L-008

AESのハードウェア実装に対するテンプレート攻撃
Template Attack on AES Hardware Implementations

金 用大[†] 菅原 健[†] 本間 尚文[†] 青木 孝文[†] 佐藤 証[‡]
Yongdae Kim Takeshi Sugawara Naofumi Homma Takafumi Aoki Akashi Satoh

1. まえがき

近年，暗号を実装したモジュール（暗号モジュール）の物理的な挙動（サイドチャネル情報）を計測することで，暗号解読を行う攻撃手法 (サイドチャネル攻撃) の危険性が指摘されている．たとえアルゴリズムが計算量的に安全であったとしても，サイドチャネル攻撃により演算過程が観測されることで解読される可能性がある．研究レベルでは，攻撃の成功例が多く発表されており [1]–[3]，民生品への攻撃も現実的な脅威とみなされている．

サイドチャネル攻撃の一つである，電力解析攻撃 [1] は，暗号モジュールの消費電力から秘密情報を抽出する攻撃である．比較的安価な計測装置（オシロスコープ等）で実行可能であり，攻撃の痕跡も残らないことから，特に重要な攻撃と位置づけられている．文献 [1] では，基本的な解析手法である単純電力解析および差分電力解析が示されている．これを契機として，これまで多くの攻撃法および防御法の研究がなされている．その中でも，テンプレート攻撃 [4] は最も強力な手法の１つとして知られている．テンプレート攻撃の特徴は，攻撃者が自由に操作・観測できる参照モジュールを用い，攻撃対象モジュールの情報収集（プロファイリング）を事前に行うことにある．プロファイリングによって得た知識を用いることで，実際の攻撃時は対象モジュールを少ない計測回数で効率的に解読できる．

これまで，テンプレート攻撃についても広く研究が行われており，多くの成果が発表されている [4]–[9]．

しかし，これらの多くは，暗号アルゴリズムの実装形態として 8 ビットマイクロコントローラを対象にするものである．これに対し，専用のハードウェア実装へ適用した事例は少ない [7]–[9]．これは，専用回路の持つ高い並列性により，マイクロプロセッサと同様の手法を適用できないためである．ハードウェア実装を対象とした既存研究 [7]–[9] の結果も，暗号アルゴリズムのサブセットを対象とした基礎実験であり，暗号アルゴリズム全体を実装したハードウェアへの攻撃および全鍵空間のテンプレート攻撃は示されていない．

本稿では，並列性の高い専用ハードウェア実装に対しても有効なテンプレート攻撃を提案する．テンプレート攻撃をハードウェア実装向けに拡張する手法を述べるとともに，FPGA (Field Programmable Gate Array) 実装を用いた実験によりその有効性を示す．実験では，専用基板 SASEBO[10] 上の FPGA に AES(Advanced Encryption Standard [NIST]) の暗号コアを計測対象とした．また，計測した電力波形の解析では，比較のため，上述のテンプレート攻撃に加えて，従来法である CPA(Correlation Power Analysis) を適用した．以上の比較実験から，専用ハードウェア実装に対してもテンプレート攻撃が成功すること，従来の CPA よりも高い攻撃能力を有することを示す．

[†]東北大学大学院情報科学研究科
〒 980-8579 宮城県仙台市青葉区荒巻字青葉 6-6-05
kimyd, sugawara, homma@aoki.ecei.tohoku.ac.jp,
aoki@aoki.ecei.tohoku.ac.jp
[‡]独立行政法人産業技術総合研究所 情報セキュリティ研究センター
〒 101-0021 東京都千代田区外神田 1-18-13
akashi.satoh@aist.go.jp

2. テンプレート攻撃

本章では，テンプレート攻撃のアルゴリズムを概説し，ハードウェア実装に適用する場合の問題点について述べる．

2.1 テンプレート攻撃の概要

テンプレート攻撃は、(i) プロファイリングフェーズおよび (ii) 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，攻撃者が攻撃対象と同種類のモジュール（参照モジュール M）を用いて，対象となるモジュールの特性を求める．その後，攻撃フェーズで，攻撃対象（入力等を操作できない）モジュール M' に対して鍵推定を行う．その際，プロファイリングフェーズで得た事前知識を用いることで，少ない観測での効率的な攻撃が可能となる．自由に操作可能な参照モジュールを必要とするため，攻撃が可能となる条件は単純電力解析や差分電力解析と比べて厳しいと言える．しかし，製品化される暗号モジュールは通常量産されるため，参照モジュールの入手は現実的に可能である場合が多い．

2.2 プロファイリングフェーズ

プロファイリングフェーズでは，計測する電力波形を，確率分布によりモデル化することでテンプレートを作成する．

テンプレート攻撃では，波形 $\mathbf{w}$ が暗号処理の中間値 x に応じて変化することを想定する．この際，$\mathbf{w}$ を確率変数とみなし，その確率分布を求めることを考える．そのため，中間値 x のとりうる全ての場合について確率密度関数 $p(\mathbf{w}|x)(x \in x^*)$ を考える．ここで，x^* は，中間値 x の取り得る全ての値の集合である．そのため，テンプレート数は $|x^*|$ となる．いま，計測波形 $\mathbf{w}$ が多変量ガウス分布:

$$\mathbf{w} \sim \frac{exp(-\frac{1}{2}(\mathbf{w}-\mathbf{m})^T\mathbf{C}^{-1}(\mathbf{w}-\mathbf{m}))}{\sqrt{(2\pi)^L det(\mathbf{C})}}$$

に従うことを仮定した場合，確率密度関数は平均 $\mathbf{m}$ と共分散行列 $\mathbf{C}$ により特徴付けられる．そこで，$\mathbf{m}$ と $\mathbf{C}$ のペアをテンプレート T_x とする．

$$T_x = (\mathbf{m}, \mathbf{C})_x \quad (1)$$

プロファイリングフェーズでは，操作可能な参照モジュールから任意の波形を取得し，上述の $(\mathbf{m}, \mathbf{C})_x$ を求める．具体的には，平文 PT_i に対して取得した時刻 t の計測波形 $w_{i,t}$ に対し，

$$m_t = \frac{1}{D}\sum_{i=1}^{D} w_{i,t} \quad (2)$$

$$C_{t_1,t_2} = \sum_{i=1}^{D} \frac{(w_{i,t_1} - m_{t_1})(w_{i,t_2} - m_{t_2})}{D} \quad (3)$$

