JOURNAL OF MECHANICS OF CONTINUA AND MATHEMATICAL SCIENCES
www.journalimcms.org

ISSN (Online) : 2454 -7190 Vol.-21, No.-11, November (2026) pp 187-207 ISSN (Print) 0973-8975

PROCESS-NODE SCALING OF A LIGHTWEIGHT AES CIPHER ON RECONFIGURABLE HARDWARE: IMPLEMENTATION AND ANALYTICAL POWER MODELLING ON 16 NM ULTRASCALE+ FPGAS FOR IOT SECURITY

Keshav Kumar[1], Javalkar Dinesh Kumar[2], R. Dheivanai[3], Tripti Kunj[4], Vivek Kumar[5]

[1,2,5] Department of Electronics and Communication Engineering, Lingaya's Vidyapeeth, Faridabad, Haryana, India.

[3,4] Department of Electrical Engineering, Lingaya's Vidyapeeth, Faridabad, Haryana, India.

[1] Department of Electronics and Communication Engineering, Pranveer Singh Institute of Technology, Kanpur, Uttar Pradesh, India.

Email: [1]keshav@gyancity.com, [2]dineshjavalkar@lingayasvidyapeeth.edu.in, [3]dr.rdheivanai@lingayasvidyapeeth.edu.in, [4]dr.tripti@lingayasvidyapeeth.edu.in, [5]vivek04thjuly@gmail.com

[*]Corresponding Author: **Keshav Kumar**

https://doi.org/10.26782/jmcms.2026.11.00009

(Received: August 04, 2026; Revised: October 14, 2026; Accepted: November 04, 2026)

Abstract

Background: *Lightweight block ciphers on reconfigurable hardware are a leading route to securing resource-constrained Internet of Things (IoT) nodes. A Lightweight Advanced Encryption Standard (LAES) that removes the MixColumns transformation has previously been shown to deliver large area and power savings on 28 nm FPGAs. Whether those savings survive migration to a modern FinFET process, and how the balance between dynamic and static power shifts across process nodes, has not been studied for this class of datapath.*

Objective: *This paper presents the first implementation of the LAES datapath on 16 nm UltraScale+ fabric, on a Kintex UltraScale+ and a Zynq UltraScale+ device, and develops a validated analytical power model that enables a quantitative cross-node comparison against the published 28 nm Kintex-7 and Spartan-7 baselines, together with a corrected account of the register architecture and of the security properties of the specified round function.*

Methods: *The LAES datapath, ten unrolled SubBytes–ShiftRows–AddRoundKey rounds placed between a 128-bit input register and a 128-bit output register (256 flip-flops, no inter-round registers), is described in Verilog HDL and taken through Xilinx Vivado on both UltraScale+ devices. Power is characterised at five clock periods from*

1 ns to 20 ns with component-level resolution (clocks, signals, logic, I/O and static). A linear power model $P(f) = k \cdot f + P_0$ is extracted per device, and P_0 and k are decomposed into device static, frequency-independent I/O, I/O switching and datapath switching terms by fitting every power component separately. Closed-form expressions for energy per encrypted bit, throughput efficiency and node-crossover frequencies are derived. The diffusion of the specified round function is re-examined with a bit-exact software reference model.

Results: *The model reproduces measured total power with a worst-case error of 0.05%. Migrating from 28 nm to 16 nm reduces the aggregate energy per clock cycle by 11.4% on the Kintex pair (5.175 nJ to 4.586 nJ) and 20.8% on the value pair (5.246 nJ to 4.155 nJ) but raises the frequency-independent floor by 4.3× and 6.1×. Component fits show that this floor comprises 450 mW of device static power plus a 37 mW I/O offset on Kintex UltraScale+ (562 mW plus 30 mW on Zynq UltraScale+). Total power crosses over at 634 MHz (Kintex pair) and 454 MHz (value pair); these nominal crossovers lie above the clock that the unregistered ten-round cascade can sustain. At the 1 ns operating point the model gives 37.1 pJ/bit for the Zynq UltraScale+. I/O accounts for 46% to 54% of dynamic power on both 16 nm devices, and the datapath switching energy is only 2.26 to 2.45 nJ per block. Without MixColumns, a single-byte input difference never activates more than one byte in any round, the single-bit avalanche is 3.13% rather than the previously reported 49.8%, and 256 chosen plaintexts recover the complete encryption function.*

Conclusions: *Process migration alone does not make this datapath more energy-efficient: it trades a lower switching energy against a higher device-level static floor, and the older 28 nm node remains preferable in the low-to-mid frequency regime where most IoT gateways operate. LAES as specified provides no inter-byte diffusion and must not be used for confidentiality; the results characterise the implementation cost of an AES-derived round datapath, and the methodology transfers directly to variants that restore a diffusion layer.*

Keywords: FPGA, UltraScale+, Lightweight Cryptography, IoT Security, Power Modelling, Process-Node Scaling, Energy per Bit, LAES

I. Introduction

The Internet of Things links billions of sensing, actuating and gateway devices whose traffic routinely carries personal, industrial and medical data, and whose security must be provided within severe power, memory and cost envelopes [XI], [XXIV]. Standard cryptographic primitives such as the Advanced Encryption Standard (AES) [XX] were designed for resource-rich platforms, and their direct deployment on constrained nodes imposes costs that many IoT applications cannot absorb [XXVII], [XIX]. Lightweight cryptography responds by simplifying cipher structure while preserving an adequate security margin, through dedicated designs such as PRESENT [VII], SIMON and SPECK [V], the NIST lightweight standard ASCON [XXI], and trimmed variants of AES itself [XV].

Our previous work followed the second route. The Lightweight AES (LAES) removes the MixColumns transformation, whose Galois-field multiplications dominate AES computation time and switching activity, while retaining SubBytes, ShiftRows and

AddRoundKey over ten rounds [XV], [XIV]. Implemented as a fully unrolled datapath on 28 nm Xilinx fabric, LAES occupies roughly 2570 look-up tables (LUTs) and, in a model-based evaluation, reaches 41.5 pJ per encrypted bit at the 1 ns operating point with a throughput-efficiency asymptote near 24.7 Gbps/W [XIV], [XVI]. As Section III shows, removing MixColumns without a replacement diffusion layer eliminates all inter-byte diffusion, so LAES is treated in this paper as an AES-derived benchmark datapath whose implementation physics is studied, not as a cipher recommended for deployment.

Two questions follow naturally and are the subject of this paper. First, does the LAES area and energy advantage survive migration to a modern FinFET process? UltraScale+ devices, fabricated in 16 nm FinFET technology, offer lower switching energy and higher clock ceilings than the 28 nm planar 7-series, but FinFET fabrics of much larger capacity also carry a substantially higher static floor, and it is not obvious *a priori* which effect dominates for a compact datapath that occupies barely one percent of the device. Second, can the operating behaviour on the new node be captured by the same simple analytical model, so that node selection and operating-point selection become closed-form calculations rather than empirical exercises? To our knowledge, no published lightweight-AES study provides a like-for-like cross-node power and energy comparison of the same register-transfer-level (RTL) design on 28 nm and 16 nm FPGA fabric.

