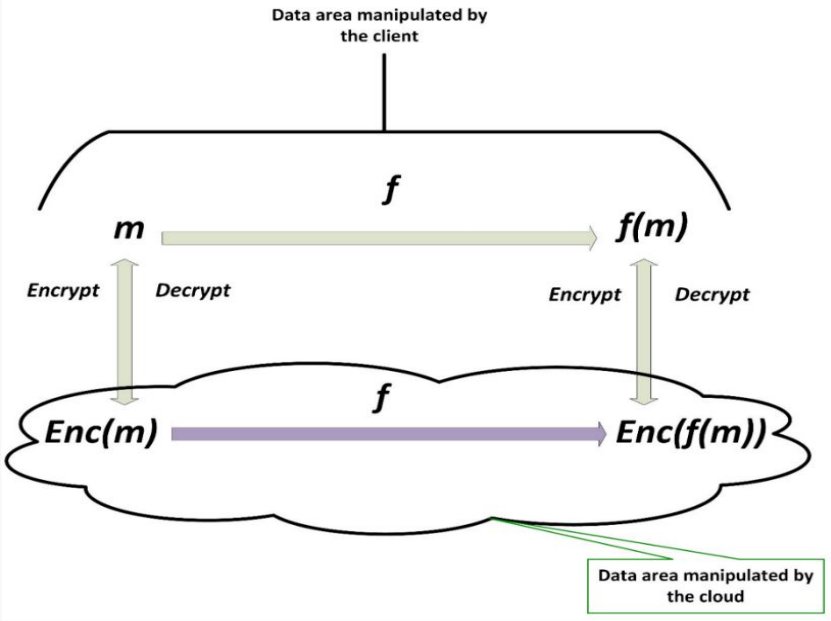
Hardware Acceleration of HE

B Sathiya Naraayanan
IMT2020534

Homomorphic Encryption

Homomorphic Encryption

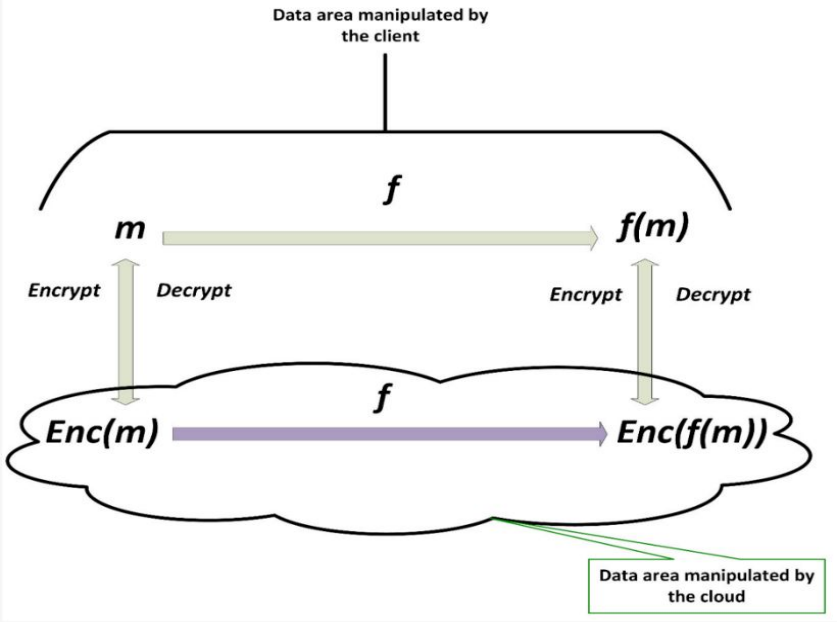
- Homomorphic encryption is the conversion of data into ciphertext that can be analyzed and worked with as if it were still in its original form.
- Homomorphic encryption enables complex mathematical operations to be performed on encrypted data without compromising the encryption.



