CRYPTREC 暗号技術ガイドライン（軽量暗号）2023 年度版

CRYPTREC 暗号技術評価委員会

2024 年 3 月

CRYPTREC 軽量暗号 WG 委員構成（2013 年度〜2016 年度）

主査	本間　尚文	国立大学法人東北大学
委員	青木　和麻呂	日本電信電話株式会社
委員	岩田　哲	国立大学法人名古屋大学
委員	小川　一人	日本放送協会
委員	小熊　寿	株式会社トヨタ IT 開発センター
委員	崎山　一男	国立大学法人電気通信大学
委員	渋谷　香士	ソニーグローバルマニュファクチャリング＆オペレーションズ株式会社
委員	鈴木　大輔	三菱電機株式会社
委員	成吉　雄一郎	ルネサスエレクトロニクス株式会社
委員	峯松　一彦	日本電気株式会社
委員	三宅　秀享	株式会社東芝
委員	渡辺　大	株式会社日立製作所

ガイドライン改定のための外部評価依頼先（2021 年度〜2023 年度）

井上　明子	日本電気株式会社
伊藤　竜馬	国立研究開発法人情報通信研究機構
岩田　哲	国立大学法人東海国立大学機構 名古屋大学
菅野　哲	GMO サイバーセキュリティ by イエラエ株式会社
崎山　一男	国立大学法人電気通信大学
藤堂　洋介	日本電信電話株式会社
内藤　祐介	三菱電機株式会社
本間　尚文	国立大学法人東北大学
峯松　一彦	日本電気株式会社

改定履歴

「CRYPTREC 暗号技術ガイドライン（軽量暗号）」（以下、「2016 年度版ガイドライン」という）

- 新規発行 – 2017 年 6 月 30 日（第 1 版）

「CRYPTREC 暗号技術ガイドライン（軽量暗号）2023 年度版」（以下、「2023 年度版ガイドライン」という）

- 追加
  - 2.2 節：軽量暗号の標準化動向（CAESAR コンペティション、NIST LWC プロジェクト）を追加
  - 3.3 節：NIST LWC 最終選考方式 Ascon の実装性能に関する評価結果を追加
  - 4.1 節：軽量ブロック暗号 LEA に関する情報を追加
  - 4.3 節：軽量ハッシュ関数 Lesamnta-LW に関する情報を追加
  - 4.4 節：軽量メッセージ認証コード Chaskey、LightMAC、Tsudik's keymode に関する情報を追加
  - 4.5 節：軽量認証暗号 Grain-128A に関する情報を追加
  - 付録 A：Ascon に対するサイドチャネル攻撃対策手法と物理攻撃手法に関する情報を追加
  - 付録 B：CAESAR final portfolio である AEGIS、COLM に関する情報を追加
  - 付録 C：NIST LWC ファイナリスト（Ascon を除く 9 方式）に関する情報を追加
- 改定
  - 2.1 節：本ガイドラインで紹介する代表的な軽量暗号技術（表 2.1）を改定
  - 4 章：2016 年度版ガイドライン掲載の代表的な軽量暗号に関する安全性解析状況と標準化状況を改定
  - 4.5 節：軽量認証暗号 Ascon に関する情報を改定

目次

第 1 章　はじめに 1

第 2 章　軽量暗号とその活用法 4

2.1 軽量暗号とは . . . 4

2.2 軽量暗号の標準化動向 . . . 5

2.2.1 CAESAR コンペティション . . . 5

2.2.2 NIST LWC プロジェクト . . . 6

2.2.2.1 第 1 ラウンド . . . 7

2.2.2.2 第 2 ラウンド . . . 7

2.2.2.3 最終ラウンド . . . 8

2.2.2.4 Ascon に関する評価 . . . 9

2.2.3 他標準化団体における Ascon の検討状況 . . . 10

2.3 軽量暗号はどこに使えるのか . . . 11

2.3.1 家電・スマートテレビ . . . 12

2.3.2 RFID タグ利用のアプリケーション（物流管理等） . . . 12

2.3.3 センサーを利用したスマート農業 . . . 13

2.3.4 医療 . . . 13

2.3.5 産業用システム . . . 14

2.3.6 自動車 . . . 14

2.4 どんな軽量暗号、パラメータを選べばいいか . . . 15

2.4.1 一般的方針 . . . 15

2.4.2 鍵長の選択 . . . 15

2.4.3 ブロック長の選択 . . . 15

2.4.4 処理データ量と鍵更新、その他の対策 . . . 15

2.4.5 利用シナリオ . . . 16

2.4.6 その他の留意点 . . . 16

2.4.7 CRYPTREC 暗号リストの暗号との違い . . . 17

2.5 軽量暗号活用例と効果 . . . 17

2.5.1 家電・スマートテレビ . . . 17

2.5.2 RFID タグ利用のアプリケーション（物流管理等） . . . 17

2.5.3 センサーを利用したスマート農業 . . . 17

2.5.4 医療 . . . 18

2.5.5 産業用システム . . . 18

2.5.6 自動車 . . . 18

第 3 章　軽量暗号の実装性能 21

3.1 ブロック暗号の実装性能 . . . 22

3.1.1 ハードウェア実装評価 . . . 22

3.1.1.1 性能比較 . . . 22
3.1.1.2 評価方法の概要 . . . 23
3.1.2 ソフトウェア実装評価 . . . 38
3.1.2.1 性能評価 . . . 38
3.1.2.2 性能比較 . . . 45
3.1.2.3 評価方法の概要 . . . 55
3.2 認証暗号の実装性能 . . . 56
3.2.1 ソフトウェア実装評価 . . . 56
3.2.1.1 性能比較 . . . 56
3.2.1.2 評価方法の概要 . . . 61
3.3 Ascon の実装性能 . . . 73
3.3.1 ハードウェア実装性能 . . . 73
3.3.1.1 調査対象と性能評価環境 . . . 73
3.3.1.2 実装性能 . . . 74
3.3.2 ソフトウェア実装性能 . . . 83
3.3.2.1 調査対象と性能評価環境 . . . 83
3.3.2.2 実装性能 . . . 83
3.3.3 物理攻撃耐性 . . . 84
3.3.3.1 用語 . . . 84
3.3.3.2 サイドチャネル攻撃対策が施された実装への評価結果 . . . 84
3.3.3.3 物理攻撃耐性評価 . . . 87

第 4 章 代表的な軽量暗号 94
4.1 ブロック暗号 . . . 94
4.2 ストリーム暗号 . . . 116
4.3 ハッシュ関数 . . . 127
4.4 メッセージ認証コード . . . 137
4.5 認証暗号 . . . 147

付録 A Ascon の物理攻撃耐性 168
A.1 サイドチャネル攻撃対策手法 . . . 168
A.1.1 Threshold Implementation (TI) . . . 168
A.1.2 Domain Oriented Masking (DOM) . . . 169
A.2 サイドチャネル解析・漏えい評価手法 . . . 170
A.2.1 相関電力解析（CPA: Correlation Power Analysis） . . . 170
A.2.2 故障利用攻撃（FA: Fault Attack） . . . 170
A.2.3 Test Vector Leakage Assessment (TVLA) . . . 170
A.2.4 テンプレート攻撃（TA: Templete Attack） . . . 171

付録 B CAESAR final portfolio: AEGIS, COLM 174

付録 C NIST LWC ファイナリスト（Ascon を除く） 178

第 1 章

はじめに

限られた実装環境でも安全で高性能な暗号技術を搭載したいというニーズは古くからあり、このニーズに応えて小型で高速な暗号技術の研究開発が進められてきた。昨今は、IoT (Internet of Things) の発展により、センサーやアクチュエータ等の計算リソースの限られたデバイスがネットワークに接続され、セキュリティやプライバシー上の脅威が高まっていることから、暗号技術に対してより多様な実装要件が求められている。

計算リソースの限られたデバイスにも実装可能な「軽量暗号」は、車載機器や医療機器をはじめとする様々な用途での利用が期待されているが、どの軽量暗号を選べばいいのか、運用時にどのようなことに注意すればいいのか、など実際に利用する際には専門家以外では判断が困難な場合も多い。

CRYPTREC では、主として電子政府で利用する暗号技術について検討を行っているが、それに加えて、今後さまざまな領域で利用が想定される暗号技術について技術調査を行い、社会に役立つ形で情報提供を行うことを目指している。特に、軽量暗号技術が求められる製品やサービスにおいて、利用者が最適な暗号方式を選択でき、容易に調達できることを目指し、2013 年度より CRYPTREC 暗号技術評価委員会の下に軽量暗号ワーキンググループ（WG）が設置された。2016 年度版ガイドライン [1, 2] は、軽量暗号の方式を選択・利用する際の技術的判断に資すること、今後の利用促進をはかることを目的として、軽量暗号 WG が作成したものである。その後、軽量暗号に関する研究開発の最新動向や米国 NIST (National Institute of Standards and Technology) 等の標準化動向を踏まえ、2021 年度から 2023 年度にかけて軽量暗号の安全性、実装性能、標準化動向に関する技術動向調査・評価を国内有識者に依頼し、これらの調査結果・評価結果を外部評価報告書として CRYPTREC ホームページに公開した [3, 4, 5, 6, 7, 8, 9, 10, 11]。2023 年度版ガイドラインは、これらの外部評価報告書に基づいて CRYPTREC 事務局が執筆・編集し、2016 年度版ガイドラインの改定版として公開するものである。本ガイドラインの主たる読者として、情報システムのセキュリティ機能の設計・開発・実装において暗号技術を活用する技術者を想定しているが、軽量暗号技術に興味をもつ方に広く読んで頂ければ幸いである。

1 章は、本ガイドラインの総説である。2 章では、軽量暗号を概説する。本ガイドラインの対象となる軽量暗号技術の概要、軽量暗号の標準化動向、そして軽量暗号の活用例を示し、その上で、軽量暗号を実際に活用する際の手引きを示している。特に、軽量暗号の特徴、代表的なユースケース、方式、パラメータの選択方法、使用時の留意点などを記載している。3 章では、代表的な軽量暗号の実装性能を示す。多くの軽量暗号方式が提案されているブロック暗号と認証暗号の技術分野において、代表的ないくつかの方式を取り上げ、これらの方式の実装性能を比較している。性能比較は、ハードウェア実装とソフトウェア実装で、それぞれ同一の実装環境下で行っている。ハードウェア実装では回路規模、消費電力、レイテンシの比較結果を、ソフトウェア実装では必要なメモリサイズの比較結果を示している。加えて、NIST が主催する軽量暗号標準化プロジェクトで最終選考方式に選定された Ascon の実装性能を示す。特に、サイドチャネル攻撃等への対策を施さない場合と対策を施した場合におけるソフトウェア実装とハードウェア実装の性能、そしてサイドチャネル攻撃耐性の評価結果について、公開されている情報に基づき記載している。4 章では、代表的な軽量暗号技術の基本情報をブロック暗号、ストリーム暗号、ハッシュ関数、メッセージ認証コード、認証暗号の技術分野別に示す。

本ガイドラインで紹介している軽量暗号技術は、執筆時点までに主要国際学会で発表されており、有力な攻撃法が発見されておらず、かつ十分な実装性能を持つと考えられる方式を選んでいる。その上で、可能な限り最新の情報に基づき安全性や実装性能、標準化動向等を紹介している。しかしながら、軽量暗号の研究開発は今まさに盛んに行われており、年々新たな方式や評価結果が出ていることから、記載内容が執筆時点（特に断りがない限り、2023 年 9 月現在）のものであることに

留意いただきたい。

2016 年度版ガイドライン策定に関する謝辞

2016 年度版ガイドラインは以下に示す軽量暗号 WG 委員および CRYPTREC 事務局で執筆・編集を行いました。所属は 2016 年 10 月時点のものです。また、2016 年度版ガイドラインを執筆する上で、軽量暗号の応用に関して株式会社日立製作所の大和田徹氏に、軽量暗号の実装評価に関して三菱電機株式会社の松井充氏、菅原健氏、村上ユミコ氏、梨本翔永氏に、それぞれ多大なご貢献をいただきました。この場を借りて皆様に深く感謝申し上げます。

主査	本間　尚文	国立大学法人東北大学
委員	青木　和麻呂	日本電信電話株式会社
委員	岩田　哲	国立大学法人名古屋大学
委員	小川　一人	日本放送協会
委員	小熊　寿	株式会社トヨタ IT 開発センター
委員	崎山　一男	国立大学法人電気通信大学
委員	渋谷　香士	ソニーグローバルマニュファクチャリング＆オペレーションズ株式会社
委員	鈴木　大輔	三菱電機株式会社
委員	成吉　雄一郎	ルネサスエレクトロニクス株式会社
委員	峯松　一彦	日本電気株式会社
委員	三宅　秀享	株式会社東芝
委員	渡辺　大	株式会社日立製作所
事務局	盛合　志帆	国立研究開発法人情報通信研究機構
事務局	大久保　美也子	国立研究開発法人情報通信研究機構
事務局	金森　祥子	国立研究開発法人情報通信研究機構

2023 年度版ガイドライン策定に関する謝辞

2023 年度版ガイドラインは以下に示す有識者の皆様にご執筆いただいた外部評価報告書に基づき、CRYPTREC 事務局にて執筆・編集を行いました。また、以下に示す有識者の皆様に 2023 年度版ガイドライン（ドラフト版）の改定内容についてレビューいただくとともに、改定内容の認識誤りなどについてご指摘いただきました。所属は 2024 年 3 月時点のものです。この場を借りて皆様に深く感謝申し上げます。

外部評価 [6]	井上　明子	日本電気株式会社
外部評価 [5]、ガイドライン執筆・編集	伊藤　竜馬	国立研究開発法人情報通信研究機構
外部評価 [7]	岩田　哲	国立大学法人東海国立大学機構 名古屋大学
外部評価 [3, 4]	菅野　哲	GMO サイバーセキュリティ by イエラエ株式会社
外部評価 [8, 9]	崎山　一男	国立大学法人電気通信大学
外部評価 [3, 4]	酒見　由美	GMO サイバーセキュリティ by イエラエ株式会社
外部評価 [10]	藤堂　洋介	日本電信電話株式会社
外部評価 [11]	内藤　祐介	三菱電機株式会社
ガイドライン（ドラフト版）のレビュー	本間　尚文	国立大学法人東北大学
ガイドライン（ドラフト版）のレビュー	峯松　一彦	日本電気株式会社

参考文献

[1] CRYPTREC Lightweight Cryptography Working Group: CRYPTREC Cryptographic Technology Guideline (Lightweight Cryptography) (Document ID: CRYPTREC GL-2003-2016EN) (2017), https://www.cryptrec.go.jp/report/cryptrec-gl-2003-2016en.pdf

[2] CRYPTREC 軽量暗号ワーキンググループ: CRYPTREC 暗号技術ガイドライン（軽量暗号）（文書番号：CRYPTREC GL-2003-2016JP）(2017), https://www.cryptrec.go.jp/report/cryptrec-gl-2003-2016jp.pdf

[3] GMO サイバーセキュリティ by イエラエ株式会社: 軽量暗号の評価指標、標準化動向に関する調査（NIST 軽量暗号コンペティションファイナリストなど）（文書番号：CRYPTREC EX-3206-2022）(2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3206-2022.pdf

[4] GMO サイバーセキュリティ by イエラエ株式会社: 軽量暗号 Ascon などに関わる標準化動向調査（文書番号：CRYPTREC EX-3302-2023）(2023)

[5] 伊藤竜馬:「CRYPTREC 暗号技術ガイドライン（軽量暗号）」掲載の暗号方式に関する安全性評価の動向調査（文書番号：CRYPTREC EX-3101-2021）(2021), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3101-2021.pdf

[6] 井上明子: 軽量暗号の安全性に関する調査及び評価（Elephant,ISAP,Romulus）（文書番号：CRYPTREC EX-3204-2022）(2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3204-2022.pdf

[7] 岩田哲: 軽量暗号の安全性に関する調査及び評価（Photon-Beetle,Sparkle,Tsudik's keymode）（文書番号：CRYPTREC EX-3201-2022）(2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3201-2022.pdf

[8] 崎山一男: 軽量暗号の実装性能に関する調査及び評価（NIST 軽量暗号コンペティションファイナリスト）（文書番号：CRYPTREC EX-3205-2022）(2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3205-2022.pdf

[9] 崎山一男: 軽量暗号 Ascon の実装性能に関する調査及び評価（文書番号：CRYPTREC EX-3301-2023）(2023)

[10] 藤堂洋介: 軽量暗号の安全性に関する調査及び評価（Ascon,Grain-128AEAD,TinyJambu）（文書番号：CRYPTREC EX-3203-2022）(2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3203-2022.pdf

[11] 内藤祐介: 軽量暗号の安全性に関する調査及び評価（GIFT-COFB,Xoodyak）（文書番号：CRYPTREC EX-3202-2022）(2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3202-2022.pdf

第 2 章

軽量暗号とその活用法

2.1 軽量暗号とは

近年、計算リソースの限られたデバイスにも実装可能な「軽量暗号」の研究開発が進展しており、多くの方式が学会等で提案されている。欧州では 2004 年から European Commission の第 6–7 次 Framework Programme の研究プロジェクト ECRYPT I と ECRYPT II のテーマとしても取り上げられてきた。日本も小型実装に適した暗号技術等で強みを持っている分野である。軽量暗号の標準化も進んでおり、軽量暗号アルゴリズムを技術分野毎に定めた ISO/IEC 29192 や RFID 向けの暗号技術を定めた ISO/IEC 29167 が策定され、米国 NIST も 2015 年より軽量暗号の標準化の検討を開始している。

低コスト・低消費電力で動作可能な軽量暗号技術は、今後も車載機器や医療機器など様々な機器で利用される可能性があり、IoT や CPS (Cyber Physical System) といった次世代のネットワークサービスを構築する上で有効なセキュリティ技術の一つとなることが期待されている。

一方で、一般的に合意されている軽量暗号の定義はない。これまで提案されてきた軽量暗号技術には、ハードウェア実装のサイズ・消費電力量の観点で軽量性を追求したもの、組み込みソフトウェア実装で必要なメモリサイズの軽量性を追求したもの、などの様々な性能指標で最適化された方式が存在する。また、性能と安全性のトレードオフもあり、実際の性能は多岐に渡っている。このような状況を踏まえ、本ガイドラインでは「実装性能と安全性のトレードオフを勘案した上で、従来の暗号技術に対して特定の性能指標で優位性（軽量性）を持つように設計された暗号技術」をスコープとし、用途が想定される代表的な性能指標に対して優位性を主張する暗号を主な対象とする。また、現時点で、公開鍵暗号系において軽量暗号として広くコンセンサスがとれている方式はほとんどないため、本ガイドラインでは共通鍵暗号系の軽量暗号を対象としている。

本ガイドラインでは、下記を軽量暗号の代表的な性能指標とする。

- ハードウェア実装における回路規模、消費電力量、レイテンシ
- 組み込みソフトウェア実装におけるメモリサイズ（ROM/RAM）

■回路規模　ハードウェア実装の回路規模は半導体のコストに直結し、また、消費電力の指標にもなり得ることが知られている。回路規模の小型化は、RFID をはじめとする回路実装面積の要求条件が厳しいアプリケーションで重要な要件である。また、非接触 IC カードのようにバッテリーや外部供給電源がなく、電磁誘導等で駆動するデバイスにおいても重要な要件である。

■消費電力量　消費電力量の低減は、人体へ埋め込まれたり密着装備される医療機器をはじめ、バッテリーで駆動するあらゆるデバイスで求められる要件である。

■レイテンシ　本ガイドラインにおいて、レイテンシ（遅延時間）は 1 回の暗号化（復号）処理に必要な時間を意味する。低遅延性はメモリ暗号化や車載機器などのリアルタイム性が求められるアプリケーションで必須の要件である。

■メモリサイズ　組み込みソフトウェア実装では、組み込みマイコン上で実現される様々なアプリケーションの一部として、暗号機能を実装することが多い。組み込みマイコンでは、ROM や RAM のサイズが限られており、小さく実装できる暗号ほど、選択できるマイコンの幅が広がり、コストを下げられる等の利点がある。組み込みマイコンは家電機器やセンサー、

車載向け等で広く利用されており、実装に必要なメモリサイズ（ROM/RAM）が少ないことはこれらのアプリケーションで重要な要件である。

性能指標	アプリケーションの例
回路規模（消費電力、コスト）	RFID、低コストセンサー
消費電力量	医療機器、バッテリ駆動デバイス
レイテンシ（リアルタイム性能）	メモリ暗号化、車載機器、産業向け I/O デバイス制御
メモリサイズ（ROM/RAM）	家電機器、センサー、車載機器

本ガイドラインでは、軽量暗号の代表的な方式を、ブロック暗号、ストリーム暗号、ハッシュ関数、メッセージ認証コード、認証暗号の技術分野別に分類している。それぞれの技術で実現できる機能（秘匿、認証など）は既存の暗号技術と同じである。

本ガイドラインで紹介する代表的な軽量暗号技術を表 2.1 に示す。これらの軽量暗号技術は、執筆時点までに主要国際学会等で発表されており、有力な攻撃法が発見されておらず、十分な実装性能を持ち、計算リソースの限られた実装条件下で有用と考えられるアルゴリズムを選んでいる。軽量暗号技術の概要については、4 章、付録 C にまとめている。

表 2.1　本ガイドラインで紹介する代表的な軽量暗号技術

ブロック暗号	CLEFIA, LED, Midori, Piccolo、PRESENT, PRINCE, SIMON, SPECK, TWINE, LEA
ストリーム暗号	ChaCha20, Enocoro, Grain v1, MICKEY 2.0, Trivium
ハッシュ関数	Keccak, PHOTON, QUARK, SPONGENT, Lesamnta-LW
メッセージ認証コード	SipHash, Chaskey, LightMAC, Tsudik's keymode
認証暗号	Ascon, ACORN, AES-JAMBU, AES-OTR, CLOC and SILC, Deoxys, Joltik, Ketje, Minalpher, OCB, PRIMATEs, Grain-128A, Elephant, GIFT-COFB, Grain-128AEAD, ISAP, PHOTON-Beetle, Romulus, Sparkle, TinyJAMBU, Xoodyak

なお、AEGIS と COLM の 2 方式については軽量性を謳った方式ではないものの、CAESAR final portfolio に選出された方式であるということを鑑み、付録 B で調査結果をまとめている。

以降、2.2 節では、軽量暗号の選定に関する標準化プロジェクトの動向を記載する。具体的には、CAESAR（Competition for Authenticated Encryption: Security, Applicability, and Robustness）コンペティションと NIST 軽量暗号（LWC: Lightweight Cryptography）プロジェクトについてまとめる。その他、表 2.1 で示す代表的な軽量暗号技術の標準化動向について、標準化されている場合に限り、その標準化状況を 4 章と付録で記載している。次に、2.3 節で軽量暗号がどのような場面で使えるかという活用例を示し、2.4 節で方式や鍵長、ブロック長を選ぶ際の留意点を記載する。最後に、2.5 節では、2.3 節で取り上げる活用例において、どのような点に着目して軽量暗号を選定すればよいか具体例を示す。なお、2.3–2.5 節の記載内容は 2016 年度版ガイドライン執筆時点（2017 年 3 月現在）のものであることに留意いただきたい。

2.2　軽量暗号の標準化動向

軽量暗号の選定に関する代表的なプロジェクトとして広く知られている CAESAR コンペティションと NIST LWC プロジェクトの標準化動向について紹介する。また、NIST LWC プロジェクトの最終選考方式として Ascon が選出されたことを受け、標準化団体における Ascon の検討状況についても紹介する。これらの標準化動向については、2022 年度と 2023 年度に公開された CRYPTREC 外部評価報告書 [11, 12] に基づき、2023 年 9 月現在の調査結果を記載している。

2.2.1　CAESAR コンペティション

CAESAR（Competition for Authenticated Encryption: Security, Applicability, and Robustness）コンペティション [4] は、認証暗号技術に関する設計の促進を図るために国際的な暗号研究コミュニティによって運営されたコンペティショ

ンである。2013 年 1 月に開催された Early Symmetric Crypto workshop[*1]でコンペティション開催についてアナウンスされ、2014 年 3 月に 57 件の応募があった。その後、第 1 ラウンド候補として 48 件に絞り込まれた上で、第 1 ラウンド、第 2 ラウンド、第 3 ラウンド、そして最終ラウンドの 4 回の評価フェーズを経て 2019 年 2 月に最終的なポートフォリオが 6 件発表された。表 2.2 は、CAESAR コンペティションにおける各評価フェーズの期間と評価対象アルゴリズムの数をまとめたものである。

表 2.2　CAESAR コンペティションにおける選定プロセスの動向

評価フェーズ	期間	対象アルゴリズム数
1	2014 年 3 月–2015 年 7 月	57 → 48
2	2015 年 7 月–2016 年 8 月	30
3	2016 年 8 月–2018 年 3 月	15
最終	2018 年 3 月–2019 年 2 月	7

最終的なポートフォリオ数：6

最終的なポートフォリオは、ユースケース 1 の軽量アプリケーション（リソースに制約のある環境）、ユースケース 2 の高性能アプリケーション、ユースケース 3 の多層防御、という 3 部構成となっており、軽量暗号技術に該当するユースケース 1 の第 1 候補として Ascon、第 2 候補として ACORN が選定された。参考までに、ユースケース 2 の第 1 候補は AEGIS-128、第 2 候補は OCB であり、ユースケース 3 の第 1 候補は Deoxis-II、第 2 候補は COLM である。これら 6 件の最終的なポートフォリオの概要については、4.5 節と付録 B でまとめている。

CAESAR コンペティションでは評価基準が明確に示されていないものの、AES-GCM よりも優れた利点を有し、幅広い領域で採用される認証暗号を選択することを目的としている[*2]。また、ソフトウェアとハードウェアの実装性能評価に関し、統一されたフレームワークを使用して測定できる仕組みを導入している[*3]。

2.2.2　NIST LWC プロジェクト

NIST 軽量暗号（LWC: Lightweight Cryptography）プロジェクト [3] は、制約のあるデバイス上などで限定的に使用が認められる軽量暗号アルゴリズムと暗号利用モードのポートフォリオを開発および維持することを目的として開催されたコンペティション形式のプロジェクトである。2013 年、NIST は軽量暗号の標準化に向けたプロジェクトを開始し、2015 年 7 月開催の第 1 回ワークショップと 2016 年 10 月開催の第 2 回ワークショップを経て、2017 年 3 月に NIST は軽量暗号に関するレポート NISTIR 8114 [16] を発行するとともに、オープンプロセスを通じて軽量暗号アルゴリズムのポートフォリオを作成することを決定したとアナウンスした。2018 年 8 月、NIST は軽量暗号標準化プロセスのための応募要件と評価基準 [19] を公開して新しい軽量暗号アルゴリズムの募集を開始し、2019 年 3 月の締切時に 57 件の応募を受け取った。その後、第 1 ラウンド候補として 56 件に絞り込まれた上で、第 1 ラウンド、第 2 ラウンド、そして最終ラウンドの 3 回の評価フェーズを経て、2023 年 2 月 7 日に最終選考方式として Ascon が選定された。表 2.3 は、NIST LWC プロジェクトにおける各評価フェーズの期間と評価対象アルゴリズムの数をまとめたものである。以下、これら 3 回の評価フェーズを概説する。

[*1] https://www.cryptolux.org/mediawiki-esc2013/index.php/ESC_2013

[*2] Call for submissions のページ（https://competitions.cr.yp.to/caesar-call.html）において、“CAESAR (Competition for Authenticated Encryption: Security, Applicability, and Robustness) will identify a portfolio of authenticated ciphers that (1) offer advantages over AES-GCM and (2) are suitable for widespread adoption. Cryptographic algorithm designers are invited to submit proposals of authenticated ciphers to CAESAR. All proposals will be made public for evaluation.” と記載されている。その他、同ページには応募に際して様々な要件（機能要件、ソフトウェア要件、ハードウェア要件、応募要件）が指定されている。

[*3] Submissions のページ（https://competitions.cr.yp.to/caesar-submissions.html）において、“See https://bench.cr.yp.to/supercop.html for software implementations, and https://cryptography.gmu.edu/athena/index.php?id=CAESAR_source_codes for VHDL implementations.” と記載されている。

表 2.3　NIST LWC における選定プロセスの動向

評価フェーズ	期間	対象アルゴリズム数
1	2019 年 3 月–2019 年 8 月	57 → 56
2	2019 年 8 月–2021 年 3 月	32
最終	2021 年 3 月–2023 年 2 月	10

最終選考方式の数：1

2.2.2.1　第 1 ラウンド

NIST は、2019 年 3 月に 57 件の応募を受け取った後、軽量暗号標準化プロセスのための応募要件と評価基準 [19] で示した要件に基づき、この要件に対する完全性と妥当性の観点から選定を行った。2019 年 4 月、NIST は 57 件の応募のうち、第 1 ラウンド候補のアルゴリズムとして 56 件の応募を承認し、第 1 ラウンドの評価フェーズを開始した。2019 年 8 月に第 1 ラウンドの評価フェーズが終了し、第 2 ラウンド候補として 32 件のアルゴリズムが発表された。その後、NIST はステータスレポート NISTIR 8268 [23] を発行し、第 1 ラウンドにおける評価基準や選定プロセスを明確にした。

NISTIR 8268 [23] によると、第 1 ラウンドにおける評価基準や選定プロセスの中で最も重要な観点は暗号アルゴリズムの安全性である。この理由は、以下の 2 点が挙げられる。

1. 第三者による安全性評価が公開されていない、または応募資料において安全性の主張を裏付ける情報が不十分である暗号アルゴリズムについては第 2 ラウンド候補から除外された。
2. 第三者による安全性評価によって安全性上の懸念が生じた暗号アルゴリズムについては第 2 ラウンド候補から除外された。具体的には、偽造攻撃、Length-extension attack、識別攻撃、その他の予期しない性質（Undesirable properties）が存在する暗号アルゴリズムが整理された。

なお、実装のバグによる実用的な攻撃（例えば、偽造攻撃）については、第 2 ラウンド候補から除外されていない。

2.2.2.2　第 2 ラウンド

NIST は、2019 年 8 月に第 2 ラウンド候補となる 32 件のアルゴリズムを発表した後、第 2 ラウンドの評価フェーズを開始した。2021 年 3 月に第 2 ラウンドの評価フェーズが終了し、最終ラウンド候補のアルゴリズム（ファイナリスト）として Ascon、Elephant、GIFT-COFB、Grain-128AEAD、ISAP、Romulus、PHOTON-Beetle、SPARKLE、TinyJAMBU、Xoodyak の 10 件が発表された。その後、NIST はステータスレポート NISTIR 8369 [24] を発行し、第 2 ラウンドにおける評価基準や選定プロセスを明確にした。なお、これら 10 件のファイナリストの概要については、4.5 節と付録 C でまとめている。

NISTIR 8369 [24] によると、第 1 ラウンドと同様、第 2 ラウンドにおいても暗号学的な安全性が最も重要な評価基準となっている。具体的には、以下のいずれかに該当するアルゴリズムがファイナリスト選定時において重要な評価指標となった。

1. 第三者による安全性評価が十分に行われたアルゴリズム
2. 十分に認知されている設計原理と安全性証明に基づく安全性主張が明確であるアルゴリズム

つまり、第三者による安全性評価の結果により、安全性主張の妥当性について懸念が生じたアルゴリズムについては、ファイナリストから除外されている。特に、除外されたアルゴリズムについては、暗号プリミティブ（例えば、暗号学的置換、ブロック暗号、tweakable ブロック暗号、ストリーム暗号、など）に対する識別攻撃や弱鍵クラスの存在が懸念された。加えて、現実的な偽造攻撃や鍵回復攻撃など、安全性主張を無効にする評価結果も提供された。

制約のある環境でのハードウェアとソフトウェアの実装性能についても重要な評価基準となっている。第 2 ラウンド候補は様々な実装性能とコストの観点から評価・比較され、現在の NIST 標準である AES-GCM [9] や SHA-2 [18] よりも著しく性能に優れたアルゴリズムであるということがファイナリスト選定時において重要な評価指標となった。なお、NIST は

公開されている様々なハードウェア・ソフトウェアのベンチマークに加えて、独自のソフトウェアベンチマークを用いて評価した。

追加の評価基準として、サイドチャネル攻撃耐性、nonce-misuse 安全性、releasing unverified plaintext (RUP) 安全性、内部状態復元の影響、耐量子安全性を含む様々な性質を満たすかについても考慮された。

その他、候補の多様性についても考慮された。具体的には、アルゴリズムの根底を担う暗号プリミティブをベースとした複数の有望な候補が存在する場合、暗号プリミティブの種類によってグループ分けされるとともに、グループ内で相互比較され、候補が絞り込まれた。

2.2.2.3 最終ラウンド

NIST は、2021 年 3 月にファイナリストとなる 10 件のアルゴリズムを発表した後、最終ラウンドの評価フェーズを開始した。当初、NIST LWC プロジェクトにおいて、制約のある環境に適した認証暗号とハッシュ機能として 1 つまたは複数のアルゴリズムを選定するために標準化プロセスが開始されたが、2023 年 2 月 7 日に最終ラウンドの評価フェーズが終了し、最終選考方式として Ascon を選定したことが発表された。その後、NIST はステータスレポート NISTIR 8454 [25] を発行し、最終ラウンドにおける評価基準や選定プロセスを明確にした。なお、Ascon の概要と実装性能評価については、それぞれ 4.5 節と 3.3 節でまとめている。

以下、NISTIR 8454 [25] に基づき、最終ラウンドにおける評価基準と選定プロセスについてまとめる。

■評価基準 以下の 4 項目が主な評価基準となっている。

1. 暗号学的安全性
2. 制約のある環境下におけるソフトウェア及びハードウェアでの実装性能
3. サイドチャネル攻撃や故障利用攻撃への耐性
4. 知的財産

最も重要な評価基準は、暗号学的安全性である。設計者自身の安全性評価、安全性主張、安全性証明、公開されている第三者による安全性評価などの情報を幅広く評価している。また、明示的に要求されていないが、第 2 ラウンドと同様、nonce-misuse 安全性、RUP 安全性、内部状態復元の影響、耐量子安全性などが追加の考慮事項として挙げられている。なお、ファイナリストに対する安全性評価については、NISTIR 8454 [25] の第 3 章で整理されている。

もう 1 つの重要な評価基準は、制約のある環境下におけるソフトウェア及びハードウェアでの実装性能である。様々な性能やコストに関する測定基準において、ファイナリスト同士や現在の NIST 標準である AES-GCM [9]（認証暗号としての比較対象）と SHA-2 [18]（ハッシュ関数としての比較対象）との比較、評価が行われた。広く採用されている AES-GCM や SHA-2 に対し、大幅に優れた性能を発揮することが期待されている。なお、ファイナリストの性能比較結果については、NISTIR 8454 [25] の第 4 章と付録 B で整理されている。

その他、サイドチャネル攻撃への耐性を提供する必要はないが、容易かつ低コストで実現できることが要望されている（細部は、NISTIR 8454 [25] の第 4.3 節を参照されたい）。知的財産についてもまた、アルゴリズムの使用や実装における特許請求の必要性について反対しないものの、評価フェーズでの選定を妨げる可能性があると示されている（細部は、NISTIR 8454 [25] の第 2.2 節を参照されたい）。

■選定プロセス NIST LWC プロジェクトの初期段階において、NIST はターゲットアプリケーションに関するパブリックフィードバックに基づき、次の 2 つのプロファイルを指定した。

- プロファイル 1：制約のある環境でのソフトウェア及びハードウェア環境向けの認証暗号とハッシュ
- プロファイル 2：制約のある環境でのハードウェア環境向けの認証暗号

当初、両方のプロファイルをカバーするアルゴリズムの提出が求められたが、NIST は標準化のために複数のアルゴリズム（例えば、各プロファイルに対して 1 つのアルゴリズム）を選定することも検討した。

NISTIR 8454 [25] によると、ファイナリストに対する評価プロセスは、第三者による安全性評価、バリエーション、設計の微調整、ベンチマーク、耐量子安全性、知的財産に関する声明の 6 つの観点から検討されていることがわかる。なお、

表 2.4　NIST LWC 最終ラウンドにおける評価基準と選定プロセスの関係

番号	評価基準	選定プロセス
1	暗号学的安全性	第三者による安全性評価、耐量子安全性
2	制約のある環境下におけるソフトウェア及びハードウェアでの実装性能	ベンチマーク
3	サイドチャネル攻撃や故障利用攻撃への耐性	ベンチマーク
4	知的財産	知的財産に関する声明
5	その他	バリエーション、設計の微調整

表 2.4 は NIST LWC 最終ラウンドにおける評価基準と選定プロセスの対応関係をまとめたものである。

第三者による安全性評価　ファイナリストに対し、第三者による多くの安全性評価が行われた。その細部については、NISTIR 8454 [25] の第 3 章でまとめられている。公開されている安全性評価は、いずれも単一鍵または nonce-respecting 設定での安全性主張を無効にするものではなく、ファイナリストの多くは十分に安全性マージンが確保されている状況である。

バリエーション　応募されたアルゴリズムの公正な比較を行うために、NIST は各設計チームに対して特定の入出力サイズを持つ認証暗号とハッシュのバリエーションを提出するように求めた。一方で、NIST は、異なる入出力サイズ、異なる種類のベースとなる構成要素など、最大 10 種類までの複数のバリエーションを提出することも許容した。このような要望に対し、いくつかの設計チーム（例えば、Ascon、SPARKLE、Xoodyak、など）は eXtendable Output Function (XOF) のバリエーションを提出した。XOF は正式なバリエーションとはみなされないものの、XOF 機能を提供できる柔軟性は選定プロセスにおいて設計の利点であるとみなされた。

設計の微調整　最終ラウンドの初期段階において、安全性や実装性能を向上させるための軽微な設計変更が許容されたが、Ascon、GIFT-COFB、ISAP、PHOTON-Beetle、そして SPARKLE については、設計上の微調整が実施されなかった。

ベンチマーク　ベンチマーク結果に関して、NIST 標準である AES-GCM [9] と SHA-2 [18] よりも大幅に優れた性能を発揮することが期待された。具体的には、ソフトウェアでのベンチマーク結果、ハードウェアでのベンチマーク結果、サイドチャネル攻撃や故障利用攻撃への耐性、そしてこれらの攻撃を軽減させるために必要な実装オーバーヘッドに関する評価が行われた。また、特定の用途に応じて最適な実装を行うために、実装者がコストと性能のトレードオフを考慮できる柔軟性もまた、軽量暗号の重要な要素の 1 つとされた。

耐量子安全性　暗号アルゴリズムの長期利用の観点から、量子計算機の脅威に対する安全性が考慮された。共通鍵暗号アルゴリズムの耐量子安全性として最も一般的な量子アルゴリズムは Grover のアルゴリズム [13] である。例えば、量子計算機ではない現在の計算機（古典計算機）では計算量 $\mathcal{O}(2^n)$ で秘密鍵の全数探索が実行可能であるのに対し、量子計算機では Grover のアルゴリズムを使用して計算量 $\mathcal{O}(2^{n/2})$ で秘密鍵の全数探索が実行可能となる。また、古典計算機では計算量 $\mathcal{O}(2^{n/2})$ でハッシュ関数の衝突探索が可能であるのに対し、量子計算機では Grover のアルゴリズムを使用して計算量 $\mathcal{O}(2^{n/3})$ でハッシュ関数の衝突探索が可能となる。この量子アルゴリズムに対して安全性を確保するために、より大きな鍵サイズ（又は、ハッシュ関数におけるより大きな出力サイズ）が必要となる。なお、ファイナリストのうち、Ascon、SPARKLE、TinyJAMBU が 128 ビットよりも長い秘密鍵をサポートしている。

知的財産に関する声明　各設計チームに対し、NIST は候補アルゴリズムの実装によって侵害の恐れがある知的財産を全て特定するよう要求していた。結果として、PHOTON-Beetle のみが既存の知的財産を有することとなった。一方で、この知的財産に関する事項は、選定プロセスの決定には影響を及ぼしていないと示されている。

2.2.2.4　Ascon に関する評価

最終ラウンドにおける評価基準と選定プロセスに従い、NIST は標準化対象のアルゴリズムとして Ascon を選定した。具体的には、以下の観点が特に評価されている。

機能　Ascon ファミリーには、認証暗号とハッシュに加えて、追加の XOF が含まれている。これにより、幅広いアプリケー

ションのニーズを満たすことができる。また、暗号学的置換ベースの設計であることから、追加機能を実装するための追加コストが少ないことが期待されている。

成熟度　認証暗号のバリエーションは、CAESAR コンペティションで最終的なポートフォリオの 1 つである軽量アプリケーション（リソースに制約のある環境）の第 1 候補として選定された実績がある。また、最終ラウンドで行われた設計の微調整において、バリエーションが追加されたものの、第 2 ラウンドのバリエーションに関しては設計の変更が行われなかった。これらの事実から、安全性や実装性能を向上させるために設計の微調整を行なった他ファイナリストよりも高い成熟度を満たしていると言える。

安全性　初期バージョンの仕様が公開されてから長い歴史があり、豊富な評価・分析が行われてきたことから、第三者による安全性評価が最も多いファイナリストである。他ファイナリストと比較すると、安全性評価が先行して行われているにも関わらず、依然として高い安全性を維持している。特に、認証暗号のバリエーションは、nonce-respecting 設定で高い安全性マージンを提供するとともに、nonce-misuse 設定でも高い完全性を保証する。さらに、認証暗号モードは、耐漏洩安全性のためにモードレベルの保護メカニズムも提供している。

実装性能　ソフトウェア及びハードウェアで非常に優れた性能を発揮するとともに、コストと性能の間の様々なトレードオフをサポートする実装の柔軟性を実証した。特に、リソースが限られている様々なソフトウェア及びハードウェア上で、現在の NIST 標準である AES-GCM と SHA-2 より性能が優れている。また、サイドチャネル攻撃等への対策を施すことで保護された実装は、保護されていない実装よりも追加コストが低いことも示された。

■制約事項　重要な制約事項の 1 つは、256 ビット鍵のオプションがないことである。これは、量子攻撃に対する 128 ビット安全性が必要な場合に問題となる可能性がある。現状、耐量子安全性として 256 ビット鍵が必要となる場合、AES-GCM を使用することが推奨されており、NIST は今後必要に応じてより高い耐量子安全性を満たすバリエーションの追加を検討する可能性があると示唆している。

NISTIR 8454 [25] の発行時点において、NIST は第二候補のアルゴリズムを選定する必要がないと判断している。当面の間、Ascon ファミリーが制約のある環境下で十分な安全性を提供でき、実装性能においてもターゲットデバイスやターゲットアプリケーションで許容されると予想しているためである。

2.2.3　他標準化団体における Ascon の検討状況

NIST LWC プロジェクトの最終選考方式として選定された Ascon に関し、NIST 以外の組織での標準化動向についてまとめる。調査対象の標準化団体は、Internet Engineering Task Force (IETF)、World Wide Web Consortium (W3C)、International Organization for Standardization/International Electrotechnical Commission (ISO/IEC)、International Telecommunication Union Telecommunication Standardization Sector (ITU-T)、Global Platform の 5 団体である。

2023 年 9 月現在、IETF を除く 4 団体において標準化が行われていない。IETF では、以下のとおり Ascon が取り上げられている。

- インターネットドラフト “Secure UAS Network RID and C2 Transport” [17] の第 5.3 節において、無人航空機で Ascon を選択することが最善である、と記載されている。その際、Encapsulating Security Payload (ESP) [15] や Datagram Transport Layer Security (DTLS) [22] の拡張が必要であると記載されている。
- インターネットドラフト “Properties of AEAD algorithms” [5] の第 4.4.2 節において、NIST LWC プロジェクトに関する参照が行われている。
- IETF 117 で開催された TLS ワーキンググループでの発表 “New Post-Quantum Signatures on the Horizon” [26] において、Ascon-Sign (SPHINCS$^+$ with Ascon) が取り上げられていた。

その他、産業界では、NIST LWC プロジェクトの結果を受け、Ascon を利用可能な環境を提供するような動向がある。例えば、IP コア関連では、Rambus 社の「Ascon-IP-41 暗号エンジン」 [21]、Xiphera 社の「XIP2201B: Ascon」 [27]、CAST 社の「Ascon-F」 [6] が提供されており、暗号ライブラリ関連では、Bouncy Castle 1.7.3 以降 [7]、CIRCL [8] などで提供されている。NIST による Ascon の最終的な標準化仕様が公開されることで、他標準化団体や産業界での活動が活性化されることが期待される。

2.3 軽量暗号はどこに使えるのか

「いつでも、どこでも、何でも、誰でも」ネットワークにつながるユビキタスネットワークの構築は、IoT というキーワードで表現されるようになっている。IoT というコンセプトの下、パソコンやスマートフォン、タブレットといった従来型の ICT 端末だけでなく、自動車、家電、ロボット、施設などがインターネットに繋がることとなる [30]。ただし、どのようなデバイスがネットワークに接続されるかについて明言することはできず、デバイス上でどのような処理が行われるかも不明である。IoT の時代だからこそ、今までは考えられない状況を考えなければならない。

これら IoT 端末は、先述したユビキタスネットワーク構築のために、我々の生活空間へシームレスに浸透しつつあるため、プライバシーや情報秘匿、また情報の完全性担保を目的としたセキュリティ機能が必須となる。加えて、サービス利用者の目的はセキュリティ機能によるメリットを享受することではないため、IoT 端末を利用した円滑なサービス提供をセキュリティ機能が妨げるべきではない。また、全てのデバイスに高機能の CPU が搭載されるとは考えにくい。従来型 ICT 端末に比べて処理能力、回路規模、消費電力、そしてメモリサイズなどの制約を含め、計算機資源が乏しいデバイスも想定しなければならない。例えば、IoT 端末の 1 つである自動車において、図 2.1 のような要求条件が述べられている。

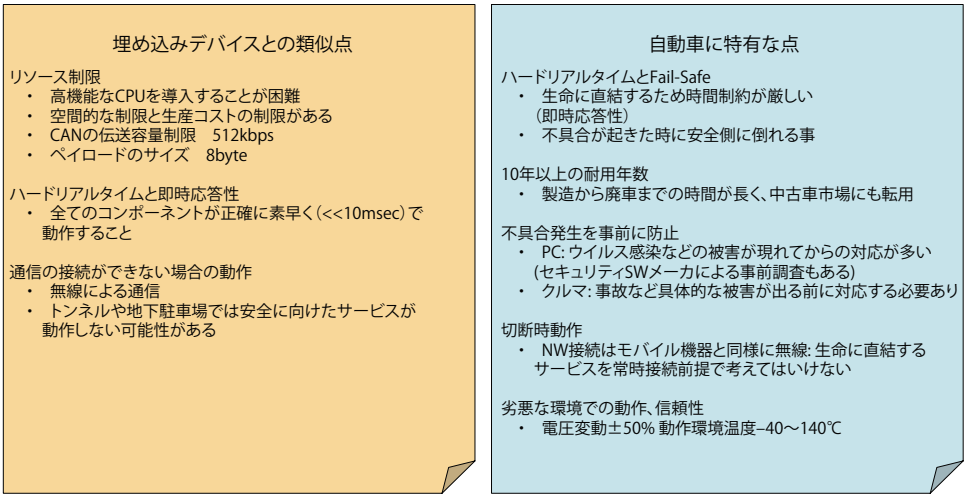

図 2.1 自動車の機能に関する要求条件

そこで、有望となるのが軽量暗号である。軽量暗号には、CPU 負荷が軽い、使用するメモリサイズが小さい、低レイテンシ性などの特徴がある。このような軽量暗号は、計算リソースが比較的乏しい IoT 端末に適していると考えられており、特徴に応じて、例えば次のような場面での使用が期待されている。

- CPU やメモリなどを多くのアプリケーションで共有しなければならない場合、CPU コストが小さく、メモリ使用量が少ない暗号として軽量暗号の使用が期待される。例えば、スマートフォンやタブレット端末、スマートテレビのような高機能化されたデバイスなどがこの場合に該当する。
- 装置そのものがバッテリーで動作している場合、消費電力が少ない暗号として軽量暗号の使用が期待される。例えば、電気が通っていない場所に置かれることが多い環境測定用のデバイスなどがこの場合に該当する。また、埋め込み型の医療機器に関して、人体に埋め込まれるデバイスであればバッテリー駆動以外は考えられない。埋め込み機器として可能な限り小さいデバイス、かつ人体への影響が少ないデバイスであることが望まれることから、このような場合において軽量暗号の使用が期待される。
- 即時性が求められる場合、低レイテンシ性を持つ暗号として軽量暗号の使用が期待される。例えば、バッテリーの消費を極力抑えるために、データを送信する場合にのみ電源スイッチが ON になり、一瞬だけ動作してデータを伝送し、伝送が終了次第電源スイッチが OFF になるように運用している機器などがこの場合に該当する。また、自動車に関しても、制御するためにかかる時間が遅い場合に安全性に影響が出る可能性が高いため、この場合に該当する。

これら軽量暗号の活用例について、スマートテレビ、RFID、農地での環境測定、医療機器、工場の機器制御、自動車を例

にして紹介する。

2.3.1 家電・スマートテレビ

スマートフォンやタブレット端末、スマートテレビのようなデバイスに搭載されている CPU には様々な種類があり、低機能なものから高機能なものまである。高機能の CPU を使用している場合は問題ないが、低機能の CPU を使用している場合は当然、実現できる機能が制限される。

例えば、テレビ内の処理は、スクランブルの解除、圧縮の復号、映像・音声の提示のみがハードウェア処理であり、それ以外の処理は内蔵されている CPU で処理される。また、CPU は実に多くの作業をしており、ほとんど負荷 100% で使用されている。さらに、今後の発展としてネットワークと接続して通信アプリケーションをテレビ上で動作させようとしている。テレビを少しでも安価にするためには、搭載する CPU も安価なものが好ましく、このため高機能な CPU が使われていないのが現状である。そのようなテレビにおいて、テレビ機能だけでなく、アプリケーションをシームレスで動作させる場合には、CPU やメモリリソースの取り合いが生じる。その上、暗号化すべきデータが存在するならば、暗号化は使用メモリ（ROM/RAM）サイズが小さく、CPU 負荷が小さいものが推奨される。テレビ内で暗号用途の特別なチップを導入することはなく、ソフトウェアで暗号化処理を行うこととなるため、ソフトウェアで CPU 使用時間が短く ROM の使用サイズが小さい暗号の利用価値が高いと考えられる。

IoT 時代では、テレビ以外にも多くの家電がネットワークに接続される。ネットワークを通じてやりとりするデータの中には秘匿しなければならないデータもあり得る。仮に、エアコン、ガスコンロなどがネットワークに接続された場合、その制御信号がネットワーク上を流れるサービスでは、制御信号が外部から不正アクセスされて改ざんされたり、任意のコマンドが入力されることで異常動作しないようにしなければならない。また、家庭内のロボットが個人データを持つ場合、プライバシー保護の観点からデータを秘匿しなければならない。一方、これらのデバイスは 10 年以上使用し続けることも考えなければならない。その上、小さな・安価なデバイスにおいては特殊なハードウェアが搭載されることは考えにくい。デバイスによっては、スマートテレビと同じようにリソースの取り合いが生じることもあり得る。これらのデバイスでは、アップデートが可能なソフトウェア処理がメインとなり、それを司る CPU は低機能で安価なものになるであろう。このような CPU でデータ保護を実現するためには、ソフトウェアで CPU 使用時間が短いタイプの暗号が必要となる。

2.3.2 RFID タグ利用のアプリケーション（物流管理等）

RFID とは、無線を利用して物を認識するシステムのことであり、様々な用途で使用されている。例えば、倉庫内の在庫管理、物流における物品管理、CD/DVD ショップでの盗難防止、物品の履歴管理、電子マネー、交通用カード、社員証カードなどである。動物の生態を調べるための追跡用にも使われることがある。同様に、人の居場所を追跡するために利用することなども考えられている。さらには、IoT 時代の家庭内の物品識別、例えば、冷蔵庫内に入っているものを冷蔵庫が認識するために利用することも考えられている。

RFID による無線電波は強くはないため、近距離伝送で利用されることが多い。したがって、例えば、倉庫内にどのような物品が保管されているかを別の同業者が知るために、RFID で発信されるデータを倉庫の外部から盗聴するようなことは困難である。倉庫内での在庫管理においては、暗号を実装する必要性は少ないと考えられる。

これに対し、CD ショップでの万引き防止、電子マネー、交通用カード、社員証カードの偽造防止、人の追跡や人の持ち物の追跡におけるプライバシー保護、動物の追跡における密漁防止等の目的で使用される場合は、データは秘匿されることが必須である。

さて、RFID タグは RFID で用いられるチップのことであり、そのサイズは数十 μm 角から、数 cm 角のものまである。このチップの中にプロセッサ、メモリなどが凝縮されている。図 2.2 に RFID の構造を示す。
送信部と書かれている部分はアンテナであり、使用されている周波数（kHz〜MHz）に応じてその大きさは様々である。送信部以外は小型化が可能であるが、データ記憶容量や使用されるプロセスによって大きさが決まる。

RFID では、全体の回路規模に対して暗号機能に使える回路サイズには限界がある。RFID がパッシブタグ、すなわち電池を内蔵していない場合、電波から給電するため消費電力に大きな制約が存在する。このような制約の下でデータを保護するため、ハードウェアが小さく、消費電力の少ない暗号が求められる。

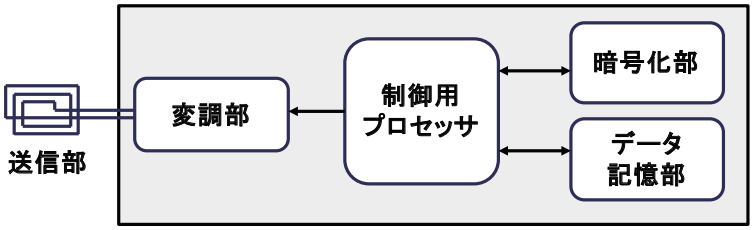

図 2.2　RFID の構造

2.3.3　センサーを利用したスマート農業

農作物を栽培する際に、気候の変化に応じた対応を講じることで、収穫量を安定させるなど生産性を向上させたり、作物の品質を向上させることが可能になると言われている。これまで多くの農家では、このような対応を熟練者の経験や勘で補ってきた。例えば、気候（気温、湿度、日照時間、日照量、土壌水分、風向、風速、降水量など）をモニタリングし、これまで経験・勘に頼っていた部分を数値化・データ化することができれば、熟練者のみならず経験の浅い方々でも安定した農業を営むことができる。具体的には、これらのモニターにより、水まき時間、量を制御したり、ビニールハウスの窓の開け閉めを自動化するなどが可能となる。熟練者にとってもこのようなモニタリングは、気候への対応を容易にし、作業計画・スケジューリング、病害虫駆除対策等が容易になるだけでなく、農地そのものを評価し・改良するための指針を与えることになり、安定した収穫への手助けとなる。

モニタリングするデータは細かければ細かいほど分析精度が高くなる。特に、農地の評価を行う上では、広大な農地を区画に分けて区画ごとのデータを取ることが望ましい。農家の方々は、少ない労力で長期的に、安価に利用できるモニタリング装置を要望することとなる。このため、徐々に環境センサーネットワークの利用が始まっている。

このセンサーへの条件として、自律駆動、小型、省電力、大量設置、などがある。また、細かいデータ取得には、膨大な数のセンサーが必要となるため、これらの条件を満たすためにも軽量暗号の使用が望まれる。

同じような状況が防災上も考えられる。天候変化だけでなく、地殻の変動をセンシングすることで地震予知に役立てたり、火山噴火予知、土砂崩れなどの予測に役立てることが考えられている。これらの用途のセンサーも、人の出入りが困難であり、給電が困難な場所に置かれることがしばしばある。また、防災関連のデータは重要であり、盗聴されることは許容できるが、改ざんなどがあってはならない。データが改ざんされることで、不要な警報が発せられることがあってはならないからである。このような秘匿性のあるデータの改ざん防止、保護のためには、認証付きの軽量暗号が有望と考えられる。

2.3.4　医療

入院経験のある方は少なくないと思うが、入院患者には心電図・血圧・脈拍・血糖値・酸素濃度などを測定する様々なセンサーが取り付けられる。基本的にこのようなセンサーは有線で接続されており、電源に接続されている。ただし、患者が移動する場合、このようなセンサーも患者とともに移動できることが望ましい。特に、帰宅等の外出が可能な方、在宅・職場であっても種々のデータをモニタリングする必要のある方もいる。さらに、体内に測定装置を埋め込まなければならない場合も考えられる。定期的なデータ測定、投薬時間の連絡などのためである。特に、デバイスを埋め込む場合、何度もデバイスを取り出すことは考えにくい。このため、無線利用のデータ通信、バッテリーによる長時間動作が可能なデバイスでなければならない。埋め込み型ではなくとも、有線では移動の自由度が制限されるため無線であることが望ましく、必然的にバッテリーで長時間動作するセンサーであることが必須となる。

これらのセンサーデータは個人情報そのものであり、プライバシー保護の観点から全てのデータが秘匿されなければならない。現在、特に埋め込み型センサーでは小型化の研究・開発が進められており、nm サイズのデバイスが開発されているところである。当然、それに付随する暗号化処理部分も小型化が必須である。

最近ではウェアラブルデバイスの進展に伴い、mHealth と呼ばれる概念が出てきている。Mobile Health の略とも言われるが、確立した定義は存在していない。例えば、mHealth として次のようなことが考えられている。ウェアラブルデバイスを生体データの観測装置として利用し、日々の健康維持のために心肺活動などのデータを観測する。家庭内で日々のデー

タを記録し、健康診断や通院の際に利用するものである。また、日常生活の中のデータ保存のみならず、健康維持のために行っている活動の記録としてもしばしば利用する。従来からの健康機器である歩数計も GPS による位置情報の取得など、個人情報を取り扱うものが出てきている。

この mHealth では、人の動作により発電する発電デバイスを組み込んだデバイスもあるが、基本的にはウェアラブルデバイスが測定装置となるため、バッテリー運用となる。使用方法にも依存するが、常時データを測定し、観測、集計、分析するのであれば、無線による伝送が必要となる。これらのデータも個人情報そのものであり、プライバシー保護の観点から全てのデータが秘匿されなければならない。

2.3.5 産業用システム

工場などでは資材の運搬や加工、組み立てなどの工程を自動化し、効率的に運用することが考えられている。例えば、工作機械やロボットを動作させるにあたって、ネットワークを通じて情報を共有したり、センサーを用いた工程管理を行うことが可能となる。さらに、ネットワークを通じて全ての情報を一か所に集めることで、集中管理が可能となる。

もっとも先進的にこのような自動化に取り組んでいたのが、ドイツであると思われる。国家プロジェクトとして多額の予算を組み、インダストリー 4.0 として種々の機械・装置がセンサーを持ち、そのデータに応じて考えながら動作するスマートファクトリ（考える工場）という概念を実現しようとしてきた。

データを集中管理するにあたり、ネットワークについても EtherCAT と呼ばれる方式が提案された。工場のデータ管理では、個々のデータが重要な意味を持っている。従来のインターネットプロトコルである TCP/IP ではデータのネットワーク上での衝突等が起こり、データの遅延、消失などが生じてしまうため、工場データの管理に利用するには弊害があった。そこで、EtherCAT では、個々の機器をスレーブ化し、その接続をシリアルにする方式としている。

これらのセンサーは工場内のあらゆる場所に配置される。当然、人の手が入りにくい場所もある。電源が装置に備え付けてある場合が多いものの、全てをシリアルに接続することが困難な状況も生じる。これに対応するためには、無線でデータを送ることも考えられている。無線でデータを送る場合は、工場内で送受信できるようにするため、ある程度の距離を無線伝送することになる。このためには、暗号化してデータを送受信することが必要である。

2.3.6 自動車

自動車では、車内でのデータ通信だけでなく、車外との通信も行うようになってきた。この自動車の通信においては、自動車と自動車（車車間）ならびに自動車と信号機や道路標識などのインフラ（路車間）との相互通信により安全運転の支援を行う自動運転支援システム（Car2X communication）に対応するために大量の情報を処理し、かつクラウドと連携してコンテンツなどのサービスと連携する車載情報システム、ならびにレーダなど外部からの情報を各種センサーで取得しボディ系、シャシー系など複数の電子制御ユニット（ECU）間において互いに情報をやりとりしながら協調制御をおこなう車載制御などがある。

車内の ECU 間の協調制御をおこなうため、車内は CAN (Controller Area Network)、LIN (Local Interconnect Network)、Ethernet など各種方式によるネットワークが張り巡らされている。CAN は車載において基幹となるネットワークであり、パワートレイン系、シャシー系、ボディ系などとの協調制御に広く使用されている。例えば、車間をミリ波レーダで計測し、車間が狭くなると警告表示あるいは警告ブザーを開始、ブレーキ制御、シートベルト制御といった衝突検知システムを CAN を通じて情報共有を行うことで実現している。車載 LAN においても攻撃の実証例があり、CAN に誤ったメッセージが流れるとブレーキ制御が誤るなどの重大な事態につながりかねないため、メッセージを暗号化するとともに改ざんされていないか認証を行う必要がある。車車間、路車間のように非常に高い暗号処理性能、低レイテンシ暗号までは要求されないが、リアルタイム性は求められており、高速な暗号処理の実現が必須である。

車の自動運転支援システムにおいては車車間、路車間と相互通信しなければいけない機器の数が非常に多く、かつ個人につながる情報の漏洩を防ぐため暗号処理にも対応するため、回路規模が小さく、低レイテンシな暗号を実現する必要がある。

車載情報端末においては、ほかにもクラウドを経由した様々なサービス、渋滞予測などの交通情報あるいはコンテンツなども取り扱っており、情報の保護あるいは改ざんの防止が求められる。特に、コンテンツ保護においては高スループットの暗号処理が求められる。

規格関連の状況では、欧州 AUTOSAR では車内通信用にメッセージ認証技術の必要性が述べられている。すなわち、多くの自動車で採用されている CAN に、AUTOSAR 規格の R4.2.2 における Secure Onboard Communication (SecOC) としてカウンタとメッセージ認証コード (MAC) によるメッセージ認証が明記されている [1]。車外通信では、欧州の Car 2 Car Communication Consortium (C2C-CC) による車車間通信へ軽量暗号の活用が期待できる [2]。

2.4 どんな軽量暗号、パラメータを選べばいいか

2.4.1 一般的方針

2.1 節で挙げたように、軽量暗号は既存暗号と比べて実装時に回路規模、消費電力量、レイテンシ、メモリサイズのいずれか、もしくは複数の性能指標において優位性を持つ暗号である。逆にいえば、ある軽量暗号について既存暗号と比べて回路規模については小さく実装可能であるものの、その回路規模が小さな実装では既存暗号の実装より消費電力量は増えている可能性もあり得る。このように軽量暗号は万能のものではない。暗号利用システムにおいて暗号に対する性能指標の要求条件を明らかにすることは困難であることが多いが、やみくもに軽量暗号を利用するのではなく、まずは要求条件をある程度は明らかにしなければならない。その上で、従来暗号、特に CRYPTREC 暗号リストに掲載されている暗号の利用を検討し、要求される性能指標に対して達成困難な場合には軽量暗号の利用を検討するのが望ましい。

軽量暗号を使用するにあたり、秘匿のみが必要な場合、データ認証のみが必要な場合、両方が必要な場合、といった場合が考えられ、目的に応じて方式を選択する必要がある。例えば、ブロック暗号を使用する場合、使用する利用モードにブロック暗号の復号が必要ないのであればその分の実装コストを削減できる。秘密鍵をハードコードするような利用形態であれば、その分の実装コストも削減可能である。利用目的と実装からの制約に照らし合わせ、適した方式を選択することが求められる。

2.4.2 鍵長の選択

鍵長は安全性の基準となる最も重要なパラメータであり、慎重な選択が求められる。ブロック暗号においては一組、あるいは少数の入出力があれば全数探索が可能であり、次節において述べる多くのユースケースにおいても全数探索のシナリオが成立する。例えば、鍵長を 128 ビットから 80 ビットに減らしたとする。回路集積効率に関する 3 年で 4 ($= 2^2$) 倍というムーアの法則が今後も続くとすると、$(128 - 80)/2 \times 3 = 72$ 年寿命が短くなることに注意しなければならない。

2.4.3 ブロック長の選択

ブロック暗号におけるブロック長も安全性に直結する重要なパラメータである。特に、ブロック暗号利用モードや認証暗号にブロック長の短いブロック暗号を使用した場合、安全性が保たれるデータ量に厳しい制限が加わり、このための対策が必要となる。

例として CTR モードの安全性の評価法を紹介する。CTR モードの安全性は、利用するブロック暗号のブロック長を n ビット、同じ鍵のもとブロック暗号が呼ばれる回数を σ とおくと、一様ランダムなビット列との識別が確率 $\sigma^2/2^{n+1}$ 以下であることが示されている（例えば、CRYPTREC 技術報告書 No.2012 (2011/3/4 更新版) 47 頁 [20])。この確率に基づき、システムの利用者のうち一人程度は暗号文とランダムな文字列との識別が可能となるリスクまで受容出来る場合の同一の鍵で処理できる最大データ長を求めると表 2.5 の通りとなる。それほど大きなデータではないことに注意が必要である。

別の注意点として、ブロック暗号を構成要素として用いてハッシュ関数を構成する方法が知られている。これらは十分に長いブロック長を有するブロック暗号を用いた場合に安全性が保たれるものであり、ブロック長の短い軽量ブロック暗号はこのような用途には適さないと考えられる。

2.4.4 処理データ量と鍵更新、その他の対策

利用形態に応じて、例えば秘密鍵を更新可能な環境であれば、頻繁に更新するといった対策が考えられる。あるいは、秘密鍵をハードウェアとしてハードコードするような実装においては秘密鍵を更新できないため、処理するデータ量に制限を

表 2.5　利用者一人の暗号文がランダムな文字列と識別可能となる最大データ長

ブロック長 n (ビット)	利用者数	データ長 $n\sigma$
64	10^3	1.4 Gbyte
	10^6	46.3 Mbyte
	10^9	1.4 Mbyte
48	10^3	4.3 Mbyte
	10^6	139.0 Kbyte
	10^9	4.4 Kbyte

設け、それを超過する前にデバイス自体を破棄するといった運用が考えられる。

鍵更新のタイミングについて考えると、一般的にデータを処理すればするほど暗号方式の安全性は徐々に低下する。このため、任意の攻撃者による攻撃成功確率が十分に許容できるほど小さい範囲にある間に鍵を更新することが望まれる。例えば、[10] にある CMAC では、一般的なアプリケーションにおいて、ブロック長 128 ビットの AES を利用した場合は 2^{48} ブロックのデータ（2^{22}G バイト）を処理する前の鍵更新を、あるいは 64 ビットブロック暗号の TDEA を利用した場合は 2^{21} ブロックのデータ（16M バイト）を処理する前の鍵更新を推奨している。これらの制限により、攻撃者の攻撃成功確率は AES の場合は 10 億分の 1、TDEA の場合は 100 万分の 1 以下となることが期待される。許容できる攻撃成功確率は暗号方式を使用するアプリケーションに依存し、慎重な選択が求められる。

鍵更新の方法についても、アプリケーションに応じてそれぞれに適した方法を選択することが必要である。鍵共有プロトコルを実行できる環境であれば、鍵更新は問題とはならない。あるいは、マスター鍵からセッション鍵を生成し、鍵更新をしながら同期をとるような利用形態が考えられる。

いずれにせよあるタイミングで鍵を更新、あるいは破棄する必要がある。この頻度を遅らせることができるような暗号方式も提案されており、例えば MAC であれば SUM-ECBC [28] や PMAC_Plus [29]、暗号化であれば CENC [14] といった例が挙げられる。これらの方式では、利用するブロック暗号のブロック長が n ビット、ブロック暗号が呼ばれる回数が σ であれば、攻撃者の攻撃成功確率はおおよそ $\sigma^3/2^{2n}$ 以下となる。先述の方式と比べ、64 ビットブロック暗号であれば敵の攻撃成功確率は $\sigma/2^{64}$ 倍小さくなり、攻撃成功確率の閾値に達するまでにより多くのデータを処理することができるようになる。

2.4.5　利用シナリオ

軽量ブロック暗号は一般的に線形攻撃や差分攻撃などの暗号学的な攻撃に対して、十分な耐性を有するように設計されている。これらは軽量ブロック暗号に限らず、通常のブロック暗号についても考慮される安全性である。一方で、通常のブロック暗号では関連鍵攻撃や既知鍵攻撃、選択鍵攻撃といった攻撃者側にとりわけ有利な状況を考え、その安全性を評価することが行われている。実装効率の観点から、軽量ブロック暗号では簡素な設計を採用する方式が多くあり、必ずしもこれらの攻撃に対する安全性が十分ではない、あるいは十分な評価が実施されていない、といったケースが考えられる。関連鍵攻撃や選択鍵攻撃に対する耐性が十分ではないことが分かっている方式も存在し、これらの方式を採用する場合には攻撃シナリオが成立しないような運用が求められる。

2.4.6　その他の留意点

ソフトウェア実装に適した方式やハードウェアに特化した方式など様々な選択肢があり、実装環境に応じて使用する方式を選択する。このとき、暗号学的な攻撃手法のみならず、サイドチャネル攻撃に対する対策の必要性について検討することが重要である。一般的に多くのユースケースにおいてサイドチャネル攻撃が可能な環境が考えられ、実装レベルでの対策の必要性を検討する必要がある。

2.4.7 CRYPTREC 暗号リストの暗号との違い

ブロック暗号を例にして考える。CRYPTREC 暗号リストに掲載されているブロック暗号はブロック長は 64 ビットもしくは 128 ビット、鍵長は 128 ビット以上となっている。

一方、軽量ブロック暗号については、ブロック長は 32 ビット、また鍵長も 80 ビットなど CRYPTREC 暗号リスト掲載の方式より短いものが数多く提案されている。ブロック長と鍵長は安全性に直結するパラメータであり、ブロック長や鍵長を短くすることにより健在化するリスクが利用するシステムにおいて受容可能かどうかを判断しなければならない。

CRYPTREC 暗号リストにある注釈無しの暗号の利用で安全に利用できる範囲であっても小さなパラメータの軽量暗号を採用する場合には上述の例のように利用データ量などについて、再評価する必要がある。

軽量暗号に限らず、CRYPTREC 暗号リストの暗号と同様、無条件に永遠に安全である効率的な暗号はない。利用目的とリスク管理を適切に行ない、従来暗号の利用が困難であるが軽量暗号が使える場面では積極的な利用を推奨する。

2.5 軽量暗号活用例と効果

2.3 節で取り上げた活用例において、軽量暗号、その中でもブロック暗号、あるいはメッセージの改ざんも検出できるような認証暗号を利用する場合において、どのような点に着目して軽量暗号を選定していけばよいかについて本節で例示する。以下、選び方の一例として本ガイドラインの 3 章に記載している軽量暗号の性能比較を元に記載する。軽量暗号を適用する際には 2.4 節に記載されている鍵長の選択などの安全性にも配慮することに留意されたい。

2.5.1 家電・スマートテレビ

スマートテレビのように計算リソースの取り合いが生じる場合、3 章の図 3.40 に基づき、例えばソフトウェアで CPU 使用時間が短く ROM の使用サイズが小さい SPECK、SIMON、Piccolo、TWINE のような軽量暗号を選定することを検討できる。

また、家電におけるデータ保護については、アップデートが可能なソフトウェア処理がメインとなり、それを司る CPU は低機能で安価なものになるであろう。このような CPU でデータ保護を実現するものの一例として、3 章の図 3.34 に基づき SPECK のようなソフトウェアで CPU 使用時間が短いタイプの軽量暗号を選定することも考えられる。

2.5.2 RFID タグ利用のアプリケーション（物流管理等）

RFID では、全体の回路規模に対して暗号機能に使える回路サイズの限界もさることながら、消費電力に大きな制約が存在する。消費電力の低減が可能な暗号方式として、例えば 3 章の図 3.18 から SIMON、SPECK、Piccoro、PRINCE などの軽量暗号を選定することが考えられる。特に 180nm 以上のレガシーのプロセスでは、この差がクリティカルである。40nm 世代のプロセスであっても、50μm 角クラスの極めて小さなチップであれば、この差が搭載可否に影響を与える。

2.5.3 センサーを利用したスマート農業

農作物の生産向上のためには、細かいデータ取得が必要であり、膨大な数のセンサーが必要となる。これらのデータのすべてを暗号化するためには、安価な軽量暗号が望まれる。

また、防災上では、データを暗号化し、メッセージ認証コード（MAC）を付与する必要がある場合も考えられる。認証暗号を実装するにあたり、3 章の図 3.45 によると、ROM サイズの使用量が少なく小型実装に向いている JAMBU-SIMON、SILC-PRESENT、ACORN、Ascon、Minalpher などの軽量暗号の活用が例として考えられる。ブロック暗号の実装を考えるのであれば、例えば 3 章の図 3.42 から AES と比較して 1 回あたりの処理 cycle 数が少ないことから消費電力量が小さく、バッテリー寿命を長くすることができる SPECK、SIMON、PRESENT、TWINE、Midori などの軽量暗号の使用も検討に値する。メッセージが短く秘匿を要しない場合は、4.4 節に記載のメッセージ認証コードの利用、あるいはブロック暗号に軽量暗号を適用し CMAC モードを使用することで小型化が図られる。

2.5.4 医療

医療用センサーデータは個人の情報そのものであり、プライバシー保護の観点からは全てのデータが秘匿されなければならない。現在、特に埋め込み型センサーでは小型化の研究・開発が進められており、nm サイズのデバイスが開発中である。当然、それに付随する部分も小型化が必須である。これらを併せ持つ暗号としては、ハードウェアで省電力の軽量暗号、例えば 3 章の図 3.15 を参考にすると、SIMON、SPECK、Piccolo、PRESENT などが候補となりうる。

また、mHealth の場合は、健常者のウェアラブル端末を利用することが主な想定となっているため、小型化、長時間化ということはあまり大きな問題とはならないが、ウェアラブル端末の CPU の小型化・低廉化のために、軽量暗号が望ましい。

2.5.5 産業用システム

工場などの産業用オートメーションにおいて、フィールドネットワークのオープン化が進む中、EtherCAT などの超高速の産業用オープンネットワークが注目されている。例えば、このネットワークにつながるノードの 1000 点デジタル I/O の読み書きに求められる速度は 30μs であり、1 つのイーサネットフレームでは 1486 バイトまでのプロセスデータを交換できる。この通信路の秘匿と改ざん防止を AES で実装しようとした場合、MAC 検証、復号、（解釈、書き換え）暗号化、MAC 生成で 1 ブロックあたり 4 回の暗号化回路を call する必要がある。AES 1 ブロック暗号化に 100ns、処理速度にして 1.3Gbps かかるとすると 37.2μs 必要になり、AES では厳しい条件となる。これに対して、3 章の図 3.8 によると、例えば軽量暗号の Midori あるいは PRINCE を Unrolled 実装することで回路規模を AES より抑えた上でそれぞれ 3.9Gbps、3.6Gbps の処理速度で演算が出来るため、リアルタイム性が求められる用途での活用が期待される。

2.5.6 自動車

自動車内部にハードウェア実装するための暗号方式を選定する例として、3 章の図 3.15 と図 3.16 を参考に、AES と比較し回路規模を抑えた上で処理速度が高速である Midori、PRINCE、PRESENT、SIMON などの実装が一例として候補としてあげることができる。

車の自動運転支援システムにおける暗号方式としては、レイテンシを低くするため、1 回分のラウンド処理ではなく複数のラウンド処理をハードウェアに実装することになる。このため、例えば 3 章の図 3.7 と図 3.8 を参考に、AES と比較し回路規模を抑えた上で処理速度が高速である Midori、PRINCE、PRESENT、SIMON などの利用も考えられる。

車載情報端末における情報を保護するため、特にコンテンツ保護においては高スループットの暗号処理が求められることから、例えば 3 章の図 3.16 をもとに軽量暗号である Midori の実装も考えられる。

参考文献

[1] AUTOSAR Specification of Secure Onboard Communication, https://www.autosar.org/fileadmin/standards/R4-3/CP/AUTOSAR_SWS_SecureOnboardCommunication.pdf（2023-10-07 閲覧）

[2] CAR 2 CAR Communication Consortium, https://www.car-2-car.org/index.php?id=5（2023-10-07 閲覧）

[3] Bassham, L., Chang, D., Kang, J., Kelsey, J., McKay, K., Turan, M.S., Waller, N.: NIST Lightweight Cryptography Project, https://csrc.nist.gov/projects/lightweight-cryptography

[4] Bernstein, D.J.: CAESAR: Competition for Authenticated Encryption: Security, Applicability, and Robustness, https://competitions.cr.yp.to/caesar.html（2023-10-04 閲覧）

[5] Bozhko, A.: Properties of AEAD algorithms. Internet-Draft draft-irtf-cfrg-aead-properties-01, Internet Engineering Task Force (Mar 2023), https://datatracker.ietf.org/doc/draft-irtf-cfrg-aead-properties/01/, work in Progress

[6] CAST: Ascon-F, Ascon Authenticated Encryption & Hashing Engine, https://www.cast-inc.com/security/encryption-primitives/ascon-f（2023-10-07 閲覧）

[7] Castle, B.: The Legion of the Bouncy Castle, https://www.bouncycastle.org/releasenotes.html#r1rv73（2023-10-07 閲覧）

[8] Cloudflare: CIRCL (Cloudflare Interoperable, Reusable Cryptographic Library), https://github.com/cloudflare/circl（2023-10-07 閲覧）

[9] Dworkin, M.: NIST SP800-38D – Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC (August 2015), https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38d.pdf

[10] Dworkin, M.: Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication (May 2015), https://nvlpubs.nist.gov/nistpubs/specialpublications/nist.sp.800-38b.pdf

[11] GMO サイバーセキュリティ by イエラエ株式会社: 軽量暗号の評価指標、標準化動向に関する調査（NIST 軽量暗号コンペティションファイナリストなど）（文書番号：CRYPTREC EX-3206-2022）(2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3206-2022.pdf

[12] GMO サイバーセキュリティ by イエラエ株式会社: 軽量暗号 Ascon などに関わる標準化動向調査（文書番号：CRYPTREC EX-3302-2023）(2023)

[13] Grover, L.K.: A fast quantum mechanical algorithm for database search. In: Miller, G.L. (ed.) Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing, Philadelphia, Pennsylvania, USA, May 22-24, 1996. pp. 212–219. ACM (1996), https://doi.org/10.1145/237814.237866

[14] Iwata, T.: New Blockcipher Modes of Operation with Beyond the Birthday Bound Security. In: Robshaw, M.J.B. (ed.) Fast Software Encryption, 13th International Workshop, FSE 2006, Graz, Austria, March 15-17, 2006, Revised Selected Papers. Lecture Notes in Computer Science, vol. 4047, pp. 310–327. Springer (2006), https://doi.org/10.1007/11799313_20

[15] Jokela, P., Moskowitz, R.G., Melén, J.: Using the Encapsulating Security Payload (ESP) Transport Format with the Host Identity Protocol (HIP). RFC 7402, 1–40 (2015), https://doi.org/10.17487/RFC7402

[16] McKay, K.A., Bassham, L., Turan, M.S., Mouha, N.: NISTIR 8114: Report on Lightweight Cryptography, https://nvlpubs.nist.gov/nistpubs/ir/2017/NIST.IR.8114.pdf

[17] Moskowitz, R., Card, S.W., Wiethuechter, A., Gurtov, A.: Secure UAS Network RID and C2 Transport. Internet-Draft draft-moskowitz-drip-secure-nrid-c2-13, Internet Engineering Task Force (Sep 2023), https://datatracker.ietf.org/doc/draft-moskowitz-drip-secure-nrid-c2/13/, work in Progress

[18] National Institute of Standards and Technology: FIPS 180-4 – Secure Hash Standard (SHS) (August 2015), https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.180-4.pdf

[19] NIST: Submission Requirements and Evaluation Criteria for the Lightweight Cryptography Standardization Process, https://csrc.nist.gov/CSRC/media/Projects/Lightweight-Cryptography/documents/final-lwc-submission-requirements-august2018.pdf

[20] Phillip Rogaway: Evaluation of some Blockcipher Modes of Operation（文書番号：CRYPTREC EX-2012-2010R1）(2010), https://www.cryptrec.go.jp/exreport/cryptrec-ex-2012-2010r1.pdf

[21] Rambus: Rambus IP Solution Supports New NIST Lightweight Cryptography Algorithm, https://www.rambus.com/blogs/rambus-ip-solution-supports-new-nist-lightweight-cryptography-algorithm/（2023-10-07 閲覧）

[22] Rescorla, E., Tschofenig, H., Modadugu, N.: The Datagram Transport Layer Security (DTLS) Protocol Version 1.3. RFC 9147, 1–61 (2022), https://doi.org/10.17487/RFC9147

[23] Turan, M.S., McKay, K.A., Çalik, Ç., Chang, D., Bassham, L.: NISTIR 8268: Status Report on the First Round of the NIST Lightweight Cryptography Standardization Process, https://nvlpubs.nist.gov/nistpubs/ir/2019/NIST.IR.8268.pdf

[24] Turan, M.S., McKay, K.A., Çalik, Ç., Chang, D., Bassham, L., Kang, J., Kelsey, J.: NISTIR 8369: Status Report on the Second Round of the NIST Lightweight Cryptography Standardization Process, https://nvlpubs.nist.gov/nistpubs/ir/2021/NIST.IR.8369.pdf

[25] Turan, M.S., McKay, K.A., Chang, D., Bassham, L., Kang, J., Waller, N.D., Kelsey, J., Hong, D.: NISTIR 8454: Status Report on the Final Round of the NIST Lightweight Cryptography Standardization Process, https://nvlpubs.nist.gov/nistpubs/ir/2023/NIST.IR.8454.pdf

[26] Westerbaan, B., Wiggers, T.: New Post-Quantum Signatures on the Horizon, https://datatracker.ietf.org/meeting/117/materials/slides-117-tls-new-post-quantum-signature-algorithms-on-the-horizon-00

[27] Xiphera: XIP2201B: Ascon, A Lightweight Cryptographic Suite for AEAD and Hashing, https://xiphera.com/products/pdf/XIP2201B_PB.pdf（2023-10-07 閲覧）

[28] Yasuda, K.: The Sum of CBC MACs Is a Secure PRF. In: Pieprzyk, J. (ed.) Topics in Cryptology - CT-RSA 2010, The Cryptographers' Track at the RSA Conference 2010, San Francisco, CA, USA, March 1-5, 2010. Proceedings. Lecture Notes in Computer Science, vol. 5985, pp. 366–381. Springer (2010), https://doi.org/10.1007/978-3-642-11925-5_25

[29] Yasuda, K.: A New Variant of PMAC: Beyond the Birthday Bound. In: Rogaway, P. (ed.) Advances in Cryptology - CRYPTO 2011 - 31st Annual Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2011. Proceedings. Lecture Notes in Computer Science, vol. 6841, pp. 596–609. Springer (2011), https://doi.org/10.1007/978-3-642-22792-9_34

[30] 総務省: 平成 27 年度版 情報通信白書 (2015), https://www.soumu.go.jp/johotsusintokei/whitepaper/ja/h27/pdf/index.html

第 3 章

軽量暗号の実装性能

3.1 節と 3.2 節では、同一プラットフォーム上で、同一の実装者または統一的な実装ポリシーによりハードウェア実装・ソフトウェア実装を行い、統一的な評価環境で性能比較を行った結果を示す。ハードウェア実装についてはブロック暗号を、ソフトウェア実装についてはブロック暗号及び認証暗号を評価対象とした。評価対象としたブロック暗号は表 3.1 に示す 12 種類である。また、評価対象とした認証暗号は表 3.2 に示す 10 種類である。なお、3.1 節と 3.2 節の記載内容は 2016 年度版ガイドライン執筆時点（2017 年 3 月現在）のものであることに留意いただきたい。

表 3.1　評価対象としたブロック暗号

ブロック暗号	ブロック長/鍵長	参照仕様書
AES	128/128	[37]
Camellia	128/128	[1]
CLEFIA	128/128	[45]
TDES	64/168	[4]
LED	64/128	[20]
PRINCE	64/128	[8]
PRESENT	64/80	[28]
Piccolo	64/80	[44]
TWINE	64/80	[46]
Simon	32/64, 64/96, 64/128, 128/128	[6]
Speck	32/64, 64/96, 64/128, 128/128	[6]
Midori	64/128, 128/128	[3]

表 3.2　評価対象とした認証暗号

認証暗号	参照仕様書
ACORN	[49]
AES-GCM	[13]
AES-OTR	[33]
Ascon	[12]
CLOC	[24]
SILC	[23]
JAMBU	[50]
Ketje	[11]
Minalpher	[43]
OCB	[29]

3.3 節では、NIST LWC プロジェクトの最終選考方式として選定された Ascon の実装性能に関する評価結果を紹介する。具体的には、3.3.1 節でサイドチャネル攻撃対策を施していない Ascon のハードウェア実装に関する評価結果、3.3.2 節でサイドチャネル攻撃対策を施していない Ascon のソフトウェア実装に関する評価結果、そして 3.3.3 節で Ascon-128 の物理攻撃耐性を含めた実装性能についてまとめている。これらの Ascon の実装性能については、2022 年度と 2023 年度に公開された CRYPTREC 外部評価報告書 [53, 54] に基づき、2023 年 9 月現在の調査結果を記載している。

3.1 ブロック暗号の実装性能

3.1.1 ハードウェア実装評価

暗号回路の実装方式は用途に応じて様々な形態が考えられるが、本ガイドラインでは図 3.1 記載の「Unrolled 実装」、「Round 実装」、「Serial 実装」の 3 つの基本実装方式を採る。図 3.1 中、Round Function は、各暗号アルゴリズムで規定される基本関数の演算を行う組み合わせ回路を指す。12 種類のアルゴリズムに対するハードウェア実装評価では、暗号化演算のみと暗号化・復号演算の双方を同一のモジュールで実行し、その切り替えは制御信号でのみ行う実装の 2 通りに対して実装評価を行う。なお、本ガイドラインでは前者の実装を「Enc」と後者を「Enc/Dec」と表記する。

ブロック暗号アルゴリズムは一般に鍵スケジューリング機能と、暗号化・復号機能に分割できる。本ガイドラインにおける評価では、前記の両機能を持つ暗号回路を構成する。当該暗号化回路は、鍵スケジューリング機能、暗号化・復号機能ともに同一のクロックで動作する仕様で構成する。また、鍵スケジューリングにレジスタが不要なアルゴリズムでは、当該レジスタを削除して実装する。

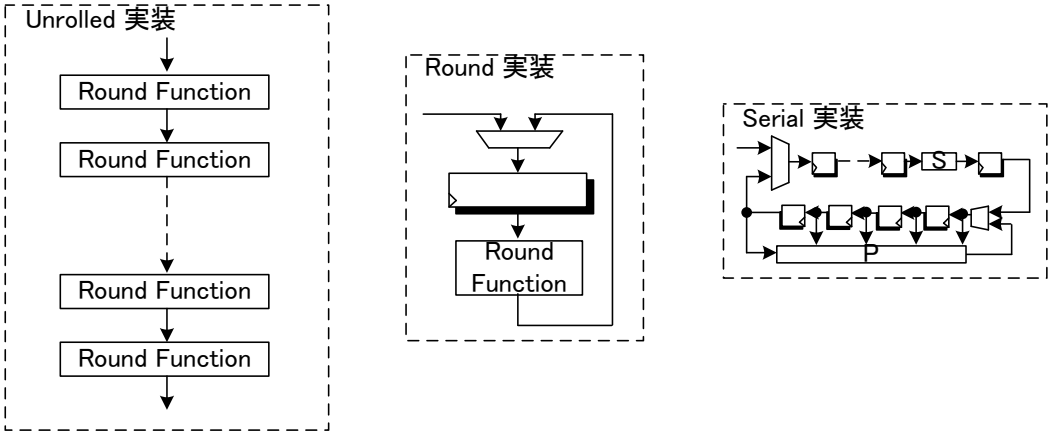

図 3.1　基本実装方式

3.1.1.1 性能比較

以下、表 3.4 から表 3.6 に実装評価結果のデータを示す。また、インターフェースの回路規模を除いた各実装における回路規模の比較結果を表 3.7 に示す。さらに、各実装に対する回路規模、処理速度、ピーク電流、リーク電流についてグラフによる比較結果を図 3.3 から図 3.26 に示す。表中、(comp) はブロック暗号の構成要素である S-box を合成体上の演算回路として実装したことを表し、(table) は S-box をテーブル参照として記述し、合成ツール依存で回路を構成したことを表す。

