Wprowadzenie do komputera kwantowego – tutorial

Fizyka dla informatyków. Wykład QI - część I .

Literatura do wykładu QI:

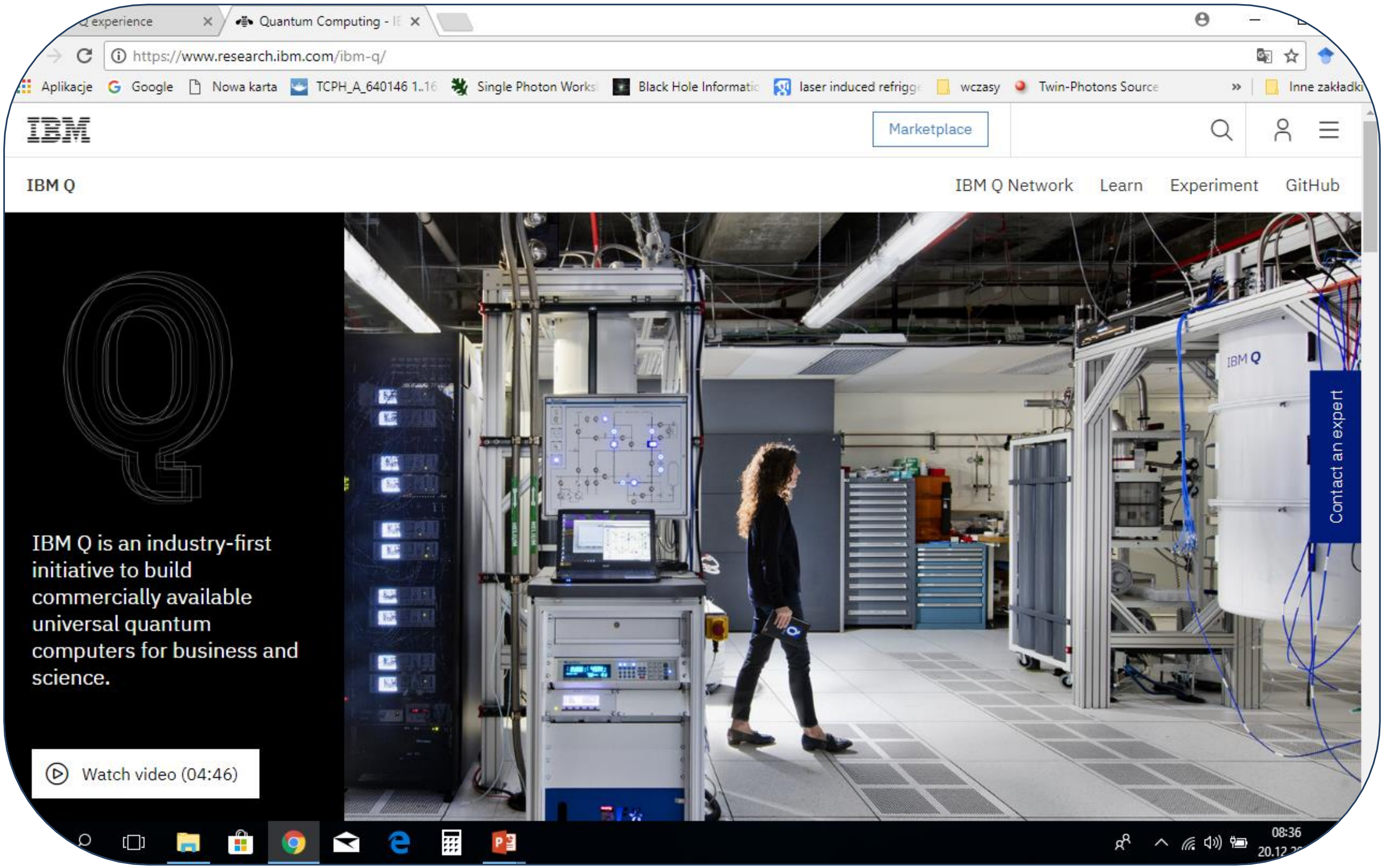
https://qiskit.org/textbook/preface.html

dr Gustaw Szawioła, doc. dydak.

Politechnika Poznańska
Wydział Inżynierii Materiałowej i Fizyki Technicznej
Instytut Badań Materiałowych I Inżynierii Kwantowej

0. HARDWARE KWANTOWY I BIBLIOTEKA QISKIT W NOTATNIKU JUPYTER – – PODSTAWOWE INFORMACJE

Możliwości kwantowego eksperymentowania IBM Q



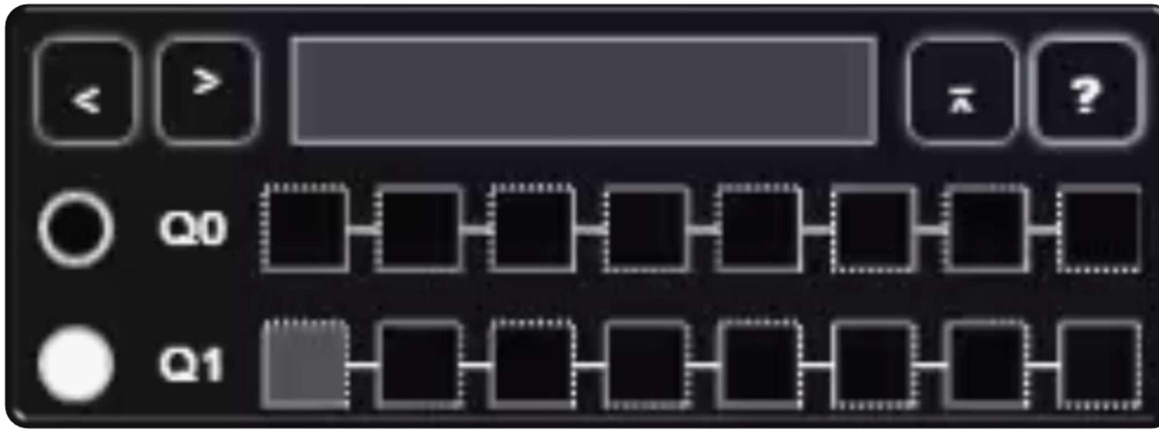
Nadprzewodnictwo – zjawisko Meissnera Lewitacja magnesu nad nadprzewodnikiem

https://www.youtube.com/watch?v=L01q4IghrmI&list=RDCMUC3j3w-oUtIAm_KI857ydvUA&index=1

