A Deeper Look at the Energy Consumption of Lightweight Block Ciphers

Andrea Caforio[1], Fatih Balli[1], Subhadeep Banik[1], Francesco Regazzoni[2,3]
[1]*LASEC, EPFL*, Lausanne, Switzerland; andrea.caforio@epfl.ch, subhadeep.banik@epfl.ch
[2]*University of Amsterdam*, Amsterdam; The Netherlands, f.regazzoni@uva.nl
[3]*ALaRI - USI*, Lugano, Switzerland; regazzoni@alari.ch

***Abstract*—In the last few years, the field of lightweight cryptography has seen an influx in the number of block ciphers and hash functions being proposed. In the past there have been numerous papers that have looked at circuit level implementation of block ciphers with respect to lightweight metrics like area power and energy. In the paper by Banik et al. (SAC'15), for example, by studying the energy consumption model of a CMOS gate, it was shown that the energy consumed per cycle during the encryption operation of an r-round unrolled architecture of any block cipher is a quadratic function in r.**

However, most of these explorative works were at a gate level, in which a circuit synthesizer would construct a circuit using gates from a standard cell library, and the area power and energy would be estimated by estimating the switching statistics of the nodes in the circuit. Since only a part of the EDA design flow was done, it did not account for issues that might arise when the circuit is finally mapped into silicon post route. Metrics like area, power and energy would need to be re-estimated due to the effect of the parasitics introduced in the circuit by the connecting wires, nodes and interconnects. In this paper, we look to plug this very gap in literature by re-examining the designs of lightweight block ciphers with respect to their performances after completing the placement and routing process. This is a timely exercise to do since three of the block ciphers we analyze in the paper are used in around 13 of the 32 candidates in the second round of the NIST lightweight competition being conducted currently.

***Index Terms*—Block Ciphers, Low Energy, Place and Route.**

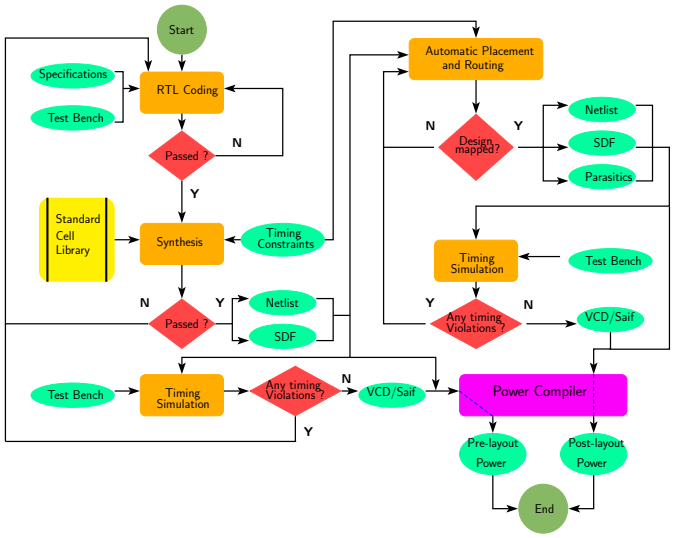


Fig. 1: ASIC Design flow that computes average power both pre and post the placement and routing

I. Introduction

In the past few years, lightweight cryptography has become a popular research discipline with a number of ciphers and hash functions proposed. Energy is a crucial measure of goodness for an algorithm. Indeed, any construction optimized with respect to energy has wide applications, especially in constrained environments running on a tight power/energy budget. Over the years there have been a lot of papers that study energy as an optimizable metric in block/stream cipher based protocols [1]–[5]. However, in all these papers energy is computed by calculating the average power consumption after the design is synthesized by a circuit compiler. A timing simulation over the synthesized netlist using a large enough set of test vectors is performed to either extract either a VCD or a SAIF file for the netlist. The former usually contains the timing waveforms for every node in the netlist recorded over the given simulation duration, while the latter only contains the average switching information for every node. Either of these files can be analyzed using any commercially available power compiler which then provides an estimate of the average power consumed by the circuit. The energy is usually computed as the product of the average power and the total physical time required to execute a given operation on the circuit.

We aim to address this gap that exists in literature by extending the simulation flow to until after the placement and routing (PR) has been done. After mapping any circuit into silicon the software which constructs the layout of the circuit, is also able to extract the parasitic capacitances associated with every node and recompute the area and the length of the timing paths in the circuit taking into account the additional resources required to accommodate the connections between each node, sometimes at different metal layers. It is possible then to do a timing simulation as done before, for this updated netlist and regenerate a VCD/SAIF file for the circuit. These files can again be analyzed using any power compiler to provide an estimate of the average power consumed by the circuit, taking into account the parasitics and reconstructed timing paths. The whole design flow has been explained in Figure 1.

