IPE

Wednesday, June 21, 2023 17:12

* **Phase Estimation:**

Given: Unitary Matrix U

Eigenstate of U : $|\psi\rangle$

Unknown eigenvalue of U : $e^{i2\pi\phi}$

Estimate : ϕ

U and $|\psi\rangle$ will be given as a quantum circuit (or gate) to us

Binary expansion of ϕ is

$$\phi = \frac{\phi_1}{2} + \frac{\phi_2}{4} + \frac{\phi_3}{8} + \dots + \frac{\phi_n}{2^n} = 0.\phi_1\phi_2\phi_3\dots\phi_n \qquad \varphi_i \in \{0,1\}$$

↳ decimal point notation in base 2

We need two additional qubits:

q_0 - auxiliary qubit (initialized to $|+\rangle$ state)

q_1 - qubit on which U operates (i.e. eigenstate $|\psi\rangle$)

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

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

$$|+\rangle|\psi\rangle \xrightarrow{\text{controlled } U^{2^t}} \frac{|0\rangle|\psi\rangle + e^{i2\pi 2^t\phi}|1\rangle|\psi\rangle}{\sqrt{2}}$$

$$= \left(\frac{|0\rangle + e^{i2\pi 2^t\phi}|1\rangle}{\sqrt{2}}\right)|\psi\rangle$$

→ system qubit (i.e eigenstate) remained same, phase $e^{i2\pi 2^t\phi}$ is kicked back into auxiliary qubit

Now, lets analyse phase

$$e^{i2\pi 2^t\phi} = e^{i2\pi 2^t(0.\phi_1\phi_2\dots\phi_m)}$$

$$2^t(0.\phi_1\phi_2\dots\phi_n) = 2^t \times \left(\frac{\phi_1}{2} + \frac{\phi_2}{4} + \frac{\phi_3}{8} + \dots + \frac{\phi_m}{2^m}\right)$$

$\varphi_1 \to 2^t/2 = 2^{t-1}$

$\varphi_2 \to 2^t/4 = 2^{t-2}$

$\varphi_3 \to 2^t/8 = 2^{t-3}$

$\vdots$

$\varphi_t \to 2^t/2^t = 1$

$\varphi_{t+1} \to 2^t/2^{t+1} = 1/2$

$\varphi_{t+2} \to 2^t/2^{t+2} = 1/4$

$\vdots$

$\varphi_m \to 2^t/2^m = 2^{t-m} = 1/2^{m-t}$

$$e^{i2\pi\left(2^{t-1}\varphi_1 + 2^{t-2}\varphi_2 + \dots \varphi_t + \frac{\varphi_{t+1}}{2} + \dots + \frac{\varphi_m}{2^{m-t}}\right)}$$

powers are ≥ 0

$e^{i2\pi n} = 1$ where $n \geq 0$

$$= e^{i2\pi\left(\frac{\varphi_{t+1}}{2} + \frac{\varphi_{t+2}}{4} + \dots + \frac{\varphi_m}{2^{m-t}}\right)} \rightarrow \text{new binary decimal}$$

$$= e^{i2\pi(0.\varphi_{t+1}\varphi_{t+2}\dots\varphi_m)}$$

$\therefore$ for $t=0$, $e^{i2\pi 2^0\varphi} = e^{i2\pi\varphi} = e^{i2\pi 0.\varphi_1\varphi_2\dots\varphi_m}$

$t=1$, $e^{i2\pi 2\varphi} = e^{i2\pi 2\cdot\varphi} = e^{i2\pi\varphi_1} e^{i2\pi 0.\varphi_2\varphi_3\dots\varphi_m} = e^{i2\pi 0.\varphi_2\varphi_3\dots\varphi_m}$

$t=2$, $e^{i2\pi 4\varphi} = e^{i2\pi 4\varphi} = e^{i2\pi 2\cdot\varphi_1} e^{i2\pi\varphi_2} e^{i2\pi 0.\varphi_3\varphi_4\dots\varphi_m} = e^{i2\pi 0.\varphi_3\varphi_4\dots\varphi_m}$

