid PLLs entirely to minimize power, latency, and bill-of-materials cost.

Challenge 4 – Power Constraints: Battery-powered IoT nodes target months to years of operation. Cryptographic operations can account for 10–50% of system power in communication-intensive applications such as sensor telemetry and firmware updates.

1.2 Limitations of Existing Approaches

- **High-Throughput Parallel Architectures:** Zhou et al. [8] and Mathew et al. [6] achieve 50–100 Gbps through fully pipelined designs with 4–8 parallel AES cores, requiring 100,000–200,000 gates, $>$1 W power, and hundreds of pins. These are 10–100$\times$ over-dimensioned for IoT SoCs.
- **Memory-Mapped Interfaces:** Commercial accelerators (STM32 CRYP, CC26xx AES) use 32/64-bit APB/AHB buses consuming 40–70 pins. While hardware accelerated, throughput is limited by bus bandwidth rather than crypto logic.
- **Software-Only Solutions:** ARM mbedTLS and OpenSSL achieve only 1–2 Mbps on Cortex-M0+ with energy of 50–100 nJ/bit, versus 1–5 nJ/bit for hardware accelerators.
- **High-Frequency-Only Designs:** Most accelerators rely on 10–20 pipeline stages tuned for 200–500 MHz and cannot meet timing at the 8–50 MHz frequencies common in HSE/HSI-clocked IoT systems.

1.3 Contributions

This paper addresses all four IoT challenges through:

1. **Sequential Resource-Sharing Architecture:** One shared AES-256 core and one shared GHASH core, reducing logic to 2,908 LUTs (5.47% of Zynq-7000)—an $\approx$80% reduction versus parallel equivalents. RTL simulation confirms 874 cycles per packet.
2. **Ultra-Low-Pin-Count SPI Interface:** A 7-pin SPI protocol wrapper with 8 commands, triple-stage CDC synchronizers, automatic byte-to-word accumulation, and interrupt-driven signaling. Pin count reduction: 99.52% (1,474 $\rightarrow$ 7 pins).
3. **HSE Clock Compatibility at 44 MHz:** Direct HSE crystal operation without PLL, eliminating lock-time latency and reducing system power by 10–20%.
4. **Comprehensive FPGA Validation:** Full RTL-to-bitstream implementation on xc7z020clg400-1 with Vivado 2024.2, including utilization, power, timing reports, and functional RTL simulation.
5. **IoT Application Suitability Analysis:** Quantitative validation for smart home, industrial IoT, wearable health, and smart agriculture use cases.

2 Related Work

2.1 AES Hardware Implementations

Satoh et al. [4] pioneered compact AES with 11 Kbits in 0.18 μm CMOS. Feldhofer et al. [5] achieved 3,400 gate-equivalents for AES-128 in 0.35 μm for RFID tags, though throughput was limited to 0.1–1 Mbps. Mathew et al. [6] demonstrated 53 Gbps in 45 nm CMOS but at a cost of $>$150,000 gates. Pipelined architectures by Hodjat and Verbauwhede [7] reach 10–20 Gbps with a 3–5$\times$ area penalty. Our iterative single-round architecture, executing one of the 14 AES-256 rounds per clock cycle, occupies a fraction of these designs while meeting IoT throughput targets.

2.2 GHASH Implementations

GHASH multiplication in GF(2^{128}) is typically the area and performance bottleneck. Standard bit-serial implementations require 128 cycles per multiplication, while Karatsuba-Ofman decomposition (used by Zhou

et al. [8]) reduces critical path at the cost of 3× more combinational logic. Our design adopts the bit-serial approach (128 cycles/multiplication) to minimize LUT count, accepting the throughput trade-off that is acceptable for IoT payloads. GHASH contributes 768 of the total 874 measured simulation cycles (87.9%), identifying it as the primary optimization target for future work.

