Horizontal Side-Channel Vulnerabilities of Post-Quantum Key Exchange Protocols

Aydin Aysu, Youssef Tobah, Mohit Tiwari, Andreas Gerstlauer, and Michael Orshansky
Department of Electrical and Computer Engineering
The University of Texas at Austin, Austin, TX, USA
{aydinay,ytobah,tiwari,gerst,orshansky}@utexas.edu

***Abstract*—Key exchange protocols establish a secret key to confidentially communicate digital information over public channels. Lattice-based key exchange protocols are a promising alternative for next-generation applications due to their quantum-cryptanalysis resistance and implementation efficiency. While these constructions rely on the theory of quantum-resistant lattice problems, their practical implementations have shown vulnerability against side-channel attacks in the context of public-key encryption or digital signatures. Applying such attacks on key exchange protocols is, however, much more challenging because the secret key changes after each execution of the protocol, limiting the side-channel adversary to a single measurement.**

In this paper, we demonstrate the first successful power side-channel attack on lattice-based key exchange protocols. The attack targets the hardware implementation of matrix and polynomial multiplication used in these protocols. The crux of our idea is to apply a horizontal attack that makes hypothesis on several intermediate values within a single execution all relating to the same secret and to combine their correlations for accurately estimating the secret key. We illustrate that the design of key exchange protocols combined with the nature of lattice arithmetic enables our attack. Since a straightforward attack suffers from false positives, we demonstrate a novel procedure to recover the key by following the sequence of intermediate updates during multiplication.

We analyzed two key exchange protocols, `NewHope` (USENIX'16) and `Frodo` (CCS'16), and show that their implementations can be vulnerable to our attack. We test the effectiveness of the proposed attack using concrete parameters of these protocols on a physical platform with real measurements. On a SAKURA-G FPGA Board, we show that the proposed attack can estimate the entire secret key from a single power measurement with over 99% success rate.

I. Introduction

Key exchange protocols enable agreement on a secret key between two parties. Traditional key exchange protocols such as Diffie-Hellman rely on the difficulty of solving the discrete-logarithm problem over various groups, which are widely regarded to be infeasible for large numbers. These problems, however, can be solved in polynomial-time with quantum algorithms [1, 2], generating a significant interest in alternative key exchange protocols that will future-proof security systems [3–5] in case of a breakthrough in quantum computing technology. Lattice-based cryptography provides a wide array of such constructions that are resistant to quantum attacks. Among lattice solutions, proposals relying on Learning-With-Errors (LWE) [6] and Ring-Learning-With-Errors (R-LWE) [7] are especially favored due to their implementation efficiency [8, 9].

While these constructions rely on the theory of quantum-resistant lattice problems, their practical implementations have shown vulnerability against power side-channel attacks in the context of public-key encryption or digital signatures [10–20]. These attacks find the secret key by exploiting the reflection of key-dependent computations on dynamic power consumption. To extract this dependence, prior attacks on lattice problems use repeated computations performed with the same secret key.

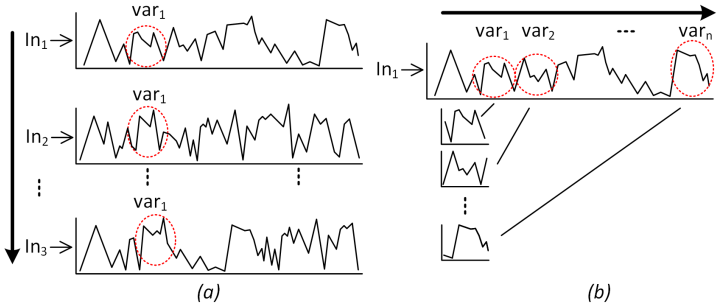

Fig. 1. Vertical DPA (a) targets a single intermediate computation and seeks correlation across multiple traces each using a distinct input, while Horizontal DPA (b) targets multiple intermediate computations within a trace and seeks a correlation among them.

To that end, differential power analysis (DPA)[21] is a powerful side-channel attack that works with a divide-and-conquer principle: It makes an estimation on distinct parts of the key (called *sub-key*) and checks those estimations through multiple tests. These tests are required to remove the noise and reveal the underlying correlation between the sub-key and the power measurement. Once the target sub-key is derived, the adversary can attack the next sub-key by using the same set of measurements. Applying DPA is, however, signficantly more challenging on key exchange protocols because these protocols, unlike public-key decryption or digital signatures, work with *ephemeral* secrets: each invocation of the protocol will process a unique value that will result in a new, distinct secret key. Therefore, the adversary is limited to a single power measurement.

Figure 1(a) illustrates the classic DPA, known as *Vertical* DPA. In this attack, the adversary performs a single test on the power trace and collects multiple measurements (each with a different input) to extract sub-key correlations from noise. The intermediate computations on the variable var_1 is selected as the target because its value depends on the public input and a sub-key. *Horizontal* DPA [22], Figure 1(b), by contrast, performs multiple tests by targeting different computations and combines them to extract the sub-key from a single power measurement. Horizontal DPA thus focuses on intermediate computations that use variables var_i $(i \in 1, 2, ..., n)$ that all depend on the public input and the same sub-key. The main challenge of applying Horizontal DPA is in finding multiple tests to be performed within a single computation that will leak the same sub-key. Such approaches have been successfully applied to modular exponentiation [22] and elliptic curve multiplication [23], but lattice arithmetic is based on fundamentally different computations for which no successful horizontal side-channel attack has been shown so far.

In this paper, we demonstrate, for the first time, that lattice arithmetic used in key exchange protocols is vulnerable to Horizontal DPA attacks. Specifically, we demonstrate that the

matrix and polynomial multiplication used in these protocols have a large number of intermediate computations that depend on the same sub-key. A straightforward attack on these operations, however, causes false positives because similar sub-keys will produce similar multiplication output (e.g. sub-keys of '1' and '2' will generate the same values shifted by one binary digit). We show that an adversary can address this limitation and still succeed by using a novel attack that targets intermediate state updates of these multiplications starting from the first sub-key to successively recover a chain of subsequent sub-keys. This attack effectively removes false positives after the first sub-key.

We analyze the feasibility of the proposed attack on two state-of-the-art key exchange protocols: `Frodo` [24] and `New Hope` [25]. While `Frodo` performs matrix multiplication, `New Hope` uses polynomial multiplication. Both protocols perform multiplications between a secret ephemeral value and a public input, making them susceptible to the proposed Horizontal DPA attack. Once the ephemeral secret is discovered with the proposed attack, we show that an adversary can recover the exchanged secret key in these protocols. Moreover, we show that both parties engaged in the protocol are vulnerable to our attack.

To test the practical validity of our attacks, we design the matrix and polynomial multiplication in hardware using the specific parameter sets of the two protocols, and we apply the attack using real power measurements taken from a SAKURA-G FPGA platform. We validate that the number of horizontal tests, which is 1024 for `NewHope` and 752 for `Frodo`, is sufficient to extract the key. The results show that the proposed attack is able to recover secret keys with over 99% probability.

