GIFT-COFB: NIST LWC Second-round Candidate Status Update

Subhadeep Banik[1], Avik Chakraborti[2], Tetsu Iwata[3], Kazuhiko Minematsu[4], Mridul Nandi[5], Thomas Peyrin[6,7], Yu Sasaki[2], Siang Meng Sim[6] and Yosuke Todo[2]

[1] LASEC, Ecole Polytechnique Fédérale de Lausanne, Switzerland
[2] NTT Secure Platform Laboratories, Japan
[3] Nagoya University, Japan
[4] NEC Corporation, Japan
[5] Indian Statistical Institute, Kolkata, India
[6] Nanyang Technological University, Singapore
[7] Temasek Laboratories@NTU, Singapore

giftcofb@googlegroups.com
https://www.isical.ac.in/~lightweight/COFB/

Introduction

This is a short update on our submission GIFT-COFB. Most of the new information have been uploaded on Cryptology ePrint Archive 2020/738 [4].

1 New proofs/arguments supporting the security claims

We have an updated security proof in [4, §5.1]. The published proof in [9] does not directly cover the NIST submission due to differences in the specification (for instance, the specification in [9] fixes the nonce length to $n/2$ bits), and the updated security proof in [4, §5.1] directly covers the NIST submission. The proved security bound is essentially the same as the one presented in the Round-2 submission document [3], where the difference comes from minor updates in case analyses of the security proof.

2 New software and hardware implementations

2.1 Software

Regarding software implementations, we have found a new representation of the GIFT-64 and GIFT-128 bit permutations that makes it efficient and simple to implement in software. This strategy, named *fix-slicing* [1], indeed leads to very efficient one-block constant-time GIFT-128 implementations on 32-bit architectures such as ARM Cortex-M family of processors (79 cycles/ byte on ARM Cortex-M3), making GIFT-COFB one of the most efficient candidate according to microcontroller benchmarks [16, 17]. Using smaller architecture will not be an issue as we will actually save more operations comparatively, since part of the bit permutation can be done by proper unrolling and register scheduling. This is confirmed with 8-bit AVR benchmarks [16, 17] where GIFT-COFB is again ranked among

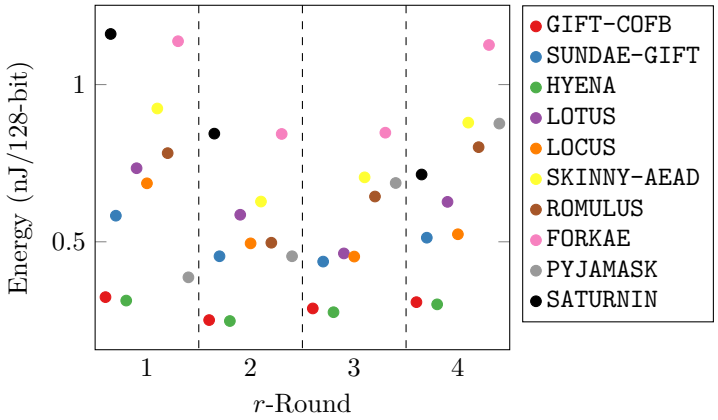

Figure 1: Energy consumption (nJ/128-bit) comparison chart for the r-round partially-unrolled implementations with $r \in \{1, 2, 3, 4\}$. For each candidate the best obtained energy value obtained through techniques is used.

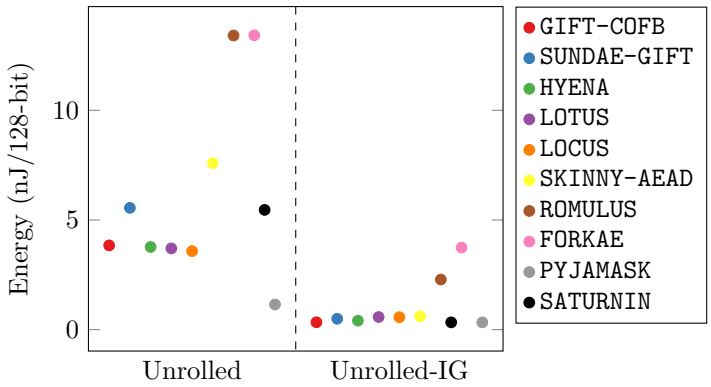

Figure 2: Energy consumption (nJ/128-bit) comparison chart for the fully-unrolled implementations with and without inverse-gating.

the top candidates. Note that using exactly this implementation will also provide decent performance on recent high-end processors (and excellent performances if parallel computations of `GIFT-COFB` instances are considered and vector instructions are used).

