How to gamble arithmetic circuits

如何混淆代数电路

刘天任 北京大学

Based on joint works with

Marshall Ball
Columbia

Hanjun Li
UW

Rachel Lin
UW

Garbled Circuits (GC)

混淆电路

- 由姚期智先生发明 [Yao86]

Garbled Circuits (GC)

混淆电路

- ☆ 由姚期智先生发明 [Yao86]
- ☆ 非常好用（理论方面）
- ☆ 非常有用（应用方面）

Garbled Circuits (GC)

混淆电路

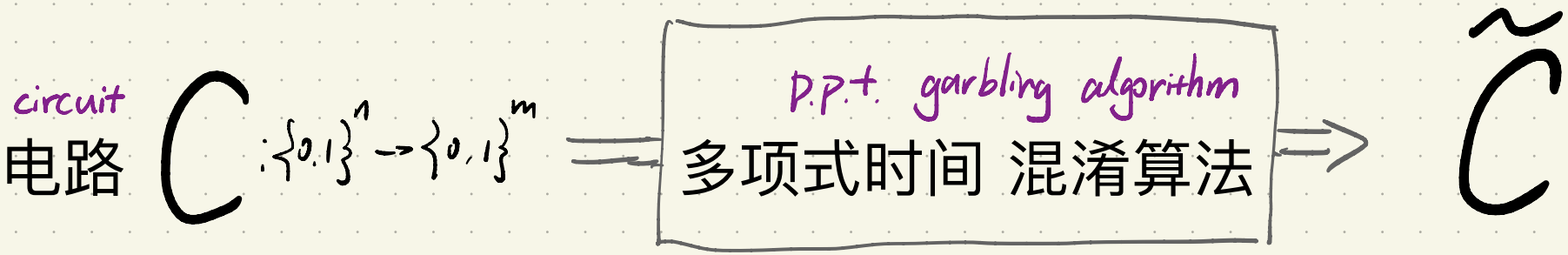
- ☆ 由姚期智先生发明 [Yao86]
- ☆ 非常好用（理论方面）
- ☆ 非常有用（应用方面）
- ☆ 效率瓶颈：大量的优化工作

FreeXOR, FleXOR

HalfGate, ThreeHalves

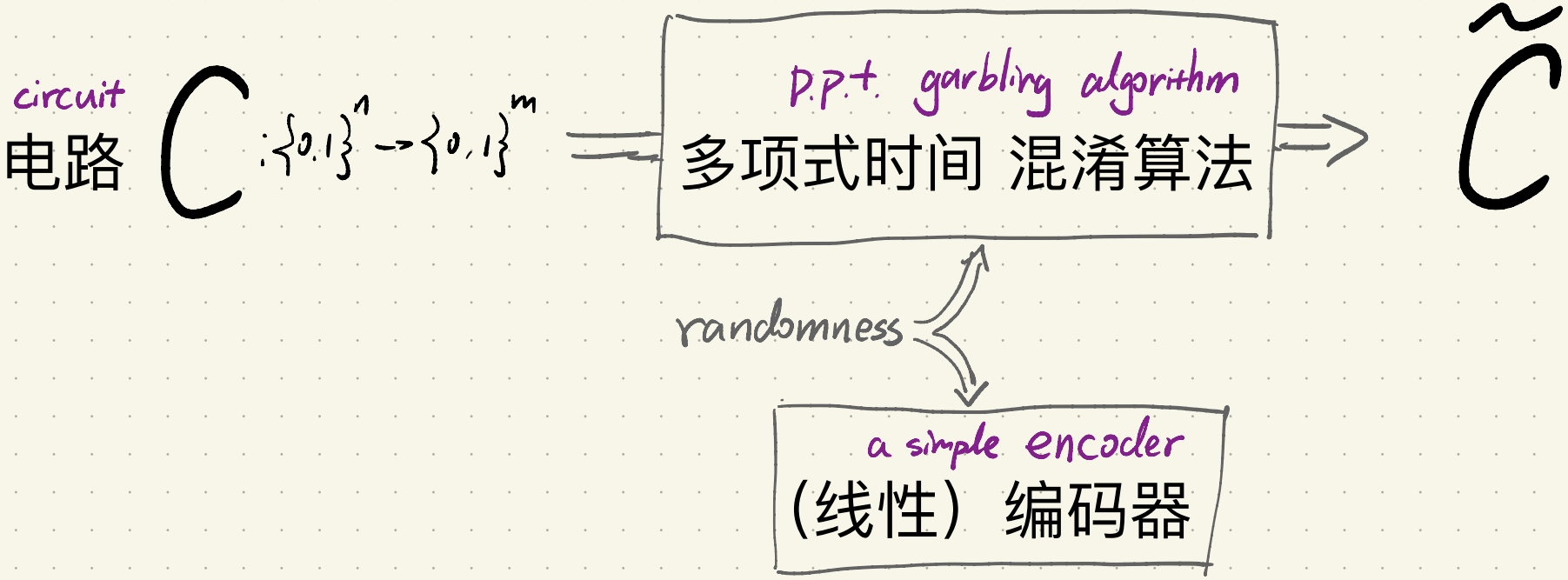
Garbled Circuits (GC)

混淆电路



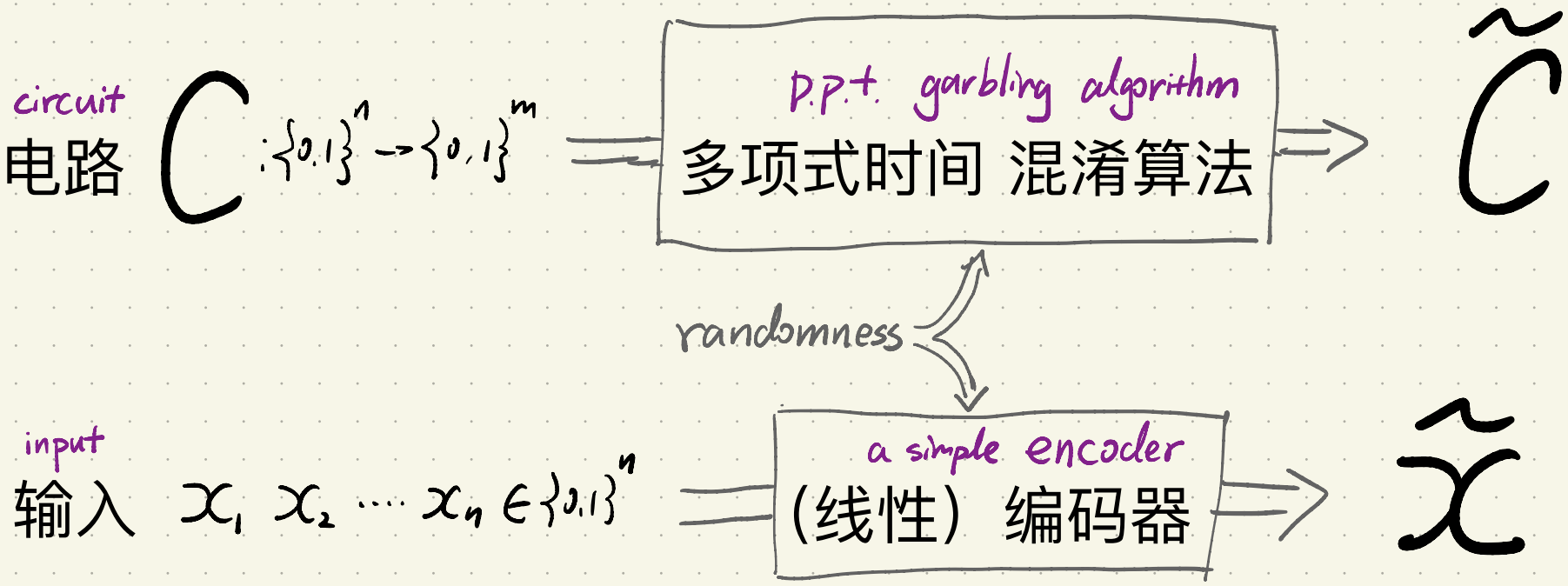
Garbled Circuits (GC)

混淆电路



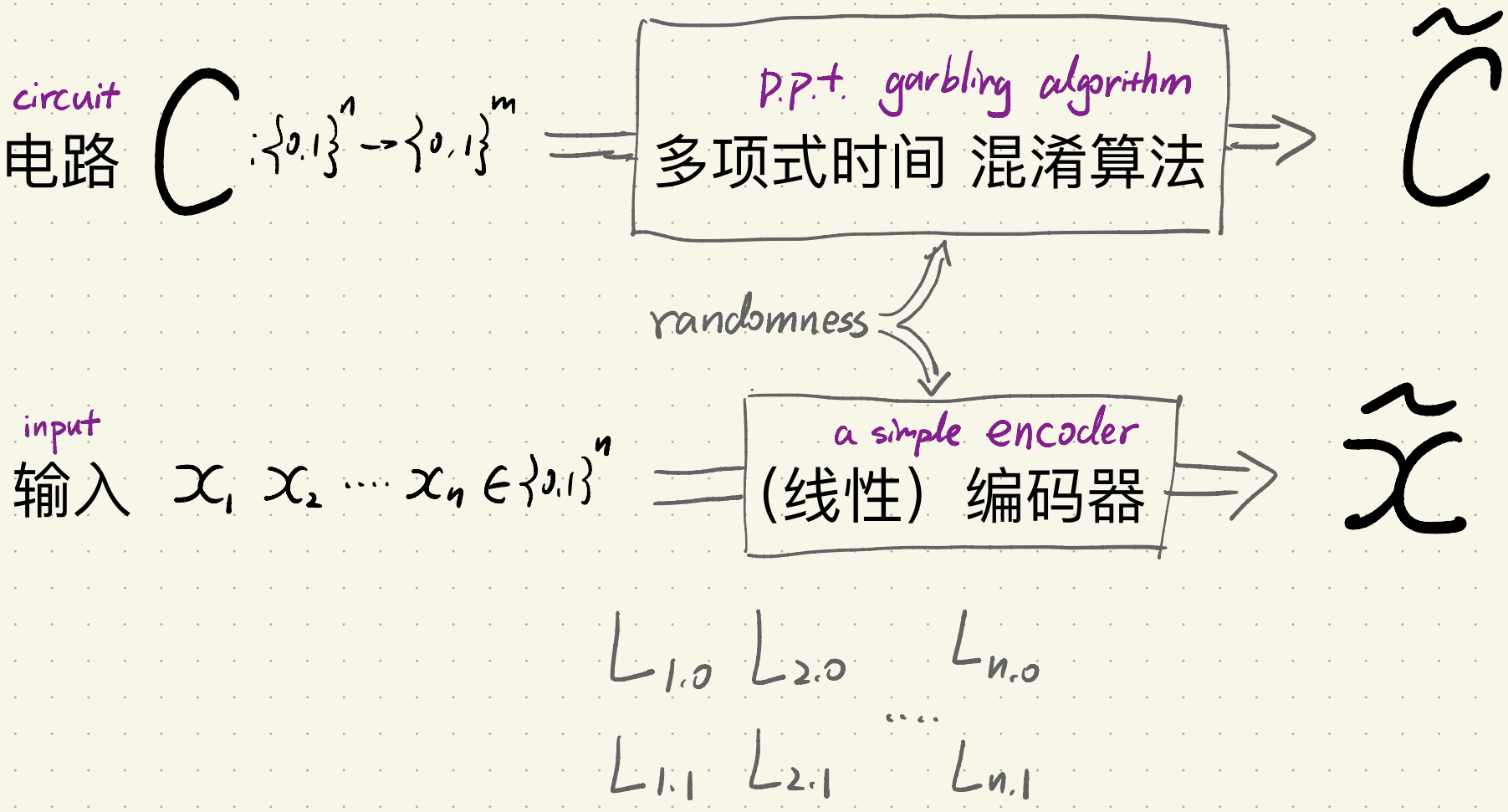
Garbled Circuits (GC)

混淆电路



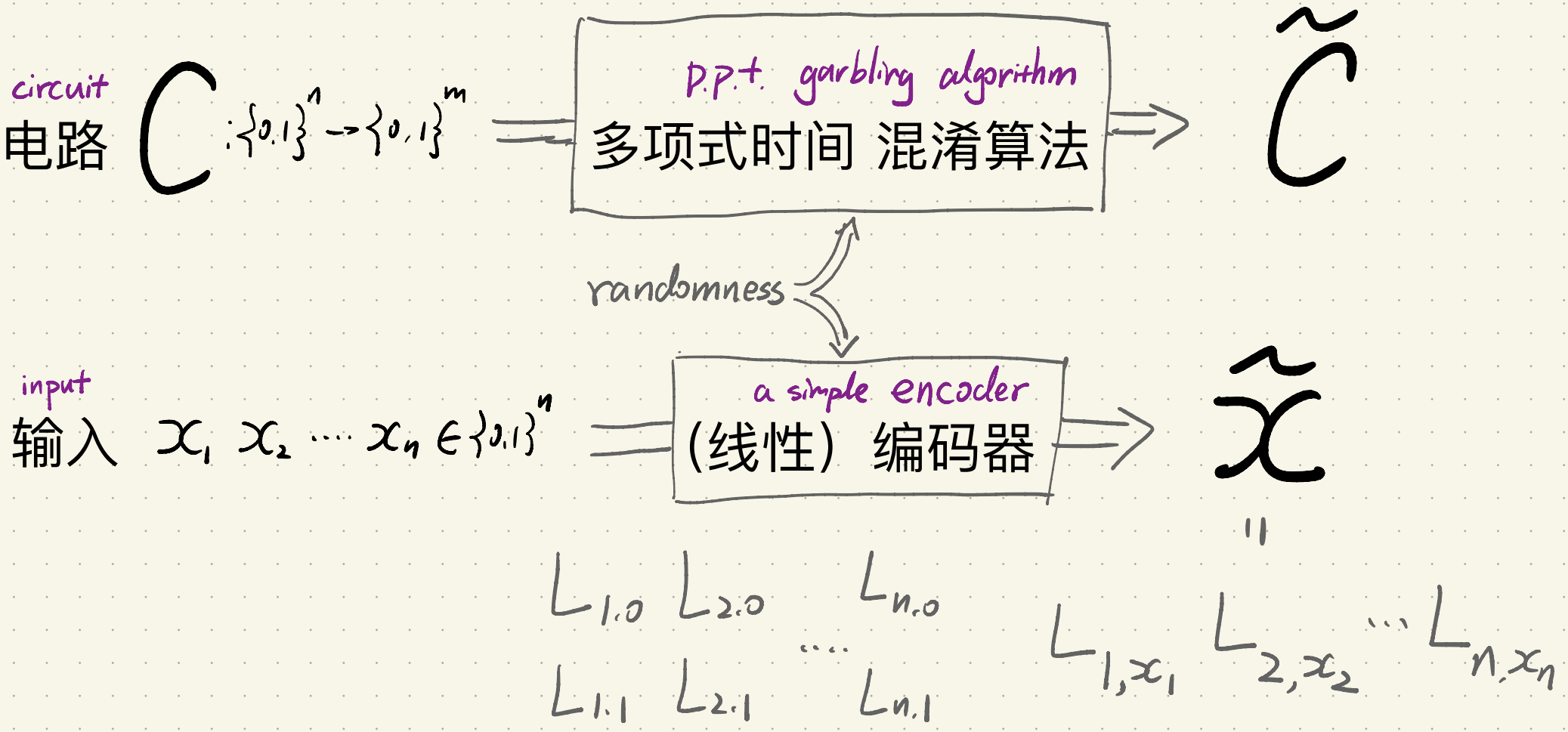
Garbled Circuits (GC)

混淆电路



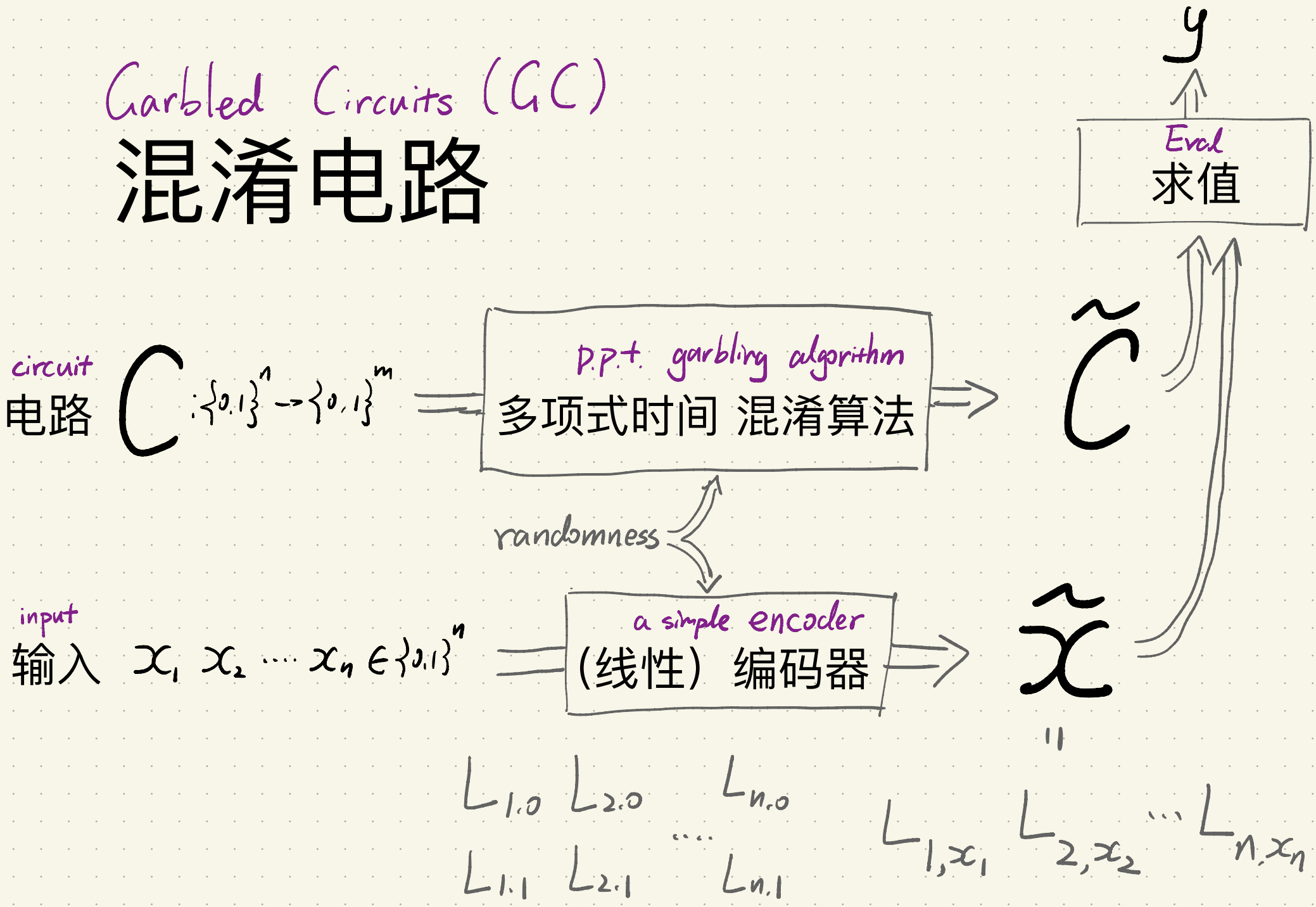
Garbled Circuits (GC)

混淆电路



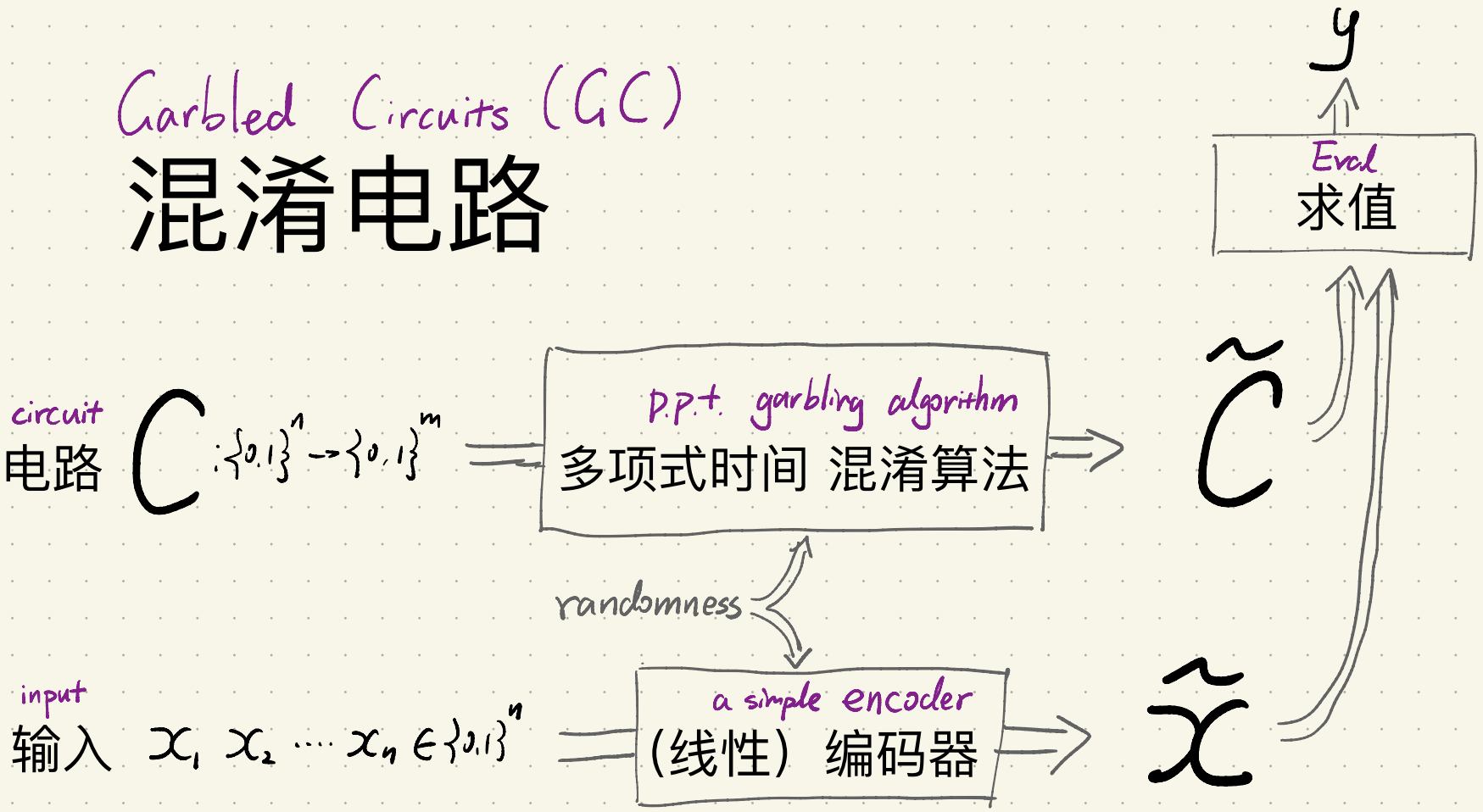
Garbled Circuits (GC)

混淆电路



Garbled Circuits (GC)

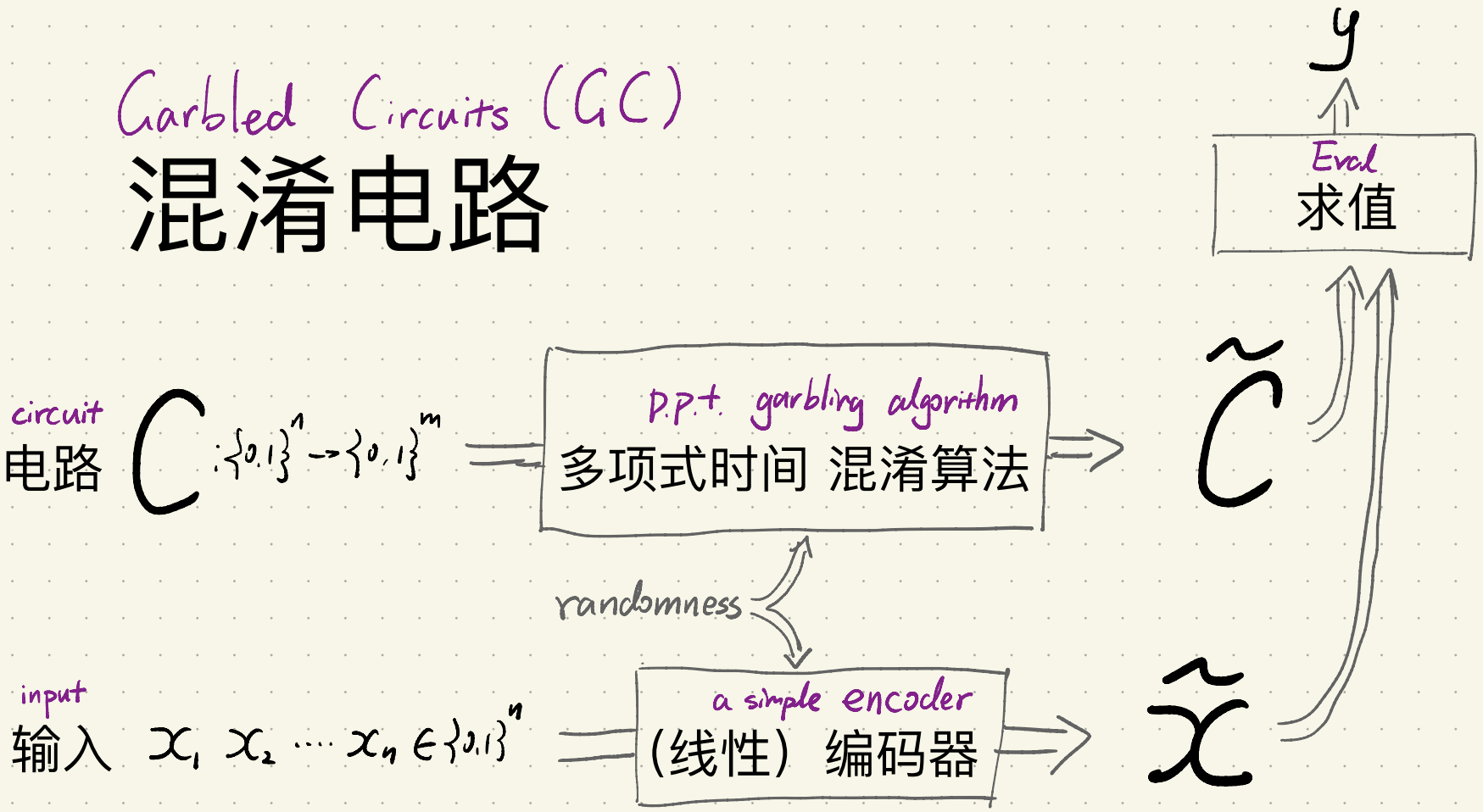
混淆电路



- 正确性
- 安全性

Garbled Circuits (GC)

混淆电路

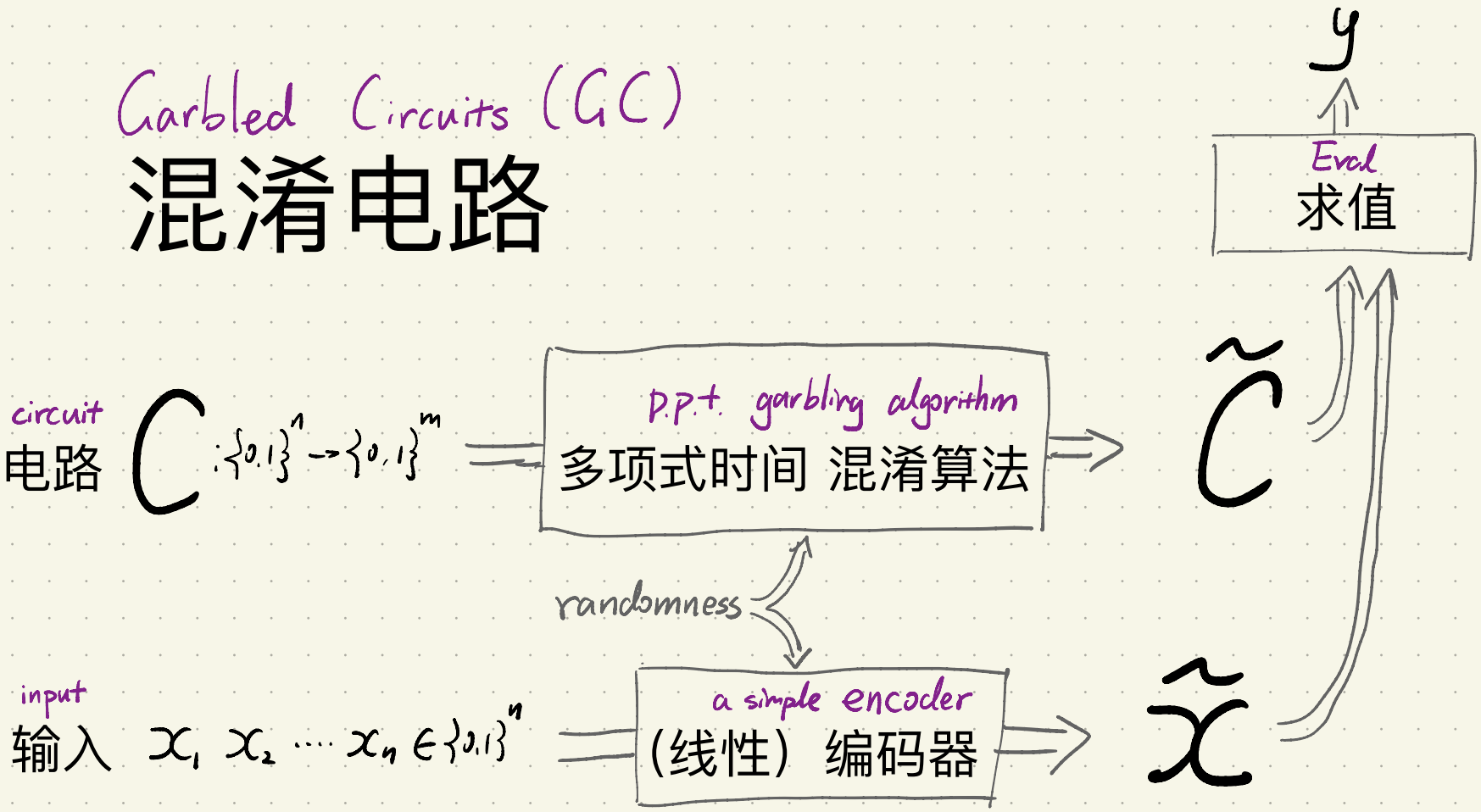


- 正确性 $y = C(x)$

- 安全性

Garbled Circuits (GC)