The rest of the paper is organized as follows. Section II describes the threat model of DPA. Section III provides a background on power side-channels and discusses the related work. Section IV analyzes the protocols under attack and gives a high-level description of the proposed attack. Section V shows the details of hardware architectures used in analysis and evaluates the attack efficiency with real measurements. Section VI discusses the limitations and countermeasures of our attack. Section VII concludes the paper.

II. Threat Model

Our threat model follows the typical power side-channel model of prior work [10–21]. The adversary in DPA attacks has physical access to the device and can read power measurements as the device computes the cryptographic routine. Therefore, the adversary is equipped with a reasonable measurement setup that can obtain power consumption information several times within a clock period. In our experimental setup, which targets a device operating in the MHz frequency range, a low-end digital oscilloscope with passive probes is sufficient to apply the attack.

We assume that the attacker in our model can record a single power measurement of the entire application (e.g. HTTPS/TLS) that uses the key exchange protocol. We also assume that the DPA adversary knows details of hardware architecture, such as its data flow, parallelization and pipelining. Therefore, unless there is a specific countermeasure to obfuscate the timing of the underlying operations, the adversary can estimate when the targeted computations will *likely* occur and apply the attack around those clock cycles. Engineering aspects of locating cryptographic routines (and specific operations inside those routines) among other applications within a system has been described in prior work, both in the context of physical [26] and digital side-channels [27, 28]. We do not cover this aspect in the scope of this work and assume that the adversary can locate around which clock cycle to start the DPA analysis using prior techniques. Note that the adversary does not have to know the exact timing information of side-channel leaks within the clock period; it can exhaustively evaluate the attack on all sampling points within a period to identify where the side-channel leak occurs.

We follow the assumption that the DPA adversary can eavesdrop on the communication to record the public messages that are exchanged between two parties. We conduct the DPA attack on a hardware implementation with a fixed execution time and with no conditional checks based on the value of sub-keys. Therefore, unlike software implementations of the same protocols, the attacker cannot use timing side-channels or information leaks of conditional branches in the analysis.

III. Background and Related Work

Although power-based side-channel attacks have been an active area of research for the last two decades [21], analysis of lattice-based constructions is relatively limited [10–20]. These attacks consider other applications with lattices such as public-key decryption or digital signatures. None of the existing proposals are applicable to our problem where the attacker has to extract the entire key from a single power measurement on a hardware implementation that has no conditional branch leaks or large power variations (exploitable e.g. SPA attacks) based on secret key values. Therefore, to our best knowledge, our work is the first power-based attack on a hardware designed for lattice-based key exchange protocols.

We can broadly categorize power side-channel attacks into three groups: simple power analysis (SPA), DPA, and template attacks. SPA is based on purely visual observations to capture large variations related to secret key with a few traces. By contrast, DPA extracts small correlations by evaluating many traces with statistical methods. We refer to attacks that require a pre-characterization of the device as template attacks [29]. In our classification, we consider attacks using electromagnetic radiation synonymous to power-based attacks as they use the same principles, and we therefore review EM attacks in our related work. In the following, we evaluate the prior work on lattice-based side-channels in detail and discuss why we need a new approach to attack key exchange protocols in hardware.

SPA Attacks: Park *et al.* proposed an SPA attack on the polynomial mulitplication of the R-LWE-based public-key decryption process, exploiting the SPA power leaks of an implementation on 8-bit AVR microcontrollers. This attack does not succeed in our scenario for two reasons. First, we design a hardware without this vulnerability and second, the attack requires multiple measurements of the same decryption key to extract the full key.

DPA Attacks: Several DPA attacks have been proposed on the multiplications of lattice-based public-key decryption [11–17]. However, all of these attacks are instances of *Vertical* DPA, focusing on one intermediate variable within the power trace and extracting the key using many power traces. Therefore, their attack procedure cannot be used in our setting.

Template Attacks: Primas *et al.* recently proposed a single-trace template attack on the Number Theoretic Transform (NTT) based polynomial multiplication of an R-LWE decryption procedure [18]. This attack, similar to the majority of template attacks,

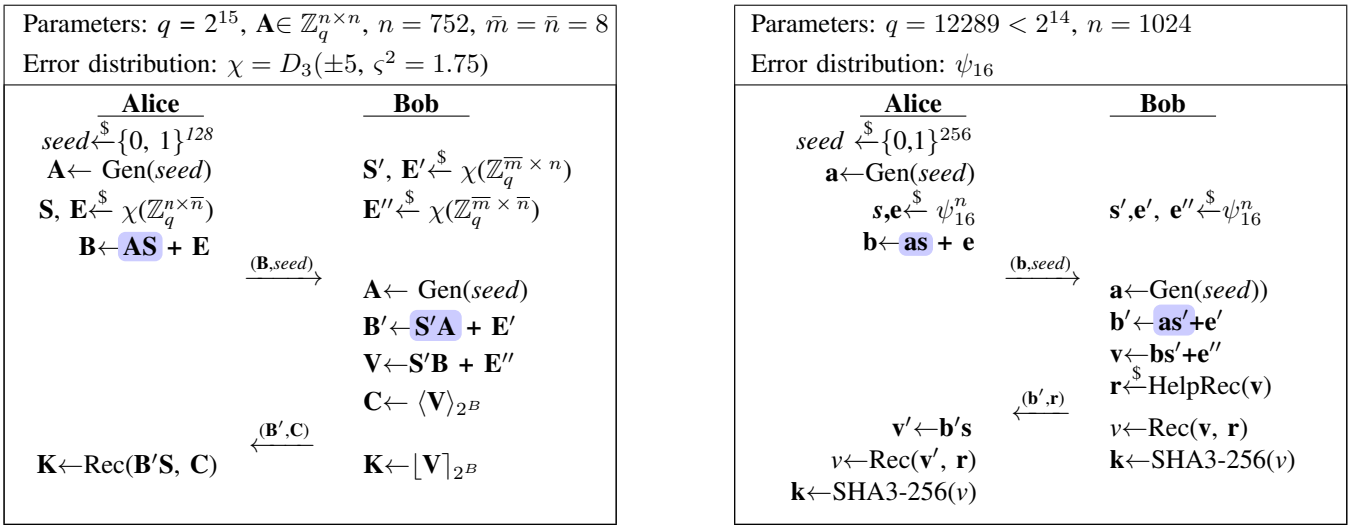

Fig. 2. `Frodo` (a) and `NewHope` (b) key exchange protocol and the parameters we select for their instantiations. We highlight the target of DPA attack; in both protocols, the ephemeral secrets ($\mathbf{S}$, $\mathbf{S}'$ or $\mathbf{s}$, $\mathbf{s}'$) are multiplied with a public value ($\mathbf{A}$ or $\mathbf{a}$).

requires a precise power-behavior characterization of the target device. Therefore, it assumes that the adversary can configure the target device for each possible sub-key guess and collect a large set (consisting of 100 million samples) of traces to build templates. These templates are later matched for a given power trace and secret key is estimated with a maximum likelihood test. This attack is proposed on a software implementation of R-LWE decryption and it exploits several ARM ISA specific vulnerabilities, including a *timing* side-channel of modular division instructions. Our attack, on the one hand, complements their analysis since we target matrix multiplication and regular polynomial multiplication, and since we analyze hardware implementations that have no timing side-channels. On the other hand, our attack enables a simpler, more practical recovery of secret keys via DPA, which does not require costly pre-characterization or the ability to configure the target device with different keys. Pessl *et al.* demonstrated a template attack on a software implementation of lattice signature scheme [19]. This attack estimates the control flow (i.e. data dependent branch operations) and intermediate variables of lattice sampling. Then, using *multiple* measurements and associated signatures, it extracts the secret key with lattice-reduction techniques. The requirement of multiple measurements again makes this attack infeasible in our scenario.