Subhadeep Banik was supported by the Swiss National Science Foundation (SNSF) through the Ambizione Grant PZ00P2_179921

A. Contribution and Organization

In this paper, one of our primary goals was to understand how the power/energy consumption of block ciphers vary when computed via two different routes both pre and post routing. To achieve this goal we select a set of 11 lightweight block ciphers and run the ASIC design flow to generate two parallel sets of figures for the power consumption pre and post the placement and routing process. In addition we experiment with different architectures of each block cipher in which parts of the circuit are replicated in hardware to execute all encryption operations in fewer number of clock cycles (also called r-round unrolled architectures for r such replications). One of the conclusions we draw is the discrepancy between the pre and post place/route power/energy figures (which occur mainly due to the additional parasitic capacitances included in the netlist post place/route) depend heavily on the gate-level architecture of the specific block cipher. Furthermore, this discrepancy becomes much more significant when the physical area of the circuit becomes larger.

Such an exercise is timely considering the fact that AES, Gift and Skinny block cipher families are used as the underlying encryption primitive in 13 of the 32 candidates in the 2nd round of the NIST lightweight competition currently being conducted. Since one of the findings of the paper is that discrepancy between the power reported pre and post the place/route process has some correlation with the physical area of the circuit, it establishes that for lower circuit area the effect of parasitics is not significant enough to cause too much difference between the pre and post power energy figures. This gives any circuit designer the option to choose the degree of unrolling r of the circuit architecture (which naturally increases circuit area) as per the power/energy budget available.

The rest of the paper is organized in the following manner. Section II revisits the state of the art and brings the reader up-to-date with previous research articles on the topic of energy efficiency of block ciphers and introduces some relevant notation for the rest of the paper. Section III introduces the circuit level details of the block ciphers that we benchmark in this paper and also explains in detail the experimental setup we use in the paper. In section IV, we tabulate all the simulation results and explain them vis-a-vis theoretical considerations already introduced in the paper. Section V concludes the paper.

II. Preliminaries

A **block cipher** is a keyed permutation function. It is a deterministic algorithm operating on fixed-length groups of bits, called a block, with an unvarying transformation that is specified by a symmetric key (which is another bit string of short length). The basic component of a block cipher is an easy to compute permutation called a **round function** (sometimes simply referred to as a round). The encryption operation involves repeated application of the round function for a specified number of iterations. An r round unrolled implementation is one in which there are r copies of the round function circuit (RF_1 to RF_r) connected sequentially as shown

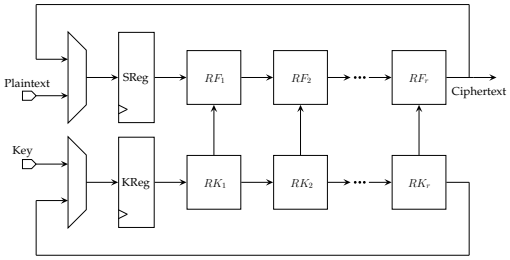


Fig. 2: Round unrolled Block cipher circuit

in Figure 2. The circuit may or may not require round key generation circuits (RK_1 to RK_r) depending on the design specifications. The advantage of unrolling a circuit is that it allows for faster computation and hence greater throughput: for example in the case of a block cipher if the design specifies R invocations of the round function, the computation is completed in $\lceil \frac{R}{r} \rceil$ clock cycles in an r round unrolled circuit. However, due to increased number of logic gates these circuits generally occupy higher area and consume higher power. In [3], the authors looked at design strategies like serialization and round unrolling and the effect it has on the energy consumption required to encrypt a single block of data. The authors then proposed a formal model for energy consumption in any r-round unrolled block cipher implementation. A study of the energy consumption of block ciphers depending on the number of unrolled rounds was done in [3]. The authors concluded that the energy consumed for encrypting one block of plaintext, for any r-round unrolled implementation had a quasi-quadratic form (a, b, c are constants and R is the number of iterations of the round function prescribed for the design): $E_r = (ar^2 + br + c) \cdot \left(1 + \lceil \frac{R}{r} \rceil\right)$. The logic used by the authors was as follows: since such a structure has r copies of the round function circuit connected serially one after the other, the glitches produced due to signal delays in the i^{th} round function, are compounded in the $(i+1)^{st}$ round function and is compounded further in the $(i+2)^{nd}$ round function. It was then shown that the energies consumed in each round function formed a simple arithmetic sequence. Since the total energy consumed is a sum of these r terms of the sequence, it results in a quadratic function in r. Multiplying this by the total time taken for computation i.e. $\left(1 + \lceil \frac{R}{r} \rceil\right)$ cycles gives us E_r.

