



# DESCIFRAR RSA CON COMPUTACIÓN CUÁNTICA

Cardenas Huaman Fabricio Yared  
Espirilla Machaca Joseph Ode  
Huaman Quispe Andy Marcelo  
Quispe Clemente Saman  
Valencia Naupa Marko Leonel

The background of the slide is a solid black field. On the right side, there is a complex, abstract pattern of glowing lines. These lines are primarily blue and orange, with some white highlights. They form a dense, swirling, and overlapping web of paths that suggest motion and connectivity, resembling a network or a complex data visualization. The lines are more concentrated on the right and fade out towards the left.

01

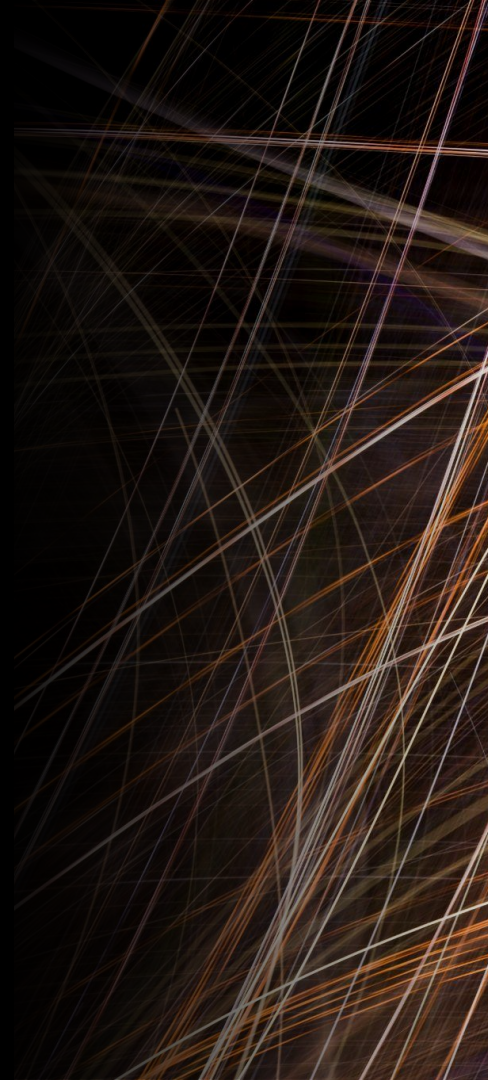
# INTRODUCCION

# INTRODUCCION

Criptografía RSA: Pilar de seguridad en internet basado en factorización de números primos. Amenazado por avances en computación cuántica.

Estructura del documento: Abstract, marco teórico (RSA y computación cuántica), metodología (funcionamiento y descifrado de RSA).

Desarrollo: Algoritmo del proyecto, módulos utilizados, resultados obtenidos. Conclusiones y bibliografía.





02

MARCO TEORICO



# CRIPTOGRAFÍA RSA

RSA: Criptografía de Clave Pública

Basado en factorización de números grandes  
Generación de claves:

Primos  $p$  y  $q \rightarrow n = pq$

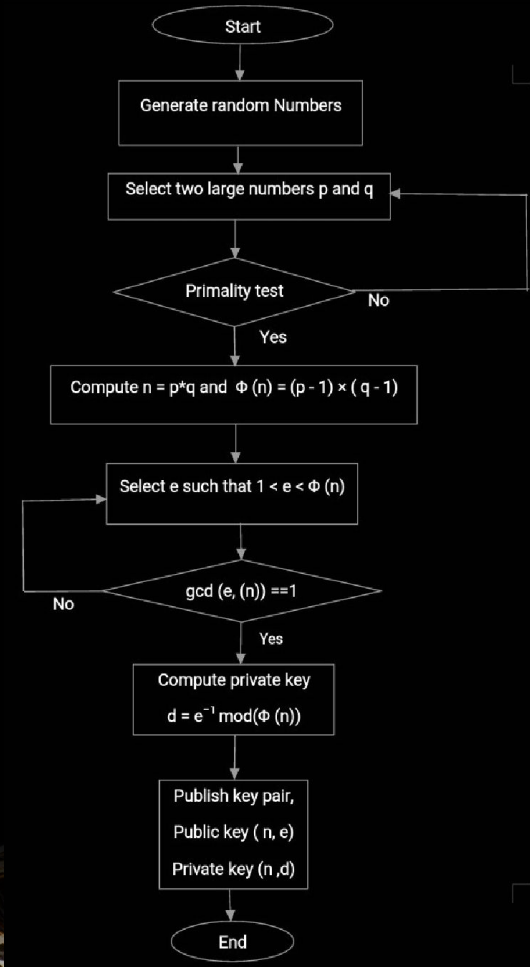
Exponente de cifrado  $e$  coprimo con  $(p-1)(q-1)$

