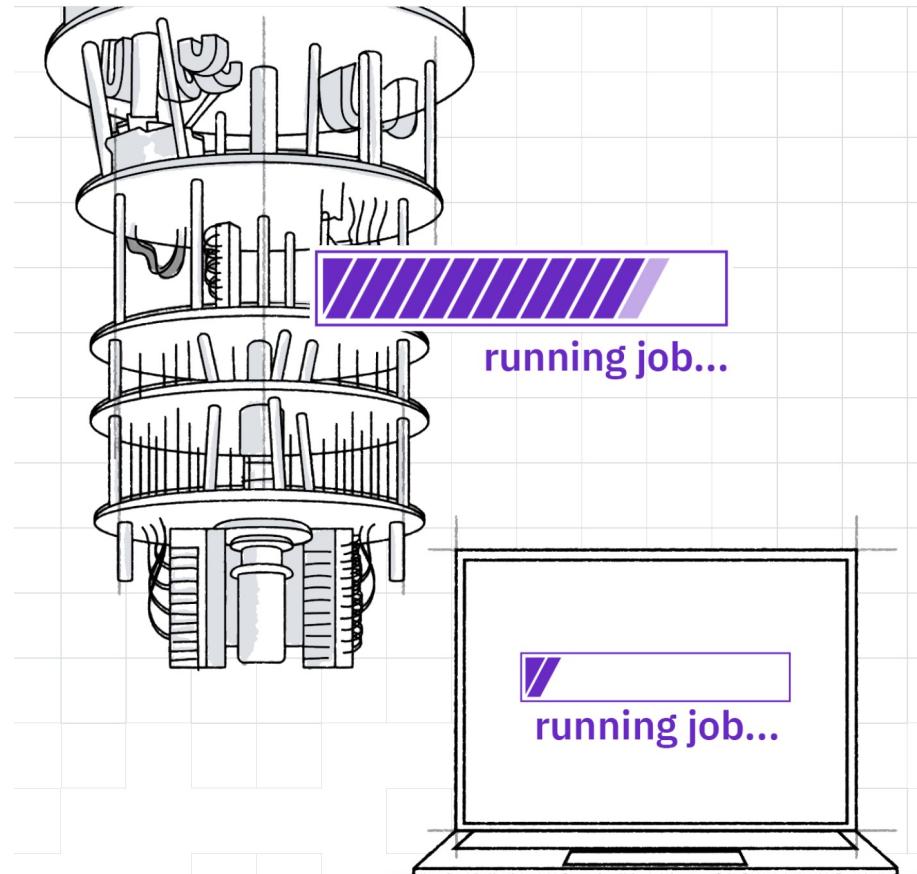


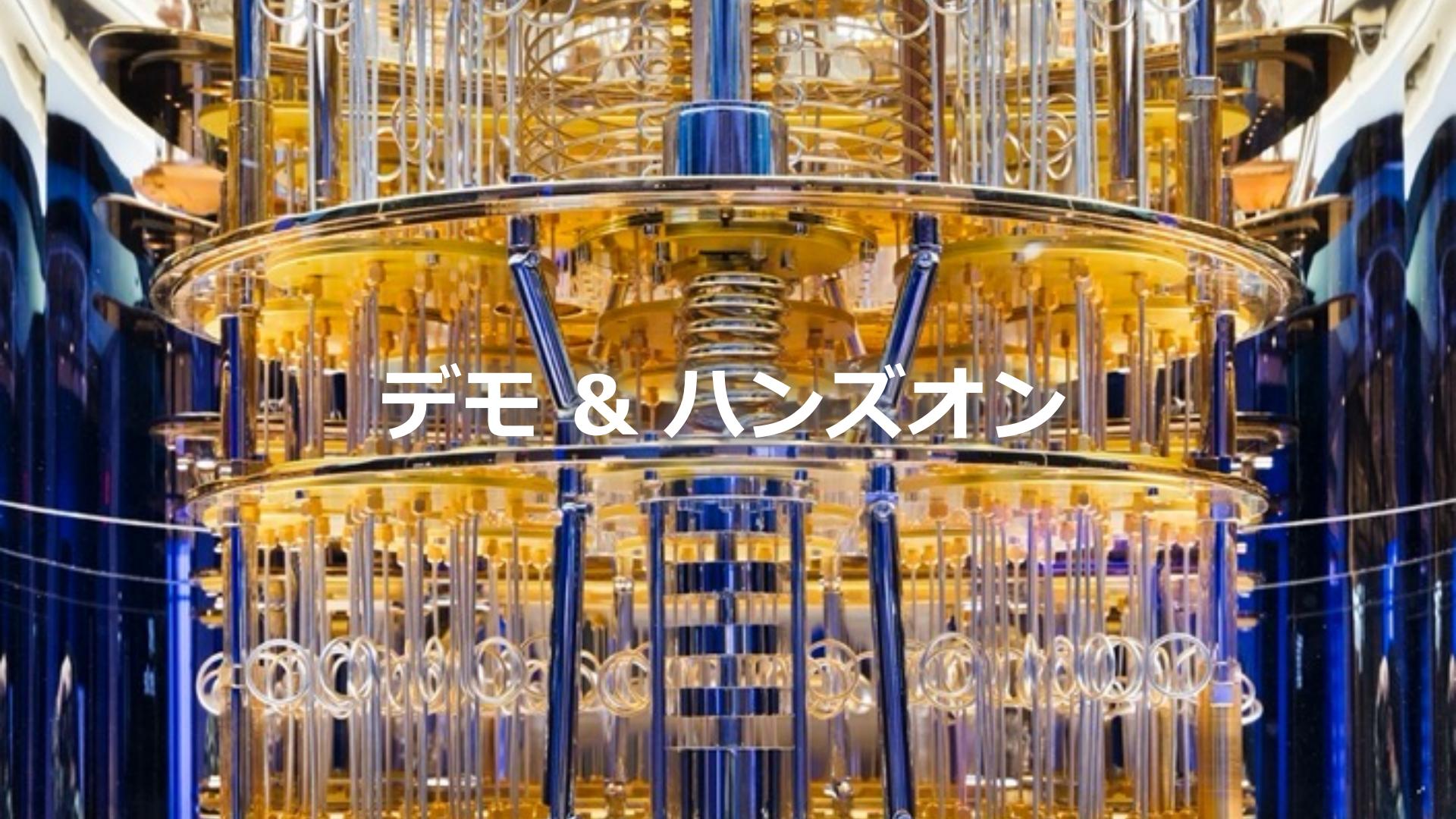
始めてみよう！ 量子コンピューター

沼田 祐史(Kifumi Numata)
Aug 2023



@kifuminumata



A close-up photograph of a quantum computing prototype. The device features a complex arrangement of blue and yellow cylindrical components, likely superconducting qubits and readout systems, all housed within a transparent protective enclosure.

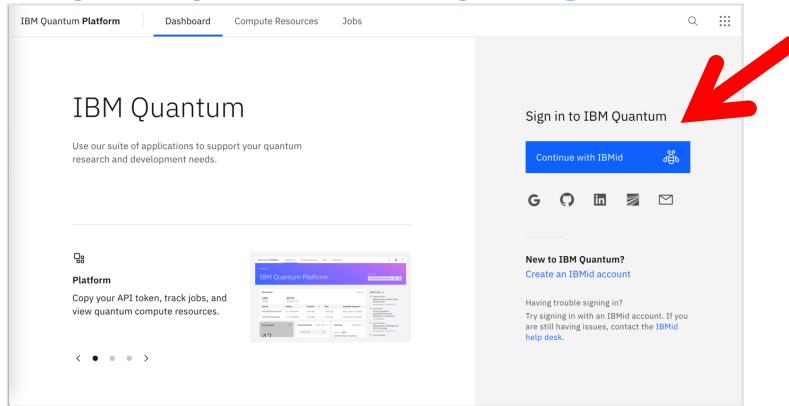
デモ & ハンズオン

ハンズオン: IBM Quantum Composer

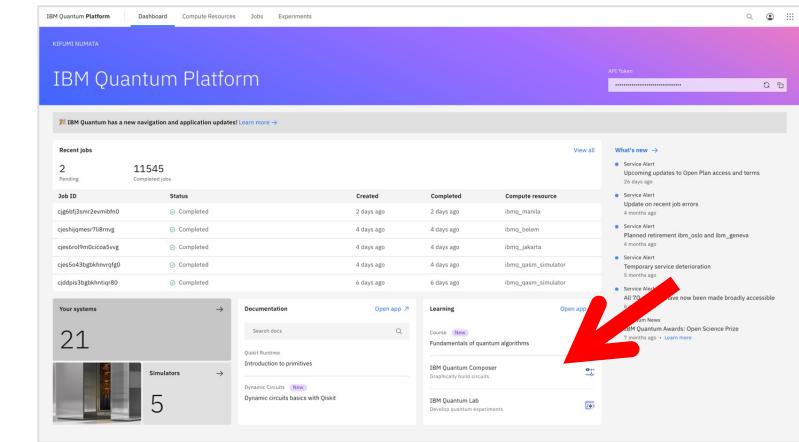
ハンズオンの資料 : <https://github.com/kifumi/teleportation-composer>

