



CLOAQ: Ensuring the Cloud Quantum Computer Runs Your Program... But Learns Nothing

Speakers: Vivek Balachandran and Amal Raj

WHO ARE WE



Vivek Balachandran

Associate Professor
Singapore Institute of Technology

vivek.b@singaporetech.edu.sg

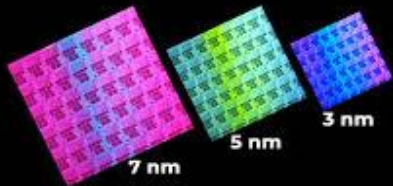
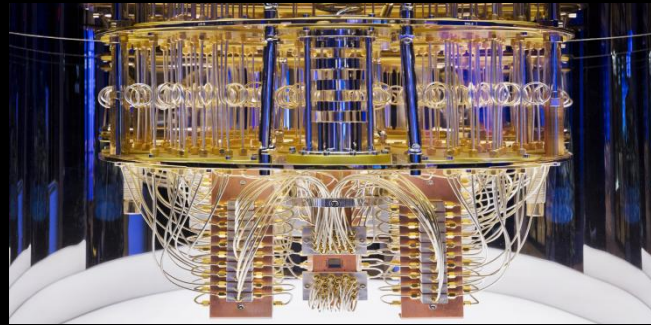
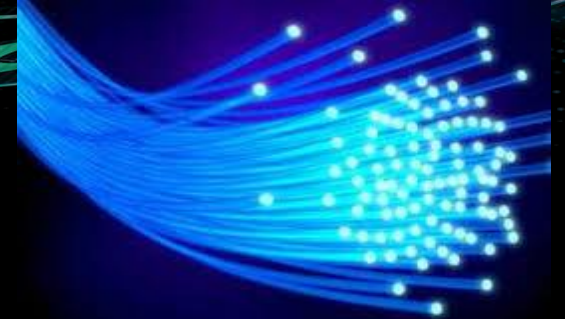


Amal Raj

Research Engineer
Singapore Institute of Technology

amal.raj@singaporetech.edu.sg

QUANTUM PHYSICS HAS ALREADY CHANGED OUR WORLD



FASTEST SUPERCOMPUTER EL CAPITAN



1.7420.000.000.000.000.000
Ops per Second (1.742 exaflops)

FASTEST SUPERCOMPUTER EL CAPITAN



1.7420.000.000.000.000.000
Ops per Second (1.742 exaflops)

~1.000.000 x
NVIDIA RTX 5080



FASTEST SUPERCOMPUTER EL CAPITAN

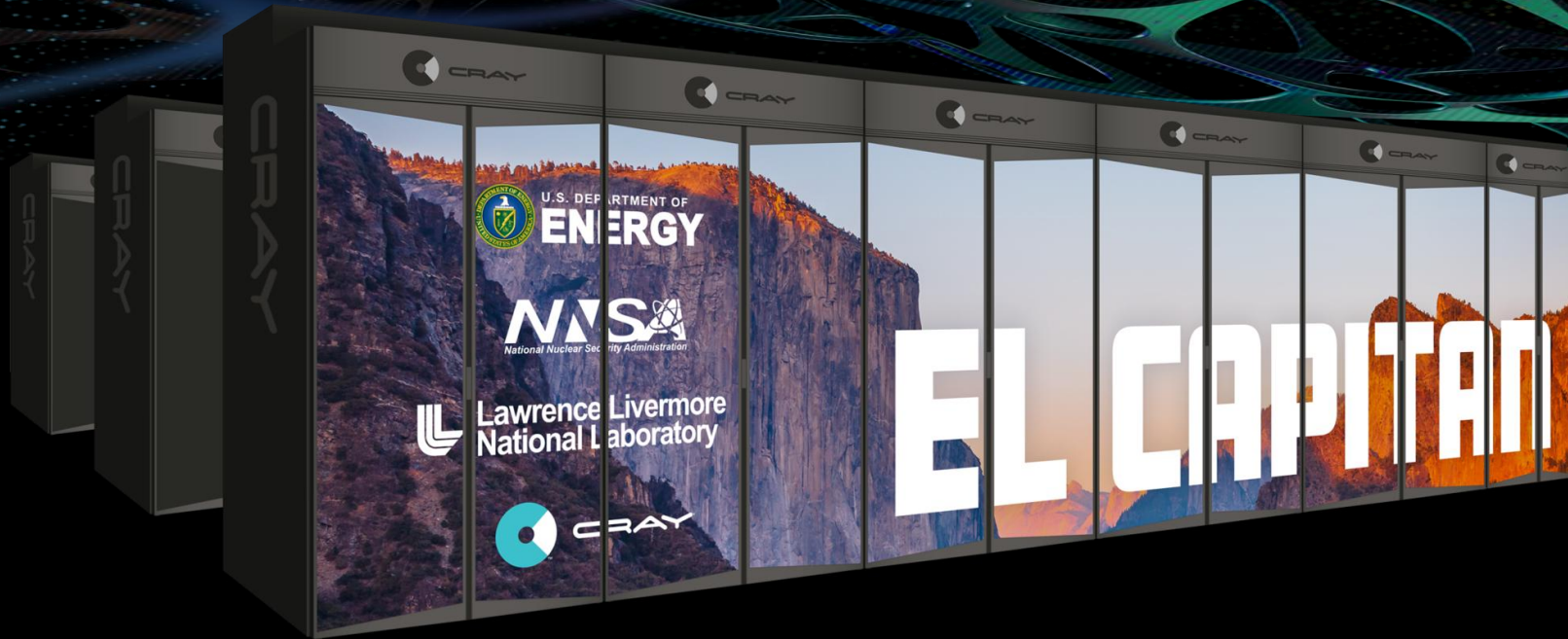


1.7420.000.000.000.000.000
Ops per Second (1.742 exaflops)



NVIDIA DGX GB300
Ops per Second (0.35 exaflops)

FASTEST SUPERCOMPUTER EL CAPITAN

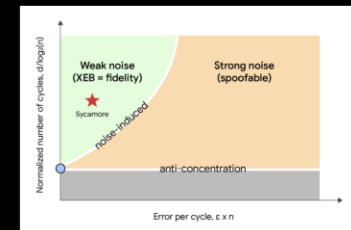


1.7420.000.000.000.000.000
Ops per Second (1.742 exaflops)

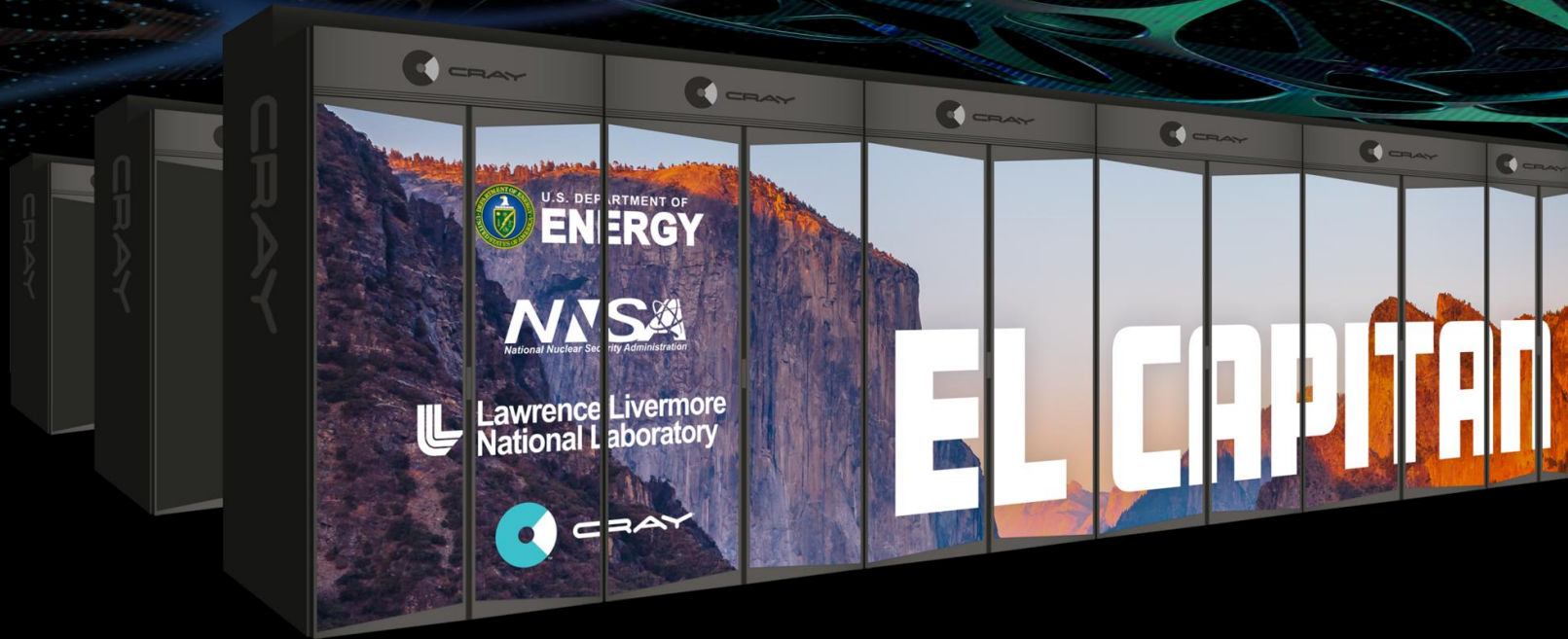
10.000.000.000.000.000.000.000.000
years for benchmark operation

Validating random circuit sampling as a benchmark for measuring quantum progress

October 10, 2024 · Kostyantyn Kechedzhi and Alexis Morvan, Research Scientists, Google Quantum AI



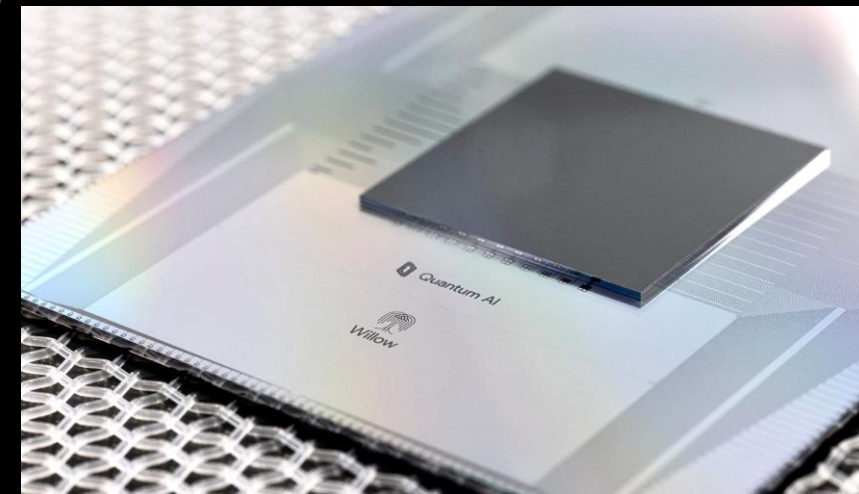
FASTEST SUPERCOMPUTER EL CAPITAN

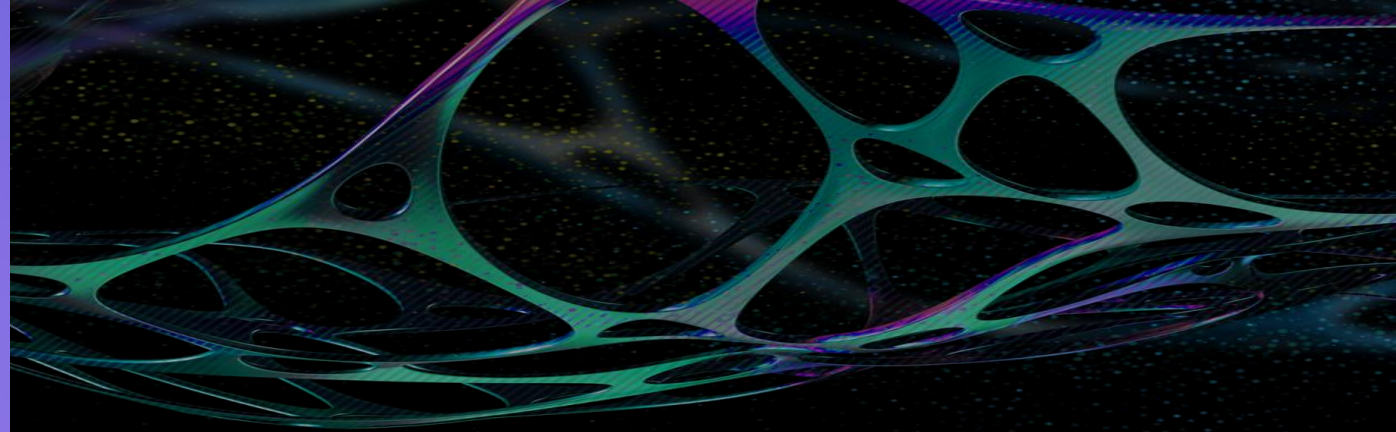


Google Quantum
Willow
5 minutes

1.7420.000.000.000.000.000
Ops per Second (1.742 exaflops)

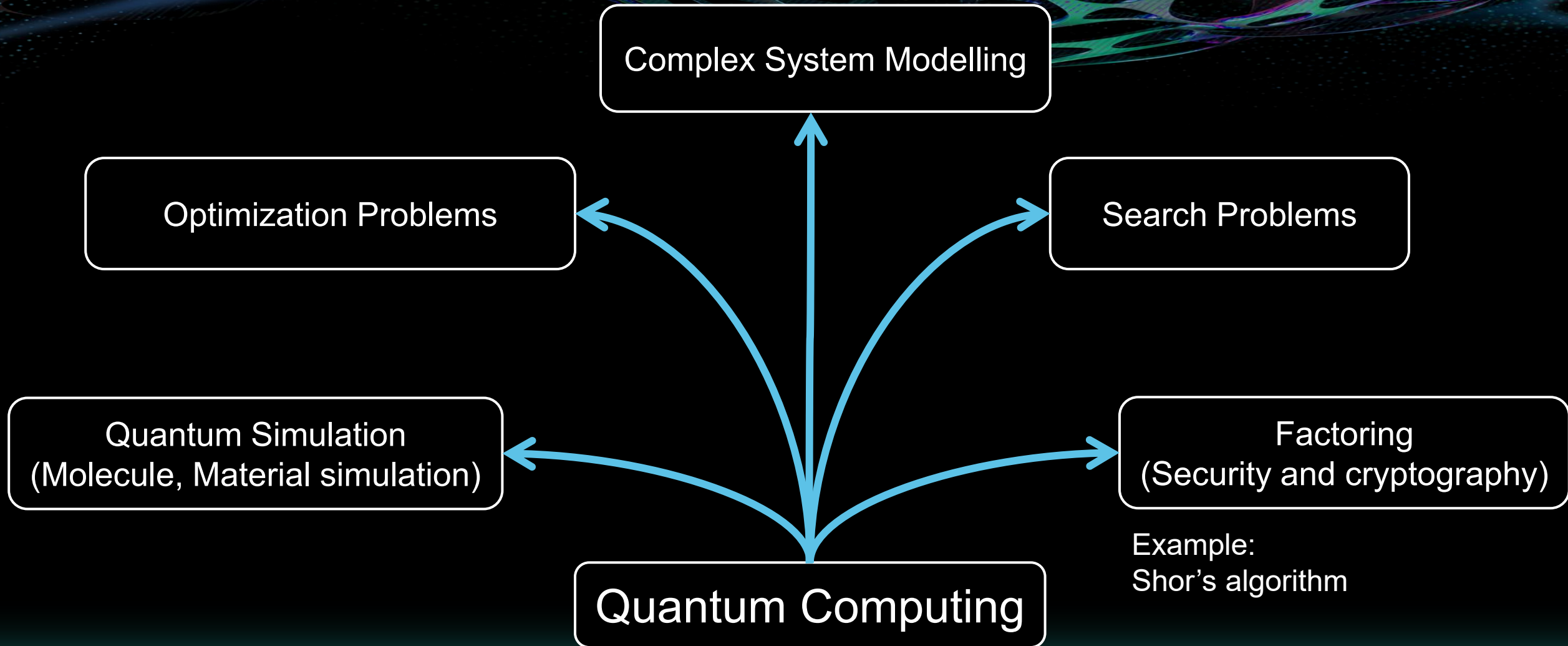
10.000.000.000.000.000.000.000.000
years for benchmark operation



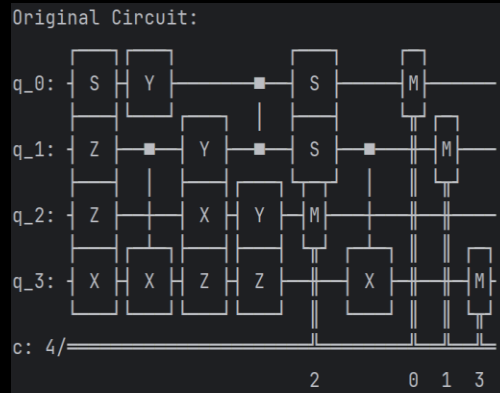


But, Will It Be Useful In Reality?

