ARIA 양자 구현 논문 리뷰

https://youtu.be/vEVPVxSrZyQ

한성대학교
HANSUNG UNIVERSITY

CryptoCraft LAB

논문

- Chauhan, A.K., Sanadhya, S.K.: **Quantum resource estimates of grover's key search on aria.** In: Security, Privacy, and Applied Cryptography Engineering: 10th International Conference, SPACE 2020, Kolkata, India, December 17–21, 2020, Proceedings 10, Springer (2020) 238–258 2, 9, 10, 11, 12, 13, 14, 15

- Yang, Y., Jang, K., Oh, Y., & Seo, H. (2023). **Depth-Optimized Quantum Implementation of ARIA**. *Cryptology ePrint Archive*.

Substitution Layer

$$S_1(\alpha) := \mathbf{A}.\alpha^{-1} + \mathbf{a}$$

$$\mathbf{A} = \begin{bmatrix} 1&0&0&0&1&1&1&1 \\ 1&1&0&0&0&1&1&1 \\ 1&1&1&0&0&0&1&1 \\ 1&1&1&1&0&0&0&1 \\ 1&1&1&1&1&0&0&0 \\ 0&1&1&1&1&1&0&0 \\ 0&0&1&1&1&1&1&0 \\ 0&0&0&1&1&1&1&1 \end{bmatrix} \text{ and } \mathbf{a} = \begin{bmatrix} 1\\1\\0\\0\\0\\1\\1\\0 \end{bmatrix}$$

$$S_1^{-1}(\alpha) := (\mathbf{A}^{-1}.(\alpha + \mathbf{a}))^{-1}$$

$$\mathbf{A}^{-1} = \begin{bmatrix} 0&0&1&0&0&1&0&1 \\ 1&0&0&1&0&0&1&0 \\ 0&1&0&0&1&0&0&1 \\ 1&0&1&0&0&1&0&0 \\ 0&1&0&1&0&0&1&0 \\ 0&0&1&0&1&0&0&1 \\ 1&0&0&1&0&1&0&0 \\ 0&1&0&0&1&0&1&0 \end{bmatrix} \text{ and } \mathbf{a} = \begin{bmatrix} 1\\1\\0\\0\\0\\1\\1\\0 \end{bmatrix}$$

$$S_2(\alpha) := \mathbf{B}.\alpha^{247} + \mathbf{b}$$

$$S_2(\alpha) := \mathbf{B}.(\alpha^{-1})^8 + \mathbf{b} = \mathbf{B}.\mathbf{C}.\alpha^{-1} + \mathbf{b}$$

$$= \mathbf{D}.\alpha^{-1} + \mathbf{b}$$

$$\mathbf{D} = \begin{bmatrix} 0&1&0&1&0&1&1&1 \\ 0&0&1&1&1&1&1&1 \\ 1&1&1&0&1&1&0&1 \\ 1&1&0&0&0&0&1&1 \\ 0&1&0&0&0&0&1&1 \\ 1&1&0&0&1&1&1&0 \\ 0&1&1&0&0&0&1&1 \\ 1&1&1&1&0&1&1&0 \end{bmatrix} \text{ and } \mathbf{b} = \begin{bmatrix} 0\\1\\0\\0\\0\\1\\1\\1 \end{bmatrix}$$

$$S_2^{-1}(\alpha) = (\mathbf{D}^{-1}.(\alpha + \mathbf{b}))^{-1}$$

$$\mathbf{D}^{-1} = \begin{bmatrix} 0&0&0&1&1&0&0&0 \\ 0&0&1&0&0&1&1&0 \\ 0&0&0&0&1&0&1&0 \\ 1&1&1&0&0&0&1&1 \\ 1&1&1&0&1&1&0&0 \\ 0&1&1&0&1&0&1&1 \\ 1&0&1&1&1&1&0&1 \\ 1&0&0&1&0&0&1&1 \end{bmatrix} \text{ and } \mathbf{b} = \begin{bmatrix} 0\\1\\0\\0\\0\\1\\1\\1 \end{bmatrix}$$