まず、表 3.4 の注目点について述べる。Unrolled 実装の回路規模は AES を含む CRYPTREC 暗号リスト掲載のブロック暗号が軽量暗号・低遅延暗号（LED は除く）に対して突出して大きい。この要因は表 3.8 に示されるように、S-box の性能差が支配的である。8-bit S-box は遅延優先の Table 実装の場合、PRESENT や PRINCE などの 4-bit S-box に対して 100 倍以上大きくなる一方、遅延は 5 倍程度大きい。仮に 8-bit S-box で 4-bit S-box と同等の遅延性能を達成しようとした場合、4-bit S-box よりもラウンド数を 1/4 程度にすることができないと遅延の観点からは効率が悪く、遅延が同等になっても S-box に関する回路規模は 100 倍近く大きくなることを意味する。この視点から PRINCE はほぼ AES と同じラウンド数となっているため、S-box の性能差がダイレクトに全体性能の差になって表れている。PRINCE は PRESENT に対してラウンド数がおよそ 1/3 であるが、表 3.9 に示すように P 層は PRESENT の方が高速であるため、3 倍の差はなく、回路規模、遅延ともに 2 倍程度 PRINCE が優れる結果となる。復号の影響については、GFN (Generalized Feistel Networks) 型や α-reflection property の効果により、PRINCE、TWINE、Piccolo と PRESENT や LED の回路規模の差はさらに広がる。また、SIMON と Midori については PRINCE と同程度の処理性能を持つ。

Unrolled 実装の遅延についてもう一つ着目すべき点は、AES、LED など復号鍵を一端生成しないと復号できないアルゴリズムや Camellia や CLEFIA のような中間鍵を生成するアルゴリズムはクリティカルパスにその分の遅延が乗るため、

Unrolled 実装では不利になる。Piccolo や PRINCE は暗号化と暗号化・復号回路の最大動作周波数がほぼ同じように構成できる。

次に、round 実装と serial 実装との性能差を比較する。AES では 9kgate 程度の削減が可能であるのに対して、PRESENT、PRINCE は数百 gate から 1kgate の削減に留まる。S-box や P 層などの演算器を削減しても、表 3.8 と表 3.9 からわかるように、その削減効果が限定的であるためである。その一方で、処理性能は 1/10 以下なるため、例えば処理速度 / 回路規模などの指標を導入すれば効率が悪い。コードとしての可読性も悪いため、回路規模に対してなんらかの強い実装制約がない限り、軽量暗号で serial 実装を採用する必要はないであろう。

最後に、serial 実装について述べる。文献 [36] において AES は 2.4kgate で実装されていたが、本実装では 1kgate 程度増加している。その要因は、フリップフロップ 1 つあたりのゲート換算が文献 [36] よりもおよそ 1 ゲート程度大きいことや、合成によって挿入されるバッファ、制御回路の構成などが差分として挙げられる。文献 [36] に記載されるような Scan-FF を積極的に利用するような最適化を実施していないことも差分になる。PRESENT、PRINCE については本ガイドラインの結果でも AES より 1〜2kgate 程度小さい回路となっている。PRESENT と PRINCE との間に回路規模としての差はないが、サイクル数は PRINCE が PRESENT の 1/2 程度で実装できる。

3.1.1.2 評価方法の概要

本節では、評価方法の概要を示す。今回の評価では、各種軽量暗号を、オープンソースの CMOS セルライブラリを利用して、回路リソースの使用量及び最大動作周波数などに関するデータを計測した。実装環境を表 3.3 に示す。

表 3.3　実装環境

論理合成ツール	Design Compiler (Version G-2012.06-SP5)
パワー解析ツール	PrimeTime PX (Version G-2012.06-SP3-2)
合成制約	面積最小
ライブラリ	NANGATE Open Cell Library (45-nm CMOS) `https://www.nangate.com/`
遅延条件	NangateOpenCellLibrary_slow (最悪条件の仮想遅延)

以下に実装する論理回路の機能概略を述べる。

F1. 鍵長は 80bit 以上でかつ規定される最小のパラメータで評価を行う。但し LED に関してはテストベクタが提供されている 128bit 鍵長で評価する。
F2. 暗号化、暗号化・復号回路の実装とする。
F3. CPU のコプロセッサとしての利用を想定し、コンパクトで低電力とされる APB バス [2] 接続が可能な設計とする。

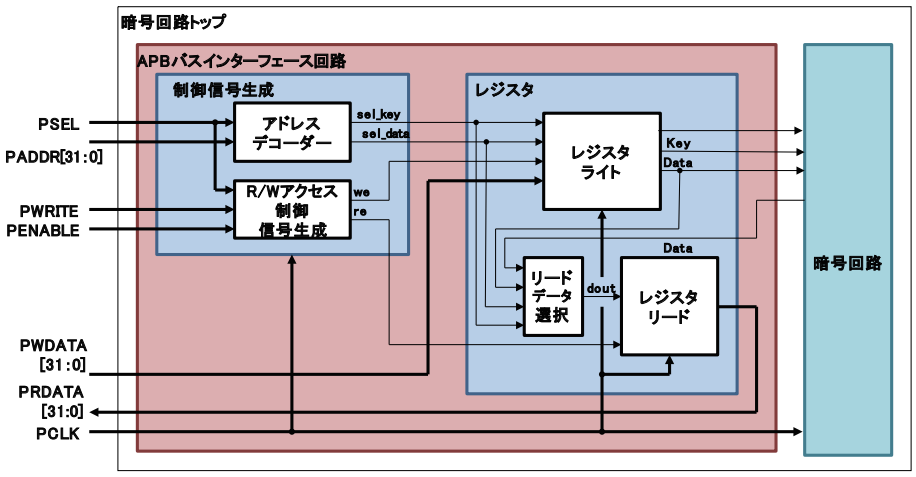

図 3.2　APB バスと暗号回路

APB バスと暗号回路のブロック図を図 3.2 に示す。図中の信号の意味は以下の通りである。

* PCLK: バスクロック信号
* PRESETn: 非同期リセット信号
* PADDR[31:0]: アドレス信号
* PSEL:IP 選択信号
* PENABLE: イネーブル信号
* PWRITE: ライト信号、1: ライト, 0: リード
* PWDATA[31:0]: ライトデータ
* PRDATA[31:0]: リードデータ

このほか、APB 信号規定としては、PSTRB[3:0]（ライトストローブ信号）や PREADY（APB 転送延長信号）があるが、今回の評価では使用していない。

次に設計方針を述べる。

P1. 各アルゴリズムに対して 3 種類の実装を行う: (i) 典型的な round ベースの実装に加え、(ii) 1 サイクルで処理が完了する Unrolled 実装、(iii) データパスを S-box のサイズとする serial 実装を行う。
P2. 鍵スケジュールは on-the-fly で実装する。
P3. CMOS セルライブラリを直接インスタンスするような最適化は行わず、ライブラリ非依存で合成可能な記述とする。

以上の方針に基づき設計した論理回路に対してサイクル数、最大動作周波数（最大遅延)、スループット、ゲート数、ピーク電力、リーク電力を評価している。

表 3.4　Unrolled 実装の評価結果

アルゴリズム	ブロック長 [bit]	鍵長 [bit]	1ブロックあたりの処理サイクル数	最大動作周波数 [MHz]	処理速度 [Gbps]	回路規模 [kgate]	ピーク電力 [mW]	リーク電力 [uW]
Unrolled, Enc								
AES(table)(128/128)	128	128	1	25.7	3.3	112.4	-	-
AES（comp)(128/128)	128	128	1	13.4	1.7	78.8	175.6	939.6
Camellia(comp)(128/128)	128	128	1	7.8	1.0	60.2	136.5	706.7
CLEFIA(128/128)	128	128	1	5.7	0.7	74.6	195.5	891.0
SIMON(128/128)	128	128	1	24.7	3.2	63.2	172.2	685.9
SPECK(128/128)	128	128	1	3.2	0.4	44.4	73.0	417.0
Midori(128/128)	128	128	1	38.5	4.9	34.6	118.2	446.1
TDES(64/168)	64	168	1	10.0	0.6	55.4	111.9	652.2
LED(64/128)	64	128	1	6.9	0.4	74.5	99.1	824.0
PRINCE(64/128)	64	128	1	57.1	3.7	9.8	28.1	107.4
SIMON(64/128)	64	128	1	27.8	1.8	23.8	71.5	260.4
SPECK(64/128)	64	128	1	7.3	0.5	19.5	35.6	183.0
Midori(64/128)	64	128	1	46.5	3.0	12.3	34.9	149.0
SIMON(64/96)	64	96	1	41.3	2.6	20.3	56.7	218.1
SPECK(64/96)	64	96	1	7.6	0.5	18.6	35.4	174.7
PRESENT(64/80)	64	80	1	34.3	2.2	23.9	57.8	259.6
Piccolo(64/80)	64	80	1	18.0	1.2	19.1	61.0	224.8
TWINE(64/80)	64	80	1	24.8	1.6	19.5	43.8	221.2
SIMON(32/64)	32	64	1	39.4	1.3	9.0	30.5	97.4
SPECK(32/64)	32	64	1	15.3	0.5	8.2	17.3	78.0
Unrolled, Enc/Dec								
AES(table)(128/128)	128	128	1	11.4	1.5	208.4	337.2	2612.0
AES（comp)(128/128)	128	128	1	6.4	0.8	144.2	294.3	1734.3
Camellia(comp)(128/128)	128	128	1	7.7	1.0	63.4	133.8	754.9
CLEFIA(128/128)	128	128	1	5.7	0.7	74.3	195.5	891.0
SIMON(128/128)	128	128	1	13.0	1.7	74.1	187.0	803.7
SPECK(128/128)	128	128	1	1.1	0.1	69.1	127.1	672.5
Midori(128/128)	128	128	1	30.7	3.9	55.6	123.7	720.2
TDES(64/168)	64	168	1	9.6	0.6	56.5	112.9	673.3
LED(64/128)	64	128	1	3.1	0.2	215.4	103.1	815.6
PRINCE(64/128)	64	128	1	56.1	3.6	10.1	29.1	108.2
SIMON(64/128)	64	128	1	16.8	1.1	27.5	83.2	299.1
SPECK(64/128)	64	128	1	2.7	0.2	29.9	62.3	290.8
Midori(64/128)	64	128	1	37.7	2.4	20.6	37.1	256.4
SIMON(64/96)	64	96	1	21.5	1.4	23.8	62.9	255.3
SPECK(64/96)	64	96	1	2.9	0.2	28.6	57.8	278.0
PRESENT(64/80)	64	80	1	26.8	1.7	43.8	127.8	505.4
Piccolo(64/80)	64	80	1	16.3	1.0	22.8	64.8	264.0
TWINE(64/80)	64	80	1	13.1	0.8	25.6	50.9	292.2
SIMON(32/64)	32	64	1	23.6	0.8	10.4	30.9	111.8
SPECK(32/64)	32	64	1	6.9	0.2	12.4	27.5	121.7

表 3.5　round 実装の評価結果

アルゴリズム	ブロック長 [bit]	鍵長 [bit]	1ブロックあたりの処理サイクル数	最大動作周波数 [MHz]	処理速度 [Gbps]	回路規模 [kgate]	ピーク電力 [mW]	リーク電力 [uW]
Round, Enc								
AES（comp)(128/128)	128	128	11	108.2	1.259	15.4	36.1	152.6
Camellia(comp)(128/128)	128	128	23	103.0	0.573	10.8	46.6	107.7
CLEFIA(128/128)	128	128	19	145.8	0.982	10.1	39.8	99.6
SIMON(128/128)	128	128	68	371.7	0.700	7.0	17.4	69.9
SPECK(128/128)	128	128	32	50.3	0.201	7.2	11.4	66.2
Midori(128/128)	128	128	20	386.1	2.471	7.1	11.9	79.7
TDES(64/168)	64	168	48	164.2	0.219	7.9	13.9	76.2
LED(64/128)	64	128	48	208.3	0.278	6.3	5.3	52.5
PRINCE(64/128)	64	128	13	234.2	1.153	5.1	16.4	47.1
SIMON(64/128)	64	128	44	371.7	0.541	5.3	12.4	51.1
SPECK(64/128)	64	128	27	95.8	0.227	5.3	10.5	48.3
Midori(64/128)	64	128	16	340.1	1.361	4.7	11.4	49.1
SIMON(64/96)	64	96	42	392.2	0.598	4.5	11.8	44.1
SPECK(64/96)	64	96	26	95.8	0.236	4.6	10.0	42.4
PRESENT(64/80)	64	80	33	326.8	0.634	4.1	4.7	33.4
Piccolo(64/80)	64	80	27	262.5	0.622	3.5	3.4	34.2
TWINE(64/80)	64	80	36	311.5	0.554	4.4	4.6	40.0
SIMON(32/64)	32	64	32	369.0	0.369	2.9	9.8	28.0
SPECK(32/64)	32	64	22	175.1	0.255	2.9	8.4	26.8
Round, Enc/Dec								
AES（comp)(128/128)	128	128	11	107.0	1.245	18.7	44.1	193.6
Camellia(comp)(128/128)	128	128	23	103.0	0.573	11.8	44.6	121.9
CLEFIA(128/128)	128	128	19	143.1	0.964	9.9	38.1	99.0
SIMON(128/128)	128	128	68	310.6	0.585	7.8	17.2	78.4
SPECK(128/128)	128	128	32	49.9	0.200	9.6	11.2	92.7
Midori(128/128)	128	128	20	271.0	1.734	8.4	11.9	96.9
TDES(64/168)	64	168	48	161.6	0.215	10.6	13.9	114.0
LED(64/128)	64	128	48	188.7	0.252	7.2	6.5	66.6
PRINCE(64/80)	64	128	13	224.7	1.106	5.3	18.7	50.3
SIMON(64/128)	64	128	44	342.5	0.498	6.0	12.4	58.2
SPECK(64/128)	64	128	27	93.5	0.222	6.7	10.6	63.2
Midori(64/128)	64	128	16	266.7	1.067	5.3	11.4	57.5
SIMON(64/96)	64	96	42	342.5	0.522	5.1	11.6	49.9
SPECK(64/96)	64	96	26	93.5	0.230	5.9	9.9	55.7
PRESENT(64/80)	64	80	33	280.9	0.545	4.7	4.9	44.8
Piccolo(64/80)	64	80	27	261.8	0.621	3.8	3.3	38.5
TWINE(64/80)	64	80	36	302.1	0.537	4.7	4.5	42.8
SIMON(32/64)	32	64	32	359.7	0.360	3.3	9.9	31.8
SPECK(32/64)	32	64	22	167.5	0.244	3.6	8.7	34.1

表 3.6　serial 実装の評価結果

アルゴリズム	ブロック長 [bit]	鍵長 [bit]	1ブロックあたりの処理サイクル数	最大動作周波数 [MHz]	処理速度 [Gbps]	回路規模 [kgate]	ピーク電力 [mW]	リーク電力 [uW]
Serial, Enc								
AES（comp)(128/128)	128	128	226	112.2	63.6	6.3	18.5	76.8
Camellia(comp)(128/128)	128	128	360	109.5	38.9	6.6	14.4	66.1
CLEFIA(128/128)	128	128	175	114.2	83.5	6.2	13.1	61.3
SIMON(128/128)	128	128	4481	269.5	7.7	4.8	8.2	47.1
SPECK(128/128)	128	128	2177	291.5	17.1	5.0	8.2	48.4
Midori(128/128)	128	128	489	254.5	66.6	4.9	11.9	49.2
LED(64/128)	64	128	1872	344.8	11.8	5.6	2.2	50.0
PRINCE(64/128)	64	128	247	246.3	63.8	3.9	8.7	40.0
SIMON(64/128)	64	128	1537	309.6	12.9	3.7	4.8	36.2
SPECK(64/128)	64	128	993	339.0	21.8	3.9	5.4	37.4
Midori(64/128)	64	128	393	253.2	41.2	3.5	11.4	35.3
SIMON(64/96)	64	96	1441	328.9	14.6	3.3	4.5	31.7
SPECK(64/96)	64	96	929	314.5	21.7	3.4	5.1	33.1
PRESENT(64/80)	64	80	563	186.9	21.2	3.9	3.4	36.4
Piccolo(64/80)	64	80	433	300.3	44.4	3.5	2.0	28.5
TWINE(64/80)	64	80	324	277.8	54.9	4.1	2.8	29.6
SIMON(32/64)	32	64	577	389.1	21.6	2.2	3.7	20.8
SPECK(32/64)	32	64	417	390.6	30.0	2.3	5.5	21.9
Serial, Enc/Dec								
AES（comp)(128/128)	128	128	226	108.6	61.5	7.2	14.5	61.2
Camellia(comp)(128/128)	128	128	360	108.3	38.5	7.3	14.8	63.1
CLEFIA(128/128)	128	128	175	113.1	82.7	6.8	12.5	59.3
SIMON(128/128)	128	128	4481	277.8	7.9	5.6	9.7	57.4
SPECK(128/128)	128	128	2177	316.5	18.6	5.9	8.3	57.2
Midori(128/128)	128	128	489	204.1	53.4	5.3	11.9	53.9
LED(64/128)	64	128	1872	303.0	10.4	6.9	1.4	34.5
PRINCE(64/128)	64	128	247	245.7	63.7	3.8	8.4	36.2
SIMON(64/128)	64	128	1537	277.0	11.5	4.5	5.6	45.3
SPECK(64/128)	64	128	993	317.5	20.5	4.8	7.6	46.2
Midori(64/128)	64	128	393	220.3	35.9	3.8	11.4	37.7
SIMON(64/96)	64	96	1441	298.5	13.3	3.9	5.1	39.0
SPECK(64/96)	64	96	929	280.1	19.3	4.1	7.6	40.1
PRESENT(64/80)	64	80	563	170.9	19.4	4.5	2.4	25.8
Piccolo(64/80)	64	80	433	292.4	43.2	3.7	2.0	23.4
TWINE(64/80)	64	80	324	270.3	53.4	4.2	2.6	28.4
SIMON(32/64)	32	64	577	299.4	16.6	2.6	4.1	25.7
SPECK(32/64)	32	64	417	295.9	22.7	2.8	6.3	27.3

表 3.7　各実装の回路規模

アルゴリズム	インターフェースの除いた暗号回路のみの回路規模[kgate]					
	Unrolled, Enc	Unrolled, Enc/Dec	Round, Enc	Round, Enc/Dec	Serial, Enc	Serial, Enc/Dec
AES(table)(128/128)	109.7	205.6	—	—	—	—
AES（comp)(128/128)	76.1	141.5	12.4	15.6	3.2	4.1
Camellia(comp)(128/128)	57.4	60.6	8.0	9.0	4.1	4.3
CLEFIA(128/128)	71.5	71.5	7.3	7.1	3.6	3.8
SIMON(128/128)	60.4	71.3	4.3	5.0	2.1	2.9
SPECK(128/128)	41.6	66.4	4.4	6.8	2.2	3.1
Midori(128/128)	31.8	52.9	4.3	5.6	2.2	2.6
TDES(64/168)	52.8	53.8	5.3	7.9	—	—
LED(64/128)	71.9	212.9	3.8	4.7	3.0	4.3
PRINCE(64/128)	7.8	8.1	2.7	3.0	1.6	1.8
SIMON(64/128)	21.8	25.4	3.2	3.9	1.7	2.5
SPECK(64/128)	17.4	27.8	3.2	4.6	1.8	2.7
Midori(64/128)	10.2	18.5	2.6	3.2	1.5	1.7
SIMON(64/96)	18.4	21.9	2.7	3.2	1.4	2.0
SPECK(64/96)	16.8	26.8	2.8	4.1	1.6	2.3
PRESENT(64/80)	22.0	42.1	2.2	2.9	2.0	2.8
Piccolo(64/80)	17.4	21.1	1.6	1.9	1.1	1.3
TWINE(64/80)	17.8	23.9	2.7	2.9	2.4	2.5
SIMON(32/64)	7.8	9.2	1.7	2.1	1.0	1.4
SPECK(32/64)	7.0	11.2	1.7	2.4	1.1	1.6

表 3.8　S-box の比較

Module	Area [gate]	Path delay [ns]
AES 8-bit S-box (Table)	3,194	2.43
AES 8=bit S-box (Composite)	315	5.75
PRESENT 4-bit S-box (Table)	26	0.57
PRINCE 4-bit S-box (Table)	18	0.48

表 3.9　P 層の比較

Module	Area [gate]	Path delay [ns]
AES 128-bit permutation	864	0.89
PRESENT 64-bit permutation	0	0
PRINCE 64-bit permutation	192	0.51

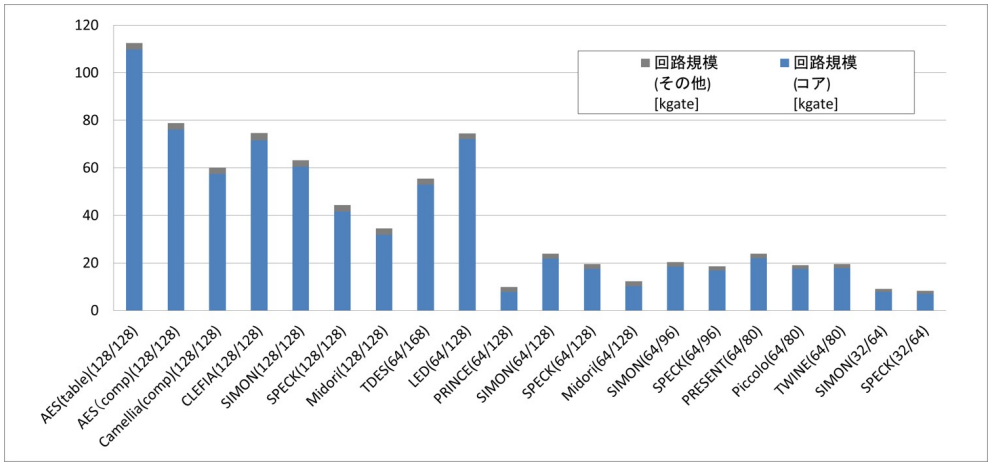

図 3.3　Enc, Unrolled 実装の回路規模

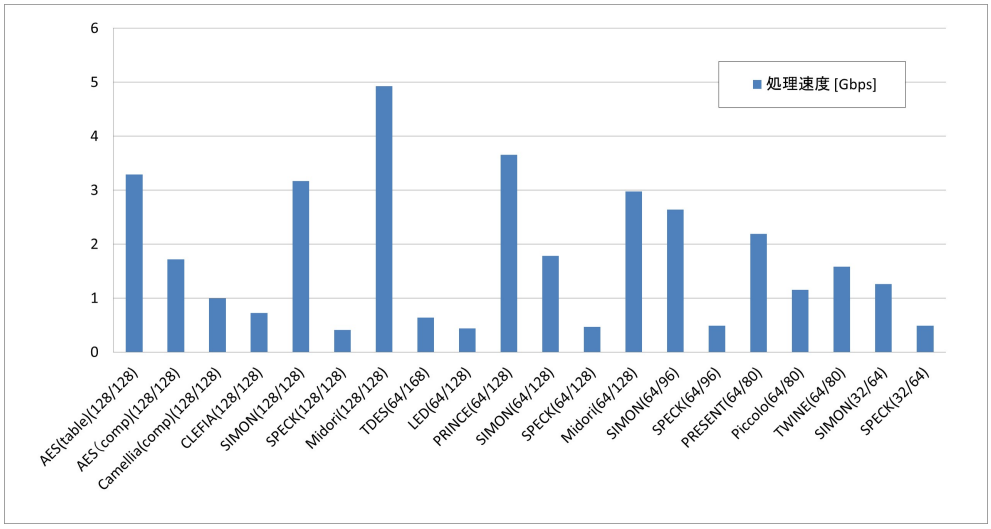

図 3.4　Enc, Unrolled 実装の処理速度

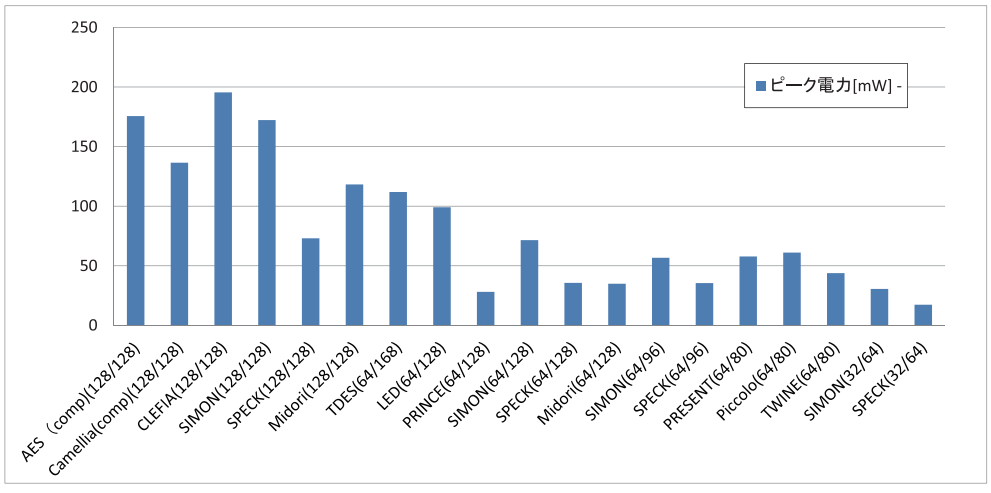

図 3.5　Enc, Unrolled 実装のピーク電流

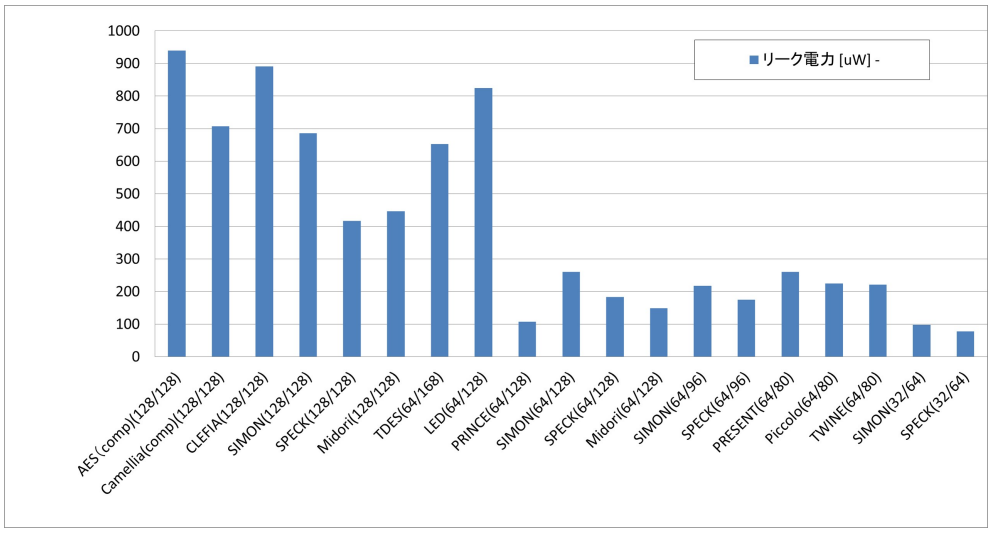

図 3.6　Enc, Unrolled 実装のリーク電流

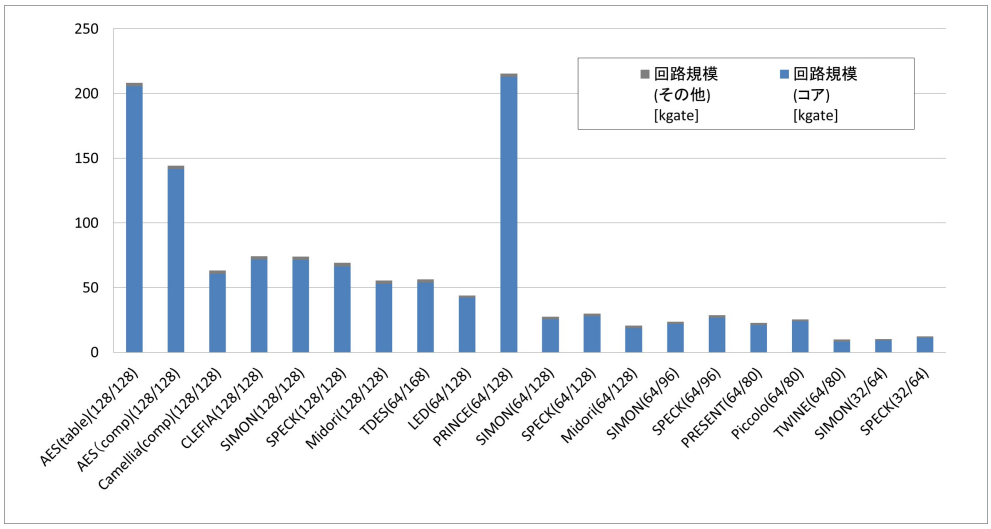

図 3.7　Enc/Dec, Unrolled 実装の回路規模

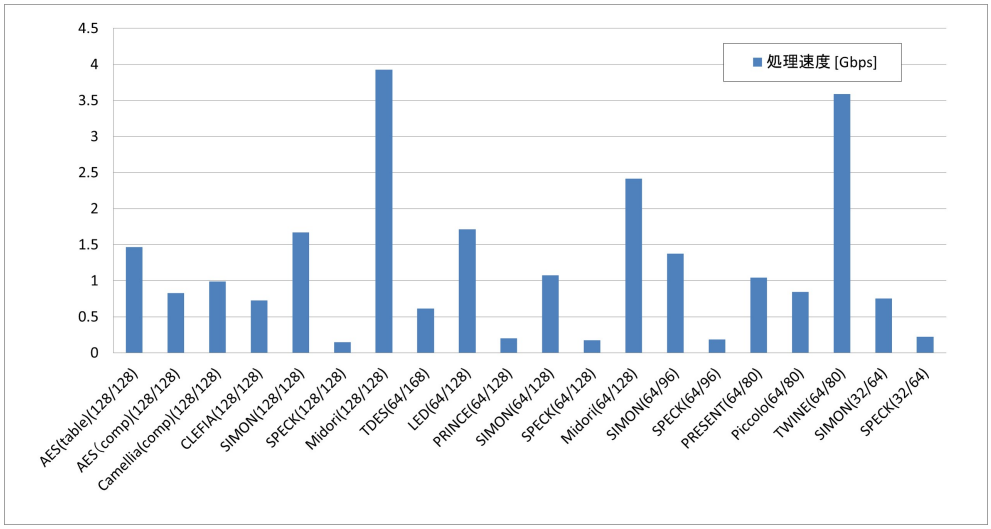

図 3.8　Enc/Dec, Unrolled 実装の処理速度

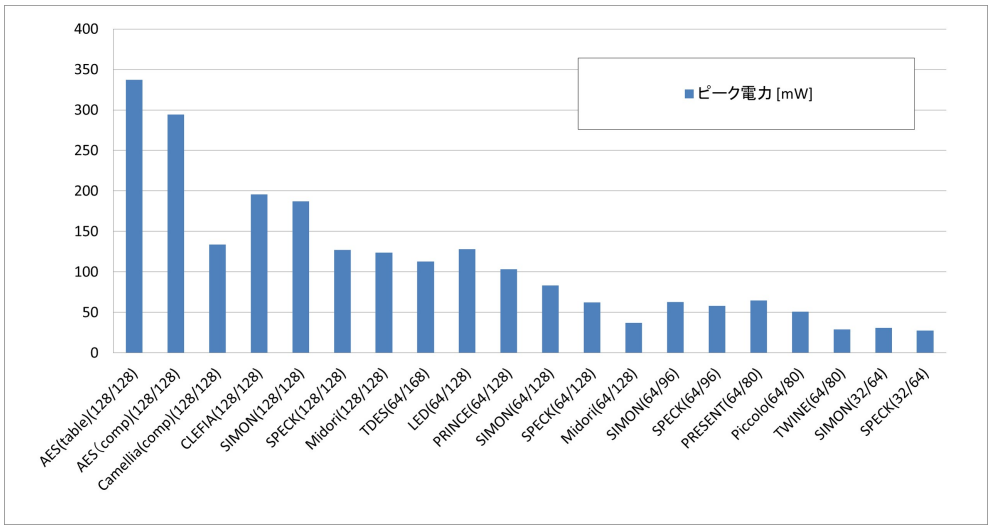

図 3.9　Enc/Dec, Unrolled 実装のピーク電流

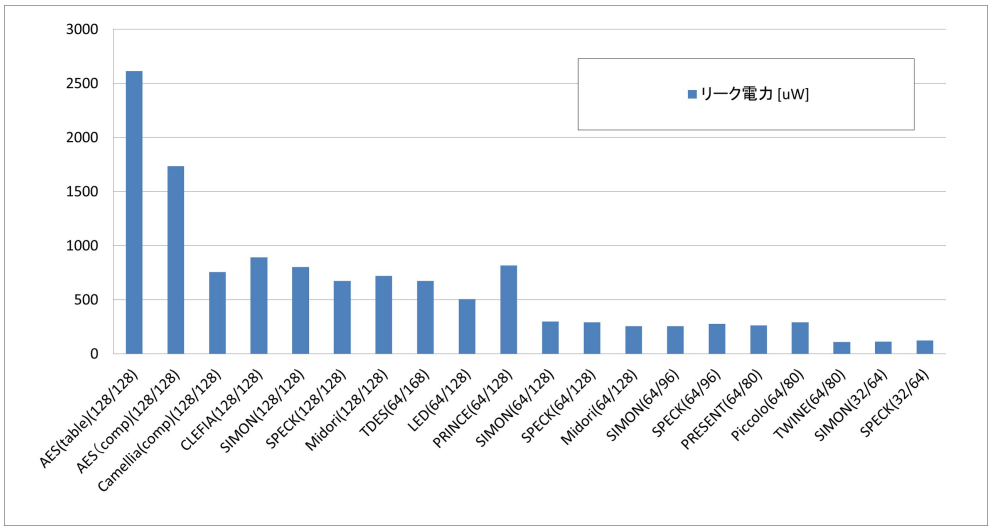

図 3.10　Enc/Dec, Unrolled 実装のリーク電流

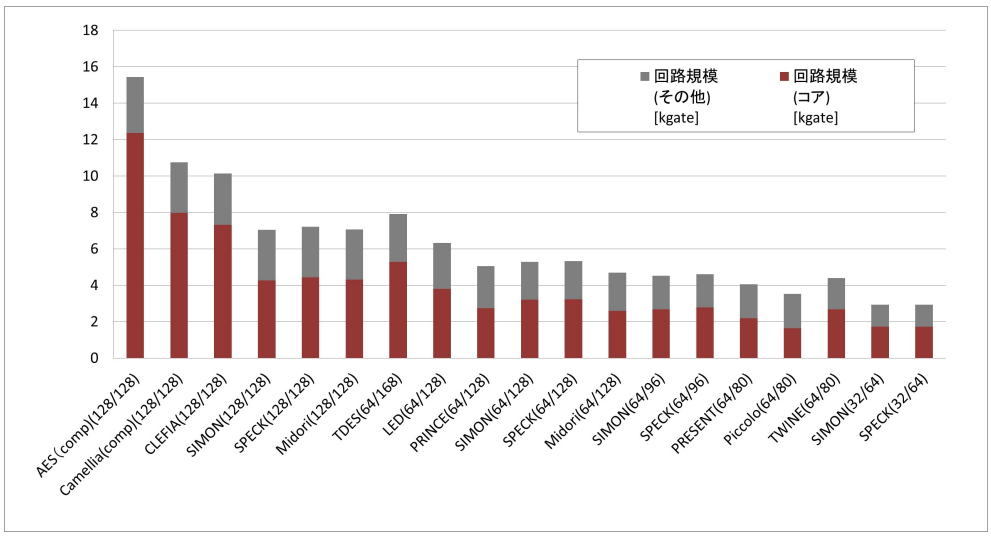

図 3.11　Enc,Round 実装の回路規模

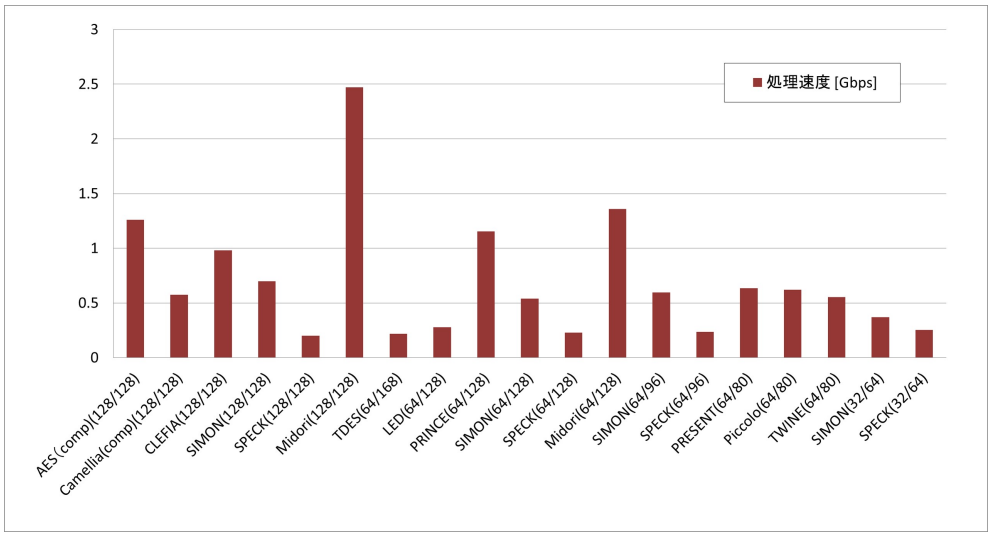

図 3.12　Enc,Round 実装の処理速度

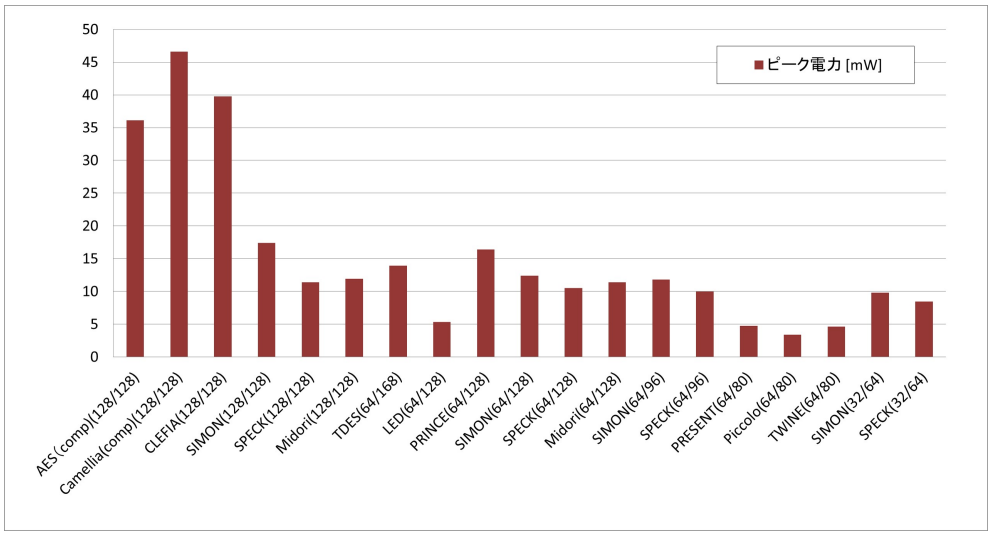

図 3.13　Enc,Round 実装のピーク電流

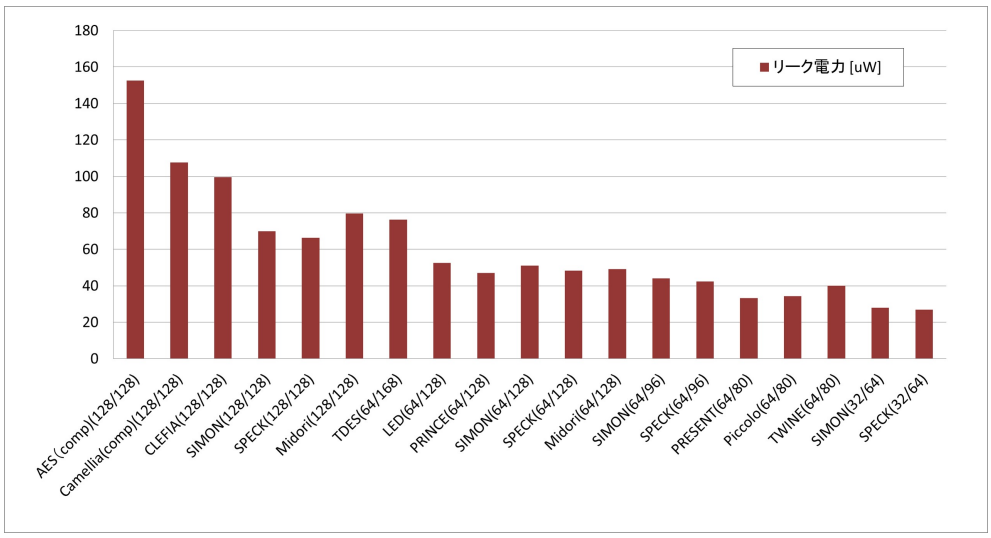

図 3.14　Enc,Round 実装のリーク電流

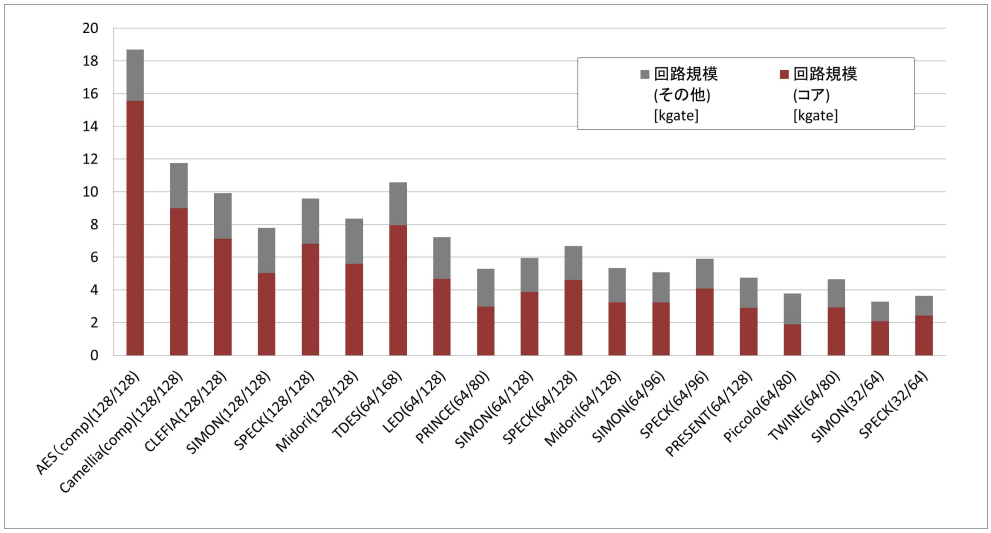

図 3.15　Enc/Dec,Round 実装の回路規模

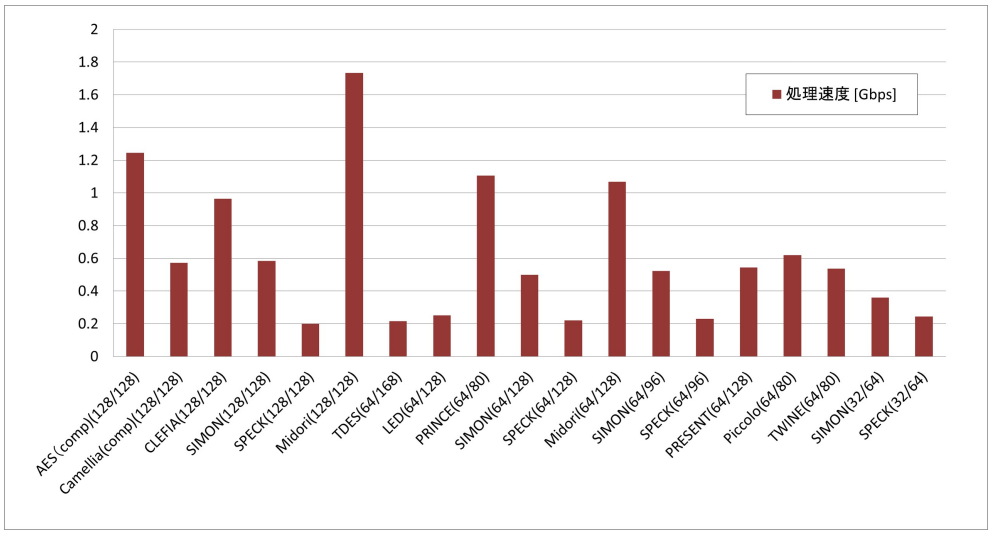

図 3.16　Enc/Dec,Round 実装の処理速度

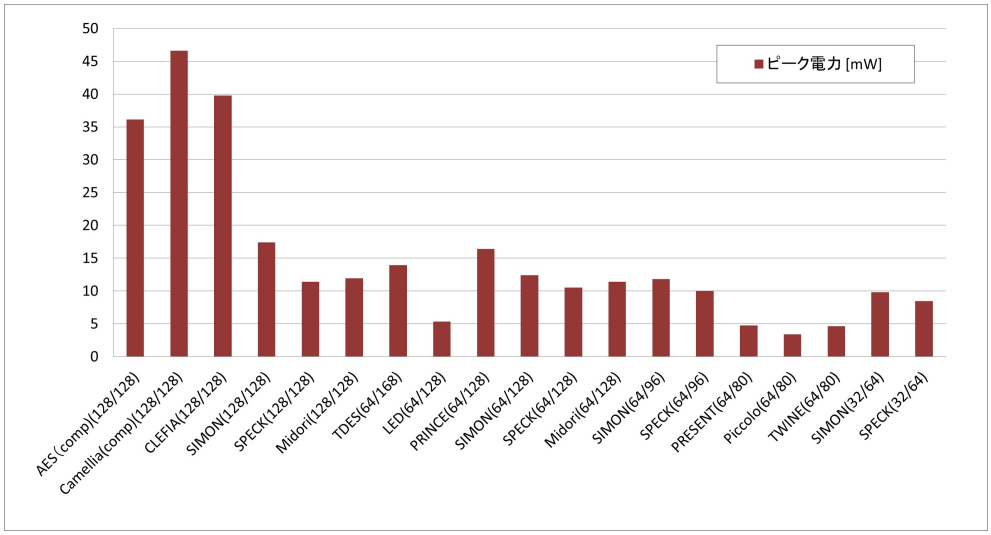

図 3.17　Enc/Dec,Round 実装のピーク電流

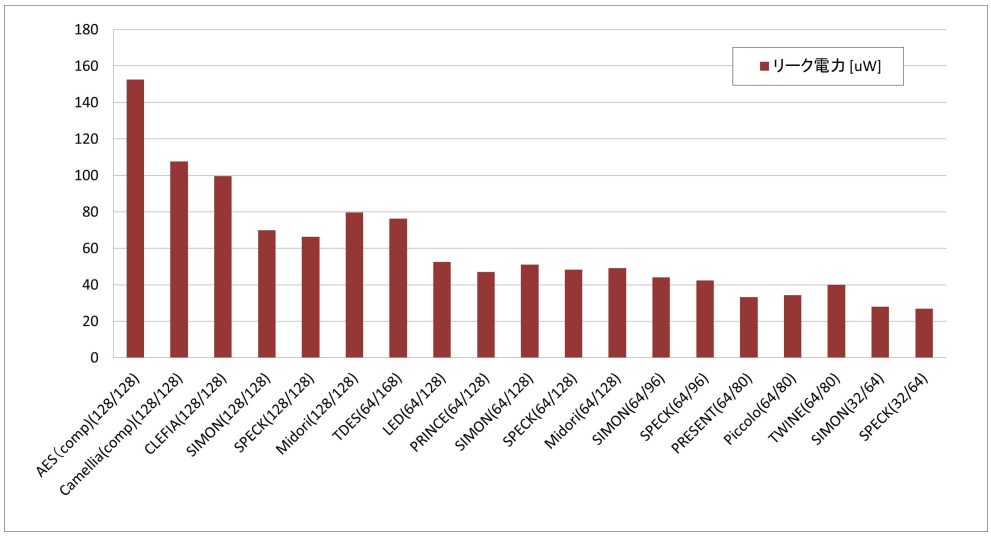

図 3.18　Enc/Dec,Round 実装のリーク電流

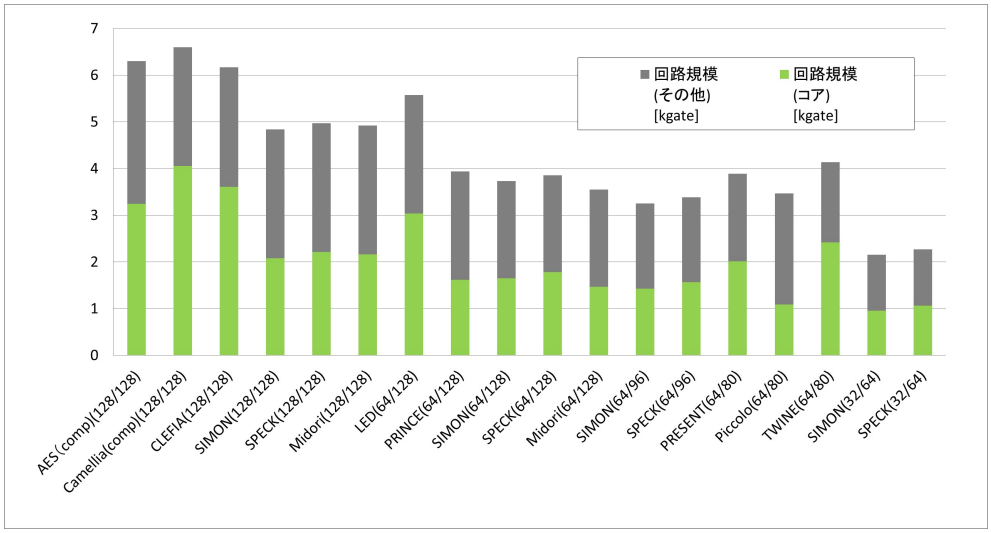

図 3.19　Enc,Serial 実装の回路規模

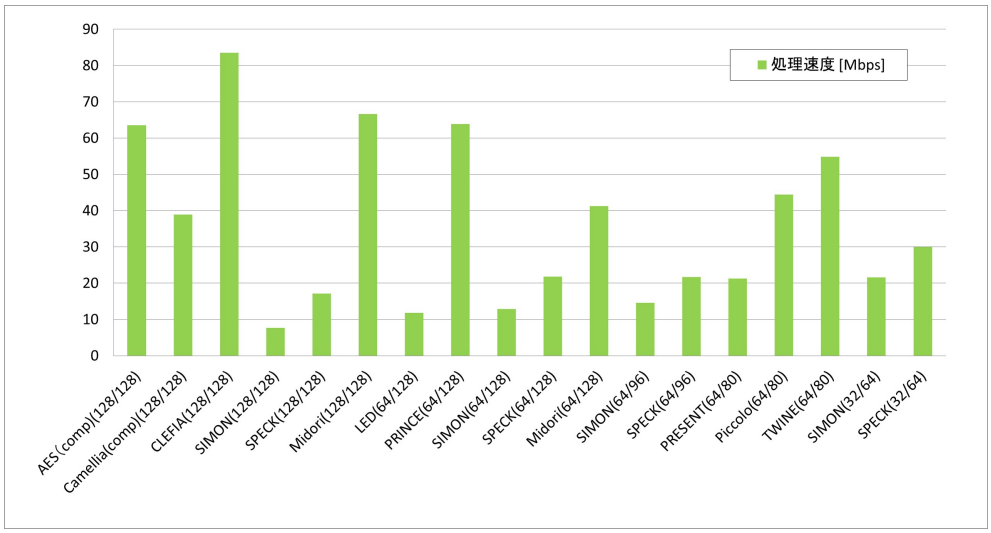

図 3.20　Enc,Serial 実装の処理速度

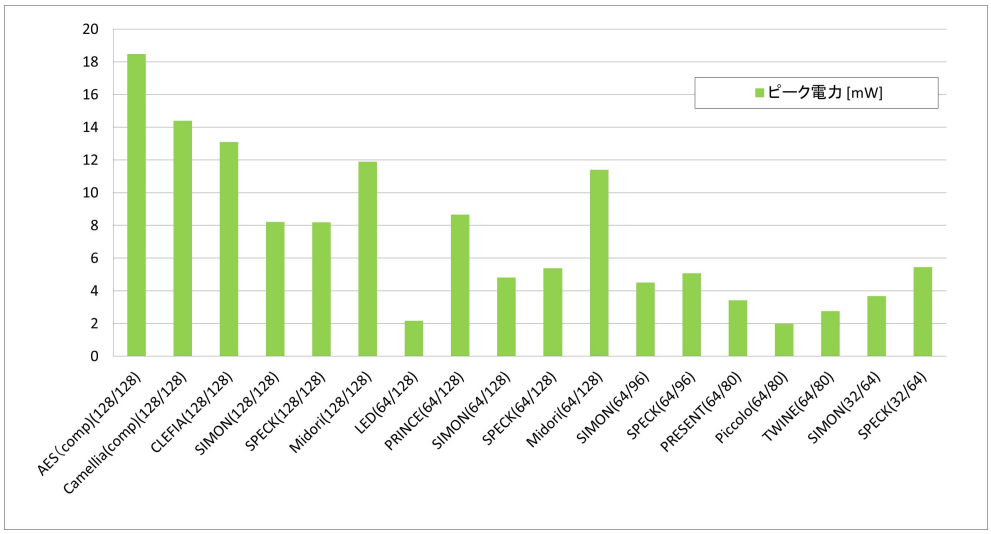

図 3.21　Enc,Serial 実装のピーク電流

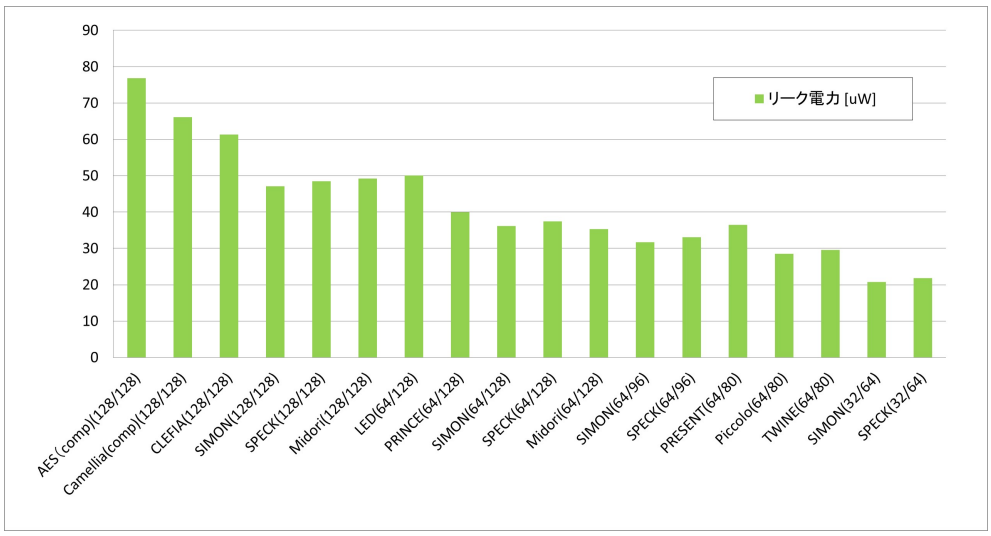

図 3.22　Enc,Serial 実装のリーク電流

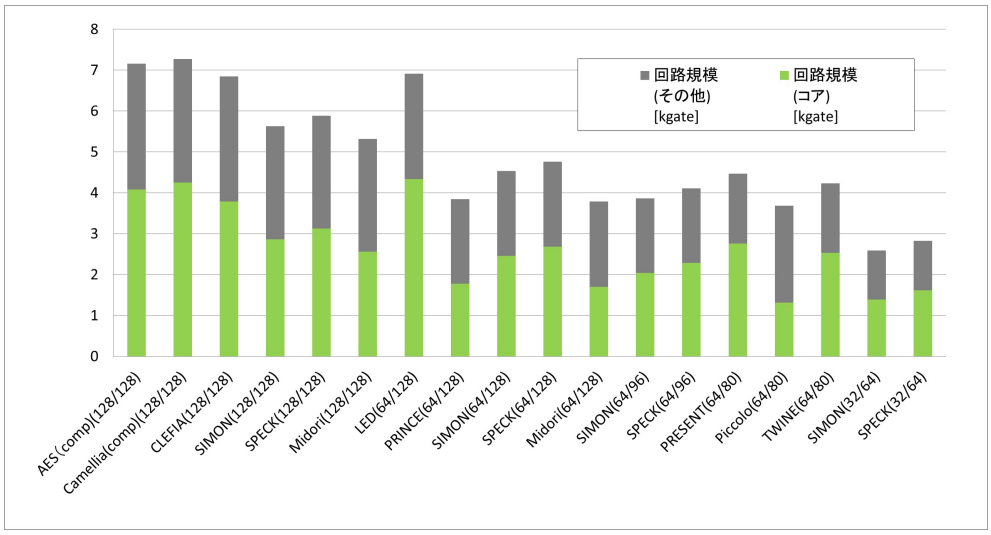

図 3.23　Enc/Dec,Serial 実装の回路規模

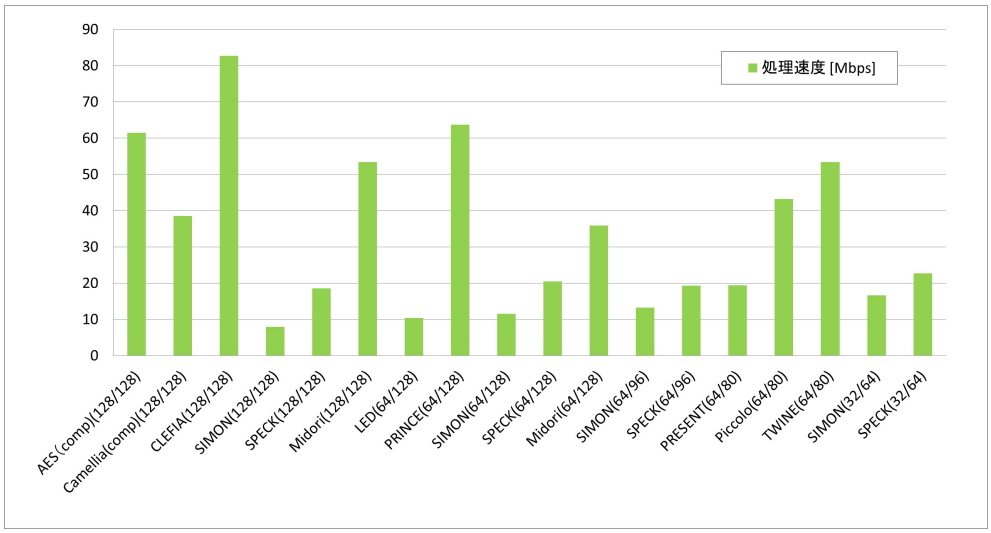

図 3.24　Enc/Dec,Serial 実装の処理速度

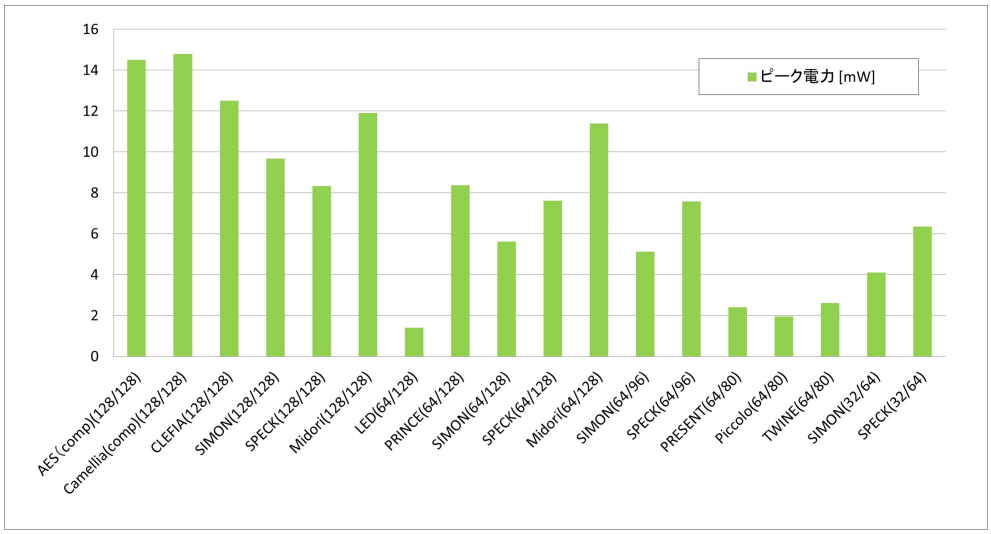

図 3.25　Enc/Dec,Serial 実装のピーク電流

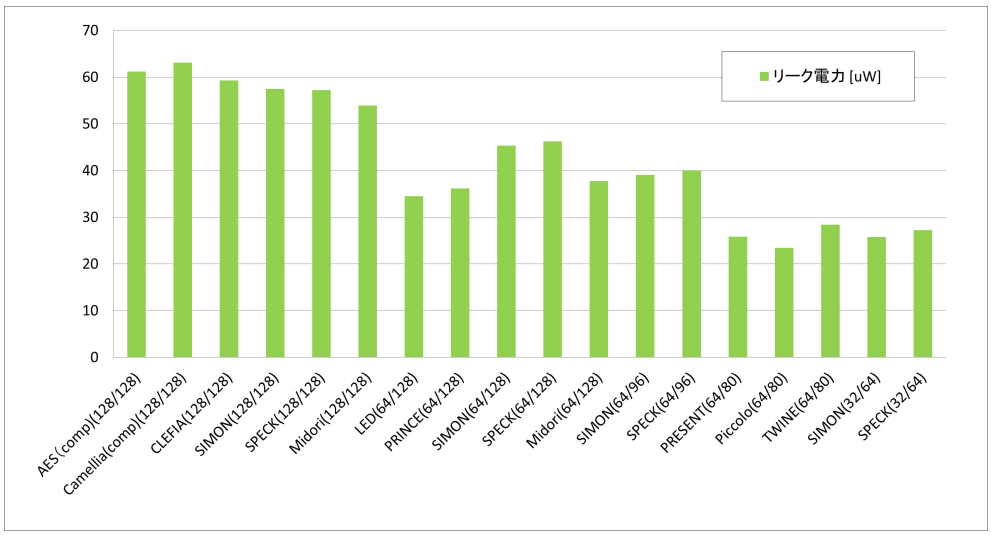

図 3.26　Enc/Dec,Serial 実装のリーク電流

3.1.2 ソフトウェア実装評価

本節では、組み込みマイコン上の限定されたメモリリソースのもとで、軽量ブロック暗号をソフトウェア実装した場合の速度性能について計測、比較した結果を示す。

3.1.2.1 性能評価

本節ではルネサスエレクトロニクス社製 16 ビットマイコン RL78 において、9 つのブロック暗号を実装した結果を示す。表 3.10〜表 3.21 の Category の欄について、ROM-Min は ROM サイズを最小化する実装を行ったもの、(n, m) は ROM サイズが n バイト以下 $(n = 512, 1024)$、RAM サイズが m バイト以下（$m = 64, 128$) で実装を行ったもの、Fast は ROM サイズが約 2K バイト以下で高速化を目指して実装を行ったものである。すべての実装は、暗号化・復号と並行して鍵スケジュールを行う実装方式（on-the-fly key scheduling）を採用している。

■AES

表 3.10 に、RL78 にて AES を実装した結果を示す。暗号化のみを実装する場合、RAM サイズは 64 バイトで十分であったため、RAM サイズ 128 バイトでの実装は省略した。また、暗号化・復号の両方を実装する場合は、S-box だけで ROM サイズを 512 バイト消費するので、ROM サイズ 1024 バイトのカテゴリのみ実装を行っている。

実装方法は、基本的にメモリサイズの制約が厳しくなるにつれ、ラウンド内のループを増やしていくものであるが、参考までにコードサイズに大きく影響する MixColumns の実装方法を一番右の欄に示した。ここで、M4 は MixColumns の行列乗算 4 つを独立にコードとして持っているもの、M1 は行列乗算 1 つだけのコードを持ち、各ラウンド M1 を 4 回ループさせて MixColumns を実行するもの、また MQ は行列の 1 行分だけの演算をするコードを持ち、この二重ループ合計 16 回により 1 ラウンド分の MixColumns 演算をするものである。

ROM 最小化に関しては、文献 [31] で示されている実装よりもさらに小型化されており、暗号化で ROM サイズ 430 バイトは現在知られている最も小さい実装であると思われる。一方、AES の暗号化・復号を実装する場合に ROM サイズ 1024 バイトというのは大きな制約であり、内部でループを多用することによる性能低下が避けられない。実際、ATtiny での既存実装がほとんど ROM サイズ 1500 バイト以上であることも、このような事情が背景にある。高速化に関しては、ROM サイズが 2K バイトあれば、ほぼすべてのループがアンロールできるので、3500 サイクル/ブロック程度が RL78 プロセッサにおける AES の最高性能であると考えられる。

表 3.10　RL78 での AES の実装結果

Algorithm	Category	ROM	RAM static+stack	Enc Speed	Dec Speed	Method
AES (E)	ROM-Min	430	66 + 14	8,753n	–	MQ
AES (E)	(512,64)	510	48 + 10	5,302n	–	M1
AES (E)	(1024,64)	926	48 + 8	3,554n	–	M4
AES (ED)	(1024,64)	1,020	48 + 14	8,193n	9,719n	M1
AES (ED)	(1024,128)	1,020	66 + 14	6,946n	1,380+8,490n	M1
AES (ED)	Fast	2,044	50 + 10	3,554n	753+5,527n	M4

■Camellia

表 3.11 に、RL78 にて Camellia を実装した結果を示す。Camellia には 128 ビットの回転シフト演算が多数含まれているが、その回転数に規則性がないため、ROM サイズが大きくなる。FL 関数や定数 Σ のサイズも小さくなく、S-box を 1 個だけもった場合でも最小 ROM サイズは 749 バイトであった。また、暗号化と復号両方を実装する場合には、コードサイズを減少させるためにサブルーチン化が必要となり、結果として利用するスタックが増加するため RAM サイズ 64 バイトでは実装が困難であった。

一方、ROM サイズが 2K バイトという条件で暗号化だけを行う場合には、AES と同程度の速度が得られている。した

がって、さらに ROM を利用できるならば、復号においては Feistel である Camellia は AES よりも高速となることが期待される。このように Camellia はメモリに比較的余裕がある場合に高性能となる方式である。

これらの実装のうち、ROM-Min 以外で暗号化のみを実装したコードは、必要な回転シフトサブルーチン群を個別に内部で持っているが、暗号化・復号の両方を実装したコードは 8 ビット回転シフトルーチンと 1 ビット回転シフトルーチンだけをコードとしてプログラムの内部に持ち、オンラインで必要なビット数の回転シフトを実現するものである。ROM-Min 実装は 1 ビット回転シフトルーチンだけを内部にコードとして持ち、オンラインで必要なビット数の回転シフトを実現するものである。参考までに、表 3.11 の一番右の欄には、それぞれの実装が何ビットの回転シフトルーチンを独立に持っているかを示した。

表 3.11　RL78 での Camellia の実装結果

Algorithm	Category	ROM	RAM	Enc Speed	Dec Speed	Method
			static+stack			
Camellia (E)	ROM-Min	749	56 + 16	58,382n	–	1
Camellia (E)	(1024,64)	1,024	54 + 10	889+4,709n	–	17,15
Camellia (E)	(1024,128)	1,018	56 + 14	884+4,520n	–	17,15
Camellia (E)	Fast	1,995	66 + 12	740+3,638n	–	34,30,17,15
Camellia (ED)	(1024,128)	1,021	58 + 22	3,034+25,470n	3,907+25,498n	8,1

■CLEFIA

表 3.12 に、RL78 にて CLEFIA を実装した結果を示す。CLEFIA は鍵スケジュール部において、中間鍵を格納するメモリサイズが多いため RAM サイズ 64 バイトで実装することは困難である。また、S-box や MixColumns が 2 つあることに加え、定数が 384 バイトあるなど ROM サイズも大きく、ROM サイズ 512 バイトで実装することは不可能である。

一方、2 つの S-box のうちの 1 つと定数は実行中に動的に生成することも可能であり、ROM-Min 実装と暗号化・復号両方を実装したものについては、これらを実際動的に生成させている。暗号のみの実装で (1024,128) のものは、定数だけを動的に生成させ、S-box は 2 つ ROM に持つ実装を行っている。この結果、ROM 最小実装でも 800 バイト必要であった。参考までに、表 3.12 の一番右の欄にどの実装方法を採用したかを記載している。S は S-box を動的生成、C は定数を動的生成させたことを意味している。

表 3.12　RL78 での CLEFIA の実装結果

Algorithm	Category	ROM	RAM	Enc Speed	Dec Speed	Method
			static+stack			
CLEFIA (E)	ROM-Min	800	58 + 22	23,854n	–	SC
CLEFIA (E)	(1024,128)	1,021	58 + 16	12,351n	–	C
CLEFIA (E)	Fast	1,681	74 + 14	3,010+5,899n	–	
CLEFIA (ED)	(1024,128)	1,018	90 + 26	19,879n	20,797n	SC

■TDES

表 3.13 に、RL78 にて TDES を実装した結果を示す。TDES は RAM サイズは 64 バイトで十分であるものの、不規則なビット演算が中心のアルゴリズムであるため、S-box やビット位置を示す表などだけで 400 バイト以上の ROM を占有する。したがって、ROM サイズ 512 バイトで全体を実装することはできない。

ROM サイズ 1832 バイトの実装は、速度にかかわる部分はほぼアンロールしているので、このプロセッサでのほぼ最高性能に近い速度が出ていると考えられる。この結果から、最高性能で比較すると TDES は AES の 1/15 程度の速度性能と言うことができる。

表 3.13　RL78 での TDES の実装結果

Algorithm	Category	ROM	RAM	Enc Speed	Dec Speed
			static+stack		
TDES (E)	ROM-Min	958	50 + 14	111,183n	–
TDES (E)	(1024,64)	1,024	50 + 14	77,708n	–
TDES (E)	Fast	1,832	50 + 8	26,697n	–
TDES (ED)	(1024,64)	1,019	50 + 14	87,879n	87,543n

■LED

表 3.14 に、RL78 にて LED を実装した結果を示す。LED は 4 ビットを 1 ワードとする AES の構造に近い。このような構造の暗号の場合 8 ビットの平文や鍵をどこかの段階で 2 つの 4 ビットデータに分割する必要がある。これをどの段階で行うか（暗号化前におこなっておくか、実行時に分割するか）でメモリサイズと速度のトレードオフが存在する。また、4 ビット S-box をあらかじめ ROM から RAM に転送しておくと、コードサイズと RAM サイズの増加と引き換えに速度が向上するという別のトレードオフがある。これは、RL78 は ROM データの読み出しに 4 サイクルかかるのに対して、RAM データの読み出しは 1 サイクルで済むからである。さらに、GF(16) 上の 2 倍算を実行時に行うのか RAM に搭載した表で行うかのトレードオフも存在する。

このようなさまざまのトレードオフの中で、それぞれの与えられたメモリサイズ条件に対してどれが最も高速になるかは複雑なパズルである。参考までに、表 3.14 の一番右の欄にどの実装を採用したかを示す記号を示した。ここで、S は S-box を RAM 転送していること、G は GF(16) 上の二倍算のテーブルを RAM 転送していること、T は平文（暗号文）を最初に 4 ビット分割していること、K は鍵を最初に 4 ビット分割していることをそれぞれ示している。

なお、暗号化のみの (1024,128) 実装は、主要な部分をすべてアンロールしているものなので、この実装が RL78 での最高速性能に近いと考えられる。

表 3.14　RL78 での LED の実装結果

Algorithm	Category	ROM	RAM	Enc Speed	Dec Speed	Method
			static+stack			
LED (E)	ROM-Min	298	54 + 12	36,779n	–	T
LED (E)	(512,64)	510	54 + 10	18,055n	–	S
LED (E)	(512,128)	504	100 + 12	17,207n	–	SGTK
LED (E)	(1024,64)	956	54 + 10	15,899n	–	S
LED (E)	(1024,128)	1,023	100 + 8	14,478n	–	SGTK
LED (ED)	(512,64)	508	54 + 10	35,726n	32,219n	T
LED (ED)	(512,128)	508	54 + 14	33,950n	31,787n	T
LED (ED)	(1024,64)	1,007	54 + 10	17,717n	17,788n	S
LED (ED)	(1024,128)	1,023	100 + 8	16,753n	17,472n	SGT

■PRINCE

表 3.15 に、RL78 にて PRINCE を実装した結果を示す。このうち、RAM サイズが 128 バイトの実装はすべて 2 つの S-box の合計 32 バイトを RAM 転送することにより高速化を目指したもので、表 3.15 の一番右に記号 S で示している。また、高速実装のものは S-box2 つを並列化した 256 バイトのテーブルを 2 つ持つことで高速化を目指した実装であり、記号 S8 で示している。

PRINCE は鍵スケジュール処理がほとんどなく、しかも暗号化と復号がほとんど同じ処理で実現できるという特長をもっているが、一方で定数が少なくないことと Matrix 演算のコードのオーバーヘッドから、最小実装の ROM サイズは他の 64 ビット軽量ブロック暗号に比べると大きくなっている。

表 3.15　RL78 での PRINCE の実装結果

Algorithm	Category	ROM	RAM	Enc Speed	Dec Speed	Method
			static+stack			
PRINCE (E)	ROM-Min	424	42 + 22	9,905n	–	
PRINCE (E)	(512,64)	512	42 + 12	7,611n	–	
PRINCE (E)	(512,128)	511	74 + 12	7,320n	–	S
PRINCE (E)	(1024,64)	1,019	42 + 12	4,928n	–	
PRINCE (E)	(1024,128)	1,020	74 + 12	4,541n	–	S
PRINCE (E)	Fast	1,789	42 + 8	3,307n	–	S8
PRINCE (ED)	(512,64)	511	44 + 20	9,925n	10,050n	
PRINCE (ED)	(512,128)	511	76 + 24	9,541n	9,810n	S
PRINCE (ED)	(1024,64)	1,007	42 + 12	5,117n	5,214n	
PRINCE (ED)	(1024,128)	1,017	74 + 12	4,745n	4,832n	S

■PRESENT

表 3.16 に、RL78 にて PRESENT を実装した結果を示す。PRESENT は規則正しい構造を持つため、この構造を利用して極めて小さいコードを作ることが可能である。暗号化での ROM 最小実装は 164 バイトを達成した。この実装は文献 [40] や文献 [31] で示されている実装よりはるかに小さいものである上、速度的にもこれらの結果よりも優れている。

PRESENT の実装方法は、基本的にはいずれも入力レジスタのデータを 1 ビットシフトし、そのキャリービットを出力レジスタに取り込むという簡単な処理の繰り返しである。RL78 の 16 ビット命令を使うことで、このキャリービットの移動が一命令でできることが小型化に貢献している。

PRESENT もテーブルの作り方、またそれを RAM に転送するかどうかでサイズと速度のトレードオフが存在する。表 3.16 の一番右の欄の Sn-m は、16 バイトのテーブルを ROM に n 個持ち、そのうち m 個を RAM に転送する実装であることを意味している。また、S8 はこのテーブル 2 つを並列に参照する 256 バイトのテーブルを 2 つ持つ実装であることを示している。

暗号化における (1024,64) 実装は 1 段を完全にアンロールしているものであり、RL78 における速度の限界を示していると考えられる。このように PRESENT は速度は遅いものの、メモリサイズの小型化で優位性のあるアルゴリズムである。

表 3.16　RL78 での PRESENT の実装結果

Algorithm	Category	ROM	RAM	Enc Speed	Dec Speed	Method
			static+stack			
PRESENT (E)	ROM-Min	164	38 + 22	93,412n	–	S1-0
PRESENT (E)	(512,64)	491	44 + 16	11,344n	–	S2-1
PRESENT (E)	(512,128)	499	60 + 16	10,560n	–	S2-2
PRESENT (E)	(1024,64)	952	28 + 10	9,007n	–	S8
PRESENT (ED)	(512,64)	512	42 + 18	16,924n	3,736+19,131n	S2-0
PRESENT (ED)	(512,128)	509	74 + 18	16,407n	3,643+18,614n	S2-2
PRESENT (ED)	(1024,64)	989	38 + 18	12,048n	1,996+12,367n	S4-0
PRESENT (ED)	(1024,128)	1,003	102 + 18	10,691n	1,903+11,010n	S4-4

■Piccolo

表 3.17 に、RL78 にて Piccolo を実装した結果を示す。表 3.17 の右端の見方は PRESENT の場合と同様である。Piccolo は RAM メモリの使用が少なく実装できるため、すべてのカテゴリにおいて 64 バイトの RAM メモリがあれば十分である。

最小実装のサイズでは PRESENT に及ばないものの、全体的に Piccolo は高速に実装できるアルゴリズムと言える。

表 3.17　RL78 での Piccolo の実装結果

Algorithm	Category	ROM	RAM	Enc Speed	Dec Speed	Method
			static+stack			
Piccolo (E)	ROM-Min	275	24 + 18	12,220n	–	S1-0
Piccolo (E)	(512,64)	498	52 + 8	5,779n	–	S2-2
Piccolo (E)	(1024,64)	1,018	40 + 8	4,961n	–	S8
Piccolo (E)	Fast	1,172	40 + 8	4,636n	–	S8
Piccolo (ED)	(512,64)	512	54 + 8	6,186n	6,084n	S2-2
Piccolo (ED)	(1024,64)	966	52 + 8	5,779n	5,779n	S2-2

■TWINE

表 3.18 に、RL78 にて TWINE を実装した結果を示す。一番右の欄の見方は PRESENT、Piccolo と同様である。TWINE はソフトウエアでオーバーヘッドが少なく、きわめて小型化が可能なアルゴリズムである。また、Piccolo と同じく RAM メモリの使用が少なく実装できるため、すべてのカテゴリにおいて 64 バイトの RAM メモリがあれば十分である。

暗号化のみの場合、ROM サイズが 512 バイト、RAM サイズが 64 バイトですでにほぼ最高速の実装が可能となっている。速度的には、TWINE は Piccolo とほぼ同程度を達成している。

表 3.18　RL78 での TWINE の実装結果

Algorithm	Category	ROM	RAM	Enc Speed	Dec Speed	Method
			static+stack			
TWINE (E)	ROM-Min	232	52 + 8	11,043n	–	S1-0
TWINE (E)	(512,64)	468	52 + 6	4,957n	–	S1-1
TWINE (ED)	(512,64)	510	54 + 10	6,132n	2,463+5,570n	S1-1
TWINE (ED)	(1024,64)	972	54 + 6	4,957n	1,727+4,892n	S1-1

■SIMON

表 3.19 に、RL78 にて SIMON を実装した結果を示す。小型実装は、共通部分のループ化やサブルーチン化を積極的に行って、できる限り ROM サイズを小さくしたものである。SIMON は Feistel 構造であるため、暗号化モジュールと復号モジュールを両方持つプログラムにおいては、データ暗号化部の暗号/復号共用を行っている。表 3.19 の Category で One と表記する一段実装は、ラウンド関数 1 つの内部をアンロールし、これをラウンド回ループさせた実装を行ったものある。データ暗号化部と鍵スケジュール部の共用は行っていないが、暗号化モジュールと復号モジュールを両方持つプログラムにおいては、初期化と鍵スケジュール部はサブルーチン化して暗号化と復号で共用している。

高速実装は、複数ラウンドをまとめてアンロールし、これを必要回数ループさせる実装を行うとともに、一切の共用化やサブルーチン化を抑制することにより、さらなる高速化を目指したものである。高速実装においてまとめてアンロールする最適なラウンド数は、m = (key size)/(word size) とするとき、SIMON の場合 LCM$(2, m)$、である。これは、SIMON がデータ暗号化部は 2 ラウンド周期、鍵スケジュール部が m ラウンド周期であるためである。

■SPECK

表 3.20 に、RL78 にて SPECK を実装した結果を示す。小型実装は、共通部分のループ化やサブルーチン化を積極的に行って、できる限り ROM サイズを小さくしたものであり、データ暗号化部と鍵スケジュール部の共用を行っている。一段実装は、ラウンド関数 1 つの内部をアンロールし、これをラウンド回ループさせた実装を行ったものある。データ暗号化部と鍵スケジュール部の共用は行っていないが、暗号化モジュールと復号モジュールを両方持つプログラムにおいては、初期化と鍵スケジュール部はサブルーチン化して暗号化と復号で共用している。高速実装は、複数ラウンドをまとめてアンロールし、これを必要回数ループさせる実装を行うとともに、一切の共用化やサブルーチン化を抑制することにより、さらなる

表 3.19　RL78 での SIMON の実装結果

Algorithm	Category	ROM	RAM	Enc Speed	Dec Speed
			static+stack		
SIMON(32/64)(E)	ROM-Min	127	20 + 8	3,706n	–
SIMON(32/64)(E)	One	171	20 + 6	2,480n	–
SIMON(32/64)(E)	Fast	413	20 + 6	1,872n	–
SIMON(64/96)(E)	ROM-Min	112	32 + 8	7,354n	–
SIMON(64/96)(E)	One	243	32 + 6	4,598n	–
SIMON(64/96)(E)	Fast	859	32 + 6	3,450n	–
SIMON(64/128)(E)	ROM-Min	128	40 + 8	9,094n	–
SIMON(64/128)(E)	One	303	40 + 6	6,404n	–
SIMON(64/128)(E)	Fast	753	40 + 6	4,688n	–
SIMON(128/128)(E)	ROM-Min	111	48 + 8	21,050n	–
SIMON(128/128)(E)	One	415	48 + 6	13,148n	–
SIMON(128/128)(E)	Fast	629	48 + 6	10,836n	–
SIMON(32/64)(ED)	ROM-Min	273	20 + 14	4,227n	6,586n
SIMON(32/64)(ED)	One	310	30 + 10	2,777n	4,473n
SIMON(32/64)(ED)	Fast	1,035	20 + 6	1,872n	3,069n
SIMON(64/96)(ED)	ROM-Min	244	32 + 14	8,035n	12,063n
SIMON(64/96)(ED)	One	436	32 + 10	4,985n	7,559n
SIMON(64/96)(ED)	Fast	1,888	32 + 6	3,450n	5,217n
SIMON(64/128)(ED)	ROM-Min	277	40 + 14	9,807n	15,408n
SIMON(64/128)(ED)	One	546	40 + 10	6,809n	11,057n
SIMON(64/128)(ED)	Fast	1,883	40 + 6	4,688n	7,551n
SIMON(128/128)(ED)	ROM-Min	203	48 + 14	22,147n	34,005n
SIMON(128/128)(ED)	One	506	48 + 10	13,767n	21,023n
SIMON(128/128)(ED)	Fast	1,457	48 + 6	10,836n	16,116n

高速化を目指したものである。高速実装においてまとめてアンロールする最適なラウンド数は、$m = (\text{key size})/(\text{word size})$ とするとき、SPECK の場合 $m-1$ である。である。これは SPECK がデータ暗号化部に周期はなく、鍵スケジュール部が $m-1$ ラウンド周期であることから導き出される。

■Midori

表 3.21 に、RL78 にて Midori を実装した結果を示す。Midori64 については 4 ビットの S-box で構成されるため、実装方法は PRESENT などと同様の方針をとる。したがって、一番右の欄の見方は PRESENT、Piccolo、TWINE と同様である。Midori128 について、速度優先の実装については 8 ビット S-Box のテーブル実装とループ展開を行う。また、小型実装においては、Midori128 における 8 ビット S-Box の 4 ビット S-Box から計算処理による作成、関数共通化と関数呼び出しの多用、ループ処理化を行っている。

表 3.20　RL78 での SPECK の実装結果

Algorithm	Category	ROM	RAM static+stack	Enc Speed	Dec Speed
SPECK(32/64)(E)	ROM-Min	96	24 + 8	1,817n	–
SPECK(32/64)(E)	One	115	20 + 6	1,249n	–
SPECK(32/64)(E)	ROM-Min	261	20 + 6	1,006n	–
SPECK(64/96)(E)	ROM-Min	90	44 + 8	6,645n	–
SPECK(64/96)(E)	One	185	32 + 6	2,335n	–
SPECK(64/96)(E)	Fast	308	32 + 6	2,062n	–
SPECK(64/128)(E)	ROM-Min	89	52 + 8	7,448n	–
SPECK(64/128)(E)	One	205	40 + 6	2,644n	–
SPECK(64/128)(E)	Fast	451	40 + 6	2,122n	–
SPECK(128/128)(E)	ROM-Min	71	67 + 8	11,432n	–
SPECK(128/128)(E)	One	205	64 + 6	5,662n	–
SPECK(128/128)(E)	Fast	309	48 + 6	4,793n	–
SPECK(32/64)(ED)	ROM-Min	211	24 + 10	2,308n	3,684n
SPECK(32/64)(ED)	One	283	20 + 6	1,249n	1,918n
SPECK(32/64)(ED)	Fast	623	20 + 6	1,006n	1,392n
SPECK(64/96)(ED)	ROM-Min	211	44 + 10	6,600n	10,837n
SPECK(64/96)(ED)	One	447	32 + 6	2,335n	3,585n
SPECK(64/96)(ED)	Fast	742	32 + 6	2,062n	3,088n
SPECK(64/128)(ED)	ROM-Min	210	52 + 10	7,078n	11,690n
SPECK(64/128)(ED)	One	499	40 + 6	2,644n	4,152n
SPECK(64/128)(ED)	Fast	1,087	40 + 6	2,122n	3,165n
SPECK(128/128)(ED)	ROM-Min	157	67 + 10	11,471n	18,074n
SPECK(128/128)(ED)	One	391	64 + 10	5,702n	8,726n
SPECK(128/128)(ED)	Fast	746	48 + 6	4,793n	7,343n

表 3.21　RL78 での Midori の実装結果

Algorithm	Category	ROM	RAM static+stack	Enc Speed	Dec Speed	Method
Midori64 (E)	Fast	871	64 + 8	6,768n	–	S2-0
Midori64 (E)	ROM-Min	232	96 + 8	16,979n	–	S2-0
Midori64 (ED)	Fast	1,576	64 + 10	6,768n	8,360n	S2-0
Midori64 (ED)	ROM-Min	374	96 + 6	17,867n	27,966n	S2-0
Midori128 (E)	Fast	1,346	64 + 8	9,217n	–	–
Midori128 (E)	ROM-Min	560	64 + 8	31,794n	–	–
Midori128 (ED)	Fast	1,745	64 + 10	9,217n	10,166n	–
Midori128 (ED)	ROM-Min	605	64 + 6	32,495n	45,586n	–

3.1.2.2 性能比較

以下、これまでの実装結果をもとに、評価対象アルゴリズムをいくつかの軸で比較する。

■メモリサイズを限定した実装（暗号化のみ）

図 3.27 は ROM サイズ 1024 バイト以下、RAM サイズ 128 バイト以下の条件で、暗号化のみを実装した場合の速度性能の比較である。図 3.28 は、見やすさのため図 3.27 から TDES を除いたものである。この程度のメモリリソースがある場合には AES が最も高速となり、SPECK がそれに続くとの結果を得た。

図 3.29 は ROM サイズ 1024 バイト以下、RAM サイズ 64 バイト以下の条件で、暗号化のみを実装した場合の速度性能の比較である。図 3.30 は、見やすさのため図 3.29 から TDES を除いたものである。CLEFIA を除けば、ROM サイズ 1024 バイト以下、RAM サイズ 128 バイト以下の場合と同様であるが、CLEFIA だけは RAM64 バイトでの実装が不可能である。これを図では値 0 として示している。

