암호 양자보안 검증 플랫폼

⟨Q|Crypton⟩

2020.07.17

이석준

한국전자통신연구원

목 차

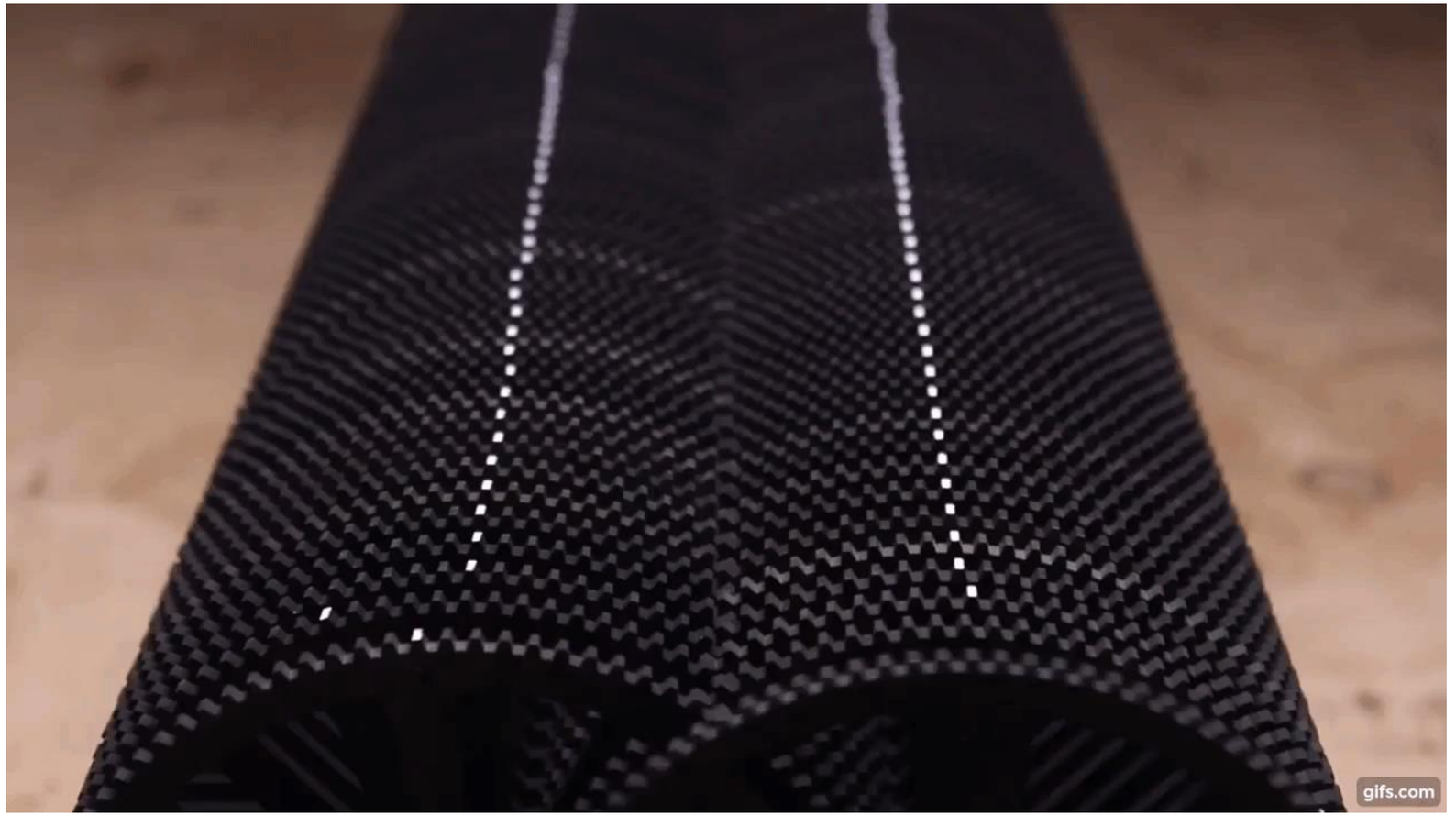
암호 안전성 검증 – 지수 증가 알고리즘 해결의 어려움

- 우주에서 가장 큰 기어 감속: Googol(10^{100}) to 1
  (https://www.youtube.com/watch?v=nFslB0AcVmM)

Daniel de Bruin의 톱니바퀴 장치 (앞바퀴가 10바퀴 돌 때 뒷바퀴가 1바퀴 돌도록 설계, 총 100개의 톱니바퀴)

암호 안전성 검증 – 지수 증가 알고리즘 해결의 어려움

- 우주에서 가장 큰 기어 감속: Googol(10^{100}) to 1
  (https://www.youtube.com/watch?v=nFslB0AcVmM)



암호 안전성 검증 – 지수 증가 알고리즘 해결의 어려움

우주에서 가장 큰 기어 감속: Googol(10^{100}) to 1

죽기 전에 마지막 바퀴가 도는 걸 볼 수 있을까?

첫번째 바퀴가 1초에 1바퀴 돈다면, 마지막 바퀴가 1바퀴 도는데 걸리는 시간은 **10^{99}초**

	1시간	1일	1년	100년 (인생)	10000년 (역사)	20만년 (현생인류)	45억년 (지구나이)	138억년 (우주나이)
초	3.6×10^3	8.64×10^4	3.15×10^7	3.15×10^9	3.15×10^{11}	7.3×10^{12}	1.42×10^{17}	4.35×10^{17}
1바퀴 이상 움직임	4번째 바퀴 (3.6회전)	5번째 바퀴 (8.64회전)	8번째 바퀴 (3.15회전)	10번째 바퀴 (3.15회전)	12번째 바퀴 (3.15회전)	13번째 바퀴 (7.3회전)	18번째 바퀴 (1.42회전)	18번째 바퀴 (4.35회전)
0.1도 이상 움직임	8번째 바퀴	9번째 바퀴	12번째 바퀴	14번째 바퀴	16번째 바퀴	17번째 바퀴	21번째 바퀴	22번째 바퀴

암호 안전성 검증 – RSA/ECC의 안전성

계산량의 지수 증가에 의존하는 암호 알고리즘 (Computational Security)

현존하는 컴퓨터로 죽기 전에 암호를 깰 수 있을까?

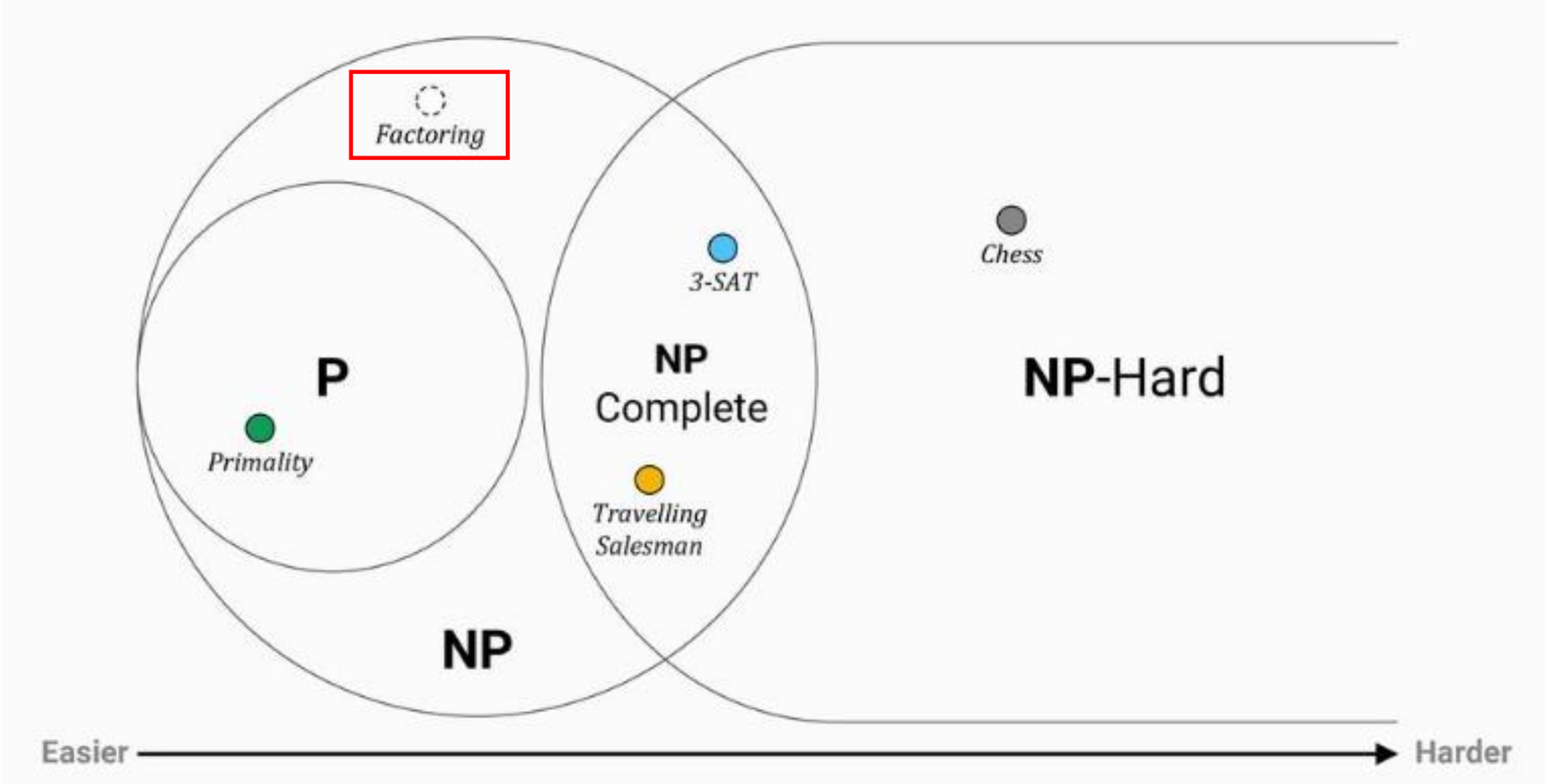
슈퍼컴퓨터 1위 (2020.06): Supercomputer Fugaku, 최대 513,854.7 Tflops (부동소수점 연산 초당 4.16×10^{17}회)

Security Level	# of Operation	Symmetric	RSA	ECC	Hash
80	2^{80} ($\approx 1.21 \times 10^{24}$)	TDEA	1024	160	SHA1
112	2^{112} ($\approx 5.19 \times 10^{33}$)	TDEA AES-128	2048	224	SHA-224
128	2^{128} ($\approx 3.40 \times 10^{38}$)	AES-128	3072	256	SHA-256
192	2^{192} ($\approx 6.28 \times 10^{57}$)	AES-192	7680	384	SHA-384
256	2^{256} ($\approx 1.16 \times 10^{77}$)	AES-256	15360	512	SHA-512

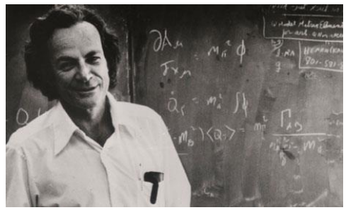
참고: NIST SP 800-57 Part 1 Revision 5, “Recommendation for Key Management: Part 1 – General”, 2020

암호 안전성 검증 – RSA/ECC의 안전성

RSA의 난이도 ≤ Factoring Problem (⊂NP, but ⊄P?)