Although an r-round unrolled cipher consumes more energy per cycle for increasing values of r, it takes less number of cycles to complete the encryption operation itself. So to determine the value of r at which the design consumes least energy is an interesting optimization problem, that was also investigated in [3]. The authors found that for some block ciphers like AES [6](not necessarily lightweight), implementations that execute one instantiation of the round function ($r = 1$) per clock cycle (such circuits are called round based circuits) are optimally energy-efficient. However for extremely lightweight ciphers like Present [7] and Simon [8], implementations with two round function executions per cycle ($r = 2$) were more energy-efficient. Building on these ideas,

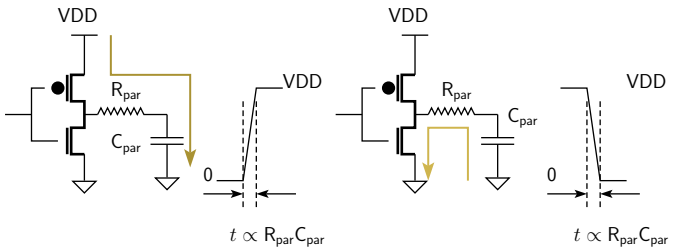


Fig. 3: Signal delay due to Parasitics

the block cipher family Midori was proposed in [9] that optimized the energy consumption per encryption.

III. Setup

It is well known in circuit theory that the time taken to charge/discharge a node in any electrical circuit is proportional to the product of the resistance and capacitance seen by the voltage source at the node (see Fig 3). In CMOS circuits, the total capacitance seen at any node can arise due to the **(a)** drain/gate to source capacitances of each transistor which compose the standard cells, and **(b)** parasitic capacitances contributed to by various gates and interconnects used in the mapping and placement of the final circuit in silicon. Out of these, the first is likely to modeled at the pre PR stage whereas the second is expected to be factored in only after placement and routing of the circuit. Thus it is only natural that the timing paths calculated by the compiler before and after the physical layout has been done would be widely different. Therefore the critical path (in ns) estimated by the compiler after gate-level synthesis is likely to be far lower than that estimated by the PR software after placement and routing.

Also the total dynamic energy consumed by any CMOS circuit (per glitch $0 \rightarrow 1/1 \rightarrow 0$ transition) is proportional to $C_{Load} \cdot V_{DD}^2$, where C_{Load} is the total load capacitance seen by the voltage source at the charging/discharging node. Since the post PR model includes parasitic capacitances too, every node in the netlist post PR, is likely to record higher capacitance value than in the corresponding netlist pre PR. It is due to this reason that the total dynamic power/energy computed by any power compiler pre PR is also likely to be much lower than the corresponding figure computed post PR. In order to put our hypotheses to test we experimented with 11 block ciphers well known in literature.

AES-128 [6] Ever since standardized by NIST, AES-128 has been the de-facto encryption worldwide. Over the years there have been various implementations of AES, each targeting the optimization of a specific metric. The circuit we use is the same that was proposed in [3] that is known to be energy-efficient.

Noekeon [10] Noekeon is a block cipher with 128-bit block-size and key which was initially a submission to the Nessie project. The block cipher is designed specifically to be involutive, in which the encryption and decryption can be both performed with minimal circuit level tweaks.

Present [7] Present is a 64-bit block cipher which has an SPN type round function. It has recently been adopted as a standard in ISO/IEC 29192-2. The only non-linear component in the round function is the 4-bit S-box, which is applied in parallel to each of the sixteen nibbles of the 64-bit state after the RoundKey addition. Thereafter the state-bits are rearranged by a permutation layer.

LED [11] LED is a 64-bit block cipher with an SPN type round function. It allows for Keys of size 64, 80, 128 bits. We will concentrate on the 128-bit version. It has an AES-like round function but uses the 4-bit Present S-box. Although it has no KeySchedule operation, the most significant bits and the least significant bits of the Key are alternately added to the state after every 4 rounds.

Prince [12] Prince is a 64-bit block cipher with an SPN type round function. It was designed with low latency for specific use in memory encryption.

Twine [13] It is a 64-bit block cipher with a Type 2 Feistel round function. Here too the cipher allows 80 and 128-bit Keys, but we will focus on the 80-bit version.