Combined Attacks: Espitau *et al.* shows two power side-channel on a software implementation of a lattice signature scheme [20]. The first one is an SPA attack on an 8-bit AVR microcontroller. The attack targets the conditional branching of rejection sampling and combines several measurements to extract the key. This attack is not possible in our scenario since it requires multiple measurements and since we are interested in hardware without conditional branch leaks. The second attack focuses on the *sparse* polynomial multiplication that is implemented as a sequence of repeated shifted additions. Their attack first estimates the indexes of non-zero elements by applying an SPA on the conditional branches, and then performs a DPA attack on those to reveal their sign. Using this information, the attacker applies an Integer Linear Programming to extract the actual values of those coefficients. This attack is not possible in our scenario because our target protocols use matrix and non-sparse multiplications and because we are interested in hardware without conditional branch leaks.

In summary, template attacks using conditional branch leaks or timing side-channels and SPA are not applicable on proper hardware-only solutions. Template attacks also require a pre-characterization of the device that is not possible in every threat model. Therefore, DPA is a suitable and effective technique but it can only work in our scenario if the adversary can find multiple tests within a single trace. This analysis leads us to applying a Horizontal DPA attack.

IV. Horizontal Side-channel Analysis of Post-Quantum Key Exchange Protocols

In this section, we first summarize the key exchange protocols that we attack, and we highlight the target operation of our Horizontal DPA. Then, we give a high-level description of our attack and how to address its challenges.

A. Post-quantum Key Exchange Protocols

Key exchange protocols establish a unique, symmetric key between two parties. Both parties in these protocols use an ephemeral secret to generate some public information and share it with the other party to successfully agree on the same key. The protocol is designed in such a way that the adversary who eavesdrops on the public information cannot capture the established key or recover the ephemeral secret values.

Figure 2 gives the description of `Frodo` (a) and `NewHope` (b), respectively. The main difference between the two is that while `Frodo` relies on the LWE problem[1], `NewHope` uses R-LWE. Therefore, while `Frodo` works with matrices (denoted with capital Latin letters), `NewHope` processes polynomials (denoted with lowercase Latin letters).

In both protocols, Alice starts by generating a public parameter ($\mathbf{A}$, $\mathbf{a}$) from a random *seed*, sampling the secret error terms ($\mathbf{S}$ and $\mathbf{E}$, $\mathbf{s}$ and $\mathbf{e}$) from a specific distribtuion, and computing the message ($\mathbf{B}$, $\mathbf{b}$), which is sent, together with the *seed* to Bob. Given $\mathbf{B}$ and $\mathbf{A}$, it should not be feasible, either with a conventional or a quantum computer, to compute the values of small error terms $\mathbf{S}$ and $\mathbf{E}$ due to the LWE problem [6] (the

[1] `Frodo` deliberately picks LWE over R-LWE in case there is a future reduction in the assumed difficulty of underlying R-LWE problems.

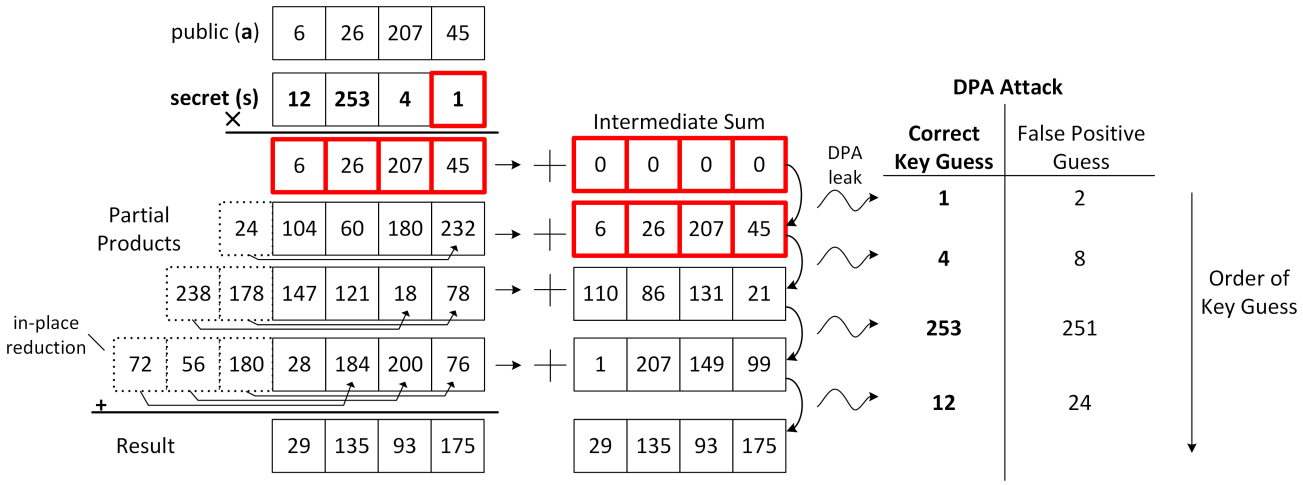

Fig. 3. An example of Horizontal DPA attack on the polynomial multiplication; same concepts also holds for matrix multiplication

analogous holds for NewHope polynomials **a**, **b**, **s**, and **e** with the R-LWE problem [7]). Bob then generates his share of the secret key ($\mathbf{B}'$, $\mathbf{b}'$) by using his ephemeral error samples ($\mathbf{S}'$ and $\mathbf{E}'$, $\mathbf{s}'$ and $\mathbf{e}'$) and sends it to Alice. Alice and Bob now can agree on the secret value by evaluating each others' terms with their ephemeral secrets. Since Alice and Bob will achieve a similar but noisy term, they can *reconcile* by recovering from this noise through a thresholding scheme. Interested readers can refer to [25] and [24] for details of these protocols.

Frodo has several parameter options depending on the desired security level. For our analysis, the parameters for the Frodo are selected from the "Recommended" scheme. This set uses matrices of sizes $n \times n$, $n \times \bar{n}$, $\bar{m} \times \bar{n}$, and $\bar{m} \times \bar{n}$, where n, $\bar{n}$, and $\bar{m}$ are, respectively, 752, 8, and 8 with integer elements modulo 2^{15}. The error distribution is a Rényi divergence approximation to a rounded continuous Gaussian with variance ς^2=1.75, which is centered at 0 with a tail cut at ± 5. NewHope has a fixed parameter set that operates in the ring $\mathcal{R}_q = \mathbb{Z}_q[x]/\langle x^n+1\rangle$, which uses polynomials of degree 1023 with integer coefficients modulo 12289 and a binomial distribution for small (error) polynomials with integers coefficients from -16 to 16, centered at 0.