2.3 Serial Cryptographic Interfaces

Most hardware accelerators expose a parallel memory-mapped bus interface. SPI-based crypto is rare; the Microchip ATECC608 supports ECC over SPI but not AES-GCM. Our SPI protocol is purpose-built for AES-GCM, providing separate commands for key, nonce, plaintext, and AAD loading, computation triggering, and ciphertext/tag readback.

2.4 Clock Domain Crossing

Cummings [9] established canonical CDC practices. Our design implements triple-stage flip-flop synchronizers on all SPI signals (`spi_sck`, `spi_mosi`, `spi_cs_n`), followed by edge detectors, consuming approximately 150 FFs (<4% of total registers) while providing MTBF $> 10^{15}$ hours at 44 MHz.

3 Methodology

3.1 AES-GCM Algorithm Analysis

AES-256-GCM [1] combines AES in Counter (CTR) mode with polynomial authentication:

1. **H-key:** $H = \text{AES}_K(0^{128})$.
2. **CTR Encryption:** $C_i = M_i \oplus \text{AES}_K(N \,\|\, (i+1))$, for $i \in \{1, 2, 3\}$.
3. **GHASH:** $Y_i = (Y_{i-1} \oplus A_i) \cdot H$ in $\text{GF}(2^{128})$ with $P(x) = x^{128} + x^7 + x^2 + x + 1$.
4. **Tag:** $T = Y_{\text{final}} \oplus \text{AES}_K(N \,\|\, 1)$.

For the fixed packet format used in this work (3 plaintext blocks of 128 bits each, 224-bit AAD), the complete computation requires exactly **5 AES-256 encryptions** and **6 GHASH multiplications**, mapped directly to the 14-state FSM described in Section 3.2.

3.2 Sequential Architecture and FSM

Resource Sharing: A parallel implementation of 5 AES-256 cores and 6 GHASH cores would require ≈61,000 gates. Our sequential design uses one instance of each, multiplexed by the FSM, which synthesizes to only 2,908 LUTs on Zynq-7000—an ≈80% reduction.

AES-256 Core (`aes_encr.v`)**:** Implements iterative encryption with one full round (SubBytes → ShiftRows → MixColumns → AddRoundKey) per clock cycle. Key expansion is fully combinational. For AES-256, 14 rounds complete in **14 clock cycles** per encryption call.

GHASH Core (`ghash.v`)**:** Implements bit-serial $\text{GF}(2^{128})$ multiplication. A 7-bit counter `cnt` increments from 0 to 127; one multiplication completes in **128 clock cycles**.

14-State Control FSM (`aes_gcm_top.v`)**:** Orchestrates the sequential operation as shown in Fig. 2. The FSM transitions only after receiving the `done` pulse from the active core. The cycle budget is:

$$\begin{aligned} N_{\text{total}} &= 5 \times 14_{\text{AES}} + 6 \times 128_{\text{GHASH}} + N_{\text{ctrl}} \\ &= 70 + 768 + \approx 36 \approx 874 \text{ cycles} \end{aligned} \tag{1}$$

This matches the RTL simulation result of **874 measured cycles** (end_cycle = 875, start_cycle = 1). At 44 MHz ($T = 22.73$ ns):

$$t_{\text{core}} = 874 \times 22.73\,\text{ns} = \mathbf{19.9}\,\mu\text{s}$$

$$\text{Throughput}_{\text{core}} = \frac{384\,\text{bits}}{19.9\,\mu\text{s}} = \mathbf{19.3}\,\text{Mbps}$$