https://www.mdpi.com/2227-7080/7/1/21

Homomorphic Encryption

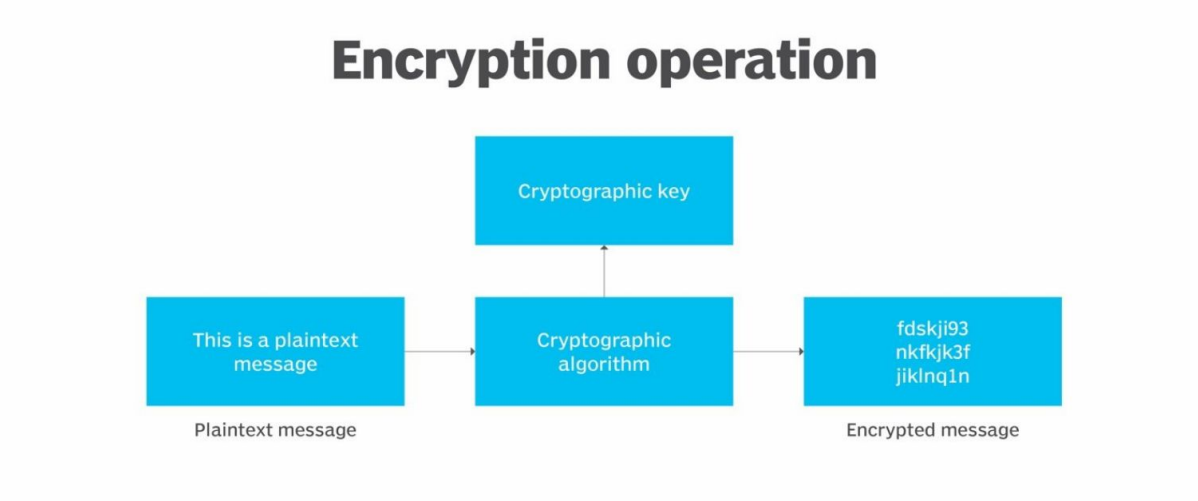
- The term *homomorphic* describes the transformation of one data set into another while preserving relationships between elements in both sets.
- The data in a homomorphic encryption scheme retains the same structure, identical mathematical operations will provide equivalent results regardless of whether the action is performed on encrypted or decrypted data.



https://www.mdpi.com/2227-7080/7/1/21

Homomorphic Encryption

- Enables mathematical computations to be performed directly on the encrypted data
- Can make the handling of user data by third parties safer
- Is implemented so that it's hidden from observers.

