KIPS Trans. Comp. and Comm. Sys.
Vol.10, No.4 pp.101~106
ISSN: 2287-5891 (Print), ISSN 2734-049X (Online)

https://doi.org/10.3745/KTCCS.2021.10.4.101

Optimization of LEA Quantum Circuits to Apply Grover's Algorithm

Kyung Bae Jang† · Hyun Jun Kim† · Jae Hoon Park† · Gyeung Ju Song†† · Hwa Jeong Seo†††

ABSTRACT

Quantum algorithms and quantum computers can break the security of many of the ciphers we currently use. If Grover's algorithm is applied to a symmetric key cipher with n-bit security level, the security level can be lowered to $(n/2)$-bit. In order to apply Grover's algorithm, it is most important to optimize the target cipher as a quantum circuit because the symmetric key cipher must be implemented as a quantum circuit in the oracle function. Accordingly, researches on implementing AES(Advanced Encryption Standard) or lightweight block ciphers as quantum circuits have been actively conducted in recent years. In this paper, korean lightweight block cipher LEA was optimized and implemented as a quantum circuit. Compared to the previous LEA quantum circuit implementation, quantum gates were used more, but qubits were drastically reduced, and performance evaluation was performed for this tradeoff problem. Finally, we evaluated quantum resources for applying Grover's algorithm to the proposed LEA implementation.

Keywords : Quantum Computer, Grover's Alogoritm, LEA, Quantum Resource

그루버 알고리즘 적용을 위한 LEA 양자 회로 최적화

장 경 배† · 김 현 준† · 박 재 훈† · 송 경 주†† · 서 화 정†††

요 약

양자 알고리즘과 양자 컴퓨터는 우리가 현재 사용하고 있는 많은 암호들의 안전성을 깨뜨릴 수 있다. 그루버 알고리즘을 n-bit 보안레벨을 가지는 대칭키 암호에 적용한다면 보안레벨을 $(n/2)$-bit 까지 낮출 수 있다. 그루버 알고리즘을 적용하기 위해서는 오라클 함수에 대칭키 암호가 양자 회로로 구현되어야 하기 때문에 대상 암호를 양자 회로로 최적화하는 것이 가장 중요하다. 이에 AES 또는 경량 블록암호를 양자 회로로 구현하는 연구들이 최근 활발히 진행되고 있다. 본 논문에서는 국산 경량 블록암호 LEA를 양자 회로로 최적화하여 구현 하였다. 기존의 LEA 양자 회로 구현과 비교하여 양자 게이트는 더 많이 사용하였지만, 큐빗을 획기적으로 줄일 수 있었으며 이러한 트레이드오프 문제에 대한 성능 평가를 수행하였다. 마지막으로 제안하는 LEA 양자 회로에 그루버 알고리즘을 적용하기 위한 양자 자원들을 평가하였다.

키워드 : 양자 컴퓨터, 그루버 알고리즘, LEA, 양자 자원

1. 서 론

IoT(Internet of Things) 기술이 발전함에 따라 수많은 스마트 디바이스들이 일상생활 속에 자리 잡고 있다[1]. 그리고 IoT 디바이스들은 단순한 센서 데이터뿐만 아니라 개인에게 민감한 정보들까지 포함하여 서로 통신하고 있다. 개인정보 침해의 문제를 막기 위해서는 디바이스들의 통신 과정이 제 3자로부터 보호되어야 한다. 이러한 보안성을 달성하기 위해 암호 알고리즘이 데이터 보호에 적용된다. 하지만 작은 메모리와 낮은 성능의 IoT 디바이스에는 일반 컴퓨터에 적용되는 암호 알고리즘을 사용하기에는 무리가 있다. 이러한 상황을 해결하기 위해, 국내에서는 저전력 디바이스를 대상으로 한 경량 블록 암호 LEA(Lightweight Encryption Algorithm)[2], CHAM[3], HIGHT[4]가 개발되었다.