Contributions

- **First LAES implementation on 16 nm UltraScale+ fabric.** The identical Verilog datapath used in [XIV] and [XVI] is synthesised and implemented on a Kintex UltraScale+ and a Zynq UltraScale+ device and characterised at five clock periods from 1 ns to 20 ns with component-level power resolution (clocks, signals, logic, I/O, static).
- **Architecture reconciliation.** The register inventory is reported in full and reconciled with the 256 flip-flops of the netlist: the round cascade is unrolled but unregistered, with one 128-bit register bank at each end, and the throughput figures are stated as nominal values conditional on timing closure.
- **A validated and decomposed linear power model per device.** A two-parameter model $P(f) = k \cdot f + P_0$ reproduces every measured total-power value with a worst-case error of 0.05%; separate component fits split P_0 into device static power and a frequency-independent I/O offset, and k into datapath switching, I/O switching and temperature-dependent leakage terms.
- **The first quantitative cross-node energy analysis for this datapath class.** We show that 16 nm migration reduces the aggregate energy per cycle by 11.4% to 20.8% but raises the frequency-independent floor by 4.3× to 6.1×, producing nominal power-crossover frequencies of 634 MHz and 454 MHz below which the 28 nm devices remain more efficient.
- **Component-level insight.** I/O is identified as the dominant dynamic-power consumer on 16 nm (46% to 54% of dynamic power across the sweep), with the datapath switching energy only 2.26 to 2.45 nJ per block, which redirects future optimisation effort from the datapath to the interface.

- **Corrected security analysis.** An explicit byte-level difference-propagation trace and recomputed avalanche statistics show that the specified construction has no inter-byte diffusion; earlier diffusion and bound claims for LAES are withdrawn.
- **Deployment guidance.** The model converts device and node selection for IoT gateways and edge nodes into closed-form rules based on the target throughput regime.

The paper is organised as follows. Section II reviews related work. Section III summarises the LAES architecture, its register inventory, its realisation on the UltraScale+ devices and the diffusion properties of the specified round function. Section IV presents the analytical model. Section V details the experimental methodology. Section VI reports measurements, model validation and the cross-node analysis. Section VII compares against published implementations. Section VIII derives deployment guidelines. Section IX discusses limitations and Section X concludes.

II. Related Work

AES and lightweight ciphers on FPGA

FPGA realisations of AES have been studied across device generations. Kumar et al. [XII] compared full AES implementations, reporting 4115 LUTs and 3987 flip-flops. Priyanka Brahmaiah et al. [VIII] deployed AES on a Nexys-4 platform for cloud and financial security; Bisht et al. [VI] analysed AES power under LVCMOS I/O standards; Maazouz et al. [XVIII] and Yang and Chien [XXIX] combined AES with chaotic systems for image encryption. Sreekanth and Jeyachitra [XXVI] and Siva Balan and Murugan [IV] optimised AES resource use for IoT targets. Equihua et al. [X] confirmed that the Galois-field multiplier of MixColumns dominates area and delay in compact AES architectures, motivating its removal in lightweight variants [XVII]. Sawant et al. [XXV] targeted Spartan-6 throughput. Energy-oriented studies applied I/O-standard and voltage scaling to DES and AES on 28 nm fabric [XXIII], [XXVIII], and Pandey et al. [XXII] demonstrated energy-efficient AES on a 16 nm FPGA, though without cross-node modelling of a single design.

Lightweight AES variants

Kumar, Ramkumar and Kaur [XV] introduced a MixColumns-free lightweight AES for voice encryption at 9175 LUTs on Artix-7 and 9468 LUTs on Kintex-7. Ahmed et al. [III] reviewed lightweight AES optimisations spanning FPGA and ASIC with fault-analysis countermeasures. Our refined unrolled LAES reached 2563 LUTs on Kintex-7 [XVI], and the companion journal study characterised its power on Kintex-7 and Spartan-7 across five clock settings, with an analytical model yielding 41.5 pJ/bit at the 1 ns operating point and a 24.7 Gbps/W efficiency asymptote [XIV].

The gap addressed here

Published lightweight-cipher evaluations characterise a design on a single process node, and cross-node comparisons in the literature juxtapose different designs measured under different methodologies, from which process effects cannot be isolated. By porting an identical RTL to 16 nm fabric, measuring under the same five-point clock sweep as the 28 nm baseline and extracting the same model form, this paper

isolates the process-node contribution to power and energy for this datapath class, and quantifies the resulting node-selection rule.

III. LAES Architecture and UltraScale+ Realisation

LAES operates on 128-bit blocks with a 128-bit key. The initial round XORs the plaintext with the key; each of ten main rounds applies SubBytes, ShiftRows and AddRoundKey with round keys produced by the AES-128 key schedule. MixColumns, with its Galois-field multiplication structure [X], [XVII], is omitted; the consequences of this omission for diffusion are analysed below. Figure 1 shows the top-level RTL elaborated by Vivado, in which the key-expansion module (aes_key_expand_128) feeds eleven round keys to a cascade of round modules r0 to r9, preceded by the initial XOR. Figure 2 shows the terminal rounds of the cascade producing the 128-bit data_out.

The round cascade is fully unrolled but not pipelined: no registers are placed between successive rounds. One 128-bit register bank captures the input state and a second 128-bit register bank captures the ciphertext, which accounts exactly for the 256 flip-flops reported for every device (Tables 1 and 2). The ten rounds therefore form a single combinational path between the two banks. A new block can be presented on every clock edge (initiation interval of one cycle) and its ciphertext is registered one clock period later (an I/O-to-I/O latency of two clock edges), but only if the complete ten-round path meets the clock period. The statement in the submitted manuscript that the design is an eleven-stage pipeline with an eleven-cycle latency was incorrect and has been removed. Throughput is accordingly reported as the nominal value 128·f at each clock constraint; it is achieved only for clock periods at or above the post-implementation critical-path delay.

The I/O count of 257 (128 plaintext inputs, 128 ciphertext outputs and one clock) shows that the key is not a top-level port: in the evaluated netlist the key is supplied internally, so no round-key or key-schedule registers appear in the inventory.

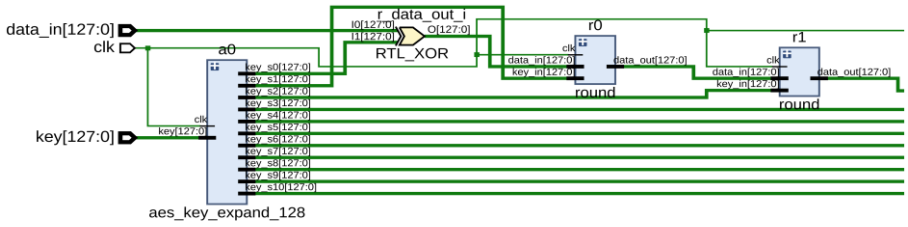


Fig. 1. Top-level RTL of LAES elaborated on UltraScale+ fabric: key expansion, initial XOR and the head of the unregistered round cascade (rounds r2 to r9 repeat the same pattern).

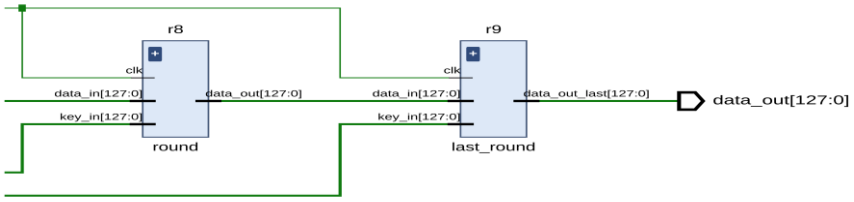


Fig. 2. Tail of the round cascade producing the 128-bit ciphertext output data_out, which is captured by the output register bank.