Substitution Layer

- Itoh-Tsujii algorithm
  - 곱셈과 제곱으로 이루어진 연산

$$\alpha^{-1} = \alpha^{254} = ((\alpha.\alpha^2).(\alpha.\alpha^2)^4.(\alpha.\alpha^2)^{16}.\alpha^{64})^2$$

- Squaring (제곱기)
  - PLU 분해 사용

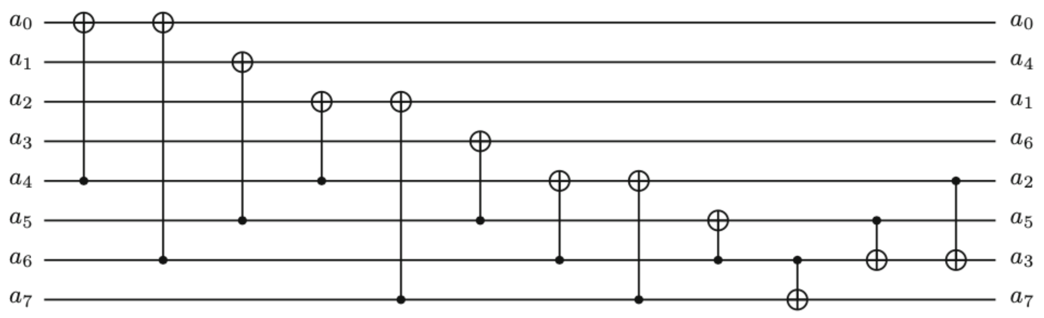


Fig. 1. Circuit for squaring in $\mathbb{F}_2[x]/(x^8 + x^4 + x^3 + x + 1)$.

CNOT gate: 12

Depth : 7

Substitution Layer

- Multiplication (곱셈기)
  - Schoolbook multiplication (Mastrovito)

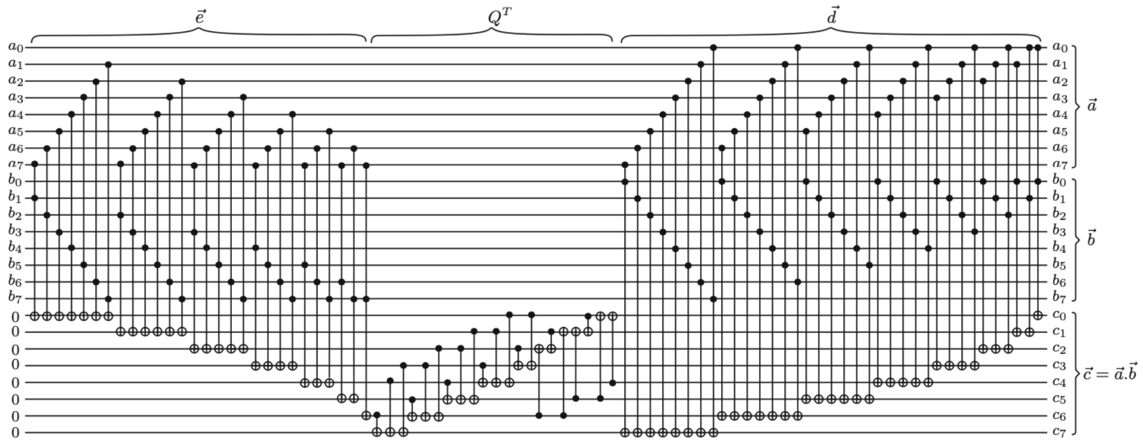


Fig. 2. Circuit for multiplier in $\mathbb{F}_2[x]/(x^8 + x^4 + x^3 + x + 1)$.

- Karatsuba Multiplication (Jang.et.al)
  - 카라추바 알고리즘을 재귀적으로 사용하여 Toffoli depth가 1인 곱셈 (81개 중 38개의 ancilla qubit 재사용)

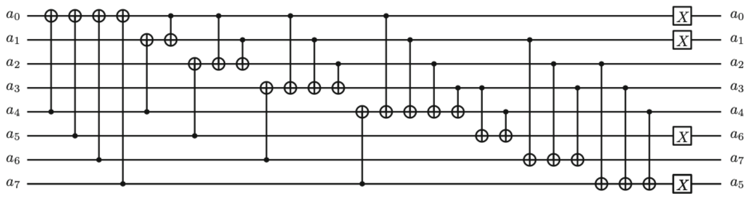
Table 1: Quantum resources required for multiplication.