Piccolo [14] Piccolo is another 64-bit Feistel cipher with support for 80 and 128 bit keys, but we will focus on the 80-bit version.

Simon 64/96 [8] The Simon and Speck family of block ciphers are a family of lightweight ciphers proposed with support for various block and key sizes. In this work, we will concentrate on Simon 64/96, which has a 64-bit block length, 96-bit key and a Feistel type round function. Unlike most other lightweight ciphers, Simon does not employ any Substitution table but generates non-linearity by employing bitwise AND operations.

Midori 128 [9] The Midori family of block ciphers was proposed with view to minimize the energy consumption per encryption operation. The family recommends ciphers of blocksizes 64 and 128. It has an SPN type round function, in which the Substitution and Linear layers are optimized so as to minimize energy consumption.

Skinny 128/128 [15] Skinny was designed as a new lightweight family of block ciphers whose goal was to compete with the Simon family in terms of hardware/software performances, while proving in addition much stronger security guarantees with regard to differential/linear attacks.

Gift 128 [16] Similarly, Gift was designed to compete with Present in terms of hardware/software performance and security against known attacks. Like Present, Gift uses only a bit permutation to achieve diffusion between rounds and is one of the most lightweight block ciphers in literature.

For each block cipher listed above, we experimented with various unrolled architectures. The following design flow was adhered to (which closely follows the flow in Figure 1). First all the circuits were implemented in VHDL. Then, a functional verification at the RTL level was first done using Mentor Graphics Modelsim software. The designs were synthesized

using the standard cell library of the TSMC 90nm 9-metal layer logic process with the Synopsys Design Compiler, with the compiler flag set to `compile_ultra` (this is a compile step that performs additional optimization routines). A timing simulation was done on the synthesized netlist to confirm the correctness of the design, by comparing the output of the timing simulation with known test vectors. Note that the frequency of operation was fixed at 10 MHz because as established in [1], [3], at high frequencies the energy consumption of block ciphers is frequency-independent. The switching activity of each gate of the circuit was collected in a SAIF file while running post-synthesis simulation. The average power was obtained using *Synopsys Power Compiler*, using the back annotated switching activity. Energy was calculated as the product of average power and time taken for one encryption/decryption. For the post synthesis, we used the Cadence Innovus software to generate the layout and additionally generate updated netlist, SDF (file that stores delay of each node in the custom format) and the SPEF (that contains custom parasitic information) file. We did an additional timing simulation on the post PR netlist and the updated SDF file to generate another SAIF file that contains the switching information of every node of the post PR netlist. Thereafter the post PR netlist, SDF, SPEF and SAIF files were input to the Prime Power PX software that estimates the post PR power consumption.

IV. Results and Discussion

The results of the simulation obtained after executing the design flow on the various architectures of the above mentioned block ciphers is presented in Table I and in graphical form in Figure 5. The simulation results in a sense confirm our hypotheses that after executing the entire flow, Post PR netlists usually report a) higher critical path and b) higher power consumption than the corresponding figures recorded just after the gate level synthesis. Undoubtedly the primary reason for this are the additional parasitic capacitances which are factored into the model only after executing the placement and routing process.

We also notice that the difference in the two reported power consumptions for each designs increase with increase in the total post PR circuit area. To see the effect clearly we plotted this power difference against the total circuit area in Figure 4 for each block cipher. Note that for ease of representation, the figure has been split into 3 sub-plots. We can see that the plot is more or less linear for most of the block ciphers we have benchmarked. We argue that as the circuit area increases due to replication of additional round function circuits, so do the number of interconnects required to connect each round function circuit to the next and then back to the state and key register. This increases proportionally the value of additional parasitic capacitances added to the circuit which causes both the critical path and the average dynamic power consumed to increase proportionally.

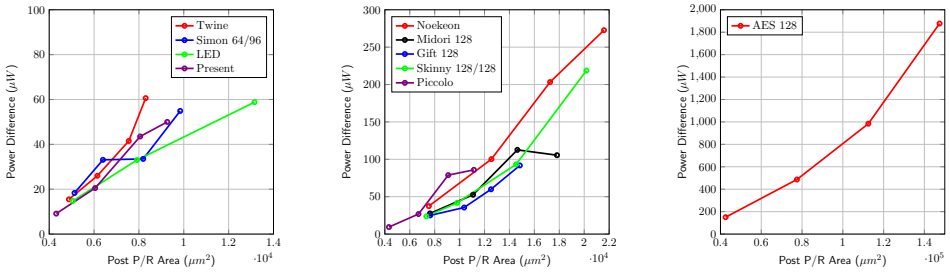
V. Conclusion

In this paper, we look at the post PR effects of energy consumption of lightweight block ciphers. In most previous articles in this topic, the exploration was limited to the gate level. It naturally did not consider issues that would arise due to the effect of the parasitics introduced in the circuit by the connecting wires, nodes and interconnects. In this paper, we executed the entire ASIC design flow from the RTL to the PR stage on 11 lightweight block ciphers and in the process extracted two sets of power consumption figures one before and after the placement and routing procedure. We were able to ascertain that due to the additional parasitic capacitances, the critical path and the average power consumption reported post PR is usually much higher than those reported pre PR. Furthermore this difference seems to increase more or less proportionally with increase in circuit area.

