Universidad de La Laguna

Circuitos Cuánticos

Microcredencial en Introducción a la Computación Cuántica

Daniel Escanez-Exposito

Jorge Garcia-Diaz

Escuela Superior de Ingeniería y Tecnología
Grupo de Investigación en Criptología - CryptULL

La Laguna, 2025

Universidad de La Laguna

Circuitos Cuánticos

Microcredencial en Introducción a la Computación Cuántica

Daniel Escanez-Exposito
Correo electrónico - jescanez@ull.edu.es

Jorge Garcia-Diaz
Correo electrónico - jgarcidi@ull.edu.es

Escuela Superior de Ingeniería y Tecnología
Grupo de Investigación en Criptología - CryptULL

Microcredencial en Criptografía e Información Cuántica

La Laguna, 2025

Índice general

2

Circuitos Cuánticos

Uno de los enfoques principales para la implementación de algoritmos clásicos a nivel de bits son los circuitos lógicos. Estos están compuestos por puertas lógicas que transforman las señales iniciales, que representan la instancia de un problema, para generar su solución. Se suelen utilizar diagramas lógicos que expresan la entrada del circuito y cómo esta es modificada por las puertas que lo componen para obtener la salida del algoritmo. En la Fig. 2.1 se expone la representación gráfica de algunas puertas lógicas ampliamente utilizadas.

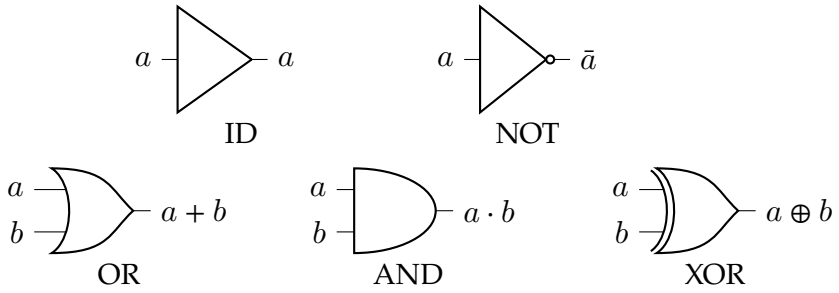


Figura 2.1: *Ejemplos de diagramas lógicos de algunas puertas*

Un ejemplo de ello es el semisumador (véase Fig. 2.2): un circuito capaz de realizar la suma de dos dígitos binarios, considerando también el acarreo de salida generado (que es un AND entre los dos valores de la entrada).

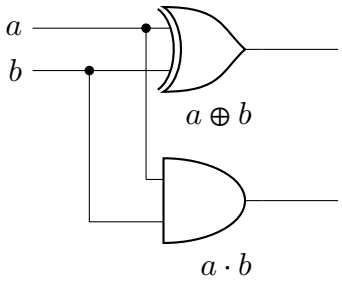
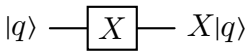


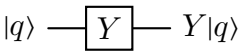
Figura 2.2: *Ejemplo de diagrama lógico de semisumador*

2.1. Representación gráfica

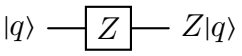
De manera análoga, es usual describir los algoritmos cuánticos con diagramas que describen la evolución del estado de los cúbits a medida que se ejecutan las puertas sobre ellos. Primeramente, será necesario definir cómo se representan nuestras unidades de información. Los cúbits, son representados como líneas horizontales sobre las que se sitúan las operaciones que se les practica. La lectura del circuito debe realizarse de izquierda a derecha, entendiendo que este es el sentido en el que transcurre el cómputo. Es similar al funcionamiento de una línea del tiempo, donde las puertas que se encuentran alineadas verticalmente se ejecutan en paralelo. En la Fig. 2.3 se exponen los diagramas de algunas de las operaciones más utilizadas. Estas son atravesadas por una línea horizontal, que representa el cúbit sobre el que se aplica. En la parte izquierda se observa el estado del cúbit cómo entrada, y en la izquierda el resultado de la misma.



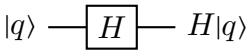
(a) *Puerta X o NOT*



(b) *Puerta Y*

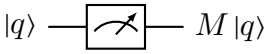


(c) *Puerta Z*

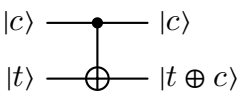


(d) *Puerta H o Hadamard*

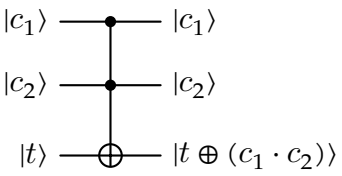
(e) *Puerta genérica U*



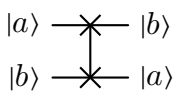
(f) *Instrucción de Medición M*



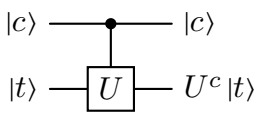
(g) *Puerta $CNOT$*



(h) *Puerta CCX o Toffoli*



(i) *Puerta $SWAP$*



(j) *Puerta genérica CU*

Figura 2.3: *Representación gráfica de las operaciones cuánticas usuales*

De esta manera, se pueden componer estas operaciones para lograr circuitos más complejos que resuelvan un determinado problema. En la Fig. 2.4, se muestra un ejemplo sencillo de circuito cuántico sobre el estado producto de dos cúbits $|q_1 q_2\rangle = |00\rangle$.

