

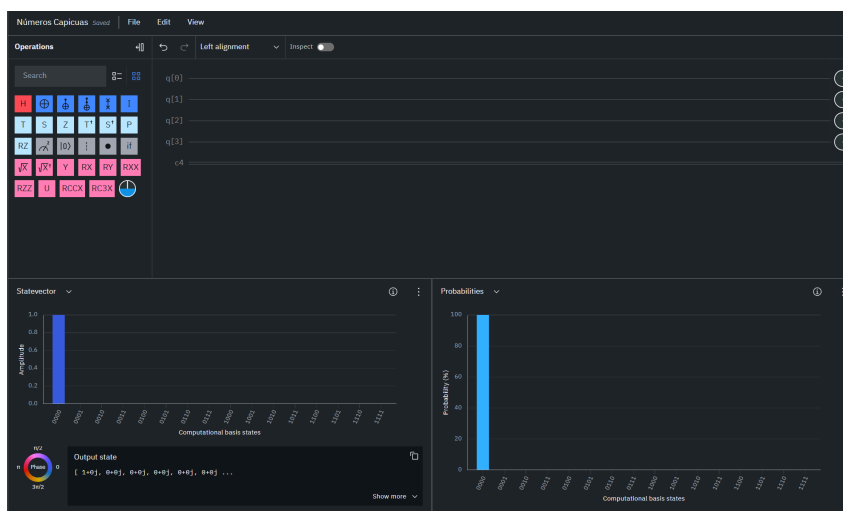
PROGRAMANDO EN LA COMPUTADORA CUÁNTICA DE IBM



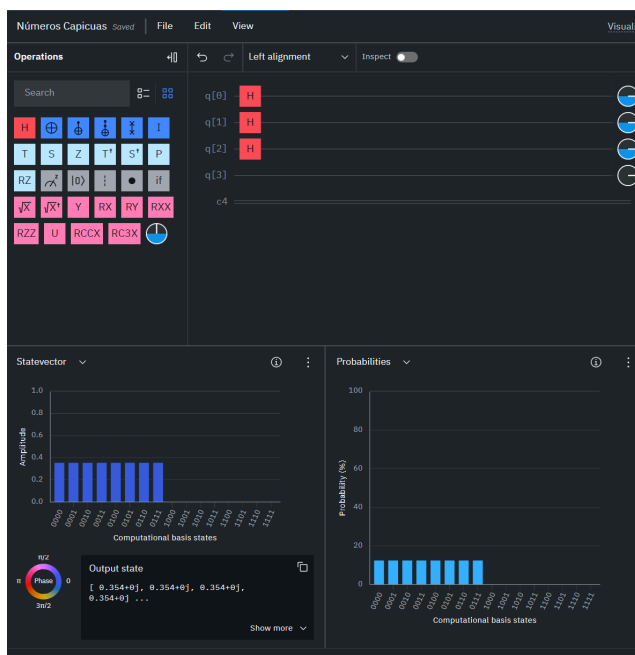
NÚMEROS CAPICÚA

A continuación, vamos a crear un circuito en una computadora cuántica para encontrar números capicúa de 3 bits.

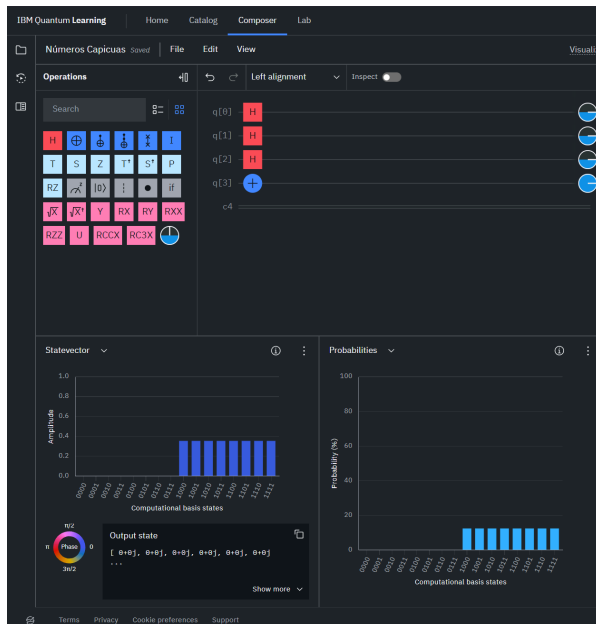
Primero, añadimos 4 cubits, los tres primeros para ingresar los dígitos binarios y el cuarto como cubit auxiliar.