References

[1] S. Kerckhof, F. Durvaux, C. Hocquet, D. Bol, and F. Standaert, "Towards green cryptography: A comparison of lightweight ciphers from the energy viewpoint," in *Cryptographic Hardware and Embedded Systems - CHES 2012 - 14th International Workshop, Leuven, Belgium, September 9-12, 2012. Proceedings*, 2012, pp. 390–407. [Online]. Available: https://doi.org/10.1007/978-3-642-33027-8_23

[2] L. Batina, A. Das, B. Ege, E. B. Kavun, N. Mentens, C. Paar, I. Verbauwhede, and T. Yalçin, "Dietary recommendations for lightweight block ciphers: Power, energy and area analysis of recently developed architectures," in *Radio Frequency Identification - Security and Privacy Issues 9th International Workshop, RFIDsec 2013, Graz, Austria, July 9-11, 2013, Revised Selected Papers*, 2013, pp. 103–112. [Online]. Available: https://doi.org/10.1007/978-3-642-41332-2_7



(a) (b) (c)

Fig. 4: Power difference computed pre and post place/route when compared with total area of circuit

	r	Cycles	Block	Pre-Place/Route						Post-Place/Route					
				Area		Power	Energy	Path	Max TP	Area		Power	Energy	Path	Max TP
				μm^2	(GE)	μW	(pJ)	(ns)	(Gbps)	μm^2	(GE)	μW	(pJ)	(ns)	(Gbps)
Twine	1	37	64	4433.99	1571.00	72.1	266.84	1.22	1.320	4879.22	1728.75	87.6	324.08	2.453	0.657
	2	19	64	5718.99	2026.29	107.4	204.06	1.25	2.510	6148.60	2178.50	133.4	253.46	3.628	0.865
	3	13	64	7118.80	2522.25	161.9	210.47	1.56	2.939	7547.80	2674.25	203.4	264.42	5.143	0.891
	4	10	64	7863.91	2786.25	219.8	219.80	2.05	2.908	8300.69	2941.00	280.4	280.40	6.712	0.888
Prince	1	13	64	6797.75	2408.50	98.5	128.10	1.93	2.376	6790.69	2406.00	123.4	160.42	4.734	0.969
	Half	3	64	22992.68	8146.50	518.1	155.43	3.20	6.209	23032.19	8160.50	643.3	192.99	10.398	1.911
	Full	1	64	22070.46	7819.80	1527.6	152.76	5.78	10.312	21112.51	7480.34	1904.5	190.45	16.551	3.601
Piccolo	1	26	64	4313.33	1528.25	65.8	170.98	1.71	1.341	4330.27	1534.25	75.2	195.42	4.873	0.47
	2	14	64	6681.33	2367.25	152.6	213.64	2.84	1.499	6702.49	2374.75	179.4	251.16	8.173	0.521
	3	10	64	9054.26	3208.00	303.5	303.50	3.80	1.569	9087.42	3219.75	382.3	382.30	10.877	0.548
	4	8	64	11117.43	3939.00	472.9	378.32	4.91	1.517	11147.07	3949.50	558.8	447.04	13.323	0.559
Simon 64/96	1	43	64	4619.56	1636.75	69.3	298.16	1.19	1.165	5128.30	1817.00	87.7	376.94	2.797	0.496
	2	22	64	5909.40	2093.75	80.0	176.00	1.52	1.782	6396.26	2266.25	113.1	248.82	2.908	0.932
	3	15	64	7693.16	2725.75	135.7	203.55	1.67	2.379	8191.31	2902.25	169.2	253.80	3.605	1.102
	4	12	64	9260.29	3281.00	182.0	218.40	1.62	3.066	9841.00	3486.75	236.9	284.28	5.202	0.955
LED	1	50	64	5040.10	1785.75	115.3	576.50	2.17	0.549	5052.09	1790.00	130.1	650.35	4.777	0.250