미래 컴퓨팅 환경과 암호 – 양자 컴퓨터의 출현



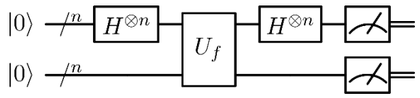
1981년 Richard Feynman
양자 컴퓨터 제안

1985년 David Deutsch
Deutsch 양자알고리즘
(최초의 지수적 성능 개선)

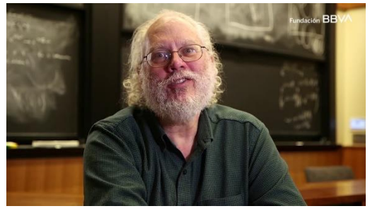
1993년 Bernstein & Vazirani
Quantum-Classical Separation 증명
(비결정적 양자 이득 존재)

양자 프로세서 및 컴퓨터 개발의 Motivation : (암호) 양자 알고리즘

1994년 Daniel Simon
Simon 알고리즘 제안
(주기알고리즘의 양자성능개선)



1994년 Peter Shor
Shor 알고리즘 제안
(다항시간 Factoring 문제해결)



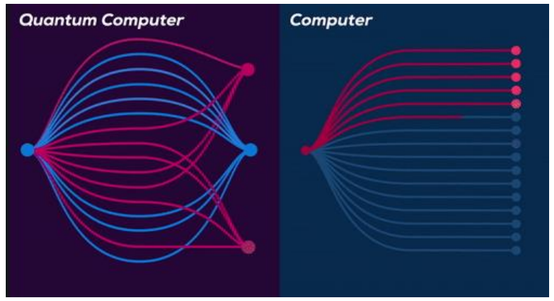
1996년 Lov Grover
Grover 알고리즘 제안
(검색알고리즘의 양자성능개선)

미래 컴퓨팅 환경과 암호 – 양자 컴퓨터의 출현

Quantum Parallelism: 양자 컴퓨팅은 왜 고전 컴퓨터 대비 빠른가?

양자컴퓨팅 기본 단위 : 큐비트 → 0과 1이 동시에 존재(중첩)

	
$0 \quad 1$	$\alpha\|0\rangle + \beta\|1\rangle$
비트 단위	큐비트 단위



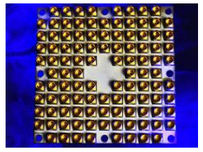
중첩된 상태(큐비트)에서 연산 가능, 관측시 하나의 상태(고전비트)로 붕괴

- 양자계산 n비트 데이터 2^n 개 동시에 표현 → 동시에 연산 → 월등한 계산능력
- 모든 문제(특히, NP)에 대해 지수적 속도 향상은 아니나, 암호해독(BQP)에는 치명적

미래 컴퓨팅 환경과 암호 – 양자 컴퓨팅

1 미래컴퓨팅 환경 변화 → 양자컴퓨팅의 가시화

- 큐비트칩 개발 경쟁



<'18. 인텔 Tangle Lake(Q49)>

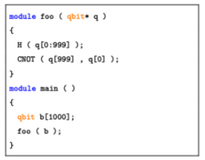
<'18. 구글 Bristlecone(Q72)>

<'19. IBM 양자컴퓨터(Q53)>

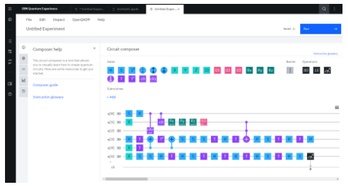
<'19. IBM Q System One(Q20)>

- 양자컴퓨팅 플랫폼 연구 (Q프로그래밍, Q컴파일러, QEC 등)

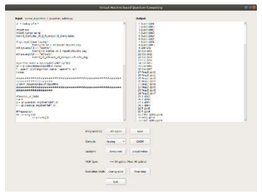
<MS Q#>



<프린스턴대학 ScaffCC>



<IBM Q Experience>



<ETRI 양자플랫폼 (Type-I QC)>

- 양자 우월성(Quantum Supremacy) 시대 진입 ('19.10.23 Nature 저널 발표 - 구글 및 UCSB, 200sec vs. 10k-year)

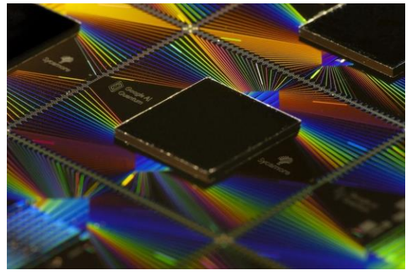
nature

Article | Published: 23 October 2019

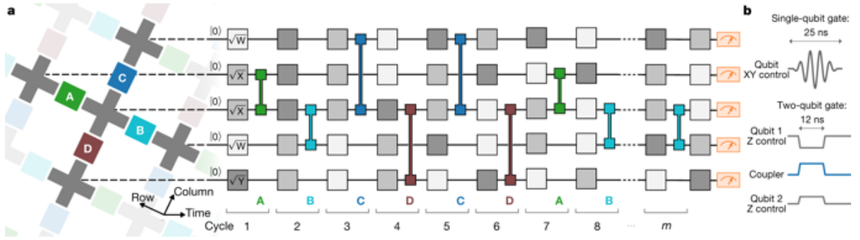
Quantum supremacy using a programmable superconducting processor

Frank Arute, Kunal Arya, [...] John M. Martinis

<Nature 발표 논문>



<'19. 구글 Sycamore 양자 프로세서 (Q54)>



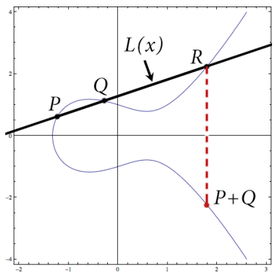
<양자 우월성 회로를 위한 제어 동작 (53Qubit, 1500+QGate)>

미래 컴퓨팅 환경과 암호 – 양자 컴퓨팅

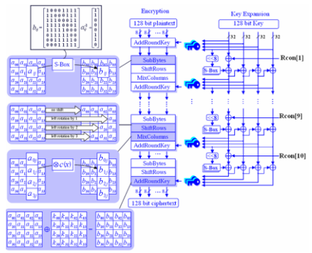
2 양자컴퓨팅에 의한 미래 위협 → 현존 암호 붕괴

- ICT 보안에 사용되는 대표 암호들

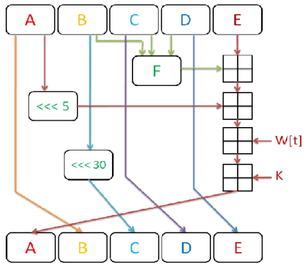
<RSA(Rivest-Shamir-Adleman)>



<ECC(Elliptic Curve Cryptography)>

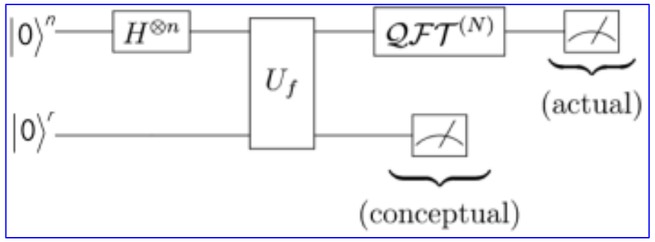


<AES 등 대칭키암호>

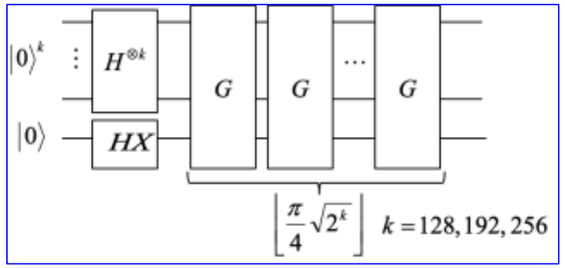


<SHA1 등 해시함수>

- Shor 양자알고리즘 → RSA, ECC 다항식 시간 해독



- Grover 양자알고리즘 → AES/해시 전수 공격 속도 증가



- Simon 양자알고리즘 → 일부 대칭키 취약점 제기

영국 GCHQ와 미국 NIST에서는 2030년경에 암호해독 전용 양자컴퓨터 등장이 가능할 것으로 전망
(GCHQ whitepaper, Quantum-safe Cryptography, '16.11)

미래 컴퓨팅 환경과 암호 – 양자 컴퓨팅

3 이에 대한 대비 → NIST 중심의 양자내성암호(PQC) 표준화 진행 중

- 2017.11 : 82 개 양자내성암호 후보 접수 → 64개 후보
- 2019.01 : 2차 라운드 후보 선정 → 26개 PQC 후보

Second Round Candidates — PKE/KEM — D. Sign

C BIKE	C LEDAcrypt	M Rainbow
C Classic McEliece	M LUOV	C ROLLO
L CRYSTALS-DILITHIUM	M MQDSS	L Round5
L CRYSTALS-KYBER	L NewHope	C L RQC
L FALCON	L NTRU	L SABER
L FrodoKEM	L NTRU Prime	I SIKE
M GeMSS	C NTS-KEM	H SPHINCS+
C HQC	H Picnic	L Three Bears
L LAC	L qTESLA	

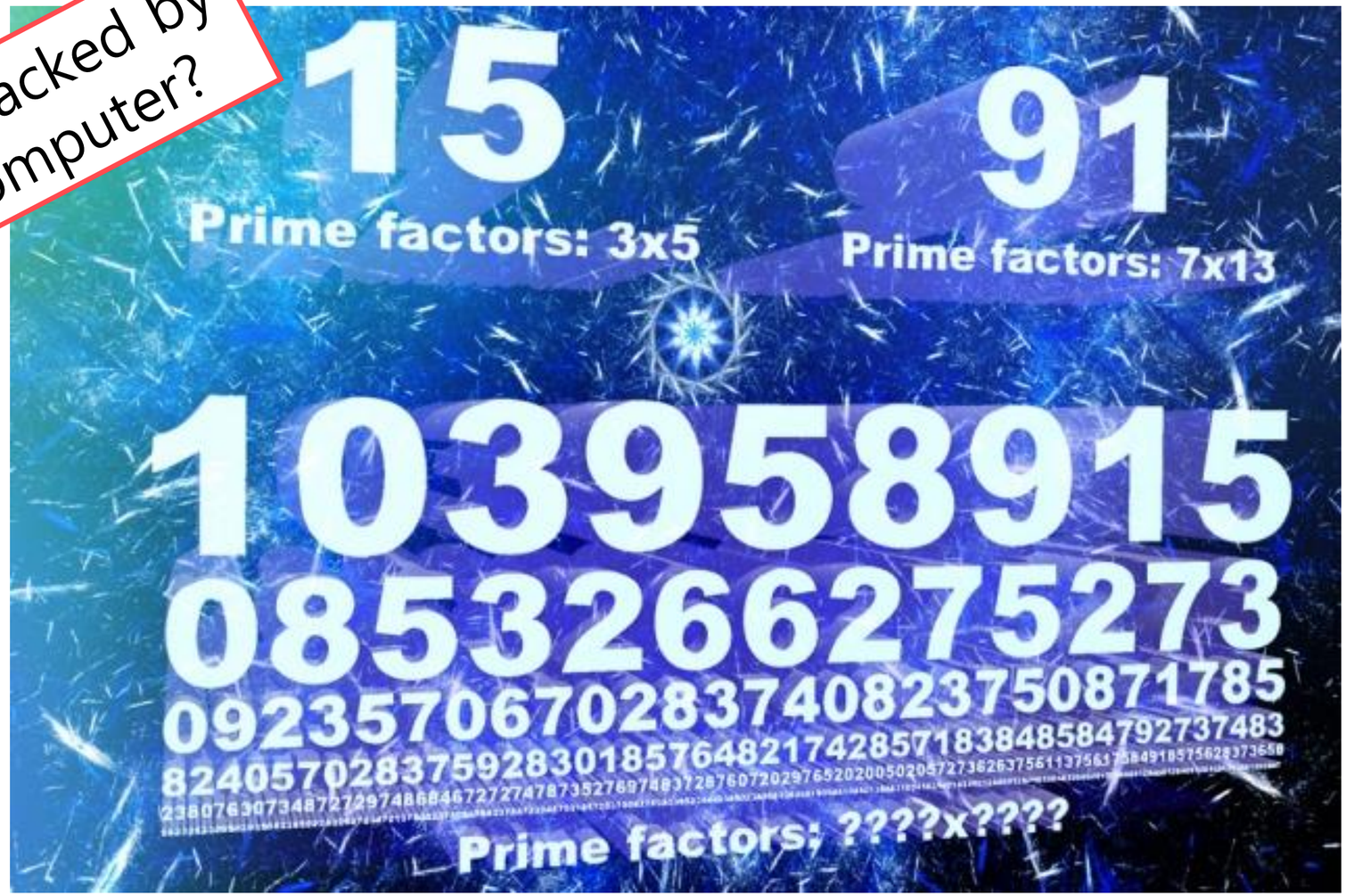
L Lattice기반 C Code기반 M MQ기반 H 해시기반 I Isogeny기반