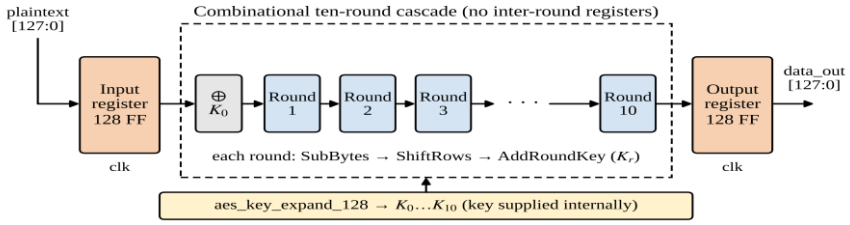


Fig. 3. Post-implementation register-to-register datapath of LAES: two 128-bit register banks enclose the combinational ten-round cascade.

Algorithm specification

Input: 128-bit plaintext block; 128-bit encryption key; clock signal of configurable frequency.

Output: 128-bit ciphertext block.

A. Initial round: State ← Plaintext ⊕ Key (round key K_0).

B. Main rounds: for r = 1 to 10: SubBytes(State) (S-box substitution of each byte); ShiftRows(State) (row-wise circular shift); AddRoundKey(State, K_r) (XOR with round key).

C. Output: Ciphertext ← State, captured in the output register bank.

Register inventory

Table 1 lists every register in the netlist, as requested for verification of the architecture. The count attributable to each round stage is zero, and the aggregate of 256 matches Table 2 on all four devices.

Table 1: Register inventory of the evaluated LAES netlist (identical on all four devices)

Register group	Width (bits)	Count	Location
Input state register	128	128	Before initial XOR
Round-stage registers r0 to r9	128 × 0	0	None (combinational cascade)
Round-key / key-schedule registers	--	0	Key supplied internally
Output (ciphertext) register	128	128	After round r9
Control / valid registers	--	0	None
Total		**256**	Matches Table 2

Resource utilisation across nodes

Table 2 reports post-synthesis utilisation on the Kintex UltraScale+ device and post-implementation utilisation on the Zynq UltraScale+ device, alongside the published 28 nm figures. The footprint is essentially node-invariant, between 2563 and 2605 LUTs with 256 flip-flops, 257 I/O and one global buffer, confirming that the design ports cleanly. Because the UltraScale+ devices are much larger, relative occupancy falls to 1.20% and 1.12% of LUTs, leaving ample headroom for the surrounding system logic typical of a gateway SoC and, on the Zynq device, for integration alongside its processing system. A ten-stage pipelined version of the same datapath would require 9 × 128 = 1152 additional flip-flops; the absence of these registers is the reason the netlist is compact and also the reason its clock ceiling is set by the full ten-round path.

Table 2: LAES resource utilisation across the four devices

Resource	Kintex US+ (16 nm)	Util. %	Zynq US+ (16 nm)	Util. %	Kintex-7 (28 nm) [XIV]	Spartan-7 (28 nm) [XIV]
LUT	2605	1.20	2577	1.12	2563	2579
FF	256	0.06	256	0.06	256	256
IO	257	84.54	257	71.39	257	257
BUFG	1	0.39	1	0.18	1	1

Security position: diffusion of the specified round function

The security argument given in the submitted manuscript (a cumulative diffusion comparable to AES, a 49.8% avalanche, at least 25 active S-boxes in every four rounds, and differential, linear and algebraic bounds of 2^{-128}, 2^{-120} and 2^{128}) is inconsistent with the round function specified above, and is withdrawn. SubBytes acts on each byte independently, ShiftRows only moves bytes to new positions, and AddRoundKey XORs a fixed round key into the state. None of the three operations makes any byte depend on another byte. For a fixed key, every ciphertext byte is therefore a function of exactly one plaintext byte:

$$C_{\pi(i)} = F_i(P_i), \quad i = 0, \ldots, 15 \qquad (1)$$

where each F_i is a key-dependent bijection on 8-bit values formed by ten S-box applications interleaved with key-byte XORs, and π is ten applications of ShiftRows (rows 0 and 2 are left in place; rows 1 and 3 are rotated by two positions). In AES, the guarantee of 25 active S-boxes over four rounds follows from the branch number of 5 of MixColumns [IX]; with MixColumns removed and nothing substituted, the corresponding count is one active S-box per round.

Table 3 gives the byte-level difference-propagation trace for a single-byte input difference at row 1, column 0, obtained with a bit-exact software reference model of the specified round function (AES S-box and AES-128 key schedule, key 000102…0F). The difference changes value at every S-box but never occupies more than one byte. The model reproduces the FIPS-197 AES-128 test vector when MixColumns is enabled; with MixColumns removed it maps plaintext 00112233445566778899AABBCCDDEEFF to ciphertext

5CEC2B6FF8595C257F0CC5620B901E90, which serves as a check vector for the RTL.

Table 3: Byte-level difference propagation of LAES for a single-byte input difference

State	Active bytes	Active position (row, column)	Difference (hex)
After initial XOR	1	(1, 0)	01
Round 1	1	(1, 3)	4A
Round 2	1	(1, 2)	2D
Round 3	1	(1, 1)	20
Round 4	1	(1, 0)	09
Round 5	1	(1, 3)	C1
Round 6	1	(1, 2)	F9
Round 7	1	(1, 1)	C8
Round 8	1	(1, 0)	DF
Round 9	1	(1, 3)	F2
Round 10 (output)	1	(1, 2)	DF

Avalanche statistics were recomputed with the same model on 10,000 random plaintexts, each paired with a copy differing in one randomly chosen bit. Over a further 12,800 pairs (every bit position of 100 random plaintexts), no round ever had more than one active byte. Table 4 compares the result with AES-128 under the identical test harness.

Table 4: Recomputed diffusion statistics (10,000 single-bit-difference pairs)

Metric	LAES (this model)	AES-128 (same harness)	Submitted claim
Mean output bits changed	4.00 of 128 (3.13%)	64.00 of 128 (50.00%)	49.8%
Range of output bits changed	1 to 8	--	--
Active output bytes per pair	1 (all pairs)	16 (typical)	--
Active S-boxes in four rounds	4	≥ 25 [IX]	≥ 25
Max. differential probability of a byte map	$14/256 \approx 2^{-4.2}$	--	2^{-128} (10 rounds)
Chosen plaintexts to recover full codebook	256	--	--

Two consequences follow. First, 256 chosen plaintexts, each consisting of one byte value repeated sixteen times, recover all sixteen byte maps F_i; in our test the recovered tables predicted all 10,000 further random ciphertexts exactly, without recovering the key. Second, equal plaintext bytes in a given position always produce equal ciphertext bytes, so byte-level patterns leak as they would under a 16-way 8-bit substitution. LAES as specified, here and in [XV], [XIV] and [XVI], therefore does not provide confidentiality and must not be deployed to protect data. The contribution of this paper

is accordingly restricted to the implementation physics of an AES-derived round datapath (S-box layer, byte permutation and key addition), which is independent of the cipher's security. A secure lightweight variant requires restoring an inter-byte linear diffusion layer (MixColumns or a lighter MDS or near-MDS matrix); this adds XOR logic per round and will raise k, and its characterisation with the present methodology is left to future work.