図 3.31 は ROM サイズ 512 バイト以下、RAM サイズ 128 バイト以下の条件で、暗号化のみを実装した場合の速度性能の比較、図 3.32 は ROM サイズ 512 バイト以下、RAM サイズ 64 バイト以下の条件で暗号化のみ実装した場合の速度性能の比較である。ROM サイズが 512 バイト以下になると CLEFIA 以外にも Camellia や TDES も実装が不可能となる。その他のアルゴリズムでは AES、SPECK が依然高速である。

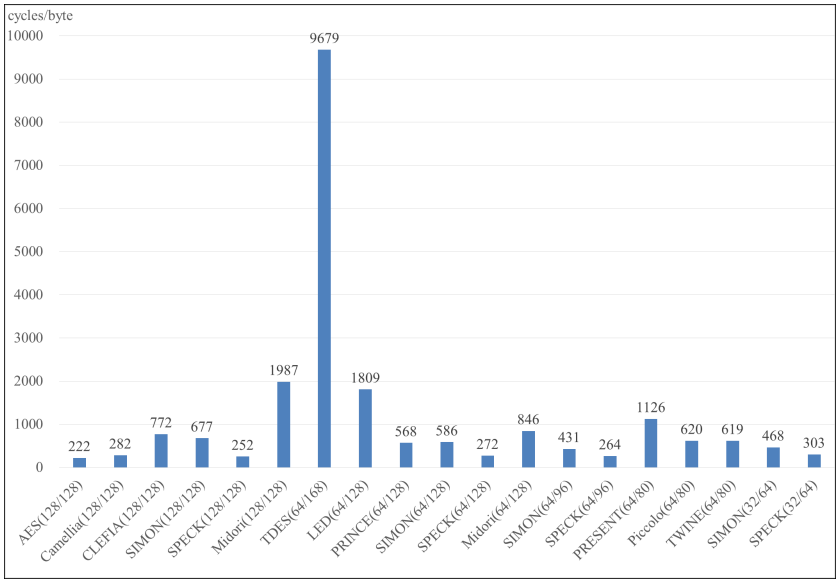

図 3.27　ROM 1024 バイト、RAM 128 バイトでの速度性能

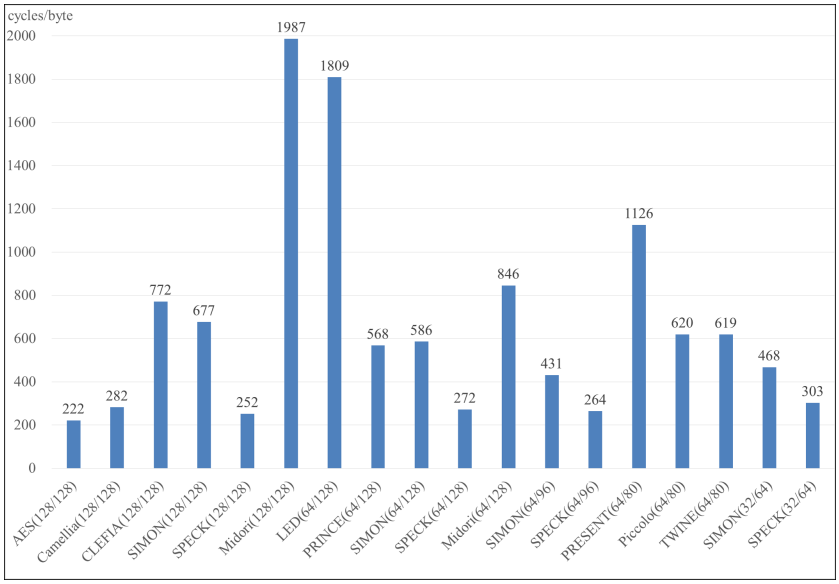

図 3.28　ROM 1024 バイト、RAM 128 バイトでの速度性能（TDES を除いた図）

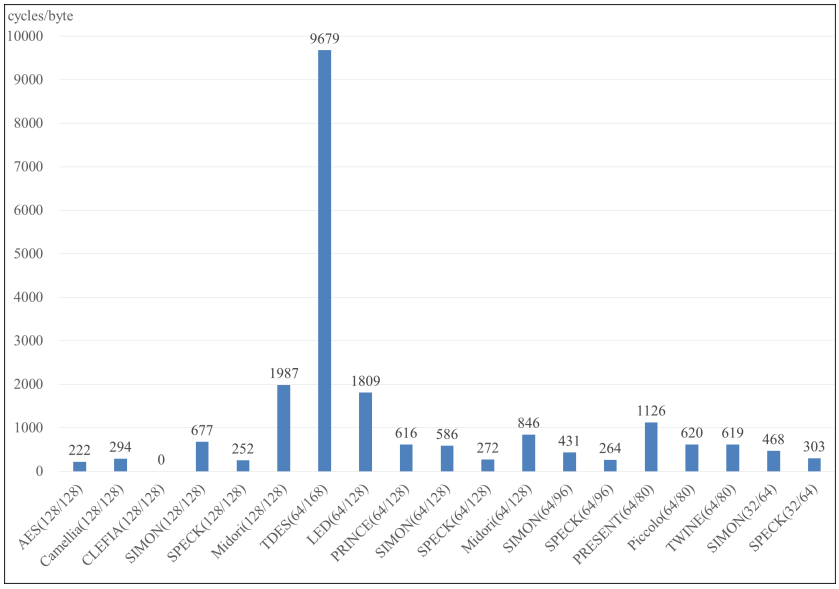

図 3.29　ROM 1024 バイト、RAM 64 バイトでの速度性能

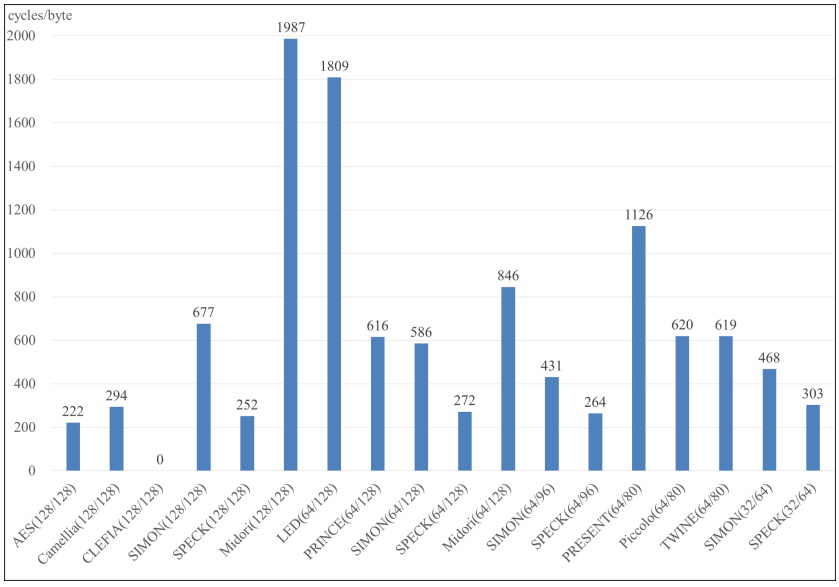

図 3.30　ROM 1024 バイト、RAM 64 バイトでの速度性能（TDES を除いた図）

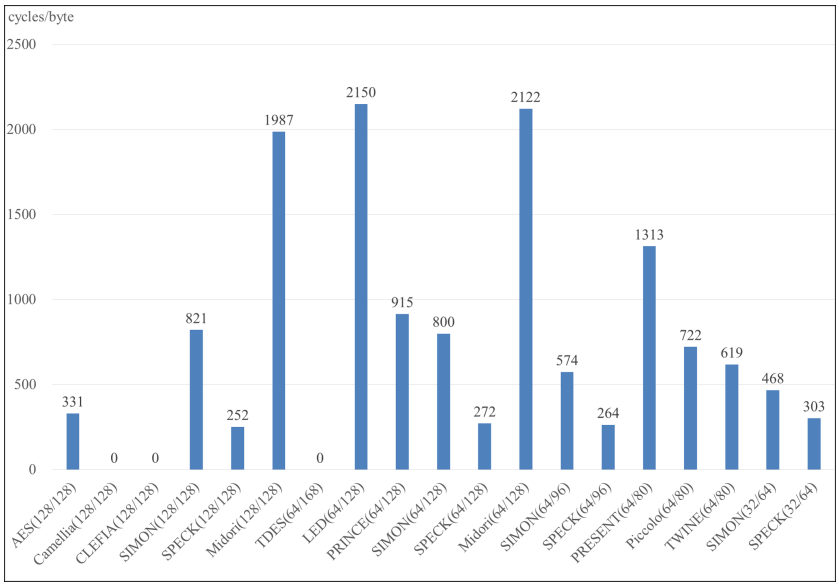

図 3.31　ROM 512 バイト、RAM 128 バイトでの速度性能

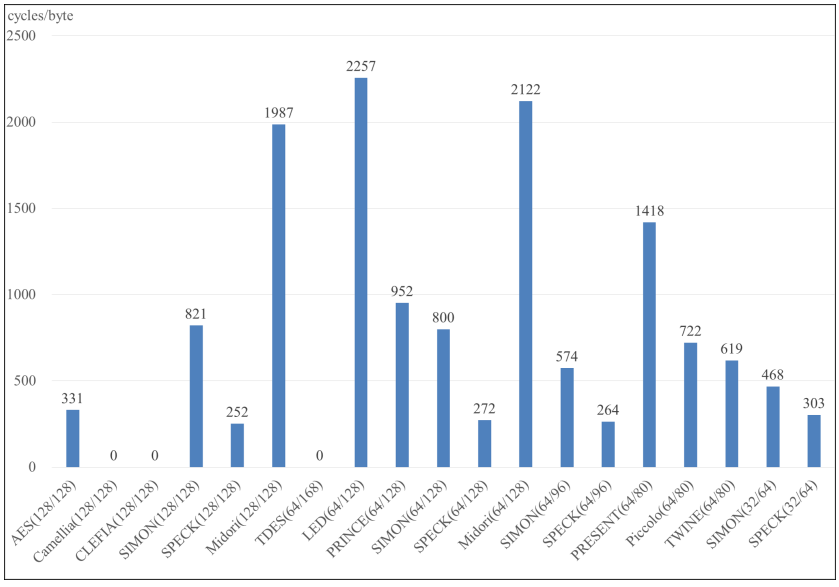

図 3.32　ROM 512 バイト、RAM 64 バイトでの速度性能

■メモリサイズを限定した実装（暗号化・復号）

図 3.33 は ROM サイズ 1024 バイト以下、RAM サイズ 128 バイト以下の条件で、暗号化・復号両方を実装した場合の速度性能の比較である。図 3.34 は、見やすさのため図 3.33 から TDES を除いたものである。ここでも SPECK が最高速であるが、暗号化のみの実装の場合と比べて Piccolo、PRINCE、TWINE との差は縮まっている。

図 3.35 は ROM サイズ 1024 バイト以下、RAM サイズ 64 バイト以下の条件で、暗号化・復号両方を実装した場合の速度性能の比較である。図 3.36 は、見やすさのため図 3.35 から TDES を除いたものである。このカテゴリでは CLEFIA と Camellia、Midori128 が実装不可能となった。また、AES の速度と Piccolo、PRINCE、TWINE との差があまりなくなっている。これは AES の速度がメモリリソース不足のため低下していることを示している。ここでも SPECK が最高速である。

図 3.37 は ROM サイズ 512 バイト以下、RAM サイズ 128 バイト以下の条件で暗号化・復号両方を実装した場合の速度性能の比較、図 3.38 は ROM サイズ 512 バイト以下、RAM サイズ 64 バイト以下の条件で暗号化・復号両方を実装した場合の速度性能の比較である。ここでは AES も実装不可能となり、結果的に 5 つのアルゴリズムだけが生き残るという結果となった。なかでも SPECK は ROM サイズの制限によってほとんど影響を受けない高速性能を示している。

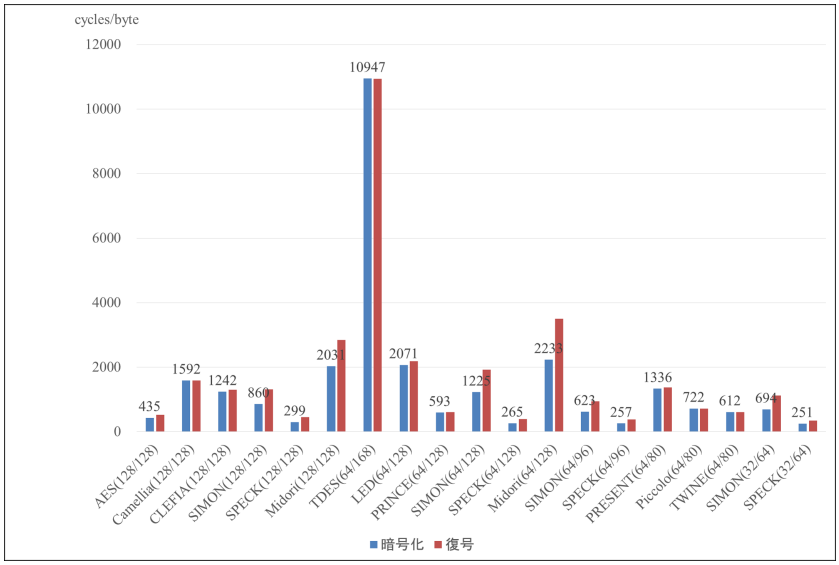

図 3.33　ROM 1024 バイト、RAM 128 バイトでの速度性能

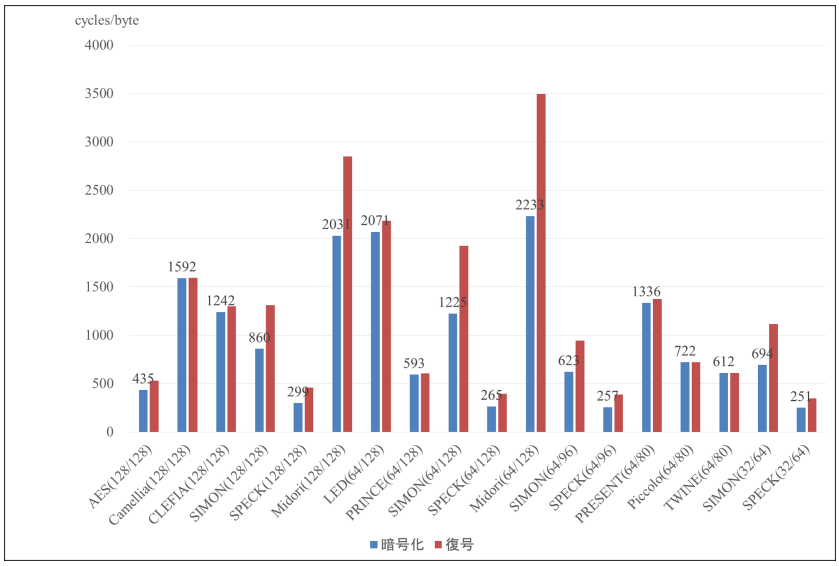

図 3.34　ROM 1024 バイト、RAM 128 バイトでの速度性能（TDES を除いた図）

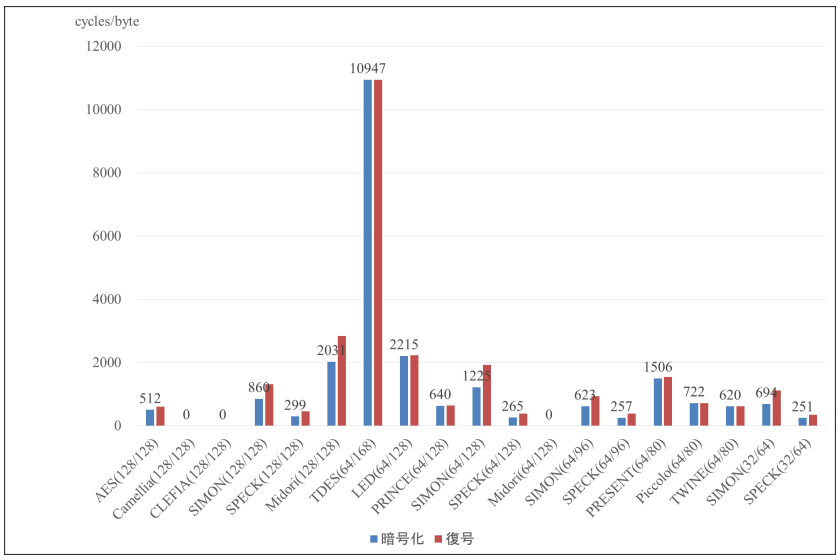

図 3.35　ROM 1024 バイト、RAM 64 バイトでの速度性能

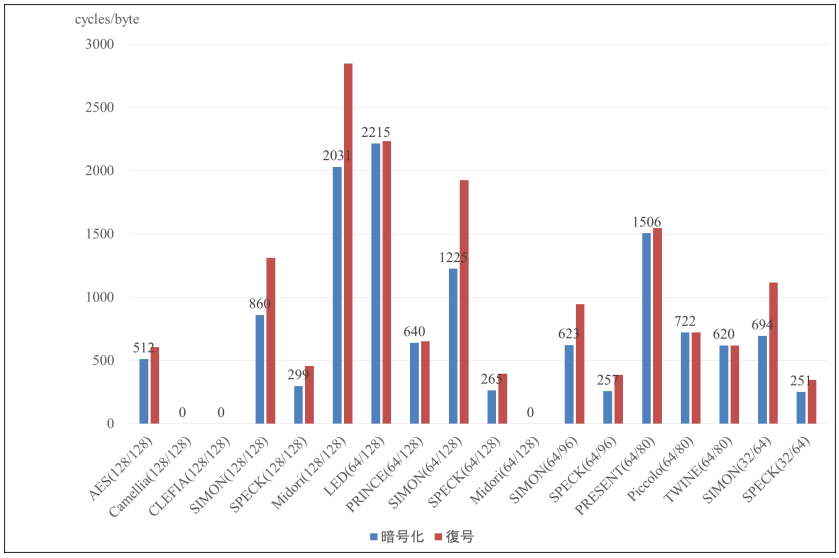

図 3.36　ROM 1024 バイト、RAM 64 バイトでの速度性能（TDES を除いた図）

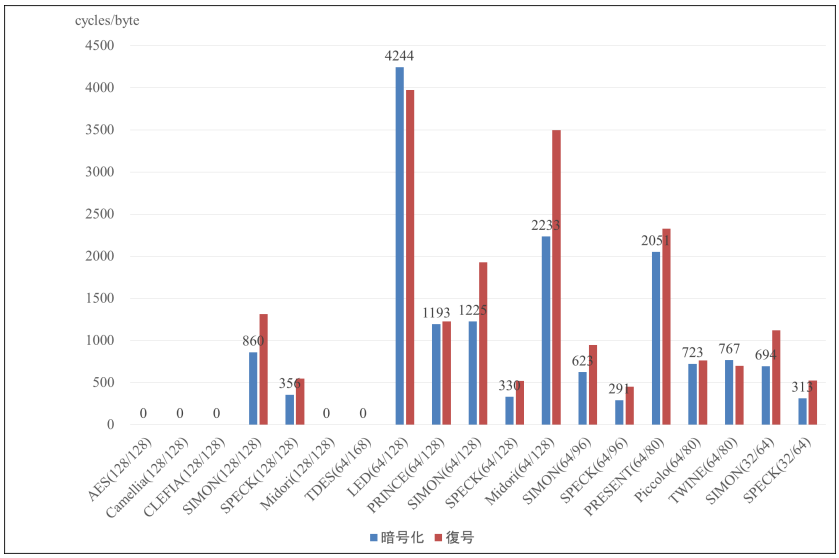

図 3.37　ROM 512 バイト、RAM 128 バイトでの速度性能

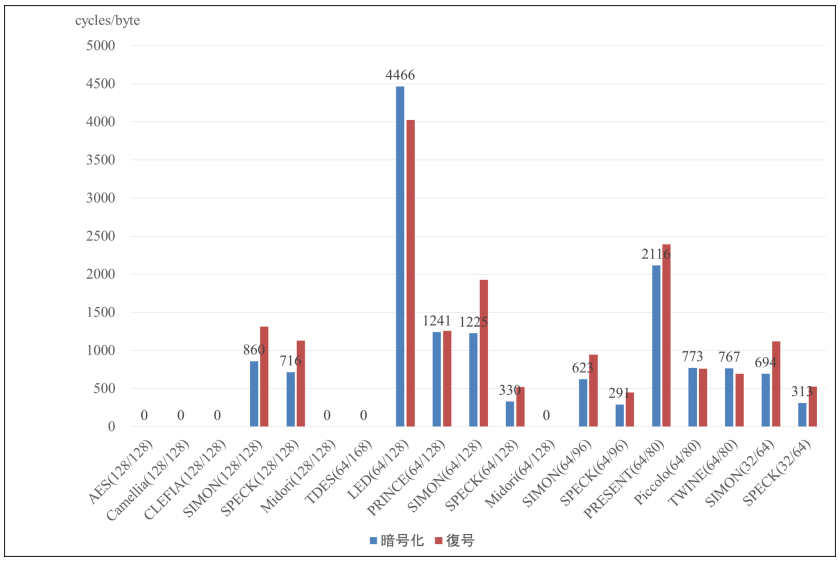

図 3.38　ROM 512 バイト、RAM 64 バイトでの速度性能

■メモリサイズを限定した実装（まとめ）

以上の結果を一つの図にまとめたものを図 3.39 ならびに図 3.40 に示す。後者は前者から TDES を除いたものである。一般的にメモリサイズの制約が厳しくなる右側に行けばいくほど性能は低下する。この性能低下があまり見られない SIMON、SPECK、Piccolo 及び TWINE はメモリサイズにまだ余裕があることを示している。

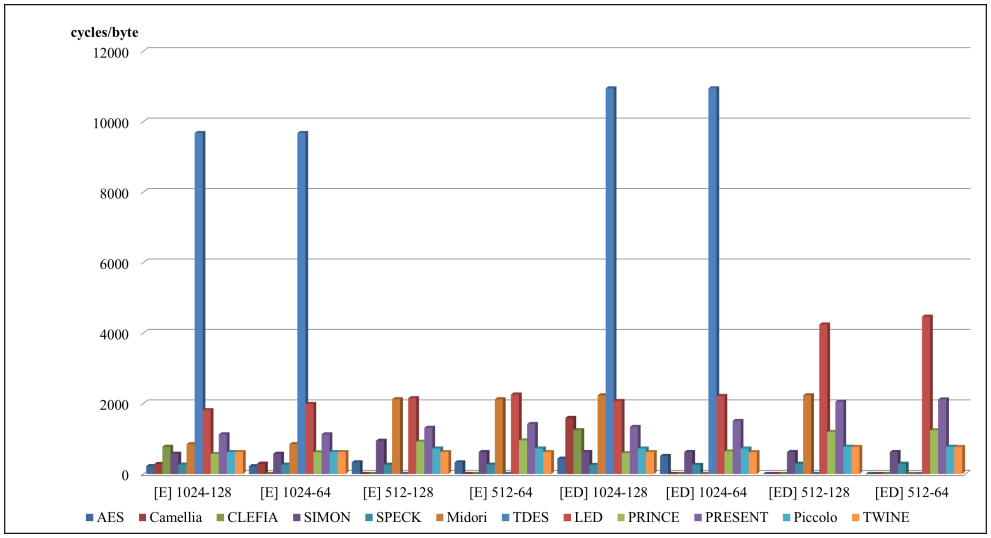

図 3.39　メモリサイズ限定速度性能一覧

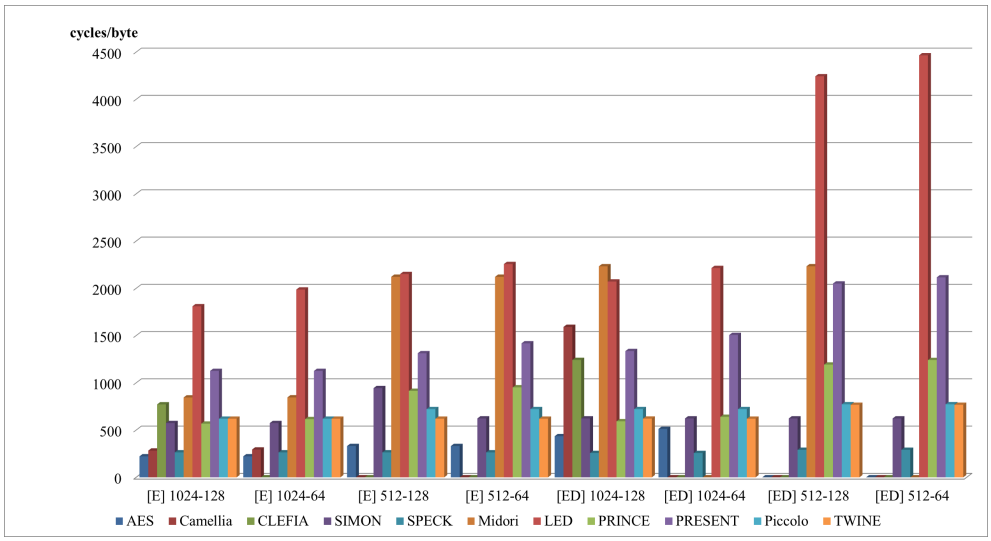

図 3.40　メモリサイズ限定速度性能一覧（TDES を除いた図）

■高速実装

図 3.41 は ROM 2KB 程度で高速実装を目指した場合の速度性能の比較である。これは RL78 プロセッサで達成できる各アルゴリズムの最高性能に近い値と考えられる。この評価では AES、Camellia、SPECK がほぼ同等の性能となることが知られる。

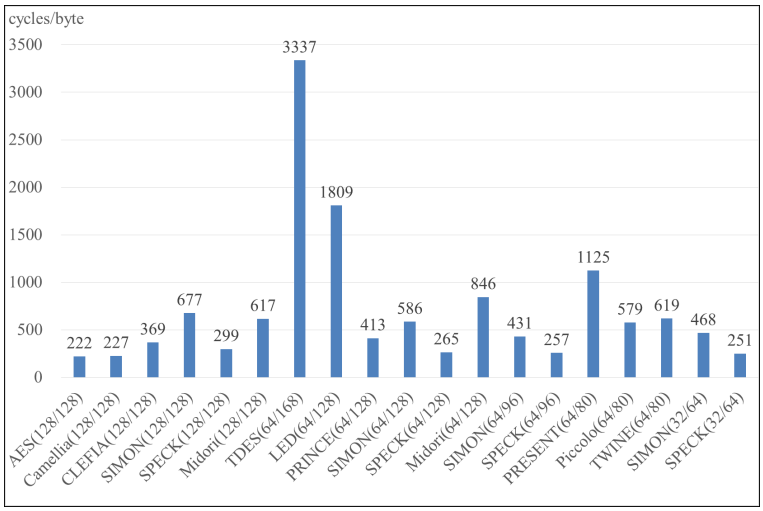

図 3.41　高速実装速度性能一覧

■最小実装

図 3.42 は ROM サイズを最小にする実装（暗号化のみ）において、そのサイズがどこまで小さくなるかを評価したものである。ここでは、最近の軽量ブロック暗号アルゴリズムと、それ以外のものの差がはっきりあらわれている。

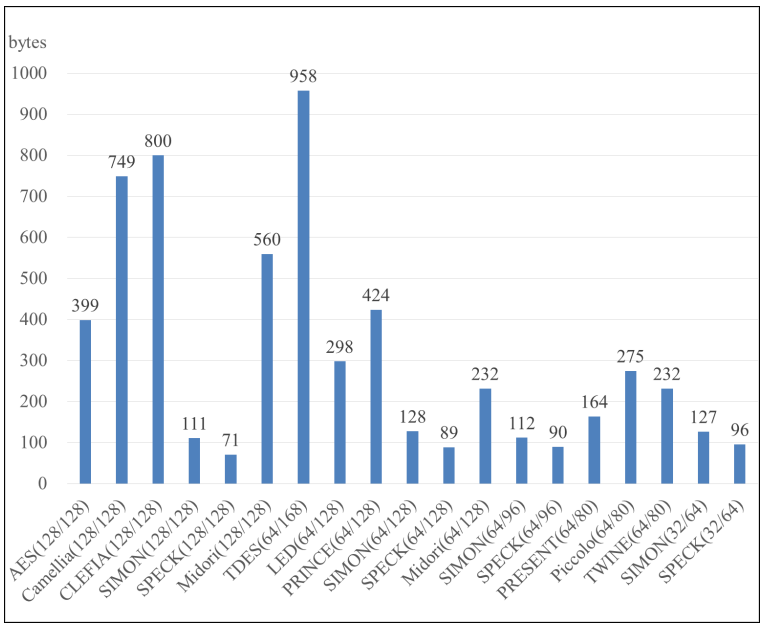

図 3.42　最小実装速度性能一覧

■その他の考察

図 3.43 は暗号化のみを実装したもののうち、ROM サイズが 512 バイト以下になるものについてすべてをプロットしたものである。ここで横軸は ROM サイズ、縦軸は速度を示している。AES については S-box テーブルを持つ限り ROM サイズ 400 バイト前後が限界であり、それ以下の領域は、より小さい S-box を持つアルゴリズムあるいは S-box を持たないアルゴリズムで可能となる。また、この図が示すように、ROM サイズが 200 バイト以下で 2000 サイクル/バイト以下は、SIMON、SPECK だけが達成できる域であり、今後のソフトウエア軽量暗号が目指す一つの方向性を示していると思われる。

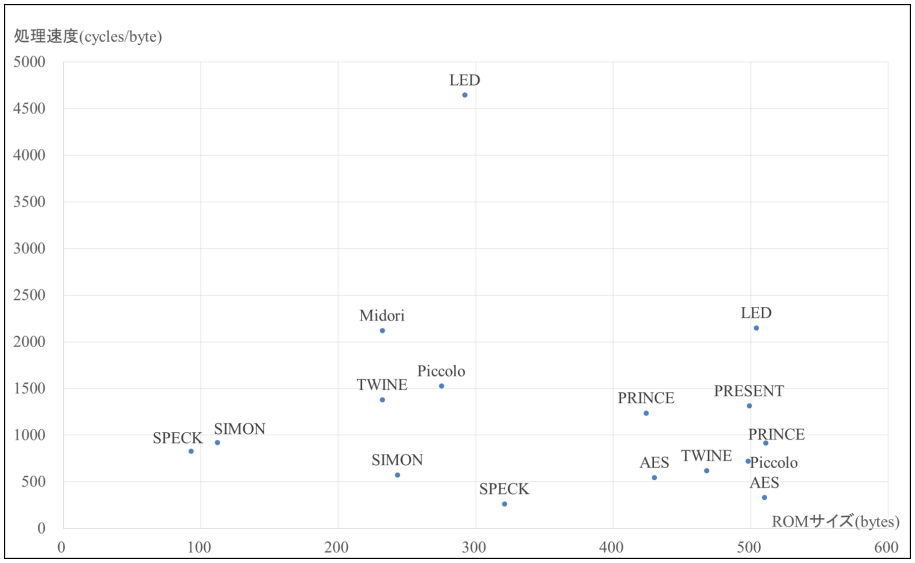

図 3.43　サイズと速度のトレードオフ

一方、今回の実装をアルゴリズム構造の観点から見ると、小型化するにはアルゴリズム全体が数少ない単純な繰り返し構造をもっている必要がある。小型化実装を行う場合、単なるデータの移動や定数も無視できないオーバーヘッドとなる。また、RAM サイズの制約がある場合 on-the-fly で鍵スケジュールを行う必要があるが、アルゴリズムによってはこの鍵スケジュールが小型化のボトルネックになることも少なくない。

さらに、プロセッサ構造の観点から見ると、回転シフト命令の効率はプロセッサに大きく依存すること、最近のプロセッサはほとんどが little endian なので big endian を前提としたアルゴリズムはバイト順の変更に伴うオーバーヘッドが発生する可能性があることなどは、軽量暗号設計において留意すべき点であると言える。

3.1.2.3 評価方法の概要

本節では、軽量ブロック暗号のソフトウェア実装性能評価において採用した評価環境と実装条件について述べる。

■組み込みマイコン RL78 と評価環境

本評価では、ルネサスエレクトロニクス社製の組み込みマイコン RL78 上で評価対象ブロック暗号の実装を行った [10]。RL78 はアキュムレータベースの 16 ビット CISC プロセッサである。その命令セットには多くの 1 バイト命令が存在し、また Load-Modify 命令がサポートされているなど、ROM サイズの小型化に適したプロセッサであると言える。一方、RL78 はすべての命令で 16 ビットデータを扱えるわけではなく、例えばブロック暗号で頻繁に必要となる論理演算や回転シフト演算は 8 ビット命令だけがサポートされている。

RL78 にはいくつかのシリーズが存在するが、例えばそのうち汎用である G1x シリーズのローエンド製品は ROM サイズが 1K バイト、RAM サイズは 128 バイトである。また、車載用に開発された F1x シリーズのローエンド製品は ROM サイズ 8K バイト、RAM サイズ 512 バイトを搭載している。

RL78 の命令セットはシリーズ共通であり（ただし乗算命令のサポートの有無は品種により異なる）、その命令長も同じである。したがって、乗算命令以外の汎用命令を使用する限り、RL78 のコードはすべての品種で動作し、またそのコードが占有するメモリサイズは同じである。実行速度については一部採用するハードウェアコア（S1、S2、S3 の 3 種類が存在する [10]）によって若干異なるが、ここではほとんどの品種で利用されている S2 コアでの速度を計測した。

なお、開発環境はルネサスエレクトロニクス社製の CubeSuite+ を使用した。

■実装条件

さまざまなメモリリソース環境での性能を評価するため、ROM サイズは 512 バイトと 1024 バイトの 2 種類、RAM サイズは 64 バイトと 128 バイトの 2 種類の合計 4 通りのメモリ制約条件のもので、暗号化機能だけを実装した場合と、暗号化機能と復号機能の両方を実装した場合の速度性能を調べた。さらにこれに加え、速度性能は考慮せずに ROM サイズを最小化する実装や、逆に ROM サイズを 2K バイト程度まで許した場合に速度がどこまで向上するかなどの観点でも評価を行った。

評価対象とするブロック暗号アルゴリズムによっては、特定のメモリ制約条件ではそもそも実装が不可能なものや、逆に少ないリソースで充分な性能が出ており、それ以上のメモリ容量が与えられてもそれ以上の性能向上が認められないものも存在する。そのような場合は個別の実装を省略した。なお、RAM サイズが最大 128 バイトでの評価であるので、結果的にすべての評価対象アルゴリズムで、鍵スケジュールは on-the-fly 実装になっている。

本ライブラリのプログラムインターフェースや ROM、RAM サイズの計算方法は文献 [31] と同様である。すなわち、プログラムはアセンブリ言語で記述し、1 ブロックのデータを暗号化および復号する機能を持つ。C 言語から呼び出し可能なサブルーチン形式とし、このサブルーチンは引数を一つだけ取り、その引数が指すアドレスに、平文、暗号文、鍵、一時利用データを格納する。また、RAM メモリ最小化ならびに実用的な観点から、以下の条件での実装を行っている。

- 平文と暗号文の領域は共用する。
- 鍵の領域はサブルーチン呼び出し時と終了時で同じ内容とする（ただし実行中に一時的に変更してもよい）。
- システムが主に利用するゼロページ領域（絶対アドレス 256 未満の領域）は利用しない。
- リロケータブルなプログラムとする（絶対アドレスをハードコーディングしない）。
- システムに依存するコーディングは行わない（例えばレジスタバンクの切り替えをしない、特殊レジスタを直接操作することはしないなど）。

ROM、RAM サイズの計算には、このサブルーチンを実行するために必要なすべてのリソースを含めている。具体的には、ROM サイズはコードならびに固定データテーブルを含み、RAM サイズは、平文（暗号文と領域を共用する）・鍵・一時データ・スタックをすべて含んでいる。したがって、例えば 128 ビットブロック、128 ビット鍵のブロック暗号においては、平文と鍵の領域だけで 32 バイトを占有する。また、関数呼び出し時に必要となるスタックフレームは 6 バイト（call 命令 4 バイト +callee save レジスタの保存 2 バイト）であるので、例えば RAM サイズ 64 バイト以下で実装する場合、自由に使える RAM はスタックを含めて 26 バイトしか存在せず、かなり厳しい制約となる。

3.2 認証暗号の実装性能

3.2.1 ソフトウェア実装評価

本節では、組み込みマイコン上の限定されたメモリリソースのもとで、認証暗号をソフトウェア実装した場合の速度性能について計測、比較した結果を示す。評価対象とした認証暗号を図 3.44 に示す。

	セキュリティレベル	鍵長	ブロック長	ナンス長	タグ長
CLOC-TWINE	32	80	64	48	32
SILC-PRESENT	32	80	64	48	32
JAMBU-SIMON	48	96	96	48	48
CLOC-AES	64	128	128	96	64
SILC-AES	64	128	128	96	64
AES-OTR	64	128	128	96	128
AES-OCB	64	128	128	96	128
JAMBU-AES	64	128	128	96	128
AES-GCM	64	128	128	96	128
ACORN	128	128	128	128	128
Minalpher	128	128	256	128	128
Ketje-SR	128	128	32	128	128
Ascon	128	128	128	128	128

図 3.44　評価対象暗号パラメータ比較

なお、ハードウェア実装については、ジョージメイソン大学の研究チームによって同一プラットフォームによる評価が行われているため、本ガイドラインでは扱わない。詳細は以下の URL を参照されたい。

Authenticated Encryption FPGA Ranking

`https://cryptography.gmu.edu/athenadb/fpga_auth_cipher/rankings_view`

3.2.1.1 性能比較

■認証暗号の実装評価結果

図 3.45 に認証暗号を小型実装した場合の評価結果を示す。図 3.45 における上の表は左から、アルゴリズム名、セキュリティレベル（bit）、ROM サイズ（bytes）、RAM サイズ（bytes）、関連データ処理に関する各関数の処理速度（赤字が cycles/byte、黒字は cycles）、内部で利用されている Core Function 名、Core Function の ROM サイズ（bytes）、Core Functions の速度（cycles/byte）である。図 3.45 における下の表は暗号化処理ならびに復号処理に関する各関数の処理速度を表している。

なお、CLOC-TWINE における ENC_NULL、DEC_NULL の速度はそれぞれ 11160、11099 cycles、CLOC-AES における ENC_NULL、DEC_NULL の速度はそれぞれ 8895、8790 cycles である。

図 3.46 に認証暗号を高速実装した場合の評価結果を示す。図の見方は図 3.45 と同様である。

なお、CLOC-TWINE における ENC_NULL、DEC_NULL の速度はそれぞれ 5074、5013 cycles、CLOC-AES における ENC_NULL、DEC_NULL の速度はそれぞれ 3748、3643 cycles である。

以上は暗号化（とタグ生成）モジュールならびに復号（とタグ検証）モジュールをともに含むプログラムの評価結果であったが、ここから復号部の関数群（DEC_0, DEC_1, DEC_2, DEC_3, DEC_4）を取り除いて暗号化（とタグ生成）機能だけにした場合は、その他の関数の性能は変わらず、ROM サイズだけが表 3.22 で示すバイト数だけ減少する。このサイズは Core Function の種類や小型版、高速版に依存せず一定である（そのように設計されている）。

ここで、Core Function で利用されるブロック暗号の暗号化・復号機能は、認証暗号としての暗号化・復号機能とは異なることに注意されたい。今回の実装評価対象とした認証暗号のうち、Minalpher と AES-OCB 以外はすべて、認証暗号としての復号機能においても Core Function の逆関数（Core Function がブロック暗号の場合はブロック暗号としての復号機能）

	Sec Level	Size		Associate Data Processing					Core Function		
		ROM	RAM	AD_0	AD_1	AD_2	AD_3	AD_4		ROM	Speed
CLOC-TWINE	32	811	108	-	11199	1409		365	TWINE	234	1380
SILC-PRESENT		537	98	93413	11706			312	PRESENT	164	11677
		600	96	38357	4796			312	PRESENT	227	4768
JAMBU-SIMON	48	522	90	12407	1050			12602	Simon (96/96)	109	1030
CLOC-AES	64	963	150	-	8966	570		637	AES	399	544
SILC-AES		772	144	9071	569			544			
AES-OTR		1202	178	142/9638	594		10134	-			
JAMBU-AES		803	128	8885	563		9010	-			
AES-OCB		1705	224	9252	564		8921		AES	1020	512/607
AES-GCM		760	172	8943	2557			-	Mul128	64	2534
ACORN	128	589	129	109075	489			15522	State	327	478
Minalpher		665	193	607/41361	1276		41220	-	P	295	1241
Ketje-SR		724	114	46827	988		3955	-	F	385	969
Ascon		617	132	40332	2491		19919	11	p^6	299	2475

	Plaintext Data Processing					Ciphertext Data Processing				
	ENC_0	ENC_1	ENC_2	ENC_3	ENC_4	DEC_0	DEC_1	DEC_2	DEC_3	DEC_4
CLOC-TWINE	11160	22365	2799	22593	-	11160	22365	2799	22532	64
SILC-PRESENT	93816	187276	23424		93786	93816	187276	23424		93778
	38343	76730	9606		38513	38543	76730	9606		38505
JAMBU-SIMON	-	1057		12619	25048	-	1057		12633	25099
CLOC-AES	9373	18211	1151	18096	-	9373	18211	1151	17991	108
SILC-AES	9443	18210	1151		9353	9443	18210	1151		9345
AES-OTR	9708	616		19839	10166	9708	609		19841	10158
JAMBU-AES	-	573		9035	17952	-	573		9042	17960
AES-OCB	9736	585		8804	8861	9736	680		8804	8885
AES-GCM	182	3124			49823	182	3124			50008
ACORN		490			62053		493			62085
Minalpher	40729	2560		81942	-	40729	2549		81525	196
Ketje-SR	-	992		23298	11724	-	993		23302	11747
Ascon		2503		206	40220		2517		300	40173

図 3.45　認証暗号の実装結果（小型実装）

表 3.22　認証暗号の暗号化モジュールのみのプログラムの ROM 削減バイト数

アルゴリズム名	AES-GCM	CLOC	SILC	AES-OTR	Ketje	Minalpher
削減されるバイト数	47	102	59	255	108	93

は必要ではない。また、Minalpher の Core Function は、その逆関数とは厳密には異なるが、Involution 構造を持つので非常に類似した構造を持っている。

本評価におけるインターフェースの作り方から、認証暗号としての暗号化部の関数群と復号部の関数群は明確に分けられているので、暗号化（とタグ生成）機能だけを持つ認証暗号モジュールのサイズは、単純に復号部の関数群のサイズを全体から引くだけで計算でき、速度性能については変わることはない。

図 3.47 に小型実装を行った場合の ROM サイズを比較したグラフを示す。図 3.48 はこの ROM サイズを、Core Function の ROM サイズ（下部）と Mode に相当する、それ以外の部分の ROM サイズ（上部）に分割したものである。また、図 3.49 に高速実装を行った場合の速度（正確には平文サイズが十分大きい時の漸近的速度）を比較したグラフを示す。図 3.50 はこの速度を、Core Function の速度（下部）と Mode に相当する、それ以外の部分の速度（上部）に分割したものである。

これらのグラフでは、いずれも認証暗号アルゴリズムのセキュリティレベルを（32 ビット、64 ビット、128 ビット）色で表現している。

	Sec Level	Size		Associate Data Processing					Core Function		
		ROM	RAM	AD_0	AD_1	AD_2	AD_3	AD_4		ROM	Speed
CLOC-TWINE	32	1049	106	-	5113	649		365	TWINE	470	620
SILC-PRESENT		896	118	10777	1349			262	PRESENT	499	1320
JAMBU-SIMON	48	1709	80	6342	531		6435	-	Simon (96/96)	477	527
CLOC-AES	64	1521	128	-	3819	248		637	AES	928	222
SILC-AES		1369	124	3924	248			438			
AES-OTR		1958	156	142/4081	239		4163	-			
JAMBU-AES		2466	102	3589	227		3710	-			
AES-OCB		2962	216	3809	238		3837		AES	2007	222/391
AES-GCM		1489	150	3669	1034			-	Mul128	239	1011
ACORN	128	871	116	55296	246			7752	State	446	237
Minalpher		1926	169	49/9580	308		9898	-	P	1455	289
		1977	425	866/8669	254		8170	-	P	1455	235
Ketje-SR		1482	114	15699	340		1361	-	f	927	321
Ascon		1966	116	11173	696		5639	11	p^6	1015	691

	Plaintext Data Processing					Ciphertext Data Processing				
	ENC_0	ENC_1	ENC_2	ENC_3	ENC_4	DEC_0	DEC_1	DEC_2	DEC_3	DEC_4
CLOC-TWINE	5074	10193	1278	10421	-	5074	10193	1278	10307	64
SILC-PRESENT	10657	21485	2689		10883	10657	21485	2689		10875
JAMBU-SIMON	-	532		6455	12699	-	532		6460	12767
CLOC-AES	3728	7585	476	7971	-	3728	7585	476	7866	108
SILC-AES	3694	7751	487		4100	3694	7751	487		4092
AES-OTR	4151	245		8140	4147	4151	243		8142	4139
JAMBU-AES	-	228		3738	7203	-	228		3745	7293
AES-OCB	4923	243		3893	3737	4923	412		3893	3888
AES-GCM	173	1271			20172	173	1271			20357
ACORN		247			31034		250			31082
Minalpher	9713	616		19739	-	9713	602		19554	196
	7985	508		16283	-	7985	494		16098	196
Ketje-SR	-	343		7734	3942	-	344		7738	3965
Ascon		698		188	11184		699		273	11312

図 3.46　認証暗号の実装結果（高速実装）

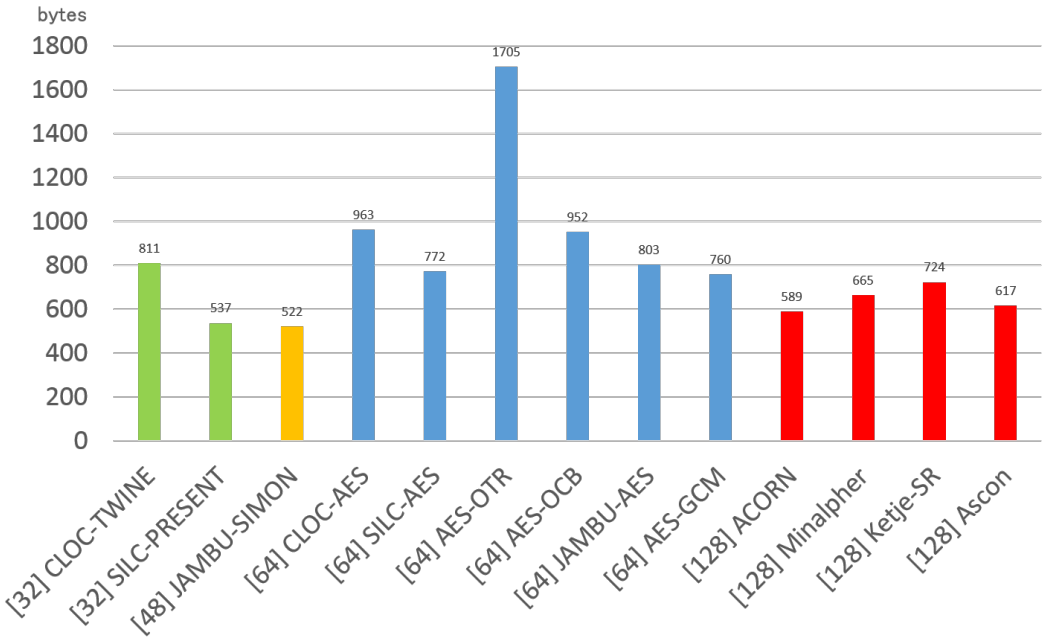

図 3.47　小型実装における ROM サイズ比較

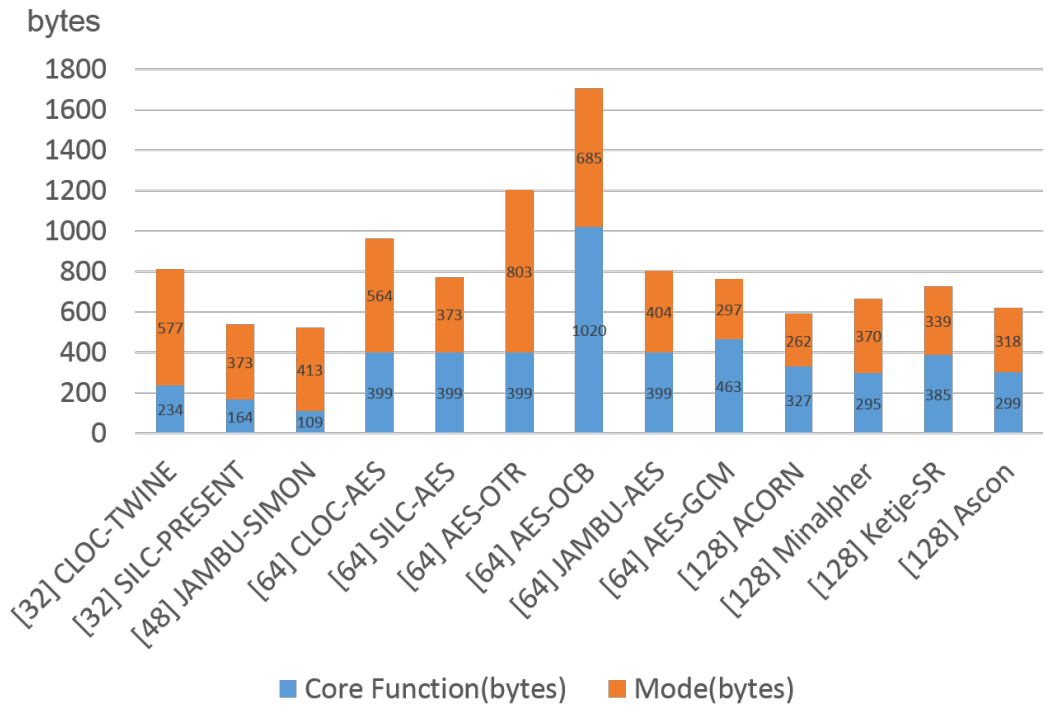

図 3.48　小型実装における ROM サイズ比較（Core Function と Mode に分割）

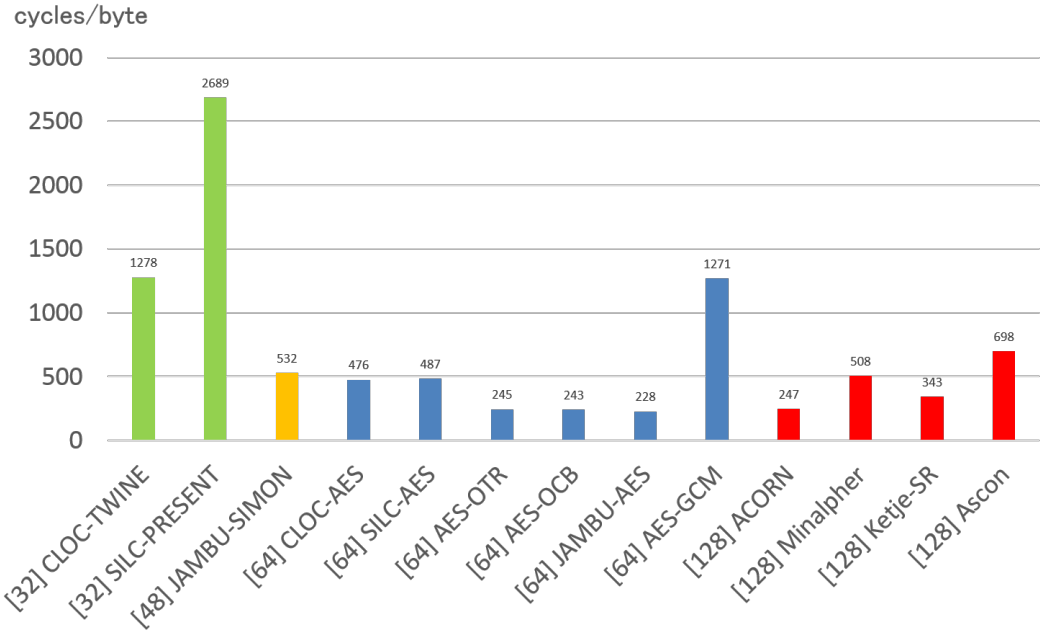

図 3.49　高速実装における暗号化速度（漸近速度）比較

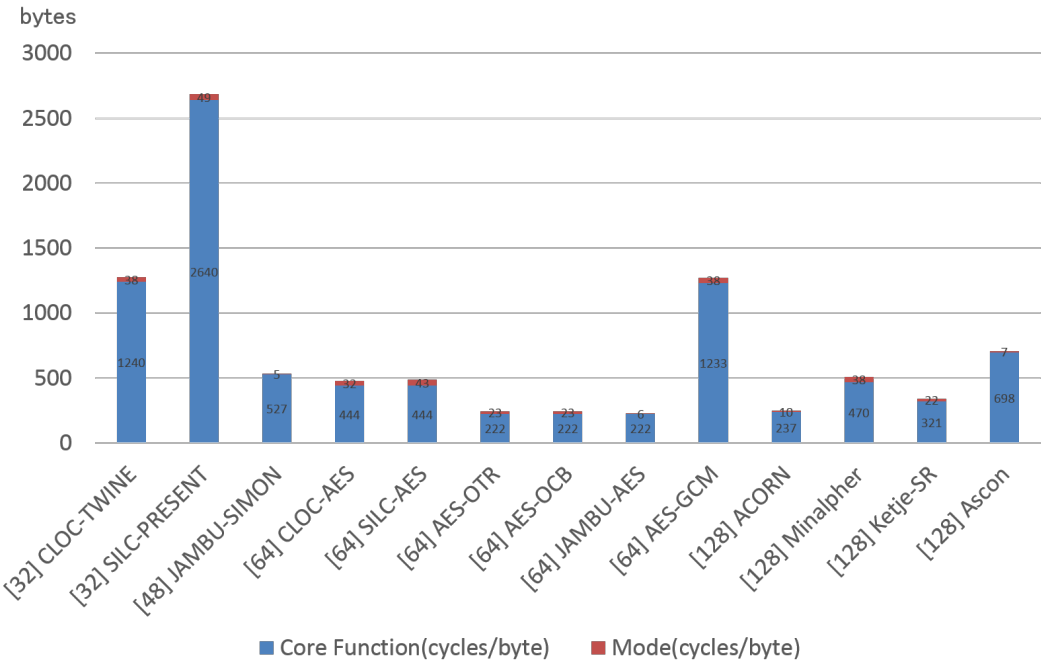

図 3.50　高速実装における暗号化速度（漸近速度）比較（Core Function と Mode に分割）

3.2.1.2 評価方法の概要

■コーディングの方針とインターフェース仕様

認証暗号は平文、暗号文、鍵に加え、関連データ（Associate Data)、ナンス（Nonce）等、入出力パラメータが多く、このため軽量暗号のように速度性能を一次元的に評価することは困難である。そこで本評価では、文献 [32] で示されたコーディングの方針に沿いつつ、各々の認証暗号アルゴリズムをいくつかの部分に分割し、その分割単位ごとに速度性能を評価する方針とした。これにより平文や関連データの長さが与えられれば、本報告で作成された表から、誰もが実際の計算にかかるサイクル数を自分で計算できる。

一方で、このようにアルゴリズムを細かく分割すると、認証暗号全体の処理を行うためには全体のフローをコントロールする上位プログラムが必要であり、そのプログラムのオーバーヘッドが大きくなると、分割単位ごとの性能データを集積しても全体の速度を正しく見積もることができなくなる。

そこで、ここでは上位プログラムのオーバーヘッドができるだけ少なくなるような、ブロック単位での分割の方法を提案する。具体的には、認証暗号のアルゴリズムを、関連データ処理部（以後 AD)、暗号化部（以後 ENC)、復号部（以後 DEC）に分割し、この 3 つの部分それぞれについて、さらに次の 5 つの機能に分割を行った。

1 初期計算（関連データや平文、暗号文を入力する前の処理）
2 第 1 ブロック計算
3 中間ブロック計算（第 2 ブロックから最終ブロックの前までの各 1 ブロックの処理）
4 最終ブロック計算
5 終了計算（関連データや平文、暗号文を入力し終わった後の処理）

これら各機能をひとつの関数としてコーディングしたが、実際には不必要な関数や同じ機能を持つ複数の関数があるため、すべての認証暗号アルゴリズムについて合計 15 個の関数を別々に実装する必要はない。また、一つの機能と別の機能との境界は一意的ではないが、アルゴリズムごとにもっとも自然と考えられる境界を設定した。

以降、これらの関数を次のように記述する。

AD_0: 関連データ処理部の初期計算関数
ENC_123: 暗号化部の第 1、中間、最終ブロック計算関数、これは ENC_1、ENC_2、ENC_3 が共通化できることを意味する。

この方法で記述した関数群を用いて認証暗号全体を記述した上位プログラムを、AES-GCM の暗号化モードを例として図 3.51 に示す。ここで、alen と mlen はそれぞれ関連データ、平文のバイト数である。BLEN はブロック長でこの場合 16 である。赤字が今回作成した関数に対応する。

```
int crypto_aead_encrypt(int mlen, int alen)
{
        int   clen=0,size;

        // Associate Data Handling

        AESGCM_AD_0();                          // Initialization

        while(alen > 0){
                size = (alen >= BLEN) ? BLEN : alen;
                AESGCM_AD_123(size);            // one block processing
                aadr += BLEN;                   // aadr = address of AD
                alen -= BLEN;
        }

        // Message Handling

        AESGCM_ENC_0();                         // Initialization

        while(mlen > 0){
                size = (mlen >= BLEN) ? BLEN : mlen;
                clen += AESGCM_ENC_123(size);   // one block processing
                madr += BLEN;                   // madr = address of MSG
                cadr += BLEN;                   // cadr = address of OUT
                mlen -= BLEN;
        }

        AESGCM_ENC_4(alen, clen);               // Tag Generation

        return clen;
}
```

図 3.51　AES-GCM の暗号化モードのコード例

■AES-GCM

本評価では、文献 [13] に記述されている AES-GCM アルゴリズムを、最も一般的と考えられる次のパラメータで実装した。

表 3.23　本評価で用いた AES-GCM のパラメータ

名称	鍵サイズ	ブロックサイズ	ナンスサイズ	タグサイズ
AES-GCM	128 ビット	128 ビット	96 ビット	64 ビット

また、図 3.52 に示すように AES-GCM アルゴリズムを機能分割した。

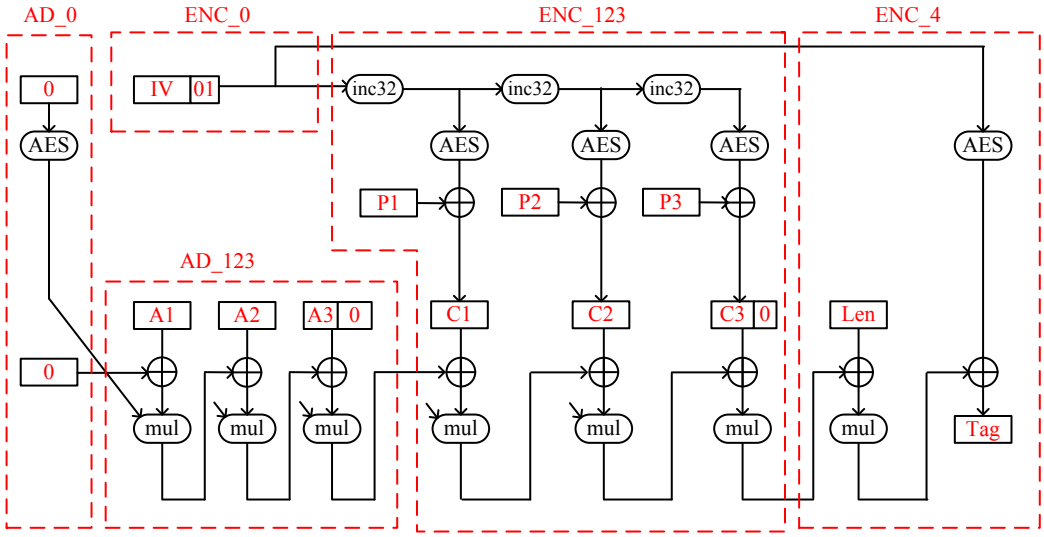

図 3.52　本評価で用いた AES-GCM の機能分割

■CLOC

CLOC アルゴリズム v2 [24] には推奨パラメータが 3 つ示されている。本評価ではこのうち 2 つを実装した。このうちひとつは Core Function として TWINE-64-80 を用いるもの、もうひとつは Core Function として AES-128-128 を用いるものである。

表 3.24　本評価で用いた CLOC のパラメータ

名称	鍵サイズ	ブロックサイズ	ナンスサイズ	タグサイズ
CLOC-TWINE	80 ビット	64 ビット	48 ビット	32 ビット
CLOC-AES	128 ビット	128 ビット	96 ビット	64 ビット

また、図 3.53 に示すように CLOC アルゴリズムを機能分割した。なお、CLOC は平文のサイズが 0 の時に特別の処理を行う仕様となっている。この特別な処理を ENC_NULL と名付けている。

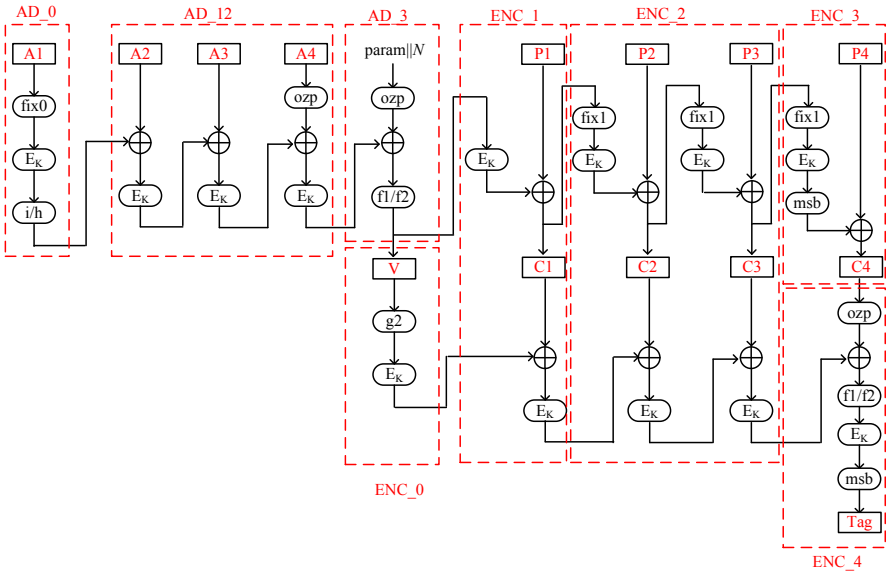

図 3.53　本評価で用いた CLOC の機能分割

■SILC

SILC アルゴリズム v2 [23] には推奨パラメータが 4 つ示されている。本評価ではこのうち 2 つを実装した。このうちひとつは Core Function として PRESENT-64-80 を用いるもの、もうひとつは Core Function として AES-128-128 を用いるものである。

表 3.25　本評価で用いた SILC のパラメータ

名称	鍵サイズ	ブロックサイズ	ナンスサイズ	タグサイズ
SILC-PRESENT	80 ビット	64 ビット	48 ビット	32 ビット
SILC-AES	128 ビット	128 ビット	96 ビット	64 ビット

また、図 3.54 に示すように SILC アルゴリズムを機能分割した。

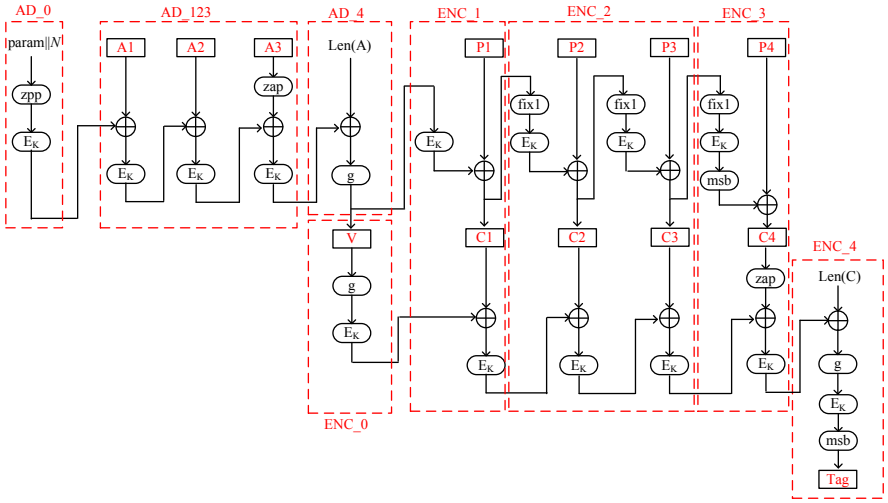

図 3.54　本評価で用いた SILC の機能分割

■Minalpher

Minalpher アルゴリズム [43] は Core Function として、Minalpher-P と呼ばれる置換が用いられている。パラメータは以下の一種類である。

表 3.26　本評価で用いた Minalpher のパラメータ

名称	鍵サイズ	ブロックサイズ	ナンスサイズ	タグサイズ
Minalpher	128 ビット	256 ビット	104 ビット	128 ビット

また、図 3.55 に示すように Minalpher アルゴリズムを機能分割した。

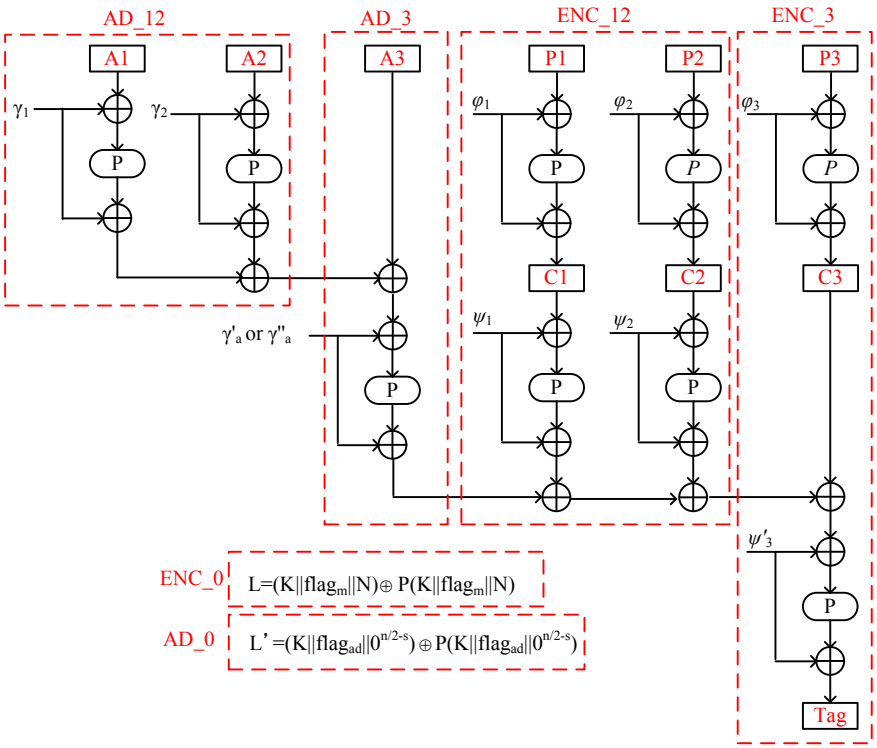

図 3.55　本評価で用いた Minalpher の機能分割

■AES-OTR

AES-OTR アルゴリズム [33] は Core Function として AES-128-128 あるいは AES-128-256 が用いられる。AES-OTR にはいくつかのパラメータが定義されているが、今回実装したのは Primary Parameter と呼ばれる以下のものであり、Core Function として AES-128-128 が使われている。

表 3.27　本評価で用いた AES-OTR のパラメータ

名称	鍵サイズ	ブロックサイズ	ナンスサイズ	タグサイズ
AES-OTR	128 ビット	128 ビット	96 ビット	128 ビット

また、図 3.56 に示すように AES-OTR アルゴリズムを機能分割した.

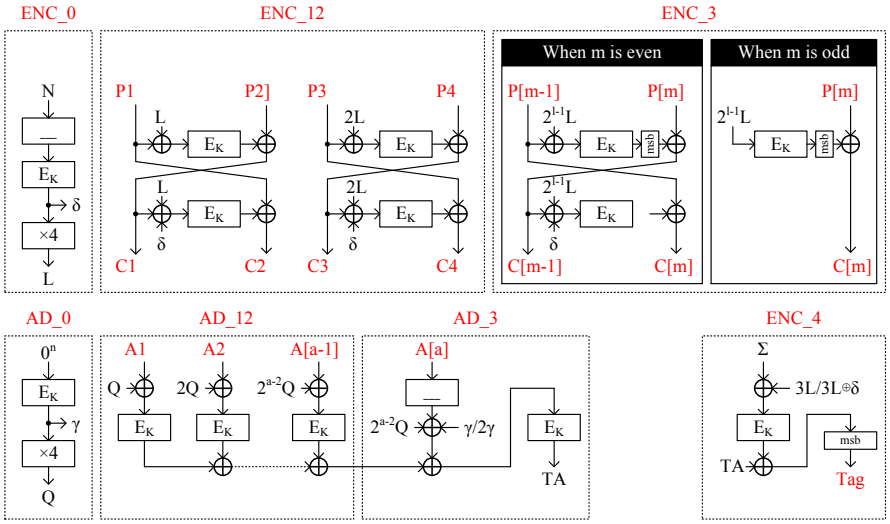

図 3.56　本評価で用いた AES-OTR の機能分割

■Ketje

Ketje アルゴリズム [11] は、Sponge 型認証暗号であり、Core Function として独自の関数 f が用いられている．Ketje には Ketje-SR と Ketje-JR の 2 つのパラメータが定義されているが、今回実装したのは Primary Recommendation とされている、50 バイトの入出力を持つ関数 f を用いたものである．

表 3.28　本評価で用いた Ketje のパラメータ

名称	鍵サイズ	ブロックサイズ	ナンスサイズ	タグサイズ
Ketje-SR	128 ビット	32 ビット	128 ビット	128 ビット

また、図 3.57 に示すように Ketje アルゴリズムを機能分割した．

なお、AD_12 と AD_3 は定数が異なる以外は同じ機能であり、関数 f の下に書かれた数字は関数内部の繰り返し回数を表している。

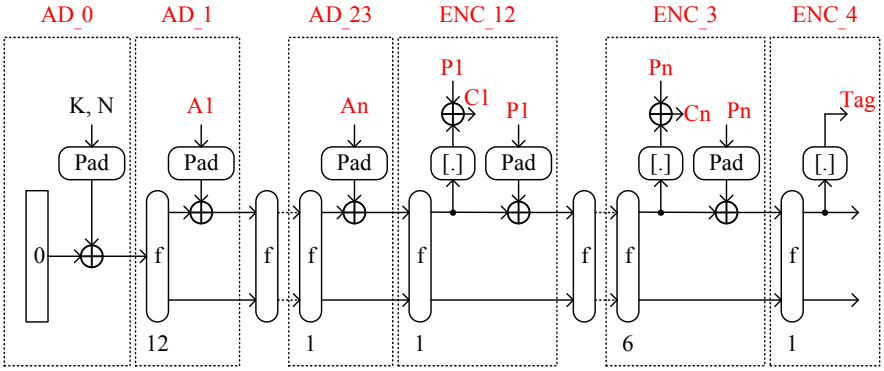

図 3.57　本評価で用いた Ketje の機能分割

■ACORN

ACORN [49] は、ストリーム暗号型の認証暗号である。ACORN では、Core function となる StateUpdate を、制御ビットと入力を変更しながら繰り返し実行することにより、293 ビットの内部状態 state を変更しながら暗号処理を実行する。本評価では、ACORN アルゴリズムを表 3.29 に示すパラメータで実装した。

表 3.29　本評価で用いた ACORN のパラメータ

名称	鍵サイズ	ブロックサイズ	ナンスサイズ	タグサイズ
ACORN	128 ビット	128 ビット	128 ビット	128 ビット

また、図 3.58 に示すように ACORN アルゴリズムを機能分割した.

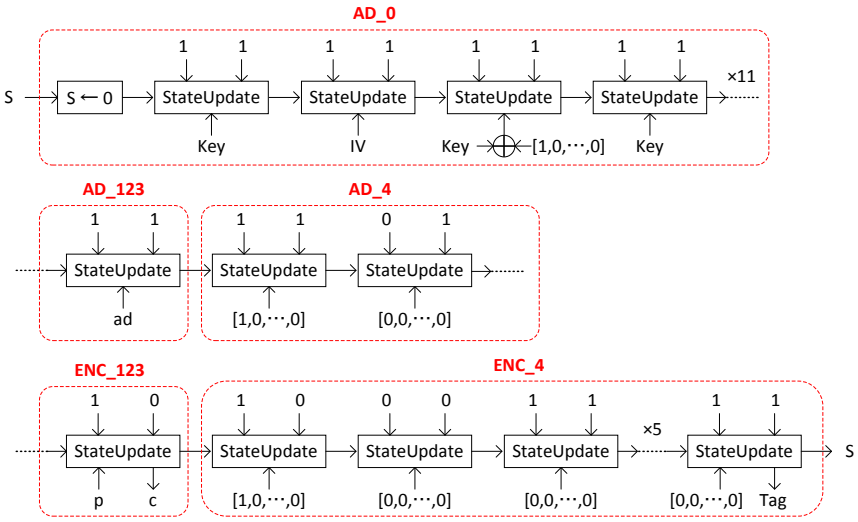

図 3.58　本評価で用いた ACORN の機能分割

■AES-OCB

AES-OCB アルゴリズム [29] は Core Function として AES-128-128、AES-128-192、あるいは AES-128-256 が用いられる。ここでは Primary Recommendation と考えられる、Core Function として AES-128-128 を用いた以下のパラメータのものを実装した。なお、AES-OCB は本報告書でとりあげた他の AES ベースの認証暗号の中で唯一（認証暗号としての）復号時に AES の復号機能を必要とする。

表 3.30　本評価で用いた AES-OCB のパラメータ

名称	鍵サイズ	ブロックサイズ	ナンスサイズ	タグサイズ
AES-OCB	128 ビット	128 ビット	96 ビット	128 ビット

また、図 3.59 に示すように AES-OCB アルゴリズムを機能分割した。

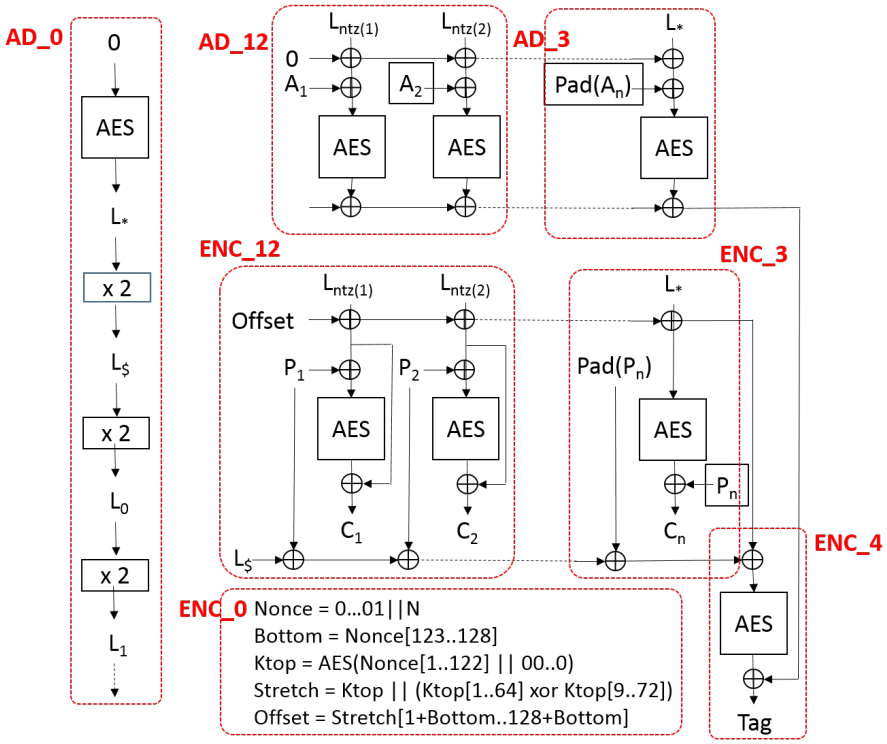

図 3.59　本評価で用いた AES-OCB の機能分割

■JAMBU

JAMBU アルゴリズム [50] は、Core Function として Simon-96-96、Simon-64-96、Simon-128-128 あるいは AES-128-128 が用いられる。ここでは Primary recommendation とされている Simon-96-96 を用いたものに加えて、AES-128-128 を用いたものの 2 種類を実装した。これらはそれぞれ以下のパラメータを持つものである。

表 3.31　本評価で用いた JAMBU のパラメータ

名称	鍵サイズ	ブロックサイズ	ナンスサイズ	タグサイズ
JAMBU-SIMON	96 ビット	96 ビット	48 ビット	48 ビット
JAMBU-AES	128 ビット	128 ビット	64 ビット	64 ビット

また、図 3.60 に示すように JAMBU アルゴリズムを機能分割した。

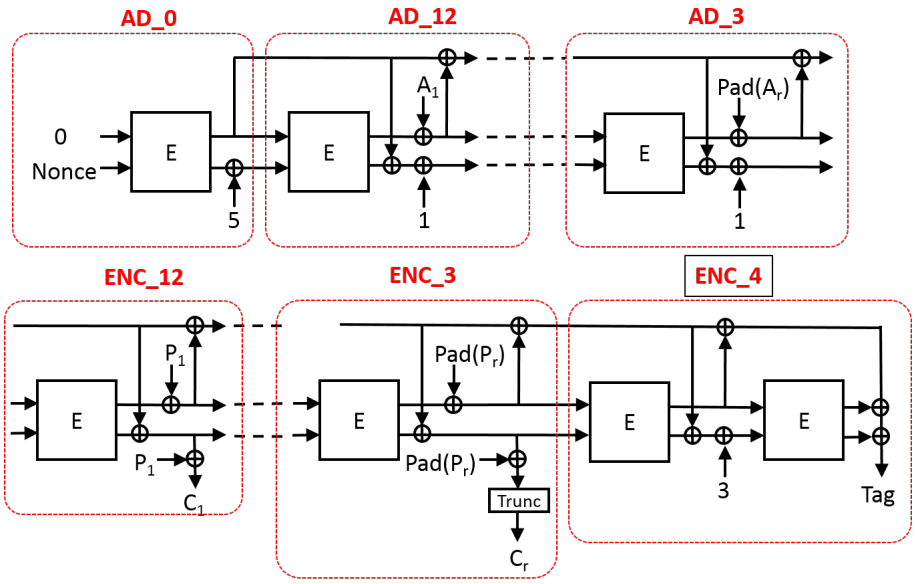

図 3.60　本評価で用いた JAMBU の機能分割

■Ascon

Ascon アルゴリズム [12] は Ketje と同じく Sponge 型認証暗号であり、独自の関数 p が内部で用いられている。ここでは Ascon の Primary Recommendation である、以下のパラメータを持つものを実装した。なお、関数 p は 40 バイトの入出力をもつものである。

表 3.32　本評価で用いた Ascon のパラメータ

名称	鍵サイズ	ブロックサイズ	ナンスサイズ	タグサイズ
Ascon	128 ビット	128 ビット	96 ビット	128 ビット

また、図 3.61 に示すように Ascon アルゴリズムを機能分割した。ここで、p^12, p^6 と示されている関数は、それぞれ関数 p を 12 回、6 回実行させることを示している。

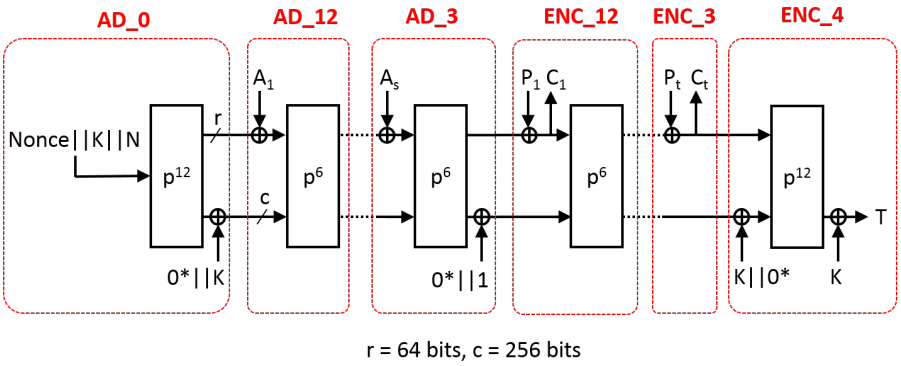

図 3.61　本評価で用いた Ascon の機能分割

3.3 Ascon の実装性能

3.3.1 ハードウェア実装性能

本節では、Ascon のハードウェア実装（特に、FPGA 実装と ASIC 実装）性能について、2022 年度に公開された CRYPTREC 外部評価報告書 [53] に基づき、2022 年 9 月現在の調査結果を掲載する。

3.3.1.1 調査対象と性能評価環境

ハードウェア実装（またはハードウェアアクセラレータ）とは、いわゆる専用回路実装と呼ばれるものであり、処理に必要なデータを供給して結果を出力させるものである。専用回路の処理ではインタフェース部がボトルネックとなる場合があるため、入出力データのインタフェース部も含めて実装する必要がある。NIST LWC ファイナリストの実装性能を可能な限り公平に評価する目的で、CAESAR コンペティションで使用された CAESAR HW API [22] と NIST LWC プロジェクトで提案された LWC HW API [26] が使用されることが多い。いずれのインタフェースも暗号処理性能を評価する上で大きな違いはないが、入力されるデータを適切に処理するために、ある程度のバッファメモリを用意する必要がある。これに伴い、全体の回路面積が増えることになるが、ハードウェア上で安定したデータ転送の実現を優先するために欠かせないものである。その他、アクセラレータがバスに対して優先的にデータを転送するためのバスアービトレーションを考慮することも重要である。このような詳細な検討は、ハードウェアアーキテクチャの仕様を厳密に策定する場合に必要となる。

■FPGA 実装　NIST LWC ファイナリストの FPGA 実装については、CAESAR コンペティションでの評価を含め、いくつかの報告がある [15, 41, 47, 51]。CAESAR コンペティションでの評価対象は、NIST LWC プロジェクトでの評価対象と仕様が異なる場合があるものの、実装性能に関する貴重な情報であることから、本ガイドラインでも紹介する。

FPGA 上の回路規模（面積コスト）の単位は統一されていない。AMD 社（旧 Xilinx 社）の FPGA は、LUT (Look-up Table) 数で評価することが一般的であり、Spartan-6、Artix-7、そして Zynq-7000 はいずれも 6 入力の LUT で評価される。一方、Intel（旧 Altera 社）の FPGA は、Cyclone-V の ALM (Adaptive Logic Module) や Cyclone 10 の LE (Logic Element) と呼ばれる複数の LUT を含むモジュールをビルディングブロックとし、その数で評価している。いずれも、2 入力 NAND ゲートを 1 単位とする GE (Gate Equivalent) に換算することが可能である。

Mohajerani らの研究 [35] では、複数の異なる FPGA に対し、入力データの違いによる認証暗号とハッシュ関数の処理性能を網羅的に比較している。インタフェースには LWC HW API [26] が使用されている。本ガイドラインでは、Mohajerani らによる評価結果の中から、特に重要と考えられる実装コストとスループット性能について紹介する。

■ASIC 実装　ASIC 実装の性能評価では、回路規模やスループット性能[*1]に加え、消費電力やエネルギー（電力量）効率がより重要となる。これはリソースの限られた IoT デバイスなどを想定しているためである。例えば、バッテリーを持たないデバイスでは消費電力が他の指標よりも重要となり、バッテリー駆動のデバイスではデバイスの寿命に直結するエネルギー効率が重要となる。エネルギーは電力を時間積分したものであるため、デバイスの使用率や消費電力の管理手法によりエネルギー効率は大きく変わる。理想的には、認証暗号が組み込まれるデバイス全体の消費電力やエネルギー効率が評価されるべきであるものの、このような実使用下のフィールドテストによる評価は難しく、アプリケーションにも大きく依存することから、暗号処理中に特化して消費電力とエネルギー効率を評価する研究が多い。

高スループットを得るために、本来であれば 1 サイクルで実行する関数（ラウンド関数など）をまとめて実行する Unrolled 実装型アーキテクチャを設計し、そのアーキテクチャの実装性能を評価するという研究が盛んに行われている。暗号アルゴリズムの種類によって Unrolled 実装に適さないものもあるが、Ascon に関しては Unrolled 実装に対して柔軟に適用可能であることが知られている。Unrolled 実装型アーキテクチャの特徴は、次のとおりである。

- アンロールするラウンド数が多くなるほど組合せ回路の面積が大きくなり、結果として全体の回路面積が大きくなる。一方、順序回路の規模は変わらない。
- 組合せ回路のクリティカルパス遅延時間が長くなり、結果として最大動作周波数が低下する。一方、組合せ回路を最

[*1] ここではレイテンシの逆数としての評価指標と位置づけ、パイプライン化による向上は想定しない。

適化し、最大動作周波数の低下を抑制することが可能である。

- 消費電力が増加する。一方、所望の処理を短時間で実行でき、エネルギー効率を向上できる。

本ガイドラインでは、Großらの研究 [19] と Elsadek らの研究 [14] で報告された評価結果を紹介する。Großらは、高いスループット性能を達成するために 1 種類の非 Unrolled 実装と 3 種類の Unrolled 実装で性能評価を実施するとともに、面積コストを抑制するために 2 種類のコンパクト実装で性能評価を実施している。また、Elsadek らはスループット性能とエネルギー効率に焦点を当てて性能評価を実施している。

3.3.1.2 実装性能

■FPGA 実装 表 3.33 は文献 [15, 41, 47, 51] で報告された FPGA 実装の評価結果をまとめたものである。この表では、FPGA の種類（プラットフォーム）、インタフェース、面積コスト、スループット性能を掲載している。

表 3.33 Ascon の FPGA 実装性能評価結果 [15, 41, 47, 51]

名称	プラットフォーム	インタフェース	面積コスト	スループット
Ascon-128 [15]	Spartan-6	CAESAR HW API	1,402 LUTs	1,906 Mbps
Ascon-128a [15]			1,712 LUTs	2,884 Mbps
Ascon-128 [51]	Spartan-6	CAESAR HW API	684 LUTs	60 Mbps
Ascon-128a [51]			684 LUTs	119 Mbps
Ascon-128 [41]	Artix-7	LWC HW API	1,898 LUTs	1,683 Mbps
	Spartan-6		1,913 LUTs	1,116 Mbps
	Cyclone-V		1,051 ALMs	1,295 Mbps
Ascon-Hash [41]	Artix-7	LWC HW API	2,181 LUTs	1,032 Mbps
	Spartan-6		2,188 LUTs	678 Mbps
	Cyclone-V		1,064 ALMs	898 Mbps
Ascon-128 [47]	Zynq-7000-6	CAESAR HW API	6,325 LUTs	–

表 3.33 の中で最もコンパクトな実装は、文献 [51] で報告された結果であり、その面積コストは 684 LUTs である。最も高速な実装は，文献 [15] で報告された結果であり、約 2.9 Gbps のスループット性能を 1,712 LUTs の面積コストで達成した。また、文献 [41] では、ハッシュ関数として機能させた場合、Artix-7 上で 1.0 Gbps の処理性能を 2,181 LUTs の回路面積で達成したと報告されている。

インタフェースの使用はハードウェアモジュールの面積コストと処理性能に大きな影響を及ぼすため、表の数値だけで Ascon の正確な実装性能を評価することは難しい。一方で、表 3.33 の結果から、Ascon が軽量実装可能であることや高い処理性能を実現可能であることを読み取ることができる。このため、Ascon には実装性能のトレードオフを模索できる柔軟性があると言える。

その他、Mohajerani らの研究 [35] を紹介する。Mohajerani らは、複数の異なる種類の FPGA に対し、入力データの違いによる認証暗号とハッシュ関数の処理性能を網羅的に比較している。評価結果が膨大であるため、面積コストとスループット性能に限定し、評価結果の一部を表 3.34 に掲載する。

表 3.34: Ascon の FPGA 実装性能評価結果 [35]