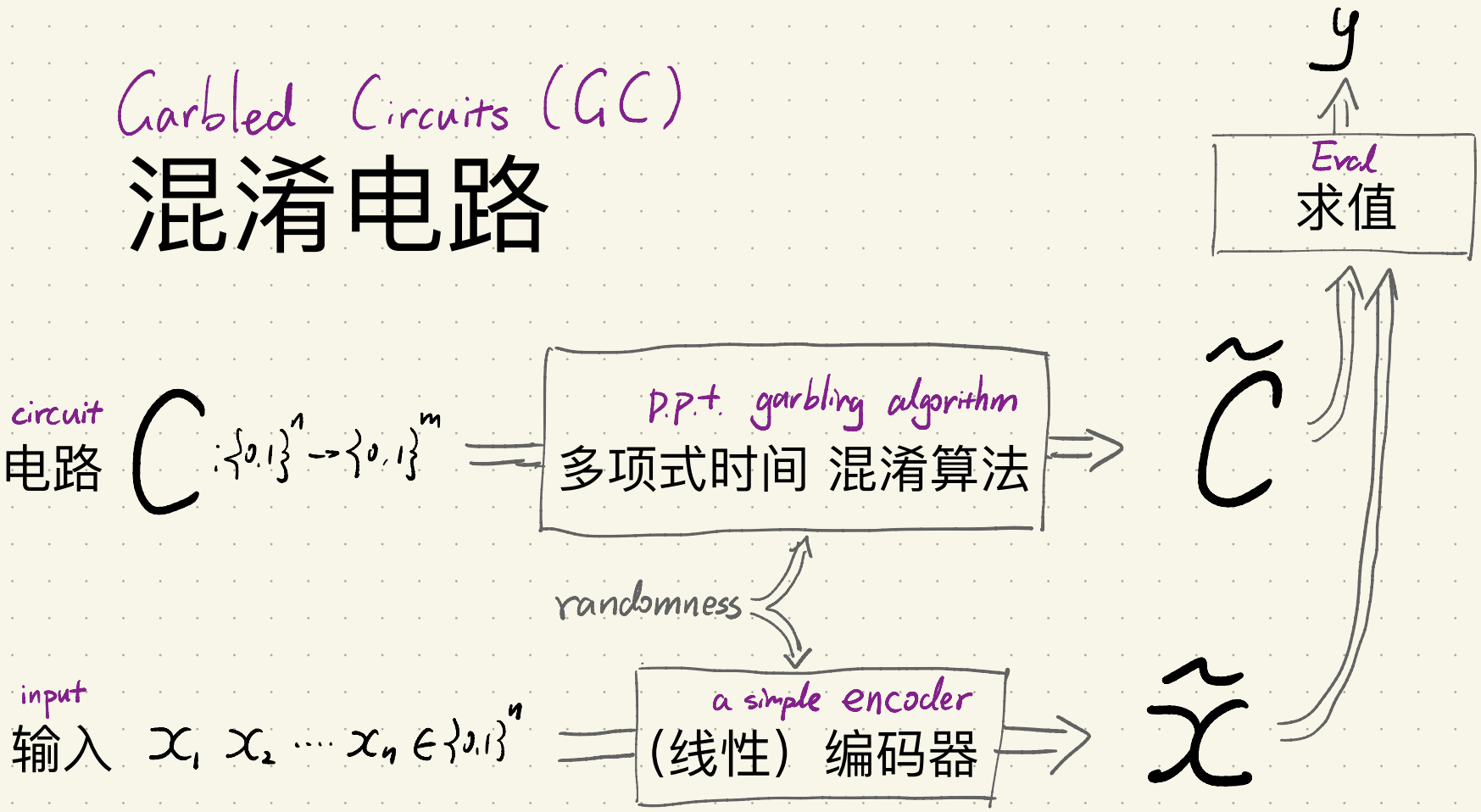
混淆电路



- 正确性 $y = C(x)$
- 安全性 "Leak no information about x, besides $C(x)$"

Garbled Circuits (GC)

混淆电路



- 正确性 $y = C(x)$

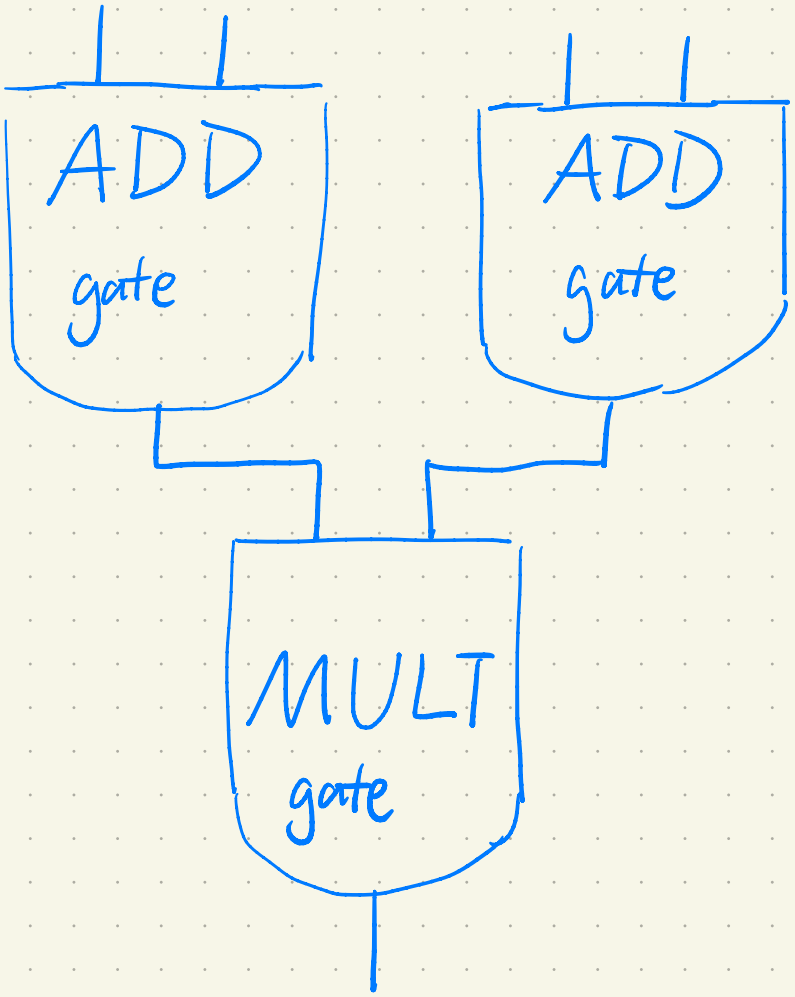
- 安全性 "Leak no information about x, besides $C(x)$"

$$\exists \text{ p.p.t. } Sim \quad (\tilde{C}, \tilde{x}) \approx_c Sim(C, C(x))$$

Arithmetic Circuits

代数电路

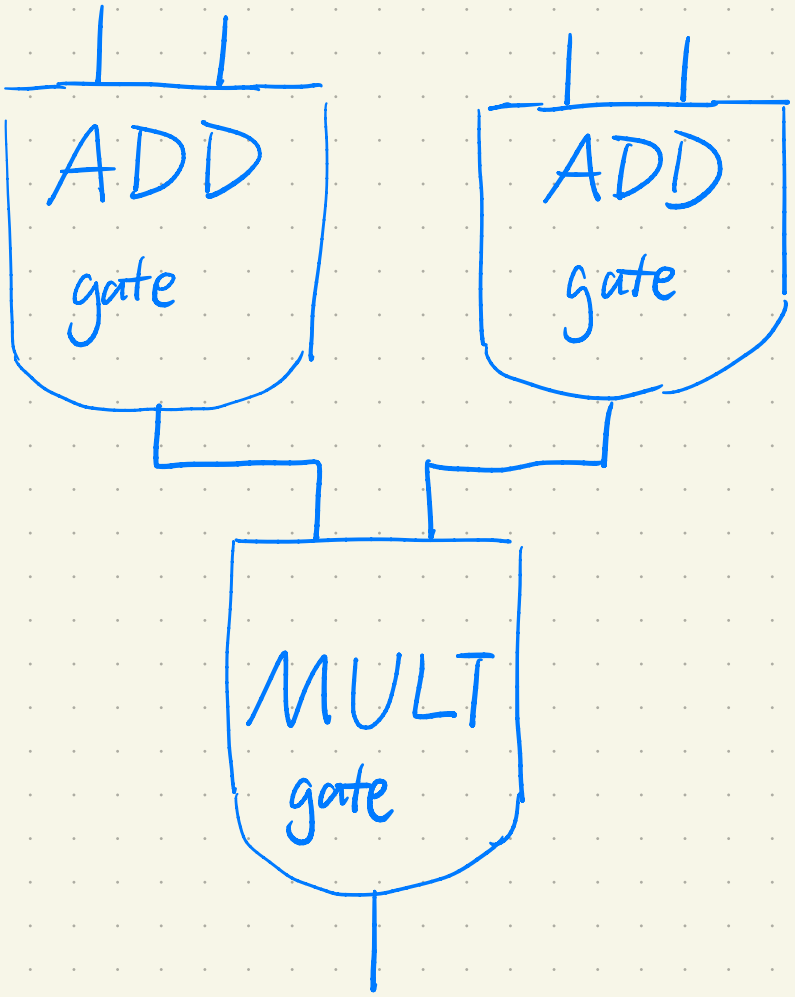
支持某个环（e.g. $\mathbb{Z}_{2^b}$）上的代数运算



Garble Arithmetic Circuits

混淆 代数电路

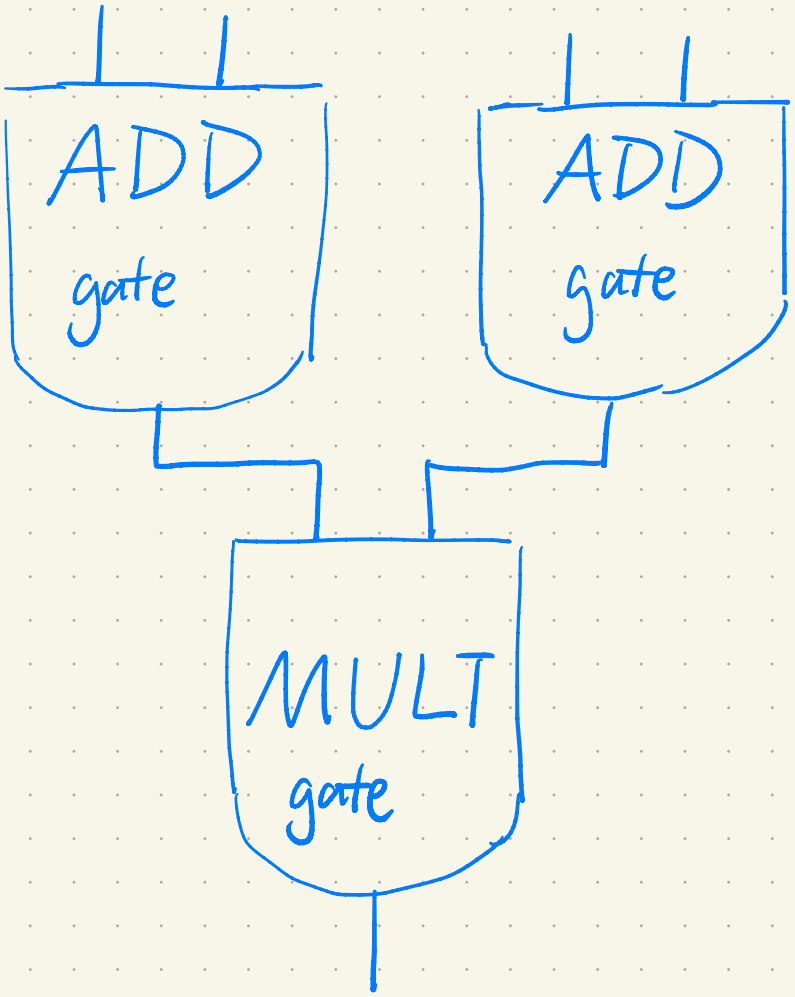
支持某个环（e.g. $\mathbb{Z}_{2^b}$）上的代数运算



Garble Arithmetic Circuits

混淆 代数电路

支持某个环（e.g. $\mathbb{Z}_{2^b}$）上的代数运算



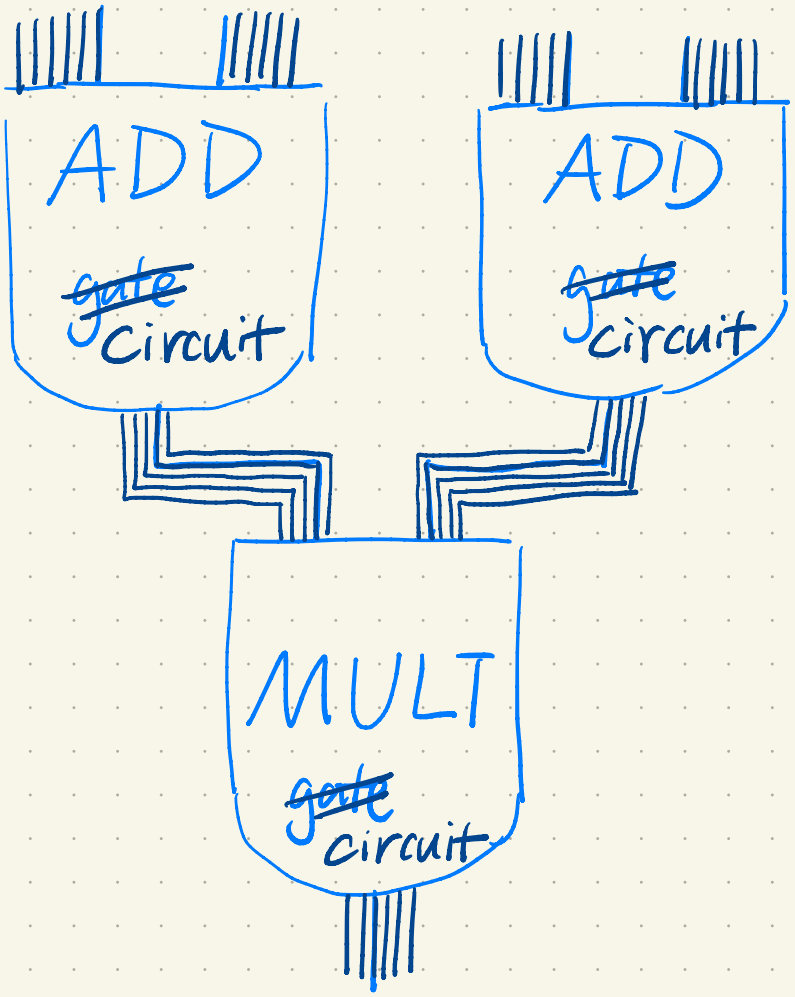
Baseline Solution:

混淆对应的布尔电路

Garble Arithmetic Circuits

混淆 代数电路

支持某个环（e.g. $\mathbb{Z}_{2^b}$）上的代数运算



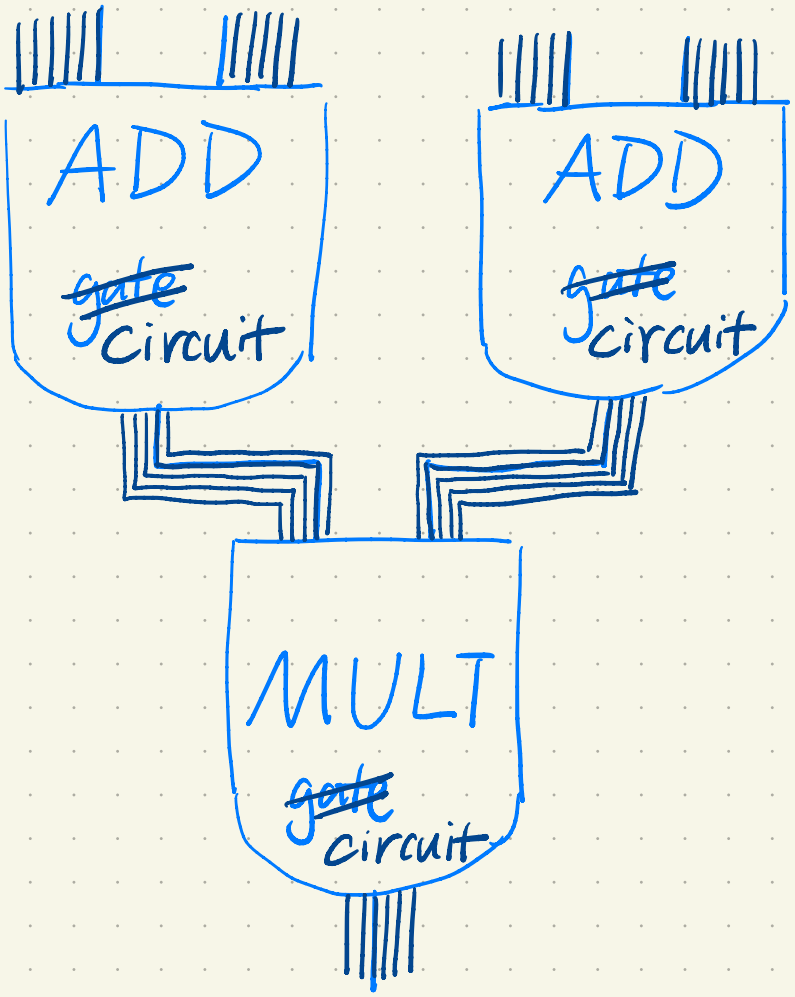
Baseline Solution:

混淆对应的布尔电路

Garble Arithmetic Circuits

混淆 代数电路

支持某个环（e.g. $\mathbb{Z}_{2^b}$）上的代数运算



Baseline Solution:

混淆对应的布尔电路

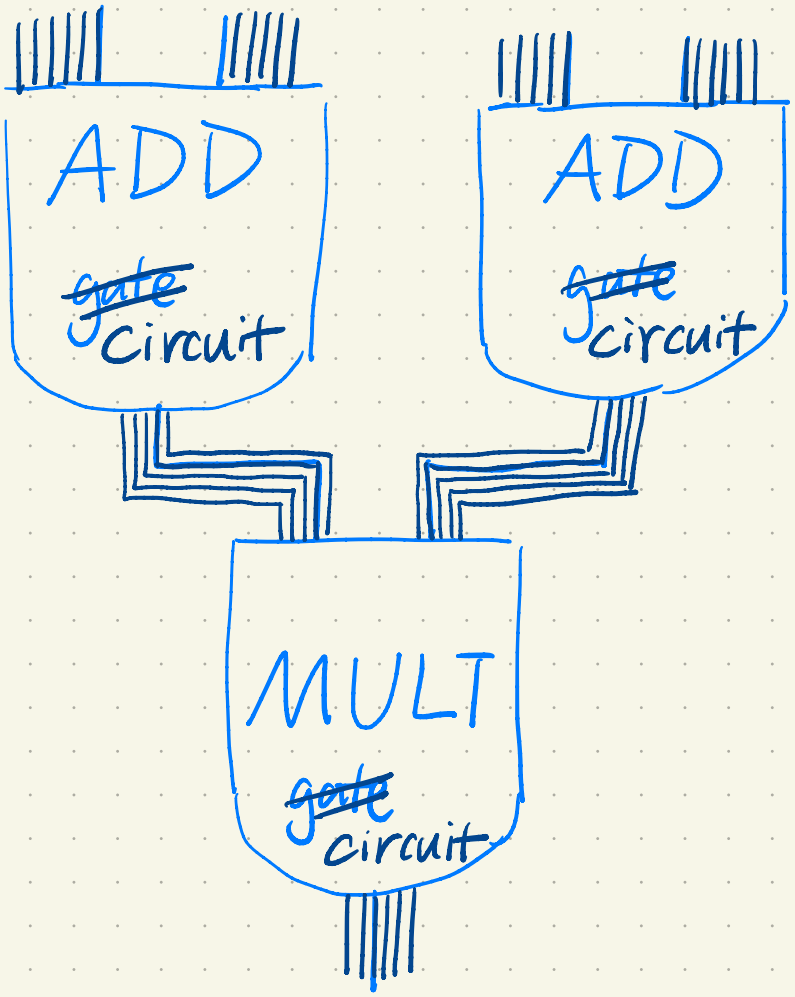
The cost?

代价是什么？

Garble Arithmetic Circuits

混淆 代数电路

支持某个环（e.g. $\mathbb{Z}_{2^b}$）上的代数运算



Baseline Solution:

混淆对应的布尔电路

The cost?

代价是什么？

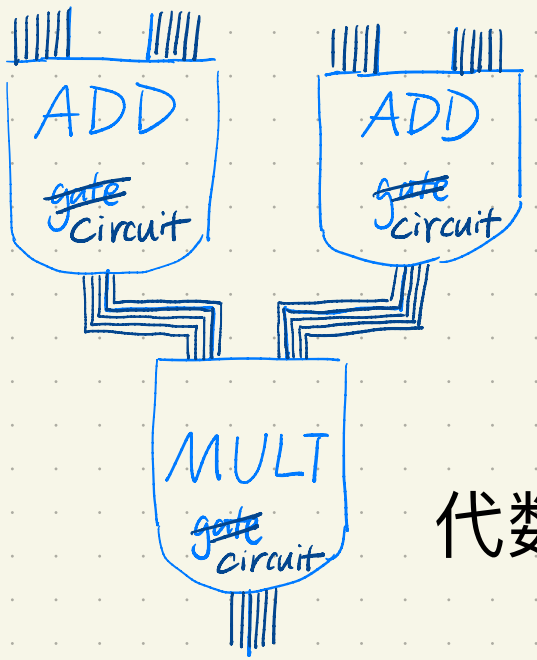
~~计算复杂性~~

通讯/存储复杂性

Garble Arithmetic Circuits

混淆 代数电路

支持某个环（e.g. $\mathbb{Z}_{2^b}$）上的代数运算



Baseline Solution:

混淆对应的布尔电路

The cost?

代价是什么?

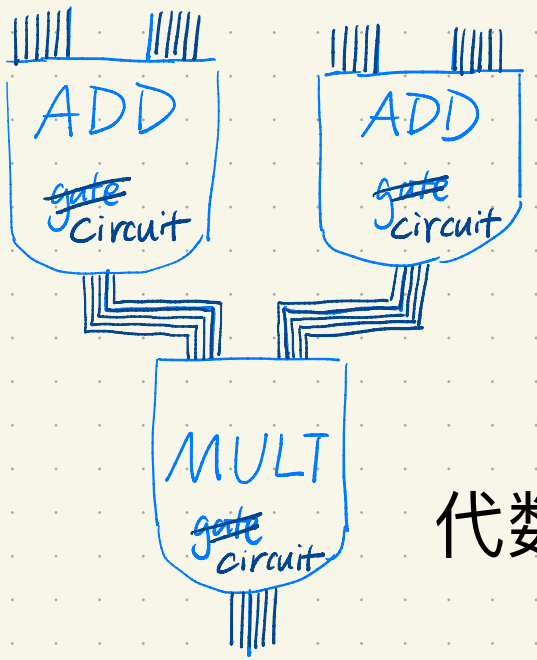
代数加法门 $\Longrightarrow$ $O(b)$个布尔门 $\Longrightarrow$ 代价：$O(b \cdot \lambda)$

λ = security parameter

Garble Arithmetic Circuits

混淆 代数电路

支持某个环（e.g. $\mathbb{Z}_{2^b}$）上的代数运算



Baseline Solution:

混淆对应的布尔电路

The cost?

代价是什么?

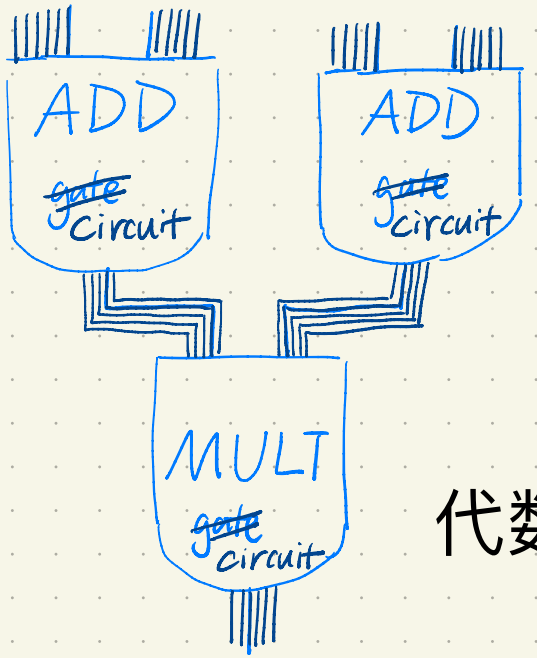
代数加法门 $\Longrightarrow$ $O(b)$个布尔门 $\Longrightarrow$ 代价：$O(b \cdot \lambda)$

代数乘法门 $\Longrightarrow$ $O(b^2)$个布尔门 $\Longrightarrow$ 代价：$O(b^2 \cdot \lambda)$

Garble Arithmetic Circuits

混淆 代数电路

支持某个环（e.g. $\mathbb{Z}_{2^b}$）上的代数运算



Baseline Solution:

混淆对应的布尔电路

The cost?

代价是什么?

代数加法门 $\Longrightarrow$ $O(b)$个布尔门 $\Longrightarrow$ 代价: $O(b \cdot \lambda)$

代数乘法门 $\Longrightarrow$ $O(b^2)$个布尔门 $\Longrightarrow$ 代价: $O(b^2 \cdot \lambda)$

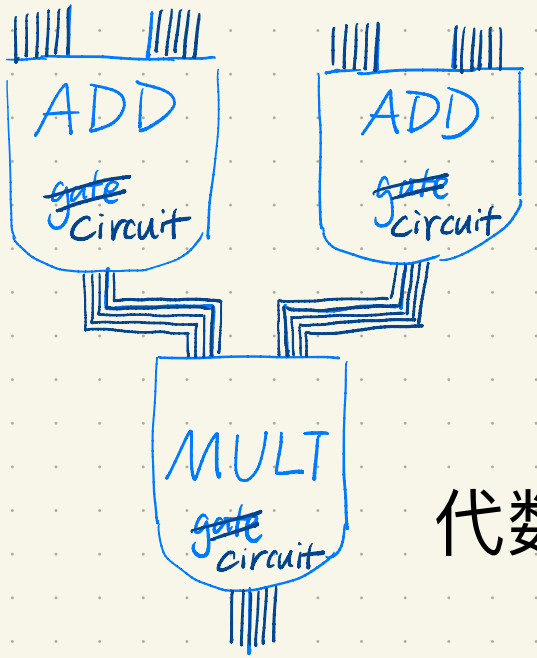
Karatsuba

$O(b^{1.58})$个布尔门 $\Longrightarrow$ 代价: $O(b^{1.58} \cdot \lambda)$

Garble Arithmetic Circuits

混淆 代数电路

支持某个环（e.g. $\mathbb{Z}_{2^b}$）上的代数运算



Baseline Solution:

混淆对应的布尔电路

The cost?

代价是什么?

代数加法门 $\Longrightarrow$ $O(b)$个布尔门 $\Longrightarrow$ 代价：$O(b \cdot \lambda)$

代数乘法门 $\Longrightarrow$ $O(b^2)$个布尔门 $\Longrightarrow$ 代价：$O(b^2 \cdot \lambda)$

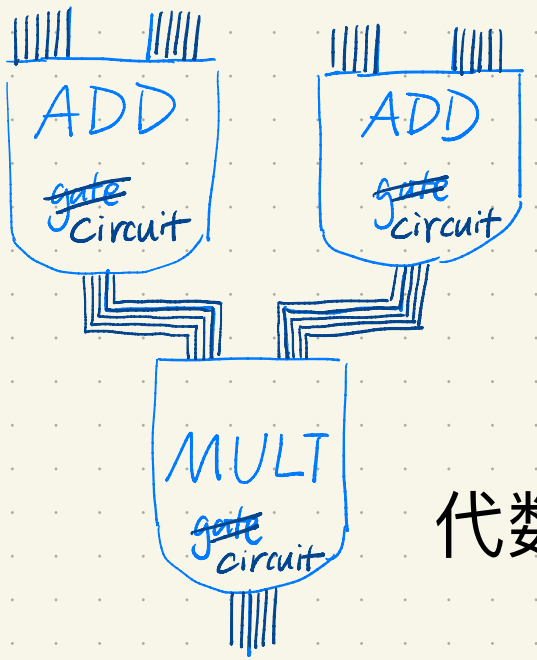
Karatsuba $\Longrightarrow$ $O(b^{1.58})$个布尔门 $\Longrightarrow$ 代价：$O(b^{1.58} \cdot \lambda)$

FFT $\Longrightarrow$ $O(b \log b)$个布尔门 $\Longrightarrow$ 代价：$O(b \log b \cdot \lambda)$

Garble Arithmetic Circuits

混淆 代数电路

支持某个环（e.g. $\mathbb{Z}_{2^b}$）上的代数运算



Baseline Solution:

混淆对应的布尔电路

The cost?

代价是什么?