Source	#Clifford	#T	Toffoli depth	Full depth
CMMP [2]	435	448	28	195
J++ [11]	390	189	1	28

※: The multiplication size n is 8.

Substitution Layer

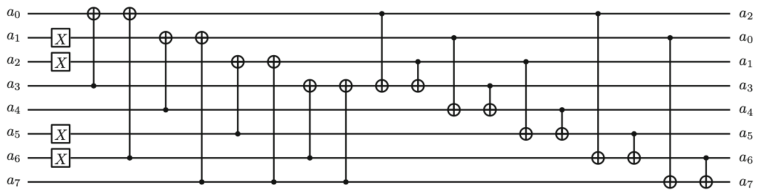
- Affine function
  - PLU 분해 사용
    - S_1



CNOT gate: 26

X gate : 4

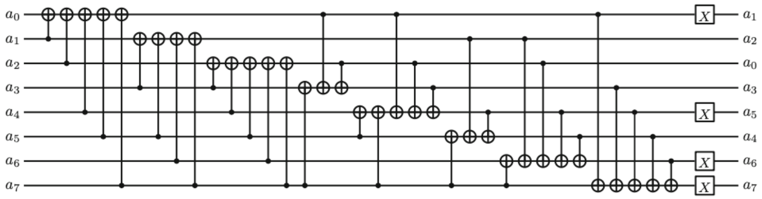
- S_1^{-1}



CNOT gate: 18

X gate : 4

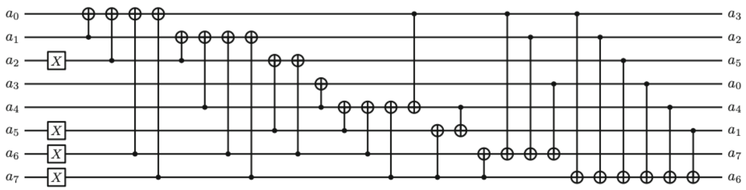
- S_2



CNOT gate: 35

X gate : 4

- S_2^{-1}



CNOT gate: 27

X gate : 4

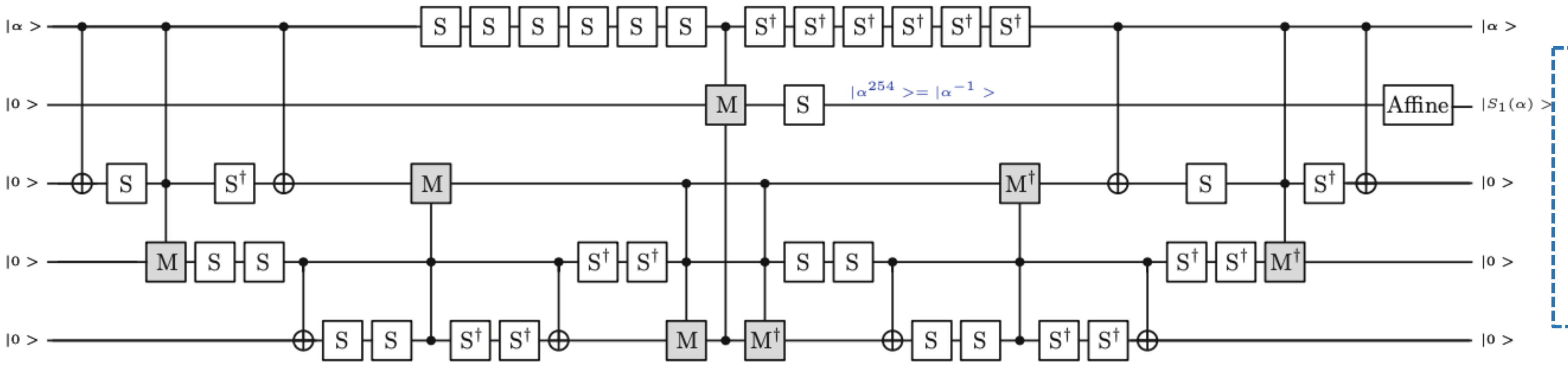
Substitution Layer

- S-box

$$\alpha^{-1} = \alpha^{254} = ((\alpha.\alpha^2).(\alpha.\alpha^2)^4.(\alpha.\alpha^2)^{16}.\alpha^{64})^2$$

