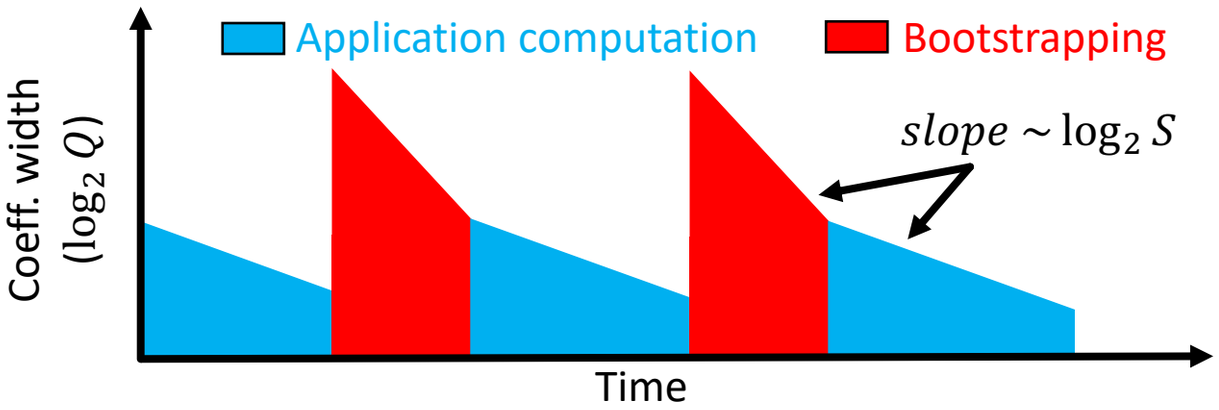
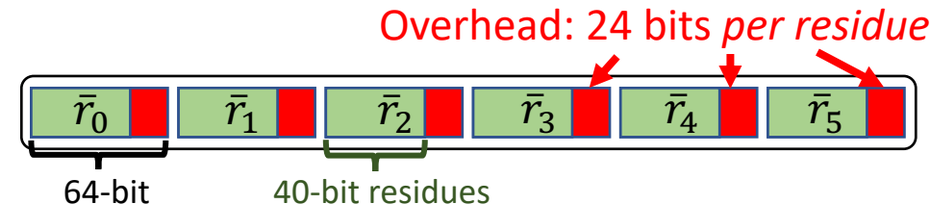
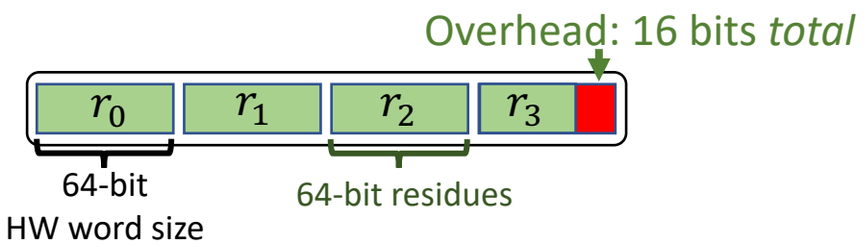
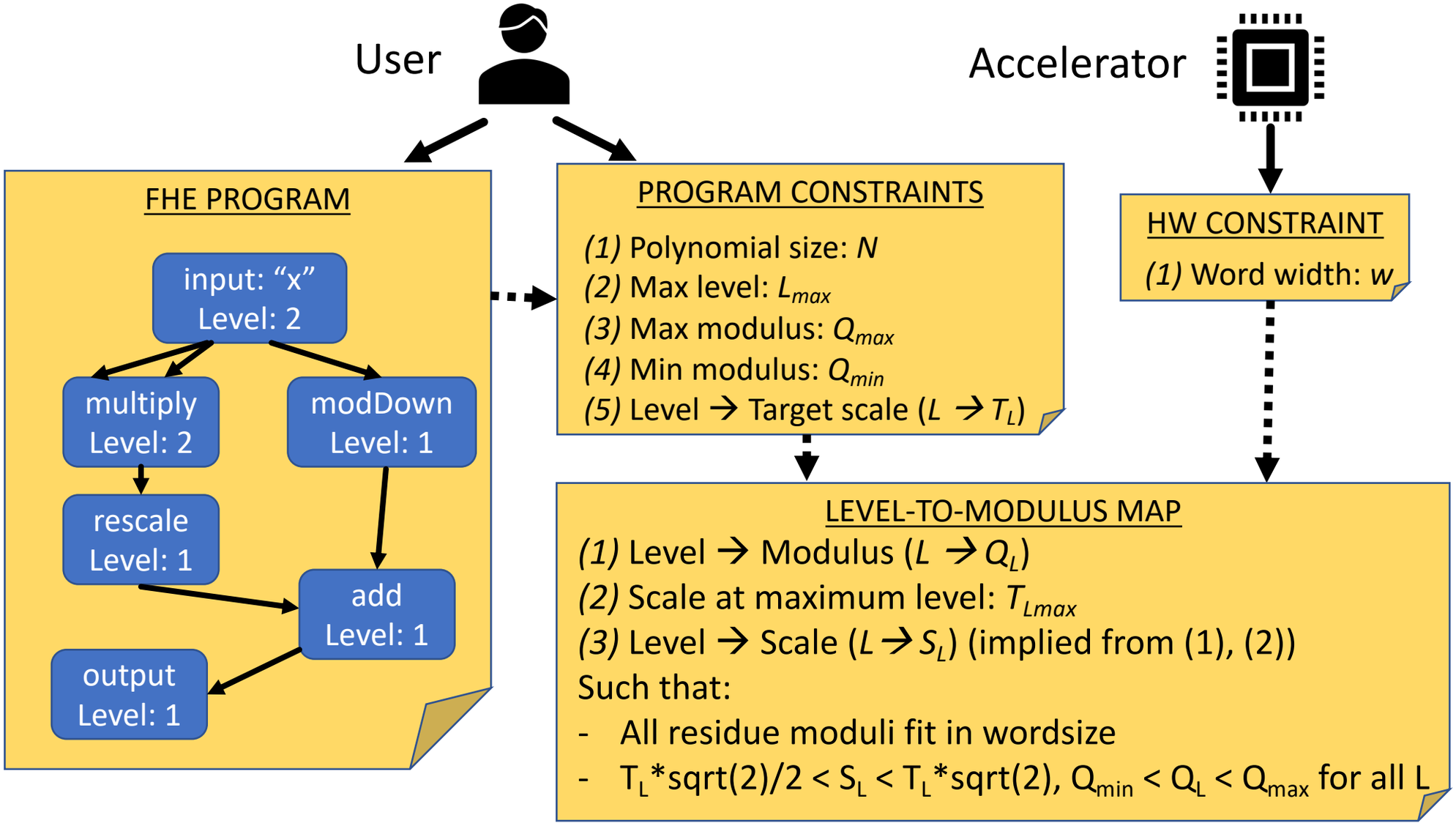