代数加法门 $\Longrightarrow$ $O(b)$个布尔门 $\Longrightarrow$ 代价：$O(b \cdot \lambda)$

代数乘法门 $\Longrightarrow$ $O(b^2)$个布尔门（$b<500$） $\Longrightarrow$ 代价：$O(b^2 \cdot \lambda)$

- Karatsuba：$O(b^{1.58})$个布尔门（$500<b<10,000$） $\Longrightarrow$ 代价：$O(b^{1.58} \cdot \lambda)$
- FFT：$O(b \log b)$个布尔门（$b>10,000$） $\Longrightarrow$ 代价：$O(b \log b \cdot \lambda)$

混淆代数电路的框架 [AIK12]

How to Garble Arithmetic Circuits*

Benny Applebaum† Yuval Ishai‡ Eyal Kushilevitz§

December 14, 2012

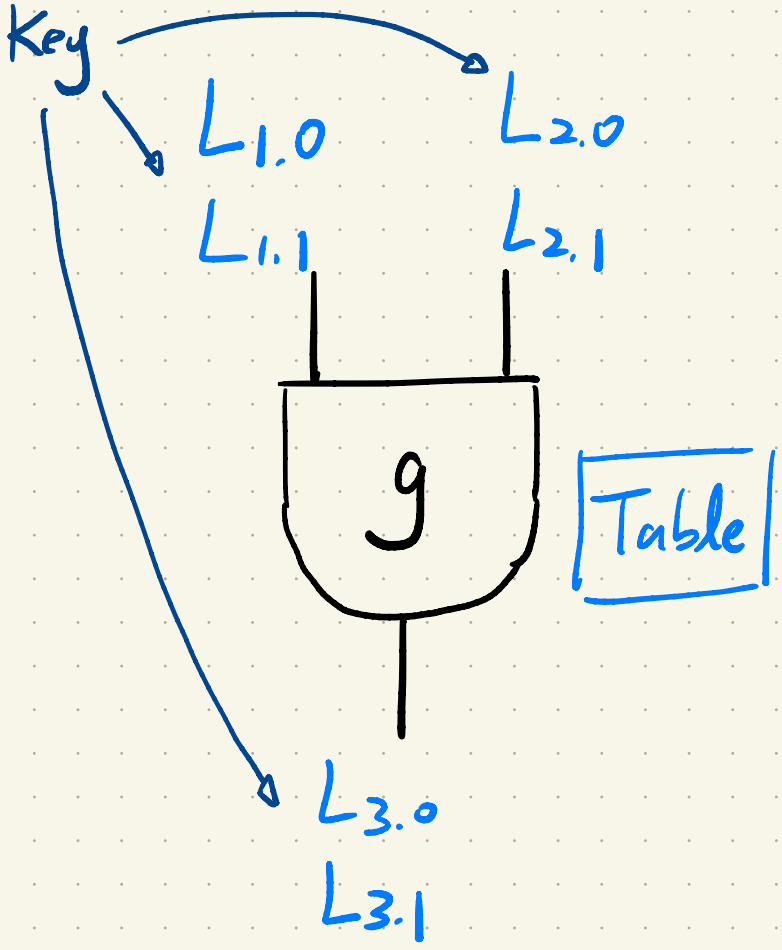
Abstract

Yao's garbled circuit construction transforms a boolean circuit $C : \{0,1\}^n \to \{0,1\}^m$ into a "garbled circuit" $\hat{C}$ along with n pairs of k-bit keys, one for each input bit, such that $\hat{C}$ together with the n keys corresponding to an input x reveal $C(x)$ and no additional information about x. The garbled circuit construction is a central tool for constant-round secure computation and has several other applications.

Motivated by these applications, we suggest an efficient arithmetic variant of Yao's original construction. Our construction transforms an arithmetic circuit $C : \mathbb{Z}^n \to \mathbb{Z}^m$ over integers from a bounded (but possibly exponential) range into a garbled circuit $\hat{C}$ along with n affine functions $L_i : \mathbb{Z} \to \mathbb{Z}^k$ such that $\hat{C}$ together with the n integer vectors $L_i(x_i)$ reveal $C(x)$ and no additional information about x. The security of our construction relies on the intractability of the learning with errors (LWE) problem.

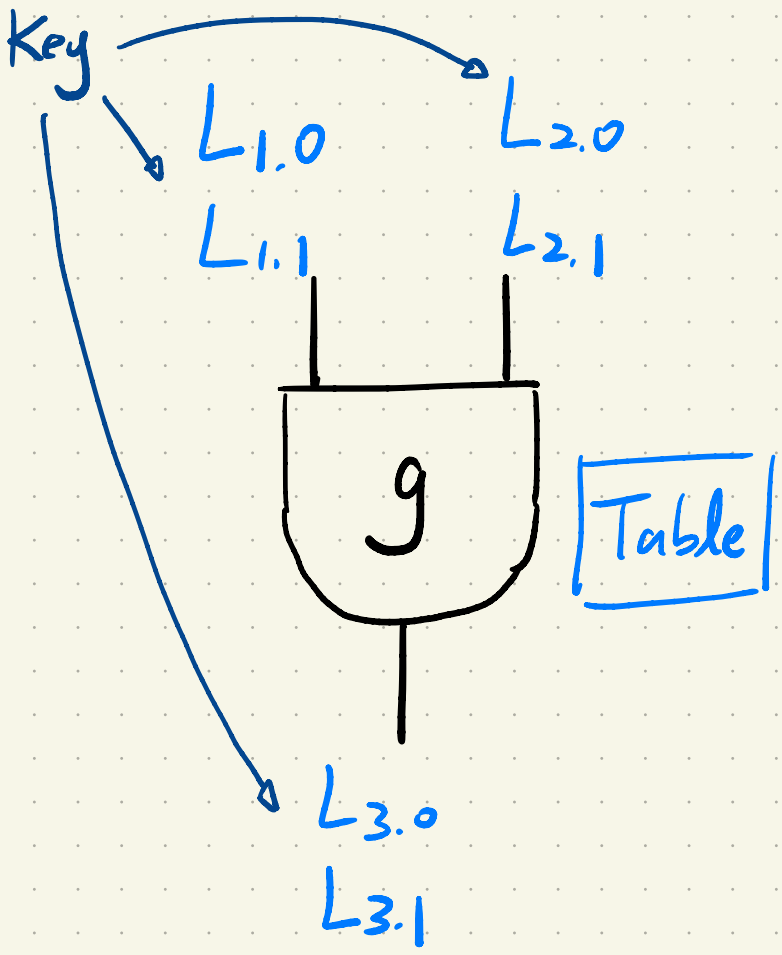
混淆~~代数~~布尔电路的框架

gate-by-gate

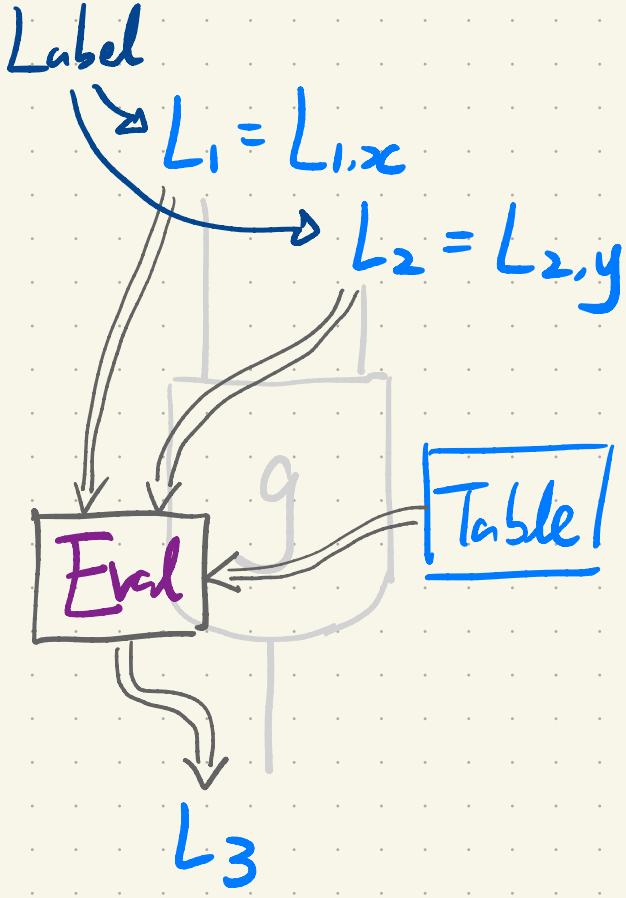


Garbling

混淆~~代数~~布尔电路的框架 gate-by-gate



Garbling



Evaluation

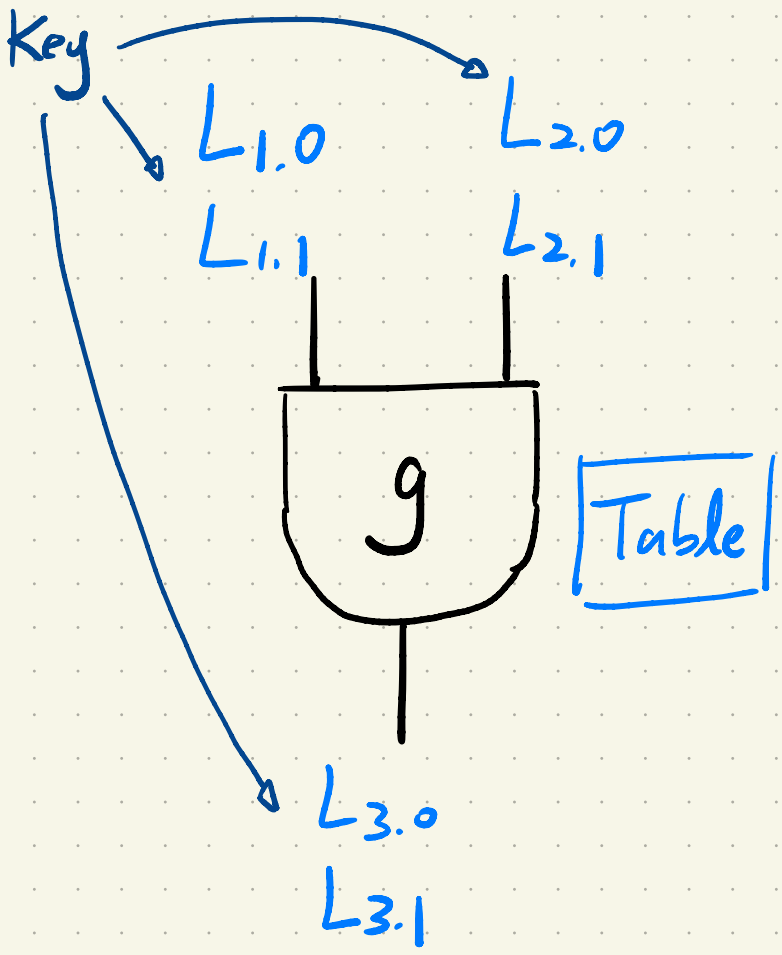
☆正确性：

$L_3 = L_{3,g(x,y)}$

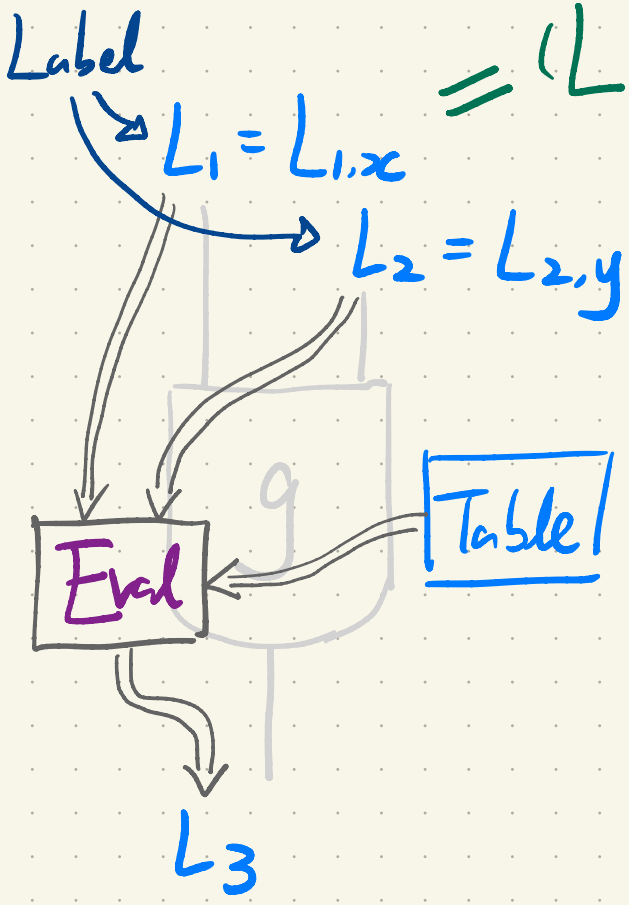
☆安全性：

$(L_1\ L_2\ \text{Table}\ L_3) \approx_c (\text{Sim}(L_3), L_3)$

混淆~~代数~~布尔电路的框架 gate-by-gate



Garbling



Evaluation

$= (L_{1,1} - L_{1,0}) \cdot x + L_{1,0}$

☆正确性：

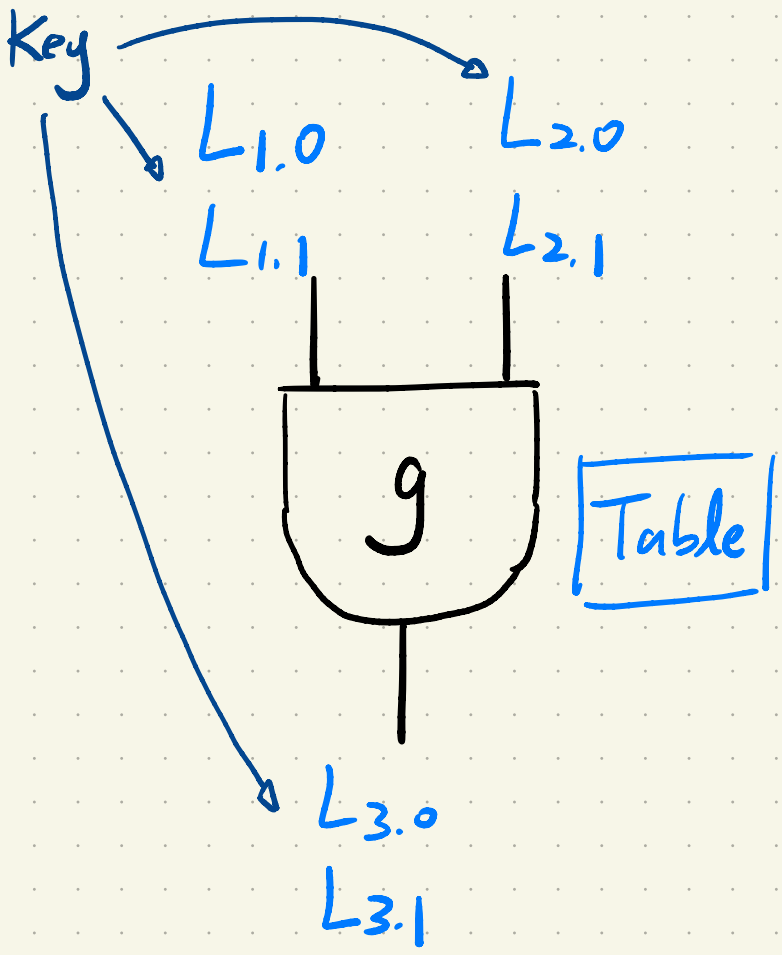
$L_3 = L_{3,g(x,y)}$

☆安全性：

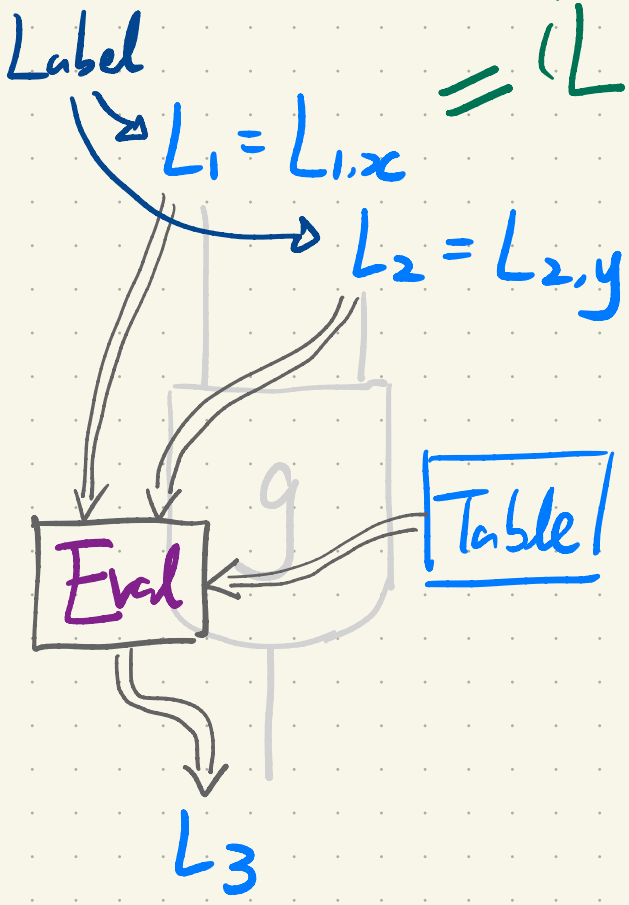
$(L_1\ L_2\ \text{Table}\ L_3) \approx_c (\text{Sim}(L_3), L_3)$

混淆~~代数~~布尔电路的框架

gate-by-gate



Garbling



Evaluation

$$A_1 \qquad B_1$$
$$= \underbrace{(L_{1,1} - L_{1,0})}_{A_1} \cdot x + \underbrace{L_{1,0}}_{B_1}$$

☆正确性：

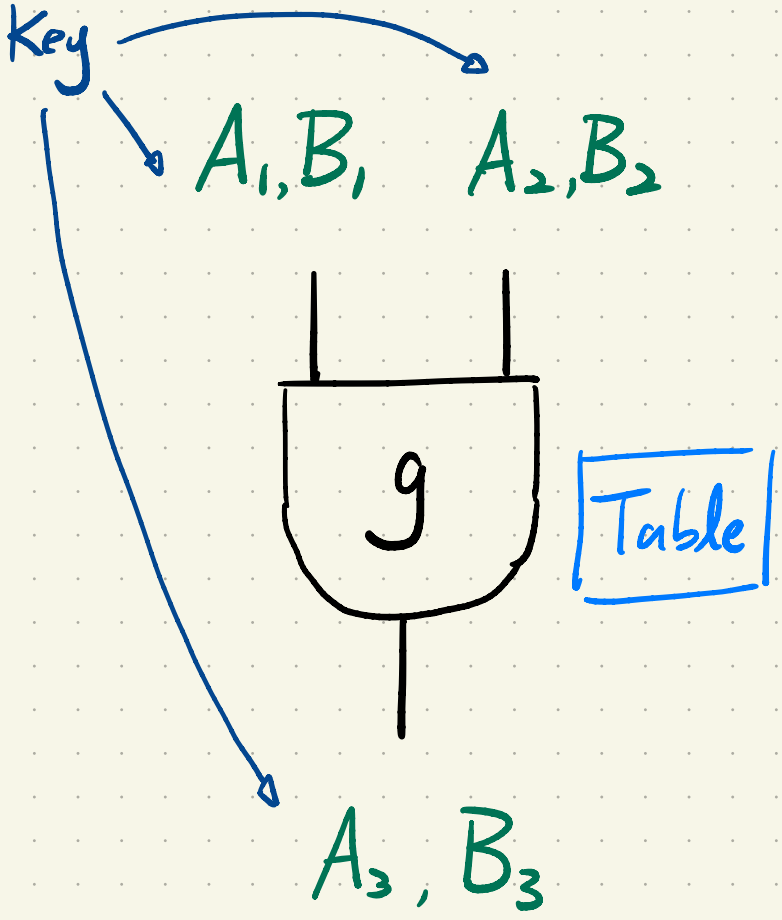
$$L_3 = L_{3,g(x,y)}$$

☆安全性：

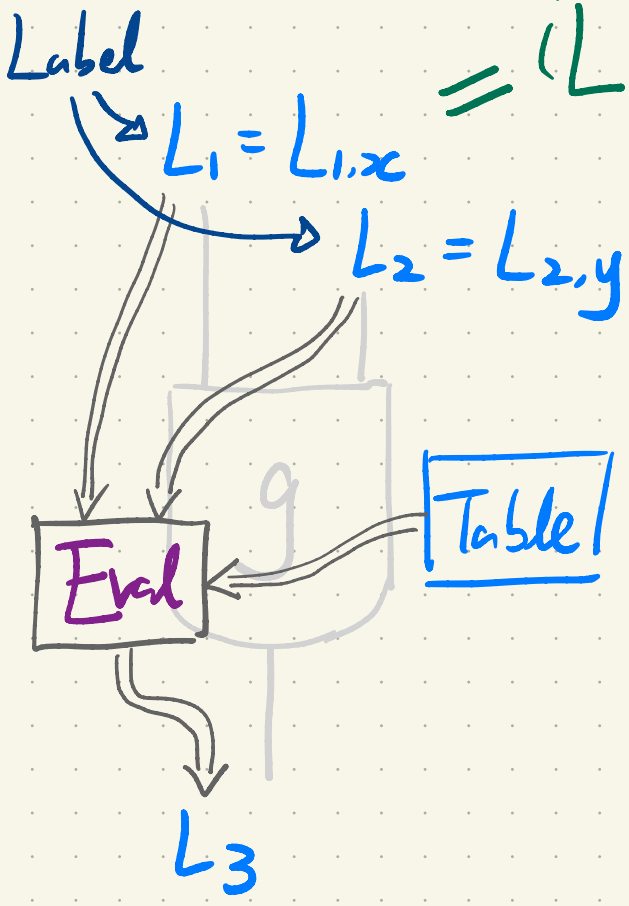
$$(L_1\ L_2\ \text{Table}\ L_3) \overset{c}{\approx} (\text{Sim}(L_3), L_3)$$

混淆~~代数~~布尔电路的框架

gate-by-gate



Garbling



Evaluation

$$\underbrace{A_1}_{(L_{1,1}-L_{1,0})\cdot x} + \underbrace{B_1}_{L_{1,0}}$$

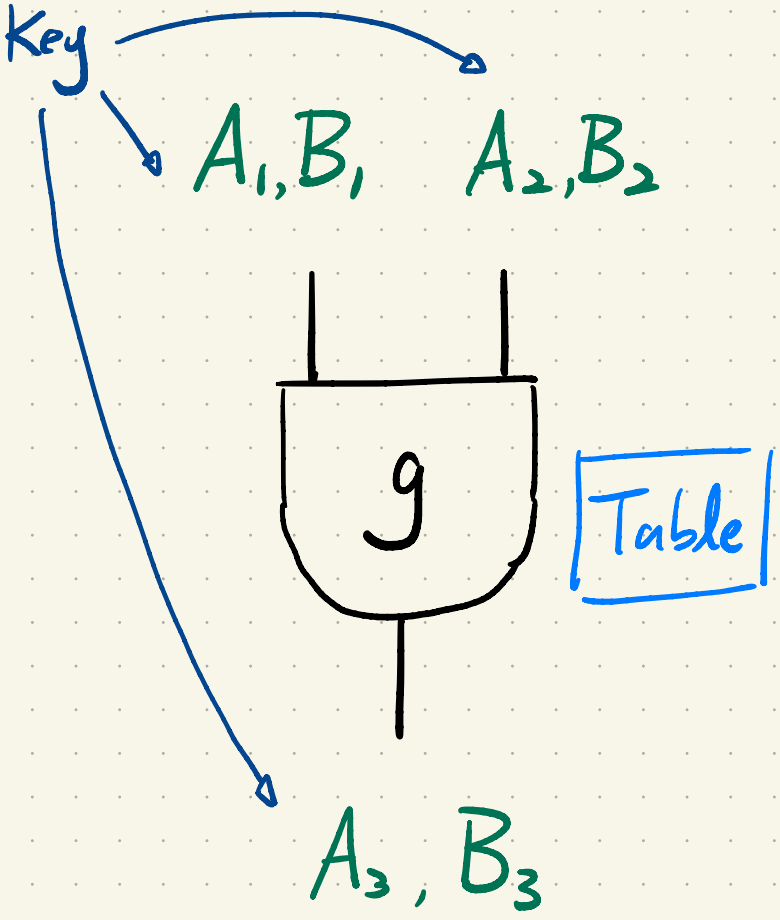
☆正确性：

$$L_3 = L_{3,g(x,y)}$$

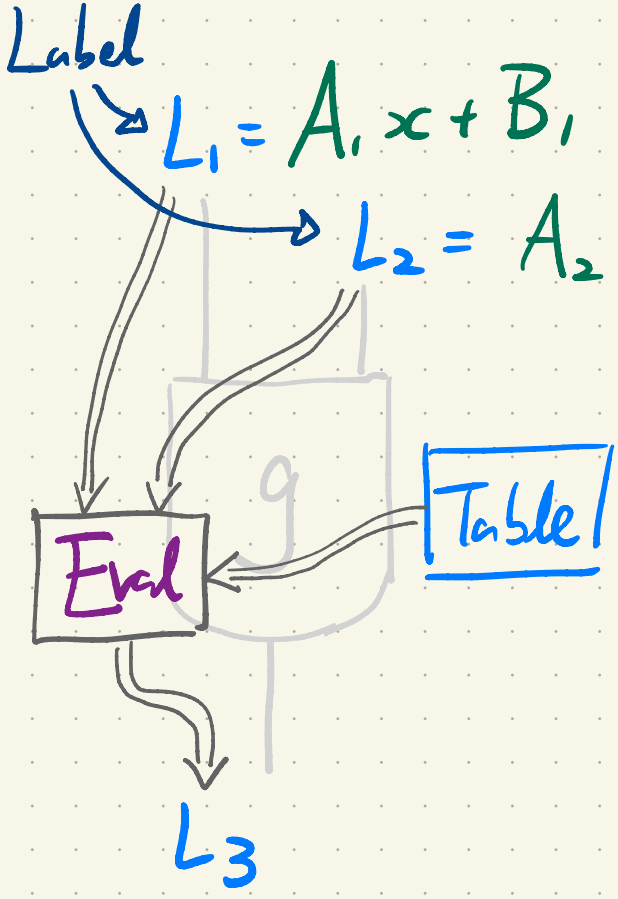
☆安全性：

$$(L_1\ L_2\ \text{Table}\ L_3) \overset{c}{\approx} (\text{Sim}(L_3), L_3)$$

混淆~~代数~~布尔电路的框架 gate-by-gate



Garbling



Evaluation

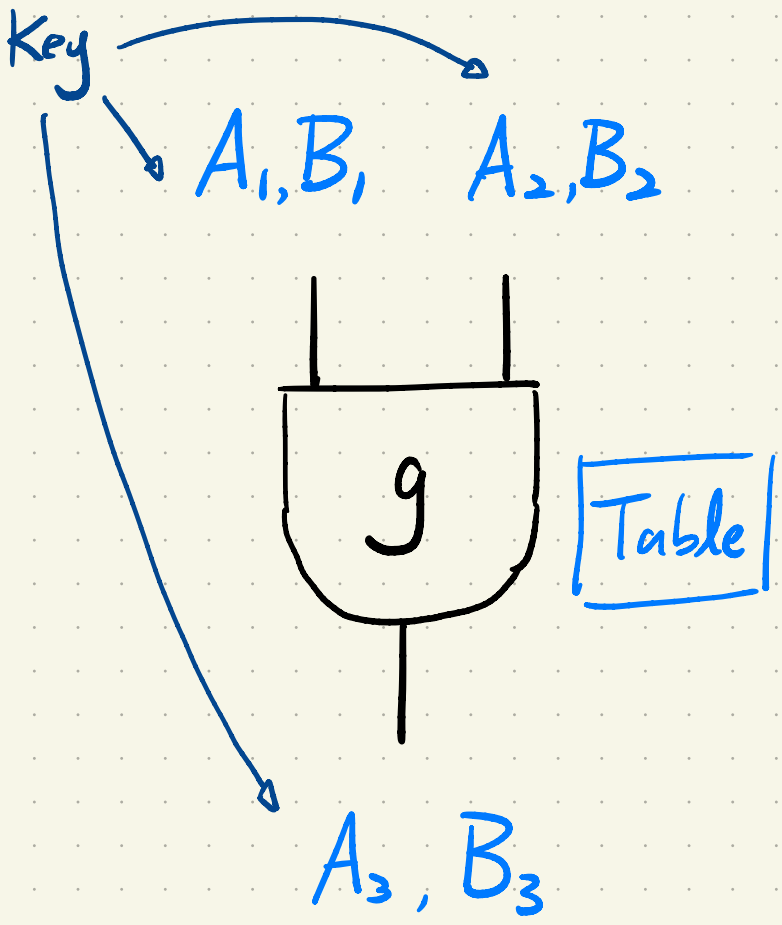
☆正确性：

$$L_3 = A_3 g(x,y) + B_3$$

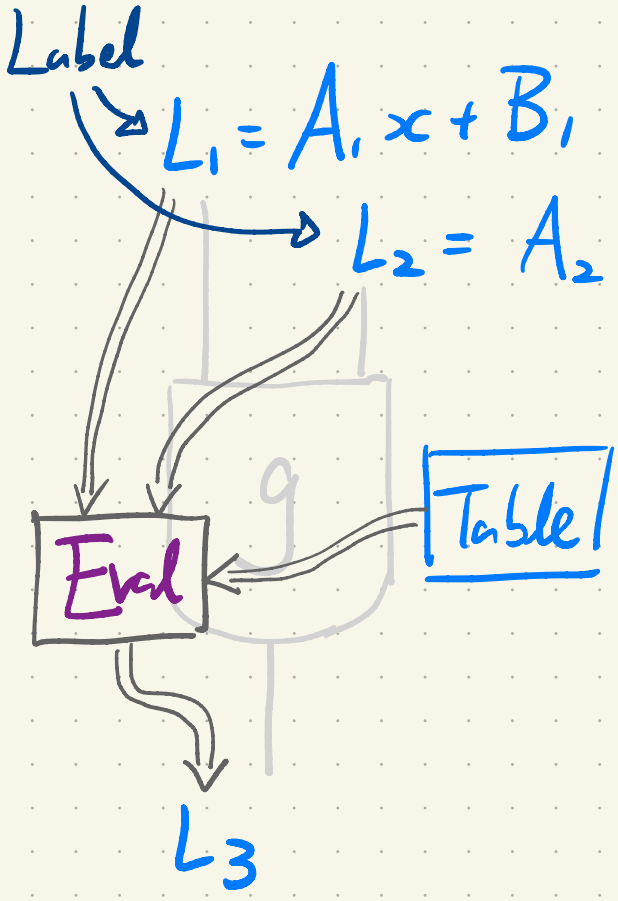
☆安全性：

$$(L_1, L_2, \text{Table}, L_3) \approx_c (\text{Sim}(L_3), L_3)$$

混淆代数电路的框架 [AIK]