- 공개키암호/키설립 (17개), 전자서명 (9개)
- Lattice기반 (13개), Code기반 (7개), MQ기반 (4개), 해시기반 (2개), Isogeny기반 (1개)

- NIST: National Institute of Standards and Technology, 미국 국립표준기술연구소
- PQC: Post-Quantum Cryptography, 양자내성암호

미래 컴퓨팅 환경과 암호 – 양자 컴퓨팅 환경에서의 암호 안전성

Is **RSA** really cracked by Quantum Computer?



이미지 출처: Jose-Luis Olivares/MIT

미래 컴퓨팅 환경과 암호 – 양자 컴퓨팅 환경에서의 암호 안전성

ECDLP 및 Factoring 문제를 풀기 위한 이론적 양자 성능

ECDLP in $E(\mathbb{F}_p)$ simulation results					Factoring of RSA modulus N interpolation from [21]		
$\lceil \log_2(p) \rceil$ bits	#Qubits	#Toffoli gates	Toffoli depth	Sim time sec	$\lceil \log_2(N) \rceil$ bits	#Qubits	#Toffoli gates
110	1014	$9.44 \cdot 10^9$	$8.66 \cdot 10^9$	273	512	1026	$6.41 \cdot 10^{10}$
160	1466	$2.97 \cdot 10^{10}$	$2.73 \cdot 10^9$	711	1024	2050	$5.81 \cdot 10^{11}$
192	1754	$5.30 \cdot 10^{10}$	$4.86 \cdot 10^{10}$	1 149	–	–	–
224	2042	$8.43 \cdot 10^{10}$	$7.73 \cdot 10^{10}$	1 881	2048	4098	$5.20 \cdot 10^{12}$
256	2330	$1.26 \cdot 10^{11}$	$1.16 \cdot 10^{11}$	3 848	3072	6146	$1.86 \cdot 10^{13}$
384	3484	$4.52 \cdot 10^{11}$	$4.15 \cdot 10^{11}$	17 003	7680	15362	$3.30 \cdot 10^{14}$
521	4719	$1.14 \cdot 10^{12}$	$1.05 \cdot 10^{12}$	42 888	15360	30722	$2.87 \cdot 10^{15}$

Table 2: Resource estimates of Shor's algorithm for computing elliptic curve discrete logarithms in $E(\mathbb{F}_p)$ versus Shor's algorithm for factoring an RSA modulus N.

출처: Martin Roetteler et al, "Quantum Resource Estimates for Computing Elliptic Curve Discrete Logarithms, 2017

미래 컴퓨팅 환경과 암호 – 양자 컴퓨팅 환경에서의 암호 안전성

이론과 실제, 그리고 Shor Algorithm

How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits

Craig Gidney[1,*] and Martin Ekerå[2]

[1] *Google Inc., Santa Barbara, California 93117, USA*
[2] *KTH Royal Institute of Technology, SE-100 44 Stockholm, Sweden*
Swedish NCSA, Swedish Armed Forces, SE-107 85 Stockholm, Sweden

(Dated: December 6, 2019)

We significantly reduce the cost of factoring integers and computing discrete logarithms in finite fields on a quantum computer by combining techniques from Shor 1994, Griffiths-Niu 1996, Zalka 2006, Fowler 2012, Ekerå-Håstad 2017, Ekerå 2017, Ekerå 2018, Gidney-Fowler 2019, Gidney 2019. We estimate the approximate cost of our construction using plausible physical assumptions for large-scale superconducting qubit platforms: a planar grid of qubits with nearest-neighbor connectivity, a characteristic physical gate error rate of 10^{-3}, a surface code cycle time of 1 microsecond, and a reaction time of 10 microseconds. We account for factors that are normally ignored such as noise, the need to make repeated attempts, and the spacetime layout of the computation. When factoring 2048 bit RSA integers, our construction's spacetime volume is a hundredfold less than comparable estimates from earlier works (Fowler et al. 2012, Gheorghiu et al. 2019). In the abstract circuit model (which ignores overheads from distillation, routing, and error correction) our construction uses $3n + 0.002n \lg n$ logical qubits, $0.3n^3 + 0.0005n^3 \lg n$ Toffolis, and $500n^2 + n^2 \lg n$ measurement depth to factor n-bit RSA integers. We quantify the cryptographic implications of our work, both for RSA and for schemes based on the DLP in finite fields.

미래 컴퓨팅 환경과 암호 – 양자 컴퓨팅 환경에서의 암호 안전성

이론과 실제, 그리고 Shor Algorithm

현재 4098큐비트 지원 양자컴퓨터가 존재하지 않음!

RSA-2048을 깨기 위한 최소한의 큐비트 수 : 4098~4099 큐비트
큐비트수 1위 구글 Bristlecone : 72큐비트

4098큐비트 지원 양자컴퓨터만 만들면 RSA-2048 알고리즘은 깨질까?

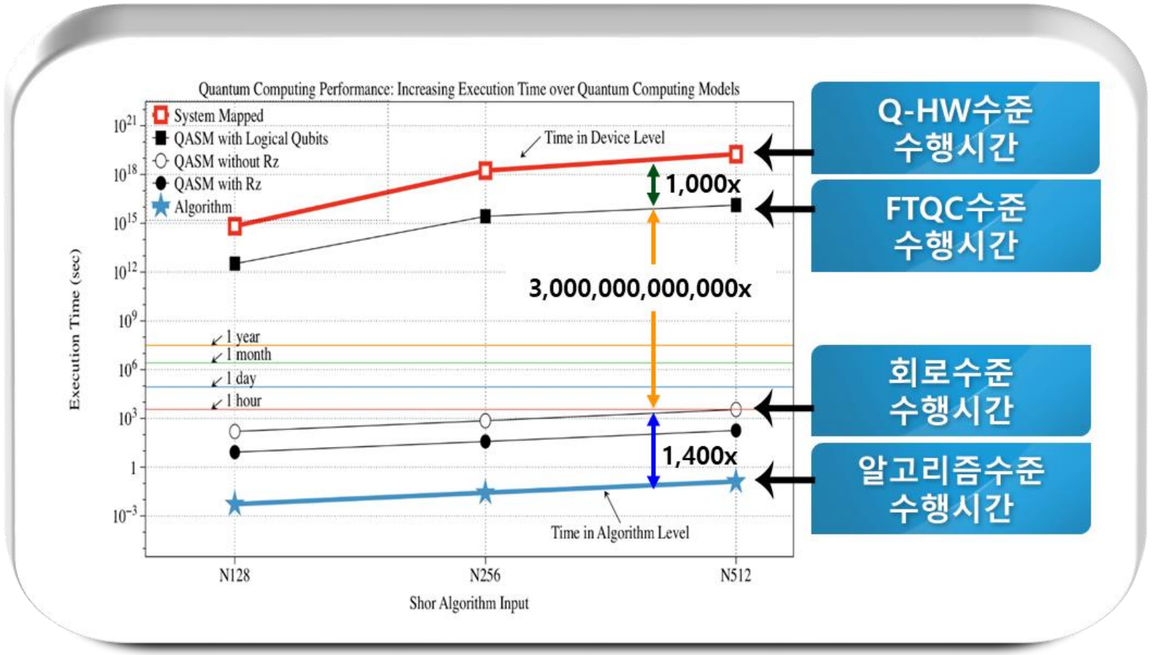
- 일반적인 양자 알고리즘과 같이 양자 프로세서에서도 큐비트간 임의 연산이 가능할까?
- 임의의 n큐비트 양자 게이트 구현이 가능할까?
- 구글 Bristlecone(에러비율: 0.6%) 등 기존 양자 프로세서가 가진 에러를 어떻게 극복할 수 있을까?
- Coherence Time(큐비트 양자 특성 유지시간, 현재 50~90ms)을 얼마나 길게 유지할 수 있을까?

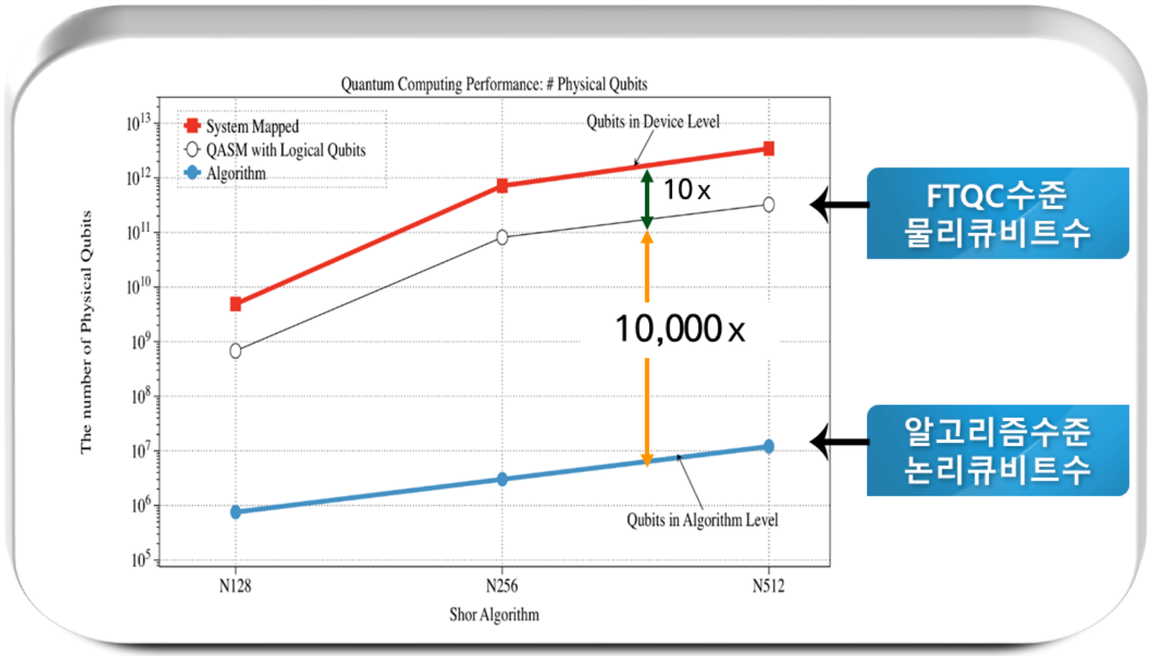
미래 컴퓨팅 환경과 암호 – 양자 컴퓨팅 환경에서의 암호 안전성

이론과 실제, 그리고 Shor Algorithm

실제는 이론 대비 대략 10,000배 이상의 큐비트를 요구할 수 있음

(Shor 알고리즘(512큐비트)에 대해 FTQC 기술 적용시)

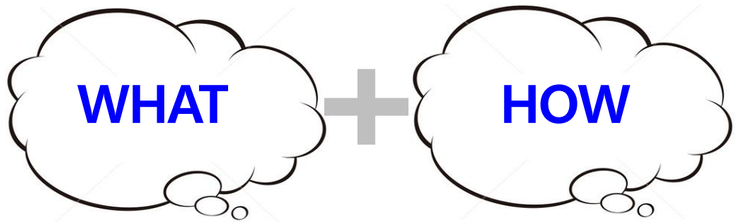




미래 컴퓨팅 환경과 암호 – 무엇이 필요한가?

