IMES DISCUSSION PAPER SERIES

暗号ハードウェアの研究開発動向：
フィジカリー・アンクローナブル・ファンクション

菅原 健（すがわら たけし）

Discussion Paper No. 2020-J-6

IMES

INSTITUTE FOR MONETARY AND ECONOMIC STUDIES

BANK OF JAPAN

日本銀行金融研究所

〒103-8660 東京都中央区日本橋本石町 2-1-1

IMES Discussion Paper Series 2020-J-6
2020 年 3 月

暗号ハードウェアの研究開発動向：
フィジカリー・アンクローナブル・ファンクション

菅原 健（すがわら たけし）*

要　　旨

フィジカリー・アンクローナブル・ファンクション（Physically Unclonable Function：PUF）とは、大量生産された製造物の個体識別を行うための技術である。PUF は実用化の段階にあり、PUF を搭載したチップが次々に市場投入されている。その中で、PUF の研究者のみならず、PUF を利用する可能性がある分野の技術者や実務家も PUF の長短所やセキュリティ特性について理解を深めておくことが一段と重要になる。本稿は、そのような読者を想定し、代表的な PUF の実現法、PUF に求められる性質、PUF の応用法、PUF の既存の攻撃法などについて述べる。PUF の代表的な利用法は 2 つある。第一は、PUF を暗号の軽量な代替として用いることであり、RFID のように極限的に計算資源が限られた機器にセキュリティを付与することができる。第二は、PUF を暗号鍵の保管庫として利用することであり、IC カードなどの高セキュリティ製品においてリバースエンジニアリング耐性を付与することができる。

キーワード：暗号ハードウェア、ハードウェア・セキュリティ、フィジカリー・アンクローナブル・ファンクション、Physically Unclonable Function

JEL classification: L86、L96、Z00

* 電気通信大学（E-mail: sugawara@uec.ac.jp）

本稿は、日本銀行金融研究所からの委託研究論文である。本稿の作成に当たっては、藤野毅教授（立命館大学）に有益なコメントを頂いた。ここに記して感謝したい。ただし、本稿に示されている意見は、筆者個人に属し、日本銀行の公式見解を示すものではない。また、ありうべき誤りはすべて筆者個人に属する。

目次

1 はじめに

フィジカリー・アンクローナブル・ファンクション (Physically Unclonable Function: PUF) とは、大量生産された製造物 — 多くの場合、半導体技術を用いて作った集積回路 (チップ) — の個体識別を行うための技術である。より詳細には、製造時に生じる制御不可能な製造ばらつきを利用して、その個体に固有なランダム関数を作る技術を指す。製造ばらつきを利用するため、大量生産した製品に、(プログラムなどを書き込むこと無く) 自動的に個性を付与できる。その個性は制御不可能な製造ばらつきに由来するので、その個体の複製 (クローン) の作成を防ぐことができる。PUF を利用することで、模倣品排除などを行うことができる。

PUF は研究の時期を乗り越え、PUF を搭載したチップが次々に市場投入されつつある (2 節 (3) で詳しく述べる)。また、国際標準化も進められており (International Organization for Standardization and International Electrotechnical Commission [2019])、今後応用が発展していくと期待される。その中で、PUF の研究者のみならず、PUF を利用する可能性がある分野の技術者や実務家も PUF の長短所やセキュリティ特性について理解を深めておくことが一段と重要になると考えられる。中でも、高い安全性の要求を満たすために IC カードを利用してきた金融分野は、他分野に先駆けて PUF の普及が進む可能性が高い。

個体差を見分ける技術としては、指紋や虹彩などを用いるバイオメトリクスが広く実用化されている (宇根・松本 [2005])。それとの対比で、「PUF は物体の指紋である」と言われることがある。これは、PUF の一面を正しく表しているものの、両者には違う点もある。特に、チップは電源を切ってもデータを保持し続けるメモリ (不揮発メモリ) を持つことができるため、不揮発メモリに書き込んだシリアル番号を用いた個体識別ができる。そのため、あえて PUF を用いるからには、シリアル番号のような代替案と比べた時の利点がなければいけない。それがどのようなときであるか (すなわち、読者のケースで有用かどうか) を判断するには、PUF が提供するセキュリティについてよく理解しなくてはいけない。

本稿は、セキュリティを扱う技術者・実務家を読者として想定し、彼らの用途に対し PUF が有用であるか、またそうであるとしたらどのように用いるべきかを判断するため

の情報を提供することを目的とする。本稿が、PUF の導入を検討している技術者、実務家への一助になることができれば幸いである[*1]

2 PUF: Physically Unclonable Function

(1) PUF とは

現代のコンピュータは半導体技術を用いた集積回路として作られている。そのような製品を作る製造者にとっては、通常、品質のばらつきが無い製品を大量生産することが目的となる。一方、品質管理や模倣品を排除するためには、大量生産された製品の各個体を見分け、追跡する必要がある。

そのような目的のため、従来は、製品に刻まれたシリアル番号などが利用されていた。シリアル番号は、悪意を持った攻撃者による攻撃には耐えられない。シリアル番号の重大な課題は 2 つある。1 つは容易にコピーされることである。シリアル番号はただの数字（もしくは文字）の列であるため、一度見れば、同じシリアル番号を持つ模倣品を作ることは簡単である。これは、データとして書き込まれたシリアル番号でも同様である。もう 1 つの課題は、多くの場合、シリアル番号と製品を分離可能なことである。すなわち、シリアル番号の書かれたシールを貼り替えたり、ある製品のシリアル番号を別の製品にコピーしたりということを行うことができる。

PUF とは、製造時に生じる制御不可能なばらつきを建設的に利用して、ある個体に固有な「関数」を作るための技術である。PUF の頭文字には次のような意味がある:

イ PUF の F (Function)

利用者から見た PUF はチップ固有の関数である。関数であるとはすなわち、入力（チャレンジ）を入れると対応する出力（レスポンス）が出てくることを意味する。チップ固有であるとは、違う個体では、チャレンジとレスポンスの関係が異なる（すなわち異なる関数である）ことを意味する。そのため、シリアル番号のように、一度見ただけでコピーされたりしない。

[*1] より網羅的かつ詳細な知識を得るための情報源としては、Maes による著書 (Maes [2013]) を推薦する。また、本論文に添付する参考文献も参照されたい。

ロ　PUF の P (Physical)

上記のような個体に固有である性質は、物理的な理由により生じる。製造者は、同一の設計図から大量の個体を生産するが、その際に発生する製造ばらつきにより、チップごとに異なる関数が自動的に実現される。

ハ　PUF の U (Unclonable)

ここまで述べたチップ固有の関数はコピーできない。(クローンを作ることができない)。すなわち、チップと PUF は切り離すことができない。そのため、シリアル番号のように、PUF だけを移植するような攻撃はできない。

(2)　PUF の簡単な歴史

製造物に固有の関数を作るというアイディアは、2001 年の Pappu による (Pappu [2001])。これは、気泡を含む透明な樹脂の個体を見分けるというものであり、樹脂にレーザを照射することで得られる干渉パターン（レーザースペックル）を出力として用いるというものであった。すなわち、気泡の含まれ方が個体ごとに違うということが個体差の根源であり、それを干渉パターンとして外部から取り出すことができる点に特徴があった。

続いて、半導体で PUF を作ることが続くチャレンジとなった。もし可能ならば、半導体を用いて作る計算機やメモリと混載することで、セキュリティ上の応用が一気に広がるためである。その後、Gassend が半導体で作った回路の性能ばらつきを用いた半導体製の PUF を提案した (Gassend [2003])。続いて Lim が、ばらつきを効率的に取り出す回路構造として、3 節 (1) で詳しく述べるアービター PUF を提案した (Lim [2004])。

半導体の加工精度を洗練することで回路を微細化することが半導体技術の大目標であり、ムーアの法則と呼ばれる経験則に基づいて継続的な微細化が達成されてきた。計算機の高速化や、メモリの大容量化は、これらの微細化の成果として達成されたものである。その中にあって、回路の個体差・製造ばらつきの低減は乗り越えるべき技術的課題であった。製造ばらつきを積極的に利用するというアイディアは、製造ばらつきに悩まされてきた回路技術者・研究者達にとって魅力的なアイディアであり、Gassend や Lim による研

究成果が発表されて以降、PUF の実現法が次々と提案された。

続いて、PUF と暗号技術の統合が研究上の大きなチャレンジとなった。暗号は通信データを秘匿したり、その改ざんを検出したりするための数理的なアルゴリズムであり、情報セキュリティの基盤技術である。PUF は、製造ばらつきを積極的に取り出すように回路を動作させるため、ノイズ入りの不確定なデータが出力される。そのような不確定性を数理的アルゴリズムとどのように組み合わせるかという点が大きな課題であった。それに対し、ファジー抽出器と呼ばれるアルゴリズム（5 節 (2) で詳しく述べる）を用いることで、PUF の出力に含まれるノイズを訂正する方法が提案された (Guajardo *et al.* [2007])。その結果、PUF の出力を用いて鍵を生成し、その鍵を用いて暗号を利用することが可能になった。

(3) PUF の実用化動向

PUF は、2000 年代初期から研究が始まり、過去の 20 年に活発に研究がされた。そこで、本論に入る前に、PUF の実用化動向について概説する。これは、プレスリリースなどの公開情報に基づくものであり、また、網羅的では無いことに注意されたい。通常、半導体チップを搭載した電子製品の製造には、異なる複数の会社が関わっている。そこで、以下では、製品の層ごとに説明を行う。