B. Protocol Operations Under Attack

Figure 2 highlights the possible points of attacks for our Horizontal DPA. We focus on the multiplication of public value **A** (resp. **a**) with the ephemeral secret **S** or $\mathbf{S}'$ (resp. **s** or $\mathbf{s}'$) in Frodo (resp. NewHope) protocols. The target operation for Frodo is therefore a matrix multiplication of the public value **A** with the ephemeral secret **S** or $\mathbf{S}'$. This is multiplication of a size 752 × 752 matrix with a size 752 × 8 or 8 × 752 matrix. For the case of NewHope, the DPA target is the multiplication of two polynomials with degree 1023.

Note that both Alice and Bob can be attacked using this approach. Once the adversary recovers the ephemeral secret, extracting the exchanged secret key becomes trivial. For Frodo, an adversary attacking Alice can compute $\mathbf{K}$=Rec($\mathbf{B}'\mathbf{S}$,**C**) and an adversary attacking Bob can first recover **V** (without the small error of $\mathbf{E}''$), and then generate $\mathbf{K}=\lfloor \mathbf{V} \rceil_{2^B}$ since $\lfloor \mathbf{V} \rceil_{2^B} = \lfloor \mathbf{V} - \mathbf{E}'' \rceil_{2^B}$. The same principle also holds for the NewHope protocol. Once the adversary captures the polynomial **s** or $\mathbf{s}'$, it can recover the secret key **k**.

C. The Crux of Horizontal DPA on Multiplication

Figure 3 illustrates the main idea behind our Horizontal DPA attack on a polynomial multiplication; the same principle applies to matrix multiplication as well. For simplicity, this example uses a polynomial of degree 3 and with coefficients modulo 2^8. The figure reflects the schoolbook polynomial multiplication of a public polynomial (e.g. **a**) with an ephemeral secret polynomial (e.g. **s**). The polynomial multiplication has two main parts:

1) **Generating Partial Products:** Each row of multiplication in Figure 3 corresponds to the product of a single secret coefficient of **s** with all coefficients of public polynomial **a**; we refer to this method as row-wise multiplication. The in-place reduction is simply a sign change (modulo 2^8) of the reduced coefficient (depicted with the dashed boxes in Figure 3) for the partial products having degree greater than 3. This degree reduction occurs since the lattice arithmetic is in $\mathcal{R}_q = \mathbb{Z}_q[x]/\langle x^n+1\rangle$, where the reduction function $f(x)$ is x^n+1.
2) **Updating Intermediate Sum:** The result of polynomial multiplication is the addition of all row-wise computations modulo 2^8. Therefore, after a partial product is computed, its value has to be accumulated into and a modular reduction be applied on the intermediate sum that holds the result of previous row computations. After all rows are processed, the value of the intermediate sum will be the result of this operation.

Figure 3 highlights, in bold red, the first row of computations and its dependence on the first secret coefficient. The crux of our attack is to observe that all of these operations rely on the same coefficient of the secret polynomial. The same concept is also true for the matrix multiplication of Frodo. The adversary, therefore, can effectively apply a Horizontal DPA using these operations: it can make a hypothesis guess on a secret coefficient, compute the hypothesis value for each intermediate computation, and test correlations between those values and corresponding activity within the single power trace.

The main problem of applying DPA on the target lattice-based constructions is the false positives of multiplication. Unlike the case of AES or other block ciphers where an S-BOX maximally diffuses similar inputs, multiplications with similar values generate a correlated output. For example, if a secret coefficient is '1' vs. '2', the output of the multiplication will be shifted by one binary digit, resulting in an output having the same Hamming weight unless there is a modular reduction. Even when there is a modular reduction, the output changes by a single overflow bit if the modulo value is a *power of two*, which is exactly the case for the modulo 2^{15} multiplication of Frodo.

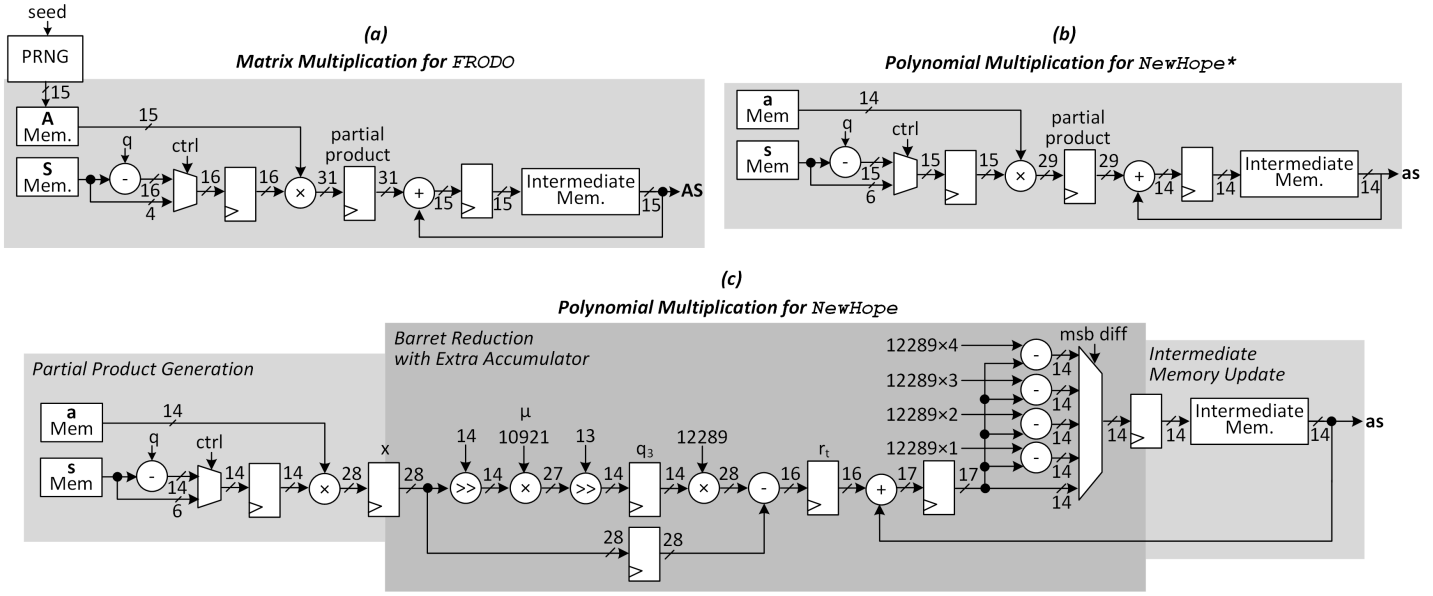

Fig. 4. Hardware architecture of operations under attack for Frodo (a), NewHope* (b), and NewHope (c)

The solution to this problem is to target the intermediate sum update and to extract one key bit at a time, successively, starting from the first coefficient of the key. Since the intermediate sum for every coefficient is '0' in the first row, attacking these sums will be equivalent to attacking partial product generation, hence resulting in false positives. However, due to the modular reductions, after the first row, there is a single guess that yields a high correlation for all previous distinct guess. Therefore, our attack will form an *ensemble* of possible keys, rather than forming a tree having multiple independent false positives at each row of computations.