Kwantowy kalkulator …

https://qiskit.org/modelq/

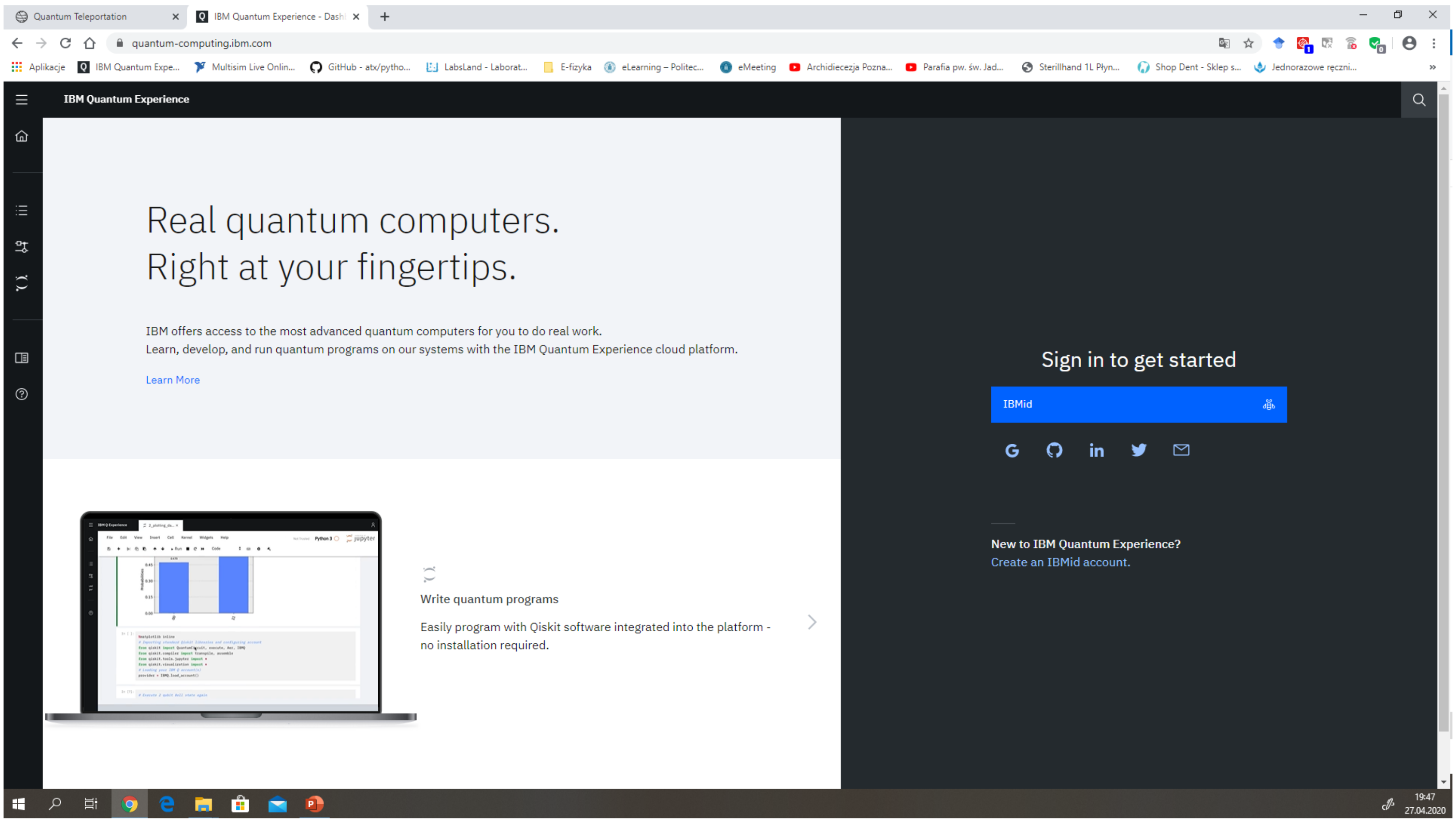
….i jego użycie



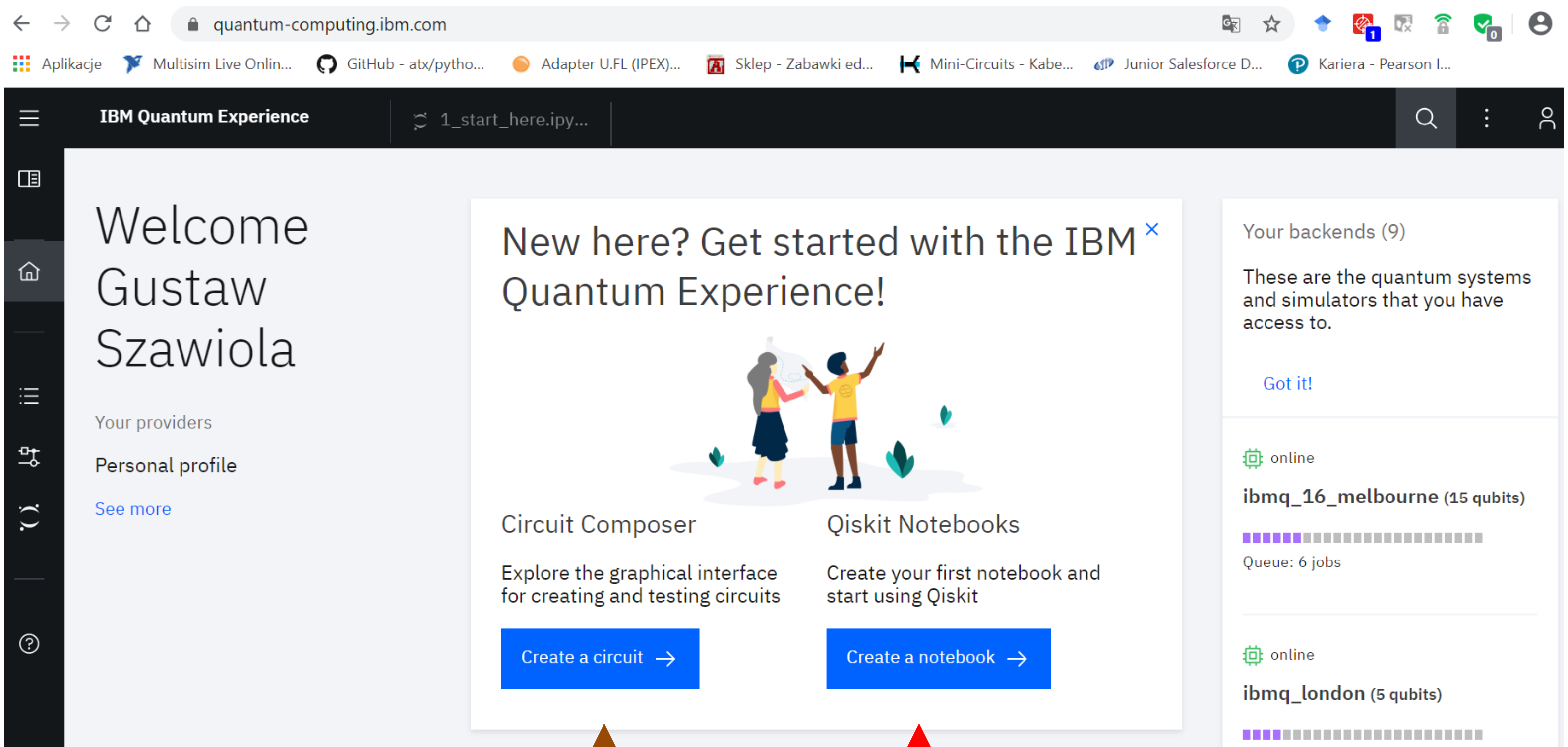
https://qiskit.org/modelq/

Logowanie na platformie IBM

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



Uruchamianie środowiska *Jupyter*



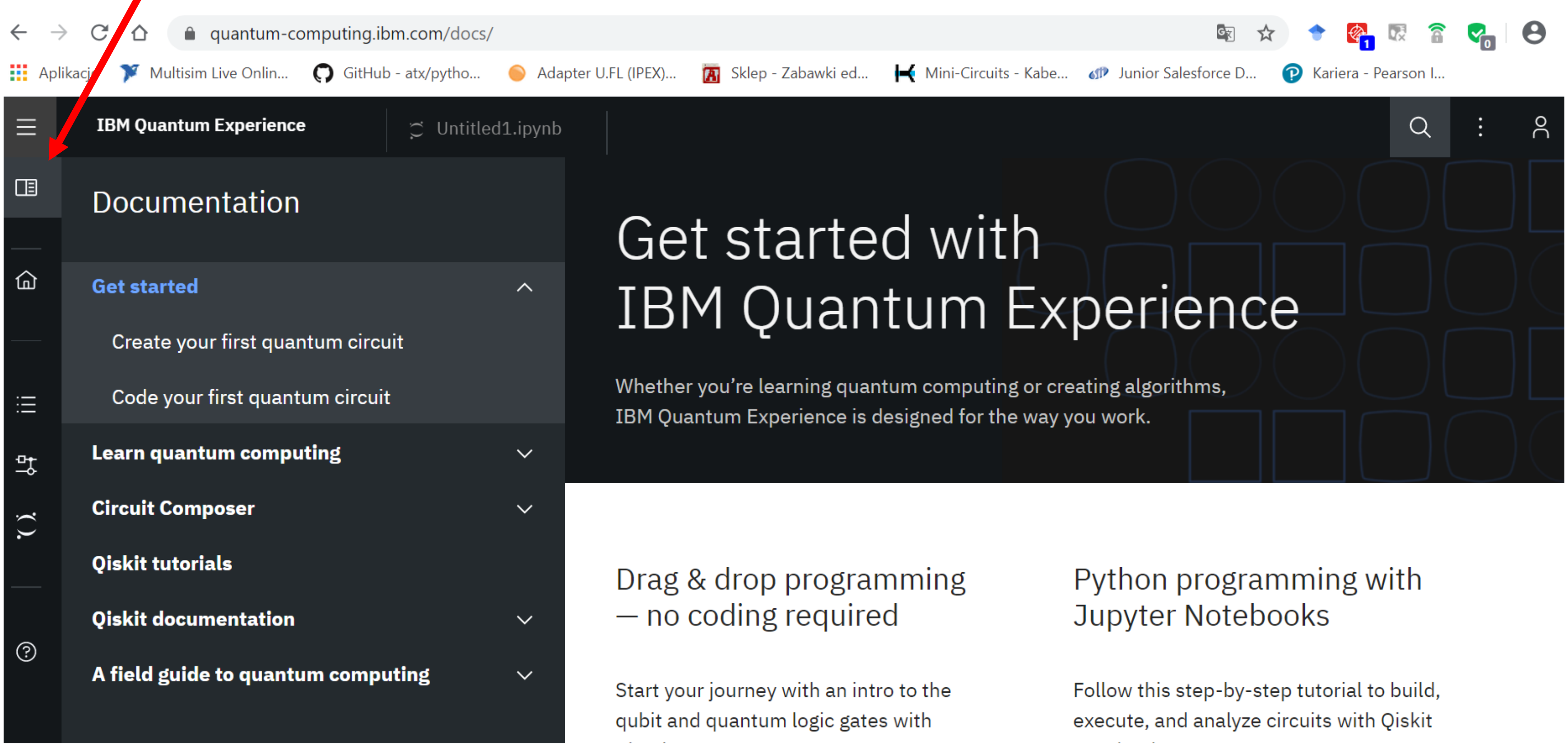
Przycisk pozwalający na przejście do trybu pracy z programowaniem w trybie graficznym

Przycisk pozwalający na przejście do trybu pracy z programowaniem w języku Python