を計算する．ここで，i, t_1, t_2 は，$1 \leq i \leq D, 1 \leq t_1, t_2 \leq L$ であり，D と L はそれぞれ波形数と一つの波形で取得した点数 (サンプル数) である．

2.3 攻撃フェーズ

攻撃フェーズでは，対象モジュールの電力波形から，その中間値 x の推定を行う．この際，プロファイリングフェーズで得たテンプレートを用いて最尤推定法を用いる．

尤度には，条件付き確率 $p(x|\mathbf{w})$ を用いる．この値は，ベイズの定理より，

$$p(x|\mathbf{w}) = \frac{p(\mathbf{w}|x) \cdot p(x)}{\sum_{x=1}^{|x^*|}(p(\mathbf{w}|x) \cdot p(x))} \quad (4)$$

と与えられる．上述のとおり，$p(\mathbf{w}|x) = p(\mathbf{w}; (\mathbf{m}, \mathbf{C})_x)$ であるため，$p(\mathbf{w}|x)$ は，

$$p(\mathbf{w}|x) = p(\mathbf{w}; (\mathbf{m}, \mathbf{C})_x) \quad (5)$$

$$= \frac{exp(-\frac{1}{2}(\mathbf{w}-\mathbf{m})^T\mathbf{C}^{-1}(\mathbf{w}-\mathbf{m}))}{\sqrt{(2\pi)^L det(\mathbf{C})}} \quad (6)$$

となる．この結果，$p(x|\mathbf{w})$ が計算され，

$$x_c = \underset{x \in x^*}{\operatorname{argmax}}\, p(x|\mathbf{w})$$

となる x_c が，中間値として推定される．

実際の攻撃では，中間値 x を秘密鍵もしくは鍵に関連した量に設定する．この結果，テンプレート攻撃により，秘密情報が抽出可能となる．

2.4 ハードウェア実装への従来法の適用

前節で述べたように，テンプレートは，中間変数 x のとりうる値の個数 $|x^*|$ 作る必要がある．データパス幅が 8 ビットのマイクロコントローラを考えた場合，その個数は $|x^*| = 2^8 = 256$ となり，十分に作成可能である．また，文献 [11] では，中間値そのものではなく，そのハミングディスタンスについてテンプレートを作成することで，その数を削減している．その場合，8 ビットのマイクロコントローラでは，8 ビットに対応する 9 種類のハミングディスタンス 0 ～ 8 を考えればよい．すなわち $|x^*| = 9$ となる．ただし，この場合に得られるのは，秘密鍵と等しいハミングディスタンス値をもつ鍵候補であることに注意されたい．

一方，高い並列性を有するハードウェア実装では，中間変数 x の取り得る範囲は非常に大きいものとなる．例えば，鍵長 128 ビットの暗号アルゴリズムに対して，128 ビット幅のデータパス回路を設計することは現実的であるが，従来のテンプレート攻撃では，テンプレートの個数が $|x^*| = 2^{128}$ となり，その作成は困難となる．

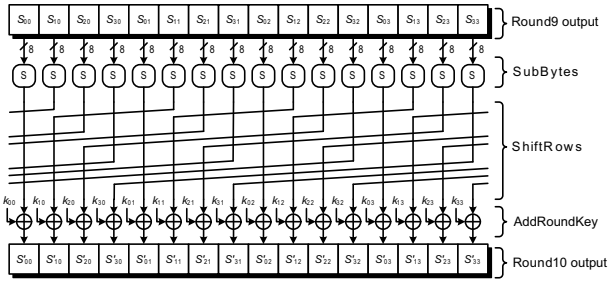


図 1: AES 暗号アルゴリズムの最後ラウンドのデータパス

ハミングディスタンスを用いてテンプレートを作成する場合，$|x^*| = 129$ 個のテンプレートは作成可能であるものの，(i) 攻撃フェーズで 2^{128} の探索が必要もしくは (ii) 得たハミングディスタンスから鍵を導出するのが困難であり，こちらも現実的には難しい．

このような背景から，これまで暗号アルゴリズムの完全なハードウェア実装に対してテンプレート攻撃に成功した例はなかった．既存の成果には，とりうる鍵空間に制限をかけているもの [8]，もしくは，暗号アルゴリズムの一部である S-Box に対して攻撃を行っているもの [7] があるのみである．

3. ハードウェア実装へのテンプレート攻撃

本章では，高い並列性を有するハードウェア実装に対しても適用可能なテンプレート攻撃を提案する．ここでは，従来困難だった鍵長分の並列性を有する実装形態を想定する．提案手法では，上述した問題を解決するため，(i) テンプレート数および (ii) 共分散行列サイズの観点から攻撃に要する計算量を削減する．また，この削減に伴い攻撃フェーズを拡張する．

3.1 テンプレート数の削減

提案手法では，テンプレート数を削減するため，データパス幅のうち所望のデータ幅にのみ注目し，残りはノイズ源として扱う．このように，注目していないデータパス幅をノイズ源と見なすことは，差分電力解析において導入されており [12]，注目していないデータによるノイズ成分は，アルゴリズミックノイズと呼ばれる．以下では，128 ビットのデータパスを有する鍵長 128 ビットの AES 回路に対して，8 ビット分の鍵推定を行う攻撃を例に説明する．このとき，注目していない残りの 120 ビットがアルゴリズミックノイズとなる．

図 1 に最後ラウンドのデータパスを示す．暗文が既知であれば，AddRoundKey で XOR 加算される最後ラウンド鍵の一部を 8 ビットごとに予測することで，各予測の鍵に対して，ハミングディスタンス値を計算することが可能である．8 ビットの部分鍵を攻撃対象にしているため，ハミングディスタンスの値は 0 から 8 となる．各ハミングディスタンス値に対する波形を大量に集めて，平均と共分散行列を計算すると，ハミングディスタンス値に応じて全 9 個のテンプレートを作ることができる．