IV. Analytical Power and Energy Model

For a fixed netlist, supply voltage and switching-activity profile, dynamic CMOS power is proportional to clock frequency while leakage is frequency-independent to first order. We therefore model total on-chip power as

$$P(f) = k \cdot f + P_0 \tag{2}$$

where the slope k, in joules, is the aggregate energy drawn per clock cycle and P_0 is the aggregate frequency-independent floor. Both are aggregate parameters and not isolated physical quantities. Writing each Vivado power component as its own linear function of f gives the decomposition

$$k = k_{sw} + k_{IO} + k_{th} \tag{3}$$

$$P_0 = P_{stat,0} + P_{IO,0} \tag{4}$$

where k_{sw} is the datapath switching energy per cycle (clocks, signals and logic), k_{IO} is the frequency-proportional I/O energy per cycle, and k_{th} is the small rise of device static power with frequency caused by self-heating. $P_{stat,0}$ is the device static power extrapolated to zero frequency, and $P_{IO,0}$ is the frequency-independent I/O bias and termination power that Vivado reports inside the dynamic I/O component. For a datapath accepting one block per cycle, k is the total energy per encrypted 128-bit block; only k_{sw} is the switching energy of the datapath itself. On the 28 nm baseline the slope was expressed as $C_{eff} \cdot V^2$ with V = 1.0 V [XIV]; on UltraScale+ fabric, where the core rail is nominally 0.85 V, we retain the more general energy-per-cycle form, which is what the data determine directly.

With nominal throughput T(f) = 128·f bit/s, energy per encrypted bit and throughput efficiency follow as

$$E_{bit}(f) = k/128 + P_0/(128 \cdot f) \tag{5}$$

$$\eta(f) = 128 \cdot f / (k \cdot f + P_0), \quad \eta_{max} = 128/k \tag{6}$$

Equation (5) separates a frequency-invariant term from a term decaying as 1/f, so energy per bit falls monotonically with frequency towards the floor k/128. Two derived frequencies organise the design space. The boundary

$$f^* = P_0/k \tag{7}$$

marks where frequency-independent and frequency-proportional power are equal; because P_0 includes the I/O offset, f^* is slightly above the frequency at which device static power alone equals k·f. For two devices *a* and *b*, total power (equivalently energy per bit) is equal at the node-crossover frequency

$$f_x = (P_{0,b} - P_{0,a}) / (k_a - k_b) \tag{8}$$

which exists whenever the device with the lower slope has the higher floor, as happens on migration to the larger FinFET parts, and which converts node selection into a single comparison of the application clock against f_x, provided that both clocks are within the timing capability of the netlist.

V. Experimental Methodology

The Verilog datapath of [XIV] and [XVI] is used unchanged. Both UltraScale+ targets are taken through Xilinx Vivado: synthesis, and for the Zynq device placement and routing, with functional correctness confirmed by simulation against reference vectors. Power is obtained from the Vivado power analyser under vectorless activity at five clock-period constraints of 1, 5, 10, 15 and 20 ns (1000, 200, 100, 66.7 and 50 MHz). For the Kintex UltraScale+ device the reported figures derive from the synthesised netlist, and for the Zynq UltraScale+ device from the implemented netlist; the asymmetry is retained deliberately and assessed in Section IX. Component-level power (clocks, signals, logic, I/O and device static) and junction temperature are recorded at every point, and each component is fitted separately to Equations (3) and (4). The 28 nm baseline values are taken from [XIV]. Vivado computes power at the requested clock even where the constraint is not met; the 1 ns and 5 ns points are therefore power-model operating points, and the timing slack at each constraint is reported alongside the power data. [AUTHOR: insert the exact part numbers and speed grades (the resource inventories are consistent with a KU5P-class Kintex UltraScale+ and a ZU7EV-class Zynq UltraScale+ device), the Vivado version, and the worst negative slack at each clock constraint.]

VI. Results and Analysis

Measured power on 16 nm UltraScale+

Tables 5 and 6 report the full component-resolved measurements. Total power on the Kintex UltraScale+ falls from 5.072 W at 1 ns to 0.716 W at 20 ns, and on the Zynq UltraScale+ from 4.747 W to 0.800 W. Junction temperature stays between 25.8 °C and 32.1 °C at every point, more than 67 °C of thermal margin, so no thermal derating affects the comparison. Figure 4 visualises the breakdown: the static band is nearly constant per device while every dynamic component scales with frequency, and I/O is the largest dynamic consumer at every operating point, 46% to 54% of dynamic power.

Table 5: Measured power of LAES on Kintex UltraScale+ (W)

Clock	Freq. (MHz)	Clocks	Signals	Logic	I/O	Dynamic	Static	Total	T_j (°C)
1 ns	1000	0.015	1.309	1.128	2.139	4.590	0.482	5.072	32.1
5 ns	200	0.003	0.262	0.226	0.457	0.948	0.456	1.403	27.0
10 ns	100	0.001	0.131	0.113	0.247	0.492	0.453	0.945	26.3
15 ns	66.7	0.001	0.087	0.075	0.177	0.340	0.452	0.792	26.1
20 ns	50	0.001	0.065	0.056	0.142	0.265	0.451	0.716	26.0

Table 6: Measured power of LAES on Zynq UltraScale+ (W)

Clock	Freq. (MHz)	Clocks	Signals	Logic	I/O	Dynamic	Static	Total	T_j (°C)
1 ns	1000	0.026	1.260	0.977	1.905	4.167	0.580	4.747	29.6
5 ns	200	0.005	0.252	0.195	0.405	0.858	0.565	1.423	26.4
10 ns	100	0.003	0.126	0.098	0.218	0.444	0.564	1.008	26.0
15 ns	66.7	0.002	0.084	0.065	0.155	0.306	0.563	0.869	25.8
20 ns	50	0.001	0.063	0.049	0.124	0.237	0.563	0.800	25.8

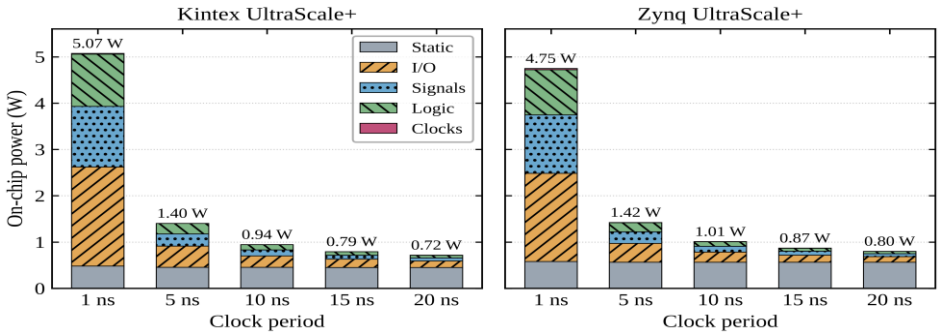


Fig. 4. Component-resolved power of LAES on the two 16 nm devices across the clock sweep. I/O dominates the dynamic budget at every operating point.

Model extraction, validation and decomposition

Fitting Equation (2) to the five total-power points of each device yields the parameters of Table 8. The fits are exceptionally tight: worst-case error is 0.05% on Kintex UltraScale+ and 0.03% on Zynq UltraScale+, against 0.40% and 0.22% for the 28 nm devices in [XIV]. Because a good total-power fit does not by itself identify the physical origin of k and P_0, every power component was also fitted separately (Table 7). The signals and logic components pass through the origin to within 0.2 mW, as switching power should. The I/O component does not: it carries a frequency-independent offset of 36.8 mW on Kintex UltraScale+ and 30.2 mW on Zynq UltraScale+, which explains why the measured Kintex UltraScale+ dynamic power at 50 MHz (0.265 W) exceeds the 0.229 W predicted by the aggregate slope alone. The static component has a small positive slope (0.032 and 0.018 nJ per cycle) arising from self-heating at the high-frequency points. The component slopes and intercepts add up to the aggregate parameters of Table 8 to within rounding, which confirms the decomposition of Equations (3) and (4).