실제 양자 컴퓨팅 환경에 대비한 암호 안전성 연구의 필요성

- 암호 양자분석 알고리즘 연구 (이론적 안전성 분석)
- 암호 양자분석 알고리즘의 효율적 구현/검증 (암호 엔지니어링)
- 암호 양자 안전성에 대한 정량적 분석 (실제 양자 환경 반영)



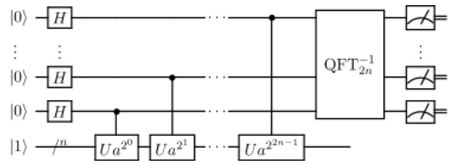
고전(AES, RSA, ECC 등) 및 PQC 암호들에 대한
양자보안강도 검증 플랫폼이 존재한다면?

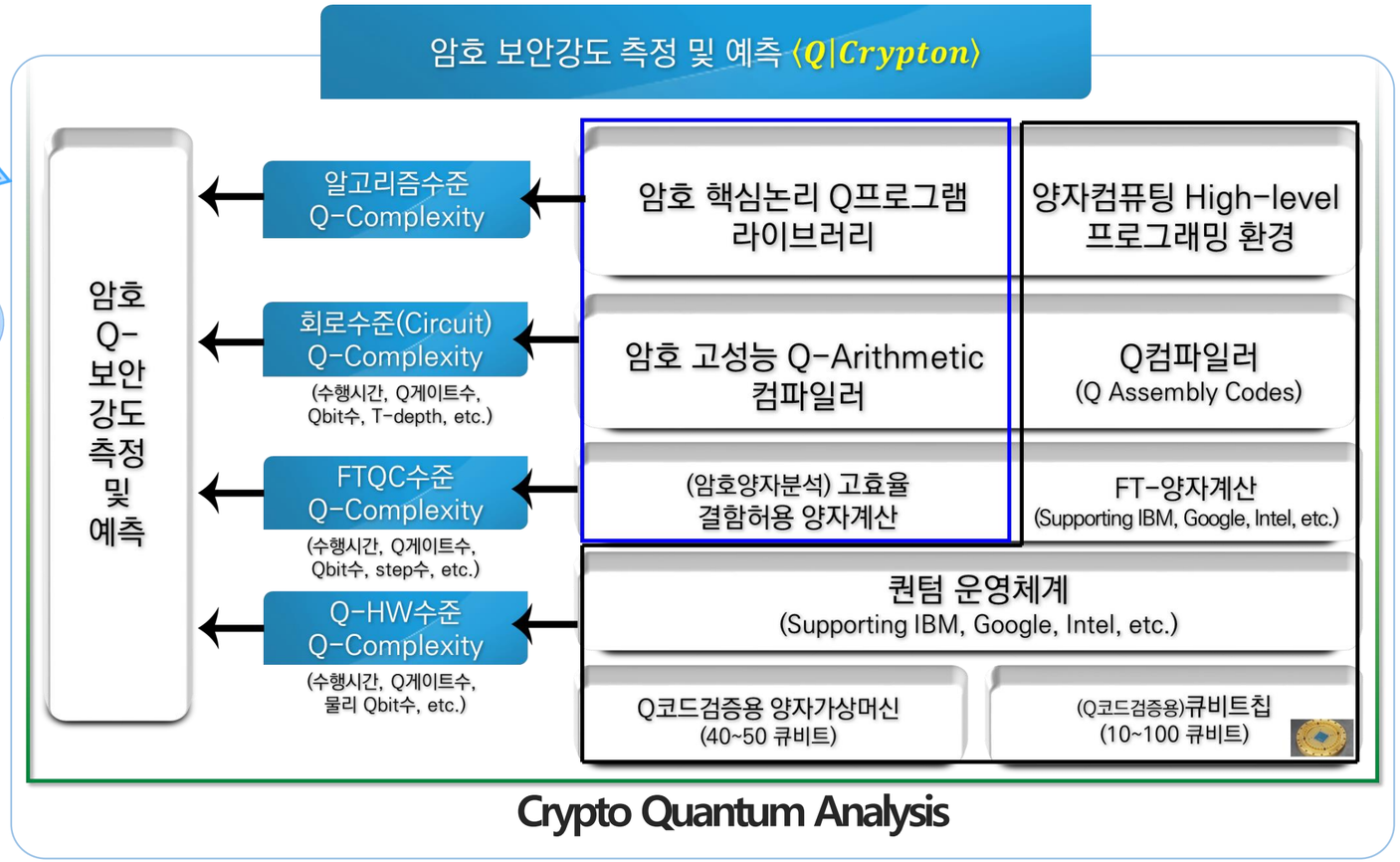
⟨Q|Crypton⟩ 플랫폼 – 기술 개념

⟨Q|Crypton⟩ 개념

현존 암호 (대칭키, RSA, ECC 등) 및 **다양한 차세대 암호**(PQC 암호 등)에 대한 미래컴퓨팅(즉, **대용량 양자컴퓨팅**) 환경에서 **정량적 계산복잡도*** **분석**을 통한 **암호의 안전성(보안강도) 측정 및 예측 기술**

- **암호 Quantum Analysis 알고리즘**





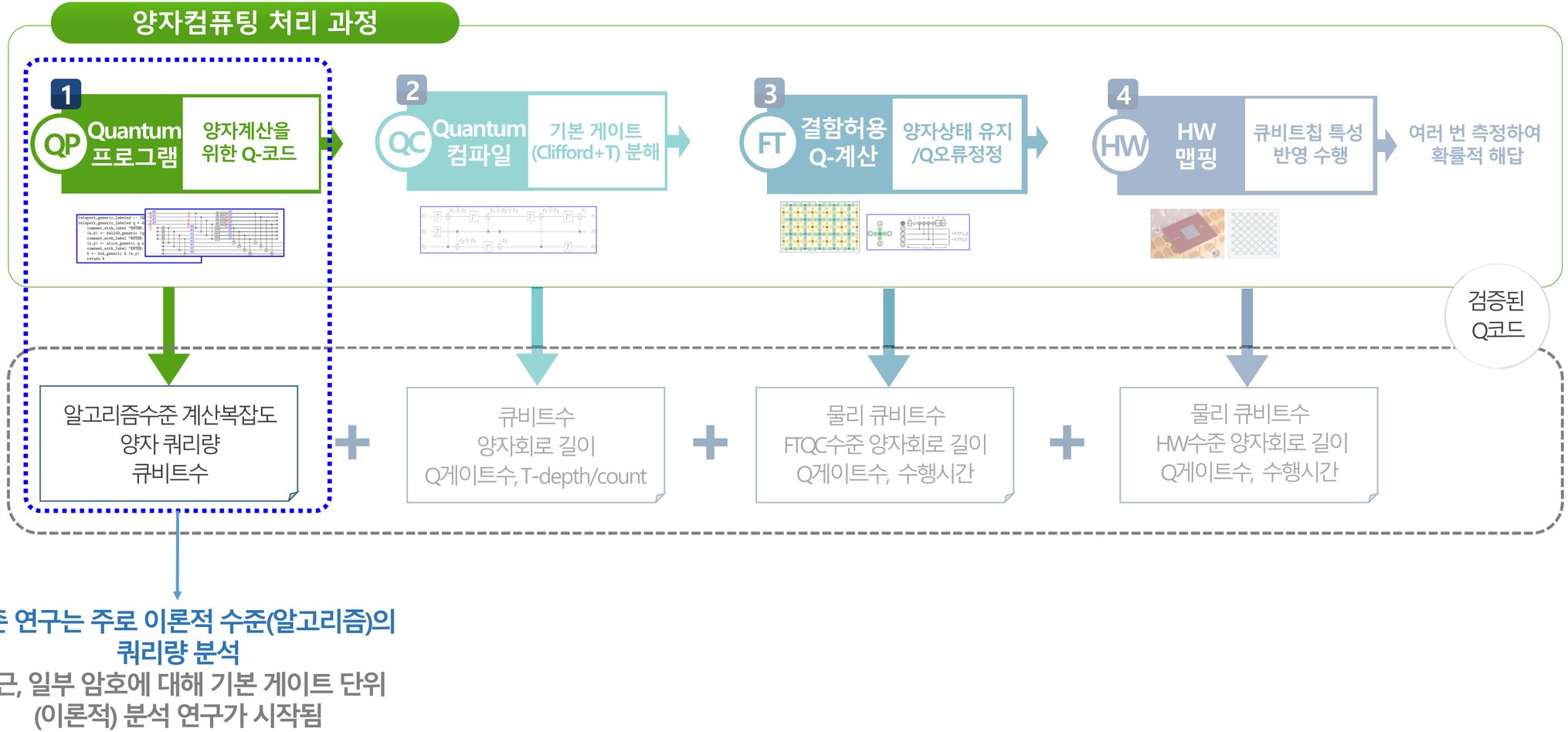
정량적 계산복잡도

양자분석 알고리즘 수준의 계산복잡도와 같은 기본적인 계산량 뿐만 아니라, 알고리즘이 수행되기 위한 양자컴퓨터 상의 전 과정에서 산출될 수 있는 전체 자원 투입량

(논리큐비트 수, 양자회로수준 양자게이트 수, 양자회로수준 논리큐비트 수, 양자회로수준 수행시간, 결함허용수준 양자게이트수, Q-HW수준 물리큐비트수, etc)

$\langle Q|Crypton\rangle$ 플랫폼 – What & How?