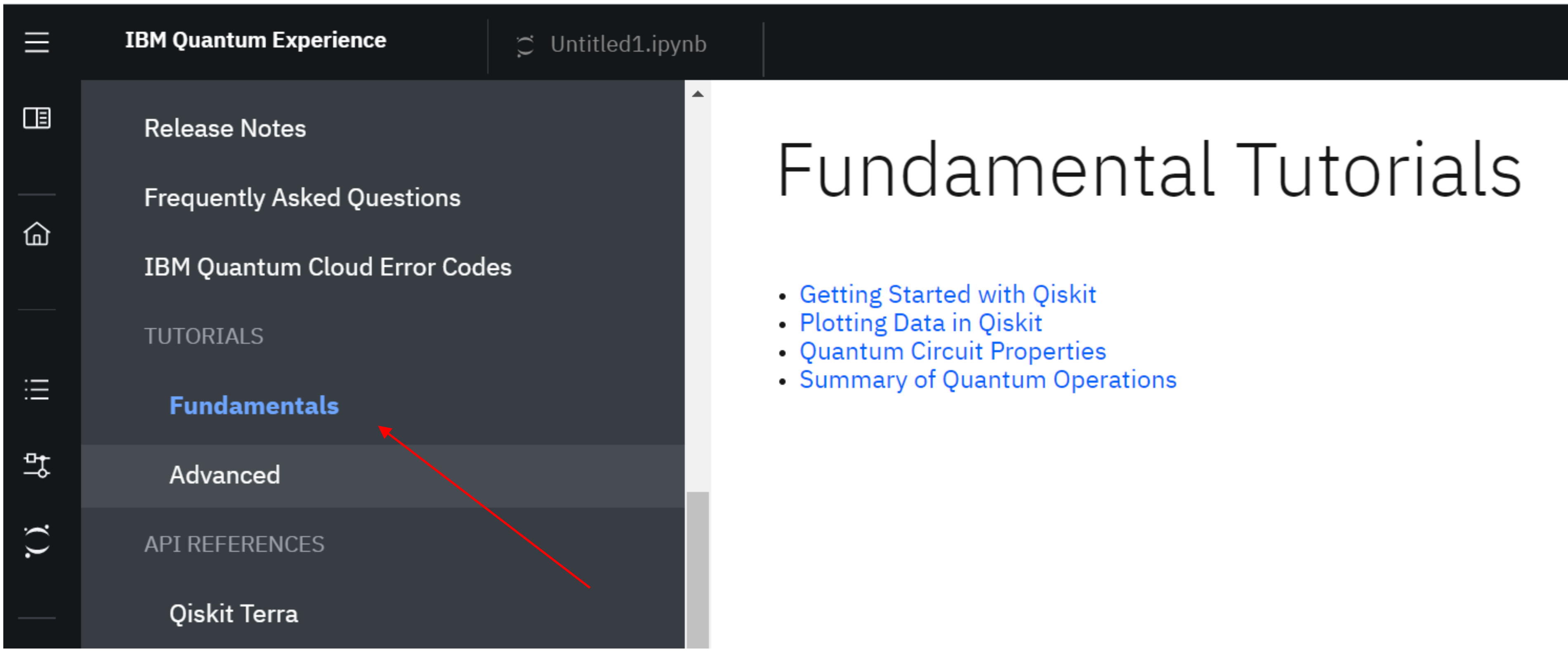
Nawigacja do przewodników

Przycisk dokumentacji i tutoriali

https://quantum-computing.ibm.com/docs/

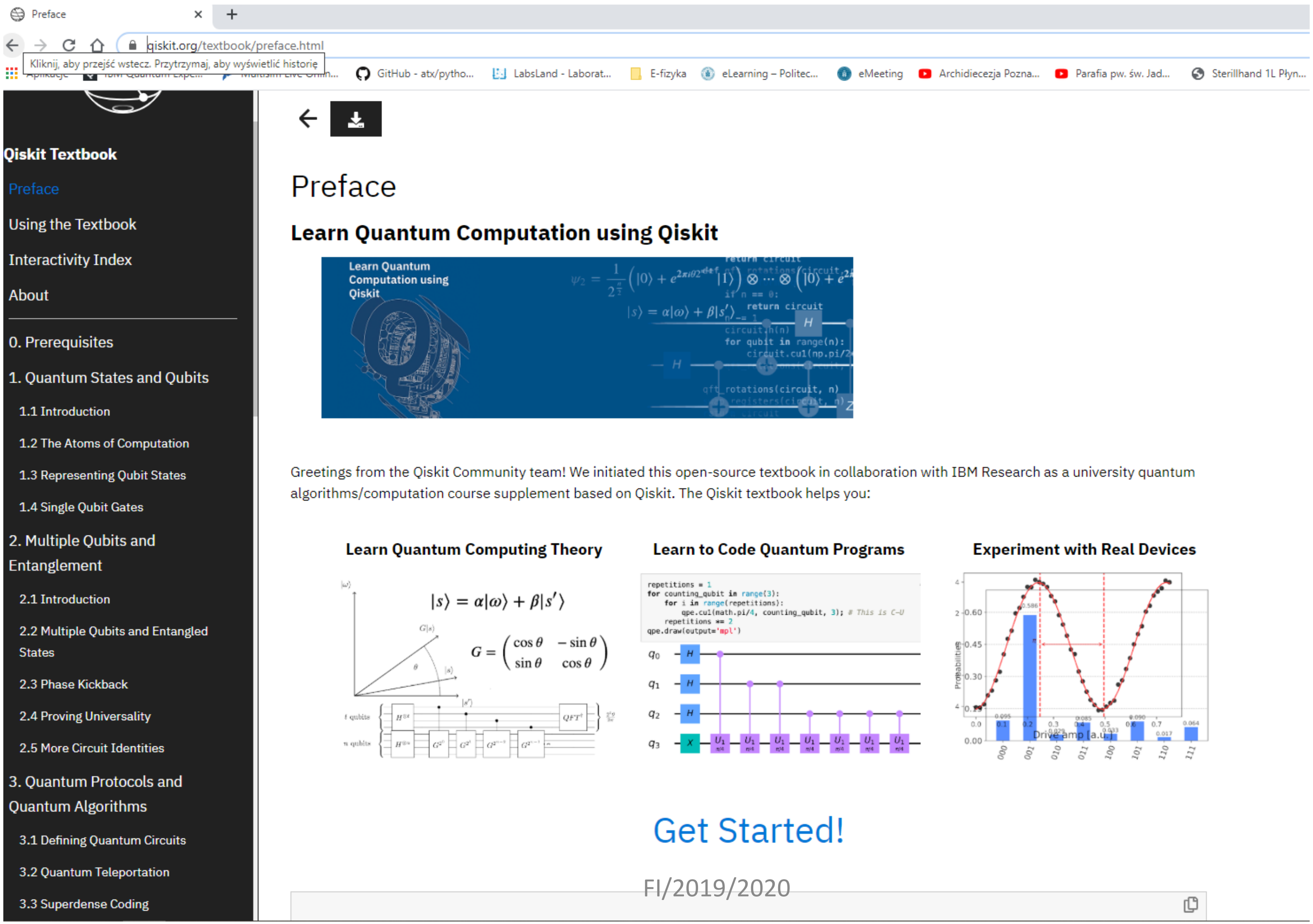


Wprowadzający przewodnik



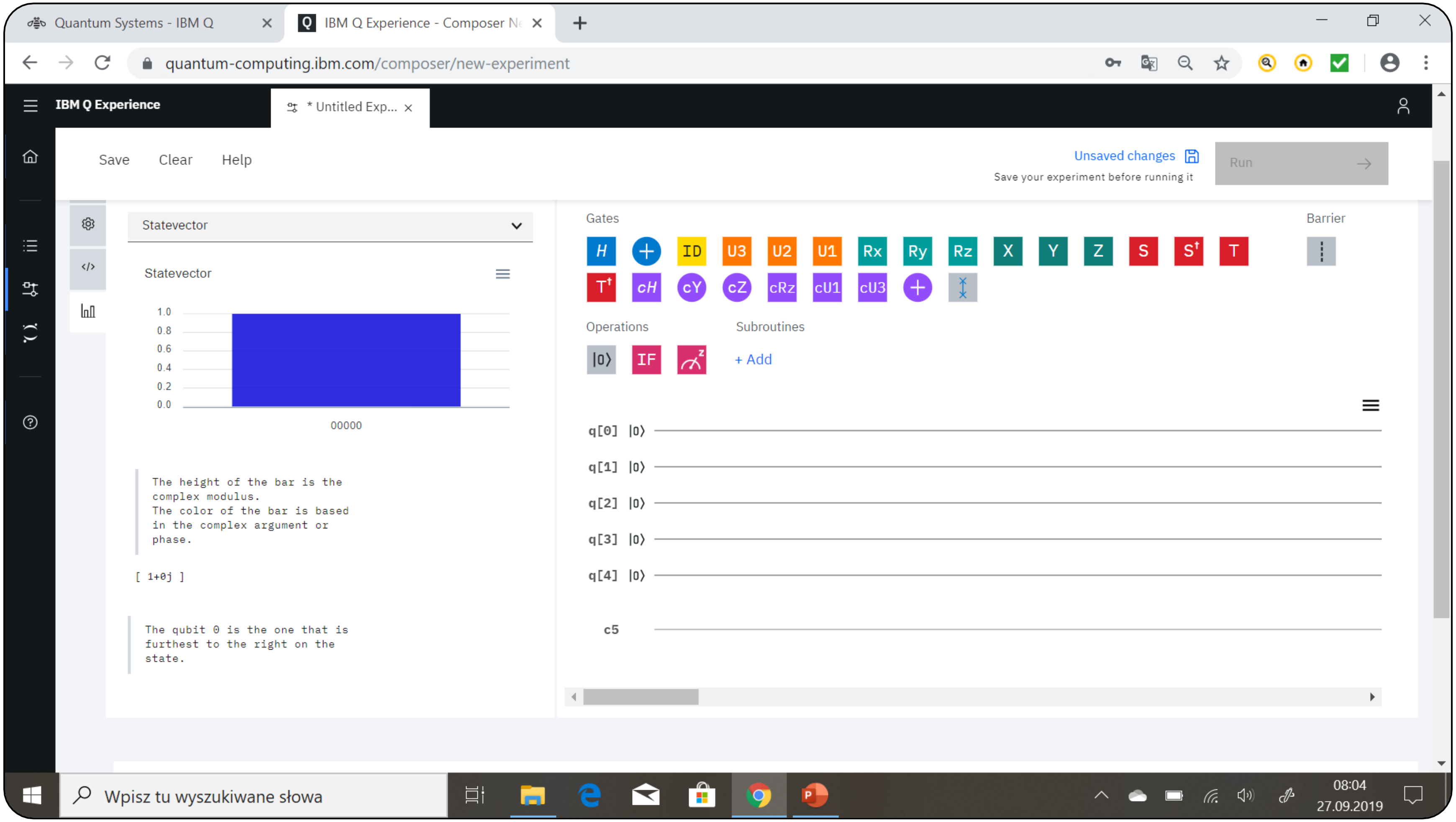
Podręcznik

https://qiskit.org/textbook/preface.html



Graficzne środowisko programistyczne komputera kwantowego – QUANTUM COMPOSER

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



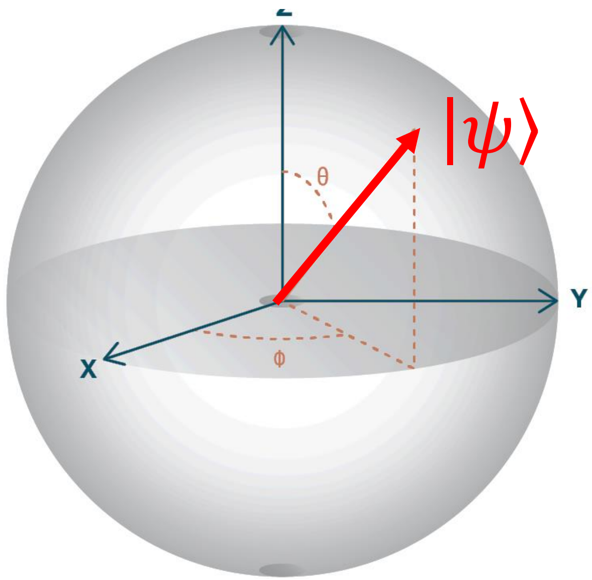
Sugestia

- Utwórz osobiste konto na platformie IBM
- Znajdź przestawione na slajdach przewodniki

1. QUBIT

Stan kubitu – superpozycja stanów bazowych

$|0\rangle$



$|1\rangle$

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$\color{red}{|\psi\rangle} = c_0|0\rangle + c_1|1\rangle$$

Algebraiczna notacja

KET

Baza przestrzeni stanów

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$|\psi\rangle = c_0\,|0\rangle + c_1\,|1\rangle = \begin{pmatrix} c_0 \\ c_1 \end{pmatrix}$$

$$+ ;\ |\psi\rangle \rightarrow |\psi\rangle^{+} = \langle\psi|$$

BRA

Baza dualnej (sprzężonej) przestrzeni stanów

$$\langle 0| = (1\ ,\ 0) \qquad \langle 1| = (0\ ,\ 1)$$

$$\langle\psi| = c_0^*\,\langle 0| + c_0^*\,\langle 1|$$
$$= \left(c_0^*\ ,\ c_1^*\right)$$

$a = |z|\cos\varphi \qquad b = |z|\sin\varphi$

Iloczyn skalarny (wewnętrzny)

$|\psi\rangle = a_0\;|0\rangle + b_1\;|1\rangle = \binom{a_0}{a_1}$

$\langle\varphi| = b_0^*\langle 0| + b_0^*\langle 1| = (b_0^* \;\;,\;\; b_1^*)$

$z = a + ib$

$z^* = a - ib$

$z = |z|e^{i\varphi} = |z|(\cos\varphi + i\sin\varphi)$

$$\boxed{\langle\varphi|\psi\rangle = (b_0^* \;\;,\;\; b_1^*)\binom{a_0}{a_1} = b_0^* a_0 + b_1^* a_1}$$

Iloczyn sklarny

$z^* z = |z|^2$

$\langle 0|0\rangle = 1 \qquad \langle 1|1\rangle = 1$

$\langle 0|1\rangle = 0$

$|\langle 0|\psi\rangle|^2 = |c_0|^2 = p$

$|\langle 1|\psi\rangle|^2 = |c_1|^2 = 1 - p$

$\langle 0|\psi\rangle = c_0$

$\langle 1|\psi\rangle = c_1$

Normalizacja wektora stanu

$c_0 = e^{i\varphi_0}|c_0|$ $\qquad$ $c_1 = e^{i\varphi_1}|c_1|$ $\qquad$ $|\psi\rangle = c_1|0\rangle + c_2|1\rangle$

$$\left|\psi\right\rangle = e^{i\varphi_0}\left(\left|c_0\right|\left|0\right\rangle + \left|c_1\right| e^{i(\varphi_1-\varphi_0)}\left|1\right\rangle\right)$$

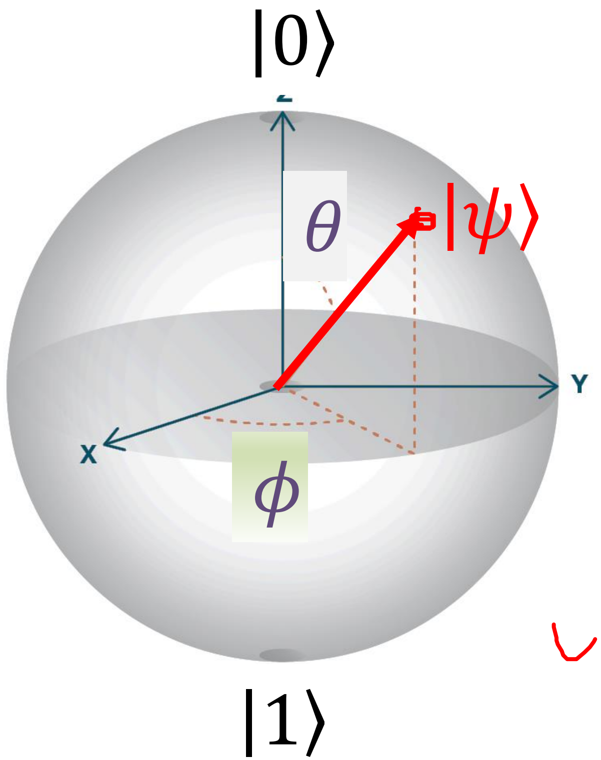
$$\left|c_0\right|^2 + \left|c_1\right|^2 = 1$$

$$\left|\psi\right\rangle = e^{i\varphi_0}\left(\cos\alpha\left|0\right\rangle + e^{i(\varphi_0-\varphi_1)}\sin\alpha\left|1\right\rangle\right)$$

$$\cos^2\alpha + \sin^2\alpha = 1$$

Stan kubitu a sfera Blocha

$$\| |\psi\rangle \|^2 = \langle\psi|\psi\rangle$$



$$|0\rangle = \begin{pmatrix}1\\0\end{pmatrix} \qquad |1\rangle = \begin{pmatrix}0\\1\end{pmatrix}$$

$$\langle\psi|\psi\rangle = \cos^2\frac{\theta}{2} + \sin^2\frac{\theta}{2} = 1$$

$$|\psi\rangle = \cos\left(\frac{\theta}{2}\right)|0\rangle + e^{i\phi}\sin\left(\frac{\theta}{2}\right)|1\rangle$$

$$p = p_{|0\rangle} = \cos^2\left(\frac{\theta}{2}\right)$$

$$p_{|1\rangle} = \sin^2\left(\frac{\theta}{2}\right) = 1 - p$$

Pomiar rzutowy typu Z – odczyt stanu kubitu

$$|\psi\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right)$$

$$p_0 = |c_0|^2 = |\langle 0|\psi\rangle|^2$$

$$p_1 = |c_1|^2 = |\langle 1|\psi\rangle|^2$$

Kod graficzny

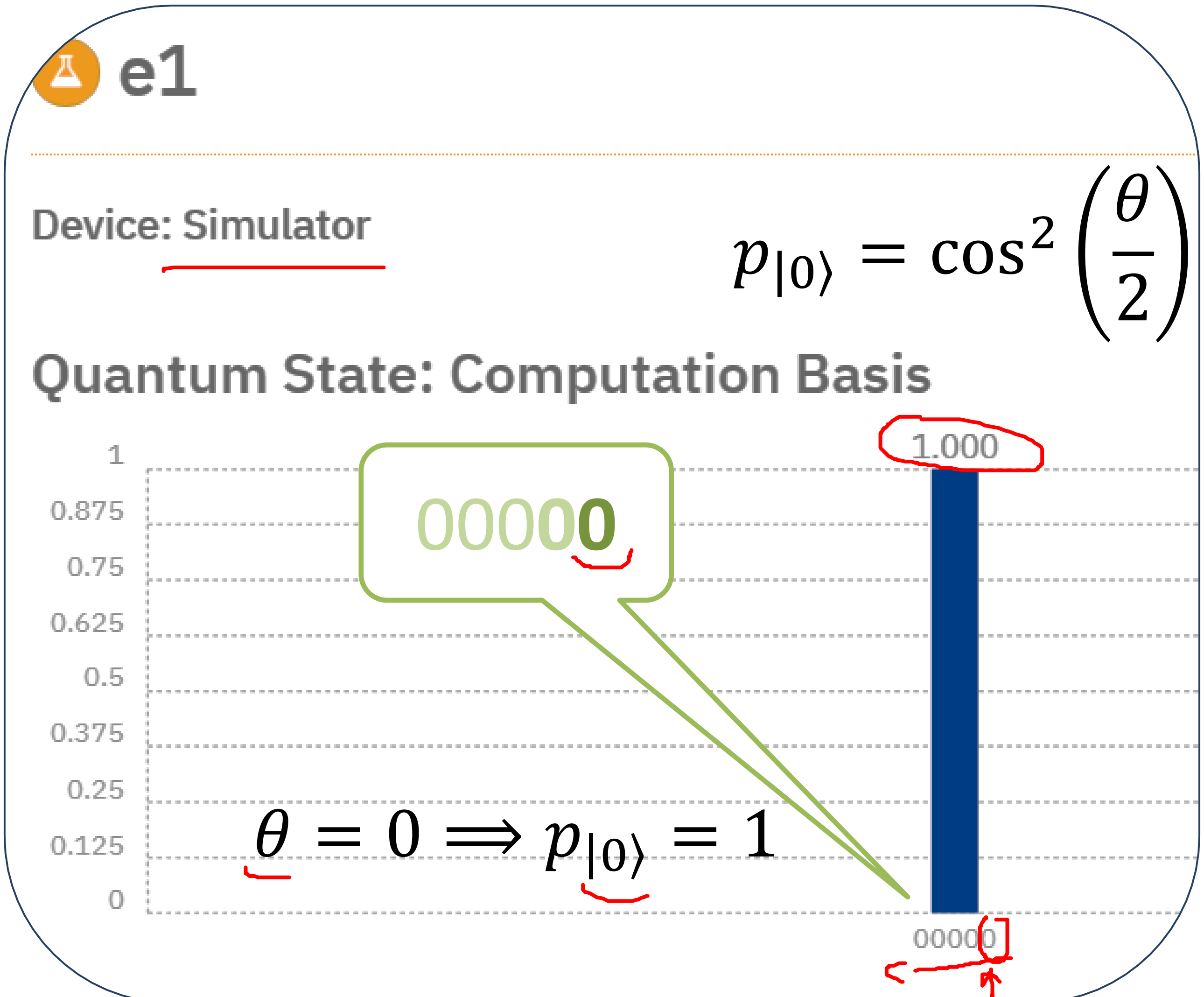
```
include "qelib1.inc";

qreg q[5];
creg c[5];

measure q[0] -> c[0];
```

Kod w asemblerze kwantowym (QASM)