	2	26	64	7801.11	2764.00	283.9	738.14	2.82	0.813	7911.89	2803.25	316.9	823.94	6.905	0.332
	4	14	64	13131.92	4652.75	780.7	1092.98	4.64	0.918	13156.62	4661.50	839.5	1175.30	11.951	0.356
PRESENT	1	33	64	4260.41	1509.50	67.2	221.62	1.23	1.468	4314.03	1528.50	76.3	251.72	2.325	0.777
	2	17	64	6021.59	2133.50	108.8	184.96	1.21	2.898	6048.40	2143.00	129.3	219.81	2.638	1.329
	3	12	64	7920.40	2806.26	158.2	189.84	1.36	3.652	8055.13	2854.00	201.7	242.04	4.138	1.200
	4	9	64	9205.96	3261.75	217.6	195.84	1.61	4.114	9263.12	3282.00	267.6	240.84	5.197	1.274
Noekeon	1	18	128	7163.25	2538.00	149.2	268.56	1.26	5.256	7532.99	2669.00	186.6	335.88	3.781	1.752
	2	10	128	12337.42	4371.25	608.1	608.10	3.30	3.612	12552.62	4447.50	708.3	708.30	10.396	1.147
	3	7	128	17081.87	6052.25	1200.7	840.49	4.70	3.623	17261.09	6115.75	1404.0	982.80	15.502	1.099
	4	6	128	21430.48	7593.00	2041.3	1224.78	5.87	3.385	21588.54	7649.00	2314.0	1388.40	20.159	0.986
AES 128	1	11	128	41945.10	14861.51	377.5	415.25	1.91	5.674	42520.86	15065.50	527.9	580.69	6.687	1.621
	2	6	128	76551.96	27123.00	912.5	547.50	3.51	5.660	77491.81	27456.00	1399.0	839.40	13.099	1.517
	3	5	128	111647.00	39557.47	1686.8	843.40	4.68	5.094	112518.00	39866.07	2671.0	1335.50	19.024	1.253
	4	4	128	146191.00	51796.70	2745.7	1098.28	6.22	4.791	147443.00	52240.29	4623.0	1849.20	23.849	1.250
Midori 128	1	21	128	6971.33	2470.00	92.4	194.08	1.33	4.268	7622.59	2700.75	120.0	252.00	2.957	1.920
	2	11	128	10648.98	3773.02	147.6	162.36	1.55	6.992	11077.92	3925.00	200.2	220.22	4.550	2.382
	3	8	128	13824.12	4898.00	248.5	198.80	1.94	7.681	14636.96	5186.00	361.1	288.88	6.350	2.347
	4	6	128	17199.71	6094.00	360.6	216.36	2.42	8.210	17813.58	6311.50	466.1	279.66	7.438	2.671
GIFT 128	1	41	128	6807.63	2412.00	107.1	439.11	0.72	4.038	7641.65	2707.50	132.1	541.61	2.334	1.246
	2	21	128	9042.26	3203.75	151.1	317.31	0.76	7.469	10363.85	3672.00	186.6	391.86	2.246	2.527
	3	15	128	11317.12	4009.75	222.7	334.05	1.40	5.677	12520.17	4436.00	282.6	423.90	3.536	2.248
	4	11	128	13330.90	4723.25	315.6	347.16	1.46	7.423	14834.53	5256.00	407.4	448.14	4.408	2.459
SKINNY 128/128	1	41	128	6509.16	2306.25	123.7	507.17	1.09	2.667	7333.30	2598.25	147.4	604.34	2.951	0.985
	2	21	128	8969.59	3178.00	197.3	414.33	2.08	2.729	9805.02	3474.00	239.1	502.11	6.010	0.945
	3	15	128	13339.37	4726.25	440.3	660.45	3.86	2.059	14533.24	5149.25	533.5	800.25	9.660	0.823
	4	11	128	19126.70	6776.75	794.4	873.84	5.00	2.167	20176.63	7148.75	1013.0	1114.30	14.778	0.733

TABLE I: Simulation results for average power and energy computed pre and post the place and route process.