(1) IBM Quantum にログインします。

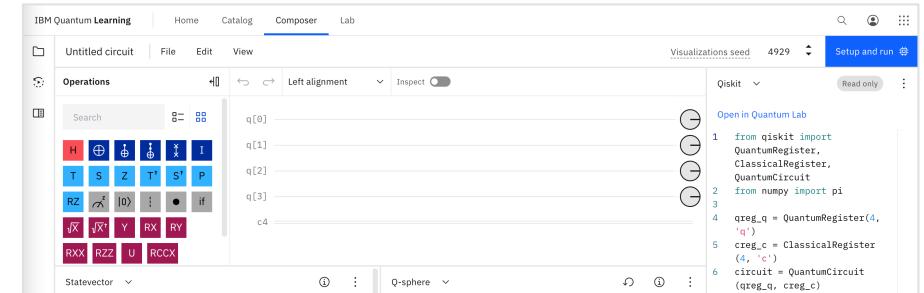
<https://quantum-computing.ibm.com/>



(2) 中央下方の「IBM Quantum Composer」をクリック。



(3) この画面になつたら準備完了です。



量子テレポーテーション

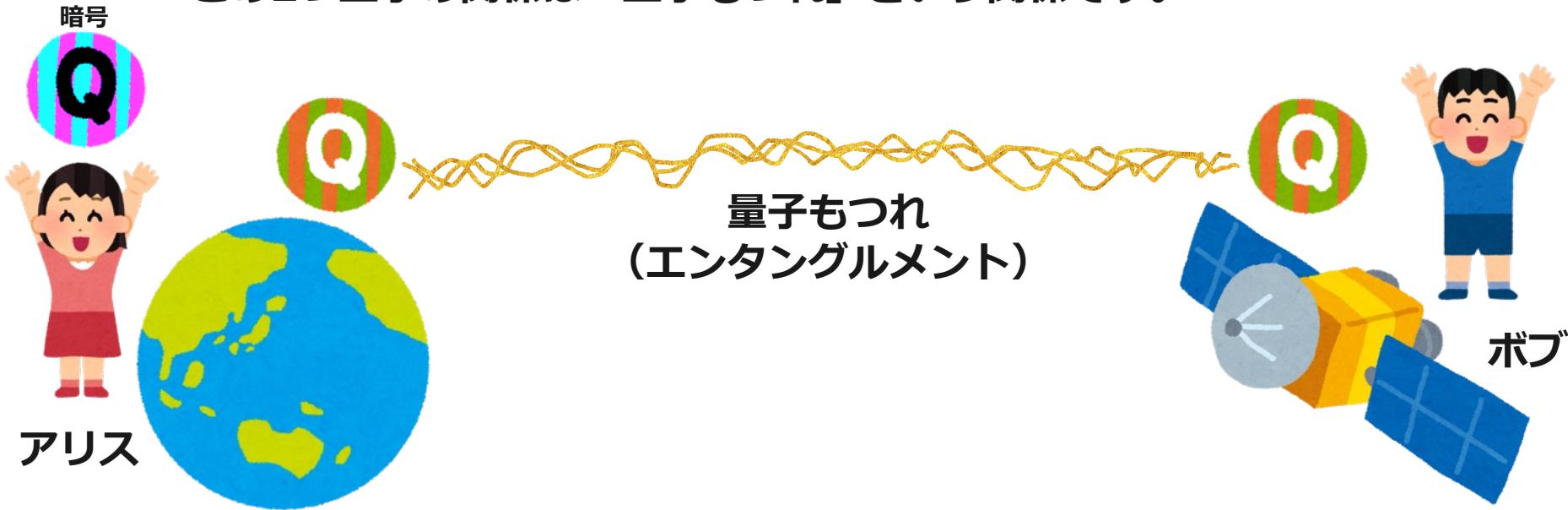
遠く離れた場所に量子の暗号を送る実験です。

エンタングルメント（量子もつれ）を使い、3量子ビットで実現できる
量子アルゴリズムです。



量子テレポーテーションのプロトコル

- (1) 地球のアリスがある量子  (暗号) を持っています。
- (2) 特別な関係にあるふたごの量子  が地球と人工衛星の上にあります。この2つ量子の関係は「量子もつれ」という関係です。



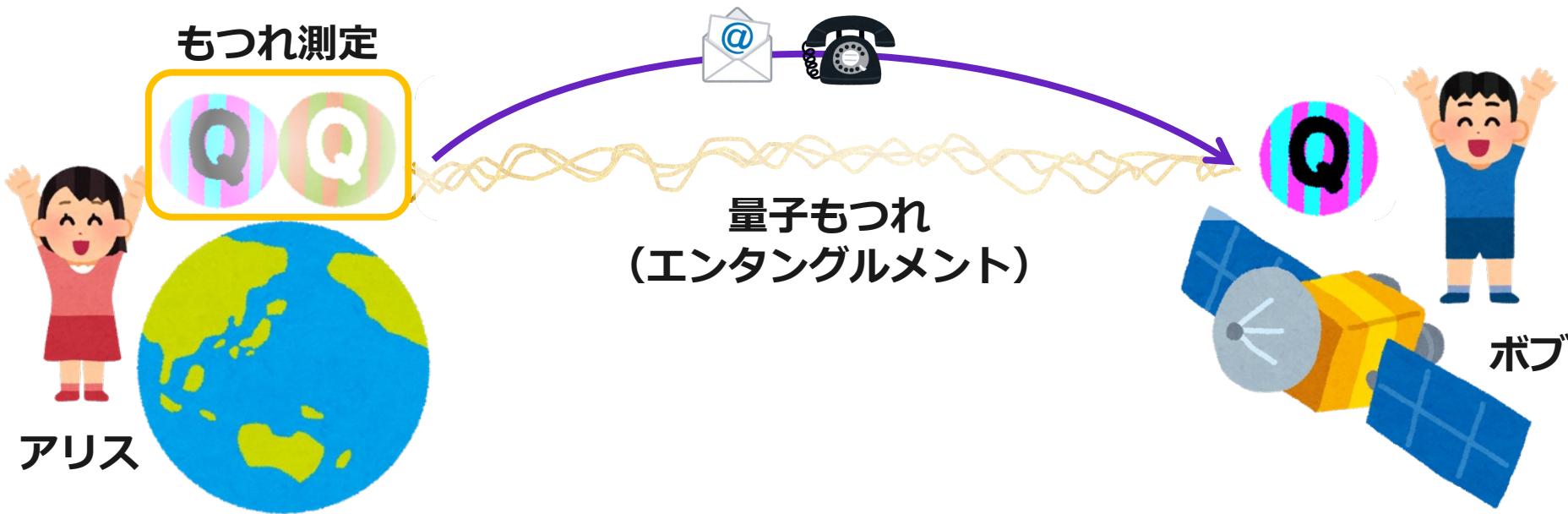
量子テレポーテーションのプロトコル

(3) 地球のアリスが地上の2つの量子に特殊な測定（もつれ測定）をします。
(量子もつれ状態にあるボブの量子の状態が瞬時に変わります。)