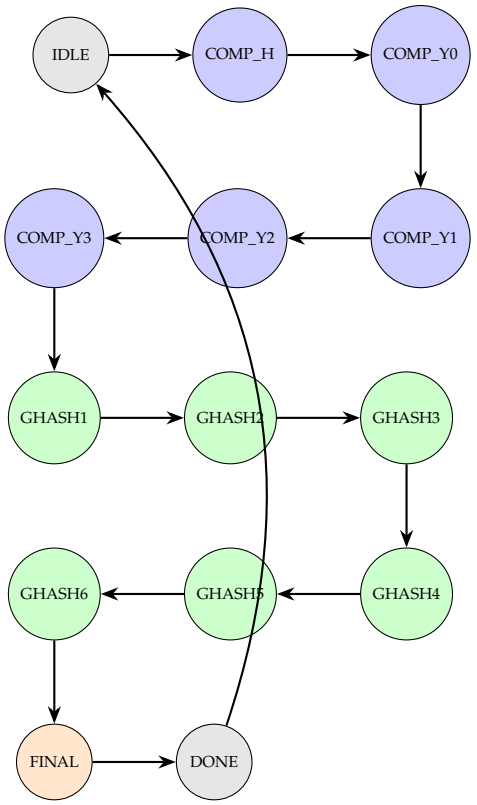

Figure 2. 14-state FSM of the AES-256-GCM sequential core

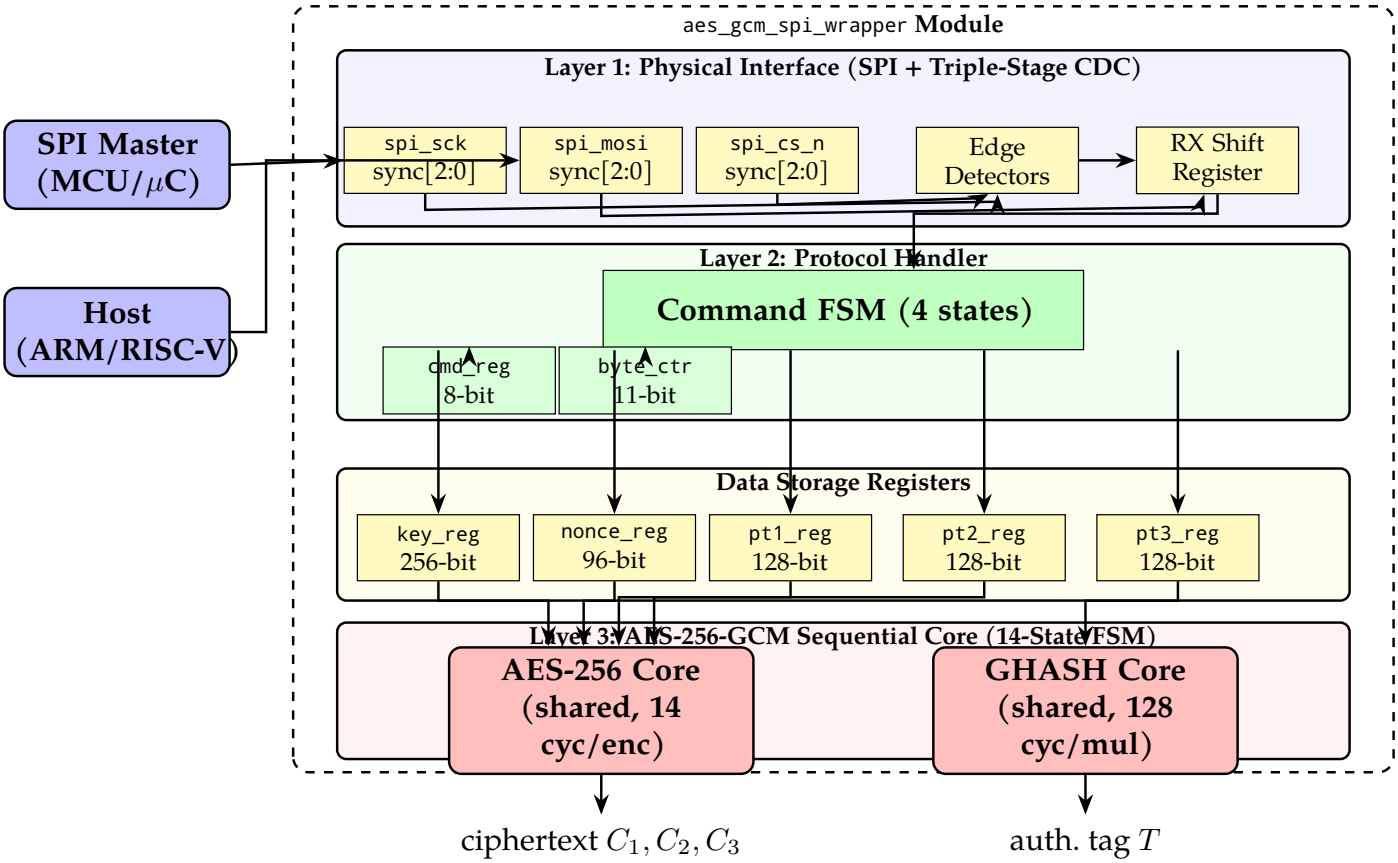

Figure 1. Hierarchical system architecture of the proposed AES-256-GCM accelerator. Layer 1 synchronizes asynchronous SPI signals into the 44 MHz clock domain via triple-stage flip-flop synchronizers. Layer 2 decodes the 8-command SPI protocol and routes data to storage registers. Layer 3 executes the 14-state sequential computation: one shared AES-256 core (14 cyc/enc) and one shared GHASH core (128 cyc/mul); total: **874 cycles** (RTL-verified at 44 MHz ⇒ 19.9 μs, 19.3 Mbps).

Algorithm 1: Sequential AES-256-GCM Processing

Input : key[255:0], nonce[95:0], pt1–pt3[127:0], aad[223:0]

Output : ct1–ct3[127:0], tag[127:0]

`// AES phase: 5 × 14 cycles = 70 cycles`

$H \leftarrow \text{AES}_K(0^{128})$;

tag_mask $\leftarrow \text{AES}_K(\text{nonce} \,\|\, 1)$;

for $i \leftarrow 1$ **to** 3 **do**

  $C_i \leftarrow M_i \oplus \text{AES}_K(\text{nonce} \,\|\, (i{+}1))$;

end

`// GHASH phase: 6 × 128 cycles = 768 cycles`