$t=m-1$, phase would be - $e^{i2\pi 2^{m-1}\phi} = e^{i2\pi 0.\varphi_m}$

for $t=m-1$: phase $= e^{i2\pi} = 1$ if $\varphi_m = 0$; phase $= e^{i2\pi\cdot\frac{1}{2}} = -1$ if $\varphi_m = 1$

for $t=m-1$: $|+\rangle|\psi\rangle = \left(\frac{|0\rangle + |1\rangle}{\sqrt{2}}\right)|\psi\rangle = |+\rangle|\psi\rangle$ if $\varphi_m = 0$

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

$\varphi_m = 0/1$ can be distinguished if measured in X basis i.e. performing Hadamard before measurement

* **IPE Algorithm**

1. Directly measure LSB of $\varphi = \varphi_m$ by:
   initializing 2-qubit registers $q_0 - |+\rangle$, $q_1 - |\psi\rangle$
   performing $CU^{2^{m-1}}$ on $q_1 \to |\psi\rangle$
   measuring $|q_0\rangle$ in pauli X basis
   $|q_0q_1\rangle = |+\rangle|\psi\rangle \xrightarrow{\text{step 1}} |-\rangle|\psi\rangle$ if $\varphi_m = 1$
2. Now we want to find out φ_{m-2}
   Reinitialize $|q_0\rangle \to |+\rangle$ & $|q_1\rangle \to |\psi\rangle$
   Apply $C.U^{2^{m-2}}(q_0, q_1)$ we get

$$2^{m-2}\left(\frac{\varphi_{m-1}}{2^{m-1}} + \frac{\varphi_m}{2^m}\right) = \left(\frac{\varphi_{m-1}}{2} + \frac{\varphi_m}{4}\right)$$

$$|+\rangle|\psi\rangle \xrightarrow{\text{step 2}} e^{i2\pi\left(\frac{\varphi_{m-1}}{2} + \frac{\varphi_m}{4}\right)} = e^{i2\pi 0.\varphi_{m-1}} \cdot e^{i2\pi\varphi_m/4}$$

→ we need to remove this to get just φ_{m-2}

- can be done by rotating q_0 around Z axis by $-\left(\frac{2\pi\varphi_m}{4}\right) = -\frac{\pi\varphi_m}{2} = -\pi/2$ if $\varphi_m = 1$

After phase correction, measure q_0 in X basis

k^{th} step: reinitialize $|q_0\rangle \to |+\rangle$ & $|q_1\rangle \to |\psi\rangle$
apply $CU^{2^{m-k}}$ to get φ_{m-k+1}
phase to be corrected $= 0.0\varphi_{m-k+2}\varphi_{m-k+3}\dots\varphi_m$ using rotation around Z axis
measure $|q_0\rangle$ in X basis

Example - S gate

$$S = \sqrt{Z} = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/2} \end{bmatrix}$$

eigenstate : $|\psi\rangle = |1\rangle$ two eigenvalues $S|0\rangle = 1|0\rangle$

eigenvalue : $e^{i\pi/2} = e^{2\pi i(1/4)}$ $\quad S|1\rangle = e^{i\pi/2}|1\rangle$

$\therefore$ phase $= 1/4$

in binary expansion $= 0\cdot\frac{1}{2^{-1}} + 1\cdot\frac{1}{2^{-2}} = 0.01 = 0.\phi_1\phi_2$

$\boxed{\phi_2 = 1 \quad \phi_2 = 0}$

```
In [8]:
from qiskit import ClassicalRegister, QuantumCircuit, QuantumRegister
import numpy as np
```

```
In [18]:
def s_gate():
    qc = QuantumCircuit(1)
    qc.s(0)
    return qc.to_gate(label="S gate")

controlled_S_gate = s_gate().control(1)
```