Garbling



Evaluation

☆正确性：

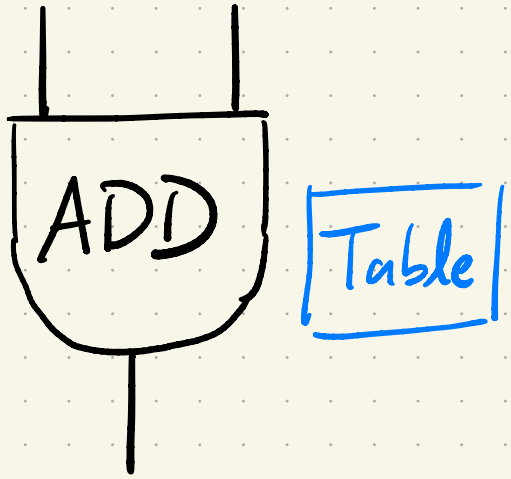
$$L_3 = A_3 g(x,y) + B_3$$

☆安全性：

$$(L_1, L_2, \text{Table}, L_3) \approx_c (\text{Sim}(L_3), L_3)$$

如何混淆代数加法

A_1, B_1 A_2, B_2



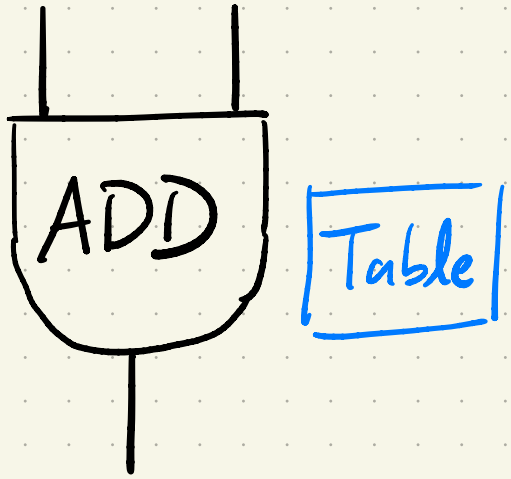
A_3, B_3

Garbling

如何混淆代数加法

- Let $A_1 = A_2 = A_3$
- Let $B_3 = B_1 + B_2$

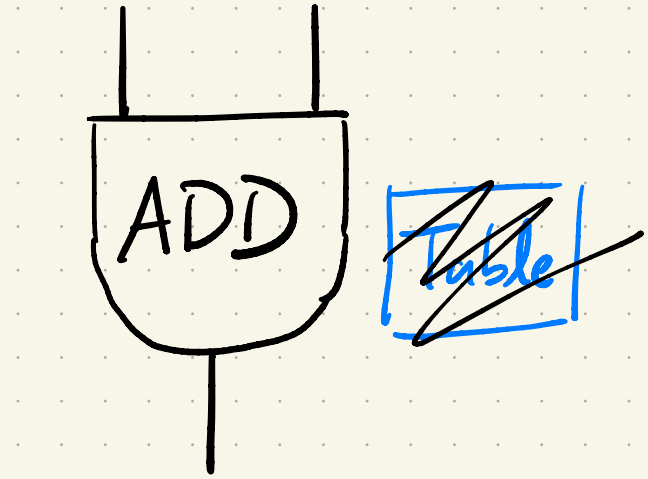
A_1, B_1 A_2, B_2



A_3, B_3

Garbling

如何混淆代数加法

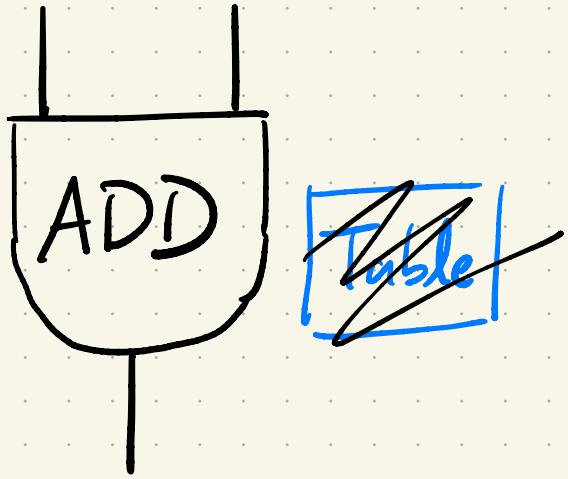


- Let $A_1 = A_2 = A_3$
- Let $B_3 = B_1 + B_2$
- Empty table

Garbling

如何混淆代数加法

A_1, B_1 A_2, B_2



A_3, B_3

Garbling

- Let $A_1 = A_2 = A_3$
- Let $B_3 = B_1 + B_2$
- Empty table
- Eval

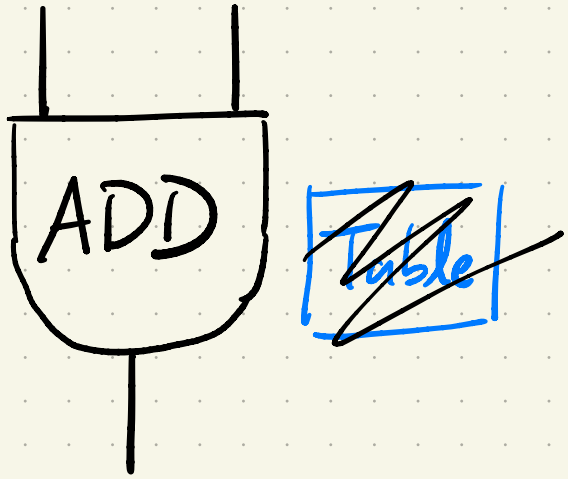
$$L_1 + L_2 = A_1 x + B_1 + A_2 y + B_2$$

$$\| $$

$$L_3 = A_3(x+y) + (B_1 + B_2)$$

如何混淆代数加法

A_1, B_1 A_2, B_2



A_3, B_3

Garbling

- Let $A_1 = A_2 = A_3$
- Let $B_3 = B_1 + B_2$
- Empty table
- Eval

$$L_1 + L_2 = A_1 x + B_1 + A_2 y + B_2$$

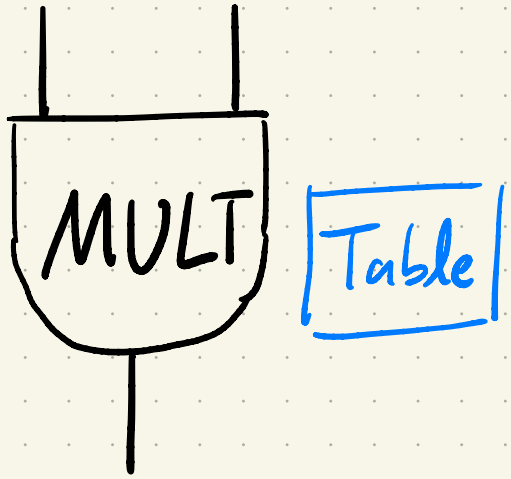
$$\| $$

$$L_3 = A_3(x+y) + (B_1 + B_2)$$

FreeADD (similar to FreeXOR)

如何混淆代数乘法

A_1, B_1 A_2, B_2

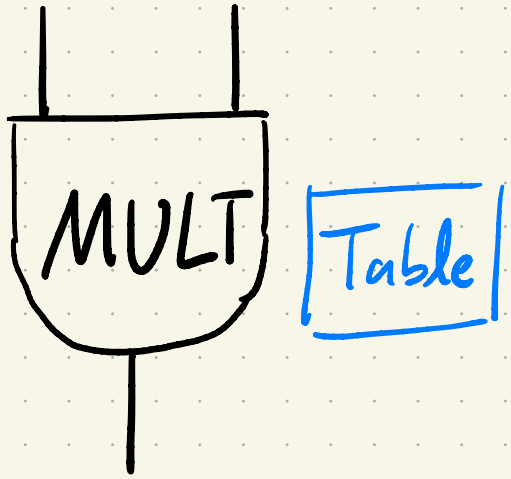


A_3, B_3

Garbling

如何混淆代数乘法

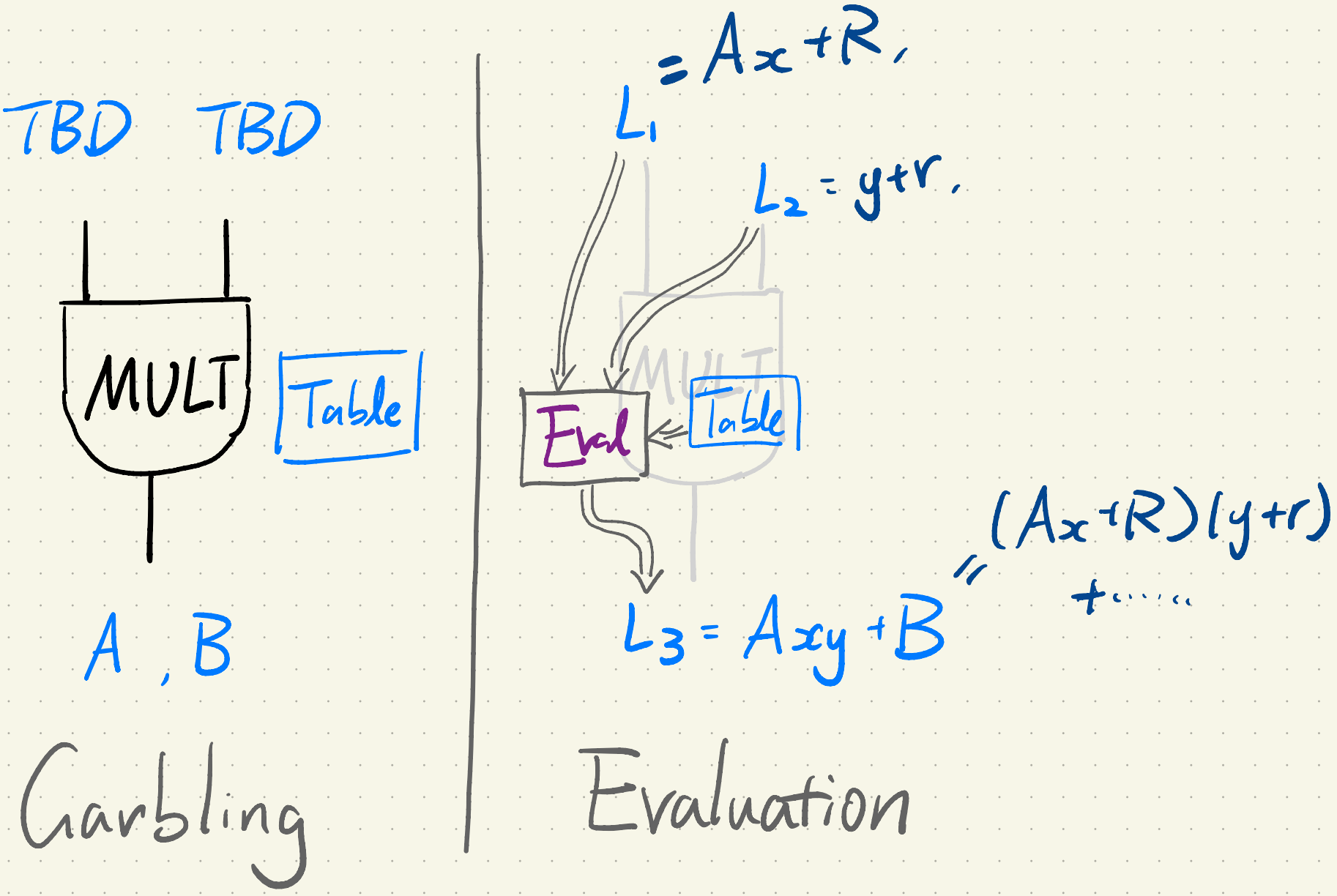
TBD TBD



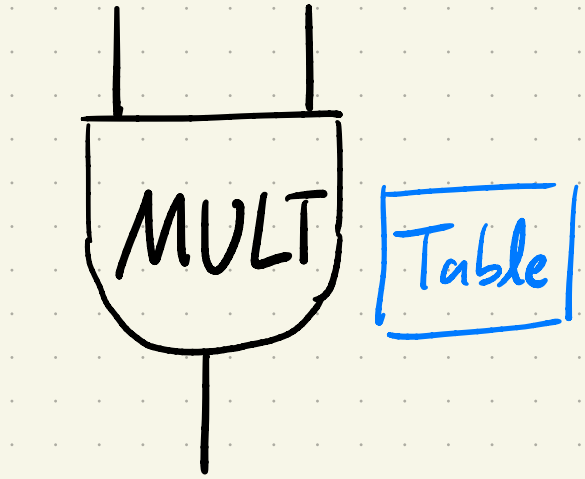
A , B

Garbling

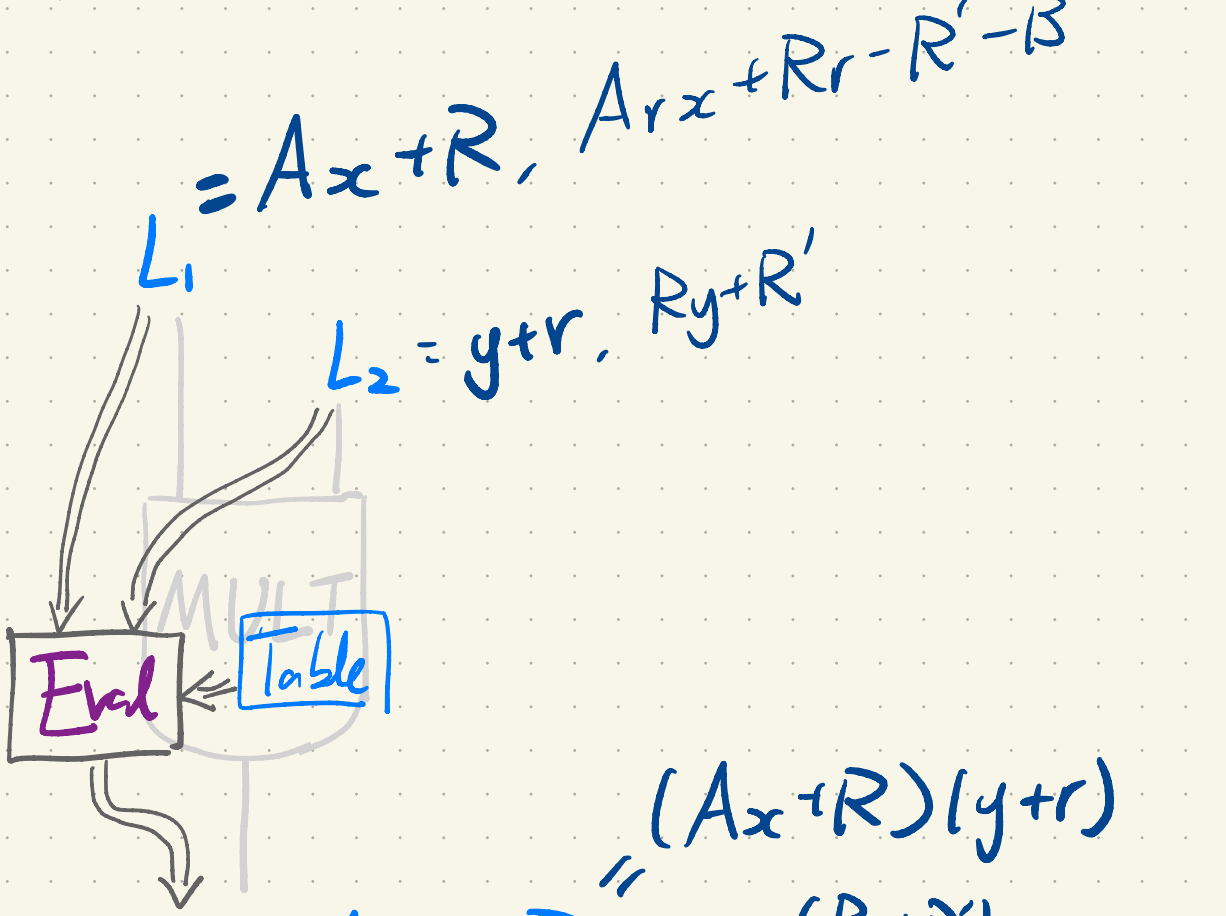
如何混淆代数乘法


TBD TBD
MULT
Table
A , B
Garbling
$L_1 = Ax + R$,
$L_2 = y + r$,
Eval
Table
$L_3 = Axy + B$
$= (Ax + R)(y + r)$
$+ \cdots\cdots$
Evaluation

如何混淆代数乘法

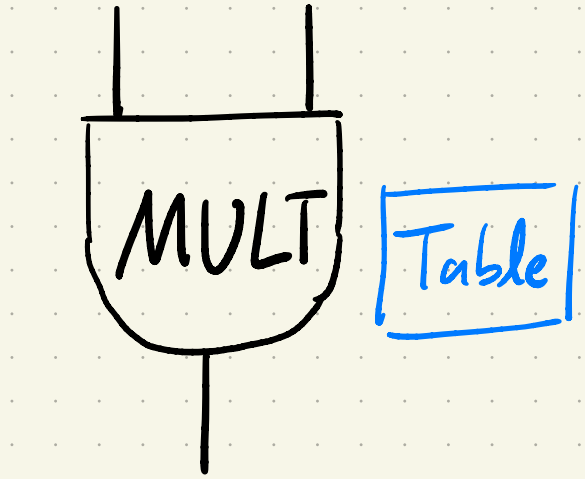


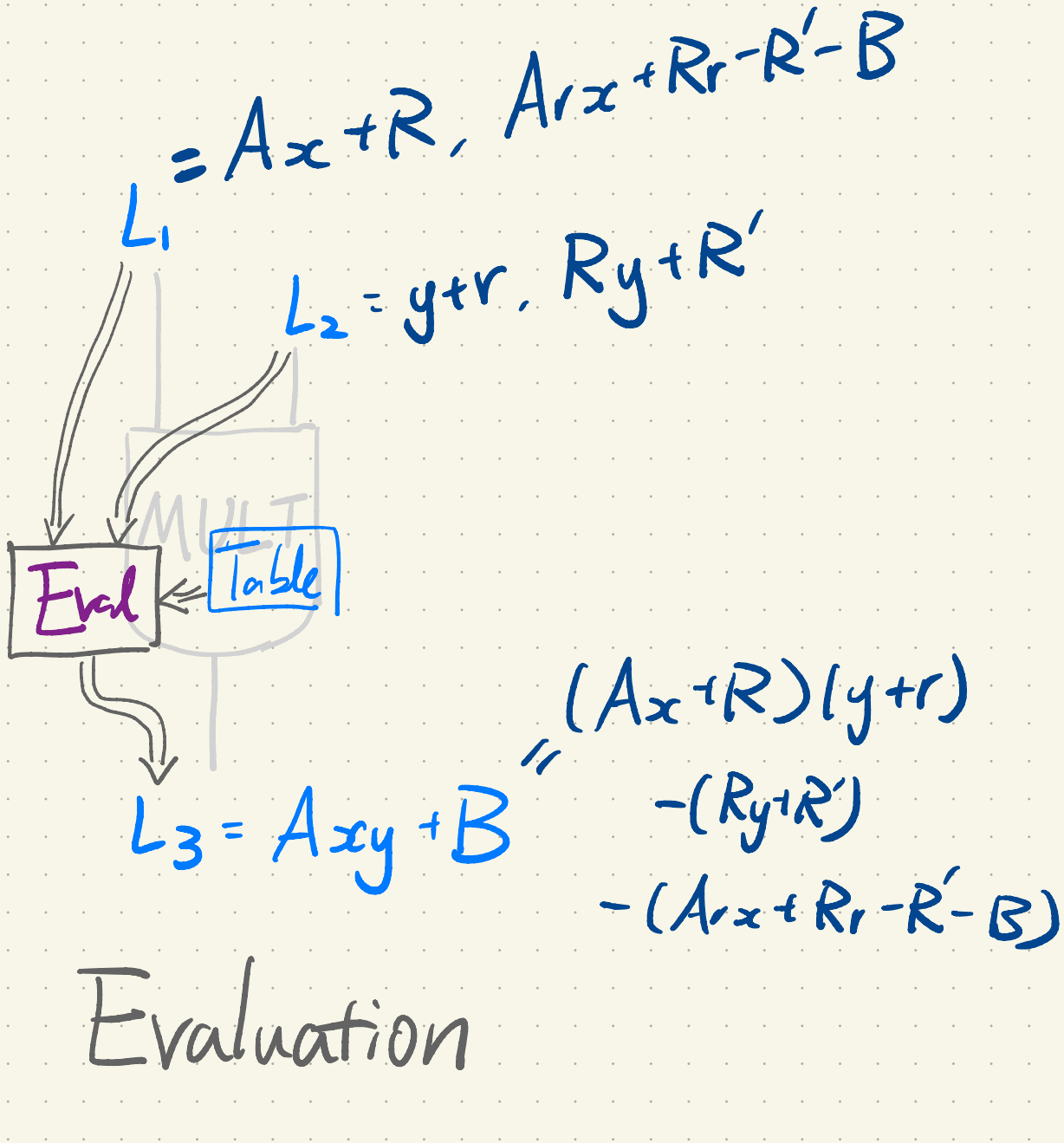
Garbling



Evaluation

如何混淆代数乘法

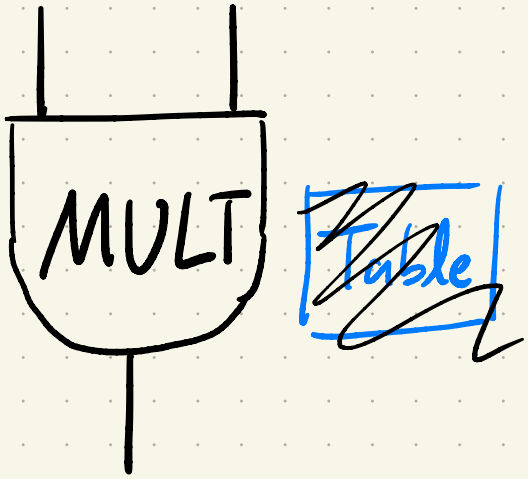

TBD
TBD
MULT
Table
A , B
Garbling


L1 = Ax+R, Arx+Rr-R'-B
L2 = y+r, Ry+R'
MULT
Eval
Table
L3 = Axy+B
(Ax+R)(y+r)
-(Ry+R')
-(Arx+Rr-R'-B)
Evaluation

如何混淆代数乘法

$A_1 = (A, Ar)$ $A_2 = (1, R)$

$B_1 = (R, Rr - R' - B)$ $B_2 = (r, R')$



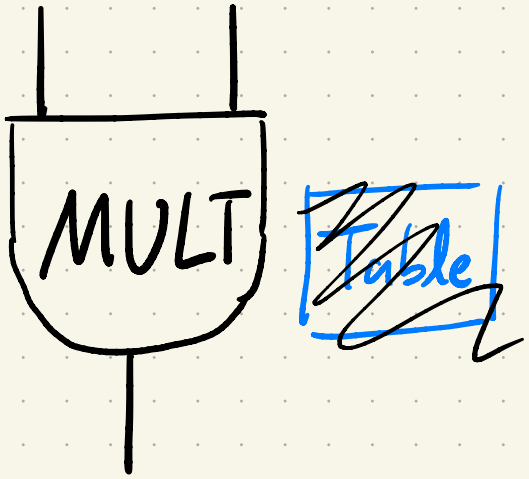
A , B

Garbling

如何混淆代数乘法

$A_1 = (A, Ar)$ $A_2 = (1, R)$

$B_1 = (R, Rr - R' - B)$ $B_2 = (r, R')$



A , B

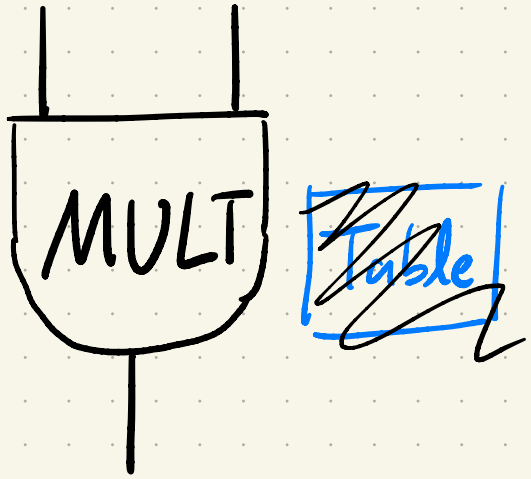
Garbling

FreeMULT?

如何混淆代数乘法

$A_1 = (A, Ar)$ $A_2 = (1, R)$

$B_1 = (R, Rr - R' - B)$ $B_2 = (r, R')$



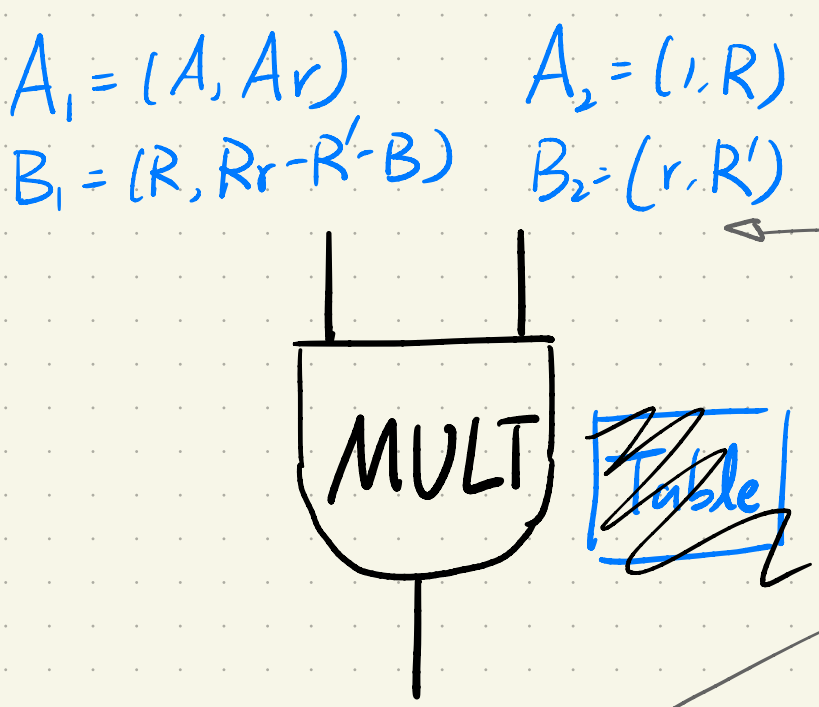
A , B

Garbling

FreeMULT?

NO! Label长度不一致

如何混淆代数乘法



A , B

Garbling

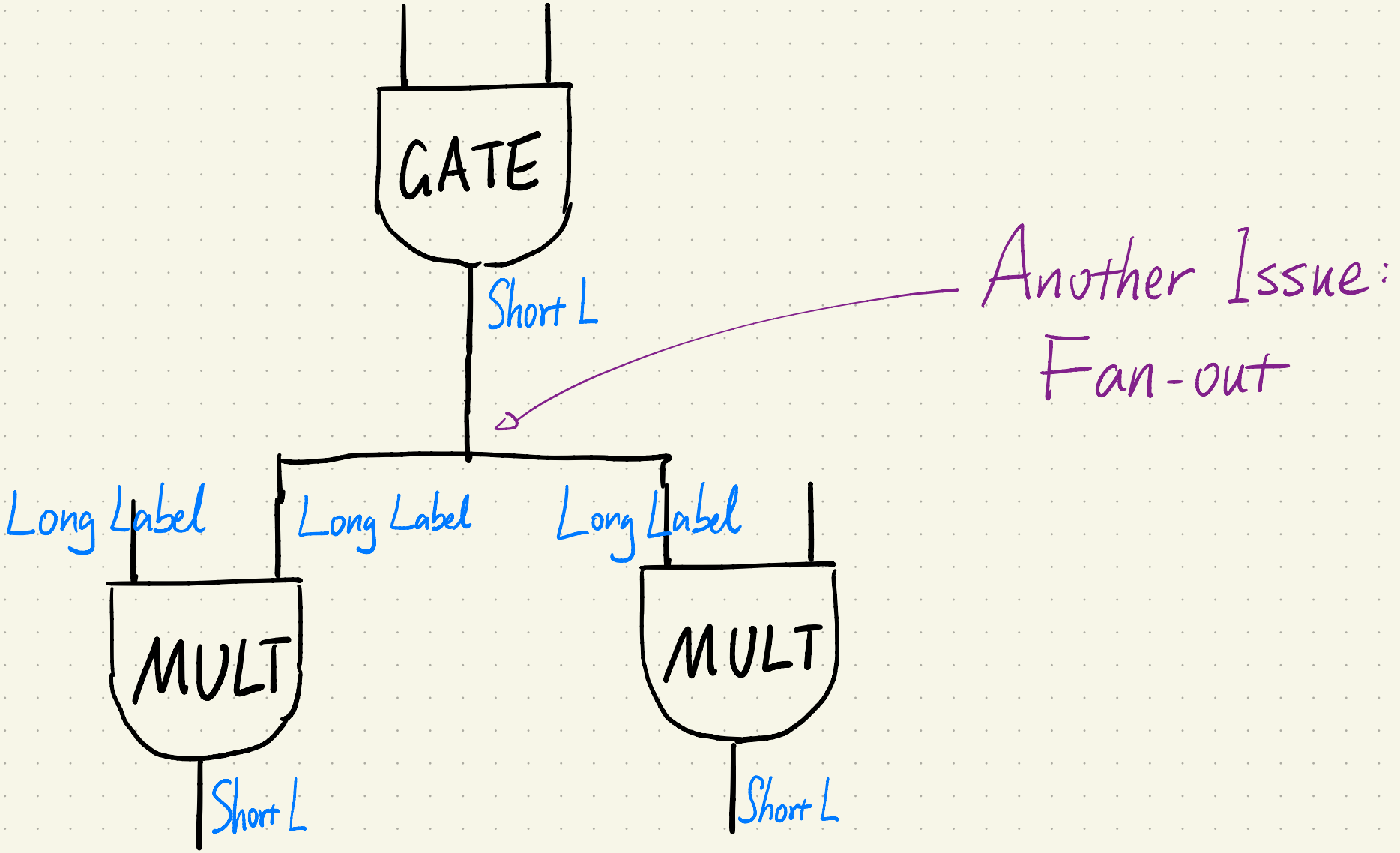
FreeMULT?

