Towards Optimizing Quantum Implementation of AES S-box

Doyoung Chung[1,2], Jooyoung Lee[2], Seungkwang Lee[1], and Dooho Choi[1,*]

[1] School of Computing, KAIST, Daejeon, 34141, Korea
{wordspqr, hicalf}@kaist.ac.kr
[2] Information Security Research Division, ETRI, Daejeon, 34129, Korea
{dhchoi, thisisdoyoung, skwang}@etri.re.kr

Abstract. Grover's search algorithm allows a quantum adversary to find a k-bit secret key of a block cipher by making $\mathrm{O}(2^{k/2})$ block cipher queries. Resistance of a block cipher to such an attack is evaluated by quantum resources required to implement Grover's oracle for the target cipher. The quantum resources are typically estimated by the T-depth of its circuit implementation (time) and the number of qubits used by the circuit (space).

Since the AES S-box is the only component which requires T-gates in the quantum implementation of AES, recent research has put its focus on efficient implementation of the AES S-box. However, any efficient implementation with low T-depth will not be practical in the real world without considering qubit consumption of the implementation.

In this work, we propose seven methods of trade-off between time and space for the quantum implementation of the AES S-box. In particular, one of our methods turns out to use the smallest number of qubits among the existing methods, significantly reducing its T-depth.

Keywords: Quantum implementation, quantum cryptanalysis, Grover's algorithm, AES, multiplicative inversion.

1 Introduction

Most cryptographic primitives are under new threats with the advent of quantum computers. Public key cryptosystems such as RSA, ECDSA, and ECDH will be completely broken by Shor's algorithm [15], a quantum algorithm that solves the order finding problem in polynomial time. When it comes to symmetric key cryptography, exhaustive key search using Grover's algorithm [8] is becoming a new threat. For example, Grover's search algorithm allows a quantum adversary to find a k-bit secret key of a block cipher by making $\mathrm{O}(2^{k/2})$ block cipher queries. Resistance of a block cipher to such an attack is evaluated by quantum resources required to implement Grover's oracle for the target cipher. The quantum resources are typically estimated by the circuit depth of the circuit implementation (time) and the number of qubits used by the circuit (space) [7, 9, 12].

Quantum circuits involve error-prone qubits, and fault-tolerant quantum computation (FTQC) is made possible by using error correcting codes. The surface code is one of the most feasible candidates for this purpose. Since T-gates are exceptionally expensive in the implementation of the surface code, *T-depth*, counting the number of sequential T-gates, dominates the overall efficiency of the quantum circuit in terms of the processing time [2]. For this reason, T-depth is widely used as a metric to estimate the time complexity of a quantum circuit.

The only component of AES that requires T-gates for its quantum implementation is the multiplicative inversion used in the AES S-box. Therefore, recent research [7, 9, 11, 12] has put its main focus on lightweight implementation of the multiplicative inversion, using tower field constructions of the underlying finite field $\mathbf{GF}(2^8)$. However, any efficient implementation with low T-depth will not be practical in the real world without considering its qubit consumption since qubits are arguably considered as the most valuable resources in quantum computation.

A classical implementation of the AES S-box based on a tower-field construction of $\mathbf{GF}(2^8)$ consists of XOR and AND gates. An XOR gate in a classical circuit can be converted to a CNOT gate in the corresponding quantum circuit, while an AND gate is converted to a Toffoli gate or a quantum AND gate. Since both gates are built on T-gates, the T-depth of the quantum circuit is determined by the AND-depth of the classical circuit.

1.1 Our Contribution

In this work, we propose seven methods of trade-off between time and space for the quantum implementation of the AES S-box. In particular, one of our methods turns out to use the smallest number of qubits among the existing methods, significantly reducing its T-depth. Precisely, it uses 32 qubits in a quantum circuit of T-depth 36. We note that the implementation by Langenberg et al. [12] uses the same number of qubits, while its T-depth is 120.

We also propose two methods balancing depth and width in the quantum circuit. In particular, those methods improve on the “balanced” implementation proposed by Jaques et al. [9] in terms of both depth and width.

The key idea behind our implementations is to adopt efficient tower-field constructions studied in [10] to reduce the AND-depth of multiplicative inversion over $\mathbf{GF}(2^8)$ from 6 (as proposed by Boyar et al. [3]) to 4 in a classical implementation. In order to further optimize the T-depth of the corresponding quantum implementation, we decomposed the 8-bit inversion into three 4-bit multiplications, one 4-bit inversion and the remaining minor operations, and to each subfield operation, applied a different quantum implementation, carefully recycling ancilla qubits, and hence reducing the overall time-space complexity of the resulting circuit. The time-space complexity of our methods is summarized in Table 1.

scheme	type	width	T-depth
Grassl et al. [7]		44	217
Langenberg et al. [12]		32	120
Jaques et al. [9]	balanced	41	35
	minimum depth	137	6
our schemes	minimum width	32	36
	balanced 1	34	31
	balanced 2	36	30
	minimum depth	54	20

Table 1: Time-space complexity of the representative quantum circuits.

1.2 Related Work

The first implementation of AES quantum circuits was proposed by Grassl *et al.* in 2016 [7]. In particular, they analyzed that every operation except S-box could be implemented by using only Clifford gates, and T-depth of the multiplicative inversion in AES S-box was evaluated. Kim *et al.* improved Grassl *et al*'s work in such a way to reduce T-depth of the multiplicative inversion [11].

Afterwards, the efficient implementations of AES quantum circuits have been studied strongly based on lightweight implementations of AES in classical computing environment, therein S-boxes are implemented using XOR and AND gates. When converting these into a quantum circuit, an XOR gate is converted to a CNOT gate without a significant performance penalty. An AND gate is, on the other hand, converted into either a Toffoli gate or a quantum AND gate which uses T-gates, imposing a performance penalty of the execution time. Consequently, reducing the AND-depth in quantum circuits is the most important factor in increasing efficiency. Based on hardware lightweight implementations of AES with a low depth of AND gates [3, 4], Langenberg *et al.* proposed an AES quantum circuit of significantly reduced T-depth [12], compared to previous work.

A Toffoli gate, also known as a controlled-controlled-NOT (CCNOT) gate from functional point of view, is used to convert an AND gate into quantum circuits. So far, the most shallow implementation of a Toffoli gate was known to have T-depth 3 [1]. In [9], an AND gate in the classical computing environment was converted to a quantum AND gate instead of a Toffoli gate. While a Toffoli gate has T-depth 3 and is symmetrical to the dagger operation, a quantum AND gate has T-depth 1, and the dagger operation has an asymmetrical T-depth 0 [6, 9].

By taking advantage of T-depth 1, Jaques *et al.* reduced T-depth of Langenberg *et al.*'s implementation. To be specific, they proposed two quantum circuits of two different cost advantages. The first one was based on [3] in order to reduce T-depth with relatively balanced time and space. The other one was based on [4] for a low depth circuit without a space restriction. Because qubits are still more expensive quantum resource, the time-space balanced circuits would be beneficial to analyze the strength of cryptography in quantitative measurement.

Here, we note that it has been the multiplicative inversion in AES S-box that makes a difference in resource requirements including T-depth and the number of qubits. This motivated us to adopt improved tower-field construction for a low-cost quantum circuit of multiplicative inversion in AES S-box. The following explains various techniques to reduce the time-space complexity of a quantum circuit.

2 Preliminaries

2.1 S-box of AES

An AES S-box is a nonlinear confusion layer for mapping an 8-bit input to an 8-bit output interpreted as polynomials over $\mathbf{GF}(2)$. The input is mapped to its multiplicative inverse in $\mathbf{GF}(2^8) = \mathbf{GF}(2)[x]/(x^8 + x^4 + x^3 + x + 1)$ and is then followed by an affine transformation. Zero, as the identity, is mapped to itself. This layer can be defined in the matrix form as follow:

$$\begin{bmatrix} s_0 \\ s_1 \\ s_2 \\ s_3 \\ s_4 \\ s_5 \\ s_6 \\ s_7 \end{bmatrix} = \begin{bmatrix} 1\,0\,0\,0\,1\,1\,1\,1 \\ 1\,1\,0\,0\,0\,1\,1\,1 \\ 1\,1\,1\,0\,0\,0\,1\,1 \\ 1\,1\,1\,1\,0\,0\,0\,1 \\ 1\,1\,1\,1\,1\,0\,0\,0 \\ 0\,1\,1\,1\,1\,1\,0\,0 \\ 0\,0\,1\,1\,1\,1\,1\,0 \\ 0\,0\,0\,1\,1\,1\,1\,1 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

where $\begin{bmatrix} s_7, ..., s_0 \end{bmatrix}$ is the output of S-box and $\begin{bmatrix} b_7, ..., b_0 \end{bmatrix}$ is the input of S-box as a vector.