[3] S. Banik, A. Bogdanov, and F. Regazzoni, "Exploring energy efficiency of lightweight block ciphers," in *Selected Areas in Cryptography - SAC 2015 - 22nd International Conference, Sackville, NB, Canada, August 12-14, 2015, Revised Selected Papers*, 2015, pp. 178–194. [Online]. Available: https://doi.org/10.1007/978-3-319-31301-6_10

[4] S. Banik, V. Mikhalev, F. Armknecht, T. Isobe, W. Meier, A. Bogdanov, Y. Watanabe, and F. Regazzoni, "Towards low energy stream ciphers," *IACR Trans. Symmetric Cryptol.*, vol. 2018, no. 2, pp. 1–19, 2018. [Online]. Available: https://doi.org/10.13154/tosc.v2018.i2.1-19

[5] A. Caforio, F. Balli, and S. Banik, "Energy analysis of lightweight AEAD circuits," *IACR Cryptol. ePrint Arch.*, vol. 2020, p. 607, 2020. [Online]. Available: https://eprint.iacr.org/2020/607

[6] J. Daemen and V. Rijmen, *The Design of Rijndael: AES - The Advanced Encryption Standard*. Berlin, Heidelberg, New York: Springer Verlag, 2002.

[7] A. Bogdanov, L. R. Knudsen, G. Leander, C. Paar, A. Poschmann, M. J. B. Robshaw, Y. Seurin, and C. Vikkelsoe, "PRESENT: an ultra-lightweight block cipher," in *Cryptographic Hardware and Embedded Systems - CHES 2007, 9th International Workshop, Vienna, Austria, September 10-13, 2007, Proceedings*, 2007, pp. 450–466. [Online]. Available: https://doi.org/10.1007/978-3-540-74735-2_31

[8] R. Beaulieu, D. Shors, J. Smith, S. Treatman-Clark, B. Weeks, and L. Wingers, "SIMON and SPECK: Block Ciphers for the Internet of Things." *IACR Cryptology ePrint Archive*, vol. 2015, p. 585, 2015.

[9] S. Banik, A. Bogdanov, T. Isobe, K. Shibutani, H. Hiwatari, T. Akishita, and F. Regazzoni, "Midori: A block cipher for low energy," in *Advances in Cryptology - ASIACRYPT 2015 - 21st International Conference on the Theory and Application of Cryptology and Information Security, Auckland, New Zealand, November 29 - December 3, 2015, Proceedings, Part II*, 2015, pp. 411–436. [Online]. Available: https://doi.org/10.1007/978-3-662-48800-3_17

[10] J. Daemen, M. Peeters, G. V. Assche, and V. Rijmen, "Nessie Proposal: NOEKEON. Available at http://gro.noekeon.org/Noekeon-spec.pdf."

[11] J. Guo, T. Peyrin, A. Poschmann, and M. J. B. Robshaw, "The LED block cipher," in *Cryptographic Hardware and Embedded Systems - CHES 2011 - 13th International Workshop, Nara, Japan, September 28 - October 1, 2011. Proceedings*, 2011, pp. 326–341. [Online]. Available: https://doi.org/10.1007/978-3-642-23951-9_22