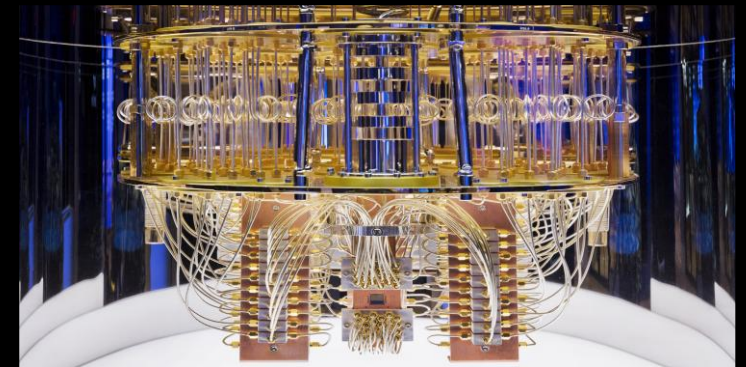
APPLICATIONS OF QUANTUM COMPUTING



QUANTUM PROGRAMMING IF YOU ARE NOT A MILLIONAIRE



**Quantum
Algorithm**

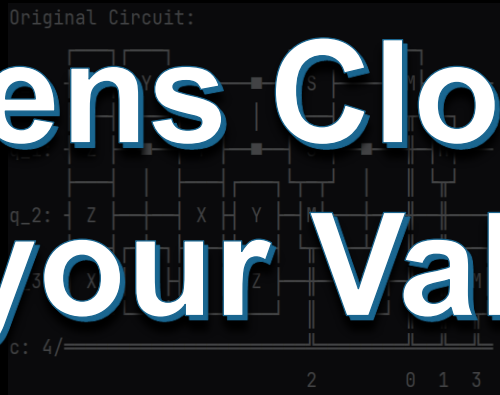


**Quantum Computing
Cloud Services**

QUANTUM PROGRAMMING IF YOU ARE NOT A MILLIONAIRE



Opens Cloud Based Attacks on your Valuable Quantum IP



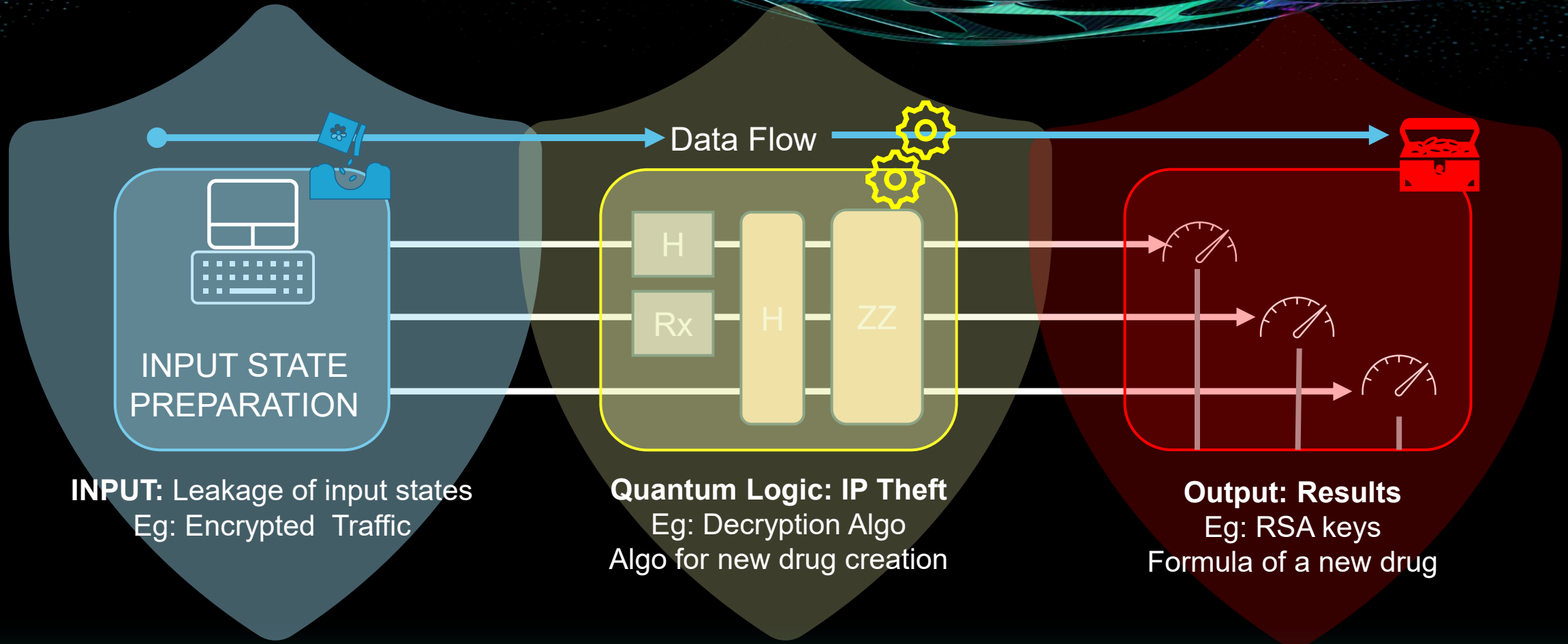
Quantum
Algorithm

Quantum Computing
Cloud Services

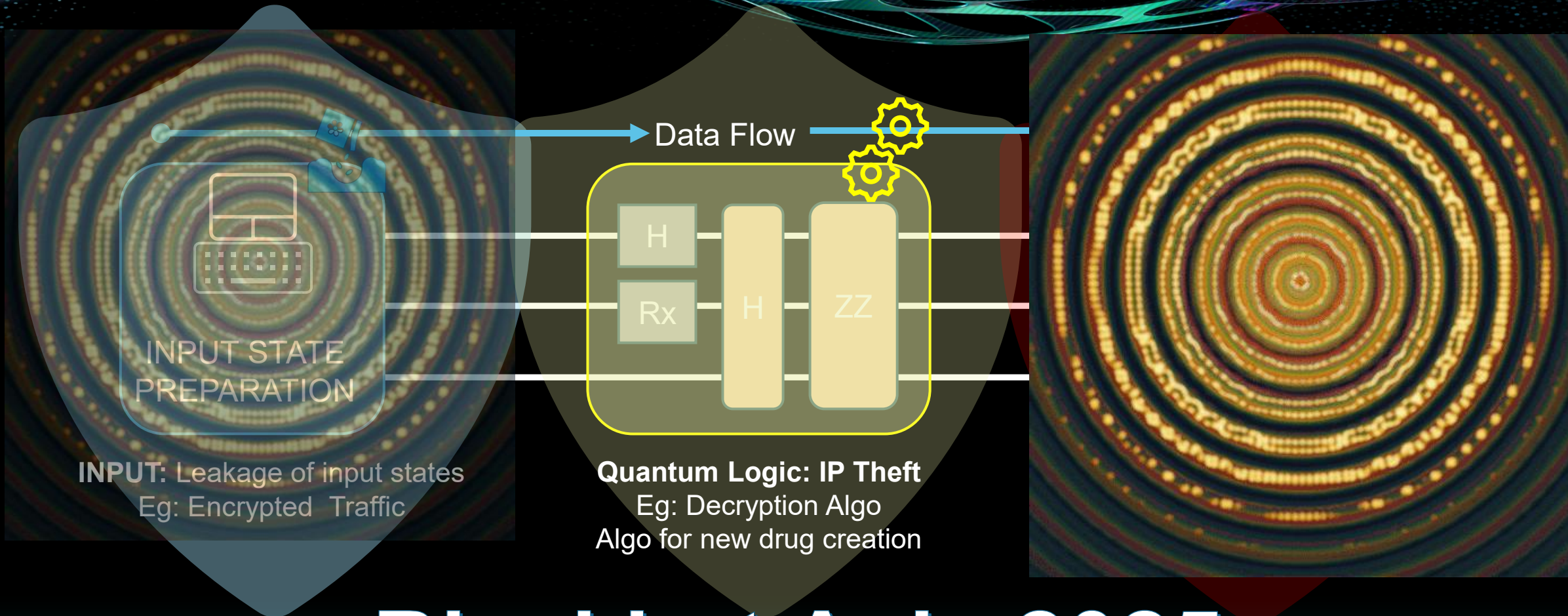


ATTACK SURFACES

ATTACK SURFACES

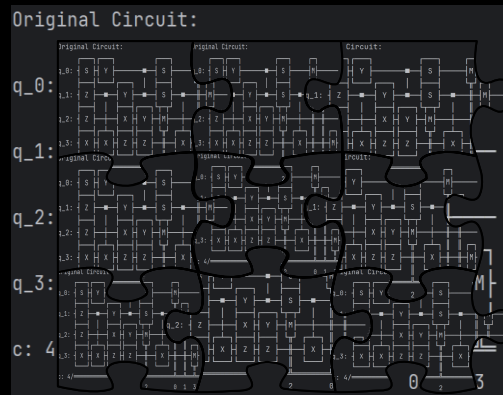


ATTACK SURFACES

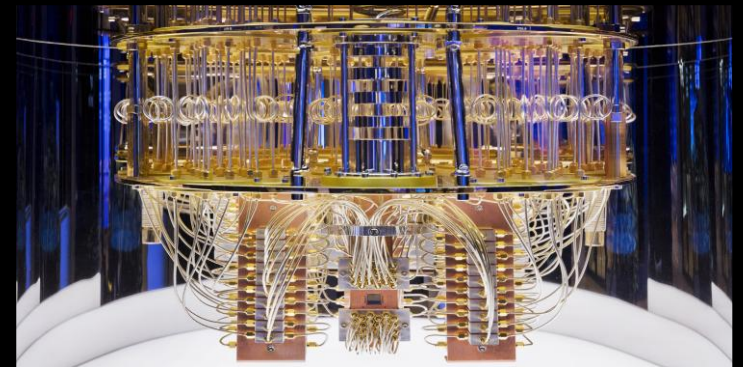


Blackhat Asia 2025

OUR SOLUTION



**OBFUSCATE BEFORE
COMPILATION**



**Quantum
Algorithm**

**Quantum Computing
Cloud Services**



OBFUSQATE

The First Automated Quantum Obfuscation Framework
"The Quantum Code Shield"

BlackHat Asia 2025 → 2026 Evolution

⚠ THE PROBLEM

Quantum is Going Cloud-Native

- Code runs on remote hardware
- Cloud Quantum APIs
- High-value IP exposed

NEW ATTACK SURFACE

- Quantum Circuit Reverse Engineering
- Quantum Algorithm Extraction

⚠ **Your Quantum IP is Exposed by Default**

OUR INNOVATION

First-of-its-kind Framework
Fully Automated Quantum Obfuscation Pipeline



Obfuscate Before Execution



OBFUSCATION TECHNIQUES

Circuit Obfuscation

- Inverse Gates
- Cloaked Gates
- Composite Gates
- ✓ Preserves Functionality
- ✗ Breaks Reverse Engineering

Code Obfuscation

- Superposition Shroud
- Entanglement-Based Branching
- Probabilistic Execution Paths

Quantum-Native Protection

QUANTUM-NATIVE SECURITY

Security through quantum physics

- Superposition
- Entanglement
- Measurement Uncertainty

LLM RESISTANCE DEMONSTRATED

LLM-Proof Quantum Code

- AI ✓ Clear Understanding (Un-obfuscated)
- AI' ✗✗ Fails / Degrades (Obfuscated)

First step towards LLM-resistant quantum code

DEFENSIVE

- Protect Quantum IP
- Secure Cloud Execution
- Anti-Reverse Engineering

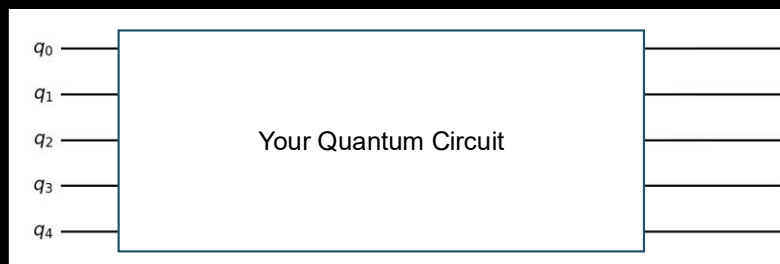
OFFENSIVE (Research)

- Obfuscated Quantum Malware
- Hidden Quantum Payloads

🔥 **Running quantum code in the cloud? Your IP is already exposed. → OBFUSQATE**

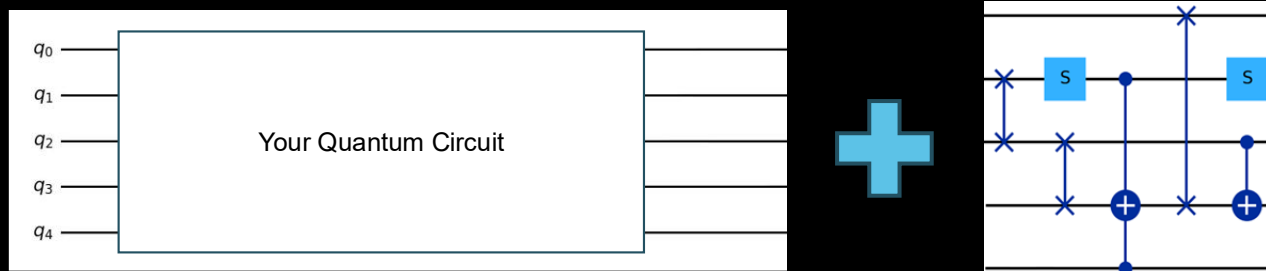
SO, CAN WE?

- Take a Quantum Circuit



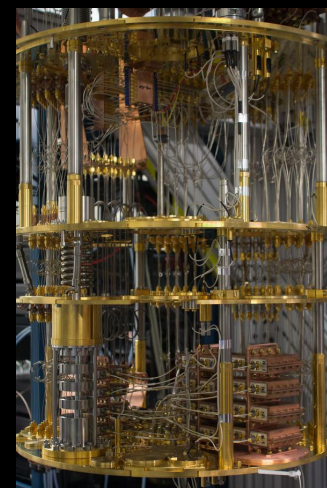
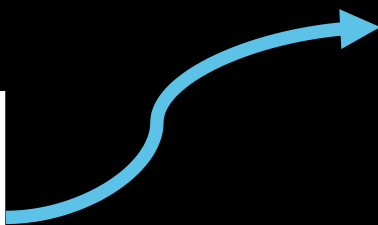
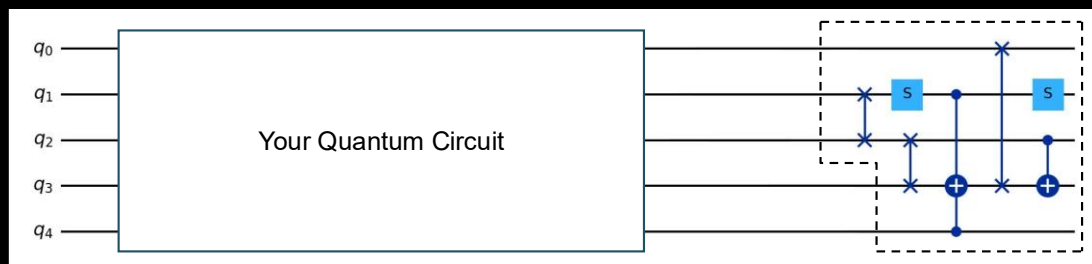
SO, CAN WE?

- Add random quantum gates to corrupt it



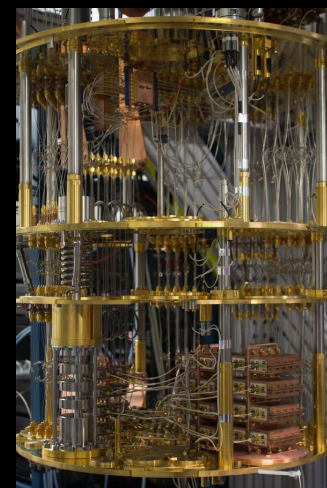
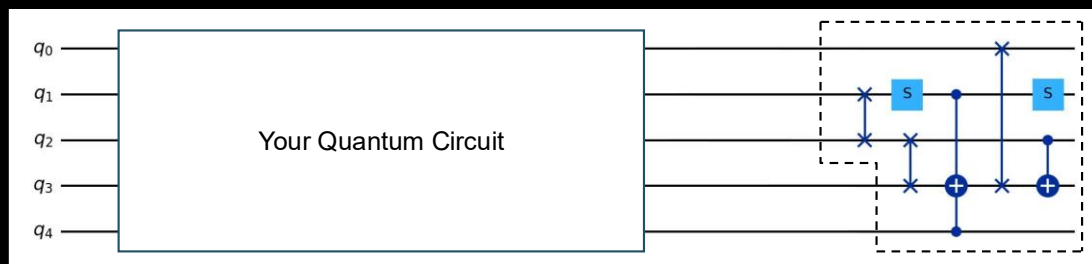
SO, CAN WE?

- Deploy the corrupted circuit to a third party quantum computer



SO, CAN WE?

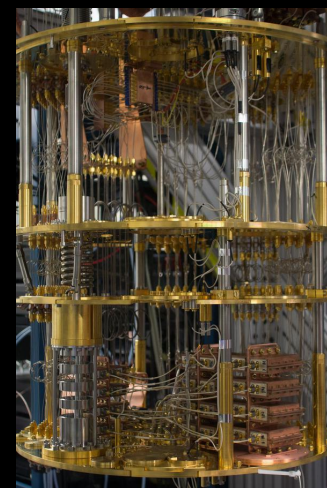
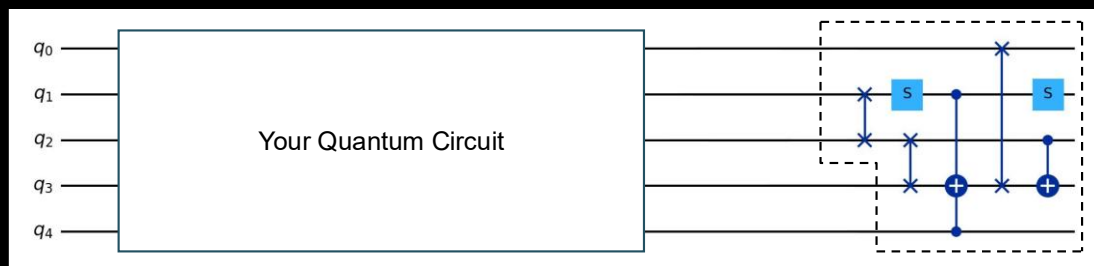
- Get a corrupted quantum solution



2@3Ab567G
fmsecefasf94
rrglmtP

SO, CAN WE?

- And magically extract meaningful answer at user side



Answer to Life,
Universe and
everything is 42!

2@3Ab567G
fmsecefasf94
rrglmtP

CLOAQ - 2026

- Protecting the Output Results
- Obfuscation scheme on Outputs
- Obscure results from cloud
- Extract correct results locally

INPUT: Leakage of input states
Eg: Encrypted Traffic

Quantum Logic: IP Theft
Eg: Decryption Algo
Algo for new drug creation

Output: Results
Eg: RSA keys
Formula of a new drug



QUANTUM CLOUD IS EXPENSIVE TOO

QUANTUM COMPUTE IS EXPENSIVE!

Typical quantum cloud pricing ~ \$100 / min

\$1000 → 10 minutes of compute

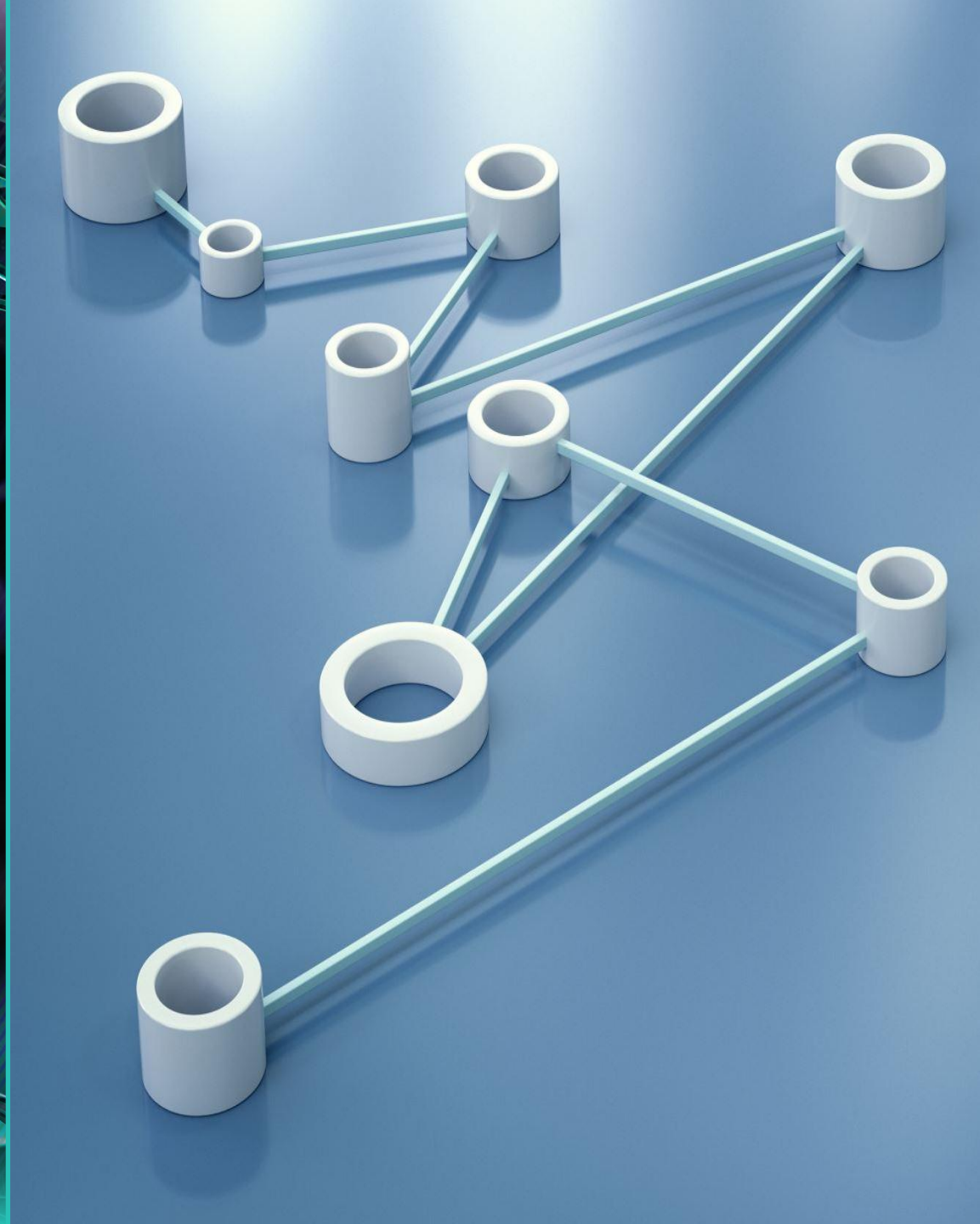
\$6000 → 1 hour of compute

Circuit Depth → longer runtime

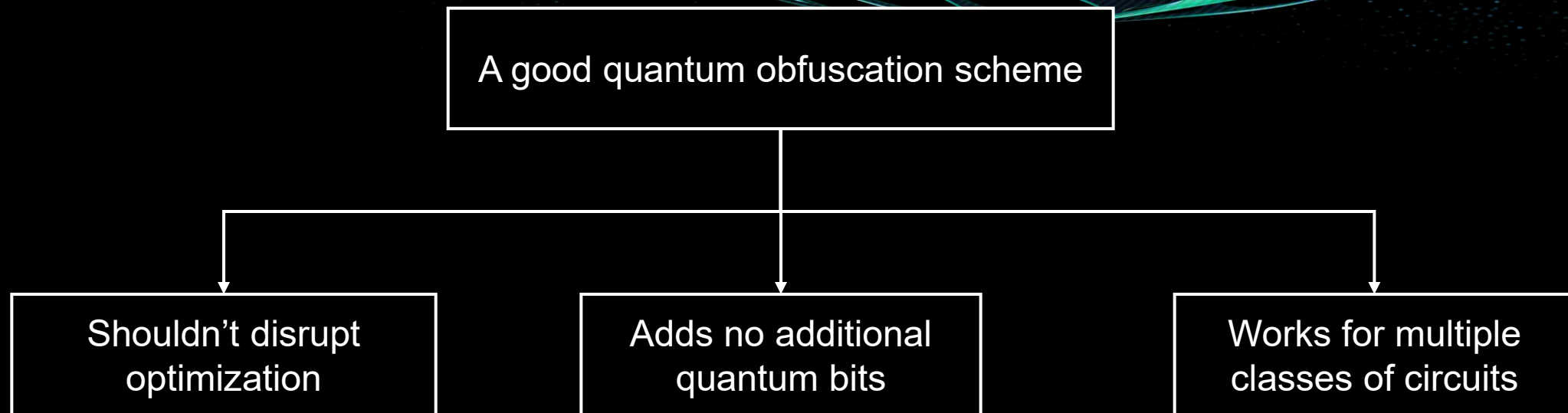
Quantum Bits → higher resource cost

Inefficient circuits cost money!

EXPECTATIONS FROM A GOOD OBFUSCATION SCHEME



EXPECTATIONS



THE OBFUSCATION LANDSCAPE

- Dummy Gate Insertion
 - Insert gates that are eliminated after cloud-side optimization.
- Logic locking
 - Add lock qubits to encrypt circuit logic, unlocked using a secret key.
- Phase Modification
 - Encode phase changes using a key and restore after execution.