In classical computing environment, an 8-bit to 8-bit lookup table is generally used for S-box in most of software implementation of AES. However, in quantum computing environment, it is efficient to perform the multiplicative inversion and the affine transformation due to the limited number of qubits. The affine transformation which follows the multiplicative inversion can be computed in-place with only X- and CNOT gates. So, it does not consume additional T-depth and qubits.

For this reason, the main issue in implementing AES S-box quantum circuit is how to implement multiplicative inversion efficiently. The following explains a technique of tower-field construction to perform this operation efficiently.

2.2 Tower-field construction

A tower of fields is an extension sequence of some fields, $\mathbb{F}$. The tower-field construction for the implementation of the AES S-box is representing the operations over $\mathbb{F}_{2^{2k}}$ with operations over $\mathbb{F}_{2^k}$ recursively. The computational cost of AES operations that are performed on $\mathbf{GF}(2^8)$ can be reduced by using isomorphic composite fields which generated by the tower-field construction. When using subfield arithmetic, it is costly to convert the original into the isomorphic composite field and vice-versa. Such conversion and re-conversion can be implemented with only CNOT gates in quantum circuits by using PLU decomposition. One of the known tower-field representations is as follows [3]:

$\mathbf{GF}(2^2)$ by adjoining a root W of x^2+x+1 over $\mathbf{GF}(2)$
$\mathbf{GF}(2^4)$ by adjoining a root Z of x^2+x+W^2 over $\mathbf{GF}(2^2)$
$\mathbf{GF}(2^8)$ by adjoining a root Y of x^2+x+WZ over $\mathbf{GF}(2^4)$.

In this paper, we will present our representation of tower-field construction that is improved for lightweight quantum circuits of the multiplicative inversion in AES S-box. This tower-field construction reduces AND-depth, imposing a dominant effect on the execution time of a quantum circuit.

2.3 Grover's algorithm

For a Boolean function $f:\{0,1\}^k \longmapsto \{0,1\}$, Grover's algorithm takes a *Grover's Oracle*, U_f, that implements $|x\rangle|y\rangle \rightarrow |x\rangle|y \oplus f(x)\rangle$, where $x \in \{0,1\}^k$ and $y \in \{0,1\}$. Basically, Grover's algorithms finds an element x_0 such that $f(x_0) = 1$ by repeatedly applying a *Grover iteration* defined below to the initial state $|\psi\rangle = H^{\otimes k}|0\rangle$.

$$Q = -H^{\otimes k}(I - 2|0\rangle\langle 0|)H^{\otimes k}U_f$$

After applying $\lfloor \frac{\pi}{4}\sqrt{\frac{K}{N}} \rfloor$ iterations on the initial state, a solution x_0 will be found with at least $1-(N/K)$ probability by measuring the entire quantum register, where K is the total number of candidates ($K = 2^k$), and N is the number of solutions ($N = |\{x : f(x) = 1\}|$).

In [5], the authors analyzed that $r_k = \lceil k/128 \rceil$ known plaintext-ciphertext pairs are sufficient to avoid false positives in an exhaustive key search for AES-k, where $k \in \{128, 192, 256\}$. In order to build U_f, each plaintext-ciphertext pair requires AES and its inverse. This gives us that the number of AES instances should be twice as many as the number of plaintext-ciphertext pairs. For each key size k, the number of Grover's operations is then given by

- 2 AES instances for $k = 128$
- 4 AES instances for $k = 192$
- 4 AES instances for $k = 256$.

2.4 Quantum AND gate

In [9], the authors used a T-depth 1 circuit for an AND gate which is a combination of Selinger [14], and Gidney [6], and that was designed by Mathias Soeken.

This gate requires one more ancilla qubit, instead of reducing T-depth compared to Toffoli gate. It has an assymetric relationship with its dagger gate. Its dagger gate requires only 3 qubits same as Toffoli gate, but does not include any T-depth.

The diagrams for the quantum AND gate and AND† gate are depicted in Fig.1 and Fig.2.

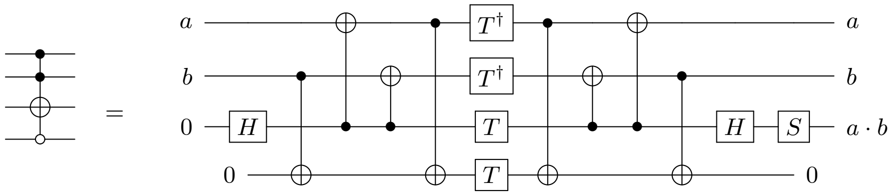


Fig. 1: Quantum AND gate.

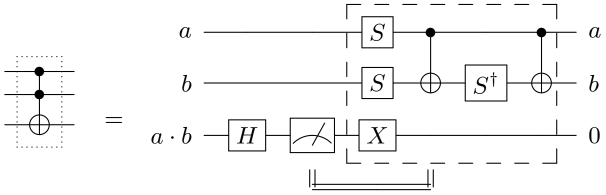


Fig. 2: Quantum AND† gate.

3 Improvement on tower-field construction

In this section, we improve the tower-field construction by using the following representations of Galois fields:

$$\begin{cases} \mathbf{GF}(2^8): & x^8+x^4+x^3+x+1 \\ \mathbf{GF}(2^4): & x^4+x+1 \\ \mathbf{GF}(2^2): & x^2+x+1, \end{cases}$$

and we suggest the following composite fields $\mathbf{GF}((2^4)^2)$ and $\mathbf{GF}((2^2)^2)$:

$$\begin{cases} \mathbf{GF}((2^4)^2): & y^2+y+\lambda \\ \mathbf{GF}((2^2)^2): & z^2+z+\phi, \end{cases}$$

where

- $\lambda \coloneqq \omega^{11} = \omega^3 + \omega^2 + \omega = \{1110\}_2$
- ω is a root of $x^4 + x + 1$
- $\phi = \{10\}_2$
- ϕ is a root of $x^2 + x + 1$.

3.1 Isomorphic mapping quantum circuit

The $\mathbf{GF}(2^8)$ of AES and other representations of $\mathbf{GF}(2^8)$ are isomorphic. The matrix of mapping between $\mathbf{GF}(2^8)$ of AES and our representation of $\mathbf{GF}(2^8) = \mathbf{GF}((2^4)^2)$, and its inverse mapping are defined as M and M^{-1}:

$$M = \begin{bmatrix} 1&0&1&0&0&0&0&0 \\ 1&0&1&0&1&1&0&0 \\ 1&1&0&1&0&0&1&0 \\ 0&1&1&1&0&0&0&0 \\ 0&0&0&1&0&1&0&0 \\ 1&0&0&0&0&0&1&0 \\ 0&0&0&0&0&1&1&0 \\ 0&1&1&1&0&0&0&1 \end{bmatrix}, \text{ and } M^{-1} = \begin{bmatrix} 1&0&1&1&0&1&0&0 \\ 1&0&0&1&1&1&1&0 \\ 0&0&1&1&0&1&0&0 \\ 1&0&1&1&1&0&1&0 \\ 0&1&1&1&0&0&1&0 \\ 1&0&1&1&0&0&1&0 \\ 1&0&1&1&0&0&0&0 \\ 0&0&0&1&0&0&0&1 \end{bmatrix}.$$

Next, the matrix of mapping between the proposed $\mathbf{GF}(2^4)$ and $\mathbf{GF}((2^2)^2)$ representations, and its inverse mapping are given by

$$M_4 = \begin{bmatrix} 1&0&0&0 \\ 1&1&1&0 \\ 1&1&0&0 \\ 0&0&0&1 \end{bmatrix}, \text{ and } M_4^{-1} = \begin{bmatrix} 1&0&0&0 \\ 1&0&1&0 \\ 0&1&1&0 \\ 0&0&0&1 \end{bmatrix}.$$