$d: ed \equiv 1 \pmod{(p-1)(q-1)}$

Clave pública:  $(e,n)$ , Clave privada:  $(d,n)$

Cifrado:  $C = Me \pmod n$

Descifrado:  $M = Cd \pmod n$





# COMPUTACIÓN CUÁNTICA

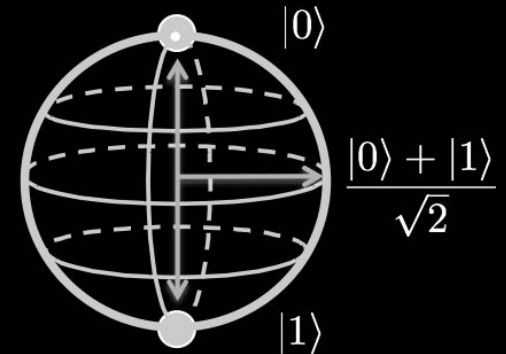
- Usa principios de mecánica cuántica
  - Superposición y entrelazamiento
- Procesa información más eficientemente
- Desarrollo ligado a teoría cuántica de información
- Desafíos prácticos significativos
  - Opiniones divididas sobre viabilidad
- Su desarrollo impulsa el conocimiento sobre:
  - Sistemas cuánticos
  - Comportamiento de la información



# QUBITS

Qubits: Unidad Básica de Información Cuántica

- Estados:  $|0\rangle$ ,  $|1\rangle$ , y superposición de ambos
- Pueden representar múltiples valores simultáneamente
- Permiten procesamiento paralelo masivo
- Pierden superposición al ser medidos
- Tipos: superconductores, iones atrapados, fotones, etc.
- Desafío: mantener coherencia cuántica