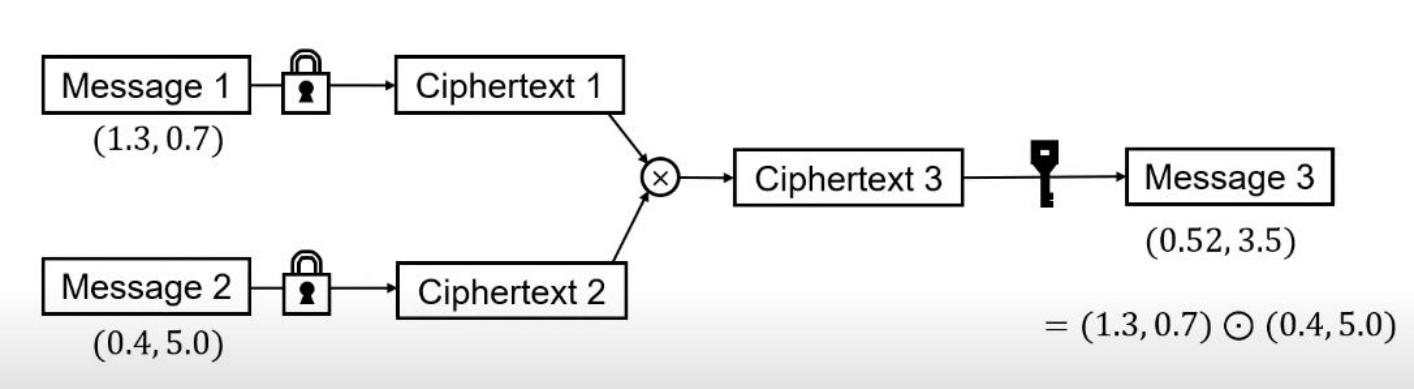


https://www.techtarget.com/searchsecurity/definition/homomorphic-encryption

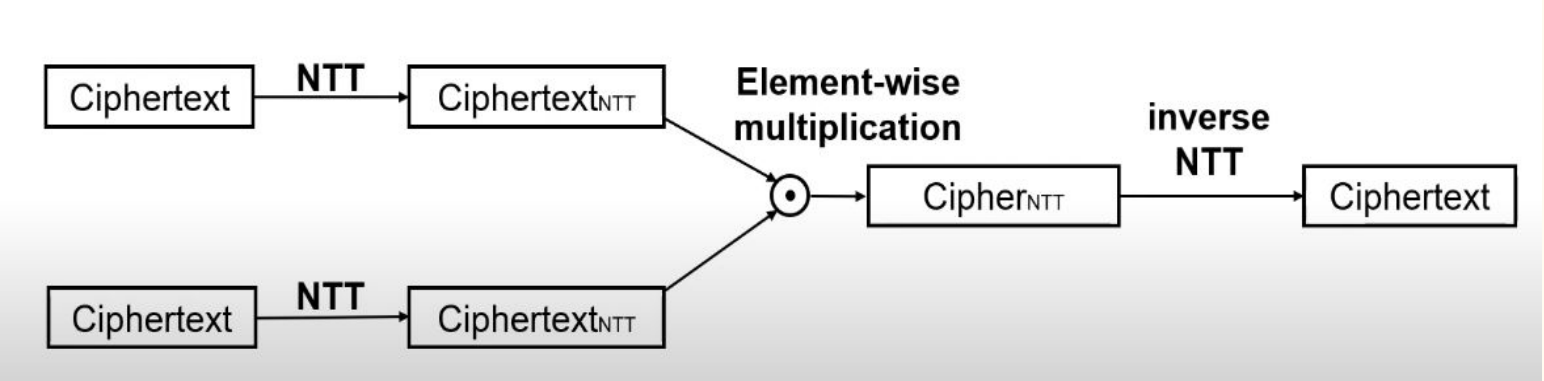
Number Theoretic Transform

Introduction

Number Theoretic Transform