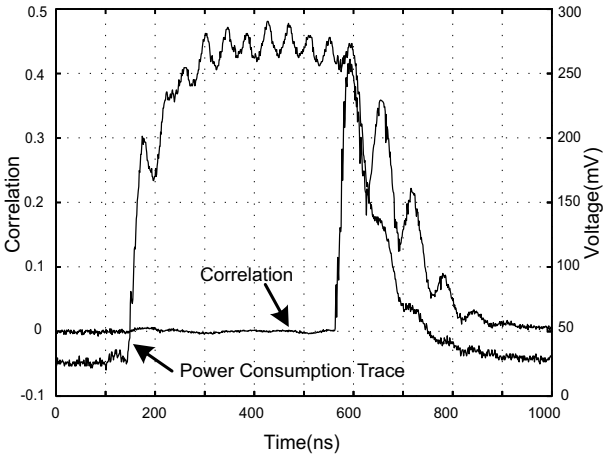


図 2: 128 ビット長のハミングディスタンス値の相関値と実測した AES の FPGA 消費電力波形

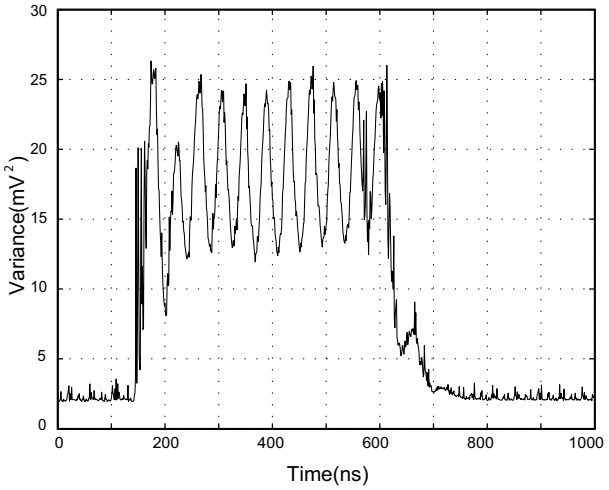


図 3: 消費電力の分散

3.2 共分散行列サイズの削減

提案手法は，テンプレート数と並んで計算量に影響を与える共分散行列のサイズも削減する．共分散行列の大きさはテンプレートで使用する時間軸上の点（*interesting points*）の数で決定される．*interesting points* を求める手法は広く議論されており，現在まで主成分分析を用いる方法 [8] や消費電力の平均を求め平均の差が大きい値をとる方法 [13] などが提案されている．

本手法では，たとえデータパス幅が広かったとしても，プロファイリングフェーズにおいては，その全体のハミングディスタンスを求めることができるという事実に基づく．これは，プロファイリングフェーズにおいて鍵が既知であるためである．本稿では，以下の手順に従い *interesting points* を選択する．

1. ハミングディスタンス値の計算
   参照モジュールを用いて，攻撃対象の最後ラウンドに対して任意の鍵をセットして波形を取得する．それと並行して，鍵長（全語長）の処理が並列に動作している場合，全語長のハミングディスタンス値を求める．このハミングディスタンス値は消費電力と線形な関係が成り立ち，かつ暗号ハードウェアまたは実験環境からのノイズが全く存在しない状態を仮定した消費電力モデルとなる．

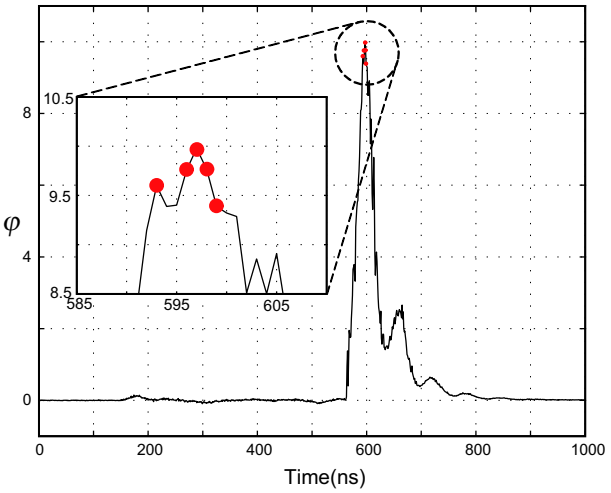


図 4: *interesting points*

2. 相関値の計算
   上記で求めたハミングディスタンス値 h_i と時刻 t の波形 $w_{i,t}$ とのピアソン相関係数 ρ_t を以下の式で求める．

$$\rho_t = \frac{\sum_{i=1}^{D}(w_{i,t} - \bar{w_t})(h_i - \bar{h})}{\sqrt{\sum_{i=1}^{D}(w_{i,t} - \bar{w_t})^2 (h_i - \bar{h})^2}} \qquad (7)$$

   ここで，h_i は最大で語長分の値をとるハミングディスタンス値であり，D は波形数である（$1 \leq i \leq D$）．また，時刻 t の波形（電圧）の平均値 $\bar{w_t}$ およびハミングディスタンスの平均値 $\bar{h}$ はそれぞれ $\bar{w_t} = \frac{1}{D}\sum_{i=1}^{D} w_{i,t}$，$\bar{h} = \frac{1}{D}\sum_{i=1}^{D} h_i$ と与えられる．例として，128 ビット長のデータパスを有する AES 回路の電力波形 100,000 を用いて相関係数を求めた結果を図 2 に示す．この相関値が高いことが *interesting points* 選択の基本となる．

3. 分散値の計算
   上記の相関値に加えて，平文の消費電力への影響を考慮するため，以下の分散値を求める．

$$v_t = \frac{1}{D}\sum_{i=1}^{D}(w_{i,t} - \bar{w_t})^2 \qquad (8)$$

   図 2 で用いた 100,000 の波形に対して上式で求めた分散をプロットしたものを図 3 に示す．ここで，分散の高い点は，平文に応じて消費電力が大きく変化することを示す．

4. *interesting points* の選択

相関値と分散を考慮した以下の式からある時刻 t における評価量 φ_t を求める．これは相関が高く，かつその変動が大きい点を探すという観点から決めたものである．

$$\varphi_t = \rho_t v_t \tag{9}$$

提案手法では，φ_t 値が上位の点を *interesting points* とし，ハミングディスタンス値からテンプレートを作成する．図 4 に φ_t の高い値から 5 点を *interesting points* とした結果を示す．