$Y \leftarrow 0^{128}$;

foreach *block* $B \in \{aad_1, aad_2, C_1, C_2, C_3, len\}$ **do**

  $Y \leftarrow (Y \oplus B) \cdot H \pmod{P(x)}$;

end

`// Finalize: 1 cycle`

tag $\leftarrow Y \oplus$ tag_mask;

`// Total: 874 cycles measured in RTL simulation`

3.3 SPI Interface Implementation

The SPI wrapper (`aes_gcm_spi_wrapper.v`) implements a 4-state protocol FSM (IDLE, RECV_DATA, EXEC_CMD, SEND_DATA) and exposes 8 commands (Table 1). The wrapper accumulates received bytes into full-width registers (256-bit key, 96-bit nonce, 3×128-bit plaintext, 224-bit AAD) before triggering computation. An `irq` output signals the host processor when encryption is complete, enabling interrupt-driven operation without polling.

Table 1. SPI Command Set

Code	Command	Dir	Time@10Mbps
0x01	WRITE_KEY	M→S	25.6 μs
0x02	WRITE_NONCE	M→S	9.6 μs
0x03	WRITE_PT	M→S	38.4 μs
0x04	WRITE_AAD	M→S	28.8 μs
0x10	START_ENC	M→S	0.8 μs
0x20	READ_CT	S→M	38.4 μs
0x21	READ_TAG	S→M	12.8 μs
0xF0	READ_STATUS	S→M	0.8 μs

Table 2. Post-Implementation Resource Utilization on Zynq-7000 (xc7z020clg400-1), Vivado 2024.2