Table 7: Component-level linear fits $P_c(f) = k_c \cdot f + P_{c,0}$ on the 16 nm devices

Component	KU+ slope (nJ)	KU+ intercept (mW)	ZU+ slope (nJ)	ZU+ intercept (mW)	Term in Eq. (3)/(4)
Clocks	0.015	≈ 0	0.026	≈ 0	k_{sw}
Signals	1.309	≈ 0	1.260	≈ 0	k_{sw}

Component	KU+ slope (nJ)	KU+ intercept (mW)	ZU+ slope (nJ)	ZU+ intercept (mW)	Term in Eq. (3)/(4)
Logic	1.128	≈ 0	0.977	≈ 0	k_{sw}
Datapath switching (sum)	**2.452**	≈ 0	**2.263**	≈ 0	**k_{sw}**
I/O	2.102	36.8	1.875	30.2	k_{IO}, $P_{IO,0}$
Device static	0.032	0.6	0.018	561.9	k_{th}, $P_{stat,0}$
Total (sum of components)	**4.586**	**486.4**	**4.156**	**592.1**	**k, P_0**

Table 8: Extracted aggregate model parameters, all four devices

Parameter	Kintex US+ (16 nm)	Zynq US+ (16 nm)	Kintex-7 (28 nm) [XIV]	Spartan-7 (28 nm) [XIV]
Aggregate slope k = energy per cycle (nJ)	4.586	4.155	5.175	5.246
Aggregate floor P_0 (W)	0.486	0.592	0.113	0.097
of which device static $P_{stat,0}$ (W)	0.450	0.562	n/r	n/r
of which I/O offset $P_{IO,0}$ (W)	0.037	0.030	n/r	n/r
Worst-case model error (%)	0.05	0.03	0.40	0.22
Boundary $f^* = P_0/k$ (MHz)	106	143	21.8	18.5
Efficiency asymptote η_{max} (Gbps/W)	27.91	30.81	24.73	24.40
Energy floor k/128 (pJ/bit)	35.8	32.5	40.4	41.0

n/r: not reported; component-level data for the 28 nm devices are not available in [XIV], so their intercepts are upper bounds on 28 nm device static power.

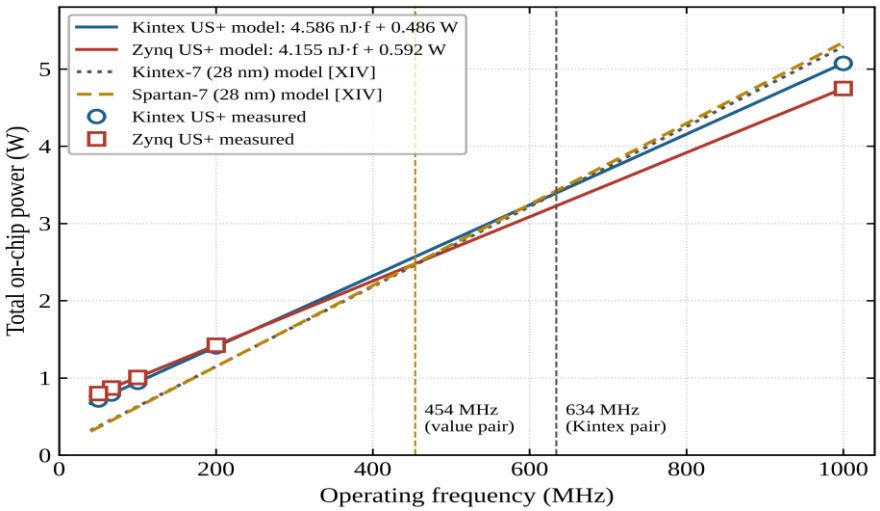


Fig. 5. Measured total power of LAES on 16 nm UltraScale+ devices against the linear model, with 28 nm baseline models dotted. Node-crossover frequencies are marked.

Process-node scaling: a dynamic–static trade

Table 8 quantifies what migration buys and what it costs. On the performance pair, the aggregate energy per cycle falls from 5.175 nJ (Kintex-7) to 4.586 nJ (Kintex UltraScale+), an 11.4% reduction; on the value pair it falls from 5.246 nJ (Spartan-7) to 4.155 nJ (Zynq UltraScale+), 20.8%. FinFET switching is cheaper, as expected. The aggregate floor rises from 0.113 W to 0.486 W and from 0.097 W to 0.592 W, factors of 4.3 and 6.1. After the I/O offset is removed, device static power alone (0.450 W and 0.562 W) still exceeds the entire 28 nm floor by at least 4.0× and 5.8×; these are lower bounds because the 28 nm intercepts may themselves contain an I/O offset. Vivado device static power is the leakage of the whole die, so the increase reflects the much larger capacity of the chosen UltraScale+ parts as much as the process node. The data therefore establish a device-level static-floor increase on migration; they do not isolate a FinFET leakage mechanism, and we do not attribute the increase to one. The boundary f^* consequently moves from about 20 MHz on 28 nm to 106 and 143 MHz on 16 nm: across most of the sub-150 MHz range in which embedded gateways typically clock, the 16 nm implementations are dominated by their frequency-independent floor.

Equation (8) locates the nominal break-even points at 634 MHz for the Kintex pair and 454 MHz for the value pair. Below these frequencies the 28 nm devices draw less total power and less energy per bit for identical work; above them the 16 nm devices would draw less. Figure 6 shows the resulting energy-per-bit picture. At 50 MHz the Spartan-7 encrypts at 56.1 pJ/bit while the Zynq UltraScale+ needs 125.0 pJ/bit, 2.2× more; at the 1 ns operating point the ranking inverts, with the Zynq UltraScale+ at 37.1 pJ/bit against 41.5 pJ/bit for Kintex-7. Both crossover frequencies, however, lie above the clock that the unregistered ten-round cascade can sustain (Section III). For the netlist evaluated here, the 28 nm devices therefore draw less power at every clock the design can actually reach. The crossovers become reachable only in a pipelined variant, and they apply to it only if inserting the 1152 additional flip-flops leaves k and P_0 essentially unchanged; since added registers increase clock and register power, this must be measured rather than assumed. The practical rule for this datapath class is stark: process migration pays, if at all, only in the high-throughput regime.

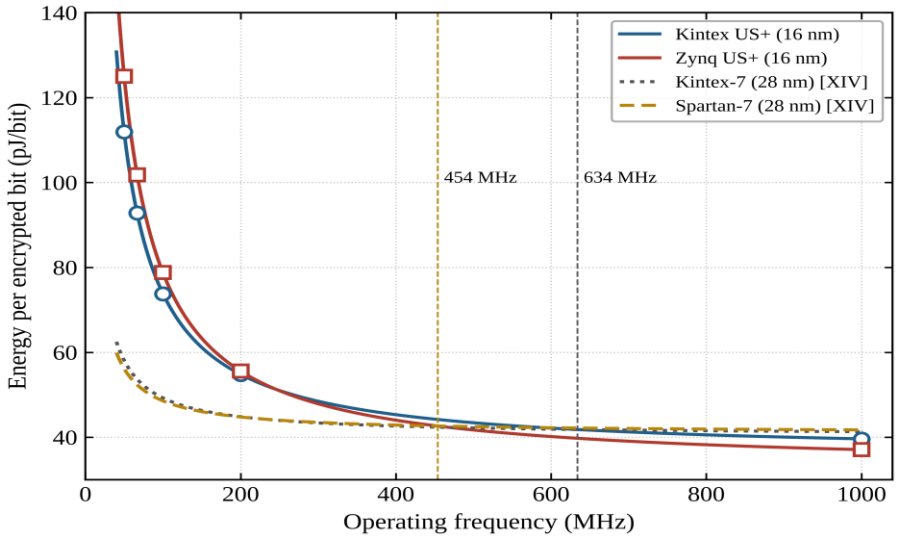


Fig. 6. Energy per encrypted bit across process nodes. The 28 nm devices win below the nominal crossover frequencies of 454 and 634 MHz; the 16 nm devices win above.