イ　設計情報の販売

現在の半導体チップの設計では、個々の回路（回路部品）を設計する会社と、それらの回路をチップとしてまとめる会社の分業が進んでいる。その結果、回路の設計情報をソフトウェアライブラリのように売買する商習慣がある。そのようなハードウェアの設計情報を IP (Intellectual Property) と呼ぶ。また、自社ではチップ全体の設計・製造は行わず、IP の開発を行うベンダを IP ベンダと呼ぶ。PUF の IP を売るベンダーとしては、アメリカ Verayo とオランダ Intrinsic-ID が有名である。Verayo は、後述する アービター PUF を主たる商品にしている。一方、Intrinsic-ID は、同じく後述する SRAM PUF を主な製品としている。

ロ　チップベンダ

チップの設計・製造に責任を持つベンダをチップベンダと呼ぶ。チップベンダは、自社が開発・製造するチップへの付加価値として PUF を搭載することがある。PUF はチップベンダが独自に開発することもあるし、IP を購入することもある。

再構成可能なハードウェアである Field-Programmable Gate Array (FPGA) は、PUF の搭載が最も進んでいるチップの 1 つである。Xilinx (Xilinx [2016]、Menhorn [2018])、Altera (現 Intel) (Altera Corporation [2015])、Microsemi (Microsemi Corporation [2016]) などの FPGA ベンダが、PUF の搭載を報告している。

CPU に PUF を搭載する事例もある。一例として、NXP Semiconductors のセキュリティ向けマイクロコントローラ (NXP Semiconductors [2016]) がある。また、Intel の CPU は、安全にソフトウェアを実行する Software Guard Extensions (SGX) と呼ばれる機能において、PUF を利用した鍵管理を行っているという報告がある (Costan and Devadas [2016])。

以上に加え、認証のみを目的とした専用のチップ（認証チップ）に PUF を搭載したものを Maxim が販売している (Jones [2017, 2018])。

ハ　メーカー

複数のチップを組み合わせた回路基板や、それらを制御するためのソフトウェアを開発して、最終製品を開発する会社を、ここではメーカーと呼ぶ。それらの企業は、上記チップベンダからチップを購入して利用することもあるし、自社でチップを製造することもある。また、外部から購入した FPGA に PUF をプログラムして使うという中間の実現法も可能である。メーカーが PUF の実用化を行った事例として、三菱電機がプレスリリースを行っている (三菱電機株式会社・立命館大学・科学技術振興機構 [2015]).

ニ　ソリューション・サービス

PUF を利用するには、ハードウェアとして PUF だけではなく、PUF を制御するためのソフトウェアであったり、PUF を登録したり追跡したりするためのシステムが必要である。そこで、PUF を利用するセキュリティソリューションを提供する形態がある。これは、元々ソリューション販売を行っている会社が PUF を取り込む場合もあれば、上

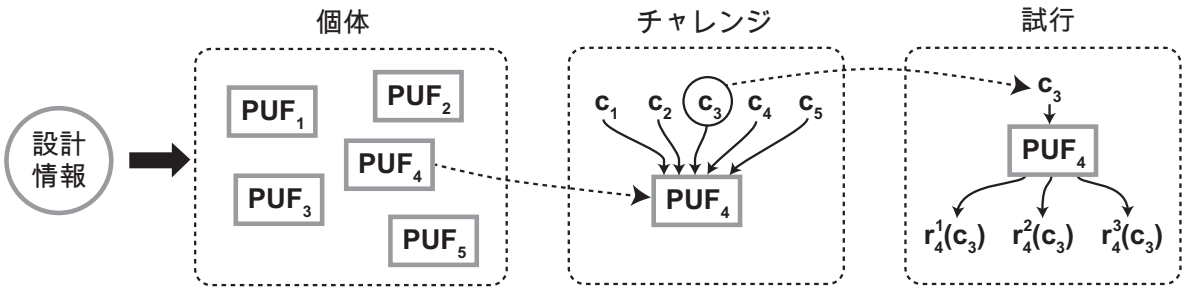

図 1　PUF の 3 つの軸: 個体、チャレンジ、試行

記の PUF の IP やチップを販売する会社が合わせてソリューションを提供する場合もある。前者として、凸版印刷 (凸版印刷株式会社 [2013]) が PUF を搭載した IC タグを用いた真贋判定のサービスを提供すること、また、それをタニタの製品に適用したこと (凸版印刷株式会社 [2015]) が知られている。

(4)　PUF の 3 つの軸

PUF の機能を理解する上では、図 1 に示す 3 つの軸が重要である。

イ　個体の軸

PUF は、ある 1 つの設計情報を元に製造されるたくさんの個体を識別することを目的としている。ある設計情報から製造した N_{PUF} 個の個体を $\mathsf{puf}_1, \cdots \mathsf{puf}_{N_{PUF}}$ と呼ぶことにしよう。各個体ごとの違いが個体の軸である。

ロ　チャレンジの軸

PUF は関数であるため、ある入力を入れると、対応する出力が得られる。PUF では、伝統的に、入力のことをチャレンジ、出力のことをレスポンスと呼ぶ。N_{chal} 個のチャレンジを $c_1, \cdots, c_{N_{chal}}$ と書くことにする。i 番目の個体に k 番目のチャレンジを入力することを $\mathsf{puf}_i(c_k)$ と書く。このように、どの値を入力したかの違いがチャレンジの軸である。

ハ　試行の軸

PUF では、微小な個体差を取り出すように回路を動作させるため、レスポンスにはノイズが含まれる。そのため、同じ個体に同じチャレンジを入力したとしても、試すたびに異なる出力（レスポンス）が得られる可能性がある。そのように、試すたびに生じる違いが試行の軸である。i 番目の個体に k 番目のチャレンジを入力する試行の j 番目の試行を

$$r_i^j(c_k) \leftarrow \mathsf{puf}_i(c_k) \tag{1}$$

と書く。$r_i^j(c_k)$ はレスポンスである。3 つの添字 i、j、k がここまで述べた 3 つの軸に対応している。

(5)　PUF に求められる性質

本節では、PUF に求められる性質を通して、PUF とはどのようなものであるかをより詳しく述べる。本節で述べる性質の有無や程度は、PUF を実現するための回路方式の良し悪しを比べる際の基準としても用いられる。ただし、何が「求められる性質」であるかは、研究コミュニティの中でも強いコンセンサスが得られている状態ではないことに注意されたい。本節の内容は、主に文献 (Maes [2013]) に基づいている。より強いコンセンサスを形成しようとする試みが、国際標準化の中で進められている (International Organization for Standardization and International Electrotechnical Commission [2019])。

イ　再現性: 出力が安定していること

前述の通り、PUF の出力にはノイズが含まれているため、同じ個体に同じチャレンジを入力したとしても、試すたびに異なるレスポンスが出る。ノイズが大きいほど、すなわち試行ごとのばらつきが大きいほど、ノイズを訂正するなどの追加の手間が増える。そのため、実用の観点からは、ノイズが小さい（出力のばらつきが小さい）ことが好ましい性質である。その性質を表すための指標が再現性 (Reproducibility) である。これは、試行の軸に関する性質である。

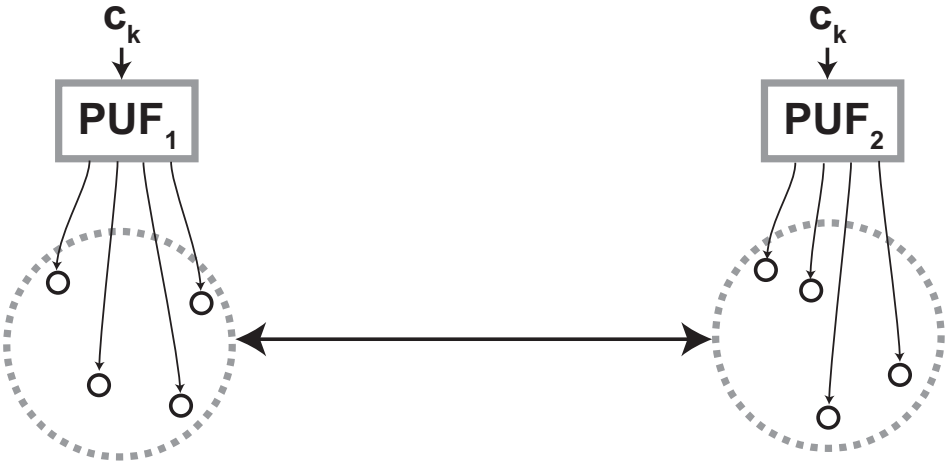

図 2　PUF の再現性とユニーク性

ロ　ユニーク性: 個体同士が似ていないこと

PUF を用いる究極的な目的は、各個体を見分けることにある。そのため、個体ごとの違いが大きいほど優れた PUF であると言える。個体ごとの類似度は、別の個体（たとえば puf_1 と puf_2）に同じチャレンジを入力した時、出力がどれくらい似ているかによって表される。もし、相異なる個体がよく似たレスポンスを出すと、本来は異なるそれらの個体を、同一であると誤判定してしまう可能性が高まってしまい問題である。そのように、個体差の大きさを示す指標をユニーク性 (Uniqueness) と呼ぶ。これは、個体の軸に関する指標である。