The number of possible horizontal tests depends on the degree of the polynomial. Since NewHope works with degree 1023 polynomials, our attack can perform 1024 tests for each key guess. For the Frodo protocol we can conduct 752 tests.

Note that our attack does not require enforcing specific patterns in the input (e.g. Fouque *et al.* [30]) to estimate the secret key. An adversary using our attack, therefore, does not have to invoke or modify an encryption request. Instead, it can simply eavesdrop on the communication (in addition to recording power consumption) to extract the secret key, which is less likely to be detected from a higher-level detector in the system.

Another important feature of our attack is the implication of using larger keys to improve theoretical security. In both protocols, a future update on the parameter set, which typically occurs due to disclosure of better attacks, can increase the polynomial size of the key—this improves the effectiveness of our attack as it increases the number of test for the horizontal analysis.

V. Evaluating the Horizontal Attacks

In this section, we evaluate the effectiveness of our proposed Horizontal DPA attack on our FPGA-based hardware implementations using real power measurements.

A. Hardware Architecture

We primarily focus on resource-constrained applications in embedded devices, such as RFID, smart cards or IoT nodes. As such, we design and analyze coefficient-serial architectures that compute a single coefficient of multiplication in a clock cycle. This design uses one multiplier as its main processing unit. Therefore, it takes approximately $\bar{m} \times n \times n$ clock cycles to compute a matrix multiplication of $\mathbf{A}\cdot\mathbf{S}$ and $n \times n$ cycles to compute the polynomial multiplication of $\mathbf{a}\cdot\mathbf{s}$.

Figure 4 (a), (b), and (c) respectively show the details of the hardware architecture for Frodo, NewHope*, and NewHope, where NewHope* is a simplified instantiation of NewHope arithmetic to evaluate the impact of modular reduction on our DPA attack. The main processing unit of Frodo and NewHope* is a multiplier to compute the partial products. The result of the partial product is accumulated to the previous value stored in intermediate memory, which updates the intermediate sum. Since these two instantiations use a modular reduction with a power of two, the modulo operation is free, which is simply a truncation of the adder output to $\log_2 q$ bits.

However, in the NewHope case, the modular reduction is with the constant integer 12289. Thus, NewHope requires a full-scale reduction after the modular multiplication. To efficiently implement this operation, we used the Barret Reduction technique [31], which computes the modular reduction with two multiplications and a small number of subtractions. Listing 1 shows the pseudo-code. This method essentially estimates the quotient of the modular division and removes the misprediction, which has a fixed range, by a sequence of subtractions and bound checking. The range, for the case of 12289, is between 0 and 12289×3. We also integrate the final addition of the intermediate memory updates with this operation and we check the bounds between 0 and 12289×4. To ensure a *constant-time* operation and minimize power side-channel leakage, we perform this bound check in parallel and find the result (based on the overflow bits of subtraction) in one clock cycle.

The size of matrix $\mathbf{A}$, which requires storing $752 \times 752 \times 15$ bits, creates a problem for the Frodo implementation. Since our target FPGA cannot store this amount of data due to BRAM limitations, it has to generate parts of $\mathbf{A}$ on-the-fly during the computation of $\mathbf{A}\cdot\mathbf{S}$. To do so, our architecture follows the guidelines of [24], and generates one column of $\mathbf{A}$ at a time and multiplies it with one row of $\mathbf{S}$ to compute the partial results for the entire $\mathbf{A}\cdot\mathbf{S}$ matrix. Only then, does the hardware generate the next column of $\mathbf{A}$ and repeat the same process until all the columns of $\mathbf{A}$ are swept. This approach minimizes the required

Algorithm 1 Barret reduction of positive integers with a fixed modulus [31]

procedure BARRET REDUCE(x, m, μ, r)
input: $x = (x_{2k-1} \cdots x_1 x_0)_2$, $m = (m_{k-1} \cdots m_1 m_0)_2$ (with $m_{k-1} = 1$), $\mu = \lfloor 2^{2k}/m \rfloor$
output: $r = x \bmod m$
  $q_1 \leftarrow \lfloor x/2^{k-1} \rfloor$
  $q_2 \leftarrow q_1 \cdot \mu$
  $q_3 \leftarrow \lfloor q_2/2^{k+1} \rfloor$
  $r_t \leftarrow x - m \cdot q_3$
  while $r_t \geq m$ **do**
    $r_t \leftarrow r_t - m$
  end while
  $r \leftarrow r_t$
end procedure

amount of storage for **A**.

All these architectures use a generic modulo q sign conversion on secret key (**S**, **s**) coefficients to handle the sign arithmetic and in-place conversions; a similar approach is also taken by prior work on an area-optimized lattice-hardware design [32]. There are two reasons for this approach. First, it allows to mitigate zero-value attacks on lattice arithmetic ([11], Appendix A) when a secret coefficient is equal to 0, and second, it enables achieving a modular design, independent of the size of **s** coefficients. Since the multiplication and additions are mapped to a DSP unit, which can compute up to a 18-bit multiplication and 48-bit addition, converting **s** (or **S**) coefficients into $\log_2 q$ bits does not carry an area overhead.

The hardware architectures we propose are implemented in Verilog HDL and mapped on to the Xilinx Spartan-6 XC6SLX75 FPGA. The synthesis, placement, and routing of the proposed designs to the target FPGA is performed using Xilinx Integrated Synthesis Environment (ISE) version 14.6.

B. Evaluation Setup

To evaluate the power attacks in a real environment, we ported our implementations on the SAKURA-G board, which includes a Xilinx Spartan-6 (XC6SLX75-2CSG484C) FPGA. We measure the voltage drop on a 1-Ω resistance and make use of the on-board amplifiers on the SAKURA-G platform to measure power consumption. The measurements for DPA analysis are taken using a low-end Oscilloscope (SDS1102X Digital Oscilloscope) that can sample at 2 ns intervals (500 MS/s) with two active channels. We use the first channel for power measurements and the second channel to trigger the oscilloscope to start recording. The design is clocked at a constant 1.5 MHz operating frequency.

Prior to DPA, the adversary has to pre-process the power measurement and divide it into smaller parts for the DPA targeted operations. Figure 5 shows the entire power trace, which is then zoomed into the regions of interest. We empirically find applying a 20 MHz low pass filter on the measured signal to be very useful as it reduces noise and achieves better detection result. We then divide the power traces into pieces of one clock period, which we refer to as *sub-trace*. To synchronize these divided sub-traces, we find the minimum point within each sub-trace and synchronize by fixing that point in the sub-trace to a certain clock index. The figure also shows an example hypotheses table for one sub-key of `Frodo`. Note that there are 11 possible values for each sub-key of **S** and as many sub-traces as there are modular multiplications, i.e., there are 752 possible tests for each key guess.