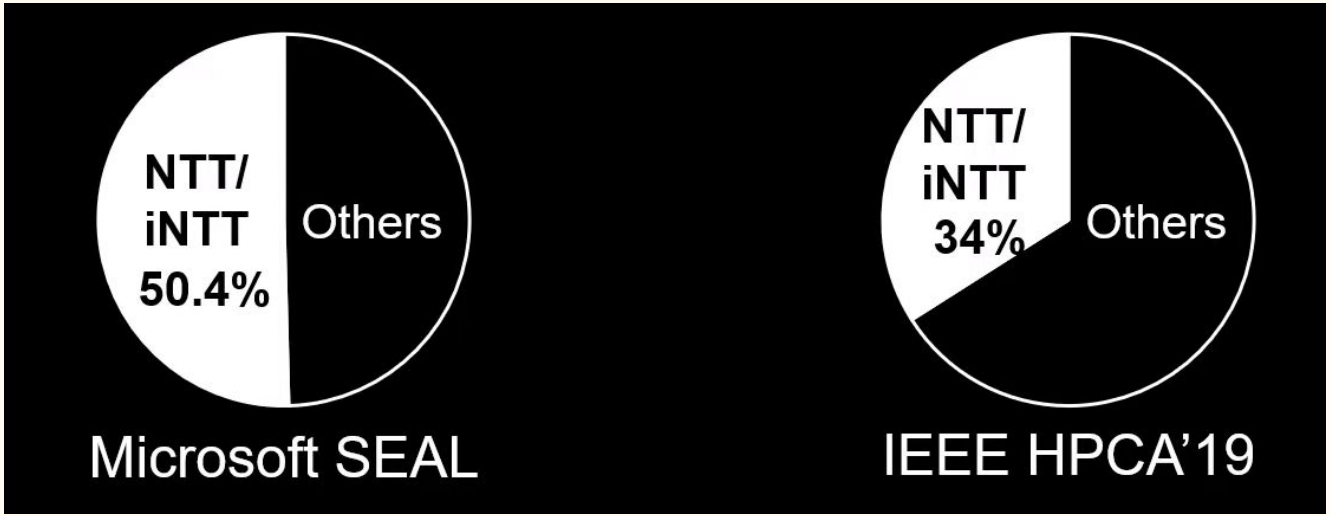
Number Theoretic Transform



Need for Accelerating NTT

Number Theoretic Transform

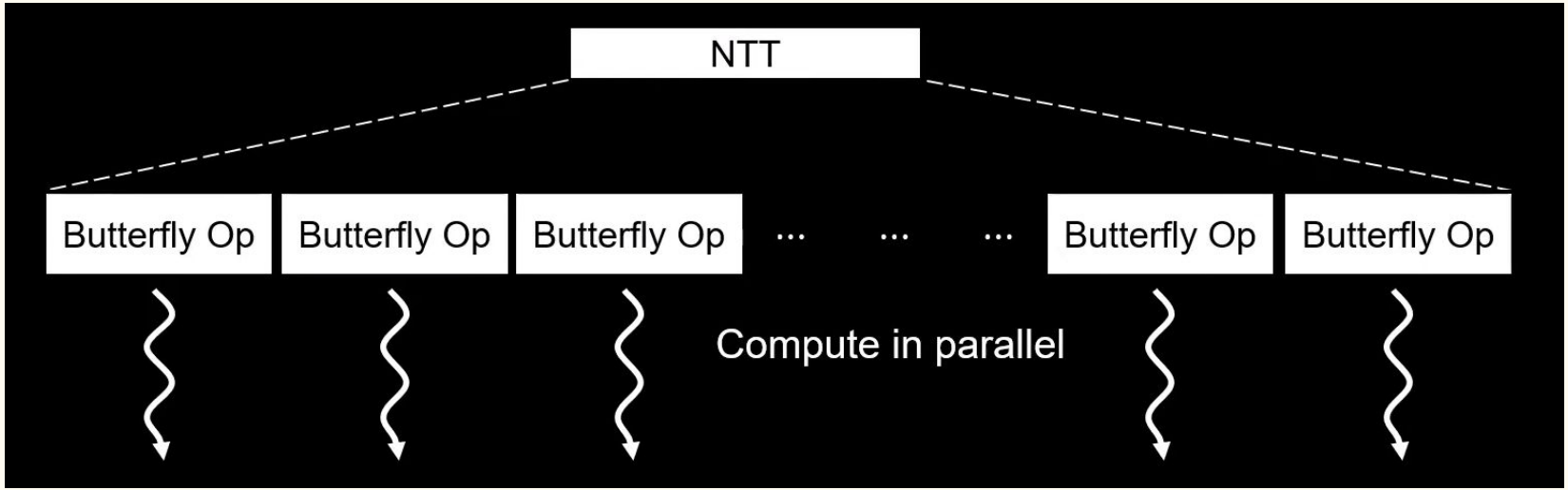
The following shows the ratio of NTT/iNTT to others in the ciphertext multiplication time.