Application computation
Bootstrapping
$slope \sim \log_2 S$
Coeff. width
($\log_2 Q$)
Time

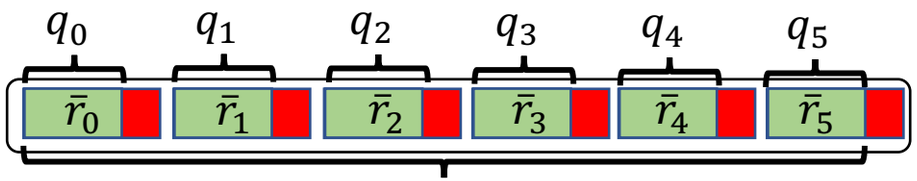
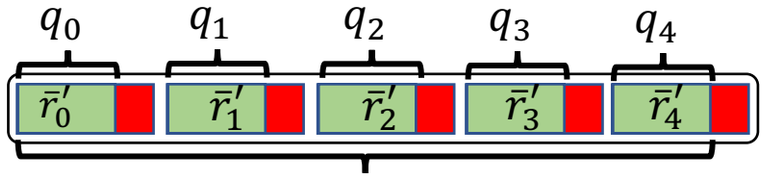
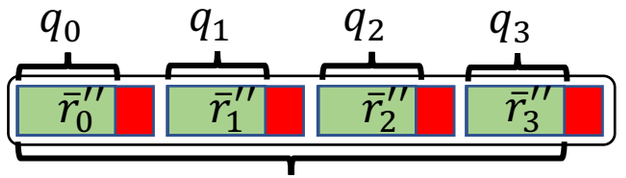


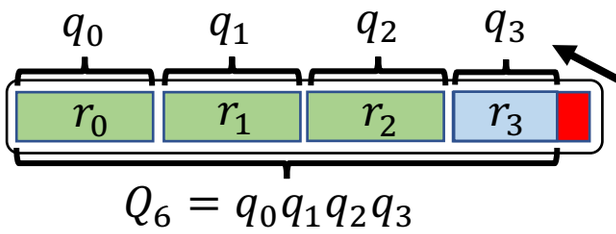
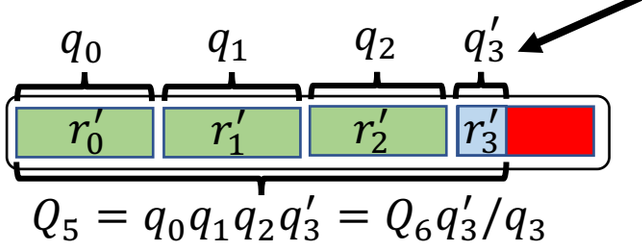
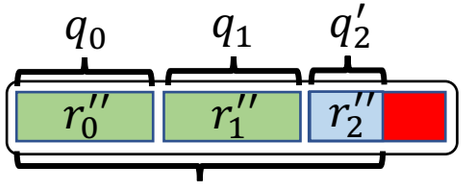
(a) RNS-CKKS (prior work)