$$p_0 = \cos^2\frac{\theta}{2}$$

$$p_1 = \sin^2\frac{\theta}{2}$$

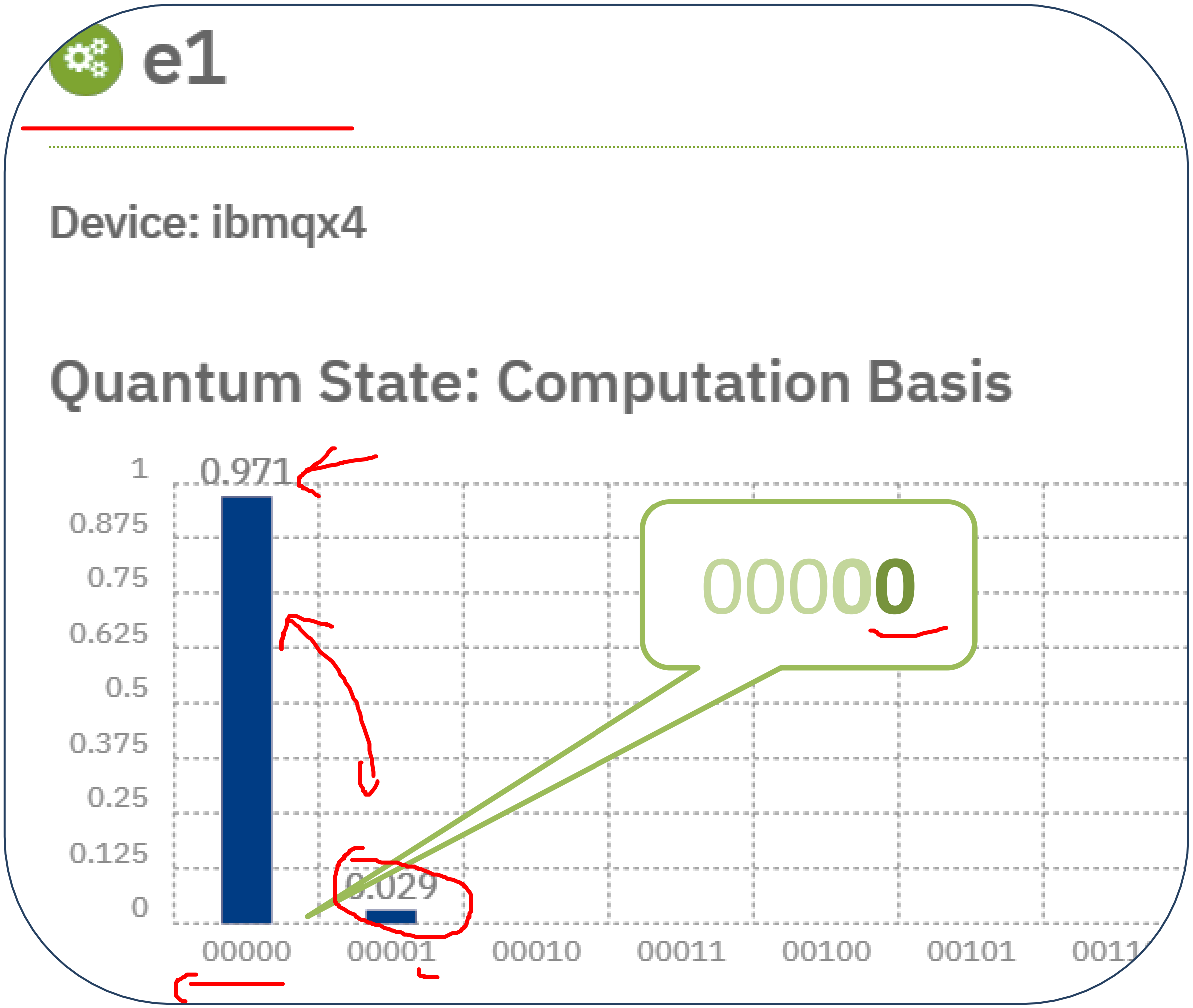
Klasyczna symulacja pomiaru



$$p_{|0\rangle} = \cos^2\left(\frac{\theta}{2}\right)$$

$$\theta = 0 \Longrightarrow p_{|0\rangle} = 1$$

Wynik ściśle kwantowych obliczeń na kwantowym komputerze



Python i biblioteka QISKIT

```
[1]: %matplotlib inline
     # Importowanie standardowej biblioteki Qiskit i konfiguracja konta
     from qiskit import QuantumCircuit,ClassicalRegister, QuantumRegister, execute,␣
      ↪Aer, IBMQ
     from qiskit.compiler import transpile, assemble
     from qiskit.tools.jupyter import *
     from qiskit.visualization import *
     # Określenie konta IBM Q
     provider = IBMQ.load_account()
```

Wczytywanie biblioteki QISKIT

qiskit.org

git

Tworzenie algorytmu kwantowego--obwodu kwantowego

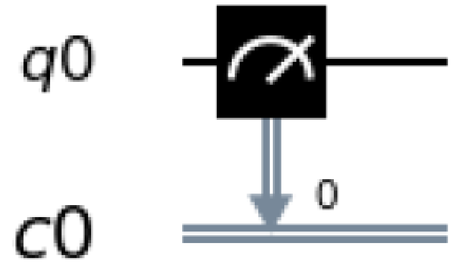
[2]:
```
n0 = 1  # Liczba kubitów i bitów
q0 = QuantumRegister(n0) # Rejestr kwantowy
c0 = ClassicalRegister(n0) # Rejestr klasyczny

circuit0 = QuantumCircuit(q0, c0) # Algorytm kwantowy - kwantowy obwód
circuit0.measure(q0[0], c0[0]) # Sprawdzenie stanów kubitów - kwantowy pomiar
```

[2]: <qiskit.circuit.instructionset.InstructionSet at 0x7f6380873850>

[3]:
```
circuit0.draw(output='mpl') # Rysowanie obwodu
```

[3]:

Pomiar stanu 1 kubitu



Wykonanie algorytmu kwantowego - obliczenia

```
[4]: # Wykonaj obliczeń kwantowych
backend = Aer.get_backend('qasm_simulator')
job_sim0 = execute(circuit0, backend)
sim_result0 = job_sim0.result()
```

```
# Liczbowa prezentacja wyników
print(sim_result0.get_counts(circuit0))
```

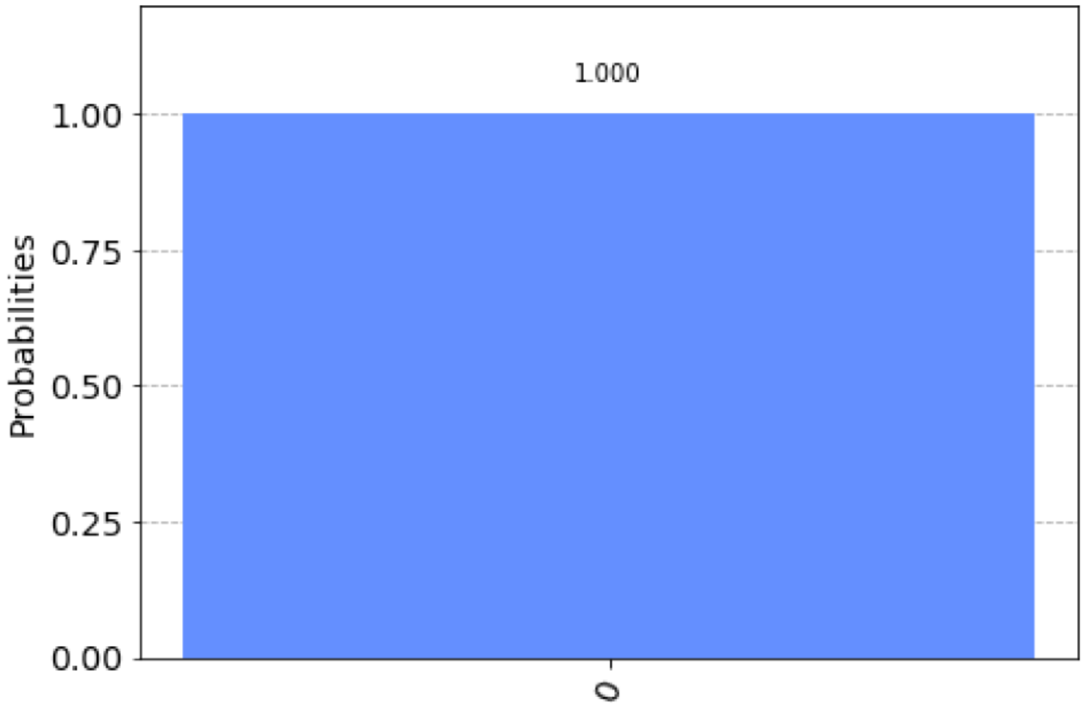
```
{'0': 1024}
```

wynik liczbowy

Graficzna prezentacja wyników

[5]:
```
# Graficzna prezentacja wyników
plot_histogram(sim_result0.get_counts(circuit0))
```

[5]:

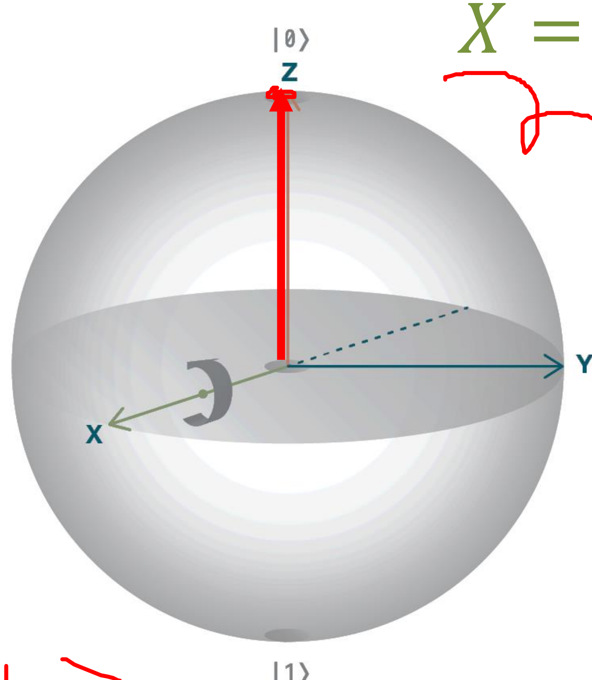


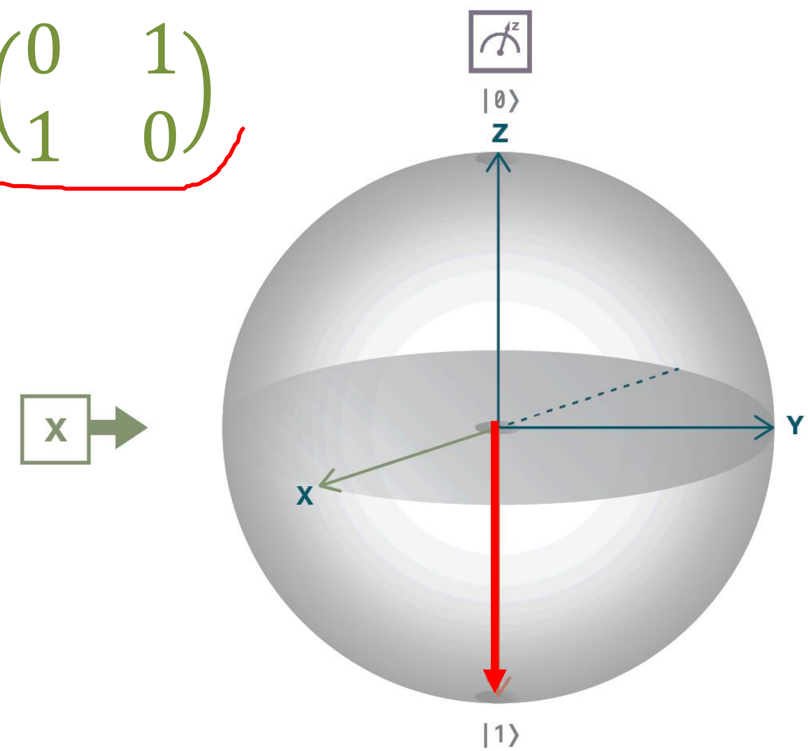
2. OPERACJE (BRAMKI) JEDNOKUBITOWE

Bramka kwantowa X = kwantowa negacja

$\hat{G}_x$, $\hat{G}_1$

$$\hat{X} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

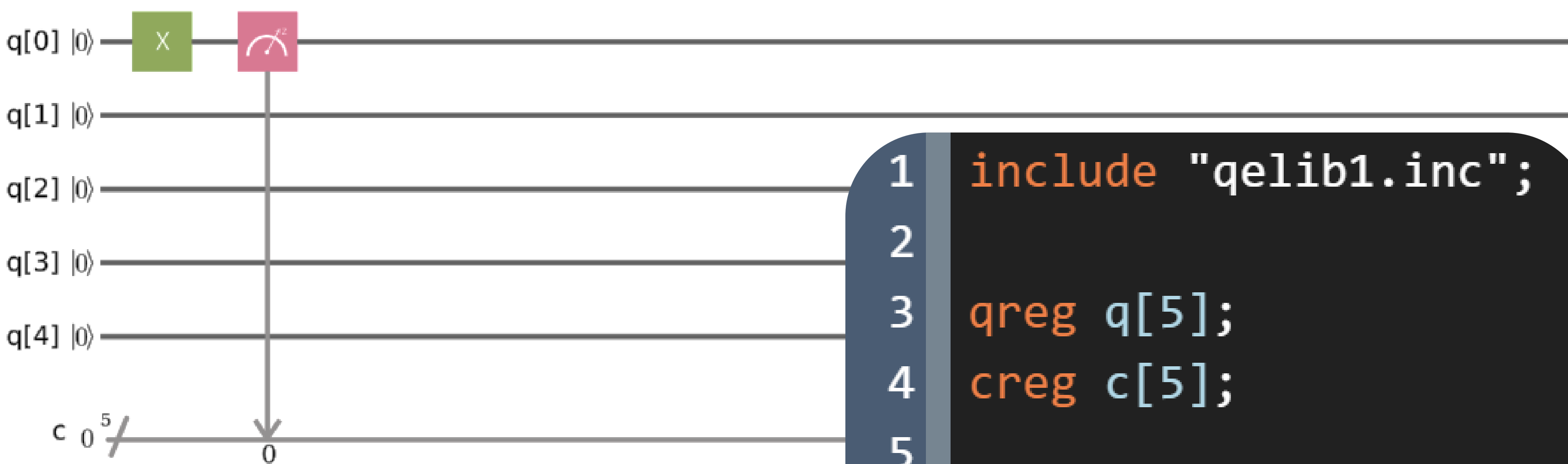




$\hat{X}|1\rangle = |0\rangle$

$$X\,|0\rangle = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} = |1\rangle$$