SO, WHERE DO THEY STAND?

A good quantum obfuscation scheme

Shouldn't disrupt optimization

Adds no additional quantum bits

Works for multiple classes of circuits

Dummy Gates	✗	✓	✓
Logic Locking	✓	✗	✓
Phase Modification	✗	✓	✗
CLOAQ (Our Solution)	✓	✓	✓

THE OBFUSCATION LANDSCAPE

- Dummy Gate Insertion
- Logic locking
- Phase Modification

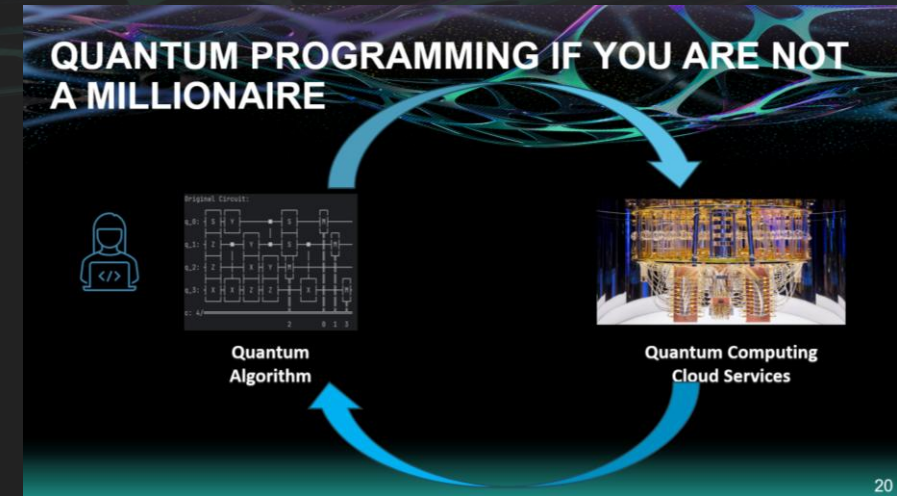
- **BUT, THERE IS ANOTHER DISADVANTAGE!**

THE OBFUSCATION LANDSCAPE

DEOBFUSCATION

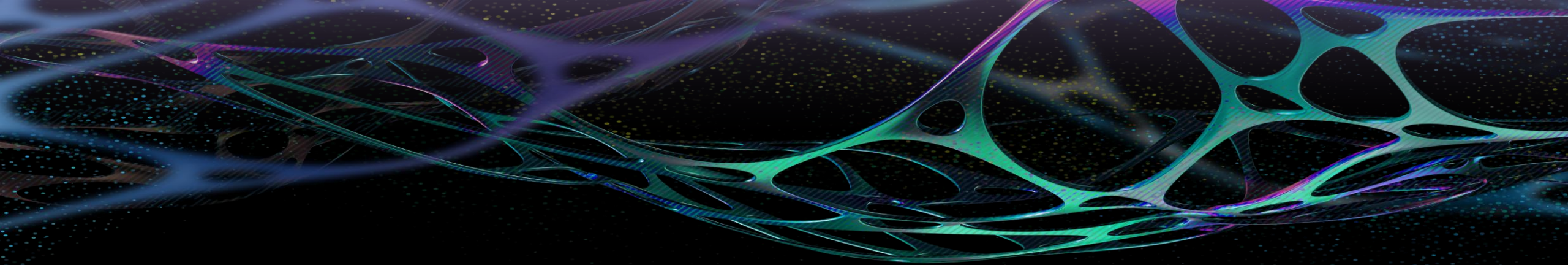
- Dummy Gate Insertion
- Logic locking
- Phase Modification

Assumes access to a local quantum computer!



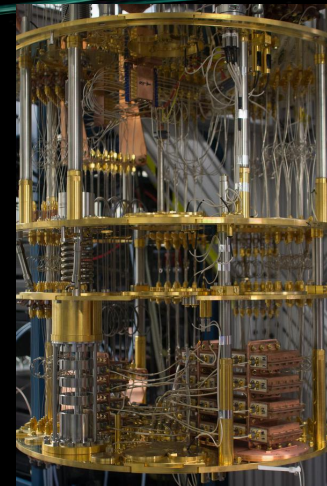
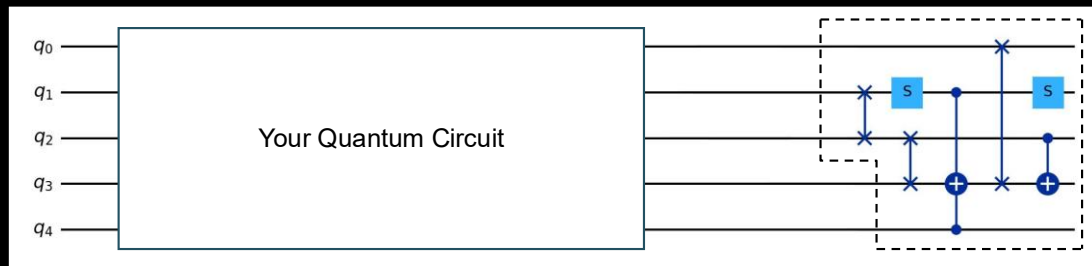
YOU WOULD HAVE TO BE A MILLIONAIRE!

• BUT, THERE IS ANOTHER DISADVANTAGE!

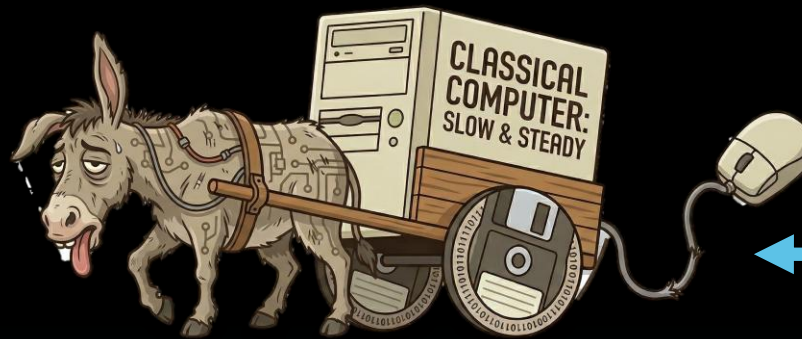


**You don't need a Local
Quantum Computer to apply
CLOAQ!**

WITH CLOAQ – EXTRACTION IS CLASSICAL



Answer to Life,
Universe and
everything is 42!



2@3Ab567G
fmsecefasf94
rrglmtP



QUANTUM COMPUTING ESSENTIALS

QUBITS AND SUPERPOSITION

WHAT IS A QUBIT?

A qubit is the quantum equivalent of a classical bit



While a classical bit is either 0 or 1, a qubit can exist in both states simultaneously

WHY IT MATTERS

With qubits, quantum computers can explore many possibilities at once.

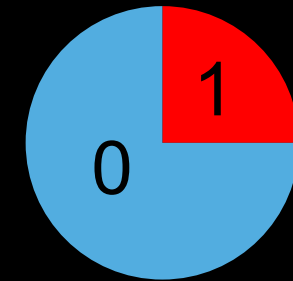
MORE PARALLELISM.
MORE POWER.

This is the foundation of quantum advantage

BUILDING BLOCKS OF QUANTUM OPERATIONS

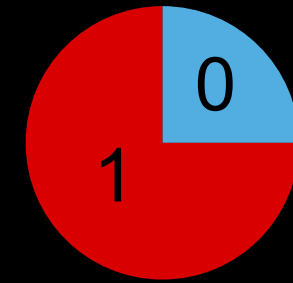


BUILDING BLOCKS OF QUANTUM OPERATIONS



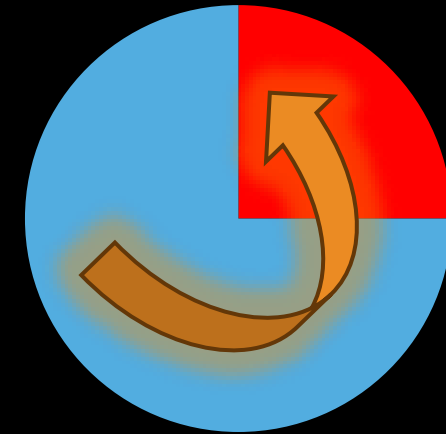
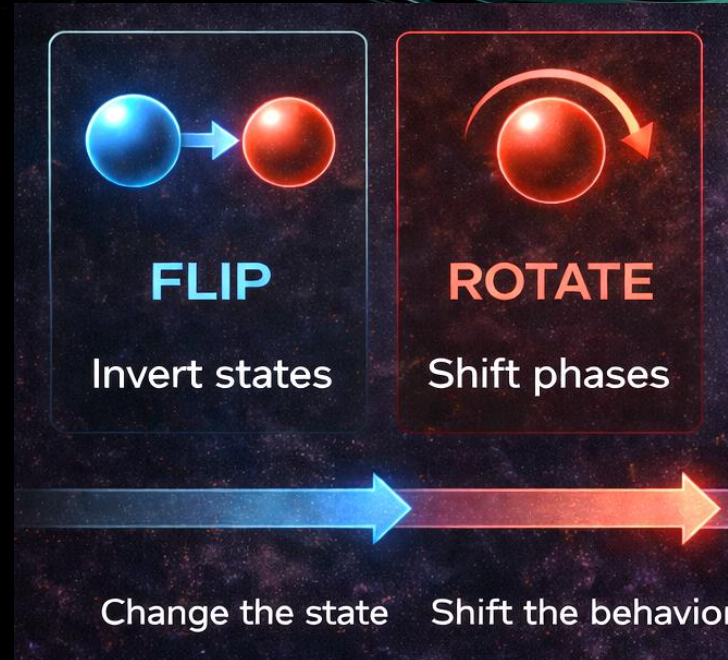
75% 0, 25% 1

BUILDING BLOCKS OF QUANTUM OPERATIONS

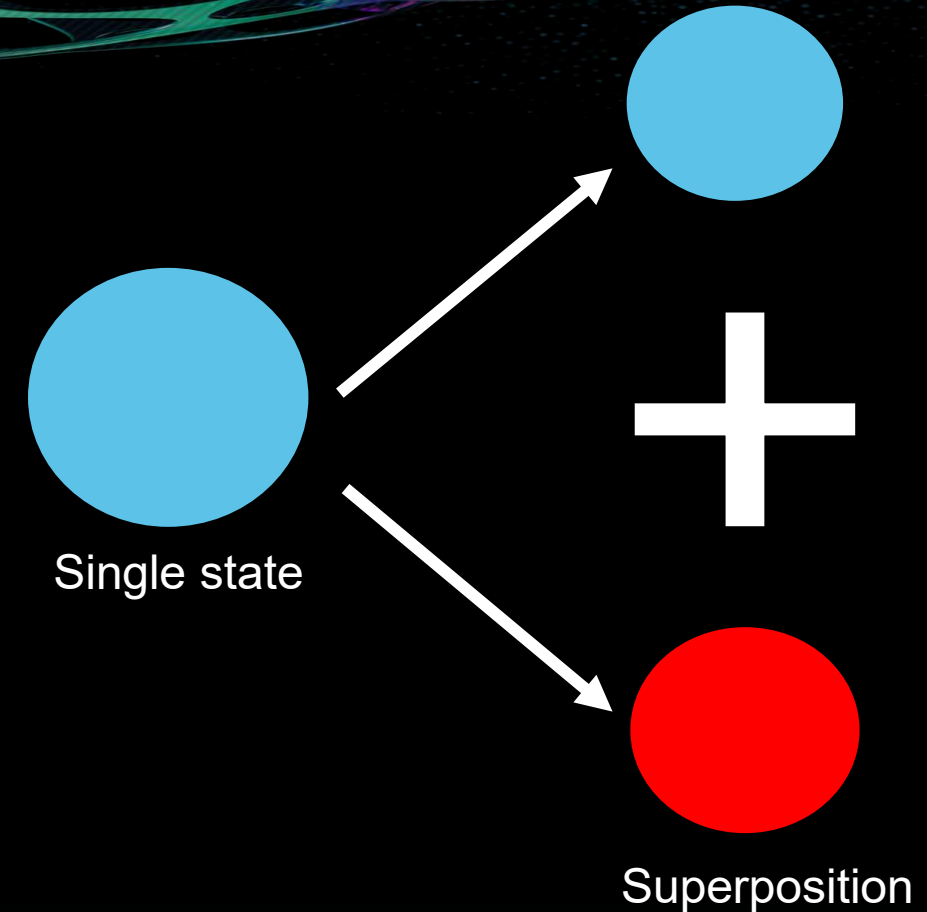
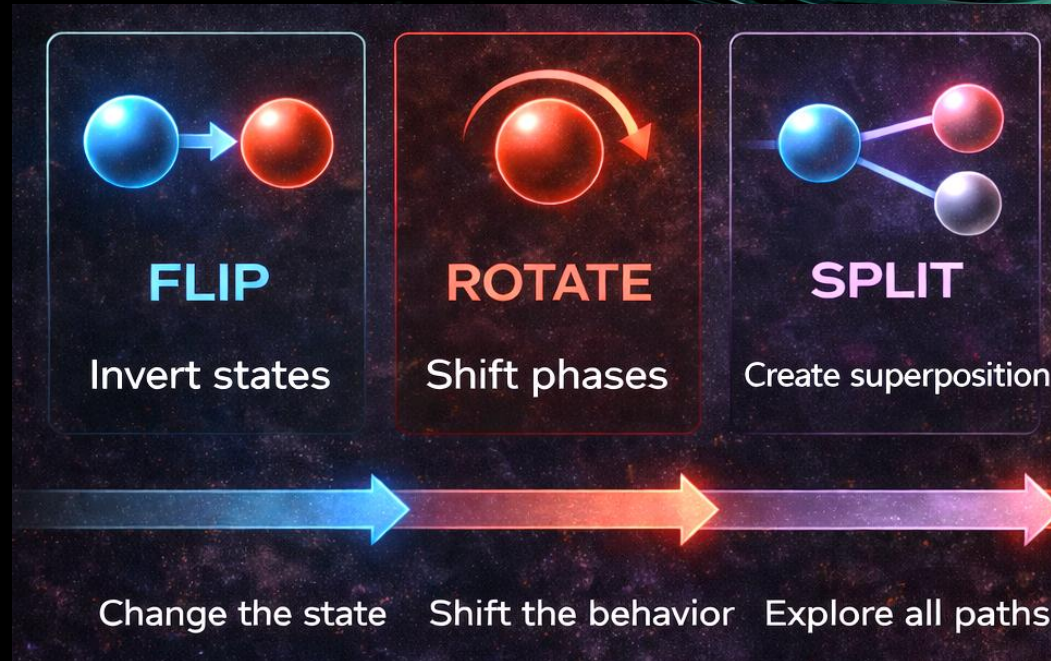


25% 0, 75% 1

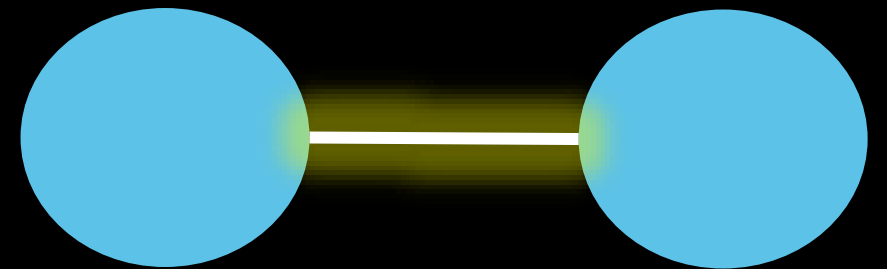
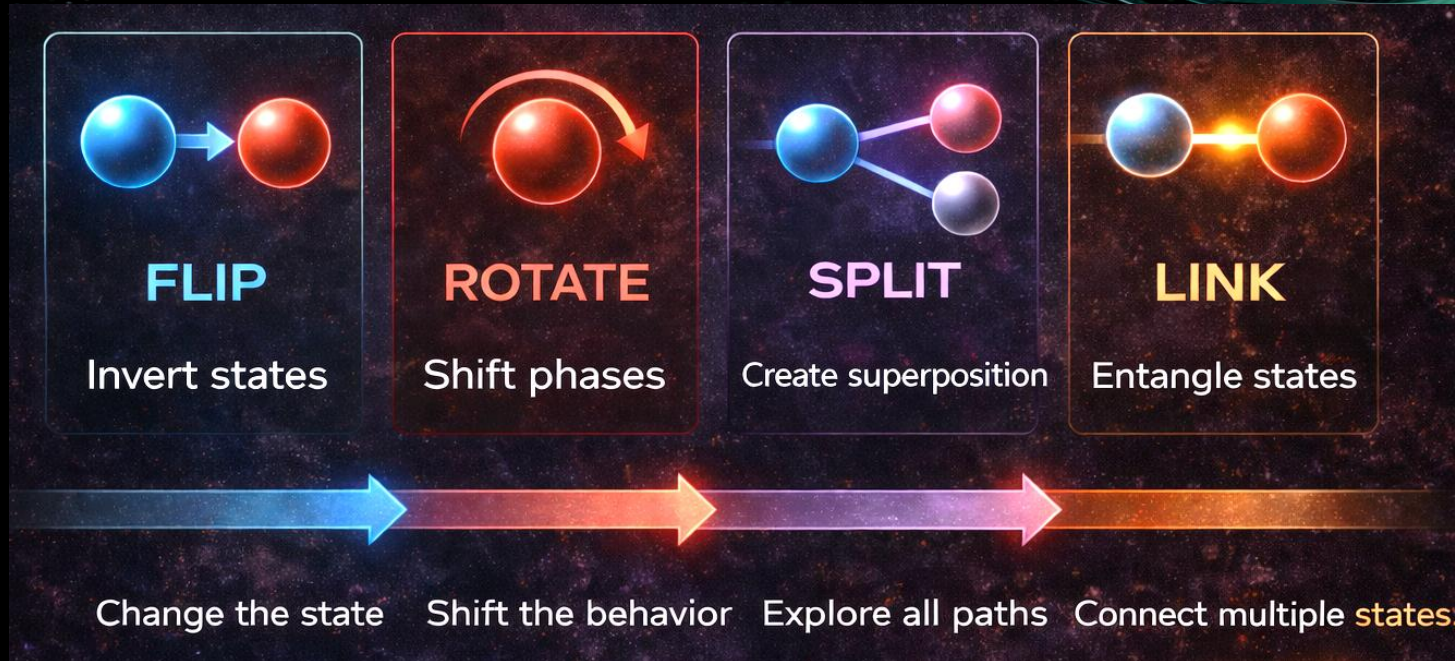
BUILDING BLOCKS OF QUANTUM OPERATIONS



BUILDING BLOCKS OF QUANTUM OPERATIONS

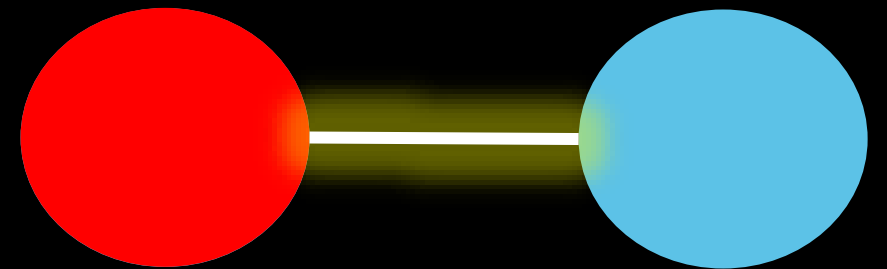
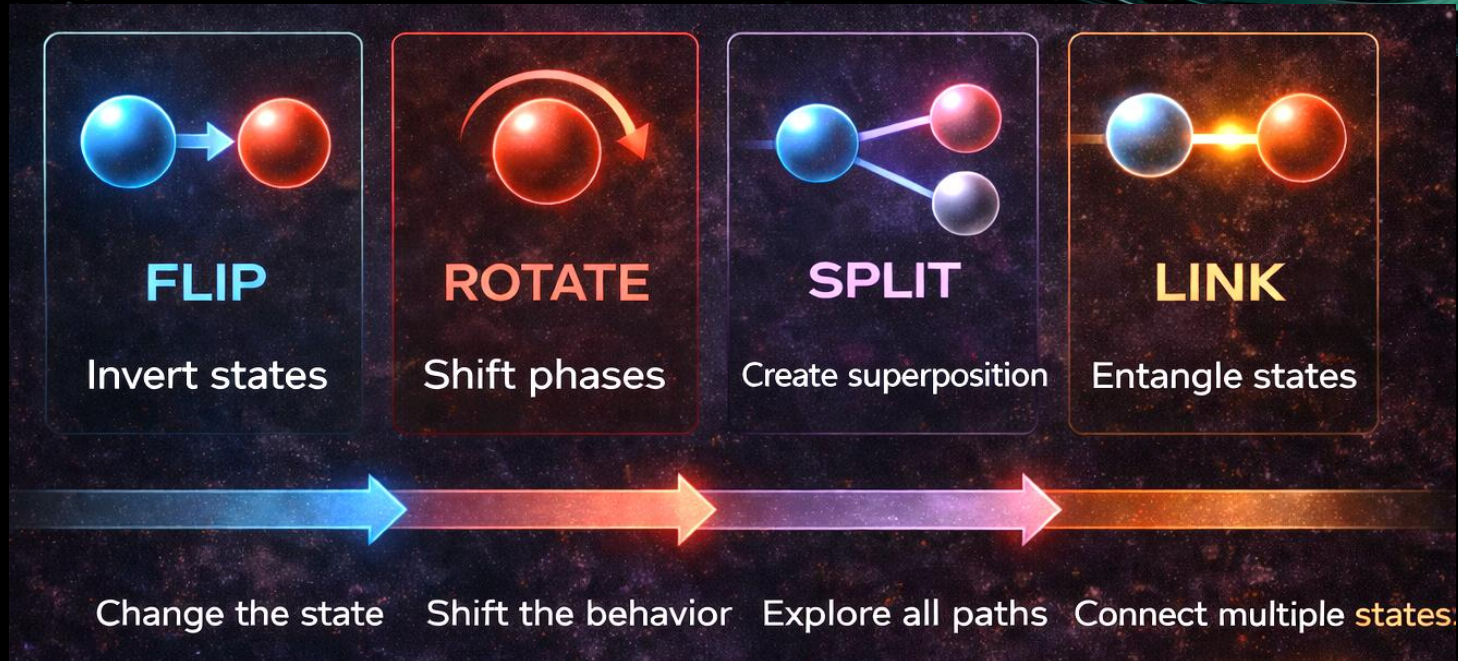