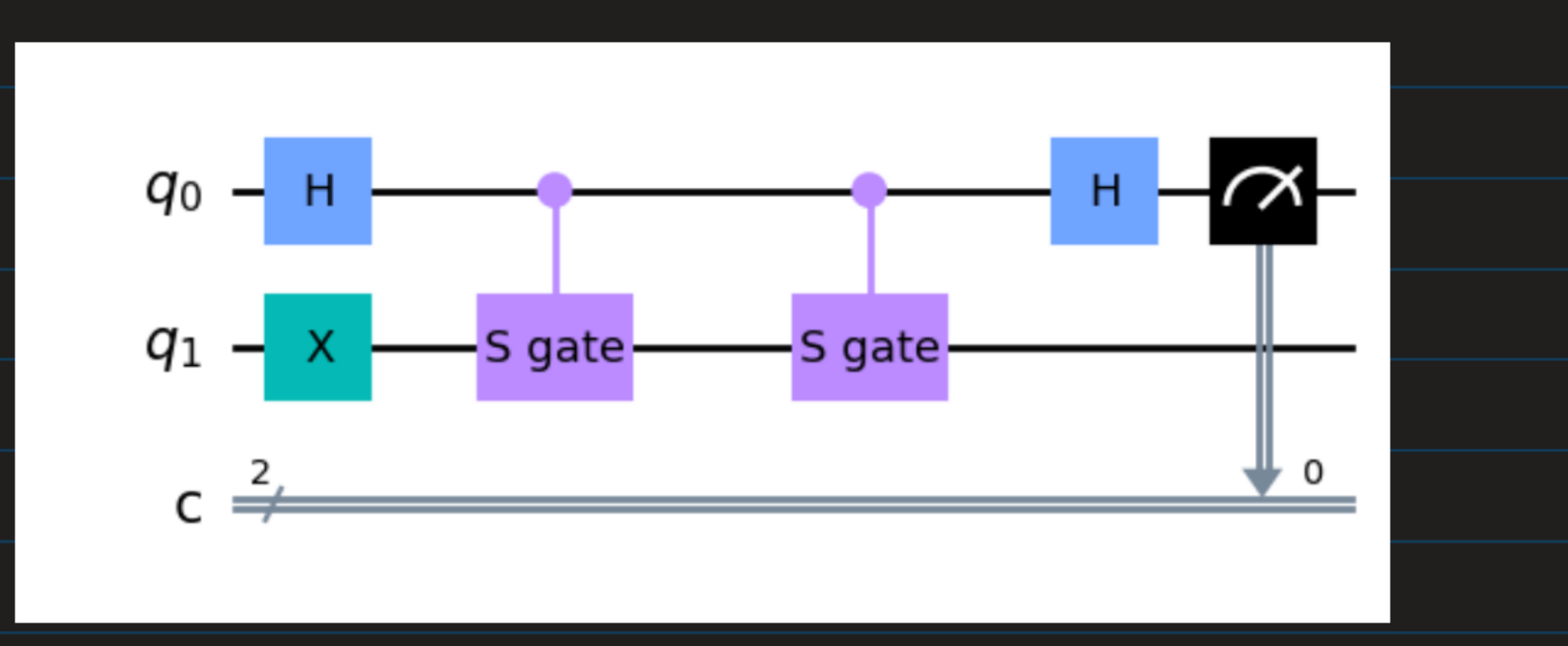
create a controlled S- Gate

```
In [28]:
qr = QuantumRegister(2, "q")
cr = ClassicalRegister(2, "c")
qc = QuantumCircuit(qr, cr)

q0, q1 = qr
qc.h(q0) # auxiliary qubit
qc.x(q1) # system qubit i.e. eigenstate |1>

# step 1
k = 1 # will get phi_2
for i in range(2**k):
    qc.append(controlled_S_gate,[q0,q1])

qc.h(q0) # measure in X basis
c0, c1 = cr
qc.measure(q0, c0)
```