Działanie i odczyt wyniku działania bramki X

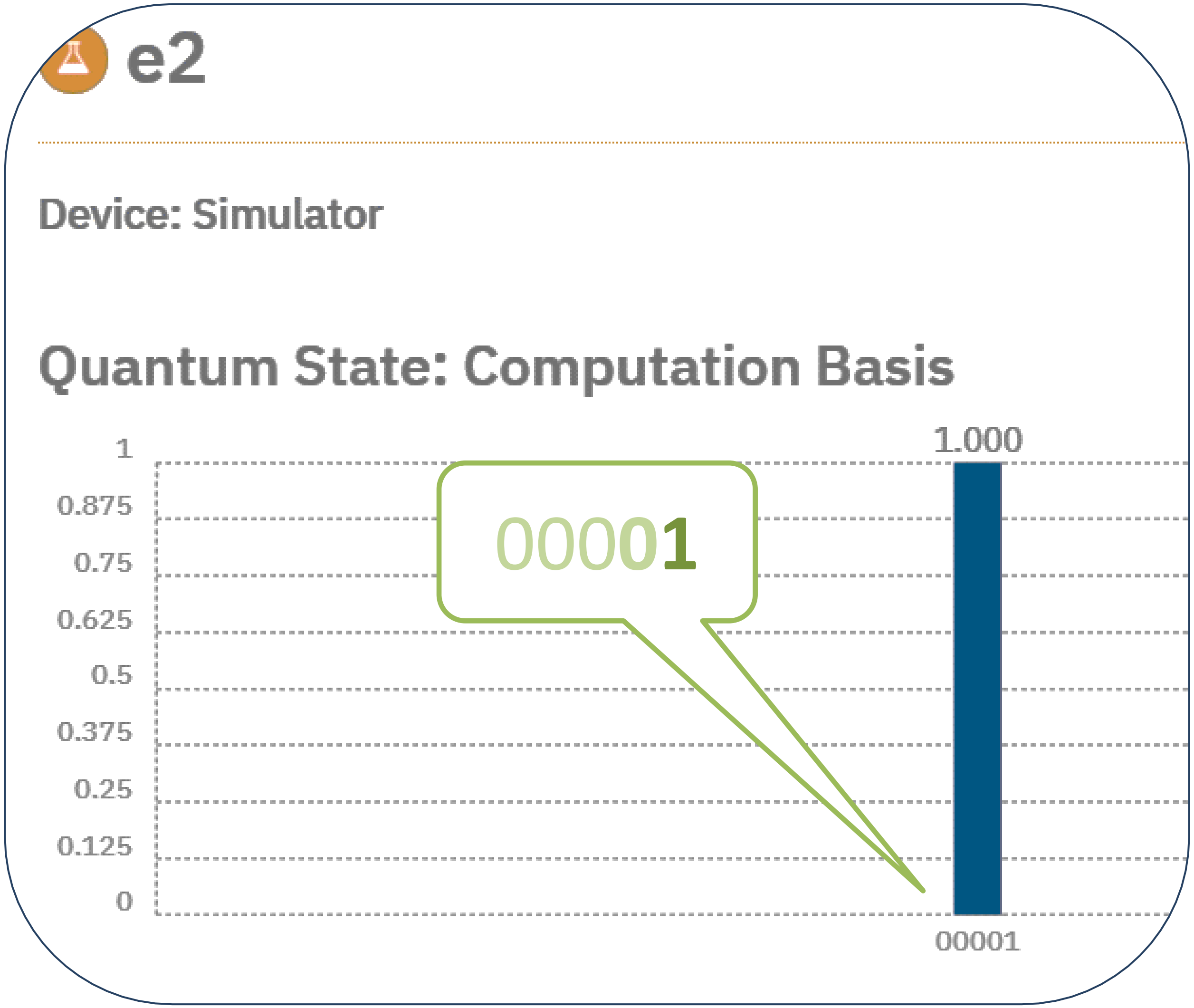


Kod graficzny

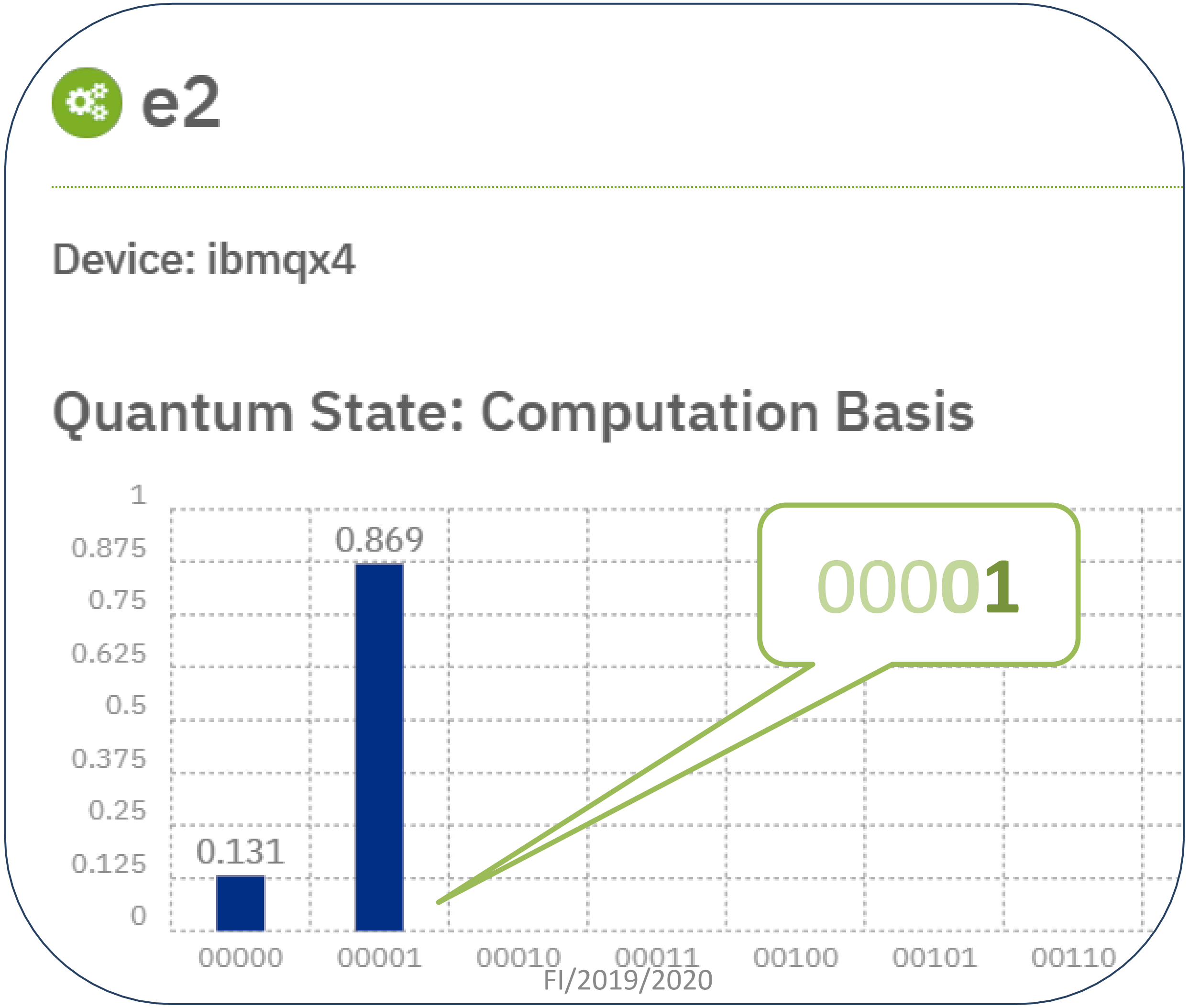
```
include "qelib1.inc";

qreg q[5];
creg c[5];

x q[0];
measure q[0] -> c[0];
```

Kod w asemblerze kwantowym (QASM)

Wynik symulacji działania bramki X



Wynik ściśle kwantowych obliczeń na kwantowym komputerze



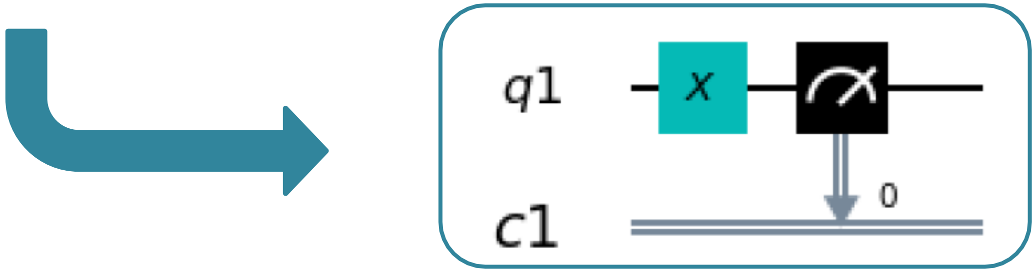
Tworzenie algorytmu kwantowego-obwodu kwantowego

[6]:
```
# Zbuduj obwód kwantowy

n = 1  # liczba kubitów
q1 = QuantumRegister(n) # rejestr kwantowy
c1 = ClassicalRegister(n) # rejestr klasyczny

circuit1 = QuantumCircuit(q1, c1)

circuit1.x(q1[0]) # Operacja X realizowana na kubicie 1[0]
circuit1.measure(q1[0], c1[0])
```

[6]:
```
<qiskit.circuit.instructionset.InstructionSet at 0x7f63807786d0>
```

[7]:
```
circuit1.draw(output='mpl') # rysowanie obwodu
```

[7]:



Wykonywanie obliczeń kwantowych

[8]:
```
# Wykonaj i zaprezentuj obliczenia
backend = Aer.get_backend('qasm_simulator')
job_sim1 = execute(circuit1, backend)
sim_result1 = job_sim1.result()

print(sim_result1.get_counts(circuit1))
```

```
{'1': 1024}
```

[9]:
```
plot_histogram(sim_result1.get_counts(circuit1))
```

Wykonywanie obliczeń kwantowych

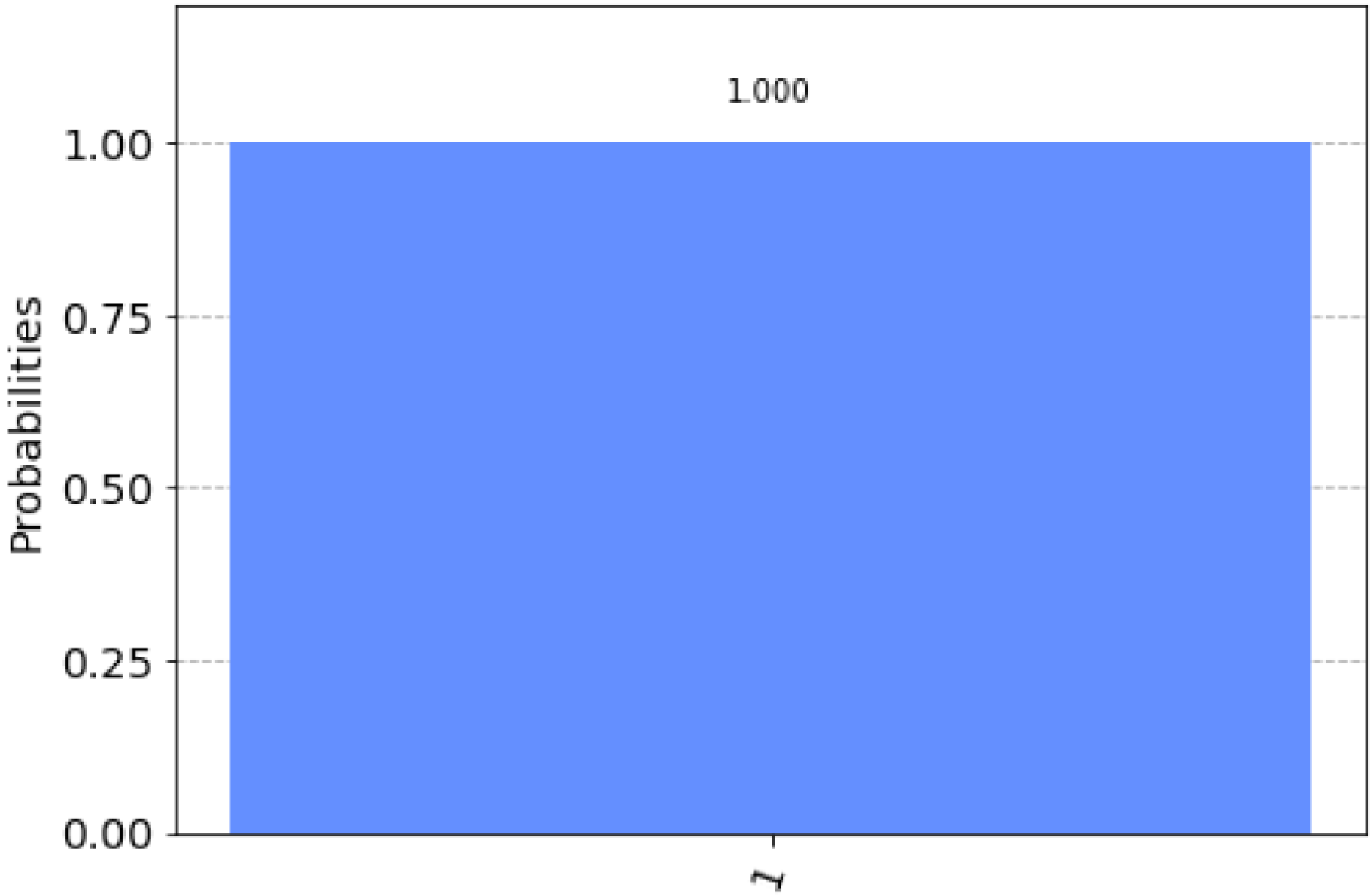
[8]:
```
# Wykonaj i zaprezentuj obliczenia
backend = Aer.get_backend('qasm_simulator')
job_sim1 = execute(circuit1, backend)
sim_result1 = job_sim1.result()

print(sim_result1.get_counts(circuit1))
```

```
{'1': 1024}
```