암호 알고리즘은 고전 컴퓨터의 공격에 대해서 안전해야 하는 것은 물론이며 양자 컴퓨터와 양자 알고리즘의 공격에 대해서도 내성을 가져야 한다. 양자 알고리즘인 그루버 알고리즘은 n-bit 보안레벨을 가지는 대칭키 암호 알고리즘에 대하여 $n/2$-bit 보안레벨까지 낮출 수 있는 블록암호에 대한 가

※ 이 성과는 부분적으로 이 논문은 2019년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임(〈Q|Crypton〉, No.2019-0-00033, 미래컴퓨팅 환경에 대비한 계산 복잡도 기반 암호 안전성 검증 기술개발) 그리고 이 성과는 부분적으로 2020년도 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No. NRF-2020R1F1A1048478). 본 연구는 서화정 교수의 한성대학교 교내학술연구비 지원과제임.
※ 이 논문은 2020년 한국정보처리학회 추계학술발표대회의 우수논문으로 "그루버 알고리즘 적용을 위한 LEA 양자 회로 최적화"의 제목으로 발표된 논문을 확장한 것임.
† 준 회 원 : 한성대학교 IT융합공학부 석사과정
†† 준 회 원 : 한성대학교 IT융합공학부 학사과정
††† 종신회원 : 한성대학교 IT융합공학부 조교수
Manuscript Received : December 14, 2020
Accepted : February 15, 2021
* Corresponding Author : Hwa Jeong Seo(hwajeong84@gmail.com)

장 효과적인 공격 방법이다[5]. 양자 컴퓨터는 고전 비트가 아닌 양자 역학 특성의 큐비트를 사용한다. 큐비트는 0과 1이 동시 확률로서 존재하는 가장 중요한 중첩이라는 성질을 가지고 있다. 큐비트를 관측하기 전까지는 중첩 상태를 유지하기 때문에 연산 과정에서 모든 경우의 수를 병렬로 계산할 수 있다. 양자 컴퓨터는 아직 개발 초기 단계이기 때문에 그루버 알고리즘을 활용하기 위한 최적의 자원을 찾는 것이 중요하며 최근에 많은 연구가 진행되고 있다. [6]은 블록암호 AES(Advanced Encryption Standard)를 양자 회로로 구현하여 그루버 알고리즘을 적용하는데 필요한 양자 자원을 추정하였고, [7,8]은 앞선 연구보다 최적화 된 AES 양자 회로를 제안하였다. [9]은 미국 NSA(National Security Agency)에서 개발한 경량 블록암호 SIMON을 양자 회로로 구현하였으며 [10]에서는 SPECK을 양자 회로로 구현하였다. 마지막으로 [11]은 국산 경량 블록 암호 HIGHT, LEA, CHAM을 양자 회로로 구현 하였다. 본 논문에서는 국산 경량 블록 암호 LEA를 양자 회로로 최적화 구현하여 그루버 알고리즘 적용 자원을 분석하였다. [11]의 LEA 구현보다 양자 게이트를 더 많이 사용하지만, 사용 큐빗의 개수를 획기적으로 줄일 수 있었다. 현재 양자 컴퓨터는 사용 가능한 큐빗의 개수를 늘려야 하는 과제에 직면해 있다. 제안하는 기법을 통해 조금의 양자 게이트 비용이 추가되는 대신 그루버 알고리즘을 적용하기 위해 필요한 큐빗의 개수를 효과적으로 줄일 수 있다.

2. 관련 연구

2.1 양자 게이트

기존 컴퓨터에서 사용되는 NOT, XOR, AND 연산을 양자 컴퓨터에서는 양자 게이트로 대체할 수 있다. 대표적인 양자 게이트로 X 게이트, CNOT 게이트, Toffoli 게이트가 있다. X 게이트는 해당 큐빗의 상태를 반전시키며 기존 컴퓨터의 NOT 연산에 해당한다. CNOT 게이트는 2개의 큐빗을 입력받아 첫 번째 큐빗이 1인 상태라면 두 번째 큐빗의 상태를 반전시킨다. CNOT 게이트는 기존 XOR 연산을 대체하며 Fig. 1과 같다.

Toffoli 게이트는 3개 큐빗을 입력받아 첫 번째와 두 번째 큐빗의 상태가 모두 1이라면, 세 번째 큐빗의 상태를 반전시킨다. 기존 AND 연산을 대체할 수 있으며 Fig. 2와 같다. Fig. 1과 Fig. 2의 회로 그림을 보면 큰 차이가 없어 보이지만 실제로 Toffoli 게이트 1개의 사용 비용은 6개의 CNOT 게이트 사용보다 높다. 양자 회로 최적화 관점에서는 Toffoli 게이트와 큐빗의 사용을 줄이는 것이 매우 중요하다.