long

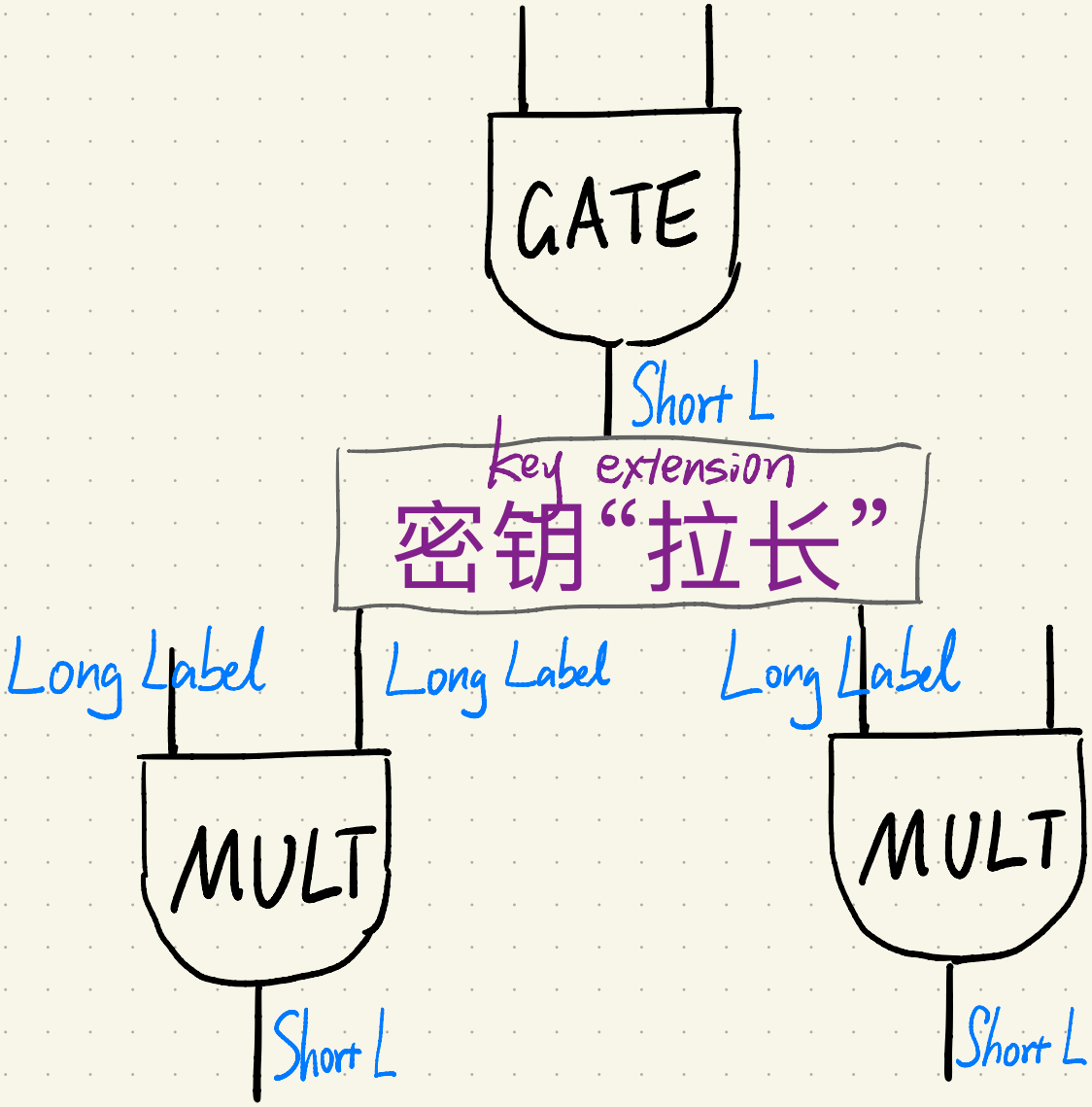
NO! Label长度不一致

short

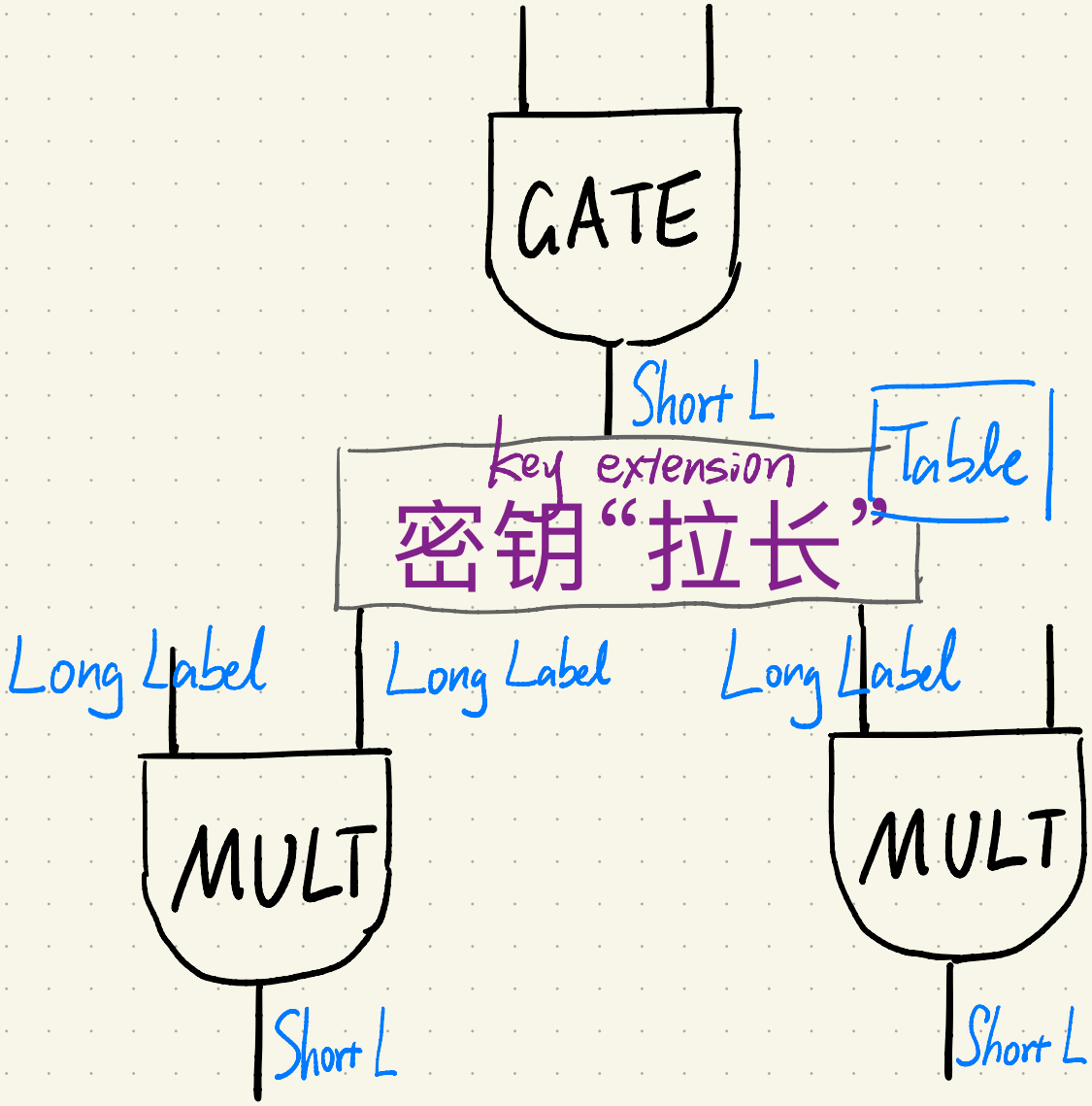
如何混淆代数乘法


GATE
Short L
Another Issue:
Fan-out
Long Label
Long Label
Long Label
MULT
MULT
Short L
Short L

如何混淆代数乘法


GATE
Short L
key extension
密钥“拉长”
Long Label
Long Label
Long Label
MULT
MULT
Short L
Short L

如何混淆代数乘法

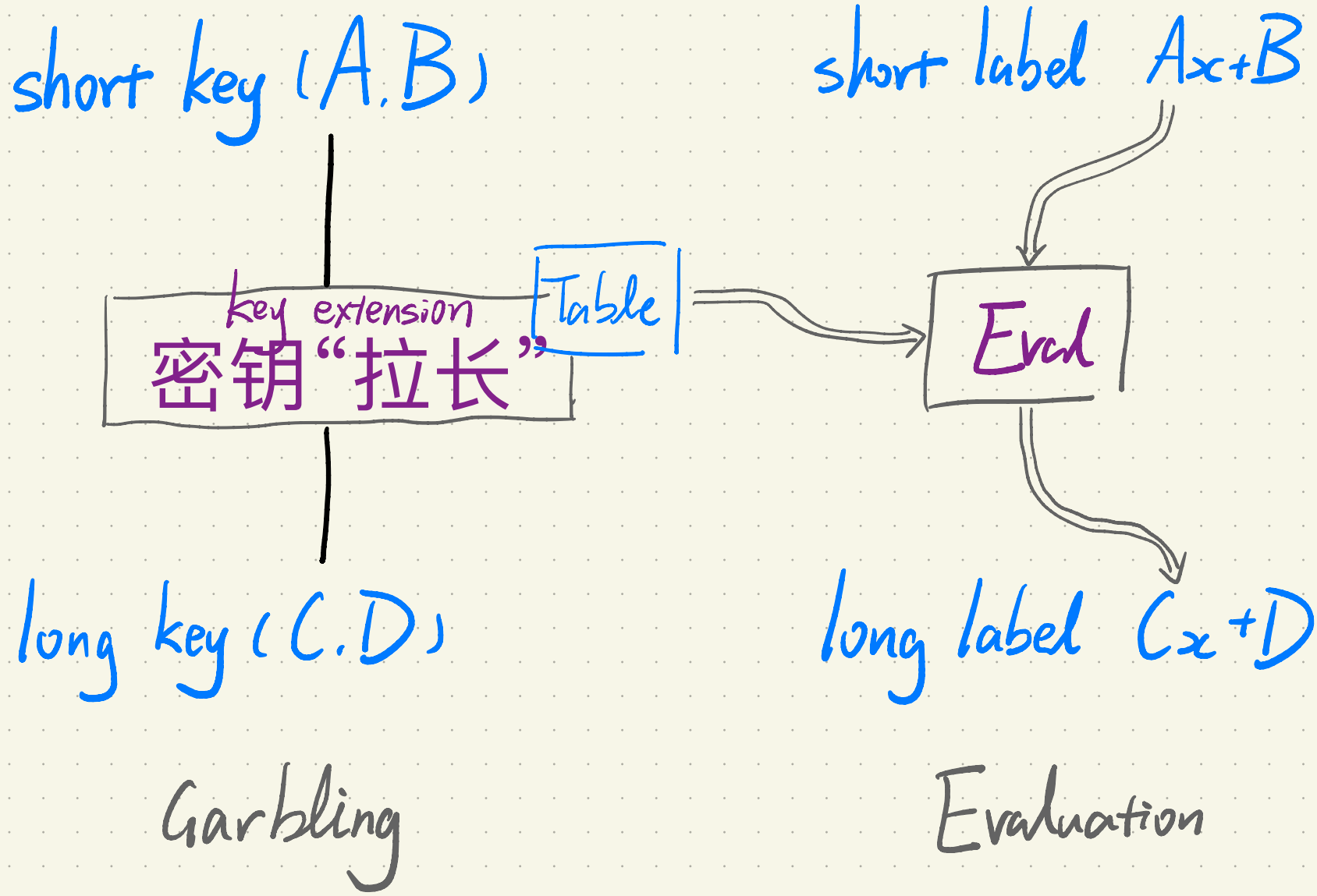


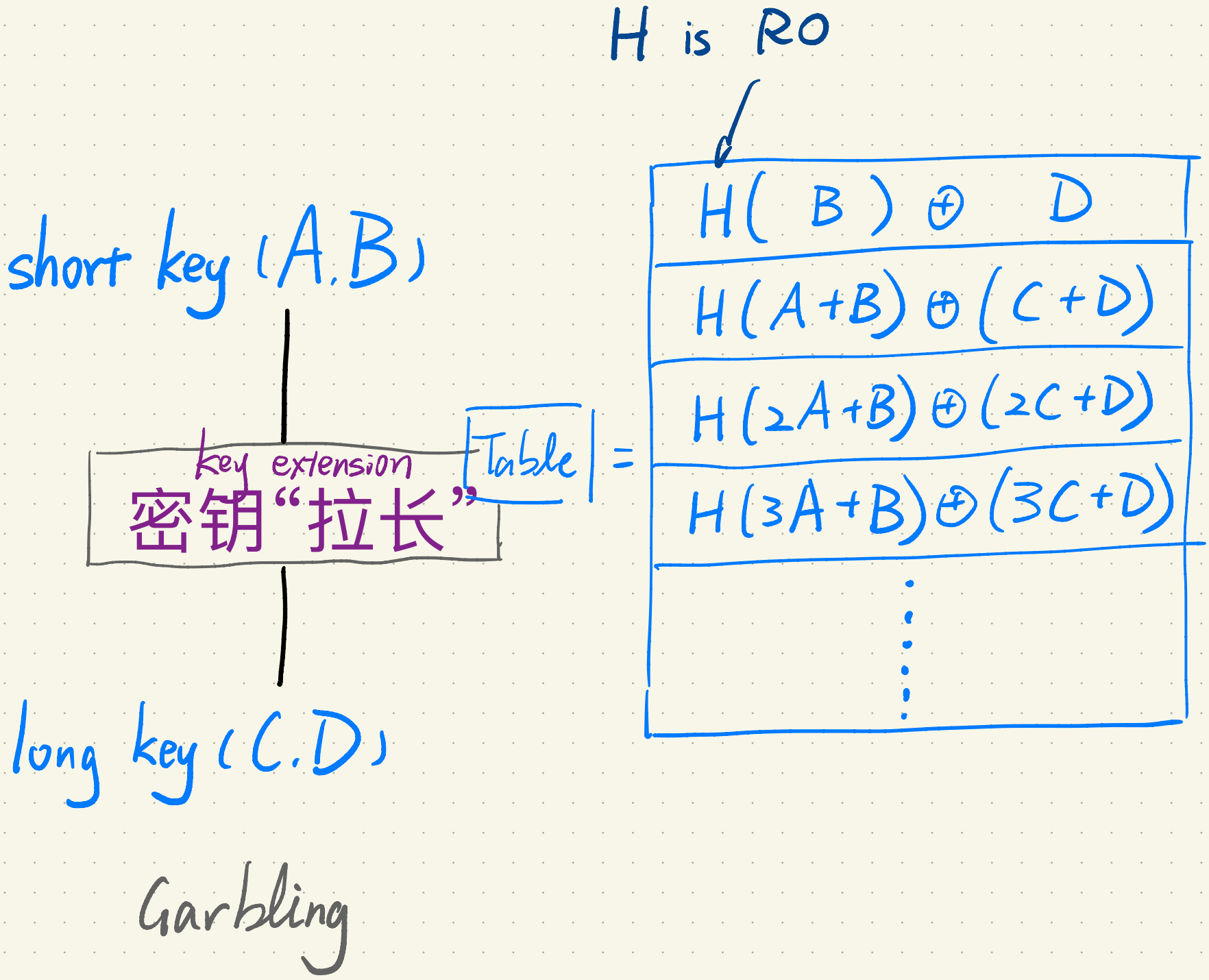
☆ 代数加法门

代价：無

☆ 代数乘法门

代价：密钥“拉长”两倍的 table 大小


short key (A,B)
short label Ax+B
key extension
密钥"拉长"
Table
Eval
long key (C,D)
long label Cx+D
Garbling
Evaluation


H is RO
short key (A.B)
key extension
密钥“拉长”
Table =
H(B) ⊕ D
H(A+B) ⊕ (C+D)
H(2A+B) ⊕ (2C+D)
H(3A+B) ⊕ (3C+D)
long key (C.D)
Garbling

Arithmetic mod P

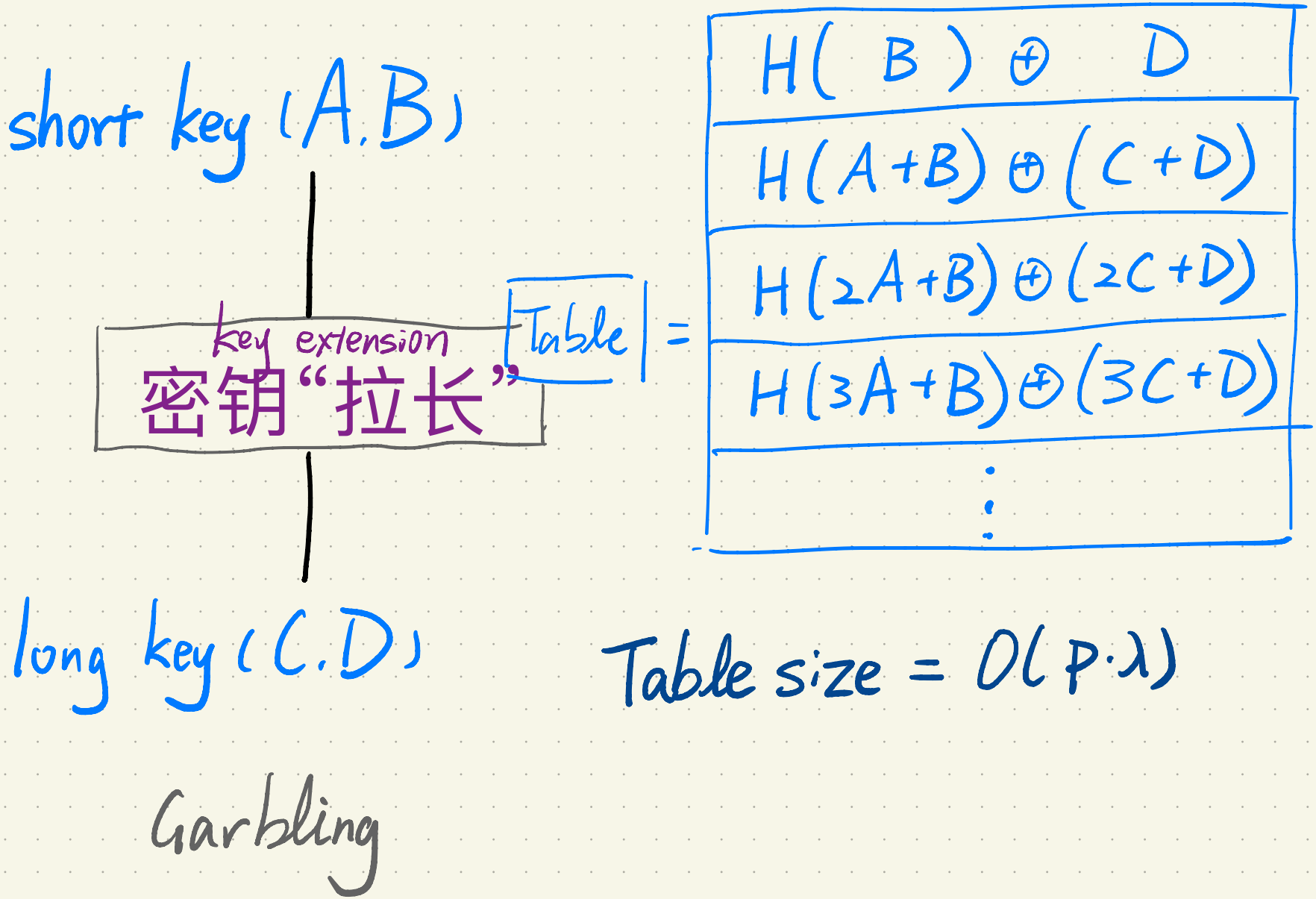


Table size = $O(p \cdot \lambda)$

Arithmetic mod $P_1P_2P_3\cdots P_k$

short key (A, B)

key extension

密钥“拉长”

Table | =

$H(B) \oplus D$
$H(A+B) \oplus (C+D)$
$H(2A+B) \oplus (2C+D)$
$H(3A+B) \oplus (3C+D)$
⋮

long key (C, D)

Table size $= O(p \cdot \lambda)$

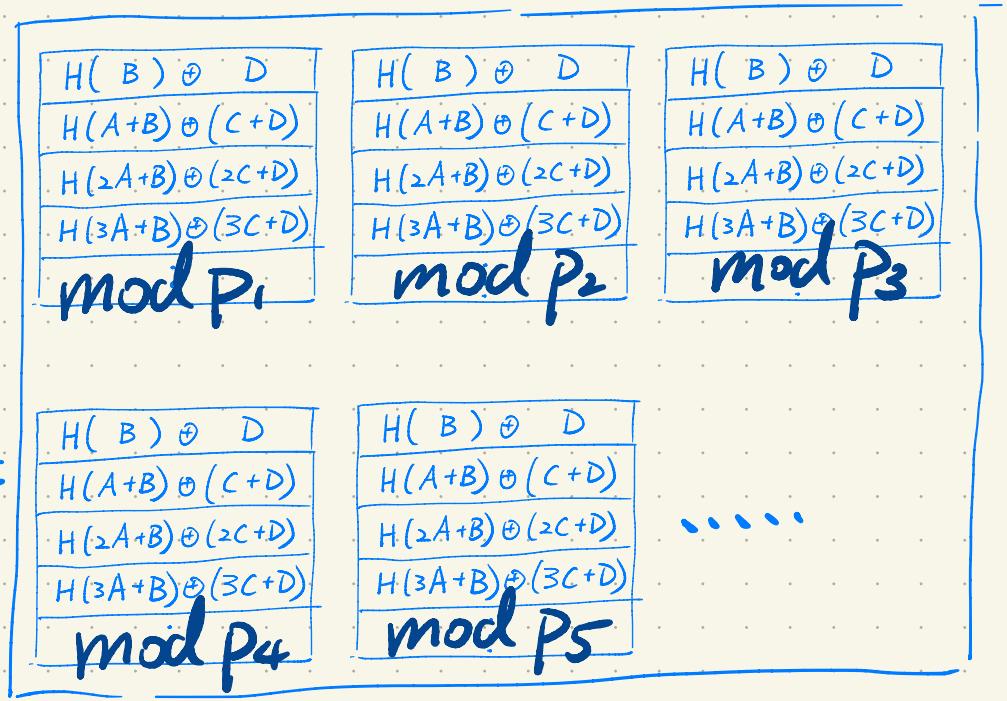
Garbling

Arithmetic mod $p_1 p_2 p_3 \cdots p_k$

short key (A, B)

key extension
密钥"拉长"

Table =



long key (C, D)

Garbling

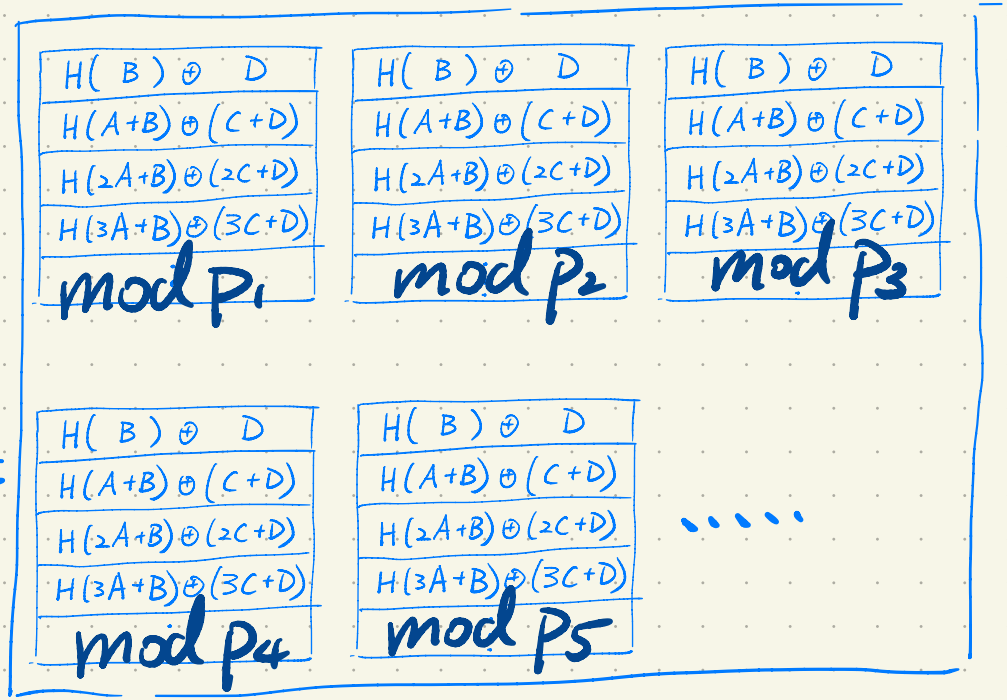
Table size $= O(p \cdot \lambda)$

Arithmetic mod $P_1P_2P_3\cdots P_k$

short key (A, B)

key extension
密钥“拉长”

Table =



long key (C, D)

Garbling

Table size = $O(P_1\lambda + P_2\lambda + \cdots + P_k\lambda)$

Arithmetic mod $\Pi p_i \approx 2^b$

short key (A, B)

key extension
密钥"拉长"

Table =

$H(B) \oplus D$
$H(A+B) \oplus (C+D)$
$H(2A+B) \oplus (2C+D)$
$H(3A+B) \oplus (3C+D)$
mod p_1

$H(B) \oplus D$
$H(A+B) \oplus (C+D)$
$H(2A+B) \oplus (2C+D)$
$H(3A+B) \oplus (3C+D)$
mod p_2

$H(B) \oplus D$
$H(A+B) \oplus (C+D)$
$H(2A+B) \oplus (2C+D)$
$H(3A+B) \oplus (3C+D)$
mod p_3

$H(B) \oplus D$
$H(A+B) \oplus (C+D)$
$H(2A+B) \oplus (2C+D)$
$H(3A+B) \oplus (3C+D)$
mod p_4

$H(B) \oplus D$
$H(A+B) \oplus (C+D)$
$H(2A+B) \oplus (2C+D)$
$H(3A+B) \oplus (3C+D)$
mod p_5

.....

long key (C, D)

Garbling

Table size $= O(\Sigma p_i \lambda)$

$= O\left(\frac{b^2}{\log b} \cdot \lambda\right)$

[BMR16]

Garbling Gadgets for Boolean and Arithmetic Circuits

Marshall Ball[1], Tal Malkin[1] and Mike Rosulek[2]

Arithmetic mod $\Pi p_i \approx 2^b$

short key (A, B)

key extension

密钥“拉长”

long key (C, D)

Garbling

Table =

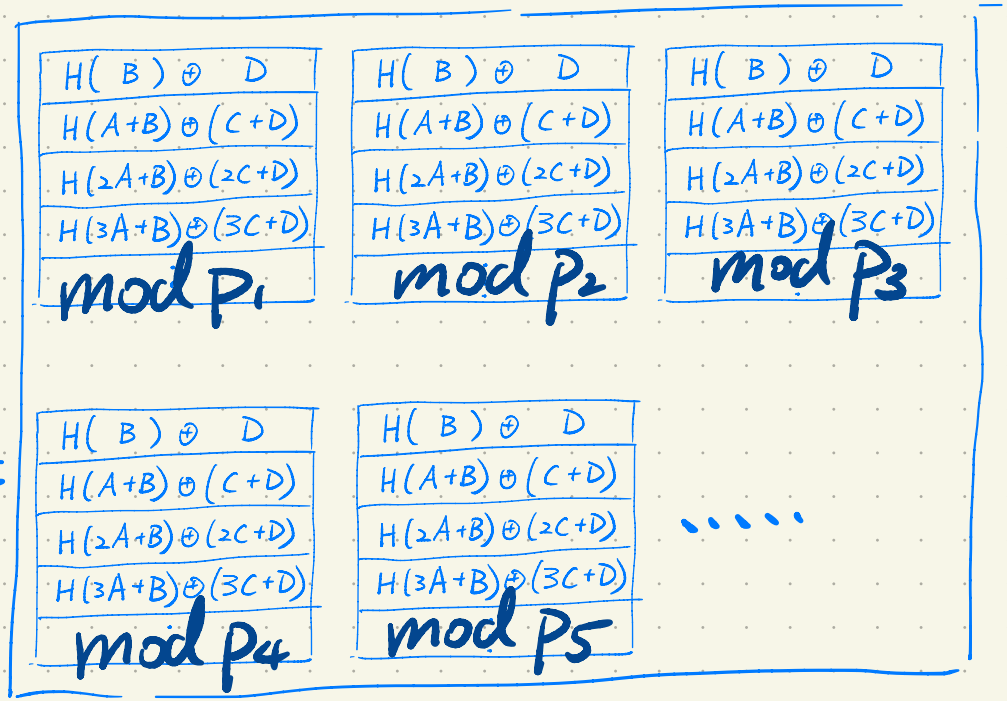
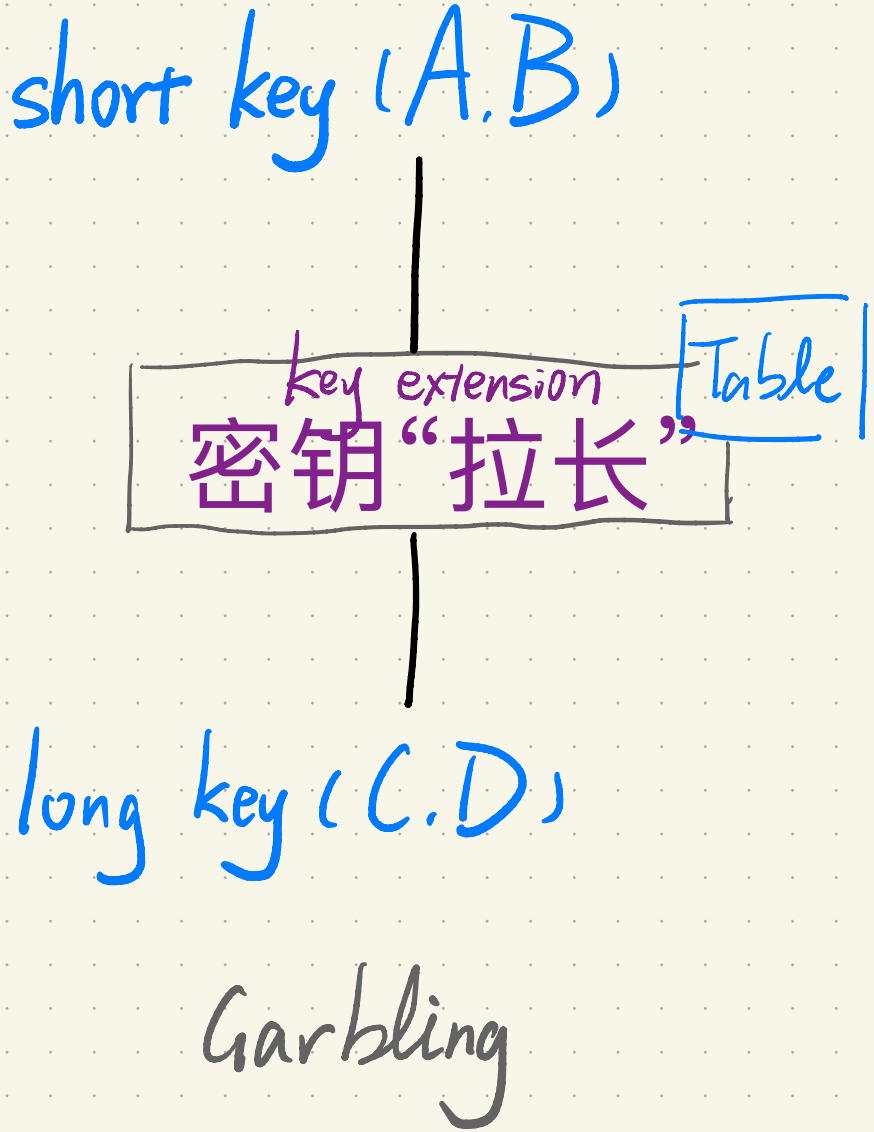


Table size $= O(\Sigma p_i \lambda)$

$= O\left(\frac{b^2}{\log b} \cdot \lambda\right)$

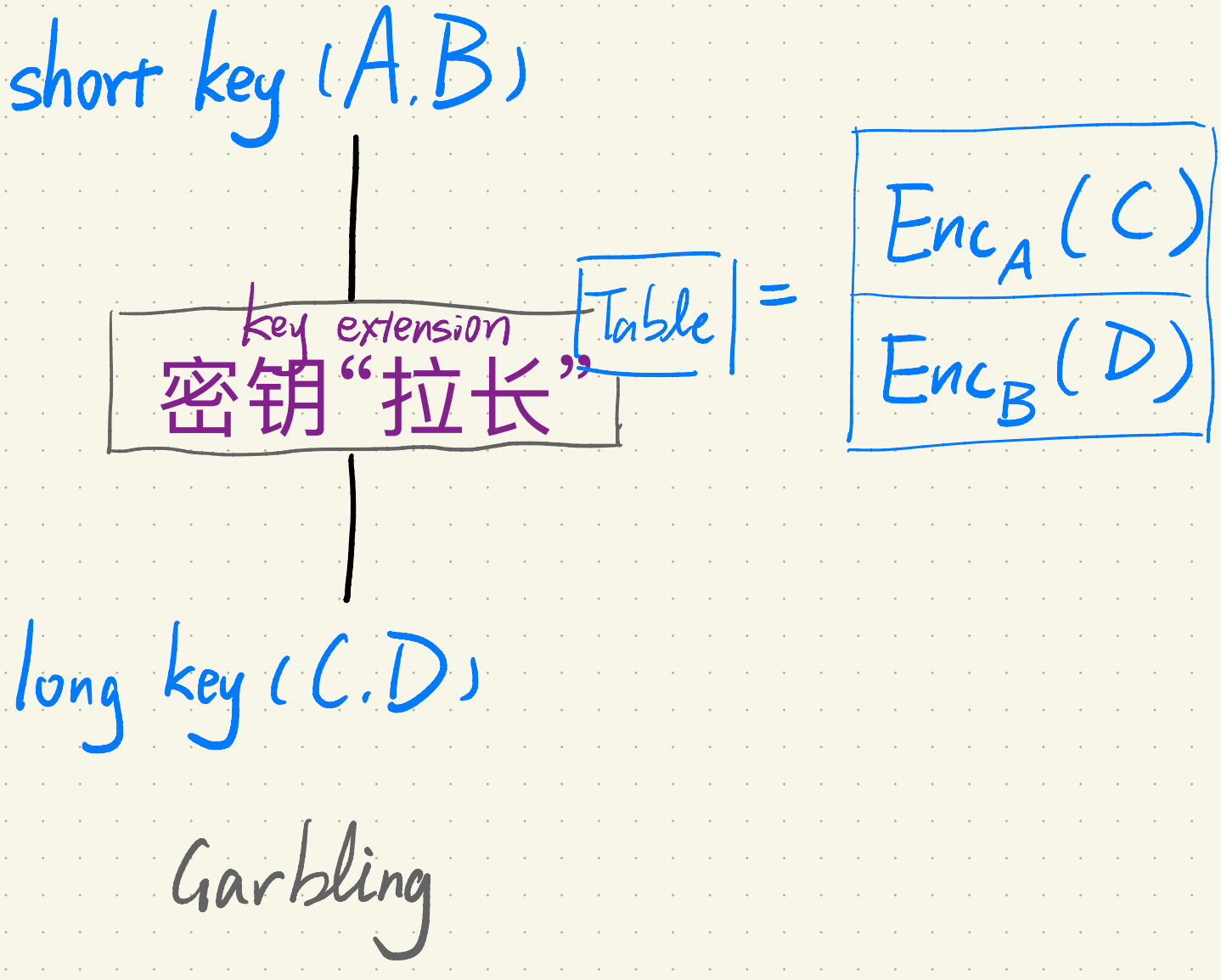
Next Idea.