名称	プラットフォーム	データ量	面積コスト	スループット
Ascon-GMU-v1	Artix-7	AD+PT (Long)	2,410 LUTs	6,297 Mbp
		Hash (Long)	–	–
	Cyclone 10	AD+PT (Long)	4,552 LEs	3,031 Mbps

（次のページに続く）

（前のページからの続き）

名称	プラットフォーム	データ量	面積コスト	スループット
		Hash (Long)	2,415 LEs	864 Mbps
	ECP5	AD+PT (Long)	5,909 LUTs	2,158 Mbps
		Hash (Long)	–	–
	Artix-7	AD+PT (1,536 Bytes)	2,410 LUTs	3,022 Mbps
		AD+PT (64 Bytes)		1,574 Mbps
		AD+PT (16 Bytes)		629 Mbps
		Hash	–	–
	Cyclone 10	AD+PT (1,536 Bytes)	4,552 LEs	1,454 Mbps
		AD+PT (64 Bytes)		757 Mbps
		AD+PT (16 Bytes)		303 Mbps
		Hash	–	–
	ECP5	AD+PT (1,536 Bytes)	5,909 LUTs	1,035 Mbps
		AD+PT (64 Bytes)		539 Mbps
		AD+PT (16 Bytes)		215 Mbps
		Hash	–	–
Ascon-GMU-v2	Artix-7	AD+PT (Long)	1,790 LUTs	4,366 Mbps
		Hash (Long)	–	–
	Cyclone 10	AD+PT (Long)	3,113 LEs	2,284 Mbps
		Hash (Long)	3,215 LEs	1,232 Mbps
	ECP5	AD+PT (Long)	4,641 LUTs	1,666 Mbps
		Hash (Long)	–	–
	Artix-7	AD+PT (1,536 Bytes)	1,790 LUTs	2,115 Mbps
		AD+PT (64 Bytes)		1,237 Mbps
		AD+PT (16 Bytes)		538 Mbps
		Hash	–	–
	Cyclone 10	AD+PT (1.536 Bytes)	3,113 LEs	1,107 Mbps
		AD+PT (64 Bytes)		647 Mbps
		AD+PT (16 Bytes)		281 Mbps
		Hash	–	–
	ECP5	AD+PT (1,536 Bytes)	4,641 LUTs	807 Mbps
		AD+PT (64 Bytes)		472 Mbps
		AD+PT (16 Bytes)		205 Mbps
		Hash	–	–
Ascon-GMU2-v1h	Artix-7	AD+PT (Long)	1,375 LUTs	2,523 Mbps
		Hash (Long)		1,358 Mbps
	Cyclone 10	AD+PT (Long)	2,415 LEs	1,605 Mbps
		Hash (Long)	4,161 LEs	1,173 Mbps
	ECP5	AD+PT (Long)	2,928 LUTs	1,006 Mbps
		Hash (Long)		541 Mbps
	Artix-7	AD+PT (1,536 Bytes)	1,375 LUTs	1,236 Mbps
		AD+PT (64 Bytes)		851 Mbps

（次のページに続く）

（前のページからの続き）

名称	プラットフォーム	データ量	面積コスト	スループット
		AD+PT (16 Bytes)		430 Mbps
		Hash (1,536 Bytes)		1,321 Mbps
		Hash (64 Bytes)		812 Mbps
		Hash (16 Bytes)		368 Mbps
	Cyclone 10	AD+PT (1,536 Bytes)	2,415 LEs	786 Mbps
		AD+PT (64 Bytes)		541 Mbps
		AD+PT (16 Bytes)		274 Mbps
		Hash (1,536 Bytes)		840 Mbps
		Hash (64 Bytes)		516 Mbps
		Hash (16 Bytes)		234 Mbps
	ECP5	AD+PT (1,536 Bytes)	2,928 LUTs	493 Mbps
		AD+PT (64 Bytes)		339 Mbps
		AD+PT (16 Bytes)		171 Mbps
		Hash (1,536 Bytes)		527 Mbps
		Hash (64 Bytes)		323 Mbps
		Hash (16 Bytes)		146 Mbps
Ascon-GMU2-v2h	Artix-7	AD+PT (Long)	2,126 LUTs	3,744 Mbps
		Hash (Long)		2,139 Mbps
	Cyclone 10	AD+PT (Long)	3,215 LEs	2,157 Mbps
		Hash (Long)	–	–
	ECP5	AD+PT (Long)	3,764 LUTs	1,427 Mbps
		Hash (Long)		815 Mbps
	Artix-7	AD+PT (1,536 Bytes)	2,126 LUTs	1,825 Mbps
		AD+PT (64 Bytes)		1,163 Mbps
		AD+PT (16 Bytes)		544 Mbps
		Hash (1,536 Bytes)		2,077 Mbps
		Hash (64 Bytes)		1,248 Mbps
		Hash (16 Bytes)		554 Mbps
	Cyclone 10	AD+PT (1,536 Bytes)	3,215 LEs	1,051 Mbps
		AD+PT (64 Bytes)		670 Mbps
		AD+PT (16 Bytes)		313 Mbps
		Hash (1,536 Bytes)		1,196 Mbps
		Hash (64 Bytes)		719 Mbps
		Hash (16 Bytes)		319 Mbps
	ECP5	AD+PT (1,536 Bytes)	3,764 LUTs	696 Mbps
		AD+PT (64 Bytes)		443 Mbps
		AD+PT (16 Bytes)		207 Mbps
		Hash (1,536 Bytes)		792 Mbps
		Hash (64 Bytes)		475 Mbps
		Hash (16 Bytes)		211 Mbps
Ascon-GMU2-v3h	Artix-7	AD+PT (Long)	2,493 LUTs	3,029 Mbps

（次のページに続く）

（前のページからの続き）

名称	プラットフォーム	データ量	面積コスト	スループット
		Hash (Long)		1,817 Mbps
	Cyclone 10	AD+PT (Long)	4,161 LEs	1,955 Mbps
		Hash (Long)	–	–
	ECP5	AD+PT (Long)	4,925 LUTs	1,305 Mbps
		Hash (Long)		783 Mbps
	Artix-7	AD+PT (1,536 Bytes)	2,493 LUTs	1,470 Mbps
		AD+PT (64 Bytes)		876 Mbps
		AD+PT (16 Bytes)		386 Mbps
		Hash (1,536 Bytes)		1,762 Mbps
		Hash (64 Bytes)		1,038 Mbps
		Hash (16 Bytes)		454 Mbps
	Cyclone 10	AD+PT (1,536 Bytes)	4,161 LEs	949 Mbps
		AD+PT (64 Bytes)		565 Mbps
		AD+PT (16 Bytes)		249 Mbps
		Hash (1,536 Bytes)		1,137 Mbps
		Hash (64 Bytes)		670 Mbps
		Hash (16 Bytes)		293 Mbps
	ECP5	AD+PT (1,536 Bytes)	4,925 LUTs	633 Mbps
		AD+PT (64 Bytes)		377 Mbps
		AD+PT (16 Bytes)		166 Mbps
		Hash (1,536 Bytes)		759 Mbps
		Hash (64 Bytes)		447 Mbps
		Hash (16 Bytes)		195 Mbps
Ascon-Graz-v1	Artix-7	AD+PT (Long)	1,465 LUTs	1,528 Mbps
		Hash (Long)		873 Mbps
	Cyclone 10	AD+PT (Long)	2,517 LEs	1,131 Mbps
		Hash (Long)		646 Mbps
	ECP5	AD+PT (Long)	2,544 LUTs	474 Mbps
		Hash (Long)		271 Mbps
	Artix-7	AD+PT (1,536 Bytes)	1,465 LUTs	752 Mbps
		AD+PT (64 Bytes)		552 Mbps
		AD+PT (16 Bytes)		301 Mbps
		Hash (1,536 Bytes)		850 Mbps
		Hash (64 Bytes)		528 Mbps
		Hash (16 Bytes)		242 Mbps
	Cyclone 10	AD+PT (1,536 Bytes)	2,517 LEs	556 Mbps
		AD+PT (64 Bytes)		409 Mbps
		AD+PT (16 Bytes)		223 Mbps
		Hash (1,536 Bytes)		629 Mbps
		Hash (64 Bytes)		391 Mbps
		Hash (16 Bytes)		179 Mbps

（次のページに続く）

（前のページからの続き）

名称	プラットフォーム	データ量	面積コスト	スループット
	ECP5	AD+PT (1,536 Bytes)	2,544 LUTs	233 Mbps
		AD+PT (64 Bytes)		171 Mbps
		AD+PT (16 Bytes)		93 Mbps
		Hash (1,536 Bytes)		263 Mbps
		Hash (64 Bytes)		164 Mbps
		Hash (16 Bytes)		75 Mbps
Ascon-Graz-v2	Artix-7	AD+PT (Long)	1,541 LUTs	2,272 Mbps
		Hash (Long)		973 Mbps
	Cyclone 10	AD+PT (Long)	2,634 LEs	1,529 Mbps
		Hash (Long)		655 Mbps
	ECP5	AD+PT (Long)	2,603 LUTs	683 Mbps
		Hash (Long)		292 Mbps
	Artix-7	AD+PT (1,536 Bytes)	1,541 LUTs	1,108 Mbps
		AD+PT (64 Bytes)		712 Mbps
		AD+PT (16 Bytes)		336 Mbps
		Hash (1,536 Bytes)		948 Mbps
		Hash (64 Bytes)		589 Mbps
		Hash (16 Bytes)		269 Mbps
	Cyclone 10	AD+PT (1,536 Bytes)	2,634 LEs	746 Mbps
		AD+PT (64 Bytes)		479 Mbps
		AD+PT (16 Bytes)		226 Mbps
		Hash (1,536 Bytes)		638 Mbps
		Hash (64 Bytes)		396 Mbps
		Hash (16 Bytes)		181 Mbps
	ECP5	AD+PT (1,536 Bytes)	2,603 LUTs	333 Mbps
		AD+PT (64 Bytes)		214 Mbps
		AD+PT (16 Bytes)		101 Mbps
		Hash (1,536 Bytes)		285 Mbps
		Hash (64 Bytes)		177 Mbps
		Hash (16 Bytes)		81 Mbps
Ascon-Graz-v3	Artix-7	AD+PT (Long)	2,142 LUTs	2,572 Mbps
		Hash (Long)		1,608 Mbps
	Cyclone 10	AD+PT (Long)	3,716 LEs	1,403 Mbps
		Hash (Long)		877 Mbps
	ECP5	AD+PT (Long)	3,305 LUTs	815 Mbps
		Hash (Long)		509 Mbps
	Artix-7	AD+PT (1,536 Bytes)	2,142 LUTs	1,260 Mbps
		AD+PT (64 Bytes)		857 Mbps
		AD+PT (16 Bytes)		428 Mbps
		Hash (1,536 Bytes)		1,564 Mbps
		Hash (64 Bytes)		961 Mbps

（次のページに続く）

（前のページからの続き）

名称	プラットフォーム	データ量	面積コスト	スループット
		Hash (16 Bytes)		436 Mbps
	Cyclone 10	AD+PT (1,536 Bytes)	3,716 LEs	687 Mbps
		AD+PT (64 Bytes)		467 Mbps
		AD+PT (16 Bytes)		233 Mbps
		Hash (1,536 Bytes)		853 Mbps
		Hash (64 Bytes)		524 Mbps
		Hash (16 Bytes)		237 Mbps
	ECP5	AD+PT (1,536 Bytes)	3,305 LUTs	399 Mbps
		AD+PT (64 Bytes)		271 Mbps
		AD+PT (16 Bytes)		135 Mbps
		Hash (1,536 Bytes)		495 Mbps
		Hash (64 Bytes)		304 Mbps
		Hash (16 Bytes)		138 Mbps
Ascon-Graz-v4	Artix-7	AD+PT (Long)	2,249 LUTs	3,296 Mbps
		Hash (Long)		1,648 Mbps
	Cyclone 10	AD+PT (Long)	3,730 LEs	1,738 Mbps
		Hash (Long)		869 Mbps
	ECP5	AD+PT (Long)	3,379 LUTs	989 Mbps
		Hash (Long)		494 Mbps
	Artix-7	AD+PT (1,536 Bytes)	2,249 LUTs	1,605 Mbps
		AD+PT (64 Bytes)		1,004 Mbps
		AD+PT (16 Bytes)		462 Mbps
		Hash (1,536 Bytes)		1,603 Mbps
		Hash (64 Bytes)		985 Mbps
		Hash (16 Bytes)		446 Mbps
	Cyclone 10	AD+PT (1,536 Bytes)	3,730 LEs	846 Mbps
		AD+PT (64 Bytes)		529 Mbps
		AD+PT (16 Bytes)		244 Mbps
		Hash (1,536 Bytes)		845 Mbps
		Hash (64 Bytes)		519 Mbps
		Hash (16 Bytes)		235 Mbps
	ECP5	AD+PT (1,536 Bytes)	3,379 LUTs	481 Mbps
		AD+PT (64 Bytes)		301 Mbps
		AD+PT (16 Bytes)		138 Mbps
		Hash (1,536 Bytes)		481 Mbps
		Hash (64 Bytes)		296 Mbps
		Hash (16 Bytes)		134 Mbps
Ascon-Graz-v5	Artix-7	AD+PT (Long)	2,797 LUTs	2,400 Mbps
		Hash (Long)		1,600 Mbps
	Cyclone 10	AD+PT (Long)	4,905 LEs	1,281 Mbps
		Hash (Long)		854 Mbps

（次のページに続く）

（前のページからの続き）

名称	プラットフォーム	データ量	面積コスト	スループット
	ECP5	AD+PT (Long)	4,646 LUTs	889 Mbps
		Hash (Long)		593 Mbps
	Artix-7	AD+PT (1,536 Bytes)	2,797 LUTs	1,173 Mbps
		AD+PT (64 Bytes)		775 Mbps
		AD+PT (16 Bytes)		376 Mbps
		Hash (1,536 Bytes)		1,555 Mbps
		Hash (64 Bytes)		948 Mbps
		Hash (16 Bytes)		426 Mbps
	Cyclone 10	AD+PT (1,536 Bytes)	4,905 LEs	626 Mbps
		AD+PT (64 Bytes)		414 Mbps
		AD+PT (16 Bytes)		201 Mbps
		Hash (1,536 Bytes)		830 Mbps
		Hash (64 Bytes)		506 Mbps
		Hash (16 Bytes)		227 Mbps
	ECP5	AD+PT (1,536 Bytes)	4,646 LUTs	435 Mbps
		AD+PT (64 Bytes)		287 Mbps
		AD+PT (16 Bytes)		139 Mbps
		Hash (1,536 Bytes)		576 Mbps
		Hash (64 Bytes)		351 Mbps
		Hash (16 Bytes)		158 Mbps
Ascon-Graz-v6	Artix-7	AD+PT	–	–
		Hash	–	–
	Cyclone 10	AD+PT	–	–
		Hash	–	–
	ECP5	AD+PT (Long)	5,346 LUTs	827 Mbps
		Hash (Long)		496 Mbps
		AD+PT (1,536 Bytes)		402 Mbps
		AD+PT (64 Bytes)		245 Mbps
		AD+PT (16 Bytes)		110 Mbps
		Hash (1,536 Bytes)		482 Mbps
		Hash (64 Bytes)		292 Mbps
		Hash (16 Bytes)		130 Mbps
Ascon-VT-v1	Artix-7	AD+PT (Long)	1,913 LUTs	1,491 Mbps
		Hash (Long)	–	–
	Cyclone 10	AD+PT (Long)	2,432 LEs	1,130 Mbps
		Hash (Long)	–	–
	ECP5	AD+PT (Long)	3,130 LUTs	543 Mbps
		Hash (Long)	–	–
	Artix-7	AD+PT (1,536 Bytes)	1,913 LUTs	735 Mbps
		AD+PT (64 Bytes)		560 Mbps
		AD+PT (16 Bytes)		320 Mbps

（次のページに続く）

（前のページからの続き）

名称	プラットフォーム	データ量	面積コスト	スループット
	Cyclone 10	AD+PT (1,536 Bytes)	2,432 LEs	557 Mbps
		AD+PT (64 Bytes)		424 Mbps
		AD+PT (16 Bytes)		243 Mbps
	ECP5	AD+PT (1,536 Bytes)	3,130 LUTs	268 Mbps
		AD+PT (64 Bytes)		204 Mbps
		AD+PT (16 Bytes)		116 Mbps
Ascon-VT-v2	Artix-7	AD+PT (Long)	1,928 LUTs	1,475 Mbps
		Hash (Long)		934 Mbps
	Cyclone 10	AD+PT (Long)	2,695 LEs	1,158 Mbps
		Hash (Long)		733 Mbps
	ECP5	AD+PT (Long)	3,041 LUTs	508 Mbps
		Hash (Long)		321 Mbps
	Artix-7	AD+PT (1,536 Bytes)	1,928 LUTs	726 Mbps
		AD+PT (64 Bytes)		544 Mbps
		AD+PT (16 Bytes)		304 Mbps
		Hash (1,536 Bytes)		910 Mbps
		Hash (64 Bytes)		572 Mbps
		Hash (16 Bytes)		264 Mbps
	Cyclone 10	AD+PT (1,536 Bytes)	2,695 LEs	570 Mbps
		AD+PT (64 Bytes)		427 Mbps
		AD+PT (16 Bytes)		239 Mbps
		Hash (1,536 Bytes)		715 Mbps
		Hash (64 Bytes)		449 Mbps
		Hash (16 Bytes)		207 Mbps
	ECP5	AD+PT (1,536 Bytes)	3,041 LUTs	250 Mbps
		AD+PT (64 Bytes)		187 Mbps
		AD+PT (16 Bytes)		104 Mbps
		Hash (1,536 Bytes)		313 Mbps
		Hash (64 Bytes)		197 Mbps
		Hash (16 Bytes)		91 Mbps

表 3.34 の結果について、認証暗号の性能は暗号化時のものであり、インタフェースは LWC HW API を使用している。名称の項目は RTL (Register Transfer Level) コードの名前であり、設計者とバージョンの違いで区別されている。データ量の項目は入力データの違いを記載しており、例えば “AD+PT (Long)” は十分にデータ長の長い関連データ (AD: associated data) と平文 (PT: plaintext) に対して暗号化処理を行う場合を表している。括弧内がバイト数の場合は、入力長を表している。表 3.34 から、同じ FPGA 実装でも入力長の違いによって処理性能が異なることがわかる。最も高速な実装性能は、2,410 LUTs の面積コストで 6 Gbps を超えるスループット性能を達成している。これは Artix-7 上で、入力データ量が AD+PT (Long) の場合の評価結果である[*2]。

■ASIC 実装　表 3.35 は文献 [19] で報告された Großらによる ASIC 実装の評価結果をまとめたものである。Großらの研究では、90 nm UMC low-K ライブラリを使用して Ascon の ASIC 実装に対し、面積コスト、スループット、消費電力、エネ

[*2] 実際に高いスループット性能を達成するためには、データの入出力がボトルネックにならないことが条件となる。

ルギー消費量を評価している。なお、インタフェース部（鍵レジスタと 64 ビットのバスインタフェース）の回路コストは、0.87 kGE から 1.18 kGE のゲートサイズである。

表 3.35　Großらによる Ascon の ASIC 実装評価結果 [19]

名称	インタフェース	面積コスト	スループット @f_{max}	消費電力 @1MHz	エネルギー消費量
Ascon-fast 1 round	no	7.08 kGE	5,524 Mbps	43 μW	33 μJ/byte
	custom	7.95 kGE			
Ascon-fast 2 rounds	no	10.61 kGE	8,425 Mbps	72 μW	27 μJ/byte
	custom	11.48 kGE			
Ascon-fast 3 rounds	no	14.26 kGE	10,407 Mbps	102 μW	25 μJ/byte
	custom	15.13 kGE			
Ascon-fast 6 rounds	no	24.93 kGE	13,218 Mbps	184 μW	23 μJ/byte
	custom	25.80 kGE			
Ascon-64-bit	no	4.99 kGE	72 Mbps	32 μW	1,397 μJ/byte
	custom	5.86 kGE			
Ascon-x-low-area	no	2.57 kGE	14 Mbps	15 μW	5,706 μJ/byte
	custom	3.75 kGE			

Ascon-fast は、高いスループット性能を実現するための実装である。Ascon-fast 1 round は、Ascon-fast をベースとして設計された非 Unrolled 実装である。また、Ascon-fast 2 rounds、Ascon-fast 3 rounds、Ascon-fast 6 rounds は、それぞれ 2、3、6 ラウンド分を 1 サイクルで実行する Unrolled 実装である。表 3.35 からも、Ascon には面積コストと処理性能のトレードオフを模索できる柔軟性があると言える。アンロール数が多くなるほど処理性能が高くなり、特に Ascon-fast 6 rounds で約 13 Gbps のスループット性能を達成していることがわかる。また、Ascon-fast 6 rounds において消費電力が最大になるものの、処理時間が短くなるためにエネルギー効率が最も良いことがわかる。

Ascon-64-bit は、64 ビットの算術論理ユニット（ALU）に基づくデータパスとして設計されたものである。ソフトウェア実装のようにラウンド処理を複数のサイクルに分けて実行するため、処理性能は低下するものの面積コストを抑制できるという利点がある。また、Ascon-x-low-area は面積コストをさらに抑制するために設計されたものである。Ascon-64-bit よりもさらに小さいデータパスを利用することでコンパクト実装を実現している。Ascon-64-bit と Ascon-x-low-area は、いずれも消費電力を抑制できたが、Ascon-fast と比べてエネルギー消費量は大幅に増加する。

表 3.36 は文献 [14] で報告された Elsadek らによる ASIC 実装の評価結果をまとめたものである。Elsadek らの研究では、GF 22 nm CMOS (GF22FDx) で合成した Ascon の ASIC 実装に対し、スループット性能とエネルギー効率を評価している。なお、インターフェース回路とレイアウトについて考慮されておらず、コア関数のみの評価となっていることに注意されたい。

入力データ長が短い（Short）場合では、16 バイトのデータを間隔を空けて送信することを想定している。また、入力データ長が長い（Long）場合では、1,536 バイトの連続した入力データの処理を想定している。Short の場合と比べると、Long の場合の方がスループット性能は高くなり、500 Mbps を超えていることがわかる。

エネルギー効率については、以下の式が利用されている。

$$\text{エネルギー効率 [bit/J]} = \frac{\text{スループット [bit/sec]}}{\text{消費電力 [W]}}$$

例えば、表 3.36 の 1 行目のデータに対する平均消費電力は、$39.1/407.2 = 0.0960$ となるため、96 μW と算出できる。また、1 ビットの処理に必要なエネルギーを求める場合は、エネルギー効率と処理データ数を用いて、$128/407.2 = 0.3144$ となるため、0.3144 μJ/bit（2.51 μJ/byte）と算出できる。Großらによる研究 [19] で報告されているエネルギー消費量と比

表 3.36　Elsadek らによる Ascon の ASIC 実装評価結果 [14]

名称	インタフェース	データ量	面積コスト	スループット @f_{max}	エネルギー効率
Ascon-128 (Enc.)	no	Short/PT	11 kGE	39 Mbps	407.2 Mbit/mJ
		Short/AD		37 Mbps	371.7 Mbit/mJ
		Short/PT+AD		70 Mbps	640.8 Mbit/mJ
Ascon-128 (Enc.)	no	Long/PT	11 kGE	522 Mbps	2,614.7 Mbit/mJ
		Long/AD		522 Mbps	2,531.3 Mbit/mJ
		Long/PT+AD		531 Mbps	2,600.4 Mbit/mJ

べると大きな違いがあることがわかる。どちらの性能評価も合成結果後のシミュレーションによる見積もりであることが原因として考えられる。電力やエネルギー効率の正確な測定には、レイアウト後の正確なシミュレーションや実チップでの測定が不可欠と言える。

上記の式から、エネルギー効率（1 ジュールのエネルギーで処理できるビット数）を高めるためには、スループット性能を高めるか消費電力を抑制することで達成できると言える。また、スループット性能を高めるためには電力が必要であることから、最適なエネルギー効率はスループット性能と消費電力のトレードオフで決まる。設計段階においては、Unrolled 実装型アーキテクチャでのトレードオフの模索が効果的であり、クロック周波数や供給電力を変更することによるエネルギー効率の最適化も有効な手段となる。

3.3.2　ソフトウェア実装性能

本節では、Ascon のソフトウェア実装性能について、2022 年度に公開された CRYPTREC 外部評価報告書 [53] に基づき、2022 年 9 月現在の調査結果を掲載する。

3.3.2.1　調査対象と性能評価環境

CAESAR コンペティションの最終的なポートフォリオや NIST LWC ファイナリストなどを対象としたソフトウェア実装性能について、eBACS (ECRYPT Benchmarking of Cryptographic Systems)[*3]で幅広い評価結果がまとめられている。ただし、eBACS では Intel Xeon や Arm Cortex-M7 のような処理性能の高い CPU 上での評価結果を中心としており、IoT デバイス向けの低消費電力 CPU 上での評価結果は掲載されていない。本ガイドラインでは、IoT デバイス向けの低消費電力 CPU 上で Ascon のソフトウェア実装性能を評価した 2 つの文献 [21, 48] における評価結果を紹介する。

一般的に、ソフトウェア実装における処理性能は、1 バイトのデータを処理するために必要なサイクル数（cycles/byte）で評価する方法とレイテンシで評価する方法がある。認証暗号やハッシュ関数の処理では、初期化処理などのオーバヘッド時間が必要となり、データ長が短い場合には処理性能が低くなる傾向にある。このため、少ないデータ量に対して暗号化処理を行うようなアプリケーションを対象としてソフトウェア実装性能を評価する場合には、サイクル数を評価するよりもレイテンシを評価する方が適している場合が多い。

一方で、十分な量のデータ量に対して暗号化処理を行う場合にはオーバーヘッドを無視することができるため、必要サイクル数の測定により対象となる暗号アルゴリズムの最適な処理性能を取得することができる。つまり、この場合にはスループット性能が重視されるべきであるため、サイクル数を評価することが適していると言える。

3.3.2.2　実装性能

表 3.37 は文献 [21] で報告された Hira らによる Arm Cortex-M0 上でのレイテンシ評価結果をまとめたものである。Hira らの研究では、Ascon 設計チームが提出したリファレンスコードを Arm Cortex-M0 に移植し、レイテンシ、ROM サイズ、そしてコードサイズを評価している。CPU の動作周波数は 48 MHz である。

[*3] https://bench.cr.yp.to/

測定では、関連データと平文を 0 バイトから 32 バイトまで変化させ、暗号化と復号にかかるレイテンシを分けて評価されている。なお、測定に使用したテストベクトルは、関連データと平文をそれぞれ 2 バイトずつ変化させ、$17 \times 17 = 289$ 通りの組み合わせで構成されている。

表 3.37　Hira らによる Arm Cortex-M0 上での Ascon のレイテンシ評価結果 [21]

名称	暗号化	復号	ROM サイズ	コードサイズ
Ascon-128a	153 msec (0.529 msec)	155 msec (0.536 msec)	30.6 Kbytes	28.6 Kbytes
Ascon-128	183 msec (0.633 msec)	185 msec (0.640 msec)	31.4 Kbytes	29.4 Kbytes
Ascon-80pq	185 msec (0.640 msec)	188 msec (0.650 msec)	31.3 Kbytes	29.3 Kbytes

表中の結果は、289 通りのテストベクトル全ての処理にかかるレイテンシの総和を表している。また、括弧内の数値は、1 つのテストベクトルを処理するために必要となるレイテンシの平均値を表している。

表 3.38 は文献 [48] で報告された Watanabe らによる Arm Cortex-M3 上と AVR ATmega 上でのレイテンシ評価結果をまとめたものである。Watanabe らの研究では、16 バイトの関連データと 16 バイトの平文に対して、暗号化と復号にかかるレイテンシを分けて評価している。CPU の動作周波数は、Arm Cortex-M3 が 84 MHz、AVR ATmega が 16 MHz である。

表 3.38　Watanabe らによる Arm Cortex-M3 上と AVR ATmega 上での Ascon のレイテンシ評価結果 [48]

名称	プラットフォーム	レイテンシ	ROM サイズ	RAM サイズ
Ascon-128 (暗号化)	AVR ATmega @16 Mhz	5.84 msec	9,732 bytes	157 bytes
Ascon-128 (復号)		5.86 msec		181 bytes
Ascon-128 (暗号化)	Arm Cortex-M3 @84 MHz	0.30 msec	4,764 bytes	196 bytes
Ascon-128 (復号)		0.31 msec		121 bytes

テストベクトルや動作周波数が異なるものの、いずれの結果でも数十バイト程度のデータであれば、数 msec 程度でのレイテンシで暗号化処理が可能であることがわかる。また、文献 [48] ではコードサイズが最適化されていることが読み取れる。実装性能をさらに向上させるためには、アルゴリズムの特徴を理解し、CPU に合わせて最適化を図る必要がある。

3.3.3　物理攻撃耐性

本節では、Ascon-128 の物理攻撃耐性を含めた実装性能について、2023 年度に公開された CRYPTREC 外部評価報告書 [54] に基づき、2023 年 9 月現在の調査結果を掲載する。

3.3.3.1　用語

本節で取り扱うサイドチャネル攻撃対策手法とサイドチャネル解析・漏えい評価手法に関する用語を表 3.39 で示す。詳細は付録 A.1 と付録 A.2 を参照されたい。

3.3.3.2　サイドチャネル攻撃対策が施された実装への評価結果

本節では、Kandi らによる Threshold Implementation (TI) を使用した評価結果 [25] と Groß による Domain Oriented Masking (DOM) を使用した評価結果 [17] を紹介する。

表 3.39　3.3.3 節で取り扱う用語

用語	説明	詳細
Threshold Implementation (TI)	2006 年に Nikova らによって提案された秘密分散法に基づくマスキング手法 [38, 39]	付録 A.1.1
Domain Oriented Masking (DOM)	2016 年に Großらによって提案された d 次のプロービングモデルに対して耐性のあるマスキング手法 [17, 18]	付録 A.1.2
相関電力解析	電力のサイドチャネル情報を効率よく解析する手法 [9]、電磁波サイドチャネルに対する解析手法は相関電磁波解析と呼ばれる。	付録 A.2.1
故障利用攻撃	暗号機能を実装したハードウェアの動作中に故意に故障を起こし、故障によって生じた計算誤りを利用して解析を行う手法 [7]	付録 A.2.2
Test Vector Leakage Assessment (TVLA)	サイドチャネルからの漏洩評価における統計的手法、ウェルチの t 検定（Welch's t-test）が利用される。	付録 A.2.3
テンプレート攻撃	事前に攻撃対象モジュールの特性を評価したテンプレートを準備し、このテンプレートを使用してパラメータを操作できない攻撃対象モジュールの秘密鍵を推定する手法	付録 A.2.4

■Kandi らによる TI を使用したサイドチャネル攻撃対策と三重化による故障利用攻撃対策 [25]　2023 年 6 月、Kandi らは Ascon のハードウェア実装性能に関する評価結果を報告した [25]。具体的には、サイドチャネル攻撃対策が施されていない実装に加え、TI を使用したサイドチャネル攻撃対策と計算の三重化による故障利用攻撃対策が施された実装への評価結果が紹介されている。サイドチャネル攻撃対策と故障利用攻撃対策は互いに独立した概念に基づき実装されることから、それぞれの対策が相互に影響しないと言われている。つまり、要求仕様に応じて、いずれかの対策を施して実装することも、両方の対策を施して実装することも可能である。

最初に、TI を使用したサイドチャネル攻撃対策が施された実装への評価結果を紹介する。表 3.40 と表 3.41 は、それぞれ Ascon の暗号化処理と復号処理に関する FPGA 実装と ASIC 実装の評価結果をまとめている。なお、括弧内の数値は、対策を施していない実装における暗号化とタグ生成の評価結果を基準とした割合を表している。FPGA 実装では 28nm テクノロジを有する Kintex-7 が使用されている。表 3.40 から、暗号化処理と復号処理での実装性能の違いはほとんど見られないことがわかる。3 シェア TI を使用した実装では、対策を施していない実装と比べ、暗号化処理と復号処理のいずれの回路サイズも 4 倍以上の LUT を必要としている。一方、クロック周期については、10% 程度の増加に抑えることができる。

表 3.40　Kandi らによる Kintex-7 上での Ascon の FPGA 実装評価結果

コア	面積コスト [LUT]	クロック周期 [psec]
暗号化とタグ生成	944 (1.00)	5,525 (1.00)
復号とタグ検証	1,058 (1.12)	5,525 (1.00)
暗号化とタグ生成（3 シェア TI）	3,977 (4.21)	6,024 (1.09)
復号とタグ検証（3 シェア TI）	3,795 (4.02)	6,010 (1.09)

ASIC 実装では STM 130nm ライブラリが使用され、TI で保護された Ascon S-box のゲートサイズが 56 gates、線形層のゲートサイズが 320 gates となっている。このことから、1 サイクルで 1 ラウンドを処理する回路において、組合せのゲートサイズは少なくとも 12.6 Kgates 程度の面積コストが必要となる。なお、実際の ASIC 実装においては、内部状態を保持するフリップフロップ回路、インタフェース回路、乱数生成器が必要となる。また、表 3.41 から、FPGA と同様に、暗号化処理と復号処理での違いがほとんど見られないことがわかる。

文献 [25] において 3 シェア TI を採用した理由は、Ascon S-box の代数次数が 2 であり、TI のシェアの数が Ascon S-box

表 3.41　Kandi らによる STM 130nm 上での Ascon の ASIC 実装評価結果（サイドチャネル攻撃対策との比較結果）

コア	面積コスト [μm^2]	クリティカルパス遅延時間 [psec]
暗号化とタグ生成	73,803 (1.00)	8,595 (1.00)
復号とタグ検証	71,873 (0.97)	8,586 (1.00)
暗号化とタグ生成（3 シェア TI）	273,857 (3.71)	10,001 (1.16)
復号とタグ検証（3 シェア TI）	274,688 (3.72)	9,981 (1.16)

表 3.42　Kandi らによる STM 130nm 上での Ascon の ASIC 実装評価結果（サイドチャネル攻撃対策、故障利用攻撃対策との比較結果）

サイドチャネル攻撃対策	故障利用攻撃対策	面積コスト [μm^2]	クリティカルパス遅延時間 [psec]
未対策	未対策	98,524 (1.00)	8,520 (1.00)
未対策	三重化	258,224 (2.62)	8,518 (1.00)
3 シェア TI	未対策	364,320 (3.70)	9,830 (1.15)
3 シェア TI	三重化	948,544 (9.63)	9,832 (1.15)

の代数次数に 1 を加えた 3 である必要があるためである。Ascon S-box に対する 3 シェア TI の構成方法の詳細は、文献 [25] の 4.3 節、または文献 [54] の 4.5.4 節を参照されたい。

次に、三重化による故障利用攻撃対策が施された実装への評価結果を紹介する。差分故障解析への有効な対策の 1 つとして暗号化処理の二重化を紹介したが、文献 [25] では暗号化処理の三重化、つまり同じ暗号化処理を 3 回行い、3 つの出力結果が全て異なる場合には乱数を出力するという対策を提案した。

暗号化処理の三重化により、Ascon のコア部分における面積コストは単純に 3 倍となる。より正確には、多数決により出力結果を決定する処理が追加されるため、3 倍よりも大きくなる。表 3.42 は、STM 130nm 上での Ascon の ASIC 実装に関し、対策が施されていない場合、三重化による対策を施した場合、3 シェア TI による対策を施した場合、3 シェア TI による対策と三重化による対策の両方を施した場合における性能評価（面積コスト、クリティカルパス遅延時間）をまとめている。この表からわかるように、インタフェースなどの面積コストが増えないため、全体としては面積コストは 3 倍弱の増加となっている。また、空間的な三重化を施しているため、クリティカル遅延時間への影響はない。

■Großによる DOM を使用したサイドチャネル攻撃対策 [17]　2018 年 6 月、Großは自身の学位論文 [17] で DOM によるサイドチャネル攻撃対策を施した Ascon の実装性能評価結果を体系的にまとめている。本学位論文では、DOM のバリエーションである Unified Masking (UMA) と Low-Latency Masking (LOLA) も提案されている。UMA は、暗号アルゴリズムのデータパスにレジスタを追加することで、安全性の観点でクリティカルとされるデータを適切に制御し、DOM の乱数コストの削減を目指したものである。レジスタを追加することから 1 ラウンドの処理に必要となるサイクル数が増加するため、レイテンシは増加しスループットは低下するものの、必要となるフレッシュな乱数は少なくて済む。UMA とは対照的に、LOLA ではレジスタによるステージ数を減らし、処理パフォーマンスの向上を目指したものである。代わりに、非線形処理におけるシェア数が増加するため、より多くのデータの冗長性や追加の回路が必要となり、乱数コストも増加する。

表 3.43 は、DOM を使用したサイドチャネル攻撃対策を施した ASIC 実装の評価結果をまとめたものである。ここで、1 次（又は 5 次）安全な DOM、UMA、LOLA とは、それぞれ 1 次（又は 5 次）プロービングモデルに耐性のある DOM、UMA、LOLA を使用した実装のことを意味する。本実装では UMC-90nm Low-K の CMOS ライブラリが使用されている。

UMA については、レイテンシとスループット性能を犠牲にすることで、乱数コストを抑えられることがわかる。ただし、現実的な実装となる 1 次 UMA は、1 次 DOM と比べて面積コストと乱数コストがほぼ同じである。シェア数が少ない場合には、UMA の実装コスト低下は限定的であると言える。5 次 UMA では、5 次 DOM と比べて必要となるフレッシュな乱数のコストを削減することに成功しているが、レジスタの追加などによって面積コストが増加してしまう。

LOLA 実装については、1 ラウンドの処理を 1 サイクルで実行可能となるため、低レイテンシが実現できていることがわかる。ただし、面積コストは DOM や UMA と比べて大きくなり、必要となる乱数のコストが多い 5 次 LOLA では、1 サイ

表 3.43　Großによる UMC-90nm Low-K 上での Ascon の ASIC 実装評価結果（DOM、UMA、LOLA との比較結果）

デザイン	面積コスト [KGE]	サイクル数 [cycle/round]	スループット性能 [Mbps]	乱数コスト [bit/cycle]
1 次安全な DOM	28.89	3	2,250	320
1 次安全な UMA	27.18	3	2,250	320
1 次安全な LOLA	42.75	1	2,770	2,048
5 次安全な DOM	161.87	3	1,860	4,800
5 次安全な UMA	220.01	7	850	3,520
5 次安全な LOLA	339.82	1	2,990	18,432

クル当たり約 18K ビットと非常に多くのフレッシュな乱数を必要としている。

通常の TI と比べて少ないシェア数でサイドチャネル攻撃対策を実現できる DOM は、設計手法としても興味深いものとなっている。DOM とその 2 つのバリエーションにより、実装コスト、処理パフォーマンス、そして必要となる乱数コストのトレードオフは大幅に広がっている。また、設計者の選択肢が増えたことに加え、設計手法自体が規則的、かつ汎用的なマスキングツールで対策を実現できることは、生産性の向上に繋がると考えられる。

3.3.3.3　物理攻撃耐性評価

本節では、Samwel らによる相関電力解析の評価結果 [42]、Betina らによる相関電力解析と TVLA の評価結果 [5]、そして Mohajerani らによる相関電力解析と TVLA の評価結果 [34] を紹介する。

その他の最新動向として、Gigerl らによる TVLA の評価結果 [16]、Liu らによるサイドチャネル情報漏洩の評価結果 [30]、そして You らによるテンプレート攻撃の評価結果 [52] が報告されている。これらの評価結果について本ガイドラインでは取り扱わないため、詳細は文献 [54] を参照されたい。

■Samwel らによる相関電力解析 [42]　2017 年 5 月、Samwel らは Ascon のハードウェア実装に対して相関電力解析を実施した結果を初めて報告した [42]。具体的には、Ascon の非線形処理である S-box の出力に対して効率の良い選択関数を提案し、この選択関数を使用してサイドチャネル攻撃対策を施していない FPGA 実装と 3 シェア TI によるサイドチャネル攻撃対策を施した FPGA 実装に対して、相関電力攻撃に成功したと報告されている。選択関数の構成方法の詳細については、文献 [42] の 5.1 節、または文献 [54] の 4.1.3 節を参照されたい。

Samwel らは、サイドチャネル攻撃対策を施していない Ascon を SAKURA-G 上に搭載された FPGA Spartan-6 に実装し、この実装に対して 50K 個の波形トレース[*4]から全ての秘密鍵ビットの導出に成功したと報告している。これにより、提案された選択関数による電力モデルが効果的であることが明らかとなった。また、3 シェア TI によるサイドチャネル攻撃対策を施した Ascon に対して、シミュレーションにて同様の相関電力攻撃を実行し、900K 個の波形トレースで全ての秘密鍵ビットの導出に成功したと報告している。

■Betina らによる相関電力解析と TVLA [5]　2022 年 8 月、Betina らは Ascon のソフトウェア実装に対する相関電力解析と TVLA を用いた安全性評価の結果を報告した [5]。Betina らは、Ascon の設計者チームが公開している Ascon-128 のソースコードを Arm-V6 上に実装し、サイドチャネル情報として電力波形を使用した。電力測定には、Riscure 社の Piñata development board を使用している。当該ボードには、32 ビットの Arm マイクロコントローラをベースとする SoC STM32F407IGT6 が搭載されており、その動作周波数は 168 MHz である。電力波形の取得には、Riscure 社のカレントプローブ（型番不明）と Picoscope 社のオシロスコープ（model 3206D）を使用している。なお、本報告での相関電力解析では、Samwel ら [42] が提案した選択関数を使用している。

最初に、Betina らは 50K 個の波形トレースを使用し、サイドチャネル攻撃対策を施していない Ascon の暗号化処理に対して TVLA を行った。その結果、Ascon の初期化処理フェーズにおいて t 値が閾値を大きく超えていることが示された。次

[*4] オシロスコープ等で取得した物理情報の時系列変化の軌跡を波形トレース、あるいは単にトレースと呼ぶ。波形トレースの単位として用いることもある。

に、Ascon の初期化処理フェーズに特化し、Samwel ら [42] の攻撃手法に従い、100K 個の波形トレースを使用した相関電力攻撃を実施して正しい秘密鍵の復元に成功したことが示された。なお、2 つある選択関数の使用において、攻撃の成功確率に差が生じることが明らかとなった。

Betina らはサイドチャネル攻撃（マスキング）対策を施した Ascon に対しても同様に TVLA と相関電力攻撃を実施した。このソフトウェア実装では、乱数をほとんど使用しない 2〜4 個のシェアで対策が施されている。最初に、15K 個の波形トレースを使用し、Ascon の初期化処理フェーズの最初で処理される Ascon permutation に対して、相関電力解析で使用する攻撃箇所（サンプル時間）の特定が行われた。この際、ナンスはランダムに変化させ、その他のパラメータは全て固定としている。その後、15K 個の波形トレースを使用して相関電力解析を実施した結果、暗号化処理の 2 個の中間値を利用する 2 次相関電力解析でも攻撃は成功しないことが示された。その原因は、使用した波形トレースの数が少なかったことにあると考察されている。

■Mohajerani らによる相関電力解析と TVLA [34]　2023 年 6 月、Mohajerani らは NIST LWC ファイナリスト 10 方式に対する物理耐性評価を行う機関を集め、サイドチャネル攻撃耐性に関する調査やベンチマーク評価行った結果を報告するとともに、物理攻撃耐性に関する一般的な評価フレームワークを提案した [34]。本研究プロジェクトに参画した大学、研究所、そして企業は以下に示す 7 つの機関である。

1. IAIK, Graz University of Technology, Austria
2. CCSL, Shanghai Jiao Tong University, China
3. HSCP Lab, Tsinghua University, Beijing, China
4. Secure-IC, France
5. CERG, George Mason University, USA
6. Ruhr-Universitat Bochum, Germany
7. CESCA Lab, Radboud University, the Netherlands

また、サイドチャネル攻撃対策技術の安全性を評価するとともに、対策技術を追加することによって実装コストと処理パフォーマンスに与える影響について実証実験を行っている。

表 3.44 は、サイドチャネル攻撃対策を施した Ascon の FPGA 実装に対する安全性評価結果をまとめたものである。ソースコード Ascon-128_Graz_d1 の実装では DOM [17] によるサイドチャネル攻撃対策が施されており、ソースコード Ascon-128_Bochum_d1 の実装ではサイドチャネル攻撃対策が施されていない Ascon-128_Graz_x1 のソースコードをベースとしてマスキング対策が施されている。また、サイドチャネル攻撃対策を施していない HDL (Hardware Description Language) コードから乱数マスキング対策を施した HDL コードを半自動生成するために、AGEMA [27] と呼ばれるツールが使用されている[*5]。

サイドチャネル攻撃対策を施した Ascon の FPGA 実装に対する安全性評価結果では、サイドチャネル情報として電力と電磁波が使用され、これらのサイドチャネル情報に対して TVLA、χ^2 検定、そして相関電力解析による情報漏洩の可能性の有無が解析された。波形トレース数は約 100 万から 1000 万程度である。CERG による評価のみ TVLA の閾値である 4.5σ を超えたと報告されているが、他の機関からは情報漏洩の可能性がないと報告されている。CERG による Ascon-128_Graz_d1 のテストでは、数個（3〜10）のサンプルで閾値の 4.5σ を超えたものの、これらのテストでは攻撃対象のクロックと同期したサンプリングクロックを使用していたことが原因であると考察されている。

表 3.45 は、サイドチャネル攻撃対策を施した Ascon のソフトウェア実装に対する安全性評価結果をまとめたものである。全て Arm Cortex-M4 上で実装されたものであり、ソースコードは Ascon-128_Graz_d1 と Ascon-128_Graz_d2 が使用されている。

いずれの安全性評価においても、電磁波に関するサイドチャネル情報から取得した波形データが使用されている。評価の結果、相関電力解析による秘密鍵の復元には成功していない。CESCA グループによる 2 次の相関電力解析では、15M 個の波形トレースを使用しても Ascon-128_Graz_d1 の秘密鍵に関する情報を明らかにできないと報告されている。参考までに、

[*5] AGEMA はサイドチャネル攻撃に対して保護する必要があるワイヤとゲートを特定し、これらに対して必要な乱数マスキング対策を施すことが可能であるものの、制御ロジックに対して乱数マスキング対策を施すことができないため、一部のコードに対しては手動でマスキング対策を施す必要がある。これが半自動生成ツールと呼ばれる理由である。

表 3.44　サイドチャネル攻撃対策を施した Ascon の FPGA 実装に対する安全性評価結果

ソースコード（評価機関）	プラットフォーム	オシロスコープ	評価手法（サイドチャネル）	波形数（トレース）	評価結果
Ascon-128_Bochum_d1（CERG）	Artix-7 (CW305)	FOBOS3 ADC	TVLA（電力）	10M	リーク有 (1.5 M トレース)
Ascon-128_Bochum_d1（IAIK）	Artix-7 (CW305)	PicoScope 6404C	TVLA（電力）	10M	リーク無
Ascon-128_Bochum_d1（CCSL）	Kintex-7 (SAKURA-X)	LeCroy 610Zi	TVLA（電磁波）	1M	リーク無
Ascon-128_Bochum_d1（CCSL）	Kintex-7 (SAKURA-X)	LeCroy 610Zi	χ^2 検定（電磁波）	1M	リーク無
Ascon-128_Bochum_d1（CCSL）	Kintex-7 (SAKURA-X)	LeCroy 610Zi	相関電力解析（電磁波）	11M	リーク無
Ascon-128_Graz_d1（HSCP）	Spartan-6 (SAKURA-G)	WaveRunner 8404M	TVLA（電力）	10M	リーク無

表 3.45　サイドチャネル攻撃対策を施した Ascon のソフトウェア実装に対する安全性評価結果

ソースコード（評価機関）	評価プラットフォーム	オシロスコープ	評価手法（サイドチャネル）	波形数（トレース）	評価結果
Ascon-128_Graz_d1（CESCA）	Arm Cortex-M4 (STM32F407)	Pico 3206D	2 次相関電力解析（電磁波）	15M	リーク無
Ascon-128_Graz_d2（CCSL）	Arm Cortex-M4 (STM32F303)	Pico 3203D	TVLA（電磁波）	60K	リーク無
Ascon-128_Graz_d2（CCSL）	Arm Cortex-M4 (STM32F303)	Pico 3203D	χ^2 検定（電磁波）	60K	リーク無
Ascon-128_Graz_d2（CCSL）	Arm Cortex-M4 (STM32F303)	Pico 3203D	相関電力解析（電磁波）	60K	リーク無

サイドチャネル攻撃対策を施していない Ascon のソフトウェア実装に対する相関電力解析では 500K 個の波形トレースを使用して秘密鍵の復元に成功している。このことから、使用したソースコードのマスキング対策が正常に機能していることがわかる。

Mohajerani らは、サイドチャネル攻撃対策を施した FPGA 実装における面積コストと処理パフォーマンスへの影響についても考察している。Ascon-128_Bochum_d1 における FPGA 実装では、サイドチャネル攻撃対策を施すことにより、面積コストが約 3 倍程度増加し、スループット性能が約 1/3 倍に低下していることが報告されている。一方、Ascon-128_Graz_d1 における FPGA 実装では、サイドチャネル攻撃対策を施すことにより、面積コストの増加が約 2 倍程度と Ascon-128_Bochum_d1 に比べて少なく、スループット性能も Ascon-128_Bochum_d1 ほど低下していないことが報告されている。これは、DOM を人手で実装したことにより、効率の良い対策技術が実現できたものと考えられる。なお、ソフトウェア実装に関する実験結果は文献 [34] には記載されていない。

参考文献

[1] Kazumaro Aoki, Tetsuya Ichikawa, Masayuki Kanda, Mitsuru Matsui, Shiho Moriai, and Junko Nakajima Toshio Tokita. Specification of Camellia - a 128-bit Block Cipher, 2001. https://info.isl.ntt.co.jp/crypt/camellia/dl/01espec.pdf（2023-10-07 閲覧）.

[2] ARM. AMBA 3 APB Protocol Specification v2.0, 2008. https://infocenter.arm.com/help/index.jsp?topic=/com.arm.doc.ihi0024c/index.html（2023-10-07 閲覧）.

[3] Subhadeep Banik, Andrey Bogdanov, Takanori Isobe, Kyoji Shibutani, Harunaga Hiwatari, Toru Akishita, and Francesco 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*, pages 411–436, 2015.

[4] William C. Barker and Elaine Barker. Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher, January 2012. https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-67r1.pdf.

[5] Lejla Batina, Ileana Buhan, Lukasz Chmielewski, Ellen Gunnarsdóttir, Vahid Jahandideh, Tom Stock, and Léo Weissbart. Side-Channel Evaluation Report on Implementations of Several NIST LWC Finalists, 2022. Nijmegen : Cryptographic Engineering & Side-Channel Analysis (CESCA) Lab, https://github.com/rweather/lwc-finalists/tree/5d2b22c9ff7744be429cabda0c078ea5b7b6f79e（2023-10-07 閲覧）.

[6] Ray Beaulieu, Douglas Shors, Jason Smith, Stefan Treatman-Clark, Bryan Weeks, and Louis Wingers. SIMON and SPECK: block ciphers for the internet of things. *IACR Cryptology ePrint Archive*, 2015:585, 2015.

[7] Eli Biham and Adi Shamir. Differential Fault Analysis of Secret Key Cryptosystems. In Burton S. Kaliski Jr., editor, *Advances in Cryptology - CRYPTO '97, 17th Annual International Cryptology Conference, Santa Barbara, California, USA, August 17-21, 1997, Proceedings*, volume 1294 of *Lecture Notes in Computer Science*, pages 513–525. Springer, 1997.

[8] Julia Borghoff, Anne Canteaut, Tim Güneysu, Elif Bilge Kavun, Miroslav Knezevic, Lars R. Knudsen, Gregor Leander, Ventzislav Nikov, Christof Paar, Christian Rechberger, Peter Rombouts, Søren S. Thomsen, and Tolga 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*, pages 208–225, 2012.

[9] Eric Brier, Christophe Clavier, and Francis Olivier. Correlation Power Analysis with a Leakage Model. In Marc Joye and Jean-Jacques Quisquater, editors, *Cryptographic Hardware and Embedded Systems - CHES 2004: 6th International Workshop Cambridge, MA, USA, August 11-13, 2004. Proceedings*, volume 3156 of *Lecture Notes in Computer Science*, pages 16–29. Springer, 2004.

[10] Renesas Electronics Corporation. RL78 ファミリ. https://japan.renesas.com/products/mpumcu/rl78/index.jsp?campaign=tb_prod（2023-10-07 閲覧）.

[11] Joan Daemen, Seth Hoffert, Michaël Peeters, Gilles Van Assche, Ronny Van Keer, and Silvia Mella. Ketje. https://ketje.noekeon.org/（2023-10-07 閲覧）.

[12] Christoph Dobraunig, Maria Eichlseder, Florian Mendel, and Martin Schläffer. Ascon v1.2. https://competitions.cr.yp.to/round3/asconv12.pdf（2023-10-07 閲覧）.

[13] Morris Dworkin. NIST SP800-38D – Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC, August 2015. https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38d.pdf.

[14] Islam Elsadek, Sohrab Aftabjahani, Doug Gardner, Erik MacLean, John Ross Wallrabenstein, and Eslam Yahya Tawfik. Hardware and Energy Efficiency Evaluation of NIST Lightweight Cryptography Standardization Finalists. In *IEEE International Symposium on Circuits and Systems, ISCAS 2022, Austin, TX, USA, May 27 - June 1, 2022*, pages 133–137. IEEE, 2022.

[15] Farnoud Farahmand, William Diehl, Abubakr Abdulgadir, Jens-Peter Kaps, and Kris Gaj. Improved Lightweight Implementations of CAESAR Authenticated Ciphers. In *26th IEEE Annual International Symposium on Field-Programmable Custom Computing Machines, FCCM 2018, Boulder, CO, USA, April 29 - May 1, 2018*, pages 29–36. IEEE Computer Society, 2018.

[16] Barbara Gigerl, Florian Mendel, Martin Schläffer, and Robert Primas. Efficient Second-Order Masked Software Implementations of Ascon in Theory and Practice, 2023. NIST, Lightweight Cryptography Workshop 2023. https://csrc.nist.gov/csrc/media/Events/2023/lightweight-cryptography-workshop-2023/documents/accepted-papers/04-efficient-second-order-masked-software.pdf.

[17] Hannes Groß. Domain-Oriented Masking–Generically Masked Hardware Implementations, 2018. PhD Thesis, IAIK, Graz University of Technology. https://diglib.tugraz.at/download.php?id=5c80ea0c43a56&location=browse（2023-10-07 閲覧）.

[18] Hannes Groß, Stefan Mangard, and Thomas Korak. Domain-Oriented Masking: Compact Masked Hardware Implementations with Arbitrary Protection Order. *IACR Cryptol. ePrint Arch.*, page 486, 2016.

[19] Hannes Groß, Erich Wenger, Christoph Dobraunig, and Christoph Ehrenhöfer. Suit up! - Made-to-Measure Hardware Implementations of Ascon. In *2015 Euromicro Conference on Digital System Design, DSD 2015, Madeira, Portugal, August 26-28, 2015*, pages 645–652. IEEE Computer Society, 2015.

[20] Jian Guo, Thomas Peyrin, Axel Poschmann, and Matthew 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*, pages 326–341, 2011.

[21] Ryota Hira, Tomoaki Kitahara, Daiki Miyahara, Yuko Hara-Azumi, Yang Li, and Kazuo Sakiyama. Software Evaluation for Second Round Candidates in NIST Lightweight Cryptography. *IACR Cryptol. ePrint Arch.*, page 591, 2022.

[22] Ekawat Homsirikamol, William Diehl, Ahmed Ferozpuri, Farnoud Farahmand, Panasayya Yalla, Jens-Peter Kaps, and Kris Gaj. CAESAR Hardware API. *IACR Cryptol. ePrint Arch.*, page 626, 2016.

[23] Tetsu Iwata, Kazuhiko Minematsu, Jian Guo, and Sumio Morioka. CLOC and SILC—Authenticated Encryption Schemes for Constrained Devices, 2014. https://www.nuee.nagoya-u.ac.jp/labs/tiwata/AE/（2023-10-07 閲覧）.

[24] Tetsu Iwata, Kazuhiko Minematsu, Jian Guo, and Sumio Morioka. CLOC: authenticated encryption for short input. In *Fast Software Encryption - 21st International Workshop, FSE 2014, London, UK, March 3-5, 2014. Revised Selected Papers*, pages 149–167, 2014.

[25] Aneesh Kandi, Anubhab Baksi, Tomas Gerlich, Sylvain Guilley, Peizhou Gan, Jakub Breier, Anupam Chattopadhyay, Ritu Ranjan Shrivastwa, Zdenek Martinasek, and Shivam Bhasin. Hardware Implementation of Ascon, 2023. NIST, Lightweight Cryptography Workshop 2023. https://csrc.nist.gov/csrc/media/Events/2023/lightweight-cryptography-workshop-2023/documents/accepted-papers/07-hardware-implementation-of-ascon.pdf.

[26] Jens-Peter Kaps, William Diehl, Michael Tempelmeier, Ekawat Homsirikamol, and Kris Gaj. Hardware API for Lightweight Cryptography, 2019. https://cryptography.gmu.edu/athena/LWC/LWC_HW_API.pdf（2023-10-07 閲覧）.

[27] David Knichel, Pascal Sasdrich, and Amir Moradi. Generic Hardware Private Circuits Towards Automated Generation of Composable Secure Gadgets. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2022(1):323–344, 2022.

[28] Lars R. Knudsen and Gregor Leander. PRESENT - block cipher. In *Encyclopedia of Cryptography and Security, 2nd Ed.*, pages 953–955. 2011.

[29] Ted Krovetz and Phillip Rogaway. OCB (v1.1). `https://competitions.cr.yp.to/round3/ocbv11.pdf`（2023-10-07 閲覧）.

[30] Zhenyuan Liu and Patrick Schaumont. Root-cause Analysis of the Side Channel Leakage from Ascon Implementations, 2023. NIST, Lightweight Cryptography Workshop 2023. `https://csrc.nist.gov/csrc/media/Events/2023/lightweight-cryptography-workshop-2023/documents/accepted-papers/13-root-cause-analysis-of-side-channel-leakage.pdf`.

[31] Mitsuru Matsui and Yumiko Murakami. Minimalism of software implementation - extensive performance analysis of symmetric primitives on the RL78 microcontroller. In *Fast Software Encryption - 20th International Workshop, FSE 2013, Singapore, March 11-13, 2013. Revised Selected Papers*, pages 393–409, 2013.

[32] Mitsuru Matsui and Yumiko Murakami. AES smaller than s-box - minimalism in software design on low end microcontrollers. In *Lightweight Cryptography for Security and Privacy - Third International Workshop, LightSec 2014, Istanbul, Turkey, September 1-2, 2014, Revised Selected Papers*, pages 51–66, 2014.

[33] Kazuhiko Minematsu. AES-OTR v1. `https://competitions.cr.yp.to/round1/aesotrv1.pdf`（2023-10-07 閲覧）.

[34] Kamyar Mohajerani, Luke Beckwith, Abubakr Abdulgadir, Eduardo Ferrufino, Jens-Peter Kaps, and Kris Gaj. SCA Evaluation and Benchmarking of Finalists in the NIST Lightweight Cryptography Standardization Process. *IACR Cryptol. ePrint Arch.*, page 484, 2023.

[35] Kamyar Mohajerani, Richard Haeussler, Rishub Nagpal, Farnoud Farahmand, Abubakr Abdulgadir, Jens-Peter Kaps, and Kris Gaj. FPGA Benchmarking of Round 2 Candidates in the NIST Lightweight Cryptography Standardization Process: Methodology, Metrics, Tools, and Results. *IACR Cryptol. ePrint Arch.*, page 1207, 2020.

[36] Amir Moradi, Axel Poschmann, San Ling, Christof Paar, and Huaxiong Wang. Pushing the limits: A very compact and a threshold implementation of AES. In *Advances in Cryptology - EUROCRYPT 2011 - 30th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tallinn, Estonia, May 15-19, 2011. Proceedings*, pages 69–88, 2011.

[37] National Institute of Standards and Technology. FIPS 197-4 – Secure Hash Standard (SHS), November 2001. `https://csrc.nist.gov/publications/fips/fips197/fips-197.pdf`.

[38] Svetla Nikova, Christian Rechberger, and Vincent Rijmen. Threshold Implementations Against Side-Channel Attacks and Glitches. In Peng Ning, Sihan Qing, and Ninghui Li, editors, *Information and Communications Security, 8th International Conference, ICICS 2006, Raleigh, NC, USA, December 4-7, 2006, Proceedings*, volume 4307 of *Lecture Notes in Computer Science*, pages 529–545. Springer, 2006.

[39] Svetla Nikova, Vincent Rijmen, and Martin Schläffer. Secure hardware implementation of nonlinear functions in the presence of glitches. *J. Cryptol.*, 24(2):292–321, 2011.

[40] Konstantinos Papagiannopoulos and Aram Verstegen. Speed and size-optimized implementations of the PRESENT cipher for tiny AVR devices. In *Radio Frequency Identification - Security and Privacy Issues 9th International Workshop, RFIDsec 2013, Graz, Austria, July 9-11, 2013, Revised Selected Papers*, pages 161–175, 2013.

[41] Behnaz Rezvani and William Diehl. Hardware Implementations of NIST Lightweight Cryptographic Candidates: A First Look. *IACR Cryptol. ePrint Arch.*, page 824, 2019.

[42] Niels Samwel and Joan Daemen. DPA on hardware implementations of Ascon and Keyak. In *Proceedings of the Computing Frontiers Conference, CF'17, Siena, Italy, May 15-17, 2017*, pages 415–424. ACM, 2017.

[43] Yu Sasaki, Yosuke Todo, Kazumaro Aoki, Yusuke Naito, Takeshi Sugawara, Yumiko Murakami, Mitsuru Matsui, and Shoichi Hirose. Minalpher. https://info.isl.ntt.co.jp/crypt/minalpher/index.html（2023-10-07 閲覧）.

[44] Kyoji Shibutani, Takanori Isobe, Harunaga Hiwatari, Atsushi Mitsuda, Toru Akishita, and Taizo Shirai. Piccolo: An ultra-lightweight blockcipher. In *Cryptographic Hardware and Embedded Systems - CHES 2011 - 13th International Workshop, Nara, Japan, September 28 - October 1, 2011. Proceedings*, pages 342–357, 2011.

[45] Taizo Shirai, Kyoji Shibutani, Toru Akishita, Shiho Moriai, and Tetsu Iwata. The 128-bit blockcipher CLEFIA (extended abstract). In *Fast Software Encryption, 14th International Workshop, FSE 2007, Luxembourg, Luxembourg, March 26-28, 2007, Revised Selected Papers*, pages 181–195, 2007.

[46] Tomoyasu Suzaki, Kazuhiko Minematsu, Sumio Morioka, and Eita Kobayashi. TWINE : A lightweight block cipher for multiple platforms. In *Selected Areas in Cryptography, 19th International Conference, SAC 2012, Windsor, ON, Canada, August 15-16, 2012, Revised Selected Papers*, pages 339–354, 2012.

[47] Michael Tempelmeier, Fabrizio De Santis, Georg Sigl, and Jens-Peter Kaps. The CAESAR-API in the real world - Towards a fair evaluation of hardware CAESAR candidates. In *2018 IEEE International Symposium on Hardware Oriented Security and Trust, HOST 2018, Washington, DC, USA, April 30 - May 4, 2018*, pages 73–80. IEEE Computer Society, 2018.

[48] Yuhei Watanabe, Hideki Yamamoto, and Hirotaka Yoshida. Performance Evaluation of NIST LWC Finalists on AVR ATmega and ARM Cortex-M3 Microcontrollers. *IACR Cryptol. ePrint Arch.*, page 1071, 2022.

[49] Hongjun Wu. ACORN v2. https://competitions.cr.yp.to/caesar-submissions.html/（2023-10-07 閲覧）.

[50] Hongjun Wu and Tao Huang. The JAMBU Lightweight Authentication Encryption Mode. https://competitions.cr.yp.to/round3/jambuv21.pdf（2023-10-07 閲覧）.

[51] Panasayya Yalla and Jens-Peter Kaps. Evaluation of the CAESAR hardware API for lightweight implementations. In *International Conference on ReConFigurable Computing and FPGAs, ReConFig 2017, Cancun, Mexico, December 4-6, 2017*, pages 1–6. IEEE, 2017.

[52] Shih-Chun You, Markus G. Kuhn, Sumanta Sarkar, and Feng Hao. Low Trace-Count Template Attacks on 32-bit Implementations of Ascon AEAD. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2023(4):344–366, 2023.

[53] 崎山一男. 軽量暗号の実装性能に関する調査及び評価（NIST 軽量暗号コンペティションファイナリスト）（文書番号：CRYPTREC EX-3205-2022）, 2022. https://www.cryptrec.go.jp/exreport/cryptrec-ex-3205-2022.pdf.

[54] 崎山一男. 軽量暗号 Ascon の実装性能に関する調査及び評価（文書番号：CRYPTREC EX-3301-2023）, 2023.

第 4 章

代表的な軽量暗号

4.1 ブロック暗号

本節では、主要な軽量ブロック暗号として CLEFIA、LED、Midori、Piccolo、PRESENT、PRINCE、SIMON、SPECK、TWINE の調査結果をまとめる。調査対象は、主要国際学会で発表されており、現段階で有力な攻撃法が発見されておらず、かつ十分な実装性能を持つと考えられるアルゴリズムとした。 また、CLEFIA、PRESENT、LEA が軽量ブロック暗号に関係する ISO/IEC（ISO/IEC 29192-2）[66] で規格化されている状況を鑑み、本稿では 2016 年度版ガイドライン [38] に掲載されていない LEA を新たな調査対象とし、その調査結果をまとめる。

各アルゴリズムのブロック長、鍵長といった基本入出力情報に加え、全体構造、および構成段数を記載している。鍵長やブロック長によって個別の名称が与えられているアルゴリズムについては、それぞれ個別の名称も記載した。アルゴリズムの特徴としては、主に提案論文で述べられている設計者らの主張を可能な限りそのまま記載した。

各アルゴリズムの安全性解析状況については、2021 年度に公開された CRYPTREC 外部評価報告書 [136] に基づき、2021 年 9 月時点の状況を記載している。文献 [136] は、2016 年度版ガイドライン [38] に掲載されている暗号アルゴリズムを中心とした代表的な軽量暗号の安全性評価に関する動向調査を行い、2021 年 9 月時点でこれらの軽量暗号に対し現実的な脅威に繋がる脆弱性が指摘されているか否かを明らかにしたものである。なお、新たに調査対象として追加した LEA について、文献 [136] では安全性解析状況だけでなく、仕様等（設計者、発表年、仕様参照先、特徴、主な実装評価結果、標準化状況）もまとめられているため、文献 [136] の記載内容に従って調査結果をまとめた。さらに、文献 [136] ではバイクリーク攻撃とその派生攻撃が提案された軽量ブロック暗号（Midori、Piccolo、PRESENT、TWINE が該当）に関し、これらの攻撃が提案された事実について記載されているものの、これらの攻撃を除いた解析手法の中から最大の攻撃可能段数を達成するものを最良の攻撃としてラベル付けされている。本ガイドラインにおいても文献 [136] の方針に従うものとする。

ハードウェア実装性能調査では、主に十分な評価が行われていると考えられる ASIC での実装性能評価を調査し、実装ゲート規模（GE）、1 ブロックの演算に必要なサイクル数（cycles/block）、および 100kHz におけるスループットを記載している。また、ソフトウェア実装性能調査では、ハイエンド CPU での実装結果として 1 バイトの処理に必要なサイクル数（cycles/byte）を記載し、ローエンド CPU での実装結果として cycles/byte に加えて ROM、RAM 使用量を記載した。

<table>
<tr><td>技術分野</td><td>ブロック暗号</td></tr>
<tr><td>名称</td><td>CLEFIA</td></tr>
<tr><td>設計者</td><td>Taizo Shirai[1], Kyoji Shibutani[1], Toru Akishita[1], Shiho Moriai[1], Tetsu Iwata[2]
(1: Sony Corporation/Japan, 2: Nagoya University/Japan)</td></tr>
<tr><td>発表年</td><td>2007 (FSE 2007 [109])</td></tr>
<tr><td>仕様参照先</td><td>FSE 2007 [109]、設計者ウェブサイト [37]</td></tr>
<tr><td>特徴</td><td>設計者らは、高い安全性を保ちつつ、ハードウェア、ソフトウェアの両実装形態で高い実装性能を持つと主張している。また、AES と同じインタフェースに対応している点も特長である。
<table>
<tr><td>全体構造</td><td colspan="3">4-line type-II 一般化 Feistel 型</td></tr>
<tr><td>ブロック長 [bit]</td><td colspan="3">128</td></tr>
<tr><td>鍵長 [bit]</td><td>128
(CLEFIA-128)</td><td>192
(CLEFIA-192)</td><td>256
(CLEFIA-256)</td></tr>
<tr><td>構成段数 [段]</td><td>18</td><td>22</td><td>26</td></tr>
</table></td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [23, 27, 81, 82, 92, 119, 125, 131] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。
最良の攻撃は、2015 年に提案された Li ら [81] による切り詰め差分攻撃であり、14 段に簡略化した CLEFIA-128、14 段に簡略化した CLEFIA-192、15 段に簡略化した CLEFIA-256 に対して、それぞれ秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果
<table>
<tr><td>Algorithm</td><td>Area
[GE]</td><td>Cycles/block</td><td>Throughput@100kHz
[kbps]</td><td>Ref.</td></tr>
<tr><td>CLEFIA-128 (Enc)</td><td>2,488</td><td>328</td><td>39.0</td><td>[4]</td></tr>
<tr><td>CLEFIA-128 (Enc/Dec)</td><td>2,604</td><td>328/320</td><td>39.0/40.0</td><td>[4]</td></tr>
<tr><td>CLEFIA-128 (Enc/Dec)</td><td>5,979</td><td>18</td><td>711.1</td><td>[109]</td></tr>
</table>
ソフトウェア実装評価結果 (ローエンド CPU)
<table>
<tr><td>Algorithm</td><td>ROM
[byte]</td><td>RAM
[byte]</td><td>Cycles/byte
[Enc/Dec]</td><td>Platform</td><td>Ref.</td></tr>
<tr><td>CLEFIA-128</td><td>1,309</td><td>78</td><td>39,357/152,023</td><td>RL78</td><td>[94]</td></tr>
<tr><td>CLEFIA-128</td><td>2,026</td><td>64</td><td>4,337/4,477</td><td>RL78</td><td>[94]</td></tr>
</table></td></tr>
<tr><td>標準化状況</td><td>ISO/IEC 29192-2 [66]、IETF RFC 6114 [74]</td></tr>
</table>

<table>
<tr><td>技術分野</td><td>ブロック暗号</td></tr>
<tr><td>名称</td><td>LED</td></tr>
<tr><td>設計者</td><td>Jian Guo[1], Thomas Peyrin[2], Axel Poschmann[2], Matt Robshaw[3]
(1: Institute for Infocomm Research/Singapore, 2: Nanyang Technological University/ Singapore, 3: Orange Labs/France)</td></tr>
<tr><td>発表年</td><td>2011（CHES 2011 [54]）</td></tr>
<tr><td>仕様参照先</td><td>CHES 2011 [54]</td></tr>
<tr><td>特徴</td><td>設計者らは、鍵スケジュールがなく、関連鍵攻撃耐性を持ち、ハードウェア実装での軽量性に特化しながらも十分なソフトウェア実装性能を持つと主張している。軽量ハッシュ関数 PHOTON と同様、serialized MDS を内部構造として採用している。
<table>
<tr><td>全体構造</td><td colspan="2">SPN 型</td></tr>
<tr><td>ブロック長 [bit]</td><td colspan="2">64</td></tr>
<tr><td>鍵長 [bit]</td><td>64 (LED-64)</td><td>128 (LED-128)</td></tr>
<tr><td>構成段数 [段]</td><td>32 (8 ステップ)</td><td>48 (12 ステップ)</td></tr>
</table></td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [44, 45, 65, 95, 98, 110, 115] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。
単一鍵設定における最良の攻撃は、2013 年に提案された Dinur ら [44] による Even-Mansour 暗号への汎用的な攻撃であり、3 ステップに簡略化した LED-64 と 8 ステップに簡略化した LED-128 に対して、それぞれ秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。関連鍵設定における最良の攻撃は、2012 年に提案された Mendel ら [95] による差分攻撃であり、4 ステップに簡略化した LED-64 に対して、秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果
<table>
<tr><th>Algorithm</th><th>Area [GE]</th><th>Cycles/block</th><th>Throughput@100kHz [kbps]</th><th>Ref.</th></tr>
<tr><td>LED-64 (Enc)</td><td>966</td><td>1,248</td><td>5.1</td><td>[54]</td></tr>
<tr><td>LED-64 (Enc)</td><td>2,695</td><td>32</td><td>200.0</td><td>[1]</td></tr>
<tr><td>LED-128 (Enc)</td><td>1,265</td><td>1,872</td><td>3.4</td><td>[54]</td></tr>
<tr><td>LED-128 (Enc)</td><td>3,036</td><td>48</td><td>133.3</td><td>[1]</td></tr>
</table>
ソフトウェア実装評価結果
<table>
<tr><th>Algorithm</th><th>Type</th><th>Cycles/byte</th><th>Platform</th><th>Ref.</th></tr>
<tr><td>LED-64</td><td>Table/VPI/Bitslice</td><td>76.0/48.1/13.1</td><td>Core i3 2367M</td><td>[13]</td></tr>
<tr><td>LED-128</td><td>Table/VPI/Bitslice</td><td>113.3/54.6/17.6</td><td>Core i3 2367M</td><td>[13]</td></tr>
</table>
その他、効率的なソフトウェア実装の結果が文献 [42] で報告されている。</td></tr>
</table>

<table>
<tr><td>技術分野</td><td>ブロック暗号</td></tr>
<tr><td>名称</td><td>Midori</td></tr>
<tr><td>設計者</td><td>Subhadeep Banik[1], Andrey Bogdanov[1], Takanori Isobe[2], Kyoji Shibutani[2], Harunaga Hiwatari[2], Toru Akishita[2], Francesco Regazzoni[3]
(1: Technical University of Denmark/Denmark, 2: Sony Corporation/Japan, 3: University of Lugano/Switzerland)</td></tr>
<tr><td>発表年</td><td>2015（ASIACRYPT 2015 [8]）</td></tr>
<tr><td>仕様参照先</td><td>ASIACRYPT 2015 [8]</td></tr>
<tr><td>特徴</td><td>設計者らは、ハードウェア実装における小型実装性能、低レイテンシ性能に加え、低エネルギー消費性能に優れたアルゴリズムであると主張している。
<table>
<tr><td>全体構造</td><td colspan="2">SPN 型</td></tr>
<tr><td>ブロック長 [bit]</td><td>64 (Midori64)</td><td>128 (Midori128)</td></tr>
<tr><td>鍵長 [bit]</td><td colspan="2">128</td></tr>
<tr><td>構成段数 [段]</td><td>16</td><td>20</td></tr>
</table></td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [5, 12, 15, 16, 33, 49, 52, 53, 56, 83, 84, 114, 115, 120, 121, 123, 132, 133] が発表されているが、弱鍵設定と関連鍵設定を除き、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。
単一鍵設定における最良の攻撃は、2015 年に提案された Liu ら [83, 84] による Midori64 への中間一致攻撃と、2016 年に提案された Tolba ら [123] による Midori128 への切り詰め差分攻撃であり、12 段に簡略化した Midori64 と 13 段に簡略化した Midori128 に対して、それぞれ秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。Midori64 には弱鍵が存在し、弱鍵設定では仕様段数であっても効率的な鍵回復攻撃 [52, 53] とメッセージ復元攻撃 [120, 121] が可能となる。関連鍵設定における最良の攻撃は、2016 年に提案された Gérault ら [49] による差分攻撃であり、Midori64 と Midori128 に対して、それぞれ仕様段数において秘密鍵の全数探索より効率的に鍵回復攻撃が実行できる。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果
<table>
<tr><th>Algorithm</th><th>Area [GE]</th><th>Cycles/block</th><th>Throughput@100kHz [kbps]</th><th>Ref.</th></tr>
<tr><td>Midori64 (Enc)</td><td>1,542</td><td>16</td><td>400.0</td><td>[8]</td></tr>
<tr><td>Midori64 (Enc/Dec)</td><td>2,450</td><td>16</td><td>400.0</td><td>[8]</td></tr>
<tr><td>Midori128 (Enc)</td><td>2,522</td><td>20</td><td>640.0</td><td>[8]</td></tr>
<tr><td>Midori128 (Enc/Dec)</td><td>3,661</td><td>20</td><td>640.0</td><td>[8]</td></tr>
</table></td></tr>
</table>

<table>
<tr><td>技術分野</td><td>ブロック暗号</td></tr>
<tr><td>名称</td><td>Piccolo</td></tr>
<tr><td>設計者</td><td>Kyoji Shibutani, Takanori Isobe, Harunaga Hiwatari, Atsushi Mitsuda, Toru Akishita, Taizo Shirai (Sony Corporation/Japan)</td></tr>
<tr><td>発表年</td><td>2011（CHES 2011 [108]）</td></tr>
<tr><td>仕様参照先</td><td>CHES 2011 [108]</td></tr>
<tr><td>特徴</td><td>設計者らは、従来の攻撃に加え、関連鍵攻撃、中間一致攻撃に対して十分な安全性を持ち、特にハードウェア実装での性能が高く、構造上、復号関数を実装したとしても大きなオーバーヘッドはなく、軽量性のみならずエネルギー効率も高いと主張している。
<table>
<tr><td>全体構造</td><td colspan="2">4-line 変形一般化 Feistel 型</td></tr>
<tr><td>ブロック長 [bit]</td><td colspan="2">64</td></tr>
<tr><td>鍵長 [bit]</td><td>80 (Piccolo-80)</td><td>128 (Piccolo-128)</td></tr>
<tr><td>構成段数 [段]</td><td>25</td><td>31</td></tr>
</table></td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [6, 55, 65, 86, 96, 108, 111, 122] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。
単一鍵設定における最良の攻撃は、2012 年に提案された Isobe ら [65] による中間一致攻撃と 2018 年に提案された Liu ら [86] による中間一致攻撃である。また、関連鍵設定における最良の攻撃は、2013 年に提案された Minier [96] による不能差分攻撃である。これらの攻撃により、14 段に簡略化した Piccolo-80 と 21 段に簡略化した Piccolo-128 に対して、それぞれ秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果
<table>
<tr><th>Algorithm</th><th>Area [GE]</th><th>Cycles/block</th><th>Throughput@100kHz [kbps]</th><th>Ref.</th></tr>
<tr><td>Piccolo-80 (Enc)</td><td>1,048</td><td>432</td><td>14.8</td><td>[108]</td></tr>
<tr><td>Piccolo-80 (Enc)</td><td>1,499</td><td>27</td><td>237.0</td><td>[108]</td></tr>
<tr><td>Piccolo-80 (Enc/Dec)</td><td>1,109</td><td>432</td><td>14.8</td><td>[108]</td></tr>
<tr><td>Piccolo-128 (Enc)</td><td>1,338</td><td>528</td><td>12.1</td><td>[108]</td></tr>
<tr><td>Piccolo-128 (Enc)</td><td>1,776</td><td>33</td><td>193.9</td><td>[108]</td></tr>
<tr><td>Piccolo-128 (Enc/Dec)</td><td>1,397</td><td>528</td><td>12.1</td><td>[108]</td></tr>
</table>
ソフトウェア実装評価結果
<table>
<tr><th>Algorithm</th><th>Type</th><th>Cycles/byte</th><th>Platform</th><th>Ref.</th></tr>
<tr><td>Piccolo-80</td><td>Bitslice</td><td>4.57</td><td>Core i7 870</td><td>[93]</td></tr>
<tr><td>Piccolo-128</td><td>Bitslice</td><td>5.52</td><td>Core i7 870</td><td>[93]</td></tr>
<tr><td>Piccolo-80</td><td>Table/VPI/Bitslice</td><td>89.3/33.3/9.2</td><td>Core i3 2367M</td><td>[13]</td></tr>
<tr><td>Piccolo-128</td><td>Table/VPI/Bitslice</td><td>103.6/41.6/10.9</td><td>Core i3 2367M</td><td>[13]</td></tr>
</table>
その他、効率的なソフトウェア実装の結果が文献 [42] で報告されている。</td></tr>
</table>

<table>
<tr><td>技術分野</td><td>ブロック暗号</td></tr>
<tr><td>名称</td><td>PRESENT</td></tr>
<tr><td>設計者</td><td>Andrey Bogdanov[1], Lars R. Knudsen[2], Gregor Leander[1], Christof Paar[1], Axel Poschmann[1], Matthew J. B. Robshaw[3], Yannick Seurin[3], C. Vikkelsoe[2]
(1: Ruhr-University Bochum/Germany, 2: Technical University Denmark/Denmark, 3: France Telecom/France)</td></tr>
<tr><td>発表年</td><td>2007（CHES 2007 [24]）</td></tr>
<tr><td>仕様参照先</td><td>CHES 2007 [24]</td></tr>
<tr><td>特徴</td><td>軽量ブロック暗号の草分け的アルゴリズムであり、特にハードウェアの小型実装において高い実装性能を持つ。
<table>
<tr><td>全体構造</td><td colspan="2">SPN 型</td></tr>
<tr><td>ブロック長 [bit]</td><td colspan="2">64</td></tr>
<tr><td>鍵長 [bit]</td><td>80 (PRESENT-80)</td><td>128 (PRESENT-128)</td></tr>
<tr><td>構成段数 [段]</td><td colspan="2">31</td></tr>
</table></td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [2, 19, 20, 21, 25, 34, 47, 71, 72, 134] が発表されているが、既知鍵設定を除き、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。
単一鍵設定における最良の攻撃は、2020 年に提案された Flórez-Gutiérrez ら [47] による多次元線形攻撃であり、28 段に簡略化した PRESENT-80 と PRESENT-128 に対して、それぞれ秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。既知鍵設定における最良の攻撃は、2015 年に提案された Blondeau ら [21] による切り詰め差分攻撃であり、PRESENT-80 と PRESENT-128 に対して、それぞれ仕様段数において効率的に識別攻撃が実行できる。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果
<table>
<tr><th>Algorithm</th><th>Area [GE]</th><th>Cycles/block</th><th>Throughput@100kHz [kbps]</th><th>Ref.</th></tr>
<tr><td>PRESENT-80 (Enc)</td><td>1,000</td><td>563</td><td>11.4</td><td>[107]</td></tr>
<tr><td>PRESENT-80 (Enc)</td><td>1,570</td><td>32</td><td>200.0</td><td>[24]</td></tr>
<tr><td>PRESENT-128 (Enc)</td><td>1,391</td><td>559</td><td>11.4</td><td>[101]</td></tr>
<tr><td>PRESENT-128 (Enc)</td><td>1,886</td><td>32</td><td>200.0</td><td>[24]</td></tr>
</table>
ソフトウェア実装評価結果 (ハイエンド CPU)
<table>
<tr><th>Algorithm</th><th>Type</th><th>Cycles/byte</th><th>Platform</th><th>Ref.</th></tr>
<tr><td>PRESENT-80/128</td><td>Bitslice</td><td>5.79</td><td>Core i7 870</td><td>[93]</td></tr>
<tr><td>PRESENT-80</td><td>Table/VPI/Bitslice</td><td>72.6/35.0/17.4</td><td>Core i3 2367M</td><td>[13]</td></tr>
<tr><td>PRESENT-128</td><td>Table/VPI/Bitslice</td><td>72.5/35.0/18.9</td><td>Core i3 2367M</td><td>[13]</td></tr>
</table>
ソフトウェア実装評価結果 (ローエンド CPU)
<table>
<tr><th>Algorithm</th><th>ROM [byte]</th><th>RAM [byte]</th><th>Cycles/byte [Enc/Dec]</th><th>Platform</th><th>Ref.</th></tr>
<tr><td>PRESENT-80</td><td>512</td><td>62</td><td>61,634/60,834</td><td>RL78</td><td>[94]</td></tr>
<tr><td>PRESENT-80</td><td>1,855</td><td>48</td><td>9,007/8,920</td><td>RL78</td><td>[94]</td></tr>
</table>
その他、効率的なソフトウェア実装の結果が文献 [42, 105] で報告されている。</td></tr>
<tr><td>標準化状況</td><td>ISO/IEC 29192-2 [66]、ISO/IEC 29167-11 [67]</td></tr>
</table>

<table>
<tr><td>技術分野</td><td>ブロック暗号</td></tr>
<tr><td>名称</td><td>PRINCE</td></tr>
<tr><td>設計者</td><td>Julia Borghoff[1], Anne Canteaut[1,2], Tim Guneysu[3], Elif Bilge Kavun[3], Miroslav Knezevic[4], Lars R. Knudsen[1], Gregor Leander[1], Ventzislav Nikov[4], Christof Paar[3], Christian Rechberger[1], Peter Rombouts[4], Soren S. Thomsen[1], Tolga Yalcin[3]
(1: Technical University of Denmark/Denmark, 2: INRIA/France, 3: Ruhr-University Bochum/Germany, 4: NXP Semiconductors/Belgium)</td></tr>
<tr><td>発表年</td><td>2012（ASIACRYPT 2012 [26]）</td></tr>
<tr><td>仕様参照先</td><td>ASIACRYPT 2012 [26]</td></tr>
<tr><td>特徴</td><td>設計者らは、ハードウェア実装における小型実装性能に加え、低レイテンシ性能にも優れたアルゴリズムであると主張している。
α-reflection と呼ばれる対称性を持つことにより、通常のブロック暗号とは異なり、128 ビット鍵を利用していても攻撃者が 2^n の平文暗号文ペアを使える場合、$(127-n)$ ビットの安全性しか主張できていない。
<table>
<tr><td>全体構造</td><td>SPN 型</td></tr>
<tr><td>ブロック長 [bit]</td><td>64</td></tr>
<tr><td>鍵長 [bit]</td><td>128</td></tr>
<tr><td>構成段数 [段]</td><td>12</td></tr>
</table></td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [5, 28, 29, 40, 41, 47, 51, 70, 80, 104] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。
単一鍵設定における最良の攻撃は、2014 年に提案された Canteaut ら [28] による多重差分攻撃であり、10 段に簡略化した PRINCE に対して、秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果
<table>
<tr><th>Algorithm</th><th>Area [GE]</th><th>Cycles/block</th><th>Throughput@100kHz [kbps]</th><th>Ref.</th></tr>
<tr><td>PRINCE (Enc/Dec)</td><td>2,953</td><td>12</td><td>533.3</td><td>[10]</td></tr>
<tr><td>PRINCE (Enc/Dec)</td><td>8,577</td><td>1</td><td>6,400.0</td><td>[10]</td></tr>
</table>
ソフトウェア実装評価結果 (ローエンド CPU)
<table>
<tr><th>Algorithm</th><th>ROM [byte]</th><th>RAM [byte]</th><th>Cycles/byte [Enc/Dec]</th><th>Platform</th><th>Ref.</th></tr>
<tr><td>PRINCE</td><td>2,382</td><td>220</td><td>225.4</td><td>ATtiny85</td><td>[100]</td></tr>
</table>
その他、効率的なソフトウェア実装の結果が文献 [42] で報告されている。</td></tr>
</table>