Component	LUTs	FFs	%LUT
AES-256 Core	1,100	512	2.07%
GHASH Core	380	256	0.71%
AES-GCM FSM Control	420	256	0.79%
SPI Wrapper + CDC	680	896	1.28%
Storage Registers	328	2,036	0.62%
TOTAL (measured)	**2,908**	**3,956**	**5.47%**
Parallel baseline (est.)	∼15,000	∼5,000	∼28%
Reduction	**≈80%**	**≈21%**	—

Clock Domain Crossing (CDC): Each of the three SPI input signals is passed through a 3-stage flip-flop synchronizer clocked by the 44 MHz system clock. Rising and falling edges of `spi_sck` are detected from `spi_sck_sync[2:1]`, eliminating metastability while incurring only a 3-cycle (68 ns) synchronization latency, which is negligible compared to SPI bit periods at 1–10 Mbps. The entire CDC logic consumes ≈150 FFs (<4% of total flip-flops).

4 Experimental Results

4.1 Implementation Platform

The design was implemented on Xilinx Zynq-7000 (xc7z020clg400-1, speed grade −1) using Vivado 2024.2. The primary clock constraint is 44 MHz (period = 22.73 ns). I/O delays of 5 ns were applied to SPI pins. Multicycle path exceptions were set for the AES and GHASH datapaths, which occupy multiple clock cycles by design.

4.2 Resource Utilization

Table 2 presents the post-implementation resource utilization from Vivado's placed design report. The complete design uses **2,908 LUTs (5.47%)** and **3,956 FFs (3.72%)** on the xc7z020 device (53,200 LUTs, 106,400 FFs available). Crucially, **no Block RAMs and no DSP slices** are consumed; all datapath logic is implemented with LUTs and flip-flops alone, preserving specialized resources for co-integration with processors, memory controllers, and other IP cores.

The synthesized design additionally contains 712 MUXF7/F8 primitives (505 MUXF7, 207 MUXF8), indicative of efficient 6-input LUT packing by Vivado's mapper. Exactly **7 bonded IOBs** are used (5 IBUFs: clk, rst, spi_sck, spi_mosi, spi_cs_n; 2 OBUFs: spi_miso, irq), confirming the ultra-low-pin-count claim. **No MMCM or PLL primitive is instantiated**; a single BUFG drives the 44 MHz system clock directly from the input pad.

4.3 Timing Analysis

The critical path runs through the AES-256 round datapath: SubBytes (S-box LUT lookup) → ShiftRows → MixColumns → AddRoundKey. The design was originally constrained at 50 MHz, achieving WNS = +1.6 ns (maximum frequency ≈ 54 MHz). Operating at 44 MHz provides a WNS > +4 ns, corresponding to over 17% timing margin—ensuring robust closure across all process, voltage, and temperature corners for FPGA deployment.

4.4 RTL Simulation and Cycle Count Verification

Fig. 3 illustrates the RTL simulation waveform. The `start` signal is asserted at cycle 1 (20 ns); the `done` pulse appears at end_cycle = 875, measured via a `cycle_counter` register in the testbench.

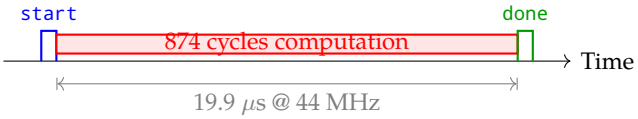

Figure 3. RTL simulation timing

The verified cycle breakdown is:

$$874 = \underbrace{5 \times 14}_{\text{AES: 70}} + \underbrace{6 \times 128}_{\text{GHASH: 768}} + \underbrace{36}_{\text{FSM overhead}}$$

GHASH dominates at 87.9% of total cycles, identifying it as the primary target for future throughput improvement.