Homomorphic Encryption



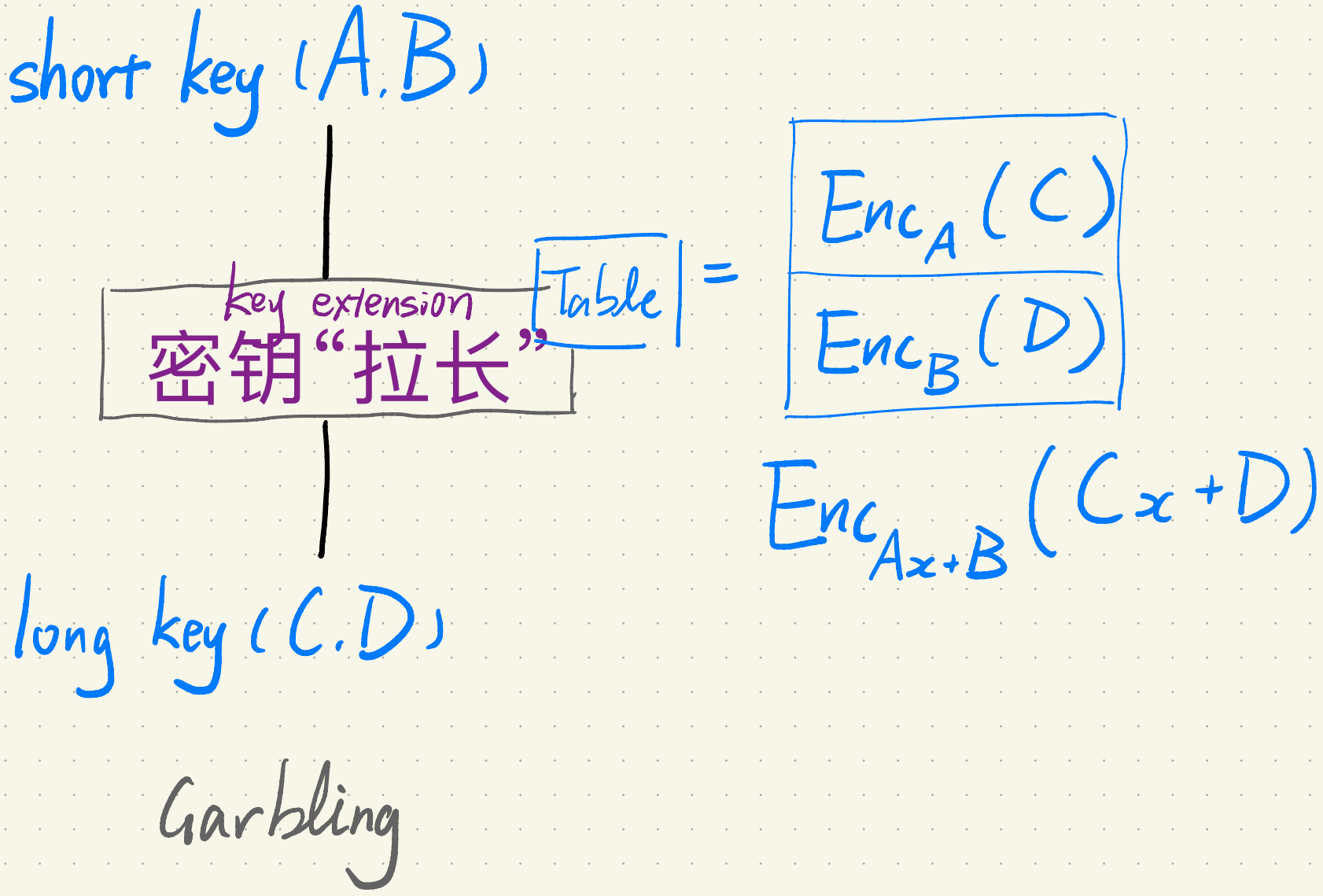
Next Idea.

Homomorphic Encryption



Next Idea.

Homomorphic Encryption

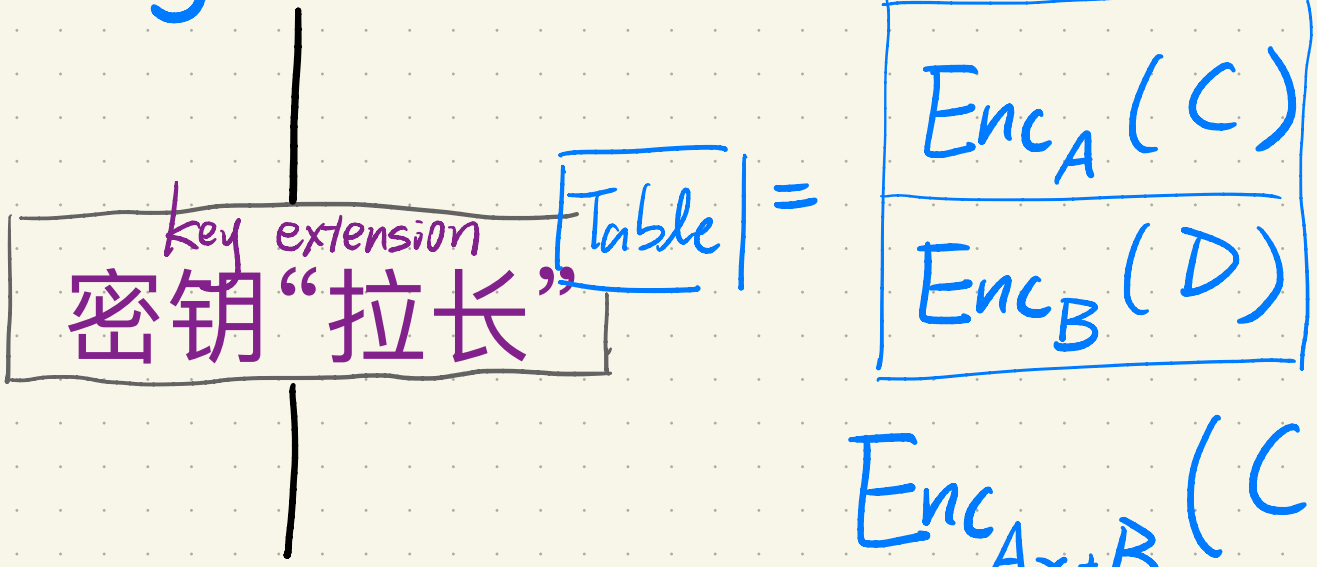


Garbling

Next Idea.

Homomorphic Encryption

short key (A, B)



$Enc_{Ax+B}(Cx+D)$

long key (C, D)

Garbling

Paillier: Table size $= O(b+\lambda)$

[BLLL'23]

New Ways to Garble Arithmetic Circuits

Marshall Ball[1] Hanjun Li[2] Huijia Lin[2] Tianren Liu[3]

Next Idea.

Homomorphic Encryption

short key (A, B)

key extension

密钥"拉长"

Table $=$ $\begin{array}{|c|}\hline \mathrm{Enc}_A(C) \\ \hline \mathrm{Enc}_B(D) \\ \hline \end{array}$

$\mathrm{Enc}_{Ax+B}(Cx+D)$

long key (C, D)

Garbling

Paillier: Table size $= O(b+\lambda)$

[BLLL'23]

New Ways to Garble Arithmetic Circuits

Marshall Ball[1] Hanjun Li[2] Huijia Lin[2] Tianren Liu[3]

Next Idea.

Homomorphic Encryption

short key (A, B)

key extension

密钥"拉长"

long key (C, D)

Table $=$

$Enc_A(C)$
$Enc_B(D)$

不同的环!

$Enc_{Ax+B}(Cx+D)$

Garbling

Paillier: Table size $= O(b+\lambda)$

[BLLL'23]

New Ways to Garble Arithmetic Circuits

Marshall Ball[1] Hanjun Li[2] Huijia Lin[2] Tianren Liu[3]

Next Idea.

Homomorphic Encryption

short key (A, B)

key extension

密钥"拉长"

Table $=$ $\mathrm{Enc}_A(C)$ | $\mathrm{Enc}_B(D)$

long key (C, D)

仅支持 Bounded Integer

不同的环!

$\mathrm{Enc}_{Ax+B}(Cx+D)$

Garbling

Paillier: Table size $= O(b+\lambda)$

Recap Paillier

primes p, $p' = 2p+1$, q, $q' = 2q+1$

let $N = pq$

DCR assumption: $g^k \bmod N^2 \approx_c g^k (1+N)^m \bmod N^2$

Paillier Encryption: $Enc_k(m) = g^k (1+N)^m \bmod N^2$

DCR assumption: primes p, $p' = 2p+1$, q, $q' = 2q+1$

let $N = p'q'$

for random k, $g^k \bmod N^2 \approx_c g^k (1+N)^m \bmod N^2$

g: order pq

DCR assumption: primes p, $p' = 2p+1$, q, $q' = 2q+1$

let $N = p'q'$

for random k, $g^k \bmod N^2 \approx_c g^k (1+N)^m \bmod N^2$

Paillier Encryption:

$$Enc_{key}(msg) = g^{key} (1+N)^{msg} \bmod N^2$$

DCR assumption: primes p, $p' = 2p+1$, q, $q' = 2q+1$

let $N = p'q'$

for random k, $g^k \bmod N^2 \approx_c g^k (1+N)^m \bmod N^2$

Paillier Encryption:

$$Enc_{Ax+B}(\vec{C}x+\vec{D}) = g^{Ax+B}(1+N)^{\vec{C}x+\vec{D}} \bmod N^2$$

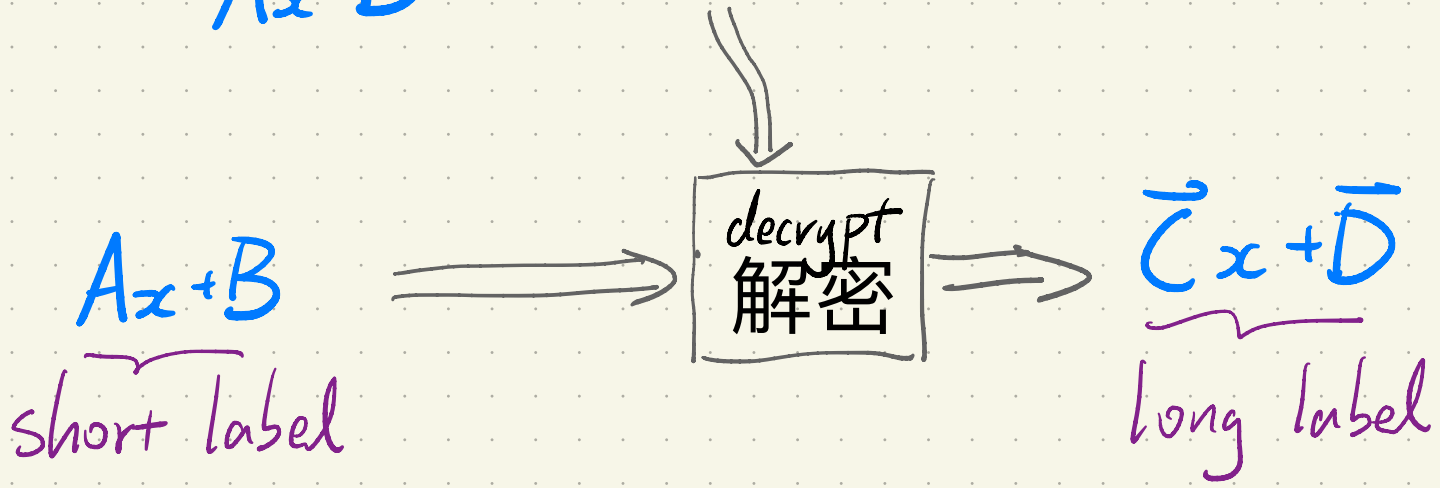
DCR assumption: primes p, $p'=2p+1$, q, $q'=2q+1$

let $N=p'q'$

for random k, $g^k \bmod N^2 \approx_c g^k (1+N)^m \bmod N^2$

Paillier Encryption:

$$Enc_{Ax+B}(\vec{C}x+\vec{D}) = g^{Ax+B}(1+N)^{\vec{C}x+\vec{D}} \bmod N^2$$



DCR assumption: primes p, $p'=2p+1$, q, $q'=2q+1$

small-key $N > 2^{2\lambda+b}$ let $N = p'q'$

for random $k \in [0, 2^{\lambda})$ $g^k \bmod N^2 \approx_c g^k (1+N)^m \bmod N^2$

assume $0 \leq x < 2^b$ Bounded Integer

Paillier Encryption:

$$Enc_{Ax+B}(\vec{C}x+\vec{D}) = g^{Ax+B}(1+N)^{\vec{C}x+\vec{D}} \bmod N^2$$

$A \xleftarrow{\$} [0, 2^{\lambda})$ $B \xleftarrow{\$} [0, 2^{\lambda+b+\lambda})$

$Ax+B$ → decrypt 解密 → $\vec{C}x+\vec{D} \bmod N = \vec{C}x+\vec{D}$

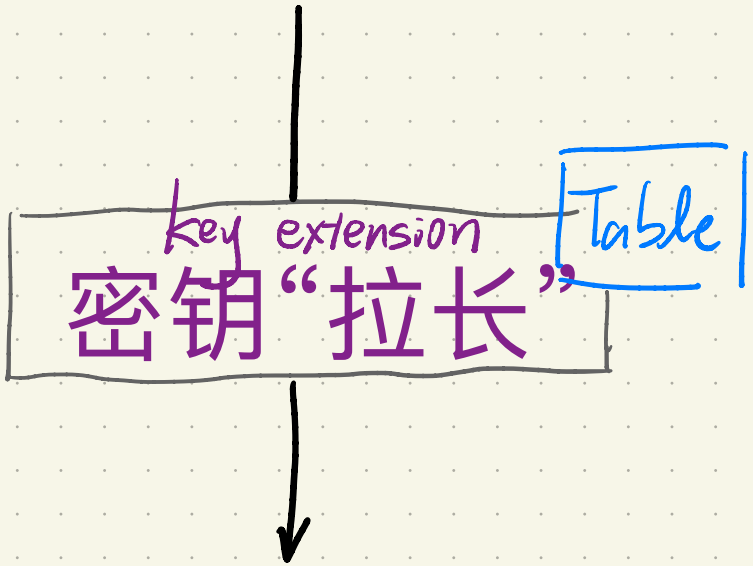
$Ax+B$: short label

$\vec{C}x+\vec{D}$: long label

Next Idea: Bit Decomposition

short arithmetic key (A.B)

label $Ax+B$



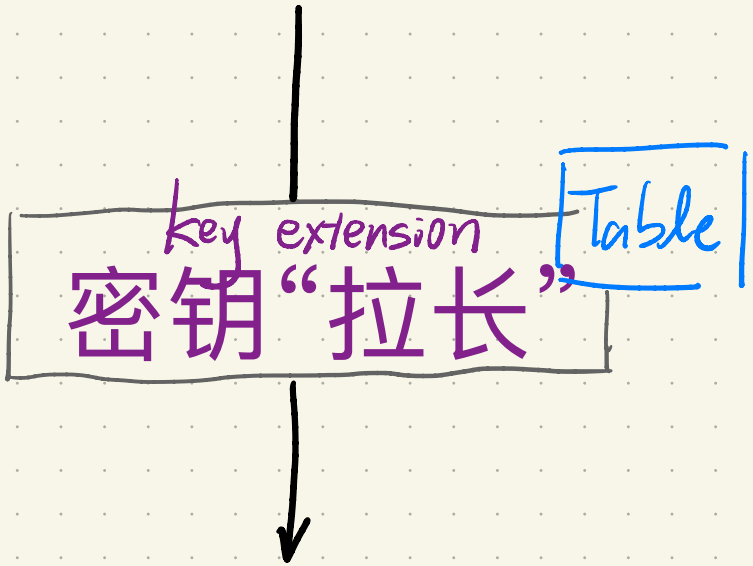
long arithmetic key (C.D)

label $Cx+D$

Next Idea: Bit Decomposition

short arithmetic key (A, B)

label $Ax + B$



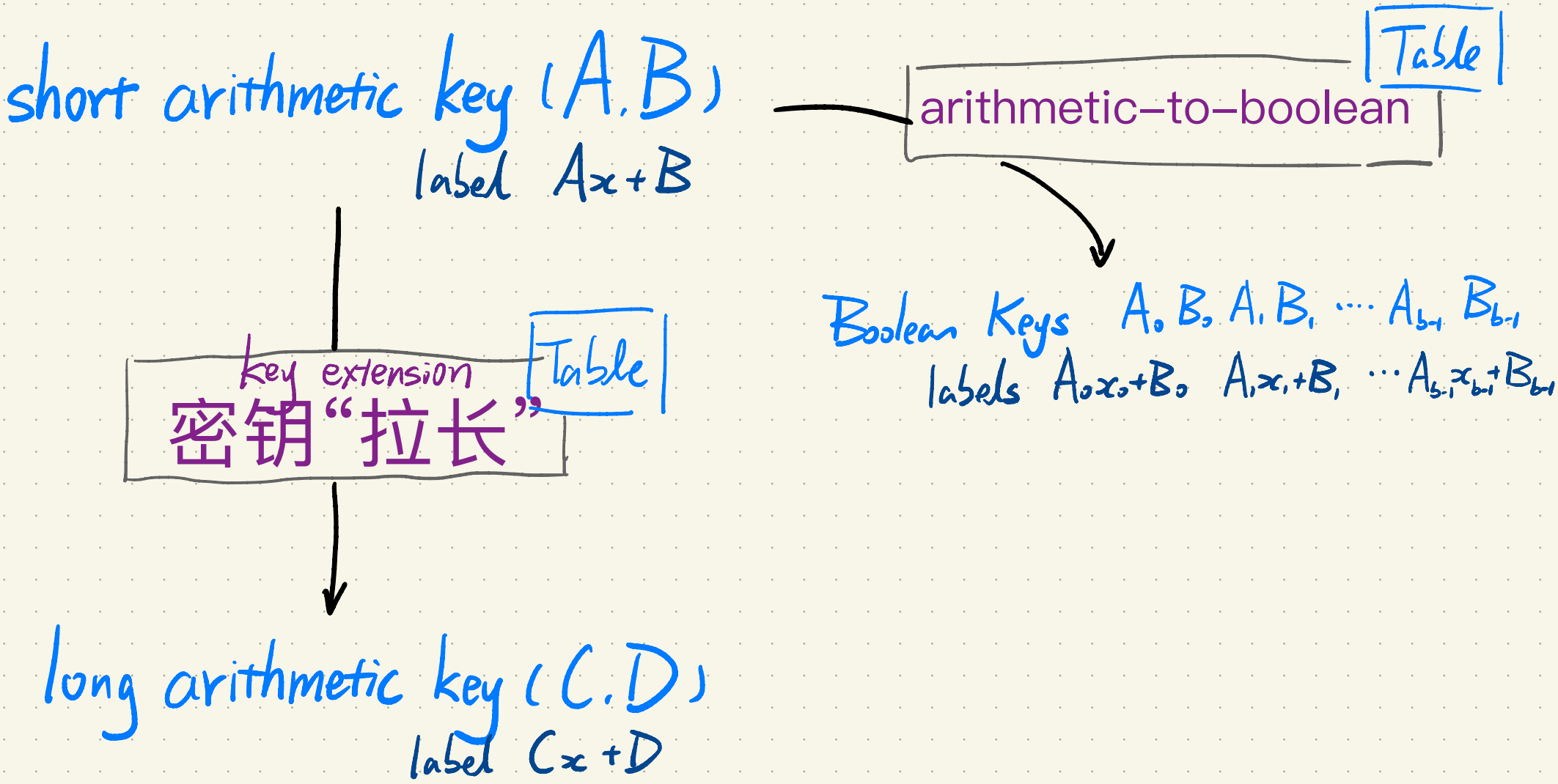
long arithmetic key (C, D)

label $Cx + D$

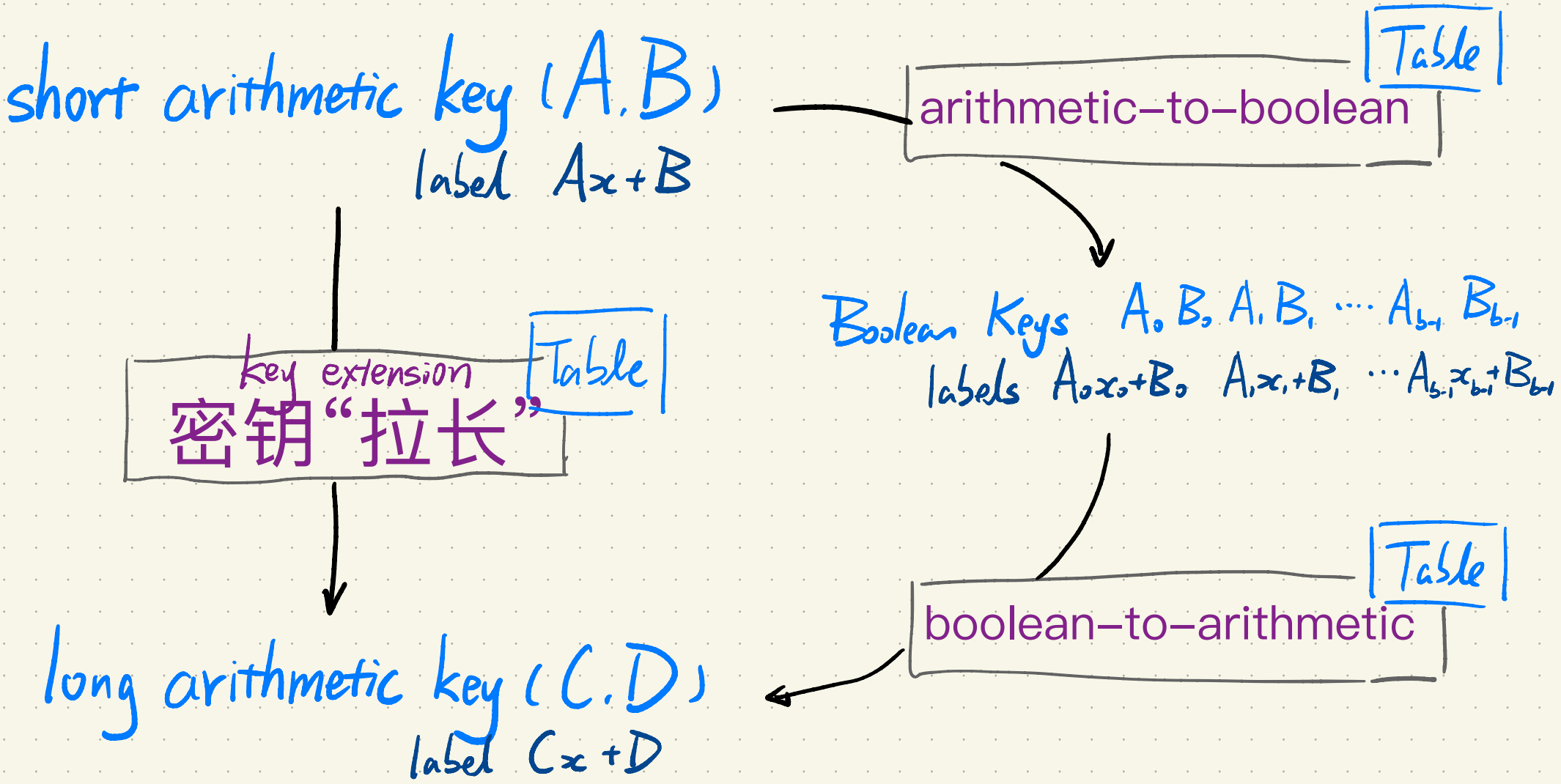
Boolean Keys $A_0\ B_0\ A_1\ B_1\ \cdots\ A_{b-1}\ B_{b-1}$