- Yang.et.al
  - Squarings : 11
  - Multiplications : 4
  - Qubits : 162 (38)

- Chauhan. et. al



- Chauhan.et.al
  - Squarings : 33
  - Multiplications : 7
  - Qubits : 40 (24)

Substitution Layer

- Substitution Layer

- S_1

  Toffoli gates : 64 x 7 = 448

  CNOT gate: 12 x 33 + 21 x 7 + 26 = 569

  X gate : 4

- S_1^{-1}

  Toffoli gates : 64 x 7 = 448

  CNOT gate: 12 x 33 + 21 x 7 + 18 = 561

  X gate : 4

- S_2

  Toffoli gates : 64 x 7 = 448

  CNOT gate: 12 x 33 + 21 x 7 + 35 = 578

  X gate : 4

- S_2^{-1}

  Toffoli gates : 64 x 7 = 448

  CNOT gate: 12 x 33 + 21 x 7 + 27 = 570

  X gate : 4

- Substitution Layer

Toffoli gates : 448 x (4 x 4) = 7,168

CNOT gate: (569 + 561 +578 +570) x 4 = 9,112

X gate : 4 x (4 x 4) = 64

Diffusion Layer

- Diffusion Layer
  - PLU 분해 사용

$$
\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \\ y_9 \\ y_{10} \\ y_{11} \\ y_{12} \\ y_{13} \\ y_{14} \\ y_{15} \end{bmatrix} = \begin{bmatrix}
0&0&0&1&1&0&1&0&1&1&0&0&0&1&1&0 \\
0&0&1&0&0&1&0&1&1&1&0&0&1&0&0&1 \\
0&1&0&0&1&0&1&0&0&0&1&1&1&0&0&1 \\
1&0&0&0&0&1&0&1&0&0&1&1&0&1&1&0 \\
1&0&1&0&0&1&0&0&1&0&0&1&0&0&1&1 \\
0&1&0&1&1&0&0&0&0&1&1&0&0&0&1&1 \\
1&0&1&0&0&0&0&1&0&1&1&0&1&1&0&0 \\
0&1&0&1&0&0&1&0&1&0&0&1&1&1&0&0 \\
1&1&0&0&1&0&0&1&0&0&1&0&0&1&0&1 \\
1&1&0&0&0&1&1&0&0&0&0&1&1&0&1&0 \\
0&0&1&1&0&1&1&0&1&0&0&0&0&1&0&1 \\
0&0&1&1&1&0&0&1&0&1&0&0&1&0&1&0 \\
0&1&1&0&0&0&1&1&0&1&0&1&1&0&0&0 \\
1&0&0&1&0&0&1&1&1&0&1&0&0&1&0&0 \\
1&0&0&1&1&1&0&0&0&1&0&1&0&0&1&0 \\
0&1&1&0&1&1&0&0&1&0&1&0&0&0&0&1
\end{bmatrix} \cdot \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \\ x_{10} \\ x_{11} \\ x_{12} \\ x_{13} \\ x_{14} \\ x_{15} \end{bmatrix}
$$

- Yang.et.al
  - CNOT gates : 768
  - Depth : 31

- Chauhan.et.al
  - CNOT gates : 768
  - Depth : 26

Round function

- Round function

Therefore, the total number of quantum gates needed to implement the substitution layer are as follows.

- Total number of Toffoli gates $= 448 \times (4 \times 4) = 7,168$
- Total number of CNOT gates $= (569 + 561 + 578 + 570) \times 4 = 9,112$
- Total number of Pauli-X gates $= 4 \times (4 \times 4) = 64$.