These results are particularly surprising, since `GIFT-128` was originally designed for hardware performances in mind.

2.2 Hardware

We have new hardware and threshold implementations in [4, §6], and software implementations in [4, §7]. Some more results were recently published in [6] that compares the the energy efficiency of `GIFT-COFB` with 9 other modes of operation using lightweight block ciphers in the NIST LWC. Because of the excellent energy characteristics of `GIFT-128` and the fact that `GIFT-COFB` is a rate 1 mode, `GIFT-COFB` was found to be one of the most energy efficient designs in the NIST LWC. We experimented with different round unrolled architectures of the core block cipher used in the design (from round-based to fully unrolled) using the TSMC 90nm standard cell library. Figure 1 charts the optimal energy per 128-bit block value for each degree of unrolling r and candidate. Table 1 details the simulation results. Note that all the charts and tables are taken from [6].

Table 1: Various GIFT-COFB implementations. Latency and energy is given for processing a single authenticated data block followed by eight message blocks.CG denotes clock gated. IG denotes "inverse-gated" implementation as per the generic energy reduction technique explained in [2]

Candidate	Implementation	Latency (cycles)	Area (GE)	TP_{max} (Mbps)	Power (μW)	Energy (nJ/128-bit)
GIFT-COFB	1-Round	400	4710	615.38	69.3	0.363
	1-Round-CG	400	4700	569.17	61.9	0.324
	2-Round	200	5548	1192.55	106.8	0.280
	2-Round-CG	200	5510	952.06	95.5	**0.251**
	3-Round	140	6372	1211.87	159.0	0.293
	3-Round-CG	140	6311	1172.16	156.2	0.288
	4-Round	100	7144	1304.64	237.0	0.314
	4-Round-CG	100	7036	1140.59	232.4	0.308
	Unrolled	10	35735	2015.75	12628.4	3.841
	Unrolled-IG	10	43584	711.15	1107.0	0.337

2.3 Threshold Implementations

The s-box of GIFT-128 belongs to the cubic class $\mathcal{C}_{172}$ which is decomposable into 2 quadratics. The algebraic expressions of the output shares of both the 3 and 4-share TI can be found in [11]. Table 2 lists the simulation results using the same measurement setup as the unshared round-based implementations. It can be seen that GIFT-COFB offers both low area and competitive energy efficiency when compared with other modes of operation.

3 New third-party analysis and its implications

3.1 Third-party analysis on GIFT-128

In short, our underlying 40-round block cipher GIFT-128 [5] remains secure with high security margin. We have summarized the latest third-party cryptanalysis results in Table 3.

[18] is the corrected version of [19] with the 22-round differential cryptanalysis on GIFT-128, the original 23-round attack was invalid.

Although GIFT-128 did not make related-key security claims, third-party analysis [7, 15] have shown that GIFT-128 is actually resistance against related-key attacks.

3.2 Third-party analysis on GIFT-COFB

To the best of our knowledge, there is no valid third-party analysis on GIFT-COFB. But this is not surprising since it has been proven secure!

There was a paper posted on Cryptology ePrint Archive 2020/698 [8] claiming forgery attack on GIFT-COFB, but we have contacted and clarified with the authors that the attack is invalid due to an oversight of the GIFT-COFB specification and the authors have since been withdrawn their paper.

Table 2: Measurements for the 1-round threshold implementations. The schemes using GIFT-128 are colored in light gray whereas, SKINNY-AEAD based schemes are in white. Note that the table has been taken from [6]

Candidate	Conf.	Shares #	Latency (cycles)	Area (GE)	TP_{max} (Mbps)	Power (mW)	Energy (nJ/128-bit)
GIFT-COFB	CG-RB	3	800	16386	208.9	0.214	2.243
	CG-RB	4	400	25850	350.8	0.358	**1.875**
SUNDAE-GIFT	RB	3	1440	13297	145.7	0.215	3.719
	RB	4	720	21848	285.2	0.357	**2.999**
HYENA	CG-RB	3	800	14769	344.9	0.212	2.216
	CG-RB	4	400	24540	497.4	0.358	**1.875**
LOTUS	CG	3	2072	14176	121.7	0.145	3.581
	CG	4	1036	19712	133.0	0.262	**3.232**
LOCUS	CG	3	2072	12366	121.7	0.137	3.362
	CG	4	1036	17597	176.8	0.255	**3.148**
SKINNY-AEAD	CG	3	2240	18501	92.83	0.2264	6.134
ROMULUS	CG-RB	3	2056	13450	130.00	0.1865	4.656
FORKAE	CG	3	3008	17008	76.60	0.2483	8.304
PYJAMASK	CG-RB	3	348	42001	620.2	0.472	1.825
	CG-RB	4	180	64577	927.6	0.814	**1.628**