2.2 그루버 알고리즘

그루버 알고리즘은 정렬되지 않은 데이터 집합에서 특정 데이터를 찾아내는 양자 알고리즘이다. 만약 특정 데이터를 찾기 위해 $O(2^n)$번의 검색이 필요한 경우, 그루버 알고리즘

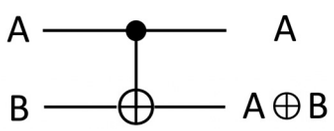

Fig. 1. CNOT Gate

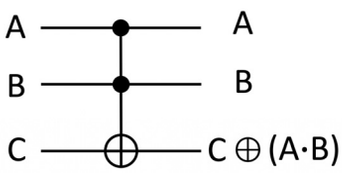

Fig. 2. Toffoli Gate

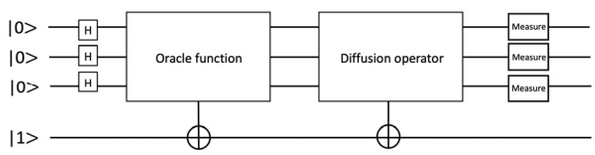

Fig. 3. Grover's Algorithm Quantum Circuit

을 사용하면 $O(2^{n/2})$번만의 검색으로 만으로 찾아낼 수 있다. 그루버 알고리즘은 정답을 분류하는 오라클과 분류된 정답이 출력 될 확률을 높이는 확산 연산자로 구성되어 있다. 그루버 알고리즘 양자 회로는 Fig. 3과 같다.

2.3 LEA

LEA는 한국인터넷진흥원에서 개발한 128-bit 경량블록암호이며 IoT, 빅데이터, 클라우드 또는 모바일 및 저전력 디바이스에서 빠른 암호화 성능과 데이터 전송 간 기밀성을 보장한다. 2013년도에 개발되었으며 128, 192 또는 256-bit의 키를 선택하여 사용할 수 있다. 소프트웨어 환경에서 AES보다 2배 빠른 성능을 보여주며, 국내 TTA(Telecommunications Technology Association) 표준으로 제정되어 있다.

2.4 국산 블록암호에 대한 그루버 알고리즘 적용 자원 추정 연구[11]

그루버 알고리즘의 공격 대상이 되는 블록암호를 양자 회로로 구현하여 이에 대한 양자 자원들을 추정하는 연구들이 활발히 수행되고 있다. 2020년, [11]에서는 국산 경량 블록암호 LEA, CHAM, HIGHT를 양자 회로로 구현할 때 사용되는 큐비트, 양자 게이트 수를 최적화하는 방법들을 제시하였다. 제안하는 기법들을 통해 양자 회로로 구현하기 위해 필요한 양자 자원들을 추정하였으며 SPECK, SIMON의 양자회로 구현 결과들과 비교 분석하였다.

각 암호 알고리즘의 특성에 따라 최적화된 양자 회로 구현 기법이 제시되었으며, 공통적으로는 사용 큐빗을 절약하기 위해 라운드 함수와 키 스케줄을 병행하여 수행하는 기법이 적용되었다. 하지만 3가지 국산 암호들 중 LEA에 가장 많은 양자 자원이 사용되었는데, 특히 큐빗의 사용 개수가 매우 높

Table 1. Previous Implementation Result [11]

	Qubit	Toffoli	CNOT	X
CHAM 64/128	196	2,400	12,285	240
CHAM 128/128	268	4,960	26,885	240
CHAM 128/256	396	5,952	32,277	304
HIGHT	201	6,272	20,523	4
LEA 128	385	10,416	28,080	68
LEA 192	513	15,624	39,816	100
LEA 256	641	17,856	45,504	130

다. 해당 논문의 구현 결과는 Table 1과 같다.