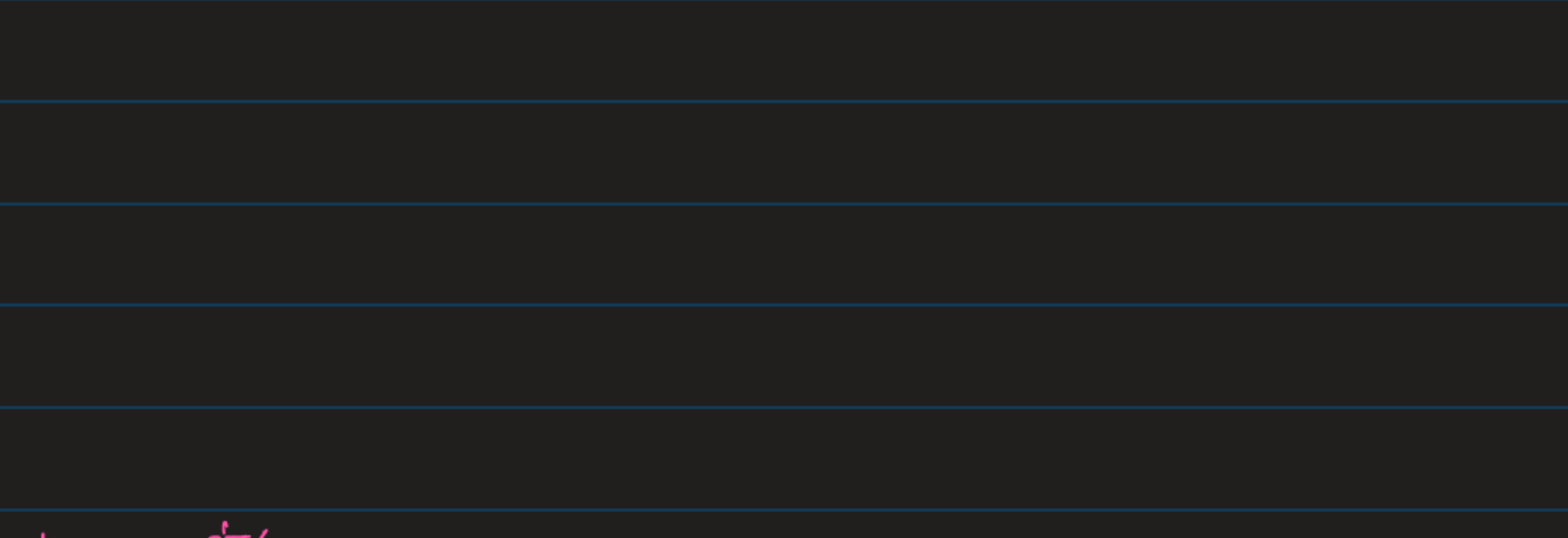
Step 1

```
# reinitialize auxiliary qubit
qc.reset(q0)
qc.h(q0)
```

<qiskit.circuit.instructionset.InstructionSet at 0x20d112d02e0>

```
# phase correction from m1
c0, c1 = cr
with qc.if_test((c0, 1)):
    qc.p(-np.pi / 2, q0)
```

```
# step 2
k = 0 # will get phi_2
for i in range(2**k):
    qc.append(controlled_S_gate,[q0,q1])
qc.h(q0) # measure in X basis
qc.measure(q0, c1)
```



Step 2

```
from qiskit_aer import AerSimulator

sim = AerSimulator()
job = sim.run(qc, shots=1000)
result = job.result()
counts = result.get_counts()
counts
```

→ Ran on simulator

{'01': 1000}

$\varphi_1 \quad \varphi_2 \qquad \therefore \varphi = \frac{\varphi_1}{2} + \frac{\varphi_2}{4} = \frac{1}{4} \qquad \therefore$ eigenvalue - $e^{i2\pi\cdot 1/4} = e^{i\pi/2}$