4 Platforms and metrics in which the candidate performs better than current NIST standards

As a mode-level comparison to GCM, GIFT-COFB offers all-round improvements: GIFT-COFB has a better rate (of 1), requires a minimal primitive of block cipher encryption function, a small state size of $1.5n$ bits which is essentially minimum to achieve the standard birthday bound security. The sole drawback is the lack of parallelizability and is well justified in the context of lightweight cryptography applications. Besides, the underlying block cipher GIFT-128 is much more lightweight than AES-128 on hardware, and is even faster on microcontrollers with a smaller memory consumption, thanks to the recent fix-slicing implementation [1, Table 4]. This makes GIFT-COFB greatly outperform AES-GCM in speed and memory, on (small) hardware and microcontrollers.

5 Target applications and use cases for which the candidate is optimized

GIFT-COFB being a very lightweight and rate 1 mode, it has very low area and power and energy footprint. In terms of energy consumption, it is suited for short as well as long messages and particularly useful for constrained devices like low end smart cards.

Table 3: Summary of third-party analysis result on `GIFT-128`. Rounds with asterisk are optimal results. SK – single-key, RK – related-key, LC – linear cryptanalysis, DC – differential cryptanalysis.

Setting	Rounds	Approach	Probability	Time	Data	Memory	Ref.
Distinguisher							
SK	11	Integral	1	-	2^{127}	-	[10]
SK	9*	LC	2^{-44}	-	-	-	[12]
SK	10*	LC	2^{-52}	-	-	-	[12]
SK	9*	DC	$2^{-45.4}$	-	-	-	[14]
SK	10*	DC	$2^{-49.4}$	-	-	-	[14]
SK	11*	DC	$2^{-54.4}$	-	-	-	[14]
SK	12*	DC	$2^{-60.4}$	-	-	-	[14]
SK	13*	DC	$2^{-67.8}$	-	-	-	[14]
SK	14*	DC	$2^{-79.000}$	-	-	-	[12]
SK	15*	DC	$2^{-85.415}$	-	-	-	[12]
SK	16*	DC	$2^{-90.415}$	-	-	-	[12]
SK	17*	DC	$2^{-96.415}$	-	-	-	[12]
SK	18	DC	2^{-109}	-	-	-	[18]
SK	18*	DC	$2^{-103.415}$	-	-	-	[12]
SK	19	DC	$2^{-110.83}$	-	-	-	[12]
SK	20	DC	$2^{-121.415}$	-	-	-	[13]
SK	21	DC	$2^{-126.4}$	-	-	-	[14]
RK	7	DC	$2^{-15.83}$	-	-	-	[7]
RK	10	DC	$2^{-72.66}$	-	-	-	[7]
RK	19	Boomerang	$2^{-121.2}$	-	-	-	[15]
Key-Recovery							
SK	22	DC	2^{-109}	2^{114}	2^{114}	2^{53}	[18]
SK	26	DC	$2^{-121.415}$	$2^{124.415}$	2^{109}	2^{109}	[13]
RK	21	RK-Boomerang	$2^{-121.2}$	$2^{126.6}$	$2^{126.6}$	$2^{126.6}$	[15]

6 Planned tweak proposals

There is no plans to make any tweak if `GIFT-COFB` advances to the final round.

References

[1] Alexandre Adomnicai, Zakaria Najm, and Thomas Peyrin. Fixslicing: A new GIFT representation fast constant-time implementations of GIFT and GIFT-COFB on ARM cortex-m. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2020(3):402–427, 2020.

[2] Subhadeep Banik, Andrey Bogdanov, Francesco Regazzoni, Takanori Isobe, Harunaga Hiwatari, and Toru Akishita. Inverse gating for low energy encryption. In *HOST*, pages 173–176. IEEE Computer Society, 2018.