図 2 に再現性とユニーク性の関係の模式図を示す。ある PUF に同一のチャレンジを入力し続けたときのレスポンスの散らばりを点線の円で表現している。再現性が良いということは、この散らばりが小さい（円の半径が小さい）状態として理解できる。一方、異なる個体から得たレスポンス同士は離れていて欲しい。ユニーク性が良いとは、このような円同士の距離が遠い状態であると理解できる。

ハ　耐クローン性: クローンを作ることができないこと

PUF では、チップ固有の関数を利用することで、ある個体を別の個体と識別する。この識別法がうまくいくための前提条件として、ある個体とよく似た別の個体（クローン）を作ることができないことが必要である。そうでなければ、クローンを用いたなりすまし

ができてしまう。耐クローン性には 2 つの意味がある。1 つは、正規製造者であっても、クローンを作ることができないこと、もう 1 つはクローンを試みる攻撃への耐性を持つこと（クローンを作るための既知の攻撃が存在しないこと）である。

正規製造者であってもクローンを作ることができないこととは、PUF の個体差の根源にある製造ばらつきが制御不可能であることを意味する。半導体では、製造時に行うナノメートル級の微細加工における製造ばらつきがあり、制御不可能であると信じられている。

もし、製造ばらつきを制御することが不可能であっても、別の方法でクローンを作ることができてしまえば問題である。実際、そのような攻撃法がいくつも知られている。そこで、(i) そのような既存の問題が無いこと、(ii) 問題が存在するが、適切な対策法が実施されていること、(iii) 攻撃が成功するリスクが許容できるレベルにあることなどが必要である。攻撃については、6 節で詳しく述べる。

ニ　予測困難性と一方向性: 暗号アルゴリズムのように使えること

PUF の理想は、暗号と遜色無い性質を持つ関数を実現することにある。そこで、暗号アルゴリズムに求められる性質を PUF に移植した性質が予測困難性と一方向性である。それらの性質があれば、既存の暗号アルゴリズムを PUF で置き換えることができる。そうすることで、暗号と PUF が融合した高機能なプロトコルなどを実現できる。

予測困難性とは、たとえ既知のチャレンジ・レスポンスペアを用いて事前学習したとしても、あるチャレンジに対応するレスポンスを（実際に計測することなく）予測することが難しいことを表す。これはすなわち、規則性の無いランダムな関数であるという理想を表したものである。それに対し一方向性は、レスポンスが与えられた時、対応するチャレンジを求めることができないことを表す。これもまた、PUF が作る関数が規則性の無いランダムなものであって欲しいという性質を表したものである。

実用的な PUF から得られるチャレンジ・レスポンスの関係には、規則性が現れがちであり、その規則性を利用すれば予測困難性・一方向性を破る攻撃ができることがあることに注意が必要である。6 節 (1) で述べる機械学習攻撃は、その 1 つである。「予測困難性を持つような PUF を作ることができるのか?」 ということが本分野の重要な未解決問題である (Maes [2013])。そのため、それらの欠如をリスクとして受容するか、それらの性質が無くても良い PUF の利用法を採用する必要がある。

ホ　耐タンパー性: チップを不正に開封しようとすると痕跡が残ること

後述するように、PUF は、チップを削って内部にプローブを当てるような、苛烈な攻撃にさらされる可能性がある。そのような攻撃が行われた時、攻撃の痕跡が残る性質のことをタンパー証跡 (Tamper Evidence) と呼ぶ。特に PUF においては、チップを削ったりする加工を行うことで、PUF に関係する物理的な状況が変わってしまい、結果としてそのチップに固有な関数が別の関数に変貌してしまうことを表す。そのような性質があれば、PUF を、そのような攻撃への対策法として利用することができる。

ただし、実験でこの性質を検証した事例は少なく (Kerst *et al.* [2005]、鳥居ほか [2015]、山本ほか [2015])、実際に耐タンパー性を有する PUF を作ることができるか、という点もまた未解決問題であるといえる。

ヘ　その他の性質

ここまで述べたセキュリティに関する性質に加え、実装に関する優れた性質を持つことが好ましい。特に、PUF は、暗号の軽量な代替として利用されることがある。そのような場合、暗号と比較して軽量に実装できることが重要な性質の指標となる。また、6 節で述べるように、特定の PUF の実現法に特有の攻撃がある。そのような攻撃が無いことや、低コストで対策できることもまた実装に関わる重要な性質である。

また、どのくらい容易に PUF を実現できるかという点も重要な性質である。いくつかの PUF は FPGA でも実現できる (Guajardo *et al.* [2007])。そのような PUF は、自社でチップを製造しない企業（FPGA のチップを調達し、それをプログラムして利用する企業）でも利用できるという利点がある。また、標準的な半導体の製造技術（CMOS 論理回路プロセス）で実現できる PUF は、低価格で製造できることや、外部の会社に製造を委託できるという利点がある。

3　PUF の実現例

本章では、PUF がどのように実現されるかを説明する。PUF の実現法はたくさんあるが、中でも最も有名なものとして アービター PUF (Lim [2004]) と SRAM PUF (Guajardo *et al.* [2007]、Holcomb, Burleson, and Fu [2009]) について説明する。

(1) アービター PUF

アービター PUF は、回路の中を信号が伝わる時間（信号遅延）を個体差の根源として利用する (Lim [2004])。回路の中では、論理値 0 を V_L ボルト、論理値 1 を V_H ボルトのように、異なる電圧で論理値を表現する。あるタイミングで電圧を V_L から V_H に切り替えると（すなわち論理値を 0 から 1 に切り替えると）その変化は電気信号の変化として回路の中を伝わる。その時どのくらい速く信号が伝わるかは、回路を構成するトランジスタや配線の出来栄えによって個体差を持つ。

図 3 に、アービター PUF の回路構造を示す。アービター PUF は、信号遅延を効率的に抽出するために、2 つの異なる経路で信号の伝搬を競争させ、どちらが速かったかによって出力を決定する。すなわち、ある一方が早ければレスポンスは 0、もう一方が速ければレスポンスは 1 というように、いずれが速かったかをレスポンスとして出力する。アービター PUF の後半部は、どちらに速く信号が届いたかを判定するための回路である。

アービター PUF の前半部は、信号が伝搬する経路である。この経路は、入力したチャレンジによって変化するようになっている。すなわち、チャレンジを変えるたびに対象の経路が変わるので、対応するレスポンスが変化するのである。そのための回路は、回路を信号が直進するか交差するかを選択するスイッチボックスを基本とする。すなわち、対応するチャレンジのビットが 0 ならば直進、1 ならば交差とする。このようなスイッチボックスをチャレンジのビット数だけ縦列接続することでアービター PUF を構成する。

(2) SRAM PUF

SRAM とは、小型・高速だが電源を切ると中身が消えるメモリ（揮発メモリ）である。マイクロコントローラなどのチップ内にあり、プログラムやデータなどをチップ上に一時的に保管するために利用されている。通常のメモリと同様に、アドレスで指定した箇所にデータを読み書きすることができる。

図 4 に SRAM の構造を示す。SRAM は、1 ビットの情報を記録する最小単位であるメモリセルを行列状に並べたものである。行列の周囲には、アドレスに応じて特定のメモリセルのみを有効化する回路（行デコーダーと列デコーダー）がある。それらを用いて特