BUILDING BLOCKS OF QUANTUM OPERATIONS



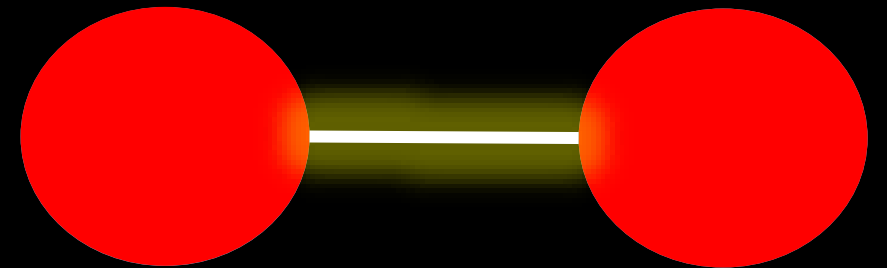
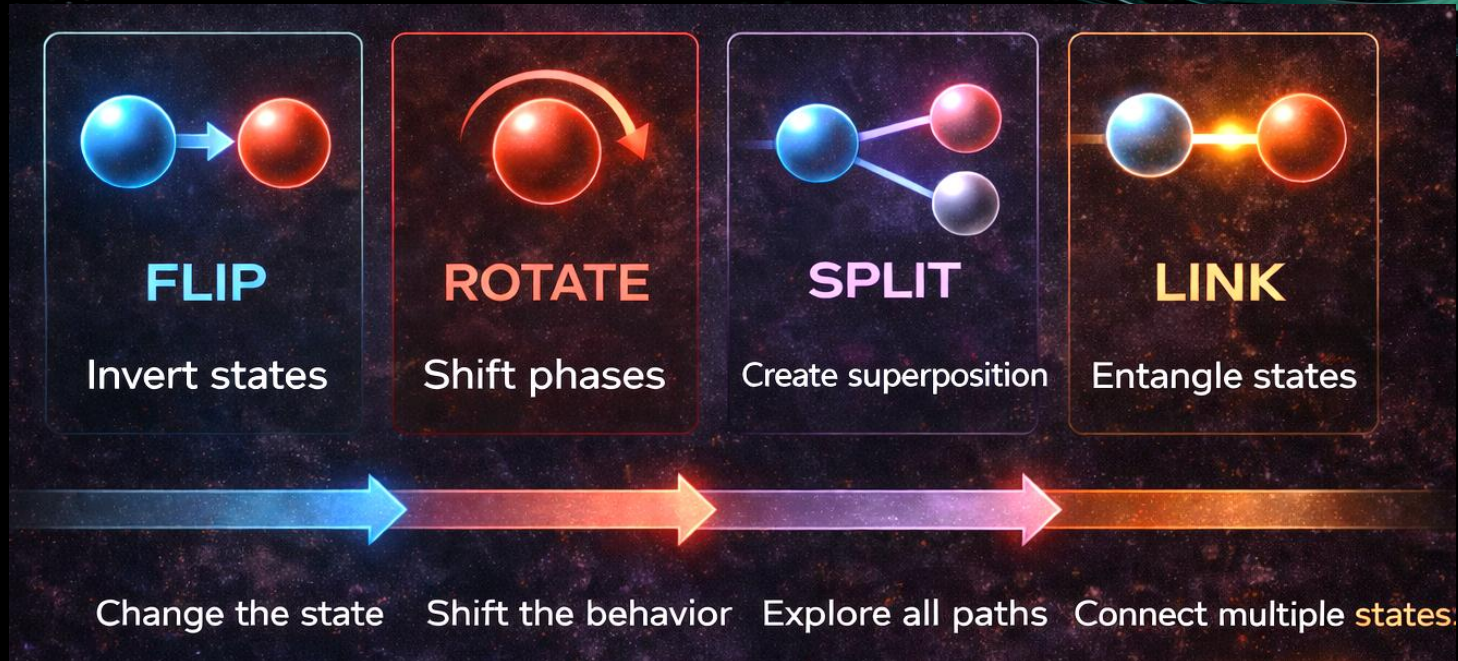
Change one... And the other follows

BUILDING BLOCKS OF QUANTUM OPERATIONS



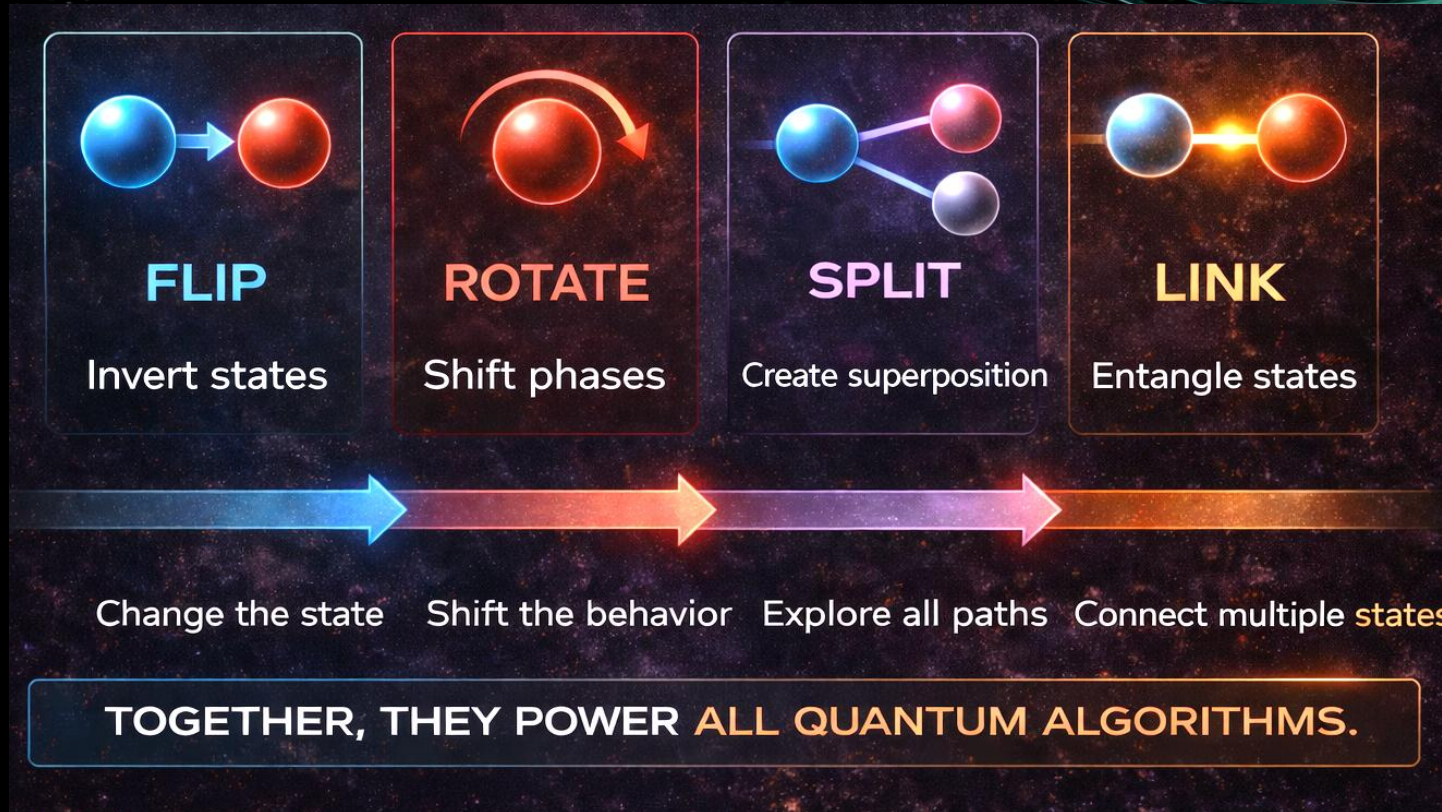
Change one... And the other follows

BUILDING BLOCKS OF QUANTUM OPERATIONS

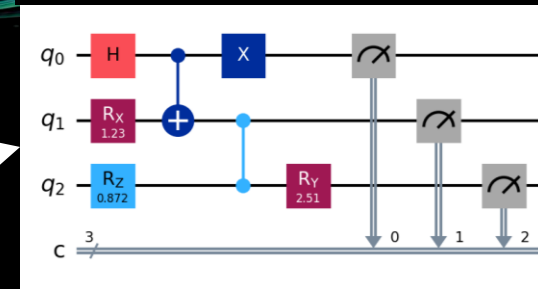


Leads to entanglement – Fate of one qubit depends on another!

BUILDING BLOCKS OF QUANTUM OPERATIONS



Quantum Circuit



QASM (Quantum Assembly)

```
OPENQASM 2.0;
include "qelib1.inc";

qreg q[3];
creg c[3];

// Random single-qubit rotations
h q[0];
rx(1.234) q[1];
rz(0.872) q[2];

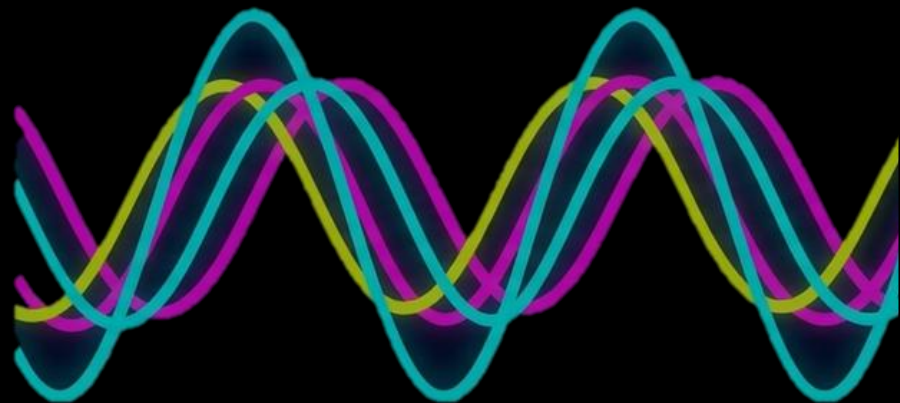
// Random entangling gates
cx q[0], q[1];
cz q[1], q[2];

// More random single-qubit gates
x q[0];
ry(2.513) q[2];

// Measurement
measure q -> c;
```

MEASUREMENT

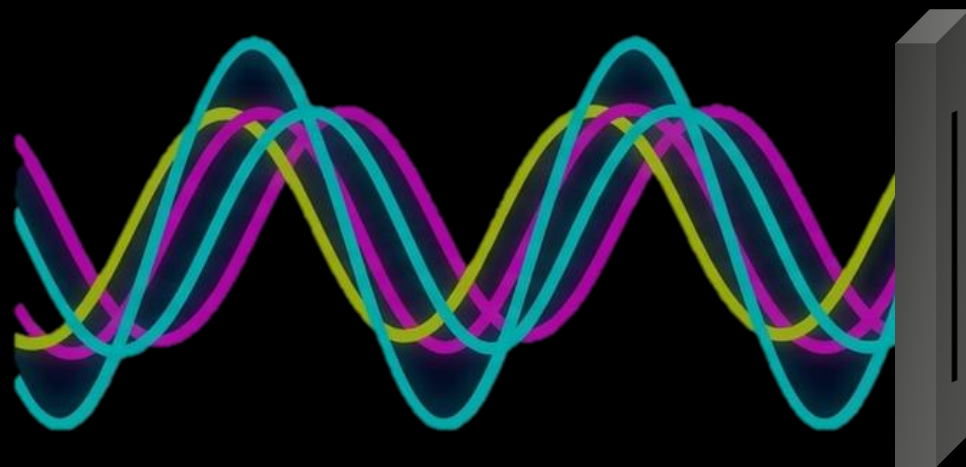
Pre-Measurement: Qubits exist as superposition of different states



MEASUREMENT

Pre-Measurement: Qubits exist as superposition of different states

Observation Lens

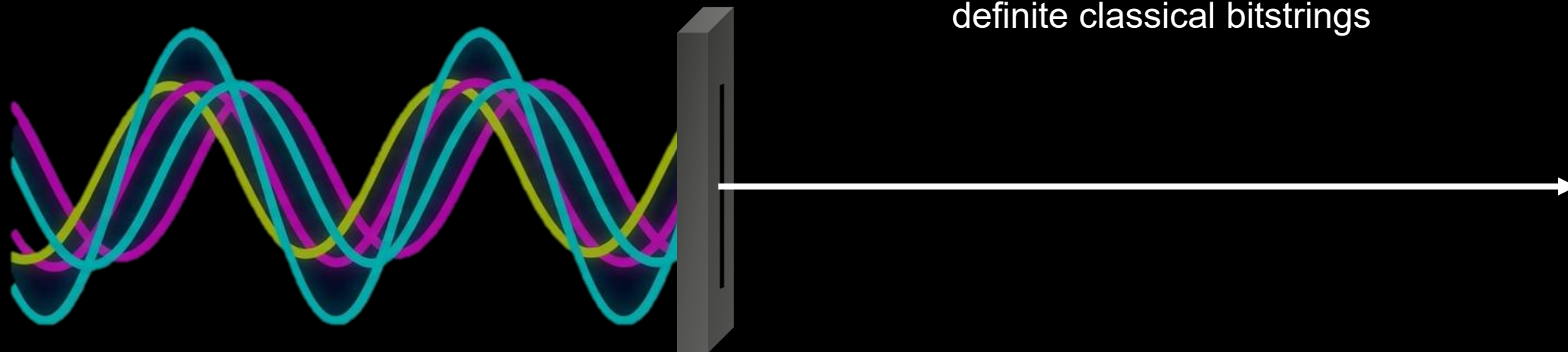


MEASUREMENT

Pre-Measurement: Qubits exist as superposition of different states

Observation Lens

The Collapse: Measurement forces superpositions to collapse into definite classical bitstrings



LET US LOOK AT SOME QUANTUM GATES

Quantum X Gate



Truth Table

q_0	$X(q_0)$
0	1
1	0

LET US LOOK AT SOME QUANTUM GATES

Quantum X Gate



Truth Table

q_0	$X(q_0)$
0	1
1	0

Does this look familiar?

Yes! This is the classical NOT gate!

LET US LOOK AT SOME QUANTUM GATES

Quantum X Gate

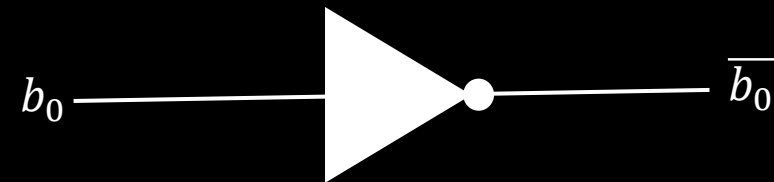


Truth Table

q_0	$X(q_0)$
0	1
1	0

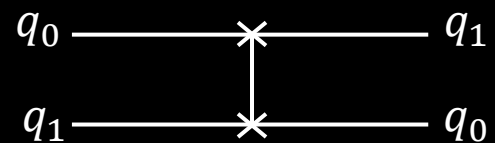


Classical NOT Gate



LET US LOOK AT SOME QUANTUM GATES

Quantum SWAP Gate

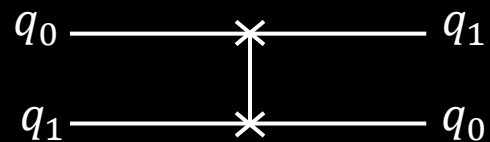


Truth Table

(q_0, q_1)	SWAP (q_0, q_1)
(0,0)	(0,0)
(0,1)	(1,0)
(1,0)	(0,1)
(1,1)	(1,1)

LET US LOOK AT SOME QUANTUM GATES

Quantum SWAP Gate



What about this?

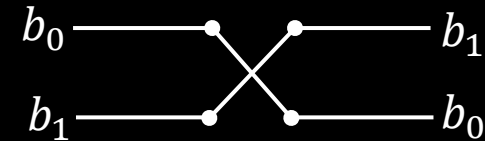
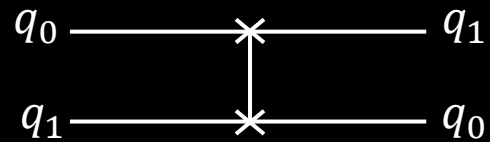
Truth Table

(q_0, q_1)	$\text{SWAP}(q_0, q_1)$
(0,0)	(0,0)
(0,1)	(1,0)
(1,0)	(0,1)
(1,1)	(1,1)

This is the classical SWAP operation!

LET US LOOK AT SOME QUANTUM GATES

Quantum SWAP Gate



Truth Table

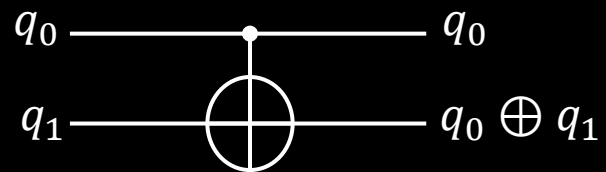
(q_0, q_1)	SWAP(q_0, q_1)
(0,0)	(0,0)
(0,1)	(1,0)
(1,0)	(0,1)
(1,1)	(1,1)



Classical SWAP

LET US LOOK AT SOME QUANTUM GATES

Quantum CNOT Gate

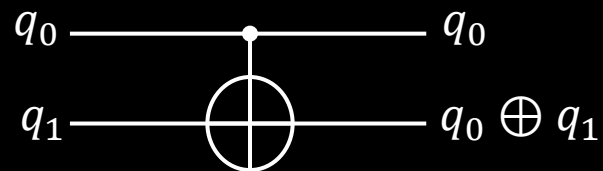


Truth Table

(q_0, q_1)	$\text{CNOT}(q_0, q_1)$
(0,0)	(0,0)
(0,1)	(0,1)
(1,0)	(1,1)
(1,1)	(1,0)

LET US LOOK AT SOME QUANTUM GATES

Quantum CNOT Gate



Truth Table

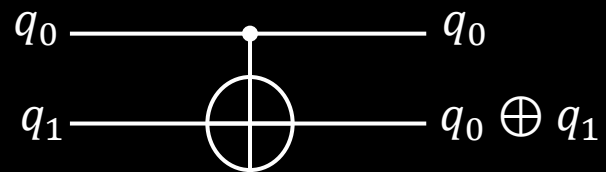
(q_0, q_1)	$\text{CNOT}(q_0, q_1)$
(0,0)	(0,0)
(0,1)	(0,1)
(1,0)	(1,1)
(1,1)	(1,0)

Is this possible classically?

This is the classical XOR gate!

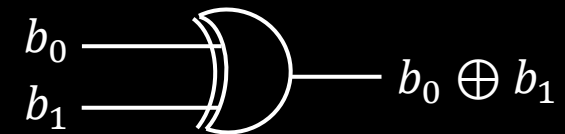
LET US LOOK AT SOME QUANTUM GATES

Quantum CNOT Gate



Truth Table

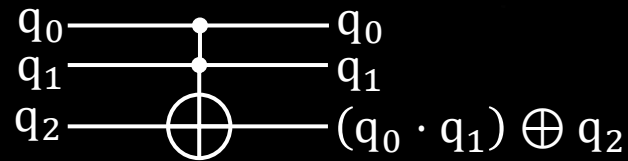
(q_0, q_1)	CNOT(q_0, q_1)
(0,0)	(0,0)
(0,1)	(0,1)
(1,0)	(1,1)
(1,1)	(1,0)



Classical XOR Gate

LET US LOOK AT SOME QUANTUM GATES

Quantum CCNOT Gate

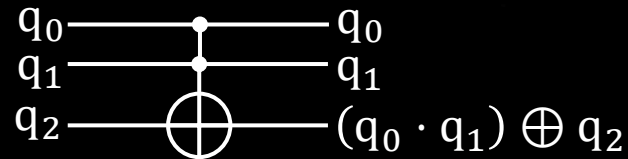


Truth Table

(q_0, q_1, q_2)	CCNOT (q_0, q_1, q_2)
(0,0,0)	(0,0,0)
(0,0,1)	(0,0,1)
(0,1,0)	(0,1,0)
(0,1,1)	(0,1,1)
(1,0,0)	(1,0,0)
(1,0,1)	(1,0,1)
(1,1,0)	(1,1,1)
(1,1,1)	(1,1,1)

LET US LOOK AT SOME QUANTUM GATES

Quantum CCNOT Gate



Truth Table

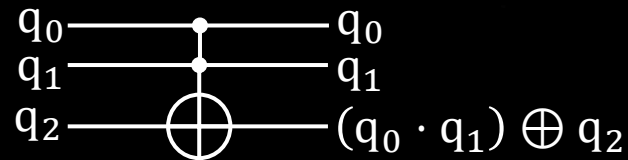
(q_0, q_1, q_2)	CCNOT (q_0, q_1, q_2)
(0,0,0)	(0,0,0)
(0,0,1)	(0,0,1)
(0,1,0)	(0,1,0)
(0,1,1)	(0,1,1)
(1,0,0)	(1,0,0)
(1,0,1)	(1,0,1)
(1,1,0)	(1,1,1)
(1,1,1)	(1,1,1)

Is this classically possible?