The matrix α of the isomorphic mapping from the input qubits for AES $\mathbf{GF}(2^8)$ to our composite field is then obtained by the multiplication of the matrices above:

$$\alpha = \begin{bmatrix} M_4 & 0 \\ 0 & M_4 \end{bmatrix} \circ M = \begin{bmatrix} 1&0&1&0&0&0&0&0 \\ 1&1&0&1&1&1&1&0 \\ 0&0&0&0&1&1&0&0 \\ 0&1&1&1&0&0&0&0 \\ 0&0&0&1&0&1&0&0 \\ 1&0&0&1&0&0&0&0 \\ 1&0&0&1&0&1&1&0 \\ 0&1&1&1&0&0&0&1 \end{bmatrix}.$$

Here, α can be implemented with only CNOT gates in a quantum circuit by using PLU decomposition. For example, $\alpha \coloneqq P \cdot L \cdot U$, where

$$\mathrm{P} \coloneqq \begin{bmatrix} 1&0&0&0&0&0&0&0\\ 0&1&0&0&0&0&0&0\\ 0&0&0&0&0&0&1&0\\ 0&0&0&0&1&0&0&0\\ 0&0&0&1&0&0&0&0\\ 0&0&1&0&0&0&0&0\\ 0&0&0&0&0&1&0&0\\ 0&0&0&0&0&0&0&1 \end{bmatrix}, \mathrm{L} \coloneqq \begin{bmatrix} 1&0&0&0&0&0&0&0\\ 1&1&0&0&0&0&0&0\\ 1&0&1&0&0&0&0&0\\ 0&0&0&1&0&0&0&0\\ 0&1&0&0&1&0&0&0\\ 1&0&1&0&0&1&0&0\\ 0&0&0&0&1&0&1&0\\ 0&1&0&0&1&0&0&1 \end{bmatrix}, \mathrm{U} \coloneqq \begin{bmatrix} 1&0&1&0&0&0&0&0\\ 0&1&1&1&1&1&1&0\\ 0&0&1&1&0&0&0&0\\ 0&0&0&1&0&1&0&0\\ 0&0&0&0&1&1&1&0\\ 0&0&0&0&0&1&1&0\\ 0&0&0&0&0&0&1&0\\ 0&0&0&0&0&0&0&1 \end{bmatrix}.$$

Based on the PLU decomposition above, the quantum circuit of the isomorphic mapping, $\alpha \coloneqq (a_7, a_6, a_5, a_4, a_3, a_2, a_1, a_0) \mapsto ((b_{hh1}, b_{hh0}, b_{hl1}, b_{hl0}), (b_{lh1}, b_{lh0}, b_{ll1}, b_{ll0}))$ is depicted as Fig. 3.

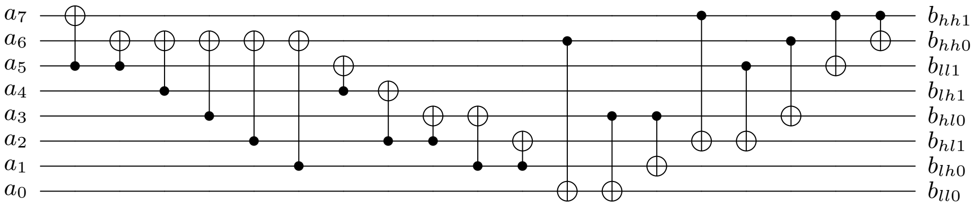


Fig. 3: Quantum circuit of the isomorphic mapping α.

3.2 Inversion method with our composite field

The multiplicative inversion for composite fields $\mathbf{GF}((2^m)^n)$ can be computed as a combination of operations in the sub-fields $\mathbf{GF}(2^m)$ [13]. Throughout this paper, the multiplicative inversion, P^{-1}, is represented as $P^{-1} = (P^{17})^{-1} \cdot P^{16}$ for $n = 2$ and $m = 4$. Then P^{-1} can be calculated by the following four steps:

1. $P^{r-1} = P^{16}$
2. $P^r = (P^{16}) \cdot P$
3. Compute $(P^r)^{-1}$ in $\mathbf{GF}(2^4)$
4. Compute $(P^r)^{-1} \cdot P^{r-1}$ using $\mathbf{GF}((2^2)^2)$ arithmetic

Step 1. First, we need to compute P^{16}.

For $P = ((b_{hh1}, b_{hh0}, b_{hl1}, b_{hl0}), (b_{lh1}, b_{lh0}, b_{ll1}, b_{ll0})) = b_h y + b_l$,

$P^{16} = (b_h y + b_l)^{16} = b_h y^{16} + b_l$
$y^{16} = y + 1$

Thus, $P^{16} = b_h y + (b_h + b_l)$.

Step 2. Then we need to compute $P^r = P^{16} {\cdot} P$. In here, $\lambda = map_4(\lambda) = \{1100\}_2$ in $\mathbf{GF}((2^2)^2)$, $z^2 + z + \phi, \phi = \{10\}_2$

$$\begin{aligned} P = P^{16} \cdot P &= (b_h y + (b_h + b_l))(b_h y + b_l) \\ &= b_h^2 y^2 + b_h^2 y + (b_h + b_l)b_l \\ &= b_h^2 (y + \lambda) + b_h^2 y + (b_h + b_l)b_l \\ &= b_h^2 \times \lambda + (b_h + b_l)b_l. \end{aligned}$$

Squaring, multiplication by λ, and multiplication in $\mathbf{GF}((2^2)^2)$ in our composite field should be implemented in quantum circuit to calculate above equations. The squaring can be calculated as

$$\begin{aligned} (p_h z + p_l)^2 &= (p_{h1}x + p_{h0})^2 z^2 + (p_{l1}x + p_{l0})^2 \\ &= (p_{h1}x^2 + p_{h0})z^2 + (p_{l1}x^2 + p_{l0}) \\ &\coloneqq (q_{h1} + q_{h0})z + (q_{l1} + q_{l0}). \end{aligned}$$

The quantum circuit for the above formula, squaring, is shown as Fig. 4.

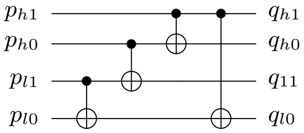


Fig. 4: Quantum circuit of the squaring in $\mathbf{GF}((2^2)^2)$.

The multiplication by λ can be calculated as

$$\begin{aligned} (p_h z + p_l) \times \lambda &= ((p_{h0} + p_{l0})x + p_{h1} + p_{h0} + p_{l1} + p_{l0})z + p_{h1}x + p_{h0} \\ &\coloneqq (q_{h1} + q_{h0})z + (q_{l1} + q_{l0}). \end{aligned}$$

The quantum circuit for the above formula, multiplication by λ, is depicted as Fig. 5. Both quantum circuits for squaring and multiplication by λ are implemented using only CNOT gates and wiring. The rest of arithmetic operations in $\mathbf{GF}((2^2)^2)$ is multiplication which can be calculated as

$$\begin{aligned} (p_h z + p_l)(q_h z + q_l) &= ((p_h + p_l)(q_h + q_l) + p_l q_l)z + (p_h q_h \phi + p_l q_l) \\ &\coloneqq r_h z + r_l. \end{aligned}$$

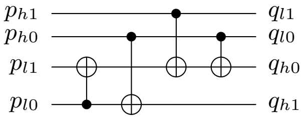


Fig. 5: Quantum circuit of the multiplication by λ in $\mathbf{GF}((2^2)^2)$.

To minimize the time-space complexity of the quantum circuit of multiplicative inversion in $\mathbf{GF}(2^8)$, we adopt various implementation of quantum circuit for multiplication in $\mathbf{GF}((2^2)^2)$. We present those quantum circuits in Section 4. To clarify the concept of the multiplication in $\mathbf{GF}((2^2)^2)$, we present the classical circuit in Fig. 6.

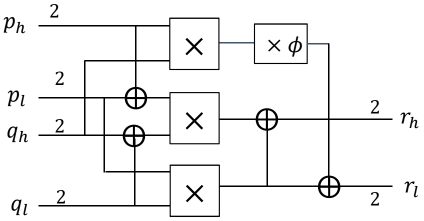


Fig. 6: Classical circuit of the multiplication in $\mathbf{GF}((2^2)^2)$.