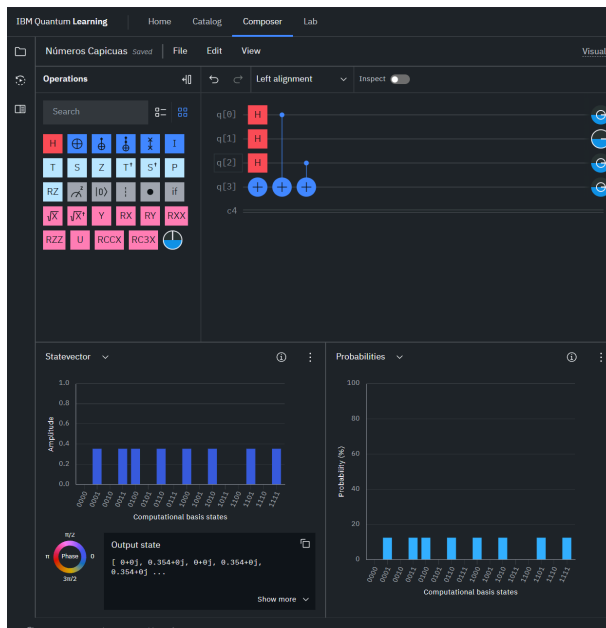
Ahora, superponemos los 3 primeros cubits, para que representen simultáneamente todos los valores entre 000 y 111.



Luego, inicializamos el cubit q3 a 1. El cuarto qubit (q3) actúa como un qubit de control que eventualmente nos ayudará a determinar si la combinación de los primeros tres qubits forma un número capicúa.



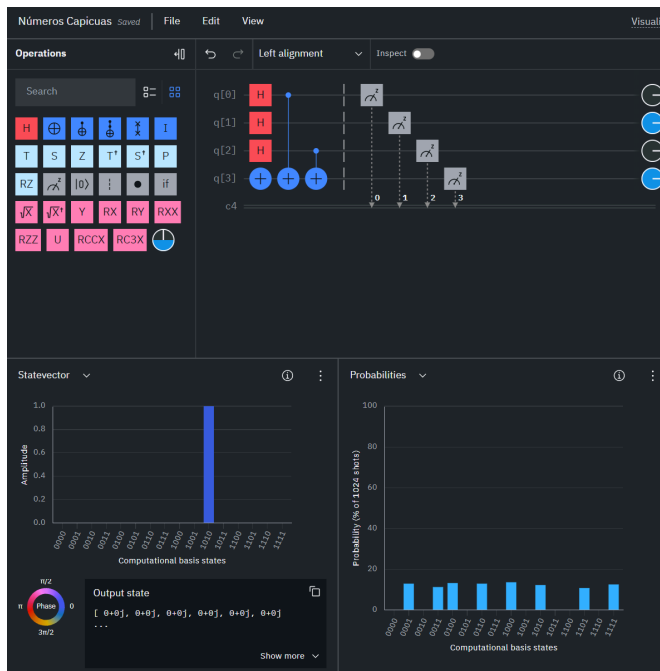
Después, añadimos dos nodos de control, uno para entrelazar el estado del qubit q3 con el q0 y otro para entrelazar los estados de los qubits q3 y q2. De esta manera si q0 o q2 son 1, q3 cambiará al valor 0.