Yes! But a bit tricky though!

LET US LOOK AT SOME QUANTUM GATES

Quantum CCNOT Gate

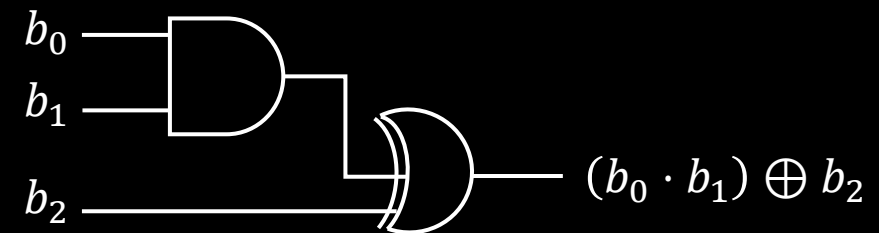


Truth Table

(q_0, q_1, q_2)	CCNOT(q_0, q_1, q_2)
(0,0,0)	(0,0,0)
(0,0,1)	(0,0,1)
(0,1,0)	(0,1,0)
(0,1,1)	(0,1,1)
(1,0,0)	(1,0,0)
(1,0,1)	(1,0,1)
(1,1,0)	(1,1,1)
(1,1,1)	(1,1,1)



Classical AND with XOR Gate



LET US LOOK AT SOME QUANTUM GATES

Quantum S Gate



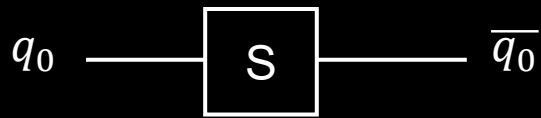
Truth Table

q_0	$S(q_0)$
0	0
1	$i1$

Complex number i ,
 $i = \sqrt{-1}$

LET US LOOK AT SOME QUANTUM GATES

Quantum S Gate



Truth Table

q_0	$S(q_0)$
0	0
1	$i1$

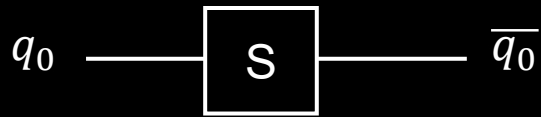
Complex number i ,
 $i = \sqrt{-1}$

Can this be corrected classically?

Very simple! Don't do anything!

LET US LOOK AT SOME QUANTUM GATES

Quantum S Gate



Truth Table

q_0	$S(q_0)$
0	0
1	1



Classical Identity (nop)



GATES SUMMARY

- Most quantum gates have such classical equivalents.
- Because of measurements
- Measurements collapses the quantum state to a classical state
- Quantum Gates added before measurement
 - Can be nullified by classical gates

GATES SUMMARY

- Most quantum gates have such classical equivalents.
- Because of measurements
- Measurements collapses the quantum state to a classical state
- Quantum Gates added before measurement
 - Can be nullified by classical gates

- But,

GATES SUMMARY

- Quantum Hadamard Gate



Creates superposition



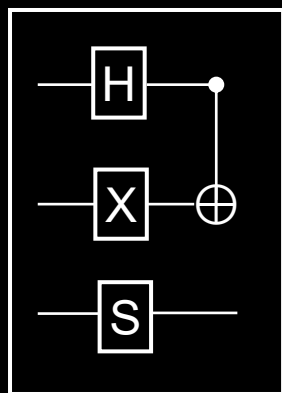
NO CLASSICAL EQUIVALENT!



CLOAQ: Chaos for the Cloud, Clarity for the User

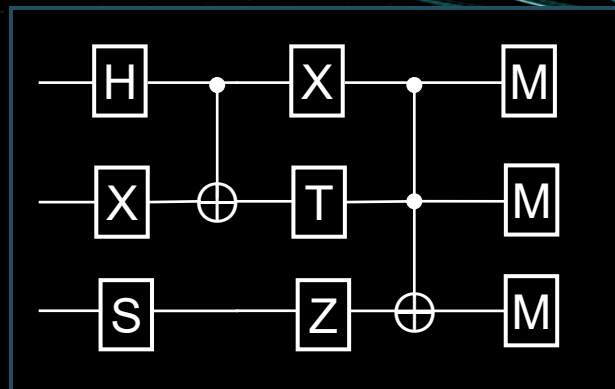
WORKFLOW OF OUR TECHNIQUE

Original Circuit



With
random
gates

Obfuscated Circuit



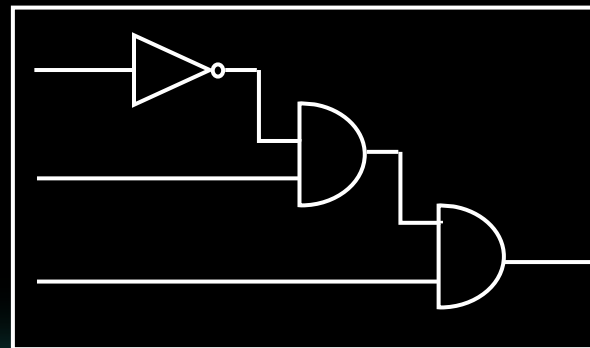
*Untrusted
Quantum
Computer*



*Corrupted
Measurement*



With
classical
logic gates

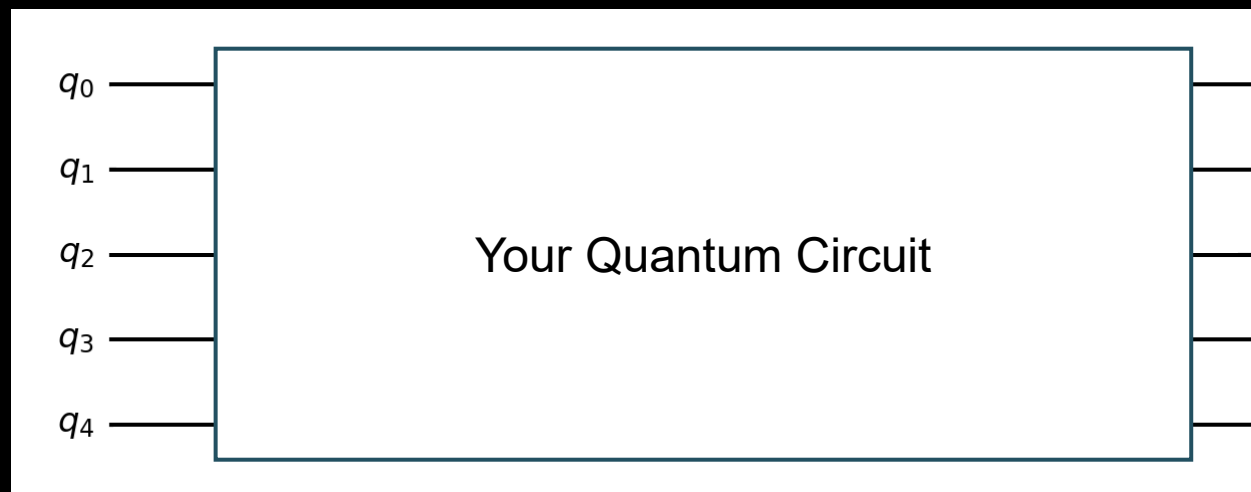


*Corrected
Measurement*

QUANTUM ENCRYPTION

- Consider a circuit of your choice.

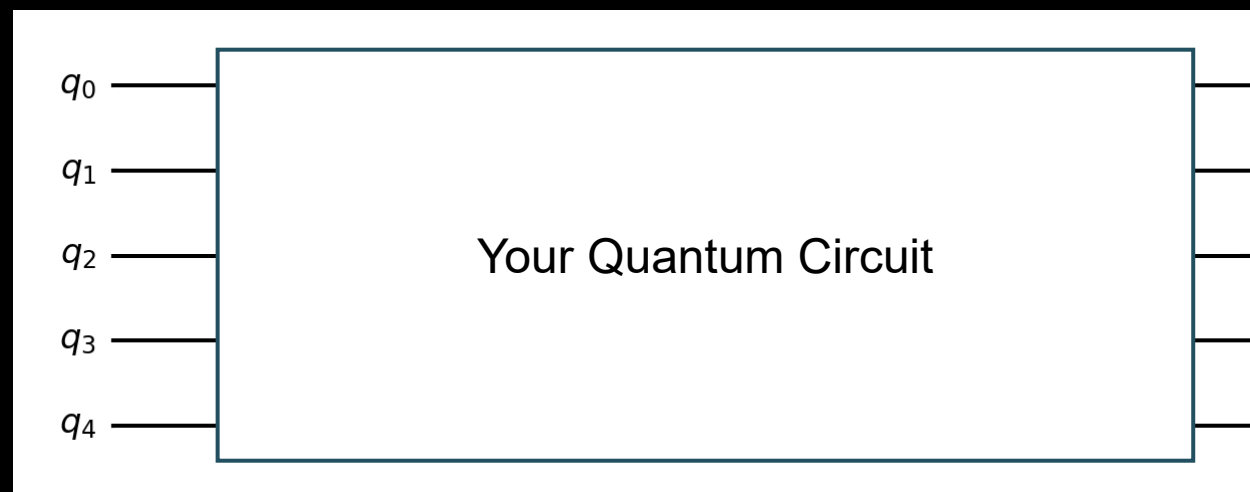
Original Circuit



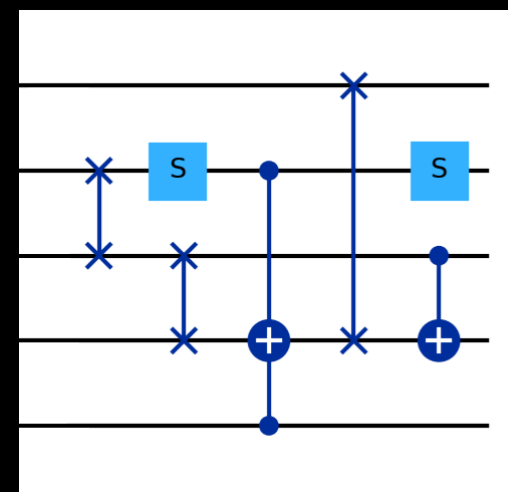
QUANTUM ENCRYPTION

- Add “encryptor gates” to corrupt the functionality.

Original Circuit



Encryptor Gates

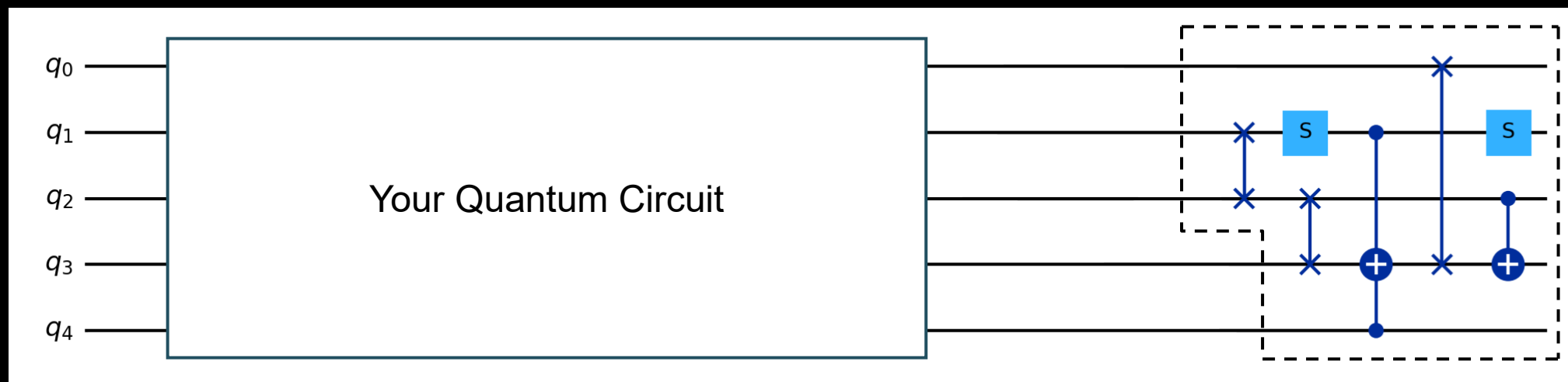


QUANTUM ENCRYPTION

- Add “encryptor gates” to corrupt the functionality.

Original Circuit

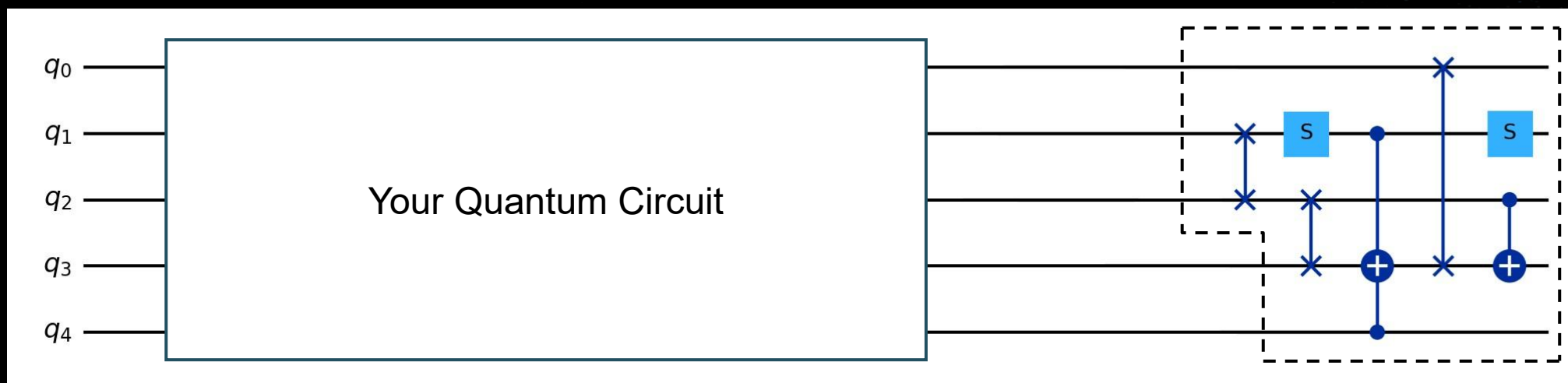
Encryptor Gates



KEY DERIVATION

Original Circuit

Encryptor Gates



Unique identity for each encryptor gate

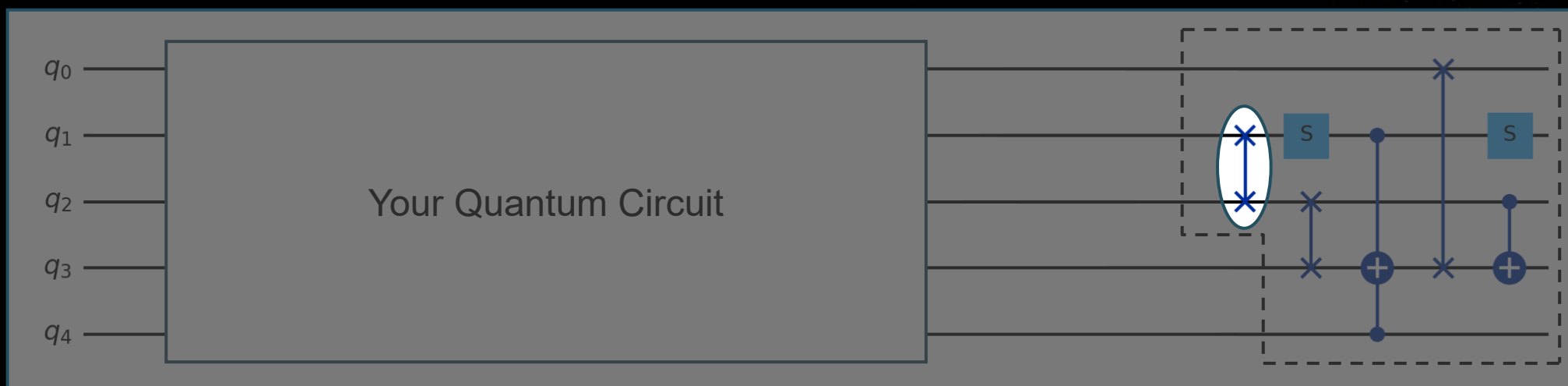


- 0: X
- 1: CNOT
- 2: SWAP
- 3: CCNOT
- 4: CSWAP
- 5: S

KEY DERIVATION

Original Circuit

Encryptor Gates



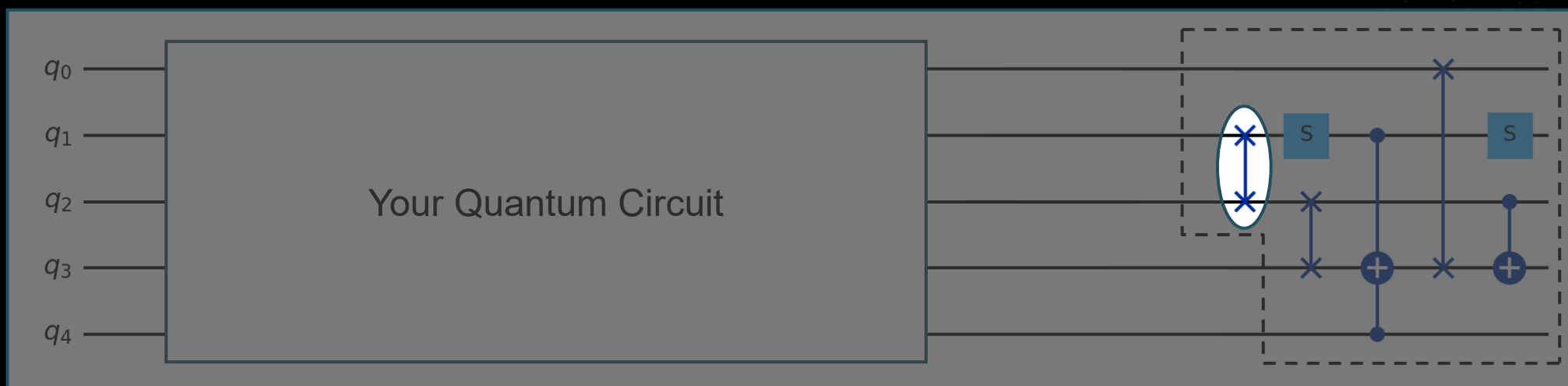
Key =

- 0: X
- 1: CNOT
- 2: SWAP
- 3: CCNOT
- 4: CSWAP
- 5: S

KEY DERIVATION

Original Circuit

Encryptor Gates



Key = 2#

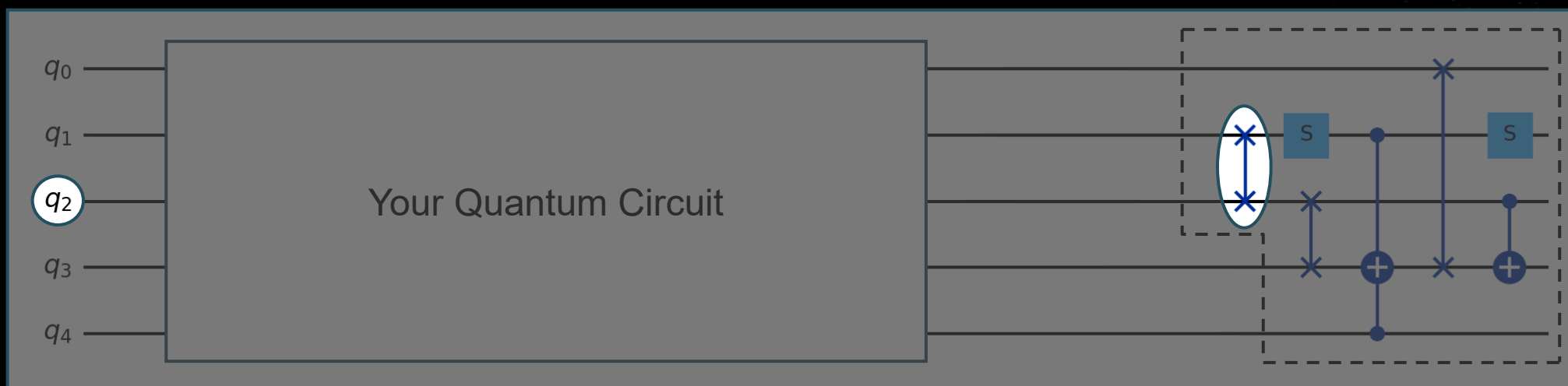
@ separates gate-qubit set, # separates gate from qubits, | separates qubits