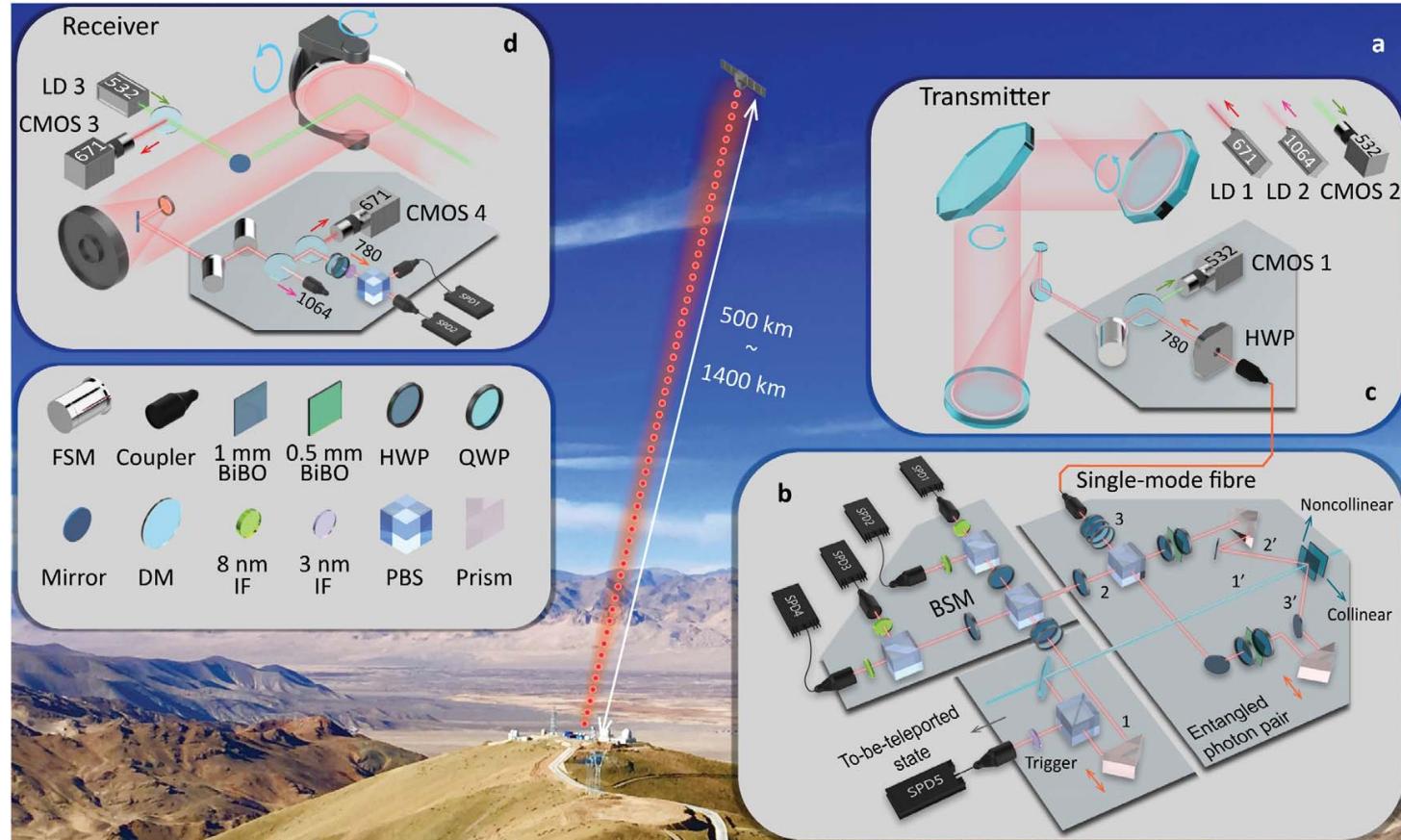


量子テレポーテーションのプロトコル

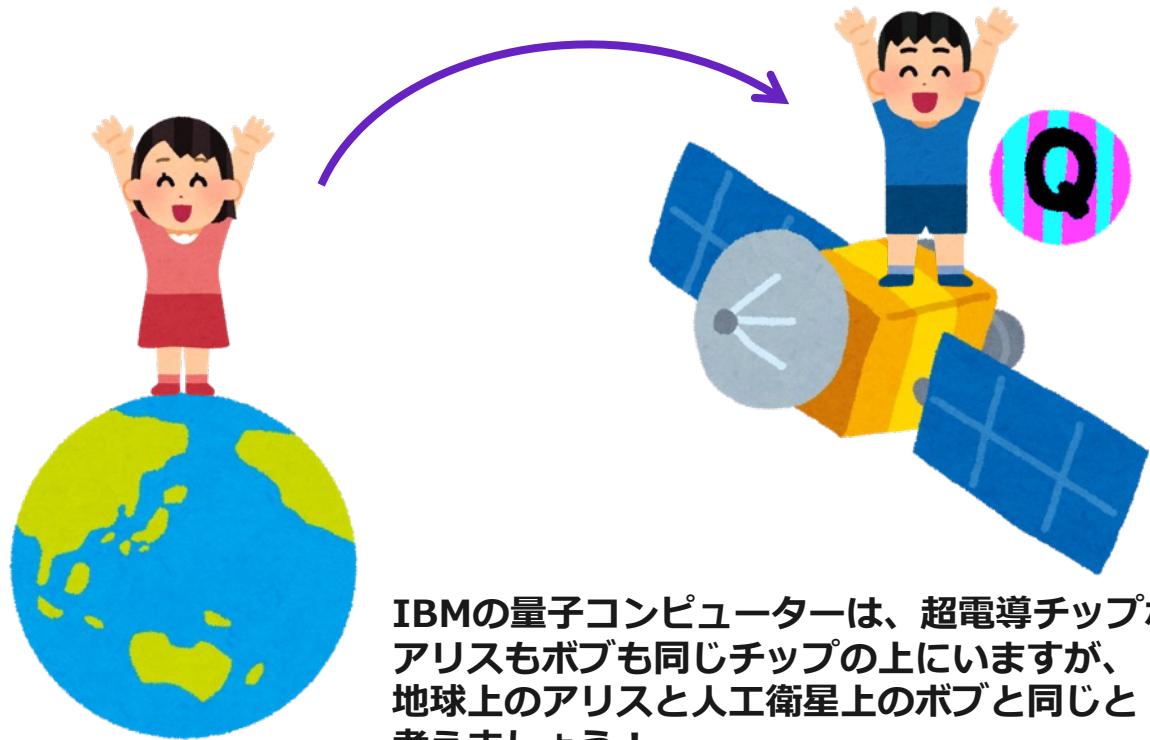
(4) アリスが測定結果をメールや電話でボブに送り、
ボブはもらった結果をもとに自分の量子を補正します。
ボブの量子がアリスの持っていた暗号に変化します！