上記の例で使用した 100,000 の波形に対して提案手法で求めた *interesting points* におけるハミングディスタンスと平均消費電力の関係を図 5(a) に示す．また，参考のため，φ_t の低い 5 点を *interesting points* としたときの場合を図 5(b) に示す．図 5 より，提案手法で選択した (a) の *interesting points* は (b) と比べて，ハミングディスタンス値と波形（電圧）値に相関があることを確認できる．

3.3 攻撃フェーズの拡張

攻撃対象のモジュールが平文 PT_i を暗号化するときに生じる消費電力波形を $\mathbf{t}_i$ とする．このとき，攻撃フェーズでは，その波形が鍵 k_j（$\in k^* = \{0 \leq k \leq 2^s - 1\}$，$s$ は一度に推定する鍵の語長）を用いて取得した波形である確率をベイズ定理から求めることになる．しかし，提案手法は，全鍵空間分のテンプレートをもたず，また，アルゴリズミックノイズの影響のため，単一の波形 $\mathbf{t}_i$ から鍵を正しく推定することはできない．そこで，複数の波形 $(\mathbf{t}_1, ..., \mathbf{t}_D)$ を用いて，確率 $p(k_j|\mathbf{t}_1, ..., \mathbf{t}_D)$ を以下のように計算する．

$$p(k_j|\mathbf{t}_1, ..., \mathbf{t}_D) = \frac{(\Pi_{i=1}^{D} p(\mathbf{t}_i|k_j)) \cdot p(k_j)}{\sum_{l=1}^{K}((\Pi_{i=1}^{D} p(\mathbf{t}_i|k_l)) \cdot p(k_l))} \tag{10}$$

ここで，K は鍵候補の数であり，128 ビットデータパスを有する AES 暗号回路の鍵を 8 ビットごとに推定する場合，$K = 256$ となる．また，$p(k_j)$ は鍵が k_j である確率であり，

$$\sum_{j=1}^{K} p(k_j|\mathbf{t}_i) = \sum_{j=1}^{K} p(k_j) = 1$$

と与えられる．各鍵の確率が一様分布とすると，$p(k_j) = 1/K$ となる．また，確率 $p(\mathbf{t}_i|k_j)$ は

$$p(\mathbf{t}_i|k_j) = p(\mathbf{t}_i; (\mathbf{m}, \mathbf{C})_h)$$

である．ただし，$(\mathbf{m}, \mathbf{C})_h$ は平文 PT_i と鍵 k_j から求めたハミングディスタンス h のテンプレートである．以上から，最尤推定を用いて，全鍵空間 k^* に対して，以下の式を満たす $k_c \in k*$ を探すことで鍵を推定する．

$$k_c = \underset{k \in k^*}{\operatorname{argmax}}\, p(k|\mathbf{t}_1, ..., \mathbf{t}_D) \tag{11}$$

なお，計算能力が許す限り，語長 s を増やすことができる．

4. 実験

本章では，上述の手法の有効性を評価するため，AES のハードウェア実装に対するテンプレート攻撃実験を示す．まず，攻撃対象とする AES の回路構造を示す．その上で，サイドチャネル評価基板を用いた実験環境について述べ，テンプレート攻撃の実験結果を示す．

4.1 AES のハードウェア実装

AES 暗号アルゴリズムの概略を図 6(a) に示す．データ攪拌部は，SPN (Substitution Permutaion Network) 構造が用いられており，鍵長 128 ビットのとき 10 ラウンドの処理から構成される．1 ラウンドは，4 種類のサブ関数 (i) AddRoundKey, (ii) SubBytes, (iii) ShiftRows, および (iv)MixColumns から構成される．それぞれのサブ関数は，128 ビットのデータブロックをステートと呼ばれる 4×4 バイトの行列と解釈し，その変換を行う．ここで，AddRoundKey はラウンド鍵とステートの排他的論理和，SubBytes は S-Box と呼ばれる置換テーブルによる非線形置換，ShiftRows はステートの各行に対する巡回シフト，MixColumns は行列演算を行う．ラウンド 1 からラウンド 9 までは 4 種類の処理が行われ，最後のラウンドだけ MixColumns 以外の 3 種類の処理が行われる．KeySchedule は各ラウンドで使用されるラウンド鍵を生成する．AES の詳細な処理については文献 [14] を参照されたい．

本稿で扱う AES コアのデータパスアーキテクチャを図 6(b) に示す．ここでは鍵長 128 ビットを想定している．図において左半分が 1 段文のラウンド関数ブロックを持つ攪拌部であり，右半分が鍵スケジュール部である．データ攪拌部では，4 つのサブ関数に対応する機能が組み合わせ回路で実現されている．データパス幅は 128 ビットであり，上述した AES の 1 ラウンドの処理を，1 クロックサイクルで実行する．AES は非常に多様なハードウェア実装が可能であるが，上述の実装はその中でも最も一般的な構造の 1 つである．

4.2 実験環境

本実験では，AES の実装にサイドチャネル標準評価ボード SASEBO[10] の FPGA を用いた．図 7 に実験システム外観とブロック図を示す．測定機器の詳細を表 1 に示す．AES の暗号コアには，S-box を合成体 $GF(((2^2)^2)^2)$ 上の演算器により実装したものを使用し，秘密鍵の値は，$(0x8b6afe6ae2687781933178365636d9f)_{128}$ とした．また，波形は，図 7(b) のように FPGA の GND 線上に挿入された抵抗部分から計測した．

本実験では，簡単のため，以上のシステムにより計測した波形をプロファイリングと攻撃の両フェーズで用いてテンプレート攻撃を実施した．本来のテンプレート攻撃では参照モジュールと攻撃対象モジュールが異なることに注意されたい．具体的には，平文の異なる 205,000 の波形を取得し，そのうち 200,000 枚の波形をプロファイリングフェーズ，残りの 5,000 枚の波形を攻撃フェーズで用いた．なお，プロファイリング時のテンプレート作成では 5 点を interesting point として使用した．