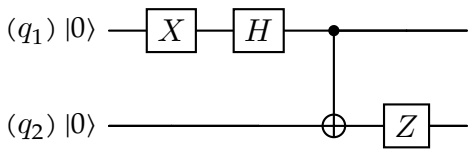


Figura 2.4: *Ejemplo de circuito cuántico*

Se puede observar como los dos cúbits inicialmente comienzan en el estado $|0\rangle$. Si no

se indica nada en el circuito, se asume que los cúbits comienzan en este estado inicial. Tras esto, se ejecuta una puerta X sobre el primero. Después, sobre el mismo cúbit se aplica una H. A continuación, se practica una puerta $CNOT$ utilizando el primer cúbit como control y el segundo como objetivo. Por último, una puerta Z es aplicada sobre el segundo cúbit.

2.2. Evaluación del circuito

Una vez se ha considerado la manera de representar el algoritmo, es conveniente entender cómo se ejecuta. Para ello, se pueden utilizar múltiples enfoques. En esta sección se verán tres maneras equivalentes para realizar el cálculo del vector de estado resultante tras aplicar un determinado circuito cuántico.

2.2.1. Notación matricial

Por un lado, sabiendo que las puertas pueden ser expresadas como matrices, y los cúbits como vectores, se puede realizar una traducción directa entre lo que se está representado gráficamente y el esquema matricial. Es posible describir el desarrollo matricial del circuito de la Fig. 2.4, paso a paso. Para las dos primeras puertas, el cálculo es realmente sencillo. Solo hay que calcular el resultado de aplicar un puerta X al estado del primer cúbit en estado $|0\rangle$. Después de esto, al resultado se le puede aplicar la puerta H.

$$\overbrace{\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}}^{X} \underbrace{\begin{pmatrix} 1 \\ 0 \end{pmatrix}}_{|0\rangle} = \underbrace{\begin{pmatrix} 0 \\ 1 \end{pmatrix}}_{|1\rangle} \tag{2.1}$$

$$\overbrace{\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}}^{H} \underbrace{\begin{pmatrix} 0 \\ 1 \end{pmatrix}}_{|1\rangle} = \underbrace{\frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ -1 \end{pmatrix}}_{|-\rangle} \tag{2.2}$$

Como siguiente puerta es de dos cúbits, es necesario calcular el vector de estado completo. De esta manera, se obtiene la entrada de la puerta CNOT, que es el resultado del producto ($\otimes$) entre el cómputo anterior como primer cúbit, y el otro en estado $|0\rangle$ como segundo:

$$\underbrace{\frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ -1 \end{pmatrix}}_{|-\rangle} \otimes \underbrace{\begin{pmatrix} 1 \\ 0 \end{pmatrix}}_{|0\rangle} = \underbrace{\frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 0 \\ -1 \\ 0 \end{pmatrix}}_{|-0\rangle} \tag{2.3}$$

Una vez calculada la entrada de la puerta CNOT, se procede a ejecutarla sobre el estado previamente calculado:

$$\overbrace{\begin{pmatrix}1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0\end{pmatrix}}^{CNOT}\underbrace{\frac{1}{\sqrt{2}}\begin{pmatrix}1\\0\\-1\\0\end{pmatrix}}_{|-0\rangle}=\underbrace{\frac{1}{\sqrt{2}}\begin{pmatrix}1\\0\\0\\-1\end{pmatrix}}_{|\Phi^-\rangle} \tag{2.4}$$

Por último, se pretende aplicar la Z restante sobre este último resultado. Sin embargo, la puerta es de un sólo cúbit, y el estado que se tiene es uno de dos cúbits. Además, se da la particularidad de que es un estado de entrelazamiento, por lo que no se puede dividir en sus dos componentes para aplicar la operación solo en la deseada. Es por ello que se debe calcular el producto ($\otimes$) entre la puerta identidad I, que se aplicará al primer cúbit (es decir, que no lo cambiará); y la puerta Z que se debe aplicar sobre el segundo cúbit.

$$\overbrace{\begin{pmatrix}1&0\\0&1\end{pmatrix}}^{I}\otimes\overbrace{\begin{pmatrix}1&0\\0&-1\end{pmatrix}}^{Z}=\overbrace{\begin{pmatrix}1&0&0&0\\0&-1&0&0\\0&0&1&0\\0&0&0&-1\end{pmatrix}}^{I\otimes Z} \tag{2.5}$$

Una vez calculada esta matriz, es posible realizar el último cálculo, aplicarlo sobre el vector resultado en la Ec. 2.4:

$$\overbrace{\begin{pmatrix}1&0&0&0\\0&-1&0&0\\0&0&1&0\\0&0&0&-1\end{pmatrix}}^{I\otimes Z}\underbrace{\frac{1}{\sqrt{2}}\begin{pmatrix}1\\0\\0\\-1\end{pmatrix}}_{|\Phi^-\rangle}=\underbrace{\frac{1}{\sqrt{2}}\begin{pmatrix}1\\0\\0\\1\end{pmatrix}}_{|\Phi^+\rangle} \tag{2.6}$$

De esta manera, se concluye que el resultado del circuito es el estado $|\Phi^+\rangle$. Desde el enfoque matricial, también es posible contemplar el sistema como uno de dos cúbits desde el principio. Para ello, es interesante apreciar que el circuito ilustrado en la Fig. 2.4 es equivalente al que se muestra en la Fig. 2.5.