**Qubit**

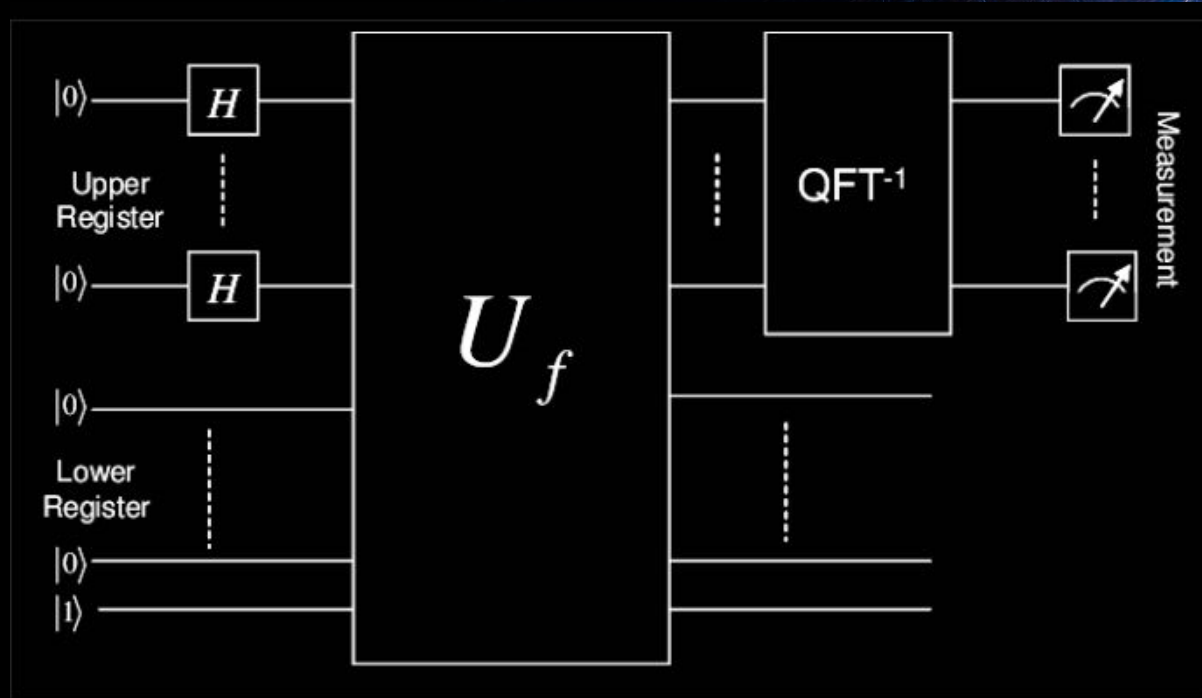


# SHOR (SHOR'S ALGORITHM)

- Desarrollado por Peter Shor en 1994
- Factoriza números enteros grandes eficientemente
- Tiempo polinómico vs exponencial en algoritmos clásicos
- Implicaciones críticas para la criptografía:
  - Puede romper RSA
  - Factoriza  $n$  para obtener claves privadas
- Amenaza significativa para sistemas de seguridad actuales
- Demuestra superioridad cuántica en tareas específicas
- Impulsa desarrollo de criptografía post-cuántica



# SHOR (SHOR'S ALGORITHM)



03

# METODOLOGIA

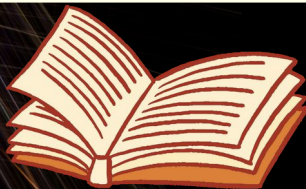




# METODOLOGIA

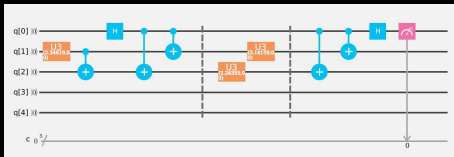
## Investigación/ Revisión teórica

- \_ Fundamentos del algoritmo de Shor
- \_ Principios de la Transformada de Fourier Cuántica (QFT)
- \_ Bases del sistema de encriptación RSA



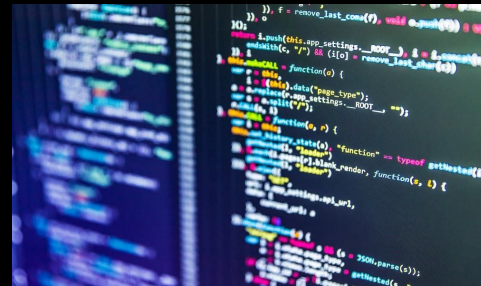
## Diseño e implementación del circuito cuántico

Comenzamos con el diseño del circuito, con los qubits necesarios y los módulos a funcionar



## Respaldo RSA

Modificamos el anterior proyecto del cifrado RSA con cuantica

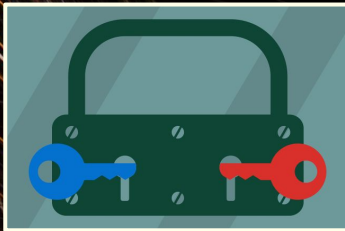


# METODOLOGIA

## Funcionamiento del algoritmo RSA

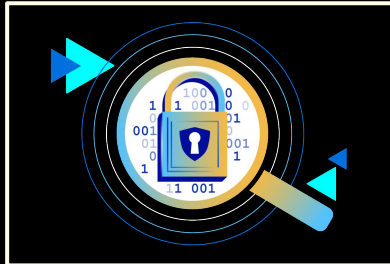
### GENERACIÓN DE CLAVES

Se generan dos números primos grandes y se utilizan para producir dos claves: una pública y una privada.



### CIFRADO

Para enviar un mensaje seguro, el remitente cifra el mensaje utilizando la clave pública del destinatario.



