Part 1

Go to https://quantum.ibm.com/

Create an *IBMid* if you don't have one, then login.

Click on *IBM Quantum Composer*.

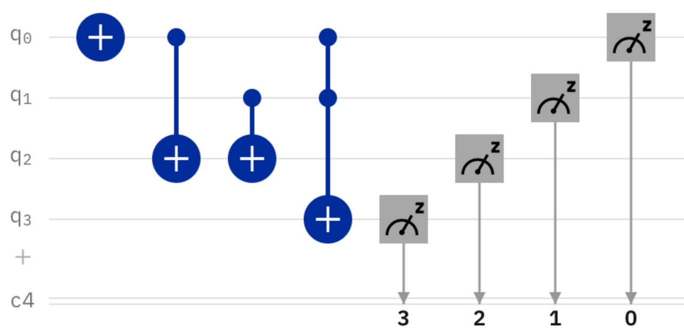
Part 2

In this lab you will use *IBM Quantum Composer* to implement a half-adder circuit.

Half Adder

carry, sum		Input	
q_3	q_2	q_1	q_0
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0

See the videos in this folder to help you build this quantum circuit, run it on a real quantum computer, and check the results.

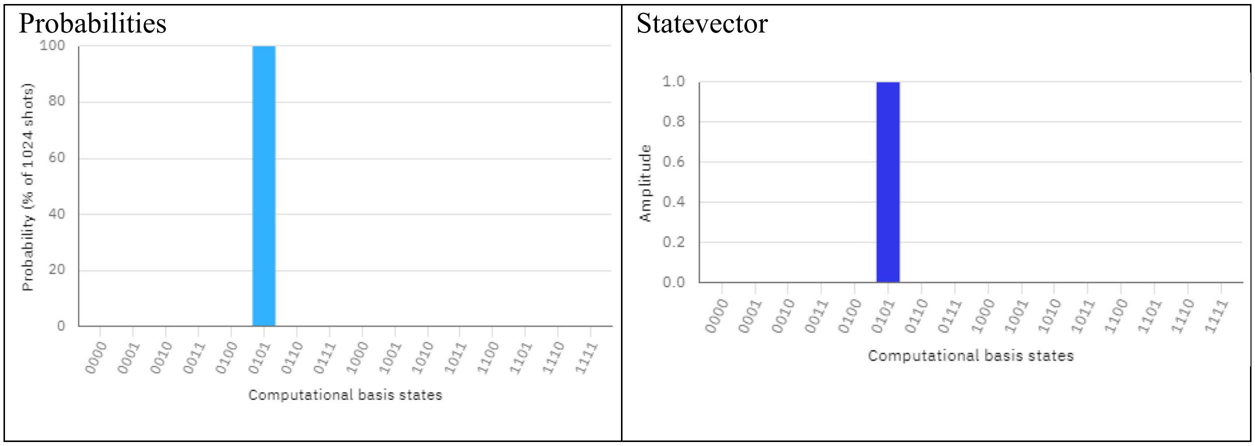


By default, all qubits are initialized to $|0\rangle$.