<table>
<tr><td>技術分野</td><td>ブロック暗号</td></tr>
<tr><td>名称</td><td>SIMON</td></tr>
<tr><td>設計者</td><td>Ray Beaulieu, Douglas Shors, Jason Smith, Stefan Treatman-Clark, Bryan Weeks, Louis Wingers（National Security Agency/USA）</td></tr>
<tr><td>発表年</td><td>2013（Cryptology ePrint Archive [11]）</td></tr>
<tr><td>仕様参照先</td><td>Cryptology ePrint Archive [11]</td></tr>
<tr><td>特徴</td><td>設計者らは、様々なブロック長、鍵長に対応したアルゴリズムであり、軽量性において、ハードウェア、ソフトウェア両方で高い実装性能を持つが、特にハードウェアでの実装性能に優れると主張している。
ブロック長 $2n$ ビット、鍵長 m ワードの SIMON を SIMON$2n/mn$ と表記する。例えば、SIMON64/128 はブロック長 64 ビット、鍵長 128 ビットの SIMON を表す。
<table>
<tr><td>全体構造</td><td colspan="10">Feistel 型</td></tr>
<tr><td>ブロック長 [bit]</td><td>32</td><td colspan="2">48</td><td colspan="2">64</td><td colspan="2">96</td><td colspan="3">128</td></tr>
<tr><td>鍵長 [bit]</td><td>64</td><td>72</td><td>96</td><td>96</td><td>128</td><td>96</td><td>144</td><td>128</td><td>192</td><td>256</td></tr>
<tr><td>構成段数 [段]</td><td>32</td><td colspan="2">36</td><td>42</td><td>44</td><td>52</td><td>54</td><td>68</td><td>69</td><td>72</td></tr>
</table></td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [9, 30, 35, 39, 57, 59, 60, 61, 76, 78, 79, 90, 91, 103, 106, 127, 128] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。
単一鍵設定における最良の攻撃は、2016 年に提案された Chen ら [30] による線形攻撃、2018 年に提案された Rohit ら [106] による correlated sequence attack、2021 年に提案された Leurent ら [79] による線形攻撃であり、27、25、31、45、56 段に簡略化したブロック長 32、48、64、96、128 ビットの SIMON に対して、それぞれ秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。既知鍵設定における最良の攻撃は、2017 年に提案された Hao ら [57] による切り詰め差分攻撃であり、29、32、37、47、63 段に簡略化したブロック長 32、48、64、96、128 ビットの SIMON に対して、それぞれ効率的に識別攻撃が実行できる。関連鍵設定における最良の攻撃は、2019 年に提案された Lee ら [78] による線形攻撃であり、23、28、34、62 段に簡略化したブロック長 32、48、64、128 ビットの SIMON に対して、それぞれ秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果
<table>
<tr><th>Algorithm</th><th>Area [GE]</th><th>Cycles/block</th><th>Throughput@100kHz [kbps]</th><th>Ref.</th></tr>
<tr><td>SIMON64/96</td><td>809</td><td>1,455</td><td>4.4</td><td>[11]</td></tr>
<tr><td>SIMON64/128</td><td>958</td><td>1,524</td><td>4.2</td><td>[11]</td></tr>
<tr><td>SIMON128/128</td><td>1,234</td><td>4,414</td><td>2.9</td><td>[11]</td></tr>
<tr><td>SIMON128/256</td><td>1,782</td><td>4,923</td><td>2.6</td><td>[11]</td></tr>
</table>
ソフトウェア実装評価結果 (ローエンド CPU)
<table>
<tr><th>Algorithm</th><th>ROM [byte]</th><th>RAM [byte]</th><th>Cycles/byte [Enc/Dec]</th><th>Platform</th><th>Ref.</th></tr>
<tr><td>SIMON64/96</td><td>274</td><td>0</td><td>239</td><td>ATtiny45</td><td>[11]</td></tr>
<tr><td>SIMON64/128</td><td>282</td><td>0</td><td>250</td><td>ATtiny45</td><td>[11]</td></tr>
<tr><td>SIMON128/128</td><td>732</td><td>0</td><td>376</td><td>ATtiny45</td><td>[11]</td></tr>
<tr><td>SIMON128/256</td><td>764</td><td>0</td><td>398</td><td>ATtiny45</td><td>[11]</td></tr>
</table>
その他、効率的なソフトウェア実装の結果が文献 [42] で報告されている。</td></tr>
<tr><td>標準化状況</td><td>ISO/IEC 29167-21 [68]</td></tr>
</table>

<table>
<tr><td>技術分野</td><td>ブロック暗号</td></tr>
<tr><td>名称</td><td>SPECK</td></tr>
<tr><td>設計者</td><td>Ray Beaulieu, Douglas Shors, Jason Smith, Stefan Treatman-Clark, Bryan Weeks, Louis Wingers（National Security Agency/USA）</td></tr>
<tr><td>発表年</td><td>2013（Cryptology ePrint Archive [11]）</td></tr>
<tr><td>仕様参照先</td><td>Cryptology ePrint Archive [11]</td></tr>
<tr><td>特徴</td><td>設計者らは、様々なブロック長、鍵長に対応したアルゴリズムであり、軽量性において、ハードウェア、ソフトウェア両方で高い実装性能を持つが、特にソフトウェアでの実装性能に優れると主張している。
SIMON 同様、ブロック長 $2n$ ビット、鍵長 m ワードの SPECK を SPECK$2n/mn$ と表記する。例えば、SPECK64/128 はブロック長 64 ビット、鍵長 128 ビットの SPECK を表す。
<table>
<tr><td>全体構造</td><td colspan="10">変形 Feistel 型</td></tr>
<tr><td>ブロック長 [bit]</td><td>32</td><td colspan="2">48</td><td colspan="2">64</td><td colspan="2">96</td><td colspan="3">128</td></tr>
<tr><td>鍵長 [bit]</td><td>64</td><td>72</td><td>96</td><td>96</td><td>128</td><td>96</td><td>144</td><td>128</td><td>192</td><td>256</td></tr>
<tr><td>構成段数 [段]</td><td colspan="2">22</td><td>23</td><td>26</td><td>27</td><td>28</td><td>29</td><td>32</td><td>33</td><td>34</td></tr>
</table></td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [9, 14, 18, 31, 32, 43, 48, 50, 62, 63, 75, 87, 88, 89, 112, 116, 126] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。
単一鍵設定における最良の攻撃は、2016 年に提案された Song ら [112] による差分攻撃であり、14、16、20、21、25 段に簡略化したブロック長 32、48、64、96、128 ビットの SPECK に対して、それぞれ秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果
<table>
<tr><th>Algorithm</th><th>Area [GE]</th><th>Cycles/block</th><th>Throughput@100kHz [kbps]</th><th>Ref.</th></tr>
<tr><td>SPECK64/96</td><td>860</td><td>1,778</td><td>3.6</td><td>[11]</td></tr>
<tr><td>SPECK64/128</td><td>996</td><td>1,882</td><td>3.4</td><td>[11]</td></tr>
<tr><td>SPECK128/128</td><td>1,280</td><td>4,267</td><td>3.0</td><td>[11]</td></tr>
<tr><td>SPECK128/256</td><td>1,840</td><td>4,571</td><td>2.8</td><td>[11]</td></tr>
</table>
ソフトウェア実装評価結果 (ローエンド CPU)
<table>
<tr><th>Algorithm</th><th>ROM [byte]</th><th>RAM [byte]</th><th>Cycles/byte [Enc/Dec]</th><th>Platform</th><th>Ref.</th></tr>
<tr><td>SPECK64/96</td><td>182</td><td>0</td><td>144</td><td>ATtiny45</td><td>[11]</td></tr>
<tr><td>SPECK64/128</td><td>186</td><td>0</td><td>150</td><td>ATtiny45</td><td>[11]</td></tr>
<tr><td>SPECK128/128</td><td>396</td><td>0</td><td>167</td><td>ATtiny45</td><td>[11]</td></tr>
<tr><td>SPECK128/256</td><td>412</td><td>0</td><td>177</td><td>ATtiny45</td><td>[11]</td></tr>
</table>
その他、効率的なソフトウェア実装の結果が文献 [42] で報告されている。</td></tr>
<tr><td>標準化状況</td><td>ISO/IEC 29167-22 [69]</td></tr>
</table>

<table>
<tr><td>技術分野</td><td>ブロック暗号</td></tr>
<tr><td>名称</td><td>TWINE</td></tr>
<tr><td>設計者</td><td>Tomoyasu Suzaki, Kazuhiko Minematsu, Sumio Morioka, Eita Kobayashi（NEC Corporation/Japan）</td></tr>
<tr><td>発表年</td><td>2011（ECRYPT Workshop on Lightweight Cryptography、SAC 2012 [118]）</td></tr>
<tr><td>仕様参照先</td><td>SAC 2012 [118]</td></tr>
<tr><td>特徴</td><td>設計者らは、ハードウェアでの軽量性のみならず、ローエンド CPU からハイエンド CPU までの幅広いソフトウェアにおいても高い実装性能を持つと主張している。FSE 2010 [117] で設計者らにより提案された改良ブロックシャッフルを採用し、安全性を高めている。
<table>
<tr><td>全体構造</td><td colspan="2">16-line 変形一般化 Feistel 型</td></tr>
<tr><td>ブロック長 [bit]</td><td colspan="2">64</td></tr>
<tr><td>鍵長 [bit]</td><td>80 (TWINE-80)</td><td>128 (TWINE-128)</td></tr>
<tr><td>構成段数 [段]</td><td colspan="2">36</td></tr>
</table></td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [3, 17, 22, 36, 73, 85, 97, 99, 124, 129, 130, 135] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。
単一鍵設定における最良の攻撃は、2016 年に提案された Lin ら [85] による多次元零相関線形攻撃であり、23 段に簡略化した TWINE-80 と 25 段に簡略化した TWINE-128 に対して、それぞれ秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果
<table>
<tr><th>Algorithm</th><th>Area [GE]</th><th>Cycles/block</th><th>Throughput@100kHz [kbps]</th><th>Ref.</th></tr>
<tr><td>TWINE-80 (Enc)</td><td>1,503</td><td>36</td><td>177.8</td><td>[118]</td></tr>
<tr><td>TWINE-80 (Enc)</td><td>1,011</td><td>393</td><td>16.3</td><td>[118]</td></tr>
<tr><td>TWINE-80 (Enc/Dec)</td><td>1,799</td><td>36</td><td>177.8</td><td>[118]</td></tr>
<tr><td>TWINE-128 (Enc)</td><td>1,866</td><td>36</td><td>177.8</td><td>[118]</td></tr>
<tr><td>TWINE-128 (Enc/Dec)</td><td>2,285</td><td>36</td><td>177.8</td><td>[118]</td></tr>
</table>
ソフトウェア実装評価結果 (ハイエンド CPU)
<table>
<tr><th>Algorithm</th><th>Type</th><th>Cycles/byte</th><th>Platform</th><th>Ref.</th></tr>
<tr><td>TWINE-80/128</td><td>Bitslice (Single/Double)</td><td>11.10/5.55</td><td>Core i7 2600S</td><td>[118]</td></tr>
</table>
ソフトウェア実装評価結果 (ローエンド CPU)
<table>
<tr><th>Algorithm</th><th>ROM [byte]</th><th>RAM [byte]</th><th>Cycles/byte [Enc/Dec]</th><th>Platform</th><th>Ref.</th></tr>
<tr><td>TWINE-80</td><td>2,294</td><td>386</td><td>163/163</td><td>ATmega163</td><td>[118]</td></tr>
<tr><td>TWINE-80</td><td>792</td><td>191</td><td>2,350/2,337</td><td>ATmega163</td><td>[118]</td></tr>
</table>
その他、効率的なソフトウェア実装の結果が文献 [42] で報告されている。</td></tr>
</table>

<table>
<tr><td>技術分野</td><td>ブロック暗号</td></tr>
<tr><td>名称</td><td>LEA</td></tr>
<tr><td>設計者</td><td>Deukjo Hong[1], Jung-Keun Lee[1], Dong-Chan Kim[1], Daesung Kwon[1], Kwon Ho Ryu[1], Dong-Geon Lee[2]
(1: Attached Institute of ETRI/Korea, 2: Pusan National University/Korea)</td></tr>
<tr><td>発表年</td><td>2013（WISA 2013 [58]）</td></tr>
<tr><td>仕様参照先</td><td>WISA 2013 [58]</td></tr>
<tr><td>特徴</td><td>設計者らは、ソフトウェア実装における高速な暗号化処理が可能であり、オーバーヘッドの軽減による低消費電力性能を持つとともに、コードサイズの小さいコンパクトな実装が可能であると主張している。また、構成段数の設定においては未知の攻撃への対策として 1.5 倍のセキュリティマージンを設けることにより、ブロック暗号に対する全ての既存攻撃に対して十分な安全性を持つと主張している。
<table>
<tr><td>全体構造</td><td colspan="3">Addition-Rotation-XOR (ARX) 型</td></tr>
<tr><td>ブロック長 [bit]</td><td colspan="3">128</td></tr>
<tr><td>鍵長 [bit]</td><td>128 (LEA-128)</td><td>192 (LEA-192)</td><td>256 (LEA-256)</td></tr>
<tr><td>構成段数 [段]</td><td>24</td><td>28</td><td>32</td></tr>
</table></td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [7, 46, 75, 112, 113] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。
単一鍵設定における最良の攻撃は、2016 年に提案された Song ら [112] による差分攻撃であり、14、14、15 段に簡略化した LEA-128、LEA-192、LEA-256 に対して、それぞれ秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。また、最良の識別攻撃は、2020 年に提案された Kim ら [75] によるブーメラン攻撃であり、16 段に簡略化した LEA-128 に対して、効率的に識別攻撃が実行できる。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果
<table>
<tr><th>Algorithm</th><th>Area [GE]</th><th>Cycles/block</th><th>Throughput@100KHz [kbps]</th><th>Ref.</th></tr>
<tr><td>LEA-128 (Enc)</td><td>3,826</td><td>168</td><td>76.19</td><td>[58]</td></tr>
<tr><td>LEA-128 (Enc)</td><td>5,426</td><td>24</td><td>533.33</td><td>[58]</td></tr>
</table>
ソフトウェア実装評価結果（ハイエンド CPU）
<table>
<tr><th>Algorithm</th><th>ROM [byte]</th><th>RAM [byte]</th><th>Cycles/byte [Enc/Dec]</th><th>Platform</th><th>Ref.</th></tr>
<tr><td>LEA-128</td><td>–</td><td>–</td><td>9.29/14.83</td><td>Intel Core 2 Quad Q6600</td><td>[58]</td></tr>
<tr><td>LEA-128</td><td>–</td><td>–</td><td>9.29/14.52</td><td>Intel Core i5-2500</td><td>[58]</td></tr>
<tr><td>LEA-128</td><td>–</td><td>–</td><td>8.85/14.50</td><td>AMD Phenom II X4 965</td><td>[58]</td></tr>
<tr><td>LEA-128</td><td>–</td><td>–</td><td>8.55/14.05</td><td>AMD Opteron 6176 SE</td><td>[58]</td></tr>
</table>
ソフトウェア実装評価結果（ローエンド CPU）
<table>
<tr><th>Algorithm</th><th>ROM [byte]</th><th>RAM [byte]</th><th>Cycles/byte [Enc/Dec]</th><th>Platform</th><th>Ref.</th></tr>
<tr><td>LEA-128</td><td>590</td><td>32</td><td>326.94/–</td><td>Arm926EJ-S</td><td>[58]</td></tr>
<tr><td>LEA-128</td><td>–</td><td>–</td><td>20.06/–</td><td>Arm926EJ-S</td><td>[58]</td></tr>
<tr><td>LEA-128</td><td>9,674</td><td>832</td><td>103.59/–</td><td>MCF5213</td><td>[58]</td></tr>
<tr><td>LEA-128</td><td>704</td><td>32</td><td>829.25/–</td><td>MCF5213</td><td>[58]</td></tr>
</table>
その他、効率的なソフトウェア実装の結果が文献 [42] で報告されている。</td></tr>
<tr><td>標準化状況</td><td>ISO/IEC 29192-2 [66]、KS X 3246 [64]</td></tr>
</table>

参考文献

[1] The LED block cipher (Dec 2013), available from https://sites.google.com/site/ledblockcipher/hardware（2023-10-04 閲覧不可）

[2] Abed, F., Forler, C., List, E., Lucks, S., Wenzel, J.: Biclique Cryptanalysis of the PRESENT and LED Lightweight Ciphers. IACR Cryptol. ePrint Arch. 2012, 591 (2012), https://eprint.iacr.org/2012/591

[3] Ahmadi, S., Ahmadian, Z., Mohajeri, J., Aref, M.R.: Biclique Cryptanalysis of Block Ciphers LBlock and TWINE-80 with Practical Data Complexity. ISC Int. J. Inf. Secur. 11(1), 57–74 (2019), https://doi.org/10.22042/isecure.2018.138036.420

[4] Akishita, T., Hiwatari, H.: Very Compact Hardware Implementations of the Blockcipher CLEFIA. In: Miri, A., Vaudenay, S. (eds.) Selected Areas in Cryptography - 18th International Workshop, SAC 2011, Toronto, ON, Canada, August 11-12, 2011, Revised Selected Papers. Lecture Notes in Computer Science, vol. 7118, pp. 278–292. Springer (2011), https://dx.doi.org/10.1007/978-3-642-28496-0_17

[5] Ankele, R., Kölbl, S.: Mind the Gap - A Closer Look at the Security of Block Ciphers against Differential Cryptanalysis. In: Cid, C., Jr., M.J.J. (eds.) Selected Areas in Cryptography - SAC 2018 - 25th International Conference, Calgary, AB, Canada, August 15-17, 2018, Revised Selected Papers. Lecture Notes in Computer Science, vol. 11349, pp. 163–190. Springer (2018), https://doi.org/10.1007/978-3-030-10970-7_8

[6] Azimi, S.A., Ahmadian, Z., Mohajeri, J., Aref, M.R.: Impossible differential cryptanalysis of Piccolo lightweight block cipher. In: 11th International ISC Conference on Information Security and Cryptology, ISCISC 2014, Tehran, Iran, September 3-4, 2014. pp. 89–94. IEEE (2014), https://doi.org/10.1109/ISCISC.2014.6994028

[7] Bagherzadeh, E., Ahmadian, Z.: MILP-based automatic differential search for LEA and HIGHT block ciphers. IET Inf. Secur. 14(5), 595–603 (2020), https://doi.org/10.1049/iet-ifs.2018.5539

[8] Banik, S., Bogdanov, A., Isobe, T., Shibutani, K., Hiwatari, H., Akishita, T., Regazzoni, F.: Midori: A Block Cipher for Low Energy. In: Iwata, T., Cheon, J.H. (eds.) 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. Lecture Notes in Computer Science, vol. 9453, pp. 411–436. Springer (2015), https://dx.doi.org/10.1007/978-3-662-48800-3_17

[9] Bao, Z., Guo, J., Liu, M., Ma, L., Tu, Y.: Conditional Differential-Neural Cryptanalysis. IACR Cryptol. ePrint Arch. 2021, 719 (2021), https://eprint.iacr.org/2021/719

[10] Batina, L., Das, A., Ege, B., Kavun, E.B., Mentens, N., Paar, C., Verbauwhede, I., Yalçin, T.: Dietary Recommendations for Lightweight Block Ciphers: Power, Energy and Area Analysis of Recently Developed Architectures. In: Hutter, M., Schmidt, J. (eds.) Radio Frequency Identification - Security and Privacy Issues 9th International Workshop, RFIDsec 2013, Graz, Austria, July 9-11, 2013, Revised Selected Papers. Lecture Notes in Computer Science, vol. 8262, pp. 103–112. Springer (2013), https://dx.doi.org/10.1007/978-3-642-41332-2_7

[11] Beaulieu, R., Shors, D., Smith, J., Treatman-Clark, S., Weeks, B., Wingers, L.: The SIMON and SPECK Families of Lightweight Block Ciphers. IACR Cryptology ePrint Archive 2013, 404 (2013), https://eprint.iacr.org/2013/404

[12] Beierle, C., Canteaut, A., Leander, G.: Nonlinear Approximations in Cryptanalysis Revisited. IACR Trans. Symmetric Cryptol. 2018(4), 80–101 (2018), https://doi.org/10.13154/tosc.v2018.i4.80-101

[13] Benadjila, R., Guo, J., Lomné, V., Peyrin, T.: Implementing Lightweight Block Ciphers on x86 Architectures. In: Lange et al. [77], pp. 324–351, https://dx.doi.org/10.1007/978-3-662-43414-7_17

[14] Benamira, A., Gérault, D., Peyrin, T., Tan, Q.Q.: A Deeper Look at Machine Learning-Based Cryptanalysis. In: Canteaut, A., Standaert, F. (eds.) Advances in Cryptology - EUROCRYPT 2021 - 40th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, October 17-21, 2021, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12696, pp. 805–835. Springer (2021), https://doi.org/10.1007/978-3-030-77870-5_28

[15] Beyne, T.: Block Cipher Invariants as Eigenvectors of Correlation Matrices. In: Peyrin, T., Galbraith, S.D. (eds.) Advances in Cryptology - ASIACRYPT 2018 - 24th International Conference on the Theory and Application of Cryptology and Information Security, Brisbane, QLD, Australia, December 2-6, 2018, Proceedings, Part I. Lecture Notes in Computer Science, vol. 11272, pp. 3–31. Springer (2018), https://doi.org/10.1007/978-3-030-03326-2_1

[16] Beyne, T.: Block Cipher Invariants as Eigenvectors of Correlation Matrices. J. Cryptol. 33(3), 1156–1183 (2020), https://doi.org/10.1007/s00145-020-09344-1

[17] Biryukov, A., Derbez, P., Perrin, L.: Differential Analysis and Meet-in-the-Middle Attack Against Round-Reduced TWINE. In: Leander, G. (ed.) Fast Software Encryption - 22nd International Workshop, FSE 2015, Istanbul, Turkey, March 8-11, 2015, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9054, pp. 3–27. Springer (2015), https://doi.org/10.1007/978-3-662-48116-5_1

[18] Biryukov, A., Velichkov, V., Corre, Y.L.: Automatic Search for the Best Trails in ARX: Application to Block Cipher Speck. In: Peyrin, T. (ed.) Fast Software Encryption - 23rd International Conference, FSE 2016, Bochum, Germany, March 20-23, 2016, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9783, pp. 289–310. Springer (2016), https://doi.org/10.1007/978-3-662-52993-5_15

[19] Blondeau, C., Nyberg, K.: Links between Truncated Differential and Multidimensional Linear Properties of Block Ciphers and Underlying Attack Complexities. In: Nguyen, P.Q., Oswald, E. (eds.) Advances in Cryptology - EUROCRYPT 2014 - 33rd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Copenhagen, Denmark, May 11-15, 2014. Proceedings. Lecture Notes in Computer Science, vol. 8441, pp. 165–182. Springer (2014), https://dx.doi.org/10.1007/978-3-642-55220-5_10

[20] Blondeau, C., Nyberg, K.: Improved Parameter Estimates for Correlation and Capacity Deviates in Linear Cryptanalysis. IACR Trans. Symmetric Cryptol. 2016(2), 162–191 (2016), https://doi.org/10.13154/tosc.v2016.i2.162-191

[21] Blondeau, C., Peyrin, T., Wang, L.: Known-Key Distinguisher on Full PRESENT. In: Gennaro, R., Robshaw, M. (eds.) Advances in Cryptology - CRYPTO 2015 - 35th Annual Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2015, Proceedings, Part I. Lecture Notes in Computer Science, vol. 9215, pp. 455–474. Springer (2015), https://doi.org/10.1007/978-3-662-47989-6_22

[22] Bogdanov, A., Boura, C., Rijmen, V., Wang, M., Wen, L., Zhao, J.: Key Difference Invariant Bias in Block Ciphers. In: Sako, K., Sarkar, P. (eds.) Advances in Cryptology - ASIACRYPT 2013 - 19th International Conference on the Theory and Application of Cryptology and Information Security, Bengaluru, India, December 1-5, 2013, Proceedings, Part I. Lecture Notes in Computer Science, vol. 8269, pp. 357–376. Springer (2013), https://doi.org/10.1007/978-3-642-42033-7_19

[23] Bogdanov, A., Geng, H., Wang, M., Wen, L., Collard, B.: Zero-Correlation Linear Cryptanalysis with FFT and Improved Attacks on ISO Standards Camellia and CLEFIA. In: Lange et al. [77], pp. 306–323, https://dx.doi.org/10.1007/978-3-662-43414-7_16

[24] Bogdanov, A., Knudsen, L.R., Leander, G., Paar, C., Poschmann, A., Robshaw, M.J.B., Seurin, Y., Vikkelsoe, C.: PRESENT: An Ultra-Lightweight Block Cipher. In: Paillier, P., Verbauwhede, I. (eds.) Cryptographic Hardware and Embedded Systems - CHES 2007, 9th International Workshop, Vienna, Austria, September 10-

13, 2007, Proceedings. Lecture Notes in Computer Science, vol. 4727, pp. 450–466. Springer (2007), https://dx.doi.org/10.1007/978-3-540-74735-2_31

[25] Bogdanov, A., Tischhauser, E., Vejre, P.S.: Multivariate Profiling of Hulls for Linear Cryptanalysis. IACR Trans. Symmetric Cryptol. 2018(1), 101–125 (2018), https://doi.org/10.13154/tosc.v2018.i1.101-125

[26] Borghoff, J., Canteaut, A., Güneysu, T., Kavun, E.B., Knezevic, M., Knudsen, L.R., Leander, G., Nikov, V., Paar, C., Rechberger, C., Rombouts, P., Thomsen, S.S., Yalçin, T.: PRINCE - A Low-Latency Block Cipher for Pervasive Computing Applications - Extended Abstract. In: Wang, X., Sako, K. (eds.) 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. Lecture Notes in Computer Science, vol. 7658, pp. 208–225. Springer (2012), https://dx.doi.org/10.1007/978-3-642-34961-4_14

[27] Boura, C., Naya-Plasencia, M., Suder, V.: Scrutinizing and Improving Impossible Differential Attacks: Applications to CLEFIA, Camellia, LBlock and Simon. In: Sarkar, P., Iwata, T. (eds.) Advances in Cryptology - ASIACRYPT 2014 - 20th International Conference on the Theory and Application of Cryptology and Information Security, Kaoshiung, Taiwan, R.O.C., December 7-11, 2014. Proceedings, Part I. Lecture Notes in Computer Science, vol. 8873, pp. 179–199. Springer (2014), https://doi.org/10.1007/978-3-662-45611-8_10

[28] Canteaut, A., Fuhr, T., Gilbert, H., Naya-Plasencia, M., Reinhard, J.: Multiple Differential Cryptanalysis of Round-Reduced PRINCE. In: Cid, C., Rechberger, C. (eds.) Fast Software Encryption - 21st International Workshop, FSE 2014, London, UK, March 3-5, 2014. Revised Selected Papers. Lecture Notes in Computer Science, vol. 8540, pp. 591–610. Springer (2014), https://dx.doi.org/10.1007/978-3-662-46706-0_30

[29] Canteaut, A., Naya-Plasencia, M., Vayssière, B.: Sieve-in-the-Middle: Improved MITM Attacks. In: Canetti, R., Garay, J.A. (eds.) Advances in Cryptology - CRYPTO 2013 - 33rd Annual Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2013. Proceedings, Part I. Lecture Notes in Computer Science, vol. 8042, pp. 222–240. Springer (2013), https://doi.org/10.1007/978-3-642-40041-4_13

[30] Chen, H., Wang, X.: Improved Linear Hull Attack on Round-Reduced Simon with Dynamic Key-Guessing Techniques. In: Peyrin, T. (ed.) Fast Software Encryption - 23rd International Conference, FSE 2016, Bochum, Germany, March 20-23, 2016, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9783, pp. 428–449. Springer (2016), https://dx.doi.org/10.1007/978-3-662-52993-5_22

[31] Chen, Y., Yu, H.: Bridging Machine Learning and Cryptanalysis via EDLCT. IACR Cryptol. ePrint Arch. 2021, 705 (2021), https://eprint.iacr.org/2021/705

[32] Chen, Y., Yu, H.: Improved Neural Aided Statistical Attack for Cryptanalysis. IACR Cryptol. ePrint Arch. 2021, 311 (2021), https://eprint.iacr.org/2021/311

[33] Chen, Z., Chen, H., Wang, X.: Cryptanalysis of Midori128 Using Impossible Differential Techniques. In: Bao, F., Chen, L., Deng, R.H., Wang, G. (eds.) Information Security Practice and Experience - 12th International Conference, ISPEC 2016, Zhangjiajie, China, November 16-18, 2016, Proceedings. Lecture Notes in Computer Science, vol. 10060, pp. 1–12 (2016), https://doi.org/10.1007/978-3-319-49151-6_1

[34] Cho, J.Y.: Linear Cryptanalysis of Reduced-Round PRESENT. In: Pieprzyk, J. (ed.) Topics in Cryptology - CT-RSA 2010, The Cryptographers' Track at the RSA Conference 2010, San Francisco, CA, USA, March 1-5, 2010. Proceedings. Lecture Notes in Computer Science, vol. 5985, pp. 302–317. Springer (2010), https://dx.doi.org/10.1007/978-3-642-11925-5_21

[35] Chu, Z., Chen, H., Wang, X., Dong, X., Li, L.: Improved Integral Attacks on SIMON32 and SIMON48 with Dynamic Key-Guessing Techniques. Secur. Commun. Networks 2018, 5160237:1–5160237:11 (2018), https://doi.org/10.1155/2018/5160237

[36] Çoban, M., Karakoç, F., Boztas, Ö.: Biclique Cryptanalysis of TWINE. In: Pieprzyk, J., Sadeghi, A., Manulis, M. (eds.) Cryptology and Network Security, 11th International Conference, CANS 2012, Darmstadt, Germany, December 12-14, 2012. Proceedings. vol. 7712, pp. 43–55. Springer (2012), https://doi.org/10.1007/

978-3-642-35404-5_5

[37] Corporation, S.: CLEFIA: The 128-bit Blockcipher, https://www.sony.net/Products/cryptography/clefia/（2023-10-04 閲覧）

[38] CRYPTREC 軽量暗号ワーキンググループ: CRYPTREC 暗号技術ガイドライン（軽量暗号）（文書番号：CRYPTREC GL-2003-2016JP）(2017), https://www.cryptrec.go.jp/report/cryptrec-gl-2003-2016jp.pdf

[39] Derbez, P., Fouque, P.: Automatic Search of Meet-in-the-Middle and Impossible Differential Attacks. In: Robshaw, M., Katz, J. (eds.) Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part II. Lecture Notes in Computer Science, vol. 9815, pp. 157–184. Springer (2016), https://doi.org/10.1007/978-3-662-53008-5_6

[40] Derbez, P., Perrin, L.: Meet-in-the-Middle Attacks and Structural Analysis of Round-Reduced PRINCE. In: Leander, G. (ed.) Fast Software Encryption - 22nd International Workshop, FSE 2015, Istanbul, Turkey, March 8-11, 2015, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9054, pp. 190–216. Springer (2015), https://doi.org/10.1007/978-3-662-48116-5_10

[41] Derbez, P., Perrin, L.: Meet-in-the-Middle Attacks and Structural Analysis of Round-Reduced PRINCE. J. Cryptol. 33(3), 1184–1215 (2020), https://doi.org/10.1007/s00145-020-09345-0

[42] Dinu, D., Corre, Y.L., Khovratovich, D., Perrin, L., Großschädl, J., Biryukov, A.: Triathlon of lightweight block ciphers for the Internet of things. J. Cryptogr. Eng. 9(3), 283–302 (2019), https://doi.org/10.1007/s13389-018-0193-x

[43] Dinur, I.: Improved Differential Cryptanalysis of Round-Reduced Speck. In: Joux, A., Youssef, A.M. (eds.) Selected Areas in Cryptography - SAC 2014 - 21st International Conference, Montreal, QC, Canada, August 14-15, 2014, Revised Selected Papers. Lecture Notes in Computer Science, vol. 8781, pp. 147–164. Springer (2014), https://dx.doi.org/10.1007/978-3-319-13051-4_9

[44] Dinur, I., Dunkelman, O., Keller, N., Shamir, A.: Key Recovery Attacks on 3-round Even-Mansour, 8-step LED-128, and Full AES^2. In: Sako, K., Sarkar, P. (eds.) Advances in Cryptology - ASIACRYPT 2013 - 19th International Conference on the Theory and Application of Cryptology and Information Security, Bengaluru, India, December 1-5, 2013, Proceedings, Part I. Lecture Notes in Computer Science, vol. 8269, pp. 337–356. Springer (2013), https://dx.doi.org/10.1007/978-3-642-42033-7_18

[45] Dinur, I., Dunkelman, O., Keller, N., Shamir, A.: Improved Linear Sieving Techniques with Applications to Step-Reduced LED-64. In: Cid, C., Rechberger, C. (eds.) Fast Software Encryption - 21st International Workshop, FSE 2014, London, UK, March 3-5, 2014. Revised Selected Papers. Lecture Notes in Computer Science, vol. 8540, pp. 390–410. Springer (2014), https://doi.org/10.1007/978-3-662-46706-0_20

[46] Dwivedi, A.D., Srivastava, G.: Differential Cryptanalysis of Round-Reduced LEA. IEEE Access 6, 79105–79113 (2018), https://doi.org/10.1109/ACCESS.2018.2881130

[47] Flórez-Gutiérrez, A., Naya-Plasencia, M.: Improving Key-Recovery in Linear Attacks: Application to 28-Round PRESENT. In: Canteaut, A., Ishai, Y. (eds.) Advances in Cryptology - EUROCRYPT 2020 - 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12105, pp. 221–249. Springer (2020), https://doi.org/10.1007/978-3-030-45721-1_9

[48] Fu, K., Wang, M., Guo, Y., Sun, S., Hu, L.: MILP-Based Automatic Search Algorithms for Differential and Linear Trails for Speck. In: Peyrin, T. (ed.) Fast Software Encryption - 23rd International Conference, FSE 2016, Bochum, Germany, March 20-23, 2016, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9783, pp. 268–288. Springer (2016), https://doi.org/10.1007/978-3-662-52993-5_14

[49] Gérault, D., Lafourcade, P.: Related-Key Cryptanalysis of Midori. In: Dunkelman, O., Sanadhya, S.K. (eds.) Progress in Cryptology - INDOCRYPT 2016 - 17th International Conference on Cryptology in India, Kolkata, India, December 11-14, 2016, Proceedings. Lecture Notes in Computer Science, vol. 10095, pp. 287–304 (2016),

https://doi.org/10.1007/978-3-319-49890-4_16

[50] Gohr, A.: Improving Attacks on Round-Reduced Speck32/64 Using Deep Learning. In: Boldyreva, A., Micciancio, D. (eds.) Advances in Cryptology - CRYPTO 2019 - 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2019, Proceedings, Part II. Lecture Notes in Computer Science, vol. 11693, pp. 150–179. Springer (2019), https://doi.org/10.1007/978-3-030-26951-7_6

[51] Grassi, L., Rechberger, C.: Practical Low Data-Complexity Subspace-Trail Cryptanalysis of Round-Reduced PRINCE. In: Dunkelman, O., Sanadhya, S.K. (eds.) Progress in Cryptology - INDOCRYPT 2016 - 17th International Conference on Cryptology in India, Kolkata, India, December 11-14, 2016, Proceedings. Lecture Notes in Computer Science, vol. 10095, pp. 322–342 (2016), https://doi.org/10.1007/978-3-319-49890-4_18

[52] Guo, J., Jean, J., Nikolic, I., Qiao, K., Sasaki, Y., Sim, S.M.: Invariant Subspace Attack Against Midori64 and The Resistance Criteria for S-box Designs. IACR Trans. Symmetric Cryptol. 2016(1), 33–56 (2016), https://doi.org/10.13154/tosc.v2016.i1.33-56

[53] Guo, J., Jean, J., Nikolic, I., Sasaki, Y., Sim, S.M.: Invariant Subspace Attack Against Midori64 and The Resistance Criteria for S-box Designs. IACR Cryptol. ePrint Arch. 2016, 973 (2016), https://eprint.iacr.org/2016/973

[54] Guo, J., Peyrin, T., Poschmann, A., Robshaw, M.J.B.: The LED Block Cipher. In: Preneel and Takagi [102], pp. 326–341, https://dx.doi.org/10.1007/978-3-642-23951-9_22

[55] Han, G., Zhang, W.: Improved Biclique Cryptanalysis of the Lightweight Block Cipher Piccolo. Secur. Commun. Networks 2017, 7589306:1–7589306:12 (2017), https://doi.org/10.1155/2017/7589306

[56] Han, G., Zhang, W., Xing, Z., Zhao, H., Lian, J.: Unbalanced biclique cryptanalysis of a full round Midori. IET Commun. 13(5), 505–511 (2019), https://doi.org/10.1049/iet-com.2018.5343

[57] Hao, Y., Meier, W.: Truncated differential based known-key attacks on round-reduced SIMON. Des. Codes Cryptogr. 83(2), 467–492 (2017), https://doi.org/10.1007/s10623-016-0242-3

[58] Hong, D., Lee, J., Kim, D., Kwon, D., Ryu, K.H., Lee, D.: LEA: A 128-Bit Block Cipher for Fast Encryption on Common Processors. In: Kim, Y., Lee, H., Perrig, A. (eds.) Information Security Applications - 14th International Workshop, WISA 2013, Jeju Island, Korea, August 19-21, 2013, Revised Selected Papers. Lecture Notes in Computer Science, vol. 8267, pp. 3–27. Springer (2013), https://doi.org/10.1007/978-3-319-05149-9_1

[59] Hou, Z., Ren, J., Chen, S.: Cryptanalysis of Round-Reduced SIMON32 Based on Deep Learning. IACR Cryptol. ePrint Arch. 2021, 362 (2021), https://eprint.iacr.org/2021/362

[60] Hou, Z., Ren, J., Chen, S.: Improve Neural Distinguisher for Cryptanalysis. IACR Cryptol. ePrint Arch. 2021, 1017 (2021), https://eprint.iacr.org/2021/1017

[61] Hou, Z., Ren, J., Chen, S.: SAT-based Method to Improve Neural Distinguisher and Applications to SIMON. IACR Cryptol. ePrint Arch. 2021, 452 (2021), https://eprint.iacr.org/2021/452

[62] Huang, M., Wang, L.: Automatic Tool for Searching for Differential Characteristics in ARX Ciphers and Applications. In: Hao, F., Ruj, S., Gupta, S.S. (eds.) Progress in Cryptology - INDOCRYPT 2019 - 20th International Conference on Cryptology in India, Hyderabad, India, December 15-18, 2019, Proceedings. Lecture Notes in Computer Science, vol. 11898, pp. 115–138. Springer (2019), https://doi.org/10.1007/978-3-030-35423-7_6

[63] Huang, M., Wang, L.: Automatic Search for the Linear (Hull) Characteristics of ARX Ciphers: Applied to SPECK, SPARX, Chaskey, and CHAM-64. Secur. Commun. Networks 2020, 4898612:1–4898612:14 (2020), https://doi.org/10.1155/2020/4898612

[64] Internet, K., (KISA), S.A.: 128-bit block cipher LEA (2016), kS X 3246, https://www.rra.go.kr/ko/reference/kcsList_view.do?nb_seq=1923&cpage=4&nb_type=6&searchCon=&searchTxt=&sortOrder= (in Korean)

[65] Isobe, T., Shibutani, K.: Security Analysis of the Lightweight Block Ciphers XTEA, LED and Piccolo. In: Susilo, W., Mu, Y., Seberry, J. (eds.) Information Security and Privacy - 17th Australasian Conference, ACISP 2012,

Wollongong, NSW, Australia, July 9-11, 2012. Proceedings. Lecture Notes in Computer Science, vol. 7372, pp. 71–86. Springer (2012), https://doi.org/10.1007/978-3-642-31448-3_6

[66] ISO/IEC: Information security – Lightweight cryptography – Part 2: Block ciphers (ISO/IEC 29192-2:2019), https://www.iso.org/standard/78477.html

[67] ISO/IEC: Information technology — Automatic identification and data capture techniques – Part 11: Crypto suite PRESENT-80 security services for air interface communications (ISO/IEC 29167-11: 2023), https://www.iso.org/standard/81489.html

[68] ISO/IEC: Information technology — Automatic identification and data capture techniques – Part 21: Crypto suite SIMON security services for air interface communications (ISO/IEC 29167-21: 2018), https://www.iso.org/standard/70388.html

[69] ISO/IEC: Information technology — Automatic identification and data capture techniques – Part 22: Crypto suite SPECK security services for air interface communications (ISO/IEC 29167-22: 2018), https://www.iso.org/standard/70389.html

[70] Jean, J., Nikolic, I., Peyrin, T., Wang, L., Wu, S.: Security Analysis of PRINCE. In: Moriai, S. (ed.) Fast Software Encryption - 20th International Workshop, FSE 2013, Singapore, March 11-13, 2013. Revised Selected Papers. Lecture Notes in Computer Science, vol. 8424, pp. 92–111. Springer (2013), https://doi.org/10.1007/978-3-662-43933-3_6

[71] Jeong, K., Kang, H., Lee, C., Sung, J., Hong, S.: Biclique Cryptanalysis of Lightweight Block Ciphers PRESENT, Piccolo and LED. IACR Cryptol. ePrint Arch. 2012, 621 (2012), https://eprint.iacr.org/2012/621

[72] Jithendra, K.B., Kassim, S.T.: New Biclique Cryptanalysis on Full-Round PRESENT-80 Block Cipher. SN Comput. Sci. 1(2), 94 (2020), https://doi.org/10.1007/s42979-020-0103-z

[73] Karakoç, F., Demirci, H., Harmanci, A.E.: Biclique cryptanalysis of LBlock and TWINE. Inf. Process. Lett. 113(12), 423–429 (2013), https://doi.org/10.1016/j.ipl.2013.03.011

[74] Katagi, M.: The 128-Bit Blockcipher CLEFIA. RFC 6114 (Mar 2011), https://www.rfc-editor.org/info/rfc6114

[75] Kim, D., Kwon, D., Song, J.: Efficient Computation of Boomerang Connection Probability for ARX-Based Block Ciphers with Application to SPECK and LEA. IEICE Trans. Fundam. Electron. Commun. Comput. Sci. 103-A(4), 677–685 (2020), https://doi.org/10.1587/transfun.2019EAP1083

[76] Koo, B., Jung, Y., Kim, W.: Rotational-XOR Rectangle Cryptanalysis on Round-Reduced Simon. Secur. Commun. Networks 2020, 5968584:1–5968584:12 (2020), https://doi.org/10.1155/2020/5968584

[77] Lange, T., Lauter, K.E., Lisonek, P. (eds.): Selected Areas in Cryptography - SAC 2013 - 20th International Conference, Burnaby, BC, Canada, August 14-16, 2013, Revised Selected Papers, Lecture Notes in Computer Science, vol. 8282. Springer (2014), https://dx.doi.org/10.1007/978-3-662-43414-7

[78] Lee, J., Koo, B., Kim, W.: A General Framework for the Related-Key Linear Attack Against Block Ciphers with Linear Key Schedules. In: Paterson, K.G., Stebila, D. (eds.) Selected Areas in Cryptography - SAC 2019 - 26th International Conference, Waterloo, ON, Canada, August 12-16, 2019, Revised Selected Papers. Lecture Notes in Computer Science, vol. 11959, pp. 194–224. Springer (2019), https://doi.org/10.1007/978-3-030-38471-5_9

[79] Leurent, G., Pernot, C., Schrottenloher, A.: Clustering Effect in Simon and Simeck. In: Tibouchi, M., Wang, H. (eds.) Advances in Cryptology - ASIACRYPT 2021 - 27th International Conference on the Theory and Application of Cryptology and Information Security, Singapore, December 6-10, 2021, Proceedings, Part I. Lecture Notes in Computer Science, vol. 13090, pp. 272–302. Springer (2021), https://doi.org/10.1007/978-3-030-92062-3_10

[80] Li, L., Jia, K., Wang, X.: Improved Meet-in-the-Middle Attacks on AES-192 and PRINCE. IACR Cryptol. ePrint Arch. 2013, 573 (2013), https://eprint.iacr.org/2013/573

[81] Li, L., Jia, K., Wang, X., Dong, X.: Meet-in-the-Middle Technique for Truncated Differential and Its Applications

to CLEFIA and Camellia. In: Leander, G. (ed.) Fast Software Encryption - 22nd International Workshop, FSE 2015, Istanbul, Turkey, March 8-11, 2015, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9054, pp. 48–70. Springer (2015), https://doi.org/10.1007/978-3-662-48116-5_3

[82] Li, Y., Wu, W., Zhang, L.: Improved Integral Attacks on Reduced-Round CLEFIA Block Cipher. In: Jung, S., Yung, M. (eds.) Information Security Applications - 12th International Workshop, WISA 2011, Jeju Island, Korea, August 22-24, 2011. Revised Selected Papers. Lecture Notes in Computer Science, vol. 7115, pp. 28–39. Springer (2011), https://doi.org/10.1007/978-3-642-27890-7_3

[83] Lin, L., Wu, W.: Meet-in-the-Middle Attacks on Reduced-Round Midori-64. IACR Cryptology ePrint Archive 2015, 1165 (2015), https://eprint.iacr.org/2015/1165

[84] Lin, L., Wu, W.: Meet-in-the-Middle Attacks on Reduced-Round Midori64. IACR Trans. Symmetric Cryptol. 2017(1), 215–239 (2017), https://doi.org/10.13154/tosc.v2017.i1.215-239

[85] Lin, L., Wu, W., Zheng, Y.: Automatic Search for Key-Bridging Technique: Applications to LBlock and TWINE. In: Peyrin, T. (ed.) Fast Software Encryption - 23rd International Conference, FSE 2016, Bochum, Germany, March 20-23, 2016, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9783, pp. 247–267. Springer (2016), https://doi.org/10.1007/978-3-662-52993-5_13

[86] Liu, Y., Cheng, L., Liu, Z., Li, W., Wang, Q., Gu, D.: Improved meet-in-the-middle attacks on reduced-round Piccolo. Sci. China Inf. Sci. 61(3), 032108:1–032108:13 (2018), https://doi.org/10.1007/s11432-016-9157-y

[87] Liu, Y., Wang, Q., Rijmen, V.: Automatic Search of Linear Trails in ARX with Applications to SPECK and Chaskey. In: Manulis, M., Sadeghi, A., Schneider, S.A. (eds.) Applied Cryptography and Network Security - 14th International Conference, ACNS 2016, Guildford, UK, June 19-22, 2016. Proceedings. Lecture Notes in Computer Science, vol. 9696, pp. 485–499. Springer (2016), https://doi.org/10.1007/978-3-319-39555-5_26

[88] Liu, Y., Witte, G.D., Ranea, A., Ashur, T.: Rotational-XOR Cryptanalysis of Reduced-round SPECK. IACR Trans. Symmetric Cryptol. 2017(3), 24–36 (2017), https://doi.org/10.13154/tosc.v2017.i3.24-36

[89] Liu, Z., Li, Y., Jiao, L., Wang, M.: A New Method for Searching Optimal Differential and Linear Trails in ARX Ciphers. IEEE Trans. Inf. Theory 67(2), 1054–1068 (2021), https://doi.org/10.1109/TIT.2020.3040543

[90] Liu, Z., Li, Y., Wang, M.: Optimal Differential Trails in SIMON-like Ciphers. IACR Trans. Symmetric Cryptol. 2017(1), 358–379 (2017), https://doi.org/10.13154/tosc.v2017.i1.358-379

[91] Lu, J., Liu, Y., Ashur, T., Sun, B., Li, C.: Rotational-XOR Cryptanalysis of Simon-Like Block Ciphers. In: Liu, J.K., Cui, H. (eds.) Information Security and Privacy - 25th Australasian Conference, ACISP 2020, Perth, WA, Australia, November 30 - December 2, 2020, Proceedings. Lecture Notes in Computer Science, vol. 12248, pp. 105–124. Springer (2020), https://doi.org/10.1007/978-3-030-55304-3_6

[92] Mala, H., Dakhilalian, M., Shakiba, M.: Impossible differential attacks on 13-round CLEFIA-128. J. Comput. Sci. Technol. 26(4), 744–750 (2011), https://dx.doi.org/10.1007/s11390-011-1173-0

[93] Matsuda, S., Moriai, S.: Lightweight Cryptography for the Cloud: Exploit the Power of Bitslice Implementation. In: Prouff, E., Schaumont, P. (eds.) Cryptographic Hardware and Embedded Systems - CHES 2012 - 14th International Workshop, Leuven, Belgium, September 9-12, 2012. Proceedings. Lecture Notes in Computer Science, vol. 7428, pp. 408–425. Springer (2012), https://dx.doi.org/10.1007/978-3-642-33027-8_24

[94] Matsui, M., Murakami, Y.: Minimalism of Software Implementation - Extensive Performance Analysis of Symmetric Primitives on the RL78 Microcontroller. In: Moriai, S. (ed.) Fast Software Encryption - 20th International Workshop, FSE 2013, Singapore, March 11-13, 2013. Revised Selected Papers. Lecture Notes in Computer Science, vol. 8424, pp. 393–409. Springer (2013), https://dx.doi.org/10.1007/978-3-662-43933-3_20

[95] Mendel, F., Rijmen, V., Toz, D., Varici, K.: Differential Analysis of the LED Block Cipher. In: Wang, X., Sako, K. (eds.) 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. Lecture Notes in Computer Science, vol. 7658, pp. 190–207. Springer (2012), https://doi.org/10.1007/978-3-642-34961-4_

13

[96] Minier, M.: On the Security of Piccolo Lightweight Block Cipher against Related-Key Impossible Differentials. In: Paul, G., Vaudenay, S. (eds.) Progress in Cryptology - INDOCRYPT 2013 - 14th International Conference on Cryptology in India, Mumbai, India, December 7-10, 2013. Proceedings. Lecture Notes in Computer Science, vol. 8250, pp. 308–318. Springer (2013), https://dx.doi.org/10.1007/978-3-319-03515-4_21

[97] Najarkolaei, S.R.H., Ahangarkolaei, M.Z., Ahmadi, S., Aref, M.R.: Biclique cryptanalysis of TWINE-128. In: 13th International Iranian Society of Cryptology Conference on Information Security and Cryptology, ISCISC 2016, Tehran, Iran, September 7-8, 2016. pp. 46–51. IEEE (2016), https://doi.org/10.1109/ISCISC.2016.7736450

[98] Nikolic, I., Wang, L., Wu, S.: Cryptanalysis of Round-Reduced LED. In: Moriai, S. (ed.) Fast Software Encryption - 20th International Workshop, FSE 2013, Singapore, March 11-13, 2013. Revised Selected Papers. Lecture Notes in Computer Science, vol. 8424, pp. 112–129. Springer (2013), https://doi.org/10.1007/978-3-662-43933-3_7

[99] Niu, C., Li, M., Sun, S., Wang, M.: Zero-Correlation Linear Cryptanalysis with Equal Treatment for Plaintexts and Tweakeys. In: Paterson, K.G. (ed.) Topics in Cryptology - CT-RSA 2021 - Cryptographers' Track at the RSA Conference 2021, Virtual Event, May 17-20, 2021, Proceedings. Lecture Notes in Computer Science, vol. 12704, pp. 126–147. Springer (2021), https://doi.org/10.1007/978-3-030-75539-3_6

[100] Papapagiannopoulos, K.: High Throughput in Slices: The Case of PRESENT, PRINCE and KATAN64 Ciphers. In: Saxena, N., Sadeghi, A. (eds.) Radio Frequency Identification: Security and Privacy Issues - 10th International Workshop, RFIDSec 2014, Oxford, UK, July 21-23, 2014, Revised Selected Papers. Lecture Notes in Computer Science, vol. 8651, pp. 137–155. Springer (2014), https://dx.doi.org/10.1007/978-3-319-13066-8_9

[101] Poschmann, A.: Lightweight Cryptography - Cryptographic Engineering for a Pervasive World. IACR Cryptology ePrint Archive 2009, 516 (2009), https://eprint.iacr.org/2009/516

[102] Preneel, B., Takagi, T. (eds.): Cryptographic Hardware and Embedded Systems - CHES 2011 - 13th International Workshop, Nara, Japan, September 28 - October 1, 2011. Proceedings, Lecture Notes in Computer Science, vol. 6917. Springer (2011), https://dx.doi.org/10.1007/978-3-642-23951-9

[103] Qiao, K., Hu, L., Sun, S.: Differential Analysis on Simeck and SIMON with Dynamic Key-Guessing Techniques. In: Camp, O., Furnell, S., Mori, P. (eds.) Information Systems Security and Privacy - Second International Conference, ICISSP 2016, Rome, Italy, February 19-21, 2016, Revised Selected Papers. Communications in Computer and Information Science, vol. 691, pp. 64–85. Springer (2016), https://doi.org/10.1007/978-3-319-54433-5_5

[104] Rasoolzadeh, S., Raddum, H.: Cryptanalysis of PRINCE with Minimal Data. In: Pointcheval, D., Nitaj, A., Rachidi, T. (eds.) Progress in Cryptology - AFRICACRYPT 2016 - 8th International Conference on Cryptology in Africa, Fes, Morocco, April 13-15, 2016, Proceedings. Lecture Notes in Computer Science, vol. 9646, pp. 109–126. Springer (2016), https://doi.org/10.1007/978-3-319-31517-1_6

[105] Reis, T.B.S., Aranha, D.F., López-Hernández, J.C.: PRESENT Runs Fast - Efficient and Secure Implementation in Software. In: Fischer, W., Homma, N. (eds.) Cryptographic Hardware and Embedded Systems - CHES 2017 - 19th International Conference, Taipei, Taiwan, September 25-28, 2017, Proceedings. Lecture Notes in Computer Science, vol. 10529, pp. 644–664. Springer (2017), https://doi.org/10.1007/978-3-319-66787-4_31

[106] Rohit, R., Gong, G.: Correlated Sequence Attack on Reduced-Round Simon-32/64 and Simeck-32/64. IACR Cryptol. ePrint Arch. 2018, 699 (2018), https://eprint.iacr.org/2018/699

[107] Rolfes, C., Poschmann, A., Leander, G., Paar, C.: Ultra-Lightweight Implementations for Smart Devices - Security for 1000 Gate Equivalents. In: Grimaud, G., Standaert, F. (eds.) Smart Card Research and Advanced Applications, 8th IFIP WG 8.8/11.2 International Conference, CARDIS 2008, London, UK, September 8-11, 2008. Proceedings. Lecture Notes in Computer Science, vol. 5189, pp. 89–103. Springer (2008), https://dx.

doi.org/10.1007/978-3-540-85893-5_7

[108] Shibutani, K., Isobe, T., Hiwatari, H., Mitsuda, A., Akishita, T., Shirai, T.: Piccolo: An ultra-lightweight blockcipher. In: Preneel and Takagi [102], pp. 342–357, https://dx.doi.org/10.1007/978-3-642-23951-9_23

[109] Shirai, T., Shibutani, K., Akishita, T., Moriai, S., Iwata, T.: The 128-bit blockcipher CLEFIA (extended abstract). In: Biryukov, A. (ed.) Fast Software Encryption, 14th International Workshop, FSE 2007, Luxembourg, Luxembourg, March 26-28, 2007, Revised Selected Papers. Lecture Notes in Computer Science, vol. 4593, pp. 181–195. Springer (2007), https://dx.doi.org/10.1007/978-3-540-74619-5_12

[110] Soleimany, H.: Probabilistic Slide Cryptanalysis and Its Applications to LED-64 and Zorro. In: Cid, C., Rechberger, C. (eds.) Fast Software Encryption - 21st International Workshop, FSE 2014, London, UK, March 3-5, 2014. Revised Selected Papers. Lecture Notes in Computer Science, vol. 8540, pp. 373–389. Springer (2014), https://doi.org/10.1007/978-3-662-46706-0_19

[111] Song, J., Lee, K., Lee, H.: Biclique cryptanalysis on lightweight block cipher: HIGHT and Piccolo. Int. J. Comput. Math. 90(12), 2564–2580 (2013), https://doi.org/10.1080/00207160.2013.767445

[112] Song, L., Huang, Z., Yang, Q.: Automatic Differential Analysis of ARX Block Ciphers with Application to SPECK and LEA. In: Liu, J.K., Steinfeld, R. (eds.) Information Security and Privacy - 21st Australasian Conference, ACISP 2016, Melbourne, VIC, Australia, July 4-6, 2016, Proceedings, Part II. Lecture Notes in Computer Science, vol. 9723, pp. 379–394. Springer (2016), https://doi.org/10.1007/978-3-319-40367-0_24

[113] Sun, L., Wang, W., Wang, M.: Automatic Search of Bit-Based Division Property for ARX Ciphers and Word-Based Division Property. In: Takagi, T., Peyrin, T. (eds.) Advances in Cryptology - ASIACRYPT 2017 - 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part I. Lecture Notes in Computer Science, vol. 10624, pp. 128–157. Springer (2017), https://doi.org/10.1007/978-3-319-70694-8_5

[114] Sun, L., Wang, W., Wang, M.: More Accurate Differential Properties of LED64 and Midori64. IACR Trans. Symmetric Cryptol. 2018(3), 93–123 (2018), https://doi.org/10.13154/tosc.v2018.i3.93-123

[115] Sun, L., Wang, W., Wang, M.: MILP-aided bit-based division property for primitives with non-bit-permutation linear layers. IET Inf. Secur. 14(1), 12–20 (2020), https://doi.org/10.1049/iet-ifs.2018.5283

[116] Sun, L., Wang, W., Wang, M.: Accelerating the Search of Differential and Linear Characteristics with the SAT Method. IACR Trans. Symmetric Cryptol. 2021(1), 269–315 (2021), https://doi.org/10.46586/tosc.v2021.i1.269-315

[117] Suzaki, T., Minematsu, K.: Improving the Generalized Feistel. In: Hong, S., Iwata, T. (eds.) Fast Software Encryption, 17th International Workshop, FSE 2010, Seoul, Korea, February 7-10, 2010, Revised Selected Papers. Lecture Notes in Computer Science, vol. 6147, pp. 19–39. Springer (2010), https://doi.org/10.1007/978-3-642-13858-4_2

[118] Suzaki, T., Minematsu, K., Morioka, S., Kobayashi, E.: TWINE : A Lightweight Block Cipher for Multiple Platforms. In: Knudsen, L.R., Wu, H. (eds.) Selected Areas in Cryptography, 19th International Conference, SAC 2012, Windsor, ON, Canada, August 15-16, 2012, Revised Selected Papers. Lecture Notes in Computer Science, vol. 7707, pp. 339–354. Springer (2012), https://dx.doi.org/10.1007/978-3-642-35999-6_22

[119] Tezcan, C., Selçuk, A.A.: Improved improbable differential attacks on ISO standard CLEFIA: Expansion technique revisited. Inf. Process. Lett. 116(2), 136–143 (2016), https://doi.org/10.1016/j.ipl.2015.09.010

[120] Todo, Y., Leander, G., Sasaki, Y.: Nonlinear Invariant Attack - Practical Attack on Full SCREAM, iSCREAM, and Midori64. In: Cheon, J.H., Takagi, T. (eds.) Advances in Cryptology - ASIACRYPT 2016 - 22nd International Conference on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, December 4-8, 2016, Proceedings, Part II. Lecture Notes in Computer Science, vol. 10032, pp. 3–33 (2016), https://doi.org/10.1007/978-3-662-53890-6_1

[121] Todo, Y., Leander, G., Sasaki, Y.: Nonlinear Invariant Attack: Practical Attack on Full SCREAM, iSCREAM, and Midori64. J. Cryptol. 32(4), 1383–1422 (2019), https://doi.org/10.1007/s00145-018-9285-0

[122] Tolba, M., Abdelkhalek, A., Youssef, A.M.: Meet-in-the-Middle Attacks on Reduced Round Piccolo. In: Güneysu, T., Leander, G., Moradi, A. (eds.) Lightweight Cryptography for Security and Privacy - 4th International Workshop, LightSec 2015, Bochum, Germany, September 10-11, 2015, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9542, pp. 3–20. Springer (2015), https://doi.org/10.1007/978-3-319-29078-2_1

[123] Tolba, M., Abdelkhalek, A., Youssef, A.M.: Truncated and Multiple Differential Cryptanalysis of Reduced Round Midori128. In: Bishop, M., Nascimento, A.C.A. (eds.) Information Security - 19th International Conference, ISC 2016, Honolulu, HI, USA, September 3-6, 2016, Proceedings. Lecture Notes in Computer Science, vol. 9866, pp. 3–17. Springer (2016), https://doi.org/10.1007/978-3-319-45871-7_1

[124] Tolba, M., Youssef, A.M.: Generalized MitM attacks on full TWINE. Inf. Process. Lett. 116(2), 128–135 (2016), https://doi.org/10.1016/j.ipl.2015.09.011

[125] Tsunoo, Y., Tsujihara, E., Shigeri, M., Saito, T., Suzaki, T., Kubo, H.: Impossible Differential Cryptanalysis of CLEFIA. In: Nyberg, K. (ed.) Fast Software Encryption, 15th International Workshop, FSE 2008, Lausanne, Switzerland, February 10-13, 2008, Revised Selected Papers. Lecture Notes in Computer Science, vol. 5086, pp. 398–411. Springer (2008), https://doi.org/10.1007/978-3-540-71039-4_25

[126] Wang, G., Wang, G.: Improved Differential-ML Distinguisher: Machine Learning Based Generic Extension for Differential Analysis. In: Gao, D., Li, Q., Guan, X., Liao, X. (eds.) Information and Communications Security - 23rd International Conference, ICICS 2021, Chongqing, China, November 19-21, 2021, Proceedings, Part II. Lecture Notes in Computer Science, vol. 12919, pp. 21–38. Springer (2021), https://doi.org/10.1007/978-3-030-88052-1_2

[127] Wang, N., Wang, X., Jia, K., Zhao, J.: Differential attacks on reduced SIMON versions with dynamic key-guessing techniques. Sci. China Inf. Sci. 61(9), 098103:1–098103:3 (2018), https://doi.org/10.1007/s11432-017-9231-5

[128] Wang, X., Wu, B., Hou, L., Lin, D.: Automatic Search for Related-Key Differential Trails in SIMON-like Block Ciphers Based on MILP. In: Chen, L., Manulis, M., Schneider, S.A. (eds.) Information Security - 21st International Conference, ISC 2018, Guildford, UK, September 9-12, 2018, Proceedings. Lecture Notes in Computer Science, vol. 11060, pp. 116–131. Springer (2018), https://doi.org/10.1007/978-3-319-99136-8_7

[129] Wang, Y., Wu, W.: Improved Multidimensional Zero-Correlation Linear Cryptanalysis and Applications to LBlock and TWINE. In: Susilo, W., Mu, Y. (eds.) Information Security and Privacy - 19th Australasian Conference, ACISP 2014, Wollongong, NSW, Australia, July 7-9, 2014. Proceedings. Lecture Notes in Computer Science, vol. 8544, pp. 1–16. Springer (2014), https://doi.org/10.1007/978-3-319-08344-5_1

[130] Wei, Y., Xu, P., Rong, Y.: Related-key impossible differential cryptanalysis on lightweight cipher TWINE. J. Ambient Intell. Humaniz. Comput. 10(2), 509–517 (2019), https://doi.org/10.1007/s12652-017-0675-1

[131] Yi, W., Wu, B., Chen, S., Lin, D.: Improved Integral and Zero-correlation Linear Cryptanalysis of CLEFIA Block Cipher. In: Chen, K., Lin, D., Yung, M. (eds.) Information Security and Cryptology - 12th International Conference, Inscrypt 2016, Beijing, China, November 4-6, 2016, Revised Selected Papers. Lecture Notes in Computer Science, vol. 10143, pp. 33–46. Springer (2016), https://doi.org/10.1007/978-3-319-54705-3_3

[132] Zhao, H., Han, G.: Biclique cryptanalysis on Midori block cipher. Int. J. Embed. Syst. 11(2), 229–239 (2019), https://doi.org/10.1504/IJES.2019.098299

[133] Zhao, H., Han, G., Wang, L., Wang, W.: MILP-Based Differential Cryptanalysis on Round-Reduced Midori64. IEEE Access 8, 95888–95896 (2020), https://doi.org/10.1109/ACCESS.2020.2995795

[134] Zheng, L., Zhang, S.: FFT-based multidimensional linear attack on PRESENT using the 2-bit-fixed characteristic. Secur. Commun. Networks 8(18), 3535–3545 (2015), https://doi.org/10.1002/sec.1278

[135] Zheng, X., Jia, K.: Impossible Differential Attack on Reduced-Round TWINE. In: Lee, H., Han, D. (eds.) Information Security and Cryptology - ICISC 2013 - 16th International Conference, Seoul, Korea, November 27-29, 2013, Revised Selected Papers. Lecture Notes in Computer Science, vol. 8565, pp. 123–143. Springer (2013), https://doi.org/10.1007/978-3-319-12160-4_8

[136] 伊藤竜馬:「CRYPTREC 暗号技術ガイドライン（軽量暗号）」掲載の暗号方式に関する安全性評価の動向調査（文書番号：CRYPTREC EX-3101-2021）(2021), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3101-2021.pdf

4.2 ストリーム暗号

本節では軽量なストリーム暗号を取り上げる。ストリーム暗号はデータの秘匿を行うための関数だが、同じ機能を提供するブロック暗号と異なり、機能を秘匿のみに絞っている。したがって、（ソフトウェア実装での）プログラム領域や、（ハードウェア実装での）回路規模などリソースに関する制約が強く、複数の暗号化関数を実装できないような場合、たとえば単一の暗号化関数を用いてメッセージ認証子も生成したい、といった場合には、ブロック暗号を選択するのが適当である。また、多くのストリーム暗号アルゴリズムは、初期化に要する時間が長くなりがちであり、短いデータを処理する場合にはブロック暗号が適する場合が多い。逆に、秘匿機能だけで良いが、省リソースで高速な処理を行いたい、といった場合にはストリーム暗号が適する。

軽量暗号の中では、ストリーム暗号の提案／評価が先行して実施された経緯もあり、他に比べて成熟しているアルゴリズムが多い。そこで、本節では、安全性評価が十分に行われたと考えられる eStream portfolio [30] と ISO/IEC 29192-3 [58] に掲載されたアルゴリズムを中心に紹介する。eStream プロジェクトでは、ソフトウェア向けの Profile 1 とハードウェア向けの Profile 2 に分けてアルゴリズムの公募、評価を行っているが、本節では Profile 2 で portfolio に掲載された 3 つのアルゴリズム Grain v1 [37]、MICKEY 2.0 [4]、Trivium [12] を取り上げる。ソフトウェア向けの Profile 1 では 4 つのアルゴリズム HC-128、Rabbit、Salsa20/12、SOSEMANUK が portfolio に掲載されている。eBACS の PC、サーバ向け CPU を使った処理性能比較によれば、長いメッセージの処理については Salsa20/12 がもっとも優れている。ここでは、Salsa20/12 のかわりに、2015 年に RFC 化された ChaCha20 [10] を紹介する。ChaCha20 は、Salsa20/12 の改良版として開発されたソフトウェア向けのストリーム暗号であり、Salsa20/12 に比べると処理速度の点で幾分劣る。しかし、多くのオープンソースに採用されており、利用しやすさという点で優れている。ISO/IEC 29292-3 [58] は 2 つのアルゴリズム Trivium と Enocoro を掲載している。そこで、本節では、上記のアルゴリズムに加えて Enocoro を取り上げる。

各アルゴリズムの安全性解析状況については、4.1 節と同様、2021 年度に公開された CRYPTREC 外部評価報告書 [71] に基づき、2021 年 9 月時点の状況を記載している。

その他、軽量暗号に関する ISO 標準としては、RFID 向けの暗号技術を取り扱う ISO/IEC 29167 があるが、安全性の観点で望ましくない「XOR 暗号」が含まれており、CRYPTREC では一般に使用を奨めない。

技術分野	ストリーム暗号
名称	ChaCha20
設計者	Daniel J. Bernstein (The University of Illinois at Chicago/USA)
発表年	2008（SASC 2008 [10]）
仕様参照先	SASC 2008 [10]、IETF RFC 7539 [54]
特徴	鍵長 256 ビット、IV 長 96 ビットのストリーム暗号であり、秘密鍵、IV に加えて 128 ビットの定数、32 ビットのブロックカウンタを入力として 512 ビットの擬似乱数を出力する。アルゴリズムは 32 ビットワードの算術加算、排他的論理和、巡回シフトで構成されており、ソフトウェア実装に適する。特に、初期化処理がほとんど無いため、短いメッセージに対しても高速であるという特徴を持つ。また、アルゴリズム中でテーブル参照を行わないため、素直に実装してもキャッシュタイミング攻撃に対して安全である。
安全性解析状況	2021 年 9 月現在、様々な解析論文 [3, 7, 8, 13, 14, 15, 16, 19, 21, 22, 23, 51, 52, 53, 57] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。 20 段のうち 7 段に簡略化した ChaCha20 に対する最良の攻撃は、2020 年に提案された Beierle ら [8] による差分線形攻撃と 2021 年に提案された Coutinho ら [15, 16] による差分線形攻撃であり、Beierle ら [8] による攻撃では秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行でき、Coutinho ら [15, 16] による攻撃では効率的に識別攻撃が実行できる。また、20 段のうち 7.25 段に簡略化した ChaCha20 に対する最良の攻撃は、2021 年に提案された Miyashita ら [52] による差分攻撃であり、秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。
主な実装評価結果	eBASC [1] での評価（2016 年 6 月現在）によれば、Intel Core i5 上で 1.2 cycles/byte を達成している。また、FELICS [2] での評価（2016 年 6 月現在）によれば、Arm Cortex-M3 上の評価で、初期化に 144 cycles を要し、スループットは 54.3 cycles/byte である。
標準化状況	IETF RFC 7539 [54]
利用実績等	主に多項式型メッセージ認証子の Poly-1305 と組み合わせた認証暗号として利用される。Google が提供するサービスの通信路 (https) 保護に利用されている [11]。
オープンソース	OpenSSL, Google Chrome, Mozilla Firefox, OpenSSH など。

技術分野	ストリーム暗号
名称	Enocoro
設計者	Hitachi, Ltd.
発表年	2008（WAIS 2008）、2010（ISITA 2010 [66]）
仕様参照先	ISITA 2010 [66]、設計者ウェブサイト [72]
特徴	鍵長 80 ビットの Enocoro-80、鍵長 128 ビットの Enocoro-128v2 の 2 つのアルゴリズムから成る。いずれのアルゴリズムについても鍵長相当の安全性を謳っているが、鍵、IV を固定するごとに出力するデータはそれぞれ 2^{32} バイト、2^{64} バイトに制限されている。軽量ストリーム暗号には珍しく、8 ビット単位の処理で構成されており、ソフトウェア実装でも AES と同等の処理速度が得られる。
安全性解析状況	2021 年 9 月現在、様々な解析論文 [26, 66, 73, 74, 76, 77] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。 Enocoro-80 に対する最良の攻撃は、2015 年に提案された Ding ら [26] による弱鍵攻撃であり、Enocoro-80 の秘密鍵は確率 2^{-8} で弱鍵であり、2^{17} 個の選択 IV を用いることで、計算量 2^{48} で鍵回復攻撃を実行できる。Enocoro-128v2 に対する最良の攻撃は、2019 年に提案された船引ら [77] によるキューブ攻撃と 2021 年に提案された芝山ら [76] による高階差分攻撃であり、96 段のうち 11 段に簡略化した Enocoro-128v2 に対して秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行でき、96 段のうち 22 段に簡略化した Enocoro-128v2 に対して効率的に識別攻撃が実行できる [76]。
主な実装評価結果	Enocoro-80 [65]. Pentium 4 上での実装では、初期化に 1,335 cycles を要し、スループットが 27 cycles/byte である。また、ハードウェア実装 (ASIC) での性能は回路規模が 2.7 KGE, 処理速度が 2,197.6 Mbps（180nm プロセス、最大動作周波数 274.7 MHz）である。 Enocoro-128v2 [75]. Intel Core2 Duo 上での実装では、初期化に 1,530 cycles を要し、スループットが 14.8 cycles/byte である。また、ハードウェア実装 (ASIC) での性能は回路規模が 2.4 KGE, 処理速度が 6,250 Mbps（90 nm プロセス、最大動作周波数 781.3 MHz）である。
標準化状況	CRYPTREC 推奨候補暗号（Enocoro-128v2） [70]、 ISO/IEC 29192-3 [42]

技術分野	ストリーム暗号
名称	Grain v1
設計者	Martin Hell[1], Thomas Johansson[1], Willi Meier[2] (1: Lund University/Sweden, 2: FH Aargau/Switzerland)
発表年	2005（eSTREAM Project）
仕様参照先	eSTREAM ウェブサイト [29]
特徴	eSTREAM portfolio に選ばれた鍵長 80 ビット（IV 長 64 ビット）、鍵長 128 ビット（IV 長 80 ビット）の 2 つのアルゴリズムから成る、ハードウェア実装向けのストリーム暗号である。ビット単位で処理を行う線形フィードバックシフトレジスタ 1 個と非線形フィードバックシフトレジスタ 1 個を組み合わせている。軽量ストリーム暗号の中でも特にハードウェア実装の軽量性に優れている。ある程度の並列処理が可能であり、ソフトウェア実装でも実用に耐える。
安全性解析状況	2021 年 9 月現在、様々な解析論文 [5, 6, 9, 17, 18, 28, 45, 46, 48, 49, 50, 55, 56, 62, 68, 69] が発表されている。 単一鍵設定における最良の攻撃は、2018 年に提案された Todo ら [62] による高速相関攻撃であり、仕様段数において効率的に内部状態復元攻撃が実行できる。なお、本攻撃では $2^{76.7}$ の計算量と $2^{75.1}$ のデータ量が必要となる。
主な実装評価結果	Grain v1 のハードウェア実装性能については、Good らの評価 [33] が詳しい。この評価では、0.13 μm スタンダードセルライブラリを用いて評価を行っている。 **鍵長 80 ビット**. 回路規模 1,294 GE で最大動作周波数は 724.6 MHz、スループットは 724.6 MHz である。また、回路を 16 個まで並列処理することが可能である。 **鍵長 128 ビット**. 回路規模 1,857 GE で最大動作周波数は 925.9 MHz、スループットは 925.9 MHz である。また、回路を 32 個まで並列処理することが可能である。
標準化状況	Grain v1 をベースにした認証暗号 Grain-128A が ISO/IEC 29192-8 [43] と ISO/IEC 29167-13 [44] にて標準化されている。

技術分野	ストリーム暗号
名称	MICKEY 2.0
設計者	Steve Babbage[1], Matthew Dodd[2] (1: Vodafone/UK, 2: Independent consultant)
発表年	2005（eSTREAM Project）
仕様参照先	eSTREAM ウェブサイト [29]
特徴	eSTREAM portfolio に選ばれた鍵長 80 ビット、IV 長 80 ビットのハードウェア実装向けストリーム暗号である。1 つの鍵に対して、利用可能な IV の数は最大 2^{40} に制限されている。また、鍵と IV のペアに対して、利用可能な鍵ストリームは 2^{40} ビットに制限されている。線形フィードバックシフトレジスタ 1 個と非線形フィードバックシフトレジスタ 1 個を組み合わせており、不規則なクロック制御を行うことを特徴としている。このクロック制御機構が原因で、並列処理は困難である。
安全性解析状況	2021 年 9 月現在、様々な解析論文 [24, 25, 27, 38, 39] が発表されている。 単一鍵設定における最良の攻撃は、2019 年に提案された Ding ら [24] によって提案されたタイムメモリデータトレードオフ攻撃であり、仕様段数において秘密鍵の全数探索より効率的に鍵回復攻撃が実行できる。ただし、本攻撃は IV 長が 64 ビットの場合に成立し、$2^{79.0}$ の計算量、2^{80} のデータ量、$2^{45.0}$ のメモリ量が必要となる。
主な実装評価結果	MICKEY 2.0 のハードウェア実装性能については、Good らの評価 [33] が詳しい。この評価では、0.13 μm スタンダードセルライブラリを用いて評価を行っている。MICKEY 2.0 の回路規模は 3,188 GE であり、最大動作周波数は 454.5 MHz、スループットは 454.5 MHz である。

技術分野	ストリーム暗号
名称	Trivium
設計者	Christophe De Cannière, Bart Preneel (KU Leuven/Belgium)
発表年	2005（eSTREAM Project）
仕様参照先	eSTREAM ウェブサイト [29]
特徴	eSTREAM portfolio に掲載された鍵長 80 ビット、IV 長 80 ビットのハードウェア実装向けのストリーム暗号である。鍵と IV のペアごとに生成される鍵ストリームは 2^{64} ビットに制限される。3 つの非線形フィードバックシフトレジスタを直列した特徴的なアルゴリズムである。ビット単位の処理を基本としながら、高い並列性を持ち、ハードウェア実装での軽量性とソフトウェア実装での高速性を両立している。ただし、初期化に要する時間が長いため、短いデータの処理には適さない。
安全性解析状況	2021 年 9 月現在、様々な解析論文 [20, 31, 32, 34, 35, 36, 40, 41, 47, 59, 60, 61, 63, 64, 67] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。 最良の攻撃は、2021 年に提案された Hu ら [40] によるキューブ攻撃であり、1152 段のうち 845 段に簡略化した Trivium の初期化フェーズに対して、秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。
主な実装評価結果	Trivium のハードウェア実装性能については、Good らの評価 [33] が詳しい。この評価では、0.13 μm スタンダードセルライブラリを用いて評価を行っている。Trivium の回路規模は 2,580 GE であり、最大動作周波数は 327.9 MHz、スループットは 327.9 Mbps である。また、Trivium のアルゴリズムは最大で 64 並列で実行することが可能であり、このときの回路規模は 4,921 GE、スループットは 22,299.6 Mbps である。 また、ソフトウェア実装については、FELICS [2] で評価が行われている。Arm Cortex-M3 上でのスループットは 49.4 cycles/byte であり、ChaCha20 よりも高速である。ただし、初期化に 7,195 cycles を要するため、短いデータの処理には適さない。
標準化状況	ISO/IEC 29192-3 [42]

参考文献

[1] eBACS: ECRYPT Benchmarking of Cryptographic Systems, https://bench.cr.yp.to/results-stream.html（2023-10-04 閲覧）

[2] FELICS Stream Ciphers Brief Results, https://www.cryptolux.org/index.php/FELICS_Stream_Ciphers_Brief_Results（2023-10-04 閲覧）

[3] Aumasson, J., Fischer, S., Khazaei, S., Meier, W., Rechberger, C.: New Features of Latin Dances: Analysis of Salsa, ChaCha, and Rumba. In: Nyberg, K. (ed.) Fast Software Encryption, 15th International Workshop, FSE 2008, Lausanne, Switzerland, February 10-13, 2008, Revised Selected Papers. Lecture Notes in Computer Science, vol. 5086, pp. 470–488. Springer (2008), https://doi.org/10.1007/978-3-540-71039-4_30

[4] Babbage, S., Dodd, M.: The MICKEY Stream Ciphers. In: Robshaw, M., O. Billet, e. (eds.) New Stream Cipher Designs: The eSTREAM Finalists. Lecture Notes in Computer Science, vol. 4986, pp. 191–209. Springer (2008)

[5] Banik, S.: Some Insights into Differential Cryptanalysis of Grain v1. In: Susilo, W., Mu, Y. (eds.) Information Security and Privacy - 19th Australasian Conference, ACISP 2014, Wollongong, NSW, Australia, July 7-9, 2014. Proceedings. Lecture Notes in Computer Science, vol. 8544, pp. 34–49. Springer (2014), https://doi.org/10.1007/978-3-319-08344-5_3

[6] Banik, S.: Conditional differential cryptanalysis of 105 round Grain v1. Cryptogr. Commun. 8(1), 113–137 (2016), https://doi.org/10.1007/s12095-015-0146-5

[7] Barbero, S., Bellini, E., Makarim, R.H.: Rotational analysis of ChaCha permutation. IACR Cryptol. ePrint Arch. 2020, 1049 (2020), https://eprint.iacr.org/2020/1049

[8] Beierle, C., Leander, G., Todo, Y.: Improved Differential-Linear Attacks with Applications to ARX Ciphers. In: Micciancio, D., Ristenpart, T. (eds.) Advances in Cryptology - CRYPTO 2020 - 40th Annual International Cryptology Conference, CRYPTO 2020, Santa Barbara, CA, USA, August 17-21, 2020, Proceedings, Part III. Lecture Notes in Computer Science, vol. 12172, pp. 329–358. Springer (2020), https://doi.org/10.1007/978-3-030-56877-1_12

[9] Berbain, C., Gilbert, H., Maximov, A.: Cryptanalysis of Grain. In: Robshaw, M.J.B. (ed.) Fast Software Encryption, 13th International Workshop, FSE 2006, Graz, Austria, March 15-17, 2006, Revised Selected Papers. Lecture Notes in Computer Science, vol. 4047, pp. 15–29. Springer (2006), https://doi.org/10.1007/11799313_2

[10] Bernstein, D.J.: ChaCha, a variant of Salsa20. In: The State of the Art of Stream Ciphers, SASC 2008. ECRYPT (2008)

[11] Bursztein, E.: Google Security Blog: Speeding up and strengthening HTTPS connections for Chrome on Android (April 24, 2014) (2014), https://security.googleblog.com/2014/04/speeding-up-and-strengthening-https.html（2023-10-04 閲覧）

[12] Cannière, C.D.: Trivium: A Stream Cipher Construction Inspired by Block Cipher Design Principles. In: Katsikas, S.K., López, J., Backes, M., Gritzalis, S., Preneel, B. (eds.) Information Security, ISC 2006. Lecture Notes in Computer Science, vol. 4176, pp. 171–186. Springer Berlin Heidelberg (2006)

[13] Choudhuri, A.R., Maitra, S.: Significantly Improved Multi-bit Differentials for Reduced Round Salsa and ChaCha. IACR Trans. Symmetric Cryptol. 2016(2), 261–287 (2016), https://doi.org/10.13154/tosc.v2016.i2.261-287

[14] Coutinho, M., Neto, T.C.S.: New Multi-bit Differentials to Improve Attacks Against ChaCha. IACR Cryptol. ePrint Arch. 2020, 350 (2020), https://eprint.iacr.org/2020/350

[15] Coutinho, M., Neto, T.C.S.: Improved Linear Approximations to ARX Ciphers and Attacks Against ChaCha. IACR Cryptol. ePrint Arch. p. 224 (2021), https://eprint.iacr.org/2021/224

[16] Coutinho, M., Neto, T.C.S.: Improved Linear Approximations to ARX Ciphers and Attacks Against ChaCha. In: Canteaut, A., Standaert, F. (eds.) Advances in Cryptology - EUROCRYPT 2021 - 40th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, October 17-21, 2021, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12696, pp. 711–740. Springer (2021), https://doi.org/10.1007/978-3-030-77870-5_25

[17] Dalai, D.K., Maitra, S., Pal, S., Roy, D.: Distinguisher and non-randomness of Grain-v1 for 112, 114 and 116 initialisation rounds with multiple-bit difference in IVs. IET Inf. Secur. 13(6), 603–613 (2019), https://doi.org/10.1049/iet-ifs.2018.5276

[18] Dalai, D.K., Pal, S.: Recovering Internal States of Grain-v1. In: Heng, S., López, J. (eds.) Information Security Practice and Experience - 15th International Conference, ISPEC 2019, Kuala Lumpur, Malaysia, November 26-28, 2019, Proceedings. Lecture Notes in Computer Science, vol. 11879, pp. 325–337. Springer (2019), https://doi.org/10.1007/978-3-030-34339-2_18

[19] Deepthi, K.K.C., Singh, K.: Cryptanalysis for reduced round Salsa and ChaCha: revisited. IET Inf. Secur. 13(6), 591–602 (2019), https://doi.org/10.1049/iet-ifs.2018.5328

[20] Delaune, S., Derbez, P., Gontier, A., Prudhomme, C.: A Simpler Model for Recovering Superpoly on Trivium. IACR Cryptol. ePrint Arch. 2021, 1191 (2021), https://eprint.iacr.org/2021/1191, (accepted on *Selected Areas in Cryptography - 28th International Workshop, SAC 2021*)

[21] Dey, S., Sarkar, S.: Improved analysis for reduced round Salsa and Chacha. Discret. Appl. Math. 227, 58–69 (2017), https://doi.org/10.1016/j.dam.2017.04.034

[22] Dey, S., Sarkar, S.: Proving the biases of Salsa and ChaCha in differential attack. Des. Codes Cryptogr. 88(9), 1827–1856 (2020), https://doi.org/10.1007/s10623-020-00736-9

[23] Dey, S., Sarkar, S.: A theoretical investigation on the distinguishers of Salsa and ChaCha. Discret. Appl. Math. 302, 147–162 (2021), https://doi.org/10.1016/j.dam.2021.06.017

[24] Ding, L., Gu, D., Wang, L.: New Key Recovery Attack on the MICKEY Family of Stream Ciphers. In: Shen, B., Wang, B., Han, J., Yu, Y. (eds.) International Conference on Frontiers in Cyber Security - FCS 2019. Communications in Computer and Information Science, vol. 1105, pp. 239–249. Springer (2019), https://doi.org/10.1007/978-981-15-0818-9_16

[25] Ding, L., Guan, J.: Cryptanalysis of MICKEY family of stream ciphers. Secur. Commun. Networks 6(8), 936–941 (2013), https://doi.org/10.1002/sec.637

[26] Ding, L., Jin, C., Guan, J.: Slide attack on standard stream cipher Enocoro-80 in the related-key chosen IV setting. Pervasive Mob. Comput. 24, 224–230 (2015), https://doi.org/10.1016/j.pmcj.2015.08.002

[27] Ding, L., Jin, C., Guan, J., Qi, C.: New Treatment of the BSW Sampling and Its Applications to Stream Ciphers. In: Pointcheval, D., Vergnaud, D. (eds.) Progress in Cryptology - AFRICACRYPT 2014 - 7th International Conference on Cryptology in Africa, Marrakesh, Morocco, May 28-30, 2014. Proceedings. Lecture Notes in Computer Science, vol. 8469, pp. 136–146. Springer (2014), https://doi.org/10.1007/978-3-319-06734-6_9

[28] Dinur, I., Shamir, A.: Breaking Grain-128 with Dynamic Cube Attacks. In: Joux, A. (ed.) Fast Software Encryption - 18th International Workshop, FSE 2011, Lyngby, Denmark, February 13-16, 2011, Revised Selected Papers. Lecture Notes in Computer Science, vol. 6733, pp. 167–187. Springer (2011), https://doi.org/10.1007/978-3-642-21702-9_10

[29] ECRYPT: FELICS – Fair Evaluation of Lightweight Cryptographic Systems, https://www.ecrypt.eu.org/stream/project.html（2023-10-04 閲覧）

[30] of Excellence, E.N.: The eSTREAM Project, https://www.ecrypt.eu.org/stream/（2023-10-04 閲覧）

[31] Fouque, P., Vannet, T.: Improving Key Recovery to 784 and 799 Rounds of Trivium Using Optimized Cube Attacks. In: Moriai, S. (ed.) Fast Software Encryption - 20th International Workshop, FSE 2013, Singapore, March 11-13, 2013. Revised Selected Papers. Lecture Notes in Computer Science, vol. 8424, pp. 502–517. Springer (2013), https://doi.org/10.1007/978-3-662-43933-3_26

[32] Fu, X., Wang, X., Dong, X., Meier, W.: A Key-Recovery Attack on 855-round Trivium. In: Shacham, H., Boldyreva, A. (eds.) Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part II. Lecture Notes in Computer Science, vol. 10992, pp. 160–184. Springer (2018), https://doi.org/10.1007/978-3-319-96881-0_6

[33] Good, T., Benaissa, M.: Hardware performance of eStream phase-III stream cipher candidates. In: The State of the Art of Stream Ciphers, SASC 2008 (2008)

[34] Hao, Y., Isobe, T., Jiao, L., Li, C., Meier, W., Todo, Y., Wang, Q.: Improved Division Property Based Cube Attacks Exploiting Algebraic Properties of Superpoly. IEEE Trans. Computers 68(10), 1470–1486 (2019), https://doi.org/10.1109/TC.2019.2909871

[35] Hao, Y., Leander, G., Meier, W., Todo, Y., Wang, Q.: Modeling for Three-Subset Division Property Without Unknown Subset - Improved Cube Attacks Against Trivium and Grain-128AEAD. In: Canteaut, A., Ishai, Y. (eds.) Advances in Cryptology - EUROCRYPT 2020 - 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12105, pp. 466–495. Springer (2020), https://doi.org/10.1007/978-3-030-45721-1_17

[36] Hao, Y., Leander, G., Meier, W., Todo, Y., Wang, Q.: Modeling for Three-Subset Division Property without Unknown Subset. J. Cryptol. 34(3), 22 (2021), https://doi.org/10.1007/s00145-021-09383-2