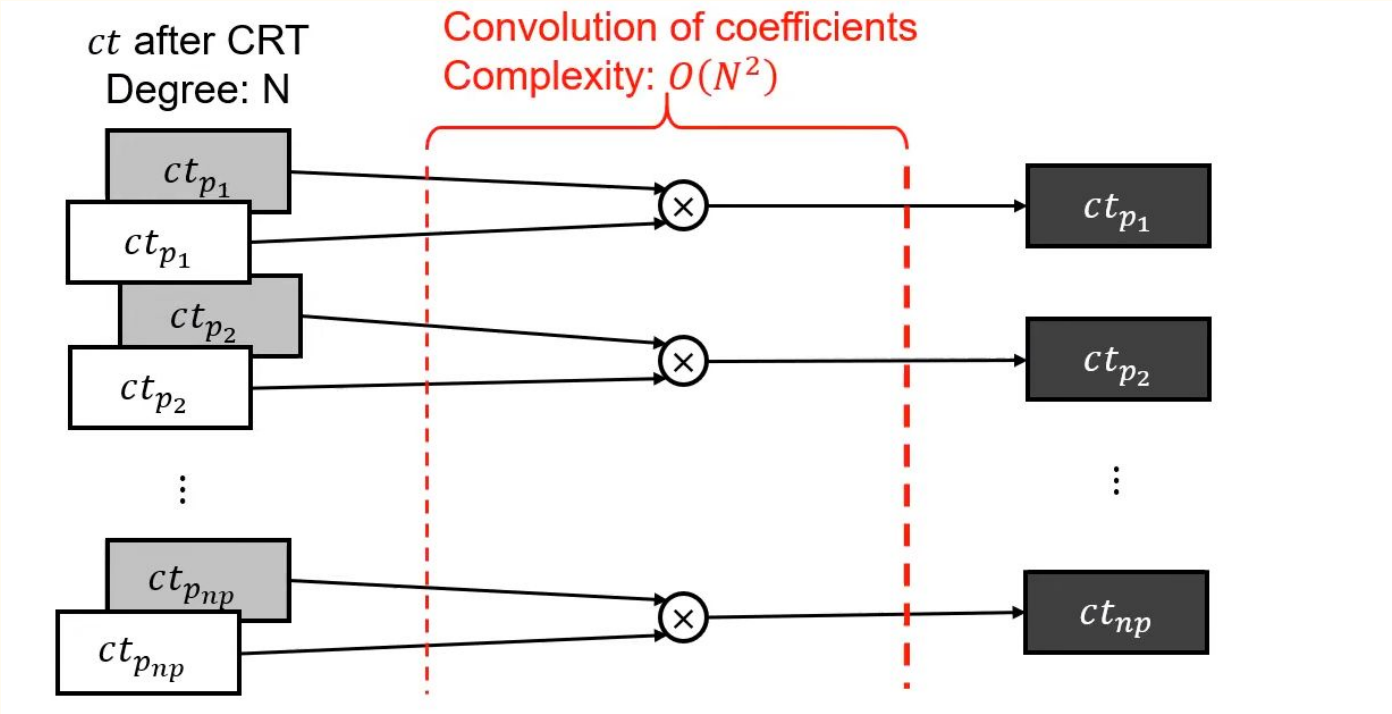
Algorithm

Number Theoretic Transform

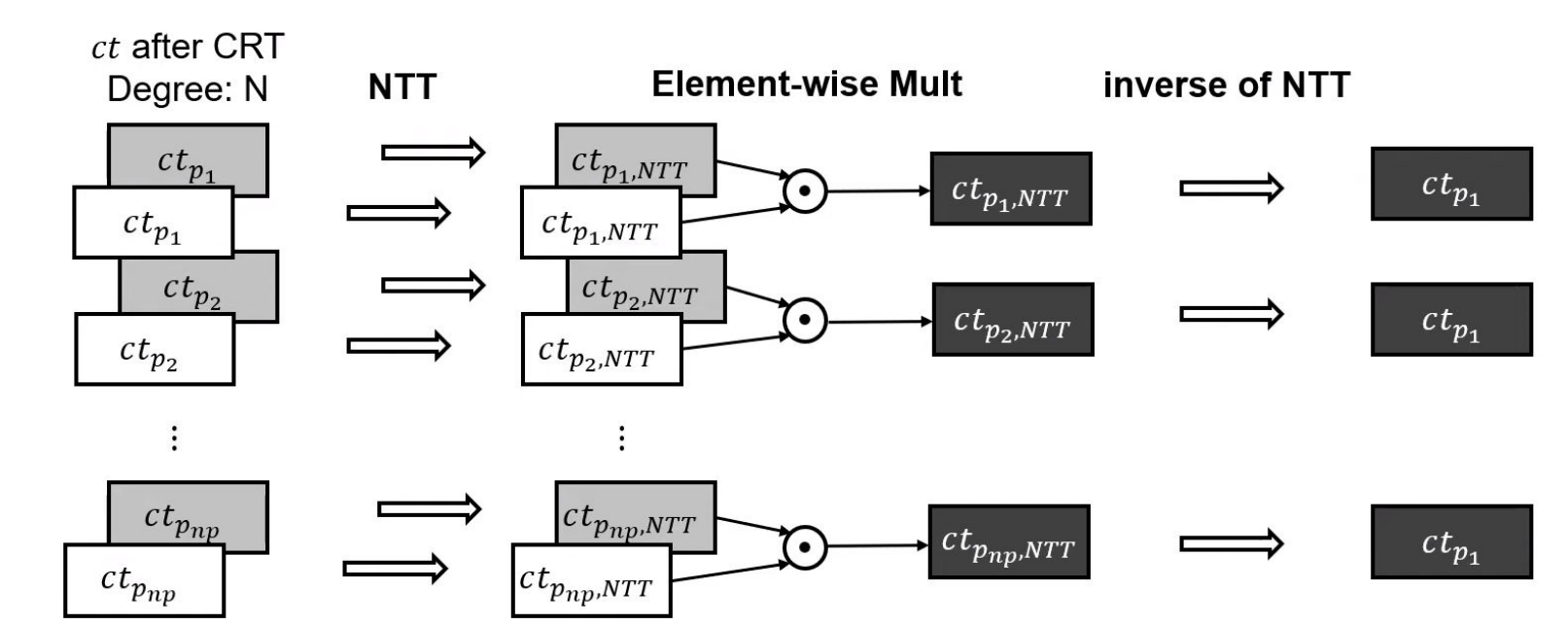
The main difference of FFT and NTT is that, the first operates over reals, while the second operates over a finite field.NTT has modulo multiplications unlike DFT's floating point.



