Fast Arithmetic Hardware Library For RLWE-Based Homomorphic Encryption

Rashmi Agrawal*, Lake Bu†, Michel A. Kinsy*
*Adaptive and Secure Computing Systems (ASCS) Laboratory, ECE Department, Boston University
*{rashmi23, mkinsy}@bu.edu
†The Charles Stark Draper Laboratory, Cambridge, MA
†{lbu}@draper.com

***Abstract*—With billions of devices connected over the internet, the rise of sensor-based electronic devices have led to cloud computing being used as a commodity technology service. These sensor-based devices are often small and limited by power, storage, or compute capabilities, and hence, they achieve these capabilities via cloud services. However, this gives rise to data privacy issues as sensitive data is stored and computed over the cloud, which at most times, is a shared resource. Homomorphic encryption can be used along with cloud services to perform computations on encrypted data, guaranteeing data privacy. While about a decade's work on improving homomorphic encryption has ensured its practicality, it is still several magnitudes slower than expected, making it expensive and infeasible to use. In this work, we propose a first-of-its-kind FPGA-based arithmetic hardware library that focuses on accelerating the key arithmetic operations involved in Ring Learning with Error (RLWE) based homomorphic encryption. We design and implement the FPGA-based Residue Number System (RNS), Chinese Remainder Theorem (CRT), modulo inverse and modulo reduction operations as a first step. For all of these operations, we include a hardware cost efficient serial, and a fast parallel implementation in the library. A modular and parameterized design approach helps in easy customization, provides flexibility to extend these operations for use in most homomorphic encryption applications, and fits well into emerging FPGA-equipped cloud architectures.**

***Index Terms*—Homomorphic Encryption, GCD, Modulo Inverse, CRT, Modulo Reduction, Barrett Reduction.**

I. Overview

Homomorphic encryption allows evaluating functions on encrypted data to generate an encrypted result. This result, when decrypted, matches the result of the same operations performed on the unencrypted data. So, a data owner can encrypt the data and then send it to cloud for processing. The cloud, running the homomorphic encryption based services, will perform computations on the encrypted data and send the results back to the data owner. The data owner, having access to the private key, performs the decryption and obtains the result. The cloud does not have access to the private key or the plain data, and hence, the security concerns related to private data processing on the cloud can be mitigated. One such illustrative scenario is shown in figure 1.

While homomorphic encryption has become a feasible form of computation [1], it remains several magnitudes slower, making it expensive and resource intensive. There are no existing homomorphic encryption schemes with performance levels that would allow large-scale practical usage. There is a need to accelerate the homomorphic encryption operation directly on the hardware to achieve maximum throughput with a low latency. With this goal, we propose an FPGA-based arithmetic library that includes major arithmetic operations involved in homomorphic encryption. To improve their power usage and performance, new cloud architectures are integrating FPGAs to offload and accelerate compute tasks such as deep learning, encryption, and video conversion. The FPGA-based design and optimization approach introduced in this work fit into this class of FPGA-equipped cloud architectures. The key contributions of the work are as follows:

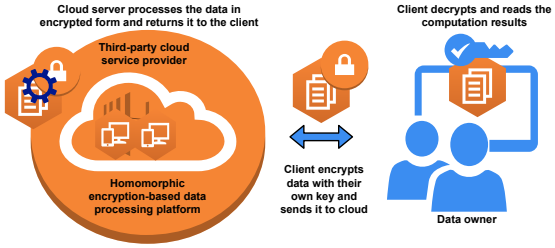

Fig. 1. Third-party cloud service provider with Homomorphic Encryption.

- A fast and hardware cost efficient FPGA-based arithmetic library to accelerate key operations in homomorphic encryption.
- A modular and parameterized design implementation that helps easy customization and provides flexibility to extend these operations for use with any application.
- An underlying RLWE-based encryption scheme that makes the implementation quantum proof with 128-bit security and ensures security even in the post-quantum era.

II. Summary

Under this work, we introduce a fast FPGA-based arithmetic hardware library with a focus on accelerating the key arithmetic operations involved in RLWE-based homomorphic encryption. We implement the Residue Number System (RNS), Chinese Remainder Theorem (CRT), modulo inverse, and modulo reduction operations as a first step towards this hardware library. For all of these operations, we include a hardware cost efficient serial implementation and a fast parallel implementation in the library. All the modules are parameterized to provide flexibility of easy plug in to any other implementations including FPGA-equipped cloud services.

References

[1] C. Gentry *et al.*, "Fully homomorphic encryption using ideal lattices." in *Stoc*, vol. 9, no. 2009, 2009, pp. 169–178.