- 0: X
- 1: CNOT
- 2: SWAP
- 3: CCNOT
- 4: CSWAP
- 5: S

KEY DERIVATION

Original Circuit

Encryptor Gates



Key = 2#2

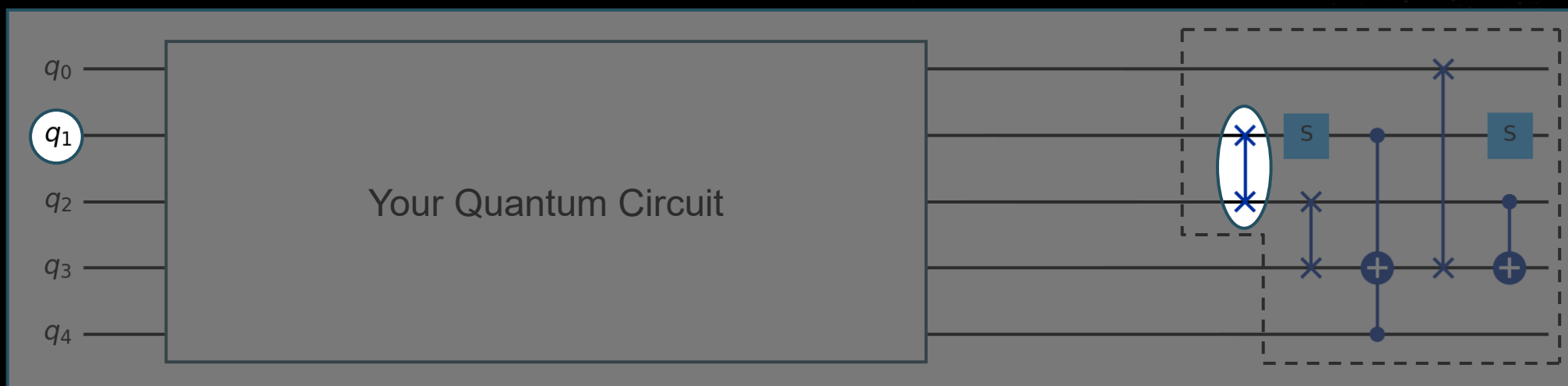
@ separates gate-qubit set, # separates gate from qubits, | separates qubits

- 0: X
- 1: CNOT
- 2: SWAP
- 3: CCNOT
- 4: CSWAP
- 5: S

KEY DERIVATION

Original Circuit

Encryptor Gates



Key = 2#2|1@

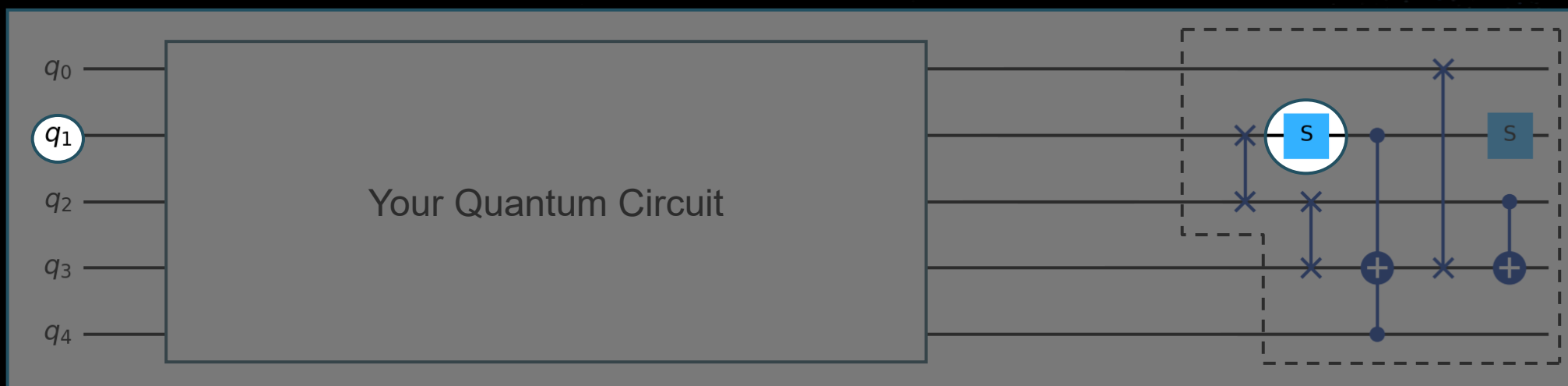
@ separates gate-qubit set, # separates gate from qubits, | separates qubits

- 0: X
- 1: CNOT
- 2: SWAP
- 3: CCNOT
- 4: CSWAP
- 5: S

KEY DERIVATION

Original Circuit

Encryptor Gates



Key = 2#2|1@5#1@

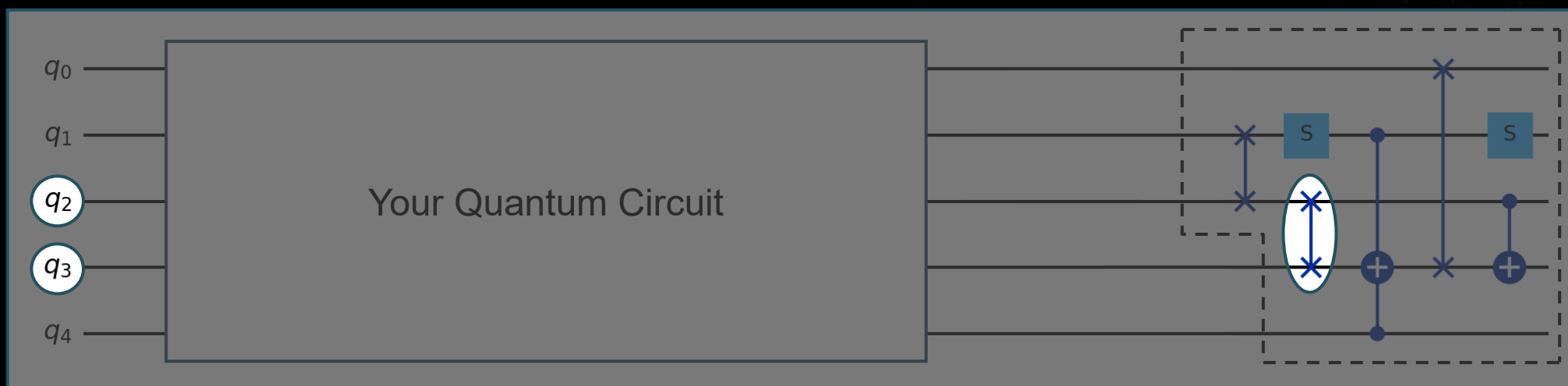
@ separates gate-qubit set, # separates gate from qubits, | separates qubits

- 0: X
- 1: CNOT
- 2: SWAP
- 3: CCNOT
- 4: CSWAP
- 5: S

KEY DERIVATION

Original Circuit

Encryptor Gates



Key = 2#2|1@5#1@2#2|3@

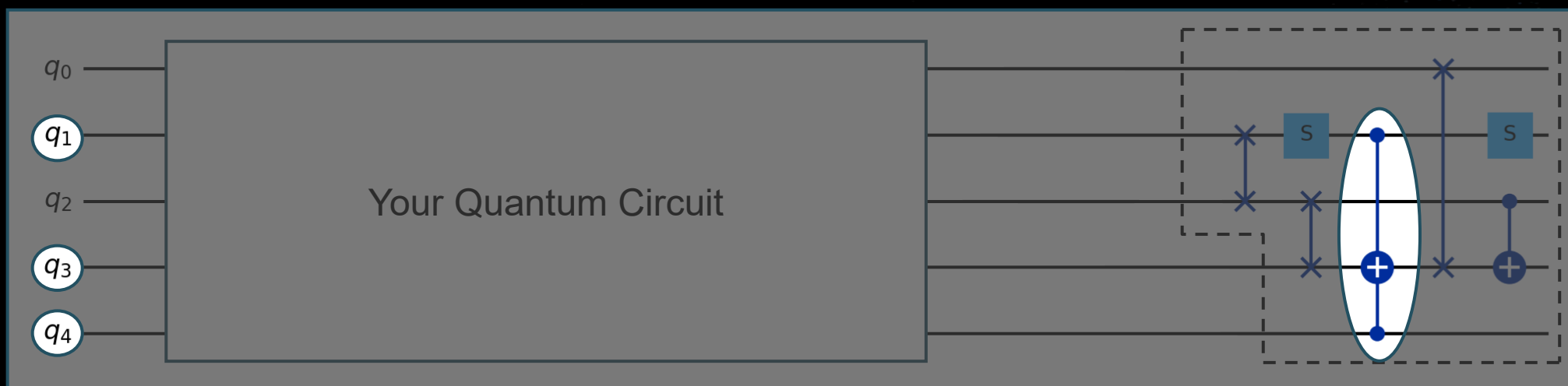
@ separates gate-qubit set, # separates gate from qubits, | separates qubits

- 0: X
- 1: CNOT
- 2: SWAP
- 3: CCNOT
- 4: CSWAP
- 5: S

KEY DERIVATION

Original Circuit

Encryptor Gates



Key = 2#2|1@5#1@2#2|3@3#1|4|3@

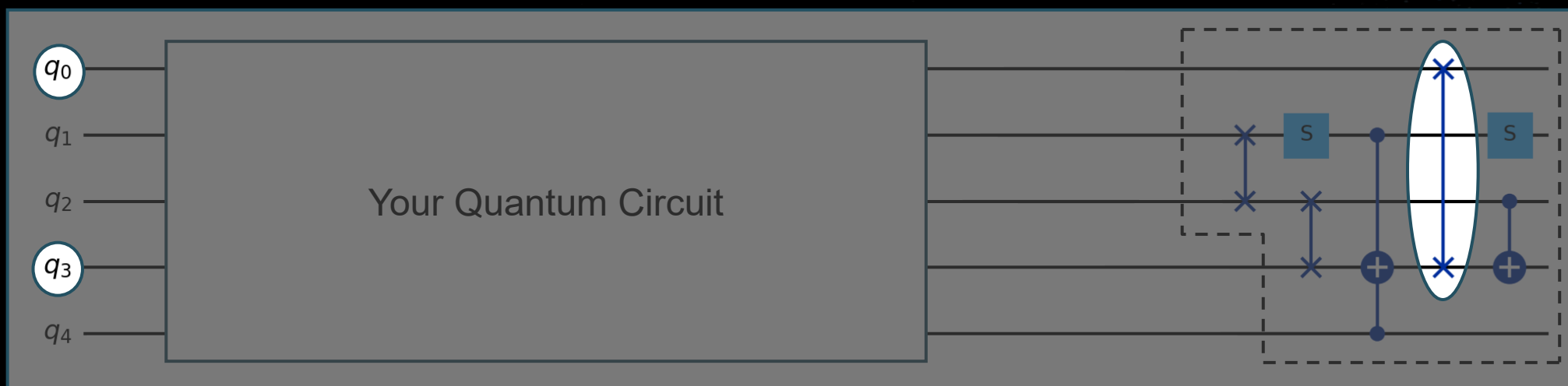
@ separates gate-qubit set, # separates gate from qubits, | separates qubits

- 0: X
- 1: CNOT
- 2: SWAP
- 3: CCNOT
- 4: CSWAP
- 5: S

KEY DERIVATION

Original Circuit

Encryptor Gates



Key = 2#2|1@5#1@2#2|3@3#1|4|3@2#3|0@

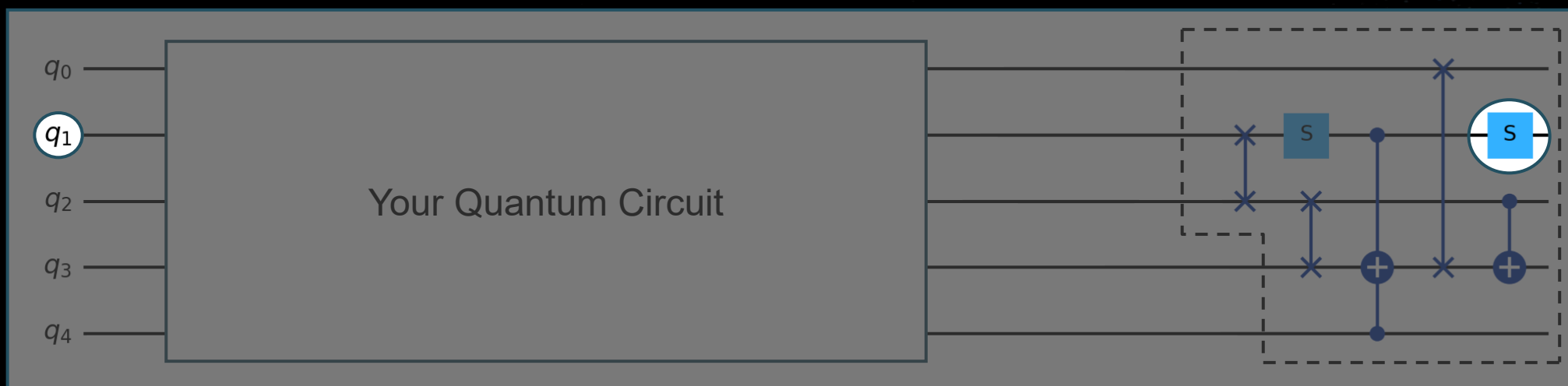
@ separates gate-qubit set, # separates gate from qubits, | separates qubits

- 0: X
- 1: CNOT
- 2: SWAP
- 3: CCNOT
- 4: CSWAP
- 5: S

KEY DERIVATION

Original Circuit

Encryptor Gates



Key = 2#2|1@5#1@2#2|3@3#1|4|3@2#3|0@5#1@

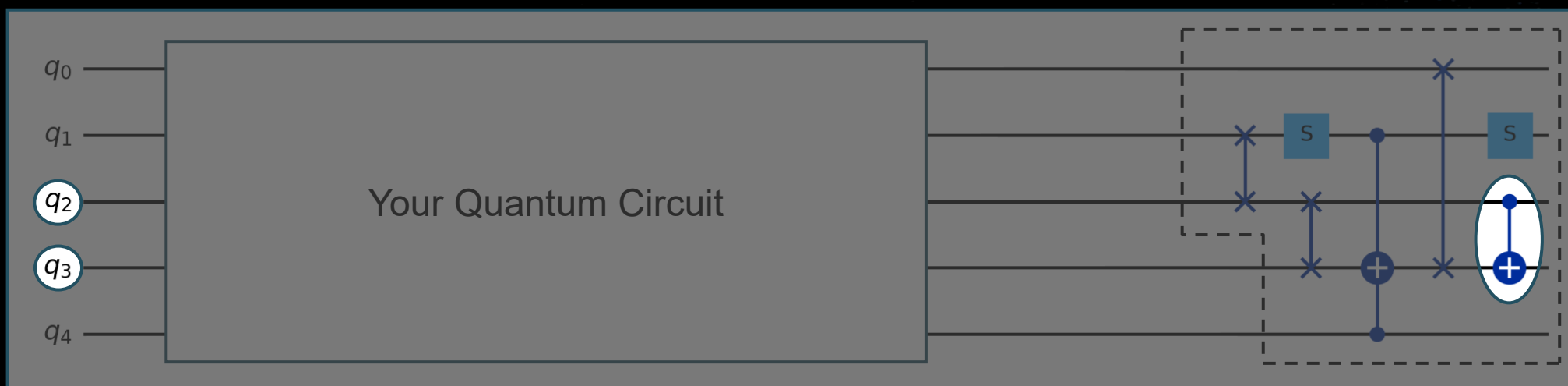
@ separates gate-qubit set, # separates gate from qubits, | separates qubits

- 0: X
- 1: CNOT
- 2: SWAP
- 3: CCNOT
- 4: CSWAP
- 5: S

KEY DERIVATION

Original Circuit

Encryptor Gates



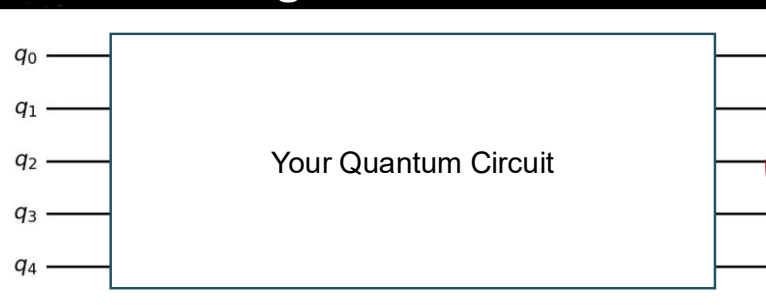
Key = 2#2|1@5#1@2#2|3@3#1|4|3@2#3|0@5#1@1#2|3

@ separates gate-qubit set, # separates gate from qubits, | separates qubits

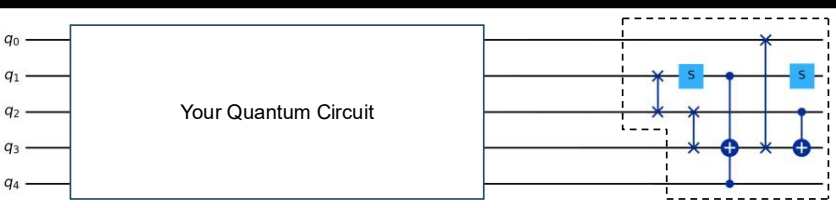
- 0: X
- 1: CNOT
- 2: SWAP
- 3: CCNOT
- 4: CSWAP
- 5: S