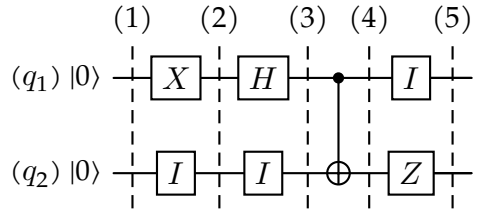


Figura 2.5: *Circuito cuántico con puertas I y capas numeradas*

A continuación, se desarrollará el resultado con este enfoque, comprobando que ofrece una igual solución.

$$\left(\overbrace{\begin{pmatrix}0&1\\1&0\end{pmatrix}}^{X} \otimes \overbrace{\begin{pmatrix}1&0\\0&1\end{pmatrix}}^{I}\right)\underbrace{\begin{pmatrix}1\\0\\0\\0\end{pmatrix}}_{|00\rangle} = \overbrace{\begin{pmatrix}0&0&1&0\\0&0&0&1\\1&0&0&0\\0&1&0&0\end{pmatrix}}^{X\otimes I}\underbrace{\begin{pmatrix}1\\0\\0\\0\end{pmatrix}}_{|00\rangle} = \underbrace{\begin{pmatrix}0\\0\\1\\0\end{pmatrix}}_{|10\rangle} \tag{2.7}$$

$$\left(\overbrace{\frac{1}{\sqrt{2}}\begin{pmatrix}1&1\\1&-1\end{pmatrix}}^{H} \otimes \overbrace{\begin{pmatrix}1&0\\0&1\end{pmatrix}}^{I}\right)\underbrace{\begin{pmatrix}0\\0\\1\\0\end{pmatrix}}_{|10\rangle} = \overbrace{\frac{1}{\sqrt{2}}\begin{pmatrix}1&0&1&0\\0&1&0&1\\1&0&-1&0\\0&1&0&-1\end{pmatrix}}^{H\otimes I}\underbrace{\begin{pmatrix}0\\0\\1\\0\end{pmatrix}}_{|10\rangle} = \underbrace{\frac{1}{\sqrt{2}}\begin{pmatrix}1\\0\\-1\\0\end{pmatrix}}_{|-0\rangle} \tag{2.8}$$

$$\overbrace{\begin{pmatrix}1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0\end{pmatrix}}^{CNOT}\underbrace{\frac{1}{\sqrt{2}}\begin{pmatrix}1\\0\\-1\\0\end{pmatrix}}_{|-0\rangle} = \underbrace{\frac{1}{\sqrt{2}}\begin{pmatrix}1\\0\\0\\-1\end{pmatrix}}_{|\Phi^-\rangle} \tag{2.9}$$

$$\left(\overbrace{\begin{pmatrix}1&0\\0&1\end{pmatrix}}^{I} \otimes \overbrace{\begin{pmatrix}1&0\\0&-1\end{pmatrix}}^{Z}\right)\underbrace{\frac{1}{\sqrt{2}}\begin{pmatrix}1\\0\\0\\-1\end{pmatrix}}_{|\Phi^-\rangle} = \overbrace{\begin{pmatrix}1&0&0&0\\0&-1&0&0\\0&0&1&0\\0&0&0&-1\end{pmatrix}}^{I\otimes Z}\underbrace{\frac{1}{\sqrt{2}}\begin{pmatrix}1\\0\\0\\-1\end{pmatrix}}_{|\Phi^-\rangle} = \underbrace{\frac{1}{\sqrt{2}}\begin{pmatrix}1\\0\\0\\1\end{pmatrix}}_{|\Phi^+\rangle} \tag{2.10}$$

La ejecución completa del circuito puede verse como una sola expresión matricial, concatenando los productos de matrices en el orden correcto (el inverso a la aparición en el diagrama):

$$\underbrace{\begin{pmatrix}1&0&0&0\\0&-1&0&0\\0&0&1&0\\0&0&0&-1\end{pmatrix}}_{I\otimes Z}\overbrace{\begin{pmatrix}1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0\end{pmatrix}}^{CNOT}\underbrace{\frac{1}{\sqrt{2}}\begin{pmatrix}1&0&1&0\\0&1&0&1\\1&0&-1&0\\0&1&0&-1\end{pmatrix}}_{H\otimes I}\overbrace{\begin{pmatrix}0&0&1&0\\0&0&0&1\\1&0&0&0\\0&1&0&0\end{pmatrix}}^{X\otimes I}\underbrace{\begin{pmatrix}1\\0\\0\\0\end{pmatrix}}_{|00\rangle} \tag{2.11}$$

Por asociatividad, es también posible realizar el producto entre las matrices y luego aplicar el producto entre la matriz resultante y el vector de estado inicial.

2.2.2. Notación ket

Otro forma de representar la ejecución del circuito es mediante el uso de la notación ket. Al conocer cómo se comporta cada uno de los operadores para los estados básicos (que es justamente la información contenida en su representación matricial), es posible describir el comportamiento de cualquier otro estado, por linealidad.