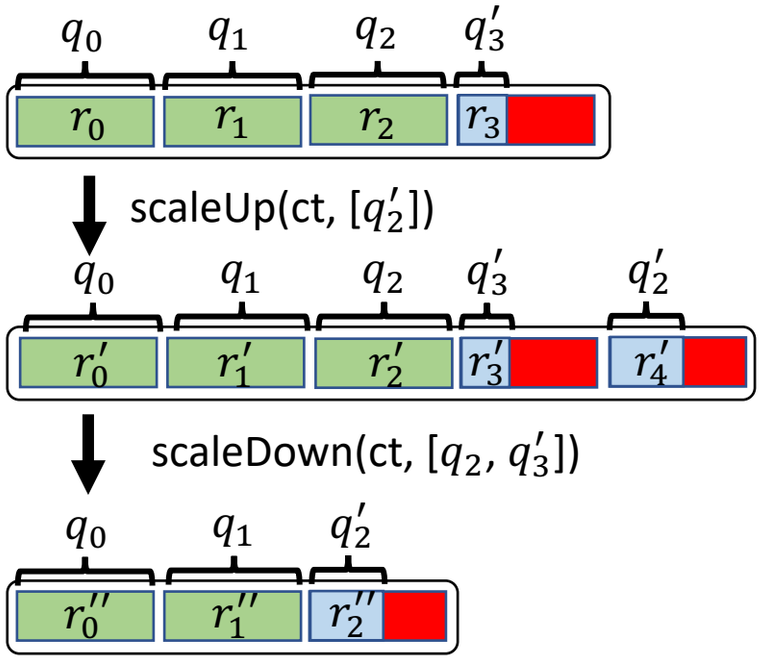


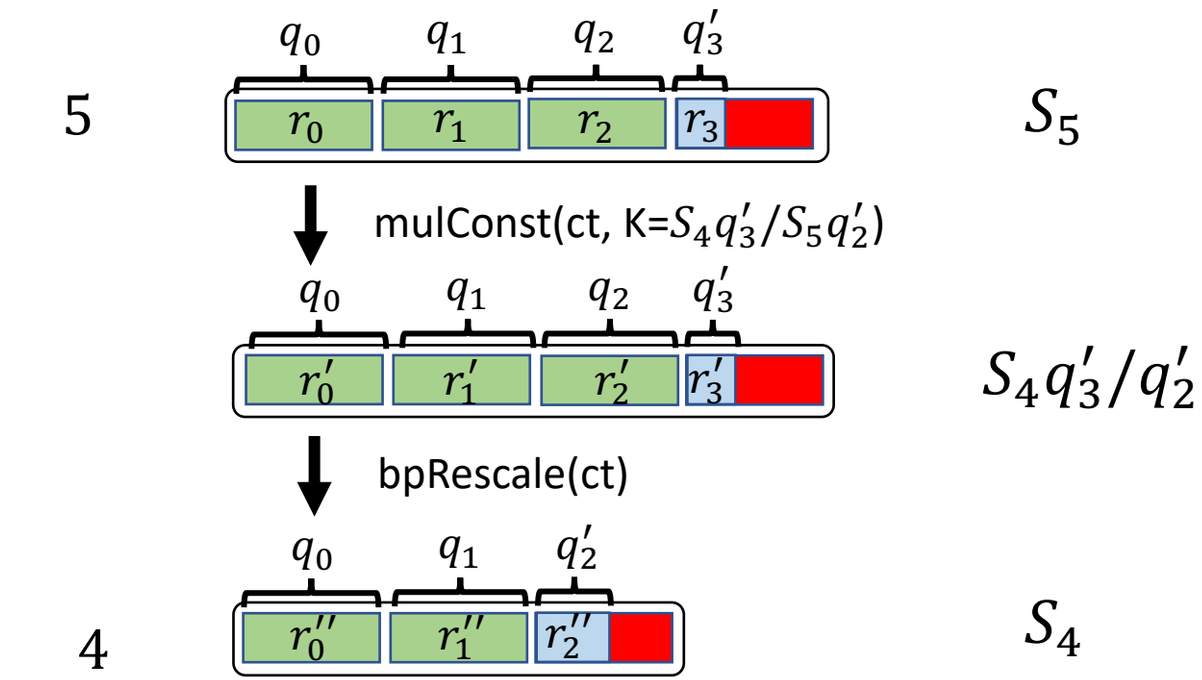
(b) BitPacker (this work)

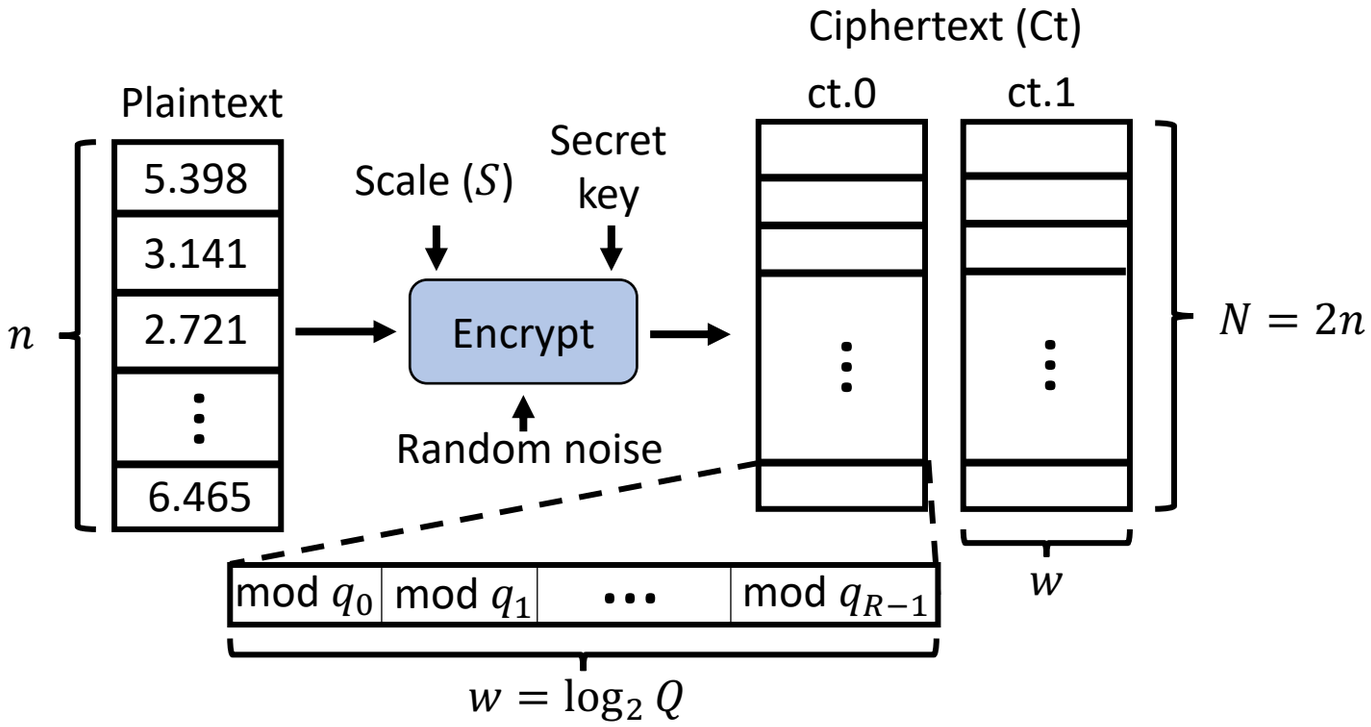

User
Accelerator
FHE PROGRAM
input: "x"
Level: 2
multiply
Level: 2
modDown
Level: 1
rescale
Level: 1
add
Level: 1
output
Level: 1
PROGRAM CONSTRAINTS
(1) Polynomial size: N
(2) Max level: L_{max}
(3) Max modulus: Q_{max}
(4) Min modulus: Q_{min}
(5) Level → Target scale ($L \rightarrow T_L$)
HW CONSTRAINT
(1) Word width: w
LEVEL-TO-MODULUS MAP
(1) Level → Modulus ($L \rightarrow Q_L$)
(2) Scale at maximum level: T_{Lmax}
(3) Level → Scale ($L \rightarrow S_L$) (implied from (1), (2))
Such that:
- All residue moduli fit in wordsize
- T_L*sqrt(2)/2 < S_L < T_L*sqrt(2), Q_{min} < Q_L < Q_{max} for all L