LEA의 키 스케줄 함수에서 상수 덧셈이 수행되는데 이때 필요한 상수들을 모두 큐빗으로 할당하였기 때문이다. 키 스케줄에서 사용하는 상수들은 사전에 정의된 상수들을 사용하며 모든 키 스케줄에서 사용되는 상수를 사전에 계산할 수 있다. 사전 계산이 가능하기 때문에 본 논문에서는 하나의 상수 큐빗 배열을 재활용하며 키 스케줄을 수행한다. 상수를 변환하는 작업에 양자 게이트가 소모되었지만, 다른 상수들을 위한 큐빗을 할당하지 않음으로써 큐빗의 개수를 줄일 수 있었다.

3. 제안 기법

본 논문에서는 LEA 키 스케줄의 상수 덧셈 부분을 최적화하였다. LEA의 키 스케줄에서 상수 값은 'L', 'E', 'A'를 ASCII 코드로 표현한 δ를 사용하며 Equation (1)과 같다.

$$\begin{aligned}
&\delta[0]=0xc3efe9db, && \delta[1]=0x44626b02\\
&\delta[2]=0x79e27c8a, && \delta[3]=0x78df30ec\\
&\delta[4]=0x715ea49e, && \delta[5]=0xc785da0a\\
&\delta[6]=0xe04ef22a, && \delta[7]=0xe5c40957
\end{aligned} \tag{1}$$

LEA의 키 스케줄에서는 입력 키 값 K와 상수 δ의 모듈러 덧셈 그리고 n-bit 좌측 로테이션 ROL_n 연산을 통해 라운드 키 RK_i를 생성한다. LEA-128의 키 스케줄은 Equation (2)와 같다.

$$\begin{aligned}
K[0] &= ROL_1(K[0] \boxplus ROL_i(\delta[i \bmod 4]))\\
K[1] &= ROL_3(K[1] \boxplus ROL_{i+1}(\delta[i \bmod 4]))\\
K[2] &= ROL_6(K[2] \boxplus ROL_{i+2}(\delta[i \bmod 4]))\\
K[3] &= ROL_{11}(K[3] \boxplus ROL_{i+3}(\delta[i \bmod 4]))\\
RK_i &= (K[0], K[1], K[2], K[1], K[3], K[1])
\end{aligned} \tag{2}$$

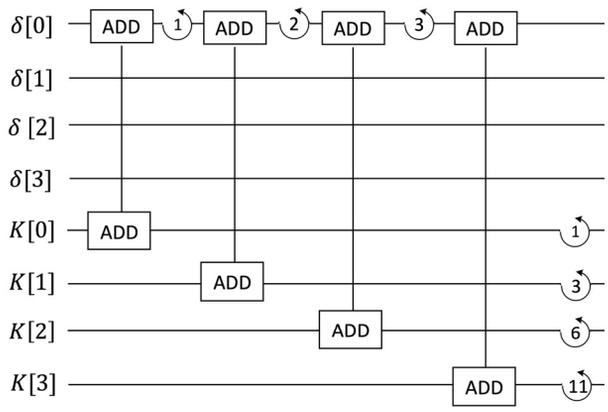

Fig. 4. Previous LEA-128 Key Schedule Quantum Circuit

기존의 LEA 구현에서는 K에 로테이션 된 δ 의 덧셈을 수행하기 위해, LEA-128의 경우 δ[0], δ[1], δ[2], δ[3] 값을 위한 큐빗들을 할당하였다. δ 하나당 32 비트로 구성되어 각 32 큐빗이 사용되고 총 128(32×4)개의 큐빗이 사용 되었다. LEA-192은 192개의 큐빗, LEA-256은 256개의 큐빗이 δ 값 선언에 사용되었다. 때문에 CHAM, HIGHT보다 훨씬 많은 큐빗들이 사용되었다. LEA-128의 첫 라운드 키 스케줄에 대한 양자 회로는 Fig. 4와 같다.

사용 가능 큐빗의 개수를 늘리는 것은 양자 컴퓨터 개발에 있어 핵심 쟁점이다. 따라서 한정된 큐빗 안에서 계산을 진행해야 하기 때문에 구현 시 큐빗의 개수를 줄이는 것이 매우 중요하다.