### Descifrado

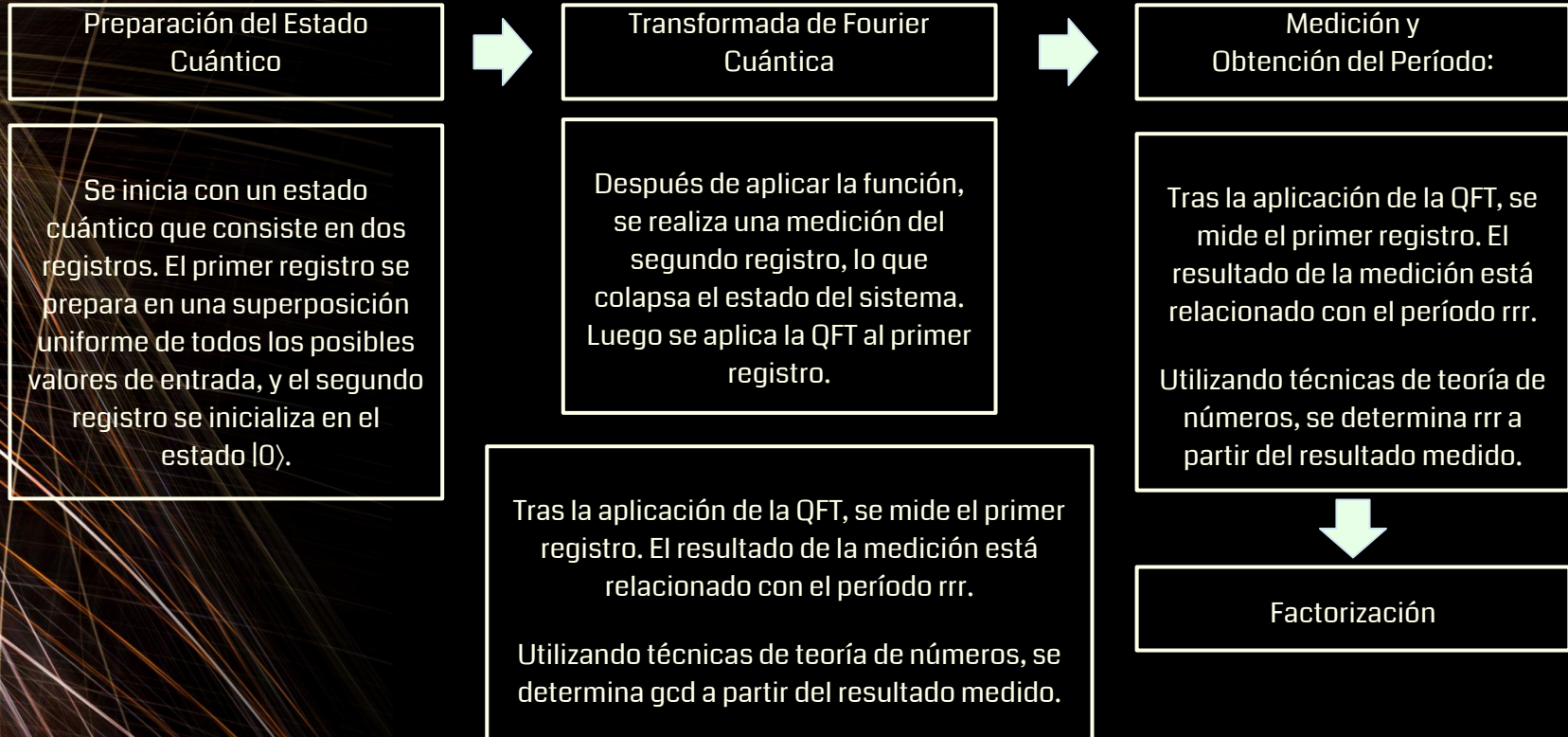
El destinatario utiliza su clave privada para descifrar el texto cifrado y recuperar el mensaje original.





# METODOLOGIA

Shor puede descomponer un número entero grande en sus factores primos en tiempo polinómico, algo que es intratable para los algoritmos clásicos.