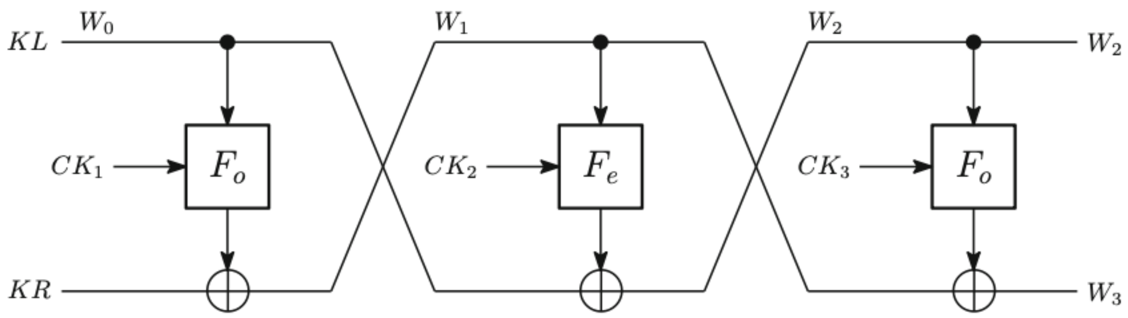
3. **Diffusion Layer (DL):** It is a linear operation and implementing it requires only 768 CNOT gates.

Therefore, one round of ARIA requires the following number of gates:

- Total number of Toffoli gates $= 7,168$
- Total number of CNOT gates $= 128 + 9,112 + 768 = 10,008$
- Total number of Pauli-X gates $= 64$.

Key Schedule

- Key Schedule - Chauhan.et.al
  - Key initialization

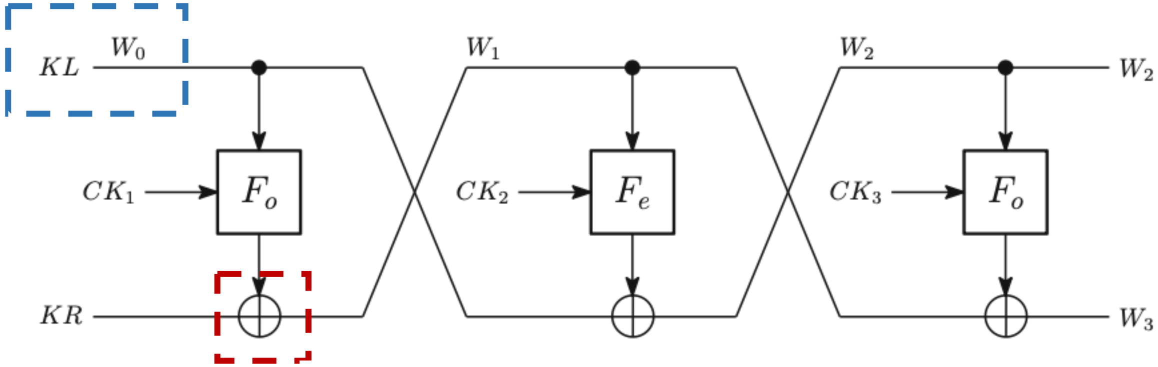


- Total number of Toffoli gates $= 7,168$
- Total number of CNOT gates $= 128 + 9,112 + 768 = 10,008$
- Total number of Pauli-X gates $= 64$.

KeyWords (W_i)	# Pauli-X	# CNOT	# Toffoli
W_0	0	128	0
W_1	$64 + 65 = 129$	$10,008 + 128 = 10,136$	7,168
W_2	$64 + 65 = 129$	$10,008 + 128 = 10,136$	7,168
W_3	$64 + 57 = 121$	$10,008 + 128 = 10,136$	7,168
Total	379	30,536	21,504
Round Subkeys	# Pauli-X	# CNOT	# Toffoli
RK_i for each i	0	$128 \times 4 = 512$	0