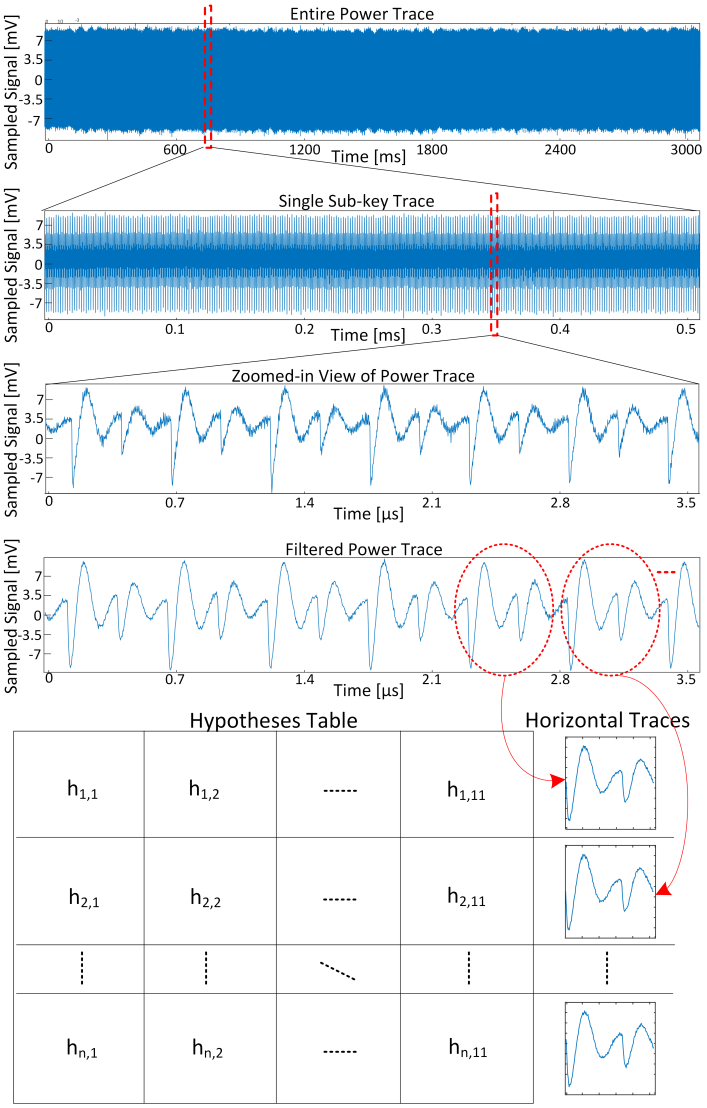

Fig. 5. Pre-processing of power traces for the Horizontal DPA for `Frodo` and generation of the hypotheses table for one sub-key

Power models in our analysis use the Hamming distance of the registers and we consider all registers in the datapath. We use the Pearson correlation coefficient based distinguisher for the differential side-channel attack [33]. This test aims to differentiate populations through their covariance, i.e., by checking if deviations from mean occur in a similar fashion. Correlation trace $c_{i,j}$ for a sub-key guess i is defined as

$$c_{i,j} = \frac{\sum_{d=1}^{D} \left[\left(h_{d,i} - \overline{h_i} \right) \left(t_{d,j} - \overline{t_j} \right) \right]}{\sqrt{\sum_{d=1}^{D} \left(h_{d,i} - \overline{h_i} \right)^2 \sum_{d=1}^{D} \left(t_{d,j} - \overline{t_j} \right)^2}} \qquad (1)$$

where D is the number of traces each having T data points, $t_{d,j}$ is a power trace with $0 < d \leq D$ and $0 < j \leq T$, $\overline{t_j}$ is the mean power trace, $h_{d,i}$ is a power estimate in trace d for the key guess value i, and $\overline{h_i}$ is the mean power estimate. The result $c_{i,j}$ returns a correlation trace with values between [-1,1] that estimates the linear relationship between the sub-key guess c_i and the power measurement for each guess i and time j. This trace depicts the significance and the timing information of the DPA leak.

C. Empirical Validation of Horizontal Attacks

Figure 6 presents the evaluation results of our Horizontal DPA attack on the key exchange protocols and validates the effects we

