Untitled circuit_Aug 20, 2020 1:39 PM

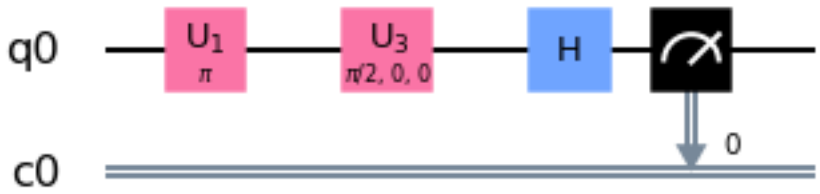
August 25, 2020

[1]:
```
from qiskit import QuantumCircuit, ClassicalRegister, QuantumRegister, execute,␣
 ↪Aer, IBMQ
from qiskit.compiler import transpile, assemble
from qiskit.tools.jupyter import *
from qiskit.visualization import *
from math import pi

#provider = IBMQ.load_account()

backend = Aer.get_backend('qasm_simulator')
n=1
seria=2048

A1 = 383
A2 = 403
A3 = 208

k1 = (A1+A2+A3)%2
k2 = (A1+A2+A3)%4
k3 = (A1+A2+A3)%8

if k2:
    th0 = pi/k2
else:
    th0 = 0

if k3:
    fi0 = 2*pi/k3
else:
    fi0 = 0
```

[2]:
```
#Pomiar typu X
nx=n
qx=QuantumRegister(nx)
cx=ClassicalRegister(nx)
circuitX=QuantumCircuit(qx,cx)
```

```
if k1:
    circuitX.x(qx[0])
circuitX.u1(fi0,qx[0])
circuitX.u3(th0,0,0,qx[0])
circuitX.h(qx[0])

circuitX.measure(qx[0],cx[0])
circuitX.draw(output='mpl')
```

[2]:

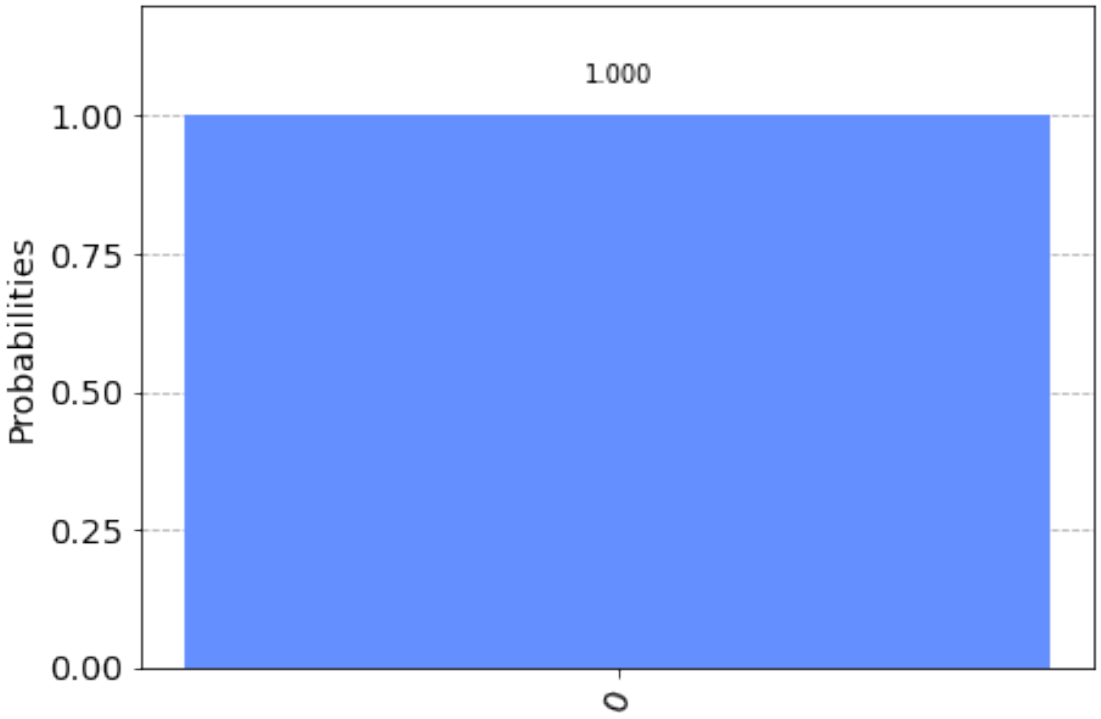


[3]:
```
job_simX=execute(circuitX,backend,shots=seria)
sim_resultX=job_simX.result()
print(sim_resultX.get_counts(circuitX))
```

```
{'0': 2048}
```

[4]:
```
plot_histogram(sim_resultX.get_counts(circuitX))
```

[4]:



[5]:
```
#Pomiar typu Y
ny=n
qy=QuantumRegister(ny)
cy=ClassicalRegister(ny)
circuitY=QuantumCircuit(qy,cy)

if k1:
    circuitY.x(qy[0])
circuitY.u1(fi0,qy[0])
circuitY.u3(th0,0,0,qy[0])
circuitY.sdg(qy[0])
circuitY.h(qy[0])

circuitY.measure(qy[0],cy[0])
circuitY.draw(output='mpl')
```

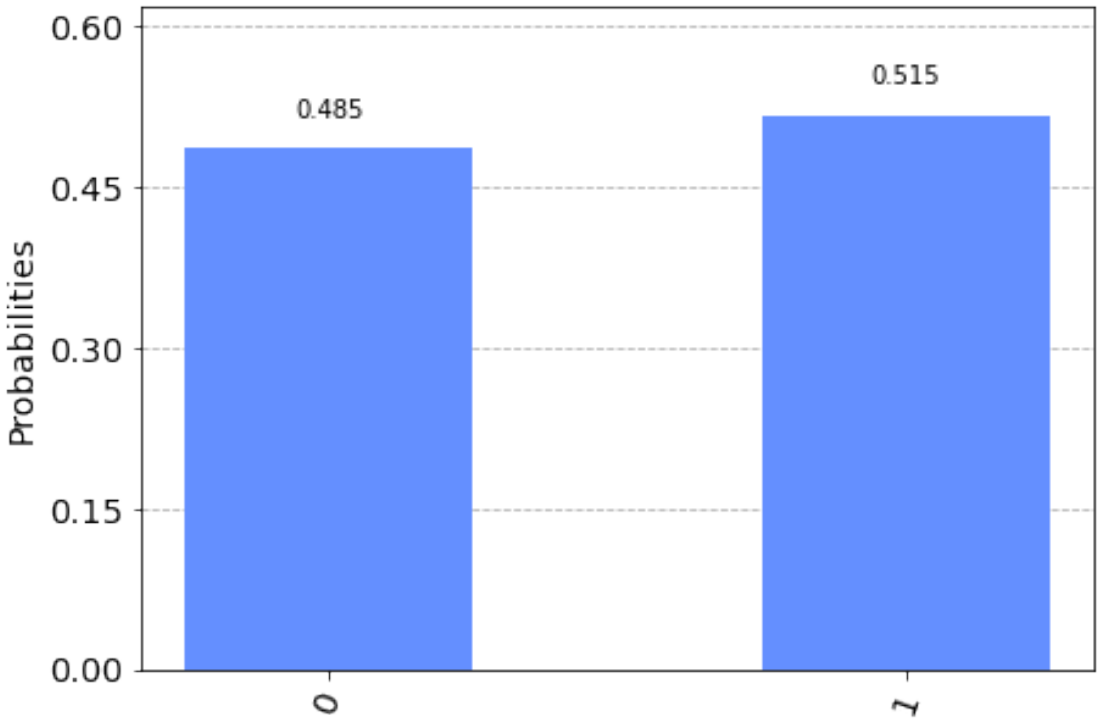
[5]:

[6]:
```
job_simY=execute(circuitY,backend,shots=seria)
sim_resultY=job_simY.result()
print(sim_resultY.get_counts(circuitY))
```

```
{'0': 993, '1': 1055}
```

[7]:
```
plot_histogram(sim_resultY.get_counts(circuitY))
```

[7]:



[8]:
```
#Pomiar typu Z
nz=n
qz=QuantumRegister(nz)
```

```
cz=ClassicalRegister(nz)
circuitZ=QuantumCircuit(qz,cz)

if k1:
    circuitZ.x(qz[0])
circuitZ.u1(fi0,qz[0])
circuitZ.u3(th0,0,0,qz[0])

circuitZ.measure(qz[0],cz[0])
circuitZ.draw(output='mpl')
```

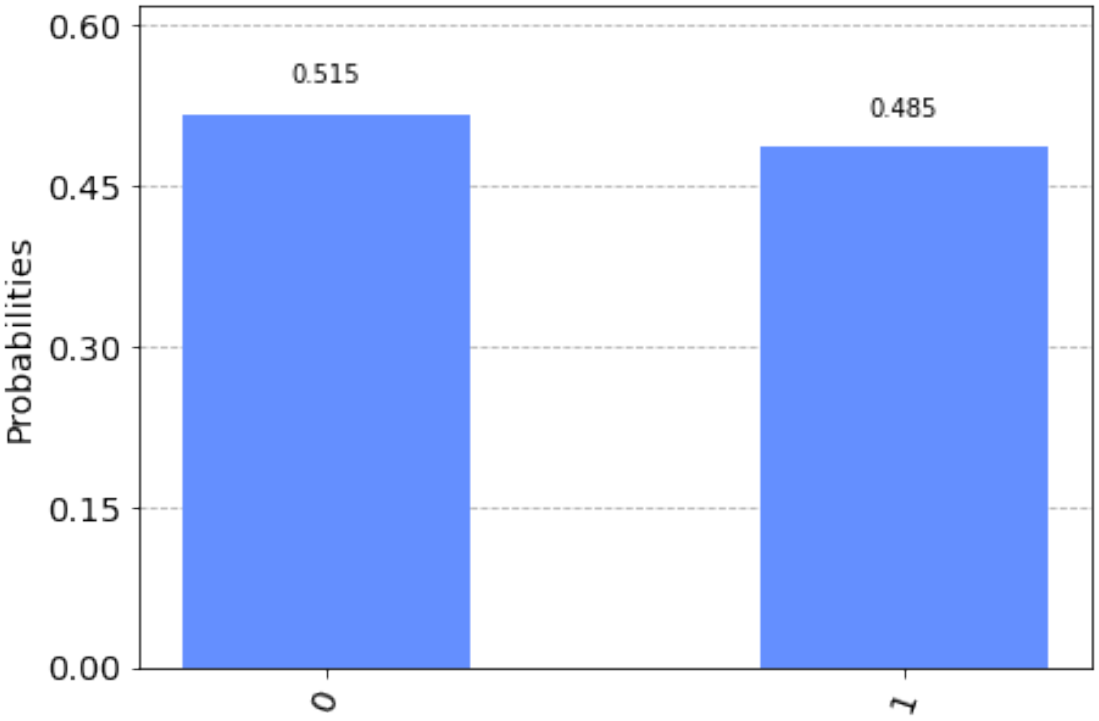
[8]:

[9]:
```
job_simZ=execute(circuitZ,backend,shots=seria)
sim_resultZ=job_simZ.result()
print(sim_resultZ.get_counts(circuitZ))
```

```
{'0': 1055, '1': 993}
```

[10]:
```
plot_histogram(sim_resultZ.get_counts(circuitZ))
```

[10]:


0.60
0.45
0.30
0.15
0.00
Probabilities
0.515
0.485
0
1