04

DESARROLLO

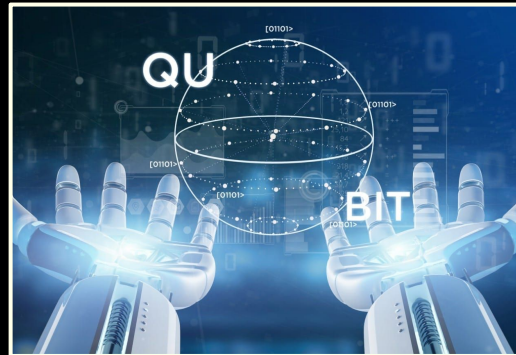




# DESARROLLO

## Circuito Cuántico

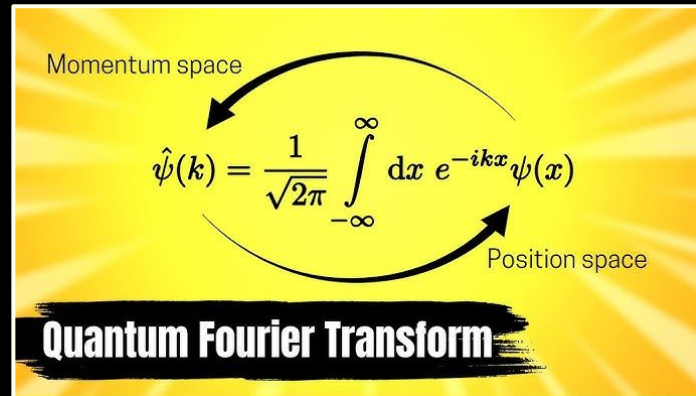
- Sea  $N$  el segundo elemento de la tupla de la clave pública  $T = 2(\log_2(N) + 1)$
- Se crea un circuito cuántico con  $T$  qubits donde se implementa el módulo “a” con respecto a  $N$ .
- Se aplica una compuerta  $X$  en los qubits pertinentes.
- Se aplican compuertas de control  $CX$  entre varios qubits para establecer el cálculo modular.
- Finalmente, se crean compuertas de control  $CCX$  (Toffoli) para los qubits de medición los cuales son la mitad.
- Se convierte en una compuerta cuántica personalizada y se nombra " $a^x \bmod N$ ".



# DESARROLLO

## Transformada de Fourier Cuántica (QFT)

- Se crea un circuito cuántico con n qubits donde se implementa la QFT.
- Se aplican compuertas Hadamard H y compuertas de fase controlada CP.
- Se intercambian los qubits (swap) para completar la QFT.
- Se convierte en una compuerta cuántica personalizada y se nombra "QFT" seguido por el número de qubits.



The diagram illustrates the Quantum Fourier Transform (QFT) as a transformation between two spaces. It features a yellow background with a sunburst pattern. At the top left, the text "Momentum space" is written. At the bottom right, the text "Position space" is written. In the center, the mathematical formula for the QFT is displayed: 
$$\hat{\psi}(k) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} dx e^{-ikx} \psi(x)$$
 Two curved arrows form a cycle around the equation: one arrow points from "Position space" up to "Momentum space", and the other points from "Momentum space" down to "Position space". At the bottom of the diagram, the text "Quantum Fourier Transform" is written in a bold, black, brush-stroke style font.

Quantum Fourier Transform



# DESARROLLO

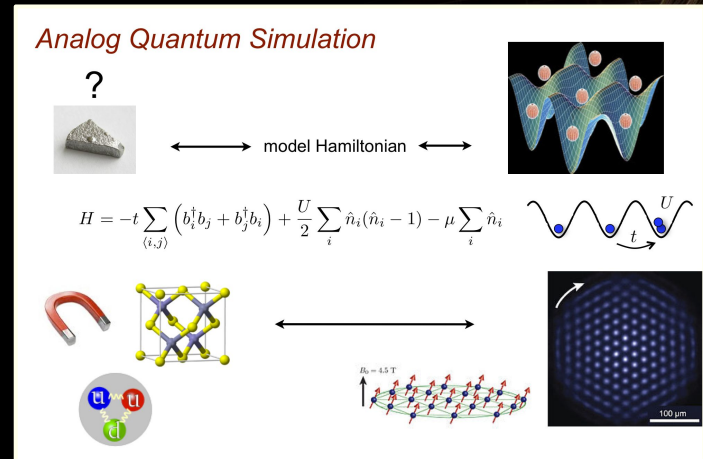
## Construcción del Circuito

- Se crea un circuito cuántico con  $T$  qubits con la mitad para las mediciones.
- Se aplican compuertas Hadamard  $H$  a los primeros  $T/2$  qubits.
- Se añade la compuerta " $a^x \bmod N$ " al circuito.
- Se miden los qubits  $T/2$  a  $T$  en los bits clásicos.
- Se aplica una barrera para separar las operaciones.
- Se añade la compuerta QFT de  $T/2$  qubits.
- Se miden los primeros  $T/2$  qubits en los bits clásicos.
- Se dibuja el circuito.

