A Lightweight and SCA-resistant NTT IP core for Kyber and (CRYSTALS)-like schemes

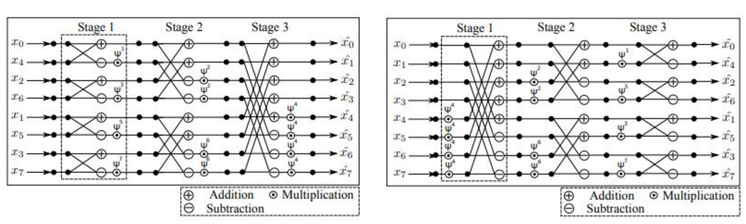
Kashif Nawaz*, Eugenio SalazarBrenes and Jeroen Delvaux
Cryptography Research Centre, Technology Innovation Institute, Abu Dhabi, United Arab Emirates
Corresponding author: kashif.nawaz@tii.ae

1. Abstract

While the NTT is the method of choice of polynomial multiplication in schemes such as Kyber, it is often the bottleneck in software implementations. A re-configurable hardware accelerator that is able to offload the computationally intensive NTT, allows for a lower latency (compared to software implementations) and higher throughput. However, for lightweight (LW) applications, lower area and in-built side-channel resistance are favored over increased speeds. In this ongoing work-in-progress, we present a secure NTT core with a low-area footprint, which can be the HW accelerator of choice for LW applications. While the hardware optimizations in this case are for the (16-bit) Kyber scheme only, our work can easily be extended to similar schemes that reply on the NTT (and consequently, the Inverse-NTT), such as the (32-bit) Dilithium.

2. Motivation

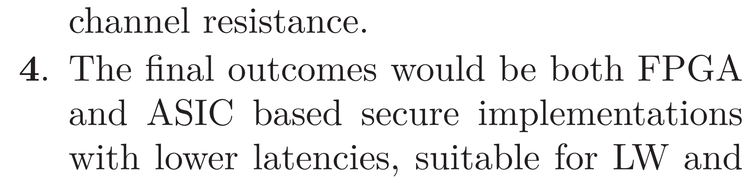
Polynomial multiplication in the case of Kyber (or Dilithium), the NTT transformation and multiplication is utilized in both the encapsulation and decapsulation phases of the algorithm. Additionally, since the multiplication often involves the secret key (e.g., in the case of decryption), it is imperative that such operations be resistant to any side-channel analysis leakage.



Generic NTT, from [1]

3. Existing State-of-Art

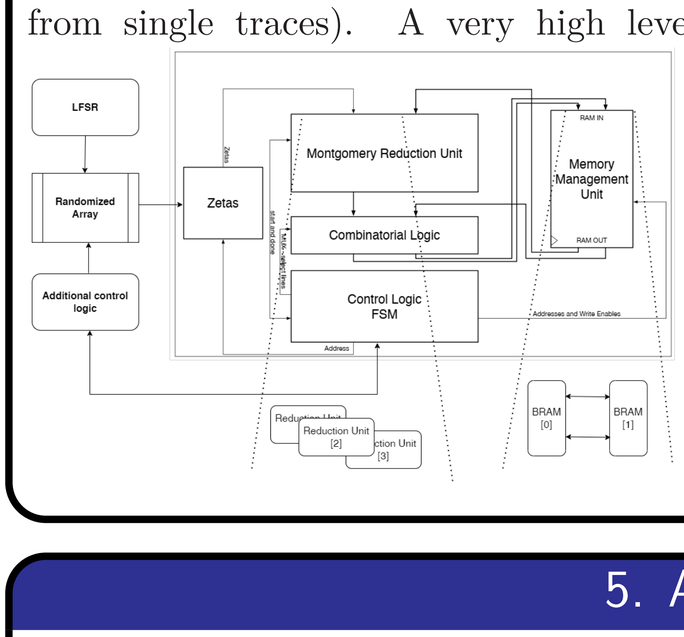
Existing implementations of the NTT mostly focused on optimizing the latency, and subsequently, neglecting the area-footprint and the inherent side-channel resistance of the NTT module. The work of Ravi et al.[1] provided the first directions for a secure (masking or shuffling as countermeasures) SW implementation. To the best of our knowledge, ours is the first work (in-progress), on securing the NTT for HW implementations. Further, the work of Hamburg et al. demonstrated that the Belief Propagation attack could weaken the shuffled implementations of [1], but not fully and attacks on the masked implementations still remain an open question.