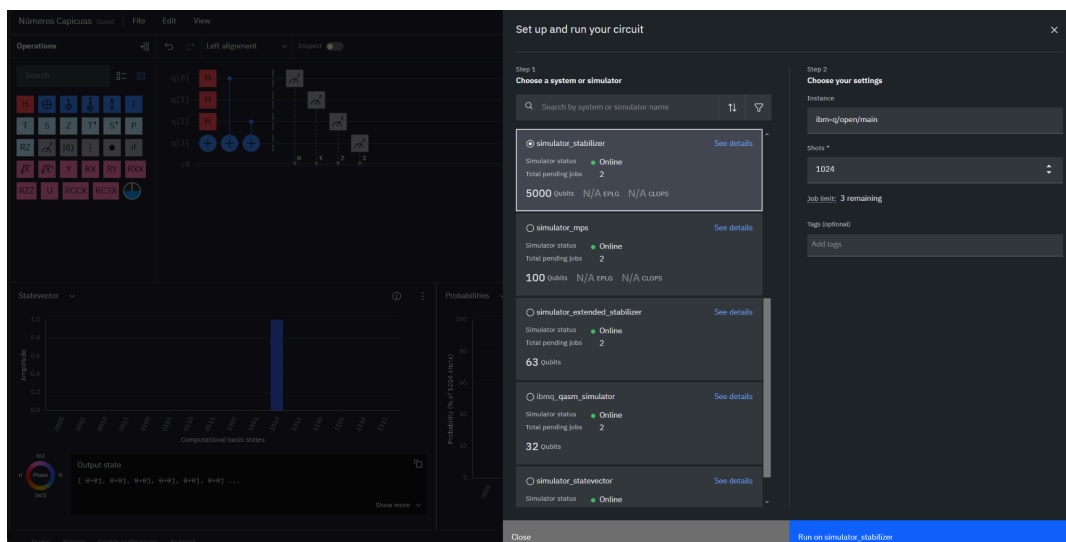
En este punto el problema ya está resuelto, cuando el qubit q3 esté en 1 corresponde a un estado capicúa y cuando esté en 0 corresponderá a estados no capicúas.

Por último, vamos a lanzarlo a una máquina virtual.

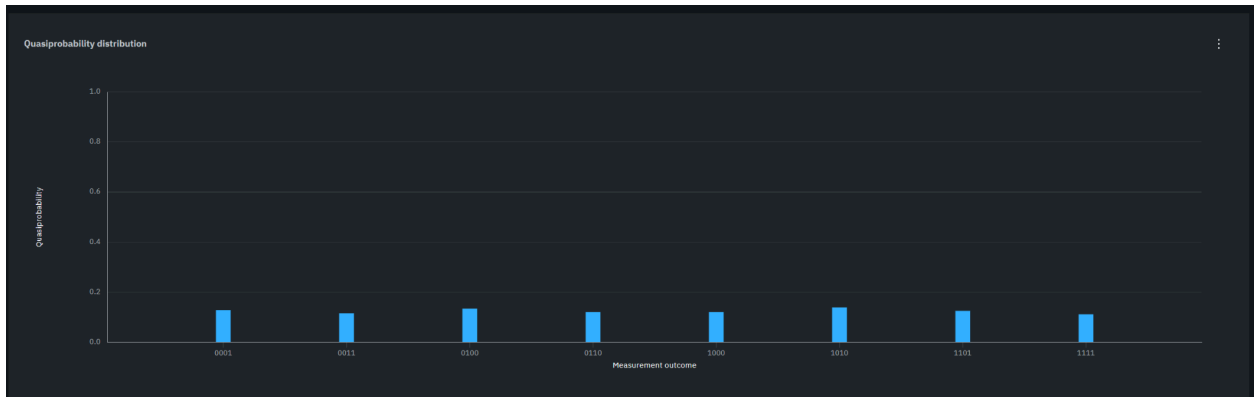
Para ello primero ponemos una barrera para mejor legibilidad y leemos los cubits.