[37] Hell, M., Johansson, T., Maximov, A., Meier, W.: The Grain Family of Stream Ciphers. In: Robshaw, M., O. Billet, e. (eds.) New Stream Cipher Designs: The eSTREAM Finalists. Lecture Notes in Computer Science, vol. 4986, pp. 179–190. Springer (2008)

[38] Helleseth, T., Jansen, C.J.A., Kazymyrov, O., Kholosha, A.: State space cryptanalysis of the MICKEY cipher. In: 2013 Information Theory and Applications Workshop, ITA 2013, San Diego, CA, USA, February 10-15, 2013. pp. 1–10. IEEE (2013), https://doi.org/10.1109/ITA.2013.6502941

[39] Hong, J., Kim, W.: TMD-Tradeoff and State Entropy Loss Considerations of Streamcipher MICKEY. In: Maitra, S., Madhavan, C.E.V., Venkatesan, R. (eds.) Progress in Cryptology - INDOCRYPT 2005, 6th International Conference on Cryptology in India, Bangalore, India, December 10-12, 2005, Proceedings. Lecture Notes in Computer Science, vol. 3797, pp. 169–182. Springer (2005), https://doi.org/10.1007/11596219_14

[40] Hu, K., Sun, S., Todo, Y., Wang, M., Wang, Q.: Massive Superpoly Recovery with Nested Monomial Predictions. In: Tibouchi, M., Wang, H. (eds.) Advances in Cryptology - ASIACRYPT 2021 - 27th International Conference on the Theory and Application of Cryptology and Information Security, Singapore, December 6-10, 2021, Proceedings, Part I. Lecture Notes in Computer Science, vol. 13090, pp. 392–421. Springer (2021), https://doi.org/10.1007/978-3-030-92062-3_14

[41] Hu, K., Sun, S., Wang, M., Wang, Q.: An Algebraic Formulation of the Division Property: Revisiting Degree Evaluations, Cube Attacks, and Key-Independent Sums. In: Moriai, S., Wang, H. (eds.) Advances in Cryptology - ASIACRYPT 2020 - 26th International Conference on the Theory and Application of Cryptology and Information Security, Daejeon, South Korea, December 7-11, 2020, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12491, pp. 446–476. Springer (2020), https://doi.org/10.1007/978-3-030-64837-4_15

[42] ISO/IEC: Information security – Lightweight cryptography – Part 3: Stream ciphers (ISO/IEC 29192-3:2012), https://www.iso.org/standard/56426.html

[43] ISO/IEC: Information security – Lightweight cryptography – Part 8: Authenticated encryption (ISO/IEC 29192-8:2022), https://www.iso.org/standard/80114.html

[44] ISO/IEC: Information technology – Automatic identification and data capture techniques – Part 13: Crypto suite Grain-128A security services for air interface communications (ISO/IEC 29167-13: 2015), https://www.iso.org/standard/60682.html

[45] Li, J., Guan, J.: Advanced conditional differential attack on Grain-like stream cipher and application on Grain v1. IET Inf. Secur. 13(2), 141–148 (2019), https://doi.org/10.1049/iet-ifs.2018.5180

[46] Li, J., Guan, J.: Improved Conditional Differential Attacks on Round-Reduced Grain v1. KSII Trans. Internet Inf. Syst. 12(9), 4548–4559 (2018), https://doi.org/10.3837/tiis.2018.09.023

[47] Liu, M., Yang, J., Wang, W., Lin, D.: Correlation Cube Attacks: From Weak-Key Distinguisher to Key Recovery. In: Nielsen, J.B., Rijmen, V. (eds.) Advances in Cryptology - EUROCRYPT 2018 - 37th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tel Aviv, Israel, April 29 - May 3, 2018 Proceedings, Part II. Lecture Notes in Computer Science, vol. 10821, pp. 715–744. Springer (2018), https://doi.org/10.1007/978-3-319-78375-8_23

[48] Ma, Z., Tian, T., Qi, W.: Improved conditional differential attacks on Grain v1. IET Inf. Secur. 11(1), 46–53 (2017), https://doi.org/10.1049/iet-ifs.2015.0427

[49] Ma, Z., Tian, T., Qi, W.: Internal state recovery of Grain v1 employing guess-and-determine attack. IET Inf. Secur. 11(6), 363–368 (2017), https://doi.org/10.1049/iet-ifs.2017.0232

[50] Ma, Z., Tian, T., Qi, W.: A New Distinguishing Attack on Grain-V1 with 111 Initialization Rounds. J. Syst. Sci. Complex. 32(3), 970–984 (2019), https://doi.org/10.1007/s11424-018-7170-4

[51] Maitra, S.: Chosen IV cryptanalysis on reduced round ChaCha and Salsa. Discret. Appl. Math. 208, 88–97 (2016), https://doi.org/10.1016/j.dam.2016.02.020

[52] Miyashita, S., Ito, R., Miyaji, A.: PNB-based Differential Cryptanalysis of ChaCha Stream Cipher. IACR Cryptol. ePrint Arch. 2021, 1537 (2021), https://eprint.iacr.org/2021/1537

[53] Neves, S., Araújo, F.: An observation on NORX, BLAKE2, and ChaCha. Inf. Process. Lett. 149, 1–5 (2019), https://doi.org/10.1016/j.ipl.2019.05.001

[54] Nir, Y., Langley, A.: ChaCha20 and Poly1305 for IETF Protocols, Request For Comments, vol. RFC7539 (May), https://tools.ietf.org/html/rfc7539

[55] Pan, S., Wu, Y., Wang, L.: Optimizing Fast Near Collision Attack on Grain Using Linear Programming. IEEE Access 7, 181191–181201 (2019), https://doi.org/10.1109/ACCESS.2019.2959334

[56] Rahimi, M., Barmshory, M., Mansouri, M.H., Aref, M.R.: Dynamic cube attack on Grain-v1. IET Inf. Secur. 10(4), 165–172 (2016), https://doi.org/10.1049/iet-ifs.2014.0239

[57] Shi, Z., Zhang, B., Feng, D., Wu, W.: Improved Key Recovery Attacks on Reduced-Round Salsa20 and ChaCha. In: Kwon, T., Lee, M., Kwon, D. (eds.) Information Security and Cryptology - ICISC 2012 - 15th International Conference, Seoul, Korea, November 28-30, 2012, Revised Selected Papers. Lecture Notes in Computer Science, vol. 7839, pp. 337–351. Springer (2012), https://doi.org/10.1007/978-3-642-37682-5_24

[58] for Standards, I.O.: ISO/IEC 29192-3:2012 Information technology – Security techniques – Lightweight cryptography – Part 3: Stream ciphers (October 2012)

[59] Sun, Y.: Cube Attack against 843-Round Trivium. IACR Cryptol. ePrint Arch. p. 547 (2021), https://eprint.iacr.org/2021/547

[60] Todo, Y., Isobe, T., Hao, Y., Meier, W.: Cube Attacks on Non-Blackbox Polynomials Based on Division Property. In: Katz, J., Shacham, H. (eds.) Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part III. Lecture Notes in Computer Science, vol. 10403, pp. 250–279. Springer (2017), https://doi.org/10.1007/978-3-319-63697-9_9

[61] Todo, Y., Isobe, T., Hao, Y., Meier, W.: Cube Attacks on Non-Blackbox Polynomials Based on Division Property. IEEE Trans. Computers 67(12), 1720–1736 (2018), https://doi.org/10.1109/TC.2018.2835480

[62] Todo, Y., Isobe, T., Meier, W., Aoki, K., Zhang, B.: Fast Correlation Attack Revisited - Cryptanalysis on

Full Grain-128a, Grain-128, and Grain-v1. In: Shacham, H., Boldyreva, A. (eds.) Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part II. Lecture Notes in Computer Science, vol. 10992, pp. 129–159. Springer (2018), https://doi.org/10.1007/978-3-319-96881-0_5

[63] Wang, Q., Hao, Y., Todo, Y., Li, C., Isobe, T., Meier, W.: Improved Division Property Based Cube Attacks Exploiting Algebraic Properties of Superpoly. In: Shacham, H., Boldyreva, A. (eds.) Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part I. Lecture Notes in Computer Science, vol. 10991, pp. 275–305. Springer (2018), https://doi.org/10.1007/978-3-319-96884-1_10

[64] Wang, S., Hu, B., Guan, J., Zhang, K., Shi, T.: MILP-aided Method of Searching Division Property Using Three Subsets and Applications. In: Galbraith, S.D., Moriai, S. (eds.) Advances in Cryptology - ASIACRYPT 2019 - 25th International Conference on the Theory and Application of Cryptology and Information Security, Kobe, Japan, December 8-12, 2019, Proceedings, Part III. Lecture Notes in Computer Science, vol. 11923, pp. 398–427. Springer (2019), https://doi.org/10.1007/978-3-030-34618-8_14

[65] Watanabe, D., Ideguchi, K., Kitahara, J., Muto, K., Furuichi, H., Kaneko, T.: Enocoro-80: A Hardware Oriented Stream Cipher. In: Second International Workshop on Advances in Information Security (2008)

[66] Watanabe, D., Owada, T., Okamoto, K., Igarashi, Y., Kaneko, T.: Update on Enocoro stream cipher. In: Proceedings of the International Symposium on Information Theory and its Applications, ISITA 2010, 17-20 October 2010, Taichung, Taiwan. pp. 778–783. IEEE (2010), https://doi.org/10.1109/ISITA.2010.5649627

[67] Ye, C., Tian, T.: Revisit Division Property Based Cube Attacks: Key-Recovery or Distinguishing Attacks? IACR Trans. Symmetric Cryptol. 2019(3), 81–102 (2019), https://doi.org/10.13154/tosc.v2019.i3.81-102

[68] Zhang, B., Li, Z., Feng, D., Lin, D.: Near Collision Attack on the Grain v1 Stream Cipher. In: Moriai, S. (ed.) Fast Software Encryption - 20th International Workshop, FSE 2013, Singapore, March 11-13, 2013. Revised Selected Papers. Lecture Notes in Computer Science, vol. 8424, pp. 518–538. Springer (2013), https://doi.org/10.1007/978-3-662-43933-3_27

[69] Zhang, B., Xu, C., Meier, W.: Fast Near Collision Attack on the Grain v1 Stream Cipher. In: Nielsen, J.B., Rijmen, V. (eds.) Advances in Cryptology - EUROCRYPT 2018 - 37th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tel Aviv, Israel, April 29 - May 3, 2018 Proceedings, Part II. Lecture Notes in Computer Science, vol. 10821, pp. 771–802. Springer (2018), https://doi.org/10.1007/978-3-319-78375-8_25

[70] デジタル庁・総務省・経済産業省: 電子政府における調達のために参照すべき暗号のリスト（CRYPTREC 暗号リスト）（文書番号：CRYPTREC LS-0001-2022）(2023), https://www.cryptrec.go.jp/list/cryptrec-ls-0001-2022.pdf

[71] 伊藤竜馬:「CRYPTREC 暗号技術ガイドライン（軽量暗号）」掲載の暗号方式に関する安全性評価の動向調査（文書番号：CRYPTREC EX-3101-2021）(2021), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3101-2021.pdf

[72] 株式会社日立製作所: 擬似乱数生成器 Enocoro, https://www.hitachi.co.jp/rd/yrl/crypto/enocoro/（2023-10-04 閲覧）

[73] 五十嵐保隆, 岡本和人, 金子敏信: 関連鍵攻撃による Enocoro-128v1.1 の弱鍵復元の検討 (II). In: 電子情報通信学会総合大会講演論文集 (2010)

[74] 五十嵐保隆, 岡本和人, 金子敏信: 関連鍵攻撃による Enocoro の弱鍵復元の検討. pp. 275–280 (2010)

[75] 三上修吾, 渡辺大: ストリーム暗号 Enocoro-128v2 のソフトウェアおよびハードウェア実装と評価. In: コンピュータセキュリティシンポジウム 2012 論文集. pp. 742–748 (2012)

[76] 芝山直喜, 五十嵐保隆, 金子敏信: ストリーム暗号 Enocoro-128v2 の高階差分特性. In: 暗号と情報セキュリティシンポジウム, SCIS2021, 1B1-4 (2021)

[77] 船引悠生, 藤堂洋介, 五十部孝典, 森井昌克: Enocoro-128v2 の Cube 攻撃に対する安全性評価. In: 暗号と情報セキュリティシンポジウム, SCIS2019, 2B1-1 (2019)

4.3 ハッシュ関数

本節では軽量ハッシュ関数について記載する。ここで取り上げるアルゴリズムは、軽量ハッシュ関数として主要国際会議等に採録実績のある Keccak、PHOTON、QUARK、SPONGENT を調査対象とする。ただし、軽量という観点から、Keccak は SHA-3 として選定されたフルスペックのものでは無く、置換関数のビット幅が小さいもののみを対象とする。 また、PHOTON、SPONGENT、Lesamnta-LW が軽量ハッシュ関数に関係する ISO/IEC（ISO/IEC 29192-5）[17] で規格化されている状況を鑑み、本稿では 2016 年度版ガイドライン [9] に掲載されていない Lesamnta-LW を新たな調査対象とし、その調査結果をまとめる。

各アルゴリズムの安全性解析状況については、4.1 節と同様、2021 年度に公開された CRYPTREC 外部評価報告書 [42] に基づき、2021 年 9 月時点の状況を記載している。なお、新たに調査対象として追加した Lesamnta-LW について、文献 [42] では安全性解析状況だけでなく、仕様等（設計者、発表年、仕様参照先、特徴、主な実装評価結果、標準化状況）もまとめられているため、文献 [42] の記載内容に従って調査結果をまとめた。 また、実装評価結果は基本的に提案論文から抽出しており、それぞれが同一環境で評価されたものではないことに注意されたい。

<table>
<tr><td>技術分野</td><td>ハッシュ関数</td></tr>
<tr><td>名称</td><td>Keccak</td></tr>
<tr><td>設計者</td><td>Guido Bertoni[1], Joan Daemen[1], Michaël Peeters[2], Gilles Van Assche[1]
(1: STMicroelectronics/Switzerland, 2: NXP Semiconductors/Belgium)</td></tr>
<tr><td>発表年</td><td>2008（NIST SHA-3 Competition）</td></tr>
<tr><td>仕様参照先</td><td>設計者ウェブサイト [10]</td></tr>
<tr><td>特徴</td><td>Keccak はスポンジ構造から成り、内部パラメータにより様々なバリエーションを構成できる。置換関数は 7 種類が定義されており、それぞれ Keccak-$f[b]$ ($b \in$ 25、50、100、200、400、800、1600) と表される。ここでは、軽量暗号の観点から、Keccak-$f[100]$、Keccak-$f[200]$、Keccak-$f[400]$ を利用した方式について掲載する。
<table>
<tr><th>Keccak-$f[b]$</th><th>n</th><th>r</th><th>r'</th><th>構成段数 [段]</th></tr>
<tr><td>Keccak-$f[100]$</td><td>80</td><td>20</td><td>20</td><td>16</td></tr>
<tr><td>Keccak-$f[200]$</td><td>64</td><td>72</td><td>72</td><td>18</td></tr>
<tr><td>Keccak-$f[400]$</td><td>128</td><td>144</td><td>144</td><td>20</td></tr>
</table>
$*$ n: 出力長、r: 入力ブロック長、r': 出力ブロック長</td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、SHA-3 として標準化された方式も含め、様々な解析論文 [5, 7, 8, 11, 12, 14, 19, 21, 22, 23, 24, 25, 26, 28, 29, 32, 33, 34, 35, 36] が発表されているが、仕様段数においてハッシュ関数の安全性基準を脅かす攻撃は発表されていない。
最良の衝突攻撃は、2021 年に提案された Boissier ら [5] による代数攻撃であり、2 段に簡略化した Keccak-$f[200]$ と 2 段に簡略化した Keccak-$f[400]$ に対して効率的に衝突計算を実行できる。
また、最良の原像攻撃は、2017 年に提案された Li ら [22] による攻撃であり、3 段に簡略化した Keccak-$f[400]$ に対して効率的に原像計算を実行できる。
最良の識別攻撃は、2011 年に Boura ら [8] によって提案されたゼロサム攻撃である。最大 24 段までの Keccak-f に対してゼロサム識別子が構成可能であると報告されているが、提案者 [8] が述べているように、この攻撃はハッシュ関数の安全性を脅かすものではない。その他、積分攻撃 [8, 36]、リバウンド攻撃 [11]、ブーメラン攻撃 [36]、差分攻撃 [26] などが報告されているが、これらの攻撃もまたハッシュ関数の安全性を脅かすものではない。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装 [20]（130nm process）
<table>
<tr><th></th><th>Area [GE]</th><th>Latency [cycles/block]</th><th>Throughput [kbps]</th></tr>
<tr><td>Keccak-$f[100]$</td><td>1,250</td><td>800</td><td>2.5</td></tr>
<tr><td>Keccak-$f[200]$</td><td>2,520</td><td>900</td><td>8.0</td></tr>
<tr><td>Keccak-$f[400]$</td><td>5,090</td><td>1,000</td><td>14.4</td></tr>
</table></td></tr>
<tr><td>標準化状況</td><td>Keccak-$f[1600]$ を利用した方式は SHA-3 (FIPS 202 [27]) に採用されている。</td></tr>
<tr><td>利用実績等</td><td>SHA-3 としては多くのアプリケーションで導入されつつある。
https://csrc.nist.gov/groups/STM/cavp/documents/sha3/sha3val.html
https://www.3gpp.org/DynaReport/35-series.html
（いずれも 2023-10-04 閲覧）</td></tr>
<tr><td>オープンソース</td><td>https://keccak.team/archives.html
https://github.com/gvanas/KeccakCodePackage
（いずれも 2023-10-04 閲覧）</td></tr>
</table>

<table>
<tr><td>技術分野</td><td>ハッシュ関数</td></tr>
<tr><td>名称</td><td>PHOTON</td></tr>
<tr><td>設計者</td><td>Jian Guo[1], Thomas Peyrin[2], Axel Poschmann[2]
(1: Institute for Infocomm Research/Singapore, 2: Nanyang Technological University/Singapore)</td></tr>
<tr><td>発表年</td><td>2011（CRYPTO 2011 [13]）</td></tr>
<tr><td>仕様参照先</td><td>CRYPTO 2011 [13]</td></tr>
<tr><td>特徴</td><td>PHOTON はスポンジ構造から成り、内部パラメータにより様々なバリエーションを構成できる。ISO/IEC 29192 では 5 種のバリエーションが示されている（下表）。
使用する暗号学的置換は AES と似た構成であり、AddConstants、SubCells、ShiftRows、MixColumnsSerial の 4 ステップを 12 ラウンド繰り返す。SubCells での変換には PRESENT の S-box を利用する。
<table>
<tr><th>PHOTON-$n/r/r'$</th><th>n</th><th>r</th><th>r'</th><th>構成段数 [段]</th></tr>
<tr><td>PHOTON-80/20/16</td><td>80</td><td>20</td><td>16</td><td>12</td></tr>
<tr><td>PHOTON-128/16/16</td><td>128</td><td>16</td><td>16</td><td>12</td></tr>
<tr><td>PHOTON-160/36/36</td><td>160</td><td>36</td><td>36</td><td>12</td></tr>
<tr><td>PHOTON-224/32/32</td><td>224</td><td>32</td><td>32</td><td>12</td></tr>
<tr><td>PHOTON-256/32/32</td><td>256</td><td>32</td><td>32</td><td>12</td></tr>
</table>
∗ n: 出力長、r: 入力ブロック長、r': 出力ブロック長</td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [18, 37, 38] が発表されているが、仕様段数においてハッシュ関数の安全性基準を脅かす攻撃は発表されていない。
最良の攻撃は、2017 年に提案された Wang ら [37, 38] によって提案されたゼロサム攻撃であり、11 段に簡略化した PHOTON-80 の暗号学的置換とフルスペックの PHOTON-128/160/224 の暗号学的置換に対して、それぞれ効率的に識別攻撃を実行できる。なお、PHOTON に対する識別攻撃は原則的にハッシュ関数の必須安全性基準（原像計算困難性、第 2 原像計算困難性、衝突困難性）を脅かすものではない。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装 [13]（180nm process）
<table>
<tr><th></th><th>Area [GE]</th><th>Latency [cycles/block]</th><th>Throughput [kbps]</th></tr>
<tr><td>PHOTON-80/20/16</td><td>865/1,168</td><td>708/132</td><td>2.82/15.15</td></tr>
<tr><td>PHOTON-128/16/16</td><td>1,122/1,708</td><td>996/156</td><td>1.61/10.26</td></tr>
<tr><td>PHOTON-160/36/36</td><td>1,396/2,117</td><td>1,332/180</td><td>2.70/20.00</td></tr>
<tr><td>PHOTON-224/32/32</td><td>1,735/2,786</td><td>1,716/204</td><td>1.86/15.69</td></tr>
<tr><td>PHOTON-256/32/32</td><td>2,177/4,362</td><td>996/156</td><td>3.21/20.51</td></tr>
</table>
ソフトウェア実装 [13] (Intel Core i7 @1.6GHz)
<table>
<tr><th></th><th>32-bit optimized implementation [cycles/byte]</th></tr>
<tr><td>PHOTON-80/20/16</td><td>95</td></tr>
<tr><td>PHOTON-128/16/16</td><td>156</td></tr>
<tr><td>PHOTON-160/36/36</td><td>116</td></tr>
<tr><td>PHOTON-224/32/32</td><td>227</td></tr>
<tr><td>PHOTON-256/32/32</td><td>135</td></tr>
</table></td></tr>
<tr><td>標準化状況</td><td>ISO/IEC 29192-5 [17]</td></tr>
</table>

<table>
<tr><td>技術分野</td><td>ハッシュ関数</td></tr>
<tr><td>名称</td><td>QUARK</td></tr>
<tr><td>設計者</td><td>Jean-Philippe Aumasson[1], Luca Henzen[2], Willi Meier[3], Maria Naya-Plasencia[3]
(1: Nagravision SA/Switzerland, 2: ETH Zurich/Switzerland, 3: FHNW/Switzerland)</td></tr>
<tr><td>発表年</td><td>2010（CHES 2010 [3]）</td></tr>
<tr><td>仕様参照先</td><td>CHES 2010 [3]、設計者 Web ページ [2]</td></tr>
<tr><td>特徴</td><td>QUARK はスポンジ構造から成り、内部パラメータにより様々なバリエーションを構成できる。パラメータの違いにより、U-QUARK、D-QUARK、S-QUARK の 3 種類が示されている。使用する暗号学的置換はストリーム暗号 Grain とブロック暗号 KATAN の利点を組み合わせた構成となっている。ラウンド数はそれぞれ 544、704、1024 である。
<table>
<tr><th></th><th>n</th><th>r</th><th>r'</th><th>構成段数 [段]</th></tr>
<tr><td>U-QUARK</td><td>128</td><td>8</td><td>8</td><td>544</td></tr>
<tr><td>D-QUARK</td><td>160</td><td>16</td><td>16</td><td>704</td></tr>
<tr><td>S-QUARK</td><td>224</td><td>32</td><td>32</td><td>1024</td></tr>
</table>
$*$ n: 出力長、r: 入力ブロック長、r': 出力ブロック長</td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [39, 41] が発表されているが、仕様段数においてハッシュ関数の安全性基準を脅かす攻撃は発表されていない。
最良の攻撃は、2018 年に提案された Yang ら [39] によって提案された条件付き差分攻撃であり、155、166、259 段に簡略化した U/D/S-QUARK に対して、それぞれ効率的に識別攻撃を実行できる。なお、QUARK に対する識別攻撃は原則的にハッシュ関数の必須安全性基準（原像計算困難性、第 2 原像計算困難性、衝突困難性）を脅かすものではない。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装 [3]（180nm process）
<table>
<tr><th></th><th>Area [GE]</th><th>Latency [cycles/block]</th><th>Throughput [kbps]</th></tr>
<tr><td>U-QUARK</td><td>1,379/2,392</td><td>544/68</td><td>1.47/11.76</td></tr>
<tr><td>D-QUARK</td><td>1,702/2,819</td><td>704/88</td><td>2.27/18.18</td></tr>
<tr><td>S-QUARK</td><td>2,296/4,640</td><td>1,024/64</td><td>3.13/50.00</td></tr>
</table></td></tr>
<tr><td>オープンソース</td><td><code>https://aumasson.jp/quark/</code>（2023-10-04 閲覧）
<code>https://github.com/veorq/Quark</code>（2023-10-04 閲覧）</td></tr>
</table>

<table>
<tr><td>技術分野</td><td>ハッシュ関数</td></tr>
<tr><td>名称</td><td>SPONGENT</td></tr>
<tr><td>設計者</td><td>Andrey Bogdanov[1], Miroslav Knežević[2], Gregor Leander[3], Deniz Toz[1], Kerem Varıcı[1], Ingrid Verbauwhede[1]
(1: KU Leuven/Belgium, 2: NXP Semiconductors/Belgium, 3: Technical University of Denmark/Denmark)</td></tr>
<tr><td>発表年</td><td>2011（CHES 2011 [4]）</td></tr>
<tr><td>仕様参照先</td><td>CHES 2011 [4]</td></tr>
<tr><td>特徴</td><td>SPONGENT は PRESENT タイプの暗号学的置換を用いたスポンジ構造から成り、内部パラメータにより様々なバリエーションを構成できる。提案者により 13 種のバリエーションが示されており、そのうち 5 種が ISO/IEC 29192-5 として標準化されている（表中の*印）。
<table>
<tr><th>SPONGENT-$n/c/r$</th><th>n</th><th>c</th><th>r</th><th>構成段数 [段]</th></tr>
<tr><td>SPONGENT-88/80/8*</td><td>88</td><td>80</td><td>8</td><td>45</td></tr>
<tr><td>SPONGENT-88/176/88</td><td>88</td><td>176</td><td>88</td><td>135</td></tr>
<tr><td>SPONGENT-128/128/8*</td><td>128</td><td>128</td><td>8</td><td>70</td></tr>
<tr><td>SPONGENT-128/256/128</td><td>128</td><td>256</td><td>128</td><td>195</td></tr>
<tr><td>SPONGENT-160/160/16*</td><td>160</td><td>160</td><td>16</td><td>90</td></tr>
<tr><td>SPONGENT-160/160/80</td><td>160</td><td>160</td><td>80</td><td>120</td></tr>
<tr><td>SPONGENT-160/320/160</td><td>160</td><td>320</td><td>160</td><td>240</td></tr>
<tr><td>SPONGENT-224/224/16*</td><td>224</td><td>224</td><td>16</td><td>120</td></tr>
<tr><td>SPONGENT-224/224/112</td><td>224</td><td>224</td><td>112</td><td>170</td></tr>
<tr><td>SPONGENT-224/448/224</td><td>224</td><td>448</td><td>224</td><td>340</td></tr>
<tr><td>SPONGENT-256/256/16*</td><td>256</td><td>256</td><td>16</td><td>140</td></tr>
<tr><td>SPONGENT-256/256/128</td><td>256</td><td>256</td><td>128</td><td>195</td></tr>
<tr><td>SPONGENT-256/512/256</td><td>256</td><td>512</td><td>256</td><td>385</td></tr>
</table>
∗ n: 出力長、c: capacity、r: rate(入力ブロック長)</td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、いくつかの解析論文 [1, 40] が発表されているが、仕様段数においてハッシュ関数の安全性基準を脅かす攻撃は発表されていない。
最良の攻撃は、2017 年に提案された Zhang ら [40] による切り詰め差分攻撃であり、簡略化した全バリエーションの SPONGENT の暗号学的置換に対して効率的に識別攻撃が実行できる。なお、SPONGENT に対する識別攻撃は原則的にハッシュ関数の必須安全性基準（原像計算困難性、第 2 原像計算困難性、衝突困難性）を脅かすものではない。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装 [4]（130nm process）
<table>
<tr><th></th><th>Area [GE]</th><th>Latency [cycles/block]</th><th>Throughput [kbps]</th></tr>
<tr><td>SPONGENT-88/80/8</td><td>738/1,127</td><td>990/45</td><td>0.81/17.78</td></tr>
<tr><td>SPONGENT-128/128/8</td><td>1,060/1,687</td><td>2,380/70</td><td>0.34/11.43</td></tr>
<tr><td>SPONGENT-160/160/16</td><td>1,329/2,190</td><td>3,960/90</td><td>0.40/17.78</td></tr>
<tr><td>SPONGENT-224/224/16</td><td>1,728/2,903</td><td>7,200/120</td><td>0.22/13.33</td></tr>
<tr><td>SPONGENT-256/256/16</td><td>1,950/3,281</td><td>9,520/140</td><td>0.17/11.43</td></tr>
</table></td></tr>
<tr><td>標準化状況</td><td>ISO/IEC 29192-5 [17]</td></tr>
</table>

<table>
<tr><td>技術分野</td><td>ハッシュ関数</td></tr>
<tr><td>名称</td><td>Lesamnta-LW</td></tr>
<tr><td>設計者</td><td>Shoichi Hirose[1], Kota Ideguchi[2], Hidenori Kuwakado[3], Toru Owada[2], Bart Preneel[4], Hirotaka Yoshida[2,4]
(1: University of Fukui/Japan, 2: Hitachi, Ltd./Japan, 3: Kobe University/Japan, 4: KU Leuven/Belgium)</td></tr>
<tr><td>発表年</td><td>2010（ICISC 2010 [15]）</td></tr>
<tr><td>仕様参照先</td><td>ICISC 2010 [15]</td></tr>
<tr><td>特徴</td><td>Lesamnta-LW は LW1 モードと呼ばれるドメイン拡張型の Merkle-Damgård 構造から成り、その基礎となるコンポーネントは AES ベースのブロック暗号（Lesamnta-LW-BC）を利用する。出力長は 256 ビットであり、原像攻撃や衝突攻撃に対して 2^{120} のセキュリティレベルを有するよう設計されている。なお、Lesamnta-LW は SHA-3 competition に応募された Lesamnta の軽量版として提案された。
Lesamnta-LW-BC は 4-branch type-1 一般化 Feistel network 型のブロック暗号であり、仕様段数は 64 段、ブロックサイズは 256 ビット、秘密鍵サイズは 128 ビット、ラウンド関数は AES のコンポーネントである MixColumns と SubBytes を使用する。</td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [6, 16, 30, 31] が発表されているが、仕様段数においてハッシュ関数の安全性基準を脅かす攻撃は発表されていない。
単一鍵設定における最良の攻撃は、2021 年に提案された Shiba ら [31] による積分攻撃であり、20 段に簡略化した Lesamnta-LW-BC に対して、秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。また、2020 年に提案された Hirose ら [16] による差分攻撃では、29 段に簡略化した Lesamnta-LW-BC に対して、効率的に識別攻撃が実行できる。既知鍵設定における最良の攻撃は、2021 年に提案された Shiba ら [31] によるゼロサム攻撃であり、47 段に簡略化した Lesamnta-LW-BC に対して、効率的に識別攻撃が実行できる。なお、Lesamnta-LW-BC に対する識別攻撃は原則的にハッシュ関数の必須安全性基準（原像計算困難性、第 2 原像計算困難性、衝突困難性）を脅かすものではない。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装 [15] (90nm Logic Process)
<table>
<tr><td>lgorithm</td><td>Area [GE]</td><td>Clock [MHz]</td><td>Throughput@30MHz [Mbit/s]</td></tr>
<tr><td>Lesamnta-LW</td><td>8,240</td><td>188.3</td><td>20.00</td></tr>
</table>
ソフトウェア実装評価結果 [15]
<table>
<tr><td>Algorithm</td><td>RAM [byte]</td><td>Cycles/byte</td><td>Platform</td></tr>
<tr><td>Lesamnta-LW</td><td>50</td><td>1,650.9</td><td>Renesas H8 (8-bit CPU)</td></tr>
<tr><td>Lesamnta-LW</td><td>–</td><td>39.5</td><td>Intel Core i5 (32-bit CPU)</td></tr>
</table></td></tr>
<tr><td>標準化状況</td><td>ISO/IEC 29192-5 [17]</td></tr>
<tr><td>オープンソース</td><td><code>https://github.com/kuwakado/Lesamnta-LW</code>（2023-10-04 閲覧）</td></tr>
</table>

参考文献

[1] Abdelraheem, M.A.: Estimating the Probabilities of Low-Weight Differential and Linear Approximations on PRESENT-Like Ciphers. In: Kwon, T., Lee, M., Kwon, D. (eds.) Information Security and Cryptology - ICISC 2012 - 15th International Conference, Seoul, Korea, November 28-30, 2012, Revised Selected Papers. Lecture Notes in Computer Science, vol. 7839, pp. 368–382. Springer (2012), https://doi.org/10.1007/978-3-642-37682-5_26

[2] Aumasson, J., Henzen, L., Meier, W., Naya-Plasencia, M.: Quark: a lightweight hash, https://www.aumasson.jp/quark/（2023-10-04 閲覧）

[3] Aumasson, J., Henzen, L., Meier, W., Naya-Plasencia, M.: Quark: A Lightweight Hash. In: Mangard, S., Standaert, F. (eds.) Cryptographic Hardware and Embedded Systems, CHES 2010, 12th International Workshop, Santa Barbara, CA, USA, August 17-20, 2010. Proceedings. Lecture Notes in Computer Science, vol. 6225, pp. 1–15. Springer (2010), https://doi.org/10.1007/978-3-642-15031-9_1

[4] Bogdanov, A., Knezevic, M., Leander, G., Toz, D., Varici, K., Verbauwhede, I.: SPONGENT: A Lightweight Hash Function. In: Preneel, B., Takagi, T. (eds.) Cryptographic Hardware and Embedded Systems - CHES 2011 - 13th International Workshop, Nara, Japan, September 28 - October 1, 2011. Proceedings. Lecture Notes in Computer Science, vol. 6917, pp. 312–325. Springer (2011), https://doi.org/10.1007/978-3-642-23951-9_21

[5] Boissier, R.H., Noûs, C., Rotella, Y.: Algebraic Collision Attacks on Keccak. IACR Trans. Symmetric Cryptol. 2021(1), 239–268 (2021), https://doi.org/10.46586/tosc.v2021.i1.239-268

[6] Bouillaguet, C., Dunkelman, O., Leurent, G., Fouque, P.: Attacks on Hash Functions Based on Generalized Feistel: Application to Reduced-Round *Lesamnta* and *SHAvite-3*$_{512}$. In: Biryukov, A., Gong, G., Stinson, D.R. (eds.) Selected Areas in Cryptography - 17th International Workshop, SAC 2010, Waterloo, Ontario, Canada, August 12-13, 2010, Revised Selected Papers. Lecture Notes in Computer Science, vol. 6544, pp. 18–35. Springer (2010), https://doi.org/10.1007/978-3-642-19574-7_2

[7] Boura, C., Canteaut, A.: Zero-sum distinguishers for iterated permutations and application to keccak-f and hamsi-256. In: Biryukov, A., Gong, G., Stinson, D.R. (eds.) Selected Areas in Cryptography - 17th International Workshop, SAC 2010, Waterloo, Ontario, Canada, August 12-13, 2010, Revised Selected Papers. Lecture Notes in Computer Science, vol. 6544, pp. 1–17. Springer (2010), https://doi.org/10.1007/978-3-642-19574-7_1

[8] Boura, C., Canteaut, A., Cannière, C.D.: Higher-order differential properties of keccak and *Luffa*. In: Joux, A. (ed.) Fast Software Encryption - 18th International Workshop, FSE 2011, Lyngby, Denmark, February 13-16, 2011, Revised Selected Papers. Lecture Notes in Computer Science, vol. 6733, pp. 252–269. Springer (2011), https://doi.org/10.1007/978-3-642-21702-9_15

[9] CRYPTREC 軽量暗号ワーキンググループ: CRYPTREC 暗号技術ガイドライン（軽量暗号）（文書番号：CRYPTREC GL-2003-2016JP）(2017), https://www.cryptrec.go.jp/report/cryptrec-gl-2003-2016jp.pdf

[10] Daemen, J., Hoffert, S., Peeters, M., Assche, G.V., Keer, R.V., Mella, S.: Team Keccak: Xoodyak, https://keccak.team/keccak.html

[11] Duc, A., Guo, J., Peyrin, T., Wei, L.: Unaligned rebound attack: Application to keccak. In: Canteaut, A. (ed.) Fast Software Encryption - 19th International Workshop, FSE 2012, Washington, DC, USA, March 19-21, 2012. Revised Selected Papers. Lecture Notes in Computer Science, vol. 7549, pp. 402–421. Springer (2012),

https://doi.org/10.1007/978-3-642-34047-5_23

[12] Guo, J., Liao, G., Liu, G., Liu, M., Qiao, K., Song, L.: Practical Collision Attacks against Round-Reduced SHA-3. J. Cryptol. 33(1), 228–270 (2020), https://doi.org/10.1007/s00145-019-09313-3

[13] Guo, J., Peyrin, T., Poschmann, A.: The PHOTON Family of Lightweight Hash Functions. In: Rogaway, P. (ed.) Advances in Cryptology - CRYPTO 2011 - 31st Annual Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2011. Proceedings. Lecture Notes in Computer Science, vol. 6841, pp. 222–239. Springer (2011), https://doi.org/10.1007/978-3-642-22792-9_13

[14] He, L., Lin, X., Yu, H.: Improved Preimage Attacks on 4-Round Keccak-224/256. IACR Trans. Symmetric Cryptol. 2021(1), 217–238 (2021), https://doi.org/10.46586/tosc.v2021.i1.217-238

[15] Hirose, S., Ideguchi, K., Kuwakado, H., Owada, T., Preneel, B., Yoshida, H.: A Lightweight 256-Bit Hash Function for Hardware and Low-End Devices: Lesamnta-LW. In: Rhee, K.H., Nyang, D. (eds.) Information Security and Cryptology - ICISC 2010 - 13th International Conference, Seoul, Korea, December 1-3, 2010, Revised Selected Papers. Lecture Notes in Computer Science, vol. 6829, pp. 151–168. Springer (2010), https://doi.org/10.1007/978-3-642-24209-0_10

[16] Hirose, S., Sasaki, Y., Yoshida, H.: Lesamnta-LW Revisited: Improved Security Analysis of Primitive and New PRF Mode. In: Conti, M., Zhou, J., Casalicchio, E., Spognardi, A. (eds.) Applied Cryptography and Network Security - 18th International Conference, ACNS 2020, Rome, Italy, October 19-22, 2020, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12146, pp. 89–109. Springer (2020), https://doi.org/10.1007/978-3-030-57808-4_5

[17] ISO/IEC: Information security – Security techniques – Lightweight cryptography – Part 5: Hash-functions (ISO/IEC 29192-5:2016), https://www.iso.org/standard/67173.html

[18] Jean, J., Naya-Plasencia, M., Peyrin, T.: Improved Rebound Attack on the Finalist Grøstl. In: Canteaut, A. (ed.) Fast Software Encryption - 19th International Workshop, FSE 2012, Washington, DC, USA, March 19-21, 2012. Revised Selected Papers. Lecture Notes in Computer Science, vol. 7549, pp. 110–126. Springer (2012), https://doi.org/10.1007/978-3-642-34047-5_7

[19] Jean, J., Nikolic, I.: Internal differential boomerangs: Practical analysis of the round-reduced keccak- f f permutation. In: Leander, G. (ed.) Fast Software Encryption - 22nd International Workshop, FSE 2015, Istanbul, Turkey, March 8-11, 2015, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9054, pp. 537–556. Springer (2015), https://doi.org/10.1007/978-3-662-48116-5_26

[20] Kavun, E.B., Yalçin, T.: A Lightweight Implementation of Keccak Hash Function for Radio-Frequency Identification Applications. In: Radio Frequency Identification: Security and Privacy Issues - 6th International Workshop, RFIDSec 2010, Istanbul, Turkey, June 8-9, 2010, Revised Selected Papers. pp. 258–269 (2010), https://dx.doi.org/10.1007/978-3-642-16822-2_20

[21] Li, T., Sun, Y.: Preimage Attacks on Round-Reduced Keccak-224/256 via an Allocating Approach. In: Ishai, Y., Rijmen, V. (eds.) Advances in Cryptology - EUROCRYPT 2019 - 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19-23, 2019, Proceedings, Part III. Lecture Notes in Computer Science, vol. 11478, pp. 556–584. Springer (2019), https://doi.org/10.1007/978-3-030-17659-4_19

[22] Li, T., Sun, Y., Liao, M., Wang, D.: Preimage Attacks on the Round-reduced Keccak with Cross-linear Structures. IACR Trans. Symmetric Cryptol. 2017(4), 39–57 (2017), https://doi.org/10.13154/tosc.v2017.i4.39-57

[23] Li, Z., Dong, X., Bi, W., Jia, K., Wang, X., Meier, W.: New Conditional Cube Attack on Keccak Keyed Modes. IACR Trans. Symmetric Cryptol. 2019(2), 94–124 (2019), https://doi.org/10.13154/tosc.v2019.i2.94-124

[24] Lin, X., He, L., Yu, H.: Improved Preimage Attacks on 3-Round Keccak-224/256. IACR Trans. Symmetric Cryptol. 2021(3), 84–101 (2021), https://doi.org/10.46586/tosc.v2021.i3.84-101

[25] Liu, G., Qiu, W., Tu, Y.: New Techniques for Searching Differential Trails in Keccak. IACR Trans. Symmetric

Cryptol. 2019(4), 407–437 (2019), https://doi.org/10.13154/tosc.v2019.i4.407-437

[26] Mella, S., Daemen, J., Assche, G.V.: New techniques for trail bounds and application to differential trails in keccak. IACR Trans. Symmetric Cryptol. 2017(1), 329–357 (2017), https://doi.org/10.13154/tosc.v2017.i1.329-357

[27] National Institute of Standards and Technology: FIPS 202 – SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions, https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.202.pdf

[28] Qiao, K., Song, L., Liu, M., Guo, J.: New Collision Attacks on Round-Reduced Keccak. In: Coron, J., Nielsen, J.B. (eds.) Advances in Cryptology - EUROCRYPT 2017 - 36th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Paris, France, April 30 - May 4, 2017, Proceedings, Part III. Lecture Notes in Computer Science, vol. 10212, pp. 216–243 (2017), https://doi.org/10.1007/978-3-319-56617-7_8

[29] Rajasree, M.S.: Cryptanalysis of Round-Reduced KECCAK Using Non-linear Structures. In: Hao, F., Ruj, S., Gupta, S.S. (eds.) Progress in Cryptology - INDOCRYPT 2019 - 20th International Conference on Cryptology in India, Hyderabad, India, December 15-18, 2019, Proceedings. Lecture Notes in Computer Science, vol. 11898, pp. 175–192. Springer (2019), https://doi.org/10.1007/978-3-030-35423-7_9

[30] Sasaki, Y., Aoki, K.: Improved Integral Analysis on Tweaked Lesamnta. In: Kim, H. (ed.) Information Security and Cryptology - ICISC 2011 - 14th International Conference, Seoul, Korea, November 30 - December 2, 2011. Revised Selected Papers. Lecture Notes in Computer Science, vol. 7259, pp. 1–17. Springer (2011), https://doi.org/10.1007/978-3-642-31912-9_1

[31] Shiba, R., Sakamoto, K., Liu, F., Minematsu, K., Isobe, T.: Integral and Impossible Differential Attacks on the Reduced-Round Lesamnta-LW-BC. In: 暗号と情報セキュリティシンポジウム, SCIS2021, 1B1-2 (2021)

[32] Song, L., Guo, J.: Cube-Attack-Like Cryptanalysis of Round-Reduced Keccak Using MILP. IACR Trans. Symmetric Cryptol. 2018(3), 182–214 (2018), https://doi.org/10.13154/tosc.v2018.i3.182-214

[33] Song, L., Guo, J., Shi, D., Ling, S.: New MILP Modeling: Improved Conditional Cube Attacks on Keccak-Based Constructions. In: Peyrin, T., Galbraith, S.D. (eds.) Advances in Cryptology - ASIACRYPT 2018 - 24th International Conference on the Theory and Application of Cryptology and Information Security, Brisbane, QLD, Australia, December 2-6, 2018, Proceedings, Part II. Lecture Notes in Computer Science, vol. 11273, pp. 65–95. Springer (2018), https://doi.org/10.1007/978-3-030-03329-3_3

[34] Song, L., Liao, G., Guo, J.: Non-full Sbox Linearization: Applications to Collision Attacks on Round-Reduced Keccak. In: Katz, J., Shacham, H. (eds.) Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part II. Lecture Notes in Computer Science, vol. 10402, pp. 428–451. Springer (2017), https://doi.org/10.1007/978-3-319-63715-0_15

[35] Suryawanshi, S., Saha, D., Sachan, S.: New Results on the SymSum Distinguisher on Round-Reduced SHA3. In: Nitaj, A., Youssef, A.M. (eds.) Progress in Cryptology - AFRICACRYPT 2020 - 12th International Conference on Cryptology in Africa, Cairo, Egypt, July 20-22, 2020, Proceedings. Lecture Notes in Computer Science, vol. 12174, pp. 132–151. Springer (2020), https://doi.org/10.1007/978-3-030-51938-4_7

[36] Todo, Y.: Structural evaluation by generalized integral property. In: Oswald, E., Fischlin, M. (eds.) Advances in Cryptology - EUROCRYPT 2015 - 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, April 26-30, 2015, Proceedings, Part I. Lecture Notes in Computer Science, vol. 9056, pp. 287–314. Springer (2015), https://doi.org/10.1007/978-3-662-46800-5_12

[37] Wang, Q., Grassi, L., Rechberger, C.: Zero-Sum Partitions of PHOTON Permutations. IACR Cryptol. ePrint Arch. 2017, 1211 (2017), https://eprint.iacr.org/2017/1211

[38] Wang, Q., Grassi, L., Rechberger, C.: Zero-Sum Partitions of PHOTON Permutations. In: Smart, N.P. (ed.) Topics in Cryptology - CT-RSA 2018 - The Cryptographers' Track at the RSA Conference 2018, San Francisco, CA, USA, April 16-20, 2018, Proceedings. Lecture Notes in Computer Science, vol. 10808, pp. 279–299. Springer

(2018), https://doi.org/10.1007/978-3-319-76953-0_15

[39] Yang, J., Liu, M., Lin, D., Wang, W.: Symbolic-Like Computation and Conditional Differential Cryptanalysis of QUARK. In: Inomata, A., Yasuda, K. (eds.) Advances in Information and Computer Security - 13th International Workshop on Security, IWSEC 2018, Sendai, Japan, September 3-5, 2018, Proceedings. Lecture Notes in Computer Science, vol. 11049, pp. 244–261. Springer (2018), https://doi.org/10.1007/978-3-319-97916-8_16

[40] Zhang, G., Liu, M.: A distinguisher on PRESENT-like permutations with application to SPONGENT. Sci. China Inf. Sci. 60(7), 72101 (2017), https://doi.org/10.1007/s11432-016-0165-6

[41] Zhang, K., Guan, J., Fei, X.: Improved conditional differential cryptanalysis. Secur. Commun. Networks 8(9), 1801–1811 (2015), https://doi.org/10.1002/sec.1144

[42] 伊藤竜馬:「CRYPTREC 暗号技術ガイドライン(軽量暗号)」掲載の暗号方式に関する安全性評価の動向調査(文書番号:CRYPTREC EX-3101-2021) (2021), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3101-2021.pdf

4.4 メッセージ認証コード

本節では、軽量なメッセージ認証コード (Message Authentication Code: MAC) を取り上げる。汎用的に用いられている MAC は、主にブロック暗号のモード (CMAC [38]) やハッシュ関数のモード (HMAC [19, 39]) である。CMAC や HMAC は、モード自体のオーバーヘッドがそれほど大きくないので、4.1 節に掲載されている軽量ブロック暗号や、4.3 節に掲載されている軽量ハッシュ関数と組み合せることで、軽量な MAC を構成することができる。HMAC はハッシュ関数を 2 回呼び出すため、処理するメッセージ長が非常に短い場合には、ブロック暗号ベースの CMAC を用いる方が効率的である可能性が高い。

ソフトウェア実装の性能に限れば、軽量暗号のベンチマークを行っている FELICS (Fair Evaluation of Lightweight Cryptographic Systems) プロジェクト [12] がブロック暗号やハッシュ関数を選択する参考になる。FELICS では、Atmel AVR ATmega128 (8-bit)、Texas Instruments MSP430F1611 (16-bit)、Arduino Due Arm Cortex-M3 (32-bit) 上で多数のアルゴリズム (ブロック暗号、ストリーム暗号、ハッシュ関数) を比較している。

ブロック暗号やハッシュ関数のモードではない、専用に設計された軽量 MAC はそれほど多く知られていないが、短いメッセージの処理に特化した擬似ランダム関数 SipHash [1, 2] は MAC として利用可能であり、多くの利用実績がある。ただし、SipHash は内部処理で 64-bit 加算などを利用しているので、「比較的ハイエンドの CPU 上で高速」という意味での軽量 MAC であり、8〜32-bit CPU での使用には適さない。

この他に、ローエンド CPU 向けの軽量 MAC として Chaskey [26, 28] がある。FELICS では Chaskey は軽量ブロック暗号に分類されており、多くの項目で最も優秀な成績を収めている。その一方で、8 ラウンド中 7 ラウンドについて鍵回復攻撃が可能であることが報告されており [20]、安全性の観点ではセキュリティマージンが小さい。この点を改善するために、ラウンド数を 8 から 12 に増やした Chaskey-12 [27] が提案されている。

本節では、2016 年度版ガイドライン [8] で掲載されている SipHash に加え、Chaskey、LightMAC、Tsudik's keymode が軽量 MAC に関係する ISO/IEC（ISO/IEC 29192-6）[15] で規格化されている状況を鑑み、これら 3 方式も新たな調査対象とし、その調査結果をまとめる。 なお、CMAC, HMAC については、本節では特に取り上げない。

SipHash の安全性解析状況については、4.1 節と同様、2021 年度に公開された CRYPTREC 外部評価報告書 [49] に基づき、2021 年 9 月時点の状況を記載している。なお、新たに調査対象として追加した Chaskey、LightMAC、Tsudik's keymode について、文献 [49] では安全性解析状況だけでなく、仕様等（設計者、発表年、仕様参照先、特徴、主な実装性能結果、標準化状況）もまとめられているため、文献 [49] の記載内容に従って調査結果をまとめた。また、Tsudik's keymode の安全性解析状況については、2022 年度に公開された CRYPTREC 外部評価報告書 [50] に基づき、2022 年 9 月現在の解析状況を記載している。

技術分野	メッセージ認証コード
名称	SipHash
設計者	Jean-Philippe Aumasson[1], Daniel J. Bernstein[2] (1: Kudelski Security/Switzerland, 2: University of Illinois at Chicago/USA)
発表年	2012（INDOCRYPT 2012 [2]）
仕様参照先	INDOCRYPT 2012 [2]、設計者ウェブサイト [1]
特徴	SipHash は、連想配列に用いるハッシュ関数として開発された鍵長 128 ビット、出力長 64 ビットの鍵付きハッシュ関数である。入力されるメッセージ長の上限は 2039 バイトであり、汎用のハッシュ関数に比べて短い。SipHash のアルゴリズムは c ラウンドの圧縮フェーズと d ラウンドの最終処理フェーズからなり、SipHash-c-d と表される。一般に利用されているのは、$c = 2, d = 4$ の SipHash-2-4 である。64 ビットワードを単位とし、算術加算、排他的論理和、巡回シフトを組み合わせたアルゴリズムであり、64 ビット演算をサポートする CPU 上で高速に動作する。また、アルゴリズム中でテーブル参照を行わないため、素直に実装してもキャッシュタイミング攻撃に対して安全である。
安全性解析状況	2021 年 9 月現在、様々な解析論文 [11, 22, 44] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。 SipHash-2-4 に対するに対する最良の鍵回復攻撃は、2014 年に提案された Dobraunig ら [11] による差分攻撃であり、差分特性確率 $2^{-236.3}$ の差分パスが発見されたが、鍵の総当たり攻撃の方がはるかに効率的であるので、この結果は SipHash の安全性を脅かすものではない。圧縮フェーズのみを簡略化した SipHash に対する最良の攻撃は、2019 年に提案された Xin ら [44] による差分攻撃であるが、この結果においても SipHash の安全性を脅かすものではない。
主な実装評価結果	提案論文 [2] によれば、SipHash のスループットは amd64 アーキテクチャ上で 1.5〜3.0 cycles/byte である。メッセージ長が短い場合には最終処理のオーバーヘッドが大きく、8 バイトのデータでは 10〜30 cycles/byte となる。
標準化状況	現時点では、(デジュールの) 標準化には提案されていない。しかし、多くのオープンソースライブラリに実装されており、デファクト標準の地位を固めつつある。
利用実績等	オープンソース、特に軽量プログラミング言語（Perl, Python, Ruby 等）の連想配列で用いるハッシュ関数として広く採用されている。この他にも、[1] によれば Wireguard, Bloomberg, OpenBSD, Shardmap, SoundHound, FreeBSD, Hashable, Rubinius, JRuby, Redis, OpenDNS, Rust, Sodium が SipHash を採用している。
オープンソース	前項を参照のこと。

<table>
<tr><td>技術分野</td><td>メッセージ認証コード</td></tr>
<tr><td>名称</td><td>Chaskey</td></tr>
<tr><td>設計者</td><td>Nicky Mouha[1], Bart Mennink[1], Anthony Van Herrewege[1], Dai Watanabe[2], Bart Preneel[1], Ingrid Verbauwhede[1]
(1: KU Leuven/Belgium, 2: Hitachi, Ltd./Japan)</td></tr>
<tr><td>発表年</td><td>2014（SAC 2014 [29])、2015（Cryptology ePrint Archive [27]）</td></tr>
<tr><td>仕様参照先</td><td>SAC 2014 [29]、Cryptology ePrint Archive [27]</td></tr>
<tr><td>特徴</td><td>Chaskey は、算術加算、排他的論理和、巡回シフトの組み合わせで構成される暗号学的置換を用いたメッセージ認証コードアルゴリズムである。暗号学的置換の仕様段数は 8 段又は 12 段、鍵長とブロックサイズは 128 ビット、タグ長は 64 ビット以上が推奨されている。なお、ISO/IEC 29192-6 [15] では 12 段の Chaskey が規格化されている。32 ビットワードを単位として演算が実行されることから、32 ビット演算をサポートするマイクロコントローラー上で効率的に動作する。また、全ての演算にかかる実行時間が一定であり、サイクル数がメッセージ長のみに依存するため、Chaskey はタイミング攻撃に対して安全である。</td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、いくつかの解析論文 [3, 6, 18, 20, 25, 45] が発表されている。
単一鍵設定における最良の鍵回復攻撃は、2021 年に提案された Broll ら [6] による差分線形攻撃であり、7.5 段に簡略化した Chaskey に対して、秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。また、Chaskey については弱鍵が存在し、関連鍵設定においてその弱鍵を使用している場合、仕様段数において効率的に鍵回復攻撃と偽造攻撃が実行できる [18]。</td></tr>
<tr><td>主な実装評価結果</td><td>ソフトウェア実装評価結果 [29]
<table>
<tr><th>Data [byte]</th><th>ROM [byte]</th><th>Cycles/byte</th><th>Platform</th></tr>
<tr><td>16</td><td>414</td><td>21.8</td><td>Cortex-M0</td></tr>
<tr><td>16</td><td>1,308</td><td>21.3</td><td>Cortex-M0</td></tr>
<tr><td>128</td><td>414</td><td>16.9</td><td>Cortex-M0</td></tr>
<tr><td>128</td><td>1,308</td><td>18.3</td><td>Cortex-M0</td></tr>
<tr><td>16</td><td>402</td><td>16.1</td><td>Cortex-M3/M4</td></tr>
<tr><td>16</td><td>908</td><td>10.6</td><td>Cortex-M3/M4</td></tr>
<tr><td>128</td><td>402</td><td>11.2</td><td>Cortex-M3/M4</td></tr>
<tr><td>128</td><td>908</td><td>7.0</td><td>Cortex-M3/M4</td></tr>
</table>
なお、Chaskey-12 [27] は Chaskey [29] と比較すると 32-bit Arm Cortex-M microcontrollers で 15 % 低速であると言及されている。
その他、効率的なソフトウェア実装の結果が文献 [10] で報告されている。</td></tr>
<tr><td>標準化状況</td><td>ISO/IEC 29192-6 [15]</td></tr>
</table>

<table>
<tr><td>技術分野</td><td>メッセージ認証コード</td></tr>
<tr><td>名称</td><td>LightMAC</td></tr>
<tr><td>設計者</td><td>Atul Luykx[1,2], Bart Preneel[1,2], Elmar Tischhauser[3], Kan Yasuda[4]
(1: KU Leuven/Belgium, 2: iMinds/Belgium, 3: Technical University of Denmark/ Denmark, 4: NTT/Japan)</td></tr>
<tr><td>発表年</td><td>2016（FSE 2016 [23]）</td></tr>
<tr><td>仕様参照先</td><td>FSE 2016 [23]</td></tr>
<tr><td>特徴</td><td>LightMAC は、ブロック暗号を利用した暗号利用モードによるメッセージ認証コードアルゴリズムである。従来のメッセージ認証技術では、ブロック長の短い軽量ブロック暗号を利用した場合、大きなデータを処理すると安全性が低下してしまうという課題があったが、LightMAC ではブロック暗号に対して独特の繰り返し方法を用いることにより、この課題を解決した。これにより LightMAC は既存の軽量ブロック暗号の実装を有効活用しつつ必要な安全性を確保することができる（参考記事：NTT 持株会社ニュースリリース [51]）。</td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [9, 30, 31, 43] が発表されている。
基礎となるブロック暗号として Simeck32/64 を利用した LightMAC に対し、3 種類の偽造攻撃が現実的な計算量で実行可能であることが報告されている [9, 43]。しかしながら、これらの攻撃は設計者 [23] が示す安全性上界のバウンドを脅かすものではない。</td></tr>
<tr><td>主な実装評価結果</td><td>ソフトウェア実装評価結果（Intel Core i7-6700 CPU） [23]
<table>
<tr><td rowspan="2">Underlying Block Cipher</td><td rowspan="2">Rate</td><td colspan="7">Message length [bytes]</td></tr>
<tr><td>128</td><td>256</td><td>512</td><td>1,024</td><td>2,048</td><td>4,096</td><td>8,192</td></tr>
<tr><td>PRESENT</td><td>1/2</td><td>25.50</td><td>23.67</td><td>22.75</td><td>22.32</td><td>22.08</td><td>21.97</td><td>21.92</td></tr>
<tr><td>PRESENT</td><td>2/3</td><td>25.70</td><td>21.21</td><td>20.17</td><td>19.03</td><td>18.09</td><td>17.80</td><td>17.80</td></tr>
<tr><td>PRESENT</td><td>7/8</td><td>20.31</td><td>18.34</td><td>14.65</td><td>13.48</td><td>–</td><td>–</td><td>–</td></tr>
<tr><td>AES</td><td>1/2</td><td>1.33</td><td>1.29</td><td>1.27</td><td>1.26</td><td>1.26</td><td>1.26</td><td>1.25</td></tr>
<tr><td>AES</td><td>2/3</td><td>1.37</td><td>1.31</td><td>1.12</td><td>1.04</td><td>0.95</td><td>0.95</td><td>0.92</td></tr>
<tr><td>AES</td><td>15/16</td><td>1.38</td><td>1.00</td><td>0.82</td><td>0.80</td><td>0.72</td><td>–</td><td>–</td></tr>
</table>
なお、数値は cycles/byte である。</td></tr>
<tr><td>標準化状況</td><td>ISO/IEC 29192-6 [15]</td></tr>
</table>

技術分野	メッセージ認証コード
名称	Tsudik's keymode
設計者	Gene Tsudik (University of Southern California/USA)
発表年	1992（ACM INFOCOM 1992 [40]）
仕様参照先	ACM INFOCOM 1992 [40]
特徴	Tsudik's keymode は、一方向性ハッシュ関数を用いた MAC であり、提案論文 [40] では MD4 を用いてアルゴリズムを紹介している。 $H : \{0,1\}^* \to \{0,1\}^n$ を出力長 n ビットのハッシュ関数とする。鍵長 k ビット、タグ長 t ビットの Tsudik's keymode $\mathsf{TKM} : \{0,1\}^k \times \{0,1\}^* \to \{0,1\}^t$ は、 1. $\mathsf{TKM}_K(M) = \lfloor H(K \parallel M) \rfloor_t$（secret prefix 方式） 2. $\mathsf{TKM}_K(M) = \lfloor H(M \parallel K) \rfloor_t$（secret suffix 方式） 3. $\mathsf{TKM}_{K,K'}(M) = \lfloor H(K \parallel M \parallel K') \rfloor_t$（ハイブリッド方式） の 3 種類が定義されている。なお、ISO/IEC 29192-6 [15] では secret prefix 方式のみ標準化されており、使用するハッシュ関数は ISO/IEC 29192-5 [16] で標準化されている PHOTON、SPONGENT、Lesamnta-LW の 3 方式が推奨されている。
安全性解析状況	ISO/IEC 29192-6 [15] において、国際標準方式である secret prefix 方式の安全性が述べられている。使用するハッシュ関数が衝突困難性を有すること、length-extension attack が実行できないことが要件として挙げられている。また、可変長入力のランダムオラクル [5] から強識別不可能性 [7, 24] を有するハッシュ関数であれば、Tsudik's keymode での使用に適していることが言及されている。推奨されているハッシュ関数の PHOTON と SPONGENT はこれに該当する。Lesamnta-LW は length-extension attack が実行可能な方式であるものの、設計者 [13] が Tsudik's keymode で使用した場合の擬似ランダム性を証明している。したがって、これら 3 方式は Tsudik's keymode での使用に適している。 Tsudik's keymode に対する第三者評価として、Preneel ら [17, 32, 34] による現実的な鍵回復攻撃と偽造攻撃が報告されている。この攻撃では length-extension attack が実行可能なハッシュ関数を使用した場合を想定しており、適切なハッシュ関数を使用することで攻撃を回避できる。その他、類似した方式に対するいくつかの解析結果 [4, 14, 21, 32, 33, 35, 36, 37, 41, 42, 46, 47, 48] が報告されている。これらの解析結果は Tsudik's keymode（特に、国際標準方式の secret prefix 方式）の安全性を脅かすものではない。
主な実装評価結果	Tsudik's keymode の実装性能は使用する一方向性ハッシュ関数に依存する。
標準化状況	ISO/IEC 29192-6 [15]

参考文献

[1] Aumasson, J.P.: SipHash: A Fast Short-input PRF, https://131002.net/siphash/（2023-10-04 閲覧）

[2] Aumasson, J.P., Bernstein, D.J.: SipHash: A Fast Short-Input PRF. In: Galbraith, S., Nandi, M. (eds.) Progress in Cryptology – INDOCRYPT 2012. Lecture Notes in Computer Science, vol. 7668, pp. 489–508. Springer-Verlag Berlin Heidelberg (2012)

[3] Beierle, C., Leander, G., Todo, Y.: Improved Differential-Linear Attacks with Applications to ARX Ciphers. In: Micciancio, D., Ristenpart, T. (eds.) Advances in Cryptology - CRYPTO 2020 - 40th Annual International Cryptology Conference, CRYPTO 2020, Santa Barbara, CA, USA, August 17-21, 2020, Proceedings, Part III. Lecture Notes in Computer Science, vol. 12172, pp. 329–358. Springer (2020), https://doi.org/10.1007/978-3-030-56877-1_12

[4] Bellare, M., Canetti, R., Krawczyk, H.: Pseudorandom Functions Revisited: The Cascade Construction and Its Concrete Security. In: 37th Annual Symposium on Foundations of Computer Science, FOCS '96, Burlington, Vermont, USA, 14-16 October, 1996. pp. 514–523. IEEE Computer Society (1996), https://doi.org/10.1109/SFCS.1996.548510

[5] Bellare, M., Rogaway, P.: Random Oracles are Practical: A Paradigm for Designing Efficient Protocols. In: Denning, D.E., Pyle, R., Ganesan, R., Sandhu, R.S., Ashby, V. (eds.) CCS '93, Proceedings of the 1st ACM Conference on Computer and Communications Security, Fairfax, Virginia, USA, November 3-5, 1993. pp. 62–73. ACM (1993), https://doi.org/10.1145/168588.168596

[6] Broll, M., Canale, F., David, N., Flórez-Gutiérrez, A., Leander, G., Naya-Plasencia, M., Todo, Y.: Further Improving Differential-Linear Attacks: Applications to Chaskey and Serpent. IACR Cryptol. ePrint Arch. 2021, 820 (2021), https://eprint.iacr.org/2021/820

[7] Coron, J., Dodis, Y., Malinaud, C., Puniya, P.: Merkle-Damgård Revisited: How to Construct a Hash Function. In: Shoup, V. (ed.) Advances in Cryptology - CRYPTO 2005: 25th Annual International Cryptology Conference, Santa Barbara, California, USA, August 14-18, 2005, Proceedings. Lecture Notes in Computer Science, vol. 3621, pp. 430–448. Springer (2005), https://doi.org/10.1007/11535218_26

[8] CRYPTREC 軽量暗号ワーキンググループ: CRYPTREC 暗号技術ガイドライン（軽量暗号）（文書番号：CRYPTREC GL-2003-2016JP）(2017), https://www.cryptrec.go.jp/report/cryptrec-gl-2003-2016jp.pdf

[9] Darumaya, T.A., Susanti, B.H.: Forgery Attack on LightMAC Hash Function Scheme using SIMECK 32/64 Lightweight Block Cipher. IOP Conference Series: Materials Science and Engineering 453, 012014 (nov 2018), https://doi.org/10.1088/1757-899x/453/1/012014

[10] Dinu, D., Corre, Y.L., Khovratovich, D., Perrin, L., Großschädl, J., Biryukov, A.: Triathlon of lightweight block ciphers for the Internet of things. J. Cryptogr. Eng. 9(3), 283–302 (2019), https://doi.org/10.1007/s13389-018-0193-x

[11] Dobraunig, C., Mendel, F., Schläffer, M.: Differential Cryptanalysis of SipHash. In: Joux, A., Youssef, A.M. (eds.) Selected Areas in Cryptography - SAC 2014 - 21st International Conference, Montreal, QC, Canada, August 14-15, 2014, Revised Selected Papers. Lecture Notes in Computer Science, vol. 8781, pp. 165–182. Springer (2014), https://doi.org/10.1007/978-3-319-13051-4_10

[12] FELICS: Fair Evaluation of Lightweight Cryptographic Systems, https://www.cryptolux.org/index.php/

[13] Hirose, S., Ideguchi, K., Kuwakado, H., Owada, T., Preneel, B., Yoshida, H.: An AES Based 256-bit Hash Function for Lightweight Applications: Lesamnta-LW. IEICE Trans. Fundam. Electron. Commun. Comput. Sci. 95-A(1), 89–99 (2012), https://doi.org/10.1587/transfun.E95.A.89

[14] Hosoyamada, A., Sasaki, Y.: Cryptanalysis Against Symmetric-Key Schemes with Online Classical Queries and Offline Quantum Computations. In: Smart, N.P. (ed.) Topics in Cryptology - CT-RSA 2018 - The Cryptographers' Track at the RSA Conference 2018, San Francisco, CA, USA, April 16-20, 2018, Proceedings. Lecture Notes in Computer Science, vol. 10808, pp. 198–218. Springer (2018), https://doi.org/10.1007/978-3-319-76953-0_11

[15] ISO/IEC: Information security – Lightweight cryptography – Part 6: Message authentication codes (MACs) (ISO/IEC 29192-6:2019), https://www.iso.org/standard/71116.html

[16] ISO/IEC: Information security – Security techniques – Lightweight cryptography – Part 5: Hash-functions (ISO/IEC 29192-5:2016), https://www.iso.org/standard/67173.html

[17] Koblitz, N., Menezes, A.: Another Look at Security Theorems for 1-Key Nested MACs. In: Koç, Ç.K. (ed.) Open Problems in Mathematics and Computational Science, pp. 69–89. Springer (2014), https://doi.org/10.1007/978-3-319-10683-0_4

[18] Kraleva, L., Ashur, T., Rijmen, V.: Rotational Cryptanalysis on MAC Algorithm Chaskey. In: Conti, M., Zhou, J., Casalicchio, E., Spognardi, A. (eds.) Applied Cryptography and Network Security - 18th International Conference, ACNS 2020, Rome, Italy, October 19-22, 2020, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12146, pp. 153–168. Springer (2020), https://doi.org/10.1007/978-3-030-57808-4_8

[19] Krawczyk, H., Bellare, M., Canetti, R.: HMAC: Keyed-Hashing for Message Authentication, Request For Comments, vol. RFC2104 (February), https://tools.ietf.org/html/rfc2104

[20] Leurent, G.: Improved Differential-Linear Cryptanalysis of 7-Round Chaskey with Partitioning. In: Fischlin, M., Coron, J. (eds.) Advances in Cryptology - EUROCRYPT 2016 - 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part I. Lecture Notes in Computer Science, vol. 9665, pp. 344–371. Springer (2016), https://doi.org/10.1007/978-3-662-49890-3_14

[21] Liu, F., Xie, T., Shen, C.: Breaking H^2-MAC Using Birthday Paradox. IACR Cryptol. ePrint Arch. p. 647 (2011), https://eprint.iacr.org/2011/647

[22] Liu, Y., Sun, S., Li, C.: Rotational Cryptanalysis from a Differential-Linear Perspective - Practical Distinguishers for Round-Reduced FRIET, Xoodoo, and Alzette. In: Canteaut, A., Standaert, F. (eds.) Advances in Cryptology - EUROCRYPT 2021 - 40th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, October 17-21, 2021, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12696, pp. 741–770. Springer (2021), https://doi.org/10.1007/978-3-030-77870-5_26

[23] Luykx, A., Preneel, B., Tischhauser, E., Yasuda, K.: A MAC Mode for Lightweight Block Ciphers. In: Peyrin, T. (ed.) Fast Software Encryption - 23rd International Conference, FSE 2016, Bochum, Germany, March 20-23, 2016, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9783, pp. 43–59. Springer (2016), https://doi.org/10.1007/978-3-662-52993-5_3

[24] Maurer, U.M., Renner, R., Holenstein, C.: Indifferentiability, Impossibility Results on Reductions, and Applications to the Random Oracle Methodology. In: Naor, M. (ed.) Theory of Cryptography, First Theory of Cryptography Conference, TCC 2004, Cambridge, MA, USA, February 19-21, 2004, Proceedings. Lecture Notes in Computer Science, vol. 2951, pp. 21–39. Springer (2004), https://doi.org/10.1007/978-3-540-24638-1_2

[25] Mavromati, C.: Key-Recovery Attacks Against the MAC Algorithm Chaskey. In: Dunkelman, O., Keliher, L. (eds.) Selected Areas in Cryptography - SAC 2015 - 22nd International Conference, Sackville, NB, Canada, August 12-14, 2015, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9566, pp. 205–216. Springer (2015),

https://doi.org/10.1007/978-3-319-31301-6_12

[26] Mouha, N.: Chaskey, https://mouha.be/chaskey/（2023-10-04 閲覧）

[27] Mouha, N.: Chaskey: A MAC Algorithm for Microcontrollers – Status Update and Proposal of Chaskey-12 –, https://eprint.iacr.org/2015/1182

[28] Mouha, N., Mennink, B., Herrewege, A.V., Watanabe, D., Preneel, B., Verbauwhede, I.: Chaskey: An Efficient MAC Algorithm for 32-bit Microcontrollers. In: Joux, A., Youssef, A. (eds.) Selected Areas in Cryptography – SAC 2014. Lecture Notes in Computer Science, vol. 8781, pp. 306–323. Springer (2014)

[29] Mouha, N., Mennink, B., Herrewege, A.V., Watanabe, D., Preneel, B., Verbauwhede, I.: Chaskey: An Efficient MAC Algorithm for 32-bit Microcontrollers. In: Joux, A., Youssef, A.M. (eds.) Selected Areas in Cryptography - SAC 2014 - 21st International Conference, Montreal, QC, Canada, August 14-15, 2014, Revised Selected Papers. Lecture Notes in Computer Science, vol. 8781, pp. 306–323. Springer (2014), https://doi.org/10.1007/978-3-319-13051-4_19

[30] Naito, Y.: Blockcipher-Based MACs: Beyond the Birthday Bound Without Message Length. In: Takagi, T., Peyrin, T. (eds.) Advances in Cryptology - ASIACRYPT 2017 - 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part III. Lecture Notes in Computer Science, vol. 10626, pp. 446–470. Springer (2017), https://doi.org/10.1007/978-3-319-70700-6_16

[31] Naito, Y.: Improved Security Bound of LightMAC_Plus and Its Single-Key Variant. In: Smart, N.P. (ed.) Topics in Cryptology - CT-RSA 2018 - The Cryptographers' Track at the RSA Conference 2018, San Francisco, CA, USA, April 16-20, 2018, Proceedings. Lecture Notes in Computer Science, vol. 10808, pp. 300–318. Springer (2018), https://doi.org/10.1007/978-3-319-76953-0_16

[32] Preneel, B., van Oorschot, P.C.: MDx-MAC and Building Fast MACs from Hash Functions. In: Coppersmith, D. (ed.) Advances in Cryptology - CRYPTO '95, 15th Annual International Cryptology Conference, Santa Barbara, California, USA, August 27-31, 1995, Proceedings. Lecture Notes in Computer Science, vol. 963, pp. 1–14. Springer (1995), https://doi.org/10.1007/3-540-44750-4_1

[33] Preneel, B., van Oorschot, P.C.: On the Security of Two MAC Algorithms. In: Maurer, U.M. (ed.) Advances in Cryptology - EUROCRYPT '96, International Conference on the Theory and Application of Cryptographic Techniques, Saragossa, Spain, May 12-16, 1996, Proceeding. Lecture Notes in Computer Science, vol. 1070, pp. 19–32. Springer (1996), https://doi.org/10.1007/3-540-68339-9_3

[34] Preneel, B., van Oorschot, P.C.: On the Security of Iterated Message Authentication Codes. IEEE Trans. Inf. Theory 45(1), 188–199 (1999), https://doi.org/10.1109/18.746787

[35] Qiao, S., Wang, W., Jia, K.: Distinguishing Attack on Secret Prefix MAC Instantiated with Reduced SHA-1. In: Lee, D.H., Hong, S. (eds.) Information, Security and Cryptology - ICISC 2009, 12th International Conference, Seoul, Korea, December 2-4, 2009, Revised Selected Papers. Lecture Notes in Computer Science, vol. 5984, pp. 349–361. Springer (2009), https://doi.org/10.1007/978-3-642-14423-3_23

[36] Sasaki, Y.: Cryptanalyses on a Merkle-Damgård Based MAC - Almost Universal Forgery and Distinguishing-H Attacks. In: Pointcheval, D., Johansson, T. (eds.) Advances in Cryptology - EUROCRYPT 2012 - 31st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cambridge, UK, April 15-19, 2012. Proceedings. Lecture Notes in Computer Science, vol. 7237, pp. 411–427. Springer (2012), https://doi.org/10.1007/978-3-642-29011-4_25

[37] Sasaki, Y.: Cryptanalyses on a Merkle-Damgård Based MAC - Almost Universal Forgery and Distinguishing-H Attacks. IEICE Trans. Fundam. Electron. Commun. Comput. Sci. 97-A(1), 167–176 (2014), https://doi.org/10.1587/transfun.E97.A.167

[38] of Standards, N.I., Technology: Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication. NIST Special Publication 800-38B (May 2005), https://csrc.nist.gov/publications/

nistpubs/800-38B/SP_800-38B.pdf

[39] of Standards, N.I., Technology: The Keyed-Hash Message Authentication Code (HMAC). Federal Information Processing Standards Publication FIPS 198-1 (July 2008), https://csrc.nist.gov/publications/fips/fips198-1/FIPS-198-1_final.pdf

[40] Tsudik, G.: Message Authentication with One-Way Hash Functions. In: Proceedings IEEE INFOCOM '92, The Conference on Computer Communications, Eleventh Annual Joint Conference of the IEEE Computer and Communications Societies, One World through Communications, Florence, Italy, May 4-8, 1992. pp. 2055–2059. IEEE Computer Society (1992), https://doi.org/10.1109/INFCOM.1992.263477

[41] Wang, G.: Distinguishing Attacks on LPMAC Based on the Full RIPEMD and Reduced-Step RIPEMD-{256, 320}. In: Lai, X., Yung, M., Lin, D. (eds.) Information Security and Cryptology - 6th International Conference, Inscrypt 2010, Shanghai, China, October 20-24, 2010, Revised Selected Papers. Lecture Notes in Computer Science, vol. 6584, pp. 199–217. Springer (2010), https://doi.org/10.1007/978-3-642-21518-6_15

[42] Wang, X., Wang, W., Jia, K., Wang, M.: New Distinguishing Attack on MAC Using Secret-Prefix Method. In: Dunkelman, O. (ed.) Fast Software Encryption, 16th International Workshop, FSE 2009, Leuven, Belgium, February 22-25, 2009, Revised Selected Papers. Lecture Notes in Computer Science, vol. 5665, pp. 363–374. Springer (2009), https://doi.org/10.1007/978-3-642-03317-9_22

[43] Windarta, S., Ramli, K., Sudiana, D.: Security Evaluation of LIGHTMAC: Second Preimage Attack using Existential Forgery. In: 2020 1st International Conference on Information Technology, Advanced Mechanical and Electrical Engineering (ICITAMEE). pp. 265–269. IEEE (2020)

[44] Xin, W., Liu, Y., Sun, B., Li, C.: Improved Cryptanalysis on SipHash. In: Mu, Y., Deng, R.H., Huang, X. (eds.) Cryptology and Network Security - 18th International Conference, CANS 2019, Fuzhou, China, October 25-27, 2019, Proceedings. Lecture Notes in Computer Science, vol. 11829, pp. 61–79. Springer (2019), https://doi.org/10.1007/978-3-030-31578-8_4

[45] Xu, Y., Wu, B., Lin, D.: Rotational-Linear Attack: A New Framework of Cryptanalysis on ARX Ciphers with Applications to Chaskey. In: Gao, D., Li, Q., Guan, X., Liao, X. (eds.) Information and Communications Security - 23rd International Conference, ICICS 2021, Chongqing, China, November 19-21, 2021, Proceedings, Part II. Lecture Notes in Computer Science, vol. 12919, pp. 192–209. Springer (2021), https://doi.org/10.1007/978-3-030-88052-1_12

[46] Yasuda, K.: "Sandwich" Is Indeed Secure: How to Authenticate a Message with Just One Hashing. In: Pieprzyk, J., Ghodosi, H., Dawson, E. (eds.) Information Security and Privacy, 12th Australasian Conference, ACISP 2007, Townsville, Australia, July 2-4, 2007, Proceedings. Lecture Notes in Computer Science, vol. 4586, pp. 355–369. Springer (2007), https://doi.org/10.1007/978-3-540-73458-1_26

[47] Yasuda, K.: HMAC without the "Second" Key. In: Samarati, P., Yung, M., Martinelli, F., Ardagna, C.A. (eds.) Information Security, 12th International Conference, ISC 2009, Pisa, Italy, September 7-9, 2009. Proceedings. Lecture Notes in Computer Science, vol. 5735, pp. 443–458. Springer (2009), https://doi.org/10.1007/978-3-642-04474-8_35

[48] Yu, H., Wang, X.: Distinguishing Attack on the Secret-Prefix MAC Based on the 39-Step SHA-256. In: Boyd, C., Nieto, J.M.G. (eds.) Information Security and Privacy, 14th Australasian Conference, ACISP 2009, Brisbane, Australia, July 1-3, 2009, Proceedings. Lecture Notes in Computer Science, vol. 5594, pp. 185–201. Springer (2009), https://doi.org/10.1007/978-3-642-02620-1_13

[49] 伊藤竜馬:「CRYPTREC 暗号技術ガイドライン(軽量暗号)」掲載の暗号方式に関する安全性評価の動向調査(文書番号:CRYPTREC EX-3101-2021) (2021), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3101-2021.pdf

[50] 岩田哲: 軽量暗号の安全性に関する調査及び評価(Photon-Beetle,Sparkle,Tsudik's keymode)(文書番号:CRYPTREC EX-3201-2022) (2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3201-2022.pdf

[51] 日本電信電話株式会社: NTT 持株会社ニュースリリース – IoT 向けメッセージ認証技術 LightMAC が ISO 標準に採

択 –, https://journal.ntt.co.jp/article/1004（2023-10-04 閲覧）

4.5 認証暗号

本節では、CAESAR コンペティション [10] において提案された方式のうち、軽量性を謳い、かつ安全性の観点で問題が見つかっていない方式を中心に調査結果をまとめる（2021 年 9 月現在)。 加えて、Grain-128A が軽量認証暗号に関係する ISO/IEC（ISO/IEC 29192-8）[49] で規格化されている状況を鑑み、本稿では 2016 年度版ガイドライン [23] に掲載されていない Grain-128A を新たな調査対象とし、その調査結果をまとめる。

各方式（Ascon を除く）の安全性解析状況については、2021 年度に公開された CRYPTREC 外部評価報告書 [114] に基づき、2021 年 9 月時点の状況を記載している。なお、新たに調査対象として追加した Grain-128A について、文献 [114] では安全性解析状況だけでなく、仕様等（設計者、発表年、仕様参照先、特徴、主な実装評価結果、標準化状況）もまとめられているため、文献 [114] の記載内容に従って調査結果をまとめた。

ここで示す方式には、ブロック暗号ないし tweakable ブロック暗号を用いているものも多い。これらの方式については理論的速度を測る指標としてレートを導入する。レートは 1 ブロック暗号で処理可能な入力ブロック数を表す。ソフトウェアの実装評価値は特に断りのない限り eBACS 内の Supercop ベンチマークシステム [11] での十分長いメッセージ処理の結果、ハードウェアの実装評価値も同様に特に断りのない限り ATHENA ベンチマークシステム [22] の結果である。ソフトウェアの評価尺度は十分長いメッセージでのバイトあたりの処理サイクル数（Cycles/Byte、C/B と略す)、ハードウェアでの評価は FPGA のスライス数（slices）と最大動作周波数（fmax)、ASIC ハードウェア実装の場合はサイズの評価尺度は Gate equivalent (GE) を用いるものとする。その他、これらの公式ベンチマークに当てはまらない注目すべき実装についても適宜報告する。いずれの場合も最適化実装の有無・最適化の度合いにより結果は大きく変わりうるため注意が必要である。著者の所属については提案時点のものである。

2019 年 2 月 20 日に CAESAR final portfolio が発表され、Use Case 1 (Lightweight Applications) として Ascon と ACORN、Use Case 2 (High-performance Applications) として AEGIS-128 と OCB、Use Case 3 (Defense in Depth) として Deoxys-II と COLM の合計 6 方式が選出された。final portfolio に選出された方式についてはその旨を記載している。なお、2016 年度版ガイドライン [23] ではこれら 6 方式のうち AEGIS-128 と COLM の 2 方式について掲載していない。これら 2 方式については軽量性を謳った方式ではないものの、CAESAR final portfolio に選出された方式であるということを鑑み、付録 B で調査結果をまとめた。

2021 年 3 月 30 日に NIST 軽量暗号（NIST LWC）プロジェクトのファイナリストが発表され、Ascon、Elephant、GIFT-COFB、Grain-128AEAD、ISAP、PHOTON-Beetle、Romulus、Sparkle、TinyJAMBU、Xoodyak の 10 方式がファイナリストとして選出された。その後、2023 年 2 月 7 日に最終選考結果が発表され、Ascon が最終選考方式として選出された。NIST LWC プロジェクトの動向を踏まえ、Ascon の記載内容については、2016 年度版ガイドライン [23] から大幅に更新されていることに注意が必要である。具体的には、2022 年度に公開された CRYPTREC 外部評価報告書 [115, 116] に基づき、2022 年 9 月現在の内容を記載している。また、Ascon を除くファイナリスト 9 方式についても軽量性の観点で優れており、かつ安全性の観点で問題が見つかっていない方式であることから、これらの方式も同様に付録 C で調査結果をまとめた。

Ascon の実装評価結果については文献 [115] に基づき更新している。文献 [115] で多くの実装評価結果がまとめられているものの、紙面の都合上、次の項目に限定している。ハードウェア実装評価結果については、FPGA 実装に着目し、回路面積の観点からコンパクト実装である結果、またはスループットの観点で高速実装である結果を抽出している。回路面積の評価尺度は、ルックアップテーブル数（LUTs）である。ソフトウェア実装評価結果については、IoT 向けローエンド CPU、特に Arm Cortex-M0 上での実装に着目し、設計者が作成したリファレンスコードを使用した場合のレイテンシ（暗号化・復号)、ROM サイズ、コードサイズの結果をまとめている。レイテンシの評価尺度は、テストベクトルを実行した際の 1 回の処理にかかる実行時間（msec）の平均値である。その他、文献 [115] では、ASIC 実装、命令拡張のハードウェア実装、ハイエンド CPU 上でのソフトウェア実装の結果がまとめられている。

<table>
<tr><td>技術分野</td><td>認証暗号、ハッシュ関数</td></tr>
<tr><td>名称</td><td>Ascon</td></tr>
<tr><td>設計者</td><td>Christoph Dobraunig[1,2], Maria Eichlseder[2], Florian Mendel[3], Martin Schlaffer[3]
(1: Radboud University/Netherlands, 2: Graz University of Technology/Austria, 3: Infineon Technologies AG/Germany)</td></tr>
<tr><td>発表年</td><td>2014（DIAC 2014 [29]）、2019（NIST LWC ウェブサイト [7]）</td></tr>
<tr><td>仕様参照先</td><td>CAESAR ウェブサイト [10]、NIST LWC ウェブサイト [31]、設計者ウェブサイト [28]</td></tr>
<tr><td>特徴</td><td>Ascon は暗号学的置換をプリミティブとして用いた MonkeyDuplex 構造 [13, 26] に基づく 2 つの認証暗号 Ascon-128、Ascon-128a と Sponge 構造 [12] に基づく 2 つのハッシュ関数 Ascon-Hash、Ascon-Hasha をまとめた総称である。ソフトウェアとハードウェアの両面で軽量性があること、そしてサイドチャネル耐性があることを主張している。
使用する暗号学的置換 p は SPN 型のラウンド関数であり、定数加算、非線形部（5 ビット S-box）、線形部（64 ビット単位の巡回シフトと XOR）で構成されている。ブロックサイズは 320 ビット、段数の異なる 2 種類の暗号学的置換（p^a、p^b）が使用される。また、認証暗号とハッシュ関数におけるパラメータの違いは下表のとおりであり、設計者が推奨する認証暗号は Ascon-128 で、ハッシュ関数は Ascon-Hash である。
<table>
<tr><th>名称</th><th>鍵長</th><th>nonce 長</th><th>タグ長</th><th>出力長</th><th>レート</th><th>p^a の段数</th><th>p^b の段数</th></tr>
<tr><td>Ascon-128</td><td>128</td><td>128</td><td>128</td><td>–</td><td>64</td><td>12</td><td>6</td></tr>
<tr><td>Ascon-128a</td><td>128</td><td>128</td><td>128</td><td>–</td><td>128</td><td>12</td><td>8</td></tr>
<tr><td>Ascon-Hash</td><td>–</td><td>–</td><td>–</td><td>256</td><td>64</td><td>12</td><td>12</td></tr>
<tr><td>Ascon-Hasha</td><td>–</td><td>–</td><td>–</td><td>256</td><td>64</td><td>12</td><td>8</td></tr>
</table>
なお、CAESAR final portfolio の Use Case 1 (Lightweight Applications) と NIST LWC プロジェクトの最終選考方式に選出された。</td></tr>
<tr><td>安全性解析状況</td><td>2022 年 9 月現在、いくつかの解析論文 [6, 30, 34, 37, 45, 66, 68, 71, 74, 86, 87, 112] が発表されているが、仕様書で主張される安全性を脅かす攻撃は提案されていない。
設計者 [31] は認証暗号とハッシュ関数の安全性がそれぞれ MonkeyDuplex 構造 [13, 26] と Sponge 構造 [12] の安全性に帰着できると主張している。さらに、認証暗号は一般的な MonkeyDuplex 構造とは異なり、初期化・最終処理フェーズで秘密鍵をそれぞれ 2 回適用していることから、安全性がさらに向上していると主張している。
認証暗号に対する最良の攻撃は、2021 年に提案された Rohit ら [86] によるキューブ攻撃と 2021 年に提案された Gerault ら [37] による差分攻撃であり、Rohit ら [86] は 12 段中 7 段の Ascon に対する鍵回復攻撃、Gerault ら [37] は 12 段中 4 段の Ascon に対する偽造攻撃を示した。ハッシュ関数に対する最良の攻撃は、2021 年に提案された Gerault ら [37] による差分攻撃であり、12 段中 2 段の Ascon に対して衝突攻撃が実行できる。プリミティブに対する最良の攻撃は設計者ら [30] によるゼロサム識別攻撃であり、フルラウンドの識別攻撃が可能であるが、この攻撃が認証暗号とハッシュ関数の安全性を脅かすものではないと主張されている。その他、Gerault ら [37] は 12 段のうち 7 段に簡略化したプリミティブに対して制限付き誕生日識別攻撃が実行できると報告している。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果（FPGA 実装）
<table>
<tr><th>Algorithm</th><th>Platform</th><th>Area</th><th>Throughput</th><th>Ref.</th></tr>
<tr><td>Ascon-128</td><td>Spartan-6</td><td>1,712 LUTs</td><td>2.88 Gbps</td><td>[35]</td></tr>
<tr><td>Ascon-128</td><td>Spartan-6</td><td>684 LUTs</td><td>60.10 Mbps</td><td>[104]</td></tr>
<tr><td>Ascon-Hash</td><td>Artix-7</td><td>2,181 LUTs</td><td>1.03 Gbps</td><td>[81]</td></tr>
</table>
ソフトウェア実装評価結果（Arm Cortex-M0）
<table>
<tr><th>Algorithm</th><th>Enc</th><th>Dec</th><th>ROM</th><th>Code</th><th>Ref.</th></tr>
<tr><td>Ascon-128</td><td>0.529 msec</td><td>0.536 msec</td><td>31.4 Kbyte</td><td>29.4 Kbyte</td><td>[44]</td></tr>
</table></td></tr>
</table>