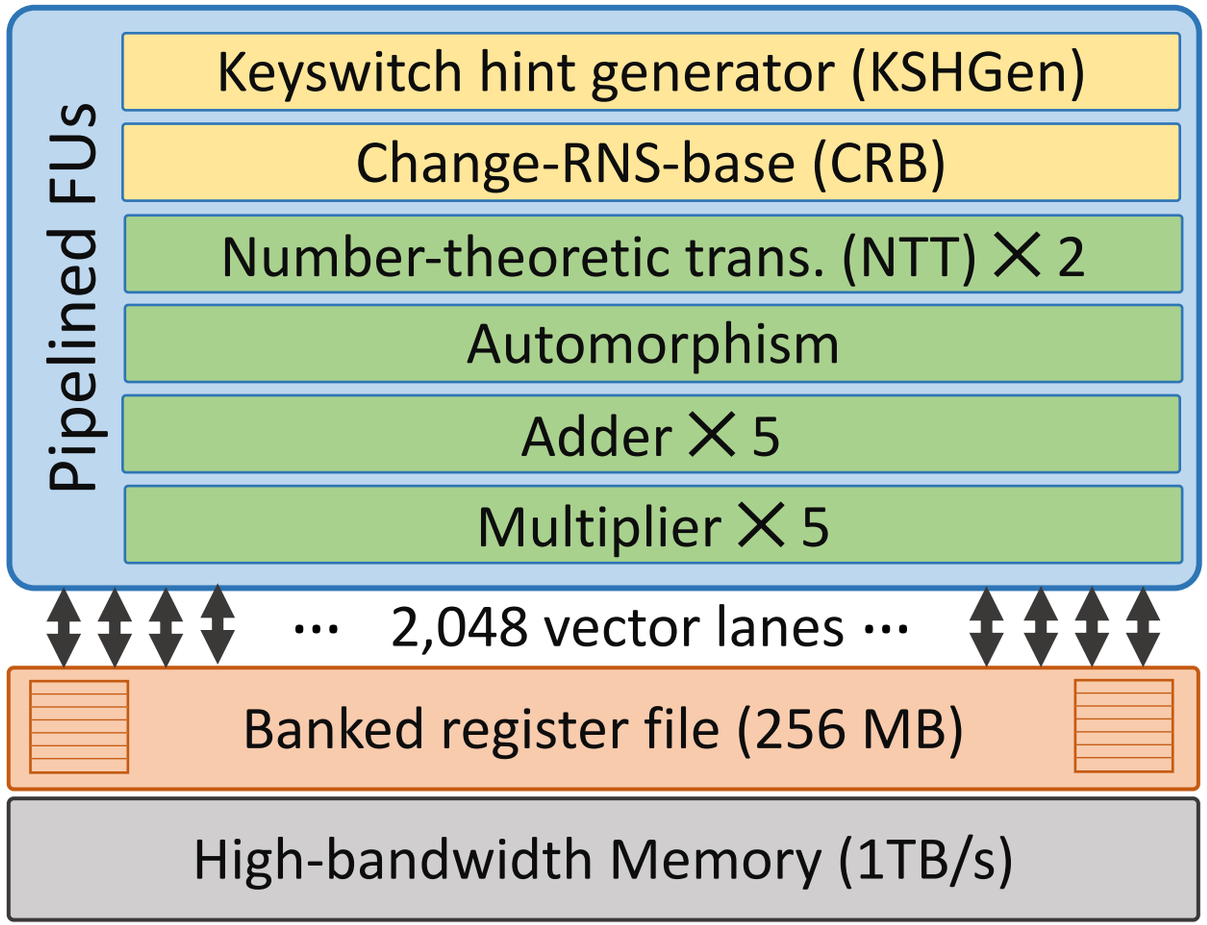
Level	Modulus	Scale
6		S_6
5		$S_5 = S_6^2/q_5$
4		$S_4 = S_5^2/q_4$
⋮	⋮	⋮