정량적 계산복잡도 기반 양자보안강도 검증 의미

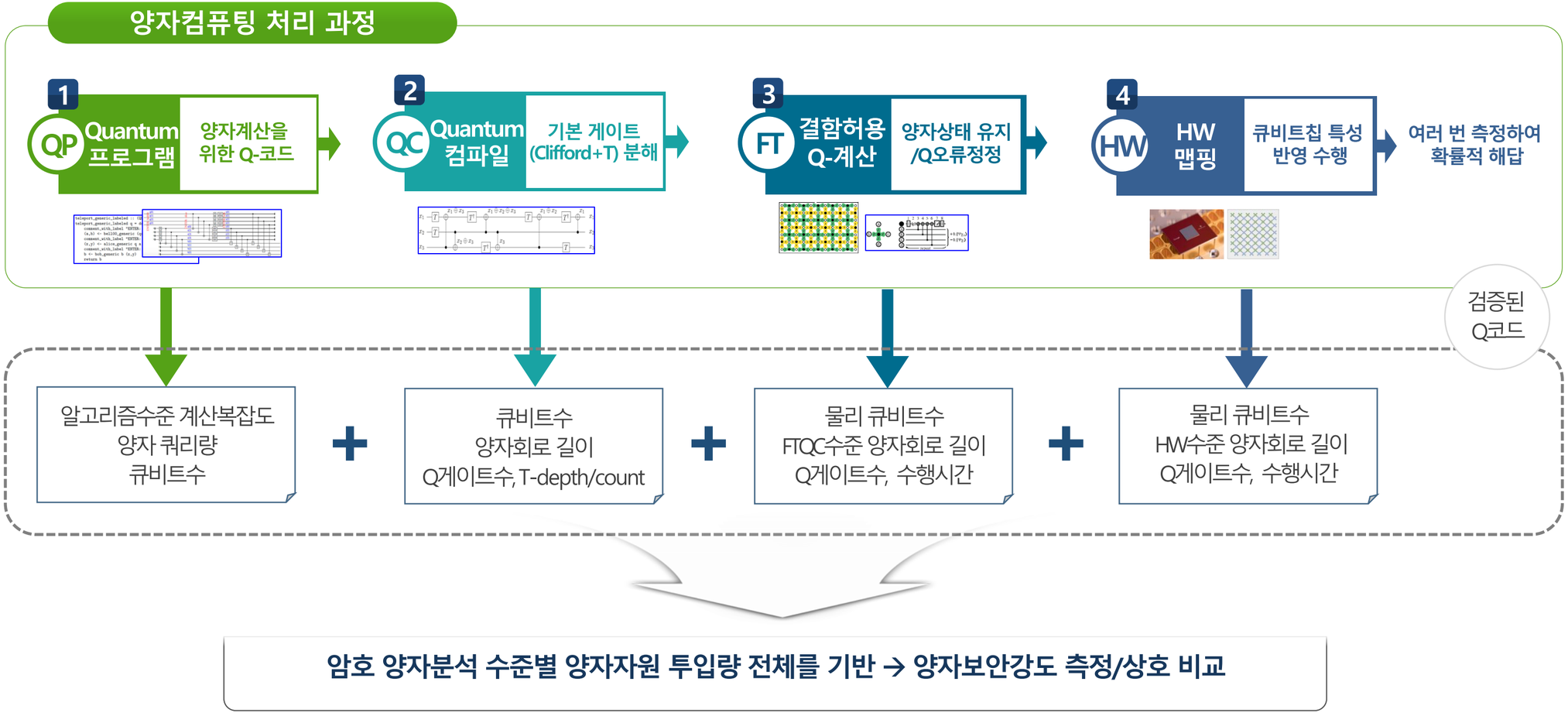


기존 연구는 주로 이론적 수준(알고리즘)의 쿼리량 분석

최근, 일부 암호에 대해 기본 게이트 단위 (이론적) 분석 연구가 시작됨

⟨Q|Crypton⟩ 플랫폼 – What & How?

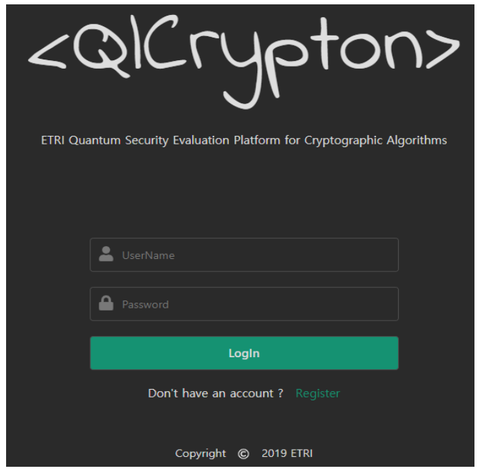
정량적 계산복잡도 기반 양자보안강도 검증 의미



⟨Q|Crypton⟩ 플랫폼 – What & How?

⟨Q|Crypton⟩ : 웹 기반 암호 양자분석 플랫폼 (양자 개발환경, 컴퓨팅, 분석 통합)

ETRI ⟨Q|Crypton⟩ 플랫폼



양자 컴파일러 (양자 어셈블리 코드 생성)

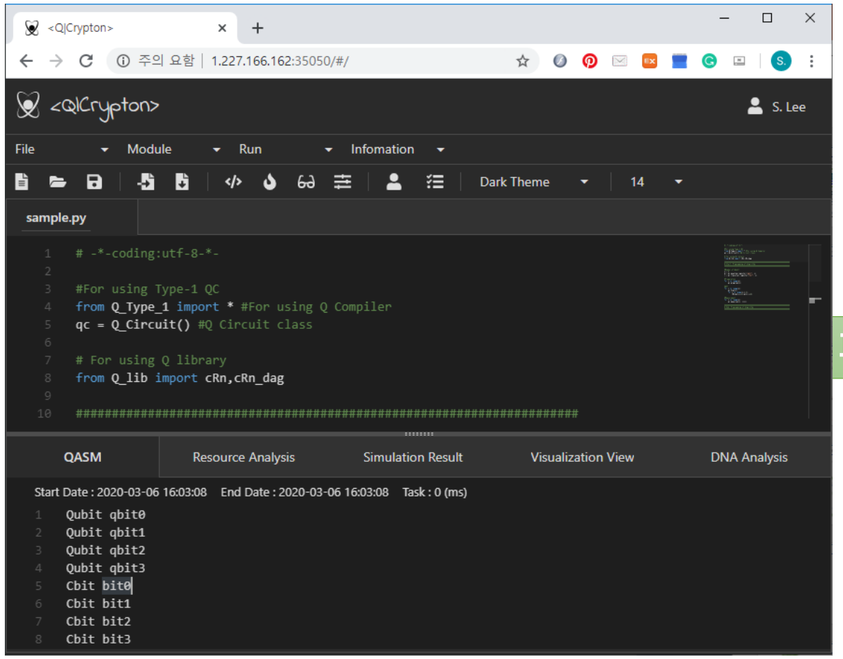
가상머신 기반 양자컴퓨팅 플랫폼 (Q40)

양자회로 · 시스템 합성 및 성능평가 시스템

- 웹 UI 기반 암호 Q-code 개발시험환경 구축 (40큐비트 양자가상머신)
  - 웹 기반 암호 Q-code 개발 전용 에디터 및 환경
  - 암호 양자 라이브러리 등록 및 호출
  - 양자 컴파일러, 가상머신, 성능평가 시스템 연동 및 QASM 기반 양자회로 시각화
- 다중사용자 시험을 위한 사용자/작업 관리

⟨Q|Crypton⟩ 플랫폼 – What & How?

⟨Q|Crypton⟩ : 웹 기반 암호 양자분석 플랫폼



파이썬 기반 양자 코드

양자 컴파일러

가상머신 실행을
통한 양자코드검증
(small scale)

Qubit qbit0
Qubit qbit1
Qubit qbit2
Qubit qbit3
PrepZ qbit0
PrepZ qbit1
PrepZ qbit2
PrepZ qbit3
H qbit0
CNOT qbit0,qbit1
T qbit0
Tdag qbit1
...

가상머신 기반
양자컴퓨팅 (Q40)

양자알고리즘
정량적 분석
(large scale)

양자회로·시스템 합성
및 성능평가 시스템

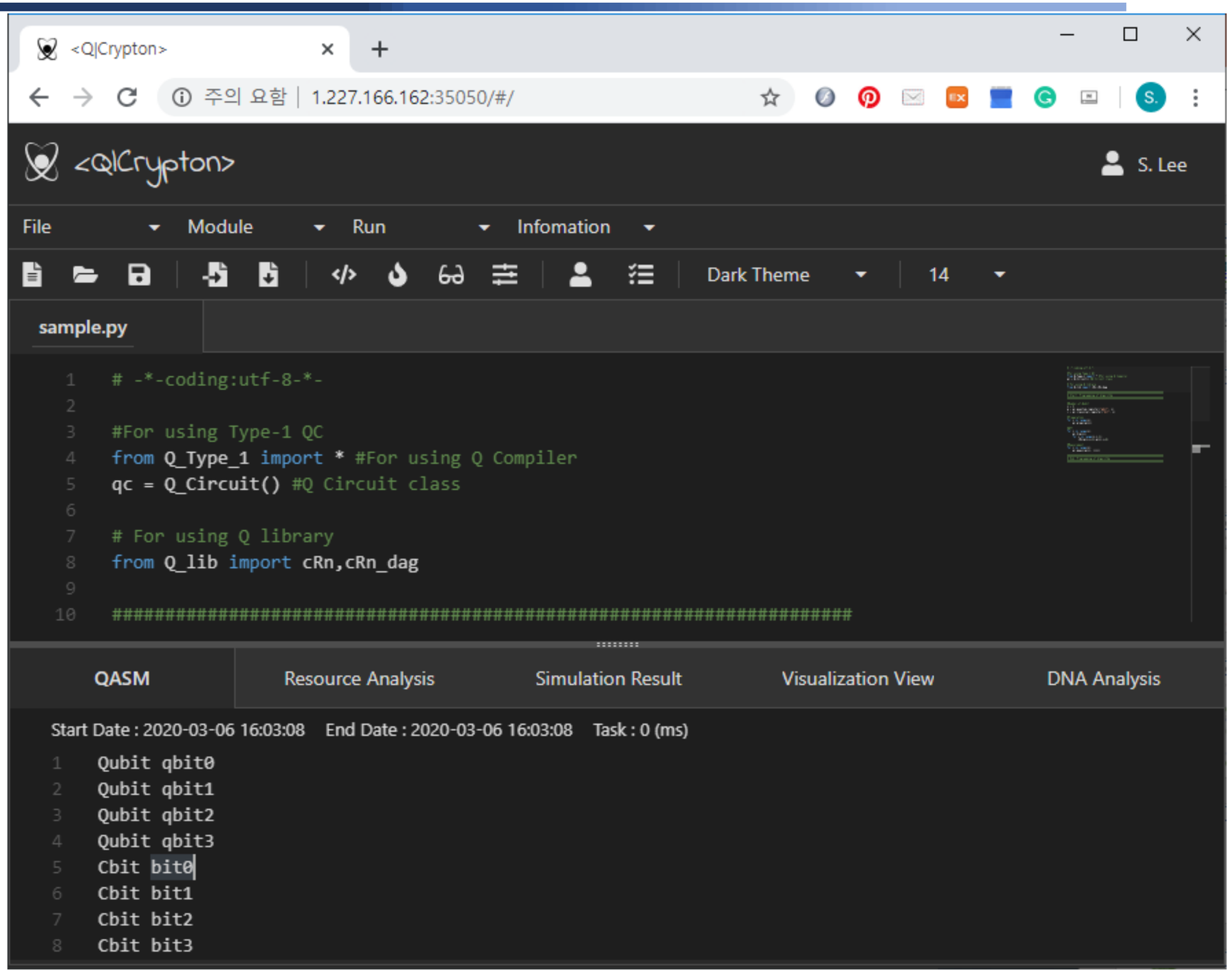
정량적 분석의 대상인 Large Scale 코드가 정확히 구현되었음을 어떻게 확인할 수 있을까?