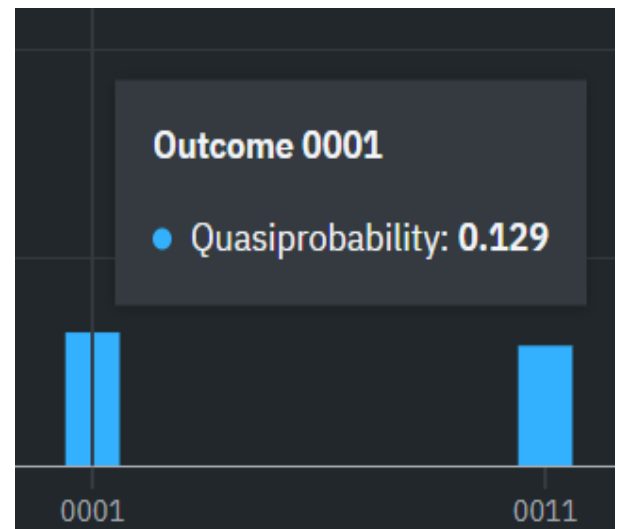
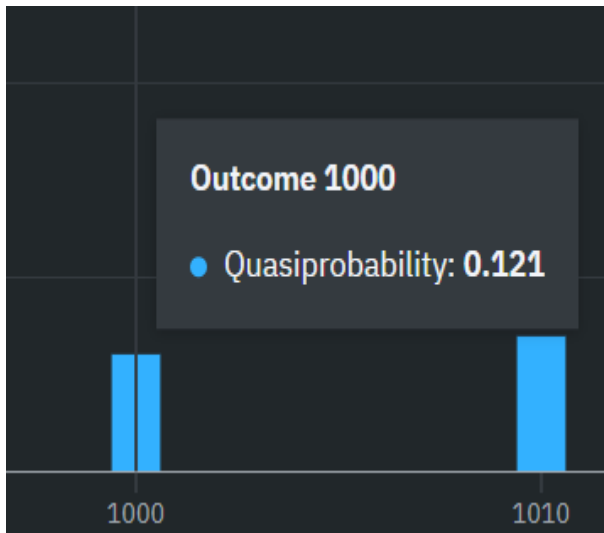
Ahora, lo arrancamos en un simulador y esperamos a que nos de los resultados.



Una vez obtenidos los resultados, nos damos cuenta que efectivamente el circuito funciona correctamente.



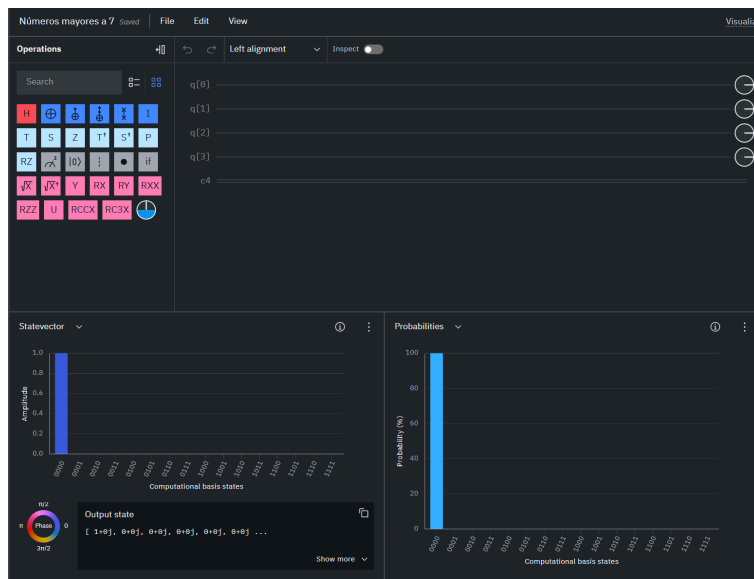
Cuando el primer dígito es 1 corresponde a un número capicúa y cuando es 0 no es un número capicúa.



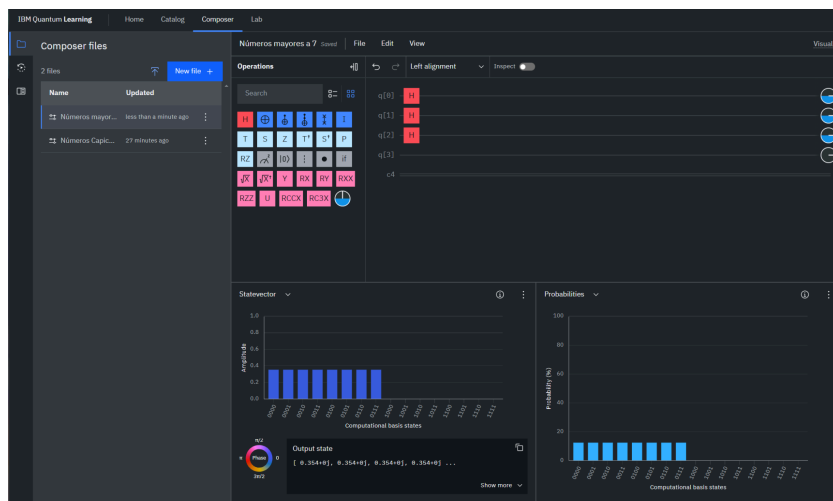
NÚMEROS MAYORES A 7

A continuación, vamos a usar el ejemplo anterior para buscar los números de 4 bits mayores a 7.