EXECUTION OF ENCRYPTED CIRCUIT

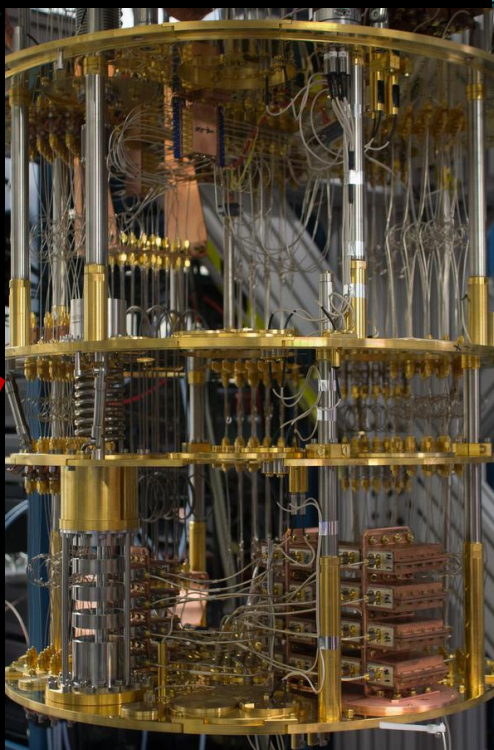
Original Circuit



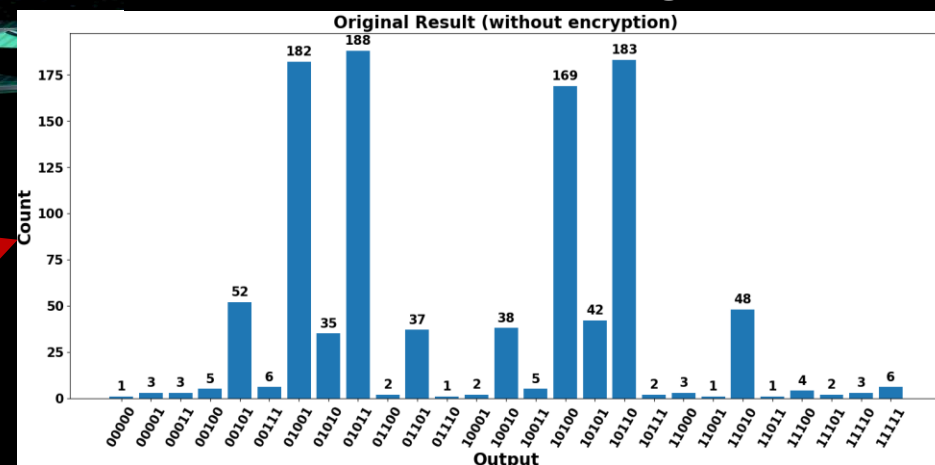
Encrypted Circuit



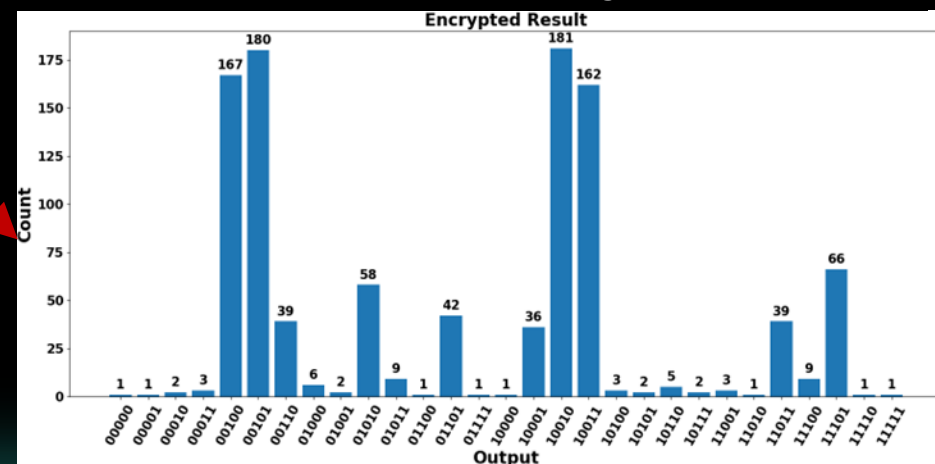
Quantum Computer



Measurement from original circuit



Measurement from encrypted circuit



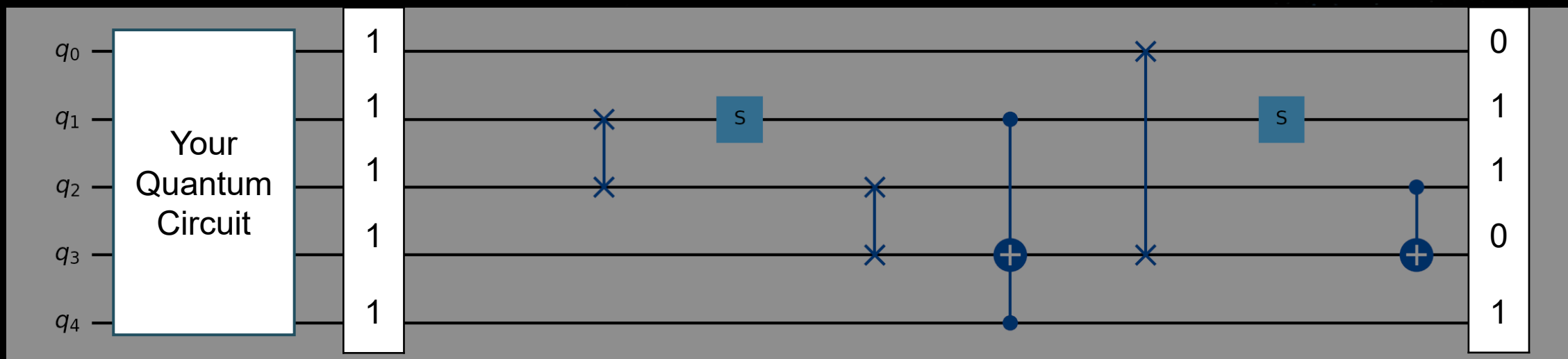


DECRYPTION

CLASSICAL DECRYPTION

Unencrypted Result

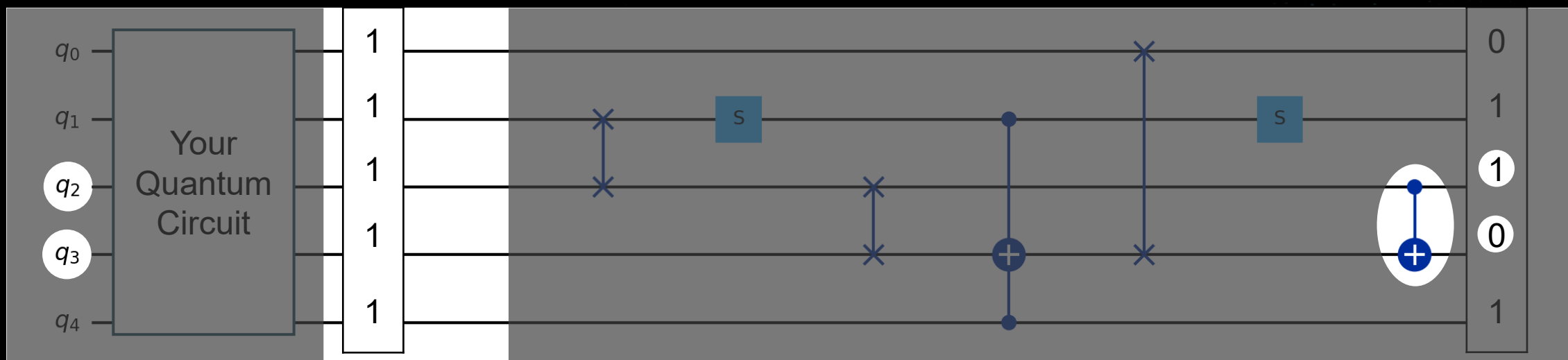
Corrupted Result



Key = 2#2|1@5#1@2#2|3@3#1|4|3@2#3|0@5#1@1#2|3

CLASSICAL DECRYPTION

Unencrypted Result



Key = 2#2|1@5#1@2#2|3@3#1|4|3@2#3|0@5#1@1#2|3

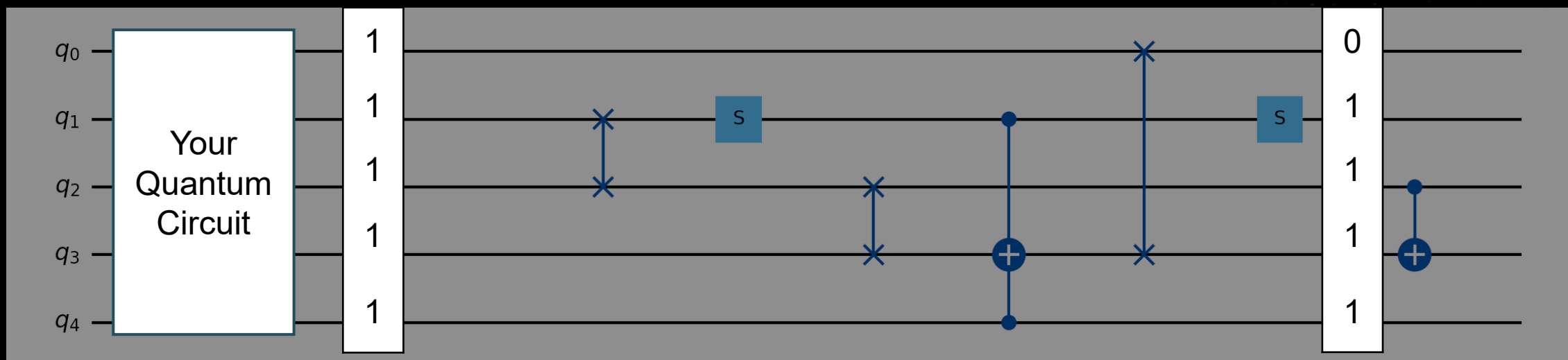


=



CLASSICAL DECRYPTION

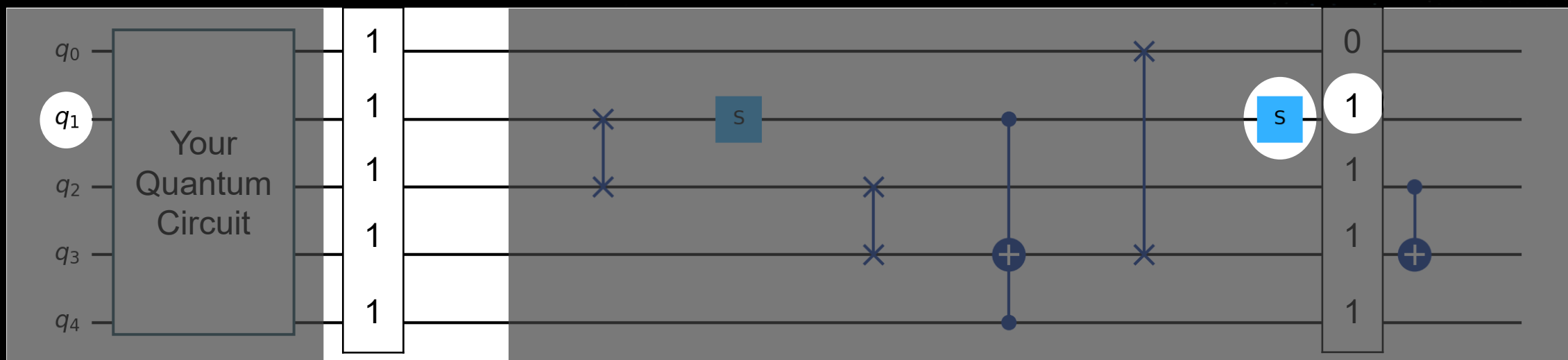
Unencrypted Result



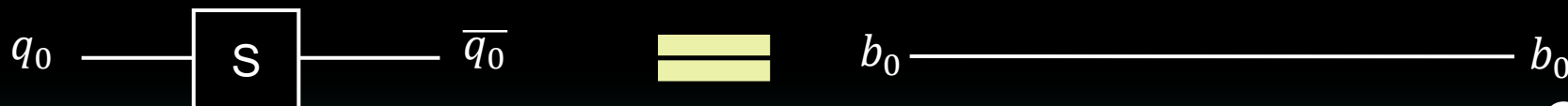
Key = 2#2|1@5#1@2#2|3@3#1|4|3@2#3|0@5#1@1#2|3

CLASSICAL DECRYPTION

Unencrypted Result

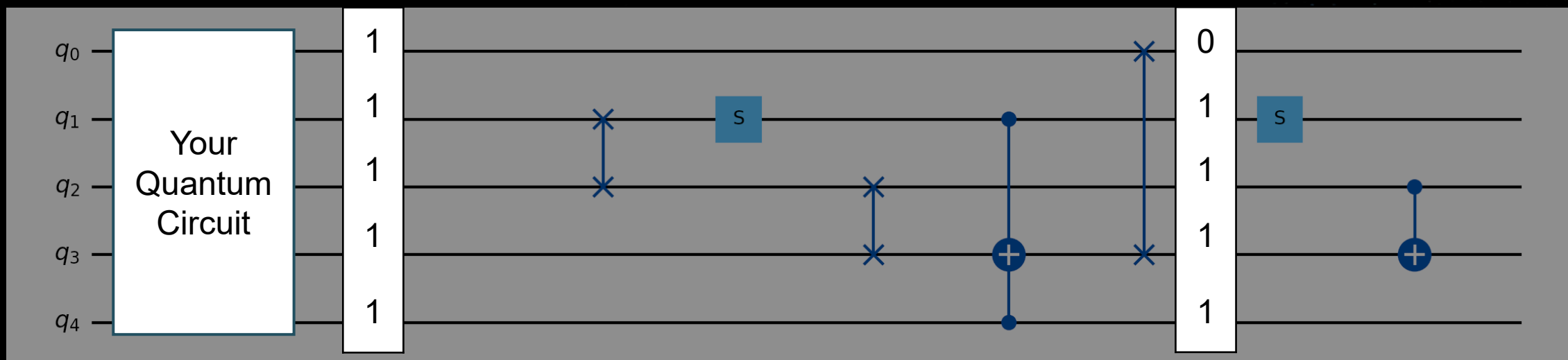


Key = 2#2|1@5#1@2#2|3@3#1|4|3@2#3|0@5#1@1#2|3



CLASSICAL DECRYPTION

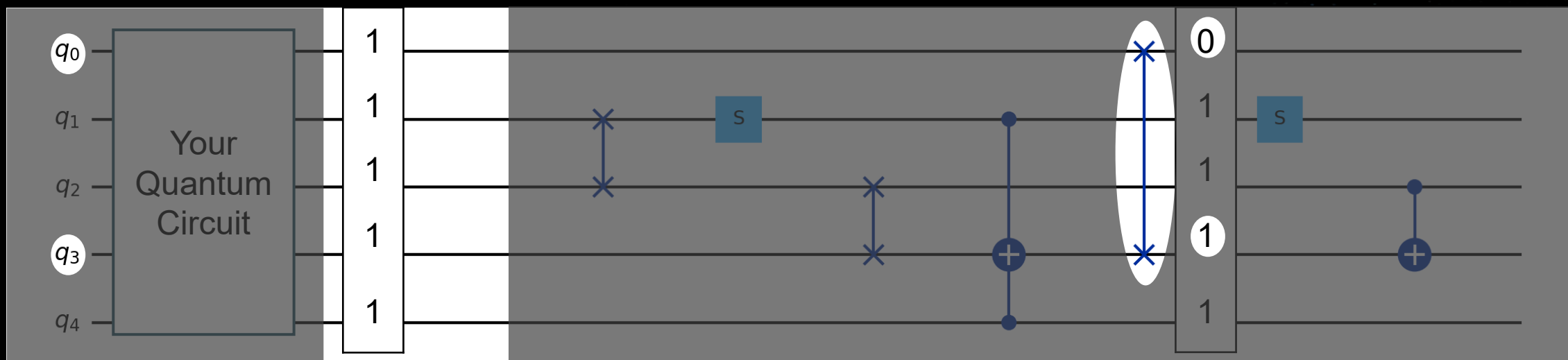
Unencrypted Result



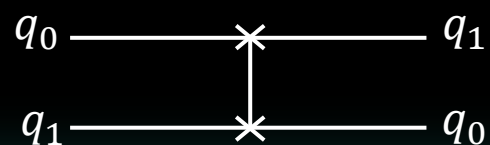
Key = 2#2|1@5#1@2#2|3@3#1|4|3@2#3|0@5#1@1#2|3

CLASSICAL DECRYPTION

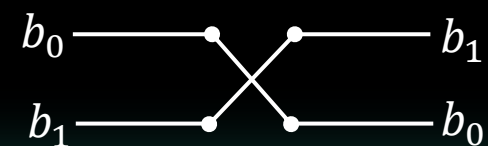
Unencrypted Result



Key = 2#2|1@5#1@2#2|3@3#1|4|3@2#3|0@5#1@1#2|3

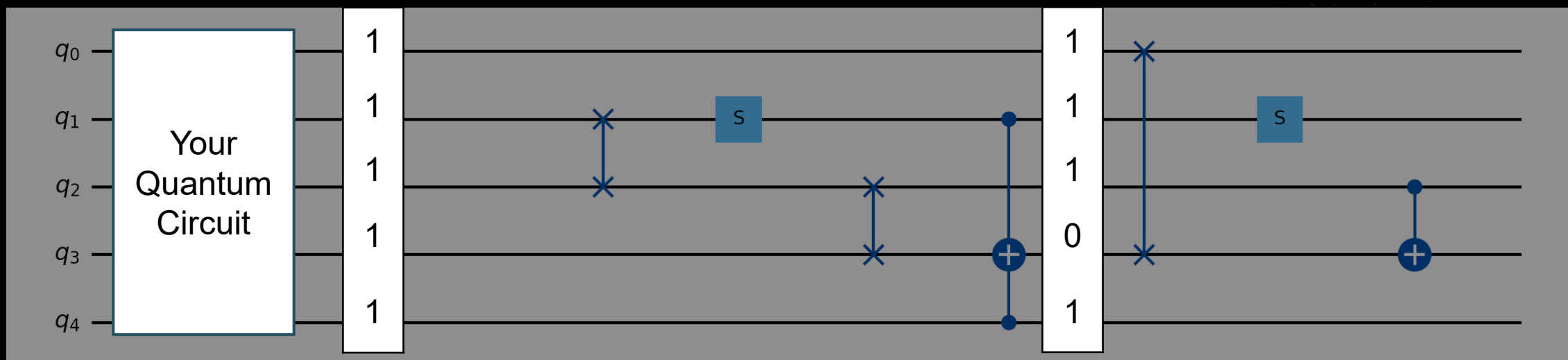


=



CLASSICAL DECRYPTION

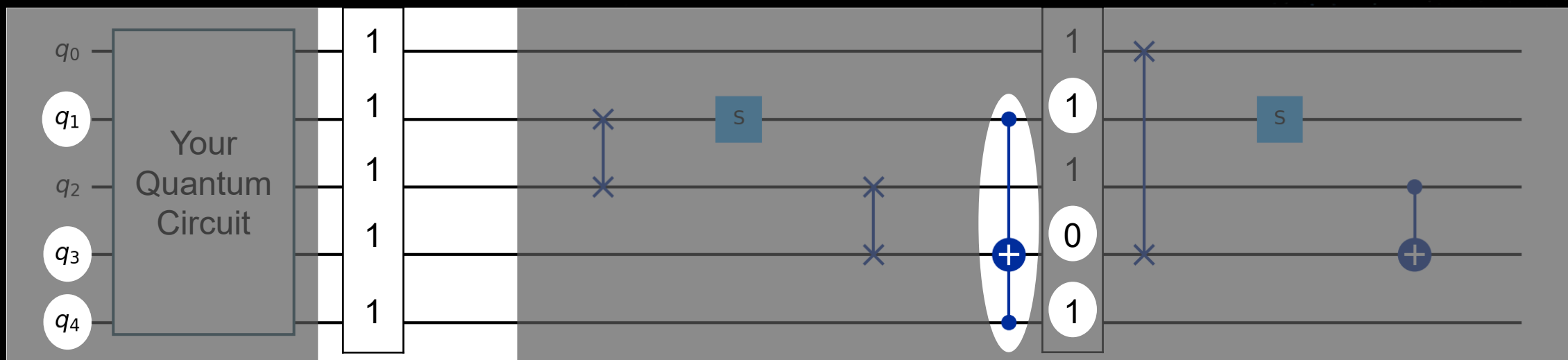
Unencrypted Result



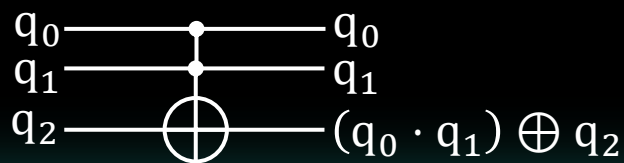
Key = 2#2|1@5#1@2#2|3@3#1|4|3@2#3|0@5#1@1#2|3

CLASSICAL DECRYPTION

Unencrypted Result

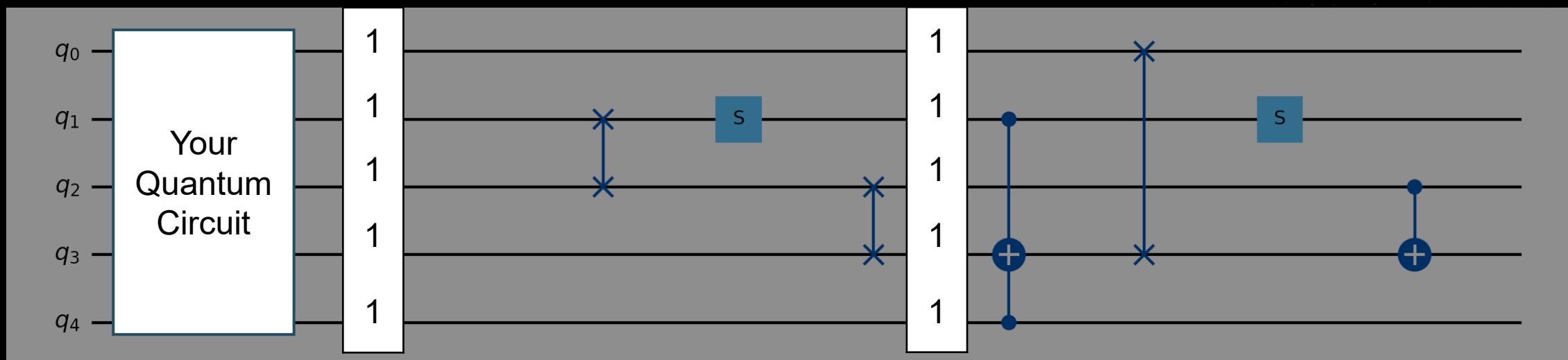


Key = 2#2|1@5#1@2#2|3@3#1|4|3@2#3|0@5#1@1#2|3



CLASSICAL DECRYPTION

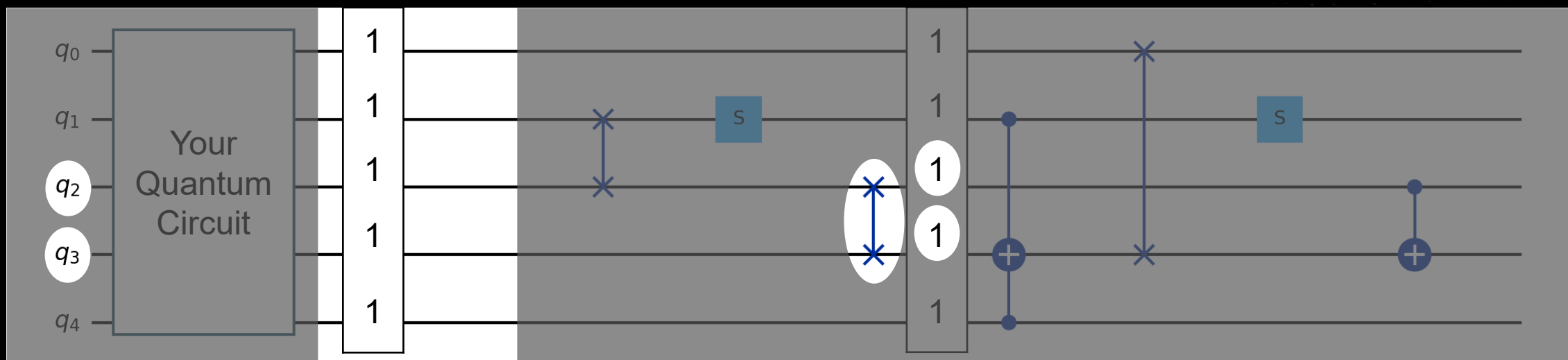
Unencrypted Result



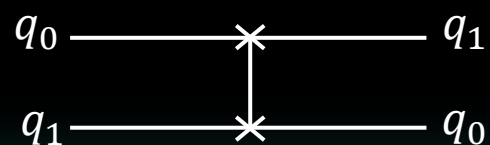
Key = 2#2|1@5#1@2#2|3@3#1|4|3@2#3|0@5#1@1#2|3