[9]:
```
plot_histogram(sim_result1.get_counts(circuit1))
```

Graficzne przedstawienie wyniku

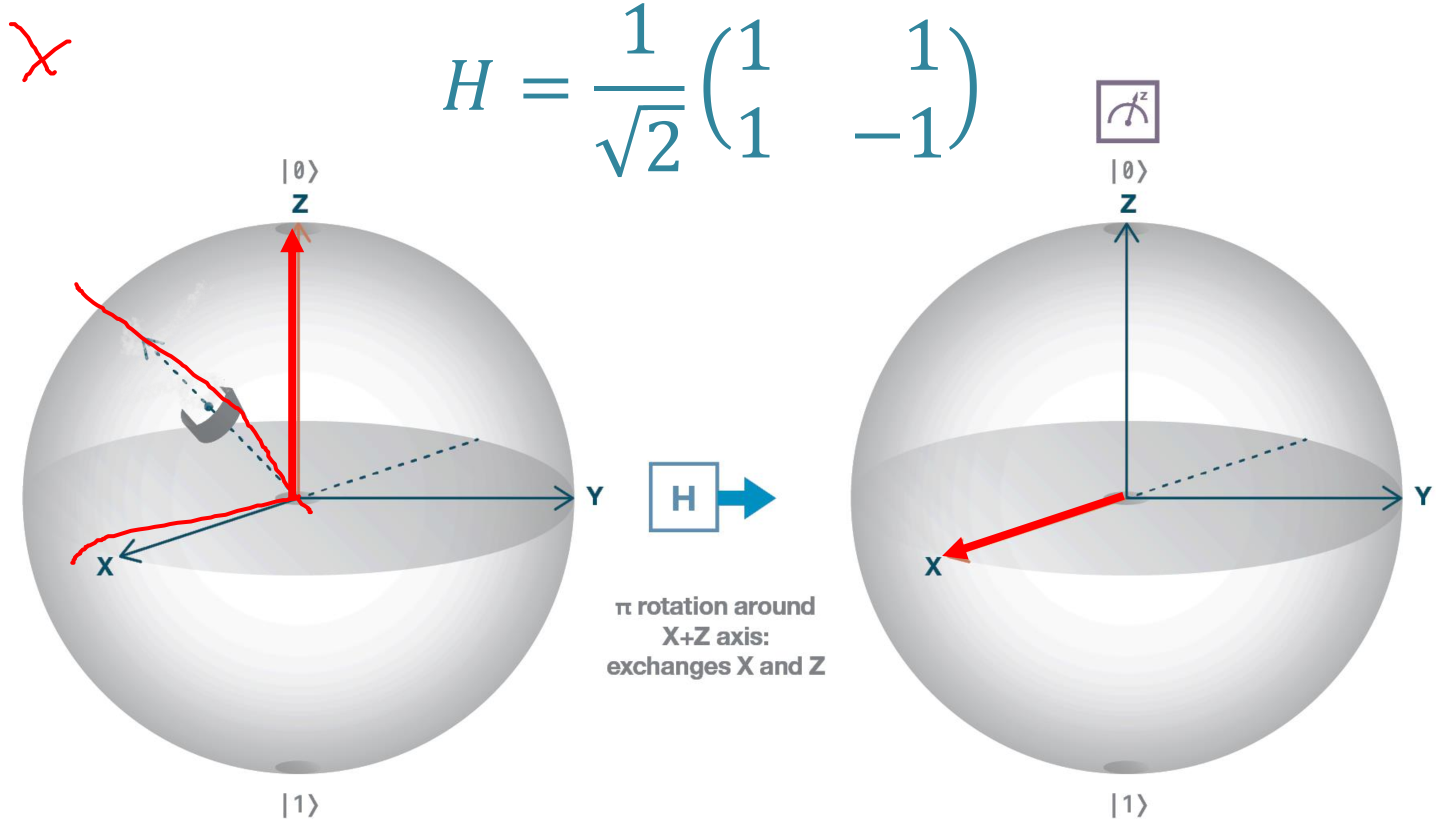
[9]:



3. SUPERPOZYCJA STANÓW

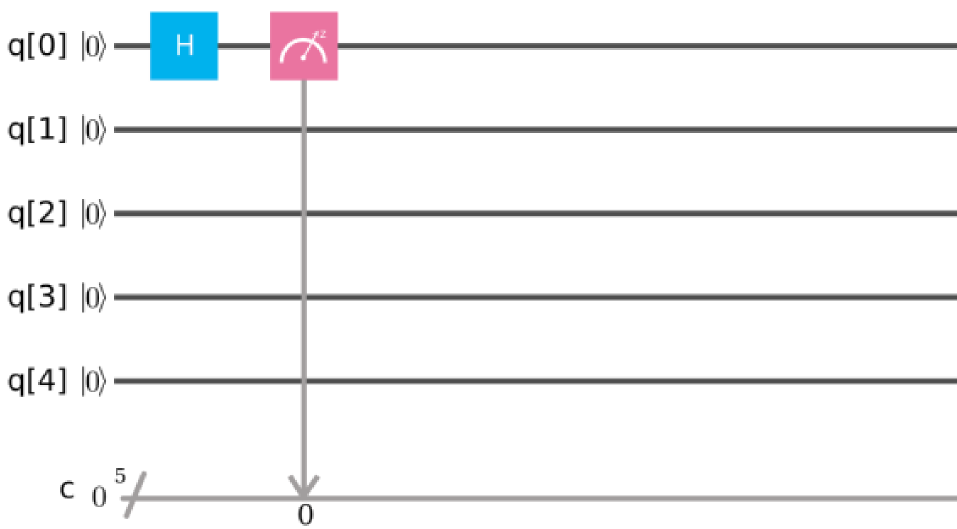
Bramka kwantowa Hadamarda

$$H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$



$$\mathrm{H}\,|0\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}\begin{pmatrix} 1 \\ 0 \end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 1 \end{pmatrix} = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

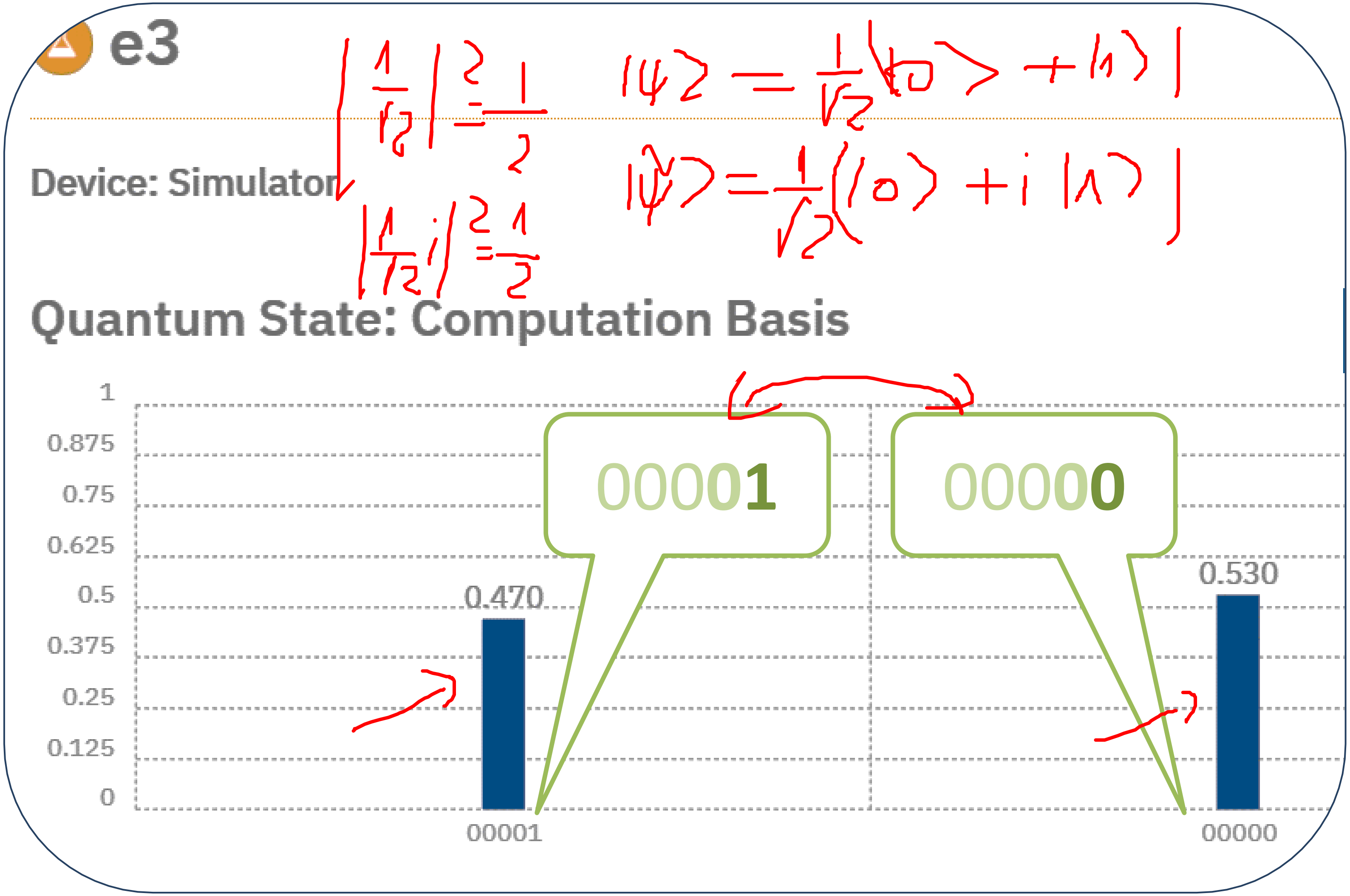
Działanie i odczyt wyniku działania bramki H (Hadamarda)




```
include "qelib1.inc";

qreg q[5];
creg c[5];

h q[0];
measure q[0] -> c[0];
```

Kod w asemblerze kwantowym (QASM)

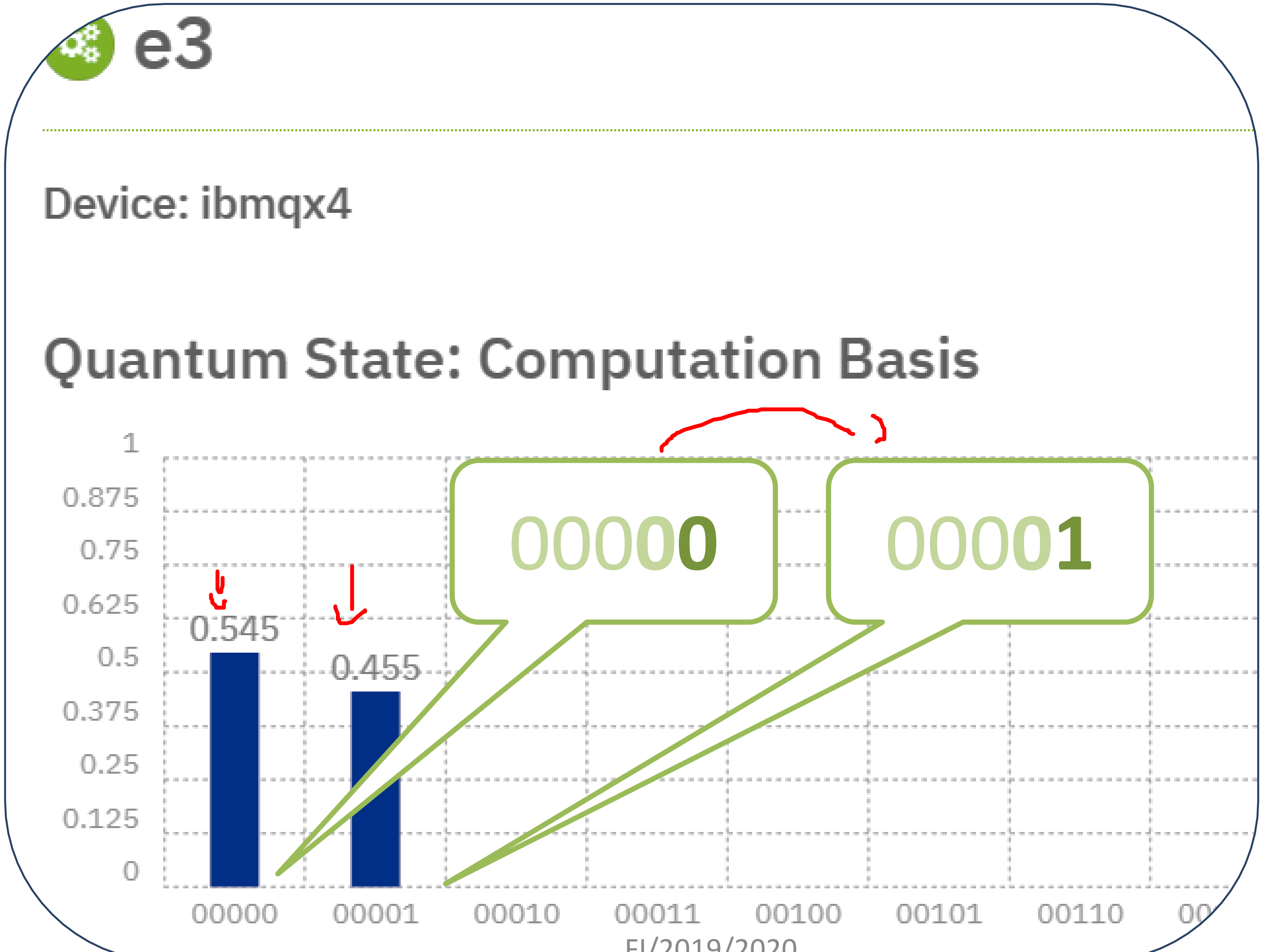
Bramka H pozwala zdefiniować generatorem liczb losowych – wynik symulacji



$$\left|\frac{1}{\sqrt{2}}\right|^2 = \frac{1}{2} \qquad |\psi\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right)$$

$$\left|\frac{i}{\sqrt{2}}\right|^2 = \frac{1}{2} \qquad |\psi\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle + i\,|1\rangle\right)$$

Bramka H – wynik obliczeń z zastosowaniem fizycznej bramki Hadamarda



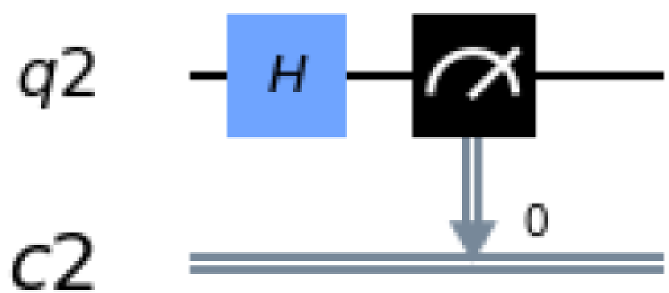
Tworzenie obwodu z bramką Hadamarda

[10]:
```
q2 = QuantumRegister(n) # rejestr kwantowy
c2 = ClassicalRegister(n) # rejestr klasyczny

circuit2 = QuantumCircuit(q2, c2)

circuit2.h(q2[0]) # Operacja H realizowana na kubicie q2[0]
circuit2.measure(q2[0], c2[0])
```

[10]:
```
<qiskit.circuit.instructionset.InstructionSet at 0x7f637fe08ad0>
```

[11]:
```
circuit2.draw(output='mpl') # rysowanie obwodu
```

[11]:



Wykonywanie obliczeń

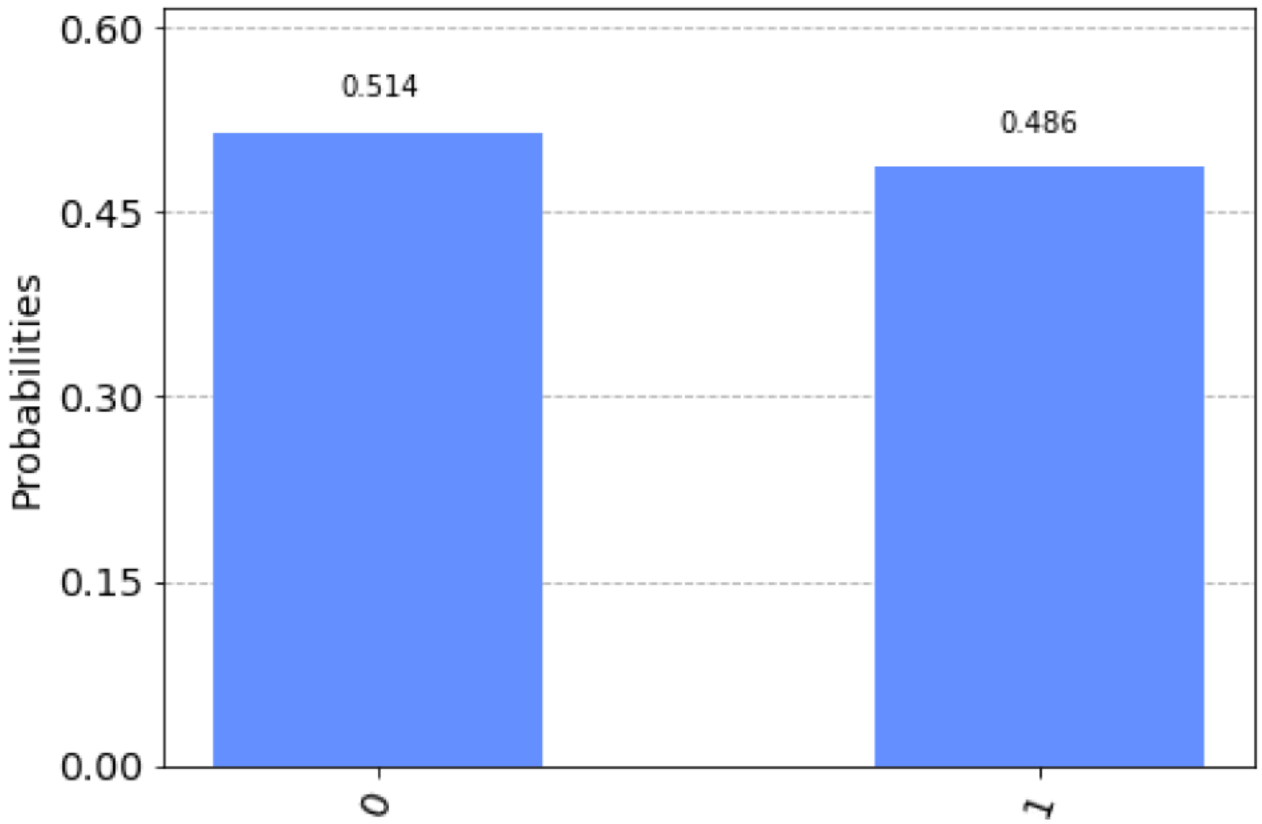
```
[12]: # Wykonaj i zaprezentuj obliczenia
job_sim2 = execute(circuit2, backend)
sim_result2 = job_sim2.result()

print(sim_result2.get_counts(circuit2))
```

```
{'0': 526, '1': 498}
```

Wykonywanie obliczeń i prezentacja wyników

```
[13]: plot_histogram(sim_result2.get_counts(circuit2))
```

[13]:

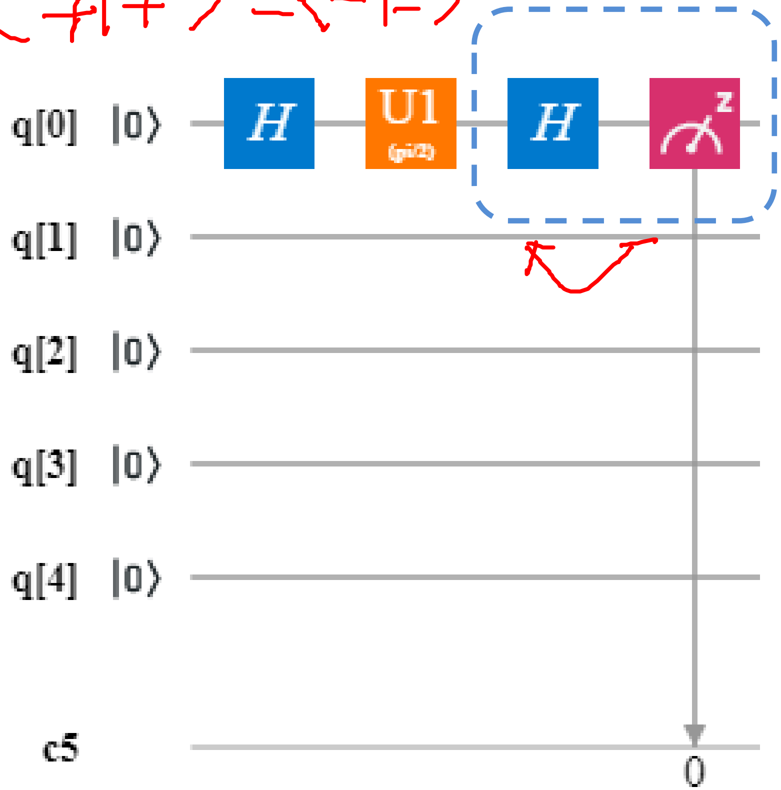


4. Tomografia stanu jednego kubitu – realizacje na komputerze kwantowym

Pomiar w bazie X

$\langle + | - \rangle = 0$

$\langle + | + \rangle = \langle - | - \rangle = 1$



X

$p_0 = |\langle 0 | \psi \rangle|^2$

$p_1 = |\langle 1 | \psi \rangle|^2$

$p_{\pm} = |\langle \pm | \psi \rangle|^2$

$$\sigma_1 = \underline{X} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

$|0\rangle$, $|1\rangle$

$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$

$|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$

$|-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$

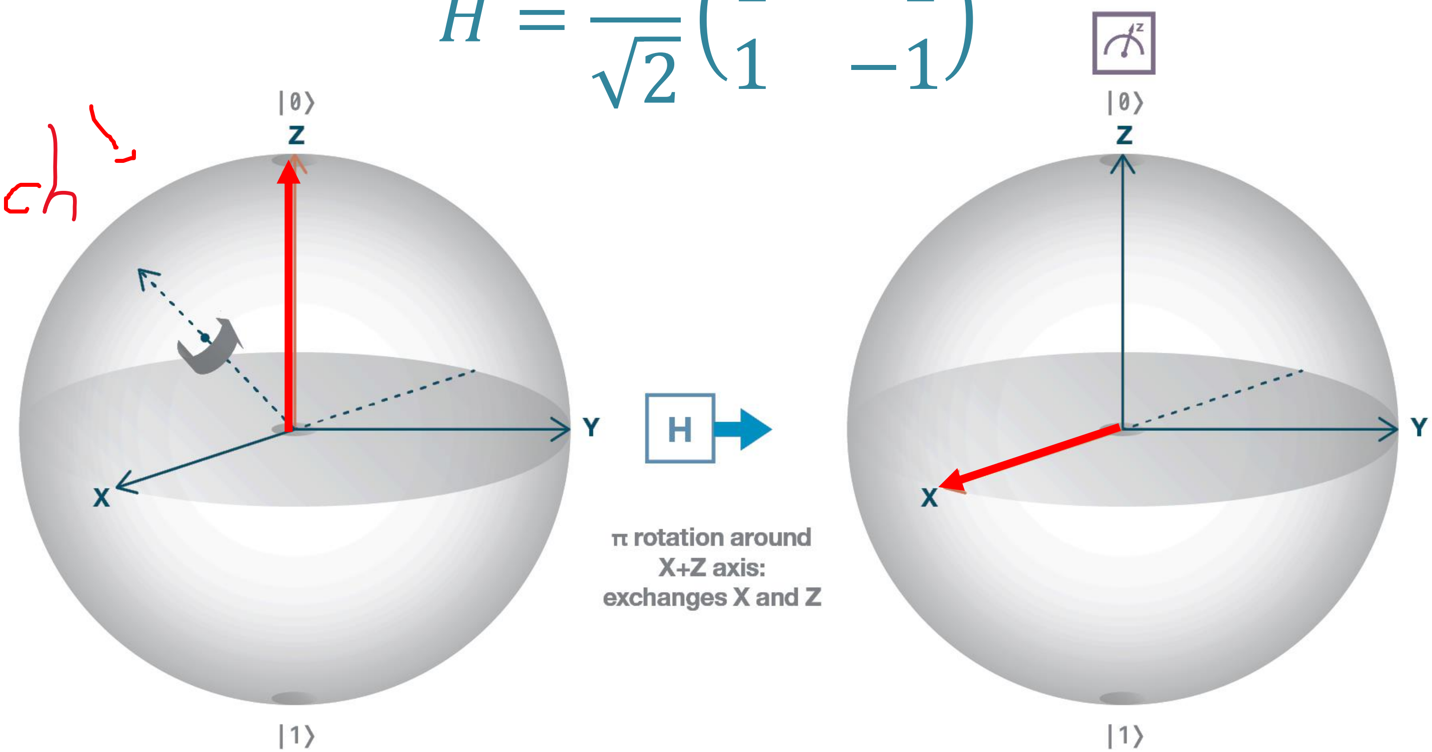
$|0\rangle \rightarrow \lambda = 1$

$|1\rangle \rightarrow \lambda = -1$

Bramka kwantowa Hadamarda

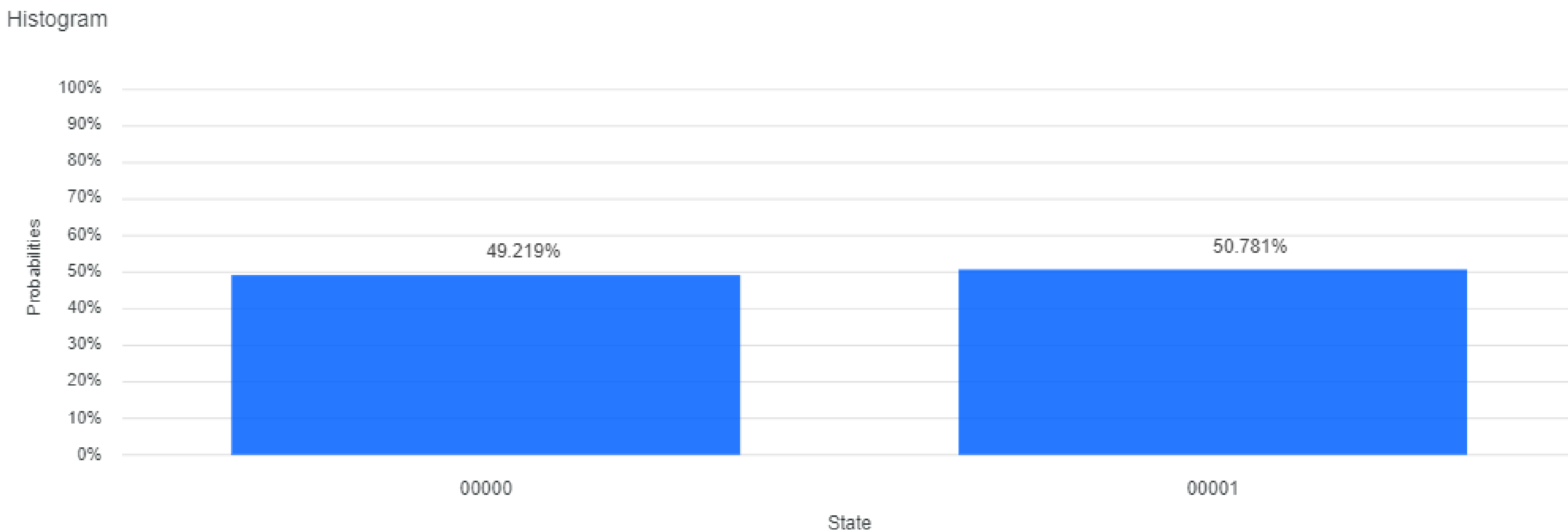
$$H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

Bloch

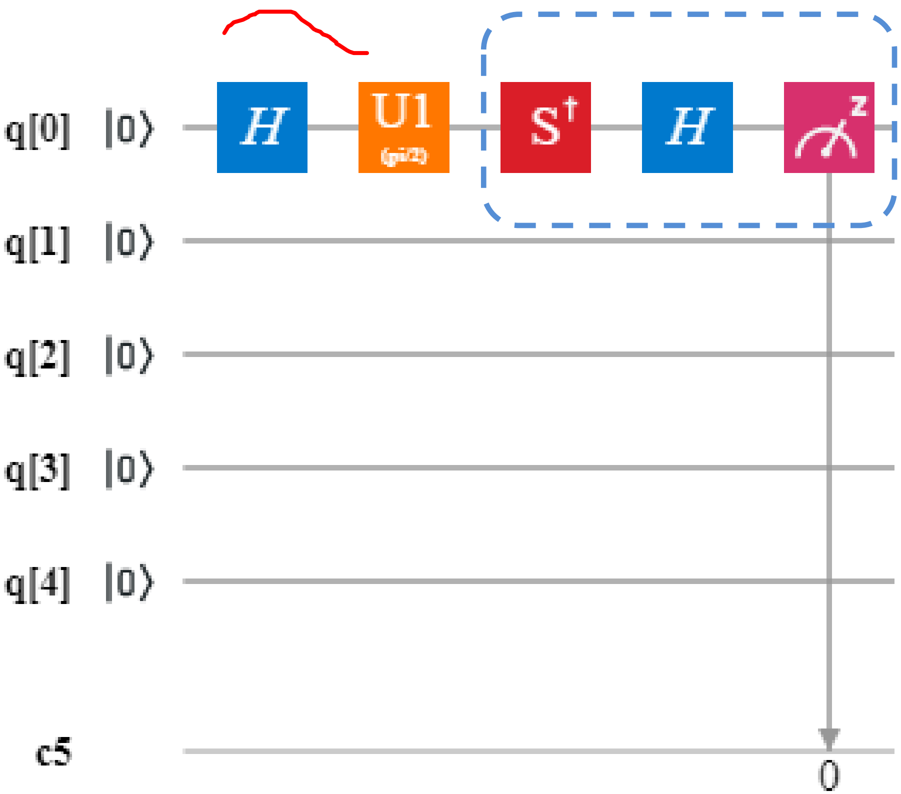


$$\mathrm{H}\,|0\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}\begin{pmatrix} 1 \\ 0 \end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 1 \end{pmatrix} = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