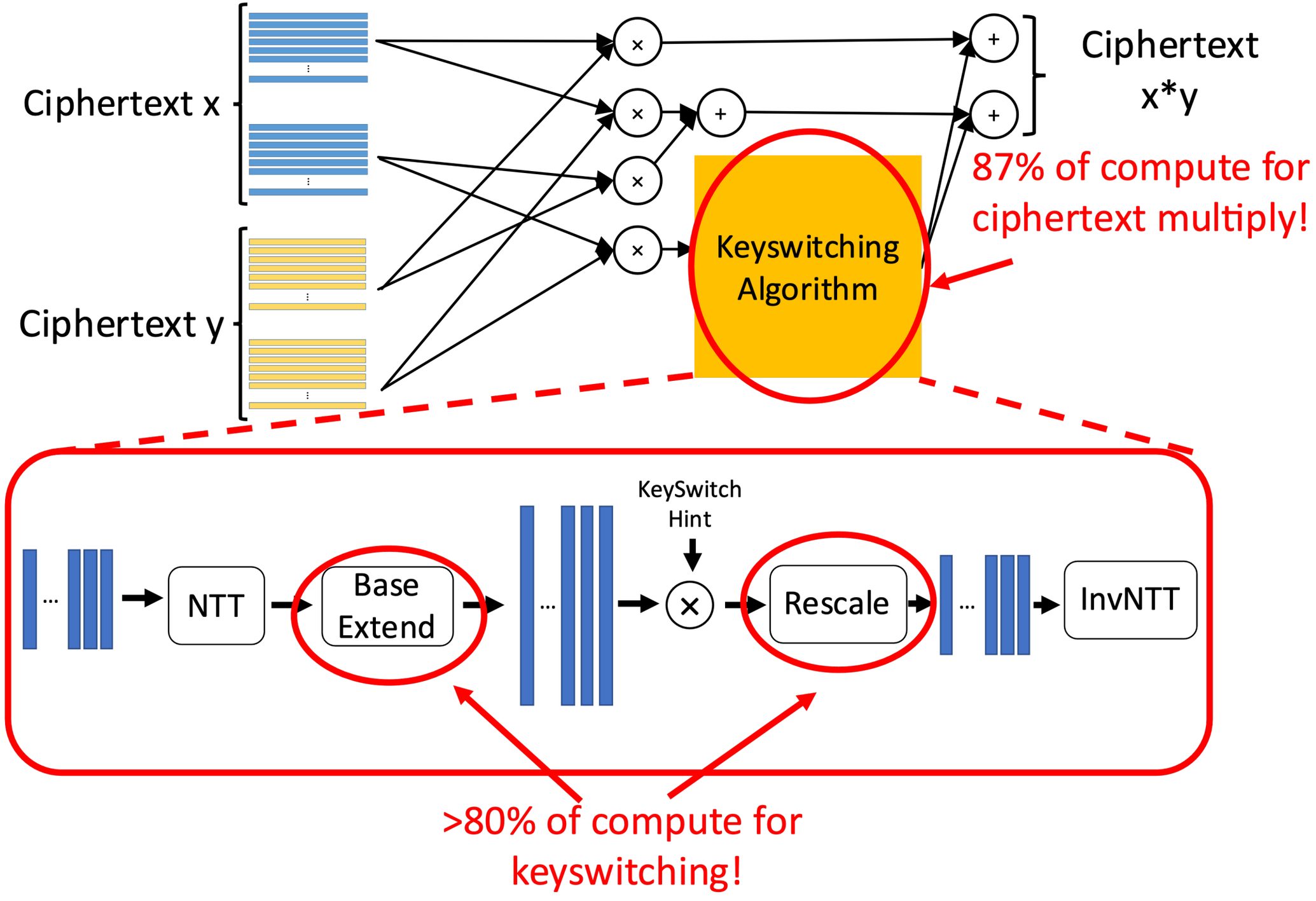
Level	Modulus	Scale
6		S_6
5		$S_5 = S_6^2 q_3'/q_3$
4		$S_4 = S_5^2 q_2'/q_2q_3'$
⋮	⋮	⋮

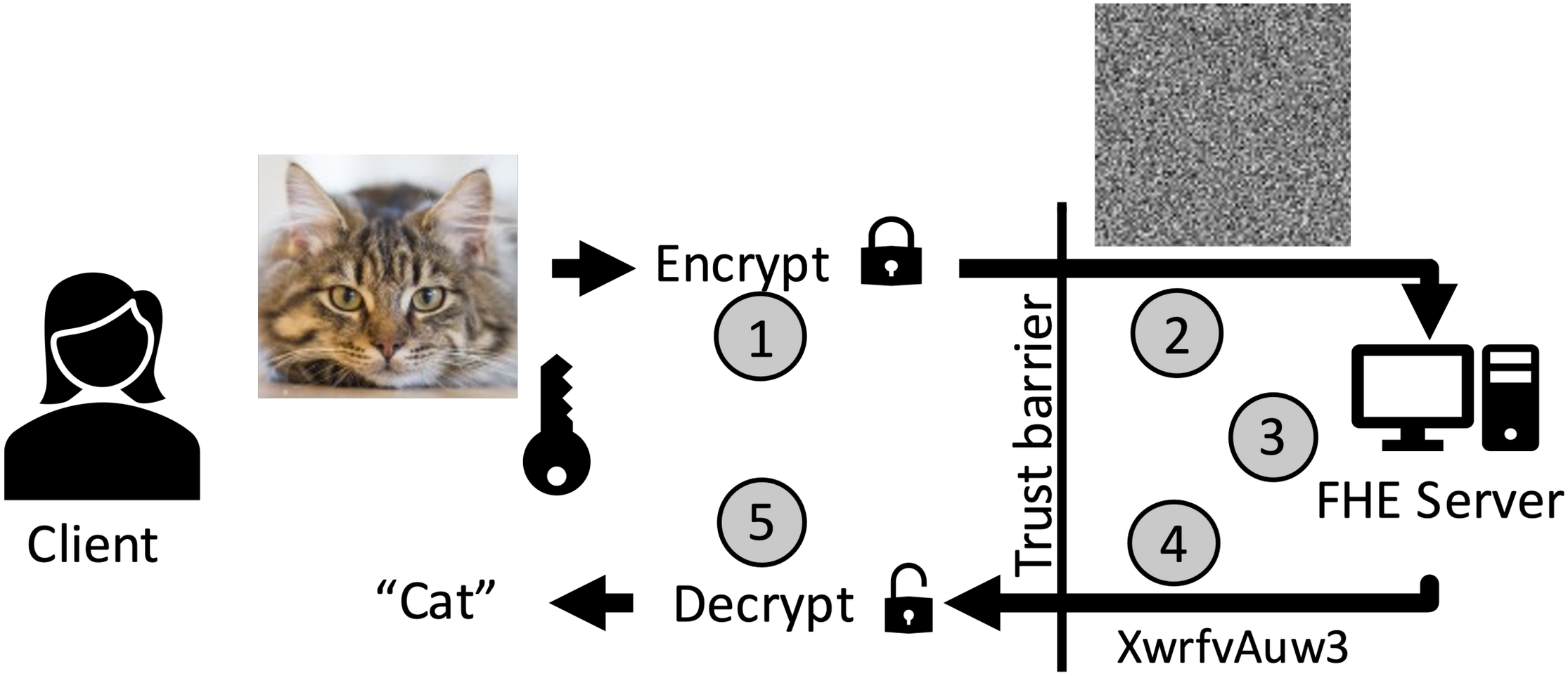
Level	Modulus	Scale
5		S_5^2
		$S_5^2 q_2'$
4		$S_4 = S_5^2 q_2' / q_2 q_3'$