Table 9: Derived throughput and energy metrics on the 16 nm devices

Clock	Nominal throughput 128·f (Gbps)	KU+ E_{bit} (pJ)	KU+ η (Gbps/W)	ZU+ E_{bit} (pJ)	ZU+ η (Gbps/W)
1 ns†	128.0	39.6	25.24	37.1	26.96
5 ns†	25.6	54.8	18.25	55.6	17.99
10 ns	12.8	73.8	13.54	78.8	12.70
15 ns	8.53	92.8	10.77	101.8	9.82
20 ns	6.40	111.9	8.94	125.0	8.00

† Power-model operating point; the unregistered ten-round path does not meet timing at this period, so the throughput is not achieved by the evaluated netlist. [AUTHOR: mark any further rows whose constraint fails timing.]

Throughput efficiency

Figure 7 plots Equation (6). The 16 nm asymptotes of 27.9 and 30.8 Gbps/W exceed the 28 nm bound of 24.7 Gbps/W by 13% and 25%, reflecting the cheaper FinFET switching; the Zynq UltraScale+ reaches 26.96 Gbps/W at the 1 ns operating point. But the higher floors mean the 16 nm curves climb far more slowly: at 100 MHz the Kintex UltraScale+ delivers 13.5 Gbps/W where the Kintex-7 delivers 20.3 Gbps/W. Efficiency leadership, like energy leadership, belongs to 16 nm only in the high-frequency regime, which this netlist cannot reach without pipelining.

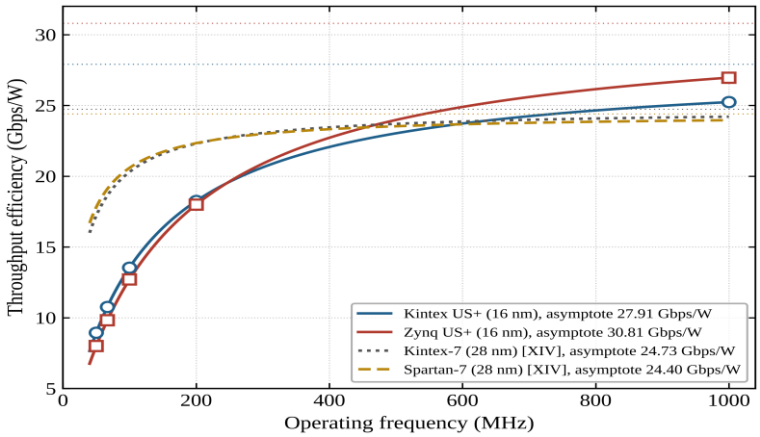


Fig. 7. Throughput-per-watt across the four devices with analytical asymptotes.

Where the power actually goes: I/O dominance

The component data of Tables 5 to 7 carry a further design lesson. The datapath switching energy k_{sw}, the sum of the clocks, signals and logic slopes, is only 2.45 nJ per block on Kintex UltraScale+ and 2.26 nJ on Zynq UltraScale+, roughly half of the aggregate slope. I/O contributes 46% to 54% of dynamic power across the sweep (its share rises at low frequency because of the 30 to 37 mW offset), a consequence of the 257 parallel pins that the unrolled 128-bit-in, 128-bit-out interface demands. On 16 nm fabric the interface, not the datapath, is the first-order optimisation target: serialising the block interface, registering it through fewer pins or embedding LAES behind an on-chip bus (natural on the Zynq device, where the processing system already provides AXI connectivity) would attack the dominant term directly. We quantify this as future work.

VII. Comparison with Published Implementations

Table 10 and Figure 8 situate the results against published AES and lightweight-AES FPGA implementations and against the 28 nm LAES baseline. In area, the UltraScale+ ports remain 71.6% to 72.5% below the lightweight AES of [XV] and 36.7% to 37.4% below the full AES of [XII], and within 1.6% of the 28 nm LAES footprint, demonstrating clean architectural portability. In power at the 20 ns setting, 0.716 W and 0.800 W remain 90.6% and 89.5% below the crypto-processor of [I] and 84.2% and 82.3% below the design of [II], though above the 0.359 to 0.371 W of the 28 nm LAES for the reasons established in Section VI. The energy figures of 37.1 and 39.6 pJ/bit are model values at the 1 ns operating point and characterise the datapath, not a secure cipher.

Table 10: Comparison with published FPGA implementations

Work	Node	Device	LUTs	FF	IO	TPC (W)	Best E_{bit} (pJ)
Lightweight AES [XV]	28 nm	Artix-7	9175	—	384	—	—
Lightweight AES [XV]	28 nm	Kintex-7	9468	—	384	—	—
AES [XII]	—	FPGA	4115	3987	269	—	—

Work	Node	Device	LUTs	FF	IO	TPC (W)	Best E_{bit} (pJ)
AES crypto processor [I]	—	FPGA	—	—	—	7.628	—
Power-efficient AES [II]	—	FPGA	—	—	—	4.528	—
Power-efficient AES [XIII]	28 nm	Artix-7	—	—	—	1.676	—
LAES [XIV]	28 nm	Kintex-7	2563	256	257	0.371	41.5‡
LAES [XIV]	28 nm	Spartan-7	2579	256	257	0.359	41.8‡
This work	16 nm	Kintex US+	2605	256	257	0.716	39.6‡
This work	16 nm	Zynq US+	2577	256	257	0.800	37.1‡

‡ Model value at the 1 ns operating point, not an achieved operating condition. TPC values are as reported by each work under its own methodology.

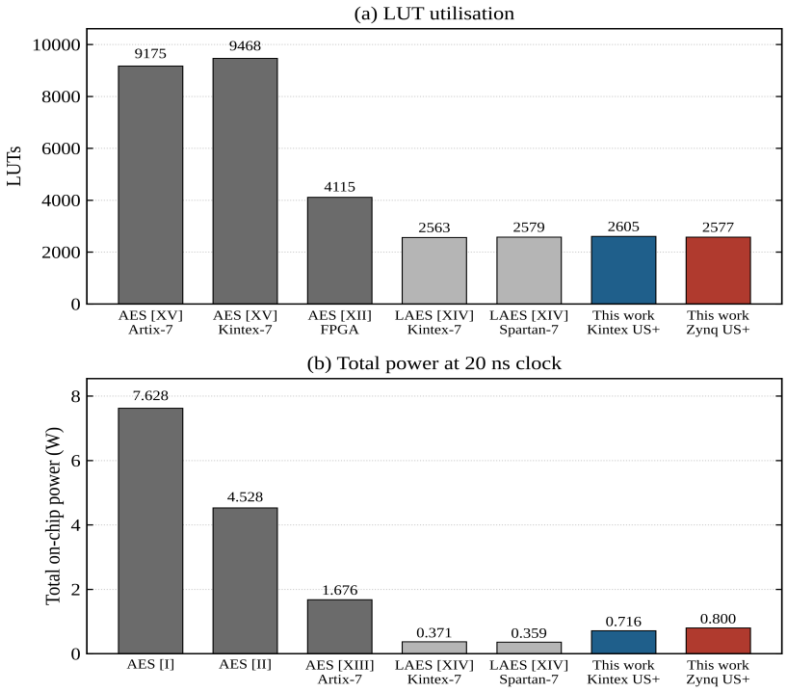


Fig. 8. (a) LUT utilisation and (b) total power at the 20 ns setting against published baselines.