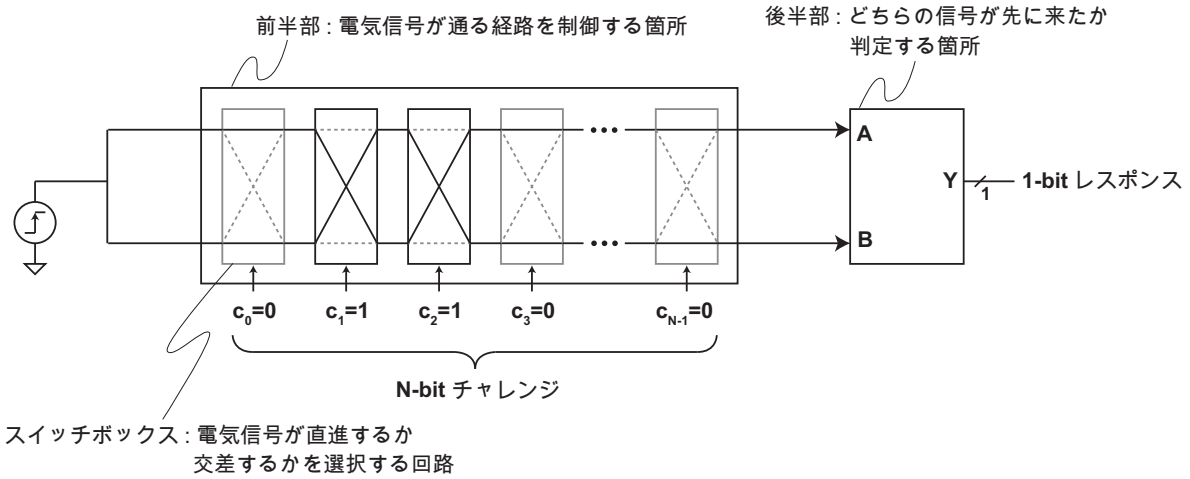

図 3　アービター PUF

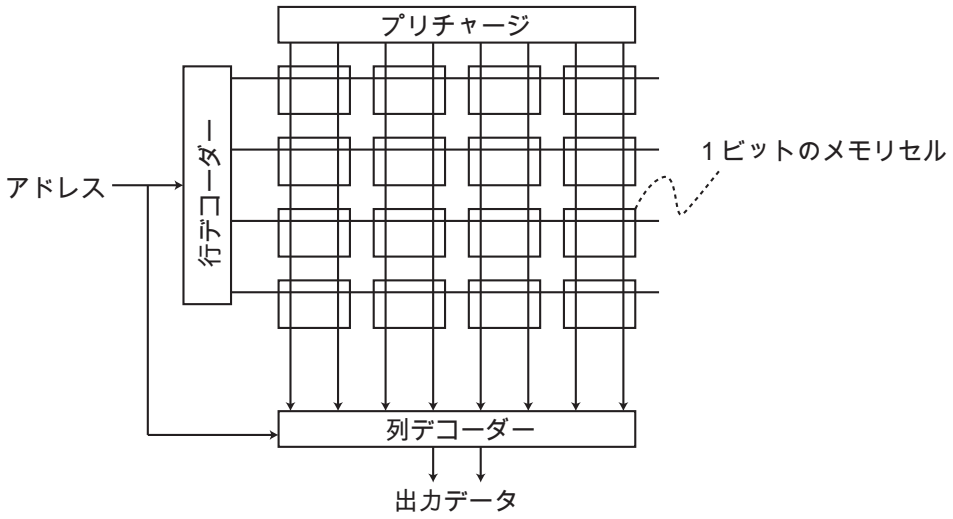

図 4　SRAM の構造

定のメモリセルを有効化することで、読み出し・書き込みを選択的に行うことができる。メモリセルは、物理的に安定な状態を 2 つ持っており、各状態が論理値 0 と 1 に対応する。ある安定状態にあるメモリセルは、電源が投入されている間はその状態を保持し続ける。そうすることでデータが記憶できる。また、外部から電圧を加えることで、片方の安定状態をもう片方の安定状態に遷移させることが、メモリへの書き込みに対応する。

SRAM PUF とは、電源投入直後に、まだ何も書き込んでいない SRAM から読み出した値（初期値）を個体差として利用する PUF である (Guajardo *et al.* [2007]、Holcomb,

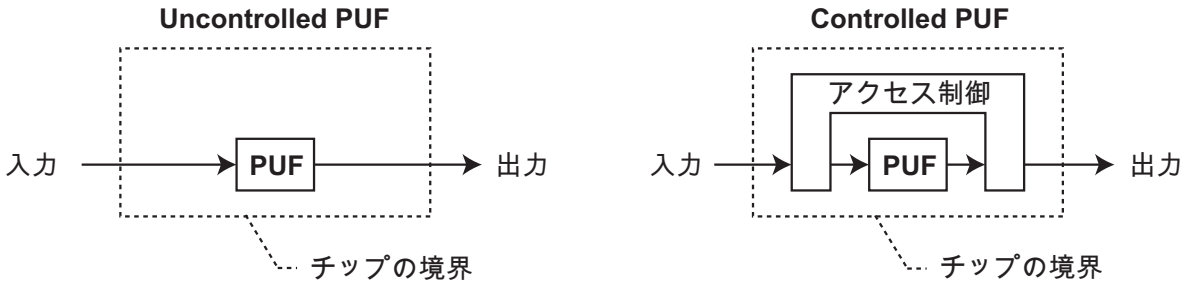

図 5　Controlled and Uncontrolled PUF

Burleson, and Fu [2009])。SRAM PUF では、アドレスがチャレンジ、そのアドレスに格納された初期値がレスポンスとなる。電源投入直後、メモリセルは不安定な状態からスタートする。しかし、起動からごく短時間のうちに、2 つの安定点のうちいずれかに遷移する。いずれに遷移するかによって、初期値が 0 になるか 1 になるかが決まる。0 と 1 のいずれに遷移するかは、メモリセルを構成するトランジスタの特性（しきい値電圧や駆動電流）によって決まるが、この特性が製造ばらつきを持つ。その結果、SRAM の初期値は個体ごとの個性を持つ。

(3)　PUF の 2 つの使い方: Controlled PUF と Uncontrolled PUF

PUF には大きく分けて 2 つの方法がある。1 つ目の方法は、ユーザに対して PUF の関数 puf_i をそのまま公開する方法である。このように、puf_i へのアクセス制御が無い PUF の利用法を Uncontrolled PUF と呼ぶ (図 5-(左))。それに対し、ユーザに PUF の生の入出力にアクセスさせない方法を Controlled PUF と呼ぶ (図 5-(右))。

Uncontrolled PUF はより素朴な利用法であり、アクセス制御のためのディジタル回路が不要であることから、回路をより小型にすることができるという利点がある。一方、ディジタル回路による制限が無い分、Uncontrolled PUF の方がより強力な攻撃にさらされる。特に、ユーザ＝攻撃者は、ありうる全てのチャレンジを puf_i に入力し、対応するレスポンスを入手して対応表を作ることができる。この対応表を利用すれば、与えられたチャレンジに対し、元の PUF と同じ出力を生成することができる。これは PUF のクローンを作る攻撃である。このような攻撃は、PUF が提供するチャレンジ・レスポンスのペア数を、全探索できないほど大きくすることでしか対策できない。

チャレンジ・レスポンスのペア数を大きくすることは、どの PUF でも可能な訳ではな

い。アービター PUF は、回路面積の増加に対して、チャレンジ数が指数的に増加する(スイッチボックスの個数 n に対し、チャレンジ数が 2^n となる)。そのため、ある程度大きい値（たとえば $n = 128$) とすれば、ペア数は 2^{128} のように莫大な数になるため、全通り試すことはできなくなる。一方、SRAM PUF は、記憶できるビット数がチャレンジ数に対応するため、回路面積に対してチャレンジ数が線形にしか増加しない。典型的な SRAM のサイズは、数キロバイトから数メガバイトであるため、SRAM PUF を全探索攻撃から防ぐことは基本的にはできない。そのため、SRAM PUF は Controlled PUF として利用するのが基本である。

4 PUF に関連する技術

PUF は、暗号と組み合わせて利用することが一般的である。そのため、PUF の利点を理解するには、暗号の基礎について知っておく必要がある。そこで本章では、暗号を用いた認証、および鍵管理と物理セキュリティについて説明する。

(1) 暗号技術による認証

イ 共通鍵暗号

共通鍵暗号とは、あらかじめ秘密鍵を交換しておいた 2 人が、情報を隠しながら（秘匿しながら）通信を行うための技術である。

暗号化アルゴリズム E とは、メッセージ m を暗号文 c に変換するアルゴリズムであり、それらの対応関係は鍵 k によって変化する。鍵 k を用いてメッセージから暗号文を得ることを

$$c \leftarrow E_k(m) \tag{2}$$

と表現する。

暗号アルゴリズムは、対となる復号アルゴリズムを持つ。送られてきた暗号文 c を、鍵 k を用いて復号し、元のメッセージを復元することを

$$m \leftarrow E_k^{-1}(c) \tag{3}$$

と表現する。秘密鍵 k を持たない人は、暗号文 c をメッセージ m に戻すことができな

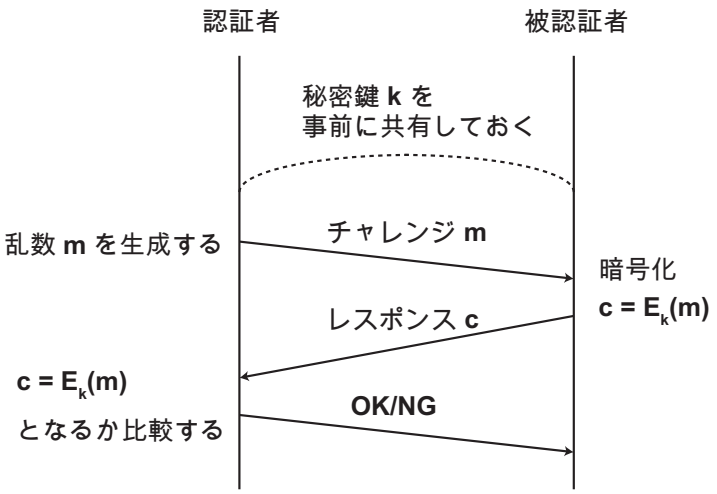

図 6　チャレンジ&レスポンス認証

い。そのため、安全に通信を行うことができる。

口　認証

通信相手が思った相手であること（すなわち、成り代わった攻撃者ではないこと）を確認すること、およびそのための技術を認証 (Authentication) と呼ぶ。相手が誰だか確認したいと思っている人を認証者、 確認される人を被認証者と呼ぶ。

あらかじめ交わしておいた対となる言葉を知っているかどうかで相手を確認する合言葉は、素朴な認証の一例である。実は暗号を用いた認証も同様であり、認証者と被認証者が、あらかじめ交換しておいた秘密（事前共有鍵）を持っているかどうかを確かめることで行う。ただし、盗聴者が居るため、秘密鍵を直接見せあって確認する訳にはいかない。