Step 3. We compute $(P^{16})^{-1}$ in $\mathbf{GF}((2^2)^2)$, $z^2 + z + \phi$, $\phi = \{00\}_2$.

Given $p_h, p_l, q_h, q_l \in \mathbf{GF}(2^2) = \mathbf{GF}(2)[x]/(x^2+x+1)$ and $(p_h z+p_l), (q_h z+q_l) \in \mathbf{GF}((2^2)^2)$, suppose that $(p_h z + p_l)^{-1} = (q_h z + q_l)$. Then, we have

$$\begin{aligned}(p_h z + p_l)(q_h z + q_l) &= ((p_h + p_l)(q_h + q_l) + p_l q_l)z + (p_h q_h \phi + p_l q_l)\\ &= 1,\end{aligned}$$

and this gives us $(p_h + p_l)q_h + p_h q_l = 0$, and $p_h \phi q_h + p_l q_l = 1$. Thus, it is easy to know that $q_h = p_h d^{-1}$ and $q_l = (p_h + p_l)d^{-1}$, where $d = p_h^2 \phi + p_l(p_h + p_l)$. The classical circuit diagram for the multiplicative inversion on $\mathbf{GF}(2^2)^2)$ designed on the above formula is depicted in Fig. 7.

Step 4. Finally, we compute $(P^r)^{-1} \cdot P^{16}$ in $\mathbf{GF}((2^2)^2)$, $z^2 + z + \phi$, $\phi = \{00\}_2$.

For $p = (p_h z + p_l)$ in $\mathbf{GF}((2^2)^2)$, where $p_h, p_l \in \mathbf{GF}(2^2)$, $x^2 + x + 1$,

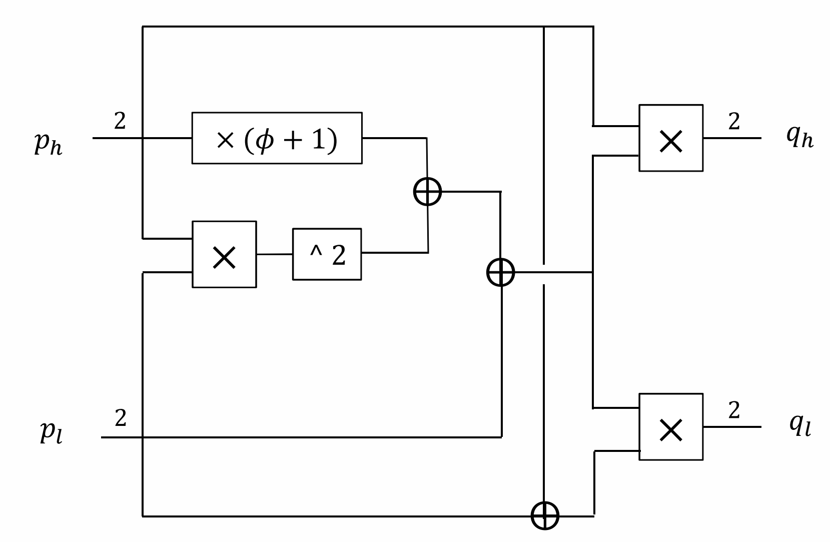


Fig. 7: Circuit diagram of the multiplication inversion in $\mathbf{GF}((2^2)^2)$.

$$\begin{aligned} p \cdot P^{16} &= p \cdot (b_h y + (b_h + b_l)), \\ &= p \cdot b_h y + p \cdot (b_h + b_l), \end{aligned}$$

because $P^{16} = b_h y + (b_h + b_l)$ in Step 1.

Following Itoh and Tsujii's inversion algorithm, $p \cdot P^{16}$ is the multiplicative inversion for the composite field, $\mathbf{GF}(2^8)$, which we suggested. The classical circuit diagram for calculating $p \cdot P^{16}$ is illustrated in Fig. 8. The square box written $L\times$ means a multiplication operation in $\mathbf{GF}(2^4)$, and the others x^2, x^{-1}, and $\times\lambda$ represent squaring, multiplicative inversion, and multiplication by λ in $\mathbf{GF}(2^4)$, respectively.

Summary. A hardware implementation of Galois field operations generally uses AND and XOR (some NOR) gates in classical computing. Among them, an XOR gate can be converted into a quantum gate as a relatively inexpensive CNOT gate. In the case of AND, it is converted to a Toffoli gate or a quantum AND gate [9]. Currently, a Toffoli gate requires T-depth 3 [1] and a quantum AND gate requires T-depth 1 [9]. Therefore, the AND-depth in a classical computing environment decides the T-depth in a quantum computing environment, and a hardware implementation with a low AND-depth will be preferred to reduce time complexity when converting to a quantum circuit.

Compared to prior work [3], our method reduces the AND-depth of the multiplicative inversion from 6 to 4, which is about a 33% decrease. Note that

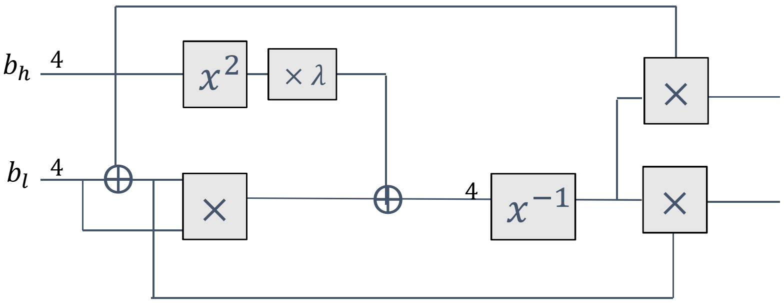


Fig. 8: Circuit diagram of the multiplication inversion in $\mathbf{GF}(2^8)$.

we only modified the multiplicative inversion; the rest part for constructing the S-box, including the isomorphic mapping, its inverse operation, and the affine mapping, can be implemented with only XOR gates. Therefore, the AND-depth of the S-box using the proposed multiplicative inversion remains the same, 4.

4 Proposed Quantum Circuit

In order to build up a shallow implementation of multiplicative inversion in $\mathbf{GF}(2^8)$, we should take into account time-space trade-offs on the corresponding quantum circuits of $\mathbf{GF}(2^4)$ and $\mathbf{GF}(2^2)$. From now on, we propose several low-cost quantum circuits of multiplication and its inversion in $\mathbf{GF}(2^8)$ with a bottom-up approach. Meanwhile, we mainly focus on the optimal combinations of quantum circuits for composite Galois field operations under consideration of time-space trade-offs. In addition, we provide some implementation techniques to minimize time-space complexity.

4.1 GF(2^2) arithmetic quantum circuits

The operations in $\mathbf{GF}(2^2)$ are the basic operation in our scheme. We utilize multiplication and its inverse (dagger) operation. For $(a_1x+a_0),(b_1x+b_0) \in \mathbf{GF}(2^2), x^2+x+1$, the multiplication operation can be written as

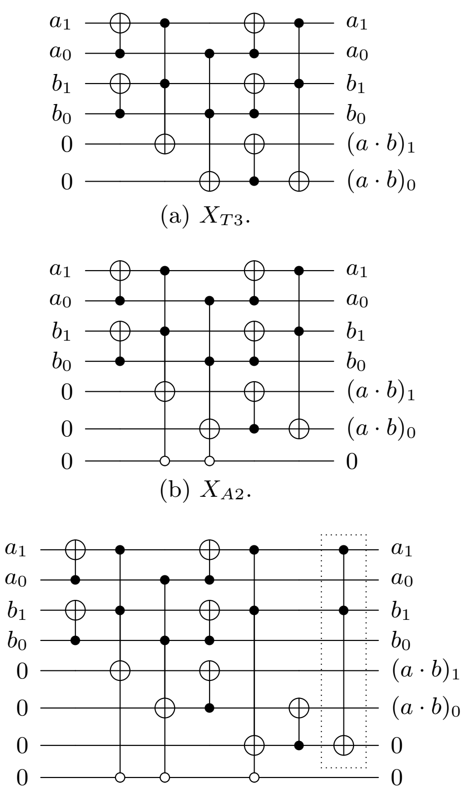
$$(a_1x+a_0)(b_1x+b_0) = ((a_1+a_0)(b_1+b_0)+a_0b_0)x + a_1b_1 + a_0b_0.$$

This equation can be implemented as a quantum circuit using CNOT and Toffoli gates. A Toffoli gate performs CCNOT for either $|0\rangle$ or $|1\rangle$ to the target register. In contrary, an AND gate does CCNOT when the input state $|0\rangle$ is given to the target register. Although an AND gate uses one more qubit temporarily, this reduces T-depth compared to Toffoli gate [9]. By using these characteristics of Toffoli and AND gates, quantum circuits for multiplication in $\mathbf{GF}(2^2)$ of various time-space complexities can be designed.