Nota

A continuación, se añade un recordatorio sobre el efecto de las puertas involucradas en el circuito de ejemplo expresado en notación ket:

$$\alpha_0 |0\rangle + \alpha_1 |1\rangle \xrightarrow{I} \alpha_0 |0\rangle + \alpha_1 |1\rangle$$
$$\alpha_0 |0\rangle + \alpha_1 |1\rangle \xrightarrow{X} \alpha_1 |0\rangle + \alpha_0 |1\rangle$$
$$\alpha_0 |0\rangle + \alpha_1 |1\rangle \xrightarrow{H} \frac{\alpha_0 + \alpha_1}{\sqrt{2}} |0\rangle + \frac{\alpha_0 - \alpha_1}{\sqrt{2}} |1\rangle$$
$$\alpha_0 |0\rangle + \alpha_1 |1\rangle \xrightarrow{Z} \alpha_0 |0\rangle - \alpha_1 |1\rangle$$
$$\alpha_{00}|00\rangle + \alpha_{01}|01\rangle + \alpha_{10}|10\rangle + \alpha_{11}|11\rangle \xrightarrow{CNOT} \alpha_{00}|00\rangle + \alpha_{01}|01\rangle + \alpha_{11}|10\rangle + \alpha_{10}|11\rangle$$

En la ejecución del primer paso, se podrían simplemente realizar las operaciones sobre el primer cúbit de manera independiente. Primero $X |0\rangle = |1\rangle$ y posteriormente $H |1\rangle = |-\rangle$. Sin embargo, como en el caso matricial, también es posible desarrollar el cómputo utilizando el enfoque del vector de estado completo desde el principio. En ese caso, se propone la siguiente expresión en notación ket que define el circuito:

$$(I \otimes Z)CNOT(H \otimes I)(X \otimes I)\ |00\rangle \tag{2.12}$$

El desarrollo de esta expresión es bastante intuitivo. Como en el caso de al representación matricial, la entrada del circuito ($|00\rangle$) se encuentra a la derecha del todo y los operadores que primero se efectúan sobre este son los situados inmediatamente a la izquierda del mismo.

$$(X \otimes I)\ |00\rangle = (X |0\rangle) \otimes (I |0\rangle) = |1\rangle \otimes |0\rangle = |10\rangle \tag{2.13}$$

$$(H \otimes I)\ |10\rangle = (H |1\rangle) \otimes (I |0\rangle) = |-\rangle \otimes |0\rangle = |-0\rangle \tag{2.14}$$

$$\begin{aligned} CNOT\ |-0\rangle &= CNOT\left((\frac{1}{\sqrt{2}} |0\rangle - \frac{1}{\sqrt{2}} |1\rangle) \otimes |0\rangle\right) \\ &= \frac{1}{\sqrt{2}} |00\rangle - \frac{1}{\sqrt{2}} |11\rangle \end{aligned} \tag{2.15}$$

$$\begin{aligned}(I \otimes Z)\,(\frac{1}{\sqrt{2}}\,|00\rangle - \frac{1}{\sqrt{2}}\,|11\rangle) &= (\frac{1}{\sqrt{2}}(I \otimes Z)\,|00\rangle - \frac{1}{\sqrt{2}}(I \otimes Z)\,|11\rangle) \\ &= \frac{1}{\sqrt{2}}\,|00\rangle + \frac{1}{\sqrt{2}}\,|11\rangle\end{aligned} \tag{2.16}$$

2.2.3. Notación algebraica

Por último, se expone la notación algebraica y cómo desarrollar la evaluación de circuitos con ella. Este enfoque está motivado por la comodidad de trabajar con los operadores de manera similar como lo hace la notación ket, intentando simplificar las expresiones resultantes, mientras se acerca al flujo de ejecución del circuito. Al contrario que en las otras dos notaciones expuestas (matricial y ket), en la notación algebraica el estado de los cúbits es lo primero que aparece en la expresión, seguido de las operaciones en orden de ejecución. Esto es algo muy similar a lo que muestra el diagrama del circuito. A cada puerta se le deberán indicar los subíndices de objetivo (aquellos índices de los cúbits en los que se aplica la puerta) y, si los hubiera, los superíndices de los cúbits de control.

Por ejemplo, el circuito de la Fig. 2.4 puede ser expresado como:

$$|00\rangle\, X_1 H_1 X_2^1 Z_2 \tag{2.17}$$

De esta manera, el estado inicial es objeto de las operaciones practicadas de izquierda a derecha, y el funcionamiento de estas queda descrito para los estados base, y por linealidad desarrolla su comportamiento en la instancia específica. Una forma de realizar la evaluación del circuito según las herramientas que ofrece la notación puede ser:

$$|00\rangle \xmapsto{X_1} |10\rangle \tag{2.18}$$

$$\xmapsto{H_1} \frac{1}{\sqrt{2}}\,|00\rangle - \frac{1}{\sqrt{2}}\,|10\rangle \tag{2.19}$$

$$\xmapsto{X_2^1} \frac{1}{\sqrt{2}}\,|00\rangle - \frac{1}{\sqrt{2}}\,|11\rangle \tag{2.20}$$

$$\xmapsto{Z_2} \frac{1}{\sqrt{2}}\,|00\rangle + \frac{1}{\sqrt{2}}\,|11\rangle \tag{2.21}$$

El uso de circuitos cuánticos es esencial para la elaboración de algoritmos cuánticos. En este capítulo se introducirán algunos algoritmos cuánticos básicos que ofrecen una mejora con respecto a su análogo clásico, enfocándose en la familia de algoritmos basados en oráculos.

2.3. Componentes básicos para algoritmos cuánticos

Los algoritmos cuánticos básicos basan su ventaja cuántica en la propiedad de la devolución de fase, la cual es posible gracias al paralelismo cuántico. El algoritmo de

Shor basa su ventaja en esencia gracias también a la devolución de fase, por lo que es de gran interés dedicarle un tiempo para entender este concepto de buena manera.

2.3.1. Paralelismo cuántico

El paralelismo cuántico es la capacidad de un ordenador cuántico para evaluar $f(x)$ para todos los distintos valores de x de forma simultánea, donde $f : \{0,1\}^n \to \{0,1\}$.