また，従来法として CPA[3] を同一の波形に対して実施し，その結果をテンプレート攻撃の結果と比較した．

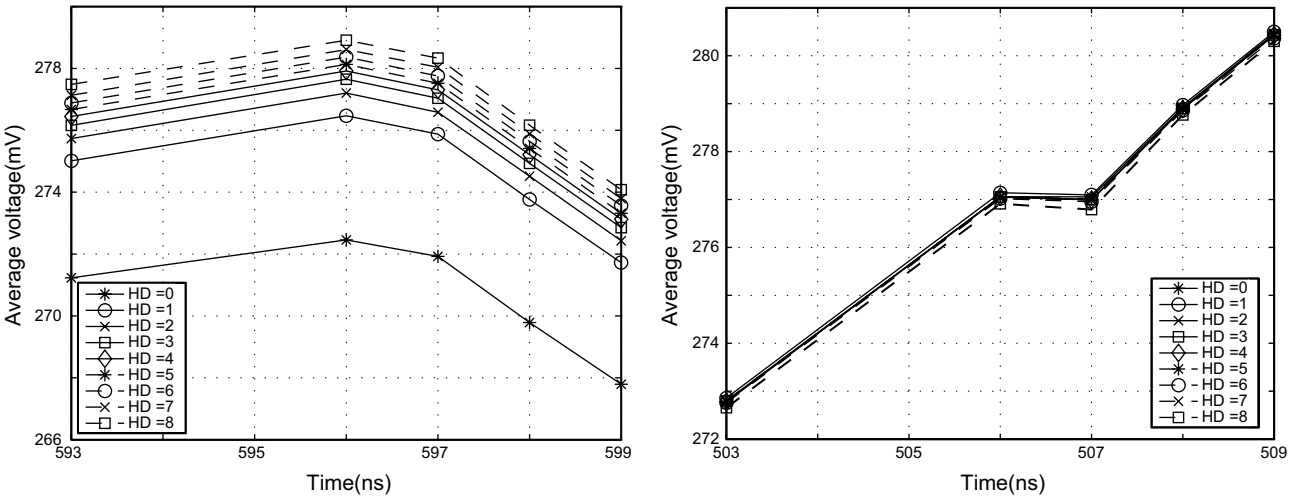


図 5: ハミングディスタンス値と平均消費電力の関係 (a) 高い φ_t を持つ 5 点 (b) 低い φ_t を持つ 5 点

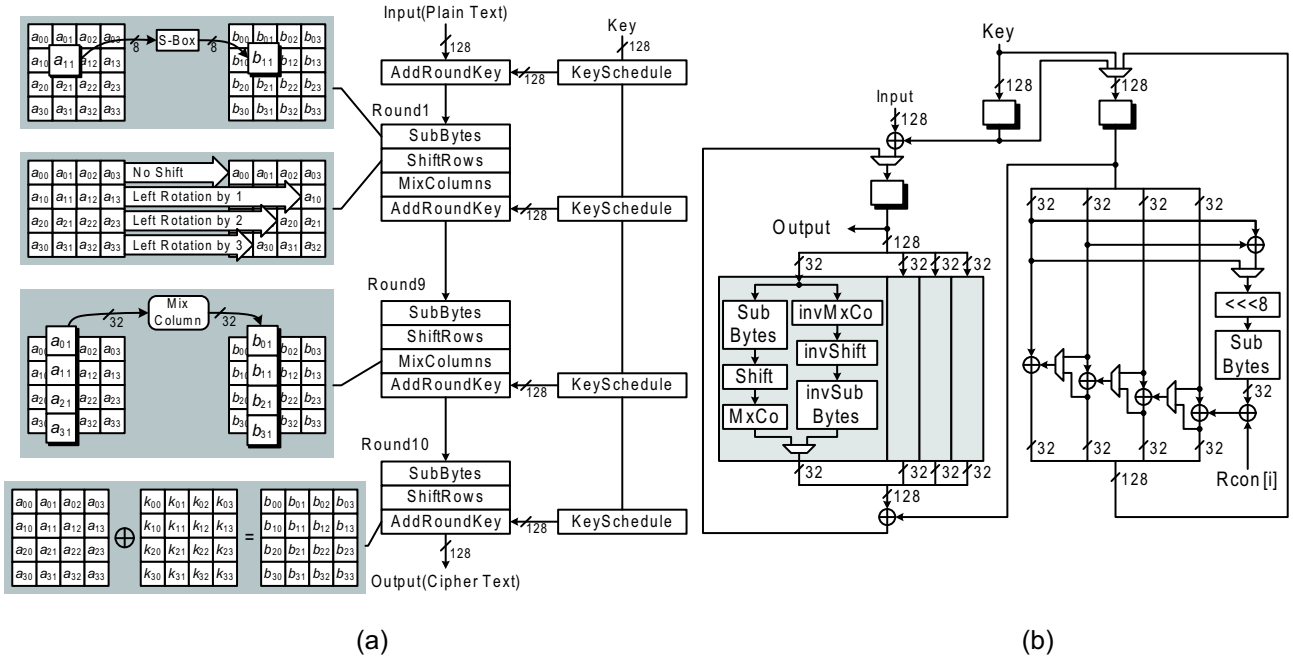


図 6: AES 暗号 (a)AES の暗号化処理 (b) 実装したデータパスアーキテクチャ

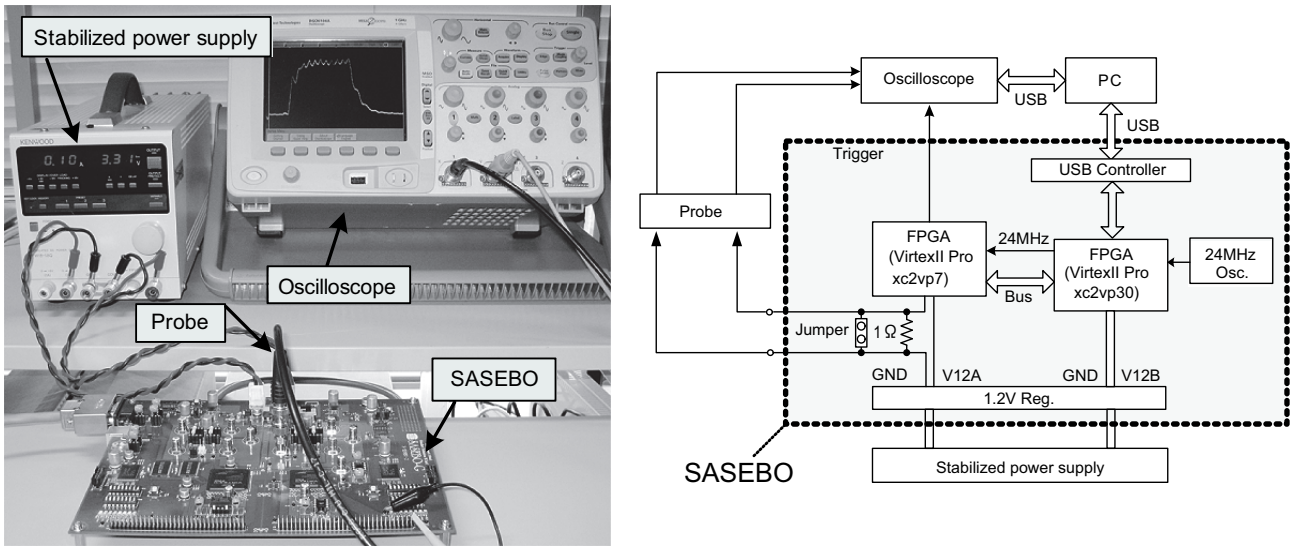
CPA では，テンプレート攻撃と同様に，図 1 に示す AES の最終ラウンドを対象とした．この時，レジスタのバイトごとにハミングディスタンスモデルに基づいて消費電力の予測値を作成した．また，上述の予測値と計測波形間の相関をピアソンの相関係数により鍵の推定を行った．