# DESARROLLO

## Simulación del Circuito

- Se selecciona un backend de simulación cuántica.
- Se compila y transpila el circuito para el backend seleccionado.
- Se ejecuta el circuito y se obtienen los resultados.
- Se visualizan los resultados mediante un histograma.



# DESARROLLO

## Verificación

- Se identifican las raíces cuadradas (4 en este caso) y se calcula el máximo común divisor (mcd) para obtener los factores primos.
- Los factores primos se calculan usando el mcd de  $4-1$  y  $4+1$  con respecto a 15, obteniendo 3 y 5.

## Código RSA

- Se generan números primos aleatorios de un número determinado de bits.
- Se asegura que los números primos  $p$  y  $q$  sean diferentes.
- Se calculan  $n$  y  $\phi$  para los números primos.
- Se elige un exponente público común  $e=65537$  y se calcula el inverso modular  $d$ .
- Se generan y retornan las claves pública y privada.



**METODO DE  
CIFRADO  
(RSA)**



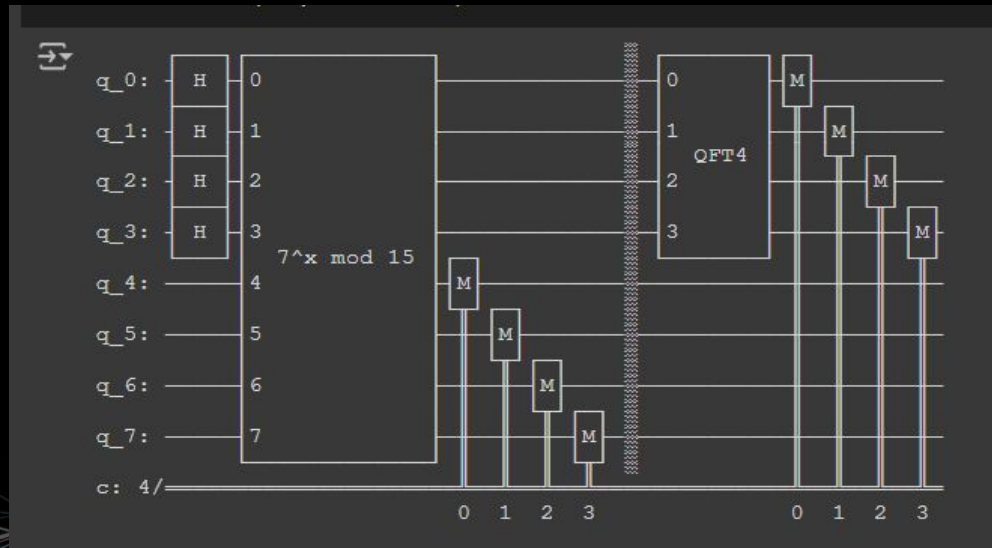
05

RESULTADOS



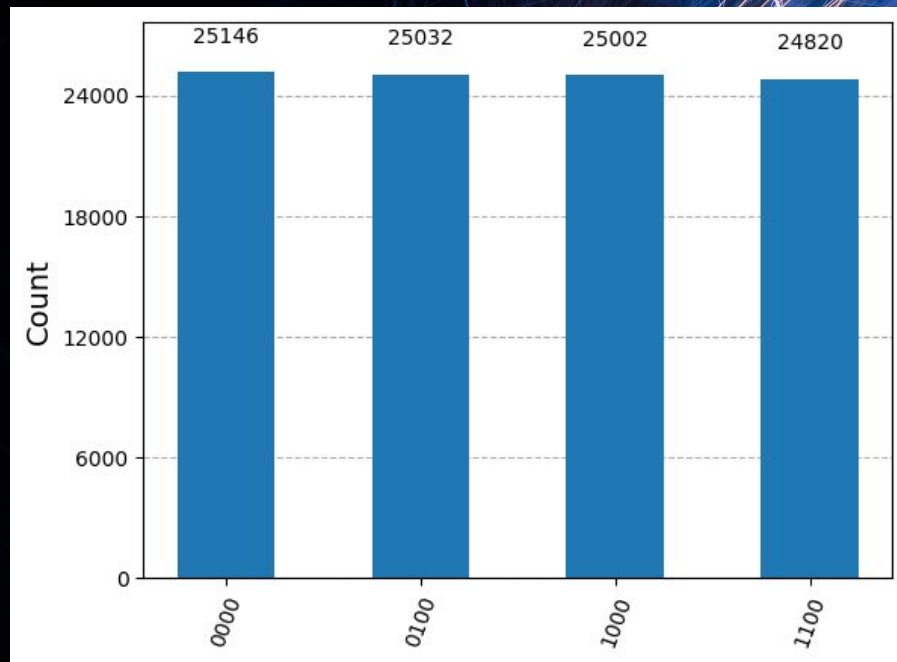
# Circuito Cuántico

El circuito cuántico implementa el cálculo de  $7^x \bmod 15$  y realiza una QFT para obtener los resultados de medición. Esto se simula en un backend cuántico.



# Simulación

El histograma muestra los resultados de la simulación del circuito cuántico descrito anteriormente. Cada barra representa la cantidad de veces que se midió un estado específico de los qubits después de ejecutar el circuito.





**Factores Primos:** A partir de el minimo comun divisor, el cual es 4, se obtienen los factores primos de 15 como 3 y 5 utilizando el mcd.

#### ▼ Verificacion

```