Esto se puede conseguir mediante el oráculo U_f que no es más que una puerta cuántica especial. El oráculo U_f opera de la siguiente manera sobre dos registros, el registro de entrada $|x\rangle$ y un cúbit auxiliar $|y\rangle$, donde $y \in \{0,1\}$:

$$(|x\rangle \otimes |y\rangle)(U_f)_{1:n+1} = |x\rangle \otimes |y \oplus f(x)\rangle \tag{2.22}$$

2.1 Ejercicio

Demostrar que U_f es Hermítica, es decir, $U_f^2 = I$.
Pista: Comprobar el efecto de U_f al aplicarlo dos veces sobre $|x\rangle \otimes |y\rangle$.

Al aplicar el oráculo U_f sobre el estado $\frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle$, por la simple linealidad de la operación se obtiene una combinación lineal de los estados de todas las imágenes de f.

$$\begin{aligned}\left(\frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle \otimes |y\rangle\right)(U_f)_{1:n+1} &= \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} (|x\rangle \otimes |y\rangle)(U_f)_{1:n+1} \\ &= \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle \otimes |y \oplus f(x)\rangle\end{aligned} \tag{2.23}$$

2.2 Ejercicio

Demostrar que $\frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle$ es un vector de estado válido.
Pista: Comprobar la propiedad $\sum_i \|\alpha_i\|^2 = 1$.

Nota

Es posible obtener el estado $\frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle$ aplicando una capa de puertas H a un registro de n cúbits:

$$|0\rangle^{\otimes n} H_{1:n} = |+\rangle^{\otimes n} = \left(\frac{|0\rangle + |1\rangle}{\sqrt{2}}\right)^{\otimes n} = \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle$$

Combinando ambos resultados, si $y = 0$, se obtendrían explícitamente todas las

entradas de f junto con sus imágenes en el cúbit auxiliar en una combinación lineal equiprobable.

$$|0\rangle^{\otimes n+1} H_{1:n}(U_f)_{1:n+1} = \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle \otimes |f(x)\rangle \tag{2.24}$$

Es lógico que pensar que de esta manera es posible obtener todas las evaluaciones de f de una sola vez. Sin embargo, para que el usuario pueda realmente saber estas evaluaciones, necesitará medir este estado. Al realizar esta acción, el estado del cúbit colapsara a un único estado $f(x_i)$ de manera uniforme (con probabilidad $\frac{1}{2^n}$ cada uno). Obteniendo de esta manera una única evaluación de f.

Una vez dicho esto, parece que en realidad el paralelismo cuántico carece de interés práctico para ganar una ventaja computacional. Sin embargo, como ya se ha venido comentando, sienta las bases para poder realizar la devolución de fase, herramienta que se verá que sí ofrece una ventaja significativa en muchos escenarios.

2.3.2. Devolución de fase

El fenómeno de devolución de fase, también conocido en inglés como *phase kickback*, se produce en realidad como una extensión directa del paralelismo cuántico. Para poder aprovechar la posible mejora del paralelismo cuántico se debe usar el estado $|-\rangle$ en el cúbit auxiliar. Aplicando U_f al estado $|x\rangle \otimes |-\rangle$:

$$|x\rangle \otimes |-\rangle (U_f)_{1:n} = |x\rangle \otimes \frac{1}{\sqrt{2}}(|0 \oplus f(x)\rangle - |1 + f(x)\rangle) = (-1)^{f(x)} |x\rangle |-\rangle \tag{2.25}$$

Haciendo esto, no se obtiene información de $f(x)$ en el segundo registro del cúbit auxiliar, sino en la fase $(-1)^{f(x)}$ del primer registro. Ahora aplicando esto para el estado $\frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle$ en el primer registro y el estado $|-\rangle$ en el segundo registro:

$$\frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle \otimes |-\rangle (U_f)_{1:n+1} = \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} (-1)^{f(x)} |x\rangle |-\rangle \tag{2.26}$$

De esta manera, la información de las imágenes $f(x)$ se transfiere a la fase del primer registro. A continuación se introducen varios algoritmos cuánticos que hacen uso de este fenómeno.

> **Nota**
>
> Dado un circuito clásico para calcular f, existe un circuito cuántico de eficiencia comparable que implemente U_f.

2.4. Algoritmo de Deutsch

El Algoritmo de Deutsch, propuesto por el físico británico David Deutsch en 1985, es un hito fundacional en la teoría de la computación cuántica.

Su principal contribución no fue resolver un problema práctico complejo, sino demostrar por primera vez que un ordenador cuántico podría realizar una tarea más deprisa que cualquier ordenador clásico. El algoritmo de Deutsch se centra en encontrar si una función de 1 bit es constante o balanceada, es decir, resolver el siguiente problema:

Entrada: $f : \{0,1\} \to \{0,1\}$.

Salida: 0 si $f(0) = f(1)$, 1 si $f(0) \neq f(1)$.

