In [1]:

```
from qiskit import QuantumCircuit, assemble, Aer, execute
from qiskit.visualization import plot_histogram
```

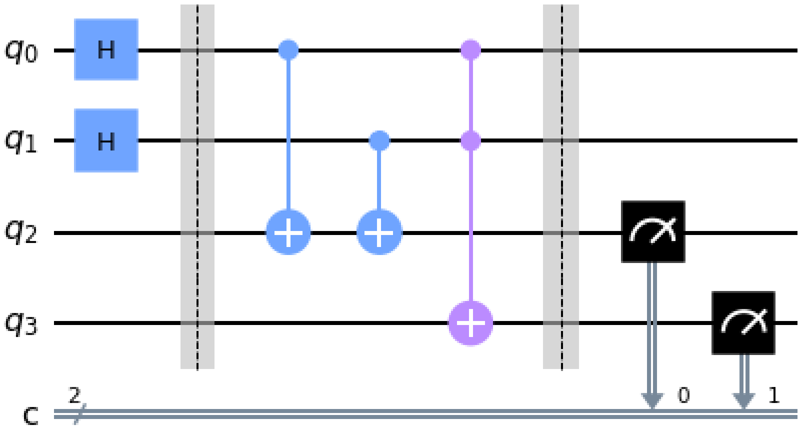
In [ ]:

```
### =-=-=-=-=-= 2-qubit Half Adder =-=-=-=-=-= ###
```

In [5]:

```
circuit = QuantumCircuit(4, 2)

circuit.h([0, 1])
# circuit.x(0)
# circuit.x(1)
circuit.barrier()

circuit.cx(0, 2)
circuit.cx(1, 2)
circuit.ccx(0, 1, 3)
circuit.barrier()

circuit.measure(2, 0)
circuit.measure(3, 1)

circuit.draw('mpl')
```

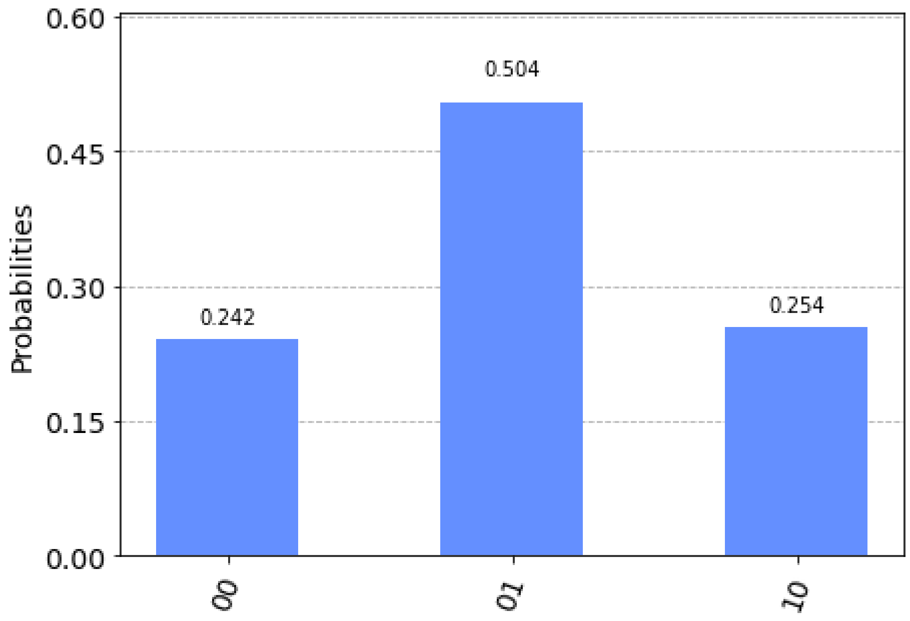
Out[5]:



In [6]:

```
sim = Aer.get_backend('qasm_simulator')
qobj = assemble(circuit, shots=5000)
counts = sim.run(qobj).result().get_counts()
plot_histogram(counts)
```

Out[6]:



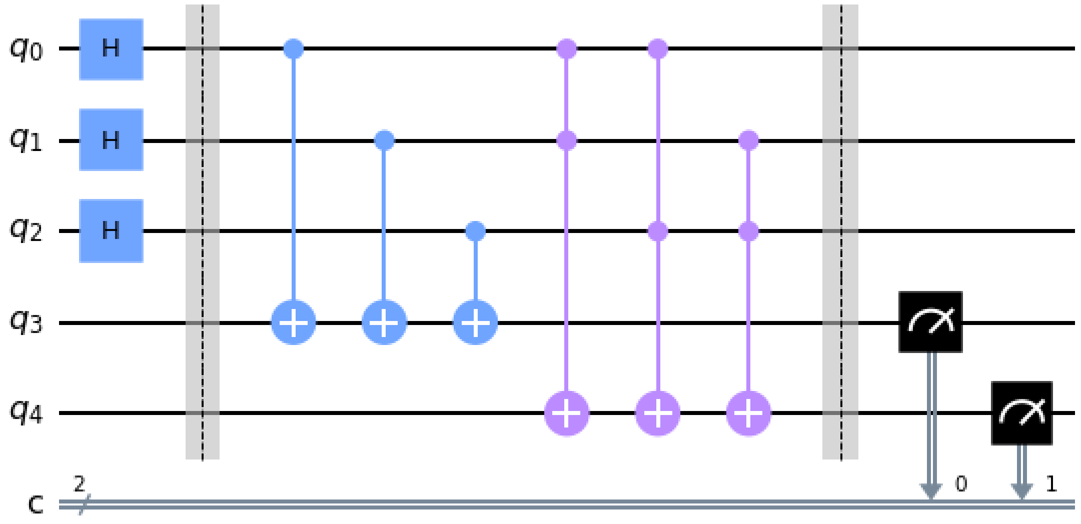
In [ ]:

```
### =-=-=-=-=-= 3-qubit Half Adder =-=-=-=-=-= ###
```

In [16]:

```
circuit = QuantumCircuit(5, 2)

circuit.h([0, 1, 2])
# circuit.x(0)
# circuit.x(1)
# circuit.x(2)
circuit.barrier()

circuit.cx(0,3)
circuit.cx(1,3)
circuit.cx(2,3)
circuit.ccx(0,1,4)
circuit.ccx(0,2,4)
circuit.ccx(1,2,4)
circuit.barrier()

circuit.measure(3,0)
circuit.measure(4,1)

circuit.draw('mpl')
```

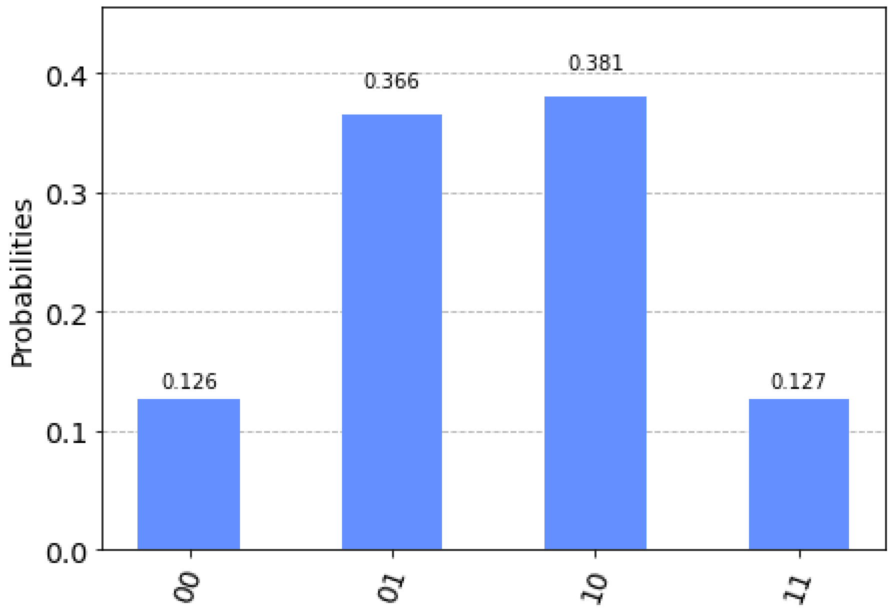
Out[16]:



In [18]:

```
sim = Aer.get_backend('qasm_simulator')
qobj = assemble(circuit, shots=5000)
counts = sim.run(qobj).result().get_counts()
plot_histogram(counts)
```

Out[18]:



In [1]:

```
import qiskit
qiskit.__qiskit_version__
```

Out[1]:

```
{'qiskit-terra': '0.16.4',
 'qiskit-aer': '0.7.5',
 'qiskit-ignis': '0.5.2',
 'qiskit-ibmq-provider': '0.11.1',
 'qiskit-aqua': '0.8.2',
 'qiskit': '0.23.6'}
```