본 논문에서는 기존 최대 8개의 δ[0~8]을 사용하는 경우에도 하나의 δ 값만 사용한다. 사용되는 δ의 값과 순서는 정해져 있다. 따라서 사전에 키 스케줄의 연산에 필요한 δ값을 X 게이트만을 활용하여 만들 수 있다. 동작 예는 다음과 같다. δ[0]가 가장 먼저 사용되기 때문에 선언한 32 큐빗의 δ를 δ[0]의 32비트 값 중 1인 위치에 X 게이트를 통과 시킨다. 예를 들어 δ[0]가 만약 0x00000002 라면 2번째 큐빗에만 X 게이트를 통과시켜 0에서 1로 반전시켜 값을 완성한다. 사용된 후에는 다음에 사용될 δ[1]의 값으로 교체해주어야 한다. 하지만 기존 컴퓨터와 달리 양자 컴퓨터에서는 기존 값을 새로운 값으로 덮어씌우는 것이 불가능하다. 따라서 기존 상태에서 X 게이트를 통과시켜 다음 값을 만들어준다. 만약 δ[1]가 0x00000003 이라면 첫 번째 큐빗에만 X 게이트를 통과시켜 값을 완성해줄 수 있다. 위와 같은 방식으로 키 스케줄에서 사용되는 모든 상수 δ 덧셈을 하나의 32큐빗 배열만을 활용하여 수행한다. 하나의 예를 들어 δ[0]에서 δ[1] 으로 X 게이트로 값을 바꾸는 자세한 과정은 Alg. 1과 같다.

값을 변경할 때 마다 추가적인 X 게이트를 사용하지만 이는 다른 Toffoli, CNOT 게이트와 비교하여 매우 비용이 낮은 게이트이다. 따라서 조금의 게이트 비용을 더함으로써 기존 연구 결과보다 LEA-128에서는 96개의 큐빗, LEA-256에서는 160개의 큐빗, LEA-256에서는 224개의 큐빗을 절약할

Algorithm 1 : Change $\delta[0]$ to $\delta[1]$

Input: $\delta_{0\sim31}$ $(0xc3efe9db)$
Output: $\delta_{0\sim31}$ $(0x44626b02)$
1: X gate(δ_0), X gate(δ_3)
2: X gate(δ_4), X gate(δ_6), X gate(δ_7)
3: X gate(δ_9)
4: X gate(δ_{15})
5: X gate(δ_{16}), X gate(δ_{18}), X gate(δ_{19})
6: X gate(δ_{23})
7: X gate(δ_{24}), X gate(δ_{25}), X gate(δ_{26})
8: X gate(δ_{31})
9: **return** $\delta[1]$ = {δ_0, δ_1,..., δ_{31}}

Alg. 1. Change $\delta[0]$ to $\delta[1]$

수 있었다. 기존 연구의 LEA-128 양자 회로는 Fig. 5와 같으며 제안하는 기법의 LEA-128 양자 회로는 Fig. 6과 같다. 이에 대한 자세한 비교는 4장에서 다시 살펴보고자 한다. Fig. 5, Fig. 6에서 키 스케줄의 로테이션 연산은 생략되었다.

4. 비교분석

라운드 함수와 키 스케줄을 병행하는 것과, 라운드 함수의 구현은 기존 기법과 동일하다. 하지만 기존 기법에서 키 스케줄에의 상수 값들을 큐빗으로 할당한 반면에, 제안 기법에서는 하나의 32큐빗 배열만을 할당한 뒤, 해당 공간의 값을 채워 넣고 변경해가는 방식으로 키 스케줄의 상수 덧셈을 수행하였다. 이를 통해 큐빗의 사용 개수를 획기적으로 줄였다. 제안 기법과 기존 기법의 비교결과는 Table 2와 같다. X 게이트의 증가와 큐빗 감소의 트레이드오프 문제에서 비용이 저렴한 X 게이트로 비용이 비싸고 많은 큐빗의 개수를 줄였기 때문에 제안하는 기법이 더 최적화 되었다고 평가할 수 있다. 제안하는 LEA 양자 회로의 양자 자원들은 IBM에서 제공하는 양자 프로그래밍 툴 ProjectQ[12]를 사용하여 측정하였다.