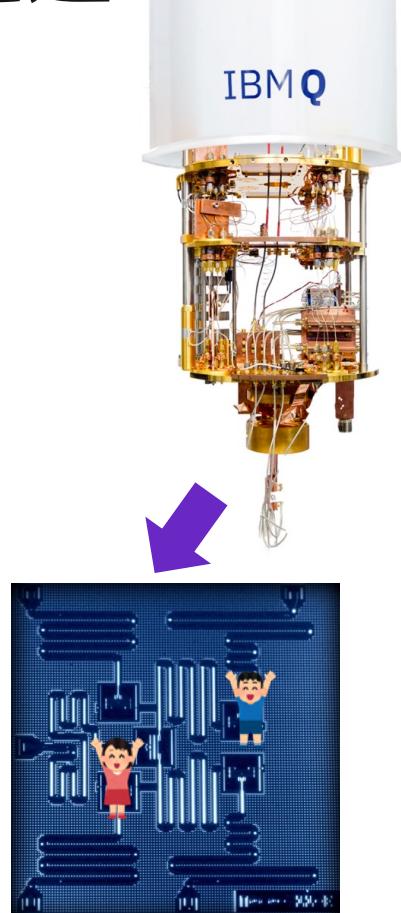
地上と通信衛星間の量子テレポーテーションの例



アリスからボブに暗号（量子状態）を送ります



IBMの量子コンピューターは、超電導チップなので
アリスもボブも同じチップの上にいますが、
地球上のアリスと人工衛星上のボブと同じと
考えましょう！

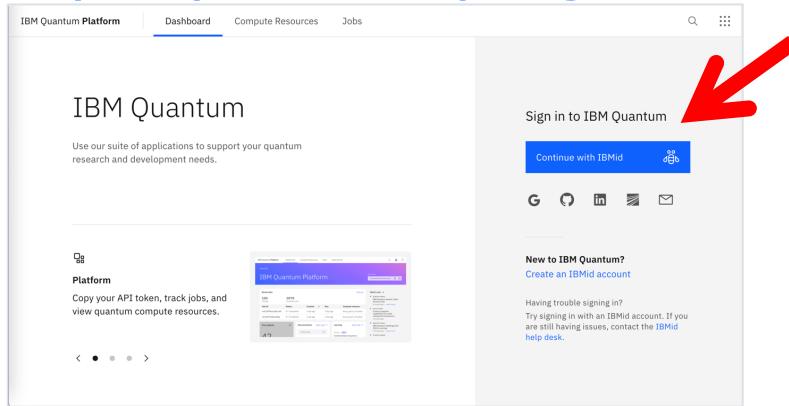


ハンズオン: IBM Quantum Composer

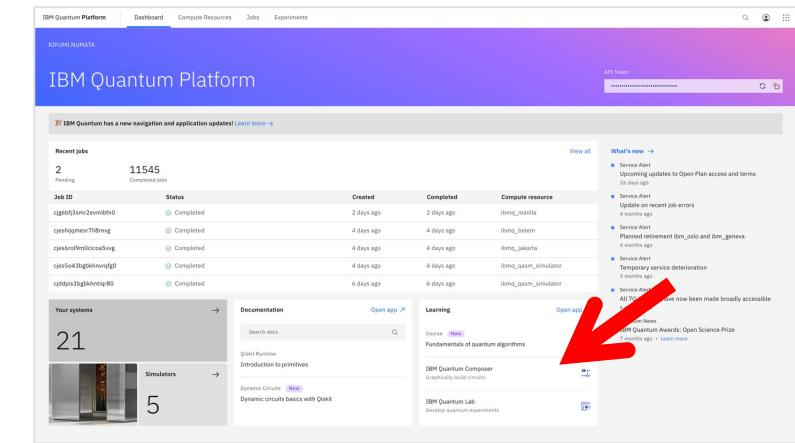
ハンズオンの資料 : <https://github.com/kifumi/teleportation-composer>

(1) IBM Quantum にログインします。

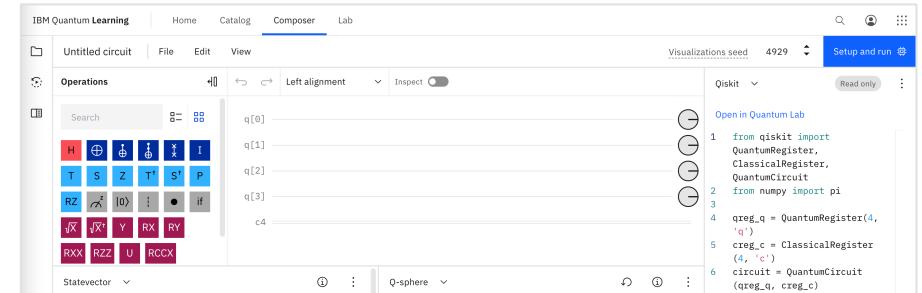
<https://quantum-computing.ibm.com/>



(2) 中央下方の「IBM Quantum Composer」をクリック。



(3) この画面になつたら準備完了です。



量子回路の準備(1)

The screenshot shows the Qiskit Quantum Circuit Editor interface. On the left, there's a toolbar with icons for operations like H, T, RZ, and various rotation gates. Below the toolbar is a search bar and a grid of quantum operation icons. To the right is a circuit canvas with four horizontal lines representing qubits: q[0], q[1], q[2], and q[3]. On the far right, there's a code editor window showing Python code for creating a quantum circuit with four qubits and four classical bits.

```
1  from qiskit import QuantumRegister, ClassicalRegister, QuantumCircuit
2  from numpy import pi
3
4  qreg_q = QuantumRegister(4, 'q')
5  creg_c = ClassicalRegister(4, 'c')
6  circuit = QuantumCircuit(qreg_q, creg_c)
7
8
```

マウスでq[3]をクリックするとゴミ箱マークが出てくるので、
クリックして消して、
q[0], q[1], q[2]の3量子ビット回路の準備をします。

量子回路の準備(2)

Untitled circuit *Saved* | File Edit View Visualizations seed 3623 ▾ Setup

Operations

Search

Operations palette:

- Row 1: H, \oplus , \odot , $\ddot{\oplus}$, $\ddot{\odot}$, I
- Row 2: T, S, Z, T^\dagger , S^\dagger , P
- Row 3: RZ, \sqrt{X} , $|0\rangle$, I, \bullet , if
- Row 4: \sqrt{X}^\dagger , Y, RX, RY, RXX
- Row 5: RZZ, U, RCCX, RC3X

Classical register c4 is highlighted with a blue box and a red arrow.

q[0] q[1] q[2] c4

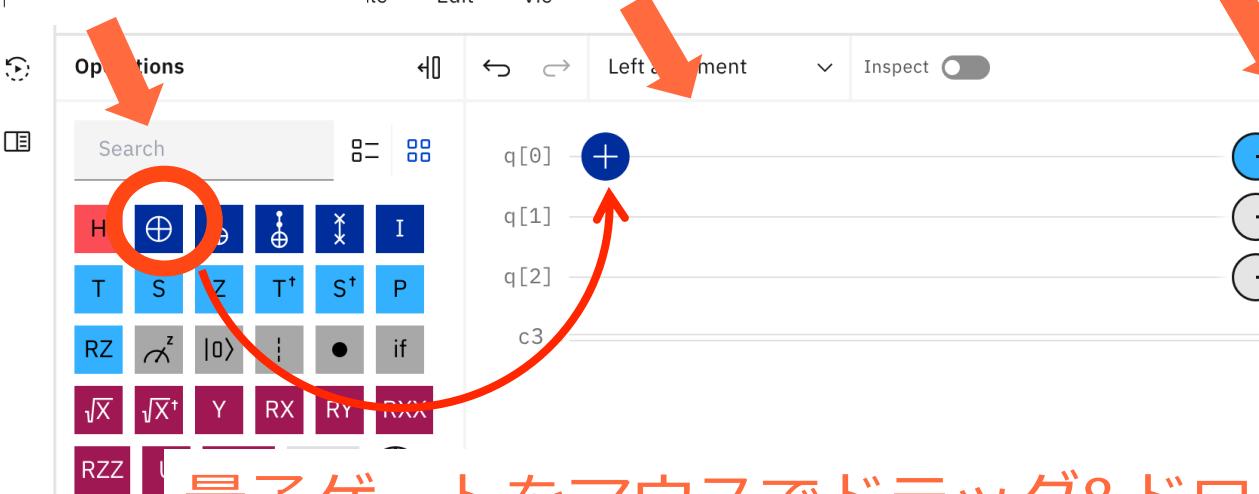
OpenQASM 2.0

```
1 OPENQASM 2.0;
2 include "qelib1.inc";
3
4 qreg q[3];
5 creg c[4];
```

マウスでc4をクリック
→「-」を選択してc3にします。

量子回路の作り方

量子ゲート 量子回路



量子プログラミング
Qiskitのコード

Qiskit

Open in Quantum Lab

```
1  from qiskit import QuantumRegister,  
2  ClassicalRegister, QuantumCircuit  
3  
4  qreg_q = QuantumRegister(3, 'q')  
5  creg_c = ClassicalRegister(4, 'c')  
6  circuit = QuantumCircuit(qreg_q, creg_c)  
7  
8  circuit.x(qreg_q[0])
```