4.3 実験結果

図 8 にテンプレート攻撃と CPA の結果を示す．ここで，横軸は解析に用いた波形数であり，縦軸は各攻撃法で鍵推定に影響を与える値である．二つの攻撃はいずれも 128 ビットの鍵を 8 ビットごと独立に推定する必要があるが，図 8 は，例として，図 1 に示すステート S_{33} (レジスタの最下位 8 ビット) に対する結果を示す．

図 8(a) はテンプレート攻撃による結果であり，縦軸は式 10 に示す確率 $p(k_j|\mathbf{t})$ である．$p(k_j|\mathbf{t})$ は，$256(=2^8)$ 通りの鍵候補に対応して 256 種類計算される．図中では，その全てを重ねて表示しており，正しい推定のみを黒，その他を灰色で示している．一方，図 8(b) は CPA による結果である．縦軸は，ピアソンの相関係数の値である．図 8(b) と同様に，正しい推定を黒，その他を灰色として，重ねて表示している．

図 8(a) より，正しい推定による確率 $p(k_j|\mathbf{t})$ が，波



(a) (b)

図 7: 実験 (a) 実験環境の概観 (b) 環境環境のブロック図

Experimental FPGA Board (SASEBO)	
FPGA	Virtex-II Pro xc2vp7
Experimental Setup	
Digital Oscilloscope	Agilent MSO6104A
Sampling Frequency	1GSa/s
Probe	Coaxial cable (50-Ohm)
Stabilized power supply	3.3V
Crystal oscillator	24-MHz
Measuring point	Resistor(1-Ohm)

表 1: 計測条件

形数の増加とともに，1 へ漸近する様子が観察される．また, おのおのの $p(k_j|\mathbf{t})$ が波形数に応じて変化する様子を図 9 に示す．縦軸軸は $p(k_j|\mathbf{t})$，横軸は k_j である．これに伴い，誤った推定による確率はゼロに収束している．これは，テンプレート攻撃が成功し，鍵が抽出されたことを表す．同様に，図 8(b) でも，使用する波形数の増加に伴い，正しい推定による相関値と，その他による相関値が分離しており，攻撃が成功したことが分かる．

このように，図 8(a) と (b) の両方とも，黒線で表示した正しい推定による値がその他と比較して高くなったとき鍵の推定に成功したと判断できる．そのため，黒線がその他の灰色の線とクロスする点 (MTD: Measurement to Disclosure) により攻撃能力を評価できる．それぞれの MTD を比較すると，テンプレート攻撃で 580，CPA で 2700 となる．この結果は，テンプレート攻撃が CPA と比較して少ない波形で攻撃に成功したことを示している．

上述の通り，図 8 は，128 ビット鍵のうち，8 ビットのみの結果である．全て（16(=128/8) 個）の部分鍵の推定に対する結果を図 10 に示す．ここで，横軸は図 8 と同様に解析に用いた波形数であり，縦軸は 16 個の部分鍵のうち，推定に失敗したバイト数（エラーレート）である．そのため，少ない波形数で，エラーレートが小さいほど，鍵の推定が速く，攻撃能力が高いといえる．図 10 より，テンプレート攻撃のエラーレートは，常に CPA の結果よりも低いことが分かる．また，全 16 個の部分鍵を正しく推定するために，テンプレート攻撃では 2400 波形が必要であったのに対し，CPA では 5,000 波形を用いても不十分であった．このように，鍵推定全体の結果からも，提案するテンプレート攻撃の攻撃能力の高さを確認できる．

4.4 プロファイリングに関する考察

前節の結果より，ハードウェア実装した暗号モジュールについても，テンプレート攻撃が可能であることが示された．以下では，テンプレート攻撃のプロファイリングについて考察する．

前節の解析では，(i) テンプレートの作成に使用する波形数，および (ii)*interesting points* を，経験的に決定していた (波形数 200,000 および *interesting points* 5 点)．多くの文献 [13, 15] でも同様に，これらのパラメータを経験的に決定している．以降では，この 2 点のパラメータを変更した際の，鍵推定の精度への影響について検討する．

4.4.1 プロファイリングに用いた波形数の影響

プロファイリングに用いる波形数を変化させ，攻撃の結果への影響を比較した．このために，プロファイリングフェーズで使った波形数を 1,000, 5,000, 10,000, 20,000, 60,000 と段階的に変化させ，それぞれの波形数でテンプレート攻撃を行った．なお，この際，*interesting points* は 5 点とした．