秘密鍵を露出することなく認証を行う手段にチャレンジ&レスポンス認証がある。これは、暗号化（式 2）を用いて認証を実現するための手続きである。チャレンジ&レスポンス認証の流れを図 6 に示す。認証者と被認証者は、事前に秘密鍵 k を共有している。認証者は、通信相手が k を持っているかどうかを確認したい。そこでまず、認証者は、ランダムなメッセージ m を生成し、チャレンジとして被認証者へ送付する。被認証者は、受け取ったチャレンジ m を鍵 k で暗号化し、レスポンス $c = E_k(m)$ を生成して返送する。認証者は、自分が送ったチャレンジ m を鍵 k で暗号化し、相手のレスポンス c と一致するかを検証する。一致した場合、認証者は、通信相手が k を持つこと、すなわち相手が正規の被認証者であることを確認できる。攻撃者は、鍵 k を持たないため、認証成功に必要

なレスポンス $c = E_k(m)$ を生成できない。そのため、攻撃者が被認証者になりすますことはできない。

チャレンジ m は、認証のたびに異なる値でなければならない。もし、同じチャレンジが繰り返されたら、次のような攻撃ができてしまう: 攻撃者は、チャレンジ m と、対応するレスポンス c の組を盗聴して記録しておく。もし、次に同じチャレンジ m が発生したら、記録しておいたレスポンス c を返信することで、認証に成功する。このように、過去の通信を記録して、後から同じデータを再生（リプレイ）する攻撃をリプレイ攻撃と呼ぶ。リプレイ攻撃を避けるためには、チャレンジは毎回異ならなくてはいけない。

(2) 鍵管理と暗号モジュール

ここまで説明したように、暗号における秘密は鍵に宿っている。これは、家や部屋のセキュリティが、扉の鍵に宿っているのと同じ関係である。そのため、鍵を安全に保管することは、暗号アルゴリズムを運用する上での大前提である。しかし、ディジタルデータである鍵の保管は、物理的な鍵の保管と比較しても難しい。鍵自体をどのようにして守るかという問題を鍵管理と呼ぶ。

物理的な鍵と同様に秘密鍵を持ち歩きたいことがある。しかし、人間が鍵を記憶するのは大変だし、また、直接鍵を入力する方式は、入力時に盗み見られるリスクがある。そこで、鍵を保管する不揮発メモリや、暗号アルゴリズムを実行するための計算機一式を 1 チップに封入する実装方式がよく用いられる。そのような実装方式を暗号モジュールと呼ぶ。

IC カードは代表的な暗号モジュールである。IC カードは、暗号を用いて、通信相手が正規リーダであるかどうかを、4 節 (1)-ロで示したような方法を用いて認証する。IC カードの内部には認証のための秘密鍵が埋め込まれており、同じ秘密鍵を持つリーダのみを相手として認めるのである。カードは、相手が正しいリーダであることが確認できた時に限り、支払いなどの安全性が必要な処理を行う[*2]。そのようにすることで、攻撃者が作ったリーダが、IC カードに不正な支払いをさせることを防いでいる。

[*2] 携帯電話の利用者認証のための Subscriber Identity Module (SIM) カードや、建物の入退室管理に用いるカードキーやトークン、車のスマートキーも暗号モジュールである。

(3) 物理攻撃とリバースエンジニアリング

暗号モジュールでは、物理的な攻撃への対策をしなくてはいけないことがある。IC カードの例を引き続き考える。IC カードの正規利用者が攻撃者になることがある。もしカードの内部にある秘密鍵を奪取できたら、IC カードを偽造するなどして不正な決済処理を行い、金銭的な利益を上げることができるためである。正規利用者＝攻撃者は、IC カードを物理的に所有しているため、チップを開封して電極を当てるような、物理的な攻撃を行うことができる。

暗号モジュール内部の不揮発メモリに記録された秘密鍵を読み取る攻撃法の 1 つに、静的リバースエンジニアリングがある。リバースエンジニアリングが静的であるとは、チップに電源が投入されていない状態で解析を行うことを表す。すなわち、チップを開封して内部を観察することで、記録された情報を復元する攻撃である。金属配線のパターンで記録する書き込み専用メモリ（マスク ROM）については、光学顕微鏡で配線パターンを見ることで、記録されたデータを復元できることが知られている (Torrance and James [2011])。また、不揮発性メモリ (フラッシュメモリや EEPROM) でも、電子顕微鏡などで観察すれば、記録された情報を復元できることが知られている (Courbon, Skorobogatov, and Woods [2016])。

以上の攻撃は、暗号などのセキュリティ機能を全て迂回して鍵を直接見に行くため、非常に強力な攻撃法である。また、電源オフ時のチップを解析するため、「侵入を検知して鍵を消去する」といった対策も取ることができない。本稿で後ほど分かるように、このような静的リバースエンジニアリングへの対策として利用できることが PUF を利用する利点の 1 つである。

5 PUF の応用

(1) PUF を用いたチャレンジ&レスポンス認証

共通鍵暗号 E_k を PUF による関数 puf_i で置き換えることで、4 節 (1)-ロで述べたチャレンジ＆レスポンス認証を PUF を用いて行うことができる (Gassend [2003])。後の説明を見れば分かるように、これは puf_i を利用者に公開する方法（すなわち Uncontrolled

PUF）である。

PUF による関数 puf_i を用いてチャレンジ＆レスポンス認証を行うには、1 つ問題がある。認証のためには、認証者と被認証者が同じ関数を共有する必要がある。共通鍵暗号を用いる方法 (4 節 (1)-ロ) では、秘密鍵 k を共有することで、その要求を満たしていた。しかし、PUF は複製不可能であるため、認証者と被認証者が同一の PUF を持つことはできない。

以上の課題を解決するために、チャレンジ・レスポンスペアのリストをあらかじめ作成しておき、それを認証者が保管するという方法がある。認証時には、被認証者が返信したレスポンスと、事前に作成しておいたリストを突合することで、相手が正しいかどうかを確認することができる。利用を始める前段階のことを登録フェーズと呼ぶ。一方、リストを用いて相手を確認することを認証フェーズと呼ぶ。詳細な流れを図 7 に示す。

登録フェーズは、PUF を搭載する製品を出荷する前、メーカーの工場などで実行する。このフェーズの目的は、将来の認証で用いるチャレンジとレスポンスのペアを記録することにある。認証者は、ランダムに生成したチャレンジ c_i を PUF に入力し、対応するレスポンス r_i を得る。そのようにして得たペア (c_i, r_i) をデータベースに記録しておく。

認証フェーズでは、登録フェーズで蓄積したペアを用いてチャレンジ＆レスポンス認証を行う。認証者は、記録しておいたペア (c_i, r_i) を選び、c_i を非認証者に送信する。非認証者は、c_i を PUF に入力し、レスポンス r'_i を取得し、その r'_i を認証者へ返信する。認証者は、あらかじめ記録してあった r_i と返信されてきた r'_i を比較し、もし両者が一致すれば認証成功となる。

ただし、レスポンスが短い場合（例: アービター PUF の 1 ビットレスポンス）は、当てずっぽうの返信でも r_i に一致してしまう可能性がある。加えて、レスポンスにはノイズが含まれることがあるため、たとえ真の個体であっても、いくつかのインデックス i で不一致（$r'_i \neq r_i$）となる可能性がある。そこで、相異なるペア (c_i, r_i) による認証を繰り返し、一定以上の割合で $r'_i = r_i$ となるときに認証成功とする方法が用いられる。

リプレイ攻撃がありうるのでペア (c_i, r_i) は再利用できない。そのため、登録フェーズで作成したペア (c_i, r_i) は、認証のたびに消費される。使い切ってしまった場合は、その PUF を破棄するか、もう一度登録フェーズを行う必要がある。そのため、その製品の寿命もしくは登録フェーズを行うことができる間隔を考えて、登録フェーズで作成するペアの個数を決める必要がある。1 度の登録フェーズによって可能になる認証回数が増えるほ

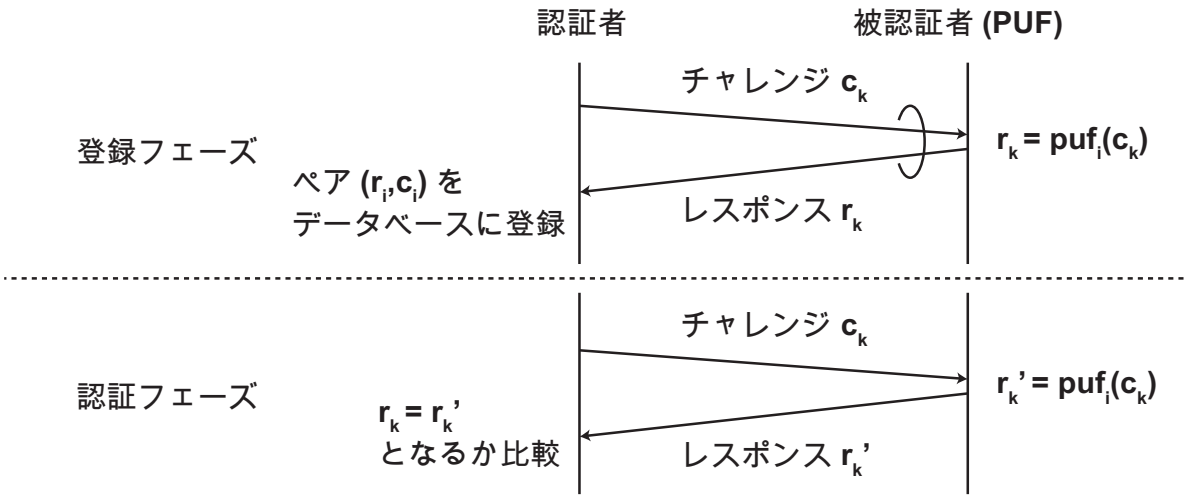

図 7　PUF を用いたチャレンジ＆レスポンス認証

ど、サーバが保存しなくてはいけないデータ量は増加する。

(2)　PUF を用いたセキュア鍵ストレージ

PUF のレスポンスを、その PUF 固有の鍵 k_{PUF} に変換し、暗号の秘密鍵として用いる手法がある。その時、レスポンスに混じるノイズが課題となる。なぜなら、1 ビットでも誤りがあれば暗号は動かないからである。そこで、レスポンスを PUF 鍵 k_{PUF} に変換するファジー抽出器と呼ばれる技法を用いる必要がある。