The usual caveats of FPGA cipher comparison apply: baselines differ in unrolling factor, I/O methodology and activity assumptions, and [I] to [XIII] implement full AES. The area and power gap to full AES is bought by removing the diffusion layer, which, as Section III shows, removes the security of the construction; the comparison is therefore one of datapath cost only. Within those limits it establishes that the LAES footprint and interface are node-invariant while its energy behaviour is strongly node-dependent, which is precisely the effect this paper models.

VIII. Node and Operating-Point Selection for IoT Systems

The rules below concern node and operating-point selection for an AES-class round datapath; they are not a recommendation to deploy LAES as specified, which

Section III shows to be insecure. The model reduces provisioning to three rules, summarised in Table 11. First, choose the node by comparing the application clock against the crossover of Equation (8): below roughly 450 to 630 MHz the 28 nm devices deliver identical throughput at lower power and lower energy per bit; above it the 16 nm devices win, which requires a pipelined datapath able to close timing there. Second, on whichever node is chosen, run as fast as the timing budget allows and idle otherwise, since Equation (5) makes energy per bit monotonically decreasing in frequency; the penalty for a slow clock is far harsher on 16 nm, where the floor is larger (at 50 MHz the Zynq UltraScale+ pays 92.5 of its 125 pJ/bit to the floor alone). Third, when the 16 nm node is selected for platform reasons (integration or clock ceiling rather than energy), attack I/O first, because it is the dominant dynamic term.

Table 11: Recommended selections by deployment class

Deployment class	Typical regime	Recommended device	Rationale from model
Battery or duty-cycled sensor aggregation	≤ 100 MHz bursts	Spartan-7 (28 nm)	Lowest floor (0.097 W); 56 pJ/bit at 50 MHz vs 125 pJ/bit on Zynq US+
Mains-powered gateway, sub-watt crypto budget	66–200 MHz	Kintex-7 or Spartan-7 (28 nm)	Below both crossovers (454/634 MHz); meets 0.5 W cap up to ~75 MHz
High-rate edge node or aggregation core	≥ 500 MHz	Zynq US+ (16 nm), pipelined datapath only	37.1 pJ/bit at the 1 ns operating point; asymptote 30.8 Gbps/W; requires timing closure
SoC-integrated secure gateway	Platform-driven	Zynq US+ (16 nm)	Processing-system integration; amortise the static floor across the whole SoC; serialise I/O behind AXI

IX. Limitations and Threats to Validity

- Power derives from vectorless Vivado analysis. Absolute silicon values will differ; the linearity of the dynamic term and the extracted slopes, which reflect netlist structure rather than the activity constant, are expected to transfer. Board-level validation with SAIF-annotated activity remains future work.
- The Kintex UltraScale+ figures derive from the synthesised netlist and its utilisation is post-synthesis, whereas the Zynq UltraScale+ figures derive from the implemented netlist. The near-identical fitted slopes (within 10%) and the sub-0.1% fit errors on both suggest the asymmetry is second-order for the conclusions drawn, but a fully post-implementation Kintex UltraScale+ characterisation would remove it.
- The static floor is a property of the chosen device sizes as much as of the process node. Smaller UltraScale+ parts would lower the floor and shift the crossover frequencies downward; the methodology, not the specific crossover values, is the transferable result. Because component data for the 28 nm devices are not available, the 16 nm-to-28 nm static ratios of 4.0× and 5.8× are lower bounds.
- The round cascade is unregistered, so the clock ceiling of the evaluated netlist is set by the full ten-round combinational path. The 1 ns and 5 ns settings, and the

128 Gbps and 25.6 Gbps figures, are power-model operating points and nominal throughputs rather than achieved operating conditions. A pipelined variant would close timing at higher clocks but would add 1152 flip-flops and change k and P_0.

- The evaluated netlist has no key port; the key is supplied internally. A deployable core would add 128 key inputs (or a key-load interface) and key-schedule logic or registers, increasing area, I/O power and $P_{IO,0}$.
- The efficiency expressions assume a fully occupied datapath; bursty traffic degrades effective efficiency toward the raw power view, mitigated by the race-to-idle policy when clock gating is available.
- As shown in Section III, the specified LAES round function has no inter-byte diffusion: each ciphertext byte depends on a single plaintext byte, and 256 chosen plaintexts recover the full codebook. LAES as specified must not be used to protect data. Deployments should use standardised primitives such as AES [XX] or ASCON [XXI]; the power-modelling results of this paper apply to the datapath, and a variant with a restored diffusion layer must be re-characterised.

X. Conclusion and Future Work

This paper reported the first realisation of the LAES datapath on 16 nm UltraScale+ fabric and, by holding the RTL fixed and the measurement protocol identical to the 28 nm baseline, isolated the process-node contribution to its power and energy behaviour. A two-parameter linear model captures every measured operating point on both new devices with at most 0.05% error and yields the scaling result in closed form: FinFET migration cuts the aggregate energy per cycle by 11.4% to 20.8% and raises the efficiency asymptote to 30.8 Gbps/W, but multiplies the frequency-independent floor by 4.3 to 6.1, of which device static power accounts for all but 30 to 37 mW. The 16 nm devices are preferable only above nominal crossover frequencies of 634 MHz (Kintex pair) and 454 MHz (value pair), which the unregistered ten-round netlist cannot reach; at gateway-class frequencies the 28 nm Spartan-7 remains more than twice as energy-efficient. Component-level analysis further shows that on 16 nm the datapath dissipates only about half of the dynamic budget, with I/O contributing 46% to 54%, redirecting optimisation effort from the datapath to the interface. The revision also corrects two errors in the submitted manuscript: the design is an unrolled but unregistered datapath rather than an eleven-stage pipeline, and the MixColumns-free round function has no inter-byte diffusion, so LAES as specified is not a secure cipher.

Future work proceeds in five directions: restoring an inter-byte diffusion layer (MixColumns or a lightweight MDS or near-MDS matrix) and re-characterising the secure variant with the same methodology; pipelining the round cascade and measuring the effect of the added registers on k and P_0; serialising or bus-embedding the block interface to attack the dominant I/O term, with the AXI fabric of the Zynq device as the natural vehicle; fine-grained clock gating and power gating to lower the effective floor on the FinFET node, using the extracted $P_{stat,0}$ as the yardstick; and board-level energy measurement with activity-annotated simulation to validate the models on silicon, together with application of the same fixed-RTL, fixed-protocol methodology to standardised lightweight ciphers such as ASCON.

Conflict of Interest

The authors declare that there was no relevant conflict of interest regarding this paper.

References

I. Aditya, Y., and Keshav Kumar. "Implementation of High-Performance AES Crypto Processor for Green Communication." *Telematique*, vol. 21, no. 1, 2022, pp. 6808–16.

II. Aditya, Y., and Keshav Kumar. "Implementation of Novel Power Efficient AES Design on High Performance FPGA." *NeuroQuantology*, vol. 20, no. 10, 2022, p. 5815.