技術分野	認証暗号
名称	ACORN
設計者	Hongjun Wu (Nanyang Technological University/Singapore)
発表年	2014 (DIAC 2014 [102])
仕様参照先	CAESAR ウェブサイト [10]
特徴	LFSR と単純な非線形処理を利用した方式。ハードウェア向けのストリーム暗号である Grain や Trivium と類似したシンプルな構造を持つ。 鍵は 128 ビットであり、LFSR を 6 つを組み合わせ、293 ビットを内部状態として保持する。 Grain や Trivium と同様にハードウェアに向いている。 なお、CAESAR final portfolio の Use Case 1 (Lightweight Applications) に選出された。
安全性解析状況	2021 年 9 月現在、様々な解析論文 [27, 40, 41, 42, 43, 57, 60, 88, 89, 101, 105, 107] が発表されているが、仕様において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。 最良の攻撃は、2021 年に提案された Hao ら [43] によるキューブ攻撃であり、1792 段のうち 775 段に簡略化した ACORN v3 の初期化フェーズに対しては、秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。
主な実装評価結果	(SW) Intel Core i5-6600 (Skylake 3.31 GHz) で 8.46 C/B。 (HW) Virtex 6 で 135 slices、fmax 389 MHz。

技術分野	認証暗号
名称	AES-JAMBU
設計者	Hongjun Wu, Tao Huang (Nanyang Technological University/Singapore)
発表年	2014 (DIAC 2014 [103])
仕様参照先	CAESAR ウェブサイト [10]
特徴	ブロック暗号利用モードである。ブロック暗号として AES-128 と SIMON [8] を採用。SIMON はブロックサイズ/鍵長（ビット）で 64/96、96/96、128/128 の 3 バージョンを指定。 ブロック暗号の入出力以外にブロックサイズの半分を状態変数として用いてシリアルに処理を行う。ブロック暗号 1 回ごとにブロックサイズの半分の暗号化を行う。状態変数のサイズが小さいため小規模ハードウェアに向いている。
安全性解析状況	一般的な暗号利用モードとは異なり、ブロック暗号の計算量的安全性に基づく安全性帰着を提案者は示していない。提案者の主張では k ビット鍵、$2n$ ビットブロック暗号のときに、暗号化の安全性で k ビット、認証の安全性で n ビットとしている。 2021 年 9 月現在、Peyrin ら [80] による解析論文の他、目立った解析論文は発表されていない。 2015 年に Peyrin ら [80] は、nonce-misuse シナリオにおいて $2^{n/2}$ 回の暗号化による攻撃と、nonce-respecting シナリオにおける CCA2 (adaptive chosen-ciphertext attack) 安全性 [9] に対する計算量 $2^{3n/2}$ の攻撃を報告している。なお、$n = 64$ である。
主な実装評価結果	(SW) AES-128 利用、 Intel Core i5-6600 (Skylake 3.31 GHz) で 5.71 C/B。 (HW) Virtex 6 で 453 slices、fmax 209.8 MHz。

技術分野	認証暗号
名称	AES-OTR
設計者	Kazuhiko Minematsu（NEC Corporation/Japan）
発表年	2014 (EUROCRYPT 2014 [76])
仕様参照先	CAESAR ウェブサイト [10]
特徴	ブロック暗号利用モードである。CAESAR 提案は AES を利用している。 OCB と類似した構造を持つ。2 ラウンドのフェイステル置換を用いており、ほぼ暗号化のみの計算量で処理が可能。並列処理も可能。OCB と異なり、認証暗号としての復号処理も AES 暗号化関数のみで実行可能であり、AES 復号を用いない。
安全性解析状況	提案論文にて、OTR の安全性がブロック暗号の擬似ランダム性（Pseudorandomness）へ帰着可能なことが示されている。n ビットブロック暗号の利用において $n/2$ ビットの証明可能安全性を有する。 2021 年 9 月現在、Bost ら [19] による解析論文の他、目立った解析論文は発表されていない。 2016 年に Bost ら [19] により内部のマスク生成における安全性証明との齟齬が指摘され、提案者により修正版が提案されている。
主な実装評価結果	(SW) AES-128 利用、Intel Core i5-6600 (Skylake 3.31 GHz) で 0.68 C/B。 (HW) Virtex 6 で 1,385 slices、fmax 256.9 MHz。 Arm v7 実装 [77]：1GHz Cortex-A8 マイコンボード上で 23.5 C/B (42.5 MByte/sec)。 Banik らによる ASIC 実装 [5]：部分的に外部メモリ利用、入力長の制約など加えた特殊条件下で実装し 6,000 GE 台

利用実績等	`https://www.nec-solutioninnovators.co.jp/ss/mobility/control.html`
	`https://www.nec-solutioninnovators.co.jp/sl/emb/pdf/automotive.pdf`
	（いずれも 2023-10-04 閲覧）

技術分野	認証暗号
名称	CLOC and SILC
設計者	Tetsu Iwata[1], Kazuhiko Minematsu[2], Jian Guo[3], Sumio Morioka[4], Eita Kobayashi[2] (1: Nagoya University/Japan, 2: NEC Corporation/Japan, 3: Nanyang Technological University/Singapore, 4: NEC Europe Ltd./UK)
発表年	2014（FSE 2014 [52]、DIAC 2014 [53]）
仕様参照先	CAESAR ウェブサイト [10]、FSE 2014 [52]、設計者ウェブサイト [51]
特徴	ブロック暗号利用モードである。CFB と CBC-MAC をベースにしたレート 1/2 の方式。鍵以外に必要なメモリ量が小さいのが特徴（n ビットブロック暗号利用で約 $2n$ ビット）。CLOC は処理のオーバーヘッドを削減し短い入力での性能向上を狙っており、組み込みソフトウェア向き。SILC は CLOC の処理を簡素化したハードウェア向けの方式。 CLOC、SILC ともに 128 ビットブロック暗号として AES を採用。64 ビットブロック暗号として CLOC は TWINE [96] を採用。SILC は PRESENT [18] および LED [38] を採用。
安全性解析状況	2021 年 9 月現在、目立った解析論文は発表されていない。 提案論文にて、CLOC と SILC の安全性が用いるブロック暗号の疑似ランダム性に帰着可能であることが示されている。n ビットブロック暗号を用いたとき $n/2$ ビットの安全性を有する。 nonce を誤って暗号化で重複させた場合でも暗号文の改ざんに対する安全性が保証されている。
主な実装評価結果	(SW) Intel Core i5-6600 (Skylake 3.31 GHz) で CLOC が 2.82 C/B。SILC は 2.78 C/B。 (HW) Virtex 6、 CLOC が 891 slices、fmax 280.9 MHz。SILC が 989 slices、fmax 280.7 MHz。 CLOC の 8 ビットマイコン実装 [52]: AVR ATmega128 (16 Mhz)。初期化に 2,000 サイクル、32 バイト暗号化に 550 C/B。 Banik らによる ASIC 実装 [5]：部分的に外部メモリ利用、入力長の制約など加えた特殊条件下で実装し、CLOC-AES、SILC-AES ともに約 3,100 GE。

技術分野	認証暗号
名称	Deoxys
設計者	Jérémy Jean, Ivica Nikolić, Thomas Peyrin (Nanyang Technological University/Singapore)
発表年	2014 (DIAC 2014 [54]、ASIACRYPT 2014 [55])
仕様参照先	CAESAR ウェブサイト [10]、ASIACRYPT 2014 [55]
特徴	専用 tweakable ブロック暗号 Deoxys-BC を利用するブロック暗号利用モード。 Deoxys-BC は 128 ビットブロック、256 ビット tweak+key、ラウンド関数は AES そのものであり、段数は 14 から 16 のいずれか。 ブロック暗号利用モードは TAE [70] と SCT [79] の 2 種類。TAE モードを用いる場合は 128 ビット安全性を有する。 TAE モードでは OCB 同様の実装面の特徴を有し、レート 1 での並列処理が可能である。一方の SCT は 2 パス、レート 1/2 のオフライン処理だが、SCT モードが deterministic AE (あるいは misuse-resistant AE) [85] の機能を有することにより、nonce の重複に対する安全性を持つ。 なお、Deoxys のバリエーションの 1 つである Deoxys-II が CAESAR final portfolio の Use Case 3 (Defense in Depth) に選出された。
安全性解析状況	2021 年 9 月現在、様々な解析論文 [21, 33, 63, 64, 72, 78, 79, 90, 108, 109, 113] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。 関連 tweakey 設定における最良の攻撃は、2019 年に提案された Zhao ら [108] と 2019 年に提案された Zhao ら [109] によって提案された rectangle attack であり、14 段のうち 13 段に簡略化した Deoxys-BC-256、16 段のうち 14 段に簡略化した Deoxys-BC-384、14 段のうち 10 段に簡略化した Deoxys-I-128-128、16 段のうち 13 段に簡略化した Deoxys-I-256-128 に対して、それぞれ秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。 なお、Deoxys-BC と Deoxys-I に関する解析論文がいくつか発表されているが、Deoxys-II に関する解析論文は発表されていない。
主な実装評価結果	(SW) Intel Core i5-6600 (Skylake 3.31 GHz) で 0.87 C/B。 (HW) Virtex 6 で 993 slices、fmax 330 MHz。 Deoxys-BC 単体の ASIC 実装が 2,860 GE [56]。

技術分野	認証暗号
名称	Joltik
設計者	Jérémy Jean, Ivica Nikolić, Thomas Peyrin (Nanyang Technological University/Singapore)
発表年 (発表学会等)	2014 (DIAC 2014 [54]、ASIACRYPT 2014 [55])
仕様参照先	CAESAR ウェブサイト [10]、ASIACRYPT 2014 [55]
特徴	専用 tweakable ブロック暗号 Joltik-BC を利用。 Joltik-BC は 64 ビットブロック、128 ビット tweak+key、ラウンド関数は 4 ビット S-box を用いた SPN 構造、段数は 24 から 32 のいずれか。 Deoxys 同様、モードは TAE と SCT の 2 種類である。
安全性解析状況	2021 年 9 月現在、様々な解析論文 [62, 65, 73, 111] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。 単一鍵設定における最良の攻撃、2019 年に提案されたは Li ら [65] による中間一致攻撃であり、24 段のうち 8 段に簡略化した Joltik-BC-128、32 段のうち 10 段に簡略化した Joltik-BC-192 に対して、それぞれ秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。関連 tweakey 設定における最長の攻撃は、2021 年に提案された Li ら [62] による中間一致攻撃であり、24 段のうち 9 段に簡略化した Joltik-BC-128、32 段のうち 11 段に簡略化した Joltik-BC-192 に対して、それぞれ秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。 なお、Joltik-BC に関する解析論文がいくつか発表されているが、認証暗号としての Joltik に関する解析論文は発表されていない。
主な実装評価結果	(SW) Intel Core i5-6600 (Skylake 3.31 GHz) で 13.32 C/B。 (HW) Virtex 6 で 494 slices、fmax 430 MHz。

技術分野	認証暗号
名称	Ketje
設計者	Guido Bertoni[1], Joan Daemen[1], Michael Peeters[2], Gilles Van Assche[1], Ronny Van Keer[1] (1: STMicroelectronics/Switzerland, 2: NXP Semiconductors/Belgium)
発表年	2014 (DIAC 2014 [15])
仕様参照先	CAESAR ウェブサイト [10]、設計者ウェブサイト [25]
特徴	Sponge 構造を持つ。利用モードは MonkeyDuplex [24] がベースとなる。 内部の暗号学的置換は Keccak-p と呼ばれ、SHA-3 関数で用いられる Keccak-f 置換 [14] をベースとしたものである。200 ビット幅の置換を利用するものを Ketje-JR、400 ビット幅のものを Ketje-SR と呼ぶ。 メモリサイズの小ささと計算量の少なさによる、ハード・ソフト両面での軽量性を謳っている。
安全性解析状況	2021 年 9 月現在、様々な解析論文 [17, 32, 36, 67, 94, 95, 110] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。 最良の攻撃は、2018 年に提案された Song [94] によるキューブ攻撃、2018 年に提案された Song ら [95] によるキューブ攻撃、2019 年に提案された Li ら [67] による条件付きキューブ攻撃、2021 年に提案された Zhao ら [110] によるキューブ攻撃であり、12 段のうち 5 段に簡略化した Ketje Jr、12 段のうち 7 段に簡略化した Ketje Sr/Minor/Major に対して、それぞれ秘密鍵の全数探索よりも効率的に鍵回復攻撃が実行できる。また、Ketje Jr に対する内部状態復元攻撃が Fuhr ら [36] によって 2018 年に提案され、ビットレートが 40 の場合には仕様段数であっても秘密鍵の全数探索より効率的に内部状態復元攻撃が実行できるものの、推奨パラメータであるビットレートが 16 の場合には秘密鍵の全数探索より効率的に内部状態復元攻撃が実行できない。
主な実装評価結果	(SW) Ketje-SR、 Intel Core i5-6600 (Skylake 3.31 GHz) で 42.57 C/B。 (HW) Virtex 6 で 456 slices、fmax 229.5 MHz。

技術分野	認証暗号
名称	Minalpher
設計者	Yu Sasaki[1], Yosuke Todo[1], Kazumaro Aoki[1], Yusuke Naito[2], Takeshi Sugawara[2], Yumiko Murakami[2], Mitsuru Matsui[2], Shoichi Hirose[3] (1: NTT/Japan, 2: Mitsubishi Electric Corporation/Japan, 3: University of Fukui/Japan)
発表年	2014 (DIAC 2014 [93])
仕様参照先	CAESAR ウェブサイト [10]
特徴	256 ビットの暗号学的置換 Minalpher-P を用いた専用 256 ビット Tweakable Even-Mansour ブロック暗号 (TEM) を利用、モードは独自方式。 TEM が用いる内部の置換は 4 ビット S-box 利用の SPN 構造、暗号化・復号関数の統合が容易となる構造を採用している。nonce の重複に対し部分的な安全性を有する。
安全性解析状況	2021 年 9 月現在、様々な解析論文 [20, 39, 91] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。 最良の攻撃は、2017 年に提案された佐々木ら [91] による不能差分攻撃であり 17.5 段のうち 7.5 段に簡略化した Minalpher に対して、効率的に識別攻撃を実行できる。
主な実装評価結果	(SW) Intel Core i5-6600 (Skylake 3.31 GHz) で 5.81 C/B。 (HW) Virtex 6 で 1,104 slices、fmax 280.9 MHz。 SIMD 実装 [92] : Intel CPU Core i7 (Haswell) で 5.6 C/B。 8 ビット RL78 マイコン実装 [92]: 510 ROM、214 RAM バイトの利用で約 2,800 C/B、1,275 ROM、470 RAM バイトの利用で 514 C/B。

技術分野	認証暗号
名称	OCB
設計者	Ted Krovetz[1], Phillip Rogaway[2] (1: California State University/USA, 2: University of California/USA)
発表年	2001 (ACM CCS 2001 [84])、 2004 (ASIACRYPT 2004 [82])、 2011 (FSE 2011 [58])
仕様参照先	CAESAR ウェブサイト [10] 公式ウェブサイト `http://web.cs.ucdavis.edu/~rogaway/ocb/`
特徴	ブロック暗号利用モードである。AES を利用したバージョンが IETF RFC 7253 [59] にて規定されている。CAESAR 提案は IETF RFC 7253 と同じ。ECB モードと類似した構造だが、メッセージ認証は平文ブロックのチェックサム、実際は排他的論理和をとり、これを暗号化するのみで実現しており、全体の計算量はほぼ暗号化のみの計算量と同等である。さらにブロックごとの並列処理が可能である。 基本的な構造は 2001 年に提案されており、マスク生成の違いなどで後に複数のバージョンが提案されている。AES を用いるケースでは、特に AESNI 命令が利用可能な CPU において顕著な高速性を有する。 なお、CAESAR final portfolio の Use Case 2 (High-performance Applications) に選出された。OCB には 3 種類のバリエーション（OCB1、OCB2、OCB3）があるが、CAESAR への提案方式は OCB3 である。
安全性解析状況	2021 年 9 月現在、いくつかの解析論文 [16, 46, 47, 48, 106] が発表されている。 提案論文 [58, 82, 83, 84] にて、OCB の安全性がブロック暗号の強擬似ランダム性（Strong Pseudorandomness）へ帰着可能なことが示されている。n ビットブロック暗号の利用において $n/2$ ビットの証明可能安全性を有する。 2020 年に Inoue ら [47, 48] は、OCB2 の基礎となる tweakable ブロック暗号 XEX*に欠陥があることを示すとともに、既存の安全性証明にも欠陥があることを指摘した。これらの欠陥を悪用することにより、現実的な攻撃として universal forgeries と full plaintext recovery が可能となる。結果として、OCB2 が ISO/IEC 19772:2009-02 規格から除外された [1]。なお、本攻撃は OCB1 と OCB3 には影響がない。 2023 年に Liénardy と Lafitte [69] は、OCB3 の nonce 長が 6 ビット未満の場合に現実的な攻撃が可能であることを報告している。文献 [69] では OCB3 の仕様変更が提案されているが、仕様が変更されない場合でも 6 ビット以上の nonce を使用すれば安全であると主張されている。このような短い nonce の実用性に議論はあるものの、仕様において最低 nonce 長の記載がない以上はリスクがあることに注意が必要である。
主な実装評価結果	(SW) AES-128 利用、 Intel Core i5-6600 (Skylake 3.31 GHz) で 0.64 C/B。 (HW) Virtex 6 で 1,348 slices、fmax 292.7 MHz。 その他多様な CPU での実装結果が報告されている [58]。
標準化状況	IETF RFC 7253 [59]

技術分野	認証暗号
名称	PRIMATEs
設計者	Elena Andreeva[1], Begul Bilgin[1], Andrey Bogdanov[2], Atul Luykx[1], Florian Mendel[3], Bart Mennink[1], Nicky Mouha[1], Qingju Wang[1], Kan Yasuda[4] (1: KU Leuven/Belgium, 2: Technical University of Denmark/Denmark, 3: Graz University of Technology/Austria, 4: NTT/Japan)
発表年	2014 (DIAC 2014 [3]、FSE 2014 [4])
仕様参照先	CAESAR ウェブサイト [10]、FSE 2014 [4]
特徴	Sponge 構造をもつ。具体的には、それぞれ異なる利用モードを持つ HANUMAN、GIBBON、APE の 3 つの方式で構成される。いずれの方式も 200 ないし 280 ビットの置換を内部要素とし、この置換を公開ランダム置換と仮定した場合に 80 ビットないし 120 ビットセキュリティを持つことが保証されている。内部の置換は AES ないし Rijndael と類似した SPN だが S-box は 5 ビットである。HANUMAN、GIBBON はそれぞれ既知の利用モード（SpongeWrap、MonkeyWrap）をベースとするが、APE は nonce 重複など考慮した独自の利用モードである。
安全性解析状況	HANUMAN に対して、Associated Data がないときの処理の問題点を利用した現実的な偽造作成攻撃が報告されている [100]。提案者による修正が提案されている。 その他、2021 年 9 月現在において目立った解析論文は発表されていない。
主な実装評価結果	GIBBON について、 (SW) Intel Core i5-6600 (Skylake 3.31 GHz) で 1,712 C/B。 (HW) Virtex 6 で 419 slices、fmax 333.4 MHz。

<table>
<tr><td>技術分野</td><td>認証暗号</td></tr>
<tr><td>名称</td><td>Grain-128A</td></tr>
<tr><td>設計者</td><td>Martin Agren[1], Martin Hell[1], Thomas Johansson[1], Willi Meier[2]
(1: Lund University/Sweden, 2: FHNW/Switzerland)</td></tr>
<tr><td>発表年</td><td>2011（IJWMC 2011 [2]）</td></tr>
<tr><td>仕様参照先</td><td>IJWMC 2011 [2]</td></tr>
<tr><td>特徴</td><td>Grain-128A は、eSTREAM portfolio に選出された Grain v1、その派生版である Grain-128 と同様の構造を有するハードウェア実装向けのストリーム暗号であるが、認証機能をサポートしていることが大きな違いである。初期化フェーズは 256 段、鍵長は 128 ビット、nonce 長は 96 ビットであり、タグ長は任意に設定できるものの 32 ビットが推奨されている。Grain-128A は、Grain-128 に対する既存攻撃に耐性を持つよう非線形関数に改良が施されている。
なお、NIST LWC プロジェクトのファイナリストの 1 つである Grain-128AEAD も同様の構造を有しており、Grain-128A に対する既存攻撃に耐性を持つようさらに改良が施されている。</td></tr>
<tr><td>安全性解析状況</td><td>2021 年 9 月現在、様々な解析論文 [40, 61, 97, 98, 99, 101] が発表されている。
単一鍵設定における最良の攻撃は、2018 年に提案された Todo ら [99] による高速相関攻撃であり、Grain-128A に対して仕様段数であっても秘密鍵の全数探索より効率的に内部状態復元攻撃が実行できる。なお、本攻撃では $2^{115.4}$ の計算量と $2^{113.8}$ のデータ量が必要となる。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果（Cadence RTL Compiler, TSMC 90 nm ASIC）[75]
<table>
<tr><th># of parallel</th><th>Frequency [GHz]</th><th>Throughput [Gbps]</th><th>Area [μm^2]</th><th>Power [μW]</th></tr>
<tr><td>1</td><td>2.1</td><td>1.1</td><td>5,876</td><td>96.9</td></tr>
<tr><td>2</td><td>2.0</td><td>2.0</td><td>6.972</td><td>106.1</td></tr>
<tr><td>4</td><td>2.0</td><td>4.0</td><td>8,299</td><td>120.6</td></tr>
<tr><td>8</td><td>1.9</td><td>7.6</td><td>10,778</td><td>176.4</td></tr>
<tr><td>16</td><td>1.7</td><td>13.6</td><td>15,709</td><td>247.8</td></tr>
<tr><td>32</td><td>1.5</td><td>24.0</td><td>23,430</td><td>417.9</td></tr>
</table>
なお、全てオリジナル実装の結果である。</td></tr>
<tr><td>標準化状況</td><td>ISO/IEC 29167-13 [50]、ISO/IEC 29192-8 [49]</td></tr>
</table>

参考文献

[1] ISO/IEC JTC 1/SC 27 STATEMENT ON OCB2.0 – Major weakness found in a standardised cipher scheme (2019-01-09, press release), https://www.din.de/resource/blob/321470/da3d9bce7116deb510f6aded2ed0b4df/20190107-press-release-19772-2009-1st-ed-ocb2-0-data.pdf（2023-10-04 閲覧）

[2] Ågren, M., Hell, M., Johansson, T., Meier, W.: Grain-128a: a new version of Grain-128 with optional authentication. Int. J. Wirel. Mob. Comput. 5(1), 48–59 (2011), https://doi.org/10.1504/IJWMC.2011.044106

[3] Andreeva, E., Bilgin, B., Bogdanov, A., Luykx, A., Mendel, F., Mennink, B., Mouha, N., Wang, Q., Yasuda, K.: CAESAR candidates PRIMATES. DIAC - Directions in Authenticated Ciphers (2014), https://2014.diac.cr.yp.to/

[4] Andreeva, E., Bilgin, B., Bogdanov, A., Luykx, A., Mennink, B., Mouha, N., Yasuda, K.: APE: Authenticated Permutation-Based Encryption for Lightweight Cryptography. In: FSE. Lecture Notes in Computer Science, vol. 8540, pp. 168–186. Springer (2014)

[5] Banik, S., Bogdanov, A., Minematsu, K.: Low-area hardware implementations of CLOC, SILC and AES-OTR. In: HOST. pp. 71–74. IEEE Computer Society (2016)

[6] Bar-On, A., Dunkelman, O., Keller, N., Weizman, A.: DLCT: A New Tool for Differential-Linear Cryptanalysis. In: Ishai, Y., Rijmen, V. (eds.) Advances in Cryptology - EUROCRYPT 2019 - 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19-23, 2019, Proceedings, Part I. Lecture Notes in Computer Science, vol. 11476, pp. 313–342. Springer (2019), https://doi.org/10.1007/978-3-030-17653-2_11

[7] Bassham, L., Chang, D., Kang, J., Kelsey, J., McKay, K., Turan, M.S., Waller, N.: NIST Lightweight Cryptography Project, https://csrc.nist.gov/projects/lightweight-cryptography

[8] Beaulieu, R., Shors, D., Smith, J., Treatman-Clark, S., Weeks, B., Wingers, L.: The SIMON and SPECK Families of Lightweight Block Ciphers. Cryptology ePrint Archive, Report 2013/404 (2013), https://eprint.iacr.org/2013/404

[9] Bellare, M., Desai, A., Jokipii, E., Rogaway, P.: A Concrete Security Treatment of Symmetric Encryption. In: FOCS. pp. 394–403. IEEE Computer Society (1997)

[10] Bernstein, D.J.: CAESAR: Competition for Authenticated Encryption: Security, Applicability, and Robustness, https://competitions.cr.yp.to/caesar.html（2023-10-04 閲覧）

[11] Bernstein, D.J.: eBACS: ECRYPT Benchmarking of Cryptographic Systems, https://bench.cr.yp.to/results-caesar.html（2023-10-04 閲覧）

[12] Bertoni, G., Daemen, J., Peeters, M., Assche, G.V.: On the Indifferentiability of the Sponge Construction. In: Smart, N.P. (ed.) Advances in Cryptology - EUROCRYPT 2008, 27th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Istanbul, Turkey, April 13-17, 2008. Proceedings. Lecture Notes in Computer Science, vol. 4965, pp. 181–197. Springer (2008), https://doi.org/10.1007/978-3-540-78967-3_11

[13] Bertoni, G., Daemen, J., Peeters, M., Assche, G.V.: Duplexing the Sponge: Single-Pass Authenticated Encryption and Other Applications. In: Miri, A., Vaudenay, S. (eds.) Selected Areas in Cryptography - 18th

International Workshop, SAC 2011, Toronto, ON, Canada, August 11-12, 2011, Revised Selected Papers. Lecture Notes in Computer Science, vol. 7118, pp. 320–337. Springer (2011), https://doi.org/10.1007/978-3-642-28496-0_19

[14] Bertoni, G., Daemen, J., Peeters, M., Assche, G.V.: The Keccak reference (2011), https://keccak.noekeon.org/（2023-10-04 閲覧）

[15] Bertoni, G., Daemen, J., Peeters, M., Assche, G.V., Keer, R.V.: CAESAR candidates Ketje + Keyak. DIAC - Directions in Authenticated Ciphers (2014), https://2014.diac.cr.yp.to/

[16] Bhaumik, R., Nandi, M.: Improved Security for OCB3. In: Takagi, T., Peyrin, T. (eds.) Advances in Cryptology - ASIACRYPT 2017 - 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part II. Lecture Notes in Computer Science, vol. 10625, pp. 638–666. Springer (2017), https://doi.org/10.1007/978-3-319-70697-9_22

[17] Bi, W., Dong, X., Li, Z., Zong, R., Wang, X.: MILP-aided cube-attack-like cryptanalysis on Keccak Keyed modes. Des. Codes Cryptogr. 87(6), 1271–1296 (2019), https://doi.org/10.1007/s10623-018-0526-x

[18] Bogdanov, A., Knudsen, L.R., Leander, G., Paar, C., Poschmann, A., Robshaw, M.J.B., Seurin, Y., Vikkelsoe, C.: PRESENT: An Ultra-Lightweight Block Cipher. In: CHES. Lecture Notes in Computer Science, vol. 4727, pp. 450–466. Springer (2007)

[19] Bost, R., Sanders, O.: Trick or Tweak: On the (In)security of OTR's Tweaks. In: Cheon, J.H., Takagi, T. (eds.) Advances in Cryptology - ASIACRYPT 2016 - 22nd International Conference on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, December 4-8, 2016, Proceedings, Part I. Lecture Notes in Computer Science, vol. 10031, pp. 333–353 (2016), https://doi.org/10.1007/978-3-662-53887-6_12

[20] Canteaut, A., Lambooij, E., Neves, S., Rasoolzadeh, S., Sasaki, Y., Stevens, M.: Refined Probability of Differential Characteristics Including Dependency Between Multiple Rounds. IACR Trans. Symmetric Cryptol. 2017(2), 203–227 (2017), https://doi.org/10.13154/tosc.v2017.i2.203-227

[21] Cid, C., Huang, T., Peyrin, T., Sasaki, Y., Song, L.: A Security Analysis of Deoxys and its Internal Tweakable Block Ciphers. IACR Trans. Symmetric Cryptol. 2017(3), 73–107 (2017), https://doi.org/10.13154/tosc.v2017.i3.73-107

[22] Cryptographic Engineering Research Group at George Mason University: ATHENa: Automated Tools for Hardware EvaluatioN, https://cryptography.gmu.edu/athena/（2023-10-04 閲覧）

[23] CRYPTREC 軽量暗号ワーキンググループ: CRYPTREC 暗号技術ガイドライン（軽量暗号）（文書番号：CRYPTREC GL-2003-2016JP）（2017), https://www.cryptrec.go.jp/report/cryptrec-gl-2003-2016jp.pdf

[24] Daemen, J.: Permutation-based encryption, authentication and authenticated encryption. DIAC - Directions in Authenticated Ciphers (2012), https://hyperelliptic.org/DIAC/

[25] Daemen, J., Hoffert, S., Peeters, M., Assche, G.V., Keer, R.V., Mella, S.: Team Keccak: The Ketje authenticated encryption scheme, https://keccak.team/ketje.html（2023-10-04 閲覧）

[26] Daemen, J., Mennink, B., Assche, G.V.: Full-State Keyed Duplex with Built-In Multi-user Support. In: Takagi, T., Peyrin, T. (eds.) Advances in Cryptology - ASIACRYPT 2017 - 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part II. Lecture Notes in Computer Science, vol. 10625, pp. 606–637. Springer (2017), https://doi.org/10.1007/978-3-319-70697-9_21

[27] Ding, L., Wang, L., Gu, D., Jin, C., Guan, J.: Algebraic Degree Estimation of ACORN v3 Using Numeric Mapping. Secur. Commun. Networks 2019, 7429320:1–7429320:5 (2019), https://doi.org/10.1155/2019/7429320

[28] Dobraunig, C., Eichlseder, M., Mendel, F., Schläffer, M.: ASCON: Lightweight Authenticated Encryption & Hashing, https://ascon.iaik.tugraz.at/（2023-10-04 閲覧）

[29] Dobraunig, C., Eichlseder, M., Mendel, F., Schläffer, M.: CAESAR candidates Ascon. DIAC - Directions in Authenticated Ciphers (2014), https://2014.diac.cr.yp.to/

[30] Dobraunig, C., Eichlseder, M., Mendel, F., Schläffer, M.: Cryptanalysis of Ascon. In: CT-RSA. Lecture Notes in Computer Science, vol. 9048, pp. 371–387. Springer (2015)

[31] Dobraunig, C., Eichlseder, M., Mendel, F., Schläffer, M.: Ascon v1.2. Submission to the NIST Lightweight Cryptography project (2021)

[32] Dong, X., Li, Z., Wang, X., Qin, L.: Cube-like Attack on Round-Reduced Initialization of Ketje Sr. IACR Trans. Symmetric Cryptol. 2017(1), 259–280 (2017), https://doi.org/10.13154/tosc.v2017.i1.259-280

[33] Dong, X., Qin, L., Sun, S., Wang, X.: Key Guessing Strategies for Linear Key-Schedule Algorithms in Rectangle Attacks. IACR Cryptol. ePrint Arch. 2021, 856 (2021), https://eprint.iacr.org/2021/856

[34] Erlacher, J., Mendel, F., Eichlseder, M.: Bounds for the Security of Ascon against Differential and Linear Cryptanalysis. IACR Trans. Symmetric Cryptol. 2022(1), 64–87 (2022), https://doi.org/10.46586/tosc.v2022.i1.64-87

[35] Farahmand, F., Diehl, W., Abdulgadir, A., Kaps, J., Gaj, K.: Improved Lightweight Implementations of CAESAR Authenticated Ciphers. In: 26th IEEE Annual International Symposium on Field-Programmable Custom Computing Machines, FCCM 2018, Boulder, CO, USA, April 29 - May 1, 2018. pp. 29–36. IEEE Computer Society (2018), https://doi.org/10.1109/FCCM.2018.00014

[36] Fuhr, T., Naya-Plasencia, M., Rotella, Y.: State-Recovery Attacks on Modified Ketje Jr. IACR Trans. Symmetric Cryptol. 2018(1), 29–56 (2018), https://doi.org/10.13154/tosc.v2018.i1.29-56

[37] Gérault, D., Peyrin, T., Tan, Q.Q.: Exploring differential-based distinguishers and forgeries for ASCON. IACR Cryptol. ePrint Arch. 2021, 1103 (2021), https://eprint.iacr.org/2021/1103, accepted to *IACR Trans. Symmetric Cryptol., 2021(3)*

[38] Guo, J., Peyrin, T., Poschmann, A., Robshaw, M.J.B.: The LED Block Cipher. In: CHES. Lecture Notes in Computer Science, vol. 6917, pp. 326–341. Springer (2011)

[39] Guo, Z., Wu, W., Liu, R., Zhang, L.: Multi-key Analysis of Tweakable Even-Mansour with Applications to Minalpher and OPP. IACR Trans. Symmetric Cryptol. 2016(2), 288–306 (2016), https://doi.org/10.13154/tosc.v2016.i2.288-306

[40] Hao, Y., Isobe, T., Jiao, L., Li, C., Meier, W., Todo, Y., Wang, Q.: Improved Division Property Based Cube Attacks Exploiting Algebraic Properties of Superpoly. IEEE Trans. Computers 68(10), 1470–1486 (2019), https://doi.org/10.1109/TC.2019.2909871

[41] Hao, Y., Jiao, L., Li, C., Meier, W., Todo, Y., Wang, Q.: Links between Division Property and Other Cube Attack Variants. IACR Trans. Symmetric Cryptol. 2020(1), 363–395 (2020), https://doi.org/10.13154/tosc.v2020.i1.363-395

[42] Hao, Y., Leander, G., Meier, W., Todo, Y., Wang, Q.: Modeling for Three-Subset Division Property Without Unknown Subset - Improved Cube Attacks Against Trivium and Grain-128AEAD. In: Canteaut, A., Ishai, Y. (eds.) Advances in Cryptology - EUROCRYPT 2020 - 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12105, pp. 466–495. Springer (2020), https://doi.org/10.1007/978-3-030-45721-1_17

[43] Hao, Y., Leander, G., Meier, W., Todo, Y., Wang, Q.: Modeling for Three-Subset Division Property without Unknown Subset. J. Cryptol. 34(3), 22 (2021), https://doi.org/10.1007/s00145-021-09383-2

[44] Hira, R., Kitahara, T., Miyahara, D., Hara-Azumi, Y., Li, Y., Sakiyama, K.: Software Evaluation for Second Round Candidates in NIST Lightweight Cryptography. IACR Cryptol. ePrint Arch. p. 591 (2022), https://eprint.iacr.org/2022/591

[45] Hirch, S.E., Mella, S., Mehrdad, A., Daemen, J.: Improved Differential and Linear Trail Bounds for ASCON. IACR Trans. Symmetric Cryptol. 2022(4), 145–178 (2022), https://doi.org/10.46586/tosc.v2022.i4.145-178

[46] Hirose, S., Sasaki, Y., Yasuda, K.: Rate-One AE with Security Under RUP. In: Nguyen, P.Q., Zhou, J. (eds.) Information Security - 20th International Conference, ISC 2017, Ho Chi Minh City, Vietnam, November 22-24, 2017, Proceedings. Lecture Notes in Computer Science, vol. 10599, pp. 3–20. Springer (2017), https://doi.org/10.1007/978-3-319-69659-1_1

[47] Inoue, A., Iwata, T., Minematsu, K., Poettering, B.: Cryptanalysis of OCB2: Attacks on Authenticity and Confidentiality. In: Boldyreva, A., Micciancio, D. (eds.) Advances in Cryptology - CRYPTO 2019 - 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2019, Proceedings, Part I. Lecture Notes in Computer Science, vol. 11692, pp. 3–31. Springer (2019), https://doi.org/10.1007/978-3-030-26948-7_1

[48] Inoue, A., Iwata, T., Minematsu, K., Poettering, B.: Cryptanalysis of OCB2: Attacks on Authenticity and Confidentiality. J. Cryptol. 33(4), 1871–1913 (2020), https://doi.org/10.1007/s00145-020-09359-8

[49] ISO/IEC: Information security – Lightweight cryptography – Part 8: Authenticated encryption (ISO/IEC 29192-8:2022), https://www.iso.org/standard/80114.html

[50] ISO/IEC: Information technology – Automatic identification and data capture techniques – Part 13: Crypto suite Grain-128A security services for air interface communications (ISO/IEC 29167-13: 2015), https://www.iso.org/standard/60682.html

[51] Iwata, T., Minematsu, K., Guo, J., Morioka, S.: CLOC and SILC – Authenticated Encryption Schemes for Constrained Devices, hhttps://www.nuee.nagoya-u.ac.jp/labs/tiwata/AE/

[52] Iwata, T., Minematsu, K., Guo, J., Morioka, S.: CLOC: Authenticated Encryption for Short Input. In: FSE. Lecture Notes in Computer Science, vol. 8540, pp. 149–167. Springer (2014)

[53] Iwata, T., Minematsu, K., Guo, J., Morioka, S., Kobayashi, E.: CAESAR candidates SILC. DIAC - Directions in Authenticated Ciphers (2014), https://2014.diac.cr.yp.to/

[54] Jean, J., Nikolić, I., Peyrin, T.: CAESAR candidates DEOXYS + Joltik. DIAC - Directions in Authenticated Ciphers (2014), https://2014.diac.cr.yp.to/

[55] Jean, J., Nikolic, I., Peyrin, T.: Tweaks and Keys for Block Ciphers: The TWEAKEY Framework. In: ASIACRYPT (2). Lecture Notes in Computer Science, vol. 8874, pp. 274–288. Springer (2014)

[56] Jean, J., Nikolić, I., Peyrin, T.: Deoxys and Joltik. DIAC - Directions in Authenticated Ciphers (2015)

[57] Kesarwani, A., Roy, D., Sarkar, S., Meier, W.: New cube distinguishers on NFSR-based stream ciphers. Des. Codes Cryptogr. 88(1), 173–199 (2020), https://doi.org/10.1007/s10623-019-00674-1

[58] Krovetz, T., Rogaway, P.: The Software Performance of Authenticated-Encryption Modes. In: FSE. Lecture Notes in Computer Science, vol. 6733, pp. 306–327. Springer (2011)

[59] Krovetz, T., Rogaway, P.: The OCB Authenticated-Encryption Algorithm. RFC 7253 (May 2014), https://www.rfc-editor.org/info/rfc7253

[60] Lafitte, F., Lerman, L., Markowitch, O., Heule, D.V.: SAT-based cryptanalysis of ACORN. IACR Cryptol. ePrint Arch. 2016, 521 (2016), https://eprint.iacr.org/2016/521

[61] Lehmann, M., Meier, W.: Conditional Differential Cryptanalysis of Grain-128a. In: Pieprzyk, J., Sadeghi, A., Manulis, M. (eds.) Cryptology and Network Security, 11th International Conference, CANS 2012, Darmstadt, Germany, December 12-14, 2012. Proceedings. vol. 7712, pp. 1–11. Springer (2012), https://doi.org/10.1007/978-3-642-35404-5_1

[62] Li, M., Chen, S.: Improved meet-in-the-middle attacks on reduced-round Joltik-BC. IET Information Security 15(3), 247–255 (2021)

[63] Li, M., Chen, S.: Improved Meet-in-the-Middle Attacks on Reduced-Round Tweakable Block Cipher Deoxys-BC. The Computer Journal (06 2021), https://doi.org/10.1093/comjnl/bxab076

[64] Li, R., Jin, C.: Meet-in-the-middle attacks on round-reduced tweakable block cipher Deoxys-BC. IET Inf. Secur. 13(1), 70–75 (2019), https://doi.org/10.1049/iet-ifs.2018.5091

[65] Li, R., Jin, C., Pan, H.: Key recovery attacks on reduced-round Joltik-BC in the single-key setting. Inf. Process. Lett. 151 (2019), https://doi.org/10.1016/j.ipl.2019.105834

[66] Li, Y., Zhang, G., Wang, W., Wang, M.: Cryptanalysis of round-reduced ASCON. Sci. China Inf. Sci. 60(3), 38102 (2017), https://doi.org/10.1007/s11432-016-0283-3

[67] Li, Z., Dong, X., Bi, W., Jia, K., Wang, X., Meier, W.: New Conditional Cube Attack on Keccak Keyed Modes. IACR Trans. Symmetric Cryptol. 2019(2), 94–124 (2019), https://doi.org/10.13154/tosc.v2019.i2.94-124

[68] Li, Z., Dong, X., Wang, X.: Conditional Cube Attack on Round-Reduced ASCON. IACR Trans. Symmetric Cryptol. 2017(1), 175–202 (2017), https://doi.org/10.13154/tosc.v2017.i1.175-202

[69] Liénardy, J., Lafitte, F.: A weakness in OCB3 used with short nonces allowing for a break of authenticity and confidentiality. Inf. Process. Lett. 183, 106404 (2024), https://doi.org/10.1016/j.ipl.2023.106404

[70] Liskov, M., Rivest, R.L., Wagner, D.: Tweakable Block Ciphers. In: Yung, M. (ed.) CRYPTO. Lecture Notes in Computer Science, vol. 2442, pp. 31–46. Springer (2002)

[71] Liu, M., Lu, X., Lin, D.: Differential-Linear Cryptanalysis from an Algebraic Perspective. In: Malkin, T., Peikert, C. (eds.) Advances in Cryptology - CRYPTO 2021 - 41st Annual International Cryptology Conference, CRYPTO 2021, Virtual Event, August 16-20, 2021, Proceedings, Part III. Lecture Notes in Computer Science, vol. 12827, pp. 247–277. Springer (2021), https://doi.org/10.1007/978-3-030-84252-9_9

[72] Liu, Y., Shi, B., Gu, D., Zhao, F., Li, W., Liu, Z.: Improved Meet-in-the-Middle Attacks on Reduced-Round Deoxys-BC-256. Comput. J. 63(12), 1859–1870 (2020), https://doi.org/10.1093/comjnl/bxaa028

[73] Liu, Y., Shi, Y., Gu, D., Zeng, Z., Zhao, F., Li, W., Liu, Z., Bao, Y.: Improved Meet-in-the-Middle Attacks on Reduced-Round Kiasu-BC and Joltik-BC. Comput. J. 62(12), 1761–1776 (2019), https://doi.org/10.1093/comjnl/bxz059

[74] Makarim, R.H., Rohit, R.: Towards Tight Differential Bounds of Ascon A Hybrid Usage of SMT and MILP. IACR Trans. Symmetric Cryptol. 2022(3), 303–340 (2022), https://doi.org/10.46586/tosc.v2022.i3.303-340

[75] Mansouri, S.S., Dubrova, E.: An Improved Hardware Implementation of the Grain-128a Stream Cipher. In: Kwon, T., Lee, M., Kwon, D. (eds.) Information Security and Cryptology - ICISC 2012 - 15th International Conference, Seoul, Korea, November 28-30, 2012, Revised Selected Papers. Lecture Notes in Computer Science, vol. 7839, pp. 278–292. Springer (2012), https://doi.org/10.1007/978-3-642-37682-5_20

[76] Minematsu, K.: Parallelizable Rate-1 Authenticated Encryption from Pseudorandom Functions. In: EUROCRYPT. Lecture Notes in Computer Science, vol. 8441, pp. 275–292. Springer (2014)

[77] Minematsu, K., Shigeri, M., Kubo, H.: AES-OTR v2. DIAC - Directions in Authenticated Ciphers (2015)

[78] Moazami, F., Mehrdad, A., Soleimany, H.: Impossible Differential Cryptanalysis on Deoxys-BC-256. ISC Int. J. Inf. Secur. 10(2), 93–105 (2018), https://doi.org/10.22042/isecure.2018.114245.405

[79] Peyrin, T., Seurin, Y.: Counter-in-Tweak: Authenticated Encryption Modes for Tweakable Block Ciphers. In: Robshaw, M., Katz, J. (eds.) Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part I. Lecture Notes in Computer Science, vol. 9814, pp. 33–63. Springer (2016)

[80] Peyrin, T., Sim, S.M., Wang, L., Zhang, G.: Cryptanalysis of JAMBU. In: FSE. Lecture Notes in Computer Science, vol. 9054, pp. 264–281. Springer (2015)

[81] Rezvani, B., Diehl, W.: Hardware Implementations of NIST Lightweight Cryptographic Candidates: A First Look. IACR Cryptol. ePrint Arch. p. 824 (2019), https://eprint.iacr.org/2019/824

[82] Rogaway, P.: Efficient Instantiations of Tweakable Blockciphers and Refinements to Modes OCB and PMAC. In: ASIACRYPT. Lecture Notes in Computer Science, vol. 3329, pp. 16–31. Springer (2004)

[83] Rogaway, P., Bellare, M., Black, J.: OCB: A block-cipher mode of operation for efficient authenticated encryption. ACM Trans. Inf. Syst. Secur. 6(3), 365–403 (2003)

[84] Rogaway, P., Bellare, M., Black, J., Krovetz, T.: OCB: a block-cipher mode of operation for efficient authenti-

cated encryption. In: ACM Conference on Computer and Communications Security. pp. 196–205. ACM (2001)

[85] Rogaway, P., Shrimpton, T.: A Provable-Security Treatment of the Key-Wrap Problem. In: Vaudenay, S. (ed.) EUROCRYPT. Lecture Notes in Computer Science, vol. 4004, pp. 373–390. Springer (2006)

[86] Rohit, R., Hu, K., Sarkar, S., Sun, S.: Misuse-Free Key-Recovery and Distinguishing Attacks on 7-Round Ascon. IACR Trans. Symmetric Cryptol. 2021(1), 130–155 (2021), https://doi.org/10.46586/tosc.v2021.i1.130-155

[87] Rohit, R., Sarkar, S.: Diving Deep into the Weak Keys of Round Reduced Ascon. IACR Trans. Symmetric Cryptol. 2021(4), 74–99 (2021), https://doi.org/10.46586/tosc.v2021.i4.74-99

[88] Roy, D., Mukhopadhyay, S.: Some results on ACORN. IACR Cryptol. ePrint Arch. 2016, 1132 (2016), https://eprint.iacr.org/2016/1132

[89] Salam, M.I., Bartlett, H., Dawson, E., Pieprzyk, J., Simpson, L., Wong, K.K.: Investigating Cube Attacks on the Authenticated Encryption Stream Cipher ACORN. In: Batten, L., Li, G. (eds.) Applications and Techniques in Information Security - 6th International Conference, ATIS 2016, Cairns, QLD, Australia, October 26-28, 2016, Proceedings. Communications in Computer and Information Science, vol. 651, pp. 15–26 (2016), https://doi.org/10.1007/978-981-10-2741-3_2

[90] Sasaki, Y.: Improved Related-Tweakey Boomerang Attacks on Deoxys-BC. In: Joux, A., Nitaj, A., Rachidi, T. (eds.) Progress in Cryptology - AFRICACRYPT 2018 - 10th International Conference on Cryptology in Africa, Marrakesh, Morocco, May 7-9, 2018, Proceedings. Lecture Notes in Computer Science, vol. 10831, pp. 87–106. Springer (2018), https://doi.org/10.1007/978-3-319-89339-6_6

[91] Sasaki, Y., Todo, Y.: New Impossible Differential Search Tool from Design and Cryptanalysis Aspects - Revealing Structural Properties of Several Ciphers. In: Coron, J., Nielsen, J.B. (eds.) Advances in Cryptology - EUROCRYPT 2017 - 36th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Paris, France, April 30 - May 4, 2017, Proceedings, Part III. Lecture Notes in Computer Science, vol. 10212, pp. 185–215 (2017), https://doi.org/10.1007/978-3-319-56617-7_7

[92] Sasaki, Y., Todo, Y., Aoki, K., Naito, Y., Sugawara, T., Murakami, Y., Matsui, M., Hirose, S.: Minalpher v1.1, https://competitions.cr.yp.to/caesar-submissions.html

[93] Sasaki, Y., Todo, Y., Aoki, K., Naito, Y., Sugawara, T., Murakami, Y., Matsui, M., Hirose, S.: CAESAR candidates Minalpher. DIAC - Directions in Authenticated Ciphers (2014), https://2014.diac.cr.yp.to/

[94] Song, L., Guo, J.: Cube-Attack-Like Cryptanalysis of Round-Reduced Keccak Using MILP. IACR Trans. Symmetric Cryptol. 2018(3), 182–214 (2018), https://doi.org/10.13154/tosc.v2018.i3.182-214

[95] Song, L., Guo, J., Shi, D., Ling, S.: New MILP Modeling: Improved Conditional Cube Attacks on Keccak-Based Constructions. In: Peyrin, T., Galbraith, S.D. (eds.) Advances in Cryptology - ASIACRYPT 2018 - 24th International Conference on the Theory and Application of Cryptology and Information Security, Brisbane, QLD, Australia, December 2-6, 2018, Proceedings, Part II. Lecture Notes in Computer Science, vol. 11273, pp. 65–95. Springer (2018), https://doi.org/10.1007/978-3-030-03329-3_3

[96] Suzaki, T., Minematsu, K., Morioka, S., Kobayashi, E.: TWINE : A Lightweight Block Cipher for Multiple Platforms. In: Selected Areas in Cryptography. Lecture Notes in Computer Science, vol. 7707, pp. 339–354. Springer (2012)

[97] Todo, Y., Isobe, T., Hao, Y., Meier, W.: Cube Attacks on Non-Blackbox Polynomials Based on Division Property. In: Katz, J., Shacham, H. (eds.) Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part III. Lecture Notes in Computer Science, vol. 10403, pp. 250–279. Springer (2017), https://doi.org/10.1007/978-3-319-63697-9_9

[98] Todo, Y., Isobe, T., Hao, Y., Meier, W.: Cube Attacks on Non-Blackbox Polynomials Based on Division Property. IEEE Trans. Computers 67(12), 1720–1736 (2018), https://doi.org/10.1109/TC.2018.2835480

[99] Todo, Y., Isobe, T., Meier, W., Aoki, K., Zhang, B.: Fast Correlation Attack Revisited - Cryptanalysis on

Full Grain-128a, Grain-128, and Grain-v1. In: Shacham, H., Boldyreva, A. (eds.) Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part II. Lecture Notes in Computer Science, vol. 10992, pp. 129–159. Springer (2018), https://doi.org/10.1007/978-3-319-96881-0_5

[100] Vizár, D.: Ciphertext Forgery on HANUMAN. Cryptology ePrint Archive, Report 2016/697 (2016), https://eprint.iacr.org/2016/697

[101] Wang, Q., Hao, Y., Todo, Y., Li, C., Isobe, T., Meier, W.: Improved Division Property Based Cube Attacks Exploiting Algebraic Properties of Superpoly. In: Shacham, H., Boldyreva, A. (eds.) Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part I. Lecture Notes in Computer Science, vol. 10991, pp. 275–305. Springer (2018), https://doi.org/10.1007/978-3-319-96884-1_10

[102] Wu, H.: CAESAR candidates Acorn + MORUS. DIAC - Directions in Authenticated Ciphers (2014), https://2014.diac.cr.yp.to/

[103] Wu, H., Huang, T.: CAESAR candidates AEGIS + Jambu. DIAC - Directions in Authenticated Ciphers (2014), https://2014.diac.cr.yp.to/

[104] Yalla, P., Kaps, J.: Evaluation of the CAESAR hardware API for lightweight implementations. In: International Conference on ReConFigurable Computing and FPGAs, ReConFig 2017, Cancun, Mexico, December 4-6, 2017. pp. 1–6. IEEE (2017), https://doi.org/10.1109/RECONFIG.2017.8279790

[105] Yang, J., Liu, M., Lin, D.: Cube Cryptanalysis of Round-Reduced ACORN. In: Lin, Z., Papamanthou, C., Polychronakis, M. (eds.) Information Security - 22nd International Conference, ISC 2019, New York City, NY, USA, September 16-18, 2019, Proceedings. Lecture Notes in Computer Science, vol. 11723, pp. 44–64. Springer (2019), https://doi.org/10.1007/978-3-030-30215-3_3

[106] Zhang, P., Wang, P., Hu, H., Cheng, C., Kuai, W.: INT-RUP Security of Checksum-Based Authenticated Encryption. In: Okamoto, T., Yu, Y., Au, M.H., Li, Y. (eds.) Provable Security - 11th International Conference, ProvSec 2017, Xi'an, China, October 23-25, 2017, Proceedings. Lecture Notes in Computer Science, vol. 10592, pp. 147–166. Springer (2017), https://doi.org/10.1007/978-3-319-68637-0_9

[107] Zhang, X., Lin, D.: Cryptanalysis of Acorn in Nonce-Reuse Setting. In: Chen, X., Lin, D., Yung, M. (eds.) Information Security and Cryptology - 13th International Conference, Inscrypt 2017, Xi'an, China, November 3-5, 2017, Revised Selected Papers. Lecture Notes in Computer Science, vol. 10726, pp. 342–361. Springer (2017), https://doi.org/10.1007/978-3-319-75160-3_21

[108] Zhao, B., Dong, X., Jia, K.: New Related-Tweakey Boomerang and Rectangle Attacks on Deoxys-BC Including BDT Effect. IACR Trans. Symmetric Cryptol. 2019(3), 121–151 (2019), https://doi.org/10.13154/tosc.v2019.i3.121-151

[109] Zhao, B., Dong, X., Jia, K., Meier, W.: Improved Related-Tweakey Rectangle Attacks on Reduced-Round Deoxys-BC-384 and Deoxys-I-256-128. In: Hao, F., Ruj, S., Gupta, S.S. (eds.) Progress in Cryptology - INDOCRYPT 2019 - 20th International Conference on Cryptology in India, Hyderabad, India, December 15-18, 2019, Proceedings. Lecture Notes in Computer Science, vol. 11898, pp. 139–159. Springer (2019), https://doi.org/10.1007/978-3-030-35423-7_7

[110] Zhao, Z., Chen, S., Wang, M., Wang, W.: Improved cube-attack-like cryptanalysis of reduced-round Ketje-Jr and Keccak-MAC. Inf. Process. Lett. 171, 106124 (2021), https://doi.org/10.1016/j.ipl.2021.106124

[111] Zong, R., Dong, X.: MILP-Aided Related-Tweak/Key Impossible Differential Attack and its Applications to QARMA, Joltik-BC. IEEE Access 7, 153683–153693 (2019), https://doi.org/10.1109/ACCESS.2019.2946638

[112] Zong, R., Dong, X., Wang, X.: Collision Attacks on Round-Reduced Gimli-Hash/Ascon-Xof/Ascon-Hash. IACR Cryptol. ePrint Arch. 2019, 1115 (2019), https://eprint.iacr.org/2019/1115

[113] Zong, R., Dong, X., Wang, X.: Related-tweakey impossible differential attack on reduced-round Deoxys-BC-256.

Sci. China Inf. Sci. 62(3), 32102:1–32102:12 (2019), https://doi.org/10.1007/s11432-017-9382-2
[114] 伊藤竜馬: 「CRYPTREC 暗号技術ガイドライン（軽量暗号）」掲載の暗号方式に関する安全性評価の動向調査（文書番号：CRYPTREC EX-3101-2021）(2021), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3101-2021.pdf
[115] 崎山一男: 軽量暗号の実装性能に関する調査及び評価（NIST 軽量暗号コンペティションファイナリスト）（文書番号：CRYPTREC EX-3205-2022）(2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3205-2022.pdf
[116] 藤堂洋介: 軽量暗号の安全性に関する調査及び評価（Ascon,Grain-128AEAD,TinyJambu）（文書番号：CRYPTREC EX-3203-2022）(2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3203-2022.pdf

付録 A

Ascon の物理攻撃耐性

A.1 サイドチャネル攻撃対策手法

Ascon のサイドチャネル攻撃対策として有効な Threshold Implementation (TI) と Domain Oriented Masking (DOM) について概説する。

A.1.1 Threshold Implementation (TI)

TI は秘密分散法に基づくマスキング手法であり、2006 年に Nikova らによって提案された [11, 12]。TI において、計算対象の値 x は (x_1, x_2, x_3) のようにシェアと呼ばれる複数の値で表現される。

ここで、GF(2^m) 上の非線形変換 $z = \mathrm{N}(x, y)$ を考える。この際、GF(2^m) 上の非線形変換 $z = \mathrm{N}(x, y)$ に対しても、シェアの考え方を適用することができる。例えば、3 つの関数 f_1, f_2, f_3 が、以下に示す Non-Completeness（不完全性）、Correctness（正確性）、そして Uniformity（均一性）の性質を有する場合、z を 3 つのシェアに分け、2 次のプロービングモデル[*1]に耐性のある計算処理を実現できることが知られている。

Non-Completeness　各関数 f_1, f_2, f_3 は、入力変数 x, y の少なくとも 1 つのシェア値に依存しないよう、例えば、次のように計算する。

$$
\begin{aligned}
z_1 &= f_1(x_2, x_3, y_2, y_3), \\
z_2 &= f_2(x_3, x_1, y_3, y_1), \\
z_3 &= f_3(x_1, x_2, y_1, y_2).
\end{aligned}
$$

このように計算することで、2 つ以下のシェアに分けた関数の処理から、元の値 x, y の値に関する情報を知ることができない。

Correctness　各関数 f_1, f_2, f_3 は、以下の関係が満たされる場合に Correctness を満たす。

$$
\begin{aligned}
z &= z_1 \oplus z_2 \oplus z_3 \\
&= f_1(x_2, x_3, y_2, y_3) \oplus f_2(x_3, x_1, y_3, y_1) \oplus f_3(x_1, x_2, y_1, y_2) \\
&= \mathrm{N}(x, y).
\end{aligned}
$$

Uniformity　入力変数 x, y の発生確率は均一でなければならない。これは、発生確率に偏りが生じる場合、攻撃者はその偏りを利用することで、全てのシェア値を取得しなくとも正しい x, y の値を復元できる可能性があるからである。例えば、$m = 1$ の場合、つまり GF(2) の乗算において、入力変数 x の発生確率は以下を満たさなければならない。

$$
\Pr[x_1, x_2, x_3] = \frac{1}{8}.
$$

[*1] プロービングモデルとは、暗号化処理を行うハードウェアやソフトウェアに対し、攻撃者が本来観測することができない内部信号を 1 本あるいは複数のプローブ（針）を用いて観測可能とする攻撃者モデルである [8]。d 次プロービングモデルの場合、攻撃者は異なる d 本のプローブを用いて d 個の中間値を観測できると仮定する。ただし、同じ回路を使い回すシェア型のハードウェアアーキテクチャの場合、同じプローブで異なる時間の複数の中間値を取得することも想定できる [14]。

なお、関数が次数 t である場合、d 次のプロービングモデルに対してサイドチャネル攻撃耐性を持つためのシェアの数は、最小で $td+1$ であることが知られている。

A.1.2 Domain Oriented Masking (DOM)

DOM は d 次のプロービングモデルに対して耐性のあるマスキング手法であり、2016 年に Großらによって提案された [6, 7]。ドメインと呼ばれる概念を導入し、ドメインごとにシェアを構成することで、非線形演算によって増加するシェアの数を抑制することを可能にした。なお、d プロービングモデルへの耐性を実現するためには、変数ごとに $d+1$ 個のシェアを使用する必要があり、この場合におけるドメインの数は $d+1$ 個となる。

例えば、1 次プロービングモデルへの耐性を実現するために、変数 x, y のシェア $(x_0, y_0), (x_1, y_1)$ をそれぞれドメイン 0 とドメイン 1 に関連づける。また、GF(2^m) 上の 1 次安全な DOM 乗算器を考える。この時、TI と同様、入力値 x, y に関して以下の計算処理を実行する。

$$\begin{aligned} x \cdot y &= (x_0 \oplus x_1) \cdot (y_0 \oplus y_1) \\ &= x_0 \cdot y_0 \oplus x_0 \cdot y_1 \oplus x_1 \cdot y_0 \oplus x_1 \cdot y_1. \end{aligned}$$

ここで、‘·’ の演算記号は AND 演算を意味する。

$x_0 \cdot y_0$ の演算処理は、ドメイン 0 で安全に実行可能である。なぜならば、どの中間値（サイドチャネル情報）をプロービングによって読み出したとしても、入力値 x, y を復元することができないからである。同様に、$x_1 \cdot y_1$ の演算処理も、ドメイン 1 で安全に実行可能である。$x_0 \cdot y_1$ と $x_1 \cdot y_0$ の演算処理に関しても、入力値 x, y から独立しているため、これらの各処理からだけで入力値 x, y に関するサイドチャネルからの漏洩情報は観測できない。

一方、$x_0 \cdot y_1$ の演算処理をドメイン 0 に取り込み、$x_0 \cdot y_0 \oplus x_0 \cdot y_1$ を計算した場合には問題が生じる可能性がある。直接的ではないものの、異なるドメインのシェア y_0, y_1 が XOR で演算処理できるからである。これにより y の値が即座に復元できるわけではないものの、サイドチャネル情報に関する漏洩の危険性があると考えるべきである。そこで、$x_0 \cdot y_1$ と $x_1 \cdot y_0$ の演算処理は、いずれもクロスドメインで計算しなければならない処理とみなし、特定のドメイン 0 やドメイン 1 における演算処理とは切り離して考える必要がある。ここまでの手順が、DOM における計算（Calculation）ステップとなる。

次に、クロスドメインでの計算結果を特定のドメインに取り込むために、再シェア（Resharing）と呼ばれるステップを実行する。具体的には、$x_0 \cdot y_1$ と $x_1 \cdot y_0$ の演算処理後に、フレッシュな乱数 r でマスキングを行う。この再シェアでは、同じ乱数 r を使っても問題ないと知られている。つまり、$x_0 \cdot y_1 \oplus r$ と $x_1 \cdot y_0 \oplus r$ のようにマスキング処理を行うことができる。また、クロスドメインに関する一連の処理に起因して生じるグリッチの伝搬については、再シェアの結果をフリップフロップ回路に格納することで情報漏洩を抑止する。パイプライン処理では、ドメイン 0 やドメイン 0 における演算処理のタイミングを揃える必要があるため、$x_0 \cdot y_0$ と $y_0 \cdot y_1$ の計算結果もフリップフロップ回路に格納する。

最後に、統合（Integration）ステップでは、次のように特定のドメインとクロスドメインの演算結果の統合、つまり XOR 演算を行う。

$$\begin{aligned} q_0 &= (x_0 \cdot y_0) \oplus (x_0 \cdot y_1 \oplus r), \\ q_1 &= (x_1 \cdot y_1) \oplus (x_1 \cdot y_0 \oplus r). \end{aligned}$$

上記の 3 つのステップ（計算ステップ、再シェアステップ、統合ステップ）について、文献 [6] の Figure 1.1 又は文献 [7] の Figure 2 で概要図が示されているため、参考にされたい。

同様に、より高次（2 次以上）のプロービングモデルに耐性のある安全な DOM 乗算器も設計することが可能である。また、上述の 3 つのステップは、S-box などの非線形演算にも適用できる。

DOM の最大の特徴は、ドメイン単位の管理によってシェア数を適切に管理することができるとともに、再シェアステップにおいて使用する乱数を工夫することによってサイドチャネル攻撃対策にかかる実装コストの削減が期待できることである。TI による回路サイズの削減には数学的な処理の変換が必要になることが多いものの、DOM による設計手法は任意の回路に対して単純な処理ステップを繰り返すことで実現可能なため、設計の自動化が容易なマスキング手法であると言える。つまり、DOM によるマスキング実装の設計生産性は高い。なお、DOM によって生成された回路の実装コストは、最適化されていない TI 実装よりも低く、最適化された TI に匹敵する結果も得られている。

A.2 サイドチャネル解析・漏えい評価手法

Ascon に対するサイドチャネル解析・漏えい評価手法として報告されている相関電力解析（CPA: Correlation Power Analysis)、故障利用攻撃 (FA: Fault Attack)、Test Vector Leakage Assessment (TVLA) 、そしてテンプレート攻撃 (TA: Templete Attack）について概説する。

A.2.1 相関電力解析（CPA: Correlation Power Analysis）

相関電力解析は、電力のサイドチャネル情報を効率よく解析する方法として最もよく知られている [3]。なお、電磁波サイドチャネルに対する解析手法は、相関電磁波解析（CEMA: Correlation ElectroMagnetic Analysis）と呼ばれている。

差分電力解析（DPA: Differential Power Analysis） [9] では特定の 1 ビットに対する電力モデルが採用されるのに対し、相関電力解析では複数ビットの電力消費をモデル化するため、測定ノイズや処理アルゴリズムに起因するノイズの影響を軽減することが期待できる。また、差分電力解析では秘密鍵などの秘密情報の推測結果に基づいて電力波形データを 2 つのグループに分け、これら 2 つのデータの平均の差を解析するのに対し、相関電力解析では電力波形データをより多くのグループに分け、電力モデルとの相関関係を解析する。Ascon に限らず多くの暗号アルゴリズムにおいて、推測した秘密情報によってレジスタに格納される中間値の複数ビットを導出できる場合には、相関電力解析が最適である。

Ascon に対する既存の漏洩評価においても、相関電力解析を用いた安全性評価手法が採用されている。Ascon に対する相関電力解析では、認証暗号アルゴリズムの暗号化または復号処理において、攻撃者が秘密鍵を復元できるかどうかで評価されている。相関電力解析において中間値を導出するために選択関数（Selection Function）という概念を導入するが、この選択関数は初期化処理または最終処理（タグ生成処理）から構成される場合が多い。これは、初期化処理や最終処理に秘密鍵が直接関与しており、秘密鍵の予測によって中間値の予測が可能になるためである。選択関数の構成方法については、文献 [13] を参照されたい。

A.2.2 故障利用攻撃（FA: Fault Attack）

故障利用攻撃では、暗号機能を実装したハードウェアの動作中に故意に故障 (fault) を起こし、故障によって生じた計算誤りを利用して解析を行う手法である。故障利用解析の中でも差分故障解析 (DFA: Differential Fault Analysis) [2] が最もよく知られている解析手法の 1 つであり、正しい暗号文と誤りが生じた暗号文の差分を利用し、秘密鍵候補の探索空間を削減することで秘密鍵を推定する。

差分故障解析への有効な対策の 1 つとして、暗号化処理の二重化、つまり同じ暗号化処理を 2 回行い、その結果を比較することで誤った暗号文を出力しないといった対策が施される。しかし、Fault Sensitivity Analysis (FSA) [10] や Statistical Ineffective Fault Attack (SIFA) [4] といった高度な解析手法に対し、暗号化処理の二重化という単純な対策では不十分と言われている。

共通鍵暗号に対する最新の故障利用攻撃については、Baksi らによる SoK 論文 [1] を参照されたい。

A.2.3 Test Vector Leakage Assessment (TVLA)

ウェルチの t 検定（Welch's t-test）は様々な分野において幅広く利用されている統計的手法の 1 つであり、サイドチャネルからの漏洩評価における t 検定は、Test Vector Leakage Assessment (TVLA) と呼ばれている。TVLA の目的は、秘密鍵の復元や秘密情報の取得ではなく、暗号化処理デバイスの内部データとサイドチャネル情報との依存関係を評価し、潜在的な脆弱性を特定することにある。攻撃者の計算能力や攻撃手法に関係なく、暗号実装の安全性に関する汎用的な評価指標が提供できるツールとして、幅広く使用されるようになった。

具体的には、ある 2 つの基準に従って暗号アルゴリズムの処理を実行し、その際に測定したサイドチャネル情報の波形データをそれぞれデータセット A とデータセット B に分類する。これらのデータセットに含まれる各サンプル点に対し、

以下の式を用いて t 値を導出する。

$$t = \frac{\mu_A - \mu_B}{\sqrt{\frac{\sigma_A^2}{n_A} + \frac{\sigma_B^2}{n_B}}}.$$

ここで、μ、σ^2、n は、サンプル点における波形データ値の平均、分散、そして標本数であり、データセット A とデータセット B に対してそれぞれを求める。

サイドチャネルからの漏洩評価で一般的に使用されている評価基準は、秘密鍵を固定して、波形データのデータセットの 1 つを固定平文とし、もう 1 つをランダム平文とするものである [5]。$t < 4.5\sigma$ を満たす場合、そのサンプル点ではサイドチャネルからの漏洩がないものと判断される。

平易に表現すると、固定した平文の値に依存したサイドチャネルからの漏洩があるかを解析するものである。平文をランダムに入力した場合のサイドチャネル情報と比較してなんらかの差異が見られるならば、攻撃者はそのような情報を使用して内部の秘密情報を取得できる可能性があると判断する。つまり、t 検定においてサイドチャネルからの漏洩の可能性が示されたとしても、具体的な攻撃を実行できるかは不明であり、未知の攻撃を含めて安全性評価をより厳格に行う必要があると言える。

A.2.4 テンプレート攻撃（TA: Templete Attack）

テンプレート攻撃は、プロファイリングフェーズと攻撃フェーズから構成される。プロファイリングフェーズでは、攻撃対象と同種類のモジュールを使用し、入力値などのパラメータを操作しながら対象となるモジュールの特性を評価するフェーズである。攻撃フェーズでは、パラメータを操作できない攻撃対象モジュールに対して秘密鍵の推定を行うフェーズである。

上述のとおり、テンプレート攻撃の前提として、攻撃者は暗号アルゴリズムを処理するデバイスを完全に制御できなくてはならない。なぜならば、攻撃者が自由に平文や秘密鍵などの情報をデバイスに設定し、デバイスから漏洩したサイドチャネル情報の確率分布からデバイスの物理特性をプロファイリングしなければならないからである。つまり、テンプレート攻撃では、簡略化した電力モデルの代わりに、実際のデバイスから得られる物理的な振る舞いを用いて複雑なモデルを構築し、攻撃に利用する。攻撃者がデバイスから無制限に情報をプロファイリングすることができれば、測定ノイズを十分に削減することができるため、非常に強力な攻撃になりうる。

参考文献

[1] Anubhab Baksi, Shivam Bhasin, Jakub Breier, Dirmanto Jap, and Dhiman Saha. A Survey on Fault Attacks on Symmetric Key Cryptosystems. *ACM Comput. Surv.*, 55(4):86:1–86:34, 2023.

[2] Eli Biham and Adi Shamir. Differential Fault Analysis of Secret Key Cryptosystems. In Burton S. Kaliski Jr., editor, *Advances in Cryptology - CRYPTO '97, 17th Annual International Cryptology Conference, Santa Barbara, California, USA, August 17-21, 1997, Proceedings*, volume 1294 of *Lecture Notes in Computer Science*, pages 513–525. Springer, 1997.

[3] Eric Brier, Christophe Clavier, and Francis Olivier. Correlation Power Analysis with a Leakage Model. In Marc Joye and Jean-Jacques Quisquater, editors, *Cryptographic Hardware and Embedded Systems - CHES 2004: 6th International Workshop Cambridge, MA, USA, August 11-13, 2004. Proceedings*, volume 3156 of *Lecture Notes in Computer Science*, pages 16–29. Springer, 2004.

[4] Christoph Dobraunig, Maria Eichlseder, Thomas Korak, Stefan Mangard, Florian Mendel, and Robert Primas. SIFA: Exploiting Ineffective Fault Inductions on Symmetric Cryptography. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2018(3):547–572, 2018.

[5] Gilbert Goodwill, Benjamin Jun, Josh Jaffe, and Pankaj Rohatgi. A testing methodology for side channel resistance validation, 2011. NIST, Non-Invasive Attack Testing Workshop. `https://csrc.nist.gov/CSRC/media/Events/Non-Invasive-Attack-Testing-Workshop/documents/08_Goodwill.pdf`.

[6] Hannes Groß. Domain-Oriented Masking–Generically Masked Hardware Implementations, 2018. PhD Thesis, IAIK, Graz University of Technology. `https://diglib.tugraz.at/download.php?id=5c80ea0c43a56&location=browse`（2023-10-07 閲覧）.

[7] Hannes Groß, Stefan Mangard, and Thomas Korak. Domain-Oriented Masking: Compact Masked Hardware Implementations with Arbitrary Protection Order. *IACR Cryptol. ePrint Arch.*, page 486, 2016.

[8] Yuval Ishai, Amit Sahai, and David A. Wagner. Private Circuits: Securing Hardware against Probing Attacks. In Dan Boneh, editor, *Advances in Cryptology - CRYPTO 2003, 23rd Annual International Cryptology Conference, Santa Barbara, California, USA, August 17-21, 2003, Proceedings*, volume 2729 of *Lecture Notes in Computer Science*, pages 463–481. Springer, 2003.

[9] Paul C. Kocher, Joshua Jaffe, and Benjamin Jun. Differential Power Analysis. In Michael J. Wiener, editor, *Advances in Cryptology - CRYPTO '99, 19th Annual International Cryptology Conference, Santa Barbara, California, USA, August 15-19, 1999, Proceedings*, volume 1666 of *Lecture Notes in Computer Science*, pages 388–397. Springer, 1999.

[10] Yang Li, Kazuo Sakiyama, Shigeto Gomisawa, Toshinori Fukunaga, Junko Takahashi, and Kazuo Ohta. Fault Sensitivity Analysis. In Stefan Mangard and François-Xavier Standaert, editors, *Cryptographic Hardware and Embedded Systems, CHES 2010, 12th International Workshop, Santa Barbara, CA, USA, August 17-20, 2010. Proceedings*, volume 6225 of *Lecture Notes in Computer Science*, pages 320–334. Springer, 2010.

[11] Svetla Nikova, Christian Rechberger, and Vincent Rijmen. Threshold Implementations Against Side-Channel Attacks and Glitches. In Peng Ning, Sihan Qing, and Ninghui Li, editors, *Information and Communications Security, 8th International Conference, ICICS 2006, Raleigh, NC, USA, December 4-7, 2006, Proceedings*, volume 4307 of *Lecture Notes in Computer Science*, pages 529–545. Springer, 2006.

[12] Svetla Nikova, Vincent Rijmen, and Martin Schläffer. Secure hardware implementation of nonlinear functions in the presence of glitches. *J. Cryptol.*, 24(2):292–321, 2011.

[13] Niels Samwel and Joan Daemen. DPA on hardware implementations of Ascon and Keyak. In *Proceedings of the Computing Frontiers Conference, CF'17, Siena, Italy, May 15-17, 2017*, pages 415–424. ACM, 2017.

[14] Takeshi Sugawara, Yang Li, and Kazuo Sakiyama. Probing attack of share-serial threshold implementation of advanced encryption standard. *IET Electronics Letters*, 55(9):517–519, 2019.