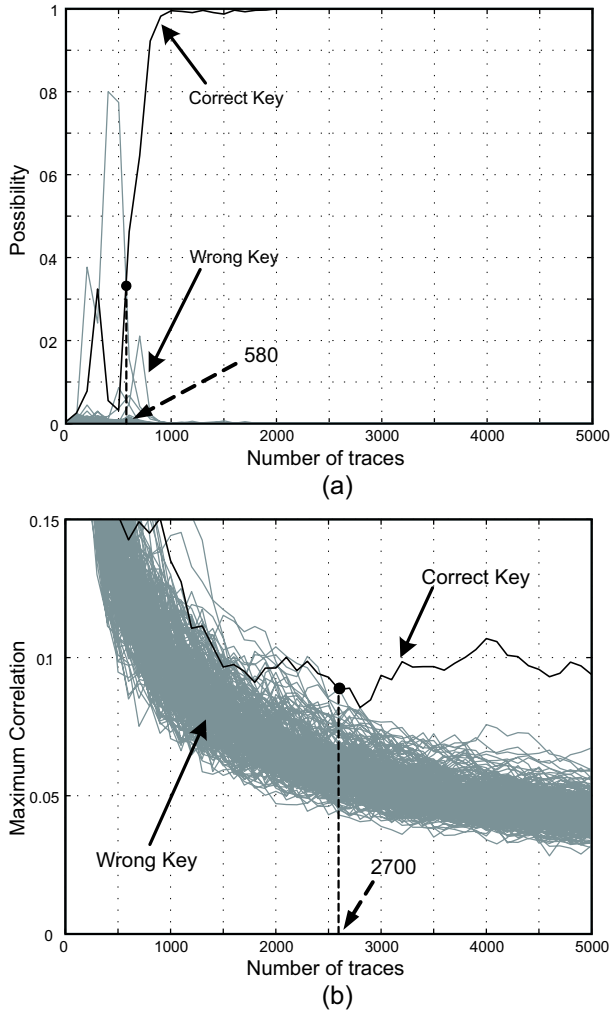


図 8: 実験結果　(a) テンプレート攻撃，(b)CPA 攻撃

結果をエラーレートで表示したものを図 11 に示す．図 11 より，プロファイリングに使う波形が少ない場合，攻撃の成功率が低下することが分かる．ただし，波形数は 10,000 個以上あれば，ほぼ同等の結果が得られることが分かる．この結果より，テンプレート攻撃による利点を得るには，十分な数の波形数を用意する必要があることが分かる．なお，参照モジュールを所持する攻撃者であれば，本実験と同様な手法により，使用した波形数が十分かどうかを確認することが可能である．

4.4.2 *interesting points* 数による影響

テンプレートの作成に用いる *interesting points* の数を段階的に変化させ，攻撃結果への影響を調査した．第 3 章で述べた方法で φ_t を求め，その上位 5，15，25，35，45，55 点を *interesting points* とした．一方，プロファイリングに使う波形数は 150,000 とした．図 12 に解析結果を示す．その結果，35 点までは点数を増やすほど結果が改善することが分かる．一方，点数をそれ以上増加した場合，逆に解析精度が低下することが分かる．これは，点数を増やすことで，φ が低い点がテンプレートに含まれ，これがノイズとなり，結果としてテンプレートの品質が低下するためであると考えられる．

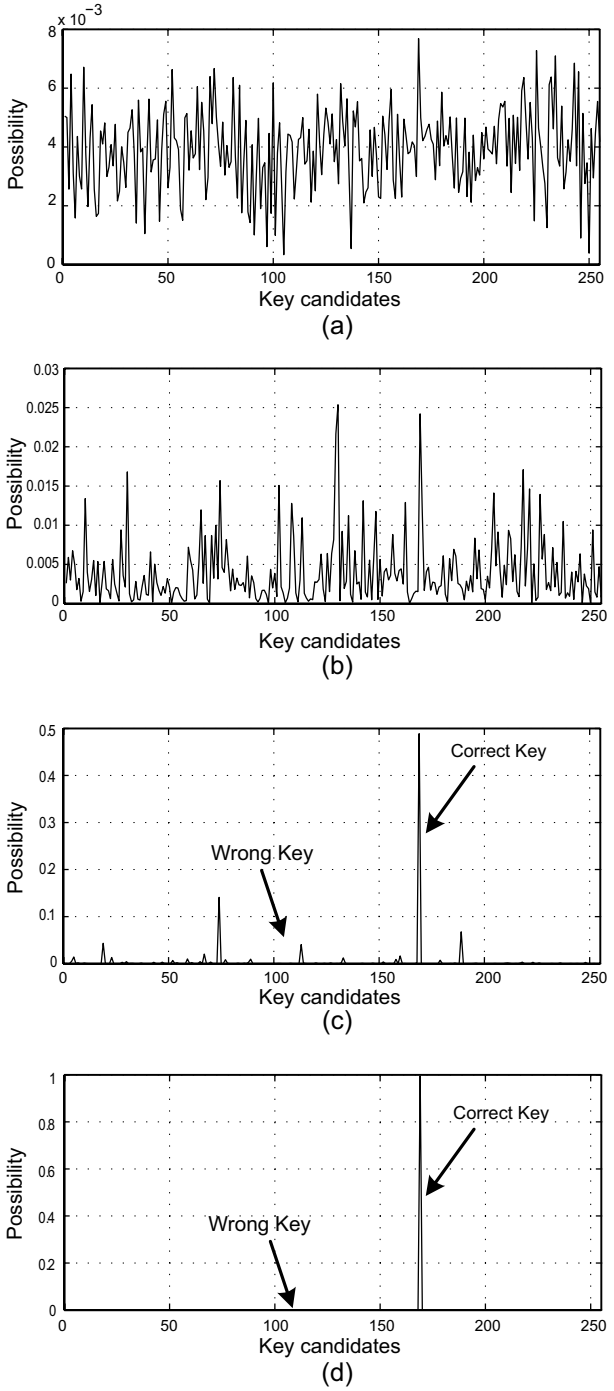


図 9: 攻撃フェーズで用いた波形数毎のテンプレート攻撃の結果 (a)10 波形 (b)100 波形 (c)600 波形 (d)1000 波形

5. むすび

本稿では，並列性の高いハードウェア実装に対しても適用可能なテンプレート攻撃を提案した．提案手法は，従来のテンプレート攻撃と比べて波形数は必要となるが，テンプレート数と共分散行列サイズの大幅な削減