Example: T gate

$$T = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix}$$

eigenstate $= |1\rangle$

eigenvalue - $e^{i\pi/4} = e^{2\pi i(\pi/8)} \to$ phase

$\therefore$ phase $= \frac{1}{8} = 0.2^{-1} + 0.2^{-2} + 1.2^{-3}$

$= (0.001)_2 = 0.\varphi_1\varphi_2\varphi_3$

Step 1 - Same as for S gate applied $T^{2^{3-1}} = T^{2^2} = T^4$

step 2 - Same as for S gate applied $T^{2^{2-1}} = T^2$

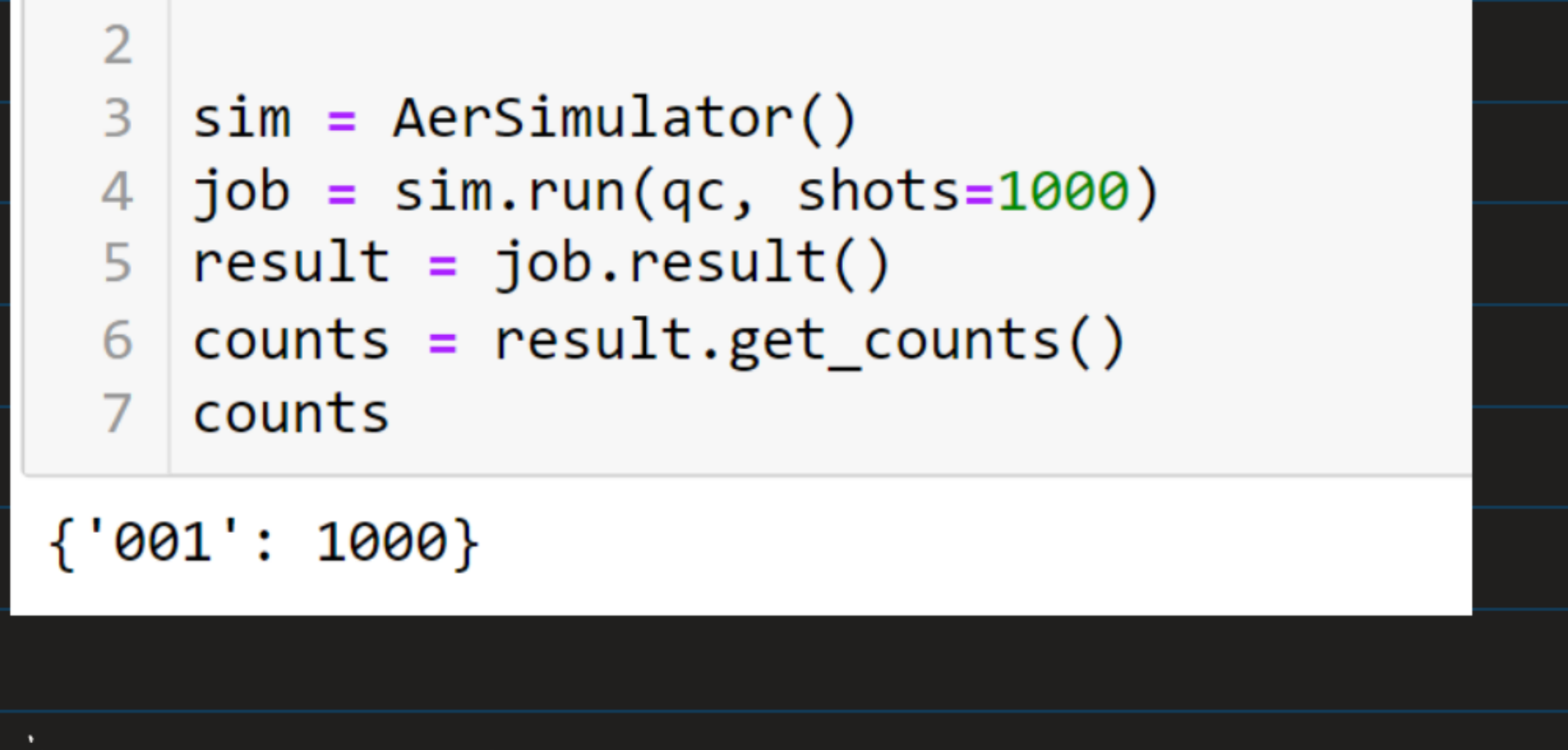
Step 3 - Remove $0.0\varphi_2\varphi_3$ ($c_1\ c_2\ c_3$) applied $T^{2^{1-1}} = T$