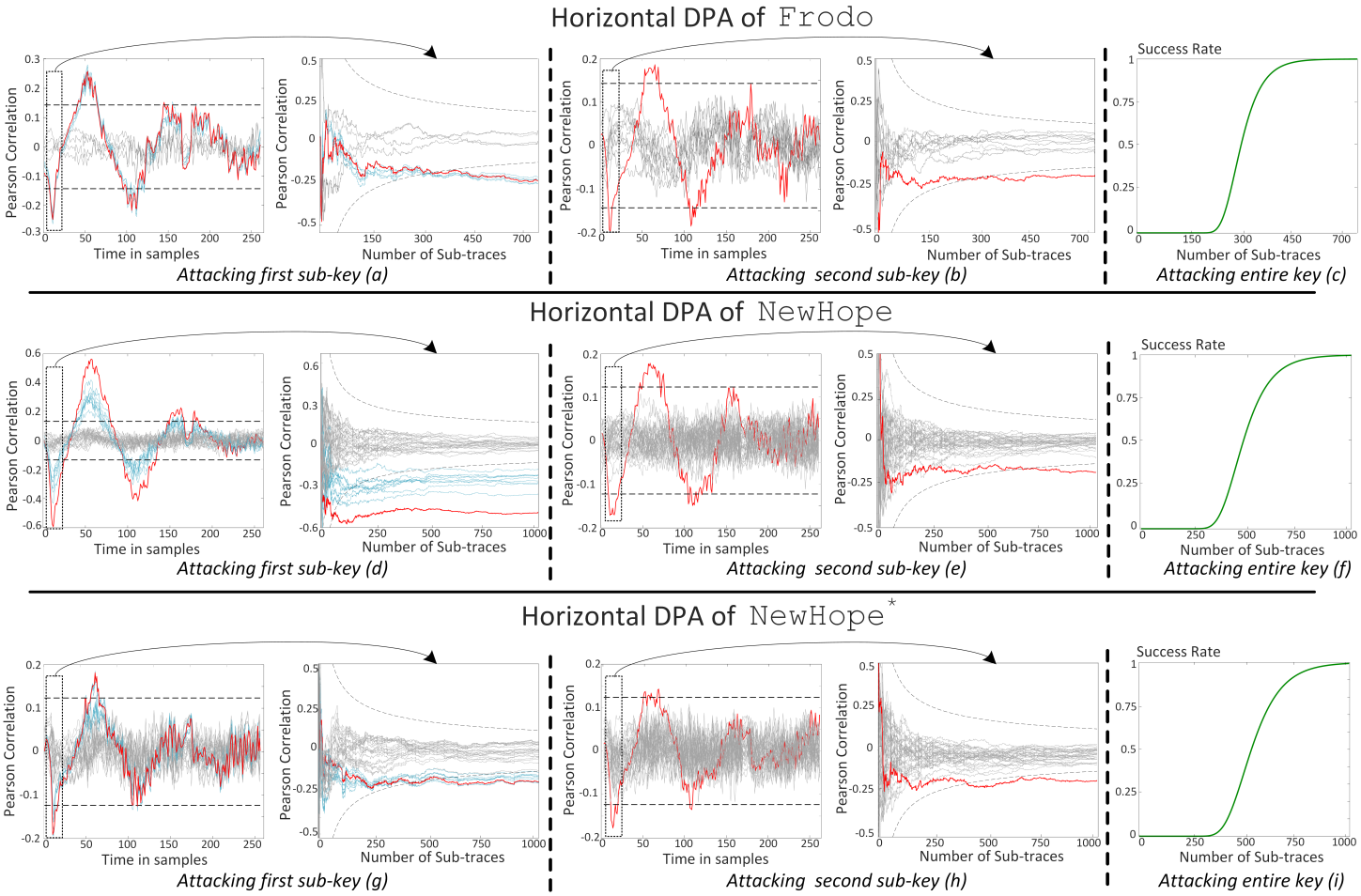

Fig. 6. Evaluation results of the proposed attack on Frodo (a, b, c), NewHope (d, e, f), and NewHope* (g, h, i) arithmetic. Correct key guesses are marked in red and false positves in blue; dashed lines mark the confidence interval of 99.99%. Attacking the first row of key (a, d, g) results in both true and false positives on Frodo and NewHope*. Starting from the second row (b, e, h), false positives are eliminated. In all cases (c, f, i), entire keys can be successfully extracted.

discussed in Section IV-C. Figures 6 (a), (d), and (g) evaluate attacking the first row of coefficients, for the case that first sub-key is '1'. This is the worst-case scenario for the attacker because it causes the maximum number of false positives. In both Frodo and NewHope*, partial product generation has false positives due to modular reduction with a power of two. However, in NewHope (Figure 6 (d)), the modulo value of 12289 generates a sufficient diffusion between similar sub-key guesses (e.g. '1' vs. '2') so the correct key guess results in a more significant correlation compared to false positives. Our tests reveal that on NewHope, the register storing the r_t variable of the Barret reduction has the maximum leak, and the attacker can target the partial product value processed in this register to attack an arbitrary row of polynomial multiplication. However, for Frodo and NewHope*, the attacker has to subsequently extract the key coefficients, starting from the first row, by targeting the intermediate memory update. The results in Figure 6 (b) and (h) show that attacking the second row removes false positives; there is only a single sub-key guess that crosses the threshold. Therefore, from this row onwards, there is a single true positive of the correlation test.

The success rate is the probability of successfully estimating coefficients of an entire key, e.g. 752×8 elements of matrix **S** in Frodo. This value is calculated by performing a hypothesis test on the difference of the correlation coefficient for the incorrect key guess having the highest correlation and the correct key guess—the details of computing this test between two correlation coefficients is given by Mangard *et al.* [34]. In all cases, we observe that there is over 4 standard deviations of disparity on average between the correlation coefficient of the best guess and the second best guess for each sub-key guesses under attack and hence the number of possible horizontal tests is statistically sufficient to estimate the entire key.

VI. Discussions

Our focus in this paper is not on defenses, but since we now demonstrate the potential of Horizontal-DPA, it provides a starting point to discuss potential countermeasures and evaluate their effectiveness/overheads, or reconsider implementation choices (e.g. order of computations, parallelization, whether or not to use NTT for polynomial multiplications) by taking horizontal-DPA and its limitations into account.

In this section, we discuss other lattice schemes that can be vulnerable to our attack, explain the limitations and examine some potential countermeasures.

A. Attacking Other Lattice-Based Cryptosystems

Although the scope of this paper is lattice-based key exchange protocols, our Horizontal DPA has the potential to also break other lattice-based constructions such as R-LWE based public-key decryption. Even though the baseline design for these applications does not necessarily require single-trace attacks as they work with long-term secret keys, masking, blinding, and re-keying based DPA countermeasures can fail against single-trace attacks [18]. Therefore, our attack can be used in the presence of such countermeasures. A potential problem however may occur when implementing our attack on schemes that work with smaller degree polynomials. These polynomials would only

allow a smaller number of horizontal tests and may thus require a better oscilloscope or EM probing to reduce noise.

B. Limitations of Our Attack

We note that, our attack is possible on matrix multiplication and the regular (i.e. schoolbook) polynomial multiplication. An alternative method to implement *polynomial* multiplication is using NTT, which is essentially an arithmetic transformation possible to trade-off area for performance in high-end application scenarios. Indeed, prior works favor schoolbook polynomial multiplication over NTT for area-constrained platforms [35], [32], while some works comment that it may even have better performance than NTT in some corner cases [36] or yield to a higher operating frequency due to its simplified control [37].

Since we focus on constrained embedded devices, we analyze serial architectures, which use sequential data processing. Parallel implementations may reduce the effectiveness of our attack as they introduce more algorithmic noise into the computations.

C. Possible Countermeasures

A common method to mitigate DPA attacks is to introduce randomness into the computations. In our case, this can be achieved by randomizing the order of computations since the result of matrix and polynomial multiplication is independent of the order in which partial products are generated. Another option might be to add dummy steps in the computation. These countermeasures would encumber the adversaries' capability to distinguish sub-traces within a power trace and to associate them with corresponding sub-key guesses.

VII. Conclusion

Key exchange is an important cryptographic routine for large-scale communication protocols. Since these protocols work with ephemeral secrets, it is possible for their side-channel analysis to be overlooked. In this paper, we validate that it is indeed a mistake to assume this limitation would, by default, prevent DPA-style side-channel attacks. As new key exchange protocols are being formulated and deployed, their side-channel evaluation (and not just SPA leaks) should play a role in the decision of their implementation choices. This paper shows that baseline matrix multiplication and polynomial multiplication is vulnerable to a novel side-channel attack. Therefore, some form of countermeasure is required in their implementation.

VIII. Acknowledgments

We thank Ashay Rane for helpful discussions. This work is sponsored in part by Lockheed Martin, National Science Foundation (Award #1453806, #1314709, #1527888, and #1441484), Semiconductor Research Corporation (SRC), and C-FAR: one of the six SRC STARnet Centers, sponsored by MARCO and DARPA. We thank anonymous reviewers for their feedback.

References

[1] P. W. Shor, "Algorithms for quantum computation: discrete logarithms and factoring," in *Proceedings 35th Annual Symposium on Foundations of Computer Science*, Nov 1994, pp. 124–134.

[2] J. Proos and C. Zalka, "Shor's discrete logarithm quantum algorithm for elliptic curves," *Quantum Info. Comput.*, vol. 3, no. 4, pp. 317–344, Jul. 2003.

[3] National Securtiy Agency, "NSA suite B cryptography," https://www.nsa.gov/what-we-do/information-assurance/.

[4] National Institute of Standards and Technology, "Post-quantum cryptography," https://csrc.nist.gov/Projects/Post-Quantum-Cryptography.

[5] M. Braithwaite, "Experimenting with post-quantum cryptography," https://security.googleblog.com/2016/07/experimenting-with-post-quantum.html.

[6] O. Regev, "On lattices, learning with errors, random linear codes, and cryptography," *Journal of the ACM (JACM)*, vol. 56, no. 6, p. 34, 2009.