We propose three types of quantum circuits for multiplication in $\mathbf{GF}(2^2)$ and two types of quantum circuits for dagger operation. The three types of quantum circuits for multiplication are summarized in Table 2, and are illustrated in Fig. 9. In addition, the two types of quantum circuits for multiplication dagger are summarized in Table 3, and are described in Fig. 10.

Notation	Description
X_{T3}	Toffoli × 3
X_{A2}	AND × 2, Toffoli × 1
X_{A3}	AND × 3, AND† × 1

Table 2: Three quantum circuits of multiplication in $\mathbf{GF}(2^2)$.



(a) X_{T3}.

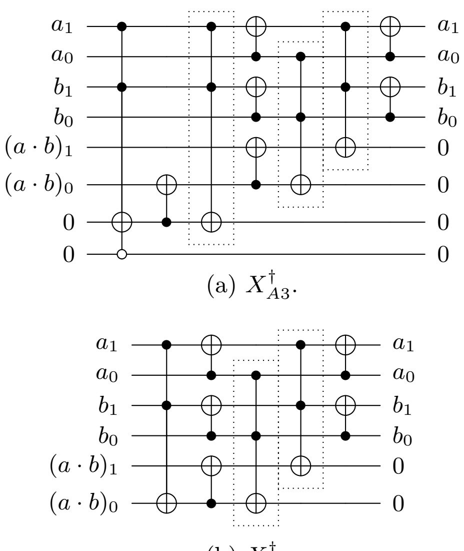
(b) X_{A2}.

(c) X_{A3}.

Fig. 9: Three quantum circuits for multiplication in $\mathbf{GF}(2^2)$.

Notation	Description
$X^\dagger_{A3}$	AND × 1, AND† × 3
$X^\dagger_{A2}$	Toffoli × 1, AND† × 2

Table 3: Two quantum circuits of multiplication dagger in $\mathbf{GF}(2^2)$.



(a) $X^\dagger_{A3}$.

(b) $X^\dagger_{A2}$.

Fig. 10: Two quantum circuits of multiplication dagger in $\mathbf{GF}(2^2)$.

4.2 GF(2^4) arithmetic quantum circuits

The arithmetic in $\mathbf{GF}(2^4)$ is performed as $\mathbf{GF}((2^2)^2)$ as explained in Section. 3. We use constant(λ) multiplication, squaring, multiplication and multiplicative inversion in $\mathbf{GF}(2^4)$. The quantum circuits of squaring and constant multiplication were previously described in Fig. 4 and Fig. 5, respectively. Note that those quantum circuits were implemented by using only Clifford gates. Now we describe our improvement on the quantum circuits of multiplication and multiplicative inversion in $\mathbf{GF}(2^4)$.

Multiplication As depicted in Fig. 6, multiplication in $\mathbf{GF}(2^4)$ consists of addition, constant(ϕ) multiplication, and three times of multiplication in $\mathbf{GF}(2^2)$. The quantum circuits for addition and constant multiplication in $\mathbf{GF}(2^2)$ can be implemented with only CNOT gates. Three times of multiplication in $\mathbf{GF}(2^2)$

can be performed either in parallel or in a combination of parallel and series, depending on the available amount of qubits. Also, the arrangement of the quantum circuits for multiplication in $\mathbf{GF}(2^2)$ introduced in Section 4.1 will have an influence on time-space complexity.

Hereafter, we analyze the number of multiplication in $\mathbf{GF}(2^2)$ that can be performed in parallel, and the proper type of quantum circuits for the multiplication. These will have a significant effect on time-space complexity of the quantum circuit for multiplication in $\mathbf{GF}(2^4)$.

We use four types of quantum circuits for multiplication in $\mathbf{GF}(2^4)$. We denote each of them by LX_{T3}, LX_{AA3}, LX_{A2}, and LX_{A3}, and these are shown in Fig. 11. Each name of the circuits characterizes the type of multiplication in $\mathbf{GF}(2^2)$ and the arrangement in the circuit.

On the other hand, we use only one type of quantum circuit for dagger operation of multiplication in $\mathbf{GF}(2^2)$, $X^{\dagger}_{A3}$. Because the proposed quantum circuits for multiplication in $\mathbf{GF}(2^4)$ have enough qubits to perform $X^{\dagger}_{A3}$ in all circuit design, we use $X^{\dagger}_{A3}$ which has the smallest T-depth.

The three types of quantum circuit for multiplication in $\mathbf{GF}(2^2)$ are arranged to reduce the T-depth as much as possible within the available number of qubits. The number of qubits dedicated to the multiplication circuit in $\mathbf{GF}(2^4)$ is shown on the left side of each figure. The quantum circuits for multiplication in $\mathbf{GF}(2^2)$ arranged in the same column represent parallel execution, and those arranged in different columns as shown in Fig. 11b represent operations performed in series.

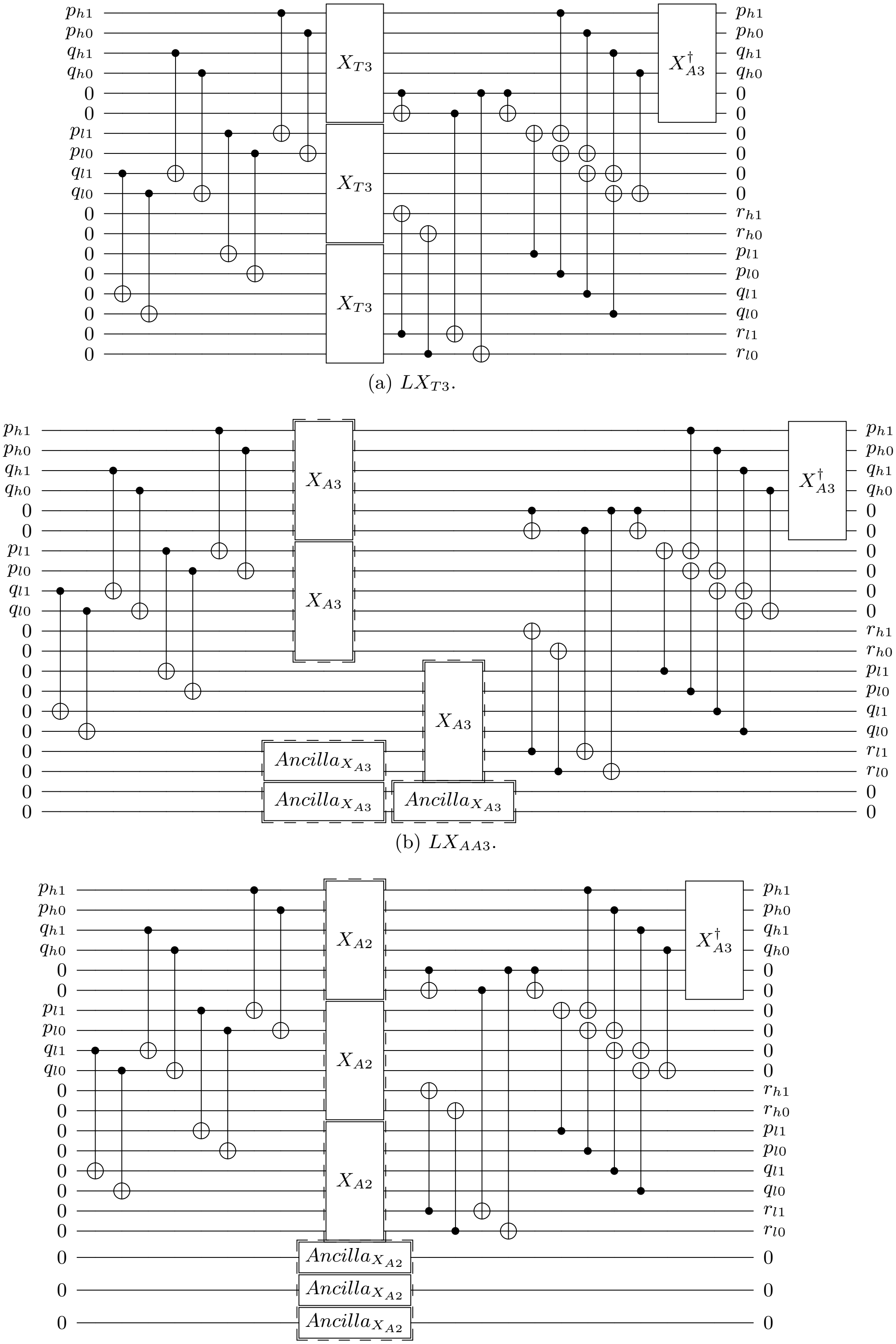
Multiplicative inversion The classical circuit diagram for multiplicative inversion in $\mathbf{GF}(2^4)$ is depicted in Fig. 7. A multiplicative inversion in $\mathbf{GF}(2^4)$ requires three quantum circuits for multiplication in $\mathbf{GF}(2^2)$ and two of three can be executed parallel. This quantum circuit requires additional two qubits for saving d^{-1} which is required during clean-up process of the multiplicative inversion in $\mathbf{GF}(2^8)$.