III. Ahmed, Salman, et al. "Lightweight AES Design for IoT Applications: Optimizations in FPGA and ASIC with DFA Countermeasure Strategies." *IEEE Access*, vol. 13, 2025, pp. 22489–509. 10.1109/ACCESS.2025.3533611.

IV. Balan, N. Siva, and B. S. Murugan. "Low Area FPGA Implementation of AES Architecture with EPRNG for IoT Application." *Journal of Electronic Testing*, vol. 38, no. 2, 2022, pp. 181–93. 10.1007/s10836-022-05997-x.

IV. Beaulieu, Ray, et al. "The SIMON and SPECK Families of Lightweight Block Ciphers." *IACR Cryptology ePrint Archive*, Report 2013/404, 2013. https://eprint.iacr.org/2013/404.

V. Bisht, Neeraj, Bishwajeet Pandey, and Sandeep Kumar Budhani. "Comparative Performance Analysis of AES Encryption Algorithm for Various LVCMOS on Different FPGAs." *World Journal of Engineering*, vol. 20, no. 4, 2023, pp. 669–80. 10.1108/WJE-05-2021-0281.

VI. Bogdanov, Andrey, et al. "PRESENT: An Ultra-Lightweight Block Cipher." *Cryptographic Hardware and Embedded Systems – CHES 2007*, Lecture Notes in Computer Science, vol. 4727, Springer, 2007, pp. 450–66. 10.1007/978-3-540-74735-2_31.

VII. Brahmaiah, V. Priyanka, et al. "Implementation of AES Algorithm." *Proceedings of the International Conference on Information and Communication Technology for Intelligent Systems*, Springer Nature Singapore, 2023, pp. 161–71.

IX. Daemen, Joan, and Vincent Rijmen. *The Design of Rijndael: AES – The Advanced Encryption Standard*. Springer, 2002. 10.1007/978-3-662-04722-4.

X. Equihua, Christian, et al. "A Low-Cost and Highly Compact FPGA-Based Encryption/Decryption Architecture for AES Algorithm." *IEEE Latin America Transactions*, vol. 19, no. 9, 2021, pp. 1443–50. 10.1109/TLA.2021.9468436.

XI. Khan, Abdullah Ayub, et al. "Internet of Things (IoT) Security with Blockchain Technology: A State-of-the-Art Review." *IEEE Access*, vol. 10, 2022, pp. 122679–95. 10.1109/ACCESS.2022.3223370.

XII. Kumar, Keshav, K. R. Ramkumar, and Amanpreet Kaur. "A Design Implementation and Comparative Analysis of Advanced Encryption Standard (AES) Algorithm on FPGA." *Proceedings of the 8th International Conference on Reliability, Infocom Technologies and Optimization (ICRITO)*, IEEE, 2020, pp. 182–85.

XIII. Kumar, Keshav, et al. "A Design of Power-Efficient AES Algorithm on Artix-7 FPGA for Green Communication." *Proceedings of the International Conference on Technological Advancements and Innovations (ICTAI)*, IEEE, 2021, pp. 561–64. 10.1109/ICTAI53825.2021.9673435.

XIV. Kumar, Keshav, Chinnaiyan Ramasubramanian, and Bishwajeet Pandey. "Efficient FPGA Realization of Lightweight AES for Low-Power IoT Security Systems." *Journal of Mechanics of Continua and Mathematical Sciences*, vol. 21, no. 1, 2026, pp. 65–79. 10.26782/jmcms.2026.01.00004.

XV. Kumar, Keshav, K. R. Ramkumar, and Amanpreet Kaur. "A Lightweight AES Algorithm Implementation for Encrypting Voice Messages Using Field Programmable Gate Arrays." *Journal of King Saud University – Computer and Information Sciences*, vol. 34, no. 6, 2022, pp. 3878–85. 10.1016/j.jksuci.2020.08.005.

XVI. Kumar, Keshav, Chinnaiyan Ramasubramanian, and Bishwajeet Pandey. "Low Area Design and Implementation of Lightweight Encryption Algorithm on FPGA for IoT Devices." *Proceedings of the IEEE 4th International Conference on AI in Cybersecurity (ICAIC)*, 2025, pp. 1–4. 10.1109/ICAIC63015.2025.10849186.

XVII. Kumar, Thanikodi Manoj, et al. "A Low Area High Speed FPGA Implementation of AES Architecture for Cryptography Application." *Electronics*, vol. 10, no. 16, 2021, p. 2023. 10.3390/electronics10162023.

XVIII. Maazouz, Mohamed, et al. "FPGA Implementation of a Chaos-Based Image Encryption Algorithm." *Journal of King Saud University – Computer and Information Sciences*, vol. 34, no. 10, 2022, pp. 9926–41. 10.1016/j.jksuci.2021.12.022.

XIX. Mohd, Bassam Jamil, and Thaier Hayajneh. "Lightweight Block Ciphers for IoT: Energy Optimization and Survivability Techniques." *IEEE Access*, vol. 6, 2018, pp. 35966–78. 10.1109/ACCESS.2018.2848586.

XX. National Institute of Standards and Technology. *Advanced Encryption Standard (AES)*. Federal Information Processing Standards Publication 197, 2001. 10.6028/NIST.FIPS.197.

XXI. National Institute of Standards and Technology. *Status Report on the Final Round of the NIST Lightweight Cryptography Standardization Process*. NIST Interagency Report 8454, 2023. 10.6028/NIST.IR.8454.

XXII. Pandey, Bishwajeet, et al. "Energy-Efficient Implementation of AES Algorithm on 16nm FPGA." *Proceedings of the 10th IEEE International Conference on Communication Systems and Network Technologies (CSNT)*, 2021, pp. 740–44.

XXIII. Pandey, Bishwajeet, et al. "Increasing Cyber Security by Energy Efficient Implementation of DES Algorithms on FPGA." *Journal of Green Engineering*, vol. 11, no. 10, 2021, pp. 72–82.

XXIV. Rana, Muhammad, Quazi Mamun, and Rafiqul Islam. "Lightweight Cryptography in IoT Networks: A Survey." *Future Generation Computer Systems*, vol. 129, 2022, pp. 77–89. 10.1016/j.future.2021.11.011.

XXV. Sawant, Anil Gopal, Vilas N. Nitnaware, and Anupama A. Deshpande. "Spartan-6 FPGA Implementation of AES Algorithm." *ICCCE 2019*, Lecture Notes in Electrical Engineering, vol. 570, Springer Singapore, 2020, pp. 205–11. 10.1007/978-981-13-8715-9_26.

XXVI. Sreekanth, Muttuluru, and R. K. Jeyachitra. "Implementation of Area-Efficient AES Using FPGA for IoT Applications." *International Journal of Embedded Systems*, vol. 15, no. 4, 2022, pp. 354–62. 10.1504/IJES.2022.125446.

XXVII. Thabit, Fursan, et al. "Cryptography Algorithms for Enhancing IoT Security." *Internet of Things*, vol. 22, 2023, p. 100759. 10.1016/j.iot.2023.100759.

XXVIII. Thind, Vandana, Bishwajeet Pandey, and D. M. Akbar Hussain. "Power Analysis of Energy Efficient DES Algorithm and Implementation on 28nm FPGA." *Proceedings of the IEEE CSE/EUC/DCABES*, 2016, pp. 600–03. 10.1109/CSE-EUC-DCABES.2016.247.

XXIX. Yang, C.-H., and Y.-S. Chien. "FPGA Implementation and Design of a Hybrid Chaos-AES Color Image Encryption Algorithm." *Symmetry*, vol. 12, no. 2, 2020, p. 189. 10.3390/sym12020189.