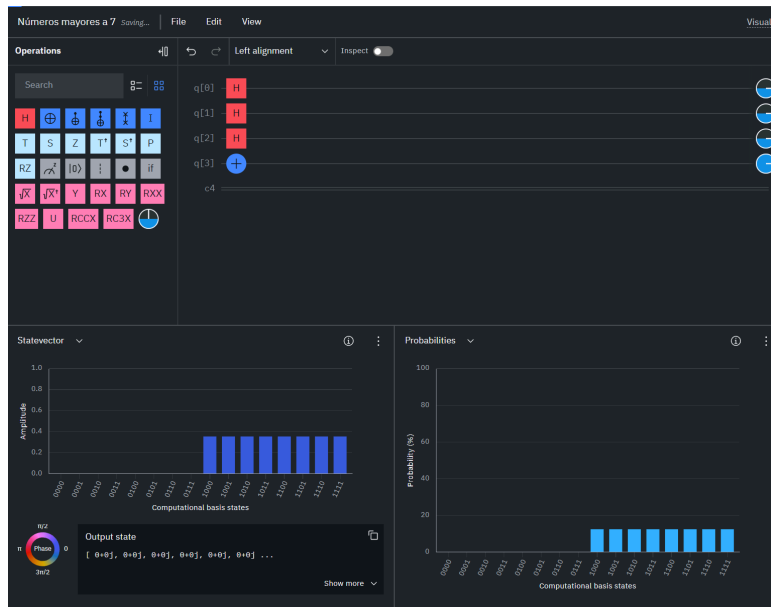
Para ello vamos a necesitar 4 cubits como en el ejemplo anterior.



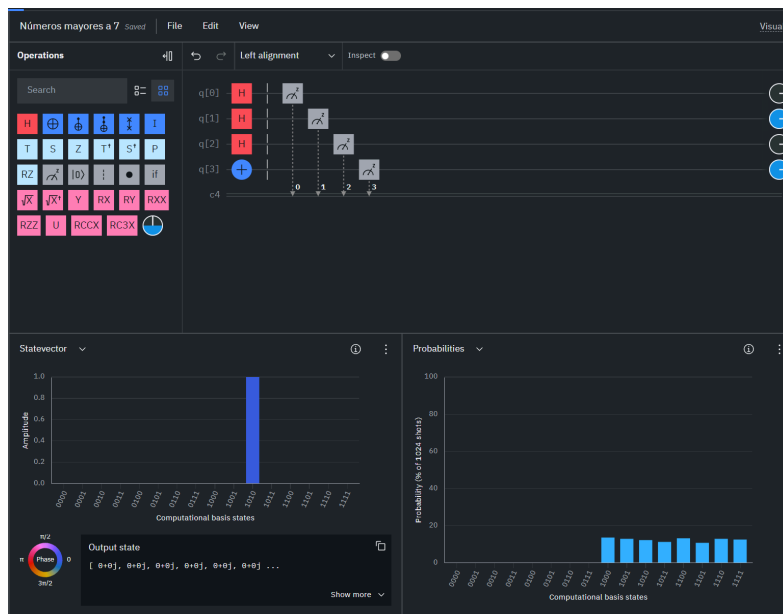
Ahora, vamos a aplicar la compuerta Hadamard a los tres primeros cubits.



Luego, aplicamos una compuerta NOT en q3 para que siempre sea 1, ya que para que el número sea mayor que 7, tiene que ser mayor que 0111, es decir que q3 siempre va a ser 1.



Con esto, el circuito ya estaría terminado, ahora, como en el caso anterior, añadimos una barrera y colocamos los lectores para poder enviárselo a una máquina cuántica virtual.



Quasiprobability distribution

Quasiprobability

Measurement outcome

Outcome 1110
Quasiprobability: 0.118

Measurement outcome	Quasiprobability
1000	0.118
1001	0.118
1010	0.118
1011	0.118
1100	0.118
1101	0.118
1110	0.118
1111	0.118