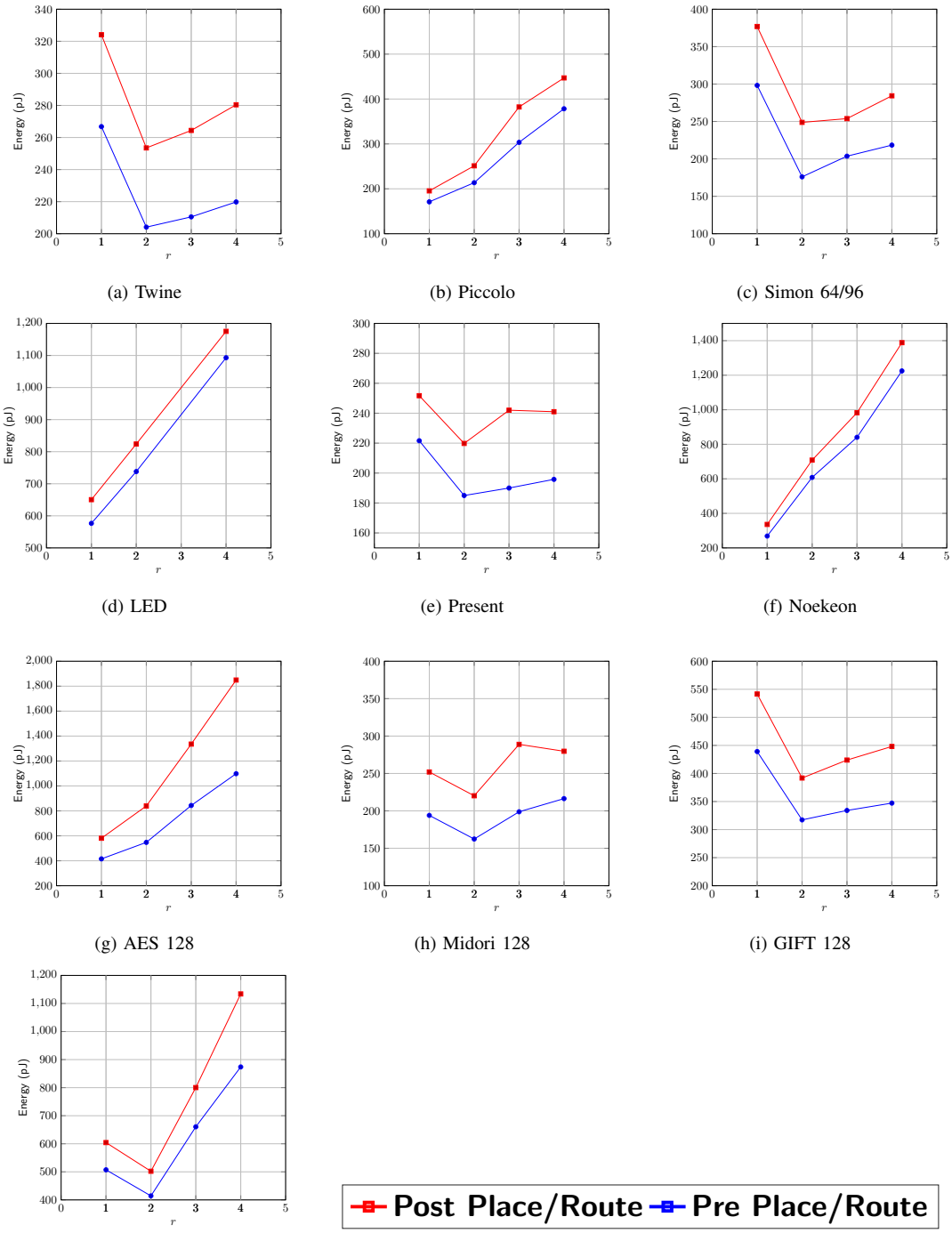
[12] J. Borghoff, A. Canteaut, T. Güneysu, E. B. Kavun, M. Knezevic, L. R. Knudsen, G. Leander, V. Nikov, C. Paar, C. Rechberger, P. Rombouts, S. S. Thomsen, and T. Yalçin, "PRINCE - A low-latency block cipher for pervasive computing applications - extended abstract," in *Advances in Cryptology - ASIACRYPT 2012 - 18th International Conference on the Theory and Application of Cryptology and Information Security, Beijing, China, December 2-6, 2012. Proceedings*, 2012, pp. 208–225. [Online]. Available: https://doi.org/10.1007/978-3-642-34961-4_14

[13] T. Suzaki, K. Minematsu, S. Morioka, and E. Kobayashi, "TWINE : A Lightweight Block Cipher for Multiple Platforms." in *Selected Areas in Cryptography*, ser. Lecture Notes in Computer Science, L. R. Knudsen and H. Wu, Eds., vol. 7707. Springer, 2012, pp. 339–354.

[14] K. Shibutani, T. Isobe, H. Hiwatari, A. Mitsuda, T. Akishita, and T. Shirai, "Piccolo: An Ultra-Lightweight Blockcipher." in *CHES*, ser. Lecture Notes in Computer Science, B. Preneel and T. Takagi, Eds., vol. 6917. Springer, 2011, pp. 342–357.

[15] C. Beierle, J. Jean, S. Kölbl, G. Leander, A. Moradi, T. Peyrin, Y. Sasaki, P. Sasdrich, and S. M. Sim, "The SKINNY family of block ciphers and its low-latency variant MANTIS," in *Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part II*, 2016, pp. 123–153. [Online]. Available: https://doi.org/10.1007/978-3-662-53008-5_5

[16] S. Banik, S. K. Pandey, T. Peyrin, Y. Sasaki, S. M. Sim, and Y. Todo, "GIFT: A small present - towards reaching the limit of lightweight encryption," in *Cryptographic Hardware and Embedded Systems - CHES 2017 - 19th International Conference, Taipei, Taiwan, September 25-28, 2017, Proceedings*, 2017, pp. 321–345. [Online]. Available: https://doi.org/10.1007/978-3-319-66787-4_16



(a) Twine (b) Piccolo (c) Simon 64/96

(d) LED (e) Present (f) Noekeon

(g) AES 128 (h) Midori 128 (i) GIFT 128

(j) SKINNY 128/128

Fig. 5: Energy consumptions computed pre and post placement and routing for increasing degree of unrolling r. Top half lists the ciphers with blocksize equal to 64 bits and the bottom half with blocksize 128 bits.