(죽기 전에 마지막 바퀴가 도는 걸 볼 수 있을까?)

검증된 Small Scale 양자프로그램(ex. RSA-16)을 이용한
양자자원 분석용 Large Scale 양자프로그램(ex. RSA-2048)의 간접 검증 방법 필요

⟨Q|Crypton⟩ 플랫폼 – 현재

현재 개발환경



⟨Q|Crypton⟩ 플랫폼 – 현재

현재 개발환경 (파이썬 기반 양자코드)

```
# -*-coding:utf-8-*-

#For using Type-1 QC
from Q_Type_1 import * #For using Q Compiler
qc = Q_Circuit() #Q Circuit class

# For using Q library
from Q_lib import cRn,cRn_dag

################################################################################
# Start: Programming of Algorithm
################################################################################

#Number_of_Qubit
n = 4
q = qc.quantum_register("qbit", n)
c = qc.classical_register("bit", n)

#Preparation
for i in range(n):
    qc.prepz(q[i])

#QFT
for i in range(n):
    qc.h(q[i])
    for j in range(n-i-1):
        cRn(q[i+j+1],q[i],j+2)

#Measurement
for i in range(n):
    qc.measz(q[i], c[i])

################################################################################
# End: Programming of Algorithm
################################################################################
```

⟨Q|Crypton⟩ 플랫폼 – 현재

현재 개발환경 (파이썬 기반 양자코드)

```
#For using Type-1 QC
from Q_Type_1 import * #For using Q Compiler
qc = Q_Circuit() #Q Circuit class

# For using Q library
from Q_lib import cRn,cRn_dag

#Number_of_Qubit
n = 4
q = qc.quantum_register("qbit", n)
c = qc.classical_register("bit", n)

#Preparation
for i in range(n):
    qc.prepz(q[i])

#QFT
for i in range(n):
    qc.h(q[i])
    for j in range(n-i-1):
        cRn(q[i+j+1],q[i],j+2)

#Measurement
for i in range(n):
    qc.measz(q[i], c[i])

########################################################################
# End: Programming of Algorithm
########################################################################
```

Type-I 클래스 모듈 Import 및 클래스 선언

Q_lib 모듈 Import (cRn, cRn_dag 함수)

큐비트 및 고전비트 선언 (각 4비트)

전체 큐비트 Preparation

QFT 로직 구현

전체 큐비트 Measurement

$\langle Q|Crypton\rangle$ 플랫폼 – 현재

현재 개발환경 (지원 양자게이트)

❑ 기본 양자게이트 (Universal)

- X, Y, Z, S, H, T, Rz
- CNOT
- PrepZ, MeasZ

❑ 확장 양자게이트 (라이브러리)

- cRn, ccRn (및 cRnd, ccRnd)
- Toffoli (CCNOT)
- QFT

⟨Q|Crypton⟩ 플랫폼 – 현재

현재 개발환경 (컴파일 결과 - QASM)

Start Date : 2020-02-26 21:10:27 End Date : 2020-02-26 21:10:27 Task : 0 (ms)

```
Qubit qbit0
Qubit qbit1
Qubit qbit2
Qubit qbit3
Cbit bit0
Cbit bit1
Cbit bit2
Cbit bit3
PrepZ qbit0
PrepZ qbit1
PrepZ qbit2
PrepZ qbit3
H qbit0
T qbit1
T qbit0
CNOT qbit1,qbit0
Tdag qbit0
CNOT qbit1,qbit0
Rz qbit2,0.392699
Rz qbit0,0.392699
CNOT qbit2,qbit0
Rz qbit0,-0.392699
CNOT qbit2,qbit0
Rz qbit3,0.19635
Rz qbit0,0.19635
CNOT qbit3,qbit0
Rz qbit0,-0.19635
CNOT qbit3,qbit0
H qbit1
T qbit2
T qbit1
CNOT qbit2,qbit1
Tdag qbit1
CNOT qbit2,qbit1
Rz qbit3,0.392699
Rz qbit1,0.392699
CNOT qbit3,qbit1
Rz qbit1,-0.392699
CNOT qbit3,qbit1
H qbit2
T qbit3
T qbit2
CNOT qbit3,qbit2
Tdag qbit2
CNOT qbit3,qbit2
H qbit3
MeasZ qbit0 -> bit0
MeasZ qbit1 -> bit1
MeasZ qbit2 -> bit2
MeasZ qbit3 -> bit3
```

⟨Q|Crypton⟩ 플랫폼 – 현재

현재 개발환경 (가상머신 실행 결과)

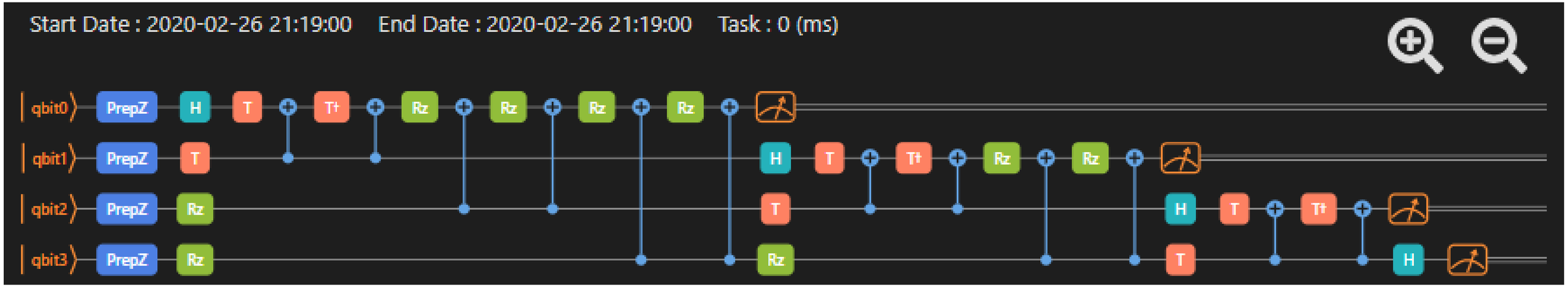
```
Start Date : 2020-02-26 21:12:19    End Date : 2020-02-26 21:12:20    Task : 1000 (ms)
MeasZ qbit0 -> bit0
+(0.35355+0.0i)|0000>
+(0.35355+0.0i)|0001>
+(0.35355+0.0i)|0010>
+(0.35355+0.0i)|0011>
+(0.35355-0.0i)|0100>
+(0.35355-0.0i)|0101>
+(0.35355-0.0i)|0110>
+(0.35355-0.0i)|0111>
MeasZ qbit1 -> bit1
+(0.5+0.0i)|0000>
+(0.5+0.0i)|0001>
+(0.5+0.0i)|0010>
+(0.5+0.0i)|0011>
MeasZ qbit2 -> bit2
+(0.70711+0.0i)|0010>
+(0.70711+0.0i)|0011>
MeasZ qbit3 -> bit3
+(1.0+0.0i)|0010>
Final State
+(1.0+0.0i)|0010>
Measurement Outcome
[0, 0, 1, 0]
    Start Time: 2020-02-26 21:12:20.111785
      End Time: 2020-02-26 21:12:20.114191
       Runtime: 0:00:00.002406
Memory Usage: 0.0 (MB)
```

0번 큐비트 Measurement 직후

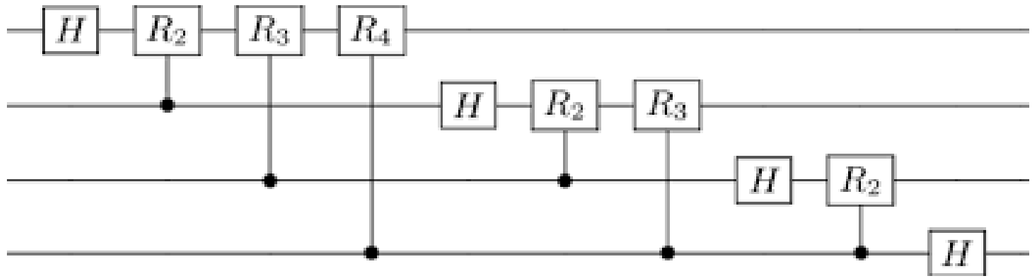
최종 관측 결과 (실행할 때마다 바뀜)

$\langle Q|Crypton\rangle$ 플랫폼 – 현재

현재 개발환경 (QASM 시각화 양자 회로)



참고: QFT 로직 알고리즘



⟨Q|Crypton⟩ 플랫폼 – 현재

현재 개발환경 (양자분석 결과)

Performance Analysis in Compile (Algorithm) Level		
No.	Item	Value
1	Algorithm Qubits	4 units
2	Computing Cycles	24 steps
3	KQ	96 units x steps
4	Total Gates	42

1. Algorithm Qubit: The quantity of qubits described in the user algorithm

2. Computing Cycles: The circuit depth composed by the compiled algorithm

3. KQ: One method to calculating the circuit cost (KQ = #Algorithm Qubits x Computing Cycles)

4. Total Gates: The quantity of the compiled (decomposed) quantum gates

Performance Analysis in FTQC System Level		
No.	Item	Value
1	Algorithm Qubits	4
2	Circuit Depth	34
3	Code Distance	3
4	Compuing Time	6.0005
5	KQ	136
6	Logical Gates	56
7	Physical Qubits	612
8	Time Overhead (Qubit Movements)	6

1. Algorithm: The name of the user algorithm

2. building_block: The information about a logical qubit encoded in quantum error-correcting code

3. global_layout: The layout for inter-modules

4. local_layout: The layout for inter-qubits of each module

$\langle Q|Crypton\rangle$ 플랫폼 – 현재, 그리고…

$\langle Q|Crypton\rangle$ 특장점

- ETRI 순수 국내 기술로 개발된 양자분석 플랫폼
- 파이썬 기반의 직관적 개발 환경을 통한 손쉬운 양자 프로그래밍
- 웹 기반 개발 환경 (별도 설치 프로그램 없음)
- FTQC 적용 및 실제 양자컴퓨팅 시스템 기반 양자자원 정량적 분석

$\langle Q|Crypton\rangle$ 플랫폼 – 현재, 그리고…

$\langle Q|Crypton\rangle$ 특장점?

"암호 양자보안강도 검증은 어떻게 하나요?"