대칭키 암호에 그루버 알고리즘을 적용하기 위해서는 $r=$(키 크기/블록 크키)쌍의 암호문/평문이 필요하다. LEA 구현에 필요한 큐빗이 q개라고 할 때, $r\times q+1$의 큐빗이 필요하며 병렬 검색을 위해 $2\times(r-1)\times$키 크기의 CNOT 게이트가 추가로 요구된다. 최종적으로, 제안하는 기법을 통해 구현한 LEA에 그루버 알고리즘을 적용하기 위한 양자 자원은 〈Table 3〉과 같다. 그루버 알고리즘을 적용하는데 있어 오라클 함수에 대칭키 암호를 구현해야 한다. 그러므로 대상 암호를 양자 회로로 최적화하여 구현하는 부분이 제일 중요하다.

5. 결 론

최근 그루버 알고리즘 적용 대상인 블록암호를 양자 회로로 최적화 구현하는 연구가 다양하게 진행되고 있다. 현재 양

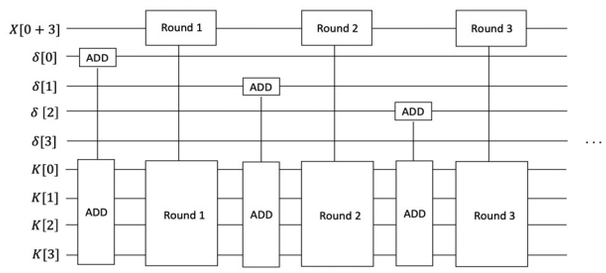

Fig. 5. Previous LEA-128 Quantum Circuit

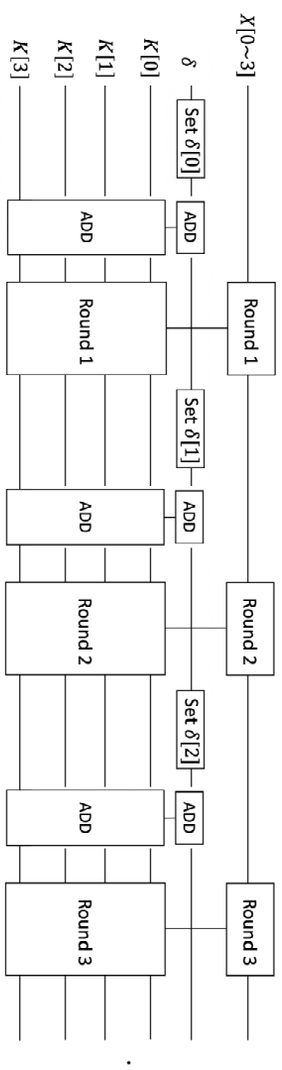

Fig. 6. Proposed LEA-128 Quantum Circuit

자 컴퓨터는 사용 가능한 큐빗의 개수를 늘리고 연산 시 생기는 오류율을 줄이는 과제에 직면해있다. 때문에 블록암호에 그루버 알고리즘을 적용하기 위해 필요한 큐빗 개수와 소모되는 양자 게이트의 수는 양자 컴퓨터의 공격에 대한 안전성의 지표가 될 수 있다. 이에 본 논문에서는 기존 LEA를 양자 회로로 구현한 기법과 비교하여 양자 게이트의 수는 증가하

Table 2. LEA Quantum Circuit Comparison

	Qubit	Toffoli	CNOT	X	Depth
LEA 128[11]	385	10,416	28,080	68	26,329
LEA 192[11]	513	15,624	39,816	100	39,453
LEA 256[11]	641	17,856	45,504	130	45,058
Proposed LEA 128	289	10,416	28,080	352	26,329
Proposed LEA 192	353	15,624	39,816	398	39,453
Proposed LEA 256	417	17,856	45,504	465	45,0,58

Table 3. Quantum Resources for Applying Grover's Algorithm to LEA

	r	Qubit	Toffoli	CNOT	X
LEA 128	1	290	20,832	56,160	704
LEA 192	2	707	62,496	40,200	1,592
LEA 256	2	835	71,424	45,760	1,860

였지만 큐빗의 사용을 감소시켰다. 제안하는 기법을 통해 매우 저렴한 X 게이트를 추가로 사용한 대신 LEA에 그루버 알고리즘을 적용하기 위한 큐빗의 개수를 최적화 할 수 있었다.