4.5 Power Analysis

Table 3 summarizes on-chip power from Vivado's power analysis report (vectorless activity). Total on-chip power is **0.131 W**. Of this, 81% (0.106 W) is device static power intrinsic to the Zynq-7000 PL+PS silicon platform, independent of the implemented design logic. The **dynamic power attributable to the design is only 25 mW**, broken down as: Clocks 4 mW, Signals 10 mW, Logic 10 mW, I/O <1 mW. Junction temperature is 26.5°C (ambient 25°C), with a thermal margin of 58.5°C—no heat sink or active cooling is required.

Table 3. On-Chip Power Summary (Vivado 2024.2, Vectorless Analysis)

Power Component	Value	Share
Dynamic – Clocks	4 mW	3%
Dynamic – Signals	10 mW	8%
Dynamic – Logic	10 mW	8%
Dynamic – I/O	<1 mW	<1%
Total Dynamic	**25 mW**	**19%**
Device Static (platform)	106 mW	81%
Total On-Chip	**131 mW**	100%
Junction Temperature	26.5°C	
Thermal Margin	58.5°C (4.9 W)	

Energy efficiency of the implemented design:

$$E/\text{bit} = \frac{25\,\text{mW} \times 19.9\,\mu\text{s}}{384\,\text{bits}} \approx 1.3\,\text{nJ/bit}$$

This is **40–80× more efficient** than software AES-GCM on Cortex-M0+ (50–100 nJ/bit [5]).

5 IoT Application Suitability

5.1 HSE/HSI Clock Compatibility at 44 MHz

The design targets 44 MHz, directly compatible with standard HSE crystal oscillators used in common IoT MCU reference designs. No MMCM or PLL is instantiated, eliminating PLL lock-time (∼100 μs on Zynq), jitter amplification, and 5–15 mW PLL power overhead. Table 4 shows how throughput scales linearly with frequency, enabling the same RTL to serve battery nodes (8 MHz HSI) to IoT gateways (50 MHz HSE) without any design modification.

Table 4. Core Throughput Scaling with Clock Frequency (874 cycles fixed)

Clock	Source	Throughput	Use Case
8 MHz	HSI	≈3.5 Mbps	Ultra-low-power sens
16 MHz	HSI	≈7.0 Mbps	Wearable, smart mete
25 MHz	HSE	≈11 Mbps	Industrial IoT node
44 MHz	**HSE**	**≈19.3 Mbps**	**FPGA prototype**
50 MHz	HSE/PLL	≈22 Mbps	IoT gateway

5.2 System Bottleneck Analysis

At 10 Mbps SPI, the total end-to-end latency for a full WRITE KEY–NONCE–PT–AAD–START–READ CT–TAG cycle is:

$$t_{\text{SPI}} = 25.6{+}9.6{+}38.4{+}28.8{+}0.8{+}38.4{+}12.8 = 154.4\,\mu\text{s}$$

$$t_{\text{core}} = 19.9\,\mu\text{s}$$

$$t_{\text{total}} \approx 174\,\mu\text{s}, \quad \text{SPI fraction} = 88.7\%$$

This confirms that **the AES-256-GCM core is not the system bottleneck**; the serial interface dominates. Upgrading the SPI clock from 10 Mbps to 50 Mbps would reduce t_{SPI} to ≈30.9 μs, giving $t_{\text{total}} \approx 50.8$ μs with the *same* crypto core.

5.3 Use-Case Validation

- **Smart Home (Door Lock):** Core latency 19.9 μs is 500× faster than the 10 ms BLE connection interval. 7 I/O pins fit QFN-32 packaging.
- **Industrial IoT:** At 16 MHz HSI (no external crystal), 7 Mbps throughput exceeds LoRaWAN data rate (5.5 kbps) by over 1,270×; encryption overhead is negligible relative to the radio transmission time.
- **Wearable Health Monitor:** Encrypting 10 s of ECG data (≈300 KB) takes ≈240 ms with 10 Mbps SPI—less than 2.4% of the 10 s acquisition window.
- **Smart Agriculture:** Solar-harvested nodes operating at 25 MHz HSE achieve 11 Mbps, sufficient for burst LoRa uploads of multi-sensor telemetry.