Level
Modulus
Scale
q_0 q_1 q_2 q'_3
5
r_0 r_1 r_2 r_3
S_5
mulConst(ct, K=$S_4 q'_3 / S_5 q'_2$)
q_0 q_1 q_2 q'_3
r'_0 r'_1 r'_2 r'_3
$S_4 q'_3 / q'_2$
bpRescale(ct)
q_0 q_1 q'_2
4
r''_0 r''_1 r''_2
S_4


Ciphertext (Ct)
ct.0
ct.1
Plaintext
5.398
3.141
2.721
6.465
n
Scale (S)
Secret key
Encrypt
Random noise
$N = 2n$
mod q_0
mod q_1
mod q_{R-1}
w
$w = \log_2 Q$

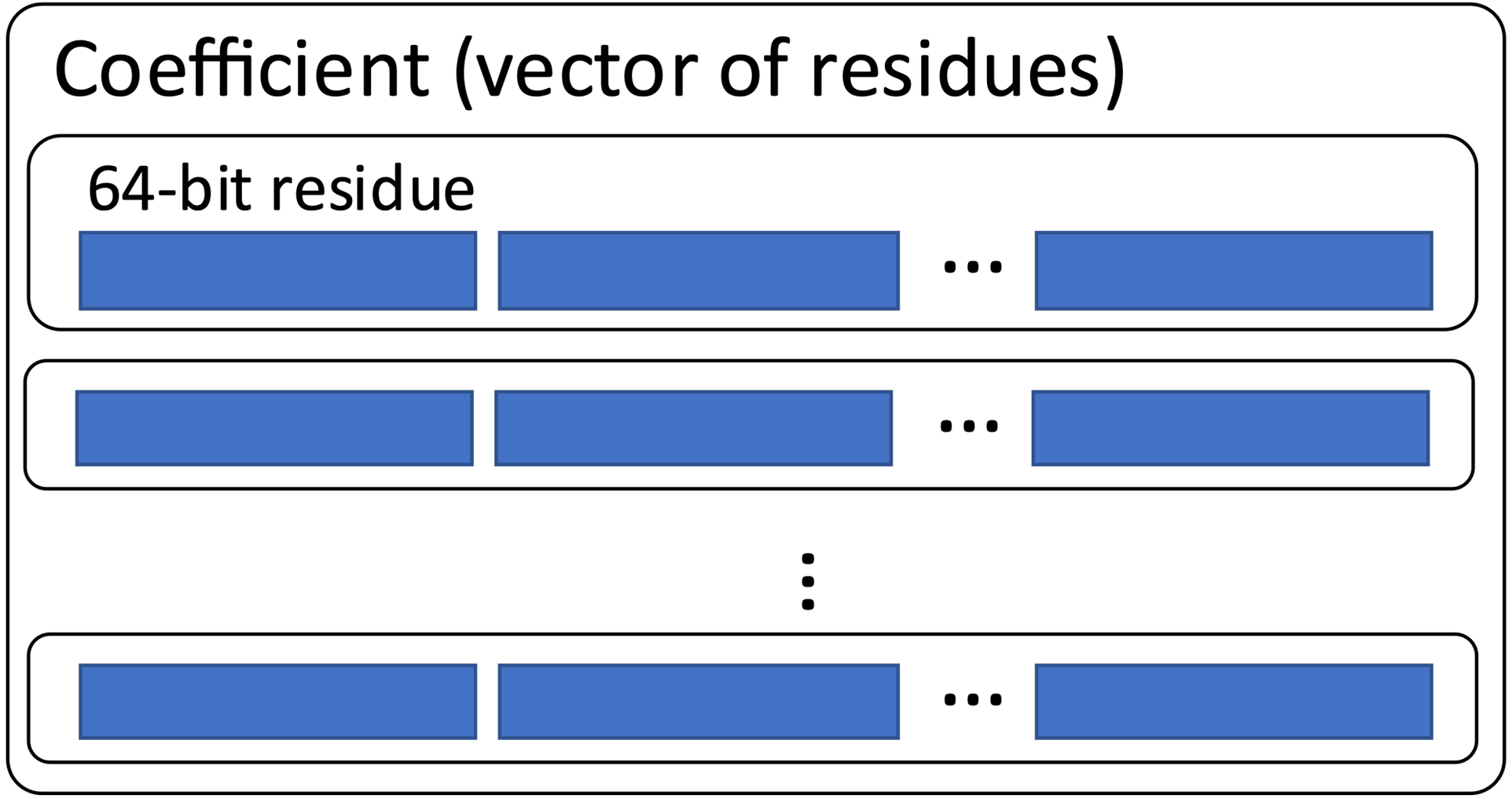

Pipelined FUs
Keyswitch hint generator (KSHGen)
Change-RNS-base (CRB)
Number-theoretic trans. (NTT) ✕ 2
Automorphism
Adder ✕ 5
Multiplier ✕ 5
··· 2,048 vector lanes ···
Banked register file (256 MB)
High-bandwidth Memory (1TB/s)


Ciphertext x
Ciphertext y
Keyswitching Algorithm
Ciphertext x*y
87% of compute for ciphertext multiply!
KeySwitch Hint
NTT
Base Extend
Rescale
InvNTT
>80% of compute for keyswitching!