付録 B

CAESAR final portfolio: AEGIS, COLM

4.5 節の冒頭で述べたとおり、2016 年度版ガイドライン [2] では CAESAR final portfolio に選出された 6 方式のうち AEGIS-128 と COLM の 2 方式について掲載していない。これら 2 方式については軽量性を謳った方式ではないものの、CAESAR final portfolio に選出された方式であるということを鑑み、本節でこれらの方式の調査結果をまとめる。なお、調査結果については、2021 年度に公開された CRYPTREC 外部評価報告書 [8] に基づいて記載した。

技術分野	認証暗号
名称	AEGIS
設計者	Hongjun Wu[1], Bart Preneel[2] (1: Nanyang Technological University/Singapore, 2: KU Leuven/Belgium)
発表年	2013（SAC 2013 [7]）
仕様参照先	CAESAR ウェブサイト [1]、SAC 2013 [7]
特徴	AEGIS は、AEGIS-128L、AEGIS-128、AEGIS-256 の 3 種のバリエーションが提案されており、AEGIS-128 が final portfolio の Use Case 2 (High-performance Applications) に選出された。AEGIS-128 は、640 (128×5) ビットの内部状態を持ち、5 つの AES ラウンド関数を並列に実行することで内部状態を更新する。鍵長、nonce 長、タグ長はそれぞれ 128 ビットを推奨している。 なお、ソフト・ハード両面での高速性が特徴として挙げられる。
安全性解析状況	2021 年 9 月現在、様々な解析論文 [4, 5, 6] が発表されているが、仕様段数において秘密鍵の全数探索よりも効率的に実行可能な攻撃は発表されていない。 単一鍵設定における最良の攻撃は、2019 年に提案された Eichlseder ら [4] による線形攻撃であり、AEGIS-256 に対して仕様段数であっても効率的に識別攻撃を実行できる。なお、AEGIS-128 に対しては仕様段数であっても 2^{132} から 2^{140} の範囲の計算量で識別攻撃を実行できる。弱鍵設定における最良の攻撃は、2021 年に提案された Liu ら [5] による積分攻撃であり、10 段のうち 5 段に簡略化した AEGIS-128 に対して、効率的に鍵回復攻撃と識別攻撃が実行できる。
主な実装評価結果	(SW) AEGIS-128、Intel Core i5-1030NG7 (Icelake 4×1.1 GHz) で 0.41 C/B。 (HW) AEGIS-128L、Virtex 6 で 1,025 slices、fmax 320.8 MHz。

技術分野	認証暗号
名称	COLM
設計者	Elena Andreeva[1], Andrey Bogdanov[2], Nilanjan Datta[3], Atul Luykx[1], Bart Mennink[1], Mridul Nandi[3], Elmar Tischhauser[2], Kan Yasuda[4] (1: KU Leuven/Belgium, 2: Technical University of Denmark/Denmark, 3: Indian Statistical Institute/India, 4: NTT/Japan)
発表年	2016（CAESAR ウェブサイト [1]）
仕様参照先	CAESAR ウェブサイト [1]
特徴	COLM は、COLM_0 と COLM_{127} の 2 種のバリエーションが提案されており、いずれも final portfolio の Use Case 3 (Defense in Depth) に選出された。当初、CAESAR submissions として AES-COPA と ELmD が投稿されたが、それぞれの長所を活かした形として COLM が設計された。 COLM は、ブロック暗号ベースの Encrypt-Linearmix-Encrypt 構造を採用しており、ブロック暗号として AES-128 を利用する。鍵長とタグ長は 128 ビット、nonce 長は 64 ビットが推奨されている。COLM_0 と COLM_{127} の主な違いはタグ生成の手順であり、COLM_{127} では暗号化処理の途中で中間タグ値を生成した後、これらの中間タグ値を用いてタグ生成が実行される。
安全性解析状況	2021 年 9 月現在、Datta ら [3] の他、目立った解析論文は発表されていない。 COLM タイプの認証暗号に対し、nonce-misuse シナリオと nonce-respecting シナリオにおける INT-RUP（タグ未検証において取得された平文の整合性）を考慮した攻撃について、2017 年に議論されている [3]。n をブロックサイズとすると、nonce-misuse シナリオにおいて暗号化・復号クエリが各 $4n$ 回、メッセージブロックサイズが $3n$ ブロックの場合に偽造攻撃が成立し、nonce-respecting シナリオにおいて暗号化クエリが 1 回、復号クエリが $2n$ 回、メッセージブロックサイズが $(n+1)n$ ブロックの場合に偽造攻撃が成立する。なお、これらの攻撃については COLM_{127} に影響しない。
主な実装評価結果	(SW) COLM_0、Intel Core i5-1030NG7 (Icelake 4 × 1.1 GHz) で 1.10 C/B。 (SW) COLM_{127}、Intel Core i5-1030NG7 (Icelake 4 × 1.1 GHz) で 30.06 C/B。 (HW) COLM_0、Virtex 6 で 2,060 slices、fmax 241.8 MHz。

参考文献

[1] Bernstein, D.J.: CAESAR: Competition for Authenticated Encryption: Security, Applicability, and Robustness, https://competitions.cr.yp.to/caesar.html（2023-10-04 閲覧）

[2] CRYPTREC 軽量暗号ワーキンググループ: CRYPTREC 暗号技術ガイドライン（軽量暗号）（文書番号：CRYPTREC GL-2003-2016JP）(2017), https://www.cryptrec.go.jp/report/cryptrec-gl-2003-2016jp.pdf

[3] Datta, N., Luykx, A., Mennink, B., Nandi, M.: Understanding RUP Integrity of COLM. IACR Trans. Symmetric Cryptol. 2017(2), 143–161 (2017), https://doi.org/10.13154/tosc.v2017.i2.143-161

[4] Eichlseder, M., Nageler, M., Primas, R.: Analyzing the Linear Keystream Biases in AEGIS. IACR Trans. Symmetric Cryptol. 2019(4), 348–368 (2019), https://doi.org/10.13154/tosc.v2019.i4.348-368

[5] Liu, F., Isobe, T., Meier, W., Sakamoto, K.: Weak Keys in Reduced AEGIS and Tiaoxin. IACR Trans. Symmetric Cryptol. 2021(2), 104–139 (2021), https://doi.org/10.46586/tosc.v2021.i2.104-139

[6] Minaud, B.: Linear Biases in AEGIS Keystream. In: Joux, A., Youssef, A.M. (eds.) Selected Areas in Cryptography - SAC 2014 - 21st International Conference, Montreal, QC, Canada, August 14-15, 2014, Revised Selected Papers. Lecture Notes in Computer Science, vol. 8781, pp. 290–305. Springer (2014), https://doi.org/10.1007/978-3-319-13051-4_18

[7] Wu, H., Preneel, B.: AEGIS: A Fast Authenticated Encryption Algorithm. In: Lange, T., Lauter, K.E., Lisonek, P. (eds.) Selected Areas in Cryptography - SAC 2013 - 20th International Conference, Burnaby, BC, Canada, August 14-16, 2013, Revised Selected Papers. Lecture Notes in Computer Science, vol. 8282, pp. 185–201. Springer (2013), https://doi.org/10.1007/978-3-662-43414-7_10

[8] 伊藤竜馬:「CRYPTREC 暗号技術ガイドライン（軽量暗号）」掲載の暗号方式に関する安全性評価の動向調査（文書番号：CRYPTREC EX-3101-2021）（2021), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3101-2021.pdf

付録 C

NIST LWC ファイナリスト（Ascon を除く）

4.5 節の冒頭で述べたとおり、Ascon を除く NIST 軽量暗号（NIST LWC）プロジェクトのファイナリスト 9 方式（Elephant、GIFT-COFB、Grain-128AEAD、ISAP、PHOTON-Beetle、Romulus、Sparkle、TinyJAMBU、Xoodyak）についても軽量性の観点で優れており、かつ安全性の観点で問題が見つかっていない方式であることから、本節でこれらの方式の調査結果をまとめる。

各方式の仕様（設計者、発表年、仕様参照先、特徴）、安全性解析状況、そして主な実装評価結果については、2022 年度に公開された CRYPTREC 外部評価報告書 [149, 151, 152, 155, 156] に基づき、2022 年 9 月現在の調査結果を記載した。なお、文献 [152] で多くの実装評価結果がまとめられているものの、紙面の都合上、次の項目に限定している。ハードウェア実装評価結果については、FPGA 実装に着目し、回路面積の観点からコンパクト実装である結果、またはスループットの観点で高速実装である結果を抽出している。回路面積の評価尺度は、ルックアップテーブル数（LUTs）である。一部、紙面に余裕がある場合には、GlobalFoundries 社の GF 22nm CMOS で合成した ASIC 実装の評価結果も掲載している。評価尺度は FPGA 実装の場合と同じである。ソフトウェア実装評価結果については、IoT 向けローエンド CPU、特に Arm Cortex-M0 上での実装に着目し、設計者が作成したリファレンスコードを使用した場合のレイテンシ（暗号化・復号）、ROM サイズ、コードサイズの結果をまとめている。レイテンシの評価尺度は、テストベクトルを実行した際の 1 回の処理にかかる実行時間（msec）の平均値である。その他、文献 [152] では、ASIC 実装、命令拡張のハードウェア実装、ハイエンド CPU 上でのソフトウェア実装の結果がまとめられている。

技術分野	認証暗号
名称	Elephant
設計者	Tim Beyne[1], Yu Long Chen[1], Christoph Dobraunig[2], Bart Mennink[2] (1: KU Leuven/Belgium, 2: Radboud University/Netherlands)
発表年	2019（NIST LWC ウェブサイト [13]）
仕様参照先	NIST LWC ウェブサイト [26]、設計者ウェブサイト [25]
特徴	Elephant は暗号学的置換をプリミティブとして用いた認証暗号モードの名称であり、3 つの認証暗号 Dumbo、Jumbo、Delirium をまとめた総称である。認証暗号モードとしての構成は Enc-then-MAC 構造であり、暗号化部分は CTR モード、MAC 部分は Protected counter sum [21, 107] と同様の構成である。また、暗号化や MAC の内部構造は Masked Even-Mansour [66] を簡易にしたもので構成されている。NIST LWC プロジェクトのファイナリストのうち入力全体での並列化が可能な唯一の方式であるという特徴がある。 3 つの認証暗号は全てモード構成が Elephant であり、使用する暗号学的置換 P がそれぞれ異なる。各方式における鍵長、nonce 長、P のサイズ、タグ長、P の違いについては、下表のとおり。なお、設計者が推奨する方式は Dumbo である。

方式	鍵長	nonce 長	P のサイズ	タグ長	P
Dumbo	128	96	160	64	Spongent-π[160]
Jumbo	128	96	176	64	Spongent-π[176]
Delirium	128	96	200	128	Keccak-f[200]

安全性解析状況	2022 年 9 月現在、いくつかの解析論文 [4, 27, 28, 97, 122, 129, 135, 147, 153] が発表されているが、仕様書で主張される安全性を脅かす攻撃は提案されていない。 設計者 [26, 27] が Elephant モードの安全性証明を示しており、シングルユーザーとマルチユーザーのいずれの場合においても、仕様書で記載される安全性を担保することが示されている。Elephant モードへの第三者評価としては、2022 年に提案された土生ら [153] による鍵回復、識別及び偽造攻撃が提案されているものの、これらの攻撃は仕様書で主張される安全性バウンドがタイトであることを示す結果となっている。 暗号プリミティブへの安全性解析状況については、Keccak（4.3 節、文献 [29, 30, 58, 96, 112, 131]）と SPONGENT（4.3 節、文献 [1, 144]）の安全性解析状況を参照されたい。その他、耐量子安全性に関する解析論文 [4, 28, 122]、サイドチャネル攻撃耐性に関する解析論文 [97, 129, 135] が報告されているが、これらの安全性について設計者は主張していないため、仕様上の安全性とは矛盾しない。
主な実装評価結果	ハードウェア実装評価結果（FPGA 実装）

Algorithm	Platform	Data	Area	Throughput	Ref.
Dumbo	Artix-7	1,536 Bytes	1,291 LUTs	214.30 Mbps	[2]
Dumbo	Artix-7	Long	2,645 LUTs	1.54 Gbps	[113]

ハードウェア実装評価結果（ASIC 実装：GF 22nm CMOS）

Algorithm	Data	Area	Throughput	Ref.
Elephant	16 Bytes	17.3 kGE	24.0 Mbps	[62]
Elephant	1,536 Bytes	17.3 kGE	70.9 Mbps	[62]

ソフトウェア実装評価結果（Arm Cortex-M0）

Algorithm	Enc	Dec	ROM	Code	Ref.
Dumbo	1,069 msec	1,069 msec	16.4 Kbyte	14.4 Kbyte	[84]
Jumbo	1,255 msec	1,255 msec	16.4 Kbyte	14.4 Kbyte	[84]
Delirium	38.39 msec	38.39 msec	17.0 Kbyte	14.9 Kbyte	[84]

<table>
<tr><td>技術分野</td><td>認証暗号</td></tr>
<tr><td>名称</td><td>GIFT-COFB</td></tr>
<tr><td>設計者</td><td>Subhadeep Banik[1], Avik Chakraborti[2], Akiko Inoue[3], Tetsu Iwata[4], Kazuhiko Minematsu[3], Mridul Nandi[5], Thomas Peyrin[6], Yu Sasaki[7], Siang Meng Sim[6], Yosuke Todo[7]
(1: FHNW/Switzerland, 2: TCG CREST/India, 3: NEC Corporation/Japan, 4: Nagoya University/Japan, 5: Indian Statistical Institute/India, 6: Nanyang Technological University/Singapore, 7: NTT/Japan)</td></tr>
<tr><td>発表年</td><td>2019（NIST LWC ウェブサイト [13]）</td></tr>
<tr><td>仕様参照先</td><td>NIST LWC ウェブサイト [7]、NIST LWC メーリングリスト [6]、設計者ウェブサイト [5]</td></tr>
<tr><td>特徴</td><td>GIFT-COFB はブロック暗号 GIFT-128 [9] をプリミティブとして用いた暗号利用モード COFB [33, 34] に基づく認証暗号である。GIFT-128 は SPN 型ブロック暗号であり、鍵長とブロックサイズが 128 ビット、ラウンド関数を 40 段繰り返す構造を持つ。COFB は n ビットブロック暗号をプリミティブとして用いた認証暗号利用モードであり、実装サイズ、特にハードウェアゲートやソフトウェア上の動作メモリを最小化することに焦点を合わせて提案されたという特徴がある。
GIFT-COFB における推奨パラメータは、鍵長、nonce 長、タグ長がそれぞれ 128 ビットである。また、設計者が主張する安全性レベルは IND-CPA（選択平文攻撃に対する識別不可能性）が 64 ビット、INT-CTXT（暗号文の整合性）が 58 ビットである。</td></tr>
<tr><td>安全性解析状況</td><td>2022 年 9 月現在、いくつかの解析論文 [89, 98, 127, 128, 148] が発表されているが、仕様書で主張される安全性を脅かす攻撃は提案されていない。
認証暗号に対する第三者評価として、2022 年に Khairallah [98] は COFB モードの提案論文 [34] で主張される安全性バウンドに矛盾があることを指摘したが、これは仕様書で記載される安全性とは矛盾がない。また、2022 年に Inoue ら [89] は GIFT-COFB v1.1 [7] で主張される安全性バウンドに矛盾があることを指摘したが、最新版の GIFT-COFB v1.2 [6] において Inoue らの指摘が反映されており、仕様上の安全性とは矛盾しない。
暗号プリミティブに対する第三者評価として、Zong ら [148] による差分攻撃、Sun ら [127, 128] による線形攻撃が提案されているが、攻撃可能段数の最大値は 40 段のうち 27 段であり、安全性マージンが十分に確保されている。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果（FPGA 実装）
<table>
<tr><th>Platform</th><th>Data</th><th>Area</th><th>Throughput</th><th>Ref.</th></tr>
<tr><td>Artix-7</td><td>Long</td><td>1,041 LUTs</td><td>733.3 Mbps</td><td>[113]</td></tr>
<tr><td>Artix-7</td><td>1,536 Bytes</td><td>1,041 LUTs</td><td>364.3 Mbps</td><td>[113]</td></tr>
<tr><td>Artix-7</td><td>Long</td><td>1,730 LUTs</td><td>3.02 Gbps</td><td>[113]</td></tr>
<tr><td>Artix-7</td><td>1,536 Bytes</td><td>1,730 LUTs</td><td>1.48 Gbps</td><td>[113]</td></tr>
</table>
ハードウェア実装評価結果（ASIC 実装：GF 22nm CMOS）
<table>
<tr><th>Data</th><th>Area</th><th>Throughput</th><th>Ref.</th></tr>
<tr><td>16 Bytes</td><td>8.1 kGE</td><td>50.4 Mbps</td><td>[62]</td></tr>
<tr><td>1,536 Bytes</td><td>8.1 kGE</td><td>159.6 Mbps</td><td>[62]</td></tr>
</table>
ソフトウェア実装評価結果（Arm Cortex-M0）
<table>
<tr><th>Enc</th><th>Dec</th><th>ROM</th><th>Code</th><th>Ref.</th></tr>
<tr><td>6.25 msec</td><td>6.25 msec</td><td>17.1 Kbyte</td><td>15.1 Kbyte</td><td>[84]</td></tr>
</table></td></tr>
</table>

<table>
<tr><td>技術分野</td><td>認証暗号</td></tr>
<tr><td>名称</td><td>Grain-128AEAD</td></tr>
<tr><td>設計者</td><td>Martin Hell[1], Thomas Johansson[1], Alexander Maximov[2], Willi Meier[3], Jonathan Sönnerup[1], Hirotaka Yoshida[4]
(1: Lund University/Sweden, 2: Ericsson AB/Sweden, 3: FHNW/Switzerland, 4: AIST/Japan)</td></tr>
<tr><td>発表年</td><td>2019（NIST LWC ウェブサイト [13]）</td></tr>
<tr><td>仕様参照先</td><td>NIST LWC ウェブサイト [81]、設計者ウェブサイト [80]</td></tr>
<tr><td>特徴</td><td>Grain は eSTREAM プロジェクトに応募された初期バージョンの Grain v0 [82] から始まり、Grain v1 [83]、Grain-128 [79]、Grain-128A [3] と、既知の脆弱性を補完 [20, 45, 47]、128 ビット安全性の確保、認証暗号モードの追加、などを経て系譜を継いできた暗号方式である。Grain-128AEAD もまたこの系譜を継ぐ認証暗号であり、Grain v1、Grain-128、Grain-128A（ストリーム暗号モードのみ）に対する既知の脆弱性 [133] に対策を施す形で提案された。また、NIST LWC 選考期間中、Grain-128AEAD の初期バージョンに対する脆弱性 [38] が指摘され、バージョン 2（Grain-128AEADv2）へと仕様が更新されている。
Grain-128AEADv2 は LFSR 型ストリーム暗号ベースの認証暗号であり、128 ビットの LFSR、128 ビットの NFSR、タグ生成用の 64 ビット Accumulator と 64 ビットレジスタから構成されている。鍵長は 128 ビット、nonce 長は 96 ビット、タグ長は 64 ビットであり、鍵と nonce をそれぞれ NFSR と LFSR にロードした後、512 段の初期化フェーズを経て内部状態を初期化する。初期化後、キーストリームを出力するが、奇数番目のキーストリームをタグ生成用として、偶数番目のキーストリームを暗号化用として利用する。</td></tr>
<tr><td>安全性解析状況</td><td>2022 年 9 月現在、Grain-128AEADv2 に対する解析論文は発表されていない。
藤堂 [155] は、Grain 型ストリーム暗号に対する強力な解読法として知られている高速相関攻撃とキューブ攻撃に着目し、Grain-128AEADv2 に対する高速相関攻撃 [20, 133] とキューブ攻撃 [46, 47, 76, 77, 78, 132, 138, 139] の適用可能性について考察した。結果として、最新の高速相関攻撃とキューブ攻撃に対し、Grain-128AEADv2 が十分に大きな安全性マージンを有していることを明らかにした。また、Grain-128A の初期化フェーズ（256 段）に対する条件付き差分攻撃 [101, 108] や関連鍵攻撃 [8, 44] が示されているものの、Grain-128AEADv2 の初期化フェーズは 512 段であり、これらの攻撃もまた安全性を脅かすものではない。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果（FPGA 実装）
<table><tr><th>Platform</th><th>Area</th><th>Throughput</th><th>Ref.</th></tr><tr><td>Spartan-3</td><td>161 LUTs</td><td>152.2 Mbps</td><td>[150]</td></tr><tr><td>Spartan-6</td><td>174 LUTs</td><td>196.8 Mbps</td><td>[150]</td></tr></table>ハードウェア実装評価結果（ASIC 実装：GF 22nm CMOS）
<table><tr><th>Library</th><th>Data</th><th>Unroll</th><th>Area</th><th>Throughput</th><th>Power</th><th>Ref.</th></tr><tr><td>GF 22nm CMOS</td><td>16 Bytes</td><td>–</td><td>4.3 kGE</td><td>9.6 Mbps</td><td>–</td><td>[62]</td></tr><tr><td>GF 22nm CMOS</td><td>1,536 Bytes</td><td>–</td><td>4.3 kGE</td><td>17.7 Mbps</td><td>–</td><td>[62]</td></tr><tr><td>STM 65nm</td><td>–</td><td>1</td><td>2.6 kGE</td><td>1.25 Gbps</td><td>0.25mW</td><td>[125]</td></tr><tr><td>STM 65nm</td><td>–</td><td>64</td><td>16.9 kGE</td><td>33.6 Gbps</td><td>2.76mW</td><td>[125]</td></tr></table>ソフトウェア実装評価結果（Arm Cortex-M0）
<table><tr><th>Enc</th><th>Dec</th><th>ROM</th><th>Code</th><th>Ref.</th></tr><tr><td>82.54 msec</td><td>82.46 msec</td><td>17.8 Kbyte</td><td>15.8 Kbyte</td><td>[84]</td></tr></table></td></tr>
</table>

技術分野	認証暗号
名称	ISAP
設計者	Christoph Dobraunig[1], Maria Eichlseder[1], Stefan Mangard[1], Florian Mendel[2], Bart Mennink[3], Robert Primas[1], Thomas Unterluggauer[1] (1: Graz University of Technology, 2: Infineon Technologies AG/Germany, 3: Radboud University/Netherlands)
発表年	2017 (IACR ToSC 2017 [50])、2019（NIST LWC ウェブサイト [13]）
仕様参照先	NIST LWC ウェブサイト [49]、設計者ウェブサイト [48]
特徴	ISAP は暗号学的置換をプリミティブとして用いた認証暗号モードの名称であり、4 つの認証暗号 ISAP-A-128A、ISAP-K-128A、ISAP-A-128、ISAP-K-128 をまとめた総称である。認証暗号モードとしての構成は Sponge 構造を採用するとともに、Fresh Rekeying [110] と呼ばれる技術から着想を得てサイドチャネル攻撃に対して堅牢となるような設計であることが特徴的であり、Rekey 関数、暗号化関数、MAC 関数で構成されている。 4 つの認証暗号は全てモード構成が ISAP であり、使用する暗号学的置換、レートサイズ、各フェーズにおけるラウンド数などのパラメータがそれぞれ異なる。パラメータの違いについては、下表のとおり（k は安全性レベル（ビット）、その他の細部は仕様参照先を確認されたい）。なお、設計者が推奨する方式は ISAP-A-128A である。

方式	暗号学的置換	ビットサイズ				ラウンド数			
		k	n	r_H	r_B	s_H	s_B	s_E	s_K
ISAP-A-128A	Ascon-p	128	320	64	1	12	1	6	12
ISAP-K-128A	Keccak-p[400]	128	400	144	1	16	1	8	8
ISAP-A-128	Ascon-p	128	320	64	1	12	12	12	12
ISAP-K-128	Keccak-p[400]	128	400	144	1	20	12	12	12

安全性解析状況	2022 年 9 月現在、いくつかの解析論文 [14, 52, 53, 55, 71, 94, 134, 145] が発表されているが、仕様書で主張される安全性を脅かす攻撃は提案されていない。 Rekey 関数と暗号化関数の安全性については、Daemen ら [42] による Keyed duplex の安全性証明に依拠していることが設計者 [49] によって述べられている。MAC 関数の安全性については、Dobraunig ら [53, 55] によって Suffix keyed sponge の安全性証明が与えられており、128 ビット安全性がタイトであると述べられている。 暗号プリミティブへの安全性解析状況については、Keccak（4.3 節、文献 [29, 30, 58, 96, 112, 131]）と Ascon（4.5 節、文献 [12, 51, 63, 64, 85, 104, 109]）の安全性解析状況を参照されたい。その他、耐漏洩安全性に関する解析論文 [52, 53, 71]、サイドチャネル攻撃耐性に関する解析論文 [14, 134, 145]、量子識別攻撃 [94] に関する解析論文が報告されているが、これらの解析に関しても仕様上の安全性とは矛盾しない。
主な実装評価結果	ハードウェア実装評価結果（FPGA 実装）

Algorithm	Platform	Data	Area	Throughput	Ref.
ISAP-A-128A	Artix-7	1,536 Bytes	3,491 LUTs	389.4 Mbps	[113]
ISAP-A-128A	Artix-7	Long	3,491 LUTs	829.6 Mbps	[113]

ソフトウェア実装評価結果（Arm Cortex-M0）

Algorithm	Enc	Dec	ROM	Code	Ref.
ISAP-A-128A	9.66 msec	9.66 msec	16.5 Kbyte	14.5 Kbyte	[84]
ISAP-A-128	39.49 msec	39.50 msec	16.5 Kbyte	14.5 Kbyte	[84]
ISAP-K-128A	161.9 msec	161.9 msec	16.6 Kbyte	14.5 Kbyte	[84]
ISAP-K-128	1,366 msec	1,366 msec	16.6 Kbyte	14.5 Kbyte	[84]

<table>
<tr><td>技術分野</td><td>認証暗号、ハッシュ関数</td></tr>
<tr><td>名称</td><td>PHOTON-Beetle</td></tr>
<tr><td>設計者</td><td>Zhenzhen Bao[1], Avik Chakraborti[2], Nilanjan Datta[3], Jian Guo[1], Mridul Nandi[3], Thomas Peyrin[1], Kan Yasuda[4]
(1: Nanyang Technological University/Singapore., 2: University of Exeter/UK, 3: Indian Statistical Institute/India 4: NTT/Japan)</td></tr>
<tr><td>発表年</td><td>2019（NIST LWC ウェブサイト [13]）</td></tr>
<tr><td>仕様参照先</td><td>NIST LWC ウェブサイト [11]、設計者ウェブサイト [10]</td></tr>
<tr><td>特徴</td><td>PHOTON-Beetle は暗号学的置換をプリミティブとして用いた認証暗号 PHOTON-Beetle-AEAD[r] とハッシュ関数 PHOTON-Beetle-Hash をまとめた総称である。なお、パラメータ r はレートサイズ（ビット）を表す。これらの方式はハッシュ関数 PHOTON [72] で使用されている 256 ビットブロックの暗号学的置換を構成要素としている。また、認証暗号は Duplex Sponge [24] を改良した Beetle モード [32] に基づいて設計され、ハッシュ関数は Sponge 構造 [23] に基づいて設計されている。
認証暗号ではレートサイズとして $r = 32$ あるいは $r = 128$ を選択でき、いずれの選択においても鍵長、nonce 長、タグ長は 128 ビットである。なお、設計者が推奨する方式は $r = 128$ の PHOTON-Beetle-AEAD[128] である。ハッシュ関数ではレートサイズとして $r = 32$ 以外の場合を推奨しておらず、任意長の入力から 256 ビットのハッシュ値を出力する。</td></tr>
<tr><td>安全性解析状況</td><td>2022 年 9 月現在、いくつかの解析論文 [36, 37, 54, 89, 93, 100, 111] が発表されているが、仕様書で主張される安全性を脅かす攻撃は提案されていない。
設計者 [11] は認証暗号とハッシュ関数に関する安全性バウンドを示している。これらの安全性バウンドは第三者評価の結果 [54, 111] が反映されている。認証暗号に対する第三者評価として、2020 年に報告された Chakraborty ら [36, 37] による安全性証明、2020 年に提案された Dobraunig ら [54] による鍵回復攻撃、2022 年に提案された Inoue ら [89] による偽造攻撃と識別攻撃が示されているが、これらは仕様書で主張される安全性に矛盾がないことを示す結果となっている。Inoue ら [89] は関連鍵設定における効率的な偽造攻撃も示したが、この攻撃は限定的なシナリオでのみ成立するものであり、このシナリオが成立しないように実装することで回避できる。ハッシュ関数に対する第三者評価として、2021 年に提案された Mége [111] による衝突攻撃、2022 年に提案された Lefevre ら [100] による原像攻撃が示されているが、これらも仕様書で主張される安全性に矛盾がないことを示す結果となっている。
暗号プリミティブへの安全性解析状況については、PHOTON（4.3 節、文献 [95, 136, 137]）の安全性解析状況を参照されたい。その他、Jana ら [93] による効率的なサイドチャネル攻撃が示されているものの、この攻撃に対しても実装面での対策が有効である。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果（FPGA 実装）
<table>
<tr><th>Algorithm</th><th>Rate</th><th>Platform</th><th>Data</th><th>Area</th><th>Throughput</th><th>Ref.</th></tr>
<tr><td>認証暗号</td><td>128</td><td>Artix-7</td><td>1,536 Bytes</td><td>2,065 LUTs</td><td>370.4 Mbps</td><td>[113]</td></tr>
<tr><td>認証暗号</td><td>128</td><td>Artix-7</td><td>Long</td><td>2,065 LUTs</td><td>747.0 Mbps</td><td>[113]</td></tr>
<tr><td>ハッシュ関数</td><td>32</td><td>Artix-7</td><td>1,536 Bytes</td><td>2,065 LUTs</td><td>228.6 Mbps</td><td>[113]</td></tr>
<tr><td>ハッシュ関数</td><td>32</td><td>Artix-7</td><td>Long</td><td>2,065 LUTs</td><td>227.8 Mbps</td><td>[113]</td></tr>
</table>
ソフトウェア実装評価結果（Arm Cortex-M0）
<table>
<tr><th>Algorithm</th><th>Rate</th><th>Enc</th><th>Dec</th><th>ROM</th><th>Code</th><th>Ref.</th></tr>
<tr><td>認証暗号</td><td>128</td><td>42.39 msec</td><td>42.40 msec</td><td>17.5 Kbyte</td><td>15.4 Kbyte</td><td>[84]</td></tr>
<tr><td>認証暗号</td><td>32</td><td>102.6 msec</td><td>102.6 msec</td><td>17.6 Kbyte</td><td>15.5 Kbyte</td><td>[84]</td></tr>
</table></td></tr>
</table>

<table>
<tr><td>技術分野</td><td>認証暗号、ハッシュ関数</td></tr>
<tr><td>名称</td><td>Romulus</td></tr>
<tr><td>設計者</td><td>Chun Guo[1], Tetsu Iwata[2], Mustafa Khairallah[3], Kazuhiko Minematsu[4], Thomas Peyrin[3]
(1: Shandong University/China, 2: Nagoya University/Japan, 3: Nanyang Technological University/Singapore, 4: NEC Corporation/Japan)</td></tr>
<tr><td>発表年</td><td>2019（NIST LWC ウェブサイト [13])、2020 (IACR ToSC 2020 [91])</td></tr>
<tr><td>仕様参照先</td><td>NIST LWC ウェブサイト [68]、設計者ウェブサイト [67]</td></tr>
<tr><td>特徴</td><td>Romulus は tweakable ブロック暗号（TBC）をプリミティブとして用いた暗号利用モードの名称であり、認証暗号 Romulus-N、Romulus-M、Romulus-T とハッシュ関数 Romulus-H をまとめた総称である。認証暗号の 3 方式はそれぞれ達成する安全性の種類が異なり、それに応じてモードの構成も変わることが特徴である。
Romulus-N は nonce-respecting 設定下で安全性が保証され、COFB モード [33, 34] をベースとしている。Romulus-M は nonce-misuse 設定下で安全性が保証され、SIV モード [119] をベースとしている。Romulus-T は耐漏洩安全性が保証され、TEDT モード [22] をベースとしている。3 方式とも nonce 長、鍵長、タグ長は 128 ビット、tweak 長は 256 ビットである。なお、設計者が推奨する方式は Romulus-N である。Romulus-H は MDPH 構造 [115] を採用しており、任意長の入力から 256 ビットのハッシュ値を出力する。
使用するプリミティブは Skinny-128-384+ であり、この方式は Skinny [19, 90] のインスタンスの 1 つである Skinny-128-384 の段数を 56 段から 40 段に削減したものである。</td></tr>
<tr><td>安全性解析状況</td><td>2022 年 9 月現在、いくつかの解析論文 [35, 56, 73, 99, 114, 145, 154] が発表されているが、仕様書で主張される安全性を脅かす攻撃は提案されていない。
設計者 [70, 91] は全モードの安全性証明を示している。認証暗号に対する第三者評価として、2020 年に報告された Lee [99] による安全性証明、2022 年に提案された Habu ら [73, 154] による識別攻撃、偽造攻撃、マッチング攻撃が示されているが、これらは仕様書で主張される安全性バウンドがタイトであることを示す結果となっている。また、ハッシュ関数に対する第三者評価として、2021 年に提案された Dong ら [56] による衝突攻撃と 2023 年に提案された Nageler ら [114] による衝突攻撃が示されているが、これらはプリミティブを簡略化した場合にのみ有効であり、ハッシュ関数の安全性を脅かすものではない。
プリミティブに対していくつかの解析論文 [31, 43, 56, 57, 74, 75, 87, 118, 124] が発表されているが、その仕様上の安全性を脅かす攻撃については提案されていない。オリジナルの Skinny-128-384 と仕様が異なるため、Skinny-128-384 に対する攻撃が必ずしも Skinny-128-384+ に適用できるとは限らない。設計者 [69] の分析によると、40 段のうち単一鍵設定では 22 段まで、関連鍵設定では 26 段まで鍵回復攻撃が可能であると見積もっている。ただし、これらは 256 ビット鍵を使用した場合における結果であることに注意されたい。</td></tr>
<tr><td>主な実装評価結果</td><td>ハードウェア実装評価結果（FPGA 実装）
<table>
<tr><th>Algorithm</th><th>Platform</th><th>Data</th><th>Area</th><th>Throughput</th><th>Ref.</th></tr>
<tr><td>Romulus-N</td><td>Artix-7</td><td>1,536 Bytes</td><td>1,280 LUTs</td><td>542.0 Mbps</td><td>[113]</td></tr>
<tr><td>Romulus-N</td><td>Artix-7</td><td>Long</td><td>1,280 LUTs</td><td>1.09 Gbps</td><td>[113]</td></tr>
<tr><td>Romulus-N</td><td>Artix-7</td><td>1,536 Bytes</td><td>953 LUTs</td><td>315.7 Mbps</td><td>[113]</td></tr>
<tr><td>Romulus-N</td><td>Artix-7</td><td>Long</td><td>953 LUTs</td><td>637.2 Mbps</td><td>[113]</td></tr>
</table>
ソフトウェア実装評価結果（Arm Cortex-M0）
<table>
<tr><th>Algorithm</th><th>Enc</th><th>Dec</th><th>ROM</th><th>Code</th><th>Ref.</th></tr>
<tr><td>Romulus-N</td><td>11.12 msec</td><td>11.13 msec</td><td>19.5 Kbyte</td><td>16.9 Kbyte</td><td>[84]</td></tr>
<tr><td>Romulus-M</td><td>14.48 msec</td><td>14.49 msec</td><td>19.7 Kbyte</td><td>17.1 Kbyte</td><td>[84]</td></tr>
</table></td></tr>
</table>

技術分野	認証暗号、ハッシュ関数
名称	Sparkle
設計者	Christof Beierle[1,2], Alex Biryukov[1], Luan Cardoso dos Santos[1], Johann Großschädl[1], Amir Moradi[2], Léo Perrin[3], Aein Rezaei Shahmirzadi[2], Aleksei Udovenko[1,4], Vesselin Velichkov[5], Qingju Wang[1] (1: University of Luxembourg/Luxembourg, 2: Ruhr University Bochum/Germany, 3: INRIA/France, 4: CryptoExperts/France, 5: University of Edinburgh/UK)
発表年	2019（NIST LWC ウェブサイト [13])、2020 (IACR ToSC 2020 [18])
仕様参照先	NIST LWC ウェブサイト [16]、設計者ウェブサイト [15]
特徴	Sparkle はブロック暗号 Alzette [17] を構成要素とした暗号学的置換であり、Sparkle を暗号プリミティブとして用いた Sponge 構造に基づく認証暗号 Schwaemm とハッシュ関数 Esch が定義されている。 Sparkle は Sparkle256、Sparkle384、Sparkle512 の 3 種類の暗号学的置換をまとめた総称である（数値：入出力サイズ)。それぞれ段数の異なる 2 つのインスタンス（slim、big）が定義されている。例えば、Sparkle384 の slim は 7 段、big は 11 段である。Schwaemm は nonce ベースの認証暗号であり、4 通りのパラメータが定義されている。設計者が推奨する方式は Sparkle384 を使用した Schwaemm256-128 であり、鍵長、タグ長、キャパシティが 128 ビット、nonce 長とレートが 256 ビット、主張する安全性が 120 ビットである。ハッシュ関数 Esch は Esch256 と Esch384 の 2 通りが定義されている。設計者が推奨する方式は Sparkle384 を使用した Esch256 であり、キャパシティが 128 ビット、レートが 256 ビット、任意長の入力から 256 ビットのハッシュ値を出力する。
安全性解析状況	2022 年 9 月現在、いくつかの解析論文 [88, 92, 105, 106, 117, 121, 126, 142, 143] が発表されているが、仕様書で主張される安全性を脅かす攻撃は提案されていない。 設計者 [16, 18] は Sparkle に対して広範な攻撃手法に対する安全性解析結果を示している。第三者評価として、2022 年に提案された Schrottenloher ら [121] による推測決定攻撃が示されているが、これは仕様上の安全性を脅かすものではない。Alzette に対する識別攻撃 [88, 105, 106, 117, 142] がいくつか発表されているものの、Alzette は Sparkle において S-box として機能しており、Alzette への識別攻撃が Sparkle の安全性を直ちに脅かすものではない。 認証暗号は Beetle モード [32] に基づいて設計されており、設計者 [16] はその安全性が Beetle の安全性バウンドに帰着できると主張している。また、ハッシュ関数の安全性は Hirose [86] が示す証明可能安全性の結果に基づいている。なお、これらの方式の安全性を脅かす第三者評価は発表されていない。岩田 [151] によると、設計者 [16] の安全性証明において一部精査が必要な箇所があるものの、大部分において問題とならないと結論付けられている。
主な実装評価結果	ハードウェア実装評価結果（FPGA 実装）

Algorithm	Platform	Data	Area	Throughput	Ref.
Schwaemm256-128	Artix-7	1,536 Bytes	3,071 LUTs	396.8 Mbps	[113]
Schwaemm256-128	Artix-7	Long	3,071 LUTs	831.2 Mbps	[113]
Esch256	Artix-7	1,536 Bytes	3,740 LUTs	481.2 Mbps	[113]
Esch256	Artix-7	Long	3,740 LUTs	489.4 Mbps	[113]

ソフトウェア実装評価結果（Arm Cortex-M0）

Algorithm	Enc	Dec	ROM	Code	Ref.
Schwaemm128-128	0.76 msec	0.77 msec	16.9 Kbyte	14.9 Kbyte	[84]
Schwaemm256-128	0.93 msec	0.93 msec	17.1 Kbyte	15.1 Kbyte	[84]

技術分野	認証暗号
名称	TinyJAMBU
設計者	Hongjun Wu, Tao Huang (Nanyang Technological University/Singapore)
発表年	2019（NIST LWC ウェブサイト [13]）
仕様参照先	NIST LWC ウェブサイト [141]
特徴	TinyJAMBU は鍵付き暗号学的置換をプリミティブとして用いた Sponge 構造に基づく認証暗号であり、TinyJAMBU-128、TinyJAMBU-192、TinyJAMBU-256 の 3 方式をまとめた総称である。CAESAR competition の第 3 ラウンド候補の 1 つである JAMBU [140] の軽量版として提案された。 鍵付き暗号学的置換の内部状態は 128 ビットの NFSR で構成されており、秘密鍵をロードしながら内部状態を更新する。また、認証暗号において 2 種類の暗号学的置換（$P1$、$P2$）を使用するが、これらは内部状態の更新回数（段数）が異なるのみである。3 種類の認証暗号方式におけるパラメータの違いについては、下表のとおり。TinyJAMBU は NIST LWC の最終ラウンドにおいて仕様が更新されたが、主な違いは $P1$ の段数であり、更新前の仕様段数は 384 段であった。なお、設計者が推奨する方式は TinyJAMBU-128 である。 方式 / 鍵長 / nonce 長 / タグ長 / $P1$ の段数 / $P2$ の段数 TinyJAMBU-128 / 128 / 96 / 64 / 640 / 1024 TinyJAMBU-192 / 192 / 96 / 64 / 640 / 1152 TinyJAMBU-256 / 256 / 96 / 64 / 640 / 1280
安全性解析状況	2022 年 9 月現在、いくつかの解析論文 [59, 60, 102, 120, 123, 130] が発表されているが、仕様書で主張される安全性を脅かす攻撃は提案されていない。 認証暗号に対する第三者評価として、2020 年に提案された Saha ら [120] の差分攻撃と線形攻撃、2022 年に提案された Li ら [102] の線形攻撃が示されている。Saha ら [120] の攻撃では $P1$ を 338 段に簡略化した場合に偽造攻撃が可能であり、この報告を受け設計者は $P1$ の段数を 640 段に修正した。Li ら [102] の攻撃では $P1$ を 387 段に簡略化した場合に鍵回復攻撃が可能であり、これは仕様修正前の TinyJAMBU が安全でなかったことを示している。最新の TinyJAMBU は安全性マージンが十分に確保されており、この攻撃が仕様上の安全性を脅かすものではない。 その他、2022 年に Sibleyras ら [123] は鍵付き暗号学的置換をブロック暗号とみなし、$P2$ の段数に関係なくスライド攻撃によって解読可能であることを示した。この攻撃はブロック暗号としての安全性を有していないことを示すものであり、認証暗号としての TinyJAMBU に対してこの攻撃は有効ではない。2022 年に Dunkelman ら [59, 60] は関連鍵設定において TinyJAMBU-192 と TinyJAMBU-256 に対する現実的な計算量での偽造攻撃が実行可能であることを示した。本攻撃が成立するような関連鍵（と nonce）の使用を避けることで安全性を確保できる。
主な実装評価結果	ハードウェア実装評価結果（FPGA 実装） Algorithm / Platform / Data / Area / Throughput / Ref. TinyJAMBU-128 / Artix-7 / 1,536 Bytes / 591 LUTs / 250.4 Mbps / [2] TinyJAMBU-128 / Artix-7 / 1,536 Bytes / 591 LUTs / 176.1 Mbps / [113] TinyJAMBU-128 / Artix-7 / Long / 591 LUTs / 354.7 Mbps / [113] ソフトウェア実装評価結果（Arm Cortex-M0） Algorithm / Enc / Dec / ROM / Code / Ref. TinyJAMBU-128 / 0.39 msec / 0.39 msec / 15.7 Kbyte / 13.7 Kbyte / [84] TinyJAMBU-192 / 4.63 msec / 4.63 msec / 15.7 Kbyte / 13.7 Kbyte / [84] TinyJAMBU-256 / 0.44 msec / 0.44 msec / 15.7 Kbyte / 13.7 Kbyte / [84]

技術分野	認証暗号、ハッシュ関数
名称	Xoodyak
設計者	Joan Daemen[1], Seth Hoffert[1,2], Michaël Peeters[2], Gilles Van Assche[2], Ronny Van Keer[2], Silvia Mella[1,2] (1: Radboud University/Netherlands, 2: STMicroelectronics/Switzerland)
発表年	2019（NIST LWC ウェブサイト [13]）
仕様参照先	NIST LWC ウェブサイト [41]、設計者ウェブサイト [40]
特徴	Xoodyak は暗号学的置換 Xoodoo [39] をプリミティブとして用いた Duplex 構造 [24, 42] に基づく認証暗号と Sponge 構造 [23] に基づくハッシュ関数の総称である。Xoodoo は Keccak-p [116] から着想を得て設計された暗号学的置換であり、ブロックサイズが 384 ビット、ラウンド関数を 12 段繰り返す構造を持つ。 認証暗号ではレートサイズが 256 ビット、鍵長、nonce 長、タグ長はそれぞれ 128 ビットが推奨されている。ハッシュ関数ではレートサイズが 128 ビットであり、任意長の入力から 256 ビットのハッシュ値を出力する。いずれも 128 ビット安全性が主張されている。
安全性解析状況	2022 年 9 月現在、いくつかの解析論文 [61, 103, 146] が発表されているが、仕様書で主張される安全性を脅かす攻撃は提案されていない。 認証暗号に関して、文献 [42] によると、推奨パラメータを使用した場合の安全性レベル 64 ビットであると証明されている。設計者の安全性主張を破る攻撃は存在しないが、安全性証明と設計者の主張に差があることには注意が必要である。その他、2020 年に提案された Zhou ら [146] による条件付きキューブ攻撃、2022 年に提案された Dunkelman ら [61] による差分線形攻撃があるが、7 段以上の Xoodoo に対して適用可能な攻撃が存在しないため、これらの攻撃が認証暗号の安全性を脅かすものではない。 ハッシュ関数に関して、文献 [23] の安全性証明と設計者の安全性主張に矛盾はない。その他、ハッシュ関数に対する第三者評価は報告されていない。 暗号プリミティブに対する第三者評価として、2020 年に提案された Liu ら [103] によるゼロサム識別攻撃がある。仕様段数の Xoodoo に対して 2^{33} の計算量でゼロサム識別子が構成可能であることが報告されているが、設計者 [41] が考察しているように、この攻撃が Xoodyak の安全性に直接影響を及ぼすものではない。 2023 年に Gilbert ら [65] は、Duplex ベースの認証暗号モードに対する汎用的な攻撃手法を提案した。この攻撃を Xoodyak に適用した場合、秘密鍵やタグの長さに依存せず、2^{148} の計算量で平文回復攻撃と偽造攻撃が可能となる。この攻撃は設計者 [41] が提供する安全性主張を破るものであるが、NIST が要求する 112 ビット安全性レベルを脅かすものではない。
主な実装評価結果	ハードウェア実装評価結果（FPGA 実装）

Algorithm	Platform	Data	Area	Throughput	Ref.
認証暗号	Artix-7	1,536 Bytes	1,808 LUTs	1.71 Gbps	[2]
認証暗号	Artix-7	1,536 Bytes	1,608 LUTs	2.89 Gbps	[113]
認証暗号	Artix-7	Long	1,608 LUTs	6.56 Gbps	[113]
ハッシュ関数	Artix-7	1,536 Bytes	1,608 LUTs	3.01 Gbps	[113]
ハッシュ関数	Artix-7	Long	1,608 LUTs	3.09 Gbps	[113]

ハードウェア実装評価結果（ASIC 実装：GF 22nm CMOS）

Algorithm	Data	Area	Throughput	Ref.
認証暗号	16 Bytes	11.9 kGE	79.2 Mbps	[62]
認証暗号	1,536 Bytes	11.9 kGE	1.02 Gbps	[62]

ソフトウェア実装評価結果（Arm Cortex-M0）

Algorithm	Enc	Dec	ROM	Code	Ref.
認証暗号	0.59 msec	0.60 msec	16.3 Kbyte	14.3 Kbyte	[84]

参考文献

[1] Abdelraheem, M.A.: Estimating the Probabilities of Low-Weight Differential and Linear Approximations on PRESENT-Like Ciphers. In: Kwon, T., Lee, M., Kwon, D. (eds.) Information Security and Cryptology - ICISC 2012 - 15th International Conference, Seoul, Korea, November 28-30, 2012, Revised Selected Papers. Lecture Notes in Computer Science, vol. 7839, pp. 368–382. Springer (2012), https://doi.org/10.1007/978-3-642-37682-5_26

[2] Abdulgadir, A., Haeussler, R., Lin, S., Kaps, J.P., Gaj, K.: Side-Channel Resistant Implementations of Three Finalists of the NIST Lightweight Cryptography Standardization Process: Elephant, TinyJAMBU, and Xoodyak. Lightweight Cryptography Workshop 2022 pp. 1–7 (2022)

[3] Ågren, M., Hell, M., Johansson, T., Meier, W.: Grain-128a: a new version of Grain-128 with optional authentication. Int. J. Wirel. Mob. Comput. 5(1), 48–59 (2011), https://doi.org/10.1504/IJWMC.2011.044106

[4] Alagic, G., Bai, C., Katz, J., Majenz, C., Struck, P.: Post-Quantum Security of the (Tweakable) FX Construction, and Applications. IACR Cryptol. ePrint Arch. p. 1097 (2022), https://eprint.iacr.org/2022/1097

[5] Banik, S., Chakraborti, A., Inoue, A., Iwata, T., Minematsu, K., Nandi, M., Peyrin, T., Sasaki, Y., Sim, S.M., Todo, Y.: GIFT-COFB Authenticated Encryption, https://www.isical.ac.in/~lightweight/COFB/（2023-10-04 閲覧）

[6] Banik, S., Chakraborti, A., Inoue, A., Iwata, T., Minematsu, K., Nandi, M., Peyrin, T., Sasaki, Y., Sim, S.M., Todo, Y.: GIFT-COFB v1.2. Submission to the NIST Lightweight Cryptography project (2022), https://groups.google.com/a/list.nist.gov/g/lwc-forum/c/7BmjTeE-NsY?pli=1（2023-10-04 閲覧）

[7] Banik, S., Chakraborti, A., Iwata, T., Minematsu, K., Nandi, M., Peyrin, T., Sasaki, Y., Sim, S.M., Todo, Y.: GIFT-COFB v1.1. Submission to the NIST Lightweight Cryptography project (2021)

[8] Banik, S., Maitra, S., Sarkar, S., Turan, M.S.: A Chosen IV Related Key Attack on Grain-128a. In: Boyd, C., Simpson, L. (eds.) Information Security and Privacy - 18th Australasian Conference, ACISP 2013, Brisbane, Australia, July 1-3, 2013. Proceedings. Lecture Notes in Computer Science, vol. 7959, pp. 13–26. Springer (2013), https://doi.org/10.1007/978-3-642-39059-3_2

[9] Banik, S., Pandey, S.K., Peyrin, T., Sasaki, Y., Sim, S.M., Todo, Y.: GIFT: A Small Present - Towards Reaching the Limit of Lightweight Encryption. In: Fischer, W., Homma, N. (eds.) Cryptographic Hardware and Embedded Systems - CHES 2017 - 19th International Conference, Taipei, Taiwan, September 25-28, 2017, Proceedings. Lecture Notes in Computer Science, vol. 10529, pp. 321–345. Springer (2017), https://doi.org/10.1007/978-3-319-66787-4_16

[10] Bao, Z., Chakraborti, A., Datta, N., Guo, J., Nandi, M., Peyrin, T., Yasuda, K.: PHOTON-Beetle Authenticated Encryption, https://www.isical.ac.in/~lightweight/beetle/（2023-10-04 閲覧）

[11] Bao, Z., Chakraborti, A., Datta, N., Guo, J., Nandi, M., Peyrin, T., Yasuda, K.: PHOTON-Beetle Authenticated Encryption and Hash Family. Submission to the NIST Lightweight Cryptography project (2021)

[12] Bar-On, A., Dunkelman, O., Keller, N., Weizman, A.: DLCT: A New Tool for Differential-Linear Cryptanalysis. In: Ishai, Y., Rijmen, V. (eds.) Advances in Cryptology - EUROCRYPT 2019 - 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19-23, 2019, Proceedings, Part I. Lecture Notes in Computer Science, vol. 11476, pp. 313–342. Springer (2019), https:

//doi.org/10.1007/978-3-030-17653-2_11

[13] Bassham, L., Chang, D., Kang, J., Kelsey, J., McKay, K., Turan, M.S., Waller, N.: NIST Lightweight Cryptography Project, https://csrc.nist.gov/projects/lightweight-cryptography

[14] Batina, L., Buhan, I., Chmielewski, L., Gunnarsdóttir, E., Jahandideh, V., Stock, T., Weissbart, L.: Side-Channel Evaluation Report on Implementations of Several NIST LWC Finalists (2022), https://cryptography.gmu.edu/athena/LWC/Reports/Radboud/Radboud_Report_SW_3_candidates.pdf（2023-10-04 閲覧）

[15] Beierle, C., Biryukov, A., dos Santos, L.C., Großschädl, J., Moradi, A., Perrin, L., Shahmirzadi, A.R., Udovenko, A., Velichkov, V., Wang, Q.: Sparkle Suite: A collection of lightweight symmetric cryptographic primitives, finalist of the ongoing NIST lightweight standardisation effort, https://sparkle-lwc.github.io/（2023-10-04 閲覧）

[16] Beierle, C., Biryukov, A., dos Santos, L.C., Großschädl, J., Moradi, A., Perrin, L., Shahmirzadi, A.R., Udovenko, A., Velichkov, V., Wang, Q.: Schwaemm and Esch: Lightweight Authenticated Encryption and Hashing using the Sparkle Permutation Family. Submission to the NIST Lightweight Cryptography project (2021)

[17] Beierle, C., Biryukov, A., dos Santos, L.C., Großschädl, J., Perrin, L., Udovenko, A., Velichkov, V., Wang, Q.: Alzette: A 64-Bit ARX-box - (Feat. CRAX and TRAX). In: Micciancio, D., Ristenpart, T. (eds.) Advances in Cryptology - CRYPTO 2020 - 40th Annual International Cryptology Conference, CRYPTO 2020, Santa Barbara, CA, USA, August 17-21, 2020, Proceedings, Part III. Lecture Notes in Computer Science, vol. 12172, pp. 419–448. Springer (2020), https://doi.org/10.1007/978-3-030-56877-1_15

[18] Beierle, C., Biryukov, A., dos Santos, L.C., Großschädl, J., Perrin, L., Udovenko, A., Velichkov, V., Wang, Q.: Lightweight AEAD and Hashing using the Sparkle Permutation Family. IACR Trans. Symmetric Cryptol. 2020(S1), 208–261 (2020), https://doi.org/10.13154/tosc.v2020.iS1.208-261

[19] Beierle, C., Jean, J., Kölbl, S., Leander, G., Moradi, A., Peyrin, T., Sasaki, Y., Sasdrich, P., Sim, S.M.: The SKINNY Family of Block Ciphers and Its Low-Latency Variant MANTIS. In: Robshaw, M., Katz, J. (eds.) Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part II. Lecture Notes in Computer Science, vol. 9815, pp. 123–153. Springer (2016), https://doi.org/10.1007/978-3-662-53008-5_5

[20] Berbain, C., Gilbert, H., Maximov, A.: Cryptanalysis of Grain. In: Robshaw, M.J.B. (ed.) Fast Software Encryption, 13th International Workshop, FSE 2006, Graz, Austria, March 15-17, 2006, Revised Selected Papers. Lecture Notes in Computer Science, vol. 4047, pp. 15–29. Springer (2006), https://doi.org/10.1007/11799313_2

[21] Bernstein, D.J.: How to Stretch Random Functions: The Security of Protected Counter Sums. J. Cryptol. 12(3), 185–192 (1999), https://doi.org/10.1007/s001459900051

[22] Berti, F., Guo, C., Pereira, O., Peters, T., Standaert, F.: TEDT, a Leakage-Resist AEAD Mode for High Physical Security Applications. IACR Trans. Cryptogr. Hardw. Embed. Syst. 2020(1), 256–320 (2020), https://doi.org/10.13154/tches.v2020.i1.256-320

[23] Bertoni, G., Daemen, J., Peeters, M., Assche, G.V.: On the Indifferentiability of the Sponge Construction. In: Smart, N.P. (ed.) Advances in Cryptology - EUROCRYPT 2008, 27th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Istanbul, Turkey, April 13-17, 2008. Proceedings. Lecture Notes in Computer Science, vol. 4965, pp. 181–197. Springer (2008), https://doi.org/10.1007/978-3-540-78967-3_11

[24] Bertoni, G., Daemen, J., Peeters, M., Assche, G.V.: Duplexing the Sponge: Single-Pass Authenticated Encryption and Other Applications. In: Miri, A., Vaudenay, S. (eds.) Selected Areas in Cryptography - 18th International Workshop, SAC 2011, Toronto, ON, Canada, August 11-12, 2011, Revised Selected Papers. Lecture Notes in Computer Science, vol. 7118, pp. 320–337. Springer (2011), https://doi.org/10.1007/978-3-642-28496-0_19

[25] Beyne, T., Chen, Y.L., Dobraunig, C., Mennink, B.: Elephant, https://www.esat.kuleuven.be/cosic/

elephant/（2023-10-04 閲覧）

[26] Beyne, T., Chen, Y.L., Dobraunig, C., Mennink, B.: Elephant v2.0. Submission to the NIST Lightweight Cryptography project (2021)

[27] Beyne, T., Chen, Y.L., Dobraunig, C., Mennink, B.: Multi-user Security of the Elephant v2 Authenticated Encryption Mode. In: AlTawy, R., Hülsing, A. (eds.) Selected Areas in Cryptography - 28th International Conference, SAC 2021, Virtual Event, September 29 - October 1, 2021, Revised Selected Papers. Lecture Notes in Computer Science, vol. 13203, pp. 155–178. Springer (2021), https://doi.org/10.1007/978-3-030-99277-4_8

[28] Bonnetain, X., Jaques, S.: Quantum Period Finding against Symmetric Primitives in Practice. IACR Trans. Cryptogr. Hardw. Embed. Syst. 2022(1), 1–27 (2022), https://doi.org/10.46586/tches.v2022.i1.1-27

[29] Boura, C., Canteaut, A.: Zero-sum distinguishers for iterated permutations and application to keccak-f and hamsi-256. In: Biryukov, A., Gong, G., Stinson, D.R. (eds.) Selected Areas in Cryptography - 17th International Workshop, SAC 2010, Waterloo, Ontario, Canada, August 12-13, 2010, Revised Selected Papers. Lecture Notes in Computer Science, vol. 6544, pp. 1–17. Springer (2010), https://doi.org/10.1007/978-3-642-19574-7_1

[30] Boura, C., Canteaut, A., Cannière, C.D.: Higher-order differential properties of keccak and *Luffa*. In: Joux, A. (ed.) Fast Software Encryption - 18th International Workshop, FSE 2011, Lyngby, Denmark, February 13-16, 2011, Revised Selected Papers. Lecture Notes in Computer Science, vol. 6733, pp. 252–269. Springer (2011), https://doi.org/10.1007/978-3-642-21702-9_15

[31] Boura, C., David, N., Derbez, P., Leander, G., Naya-Plasencia, M.: Differential Meet-In-The-Middle Cryptanalysis. IACR Cryptol. ePrint Arch. p. 1640 (2022), https://eprint.iacr.org/2022/1640

[32] Chakraborti, A., Datta, N., Nandi, M., Yasuda, K.: Beetle Family of Lightweight and Secure Authenticated Encryption Ciphers. IACR Trans. Cryptogr. Hardw. Embed. Syst. 2018(2), 218–241 (2018), https://doi.org/10.13154/tches.v2018.i2.218-241

[33] Chakraborti, A., Iwata, T., Minematsu, K., Nandi, M.: Blockcipher-Based Authenticated Encryption: How Small Can We Go? In: Fischer, W., Homma, N. (eds.) Cryptographic Hardware and Embedded Systems - CHES 2017 - 19th International Conference, Taipei, Taiwan, September 25-28, 2017, Proceedings. Lecture Notes in Computer Science, vol. 10529, pp. 277–298. Springer (2017), https://doi.org/10.1007/978-3-319-66787-4_14

[34] Chakraborti, A., Iwata, T., Minematsu, K., Nandi, M.: Blockcipher-Based Authenticated Encryption: How Small Can We Go? J. Cryptol. 33(3), 703–741 (2020), https://doi.org/10.1007/s00145-019-09325-z

[35] Chakraborty, A., Singh, N., Bhattacharya, S., Rebeiro, C., Mukhopadhyay, D.: Timed speculative attacks exploiting store-to-load forwarding bypassing cache-based countermeasures. In: Oshana, R. (ed.) DAC '22: 59th ACM/IEEE Design Automation Conference, San Francisco, California, USA, July 10 - 14, 2022. pp. 553–558. ACM (2022), https://doi.org/10.1145/3489517.3530493

[36] Chakraborty, B., Jha, A., Nandi, M.: On the Security of Sponge-type Authenticated Encryption Modes. IACR Cryptol. ePrint Arch. p. 1475 (2019), https://eprint.iacr.org/2019/1475

[37] Chakraborty, B., Jha, A., Nandi, M.: On the Security of Sponge-type Authenticated Encryption Modes. IACR Trans. Symmetric Cryptol. 2020(2), 93–119 (2020), https://doi.org/10.13154/tosc.v2020.i2.93-119

[38] Chang, D., Turan, M.S.: Recovering the Key from the Internal State of Grain-128AEAD. IACR Cryptol. ePrint Arch. p. 439 (2021), https://eprint.iacr.org/2021/439

[39] Daemen, J., Hoffert, S., Assche, G.V., Keer, R.V.: The design of Xoodoo and Xoofff. IACR Trans. Symmetric Cryptol. 2018(4), 1–38 (2018), https://doi.org/10.13154/tosc.v2018.i4.1-38

[40] Daemen, J., Hoffert, S., Peeters, M., Assche, G.V., Keer, R.V., Mella, S.: Team Keccak: Xoodyak, https://keccak.team/xoodyak.html（2023-10-04 閲覧）

[41] Daemen, J., Hoffert, S., Peeters, M., Assche, G.V., Keer, R.V., Mella, S.: Xoodyak, a lightweight cryptographic scheme. Submission to the NIST Lightweight Cryptography project (2021)

[42] Daemen, J., Mennink, B., Assche, G.V.: Full-State Keyed Duplex with Built-In Multi-user Support. In: Takagi, T., Peyrin, T. (eds.) Advances in Cryptology - ASIACRYPT 2017 - 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part II. Lecture Notes in Computer Science, vol. 10625, pp. 606–637. Springer (2017), https://doi.org/10.1007/978-3-319-70697-9_21

[43] Delaune, S., Derbez, P., Vavrille, M.: Catching the Fastest Boomerangs Application to SKINNY. IACR Trans. Symmetric Cryptol. 2020(4), 104–129 (2020), https://doi.org/10.46586/tosc.v2020.i4.104-129

[44] Ding, L., Guan, J.: Related Key Chosen IV Attack on Grain-128a Stream Cipher. IEEE Trans. Inf. Forensics Secur. 8(5), 803–809 (2013), https://doi.org/10.1109/TIFS.2013.2256419

[45] Dinur, I., Güneysu, T., Paar, C., Shamir, A., Zimmermann, R.: An Experimentally Verified Attack on Full Grain-128 Using Dedicated Reconfigurable Hardware. In: Lee, D.H., Wang, X. (eds.) Advances in Cryptology - ASIACRYPT 2011 - 17th International Conference on the Theory and Application of Cryptology and Information Security, Seoul, South Korea, December 4-8, 2011. Proceedings. Lecture Notes in Computer Science, vol. 7073, pp. 327–343. Springer (2011), https://doi.org/10.1007/978-3-642-25385-0_18

[46] Dinur, I., Shamir, A.: Cube Attacks on Tweakable Black Box Polynomials. In: Joux, A. (ed.) Advances in Cryptology - EUROCRYPT 2009, 28th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cologne, Germany, April 26-30, 2009. Proceedings. Lecture Notes in Computer Science, vol. 5479, pp. 278–299. Springer (2009), https://doi.org/10.1007/978-3-642-01001-9_16

[47] Dinur, I., Shamir, A.: Breaking Grain-128 with Dynamic Cube Attacks. In: Joux, A. (ed.) Fast Software Encryption - 18th International Workshop, FSE 2011, Lyngby, Denmark, February 13-16, 2011, Revised Selected Papers. Lecture Notes in Computer Science, vol. 6733, pp. 167–187. Springer (2011), https://doi.org/10.1007/978-3-642-21702-9_10

[48] Dobraunig, C., Eichlseder, M., Mangard, S., Mendel, F., Mennink, B., Primas, R., Unterluggauer, T.: ISAP: Lightweight Authenticated Encryption, https://isap.iaik.tugraz.at/index.html（2023-10-04 閲覧）

[49] Dobraunig, C., Eichlseder, M., Mangard, S., Mendel, F., Mennink, B., Primas, R., Unterluggauer, T.: ISAP v2.0. Submission to the NIST Lightweight Cryptography project (2021)

[50] Dobraunig, C., Eichlseder, M., Mangard, S., Mendel, F., Unterluggauer, T.: ISAP - Towards Side-Channel Secure Authenticated Encryption. IACR Trans. Symmetric Cryptol. 2017(1), 80–105 (2017), https://doi.org/10.13154/tosc.v2017.i1.80-105

[51] Dobraunig, C., Eichlseder, M., Mendel, F., Schläffer, M.: Cryptanalysis of Ascon. In: CT-RSA. Lecture Notes in Computer Science, vol. 9048, pp. 371–387. Springer (2015)

[52] Dobraunig, C., Mennink, B.: Leakage Resilience of the Duplex Construction. In: Galbraith, S.D., Moriai, S. (eds.) Advances in Cryptology - ASIACRYPT 2019 - 25th International Conference on the Theory and Application of Cryptology and Information Security, Kobe, Japan, December 8-12, 2019, Proceedings, Part III. Lecture Notes in Computer Science, vol. 11923, pp. 225–255. Springer (2019), https://doi.org/10.1007/978-3-030-34618-8_8

[53] Dobraunig, C., Mennink, B.: Security of the Suffix Keyed Sponge. IACR Trans. Symmetric Cryptol. 2019(4), 223–248 (2019), https://doi.org/10.13154/tosc.v2019.i4.223-248

[54] Dobraunig, C., Mennink, B.: Key Recovery Attack on PHOTON-Beetle. NIST lightweight-crypto mailing list, ROUND 2 OFFICIAL COMMENT: PHOTON-Beetle (2020)

[55] Dobraunig, C., Mennink, B.: Tightness of the Suffix Keyed Sponge Bound. IACR Trans. Symmetric Cryptol. 2020(4), 195–212 (2020), https://doi.org/10.46586/tosc.v2020.i4.195-212

[56] Dong, X., Hua, J., Sun, S., Li, Z., Wang, X., Hu, L.: Meet-in-the-Middle Attacks Revisited: Key-Recovery, Collision, and Preimage Attacks. In: Malkin, T., Peikert, C. (eds.) Advances in Cryptology - CRYPTO 2021 - 41st Annual International Cryptology Conference, CRYPTO 2021, Virtual Event, August 16-20, 2021, Proceedings, Part III. Lecture Notes in Computer Science, vol. 12827, pp. 278–308. Springer (2021), https://doi.org/10.

1007/978-3-030-84252-9_10

[57] Dong, X., Qin, L., Sun, S., Wang, X.: Key Guessing Strategies for Linear Key-Schedule Algorithms in Rectangle Attacks. In: Dunkelman, O., Dziembowski, S. (eds.) Advances in Cryptology - EUROCRYPT 2022 - 41st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Trondheim, Norway, May 30 - June 3, 2022, Proceedings, Part III. Lecture Notes in Computer Science, vol. 13277, pp. 3–33. Springer (2022), https://doi.org/10.1007/978-3-031-07082-2_1

[58] Duc, A., Guo, J., Peyrin, T., Wei, L.: Unaligned rebound attack: Application to keccak. In: Canteaut, A. (ed.) Fast Software Encryption - 19th International Workshop, FSE 2012, Washington, DC, USA, March 19-21, 2012. Revised Selected Papers. Lecture Notes in Computer Science, vol. 7549, pp. 402–421. Springer (2012), https://doi.org/10.1007/978-3-642-34047-5_23

[59] Dunkelman, O., Ghosh, S., Lambooij, E.: Practical related-key forgery attacks on full-round tinyjambu-192/256. IACR Trans. Symmetric Cryptol. 2023(2), 176–188 (2023), https://doi.org/10.46586/tosc.v2023.i2.176-188

[60] Dunkelman, O., Lambooij, E., Ghosh, S.: Practical Related-Key Forgery Attacks on the Full TinyJAMBU-192/256. IACR Cryptol. ePrint Arch. p. 1122 (2022), https://eprint.iacr.org/2022/1122

[61] Dunkelman, O., Weizman, A.: Differential-Linear Cryptanalysis on Xoodyak. In: NIST LWC Workshop 2022 (2022), https://csrc.nist.gov/csrc/media/Events/2022/lightweight-cryptography-workshop-2022/documents/papers/differential-linear-cryptanalysis-on-xoodyak.pdf

[62] Elsadek, I., Aftabjahani, S., Gardner, D., MacLean, E., Wallrabenstein, J.R., Tawfik, E.Y.: Hardware and Energy Efficiency Evaluation of NIST Lightweight Cryptography Standardization Finalists. In: IEEE International Symposium on Circuits and Systems, ISCAS 2022, Austin, TX, USA, May 27 - June 1, 2022. pp. 133–137. IEEE (2022), https://doi.org/10.1109/ISCAS48785.2022.9937643

[63] Erlacher, J., Mendel, F., Eichlseder, M.: Bounds for the Security of Ascon against Differential and Linear Cryptanalysis. IACR Trans. Symmetric Cryptol. 2022(1), 64–87 (2022), https://doi.org/10.46586/tosc.v2022.i1.64-87

[64] Gérault, D., Peyrin, T., Tan, Q.Q.: Exploring differential-based distinguishers and forgeries for ASCON. IACR Cryptol. ePrint Arch. 2021, 1103 (2021), https://eprint.iacr.org/2021/1103, accepted to *IACR Trans. Symmetric Cryptol., 2021(3)*

[65] Gilbert, H., Boissier, R.H., Khati, L., Rotella, Y.: Generic attack on duplex-based AEAD modes using random function statistics. In: Hazay, C., Stam, M. (eds.) Advances in Cryptology - EUROCRYPT 2023 - 42nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Lyon, France, April 23-27, 2023, Proceedings, Part IV. Lecture Notes in Computer Science, vol. 14007, pp. 348–378. Springer (2023), https://doi.org/10.1007/978-3-031-30634-1_12

[66] Granger, R., Jovanovic, P., Mennink, B., Neves, S.: Improved Masking for Tweakable Blockciphers with Applications to Authenticated Encryption. In: Fischlin, M., Coron, J. (eds.) Advances in Cryptology - EUROCRYPT 2016 - 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part I. Lecture Notes in Computer Science, vol. 9665, pp. 263–293. Springer (2016), https://doi.org/10.1007/978-3-662-49890-3_11

[67] Guo, C., Iwata, T., Khairallah, M., Minematsu, K., Peyrin, T.: Romulus: Authenticated Encryption / Hash, https://romulusae.github.io/romulus/（2023-10-04 閲覧）

[68] Guo, C., Iwata, T., Khairallah, M., Minematsu, K., Peyrin, T.: Romulus v1.3. Submission to the NIST Lightweight Cryptography project (2021)

[69] Guo, C., Iwata, T., Khairallah, M., Minematsu, K., Peyrin, T.: Final-round updates on Romulus (2022), https://csrc.nist.gov/csrc/media/Projects/lightweight-cryptography/documents/finalist-round/status-updates/romulus-update.pdf（2023-10-04 閲覧）

[70] Guo, C., Iwata, T., Khairallah, M., Minematsu, K., Peyrin, T.: Security Proof for Romulus-T (2022), https://romulusae.github.io/romulus/docs/Romulus_T_proof.pdf（2023-10-04 閲覧）

[71] Guo, C., Pereira, O., Peters, T., Standaert, F.: Towards Low-Energy Leakage-Resistant Authenticated Encryption from the Duplex Sponge Construction. IACR Trans. Symmetric Cryptol. 2020(1), 6–42 (2020), https://doi.org/10.13154/tosc.v2020.i1.6-42

[72] Guo, J., Peyrin, T., Poschmann, A.: The PHOTON Family of Lightweight Hash Functions. In: Rogaway, P. (ed.) Advances in Cryptology - CRYPTO 2011 - 31st Annual Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2011. Proceedings. Lecture Notes in Computer Science, vol. 6841, pp. 222–239. Springer (2011), https://doi.org/10.1007/978-3-642-22792-9_13

[73] Habu, M., Minematsu, K., Iwata, T.: Matching attacks on Romulus-M. IET Inf. Secur. 16(6), 459–469 (2022), https://doi.org/10.1049/ise2.12075

[74] Hadipour, H., Bagheri, N., Song, L.: Improved Rectangle Attacks on SKINNY and CRAFT. IACR Trans. Symmetric Cryptol. 2021(2), 140–198 (2021), https://doi.org/10.46586/tosc.v2021.i2.140-198

[75] Hadipour, H., Sadeghi, S., Eichlseder, M.: Finding the Impossible: Automated Search for Full Impossible-Differential, Zero-Correlation, and Integral Attacks. In: Hazay, C., Stam, M. (eds.) Advances in Cryptology - EUROCRYPT 2023 - 42nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Lyon, France, April 23-27, 2023, Proceedings, Part IV. Lecture Notes in Computer Science, vol. 14007, pp. 128–157. Springer (2023), https://doi.org/10.1007/978-3-031-30634-1_5

[76] Hao, Y., Jiao, L., Li, C., Meier, W., Todo, Y., Wang, Q.: Links between Division Property and Other Cube Attack Variants. IACR Trans. Symmetric Cryptol. 2020(1), 363–395 (2020), https://doi.org/10.13154/tosc.v2020.i1.363-395

[77] Hao, Y., Leander, G., Meier, W., Todo, Y., Wang, Q.: Modeling for Three-Subset Division Property Without Unknown Subset - Improved Cube Attacks Against Trivium and Grain-128AEAD. In: Canteaut, A., Ishai, Y. (eds.) Advances in Cryptology - EUROCRYPT 2020 - 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12105, pp. 466–495. Springer (2020), https://doi.org/10.1007/978-3-030-45721-1_17

[78] Hao, Y., Leander, G., Meier, W., Todo, Y., Wang, Q.: Modeling for Three-Subset Division Property without Unknown Subset. J. Cryptol. 34(3), 22 (2021), https://doi.org/10.1007/s00145-021-09383-2

[79] Hell, M., Johansson, T., Maximov, A., Meier, W.: A Stream Cipher Proposal: Grain-128. In: Proceedings 2006 IEEE International Symposium on Information Theory, ISIT 2006, The Westin Seattle, Seattle, Washington, USA, July 9-14, 2006. pp. 1614–1618. IEEE (2006), https://doi.org/10.1109/ISIT.2006.261549

[80] Hell, M., Johansson, T., Maximov, A., Meier, W., Sönnerup, J., Yoshida, H.: Grain-128AEAD, https://grain-128aead.github.io/（2023-10-04 閲覧）

[81] Hell, M., Johansson, T., Maximov, A., Meier, W., Sönnerup, J., Yoshida, H.: Grain-128AEADv2 – A lightweight AEAD stream ciphe. Submission to the NIST Lightweight Cryptography project (2021)

[82] Hell, M., Johansson, T., Meier, W.: Grain – A Stream Cipher for Constrained Environments (2005), https://www.ecrypt.eu.org/stream/

[83] Hell, M., Johansson, T., Meier, W.: Grain – A Stream Cipher for Constrained Environments. Int. J. Wirel. Mob. Comput. 2(1), 86–93 (2007), https://doi.org/10.1504/IJWMC.2007.013798

[84] Hira, R., Kitahara, T., Miyahara, D., Hara-Azumi, Y., Li, Y., Sakiyama, K.: Software Evaluation for Second Round Candidates in NIST Lightweight Cryptography. IACR Cryptol. ePrint Arch. p. 591 (2022), https://eprint.iacr.org/2022/591

[85] Hirch, S.E., Mella, S., Mehrdad, A., Daemen, J.: Improved Differential and Linear Trail Bounds for ASCON. IACR Trans. Symmetric Cryptol. 2022(4), 145–178 (2022), https://doi.org/10.46586/tosc.v2022.

i4.145-178

[86] Hirose, S.: Sequential Hashing with Minimum Padding. Cryptogr. 2(2), 11 (2018), https://doi.org/10.3390/cryptography2020011

[87] Hua, J., Liu, T., Cui, Y., Qin, L., Dong, X., Cui, H.: Low-Data Cryptanalysis On SKINNY Block Cipher. Comput. J. 66(4), 970–986 (2023), https://doi.org/10.1093/comjnl/bxab208

[88] Huang, M., Xu, Z., Wang, L.: On the Probability and Automatic Search of Rotational-XOR Cryptanalysis on ARX Ciphers. Comput. J. 65(12), 3062–3080 (2022), https://doi.org/10.1093/comjnl/bxab126

[89] Inoue, A., Iwata, T., Minematsu, K.: Analyzing the Provable Security Bounds of GIFT-COFB and Photon-Beetle. In: Ateniese, G., Venturi, D. (eds.) Applied Cryptography and Network Security - 20th International Conference, ACNS 2022, Rome, Italy, June 20-23, 2022, Proceedings. Lecture Notes in Computer Science, vol. 13269, pp. 67–84. Springer (2022), https://doi.org/10.1007/978-3-031-09234-3_4

[90] ISO/IEC: Information security – Encryption algorithms – Part 7: Tweakable block ciphers (ISO/IEC 18033-7:2022), https://www.iso.org/standard/80505.html

[91] Iwata, T., Khairallah, M., Minematsu, K., Peyrin, T.: Duel of the Titans: The Romulus and Remus Families of Lightweight AEAD Algorithms. IACR Trans. Symmetric Cryptol. 2020(1), 43–120 (2020), https://doi.org/10.13154/tosc.v2020.i1.43-120

[92] Jagielski, A., Kanciak, K.: Grover on sparkle quantum resource estimation for a NIST LWC call finalist. Quantum Inf. Comput. 22(13&14), 1132–1143 (2022), https://doi.org/10.26421/QIC22.13-14-3

[93] Jana, A., Paul, G.: Differential Fault Attack on PHOTON-Beetle. In: Chang, C., Rührmair, U., Mukhopadhyay, D., Forte, D. (eds.) Proceedings of the 2022 Workshop on Attacks and Solutions in Hardware Security, ASHES 2022, Los Angeles, CA, USA, 11 November 2022. pp. 25–34. ACM (2022), https://doi.org/10.1145/3560834.3563824

[94] Janson, C., Struck, P.: Sponge-Based Authenticated Encryption: Security Against Quantum Attackers. In: Cheon, J.H., Johansson, T. (eds.) Post-Quantum Cryptography - 13th International Workshop, PQCrypto 2022, Virtual Event, September 28-30, 2022, Proceedings. Lecture Notes in Computer Science, vol. 13512, pp. 230–259. Springer (2022), https://doi.org/10.1007/978-3-031-17234-2_12

[95] Jean, J., Naya-Plasencia, M., Peyrin, T.: Improved Rebound Attack on the Finalist Grøstl. In: Canteaut, A. (ed.) Fast Software Encryption - 19th International Workshop, FSE 2012, Washington, DC, USA, March 19-21, 2012. Revised Selected Papers. Lecture Notes in Computer Science, vol. 7549, pp. 110–126. Springer (2012), https://doi.org/10.1007/978-3-642-34047-5_7

[96] Jean, J., Nikolic, I.: Internal differential boomerangs: Practical analysis of the round-reduced keccak- f f permutation. In: Leander, G. (ed.) Fast Software Encryption - 22nd International Workshop, FSE 2015, Istanbul, Turkey, March 8-11, 2015, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9054, pp. 537–556. Springer (2015), https://doi.org/10.1007/978-3-662-48116-5_26

[97] Joshi, P., Mazumdar, B.: Single Event Transient Fault Analysis of ELEPHANT cipher. CoRR abs/2106.09536 (2021), https://arxiv.org/abs/2106.09536

[98] Khairallah, M.: Security of COFB against Chosen Ciphertext Attacks. IACR Trans. Symmetric Cryptol. 2022(1), 138–157 (2022), https://doi.org/10.46586/tosc.v2022.i1.138-157

[99] Lee, J.: Security evaluation of romulus, https://romulusae.github.io/romulus/docs/Security_evaluation_Romulus_Jooyoung_Lee.pdf（2023-10-04 閲覧）

[100] Lefevre, C., Mennink, B.: Tight Preimage Resistance of the Sponge Construction. In: Dodis, Y., Shrimpton, T. (eds.) Advances in Cryptology - CRYPTO 2022 - 42nd Annual International Cryptology Conference, CRYPTO 2022, Santa Barbara, CA, USA, August 15-18, 2022, Proceedings, Part IV. Lecture Notes in Computer Science, vol. 13510, pp. 185–204. Springer (2022), https://doi.org/10.1007/978-3-031-15985-5_7

[101] Lehmann, M., Meier, W.: Conditional Differential Cryptanalysis of Grain-128a. In: Pieprzyk, J., Sadeghi, A.,

Manulis, M. (eds.) Cryptology and Network Security, 11th International Conference, CANS 2012, Darmstadt, Germany, December 12-14, 2012. Proceedings. vol. 7712, pp. 1–11. Springer (2012), https://doi.org/10.1007/978-3-642-35404-5_1

[102] Li, M., Mouha, N., Sun, L., Wang, M.: Revisiting the Extension of Matsui's Algorithm 1 to Linear Hulls: Application to TinyJAMBU. IACR Trans. Symmetric Cryptol. 2022(2), 161–200 (2022), https://doi.org/10.46586/tosc.v2022.i2.161-200

[103] Liu, F., Isobe, T., Meier, W., Yang, Z.: Algebraic Attacks on Round-Reduced Keccak/Xoodoo. IACR Cryptol. ePrint Arch. p. 346 (2020), https://eprint.iacr.org/2020/346

[104] Liu, M., Lu, X., Lin, D.: Differential-Linear Cryptanalysis from an Algebraic Perspective. In: Malkin, T., Peikert, C. (eds.) Advances in Cryptology - CRYPTO 2021 - 41st Annual International Cryptology Conference, CRYPTO 2021, Virtual Event, August 16-20, 2021, Proceedings, Part III. Lecture Notes in Computer Science, vol. 12827, pp. 247–277. Springer (2021), https://doi.org/10.1007/978-3-030-84252-9_9

[105] Liu, Y., Niu, Z., Sun, S., Li, C., Hu, L.: Rotational Differential-Linear Cryptanalysis Revisited. J. Cryptol. 36(1), 3 (2023), https://doi.org/10.1007/s00145-022-09440-4

[106] Liu, Y., Sun, S., Li, C.: Rotational Cryptanalysis from a Differential-Linear Perspective - Practical Distinguishers for Round-Reduced FRIET, Xoodoo, and Alzette. In: Canteaut, A., Standaert, F. (eds.) Advances in Cryptology - EUROCRYPT 2021 - 40th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, October 17-21, 2021, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12696, pp. 741–770. Springer (2021), https://doi.org/10.1007/978-3-030-77870-5_26

[107] Luykx, A., Preneel, B., Tischhauser, E., Yasuda, K.: A MAC Mode for Lightweight Block Ciphers. In: Peyrin, T. (ed.) Fast Software Encryption - 23rd International Conference, FSE 2016, Bochum, Germany, March 20-23, 2016, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9783, pp. 43–59. Springer (2016), https://doi.org/10.1007/978-3-662-52993-5_3

[108] Ma, Z., Tian, T., Qi, W.: Conditional differential attacks on Grain-128a stream cipher. IET Inf. Secur. 11(3), 139–145 (2017), https://doi.org/10.1049/iet-ifs.2016.0060

[109] Makarim, R.H., Rohit, R.: Towards Tight Differential Bounds of Ascon A Hybrid Usage of SMT and MILP. IACR Trans. Symmetric Cryptol. 2022(3), 303–340 (2022), https://doi.org/10.46586/tosc.v2022.i3.303-340

[110] Medwed, M., Standaert, F., Großschädl, J., Regazzoni, F.: Fresh Re-keying: Security against Side-Channel and Fault Attacks for Low-Cost Devices. In: Bernstein, D.J., Lange, T. (eds.) Progress in Cryptology - AFRICACRYPT 2010, Third International Conference on Cryptology in Africa, Stellenbosch, South Africa, May 3-6, 2010. Proceedings. Lecture Notes in Computer Science, vol. 6055, pp. 279–296. Springer (2010), https://doi.org/10.1007/978-3-642-12678-9_17

[111] Mége, A.: Official comment: PHOTON-Beetle. NIST lightweight-crypto mailing list, ROUND 2 OFFICIAL COMMENT: PHOTON-Beetle (2021)

[112] Mella, S., Daemen, J., Assche, G.V.: New techniques for trail bounds and application to differential trails in keccak. IACR Trans. Symmetric Cryptol. 2017(1), 329–357 (2017), https://doi.org/10.13154/tosc.v2017.i1.329-357

[113] Mohajerani, K., Haeussler, R., Nagpal, R., Farahmand, F., Abdulgadir, A., Kaps, J., Gaj, K.: FPGA Benchmarking of Round 2 Candidates in the NIST Lightweight Cryptography Standardization Process: Methodology, Metrics, Tools, and Results. IACR Cryptol. ePrint Arch. p. 1207 (2020), https://eprint.iacr.org/2020/1207

[114] Nageler, M., Pallua, F., Eichlseder, M.: Finding Collisions for Round-Reduced Romulus-H. IACR Trans. Symmetric Cryptol. 2023(1), 67–88 (2023), https://doi.org/10.46586/tosc.v2023.i1.67-88

[115] Naito, Y.: Optimally Indifferentiable Double-Block-Length Hashing Without Post-processing and with Support for Longer Key Than Single Block. In: Schwabe, P., Thériault, N. (eds.) Progress in Cryptology - LATINCRYPT 2019 - 6th International Conference on Cryptology and Information Security in Latin America, Santiago de Chile,

Chile, October 2-4, 2019, Proceedings. Lecture Notes in Computer Science, vol. 11774, pp. 65–85. Springer (2019), https://doi.org/10.1007/978-3-030-30530-7_4

[116] National Institute of Standards and Technology: FIPS 202 – SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions, https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.202.pdf

[117] Niu, Z., Sun, S., Liu, Y., Li, C.: Rotational Differential-Linear Distinguishers of ARX Ciphers with Arbitrary Output Linear Masks. In: Dodis, Y., Shrimpton, T. (eds.) Advances in Cryptology - CRYPTO 2022 - 42nd Annual International Cryptology Conference, CRYPTO 2022, Santa Barbara, CA, USA, August 15-18, 2022, Proceedings, Part I. Lecture Notes in Computer Science, vol. 13507, pp. 3–32. Springer (2022), https://doi.org/10.1007/978-3-031-15802-5_1

[118] Qin, L., Dong, X., Wang, X., Jia, K., Liu, Y.: Automated Search Oriented to Key Recovery on Ciphers with Linear Key Schedule Applications to Boomerangs in SKINNY and ForkSkinny. IACR Trans. Symmetric Cryptol. 2021(2), 249–291 (2021), https://doi.org/10.46586/tosc.v2021.i2.249-291

[119] Rogaway, P., Shrimpton, T.: A Provable-Security Treatment of the Key-Wrap Problem. In: Vaudenay, S. (ed.) EUROCRYPT. Lecture Notes in Computer Science, vol. 4004, pp. 373–390. Springer (2006)

[120] Saha, D., Sasaki, Y., Shi, D., Sibleyras, F., Sun, S., Zhang, Y.: On the Security Margin of TinyJAMBU with Refined Differential and Linear Cryptanalysis. IACR Trans. Symmetric Cryptol. 2020(3), 152–174 (2020), https://doi.org/10.13154/tosc.v2020.i3.152-174

[121] Schrottenloher, A., Stevens, M.: Simplified MITM Modeling for Permutations: New (Quantum) Attacks. In: Dodis, Y., Shrimpton, T. (eds.) Advances in Cryptology - CRYPTO 2022 - 42nd Annual International Cryptology Conference, CRYPTO 2022, Santa Barbara, CA, USA, August 15-18, 2022, Proceedings, Part III. Lecture Notes in Computer Science, vol. 13509, pp. 717–747. Springer (2022), https://doi.org/10.1007/978-3-031-15982-4_24

[122] Shi, T., Wu, W., Hu, B., Guan, J., Wang, S.: Breaking LWC candidates: sESTATE and Elephant in quantum setting. Des. Codes Cryptogr. 89(7), 1405–1432 (2021), https://doi.org/10.1007/s10623-021-00875-7

[123] Sibleyras, F., Sasaki, Y., Todo, Y., Hosoyamada, A., Yasuda, K.: Birthday-Bound Slide Attacks on TinyJAMBU's Keyed-Permutations for All Key Sizes. In: Cheng, C., Akiyama, M. (eds.) Advances in Information and Computer Security - 17th International Workshop on Security, IWSEC 2022, Tokyo, Japan, August 31 - September 2, 2022, Proceedings. Lecture Notes in Computer Science, vol. 13504, pp. 107–127. Springer (2022), https://doi.org/10.1007/978-3-031-15255-9_6

[124] Song, L., Zhang, N., Yang, Q., Shi, D., Zhao, J., Hu, L., Weng, J.: Optimizing Rectangle Attacks: A Unified and Generic Framework for Key Recovery. In: Agrawal, S., Lin, D. (eds.) Advances in Cryptology - ASIACRYPT 2022 - 28th International Conference on the Theory and Application of Cryptology and Information Security, Taipei, Taiwan, December 5-9, 2022, Proceedings, Part I. Lecture Notes in Computer Science, vol. 13791, pp. 410–440. Springer (2022), https://doi.org/10.1007/978-3-031-22963-3_14

[125] Sönnerup, J., Hell, M., Sönnerup, M., Khattar, R.: Efficient Hardware Implementations of Grain-128AEAD. In: Hao, F., Ruj, S., Gupta, S.S. (eds.) Progress in Cryptology - INDOCRYPT 2019 - 20th International Conference on Cryptology in India, Hyderabad, India, December 15-18, 2019, Proceedings. Lecture Notes in Computer Science, vol. 11898, pp. 495–513. Springer (2019), https://doi.org/10.1007/978-3-030-35423-7_25

[126] Speel, T.: Cryptanalysis of SPARKLE's ARX-box Alzette. Bachelor Thesis, Radboud University (2022)

[127] Sun, L., Wang, W., Wang, M.: Linear Cryptanalyses of Three AEADs with GIFT-128 as Underlying Primitives. IACR Trans. Symmetric Cryptol. 2021(2), 199–221 (2021), https://doi.org/10.46586/tosc.v2021.i2.199-221

[128] Sun, L., Wang, W., Wang, M.: Addendum to Linear Cryptanalyses of Three AEADs with GIFT-128 as Underlying Primitives. IACR Trans. Symmetric Cryptol. 2022(1), 212–219 (2022), https://doi.org/10.46586/tosc.v2022.i1.212-219

[129] Takemoto, S., Ikezaki, Y., Nozaki, Y., Yoshikawa, M.: Hardware Trojan for Lightweight Cryptoraphy Elephant. In: 10th IEEE Global Conference on Consumer Electronics, GCCE 2021, Kyoto, Japan, October 12-15, 2021. pp. 944–945. IEEE (2021), https://doi.org/10.1109/GCCE53005.2021.9622003

[130] Teng, W.L., Salam, M.I., Yau, W., Pieprzyk, J., Phan, R.C.: Cube Attacks on Round-Reduced TinyJAMBU. IACR Cryptol. ePrint Arch. p. 1164 (2021), https://eprint.iacr.org/2021/1164

[131] Todo, Y.: Structural evaluation by generalized integral property. In: Oswald, E., Fischlin, M. (eds.) Advances in Cryptology - EUROCRYPT 2015 - 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, April 26-30, 2015, Proceedings, Part I. Lecture Notes in Computer Science, vol. 9056, pp. 287–314. Springer (2015), https://doi.org/10.1007/978-3-662-46800-5_12

[132] Todo, Y., Isobe, T., Hao, Y., Meier, W.: Cube Attacks on Non-Blackbox Polynomials Based on Division Property. In: Katz, J., Shacham, H. (eds.) Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part III. Lecture Notes in Computer Science, vol. 10403, pp. 250–279. Springer (2017), https://doi.org/10.1007/978-3-319-63697-9_9

[133] Todo, Y., Isobe, T., Meier, W., Aoki, K., Zhang, B.: Fast Correlation Attack Revisited - Cryptanalysis on Full Grain-128a, Grain-128, and Grain-v1. In: Shacham, H., Boldyreva, A. (eds.) Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part II. Lecture Notes in Computer Science, vol. 10992, pp. 129–159. Springer (2018), https://doi.org/10.1007/978-3-319-96881-0_5

[134] Udvarhelyi, B., Bronchain, O., Standaert, F.: Security Analysis of Deterministic Re-keying with Masking and Shuffling: Application to ISAP. In: Bhasin, S., Santis, F.D. (eds.) Constructive Side-Channel Analysis and Secure Design - 12th International Workshop, COSADE 2021, Lugano, Switzerland, October 25-27, 2021, Proceedings. Lecture Notes in Computer Science, vol. 12910, pp. 168–183. Springer (2021), https://doi.org/10.1007/978-3-030-89915-8_8

[135] Vialar, L.: Fast Side-Channel Key-Recovery Attack against Elephant Dumbo. IACR Cryptol. ePrint Arch. p. 446 (2022), https://eprint.iacr.org/2022/446

[136] Wang, Q., Grassi, L., Rechberger, C.: Zero-Sum Partitions of PHOTON Permutations. IACR Cryptol. ePrint Arch. 2017, 1211 (2017), https://eprint.iacr.org/2017/1211

[137] Wang, Q., Grassi, L., Rechberger, C.: Zero-Sum Partitions of PHOTON Permutations. In: Smart, N.P. (ed.) Topics in Cryptology - CT-RSA 2018 - The Cryptographers' Track at the RSA Conference 2018, San Francisco, CA, USA, April 16-20, 2018, Proceedings. Lecture Notes in Computer Science, vol. 10808, pp. 279–299. Springer (2018), https://doi.org/10.1007/978-3-319-76953-0_15

[138] Wang, Q., Hao, Y., Todo, Y., Li, C., Isobe, T., Meier, W.: Improved Division Property Based Cube Attacks Exploiting Algebraic Properties of Superpoly. In: Shacham, H., Boldyreva, A. (eds.) Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part I. Lecture Notes in Computer Science, vol. 10991, pp. 275–305. Springer (2018), https://doi.org/10.1007/978-3-319-96884-1_10

[139] Wang, S., Hu, B., Guan, J., Zhang, K., Shi, T.: MILP-aided Method of Searching Division Property Using Three Subsets and Applications. In: Galbraith, S.D., Moriai, S. (eds.) Advances in Cryptology - ASIACRYPT 2019 - 25th International Conference on the Theory and Application of Cryptology and Information Security, Kobe, Japan, December 8-12, 2019, Proceedings, Part III. Lecture Notes in Computer Science, vol. 11923, pp. 398–427. Springer (2019), https://doi.org/10.1007/978-3-030-34618-8_14

[140] Wu, H., Huang, T.: CAESAR candidates AEGIS + Jambu. DIAC - Directions in Authenticated Ciphers (2014), https://2014.diac.cr.yp.to/

[141] Wu, H., Huang, T.: TinyJAMBU: A Family of Lightweight Authenticated Encryption Algorithms (Version 2). Submission to the NIST Lightweight Cryptography project (2021)

[142] Xu, Z., Li, Y., Jiao, L., Wang, M., Meier, W.: Do NOT Misuse the Markov Cipher Assumption - Automatic Search for Differential and Impossible Differential Characteristics in ARX Ciphers. IACR Cryptol. ePrint Arch. p. 135 (2022), https://eprint.iacr.org/2022/135

[143] Yang, Y., Jang, K., Kim, H., Song, G., Seo, H.: Grover on SPARKLE. In: You, I., Youn, T. (eds.) Information Security Applications - 23rd International Conference, WISA 2022, Jeju Island, South Korea, August 24-26, 2022, Revised Selected Papers. Lecture Notes in Computer Science, vol. 13720, pp. 44–59. Springer (2022), https://doi.org/10.1007/978-3-031-25659-2_4

[144] Zhang, G., Liu, M.: A distinguisher on PRESENT-like permutations with application to SPONGENT. Sci. China Inf. Sci. 60(7), 72101 (2017), https://doi.org/10.1007/s11432-016-0165-6

[145] Zhang, X., Wang, T., Cao, P.: Side-Channel Evaluation on Protected Implementations of Several NIST LWC Finalists (2022), https://cryptography.gmu.edu/athena/LWC/Reports/SJTU/SJTU_Report_HW_4_candidates_RUB.pdf（2023-10-04 閲覧）

[146] Zhou, H., Li, Z., Dong, X., Jia, K., Meier, W.: Practical Key-Recovery Attacks On Round-Reduced Ketje Jr, Xoodoo-AE And Xoodyak. Comput. J. 63(8), 1231–1246 (2020), https://doi.org/10.1093/comjnl/bxz152

[147] Zhou, H., Zong, R., Dong, X., Jia, K., Meier, W.: Interpolation Attacks on Round-Reduced Elephant, Kravatte and Xoofff. Comput. J. 64(4), 628–638 (2021), https://doi.org/10.1093/comjnl/bxaa101

[148] Zong, R., Dong, X., Chen, H., Luo, Y., Wang, S., Li, Z.: Towards Key-recovery-attack Friendly Distinguishers: Application to GIFT-128. IACR Trans. Symmetric Cryptol. 2021(1), 156–184 (2021), https://doi.org/10.46586/tosc.v2021.i1.156-184

[149] 井上明子: 軽量暗号の安全性に関する調査及び評価（Elephant,ISAP,Romulus）（文書番号：CRYPTREC EX-3204-2022）(2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3204-2022.pdf

[150] 岡部忠: 軽量ストリーム暗号のハードウェア実装 〜 FPGA を対象デバイスとした実装性能の比較 〜. In: 情報処理学会講演論文集. p. 2 (2022)

[151] 岩田哲: 軽量暗号の安全性に関する調査及び評価（Photon-Beetle,Sparkle,Tsudik's keymode）（文書番号：CRYPTREC EX-3201-2022）(2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3201-2022.pdf

[152] 崎山一男: 軽量暗号の実装性能に関する調査及び評価（NIST 軽量暗号コンペティションファイナリスト）（文書番号：CRYPTREC EX-3205-2022）(2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3205-2022.pdf

[153] 土生亮, 岩田哲: Elephant に対する鍵回復, 識別及び偽造攻撃. In: 暗号と情報セキュリティシンポジウム, SCIS2022, 1F2-5 (2022)

[154] 土生亮, 峯松一彦, 岩田哲: Romulus-N 及び Romulus-M に対する識別攻撃及び偽造攻撃. 信学技報 121(22, ISEC2021-6), 25–31 (2022)

[155] 藤堂洋介: 軽量暗号の安全性に関する調査及び評価（Ascon,Grain-128AEAD,TinyJambu）（文書番号：CRYPTREC EX-3203-2022）(2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3203-2022.pdf

[156] 内藤祐介: 軽量暗号の安全性に関する調査及び評価（GIFT-COFB,Xoodyak）（文書番号：CRYPTREC EX-3202-2022）(2022), https://www.cryptrec.go.jp/exreport/cryptrec-ex-3202-2022.pdf

CRYPTREC 暗号技術ガイドライン（軽量暗号）2023 年度版
[CRYPTREC GL-2006-2023]

発行日：2024 年 3 月 31 日（第 1 版）
発行者

- 〒184-8795
  東京都小金井市貫井北町四丁目２番１号
  国立研究開発法人情報通信研究機構
  （サイバーセキュリティ研究所　セキュリティ基盤研究室）
  NATIONAL INSTITUTE OF INFORMATION AND COMMUNICATIONS TECHNOLOGY
  4-2-1 NUKUI-KITAMACHI, KOGANEI
  TOKYO, 184-8795 JAPAN
- 〒113-6591
  東京都文京区本駒込二丁目２８番８号
  独立行政法人情報処理推進機構
  （セキュリティセンター セキュリティ技術評価部 暗号グループ）
  INFORMATION-TECHNOLOGY PROMOTION AGENCY, JAPAN
  2-28-8 HONKOMAGOME, BUNKYO-KU
  TOKYO, 113-6591 JAPAN