Wynik pomiaru w bazie X



Pomiar w bazie Y



$$p_{R/L} = |\langle R/L | \psi \rangle|^2$$

$$\sigma_2 = \cancel{X}\,Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$$

$$|R\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle + i|1\rangle\right)$$

$$|L\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle - i|1\rangle\right)$$

$$Y|R\rangle = +|R\rangle$$

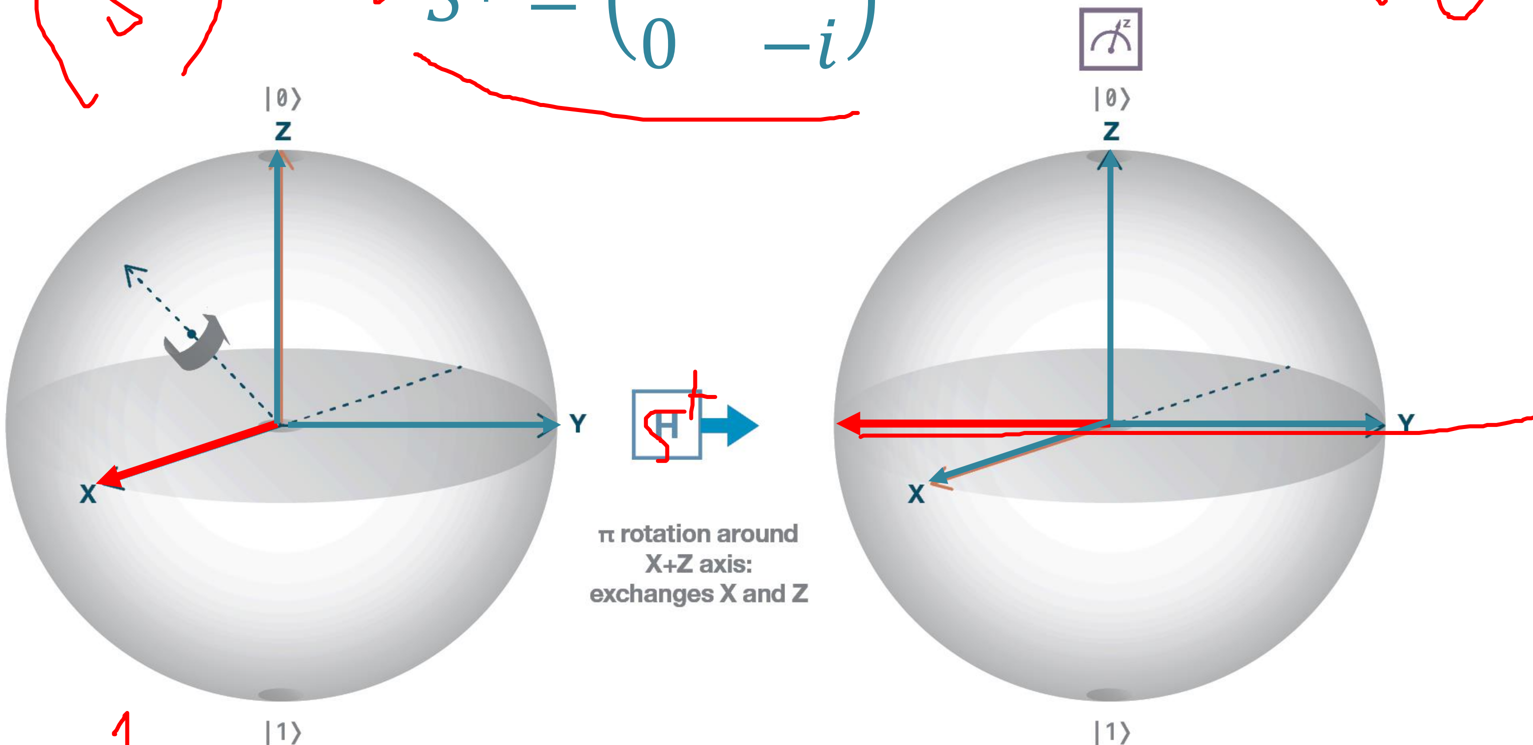
$$Y|L\rangle = -|L\rangle$$

Bramka kwantowa Hadamarda

$S^{\dagger} = (S^{T})^{*}$

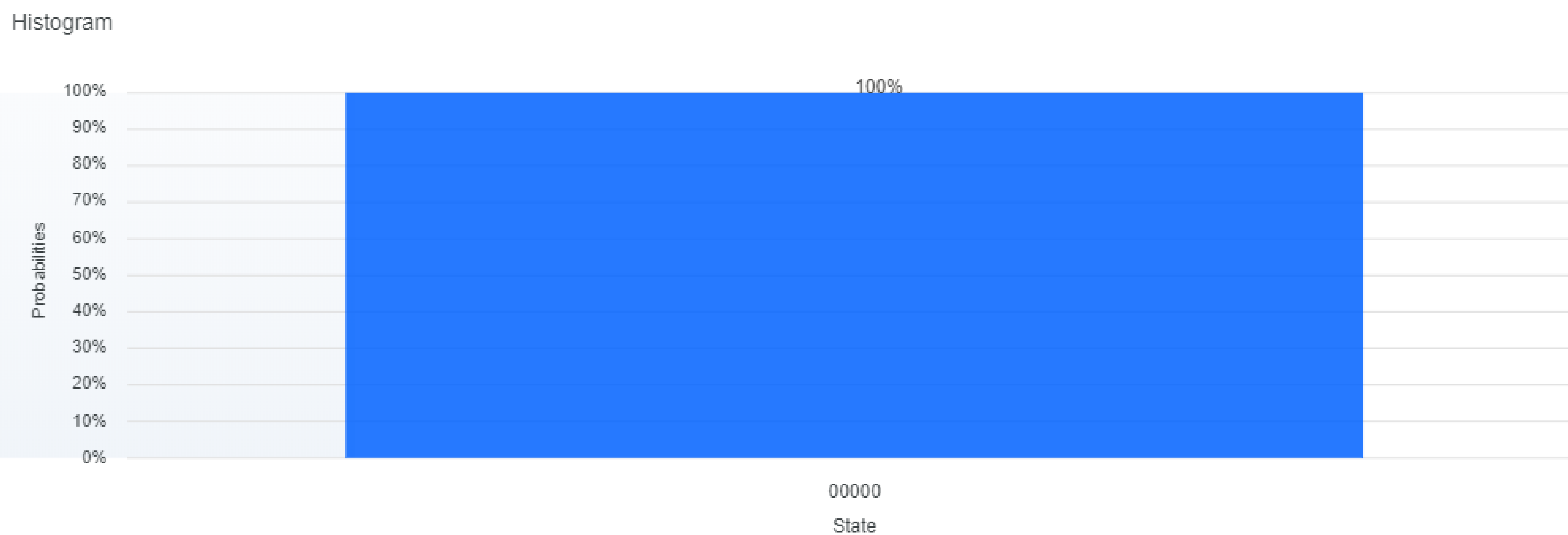
$$S^{\dagger} = \begin{pmatrix} 1 & 0 \\ 0 & -i \end{pmatrix}$$

$S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$

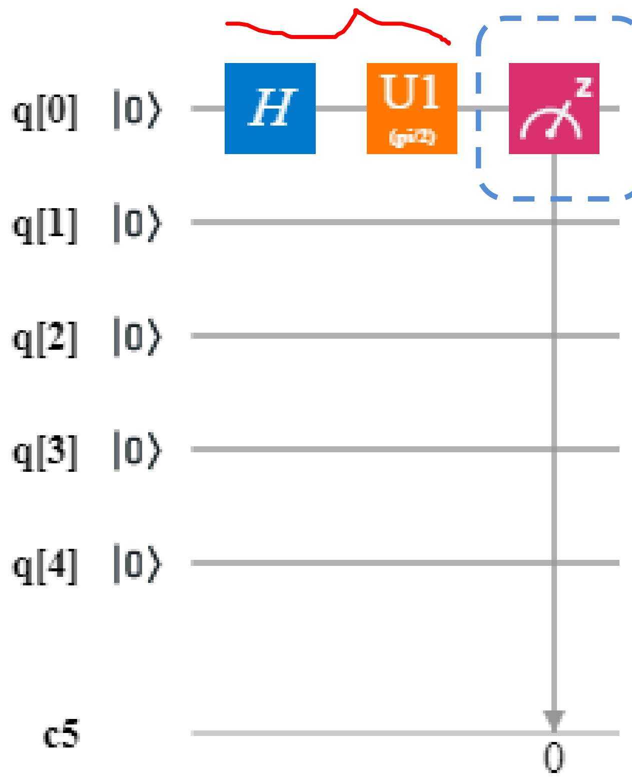


$$S^{\dagger}\,|0\rangle = \begin{pmatrix} 1 & 0 \\ 0 & -i \end{pmatrix} \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -i \end{pmatrix} = \frac{1}{\sqrt{2}} (|0\rangle - i|1\rangle)$$

Wynik pomiaru w bazie Y



Pomiar w bazie Z

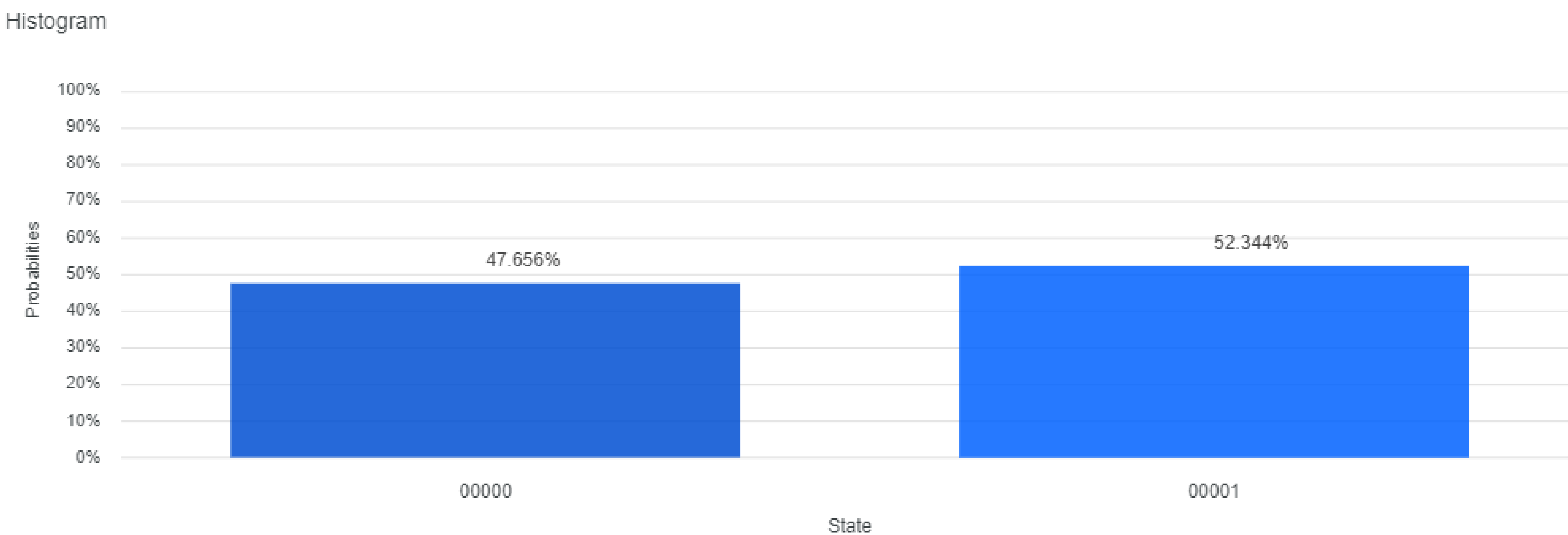


$$\sigma_3 = Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

$\rightarrow |0\rangle$

$|1\rangle$

Wynik pomiaru w bazie Z



POMIAR WEKTORA STANU-
-POMIAR PO ZESPOLE

$p_{\psi \to \pm} = |\langle 0/1|\psi\rangle|^2 = |c_\pm|^2$

$|\psi\rangle = \cos\frac{\theta}{2}\,|0\rangle + e^{i\phi}\sin\frac{\theta}{2}\,|1\rangle$

$p_{\psi \to 0/1} \approx \frac{n_{0/1}}{n}$

$$p = \cos^2\frac{\theta}{2} \qquad 1 - p = \sin^2\frac{\theta}{2}$$

dla bardzo dużej liczby cząstek pomiar σ_3

$$\cos\varphi = \frac{q - 1/2}{\sqrt{p(1-p)}}$$

q –prawdopodobieństwo znalezienia kubitu „0” w pomiarze X

dla bardzo dużej liczby cząstek pomiar σ_1

ZADANIE 2

Inne jednokubitowe bramki kwantowe

$$U_1(\lambda) = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\lambda} \end{bmatrix},$$

$$U_2(\lambda, \phi) = \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{e^{i\lambda}}{\sqrt{2}} \\ \frac{e^{i\phi}}{\sqrt{2}} & \frac{e^{i(\lambda+\phi)}}{\sqrt{2}} \end{bmatrix}$$

$$U_3(\theta, \phi, \lambda) = \begin{bmatrix} \cos\left(\frac{\theta}{2}\right) & -e^{i\lambda}\sin\left(\frac{\theta}{2}\right) \\ e^{i\phi}\sin\left(\frac{\theta}{2}\right) & e^{i(\lambda+\phi)}\cos\left(\frac{\theta}{2}\right) \end{bmatrix}$$

Pobieranie notatnika *Jupyter* w formacie pdf