イ　ファジー抽出器 (Fuzzy Extractor)

ファジー抽出器とは、セキュリティを保ちながら、ノイズ入りデータの誤り訂正を行う技術であるといえる。この技術を用いると、ノイズ入りのデータから秘密鍵を作ることができる。ファジー抽出器は、元々はバイオメトリクス情報から秘密鍵を作るために考案され (Dodis, Reyzin, and Smith [2004])、後に PUF に応用された (Tuyls *et al.* [2006])。

ファジー抽出器の基本的なアイディアは、通信や記録メディアで生じるノイズ（誤り）を訂正するための技術である誤り訂正符号を用いることで、ノイズの影響を打ち消すことにある。ただし、誤り訂正符号は、予め決めておいたデータ（符号語）に入ったノイズしか訂正できないという制限があるため、PUF の出力を直接訂正することはできない。そのギャップを埋めるために、ヘルパーデータという（公開の）補助情報を用いる。加えて、

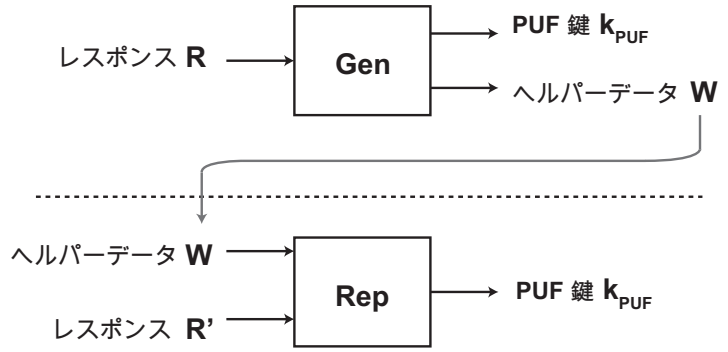

図 8　ファジー抽出器

ヘルパーデータの公開に伴う情報の漏れを補正するための仕組みを組み込んだものがファジー抽出器である。

ファジー抽出器は、レスポンス R から秘密鍵を作るためのアルゴリズム Gen と、ノイズ入りのレスポンス R' から同じ秘密鍵を再生するためのアルゴリズム Rep から構成する（図 8)。なお、PUF のレスポンスが短い場合（例: アービター PUF のレスポンスは 1 ビットであった）は、異なるチャレンジに対応する複数のレスポンスを取得し、それらを連接したものを R として用いる。

Gen は、PUF の個体や、それを搭載する製品を製造した時に、一度だけ行う。アルゴリズムへの入力はレスポンス R であり、その結果、その PUF 固有の鍵である k_{PUF} と、前述の補助情報であるヘルパーデータ W を得る。このヘルパーデータは、後で使うため、どこかに記録しておく必要がある。ただし、ヘルパーデータは公開情報であり、攻撃者に見られても問題無いため、チップの外部にある不揮発メモリ（SD カードなど）に記録しておくのが典型的な保存方法である。

Rep は、PUF 固有の鍵である k_{PUF} を再生するためのアルゴリズムであり、PUF を用いる都度（典型的には、PUF を搭載するチップが起動する都度）に実行される。入力はノイズ入りのレスポンス R' と、あらかじめ保存しておいたヘルパーデータ W である。その結果、鍵 k_{PUF} が出力される。Rep 内部で実行される誤り訂正処理により、レスポンス R' が含まれるノイズが許容範囲であった場合、この鍵は Gen において得たものと同一となる。

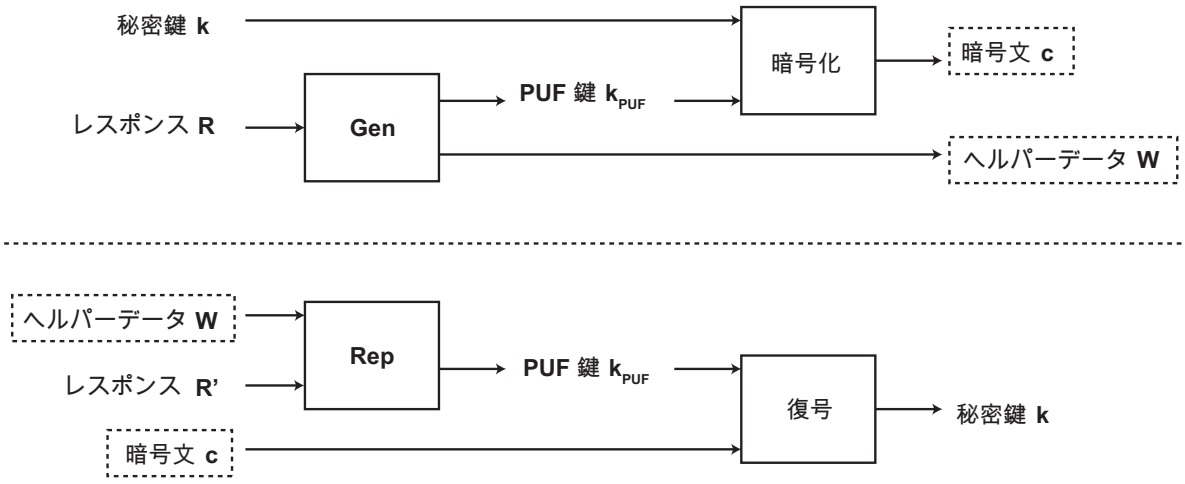

図 9　PUF とファジー抽出器を用いた鍵ストレージ

ロ　PUF を用いた鍵ストレージ

ファジー抽出器を用いると、PUF のレスポンスから、PUF 固有の秘密鍵を生成できることを示した。このような PUF 固有鍵を用いると、秘密鍵を安全に保管するための保管庫が実現できる (Tuyls *et al.* [2006]、Menhorn [2018])。図 9 に、概略図を示す。この方式の目的は、秘密鍵 k を安全に保管することである。

秘密鍵 k の登録は次のように行う。まず、Gen を呼び出して PUF 鍵 k_{PUF} とヘルパーデータ W を得る。続いて、PUF 鍵 k_{PUF} を用いて秘密鍵 k を暗号化して、暗号文 $c = E_{k_{PUF}}(k)$ を得る。以上の結果得られたヘルパーデータ W と暗号文 c は、チップ外部の（安全ではない）不揮発メモリに記録しておく。以上が済んだら、チップは k_{PUF} を消去して停止する。

続いて、次にチップが起動した時の動作を説明する。チップは Rep を呼び出すことで、PUF 鍵 k_{PUF} を再生する。チップは、続いて、保存しておいた暗号文 $c = E_{k_{PUF}}(k)$ を受け取り、これを復号することで秘密鍵 k を復元する。

攻撃者はチップ外部に記録されたヘルパーデータ W と暗号化された鍵 c を盗聴できるが、それらから PUF 鍵 k_{PUF} を得ることはできない。また、PUF 鍵 k_{PUF} は、一度もチップの外に出ることは無いため、k_{PUF} が漏洩することは無い。

(3) PUF を活用する際に留意すべき点

本節では、ここまで述べた PUF の代表的な利用例であるチャレンジ＆レスポンス認証と鍵ストレージに関する利害得失について論じる。

イ チャレンジ＆レスポンス認証

組込機器にネットワーク接続機能を付加することで、機器に付加価値を追加する技術的潮流がある。そこで、限られた計算資源を用いて暗号を実現するための軽量暗号の研究が進められている (CRYPTREC 軽量暗号ワーキンググループ [2017])。しかし、暗号としての複雑さを実現するために必要なハードウェアの資源の量には限界がある。そのように計算能力が制限された用途としては、RFID などが考えられる。

PUF には、暗号よりも少ないハードウェア資源で実現できるものがある。そのような PUF を用いれば、暗号を実装することができないほど計算能力が制限された機器でもチャレンジ＆レスポンス認証を行うことができる。この軽量さが、PUF を暗号の代替として利用する利点である。

一方、5 節 (1) で述べたように、認証に必要なチャレンジ＆レスポンスペアを予めサーバ側に保存しておかなくてはいけない点が欠点である。そのため、製品の寿命において利用するチャレンジ＆レスポンスペア数が制御可能な用途に適している。また、Uncontrolled PUF として利用することから、機械学習攻撃などを考慮する必要がある。