Para resolver este problema clásicamente, se deben hacer dos consultas al oráculo de f, uno para obtener $f(0)$ y otro para $f(1)$, para luego comparar ambos valores. Sin embargo, de manera cuántica, solo se debe realizar una "consulta" al oráculo cuántico U_f de f. El circuito cuántico asociado al algoritmo de Deutsch se introduce a continuación:

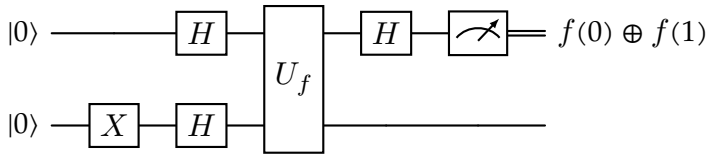


Figura 2.6: *Circuito de Deutsch*

Y en notación algebraica:

$$\boxed{|00\rangle\, X_2 H_{1:2} (U_f)_{1:2} H_1 M_1}$$

A continuación se desarrollan los estados intermedios del circuito para demostrar el funcionamiento del mismo:

$$|00\rangle \xmapsto{X_2} |10\rangle \tag{2.27}$$

$$\xmapsto{H_{1:2}} |+-\rangle = \frac{1}{\sqrt{2}}(|0\rangle|-\rangle + |1\rangle|-\rangle) \tag{2.28}$$

$$\xmapsto{(U_f)_{1:2}} \frac{1}{\sqrt{2}}((-1)^{f(0)}|0\rangle|-\rangle + (-1)^{f(1)}|1\rangle|-\rangle) \tag{2.29}$$

$$\xmapsto{H_1} \frac{1}{\sqrt{2}}((-1)^{f(0)}|+\rangle|-\rangle + (-1)^{f(1)}|-\rangle|-\rangle) \tag{2.30}$$

$$= \frac{1}{2}\left[\left((-1)^{f(0)} + (-1)^{f(1)}\right)|0\rangle + \left((-1)^{f(0)} - (-1)^{f(1)}\right)|1\rangle\right] \otimes |-\rangle = |\psi\rangle \tag{2.31}$$

Ahora se estudian los dos posibles escenarios del problema:

- **Función Constante** ($f(0) \oplus f(1) = 0$):
  Si $f(0) = f(1)$, entonces las fases son iguales: $(-1)^{f(0)} = (-1)^{f(1)} = \lambda$.

Sustituyendo λ en la expresión:

$$\begin{aligned}|\psi\rangle &= \frac{1}{2}\left[(\lambda+\lambda)\,|0\rangle + (\lambda-\lambda)\,|1\rangle\right] \otimes |-\rangle \\ &= \frac{1}{2}\left[2\lambda\,|0\rangle + 0\,|1\rangle\right] \otimes |-\rangle \\ &= \lambda\,|0\rangle \otimes |-\rangle \\ &= \lambda\,|f(0) \oplus f(1)\rangle \otimes |-\rangle\end{aligned}$$

El estado del primer qubit es $|0\rangle$, que corresponde a $|f(0) \oplus f(1)\rangle$ cuando el resultado es 0.

- **Función Balanceada** $(f(0) \oplus f(1) = 1)$:
  Si $f(0) \neq f(1)$, entonces las fases son opuestas: $(-1)^{f(0)} = -\left((-1)^{f(1)}\right)$. Se puede expresar esto como: $(-1)^{f(1)} = -(-1)^{f(0)}$. Si $\lambda = (-1)^{f(0)}$, entonces $(-1)^{f(1)} = -\lambda$.
  Sustituyendo λ y $-\lambda$ en la expresión:

$$\begin{aligned}|\psi\rangle &= \frac{1}{2}\left[(\lambda+(-\lambda))\,|0\rangle + (\lambda-(-\lambda))\,|1\rangle\right] \otimes |-\rangle \\ &= \frac{1}{2}\left[0\,|0\rangle + 2\lambda\,|1\rangle\right] \otimes |-\rangle \\ &= \lambda\,|1\rangle \otimes |-\rangle \\ &= \lambda\,|f(0) \oplus f(1)\rangle \otimes |-\rangle\end{aligned}$$

  El estado del primer qubit es $|1\rangle$, que corresponde a $|f(0) \oplus f(1)\rangle$ cuando el resultado es 1.

Dado que en ambos casos el primer qubit revela el resultado de la operación $f(0) \oplus f(1)$ y el segundo qubit es siempre el estado auxiliar $|-\rangle$, se ignora la fase global λ (ya que no es medible) y el qubit auxiliar $|-\rangle$. El estado resultante sería el siguiente:

$$|\psi\rangle \propto |f(0) \oplus f(1)\rangle \otimes |-\rangle$$

Y por tanto, la medición del primer qubit revela directamente la propiedad deseada de la función f haciendo tan solo una única.

> **Nota**
>
> Si se tienen dos vectores de estado $|\psi\rangle$ y $|\phi\rangle$ que solo difieren en una fase global (que no es medible) de denota como:
>
> $$|\psi\rangle \propto |\phi\rangle$$

2.5. Algoritmo de Deutsch-Jozsa

El Algoritmo de Deutsch-Jozsa, propuesto por David Deutsch y Richard Jozsa en 1992, es una generalización directa del Algoritmo de Deutsch de 1985 y es reconocido por haber proporcionado la primera demostración clara de una separación exponencial en la complejidad de un problema entre computadoras cuánticas y clásicas. El algoritmo de Deutsch-Jozsa generaliza el dominio de la función de entrada $f(x)$. En el algoritmo de Deutsch, el espacio de salida era igual al espacio de llegada $(\{0,1\})$. Sin embargo, para este algoritmo de Deutsch-Jozsa, el espacio de salida se cambia a $\{0,1\}$ para cualquier $n \in \mathbb{N}$.

Entrada: $f : \{0,1\}^n \to \{0,1\}$.

Promesa: f es constante o balanceada.

Salida: $\underbrace{0 \ldots 0}_{n}$ si f es constante, cualquier otro resultado si f es balanceada.