⟨Q|Crypton⟩ 플랫폼 – 현재, 그리고 발전 방향

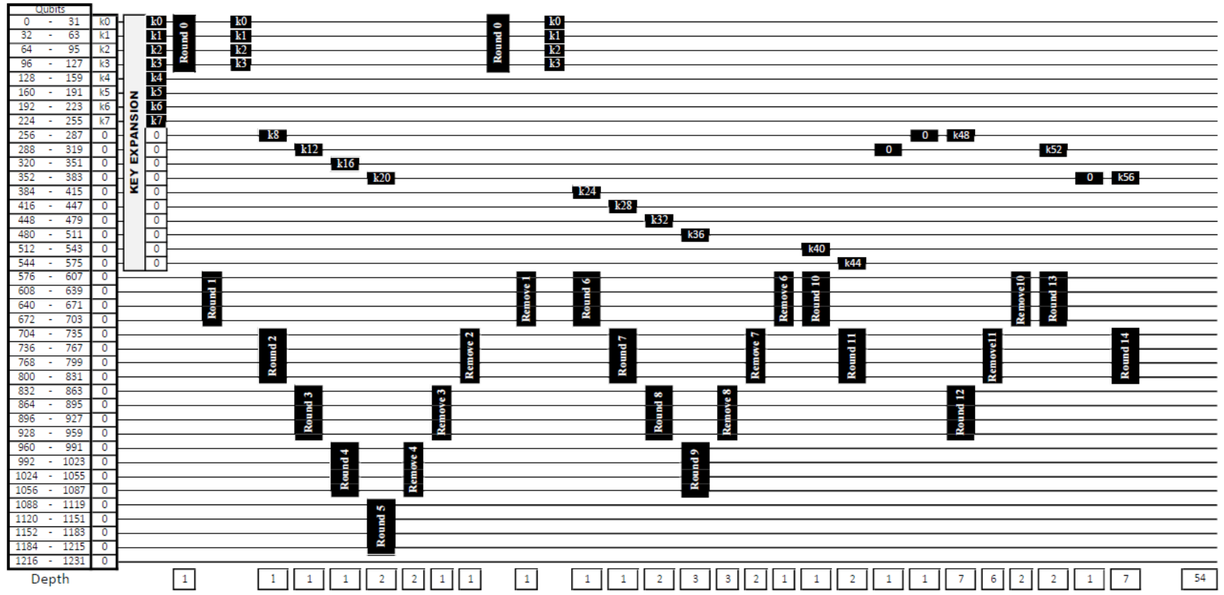
⟨Q|Crypton⟩ 진행 중인 개선 사항 1

사용자 편의성 개선

고수준 시각화 프로그래밍 기능 강화
- 저수준(단위게이트)이 아닌 고수준(양자 라이브러리/대용량 큐비트) 시각화 프로그래밍 및 파이썬 코드 자동 변환 기능

라이브러리 관리 기능 강화 및 고수준 시각화 프로그래밍 연동
- 사용자 라이브러리 등록/공유를 통한 효율적 프로그래밍
- 고수준 시각화 회로 UI 상에서 라이브러리 코드 삽입 기능

개인 프로젝트 및 작업 관리 기능 강화
- 양자 프로그래밍 통합개발환경 제공 및 비동기식 작업 관리 기능 (양자 프로그래밍 특성 반영)



대용량 큐비트 시각화 프로그래밍의 필요
(AES-256 양자회로 다이어그램)

⟨Q|Crypton⟩ 플랫폼 – 현재, 그리고 발전 방향

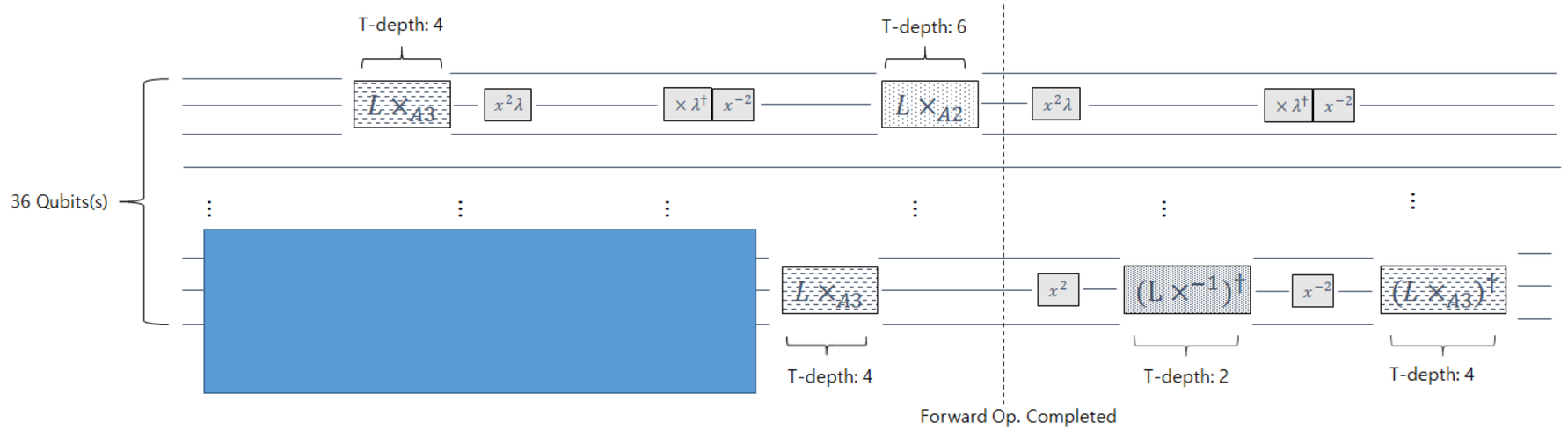
⟨Q|Crypton⟩ 진행 중인 개선 사항 2

암호 분석용 양자 라이브러리 공유 지원

암호 기본 단위연산(Arithmetic) 성능개선 연구 및 구현
- AES S-Box 상의 Multiplicative Inversion 양자회로 성능개선 연구 (한국정보보호학회 하계학술대회 우수논문)
- RSA, ECC 및 PQC의 기본 단위연산 성능 개선 등 병렬연구 진행 중

⟨Q|Crypton⟩을 통하여 개선된 암호 라이브러리 공유
- ⟨Q|Crypton⟩ 연구개발자들이 각자 구현한 암호 라이브러리 공유 ➔ 보다 효율적인 암호 양자분석 연구 가능

Small scale 코드 기반 Large scale 코드 검증이 가능한 방법론 제안
- 고수준 시각화 회로상에서 단위/부분(Small Scale) 양자코드 검증을 통한 전체(Large Scale) 양자코드 검증



예) GF(2^8)의 Multiplicative Inversion 양자 회로 효율성 개선 (ETRI 2020년 연구결과)

⟨Q|Crypton⟩ 플랫폼 – 현재, 그리고 발전 방향

⟨Q|Crypton⟩ 진행 중인 개선 사항 3

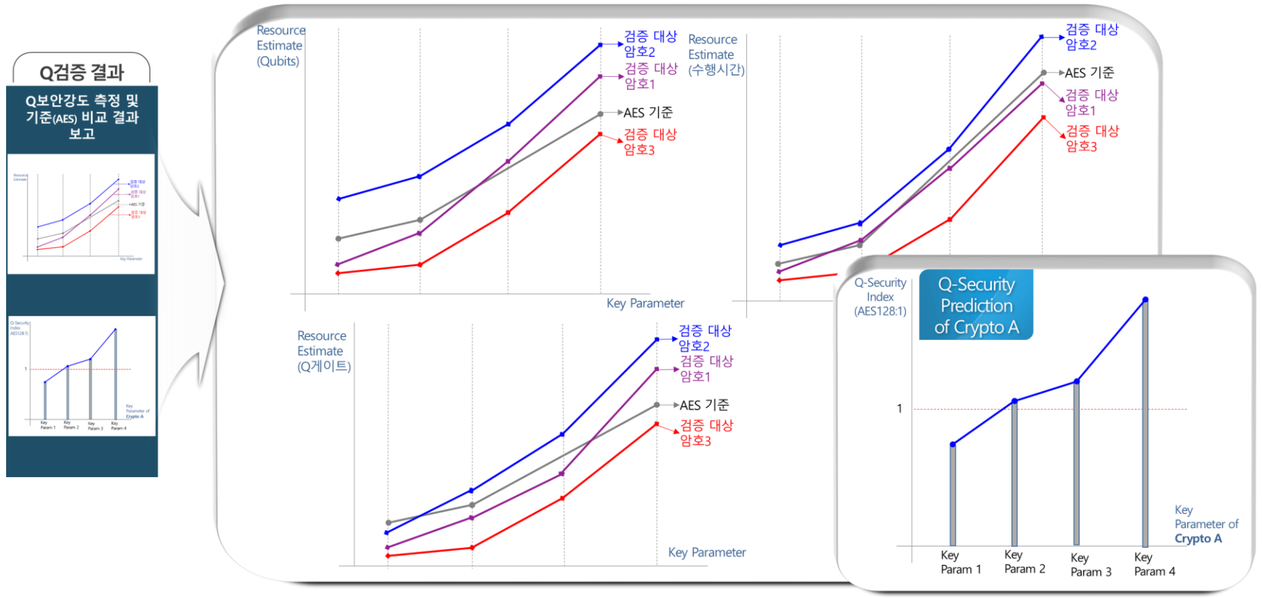
암호 양자 분석 및 비교 시각화 표현

검증 대상 암호별 양자자원(큐비트 수, 게이트 수, KQ, 수행 시간) 비교 자동화

- 기준 암호(ex. AES-256)와 검증 대상 암호 분석시 요구되는 양자자원량 비교를 통한 정량적 분석
- 특정 암호의 내부 파리미터(ex. Key size, 타원곡선 커브 등)에 따르는 양자자원량 비교
- 오류정정코드 및 양자 프로세서의 특성에 따르는 양자자원량 자동 비교

분석결과 추이그래프를 통한 결과 분석 기능 강화

- 분석 결과 저장, 선택, 그래프 생성 등을 손쉽게 할 수 있는 인터페이스 제공

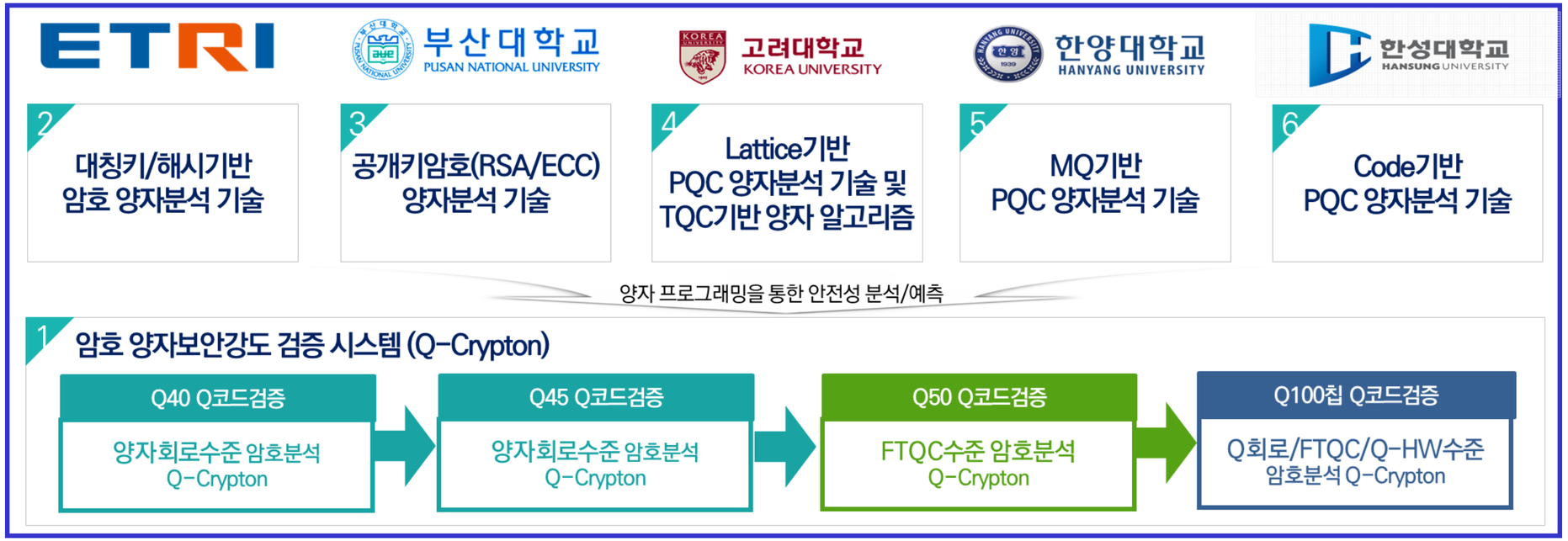


암호 양자분석 평가 방법론

⟨Q|Crypton⟩ 플랫폼 – 공동연구 및 국제협력

암호 양자안전성 연구 공동연구기관 협력

- **참여기관별 연구업무 분담을 통한 암호 전반에 대한 양자분석 기술 확보**
  - ETRI: ⟨Q|Crypton⟩ 플랫폼 지원 및 암호 양자분석 방법론 연구
  - ETRI 및 부산대: **기존 암호(대칭키/RSA/ECC)** 양자분석 및 보안강도 측정 기술 연구
  - 고려대, 한양대, 한성대: **양자내성암호(Lattice/MQ/Code 기반)** 양자분석 및 보안강도 측정 기술 연구
- **성공적인 연구를 위하여 암호 양자분석 기술 적극 교류 및 상호 협력체계 구축**