量子ゲートをマウスでドラッグ&ドロップして、
量子回路を作ります。

(右側には、量子プログラミングQiskitのコードが自動生成されます。)

テレポーテーションの量子回路

アリス



{

$q[0]$

$q[1]$

ボブ



$q[2]$

テレポーテーションの量子回路



(1) 地球のアリスがある量子 **Q**(暗号) を持っています。
(今回は「1」を暗号とします。 **⊕** を使います。)

テレポーテーションの量子回路

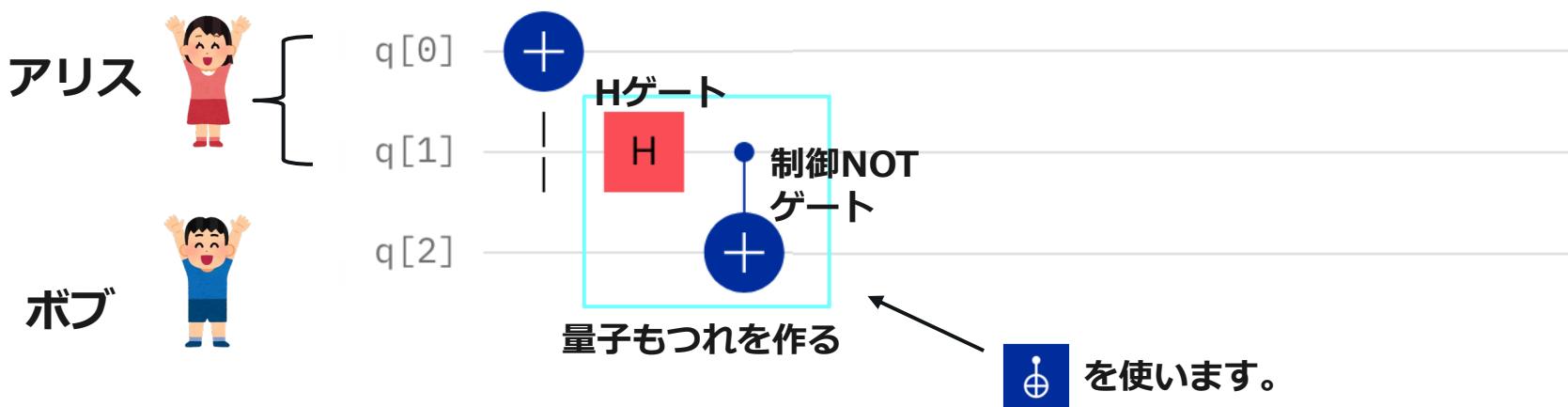


(1) 地球のアリスがある量子 **Q**(暗号) を持っています。
(今回は「1」を暗号とします。 **⊕** を使います。)

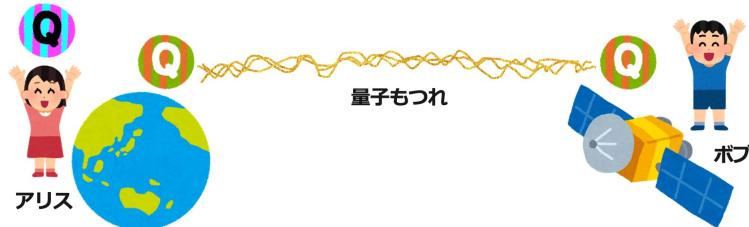


: 回路を見やすくするためのゲート **|** です。

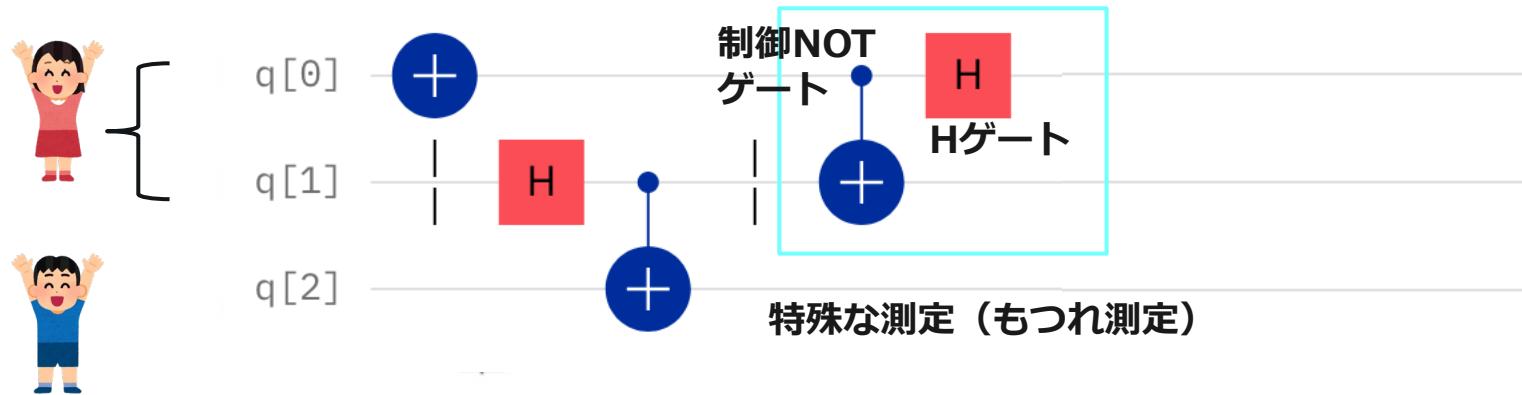
テレポーテーションの量子回路



(2) 特別な関係にあるふたごの量子 Q が地球と人工衛星の上にあります。この2つ量子の関係は「量子もつれ」という関係です。



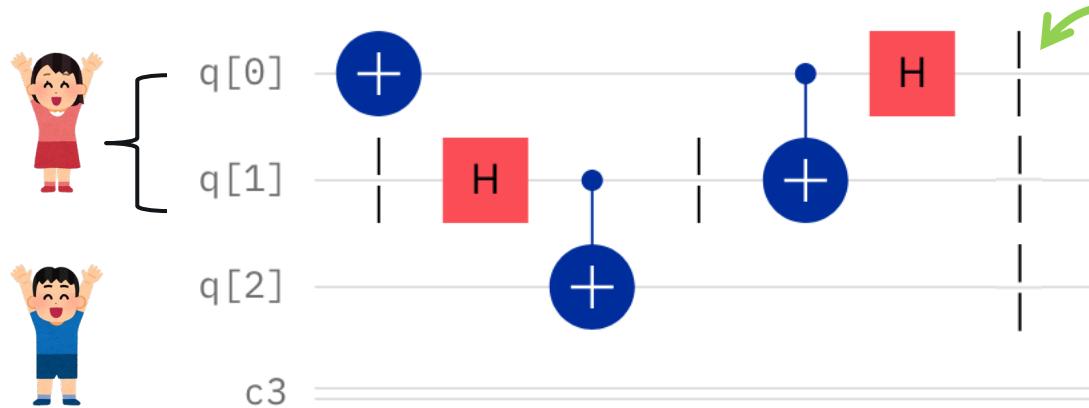
テレポーテーションの量子回路



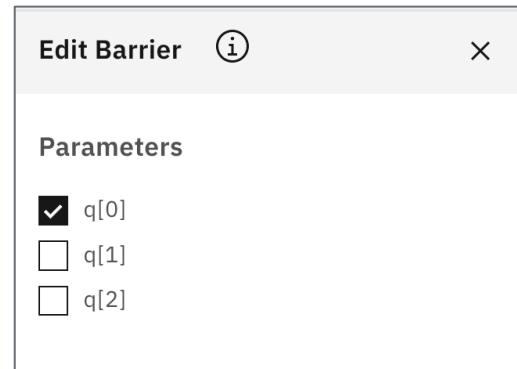
(3) 地球のアリスが地上の2つの量子に特殊な測定（もつれ測定）をします。
（量子もつれ状態にあるボブの量子の状態が瞬時に変わります。）
今回は、測定自体は最後に行います。