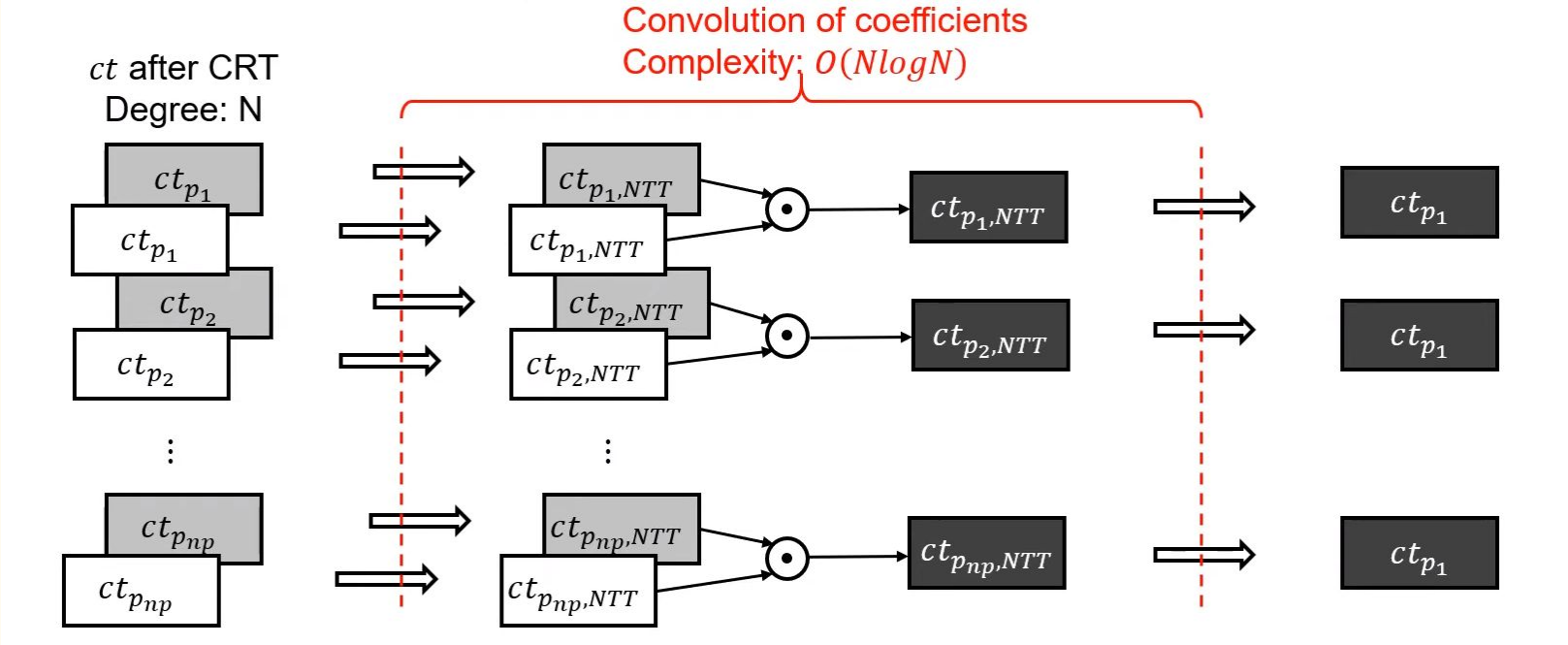
Number Theoretic Transform



Number Theoretic Transform



Number Theoretic Transform

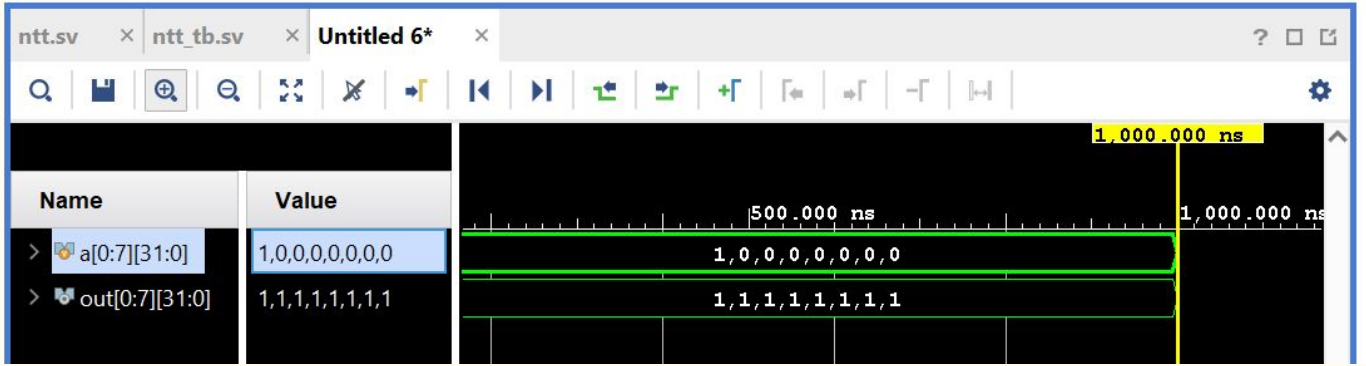


Results

TESTCASES

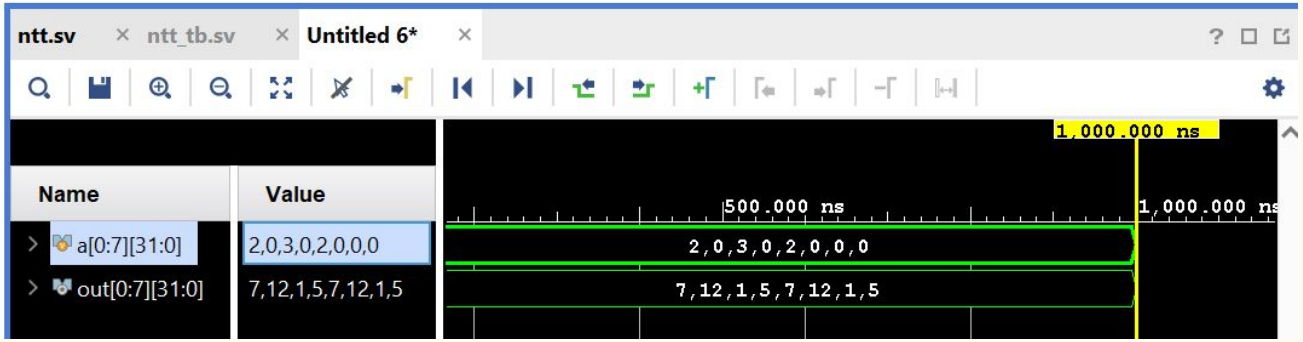
Input vector (X):	[1,1,0,0,0,0,0,0]
Minimum working modulus (M):	11 (optional)
nth root of unity (ω):	2 (optional)
Vector length (n):	8
Chosen modulus (N):	17
Chosen nth root of unity (ω):	2
Output vector (Y):	[2, 3, 5, 9, 0, 16, 14, 10]
	Calculate

TESTCASES



Input vector (X):	[1,0,0,0,0,0,0,0]
Minimum working modulus (M):	11 (optional)
nth root of unity (ω):	2 (optional)
Vector length (n):	8
Chosen modulus (N):	17
Chosen nth root of unity (ω):	2
Output vector (Y):	[1, 1, 1, 1, 1, 1, 1, 1]
	Calculate

TESTCASES



Input vector (X):	[2,0,3,0,2,0,0,0]
Minimum working modulus (M):	11 (optional)
nth root of unity (ω):	2 (optional)
Vector length (n):	8
Chosen modulus (N):	17
Chosen nth root of unity (ω):	2
Output vector (Y):	[7, 12, 1, 5, 7, 12, 1, 5]
	Calculate

RESULTS

8 inputs 4 bits sw 32.24

Hw 13.11

8 inputs 8 bits sw 32.412

Hw 25.635

8 inputs 16 bits sw 36.22

Hw 82.74

8 inputs 32 bits sw 37.44

Hw 91.95

RESULTS

16 inputs 4 bits sw 33.12

Hw 23.10

16 inputs 8 bits sw 35.42

Hw 51.24