Because the quantum circuit for multiplicative inversion in $\mathbf{GF}(2^8)$ has sufficient number of available qubits, we use only X_{A3} to build the quantum circuit for multiplicative inversion in $\mathbf{GF}(2^4)$.

4.3 $\mathbf{GF(2^8)}$ multiplicative inversion quantum circuits

The multiplicative inversion in $\mathbf{GF}(2^8)$ consists of squaring, constant(λ) multiplication, multiplicative inversion, and multiplications in $\mathbf{GF}(2^4)$ as depicted in Fig. 8. Squaring and constant multiplication can be implemented by CNOT gates only.

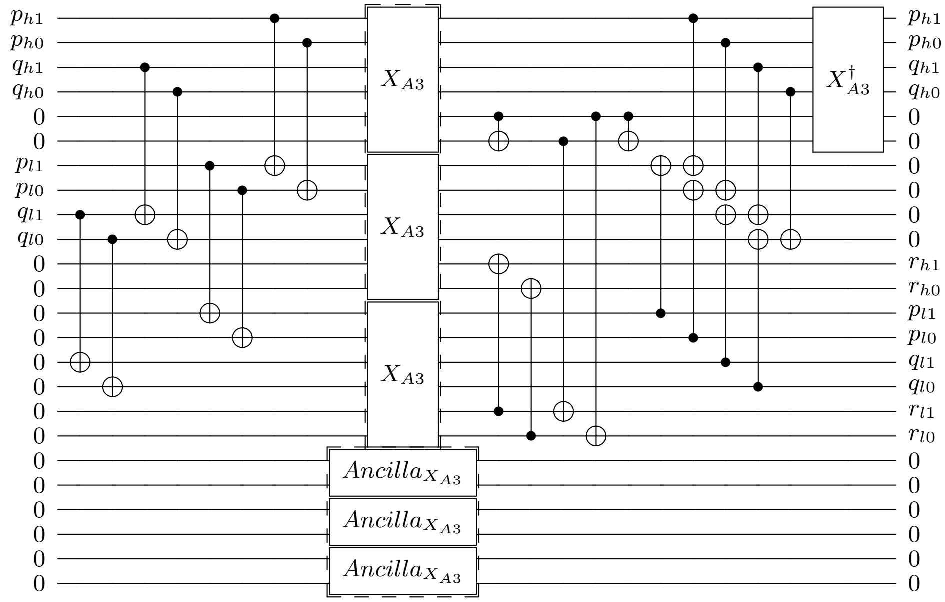
The quantum circuit for multiplicative inversion in $\mathbf{GF}(2^8)$ requires clean-up process. The clean-up process means to make the qubits that store the intermediate values generated during the calculation process into a specific state (usually $|0\rangle$). This process is necessary to improve the reusability of quantum resources such as qubits when the proposed quantum circuit is combined in other



(a) LX_{T3}.

(b) LX_{AA3}.

(c) LX_{A2}.



(d) LX_{A3}.

Fig. 11: Four quantum circuits for multiplication in $GF(2^4)$.

quantum circuits such as AES S-box. The quantum circuit for clean-up process is similar to that of the target operation in reverse, but it is not symmetric because the result of the target operation must be maintained. Due to this reason, the suggested quantum circuit has a complicated structure compared to the circuit diagram in Fig. 8.

As shown in Fig. 12 to Fig. 14, the quantum circuit can be largely divided into two parts. Forward operation part calculates multiplicative inversion in $\mathbf{GF}(2^8)$ and another part performs clean-up process. In the forward operation, unlike the circuit diagram in Fig.8, one more $\mathbf{GF}(2^4)$ multiplication is included. This is because keeping $b_h \times b_l$ reduces the time-space complexity of the entire quantum circuit which includes clean-up process.

A detailed quantum circuit description, including squaring, addition and constant multiplication, is provided through the source code in the Appendix. What we mainly describe here are what kinds of quantum circuit for multiplication in $\mathbf{GF}(2^4)$ are used and how these are arranged in the quantum circuit for multiplicative inversion in $\mathbf{GF}(2^8)$. These two factors have a significant effect on the time-space complexity of the quantum circuit for multiplicative inversion in $\mathbf{GF}(2^8)$.

In section 5, we analyze time-space complexity for seven quantum circuits for multiplicative inversion in $\mathbf{GF}(2^8)$. However, in this section, we depict only three of them which have significant improvement. Each is a quantum circuit that minimizes qubit consumption, a quantum circuit that minimizes T-depth, and a quantum circuit that balances qubit consumption and T-depth.

4.4 Affine transformation quantum circuit

The affine transformation is expressed as the following equation.

$$\{b\} = M\{b^{'}\} \oplus \{v\}$$

where M is the matrix below, $\{v\}$ is a fixed vector and $\{b^{'}\} = (b_0^{'}, b_1^{'}, b_2^{'}, b_3^{'}, b_4^{'}, b_5^{'}, b_6^{'}, b_7^{'})$ is the result of the multiplicative inversion for the input of AES S-box.

$$M\{b^{'}\} = \begin{bmatrix} 1\,0\,0\,0\,1\,1\,1\,1 \\ 1\,1\,0\,0\,0\,1\,1\,1 \\ 1\,1\,1\,0\,0\,0\,1\,1 \\ 1\,1\,1\,1\,0\,0\,0\,1 \\ 1\,1\,1\,1\,1\,0\,0\,0 \\ 0\,1\,1\,1\,1\,1\,0\,0 \\ 0\,0\,1\,1\,1\,1\,1\,0 \\ 0\,0\,0\,1\,1\,1\,1\,1 \end{bmatrix} \begin{bmatrix} b_0^{'} \\ b_1^{'} \\ b_2^{'} \\ b_3^{'} \\ b_4^{'} \\ b_5^{'} \\ b_6^{'} \\ b_7^{'} \end{bmatrix} \text{ and } \{v\} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

The matrix M can be decomposed into three matrices, P, L and U by PLU decomposition. These matrices can be implemented in a quantum circuit with only CNOT gates. On the other hand, the modular addition of $\{v\}$ can be implemented in a quantum circuit with only X-gates. In Fig. 15, we depict a quantum circuit for affine transformation of AES S-box which combined the above two operations.

$$\mathrm{P} \coloneqq \begin{bmatrix} 1\,0\,0\,0\,0\,0\,0\,0 \\ 0\,1\,0\,0\,0\,0\,0\,0 \\ 0\,0\,1\,0\,0\,0\,0\,0 \\ 0\,0\,0\,1\,0\,0\,0\,0 \\ 0\,0\,0\,0\,1\,0\,0\,0 \\ 0\,0\,0\,0\,0\,0\,1\,0 \\ 0\,0\,0\,0\,0\,0\,0\,1 \\ 0\,0\,0\,0\,0\,1\,0\,0 \end{bmatrix}, \mathrm{L} \coloneqq \begin{bmatrix} 1\,0\,0\,0\,0\,0\,0\,0 \\ 1\,1\,0\,0\,0\,0\,0\,0 \\ 1\,1\,1\,0\,0\,0\,0\,0 \\ 1\,1\,1\,1\,0\,0\,0\,0 \\ 1\,1\,1\,1\,1\,0\,0\,0 \\ 0\,0\,0\,1\,1\,1\,0\,0 \\ 0\,1\,1\,1\,0\,0\,1\,0 \\ 0\,0\,1\,1\,1\,0\,0\,1 \end{bmatrix}, \mathrm{U} \coloneqq \begin{bmatrix} 1\,0\,0\,0\,1\,1\,1\,1 \\ 0\,1\,0\,0\,1\,0\,0\,0 \\ 0\,0\,1\,0\,0\,1\,0\,0 \\ 0\,0\,0\,1\,0\,0\,1\,0 \\ 0\,0\,0\,0\,1\,0\,0\,1 \\ 0\,0\,0\,0\,0\,1\,0\,0 \\ 0\,0\,0\,0\,0\,0\,1\,0 \\ 0\,0\,0\,0\,0\,0\,0\,1 \end{bmatrix}.$$

As shown in Fig.15, the quantum circuit of the affine transformation is executable in-place and does not include T-gate. Therefore, the affine transformation does not occur additional T-depth or qubits.