Client
Encrypt
1
Trust barrier
2
3
FHE Server
4
XwrfvAuw3
5
Decrypt
“Cat”

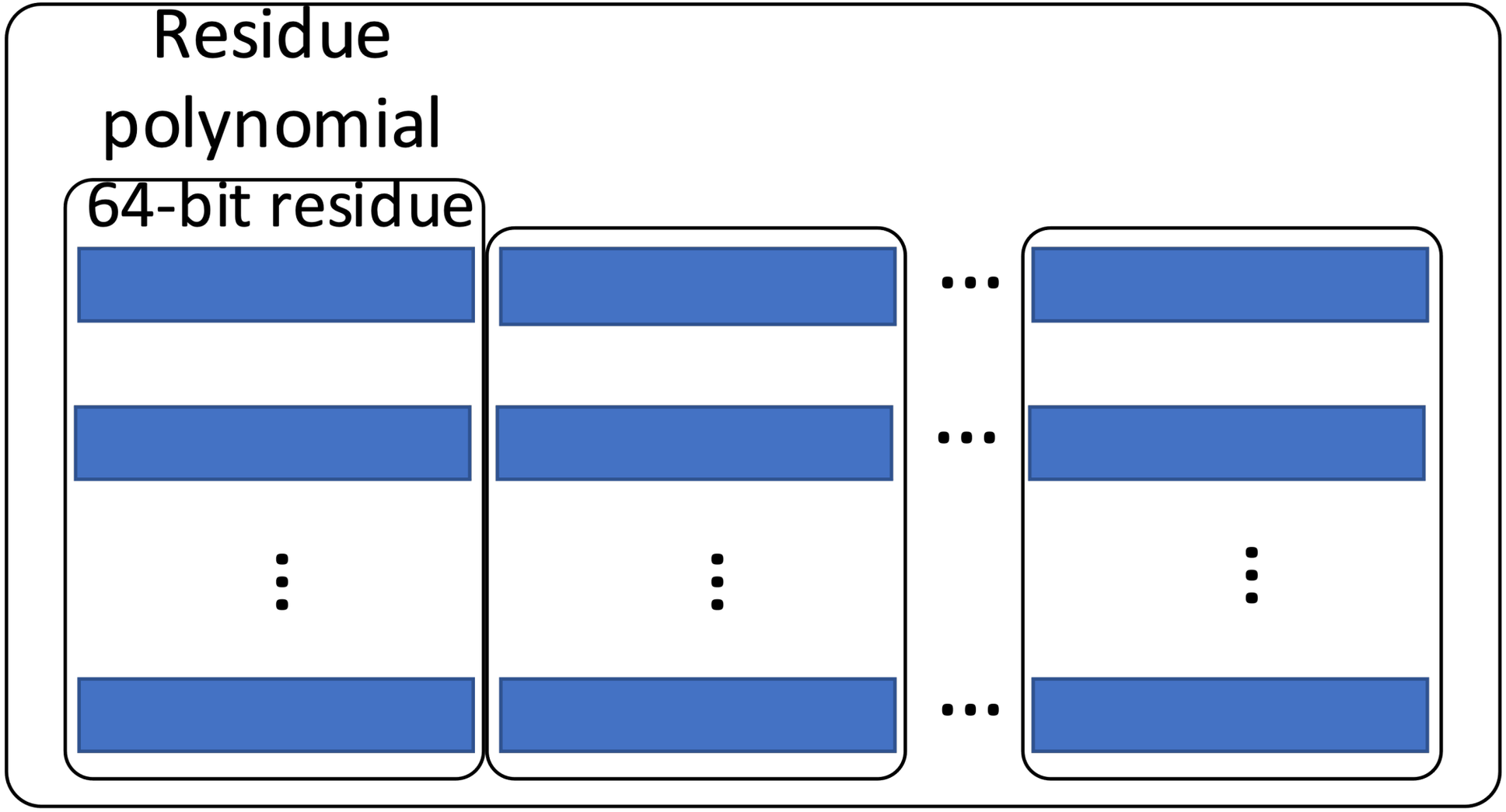
Old Polynomial Layout

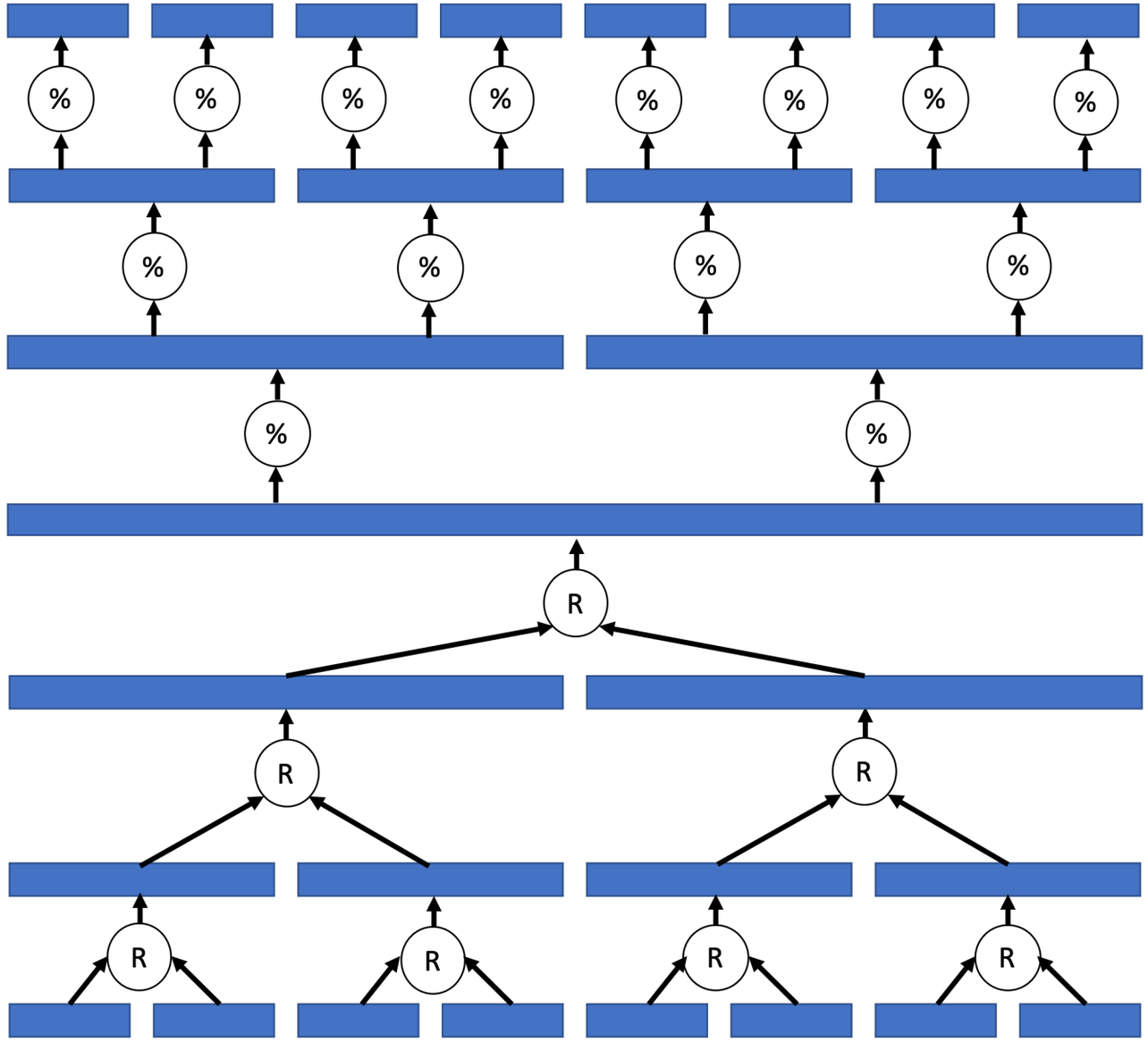
Polynomial (vector of RnsInt's)