ロ 鍵ストレージ

PUF における鍵ストレージの第一の利点は不揮発メモリを持たないチップでも、安全に秘密鍵を保管できることにある。不揮発メモリを混載したチップの製造は高コストであり、多くのチップは不揮発メモリを持つことができない。そのため、起動後に、外部の不揮発メモリに保存してあった鍵を読み込む必要があり、それが攻撃の標的になる可能性がある。PUF による鍵ストレージを用いれば、そのような攻撃への耐性を獲得することができる。

PUF における鍵ストレージの第二の利点はリバースエンジニアリングへの耐性である。4 節 (3) で述べたように、不揮発性メモリは、電子の有無などの物理的な違いによりデー

タを記録するため、そのような物理状態を読み取ることができる顕微鏡を用いると、中身のデータを読み出せてしまうことがある (Courbon, Skorobogatov, and Woods [2016])。それに対し PUF は、動作時にしか鍵が存在しないため、電源オフ時のチップを解析しても鍵が漏洩することは無い。

一方、鍵ストレージとして利用する際の欠点は、もはや軽量とは言えないことである。なぜなら、PUF そのものに加えて、ファジー抽出器に必要な誤り訂正符号やハッシュ関数なども実装しなくてはならないためである。また、PUF を用いて復元した鍵を用いるための暗号実装も必要である。

以上より、チップ内に安全な鍵保管用の不揮発メモリを搭載することが難しいプラットフォームにおいて、安全な鍵保管庫を実現することが、PUF による鍵ストレージの典型的な利用例であると言える。適用先としては、スマートフォンなどのメインのプロセッサである System on Chip (SoC) や、特定用途向けのカスタム IC (標準ロジックプロセスで開発した ASIC)、FPGA などがある。加えて、不揮発メモリを混載することができる IC カードなどの機器においても、優れたリバースエンジニアリング耐性のために利用することが考えられる。

一方、特定の攻撃（静的リバースエンジニアリング）にしか効果が無いことに注意が必要である。システムのセキュリティは最弱点で決まるため、暗号実装を狙った物理攻撃などは、さらに別の対策法で防ぐ必要がある。

6 PUF への攻撃

PUF に限らず、実装に関わるセキュリティでは、攻撃者の払うことができる攻撃コスト（攻撃に利用する機器の価格や、攻撃者が有する知識・スキルの度合い）が上がるほど、攻撃の成功率が上がる。対策法も同様であり、回路面積や時間をかけるほど強くすることができるが、そのためにはコストが発生する。そこで、脅威を見極め、リスクが受容できるレベルになるように対策法を選んでいくということをセキュリティの設計において行う。正しい判断をするためには、潜在的にどのような脅威があり、それがどれくらいの難易度なのかを知る必要がある。

以上の背景から、セキュリティの研究分野では、攻撃の研究を行うことの価値が認められている。PUF も例外ではなく、これまでにいくつもの攻撃が提案されてきた。本章で

は、いくつかの重要な攻撃法について概説する。

(1) 機械学習攻撃

チャレンジ＆レスポンス認証を安全に行うには、過去に使ったチャレンジ・レスポンスのペアから、未使用のペアが予測できないという条件が必要である。言い換えれば、未使用のチャレンジに対応するレスポンスを予測しようという試みは、PUF への攻撃となる。

これは、機械学習における教師あり学習の枠組みで考えることができる。すなわち、puf_i から取得したチャレンジとレスポンスのペアを訓練データとして用いて機械学習を行うのである。その結果得られる関数 F は、訓練データを採取した個体 puf_i の近似となっている。もし、学習に成功すれば、未知のチャレンジ c に対しても、高い確率で $\mathsf{puf}_i(c) = F(c)$ が成立する。すなわち、F を puf_i のクローンとして使うことができる。

以上のように、機械学習アルゴリズムを用いて PUF のクローンを作る攻撃を機械学習攻撃と呼ぶ (Rührmair *et al.* [2013])。これまでに、アービター PUF を含む複数の PUF の実装に対し、機械学習攻撃が成功することが知られている[*3]。PUF をチャレンジ＆レスポンス認証に利用する時は、機械学習攻撃への耐性を考慮しなくてはいけない。

機械学習攻撃に対策するためには、規則性が無くなるように PUF の作り方を工夫するというアプローチがありうる。特に、アービター PUF は非常に規則性が高くて機械学習攻撃に弱いことが知られているため、複数のアービター PUF を束ねて複雑性を増す方式などが研究されている。加えて、機械学習攻撃は Uncontrolled PUF を対象したものであるため、Controlled PUF として利用することでも攻撃を避けることができる。

(2) PUF の中身を盗み見る攻撃

PUF が安全になるためには、攻撃者は PUF の内部は見ることができないという（暗黙の）前提条件が必要である。それに対し、チップが生じる消費電力や漏洩電磁波を観測してチップ内部を覗き見ることで、(暗黙の）前提条件を覆す攻撃がありうる。

3 節 (1) で述べたように、アービター PUF は、一対の遅延線において、どちらを速く信号が伝わるかが個体差を持つことを利用していた。信号が伝わるということはす

[*3] 同様の攻撃を暗号アルゴリズムに適用することもできるが、成功した例は知られていない。

なわち、電流が流れるということである。よって、チップに出入りする電流を計測器で観察すると、チップの内部でどのような信号伝搬が生じたかをある程度知ることができる。そこで、アービター PUF の消費電力を観察することで、内部でどのような信号の伝搬が生じたかの情報を盗み、その情報を元にクローンを作成する攻撃が存在する (Rührmair *et al.* [2014])。このように物理的に取得した情報を利用する攻撃をサイドチャネル攻撃と呼ぶ (崎山・菅原・李 [2019])。

アービター PUF だけではなく、別の種の PUF でも同様のサイドチャネル攻撃が知られている (Schuster [2010])。また、PUF そのものではなく、ファジー抽出器を対象にしたサイドチャネル攻撃も存在する (Merli *et al.* [2011])。すなわち、ファジー抽出器の内部で用いる誤り訂正やハッシュ関数の計算において生じる消費電力や漏洩電磁波を解析することで、PUF の出力や PUF 鍵の情報を盗み取ることができる。

サイドチャネル攻撃に対抗するには、消費電力や電磁波として漏洩する情報と、チップ内部で扱っているデータの関係を断ち切る必要がある。そのための方法として、乱数を加えて計算方法をランダム化する方法などが知られている (崎山・菅原・李 [2019])。

(3) PUF の挙動を操作する攻撃

前節で述べた PUF の挙動を盗み見る攻撃に対応して、物理的な手段により PUF の出力を操作する攻撃がある。そのような攻撃の例として、SRAM PUF における残留データを用いた攻撃がある (Zeitouni *et al.* [2016])。

前提条件として、ユーザ（攻撃者）は、対象の SRAM に自由にデータが書き込めるものとする。攻撃者は、SRAM の中身をゼロで埋めた後、電源を一瞬だけ切って再起動する。SRAM に記録されたデータは、電源を喪失後も、ごく短時間であれば残留している。そのため、電源喪失がごく短ければ、SRAM の初期値は変わらずゼロのままである。そこから電源を遮断する時間を長くしていくと、PUF のレスポンスはゼロから本来の値へと徐々に近づいていく。すなわち、攻撃者は、PUF の出力を（ある程度）操作できる。

詳細は省略するが、この性質を巧みに使うと、本来は秘密にしなくてはいけない PUF の出力（すなわち SRAM の初期値）を復元することができる [*4]。この攻撃を行うには、「PUF に利用する SRAM にユーザが書き込める」という前提条件が必要である。そのた

[*4] 詳細は、文献 (Zeitouni *et al.* [2016]) のアルゴリズム 1 を参照されたい。

め、PUF に利用する SRAM は汎用メモリとは分けるような方法で対策をすることができる (Zeitouni *et al.* [2016])。

(4) 動的リバースエンジニアリング

PUF は、電源投入後にしか鍵が顕在化しないため、電源オフ時に行う静的リバースエンジニアリングに耐性を持つと述べた（4 節 (3) を参照)。それに対して、動作しているチップに電極を当て、再生成された PUF 鍵を直接読み取るような攻撃がある。このような攻撃を動的リバースエンジニアリングと呼ぶ。

電極を当てるような攻撃は、PUF に耐タンパー性（2 節 (5)-ホ）があれば問題にならない。また、動的リバースエンジニアリングを受けているとき、チップは正常に動作しているため、電極の接触をセンサを用いて検出するような対策法を取ることもできる (Anderson [2008])。一方、これをさらに迂回する方法として、非接触で行うことができる動的リバースエンジニアリング法が研究されており、PUF を攻撃した研究事例が存在する (Lohrke *et al.* [2016])。

この方法は、半導体チップの不具合を解析するために使われるレーザー電圧プロービングという計測法を用いる。これは、対象のチップにレーザーを当て、反射光の強さを計測することで、レーザーを照射した箇所の電圧を計測する計測法である。ある秘密のビットが保存されたメモリにレーザーを当てれば、そのデータを外部から読み取ることができる。光により非接触で計測を行うため、電極を検知するセンサで見つけることができない。この攻撃への対策法として、回路の挙動をランダム化して計測を妨害する方法が提案されている (Lohrke *et al.* [2016])。

7 おわりに

本稿では、製造ばらつきを用いてチップの個体識別を行う技術である PUF について、特に利用者の観点で解説を行った。セキュリティ設計では、限られたコストの制限の中で、想定される攻撃を脅威が大きい順に対策を選んでいくということを行う。ある利用シーンで PUF が有効であるかどうかは、セキュリティ設計の細部に宿っている。そうした中で、考慮すべきと考えられる脅威に対して PUF の活用を検討する際に、本稿で説明した PUF の特性や利点・欠点に関する考え方が参考になると考えられる。