5 Evaluation

We evaluate our proposed method in terms of time-space complexity. To be specific, the number of qubits is used as the unit of space complexity and the

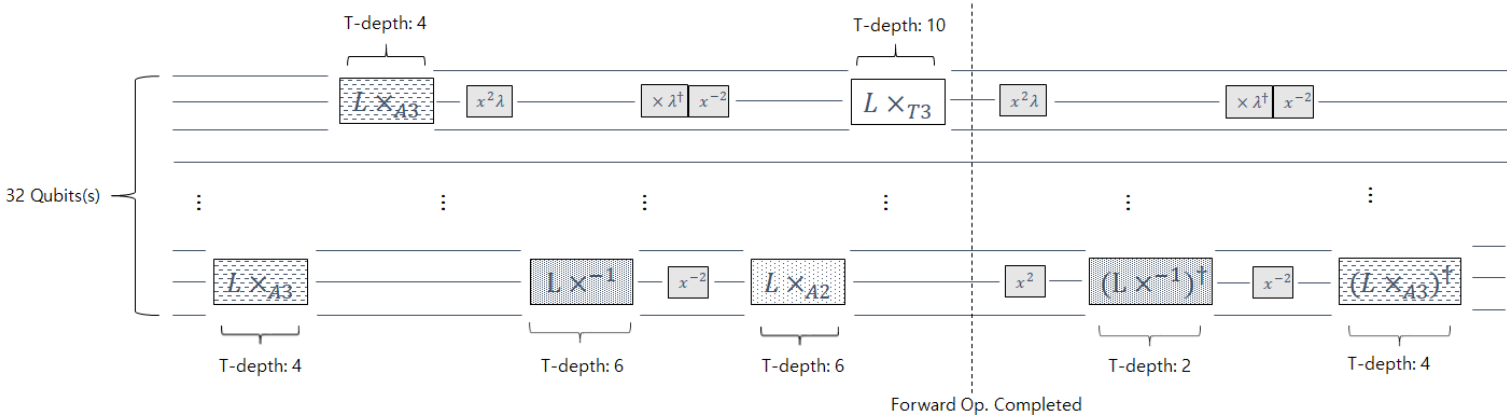


Fig. 12: Quantum circuit for the multiplicative inversion in $\mathbf{GF}(2^8)$, minimum width.

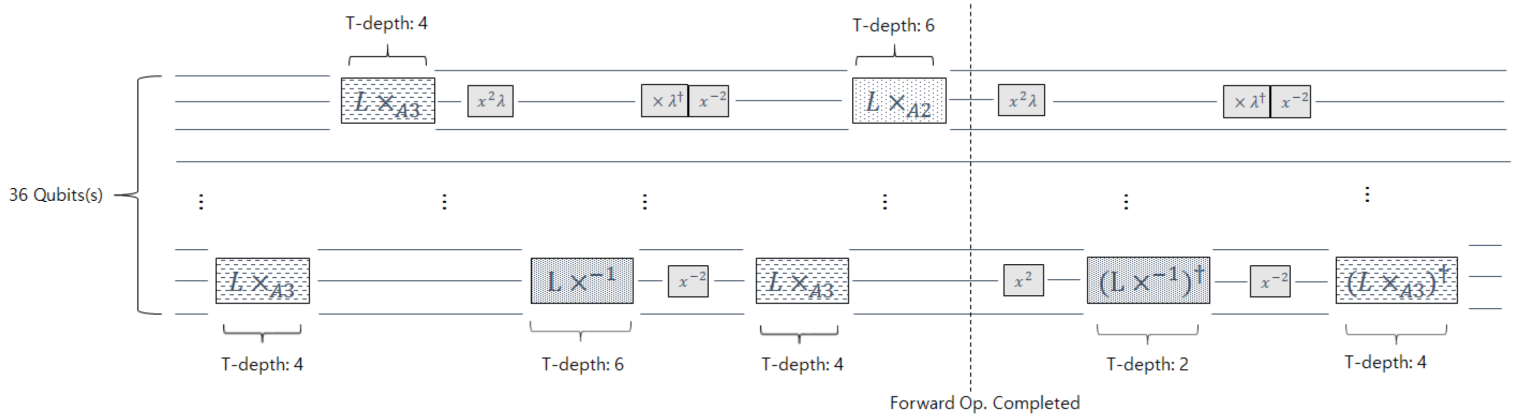


Fig. 13: Quantum circuit for the multiplicative inversion in $\mathbf{GF}(2^8)$, balanced.

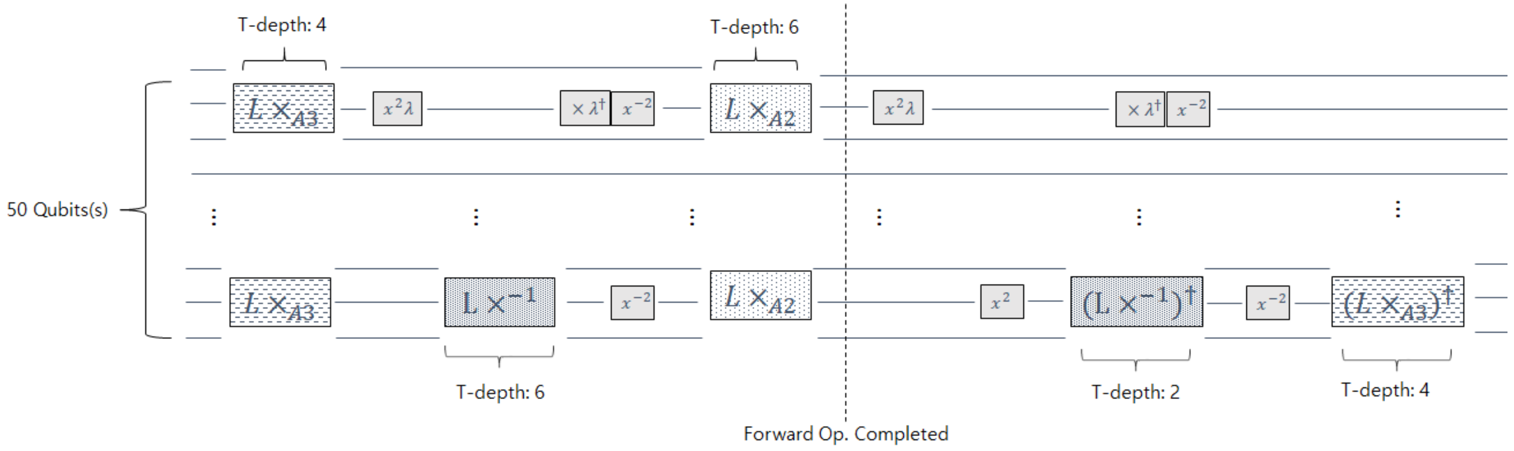


Fig. 14: Quantum circuit for the multiplicative inversion in $\mathbf{GF}(2^8)$, minimum depth.