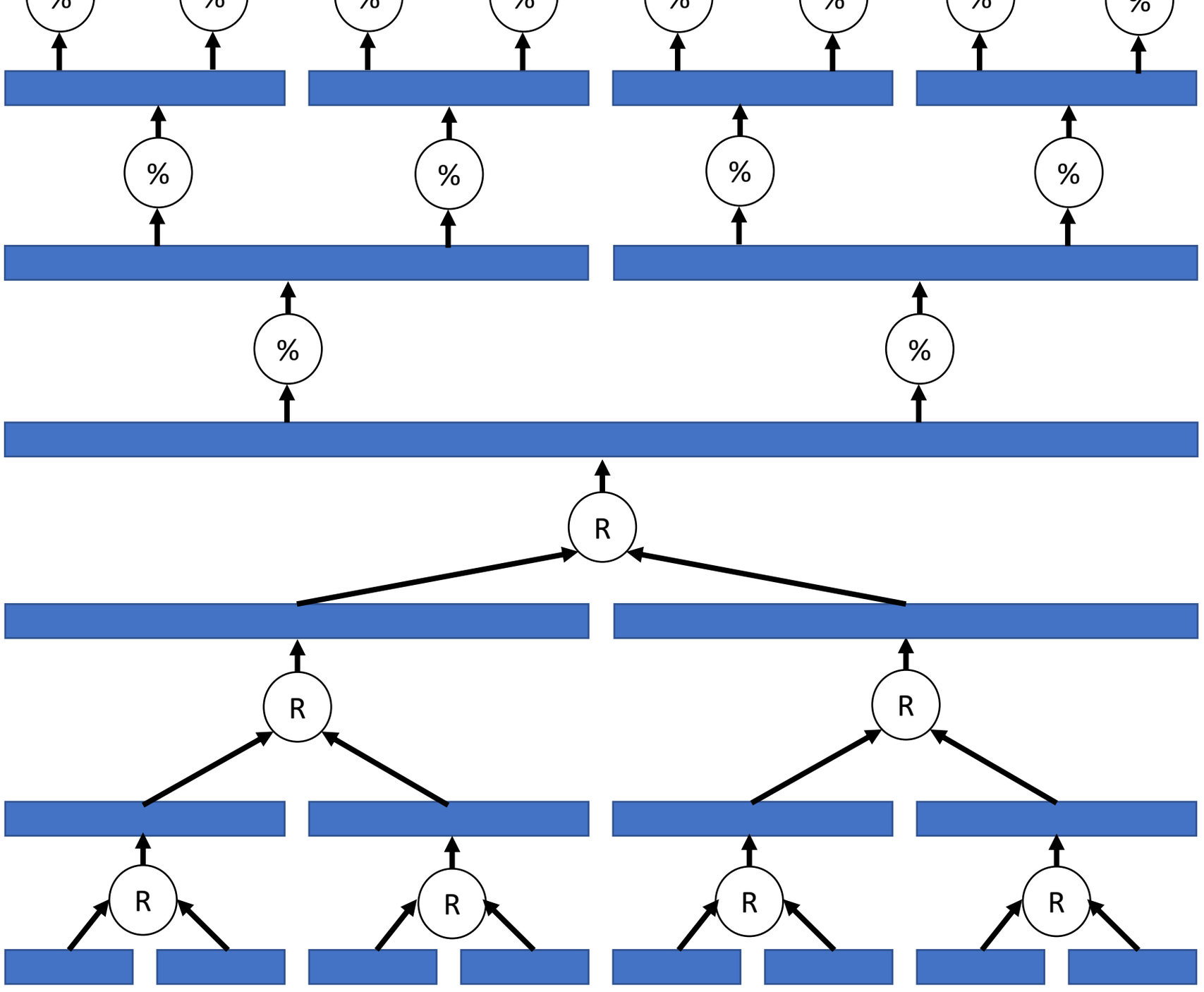


New Polynomial Layout

Polynomial (vector of residue polynomials)




%
%
%
%
%
%
%
%
%
%
%
%
%
%
%
R
R
R
R
R
R
R


%
%
%
%
%
%
%
%
%
%
R
R
R
R
R
R
R