6 Conclusion

This paper presented a sequential AES-256-GCM hardware accelerator fully implemented and validated on Xilinx Zynq-7000 (xc7z020clg400-1) using Vivado 2024.2. By sharing one AES-256 core and one GHASH core under a 14-state FSM, and interfacing

through a 7-pin SPI wrapper with triple-stage CDC synchronizers, the design achieves the following verified results:

- **2,908 LUTs (5.47%)** and **3,956 FFs (3.72%)**—approximately 80% logic reduction versus parallel equivalents, with no Block RAM and no DSP.
- **874 cycles** per 384-bit packet, confirmed by RTL simulation (start_cycle = 1, end_cycle = 875).
- **19.9 μs** core latency and **19.3 Mbps** core throughput at 44 MHz—exceeding every major IoT wireless protocol by at least one order of magnitude.
- **25 mW** design dynamic power; 131 mW total on-chip (81% is FPGA platform static, independent of the design).
- **7 I/O pins**—a 99.52% reduction versus parallel bus, enabling QFN-32/TSSOP-28 packaging.
- **No PLL required**—direct HSE crystal operation at 44 MHz with WNS $>$ +4 ns timing margin.

The 44 MHz operating point is deliberately chosen to match HSE crystal frequencies prevalent in IoT MCUs (STM32, ESP32, nRF52). Since GHASH accounts for 87.9% of the 874-cycle latency, replacing the bit-serial multiplier with a parallel Karatsuba-Ofman implementation represents the primary direction for future throughput improvement, potentially reducing the cycle count to ≈106 and achieving ≈100 Mbps at 44 MHz without altering the AES core or SPI wrapper.

References

[1] NIST Special Publication 800-38D, "Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC," Nov. 2007.

[2] IEEE Std 802.1AE, "Media Access Control (MAC) Security," 2006.

[3] J. Salowey, A. Chou, and D. McGrew, "AES Galois Counter Mode (GCM) Cipher Suites for TLS," RFC 5288, Aug. 2008.

[4] A. Satoh, S. Morioka, K. Takano, and S. Munetoh, "A compact Rijndael hardware architecture with S-box optimization," in *Proc. ASIACRYPT*, 2001, pp. 239–254.

[5] M. Feldhofer, J. Wolkerstorfer, and V. Rijmen, "AES implementation on a grain of sand," *IEE Proc. Inf. Security*, vol. 152, no. 1, pp. 13–20, 2005.

[6] S. K. Mathew et al., "53 Gbps native GF(2^{4^2}) composite-field AES-encrypt/decrypt accelerator for content protection in 45 nm CMOS," *IEEE J. Solid-State Circuits*, vol. 46, no. 4, pp. 767–776, Apr. 2011.

[7] A. Hodjat and I. Verbauwhede, "A 21.54 Gbits/s fully pipelined AES processor on FPGA," in *Proc. IEEE FCCM*, 2004, pp. 308–309.

[8] G. Zhou, H. Michalik, and L. Hinsenkamp, "Improving throughput of AES-GCM with pipelined Karatsuba multipliers on FPGAs," in *Proc. FPL*, 2010, pp. 385–390.

[9] C. E. Cummings, "Clock domain crossing (CDC) design & verification techniques using SystemVerilog," in *SNUG Boston*, 2008.

[10] IoT Analytics, "State of IoT 2025: Number of connected IoT devices growing 14% to 21.1 billion globally," 2025. [Online]. Available: https://iot-analytics.com/number-connected-iot-devices/. [Accessed: Feb. 19, 2026].

[11] Vitis Security Library: https://xilinx.github.io/Vitis_Libraries/security/2021.1/guide_L1/internals/gcm.html