テレポーテーションの量子回路

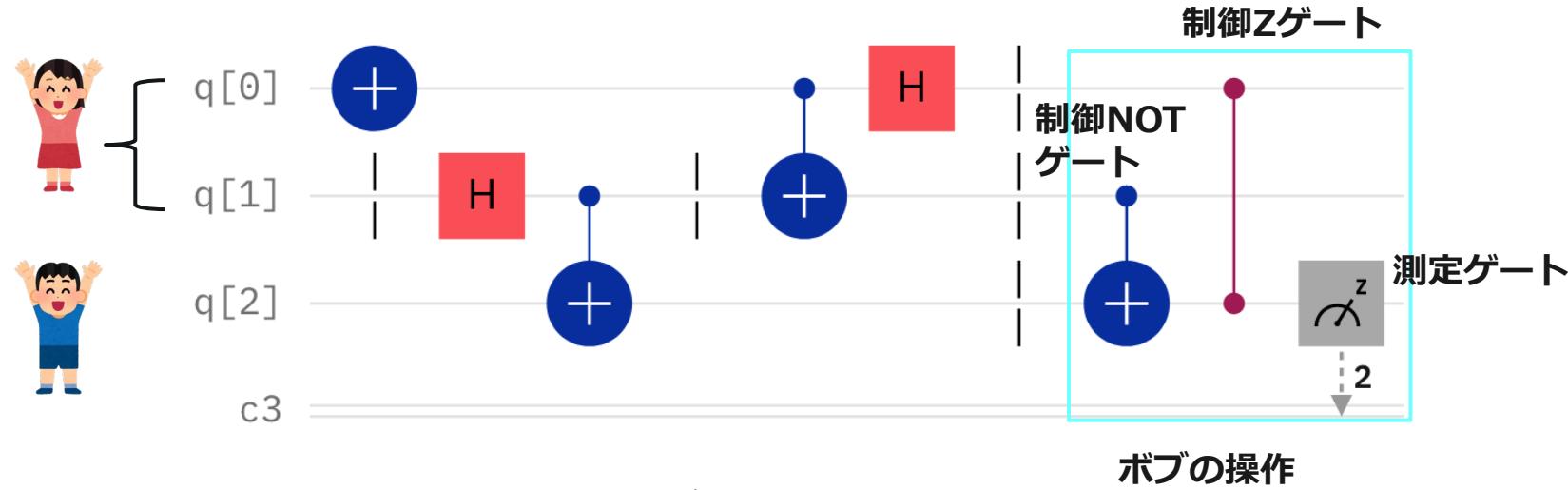


：回路を見やすくするためのゲートです。
1つおいたら、ダブルクリックします。

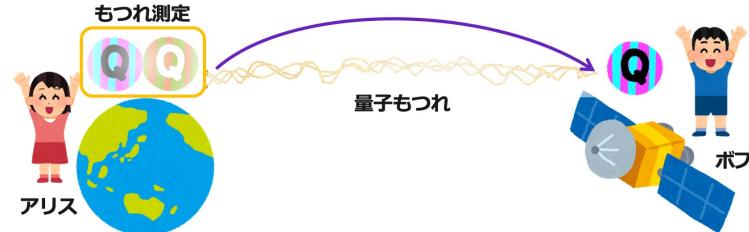


q[1]とq[2]も選択します。

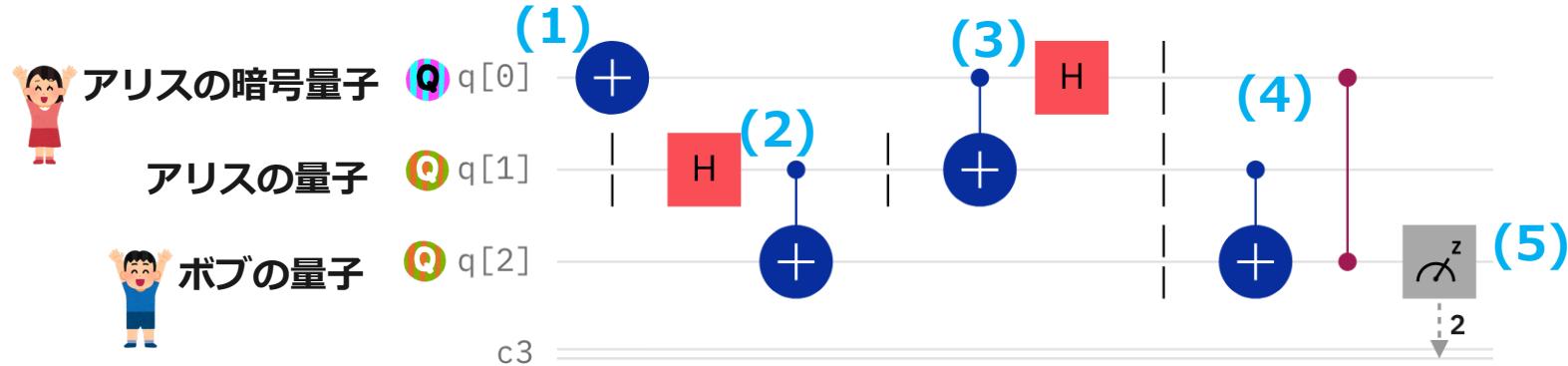
テレポーテーションの量子回路



(4) アリスが測定結果をボブに送り、
ボブはもらった結果をもとに自分の量子を補正します。
ボブの量子がアリスの持っていた暗号に変化します！



量子テレポーテーションの回路



\oplus : (1) アリスは暗号を作ります。
(今回は「1」を暗号とします。)

$+$: 回路を見やすくするためのゲート \square です。

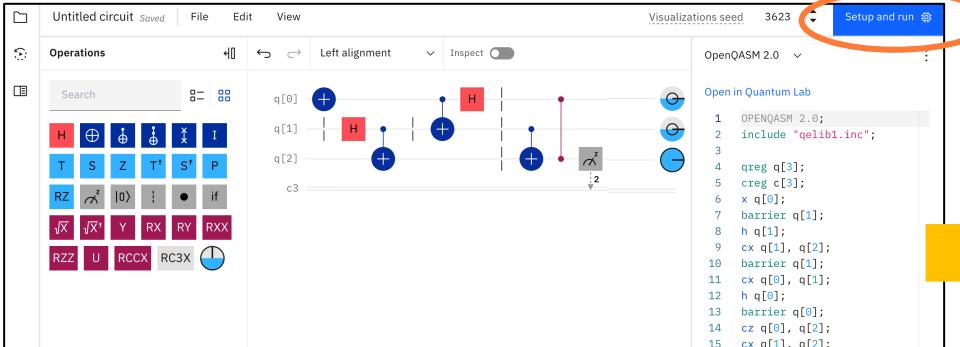
H \oplus : (2) 量子もつれを作ります。
 \oplus を使います。

H $+$: (3) アリスはもつれ測定をします。

$+$: (4) アリスの測定結果をボブに送り、
ボブは自分の量子を補正します。
の作り方：
 Z を $q[2]$ に置き、 \bullet を重ねる。
位置はあとから調整します。

Z $\downarrow 2$: (5) ボブの量子を測定します。

実験します：量子シミュレーター



(1)