[3] Subhadeep Banik, Avik Chakraborti, Tetsu Iwata, Kazuhiko Minematsu, Mridul Nandi, Thomas Peyrin, Yu Sasaki, Siang Meng Sim, and Yosuke Todo. GIFT-COFB, NIST LWC Round 2 Submission, 2019. https://csrc.nist.gov/CSRC/media/Projects/lightweight-cryptography/documents/round-2/spec-doc-rnd2/gift-cofb-spec-round2.pdf.

[4] Subhadeep Banik, Avik Chakraborti, Tetsu Iwata, Kazuhiko Minematsu, Mridul Nandi, Thomas Peyrin, Yu Sasaki, Siang Meng Sim, and Yosuke Todo. GIFT-COFB. Cryptology ePrint Archive, Report 2020/738, 2020. https://eprint.iacr.org/2020/738.

[5] Subhadeep Banik, Sumit Kumar Pandey, Thomas Peyrin, Yu Sasaki, Siang Meng Sim, and Yosuke Todo. GIFT: A small present - towards reaching the limit of lightweight encryption. In *CHES*, volume 10529 of *Lecture Notes in Computer Science*, pages 321–345. Springer, 2017.

[6] Andrea Caforio, Fatih Balli, and Subhadeep Banik. Energy analysis of lightweight AEAD circuits. Accepted in Ceryptography and Network Security (CANS) 2020.

[7] Meichun Cao and Wenying Zhang. Related-key differential cryptanalysis of the reduced-round block cipher GIFT. *IEEE Access*, 7:175769–175778, 2019.

[8] Zhe CEN, Xiutao FENG, Zhangyi Wang, and Chunping CAO. (–Withdrawn–) Forgery attack on the authentication encryption GIFT-COFB. Cryptology ePrint Archive, Report 2020/698, 2020. https://eprint.iacr.org/2020/698.

[9] Avik Chakraborti, Tetsu Iwata, Kazuhiko Minematsu, and Mridul Nandi. Blockcipher-based authenticated encryption: How small can we go? *J. Cryptology*, 33(3):703–741, 2020.

[10] Zahra Eskandari, Andreas Brasen Kidmose, Stefan Kölbl, and Tyge Tiessen. Finding integral distinguishers with ease. In *SAC*, volume 11349 of *Lecture Notes in Computer Science*, pages 115–138. Springer, 2018.

[11] Arpan Jati, Naina Gupta, Anupam Chattopadhyay, Somitra Kumar Sanadhya, and Donghoon Chang. Threshold implementations of GIFT: A trade-off analysis. *IEEE Trans. Inf. Forensics Secur.*, 15:2110–2120, 2020.

[12] Fulei Ji, Wentao Zhang, and Tianyou Ding. Improving matsui's search algorithm for the best differential/linear trails and its applications for des, DESL and GIFT. *IACR Cryptol. ePrint Arch.*, 2019:1190, 2019.

[13] Lingchen Li, Wenling Wu, Yafei Zheng, and Lei Zhang. The relationship between the construction and solution of the MILP models and applications. *IACR Cryptol. ePrint Arch.*, 2019:49, 2019.

[14] Yu Liu, Huicong Liang, Muzhou Li, Luning Huang, Kai Hu, Chenhe Yang, and Meiqin Wang. STP models of optimal differential and linear trail for s-box based ciphers. *IACR Cryptol. ePrint Arch.*, 2019:25, 2019.

[15] Yunwen Liu and Yu Sasaki. Related-key boomerang attacks on GIFT with automated trail search including BCT effect. In *ACISP*, volume 11547 of *Lecture Notes in Computer Science*, pages 555–572. Springer, 2019.

[16] Sebastian Renner, Enrico Pozzobon, and Jürgen Mottok. NIST LWC Software Performance Benchmarks on Microcontrollers, 2020. https://lwc.las3.de/.

[17] Rhys Weatherley. Lightweight Cryptography Primitives, 2020. https://rweather.github.io/lightweight-crypto/index.html.

[18] Baoyu Zhu, Xiaoyang Dong, and Hongbo Yu. Milp-based differential attack on round-reduced GIFT. *IACR Cryptol. ePrint Arch.*, 2018:390, 2018.

[19] Baoyu Zhu, Xiaoyang Dong, and Hongbo Yu. Milp-based differential attack on round-reduced GIFT. In *CT-RSA*, volume 11405 of *Lecture Notes in Computer Science*, pages 372–390. Springer, 2019.