[ ] 1 # con esto sabemos que 4 es raiz cuadrada ya que el mcd de 4 8 12 es 4
    2 # sabiendo esto podemos calcular los factores primos de la siguiente manera:
    3
    4 import math
    5
    6 primer_factor = math.gcd(4-1, 15)
    7 segundo_factor = math.gcd(4+1, 15)
    8 print("factores primos:", primer_factor, segundo_factor)
```

↗ factores primos: 3 5

Código RSA:

Clave publica: (65537, 691823369496328420171130732762179039)

Clave privada: (686587481505267202770305664470608681, 691823369496328420171130732762179039)

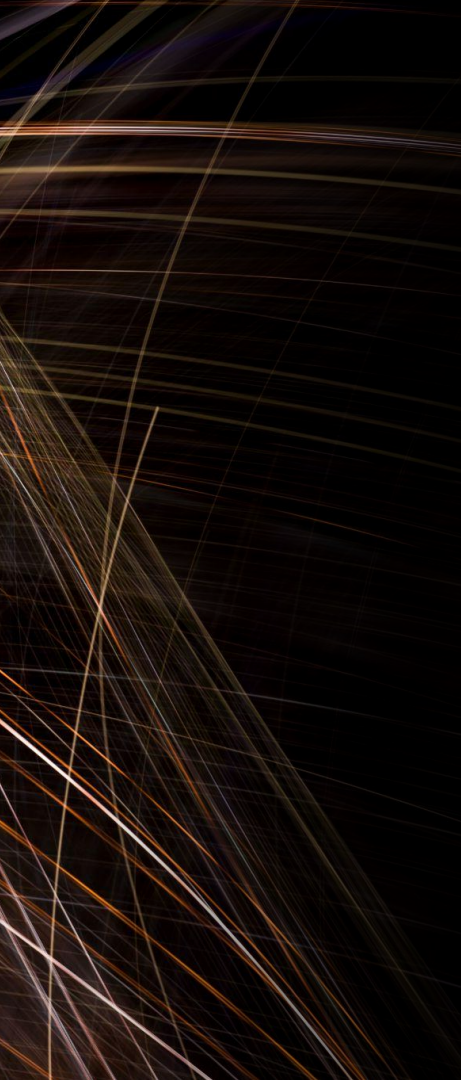
Ciphertext: 400835043659274251263459860950663396

Desencriptar mensaje: Encriptar RSA

06

# CONCLUSIONES



- 
- El circuito cuántico implementado es capaz de calcular  $7^x \bmod 15$  y utilizar la QFT para obtener resultados útiles en el algoritmo de Shor, que se puede usar para la factorización de números enteros.
  - La simulación cuántica confirma la precisión del circuito diseñado.
  - Usando el método descrito, se pueden calcular los factores primos de un número compuesto como 15, lo cual es fundamental para el algoritmo de Shor y otros algoritmos cuánticos de factorización.
  - La generación de claves RSA es correcta y el cifrado/descifrado funciona como se espera, demostrando la implementación efectiva del algoritmo RSA.
  - La elección de claves pequeñas en el ejemplo muestra el funcionamiento básico del RSA, aunque en la práctica se usan claves mucho más grandes para mayor seguridad.



GRACIAS