labels $A_0x_0 + B_0\quad A_1x_1 + B_1\quad \cdots A_{b-1}x_{b-1} + B_{b-1}$

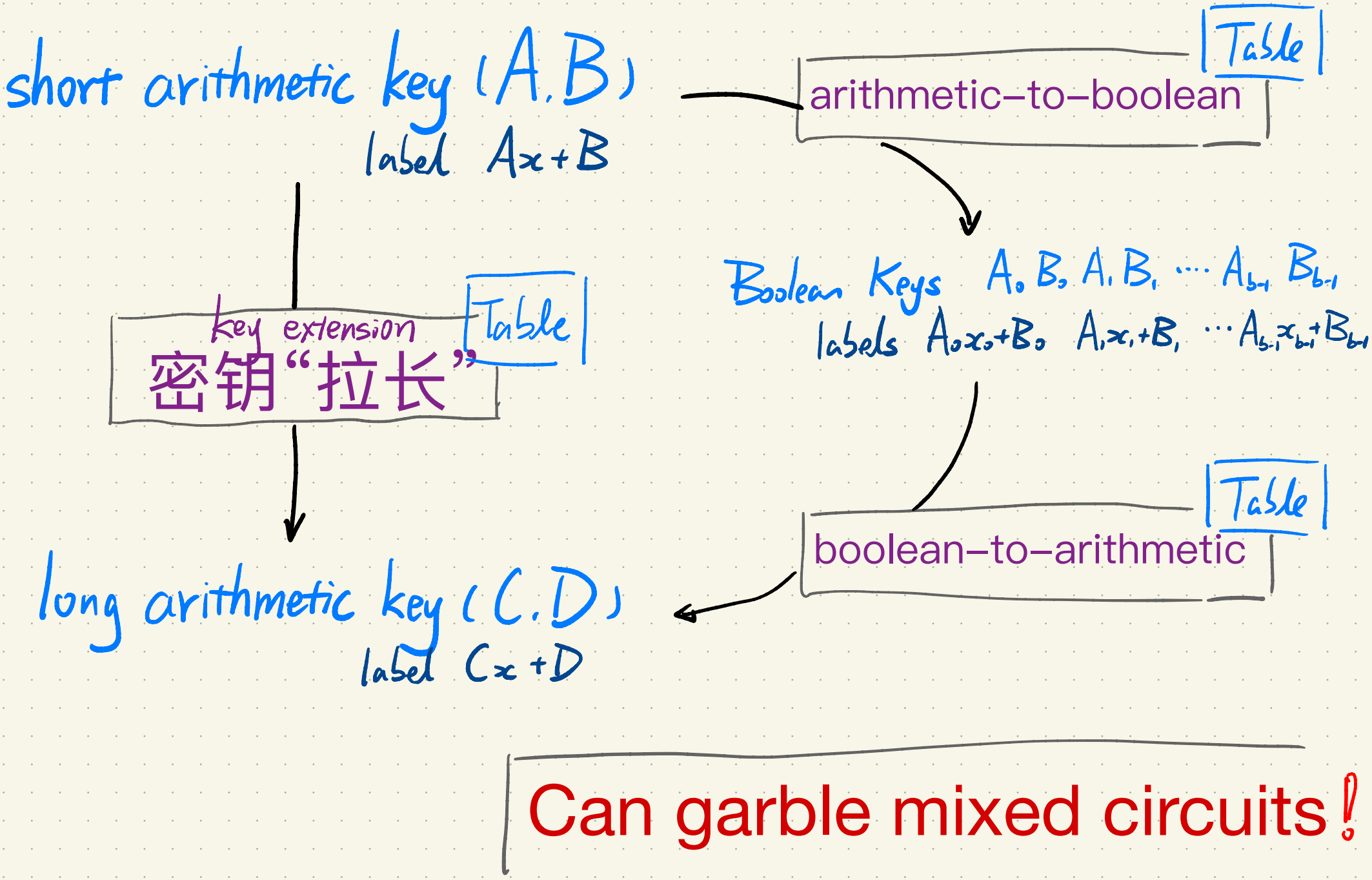
Next Idea: Bit Decomposition



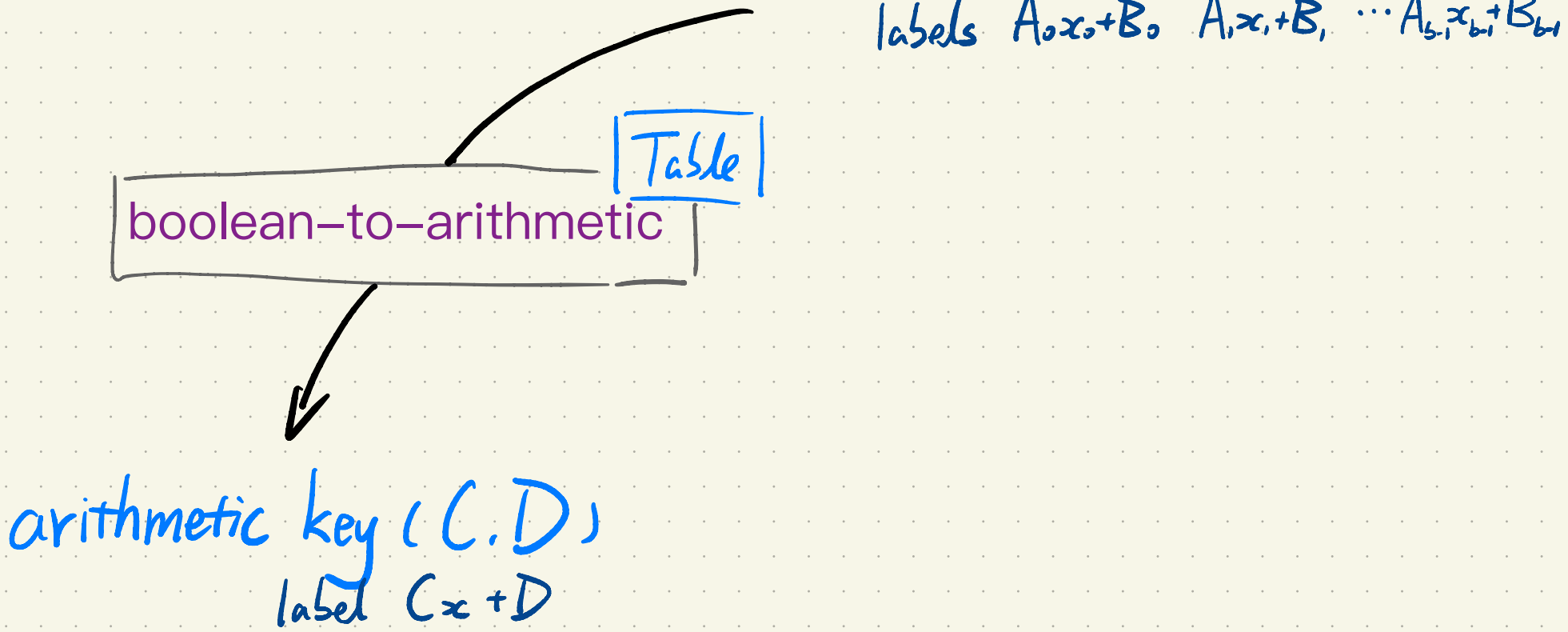
Next Idea: Bit Decomposition

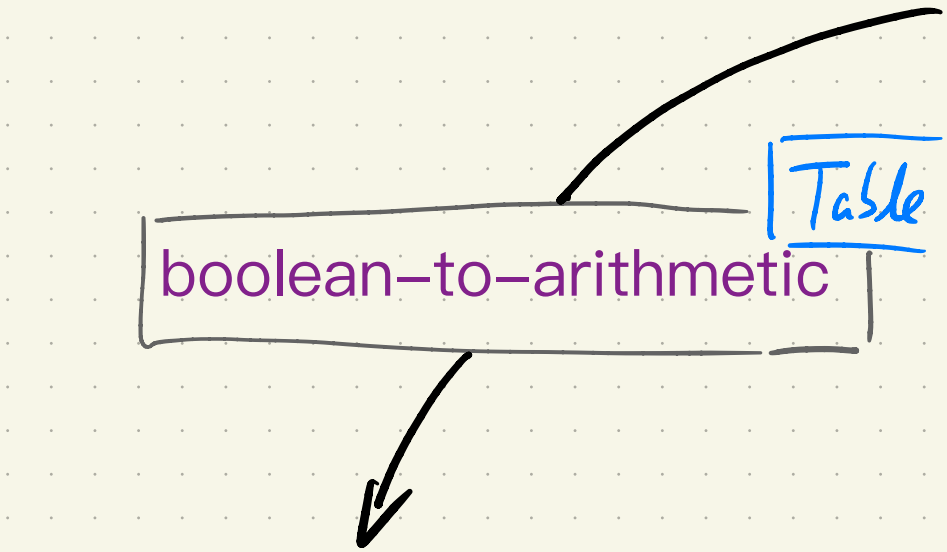


Next Idea: Bit Decomposition



Easy Direction





Boolean Keys $A_0\ B_0\ A_1\ B_1\ \cdots\ A_{b-1}\ B_{b-1}$

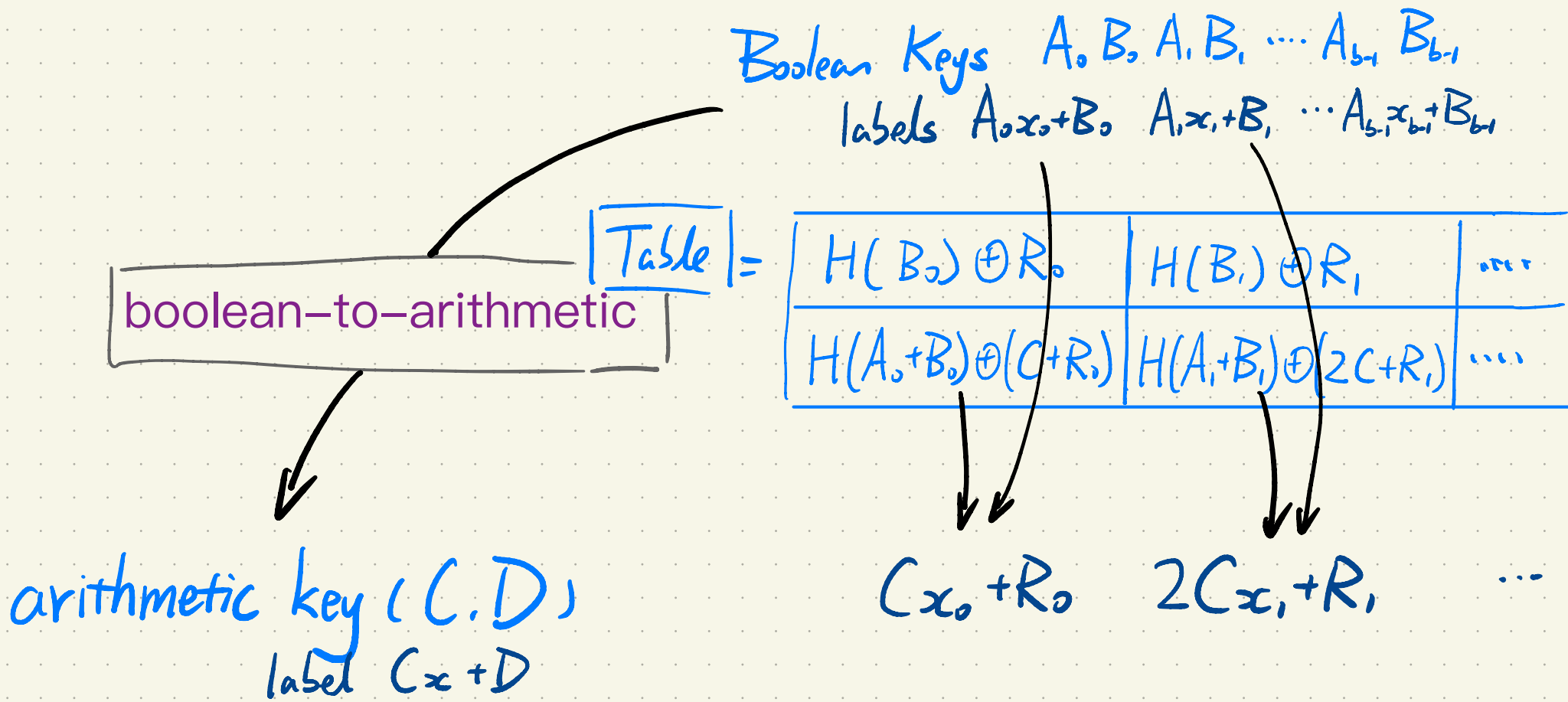
labels $A_0x_0+B_0\quad A_1x_1+B_1\quad \cdots A_{b-1}x_{b-1}+B_{b-1}$

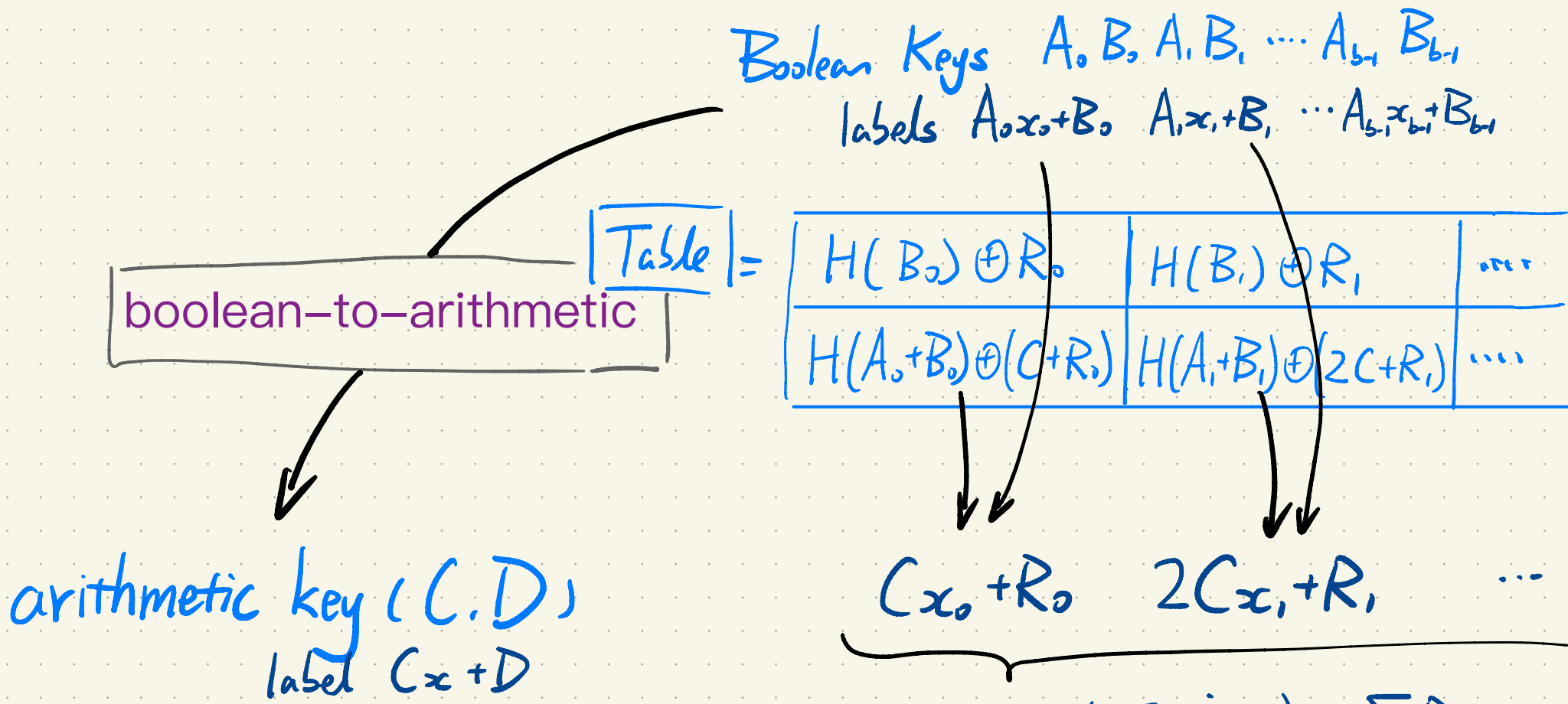
Table =

$H(B_0)\oplus R_0$	$H(B_1)\oplus R_1$	...
$H(A_0+B_0)\oplus(C+R_0)$	$H(A_1+B_1)\oplus(2C+R_1)$	...

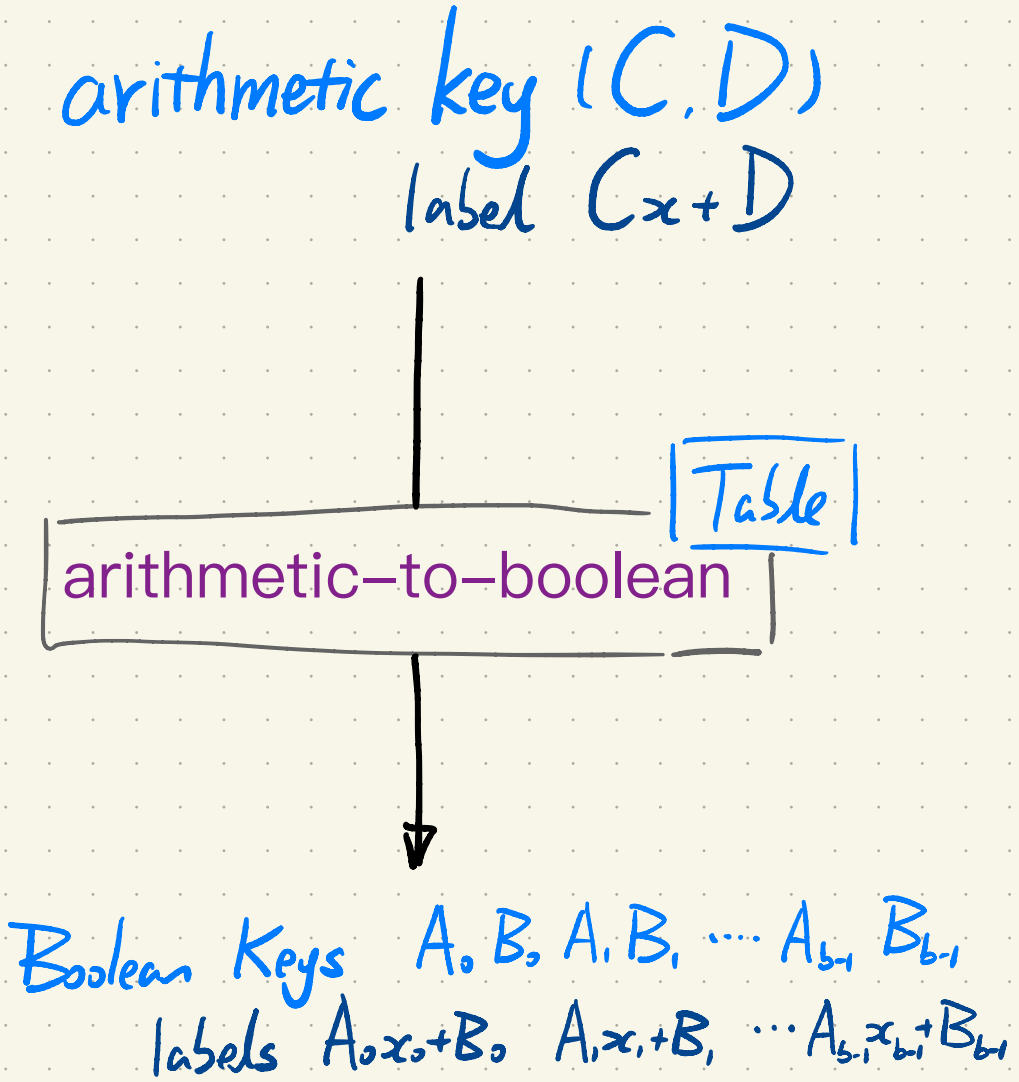
arithmetic key (C, D)

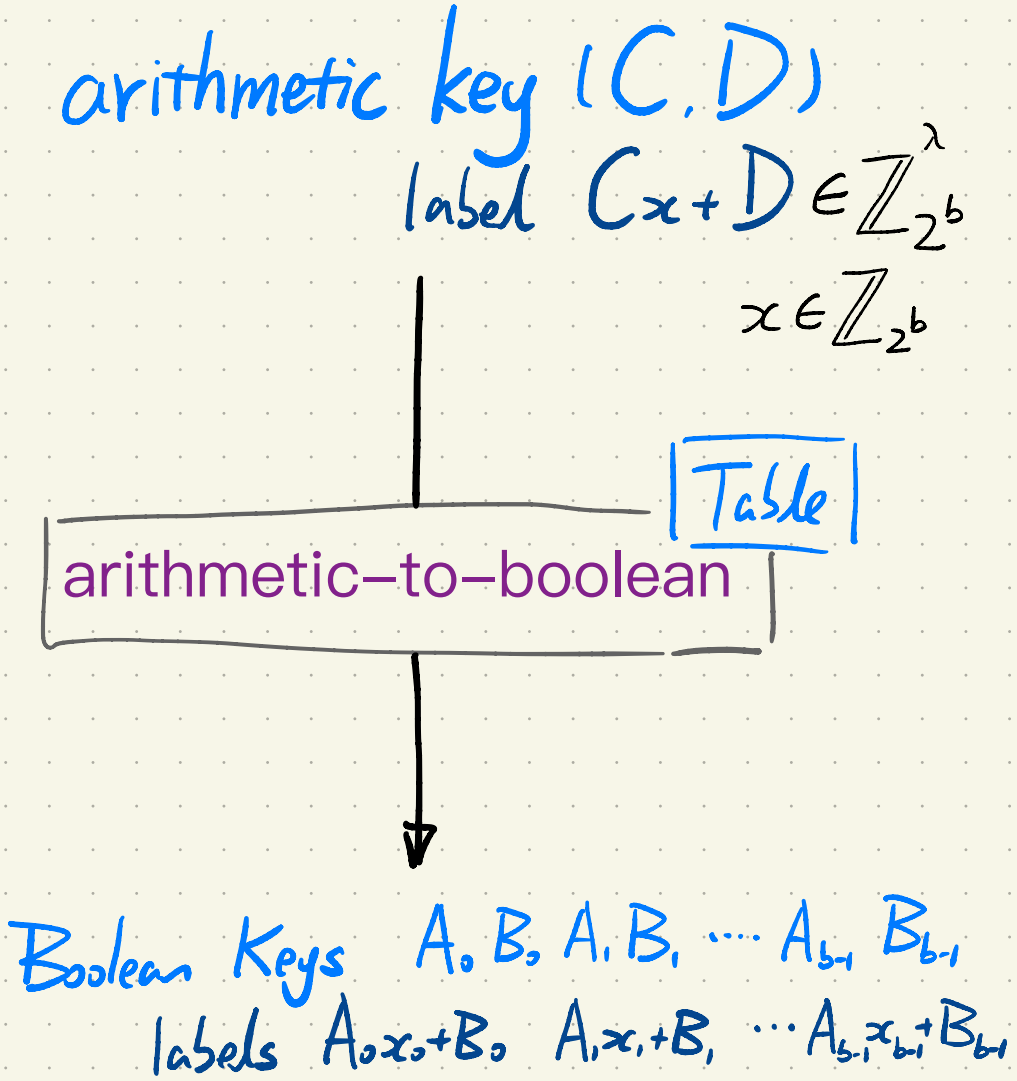
label $Cx+D$


Boolean Keys $A_0\ B_0\ A_1\ B_1 \cdots A_{b-1}\ B_{b-1}$
labels $A_0x_0+B_0$ $A_1x_1+B_1$ $\cdots A_{b-1}x_{b-1}+B_{b-1}$
Table =
$H(B_0)\oplus R_0$
$H(B_1)\oplus R_1$
$H(A_0+B_0)\oplus(C+R_0)$
$H(A_1+B_1)\oplus(2C+R_1)$
boolean-to-arithmetic
arithmetic key (C, D)
label $Cx+D$
Cx_0+R_0
$2Cx_1+R_1$


Boolean Keys $A_0\ B_0\ A_1\ B_1\ \cdots\ A_{b-1}\ B_{b-1}$
labels $A_0x_0+B_0$ $A_1x_1+B_1$ $\cdots A_{b-1}x_{b-1}+B_{b-1}$
Table =
$H(B_0)\oplus R_0$
$H(B_1)\oplus R_1$
$H(A_0+B_0)\oplus(C+R_0)$
$H(A_1+B_1)\oplus(2C+R_1)$
boolean-to-arithmetic
arithmetic key (C, D)
label $Cx+D$
Cx_0+R_0 $2Cx_1+R_1$ $\cdots$
sum = $C(\sum_i 2^i x_0)+\sum_i R_i$
$= Cx + D$

Hard Direction:

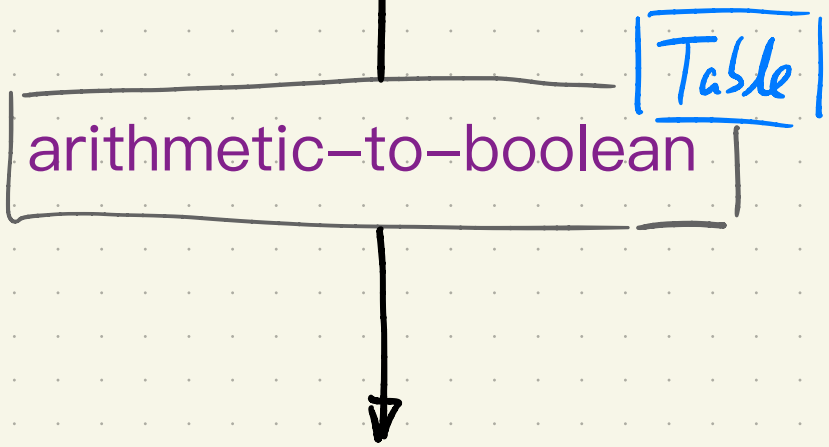



arithmetic key (C, D)
label $Cx+D \in \mathbb{Z}_{2^b}^{\lambda}$
$x \in \mathbb{Z}_{2^b}$
Table
arithmetic–to–boolean
Boolean keys $A_0 B_0 A_1 B_1 \cdots A_{b-1} B_{b-1}$
labels $A_0 x_0 + B_0$ $A_1 x_1 + B_1$ $\cdots A_{b-1} x_{b-1} + B_{b-1}$

arithmetic key (C, D)

label $Cx + D \in \mathbb{Z}_{2^b}^{\lambda}$

$x \in \mathbb{Z}_{2^b}$



Boolean Keys $A_0\ B_0\ A_1\ B_1\ \cdots\ A_{b-1}\ B_{b-1}$

labels $A_0 x_0 + B_0\ \ A_1 x_1 + B_1\ \cdots\ A_{b-1} x_{b-1} + B_{b-1}$

$$Cx + D \mod 2$$

$$= C_{\mod 2}\, x_{\mod 2} + D_{\mod 2}$$

$$= C_{\mod 2}\, x_0 + D_{\mod 2}$$

arithmetic key (C, D)

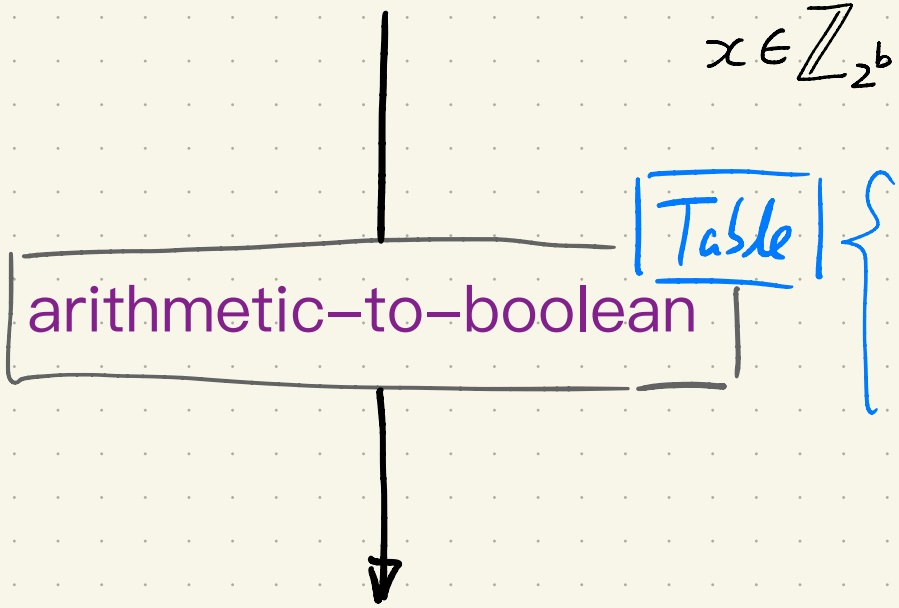
label $Cx + D \in \mathbb{Z}_{2^b}^{\lambda}$

$x \in \mathbb{Z}_{2^b}$

$$Cx + D \mod 2$$

$$= C_{\mod 2}\, x_{\mod 2} + D_{\mod 2}$$

$$= C_{\mod 2}\, x_0 + D_{\mod 2}$$

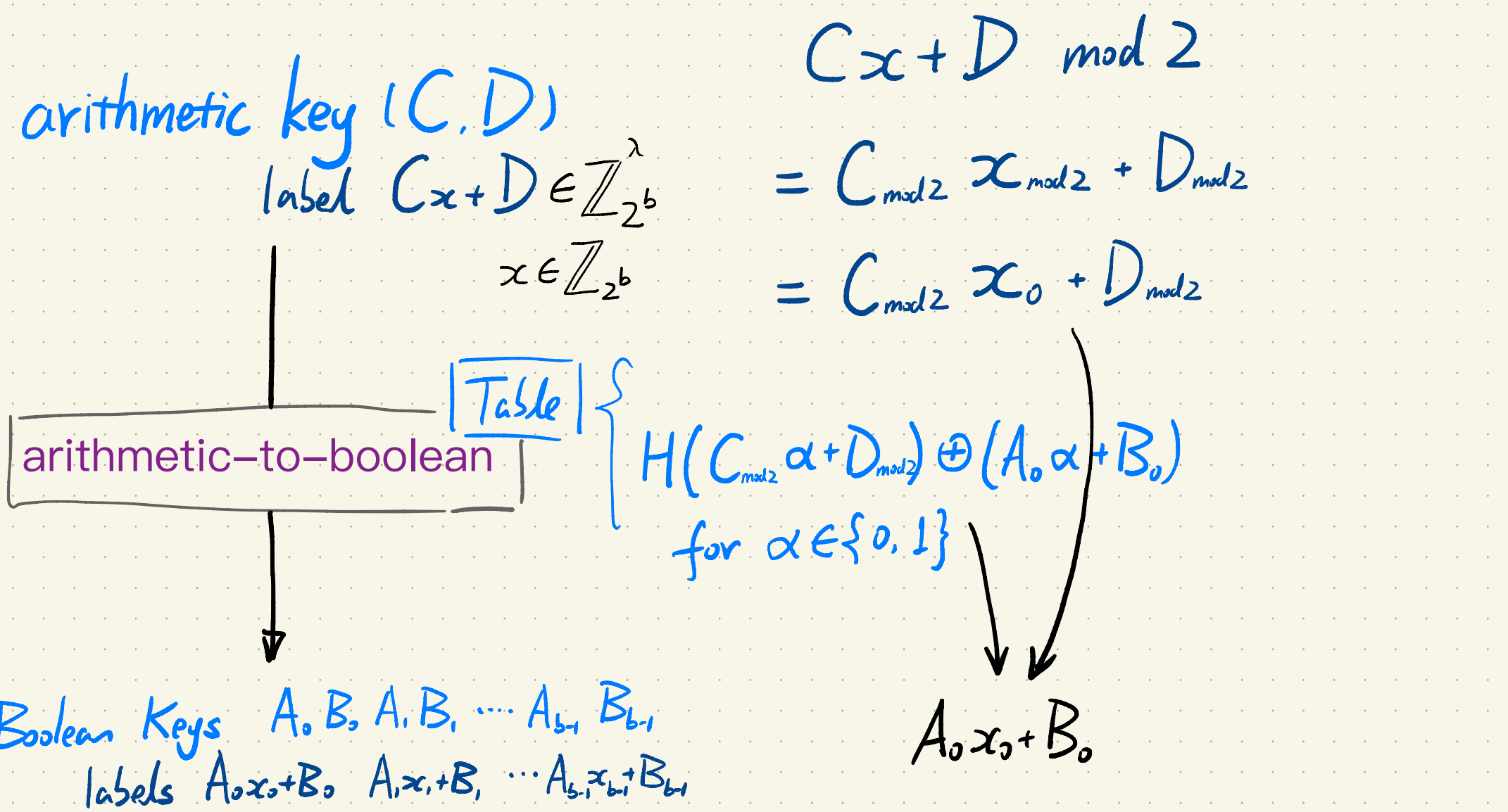


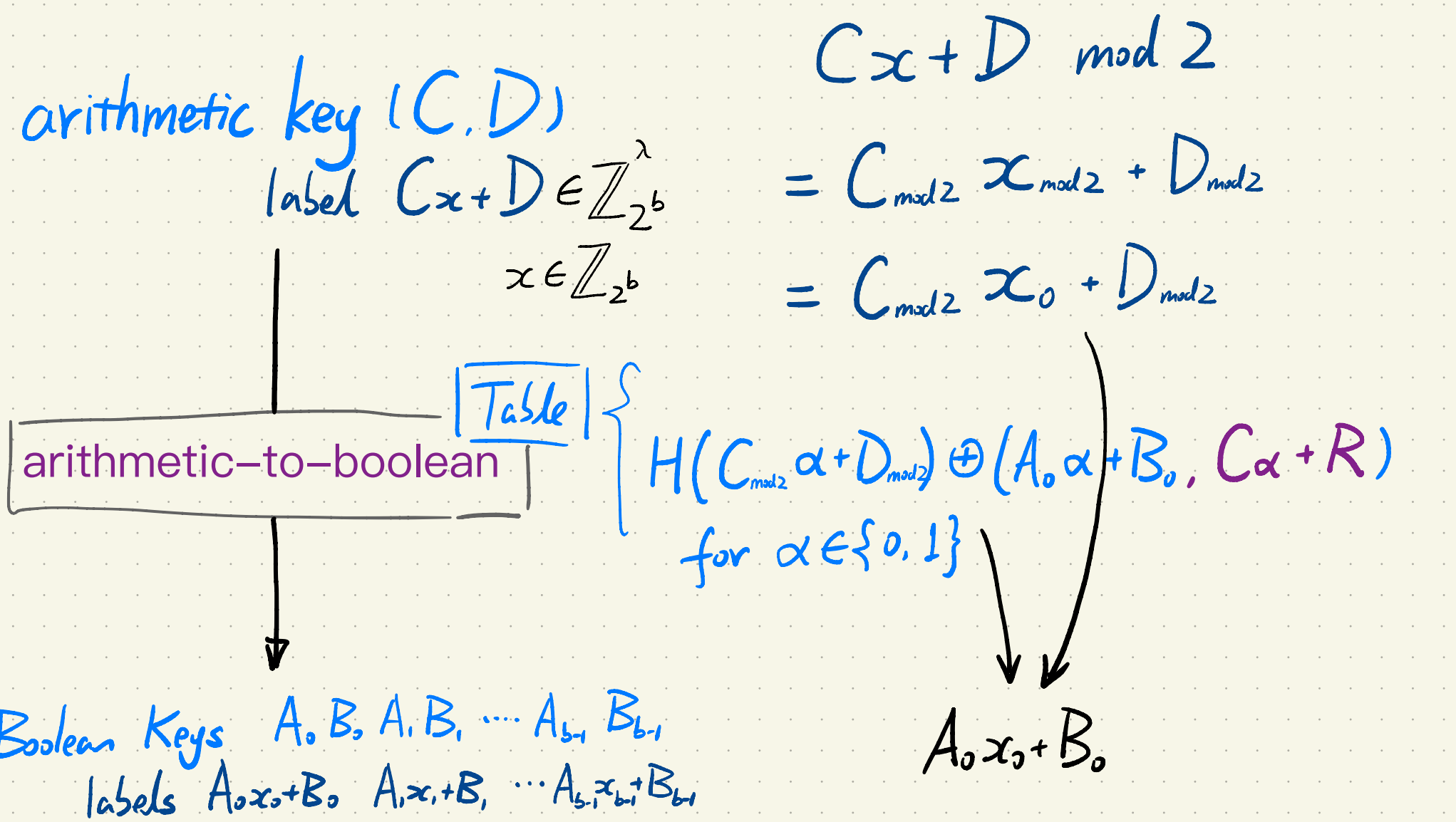
$$H(C_{\mod 2}\,\alpha + D_{\mod 2}) \oplus (A_0\,\alpha + B_0)$$