⟨Q|Crypton⟩ 플랫폼 – 공동연구 및 국제협력

공동연구개발지원 테스트베드 구축

- **ETRI 양자컴퓨팅 플랫폼 및 개발 환경 공동연구기관 및 협력기관에 공개 (교육 지원 및 환경 공유)**
  - 40큐비트 양자회로 동작 검증 플랫폼 및 **원격 개발환경 구축** (ETRI 융합기술연구생산센터 229-3호)
  - 산학연 공동연구 협업용(**양자 프로그래밍 교육, 실습, 시연** 등) 시설장비 설치 (ETRI 6동 L01호)
- **2020년 하반기부터 본격적으로 해당시설 활용을 통한 연구개발 경험, 노하우 공유 (현재는 공동연구기관 중심 공개)**

시설 운영 방안

양자 컴퓨팅 교육

- 양자 프로그래밍 및 회로 검증 교육
- 양자컴퓨팅 합성 및 양자하드웨어 수준 분석 방법 교육
- **양자회로 동작검증용 30큐비트 서버 구축 및 개발 PC/노트북 지원**

원격 개발 환경 지원

- **40큐비트 양자 프로그래밍 및 회로 동작 검증용 서버 구축**
- 다중 사용자/양자작업/양자플랫폼 지원 가능한 원격 개발환경 구축
- 고수준 시각화 프로그래밍을 위한 기능 개발

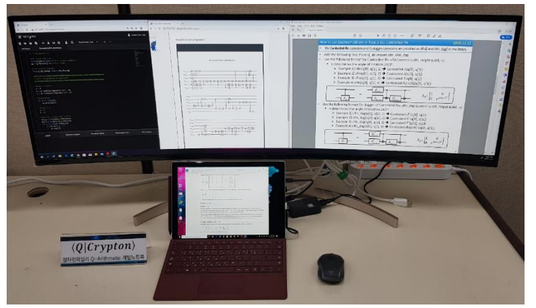
공동연구 세미나 및 회의

- **최소 2달에 1번 공동세미나/회의**
- 최신 양자컴퓨팅 및 양자내성암호 기술, 안전성 분석 연구 동향 교류
- 차세대 암호 Q프로그램 및 안전성 검증 기술 개발 경험 공유
- 결함허용을 위한 양자회로 최적화 구성 기법 (큐비트, T-depth) 공유

대외 홍보 및 협력 강화

- 양자컴퓨팅, 양자내성암호 등 관련 연구 수행 중인 국내외 전문가 초빙
- 양자컴퓨팅 플랫폼 및 양자내성암호 안전성 검증 기술 시연/홍보
- 세계 최고 수준의 연구개발시험 환경 구축 → **미래컴퓨팅 환경 안전성 검증 연구 글로벌 리더**

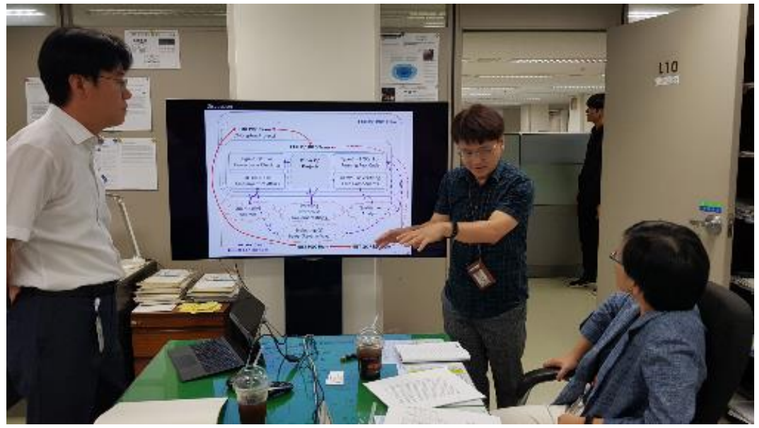
공동연구개발지원 테스트베드 활용 모습



⟨Q|Crypton⟩ 플랫폼 – 공동연구 및 국제협력

글로벌 협력 추진을 통한 기술 주도

- **핵심 수요처 확보 및 우수 양자분석 기술 확보를 위한 글로벌 연구협력 추진**
  - PQC 기술 연구 및 표준화를 선도하는 NIST(미국)/QUB(영국)와 연구협력체계 구축 중
  - Asiacrypt 2020 및 PQCrypto 2021 행사(한국 개최)에 <Q|Crypton> 소개·전시를 통한 수요처 확보
  - 국내외 최고 PQC 전문가의 <Q|Crypton> 기반 암호 양자분석 연구 유도 → 우수 양자내성암호 및 암호 양자분석기술 확보
- **PQC 연구/표준화 기관과 학술교류 및 실질적 협력을 통한 글로벌 기술 주도**

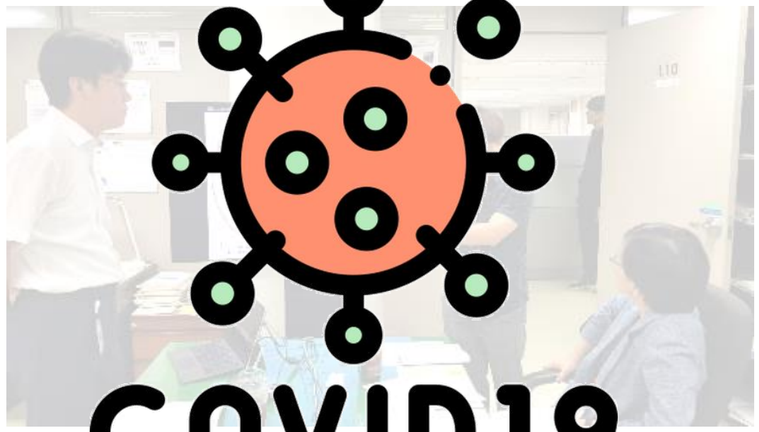


국내/국제협력을 통한 연구협력체계 구축 중 (NIST/QUB/KISTI/NSR 등, 2019~)

$\langle Q|Crypton\rangle$ 플랫폼 – 공동연구 및 국제협력

글로벌 협력 추진을 통한 기술 주도

아쉽게도...



COVID19

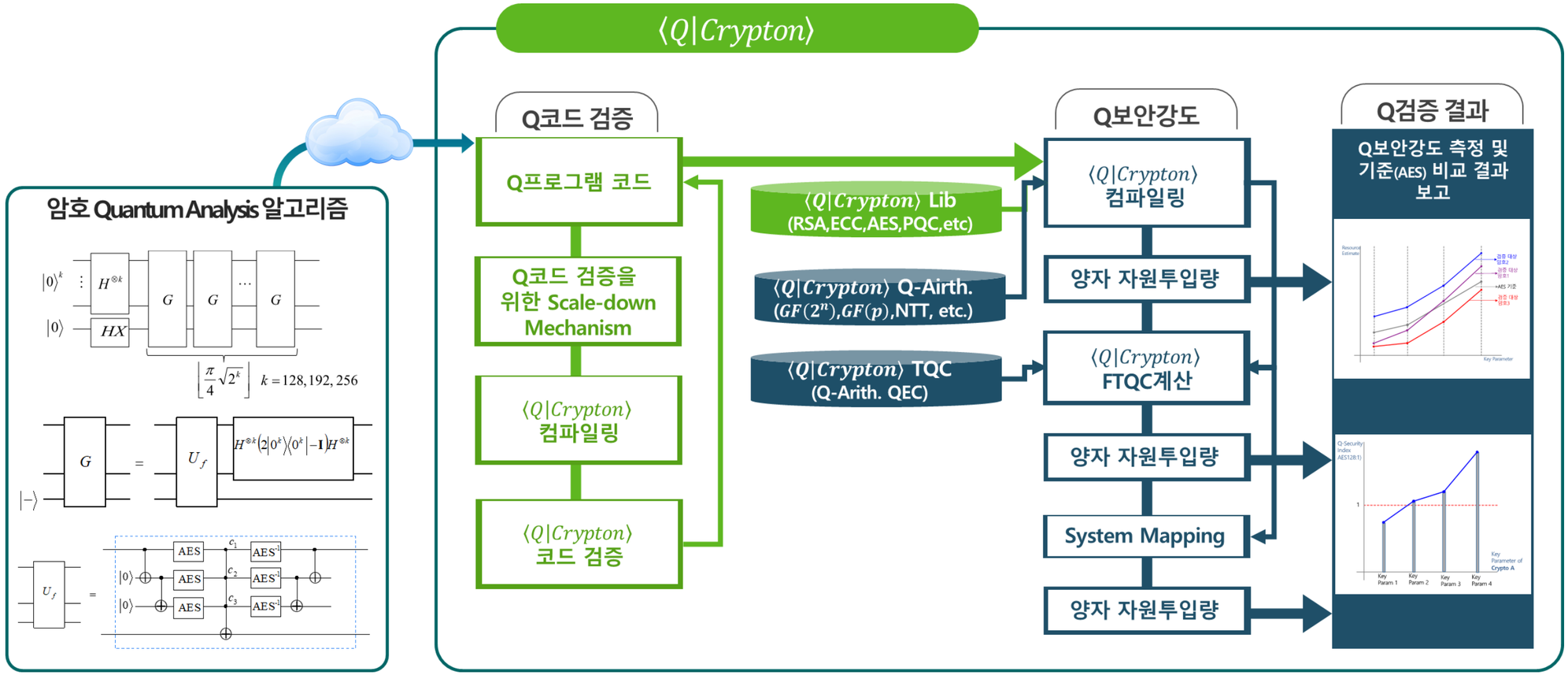
그러나

.
.
.

핵심 연구와 국내 협력을 강화할 좋은 기회!!

⟨Q|Crypton⟩ 플랫폼 – 요약

⟨Q|Crypton⟩ 플랫폼 요약



Thank you!