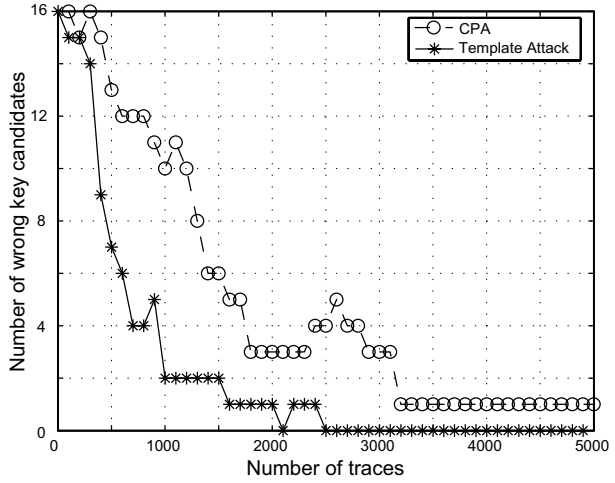


図 10: エラーレート

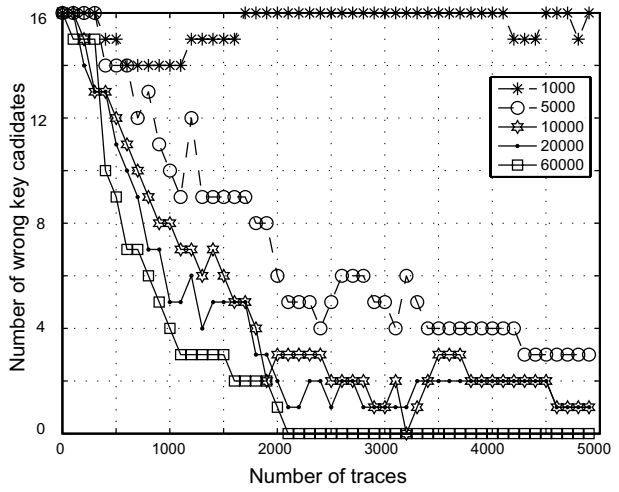


図 11: プロファイリングに用いた波形毎のエラーレート

が可能であり，鍵長と同一のデータパスをもつ暗号回路に対しても現実的な時間で計算が可能となる．本稿では，その有効性を評価するため，AES 回路の FPGA 実装に対する攻撃実験を示した．その結果，提案手法により全ての部分鍵の推定に成功した．また，CPA 攻撃と比較したところ，より少ない波形数で攻撃に成功することを確認した．さらに，波形数およびテンプレート点数の評価から，攻撃に成功する条件を考察した．

今後はテンプレート攻撃に対する対策法を開発するとともに，その対策を施した実装に対する攻撃実験を実施する．また，ASIC 実装に対する有効性も合わせて評価する予定である．

参考文献

[1] P. Kocher, J. Jaffe, and B. Jun. Differential power analysis. *Lecture Notes in Computer Science*, Vol. 1666, pp. 388–397, August 1999.

[2] K. Gandolfi, C. Mourtel, and F. Olivier. Electromagnetic analysis: Concrete results. *Lecture Notes in Computer Science*, Vol. 2162, pp. 251–261, May 2001.

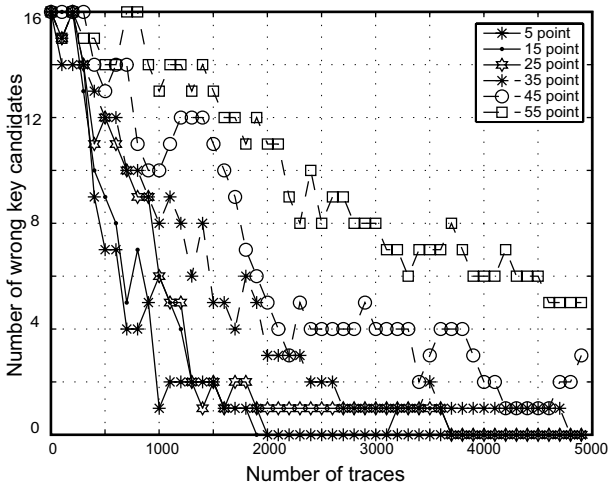


図 12: *interesting points* 数毎のエラーレート

[3] E. Brier, C. Clavier, and F. Olivier. Correlation power analysis with a leakage model. *Lecture Notes in Computer Science*, Vol. 3156, pp. 135–152, 2004.

[4] D. Agrawal, J. R. Rao, P. Rohatgi, and K. Schramm. Templates as master keys. *Lecture Notes in Computer Science*, Vol. 3659, pp. 15–29, 2005.

[5] M. Medweb and E. Oswald. Template attacks on ECDSA. *Workshop on Information Security Applications*, pp. 14–27, 2008.

[6] E. Oswald and S. Mangard. Template attacks on masking - resistance is futile. *Lecture Notes in Computer Science*, Vol. 4377, pp. 243–256, 2006.

[7] E. Peeters, F.-X. Standaert, N. Donckers, and J.-J. Quisquater. Improved higher-order side-channel attacks with FPGA experiments. *Lecture Notes in Computer Science*, Vol. 3659, pp. 309–325, 2005.

[8] C. Archambeau, E. Peeters, F.-X. Standaert, and J.-J. Quisquater. Template attacks in principal subspaces. *Lecture Notes in Computer Science*, Vol. 4249, pp. 1–14, 2006.

[9] M. A. El Aaabid, S. Guilley, and P. Hoogvorst. Template attacks with a power model. *Cryptology ePrint Archive*, 2007. http://eprint.iacr.org/2007/443.pdf.

[10] AIST Research Center for Information Security. Side-channel Attack Standard Evaluation BOard (SASEBO). http://www.rcis.aist.go.jp/special/SASEBO/index-en.html.

[11] S. Mangard, E. Oswald, and T. Popp. *Power Analysis Attacks*. Springer, 2006.

[12] F.-X. Standaert, S. B. Ors, J.-J. Quisquater, and B. Preneel. Power analysis attacks against FPGA implementations of the DES. *Lecture Notes in Computer Science*, Vol. 3203, pp. 84–94, Aug 2004.

[13] C. Rechberger and E. Oswald. Practical template attacks. *Lecture Notes in Computer Science*, Vol. 3325, pp. 440–456, 2005.

[14] NIST. Advanced Encryption Standard (AES). *Federal Information Processing Standards Publications*, Vol. 197, , 2001. http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf.

[15] T.-H. Le, C. Canovas, and J. Clediere. An overview of side channel analysis attacks. *Proceedings of the 2008 ACM symposium on Information, computer and communications security*, pp. 33–43, 2008.