参考文献

宇根正志・松本　勉、「生体認証システムにおける脆弱性について：身体的特徴の偽造に関する脆弱性を中心に」、『金融研究』、第 24 巻第 2 号、日本銀行金融研究所、2005 年、35～83 頁

崎山一男・菅原　健・李　陽、『暗号ハードウェアのセキュリティ』、コロナ社、2019 年

凸版印刷株式会社、「凸版印刷、世界初、半導体の個体差を用いた PUF 技術搭載 IC タグ『SMARTICS-V』による真贋判定サービスの提供を開始～NFC 対応スマートフォンで、生活者自身による製品の真贋判定が可能～」、凸版印刷株式会社、2013 年

――――、「凸版印刷が提供する、PUF 技術搭載 IC タグによる真贋判定サービスがタニタの海外向けポケッタブルスケールで採用～NFC 対応スマートフォンで、生活者自身による製品の真贋判定が可能～」、凸版印刷株式会社、2015 年

鳥居直哉・山本　大・武仲正彦・松本　勉、「FIB 加工とプローブ測定に対する RS ラッチの挙動(I)」、2015 年暗号と情報セキュリティ・シンポジウム発表論文、電子情報通信学会、2015 年

三菱電機株式会社・立命館大学・科学技術振興機構、「『IoT 時代に向けたセキュリティー技術』を開発」、開発 No. 1504、三菱電機株式会社、2015 年

山本　大・鳥居直哉・武仲正彦・松本　勉、「FIB 加工とプローブ測定に対する RS ラッチの挙動(II)」、2015 年暗号と情報セキュリティ・シンポジウム発表論文、電子情報通信学会、2015 年

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）」、情報通信研究機構・情報処理推進機構、2017 年

Altera Corporation, “Altera Partners with Intrinsic-ID to Develop World’s Most Secure High-End FPGA,” Intel, 2015 (available at: https://newsroom.intel.com/news-releases/altera-partners-intrinsic-id-develop-worlds-secure-high-end-fpga/#gs.oagadm、2020 年 2 月 21 日).

Anderson, Ross, “Chapter 16 Physical Tamper Resistance,” *Security Engineering: A Guide to Building Dependable Distributed Systems (Second Edition)*, Wiley, 2008, pp. 483-522.

Costan, Victor, and Srinivas Devadas, “Intel SGX Explained,” Cryptology ePrint Archive, Report 2016/086, International Association for Cryptologic Research, 2016.

Courbon, Franck, Sergei Skorobogatov, and Christopher Woods, “Reverse Engineering Flash EEPROM Memories Using Scanning Electron Microscopy,” in Kerstin

Lemke-Rust and Michael Tunstall, eds, *Smart Card Research and Advanced Applications*, Springer-Verlag, 2016, pp. 57-72.

Dodis, Yevgeniy, Leonid Reyzin, and Adam Smith, "Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy Data," *Proceedings of EUROCRYPT 2004*, *Lecture Notes in Computer Science*, 3027, Springer-Verlag, 2004, pp. 523-540.

Gassend, Blaise Laurent Patrick, "Physical Random Functions," Master's Thesis, Massachusetts Institute of Technology, 2003.

Guajardo, Jorge, Sandeep S. Kumar, Geert-Jan Schrijen, and Pim Tuyls, "FPGA Intrinsic PUFs and Their Use for IP Protection," *Proceedings of International Workshop on Cryptographic Hardware and Embedded Systems (CHES) 2007*, *Lecture Notes in Computer Science*, 4727, Springer-Verlag, 2007, pp. 63-80.

Holcomb, Daniel E., Wayne Burleson, and Kevin Fu, "Power-Up SRAM State as an Identifying Fingerprint and Source of True Random Numbers," *IEEE Transactions on Computers*, 58(9), IEEE, 2009, pp. 1198-1210.

International Organization for Standardization and International Electrotechnical Commission, "ISO/IEC DIS 20897-1 Information Technology — Information Security, Cybersecurity and Privacy Protection — Security Requirements and Test Methods for Physically Unclonable Functions for Generating Non-Stored Security Parameters — Part 1: Security Requirements," International Organization for Standardization and International Electrotechnical Commission, 2019 (available at: https://www.iso.org/standard/76353.html、2020 年 2 月 21 日).

Jones, Scott, "How Unclonable, Turnkey Embedded Security Protects Designs from the Ground Up," Maxim Integrated Products, Inc., 2017.

————, "A Reverse-Engineering Assessment of a Secure Authenticator with PUF Technology," Maxim Integrated Products, Inc., 2018.

Kerst, Uwe, Rudolf Schlangen, Alexander Kabakow, Erwan Le Roy, Ted R. Lundquist, and Siegfied Pauthner, "Impact of Back Side Circuit Edit on Active Device Performance in Bulk Silicon ICs," *Proceedings of IEEE International Conference on Test 2005*, IEEE, 2005, pp. 1236-1244.

Lim, Daihyun, "Extracting Secret Keys from Integrated Circuits," Master's Thesis, Massachusetts Institute of Technology, 2004.

Lohrke, Heiko, Shahin Tajik, Christian Boit, and Jean-Pierre Seifert, "No Place to Hide: Contactless Probing of Secret Data on FPGAs," *Proceedings of International Workshop on Cryptographic Hardware and Embedded Systems (CHES) 2016*, *Lecture Notes in Computer Science*, 9813, Springer-Verlag, 2016, pp. 147-167.

Maes, Roel, *Physically Unclonable Functions: Constructions, Properties and Applications*, Springer-Verlag, 2013.

Menhorn, Nathan, "External Secure Storage Using the PUF," XAPP1333 (v1.0), Xilinx, Inc., 2018 (available at: https://www.xilinx.com/support/documentation/application_notes/xapp1333-external-storage-puf.pdf、2020 年 2 月 21 日).

Merli, Dominik, Dieter Schuster, Frederic Stumpf, and Georg Sigl, "Side-Channel Analysis of PUFs and Fuzzy Extractors," *Proceedings of International Conference on Trust and Trustworthy Computing (TRUST) 2011*, *Lecture Notes in Computer Science*, 6740, Springer-Verlag, 2011, pp. 33-47.

Microsemi Corporation, "Using SRAM PUF System Service in SmartFusion2 - Libero SoC v11.7," Application Note AC434, Microsemi Corporation, 2016.

NXP Semiconductors, "NXP Delivers Enhanced Security Solution to Protect Personal Data for Payment and eGovernment Services," NXP Semiconductors, 2016 (available at: https://media.nxp.com/news-releases/news-release-details/nxp-delivers-enhanced-security-solution-protect-personal-data/、2020 年 2 月 21 日).

Pappu, Ravikanth Srinivasa, "Physical One-Way Functions," Ph.D. Thesis, Massachusetts Institute of Technology, 2001.

Rührmair, Ulrich, Jan Sölter, Frank Sehnke, Xiaolin Xu, Ahmed Mahmoud, Vera Stoyanova, Gideon Dror, Jürgen Schmidhuber, Wayne Burleson, and Srinivas Devadas, "PUF Modeling Attacks on Simulated and Silicon Data," *IEEE Transactions on Information Forensics and Security*, 8(11), IEEE, 2013, pp. 1876-1891.

————, Xiaolin Xu, Jan Sölter, Ahmed Mahmoud, Mehrdad Majzoobi, Farinaz Koushanfar, and Wayne Burleson, "Efficient Power and Timing Side Channels for Physical Unclonable Functions," *Proceedings of International Workshop on Cryptographic Hardware and Embedded Systems (CHES) 2014*, *Lecture Notes in Computer Science*, 8731, Springer-Verlag, 2014, pp. 476-492.

Schuster, Dieter, "Side Channel Analysis of Physical Unclonable Functions (PUFs)," Diploma Thesis, Technische Universität München, 2010.

Torrance, Randy, and Dick James, "The State-of-the-Art in Semiconductor Reverse Engineering," *Proceedings of Design Automation Conference (DAC) 2011*, Association for Computing Machinery, 2011, pp. 333-338.

Tuyls, Pim, Geert-Jan Schrijen, Boris Škorić, Jan van Geloven, Nynke Verhaegh, and Rob Wolters, "Read-Proof Hardware from Protective Coatings," *Proceedings of International Workshop on Cryptographic Hardware and Embedded Systems (CHES) 2006*, *Lecture Notes in Computer Science*, 4249, Springer-Verlag, 2006,

pp. 369-383.

Xilinx, Inc., “Xilinx Addresses Rigorous Security Demands at Fifth Annual Working Group for Broad Range of Applications,” Xilinx, Inc., 2016 (available at: https://www.xilinx.com/news/press/2016/xilinx-addresses-rigorous-security-demands-at-fifth-annual-working-group-for-broad-range-of-applications.html 、2020 年 2 月 21 日).

Zeitouni, Shaza, Yossef Oren, Christian Wachsmann, Patrick Koeberl, and Ahmad-Reza Sadeghi, “Remanence Decay Side-Channel: The PUF Case,” *IEEE Transactions on Information Forensics and Security*, 11(6), IEEE, 2016, pp. 1106-1116.