Para resolver este algoritmo clásicamente se deben realizar $2^{n-1} + 1$ evaluaciones de f. Como se puede observar, el numero de consultas clásicas de f ha aumentado exponencialmente. Esto tiene sentido pues el cardinal del dominio también ha aumentado exponencialmente. Y por esto tendría sentido que el algoritmo cuántico también tenga que aumentar el número de consultas en gran medida, pero como se verá a continuación, no es necesario. El algoritmo de Deutsch-Jozsa es capaz de resolverlo con una única evaluación del oráculo U_f. El circuito cuántico asociado al algoritmo de Deutsch-Jozsa se introduce en la Figura 2.7.

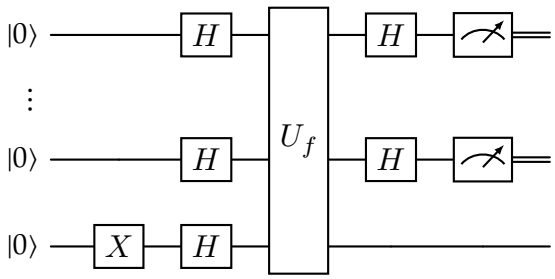


Figura 2.7: *Circuito de Deutsch-Jozsa*

Y en notación algebraica:

$$\boxed{|0\rangle^{\otimes n+1} X_{n+1} H_{1:n+1} (U_f)_{1:n+1} H_{1:n} M_{1:n}}$$

A continuación se procede a desarrollar los estados intermedios del circuito para demostrar el funcionamiento del algoritmo. Aunque primero es conveniente desarrollar dos ideas que facilitarán el desarrollo del algoritmo.

Nota

Dado que la acción de la puerta H sobre los estados básicos es la siguiente:

$$|0\rangle H_1 = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle \quad \wedge \quad |1\rangle H_1 = \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle$$

Se puede ver que la acción de la puerta H sobre cualquier $a \in \{0,1\}$ se puede ver como lo siguiente:

$$|a\rangle H_1 = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}(-1)^a|1\rangle = \frac{1}{\sqrt{2}}\sum_{b\in\{0,1\}}(-1)^{ab}|b\rangle$$

Gracias a esta expresión de la acción de la puerta H sobre los estados básicos, es posible generalizar la acción de esta misma puerta para un registro un estado básico de n cúbits cualquiera $|x\rangle$, $\forall x \in \{0,1\}^n$:

$$\begin{aligned}
|x\rangle H_{1:n} &= |x_n \dots x_2 x_1\rangle H_{1:n} \\
&= (|x_n\rangle H_1) \otimes \cdots \otimes (|x_1\rangle H_1) \\
&= \left(\frac{1}{\sqrt{2}}\sum_{y_n\in\{0,1\}}(-1)^{x_n y_n}|y_n\rangle\right) \otimes \cdots \otimes \left(\frac{1}{\sqrt{2}}\sum_{y_1\in\{0,1\}}(-1)^{x_1 y_1}|y_1\rangle\right) \\
&= \frac{1}{\sqrt{2^n}}\sum_{y\in\{0,1\}^n}(-1)^{x\cdot y}|y\rangle
\end{aligned} \tag{2.32}$$

Donde se denota a $x \cdot y$ como el *binary dot product*, es decir, $\sum_{1\le i\le n} x_i \cdot y_i$.

Con estas herramientas ya se está en condiciones de estudiar los estados intermedios del circuito de Deutsch-Jozsa:

$$|0\rangle^{\otimes n+1} \xrightarrow{X_{n+1}} |0\rangle^{\otimes n} \otimes |1\rangle \tag{2.33}$$

$$\xrightarrow{H_{1:n+1}} |+\rangle^{\otimes n} \otimes |-\rangle = \frac{1}{\sqrt{2^n}}\sum_{x\in\{0,1\}^n}|x\rangle \otimes |-\rangle \tag{2.34}$$

$$\xrightarrow{(U_f)_{1:n+1}} \frac{1}{\sqrt{2^n}}\sum_{x\in\{0,1\}^n}(-1)^{f(x)}|x\rangle \otimes |-\rangle \tag{2.35}$$

$$\xrightarrow{H_{1:n}} \frac{1}{2^n}\sum_{x\in\{0,1\}^n}\sum_{y\in\{0,1\}^n}(-1)^{f(x)+x\cdot y}|y\rangle \otimes |-\rangle \tag{2.36}$$

Por tanto, la probabilidad de medir 0 en el primer registro ($|y\rangle = |0\rangle^{\otimes n}$) es la siguiente:

$$\left\|\frac{1}{2^n}\sum_{x\in\{0,1\}^n}(-1)^{f(x)}\right\|^2 = \begin{cases} 1 & \text{si } f \text{ es constante} \\ 0 & \text{si } f \text{ es balanceada} \end{cases} \tag{2.37}$$

Si f es constante, entonces o bien $f(x) = 0$ $\forall x \in \{0,1\}^n$, en cuyo caso el valor de la

suma sería 2^n, o $f(x) = 1\ \forall x \in \{0,1\}^n$, en cuyo caso el valor de la suma es -2^n.

Si por el contrario, f es una función balanceada, la mitad de las cadenas tienen como imagen 0 y la otra mitad 1, anulando de esta manera el sumatorio.