「Setup and run」をクリック。

Set up and run your circuit

Step 1
Choose a system or simulator

Provider: ibm-q-internal/deployed/default

Step 2
Choose your settings

Total pending jobs: 1

63 Qubits

ibmq_qasm_simulator See details

Simulator status: Online

Total pending jobs: 1

32 Qubits

Job limit: None

Optional
Name your job

Job name

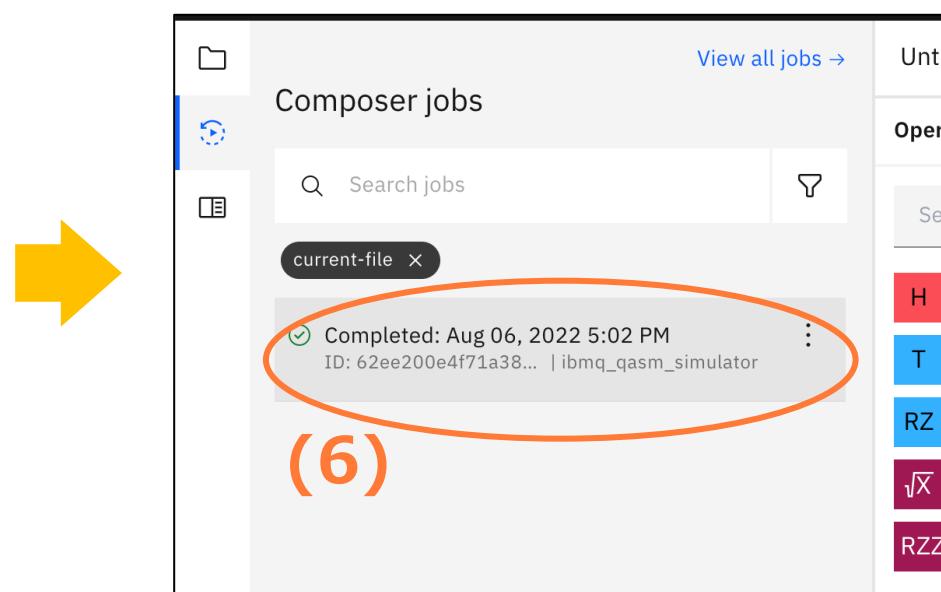
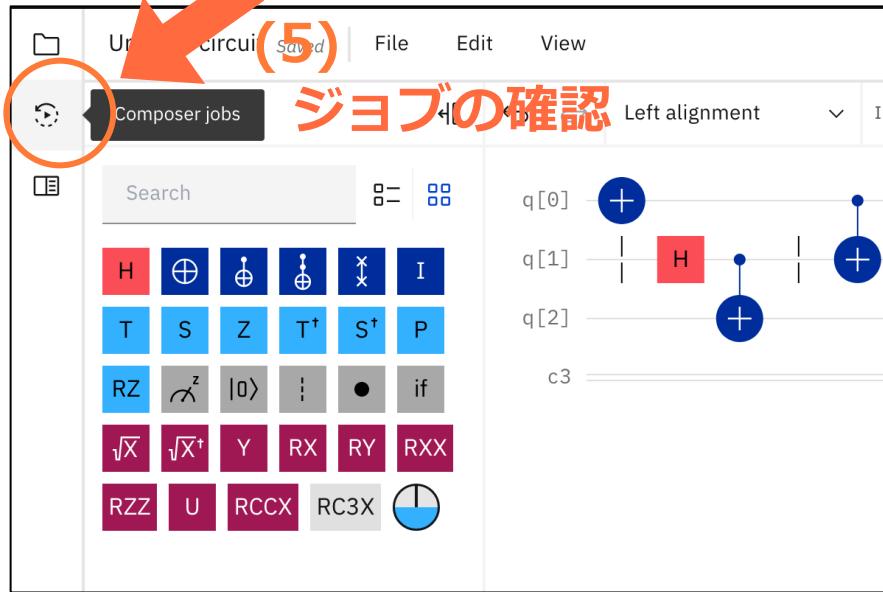
(2)
スクロール

(3)下から3番目の
ibmq_qasm_simulatorを選ぶ

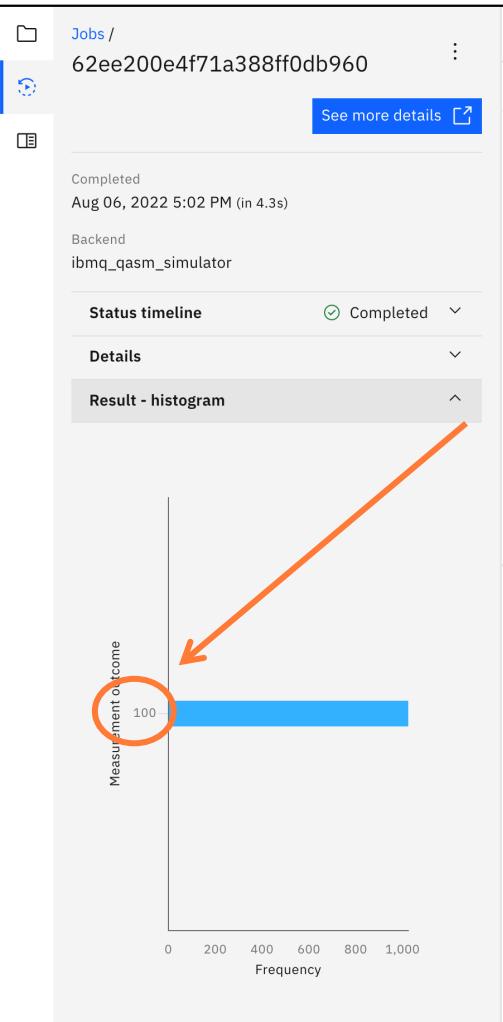
(4) Run on ibmq_qasm_simulator

- (1) 右上の青の「Setup and run」をクリック。
- (2) スクロール。
- (3) 「ibmq_qasm_simulator」を選ぶ。
- (4) 青の「Run on ibmq_qasm_simulator」をクリックして実行。

結果を見ます



結果：量子シミュレーターの場合

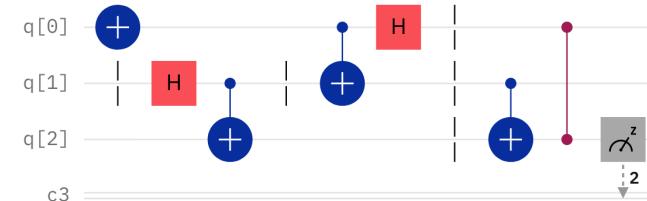


結果：

「100」のみになりましたか？

1024回の実験を行なった結果、すべて「100」だったということです。

「100」は、 $q[2] q[1] q[0]$ の順に並んでるので、測定した $q[2]$ の値は「1」だったという意味です。



アリスの暗号「1」がボブに100%テレポートされたことが分かります！

シミュレーションは、理想的な量子計算をするので、理論的に予想される結果が観測されました。

本物で実験

クラウド上の量子コンピューターに実験をリクエストします。

The image shows the Qiskit interface. On the left, a sidebar displays 'Jobs /' with a job ID '62ee200e4f71a388ff0db960' and a 'See more details' button. Below it are sections for 'Completed' (Aug 06, 2022 5:02 PM), 'Backend' (ibmq_qasm_simulator), 'Status timeline' (Completed), 'Details', and 'Result - histogram'. The main area shows an 'Untitled circuit' with three qubits (q[0], q[1], q[2]) and one classical register (c3). A circuit diagram is shown with various operations like H, T, RZ, RXX, CNOT, and MEASURE. To the right of the circuit is its corresponding OpenQASM 2.0 code:

```
OPENQASM 2.0;
include "qelib1.inc";
qreg q[3];
creg c[3];
x q[0];
barrier q[1];
h q[1];
cx q[1], q[2];
barrier q[1];
cx q[0], q[1];
h q[0];
barrier q[0];
cz q[0], q[2];
cx q[1], q[2];
measure q[2] -> c[2];
```

At the top right, there is a 'Setup and run' button with a dropdown menu. A large orange arrow points from this button to a callout labeled (1).

(1) もう一度「Setup and run」をクリック。

A second orange arrow points from the 'Setup and run' button to a 'Set up and run your circuit' dialog window. This window has two tabs: 'Step 1 Choose a system or simulator' and 'Step 2 Choose your settings'. In 'Step 1', a list of backends is shown, with 'ibm_nairobi' selected. It shows 'System status Online', 'Total pending jobs 50', and '7 Qubits 32 QV 2.6K CLOPS'. In 'Step 2', the provider is set to 'ibm-q/open/main', shots are set to 1024, and the job limit is 5 remaining. A red box highlights the 'ibm_nairobi' entry, and another red box highlights the 'Run on ibm_nairobi' button at the bottom right.