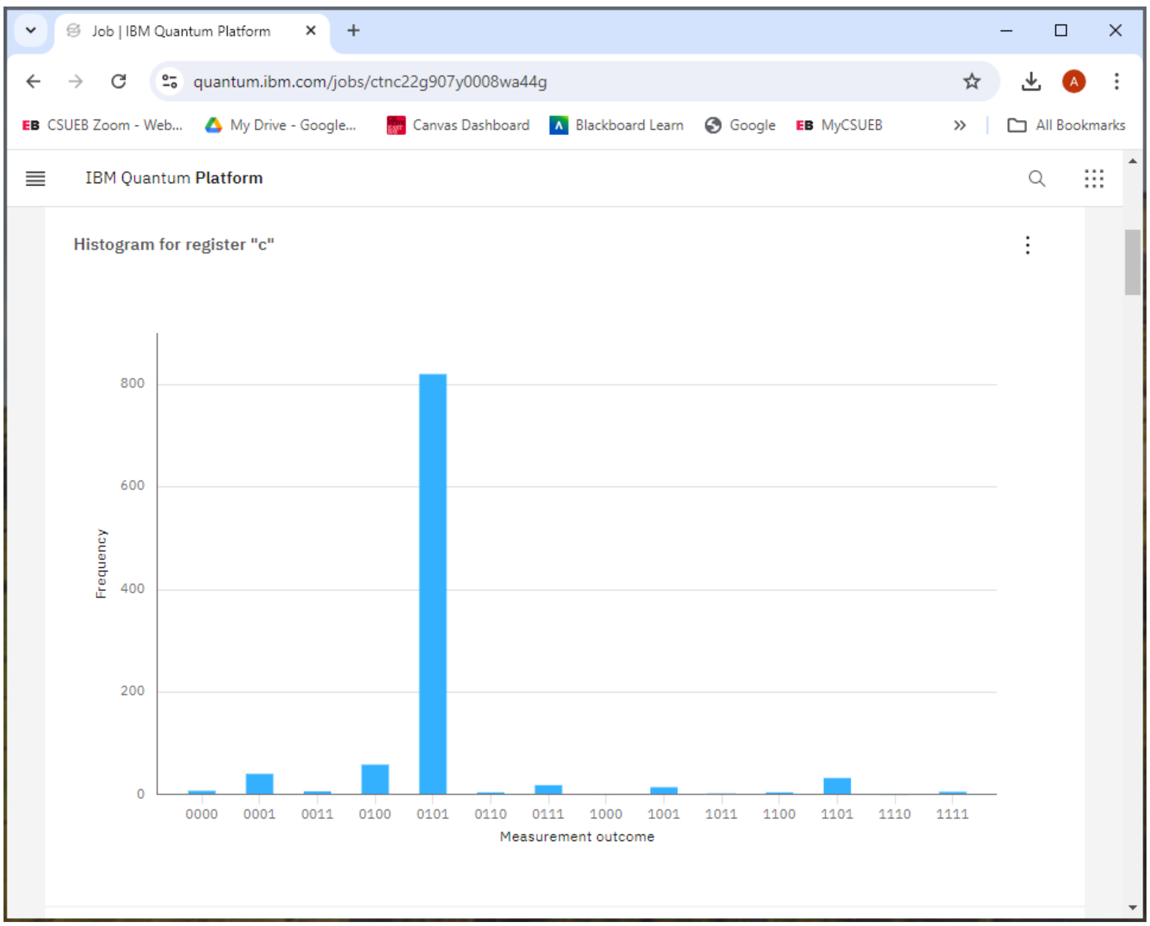
A few things to keep in mind:

- Don't worry if you don't understand how the circuit works.
- The dark blue $\oplus$ icon on the top left is called an X gate. Applying this gate flips the qubit state from $|0\rangle$ to $|1\rangle$, or from $|1\rangle$ to $|0\rangle$.

Examine the Probabilities and the State Vector panes below showing the state 0101. The top qubit is q_0.



Once the job you submitted is completed, you can see the histogram of the result.



Part 3

Convince yourself that the circuit works in all possible cases:

- Add a similar X gate to q_1. The circuit should compute 1+1=10. The state vector should show 1011.
- Delete the X gate in q_0 to compute 1+0=01. The state vector should show 0110.
- Delete the X gate in q_1 to compute 0+0=00. The state vector should show 0000.

Submit screenshots:

- Histogram generated by the quantum computer showing 0101 as the result (the first case).
- Quantum circuit to compute 0+0 (the last case).

FYI:

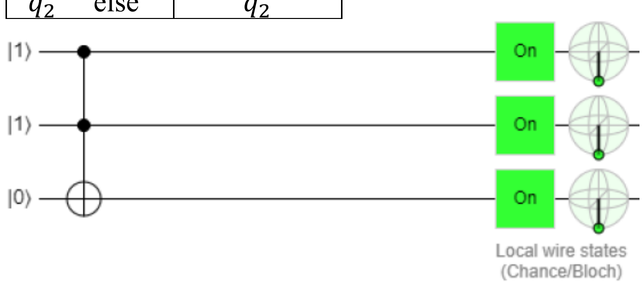
CX (a.k.a. 'conditional NOT', or CNOT) gate

q_1	q_0	output
0	0	0
0	1	1
1	0	1
1	1	0

q_0, q_1 are the control qubit and target qubit, respectively.

CCX gate

q_2	q_1	q_0	output
q_2	1	1	$\bar{q}_2$
q_2	else		q_2



Other Resources: https://learning.quantum.ibm.com/