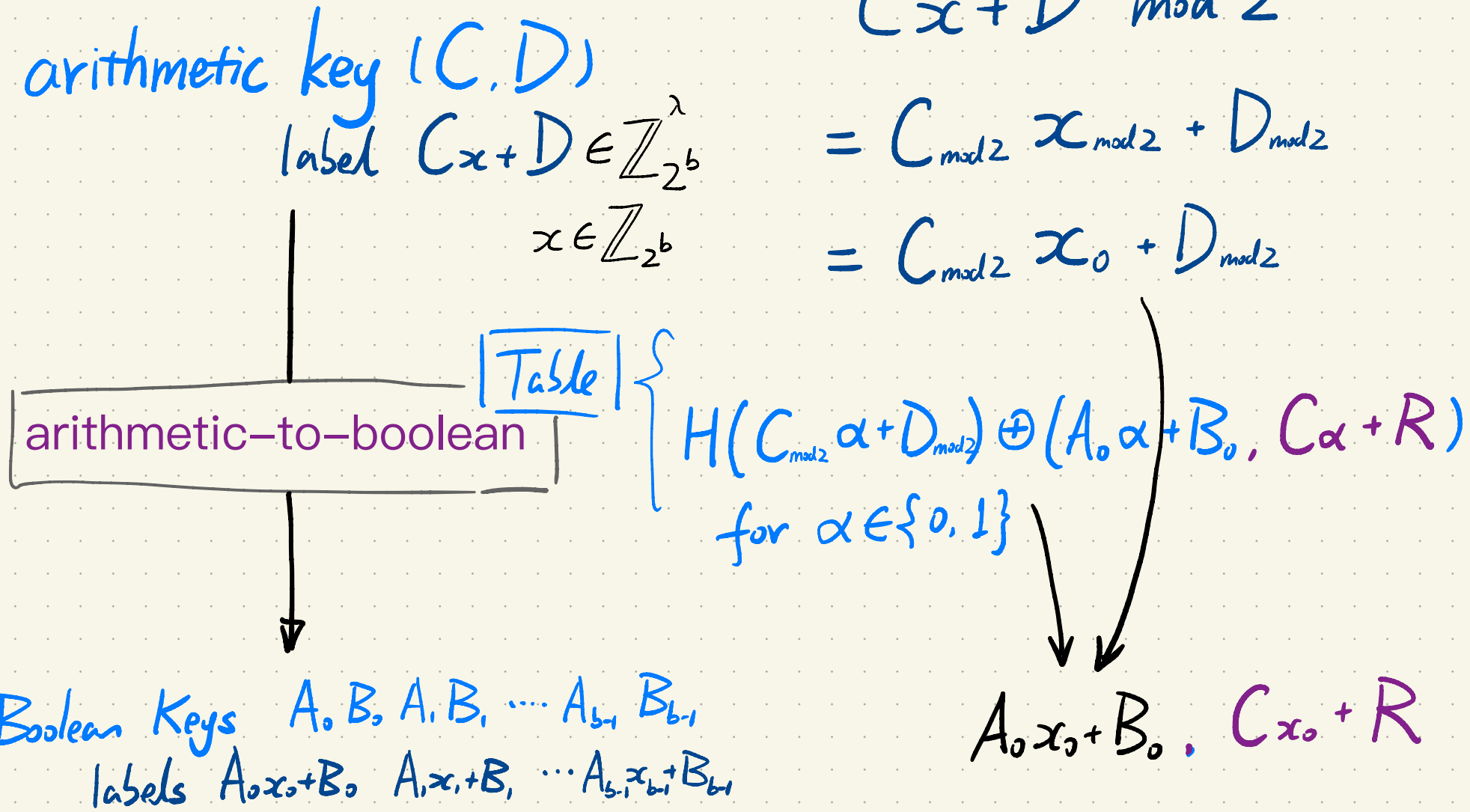
for $\alpha \in \{0, 1\}$

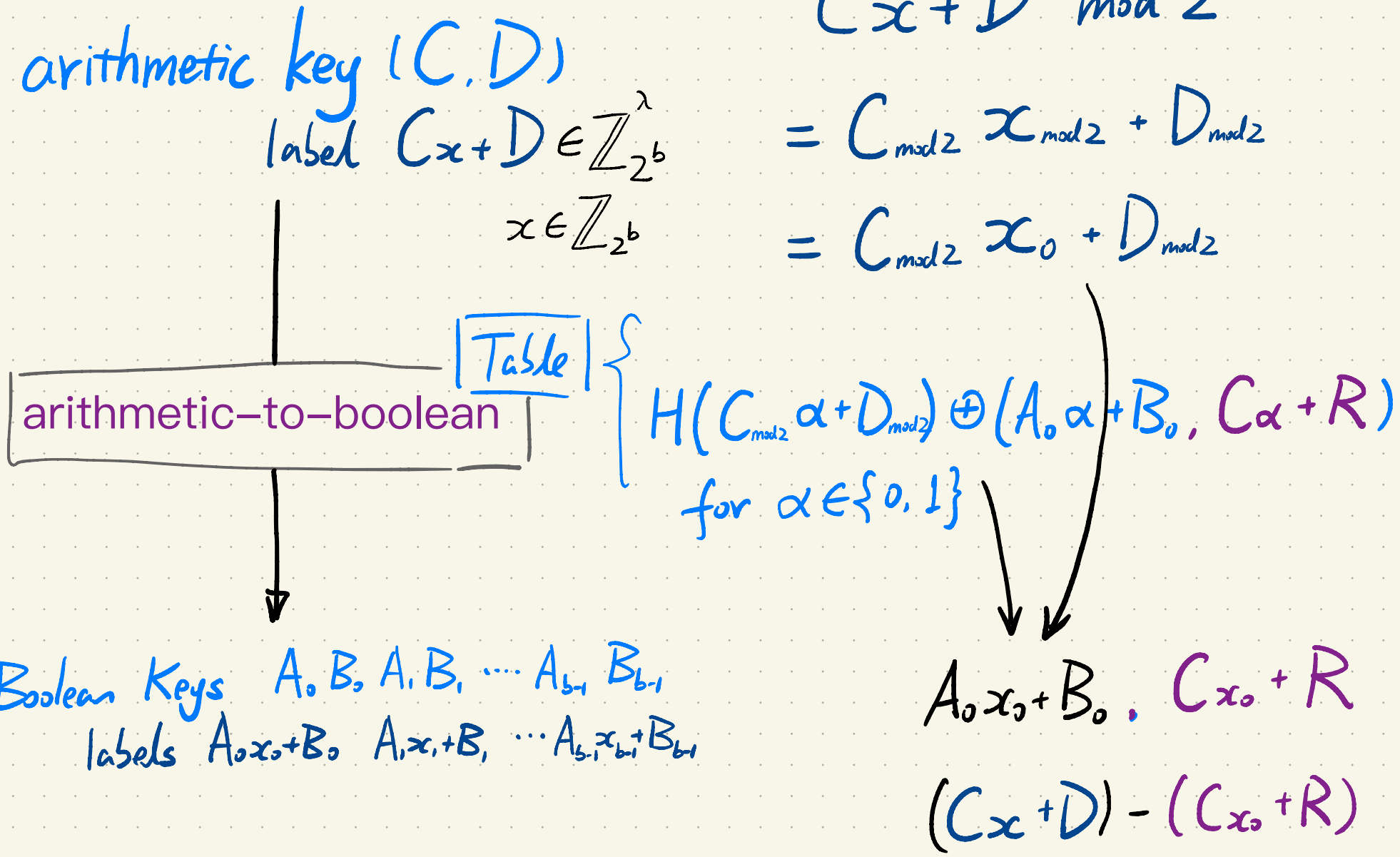
Boolean Keys $A_0\ B_0\ A_1\ B_1\ \cdots\ A_{b-1}\ B_{b-1}$

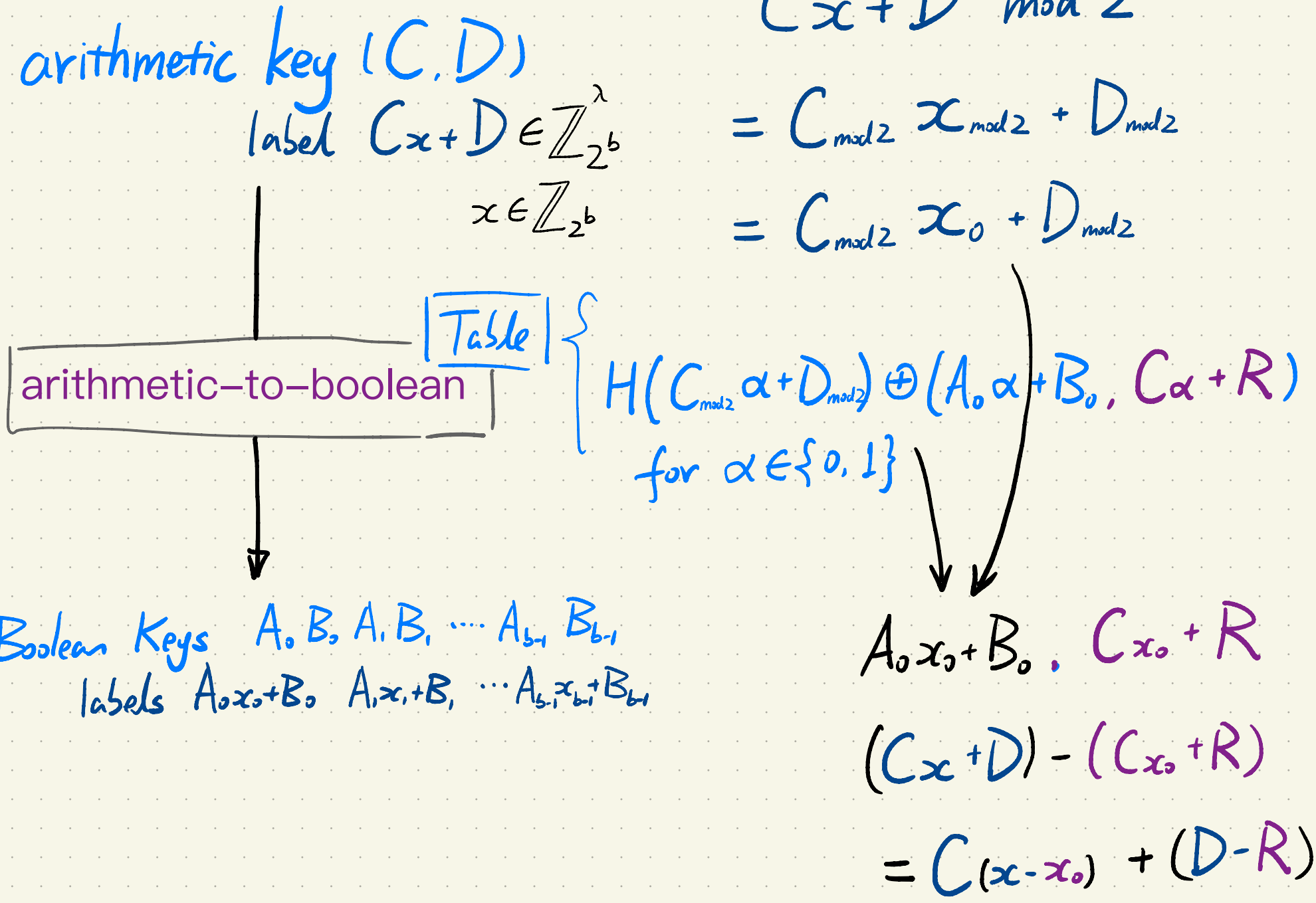
labels $A_0 x_0 + B_0\ \ A_1 x_1 + B_1\ \cdots\ A_{b-1} x_{b-1} + B_{b-1}$

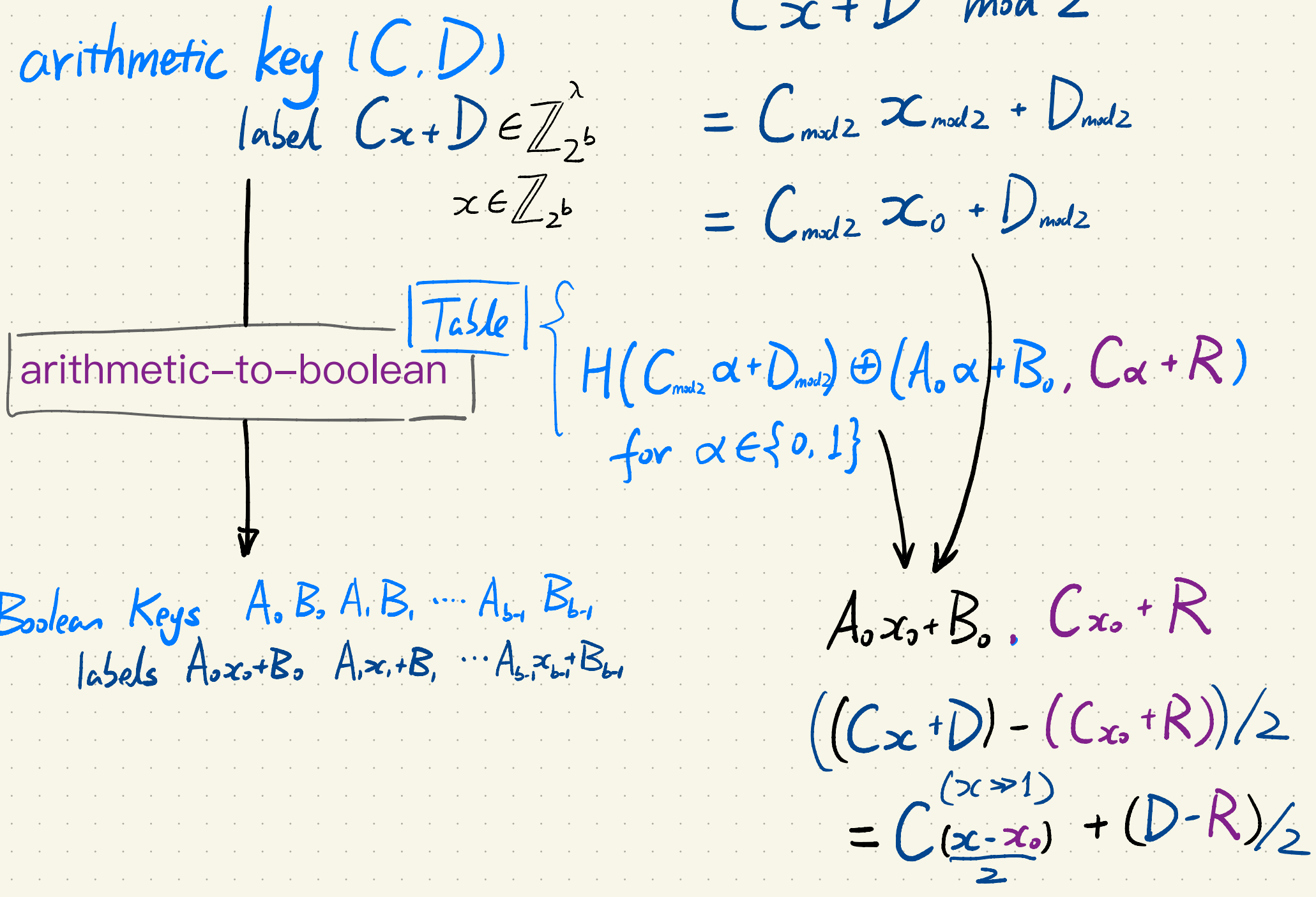

arithmetic key (C, D)
label $Cx+D \in \mathbb{Z}_{2^b}^{\lambda}$
$x \in \mathbb{Z}_{2^b}$
$Cx+D \mod 2$
$= C_{\mod 2}\, x_{\mod 2} + D_{\mod 2}$
$= C_{\mod 2}\, x_0 + D_{\mod 2}$
Table
arithmetic–to–boolean
$H(C_{\mod 2}\,\alpha + D_{\mod 2}) \oplus (A_0\,\alpha + B_0)$
for $\alpha \in \{0,1\}$
Boolean Keys $A_0\ B_0\ A_1\ B_1 \cdots A_{b-1}\ B_{b-1}$
labels $A_0x_0+B_0\ \ A_1x_1+B_1 \cdots A_{b-1}x_{b-1}+B_{b-1}$
$A_0x_0+B_0$


arithmetic key (C, D)
label $Cx+D \in \mathbb{Z}_{2^b}^{\lambda}$
$x \in \mathbb{Z}_{2^b}$
$Cx+D \mod 2$
$= C_{\mod 2}\, x_{\mod 2} + D_{\mod 2}$
$= C_{\mod 2}\, x_0 + D_{\mod 2}$
Table
arithmetic–to–boolean
$H(C_{\mod 2}\alpha + D_{\mod 2}) \oplus (A_0\alpha + B_0, C\alpha + R)$
for $\alpha \in \{0,1\}$
Boolean Keys $A_0\ B_0\ A_1\ B_1 \cdots A_{b-1}\ B_{b-1}$
labels $A_0x_0+B_0\ \ A_1x_1+B_1 \cdots A_{b-1}x_{b-1}+B_{b-1}$
$A_0x_0+B_0$


arithmetic key (C, D)
label $Cx+D \in \mathbb{Z}_{2^b}^{\lambda}$
$x \in \mathbb{Z}_{2^b}$
$Cx+D \mod 2$
$= C_{\mod 2}\, x_{\mod 2} + D_{\mod 2}$
$= C_{\mod 2}\, x_0 + D_{\mod 2}$
Table
arithmetic–to–boolean
$H(C_{\mod 2}\alpha + D_{\mod 2}) \oplus (A_0\alpha + B_0,\ C\alpha + R)$
for $\alpha \in \{0,1\}$
Boolean Keys $A_0\ B_0\ A_1\ B_1 \cdots A_{b-1}\ B_{b-1}$
labels $A_0x_0+B_0\ \ A_1x_1+B_1 \cdots A_{b-1}x_{b-1}+B_{b-1}$
$A_0x_0+B_0,\ Cx_0+R$


arithmetic key (C, D)
label $Cx+D \in \mathbb{Z}_{2^b}^{\lambda}$
$x \in \mathbb{Z}_{2^b}$
$Cx+D \mod 2$
$= C_{\mod 2}\, x_{\mod 2} + D_{\mod 2}$
$= C_{\mod 2}\, x_0 + D_{\mod 2}$
Table
arithmetic–to–boolean
$H(C_{\mod 2}\alpha + D_{\mod 2}) \oplus (A_0\alpha + B_0,\ C\alpha + R)$
for $\alpha \in \{0,1\}$
Boolean Keys $A_0\ B_0\ A_1\ B_1 \cdots A_{b-1}\ B_{b-1}$
labels $A_0x_0+B_0\ \ A_1x_1+B_1 \cdots A_{b-1}x_{b-1}+B_{b-1}$
$A_0x_0+B_0,\ Cx_0+R$
$(Cx+D)-(Cx_0+R)$


arithmetic key (C, D)
label $Cx+D \in \mathbb{Z}_{2^b}^{\lambda}$
$x \in \mathbb{Z}_{2^b}$
arithmetic–to–boolean
Table
$H(C_{mod2}\alpha + D_{mod2}) \oplus (A_0\alpha + B_0, C\alpha + R)$
for $\alpha \in \{0,1\}$
Boolean Keys $A_0\ B_0\ A_1\ B_1 \cdots A_{b-1}\ B_{b-1}$
labels $A_0x_0+B_0\ \ A_1x_1+B_1 \cdots A_{b-1}x_{b-1}+B_{b-1}$
$Cx + D \mod 2$
$= C_{mod2}\, x_{mod2} + D_{mod2}$
$= C_{mod2}\, x_0 + D_{mod2}$
$A_0x_0+B_0,\ Cx_0+R$
$(Cx+D) - (Cx_0+R)$
$= C(x-x_0) + (D-R)$


arithmetic key (C, D)
label $Cx+D \in \mathbb{Z}_{2^b}^{\lambda}$
$x \in \mathbb{Z}_{2^b}$
arithmetic–to–boolean
Table
$H(C_{mod2}\alpha + D_{mod2}) \oplus (A_0\alpha + B_0, C\alpha + R)$
for $\alpha \in \{0,1\}$
Boolean Keys $A_0\ B_0\ A_1\ B_1 \cdots A_{b-1}\ B_{b-1}$
labels $A_0x_0+B_0\ \ A_1x_1+B_1 \cdots A_{b-1}x_{b-1}+B_{b-1}$
$Cx + D \mod 2$
$= C_{mod2}\, x_{mod2} + D_{mod2}$
$= C_{mod2}\, x_0 + D_{mod2}$
$A_0x_0+B_0,\ Cx_0+R$
$((Cx+D) - (Cx_0+R))/2$
$= C^{(x \gg 1)}_{\frac{(x-x_0)}{2}} + (D-R)/2$

arithmetic-to-boolean

Table

Ring	Table size
$\mathbb{Z}_{2^b}$	$O(b^2\lambda)$

arithmetic-to-boolean

Table

Ring

Table size

$(p^k \approx 2^b)$

$\mathbb{Z}_{p^k}$

$O(k(k \log p + p)\lambda)$

optimization tricks

arithmetic-to-boolean

Table

Ring

Table size

$(p^k \approx 2^b)$ $\mathbb{Z}_{p^k}$ $O(k(k\log p + p)\lambda) = O(\frac{b^2}{\log b}\lambda)$ (when $p \approx b$)

arithmetic-to-boolean

Table

Ring

Table size

$(p^k \approx 2^b)$ $\mathbb{Z}_{p^k}$ $O(k(k\log p + p)\lambda) = O(\frac{b^2}{\log b}\lambda)$ (when $p \approx b$)

CRT

$(p_1^{k_1} \cdots p_t^{k_t} \approx 2^b)$ $\mathbb{Z}_{p_1^{k_1} p_2^{k_2} \cdots p_t^{k_t}}$ $O(b^{1.5}\lambda)$

arithmetic-to-boolean

Table

Ring

Table size

$(p^k \approx 2^b)$ $\mathbb{Z}_{p^k}$ $O(k(k\log p + p)\lambda) = O(\frac{b^2}{\log b}\lambda)$ (when $p \approx b$)

arithmetic-to-boolean

Table

Ring	Table size
$(p^k \approx 2^b)$ $\mathbb{Z}_{p^k}$	$O(k(k\log p + p)\lambda) = O(\frac{b^2}{\log b}\lambda)$ (when $p \approx b$)
multiple-and-shift trick	
$q \approx 2^b$ $\mathbb{Z}_q$	$O(\frac{b^2}{\log b}\lambda)$

arithmetic-to-boolean

Table

Ring

Table size

(when $p \approx b$)

$(p^k \approx 2^b)$ $\quad \mathbb{Z}_{p^k} \quad O(k(k \log p + p)\lambda) = O(\frac{b^2}{\log b}\lambda)$

multiple-and-shift trick

$q \approx 2^b$ $\quad \mathbb{Z}_q \quad O(\frac{b^2}{\log b}\lambda)$

multiple-and-shift trick

In $\forall q \; \exists m, k_\varepsilon$ s.t. $\forall 0 \le x < p^k$

$$\left\lfloor \frac{x}{q} \right\rfloor = \left\lfloor \frac{m x \mod p^{2k+l}}{p^{k+k_\varepsilon}} \right\rfloor$$

arithmetic-to-boolean

Table

Ring	Table size
$(p^k \approx 2^b)$ $\mathbb{Z}_{p^k}$	$O(k(k\log p + p)\lambda) = O(\frac{b^2}{\log b}\lambda)$ (when $p \approx b$)
multiple-and-shift trick	
$q \approx 2^b$ $\mathbb{Z}_q$	$O(\frac{b^2}{\log b}\lambda)$
CRT	
$(p_1^{k_1} \cdots p_t^{k_t} \approx 2^b)$ $\mathbb{Z}_{p_1^{k_1} p_2^{k_2} \cdots p_t^{k_t}}$	$O(\frac{b \log b}{\log \log b}\lambda)$

混淆一个代数加法门的代价

化为布尔电路 VS. 直接混淆代数电路

$O(b \cdot \lambda)$

無

混淆一个代数乘法门的代价

化为布尔电路 VS. 直接混淆代数电路

$O(b^2 \cdot \lambda)$

$O(b^{1.58} \cdot \lambda)$

Karatsuba $b > 500$

$O(b \cdot \log b \cdot \lambda)$

DFT $b > 10,000$

Cost	Assumption	Modulo
$O(\frac{b^2}{\log b} \cdot \lambda)$	RO	some large number

混淆一个代数乘法门的代价

化为布尔电路 VS. 直接混淆代数电路

$O(b^2 \cdot \lambda)$

$O(b^{1.58} \cdot \lambda)$

Karatsuba $b > 500$

$O(b \cdot \log b \cdot \lambda)$

DFT $b > 10,000$

Cost	Assumption	Modulo
$O(\frac{b^2}{\log b} \cdot \lambda)$	RO	some large number
$O(b+\lambda)$	DCR	bounded integer

混淆一个代数乘法门的代价

化为布尔电路 V.S. 直接混淆代数电路

$O(b^2 \cdot \lambda)$

$O(b^{1.58} \cdot \lambda)$

Karatsuba $b > 500$

$O(b \cdot \log b \cdot \lambda)$

DFT $b > 10,000$

Cost	Assumption	Modulo
$O(\frac{b^2}{\log b} \cdot \lambda)$	RO	some large number
$O(b+\lambda)$	DCR	bounded integer
$\tilde{O}((b+\lambda)^2)$	DCR LWE	some large number

混淆一个代数乘法门的代价

化为布尔电路 VS. 直接混淆代数电路

	Cost	Assumption	Modulo
	$O(b^2 \cdot \lambda)$		
	$O(b^{1.58} \cdot \lambda)$ Karatsuba $b > 500$		
	$O(b \cdot \log b \cdot \lambda)$ DFT $b > 10{,}000$		
Our work	$O(\frac{b^2}{\log b} \cdot \lambda)$	RO	some large number
	$O(b+\lambda)$	DCR	bounded integer
	$\hat{O}((b+\lambda)^2)$	DCR LWE	some large number

混淆一个代数乘法门的代价

化为布尔电路 VS. 直接混淆代数电路

$O(b^2 \cdot \lambda)$

$O(b^{1.58} \cdot \lambda)$

Karatsuba $b > 500$

$O(b \cdot \log b \cdot \lambda)$

DFT $b > 10,000$

	Cost	Assumption	Modulo
	$O(\frac{b^2}{\log b} \cdot \lambda)$	RO	some large number
Our work	$O(b+\lambda)$	DCR	bounded integer
Our work	$\hat{O}((b+\lambda)^2)$	DCR LWE	some large number
New Work	$O(\frac{b \log b \lambda}{\log \log b})$	RO	some large number
New Work	$O(b\lambda)$	DCR	any large number

混淆一个代数乘法门的代价

化为布尔电路 V.S. 直接混淆代数电路

$O(b^2 \cdot \lambda)$

$O(b^{1.58} \cdot \lambda)$

Karatsuba $b > 500$

$O(b \cdot \log b \cdot \lambda)$

DFT $b > 10,000$

	Cost	Assumption	Modulo
	$O(\frac{b^2}{\log b} \cdot \lambda)$	RO	some large number
Our work	$O(b+\lambda)$	DCR	bounded integer
Our work	$\hat{O}((b+\lambda)^2)$	DCR LWE	some large number
New Work	$O(\frac{b \log b \lambda}{\log \log b})$	RO	some large number & mixed computation
New Work	$O(b\lambda)$	DCR	any large number & mixed computation

Thanks

Q&A