(2) スクロール

(3) 実デバイスを選択します。
7Qubitsか5Qubitsのものです。

(4) Run on ibm_nairobi

結果を待ちます：本物の量子コンピューター

The screenshot shows the IBM Quantum Experience interface. A red circle highlights the title "Job(ジョブ)の確認" (Job Confirmation) and the job ID "62d93e52914ea335bfa91772". Below it, the status is listed as "Completed" on "Jul 21, 2022 8:55 PM (in 1m 8s)". The "Backend" is "ibmq_qasm_simulator". A yellow arrow points down to the "Status timeline" section, which shows the job has completed.

Jobs / Job(ジョブ)の確認
62d93e52914ea335bfa91772

See more details

Completed
Jul 21, 2022 8:55 PM (in 1m 8s)

Backend
ibmq_qasm_simulator

Status timeline

Completed

Operations

H \oplus \ominus \oplus \ominus
T S Z T[†] S
RZ α^2 $|0\rangle$ I

The screenshot shows the IBM Quantum Experience interface. A red circle highlights the estimated completion time "Estimated time to completion in 4 minutes". The "Backend" is "ibm_nairobi". A yellow arrow points from the bottom of the previous screenshot to this one, indicating the progression. The "Status timeline" shows the job is currently "In queue" and "Running".

Jobs / 62ee25770be8be49515a49ab

See more details

Estimated time to completion
in 4 minutes

Backend
ibm_nairobi

Status timeline

Pending

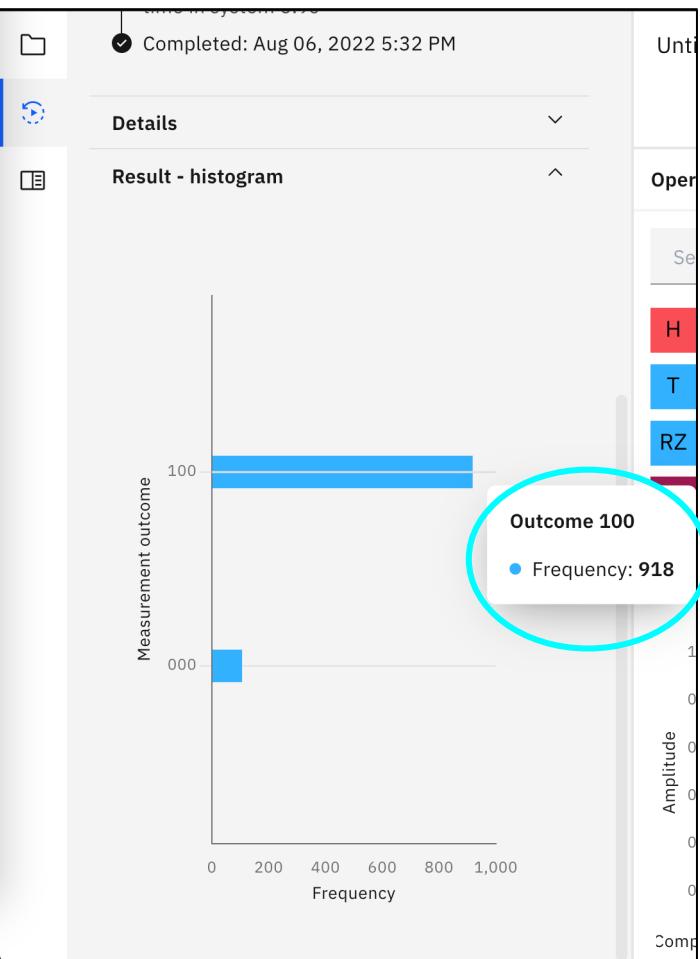
Created: Aug 06, 2022 5:25 PM
Transpiling: 907ms
Validating: 747ms
In queue
Running
expected to run in 4 minutes, project pos: 3, system pos: 3
Completed

世界中の人がクラウド経由でIBMの量子コンピューターを使っているため順番がくるまで時間がかかります。

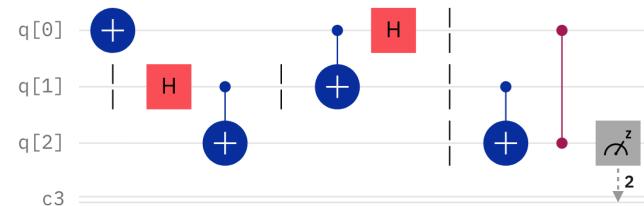
「COMPLETED(完了)」になったら、クリックして結果を見ます。

結果：本物の場合

(結果がまだの場合は後で確認してください)



参考結果：「100」が918回観測されました。 $q[2]$ $q[1]$ $q[0]$ の順に並んでいるので、 $q[2]$ の値は、1024回の実験を行なった結果、918回は「1」、106回は「0」が観測されたということです。



アリスの暗号「1」は約90%の確率でボブにテレポーテーションが成功したことが分かります。

量子コンピューターには、まだ誤差があります。エラー訂正の開発とともに、エラー緩和（ソフトウェアでエラーに対処）の研究開発も行われています。

詳しい説明はぜひQiskitテキストブックで！

<https://ja.learn.qiskit.org/course/ch-algorithms/quantum-teleportation>

The screenshot shows a web browser window displaying the Qiskit Textbook at <https://ja.learn.qiskit.org/course/ch-algorithms/quantum-teleportation>. The page title is "量子テレポーテーション". The left sidebar lists various quantum algorithms, with "量子テレポーテーション" selected. The main content area contains an introduction to quantum teleportation, mentioning the no-cloning theorem and the process of sending quantum information from Alice to Bob. A Python Scratchpad is available for testing code. The footer includes links to "Back to textbook home" and "Cookieの設定".

量子プロトコルと量子アルゴリズム

- 量子回路
- ディチ-ジョサのアルゴリズム
- ベルンシュタイン・ヴァジラニ アルゴリズム
- サイモンのアルゴリズム
- 量子フーリエ変換
- 量子位相推定
- ショアのアルゴリズム
- グローバーのアルゴリズム
- 量子数え上げ
- 量子テレポーテーション**

読み時間: ~25 min

量子テレポーテーション

このノートでは量子テレポーテーションについて説明します。まず始めに、作成した量子回路を Qiskit に組み込んでいるシミュレーターを使ってテストします。その後、その回路を実際の量子コンピューターで動かしてみます。

1. 概要

アリスがボブに量子情報を送りたいとしましょう。これを具体的に、アリスがボブに状態 $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ を送りたい、と仮定します。その為にはまず、 α と β の情報をボブに渡す必要があります。

量子力学には、未知の量子状態の単純な複製を正確に作ることは出来ない、という定理が存在します。量子複製不可能定理 (no-cloning theorem) として知られています。この定理ゆえ、アリスが単純に $|\psi\rangle$ の複製を生成してそれをボブに渡す、という事は出来ません。状態の複製は古典的な計算でのみ可能なのです（重ね合わせはコピーできません）。

しかし、2つの古典的ビットとエンタングルメントを利用することで、アリスは状態 $|\psi\rangle$ をボブに転送できます。その結果、最終的にボブが $|\psi\rangle$ を持つ、アリスがもはや何も持っていない事から、我々はこれをテレポーテーションと呼んでいます。

2. 量子テレポーテーション・プロトコル

Scratchpad

This is your Python Scratchpad. Use this however you like while you work or copy the code over from the chapter.

Note: Code in the scratchpad will not be saved.

Run

Feedback

Overview Learn Community Documentation

Japanese

Cookieの設定

まとめ

量子コンピューターは、量子重ね合わせ、エンタングルメントなどの量子現象を使った新しいコンピューター。

クラウド上で433量子ビットコンピューターが動き、日本では今年127量子ビットコンピューターが稼働予定。

10年後の10万量子ビット実現にむけ、大学、研究機関、企業が一丸となって取り組んでいる。

量子テレポーテーションといった基本的で重要なアルゴリズムを誰でも実験ができる環境がある。

自分のペースで学習できる 『Quantum Explorers 2023』

IBM Quantum

量子コンピューターの自習用プログラムで、入門的な内容から応用まで幅広いトピックを扱っています。

それぞれのトピックのモジュールの最後のクイズを完了することでバッヂがもらえるゲーム要素もあります。

期間：2023年7月から2024年2月



<https://ibm.biz/qexp23ja>