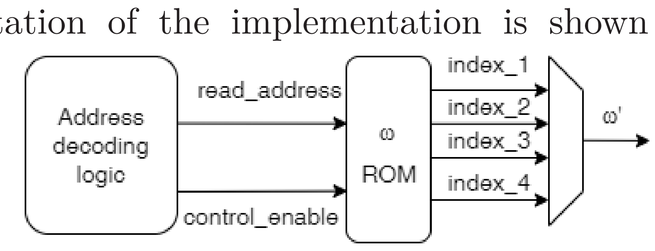


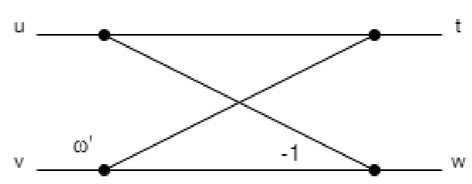
Masked NTT, from [1]

4. Secure Implementation

In this work, we build on the existing masked and shuffled implementations (in software) from [1]. The countermeasure consists of masking the twiddle constants (locations) into multiple shares and secondly, by randomizing the execution order of such operations is regarded as an effective technique against horizontal side-channel attacks (which often rely on leakages obtained from single traces). A very high level representation of the implementation is shown below.







5. Area Comparisons

Initial results, for now, in table below ($*^a$) show that our secure NTT implementations perform reasonably well in terms of the overall area overhead while maintaining a relatively similar levels of frequency of operation. Our next steps already consists of the actual implementation in the target FPGAs (both SAKURA-G and the ChipWhisperer CW-305 and perform side-channel analysis (using TVLA, for instance, as one of the evaluation techniques). Although protected implementations have an increased area, the overall increase is fairly low, considering similar latencies and randomness requirements, often required for masked implementations.

Implementation	LUTs	FFs	DSP	BRAM	f, MHz
Masked Generic*	471	403	28	1.5	193.08
Masked Fine*	379	399	28	1	155.18
Shuffling*	172	118	7	0.5	155.18
[2] (Virtex-7)	4670	4315	-	-	200
[2] (Virtex-7)	36587	34205	-	32	140
[3] (Spartan-6)	985	444	1	5	138
[3] (Artix7)	948	352	1	2.5	190
[4] (Artix7)	980	395	26	2	-
[5] (Artix7)	1349	860	1	2	-

[a] this work

6. Future Directions

This work-in-progress is currently being updated with the following

1. Currently, implementation on a SAKURA-G board is ongoing. The goal is to have a quick appreciation of the side-channel resistance using techniques such as TVLA.
2. Implementations on the CW-305 Artix target to conduct side-channel analysis and measurements.
3. Design and fabrication of a complete chip to measure ASIC performance and side-channel resistance.
4. The final outcomes would be both FPGA and ASIC based secure implementations with lower latencies, suitable for LW and IoT based applications.

7. References

[1] Prasanna Ravi, Romain Poussier, Shivam Bhasin, and Anupam Chattopadhyay. On configurable SCA countermeasures against single trace attacks for the NTT. In *International Conference on Security, Privacy, and Applied Cryptography Engineering*, pages 123–146. Springer, 2020.

[2] Austin Hartshorn, Humberto Leon, Noel Qiao, and Scott Weber. Number theoretic transform (NTT) FPGA accelerator. *Worcester Polytech. Inst., Worcester, MA, USA, Tech. Rep. E-project-051420-162339*, pages 1–37, 2020.

[3] Ferhat Yaman, Ahmet Can Mert, Erdinç Öztürk, and Erkay Savaş. A hardware accelerator for polynomial multiplication operation of CRYSTALS-KYBER PQC scheme. In *2021 Design, Automation Test in Europe Conference Exhibition (DATE)*, pages 1020–1025, 2021.

[4] Tim Fritzmann and Johanna Sepúlveda. Efficient and flexible low-power NTT for lattice-based cryptography. In *2019 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*, pages 141–150, 2019.

[5] Sujoy Sinha Roy, Frederik Vercauteren, Nele Mentens, Donald Donglong Chen, and Ingrid Verbauwhede. Compact Ring-LWE Cryptoprocessor. volume 8731 of *Lecture Notes in Computer Science*, pages 371–391. Springer, 2014.