Key Schedule

- Key Schedule - Yang.et.al
  - Key initialization
    - CNOT gate를 X gate로 대체

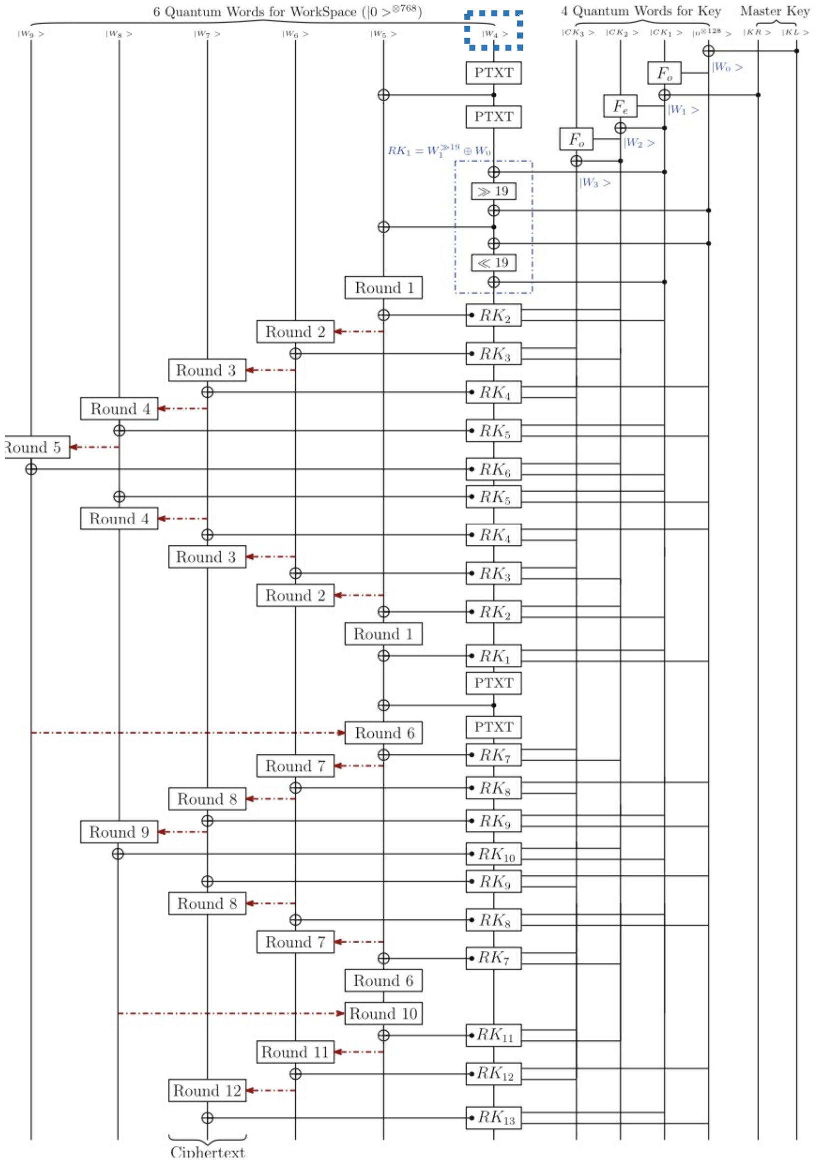


KeyWords (W_i)	# Pauli-X	# CNOT	# Toffoli
W_0	0	$\cancel{128}$	0
W_1	$64 + 65 = 129$	$10,008 + \cancel{128} = 10,136$	7,168
W_2	$64 + 65 = 129$	$10,008 + 128 = 10,136$	7,168
W_3	$64 + 57 = 121$	$10,008 + 128 = 10,136$	7,168
Total	379	30,536	21,504
Round Subkeys	# Pauli-X	# CNOT	# Toffoli
RK_i for each i	0	$128 \times 4 = 512$	0

Key Schedule

- Key Schedule - Chauhan.et.al
  - Key generation
    - RK_i 하나의 큐비트 세트(w_4)만 사용
      → 라운드 연산 후 역연산을 통해 초기화
    - ‘zig-zag’ 방식 사용 → 큐비트 최적화

KeyWords (W_i)	# Pauli-X	# CNOT	# Toffoli
W_0	0	128	0
W_1	$64 + 65 = 129$	$10,008 + 128 = 10,136$	7,168
W_2	$64 + 65 = 129$	$10,008 + 128 = 10,136$	7,168
W_3	$64 + 57 = 121$	$10,008 + 128 = 10,136$	7,168
Total	379	30,536	21,504
Round Subkeys	# Pauli-X	# CNOT	# Toffoli
RK_i for each i	0	$128 \times 4 = 512$	0