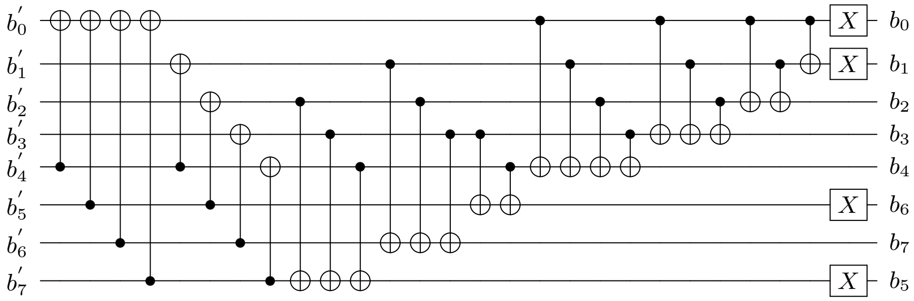


Fig. 15: Quantum circuit of the Affine transformation.

T-depth is used as the unit of time complexity. To design a quantum circuit that calculates multiplicative inversion in $\mathbf{GF}(2^8)$, we proposed a quantum circuit design with varying time-space complexity for smaller Galois field, such as $\mathbf{GF}(2^4)$ and $\mathbf{GF}(2^2)$. By selecting the quantum circuit for the smaller Galois field operation according to the situation, the time-space complexity of the quantum circuit that calculates the multiplicative inversion in $\mathbf{GF}(2^8)$ can be reduced in a trade-off manner.

5.1 Multiplication and dagger in $\mathbf{GF(2^2)}$

Symbol	# qubit	T-depth	Description
X_{T3}	6	9	Toffoli × 3
X_{A2}	7	5	AND × 2, Toffoli × 1
X_{A3}	8	3	AND × 3, AND† × 1
$X_{A3}^\dagger$	10	1	AND × 1, AND† × 3
$X_{A2}^\dagger$	6	3	Toffoli × 1, AND† × 2

Table 4: Time-space complexity of multiplication and its dagger in $\mathbf{GF}(2^2)$.

As explained in Section 3, the multiplication operation quantum circuit in $\mathbf{GF}(2^2)$ varies in time-space complexity according to the combination of AND gates (and its dagger) and Toffoli gates. In the case of AND gate, compared to Toffoli gate, qubit consumption is increased, but T-Depth is reduced. The summary of multiplication operation in $\mathbf{GF}(2^2)$ used in our proposed scheme is depicted as Table 4

5.2 Multiplication in $\mathbf{GF(2^4)}$

The quantum circuit for multiplication in $\mathbf{GF}(2^4)$ differs in time-space complexity depending on which quantum circuit for multiplication in $\mathbf{GF}(2^2)$ is used,

Symbol	# qubit	T-depth	Description
LX_{T3}	18	10	$X_{T3}\times$ 3, $X^{\dagger}_{A3}\times 1$
LX_{AA3}	20	7	$X_{A3}\times$ 2, $X_{A3}\times$ 1, $X^{\dagger}_{A3}\times 1$
LX_{A2}	21	6	$X_{A2}\times$ 3, $X^{\dagger}_{A3}\times 1$
LX_{A3}	24	4	$X_{A3}\times$ 3, $X^{\dagger}_{A3}\times 1$
LX^{-1}	18	6	$X_{A3}\times$ 1, $X_{A3}\times 2$

Table 5: Time-space complexity of multiplication and its dagger in $\mathbf{GF}(2^4)$.

and whether the quantum circuits are arranged in sequential or parallel. The summary of multiplication operation in $\mathbf{GF}(2^2)$ used in our proposed scheme is depicted as Table 5.

5.3 Multiplicative Inversion in $\mathbf{GF(2^8)}$

Scheme	type	# qubit	T-depth
Grassl et al. [7]		44	217
Langenberg et al. [12]		32	120
Jaques et al. [9]	balanced	41	35
	minimum depth	137	6
Our scheme	minimum width	32	36
	balanced 1	34	31
	balanced 2	36	30
		42	27
		46	25
		50	22
	minimum T-depth	54	20

Table 6: Comparison of time-space complexity with prior work.

The quantum circuit of multiplicative inversion in $\mathbf{GF}(2^8)$ has a more complex structure than the quantum circuit of $\mathbf{GF}(2^4)$ and $\mathbf{GF}(2^2)$. However, the dominant factor on its time-space complexity is which quantum circuits of $\mathbf{GF}(2^4)$ is used and how they are arranged. The Table 6 summarizes the time-space complexity of multiplicative inversion in $\mathbf{GF}(2^8)$ for each possible combination.

Acknowledgment

This work was supported by Institute for Information communications Technology Planning Evaluation (IITP) grant funded by the Korea government(MSIT)

($\langle$Q|Crypton$\rangle$, No.2019-0-00033, Study on Quantum Security Evaluation of Cryptography based on Computational Quantum Complexity)

References

1. Matthew Amy, Dmitri Maslov, Mchele Mosca, and Martin Roetteler. A meet-in-the-middle algorithm for fast synthesis of depth-optimal quantum circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 32(6):818–830, 2013.
2. Fowler Austion G., Stephens Ashley M., and Groszkowski Peter. High-threshold universal quantum computation on the surface code. *Phys. Rev. A*, 80(5):052312, 2009. Full version available at `https://journals.aps.org/pra/cited-by/10.1103/PhysRevA.80.052312`.
3. Joan Boyar and René Peralta. A New Combinational Logic Minimization Technique with Applications to Cryptology. In Paola Festa, editor, *Experimental Algorithms. SEA 2010*, volume 6049 of *LNCS*, pages 178–189. Springer, 2010.
4. Joan Boyar and René Peralta. A small depth-16 circuit for the AES S-Box. In Dimitris Gritzalis, Steven Furnell, and Marianthi Theoharidou, editors, *Information Security and Privacy Research. SEC 2012*, volume 376 of *IFIPAICT*, pages 287–298. Springer, 2012.
5. Amento-Adelmann Brittanney, Markus Grassl, Brandon Langenberg, Yi-Kai Liu, Eddie Schoute, and Rainer Steinwandt. Quantum Cryptanalysis of Block Ciphers: A Case Study. In *Poster at Quantum Information Processing QIP*, 2018.
6. Craig Gidney. Halving the cost of quantum addition. *Quantum*, 2:74, 2018.
7. Markus Grassl, Brandon Langenberg, Martin Roetteler, and Rainer Steinwandt. Applying Grover's algorithm to AES: quantum resource estimates. In Tsuyoshi Takagi, editor, *Post-Quantum Cryptography - PQCrypto 2016*, volume 9606 of *LNCS*, pages 29–43. Springer, 2016.
8. Lov K. Grover. A fast quantum mechanical algorithm for database search. In Gary L. Miller, editor, *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing - STOC '96*, pages 212–219. ACM, 1996.
9. Samuel Jaques, Michael Naehrig, Martin Roetteler, and Fernando Virdia. Implementing Grover oracles for quantum key search on AES and LowMC. In Anne Canteaut and Yuval Ishai, editors, *Advances in Cryptology - EUROCRYPT 2020 (Proceedings, Part II)*, volume 12106 of *LNCS*, pages 280–310. Springer, 2020.
10. Junki Kang, Dooho Choi, Yong-Je Choi, and Dong-Guk Han. Secure Hardware Implementation of ARIA Based on Adaptive Random Masking Technique. *ETRI Journal*, 34(1), 2012.
11. Panjin Kim, Daewan Han, and Kyung Chul Jeong. Time-space complexity of quantum search algorithms in symmetric cryptanalysis: applying to AES and SHA-2. *Quantum Information Processing*, 17(12):339, 2018.
12. Brandon Langenberg, Hai Pham, and Rainer Steinwandt. Reducing the Cost of Implementing the Advanced Encryption Standard as a Quantum Circuit. *IEEE Transactions on Quantum Engineering*, 1:1–12, 2020.
13. Akashi Satoh, Sumio Morioka, Kohji Takano, and Seiji Munetoh. A compact Rijndael Hardware Architecture with S-Box Optimization. In Colin Boyd, editor, *Advances in Cryptology - ASIACRYPT 2001*, volume 2248 of *LNCS*, pages 239–254. Springer, 2001.

14. Peter Selinger. Quantum circuits of T-depth one. *Phys. Rev. A*, 87(4):042302, 2013. Full version available at `https://journals.aps.org/pra/abstract/10.1103/PhysRevA.87.042302`.
15. Peter W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM review*, 4(2):303–332, 1999.

Appendix

$GF(2^8)$ multiplicative inversion