CLASSICAL DECRYPTION

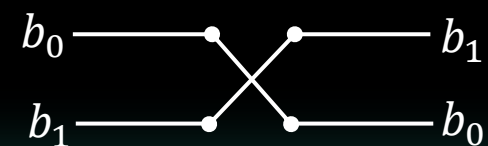
Unencrypted Result



Key = 2#2|1@5#1@2#2|3@3#1|4|3@2#3|0@5#1@1#2|3

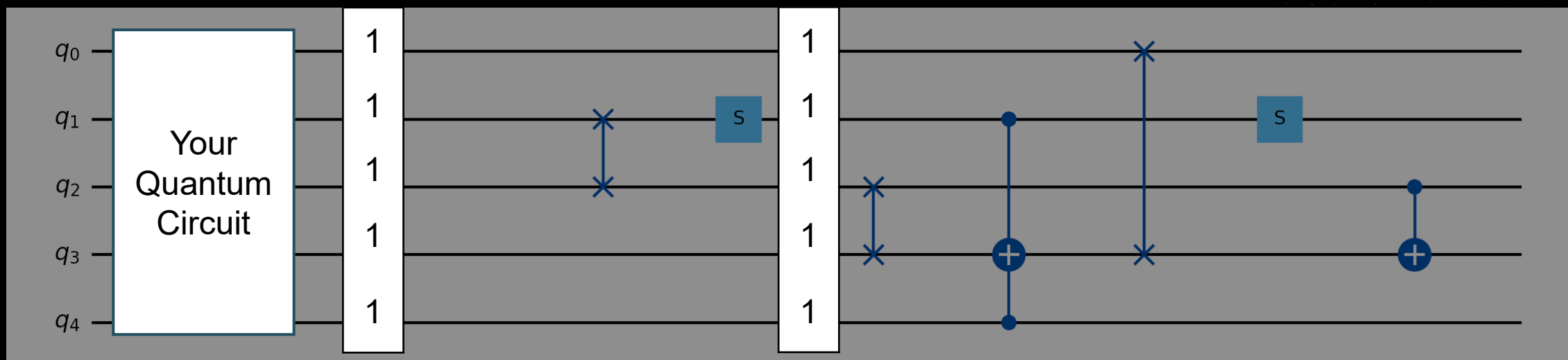


=



CLASSICAL DECRYPTION

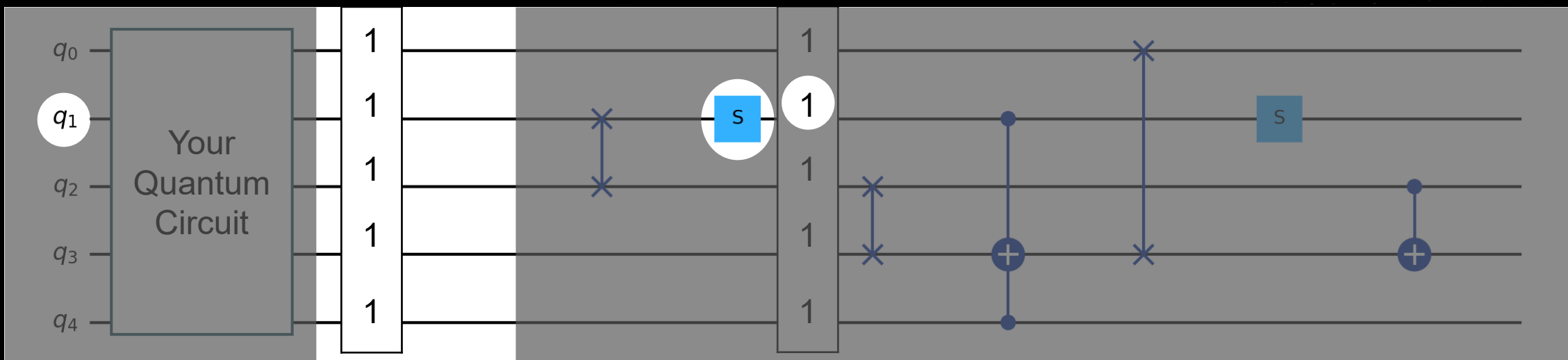
Unencrypted Result



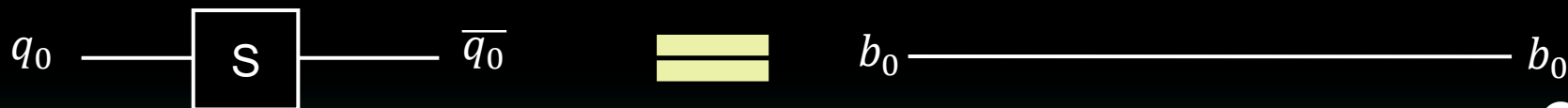
Key = 2#2|1@5#1@2#2|3@3#1|4|3@2#3|0@5#1@1#2|3

CLASSICAL DECRYPTION

Unencrypted Result

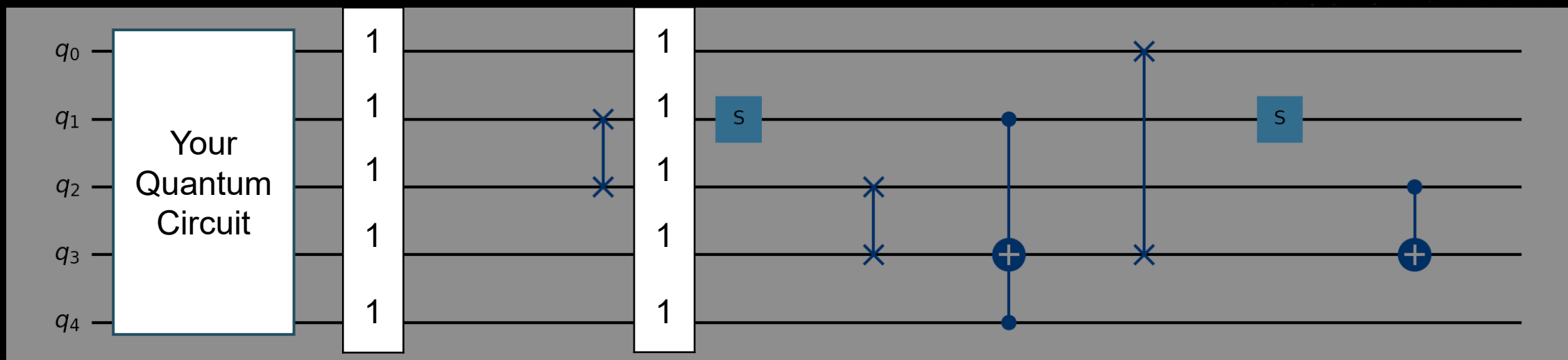


Key = 2#2|1@5#1@2#2|3@3#1|4|3@2#3|0@5#1@1#2|3



CLASSICAL DECRYPTION

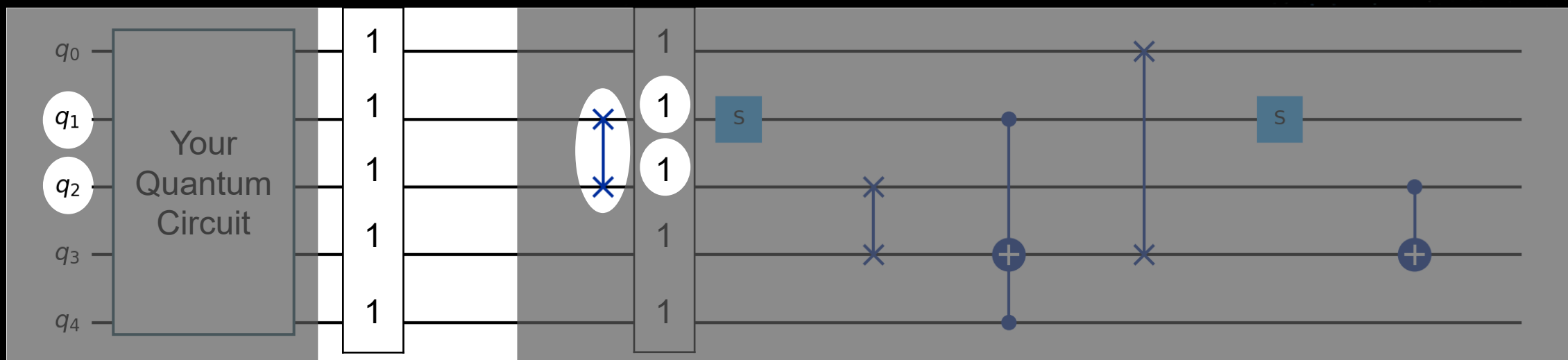
Unencrypted Result



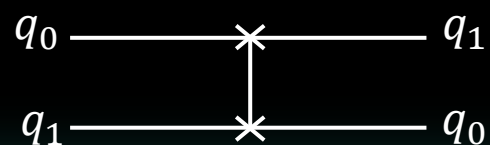
Key = 2#2|1@5#1@2#2|3@3#1|4|3@2#3|0@5#1@1#2|3

CLASSICAL DECRYPTION

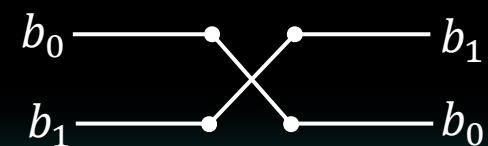
Unencrypted Result



Key = 2#2|1@5#1@2#2|3@3#1|4|3@2#3|0@5#1@1#2|3



=

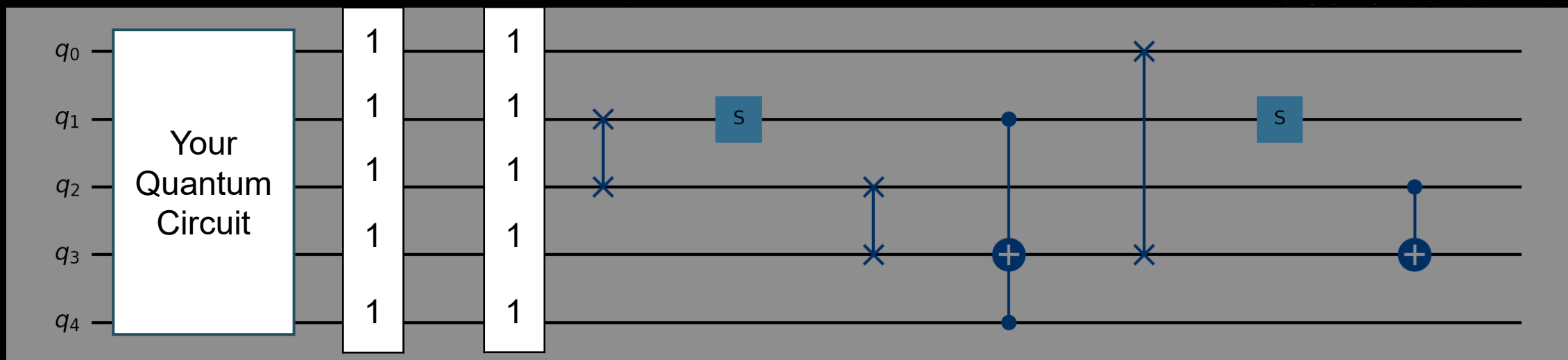


CLASSICAL DECRYPTION

Unencrypted Result

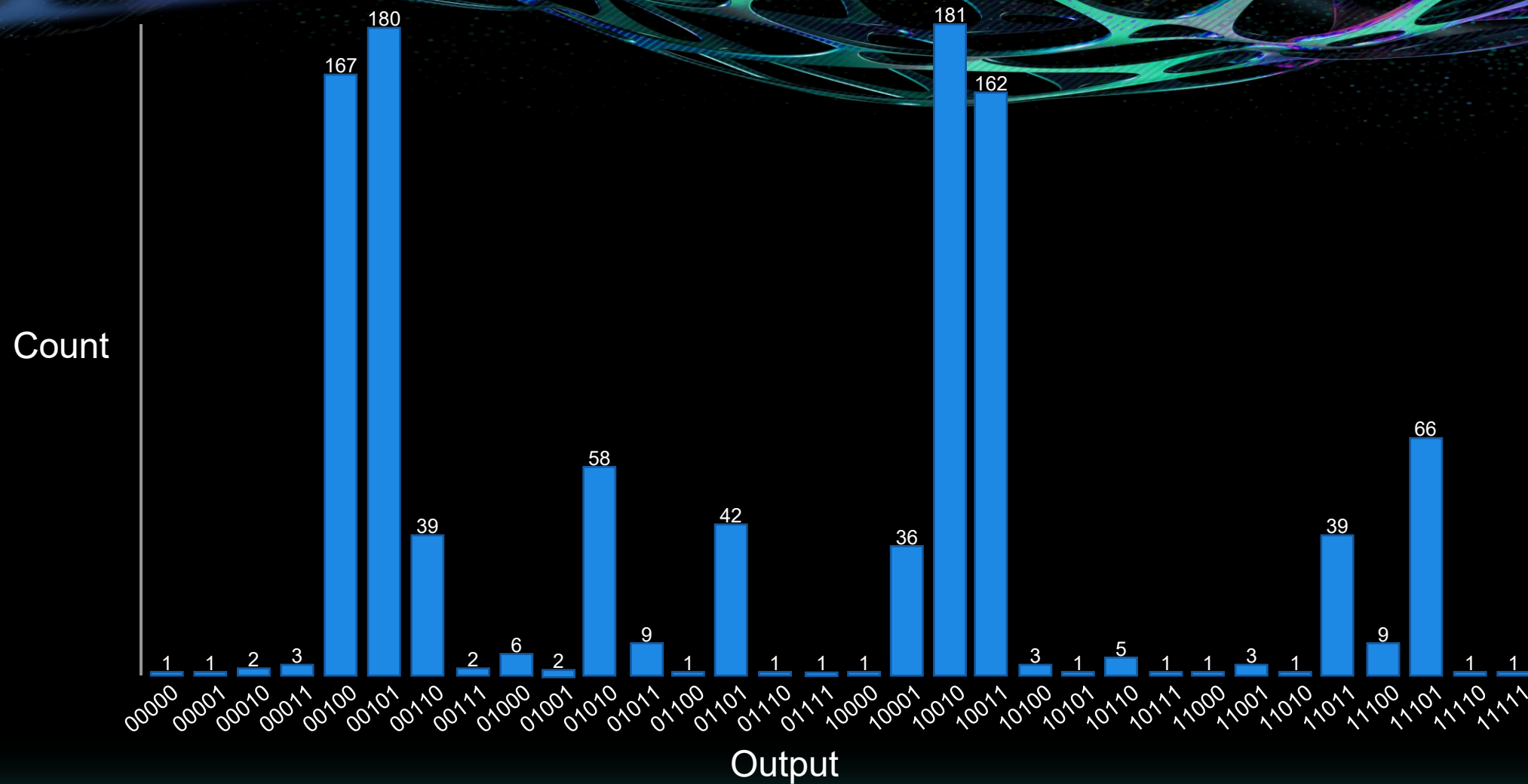


Decrypted Result

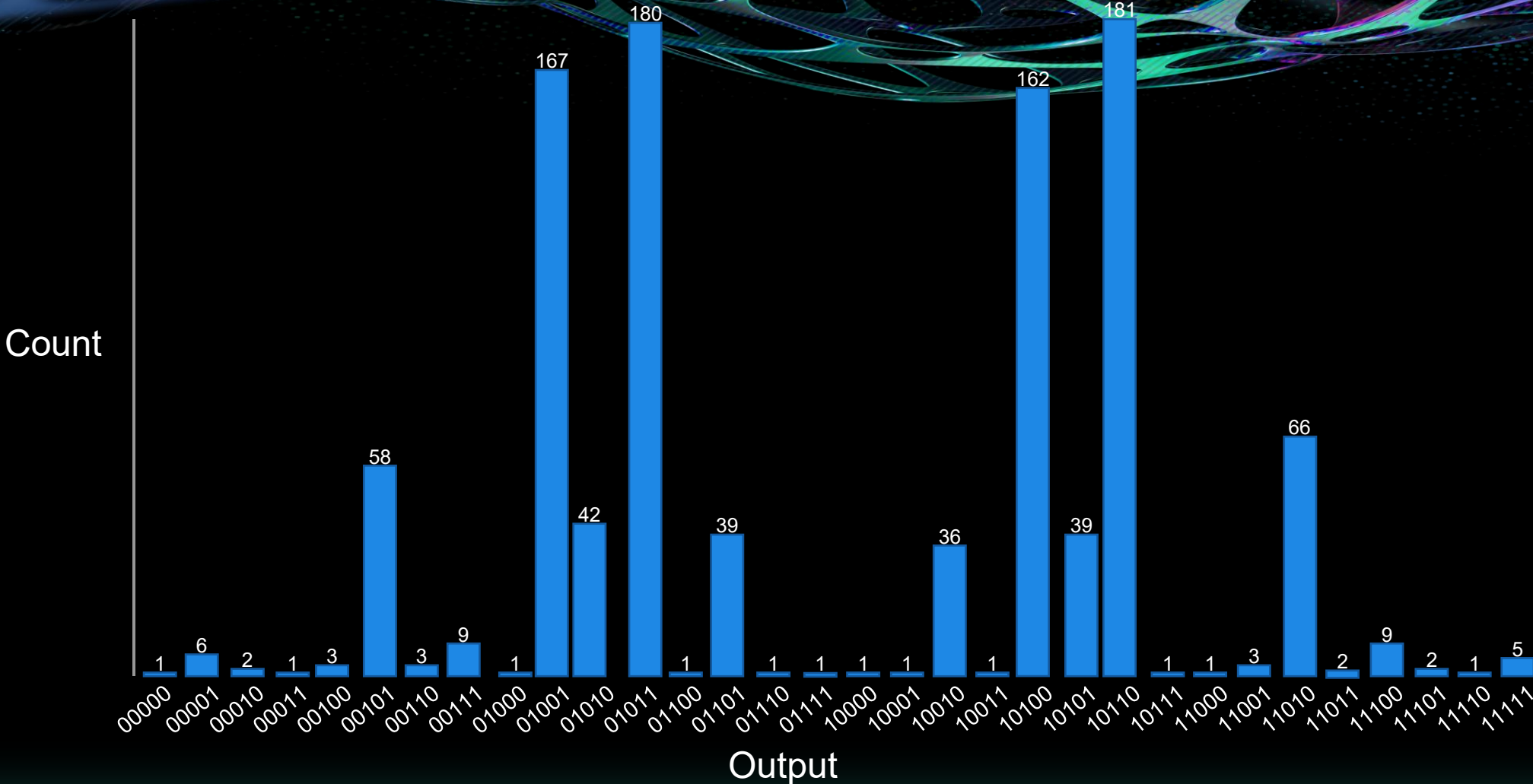


Key = 2#2|1@5#1@2#2|3@3#1|4|3@2#3|0@5#1@1#2|3

DECRYPT THE RESULTS



DECRYPT THE RESULTS

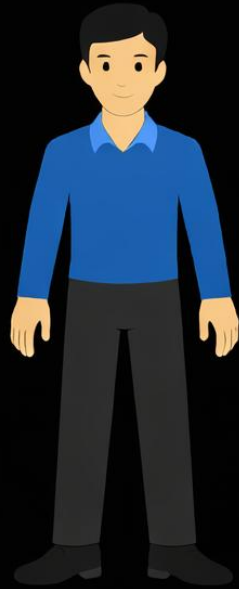




EVALUATION METRICS

EVALUATION METRICS

- Total Variation Distance (TVD)
 - Indicates how different the measurement outcomes are, after obfuscation.



Reference position
TVD = 0 (min)



Small deviation
TVD = 0.3 (low)



High difference
TVD = 1 (max)

Higher the TVD, higher the divergence and harder for the cloud to understand!

EVALUATION METRICS

Prior Works	Average TVD
Dummy Gate Insertion	0.41
Phase Modification	0.502
Logic Locking	0.732
CLOAQ (Our solution)	0.809

Comparison of average TVD for Cloaq with that of prior works

CLOAQ provides a more obfuscated output relative to prior works

WINTER WILL COME AND GO.. QUANTUM IS HERE TO STAY

Future of Coding is
Quantum + Classical

QUANTUM
IS COMING

GAME OF
THRONES

WINTER WILL COME AND GO.. QUANTUM IS HERE TO STAY

Future of Coding is
Quantum + Classical

You will add
quantum features to
your code base

QUANTUM
IS COMING

GAME OF
THRONES

WINTER WILL COME AND GO.. QUANTUM IS HERE TO STAY

Future of Coding is
Quantum + Classical

You will add
quantum features to
your code base

In the near term,
your code runs in a
computer far far
away

QUANTUM
IS COMING

GAME OF
THRONES

WINTER WILL COME AND GO.. QUANTUM IS HERE TO STAY

Future of Coding is
Quantum + Classical

You will add
quantum features to
your code base

In the near term,
your code runs in a
computer far far
away

And, if you care the
precious potions, it
makes

GAME OF
THRONES

WINTER WILL COME AND GO.. QUANTUM IS HERE TO STAY

Future of Coding is
Quantum + Classical

You will add
quantum features to
your code base

CLOAQ

In the near term,
your code runs in a
computer far far
away

And, if you care the
precious potions, it
makes

GAME OF
THRONES

WRAP-UP

ONLINE

<https://cloaq.obfusgate.com>

DOCKER

```
docker run -p 8000:8000 cloaq/cloaq:latest
```

WHITE PAPER

<https://doi.org/10.1145/3733817.3762698>