[7] V. Lyubashevsky, C. Peikert, and O. Regev, "On ideal lattices and learning with errors over rings," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, 2010, pp. 1–23.

[8] J. Howe, C. Moore, M. O'Neill, F. Regazzoni, T. Güneysu, and K. Beeden, "Lattice-based encryption over standard lattices in hardware," in *Proceedings of the 53rd Annual Design Automation Conference*. ACM, 2016, p. 162.

[9] T. Oder, T. Güneysu, F. Valencia, A. Khalid, M. O'Neill, and F. Regazzoni, "Lattice-based cryptography: From reconfigurable hardware to ASIC," in *2016 International Symposium on Integrated Circuits*, Dec 2016, pp. 1–4.

[10] A. Park and D. G. Han, "Chosen ciphertext simple power analysis on software 8-bit implementation of Ring-LWE encryption," in *2016 IEEE Asian Hardware-Oriented Security and Trust*, Dec 2016, pp. 1–6.

[11] O. Reparaz, S. S. Roy, R. de Clercq, F. Vercauteren, and I. Verbauwhede, "Masking ring-lwe," *Journal of Cryptographic Engineering*, vol. 6, no. 2, pp. 139–153, 2016.

[12] O. Reparaz, R. de Clercq, S. S. Roy, F. Vercauteren, and I. Verbauwhede, "Additively homomorphic ring-lwe masking," in *International Workshop on Post-Quantum Cryptography*. Springer, 2016, pp. 233–244.

[13] T. Oder, T. Schneider, T. Pöppelmann, and T. Güneysu, "Practical CCA2-secure and masked Ring-LWE implementation," Cryptology ePrint Archive, Report 2016/1109, 2016, http://eprint.iacr.org/2016/1109.

[14] A. Atici, L. Batina, B. Gierlichs, and I. Verbauwhede, "Power analysis on NTRU implementations for RFIDs: First results," pp. 128–139, 2008.

[15] M.-K. Lee, J. E. Song, D. Choi, and D.-G. Han, "Countermeasures against power analysis attacks for the NTRU public key cryptosystem," *IEICE transactions on fundamentals of electronics, communications and computer sciences*, vol. 93, no. 1, pp. 153–163, 2010.

[16] A. Wang, X. Zheng, and Z. Wang, "Power analysis attacks and countermeasures on NTRU-based wireless body area networks," *KSII Transactions on Internet and Information Systems (TIIS)*, vol. 7, no. 5, pp. 1094–1107, 2013.

[17] X. Zheng, A. Wang, and W. Wei, "First-order collision attack on protected NTRU cryptosystem," *Microprocessors and Microsystems*, vol. 37, no. 6, pp. 601–609, 2013.

[18] R. Primas, P. Pessl, and S. Mangard, "Single-trace side-channel attacks on masked lattice-based encryption," in *International Conference on Cryptographic Hardware and Embedded Systems*. Springer, 2017, pp. 513–533.

[19] P. Pessl, "Analyzing the shuffling side-channel countermeasure for lattice-based signatures," in *Progress in Cryptology–INDOCRYPT 2016: 17th International Conference on Cryptology in India, Kolkata, India, December 11-14, 2016, Proceedings 17*. Springer, 2016, pp. 153–170.

[20] T. Espitau, P.-A. Fouque, B. Gerard, and M. Tibouchi, "Side-channel attacks on BLISS lattice-based signatures – exploiting branch tracing against strongswan and electromagnetic emanations in microcontrollers," Cryptology ePrint Archive, Report 2017/505, 2017, http://eprint.iacr.org/2017/505.

[21] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Advances in cryptologyCRYPTO99*. Springer, 1999, pp. 789–789.

[22] C. Clavier, B. Feix, G. Gagnerot, M. Roussellet, and V. Verneuil, "Horizontal correlation analysis on exponentiation." in *ICICS*, vol. 6476. Springer, 2010, pp. 46–61.

[23] A. Bauer, E. Jaulmes, E. Prouff, and J. Wild, *Horizontal Collision Correlation Attack on Elliptic Curves*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 553–570.

[24] J. Bos, C. Costello, L. Ducas, I. Mironov, M. Naehrig, V. Nikolaenko, A. Raghunathan, and D. Stebila, "Frodo: Take off the ring! practical, quantum-secure key exchange from lwe," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2016, pp. 1006–1018.

[25] E. Alkim, L. Ducas, T. Pöppelmann, and P. Schwabe, "Post-quantum key exchange—a new hope." in *USENIX Security Symposium*, 2016, pp. 327–343.

[26] J. Balasch, B. Gierlichs, O. Reparaz, and I. Verbauwhede, *DPA, Bitslicing and Masking at 1 GHz*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2015, pp. 599–619.

[27] Y. Zhang, A. Juels, M. K. Reiter, and T. Ristenpart, "Cross-VM side channels and their use to extract private keys," in *Proceedings of the 2012 ACM Conference on Computer and Communications Security*, ser. CCS '12. New York, NY, USA: ACM, 2012, pp. 305–316.

[28] M. S. İnci, B. Gulmezoglu, G. Irazoqui, T. Eisenbarth, and B. Sunar, *Cache Attacks Enable Bulk Key Recovery on the Cloud*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 368–388.

[29] S. Chari, J. R. Rao, and P. Rohatgi, "Template attacks," in *International Workshop on Cryptographic Hardware and Embedded Systems*. Springer, 2002, pp. 13–28.

[30] P.-A. Fouque and F. Valette, "The doubling attack-why upwards is better than downwards," in *CHES*, vol. 2779. Springer, 2003, pp. 269–280.

[31] A. J. Menezes, P. C. Van Oorschot, and S. A. Vanstone, *Handbook of applied cryptography*. CRC press, 1996.

[32] T. Pöppelmann and T. Güneysu, "Area optimization of lightweight lattice-based encryption on reconfigurable hardware," in *2014 IEEE International Symposium on Circuits and Systems (ISCAS)*, June 2014, pp. 2796–2799.

[33] E. Brier, C. Clavier, and F. Olivier, "Correlation power analysis with a leakage model," in *International Workshop on Cryptographic Hardware and Embedded Systems*. Springer, 2004, pp. 16–29.

[34] S. Mangard, E. Oswald, and T. Popp, *Statistical Characteristics of Power Traces*. Boston, MA: Springer US, 2007, pp. 61–99.

[35] T. Güneysu, V. Lyubashevsky, and T. Pöppelmann, *Practical Lattice-Based Cryptography: A Signature Scheme for Embedded Systems*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 530–547.

[36] T. Pöppelmann and T. Güneysu, "Towards efficient arithmetic for lattice-based cryptography on reconfigurable hardware," in *Proceedings of the 2nd International Conference on Cryptology and Information Security in Latin America*. Berlin, Heidelberg: Springer-Verlag, 2012, pp. 139–158.

[37] J. Buchmann, F. Göpfert, T. Güneysu, T. Oder, and T. Pöppelmann, "High-performance and lightweight lattice-based public-key encryption," in *Proceedings of the 2Nd ACM International Workshop on IoT Privacy, Trust, and Security*, ser. IoTPTS '16. New York, NY, USA: ACM, 2016, pp. 2–9.