References

[1] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Computer Networks*, Vol.54, No.15, pp.2787-2805, 2010.

[2] D. Hong, J. K. Lee, D. C. Kim, D. Kwon, K. H. Ryu, and D. G. Lee, "LEA: A 128-bit block cipher for fast encryption on common processors," *in International Workshop on Information Security Applications*, Springer, pp.3-27, 2013.

[3] B. Koo, D. Roh, H. Kim, Y. Jung, D.G. Lee, and D. Kwon, "CHAM: A Family of Lightweight Block Ciphers for Resource-Constrained Devices," in *International Conference on Information Security and Cryptology (ICISC'17)*, 2017.

[4] D. Hong, et al., "HIGHT: A new block cipher suitable for low-resource device," in *International Workshop on Cryptographic Hardware and Embedded Systems*, Springer, pp.46-59, 2006.

[5] L. K. Grover, "A fast quantum mechanical algorithm for database search," in *Proceedings of the Twenty-eighth Annual ACM Symposium on Theory of Computing*, pp.212-219, 1996.

[6] M. Grassl, B. Langenberg, M. Roetteler, and R. Steinwandt, "Applying Grover's algorithm to AES: quantum resource estimates. Post-Quantum Cryptography," *PQCrypto 2016*, Springer, pp.29-43, 2016.

[7] B. Langenberg, H. Pham, and R. Steinwandt, "Reducing the cost of implementing AES as a quantum circuit," *Technical Report, Cryptology ePrint Archive*, Report 2019/854, 2019.

[8] S. Jaques, M. Naehrig, M. Roetteler, and F. Virdia, "Implementing Grover oracles for quantum key search on AES and LowMC," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Springer, pp.280-310, 2020.

[9] R. Anand, A. Maitra, and S. Mukhopadhyay, "Grover on SIMON," *arXiv:2004.10686*, 2020.

[10] K. B. Jang, S. J. Choi, H. D. Kwon, and H. J. Seo, "Grover on SPECK : Quantum Resource Estimates," *ePrint Archive, Report 2020/640*, 2020.

[11] K. B. Jang, S. J. Choi, H. D. Kwon, H. J. Kim, J. H. Park, and H. J. Seo, "Grover on Korean Block Ciphers," *Applied Sciences*, Vol.10, No.18, pp.6407, 2020.

[12] D.S . Steiger, T. Häner, and M. Troyer, "ProjectQ: An Open Source Software Framework for Quantum Computing," *arXiv:1612.08091*, 2016.

장 경 배

https://orcid.org/0000-0001-5963-7127
e-mail : starj1023@gmail.com
2019년 한성대학교 IT응용시스템공학부(학사)
2019년 ~ 현 재 한성대학교 IT융합공학부 석사과정
관심분야 : 정보보호, IoT, 양자컴퓨터

김 현 준

https://orcid.org/0000-0002-6847-6772
e-mail : khj930704@gmail.com
2019년 한성대학교 IT응용시스템공학부(학사)
2019년 ~ 현 재 한성대학교 IT융합공학부 석사과정
관심분야 : 시스템 보안, 부채널 분석, 블록체인

박 재 훈

https://orcid.org/0000-0003-1725-3621
e-mail : p9595jh@gmail.com
2020년 한성대학교 IT응용시스템공학부(학사)
2020년 ~ 현 재 한성대학교 IT융합공학부 석사과정
관심분야 : 웹 보안, 블록체인

송 경 주

https://orcid.org/0000-0001-4337-1843
e-mail : thdrudwn98@gmail.com
2019년 ~ 현 재 한성대학교 IT융합공학부 학사과정
관심분야 : 양자 컴퓨터, 정보 보안

서 화 정

https://orcid.org/0000-0003-0069-9061
e-mail : hwajeong84@gmail.com
2010년 부산대학교 컴퓨터공학과(학사)
2012년 부산대학교 컴퓨터공학과(석사)
2012년 ~ 2016년 부산대학교 컴퓨터공학과 (박사)
2016년 ~ 2017년 싱가포르 과학기술청 연구원
2019년 ~ 현 재 한성대학교 IT융합공학부 조교수
관심분야 : 정보보호, 암호화 구현, IoT