Key Schedule

- Key Schedule - Yang.et.al
  - Key generation
    - RK_i 하나의 큐비트 세트(w_4)만 사용

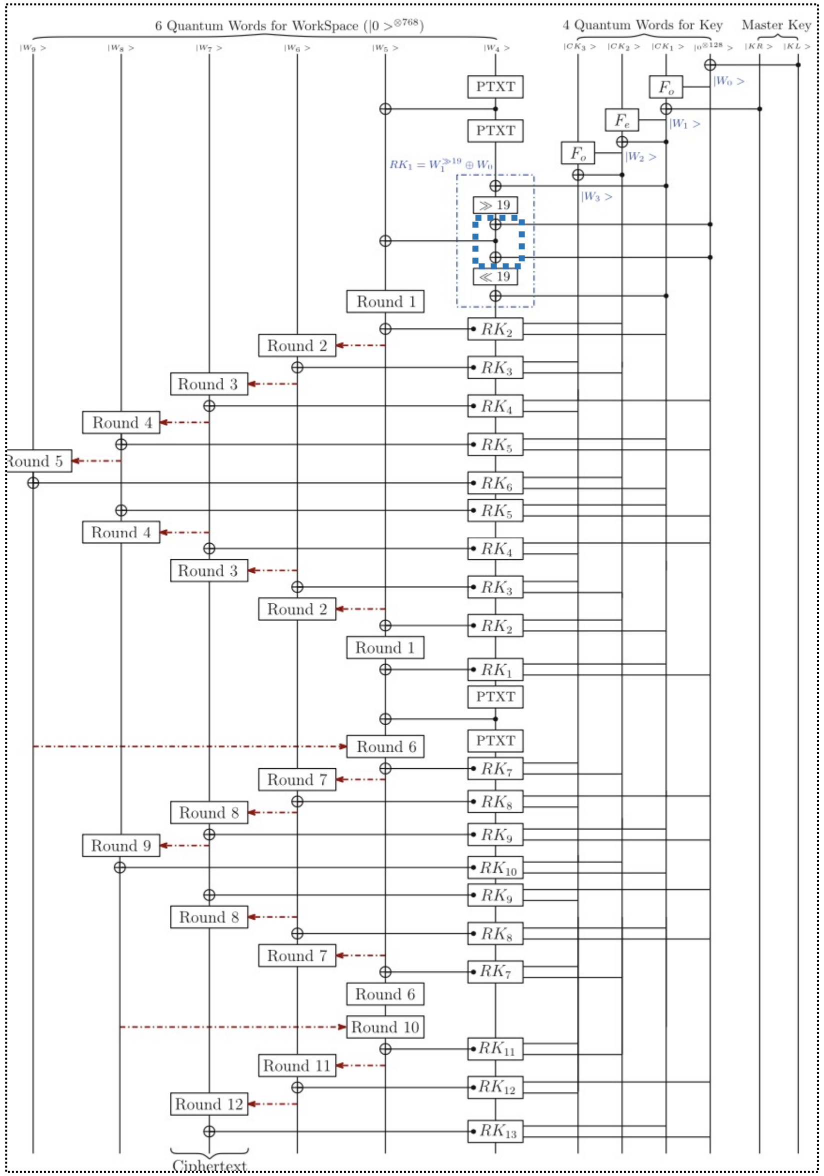
      → 라운드 연산 후 역연산을 통해 초기화
    - RK_i 생성 시 w_0 을 사용하는 경우 X gate로 대체

      → CNOT gate 256개 감소
    - ‘pipeline’ 방식 사용 → depth 최적화

$$ek_1 = (W_0) \oplus (W_1 \ggg 19),\quad ek_2 = (W_1) \oplus (W_2 \ggg 19)$$
$$ek_3 = (W_2) \oplus (W_3 \ggg 19),\quad ek_4 = (W_0 \ggg 19) \oplus (W_3)$$
$$ek_5 = (W_0) \oplus (W_1 \ggg 31),\quad ek_6 = (W_1) \oplus (W_2 \ggg 31)$$
$$ek_7 = (W_2) \oplus (W_3 \ggg 31),\quad ek_8 = (W_0 \ggg 31) \oplus (W_3)$$
$$ek_9 = (W_0) \oplus (W_1 \lll 61),\quad ek_{10} = (W_1) \oplus (W_2 \lll 61)$$
$$ek_{11} = (W_2) \oplus (W_3 \lll 61),\quad ek_{12} = (W_0 \lll 61) \oplus (W_3)$$
$$ek_{13} = (W_0) \oplus (W_1 \lll 31),\quad ek_{14} = (W_1) \oplus (W_2 \lll 31)$$
$$ek_{15} = (W_2) \oplus (W_3 \lll 31),\quad ek_{16} = (W_0 \lll 31) \oplus (W_3)$$
$$ek_{17} = (W_0) \oplus (W_1 \lll 19)$$