2.6. Algoritmo de Bernstein-Vazirani

El Algoritmo de Bernstein-Vazirani, publicado por Ethan Bernstein y Umesh Vazirani en 1997, es considerado un hito crucial en la historia de la computación cuántica, ya que extendió y robusteció la demostración de la ventaja cuántica iniciada por el Algoritmo de Deutsch-Jozsa. El objetivo de este algoritmo es encontrar la cadena binaria s para la función oráculo dada $f(x) = \sum_i x_i \cdot s_i \mod 2$, donde mód 2 indica el resto dividiendo por 2 (0 si es par o 1 si es impar). La manera clásica más eficiente de resolver este problema sería realizando n llamadas a este oráculo, "consultando" el valor bit a bit como se observa en la Ec. 2.38:

$$\begin{cases} f_s(100\ldots0) = s_n \\ f_s(010\ldots0) = s_{n-1} \\ f_s(001\ldots0) = s_{n-2} \\ \quad\vdots \\ f_s(000\ldots1) = s_1 \end{cases} \quad , \quad s = s_n s_{n-1} s_{n-2} \cdots s_1 \tag{2.38}$$

Por tanto, el algoritmo clásico para resolver este problema haciendo n consultas a la función f. En cambio, su equivalente cuántico es capaz de resolverlo haciendo tan solo una única consulta al oráculo U_f. Es decir, el algoritmo cuántico lo resuelve al instante para cualquier tamaño de clave a encontrar. En la Figura 2.8 se muestra el circuito cuántico del algoritmo de Bernstein-Vazirani.

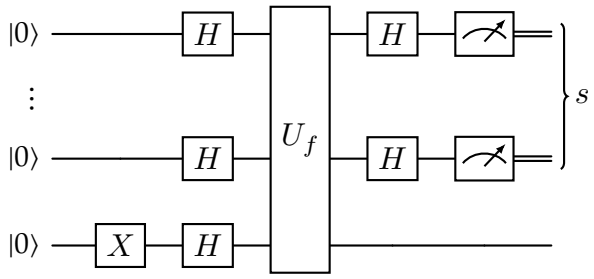


Figura 2.8: *Circuito de Bernstein-Vazirani*

Notación algebraica del circuito de Bernstein-Vazirani:

$$\boxed{|0\rangle^{\otimes n+1}\, X_{n+1} H_{1:n+1} (U_f)_{1:n+1} H_{1:n} M_{1:n}}$$

Este problema es un muy buen ejemplo de las ventajas que aporta la computación

cuántica para ciertos problemas. A continuación se procede a explicar el funcionamiento del circuito:

$$|0\rangle^{\otimes n+1} \xmapsto{X_{n+1}} |0\rangle^{\otimes n} \otimes |1\rangle \tag{2.39}$$

$$\xmapsto{H_{1:n+1}} |+\rangle^{\otimes n} \otimes |-\rangle = \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle \otimes |-\rangle \tag{2.40}$$

$$\xmapsto{(U_f)_{1:n+1}} \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} (-1)^{f(x)} |x\rangle \otimes |-\rangle \tag{2.41}$$

$$\xmapsto{H_{1:n}} \frac{1}{2^n} \sum_{x\in\{0,1\}^n} \sum_{y\in\{0,1\}^n} (-1)^{f(x)+x\cdot y} |y\rangle \otimes |-\rangle \tag{2.42}$$

$$= \frac{1}{2^n} \sum_{x,y\in\{0,1\}^n} (-1)^{f(x)+x\cdot y} |y\rangle \otimes |-\rangle \tag{2.43}$$

Ahora se puede simplificar la expresión obtenida. Teniendo en cuenta que $f(x) = s \cdot x$, se puede escribir:

$$(-1)^{f(x)+x\cdot y} = (-1)^{s\cdot x+x\cdot y} = (-1)^{(s\cdot x)\oplus(x\cdot y)} \tag{2.44}$$

Ahora, teniendo en cuenta la siguiente propiedad:

$$\begin{aligned}(s\cdot x)\oplus(x\cdot y) &= (s_1x_1)\oplus\cdots\oplus(s_nx_n)\oplus(y_1x_1)\oplus\cdots\oplus(y_nx_n)\\ &= (s_n\oplus y_n)x_n\oplus\cdots\oplus(s_1\oplus y_1)x_1\\ &= (s\oplus y)\cdot x\end{aligned} \tag{2.45}$$

se obtiene la siguiente expresión:

$$\begin{aligned}\frac{1}{2^n}\sum_{x,y\in\{0,1\}^n}(-1)^{f(x)+x\cdot y}|y\rangle\otimes|-\rangle &= \frac{1}{2^n}\sum_{x,y\in\{0,1\}^n}(-1)^{(s\oplus y)\cdot x}|y\rangle\otimes|-\rangle\\ &= \sum_{y\in\{0,1\}^n}\frac{1}{2^n}\sum_{x\in\{0,1\}^n}(-1)^{(s\oplus y)x}|y\rangle\otimes|-\rangle\end{aligned} \tag{2.46}$$

El paso final de este algoritmo se basa en la siguiente expresión:

$$\frac{1}{2^n}\sum_{x\in\{0,1\}^n}(-1)^{z\cdot x} = \begin{cases}1 & \text{si } z=0\\ 0 & \text{si } z\neq 0\end{cases}$$

Y teniendo en cuenta que $s\oplus y = \underbrace{0\cdots 0}_{n} \iff y = s$:

$$\sum_{y\in\{0,1\}^n}\frac{1}{2^n}\sum_{x\in\{0,1\}^n}(-1)^{(s\oplus y)\cdot x}|y\rangle\otimes|-\rangle = |s\rangle\otimes|-\rangle \tag{2.47}$$

Por tanto, midiendo los n primeros cúbits se obtiene la cadena binaria s con probabili-

dad 1.