i.e if $c_2 = 1$

$$P(-\pi/2) \longrightarrow 0.0\varphi_2 \rightarrow \frac{|0\rangle + e^{2i\pi(0.0\varphi_2)}|1\rangle}{\sqrt{2}} \qquad \frac{\varphi_2}{4} \times 2 = \frac{\varphi_2}{2} \text{ to remove}$$

if $c_3 = 1$

$$P(-\pi/4) \longrightarrow 0.00\varphi_3 \rightarrow \frac{|0\rangle + e^{i\pi}(0.00\varphi_3)|1\rangle}{\sqrt{2}} \qquad \frac{\varphi_3}{8} \times 2 = \frac{\varphi_3}{4} \text{ to remove}$$




```
qr, cr = QuantumRegister(2), ClassicalRegister(3)
qc = QuantumCircuit(qr,cr)

# initialize
qc.h(qr[0])
qc.x(qr[1])

# step 1
k = 2
for i in range(2**k):
    qc.append(controlled_T_gate, qr)

# base pauli X
qc.h(qr[0])

qc.measure(qr[0],cr[0])

# reset aux qubit 0
qc.reset(qr[0])
# reinitialize
qc.h(qr[0])

# phase correction
with qc.if_test((cr[0], 1)):
    qc.p(-np.pi/2, qr[0])

# step 2
k = 1
for i in range(2**k):
    qc.append(controlled_T_gate, qr)

# base pauli X
qc.h(qr[0])

qc.measure(qr[0],cr[1])

# reset aux qubit 0
qc.reset(qr[0])
# reinitialize
qc.h(qr[0])

# phase correction
with qc.if_test((cr[0], 1)):
    qc.p(-np.pi/4, qr[0])
with qc.if_test((cr[1], 1)):
    qc.p(-np.pi/2, qr[0])


# step 3
k = 0
for i in range(2**k):
    qc.append(controlled_T_gate, qr)

# base pauli X
qc.h(qr[0])

qc.measure(qr[0],cr[2])
```

```
from qiskit_aer import AerSimulator

sim = AerSimulator()
job = sim.run(qc, shots=1000)
result = job.result()
counts = result.get_counts()
counts
```

{'001': 1000}

ip $q_0q_1q_2q_3$	$x(ip)$ $q_0q_1q_2q_3$		op $(q_1q_2q_3q_0)$
0 0 0 0	1 1 1 1	→	1 1 1 1
0 0 0 1	1 1 1 0	→	1 1 0 1
0 0 1 0	1 1 0 1	→	1 0 1 1
0 0 1 1	1 1 0 0	→	1 0 0 1
0 1 0 0	1 0 1 1	→	0 1 1 1
0 1 0 1	1 0 1 0	→	0 1 0 1
0 1 1 0	1 0 0 1	→	0 0 1 1
0 1 1 1	1 0 0 0	→	0 0 0 1
1 0 0 0	0 1 1 1	→	1 1 1 0
1 0 0 1	0 1 1 0	→	1 1 0 0
1 0 1 0	0 1 0 1	→	1 0 1 0
1 0 1 1	0 1 0 0	→	1 0 0 0
1 1 0 0	0 0 1 1	→	0 1 1 0
1 1 0 1	0 0 1 0	→	0 1 0 0
1 1 1 0	0 0 0 1	→	0 0 1 0
1 1 1 1	0 0 0 0	→	0 0 0 0