KeyWords	# Toffoli
W_0	0
W_1	7,168
W_2	7,168
W_3	7,168
Total	21,504

Round Subkeys	# Pauli-X	# CNOT	# Toffoli
RK_i for each i	0	$128 \times 4 = 512$	0



Quantum resource estimation

- ARIA 양자 자원 추정

Table 2: Required quantum resources for ARIA quantum circuit implementation

Cipher	Source	#X	#CNOT	#Toffoli	Toffoli depth	#Qubit	Depth	TD-M cost
ARIA-128	CS[2]	1,595	231,124	157,696	4,312	1,560	9,260	6,726,720
	This work	1,408	285,784	25,920	60	29,216	3,500	1,752,960
ARIA-192	CS[2]	1,851	273,264	183,368	5,096	1,560	10,948	7,949,760
	This work	1,624	324,136	29,376	68	32,928	3,978	2,239,104
ARIA-256	CS[2]	2,171	325,352	222,208	6,076	1,688	13,054	10,256,288
	This work	1,856	362,488	32,832	76	36,640	4,455	2,784,640

Table 3: Required decomposed quantum resources for ARIA quantum circuit implementation

Variant		#Cliford	$\#T$	T-depth	#Qubit	Full depth
ARIA-128	CS[2]$^{\diamond}$	1,494,287	1,103,872	17,248	1,560	37,882
	This work	494,552	181,440	240	29,216	4,650
ARIA-192	CS[2]$^{\diamond}$	1,742,059	1,283,576	20,376	1,560	44,774
	This work	560,768	205,632	272	32,928	5,285
ARIA-256	CS[2]$^{\diamond}$	2,105,187	1,555,456	24,304	1,688	51,666
	This work	627,000	229,824	304	36,640	5,919

$\diamond$ Extrapolated result

Grover's key search

- ARIA Grover 공격 비용 추정
  - Grover 공격 최적 iteration $[\frac{\pi}{4}\sqrt{2^k}]$
  - Oracle에는 2개의 회로 필요 → 2 x $[\frac{\pi}{4}\sqrt{2^k}]$ x quantum resources
  - $r = [key\ size/\ block\ size]$개의 평문-암호문 쌍을 얻는 것이 고유한 키를 식별할 수 있음.

    → Grover 공격 비용 : 2 x r x $[\frac{\pi}{4}\sqrt{2^k}]$ x quantum resource
  - **ARIA 는 NIST Level 1, 3, 5를 달성**

Table 4: Cost of the Grover's key search for ARIA

Cipher	Source	Total gates	Full depth	Cost (complexity)	#Qubit	FD-M cost
ARIA-128	CS[2]	$1.998 \cdot 2^{85}$	$1.816 \cdot 2^{79}$	$1.814 \cdot 2^{165}$	1,561	$1.26 \cdot 2^{86}$
	This work	$1.117 \cdot 2^{84}$	$1.783 \cdot 2^{76}$	$1.991 \cdot 2^{160}$	29,217	$1.313 \cdot 2^{84}$
ARIA-192	CS[2]	$1.146 \cdot 2^{119}$	$1.073 \cdot 2^{112}$	$1.23 \cdot 2^{231}$	3,121	$1.489 \cdot 2^{118}$
	This work	$1.2 \cdot 2^{117}$	$1.013 \cdot 2^{109}$	$1.216 \cdot 2^{226}$	65,857	$1.677 \cdot 2^{116}$
ARIA-256	CS[2]	$1.384 \cdot 2^{151}$	$1.238 \cdot 2^{144}$	$1.714 \cdot 2^{295}$	3,377	$1.921 \cdot 2^{150}$
	This work	$1.336 \cdot 2^{149}$	$1.135 \cdot 2^{141}$	$1.516 \cdot 2^{290}$	72,081	$1.043 \cdot 2^{149}$

Q & A